



**UNIVERSIDADE FEDERAL DO CEARÁ**  
**CENTRO DE CIÊNCIAS**  
**PROGRAMA DE PÓS-GRADUAÇÃO EM CIÊNCIA DA COMPUTAÇÃO**  
**MESTRADO ACADÊMICO EM CIÊNCIA DA COMPUTAÇÃO**

**IAGO CHAVES**

**REDES BAYESIANAS PARA PREVISÃO DE FALHAS EM DISCOS RÍGIDOS**

**FORTALEZA**

**2017**

IAGO CHAVES

REDES BAYESIANAS PARA PREVISÃO DE FALHAS EM DISCOS RÍGIDOS

Dissertação apresentada ao Curso de Mestrado Acadêmico em Ciência da Computação do Programa de Pós-Graduação em Ciência da Computação do Centro de Ciências da Universidade Federal do Ceará, como requisito parcial à obtenção do título de mestre em Ciência da Computação. Área de Concentração: Ciência da Computação

Orientador: Prof. Dr. Javam de Castro Machado

Co-Orientador: Prof. Dr. João Paulo Pordeus Gomes

FORTALEZA

2017

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)

---

C438r Chaves, Iago.

Redes Bayesianas para Previsão de Falhas em Discos Rígidos / Iago Chaves. – 2017.  
51 f. : il. color.

Dissertação (mestrado) – Universidade Federal do Ceará, Centro de Ciências, Programa de Pós-Graduação em Ciência da Computação, Fortaleza, 2017.

Orientação: Prof. Dr. Javam de Castro.

Coorientação: Prof. Dr. João Paulo Gomes.

1. Discos Rígidos. 2. Previsão de Falhas. 3. Redes Bayesianas. I. Título.

CDD 005

---

IAGO CHAVES

REDES BAYESIANAS PARA PREVISÃO DE FALHAS EM DISCOS RÍGIDOS

Dissertação apresentada ao Curso de Mestrado Acadêmico em Ciência da Computação do Programa de Pós-Graduação em Ciência da Computação do Centro de Ciências da Universidade Federal do Ceará, como requisito parcial à obtenção do título de mestre em Ciência da Computação. Área de Concentração: Ciência da Computação

Aprovada em:

BANCA EXAMINADORA

---

Prof. Dr. Javam de Castro Machado (Orientador)  
Universidade Federal do Ceará (UFC)

---

Prof. Dr. João Paulo Pordeus  
Gomes (Co-Orientador)  
Universidade Federal do Ceará (UFC)

---

Prof. Dr. José Antônio Fernandes de Macêdo  
Universidade Federal do Ceará (UFC)

---

Prof. Dr. Leonardo Ramos Rodrigues  
Instituto de Aeronáutica e Espaço (IAE)

À minha família, por sua capacidade de acreditar e investir em mim. Mãe, seu cuidado e dedicação foi que deram, em alguns momentos, a esperança para seguir. Pai, sua presença significou segurança e certeza de que não estou sozinho nessa caminhada.

## **AGRADECIMENTOS**

Aos meus pais, Ismael Chaves e Iracema Castro, pelo apoio e confiança despejados em mim.

Aos Prof. Dr. Javam de Castro e Prof. Dr. João Paulo Gomes por me orientar e aconselhar em minha carreira acadêmica.

À minha namorada, Lívia Macêdo, pela compreensão, paciência e apoio nos momentos mais delicados.

Aos amigos André Luís Mendonça, Leonardo Linhares, Manoel Rui de Paula, Lucas Gonçalves, Lucas Queiroz, Eduardo Neto, Isabel Fernandes e Felipe Timbó pelo companheirismo e ajuda no desenvolvimento deste trabalho.

À todos os colegas do Laboratório de Sistemas e Banco de Dados (LSBD) pela contínua motivação.

Ao LSBD por ter fornecido uma estrutura adequada para o desenvolvimento da minha dissertação e apoio financeiro necessário para eventos científicos.

“O sucesso é ir de fracasso em fracasso sem perder entusiasmo. ”

(Winston Churchill)

## RESUMO

A capacidade de prever falhas em discos rígidos tornou-se um dos grandes objetivos de fabricantes de tais equipamentos visto que falhas inesperadas podem acarretar em significativas perdas de dados. Como consequência, este tópico de pesquisa tem atraído muita atenção em anos recentes. Atualmente, a maioria dos discos rígidos está equipada com um sistema de monitoramento de falhas denominado Self-Monitoring, Analysis and Reporting Technology (SMART). Este sistema coleta diversos parâmetros relativos ao desempenho do equipamento e detecta anomalias utilizando um método baseado em violação de limiares. Apesar da grande popularidade do sistema SMART, estudos recentes mostram que somente 3% a 10% das falhas são identificadas previamente. Com isso, muitos trabalhos têm sido desenvolvidos como alternativas ao SMART. Este trabalho apresenta um novo método para previsão de falhas com base em Redes Bayesianas, e utilizando os parâmetros coletados pelo sistema SMART. Para a demonstração da efetividade do método proposto, foram realizados experimentos em um conjunto de dados contendo 49.056 discos rígidos.

**Palavras-chave:** Discos Rígidos. Redes Bayesianas. Predição de Falhas.

## ABSTRACT

The ability to predict failures in Hard Disk Drives (HDD) is a major objective of HDD manufacturers since avoiding unexpected failures may prevent data loss. As a consequence, failure prediction in HDDs became a topic that attracted much attention in recent years. Nowadays, most HDDs are equipped with a threshold-based monitoring system named Self-Monitoring, Analysis and Reporting Technology (SMART). The system collects several performance parameters and detects anomalies that may indicate incipient failures. Although the SMART system is very popular, it achieves failure detection rates of 3% to 10%. Moreover, SMART works as an incipient failure detection method and does not provide an estimate of the remaining life of the HDD. In this paper, we propose a failure prediction method using SMART attributes and a Bayesian Network. The proposed method uses a subset of the SMART attributes and a set of SMART trend related attributes to provide remaining life estimates of HDDs. To demonstrate practical usefulness, this method was applied to a dataset consisting of 49,056 hard drives from Backblaze's data centers.

**Keywords:** Hard Disks. Failure Prediction. Bayesian Networks.

## LISTA DE ILUSTRAÇÕES

Figura 1 – Representação gráfica dos níveis de saúde. . . . .	21
Figura 2 – Esquema representativo das relações de causa e efeito para falta, erro e falha. . . . .	24
Figura 3 – Exemplo de Sistema de predição de falhas. . . . .	25
Figura 4 – Ilustração representativa da métrica horizonte de predição. O eixo $x$ refere-se ao tempo de vida do dispositivo e o eixo $y$ ao <i>Remaining Useful Life</i> (RUL) estimado a ser analisado (GOMES <i>et al.</i> , 2016). . . . .	26
Figura 5 – Ilustração representativa da métrica desempenho $\alpha$ - $\lambda$ . O eixo $x$ refere-se ao tempo de vida do dispositivo e o eixo $y$ ao RUL estimado a ser analisado. Nos instantes de tempo $t_2$ e $t_5$ a condição, dada pela equação 2.2, foi satisfeita (GOMES <i>et al.</i> , 2016). . . . .	27
Figura 6 – Ilustração representativa da Acurácia Relativa (RA). Temos que o eixo $x$ refere-se ao tempo de vida do dispositivo e o eixo $y$ o RUL estimado a ser analisado (GOMES <i>et al.</i> , 2016). . . . .	28
Figura 7 – Elementos básicos de uma rede bayesiana. . . . .	29
Figura 8 – Exemplificação da hierarquia de uma rede bayesiana . . . . .	29
Figura 9 – Exemplo de rede bayesiana para modelagem da dependência probabilística do regador, grama com o tempo (clima). . . . .	30
Figura 10 – Ilustração de uma <i>Random Forest</i> . Temos as árvores de decisão que recebem um subconjunto dos dados. E por fim o algoritmo retorna a predição após a votação de cada árvore. . . . .	35
Figura 11 – Representação gráfica do fluxo do método proposto. $\mathcal{D}$ representa a base de dados <i>Self-Monitoring, Analysis and Reporting Technology</i> (SMART) e $\langle G, \Theta \rangle$ o modelo da rede bayesiana. A etapa de criação de atributos é uma evolução proposta somente na variante 2 do trabalho. . . . .	37
Figura 12 – O processo de deslizamento da janela é executado até que a janela chegue ao nó C. . . . .	39
Figura 13 – Comparativo dos modelos. Figura 13a representa o modelo base, onde só é capturado o aspecto temporal. Figura 13b é a primeira variação do método proposto, onde é capturada informações sobre atributos SMART. E por fim a segunda variação que apresenta os indicadores de tendências, além dos SMART, que está representada na Figura 13c. . . . .	41

Figura 14 – Comparativo dos modelos. Figura 14a é a primeira variação do método proposto, onde é capturada informações sobre atributos SMART. E por fim a segunda variação que apresenta os indicadores de tendências, além dos SMART, que está representada na Figura 14b. . . . .	45
Figura 15 – Métrica PH para o disco do número serial Z301480S com os parâmetros $\alpha = 1$ e $\beta = 0.58$ . . . . .	47
Figura 16 – Métrica PH para o disco do número serial W300B33Z com os parâmetros $\alpha = 1$ e $\beta = 0.58$ . . . . .	48
Figura 17 – Desempenho $\alpha$ - $\lambda$ para o disco do número serial W300KL20 com $\alpha = 1$ e $\beta = 0.5$ . . . . .	48
Figura 18 – Desempenho $\alpha$ - $\lambda$ para o disco do número serial W300AZTP com $\alpha = 1$ e $\beta = 0.5$ . . . . .	48

## LISTA DE TABELAS

Tabela 1	– Tabela de probabilidade condicional $P(C A,B)$ . . . . .	30
Tabela 2	– Exemplo de saída do método proposto. Onde $t_i$ representa um instante de tempo e o valor da célula é a probabilidade do dispositivo falhar dentro do intervalo $[t_{i-1}, t_i]$ . . . . .	36
Tabela 3	– Importância dos atributos retornada pelo método RFE e pontos de corte gerados pelo MDLP. . . . .	45
Tabela 4	– Resultado geral da métrica Horizonte de Predição (PH) com os parâmetros $\alpha = 1$ e $\beta = 0.58$ . Resultado da métrica de RA. Resultado geral da métrica Desempenho $\alpha$ - $\lambda$ com os parâmetros $\alpha = 1$ e $\beta = 0.5$ . . . . .	46

## LISTA DE ALGORITMOS

Algoritmo 1 – Eliminação Recursiva de Atributos . . . . .	33
Algoritmo 2 – Bagging . . . . .	34

## LISTA DE ABREVIATURAS E SIGLAS

EoL	<i>End of Life</i>
FAR	<i>False Alarm Rate</i>
FMMEA	<i>Failure Modes, Mechanisms, and Effects Analysis</i>
HDD	<i>Hard Disk Drive</i>
MDLP	<i>Minimum Description Length Principle</i>
PH	Horizonte de Predição
POH	<i>Power on Hours</i>
RA	Acurácia Relativa
RF	<i>Random Forest</i>
RFE	<i>Recursive Feature Elimination</i>
RNN	<i>Recursive Neural Network</i>
ROC	<i>Receiver Operating Characteristic</i>
RPN	<i>Risk Priority Number</i>
RUL	<i>Remaining Useful Life</i>
SMART	<i>Self-Monitoring, Analysis and Reporting Technology</i>
TPC	Tabela de Probabilidade Condicional

## LISTA DE SÍMBOLOS

$P$	Probabilidade
$\alpha$	Margem de erro
$\beta$	Limiar da massa de probabilidade total
$T$	Conjunto de tempos
$d$	Dispositivo
$r^d$	RUL predito de $d$
$r_*^d$	RUL real de $d$
$t$	Instante de tempo
$B$	Rede bayesiana
$G$	Grafo de comportamento
$\Theta$	Tabela de probabilidade condicional
$X$	Conjunto de variáveis aleatórias
$\theta_i$	Probabilidades de $X_i$
$\pi_i$	Pais do nó que representa $X_i$
$\delta$	Largura de discretização
$k$	Número de categorias para discretização
$S$	Conjunto de instancias
$A$	Atributo
$c$	Ponto de corte
$C$	Classes
$P_p$	Proporção
$N$	Tamanho de $S$
$p$	Número de atributos
$M$	Método de classificação
$F$	Conjunto de atributos
$E$	Conjunto de árvores de classificação

$Y$	Conjunto de rótulos
$w$	Tamanho da janela
$P_{\text{falhar}}$	Probabilidade de falhar
$SM_x$	Atributo SMART $x$
$V_x$	Conjunto de valores do SMART $x$
$\kappa$	Parâmetro de suavização
$v_{SM_x}$	Valor do SMART $x$

## SUMÁRIO

<b>1</b>	<b>INTRODUÇÃO</b>	18
<b>1.1</b>	<b>Trabalhos Relacionados</b>	19
<b>1.2</b>	<b>Objetivos</b>	22
<i>1.2.1</i>	<i>Objetivos Gerais</i>	22
<i>1.2.2</i>	<i>Objetivos Específicos</i>	22
<b>1.3</b>	<b>Contribuições</b>	22
<i>1.3.1</i>	<i>Produção Científica</i>	22
<b>1.4</b>	<b>Estrutura da Dissertação</b>	23
<b>2</b>	<b>FUNDAMENTAÇÃO TEÓRICA</b>	24
<b>2.1</b>	<b>Previsão de Falhas</b>	24
<i>2.1.1</i>	<i>Métricas</i>	25
<i>2.1.1.1</i>	<i>Horizonte de Predição</i>	25
<i>2.1.1.2</i>	<i>Desempenho <math>\alpha</math>-<math>\lambda</math></i>	26
<i>2.1.1.3</i>	<i>Acurácia Relativa</i>	27
<b>2.2</b>	<b>Redes Bayesianas</b>	28
<i>2.2.1</i>	<i>Discretização</i>	31
<b>2.3</b>	<b>Seleção de Atributos</b>	32
<i>2.3.1</i>	<i>Eliminação Recursiva de Atributos</i>	33
<i>2.3.1.1</i>	<i>Random Forest</i>	34
<b>3</b>	<b>MÉTODO PROPOSTO</b>	36
<b>3.1</b>	<b>Preprocessamento</b>	37
<i>3.1.1</i>	<i>Seleção de Atributos</i>	37
<i>3.1.2</i>	<i>Discretização de Atributos</i>	38
<i>3.1.3</i>	<i>Criação de Novos Atributos</i>	39
<b>3.2</b>	<b>Definição da Estrutura e Cálculo de Parâmetros</b>	40
<b>4</b>	<b>RESULTADOS</b>	43
<b>4.1</b>	<b>Conjunto de Dados</b>	43
<b>4.2</b>	<b>Avaliação de Desempenho</b>	44
<b>5</b>	<b>CONCLUSÕES E TRABALHOS FUTUROS</b>	49
<b>5.1</b>	<b>Trabalhos Futuros</b>	49

**REFERÊNCIAS** ..... 51

## 1 INTRODUÇÃO

Com o advento de dispositivos computacionais modernos, a quantidade de dados gerados tomou proporções impressionantes. Esse fato ocorre devido aos sistemas buscarem, cada vez mais, armazenar todos os tipos de dados possíveis, para que seja possível obter informações através de algoritmos de mineração de dados ou aprendizado de máquina. Áreas como análise de dados, *Big Data* e *Business Intelligence* refletem essa necessidade por grandes volumes de dados.

Quase 90% desses dados são persistidos em *Hard Disk Drive* (HDD) (PINHEIRO *et al.*, 2007). Sob essas circunstâncias, as preocupações com a confiabilidade dos discos aumentaram. Portanto, os fabricantes elevaram o interesse em métodos de predição de falhas em discos rígidos, com o intuito de poderem fornecer um maior nível de confiabilidade para provedores de serviços e usuários finais.

Assim sendo, é fundamental melhorar a nossa compreensão acerca de quão robusto são estes componentes e de como elevar a confiabilidade dos dispositivos. Esse entendimento pode ser particularmente útil para orientar o desenvolvimento dos sistemas de armazenamento, bem como para elaborar estratégias capazes de evitar a perda de dados.

Atualmente, a maioria dos discos rígidos são equipados com um sistema de monitoramento chamado SMART. O SMART é um sistema de detecção de anomalia, baseado em limiares, que coleta inúmeros parâmetros de desempenho e verifica se algum deles excede um limiar predefinido. Uma anomalia detectada pelo SMART pode indicar uma falha iminente. Embora o SMART seja amplamente utilizado em discos modernos, estudos recentes mostram que ele só alcançou uma taxa de detecção de falhas de 3% até 10% (WANG *et al.*, 2013).

Inúmeros trabalhos propuseram métodos para solução do problema da baixa taxa de detecção de falhas do SMART (WANG *et al.*, 2013; WANG *et al.*, 2014; XU *et al.*, 2016). Importante mencionar que a maioria desses trabalhos são baseados nos atributos SMART e utilizam diferentes métodos para a detecção de falha incipiente (*fault detection*). Em alguns casos, os métodos propostos são comparados com o sistema padrão do SMART, e mostram melhorias notáveis na taxa de detecção de falha. Uma visão alternativa do problema de detecção de falhas é apresentada em (XU *et al.*, 2016). Nesse trabalho, os autores definem o problema de predição de disfunções como tarefa de estimar o tempo de vida remanescente (RUL) de um equipamento. O RUL é definido como a quantidade de tempo de funcionamento do equipamento até o evento de falha, levando-o a parar de funcionar.

Em (XU *et al.*, 2016), os autores utilizam uma rede neural recorrente para estimar

o RUL. A utilização da rede neural recorrente é motivada pela hipótese de que existe uma correlação temporal de alguns atributos SMART. Os autores afirmam que o uso de uma rede neural recorrente poderia fornecer melhores resultados, já que esses modelos podem capturar essas relações diretamente. Vale mencionar que o método em (XU *et al.*, 2016) apresenta estimativas pontuais da RUL, portanto, falta fornecer qualquer medida de incerteza de suas previsões. De acordo com Vachtsevanos et al. (VACHTSEVANOS *et al.*, 2006) fornecer estimativas RUL com limites de incerteza é uma parte fundamental do processo de tomada de decisão usando a informação de previsão de falhas.

Neste trabalho é apresentado um método baseado em redes bayesianas para estimativa do RUL. O método proposto faz uso da quantidade de tempo de funcionamento de um disco juntamente com um subconjunto dos atributos SMART. Também são utilizados, para enriquecer nosso método, atributos relacionados à indicadores de tendência para retratar a dependência temporal. A motivação por trás desta estratégia é similar à apresentada pelo trabalho (XU *et al.*, 2016), ao incluir esses indicadores, nosso modelo pode capturar a relação temporal dos atributos SMART. O desempenho dos métodos foi avaliado usando três métricas de previsão amplamente conhecidas, e os resultados mostraram que o método proposto apresentou o melhor desempenho geral.

## 1.1 Trabalhos Relacionados

Os trabalhos (WANG *et al.*, 2014; XU *et al.*, 2016) representam o estado da arte para o problema em previsão de falhas em discos rígidos. É apresentado um breve resumo do estado da arte para motivar e explicitar as diferenças do modelo proposto.

O objetivo do artigo (WANG *et al.*, 2014) é desenvolver um método efetivo para predição de falhas em discos rígidos com uma alta taxa de acurácia e com um baixo *False Alarm Rate* (FAR). O TSP faz uso da vantagem da natureza dos dados dos discos saudáveis e defeituosos ao invés de rotular individualmente cada observação. O objetivo é lidar com a incerteza de cada rótulo, dado que os métodos padrões de aprendizado supervisionado não o fazem. Durante a etapa de treinamento, as observações de um determinado disco são tratadas como uma série temporal e, portanto, no decorrer da evolução, é possível extrair informações da degradação do dispositivo. As informações da série temporal, juntamente com o conhecimento prévio dos dispositivos, alicerçam o desenvolvimento de uma superfície de decisão. Rótulos de observações individuais podem ser inferidos com a utilização da superfície de decisão, como

resultado do método proposto.

Para obter uma predição confiável com o mínimo de falso alarme, o TSP realiza um esforço para utilizar bem as informações extraídas dos dados, e para isso o método é dividido em duas etapas. A primeira etapa visa obter uma medida, para cada observação, de divergência para o estado saudável. A segunda etapa acompanha a progressão da anomalia através do modelo probabilístico temporal, que faz uso não só da observação atual, mas também das observações vizinhas.

Prover tempo para que seja possível realizar a cópia dos dados é uma tarefa crítica que os métodos tradicionais não resolvem. Para resolver este problema, uma nova função de custo foi incorporada ao método, chamada de distância de alarme (*alarm distance*). A distância de alarme mede a duração entre o ponto de alarme e o ponto de alarme real.

*Failure Modes, Mechanisms, and Effects Analysis* (FMMEA) é um método sistemático para analisar as físicas de falha de um sistema. A criticidade dos mecanismos de falhas pode ser quantificada pelos seu *Risk Priority Number* (RPN). De acordo com os valores de RPN, mecanismos de falhas podem ser priorizados. Portanto, o artigo seleciona os atributos dos dados de HDD com base em seus RPN.

A técnica para detecção de anomalias visa distinguir comportamentos anômalos dos eventos saudáveis para cada dispositivo. Os dados de treinamento são um subconjunto dos dados saudáveis. A técnica ainda utiliza a transformação do espaço multidimensional para o unidimensional através da distância de Mahalanobis.

Para transformar os valores da distância de Mahalanobis em variáveis normalmente distribuídas, a transformação de *Box-Cox* é utilizada (WANG *et al.*, 2014). Para detectar as anomalias é utilizado o limiar ( $\mu + 3\sigma$ ), onde o  $\mu$  é a média e o  $\sigma$  é o desvio padrão dos valores da MD após a transformação Box-Cox. Este limiar é utilizado pois garante 99.7% do nível de confiança.

Quando é detectada alguma anomalia no HDD, a falha pode não acontecer imediatamente. As anomalias são sintomas de que o disco se desviou do estado saudável, entretanto não é possível afirmar que isso é um sintoma de disfunção. Outro ponto é que as anomalias medidas podem ser resultado de ruído. Motivado por esse problema, o artigo utiliza uma janela deslizante baseada no teste da razão generalizada de verossimilhança para capturar a progressão da anomalia e prever a falha do HDD.

Com isso é possível denotar a probabilidade de uma amostra ser anômala por

$P(X_i = 1) = p_i$ .  $X_i$  pode ser visto como uma sequência de Bernoulli. Por fim, é possível calcular a razão generalizada de verossimilhança para cada instância. Se o valor da razão generalizada de verossimilhança ultrapassar um limiar  $T_F$  de entrada, significa que o disco irá falhar. O trabalho tenta estimar um valor ótimo para o limiar  $T_F$  e, para isso, utiliza uma curva *Receiver Operating Characteristic* (ROC).

No trabalho do (XU *et al.*, 2016), temos que os atributos SMART são persistidos com uma marca de tempo (*timestamp*), com o intuito de monitorar as atividades internas do disco. Neste caso, é natural aplicar modelos de análise temporal para os atributos SMART. As *Recursive Neural Network* (RNN) provaram ser uma poderosa ferramenta para modelagem de dependência temporal. Por esse motivo, os autores utilizam a RNN para avaliar a saúde e para a predição de falhas em discos rígidos.

Mais especificamente, a RNN é um tipo de rede neural artificial, onde as conexões entre neurônios formam um ciclo direto, que habilita a modelagem do comportamento temporal. A rede é definida como camada de entrada  $i$ , camada de saída  $y$  e camada oculta  $h$ . Uma entrada na rede no tempo  $t$  é denotada por  $i(t)$ . É utilizado  $h(t)$  para denotar a saída da camada oculta no tempo  $t$ , que por sua vez mantém a representação do histórico dos atributos SMART. A camada oculta pode ser considerada como uma memória interna que mantém armazenada dinamicamente a sequência de estados, ou seja, a estrutura é capaz de capturar o contexto histórico do status da saúde do HDD.

O artigo comenta que, antes de um disco falhar completamente, existe uma tendência do HDD de se aproximar de um estado anormal, do ponto de vista dos atributos SMART. Isso é o processo de deterioração da saúde do disco, e para quantificá-la é criado um grau de saúde, que é composto por diferentes intervalos de tempo antes da falha. O grau de saúde proposto apresenta 6 níveis, onde o sexto nível indica que o disco funciona corretamente e os níveis mais baixos significam que o HDD vai falhar. O primeiro nível é um alerta de que o disco falhará em pelo menos de 72 horas.

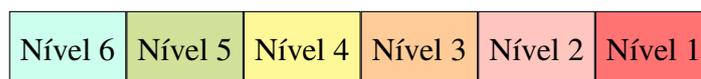


Figura 1 – Representação gráfica dos níveis de saúde.

Na fase de treinamento, o método tenta maximizar a verossimilhança da predição correta. O método tradicional de *backpropagation* não leva em consideração possíveis informações úteis, como as  $n$  amostras anteriores dos dados de treino. Portanto, o artigo faz uso de uma

outra técnica chamada *Backpropagation* através do tempo (*Backpropagation Through Time*), a qual é uma generalização do *backpropagation* (XU *et al.*, 2016).

Por fim, o método implementa um algoritmo de detecção de falha baseado em votos, para avaliar a predição de falhas. Dadas as últimas  $N$  amostras consecutivas, antes do tempo atual, o algoritmo interpreta a saída como o grau de saúde, ou seja, níveis de 1 - 4 são interpretados como falha e nível 6 como saudável. O nível 5 é considerado um grau de saúde regular.

## 1.2 Objetivos

### 1.2.1 *Objetivos Gerais*

O objetivo geral desse trabalho é propor um método para previsão de tempo de vida remanescente para discos rígidos.

### 1.2.2 *Objetivos Específicos*

Os seguintes objetivos específicos foram estabelecidos para alcançar o objetivo geral deste trabalho:

- Desenvolver um método de predição de falhas baseado em redes bayesianas;
- Selecionar e discretizar os atributos de um grande conjunto de dados de forma eficaz;
- Comparar o método proposto com outros métodos disponíveis na literatura.

## 1.3 Contribuições

O principal resultado dessa dissertação é a definição e a experimentação de um método para previsão de tempo de vida remanescente de discos rígidos.

### 1.3.1 *Produção Científica*

As contribuições científicas apresentadas neste trabalho possibilitaram as seguintes publicações:

- Chaves, I. C., de Paula, M. R. P., Leite, L. G., Queiroz, L. P., Gomes, J. P. P., & Machado, J. C. (2017, Agosto). A Bayesian Network approach to predict Hard Disk Drives Failure. Em processo de submissão.
- Chaves, I. C., de Paula, M. R. P., Leite, L. G., Queiroz, L. P., Gomes, J. P. P., & Machado, J.

C. (2016, Outubro). BaNHFaP: A Bayesian Network Based Failure Prediction Approach for Hard Disk Drives. In Intelligent Systems (BRACIS), 2016 5th Brazilian Conference on (pp. 427-432). IEEE.

Os seguintes trabalhos, desenvolvidos em parceria, atacam problemas de predição de falhas e detecção de faltas para discos rígidos:

- Queiroz, L. P., Rodrigues, F. C. M., Gomes, J. P. P., Brito, F. T., Chaves, I. C., Paula, M. R. P., ... & Machado, J. C. (2017). A fault detection method for hard disk drives based on mixture of gaussians and nonparametric statistics. *IEEE Transactions on Industrial Informatics*, 13(2), 542-550.
- Queiroz, L. P., Gomes, J. P. P., Rodrigues, F. C. M., Brito, F. T., Chaves, I. C., Leite, L. G., & Machado, J. C. (2017). Fault Detection in Hard Disk Drives Based on a Semi Parametric Model and Statistical Estimators. *New Generation Computing*, 1-15.
- Queiroz, L. P., Rodrigues, F. C. M., Gomes, J. P. P., Brito, F. T., Brito, I. C., & Machado, J. C. (2016, Outubro). Fault detection in hard disk drives based on mixture of Gaussians. In Intelligent Systems (BRACIS), 2016 5th Brazilian Conference on (pp. 145-150). IEEE.

#### **1.4 Estrutura da Dissertação**

Esta dissertação está organizada da seguinte forma: No capítulo 2 são apresentados conceitos e definições fundamentais para o entendimento deste trabalho; O capítulo 3 apresenta um método para a predição de tempo de vida remanescente para discos rígidos; O capítulo 4 apresenta os resultados para o método proposto obtidos por um conjunto de experimentos realizados considerando diferentes cenários e métricas; Finalmente, o capítulo 5 conclui o trabalho apresentando um resumo dos resultados alcançados e mostrando direções de pesquisas futuras.

## 2 FUNDAMENTAÇÃO TEÓRICA

Nesse capítulo serão descritas técnicas necessárias para explicação do método proposto por esse trabalho. A primeira seção elucida sobre o que é previsão de falha e de como é possível mensurar a qualidade das previsões. Logo após, uma seção descritiva de Redes Bayesianas e de como discretizar as variáveis para sua utilização com o método. Por fim, pormenorizar a seleção de atributos e mais especificamente a técnica recursiva de eliminação de atributos.

### 2.1 Previsão de Falhas

O propósito de um sistema de previsão de falhas é prever quando uma falha ocorrerá para que seja possível tomar decisões. Entretanto, quando delimitamos o escopo do problema encontramos três fundamentais termos: falta, erro e falha (JOHNSON, 1988). Existem relações de causa e efeito para cada um dos termos, demonstrados na figura 2. Especificamente, erros são causados pelas faltas, e falhas são causadas por erros.

Uma falta é uma deformidade ou defeito que ocorreu em algum *hardware* ou *software*. Um erro é a manifestação da falta, especificamente é um desvio de precisão ou de *corretude*. Finalmente, se o erro resultar na execução incorreta do sistema de uma das suas funções, ocorreu uma falha no sistema (JOHNSON, 1988).

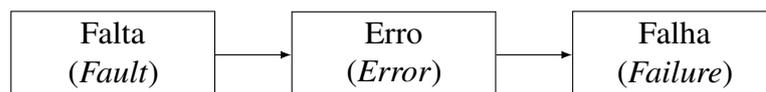


Figura 2 – Esquema representativo das relações de causa e efeito para falta, erro e falha.

Portanto, temos métodos preditivos de detecção de faltas (falhas incipientes) (*fault detection*) e de predição de falhas (*failure prediction*). A detecção de falhas incipientes consiste na identificação de comportamentos imperfeitos de um equipamento. Essa situação pode indicar que, embora o equipamento ainda esteja funcionando, ocorreu uma falta. A tarefa de prever a quantidade de tempo até que uma falha ocorra é abordada por métodos de predição de falha (disfunção). Predição de falha pode ser definida como a tarefa de estimar o RUL de um sistema ou componente (RODRIGUES *et al.*, 2015).

A figura 3 representa uma ilustração de um sistema de predição de falhas. Observamos um conjunto de atributos de um dispositivo e calculamos o seu nível de dano. O nível de dano ao longo do tempo se configura como um série temporal. Definimos dois limiares: i) limiar

de falta e ii) limiar de falha. Após o sistema alcançar o limiar de falta, é realizado o processo de predição para identificar quando o nível de dano ultrapassará o limiar de falha. Dessa forma podemos realizar a estimativa do RUL para o dispositivo. Importante ressaltar que é desejável uma confiabilidade nas predições e, por isso é utilizado distribuições de probabilidade sobre o RUL.

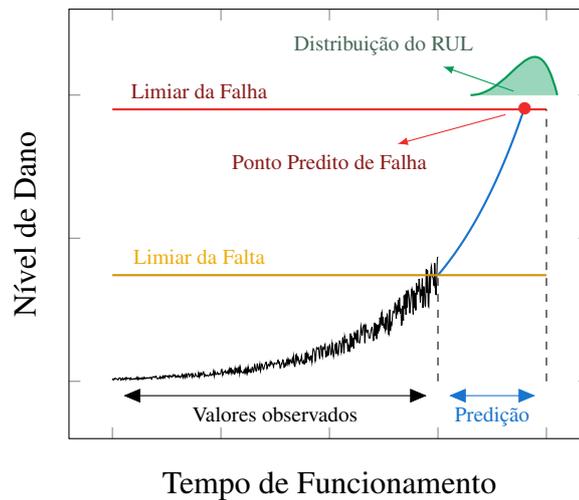


Figura 3 – Exemplo de Sistema de predição de falhas.

Os métodos de predição de falhas, como explicitado anteriormente, são importantes para as tomadas de decisões, e com isso torna-se necessário verificar a qualidade das previsões realizadas. Métricas para quantificar a confiança nos métodos foram propostas por (SAXENA *et al.*, 2009). A subseção a seguir (2.1.1) descreverá as métricas: Horizonte de predição, Desempenho  $\alpha$ - $\lambda$  e RA.

### 2.1.1 Métricas

#### 2.1.1.1 Horizonte de Predição

O PH objetiva verificar se um dado algoritmo é capaz de estimar precisamente o fim de vida com antecedência. Em outras palavras, essa métrica calcula a diferença entre o fim da vida do dispositivo e o tempo no qual o algoritmo de predição faz a primeira previsão que satisfaz uma margem de erro  $\alpha$ . O PH da seguinte forma:

$$PH = EoL - t \quad (2.1)$$

onde  $t = \min\{i \mid (i \in T) \wedge (P[r^d(i)]_{-\alpha}^{+\alpha} \geq \beta)\}$  é o primeiro instante onde a massa de

probabilidade total, dentro da margem de erro  $\alpha$ , é maior que o limiar  $\beta$ , *End of Life* (EoL) é o fim da vida do dispositivo,  $T$  é o conjunto de trimestres do dispositivo  $d$  e  $r^d(i)$  é o RUL estimado pelo método preditivo no tempo  $i$  em relação ao dispositivo  $d$ . Para o PH, valores mais elevados indicam que o método de prognóstico é capaz de fornecer previsões confiáveis mais cedo. A ilustração da métrica PH é mostrada na figura. 4

Observa-se que quanto menor o instante  $t$ , maior a pontuação obtida pelo PH. Assim, podemos afirmar que um algoritmo  $A$  produz resultados confiáveis mais adiantado que um algoritmo  $B$ , se  $A$  receber um PH maior que  $B$ .

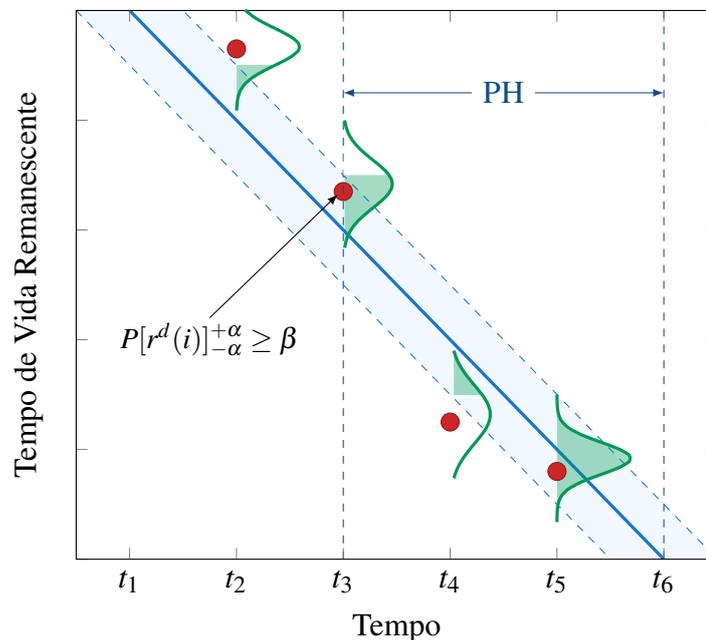


Figura 4 – Ilustração representativa da métrica horizonte de predição. O eixo  $x$  refere-se ao tempo de vida do dispositivo e o eixo  $y$  ao RUL estimado a ser analisado (GOMES *et al.*, 2016).

#### 2.1.1.2 Desempenho $\alpha$ - $\lambda$

A métrica de Desempenho  $\alpha$ - $\lambda$  verifica se o algoritmo retorna um resultado com certa margem de erro  $\alpha$  para um dado instante  $\lambda$ . No escopo desse trabalho, utilizamos o desempenho  $\alpha$ - $\lambda$  para medir a acurácia do método proposto em um dado tempo. Para facilitar a descrição, o instante  $\lambda$  que é dado como entrada para a métrica será definido em relação à porcentagem da vida útil restante do dispositivo. O desempenho  $\alpha$ - $\lambda$  é formalmente descrito

pela equação abaixo:

$$\alpha-\lambda = \begin{cases} \text{sim} & \text{se } P[r^d(\lambda)]_{-\alpha}^{+\alpha} \geq \beta \\ \text{nao} & \text{caso contrário} \end{cases} \quad (2.2)$$

onde  $\alpha$  é a margem de erro,  $\lambda$  é um instante da vida do dispositivo  $d$ ,  $\beta$  é um limiar predeterminado e  $r^d(\lambda)$  estimado pelo método preditivo no instante  $\lambda$  em relação ao dispositivo  $d$ . Nesse trabalho, consideramos a métrica  $\alpha-\lambda$  como taxa de sucesso computada (número de vezes que a métrica  $\alpha-\lambda$  retorna "sim"). A métrica  $\alpha-\lambda$  está demonstrada na Figura 5.

Em outras palavras, se a equação 2.2 retorna um "sim" como saída, o algoritmo proposto retornou um resultado dentro da margem de erro  $\alpha$  no instante  $\lambda$ . Com isso, podemos ver que o desempenho  $\alpha-\lambda$  complementa o horizonte de previsão, de modo que podemos verificar se o desempenho da previsão permanece dentro da margem de erro até o fim da vida do dispositivo.

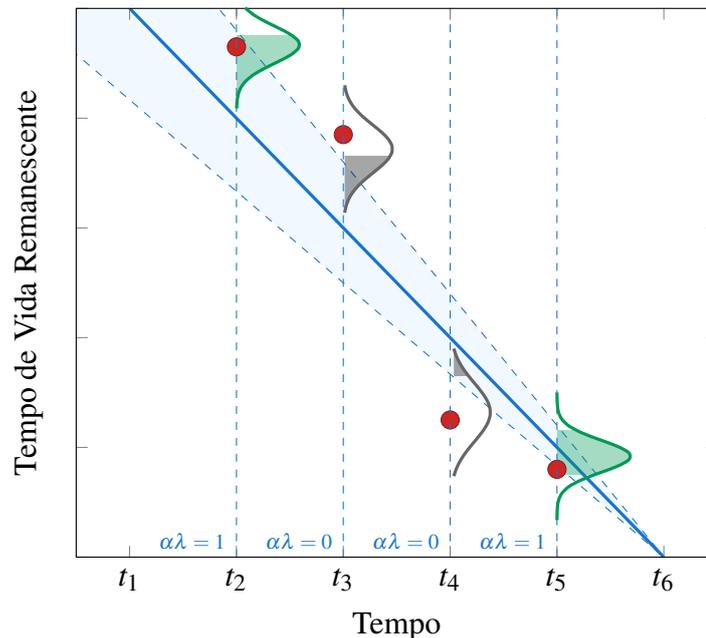


Figura 5 – Ilustração representativa da métrica desempenho  $\alpha-\lambda$ . O eixo  $x$  refere-se ao tempo de vida do dispositivo e o eixo  $y$  ao RUL estimado a ser analisado. Nos instantes de tempo  $t_2$  e  $t_5$  a condição, dada pela equação 2.2, foi satisfeita (GOMES *et al.*, 2016).

### 2.1.1.3 Acurácia Relativa

O método de RA busca verificar a qualidade da previsão em um dado instante. Dessa vez, contudo, essa análise não está limitada a verificar se o prognóstico foi gerado com uma certa

margem de erro, mas sim saber o quão próximo o prognóstico foi do verdadeiro RUL. A equação que descreve a RA é mostrada abaixo:

$$RA = 1 - \frac{|r_*(\lambda) - r^d(\lambda)|}{r_*(\lambda)} \quad (2.3)$$

onde  $\lambda$  é um instante da vida do dispositivo  $d$ ,  $r_*(\lambda)$  e  $r^d(\lambda)$  são, respectivamente o RUL verdadeiro e o RUL estimado pelo método preditivo no instante  $\lambda$  em relação ao dispositivo  $d$ . Da equação 2.3 podemos ver que, quanto mais próximo de 1, melhor é o prognóstico para o instante  $\lambda$ . Figura 6 apresenta um exemplo da métrica RA.

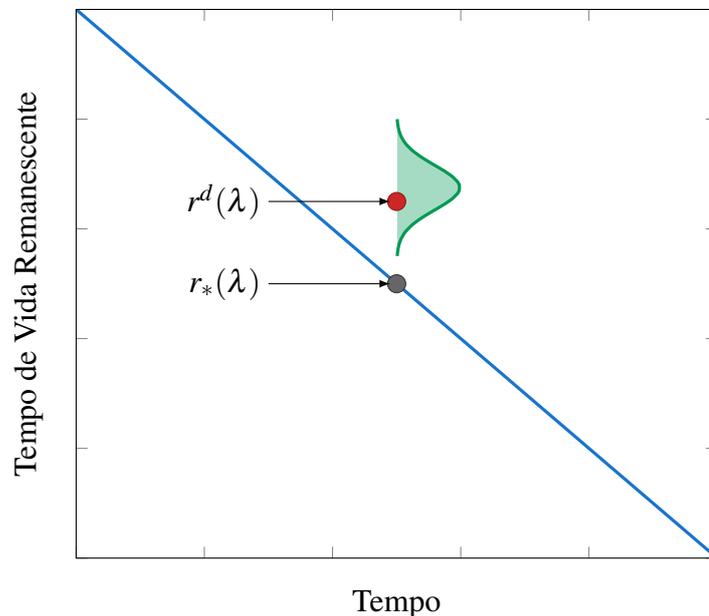


Figura 6 – Ilustração representativa da RA. Temos que o eixo  $x$  refere-se ao tempo de vida do dispositivo e o eixo  $y$  o RUL estimado a ser analisado (GOMES *et al.*, 2016).

## 2.2 Redes Bayesianas

Uma Rede Bayesiana é um modelo gráfico e probabilístico que representa um conjunto de variáveis aleatórias e suas dependências condicionais (HECKERMAN, 1995). Uma rede bayesiana pode inferir informações sobre um dado conjunto de observações de um objeto. As redes bayesianas são utilizadas em diversas áreas como em economia (DABROWSKI *et al.*, 2016; ZUO; KITA, 2012), engenharia de software (PERKUSICH *et al.*, 2015), agricultura (YET *et al.*, 2016), dentre outras grandes áreas.

Em outras palavras, as redes bayesianas são uma representação gráfica de variáveis e suas relações. Essa representação pode ser chamada de grafo. Um grafo apresenta um conjunto de

nós e um de arestas. Os nós, ou vértices, são os elementos principais de um grafo e representam as variáveis aleatórias. As arestas, ou arcos, explicitam a relação de dependência de um nó com outro. Os elementos são representados na figura 7.



Figura 7 – Elementos básicos de uma rede bayesiana.

Outros termos comuns são os de nó pai e de nó filho (descendente). Estes termos estão relacionados à hierarquia da rede, uma vez que o nó pai representa o nó de onde parte um arco direcionado ao nó filho. A figura 8 exemplifica a situação onde o nó  $A$  é pai do nó  $B$ , ou o nó  $B$  é descendente do nó  $A$ .



Figura 8 – Exemplificação da hierarquia de uma rede bayesiana

Formalmente, uma rede bayesiana é composta por uma estrutura  $B = \langle G, \Theta \rangle$ , onde  $G$  é grafo direcionado acíclico, no qual seus conjuntos de vértices são variáveis aleatórias  $X_1, \dots, X_n$  e suas arestas representam dependência condicionais entre os vértices de origem e destino. Essa dependência é representada pelo conjunto de parâmetros  $\Theta$ . Esse conjunto contém os parâmetros  $\theta_{x_i} = P(x_i | \pi_i)$  para cada  $x_i \in X_i$  condicionado por  $\pi_i$ , que por sua vez é o conjunto dos nós pais de  $x_i$ . A equação a seguir representa a distribuição conjunta definida pela rede bayesiana sobre o conjunto de variáveis aleatórias:

$$P(X_1, \dots, X_n) = \prod_{i=1}^n P(x_i | \pi_i) \quad (2.4)$$

Comumente a representação dos conjuntos de funções probabilísticas é realizada utilizando Tabela de Probabilidade Condicional (TPC), ou seja, trata-se da exibição dos parâmetros de probabilidade condicionada pelos nós pais ( $\theta_{x_i}$ ). Por exemplo, dado o conjunto de três variáveis  $A$ ,  $B$  e  $C$ , todas assumindo valores binários, onde  $A$  e  $B$  são pais da variável  $C$ , temos a tabela 1.

$C$	$A$	$B$	$P(C A,B)$
1	1	1	$\theta_1$
1	1	0	$\theta_2$
1	0	1	$\theta_3$
1	0	0	$\theta_4$
0	1	1	$\theta_5$
0	1	0	$\theta_6$
0	0	1	$\theta_7$
0	0	0	$\theta_8$

Tabela 1 – Tabela de probabilidade condicional  $P(C|A,B)$ .

A saída de uma rede bayesiana é uma resposta de pergunta, normalmente chamada de inferência. A pergunta é realizada como um conjunto de observações, que uma vez evidenciadas realizam uma atualização de crença na rede fazendo com que a distribuições de probabilidades sejam atualizadas. Entretanto devido as estruturas complexas e a quantidade de variáveis aleatórias, os cálculos para as atualizações das probabilidades exigem um grande poder computacional para serem executados em um tempo hábil. Buscando contornar esse problema, a maior parte das ferramentas utilizam métodos aproximativos estatísticos para resolução do problema.

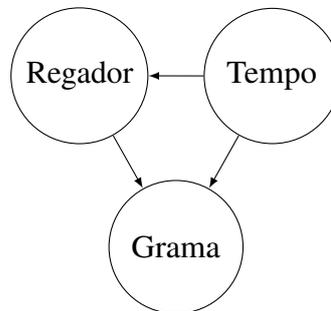


Figura 9 – Exemplo de rede bayesiana para modelagem da dependência probabilística do regador, grama com o tempo (clima).

Na figura 9 temos um exemplo de rede bayesiana. Para o exemplo dado a condição do tempo pode ter dois estados: ensolarado ou chovendo. Também observamos o estado da grama que pode encontrar-se molhada ou seca e o estado do regador que pode ser ligado ou desligado. Devemos observar que existem conexões causais no mundo, portanto se está chovendo, então isso fará com que a grama esteja molhada. Entretanto se está ensolarado a muito tempo, a grama também pode estar molhada, dessa vez indiretamente pelo regador que foi ligado.

Na sua formulação básica, uma rede bayesiana é construída com base nas probabilidade estimadas sobre variáveis discretas. Entretanto, em várias aplicações somente variáveis contínuas são disponibilizadas. Em alguns cenários, é possível utilizar algum método de dis-

cretização como a variante do *Minimum Description Length Principle* (MDLP), proposto em (FAYYAD; IRANI, 1993), o método considera a discretização baseada heurística de minimização da entropia. O algoritmo usa a entropia dos rótulos para selecionar um limiar de separação para os atributos. O algoritmo é aplicado recursivamente para cada um das partes resultantes do limiar de separação.

### 2.2.1 Discretização

Grande parte dos conjuntos de dados apresentam valores contínuos para representar alguma variável. Entretanto a utilização de valores contínuos nem sempre é possível ou mesmo desejável. Em algumas técnicas estatísticas ou de aprendizado de máquina somente são possíveis a utilização de variáveis nominais ou discretas. Então, torna-se necessário a transformação de variáveis contínuas em discretas, em um processo comumente chamado de discretização. Algumas técnicas fazem uso de pontos de corte para dividir e categorizar uma variável contínua. O termo *ponto de corte* refere-se a um valor real dentro da faixa de valores contínuos que divide o intervalo em dois segmentos, um intervalo é menor ou igual ao ponto de corte e o outro intervalo é maior do que o ponto de corte.

Um dos métodos mais utilizados é o Intervalo de Largura Fixa (*Equal Width Interval Binning*) (DOUGHERTY *et al.*, 1995). O método ordena os valores observados de um atributo contínuo e divide o intervalo de valores observados para a variável em categorias de tamanho igual a  $k$  proposto pelo usuário. Se uma variável  $z$  é observada dentro dos limites  $z_{min}$  e  $z_{max}$ , então o método computa uma largura de categoria como:

$$\delta = \frac{z_{max} - z_{min}}{k}, \quad (2.5)$$

então os pontos de cortes são definidos como  $z_{min} + i\delta$  onde  $i = 1, \dots, k - 1$ . O método é aplicado para cada varável contínua independentemente.

Alguns métodos de discretização utilizam medidas de entropia para avaliar possíveis pontos de corte. Isso significa que os métodos baseados em entropia usarão a informação da entropia da classe para selecionar limites de discretização. A entropia de classe é uma medida de pureza que mede a quantidade de informação que deve ser necessária para especificar a qual classe pertence uma instância.

O método MDLP faz uso de uma heurística para minimização da entropia (FAYYAD; IRANI, 1993). O método acha um ponto de corte único que minimiza a função de entropia e

recursivamente acha um ponto de corte para cada uma das partições geradas. O MDLP é aplicado com um critério de parada para a recursão.

Dado um conjunto  $S$  de instâncias, um atributo  $A$  e um ponto de corte  $c$ :

$$E(A, c; S) = \frac{S_1}{S} Ent(S_1) + \frac{S_2}{S} Ent(S_2), \quad (2.6)$$

onde  $Ent(S_i)$  é a classe da entropia do subconjunto  $S_i$ , definida por

$$Ent(S_i) = - \sum_{j=1}^k P(C_j, S_i) \log(P(C_j, S_i)), \quad (2.7)$$

onde existem  $k$  classes  $C_1, \dots, C_k$  e  $P_p(C_j, S_i)$  é a proporção de exemplos em  $S_i$  que possuem a classe  $C_j$ . Para um atributo  $A$ , o método MDLP seleciona um ponto de corte  $T_A$  para que  $E(A, T_A; S)$  seja mínimo ao longo de todos os pontos. Ou seja, o método escolhe um ponto de corte binário que minimiza a função entropia dentro de todos os pontos de cortes possíveis. O conjunto é dividido em duas partições pelo ponto de corte escolhido  $c_A$ . Então o método é aplicado recursivamente para ambas partições induzidas por  $c_A$  enquanto a seguinte condição é satisfeita:

$$Gain(A, c; S) \leq \frac{\log_2(N-1)}{N} + \frac{\Delta(A, c; S)}{N}, \quad (2.8)$$

onde  $N$  é o número de exemplos em  $S$ ,

$$Gain(A, c; S) = Ent(S) - E(A, c; S), \quad (2.9)$$

$$\Delta(A, c; S) = \log_2(3^k - 2) - [k \cdot Ent(S) - k_1 \cdot Ent(S_1) - k_2 \cdot Ent(S_2)], \quad (2.10)$$

e  $k_i$  é o número de classes rotuladas representadas no conjunto  $S_i$ . Uma vez que as partições ao longo de cada recursão são avaliadas de forma independente, algumas áreas nos espaços contínuos serão bem fragmentadas, enquanto outras, que têm entropia relativamente baixa, serão divididas grosseiramente.

### 2.3 Seleção de Atributos

Dado um conjunto de dados, a seleção de atributos escolhe um subconjunto de componentes que satisfatoriamente representa a totalidade dos dados. A utilização de seleção de atributos faz-se necessário em muitas técnicas de estatísticas e de aprendizado de máquina.

Geralmente a necessidade de utilização vem com o objetivo de: i) simplificar o modelo para interpretação; ii) diminuir o tempo de execução do treinamento de um método; iii) reduzir problemas de dimensionalidade e iv) reduzir o sobre-ajuste (*overfitting*).

### 2.3.1 Eliminação Recursiva de Atributos

Eliminação recursiva de atributos (*Recursive Feature Elimination*, RFE) (GUYON *et al.*, 2002) é uma técnica que procura eliminar atributos irrelevantes ou redundantes, resultando em um dado mais representativo, e menor em tamanho. O método consiste em remover os atributos recursivamente, determinando pesos para atributos utilizando um estimador externo, como máquina de suporte de vetores ou florestas aleatórias, e removendo os atributos menos relevantes. O algoritmo 1 está explicitando o funcionamento do método. A execução é finalizada quando a quantidade de atributos atinge um limiar preestabelecido como entrada.

---

#### Algoritmo 1: Eliminação Recursiva de Atributos

---

**Input:** Conjunto de treino  $S$

Conjunto de  $p$  atributos  $F = \{f_1, \dots, f_p\}$

Método de classificação  $M(S, F)$

**Output:** Classificação Final

**for**  $i$  in  $0, \dots, p$  **do**

    Classificar  $F$  utilizando  $M(S, F)$ ;

$f^* \leftarrow$  último atributo classificado de  $F$ ;

$R(p - i + 1) \leftarrow f^*$ ;

$F \leftarrow F - f^*$ ;

**end**

---

A recursão do método é necessária porque algumas medidas a importância relativa de cada atributo pode mudar substancialmente quando comparado com um subconjunto de atributos durante o processo de eliminação gradual, em particular para atributos com alta correlação.

Pequenas mudanças no conjunto de atributos podem levar a grande diferença nos pesos determinados pelo estimador externo (GUYON *et al.*, 2002). RFE efetivamente captura essa propriedade devido à avaliação do impacto de todos subconjuntos de atributos.

RFE tem aplicações que vão desde a genética (GUYON *et al.*, 2002; JOHANNES *et al.*, 2010), até problemas agro-industriais (GRANITTO *et al.*, 2006; GRANITTO *et al.*,

2008) e interfaces cérebro-computador (GYSELS *et al.*, 2005; HILL *et al.*, 2004). A maior parte da pesquisa usando RFE funciona em conjunto com Máquinas de Vetores de Suporte (Support Vector Machines, SVM) (CORTES; VAPNIK, 1995) como estimador. Mesmo assim, nesse trabalho escolhemos o método de *Random Forest* (RF) (GRANITTO *et al.*, 2006) como estimador, uma vez que alcançamos ótimos resultados e com uma execução mais rápida que com SVM.

### 2.3.1.1 *Random Forest*

O método RF é um termo geral para o conjunto de método que utilizam classificadores de tipo árvore  $\{h(z, \Xi_z), k = 1, \dots\}$  onde  $\{\Xi_k\}$  são vetores aleatórios independentes de distribuição idêntica e  $z$  é o vetor de entrada (BREIMAN, 2001). Cada árvore de classificação  $\Xi_i$  é iniciada com uma amostra dos dados de entrada e em cada divisão para cada árvore um subconjunto de atributos é escolhido. Em outras palavras, o RF utiliza abordagens de inicialização agregada (bootstrap aggregating, bagging) e seleção de variáveis aleatórias para criação das árvores.

Dado o conjunto de treino  $S = s_1, \dots, s_n$  com rótulos  $Y = y_1$ , o processo de *bagging* segue o algoritmo:

---

#### **Algoritmo 2:** Bagging

---

**Input:** Número de bagging  $B$

**for**  $b = 1, \dots, B$  **do**

Obtém amostra com substituição  $S_b$  e  $Y_b$ ;

Treina a árvore de decisão  $\Xi_b$  sobre a amostra  $X_b$  e  $Y_b$ ;

**end**

---

Então, após o treino de todas as árvores  $\Xi_i$  o resultado é dado pelo voto da maioria das respostas provenientes de cada preditor. É importante notar que o fato de utilizar a técnica de *bagging* faz com que haja uma redução na variância do erro de saída (BÜCHLMANN; YU, 2002).

A seleção de variáveis aleatórias, ou *bagging* de atributos, diminui a correlação entre as árvores não podadas e mantém o viés baixo (BREIMAN, 2001). Importante ressaltar que se um ou alguns atributos são de alta relevância para a variável de resposta (rótulo), esses recursos serão selecionados em muitas das árvores, fazendo com que elas se correlacionem. Portanto, o fato de selecionar os atributos de forma aleatória faz com que seja possível a redução da

correlação entre as árvores preditoras.

Com isso, o algoritmo de Floresta Aleatória produz um conjunto de preditores de baixo viés (*bias*) e baixa variância. Motivo pelo o qual transforma o RF em estado da arte no que refere à algoritmos de classificação.

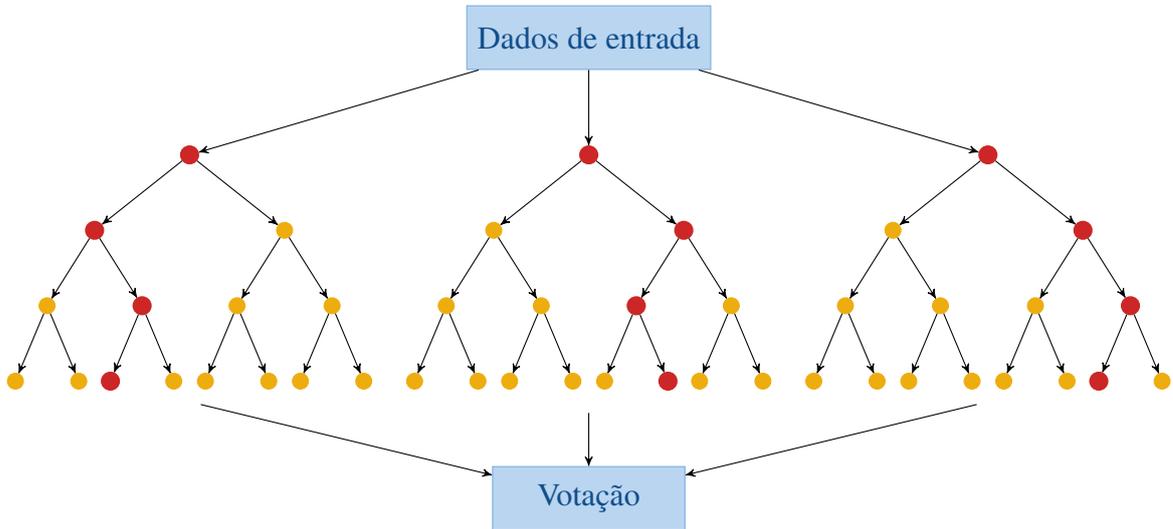


Figura 10 – Ilustração de uma *Random Forest*. Temos as árvores de decisão que recebem um subconjunto dos dados. E por fim o algoritmo retorna a previsão após a votação de cada árvore.

### 3 MÉTODO PROPOSTO

O método proposto estima o RUL de um disco rígido, utilizando um conjunto de observações. Estas observações de entrada são, mais especificamente, atributos SMART e a saída consiste em uma distribuição de probabilidade sobre o RUL. A tabela 2 exemplifica uma saída do método, onde  $t_i$  representa um instante de tempo e o valor da célula é a probabilidade do dispositivo falhar dentro do intervalo  $[t_{i-1}, t_i]$ . Podemos pensar no instante de tempo como dias, semanas, meses, trimestres ou qualquer unidade de tempo possível. De forma ilustrativa, utilizaremos a unidade de tempo como trimestres.

$t_1$	$t_2$	$t_3$	$t_4$	$t_5$	$t_6$
0.1	0.4	0.2	0.1	0.2	0.0

Tabela 2 – Exemplo de saída do método proposto. Onde  $t_i$  representa um instante de tempo e o valor da célula é a probabilidade do dispositivo falhar dentro do intervalo  $[t_{i-1}, t_i]$ .

Portanto, temos que a tabela 2 nos diz que a probabilidade do dispositivo falhar de agora até os próximos três meses é de 10%, de três a seis meses é de 40% e assim por diante.

A distribuição de probabilidade como resposta do método é de grande valia, visto que apresenta limites de incerteza. Para o processo de tomada de decisão os limites de incertezas são fundamentais (VACHTSEVANOS *et al.*, 2006). Como ferramenta para prover essa distribuição de probabilidade sobre o RUL, utilizaremos rede bayesiana, descrita na seção 2.2, como método preditor. Apresentaremos duas variantes do mesmo método: i) utilizando os atributos SMART como entrada; ii) utilizando atributos SMART e ainda apresentando uma etapa de criação de um indicador de tendência para reforçar a dependência temporal sobre cada atributo SMART. Importante ressaltar que as variantes são apresentadas e analisadas separadas para que seja possível a avaliação da etapa de criação de atributos e como o aumento da correlação temporal implica no resultado do método.

O método é composto dos seguintes módulos: i) Preprocessamento, que implementa seleção de atributos; ii) Discretização de Atributos, que transforma valores contínuos em discretos; iii) Criação de Atributos, que concebe atributos com base nos valores SMART, estando presente somente na variante 2; iv) Definição da Estrutura e Cálculo de Parâmetros, que é responsável pela computação das distribuições de probabilidade condicional para cada nó da rede proposta. A Figura 11 representa graficamente o modelo proposto. Importante ressaltar que a etapa de criação de atributos não está presente na variante 1 e por isso está representada

graficamente diferente.

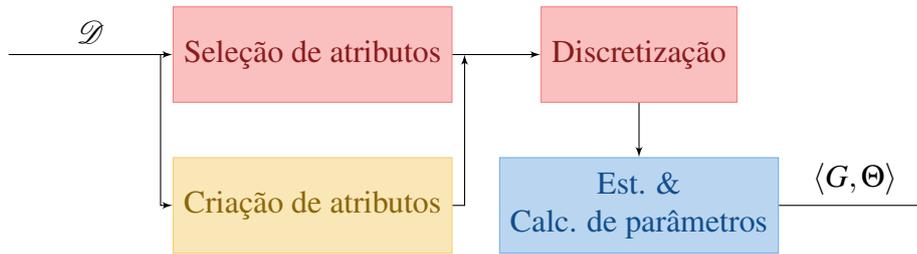


Figura 11 – Representação gráfica do fluxo do método proposto.  $\mathcal{D}$  representa a base de dados SMART e  $\langle G, \Theta \rangle$  o modelo da rede bayesiana. A etapa de criação de atributos é uma evolução proposta somente na variante 2 do trabalho.

### 3.1 Preprocessamento

A fase de preprocessamento na variante 1 é dividida em duas etapas: seleção de atributos e discretização. Diferentemente da variante 2 que é decomposto em 3 etapas: seleção de atributos, criação de atributos e discretização.

#### 3.1.1 Seleção de Atributos

O processo de seleção de atributos encontra o subconjunto de atributos SMART que melhor descreve o dado. Assim, dado um conjunto de atributos o processo de seleção encontra, dentre as possibilidades, uma combinação dos elementos do conjunto que representa de forma satisfatória os dados de entrada.

O objetivo dessa fase é remover atributos de baixa importância para representatividade dos dados. Um atributo que possui uma alta correlação com outro possui uma baixa importância para representatividade dos dados, visto que eles se comportam em sincronia. Posto isso, o processo implica na simplificação do modelo da estrutura proposta, além de reduzir a complexidade para o cálculo das tabelas de probabilidades da rede bayesiana.

A quantidade de atributos está correlacionada com a complexidade de um modelo de rede bayesiana. Quando um atributo apresenta uma correlação com outro, uma aresta deve ser criada e portanto o número de cálculos necessário para a computação da probabilidade condicional é incrementado, como visto na seção 2.2. Ou seja, o processo de seleção de atributos tem como um dos objetivos reduzir a quantidade de atributos e consequentemente reduzir a complexidade do modelo proposto.

Por consequência, atributos de maiores importâncias são selecionados pelo método

recursivo de eliminação de atributos, como descrito na seção 2.3.1. O *Recursive Feature Elimination* (RFE) é um método de boa eficácia com um moderado esforço computacional (GUYON *et al.*, 2002). Em nossa abordagem utilizamos como método preditor o algoritmo de RF.

Como visto na seção 2.3.1.1 o método de RF faz a utilização de um conjunto de árvores de decisão para a predição final. Cada uma das árvores do conjunto realiza um processo de predição e no final uma votação, com a resposta de cada árvore, é feita para descoberta do resultado da RF. Devido a característica de seleção de atributos diferentes para árvores diferentes, é possível capturar a relevância de cada atributo para realização do processo preditivo. Este valor de relevância é utilizado pelo processo recursivo do método de seleção de atributos, o RFE.

### **3.1.2 Discretização de Atributos**

Uma vez que temos os atributos selecionados para a modelagem da rede bayesiana é preciso que seja realizado a discretização. A técnica tradicional de redes bayesianas faz uso somente de atributos discretos, de acordo com a seção 2.2. Como os atributos SMART são contínuos, uma técnica de discretização deve ser empenhada para que seja possível utilizá-los com as redes bayesianas.

Para que seja possível discretizar um atributo, precisamos definir os pontos de corte. Os pontos de corte definem como os valores contínuos serão transformados para o espaço discreto. Existem dois tipos de técnicas para realização da discretização: i) assistida, onde é necessário informar a quantidade de pontos de cortes ou até mesmo os valores dos pontos de corte em si; ii) não assistida, onde a técnica faz uso de uma métrica para selecionar os pontos de cortes e discretizar os dados.

Utilizamos a técnica assistida para realizar a discretização dos atributos de tempo *Power on Hours* (POH) e RUL. A técnica é o Intervalo de Largura Fixa, vide seção 2.2.1. A necessidade da utilização de intervalos de largura fixa, para os atributos de tempo, é que não haja perda de intuição na hora de realizar inferências nas redes bayesianas.

O restante dos atributos SMART foram discretizados utilizando uma técnica não assistida, visto que não temos informações sobre ótimos pontos de cortes. Então utilizamos a técnica MDLP, descrita na seção 2.2.1, para que seja escolhida a quantidade ideal e a localização dos pontos de cortes. A técnica faz o uso da entropia de classe para escolher os pontos de corte.

### 3.1.3 Criação de Novos Atributos

A etapa de criação de atributos, presente somente na evolução do método proposto, realiza um processo de obtenção de um indicativo de tendência que quantifica uma variação nos valores dos atributos. O objetivo do processo é criar atributos que expressem a dependência temporal visto que o tempo é um fator importante para desgaste do disco. Para a criação desse indicador de tendência utilizamos uma janela deslizante.

Para a utilização da janela deslizante sobre um atributo  $a$ , colocamos as amostras em ordem temporal. Definimos o tamanho da janela como  $w$ . Obtemos todos os  $w$  primeiros elementos da série temporal do atributo  $a$  no tempo 0. No tempo 1 deslocamos a janela um elemento para direita. O processo é realizado até atingir o último elemento da série temporal. A figura 12 é um esquema gráfico representativo do algoritmo de janelas deslizantes.

Para cada janela no processo de deslizamento até o fim da série temporal calculamos o indicador de tendência para o atributo alvo. A cada iteração, calculamos o valor que representa o indicador de tendência da janela dos  $w$  elementos da janela. Os valores dos indicadores de tendências são persistidos para formar um novo atributo. Note que o novo atributo terá  $w - 1$  elementos a menos que o dado original. Assim, preenchamos os primeiros valores do novo atributo como 0, para que a quantidade de elementos do novo atributo seja igual ao do atributo alvo.

Definimos como indicador de tendência o coeficiente angular da reta que ajusta os dados de uma janela. Para ajustarmos uma reta sobre os dados de uma janela utilizamos o algoritmo de mínimos quadrados. O algoritmo linear calcula os coeficientes angular e linear da reta que melhor representa os dados.

O coeficiente angular retrata a tendência dos dados, de forma semelhante à uma derivada. Com isso, podemos quantificar a variação dos dados e conseqüentemente construir um indicador de tendência.



(a) Janela deslizante no tempo 0



(b) Janela deslizante no tempo 1

Figura 12 – O processo de deslizamento da janela é executado até que a janela chegue ao nó C.

### 3.2 Definição da Estrutura e Cálculo de Parâmetros

O processo de estimação encontra os parâmetros de uma rede bayesiana representado por  $\Theta$ . A estrutura da rede bayesiana  $G$  foi definida para refletir a relação de dependência entre os atributos SMART e o RUL.

O modelo proposto, mostrado na figura 13, usa informações sobre os atributos SMART ao decorrer do tempo e do POH de cada disco rígido para estimar o RUL. O POH é uma variável aleatória que quantifica o tempo que cada disco operou e o RUL é o indicativo de tempo remanescente de vida, portanto quanto mais velho um HDD, menos tempo de vida ele tem. Concluímos que a variável aleatória RUL é probabilisticamente dependente do POH e, portanto, modelamos o nó do RUL como descendente do nó POH na estrutura da rede bayesiana.

É conhecido que os atributos SMART representam o estado do disco rígido (PINHEIRO *et al.*, 2007). Com intuito de modelar a série temporal do estado do disco e, consequentemente, obter o modelo de deterioração progressiva descrevemos os nós desses atributos SMART como descendentes do nó do RUL no modelo  $G$ . Valores anômalos para um determinado atributo SMART podem indicar uma possível falha. Para que esses valores venham a sair do comportamento normal é possível que: o desgaste natural assolou o disco; ou que, haja um problema na construção do dispositivo. Para a solução da primeira hipótese, é utilizado a construção dos nós SMART como filhos do RUL na estrutura, que dessa forma capturará a deterioração gradativa do dispositivo. Quanto a segunda hipótese, vale ressaltar que dispositivos com defeitos de fabricação são uma pequena porcentagem do total (BACKBLAZE, 2015). Entretanto, no modelo proposto, a tabela de probabilidades fará o papel de capturar essas ocorrências e levá-las em consideração. O objetivo é representar bem o comportamento do disco, portanto confeccionar um modelo que representa a dependência do estado com o tempo é de suma importância para realização de inferências úteis no processo de tomada de decisão.

É importante ressaltar que o modelo proposto difere do modelo simples de predição de falhas (modelo base). O modelo simples faz uso somente do atributo de tempo para realizar a predição de RUL (PECHT, 2008). O modelo proposto faz uso de atributos que refletem o estado do disco rígido, não se baseando exclusivamente do tempo de funcionamento. A utilização desses atributos fazem com que o modelo proposto seja mais complexo que o modelo base.

Uma vez que modelamos o  $G$  da rede bayesiana, o parâmetro  $\Theta$  deve ser calculado. Para isso, devemos obter uma tabela de frequências para cada evidência do conjunto de dados. Essa tabela de frequências é conhecida como TPC. A TPC é a soma de todas as as ocorrências

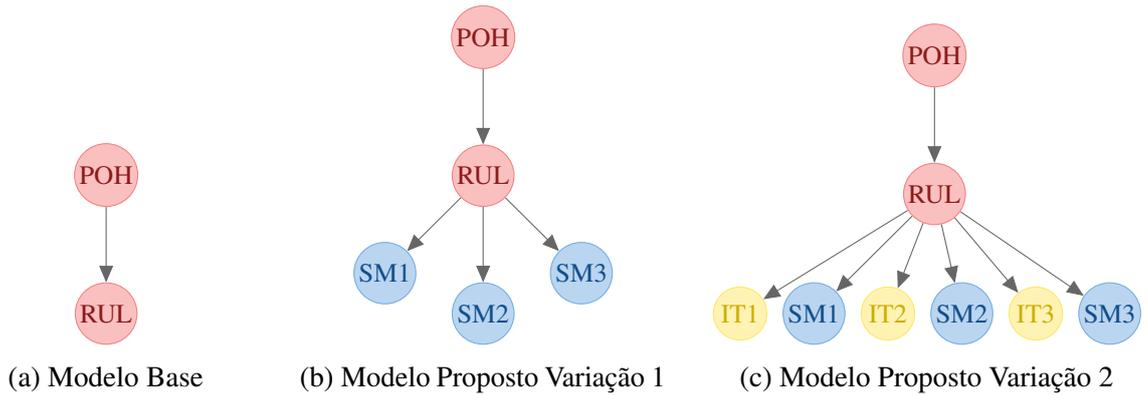


Figura 13 – Comparativo dos modelos. Figura 13a representa o modelo base, onde só é capturado o aspecto temporal. Figura 13b é a primeira variação do método proposto, onde é capturada informações sobre atributos SMART. E por fim a segunda variação que apresenta os indicadores de tendências, além dos SMART, que está representada na Figura 13c.

normalizadas. Em outras palavras, a frequência relativa dos valores que cada nó possui define a distribuição de probabilidade a posteriori.

Mais detalhadamente, calcularemos as probabilidades relacionadas a cada nó. Para o nó POH, calcularemos a probabilidade de um dispositivo falhar dado o tempo de funcionamento dele:

$$P_{\text{falhar}}(\text{POH} = t_i), \quad (3.1)$$

onde  $i = 1, \dots, k_{\text{POH}}$  e  $k_{\text{POH}}$  representa a quantidade de categorias do atributo POH. Quanto ao nó do RUL, podemos calcular as probabilidades da seguinte forma:

$$P_{\text{falhar}}(\text{RUL} = t_i | \text{POH} = t_j), \quad (3.2)$$

onde  $i = 1, \dots, k_{\text{RUL}}$ ,  $j = 1, \dots, k_{\text{POH}}$  e  $k_{\text{RUL}}$ ,  $k_{\text{POH}}$  representam respectivamente a quantidade de categorias dos atributos RUL e POH. Por fim, os cálculos das probabilidades dos nós que representam os SMART e os indicadores de tendências são realizados da seguinte forma:

$$P_{\text{falhar}}(\text{SM}_x = V_x | \text{POH} = t_j, \text{RUL} = t_i), \quad (3.3)$$

onde  $\text{SM}_x$  representa um atributo SMART ou um atributo de tendência. O  $V_x$  é um dos valores possíveis, provenientes da discretização, para o atributo associado.

Para o cálculo preciso do  $\Theta$  é necessário uma quantidade satisfatória de dados. Portanto, para melhor distribuir a probabilidade a posteriori, utilizamos a técnica de suavização adicional (CHEN; GOODMAN, 1996).

Na suavização adicional ou suavização de Laplace, como podemos ver na equação 3.4, a frequência das ocorrências de cada atributo  $z_i$  é acrescido de  $\kappa$ , chamado de parâmetro de suavização, e dividido por  $E + \kappa d$ , em que  $E$  é o tamanho da distribuição e  $\bar{d}$  é o número de observações.

$$\hat{\theta}_i = \frac{z_i + \kappa}{E + \kappa \bar{d}} \quad (3.4)$$

## 4 RESULTADOS

O método proposto foi desenvolvido em Python 2.7, utilizando os pacotes SciPy 0.17 (JONES *et al.*, 2001–) e scikit-learn 0.17 (PEDREGOSA *et al.*, 2011). Além do mais, adotamos a ferramenta Discretization-MDLPC como implementação do MDLP e o pacote libpgm 1.1 para criação e avaliação do modelo de rede bayesiana.

### 4.1 Conjunto de Dados

O dado utilizado nesse trabalho foi provido pela Backblaze company (BACKBLAZE, 2015). O dado possui 49.056 discos reais divididos em 26 diferentes modelos, variando de 1.0TB até 8.0TB no que diz respeito ao tamanho de armazenamento. Cada registro do dado é um *snapshot* diário do estado do disco rígido. O dado possui 29.747.966 registros (observações) e os discos que falham representam 4,81% do total. Um disco rígido rotulado como *não saudável* implica que o autoteste do SMART falhou, apesar de todos os registros serem rotulados como saudáveis, exceto o último.

Cada amostra do dado possui número de serial, modelo, capacidade, rótulo e 90 atributos de monitoramento de desempenho (SMART). Estes atributos apresentam valores crus e normalizados para 45 diferentes atributos SMART reportado pelo *driver*. A maioria dos dispositivos não reportam os valores para todos os atributos SMART, e por isso existem campos em branco em todas as amostras do dado. Além disso, dispositivos diferentes podem reportar estatísticas diferentes baseado no modelo e fabricante.

Antes do módulo de pré-processamento, o dado deve ser manipulado para remover atributos que existem em poucos registros e descartar amostras com poucos atributos. Nesse processo, 19.453.812 amostras e 17 colunas foram removidas.

O processo de seleção de atributos, descrito em 3.1.1, esforça-se para remover atributos desnecessários. O método RFE foi utilizado no conjunto de dados SMART em conjunto com o *Random Forest* como algoritmo preditor. Nós realizamos validação cruzada *3-fold* com a floresta de 10 árvores de decisões. A técnica selecionou 8 atributos: SMART 187 CRU, SMART 240 CRU, SMART 5 CRU, SMART 184 CRU, SMART 190 CRU, SMART 7 CRU, SMART 188 CRU, SMART 197 CRU. A tabela 3 apresenta as importâncias dos atributos retornados pelo RFE. Valores de maior importância têm um impacto mais significativo no resultado do modelo.

A maioria dos atributos obtidos pelo método do RFE são consistentes com a lista

de atributos da Backblaze (BACKBLAZE, 2015) que indicam falha eminente no disco rígido. Os atributos SMART selecionados (187, 240, 5, 184, 190, 7, 188 e 197) podem ser definidos respectivamente como: erros relatados não corrigidos, horas gastas para posicionamento da cabeça da unidade, contagem de setores reatribuídos, erros de ponta a ponta, diferença de temperatura para 100, taxa de erro de procura, tempo limite do comando e contagem pendente do setor atual.

A seguir, somente para a segunda variação do método, criamos o indicador de tendência (IT) para cada atributo selecionado pelo processo de atributos, exceto os SMART 184 e 190 que devido à análise gráfica constatamos que eles não exibem um comportamento gradual. Esses atributos são adequados para indicar falha eminente, entretanto os valores permanecem praticamente constantes durante a vida de um disco. Para criar o IT para cada HDD uma janela deslizante percorre através da série temporal realizando uma regressão linear. Para cada janela adquirimos uma equação linear aplicando o algoritmo de mínimos quadrados. Em outras palavras, obtemos o coeficiente angular da linha ajustada para os dados para cada janela. A ideia principal é representar o comportamento histórico de cada atributo.

Após a seleção de atributos relevantes, os discretizamos utilizando o MDLP, descrito na seção 3.1.2. Os pontos de corte, que representam as categorias, obtidos pela aplicação do método estão explicitados na tabela 3. O SMART 9 (POH) e RUL são discretizados utilizando o processo de categorização de largura fixa. A largura de cada categoria representa um quarto de ano, ou seja 3 meses, portanto somos capazes de prever em qual trimestre a frente o disco irá falhar com certo nível de certeza.

## 4.2 Avaliação de Desempenho

Definimos nosso modelo de acordo com a seção 3.2. Primeiramente, montamos o grafo  $G$  com os atributos originados do módulo de pré-processamento. O grafo  $G$  para a primeira variação do método é representado pela figura 14a. Já para a segunda variação temos o grafo representado na figura 14b.

Completamos a construção do método montando as TPCs, representadas por  $\Theta$ . O processo foi feito de acordo com o modelo proposto na seção 3.2, além de utilizarmos a técnica explicada na mesma seção para calcular os valores das tabelas. Assim, temos o método completo  $(G, \Theta)$ , e agora somos capazes de realizar previsões.

Dado um conjunto de observações  $O$  de um dispositivo em um determinado instante,

Atributo	Importância	Pontos de Corte	Pontos de Corte do IT
SMART 187	0.67363818	0.5, 30.5	0.015
SMART 5	0.34882314	1.5, 1540.0	-926.4, 0.5
SMART 184	0.24467154	1.5	-
SMART 7	0.23996446	258588813.5, 858525788.5, 1422218768.5, 3707446036.5, 10685231764	30503.0, 866220.1, 1392885.45, 1785367.35, 648377432.75
SMART 240	0.21005881	16464	13.85, 138083198590.0, 1.45285858722e+13, 5.03958577611e+13
SMART 190	0.15997393	18.5	-
SMART 188	0.12123306	0.5, 15032614946	0.15
SMART 197	0.08403013	0.5, 33.5	-0.15, 0.15

Tabela 3 – Importância dos atributos retornada pelo método RFE e pontos de corte gerados pelo MDLP.

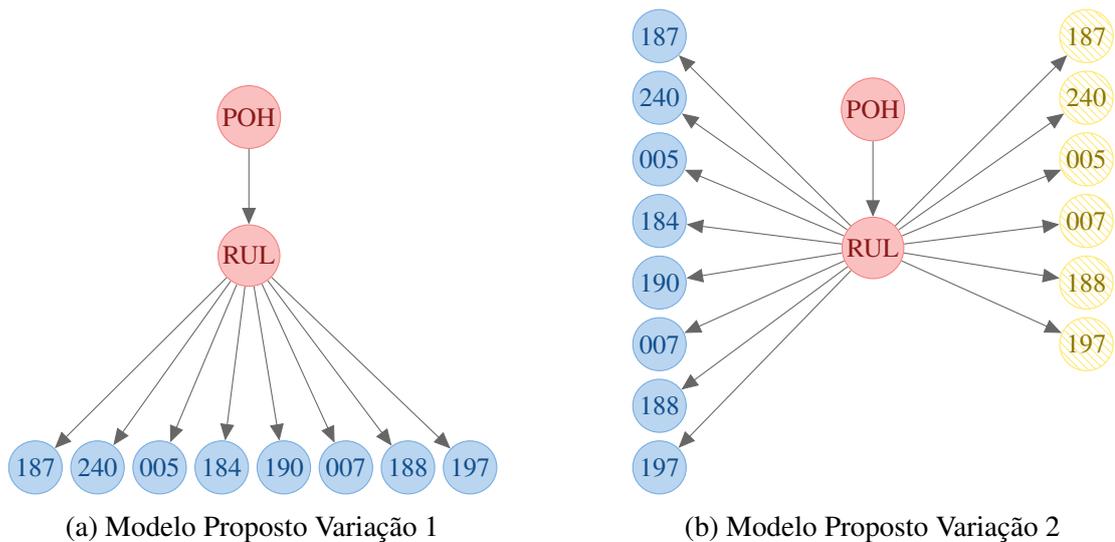


Figura 14 – Comparativo dos modelos. Figura 14a é a primeira variação do método proposto, onde é capturada informações sobre atributos SMART. E por fim a segunda variação que apresenta os indicadores de tendências, além dos SMART, que está representada na Figura 14b.

as predições são feitas da seguinte forma:

$$P_{\text{falhar}}(\text{RUL} = t_i | \text{POH} = v_{\text{POH}}, \text{SM}_1 = v_{\text{SM}_1}, \dots, \text{SM}_x = v_{\text{SM}_x}), \quad (4.1)$$

onde  $\text{SM}_1 = v_{\text{SM}_1}, \dots, \text{SM}_x = v_{\text{SM}_x}$  representam toda a informação, de atributos SMART e de tendências, necessária para realização da inferência sobre o RUL. Para o caso da primeira variação do modelo proposto, temos que  $x \in \{\text{SM}187, \text{SM}240, \text{SM}005, \text{SM}184, \text{SM}190, \text{SM}007, \text{SM}188, \text{SM}197\}$ . Para segunda variação, temos que  $x \in \{\text{SM}187, \text{SM}240, \text{SM}005,$

SM184, SM190, SM007, SM188, SM197, IT187, IT240, IT005, IT007, IT188, IT197}. Variamos o  $t_i$ , dentre as possibilidades, com o objetivo de capturarmos as probabilidades do dispositivo falhar em cada momento possível, criando assim uma distribuição de probabilidade (discreta). A inferência sempre será feita em cima do RUL, uma vez que o intuito do método é prever o tempo de vida remanescente de um determinado dispositivo.

Utilizaremos o método proposto nesse trabalho em suas duas variantes, uma Rede Neural Recorrente (*Recurrent Neural Network*, RNN) (XU *et al.*, 2016) e o modelo simples de predição de falhas em comparativo. No modelo simples, o RUL é estimado usando apenas o POH de cada disco. Como forma de avaliação dos métodos, utilizaremos três métricas (SAXENA *et al.*, 2009): PH, Desempenho  $\alpha$ - $\lambda$  e RA. É importante evidenciar que para o RNN só é possível calcular a RA, visto que o método não fornece uma distribuição do RUL, e sim um ponto estimado.

Para todos os discos rígidos, no conjunto de dados, foram executadas as três métricas. A média dos valores das métricas estão apresentados na tabela 4. Para métrica de PH foi utilizado os parâmetros  $\alpha = 1$  e  $\beta = 0.58$ . Já para a métrica de Desempenho  $\alpha$ - $\lambda$  os parâmetros escolhidos foram  $\alpha = 1$  e  $\beta = 0.50$ . Os parâmetros escolhidos para ambas as métricas foram originados de testes empíricos.

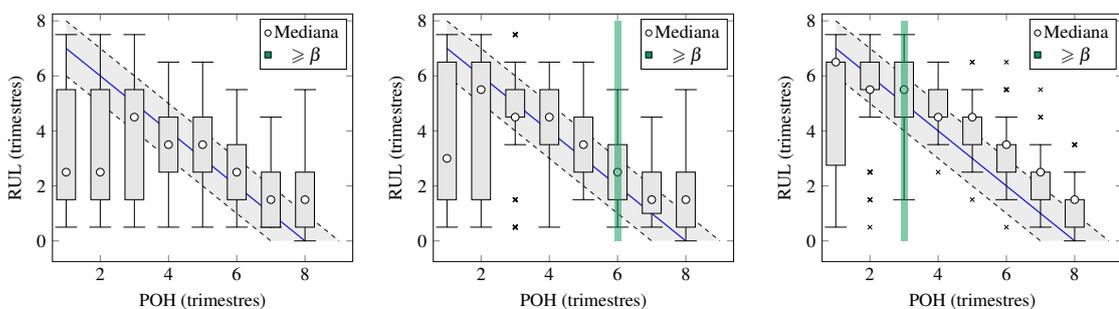
Grandes valores de  $\beta$  para baixos valores de  $\alpha$  são desejáveis, entretanto é importante encontrar um equilíbrio de valores para que seja possível a comparação dos métodos. Se escolhermos um valor para  $\alpha$  grande demais, temos que equilibrar todos métodos, uma vez que a margem de erro fica muito permissiva. Já para valores bem baixos de  $\alpha$  a margem de erro fica muito restritiva, fazendo a comparação dos métodos inviável. Temos que o comportamento do parâmetro  $\beta$  é análogo ao do  $\alpha$ , valores altos ou baixos fazem com que a métrica seja muito permissiva ou muito restritiva. Os testes evidenciaram que pequenas mudanças nos parâmetros apresentam grande influência no resultado. Mudanças de meio décimo no desempenho  $\alpha$ - $\lambda$  apresentam melhorias de 400% na comparação das variações do método proposto. Portanto, os parâmetros escolhidos procuraram maximizar a comparação dos resultados.

Métrica	Modelo Base	RNN (XU <i>et al.</i> , 2016)	Varição 1	Varição 2
PH	1.33807	-	0.70633	2.14824
RA	0.48837	0.75464	0.60559	0.72288
$\alpha$ - $\lambda$	0.00998	-	0.19901	0.29790

Tabela 4 – Resultado geral da métrica PH com os parâmetros  $\alpha = 1$  e  $\beta = 0.58$ . Resultado da métrica de RA. Resultado geral da métrica Desempenho  $\alpha$ - $\lambda$  com os parâmetros  $\alpha = 1$  e  $\beta = 0.5$ .

Como visto na tabela 4, o uso dos atributos SMART pode melhorar significativamente o desempenho dos métodos de predição de falha. Esse fato pode ser observado uma vez que o método simples apresentou o pior desempenho preditivo dos métodos comparados. Analisando o RA podemos ver que a variação 2 do método proposto e o RNN apresentam resultados similares e tiveram um desempenho significativamente melhor do que a variação 1. A pequena diferença de desempenho entre a variação 2 e o RNN, além da diferença significativa para a variação 1, evidenciam que o uso dos indicadores de tendência podem ter um impacto positivo na capacidade de predição. Considerando todas as outras métricas, a variação 2 superou os resultados dos outros métodos. Esses resultados indicam que a variação 2 pode fornecer previsões confiáveis mais cedo e mais precisamente que os outros métodos.

Mesmo que o conceito por trás do RA seja simples, o mesmo não pode ser dito sobre o PH e  $\alpha\text{-}\lambda$  uma vez que essas métricas são altamente relacionadas com os intervalos de predições. Para ilustrar os resultados das métricas, nós selecionamos alguns modelos de discos e mostramos as predições para o modelo simples, a variação 1 e a variação 2. Juntamente com as previsões, mostramos um indicador para as métricas PH e  $\alpha\text{-}\lambda$ . Figuras 15 e 16 mostram as predições para os HDDs Z301480S, W300B33Z e o valor da métrica PH.

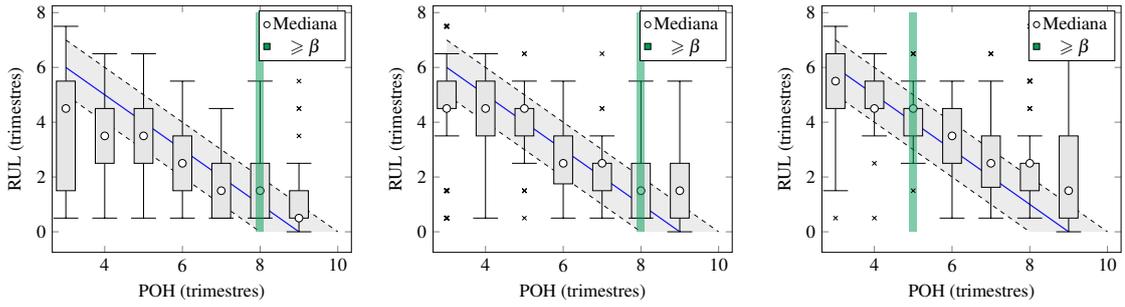


(a) Modelo Simples      (b) Método Proposto Variação 1      (c) Método Proposto Variação 2

Figura 15 – Métrica PH para o disco do número serial Z301480S com os parâmetros  $\alpha = 1$  e  $\beta = 0.58$ .

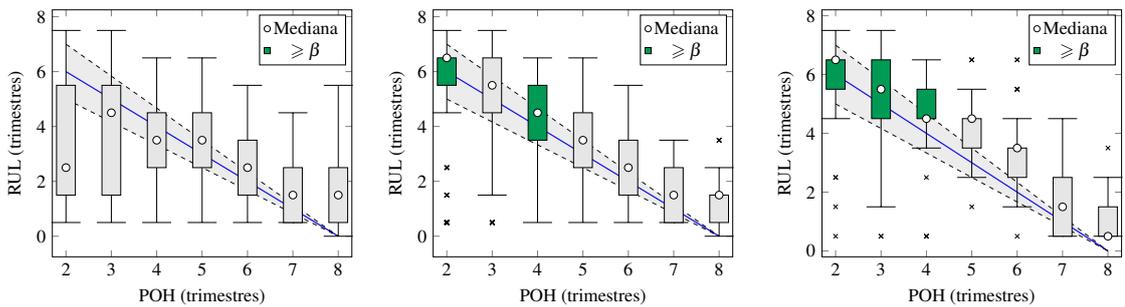
Como pode ser notado em ambas figuras 15 e 16, a variação 2 tem predições com uma incerteza reduzida quando comparada com os outros métodos. Consequentemente, produz predições confiáveis antes, de acordo com a métrica de PH. Um resultado semelhante pode ser observado na métrica  $\alpha\text{-}\lambda$ , como mostrado nas figuras 17 e 18.

Os resultados para os discos W300KL20 e W300AZTP mostram que o número de predições confiáveis aumenta do modelo simples, para a variação 1 e por fim a variação 2. Vale a pena mencionar que, embora esses números sejam apresentados apenas como ilustrações, este é



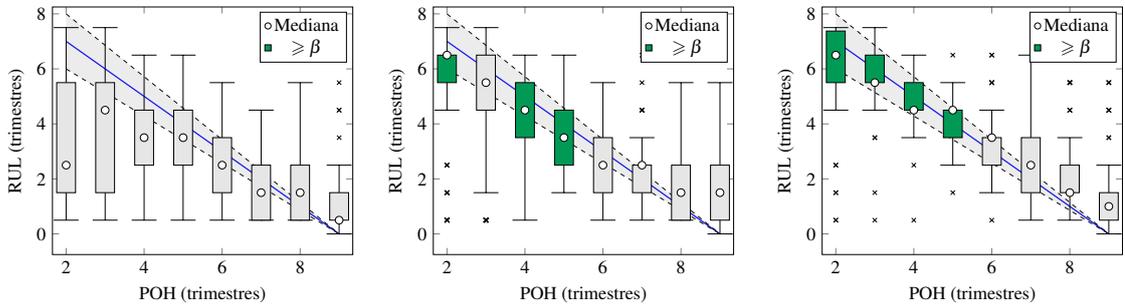
(a) Modelo Simples (b) Método Proposto Variação 1 (c) Método Proposto Variação 2

Figura 16 – Métrica PH para o disco do número serial W300B33Z com os parâmetros  $\alpha = 1$  e  $\beta = 0.58$ .



(a) Modelo Simples (b) Método Proposto Variação 1 (c) Método Proposto Variação 2

Figura 17 – Desempenho  $\alpha$ - $\lambda$  para o disco do número serial W300KL20 com  $\alpha = 1$  e  $\beta = 0.5$ .



(a) Modelo Simples (b) Método Proposto Variação 1 (c) Método Proposto Variação 2

Figura 18 – Desempenho  $\alpha$ - $\lambda$  para o disco do número serial W300AZTP com  $\alpha = 1$  e  $\beta = 0.5$ .

um comportamento típico ao considerar todos os discos do conjunto de dados.

## 5 CONCLUSÕES E TRABALHOS FUTUROS

Neste trabalho, propomos um método baseado em redes bayesiana para predição de falhas em discos rígidos. O método utiliza um subconjunto dos atributos SMART em conjunto com indicadores de tendência, criados com base nos próprios atributos, para realizar predições sobre o RUL de um dispositivo. As predições de RUL consistem em distribuições de probabilidade de quando o disco irá falhar. O método é composto por duas variações, uma com a utilização dos indicadores de tendência e a outra não.

Para avaliar o desempenho do método, conduzimos experimentos com um conjunto de dados reais compostos de 26 modelos de discos rígidos. O método foi comparado com um modelo simples de predição de falha e com redes neurais recorrentes (XU *et al.*, 2016). Vale ressaltar que avaliamos as duas variações do método. Todos os métodos foram comparados de acordo com três métricas tradicionais na área de *Prognostics and Health Management*: desempenho  $\alpha$ - $\lambda$ , horizonte de prognóstico e acurácia relativa.

Com base nos nossos experimentos, verificamos que o método proposto superou os resultados dos métodos comparados. A comparação do nosso método com a rede neural recorrente pôde ser feita somente no método de acurácia relativa, visto que a RNN não fornece predições com um grau de incerteza. Sobre o cenário da acurácia relativa, a variação 2 do método proposto apresentou um resultado similar com a RNN.

### 5.1 Trabalhos Futuros

Com base nos resultados, tivemos a possibilidade de avaliar o impacto dos indicadores de tendência na tarefa de predição. Com isso, desejamos realizar teste com redes bayesianas dinâmicas (*Dynamic Bayesian Networks*). As redes dinâmicas apresentam um objetivo semelhante ao realizado pelos indicadores de tendência.

Também foi observado, que a quantidade de dados para determinados modelos de discos é pequena. Para isso foi pensado em utilizar técnicas de transferência de aprendizado (*Transfer Learning*) para que fosse possível transferir o conhecimento de um modelo treinado, para outro modelo com uma menor quantidade de observações no conjunto de dados.

O bom resultado da RNN aponta para a utilização de redes profundas (*deep nets*) com o intuito de realçar a dependência temporal. Um algoritmo conhecido é o *long short-term memory* (LSTM) que faz a utilização de longas relações temporais para modelar a solução do

problema.

## REFERÊNCIAS

- BACKBLAZE. **Hard Drive Data and Stats**. 2015. [Online; accessed 2016-05-30]. Disponível em: <<https://www.backblaze.com/b2/hard-drive-test-data.html>>.
- BREIMAN, L. Random forests. **Machine learning**, Springer, v. 45, n. 1, p. 5–32, 2001.
- BÜCHLMANN, P.; YU, B. Analyzing bagging. **Annals of Statistics**, JSTOR, p. 927–961, 2002.
- CHEN, S. F.; GOODMAN, J. An empirical study of smoothing techniques for language modeling. In: ASSOCIATION FOR COMPUTATIONAL LINGUISTICS. **Proceedings of the 34th annual meeting on Association for Computational Linguistics**. [S.l.], 1996. p. 310–318.
- CORTES, C.; VAPNIK, V. Support-vector networks. **Machine learning**, Springer, v. 20, n. 3, p. 273–297, 1995.
- DABROWSKI, J. J.; BEYERS, C.; VILLIERS, J. P. de. Systemic banking crisis early warning systems using dynamic bayesian networks. **Expert Systems with Applications**, v. 62, p. 225 – 242, 2016. ISSN 0957-4174. Disponível em: <<http://www.sciencedirect.com/science/article/pii/S0957417416303062>>.
- DOUGHERTY, J.; KOHAVI, R.; SAHAMI, M. *et al.* Supervised and unsupervised discretization of continuous features. In: **Machine learning: proceedings of the twelfth international conference**. [S.l.: s.n.], 1995. v. 12, p. 194–202.
- FAYYAD, U. M.; IRANI, K. B. Multi-interval discretization of continuous-valued attributes for classification learning. In: BAJCSY, R. (Ed.). **Proceedings of the 13th International Joint Conference on Artificial Intelligence. Chambéry, France, August 28 - September 3, 1993**. [S.l.]: Morgan Kaufmann, 1993. p. 1022–1029. ISBN 1-55860-300-X.
- GOMES, J. P. P.; RODRIGUES, L. R.; LEÃO, B. P.; GALVÃO, R. K. H.; YONEYAMA, T. Using degradation messages to predict hydraulic system failures in a commercial aircraft. **IEEE Transactions on Automation Science and Engineering**, IEEE, 2016.
- GRANITTO, P. M.; BIASIOLI, F.; FURLANELLO, C.; GASPERI, F. Efficient feature selection for ptr-ms fingerprinting of agroindustrial products. In: **Artificial Neural Networks-ICANN 2008**. [S.l.]: Springer, 2008. p. 42–51.
- GRANITTO, P. M.; FURLANELLO, C.; BIASIOLI, F.; GASPERI, F. Recursive feature elimination with random forest for ptr-ms analysis of agroindustrial products. **Chemometrics and Intelligent Laboratory Systems**, Elsevier, v. 83, n. 2, p. 83–90, 2006.
- GUYON, I.; WESTON, J.; BARNHILL, S.; VAPNIK, V. Gene selection for cancer classification using support vector machines. **Machine learning**, Springer, v. 46, n. 1-3, p. 389–422, 2002.
- GYSELS, E.; RENEVEY, P.; CELKA, P. Svm-based recursive feature elimination to compare phase synchronization computed from broadband and narrowband eeg signals in brain–computer interfaces. **Signal Processing**, Elsevier, v. 85, n. 11, p. 2178–2189, 2005.
- HECKERMAN, D. **A Tutorial on Learning With Bayesian Networks**. [S.l.], 1995. 57 p. Disponível em: <<http://research.microsoft.com/apps/pubs/default.aspx?id=69588>>.

- HILL, N. J.; LAL, T. N.; BIERIG, K.; BIRBAUMER, N.; SCHÖLKOPF, B. An auditory paradigm for brain-computer interfaces. In: **NIPS**. [S.l.: s.n.], 2004. p. 569–576.
- JOHANNES, M.; BRASE, J. C.; FRÖHLICH, H.; GADE, S.; GEHRMANN, M.; FÄLTH, M.; SÜLTSMANN, H.; BEISSBARTH, T. Integration of pathway knowledge into a reweighted recursive feature elimination approach for risk stratification of cancer patients. **Bioinformatics**, Oxford Univ Press, v. 26, n. 17, p. 2136–2144, 2010.
- JOHNSON, B. W. **Design & analysis of fault tolerant digital systems**. [S.l.]: Addison-Wesley Longman Publishing Co., Inc., 1988.
- JONES, E.; OLIPHANT, T.; PETERSON, P. *et al.* **SciPy: Open source scientific tools for Python**. 2001–. [Online; accessed 2016-05-30]. Disponível em: <<http://www.scipy.org/>>.
- PECHT, M. **Prognostics and health management of electronics**. [S.l.]: Wiley Online Library, 2008.
- PEDREGOSA, F.; VAROQUAUX, G.; GRAMFORT, A.; MICHEL, V.; THIRION, B.; GRISEL, O.; BLONDEL, M.; PRETTENHOFER, P.; WEISS, R.; DUBOURG, V. *et al.* Scikit-learn: Machine learning in python. **The Journal of Machine Learning Research**, JMLR. org, v. 12, p. 2825–2830, 2011.
- PERKUSICH, M.; SOARES, G.; ALMEIDA, H.; PERKUSICH, A. A procedure to detect problems of processes in software development projects using bayesian networks. **Expert Systems with Applications**, v. 42, n. 1, p. 437 – 450, 2015. ISSN 0957-4174. Disponível em: <<http://www.sciencedirect.com/science/article/pii/S0957417414004916>>.
- PINHEIRO, E.; WEBER, W.-D.; BARROSO, L. A. Failure trends in a large disk drive population. In: **Proceedings of the 5th USENIX Conference on File and Storage Technologies**. Berkeley, CA, USA: USENIX Association, 2007. (FAST '07), p. 2–2.
- RODRIGUES, L. R.; GOMES, J. P. P.; FERRI, F. A. S.; MEDEIROS, I. P.; GALVAO, R. K. H.; JUNIOR, C. L. N. Use of phm information and system architecture for optimized aircraft maintenance planning. **IEEE Systems Journal**, v. 9, n. 4, p. 1197–1207, Dec 2015.
- SAXENA, A.; CELAYA, J.; SAHA, B.; SAHA, S.; GOEBEL, K. On applying the prognostic performance metrics. In: **Annual Conference of the Prognostics and Health Management Society (PHM09)**. [S.l.: s.n.], 2009.
- VACHTSEVANOS, G.; LEWIS, F. L.; ROEMER, M.; HESS, A.; WU, B. **Intelligent Fault Diagnosis and Prognosis for Engineering Systems: Methods and Case Studies**. [S.l.]: John Wiley & Sons, 2006.
- WANG, Y.; MA, E.; CHOW, T.; TSUI, K.-L. A two-step parametric method for failure prediction in hard disk drives. **Industrial Informatics, IEEE Transactions on**, v. 10, n. 1, p. 419–430, 2014.
- WANG, Y.; MIAO, Q.; MA, E.; TSUI, K.-L.; PECHT, M. Online anomaly detection for hard disk drives based on mahalanobis distance. **Reliability, IEEE Transactions on**, v. 62, n. 1, p. 136–145, 2013.
- XU, C.; WANG, G.; LIU, X.; GUO, D.; LIU, T. Y. Health status assessment and failure prediction for hard drives with recurrent neural networks. **IEEE Transactions on Computers**, v. 65, n. 11, p. 3502–3508, Nov 2016.

YET, B.; CONSTANTINOU, A.; FENTON, N.; NEIL, M.; LUEDELING, E.; SHEPHERD, K. A bayesian network framework for project cost, benefit and risk analysis with an agricultural development case study. **Expert Systems with Applications**, v. 60, p. 141 – 155, 2016. ISSN 0957-4174. Disponível em: <<http://www.sciencedirect.com/science/article/pii/S0957417416302238>>.

ZUO, Y.; KITA, E. Stock price forecast using bayesian network. **Expert Systems with Applications**, v. 39, n. 8, p. 6729 – 6737, 2012. ISSN 0957-4174. Disponível em: <<http://www.sciencedirect.com/science/article/pii/S0957417411017064>>.