UNIVERSIDADE FEDERAL DO CEARÁ

CENTRO DE CIÊNCIAS

DEPARTAMENTO DE FÍSICA

PROGRAMA DE PÓS-GRADUAÇÃO EM FÍSICA

LUAN MISAEL GOMES DE MOURA

APRENDIZAGEM DE MÁQUINA, ESTATÍSTICA APLICADA E TEORIA DOS GRAFOS: ESTUDO EM ANÁLISE ESPECTRAL E MERCADO FINANCEIRO

FORTALEZA

2019

LUAN MISAEL GOMES DE MOURA

APRENDIZAGEM DE MÁQUINA, ESTATÍSTICA APLICADA E TEORIA DOS GRAFOS: ESTUDO EM ANÁLISE ESPECTRAL E MERCADO FINANCEIRO

Dissertação apresentada ao Programa de Pós-Graduação em Física da Universidade Federal do Ceará, como requisito parcial à obtenção do título de Mestre em Física. Área de concentração: Física da Matéria Condensada.

Orientador: Prof. Dr. Carlos Lenz Cesar.

Fortaleza

2019

Dados Internacionais de Catalogação na Publicação
Universidade Federal do Ceará
Biblioteca Universitária
Gerada automaticamente pelo módulo Catalog, mediante os dados fornecidos pelo(a) autor(a)

M887a Moura, Luan Misael Gomes de.
Aprendizagem de máquina, estatística aplicada e teoria dos grafos : estudo em análise espectral e mercado financeiro / Luan Misael Gomes de Moura. – 2019.
218 f. : il. color.

Dissertação (mestrado) – Universidade Federal do Ceará, Centro de Ciências, Programa de Pós-Graduação em Física, Fortaleza, 2019.
Orientação: Prof. Dr. Carlos Lenz Cesar.

1. Aprendizagem de máquina. 2. Estatística. 3. Teoria dos grafos. 4. Econofísica. I. Título.
CDD 530

LUAN MISAEL GOMES DE MOURA

APRENDIZAGEM DE MÁQUINA E ESTATÍSTICA APLICADA: ESTUDO EM ANÁLISE ESPECTRAL E MERCADO FINANCEIRO

Tese ou Dissertação apresentada ao Programa de Pós-Graduação em Física da Universidade Federal do Ceará, como requisito parcial à obtenção do título de Mestre em Física. Área de concentração: Física da Matéria Condensada.

Aprovada em: 11/03/2019.

BANCA EXAMINADORA

_____

Prof. Dr. Carlos Lenz Cesar (Orientador)
Universidade Federal do Ceará (UFC)

_____

Prof. Dr. Francisco Nepomuceno Filho
Universidade Federal do Ceará (UFC)

_____

Prof. Dr. José Maria Ferreira Jardim Da Silveira
Universidade Estadual de Campinas (UNICAMP)

Ao Universo, meu Mestre e meu Padrinho.

Aos meus pais, família e companheira.

AGRADECIMENTOS

O presente trabalho foi realizado com apoio da Coordenação de Aperfeiçoamento de Pessoal de Nível Superior - Brasil (CAPES) - Código de Financiamento 001. Agradeço então à CAPES e Petrobrás, pelo apoio financeiro.

À CAPES e Petrobrás, pelo apoio financeiro. Ao professor Alejandro Ayala, à Rosélia Machado, André Auto, Saulo Reis, Renan Landim, William Pachoal.

À meu orientador, Carlos Lenz César pelo excelente direcionamento e aos professores da banca Franscisco Nepomuceno Filho e José Maria Silveira.

Ao meu pai Sebastião Moura e mãe Denis Moura, meu irmão Lucas Moura, irmã Débora Moura, família e minha companheira Cecília Mesquita pelo apoio.

Aos colegas de mestrado e graduação Débora, Matheus, Michelângelo, Michel Rodrigues e Levi.

Aos colegas do LASCO roots, Kevin, Mapse, Jeff, o comandante e os colegas dos sistemas complexos Nena, João Paulo, Morais, Wagner.

Aos antigos Eliezers Mardônio França, Lucas Miranda e Daniel Brito.

Aos brothers da república em Barão que me ajudaram nos momentos finais da criação deste texto, Virsão, Fidel, Mineiro, Jimmy, Bruno e Manoel.

Ao Samuel Lima por me ajudar a abrir meu crânio e Igor Bragaia pelas conexões.

À UFC e UNICAMP por fornecer os meios necessários para esta pesquisa.

Ao Alberto Cerqueira, Paulo Cassiano e César Ximenes por me colocarem no eixo no meu momento mais difícil.

À igreja Céu da Flor do Cajueiro e aos irmão Rodrigo, Zerivan, Sidarta, Junin, Márcio e Nirvando. Ao Céu de Maria e os irmãos Giovani e Carlos.

Ao Glauco, ao Padrinho Sebastião, ao Mestre Irineu, à Rainha da Floresta e toda irmandade. Aos seres divinos, à Força e às plantas de poder.

Ao som do bandolim.

“If you torture the data long enough, it will confess.”

– Ronald Coase

RESUMO

A automatização de análise de dados espectrais é uma necessidade do projeto Física do petróleo em meios porosos. Através dos algoritmos de aprendizagem de máquina (subcampo de ciência da computação) é possível classificar dados onde a máquina é capaz aprender os paramêtros corretos em modelos de classificação de minerais e óleos. O método vai além, sendo aplicável em diferentes áreas. O estudo de grafos possui conexão intrínseca na definição da arquitetura de tais algoritmos e a partir de grafos de *minimum spanning trees* (MST) visualizamos e agrupamos os dados. Também aplicamos MST em ações da bolsa de valores americana. Dentro do mercado financeiro desenvolvemos ferramentais estatísticos que descrevem o movimento das ações e a precificação de opções americanas e europeias. Os métodos estatísticos e de aprendizagem de máquina são utilizados em tarefas de predição e inferência. Na inferência queremos descrever o padrão de um conjunto de dados através de um modelo probabilistico que é o foco maior da estatística. A predição é a habilidade de classificar corretamente amostras desconhecidas. Os modelos preditivos de aprendizagem reforçada são criados de forma que o algoritmo encontra padrões de alta complexidade difíceis de serem reconhecidos por humanos. Com tudo isso em mente, o principal objetivo deste trabalho é construir um conjunto de ferramentas de análise automática baseadas em gráfos, aprendizagem de máquina e estatística podendo ser aplicáveis em diversas áreas.

Palavras-chave: Aprendizagem de máquina. Estatística. Teoria dos grafos , Econofísica.

ABSTRACT

The automation of spectral data analysis is a necessity of the project Physics of petroleum in porous media. Through machine learning algorithms (subfield of computer science) it is possible to classify data where the machine is able to learn the correct parameters of classification models of minerals and oils. The method goes beyond, being applicable in different areas. The study of graphs has intrinsic connection in the definition of the architecture of such algorithms and from graphs of minimum spanning trees (MST) we visualize and group the data. We also apply MST in shares of the American stock exchange. Within the financial market we develop statistical tools that describe the movement of stocks and the pricing of American and European options. Statistical and machine learning methods are used in prediction and inference tasks. For inference models, we want to describe the pattern of a data set through a probabilistic model which is the major focus of statistics. Prediction is the ability to correctly sort unfamiliar samples. Predictive models of deep learning are created in such a way that the algorithm finds high complexity patterns that are hard for a human to identify. With all that in mind, the main objective of this work is to build a set of automatic analysis tools based on graphs, machine learning and statistics that can have great applicability in many areas.

Keywords: Machine learning, Statistics, Graph Theory, Econophysics.

SUMÁRIO

1 INTRODUÇÃO

O Departamento de Física da UFC conseguiu aprovar vários projetos na área de petróleo junto à Petrobrás e Petrogal, já ativos, e Sinochem, aprovado, mas ainda não implementado. Não estamos envolvidos na prospecção de petróleo em si, ou seja, em localizar posições com maior probabilidade de encontrar petróleo, mas apenas na otimização de extração de óleo em poços já existentes e ativos. O objetivo do grupo como um todo é fornecer simulações que permitam que os poços sejam explorados da forma mais eficiente possível, através de diversas simulações. Essas simulações, entretanto, necessitam dos parâmetros relativos à rocha e ao fluído como input. Por isso, um subgrupo experimental ficou responsável pelas caracterizações, fundamentalmente, de rochas e óleos, usando várias técnicas como Espectroscopia Raman, Fluorescência, Microscopia Eletrônica de Varredura com EDS [Energy Dispersive x-Ray Spectroscopy], raios-x, etc, em amostras fornecidas pelas petrolíferas das rochas retiradas das perfurações, e dos óleos extraídos dos poços.

Dentro desse contexto, o objetivo dessa proposta é o desenvolvimento de metodologias automáticas para a análise do grande número de dados gerados pelo grupo experimental.

Para ter uma ideia da quantidade de dados, em uma mesma rocha pode-se fazer 4 a 5 mapas com 30×30 pontos – 4.500 espectros por rochas – cada espectro contendo da ordem de 1000 pontos – 5.400.000 pontos incluindo ruído – e, dentro do projeto Petrobrás teremos da ordem de 1.000 amostras para analisar. A análise de tais dados deve ser automatizada utilizando métodos de aprendizagem de máquina e métodos utilizados em química analítica (Resolução de curvas multivariada).

O reconhecimento de padrões em dados através do uso de algoritmos de aprendizagem de máquina proporciona a classificação dos dados em diferentes categorias [41]. Aprendizagem de máquina (cujo termo popular do inglês é *machine learning*, poderemos nos referir simplesmente como ML) é um subcampo da ciência da computação advindo do estudo de reconhecimento de padrões e da teoria do aprendizado computacional em **inteligência artificial**. Definiu-se aprendizagem de máquina como o "campo de estudo que dá aos computadores a habilidade de aprender sem serem explicitamente programados" [67]. Os algoritmos de machine learning aprendem com seus erros diferentemente de outros algoritmos que seguem inflexíveis e estáticas instruções programadas. A partir de uma fonte de inputs de dados amostrais, os modelos construídos pelos algoritmos podem fazer previsões ou ajudar na tomada de decisão guiada pelos dados [68].

Um algoritmo recebe um conjunto de dados de *treino* que é utilizado para

encontrar os parâmetros do modelo. Verificamos a eficácia do modelo em generalizar aplicando os parâmetros encontrados na fase de aprendizagem do algoritmo em um novo conjunto de *teste*. Os conjuntos de dados podem receber pré-processamentos como a seleção das variáveis corretas derivadas dos dados para treinar o modelo evitando ruído, aumentando a eficiência e velocidade do algoritmo. Os passos de pré-processamento dos dados de treino devem ser efetuados em qualquer novo dado de teste [90]. Alguns algoritmos como as redes neurais convolucionais (CNN) são capazes de identificar as características em imagens, sons e outros tipos de dados superior ao sentido humano [74,87]. As CNN são inspiradas nos arranjos neurais do córtex visual [86]. Os problemas nos quais se sabe previamente o que representa os conjuntos de dados de treino e deseja-se generalizar para um novo conjunto é chamado de *aprendizagem supervisionada*. Diferencia-se entre problemas de *classificação* onde o output é uma categorização discreta e problemas de *regressão* onde o output é contínuo. Podemos exemplificar com a classificação de minerais, onde o input é o seu espectro Raman e o output é um vetor binário cuja entrada não-nula representa a indicação do mineral. Um problema de regressão poderia consistir na identificação de uma curva de *background* de um espectro através de um conjunto de dados de treino constituído de mínimos locais, representando um *fitting* nos dados, neste caso esta regressão não se encaixa na definição de machine learning, mas sim um método estatístico identificando o padrão de curva da amostragem.

A regressão linear iterativa que vai atualizando seus paramêtros ao longo das épocas de treino nos dá o alicerce da definição da regressão logística, redes neurais e outros algoritmos de machine learning. Apesar de ser um modelo bastante simples podemos entender sua aplicação no contexto de machine learning, por exemplo, na precificação de imóveis utilizando como input a área, localização e outras características do imóvel. Um conjunto de dados de treino já classificado é recebido e o algoritmo de regressão poderá precificar novos dados de imóveis a partir do modelo *fittado* nos dados de treino. Do ponto de vista da análise de dados:

- Aprendizagem de máquina é um algoritmo que aprende e gera modelos através dos seus dados e o programador só irá definir sua arquitetura;
- Um modelo estatístico descreve formalmente a relação entre as variáveis dos dados matematicamente.

Os dois maiores objetivos na criação de modelos de ML e estatísticos são **inferência** e **predição**. A inferência corresponde a criar modelos matemáticos para descrever o processo de geração de um conjunto de dados e como consequência, descrever os sistemas

representados pelos dados. Já a predição tem o dever de classificar corretamente novas amostras desconhecidas ou mesmo prever comportamentos futuros a partir dos dados. Os métodos estatísticos possuem um foco maior em inferência através de ferramentas advindas da teoria da probabilidade. Os algoritmos de ML são focados na predição aprendendo e encontrando padrões impossíveis de serem reconhecidos por humanos. Estes algoritmos são capazes de reconhecer interações não-lineares e a maneira com que estes geram seus modelos é tido por muitos como uma caixa-preta de difícil entendimento [144].

Na *aprendizagem não-supervisionada* o conjunto de dados de treino não está previamente identificado e deseja-se identificar similaridades nos dados e agrupá-los, ou projetar os dados de uma alta dimensão para duas ou três dimensões para facilitação da visualização ou identificação de componentes principais.

A definição dos algoritmos estatísticos e de machine learning estudos dependem de alcerces mais básicos. No Capítulo 2 definimos conceitos como Minimum Spanning Tree (MST), probabilidade, grafos, distância e produto interno. Aplicaremos MST's no estudo de espectros de fluorescência de óleos no capítulo 3 e ações americanas no capítulo 5. Os conceitos explorados nos ajudarão também a definir as principais distribuições probabilísticas utilizadas no mercado financeiro. Esses conceitos serão aplicados diretamente na classificação e agrupamento de amostragens estatísticas no capítulo 4, usando Machine Learning aplicado à espectroscopia Raman, além da conexão entre a representação da arquitetura de redes neurais artificiais com grafos. Teoria dos grafos já é aplicada em áreas como ciência da computação, matemática, telecomunicações, biologia, neurociência, entre outros [9,10,11]. Redes neurais também são aplicáveis em predição de *time series* como preços futuros de ações através de *redes neurais recorrentes* [12, 13].

As MST's são árvores que nos permitem visualizar os clusters de diferentes tipos de dados e como estes se ligam. As aplicações vão desde o mercado de ações, commodities, setores da indústria, biologia, medicina, análise espectral, entre outros. A ideia de usar covariância, coeficiente de correlação e distância de correlação para extrair ***Minimum Spanning Tree*** (MST) de similaridades através de medidas experimentais ou outras observações está atualmente em destaque na comunidade de econofísica, com aplicações desde o mercado financeiro, análise de risco, até a classificação de materiais por proximidade de suas propriedades. Junto com a tradicional metodologia de análise de componentes principais permite uma forma automática de tratamento e visualização de um conjunto enorme de dados espectrais.

Fluorescência é o sinal óptico mais intenso com comprimento de onda diferente

do feixe de luz incidente. O objetivo do grupo experimental foi, portanto, utilizar medidas de fluorescência para discriminar óleos crus de diferentes origens. Nosso objetivo principal, portanto, foi desenvolver uma metodologia de classificação de óleos através da fluorescência. Nesse aspecto, criar uma medida de distância entre os óleos e classificá-los através da MST é a forma mais natural de diferenciação das amostras de óleo. Entretanto existem desafios computacionais para esse objetivo devido à grande quantidade de dados gerados que necessitam de uma análise automática. As análises por fluorescência podem ser realizadas em apenas um ponto, no caso de amostras homogêneas, como um volume de óleo líquido, ou em até 1000×1000 pontos, no caso de imagens de fluorescência por microscopia confocal em rochas. Já nas análises de PLE [Photoluminescence Excitation Spectroscopy] por dois fótons [2p-PLE] se varre o laser de excitação entre 700 a 1000 nm em passos de 5 nm [1,25]. São, portanto, 60 espectros em cada uma das 4 componentes [amostra é fracionada em 4 partes usando diferentes solventes] de amostra de óleo. Nas imagens obtidas por fluorescência com 1000×1000 esse número é multiplicado por 10^6. Ou seja, 240 bilhões de pontos/amostra – em 100 amostras, teremos 24 trilhões de pontos. Além disso, o projeto de Física do petróleo em meios porosos requer uma automatização da classificação de um grande número de mapas Raman de minerais que vêm sempre contaminada com a fluorescência dos óleos. No capítulo 4 lidaremos com os espectros Raman, inclusive com uma metodologia para extrair a fluorescência dos espectros obtidos. Entretanto, também é importante caracterizar as fluorescências que aparecem nas rochas.

Aplicaremos a MST também no estudo de correlação de ações americanas onde poderemos investigar os aglomerados formados com seus setores econômicos. No último capítulo trabalharemos em modelos estatísticos de precificação de opções européias utilizando o modelo de Black & Scholes [B&S] e a generalização de tal modelo para opções americanas. A equação de B&S utilizada para precificar opções europeias é conhecida como fórmula de Midas, publicada em 1973 em um artigo chamado *The pricing of options and corporate liabilities* [141]. Black e Scholes desenvolveram sua equação exatamente para precificar o menor prêmio de opções europeias, na qual o titular só pode decidir exercer ou não a opção no final do contrato, em 1973, na mesma época em que surgiu a maior bolsa de opções até hoje, a Chicago Board Options Exchange [CBOE] [121], além de encontrarem uma fórmula simples, eles mostraram como o vendedor da opção podia se proteger dos riscos inerentes da operação, que foi a contribuição importante de Merton. O sucesso dessa fórmula foi tão grande que Scholes e Merton ganharam o prêmio Nobel de Economia em 1997 [Black não ganhou porque morreu em 1995] [137]. Apenas 6 meses após a publicação do trabalho a

Texas Instruments lançou no mercado uma calculadora com o anúncio: “Agora você pode encontrar o valor de Black-Scholes usando nossa calculadora”. Ian Stewart incluiu a equação de B&S no seu livro “17 equações que mudaram o mundo”, chamando-a de fórmula de Midas, o rei da lenda grega que transformava tudo o que tocava em ouro. Segundo Stewart: “Em 1998, o sistema financeiro internacional negociou aproximadamente 100 trilhões de dólares americanos em derivativos. Em 2007, esse valor havia crescido para 1 quatrilhão de dólares... Para contextualizar o número, o valor total de todos os produtos fabricados pelas indústrias de todo o mundo, nos últimos mil anos, é de cerca de 100 trilhões de dólares americanos, corrigidos pela inflação. Isso equivale a um décimo dos negócios com derivativos em um ano” [134]. A maioria dos contratos de opção negociados no mercado é na modalidade americana, que pode ser exercida a qualquer momento até o contrato expirar, e não a europeia, objeto da fórmula de B&S. O modelo de Cox-Ross-Rubinstein [CRR], embora bem mais simplificado do que o modelo B&S, permite, entretanto, precificar opções americanas. Além disso, conforme demonstraremos, ele converge para o B&S no limite de muitos passos discretos. Nesse trabalho mostraremos como utilizar o CRR para expandir o B&S no caso de opções americanas.

O principal objetivo neste trabalho é demonstrar a capacidade de lidar com diferentes problemas e fontes de dados distintas mas que se relacionam através do método utilizado. A princípio poderiamos utilizar os métodos aplicados em espectroscopia e mercado financeiro em outras áreas e dados diferentes.

2 CONCEITOS FUNDAMENTAIS

Neste capítulo apresentaremos os conceitos básicos utilizados nessa dissertação tais como Minimum Spanning Tree (MST), probabilidade, grafos, distância e produto interno. Aplicaremos MST's no estudo de óleos no capítulo 3 e ações no capítulo 5. Os conceitos explorados nos ajudarão também a definir as principais distribuições probabilísticas utilizadas no mercado financeiro. Esses conceitos serão aplicados diretamente na classificação e agrupamento de amostragens estatísticas no capítulo 4, usando Machine Learning aplicado à espectroscopia Raman, além da conexão entre a representação da arquitetura de redes neurais artificiais com grafos. Teoria dos grafos já é aplicada em áreas como ciência da computação, matemática, telecomunicações, biologia, neurociência, entre outros [9,10,11]. Redes neurais também são aplicáveis em predição de *time series* como preços futuros de ações através de *redes neurais recorrentes* [12, 13].

As MST's são árvores que nos permitem visualizar os clusters de diferentes tipos de dados e como estes se ligam. As aplicações vão desde o mercado de ações, commodities, setores da indústria, biologia, medicina, análise espectral, entre outros. Neste capítulo construiremos as bases para aplicar o conhecimento de análise em espectros de fluorescência de óleos no capítulo 3, análise de espectros Raman no capítulo 4 e no capítulo 5, mostraremos sua utilização na análise de ações do mercado americano. Para tanto necessitaremos das definições da teoria da probabilidade, grafos, **distância** e distância de correlação. Finalmente, discutiremos os algoritmos mais atuais para a obtenção da MST.

2.1 Grafos

Nesta seção definiremos os conceitos básicos necessários para as nossas aplicações utilizando MSTs, salientando que as aplicações da teoria dos grafos são muito mais amplas do que apresentaremos nesta seção.

2.1.1 Introdução aos Grafos

Os **grafos** são objetos fundamentais na ciência de redes [3] definidos como $G(V,E)$ onde $V:\{v_1,v_2,...,v_n\}$ é um conjunto não vazio finito de vértices e $E=\left\{\left\{v_i,v_j\right\},\quad v_i,v_j\in V\right\}$ é um conjunto de subconjuntos entre dois elementos de V

expressando uma relação entre dois vértices chamadas de arestas. Para evitar ambiguidades de notação assumimos que $V \cap E = \varnothing$. O conjunto de vértices de G é $V(G)$ e o conjunto de arestas é $E(G)$. Essa notação, por exemplo, para um grafo $H = (U, P)$, os vértices são $V(H) = U$ e $E(H) = P$ são as arestas. O número de vértices em $G(V, E)$ é chamado **ordem** de G e é denotado por $|G|$. Os grafos podem ser *finitos* ou *infinitos* de acordo com sua ordem. O número de arestas em $G(V, E)$ é o **tamanho** de G denotado por $\|G\|$. Uma aresta $e = \{u, v\} \in E(U, V)$ incide nos vértices $u \in U$ e $v \in V$ é comumente escrita como uv, portanto u e v são *adjacentes*. As arestas podem ser escritas nas formas u e $\{u\}$ onde $u = \{u\} \in U$. O número de arestas que incidem em um vértice v é chamado de *grau* do vértice v indicado como $\deg(v)$ e tais arestas são indicadas por uma **matriz de incidência**:

$$\delta_{ik}(G) = \begin{cases} 1 & \text{se } v_i \text{ é incidente em } e_k \\ 0 & \text{senão} \end{cases}$$

cuja dimensão é $|G| \times \|G\|$.

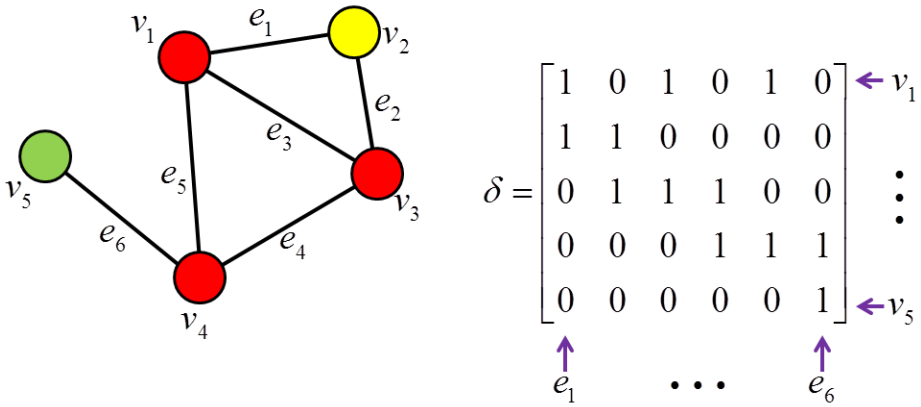


Figura 1 – Grafo com $|G| = 5$, $\|G\| = 6$ e sua matriz de incidência. Vértices com grau 1 em verde, grau 2 em amarelo e grau 3 em vermelho.

O desenho do grafo é feito de forma que os vértices podem ser representados por pontos, ou círculos, ou outra figura geométrica, ligados por linhas, que representam as arestas como na Figura 1. Podemos representar um grafo $G(V, E)$ de ordem n através de uma matriz de adjacência $A(G)$, onde esta será uma matriz quadrada $n \times n$ com as seguintes elementos:

$$a_{ij} = \begin{cases} 1, & \text{se } \{v_i, v_j\} \in E \text{ para } v_i, v_j \in V; \\ 0, & \textit{senão.} \end{cases}$$

No caso do grafo da Figura 1:

$$A(G) = \begin{bmatrix} 0 & 1 & 1 & 1 & 0 \\ 1 & 0 & 1 & 0 & 0 \\ 1 & 1 & 0 & 1 & 0 \\ 1 & 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 1 & 0 \end{bmatrix}.$$

A matriz de adjacência possui as seguintes propriedades:

- $A(G)$ é real e simétrica, Hermitiana, portanto, significando que os autovalores são reais, e formada apenas por 0's e 1's. ;
- A soma dos autovalores é zero, pois o traço de A é nulo (não há arestas ligando um vértice a ele mesmo);
- A soma dos elementos na i-ésima linha é o grau do i-ésimo vértice.

Quando o grafo é muito esparso (com poucas conexões), como o caso da MST, as matrizes de adjacência possuem muitas entradas nulas ocupando muito espaço na memória do computador e podem ser substituídas por **listas de adjacência**. A i-ésima lista de adjacência possui apenas os indicadores dos vértices ligados ao i-ésimo vértice. No caso da Figura 1 teremos cinco listas L_i :

$L_1 = [2, 3, 4];$
$L_2 = [1, 3];$
$L_3 = [1, 2, 4];$
$L_4 = [1, 3, 5];$
$L_5 = [4].$

As relações em E podem ser arcos além de arestas. As **arestas** são relações simétricas, isto é, dado um $e_k = \{v_i, v_j\} \in E$ e um $e_l = \{v_j, v_i\} \in E$ então $e_k = e_l$. Um grafo cujas ligações são arestas é chamado de **grafo não-direcionado**, as ligações são biunívocas, do tipo, se João é irmão de José então José é irmão de João. Os **arcos** são ligações unívocas, com direcionalidade, do tipo João quer almoçar com José mas José não quer almoçar com João. Grafos com estas ligações são **grafos direcionados** [3]. Enquanto as arestas são representadas por uma linha ligando os vértices os arcos são representados por setas. O grafo também pode ser misto. Nessa dissertação trabalharemos apenas com grafos não-direcionados de modo que a palavra **grafo** será utilizada como sinônimo de um grafo não-direcionado.

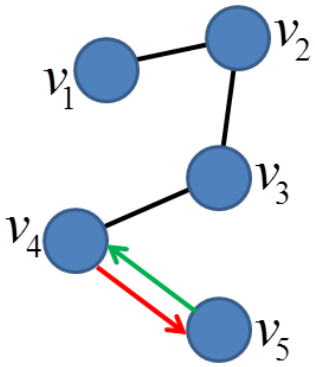


Figura 2 – Arestas em preto e arcos em vermelho e verde.

Os grafos podem conter diferentes pesos entre cada relação E de forma que estes serão definidos como $G \equiv G(V, E, W, f)$ onde a matriz W é um conjunto de pesos positivos e reais mapeados através da função f na forma:

$$f : E \to W,$$
$$f : \{v_i, v_j\} \to w_{ij} \in W \mid w_{ij} \in \mathbb{R}^+.$$

No caso das MSTs, os pesos W representam as distâncias entre os vértices, quanto menor o peso, maior a proximidade entre os vértices.

2.1.2 Subgrafos

Sejam dois diferentes grafos G e G', definimos as operações de união e interseção como $G \cup G' = (V \cup V', E \cup E')$ e $G \cap G' = (V \cap V', E \cap E')$. Caso $G \cap G' = \varnothing$ então G e G' são *disjuntos*. No caso em que $V' \subset V$ e $E' \subset E$, G' é um subgrafo de G escrito na forma $G' \subset G$.

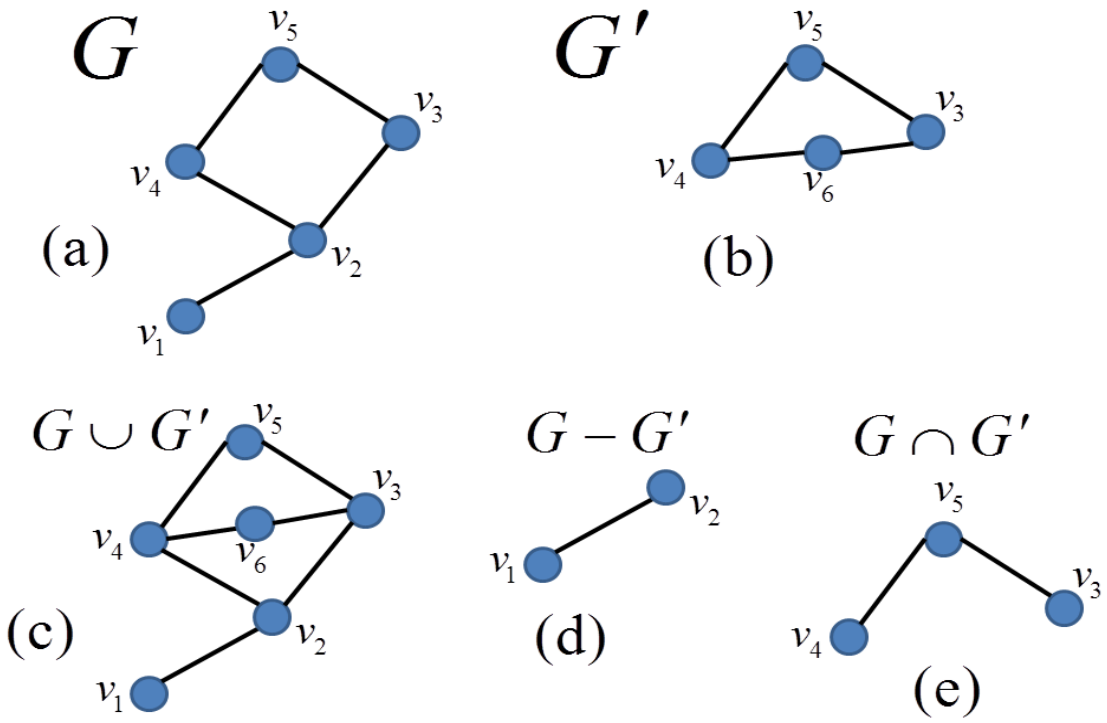


Figura 3 – União, diferença e interseção (embaixo) dos conjunto G e G'.

No caso em que $G' \subset G$ onde $V' = V$, G' é um subgrafo *abragente* (*spanning*) de G.

Seja U conjunto de vértices, usualmente um subconjunto do grafo $G(V, E)$. Escrevemos $G - U$ como sendo $G[V \setminus U]$ onde a operação $V \setminus U$ representa a remoção dos vértices $U \cap V$ de V e suas arestas incidentes. Para um subconjunto F de $[V]^2$ (arestas) escrevemos $G - F = (V, E \setminus F)$ e $G + F = (V, E \cup F)$.

2.1.3 Caminho e Ciclo

Um **caminho** é um grafo que deve começar e terminar em vértices diferentes, e é, também, um grafo não-vazio $P = P(V, E)$ com: $V = \{x_1, x_2, \ldots, x_k\}$ e $E = \{x_1x_2, x_2x_3, \ldots, x_{k-1}x_k\}$. Os vértices x_1 e x_k estão ligados por P e são seus vértices externos enquanto que os vértices $x_2, \ldots, x_{k-1}$ são seus vértices internos. O número de arestas no caminho $\|P\|$ é seu *comprimento*. Um caminho de comprimento *k* é escrito como P^k, $\|P^k\| = k$. Geralmente representamos um caminho como sua sequência natural de vértices:

$P = x_1 x_2 \ldots x_k$ para um caminho de x_1 para x_k. Se um $P = x_1 \ldots x_k$ é um caminho com $k \geq 3$ então o grafo $C = P + x_k x_1$ é chamado de **ciclo**. A Figura 4 representa um ciclo:

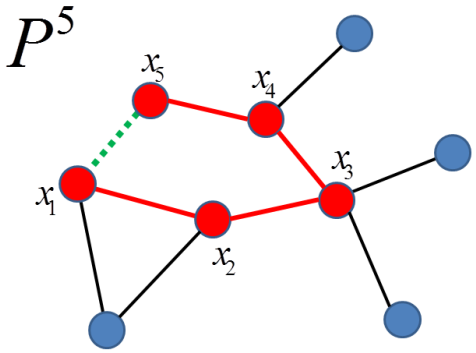


Figura 4 – Caminho P^5 em vermelho e aresta $x_1 x_5$ em verde. Ciclo $C = P^5 + x_1 x_5$.

Um grafo G é *conectado* se dois vértices quaisquer estão ligados por um caminho em G. Se nenhum vértice se repetiu no caminho, chamamos este de *caminho simples*. Chamamos de *caminho hamiltoniano* o caminho que passa uma única vez em todos os vértices do grafo. A **distância** entre dois vértices é o comprimento do caminho mais curto entre eles e o **diâmetro** do grafo é a maior distância entre os vértices do grafo [4].

2.1.4 Árvores

Um grafo *acíclico* é um grafo que não contém nenhum ciclo e é chamado de **árvore**, os seus vértices de grau 1 são suas *folhas*, se removermos uma folha de uma árvore, esta ainda será uma árvore. Toda árvore possui folhas.

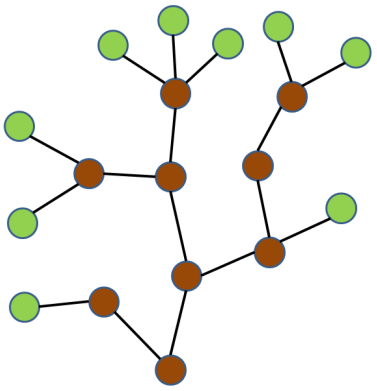


Figura 5 – Árvore. Folhas em verde.

As afirmativas abaixo devem ser verdade para todo grafo de árvore *T*:

- Quaisquer dois vértices de *T* estão ligados por um caminho único em *T*;
- *T* é minimamente conectado, ou seja, o grafo $T-e$ é desconectado, para uma aresta $e \subset E(T)$;
- *T* é acíclica mas um $T+xy$ deve conter ciclos, para qualquer $x, y \subset V(T)$ dois vértices não adjacentes;
- Um grafo conectado com *n* vértices é uma árvore se e somente possuir $n-1$ arestas.

A partir destas afirmativas concluímos que uma *spanning tree* (ST) é um subgrafo de um grafo com mesma ordem *n* mas com um tamanho $n-1$.

2.1.5 Grafo completo

A árvore é minimamente conectada. O contrário disto é um **grafo completo**. Seja um grafo $G(V,E)$ dito completo de grau $|G|=n$, cada vértice conecta-se com os outros $n-1$. Como temos n vértices, o número de arestas é $n(n-1)$, mas desta forma contamos cada aresta duas vezes, portanto $\|G\| = \frac{n(n-1)}{2}$. Um grafo completo com *n* vértices é denotado por K_n, na figura abaixo representamos alguns grafos completos [5]:

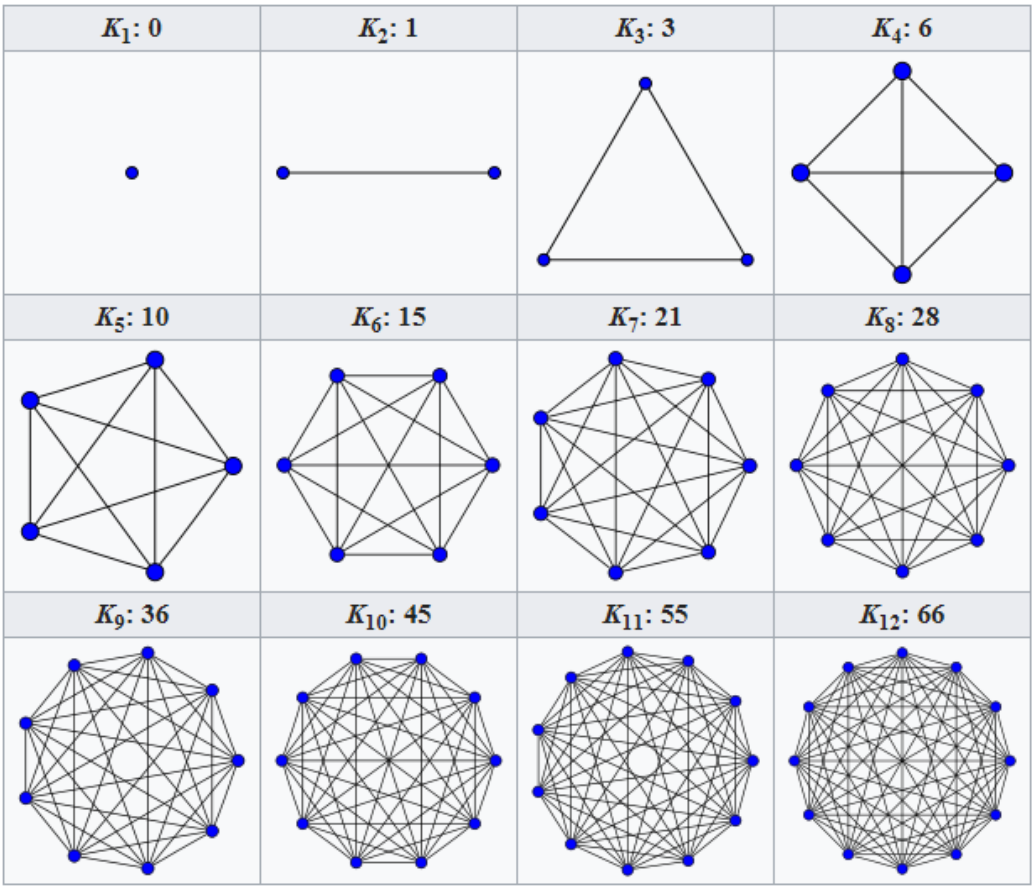


Figura 5 – Fonte: [14]. Grafos completos.

2.1.6 Teorema de Kirchoff

Gustav Kirchoff no século 19 estudava circuitos elétricos e os descrevia através de grafos. Kirchoff descobriu que através da matriz Laplaciana de um grafo é possível obter o número de ST possíveis [8]. Para um grafo $G(V,E)$ com n vértices $V=\{v_1,\ldots,v_n\}$, sua matriz Laplaciana é

$$L_{ij}=\begin{cases}\deg(v_j), & Se\ i=j\\ -1, & Se\ i\neq j\ e\ (v_i,v_j)\in E\\ 0, & senão\end{cases}$$

Lembrando que $\deg(v_j)$ é o grau de v_j portanto $L=D-A$ onde D é a matriz diagonal $D_{jj}=\deg(v_j)$ e A é a matriz de adjacência do grafo G.

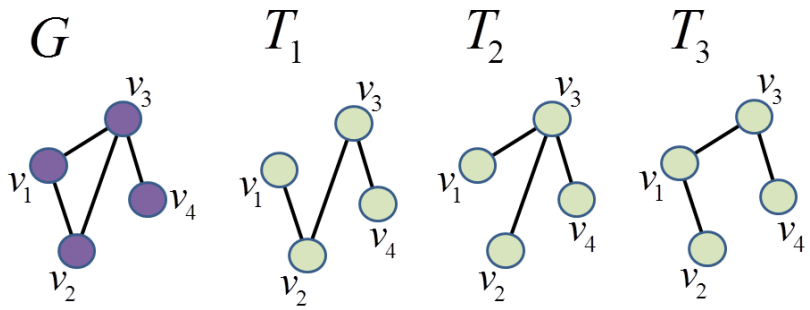


Figura 6 – Grafo $G(V,E)$ e *spanning trees* T_1, T_2, T_3.

Na figura acima, a matriz Laplaciana é:

$$L = D - A = \begin{bmatrix} 2 & 0 & 0 & 0 \\ 0 & 2 & 0 & 0 \\ 0 & 0 & 3 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} - \begin{bmatrix} 0 & 1 & 1 & 0 \\ 1 & 0 & 1 & 0 \\ 1 & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{bmatrix} = \begin{bmatrix} 2 & -1 & -1 & 0 \\ -1 & 2 & -1 & 0 \\ -1 & -1 & 3 & -1 \\ 0 & 0 & -1 & 1 \end{bmatrix}$$

O teorema de Kirchoff diz que, para um grafo $G(V,E)$ com matriz laplaciana *L*, o número de ST's N_T contidas em *G* é dada pelos seguintes passos:

- Escolha um vértice v_j arbitrário e elimine a *j*-ésima linha e coluna de *L* para obter a matriz $\hat{L}_j$;
- Calcule $N_T = \det\left(\hat{L}_j\right)$.

Independente do vértice selecionado o resultado será o mesmo, para o grafo da Figura 6:

$$N_T = \begin{vmatrix} 2 & -1 & 0 \\ -1 & 3 & -1 \\ 0 & -1 & 1 \end{vmatrix} = \begin{vmatrix} 2 & -1 & 0 \\ -1 & 3 & -1 \\ 0 & -1 & 1 \end{vmatrix} = \begin{vmatrix} 2 & -1 & 0 \\ -1 & 2 & 0 \\ 0 & 0 & 1 \end{vmatrix} = \begin{vmatrix} 2 & -1 & -1 \\ -1 & 2 & -1 \\ -1 & -1 & 3 \end{vmatrix} = 3$$

As três ST's estão representadas na Figura 6 [8].

2.1.7 Isomorfismo

Chamamos dois grafos $G(V,E)$ e $G'(V',E')$ *isomórficos* se existir uma relação de bijeção $\varphi : V \to V'$ aplicada nas arestas onde $xy \in E \Leftrightarrow \varphi(x)\varphi(y) \in E' \ \forall x, y \in V$. Tal mapa φ é chamado de *isomorfismo*. Denotamos a relação de isomorfismo entre dois grafos como $G \simeq G'$. Vemos na Figura 6 que um grafo e todas suas ST's são isomórficos. A relação $G = G'$ chama-se *automorfismo*.

2.2 Minimum Spanning Trees

O pioneiro da área da MST foi o matemático tcheco Otakar Borůvka (1899 – 1995) que nasceu em Uherský Ostroh, na Morávia. Quando ele nasceu a Morávia pertencia ao império Austro-Húngaro, depois passou a fazer parte da Checoslováquia (1918-1992) e hoje faz parte da República Checa. Borůvka se deparou com o problema de projetar uma rede elétrica conectando um conjunto de cidades junto com suas distâncias conhecidas, com o menor custo possível. A Primeira minimização do custo corresponderia a minimização do comprimento total dos cabos elétricos. Entretanto, a minimização do comprimento total dos cabos pode não corresponder a minimização do custo total porque a instalação dos cabos em terrenos montanhosos pode ser mais cara do que nas planícies. Sem falar de outros acidentes geográficos como rios, lagos etc. Esse problema é facilmente resolvido colocando pesos e redefinindo as distâncias em custos.

Na MST todas as cidades estariam interconectadas sem redundância. Em 1926 ele publicou o trabalho ***"jistém problému minimálním"***, ou ***"Sobre um certo problema de mínimo"***. A MST de uma rede em que todas as distâncias, ou pesos, são distintas admite uma solução única. Uma rede de conexões elétricas, ou de comunicações, necessita de redundâncias para garantir a robustez, caso contrário, a quebra de uma conexão impede a operação satisfatória da mesma. No caso de uma rede de comunicação, por exemplo, um tráfico intenso em certos trechos, mesmo sem nenhum dano físico à rede, atrasaria as comunicações entre usuários, podendo ser vantajoso usar um caminho alternativo menos congestionado. A empresa americana American Telephone and Telegraph (AT&T) se deparou com o seguinte problema legal: ela deveria cobrar dos consumidores pela conexão mais barata entre os dois, e não pela conexão na trajetória realmente utilizada. Esse é um problema típico de MST e não surpreende, portanto, que os dois algoritmos mais utilizados hoje, de PRIM e de Kruskal, tenham sido desenvolvidos no AT&T Bell Laboratories. Embora o algoritmo de Kruskal seja o mais rápido, o algoritmo de PRIM tem vantagens na reorganização de matrizes de distância e correlação, como veremos nas aplicações. No algoritmo de PRIM buscamos os vértices mais próximos, e depois procuramos um vértice que ainda não pertence à MST mais próximo de qualquer outro vértice da MST, e assim vamos construindo a rede.

Um algoritmo ingênuo seria um que gera todas as ST's e soma os pesos de todas as arestas de cada ST selecionando a MST cuja soma dos pesos deve ser a menor. O problema é que para grafos maiores, o número de ST's pode ser muito grande consumindo muita memória do computador além de muito tempo para computar. Se a soma dos pesos de cada

ST é única então deve existir apenas uma cuja soma dos pesos é a menor possível que é a MST. Atualmente o algoritmo mais rápido é o algoritmo randomizado de Karger, Klein e Tarjan [16].

2.2.1 Unicidade da MST

Vamos provar por absurdo. Suponham que existam T_1 e T_2 duas MST's distintas geradas a partir de um mesmo grafo G visualizadas na Figura 7. Ambas árvores têm os mesmos vértices, mas deve haver pelo menos uma aresta que pertence a uma, mas não a outra. Denotamos estas arestas como $e_1 \in E(T_1)$ e $e_2 \in E(T_2)$. Assumindo, sem perda de generalidade, que a aresta com menor peso entre as duas é e_1, sabemos que $e_1 \notin E(T_2)$ então $T_2 \cup \{e_1\}$ deve conter um ciclo e uma aresta deste ciclo é $e_2 \notin E(T_1)$. Como $e_2 \neq e_1$ e $w(e_1) < w(e_2)$, a árvore $T = T_2 \cup \{e_1\} \setminus \{e_2\}$ é uma SP cujo peso total é menor que o peso total de T_2, o que contradiz a hipótese inicial de que T_2 é uma MST. Concluímos, portanto, que só deve haver uma MST se o grafo possui pesos distintos entre suas arestas. Se todos os pesos de um grafo *G* forem iguais, então todas as SP's deste grafo terão peso total mínimo, portanto todas serão MST's.

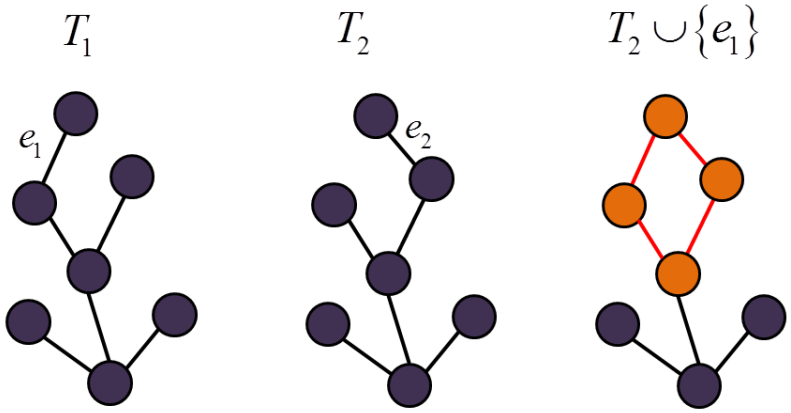


Figura 7 – Ciclo em $T_2 \cup \{e_1\}$.

2.2.2 Algoritmo de Kruskal

No algoritmo Kruskal escolhe-se a aresta com o menor peso de todas e vai-se selecionando as próximas menores arestas sujeito a restrição de não formar ciclos, como no exemplo [15]:

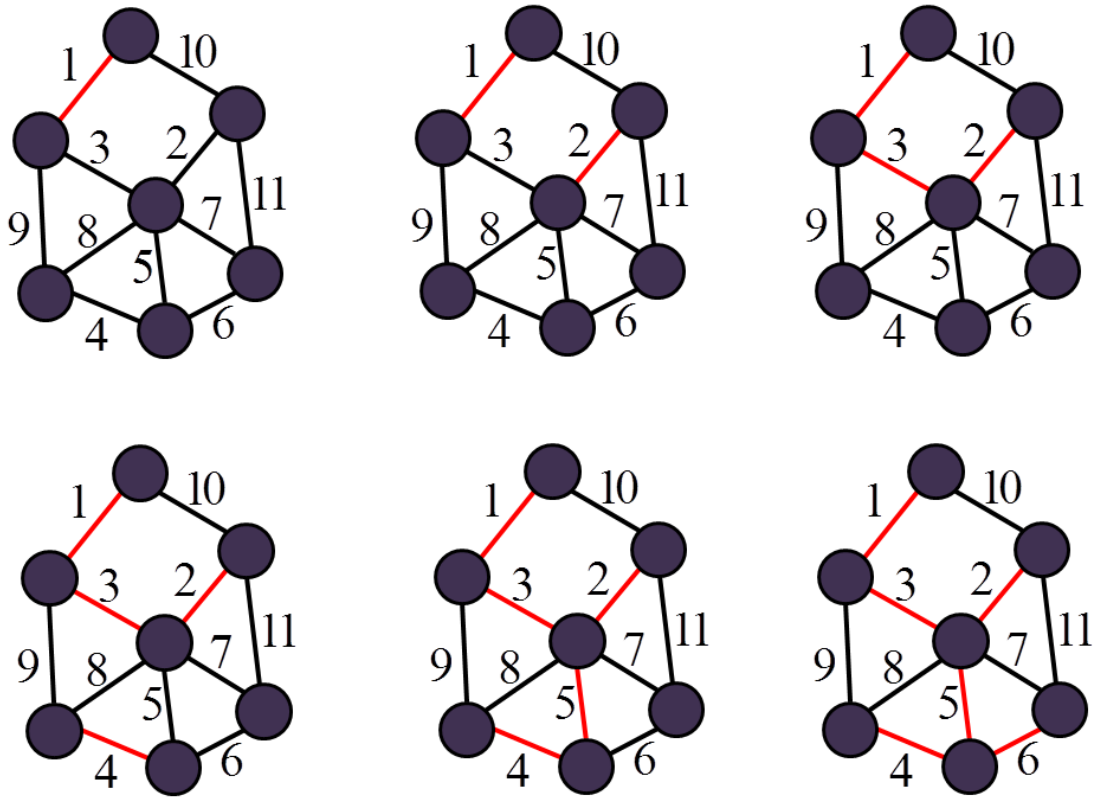


Figura 8 – *MST* formada através do algoritmo de Kruskal.

2.2.3 Algoritmo de Prim

Através do método PRIM, os vértices são conectados de forma a gerar uma lista ordenada e podemos usar esse ordenamento para reorganizar a matriz de correlação ou distância, mostrando clusters dos vértices mais próximos. Incorporaremos a escolha da aresta com peso mínimo global como aresta inicial da MST [17].

As MST via algoritmo de PRIM são formadas através de redes indiretas no qual reduziu-se o conjunto de arestas *E* do grafo *G* para um conjunto com $n-1$ arestas dado por $E' \in E$ onde *n* é o número de vértices sob a restrição de que só haja um caminho possível de ligação entre dois vértices qualquer. Desta forma gera-se um subgrafo $H(V, E', W', f')$ onde *H* é um

subgrafo de *G*, ou seja, $H \subset G$ [2,1].

O algoritmo é implementado na ordem dos passos:

1. Recebe-se o *input* $G(V, E, W, f)$;
2. Inicializa-se o conjunto de vértices *S* da $MST(S, F)$ selecionando os dois vértices cuja aresta e' possui menor peso em *V* e adiciona-se e' a *F*;
3. $V \coloneqq V \setminus \{V \cap S\}$, isto é, o subconjunto de vértices $V \cap S$ é removido do conjunto *V*. Similarmente $E \coloneqq E \setminus \{E \cap F\}$;
4. Procura-se as arestas $e = \{u, v\}$ onde $u \in S$ e $v \in V$, ou seja, as arestas dos vértices em *V* adjacentes aos vértices em *S*;
5. Identifica-se a aresta com menor peso $e' = \{u', v'\}$ onde $u' \in S$ e $v' \in V$;
6. Adiciona-se o vértice v' a *S* e a aresta e' à *F* da MST;
7. Repetir a partir do passo 3 até que $V \setminus S = \varnothing$.

Abaixo ilustramos a obtenção de uma MST utilizando um grafo não-direcionado *G* com o algoritmo de PRIM:

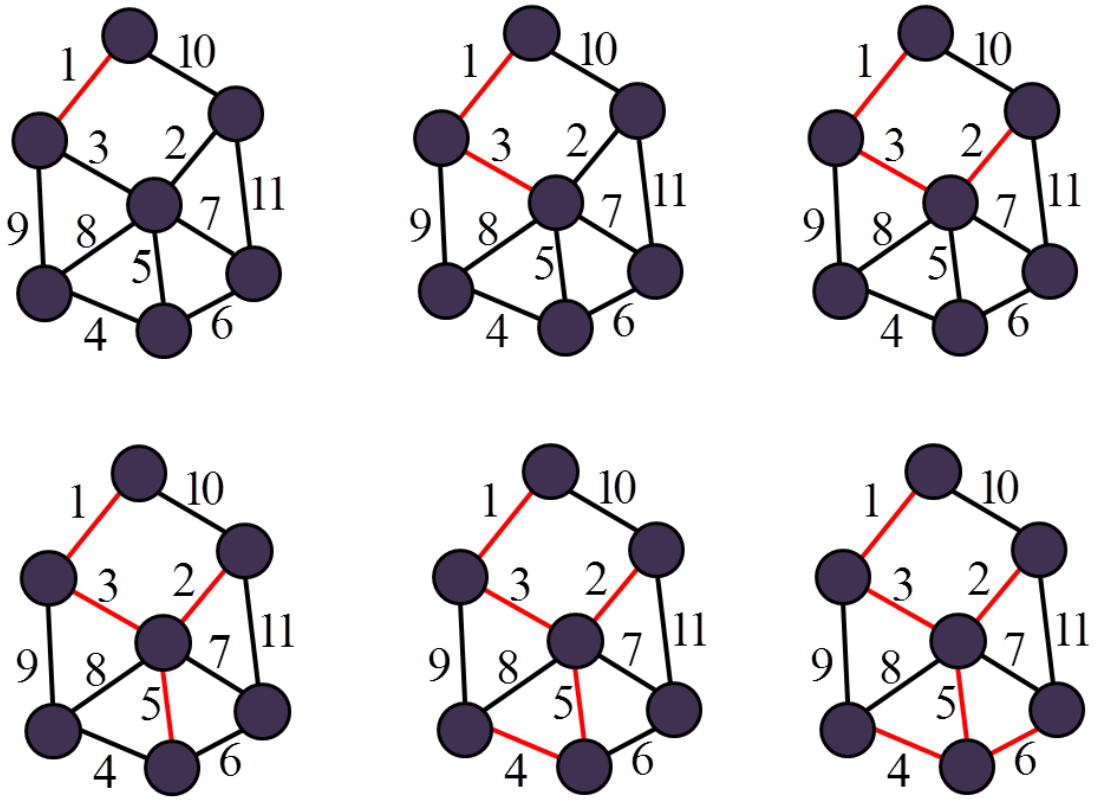


Figura 9 – *MST* de um grafo com algoritmo de PRIM.

Uma *MST* também pode ser representada através do dendograma:

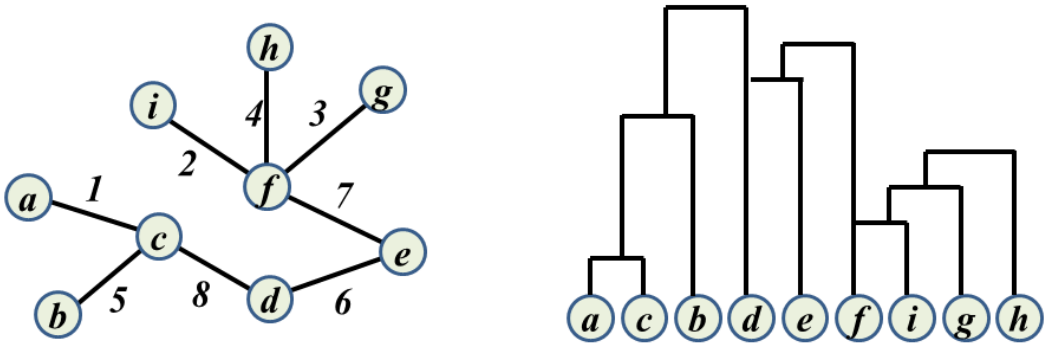


Figura 10 – Representação das ligações de uma *MST* através do dendograma.

2.3 Espaços métricos e axiomas de uma distância

Nessa dissertação utilizaremos a distância de correlação entre amostras para gerar a MST, por isso é necessário definir axiomaticamente o que é uma distância, e mostrar que a distância de correlação obedece aos axiomas de uma distância. Também vamos definir as funções distribuição e densidade de probabilidade, seus momentos centrados e não centrados e como obter o coeficiente de correlação de Pearson. Com ele mostramos, então, como se define uma distância de correlação.

2.3.1 Definição de espaço métrico

Para que o espaço seja métrico, deve existir uma função distância $d(x,y): E \times E \to \mathbb{R}$ onde $\forall x, y \in E$ que satisfaz os axiomas:

1. Desigualdade triangular: $d(x,z) \leq d(x,y) + d(y,z)$;
2. Se $d(x,y) = 0$ então $x = y$;
3. $d(x,y) = d(y,x)$.

Uma consequência direta dos axiomas é que $d(x,y) \geq 0$. Para demonstrar basta fazer $z = x$ no axioma 1: $d(x,x) \leq d(x,y) + d(y,x)$ e utilizando os axiomas (2) e (3) $2d(x,y) \geq 0$, logo $d(x,y) \geq 0$. A função distância deve ser real e positiva. Caso a função

exista, esta será a métrica do espaço, de forma que poderemos medir distâncias entre os elementos do conjunto E.

2.3.2 Definição de Produto Interno

O produto interno pode ser definido de forma generalizada através dos seguintes axiomas. Suponha um conjunto de elementos $\varphi_k \quad k \in \mathbb{Z}^+ \equiv \{0,1,2,3,\cdots\}$, que podem ser funções, vetores, matrizes, etc, e que admite uma operação denominada produto interno (*inner product*) $\langle \varphi_i | \varphi_j \rangle : E \to \mathbb{C}$ definida através de 3 axiomas:

1. Simetria conjugada $\langle \varphi_j | \varphi_i \rangle^* = \langle \varphi_i | \varphi_j \rangle$;
2. Linearidade $\langle \varphi_k | a\varphi_i + b\varphi_j \rangle = a\langle \varphi_k | \varphi_i \rangle + b\langle \varphi_k | \varphi_j \rangle$;
3. Positiva definida $\langle \varphi_i | \varphi_i \rangle \geq 0$ e, se $\langle \varphi_i | \varphi_i \rangle = 0$, então $\varphi_i = 0$.

Do axioma (1) podemos mostrar que $\langle \varphi_i | \varphi_i \rangle \in \mathbb{R}$ pois $\langle \varphi_i | \varphi_i \rangle^* = \langle \varphi_i | \varphi_i \rangle$. Além disso $\langle \varphi_i | \varphi_i \rangle \geq 0$, pelo axioma (3). O produto interno então pode ser usado na definição de uma norma:

$$\|\varphi_i\| \equiv \sqrt{\langle \varphi_i | \varphi_i \rangle}$$

Não confundir com a notação utilizada em grafos. O produto interno de um elemento com ele mesmo é real e positivo e também pode ser usado como uma métrica definindo a distância como:

$$d_{ij}^2 = \langle \varphi_i - \varphi_j | \varphi_i - \varphi_j \rangle = \langle \varphi_i | \varphi_i \rangle - \left[\langle \varphi_i | \varphi_j \rangle + \langle \varphi_j | \varphi_i \rangle \right] + \langle \varphi_j | \varphi_j \rangle .$$

Note que essa definição já satisfaz aos axiomas (2) $d_{ij} = d_{ji}$ e (3) pois se $d_{ij}^2 = 0 \quad \leftrightarrow \quad \varphi_i = \varphi_j$.

Como $\langle \varphi_j | \varphi_i \rangle = \langle \varphi_i | \varphi_j \rangle^*$ então

$$\langle \varphi_i | \varphi_j \rangle + \langle \varphi_j | \varphi_i \rangle = \langle \varphi_i | \varphi_j \rangle + \langle \varphi_i | \varphi_j \rangle^* = 2\,\mathrm{Re}\left[\langle \varphi_i | \varphi_j \rangle \right] \qquad (2.1)$$

de modo que:

$$d_{ij}^2 = \langle \varphi_i | \varphi_i \rangle - 2\,\mathrm{Re}\left[\langle \varphi_i | \varphi_j \rangle \right] + \langle \varphi_j | \varphi_j \rangle .$$

Assim percebemos que $d_{ij}^2 \in \mathbb{R}$, mas para ser uma distância é necessário que $d_{ij}^2 \geq 0$. Com a desigualdade de Schwartz provamos não apenas que $d_{ij}^2 \geq 0$ quanto a desigualdade triangular.

2.3.3 Desigualdade de Schwartz

Sabemos que a norma é sempre positiva e real. A desigualdade para os conjuntos φ_i, φ_j é dada por:

$$\left\|\varphi_i - \lambda\varphi_j\right\| \geq 0 \quad \forall\lambda \in \mathbb{R}$$

Elevando ao quadrado temos

$$\left\|\varphi_i - \lambda\varphi_j\right\|^2 = \left\langle \varphi_i - \lambda\varphi_j \middle| \varphi_i - \lambda\varphi_j \right\rangle = \left\langle \varphi_i \middle| \varphi_i \right\rangle - \lambda\left[\left\langle \varphi_i \middle| \varphi_j \right\rangle + \left\langle \varphi_i \middle| \varphi_j \right\rangle^*\right] + \lambda^2 \left\langle \varphi_j \middle| \varphi_j \right\rangle.$$

Utilizou-se o axioma 2 do produto interno para obter o termo cruzado conjugado. O termo central é substituído usando a expressão (2.1):

$$\left\|\varphi_i - \lambda\varphi_j\right\|^2 = \left\langle \varphi_i \middle| \varphi_i \right\rangle - 2\lambda\,\mathrm{Re}\left[\left\langle \varphi_i \middle| \varphi_j \right\rangle\right] + \lambda^2 \left\langle \varphi_j \middle| \varphi_j \right\rangle.$$

Agora:

$$\lambda^2 \left\|\varphi_j\right\|^2 - 2\lambda\,\mathrm{Re}\left[\left\langle \varphi_i \middle| \varphi_j \right\rangle\right] + \left\|\varphi_i\right\|^2 \geq 0 \quad \forall\lambda$$

Em termos de λ temos uma equação quadrática da forma $P(\lambda) = a\lambda^2 + b\lambda + c$ com $a > 0$, que não pode ser negativa, logo não admite raízes reais, ou seja, $b^2 - 4ac \leq 0$. Então:

$$4\,\mathrm{Re}^2\left[\left\langle \varphi_i \middle| \varphi_j \right\rangle\right] - 4\left\|\varphi_j\right\|^2 \left\|\varphi_i\right\|^2 \leq 0$$

Portanto $\left[\dfrac{\mathrm{Re}\left[\left\langle \varphi_i \middle| \varphi_j \right\rangle\right]}{\left\|\varphi_j\right\|\left\|\varphi_i\right\|}\right]^2 \leq 1$ ou ainda $-1 \leq \dfrac{\mathrm{Re}\left[\left\langle \varphi_i \middle| \varphi_j \right\rangle\right]}{\left\|\varphi_j\right\|\left\|\varphi_i\right\|} \leq 1.$

Este número se encontra no intervalo $[-1, +1]$ podendo ser associado a uma função coseno:

$$\frac{\mathrm{Re}\left[\left\langle \varphi_i \middle| \varphi_j \right\rangle\right]}{\left\|\varphi_j\right\|\left\|\varphi_i\right\|} = \cos\theta.$$

Dessa forma, o produto interno (escalar) de dois elementos é dado por $\mathrm{Re}\left[\left\langle \varphi_i \middle| \varphi_j \right\rangle\right] = \left\|\varphi_j\right\|\left\|\varphi_i\right\|\cos\theta$. Com isso podemos definir uma ortogonalidade se $\theta = \dfrac{\pi}{2}$ quando $\mathrm{Re}\left[\left\langle \varphi_i \middle| \varphi_j \right\rangle\right] = 0$.

A desigualdade de Schwartz também prova que $d_{ij}^2 \geq 0$ pois:

$$d_{ij}^2 = \left\|\varphi_i\right\|^2 - 2\frac{\mathrm{Re}\left[\left\langle \varphi_i \middle| \varphi_j \right\rangle\right]}{\left\|\varphi_i\right\|\left\|\varphi_j\right\|}\left\|\varphi_i\right\|\left\|\varphi_j\right\| + \left\|\varphi_j\right\|^2 = \left\|\varphi_i\right\| - 2\left\|\varphi_i\right\|\left\|\varphi_j\right\|\cos\theta + \left\|\varphi_j\right\| \geq \left[\left\|\varphi_i\right\| - \left\|\varphi_j\right\|\right]^2 \geq 0.$$

A desigualdade triangular também sai da desigualdade de Schwartz. O termo d_{ij}^2 é:

$$d_{ij}^2 = \langle \varphi_i - \varphi_j | \varphi_i - \varphi_j \rangle = \langle (\varphi_i - \varphi_k) + (\varphi_k - \varphi_j) | (\varphi_i - \varphi_k) + (\varphi_k - \varphi_j) \rangle =$$
$$= \langle (\varphi_i - \varphi_k) | (\varphi_i - \varphi_k) \rangle + 2\,\mathrm{Re}\left[\langle (\varphi_i - \varphi_k) | (\varphi_k - \varphi_j) \rangle \right] + \langle (\varphi_k - \varphi_j) | (\varphi_k - \varphi_j) \rangle =$$
$$= d_{ik}^2 + 2\frac{\mathrm{Re}\left[\langle (\varphi_i - \varphi_k) | (\varphi_k - \varphi_j) \rangle \right]}{d_{ik} d_{kj}} d_{ik} d_{kj} + d_{kj}^2.$$

Teremos a desigualdade:

$$d_{ij}^2 = d_{ik}^2 + 2\frac{\mathrm{Re}\left[\langle (\varphi_i - \varphi_k) | (\varphi_k - \varphi_j) \rangle \right]}{d_{ik} d_{kj}} d_{ik} d_{kj} + d_{kj}^2 \le d_{ik}^2 + 2 d_{ik} d_{kj} \cos\theta + d_{kj}^2 = \left(d_{ik} + d_{kj} \right)^2$$

Logo:

$$d_{ij} \le d_{ik} + d_{kj}$$

Com isso mostramos que $d_{ij} = \sqrt{\langle \varphi_i - \varphi_j | \varphi_i - \varphi_j \rangle}$ satisfaz à desigualdade triangular e aos axiomas de uma distância.

2.3.4 Distância Euclidiana

O produto escalar vetorial padrão é dado por

$$\vec{X} \cdot \vec{Y} = \begin{bmatrix} x_1 & x_2 & \cdots & x_n \end{bmatrix} \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{bmatrix} = x_1 y_1 + x_2 y_2 + \cdots + x_n y_n = \sum_j x_j y_j$$

ou simplesmente $x_j y_j$ em notação de Einstein. Os produtos acima dos vetores $x_j \in \mathbb{R} \quad e \quad y_j \in \mathbb{R}$ satisfazem aos axiomas de um produto interno. A simetria conjugada é obedecida: $\left(\vec{X} \cdot \vec{Y} \right)^* = \vec{X} \cdot \vec{Y} = \vec{Y} \cdot \vec{X}$, pois os vetores são reais. O produto interno é um operador linear: $\vec{X} \cdot \left(a\vec{Y} + b\vec{Z} \right) = a\vec{X} \cdot \vec{Y} + b\vec{X} \cdot \vec{Z}$ e $\vec{X} \cdot \vec{X} \ge 0$.

Daí definimos a distância Euclidiana, mais conhecida de todas, desde Pitágoras, e que serviu de paradigma para a definição axiomática de uma distância, através de:

$$d\left(\vec{x}, \vec{y} \right) = \sqrt{\left(\vec{X} - \vec{Y} \right) \cdot \left(\vec{X} - \vec{Y} \right)} = \sqrt{\sum_{j=1}^{n} \left(x_j - y_j \right)^2}.$$

2.4 Teoria da Probabilidade

2.4.1 Distribuição de Probabilidade

A função de distribuição acumulada, que nos referiremos pela sua sigla em inglês CDF de *Cumulative Density Function* é dada por $F(x) = P(\{x \leq y\}) \quad \forall y \in \mathbb{R}$ onde o conjunto $\{x \leq y\} \forall y \in \mathbb{R}$ é definido como um evento. Vamos listar algumas de suas propriedades:

1) $F(+\infty) = 1$ e $F(-\infty) = 0$;

2) $F(x)$ é sempre crescente e contínua pela direita;

3) A probabilidade de um evento entre x_2 e x_1 é dada pela diferença: $P(\{x_1 < x \leq x_2\}) = F(x_2) - F(x_1)$.

A função densidade de probabilidade (FDP) $f(x)$ será a derivada da CDF:

$$f(x) = \frac{dF(x)}{dx} \qquad \rightarrow \qquad F(x) = \int_{-\infty}^{x} f(x)dx.$$

Algumas de suas propriedades são:

1) $f(x) \geq 0$ pois $F(x)$ é sempre crescente.

2) O item 3 da lista anterior pode ser reescrito em termos da FDP: $P(\{x_1 < x < x_2\}) = \int_{x_1}^{x_2} f(x)dx$. Tomando esta última propriedade, para encontrar a probabilidade em um intervalo dx: $P(\{x < x' \leq x + dx\}) = f(x)dx$.

2.4.2 Distância entre funções

Como aplicação direta dos conhecimentos das seções anteriores e através da definição da função FDP, podemos definir a distância entre duas funções. Para isto vamos definir uma operação que satisfaz as condições de produto interno de funções $\varphi(x): \mathbb{R} \rightarrow \mathbb{C}$ dada por:

$$\langle \varphi_i | \varphi_j \rangle = \int_a^b \varphi_i^*(x)\varphi_j(x)w(x)dx \quad \text{com} \quad b > a \quad \text{e a função peso} \quad w(x): \mathbb{R} \rightarrow \mathbb{R} \geq 0 \quad \forall x \in [a,b].$$

Para a definição de produto interno a restrição da função peso ser positiva é suficiente. Entretanto, como a média ponderada pode ser extraída através de pesos normalizados, sempre

podemos exigir, sem perda de generalidade, que $\int_a^b w(x)\,dx = 1$. Se $\int_a^b w(x)\,dx \neq 1$, redefinimos $w'(x) = \frac{w(x)}{\int_a^b w(x)\,dx}$ de modo que $\int_a^b w'(x)\,dx = 1$. Notamos que as exigências de que $w(x) \geq 0 \quad \forall x \in [a,b]$ e $\int_a^b w(x)\,dx \neq 1$ coincidem com a exigência de que $w(x)$ seja uma função densidade de probabilidade no intervalo $[a,b]$.

Note que dessa definição temos que $\langle \varphi_j | \varphi_i \rangle = \int_a^b \varphi_j^*(x)\varphi_i(x) f(x)\,dx$ é um produto interno legítimo e que, portanto, $d(\varphi_i, \varphi_j) = \int_a^b |\varphi_j(x) - \varphi_i(x)|^2 f(x)\,dx$ é uma distância.

2.4.3 Distribuições Discretas: Tratando descontinuidades

O caso das distribuições discretas é um caso particular das distribuições contínuas através do uso das funções impulso, ou delta de Dirac, que tem as seguintes propriedades:

(1) $\int_{-\infty}^{+\infty} \delta(x - x_o)\,dx = 1$

(2) $\int_{-\infty}^{+\infty} f(x)\delta(x - x_o)\,dx = f(x_o)$,

Ou seja, $\delta(x - x_o) = 0 \quad \forall x \neq x_o$, mas tem área unitária. Trata-se portanto de uma função com largura nula mas altura infinita para garantir a área igual a 1. Vale também notar que a área unitária da função Delta de Dirac implica que a mesma tem dimensão de $1/x$. A propriedade dessa função que conecta as distribuições discretas com as contínuas vem da derivada da função degrau de Heaviside, dada por $\frac{d}{dx} H(x - x_o) = \delta(x - x_o)$. A função de Heaviside, ou função degrau, é definida por:

$$H(x - x_0) = \begin{cases} 1, & x > x_0 \\ \frac{1}{2}, & x = x_0 \\ 0, & x < x_0 \end{cases}$$

A descontinuidade em $x = x_0$ a torna não diferenciável neste ponto. Para contornar isto definiremos uma nova função de Heaviside:

$$H_n(x - x_0) = \frac{1}{1 + e^{-n(x - x_0)}}.$$

Que é diferenciável em todo o seu domínio, com os seguintes limites no infinito na H_n:

$\lim_{x \to -\infty} H_n(x - x_0) = 0$ e $\lim_{x \to +\infty} H_n(x - x_0) = 1$ e que em $x = x_0$ vale $H_n(0) = \frac{1}{2}$. A área sobre sua derivada deve ser unitária, pois:

$$\int_{-\infty}^{+\infty} \frac{dH_n}{dx} dx = H_n(+\infty) - H_n(-\infty) = 1.$$

Quando o n na H_n tende ao infinito a função se aproxima da função degrau, ou $\underset{n \to \infty}{Lim} H_n(x - x_o) = H(x - x_o)$, como vemos na Figura 11.

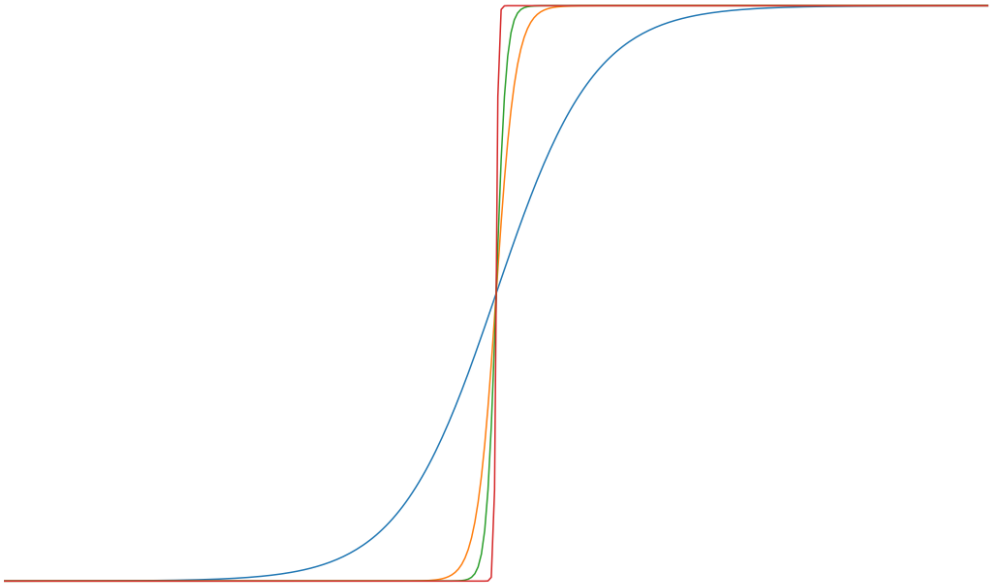


Figura 11 – Função H_n com $n = 1$ em azul a $n = 50$ em vermelho.

Agora tomando a derivada da H_n obtemos:

$$\frac{dH_n}{dx} = \frac{n \exp(-n(x - x_0))}{[1 + \exp(-n(x - x_0))]^2}$$

que tem a propriedade de ter área unitária. Fazendo o n tender ao infinito, como vemos na Figura 12, esta função tenderá a distribuição *delta de Dirac* [19]:

$$\lim_{n\to\infty}\frac{d}{dx}H_n(x-x_0)=\delta(x-x_0)$$

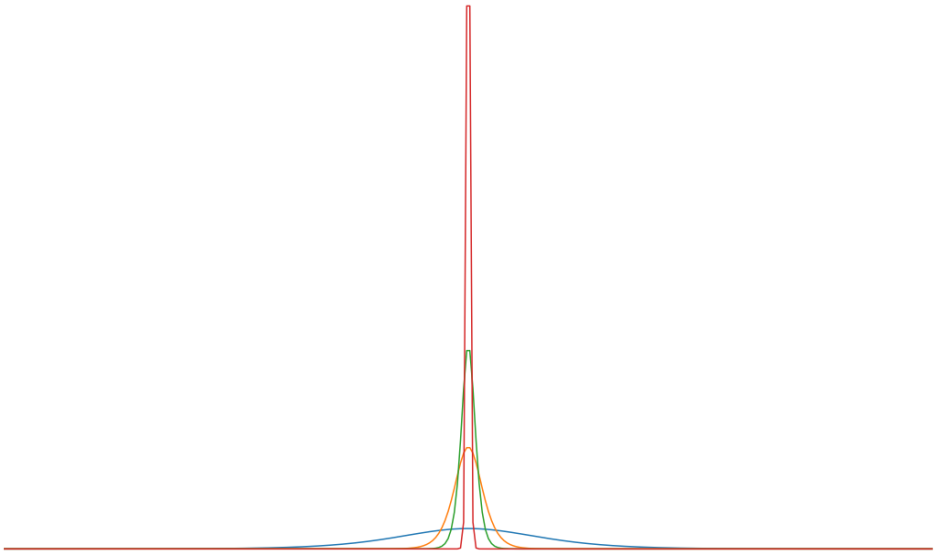


Figura 12 – Derivada de H_n com $n=1$ em azul a $n=50$ em vermelho.

Com a derivada da função degrau de Heaviside estamos em condições de analisar derivadas de funções descontínuas. Para tanto vamos tomar a seguinte função:

$$f(x)=\begin{cases} f_1(x), & x<x_0 \\ f_2(x), & x\geq x_0 \end{cases}$$

que pode ser escrita utilizando a função de Heaviside na forma:

$$f(x)=f_1(x)+[f_2(x)-f_1(x)]H(x-x_0).$$

Vamos usar a notação $(')$ como derivada em relação ao argumento. Derivando a expressão acima, obtemos:

$f'(x)=f_1'(x)+[f_2'(x)-f_1'(x)]H(x-x_0)+\Delta f\,\delta(x-x_0)$ onde $\Delta f=f_2(x_0)-f_1(x_0)$ é a descontinuidade no ponto x_0. Usualmente, representamos a função delta de Dirac na forma $h\delta(x-x_o)$ em um gráfico como uma seta no ponto x_0 com a altura h. A utilização das funções delta de Dirac generaliza o formalismo das distribuições de probabilidade quer sejam contínuas, discretas ou mistas. Além disso, resolve também o problema da dimensão. Notamos que $F(x)$ é probabilidade e, portanto, adimensional. Por outro lado $f(x)=\frac{dF}{dx}$ é probabilidade por unidade de x com dimensão $[f]=\left[\frac{1}{x}\right]$, portanto uma FDP da forma $f(x)=q\delta(x)+p\delta(x-1)$, do tipo Bernoulli, jogar uma moeda com probabilidade q de zero,

cara, e p de um, coroa, tem a dimensão correta de $1/x$ mesmo com q e p adimensionais, por causa da Delta de Dirac.

2.4.4 Momentos

A esperança de uma variável aleatória x é dada por: $E[x]=\int_{-\infty}^{+\infty} x\, f(x)\,dx$ onde $f(x)$ é a FDP. No caso discreto, em que a variável só pode assumir valores $x_1, x_2, \cdots, x_n$ a FDP será dada por $f(x)=\sum_{j=1}^{n} p_i \delta(x-x_i)$ com $p_i \geq 0$ e $\sum_{j=1}^{n} p_i = 1$. Nesse caso a esperança será dada por $E[x]=\sum_{j=1}^{n} p_i \int_{-\infty}^{+\infty} x\,\delta(x-x_i)\,dx = \sum_{j=1}^{n} p_i x_i$, que é a expressão da esperança de distribuições discretas. A esperança de uma função $g(x)$ da v.a. x é dada por $E[g(x)]=\int_{-\infty}^{+\infty} g(x)\,f(x)dx$.

O momento de ordem n é um exemplo da esperança de funções da variável x definido na forma: $M_n = E[x^n] = \int_{-\infty}^{+\infty} x^n f(x)dx.$ O momento de ordem zero vale $M_0 = 1$, que é a área da FDP normalizada e o M_1 é a esperança de x, denotado pela letra grega μ, ou seja, $M_1 = E[x] = \mu$. Já os *momentos centrados* são definidos por:

$$m_n = E[(x-\mu)^n] = \int_{-\infty}^{+\infty} (x-\mu)^n f(x)dx$$

Com essa definição $m_o = 1$ e $m_1 = 0$ sempre, para qualquer FDP.

Podemos encontrar uma relação entre os momentos centrados e não centrados utilizando expansão binomial:

$$m_n = \sum_{k=0}^{n} \frac{n!}{k!(n-k)!}(-1)^{n-k}\,\mu^{n-k} \int_{-\infty}^{+\infty} x^k f(x)dx$$

Reconhecendo a integral como o momento não centrado M_k, temos a relação entre momentos centrados e não centrados:

$$m_n = \sum_{k=0}^{n} \binom{n}{k} (-1)^{n-k}\,\mu^{n-k} M_k.$$

Com ela podemos mostrar que $m_2 = M_2 - \mu^2 = \sigma^2$. O m_2 é chamado de variância de uma distribuição de probabilidade e denotado por σ^2, e σ é chamado de desvio padrão. Para encontrar a relação inversa somamos e subtraímos μ de x dentro da potência n na expressão

da esperança e fazemos a expansão binomial:

$$M_n = E[(x-\mu+\mu)^n] = \sum_{k=0}^{n} \frac{n!}{k!(n-k)!} \mu^{n-k} \int_{-\infty}^{+\infty} (x-\mu)^k f(x) dx$$

reconhecemos esta integral como sendo o momento centrado m_k, portanto:

$$M_n = \sum_{k=0}^{n} \binom{n}{k} \mu^{n-k} m_k$$

2.4.4.1 Função Geradora dos Momentos

Vamos considerar agora a seguinte função chamada de *função geradora dos momentos*:

$$\mathbb{M}(t) = E\left[e^{tx}\right] = \int_{-\infty}^{+\infty} e^{tx} f(x) dx$$

Expandindo a exponencial em série de Taylor obtemos:

$$\mathbb{M}(t) = \sum_{n=0}^{\infty} \frac{t^n}{n!} \int_{-\infty}^{+\infty} x^n f(x) dx = \sum_{n=0}^{\infty} \frac{M_n}{n!} t^n$$

Por outro lado, usando a série de Taylor da $\mathbb{M}(t)$ temos

$$\mathbb{M}(t) = \sum_{n=0}^{\infty} \frac{t^n}{n!} \frac{d^n \mathbb{M}}{dt^n}\bigg|_{t=0}$$

Portanto obtemos a seguinte relação:

$$M_n = \frac{d^n}{dt^n}\left[\mathbb{M}(t)\right]\bigg|_{t=0}$$

Para gerar os momentos centrados devemos multiplicar a função geradora dos momentos por $e^{-\mu t}$ e expandir:

$$e^{-\mu t}\mathbb{M}(t) = \int_{-\infty}^{+\infty} e^{t(x-\mu)} f(x) dx = \sum_{n=0}^{\infty} \frac{t^n}{n!} \int_{-\infty}^{+\infty} (x-\mu)^n f(x) dx = \sum_{n=0}^{\infty} \frac{m_n}{n!} t^n$$

Expandindo $e^{-\mu t}\mathbb{M}(t)$ e comparando com a expressão acima temos a seguinte expressão para os momentos centrados em temos da função geradora dos momentos:

$$m_n = \frac{d^n}{dt^n}\left[e^{-\mu t}\mathbb{M}(t)\right]\bigg|_{t=0}$$

2.4.5 Função Característica

Um problema que pode acontecer com a função geradora dos momentos é a divergência das integrais por causa da exponencial e^{tx}. Para evitar essa divergência podemos usar e^{itx} no lugar de e^{tx}, uma vez que $\left|e^{itx}\right|=1$, o que nos leva à *função característica*:

$$\varphi(t)=E\left[e^{itx}\right]=\int_{-\infty}^{+\infty}e^{itx}f(x)dx$$

Percebemos então que a função característica é a Transformada de Fourier da FDP com o fator 1 no lugar de $1/\sqrt{2\pi}$ usualmente utilizada pelos físicos. A relação entre a $f(x)$ e $\varphi(t)$ é biunívoca, ou seja, dada uma $f(x)$ existe apenas uma $\varphi(t)$ e vice versa, ou seja, $f(x) \leftrightarrow \varphi(t)$. A transformada inversa é dada por (ver apêndice A):

$$f(x)=\frac{1}{2\pi}\int_{-\infty}^{+\infty}e^{-itx}\varphi(t)dt$$

Na teoria da probabilidade as transformadas de Fourier são definidas com o fator 1 na ida e $\frac{1}{2\pi}$ na volta, ao contrário das transformadas simetrizadas dos físicos e engenheiros:

	Transformada de Fourier	Transformada de Fourier Inversa
Probabilidade	$\varphi(t)=\int_{-\infty}^{+\infty}e^{itx}f(x)dx$	$f(x)=\frac{1}{2\pi}\int_{-\infty}^{+\infty}e^{-itx}\varphi(t)dt$
Física	$\varphi(t)=\frac{1}{\sqrt{2\pi}}\int_{-\infty}^{+\infty}e^{itx}f(x)dx$	$f(x)=\frac{1}{\sqrt{2\pi}}\int_{-\infty}^{+\infty}e^{-itx}\varphi(t)dt$

Além disso os físicos e engenheiros lidam muito com ondas cujo paradigma são as ondas planas, descritas pela função $\varphi(x,t)=e^{i(kx-\omega t)}$. Por isso, as transformadas da coordenada espacial são definidas por $F(k)=\frac{1}{\sqrt{2\pi}}\int_{-\infty}^{+\infty}e^{ikx}f(x)dx$ enquanto as transformadas da coordenada temporal são definidas por $F(\omega)=\frac{1}{\sqrt{2\pi}}\int_{-\infty}^{+\infty}e^{-i\omega t}f(t)dx$. Se percebe então, que existe uma liberdade na escolha do Kernel da transformada como e^{ixt} ou e^{-ixt} e na escolha dos fatores entre a transformada e sua inversa $\varphi(t)=\alpha\int_{-\infty}^{+\infty}e^{\pm itx}f(x)dx$ e $f(x)=\beta\int_{-\infty}^{+\infty}e^{\mp itx}\varphi(t)dt$ de tal forma que o produto $\alpha\beta=\frac{1}{2\pi}$. Se a $\mathbb{M}(t)$ existe, então $\varphi(t)=\mathbb{M}(it)$, devido à troca de *t* por *it* na exponencial. Vemos que

$\mathbb{M}(0)=\varphi(0)=1$simplesmente porque a área da distribuição $f(x)$ deve ser unitária. Além disso, o módulo da função característica é sempre menor ou igual a 1, ou seja $|\varphi(t)|\leq 1$, pois:

$$\left|\varphi(t)\right|=\left|\int_{-\infty}^{+\infty}e^{itx}f(x)dx\right|\leq\int_{-\infty}^{+\infty}\left|e^{itx}f(x)\right|dx=\int_{-\infty}^{+\infty}\left|e^{itx}\right|f(x)dx=\int_{-\infty}^{+\infty}f(x)dx=1$$

Expandindo a função característica em série de Taylor, podemos escrevê-la em termos dos momentos:

$$\varphi(t)=\sum_{n=0}^{\infty}\frac{i^n t^n}{n!}\int_{-\infty}^{+\infty}x^n f(x)dx=\sum_{n=0}^{\infty}\frac{i^n M_n}{n!}t^n$$

Usando a definição da série de Taylor, como fizemos com a função geradora dos momentos, a φ tem a forma:

$$\varphi(t)=\sum_{n=0}^{\infty}\frac{t^n}{n!}\frac{d^n\varphi}{dt^n}\bigg|_{t=0}$$

assim obtemos uma expressão para os momentos através da função característica:

$$M_n=(-i)^n\frac{d^n}{dt^n}\varphi(t)\bigg|_{t=0}$$

Para os momentos centrados, basta repetir este procedimento multiplicando a função característica por $e^{-i\mu t}$ obtendo a expressão:

$$m_n=(-i)^n\frac{d^n}{dt^n}\left[e^{-i\mu t}\varphi(t)\right]\bigg|_{t=0}$$

2.4.6 Cumulantes

A expansão dos cumulantes é definida através da função característica na seguinte forma:

$$\ln\left[\varphi(t)\right]=\sum_{k=0}^{\infty}\frac{i^k c_k}{k!}t^k$$

Comparando com a série de Taylor teremos

$$c_k=(-i)^k\frac{d^k}{dt^k}\ln\left[\varphi(t)\right]\bigg|_{t=0}$$

Para um produto de funções características da forma $\varphi_z=\varphi_1\varphi_2\ldots\varphi_n$ o logaritmo será uma soma e os cumulantes vão somando (acumulando): $c_{k,z}=c_{k,1}+c_{k,2}+\ldots+c_{k,n}$.

Calculando as derivadas do logaritmo obtemos as seguintes expressões dos cumulantes em

termos dos momentos centrados e não centrados:

- $c_1 = -i\varphi^{-1}\varphi'|_0 = M_1 = \mu;$
- $c_2 = -(\varphi^{-1}\varphi'' - \varphi^{-2}\varphi'^2)|_0 = M_2 - \mu^2 = \sigma^2.$

Estas relações são utilizadas para demonstrar o teorema central do limite.

2.4.7 Distribuição Normal

Podemos definir uma distribuição normal através dos seus cumulantes como a distribuição com apenas dois cumulantes, $c_1 = \mu$ e $c_2 = \sigma^2$. Neste caso $\ln\left[\varphi_N(t)\right] = i\mu t - \frac{\sigma^2}{2}t^2$ e $\varphi_N(t) = e^{i\mu t - \frac{\sigma^2}{2}t^2}$.

Através da transformada de Fourier inversa podemos encontrar a FDP da distribuição normal

$$f(x) = \frac{1}{2\pi}\int_{-\infty}^{+\infty} \mathrm{e}^{-ixt}\mathrm{e}^{i\mu t - \frac{\sigma^2 t^2}{2}}\, dt$$

completando o quadrado e multiplicando e dividindo por $\sigma/\sqrt{2}$:

$$f(x) = \frac{e^{-\frac{[x-\mu]^2}{2\sigma^2}}}{2\pi}\frac{\sqrt{2}}{\sigma}\int_{-\infty}^{+\infty} e^{-\frac{\sigma^2}{2}\left(t + i\frac{[x-\mu]}{\sigma^2}\right)^2} d\left(\frac{\sigma t}{\sqrt{2}}\right) = \frac{e^{-\frac{[x-\mu]^2}{2\sigma^2}}}{\sqrt{2}\pi\sigma}\int_{-\infty}^{+\infty} e^{-u^2} du$$

Usando o fato de que $\int_{-\infty}^{+\infty} e^{-u^2} du = \sqrt{\pi}$ obtemos a FDP da distribuição normal dada por:

$$N_{\left[\mu,\sigma^2\right]}(x) = \frac{e^{-\frac{[x-\mu]^2}{2\sigma^2}}}{\sqrt{2\pi}\,\sigma}$$

Vemos na Figura 12 a distribuição normal para diferentes valores de μ e σ^2.

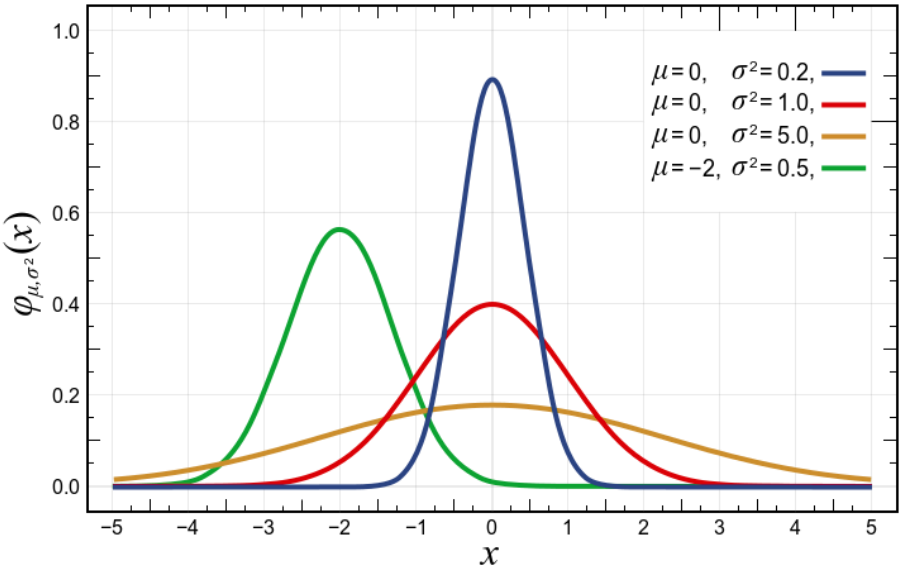


Figura 13 – Fonte: [20]. Distribuição Normal.

A distribuição normal padrão possui esperança nula (centrada em $x=0$) e variância unitária. É denotada por

$$N_{[0,1]}(x)=\frac{1}{\sqrt{2\pi}}\mathrm{e}^{-\frac{x^2}{2}}$$

e a sua distribuição cumulativa ou CDF é dada pela seguinte integral:

$$\Phi(x)=\frac{1}{\sqrt{2\pi}}\int_{-\infty}^{x}e^{-\frac{t^2}{2}}dt$$

vemos na Figura 14 os gráficos da distribuição normal cumulativa Φ_{μ,σ^2} para alguns μ e σ^2 diferentes.

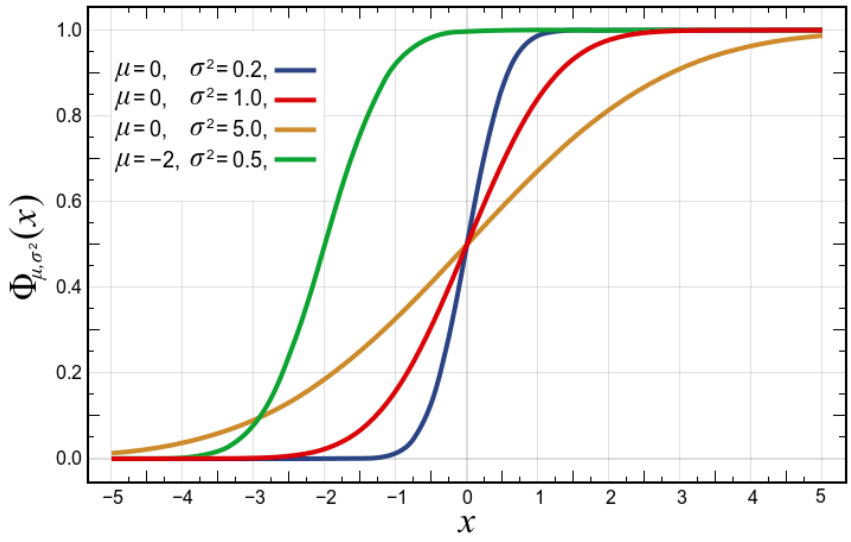


Figura 14 – Fonte: [20]. Distribuição Normal Cumulativa.

No caso da CDF padrão utilizaremos apenas a notação $\Phi(x)$ no lugar de $\Phi_{0,1}$.

2.4.8 Análise Multivariada

Seja a seguinte função vetorial de um conjunto $\vec{x}_{\boldsymbol{v}}(A):\Omega\to\mathbb{R}^n$, onde Ω é o espaço amostral, incluindo todas as possibilidades, de modo que $P(\Omega)=1$, cujas componentes são $x_1, x_2, \cdots, x_n$ e cada x_j é uma variável aleatória (v.a.). Assim $\{x_{vi}\le x_i\}$ e $\{x_{vj}\le x_j\}$ são dois eventos e $\{x_{vi}\le x_i\}+\{x_{vj}\le x_j\}=\{x_{vi}\le x_i, x_{vj}\le x_j\}$ é um evento. Nesse ponto vamos lidar apenas com o caso bivariado, ou seja, duas v.a.s, chamando uma v.a. de x e a outra de y. Extensão para n v.a.s é imediata.

2.4.8.1 Distribuição conjunta

A distribuição de probabilidade cumulativa para as duas variáveis, dados os dois eventos é

$$F(x,y)=P\{x_v\le x, y_v\le y\}$$

Possuindo a propriedade: $F(-\infty,y)=F(x,-\infty)=0$ e $F(+\infty,+\infty)=1$. Podemos demonstrar esta propriedade notando que:

$\{x_v\le-\infty, y_v\le y\}\subset\{x_v=-\infty\}\quad\rightarrow\quad P\{x_v\le-\infty, y_v\le y\}\le P\{x_v=-\infty\}=0$, então $0\ge P\{x_v\le-\infty, y_v\le y\}\le 0$ logo $F(-\infty,y)=0$. A demonstração também é válida para $\{y_v=-\infty\}$. Para a segunda parte basta notar que $\{x_v\le+\infty, y_v\le+\infty\}=\Omega$ logo $P\{x_v\le+\infty, y_v\le+\infty\}=P\{\Omega\}=1$.

2.4.8.2 Densidade de probabilidade conjunta

Definimos a fdp conjunta como:

$$f(x,y)=\frac{\partial^2 F(x,y)}{\partial x\partial y}.$$

A CDF é dada por:

$$F(x,y)=\int_{-\infty}^{x}\int_{-\infty}^{y} f(u,v)\,du\,dv$$

Também exigimos aqui que:

$$F(\infty,\infty)=\int_{-\infty}^{+\infty}\int_{-\infty}^{+\infty} f(x,y)\,dx\,dy=1$$

com $f(x,y)\geq 0$.

2.4.8.3 Operação esperança multivariada

Podemos definir a operação esperança para qualquer função escalar z das v.a.s x e y dada por $z=g(x,y)$. A esperança é:

$$E\left[g(x,y)\right]=\int_{-\infty}^{\infty} z\, f_z(z)\,dz=\int_{-\infty}^{\infty}\int_{-\infty}^{\infty} g(x,y)\, f(x,y)\,dx\,dy$$

Podemos verificar as propriedades da esperança:

1. $E[k]=k$

Se $g(x,y)=k$ é uma constante então:

$$E[k]=\int_{-\infty}^{\infty}\int_{-\infty}^{\infty} k\, f(x,y)\,dx\,dy=k\int_{-\infty}^{\infty}\int_{-\infty}^{\infty} f(x,y)\,dx\,dy=k;$$

2. $E\left[\alpha g(x,y)+\beta h(x,y)\right]=\alpha E\left[g(x,y)\right]+\beta E\left[h(x,y)\right]$

$$E\left[\alpha g(x,y)+\beta h(x,y)\right]=\alpha\int_{-\infty}^{\infty}\int_{-\infty}^{\infty} g(x,y)\, f(x,y)\,dx\,dy+\beta\int_{-\infty}^{\infty}\int_{-\infty}^{\infty} h(x,y)\, f(x,y)\,dx\,dy$$

de onde extraímos o caso trivial $E[x+y]=E[x]+E[y]$

2.4.8.4 Momentos conjuntos

Os momentos para o caso bivariado são dados por:

$$M_{kp}=E\left[x^k y^p\right]=\int_{-\infty}^{\infty}\int_{-\infty}^{\infty} x^k y^p\, f(x,y)\,dx\,dy$$

E os momentos centrados:

$$m_{kp} = E\left[\left(x-\mu_x\right)^k\left(y-\mu_y\right)^p\right] = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty}\left(x-\mu_x\right)^k\left(y-\mu_y\right)^p f\left(x,y\right)dxdy$$

Onde as esperanças são dadas pelos momentos:

$$M_{10} = \mu_x = E\left[x\right] = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} x\, f\left(x,y\right)dxdy$$

$$M_{01} = \mu_y = E\left[y\right] = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} y\, f\left(x,y\right)dxdy$$

Notamos imediatamente que: $M_{00} = m_{00} = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} f\left(x,y\right)dxdy = 1$ e que $m_{10} = m_{01} = 0$.

Os momentos centrados com nomes específicos são as variâncias:

$$V\left(x\right) = \sigma_x^2 = m_{20} = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty}\left(x-\mu_x\right)^2 f\left(x,y\right)dxdy$$

$$V\left(y\right) = \sigma_y^2 = m_{02} = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty}\left(y-\mu_y\right)^2 f\left(x,y\right)dxdy$$

Já a covariância é dada pelo momento de ordem 11:

$$\mathrm{cov}\left(x,y\right) = m_{11} = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty}\left(x-\mu_x\right)\left(y-\mu_y\right) f\left(x,y\right)dxdy$$

Note que dado que $f\left(x,y\right)\geq 0 \;\; \forall x,y$ a covariância é um produto interno $\mathrm{cov}\left(x,y\right) = \left\langle \left(x-\mu_x\right) \middle| \left(y-\mu_y\right) \right\rangle$. Também podemos associar a variância em termos da covariância na forma $V\left(x\right) = \mathrm{cov}\left(x,x\right) = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty}\left(x-\mu_x\right)^2 f\left(x,y\right)dxdy$ e $V\left(y\right) = \mathrm{cov}\left(y,y\right) = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty}\left(y-\mu_y\right)^2 f\left(x,y\right)dxdy$. Ou seja $\sigma_x^2 = \langle x|x\rangle$ e $\sigma_y^2 = \langle y|y\rangle$, logo $\sigma_x = \sqrt{\langle x|x\rangle} = \left\|\langle x|x\rangle\right\|$ e $\sigma_y = \sqrt{\langle y|y\rangle} = \left\|\langle y|y\rangle\right\|$

2.4.8.5 Propriedades da Matriz de variância-covariância

A covariância tem as seguintes propriedades:

1. $\mathrm{cov}(x,y)=\mathrm{cov}(y,x)$;
2. $\mathrm{cov}(x,y)=E[xy]-E[x]E[y];$
3. $\mathrm{cov}(\alpha x,\beta y)=\alpha\beta\,\mathrm{cov}(x,y);$
4. $\mathrm{cov}(x,k)=0$ onde k é uma constante. As demonstrações estão no apêndice B. Estas dão origem as seguintes propriedades da variância:

1. $V(x)=E\left[x^2\right]-\left(E[x]\right)^2$;
2. $V[kx]=k^2V[x]$;
3. $V[\alpha+\beta x]=\beta^2V[x]$;

Definindo a matriz de $n\times n$ $V_{ij}=\mathrm{cov}\left(x_i,x_j\right)$ como a matriz de variância-covariância percebemos que a diagonal representa a variância de cada v.a. Na prática calculamos a matriz de covariância de um conjunto de dados reais na forma discreta com a fórmula $V_{ij}=\frac{1}{n-1}\sum_k\left(x_{ik}-\overline{x}_i\right)\left(x_{jk}-\overline{x}_j\right)$, onde x_{ik} é o valor da k-ésima observação da v.a. x_i e $\overline{x}_i=\frac{1}{n}\sum_k x_{ik}$ é um estimador para μ_i, que substitui a operação esperança: $E[Q]\to\overline{Q}=\frac{1}{n}\sum_k Q_k$. Este estimador se aproxima da média quando o número de pontos da amostragem tende ao infinito. Entretanto, as propriedades da matriz de variância-covariância continuam válidas:

1. É simétrica: $V_{ij}=V_{ji}$
2. É uma matriz definida positiva:

Para que a matriz seja definida positiva, o produto a seguir deve ser positivo:

$$\vec{\overline{h}}\,M\,\vec{h}=\left(h_1,h_2,\cdots,h_n\right)M\begin{pmatrix}h_1\\h_2\\\vdots\\h_n\end{pmatrix}=\sum_i\sum_j M_{ij}h_ih_j>0$$

onde $\vec{h}$ é um vetor qualquer não nulo. Para provar, consideremos $z=\sum_i h_i\left(x_i-\mu_i\right)\neq 0$ de

forma que $z^2 = \sum_i \sum_j \left(x_i - \mu_i\right)\left(x_j - \mu_j\right) h_i h_j > 0$. Então, como $E\left[z^2\right] > 0$ então

$$\sum_i \sum_j E\left[\left(x_i - \mu_i\right)\left(x_j - \mu_j\right)\right] h_i h_j = \sum_i \sum_j V_{ij} h_i h_j > 0$$

2.4.8.6 Autovalores e autovetores da matriz de covariância

Estas duas propriedades são importantes para que seja possível diagonalizar e extrair os autovalores e autovetores da matriz de covariância. O fato de que se trata de uma matriz real e simétrica implica em que seus autovalores serão reais. Além disso, o fato de que é uma matriz definida positiva significa que os autovalores serão todos positivos. A variância de uma combinação linear de $x's$ do tipo $z = \sum_k \alpha_k x_k$ é dada por $\sigma_z^2 = \vec{\alpha}' V \vec{\alpha}$ onde o vetor linha $\vec{\alpha}'$ é o transposto do vetor coluna $\vec{\alpha}$.

Prova: $\mu_z = E[z] = \sum_k \alpha_k E[x_k] = \sum_k \alpha_k \mu_k$

$$z - \mu_z = \sum_k \alpha_k \left(x_k - \mu_k\right).$$

Assim $\left(z - \mu_z\right)^2 = \sum_k \sum_k \alpha_\ell \alpha_k \left(x_\ell - \mu_\ell\right)\left(x_k - \mu_k\right)$

Logo $\sigma_z^2 = E\left[\left(z - \mu_z\right)^2\right] = \sum_k \sum_k \alpha_\ell \alpha_k E\left[\left(x_\ell - \mu_\ell\right)\left(x_k - \mu_k\right)\right]$

Ou seja $\sigma_z^2 = \sum_k \sum_\ell \alpha_\ell \alpha_k V_{k\ell} = \vec{\alpha}' V \vec{\alpha}$

Considere agora o problema de encontrar um vetor $\vec{\alpha}$ unitário, $\sum_k \alpha_k^2 = 1$, que maximiza a variância. Em termos dos multiplicadores de Lagrange esse problema é descrito pela lagrangeana

$$L = \sum_k \sum_\ell \alpha_\ell \alpha_k V_{k\ell} - \lambda\left(\sum_k \alpha_k^2 - 1\right)$$

Cuja solução é dada por $\frac{\partial L}{\partial \lambda} = 0 \quad e \quad \frac{\partial L}{\partial \alpha_j} = 0 \quad \forall j$.

$$\frac{\partial L}{\partial \alpha_j} = \sum_k \sum_\ell \frac{\partial \alpha_\ell}{\partial \alpha_j} \alpha_k V_{k\ell} + \sum_k \sum_\ell \alpha_\ell \frac{\partial \alpha_k}{\partial \alpha_j} V_{k\ell} - \lambda \sum_k \frac{\partial \alpha_k^2}{\partial \alpha_j} = \sum_k \sum_\ell \delta_{\ell j} \alpha_k V_{k\ell} + \sum_k \sum_\ell \alpha_\ell \delta_{kj} V_{k\ell} - 2\lambda \sum_k \alpha_k \delta_{kj}$$

$$\frac{\partial L}{\partial \alpha_j} = \sum_k \alpha_k V_{kj} + \sum_\ell \alpha_\ell V_{j\ell} - 2\lambda\alpha_j = 2\sum_\ell V_{j\ell}\alpha_\ell - 2\lambda\alpha_j = 0$$

Ou seja, em termos matriciais, esse problema é escrito como $V\vec{\alpha} = \lambda\vec{\alpha}$, e os vetores $\vec{\alpha}$ são os autovetores da matriz de covariância. Agora, como a matriz V é real e simétrica, seus autovalores são reais e seus autovetores ortogonais. Ela pode, então, ser diagonalizada através dos seus autovetores normalizados na forma $S'VS = D_\lambda$, onde $S = \begin{pmatrix} \vec{v}_1 & \vec{v}_2 & \cdots & \vec{v}_n \end{pmatrix}$ com $V\ \vec{v}_k = \lambda_k \vec{v}_k$, ou seja $\vec{v}_k$ é um autovetor de V normalizado $\langle \vec{v}_k | \vec{v}_\ell \rangle = \delta_{k\ell}$. Assim $S' = \begin{pmatrix} \vec{v}_1' \\ \vec{v}_2' \\ \vdots \\ \vec{v}_n' \end{pmatrix}$ e $(S'S)_{k\ell} = \vec{v}_k' \vec{v}_\ell = \langle v_k | v_\ell \rangle = \delta_{k\ell}$, ou seja, $S'S = I \ \rightarrow \ S' = S^{-1}$.

Logo podemos re-escrever $\sigma_z^2 = \vec{\alpha}'V\vec{\alpha} = \vec{\alpha}'SS'VSS'\vec{\alpha} = (S'\vec{\alpha})'\, D_\lambda\, (S'\vec{\alpha}) = \vec{\beta}' D_\lambda \vec{\beta}$ chamando $\vec{\beta} = S'\vec{\alpha}$ temos que $\sigma_z^2 = \vec{\beta}' D_\lambda \vec{\beta}$. Quando $\vec{\beta} = \vec{v}_\ell$ então $\sigma_z^2 = \lambda_\ell$. Como, além de simétrica, a matriz de covariância é definida positiva, todos os autovalores serão positivos. Ordenando os autovalores do maior para o menor, o primeiro autovetor é a combinação linear de $\vec{x}$ com a maior variância, o segundo a combinação linear com a segunda maior variância e assim por diante. Essa é a base da análise de componentes principais. Cada autovetor da matriz de covariância forma uma componente principal. Usualmente, 2 ou 3 componentes principais explicam quase toda a variação observada na amostra e em lugar de lidar com n autovetores podemos reduzir a análise para 3 ou 4.

2.4.8.7 Regressão linear via mínimos quadrados

Suponha que uma variável dependente *y* seja uma função linear das variáveis X_1, X_2, ... X_k, da forma $y_i = a_1 X_{1i} + a_2 X_{2i} + \cdots + a_k X_{ki} + e_i$., onde $\boldsymbol{e_i}$ são os erros. Supõe-se que os erros são independentes e têm esperança nula, i.e., $\boldsymbol{E}[\boldsymbol{e_i}] = 0$ e $\boldsymbol{E}[\boldsymbol{e_i e_j}] = \boldsymbol{\sigma^2 \delta_{ij}}$. Queremos encontrar um bom estimador não tendencioso para os coeficientes $(\hat{\boldsymbol{a}}_1, \hat{\boldsymbol{a}}_2, \cdots, \hat{\boldsymbol{a}}_k)$. Denotamos um estimador com um circunflexo ^ em cima, significando uma regra para calcular o valor da grandeza. Por exemplo, o estimador do μ de uma distribuição Normal $N(\mu, \sigma^2)$ com centro em μ e variância σ^2, é dado pela regra $\hat{\boldsymbol{\mu}} = \dfrac{1}{\boldsymbol{n}} \sum_{\boldsymbol{i}} \boldsymbol{x_i}$.

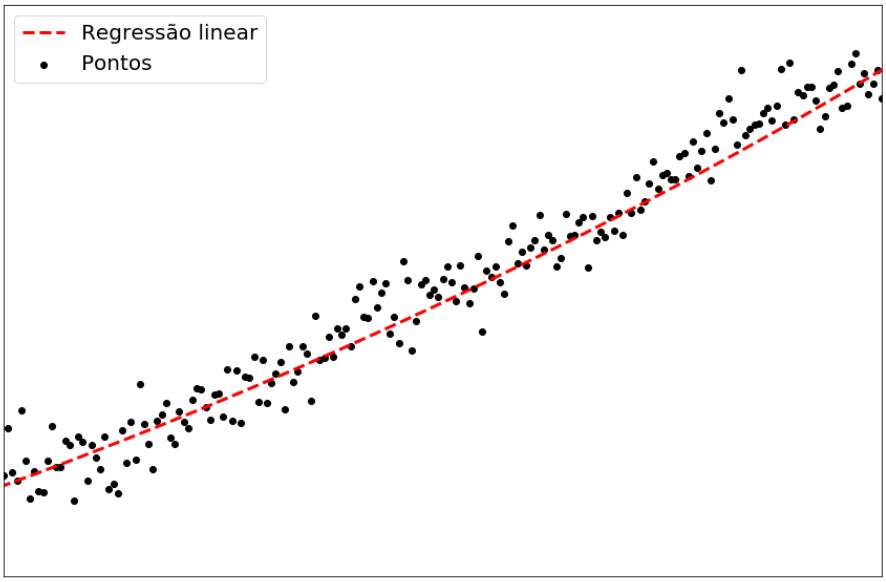


Figura 15 – Regressão linear.

Se cada x_i é dado por $\boldsymbol{x_i = \mu + e_i}$, com $\boldsymbol{E[e_i] = 0}$, então $\boldsymbol{E[\hat{\mu}] = E\left[\frac{1}{n}\sum_i x_i\right] = \frac{1}{n}\sum_i E[x_i] = \frac{1}{n}\sum_i E[\mu + e_i] = \frac{1}{n}\sum_i \{\mu + E[e_i]\} = \frac{1}{n}\sum_i \mu = \frac{n\mu}{n} = \mu}$, logo $\boldsymbol{\hat{\mu} = \frac{1}{n}\sum_i x_i}$ é um estimador não tendencioso de μ. Além de não tendencioso procuramos estimadores de variância mínima. O estimador de MQO (mínimos quadrados ordinários) é um estimador não tendencioso de variância mínima. Esse estimador é obtido pela regra de escolher $(\boldsymbol{a_1, a_2, \cdots, a_k})$ de forma a minimizar a Soma dos Quadrados dos Resíduos (SQR).

Temos n observações da amostragem y e dos valores de X_1, X_2, ... X_k. As equações $\boldsymbol{y_i = a_1 X_{1i} + a_2 X_{2i} + \cdots + a_k X_{ki} + e_i}$ para todos os i's podem ser escritas como:

$$\begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{bmatrix} = \begin{bmatrix} X_{11} & X_{21} & \cdots & X_{k1} \\ X_{12} & X_{22} & \cdots & X_{k2} \\ \vdots & \vdots & & \vdots \\ X_{1n} & X_{2n} & \cdots & X_{kn} \end{bmatrix} \begin{bmatrix} a_1 \\ a_2 \\ \vdots \\ a_k \end{bmatrix} + \begin{bmatrix} \mathrm{e}_1 \\ \mathrm{e}_2 \\ \vdots \\ e_n \end{bmatrix}$$

Em termos de vetores e matrizes: $\vec{y} = \boldsymbol{X}\,\vec{a} + \vec{e}$ onde $\vec{\bar{y}} = [y_1 \quad y_2 \quad \cdots \quad y_n]$ e $\vec{\bar{e}} = [e_1 \quad e_2 \quad \cdots \quad e_n]$ são vetores coluna com n elementos, $\vec{\bar{a}} = [a_1 \quad a_2 \quad \cdots \quad a_k]$ um vetor coluna com k elementos e X é a matriz $n \times k$:

$$\boldsymbol{X} = \begin{bmatrix} \boldsymbol{X}_{11} & \boldsymbol{X}_{21} & \cdots & \boldsymbol{X}_{k1} \\ \boldsymbol{X}_{12} & \boldsymbol{X}_{22} & \cdots & \boldsymbol{X}_{k2} \\ \vdots & \vdots & & \vdots \\ \boldsymbol{X}_{1n} & \boldsymbol{X}_{2n} & \cdots & \boldsymbol{X}_{kn} \end{bmatrix}$$

A matriz $n \times n$ simétrica dada pela multiplicação dos resíduos é:

$$\vec{\boldsymbol{e}}\,\bar{\vec{\boldsymbol{e}}} = \begin{bmatrix} \mathrm{e}_1 \\ \mathrm{e}_2 \\ \vdots \\ \boldsymbol{e}_n \end{bmatrix} \begin{bmatrix} \mathrm{e}_1, \mathrm{e}_2, \cdots, \mathrm{e}_n \end{bmatrix} = \begin{bmatrix} \boldsymbol{e}_1 \boldsymbol{e}_1 & \boldsymbol{e}_1 \boldsymbol{e}_2 & \cdots & \boldsymbol{e}_1 \boldsymbol{e}_n \\ \boldsymbol{e}_2 \boldsymbol{e}_1 & \boldsymbol{e}_2 \boldsymbol{e}_2 & \cdots & \boldsymbol{e}_2 \boldsymbol{e}_n \\ \vdots & \vdots & & \vdots \\ \boldsymbol{e}_n \boldsymbol{e}_1 & \boldsymbol{e}_n \boldsymbol{e}_2 & \cdots & \boldsymbol{e}_n \boldsymbol{e}_n \end{bmatrix}$$

que possui a seguinte propriedade:

$$\boldsymbol{E}[\vec{\boldsymbol{e}}\,\bar{\vec{\boldsymbol{e}}}] = \begin{bmatrix} \boldsymbol{E}[\boldsymbol{e}_1 \boldsymbol{e}_1] & \boldsymbol{E}[\boldsymbol{e}_1 \boldsymbol{e}_2] & \cdots & \boldsymbol{E}[\boldsymbol{e}_1 \boldsymbol{e}_n] \\ \boldsymbol{E}[\boldsymbol{e}_2 \boldsymbol{e}_1] & \boldsymbol{E}[\boldsymbol{e}_2 \boldsymbol{e}_2] & \cdots & \boldsymbol{E}[\boldsymbol{e}_2 \boldsymbol{e}_n] \\ \vdots & \vdots & & \vdots \\ \boldsymbol{E}[\boldsymbol{e}_n \boldsymbol{e}_1] & \boldsymbol{E}[\boldsymbol{e}_n \boldsymbol{e}_2] & \cdots & \boldsymbol{E}[\boldsymbol{e}_n \boldsymbol{e}_n] \end{bmatrix} = \begin{bmatrix} \boldsymbol{\sigma}^2 & 0 & \cdots & 0 \\ 0 & \boldsymbol{\sigma}^2 & \cdots & 0 \\ \vdots & \vdots & & \vdots \\ 0 & 0 & \cdots & \boldsymbol{\sigma}^2 \end{bmatrix} = \boldsymbol{\sigma}^2 \boldsymbol{I}_{nxn}$$

Quando $\boldsymbol{E}[\vec{\boldsymbol{e}}\,\bar{\vec{\boldsymbol{e}}}] = \boldsymbol{\sigma}^2 \boldsymbol{I}_{nxn}$ afirmamos que existe homocedasticidade, ou seja, a variância nos resíduos é homogênea, independente de *x*. A SQR definida por $\boldsymbol{SQR} = \sum_{\boldsymbol{i}} \boldsymbol{e}_{\boldsymbol{i}} \boldsymbol{e}_{\boldsymbol{i}}$ e pode ser escrita pelo produto escalar:

$$\boldsymbol{SQR} = \bar{\vec{\boldsymbol{e}}}\,\vec{\boldsymbol{e}} = \begin{bmatrix} \mathrm{e}_1, \mathrm{e}_2, \cdots, \mathrm{e}_n \end{bmatrix} \begin{bmatrix} \mathrm{e}_1 \\ \mathrm{e}_2 \\ \vdots \\ \boldsymbol{e}_n \end{bmatrix}$$

O estimador MQO requer que SQR seja mínima. Da Primeira equação extraímos: $\vec{\boldsymbol{e}} = \vec{\boldsymbol{y}} - \boldsymbol{X}\,\vec{\boldsymbol{a}}$, logo as transpostas são $\bar{\vec{\boldsymbol{e}}} = \bar{\vec{\boldsymbol{y}}} - \bar{\vec{\boldsymbol{a}}}\,\bar{\boldsymbol{X}}$ com $\bar{\vec{\boldsymbol{y}}} = [\boldsymbol{y}_1, \boldsymbol{y}_2, \cdots, \boldsymbol{y}_n]$, $\bar{\vec{\boldsymbol{a}}} = [\boldsymbol{a}_1, \boldsymbol{a}_2, \cdots, \boldsymbol{a}_k]$ e $\bar{\boldsymbol{X}}$ a matriz $k \times n$:

$$\bar{\boldsymbol{X}} = \begin{bmatrix} \boldsymbol{X}_{11} & \boldsymbol{X}_{12} & \cdots & \boldsymbol{X}_{1n} \\ \boldsymbol{X}_{21} & \boldsymbol{X}_{22} & \cdots & \boldsymbol{X}_{2n} \\ \vdots & \vdots & & \vdots \\ \boldsymbol{X}_{k1} & \boldsymbol{X}_{k2} & \cdots & \boldsymbol{X}_{kn} \end{bmatrix}$$

Aqui usamos o fato de que $\overline{\boldsymbol{AB}} = \bar{\boldsymbol{B}}\,\bar{\boldsymbol{A}}$. Então:

$$\boldsymbol{SQR} = \bar{\vec{\boldsymbol{e}}}\,\vec{\boldsymbol{e}} = [\bar{\vec{\boldsymbol{y}}} - \bar{\vec{\boldsymbol{a}}}\,\bar{\boldsymbol{X}}][\vec{\boldsymbol{y}} - \boldsymbol{X}\,\vec{\boldsymbol{a}}] = \bar{\vec{\boldsymbol{y}}}\,\vec{\boldsymbol{y}} - \bar{\vec{\boldsymbol{a}}}\,\bar{\boldsymbol{X}}\,\vec{\boldsymbol{y}} - \bar{\vec{\boldsymbol{y}}}\,\boldsymbol{X}\,\vec{\boldsymbol{a}} + \bar{\vec{\boldsymbol{a}}}\,\bar{\boldsymbol{X}}\boldsymbol{X}\,\vec{\boldsymbol{a}}$$

Agora, $\bar{\vec{\boldsymbol{a}}}\,\bar{\boldsymbol{X}}\,\vec{\boldsymbol{y}}$ é um escalar de modo que $\bar{\vec{\boldsymbol{a}}}\,\bar{\boldsymbol{X}}\,\vec{\boldsymbol{y}} = \overline{\bar{\vec{\boldsymbol{a}}}\,\bar{\boldsymbol{X}}\,\vec{\boldsymbol{y}}} = \bar{\vec{\boldsymbol{y}}}\,\boldsymbol{X}\,\vec{\boldsymbol{a}}$. O estimador de MQO para $\vec{\boldsymbol{a}}$ é o vetor $\vec{\boldsymbol{a}}$ que minimiza $\boldsymbol{SQR} = \bar{\vec{\boldsymbol{y}}}\,\vec{\boldsymbol{y}} - 2\bar{\vec{\boldsymbol{a}}}\,\bar{\boldsymbol{X}}\,\vec{\boldsymbol{y}} + \bar{\vec{\boldsymbol{a}}}\,\bar{\boldsymbol{X}}\boldsymbol{X}\,\vec{\boldsymbol{a}}$, ou seja, o vetor $\vec{\boldsymbol{a}}$ solução das *k*

equações $\frac{\partial}{\partial \boldsymbol{a}_n} \boldsymbol{SQR} = 0 \quad \vee \boldsymbol{n} \in \{1, 2, \cdots, \boldsymbol{k}\}$. Para derivar em relação a componente a_n é melhor reescrever a equação como:

$$\boldsymbol{SQR} = \sum_x \boldsymbol{y}_i \boldsymbol{y}_i - 2 \sum_i \sum_j \boldsymbol{a}_i \bar{\boldsymbol{X}}_{ij} \boldsymbol{y}_j + \sum_i \sum_j \boldsymbol{a}_i (\bar{\boldsymbol{X}}\boldsymbol{X})_{ij} \boldsymbol{a}_j$$

logo:

$$\frac{\partial}{\partial \boldsymbol{a}_n} \boldsymbol{SQR} = -2 \sum_i \sum_j \frac{\partial \boldsymbol{a}_i}{\partial \boldsymbol{a}_n} \bar{\boldsymbol{X}}_{ij} \boldsymbol{y}_j + \sum_i \sum_j \frac{\partial \boldsymbol{a}_i}{\partial \boldsymbol{a}_n} (\bar{\boldsymbol{X}}\boldsymbol{X})_{ij} \boldsymbol{a}_j + \sum_i \sum_j \boldsymbol{a}_i (\bar{\boldsymbol{X}}\boldsymbol{X})_{ij} \frac{\partial \boldsymbol{a}_j}{\partial \boldsymbol{a}_n} =$$
$$= -2 \sum_i \sum_j \boldsymbol{\delta}_{in} \bar{\boldsymbol{X}}_{ij} \boldsymbol{y}_j + \sum_i \sum_j \boldsymbol{\delta}_{in} (\bar{\boldsymbol{X}}\boldsymbol{X})_{ij} \boldsymbol{a}_j + \sum_i \sum_j \boldsymbol{a}_i (\bar{\boldsymbol{X}}\boldsymbol{X})_{ij} \boldsymbol{\delta}_{jn} =$$
$$= -2 \sum_j \bar{\boldsymbol{X}}_{nj} \boldsymbol{y}_j + \sum_j (\bar{\boldsymbol{X}}\boldsymbol{X})_{nj} \boldsymbol{a}_j + \sum_i \boldsymbol{a}_i (\bar{\boldsymbol{X}}\boldsymbol{X})_{in}$$

Agora usamos o teorema de que toda matriz do tipo $\boldsymbol{C} = \boldsymbol{A}\bar{\boldsymbol{A}}$ é simétrica, i.e., $\boldsymbol{C} = \bar{\boldsymbol{C}}$, pois $\bar{\boldsymbol{C}} = \overline{\boldsymbol{A}\bar{\boldsymbol{A}}} = \bar{\bar{\boldsymbol{A}}}\,\bar{\boldsymbol{A}} = \boldsymbol{A}\bar{\boldsymbol{A}} = \boldsymbol{C}$, de modo que $(\bar{\boldsymbol{X}}\boldsymbol{X})_{in} = (\bar{\boldsymbol{X}}\boldsymbol{X})_{ni}$ e $\sum_j (\bar{\boldsymbol{X}}\boldsymbol{X})_{nj} \boldsymbol{a}_j + \sum_i (\bar{\boldsymbol{X}}\boldsymbol{X})_{ni} \boldsymbol{a}_i = 2 \sum_i (\bar{\boldsymbol{X}}\boldsymbol{X})_{ni} \boldsymbol{a}_i$. Portanto a condição de mínimo é que:

$$\frac{\partial}{\partial \boldsymbol{a}_n} \boldsymbol{SQR} = -2 \sum_j \bar{\boldsymbol{X}}_{nj} \boldsymbol{y}_j + 2 \sum_j (\bar{\boldsymbol{X}}\boldsymbol{X})_{ni} \boldsymbol{a}_i = 0 \qquad \forall\ \boldsymbol{i}$$

que pode ser reescrita matricialmente como $(\bar{\boldsymbol{X}}\boldsymbol{X})\vec{\boldsymbol{a}} = \bar{\boldsymbol{X}}\,\vec{\boldsymbol{y}}$. Assim:

$$\hat{\vec{\boldsymbol{a}}} = (\bar{\boldsymbol{X}}\boldsymbol{X})^{-1} \bar{\boldsymbol{X}}\,\vec{\boldsymbol{y}}$$
$$\bar{\hat{\vec{\boldsymbol{a}}}} = \bar{\vec{\boldsymbol{y}}} \boldsymbol{X} (\bar{\boldsymbol{X}}\boldsymbol{X})^{-1}$$

é o estimador de $\vec{\boldsymbol{a}}$ de MQO, onde $(\bar{\boldsymbol{X}}\boldsymbol{X})^{-1}$ é a matriz inversa de $(\bar{\boldsymbol{X}}\boldsymbol{X})$.

Provamos que $\hat{\vec{\boldsymbol{a}}}$ é um estimador não tendencioso utilizando as equações: $\hat{\vec{\boldsymbol{a}}} = (\bar{\boldsymbol{X}}\boldsymbol{X})^{-1} \bar{\boldsymbol{X}}\,\vec{\boldsymbol{y}}$ e $\vec{\boldsymbol{y}} = \boldsymbol{X}\vec{\boldsymbol{a}} + \vec{\boldsymbol{e}}$, unindo-as teremos:

$$\hat{\vec{\boldsymbol{a}}} = (\bar{\boldsymbol{X}}\boldsymbol{X})^{-1} \bar{\boldsymbol{X}} [\boldsymbol{X}\vec{\boldsymbol{a}} + \vec{\boldsymbol{e}}] = (\bar{\boldsymbol{X}}\boldsymbol{X})^{-1} (\bar{\boldsymbol{X}}\boldsymbol{X})\vec{\boldsymbol{a}} + (\bar{\boldsymbol{X}}\boldsymbol{X})^{-1} \bar{\boldsymbol{X}}\,\vec{\boldsymbol{e}} = \vec{\boldsymbol{a}} + (\bar{\boldsymbol{X}}\boldsymbol{X})^{-1} \bar{\boldsymbol{X}}\,\vec{\boldsymbol{e}}$$

Aplicando a operação esperança teremos então

$$\boldsymbol{E}[\hat{\vec{\boldsymbol{a}}}] = \boldsymbol{E}[\vec{\boldsymbol{a}}] + (\bar{\boldsymbol{X}}\boldsymbol{X})^{-1} \bar{\boldsymbol{X}}\,\boldsymbol{E}[\vec{\boldsymbol{e}}] = \vec{\boldsymbol{a}} + (\bar{\boldsymbol{X}}\boldsymbol{X})^{-1} \bar{\boldsymbol{X}}\,\vec{0}$$

logo $\boldsymbol{E}[\hat{\vec{\boldsymbol{a}}}] = \vec{\boldsymbol{a}}$ é um estimador não tendencioso de $\vec{\boldsymbol{a}}$.

Nota-se que estes coeficientes são obtidos automaticamente através de uma única operação [22].

2.4.8.8 Variáveis aleatórias independentes

Se os seguintes eventos $\{x_v \in A\}$ e $\{y_v \in B\}$ são independentes então $P\left[\{\{x_v \in A\}\{y_v \in B\}\}\right] = P\left[\{x_v \in A\}\right]P\left[\{y_v \in B\}\right]$. Neste caso então:

$F(x,y) = F_x(x)F_y(y)$ e $f(x,y) = f_x(x)\,f_y(y)$.

Definimos o espaço dos eventos da v.a. x como Ω_x e para y será Ω_y. Ao realizarmos um experimento conjunto, os eventos devem pertencer ao espaço amostral $\Omega = \Omega_x \times \Omega_y$ onde o resultado de um não deve interferir no outro. Temos que:

$x(\omega_1, \omega_2) = x(\omega_1)$ e $y(\omega_1, \omega_2) = y(\omega_2)$

Portanto, se as v.a.s x e y são independentes temos os seguintes teoremas:

Teorema 1: Dados x e y independentes, então $g(x)$ e $h(y)$ também devem ser independentes.

Prova: Se $\{x_v \in A\}$ e $\{y_v \in B\}$ são independentes, quaisquer dois sub-conjuntos destes serão independentes. Desta forma $\{g(x) \le g\} \subset \{x_v \in A\}$ e $\{h(y) \le h\} \subset \{y_v \in B\}$ será a condição para poder calcular as funções $g(x)$ e $h(y)$. Portanto se x e y são independentes, então $\forall\ g(x)$ e $h(y)$ também são independentes.

Teorema 2. Se x e y são independentes, então $E[xy] = E[x]E[y]$.

$$E[xy] = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} x\,y\,f(x,y)\,dxdy = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} x\,y\,f_x(x)\,f_y(y)\,dxdy$$

$$E[xy] = \left[\int_{-\infty}^{\infty} x f_x(x)dx\right]\left[\int_{-\infty}^{\infty} y\,f_y(y)dy\right]$$

$$E[xy] = E[x]E[y]$$

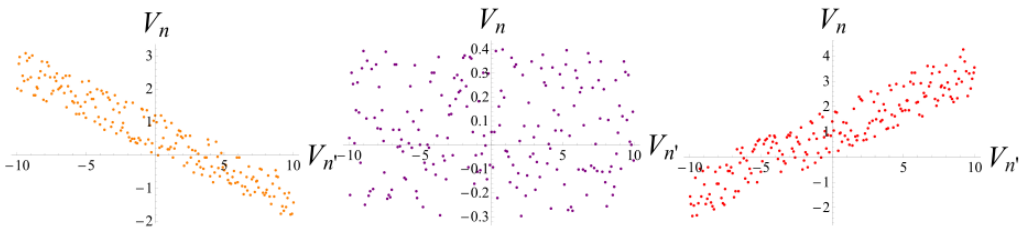
Teorema 3. Se x e y são independentes, então $\mathrm{cov}(x,y) = 0$.

$$\mathrm{cov}(x,y) = E[xy] - E[x]E[y] = E[x]E[y] - E[x]E[y] = 0$$

A covariância nos dá, então, a informação sobre a independência entre v.a.s. Neste caso $\mathrm{cov}(x,y) = 0$ para x e y independentes. Desta forma, investiguemos o que ocorre quando

$\text{cov}(x,y) \neq 0$. Os produtos $(x-\mu_x)(y-\mu_y)$ e $(x-\bar{x})(y-\bar{y})$ podem ser visualizados em um gráfico $(x-\mu_x)$ vs $(y-\mu_y)$ ou $(x-\bar{x})$ vs $(y-\bar{y})$ sendo positivos no Primeiro $(x \geq 0, y \geq 0)$ e terceiro quadrantes $(x < 0, y < 0)$ e negativos no segundo $(x < 0, y > 0)$ e quarto $(x > 0, y < 0)$ quadrantes.

A figura 16 (c) mostra uma concentração maior de pontos no Primeiro e terceiro quadrantes quando $\sum_i (x_i - \bar{x})(y_i - \bar{y})$ é positiva. Percebe-se o interligamento entre as v.a.s x e y pois ambas possuem a mesma tendência crescendo ou decrescendo juntas. O espalhamento da nuvem nos dá a informação de que esta não é uma tendência perfeita, existindo um grau de independência estatística entre as v.a.s. Na figura 16 (a) vemos uma nuvem de pontos cuja concentração se dá no segundo e quarto quadrantes com $\sum_i (x_i - \bar{x})(y_i - \bar{y})$ negativa, ou seja, uma covariância negativa.



(a) Negative covariance. (b) Null covariance. (c) Positive covariance.

Figura 16 – Fonte [21].

Percebe-se que y tende a decrescer quando a v.a. x cresce, e vice-versa. Quando as v.a.s são independentes, a nuvem se espalha igualmente pelos quatro quadrantes pois $\sum_i (x_i - \bar{x})(y_i - \bar{y}) = 0$ como mostra a figura 16 (b).

2.4.9 Coeficiente de Correlação

A covariância, de alguma forma, mede o grau de independência entre as v.a.s mas tem um problema de dimensão, pois $[\text{cov}(x,y)] = [x][y]$. Isso significa que será um número

dependente até das unidades utilizadas para x e y, variando se forem dados em litros ou metros-cúbicos, por exemplo. Para tornar a covariância em um número adimensional devemos dividi-la por algo com dimensão de $[x]\times[y]$. O coeficiente de correlação adimensional é definido na forma:

$$r_{xy} = r[x,y] = \frac{\mathrm{cov}(x,y)}{\sqrt{\mathrm{cov}(x,x)}\sqrt{\mathrm{cov}(y,y)}} = \frac{\mathrm{cov}(x,y)}{\sqrt{V[x]\,V[y]}}$$

Dessa definição notamos da nossa definição de produto interno e norma que:

$$r_{xy} = \frac{\langle x|y\rangle}{\sqrt{\langle x|x\rangle}\sqrt{\langle y|y\rangle}} = \frac{\langle x|y\rangle}{\|x\|\|y\|}$$

Além disso, a desigualdade de Schwartz nos garante que $-1 \le \frac{\langle x|y\rangle}{\|x\|\|y\|} \le +1$, logo $-1 \le r_{xy} \le +1$ e pode ser associada a um ângulo θ tal que $\frac{\langle x|y\rangle}{\|x\|\|y\|} = \cos\theta$. Isto significa que existirá um certo ângulo θ entre as variáveis *x* e *y* no qual $\mathrm{cov}(x,y) = \sqrt{V[x]V[y]}\cos\theta$. Se o coeficiente entre *x* e *y* é +1 (-1), as variáveis são totalmente correlacionadas (anticorrelacionadas), mas se o coeficiente for 0 as variáveis são independentes. Quando as variáveis são independentes, $\cos\theta = 0$ logo $\theta = \pi/2$ portanto *x* e *y* são ortogonais.

2.4.9.1 Propriedades do Coeficiente de Correlação

O coeficiente de correlação é imune a transformações lineares. Considere a transformação linear $x' = ax + b$. Nesse caso:

$\mathrm{cov}(x',y) = \mathrm{cov}(ax+b,y) = a\,\mathrm{cov}(x,y) + \mathrm{cov}(b,y) = a\,\mathrm{cov}(x,y)$

$\mathrm{cov}(x',x') = \mathrm{cov}(ax+b,ax+b) = a^2\,\mathrm{cov}(x,x) + 2a\,\mathrm{cov}(x,b) + \mathrm{cov}(b,b) = a^2\,\mathrm{cov}(x,x)$ Nesse caso $r_{x'y} = \frac{\mathrm{cov}(x',y)}{\sqrt{\mathrm{cov}(x',x')}\sqrt{\mathrm{cov}(y,y)}} = \frac{\mathrm{a}}{\sqrt{a^2}}\frac{\mathrm{cov}(x,y)}{\sqrt{\mathrm{cov}(x,x)}\sqrt{\mathrm{cov}(y,y)}} = \mathrm{sign}(a)\,r_{xy}$ então $r_{x'y} = r_{xy}$ para qualquer $a > 0$.

Isto é importante pois medidas de um aparelho para outro usualmente mudam a escala e a

linha de base, ou seja, apresentam uma transformação linear. Uma mesma amostra medida no mesmo instrumento com diluições diferentes também implica em uma variação linear entre as medidas. Por isso, a menos que se usem padrões de calibração, muitas vezes complicados, e que usualmente exigem duas medidas realizadas quase simultaneamente, os valores das medidas em si não podem ser comparados diretamente. Uma amostra obtida e caracterizada anos atrás não pode ser comparada com uma amostra recente. Mas o coeficiente de correlação entre as mesmas é imune a essas variações e permite comparar duas amostras sem uma calibração, e medidas em momentos diferentes, ou até com equipamentos diferentes. Uma amostra de óleo cru, por exemplo, pode variar de propriedades ao longo do tempo, e seria importante uma caracterização tão logo fosse extraída do poço. Como é impossível sincronizar a extração de poços diferentes, medidas realizadas logo após a extração, em momentos diferentes, garantiria que as duas sejam comparadas de forma pareada. A única restrição é que diferenças entre as medidas seja apenas de uma transformação linear, o que usualmente é o caso com equipamentos que possuem escalas lineares.

2.4.9.2 Distância de Correlação

Definindo as *VA*s padronizadas $p = \frac{x - \mu_x}{\sigma_x}$ e $q = \frac{y - \mu_y}{\sigma_y}$, notamos que $E[p] = E[q] = 0$ assim como, que $E\left[p^2\right] = E\left[q^2\right] = 1$. Isso significa que se tratam de vetores unitários. Nesse caso $r_{xy} = \mathrm{cov}(p,q) = E[pq]$. Podemos definir uma distância entre as variáveis x e y através de $d_{xy}^2 = E\left[(p-q)^2\right] = E\left[p^2\right] + E\left[q^2\right] - 2E[pq]$, logo $d_{xy}^2 = 2\left(1 - r_{xy}\right)$. Então a grandeza $d_{xy} = \sqrt{2\left(1 - r_{xy}\right)}$ se comporta como uma distância, chamada de distância de correlação. Como $-1 \le r_{xy} \le +1$ a distância de correlação varia entre $0 \le d_{xy} \le 2$. Quanto maior a correlação menor a distância. Vale notar um ponto importante aqui. Para ser uma distância exigimos que se $d(x,y) = 0$ então $x = y$. Mas $d(x,y) = 0$ significa $r_{xy} = +1$. Duas *VA*s relacionadas por uma transformação linear $x_j = a\,y_j + b$ com $a > 0$ apresentam correlação $r_{xy} = +1$ embora $x \neq y$. Entretanto, as

duas variáveis $p = \frac{x - \mu_x}{\sigma_x}$ e $q = \frac{y - \mu_y}{\sigma_y}$ são idênticas. Assim, ao realizar um mesmo experimento em duas amostras diferentes podemos medir uma distância de similaridade entre as duas através da distância de correlação dada por $d_{xy} = \sqrt{2\left(1 - r_{xy}\right)}$, com $d_{xy} = 0$ significando similaridade total, $r_{xy} = +1$, $d_{xy} = \sqrt{2}$ significando independência entre as amostras, $r_{xy} = 0$, e $d_{xy} = 2$ a oposição total entre as amostras, $r_{xy} = -1$. Os 3 casos podem ser apresentados como a subtração e soma de vetores no círculo unitário da figura 17.

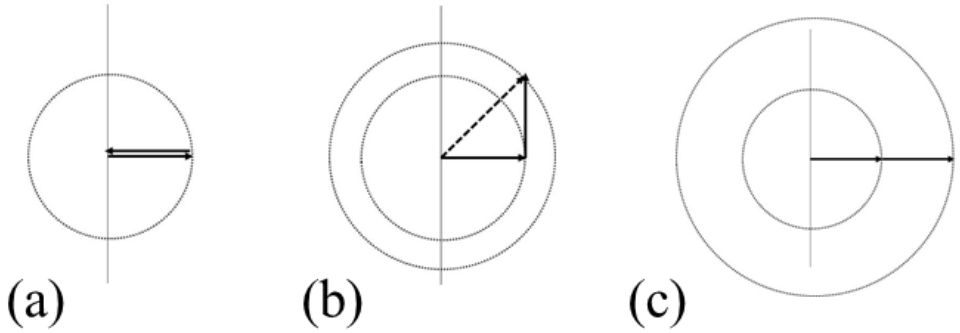


Figura 17 – Somas de vetores representando distâncias de correlação. (a) para $d_{xy} = 0$, (b) para $d_{xy} = \sqrt{2}$ e (c) para $d_{xy} = 2$.

Uma vez obtida uma matriz de distância entre diferentes amostras é possível fazer um MST de similaridade entre todas elas, e mesmo revelar clusters de amostras semelhantes. Utilizamos esse método no caso especial de similaridade de óleos por espectroscopia de excitação de fluorescência e em ações do mercado financeiro. O termo de distância de correlação d_{ij} nos dá o peso da aresta entre um vértice *i* e um segundo vértice *j*. Verificamos então que um grafo gerado a partir da matriz de distância de correlação é um grafo completo com todos os vértices conectados. Aplica-se então o algoritmo de PRIM para transformar o grafo completo em uma MST.

2.4.10 Teorema da Convolução

Suponha x e y independentes e defina a v.a. $z = x + y$. A função característica de z é dada por $\varphi_z(t) = E\left[e^{izt}\right] = E\left[e^{i(x+y)t}\right] = E\left[e^{ixt}e^{iyt}\right]$. Agora, como x e y são

independentes então e^{ixt} e e^{iyt} também são independentes e $E\left[e^{ixt}e^{iyt}\right]=E\left[e^{ixt}\right]E\left[e^{iyt}\right]=\varphi_x(t)\varphi_y(t)$. Assim, se $z=x+y$ então:

$\varphi_z(t)=\varphi_x(t)\varphi_y(t)$

Para encontrar a função característica da soma de v.a.s independentes basta multiplicar as devidas funções características. Essa é a base do Teorema Central do Limite.

2.4.11 Teorema Central do Limite

Seja uma variável aleatória $z=x_1+x_2+\ldots+x_n$ onde cada x_i é uma variável aleatória independente e com um n muito grande de variáveis. Assim a função característica de z é $\varphi_z(t)=\varphi_1(t)\varphi_2(t)\ldots\varphi_n(t)$. Se as variáveis fossem idênticas teríamos apenas $\varphi_z(t)=[\varphi_1(t)]^n$ e cada φ_i tem módulo menor ou igual a 1. Quando n é muito grande esta função tenderá a zero rapidamente para $t\neq 0$ e se concentrará em $t=0$ porque $\varphi(0)=1$ e $\left[\varphi(0)\right]^n=1$ sempre.

Expandindo o logaritmo em série de Taylor teremos a expansão dos cumulantes de forma que teremos

$$\varphi_z(t)=\exp\left\{\sum_{k=0}^{\infty}\frac{i^k c_k}{k!}t^k\sum_{j=0}^{n}c_{k,j}\right\}.$$

Indo até segunda ordem (onde $c_1=\mu$ e $c_2=\sigma^2$) teremos:

$$\varphi_z(t)\cong\exp\left\{it\sum_{j=0}^{n}c_{1,j}-\frac{t^2}{2}\sum_{j=0}^{n}c_{c,j}\right\}=\exp\left\{it\sum_{j=0}^{n}\mu_j-\frac{t^2}{2}\sum_{j=0}^{n}\sigma_j^2\right\}$$

que é uma função característica de uma distribuição normal com $\mu=\sum_j\mu_j$ e $\sigma^2=\sum_j\sigma_j^2$.

Assim, ao somarmos várias variáveis aleatórias independentes teremos uma convergência à distribuição normal, isto é o que o teorema central do limite afirma.

2.4.12 Distribuições

Nesta seção iremos partir da definição da distribuição de Bernoulli e desenvolveremos até a distribuição log-Normal que é muito importante no estudo da equação de Black & Scholes.

2.4.12.1 Distribuição de Bernoulli

Para um evento binário, como jogar uma moeda, temos apenas duas possibilidades, neste caso, o evento "cara" que atribuímos com a função da v.a. com o valor $x=1$ e o "coroa" que será $x=0$. Podemos atribuir uma probabilidade p para o evento "cara" e uma probabilidade q para o evento "coroa" onde $q=1-p$. A CDF e a FDP serão representados na forma:

$F(x)=qH(x)+pH(x-1) \quad \rightarrow \quad f(x)=q\delta(x)+p\delta(x-1).$

Sua função geradora dos momentos é dada por

$$\mathbb{M}(t)=\int_{-\infty}^{+\infty}\left[q\delta(x)+p\delta(x-1)\right]e^{xt}dx=q+pe^{t}$$

e lembrando que $\varphi(t)=\mathbb{M}(it)$, a função característica tem a forma $\varphi_{ber}(t)=q+pe^{it}$. Podemos calcular um momento de ordem $k\neq 0$ qualquer:

$$M_k=\int_{-\infty}^{+\infty}\left[q\delta(x)+p\delta(x-1)\right]x^{k}dx=p$$

Temos então que $\mu=p$. Com μ os momentos centrados podem ser encontrados através da sua relação com a função geradora dos momentos:

$$m_k=\frac{d^k}{dt^k}\left[e^{-\mu t}M(t)\right]\Big|_{t=0}=\frac{d^k}{dt^k}\left[qe^{-pt}+pe^{qt}\right]\Big|_{t=0}=pq^{k}+(-1)^{k}qp^{k}$$

Em particular $\sigma^2=m_2=pq^2+qp^2=pq(q+p)=pq$.

2.4.12.2 Distribuição Binomial

Jogando uma moeda n vezes de forma a criar uma variável aleatória $z=x_1+x_2+\cdots+x_n$ onde os x_k são independentes e identicamente distribuídos e sua função característica pode ser obtida realizando n convoluções da distribuição de Bernoulli:

$\varphi_{bin}(t)=\varphi_{ber}^{n}(t)=\left[q+pe^{it}\right]^{n}$ e a FDP pode ser encontrada através da transformada de Fourier inversa:

$$f(z)=\frac{1}{2\pi}\int_{-\infty}^{+\infty}\left[q+pe^{it}\right]^{n}e^{itz}dt.$$

Expandindo em binômio de Newton:

$$f(z)=\sum_{k=0}^{n}\binom{n}{k}q^{n-k}p^{k}\left[\frac{1}{2\pi}\int_{-\infty}^{+\infty}e^{i(k-z)t}dt\right]=\sum_{k=0}^{n}\binom{n}{k}q^{n-k}p^{k}\delta(z-k)$$

2.4.12.3 Convergência da Distribuição Binomial para a Normal

Tomaremos um n muito grande na distribuição binomial de forma que poderemos expandir a exponencial em $\varphi_{bin}(t)$ e em seguida aplicar o logaritmo. Sabemos que $pe^{it}\le p<1$ lembrando que $|e^{it}|=1$, então temos:

$$\ln\varphi_{bin}(t)=n\ln\varphi_{ber}(t)=n\ln\left[q+p+ipt-p\frac{t^2}{2}+\cdots\right]\cong n\ln\left[1+ipt-p\frac{t^2}{2}\right].$$

A expansão do logaritmo é

$\ln(1+x)=\sum_{k=1}^{\infty}\frac{(-1)^{k-1}}{k}x^{k}\cong x-\frac{x^2}{2}$ onde expandimos apenas até segunda ordem, fazendo o $x=ipt-pt^2/2$, expandimos o logaritmo em $\ln\varphi_{bin}(t)$ e teremos:

$$\ln\varphi_{bin}(t)=n\left[\left(ipt-p\frac{t^2}{2}\right)-\frac{1}{2}\left(ipt-p\frac{t^2}{2}\right)^2\right]\cong n\left[ipt-p\frac{t^2}{2}+p^2\frac{t^2}{2}\right]$$

mas podemos reorganizar esta equação na forma:

$$\ln\varphi_{bin}(t)\cong n\left[ipt-p(1-p)\frac{t^2}{2}\right]=inpt-npq\frac{t^2}{2}.$$

Comparando com a função geradora dos cumulantes da distribuição normal

$$\ln\varphi_{N}(t)=i\mu t-\sigma^2\frac{t^2}{2}$$

Percebemos que a distribuição binomial converge para a normal com $\mu=np$ e $\sigma^2=npq$.

2.4.12.4 Distribuição Log-Normal

Para uma FDP onde mudamos a variável x para y utilizando uma transformação $g(x)=y$, nossa nova FDP será um $f(y)$ e a CDF será dada por $F(y)$ onde

$$f(y)=\lim_{\delta y\to 0}\frac{F(y+\delta y)-F(y)}{\delta y}=\lim_{\delta y\to 0}\frac{P(y<y_{\nu}\le y+\delta y)}{\delta y}$$

daí teremos

$P(y<y_{\nu}\le y+dy)=f(y)dy.$ O conjunto de pontos em x que vão a $y<y_{\nu}\le y+dy$ é

simplesmente o domínio de $g(x) = y_\nu$. Vemos na figura 18 três regiões exemplificadas onde dx_1 e dx_3 são positivos e dx_2 é negativo pois vemos que a derivada nesta região é negativa.

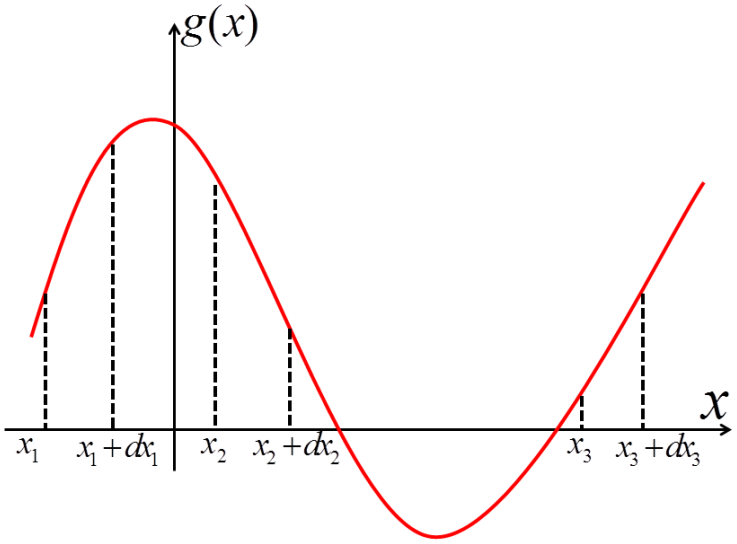


Figura 18 – Diferenciais positivos e negativos.

Onde a derivada é positiva (negativa), o *dx* também será positivo (negativo). Iremos escrever a probabilidade em y_ν separando as partes onde a derivada é positiva (denotando com o índice *i*) das negativas (índice *j*):

$$P(y < y_\nu \le y + dy) = \sum_i P(x_i < x_\nu \le x_i + dx_i) + \sum_j P(x_j + dx_j < x_\nu \le x_j) \text{ onde } dx_i > 0 \text{ e } dx_j < 0.$$

Com isso, temos:

$$P(y < y_\nu \le y + dy) = \sum_i f(x_i)dx_i - \sum_j f(x_j)dx_j = \sum_i f(x_i)\frac{dy}{g'(x_i)} - \sum_j f(x_j)\frac{dy}{g'(x_j)} =$$

$$= \sum_i f(x_i)\frac{dy}{g'(x_i)} + \sum_j f(x_j)\frac{dy}{[-g'(x_j)]} = \sum_k f(x_k)\frac{dy}{|g'(x_k)|}.$$

Assim, para realizarmos uma mudança de variável faremos [18]:

$$f(y) = \sum_k \frac{f[g^{-1}(y)]}{|g'[g^{-1}(y)]|}.$$

Para obter a distribuição log-normal mudamos a variável para $y = e^x$. Assim, $g(x) = e^x = g'(x)$ e a função inversa é $g^{-1}(y) = \ln y$ portanto $y \in [0, +\infty)$. Temos a distribuição log-normal dada por [23]:

$$logN_{\{\mu,\sigma^2\}}(y) = \frac{exp\left\{-\frac{(\ln y - \mu)^2}{2\sigma^2}\right\}}{\sqrt{2\pi}\sigma y}$$

Para obter a CDF bastaria integrar esta FDP no intervalo no qual o y pertence. Vemos nos gráficos abaixo o comportamento desta distribuição para alguns μ e σ:

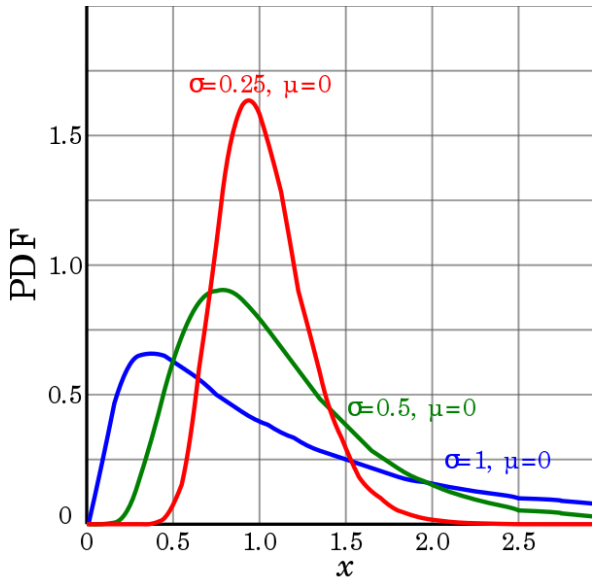


Figura 19 – Distribuições Log-Normal.

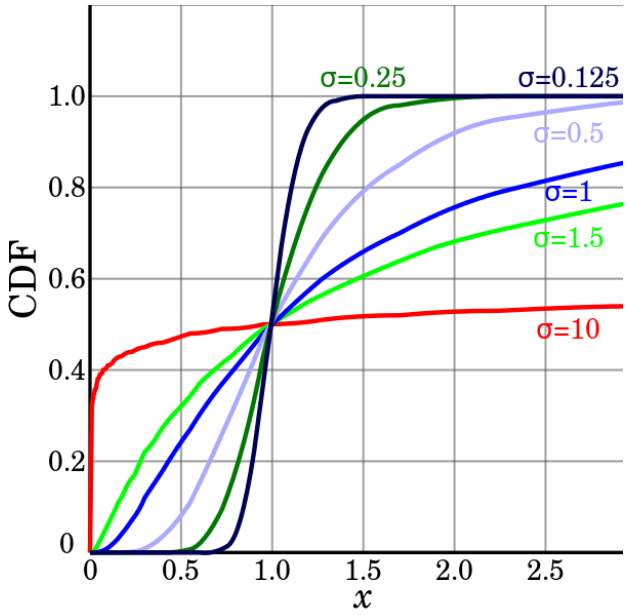


Figura 20 – Distribuições Cumulativas Log-Normal.

Tanto a função geradora dos momentos quanto a função característica da log-nomal apresentam problemas de convergência, mas podemos calcular os momentos da Log-Normal $\boldsymbol{M_n = \int_0^{\infty} y^n \frac{e^{-\frac{(\ln y - \mu)^2}{2\sigma^2}}}{\sqrt{2\pi}\,\sigma} \frac{dy}{y}}$ mudando a variável de integração para $\boldsymbol{\ln y = x}$, $\boldsymbol{y = e^x}$, $\boldsymbol{\frac{dy}{y} = dx}$,

quando $y \to 0 \quad x \to -\infty$ e quando $y \to +\infty \quad x \to +\infty$. Nesse caso:

$$M_n = \frac{1}{\sqrt{2\pi}\,\sigma} \int_{-\infty}^{\infty} e^{nx} e^{-\frac{(x-\mu)^2}{2\sigma^2}} dx .$$

Deve-se completar quadrado no expoente:

$$e^{nx} e^{-\frac{(x-\mu)^2}{2\sigma^2}} = e^{-\frac{(x-\mu)^2 - 2\sigma^2 nx}{2\sigma^2}} = e^{-\frac{x^2 - 2\mu x + \mu^2 - 2\sigma^2 nx}{2\sigma^2}} = e^{-\frac{x^2 - 2(\mu + n\sigma^2)x + (\mu + n\sigma^2)^2 - (\mu + n\sigma^2)^2 + \mu^2}{2\sigma^2}}$$ continuando

$$e^{nx} e^{-\frac{(x-\mu)^2}{2\sigma^2}} = e^{-\frac{(x-\mu x - n\sigma^2)^2 - (\mu^2 + 2\mu n\sigma^2 + n^2\sigma^4) + \mu^2}{2\sigma^2}} = e^{\mu n + n^2 \frac{\sigma^2}{2}} e^{-\frac{(x-\mu x - n\sigma^2)^2}{2\sigma^2}} .$$

Daí vemos que $M_n = e^{\mu n + n^2 \frac{\sigma^2}{2}} \left[\frac{1}{\sqrt{2\pi}\,\sigma} \int_{-\infty}^{\infty} e^{-\frac{(x-\mu x - n\sigma^2)^2}{2\sigma^2}} dx \right]$.

A integral entre colchetes vale 1 e temos todos os momentos de ordem n dados por $M_n = e^{\mu n + n^2 \frac{\sigma^2}{2}}$. Em particular temos $M_0 = 1;\ M_1 = e^{\mu + \frac{\sigma^2}{2}}$; $M_2 = e^{2\mu + 2\sigma^2}$. Podemos calcular os momentos centrados usando binômio de Newton $m_n = \sum_{k=0}^{n} \binom{n}{k} (-1)^{n-k} \mu^{n-k} M_k$. Já sabemos que $m_o = 1$ e $m_1 = 0$. Para a variância temos: $m_2 = M_2 - M_1^2$ logo $m_2 = e^{2\mu + 2\sigma^2} - e^{2\mu + \sigma^2} = e^{2\mu + \sigma^2}(e^{\sigma^2} - 1)$, $V[y] = e^{2\mu + \sigma^2}(e^{\sigma^2} - 1)$ e $\sqrt{V[y]} = e^{\mu + \frac{\sigma^2}{2}} \sqrt{(e^{\sigma^2} - 1)}$.

3 ANÁLISE DE ÓLEOS CRUS

Neste capítulo mostramos a aplicação dos conceitos do capítulo 2 na análise de espectros de fluorescência de óleos. Desde o início as companhias petrolíferas nos solicitaram uma metodologia capaz de diferenciar óleos entre si. A ideia de usar covariância, coeficiente de correlação e distância de correlação para extrair ***Minimum Spanning Tree*** (MST) de similaridades através de medidas experimentais ou outras observações está atualmente em destaque na comunidade de econofísica, com aplicações desde o mercado financeiro, análise de risco, até a classificação de materiais por proximidade de suas propriedades. Junto com a tradicional metodologia de análise de componentes principais permite uma forma automática de tratamento e visualização de um conjunto enorme de dados espectrais.

Fluorescência é o sinal óptico mais intenso com comprimento de onda diferente do feixe de luz incidente. O objetivo do grupo experimental foi, portanto, utilizar medidas de fluorescência para discriminar óleos crus de diferentes origens. Nosso objetivo principal, portanto, foi desenvolver uma metodologia de classificação de óleos através da fluorescência. Nesse aspecto, criar uma medida de distância entre os óleos e classificá-los através da MST é a forma mais natural de diferenciação das amostras de óleo. Entretanto existem desafios computacionais para esse objetivo devido à grande quantidade de dados gerados que necessitam de uma análise automática. As análises por fluorescência podem ser realizadas em apenas um ponto, no caso de amostras homogêneas, como um volume de óleo líquido, ou em até 1000×1000 pontos, no caso de imagens de fluorescência por microscopia confocal em rochas. Já nas análises de PLE [Photoluminescence Excitation Spectroscopy] por dois fótons [2p-PLE] se varre o laser de excitação entre 700 a 1000 nm em passos de 5 nm [1,25]. São, portanto, 60 espectros em cada uma das 4 componentes [amostra é fracionada em 4 partes usando diferentes solventes] de amostra de óleo. Nas imagens obtidas por fluorescência com 1000×1000 esse número é multiplicado por 10^6. Ou seja, 240 bilhões de pontos/amostra – em 100 amostras, teremos 24 trilhões de pontos. Além disso, o projeto de Física do petróleo em meios porosos requer uma automatização da classificação de um grande número de mapas Raman de minerais que vêm sempre contaminada com a fluorescência dos óleos. No capítulo 4 lidaremos com os espectros Raman, inclusive com uma metodologia para extrair a fluorescência dos espectros obtidos. Entretanto, também é importante caracterizar as fluorescências que aparecem nas rochas.

Vale salientar que a minha contribuição nesse projeto foi na análise dos dados e não nas medidas experimentais em si, realizadas por colegas do grupo. Incluímos uma breve

descrição da medida em si por uma questão de completeza do capítulo.

3.1 Fluorescência de óleos crus

Decidimos utilizar a fluorescência para a caracterização dos óleos crus porque é o maior entre os sinais ópticos com comprimento de onda diferente do comprimento de onda de excitação, chegando à sensibilidade de detecção de uma única molécula. Trata-se, portanto, de uma medida óptica muito rápida e eficiente. Além disso, trata-se de uma medida que pode ser realizada remotamente, através de fibras ópticas, inclusive no interior de poços de petróleo, pois as fibras suportam as altas temperaturas e ambiente químico corrosivo no interior dos mesmos. Tipicamente, detectamos fluorescência na região de comprimento de onda entre 400 nm a 1000 nm. Nem toda molécula é fluorescente nessa região. Os níveis de energia das ligações atômicas são muito altos e a fluorescência cai na região ultravioleta e raios-x. Entretanto, nas moléculas orgânicas que apresentam conjugações, ligações duplas seguidas de uma simples, o elétron π pode se movimentar livremente, como no caso de uma partícula em uma caixa. Quanto maior a cadeia de conjugações mais para o vermelho/infravermelho se desloca a fluorescência. Petróleo possui um número muito grande de componentes, entre as quais muitas são fluorescentes. Baseados nessas moléculas podemos diferenciar as diversas amostras de óleo de uma forma rápida e eficiente.

O processo de fluorescência envolve três etapas: a primeira é a excitação, na qual o elétron absorve um fóton e muda do estado fundamental (de menor energia) para um estado excitado. A segunda é a dinâmica desse elétron no estado excitado, tipicamente decaindo para o estado excitado de menor energia onde permanece por algum tempo. O terceiro é o retorno para o estado fundamental emitindo um fóton, o fóton de fluorescência que detectamos. Assim o processo completo envolve a excitação, através de fótons que estejam dentro da banda de excitação, seguido por uma dinâmica interna do elétron na molécula, e a emissão em comprimentos de onda maiores, que formam a banda de emissão. A dinâmica do decaimento do elétron para o estado de energia mais baixo do nível excitado é muito rápida, da ordem de 1 ps, pois envolve apenas vibrações moleculares, ou emissão de fônons. Nesse nível mais baixo pode sobreviver por ns, tipicamente entre 500 ps até 3-4 ns, para a maioria das moléculas orgânicas, antes de decair por emissão de fótons. Dessa forma, podemos caracterizar a fluorescência de uma molécula através das medidas das bandas de excitação e emissão e também através da medida do tempo de vida de fluorescência, ou seja, em quanto tempo após a excitação o elétron decai emitindo um fóton.

Até a década de 1990 a excitação foi tipicamente realizada com a absorção direta de um fóton. Entretanto, percebeu-se que essa excitação poderia ser feita através da absorção de 2 fótons, desde que o dobro da energia dos mesmos (metade do comprimento de onda) caia dentro da banda de excitação, e que os 2 fótons cheguem juntos, superpostos no tempo. Usando lasers pulsados de femtossegundos se garante a superposição temporal dos fótons e o efeito de excitação via absorção de dois fótons [Two Photon Excited Fluorescence – TPEF] se torna tão eficiente quanto a excitação via 1 fóton. Vários pesquisadores já mostraram que a fluorescência poderia ser utilizada na classificação de óleos crus, entretanto sempre restringiram suas medidas a um conjunto pequeno de comprimentos de onda de excitação, e observando as bandas de emissão. Isso porque utilizam lasers contínuos para excitar a fluorescência.

Nossa proposta nesse projeto foi mais abrangente. Para não perder a fluorescência de qualquer molécula excitada desde 350 nm até 500 nm decidimos medir fluorescência excitada por dois fótons [TPEF] usando um laser de Ti:safira de femtossegundos sintonizável de 700 nm até 1000 nm. Como se tata de um laser pulsado com pulsos se repetindo a cada 12,5 ns, fica fácil também fazer a medida de tempo de vida, através da técnica de FLIM [Fluorescence Lifetime Imaging Microscopy]. Com essa medida caracterizamos os 3 aspectos da fluorescência – banda de excitação, tempo de vida e banda de emissão. Usando apenas excitações fixas não excitamos a fluorescência de outras moléculas cuja banda de excitação não contém o comprimento de onda utilizado.

3.1.1 Componentes do óleo cru

As amostras dos 4 tipos de óleo foram recebidas dos seguintes poços: Pré-sal Poço 3-RJS-646, Amostra de óleo do Pós-Sal da Bacia de Santos 3-RJS-622, mais duas amostras de óleo uma do campo de São João e outra do Contrato 80340. Anteriormente tínhamos recebido 15 amostras de óleo da Bacia Potiguar.

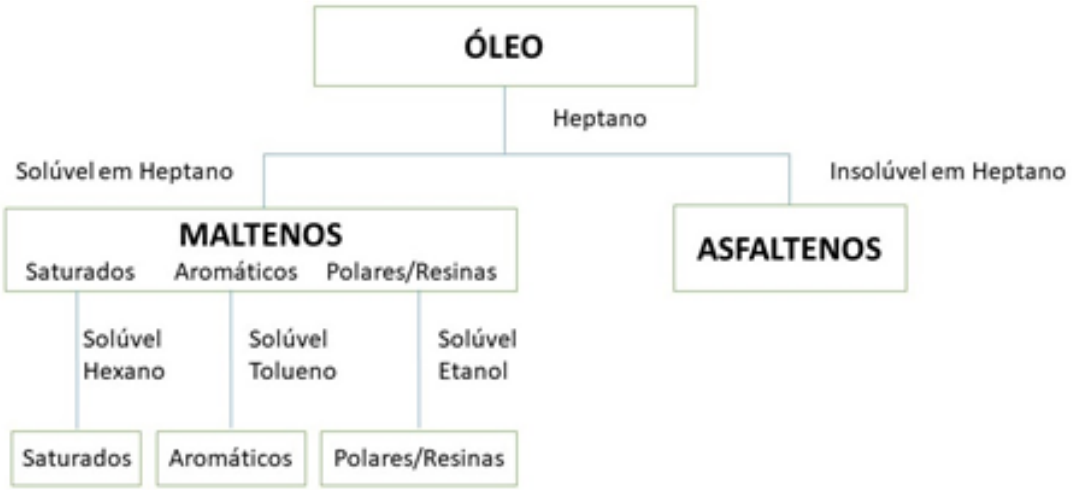


Figura 21 – Esquema de fracionamento do óleo cru em asfaltenos, aromáticos e polares.

A figura 21 mostra o esquema do fracionamento das amostras de óleo cru utilizados parra separar as componentes em asfaltenos e maltenos, e os maltenos em aromáticos e polares. Os saturados não interessam porque tipicamente, não apesentam fluorescência.

Figura 22 mostra os mapas "térmicos" (quanto maior sinal mais vermelho é a cor), no fundo um mapa de curvas de níveis, das 3 componentes de uma amostra de óleo cru obtida da fazenda Belém. No eixo vertical está o comprimento de onda do laser de femtossegundos e no eixo horizontal o comprimento de onda da fluorescência emitida. Vale notar que 800 nm em um processo de dois fótons equivale à 400 nm no processo de 1 fóton. A linha inclinada à esquerda é sinal de SHG da uréia utilizado na normalização dos espectros.

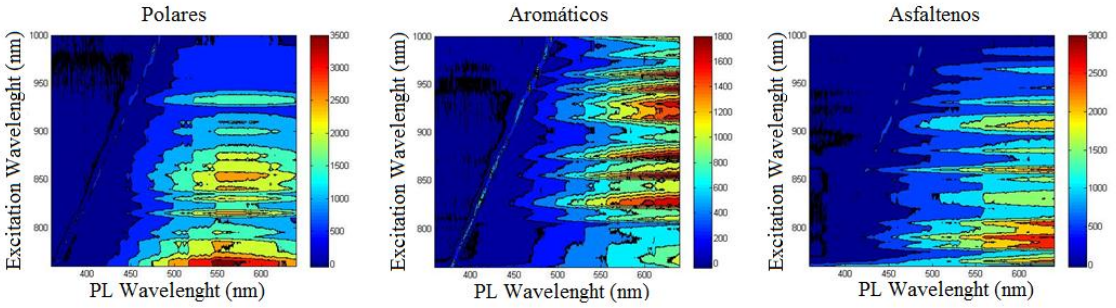


Figura 22 – Mapa de espectroscopia fotoluminescência de 2 fótons com a intensidade de fotoluminescência (cor) variando com a excitação e emissão dos componentes (a) polares; (b) aromáticos e (c) asfaltenos do petróleo obtido da Fazenda Belém.

Vale notar a presença de vários elipsoides nesses mapas que correspondem a moléculas

diferentes, eventualmente uma combinação de moléculas. Esses mapas mostram claramente o quanto a espectroscopia de excitação por dois fótons é capaz de discriminar amostras diferentes via uma parcela de seus componentes. Óleo cru contém, portanto, uma grande porção de moléculas fluorescentes, que atuam como uma "impressão digital", do mesmo. Essa medida também demonstra a capacidade muito maior de discriminação da nossa metodologia comparada com as medidas de fluorescência com apenas poucas linhas de laser de excitação. Também vale salientar que o processo foi todo executado com lasers, o que significa que pode ser utilizado com fibras ópticas a km de distância da amostra, ao contrário de experimentos com lâmpadas.

3.1.2 Calibração das medidas

Existe uma diferença grande nos processos de calibração da espectroscopia de excitação por um fóton e por dois fótons. No processo de 1 fóton o sinal obtido depende apenas da potência do feixe de excitação, que, obviamente, varia com o comprimento de onda. Dessa forma, dividir uma pequena fração do feixe para um detector de potência permite normalizar os espectros de todas as emissões dividindo o sinal pela potência do feixe de excitação. Já no caso da excitação por dois fótons, o sinal depende do quadrado da intensidade do feixe, ou seja, do quadrado da potência dividida pela área e pela duração do pulso do laser. Todos esses parâmetros, potência, largura temporal e espacial do feixe, mudam quando sintonizamos o laser para outro comprimento de onda, dificultando a normalização. A forma como procedemos na normalização foi medir outro sinal que também depende do quadrado da intensidade do feixe incidente. Na Geração de Segundo Harmônico [SHG – Second Harmonic Generation] a amostra gera um sinal com o dobro da frequência, metade do comprimento de onda, do feixe incidente. Esse sinal também depende da intensidade quadrática do feixe incidente e pode ser utilizado na normalização dos espectros. Uréia em pó é um excelente gerador de SHG e desenvolvemos a calibração, em um primeiro momento, misturando a uréia com a amostra, e posteriormente, fazendo medidas separadas da amostra e do SHG da uréia.
A dificuldade que encontramos no procedimento da normalização veio do fato de que o SHG de 700 a 800 nm cai na região de 350 a 400 nm, na qual a óptica do nosso sistema transmite pouca luz. Com isso os sinais de SHG nessa região foram bem menores do que deveriam ser e na divisão da normalização essa região se tornou muito mais intensa do que as outras, como se pode ver na figura 23. Uma opção seria considerar apenas os sinais acima de 800 nm, mas perdendo informação muito útil da excitação no ultravioleta, justamente a que tende a excitar

o maior número de moléculas [25].

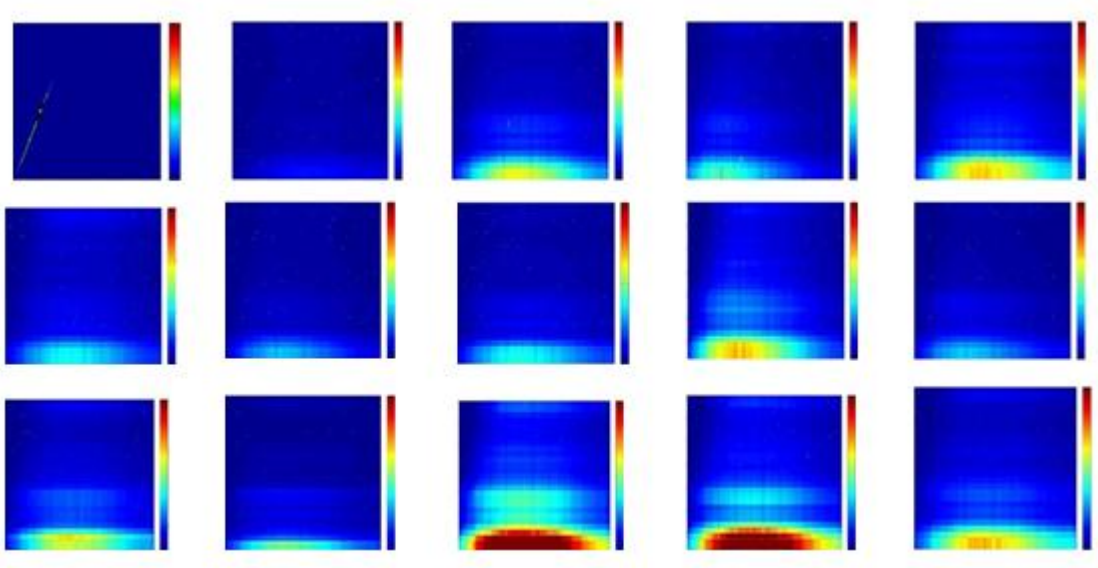


Figura 23. Mapa de espectros de excitação normalizados pelo sinal de segundo harmônico, com uma intensidade desproporcional nos comprimentos de ondas menores da excitação.

3.2 Correlação para um comprimento de onda de excitação

Para um mesmo comprimento de onda de excitação a correlação entre amostras diferentes obtida pelos espectros de emissão será imune a transformações lineares. Mesmo que houvesse uma variação da potência do laser entre uma medida e a outra a correlação seria a mesma. O espectro de emissão é obtido com o mesmo equipamento e varre todos os comprimentos de onda. Alta correlação pode significar que ambas as amostras possuem a mesma molécula. Figura 24 mostra a ideia da correlação da emissão para um mesmo comprimento de onda de excitação. Selecionando um comprimento de onda de excitação específico (destacado em vermelho na Figura 24) de cada componente do óleo obtém-se um espectro de emissão diferente para cada amostra:

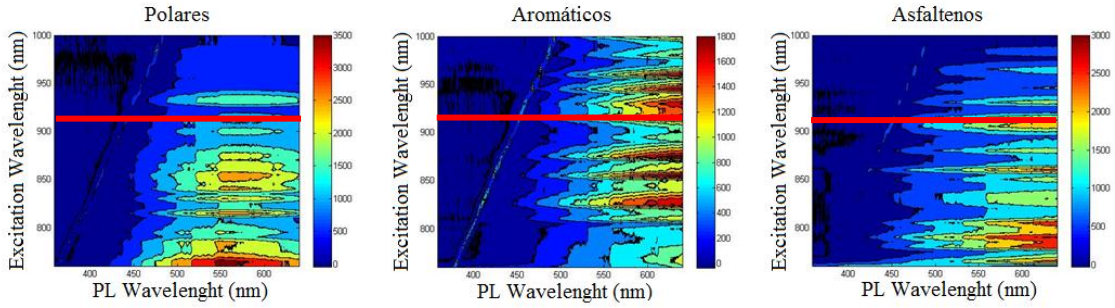


Figura 24 –Destacando um comprimento de onda de excitação.

Após a seleção dos espectros basta calcular o coeficiente de correlação r_{ij} entre estes. A partir do coeficiente de correlação obtemos a distância de correlação $d_{ij}^2 = 2\left(1 - r_{ij}\right)$. O problema com esse método é que não podemos misturar as linhas de diferentes comprimentos de onda. Existe uma correlação 2D e até 3D, mas vale lembrar que a correlação possui termos como $\left(x_i - \bar{x}\right)\left(y_i - \bar{y}\right)$ onde $\bar{x} \quad e \quad \bar{y}$ são as médias de $x \quad e \quad y$. Se misturarmos duas linhas em que a intensidade do laser mudou os valores as médias serão definidas pelas linhas com maior intensidade e a correlação 2D perde a informação que existia dentro de cada linha. Assim, podemos fazer a correlação linha por linha, mas não podemos fazer a correlação área por área, porque as intensidades das excitações mudaram na horizontal. A utilização da correlação 2D no nosso caso, portanto, só poderia ser utilizada após uma normalização muito cuidados e trabalhosa, que deveria ser feita simultaneamente com as medidas de espectroscopia de excitação. A seguir apresentamos um método que computa a correlação pareada, linha por linha e em seguida combinando os resultados.

3.3 Combinando experimentos

Aqui utilizamos a ideia de experimentos combinados para resolver o problema fundamental da calibração dos espectros de excitação por dois fótons. Podemos comparar a emissão de amostras diferentes desde que o comprimento de onda de excitação seja o mesmo. Ou seja, é possível fazer a correlação em cada linha horizontal de mesma excitação de amostras diferentes e essa correlação é invariante frente a uma transformação linear. Dessa forma, independente até de variações na intensidade do feixe de excitação. Queremos encontrar uma distância ente duas amostras de óleo, e já vimos que a distância de correlação segue os axiomas de uma distância, com a vantagem de ser imune a transformações lineares. Também sabemos que a distância Euclidiana, da forma $d^2 = d_1^2 + d_2^2 + \cdots + d_n^2$ também segue os axiomas de uma distância desde que cada d_i seja uma distância.

Dessa forma a ideia dos experimentos combinados é: (1) achar a correlação linha por linha com mesmo comprimento de onda de excitação; (2) calcular a distância entre duas amostras na mesma linha através $d_{ij}^2 = 2\left(1 - r_{ij}\right)$; (3) calcular a distância entre as mesmas amostras em todas as linhas e somar o quadrado das mesmas $d_{ij}^2 = \sum_{linhas} d_{ij\ por\ linha}^2$. Com esse procedimento a

distância entre duas amostras se torna independente do procedimento de normalização. Mesmo que exista variação na intensidade de cada linha, um fator multiplicativo, a correlação entre as mesmas não muda.

Vale a pena explicitar a metodologia: calcular a correlação em um subconjunto dos dados de forma pareada, ou seja, os subconjuntos compartilham um mesmo atributo ou parâmetro. Com isso calcular a distância de correlação entre esses subconjuntos. Calcular essa distância para todos os subconjuntos do conjunto total e somar os quadrados das distâncias dos subconjuntos. Com isso obtemos uma distância entre duas amostras imune a transformações lineares.

3.4 MST de fluorescência de óleos

Para testar a capacidade de discriminação/classificação dos óleos por fluorescência obtivemos 15 amostras de óleo cru da Bacia Potiguar. Para manter o estudo duplo cego só tivemos a informação de um número para cada amostra de óleo. Assim obtivemos os 15 mapas de espectroscopia de excitação das amostras apenas diluídas em tolueno da figura 25.

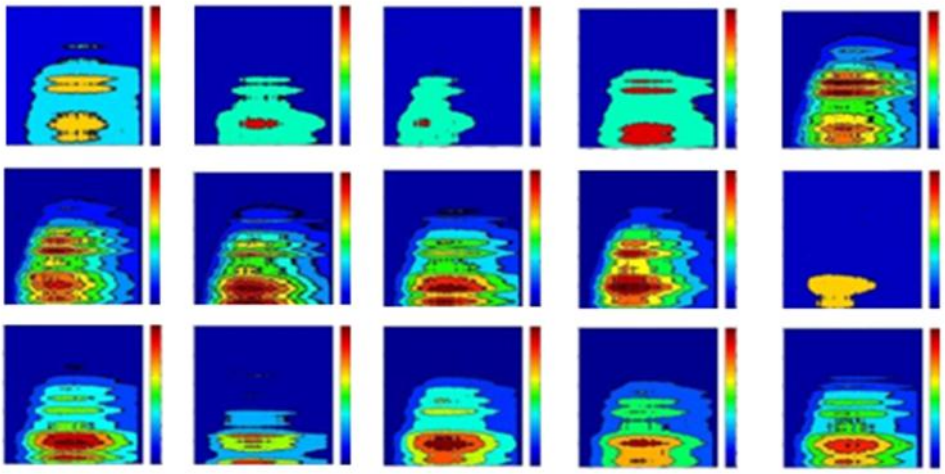


Figura 25 – Mapas das espectroscopias de excitação das 15 amostras de óleo cru obtidas junto à Petrobas.

Com esses mapas obtivemos as distâncias entre os óleos via a correlação pareada linha a linha. Com essas distâncias podemos construir a MST da distância entre os óleos usando o algoritmo de PRIM. Figura 26 mostra a matriz de distância entre os 15 óleos reorganizada

pela MST de PRIM na esquerda e a rede MST entre os óleos junto com sua numeração e mapa espectral de excitação de fluorescência com dois fótons.

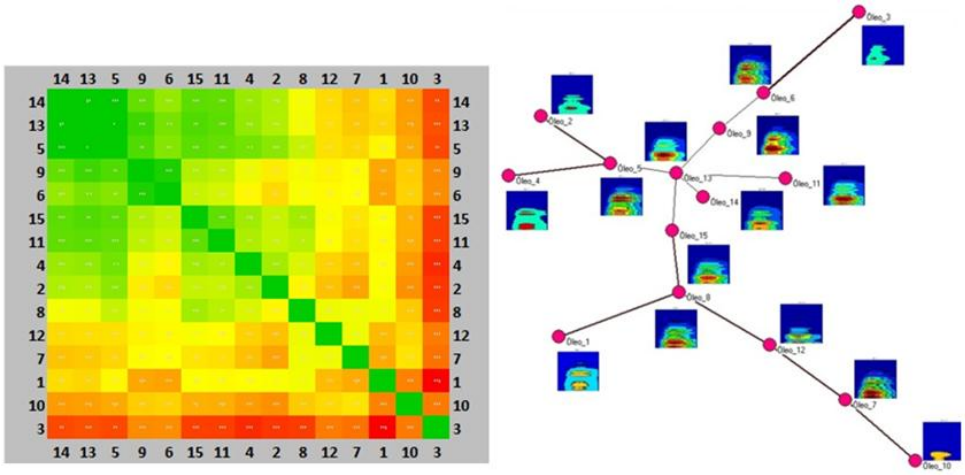


Figura 26 – Esquerda: matriz de distância de correlação pareada entre os 15 óleos reorganizada pela MST de PRIM. Direita: Rede MST dos 15 óleos obtida pela distância de correlação pareada.

Nesse ponto seria necessário saber a origem dos óleos de 1 a 15 e analisar como a semelhança encontrada entre os mesmos está relacionada com os aspectos geológicos dos diferentes poços. Ainda não conseguimos essa informação da Petrobras. A geração de um banco de dados catalogado nos dá a possibilidade de uma futura implementação de algoritmos de aprendizagem de máquina na classificação de novas amostras de óleo automaticamente.

4 ANÁLISE AUTOMÁTICA DE DADOS RAMAN

-Neste capítulo iremos utilizar técnicas como a regressão linear aplicadas na suavização e remoção de *background* de espectros de forma a tornar possível o cálculo de covariância entre os espectros, preservando apenas a informação dos picos. Outros métodos de pré-processamento são utilizados como a normalização das intensidades dos espectros que podem variar bastante para diferentes equipamentos de medida e diferentes potências do laser, temperatura e pressão. A correção do número de onda se dá via calibração do equipamento utilizando uma amostra cujo espectro é de fácil obtenção na literatura.

Com os dados tratados utilizamos a técnica não-supervisionada de análise de componentes principais (PCA) para encontrar o número ideal de componentes em um mapa de espectros, reduzindo um mapa de 100 espectros a 3 componentes principais que explicam 99% dos dados. A visualização das componentes se dá por meio da técnica de resolução de curvas multivariada (MCR) bastante utilizada em análises químicas e biológicas. O MCR possui restrições de não-negatividade nas componentes proporcionando padrões espectrais. O método é iterativo e visa encontrar uma matriz com as componentes espectrais e as suas concentrações, que são os pesos das combinações lineares das componentes para cada ponto.

A matriz de concentração nos proporciona uma imagem com cores distintas para cada componente no mapa. Agrupamos os espectros por componentes, mas estes podem ser mistos, portanto as componentes devem ser as mais puras possíveis. A seleção das componentes mais puras no mapa é dada pelo método SIMPLISMA (*Simple-to-use Interactive Self-modelling analysis*) [52]. O método é capaz de selecionar as componentes com maior taxa de pureza dada por $p_j = \sigma_j / \mu_j$ (no *j*-ésimo espectro). Quanto maior esta taxa, maior a probabilidade de seleção de um espectro não-misto. Os próximos selecionados devem ser descorrelacionados com os anteriores. Os espectros selecionados servirão de estimativa inicial no método iterativo MCR. Este irá minimizar o erro quadrático entre a matriz de espectros e o produto da matriz de concentração e componentes: $D = CS^T$ [48]. Esta última equação nos diz que um conjunto de espectros D com dimensão de $n^o\ de\ espectros \times n^o\ de\ pontos\ por\ espectro$ pode ser reescrito como o produto de uma matriz de concentração que dá o peso de cada componente principal para gerar os espectros em *D*. C possui dimensão de $n^o\ de\ espectros \times n^o\ de\ componentes$ e é multiplicado S^T com dimensão $n^o\ de\ componentes \times\ n^o\ de\ pontos\ por\ componente$.

A identificação das componentes espectrais em minerais é um problema de classificação e utiliza-se aprendizagem reforçada (*deep learning*). Os algoritmos de aprendizagem reforçada como as redes neurais são capazes de reconhecer padrões através de hipersuperfícies não-lineares de separação entre os pontos das amostras para cada classe. Através de um banco de dados catalogados de espectros treinamos CNN para reconhecer distintos minerais [73]. Este tipo de metodologia de análise pode ser aplicado em diferentes técnicas espectroscópicas para identificação automática de novas amostras.

4.1 Métodos de tratamento de dados

Neste capítulo apresentaremos alguns métodos de tratamentos dos dados que obtivemos. A limpeza dos dados é necessária antes de aplicar os métodos não-supervisionados de agrupamento para encontrar as componentes principais em um mapa Raman. As amostras adquiridas em diferentes instrumentos de medida poderão produzir diferentes picos experimentais nos espectros. Verifica-se variações nas posições do número de onda, intensidade e larguras []. As diferenças podem ter várias causas como a geometria da amostra, resolução do espectrômetro e intensidade do laser [36].

Os tratamentos consistem em (1) subtração do background de fluorescência [31-34], (2) suavização ou remoção de ruído [27-29], (3) normalização [35] e (4) calibração do número de onda [36]. O tratamento extrai toda a informação constituída pelos modos de vibração do material visualizada através dos picos Raman e esta informação é inserida no cálculo da matriz de covariância para comparação.

Na Figura 27 vemos as regiões de 2 mapas feitos em uma rocha:

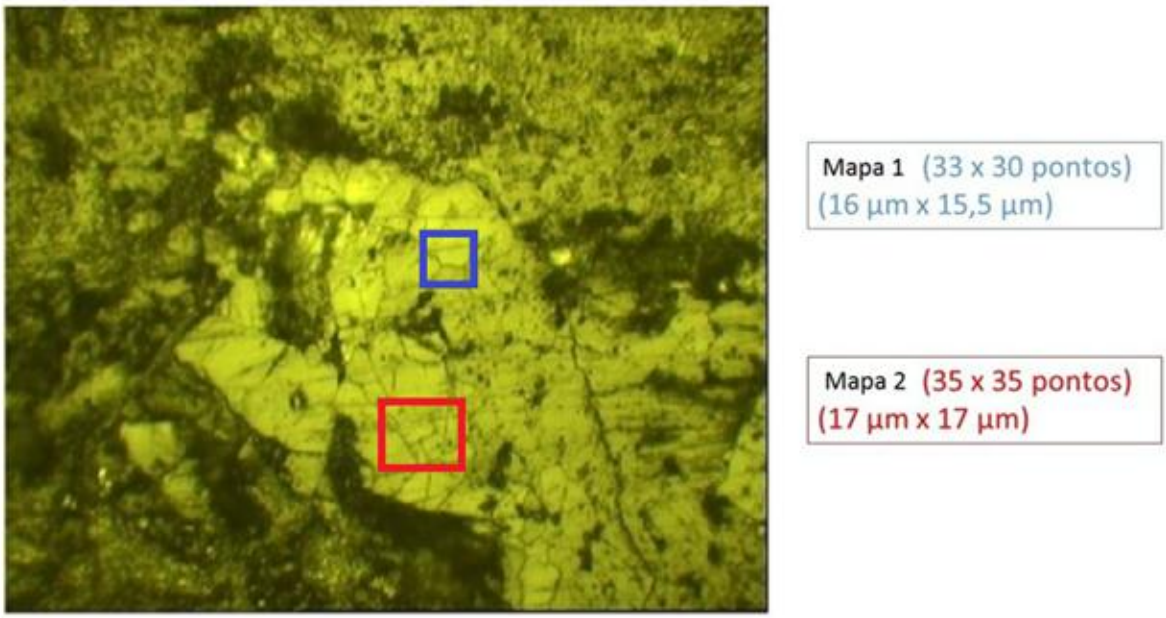


Figura 27 – Imagem óptica da lâmina de rocha mostrando as duas regiões utilizadas na aquisição de mapas de espectroscopia Raman.

A**Error! Reference source not found.Error! Reference source not found.** 28 mostra alguns espectros obtidos em diferentes posições da forma que sai do equipamento de medida. Vale notar o background largo da fluorescência sobreposto com picos finos devido ao Raman.

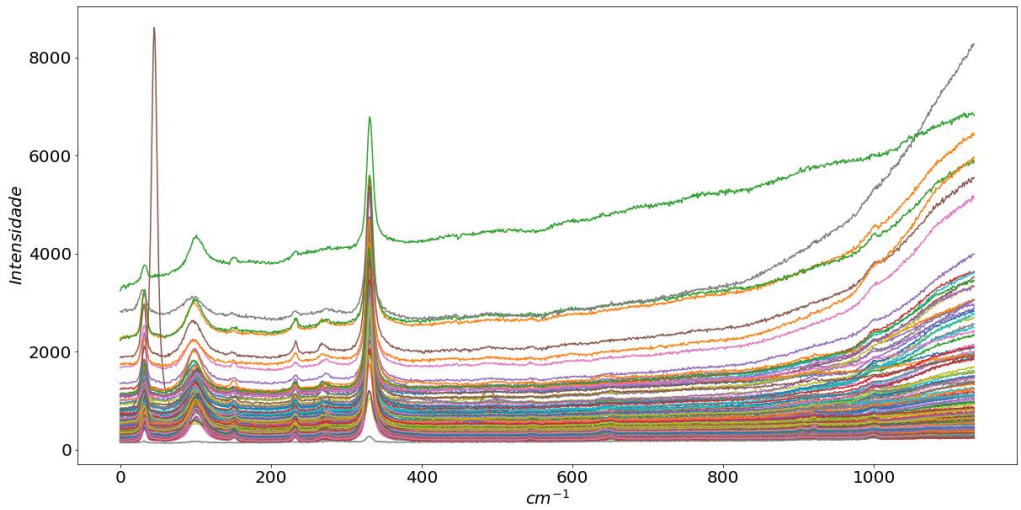


Figura 18 – Exemplos dos dados crus de espectroscopia Raman.

Também vale salientar que os backgrounds variam de ponto a ponto na amostra, devido a

diferentes contaminantes fluorescentes distribuídos na superfície da amostra [26]. A contribuição do background na matriz de covariância se sobrepõe aos picos Raman de modo que a covariância será composta principalmente por ruído.

O Primeiro passo deve ser eliminar o background para uma melhor comparação entre os espectros. Para a eliminação do background podemos usar diversos métodos. Outro tratamento é a suavização dos espectros através de filtros pois por vezes estes são bastante ruidosos. No nosso caso, utilizaremos uma regressão linear para suavizar os espectros e eliminar o background.

4.1.1 Filtro Savtzky-Golay

A suavização de curvas é tradicionalmente feita através de médias móveis ou médias móveis com decaimento exponencial, mas estes métodos descartam muita informação relevante. O filtro Savtzky-Golay (S-G) preserva bem a informação dos picos e reduz bastante o ruído.

O método é constituído por uma janela móvel com comprimento $2x_o + 1$ predefinido. Em um cada ponto com coordenada x, a janela terá um alcance de $x - x_o$ a $x + x_o$.

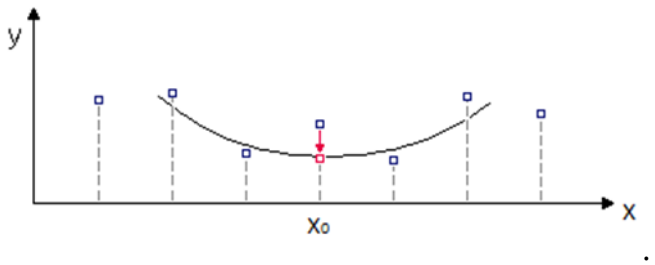


Figura 29 – Ilustração do fitting de um polinômio nos pontos selecionados.

Define-se a ordem de um polinômio (geralmente é utilizado com ordem 1 ou 2) que configurará a matriz X e em cada janela é efetuada uma regressão linear cuja amostragem são os pontos $y - y_o$ até $y + y_o$ estimando o valor de $\hat{a}$. Por último, salva-se apenas o ponto central y em um novo vetor [27-29].

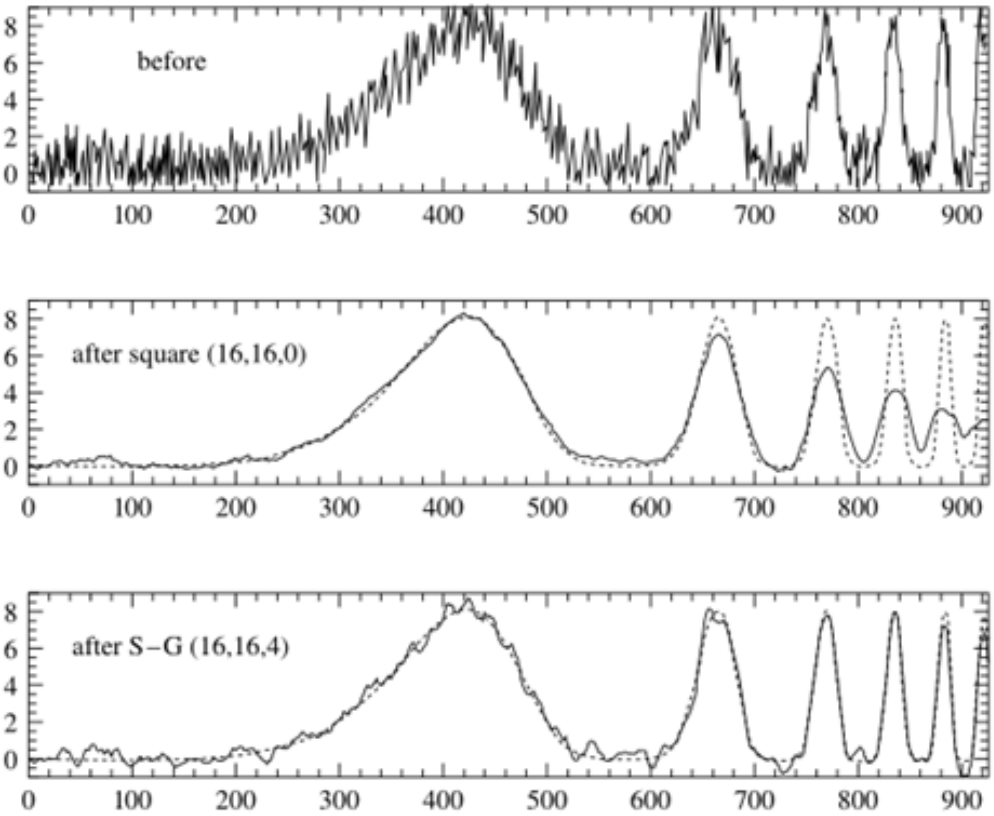


Figura 30 – No topo temos um espectro não tratado, no meio vemos a aplicação de um filtro de médias móveis com 16 pontos a esquerda e a direita na janela móvel, abaixo vemos o filtro S-G com 16 pontos a esquerda e a direita com um polinômio de grau 4 [27].

4.1.2 Subtração automática do background

Apresentamos aqui uma aplicação direta da regressão linear com mínimos quadrados ordinários e um segundo método iterativo bastante utilizado para remoção de background. Ambos os métodos visam remover a linha de base sem a seleção manual das regiões sem picos, automatizando o processo.

4.1.2.1 Regressão Linear dos pontos mínimos de terceira-ordem

Criamos uma janela móvel simétrica para cada valor do eixo *x* do espectro na qual associamos o valor do mínimo de *y* na janela ao ponto central. Fundamental que a janela seja simétrica, ou seja, os pontos devem variar entre $x - x_o$ até $x + x_o$ incluindo o ponto x. Através dos pontos mínimos selecionados, criamos uma nova janela móvel selecionando seus pontos mínimos que são chamados de mínimos de segunda ordem. Indo até terceira ordem

[42]:

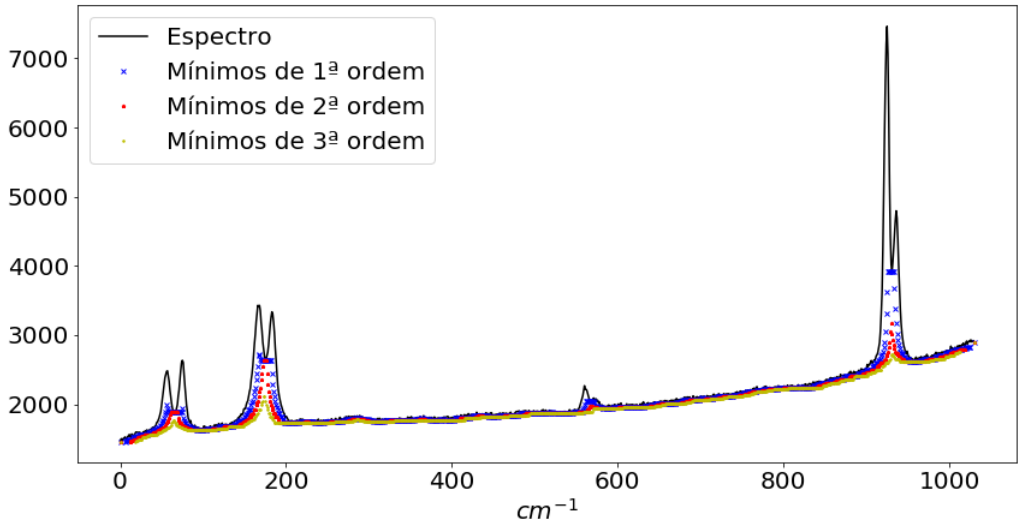


Figura 31 – Espectro Raman (azul) e o resultado obtido dos pontos mínimos coincidentes da janela móvel com um $x_0 = 6$ espectro (laranja).

Selecionamos apenas os pontos mínimos que coincidem com o espectro:

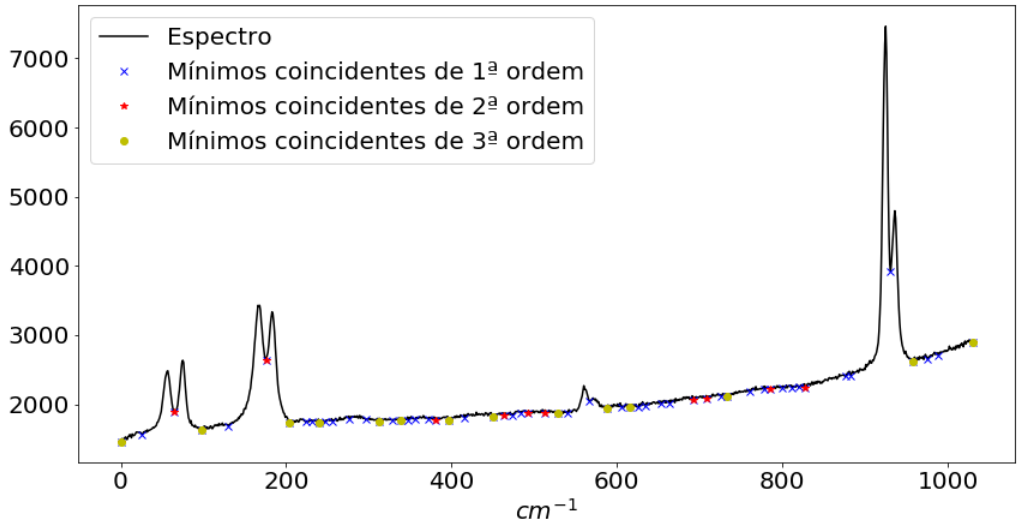


Figura 32 – Mínimos coincidentes com o espectro.

Vemos que os mínimos de coincidentes de Primeira e segunda ordem selecionam pontos fora da região de background. Pode-se aplicar uma interpolação dos pontos selecionados de

terceira ordem [34], mas aplicaremos uma regressão linear automática. Para isso usamos a forma matricial da regressão em que os coeficientes são dados por $\hat{\vec{a}} = (\bar{X}X)^{-1}\bar{X}\,\vec{y}$, onde $\hat{\vec{a}}_{k\times 1}$ é o vetor dos coeficientes lineares, a matriz $X_{n\times k}$ é dada pelo polinômio $X_{ij} = x_i^j$, onde $i \in [1,2,\cdots,n]$ com n sendo o número de pontos obtidos no espectro, $j \in [0,1,...,k]$ é a potência do polinômio, k é a ordem do polinômio, a matriz $\bar{X}_{k\times n}$ é a transposta da matriz $X_{n\times k}$ e o vetor coluna $\vec{y}_{n\times 1}$ é dado pelos pontos experimentais. Denotaremos esta operação como **regressão polinomial**.

Selecionamos um espectro de um mapa e aplicamos o método proposto, abaixo vemos o resultado da regressão utilizando polinômios de ordens 1, 3 e 6:

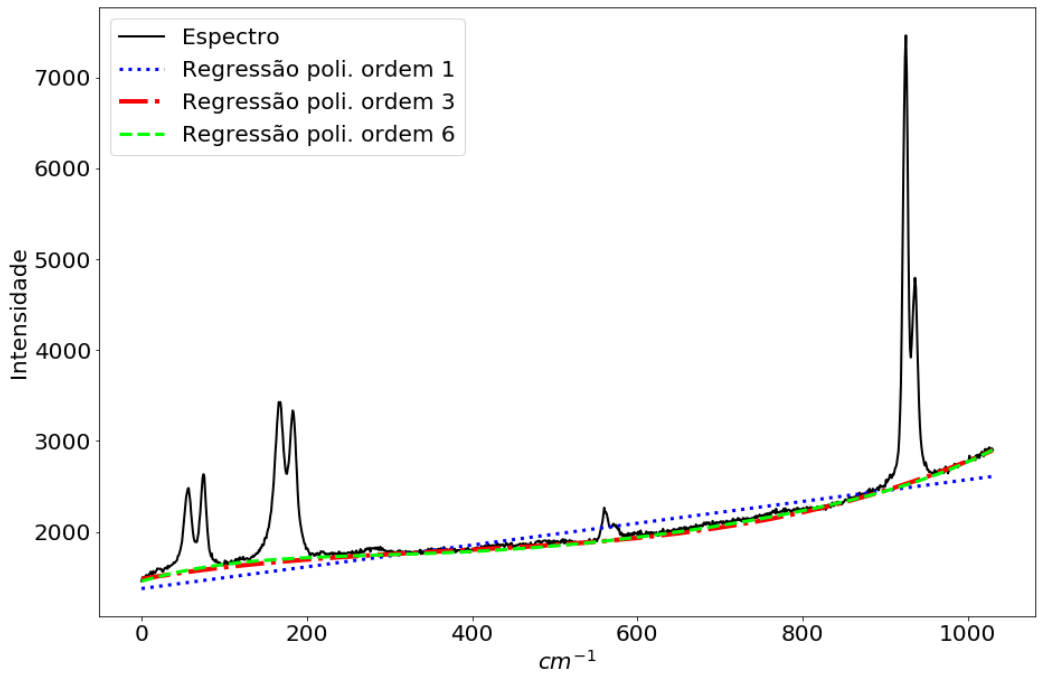


Figura 33 – Regressões polinomiais de ordens 1, 3, 6 dos pontos de mínimo.

Observamos que o polinômio de ordem 1 representa um *underfitting* pois é uma hipótese muito simples para o problema, os polinômios de ordem 3 a 6 obtiveram uma linha de base razoável.

4.1.2.2 Método arPLS

O ponto fundamental desse procedimento é a forma automática no qual é aplicado para o processamento simultâneo de milhares de espectros obtidos em um mapa Raman. Claramente, pode ser também utilizado com qualquer tipo de espectroscopia realizada ponto a ponto. Existem vários outros métodos de remoção de background como o método transformação *wavelet* que também remove ruído. Entre outros temos o airPLS (*adaptive iteratively reweighted Penalized Least Squares*) [31] que é um método iterativo e arPLS (*asymmetrically reweighted penalized least squares*) que é baseado no airPLS [32]. Deseja-se encontrar a linha de base *z* para um espectro *y* minimizando a seguinte função de custo :

$$S(z) = \sum_i (y_i - z_i)^2 + \lambda \sum_i \left(\Delta^2 z_i\right)^2 .$$

O Primeiro termo em *S* é a soma do quadrado dos resíduos que desejamos minimizar. Mas ainda é preciso suavizar a curva e para isso usamos o segundo termo, no qual $\Delta^2 z_i \equiv \left(z_i - z_{i-1}\right) - \left(z_{i-1} - z_{i-2}\right) = z_i + 2z_{i-1} + z_{i-2}$, representando, portanto, a diferença das diferenças de duas regiões vizinhas. Caso as duas diferenças sejam grandes assume-se que esta região não é suave. A minimização desse termo garante, portanto, a suavidade da curva. O problema, então, é encontrar os z_i's que melhor se ajustem aos pontos experimentais de forma suave. O segundo termo deve penalizar estes comportamentos não suaves de *z* com parâmetro λ que balanceia os dois termos. Usualmente atribuímos um alto valor para λ $\left(10^2 \le \lambda \le 10^9\right)$ nos pontos que devem ser minimizados.

Podemos ilustrar o Primeiro e segundo termo de *S* na figura abaixo:

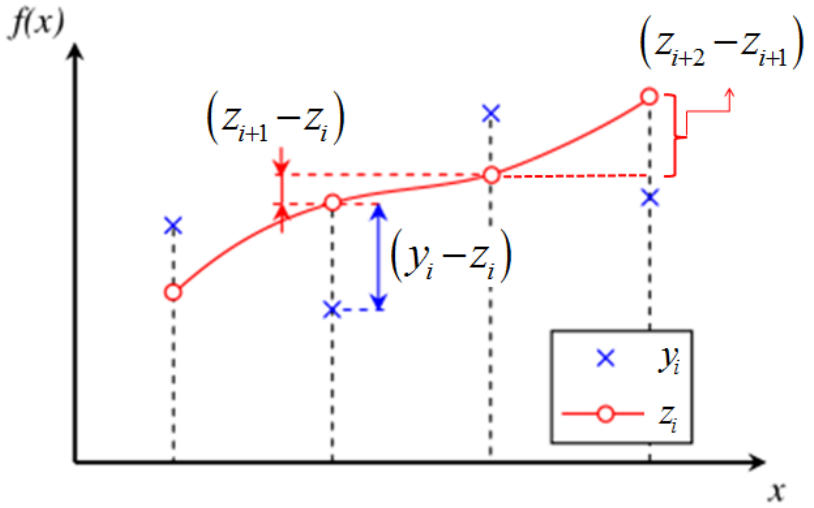


Figura 34 – Representação dos termos em *S*.

Para corrigir a linha de base um vetor peso *w* é introduzido através de uma matriz diagonal *W* com os pesos *w* em sua diagonal:

$$S(\vec{z}) = \sum_i w_i (y_i - z_i)^2 + \lambda \sum_i (z_i - 2z_{i-1} + z_{i-2})^2.$$

Onde

$$w_{n\times n} = \begin{bmatrix} w_1 & 0 & \cdots & 0 \\ 0 & w_2 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & w_n \end{bmatrix}.$$

A matriz $D = \Delta^2$ é chamada de matriz de diferença, aplicada em *z* temos:

$$D\vec{z} = D_{n\times n} z_{n\times 1} = \begin{bmatrix} 1 & -2 & 1 & 0 & \cdots & 0 & 0 & 0 \\ 0 & 1 & -2 & 1 & \cdots & 0 & 0 & 0 \\ 0 & 0 & 1 & -2 & \cdots & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & \cdots & 0 & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & 0 & \cdots & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & \cdots & -2 & 1 & 0 \\ 0 & 0 & 0 & 0 & \cdots & 1 & -2 & 1 \end{bmatrix} \begin{bmatrix} z_1 \\ z_2 \\ z_3 \\ z_4 \\ \vdots \\ z_{n-2} \\ z_{n-1} \\ z_n \end{bmatrix} = \begin{bmatrix} z_1 - 2z_2 + z_3 \\ z_2 - 2z_3 + z_4 \\ z_3 - 2z_4 + z_5 \\ z_4 - 2z_5 + z_6 \\ \vdots \\ z_{n-4} - 2z_{n-3} + z_{n-2} \\ z_{n-3} - 2z_{n-2} + z_{n-1} \\ z_{n-2} - 2z_{n-1} + z_n \end{bmatrix} = \overrightarrow{(z_i - 2z_{i-1} + z_{i-2})}.$$

Nossa equação pode, então, ser reescrita na forma:

$$S(\vec{z}) = (\vec{\bar{y}} - \vec{\bar{z}})w(\vec{y} - \vec{z}) + \lambda(\overrightarrow{\bar{D}\vec{z}})D\vec{z} = (\vec{\bar{y}} - \vec{\bar{z}})w(\vec{y} - \vec{z}) + \lambda\vec{\bar{z}}(\bar{D}D)\vec{z}.$$

Renomeando $\vec{y} - \vec{z} = \vec{x}$, a derivada do Primeiro termo em relação a *z* é:

$$\frac{\partial}{\partial x_k}\sum_i\sum_j x_i w_{ij} x_j = \sum_i\sum_j x_i w_{ij}\frac{\partial}{\partial x_k}x_j + \sum_i\sum_j \frac{\partial}{\partial x_k}x_i w_{ij} x_j = \sum_i\sum_j x_i w_{ij}\delta_{jk} + \sum_i\sum_j \delta_{ik} w_{ij} x_j =$$
$$= \sum_i x_i w_{ik} + \sum_j w_{kj} x_j = \sum_i x_i w_k \delta_{ik} + \sum_j w_k \delta_{kj} x_j = 2w_k x_k = 2w\vec{x}.$$

Derivando o segundo termo da expressão de *S*:

$$\frac{\partial}{\partial z_k}\lambda\sum_i\sum_j z_i(\bar{D}D)_{ij} z_j = \lambda\sum_i\sum_j z_i(\bar{D}D)_{ij}\frac{\partial z_j}{\partial z_k} + \lambda\sum_i\sum_j \frac{\partial z_i}{\partial z_k}(\bar{D}D)_{ij} z_j =$$
$$= \lambda\sum_i\sum_j z_i(\bar{D}D)_{ij}\delta_{jk} + \lambda\sum_i\sum_j \delta_{ik}(\bar{D}D)_{ij} z_j =$$
$$= \lambda\sum_i z_i(\bar{D}D)_{ik} + \lambda\sum_j(\bar{D}D)_{kj} z_j = \lambda\sum_i(\bar{D}D)_{ki} z_i + \lambda\sum_j(\bar{D}D)_{kj} z_j = 2\lambda\left[(\bar{D}D)\vec{z}\right]_k.$$

Na última linha trocamos os índices de $(\bar{D}D)$ pois toda matriz multiplica pela sua transposta gera uma matriz simétrica. Nosso objetivo é minimizar equação $\partial S / \partial z = 0$. Obtemos:

$$\overrightarrow{\left(\frac{\partial S}{\partial z}\right)} = -2w(\vec{y} - \vec{z}) + 2\lambda\bar{D}D\vec{z} = \vec{0}$$

$$w(\vec{y} - \vec{z}) = \lambda\bar{D}D\vec{z}$$

$$\begin{bmatrix} y_1 - z_1 & 0 & \cdots & 0 \\ 0 & y_2 - z_2 & \cdots & 0 \\ \vdots & \vdots & \ddots & 0 \\ 0 & 0 & \cdots & y_n - z_n \end{bmatrix}\begin{bmatrix} w_1 \\ w_2 \\ \vdots \\ w_n \end{bmatrix} = \lambda\bar{D}D\vec{z}.$$

Isolando o *z*:

$$(w + \lambda\bar{D}D)\vec{z} = w\vec{y}$$

$$\vec{z} = (w + \lambda\bar{D}D)^{-1} w\vec{y}$$

Definimos o termo $d^- \equiv y - z \mid y < z$, isto é, d^- é definida apenas na região $y < z$. Os novos pesos são atribuídos automaticamente através da seguinte função logística para um espectro *y*:

$$w_{new}(z, \mu_{d^-}, \sigma_{d^-}) = \left[1 + \exp\left\{2\left(\frac{y - z - (2\sigma_{d^-} - \mu_{d^-})}{\sigma_{d^-}}\right)\right\}\right]^{-1}$$

Onde μ_{d^-} e σ_{d^-} são a média e o desvio padrão do espectro d^-. Quando um ponto $y_i - z_i < 0$ a função logística tende ao peso 1 como vemos na e tende a 0 caso $y_i - z_i > 0$ assim os

pontos cuja linha de base estão acima do espectro recebem um peso *w* maior [65].

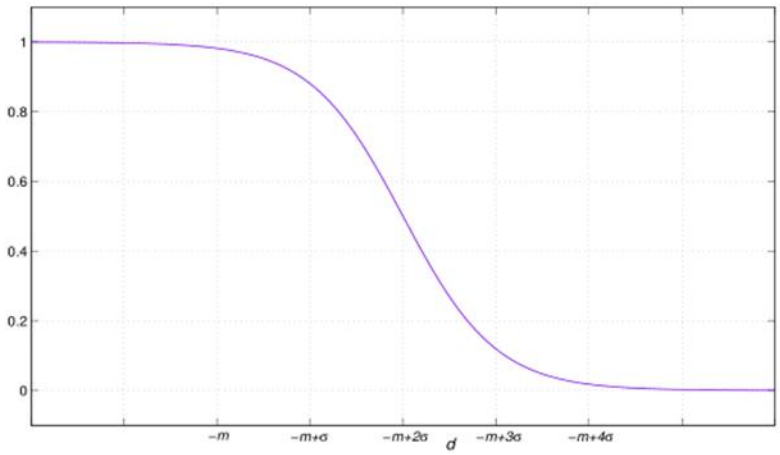


Figura e tende a 0 caso $y_i - z_i > 0$ assim os pontos cuja linha de base estão acima do espectro recebem um peso *w* maior [65].

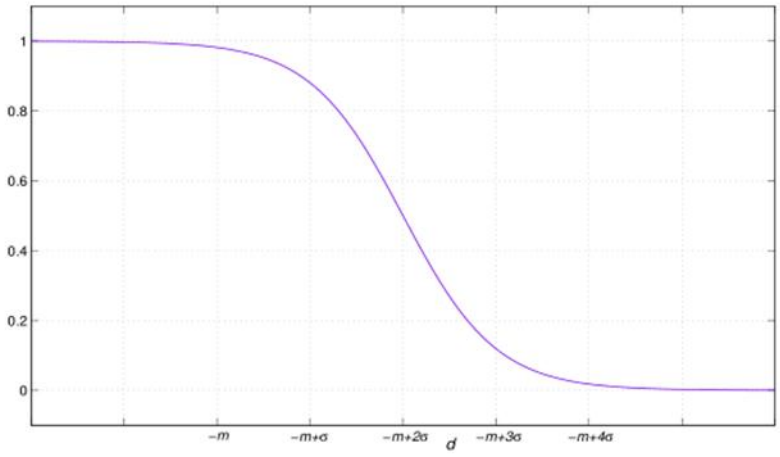


Figura 35 – Função logística do método proposto [31].

Com um novo peso w_{new} recalculamos S e em seguida calcula-se uma nova linha de base *z*. Repetimos o processo iteragindo até que atingir o critério: $\frac{\|w - w_{new}\|}{\|w\|} < 1\%$. A cada iteração atribui-se o valor do novo peso ao peso anterior $w := w_{new}$ portanto na expressão para o critério de convergência da linha de base, o w é sempre o peso da iteração anterior a w_{new} (o símbolo := significa atribuição numérica). Na figura 35 mostramos a convergência dos pesos para um espectro arbitrário cuja linha de base foi encontrada após 9 iterações (multiplicamos a intensidade do espectro por um fator de forma a conseguirmos visualizar os pesos que variam de 0 a 1 com o espectro):

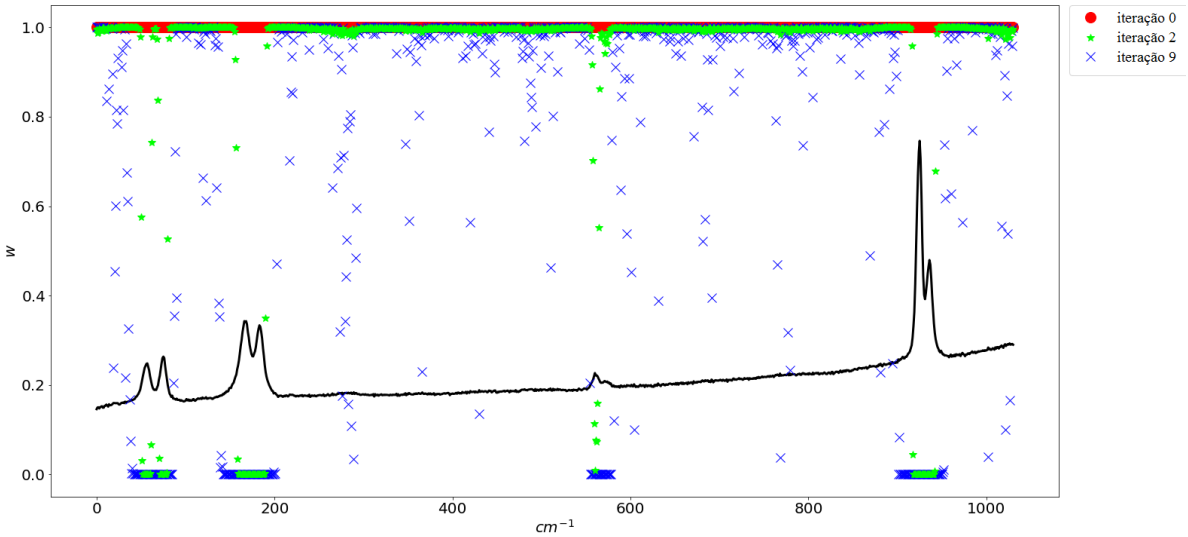


Figura 36 – Iteração 0, 2 e 9.

Na figura acima vemos que os pesos são inicialmente unitários, as regiões com picos recebem peso zero. Visualizamos também na Figura abaixo a convergência da linha de base utilizando um $\lambda = 10^5$:

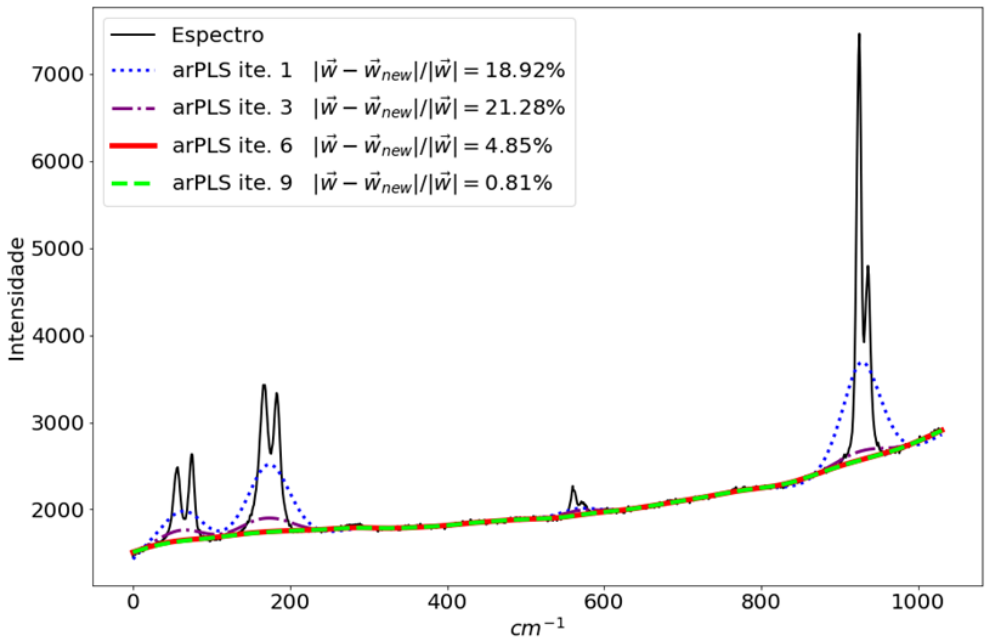


Figura 37 – Convergência da linha de base após 9 iterações com o método arPLS.

Comparando o método de regressão dos pontos de mínimo com o método arPLS mostramos a obtenção da linha de base dos espectros:

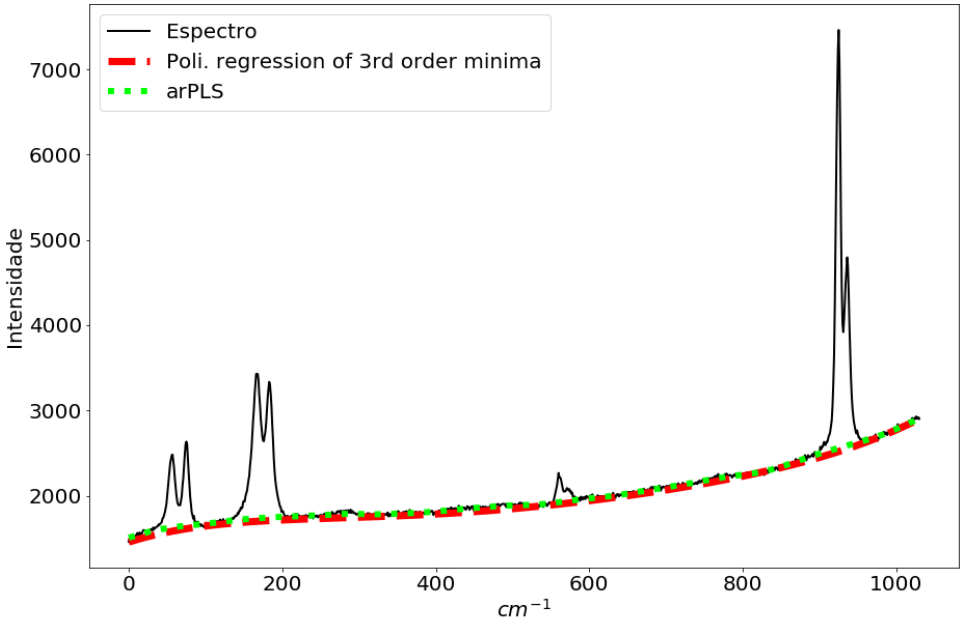


Figura 38 – Background com regressão polinomial de mínimos (esquerda) e arPLS (direita)

Vemos na Figura 38 que, para o espectro analisado, ambos os métodos reproduziram um bom ajuste da linha de base. Comparando a subtração de background de um grande número de espectros:

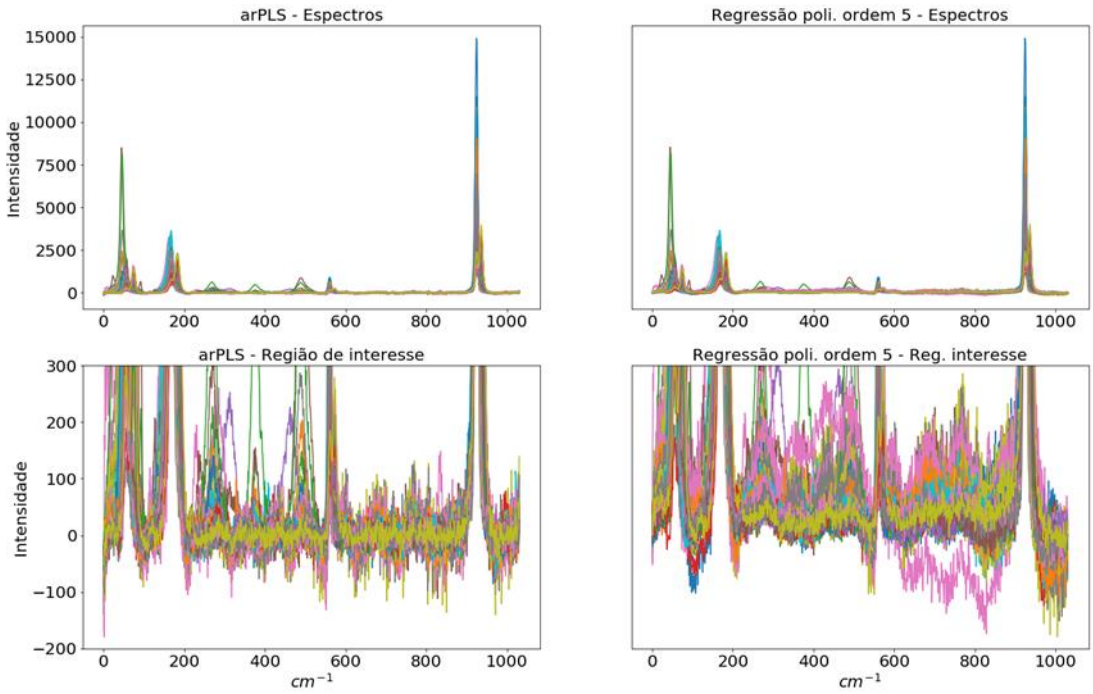


Figura 39 – Remoção de background com arPLS (esquerda) e regressão polinomial de ordem 5 para mínimos de 3ª ordem (direita).

A figura 39 nos mostra que o método arPLS reproduz um resultado superior ao método de subtração de background via pontos de mínimos locais reproduzindo uma linha de base mais precisa. Ambos os métodos são automáticos, o método de regressão dos pontos de mínimo é mais intuitivo e é executado em uma única série de operações, mas precisamos escolher a ordem do polinômio com cautela evitando underfitting e overfitting no restante da linha de base. Os métodos estão implementados no pacote Rampy [66] para Python.

4.1.3 Correção do número de onda e intensidade

Após a remoção da linha de base, redefinimos nossos espectros através de $\vec{y} := \vec{y} - \vec{z}$. Isto é, atribuímos a todos os espectros o seu valor subtraído da linha de base. Normalizamos cada um dos *i* espectros utilizando a fórmula:

$$y_i' = \frac{y_i}{\sqrt{\mu_i^2 + \sigma_i^2}}$$

Onde μ_i é a média e σ_i é o desvio padrão dos pontos do espectro y_i. Esta forma de normalização é proposta em [35] e utilizada na seleção de espectros puros para o método de visualização de componentes MCR. Como uma forma alternativa de normalização das intensidades dos espectros, divide-se cada espectro pela intensidade máxima de forma que o pico principal tenha intensidade unitária [40]. Com os espectros normalizados devemos calibrar o número de onda. Para realizar o ajuste basta utilizar um material com picos Raman bem definidos e conhecidos na literatura [,39]. No nosso caso, utilizamos o quartzo para realizar a calibração. Obtivemos 38 amostras tratadas da fonte [63] e as normalizamos como vemos na

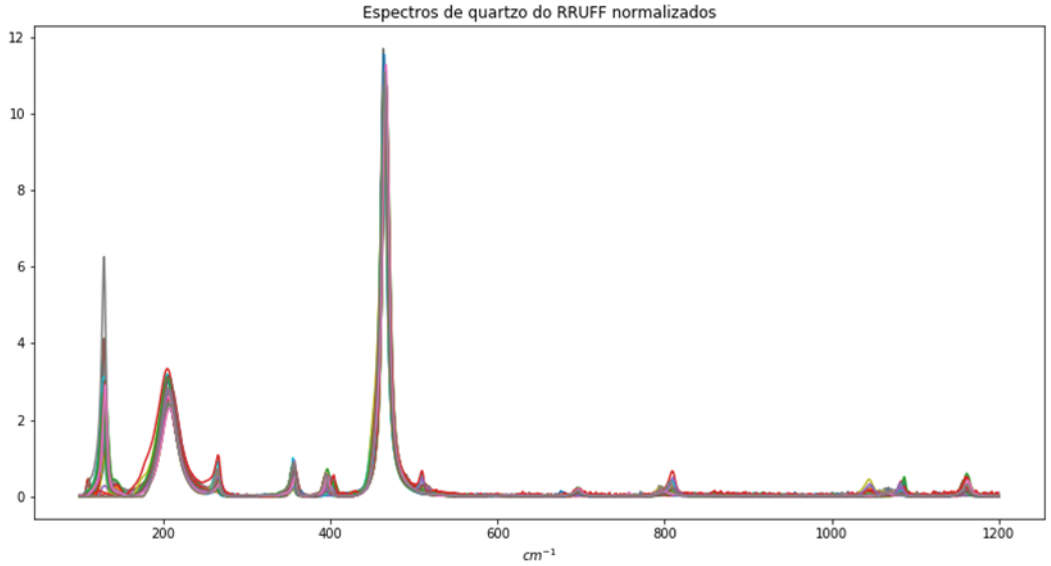


Figura :

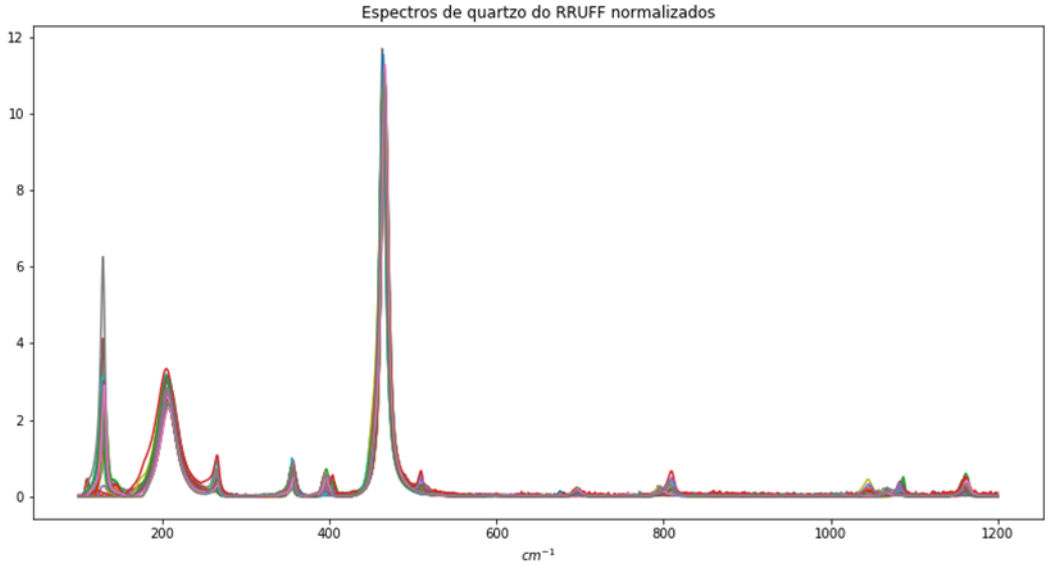


Figura 40 – 38 espectros de quartzo obtidos em RRUFF [63].

Geramos 10 espectros sintéticos novos y'_j combinando os 38 espectros de quartzo de forma que os pesos a_i sejam aleatórios sob condição que sua soma seja unitária.

$$y'_j = \sum_{i=1}^{38} a_i y_i \quad \rightarrow \quad \sum_{i=1}^{38} a_i = 1$$

Abaixo vemos os espectros gerados:

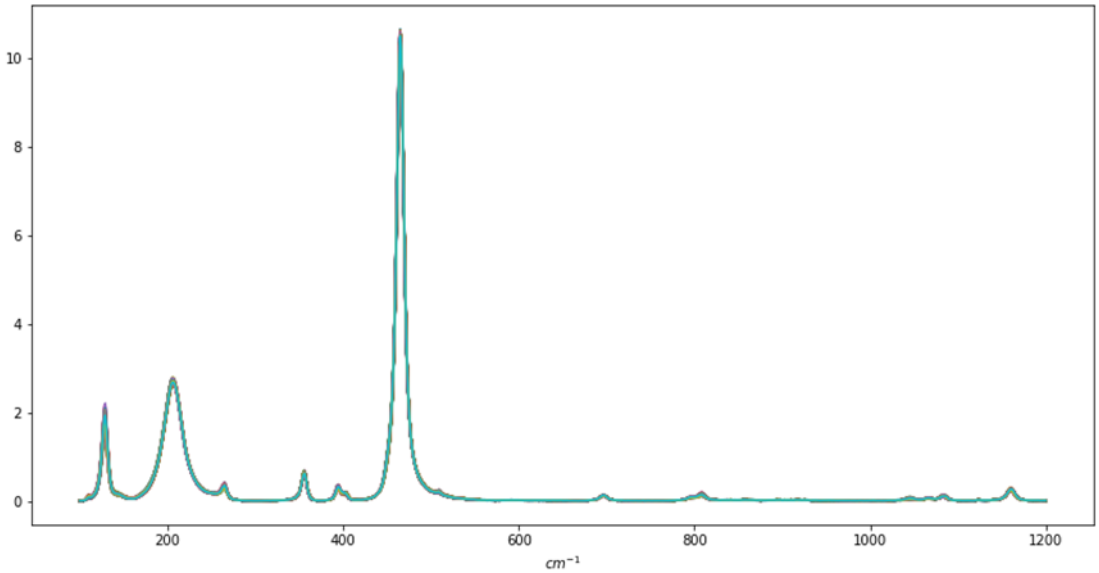


Figura 41 – Dez espectros sintéticos de quartzo sobrepostos.

Realizou-se as medidas em uma amostra de quartzo gerando um mapa no qual selecionamos dez espectros e plotamos juntos:

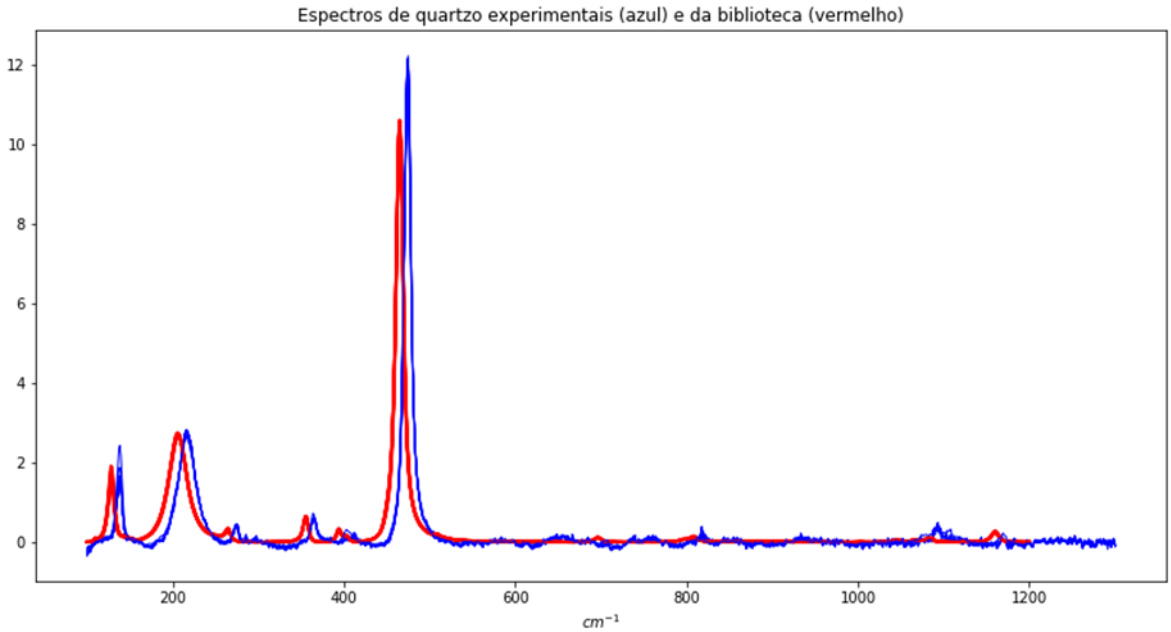


Figura 42 – Dez espectros de quartzo experimentais sobrepostos (azul) e dez espectros sintéticos gerados a partir da biblioteca (vermelho)

A correção do número de onda se deu de forma de forma automática através da distância no número de onda entre o ponto de intensidade máxima do espectro experimental utilizado para calibrar com o ponto de intensidade máxima do espectro da biblioteca. No

nosso caso, subtraiu-se $9cm^{-1}$ obtendo um encaixe melhor entre os espectros experimentais e os espectros sintéticos:

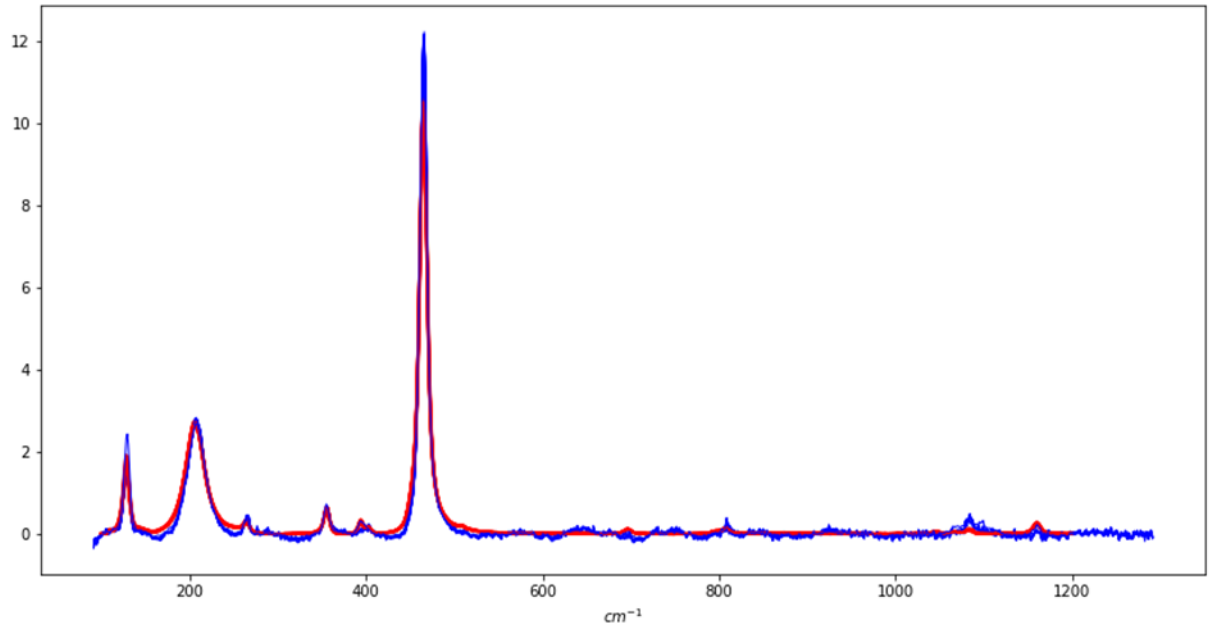


Figura 43 – Espectros experimentais (azul) e espectros sintéticos (vermelho) após calibração.

Após a calibração deseja-se padronizar os eixos de forma que espectros de diferentes fontes tenham o mesmo número de pontos e coordenada de número de onda idênticos. Por vezes os espectros de diferentes fontes possuem espaçamento no eixo cm^{-1} distintos e variantes. A padronização permite organizá-las em conjuntos de dados únicos. Para cada espectro escolhe-se um eixo cm^{-1} que será comum para todos os espectros e as novas intensidades são computadas interpolando os pontos (x_i, y_i) e (x_{i+1}, y_{i+1}) do espectro original que são vizinhos da coordenada do novo ponto x_{new}, ou seja $x_i < x_{new} < x_{i+1}$. Aqui, *x* representa a coordenada cm^{-1} e *y* é a intensidade. A nova intensidade é dada pela reta interpolada:

$$y_{new} = y_i + \left(\frac{y_{i+1} - y_i}{x_{i+1} - x_i} \right) \cdot (x_{new} - x_i)$$

Calculando para cada ponto da coordenada *x* padronizada [36].

4.2 Análise de Componentes Principais

Após o tratamento dos espectros, podemos encontrar as componentes principais do mapa via matriz de covariância e agrupar os dados por componentes. A visualização das componentes de um mapa (e correspondentes concentrações de componentes nas regiões do mapa) facilita a identificação dos materiais que ali se encontram. A análise de componentes principais estimará o número de componentes que visualizaremos utilizando o método de resolução de curvas multivariada.

4.2.1 Análise de componentes principais (PCA)

O PCA é um método de análise exploratória de componentes [43,45] e é aplicado em diversas áreas como neurociência [60], climatologia [61], mercado financeiro [62], análise de sísmica [54], entre outras. Após o tratamento de remoção de linha de base e suavização, calcula-se a matriz de covariância do mapa de espectros. Podemos extrair o correto número de componentes principais do sistema através de uma análise dos autovalores. Extrai-se as raízes do polinômio característico $p(\lambda) = \det(\mathrm{V} - \lambda I)$ que são os autovalores λ_i (V é a matriz de covariância e I é a matriz identidade), o número total de autovalores é igual ao número de espectros do sistema. Ordenando os autovalores do maior para o menor, apenas os primeiros (componentes principais) são necessários para explicar o sistema. Encontramos em seguida os autovetores resolvendo o sistema $Vx = \lambda x$ sendo x o autovetor.

Cada autovetor possui um tamanho *m* idêntico ao número de espectros e cada *i*-ésima entrada do autovetor *j* representa o quanto o *i*-ésimo espectro explica a *j*-ésima componente principal. Tomando o produto escalar do *j*-ésimo autovetor com todos os espectros encontramos a *j*-ésima componente principal. A primeira componente deve ter a maior variância que é calculada na forma $\frac{\lambda_1}{\sum_{i=1}^{n} \lambda_i}$ onde *n* é a quantidade de autovetores.

4.2.1.1 Estimando o número de componentes

Um mapa de 35×35 pixels totalizando 1225 espectros Raman tratados, organizamos seus autovalores em ordem decrescente. Utilizando a expressão $\frac{\lambda_j}{\sum_{i=1}^{n}\lambda_i}$ visualizamos o quanto as componentes relacionadas com os 4 Primeiros autovalores explicam o restante dos espectros:

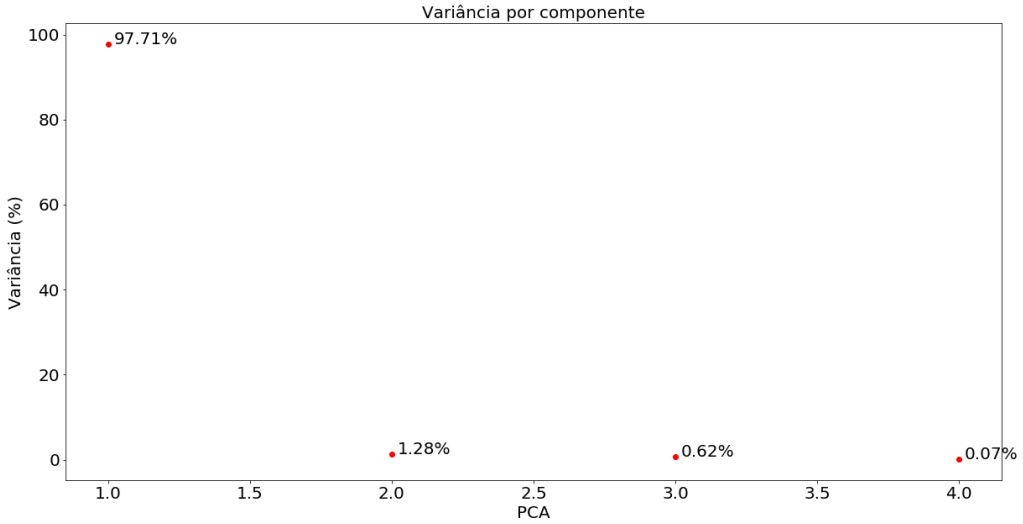


Figura 44 – A Primeira componente possui explica 84,6% de todo o conjunto de 100 espectros.

A variância cumulativa da *j*-ésima componente explica o quanto a *j*-ésima componente junto com todas as componentes anteriores a esta, explicam o conjunto de espectros. Podemos escrever sua expressão na forma $\frac{\sum_{l=1}^{j}\lambda_l}{\sum_{i=1}^{n}\lambda_i}$. Visualizamos na figura a seguir para as quatro primeiras:

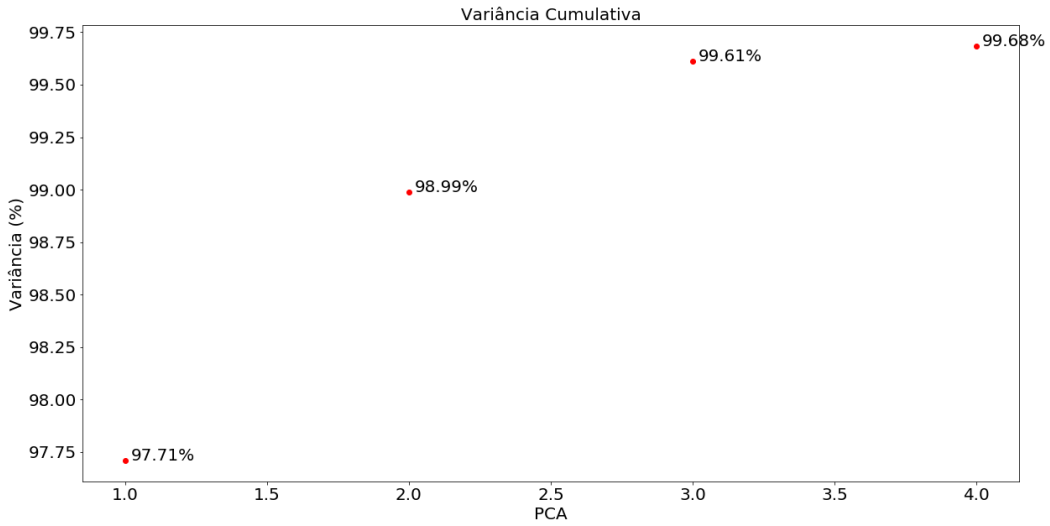


Figura 45 – Variância cumulativa das quatro Primeiras componentes. três componentes explicam 99% do mapa.

Temos uma ótima estimativa do número de componentes para explicar todo o conjunto de espectros, vemos que três componentes já explicam 99% do conjunto. Podemos automatizar o método selecionando automaticamente um número de componentes cuja variância cumulativa seja maior que 99%.

4.2.1.2 Imagem das componentes principais

Abaixo visualizamos as quatro primeiras componentes principais:

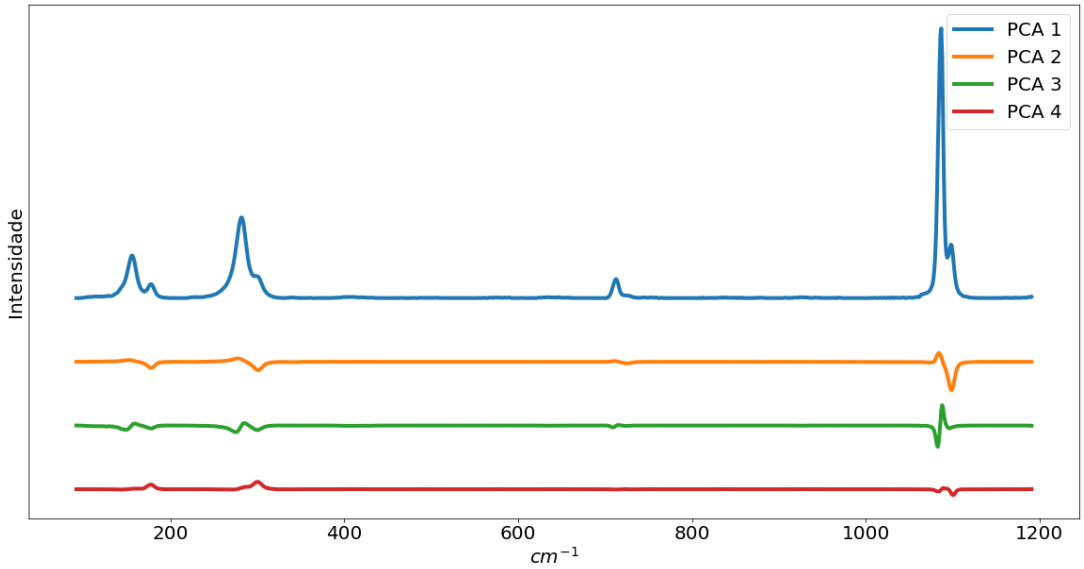


Figura 46 – Quatro Primeiras componentes principais.

Vemos que as componentes principais não representam os espectros das substâncias puras presentes no material. Vale notar o pico negativo na região de 1100 cm^{-1}. Picos negativos não têm significado físico, entretanto ele é necessário para poder cancelar algum outro pico de outra componente. Em suma, uma combinação linear dessas componentes principais gera dois conjuntos de espectros sem os picos duplos nas 4 regiões, abaixo de 200 cm^{-1}, na região de 300 cm^{-1}, na região entre 700 e 800 cm^{-1}, e na região de 1100 cm^{-1}.

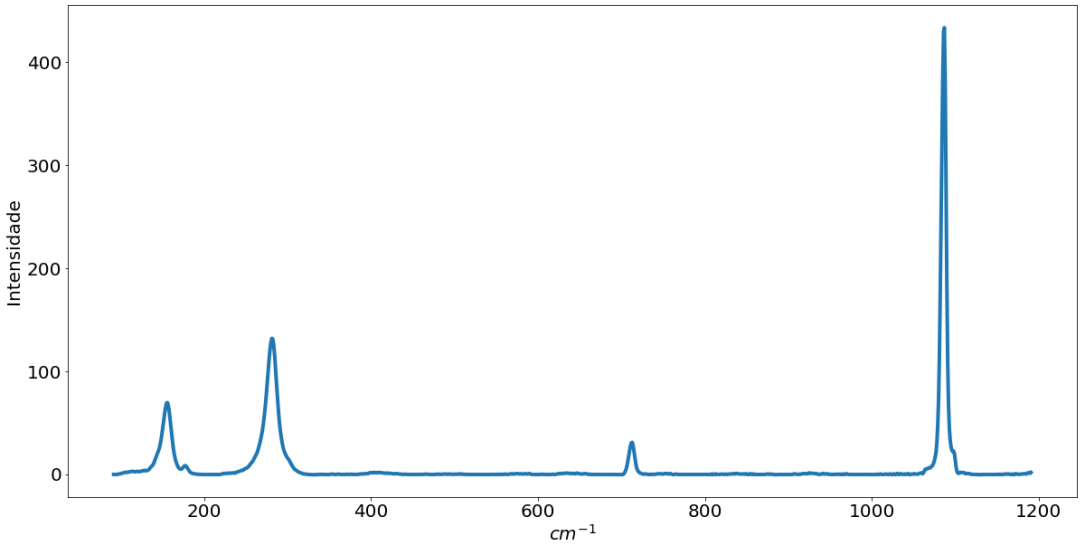


Figura 47 – Combinação da Primeira e segunda componente eliminando o pico duplo.

Sabemos que cada entrada *j* dos autovetores representa a contribuição do *j*-ésimo espectro nas

componentes principais. Podemos visualizar as regiões do mapa com cores diferentes que contribuem para cada autovetor. Utilizando as duas Primeiras componentes obtivemos a seguinte imagem utilizando o *freeware* ImageJ:

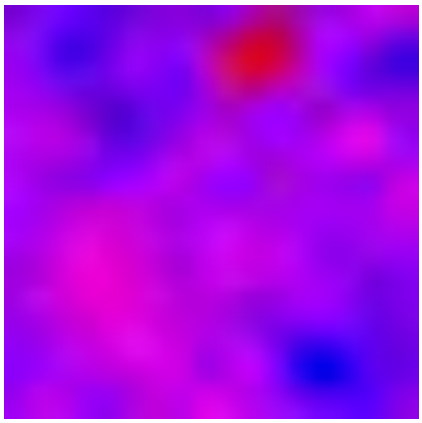


Figura 48 – A região em azul (vermelho) representa regiões onde a componente 1 (2) predomina.

4.2.2 Resolução Multivariada de Curvas (MCR)

Verificamos que as componentes principais não reproduzem padrões espectrais gerando picos negativos mas representam um bom estimador do número *k* de componentes do mapa [44,45]. Utilizando este número *k* de componentes obtido pelo PCA geramos o número de componentes do MCR. [46,47,59]. O MCR é um método iterativo que visa minimizar o erro [48]:

$$D = CS^T + E$$

onde a matriz D representa o espectros e possui dimensão de $n^o\ de\ espectros \times n^o\ de\ pontos\ por\ espectro$. A matriz C possui dimensão de $n^o\ de\ espectros \times n^o\ de\ componentes$ e é multiplicado S^T com dimensão $n^o\ de\ componentes \times\ n^o\ de\ pontos\ por\ componente$. A matriz E possui a dimensão de D e representa o erro dado pela diferença entre a matriz de espectros e o produto CS^T. O número de pontos por componente deve ser igual ao número de pontos por espectro. Assim um espectro de $y \in D$ pode ser escrito como a combinação de k componentes s na forma $y = c_1 s_1^T + c_2 s_2^T + ... + c_k s_k^T$.

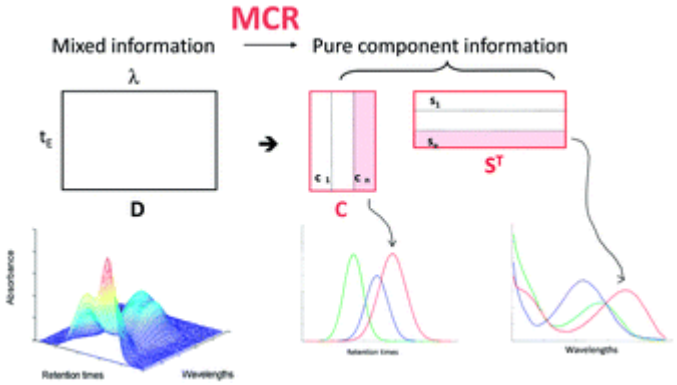


Figura 49 – Fonte [49]: Decomposição dos espectros em componentes e concentrações.

Deve-se utilizar uma estimativa inicial para as concentrações ou para as componentes espectrais. Valores aleatórios podem não ajudar o modelo a convergir para perfis de espectros mais claros, mas para isto seria necessário conhecer os espectros previamente para utilizar como perfis de componente iniciais e encontrar as concentrações [49-51].

4.2.2.1 Estimando os valores iniciais

Uma maneira de encontrar os k espectros do mapa que servirão como estimativa inicial para a matriz S é o algoritmo SIMPLISMA (*Simple-to-use Interactive Self-modelling analysis*) [52]. Inicialmente normaliza-se as intensidades dos espectros na forma:

$$y_i' = \frac{y_i}{\sqrt{\mu_i^2 + \sigma_i^2}}$$

Em seguida calcula-se qual a variável mais pura calculando os índices de pureza de todos os espectros dado por $p_i = \frac{\sigma_i}{\mu_i}$. O espectro puro representa um espectro com gráfico de intensidade dado pelo mínimo de substâncias possível, preferencialmente apenas uma substância representada no espectro [53].

O *j*-ésimo espectro mais puro é selecionado e calculamos os pesos que multiplicarão em seguida cada espectro: $w_i = \det(Y_i * Y_i^T)$ onde Y_i é chamada de matriz de dispersão, neste caso, com duas linhas, o espectro puro na primeira linha e na segunda linha um *i*-ésimo espectro como vemos na Figura 50(a):

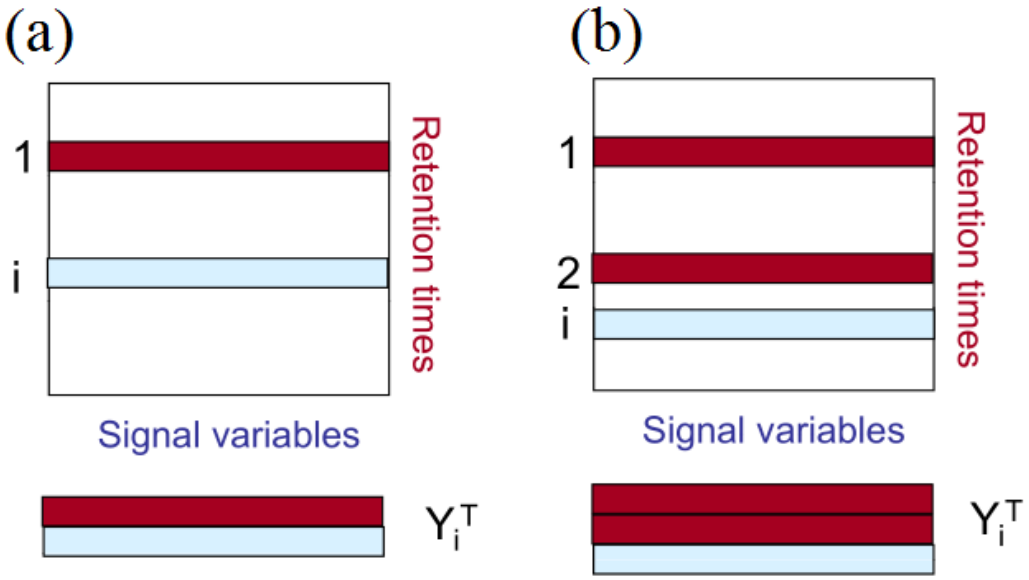


Figura 50 – Fonte [49]. Matriz de dispersão com o Primeiro espectro puro selecionado (a) e com os dois Primeiros espectros puros (b).

Recalcula-se os índices de pureza multiplicando pelos pesos: $p_i' = w_i p_i$. Os espectros que forem mais semelhantes com os puros selecionados receberão um peso baixo e os pesos dos espectros puros já selecionados serão zero devido pois teremos linhas repetidas na matriz de dispesão levando o determinante a zerar.

Ao determinar o segundo espectro puro buscando o espectro com valor máximo em p_i', recalcula-se os pesos de todos os espectros onde Y_i terá agora três linhas como na Figura 50(b). Os dois espectros puros junto com o *i*-ésimo espectro analisado e recalcula-se os índices de pureza multiplicando pelos novos pesos: $p_i'' = w_i p_i$. Repete-se o procedimento até encontrar o número desejado de espectros puros. Os espectros puros servirão de estimativa inicial para a matriz de S^T [51,55]. Obtivemos os seguintes espectros puros:

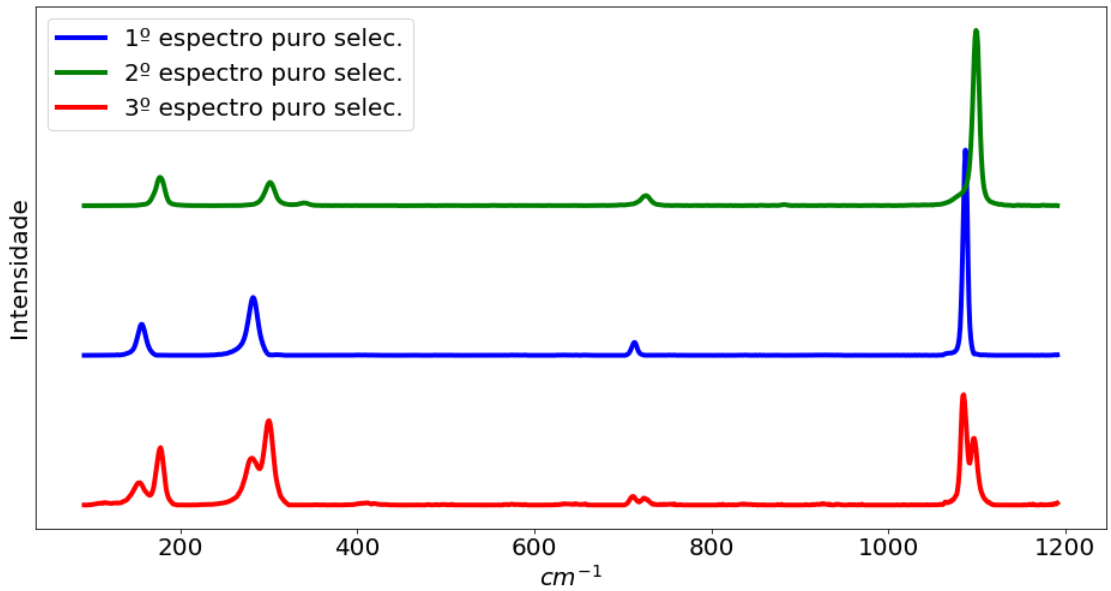


Figura 51 – Três espectros puros selecionados.

Percebemos que o terceiro espectro é uma combinação linear dos dois Primeiros espectros. Calculando a matriz com os coeficientes de correlação para estes três espectros:

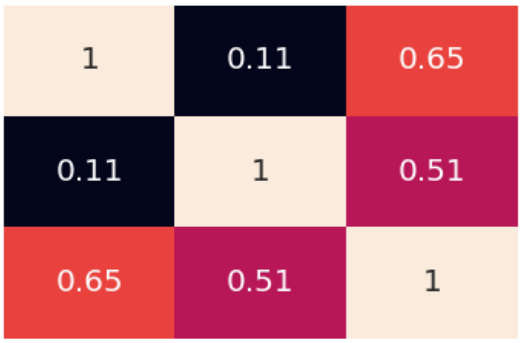


Figura 52 – Matriz de correlação entre os espectros puros.

O terceiro espectro possui alta correlação com o primeiro e segundo espectro. Podemos descartar automaticamente um espectro selecionado como estimativa que possui alta correlação com os demais reduzindo o número de componentes de 3 para 2.

Podemos citar outros métodos conhecidos de detectar os espectros iniciais como *Orthogonal Projection Approach* (OPA) [56], *Key Set Factor Analysis* (KSFA) [57]. Alguns modelos estimam as concentrações iniciais como o *Evolving Factor Analysis* (EFA) [58].

4.2.2.2 Mínimos quadrados alternados (ALS)

O procedimento ALS é semelhante a regressão linear de mínimos quadrados, mas alterna entre minimizar o erro da matriz S e da matriz C. No nosso caso, estimamos S^T e queremos minimizar o erro quadrático $(\mathrm{D}-CS^T)^2$ mantendo S^T constante. Derivando a expressão para o erro quadrático obtemos a concentração $C=(S^TS)^{-1}SD$. Em seguida mantemos C constante e minimizamos o erro quadrático para encontrar $S^T=DC(\mathrm{C}^T C)^{-1}$. Iteragindo alternadamente várias vezes até a convergência das matrizes S e C.

4.2.2.3 Visualização do MCR em um mapa Raman

Iteragimos 100 vezes utilizando o método ALS com a condição de não negatividade dos espectros de forma que as componentes representem padrões espectrais. Utilizamos os dois primeiros espectros puros selecionados no mapa como estimativa inicial obtendo as seguintes componentes após as iterações:

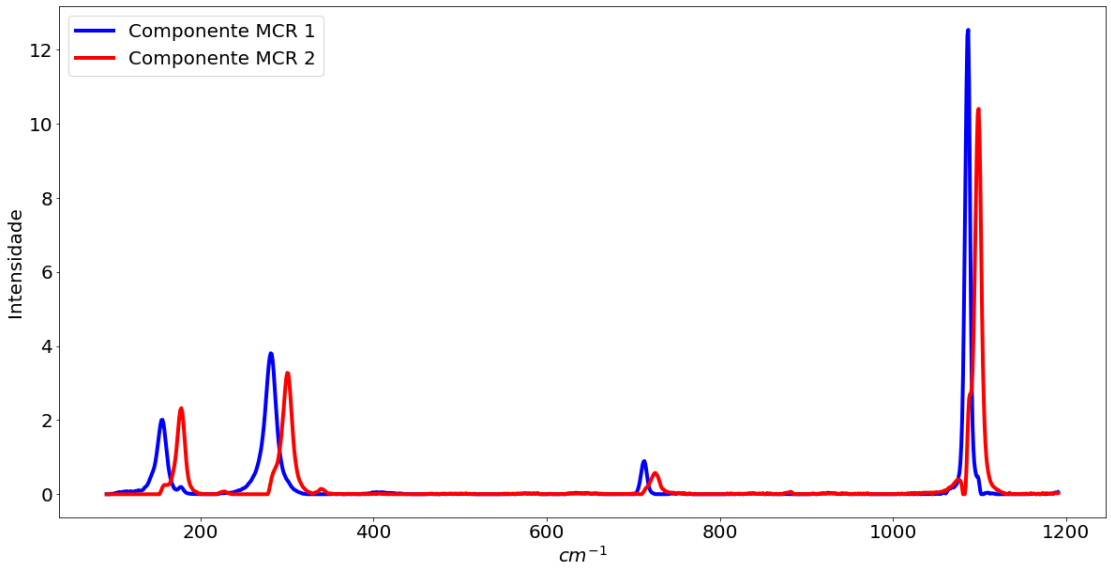


Figura 53 – Componentes obtidas após 100 iterações de mínimos quadrados alternados.

Os espectros obtidos são semelhantes com os espectros obtidos como estimativa inicial. As matrizes de concentrações obtidas para cada componente podem ser observadas como no caso das análise de PCA:

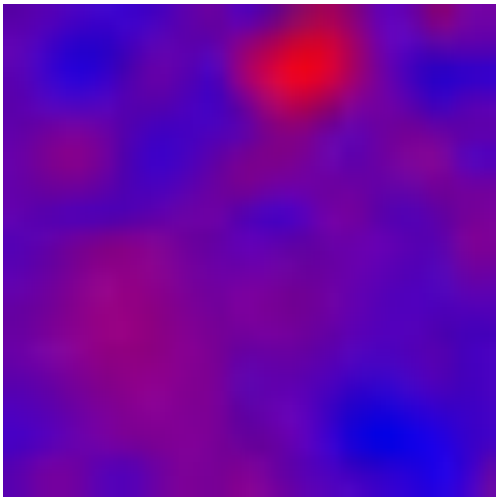


Figura 54 – Regiões onde a componente 1 (2) predomina em azul (vermelho)

Através do método proposto conseguimos agrupar os espectros originais em poucas componentes e identificar zonas onde várias componentes se misturam. Queremos descobrir quais as substâncias presentes na amostra. Na próxima seção desenvolveremos métodos de classificação através de métodos de aprendizagem reforçada capazes de identificar os minerais na amostra.

4.3 Classificação de componentes

Nesta seção definiremos alguns conceitos de classificação de dados espectrais utilizando redes neurais artificiais. Entre os métodos de classificação espectral existentes temos a correlação com uma biblioteca de espectros [82] para identificação [83] indicando uma lista de espectros semelhantes com o novo espectro desconhecido, este método tem péssimo desempenho na classificação de carbonatos cujos espectros são muito semelhantes. A *support vector machine* (SVM) que é semelhante a regressão logística (LR) busca gerar um hiperplano que separa as classes necessitando selecionar as caracteristicas corretas para usar como coordenadas [84].

Avançando para aprendizagem reforçada, temos a classificação utilizando redes neurais artificiais clássicas (ANN) com uma arquitetura capaz de encontrar contornos não-lineares porém é incapaz de resolver problemas de grande escala necessitando extrair as caracteristicas corretas para treinar a rede [85].

Utilizaremos uma abordagem de classificação através de redes neurais convolucionais (CNN) tratando os espectros como imagens unidimensionais [73]. As CNN são modelos

computacionais inspirados no arranjo celular dos neurônios do cortex visual dos mamíferos [86].

A arquitetura das CNN é capaz de extrair as principais características dos dados identificando padrões com um desempenho superior aos humanos [6, 19]. Entre as aplicações das CNN temos o reconhecimento de imagens [70], movimentos humanos [71], voz [72], caligrafia [75], classificação de sentimentos em textos [76], diagnóstico de câncer [77-79], análise de tráfego [80,81] e outros. Os detalhes deste método são encontrados no Apêndice C.

4.3.1 Regressão Logística

Podemos exemplificar o problema de classificação através de um conjunto de pontos em um gráfico:

Plotando todos os pontos selecionados em um gráfico de largura *vs* posição:

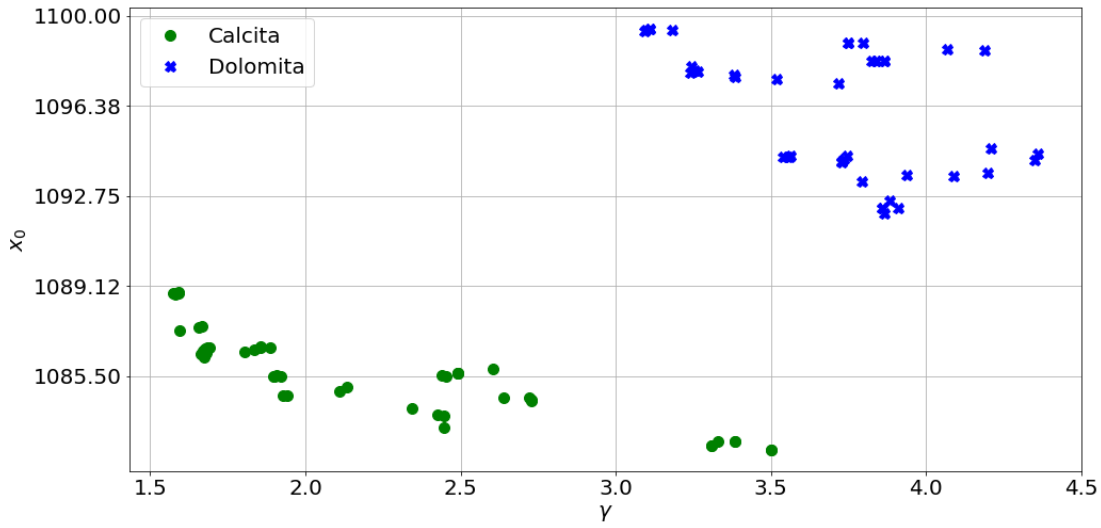


Figura 55 – Posição e largura do pico principal das dolomitas (azul) e calcitas (verde).

A figura acima foi obtida através de espectros de calcita e dolomita do banco de dados RRUFF [63] onde extraimos a largura e número de onda do pico principal dos espectros. A tarefa de classificação é encontrar a reta que é capaz de separar as duas classes.

Após normalizarmos as variáveis, construímos um modelo de ***regressão logística*** que se auto-corrige. Inicializamos uma reta aleatória e caso os pontos verdes (azuis) estiverem abaixo (acima) da reta, estes estão bem classificados, e mal classificados, caso contrário. Os erros são gerados a partir dos pontos mal classificados devendo-se minimizar este erro para encontrar a

reta:

Plotando todos os pontos selecionados em um gráfico de largura *vs* posição:

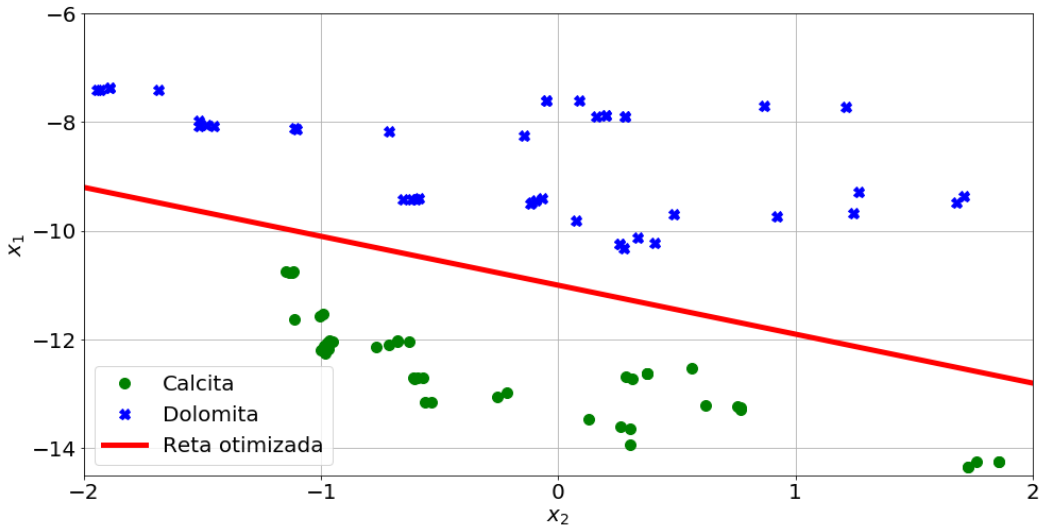


Figura 56: Linha otimizada que separa as classes 0 e 1.

Para novos espectros desconhecidos, ao extraírmos sua larguras e números de onda do pico principal, estes devem ser classificados através desta reta. Este caso é bastante simples e não generalizável, mas nos dará a intuição para um modelo de **redes neurais artificiais** (ANN).

4.3.2 Percéptrons

Os algoritmos ANN são inspirados nos arranjos celulares de neurônios, na figura abaixo temos a representação de um neurônio:

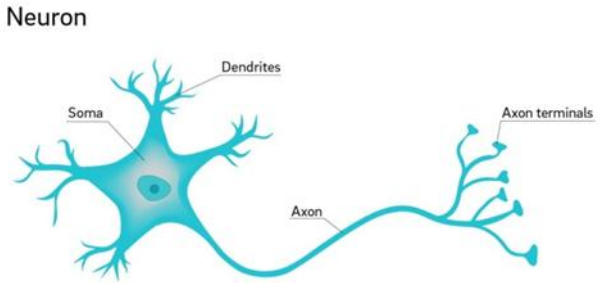


Figura 57 – Representação de um neurônio [87].

Os dendritos recebem sinais elétricos que são interpretados no núcleo e os axônios enviam sinais para outros neurônios. Os *perceptrons* (neurônio artificial) imitam estas células recebendo informação nos nós da rede, processam a informação através de cálculos computacionais e enviam a informação para novos nós. Representamos abaixo um perceptron que simula a regressão logística:

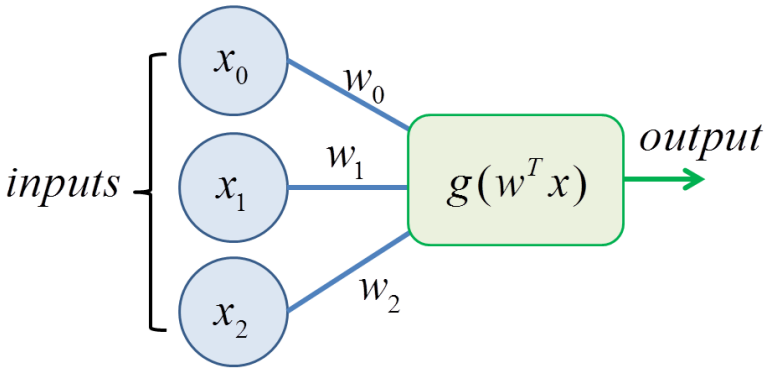


Figura 58 – Um neurônio artificial recebe as entradas x ($w_0 = b$ e $x_0 = 1$) que são multiplicados pelos pesos w e são introduzidos em uma função de ativação g.

Para uma reta dada por $w_1 x + w_2 y + b = 0$ que separa duas classes, se o ponto $(x = x_1, y = x_2)$ está acima da reta, isto é, $w_1 x_1 + w_2 x_2 + b > 0$, este ponto é classificado com sendo da classe 1 através de uma função de ativação, neste caso é a função degrau:

$$g\left(w^T x\right) = H\left(w^T x\right) = \begin{cases} 1, & w^T x \geq 0 \\ 0, & w^T x < 0 \end{cases}$$

4.3.3 Redes Neurais Artificiais

Os percéptrons podem ser combinados formando ***percéptrons multi-camadas*** (MLP). As MLP são redes neurais que possuem várias camadas. Uma camada com as entradas e uma de saída como no percéptron, na Figura 59 temos na camada vertical central três nós $a_1^{(2)}, a_2^{(2)}, a_3^{(2)}$ que funcionam como três percéptrons cujas saídas se conectam em um novo percéptros $a_1^{(3)}$. Chamamos de camadas ocultas as camadas intermediárias entre a saída e entradas. Os nós $a_0, \ldots, a_n$ são chamados de unidades de ativação. O nó $a_i^{(j)}$ representa a unidade de ativação *i* na camada *j*. Os nós $a_i^{(1)}$ são as entradas.

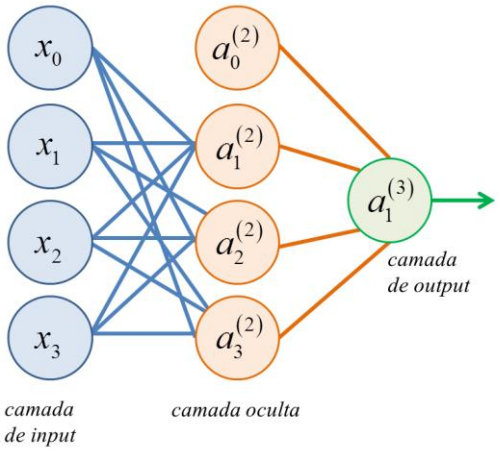


Figura 59 – Representação de uma rede neural com uma camada oculta. O nó $a_0^{(2)}$ representa um bias na segunda camada que não está conectada com a Primeira camada.

Vemos que a arquitetura da MLP é uma estrutura semelhante aos grafos. Vimos que cada unidade de ativação recebe os valores anteriores multiplicando pelos pesos, somando e o resultado é inserido na função de ativação. Podemos pensar neste processo de **alimentação da rede** como um grafo onde as conexões são arcos que levam os valores das camadas anteriores para as próximas [9]. As redes neurais podem ter diferentes arquiteturas com muitas camadas ocultas, diferentes números de nós da rede e classificar várias classes.

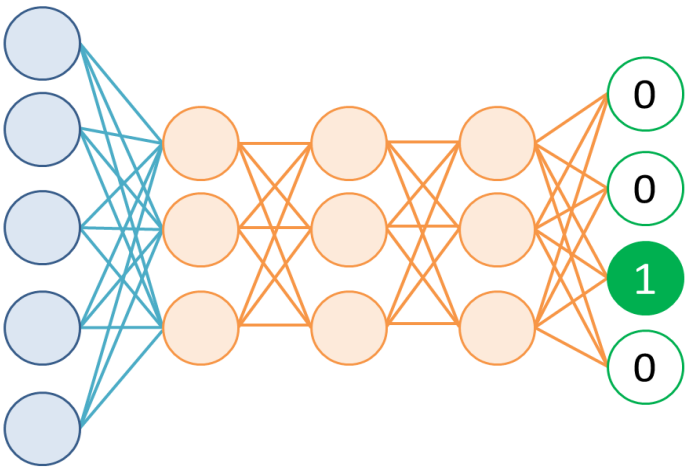


Figura 60 – Arquitetura de uma rede neural com três camadas ocultas e quatro classes.

Na figura acima temos uma rede neural com quatro nós na camada de saída. A última camada é então um vetor de dimensão quatro. A saída deve ser da forma:

$\bar{a}^{(5)} = \begin{bmatrix} a_1^{(5)} & a_2^{(5)} & a_3^{(5)} & a_4^{(5)} \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 \end{bmatrix}$ se *x* é da primeira classe ou $\bar{a}^{(5)} = \begin{bmatrix} 0 & 1 & 0 & 0 \end{bmatrix}$ é da segunda classe e assim sucessivamente.

As MLP aprendem os quais os parâmetros de peso dos nós através do **treino**. O treino consiste em entregar dados já classificados (amostras que já se sabe previamente o que é) à rede neural. A MLP gera um modelo inicialmente com pesos (e bias) aleatórios (como os coeficientes angulares da reta do caso da regressão logística). Alimenta-se a rede com os inputs e as camadas ocultas devem realizar diversas operações de multiplicação dos inputs com pesos e soma com um *bias* e o resultado deve ser ruim pois os pesos foram iniciados aleatoriamente, porém cada amostra de treino já foi previamente classificada e com isso é computado o erro que é a distância entre o output e a classe correta. Computa-se a derivada do erro em relação aos pesos das camadas anteriores e iterage-se até convergir. Cada iteração é chamada de **época** de treino. Após o treino da MLP, esta deve retornar a classe de amostra desconhecida utilizada como entrada no modelo gerado.

4.3.4 Redes Neurais Convolucionais

As redes neurais da sessão anterior são chamadas de redes neurais de camada densa. Nesta sessão vamos definir dois tipos de camadas chamadas camadas convolucionais e camadas de agrupamento. Ambas serão utilizadas no problema de classificação das imagens unidimensionais dos espectros. Redes neurais convolucionais (CNN) possuem diversas aplicações como vimos na introdução do capítulo. Uma das aplicações mais documentadas é em *visão computacional* que é aplicada em reconhecimento de imagens em carros autônomos [105]. Com esta motivação vamos definir as CNN, demonstrar como aplicar CNN em classificação de imagens e mostrar o quão estas são superiores as redes neurais convencionais MLP empregadas em detecção de imagens e problemas de grande escala (uma única imagem em escala de cinza com 1000×1000 pixels possui 10^6 *inputs*). Por fim aplicaremos as CNN na tarefa de classificar espectros de minerais [73] e classificar as componentes obtidas no MCR do capítulo anterior.

Uma camada convolucional corresponde a filtros que são aplicados na imagem:

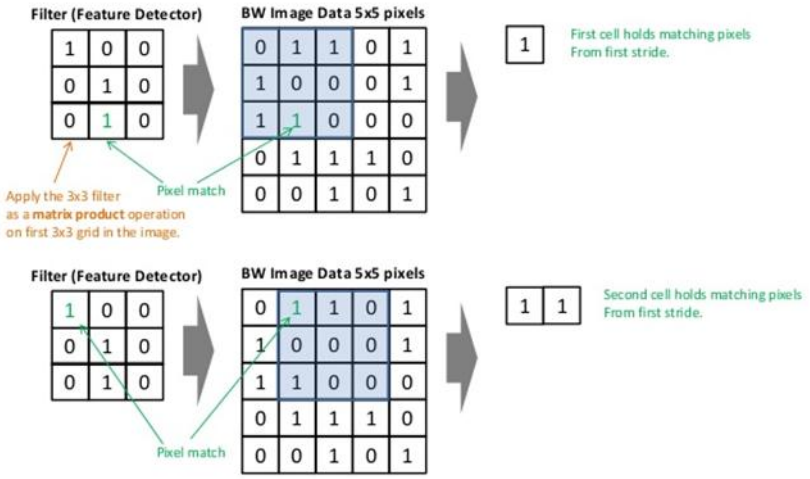


Figura 61 – Funcionamento da extração de caracteristicas realizada pelos filtros.

Na figura acima temos uma pequena matriz 5×5 (no centro) que é a imagem analisada, no lado esquerdo temos um filtro que é uma matrix 3×3. A convolução da matriz da imagem x com o filtro w resulta em uma nova matriz: $x*w=z$. A operação de convolução se dá na seguinte forma: A matriz 5×5 é decomposta em várias submatrizes do tamanho do filtro 3×3 onde cada elemento do filtro w_{ij} multiplica o elemento na mesma í-esima linha e j-ésima coluna da submatriz de x_{ij} destacada em azul na Figura 61 no canto superior central e em seguida soma-se todas as multiplicações formando o primeiro output *output* $\sum_{k=1}^{3}\sum_{l=1}^{3} w_{kl}x_{kl} = w_{11}x_{11} + w_{12}x_{12} + ... + w_{33}x_{33} = z_{11}$. Os índices vão de 1 a 3 onde 3 é a altura/largura do filtro. Para simplificar, trataremos apenas imagens quadradas. No exemplo da Figura 61 vemos no canto superior direito que $z_{11}=1$.

A próxima submatrix é representada na parte inferior central destaca em azul na Figura 61 onde movimentou-se uma coluna para direita selecionando uma nova submatriz. O output é $\sum_{k=1}^{3}\sum_{l=1}^{3} w_{kl}x_{k,l+1} = w_{11}x_{12} + w_{12}x_{13} + ... + w_{33}x_{34} = z_{12}$.

Os outros elementos são dados pela fórmula:

$$z_{ij} = \sum_{k=1}^{3}\sum_{l=1}^{3} w_{kl}x_{k+i-1,l+j-1}. \qquad \text{(C.4)}$$

A matriz final desta operação deve ser 3x3 pois só se consegue encaixar o filtro 3x3 na matriz 5x5 em 3x3 posições diferentes com um passo de cada vez. O tamanho da matriz resultante então é $\zeta^{(2)}\times\zeta^{(2)} = \left(\zeta^{(1)} - \nu + 1\right)\times\left(\zeta^{(1)} - \nu + 1\right)$. Onde $\zeta^{(2)}$ é a altura/largura da nova matriz

gerada, $\zeta^{(1)}$ é a altura/largura da imagem original e ν a largura do filtro. Definimos então os possíveis valores de *i* e *j*: $i, j \in \{1, 2, ..., \zeta^{(1)} - \nu + 1\}$.

A arquitetura utilizada é chamada de comumente de LeNet-5 [116] e é constituída de várias camadas convolucionais conectadas e estam conectam-se com camadas densas (MLP) gerando o output da classe. As camadas convolucionais extraem características invisíveis ao olho humano enquanto que as camadas densas geram a classificação.

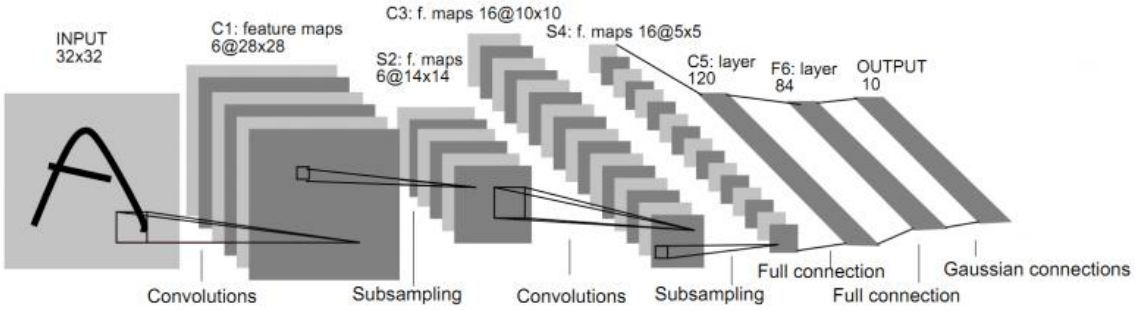


Figura 62 – Fonte [117]. Arquitetura *LeNet-5*.

Os pesos da CNN são computados de maneira semelhante aos pesos da MLP. CNN pode ser utilizada na tarefa de classificar grafos através de suas caracteristicas, por exemplo, dado um grande número de grafos, classificar entre grafos completos ou árvores. Em uma aplicação real podemos exemplificar com um conjunto de dados de móleculas representadas por grafos como input e deseja-se classificá-las quanto o seu nível de atividade contra câncer para classificar automaticamente novas amostras [10]. Existem também redes neurais definidas que recebem grafos como inputs no lugar de vetores e matrizes e são chamadas de *Graph Neural Networks* (GNN) [118].

4.3.5 Aplicação em Raman

Geramos uma CNN para imagens unidimensionais usando os dados de treino da RRUFF [63] para seis minerais distintos de amostras orientadas. A princípio poderiamos generalizar este modelo para um grande número de diferentes minerais treinando a rede utilizando computação em nuvem. Para cada conjunto de dados coletados separou-se cerca de 70% de espectros para o treino e 30% para o teste.

mineral	total	train	test
Magnesite	29	20	9
Fayalite	36	25	11
Dolomite	52	36	16
Calcite	52	36	16
Forsterite	85	59	26
Quartz	24	16	8

Tabela 1 – Número de espectros coletados e separados em treino e teste.

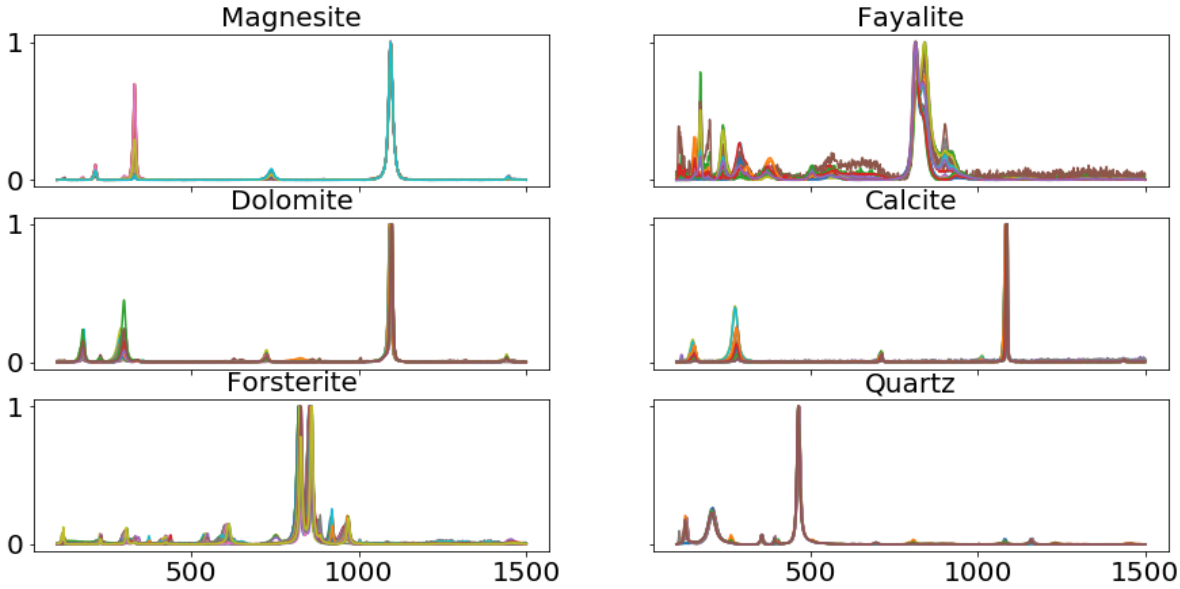


Figura 63 – Espectros selecionados para o treino.

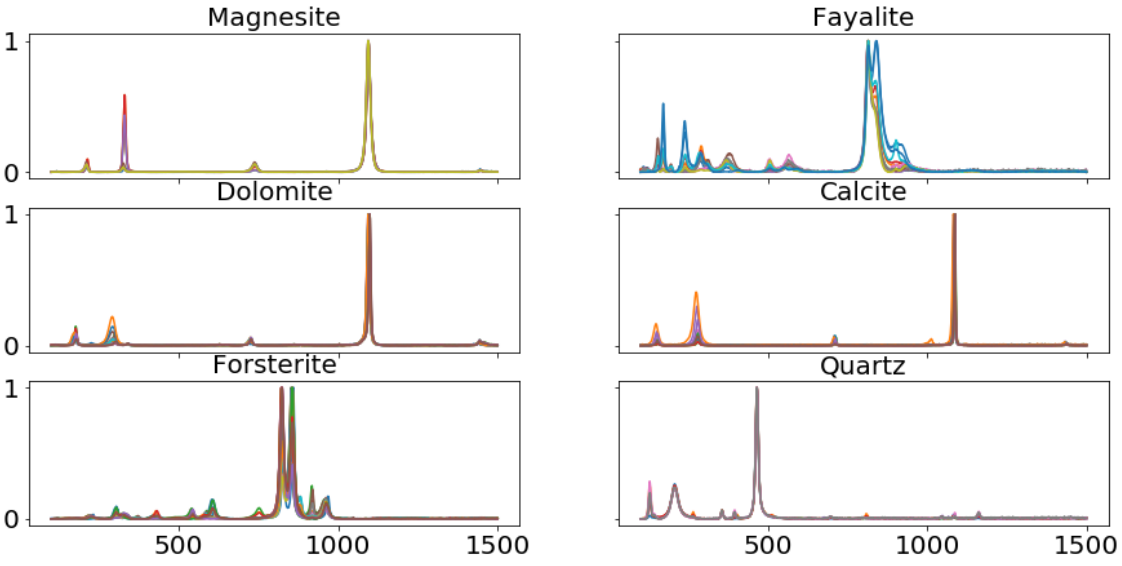


Figura 64 – Espectros selecionados para o teste.

Como o número de espectros é muito pequeno e as CNN são modelos famintos por dados, geramos novos espectros através de combinações lineares dos espectros de cada subconjunto adicionado de ruído. Geramos 500 espectros de cada um dos 6 minerais totalizando 3000 espectros para o treino e 200 espectros de cada para o teste totalizando 1200. Os espectros tem um total de 1400 pontos cada. Temos 3000 espectros de treino de forma que o *input* será uma matriz de dimensão $1400 \times 1 \times 3000$. A rede foi capaz de identificar com 100% de acurácia os espectros de teste. Em seguida aplicamos o modelo nas três componentes geradas através do MCR identificando que a componentes 1 é uma calcitas e a componente 2 é dolomita.

5 MERCADO DE OPÇÕES

O mercado de derivativos é o mercado onde são negociados ativos que derivam de outro ativo, chamado de ativo-objeto, como por exemplo, o mercado futuro, onde se vende hoje o ativo no futuro. Trata-se, essencialmente, de um mercado voltado à diminuição de risco dos participantes. Produtores agrícolas estão entre os principais agentes que participam desse mercado, porque só podem vender seus produtos após a colheita. Dentre os derivativos estão as opções, de compra ou venda, nas quais se negociam no presente, a opção de comprar, ou vender, o ativo objeto por um preço determinado. Isso significa que o titular da opção pode comprar, ou vender, no final do contrato o ativo objeto pelo preço combinado, mas sem a obrigação de comprar ou vender. Contrário ao mercado futuro onde os dois lados assumem o compromisso de comprar/vender pelo preço estipulado. Assim, no mercado de opções, se o preço no final do contrato estiver abaixo do estipulado, o titular da opção de compra prefere comprar pelo preço à vista. Mas se o preço à vista estiver mais alto ele exerce sua opção e o vendedor da opção é obrigado a vender pelo valor combinado. Isso mostra que o titular da opção tem todos os direitos e o vendedor todas as obrigações. Assim, apenas cobrando um prêmio no início do contrato existem vendedores de opções. O cálculo desse prêmio sempre foi o problema mais relevante para o estabelecimento desse mercado – se o prêmio for muito pequeno os vendedores de opção irão à falência, e se for muito alto, não existirão compradores. Se o mercado persiste por tempos longos é porque nem os vendedores faliram e os compradores estão satisfeitos com a diminuição de riscos que ele proporciona.

A equação de Black & Scholes [B&S] utilizada para precificar opções europeias é conhecida como fórmula de Midas, publicada em 1973 em um artigo chamado *The pricing of options and corporate liabilities*. Black e Scholes desenvolveram sua equação exatamente para precificar o menor prêmio de opções europeias, na qual o titular só pode decidir exercer ou não a opção no final do contrato na mesma época em que surgiu a maior bolsa de opções até hoje, a Chicago Board Options Exchange [CBOE] [121]. Além de encontrarem uma fórmula simples, eles mostraram como o vendedor da opção podia se proteger dos riscos inerentes da operação, que foi a contribuição importante de Merton. O sucesso dessa fórmula foi tão grande que Scholes e Merton ganharam o prêmio Nobel de Economia em 1997 [Black não ganhou porque morreu em 1995] [137]. Apenas 6 meses após a publicação do trabalho a Texas Instruments lançou no mercado uma calculadora com o anúncio: “Agora você pode encontrar o valor de Black-Scholes usando nossa calculadora”. Ian Stewart incluiu a equação de B&S no seu livro “17 equações que mudaram o mundo”, chamando-a de fórmula de

Midas, o rei da lenda grega que transformava tudo o que tocava em ouro. Segundo Stewart: “Em 1998, o sistema financeiro internacional negociou aproximadamente 100 trilhões de dólares americanos em derivativos. Em 2007, esse valor havia crescido para 1 quatrilhão de dólares... Para contextualizar o número, o valor total de todos os produtos fabricados pelas indústrias de todo o mundo, nos últimos mil anos, é de cerca de 100 trilhões de dólares americanos, corrigidos pela inflação. Isso equivale a um décimo dos negócios com derivativos em um ano” [134].

A maioria dos contratos de opção negociados no mercado é na modalidade americana, que pode ser exercida a qualquer momento até o contrato expirar, e não a europeia, objeto da fórmula de B&S. O modelo de Cox-Ross-Rubinstein [CRR], embora bem mais simplificado do que o modelo B&S, permite, entretanto, precificar opções americanas. Além disso, conforme demonstraremos, ele converge para o B&S no limite de muitos passos discretos. Nesse trabalho mostraremos como utilizar o CRR para expandir o B&S no caso de opções americanas.

Iremos abordar um pouco de matemática financeira, abordaremos uma visão geral do mercado financeiro concentrando nos derivativos, particularmente nas opções, que é um derivativo. Neste ponto discutiremos características das opções como suas rentabilidades para diferentes operações e seus correspondentes limites. Definiremos o modelo CRR e o modelo de Black & Scholes. Mostraremos que o modelo CRR converge para a fórmula do Black & Scholes e deduziremos em seguida a B&S da maneira que foi feito no trabalho original utilizando equação da difusão.

5.1 Conceitos básicos de Finanças

Esta seção tem como objetivo abordar vários conceitos sobre finanças, iremos explorar um pouco da matemática envolvida definindo grandezas fundamentais, juros e retornos. Estes assuntos simples servirão como base para desenvolvermos o restante do trabalho desta dissertação.

5.1.1 Grandezas Fundamentais

Vamos definir, inicialmente, três grandezas básicas:

- Estoque: Definiremos com o símbolo $, representa a quantidade de moeda

generalizada que se possui em determinado momento do tempo. Pode ser dado em unidades monetárias como o real, dólar, euro ou outras;

- Retorno: Para um investimento inicial de $\$_t$ que Δt depois valerá $\$_{t+\Delta t}$, o retorno deste investimento é definido na seguinte forma: $R = \frac{\$_{t+\Delta t} - \$_t}{\$_t}$ e a taxa de retorno é dada por $\tau = \frac{1}{\$_t}\left(\frac{\$_{t+\Delta t} - \$_t}{\Delta_t}\right)$. Portanto $R = \tau \Delta t$.

Vemos que o retorno é adimensional e geralmente escrita como uma porcentagem, a taxa de retorno é mais indicada para ver esta porcentagem relacionada a um determinado período de tempo.

5.1.2 Evolução dos Retornos

Inicialmente temos um retorno R_0 no tempo $t = 0$ e em $t = 1$ teremos R_1 e após n períodos chegamos no tempo $t = n-1$ com o retorno R_{n-1}. O valor final do nosso investimento, seguindo a regra dos juros compostos, fica:

$\$_n = (1+R_{n-1})(1+R_{n-2})\ldots(1+R_0)\$_0.$

Definindo a variável auxiliar $Z = 1+R$ para facilitar os nossos cálculos, a expressão acima fica na forma:

$\$_n = Z_{n-1}Z_{n-2}\ldots Z_0\$_0.$ O Z_{av} é dado pela média geométrica

$$Z_{av} = \left(Z_{n-1}Z_{n-2}\ldots Z_0\right)^{1/n} = \left(\prod_{j=0}^{n-1} Z_j\right)^{1/n}.$$

Desta forma podemos transformar um retorno anual em diário na forma $Z_{dia} = Z_{ano}^{1/252}$ onde utilizamos o número de dias úteis de um ano que é 252 pois no mercado financeiro só se opera em dias úteis.

Vamos supor que compramos uma ação em $t = 0$ por p_0 que evolui até p_n no $t = n$ de modo que no dia 1 obtivemos

$$R_1 = \frac{p_1 - p_0}{p_0} \qquad \rightarrow \qquad Z_1 = 1 + R_1 = \frac{p_1}{p_0}.$$

Após os n períodos teremos $Z_{total} = \frac{p_n}{p_0}$, ou seja

$$Z_{total} = \frac{p_n}{p_{n-1}} \frac{p_{n-1}}{p_{n-2}} \ldots \frac{p_1}{p_0} = Z_n Z_{n-1} \ldots Z_1.$$

Vemos que estamos trabalhando com um crescimento exponencial, um processo multiplicativo. Para ransforma e é preferível sempre trabalhar com processos aditivos, portanto iremos aplicar o logaritmo natural na Z_{total} :

$\ln\left(Z_{total}\right) = \ln\left(Z_n\right) + \ln\left(Z_{n-1}\right) + \ln\left(Z_{n-2}\right) + \ldots + \ln\left(Z_1\right)$. A partir disto iremos definir o **log-retorno** $r = \ln\left(Z\right)$ e a expressão acima fica na forma:

$$r_{total} = r_n + r_{n-1} + \ldots + r_1.$$

Aplicando o logaritmo na expressão com o produtório para Z_{av} obtemos

$$\ln\left(Z_{av}\right) = \ln\left[\left(Z_{n-1} Z_{n-2} \ldots Z_0\right)^{1/n}\right] = \frac{1}{n}\left[\ln\left(Z_{n-1}\right) + \ln\left(Z_{n-2}\right) + \ldots + \ln\left(Z_0\right)\right]$$

daí a média do log-retorno fica:

$$r_{av} = \frac{r_{n-1} + r_{n-2} + \ldots + r_0}{n} = \sum_{j=0}^{n-1} \frac{r_j}{n}$$

que é uma média aritmética e não geométrica como a que vimos anteriormente [124]. Através do log-retorno podemos obter o valor presente de um ativo que em um tempo T e aplicado a uma taxa r nos obterá o ativo com um valor S multiplicando por uma exponencial na forma $S_0 = e^{-rT} S$.

5.2 Mercado financeiro e Opções

Nesta seção falaremos um pouco sobre a estrutura do mercado financeiro, seus ativos e suas subdivisões, iremos estudar uma ramificação que é o mercado de opções. Vamos conhecer os agentes que fazem o mercado funcionar, como é dado o lucro destas opções e definiremos seus limites. Iniciaremos com um exemplo de uma MST das ações S&P 500 [2] que é um índice das 500 maiores companhias nas bolsas NYSE, NASDAQ e Cboe BZX Exchange. O S&P 500 foi desenvolvido e é mantido pela S&P Dow Jones Indices. Entre algumas das companhias que estão no índice podemos citar Amazon, Apple, Facebook, FedEx, General Motors, Goldman Sachs, HP, Intel, Morgan Stanley, entre outras.

O mercado financeiro oferece recursos para a realização de investimentos. O financiamento pode ocorrer através de um banco que intermedia os poupadores e o agente demandante de recursos cobrando a diferença das taxas de juros de cada um.

Podemos receber financiamento através do mercado de capitais que financiam através das *bonds* (títulos de dívidas) que são associadas ao mercado de renda fixa, o titular da *bond* torna-se credor da empresa, o titular empresta dinheiro para a instituição onde pode receber juros ou receber um valor prefixado ao final do período de investimento.

Temos também as *equities* (participações) onde o detentor pode tornar-se sócio de uma empresa e receber lucros, dentre elas temos os *stocks* (ações) onde os detentores podem receber *dividendos* que é uma parte dos lucros da companhia distribuída aos acionistas e é isento de Imposto de Renda (IR), depende apenas do lucro líquido da empresa ou *juros sobre capital próprio* (JSCP) que distribui lucros dos períodos anteriores que são alocados como despesas financeiras, reduzindo o IR incidente.

As negociações são realizadas na BM&FBovespa, no Brasil. Os *bonds* (títulos) são negociados na SELIC – *Sistema Especial de Liquidação e Custódia* – quando são públicos e CETIP – *Central de Custódia e de Liquidação de Títulos Privados* para os títulos privados. Dentro do mercado financeiro temos diversos tipos de mercados, como o mercado de renda fixa, mercado de renda variável, mercado de derivativos, mercado de câmbio e de fundos de investimentos. Neste trabalho trataremos especialmente do mercado de derivativos.

5.2.1 Aplicação de MST no mercado financeiro

Aplicaremos o algoritmo de PRIM de MST nas ações do S&P 500. Um requerimento para o uso de funções estatísticas como a correlação é que o dado seja estacionário, senão o resultado não terá sentido. Em geral, a correlação direta entre os preços das ações não possuem sentido. A razão para isto é que os preços não são estacionários, possuem tendências. Para isto obtemos o log-retorno das ações dado por $\ln\left(\frac{S_t}{S_{t-1}}\right)$ onde S_t é o preço de uma ação no tempo *t*. O log-retorno também é útil no cálculo da covariância entre as ações, a covariância é importante no cálculo de portfólios ótimos (carteiras de ações que minimizam o risco) como os modelos de Markowitz, Black e Tobin.

Obtivemos os dados dos preços das ações através da fonte Yahoo! Finance do ano de 1/2010 a 1/2019. Mantemos apenas a informação de 458 ações em que os dados estão disponíveis em todo o período, a lista está disponibilizada no apêndice E. Através disto calculamos os log-retornos e em seguida a sua correlação. Organizamos as ações setores [2] e geramos a seguinte MST:

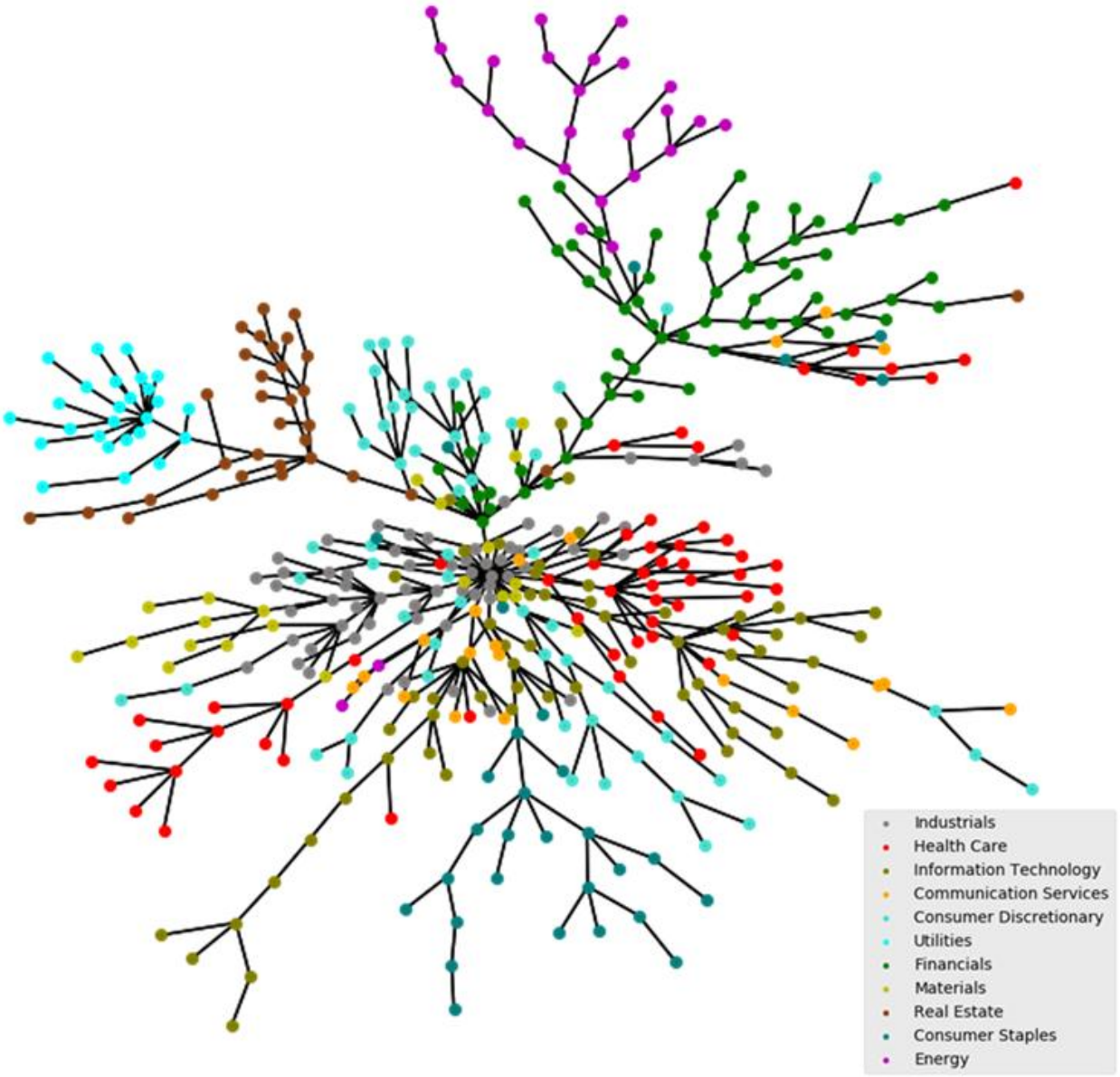


Figura 65 – Minimum Spanning Tree de 458 ações do S&P 500 Index.

Vemos que algumas ações formam clusters mais dependentes do seu setor em si nas regiões periféricas, enquanto que em nas regiões centrais os ativos se misturam mostrando uma correlação entre diversos setores. Reorganizamos a matriz de correlação a partir da ordem de vértices que aparecem nas arestas do algoritmo de PRIM, assim obtivemos a seguinte matriz visualizada através do mapa de calor:

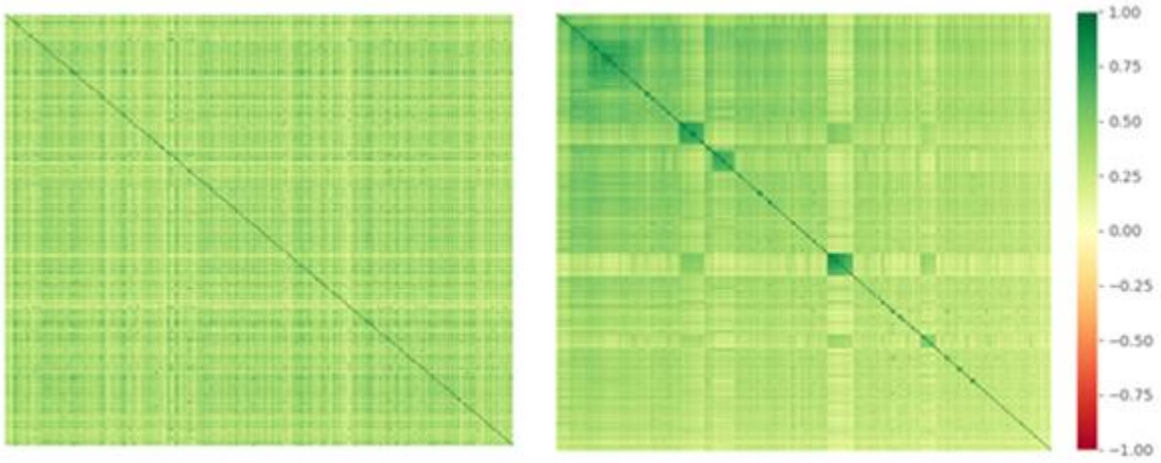


Figura 66 – Mapa de calor da matriz de correlação não organizada (esquerda) e organizada através do algoritmo de PRIM (direita).

As regiões quadradas mais escuras correspondem a clusters formados pelas ações [143]:

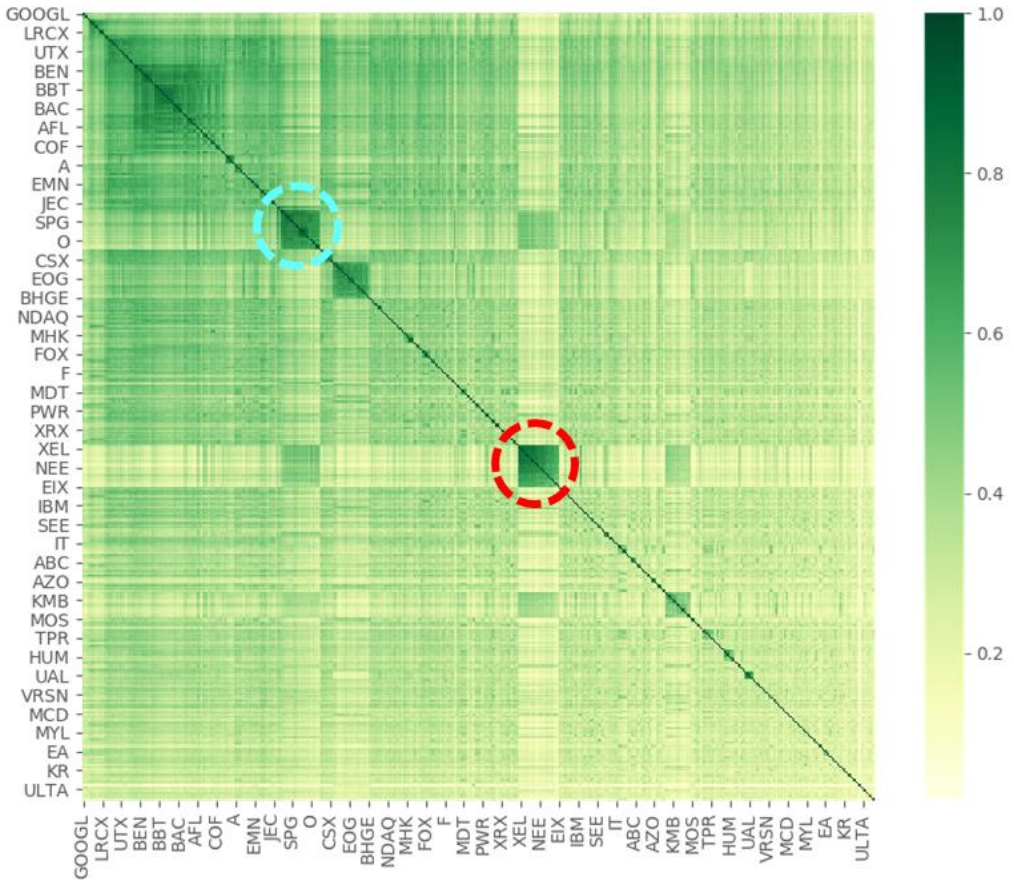


Figura 67 – Clusters formados, destacamos dois em azul e vermelho.

Aproximando a imagem no cluster do círculo vermelho:

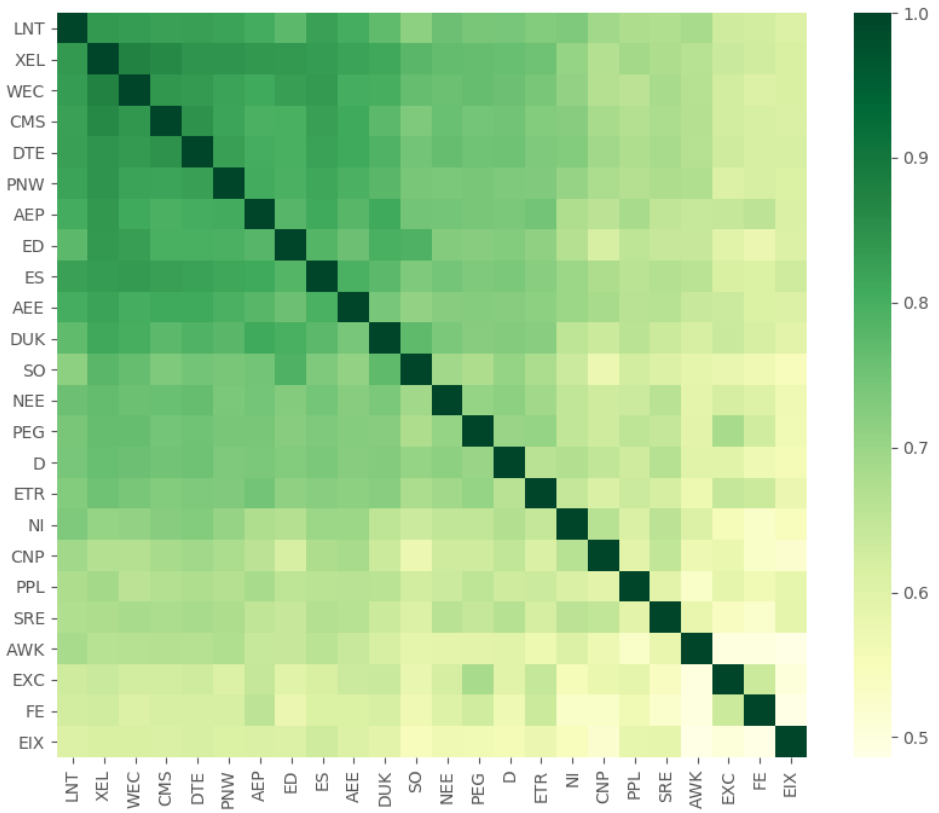


Figura 68 – Cluster aproximado, destacado em vermelho na Figura xx.

Tick	Ações	Setor
LNT	Alliant Energy Corp	Utilities
XEL	Xcel Energy Inc	Utilities
WEC	Wec Energy Group Inc	Utilities
CMS	CMS Energy	Utilities
DTE	DTE Energy Co.	Utilities
PNW	Pinnacle West Capital	Utilities
AEP	American Electric Power	Utilities
ED	Consolidated Edison	Utilities
ES	Eversource Energy	Utilities
AEE	Ameren Corp	Utilities
DUK	Duke Energy	Utilities
SO	Southern Co.	Utilities
NEE	NextEra Energy	Utilities
PEG	Public Serv. Enterprise Inc.	Utilities

D	Dominion Energy	Utilities
ETR	Entergy Corp.	Utilities
NI	NiSource Inc.	Utilities
CNP	CenterPoint Energy	Utilities
PPL	PPL Corp.	Utilities
SRE	Sempra Energy	Utilities
AWK	American Water Works Company Inc	Utilities
EXC	Exelon Corp.	Utilities
FE	FirstEnergy Corp	Utilities
EIX	Edison Int'l	Utilities

Tabela 2 – Ações do cluster destacado em vermelho na Figura xx.

Correspondendo ao mesmo setor, destacado na mesma cor dos vértices deste setor na MST da Figura 65. Fazendo a mesma análise no cluster destacado em azul na Figura 67, visualizamos o cluster aproximando a matriz:

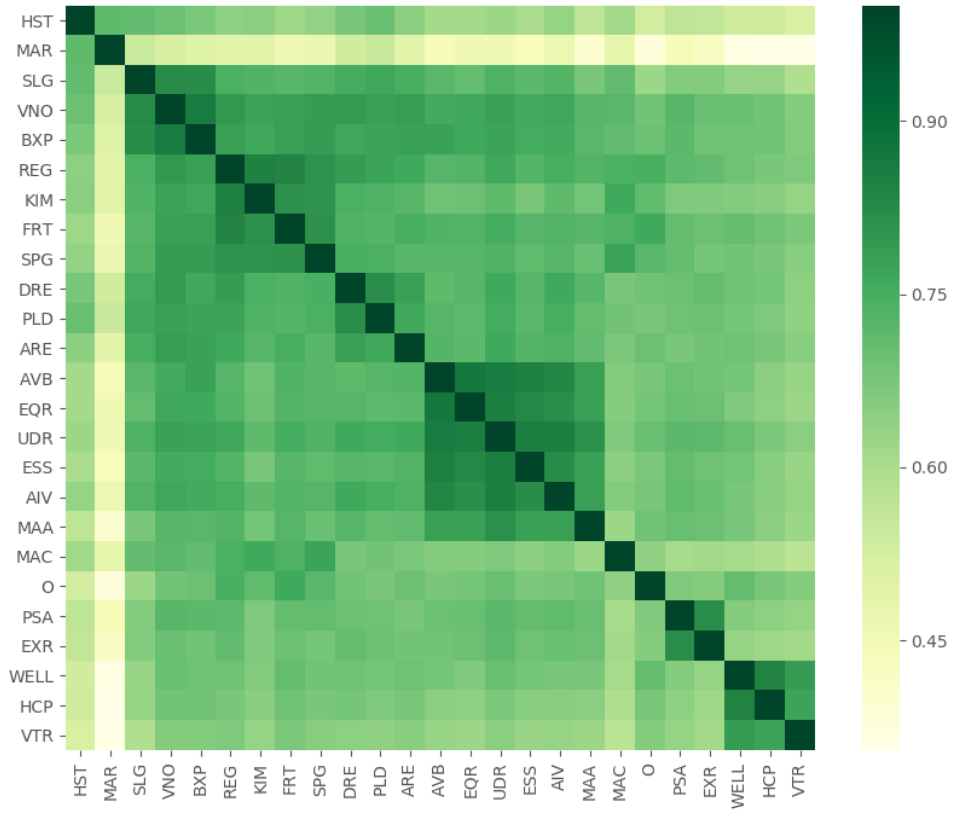


Figura 69 – Cluster destacado em azul na Figura 67.

Nesta figura vemos que a ação com tick MAR foi mal selecionado apresentando baixa

correlação com os demais. Os setores destas ações são:

Tick	Ação	Setor
HST	Host Hotels & Resorts	**Real Estate**
MAR	Marriott Int'l.	Consumer Discretionary
SLG	SL Green Realty	**Real Estate**
VNO	Vornado Realty Trust	**Real Estate**
BXP	Boston Properties	**Real Estate**
REG	Regency Centers Corporation	**Real Estate**
KIM	Kimco Realty	**Real Estate**
FRT	Federal Realty Investment Trust	**Real Estate**
SPG	Simon Property Group Inc	**Real Estate**
DRE	Duke Realty Corp	**Real Estate**
PLD	Prologis	**Real Estate**
ARE	Alexandria Real Estate Equities	**Real Estate**
AVB	AvalonBay Communities, Inc.	**Real Estate**
EQR	Equity Residential	**Real Estate**
UDR	UDR Inc	**Real Estate**
ESS	Essex Property Trust, Inc.	**Real Estate**
AIV	Apartment Investment & Management	**Real Estate**
MAA	Mid-America Apartments	**Real Estate**
MAC	Macerich	**Real Estate**
O	Realty Income Corporation	**Real Estate**
PSA	Public Storage	**Real Estate**
EXR	Extra Space Storage	**Real Estate**
WELL	Welltower Inc.	**Real Estate**
HCP	HCP Inc.	**Real Estate**
VTR	Ventas Inc	**Real Estate**

Tabela 3 - Ações do cluster destacado em azul na Figura xx.

A única ação que não pertence ao mesmo setor das demais no cluster destacado é o segundo como já previamos. Uma maneira de visualizar os clusters em uma matriz de correlação é através do dendograma. O comando *clustermap* do pacote Seaborn [146] para Python utiliza

um diferente algoritmo de minimização da variância chamado Ward [147] que organiza a matriz e gera o dendograma:

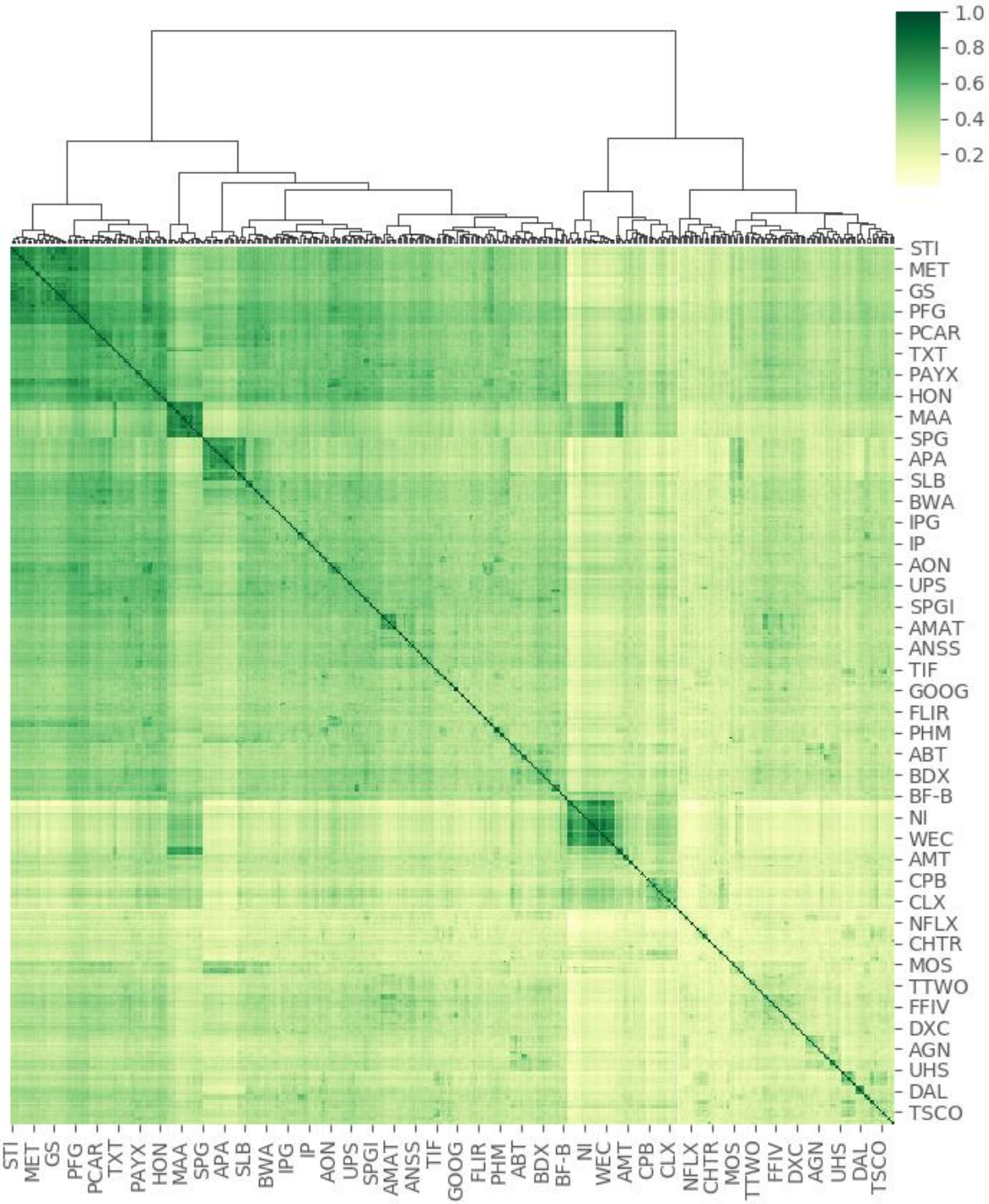


Figura 70 – Matriz de correlação e dendograma gerado a partir da matriz de correlação entre ativos no comando *clustermap* do pacote *seaborn* para *Python*.

5.2.2 Mercado de derivativos

Os derivativos são derivados e dependem de um ou mais ativos diferentes, chamados de *ativo objeto*, exemplos destes ativos financeiros são taxas de juros, moedas, ações e índices, ou ativos não financeiros constituídos por *commodities*. Existem diferentes agente que entram neste mercado: O *hedger* para se proteger contra uma variação indesejada no preço à vista de um determinado ativo no futuro. Este agente aceita até receber menos do que o esperado desde que ele reduza o risco. Um exemplo é uma empresa que utiliza produtos agrícolas e acerta um contrato para o fornecimento da mercadoria a um preço prefixado de forma a se proteger de uma provável alta nos preços, ou do próprio agricultor que também deseja eliminar o risco, mas neste caso, de uma possível queda nos preços. Um outro exemplo é de um exportador que quer garantir que a taxa de câmbio esteja mais previsível e em contrapartida temos o produtor nacional que utiliza insumos importados, querendo se proteger de elevações nos custos provenientes de uma desvalorização cambial. Operações de hedging são como estratégias de administração de riscos de ativos possuídos no presente ou que venham a ser possuídos no futuro, factíveis de serem executadas nos mercados futuros [123].

O *especulador*, ao contrário do *hedger*, aceita o risco de uma alta volatilidade (onde a volatilidade está associada a amplitude de oscilação do preço de um ativo) nos preços na expectativa de ganhos financeiros e seu lucro dependerá de sua capacidade de predizer os preços corretamente, levando em consideração o fluxo de ordens no mercado, através de estratégias estatísticas ou mesmo através de uma análise fundamentalista. Este agente assume o risco indesejado pelos *hedgers* e dá liquidez aos mercados. Uma liquidez alta expressa a habilidade de converter o ativo em dinheiro rapidamente pois tem-se muitos compradores e vendedores dispostos a negociar. O especulador geralmente é quem faz o mercado se movimentar, especialmente os chamados *big players*, que são grandes investidores e especuladores que precisam fracionar sua entrada no mercado para não haver um crescimento agressivo do preço devido ao grande número de ordens realizada e da oferta e demanda. O contrário também se faz necessário, ao desfazer uma posição (vender seus ativos) ele precisa ir vendendo aos poucos para não causar uma grande queda do preço do ativo. Uma analogia interessante criada pelos próprios especuladores é que a lógica dos pequenos especuladores deveria ser sempre identificar os *big players* e atuar junto com eles como nas relações de comensalismo das pequenas rêmoras que nadam do lado dos tubarões e não a lógica de se tornarem as sardinhas que irão ser consumidas por eles.

Uma operação de arbitragem é uma operação em que o investidor não entra com

capital próprio, não assume nenhum risco, e tem lucro no final. Como o investidor não entra com capital próprio, não há limite para o quanto pode investir e no limite, os arbitradores poderiam capturar toda a renda mundial. Por isso, os modelos econômicos usam como axioma que mercado não permite operações de arbitragem. Entretanto, a oportunidade de arbitragem depende de diferenças de preços entre ativos, e esses preços flutuam. Se a flutuação não for simultânea pode-se criar uma oportunidade de arbitragem até o preço do outro ativo flutuar fechando a oportunidade. Sempre que um ativo for negociado com preços discrepantes em mercados ou bolsas diferentes, a própria atuação dos arbitradores atua para deixar os preços em equilíbrio [125].

Nesse aspecto o equilíbrio sem arbitragem é semelhante ao equilíbrio termodinâmico, que é destruído assim que é atingido para ser reestabelecido no momento seguinte, dando origem às flutuações termodinâmicas. Essas oportunidades de arbitragem no atual mundo conectado na velocidade da luz se abrem e se fecham em tempos tão curtos que os profissionais da arbitragem atuam no mercado através de robôs HFT's (High Frequency Trading), que operam em milissegundos, comprando e vendendo ativos em diferentes bolsas em todo o mundo, como a NYSE, Nasdaq, AMEX e várias outras [136]. Operar em bolsas bem estabelecidas leva o risco de que a contraparte não honre seu compromisso a praticamente zero, pois as bolsas exigem depósito de margens de segurança e podem renegociar as posições dos devedores em tempos muito curtos. Mesmo assim o risco dos arbitradores não é nulo, por conta do atraso no tempo entre a decisão de fechar a operação e o fechamento real da mesma. Nesse tempo os preços podem ter mudado e uma operação que geraria lucro pode dar prejuízo. Esses operadores aceitam esse risco pequeno, equilibram o mercado rapidamente e dão liquidez ao mesmo [130].

O mercado de derivativos se subdivide em mercado de futuros, mercado a termo, mercado de opções e mercado de *swaps*, iremos focalizar apenas no mercado de opções.

5.2.3 Introdução ao Mercado de Opções

Analisaremos operações com opções do ponto de vista do investidor, o qual tem um custo inicial para obter lucro no final do período. Iremos convencionar que no momento da compra da opção, no tempo $t = 0$, quando gastarmos dinheiro iremos colocar um sinal positivo e se recebermos dinheiro colocaremos um sinal negativo. Nos tempos posteriores e na maturidade ($t = T$) iremos inverter, saídas com sinal negativo e entradas com sinal positivo.

O preço à vista, ou *spot price*, do ativo objeto (que chamaremos pela sigla A-O) em $t = 0$ será denominado por S. No tempo $t = T$, chamaremos de S_T. No intervalo entre o momento inicial e a maturidade a notação utilizada será S_t. O preço de exercício (*strike price*) é o valor estipulado pelo contrato para o A-O em $t = T$, será denominado por X. Neste mercado nós temos:

- Lançador: é o agente vendedor da opção e sempre tem a obrigação de cumprir com o contrato;
- Titular: é o comprador da opção que tem o direito de exercer a opção de acordo com o contrato, mas não tem a obrigação.

Existem dois tipos principais de opções:

- CALL: É um contrato que dá ao titular o direito de comprar o A-O do lançador pelo *strike price* em T;
- PUT: Contrato que dá ao titular o direito de vender o A-O para o lançador pelo *strike price* em T.

Uma outra característica importante das opções é distiguir se ela é do tipo:

- Americana: Te dá o direito exercer a sua CALL (PUT) para comprar (vender) o A-O a qualquer momento anterior a maturidade, ou seja, em $t \leq T$;
- Europeia: Poderá ser exercida apenas no vencimento ou maturidade em $t = T$.

No Brasil, no mercado de opções, as CALLs americanas concentram maior liquidez.

5.2.4 CALLs de ativos que não pagam dividendos

Iremos analisar inicialmente a seguinte inequação:

$C \leq c \leq \max\left[S - \frac{X}{(1+R)^T}, 0 \right]$ onde C (c) é o prêmio da CALL americana (europeia). Supondo que $S - \frac{X}{(1+R)^T} > 0$, vamos operar na seguinte forma: em $t = 0$ vendemos o A-O por S, compramos uma CALL por c e aplicamos $\frac{X}{(1+R)^T}$ na taxa R.

	$t=0$	$t=T$	$t=T$
Operação	\$	$S_T < X$	$X \le S_T$
Vende x A-Os	$-S\,x$		
Compra x A-Os		$-x\,S_T$	$-x\,S_T$
Compra x CALLs X	cx		$\left(X - S_T\right)x$
Aplica xX / Z^T	$x\left(\frac{X}{Z^T}\right)$	xX	xX
TOTAL	$\left(-S + c + \frac{X}{Z^T}\right)x$	$\left(X - S_T\right)x > 0$	0

Tabela 4 – Operações para diferentes $S_T's$, a coluna \$ refere-se a entrada e saída de capital em

No tempo $t=T$ os ativos foram recompostos e só existem ganhos positivos ou nulos portanto deve haver uma oportunidade de arbitragem no início, concluímos que: $t=0$. Temos:

$$-S + c + \frac{X}{(1+R)^T} > 0 \qquad \text{ou} \qquad c > S - \frac{X}{(1+R)^T}.$$

Daí vale a desigualdade:

$$c \ge \max\left[S - \frac{X}{(1+R)^T}, 0\right].$$

Caso o A-O então não pague dividendos, não há vantagem em exercer a CALL americana antecipadamente para receber o A-O, logo, o preço das CALLs europeias e americanas devem ser iguais, $C = c$. Para um $t < T$, poderemos exercer a CALL americana se $S_t - X > 0$. O prêmio de uma CALL com maturidade T em um tempo t será maior do que

$$c_t \ge S_t - \frac{X}{(1+R)^{(T-t)}},$$

mas

$$\frac{X}{(1+R)^{(T-t)}} < X \qquad \text{logo} \qquad c_t \ge S_t - \frac{X}{(1+R)^{(T-t)}} \ge S_t - X$$ e é preferível manter a opção. Isso é verdade para a CALL mas não é verdade para a PUT. Por isso podemos comparar o prêmio da Call americana com o da europeia, para a qual foi desenvolvida o modelo de Black & Scholes.

5.2.5 Modelo de Cox-Ross-Rubinstein [CRR]

Este modelo será o principal foco deste trabalho, iremos introduzi-lo nesta seção e veremos que ele é bastante simples, mas ele converge para a equação de Black & Scholes e pode ser usado para precificar opções europeias e americanas, CALLs e PUTs e o modelo de Black & Scholes que só se aplica a CALLs europeias [142].

5.2.5.1 Portfólio Replicante usando CRR

Vamos supor que uma ação custa S e só tem a possibilidade de ir para dois preços diferentes no próximo período, denominados estados, um preço no estado S_U ou um preço no estado S_D onde $S_U > S_D$. No modelo CRR os preços são dados por dois fatores multiplicativos U e D:

$$\begin{cases} S_U = US \\ S_D = DS \end{cases}$$

Agora vamos supor que existe uma opção que pode dar um lucro L_U e L_D no próximo período e queremos replicar este derivativo comprando q_S ações e aplicando q_B em títulos com valor B e um retorno R. Usaremos o log-retorno com $e^r = (1+R)$ e teremos as seguintes possibilidades:

$$\begin{cases} q_S US + Be^r q_B = L_U; \\ q_S DS + Be^r q_B = L_D. \end{cases}$$

O nosso portfólio custou em $t = 0$ a quantia

$p_{rep} = q_S S + q_B B.$ Resolvendo o sistema de equações acima obtemos a expressão para o q:

$q_S = \dfrac{L_U - L_D}{S_U - S_D} = \dfrac{\Delta L}{\Delta S}$ e substituindo o q_S em qualquer uma das duas equações do sistema obtemos a expressão para q_B:

$$q_B = \left(\frac{L_D U - L_U D}{B[U - D]} \right) e^{-r}.$$

Estas duas expressões para q_S e q_B são chamadas de risco neutras pois é um hedge que nos mostra o quanto devemos comprar de ações e de títulos de forma a eliminarmos o risco. Os preços dos portfólios replicantes devem ser iguais aos preços das opções para não haver

oportunidade de arbitragem. Substituindo o q_S e o q_B na expressão para o preço do portfólio temos:

$$p_{rep} = \left(\frac{1 - De^{-r}}{[U - D]}\right) L_U + \left(\frac{Ue^{-r} - 1}{[U - D]}\right) L_D$$

Suporemos agora que as probabilidades do preço ir para S_U é π_U e de ir para S_D é π_D portanto $\pi_U + \pi_D = 1$ e a esperança de lucro é a soma do lucro multiplicado pela probabilidade de ambos os casos: $E[L] = \pi_U L_U + \pi_D L_D$. Se o lançador cobrou o valor equivalente ao portfólio replicante p_{rep} pela opção, ele pode aplicá-la com um retorno R e terá $e^r p_{rep}$ no próximo período que deve ser a esperança do lucro da opção para que não haja arbitragem, teremos a seguinte expressão para o *prêmio justo*:

$$p_{rep} = \pi_U e^{-r} L_U + \pi_D e^{-r} L_D$$

Comparando a expressão do prêmio justo com a o portfolio replicante identificamos os valores de π_U e π_D por inspeção:

$$\begin{cases} \pi_U = \dfrac{e^r}{(U - D)}\left[1 - e^{-r} D\right] = \dfrac{1}{(U - D)}\left[e^r - D\right] \\ \pi_D = \dfrac{e^r}{(U - D)}\left[e^{-r} U - 1\right] = \dfrac{1}{(U - D)}\left[U - e^r\right]. \end{cases}$$

Somando π_U e π_D temos

$$\pi_U + \pi_D = \frac{1}{(U - D)}(\cancel{e^r} - D + U - \cancel{e^r}) = 1$$

Devemos levar em conta a condição $S_D < e^r S < S_U$ que garante que o mercado de ações está em equilíbrio, pois se $e^r S > S_U$ bastaria aplicarmos S em títulos e não se compraria ações e no caso onde $S_D > e^r S$ não se aplicaria em títulos pois até o menor valor de retorno de uma ação iria ser superior a este título.

Estas probabilidades risco neutras π_U e π_D, devido ao hedging perfeito, não são necessariamente as probabilidades reais do mercado p e q, estas probabilidades impedem que ocorram operações de arbitragem e são denominadas de *Martingales* que, resumidamente, significa que os passos não guardam memória dos passos anteriores [127,128].

5.2.5.2 CRR para um período

Para um período, os *strike prices* deverão estar no intervalo $DS \leq X \leq US$ e os lucros do titular da CALL em ambos os casos serão:

$L_U = \max(US - X, 0) = US - X$;
$L_D = \max(DS - X, 0) = 0$

desta forma os preços das componentes do portfólio replicante com as expressões para os lucros serão dados por:

$$q_S = \frac{US - X}{S(U - D)} \qquad \text{e} \qquad q_B = -\frac{D(US - X)e^{-r}}{(U - D)B}$$

assim o portfólio replicante é

$$p_{rep} = \frac{US - X}{(U - D)} - \frac{e^{-r}D(US - X)}{(U - D)}$$

e este será o preço para o prêmio justo $p_{rep} = c$, com um pouco mais de álgebra chegamos na forma

$$c = \left(\frac{US - X}{U - D}\right)(1 - De^{-r}).$$

No caso da PUT, os lucros serão dados por:

$L_U = \max(X - US, 0) = 0$
$L_D = \max(X - DS, 0) = X - DS$

assim, o q_S e q_B serão:

$$q_S = -\frac{X - DS}{S(U - D)} \qquad \text{e} \qquad q_B = \frac{U(X - DS)e^{-r}}{B(U - D)}$$

e o preço do portfólio replicante fica:

$p_{rep} = -\frac{X - DS}{U - D} + \frac{U(X - DS)e^{-r}}{U - D}$. Lembrando que este é o preço do prêmio justo, $p_{rep} = p$, com um pouco de álgebra temos:

$$p = \left(\frac{X - DS}{U - D}\right)(Ue^{-r} - 1).$$

Assim, obtemos os prêmios da CALL e PUT para um período utilizando o modelo CRR.

5.2.5.3 Opções européias em n períodos

Sabemos que em $t=0$ o titular da opção paga o prêmio, se o lançador aplica o valor do prêmio a uma taxa R, em $t=n$ ele terá um valor que deve ser a esperança de lucro, se o prêmio é justo, dado por:

$ce^{rn} = E[L_{call}]$; e $pe^{rn} = E[L_{put}]$.

Como eliminamos o risco, utilizaremos as probabilidades risco-neutras. Após n períodos, teremos uma distribuição binomial com $n-k$ estados *up* e k estados *down*:

$P(n,k) = \binom{n}{k} \pi_D^k \pi_U^{n-k}$ e o preço da ação foi para

$S_n = D^k U^{n-k} S$. Os lucros do titular para uma CALL e PUT são dados pela S_n:

$L_{call} = \max(D^k U^{n-k} S - X, 0)$

$L_{put} = \max(X - D^k U^{n-k} S, 0)$.

Utilizando as expressões para o prêmio aplicado a uma taxa *R*, a expressão da expansão binomial e os lucros acima:

$$pe^{rn} = \sum_{k=0}^{n} \frac{n!}{k!(n-k)!} \pi_D^k \pi_U^{n-k} \max(X - D^k U^{n-k} S, 0);$$

$$ce^{rn} = \sum_{k=0}^{n} \frac{n!}{k!(n-k)!} \pi_D^k \pi_U^{n-k} \max(D^k U^{n-k} S - X, 0).$$

Podemos definir um k_{cut} onde

$X = D^{k_{cut}} U^{n-k_{cut}} S$ assim as expressões acima podem ser reescritas redefinindo os limites do somatório e eliminando a função max:

$$pe^{rn} = \sum_{k=0}^{k_{cut}} \frac{n!}{k!(n-k)!} \pi_D^k \pi_U^{n-k} (X - D^k U^{n-k} S);$$

$$ce^{rn} = \sum_{k=k_{cut}+1}^{n} \frac{n!}{k!(n-k)!} \pi_D^k \pi_U^{n-k} (D^k U^{n-k} S - X).$$

Vamos separar os termos com S e os de X e e isolar os prêmios:

$$p = Xe^{-rn} \sum_{k=0}^{k_{cut}} \binom{n}{k} \pi_D^k \pi_U^{n-k} - Se^{-rn} \sum_{k=0}^{k_{cut}} \binom{n}{k} (D\pi_D)^k (U\pi_U)^{n-k};$$

$$c = Se^{-rn} \sum_{k=k_{cut}+1}^{n} \binom{n}{k} (D\pi_D)^k (U\pi_U)^{n-k} - Xe^{-rn} \sum_{k=k_{cut}+1}^{n} \binom{n}{k} \pi_D^k \pi_U^{n-k}.$$

Utilizando as equações das probabilidades risco-neutras π_U e π_D, vamos analisar os seguintes

termos que definiremos como novas probabilidades π_U^* e π_D^*:

$$\pi_U^* = e^{-r}\pi_U U = \frac{U}{U-D}(1-De^{-r})$$
$$\pi_D^* = e^{-r}\pi_D D = \frac{D}{U-D}(Ue^{-r}-1)$$

vemos que as suas somas dão 1:

$$\pi_U^* + \pi_D^* = \frac{U}{U-D}(1-De^{-r}) + \frac{D}{U-D}(Ue^{-r}-1) = \frac{U}{U-D} - \frac{D}{U-D} = 1.$$

Reescrevendo os prêmios *p* e *c* em termos das novas probabilidades (onde dividimos $e^{-rn} = e^{-r(n-k)}e^{-rk}$) temos:

$$p = Xe^{-rn}\sum_{k=0}^{k_{cut}}\binom{n}{k}\pi_D^k\pi_U^{n-k} - S\sum_{k=0}^{k_{cut}}\binom{n}{k}\pi_D^{*k}\pi_U^{*n-k};$$
$$c = S\sum_{k=k_{cut}+1}^{n}\binom{n}{k}\pi_D^{*k}\pi_U^{*n-k} - Xe^{-rn}\sum_{k=k_{cut}+1}^{n}\binom{n}{k}\pi_D^k\pi_U^{n-k}.$$

Uma expressão para este k_{cut} pode ser encontrado com simples álgebra, lembrando da $X = D^{k_{cut}}U^{n-k_{cut}}S$, podemos isolar os termos com k_{cut}. Teremos $\left(\frac{U}{D}\right)^{k_{cut}} = \frac{X}{S}D^{-n}$, aplicando o logaritmo, podemos isolar o k_{cut} e tomar sua parte inteira para que entre no somatório sem problemas:

$$k_{cut} = int\left(\frac{\ln\left[\frac{X}{S}\right] - n\ln[D]}{\ln\left[\frac{U}{D}\right]}\right).$$

Estamos em condições de aplicar o modelo CRR e precificar as opções europeias, para isto devemos calcular as probabilidades risco-neutras em *n* períodos utilizando o binomial $P(n,k)$ e criarmos a seguinte tabela onde cada passo *up* é representado como uma continuação na linha horizontal e cada passo *down* o faz descer uma linha criando uma tabela estilo escada [123].

Aqui exemplificaremos com um caso simples e ilustrativo com um $U = 1.10$ e um $D = 0.90$ para uma ação que custa inicialmente $S = 25$ e um *strike price* de $X = 28$ subdivido em $n = 10$ períodos onde a taxa é $r = 0.005$ ao longo dos 10 períodos, lembrando que a taxa em cada período é sempre dividida pelo número de subdivisões *n*.

	0	1	2	3	4	5	6	7	8	9	10
0	1	0.5251	0.2757	0.1448	0.076	0.0399	0.021	0.011	0.0058	0.003	0.0016
1		0.4749	0.4987	0.3928	0.275	0.1805	0.1137	0.0697	0.0418	0.0247	0.0144
2			0.2256	0.3553	0.3731	0.3265	0.2572	0.189	0.1323	0.0893	0.0586
3				0.1071	0.225	0.2953	0.3102	0.285	0.2394	0.1886	0.1414
4					0.0509	0.1336	0.2104	0.2578	0.2707	0.2558	0.2239
5						0.0242	0.0761	0.1399	0.1959	0.2314	0.243
6							0.0115	0.0422	0.0886	0.1396	0.1832
7								0.0055	0.0229	0.0541	0.0947
8									0.0026	0.0122	0.0321
9										0.0012	0.0065
10											0.0006

Tabela 5 – Evolução das probabilidades risco-neutras.

Criamos também uma tabela com a evolução do preço da ação utilizando a fórmula $S_n = D^k U^{n-k} S$:

	0	1	2	3	4	5	6	7	8	9	10
0	25	27.5	30.25	33.28	36.6	40.26	44.29	48.72	53.59	58.95	64.84
1		22.5	24.75	27.23	29.95	32.94	36.24	39.86	43.85	48.23	53.05
2			20.25	22.28	24.5	26.95	29.65	32.61	35.87	39.46	43.41
3				18.23	20.05	22.05	24.26	26.68	29.35	32.29	35.52
4					16.4	18.04	19.85	21.83	24.01	26.42	29.06
5						14.76	16.24	17.86	19.65	21.61	23.77
6							13.29	14.61	16.08	17.68	19.45
7								11.96	13.15	14.47	15.92
8									10.76	11.84	13.02
9										9.69	10.65
10											8.72

Tabela 6 – Evolução dos preços de uma ação.

A partir da Tabela com a evolução dos preços da ação podemos montar uma Tabela com os possíveis lucros de uma CALL europeia, lembrando que seu lucro deve ser $L = \max(S_t - X, 0)$. Comparamos então qual seria o lucro em diferentes períodos n para um

mesmo X. Podemos calcular o prêmio c para cada período n. Tratando cada coluna como um vetor, podemos fazer um produto escalar para somar cada lucro multiplicado por sua probabilidade obtendo a esperança de lucro naquele período, após isso dividimos por e^{rn} (correspondendo ao valor que deveríamos aplicar a uma taxa r para obter este valor n períodos depois) para retornarmos ao período inicial, n períodos antes. Para que não haja oportunidade de arbitragem, devemos cobrar este valor como prêmio para que seja justo, formulamos na forma: $c_n = (P_n \bullet L_n)e^{-rn}$.

Exemplificando através da Tabela com os lucros, teremos:

	0	1	2	3	4	5	6	7	8	9	10
0	0	0	2.25	5.275	8.6025	12.2628	16.289	20.7179	25.5897	30.9487	36.8436
1		0	0	0	1.9475	4.94225	8.23648	11.8601	15.8461	20.2307	25.0538
2			0	0	0	0	1.64803	4.61283	7.87411	11.4615	15.4077
3				0	0	0	0	0	1.35154	4.2867	7.51537
4					0	0	0	0	0	0	1.05803
5						0	0	0	0	0	0
6							0	0	0	0	0
7								0	0	0	0
8									0	0	0
9										0	0
10											0
Prêmio	0	0	0.614132	0.752214	1.16584	1.34729	1.6515	1.8599	2.09058	2.31898	2.49498

Tabela 7 – Evolução do lucro para uma CALL europeia e os prêmios para cada período na última linha.

A Tabela 7 acima pode ser interpretada como a precificação da CALL europeia em diferentes valores de $T = n$. Vemos que para $n = 0$ e $n = 1$ não há preços pois a esperança de lucro é nula. Podemos realizar o mesmo procedimento anterior para as PUTs, apenas trocando a expressão para o lucro que é $L = \max(X - S_t, 0)$ e calculando os prêmios da mesma forma $p_n = (P_n \bullet L_n)e^{-rn}$:

	0	1	2	3	4	5	6	7	8	9	10
0	3	0.5	0	0	0	0	0	0	0	0	0
1		5.5	3.25	0.775	0	0	0	0	0	0	0
2			7.75	5.725	3.4975	1.04725	0	0	0	0	0
3				9.775	7.9525	5.94775	3.74252	1.31678	0	0	0
4					11.5975	9.95725	8.15297	6.16827	3.9851	1.58361	0
5						13.2377	11.7615	10.1377	8.35145	6.38659	4.22525
6							14.714	13.3854	11.9239	10.3163	8.54793
7								16.0426	14.8468	13.5315	12.0847
8									17.2383	16.1622	14.9784
9										18.3145	17.3459
10											19.283
Prêmio	3	2.86035	3.33553	3.33535	3.61141	3.65597	3.82398	3.89685	3.99268	4.08691	4.12941

Tabela 8 – Evolução do lucro para uma PUT europeia e os prêmios para cada período na última linha.

A Tabela 8 pode ser interpretada da mesma forma que no caso das CALLs na precificação. O caso $n=0$ pode ser interpretado como uma PUT que será exercida naquele momento. Um possível titular deseja comprar a PUT naquele momento pelo *strike price* de 28 e vender no mercado a termo por 25 lucrando 3 na operação de arbitragem, por isto o preço desta PUT que será exercida agora é 3.

5.2.5.4 Opções americanas em n períodos

Sabemos que não há distinção de lucro entre as opções americanas e europeias na maturidade, portanto no $t=n$ todos os lucros destas são iguais, porém ao retroagirmos comparando os valores de L em cada tempo até retornarmos ao tempo inicial conseguiremos calcular os prêmios para as opções americanas, CALLs e PUTs. A esperança de lucro no próximo período (já calculados no modelo europeu) é denotada por L_U para o caso *up* e L_D para o caso *down*. Assim o lucro em $t=k-1$ é dado por $L_{k-1}=\max\left[e^{-r}\left(\pi_U L_U\left(t=k\right)+\pi_D L_D\left(t=k\right)\right), X-S_{k-1}, 0\right]$ para um período k qualquer de uma PUT. O titular sempre compara entre exercer a opção e receber o lucro $X-S_{k-1}$ (para a CALL é $S_{k-1}-X$) ou não exercer na expectativa que o preço seguinte seja $e^{-r}\left(\pi_U L_U\left(t=k\right)+\pi_D L_D\left(t=k\right)\right)$ naquele momento. Obtemos então as tabelas para os lucros e

prêmios das CALLs e PUTs através das esperanças dos lucros em $t = n$ das opções europeiras e calculando as esperanças dos lucros em $t = n-1$ a $t = 0$.

	0	1	2	3	4	5	6	7	8	9	10
0	2.495	3.628	5.174	7.221	9.837	13.054	16.843	21.135	25.868	31.088	36.844
1		1.269	1.957	2.966	4.404	6.385	9.002	12.277	16.125	20.37	25.054
2			0.522	0.862	1.407	2.26	3.561	5.476	8.153	11.601	15.408
3				0.151	0.27	0.479	0.845	1.481	2.574	4.426	7.515
4					0.022	0.041	0.079	0.151	0.289	0.553	1.058
5						0	0	0	0	0	0
6							0	0	0	0	0
7								0	0	0	0
8									0	0	0
9										0	0
10											0
Prêmio	2.495										

Tabela 9 – Evolução do lucro para uma CALL americana e os prêmios para cada período na última linha.

	0	1	2	3	4	5	6	7	8	9	10
0	4.3853	3.0333	1.8863	1.0027	0.4106	0.0996	0	0	0	0	0
1		5.9263	4.3333	2.8831	1.6678	0.7589	0.2107	0	0	0	0
2			7.75	5.9822	4.2571	2.6903	1.3729	0.4459	0	0	0
3				9.775	7.9525	6.0341	4.1752	2.4122	0.9436	0	0
4					11.5975	9.9572	8.153	6.1683	4.0613	1.9967	0
5						13.2377	11.7615	10.1377	8.3514	6.3866	4.2252
6							14.714	13.3854	11.9239	10.3163	8.5479
7								16.0426	14.8468	13.5315	12.0847
8									17.2383	16.1622	14.9784
9										18.3145	17.3459
10											19.283
Prêmio	4.3853										

Tabela 10 – Evolução do lucro para uma PUT americana e os prêmios para cada período na última linha.

Nos casos acima das Tabelas 9 e 10, interpretamos diferentemente das Tabelas 7 e 8. As

tabelas 9 e 10 podem ser interpretadas como uma opção que tem vencimento em $n=10$ podendo ser exercida a qualquer momento. O preço da CALL e da PUT é justamente a esperança de lucro que foi retroagida ao momento presente.

5.3 Modelo de Black & Scholes

Demonstraremos a obtenção da equação de Black & Scholes através da equação da difusão seguindo o trabalho original de 1973 onde é utilizado sucessivas mudanças de variáveis para converter a equação diferencial de Black & Scholes na equação da difusão [141] e também através da convergência do modelo CRR [142].

5.3.1 Movimento Browniano

Vamos definir alguns processos estocásticos importantes que utilizaremos para definir a equação diferencial de Black & Scholes, iniciando com o movimento browniano padrão e desenvolver até o movimento browniano geométrico que descreverá o movimento dos prêmios das opções.

O movimento browniano é um movimento aleatório de partículas em um fluido, o movimento aleatório das partículas é ocasionado por colisões caóticas com moléculas do fluido que estão em vibração e movimentando-se de acordo com a temperatura. O primeiro exemplo observado por Robert Brown foi de grãos de pólen em um recipiente com água [126, 132].

5.3.2 Fórmula de Black & Scholes através da equação da difusão

A formula de Black & Scholes foi originalmente obtida através da equação da difusão, fazendo algumas substituições para adaptá-la. Esta é uma forma interessante por suas semelhanças com inúmeros fenômenos físicos como na estatística de partículas que atravessam uma determinada área ou na equação do calor que nos mostra o fluxo de energia calorífica em uma determinada área [133] em um condutor ou mesmo na relação entre cargas passando em uma determinada área com a corrente produzida por estar [135].

5.3.2.1 Equação da Difusão

Podemos definir um fluxo unidimensional de partículas $J(x)$ como uma quantidade Δn de partículas que estão atravessando uma área ΔA em um intervalo de tempo Δt de forma que teremos uma expressão para a equação da continuidade dado por:

$$\frac{\partial J}{\partial x}+\frac{\partial c}{\partial t}=0.$$

Considerando a lei de Fourier

$J=-D\frac{\partial c}{\partial x}$ que nos mostra que o fluxo sempre acontece de uma zona de maior concentração para uma de menor concentração onde D é chamado de coeficiente de difusão, assim conectando a lei de Fourier com a equação da continuidade obtêm-se a equação da difusão

$\frac{\partial^2 c}{\partial x^2}-\frac{1}{D}\frac{\partial c}{\partial t}=0$ cuja solução é dada por $c(x,t)=\frac{1}{\sqrt{4\pi Dt}}\exp\left\{-\frac{x^2}{4Dt}\right\}$ que é uma gaussiana normalizada (com área unitária) e vemos que o movimento browniano é também solução desta equação, de fato, igualando esta última expressão com a distribuição normal $f(x,t)$, vemos que $2D=\sigma^2$. Fazendo o limite $\lim_{t\to 0}c(x,t)$ esta função tenderá a uma delta de Dirac, daí $c(x,0)=\delta(x)$. No caso em que é dado uma condição inicial $c(x,0)$ podemos resolver a equação através do teorema da convolução e tomando o limite de $t\to 0$, assim a solução desta equação dada uma condição inicial é

$$c(x,t)=\int_{-\infty}^{+\infty}\left(\frac{c(z,0)}{\sqrt{4\pi Dt}}\right)\exp\left\{-\frac{(x-z)^2}{4Dt}\right\}dz$$

5.3.2.2 Equação de Black & Scholes

No nosso caso não trataremos de partículas, mas sim do movimento dos prêmios das opções (CALLs) em relação ao preço das ações e do tempo. Definindo as notações, chamamos D de dinheiro, S é o preço da ação e c é prêmio da CALL. Sob a perspectiva do investidor convencionamos que em $t=0$ o $D>0$ significa que estamos gastando para adquirir nosso portfólio e se $D<0$ estamos tomando emprestado ou recebendo de alguma fonte. Nos tempos seguintes teremos o contrário, $D>0$ representa o recebimento de um certo valor fruto do investimento e $D<0$ significa que tivemos algum prejuízo.

Faremos uma série de considerações para criar este modelo. Supomos que o mercado é perfeito, a taxa da renda fixa tanto para tomar emprestado e para emprestar é dada por

$$r = \frac{1}{D}\frac{dD}{dt} \qquad \rightarrow \qquad dD = rDdt$$

sem haver um limite para obter ou emprestar crédito.

Um outro axioma deste modelo que é bastante debatido quanto a validade do mesmo é que o preço das ações segue uma distribuição browniana geométrica com

$$dS = \mu Sdt + \sigma Sdf$$

onde μ é o *drift* ou rendimento determinístico, o σ é a volatilidade e f é o movimento browniano padrão e também determinamos que o retorno de um portfólio de arbitragem sem riscos é nulo.

Obtemos *dc* expandindo $c(S,t)$ e utilizando o lema de Itô:

$$dc = \left(\frac{\partial c}{\partial t} + \mu S \frac{\partial c}{\partial S} + \frac{\sigma^2 S^2}{2} \frac{\partial^2 c}{\partial S^2} \right) dt + \sigma S \frac{\partial c}{\partial S} df$$

Podemos criar um portfólio p aplicando D em renda fixa e comprando q ações a um preço S cada e vendendo uma opção de compra por c. Teremos $p = D + qS - c$ e o retorno deste portfólio é simplesmente $dp = dD + qdS - dc$, onde ao substituirmos as equações para *dD, dS* e *dc* na expressão de *dp* teremos:

$$dp = rDdt + q(\mu Sdt + \sigma Sdf) - \left[\left(\frac{\partial c}{\partial t} + \mu S \frac{\partial c}{\partial S} + \frac{\sigma^2 S^2}{2} \frac{\partial^2 c}{\partial S^2} \right) dt + \sigma S \frac{\partial c}{\partial S} df \right]$$

que simplificamos na forma:

$$dp = \left[rD + \mu S \left(q - \frac{\partial c}{\partial S} \right) - \frac{\partial c}{\partial t} - \frac{\sigma^2 S^2}{2} \frac{\partial^2 c}{\partial S^2} \right] dt + \sigma S \left[q - \frac{\partial c}{\partial S} \right] df$$

agora vemos que nesta equação temos um termo estocástico que podemos eliminar fazendo um *hedge* dinâmico, comprando um número de ações igual a variação do preço do prêmio em relação ao preço da ação:

$$q = \frac{\partial c}{\partial S} \qquad \rightarrow \qquad q - \frac{\partial c}{\partial S} = 0.$$

Vemos que para eliminar o risco devemos estar sempre atualizando a quantidade de ações que temos de acordo com a expressão acima. Uma consequência direta da condição acima é a eliminação do *drift* e a obtenção de uma expressão mais simples:

$$dp = \left[rD - \frac{\partial c}{\partial t} - \frac{\sigma^2 S^2}{2} \frac{\partial^2 c}{\partial S^2} \right] dt$$

Agora devemos analisar a condição para portfólios de arbitragem sem risco serem nulos. Este portfólio deve ter custo inicial nulo portanto

$$p_{arb} = D + qS - c = 0 \qquad \rightarrow \qquad D = c - S\frac{\partial c}{\partial S}$$

daí o retorno também deve ser nulo:

$$dp_{arb} = \left[rc - \frac{\partial c}{\partial t} - rS\frac{\partial c}{\partial S} - \frac{\sigma^2 S^2}{2}\frac{\partial^2 c}{\partial S^2} \right] dt = 0$$

que nos leva a equação diferencial de Black & Scholes:

$$\frac{\partial c}{\partial t} + rS\frac{\partial c}{\partial S} + \frac{\sigma^2 S^2}{2}\frac{\partial^2 c}{\partial S^2} - rc = 0.$$

Para resolver esta equação devemos levar em conta algumas condições. Se a ação tem sempre valor nulo então o prêmio da opção também deve ser nulo: $c(0,t) = 0$. Para uma opção instantânea, isto é, lançada e exercida em $t = T$ teremos duas possibilidades, se $S_T > X$ o lançador cobra $c = S_T - X$ para não haver prejuízo, mas se $S_T < X$ a opção não custa nada e também não será exercida e isto podemos escrever na forma: $c(S_T, T) = \max[S_T - X, 0]$.

Iremos fazer algumas mudanças de variáveis de modo a transformar a equação de B&S na equação da difusão. Primeiro substituiremos S por u onde $u = \ln(S/X)$ onde X e S tem a mesma dimensão e a fração é adimensional. A derivada em relação ao tempo não muda, porém

$$\frac{\partial c}{\partial S} = \frac{\partial c}{\partial u}\frac{\partial u}{\partial S} = \frac{1}{S}\frac{\partial c}{\partial u}$$

$$\frac{\partial^2 c}{\partial S^2} = \frac{\partial}{\partial S}\left(\frac{1}{S}\frac{\partial c}{\partial u}\right) = -\frac{1}{S^2}\frac{\partial c}{\partial u} + \frac{1}{S^2}\frac{\partial^2 c}{\partial u^2}.$$

Assim a nossa equação ficará na forma:

$$\frac{\partial c}{\partial t} + \left(r - \frac{\sigma^2}{2}\right)\frac{\partial c}{\partial u} + \frac{\sigma^2}{2}\frac{\partial^2 c}{\partial u^2} = rc.$$

Para nos livrarmos do termo *rc* queremos que $\partial_t c = rc$, para isto supomos que $c \propto e^{rt}$ ou $c \propto e^{-r(T-t)}$. Portanto chutamos $c(u,t) = e^{-r(T-t)}y(u,t)$ de forma que a equação em relação a y agora fica na forma:

$$\frac{\partial y}{\partial t} + \left(r - \frac{\sigma^2}{2}\right)\frac{\partial y}{\partial u} + \frac{\sigma^2}{2}\frac{\partial^2 y}{\partial u^2} = 0.$$

Devemos nos livrar das constantes e trocar o sinal da derivada parcial $\partial_t y$, para isto faremos

as seguintes mudanças em t e u (mais detalhes no apêndice D):

$$t' = \frac{\left(r - \frac{\sigma^2}{2}\right)^2}{\frac{\sigma^2}{2}}(T-t) \qquad \text{e} \qquad u' = \frac{\left(r - \frac{\sigma^2}{2}\right)^2}{\frac{\sigma^2}{2}}u$$

de modo que as derivadas assumem a forma:

$$\frac{\partial y}{\partial t} = -\frac{\left(r - \frac{\sigma^2}{2}\right)^2}{\frac{\sigma^2}{2}}\frac{\partial y}{\partial t'} \quad , \quad \frac{\partial y}{\partial u} = \frac{\left(r - \frac{\sigma^2}{2}\right)}{\frac{\sigma^2}{2}}\frac{\partial y}{\partial u'} \quad \text{e} \quad \frac{\partial^2 y}{\partial u^2} = \frac{\left(r - \frac{\sigma^2}{2}\right)^2}{\left(\frac{\sigma^2}{2}\right)^2}\frac{\partial^2 y}{\partial u'^2}$$ e a nossa

equação simplifica para a forma

$$-\frac{\partial y}{\partial t'} + \frac{\partial y}{\partial u'} + \frac{\partial^2 y}{\partial u'^2} = 0.$$

Faremos agora a substituição $z = u' + t$ de modo que $y(z,t') = y(u+t',t')$ para nos livrarmos do termo $\partial_{u'} y$. Assim, vamos checar as derivadas:

$$\frac{\partial y}{\partial z} = \frac{\partial y}{\partial u'} \qquad \text{e} \qquad \frac{\partial^2 y}{\partial z^2} = \frac{\partial^2 y}{\partial u'^2}$$

e a derivada temporal passa de

$\frac{\partial}{\partial t'} y(u',t') \to \frac{\partial}{\partial t'} y(z,t') + \frac{\partial}{\partial z} y(z,t')$ assim teremos:

$$-\frac{\partial y}{\partial t'} + \frac{\partial^2 y}{\partial z^2} = 0 \qquad \text{com} \qquad \text{solução} \qquad y = \frac{\exp\left\{-\frac{(u'+z')^2}{4t'}\right\}}{\sqrt{4\pi t'}} \qquad \text{e} \qquad \text{sabemos} \qquad \text{que}$$

$$y(z,t') = \int_{-\infty}^{+\infty} y(z',0)\frac{\exp\left\{-\frac{(z-z')^2}{4t'}\right\}}{\sqrt{4\pi t'}}dz'$$

resolvendo e desfazendo todas as mudanças obtemos a fórmula de Black & Scholes para o preço de uma CALL europeia [131,138]:

$$c(S,t) = S\Phi(d_+) - e^{-r(T-t)}X\Phi(d_-) \text{ com } d_\pm = \frac{\ln\frac{S}{X} + \left(r \pm \frac{\sigma^2}{2}\right)(T-t)}{\sigma\sqrt{T-t}}$$

5.4 Convergência do modelo CRR com o B&S

Vamos encontrar o desvio σ e o drift μ para uma log-normal (que é o movimento descrito pela equação de B&S através de um processo multiplicativo estocástico) e verificar a convergência do CRR. Para isto vamos analisar um processo estocástico aditivo e um multiplicativo [139]. Quando temos um n muito grande, a distribuição binomial tende à normal:

$$\binom{n}{k} q^{n-k} p^k \to \frac{1}{\sqrt{2\pi npq}} \exp\left\{-\frac{(k-np)^2}{2npq}\right\}.$$

Podemos ter um **processo aditivo:** Em um período n temos um preço para um A-O dado por S_n. No próximo período teremos uma probabilidade p de $S_{n+1} = S_n + \delta_U$ e q para $S_{n+1} = S_n + \delta_D$ e $\delta_U > \delta_D$. Suponha que damos k passos *up* e $n-k$ passos *down* de modo que após n períodos teremos $S_{n,k} + S + n\delta_D + k(\delta_U - \delta_D)$. Mudamos a variável:

$$k = \frac{S_n - S - n\delta_D}{\delta_U - \delta_D} \qquad \leftrightarrow \qquad k - np = \frac{S_n - S - n\bar{\delta}}{\delta_U - \delta_D}$$

com $\bar{\delta} = E[\delta] = p\delta_U + q\delta_D$. Por inspeção concluímos que o movimento browniano é

$$MB(S_n) = \frac{1}{(\delta_U - \delta_D)\sqrt{2\pi npq}} \exp\left\{-\frac{[S_n - S - n\bar{\delta}]^2}{2npq(\delta_U - \delta_D)^2}\right\}$$

No nosso caso temos um **processo multiplicativo**, portanto será um movimento browniano geométrico dado pela log-normal. Temos $S_{n+1} = US_n$ com probabilidade p e q para $S_{n+1} = DS_n$ onde o preço depois de n passos é $S_{n,k} = D^{n-k}U^k S = \left(\frac{U}{D}\right)^k D^n S$. Aplicando o logaritmo obtemos o k e o $k - np$:

$$k = \frac{\ln S_n - \ln S - n\delta_D}{\ln\left(\frac{U}{D}\right)} \qquad \leftrightarrow \qquad k - np = \frac{\ln\left(\frac{S_n}{S}\right) - n(p\ln U + q\ln D)}{\ln\left(\frac{U}{D}\right)}.$$

O movimento browniano geométrico é [129]:

$$MBG(S_n) = \frac{1}{S_n \ln\left(\frac{U}{D}\right)\sqrt{2\pi npq}} \exp\left\{-\frac{\left[\ln\left(\frac{S_n}{S}\right) - n(p\ln U + q\ln D)\right]^2}{2npq\ln^2\left(\frac{U}{D}\right)}\right\}$$

Para um período n e com as probabilidades risco-neutras teremos:

$$MBG(S_n) = \frac{1}{S_n \ln\left(\frac{U}{D}\right)\sqrt{2\pi n \pi_U \pi_D}} \exp\left\{ -\frac{\left[\ln\left(\frac{S_n}{S}\right) - n(\pi_U \ln U + \pi_D \ln D)\right]^2}{2n\pi_U \pi_D \ln^2\left(\frac{U}{D}\right)} \right\}$$

Comparando com a log-normal para um período T de uma variável S_t:

$$logN = \frac{\exp\left\{ -\frac{\left[\ln(S_t/S) - \mu T\right]^2}{2\sigma^2 T} \right\}}{S_t\sqrt{2\pi\sigma^2 T}}$$

vemos que

$$\mu T = n(\pi_U \ln U + \pi_D \ln D) \quad , \quad \sigma^2 T = n\pi_U \pi_D \ln^2\left(\frac{U}{D}\right)$$

Na Figura 71 visualizamos diferentes PDFs obtidas através dos passos multiplicativos do modelo CRR para valores de T distintos:

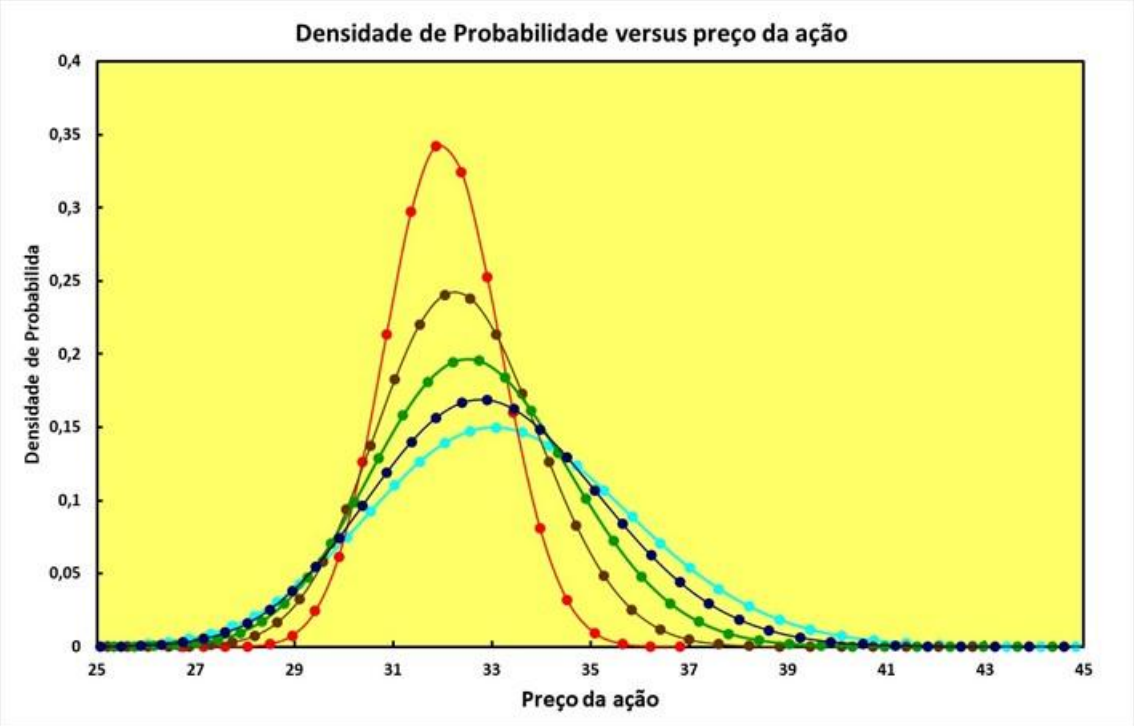


Figura 71 – PDF com passos multiplicativos para valores de *T* distintos. Vermelho: $T = 5$, Marrom: $T = 20$, verde: $T = 40$, azul escuro: $T = 60$, azul claro: $T = 80$.

Assim, utilizando a ação da Petrobrás cuja cotação atual é de $S = 25,33$ em reais para uma opção com strike de $X = 27$ que vencerá um ano depois. Subdividimos em $n = 255$ que é o número de dias úteis de 2019. A taxa de renda fixa SELIC é de $6,5\%$ ao ano, portanto $Z_{ano} = 1,065$. Temos então $Z_{dia} = 1,00025$ por dia. Os dados foram obtidos em: [145]. Os passos deve ser próximos de 1, escolhemos $U = 1,01$ e $D = 0,99$. Calculando o σ e o μ com esses números na expressão acima, plotamos uma linha com a log-normal e em seguida geramos o modelo CRR. Lembrando que $\begin{cases} \pi_U = \dfrac{1}{(U-D)}[Z-D] \\ \pi_D = \dfrac{1}{(U-D)}[U-Z]. \end{cases}$

Os valores obtidos foram $\pi_U = 51,27\%$ e $\pi_D = 48,72\%$ que não obedece a propriedade $\pi_U + \pi_D = 1$ mas redefinindo os termos com o módulo evitaremos este problema:

Agora $\pi_U = 49,467\%$ e $\pi_D = 50,467\%$. Visualizamos as matrizes com as esperanças de lucro, como na seção anterior através de um mapa térmico na Figura 72:

Figura 72 – Matriz com esperanças de lucro da CALL americana ao longo dos períodos.

Na Figura 73 visualizamos o mapa térmico da matriz de esperança de lucro ao longo dos

períodos:

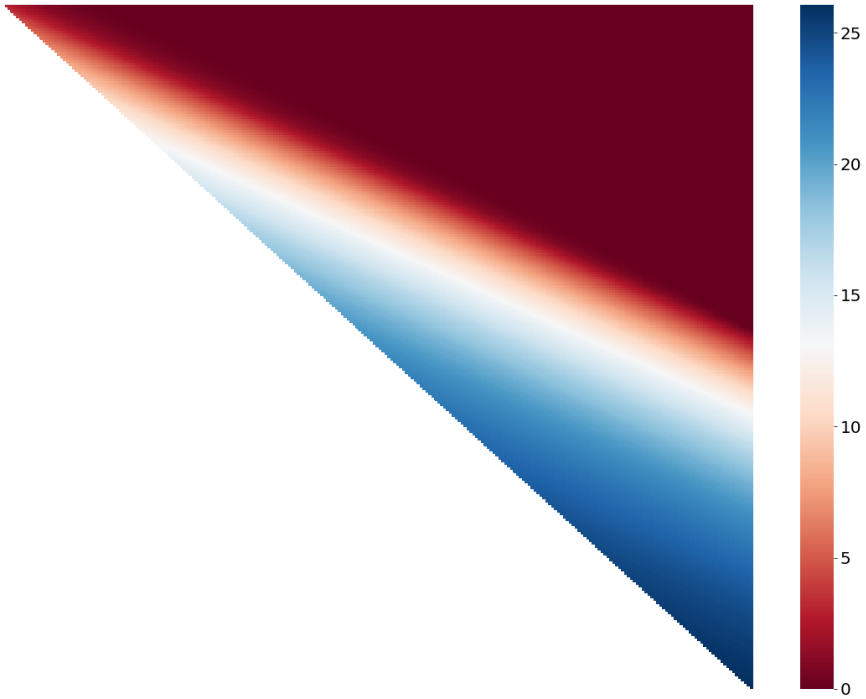


Figura 73 – Matriz com esperanças de lucro da PUT americana ao longo dos períodos.

Gerando uma log-normal com estes dados, comparamos com os pontos com preços *vs* probabilidade gerados no modelo CRR no período final ($T = 255$):

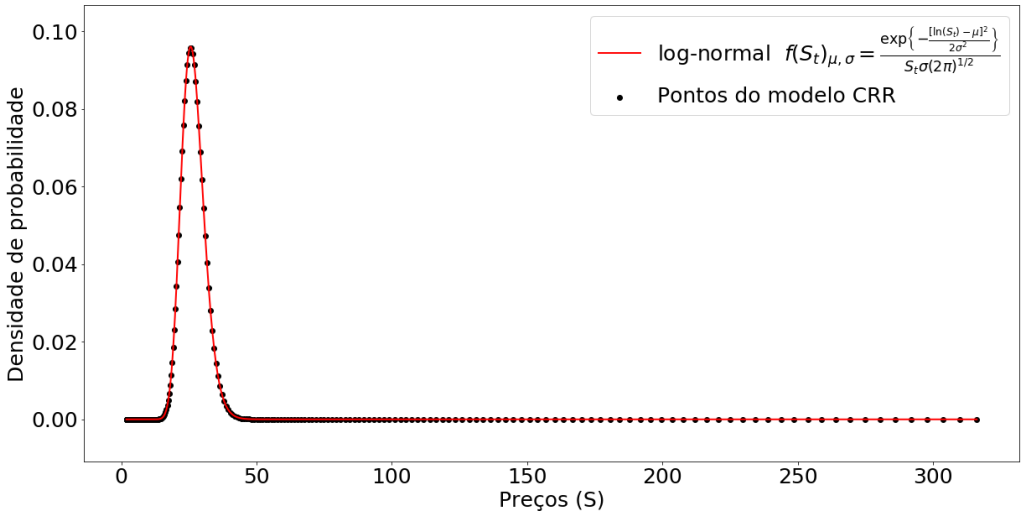


Figura 74 – Log-normal com μ e σ dados pelas expressões de μT e $\sigma^2 T$.

Apresentamos os preços dos prêmios das opções na Tabela 11 aproximados com duas casas decimais:

Opção	Prêmio (R$)
CALL europeia	1,09
CALL americana	1,09
PUT europeia	2,32
PUT americana	3,03

Tabela 11 – Prêmios das opções.

Onde os preços das CALLs americana e europeia são idênticas como já esperavamos que fosse. Há diferença entre os preços das PUTs, sendo a PUT americana mais cara, o que era esperado pois a opção americana dá mais privilégios ao titular do que a europeia. A Figura 75 nos mostra a evolução dos preços ao longo do tempo:

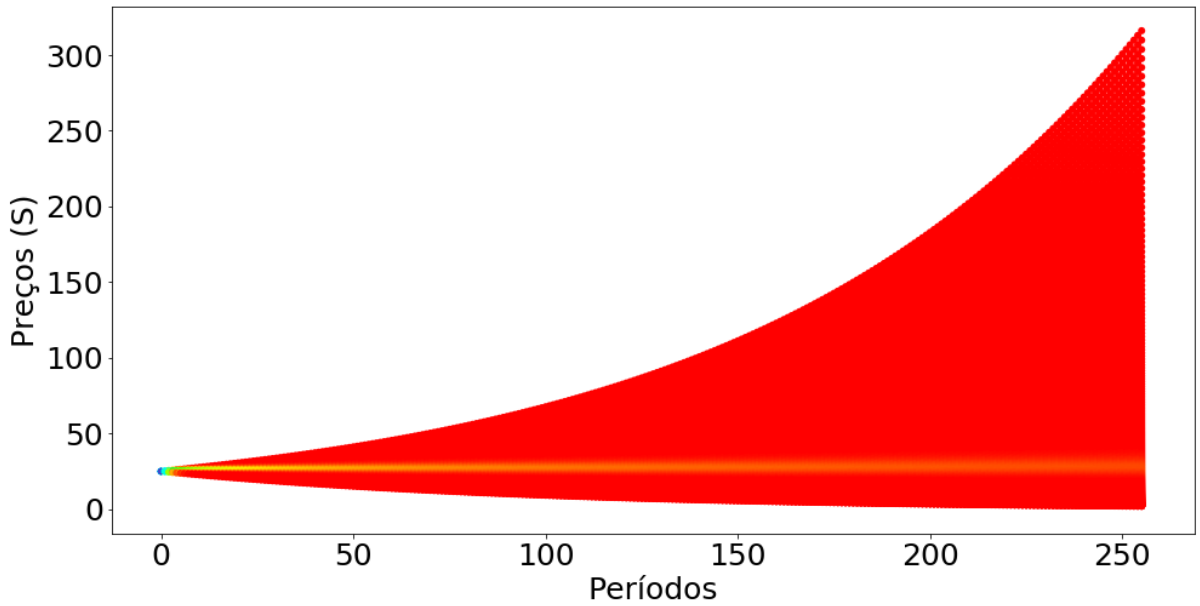


Figura 75 – Evolução dos preços no modelo CRR.

A cor na Figura 75 representa a probabilidade de obtermos tal preço naquele período. Em vermelho representamos uma probabilidade muito baixa, próxima a zero e mais alta em azul. Vemos que as preços os primeiros preços possuem alta probabilidade. Uma linha fina amarela onde os preços se concentram pode ser observada com um pouco de esforço, destacando-a vemos:

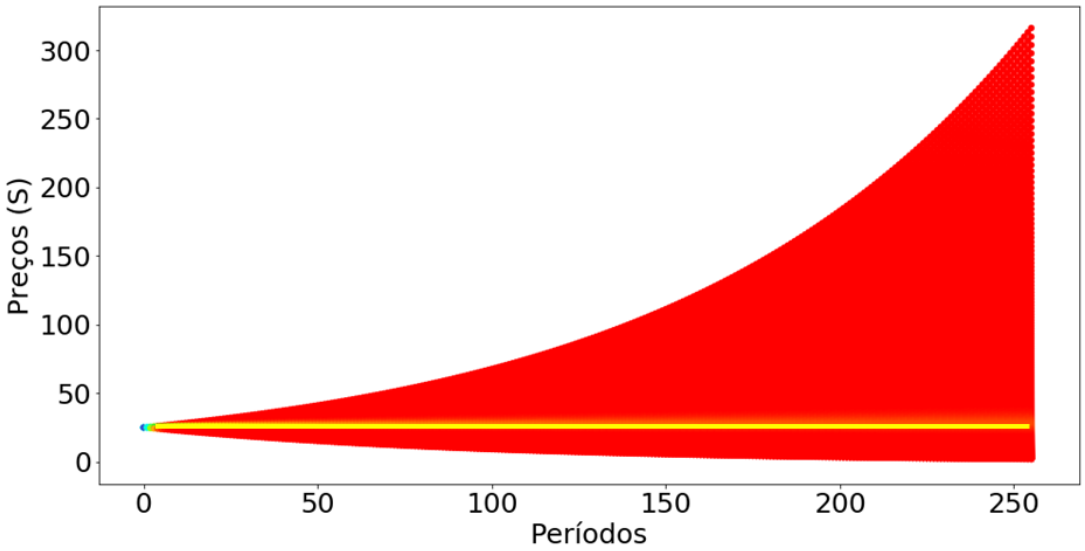


Figura 76 – Evolução dos preços no modelo CRR destacando a região amarela.

6 CONCLUSÃO E PERSPECTIVAS

No capítulo 3 mostramos como obter uma medida de similaridade, ou distância, entre duas amostras independente de uma calibração complicada e demorada, exigindo medidas simultâneas e padrões. Mostramos isso para o caso da fluorescência obtida com PLE de dois fótons. Entretanto, a metodologia é mais geral do que o caso particular apresentado.
A importância desse resultado é que permite medir similaridades entre amostras, ou materiais, ou qualquer conjunto de medidas, independente de transformações lineares. Porque esse fato é importante? Medidas de um aparelho para outro usualmente mudam a escala, ou seja, multiplicam por um fator diferente, e também a linha de base, ou seja, adicionam uma constante. Uma mesma amostra medida no mesmo instrumento com diluições diferentes também implica em uma variação linear entre as medidas. Por isso, a menos que se usem padrões de calibração, muitas vezes complicados, e que usualmente exigem duas medidas realizadas quase simultaneamente, os valores das medidas em si não podem ser comparados diretamente. Uma amostra obtida e caracterizada anos atrás não pode ser comparada com uma amostra recente. Mas o coeficiente de correlação entre as mesmas é imune a essas variações e permite comparar duas amostras sem uma calibração, além de medidas em momentos diferentes, ou até com equipamentos diferentes. A distância de correlação permite a medida de similaridades pareadas em diferentes experimentos. Como exemplo suponha que se mediu a resposta de duas amostras para um conjunto de diferentes frequências de uma onda incidente. Como as intensidades de cada frequência mudam só seria possível extrair a correlação para as respostas da mesma frequência, que será imune às variações das intensidades das ondas incidentes. O único fator importante é que a frequência seja a mesma nas duas amostras. Combinando as correlações obtidas para cada uma das frequências podemos obter uma distância de similaridade entre as amostras.

A análise de MST foi primeiramente aplicada à espectros de fluorescência de óleos crus que, através da distância de correlação de vários comprimentos de onda de excitação de laser pareadas considerando-os como experimentos distintos. O resultado nos mostra que é possível agrupar os diferentes óleos, portanto podemos também classificá-los. Aplicamos o mesmo método de MST em dados de ações do mercado financeiro de forma que obtivemos uma imagem representando a formação de aglomerados de dados com a visualização facilitada através de cores diferentes por setores.

Dentro do mercado financeiro, no capítulo 5, mostramos que o método de precificação de opções através do modelo CRR consegue convergir para o modelo de Black &

Scholes e nos dá o poder de precificar as opções de compra e venda americanas e europeias em vários períodos diferentes. Tais métodos consistem no uso de ferramentas probabilísticas como o movimento browniano geométrico na inferência estatística de dados reais. A inferência de dados foi também útil através da regressão via mínimos quadrados ordinários que foi utilizada em diversos métodos de análise de espectros Raman. Alguns exemplos são a descrição da linha de base dos espectros, suavização e decomposição em componentes e concentração via mínimos quadrados alternados. Os métodos de tratamento utilizados tornam possível a correlação e covariância entre vários espectros.

A classificação de espectros Raman, no capítulo 4, através de CNN mostrou-se eficiente apesar da escassez de dados. O modelo é generalizável de forma a aprender a distinguir entre uma quantidade muito maior de classes (minerais). Para isto é necessário utilizar computação em nuvem para o treino com um conjunto de dados e arquitetura da rede mais robustos. O principal problema deste método é a má classificação de minerais caso o mapa Raman utilizado como input não contenha espectros puros de minerais. No caso onde todos os espectros são mistos, as componentes MCR serão também mistas e o modelo CNN falha em classificar o mineral $A+B$ pois o modelo aprende apenas o que é A ou B. Isto geralmente não deve ser problema para mapas grandes, onde a probabilidade de encontrar os espectros puros será maior.

Como perspectivas para estes trabalhos temos o desenvolvimento de uma nova arquitetura de rede neural capaz de identificar espectros mistos e seus fatores de combinação linear. A princípio todos os métodos utilizados na análise de Raman podem e devem ser também explorados nos espectros de fluorescência. Vamos também estudar algoritmos de DL voltados à análise de séries temporais para aplicação no mercado financeiro [12 , 120 , 13] e buscar formas de aplicações em opções. Devemos investigar a precifiação para opções de ações que pagam dividendos com os modelos de B&S e CRR. Assim devemos saber como introduzir este fator no nosso modelo tendo em vista que ações que pagam dividendos são muito interessantes de se investir [140].

As aplicações dos métodos vão além. A idéia dos algoritmos de ML e DL é antiga porém apenas nos dias atuais onde o custo computacional se reduziu que estes algoritmos se tornaram relevantes. Com os adventos do computador quântico, futuramente estes algoritmos se tornarão cada vez mais presentes no dia a dia em uma era regida por grandes quantidades dados gerados a cada segundo onde a relevância destes algoritmos na tomada de decisão em projetos de risco será cada vez mais forte.

APÊNDICE A – FUNÇÃO DELTA

A função delta especial é obtida através da seqüência $\delta_n\left(x-x_o\right)=A\dfrac{\sin\left[n\left(x-x_o\right)\right]}{\left(x-x_o\right)}$ cuja curva é mostrada na Figura 1.

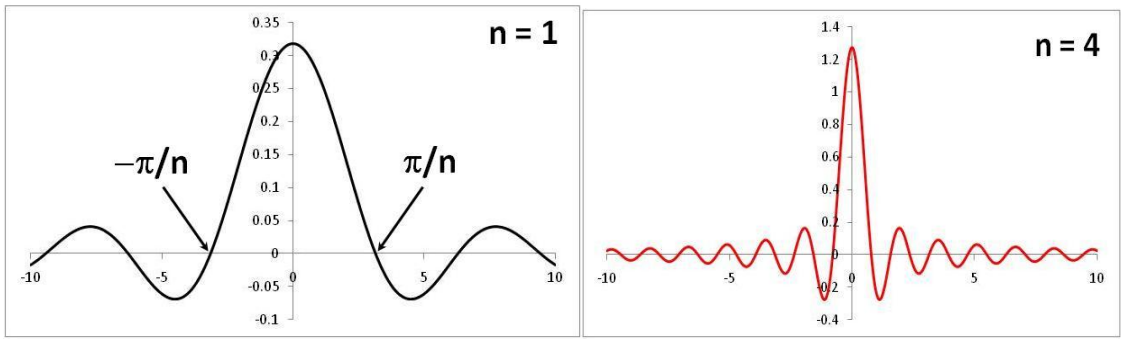


Figura 1 – Fonte: [18]. Gráfico da função $\delta_n\left(x-x_o\right)=\dfrac{\sin\left[n\left(x-x_o\right)\right]}{\pi\left(x-x_o\right)}$ para $x_o=0$ e $n=1,\ 4$.

A altura sobe com *n* e a largura diminui. A distância entre as duas Primeiras raízes vale $2\pi/n$.

A área desta curva é:

$$I=\int_{-\infty}^{+\infty}\delta_n\left(x-x_o\right)dx=A\int_{-\infty}^{+\infty}\frac{\sin\left[n\left(x-x_o\right)\right]}{\left(x-x_o\right)}dx=A\int_{-\infty}^{+\infty}\frac{\sin\left[n\left(x-x_o\right)\right]}{n\left(x-x_o\right)}d\left(nx\right)=A\int_{-\infty}^{+\infty}\frac{\sin u}{u}du$$

Precisamos mostrar que integral $\int_{-\infty}^{+\infty}\dfrac{\sin u}{u}du$ converge. Note que $\dfrac{\sin u}{u}$ é uma função par e portanto $\int_{-\infty}^{+\infty}\dfrac{\sin u}{u}du=2\int_{0}^{+\infty}\dfrac{\sin u}{u}du$. Por causa do u do denominador de $\dfrac{\sin u}{u}$ as áreas entre duas raízes se tornam cada vez menores, como mostra a Figura 2 para valores de *u* positivos. Este fato nos mostra que $A_o+A_1+A_2+A_3+\cdots>0$ e que $A_1+A_2+A_3+\cdots<0$. Portanto $2\left(A_o-\left|A_1\right|\right)\le\int_{-\infty}^{+\infty}\dfrac{\sin u}{u}du\le 2A_o$, logo a integral converge. Sabendo que $\lim_{u\to 0}\dfrac{\sin u}{u}=1$ podemos afirmar usando um retângulo de altura 1 e largura p que $0<\int_{-\infty}^{+\infty}\dfrac{\sin u}{u}du<2\pi$.

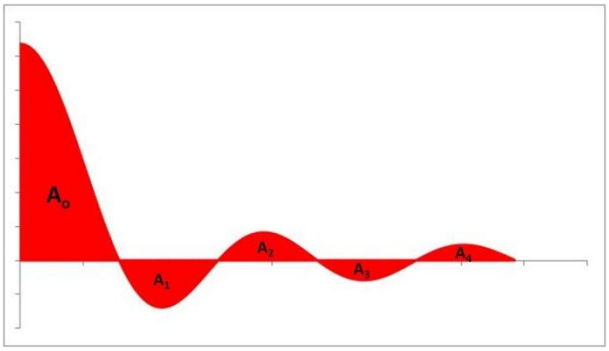


Figura 2 – Fonte: [18]. Área sobre a curva $\frac{\sin u}{u}$. As áreas pares são positivas e as ímpares negativas e as mesmas vão diminuindo com a distância $|A_{n+1}| < |A_n|$.

Sabe-se que $\int_{-\infty}^{+\infty} \frac{\sin u}{u} du = \pi$. Portanto, para garantir que a área seja unitária com esse resultado precisamos fazer $A = \frac{1}{\pi}$. Dessa forma a função $\delta_n(x - x_o) = \frac{\sin[n(x - x_o)]}{\pi(x - x_o)}$ se torna a função delta de Dirac no limite $n \to \infty$:

$$\delta(x - x_o) = \lim_{n\to\infty} \delta_n(x - x_o) = \lim_{n\to\infty} \frac{\sin[n(x - x_o)]}{\pi(x - x_o)}$$

Por outro lado podemos usar a fórmula de Euler $e^{ix} = \cos x + i \sin x$ para calcular $\frac{1}{2\pi}\int_{-n}^{+n} e^{i(x-x_o)t} dt = \frac{1}{2\pi} \frac{e^{i(x-x_o)t}}{i(x - x_o)}\Big|_{-n}^{+n} = \frac{1}{\pi(x - x_o)} \frac{e^{in(x-x_o)} - e^{-in(x-x_o)}}{2i} = \frac{\sin[n(x - x_o)]}{\pi(x - x_o)}$.

Extraindo a importante identidade:

$$\frac{1}{2\pi}\int_{-\infty}^{+\infty} e^{i(x-x_o)t} dt = \delta(x - x_o)$$

Esta identidade é útil na transformada inversa de Fourier, no teorema da convolução e no teorema central do limite [18].

APÊNDICE B – PROPRIEDADES DA VARIÂNCIA E COVARIÂNCIA

As propriedades da covariância são:

1) $\operatorname{cov}(x_1, x_2) = \operatorname{cov}(x_2, x_1)$

pois $(x_1 - \mu_1)(x_2 - \mu_2) = (x_2 - \mu_2)(x_1 - \mu_1)$

2) $\operatorname{cov}(x_1 + x_2, x_3) = \operatorname{cov}(x_1, x_3) + \operatorname{cov}(x_2, x_3)$, pois:

$$(x_1 + x_2 - \mu_1 - \mu_2)(x_3 - \mu_3) = (x_1 - \mu_1)(x_3 - \mu_3) + (x_2 - \mu_2)(x_3 - \mu_3)$$

3) $\operatorname{cov}(x, y) = E[xy] - E[x]E[y]$

onde

$$\operatorname{cov}(x, y) = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} (x - \mu_x)(y - \mu_y) f(x, y)\, dxdy =$$

$$= \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} (xy - \mu_x y - \mu_y x + \mu_y \mu_x) f(x, y)\, dxdy$$

portanto

$$\operatorname{cov}(x, y) = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} xy\, f(x, y)\, dxdy - \mu_x \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} y\, f(x, y)\, dxdy +$$

$$-\mu_y \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} x\, f(x, y)\, dxdy + \mu_y \mu_x \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} f(x, y)\, dxdy$$

$$\operatorname{cov}(x, y) = E[xy] - \mu_x E[y] - \mu_y E[x] + \mu_y \mu_x$$

$$\operatorname{cov}(x, y) = E[xy] - \mu_x \mu_y - \mu_y \mu_x + \mu_y \mu_x = E[xy] - \mu_x \mu_y = E[xy] - E[x]E[y]$$

4) $\operatorname{cov}(\alpha x, \beta y) = \alpha\beta \operatorname{cov}(x, y)$

$$\operatorname{cov}(\alpha x, \beta y) = E[\alpha\beta xy] - E[\alpha x]E[\beta y] = \alpha\beta E[xy] - \alpha\beta E[x]E[y] =$$
$$= \alpha\beta\left(E[xy] - E[x]E[y]\right)$$

5) $\operatorname{cov}(x, k) = 0$ onde k é uma constante.

$$\operatorname{cov}(x,k) = E[kx] - E[x]E[k] = kE[x] - kE[x] = 0.$$

Tais propriedades dão origem as seguintes propriedades da variância:

1) $V(x) = E\left[x^2\right] - \left(E[x]\right)^2$ pois $V(x) = \operatorname{cov}(x,x) = E\left[x^2\right] - E[x]E[x]$

2) $V[kx] = k^2 V[x]$ pois $V[kx] = \operatorname{cov}(kx,kx) = k^2 \operatorname{cov}(x,x)$

3) $V[\alpha + \beta x] = \beta^2 V[x]$

$$V[\alpha + \beta x] = \operatorname{cov}(\alpha + \beta x, \alpha + \beta x) = \operatorname{cov}(\alpha, \alpha + \beta x) + \operatorname{cov}(\beta x, \alpha) + \operatorname{cov}(\beta x, \beta x) =$$
$$= \beta^2 \operatorname{cov}(x,x)$$

4) $V[\alpha x + \beta y] = \alpha^2 V[x] + \beta^2 V[y] + 2\alpha\beta \operatorname{cov}(x,y)$

Corolário: $V[x \pm y] = V[x] + V[y] \pm 2\operatorname{cov}(x,y)$

$$V[\alpha x + \beta y] = \operatorname{cov}(\alpha x + \beta y, \alpha x + \beta y) =$$
$$= \operatorname{cov}(\alpha x, \alpha x) + \operatorname{cov}(\alpha x, \beta y) + \operatorname{cov}(\beta y, \alpha x) + \operatorname{cov}(\beta y, \beta y) =$$
$$= \alpha^2 \operatorname{cov}(x,x) + \alpha\beta \operatorname{cov}(x,y) + \alpha\beta \operatorname{cov}(y,x) + \beta^2 \operatorname{cov}(y,y) =$$
$$= \alpha^2 \operatorname{cov}(x,x) + 2\alpha\beta \operatorname{cov}(x,y) + \beta^2 \operatorname{cov}(y,y)$$

APÊNDICE C – APRENDIZAGEM REFORÇADA

Os algoritmos de aprendizagem reforçada (*deep learning* ou DL) são sofisticados a ponto de gerarem seus próprios modelos a partir dos dados. Os modelos geralmente demoram para ser treinados mas depois desta fase o modelo é capaz de classificar nos dados rapidamente. Alguns destes algoritmos são redes neurais, vamos definir inicialmente a regressão logística que visa separar duas classes, em seguida vamos transformar a regressão logística em um *perceptron*. Um perceptron funciona com um simples neurônio de uma rede neural, depois generalizaremos para redes profundas com vários perceptrons conectados capazes de reconhecer regiões não-lineares. Por último definimos a rede neural convolucional.

C.1 Regressão Logística

A regressão logística mais simples tem a tarefa de iterativamente encontrar uma reta que separa duas classes. Para ilustrar exemplificaremos com uma classificação de dois minerais, dolomita e calcita. Temos 43 espectros de dolomita e 49 de calcita baixados da biblioteca RRUFF [63].

C.1.1 Seleção de características

Para classificar selecionamos como características de cada espectro o número de onda e largura do pico principal. Selecionamos apenas uma região de interesse dos espectros e normalizamos pelo máximo de cada espectro. Um procedimento automático para encontrar as posições dos picos do *i*-ésimo espectro seria utilizar uma janela móvel selecionando os máximos locais e salvando apenas os pontos coincidentes com o espectro em um vetor $V_{\max}^{(i)}$ (semelhante a janela móvel seletora de mínimos locais no caso de remoção de background) selecionando os pontos maiores que o desvio padrão do próprio vetor $V_{\max}^{(i)} > \sigma_V$. Como estamos já em uma região de interesse selecionada, a informação da posição do pico é dada pelo ponto com maior intensidade:

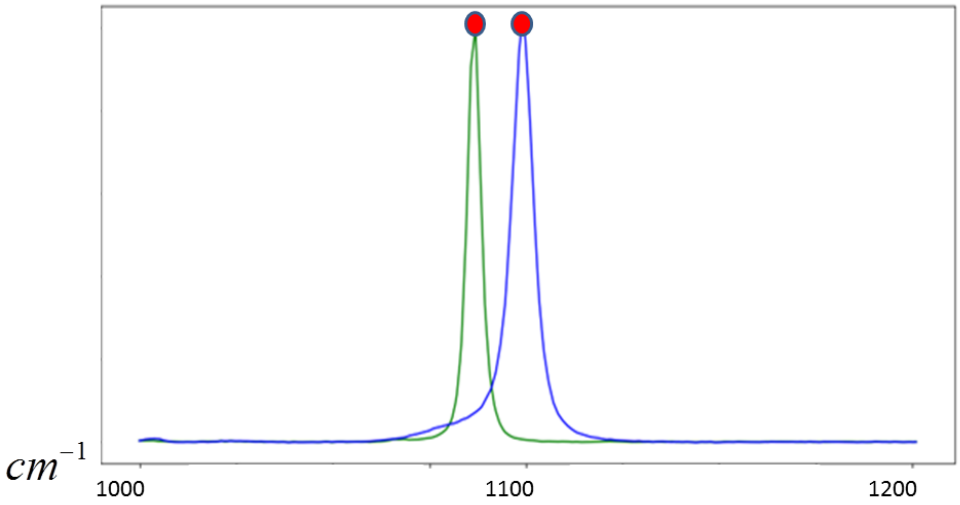


Figura 3 – Uma calcita (verde) e uma dolomita (azul) com os máximos selecionado.

Para extrair as larguras fazemos um *fitting* de uma distribuição similar as Lorentzianas em cada pico. A Lorentziana é dada por:

$$L(x;x_0,\gamma)=\frac{1}{\pi\gamma\left[1+\left(\frac{x-x_0}{\gamma}\right)^2\right]}$$

e a nossa hipótese será:

$$g(x;x_0,h,\gamma)=\frac{h}{1+\left(\frac{x-x_0}{\gamma}\right)^2}$$

As alturas *h* são unitárias (os espectros são normalizados pela intensidade máxima de cada espectro), x_0 é a posição do pico (em número de onda), γ é a largura e x é o eixo do número de onda. Aplicamos o algoritmo de descida de gradiente para minimizar uma função de custo na forma:

$$J(\gamma)=\frac{1}{2m}\sum_{i=1}^{m}(g(x_i)-Y_i)^2$$ onde m é o número de pontos. Inicializamos as larguras com valor unitário e iteragimos 100 vezes para cada largura com a fórmula a seguir:

$$\gamma := \gamma-\alpha\frac{\partial}{\partial\gamma}J(\gamma)$$

O fator α é chamado de taxa de aprendizagem e serve para atualizar o parâmetro dando pequenos passos em direção ao gradiente da função de custo de modo a encontrar uma região de minimização. Utilizamos $\alpha=0,01$. A derivada é:

$$\frac{\partial}{\partial\gamma}J(\gamma)=\frac{1}{2m}\sum_{i=1}^{m}\frac{1}{\gamma^3\beta^3}(1-Y_i\beta_i)(x-x_0)^2 \qquad \text{com} \qquad \beta=1+\left(\frac{x-x_0}{\gamma}\right)^2.$$

Encontrando uma largura aproximada como vemos na figura:

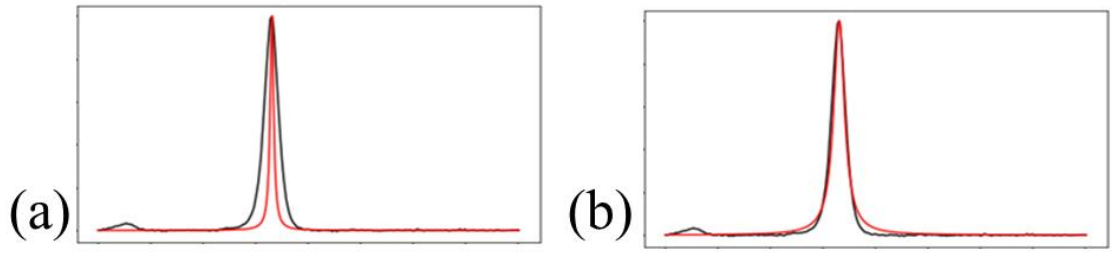


Figura 4 – Minimizando a função de custo *J* em relação ao parâmetro γ. Curva com largura unitária (a) e largura ótima (b).

Plotando todos os pontos selecionados em um gráfico de largura *vs* posição:

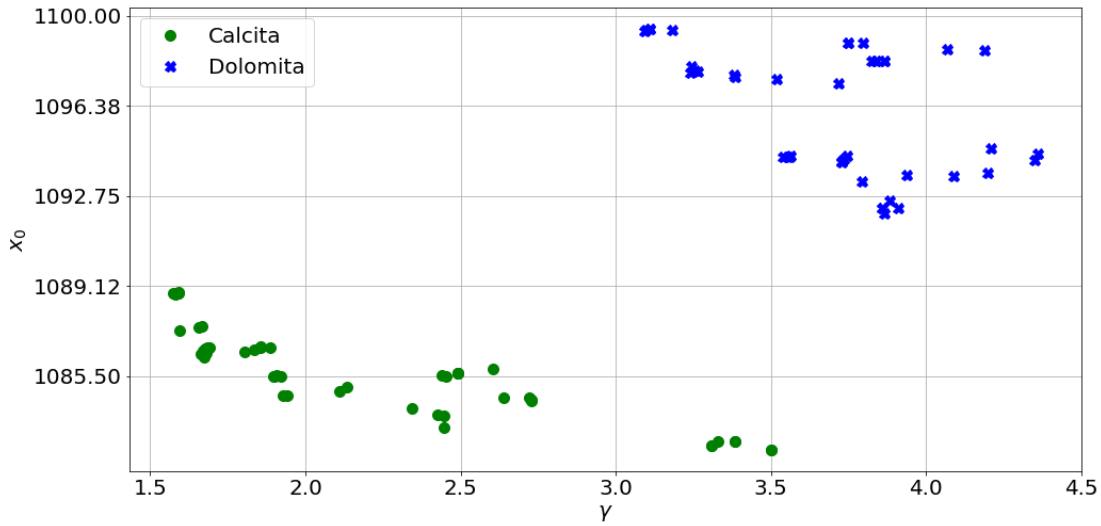


Figura 5 – Posição e largura do pico principal das dolomitas (azul) e calcitas (verde)

Vale ressaltar que deve-se escolher uma taxa de aprendizagem apropriada, uma taxa muito pequena levaria pequenos passos em direção ao mínimo ocasionando uma lenta convergência. Uma taxa de aprendizagem muito alta ocasionaria uma divergência, pois os passos poderiam atravessar a região de mínimo e continuar ricocheteando sem convergir:

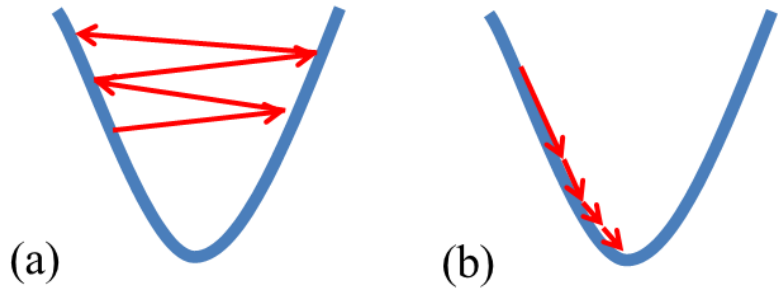


Figura 6 – Uma taxa de aprendizagem alta levando a divergência (a) e uma taxa baixa levando a lenta convergência (b) [96].

C.1.2 Criando uma função de custo

As escalas devem ser normalizadas, para isso geramos as novas variáveis:

$x_1 = \frac{x_0 - \overline{x}_0}{\sigma_{x_0}}$ e $x_2 = \frac{\gamma - \overline{\gamma}}{\sigma_\gamma}$ para cada ponto.

Para separar as classes dolomita e calcita desejamos encontrar uma reta $w_1 x_1 + w_2 x_2 + b = 0$ ou de forma mais compacta: $WX + b = 0$. Atribuímos valores aleatórios para os pesos *w* e o *bias b*:

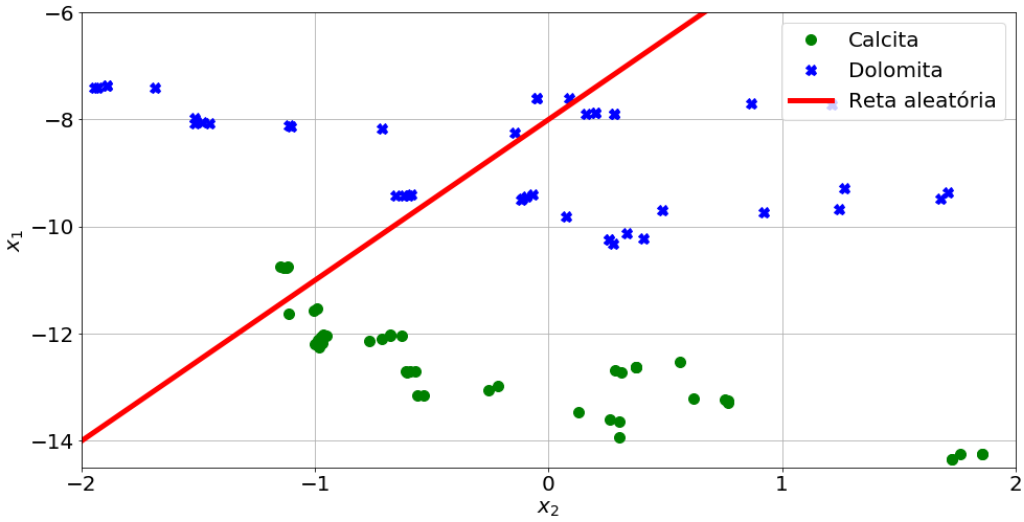


Figura 7 – Linha com parâmetros aleatórios e pontos com escalas normalizadas

Queremos que a reta separe as classes de forma que as dolomitas (azul) fiquem acima da reta, assim as classes são dadas por valores binários. Para cada ponto:

$$y = \begin{cases} 1, & dolomita \\ 0, & calcita \end{cases}$$

A classe predita pela reta é:

$$\hat{y} = \begin{cases} 1, & WX + b \geq 0 \\ 0, & WX + b < 0 \end{cases}$$

Os pontos abaixo (acima) da reta são classificadas como calcita (dolomita). Vemos que a classe predita é dada pela função degrau $\hat{y} = H(WX + b)$. Se desejarmos a probabilidade do ponto pertencer a classe 1 trocamos a função degrau pela função *sigmoid*: $g(z) = \frac{1}{1+e^{-z}}$.

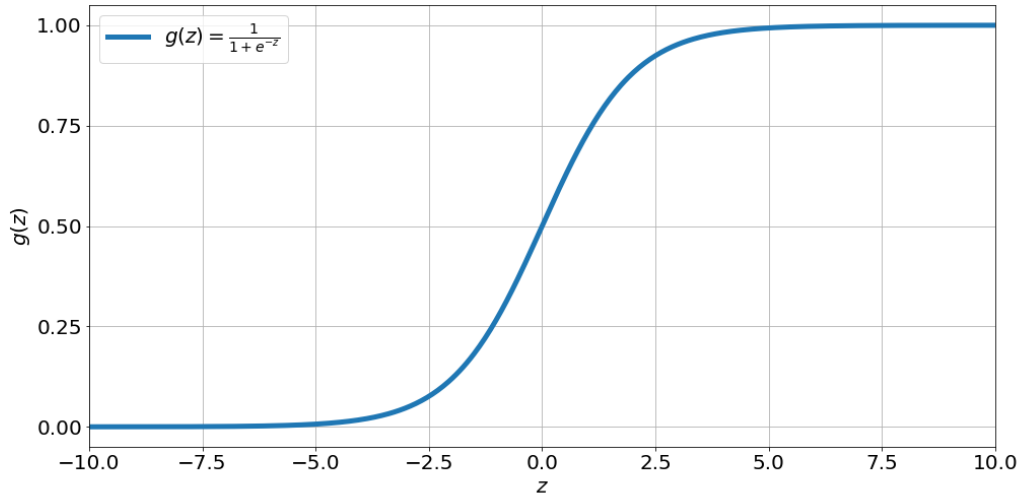


Figura 8 – Função sigmoid.

A probabilidade dada pela reta é $\hat{P} = g(WX + b)$. A probabilidade real das classes de cada ponto poderia ser definida como o produto: $P_{total} = P_1 \cdot P_2 ... P_{m-1} \cdot P_m$. Para transformar em uma soma e facilitar, vamos trabalhar com a entropia cruzada (tomando o negativo do logaritmo das probabilidade). A função de custo E de todos os pontos juntos é:

$$E = -\frac{1}{m}\sum_{i=1}^{m}\left(y_i \ln(\hat{P}) + (1 - y_i)\ln(1 - \hat{P}_i)\right)$$

Como a função *sigmoid* é praticamente 0 e 1 para valores com módulo grande, na maioria dos casos podemos dizer que $\hat{P} \approx \hat{y}$ de forma que o erro será dado por:

$$E = -\frac{1}{m}\sum_{i=1}^{m}\left(y_i \ln(\hat{y}_i) + (1 - y_i)\ln(1 - \hat{y}_i)\right) \qquad \text{(C.1)}$$

Tomando apenas o erro individual de um *i*-ésimo ponto cuja classe é $y_i = 1$, o segundo

termo da equação acima zera e o erro naquele ponto é $e_i = -y_i \ln(\hat{P}_i)$. Se a reta classifica bem o ponto com uma probabilidade de, por exemplo, $\hat{P}_i = 0.87$, o erro é $e_i \approx 0{,}139$ mas se o ponto foi mal classificado com um $\hat{P}_i = 0.21$ o erro será $e_i \approx 1{,}561$. Similarmente podemos verificar o caso quando a classe é $y_i = 0$. Verifica-se que a entropia cruzada retorna um valor alto quando o ponto é mal classificado portanto devemos minimizar a *E*.

C.1.3 Minimizando a função de custo

Para minimizar a função de custo vamos utilizar novamente a descida de gradiente iteragindo várias vezes a sequência de operações:

$$w_1 := w_1 - \alpha \frac{\partial E}{\partial w_1}$$

$$w_2 := w_2 - \alpha \frac{\partial E}{\partial w_2}$$

$$b := b - \alpha \frac{\partial E}{\partial b}$$

Desta forma damos pequenos passos em direção a região de mínimo. Abaixo vemos uma figura que ilustra:

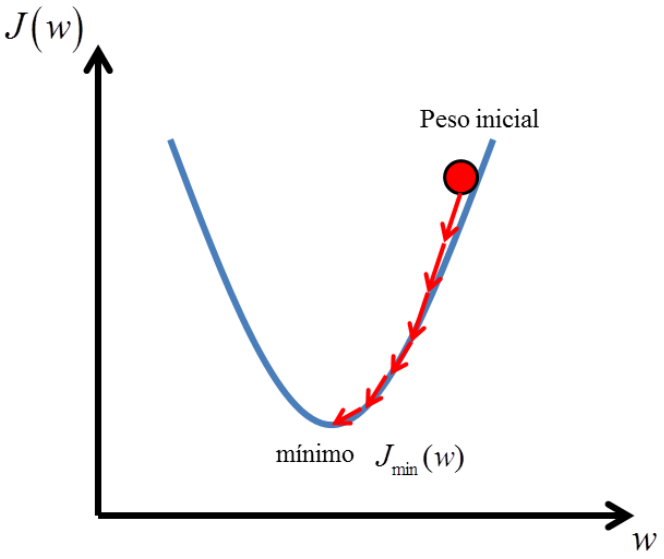


Figura 9 – Os pesos aleatórios representados pelo ponto azul caminham em direção ao mínimo em vermelho seguindo a direção do gradiente da função erro [89].

Devemos calcular as derivadas da função erro, para isso vamos derivar a função *sigmoid:*

$$g'(z) = \frac{\partial}{\partial z}\frac{1}{1+e^{-z}} = \frac{1}{\left(1+e^{-z}\right)}\frac{e^{-z}}{\left(1+e^{-z}\right)} = g(z)(1-g(z))$$

A derivada das probabilidades preditas em relação aos pesos w_j é (utilizando a notação de Einstein para índices repetidos $\sum_i w_i x_i \equiv w_i x_i$) :

$$\frac{\partial}{\partial w_j}\hat{y} = \frac{\partial}{\partial w_j} g(w_i x_i + b)$$
$$= g(w_i x_i + b)\left(1 - g(w_i x_i + b)\right)\cdot\frac{\partial}{\partial w_j}(w_i x_i + b)$$
$$= \hat{y}\left(1-\hat{y}\right)\cdot\frac{\partial}{\partial w_j}(w_i x_i + b)$$
$$= \hat{y}\left(1-\hat{y}\right)\cdot x_j$$

O erro de um único ponto é dado por:

$$E = -y\ln(\hat{y}) - (1-y)\ln\left(1-\hat{y}\right) \qquad \text{(C.2)}$$

E a derivada do erro em relação aos pesos é dada por:

$$\frac{\partial}{\partial w_j}E = \frac{\partial}{\partial w_j}\left[-y\ln(\hat{y}) - (1-y)\ln\left(1-\hat{y}\right)\right]$$
$$= -\frac{y}{\hat{y}}\frac{\partial}{\partial w_j}\hat{y} - \left(\frac{1-y}{1-\hat{y}}\right)\frac{\partial}{\partial w_j}\left(1-\hat{y}\right)$$
$$= -\frac{y}{\hat{y}}\cdot\hat{y}(1-\hat{y})x_j - \left(\frac{1-y}{1-\hat{y}}\right)\cdot(-1)\hat{y}(1-\hat{y})x_j$$
$$= -y(1-\hat{y})x_j + \hat{y}(1-\hat{y})x_j$$
$$= -\left(y-\hat{y}\right)x_j$$

Similarmente calculamos a derivada da probabilidade predita em relação ao bias:

$$\frac{\partial}{\partial b}\hat{y} = \hat{y}\left(1-\hat{y}\right)\cdot\frac{\partial}{\partial b}(w_i x_i + b)$$
$$= \hat{y}\left(1-\hat{y}\right)$$

e a derivada do erro fica:

$$\frac{\partial}{\partial b}E = -\frac{y}{\hat{y}}\frac{\partial}{\partial b}\hat{y} - \left(\frac{1-y}{1-\hat{y}}\right)\frac{\partial}{\partial b}(1-\hat{y})$$
$$= -\frac{y}{\hat{y}}\cdot\hat{y}(1-\hat{y}) - \left(\frac{1-y}{1-\hat{y}}\right)\cdot(-1)\hat{y}(1-\hat{y})$$
$$= -(y-\hat{y})$$

C.1.4 Algoritmo gradiente descendente

Estamos prontos para atualizar os valores dos pesos e dos bias. Para todos os ponto, somamos as derivadas dos erros e dividimos por *m* para recuperar o erro total (C.1) e realizamos as operações:

$$w_1 := w_1 - \alpha\frac{1}{m}\sum_{i=1}^{m}(y_i - \hat{y}_i)x_{1i}$$
$$w_2 := w_2 - \alpha\frac{1}{m}\sum_{i=1}^{m}(y_i - \hat{y}_i)x_{2i} \qquad \text{(C.3)}$$
$$b := b - \alpha\frac{1}{m}\sum_{i=1}^{m}(y_i - \hat{y}_i)$$

onde x_{ji} corresponde a coordenada *j* do ponto *i*. Em seguida repetimos o mesmo processo várias vezes. Após 100 iterações com um $\alpha = 0,01$ obtivemos o seguinte resultado:

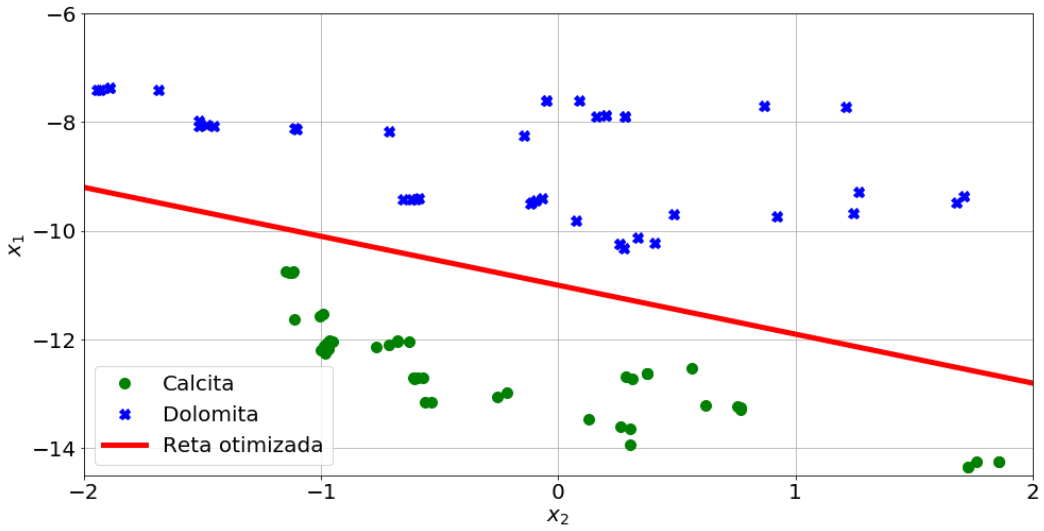


Figura 10 – Linha otimizada que separa as classes 0 e 1.

Aqui chamamos atenção para a importância da normalização. As escalas diferentes fazem como que a função de custo fique alongada nas dimensões com escalas maiores

tornando o gradiente descendente lenta.

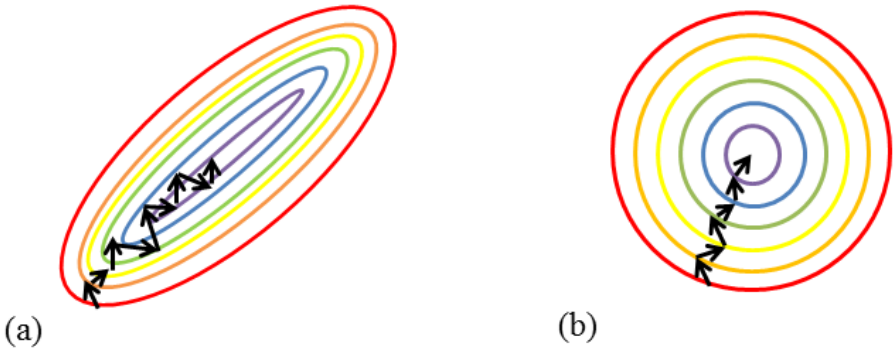


Figura 11 – Fonte: [94]. Uma função de custo com pesos não normalizados (a) e uma normalizada (b). A função normalizada tende a ter uma convergencia a região de mínimo mais rápida.

Este método teve o intuito de ilustrar a regressão logística. É possível encontrar regiões não-lineares adicionando como inputs os termos cruzados, mas para um grande número de inputs isto se torna cada vez mais custoso computacionalmente. Além disso, este método não seria bom para um caso mais geral de classificação de vários minerais. Os picos principais podem coincidir com os picos de outros minerais, os picos secundários mudam de posição e forma de um espectrômetro para outro, mas visualizando a imagem do espectro é possível identificar o mineral, esta é a principal motivação para a ideia de construir uma rede neural que imita córtex visual.

C.2.1 Operadores lógicos

Os perceptrons podem representar operadores lógicos, a motivação aqui é demonstrar a classificação em regiões não lineares combinando os perceptrons, formando redes neurais. Os principais operadores lógicos são AND, OR e NOT. Para representar os operadores, x_1 e x_2 serão binários. O mais simples é o operador NOT que retorna o contrário do *input*: NOT 0 = 1 e NOT 1 = 0 e pode ser representada por uma função degrau invertida:

$$H(-z) = \begin{cases} 1, & z \leq 0 \\ 0, & z > 0 \end{cases}$$

Os operadores AND e OR comparam x_1 e x_2. O operador AND com a função de ativação do operador NOT retorna o operador NAND:

x_1	x_2	AND	OR	NAND
1	1	1	1	0
1	0	0	1	1
0	1	0	1	1
0	0	0	0	1

Tabela 1 – Outputs dos operadores AND, OR e NAND.

Os gráficos dos operadores OR e AND são representados na Figura 12 traçando uma linha para separar os 4 pontos da Tabela acima:

-

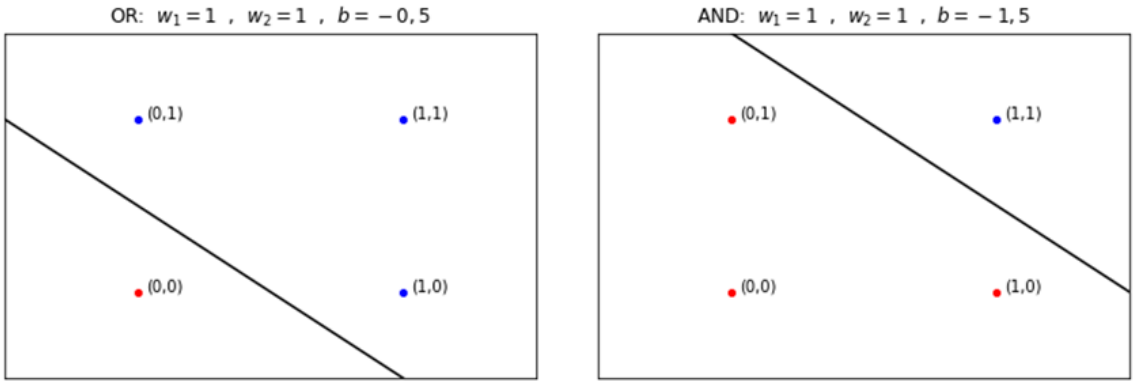


Figura 12 – Gráficos dos perceptrons OR e AND.

A reta para os operadores OR e AND classificam os pontos acima da reta como 1 pois sua função de ativação é a função degrau $H(z)$ com $z = WX_i + b$ para um ponto *i*. O gráfico para o NAND é idêntico ao do operador AND mas classificando como 1 a região abaixo da linha utilizando como função de ativação $H(-z)$.

Queremos construir o operador XOR (exclusive OR) que retorna 0 se ambas entradas forem iguais:

x_1	x_2	XOR
1	1	0
1	0	1
0	1	1
0	0	0

Tabela 2 – Outputs do operador XOR

Devemos combinar os perceptrons OR, NAND e AND:

Figura 13 – Representação de uma combinação de perceptrons para gerar o operador XOR (suprimimos o x_0).

C.3.1 Alimentação da rede

Representaremos cada elemento da matriz de pesos na forma $w_{il}^{(j)}$ iniciada aleatoriamente que mapeia a camada *j* na camada *j+1*. O índice *i* representa a unidade de ativação na camada *j+1* e o índice *l* representa a unidade de ativação da camada anterior *j*. Em cada camada *j* teremos d_j unidades de forma que a dimensão de $w^{(j)}$ será $(d_{j+1}-1)\cdot d_j$. O fator de subtração vem do fato de que o elemento *bias* $a_0^{(j+1)}$ não interage com a camada anterior. Desta forma, na rede da figura 14, os nós da segunda camada são:

$$a_1^{(2)} = g\left(w_{10}^{(1)}x_0 + w_{11}^{(1)}x_1 + w_{12}^{(1)}x_2\right) = g\left(z_1^{(1)}\right)$$
$$a_2^{(2)} = g\left(w_{20}^{(1)}x_0 + w_{21}^{(1)}x_1 + w_{22}^{(1)}x_2\right) = g\left(z_2^{(1)}\right)$$

Os bias $a_0^{(j)}$ são unitários e por vezes os omitiremos dos diagramas. De uma forma mais compacta, podemos escrever as expressões acima como: $a_i^{(2)} = g\left(\sum_{k=0}^{2} w_{il}^{(1)}x_i\right) = g(z_i^{(1)})$.

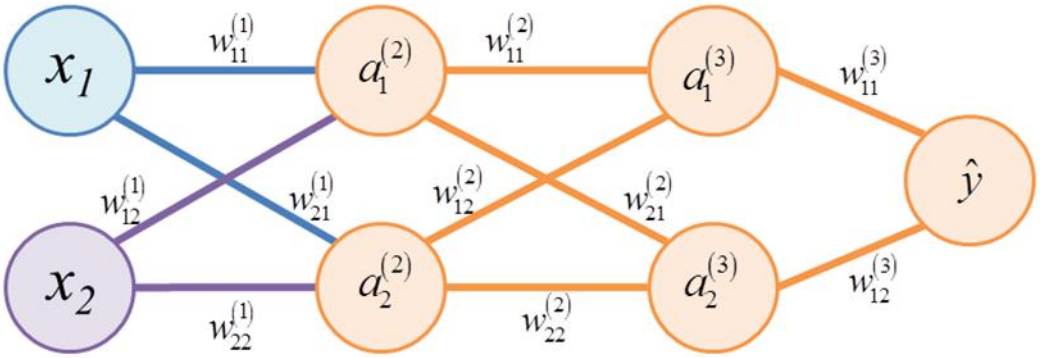


Figura 14 – Rede neural com duas camadas ocultas, os *bias* foram suPrimidos.

Neste caso vemos que a dimensão de $w^{(1)}$ é 2×3. Os pesos $w^{(1)}$ geram regiões lineares nas unidades $a^{(2)}$ e os pesos $w^{(2)}$ combinam as regiões lineares formando regiões não-lineares na próxima camada $a^{(3)}$.

Na camada de saída temos:

$$\hat{y} = a_1^{(4)} = g\left(w_{10}^{(3)}a_0^{(3)} + w_{11}^{(3)}a_1^{(3)} + w_{12}^{(3)}a_2^{(3)}\right) = g\left(z_1^{(2)}\right)$$

Com a função de ativação logística. Em uma forma mais geral temos: $a_i^{(j)} = g\left(\sum_{k=0}^{d_{j-1}} w_{il}^{(j-1)}a_i^{(j-1)}\right) = g\left(z_i^{(j-1)}\right)$.

A dimensão da saída é um escalar que dá a probabilidade do ponto pertencer a classe 1. A classe é dada por $H\left(a_1^{(4)}\right)$ onde *H* é a função degrau, classificando se o ponto é da classe (1) ou não. Como a função *sigmoid* é muito próxima da função degrau para valores distantes da origem, vamos computar as predições como $\hat{y} = a_1^{(3)} = g\left(z_1^{(2)}\right)$.

Vemos que a predição é alimentada do início ao fim da rede propagando os valores iniciais $\hat{y} = g \circ w^{(2)} \circ g \circ w^{(1)}(x)$. Para uma rede com *n* camadas (contando os inputs e outputs) a predição da classe é dada por $\hat{y} = g \circ w^{(n-1)} \circ g \circ w^{(n-2)} \ldots \circ g \circ w^{(2)} \circ g \circ w^{(1)}(x)$.

No caso de *m* pontos, as coordenadas x_i serão vetores onde x_1 contém a as informações da Primeira entrada dos *m* pontos e assim por diante. As unidades de ativação também terão dimensão de vetor com tamanho *m*. Cada unidade de peso $w_{ik}^{(j)}$ continua como sendo um escalar que é multiplicado pelos vetores das unidades de ativações da camada anterior.

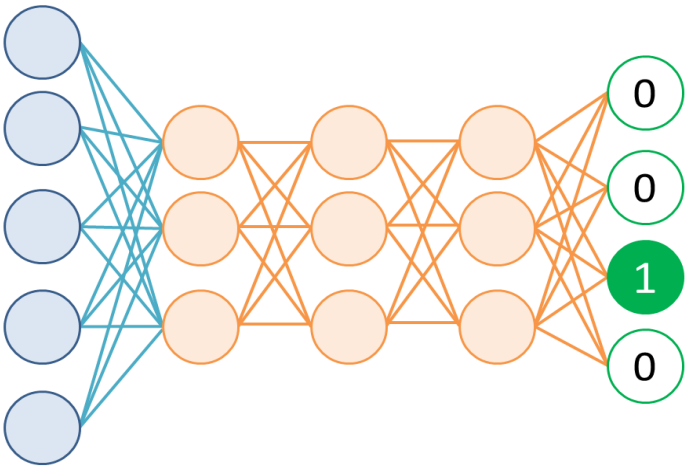


Figura 15 – Arquitetura de uma rede neural com três camadas ocultas e quatro classes.

Cada unidade de ativação do output $a_k^{(L)}$ (onde L é o número de camadas) será um vetor com tamanho m com as predições para os m pontos com valores 0 ou 1 para o caso em que o ponto pertença a classe k. Portanto cada um dos d_L outputs $\left(k \in \{1,2,...,d_L\}\right)$ serão vetores. Visualizando, na última camada da Figura 15:

$$a^{(5)} = \begin{bmatrix} a_1^{(5)} & a_2^{(5)} & a_3^{(5)} & a_4^{(5)} \end{bmatrix} = \begin{bmatrix} \left(a_1^{(5)}\right)_1 & \left(a_2^{(5)}\right)_1 & \left(a_3^{(5)}\right)_1 & \left(a_4^{(5)}\right)_1 \\ \left(a_1^{(5)}\right)_2 & \left(a_2^{(5)}\right)_2 & \left(a_3^{(5)}\right)_2 & \left(a_4^{(5)}\right)_2 \\ \vdots & \vdots & \vdots & \vdots \\ \left(a_1^{(5)}\right)_m & \left(a_2^{(5)}\right)_m & \left(a_3^{(5)}\right)_m & \left(a_4^{(5)}\right)_m \end{bmatrix}$$

A notação acima na última matriz é bastante esquisita e foi posta apenas para melhor visualização dos outputs. Omitiremos sempre os índices que representam a identidade dos m pontos.

C.3.2 Retropropagação

Para atualizar os pesos devemos fazer o caminho inverso, calcular as derivadas do erro em relação aos pesos da última camada e retroagir.

No exemplo da figura 14 (classificação binária), A derivada do erro em relação ao peso $w_{11}^{(2)}$ pode ser expressa pelo produto das derivadas:

$$\frac{\partial E}{\partial w_{11}^{(2)}} = \frac{\partial E}{\partial a_1^{(3)}} \frac{\partial a_1^{(3)}}{\partial w_{11}^{(2)}} = \frac{\partial E}{\partial \hat{y}} \frac{\partial \hat{y}}{\partial z_1^{(2)}} \frac{\partial z_1^{(2)}}{\partial w_{11}^{(2)}}$$

Para o peso $w_{11}^{(1)}$ adicionamos um novo termo:

$$\frac{\partial E}{\partial w_{11}^{(1)}} = \frac{\partial E}{\partial a_1^{(3)}} \frac{\partial a_1^{(3)}}{\partial a_1^{(2)}} \frac{\partial a_1^{(2)}}{\partial w_{11}^{(1)}} = \frac{\partial E}{\partial \hat{y}} \frac{\partial \hat{y}}{\partial z_1^{(2)}} \frac{\partial z_1^{(2)}}{\partial z_1^{(1)}} \frac{\partial z_1^{(1)}}{\partial w_{11}^{(1)}}$$

Para o erro em um único ponto dado por (C.2), calculando cada derivada separadamente temos:

$$\frac{\partial E}{\partial \hat{y}} = \left(\frac{1-y}{1-\hat{y}}\right) - \frac{y}{\hat{y}}; \qquad \frac{\partial \hat{y}}{\partial z_1^{(2)}} = g\left(z_1^{(2)}\right)\left[1 - g\left(z_1^{(2)}\right)\right];$$

$$\frac{\partial z_1^{(2)}}{\partial z_1^{(1)}} = w_{11}^{(2)} \cdot g\left(z_1^{(1)}\right)\left[1 - g\left(z_1^{(1)}\right)\right]; \qquad \frac{\partial z_1^{(1)}}{\partial w_{11}^{(1)}} = x_1$$

Ao calcularmos a derivada em relação ao peso $w_{21}^{(1)}$ teremos (neste caso $\hat{y} = a_1^{(4)}$):

$$\frac{\partial E}{\partial w_{21}^{(1)}} = \frac{\partial E}{\partial a_1^{(4)}} \left(\frac{\partial a_1^{(4)}}{\partial a_1^{(3)}} \frac{\partial a_1^{(3)}}{\partial a_2^{(2)}} + \frac{\partial a_1^{(4)}}{\partial a_2^{(3)}} \frac{\partial a_2^{(3)}}{\partial a_2^{(2)}} \right) \frac{\partial a_2^{(2)}}{\partial w_{21}^{(1)}}$$

Vemos que ambos os termos da terceira camada dependem de $a_2^{(2)}$. Podemos generalizar utilizando a forma simples:

$$\frac{\partial E}{\partial w_{il}^{(j)}} = \frac{\partial E}{\partial a^{(L)}} \frac{\partial a^{(L)}}{\partial a^{(L-1)}} \cdots \frac{\partial a^{(j+1)}}{\partial w_{il}^{(j)}} = \sum_{\eta=1}^{d_L} \frac{\partial E}{\partial a_\eta^{(L)}} \left(\sum_{\varphi=1}^{d_{L-1}} \frac{\partial a_\eta^{(L)}}{\partial a_\varphi^{(L-1)}} (...) \right) \frac{\partial a_i^{(j+1)}}{\partial w_{il}^{(j)}}$$

Que já engloba o caso multiclasse (camada *L*). O erro total é dado por:

$$E = -\frac{1}{m} \sum_{k=1}^{K} \sum_{i=1}^{m} \left(y_{ik} \ln(\hat{y}_{ik}) + (1 - y_{ik}) \ln(1 - \hat{y}_{ik}) \right)$$

Onde *K* é o total de classes. O erro individual para cada ponto é:

$$E = -\sum_{k=1}^{K} \left(y_k \ln(\hat{y}_k) + (1 - \hat{y}_k) \ln(1 - \hat{y}_k) \right)$$

Para redes maiores as derivadas tornam-se cada vez mais complicadas, mas são computadas numericamente. Os valores iniciais dos pesos devem ser inicializados aleatoriamente mas com a condição de que a variância dos pesos em cada camada seja dada por $\text{var}(w^{(j)}) = \frac{1}{n_j}$ onde n_j é o número de pesos na camada *j*. Este procedimento é importante pois como os pesos se propagam ao longo da rede, se os pesos forem muito pequenos (grandes) os gradientes tendem a zero (explodir).

As derivadas são sempre calculadas da direita para a esquerda de forma a retroagir seus valores. Uma vez calculada as derivadas aplica-se a descida de gradiente. Para cada ponto subtrai-se do peso a derivada multiplicada pela taxa de aprendizagem como no caso da

regressão logística [90] repetindo o processo pelo número de épocas de treino.

C.3.3 Evitando Underfitting e Overfitting

Podemos ilustrar na figura abaixo três modelos de classificação que geram um *underfitting*, um *overfitting* e um modelo ideal.

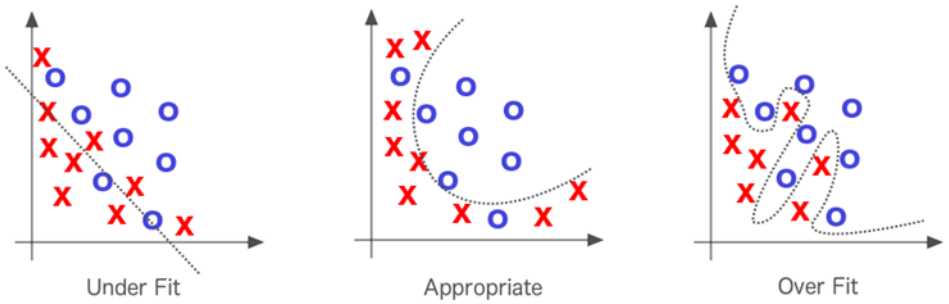


Figura 16 – Três modelos que geram um *underfiting*, um modelo ideal e um modelo que gera um *overfitting* [91].

Os modelos seguintes poderiam ser exemplificados através das seguintes redes neurais:

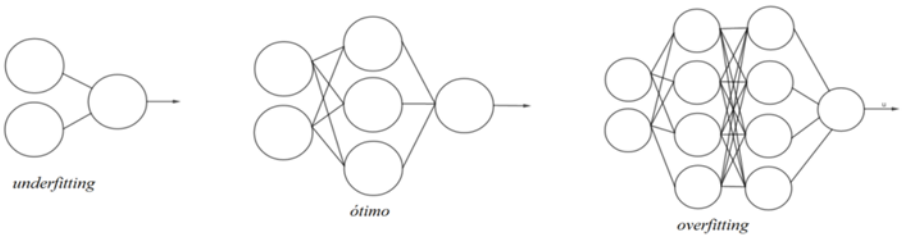


Figura 17 – Três redes neurais, um gera um *underfiting*, um é o modelo ideal e um modelo que gera um *overfitting*.

O modelo que gera o *underfitting* é muito simples para descrever a curva que separa as duas classes, o modelo que gera o *overfitting* é muito complexo e falhará ao classificar novos pontos além dos utilizados. O modelo ótimo erra em alguns pontos mas generalizará melhor. Para sermos capazes de avaliar se o modelo é ótimo devemos ter dois conjuntos de dados

separados, um conjunto de treino que servirá para treinar a rede e um conjunto de teste avalia o modelo. Existe um terceiro conjunto chamado de conjunto de validação que é um subconjunto dos dados de treino. Ao treinar a rede em cada época, separa-se cerca de 10% do conjunto de dados de treino aleatoriamente e treina-se a rede com o restante. Avalia-se a acurácia nos dados de validação. Em seguida testa-se o modelo gerado nos dados de teste. Um modelo que *underfitta* os dados terá uma péssima acurácia nos dados de treino e teste, um modelo que *overfitta* terá uma excelente acurácia nos dados de treino e péssima nos dados de teste enquanto que o modelo ideal deve ter uma boa acurácia nos dois conjuntos. A acurácia é a medida do quanto o modelo acerta em suas predições.

Uma maneira de resolver o problema de *overfitting* em arquiteturas complexas é descartar alguns nós em cada camada com uma certa probabilidade em cada época de treino com o algoritmo **Dropout** [92]:

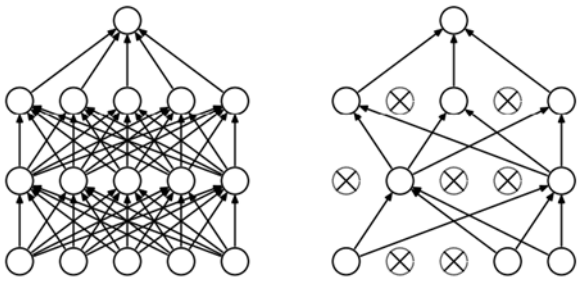


Figura 18 – Uma rede neural totalmente conectada (esquerda) e uma rede neural após descartar alguns nós – Fonte [92].

Em cada época teremos diferentes arquiteturas isomórficas dadas pelas probabilidades dos nós serem descartados que serão combinadas para criar um modelo mais geral. Ao descartarmos alguns nós, a rede neural é forçada a reconhecer características mais robustas dadas pelos conjuntos aleatórios formados quebrando a grande dependência gerada pelos nós nas redes totalmente conectadas. Desta forma o erro é retropropagado uniformemente ao longo das épocas na rede.

Alguns cuidados devem ser tomados. Em cada camada l as unidades de ativação possuem uma probabilidade $P^{(l)}$ de não serem descartadas, então para que o valor esperado de $z^{(l+1)}$ não mude, divide-se as unidades mantidas pela probabilidade:

$$z^{(l+1)} = w^{(l+1)} \frac{a^{(l)}}{P^{(l)}} + b^{(l+1)}.$$

Uma desvantagem deste método é que as derivadas não são bem definidas em cada

época.

Outra forma é adicionar um termo de **regularização** na função erro:

$$E = -\frac{1}{m}\sum_{k=1}^{K}\sum_{i=1}^{m}\left(y_{ik}\ln(\hat{p}_{ik}) + (1-y_{ik})\ln(1-\hat{p}_{ik})\right) + \frac{\lambda}{2m}\sum_{l=1}^{L-1}\sum_{i=1}^{d_l}\sum_{j=1}^{d_{l+1}}\left(w_{ji}^{(l)}\right)^2$$

O termo extra regula os pesos de forma que quando um peso é muito grande, o termo de regularização será maior, por consequência o erro também será maior, forçando o modelo a equilibrar os pesos [90].

Ao longo das épocas de treino o modelo também pode sofrer com um *underfitting* ou *overfitting* para uma mesma arquitetura da rede.

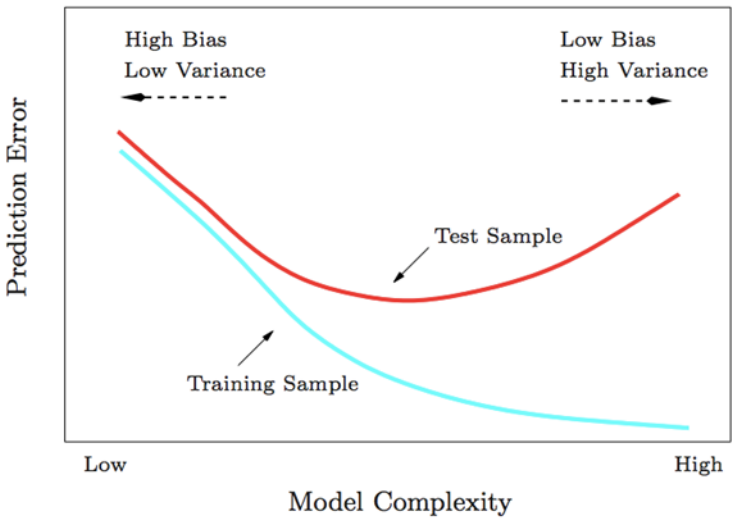


Figura 19 – Curva de complexidade do modelo.

A Figura 19 exemplifica isto. Quanto mais épocas de treino, mais complexo o modelo se torna criando curvas com alta variância que classificam bem no treino mas falham ao generalizar. Uma maneira de prevenir isto é parar o treino antecipadamente. Em cada época de treino alimenta-se a rede e retropropaga-se os erros atualizando os pesos, testa-se o modelo nos dados de validação para verificar a acurácia nos dados de treino e testa-se nos dados de teste para verificar a acurácia no teste. No momento em que os erros no teste começam a aumentar o algoritmo deve parar o treino. Este algoritmo é conhecido como **Parada antecipada** [98].

C.4 Otimizando as redes neurais

Nesta seção propõe-se diversos métodos importantes na otimização das redes neurais de

forma que estas convirjam mais rapidamente a uma solução.

C.4.1 Descida de Gradiente em mini-lotes

O método proposto anteriormente de cálculo da descida do gradiente é chamado de **Descida de gradiente em lote** (BGD). O procedimento consiste em computar todo o conjunto de dados e realizar uma atualização dos pesos em cada época. Lembrando que as unidades de ativação da Primeira camada (inputs) são vetores de tamanho *m*, $x_j = \left[\left(x_j \right)_1, \left(x_j \right)_2, ..., \left(x_j \right)_m \right]$ contendo a coordenada x_j dos *m* pontos. Consequentemente as unidades de ativação da última camada (outputs) com as predições serão vetores de tamanho *m* da forma $\hat{y}_j = \left[\left(\hat{y}_j \right)_1, \left(\hat{y}_j \right)_2, ..., \left(\hat{y}_j \right)_m \right]$ predizendo se o ponto *i* pertence a classe *j* ou não.

A descida do gradiente em lote converge para o mínimo global se a função de custo for convexa e converge a algum mínimo local caso contrário (ou ponto de sela) em passos discretos. Uma grande desvantagem é que para um conjunto de dados muito grande, a o ajuste dos pesos se torna extremamente lento pois este deve computar todos os pontos do conjunto de treino.

O método da **descida de gradiente estocástica** (SGD) consiste em cada época atualizar os pesos calculando as derivadas em relação ao erro individual (C.2) de um único ponto *i* selecionado aleatoriamente do conjunto de *m* dados.

$$w := w - \alpha \frac{\partial E}{\partial w}$$

Este procedimento é muito mais rápido em cada época e é mais ruidoso, pois o gradiente tende a mudar de direção em cada ponto diferente selecionado em cada época de treino levando um grande número de passos para encontrar uma região de mínimo. Uma vantagem é que a escolha aleatória de pontos pode fazer com que o gradiente evite convergir para uma região de mínimo local e se mova em direção do mínimo global. A desvantagem é que não é garantido que o gradiente convirja e fique dando passos ao redor da região de minimização.

Para melhorar a descida de gradiente criou-se a **descida de gradiente com mini-lotes** que é uma mistura dos dois métodos anteriores agregando as vantagens e desvantagens. Consiste em selecionar aleatoriamente *n* pontos dentre os *m* pontos. Neste caso cada coordenada x_j será um vetor com *n* entradas $x_j = \left[\left(x_j \right)_1, \left(x_j \right)_2, ..., \left(x_j \right)_n \right]$ e a saída

$\hat{y}_j = \left[\left(\hat{y}_j\right)_1, \left(\hat{y}_j\right)_2, ..., \left(\hat{y}_j\right)_n \right]$. O erro é dado por:

$$E = -\frac{1}{n}\sum_{k=1}^{K}\sum_{i=1}^{n}\left(y_{ik}\ln(\hat{y}_{ik}) + (1-y_{ik})\ln(1-\hat{y}_{ik})\right)$$

O tamanho dos lotes é ajustável. Usualmente seleciona-se um mini-lote igual a uma potência de 2 como 32, 64, 128, 256, etc. A razão disto é que hardwares como GPUs processam melhor informações destes tamanhos [94]. Na figura abaixo vemos os passos dos três métodos em direção ao mínimo:

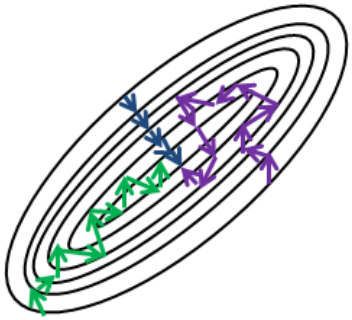


Figura 20 – Representação de descida do gradiente em lote (azul), em mini-lote (verde) e estocástica (roxo) [108].

C.4.2 Algoritmos de otimização da descida do gradiente

Nesta seção propomos algumas mudanças na forma do cálculo das atualizações dos pesos em cada época de treino de forma a tornar a descida de gradiente com mini-lotes e estocástica mais suave e rápida, para isto vamos introduzir o conceito de médias móveis com decaimento exponencial.

Definimos a seguinte notação para um ponto dependente do tempo: $q(t) \equiv q_t$. Da mesma forma definimos a média como $v_t = \beta v_{t-1} + (1-\beta) q_t$ para $t > 0$. Em $t = 0$ simplesmente $v_0 = 0$ (pois é a média de zero pontos). O parâmetro β define o número de épocas *t* da janela da média móvel dado por $\approx \frac{1}{1-\beta}$. Para um $\beta = 0{,}9$ teremos uma janela móvel que vê aproximadamente 10 épocas passadas. Isto acontece porque os pontos anteriores vão recebendo um peso cada vez menor se aproximando de zero após 10 épocas. Podemos ilustrar computando:

$$v_{10} = \beta v_9 + (1-\beta) q_9 = \beta\left(\beta v_8 + (1-\beta) q_8\right) + \ldots = \beta^2\left(\beta v_7 + (1-\beta) q_7\right) + \beta(1-\beta) q_8 + \ldots$$

As informações dos pontos anteriores vão sendo multiplicados β de forma que após 10 épocas, $\beta^{10} \sim 0,34$ ou seja os pontos 10 épocas antes da atual perderam mais de 70% de sua informação.

Os Primeiros pontos da média móvel tendem ser pequenos levando um certo período para representar a tendência de q apropriadamente. Para resolver este problema faz-se a **correção de *bias***: $v_t := \frac{v_t}{1-\beta^t}$.

Assim os Primeiros pontos recebem um ajuste fazendo-os subir acompanhando a tendência e para um t grande o denominador tende a 1.

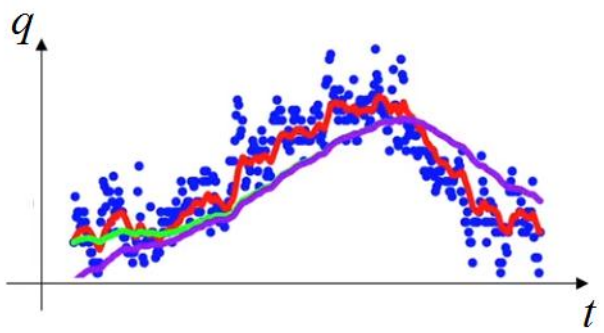


Figura 21 – Fonte: [95]. Média móvel com $\beta = 0,9$ em vermelho e médias $\beta = 0,98$ sem correção (roxo) e com correção (verde). Nota-se que a curva não corrigida não acompanha a tendência inicialmente e depois de algumas épocas ambas coincidem.

Em regiões como na ilustração a seguir:

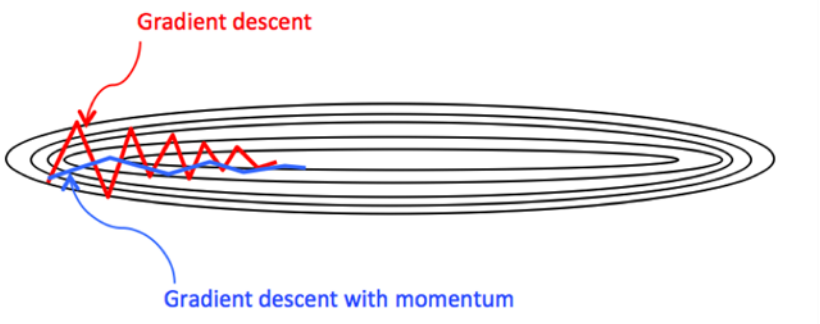


Figura 22 – Fonte: [97]. Descida do gradiente (vermelho) e descida do gradiente com momentum (azul).

Vemos que o peso que representa os passos verticais são aproximadamente nulos se fizermos uma média enquanto que o os passos na direção horizontal seriam maiores. Queremos que o algoritmo possa encontrar uma melhor direção para os passos do gradiente, para isto introduzimos a média móvel exponencial dos gradientes anteriores definindo a **descida do gradiente com momentum**. O procedimento sempre é repetido para cada novo mini-lote da nova época.

Para um peso *w* vamos definir a notação $\gamma_t = \frac{\partial E}{\partial w}$ correspondendo a derivada do peso na época *t*. As médias são dadas por $v_t = \beta_1 v_{t-1} + (1-\beta_1)\gamma_t$. As atualizações serão dadas por $w := w - \alpha v_j$.

A média dos passos na vertical tendem a zero e os passos na direção horizontal são bem mais expressivos levando a uma convergência mais rápida. Podemos pensar no termo γ_t como uma aceleração e o termo v_{t-1} como a velocidade com um fator de fricção β_1 [99].

Um algoritmo bastante semelhante é o **RMSprop** (*Root mean square prop*) que é uma média dada pelos quadrados das derivadas γ_t, em cada nova época *t* calcula-se no mini-lote:

$$s_t = \beta_2 s_{t-1} + (1-\beta_2)\gamma_t^2$$

Atualizando os pesos na forma:

$$w := w - \frac{\alpha \gamma_t}{\sqrt{s_t + \varepsilon}}$$

Analisando a fórmula acima, vemos que quando g_t é grande como no caso do peso na direção vertical na figura 22, $1/\sqrt{s_t}$ será pequeno fazendo com que os passos nessa direção sejam menores e para a direção horizontal teremos o caso contrário levando a um resultado semelhante a descida do gradiente com momentum [100].

O próximo e último algoritmo que iremos abordar chama-se **Adam** (*adaptive moment estimation*). Este algoritmo combina os dois anteriores. Faz-se a correção de bias para os dois termos:

$$v_t := \frac{v_t}{(1-\beta_1^t)}, \qquad s_t := \frac{s_t}{(1-\beta_2^t)}$$

E em cada nova época o peso *w* é atualizado na forma:

$$w := w - \alpha \frac{v_t}{\sqrt{s_t + \varepsilon}}$$

É recomendado em [101] utilizar os seguintes valores para os parâmetros: $\beta_1 = 0{,}9; \quad \beta_2 = 0{,}999; \quad \varepsilon = 10^{-8}$.

C.4.3 Normalização de lotes

A normalização de lotes (BN) tem como principal motivação acelerar o treino da rede neural da mesma forma que a normalização dos inputs. Consiste em normalizar as unidade de ativação $a_l^{(j)}$ em cada camada antes de multiplicar pelos pesos e passar pela função de ativação para obter $a_l^{(j+1)}$. Lembrando que $a^{(j+1)} = f\left(w^{(j)}a^{(j)}\right)$ então calcula-se os estimadores da média e variância:

$$\mu_j = \frac{1}{d_j}\sum_{l=1}^{d_j} a_l^{(j)} \quad \text{e} \qquad \sigma_j^2 = \frac{1}{\left(d_j - 1\right)}\sum_{l=1}^{d_j}\left(a_l^{(j)} - \mu_j\right)^2.$$

A normalização é dada na forma

$$a'^{(j)} = \frac{a^{(j)} - \mu_j}{\sqrt{\sigma_j^2 + \varepsilon}}$$

E agora deve-se minimizar os parâmetros *w* em $a^{(j+1)} = f\left(w^{(j)}a'^{(j)}\right)$.

A normalização dos lotes torna a rede mais robusta livrando-se da mudança de covariância [103]. Exemplificando com o seguinte problema de classificação:

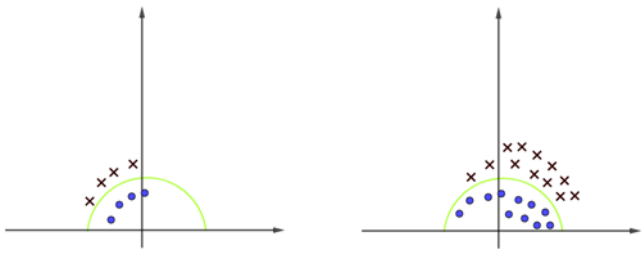


Figura 23 – Dois conjuntos de dados diferentes representando um mesmo problema de classificação.

Na Figura 23 poderíamos pensar em um classificador que identifica se uma imagem é um cão, mas o conjunto de dados da esquerda representa um classificador treinado apenas com imagens de cães brancos, ao testar no conjunto de dados do lado direito com cães de diferentes cores o modelo falha em encontrar a curva em verde. Isto acontece porque os conjuntos de dados possuem variâncias diferentes [102].

Durante o treino, em cada mini-lote calcula-se os estimadores das médias e variâncias em cada época e aplica-se uma média móvel exponencial das médias e variâncias das épocas anteriores. Os últimos estimadores da média e variância são usados na normalização do teste.

C.4.4 Funções de ativação

A derivada da função *sigmoid* vai rapidamente a zero em valores altos e baixos, isto pode fazer com que os gradientes sumam. Geralmente utiliza-se duas outras funções no lugar da *sigmoid* como ativação. No nosso caso utilizaremos a função reLU (*rectified linear*) dada por $g(x) = \max(0, x)$. Esta função tem mostrado bons resultados como função de ativação [88].

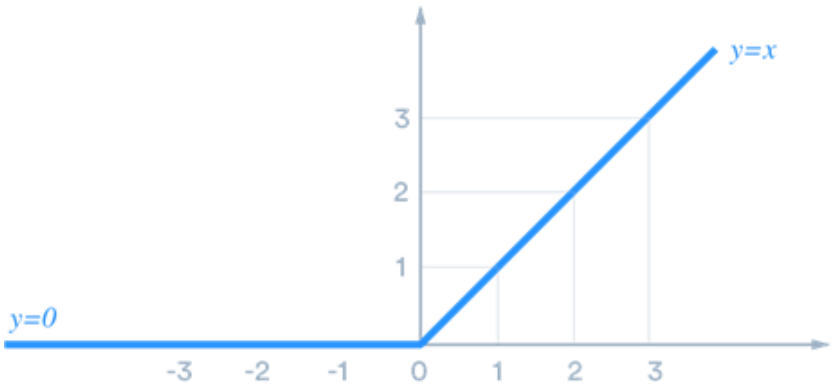


Figura 24 – Fonte: [119]. Função reLU.

Verifica-se que a derivada é a função degrau. Uma variante utilizada é a *Leaky ReLU* dada por

$$g(x) = \begin{cases} x, & x \geq 0 \\ \alpha x, & x < 0 \end{cases}$$

Como padrão, utiliza-se $0 \leq \alpha < 1$ [115]. Geralmente utiliza-se a reLU em todas as camadas menos na última em que usamos a função *softmax* quando temos um caso de classificação multiclasse e é dada por

$$g(x_i) = \frac{e^{x_i}}{\sum_i e^{x_i}}$$

que dará a probabilidade do ponto pertencer a classe *k* quando usada na unidade de ativação do output $a_k^{(L)}$ onde *L* é o número de camadas da rede [104]. Lembrando que $a_k^{(L)} = g(z_k^{(L-1)})$:

$$a_k^{(L)} = \frac{\exp\left(z_k^{(L-1)}\right)}{\sum_{i=1}^{d_{L-1}} \exp\left(z_i^{(L-1)}\right)}$$

C.5 Redes Neurais Convolucionais

C.5.1 Camada convolucional

As imagens podem ser representadas por três matrizes RGB, neste caso os filtros terão 3 dimensões:

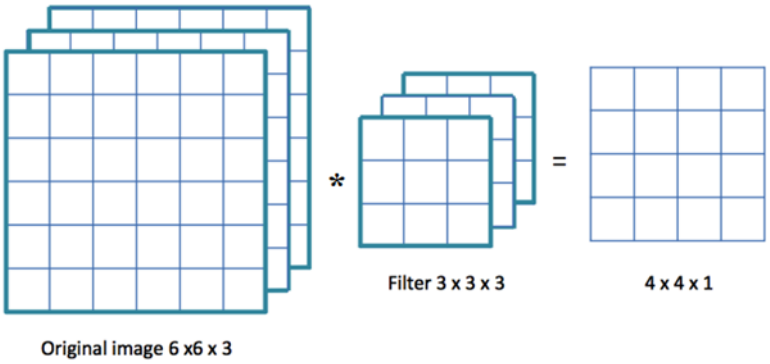


Figura 25 – Filtro e imagem 3-D [108].

Podemos pensar nesta operação da mesma forma que anteriormente mas com três imagens separadas e três filtros separados, um para cada cor. Os resultados seriam as 3 matrizes 4×4: $z^{(R)}, z^{(G)}, z^{(B)}$. Criamos uma única matriz 4×4 que recebe os valores somados das três matrizes em cada linha i e coluna j: $z_{ij} = z_{ij}^{(R)} + z_{ij}^{(G)} + z_{ij}^{(B)}$. Podemos generalizar a fórmula (C.4) para este caso adicionando um índice em w e x e somar neste índice:

$$z_{ij} = \sum_{r=1}^{3}\sum_{k=1}^{3}\sum_{l=1}^{3} w_{klr}\, x_{k+i-1,l+j-1,r} \,. \qquad \text{(C.5)}$$

Neste caso os índices i e j vão até 3 pois é a largura do filtro $(\nu = 3)$ e r vai até 3 que é a profundidade de x representando as cores RGB. Vimos que os filtros são matrizes com pesos w como no caso dos pesos entre as camadas densas das redes neurais convencionais, ou seja, os filtros são compostos por parâmetros aprendidos pela rede através da minimização da função de custo e retropropagação com descida de gradiente ou outros algoritmos como Adam [101]. Os filtros aprendem a detectar características importantes da imagem. Abaixo

exemplificamos um filtro detector de linhas verticais e horizontais:

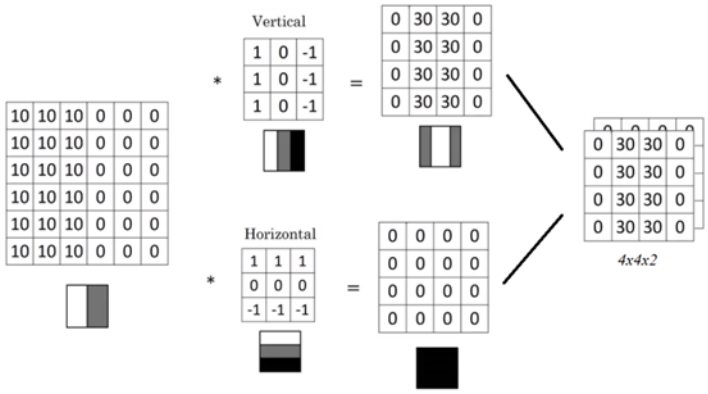


Figura 26 – Detectores de extremidades verticais e horizontais [111].

Cada camada convolucional pode conter vários filtros e as nova matrizes bidimensionais agrupam-se na terceira dimensão cuja profundidade é o número de filtros aplicados na Primeira camada $\eta^{(1)}$. Na figura 26 vemos 2 filtros gerando duas novas matrizes 2D que são unidas na terceira dimensão.

Vamos redefinir a notação dos filtros na forma $w_{klr}^{(u,q)}$ onde u representa a camada convolucional e q idêntica qual o filtro da camada (onde $q \in \{1,2,...,\eta^{(1)}\}$). No caso, com os pesos da Primeira camada temos:

$$z_{ijq} = \sum_{r=1}^{\eta^{(1)}} \sum_{k=1}^{\nu} \sum_{l=1}^{\nu} w_{klr}^{(1,q)} x_{k+i-1,l+j-1,r} \qquad \text{(C.6)}$$

A última operação é somar um *bias* aplicar a função de ativação *g* obtendo a matriz de ativação (semelhante as unidades de ativação em MLP): $a_{ijq} = g\left(z_{ijq} + b_q\right)$. Para aplicarmos uma nova camada convolucional vamos redefinir as matrizes *a* e *z* na forma $a_{ijq}^{(u)}$ e $z_{ijq}^{(u)}$ onde *u* representa a camada, assim $a_{ijq}^{(1)} = x_{ijq}$. Generalizando a expressão (C.6) para outras camadas:

$$a_{ijq}^{(u+1)} = g\left(\sum_{r=1}^{\eta^{(u)}} \sum_{k=1}^{\nu^{(u)}} \sum_{l=1}^{\nu^{(u)}} w_{klr}^{(u,q)} a_{k+i-1,l+j-1,r}^{(u)} + b_r^{(u,q)} \right). \qquad \text{(C.7)}$$

A dimensão de $a_{ijq}^{(u+1)}$ é

$$\zeta^{(u+1)} \times \zeta^{(u+1)} \times \chi^{(u+1)} = \left(\zeta^{(u)} - \nu^{(u)} + 1\right) \times \left(\zeta^{(u)} - \nu^{(u)} + 1\right) \times \eta^{(u)}.$$

Como os filtros de camadas diferentes podem ter tamanhos diferentes incluimos a

notação $\nu^{(u)}$ que é a altura/largura dos filtros na camada *u*. Os filtros da mesma camada tem a mesma dimensão. O termo $\chi^{(u)}$ representa a profundidade da matriz de ativação da camada *u* e é igual ao número de filtros na camada anterior: $\chi^{(u+1)} = \eta^{(u)} \mid u > 0$.

C.5.2 Padding

As informações nas extremidades da imagem são utilizadas menos vezes que as informações no centro da imagem. Por exemplo, o Primeiro elemento da matriz de input só entra na Primeira submatriz enquanto que um pixel na região central pode ser contado em várias submatrizes. Para resolver este problema utiliza-se uma técnica chamada preenchimento (*padding*) aplicada antes da operação de convolução e consiste em envolver a matriz com zeros:

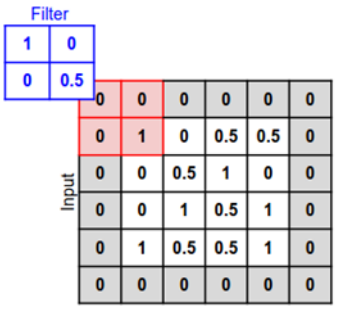


Figura 27 – Preenchimento de uma matriz [112].

O preenchimento da Figura 27 é dado por $\varphi^{(u)} = 1$ pois este preencheu uma coluna a mais a esquerda e a direita e uma linha a mais abaixo e acima. Sabemos que a largura $a^{(u)}$ de é $\left(\zeta^{(u)} - \nu^{(u)} + 1\right)$. O preenchimento da matriz $a^{(u)}$ gera uma nova matriz $\tilde{a}^{(u)}$ com dimensão: $\tilde{\zeta}^{(u)} \times \tilde{\zeta}^{(u)} \times \tilde{\chi}^{(u)} = \left(\zeta^{(u)} + 2\varphi^{(u)}\right) \times \left(\zeta^{(u)} + 2\varphi^{(u)}\right) \times \chi^{(u)}$ com $\tilde{\zeta}^{(u)} = \zeta^{(u)} + 2\varphi^{(u)}$ e $\tilde{\chi}^{(u)} = \chi^{(u)}$.

Os frameworks de *deep learning* como Keras [69] e TensorFlow [107] possuem as opções de preenchimento "valid" e "same". A opção "valid" resulta em não preenchimento e a opção "same" retorna um valor para $\varphi^{(u)}$ de forma que a nova matriz $\tilde{a}^{(u)}$ ao ser convoluída com os filtros $w^{(u,q)}$ resulte em uma matriz $a^{(u+1)}$ com mesma dimensão que a matriz original $a^{(u)}$. A dimensão de $a^{(u+1)}$ é:

$$\zeta^{(u+1)} \times \zeta^{(u+1)} \times \chi^{(u+1)} = \left(\zeta^{(u)} + 2\varphi^{(u)} - \nu^{(u)} + 1\right) \times \left(\zeta^{(u)} + 2\varphi^{(u)} - \nu^{(u)} + 1\right) \times \eta^{(u)}$$
$$= \left(\tilde{\zeta}^{(u)} - \nu^{(u)} + 1\right) \times \left(\tilde{\zeta}^{(u)} - \nu^{(u)} + 1\right) \times \eta^{(u)}$$

Para que a área de $a^{(u+1)}$ seja igual a de $a^{(u)}$:

$$\zeta^{(u+1)} = \zeta^{(u)} \rightarrow \zeta^{(u)} - \nu^{(u)} + 2\tilde{\varphi}^{(u)} + 1 = \zeta^{(u)} \rightarrow \tilde{\varphi}^{(u)} = \frac{\nu^{(u)} - 1}{2}.$$

Para $\nu^{(u)}$ ímpar maior que 2 e $\tilde{\varphi}$ representa o valor que retorna a opção "same". Vemos que a figura 27 representa a opção "same" e será a opção utilizada. Exemplificando com $\nu^{(u)} = 3$ temos $\varphi^{(u)} = 1$:

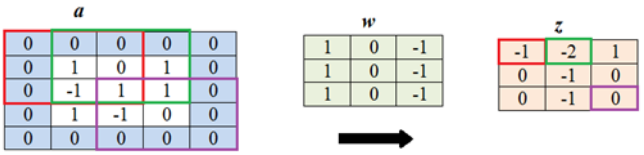


Figura 28 – Representação da convolução de um filtro com uma matriz com preenchimento $\varphi^{(u)} = 1$.

Para mapear a matriz $\tilde{a}^{(u)}$ para um dado $\varphi^{(u)}$, seus elementos serão:

$$\tilde{a}_{ijq}^{(u)} = \begin{cases} a_{i-\varphi^{(u)}, j-\varphi^{(u)}, q}^{(u)}, & \varphi^{(u)} < i, j < \zeta^{(u)} + \varphi^{(u)} + 1 \\ 0, & \textit{senão} \end{cases}$$

Por exemplo, na seguinte matriz $a^{(u)}$ bidimensional com $\zeta^{(u)} = 3$ e $\varphi^{(u)} = 2$:

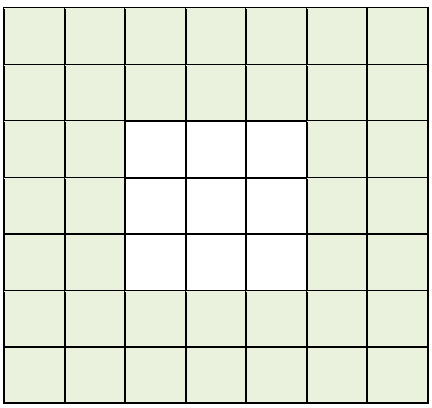


Figura 29 – Preenchimento $\varphi^{(u)} = 2$ de uma matriz com $\zeta^{(u)} = 3$.

Neste caso os elementos são dados por:

$$\tilde{a}_{ij}^{(u)} = \begin{cases} a_{i-2,j-2}^{(u)}, & 2 < i, j < 6 \\ 0, & senão \end{cases}$$

O Primeiro elemento obtido da matriz $a^{(u)}$ é justamente $a_{11}^{(u)}$ mapeado em $\tilde{a}_{33}^{(u)}$. A operação de convolução (C.7) é redefinida na forma:

$$a_{ijq}^{(u+1)} = g\left(\sum_{r=1}^{\eta^{(u)}}\sum_{k=1}^{\nu^{(u)}}\sum_{l=1}^{\nu^{(u)}} w_{klr}^{(u,q)}\tilde{a}_{k+i-1,l+j-1,r}^{(u)} + b_r^{(u,q)}\right) \tag{C.8}$$

com os índices são $i, j \in \left\{1, 2, ..., \tilde{\zeta}^{(u)} - \nu + 1\right\}$.

C.5.3 Stride

O tamanho dos passos também é um parâmetro das camadas convolucionais e é chamada de *stride* denotada por ρ. As camadas convolucionais apresentadas anteriormente representam um $\rho = 1$. Para $\rho = 2$ os filtros andam dois passos na matriz antes de convoluir, abaixo visualizamos:

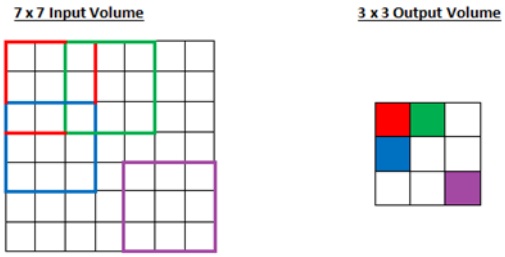


Figura 30 – Um passo de tamanho 2 representado em cores diferentes nas regiões de convolução de um filtro 3×3 em uma matriz 7×7 retornando uma nova matriz 3×3.

A largura da nova matriz é dada por:

$$\zeta^{(u+1)} = \frac{\zeta^{(u)} + 2\tilde{\varphi}^{(u)} - \nu^{(u)}}{\rho^{(u)}} + 1 = \frac{\tilde{\zeta}^{(u)} - \nu^{(u)}}{\rho^{(u)}} + 1 \tag{C.9}$$

O parâmetro $\tilde{\varphi}$ (para *padding* "same") é redefinido:

$$\zeta^{(u+1)} = \zeta^{(u)} \rightarrow \frac{\zeta^{(u)} + 2\tilde{\varphi}^{(u)} - \nu^{(u)}}{\rho^{(u)}} + 1 = \zeta^{(u)} \quad \rightarrow \quad \tilde{\varphi}^{(u)} = \frac{\nu^{(u)} - \zeta^{(u)} + \zeta^{(u)}\rho^{(u)} - \rho^{(u)}}{2}.$$

Onde $\tilde{\varphi}$ é inteiro positivo.

A operação de convolução (C.8) é modificada de forma a incluir o tamanho dos passos

$i-1 \to \rho^{(u)}(i-1)$.

$$a_{ijq}^{(u+1)} = g\left(\sum_{r=1}^{\eta^{(u)}} \sum_{k=1}^{\nu^{(u)}} \sum_{l=1}^{\nu^{(u)}} w_{klr}^{(u,q)} \tilde{a}^{(u)}_{k+\rho^{(u)}(i-1),l+\rho^{(u)}(i-1),r} + b_r^{(u,q)} \right) \tag{C.10}$$

com os índices $i, j \in \left\{1, 2, ..., \frac{\tilde{\zeta}^{(u)} - \nu^{(u)}}{\rho^{(u)}} + 1\right\}$.

Para visualizarmos melhor a fórmula da convolução (C.10) vamos aplicá-la no exemplo da Figura 28 onde $\rho = 1$ e na Figura 30 com $\rho = 2$, como as matrizes e os filtros são 2D suPrimimos a terceira dimensão (índices *r* e *q*): $a_{ij}^{(u+1)} = g\left(\sum_{k=1}^{3} \sum_{l=1}^{3} w_{kl}^{(u)} \tilde{a}^{(u)}_{k+\rho^{(u)}(i-1),l+\rho^{(u)}(j-1)} + b^{(u)} \right) = g\left(z_{ij}^{(u)} + b^{(u)} \right)$.

Vamos analisar apenas o elemento $z_{ij} = \sum_{k=1}^{3} \sum_{l=1}^{3} w_{kl} \tilde{a}_{k+\rho(i-1),l+\rho(j-1)}$ que é o output da aplicação do filtro na matriz de ativação (suPrimindo a indicação da camada *u*). O Primeiro elemento é z_{11} (destacado em vermelho) e é independente do *stride* pois os termos *i*-1 e *j*-1 anulam-se: $z_{11} = \sum_{k=1}^{3} \sum_{l=1}^{3} w_{kl} \tilde{a}_{kl} = w_{11}\tilde{a}_{11} + w_{12}\tilde{a}_{12} + ... + w_{33}\tilde{a}_{33}$.

O elemento na Primeira linha e segunda coluna (destacado em verde) será diferente. Para $\rho = 1$:

$$z_{12} = \sum_{k=1}^{3} \sum_{l=1}^{3} w_{kl} \tilde{a}_{k,l+1} = w_{11}\tilde{a}_{12} + w_{12}\tilde{a}_{13} + ... + w_{33}\tilde{a}_{34}$$

e $\rho = 2$:

$$z_{12} = \sum_{k=1}^{3} \sum_{l=1}^{3} w_{kl} \tilde{a}_{k,l+2} = w_{11}\tilde{a}_{13} + w_{12}\tilde{a}_{14} + ... + w_{33}\tilde{a}_{35}.$$

A janela de seleção dos elementos foi movida ρ passos para a direita.

Verificando a expressão (C.9) vemos que o último output de ambas matrizes é z_{33} (em roxo). Para $\rho = 1$:

$$z_{33} = \sum_{k=1}^{3} \sum_{l=1}^{3} w_{kl} \tilde{a}_{k+2,l+2}$$

e $\rho = 2$ temos:

$$z_{33} = \sum_{k=1}^{3} \sum_{l=1}^{3} w_{kl} \tilde{a}_{k+4,l+4}.$$

Neste caso vemos que a janela de seleção dos elementos foi movida 2ρ passos para a direita e para baixo.

C.5.4 Camadas de agrupamento

As camadas de agrupamento mais comuns são as camadas de agrupamento de máximo e média:

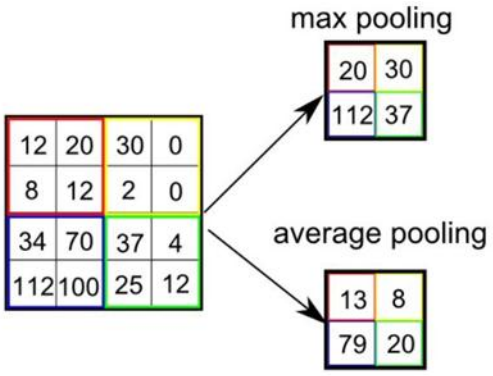


Figura 31 – Operações de agrupamento da célula maior (*max pooling*) e média das células (*average pooling*).

A operação de agrupamento mais comum utilizada em reconhecimento de imagens é a de agrupamento de máximo (*max pooling*) [109]. Vemos na Figura 31 as submatrizes destacados com cores diferentes e percebemos que as submatrizes tem tamanho dado pelo tamanho do agrupamento (*pool size*) $\xi = 2$ e o tamanho do passo para a seleção da próxima submatriz (*stride*) é $\rho = 2$. O comprimento da antiga matriz é ζ e a nova matriz é: $\frac{\zeta - \xi}{\rho} + 1$.

Geralmente não utiliza-se preenchimento nesta camada e os parâmetros sempre são $\xi = \rho = 2$ por convenção, a consequência é uma matriz com área reduzida pela metade como vemos na figura 31 onde a matriz 4×4 reduziu-se para 2×2. A camada de agrupamento não tem pesos a serem minimizados por uma função de custo e não são introduzidas em uma função de ativação.

Podemos representar matematicamente esta operação da seguinte maneira:

$$A_{ij}^{(u)} = \max\left\{a_{i+k-1,\,j+l-1}^{(u)}\right\} \quad \forall 1 \le k,l \le \xi \text{ e } \forall 1 \le i,j \le \frac{\zeta}{2}.$$

Com esta definição, as imagens devem ter altura e largura pares. Visualizando o Primeiro elemento na Figura 31, $i = j = 1$ e

$$1 \le k,l \le 2 : A_{11}^{(u)} = \max\left\{a_{1+k-1,1+l-1}^{(u)}\right\} = \max\left\{a_{11}^{(u)}, a_{12}^{(u)}, a_{21}^{(u)}, a_{22}^{(u)}\right\} = \max\{12,20,8,12\} = 20.$$

Na estrutura clássica de uma rede neural convolucional, passa-se várias camadas convolucionais (com preenchimento "same") e camadas de agrupamento de máximos com os parâmetros convencionados.

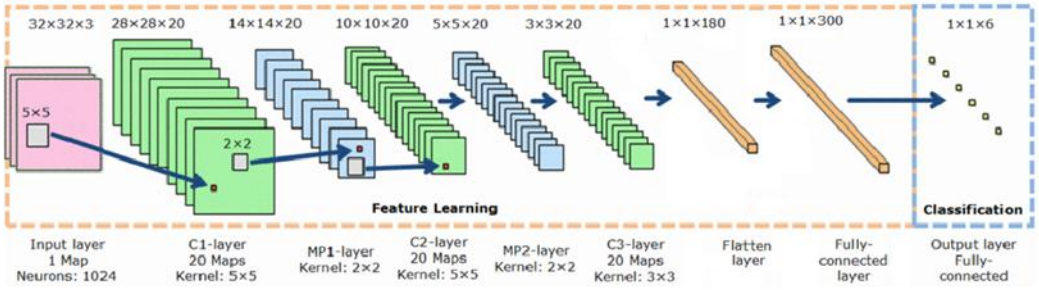


Figura 32 – Camadas convolucionais (verde) intercaladas com camadas de agrupamento (azul).

Na figura 32 vemos que as camadas convolucionais aumentam a profundidade das matrizes enquanto que a camada de agrupamento reduz suas áreas pela metade durante a fase de aprendizagem de características. Após a terceira camada convolucional (C3-layer), a matriz tem um tamanho $3\times3\times20$ e é achatada de forma a tornar-se uma matriz $1\times1\times180$ (laranja), esta é conectada em uma segunda camada densa (como nas MLP) com dimensão $1\times1\times300$ e a última camada com dimensão $1\times1\times6$ dando a probabilidade entre 6 diferentes classe dada pela função de ativação *softmax*. Vemos que o achatamento da última camada convolucional gera dali em diante uma rede neural idêntica a MLP com três camadas, de fato as camadas convolucionais intercaladas com as camadas de agrupamento servem para identificar as características para a classificação.

Nota-se também que a Primeira sequencia de camadas de convolução e agrupamento são denotadas por C1-layer e MP1-layer e a segunda sequência é C2-layer e MP2-layer. Isto representa uma convenção, a próxima camada convolucional sempre é contada apenas após a aplicação da camada de agrupamento. O input será contado como a camada u=1 e as camadas C1 e MP1 serão u=2 e assim por diante.

Após a aplicação da camada de agrupamento, preenche-se agora a matriz reduzida A obtendo $\tilde{A}$ (utilizando a opção de *padding* "same") e em seguida aplica-se a próxima camada convolucional, a equação (C.10) fica:

$$a_{ijq}^{(u+1)} = g\left(\sum_{r=1}^{\eta^{(u)}}\sum_{k=1}^{\nu^{(u)}}\sum_{l=1}^{\nu^{(u)}} w_{klr}^{(u,q)}\tilde{A}^{(u)}_{k+\rho^{(u)}(i-1),l+\rho^{(u)}(i-1),r} + b_r^{(u,q)}\right) \tag{C.11}$$

com os índices $i, j \in \left\{1,2,...,\frac{\tilde{\tau}^{(u)}-\nu^{(u)}}{\rho^{(u)}}+1\right\}$. Aqui introduzimos o símbolo $\tau^{(u)}$ que é a largura de $A^{(u)}$, portanto $\tau^{(u)} = \zeta^{(u)}/2$. O símbolo $\tilde{\tau}^{(u)}$ é a largura de $\tilde{A}^{(u)}$.

Após aplicar a próxima camada de convolução em $\tilde{A}^{(u)}$, a largura da nova matriz $a^{(u+1)}$ é dada pela equação (C.9) modificada:

$$\zeta^{(u+1)} = \frac{\tau^{(u)}+2\tilde{\varphi}^{(u)}-\nu^{(u)}}{\rho^{(u)}}+1 = \frac{\tilde{\tau}^{(u)}-\nu^{(u)}}{\rho^{(u)}}+1 .$$

Onde $\tau^{(u)}+2\tilde{\varphi}^{(u)} = \tilde{\tau}^{(u)}$. A expressão para o parâmetro $\tilde{\varphi}$ (*padding* "same") será:

$$\zeta^{(u+1)} = \tau^{(u)} \quad \rightarrow \quad \frac{\zeta^{(u)}/2 + 2\tilde{\varphi}^{(u)} - \nu^{(u)}}{\rho^{(u)}} + 1 = \zeta^{(u)}/2$$

$$\tilde{\varphi}^{(u)} = \frac{2\nu^{(u)} - \zeta^{(u)} + \zeta^{(u)}\rho^{(u)} - 2\rho^{(u)}}{4} = \frac{\nu^{(u)} - \tau^{(u)} + \tau^{(u)}\rho^{(u)} - \rho^{(u)}}{2}$$

Onde $\tilde{\varphi}$ é inteiro positivo.

A rede representada reduziu o número de inputs de $32\times32\times3 = 3072$ para 180. Este é um dos motivos do qual a CNN é superior a MLP para problemas em grande escala e difícil identificação das características. A CNN consegue ser mais eficiente que a MLP por possuir uma propriedade chamada *compartilhamento de parâmetros* [110], um filtro detector de uma característica como extremidades verticais e horizontais como da Primeira camada convolucional na figura 33 é utilizado em várias partes da imagem.

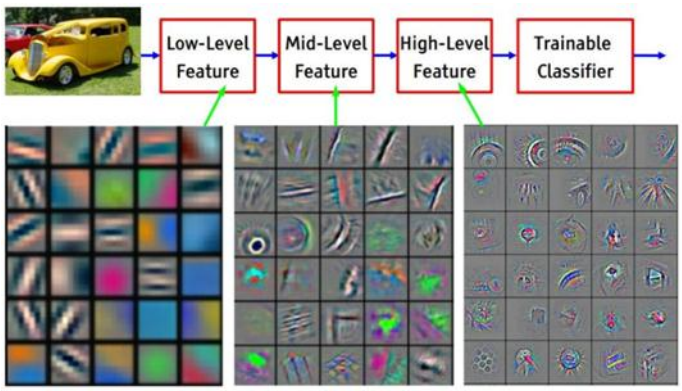


Figura 33 – Três camadas convolucionais com vários filtros [106].

Vemos na figura 33 que a Primeira camada de filtros aprendeu a detectar linhas verticais, horizontais e diagonais além de alguns padrões de cores e as outras camadas aprendem padrões bem mais complicados.

C.5.5 Retropropagação em camadas convolucionais

Para definirmos a retropropagação nas CNN vamos inicialmente exemplificar com uma matriz $x_{3\times3}$ e um filtro $w_{2\times2}$ (não usual). Sabemos que:

$$z_{11} = w_{11}x_{11} + \ldots + w_{22}x_{22}$$
$$\vdots$$
$$z_{22} = w_{11}x_{22} + \ldots + w_{22}x_{33}$$

Ou simplesmente $z_{ij} = \sum_{k=1}^{2}\sum_{l=1}^{2} w_{kl}x_{k+(i-1),l+(i-1)}$.

Os gradientes do erro em relação aos pesos serão:

$$\frac{\partial E}{\partial w_{11}} = \frac{\partial E}{\partial z_{11}}\frac{\partial z_{11}}{\partial w_{11}} + \ldots + \frac{\partial E}{\partial z_{22}}\frac{\partial z_{22}}{\partial w_{11}}$$
$$\vdots$$
$$\frac{\partial E}{\partial w_{22}} = \frac{\partial E}{\partial z_{11}}\frac{\partial z_{11}}{\partial w_{22}} + \ldots + \frac{\partial E}{\partial z_{22}}\frac{\partial z_{22}}{\partial w_{22}}$$

Que serão:

$$\frac{\partial E}{\partial w_{11}} = \frac{\partial E}{\partial z_{11}}x_{11} + \ldots + \frac{\partial E}{\partial z_{22}}x_{22}$$
$$\vdots$$
$$\frac{\partial E}{\partial w_{22}} = \frac{\partial E}{\partial z_{11}}x_{22} + \ldots + \frac{\partial E}{\partial z_{22}}x_{33}$$

Que é semelhante a operação de convolução:

$$\frac{\partial E}{\partial w_{ij}} = \sum_{k=1}^{2}\sum_{l=1}^{2}\frac{\partial E}{\partial z_{ij}}x_{k+(i-1),l+(i-1)} \quad . \qquad \text{(C.12)}$$

Vamos analisar a retropropagação na seguinte rede:

$$\tilde{A}_{6\times6}^{(L-3)} \xrightarrow[w_{3\times3}^{(L-3)}]{conv} a_{4\times4}^{(L-2)} \xrightarrow[flatten]{pool:A_{2\times2}^{(L-2)}} A_{4\times1}^{(L-2)} \xrightarrow[w^{(L-2)}]{dense} a_{2\times1}^{(L-1)} \xrightarrow[w^{(L-1)}]{class} a^{(L)} = \hat{y} \qquad \text{(C.13)}$$

As três últimas camadas são densas e a antepenúltima foi achatada de forma a tornar-se densa. Se quiser o erro em relação a um elemento da matriz $w^{(L-3)}$ fazemos a regra da cadeia como caso dos MLP:

$$\frac{\partial E}{\partial w_{ij}^{(L-3)}} = \frac{\partial E}{\partial a_1^{(L)}} \frac{\partial a_1^{(L)}}{\partial a^{(L-1)}} \frac{\partial a^{(L-1)}}{\partial A^{(L-2)}} \frac{\partial A^{(L-2)}}{\partial a^{(L-2)}} \frac{\partial a^{(L-2)}}{\partial w_{ij}^{(L-3)}}$$

A derivada de uma matriz *A* em relação a uma matriz *a* da mesma camada é definida como 1 (as camadas de agrupamento não possuem pesos, então pulamos estas na retropropagação): $\frac{\partial A^{(u)}}{\partial a^{(u)}} \equiv 1$. Se a camada for preenchida também: $\frac{\partial \tilde{A}^{(u)}}{\partial a^{(u)}} \equiv 1$.

O último termo é:

$$\frac{\partial a^{(L-2)}}{\partial w_{rq}^{(L-3)}} = \frac{\partial a^{(L-2)}}{\partial z_{ij}^{(L-3)}} \frac{\partial z_{ij}^{(L-3)}}{\partial w_{rq}^{(L-3)}} = \frac{\partial a^{(L-2)}}{\partial z_{ij}^{(L-3)}} \tilde{A}_{k+(i-1),l+(j-1)}^{(L-3)} = g'^{(u-1)} \tilde{A}_{k+(i-1),l+(j-1)}^{(L-3)}$$

Onde definimos a notação $\frac{\partial a^{(u)}}{\partial z_{ij}^{(u-1)}} = g'\left(z^{(u-1)} + b^{(u-1)}\right) \equiv g'^{(u-1)}$. Os elementos de $\tilde{A}^{(L-3)}$ são obtidos de maneira semelhante a (C.12) utilizando a expressão (C.11). Portanto:

$$\frac{\partial E}{\partial w_{ij}^{(L-3)}} = \frac{\partial E}{\partial a_1^{(L)}} \frac{\partial a_1^{(L)}}{\partial a^{(L-1)}} \frac{\partial a^{(L-1)}}{\partial A^{(L-2)}} g'^{(L-3)} \tilde{A}_{k+(i-1),l+(j-1)}^{(L-3)}$$

Retroagindo mais camadas na rede (C.13):

$$\tilde{A}_{16\times16}^{(L-4)} \xrightarrow[w_{3\times3}^{(L-4)}]{conv} a_{8\times8}^{(L-3)} \xrightarrow[pad:\varphi=1]{pool:A_{4\times4}^{(L-3)}} \tilde{A}_{6\times6}^{(L-3)}$$

Queremos obter a derivada do termo $w_{ij}^{(L-4)}$:

$$\frac{\partial E}{\partial w_{ij}^{(L-4)}} = \frac{\partial E}{\partial a_1^{(L)}} \frac{\partial a_1^{(L)}}{\partial a^{(L-1)}} \frac{\partial a^{(L-1)}}{\partial A^{(L-2)}} \frac{\partial a^{(L-2)}}{\partial \tilde{A}^{(L-3)}} \frac{\partial a^{(L-3)}}{\partial z_{rq}^{(L-4)}} \frac{\partial z_{rq}^{(L-4)}}{\partial w_{ij}^{(L-4)}} =$$

$$= \frac{\partial E}{\partial a_1^{(L)}} \frac{\partial a_1^{(L)}}{\partial a^{(L-1)}} \frac{\partial a^{(L-1)}}{\partial A^{(L-2)}} \frac{\partial a^{(L-2)}}{\partial \tilde{A}^{(L-3)}} g'^{(L-4)} \tilde{A}_{k+(i-1),l+(j-1)}^{(L-4)}$$

Podemos escrever como uma regra geral :

$$\frac{\partial E}{\partial w_{ijr}^{(u,q)}} = \frac{\partial E}{\partial a^{(L)}} \frac{\partial a_1^{(L)}}{\partial a^{(L-1)}} \frac{\partial a^{(L-1)}}{\partial a^{(L-2)}} \cdots g'^{(u,q)} \tilde{A}_{k+\rho^{(u)}(i-1),l+\rho^{(u)}(i-1),r}^{(u)}$$

Antes de cada operação de convolução no input e nas matriz de ativação $\tilde{A}^{(u)}$ faz-se a normalização de lotes e otimiza-se as os pesos *w* através do algoritmo Adam em mini-lotes [45,46].

C.5.6 Redes neurais convolucionais unidimensionais

Para a classificação de espectros iremos utilizar as CNN em 1D (tratando o espectro como uma imagem unidimensional). É mais intuitivo a aplicação de CNN em imagens, por isso começamos com exemplos em imagens 2D (não contando a terceira dimensão que inicialmente representa as 3 cores RGB e nas outras camadas representam o número de filtros da camada anterior).

As CNN unidimensionais são idênticas as CNN bidimensionais para imagens quadradas vistas anteriormente mas com uma dimensão reduzida. Abaixo vemos um exemplo de uma CNN 1D:

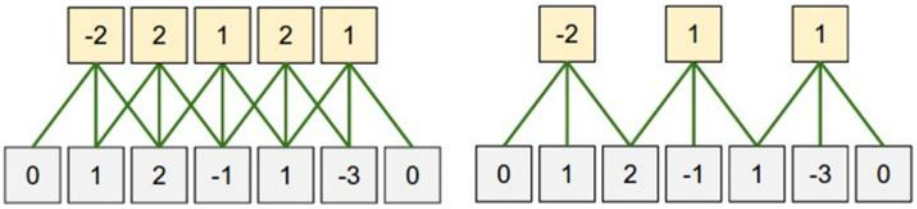


Figura 34 – Convolução com $\rho = 1$ (esquerda) e $\rho = 2$ (direita).

Na figura 34 vemos que a entrada é o vetor $x = [1 \quad 2 \quad -1 \quad 1 \quad -3]$ com um preenchimento $\varphi = 1$ e um filtro dado por $w = [1 \quad 0 \quad -1]$ e um passo de tamanho $\varphi = 1$ gerando um output de tamanho 5 e com um passo $\varphi = 2$ gerando um novo output com tamanho 2.

A fórmula para convolução agora será semelhante a (C.11) suPrimindo uma coordenada:

$$a_{iq}^{(u+1)} = g\left(\sum_{r=1}^{\eta^{(u)}}\sum_{k=1}^{\nu^{(u)}} w_{kr}^{(u,q)} \tilde{A}_{k+\rho^{(u)}(i-1),r}^{(u,q)} + b_r^{(u,q)}\right)$$

Com

$$i \in \left\{1, 2, ..., \frac{\tilde{\tau}^{(u)} - \nu^{(u)}}{\rho^{(u)}} + 1\right\}.$$

Esta expressão aplicada em mini-lotes será dada por

$$a_{iql}^{(u+1)} = g\left(\sum_{r=1}^{\eta^{(u)}}\sum_{k=1}^{\nu^{(u)}} w_{kr}^{(u,q)} \tilde{A}_{k+\rho^{(u)}(i-1),r,l}^{(u,q)} + b_r^{(u,q)}\right)$$

Onde adicionamos um índice $l \in \{1, 2, ..., n\}$ correspondendo as *n* imagens de cada mini-lote.

C.5.7 Arquitetura da CNN aplicada em Raman

Definimos uma rede com a três camadas convolucionais com 50, 75 e 100 filtros respectivamente intercaladas com uma camada de agrupamento de máximo após a Primeira camada convolucional e uma após a terceira. Após a última camada de agrupamento teremos uma matriz com tamanho $350\times100\times3000$ que é achatada formando uma matriz $35000\times1\times3000$ que é conectada a uma rede neural densa (MLP) com duas camadas entre a matriz achatada e o output de dimensão $6\times1\times3000$ indicando as classes dos 3000 espectros. As duas camadas ocultas possuem 600 e 300 unidades de ativação com chance de manter os nós de 50% e 40% respectivamente. Podemos representar no seguinte diagrama [73]:

Camada	Função de Ativação	Dimensão de saída
Convolucional 1	*Leaky ReLU* ($\alpha = 0,2$)	$1400\times50\times3000$
Agrupamento de máximo 1	-	$700\times50\times3000$
Convolucional 2	*Leaky ReLU* ($\alpha = 0,2$)	$700\times75\times3000$
Convolucional 3	*Leaky ReLU* ($\alpha = 0,2$)	$700\times100\times3000$
Agrupamento de máximo 2	-	$350\times100\times3000$
Achatamento	-	$35000\times1\times3000$
Densa 1 com *dropout*: 50%	*Leaky ReLU* ($\alpha = 0,2$)	$35000\times1\times3000$
Densa 2 com *dropout*: 40%	*Leaky ReLU* ($\alpha = 0,2$)	$35000\times1\times3000$
Densa 3	*Softmax*	$6\times1\times3000$

Tabela 3 – Arquitetura da rede neural convolucional

Treinamos a rede utilizando com validação 15% dos dados de treino selecionados aleatoriamente em cada mini lote da época. Utilizamos mini lotes com 64 espectros treinados em 400 épocas. Abaixo visualizamos a variação na função de custo e na acurácia do modelo obtidos no *TensorBoard* que é um visualizador dos dados da rede neural integrado ao *TensorFlow* [107]:

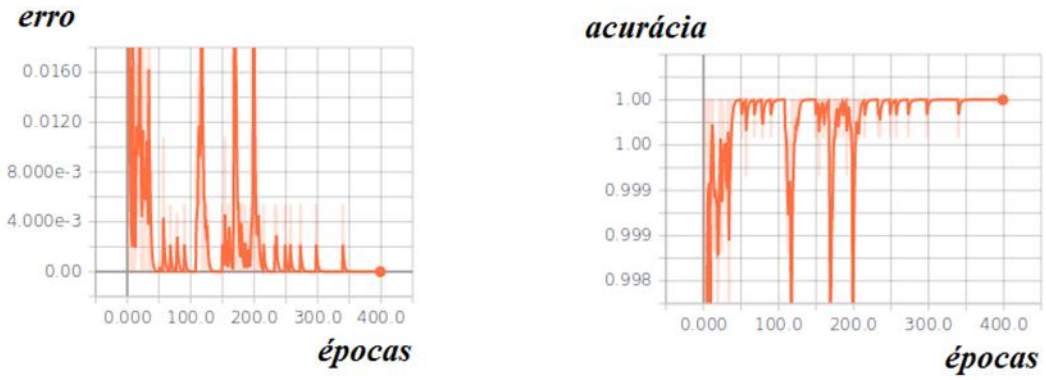


Figura 35 – Função erro e acurácia ao logo das épocas de treino.

ANEXO D – EQUAÇÃO DE B&S E EQUAÇÃO DA DIFUSÃO

D.1. Solução da equação diferencial de Black & Scholes

Verifica-se que a fórmula de Black&Scholes é:

$c = S\Phi(d_1) - Xe^{-r(T-t)}\Phi(d_2)$

onde com $d_1 = \dfrac{\ln\left(\dfrac{S}{X}\right) + \left(r + \dfrac{\sigma^2}{2}\right)(T-t)}{\sigma\sqrt{(T-t)}}$ e $d_2 = \dfrac{\ln\left(\dfrac{S}{X}\right) + \left(r - \dfrac{\sigma^2}{2}\right)(T-t)}{\sigma\sqrt{(T-t)}}$ com T definido em $t = 0$ e $0 \le t \le T$, logo $T - t$ é o tempo que falta para chegar à maturidade, satisfaz à equação diferencial estocástica:

$$\frac{\partial c}{\partial t} + rS\frac{\partial c}{\partial S} + \frac{\sigma^2 S^2}{2}\frac{\partial^2 c}{\partial S^2} - rc = 0$$

e as condições iniciais:

$c(0,t) = 0$ e $c(S_T, T) = Max[S_T - X, 0]$

Melhor separar as dependências temporais de d_1 e d_2 na forma:

$$d_1 = \frac{\ln\left(\dfrac{S}{X}\right)}{\sigma\sqrt{(T-t)}} + \frac{\left(r + \dfrac{\sigma^2}{2}\right)}{\sigma}\sqrt{(T-t)}$$

$$d_2 = \frac{\ln\left(\dfrac{S}{X}\right)}{\sigma\sqrt{(T-t)}} + \frac{\left(r - \dfrac{\sigma^2}{2}\right)}{\sigma}\sqrt{(T-t)}$$

escrever

Para isso vamos começar mostrando que $Se^{-\frac{d_1^2}{2}} = Xe^{-\frac{d_2^2}{2}}e^{-r(T-t)}$:

$$\frac{d_1^2}{2} = \frac{\ln^2\left(\dfrac{S}{X}\right)}{2\sigma^2(T-t)} + \frac{\left(r + \dfrac{\sigma^2}{2}\right)}{\sigma^2}\ln\left(\frac{S}{X}\right) + \frac{\left(r + \dfrac{\sigma^2}{2}\right)^2}{2\sigma^2}(T-t)$$

$$\frac{d_1^2}{2} = \frac{1}{2}\ln\left(\frac{S}{X}\right) + \frac{r}{2}(T-t) + \frac{r}{\sigma^2}\ln\left(\frac{S}{X}\right) + \frac{\ln^2\left(\dfrac{S}{X}\right)}{2\sigma^2(T-t)} + \frac{\left(r^2 + \dfrac{\sigma^4}{4}\right)}{2\sigma^2}(T-t)$$

$$\frac{d_2^2}{2} = \frac{\ln^2\left(\dfrac{S}{X}\right)}{2\sigma^2(T-t)} + \frac{\left(r - \dfrac{\sigma^2}{2}\right)}{\sigma^2}\ln\left(\frac{S}{X}\right) + \frac{\left(r - \dfrac{\sigma^2}{2}\right)^2}{2\sigma^2}(T-t)$$

$$\frac{d_2^2}{2} = -\frac{1}{2}\ln\left(\frac{S}{X}\right) - \frac{r}{2}(T-t) + \frac{r}{\sigma^2}\ln\left(\frac{S}{X}\right) + \frac{\ln^2\left(\frac{S}{X}\right)}{2\sigma^2(T-t)} + \frac{\left(r^2+\frac{\sigma^4}{4}\right)}{2\sigma^2}(T-t)$$

$$Se^{-\frac{d_1^2}{2}} = Se^{-\frac{1}{2}\ln\left(\frac{S}{X}\right)}e^{-\frac{r}{2}(T-t)}e^{-\frac{r}{\sigma^2}\ln\left(\frac{S}{X}\right)+\frac{\ln^2\left(\frac{S}{X}\right)}{2\sigma^2(T-t)}+\frac{\left(r^2+\frac{\sigma^4}{4}\right)}{2\sigma^2}(T-t)\frac{1}{2}}$$

$$Se^{-\frac{d_1^2}{2}} = Se^{\ln\sqrt{\frac{X}{S}}}e^{-\frac{r}{2}(T-t)}e^{-\frac{r}{\sigma^2}\ln\left(\frac{S}{X}\right)+\frac{\ln^2\left(\frac{S}{X}\right)}{2\sigma^2(T-t)}+\frac{\left(r^2+\frac{\sigma^4}{4}\right)}{2\sigma^2}(T-t)\frac{1}{2}}$$

$$Se^{-\frac{d_1^2}{2}} = \sqrt{SX}\,e^{-\frac{r}{2}(T-t)}e^{-\frac{r}{\sigma^2}\ln\left(\frac{S}{X}\right)+\frac{\ln^2\left(\frac{S}{X}\right)}{2\sigma^2(T-t)}+\frac{\left(r^2+\frac{\sigma^4}{4}\right)}{2\sigma^2}(T-t)\frac{1}{2}}$$

Por outro lado:

$$Xe^{-\frac{d_2^2}{2}}e^{-r(T-t)} = Xe^{\ln\left(\sqrt{\frac{S}{X}}\right)}e^{-r(T-t)}e^{\frac{r}{2}(T-t)}e^{-\frac{r}{\sigma^2}\ln\left(\frac{S}{X}\right)+\frac{\ln^2\left(\frac{S}{X}\right)}{2\sigma^2(T-t)}+\frac{\left(r^2+\frac{\sigma^4}{4}\right)}{2\sigma^2}(T-t)\frac{1}{2}} = \sqrt{SX}\,e^{-\frac{r}{2}(T-t)}e^{-\frac{r}{\sigma^2}\ln\left(\frac{S}{X}\right)+\frac{\ln^2\left(\frac{S}{X}\right)}{2\sigma^2(T-t)}+\frac{\left(r^2+\frac{\sigma^4}{4}\right)}{2\sigma^2}(T-t)\frac{1}{2}}$$

Logo $Se^{-\frac{d_1^2}{2}} = Xe^{-\frac{d_2^2}{2}}e^{-r(T-t)}$ CQD.

Agora $\Phi(z) = \frac{1}{\sqrt{2\pi}}\int_{-\infty}^{z} e^{-\frac{u^2}{2}}du$, logo $\Phi'(z) = \frac{1}{\sqrt{2\pi}}e^{-\frac{z^2}{2}}$

$$\frac{\partial c}{\partial t} = S\Phi'(d_1)\frac{\partial d_1}{\partial t} - Xe^{-r(T-t)}\Phi'(d_2)\frac{\partial d_2}{\partial t} - rXe^{-r(T-t)}\Phi(d_2)$$

$$\frac{\partial c}{\partial t} = \frac{1}{\sqrt{2\pi}}Se^{-\frac{d_1^2}{2}}\frac{\partial d_1}{\partial t} - \frac{1}{\sqrt{2\pi}}Xe^{-r(T-t)}e^{-\frac{d_1^2}{2}}\frac{\partial d_2}{\partial t} - rXe^{-r(T-t)}\Phi(d_2)$$

$$\frac{\partial c}{\partial t} = \frac{1}{\sqrt{2\pi}}Se^{-\frac{d_1^2}{2}}\left(\frac{\partial d_1}{\partial t} - \frac{\partial d_2}{\partial t}\right) - rXe^{-r(T-t)}\Phi(d_2)$$

$$\frac{\partial d_1}{\partial t} = \frac{\ln\left(\frac{S}{X}\right)}{2\sigma(T-t)^{\frac{3}{2}}} - \frac{\left(r+\frac{\sigma^2}{2}\right)}{2\sigma\sqrt{(T-t)}}$$

$$\frac{\partial d_2}{\partial t} = \frac{\ln\left(\frac{S}{X}\right)}{2\sigma(T-t)^{\frac{3}{2}}} - \frac{\left(r-\frac{\sigma^2}{2}\right)}{2\sigma\sqrt{(T-t)}}$$

$$\frac{\partial d_1}{\partial t}-\frac{\partial d_2}{\partial t}=\frac{\ln\left(\frac{S}{X}\right)}{2\sigma(T-t)^{\frac{3}{2}}}-\frac{\left(r+\frac{\sigma^2}{2}\right)}{2\sigma\sqrt{(T-t)}}-\frac{\ln\left(\frac{S}{X}\right)}{2\sigma(T-t)^{\frac{3}{2}}}+\frac{\left(r-\frac{\sigma^2}{2}\right)}{2\sigma\sqrt{(T-t)}}$$

$$\frac{\partial d_1}{\partial t}-\frac{\partial d_2}{\partial t}=-\frac{\sigma}{2\sqrt{(T-t)}}$$

$$\frac{\partial c}{\partial t}=-\frac{\sigma}{2\sqrt{2\pi}\sqrt{(T-t)}}Se^{-\frac{d_1^2}{2}}-rXe^{-r(T-t)}\Phi(d_2)$$

$$\frac{\partial c}{\partial S}=\Phi(d_1)+S\Phi'(d_1)\frac{\partial d_1}{\partial S}-Xe^{-r(T-t)}\Phi'(d_2)\frac{\partial d_2}{\partial S}$$

$$\frac{\partial c}{\partial S}=\Phi(d_1)+\frac{1}{\sqrt{2\pi}}Se^{-\frac{d_1^2}{2}}\frac{\partial d_1}{\partial S}-\frac{1}{\sqrt{2\pi}}Xe^{-r(T-t)}e^{-\frac{d_1^2}{2}}\frac{\partial d_2}{\partial S}$$

$$\frac{\partial c}{\partial S}=\Phi(d_1)+\frac{1}{\sqrt{2\pi}}Se^{-\frac{d_1^2}{2}}\frac{\partial d_1}{\partial S}-\frac{1}{\sqrt{2\pi}}Se^{-\frac{d_1^2}{2}}\frac{\partial d_2}{\partial S}$$

$$\frac{\partial c}{\partial S}=\Phi(d_1)+\frac{1}{\sqrt{2\pi}}Se^{-\frac{d_1^2}{2}}\left(\frac{\partial d_1}{\partial S}-\frac{\partial d_2}{\partial S}\right)$$

Agora $\frac{\partial d_1}{\partial S}=\frac{1}{\sigma S\sqrt{(T-t)}}$, $\frac{\partial d_2}{\partial S}=\frac{1}{\sigma S\sqrt{(T-t)}}$ e $\frac{\partial d_1}{\partial S}-\frac{\partial d_2}{\partial S}=0$ então:

$\frac{\partial c}{\partial S}=\Phi(d_1)$ Daí:

$$\frac{\partial^2 c}{\partial S^2}=\Phi'(d_1)\frac{\partial d_1}{\partial S}=\frac{1}{\sigma S\sqrt{(T-t)}}\frac{e^{-\frac{d_1^2}{2}}}{\sqrt{2\pi}}$$

Agora é colocar todos os termos na equação:

$$\frac{\sigma^2S^2}{2}\frac{\partial^2 c}{\partial S^2}+rS\frac{\partial c}{\partial S}+\frac{\partial c}{\partial t}-rc=$$

$$=\frac{\sigma^2S^2}{2}\frac{1}{\sigma S\sqrt{(T-t)}}\frac{e^{-\frac{d_1^2}{2}}}{\sqrt{2\pi}}+rS\Phi(d_1)-\frac{\sigma}{2\sqrt{2\pi}\sqrt{(T-t)}}Se^{-\frac{d_1^2}{2}}-rXe^{-r(T-t)}\Phi(d_2)-r\left[S\Phi(d_1)-Xe^{-r(T-t)}\Phi(d_2)\right]=$$

$$=\frac{\sigma}{2\sqrt{2\pi}\sqrt{(T-t)}}Se^{-\frac{d_1^2}{2}}-\frac{\sigma}{2\sqrt{2\pi}\sqrt{(T-t)}}Se^{-\frac{d_1^2}{2}}+r\left[S\Phi(d_1)-Xe^{-r(T-t)}\Phi(d_2)\right]-r\left[S\Phi(d_1)-Xe^{-r(T-t)}\Phi(d_2)\right]$$

Logo $\frac{\sigma^2S^2}{2}\frac{\partial^2 c}{\partial S^2}+rS\frac{\partial c}{\partial S}+\frac{\partial c}{\partial t}-rc=0$. Só falta mostrar que a solução satisfaz as

condições iniciais:

$$\underset{S\to 0}{Lim}[d_1]=\underset{S\to 0}{Lim}\left[\frac{\ln\left(\frac{S}{X}\right)}{\sigma\sqrt{(T-t)}}+\frac{\left(r+\frac{\sigma^2}{2}\right)}{\sigma}\sqrt{(T-t)}\right]\to-\infty$$ e da mesma forma $\underset{S\to 0}{Lim}[d_2]\to-\infty$.

$$\underset{S\to 0}{\mathrm{Lim}}\,\Phi(\mathrm{d_i})=\frac{1}{\sqrt{2\pi}}\int_{-\infty}^{-\infty}\mathrm{e}^{-\frac{\mathrm{u}^2}{2}}\mathrm{du}=0$$ logo:

$$\mathrm{c}(0,\mathrm{t})=0$$

O outro limite é para $\mathrm{t}\to\mathrm{T}$ nesse caso

$$\underset{t\to T}{Lim}\,d_i=\underset{t\to T}{Lim}\frac{\ln\left(\frac{S}{X}\right)}{\sigma\sqrt{(T-t)}}+\underset{t\to T}{Lim}\frac{\left(r\pm\frac{\sigma^2}{2}\right)}{\sigma}\sqrt{(T-t)}=\underset{t\to T}{Lim}\frac{\ln\left(\frac{S}{X}\right)}{\sigma\sqrt{(T-t)}}$$

Mas

$$\underset{t\to T}{Lim}\frac{\ln\left(\frac{S}{X}\right)}{\sigma\sqrt{(T-t)}}=\begin{cases}+\infty & se\ \ S>X\\ -\infty & se\ \ S<X\end{cases}$$

Se $\mathrm{d_i}\to+\infty$ então $\underset{\mathrm{d_i}\to\infty}{\mathrm{Lim}}\,\Phi(\mathrm{d_i})=\frac{1}{\sqrt{2\pi}}\int_{-\infty}^{+\infty}\mathrm{e}^{-\frac{\mathrm{u}^2}{2}}\mathrm{du}=1$ e se Se $\mathrm{d_i}\to-\infty$ então

$$\underset{\mathrm{d_i}\to-\infty}{\mathrm{Lim}}\,\Phi(\mathrm{d_i})=\frac{1}{\sqrt{2\pi}}\int_{-\infty}^{-\infty}\mathrm{e}^{-\frac{\mathrm{u}^2}{2}}\mathrm{du}=0$$ então:

$$\underset{t\to T}{Lim}c=\begin{cases}S_T-X & se\ \ S_T>X\\ 0 & se\ \ S_T<X\end{cases}=Max[S_T-X,0]$$

Assim a fórmula de Black&Scholes satisfaz à equação diferencial estocástica e às condições iniciais, logo é a solução.

D.2 Conversão da Equação de B&S na equação de Difusão

A equação de B&S $\frac{\partial C}{\partial t}+rS\frac{\partial C}{\partial S}+\frac{\sigma^2S^2}{2}\frac{\partial^2C}{\partial S^2}=rC$ envolve derivada segunda em relação à S e Primeira em relação à t . A equação de difusão, porém, envolve a Primeira derivada em relação à S, portanto o sinal da derivada em t está trocado, está multiplicada por S e S^2 além do termo extra rc. Vamos aplicar um conjunto de transformações até levar essa

equação para a equação de difusão.

Transformação 1: Para se livrar de S e S^2: fazer $u = \ln\frac{S}{X}$, mantendo o X para que $\frac{S}{X}$ seja adimensional, $\frac{\partial C}{\partial t}$ não muda mas $\frac{\partial C}{\partial S} = \frac{\partial C}{\partial u}\frac{\partial u}{\partial S} = \frac{1}{S}\frac{\partial C}{\partial u}$ e $\frac{\partial^2 C}{\partial S^2} = \frac{\partial}{\partial S}\left(\frac{1}{S}\frac{\partial C}{\partial u}\right) = -\frac{1}{S^2}\frac{\partial C}{\partial u} + \frac{1}{S^2}\frac{\partial^2 C}{\partial u^2}$.

Dessa forma a equação se transforma em $\frac{\partial C}{\partial t} + rS\frac{1}{S} + \frac{\sigma^2 S^2}{2}\frac{1}{S^2}\left[-\frac{\partial C}{\partial u} + \frac{\partial^2 C}{\partial u^2}\right] = rC$ que resulta em:

$$\frac{\partial C}{\partial t} + \left(r - \frac{\sigma^2}{2}\right)\frac{\partial C}{\partial u} + \frac{\sigma^2}{2}\frac{\partial^2 C}{\partial u^2} = rC$$

Transformação 2: Para se livrar do rC. Se $C(u,t)$ não dependesse de u teríamos $\frac{\partial C}{\partial u} = \frac{\partial^2 C}{\partial u^2} = 0$ e $\frac{\partial C}{\partial t} = rC$. Logo, $C = e^{rt}$ ou $C = e^{-r(T-t)}$. Fazendo $C(u,t) = e^{-r(T-t)}y(u,t)$ então

$$\frac{\partial C}{\partial t} = re^{-r(T-t)}y + e^{-r(T-t)}\frac{\partial y}{\partial t} = rC + e^{-r(T-t)}\frac{\partial y}{\partial t}$$

$$rC + e^{-r(T-t)}\frac{\partial y}{\partial t} + \left(r - \frac{\sigma^2}{2}\right)e^{-r(T-t)}\frac{\partial y}{\partial u} + \frac{\sigma^2}{2}e^{-r(T-t)}\frac{\partial^2 y}{\partial u^2} = rC$$

$$\frac{\partial y}{\partial t} + \left(r - \frac{\sigma^2}{2}\right)\frac{\partial y}{\partial u} + \frac{\sigma^2}{2}\frac{\partial^2 y}{\partial u^2} = 0$$

Transformação 3: Para se livrar de constantes e trocar o sinal de $\frac{\partial y}{\partial t}$

$$t' = \frac{\left(r - \frac{\sigma^2}{2}\right)^2}{\frac{\sigma^2}{2}}(T - t) \text{ e } u' = \frac{\left(r - \frac{\sigma^2}{2}\right)^2}{\frac{\sigma^2}{2}}u$$

$$\frac{\partial y}{\partial t} = \frac{\partial y}{\partial t'}\frac{\partial t'}{\partial t} = -\frac{\left(r - \frac{\sigma^2}{2}\right)^2}{\frac{\sigma^2}{2}}\frac{\partial y}{\partial t'}$$

$$\frac{\partial y}{\partial u} = \frac{\partial y}{\partial u'}\frac{\partial u'}{\partial u} = -\frac{\left(r - \frac{\sigma^2}{2}\right)^2}{\frac{\sigma^2}{2}}\frac{\partial y}{\partial u'}$$

$$\frac{\partial^2 y}{\partial u^2} = \frac{\partial y}{\partial u'}\frac{\partial u'}{\partial u} = -\frac{\left(r - \frac{\sigma^2}{2}\right)^2}{\frac{\sigma^2}{2}}\frac{\partial^2 y}{\partial u'^2}$$

Substituindo

$$-\frac{\left(\boldsymbol{r}-\frac{\boldsymbol{\sigma}^2}{2}\right)^2}{\frac{\boldsymbol{\sigma}^2}{2}}\frac{\partial \boldsymbol{y}}{\partial \boldsymbol{t}'}++\frac{\left(\boldsymbol{r}-\frac{\boldsymbol{\sigma}^2}{2}\right)^2}{\frac{\boldsymbol{\sigma}^2}{2}}\frac{\partial \boldsymbol{y}}{\partial \boldsymbol{u}'}+\frac{\boldsymbol{\sigma}^2}{2}\frac{\left(\boldsymbol{r}-\frac{\boldsymbol{\sigma}^2}{2}\right)^2}{\left(\frac{\boldsymbol{\sigma}^2}{2}\right)^2}\frac{\partial^2 \boldsymbol{y}}{\partial \boldsymbol{u}'^2}=0$$

$$-\frac{\partial \boldsymbol{y}}{\partial \boldsymbol{t}'}+\frac{\partial \boldsymbol{y}}{\partial \boldsymbol{u}'}+\frac{\partial^2 \boldsymbol{y}}{\partial \boldsymbol{u}'^2}=0$$

Transformação 4: Para se livrar de $\frac{\partial \boldsymbol{y}}{\partial \boldsymbol{u}'}$

$\boldsymbol{y}(\boldsymbol{z},\boldsymbol{t}')=\boldsymbol{y}(\boldsymbol{u}'+\boldsymbol{t}',\boldsymbol{t}')$, $\boldsymbol{z}=\boldsymbol{u}'+\boldsymbol{t}'$

$\frac{\partial \boldsymbol{y}}{\partial \boldsymbol{z}}=\frac{\partial \boldsymbol{y}}{\partial \boldsymbol{u}'}\frac{\partial \boldsymbol{u}'}{\partial \boldsymbol{z}}=\frac{\partial \boldsymbol{y}}{\partial \boldsymbol{u}'}$, $\frac{\partial^2 \boldsymbol{y}}{\partial \boldsymbol{u}'^2}=\frac{\partial^2 \boldsymbol{y}}{\partial \boldsymbol{z}^2}$

$$\frac{\partial}{\partial \boldsymbol{t}'}\boldsymbol{y}(\boldsymbol{z},\boldsymbol{t}')=\frac{\partial \boldsymbol{y}}{\partial \boldsymbol{z}}\frac{\partial \boldsymbol{z}}{\partial \boldsymbol{t}'}+\frac{\partial \boldsymbol{y}}{\partial \boldsymbol{t}'}=\frac{\partial \boldsymbol{y}}{\partial \boldsymbol{z}}+\frac{\partial \boldsymbol{y}}{\partial \boldsymbol{t}'}$$

$$-\frac{\partial \boldsymbol{y}}{\partial \boldsymbol{t}'}-\frac{\partial \boldsymbol{y}}{\partial \boldsymbol{z}}+\frac{\partial \boldsymbol{y}}{\partial \boldsymbol{z}}+\frac{\partial^2 \boldsymbol{y}}{\partial \boldsymbol{z}^2}=0$$

$$-\frac{\partial \boldsymbol{y}}{\partial \boldsymbol{t}'}+\frac{\partial^2 \boldsymbol{y}}{\partial \boldsymbol{z}^2}=0$$

Ou seja, $\boldsymbol{y}=\frac{\boldsymbol{e}^{-\frac{(\boldsymbol{u}'+\boldsymbol{z}')^2}{4\boldsymbol{t}'}}}{\sqrt{4\boldsymbol{\pi}\boldsymbol{t}'}}$ é solução de $-\frac{\partial \boldsymbol{y}}{\partial \boldsymbol{t}'}+\frac{\partial \boldsymbol{y}}{\partial \boldsymbol{u}'}+\frac{\partial^2 \boldsymbol{y}}{\partial \boldsymbol{u}'^2}=0$

Agora: $\boldsymbol{y}(\boldsymbol{z},\boldsymbol{t}')=\int_{-\infty}^{+\infty}\boldsymbol{dz}'\,\boldsymbol{y}(\boldsymbol{z}',0)\frac{\boldsymbol{e}^{-\frac{(\boldsymbol{z}-\boldsymbol{z}')^2}{4\boldsymbol{t}'}}}{\sqrt{4\boldsymbol{\pi}\boldsymbol{t}'}}$

$\boldsymbol{t}\to 0$, $\boldsymbol{t}\to \boldsymbol{T}$. $\boldsymbol{t}'=\frac{\left(\boldsymbol{r}-\frac{\boldsymbol{\sigma}^2}{2}\right)^2}{\frac{\boldsymbol{\sigma}^2}{2}}(\boldsymbol{T}-\boldsymbol{t})$. Para $\boldsymbol{t}'=0$, $\boldsymbol{z}=\boldsymbol{u}'$. O termo $\boldsymbol{e}^{-\boldsymbol{r}(\boldsymbol{T}-\boldsymbol{t})}\to 1$. A condição inicial é que: $\boldsymbol{t}=\boldsymbol{T}$, $\boldsymbol{C}=\max[\boldsymbol{S}-\boldsymbol{X},0]$, como $\boldsymbol{u}=\ln\frac{\boldsymbol{S}}{\boldsymbol{X}}\to \boldsymbol{S}=\boldsymbol{x}\boldsymbol{e}^{\boldsymbol{u}}$

$$\max[\boldsymbol{S}-\boldsymbol{X},0]=\boldsymbol{H}\left[\boldsymbol{x}\boldsymbol{e}^{\boldsymbol{u}}-\boldsymbol{x}\right]=\boldsymbol{H}\left[\boldsymbol{x}\left(\boldsymbol{e}^{\boldsymbol{u}}-1\right)\right]=\begin{cases}\boldsymbol{x}\left(\boldsymbol{e}^{\boldsymbol{u}}-1\right) & \boldsymbol{se}\ \boldsymbol{e}^{\boldsymbol{u}}-1>0\\ 0 & \boldsymbol{se}\ \boldsymbol{e}^{\boldsymbol{u}}-1<0\end{cases}$$

Em termos da variável $\boldsymbol{z}$ temos que:

$$\boldsymbol{y}(\boldsymbol{z},0)=\begin{cases}\boldsymbol{x}\left(\boldsymbol{e}^{\boldsymbol{u}}-1\right) & \boldsymbol{se}\ \boldsymbol{z}\geq 0\\ 0 & \boldsymbol{se}\ \boldsymbol{z}\leq 0\end{cases}$$

Mas $\boldsymbol{u}=\frac{\frac{\boldsymbol{\sigma}^2}{2}}{\left(\boldsymbol{r}-\frac{\boldsymbol{\sigma}^2}{2}\right)}\boldsymbol{z}$ para $\boldsymbol{t}'=0$, então:

$$y(z,0)=\begin{cases} x\left[e^{\frac{\sigma^2/2}{\left(r-\sigma^2/2\right)}}-1\right] & se\ z\geq 0 \\ 0 & se\ z<0 \end{cases}$$

$$y(z,t')=\int_0^{+\infty} x\left[e^{\frac{\sigma^2/2}{\left(r-\sigma^2/2\right)}z'}-1\right]\frac{e^{\frac{(z-z')^2}{4t'}}}{\sqrt{4\pi t'}}dz'$$

Vamos mudar a varia para $q=\frac{z'-z}{\sqrt{2t'}}$, $z'=z+\sqrt{2t'}q$

$dq=\sqrt{2t'}dq$. Se $z'=0$, $q_0=-\frac{z}{\sqrt{2t'}}$

$$q_0=-\frac{u'+t'}{\sqrt{2t'}}=-\frac{\frac{\left(r-\sigma^2/2\right)}{\sigma^2/2}u+\frac{\left(r-\sigma^2/2\right)^2}{\sigma^2/2}(T-t)}{\sqrt{2\frac{\left(r-\sigma^2/2\right)^2}{\sigma^2/2}(T-t)}}=-\frac{u+\left(r-\frac{\sigma^2}{2}\right)(T-t)}{\sigma\sqrt{T-t}}$$

$$q_0=-\frac{\ln\frac{S}{X}+\left(r-\frac{\sigma^2}{2}\right)(T-t)}{\sigma\sqrt{T-t}}=-d_2$$

$$y=\int_{-d_2}^{\infty} x\left[e^{\frac{\sigma^2/2}{\left(r-\sigma^2/2\right)}\left(\sqrt{2t'}q+z\right)}-1\right]\frac{e^{-\frac{q^2}{2}}}{\sqrt{2\pi}}dq$$

$$y=xe^{\frac{\sigma^2/2}{\left(r-\sigma^2/2\right)}z}\int_{-d_2}^{\infty} e^{\frac{\sigma^2/2}{\left(r-\sigma^2/2\right)}\sqrt{2t'}q-\frac{q^2}{2}}\frac{dq}{\sqrt{2\pi}}-x\int_{-d_2}^{\infty}\frac{e^{-\frac{q^2}{2}}}{\sqrt{2\pi}}dq$$

$$\frac{\frac{\sigma^2}{2}}{\left(r-\frac{\sigma^2}{2}\right)}z=\frac{\frac{\sigma^2}{2}}{\left(r-\frac{\sigma^2}{2}\right)}(u'+t')=\frac{\frac{\sigma^2}{2}}{\left(r-\frac{\sigma^2}{2}\right)}\left[\frac{\left(r-\frac{\sigma^2}{2}\right)}{\frac{\sigma^2}{2}}u+\frac{\left(r-\frac{\sigma^2}{2}\right)^2}{\frac{\sigma^2}{2}}(T-t)\right]=u+\left(r-\frac{\sigma^2}{2}\right)(T-t)$$

$$y=xe^u e^{\left(r-\sigma^2/2\right)(T-t)}\int_{-d_2}^{+\infty} e^{\frac{\sigma^2/2}{\left(r-\sigma^2/2\right)}\sqrt{2\frac{\left(r-\sigma^2/2\right)^2}{\sigma^2/2}(T-t)}q-\frac{q^2}{2}}\frac{dq}{\sqrt{2\pi}}-xN(d_2)$$

$e^u=e^{\ln\frac{S}{X}}=\frac{S}{X}$, então:

$$y=Se^{\left(r-\sigma^2/2\right)(T-t)}\int_{-d_2}^{+\infty} e^{\sigma\sqrt{T-t}q-\frac{q^2}{2}}\frac{dq}{\sqrt{2\pi}}-xN[d_2]$$

$$-\frac{q^2}{2}+\sigma\sqrt{T-t}q=-\frac{1}{2}\left[q^2-2\sigma\sqrt{T-t}q+\sigma^2(T-t)-\sigma^2(T-t)\right]=-\frac{1}{2}\left[q-\sigma\sqrt{T-t}\right]^2+\frac{\sigma^2}{2}(T-t)$$

$$y=Se^{r(T-t)}e^{-\frac{\sigma^2}{2}(T-t)}e^{\frac{\sigma^2}{2}(T-t)}\int_{-d_2}^{+\infty}e^{-\frac{1}{2}\left[q-\sigma\sqrt{T-t}\right]^2}\frac{dq}{\sqrt{2\pi}}-xN\left[d_2\right]$$

$$q'=q-\sigma\sqrt{T-t}\ ,\ q'_0=-d_2-\sigma\sqrt{T-t}=-d_1,\ d_1=d_2=\sigma\sqrt{T-t}$$

$$d_1=\frac{\ln\frac{S}{x}+\left(r+\frac{\sigma^2}{2}\right)(T-t)}{\sigma\sqrt{T-t}}+\sigma\sqrt{T-t}=\frac{\ln\frac{S}{x}+\left(r+\frac{\sigma^2}{2}\right)(T-t)+\sigma^2(T-t)}{\sigma\sqrt{T-t}}$$

$$d_1=\frac{\ln\frac{S}{x}+\left(r+\frac{\sigma^2}{2}\right)(T-t)}{\sigma\sqrt{T-t}}$$

$$y=Se^{r(T-t)}\int_{-d_1}^{+\infty}\frac{e^{-\frac{q'^2}{2}}}{\sqrt{2\pi}}dq'-xN\left[d_2\right]$$

$$C(s,t)=SN\left[d_1\right]-e^{-r(T-t)}XN\left[d_2\right]$$

$$d_1=\frac{\ln\frac{S}{x}+\left(r+\frac{\sigma^2}{2}\right)(T-t)}{\sigma\sqrt{T-t}}\ \text{e}\ d_2=\frac{\ln\frac{S}{x}+\left(r-\frac{\sigma^2}{2}\right)(T-t)}{\sigma\sqrt{T-t}}$$

APÊNDICE E – LISTA DE TICKS

A presente lista de ticks foi reorganizada via algortimo de PRIM (MST):

Ticks	Ação	Setor
GOOGL	Alphabet Inc Class A	Communication Services
GOOG	Alphabet Inc Class C	Communication Services
AMZN	Amazon.com Inc.	Consumer Discretionary
MSFT	Microsoft Corp.	**Information Technology**
INTC	Intel Corp.	**Information Technology**
TXN	Texas Instruments	**Information Technology**
ADI	Analog Devices, Inc.	**Information Technology**
MCHP	Microchip Technology	**Information Technology**
MXIM	Maxim Integrated Products Inc	**Information Technology**
XLNX	Xilinx	**Information Technology**
AMAT	Applied Materials Inc.	**Information Technology**
LRCX	Lam Research	**Information Technology**
KLAC	KLA-Tencor Corp.	**Information Technology**
TEL	TE Connectivity Ltd.	**Information Technology**
APH	Amphenol Corp	**Information Technology**
HON	Honeywell Int'l Inc.	**Industrials**
ITW	Illinois Tool Works	**Industrials**
PH	Parker-Hannifin	**Industrials**
ETN	Eaton Corporation	**Industrials**
ROK	Rockwell Automation Inc.	**Industrials**
MMM	3M Company	**Industrials**
CAT	Caterpillar Inc.	**Industrials**
UTX	United Technologies	**Industrials**
DOV	Dover Corp.	**Industrials**
EMR	Emerson Electric Company	**Industrials**
CMI	Cummins Inc.	**Industrials**
PCAR	PACCAR Inc.	**Industrials**
ROP	Roper Technologies	**Industrials**
AME	AMETEK Inc.	**Industrials**
DHR	Danaher Corp.	**Health Care**
TROW	T. Rowe Price Group	**Financials**
IVZ	Invesco Ltd.	**Financials**
AMG	Affiliated Managers Group Inc	**Financials**
BEN	Franklin Resources	**Financials**
AMP	Ameriprise Financial	**Financials**
PFG	Principal Financial Group	**Financials**
LNC	Lincoln National	**Financials**
PRU	Prudential Financial	**Financials**

MET	MetLife Inc.	**Financials**
UNM	Unum Group	**Financials**
TMK	Torchmark Corp.	**Financials**
BLK	BlackRock	**Financials**
USB	U.S. Bancorp	**Financials**
WFC	Wells Fargo	**Financials**
BBT	BB&T Corporation	**Financials**
PNC	PNC Financial Services	**Financials**
STI	SunTrust Banks	**Financials**
KEY	KeyCorp	**Financials**
FITB	Fifth Third Bancorp	**Financials**
RF	Regions Financial Corp.	**Financials**
CMA	Comerica Inc.	**Financials**
HBAN	Huntington Bancshares	**Financials**
ZION	Zions Bancorp	**Financials**
JPM	JPMorgan Chase & Co.	**Financials**
C	Citigroup Inc.	**Financials**
BAC	Bank of America Corp	**Financials**
MS	Morgan Stanley	**Financials**
GS	Goldman Sachs Group	**Financials**
MTB	M&T Bank Corp.	**Financials**
HIG	Hartford Financial Svc.Gp.	**Financials**
L	Loews Corp.	**Financials**
BK	The Bank of New York Mellon Corp.	**Financials**
NTRS	Northern Trust Corp.	**Financials**
STT	State Street Corp.	**Financials**
RJF	Raymond James Financial Inc.	**Financials**
SCHW	Charles Schwab Corporation	**Financials**
AFL	AFLAC Inc	**Financials**
BRK-B	Berkshire Hathaway	**Financials**
ETFC	E*Trade	**Financials**
SIVB	SVB Financial	**Financials**
CINF	Cincinnati Financial	**Financials**
TRV	The Travelers Companies Inc.	**Financials**
CB	Chubb Limited	**Financials**
PBCT	People's United Financial	**Financials**
MMC	Marsh & McLennan	**Financials**
AON	Aon plc	**Financials**
AJG	Arthur J. Gallagher & Co.	**Financials**
COF	Capital One Financial	**Financials**
DFS	Discover Financial Services	**Financials**
AXP	American Express Co	**Financials**
DE	Deere & Co.	**Industrials**
ALL	Allstate Corp	**Financials**
PNR	Pentair plc	**Industrials**

GD	General Dynamics	**Industrials**
NOC	Northrop Grumman Corp.	**Industrials**
RTN	Raytheon Co.	**Industrials**
LMT	Lockheed Martin Corp.	**Industrials**
LLL	L-3 Communications Holdings	**Industrials**
A	Agilent Technologies Inc	**Health Care**
TMO	Thermo Fisher Scientific	**Health Care**
PKI	PerkinElmer	**Health Care**
MTD	Mettler Toledo	**Health Care**
WAT	Waters Corporation	**Health Care**
FLS	Flowserve Corporation	**Industrials**
IR	Ingersoll-Rand PLC	**Industrials**
PGR	Progressive Corp.	**Financials**
SWK	Stanley Black & Decker	**Industrials**
SNA	Snap-on	**Industrials**
TXT	Textron Inc.	**Industrials**
EMN	Eastman Chemical	Materials
CE	Celanese Corp.	Materials
DWDP	DowDuPont	Materials
BA	Boeing Company	**Industrials**
FISV	Fiserv Inc	**Information Technology**
JKHY	Jack Henry & Associates Inc	**Information Technology**
ADP	Automatic Data Processing	**Information Technology**
PAYX	Paychex Inc.	**Information Technology**
FIS	Fidelity National Information Services	**Information Technology**
CBRE	CBRE Group	**Real Estate**
APD	Air Products & Chemicals Inc	Materials
JEC	Jacobs Engineering Group	**Industrials**
FLR	Fluor Corp.	**Industrials**
AIZ	Assurant	**Financials**
HST	Host Hotels & Resorts	**Real Estate**
MAR	Marriott Int'l.	Consumer Discretionary
SLG	SL Green Realty	**Real Estate**
VNO	Vornado Realty Trust	**Real Estate**
BXP	Boston Properties	**Real Estate**
REG	Regency Centers Corporation	**Real Estate**
KIM	Kimco Realty	**Real Estate**
FRT	Federal Realty Investment Trust	**Real Estate**
SPG	Simon Property Group Inc	**Real Estate**
DRE	Duke Realty Corp	**Real Estate**
PLD	Prologis	**Real Estate**
ARE	Alexandria Real Estate Equities	**Real Estate**
AVB	AvalonBay Communities, Inc.	**Real Estate**
EQR	Equity Residential	**Real Estate**
UDR	UDR Inc	**Real Estate**

ESS	Essex Property Trust, Inc.	**Real Estate**
AIV	Apartment Investment & Management	**Real Estate**
MAA	Mid-America Apartments	**Real Estate**
MAC	Macerich	**Real Estate**
O	Realty Income Corporation	**Real Estate**
PSA	Public Storage	**Real Estate**
EXR	Extra Space Storage	**Real Estate**
WELL	Welltower Inc.	**Real Estate**
HCP	HCP Inc.	**Real Estate**
VTR	Ventas Inc	**Real Estate**
AOS	A.O. Smith Corp	**Industrials**
FDX	FedEx Corporation	**Industrials**
UPS	United Parcel Service	**Industrials**
UNP	Union Pacific	**Industrials**
NSC	Norfolk Southern Corp.	**Industrials**
CSX	CSX Corp.	**Industrials**
KSU	Kansas City Southern	**Industrials**
CVX	Chevron Corp.	Energy
XOM	Exxon Mobil Corp.	Energy
COP	ConocoPhillips	Energy
MRO	Marathon Oil Corp.	Energy
DVN	Devon Energy Corp.	Energy
HES	Hess Corporation	Energy
APA	Apache Corporation	Energy
OXY	Occidental Petroleum	Energy
NBL	Noble Energy Inc	Energy
EOG	EOG Resources	Energy
PXD	Pioneer Natural Resources	Energy
CXO	Concho Resources	Energy
XEC	Cimarex Energy	Energy
APC	Anadarko Petroleum Corp	Energy
SLB	Schlumberger Ltd.	Energy
HAL	Halliburton Co.	Energy
HP	Helmerich & Payne	Energy
NOV	National Oilwell Varco Inc.	Energy
FTI	TechnipFMC	Energy
NFX	Newfield Exploration Co	Energy
BHGE	Baker Hughes, a GE Company	Energy
FMC	FMC Corporation	Materials
ROL	Rollins Inc.	**Industrials**
EXPD	Expeditors	**Industrials**
MCO	Moody's Corp	**Financials**
SPGI	S&P Global, Inc.	**Financials**
RCL	Royal Caribbean Cruises Ltd	Consumer Discretionary
CCL	Carnival Corp.	Consumer Discretionary

BWA	BorgWarner	Consumer Discretionary
CHRW	C. H. Robinson Worldwide	**Industrials**
URI	United Rentals, Inc.	**Industrials**
NDAQ	Nasdaq, Inc.	**Financials**
ALB	Albemarle Corp	Materials
RHI	Robert Half International	**Industrials**
AIG	American International Group	**Financials**
CTAS	Cintas Corporation	**Industrials**
ECL	Ecolab Inc.	Materials
SWKS	Skyworks Solutions	**Information Technology**
IP	International Paper	Materials
PKG	Packaging Corporation of America	Materials
WLTW	Willis Towers Watson	**Financials**
LEG	Leggett & Platt	Consumer Discretionary
MHK	Mohawk Industries	Consumer Discretionary
MAS	Masco Corp.	**Industrials**
LEN	Lennar Corp.	Consumer Discretionary
DHI	D. R. Horton	Consumer Discretionary
PHM	Pulte Homes Inc.	Consumer Discretionary
TSS	Total System Services	**Information Technology**
ICE	Intercontinental Exchange	**Financials**
CME	CME Group Inc.	**Financials**
NUE	Nucor Corp.	Materials
GPC	Genuine Parts	Consumer Discretionary
DIS	The Walt Disney Company	Communication Services
FOX	Twenty-First Century Fox Class B	Communication Services
FOXA	Twenty-First Century Fox Class A	Communication Services
CBS	CBS Corp.	Communication Services
ACN	Accenture plc	**Information Technology**
SNPS	Synopsys Inc.	**Information Technology**
CDNS	Cadence Design Systems	**Information Technology**
OMC	Omnicom Group	Communication Services
IPG	Interpublic Group	Communication Services
HRS	Harris Corporation	**Industrials**
RE	Everest Re Group Ltd.	**Financials**
ANSS	ANSYS	**Information Technology**
F	Ford Motor	Consumer Discretionary
CMCSA	Comcast Corp.	Communication Services
AVGO	Broadcom	**Information Technology**
BR	Broadridge Financial Solutions	**Information Technology**
AVY	Avery Dennison Corp	Materials
JBHT	J. B. Hunt Transport Services	**Industrials**
DLR	Digital Realty Trust Inc	**Real Estate**
OKE	ONEOK	Energy
IFF	Intl Flavors & Fragrances	Materials

GPN	Global Payments Inc.	**Information Technology**
SYK	Stryker Corp.	**Health Care**
MDT	Medtronic plc	**Health Care**
BDX	Becton Dickinson	**Health Care**
ZBH	Zimmer Biomet Holdings	**Health Care**
WMB	Williams Cos.	Energy
FAST	Fastenal Co	**Industrials**
GWW	Grainger (W.W.) Inc.	**Industrials**
FCX	Freeport-McMoRan Inc.	Materials
ADSK	Autodesk Inc.	**Information Technology**
ADBE	Adobe Systems Inc	**Information Technology**
INTU	Intuit Inc.	**Information Technology**
CTSH	Cognizant Technology Solutions	**Information Technology**
PWR	Quanta Services Inc.	**Industrials**
JCI	Johnson Controls International	**Industrials**
WM	Waste Management Inc.	**Industrials**
RSG	Republic Services Inc	**Industrials**
GLW	Corning Inc.	**Information Technology**
ORCL	Oracle Corp.	**Information Technology**
HD	Home Depot	Consumer Discretionary
LOW	Lowe's Cos.	Consumer Discretionary
DISCA	Discovery Inc. Class A	Communication Services
DISCK	Discovery Inc. Class C	Communication Services
WHR	Whirlpool Corp.	Consumer Discretionary
XRX	Xerox	**Information Technology**
BAX	Baxter International Inc.	**Health Care**
ARNC	Arconic Inc.	**Industrials**
ABT	Abbott Laboratories	**Health Care**
HSIC	Henry Schein	**Health Care**
XRAY	Dentsply Sirona	**Health Care**
MU	Micron Technology	**Information Technology**
MA	Mastercard Inc.	**Information Technology**
V	Visa Inc.	**Information Technology**
TFX	Teleflex Inc	**Health Care**
LNT	Alliant Energy Corp	Utilities
XEL	Xcel Energy Inc	Utilities
WEC	Wec Energy Group Inc	Utilities
CMS	CMS Energy	Utilities
DTE	DTE Energy Co.	Utilities
PNW	Pinnacle West Capital	Utilities
AEP	American Electric Power	Utilities
ED	Consolidated Edison	Utilities
ES	Eversource Energy	Utilities
AEE	Ameren Corp	Utilities
DUK	Duke Energy	Utilities

SO	Southern Co.	Utilities
NEE	NextEra Energy	Utilities
PEG	Public Serv. Enterprise Inc.	Utilities
D	Dominion Energy	Utilities
ETR	Entergy Corp.	Utilities
NI	NiSource Inc.	Utilities
CNP	CenterPoint Energy	Utilities
PPL	PPL Corp.	Utilities
SRE	Sempra Energy	Utilities
AWK	American Water Works Company Inc	Utilities
EXC	Exelon Corp.	Utilities
FE	FirstEnergy Corp	Utilities
EIX	Edison Int'l	Utilities
MGM	MGM Resorts International	Consumer Discretionary
WYNN	Wynn Resorts Ltd	Consumer Discretionary
GT	Goodyear Tire & Rubber	Consumer Discretionary
BLL	Ball Corp	Materials
MSCI	MSCI Inc	**Financials**
NVDA	Nvidia Corporation	**Information Technology**
VAR	Varian Medical Systems	**Health Care**
GE	General Electric	**Industrials**
VIAB	Viacom Inc.	Communication Services
JNJ	Johnson & Johnson	**Health Care**
IBM	International Business Machines	**Information Technology**
COG	Cabot Oil & Gas	Energy
AES	AES Corp	Utilities
LKQ	LKQ Corporation	Consumer Discretionary
PFE	Pfizer Inc.	**Health Care**
MRK	Merck & Co.	**Health Care**
HOG	Harley-Davidson	Consumer Discretionary
EFX	Equifax Inc.	**Industrials**
AMGN	Amgen Inc.	**Health Care**
CELG	Celgene Corp.	**Health Care**
ADM	Archer-Daniels-Midland Co	**Consumer Staples**
SEE	Sealed Air	Materials
CRM	Salesforce.com	**Information Technology**
CTXS	Citrix Systems	**Information Technology**
GILD	Gilead Sciences	**Health Care**
LLY	Lilly (Eli) & Co.	**Health Care**
AMT	American Tower Corp.	**Real Estate**
CCI	Crown Castle International Corp.	**Real Estate**
SBAC	SBA Communications	**Real Estate**
VMC	Vulcan Materials	Materials
MLM	Martin Marietta Materials	Materials
FLIR	FLIR Systems	**Information Technology**

IT	Gartner Inc	**Information Technology**
TDG	TransDigm Group	**Industrials**
TJX	TJX Companies Inc.	Consumer Discretionary
ROST	Ross Stores	Consumer Discretionary
JWN	Nordstrom	Consumer Discretionary
M	Macy's Inc.	Consumer Discretionary
KSS	Kohl's Corp.	Consumer Discretionary
SHW	Sherwin-Williams	Materials
RHT	Red Hat Inc.	**Information Technology**
CAH	Cardinal Health Inc.	**Health Care**
MCK	McKesson Corp.	**Health Care**
ABC	AmerisourceBergen Corp	**Health Care**
ADS	Alliance Data Systems	**Information Technology**
WDC	Western Digital	**Information Technology**
STX	Seagate Technology	**Information Technology**
BIIB	Biogen Inc.	**Health Care**
BF-B	Brown-Forman Corp.	**Consumer Staples**
SBUX	Starbucks Corp.	Consumer Discretionary
TGT	Target Corp.	Consumer Discretionary
CSCO	Cisco Systems	**Information Technology**
ALXN	Alexion Pharmaceuticals	**Health Care**
ORLY	O'Reilly Automotive	Consumer Discretionary
AZO	AutoZone Inc	Consumer Discretionary
AAP	Advance Auto Parts	Consumer Discretionary
BSX	Boston Scientific	**Health Care**
VLO	Valero Energy	Energy
HFC	HollyFrontier Corp	Energy
KMX	Carmax Inc	Consumer Discretionary
GPS	Gap Inc.	Consumer Discretionary
KO	Coca-Cola Company	**Consumer Staples**
PEP	PepsiCo Inc.	**Consumer Staples**
CL	Colgate-Palmolive	**Consumer Staples**
PG	Procter & Gamble	**Consumer Staples**
KMB	Kimberly-Clark	**Consumer Staples**
CLX	The Clorox Company	**Consumer Staples**
CHD	Church & Dwight	**Consumer Staples**
MKC	McCormick & Co.	**Consumer Staples**
HRL	Hormel Foods Corp.	**Consumer Staples**
GIS	General Mills	**Consumer Staples**
K	Kellogg Co.	**Consumer Staples**
CPB	Campbell Soup	**Consumer Staples**
SJM	JM Smucker	**Consumer Staples**
MO	Altria Group Inc	**Consumer Staples**
PM	Philip Morris International	**Consumer Staples**
MOS	The Mosaic Company	Materials

CF	CF Industries Holdings Inc	Materials
FFIV	F5 Networks	**Information Technology**
CPRT	Copart Inc	**Industrials**
HOLX	Hologic	**Health Care**
HSY	The Hershey Company	**Consumer Staples**
CHTR	Charter Communications	Communication Services
VFC	V.F. Corp.	Consumer Discretionary
PVH	PVH Corp.	Consumer Discretionary
RL	Polo Ralph Lauren Corp.	Consumer Discretionary
NKE	Nike	Consumer Discretionary
TPR	Tapestry, Inc.	Consumer Discretionary
TIF	Tiffany & Co.	Consumer Discretionary
JNPR	Juniper Networks	**Information Technology**
MDLZ	Mondelez International	**Consumer Staples**
HBI	Hanesbrands Inc	Consumer Discretionary
QCOM	QUALCOMM Inc.	**Information Technology**
EL	Estee Lauder Cos.	**Consumer Staples**
TSCO	Tractor Supply Company	Consumer Discretionary
UNH	United Health Group Inc.	**Health Care**
ANTM	Anthem Inc.	**Health Care**
CI	CIGNA Corp.	**Health Care**
HUM	Humana Inc.	**Health Care**
WCG	WellCare	**Health Care**
CNC	Centene Corporation	**Health Care**
MSI	Motorola Solutions Inc.	**Information Technology**
BKNG	Booking Holdings Inc	Consumer Discretionary
T	AT&T Inc.	Communication Services
VZ	Verizon Communications	Communication Services
LH	Laboratory Corp. of America Holding	**Health Care**
NTAP	NetApp	**Information Technology**
LUV	Southwest Airlines	**Industrials**
DAL	Delta Air Lines Inc.	**Industrials**
UAL	United Continental Holdings	**Industrials**
AAL	American Airlines Group	**Industrials**
ALK	Alaska Air Group Inc	**Industrials**
DISH	Dish Network	Communication Services
VRSK	Verisk Analytics	**Industrials**
CVS	CVS Health	**Health Care**
WBA	Walgreens Boots Alliance	**Consumer Staples**
AAPL	Apple Inc.	**Information Technology**
WU	Western Union Co	**Information Technology**
YUM	Yum! Brands Inc	Consumer Discretionary
EQIX	Equinix	**Real Estate**
VRSN	Verisign Inc.	**Information Technology**
COST	Costco Wholesale Corp.	**Consumer Staples**

LB	L Brands Inc.	Consumer Discretionary
WMT	Walmart	**Consumer Staples**
IDXX	IDEXX Laboratories	**Health Care**
CAG	Conagra Brands	**Consumer Staples**
CERN	Cerner	**Health Care**
DXC	DXC Technology	**Information Technology**
EBAY	eBay Inc.	Consumer Discretionary
TSN	Tyson Foods	**Consumer Staples**
UAA	Under Armour Class A	Consumer Discretionary
MCD	McDonald's Corp.	Consumer Discretionary
NRG	NRG Energy	Utilities
ALGN	Align Technology	**Health Care**
FL	Foot Locker Inc	Consumer Discretionary
REGN	Regeneron	**Health Care**
NWL	Newell Brands	Consumer Discretionary
HPQ	HP Inc.	**Information Technology**
IPGP	IPG Photonics Corp.	**Information Technology**
COO	The Cooper Companies	**Health Care**
TAP	Molson Coors Brewing Company	**Consumer Staples**
INCY	Incyte	**Health Care**
MYL	Mylan N.V.	**Health Care**
ISRG	Intuitive Surgical Inc.	**Health Care**
BMY	Bristol-Myers Squibb	**Health Care**
IRM	Iron Mountain Incorporated	**Real Estate**
AMD	Advanced Micro Devices Inc	**Information Technology**
FTNT	Fortinet	**Information Technology**
UHS	Universal Health Services, Inc.	**Health Care**
ILMN	Illumina Inc	**Health Care**
DLTR	Dollar Tree	Consumer Discretionary
DG	Dollar General	Consumer Discretionary
AKAM	Akamai Technologies Inc	**Information Technology**
EA	Electronic Arts	Communication Services
TTWO	Take-Two Interactive	Communication Services
ATVI	Activision Blizzard	Communication Services
AGN	Allergan, Plc	**Health Care**
DVA	DaVita Inc.	**Health Care**
RMD	ResMed	**Health Care**
DRI	Darden Restaurants	Consumer Discretionary
GRMN	Garmin Ltd.	Consumer Discretionary
PRGO	Perrigo	**Health Care**
SYY	Sysco Corp.	**Consumer Staples**
STZ	Constellation Brands	**Consumer Staples**
KR	Kroger Co.	**Consumer Staples**
EXPE	Expedia Group	Consumer Discretionary
CTL	CenturyLink Inc	Communication Services

HAS	Hasbro Inc.	Consumer Discretionary
MAT	Mattel Inc.	Consumer Discretionary
VRTX	Vertex Pharmaceuticals Inc	**Health Care**
EW	Edwards Lifesciences	**Health Care**
SYMC	Symantec Corp.	**Information Technology**
CMG	Chipotle Mexican Grill	Consumer Discretionary
NEM	Newmont Mining Corporation	Materials
NFLX	Netflix Inc.	Communication Services
ULTA	Ulta Beauty	Consumer Discretionary
MNST	Monster Beverage	**Consumer Staples**
WY	Weyerhaeuser	**Real Estate**
HRB	Block H&R	Consumer Discretionary
ABMD	ABIOMED Inc	**Health Care**
NKTR	Nektar Therapeutics	**Health Care**
BBY	Best Buy Co. Inc.	Consumer Discretionary

Tabela 4 – Lista de Ticks organizados via MST.

REFERÊNCIAS

[1] PERKOWITZ, Sidney. **Optical characterization of semiconductors: infrared, Raman, and photoluminescence spectroscopy**. Elsevier, 2012.

[2] **List of S&P 500 companies**. Wikipedia. Disponível em: <https://en.wikipedia.org/wiki/List_of_S%26P_500_companies>. Acesso em 8 jan. 2019.

[3] CORMEN, T. H.; LEISERSON, C. E.; RIVEST, R. L.; STEIN, C. **Introduction to algorithms**. MIT press, 2009.

[4] BONDY, J.A.; MURTY, U.S.R.; **Graph theory with applications**. Vol. 290. London: Macmillan, 1976.

[5] DIESTEL, Reinhard. **Graph theory**. Springer Publishing Company, Inc., 2018.

[6] DEMAINE, Erik. **Lecture 12**: Greedy Algorithms and Minimum Spanning Tree. Disponível em: <https://ocw.mit.edu/courses/electrical-engineering-and-computer-science/6-046j-design-and-analysis-of-algorithms-spring-2015/lecture-notes/MIT6_046JS15_lec12.pdf>. Acesso em: 8 fev. 2019.

[7] MULDOON, Mark. **Lecture 7**: The Matrix-Tree Theorem. Disponível em: <https://www.learneroo.com/modules/92/nodes/512>. Acesso em: 6 fev. 2019.

[8] CHAIKEN, Seth; KLEITMAN, Daniel J. Matrix tree theorems. **Journal of combinatorial theory, Series A**, v. 24, n. 3, p. 377-381, 1978.

[9] XU, J.; BAO, Z.. **Neural networks and graph theory**. *SCIENCE CHINA Information Sciences*, v. 45, n.1, p. 1-24, 2002.

[10] NIEPERT, M.; AHMED, M.; KUTZKOV, K. **Learning convolutional neural networks for graphs**. *In: International conference on machine learning*, p. 2014-2023, 2016.

[11] CAO, S.; LU, W.; XU, Q. **Deep Neural Networks for Learning Graph Representations**. In: AAAI.. p. 1145-1152, 2016.

[12] ROUT, A.K. **Forecasting financial time series using a low complexity recurrent neural network and evolutionary learning approach**. *Journal of King Saud University-Computer and Information Sciences*, v. 29, n. 4, p. 536-552, 2017.

[13] TINO, P.; SCHITTENKOPF, C.; DORFFNER, G. **Financial volatility trading using recurrent neural networks**. *IEEE Transactions on Neural Networks*, v. 12, n. 4, p. 865-874, 2001.

[14] **Complete Graph.** Wikipedia. Disponível em: <https://en.wikipedia.org/wiki/Complete_graph>. Acesso em: 8 fev. 2019.

[15] KRUSKAL, Joseph B. On the shortest spanning subtree of a graph and the traveling salesman problem. **Proceedings of the American Mathematical society**, v. 7, n. 1, p. 48-50, 1956.

[16] KARGER, David R.; KLEIN, Philip N.; TARJAN, Robert E. A randomized linear-time

algorithm to find minimum spanning trees. **Journal of the ACM (JACM)**, v. 42, n. 2, p. 321-328, 1995.

[17] PRIM, Robert Clay. Shortest connection networks and some generalizations. **Bell system technical journal**, v. 36, n. 6, p. 1389-1401, 1957.

[18] LENZ, Carlos. **Teoria da Probabilidade I**. Disponível em: <https://www.ifi.unicamp.br/~lenz/Econofisica/Teoria da Probabilidade I - analise univariada vs2.docx>. Acesso em: 8 de junho de 2018.

[19] BRACEWELL, Ronald N. **The Fourier transform and its applications**. 3ª ed. New York: McGraw-Hill, 2000.

[20] **Normal distribution**. Wikipedia. Disponível em: <https://en.wikipedia.org/wiki/Normal_distribution>. Acesso em: 8 de junho de 2018.

[21] CAMACHO, Ludwing F.M. **Brazilian House of Representatives Analysis from Network Theory Perspective.** 2017. 125 f. Dissertação (Mestrado em Física) –Universidade Estadual de Campinas, Campinas, 2017.

[22] BISHOP, C. M. **Pattern Recognition and machine learning.** 1. ed. Singapore: Springer, 2006.

[23] JOHNSON, Norman L.; KOTZ, Samuel; BALAKRISHNAN. **Lognormal Distributions**: Continuous univariate distributions. Vol. 1. 2ª ed. New York: Wiley, 1994.

[24] **Log-Normal distribution**. Disponível em: <https://en.wikipedia.org/wiki/Log-normal_distribution> Acesso em: 8 de junho de 2018.

[25] ALMEIDA, D. B.; DE THOMAZ, A. A.; CARVALHO, H. F.; CESAR, C. L. One-and two-photon photoluminescence excitation spectra of CdTe quantum dots in a cryogenic confocal microscopy platform. **Optics express**, *23*(15), 19715-19727.

[26] BEIER, B.; BERGER, A. Method for automated background subtraction from raman spectra containing known contaminants. **The Analyst**, v. 134, n. 6, p. 1198-1202, 2009.

[17] PRESS, W.; FLANNERY, B.; TEUKOLSKY, S.; VETTERLING, W. **Numerical Recipes in Fortran 77:** The Art of Scientific Computing. 2. ed. Cambridge: Cambridge University Press, 1992.

[28] HARGITTAI, S. Savitzky-Golay least-squares polynomial filters in ECG signal processing. **IEEE Computers in Cardiology**, p. 763–766, 2005.

[29] SAVITZKY A.; GOLAY, M. J. E. Smoothing and differentiation of data by simplified least squares procedures. **Analytical Chemistry**, vol. 36, n. 8, p. 1627–1639, 1964.

[30] LIU, J.; OSADCHY, M.; ASHTON, L.; FOSTER, M.; SOLOMON, J.; GIBSON, S. Deep convolutional neural networks for Raman spectrum recognition: a unified solution. **The Analyst** *(Royal Society of Chemistry)*, 2017.

[31] BAEK, S.; PARK, A.; AHN, Y.; CHOO, J. Baseline correction using asymmetrically reweighted penalized least squares smoothing, **The Analyst (Royal Society of Chemistry)**, v. 140, n. 1, p. 250-257, 2015.

[32] ZHANG, Z.; CHEN, S.; LIANG, Y. Baseline correction using adaptive iteratively reweighted penalized least squares. **The Analyst (Royal Society of Chemistry)**, v. 135, n. 5, p. 1-8, 2010.

[33] RAMOS, P.; RUISÁNCHEZ, I. Noise and background removal in Raman spectra of ancient pigments using wavelet transform. **Journal of Raman Spectroscopy**, v. 36, n. 9, p. 848-856, 2005.

[34] XIE, Y.; YANG, L.; SUN, X.; WU, D.; CHEN, Q.; ZENG, Y.; LIU, G. An auto-adaptive background subtraction method for Raman spectra. **Spectrochimica Acta Part A: Molecular and Biomolecular Spectroscopy**, v. 161, p. 58-63, 2016.

[35] SÁNCHEZ, F.; MASSART, D.L. Application of SIMPLISMA for the assessment of peak purity in liquid chromatography with diode array detection. **Analytica Chimica Acta**, v. 298, n. 3, p. 331-339, 1994.

[36] RODRIGUEZ, J.; WESTENBERGER, B.; BUHSE, L.; KAUFFMAN, J. Standardization of Raman spectra for transfer of spectral libraries across different instruments. **The Analyst (Royal Society of Chemistry)**, v. 136, n. 20, p. 4232-4240, 2011.

[37] GRYNIEWICZ-RUZICKA, C.; ARZHANTSEV, S.; PELSTER, L.; WESTENBERGER, B.; BUHSE, L.; KAUFFMAN, J. Multivariate Calibration and Instrument Standardization for the Rapid Detection of Diethylene Glycol in Glycerin by Raman Spectroscopy. **Applied Spectroscopy**, v. 65, n. 3, p. 334-341, 2011.

[38] McCREERY, R., **Raman Spectroscopy for Chemical Analysis**. 1. ed. New Jersey: John Wiley & Sons, Inc., 2001.

[39] HUTSEBAUT, D.; VANDENABEELE, P.; MOENS, L. Evaluation of an accurate calibration and spectral standardization procedure for Raman spectroscopy. **The Analyst (Royal Society of Chemistry)**, v. 130, n. 8, p. 1204–1214, 2005.

[402] GAWINKOWSKI, S.; KAMIŃSKA, A.; ROLIŃSKI, T.; WALUK, J. A new algorithm for identification of components in a mixture: application to Raman spectra of solid amino acids. **The Analyst**, v. 139, n. 22, p. 5755-5764, 2014.

[41] ANDREW, M. Machine learning applications for petroleum geoscience. **Microscopy and Analysis**, v. 154, n. 3, p. 21-25, 2018.

[42] LIU, J.; ZHANG, R.; LI, X.; CHEN, J.; LIU, J.; QIU, J.; GAO, X.; CUI, J.; HESHIG, B. Continuous background correction using effective points selected in third-order minima segments in low-cost laser-induced breakdown spectroscopy without intensified CCD. **Optics Express**, v. 26, n. 13, 2018.

[43] JOLLIFFE, I. **Principal Component Analysis**. 2. ed. New York: Springer, 2010.

[44] PARASTAR, H; JALALI-HERAVI, M; TAULER, R. Is independent component analysis appropriate for multivariate resolution in analytical chemistry? *Trends in Analytical Chemistry*, v. 31, p. 134-143, 2012.

[45] DE JUAN, A.; JAUMOT, J.; TAULER, R. Multivariate Curve Resolution (MCR). Solving the mixture analysis problem. **Analytical Methods**, v. 6, n. 14, 2014.

[46] TAULER, R. Multivariate curve resolution applied to second order data. **Chemometrics and Intelligent Laboratory Systems**, v. 30, n. 1, p. 133-146, 1995.

[47] DE JUAN, A.; TAULER, R. Chemometrics applied to unravel multicomponent processes and mixtures: Revisiting latest trends in multivariate resolution. **Analytica Chimica Acta**, v. 500, n. 1, p. 195-210, 2003.

[48] TAULER, R.; SMILDE, A; KOWALSKI, B. Selectivity, local rank, three-way data analysis and ambiguity in multivariate curve resolution. **Journal of Chemometrics**, v. 9, n. 1, p. 31-58, 1995.

[49] DE JUAN, A.; TAULER, R. Multivariate Curve Resolution (MCR) from 2000: Progress in Concepts and Applications. **Critical Reviews in Analytical Chemistry**, v. 36, n. 3, p. 163-176, 2006.

[50] RUCKEBUSCH, C.; BLANCHET, L. Multivariate curve resolution: A review of advanced and tailored applications and challenges . **Analytica Chimica Acta**, v. 765, p. 28-36, 2013.

[51] BROWN, S; TAULER, R; WALCZAK, B. **Comprehensive Chemometrics Volume 2**. 1. ed. Slovenia: Elsevier Science, 2009.

[52] SÁNCHEZ, F.; MASSART, D.L. Application of SIMPLISMA for the assessment of peak purity in liquid chromatography with diode array detection. **Analytica Chimica Acta**, v. 298, n. 3, p. 331-339, 1994.

[53] WINDIG, W. Mixture analysis of spectral data by multivariate methods. **Chemometrics and Intelligent Laboratory Systems**, v. 4, n. 3, p. 201-213, 1988.

[54] GUTIÉRREZ, E.; ZALDIVAR, J.M. The application of Karhunen–Loéve, or principal component analysis method, to study the non-linear seismic response of structures. **Earthquake Engineering & Structural Dynamics**, v. 29, n. 9, p. 1261-1286, 2000.

[55] WINDIG, W.; GUILMENT, J. Interactive self-modeling mixture analysis. **Analytical Chemistry**, v. 63, n. 14, p. 1425-1432, 1991.

[56] SÁNCHEZ, F.; RUTAN, S.C.; GARCÍA, M.D.; MASSART, D.L. Resolution of multicomponent overlapped peaks by the orthogonal projection approach, evolving factor analysis and window factor analysis. **Chemometrics and Intelligent Laboratory Systems**, v. 36, n. 2, p. 153-164, 1997.

[57] MALINOWSKI, E. Obtaining the key set of typical vectors by factor analysis and subsequent isolation of component spectra. **Analytica Chimica Acta**, v. 134, p. 129-137,

1982.

[58] MAEDER, Marcel. Evolving factor analysis for the resolution of overlapping chromatographic peaks. **Analytical Chemistry**, v. 59, n. 3, p. 527-530, 1987.

[59] TAULER, R.; BARCELÓ, D. Multivariate curve resolution applied to liquid chromatography—diode array detection. **Trends in Analytical Chemistry**, v. 12, n. 8, p. 319-327, 1993.

[60] SULTAN, F; BOWER, J. M. Quantitative Golgi study of the rat cerebellar molecular layer interneurons using principal component analysis. **The journal of Comparative Neurology**, v. 393, n. 3, p. 353-373, 1998.

[61] BENZI, R.; DEIDDA, R.; MARROCU, M. Characterization of temperature and precipitation fields over Sardinia with principal component analysis and singular spectrum analysis. **International Journal of Climatology**, v. 17, n. 11, p. 1231-1262, 1997.

[62] AGUILERA, A.; OCAÑA, F.; VALDERRAMA. M. Stochastic modelling for evolution of stock prices by means of functional principal component analysis. **Applied Stochastic Models in Business and Industry**, v. 15, n. 4, p. 227-234, 1999.

[63] **RRUFF Project**. Disponível em: <rruff.info> Acesso em: 2 de janeiro de 2019.

[64] ABADI, M.; AGARWAL, A.; BARHAM, P.; BREVDO, E.; CHEN, Z.; CITRO, C.; CORRADO, G. S.; DAVIS, A.; DEAN, J.; DEVIN, M.; GHEMAWAT, S.; GOODFELLOW, I.; HARP, A.; IRVING, G.; ISARD, M.; JIA, Y.; JOZEFOWICZ, R.; KAISER, L.; KUDLUR, M.; LEVENBERG, J.; MANÉ, D.; MONGA, R.; MOORE, S.; MURRAY, D.; OLAH, C.; SCHUSTER, M.; SHLENS, J.; STEINER, B.; SUTSKEVER, I.; TALWAR, K.; TUCKER, P.; VANHOUCKE, V.; VASUDEVAN, V.; VIÉGAS, F.; VINYALS, O.; WARDEN, P.; WATENBERG, M.; WICKE, M.; YU, Y.; ZHENG, X. **TensorFlow**: Large scale machine learning on heterogeneous systems, 2015. Sofware available from *tensorflow.org*.

[65] EILERS, Paul HC; BOELENS, Hans FM. Baseline correction with asymmetric least squares smoothing. **Leiden University Medical Centre Report**, v. 1, n. 1, p. 5, 2005.

[66] **Rampy**. Disponível em <https://github.com/charlesll/rampy>. Acesso em: 5 jan. 2019.

[67] SIMON, Phil. **Too big to ignore: the business case for big data**. John Wiley & Sons, 2013.

[68] PROVOST, Foster; KOHAVI, Ron. Guest editors' introduction: On applied research in machine learning. **Machine learning**, v. 30, n. 2, p. 127-132, 1998.

[69] CHOLLET, F. **Keras**. Disponível em: <https://github.com/fchollet/keras> Acesso em: 2 de janeiro de 2019.

[70] NEBAUER, C. Evaluation of convolutional neural networks for visual recognition. *IEEE Transactions on Neural Networks*, v. 9, n. 4, p. 685-696, 1998.

[71] JI, S.; XU, W; YANG, M.; YU, K. 3D Convolutional Neural Networks for Human Action

Recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, v. 35, n. 1, p. 221-231, 2013.

[72] ABDEL-HAMID, O.; MOHAMED, A.; JIANG, H.; DENG, L.; PENN, G.; YU, D. Convolutional Neural Networks for Speech Recognition. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, v. 22, n. 10, p. 1533-1545, 2014.

[73] LIU, J.; OSADCHY, M.; ASHTON, L.; FOSTER, M.; SOLOMON, J.; GIBSON, S. Deep convolutional neural networks for Raman spectrum recognition: a unified solution. *The Analyst (Royal Society of Chemistry)*, 2017.

[74] BJERRUM, E.; GLAHDER, M.; SKOV, T. Data Augmentation of Spectral Data for Convolutional Neural Network (CNN) Based Deep Chemometrics. *arXiv*, 2017.

[75] KIM, I.; XIE, X. Handwritten Hangul recognition using deep convolutional neural networks. *International Journal on Document Analysis and Recognition (IJDAR)*, v. 18, n. 1, p. 1-13, 2015.

[76] SUN, X.; LI, C.; REN, F. Sentiment analysis for Chinese microblog based on deep neural networks with convolutional extension features. *Neurocomputing*, 2016.

[77] HALICEK, M.; LU, G.; LITTLE, J.; WANG, X.; PATEL, M.; GRIFFITH, C.; EL-DEIRY, M.; CHEN, A.; FEI, B. Deep convolutional neural networks for classifying head and neck cancer using hyperspectral imaging. *Journal of Biomedical Optics*, v. 22, n. 6, 2017.

[78] ZHANG, S.; GRAVE, E.; SKLAR, E.; ELHADAD, N. Longitudinal analysis of discussion topics in an online breast cancer community using convolutional neural networks. *Journal of Biomedical Informatics*, v. 69, p. 1-9, 2017.

[79] LE, M.; CHEN, J.; WANG, L.; WANG, Z.; LIU, W.; CHENG, K.; YANG, X. Automated diagnosis of prostate cancer in multi-parametric MRI based on multimodal convolutional neural networks. *Physics in Medicine and Biology*, 2017.

[80] JIN, J.; FU, K.; ZHANG, C. Traffic Sign Recognition With Hinge Loss Trained Convolutional Neural Networks. *IEEE Transactions on Intelligent Transportation Systems*, v. 15, n. 5, p. 1991-2000, 2014.

[81] ZHU, J.; HUANG, C.; YANG, M.; FUNG, C.; PUI, G. Context-based prediction for road traffic state using trajectory pattern mining and recurrent convolutional neural networks. *Information Sciences*, v. 473, p. 190-201, 2019.

[82] CARRON, K.; COX, R. Qualitative Analysis and the Answer Box: A Perspective on Portable Raman Spectroscopy. *Analytical Chemistry*, v. 82, n. 9, p. 3419-3425, 2010.

[83] RODRIGUEZ, J.; WESTENBERGER, B.; BUHSE, L.; KAUFFMAN, J. Quantitative Evaluation of the Sensitivity of Library-Based Raman Spectral Correlation Methods. *Analytical Chemistry*, v. 83, n. 11, p. 4061-4067, 2011.

[84] VAPNIK, V. **The Nature of Statistical Learning Theory**. 2. ed. New York: Springer, 2000.

[85] MAQUELIN, K.; KIRSCHNER, C.; CHOO-SMITH, L.; NGO-THI, N.; VREESWIJK, T.; STAMMLER, M.; ENDTZ H.; BRUINING, H.; NAUMANN, D.; PUPPELS, G. Prospective study of the performance of vibrational spectroscopies for rapid identification of bacterial and fungal pathogens recovered from blood cultures. *Journal of Clinical Microbiology*, v. 41, p. 324–329, 2003.

[86] LeCUN, Y.; BOTOU, L.; BENGIO, Y.; HAFNER, P. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, v. 86, n. 11, p. 2278–2324, 1998.

[87] **Why are neuron axons long and spindly? Study shows they're optimizing signaling efficiency**. Disponível em: <https://medicalxpress.com/news/2018-07-neuron-axons-spindly-theyre-optimizing.html>. Acesso em: 2 de janeiro de 2019.

[88] MAAS, A. L.; HANNUN, A. Y.; NG, A. Y. Rectifier nonlinearities improve neural network acoustic models. *Proc. ICML*, v. 30, 2013.

[89] **Gradient Descent: All You Need to Know**. Disponível em: <https://hackernoon.com/gradient-descent-aynk-7cbe95a778da>. Acesso em: 7 de janeiro de 2019.

[90] BISHOP, C. M. **Pattern Recognition and machine learning.** 1. ed. Singapore: Springer, 2006.

[91] GIBSON, A.; PATTERSON, J. **Deep Learning: A Practitioner's Approach**. 1. ed. Sebastopol: O'Reilly Media, 2017.

[92] SRIVASTAVA, N.; HINTON, G.; KRIZHEVSKY, A.; SUTSKEVER, I.; SALAKHUTDINOV, R. Dropout: A simple way to prevent neural networks from overfiting. *J. Mach. Learn. Res.*, v. 15, p. 1929–1958, 2014.

[93] L., BOTTOU. **Large-Scale Machine Learning with Stochastic Gradient Descent**. Physica-Verlag HD, 2010.

[94] DABBURA, I. Gradient Descent Algorithm and Its Variants. Disponível em: <https://towardsdatascience.com/gradient-descent-algorithm-and-its-variants-10f652806a3>. Acesso em: 8 de janeiro de 2019.

[95] SAPAHIA, R. 5 min Recap for Andrew Ng Deep Learning Specialization-Course 2. Disponível em: <https://medium.com/@rishavsapahia/5-min-recap-for-andrew-ng-deep-learning-specialization-course-2-8a59fd58ca0d>. Acesso em: 8 de janeiro de 2019.

[96] SUENAGA, H. Deep Learning 2: Part 1 Lesson 1. Disponível em: <https://medium.com/@hiromi_suenaga/deep-learning-2-part-1-lesson-1-602f73869197>. Acesso em: 8 de janeiro de 2019.

[97] CAVAIONI, M. DeepLearning series: Deep Neural Networks tuning and optimization. Disponível em: <https://medium.com/machine-learning-bites/deeplearning-series-deep-neural-networks-tuning-and-optimization-39250ff7786d>. Acesso em: 8 de janeiro de 2019.

[98] ZUR, R.; JIANG, Y.; PESCE, L.; DRUKKER, K. Noise injection for training artificial neural networks: A comparison with weight decay and early stopping. *Medical Physics*, v. 36, n. 10, p. 4810-4818, 2009.

[99] ZHONG, H.; LIU, G.; XIE, S.; ZHOU, Z. Gradient descent with adaptive momentum for active contour models. *Computer Vision*, v. 8, n. 4, p. 287-298, 2014.

[100] NG, A.; KATANFOROOSH, K.; MOURRI, Y. **RMSprop**. Improving Deep Neural Networks: Hyperparameter tuning, Regularization and Optimization, Coursera, 2017.

[101] KINGMA, D; BA, J. Adam: A method for stochastic optimization. *arXiv*, 2014.

[102] IOFE, S.; SZEGEDY, C. Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift. *ArXiv*, 2015.

[103] SHIMODAIRA, H. Improving predictive inference under covariate shift by weighting the log-likelihood function. *Journal of Statistical Planning and Inference*, v. 90, n. 2, p. 227-244, 2000.

[104] TIŇO, P. Equilibria of Iterative Softmax and Critical Temperatures for Intermittent Search in Self-Organizing Neural Networks . *Neural Computation*, v. 19, n. 4, p. 1056-1081, 2007.

[105] BOJARSKI, M.; DEL TESTA, D.; DWORAKOWSKI, D. End to end learning for self-driving cars. *ArXiv*, 2016.

[106] SHARMA, H. Identifying Traffic Signs with Deep Learning. Disponível em: <https://towardsdatascience.com/identifying-traffic-signs-with-deep-learning-5151eece09cb>. Acesso em: 10 de janeiro de 2019.

[107] ABADI, M.; AGARWAL, A.; BARHAM, P.; BREVDO, E.; CHEN, Z.; CITRO, C.; CORRADO, G. S.; DAVIS, A.; DEAN, J.; DEVIN, M.; GHEMAWAT, S.; GOODFELLOW, I.; HARP, A.; IRVING, G.; ISARD, M.; JIA, Y.; JOZEFOWICZ, R.; KAISER, L.; KUDLUR, M.; LEVENBERG, J.; MANÉ, D.; MONGA, R.; MOORE, S.; MURRAY, D.; OLAH, C.; SCHUSTER, M.; SHLENS, J.; STEINER, B.; SUTSKEVER, I.; TALWAR, K.; TUCKER, P.; VANHOUCKE, V.; VASUDEVAN, V.; VIÉGAS, F.; VINYALS, O.; WARDEN, P.; WATENBERG, M.; WICKE, M.; YU, Y.; ZHENG, X. TensorFlow: Large scale machine learning on heterogeneous systems, 2015. Sofware available from *tensorflow.org*.

[108] DeepLearning series: Convolutional Neural Networks. Disponível em: <https://medium.com/machine-learning-bites/deeplearning-series-convolutional-neural-networks-a9c2f2ee1524>. Acesso em: 10 de janeiro de 2019.

[109] GIUSTI, A.; CIRESAN, D.; MASCI, J.; GAMBARDELLA, L.; SCHMIDHUBER, J.Fast image scanning with deep max-pooling convolutional neural networks. *International Conference on Image Processing (ICIP)*, p. 4034-4038, 2013.

[110] KARPATHY, A.; TODERICI, G.; SHETTY, S.; LEUNG, T.; SUKTHANKAR, R.; FEI-FEI, L. Large-Scale Video Classification with Convolutional Neural Networks. *IEEE Conference on Computer Vision and Pattern Recognition*, p. 1725-1732, 2014.

[111] NG, A.; KATANFOROOSH, K.; MOURRI, Y. **Edge Detection Example**. Convolutional Neural Networks, Coursera, 2017.

[112] Deep Learning: Convolutional Neural Networks. Disponível em: <https://labs.bawi.io/deep-learning-convolutional-neural-networks-7992985c9c7b>. Acesso em: 13 de janeiro de 2019.

[113] Backpropagation In Convolutional Neural Networks. Disponível em: <https://www.jefkine.com/general/2016/09/05/backpropagation-in-convolutional-neural-networks/>. Acesso em: 15 de janeiro de 2019.

[114] Forward And Backpropagation in Convolutional Neural Network. Disponível em: <https://medium.com/@2017csm1006/forward-and-backpropagation-in-convolutional-neural-network-4dfa96d7b37e>. Acesso em: 15 de janeiro de 2019.

[115] HENDERSON, P.; ISLAM, R.; BACHMAN, P.; PINEAU, J. Deep Reinforcement Learning that Matters. *The Thirty-Second AAAI Conference on Artificial Intelligence*, p. 3207-3214, 2018.

[116] LECUN, Yann et al. LeNet-5, convolutional neural networks. **URL: http://yann. lecun. com/exdb/lenet**, p. 20, 2015.

[117] DESHPANDE, M. **Introduction to Convolutional Neural Networks for Vision Tasks**. Disponível em: <https://pythonmachinelearning.pro/introduction-to-convolutional-neural-networks-for-vision-tasks/>. Acesso em: 5 jan. 2019.

[118] SCARSELLI, F.; GORI, M.; TSOI, A. C.; HAGENBUCHNER, M.; MONFARDINI, G. The graph neural network model. **IEEE Transactions on Neural Networks**, v.20, n.1, p. 61–80, 2009.

[119] LIU, D. A Practical Guide to ReLU. Disponível em: <https://medium.com/tinymind/a-practical-guide-to-relu-b83ca804f1f7>. Acesso em: 4 jan. 2019.

[120] ZHANG, G. Peter. Time series forecasting using a hybrid ARIMA and neural network model. **Neurocomputing**, v. 50, p. 159-175, 2003.

[121] MACKENZIE, Donald. **An Engine, Not a Camera**: How Financial Models Shape Markets. Cambridge: MIT Press, 2006.

[122] MARINS, André, Mercado de Derivativos e Análise de Risco, AMS Editora, 2004.

[123] BESSADA, O; BARBEDP, C; ARAÚJO, G. {\it Mercado de Derivativos no Brasil: Conceitos, Operações e Estratégias}. 3ª edição. Rio de Janeiro: Editora Record, 2009.

[124] ASSAF NETO, Alexandre. **Matemática Financeira e suas aplicações**. 12ª edição. São Paulo: Atlas, 2012.

[125] ASSAF NETO, Alexandre. **Mercado Financeiro**. 7ª edição. São Paulo: Atlas, 2006.

[126] FEYNMAN, Richard. **The Feynman Lectures of Physics**, Volume I. 1964.

[127] MANSUY, Roger. The origins of the Word "Martingale". **Electronic Journal for History of Probability and Statistics**, 2011.

[128] GRIMMETT, G.; STIRZAKER, D. **Probability and Random Processes**. 3ª ed. Oxford University Press, 2001.

[129] LENZ, Bruno. **O Misterioso Sorriso da Volatilidade**: Além de Black & Scholes. 157 f. Monografia (Bacharelado em Economia) – Universidade Estadual de Campinas, Campinas, 2012.

[130] BIN, Li; YI, Tang. **Quantitative analysis, derivatives modeling, and trading strategies: in the presence of counterparty credit risk for the fixed-income market**. World Scientific, 2007.

[131] STEELE, Michael. **Stochastic Calculus and Financial Applications**. 2001.

[132] ROSS, Sheldon. **Variations on Brownian Motion**. Introduction to Probability Models. 11ª ed. Amsterdam: Elsevier, 2014.

[133] FOURIER, J.B.J. **Theorie analytique de la chaleur**. Paris: 1822.

[134] STEWART, Ian. **17 Equações Que Mudaram o Mundo**. Rio de Janeiro: Editora Zahar, 2013.

[135] JACKSON, J.D. **Classical Electrodynamics**. 3ª Ed. John Wiley & Sons, 1999.

[136] GOMBER, P; ARNDT, B; LUTAT, M; UHLE, T. **High-frequency trading**. Deutsche Börse AG, Frakfurt: Goethe Universität, 2011.

[137] The Prize in Economic Sciences 1997 - Press Release. Disponível em: <https://www.nobelprize.org/nobel_prizes/economic-sciences/laureates/1997/press.html> Acesso em: 15 jun. 2018.

[138] CESAR, C. Aula 4 - Opções. Disponível em: <https://www.ifi.unicamp.br/\~lenz/Econofisica/Aula 4 Opcoes vs 1.docx> Acesso em: 8 jun. 2018.

[139] CHANCE, Don M. **Convergence of the Binomial to the Black-Scholes Model**. Disponível em: <http://www.bus.lsu.edu/academics/finance/faculty/dchance/Instructional/Instr.htm> . Acesso em: 8 jun. 2018.

[140] WHALEY, Robert E. On the valuation of American call options on stocks with known dividends. **Journal of Financial Economics**, v. 9, n. 2, p. 207-211, 1981.

[141] BLACK, Fischer; SCHOLES, Myron. The pricing of options and corporate liabilities. **Journal of political economy**, v. 81, n. 3, p. 637-654, 1973.

[142] COX, John C. et al. Option pricing: A simplified approach. **Journal of financial Economics**, v. 7, n. 3, p. 229-263, 1979.

[143] MICCICHÈ, Salvatore et al. Degree stability of a minimum spanning tree of price return and volatility. **Physica A: Statistical Mechanics and its Applications**, v. 324, n. 1-2, p. 66-73, 2003.

[144] BZDOK, D.; ALTMAN, N.; KRZYWINSKI, M. Statistics versus machine learning. **Nature Methods** v. 15, p. 233-234, 2018.

[145] Opções Petrobras PN - PETR4. ADVFN. Disponível em: < https://br.advfn.com/bolsa-de-valores/bovespa/petrobras-PETR4/opcoes>. Acesso em: 12 fev. 2019.

[146] **seaborn.clustermap**. Disponível em: <https://seaborn.pydata.org/generated/seaborn.clustermap.html>. Acesso em: 12 fev. 2019.

[147] **scipy.cluster.hierarchy.linkage**. Disponível em: <https://docs.scipy.org/doc/scipy/reference/generated/scipy.cluster.hierarchy.linkage.html> . Acesso em: 12 fev. 2019.