U.PORTO

FEUP FACULDADE DE ENGENHARIA
UNIVERSIDADE DO PORTO

Previsão de falhas em empanques mecânicos da refinaria de Matosinhos usando modelos de *Machine Learning*

Luís Filipe Gomes Pereira

Dissertação submetida a
Faculdade de Engenharia da Universidade do Porto
para o grau de:

Mestre em Engenharia Mecânica

Orientador na FEUP:
Prof. Luís Andrade Ferreira

Orientador na PETROGAL:
Eng. Carlos Fagundes

Mestrado Integrado em Engenharia Mecânica
Faculdade de Engenharia da Universidade do Porto

Porto, Setembro de 2018

O trabalho apresentado nesta dissertação foi desenvolvido na
Refinaria de Matosinhos da PETROGAL
Departamento de Integridade e Conservação de Ativos
Petróleos de Portugal - PETROGAL, S.A.
Matosinhos, Portugal.

Luís F. Pereira (up201305738@fe.up.pt)

Resumo

A presente dissertação foi desenvolvida no âmbito do Mestrado Integrado em Engenharia Mecânica, ramo de Projeto e Construção Mecânica, da Faculdade de Engenharia da Universidade do Porto e tem como objetivo principal a aplicação de ferramentas de *Machine Learning* a dados recolhidos e armazenados em contínuo na refinaria de Matosinhos para prever falhas em empanques mecânicos. Inicialmente, elabora-se sobre a origem dos dados (*software* SAP e base de dados RTDB). De seguida, analisam-se os registos do SAP e identifica-se a necessidade de estudar os empanques mecânicos das bombas centrífugas multicelulares analisadas. Posteriormente, define-se quantitativamente o modo de falha *fuga empanque*, processam-se os dados e analisam-se os empanques mecânicos à luz de informação derivada a partir dos dados já existentes. Funções Matlab foram desenvolvidas para a obtenção dessa informação. Por fim, apresentam-se os conceitos associados ao *Machine Learning*, fazem-se considerações acerca da aplicação do *Machine Learning* na manutenção e apresentam-se os resultados obtidos. O trabalho desenvolvido permitiu concluir que há um enorme potencial na aplicação de ferramentas de *Machine Learning* na manutenção e desenvolver uma metodologia onde futuros projetos se podem apoiar. Demonstrou também que os dados armazenados estão a ser subaproveitados e que as informações obtidas a partir deles são uma mais valia para a análise fiabilística.

Palavras-chave: *Machine Learning*, Manutenção Preditiva, *Big Data*, Fiabilidade, Matlab, Galp, Refinaria, Empanque mecânico, Classificação

Abstract

The present dissertation was developed in the context of the Integrated Master's Degree in Mechanical Engineering, specialization in Project and Mechanical Construction, in the Faculty of Engineering of the University of Porto. The main goal is to use Machine Learning tools to predict mechanical seal failures. Data gathered and stored continuously in an oil refinery is used. First, the origin of the data is presented (software SAP and RTDB database). Then, the SAP records are analysed and the failure mode *Mechanical Seal Leakage* is identified as the critical one (for a group of centrifugal pumps). After that, the critical failure mode is quantitatively defined, the processing of data is done and a mechanical seal failure analysis, using new information obtained from the existing signals and records, is performed. Matlab functions were developed to obtain that information. Lastly, Machine Learning concepts are introduced, ideas about the application of Machine Learning in maintenance are presented and the main results are shown. The work done has allowed to conclude that the application of Machine Learning tools in mainteance has great potential and to develop a methodology that can support future projects. It also showed that the stored data is being underexploited and the obtained information is of great value for reliability analysis.

'All models are wrong, but some are useful'

George Box

'We can think of machine learning as the inverse of programming, in the same way that the square root is the inverse of the square, or integration is the inverse of differentiation. Just as we can ask "What number squared gives 16?" or "What is the function whose derivative is $x + 1$?" we can ask, "What is the algorithm that produces this output?"'

Pedro Domingos

Agradecimentos

Começo por agradecer ao professor Luís Andrade Ferreira por me ter apresentado este enorme desafio e me ter envolvido no incrível mundo do *Machine Learning*. A sua paciência, disponibilidade e aconselhamento foram essenciais para o sucesso deste projeto. As conversas que mantivemos ao longo do semestre foram fundamentais não só para o desenvolvimento da dissertação, mas sobretudo para o meu desenvolvimento pessoal e profissional.

À Galp agradeço a oportunidade de contactar com o meio industrial. Todos os profissionais com quem contactei foram fantásticos e merecem o meu sincero agradecimento. Em particular, agradeço ao engenheiro Carlos Fagundes pela sua enorme paciência e pelas conversas agradáveis que, muitas vezes, foram o berço de muitas das ideias apresentadas na dissertação. O seu entusiasmo pelo tema foi muito motivador. Agradeço também ao engenheiro Nuno Silva pelo apoio incansável e ao engenheiro António Freitas por me fazer entender o funcionamento dos empanques mecânicos e por me ajudar a definir, muitas vezes, o melhor caminho a seguir. Ao engenheiro Hugo Araújo agradeço a visita guiada pelo complexo industrial e ao senhor Armando Durães a visita guiada à fábrica dos lubrificantes e a partilha de experiências. Ainda, agradeço ao Pedro Pina pelo apoio e companheirismo.

Sendo a dissertação o fim de um ciclo, não posso deixar de agradecer também a todas as pessoas que se cruzaram comigo durante o meu percurso na FEUP. Foram muitas que, de uma ou de outra forma, me moldaram enquanto pessoa e engenheiro. Correndo o risco de esquecer alguém, tenho de agradecer de forma especial ao professor António Torres Marques pelas oportunidades que me concedeu, ao professor José Dias Rodrigues por me expandir horizontes e dar a oportunidade de lecionar e aos professores Renato Natal Jorge e Marco Parente por me introduzirem no mundo da investigação. Agradeço ainda, pelo exemplo, ao professor Paulo Tavares de Castro e ao professor José Seabra.

O percurso apenas foi realizado com sucesso porque tive a oportunidade de caminhar ao lado de pessoas fantásticas. Um obrigado especial à Inês, ao Tiago e ao Joel.

Por fim, mas de forma mais especial, agradeço aos meus pais e ao meu irmão. É por eles que todo o esforço e dedicação valem a pena.

Conteúdo

Lista de Figuras

Lista de Tabelas

Lista de acrónimos e siglas

BS	*Binary Segmentation*
CCRE	Centro de Comando da Rede Elétrica
CPS	*Ciber-Physical System*
DICA	Direção de Gestão e Conservação de Ativos
ERP	*Enterprise Resource Planning*
FN	*False-Negative*
FP	*False-Positive*
FPE	*Final Prediction Error*
HCA	*Hierarchichal Cluster Analysis*
IBM	*International Business Machines*
IoT	*Internet of Things*
kNN	*k-Nearest Neighbor*
LDA	*Linear Discriminant Analysis*
LLE	*Locally-Linear Embedding*
MEM	Método da Máxima Entropia
mRMR	*minimum-Redundancy Maximum-Relevance*
MTBF	*Mean Time Between Failures*
MTTR	*Mean Time To Repair*
NCA	*Neighbourhood Component Analysis*
OREDA	*The Offshore and Onshore Reliability Data*
PCA	*Principal Component Analysis*
RAM	*Random-Access Memory*
RMS	*Root Mean Square*
RTDB	*Real time database*
SAP	*Systeme, Anwendungen und Produkte in der Datenverarbeitung* (*Systems, Applications & Products in Data Processing*)
SN	*Segment Neighbour*
SVM	*Suppor Vector Machine*
TN	*True-Negativa*
TP	*True-Positive*
t-SNE	*t-Distributed Stochastic Neighbor Embedding*

1. Introdução

As equipas responsáveis pela manutenção têm acesso, tipicamente, a grandes quantidades de dados que, dadas as limitações das ferramentas de análise normalmente usadas, são analisados de forma independente [1]. O advento na Internet das Coisas (IoT) que, de forma simples, consiste em conectar os objetos físicos, embebendo-os para isso com componentes eletrónicos e sensores, aumentou a capacidade desses objetos recolher e partilhar dados [1].

Aliar a capacidade de recolha de dados a uma análise efetiva e integrada dos mesmos é, atualmente, um dos grandes objetivos da manutenção preditiva. A manutenção preditiva, ao contrário, por exemplo, da manutenção preventiva, procura que as intervenções para manutenção sejam efetuadas apenas quando se revelam necessárias. Para isso, baseia-se na condição dos equipamentos [2].

A condição dos equipamentos só pode ser usada efetivamente na manutenção se for possível estabelecer um intervalo de tempo, dito "P-F", entre a deteção do início da falha (P) e o momento em que existe perda da função (F) suficientemente lato para que o processo produtivo possa ser interrompido com um mínimo de perdas económicas e a intervenção de manutenção possa ser planeada de forma efetiva [3].

Combinando a elevada quantidade de dados disponível com o potencial das ferramentas de *Machine Learning* pode-se ambicionar obter um intervalo P-F razoável, sem que isso obrigue, necessariamente, à determinação da curva P-F, cuja dependência com inúmeros fatores a torna difícil de modelar.

De forma simplista, o *Machine Learning* pode ser visto como o inverso de programar [4]: a partir de conjuntos de *inputs* e *outputs* conhecidos as ferramentas de *Machine Learning* procuram o programa que permite mapear os *inputs* nos *outputs*. Esta definição é sobretudo adequada para definir *supervised learning*, um dos modos de aprendizagem em que podem ser categorizados os algoritmos de *Machine Learning*, sendo apresentadas, no Capítulo 7, definições mais cuidadas de *Machine Learning*.

Para que tal mapeamento possa ser efetuado, é necessário que os dados possam ser classificados em categorias. Com base na necessidade de determinar o intervalo P-F, essas categorias têm de ser tempos, nomeadamente tempos até à falha. Alternativamente, as ferramentas de *Machine Learning* podem ser utilizadas para distinguir entre modos de falha [5].

A elevada constância dos processos de produção da refinaria de Matosinhos (necessária para que as ferramentas de previsão baseadas em dados passados sejam capazes de fazer previsões razoáveis do futuro) aliada à capacidade de recolha de dados em contínuo das

unidades mais recentes tornam-na um local privilegiado para a aplicação destes novos conceitos.

Além disso, verifica-se que a refinaria de Matosinhos, apesar de apresentar bons indicadores de fiabilidade (tendo em conta o *benchmarking* com a OREDA), possui algumas bombas centrífugas em que tal não se verifica. Isso deve-se, sobretudo, ao elevado número de falhas nos empanques mecânicos destas bombas comparativamente com as restantes bombas do complexo. Um caso particular onde isto se verifica é o conjunto de bombas (duas bombas, A e B, com a mesma função - redundância) designado por P-10002. Estas bombas centrífugas multicelulares são responsáveis pelo bombeamento do resíduo de vácuo proveniente do fundo da coluna de destilação a vácuo para a unidade de *visbreaking*. As características do fluido, a sua temperatura de bombeio (cerca de 350 ºC) e o tipo de bomba centrífuga permitem justificar, em parte, a baixa fiabilidade dos empanques mecânicos. Ainda assim, e tendo também em conta que a indisponibilidade simultânea de ambas as bombas origina a paragem da unidade, é necessário olhar para o "problema" de diferentes ângulos e procurar soluções que o minimizem. Um aproveitamento superior dos dados já disponíveis pode ser a chave para desbloquear um problema que não apresenta solução técnica fácil.

1.1 A refinaria de Matosinhos

O presente projeto foi realizado na refinaria de Matosinhos da Galp. Esta refinaria, em conjunto com a refinaria de Sines, tem uma capacidade de processamento diária de 330 mil barris, o que equivale a cerca de 20% da capacidade de refinação ibérica [6].

A dissertação foi elaborada na área da Fiabilidade e Engenharia da Direção de Gestão e Conservação de Ativos (DICA). O projeto foi elaborado em estreita relação com os profissionais das secções de Inspeção Dinâmica e Gestão do Desempenho de Equipamentos e Renovação.

A Fiabilidade e Engenharia é responsável por [7]:

1. Definir e implementar processos de gestão de Fiabilidade e Gestão de Ativos Físicos com vista a assegurar elevados níveis de disponibilidade e segurança na operação, atuando nos vetores da fiabilidade e manutibilidade dos equipamentos, dos processos de fabrico, da operação do aparelho produtivo e da fiabilidade humana.
2. Realizar investimentos com profissionalismo, assegurando as melhores práticas ambientais, de segurança e gestão documental. Aplicar e desenvolver as melhores regras de arte tendo como fim último a fiabilidade das instalações e a criação de valor.
3. Promover um ambiente de confiança e motivação que contribua para a satisfação dos colaboradores, orientando-os para a inovação e para a obtenção de resultados.

1.2 Objetivos

O objetivo principal da presente dissertação é aplicar ferramentas de *Machine Learning* (em particular, técnicas de classificação) a dados recolhidos e armazenados em contínuo. Mais do que encontrar um modelo que possa ser imediatamente aplicado na prática, pretende-se desenvolver uma metodologia que demonstre que é possível, com recurso apenas a dados já existentes, utilizar estas ferramentas na manutenção e, em particular, na previsão de tempos de falha. Para isso é necessário cumprir vários objetivos:

1. Aceder e analisar minuciosamente os dados existentes.
2. Combinar dados com origens distintas (*software* SAP e base de dados RTDB).
3. Estudar o funcionamento dos empanques mecânicos, por forma a definir-se *feature*s com significado.
4. Pré-processar os dados e derivar *feature*s.
5. Aplicar algoritmos de *Machine Learning*.

Para que todos os objetivos sejam cumpridos é também necessário desenvolver ferramentas que permitam obter informação relevante a partir dos sinais medidos e dos registos SAP. O desenvolvimento de tais ferramentas permite ainda o estabelecimento de um outro objetivo: voltar a analisar os dados existentes à luz das novas informações conhecidas. O *software* utilizado será, sempre que possível, em Matlab.

1.3 Estrutura da dissertação

A presente dissertação está dividida em nove capítulos.

O Capítulo 2 aborda a origem dos dados utilizados no projeto. Nele, apresenta-se as informações que podem ser retiradas de cada fonte de informação e salienta-se a necessidade de obter mais informações a partir das já existentes. Dado que essa necessidade obriga ao desenvolvimento de ferramentas Matlab, mencionam-se as funções Matlab e os macros Excel desenvolvidos para transportar os dados desde a sua origem até ao *workspace* do Matlab.

No Capítulo 3 analisam-se os registos SAP com o intuito de demonstrar que as bombas centrífugas analisadas têm uma fiabilidade baixa e identificar o problema. São apresentados diversos conceitos, como fiabilidade, disponibilidade e *modo de falha*, e os indicadores de fiabilidade tradicionais. Na análise fiabilística efetuada, além de seguir-se a metodologia tradicional, procura perceber-se de que forma a informação obtida com as ferramentas desenvolvidas afeta as conclusões observadas.

No Capítulo 4 começa por abordar-se os empanques mecânicos e apresenta-se o seu modo de funcionamento. De seguida, define-se o que se entende por *falha* de um empanque mecânico e demonstra-se que esta pode ser determinada quantitativamente. O capítulo termina com a definição de *empanque virtual*.

O Capítulo 5 versa sobre o pré-processamento dos sinais medidos e sobre a criação de informação a partir desses sinais. É neste capítulo que se apresentam as variáveis que são usadas nos modelos de *Machine Learning* e a forma como estas são obtidas. Apresentam-se ainda as metodologias usadas para reconstruir pontos em falta.

No Capítulo 6 recorre-se às variáveis obtidas no capítulo anterior para analisar cuidadosamente as intervenções nos empanques mecânicos. Informação acerca das datas de substituição e de falha e dos tempos de vida dos empanques mecânicos é apresentada. Neste capítulo aborda-se ainda o tempo de funcionamento das bombas e as variáveis associadas aos seus arranques e paragens. O capítulo termina com a apresentação de informação acerca das pressurizações.

No Capítulo 7 entra-se no *Machine Learning* propriamente dito. Apresenta-se a sua definição, os tipos de *Machine Learning* existentes, o *workflow* que deve ser seguido num projeto, o conceito de *feature engineering* e as tarefas a ele associadas e as métricas de avaliação dos modelos. A par com a apresentação dos conceitos vão sendo feitas considerações sobre

como o *Machine Learning* pode ser aplicado em manutenção.

No Capítulo 8 são apresentados os resultados da aplicação de modelos de *Machine Learning*. É feita uma breve seleção de *feature*s, a divisão e classificação dos dados e o treino e avaliação de vários modelos. A seleção exaustiva das *feature*s mais relevantes para determinados algoritmos é apresentada. Estuda-se ainda o efeito da limitação do tempo de vida dos dados inseridos nos modelos e demonstra-se a possibilidade do uso de modelos de *Machine Learning* no auxílio à compreensão de fenómenos físicos.

Por fim, no Capítulo 9 apresentam-se as conclusões e trabalhos futuros.

2. *Software* SAP e base de dados RTDB

2.1 Introdução

Os dados usados ao longo do presente projeto têm duas origens distintas: SAP (em alemão: Systeme, Anwendungen und Produkte in der Datenverarbeitung; em português: Sistemas, Aplicações e Produtos em Processamento de Dados) [8; 9] e RTDB (base de dados em tempo real). Dada a sua origem distinta, estes apresentam fiabilidades diferentes, isto é, a confiança que pode ser depositada em análises estatísticas feitas a partir deles é diferente. De facto, Joana Pinto [10], num projeto anterior realizado na refinaria, considera que os registos SAP "se encontravam incompletos, apresentando pouca precisão e homogeneidade, particularmente a nível de definição dos modos de falha, dados do equipamento e tempos envolvidos na reparação." Ainda assim, salienta-se que se observa uma melhoria da qualidade dos dados em anos mais recentes, onde é visível que estes estão mais precisos e organizados, o que demonstra a maior consciencialização da refinaria para a recolha de dados de elevada qualidade e para a manutenção de um histórico de falhas rigoroso.

Em contrapartida, os dados armazenados na RTDB são altamente fiáveis, sobretudo porque não dependem de intervenção humana (são obtidos usando instrumentos de medição). Ainda assim, tal não impede que seja necessário um tratamento rigoroso destes antes da sua utilização, nomeadamente para completar dados em falta. A falta de dados em determinados períodos de tempo pode resultar, por exemplo, de falhas nos instrumentos de medição, extração dos equipamentos de proteção ou falhas de comunicação (seja no anel do CCRE - Centro de Comando da Rede Elétrica - associado aos equipamentos de proteção, seja entre a CCRE e a RTDB).

A possibilidade de usar em simultâneo os registos SAP e os dados da RTDB permite ultrapassar, em parte, as dificuldades inerentes à obtenção de registos de falha de elevada qualidade, pois é possível usar os segundos para confirmar/corrigir os primeiros. Isso é sobretudo verdade para o caso dos empanques mecânicos, dado que a análise dos sinais de temperatura e pressão do sistema de selagem destes permite verificar a correta definição das datas em que foram intervencionados e/ou substituídos.

2.2 SAP

O SAP, uma aplicação informática ERP (*Enterprise Resource Planning*), é uma ferramenta fundamental para a operação da refinaria. Quando é detetada uma anomalia, é aberta uma nota neste *software*, que irá, posteriormente, conduzir à criação de uma ordem, que

engloba os trabalhos efetuados no terreno para resolver a anomalia. Além da sua importância do ponto de vista operacional, o SAP é fundamental para a equipa de manutenção, uma vez que é neste que os registos históricos ficam armazenados. A análise subsequente destes registos permite à refinaria ter uma perceção da fiabilidade dos seus equipamentos, bem como do seu posicionamento, em termos de manutenção, face a outras refinarias (através do *benchmarking* com a OREDA).

No contexto do presente projeto, o SAP revelou-se muito importante em várias fases. Inicialmente, permitiu a determinação dos indicadores de fiabilidade tradicionais e, assim, a deteção do problema (baixa fiabilidade e custos elevados de manutenção associados às bombas centrífugas analisadas e, em particular, aos seus empanques mecânicos). Posteriormente, permitiu a divisão dos dados da RTDB por empanque mecânico (isto é, por unidade utilizada), pois, como se verá posteriormente, os dados extraídos desta não permitem distinguir entre limpezas do circuito de selagem e substituições de um dado empanque mecânico. Esta divisão é crucial, uma vez que a classificação dos dados a usar nos algoritmos de *Machine Learning* está muito dependente da correta definição das datas de substituição dos empanques.

Para cada nota/ordem do SAP podem ser extraídos diversos campos. Os seguintes revelaram-se de maior importância para o projeto: texto breve, que permite obter informação em falta noutros campos; identificação do componente e área operacional onde opera; datas de início da avaria, fim da avaria e de intervenção; duração da parada; custos totais reais; modo de falha e número de empanques mecânicos utilizados. Destes, as datas e o número de empanques mecânicos utilizados foram os mais importantes na fase de processamento dos dados para utilização nos algoritmos de *Machine Learning*.

A exportação dos registos SAP para Excel (onde podem ser analisados estatisticamente) é, para a maioria dos campos de interesse, rápida e requer apenas que haja o cuidado de se efetuarem pesquisas de registos por diferentes campos, já que alguns registos, nomeadamente os mais antigos, não têm todos os seus campos totalmente preenchidos e podem não ser encontrados com uma pesquisa mais superficial. Ainda assim, salienta-se que a informação relativa ao número de empanques mecânicos é de mais difícil acesso, sendo necessário recorrer aos registos do armazém e/ou atentar em cada ordem individualmente e consultar os componentes utilizados. Assim, e por forma a que o processo de exportação de dados seja o mais automático possível, identifica-se a necessidade de facilitar a extração do número de empanques mecânicos utilizados. Tal pode ser efetuado com ligeiras alterações no SAP (por exemplo, obrigar o programador do trabalho, que é o responsável pela requisição do material, a indicar o número de empanques utilizados; alternativamente, pode ser feito de forma automática, contanto que o preparador do trabalho introduza os componentes utilizados na intervenção) ou através da sua introdução subsequente no *software* responsável pela aplicação dos algoritmos de *Machine Learning*. A primeira sugestão tem a vantagem de permitir que esses dados possam ser usados em análises estatísticas no Excel, bem como de evitar que essa informação se perca no tempo.

Por fim, importa salientar que as datas obtidas no SAP, embora estejam, na maioria dos casos, próximas das datas reais (nomeadamente a data de fim de avaria), não têm precisão suficiente (seria necessário precisão ao minuto para serem direta e automaticamente utilizadas em conjunto com os dados da RTDB). Assim, a utilização conjunta dos dados do SAP e da RTDB implica, à partida, a necessidade de um procedimento semi-automático de definição de datas (tal será visto com mais detalhe posteriormente).

2.3 RTDB

A RTDB apresenta-se como a principal fonte dos dados utilizados no presente projeto. Desta base de dados foi possível extrair informação acerca da corrente do motor e da pressão e temperatura no circuito de selagem. De agora em diante, sempre que forem utilizadas as variáveis *I*, *PI* e *TI*, está a referir-se à corrente do motor e à pressão e temperatura do circuito de selagem extraídas da RTDB, respetivamente. O uso das mesmas variáveis não italizadas é efetuado quando se pretende referir os instrumentos de medição asssociados à obtenção de cada uma das variáveis mencionadas. Nas Figuras 2.1 a 2.3 apresentam-se exemplos típicos dos sinais extraídos para *I*, *PI* e *TI*, respetivamente.

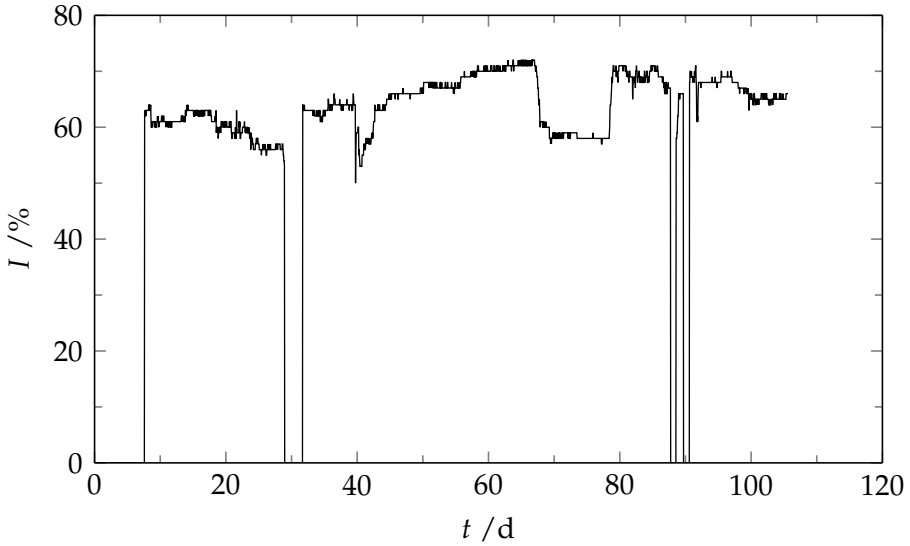


Figura 2.1: Exemplo típico do sinal de corrente extraído da RTDB.

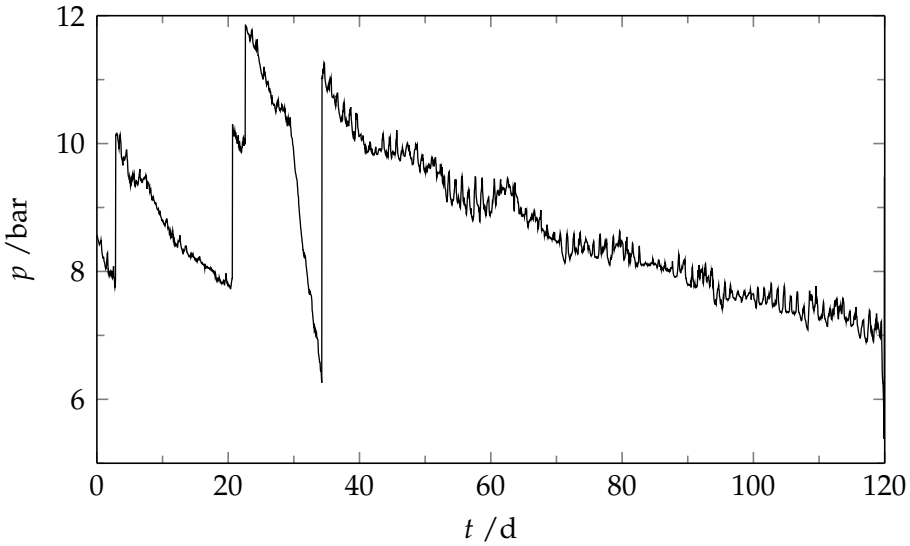


Figura 2.2: Exemplo típico do sinal de pressão extraído da RTDB.

O sinal de corrente é dado em função da percentagem de corrente nominal e varia, tipicamente, entre 0 e 70% da corrente nominal. O sinal de pressão varia tipicamente entre 6 e 12 bar. Note-se que a pressão tem declive negativo entre pressurizações (intrínseco ao

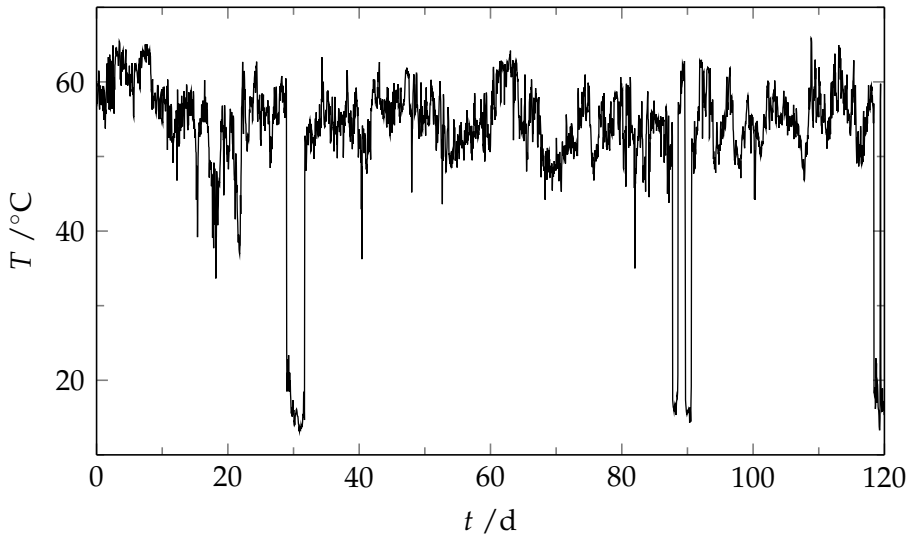


Figura 2.3: Exemplo típico do sinal de temperatura extraído da RTDB.

funcionamento de um empanque mecânico). O sinal de temperatura varia tipicamente entre a temperatura ambiente (quando a bomba está desligada) e cerca de 60ºC (quando a bomba está ligada). Na Secção 4.1 apresenta-se a localização dos instrumentos de medição de pressão e temperatura.

Atendendo a que cada bomba possui dois tipos de empanques mecânicos (acoplado e livre), verifica-se que esta base de dados permite extrair cinco variáveis para cada bomba. Esta informação, por si só, é insuficiente para que se consiga obter modelos de *Machine Learning* com precisão suficiente para a previsão de falhas. Assim, um dos objetivos propostos passou por tentar "criar dados a partir dos dados já existentes." Por exemplo, é possível obter o tempo de funcionamento da bomba com base nos dados da corrente (ou, como se verá, da temperatura), calcular o número de pressurizações usando o sinal de pressão, entre outros.

A incapacidade de prever as falhas com base apenas nas variáveis *I*, *PI* e *TI* prende-se com o facto da combinação dos seus valores num determinado instante e para um dado empanque mecânico não implicar claramente um dado estado de funcionamento, seja ele estável, pré-instável ou instável[1]. Isto é, embora a pressão e a temperatura sejam importantes do ponto de vista físico, podendo ditar a longevidade de um dado empanque mecânico, espera-se que a sua influência nos modelos de *Machine Learning* seja baixa porque para o mesmo valor de pressão e/ou temperatura existem empanques mecânicos em diferentes estados de funcionamento, ou seja, com diferentes classificações (*supervised learning*). Assim, a procura de novas variáveis não se centrará no seu impacto no tempo de vida dos empanques, mas na sua capacidade estatística de dividir os dados pelas diferentes classificações. Esta distinção entre a física do problema e a estatística é fundamental para evitar tirar conclusões precipitadas: os modelos de *Machine Learning* são modelos estatísticos.

A ideia de "criar dados a partir dos dados já existentes" pode ser vista de outro prisma: criar informação significativa a partir dos dados dos instrumentos de medição. Como se observa na Figura 2.4, este passo corresponde ao segundo nível da arquitetura proposta

[1] Ver Secção 4.2.

por Jay Lee et al. [11] para a implementação de um Sistema Ciber-Físico (Cyber-Physical System, CPS). Isto demonstra a ligação intrínseca entre termos como Indústria 4.0 (nome dado à tendência atual de automação e integração de informação nas tecnologias de manufactura e de onde é originário o conceito CPS) [12], *Big Data* (geração contínua de elevados volumes da dados) [11] e *Machine Learning* (que é o foco deste trabalho). De facto, o potencial da Indústria 4.0 apenas poderá ser atingido na totalidade se o uso de ferramentas de *Machine Learning* começar a ser recorrente, caso contrário será impossível desenvolver "máquinas inteligentes, resilientes e auto-adaptáveis" [11].

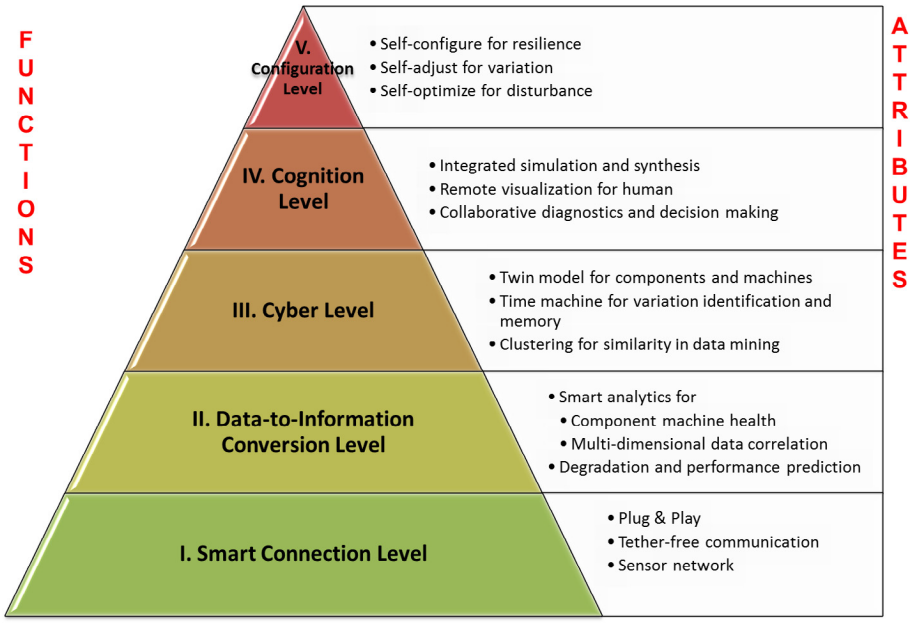


Figura 2.4: Arquitetura para implementação de um sistema ciber-físico [11].

Importa ainda ressaltar outro ponto: se é possível criar informação significativa a partir dos dados existentes, então é porque os dados estão, atualmente, a ser subaproveitados. Isso justifica que um dos objetivos do projeto seja o desenvolvimento de ferramentas (Matlab) que permitam obter de forma rápida e o mais automatizada possível essa informação adicional.

Por fim, salienta-se que é necessário exportar os dados da RTDB para Excel e, posteriormente, importá-los para Matlab, visto que é este o *software* usado na sua análise. A exportação é feita com base num procedimento muito manual e moroso. Caso se pretenda que as ferramentas desenvolvidas sejam usadas no dia-a-dia da refinaria (quer as de "obtenção de informação adicional", quer as de *Machine Learning*), será necessário encontrar uma forma alternativa de exportar os dados, visto que o procedimento atual não é viável se se pretenderem analisar vários equipamentos.

2.4 Do Excel ao Matlab

Para que as ferramentas desenvolvidas possam ser aplicadas é necessário importar os dados do Excel (já exportados do SAP e da RTDB) para o Matlab. Nesse sentido foram desenvolvidos macros Excel (código *Visual Basic* que permite substituir ações repetitivas efetuadas com o teclado e o rato) [13] e funções Matlab que permitem que o procedimento seja automático e facilmente reprodutível (sobretudo para que se possa estender a

bombas que não as analisadas neste projeto).

Os macros Excel foram desenvolvidos pela necessidade de importar os dados do SAP para o Matlab a partir de ficheiros-padrão. De forma simples, estes atuam sobre uma folha Excel principal (cujo ordem dos campos interessa e pode ser definida a partir da criação de *templates* no SAP) criando duas novas folhas. A primeira contém informação sobre as fugas por empanque mecânico. A segunda contém informação sobre as intervenções que implicaram o arrefecimento da bomba (esta informação não foi utilizada neste projeto, mas deverá ser tida em conta em projetos subsequentes). Essas duas novas folhas irão dar origem a dois novos ficheiros sobre os quais irão atuar funções Matlab para importarem os dados para o *workspace* deste último *software*.

Por sua vez, os dados da RTDB são de importação mais simples, tendo sido desenvolvidas apenas funções Matlab para proceder à sua importação. A ordem com que as variáveis estão guardadas no ficheiro Excel não é relevante, sendo que a função procura termos como “cur", “PI" e “TI." Como cada ficheiro contém informação sobre uma bomba e dois tipos de empanques mecânicos, a função questiona o utilizador acerca dos dados que correspondem a cada empanque mecânico (tipicamente os transmissores são caracterizados por uma sigla que identifica o seu tipo, seguida de um número e finalizada por uma letra que representa a bomba - e.g. TI100123A; se nada for dito em contrário, a função admite que números mais baixos estão associados a empanques do lado acoplado).

2.5 Conclusões

O capítulo atual versou sobre a origem dos dados usados ao longo do projeto (SAP e RTDB). Afirmou-se que embora os dados tenham fiabilidades diferentes, a combinação de dados de diferentes origens permite a obtenção de informação de elevada qualidade sobre as falhas (nomeadamente dos empanques mecânicos).

No que concerne aos registos SAP, mencionou-se que estes são de elevada importância para a operação e manutenção do complexo industrial e que, embora menos fiáveis que os dados da RTDB, estão atualmente mais precisos e organizados. Foram mencionados os campos que se revelaram de maior importância para o projeto, tendo sido afirmado que a utilização das diferentes datas contidas nesses registos em combinação com os dados da RTDB está sempre dependente de intervenção humana porque estes não têm (nem é viável que tenham) precisão temporal suficiente. Mencionou-se ainda que a exportação dos dados para Excel pode ser melhorada, nomeadamente em relação ao número de componentes utilizados numa dada intervenção, ainda que atualmente seja satisfatória.

Em relação ao dados da RTDB, afirmou-se que embora estes sejam muito fiáveis, a sua utilização em modelos de *Machine Learning* requer um pré-processamento cuidado. Além disso, mencionou-se que a informação contida, explicitamente, por estes é insuficiente para prever falhas, sendo por isso necessário criar informação adicional e significativa a partir dos sinais conhecidos. A possibilidade de isto ser feito demonstra que os sinais estão, atualmente, a ser subaproveitados. A generalização das ferramentas criadas (e que procuram fazer face a esse subaproveitamento) a todo o complexo industrial é apenas viável se o processo de exportação dos dados da RTDB para Excel for melhorado, visto que atualmente é muito lento e manual.

O capítulo procurou ainda consciencializar acerca da necessidade de avaliar os resultados obtidos com a aplicação de modelos de *Machine Learning* sobretudo de um ponto de vista

estatístico (e não tanto do ponto de vista físico). Apresentou ainda a interligação entre termos como Indústria 4.0, *Big Data* e *Machine Learning*. Terminou com a apresentação das ferramentas que permitem importar os dados de ficheiros Excel no *workspace* do Matlab.

3. Análise dos registos SAP e identificação do problema

3.1 Introdução

No Capítulo 1 mencionou-se que um dos principais motivos para o estudo aprofundado das bombas centrífugas que são alvo de análise no presente projeto é a sua fiabilidade inferior em relação aos restantes equipamentos do complexo industrial (e, em particular, às restantes bombas centrífugas da unidade onde estão inseridas).

De acordo com a norma NP EN 13306:2007 [14], fiabilidade é a "aptidão de um bem para cumprir uma função requerida sob determinadas condições, durante um dado intervalo de tempo." Ainda assim, do ponto de vista da refinaria importa, mais do que avaliar a fiabilidade das bombas centrífugas, avaliar a sua disponibilidade. A norma NP EN 13306:2007 [14] define disponibilidade como a "aptidão de um bem para cumprir uma função requerida sob determinadas condições, num dado instante ou durante um intervalo de tempo, assumindo que é assegurado o fornecimento dos necessários recursos externos." Note-se que um equipamento com elevada disponibilidade está apto a cumprir a sua função na maioria dos instantes de tempo em que tal é requerido. Tipicamente a fiabilidade de um componente está intrinsecamente ligada à sua disponibilidade.

A relevância dada pela refinaria ao conceito *disponibilidade* é ainda mais evidente quando se compara a fronteira definida pela norma ISO 14224:2016 [15] para o sistema bomba centrífuga (Figura 3.1) com a fronteira usada por esta para calcular os indicadores de fiabilidade (os filtros usados na filtragem do fluido bombeado são considerados no interior do sistema bomba centrífuga). Esta consideração piora, como se verá, os indicadores de fiabilidade mas, em contrapartida, permite obter informação mais precisa acerca da disponibilidade dos equipamentos. Ainda assim, para um *benchmarking* mais rigoroso da fiabilidade dos equipamentos, é importante obter os indicadores de fiabilidade para o sistema definido pela norma. No que concerne à fronteira apresentada na Figura 3.1, importa referir que, ao contrário do *driver*, o acoplamento (que permite a transmissão de potência entre o motor elétrico e a bomba centrífuga) é considerado parte do sistema bomba centrífuga.

A importância dada ao conceito *disponibilidade* demonstra que num contexto industrial é a capacidade de produção que determina a forma como a manutenção encara e resolve os problemas. Desta forma, mais do que maximizar a disponibilidade de cada bomba em particular, importa maximizar a disponibilidade do conjunto redundante[1], isto é, deve

[1] Ver Capítulo 1.

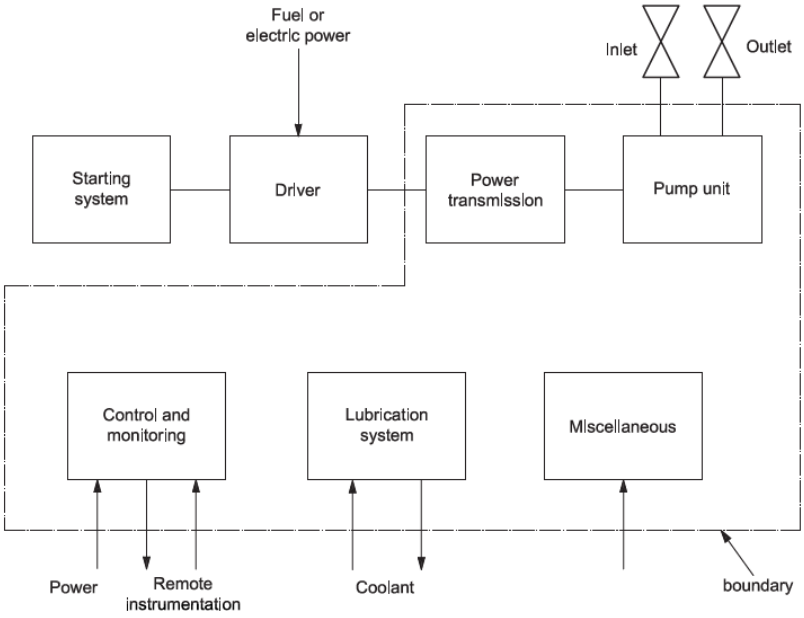


Figura 3.1: Fronteira definida pela norma ISO 14224:2016 [15] para as bombas centrífugas (imagem retirada da norma).

procurar-se que num determinado instante de tempo pelo menos uma das duas bombas centrífugas esteja disponível.

Por forma a tecer considerações sobre a fiabilidade e a disponibilidade das bombas centrífugas em análise é necessário definir os modos de falha destas. Tal é efetuado na Secção 3.2. Os indicadores de fiabilidade usados durante a análise são apresentados na Secção 3.3.

O restante capítulo procura justificar as afirmações efetuadas anteriormente acerca da baixa fiabilidade das bombas centrífugas em análise. Procura ainda demonstrar que a informação obtida através da manipulação dos dados da RTDB pode trazer benefícios à análise fiabilística tradicional.

3.2 Modos de falha

De acordo com a norma NP EN 13306:2007 [14] modo de falha é a "maneira pela qual é verificada a incapacidade de um bem para cumprir uma função requerida." Desta forma, para um sistema constituído por vários componentes, como uma bomba centrífuga, a maneira pela qual este deixa de cumprir a sua função, neste caso e de uma forma geral, bombear fluido, está intimamente relacionada com a aptidão dos seus componentes (também eles sistemas) para cumprirem as suas funções. Isto é, quando os subsistemas falham é comum que o sistema principal também perca a capacidade de cumprir a sua função, ou, pelo menos, que fique inapto a cumpri-la de forma segura. Assim, num complexo industrial de grandes dimensões e com milhares de equipamentos, uma forma simples de manter um registo cuidado das avarias de um sistema consiste em identificar, durante a realização das operações de manutenção, os subsistemas em falha e considerá-los como sendo o modo de falha. Para esta simplificação também contribui o facto do SAP ser, mais do que um *software* de manutenção, um *software* de gestão, onde muitas ve-

zes é dada preferência à rapidez de execução de um determinado trabalho em detrimento de um registo rigoroso das intervenções efetuadas.

Os modos de falha considerados para uma bomba centrífuga ao longo deste projeto são próximos dos considerados na refinaria (efetuaram-se ligeiras alterações devido à especificidade das bombas centrífugas em análise) e enumeram-se de seguida: acoplamento, casquilhos e/ou rolamentos, circuito de lubrificação, circuito de selagem, fuga - processo, fuga empanque, instrumentação, limpeza de filtro, pequena intervenção, reintervenção e válvulas.

Os nomes da maioria dos modos de falha definidos são auto-explicativos. Tendo presente a definição de modo de falha, rapidamente se constata que a consideração de *limpeza de filtro* e de *reintervenção* como modo de falha é forçada. Ainda assim, e tendo em conta o que foi afirmado na Secção 3.1 relativamente à importância do conceito *disponibilidade*, é sensato fazê-lo e, como se verá posteriormente, é fundamental para uma correta avaliação da disponibilidade do conjunto de bombas. Por sua vez, dentro do modo de falha *pequena intervenção* existem registos que também não devem ser considerados como falhas. Este grupo engloba, sobretudo, reaperto de componentes.

Por fim, salienta-se a necessidade de, por forma a obter registos de falhas mais completos, "granularizar-se" os modos de falha, isto é, começar a introduzir mais informação acerca das falhas nos registos. Por exemplo, para o modo de falha *fuga por empanque* poder-se-ia começar a registar qual o componente do empanque responsável pela falha do subsistema. Embora o sugerido seja de difícil implementação prática e a informação adicional possa não ser fundamental para as análises fiabilísticas tradicionais, poderá contribuir para aumentar a *accuracy* de modelos de *Machine Learning* de previsão de falhas.

3.3 Indicadores de fiabilidade

Embora seja importante definir fiabilidade e disponibilidade e os operadores que lidam com os equipamentos recorrentemente sejam capazes de os avaliar, de forma bastante precisa, em função desses conceitos, é necessário definir métricas que permitam avaliar quantitativamente a fiabilidade e a disponibilidade dos equipamentos. As métricas usadas no presente capítulo são a taxa de avarias, λ, o tempo médio entre falhas (*mean time between failures*), *MTBF*, o tempo médio de reparação (*mean time to repair*), *MTTR*, e a disponibilidade, *D*.

A taxa de avarias pode ser definida como a razão entre o número de falhas, n, e um determinado tempo, τ, e quantifica a frequência média de falhas de um equipamento (ou grupo de equipamentos) [7]. No caso da análise incidir sobre um grupo de equipamentos, o número de equipamentos m deve ser tido em conta na determinação da taxa de avarias, para esta refletir a frequência média de avarias por equipamento. Desta forma, tem-se que:

$$\lambda = \frac{n}{m\tau} \tag{3.1}$$

Dado os valores típicos deste parâmetro, este vem normalmente expresso em $/10^6$ horas, indicando, por isso, o número médio de avarias num período de 10^6 horas [16]. O tempo usado na determinação do parâmetro é, tipicamente, o tempo de calendário [7]. Ainda assim, e de forma a completar as análises efetuadas, pode também calcular-se a

taxa de avarias usando o tempo de funcionamento do equipamento. Esta última abordagem levanta questões de natureza prática (como determinar o tempo efetivo de funcionamento?) que, para as bombas centrífugas, podem ser ultrapassadas com recurso aos dados da RTDB.

Por sua vez, o tempo médio entre falhas pode ser obtido através de [7]:

$$MTBF = \frac{1}{\lambda} \tag{3.2}$$

O tempo médio de reparação é, de entre os parâmetros mencionados, aquele que é mais dependente da qualidade dos registos de falhas. Dado que, para cada registo, se conhece a data de início da avaria e a data de fim da avaria[2], uma forma simples de calcular este parâmetro consiste em determinar, para cada registo, a diferença entre ambas as datas e, admitindo que os tempos de reparação são normalmente distribuídos, obtê-lo a partir de uma média simples [7]. Assim, o tempo para reparação (*time to repair*), *TTR*, representa o período de tempo em que o equipamento esteve indisponível para operar devido às ações de manutenção [7]. Dado que as datas de início da avaria não implicam necessariamente que o equipamento se encontra inapto a cumprir a sua função (muitas vezes, e nomeadamente para o modo de falha *fuga empanque*, a data de início de avaria indica apenas a deteção de uma possível avaria), o tempo médio de reparação real é inferior ao calculado. Na Secção 6.3 demonstra-se as vantagens do recurso à RTDB para determinar os tempos médios de reparação dos empanques mecânicos.

O tempo médio de reparação é influenciado pelo tempo de abertura da ordem, pelo tempo até ao início de execução do trabalho, pelo tempo de preparação, pelo tempo efetivo de reparação, pelo tempo a aguardar o fornecimento de materiais e pelos tempos despendidos em logística [7].

Os três parâmetros definidos nos parágrafos anteriores estão intrinsecamente ligados ao conceito *fiabilidade*. Este, por sua vez e tal como afirmado na Secção 3.1, encontra-se relacionado com a disponibilidade, até porque uma consequência da elevada ocorrência de falhas é a diminuição da disponibilidade dos equipamentos [10]. Começa assim a ser evidente a relação entre a fiabilidade, a disponibilidade e a manutenção. O parâmetro disponibilidade, D, contempla essa relação e permite quantificar a disponibilidade de um dado equipamento [17]:

$$D = \frac{MTBF}{MTBF + MTTR} \tag{3.3}$$

Importa salientar que o valor absoluto da maioria das métricas definidas, embora tenha um significado bem definido, não é tão importante como a comparação dos valores destas para os diferentes equipamentos. É este procedimento que permite, num complexo industrial composto por milhares de equipamentos, identificar aqueles que devem ser alvo de uma análise mais fina.

Por fim, importa referir que todas as análises estatísticas efetuadas a um conjunto de dados têm a sua validade dependente da qualidade destes. Assim, após a recolha de todos os registos é fundamental revê-los, apreciar a sua relevância e fazer as correções necessárias por forma a que a sua fiabilidade seja aumentada.

[2] Ver Secção 2.2.

3.4 Tratamento dos registos do SAP

3.4.1 Análise por equipamento e modo de falha

Após a recolha, revisão e correção dos dados é necessário proceder ao seu tratamento e interpretação. Informação acerca do número de avarias, custos, tempo médio de reparação e duração da parada (tempo em que o equipamento esteve indisponível devido a ações da manutenção) pode ser rapidamente obtida e permite que a análise subsequente se centre nos pontos mais importantes.

Assim, na Tabela 3.1 apresenta-se o número de avarias, os custos, o *MTTR* e a duração da parada por bomba e modo de falha.

Tabela 3.1: Número de avarias, custos, MTTR e duração da parada por bomba e modo de falha

	Nº avarias	/% bomba	Custos /% bomba	MTTR /h	Duração parada /% bomba
A	**69**	**43.1**	**31.1**	**497.8**	**51.7**
Acoplamento	3	4.3	7.3	1994.8	17.4
Casquilhos/Rolamentos	2	2.9	5.7	603.7	3.5
Circuito lubrificação	2	2.9	0.1	771.2	4.5
Circuito selagem	3	4.3	0.5	77.6	0.7
Fuga - processo	4	5.8	0.5	74.4	0.9
Fuga empanque	10	14.5	79.4	1712.6	49.9
Instrumentação	2	2.9	0.2	221.2	1.3
Limpeza filtro	35	50.7	5.4	197.6	20.1
Pequena Intervenção	4	5.8	0.6	112.4	1.3
Válvulas	4	5.8	0.3	36.3	0.4
B	**91**	**56.9**	**68.9**	**353.0**	**48.3**
Acoplamento	3	3.3	1.9	113.1	1.1
Circuito selagem	6	6.6	0.2	126.1	2.4
Fuga - processo	5	5.5	0.5	112.5	1.8
Fuga empanque	20	22.0	92.5	937.8	58.4
Instrumentação	1	1.1	0.0	2.0	0.0
Limpeza filtro	48	52.7	4.0	237.2	35.4
Pequena Intervenção	5	5.5	0.4	27.4	0.4
Reintervenção	1	1.1	0.5	168.0	0.5
Válvulas	2	2.2	0.1	5.7	0.0
	160	100.0	100.0	415.4	100.0

Uma conclusão imediata da análise da Tabela 3.1 é que a bomba B avaria mais vezes (91) que a bomba A (69). Isto poderia ser justificado, por exemplo, pelo tempo de funcionamento das bombas (espera-se que a que funciona mais tempo avarie mais vezes). No entanto, como se verá posteriormente, a bomba A é a que funciona mais tempo. Um

olhar atento aos modos de falha permite concluir que a diferença do número de avarias prende-se, sobretudo, com o modo de falha *fuga empanque*. De facto, este é responsável por 22.0% das avarias da bomba B, enquanto que apenas se relaciona com 14.5% das avarias da bomba A.

Outro aspeto que merece atenção é a relevância da consideração do modo de falha *limpeza de filtro*. Note-se que este é responsável por 20.1% e 35.5% dos tempos de paragem para manutenção das bombas A e B, respetivamente. Note-se ainda que o modo de falha *limpeza de filtro* é o segundo modo de falha com *MTTR* mais elevado na bomba B. Com vista a reduzir o tempo necessário à manutenção dos filtros sugere-se que, à semelhança do que sucede noutras zonas do complexo industrial, se substitua a tecnologia de filtragem atual por filtros que não obriguem ao arrefecimento da bomba aquando da paragem para a sua substituição.

No que concerne ao tempo de paragem para manutenção importa notar que a bomba A, embora tenha sofrido menos avarias, é a que apresenta maior duração de parada, o que resulta do maior *MTTR* dos modos de falha *acoplamento* e *fuga empanque* em comparação com a bomba B. Por exemplo, embora a bomba B tenha sofrido o dobro das avarias pelo modo de falha *fuga empanque*, estas demoraram em média cerca de metade do tempo a ser resolvidas.

O modo de falha *fuga empanque*, que já se demonstrou ser responsável por um grande número de avarias e por tempos de paragens elevados, é também aquele que tem maior impacto económico. De facto, atentando na Tabela 3.2, verifica-se que este modo de falha é o responsável por 88% dos custos de manutenção do conjunto. Este é, portanto, o modo de falha que requer mais atenção, podendo definir-se como o modo de falha crítico para as bombas centrífugas em análise.

Tabela 3.2: Número de avarias, custos, MTTR e duração da parada por modo de falha para o grupo constituído pelas bombas A e B

Modo de falha	Nº avarias	/% total	Custos /% total	MTTR /h	Duração parada /% total
Acoplamento	6	3.8	3.6	1053.9	9.5
Casquilhos/Rolamentos	2	1.3	1.8	603.7	1.8
Circuito lubrificação	2	1.3	0.0	771.2	2.3
Circuito selagem	9	5.6	0.3	109.9	1.5
Fuga - processo	9	5.6	0.5	95.6	1.3
Fuga empanque	30	18.8	88.4	1196.1	54.0
Instrumentação	3	1.9	0.1	148.1	0.7
Limpeza filtro	83	51.9	4.5	220.5	27.5
Pequena Intervenção	9	5.6	0.5	65.2	0.9
Reintervenção	1	0.6	0.3	168.0	0.3
Válvulas	6	3.8	0.1	26.1	0.2
	160	100.0	100.0	415.4	100.0

A Tabela 3.2 permite a obtenção de uma visão mais global do comportamento do conjunto constituído pelas duas bombas.

3.4.2 Evolução das avarias ao longo do tempo

A análise efetuada na subsecção anterior permitiu obter uma ideia geral do comportamento das bombas centrífugas entre o momento em que foram colocadas em funcionamento e a atualidade. De modo a compreender melhor estes equipamentos, importa avaliar a evolução das avarias no tempo. Desta forma, representa-se na Figura 3.2 o número de avarias em função do ano e por bomba.

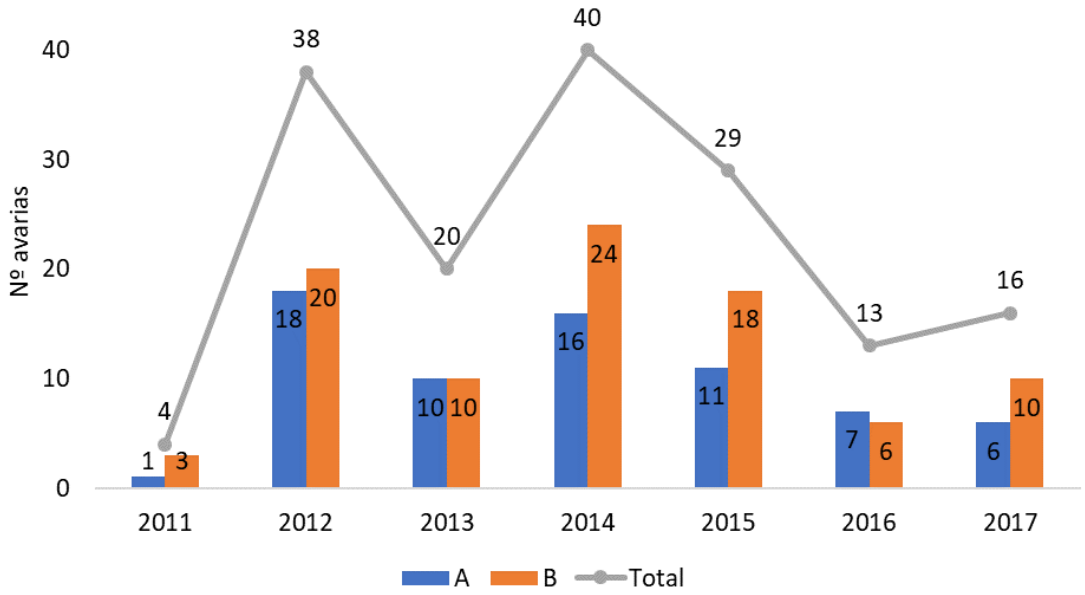


Figura 3.2: Número de avarias em função do ano e por bomba.

A conclusão de que a bomba B avaria mais do que a bomba A é verificada ao longo do tempo (apenas em dois anos tal não ocorreu). De salientar que o baixo número de avarias em 2011 se deve, sobretudo, ao menor período de tempo considerado (a unidade começou efetivamente a operar a 8 de julho de 2011). Ainda assim, poder-se-ia esperar um elevado número de avarias, visto os equipamentos se encontrarem no período de mortalidade infantil (período da vida de um sistema em que a taxa de risco decresce no tempo [17]). A justificação para tal não se verificar pode relacionar-se com o tempo de funcionamento da unidade. No entanto, como os dados da RTDB estão incompletos nesse período, esta suposição não pode ser verificada.

Note-se também que nos dois últimos anos o número de avarias foi inferior ao verificado entre 2012 e 2015, o que poderá indiciar que o período compreendido por esses anos foi um período de aprendizagem. O ano de 2018 será fundamental para perceber se o número de avarias tende a subir ou se poderá começar a estabilizar. O ano de 2013 merece também destaque por ser um ano com poucas avarias e estar situado entre os dois anos de pior desempenho. Em parte, tal pode justificar-se por uma paragem prolongada ocorrida em 2012 que pode ter sido usada para resolver avarias típicas (que deixaram de ocorrer ou passaram a ocorrer com menor frequência). Também se pode verificar, pela Figura 3.3, que nesse ano o conjunto sofreu poucas avarias relacionadas com o modo de falha *fuga empanque*.

No que concerne ao modo de falha crítico *fuga empanque* pode observar-se, na Figura 3.3, que é mais prevalecente nos anos 2012 e 2014. Estes anos podem ser considerados atípicos

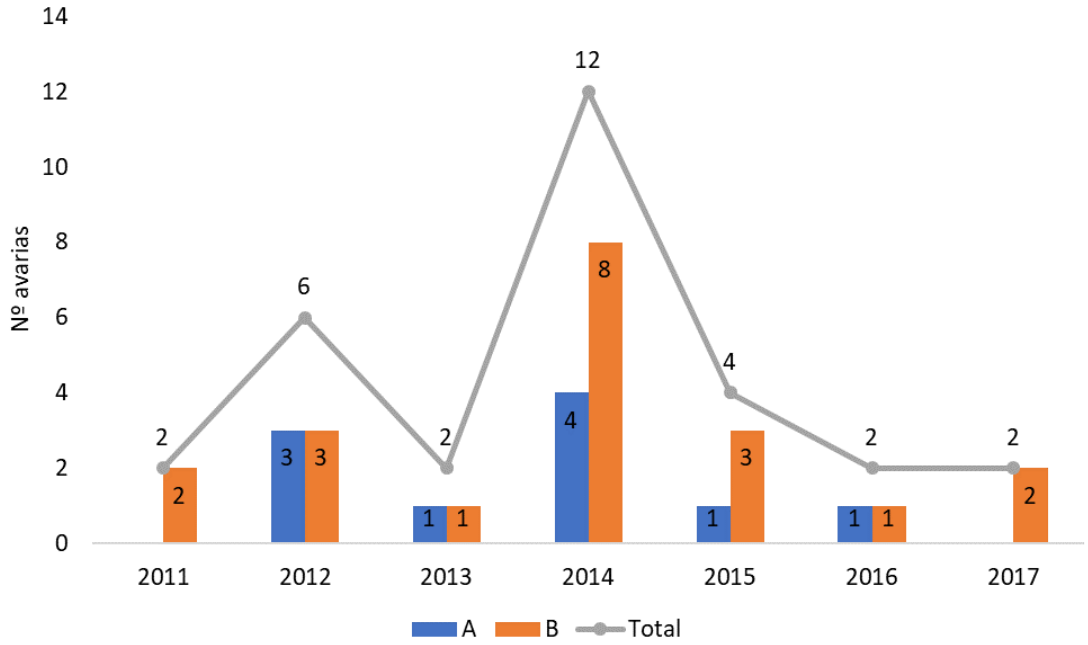


Figura 3.3: Número de avarias associadas ao modo de falha *fuga empanque* em função do ano por bomba.

em termos do número de avarias nos empanques. Mais interessante que o número de avarias associadas a este modo de falha é o número de unidades que foram utilizadas em cada ano (nem todas as avarias originaram substituição dos empanques e algumas estão associadas à substituição de mais do que um tipo de empanque). Assim, na Figura 3.4 representa-se o número de substituições de empanques em função do ano por bomba e tipo de empanque.

Na Figura 3.4 demonstra-se que, para a bomba A, o número de empanques utilizados no lado acoplado (10) foi muito superior ao usado no lado livre (2). Tal pode dever-se, por exemplo, aos procedimentos de manutenção, que diferem em função do lado do empanque. No entanto, este argumento é contrariado pelas diferenças inferiores verificadas na bomba B (acoplado: 13; livre: 16).

Difícil de justificar é a bomba B ter necessitado de 29 empanques no período em análise, enquanto que a bomba A necessitou apenas de 12, dado ambas são idênticas e cumprem a mesma função. A isto acresce a bomba B ter funcionado durante menos tempo no período em análise.

Para completar a análise da evolução das avarias ao longo do tempo atente-se na Figura 3.5, que representa o tempo conjunto de duração da parada por ano e a respetiva contribuição de cada bomba para o tempo total. Note-se que o tempo total não é real, isto é, corresponde ao somatório dos tempos associados a todas as avarias sem ter em conta que algumas delas ocorrem em simultâneo e/ou partilham parte da janela temporal associada à sua resolução. Ainda assim, este tempo é uma medida do tempo despendido pela equipa de manutenção em cada bomba. Note-se ainda a diminuição do tempo necessário para ações de manutenção nos últimos anos (que não é apenas justificada pelo menor número de avarias, dado que estas aumentaram entre 2016 e 2017). Tal demonstra que a equipa de manutenção tem sido capaz de resolver os problemas de forma mais cé-

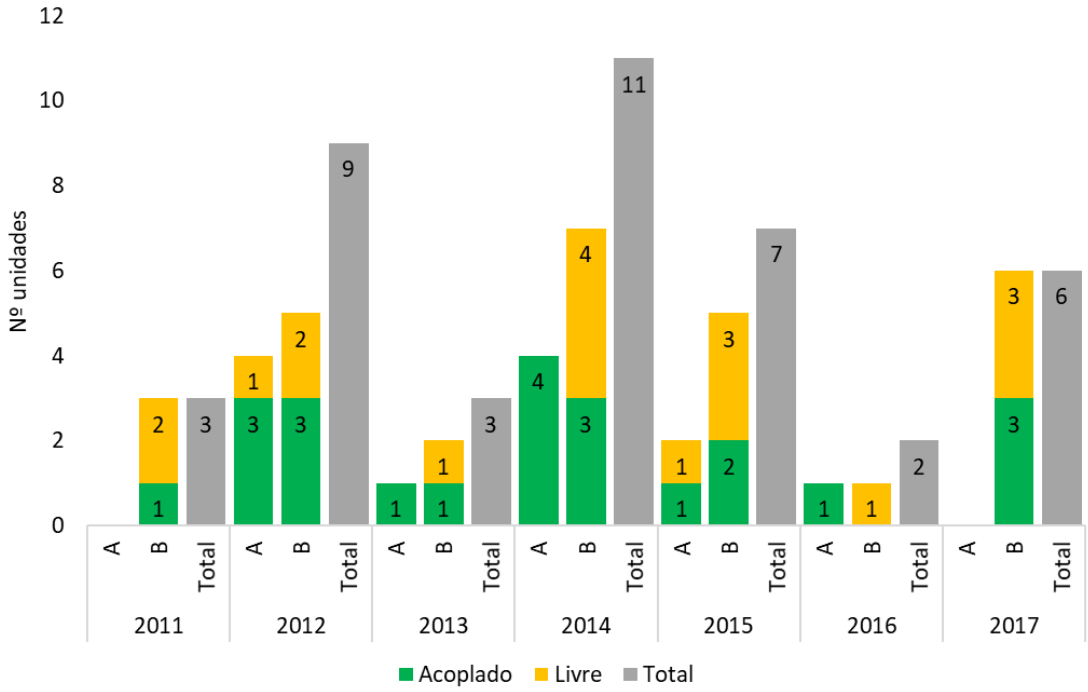


Figura 3.4: Número de substituições de empanques em função do ano por bomba e tipo de empanque.

lere e/ou que as avarias ocorridas nos últimos anos estão associadas a modos de falha de mais rápida resolução. Como se verificou anteriormente, o tempo despendido em cada bomba é equivalente.

Um exercício que pode acrescentar valor à análise é o cálculo dos tempos reais de paragem para manutenção para que, em conjunto com o conhecimento dos tempos efetivos de funcionamento, seja possível determinar a percentagem do tempo em que a bomba esteve parada por avaria e a percentagem do tempo em que esteve parada simplesmente por não ser necessária (por estar a redundante a funcionar ou por a unidade estar parada).

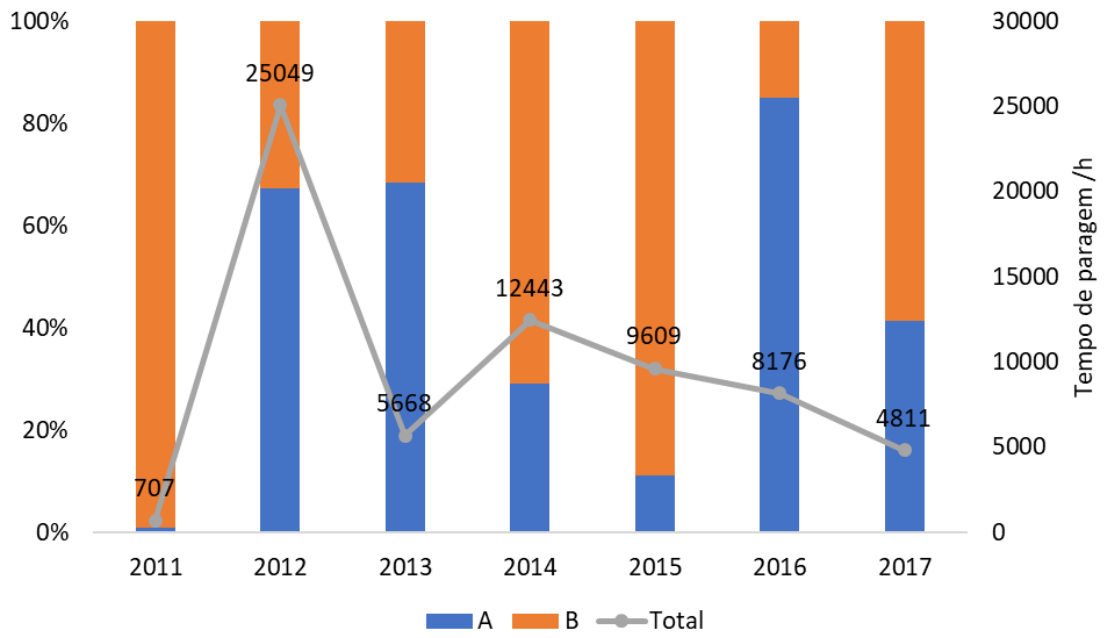


Figura 3.5: Tempo conjunto de duração da parada em função do ano e percentagem correspondente a cada bomba.

3.4.3 Taxa de avarias e disponibilidade

Para completar a análise efetuada ao longo das subsecções anteriores importa calcular os indicadores de fiabilidade apresentados na Secção 3.3.

Importância da consideração dos modos de falha associados à operação do equipamento

Na Tabela 3.3 apresenta-se os indicadores de fiabilidade e disponibilidade calculados para todos os registos e após a descaracterização[3] dos registos que, como se mencionou anteriormente, não podem realmente ser considerados falhas e faz-se a comparação entre os resultados obtidos. O ano de 2011 não é considerado nas análises subsequentes.

Tabela 3.3: Comparação dos indicadores de fiabilidade calculados usando todos os registos com os calculados usando apenas os que se consideram realmente como falha

Bomba	Todos os registos			Descaracterizando registos			Comparação		
	λ	$MTBF$	D	λ	$MTBF$	D	λ	$MTBF$	D
	$/10^{-6}$ horas	/d		$/10^{-6}$ horas	/d		/%	/%	/%
A	1292.3	32.2	0.61	589.2	70.7	0.65	-54.4	119.4	6.9
B	1672.4	24.9	0.55	665.2	62.6	0.62	-60.2	151.4	14.1
	1482.4	28.1	0.58	627.2	66.4	0.64	-57.7	136.4	10.6

Começando por atentar-se na taxa de avarias verifica-se que a bomba B apresenta menor fiabilidade que a bomba A, dado que as frequência média de avarias é superior. Importa também notar que o valor agregado da taxa de avarias, isto é, considerando o grupo de

[3]Por descaracterização entende-se a não consideração dos dados/registos na análise efetuada.

equipamentos, é uma métrica que, embora permita obter uma visão geral do conjunto, oculta informação relevante.

A descaracterização de registos tem um elevado impacto nas taxas de avarias obtidas. De facto, verifica-se que estas são mais baixas cerca de 60% em todos os casos analisados. Isto demonstra a importância do *benchmarking* ser feito com os registos descaracterizados, sob pena de se concluir que o desempenho da refinaria é muito inferior ao real.

Por sua vez, verifica-se que o *MTBF* das bombas centrífugas é cerca de um mês, o que demonstra a importância da redundância (atendendo que a indisponibilidade das bombas implica a paragem da unidade). O impacto da descaracterização de registos é também muito elevado.

No que concerne à disponibilidade, verifica-se que a descaracterização dos registos conduz a valores cerca de 10% superiores. Assim, é muito importante manter o procedimento atual: por um lado descaracterizar determinados registos para um correto *benchmarking* e, por outro, considerar todos os registos para se obter valores de disponibilidade mais próximos dos reais. É importante notar que, ao contrário da taxa de avarias e do *MTBF*, a disponibilidade contém informação acerca da atuação da equipa de manutenção.

Com o intuito de perceber a evolução no tempo da taxa de avarias e da disponibilidade, apresenta-se, nas Figuras 3.6 e 3.7, as suas evoluções, por bomba e com e sem descaracterização de registos, para os diferentes anos.

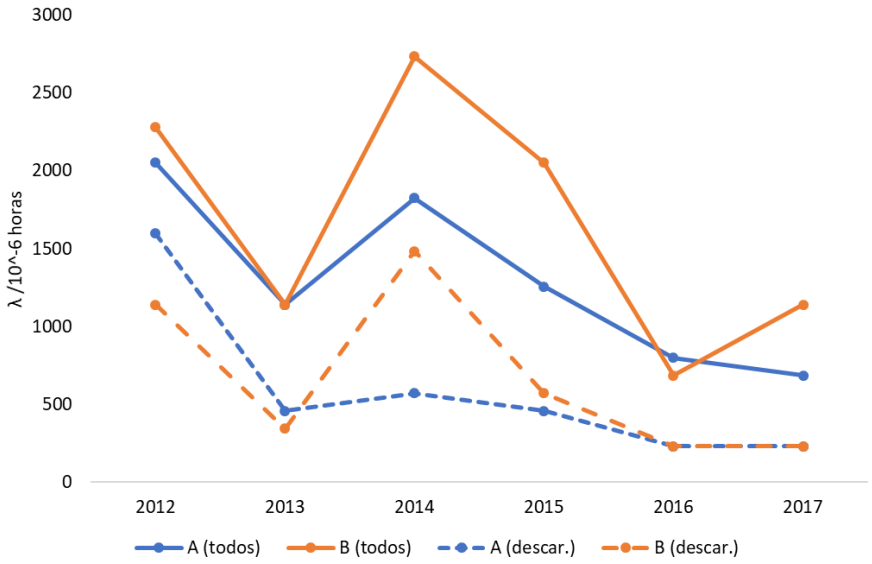


Figura 3.6: Evolução da taxa de avarias por bomba calculada usando todos os registos e calculada descaracterizando os registos que não são considerados falha.

Repare-se que a taxa de avarias tem diminuído nos últimos anos (para a bomba B aparenta subir, mas esse comportamento é devido, sobretudo, aos registos que não são realmente falhas). Note-se também que a taxa de diminuição é menor com o passar do tempo, o que corrobora a afirmação de que a fiabilidade das bombas em análise poderá vir a estabilizar num futuro próximo.

Já a disponibilidade, embora apresente um comportamento mais oscilatório que a taxa

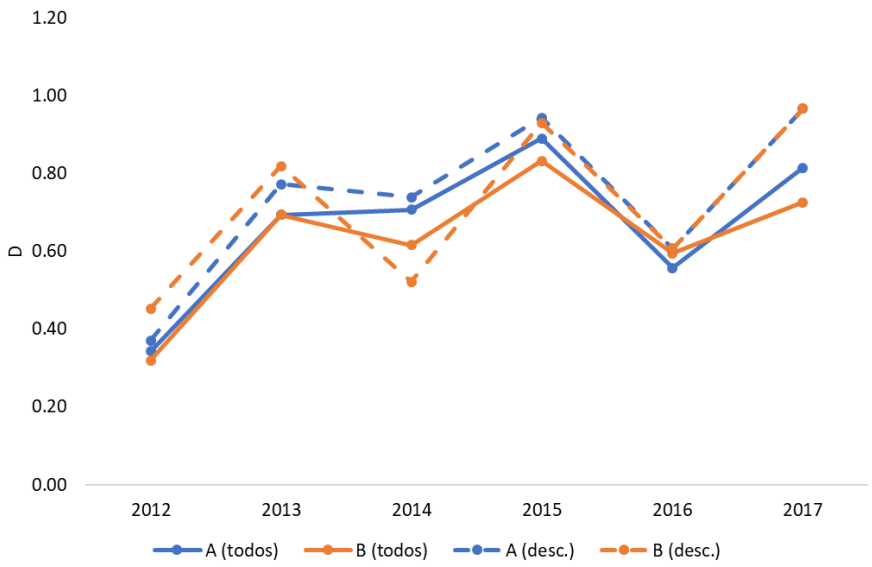


Figura 3.7: Evolução da disponibilidade calculada usando todos os registos e calculada descaracterizando os registos que não são considerados falha.

de avarias (sobretudo devido ao comportamento do *MTTR*), aparenta estar a aumentar com o tempo. Tal deve-se à conjugação de dois fatores: por um lado a diminuição da taxa de avarias (e conseguente aumento do *MTBF*) e, por outro, a maior experiência da equipa de manutenção, que permite resolver os problemas de forma mais célere (além de se poder preparar melhor para a sua ocorrência).

A análise efetuada até ao momento permitiu concluir que a bomba B apresenta menor fiabilidade que a bomba A, mas não aferir acerca do comportamento do conjunto em comparação com as restantes bombas da unidade em que está inserido. Assim, apresenta-se na Tabela 3.4 a taxa de avarias determinada pela refinaria para as bombas centrífugas da unidade onde as bombas em análise estão inseridas [7] e faz-se a comparação com os valores obtidos.

Tabela 3.4: Taxa de avarias determinada pela refinaria para as bombas da unidade onde as bombas centrífugas em análise estão inseridas [7] e comparação com a taxa de avarias calculada para as bombas em análise

Ano	λ /10^{-6} h		Comparação /%
	Análise	Refinaria	
2014	2280.6	487.1	-368.2
2015	1653.4	433.8	-281.2
2016	741.2	300.6	-146.6

Note-se desde logo que, à semelhança do que sucede com as bombas em análise, a taxa de avarias das bombas da unidade tem diminuído com o tempo. Mais importante que isso é notar que o grupo de bombas da unidade tem uma taxa de avarias muito inferior à das bombas em análise. Este aspeto faz destas bombas *bad actors*, isto é, equipamentos com comportamento muito inferior ao desejado, e justifica a necessidade de um estudo aprofundado das mesmas. Por sua vez, tinha sido concluído anteriormente que o modo

de falha *fuga empanque* era o crítico destas bombas. Assim, justifica-se a relevância dada a esse modo de falha ao longo do presente projeto. Acresce ao anteriormente exposto a clara importância dada pelos colaboradores da refinaria a equipamentos que possuem empanques mecânicos e a tentativa de perceber como a sua frequência de avarias pode ser reduzida.

Influência do tempo de funcionamento na avaliação da fiabilidade das bombas

O recurso à RTDB permitiu, após tratamento da informação daí retirada, obter o tempo de funcionamento das bombas (Tabela 3.5). A bomba A, que é a que tem menos avarias, é a que funciona mais tempo. De notar que o tempo agregado de funcionamento é muito próximo do tempo de calendário, o que demonstra que pelo menos uma das bombas está, normalmente, em funcionamento (funcionam em simultâneo em apenas 0.5% do tempo e estão ambas paradas em 5.1% do tempo). Demonstra-se assim que a unidade se encontra em operação contínua (o tempo de paragem é sobretudo devido à manutenção). Mais considerações sobre o tempo de funcionamento serão efetuadas na Secção 6.4.

Tabela 3.5: Tempo de funcionamento das bombas centrífugas por ano e percentagem de funcionamento em relação ao tempo de calendário

Ano	Bomba A		Bomba B		Conjunto	
	/h	/%	/h	/%	/h	/%
2013	4803.5	54.8	3672.7	41.9	8476.2	96.7
2014	5247.7	59.8	3416.2	39.0	8663.8	98.8
2015	4250.7	48.5	4159.5	47.4	8410.2	95.9
2016	4946.5	56.4	2976.8	33.9	7923.3	90.3
2017	5030.2	57.4	3528.8	40.2	8559.0	97.6
	24278.5	55.4	17754.0	40.5	42032.5	95.9

A consideração do tempo de funcionamento efetivo no cálculo das taxas de avarias conduz aos valores apresentados na Tabela 3.6. Apresenta-se também a comparação com os valores obtidos usando o tempo de calendário.

Tabela 3.6: Taxa de avarias calculada com o tempo de funcionamento e comparação com a calculada com o tempo de calendário

Ano	λ /10^{-6} h		Comparação /%
	$t_{calendario}$	$t_{funcionamento}$	
A	1140.3	2800.8	145.6
B	1550.8	4956.6	219.6
	1396.9	3711.4	165.7

Como seria de esperar, uma vez que o tempo de funcionamento de cada bomba é inferior ao tempo de calendário, a taxa de avarias é muito mais elevada. Ainda assim, o aspeto que importa realçar é o impacto do tempo de funcionamento ser superior na taxa de avarias da bomba B (que foi a que funcionou menos tempo). Desta forma, a fiabilidade da bomba B, em comparação com a da bomba A, vem agravada. Assim, conclui-se que a

consideração do tempo de funcionamento no cálculo da taxa de avarias permite identificar, de forma mais assertiva, os equipamentos mais críticos. Note-se que não faz sentido determinar o *MTBF* a partir das taxas de avaria obtidas (nem a disponibilidade), dado que estes parâmetros estão intrinsecamente ligados ao tempo de calendário.

3.5 Conclusões

A análise efetuada ao longo do presente capítulo permitiu concluir que as bombas centrífugas em análise merecem especial atenção (em comparação com as restantes bombas da unidade) porque apresentam taxas de avaria (que é um indicador da sua fiabilidade) muito elevadas. Permitiu também concluir que o modo de falha *fuga empanque* é o modo de falha crítico. Estas constatações associadas à tentativa dos colaboradores da refinaria em perceber como a sua frequência de avarias pode ser reduzida justificam toda a atenção que estes componentes são alvo durante o presente projeto.

A importância da qualidade dos registos de falha foi ressaltada, bem como a necessidade da “granulização" dos modos de falha para aumentar a probabilidade de sucesso da previsão de falhas usando modelos de *Machine Learning*.

Através da análise de registos de falhas foi possível concluir que a bomba B apresenta fiabilidade inferior à bomba A. Verificou-se também que o modo de falha *fuga empanque* é mais prevalecente na bomba B e que os empanques do tipo acoplado são substituídos com uma frequência muito superior aos empanques do tipo livre na bomba A (diferença pouco significativa na bomba B). Concluiu-se que a substituição da tecnologia de filtragem atual por filtros que não obriguem à paragem com arrefecimento da bomba pode resultar em tempos de paragem para manutenção muito inferiores. Concluiu-se ainda que a fiabilidade do conjunto está a aumentar e, possivelmente, irá estabilizar num futuro próximo.

A necessidade de duas análises (uma com todos os registos considerados e outra descaracterizando os registos dos modos de falha que não o são verdadeiramente) foi estudada. Por um lado, demonstrou-se que a consideração de todos os registos conduz a taxas de avaria muito elevadas que se usadas no *benchmarking* com a OREDA podem levar à conclusão que o desempenho da refinaria é inferior ao real. Por outro, provou-se que o uso de todos os registos é fundamental para se obter a verdadeira disponibilidade dos equipamentos, que é fundamental para a programação da produção da unidade.

A importância de um melhor aproveitamento dos dados da RTDB, produzindo informação a partir de dados, foi comprovada. Demonstrou-se que o conhecimento do tempo efetivo de funcionamento conduz à determinação de taxas de avarias que permitem uma identificação mais assertiva dos equipamentos mais críticos, visto que agrava a fiabilidade dos equipamentos que operam menos tempo. Foi também possível demonstrar que é possível determinar o tempo de operação da unidade a partir do conhecimento do tempo de funcionamento efetivo do conjunto de bombas. Foi ainda sugerido que a determinação do tempo real de paragem para manutenção (tendo presente as avarias que ocorrem em simultâneo e/ou partilham parte da janela temporal de paragem) permite, em conjunto com o conhecimento do tempo de funcionamento efetivo, a divisão do tempo de paragem das bombas em tempo de paragem para manutenção e tempo de paragem por não serem necessárias.

4. Empanques mecânicos

4.1 O que é um empanque mecânico?

Um empanque mecânico (Figura 4.1) é um dispositivo usado para controlar a fuga de um fluido entre um veio rotativo e a carcaça de um equipamento dinâmico [18]. O empanque de corda (*gland packing*) é uma solução alternativa ao empanque mecânico [19; 20]. Nestes, um material fibroso é enrolado em torno do veio, preenchendo fisicamente a folga existente entre o veio e a carcaça [18–20]. Relativamente a esta solução, os empanques mecânicos permitem menores perdas mecânicas devidas a fricção, menor desgaste do veio e/ou camisa, menor quantidade de fluido derramada, menores consumos de água (necessário nos empanques de corda para evitar o aquecimento excessivo), redução do tempo de manutenção e a vedação de fluidos a pressões superiores [19–21].

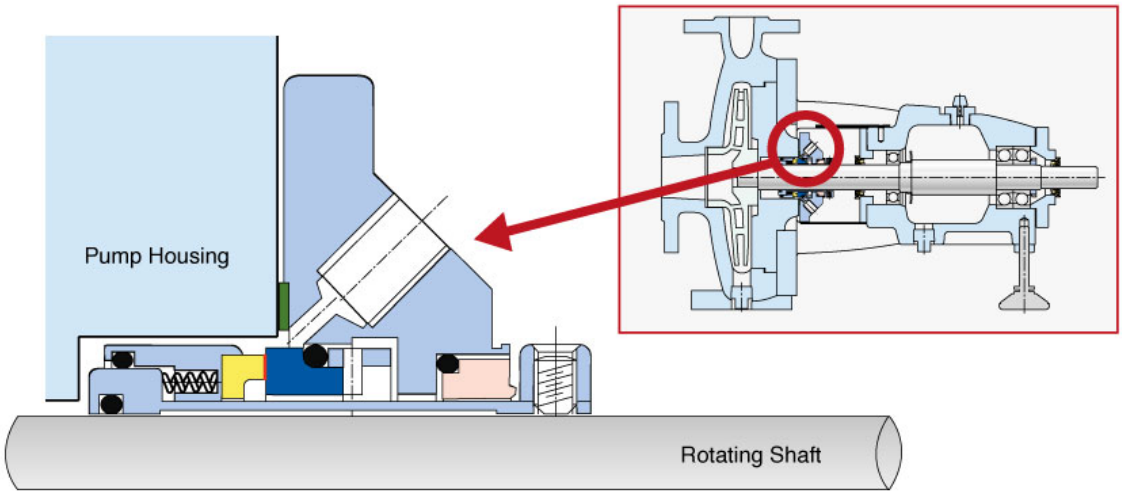


Figura 4.1: Empanque mecânico [20].

Um empanque mecânico básico tem três pontos de vedação [20]: 1) um vedante estático é montado entre a parte estacionária do empanque mecânico e a carcaça do equipamento; 2) um vedante estático é montado entre a parte rotativa do empanque mecânico e o veio; 3) um conjunto anel estacionário-anel rotativo veda o interior do equipamento dinâmico (Figura 4.2). Este último ponto de vedação é a base do *design* do empanque mecânico e é essencial para a sua efetividade [20].

As superfícies dos anéis de vedação são lapidadas [20; 22]. Esta planeza das superfícies torna os empanques mecânicos muito sensíveis a erros de montagem, dado que uma

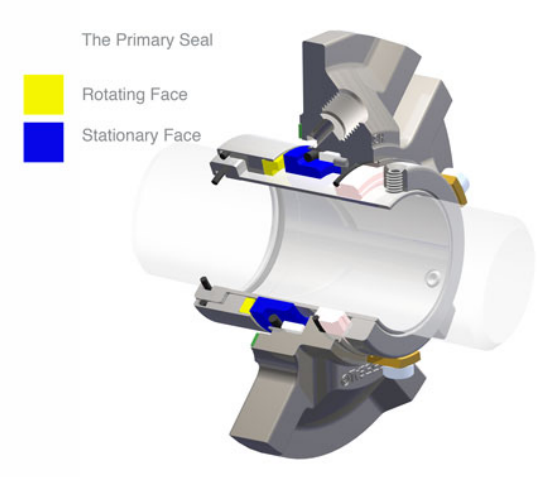


Figura 4.2: Conjunto anel estacionário-anel rotativo [20].

pequena quantidade de resíduos ou óleo, ou até mesmo impressões digitais, podem impedir o alinhamento das faces [23].

Entre as faces dos anéis dos empanques mecânicos é mantido um filme muito fino de fluido (fluido do processo ou um fluido externo) [20]. O intervalo entre as faces dos anéis, na ordem de 1 µm, impede que partículas entrem no contacto e permite que a quantidade de fuga de fluido seja mínima [20]. A ausência de fluido no contacto é conhecida como *dry running* e conduz à rápida degradação do empanque mecânico [20]. Um elemento elástico (mola ou fole) força um anel contra o outro. As forças hidráulicas presentes também influenciam o contacto [18; 20].

Embora as fugas de fluido sejam mínimas, é importante notar que estas são intrínsecas à tecnologia. As fugas não são, tipicamente, possíveis de detetar visualmente [20].

Tipicamente o anel rotativo é de grafite e roda contra um anel estacionário de carboneto de tungsténio (WC), carboneto de silício (SiC), ferro fundido "Ni-Resist," aço inoxidável ou outros cerâmicos [22; 24]. Ainda assim, em situações onde a abrasão é elevada ambos os anéis têm de ser de materiais duros, sugerindo-se, por exemplo, a combinação SiC-SiC ou SiC-WC [22].

Os empanques mecânicos são tratados na norma ANSI/API 682 [25]. Para o presente projeto importa mencionar que os empanques mecânicos são normalizados de acordo com planos. Para uma dada aplicação específica deve ser selecionado o plano mais adequado. Os empanques mecânicos das bombas analisadas são do plano 53B. Note-se que este plano apresenta dois conjuntos anel rotativo-anel estacionário e tem um circuito de selagem, que é responsável pelo arrefecimento e limpeza da câmara do empanque mecânico. A existência de dois conjuntos de anéis aumenta a fiabilidade do dispositivo (maior número de barreiras entre o fluido bombeado e o exterior). Observe-se ainda a localização dos indicadores de pressão e temperatura.

Por fim, importa mencionar que as bombas centrífugas analisadas são multicelulares e o diferencial de pressão entre a descarga e a admissão é de cerca de 33 bar [7].

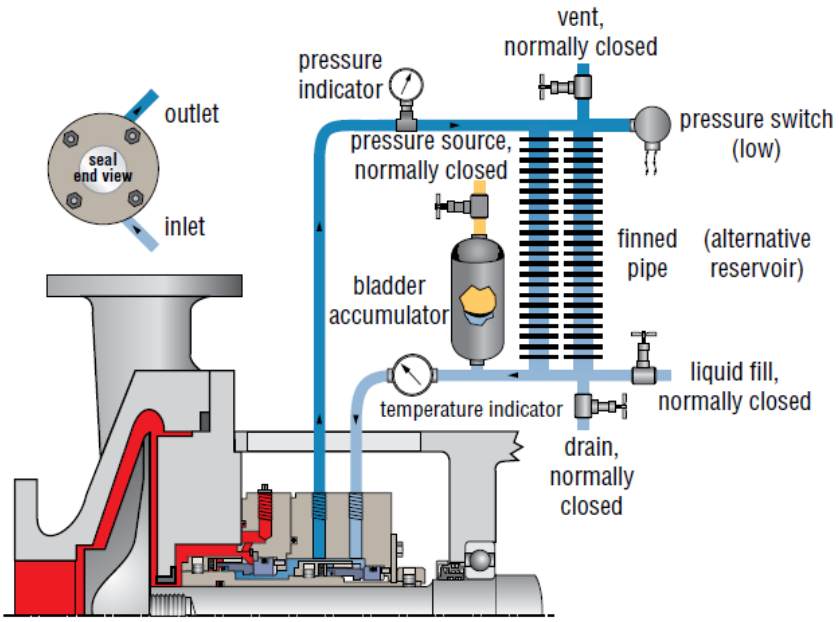


Figura 4.3: Empanque mecânico - plano 53B [26].

4.2 Definição de *falha*

De acordo com a norma NP EN 13306:2007 [14] o estado de falha de um bem define-se como "estado de um bem inapto para cumprir uma função requerida, excluindo a inaptidão devida à manutenção preventiva ou outras ações programadas, ou devida à falta de recursos externos." Atendendo a esta definição, pode entender-se instante de falha como o instante em que um bem passa de um estado de aptidão para cumprir uma função requerida para um estado de inaptidão. A diferença entre os dois estados não é, muitas vezes, evidente, sendo assim mais interessante considerar que um item, equipamento ou sistema falhou quando não consegue atingir os níveis de desempenho definidos e requeridos [27].

Assim, tendo um empanque mecânico como função estancar o fluido bombeado, poder-se-ia dizer, com base na definição da norma, que o instante de falha ocorre quando há fluido bombeado a derramar para o exterior. Tal definição é muito insatisfatória. Em primeiro lugar, porque a elevada temperatura do fluido bombeado (cerca de 350º) faz com que este não deva, em situação alguma, aceder ao exterior (caso contrário, a segurança dos operadores é colocada em risco). Em segundo lugar, e tendo em conta que cada empanque mecânico é constituído por dois conjuntos anel estacionário-anel rotativo, porque a falha do conjunto interior, apesar de não implicar fugas para o exterior, implica uma fuga de fluido para o sistema de selagem (após todo o óleo de selagem escapar para o interior da bomba), que, além de, por razões operacionais, ter de ser evitada, coloca o fluido bombeado mais próximo de aceder ao exterior. Por fim, porque a falha do conjunto exterior, apesar de não conduzir à fuga de fluido bombeado, diminui o número de barreiras entre o fluido bombeado e o exterior, diminuindo a fiabilidade do conjunto. Acresce a isto a dificuldade em quantificar uma falha definida desta forma (embora fosse fácil de detetar por inspeção visual).

Do parágrafo anterior, parece correto afirmar que a falha de um empanque mecânico coincide com a falha de um dos seus conjuntos anel estacionário-anel rotativo, já que

qualquer uma destas falhas conduz a uma diminuição do número de barreiras entre o fluido bombeado e o exterior e, portanto, há diminuição da fiabilidade do empanque mecânico. Posto isto, importa agora definir o que se entende por falha do conjunto anel estacionário-anel rotativo. Contudo, é primeiro necessário definir a sua função. Se se atentar no conjunto interior, este procura impedir que haja fuga do óleo de selagem para o interior da bomba (dado que em funcionamento normal a pressão deste é superior à pressão do fluido bombeado). Por sua vez, o conjunto exterior procura evitar que haja fuga do óleo de selagem para o exterior. Assim, pode afirmar-se que o conjunto anel estacionário-anel rotativo tem como função evitar a fuga do óleo de selagem (o que, indiretamente, permite ao empanque mecânico evitar que o fluido bombeado derrame).

Assim, uma métrica para determinar a falha de um empanque mecânico pode ser a quantidade de óleo de selagem existente no sistema. Esta, por sua vez, está relacionada com a pressão do sistema de selagem (as quedas de pressão verificadas resultam da fuga de óleo de selagem). Ou seja, a pressão do sistema de selagem pode ser usada para definir o estado de um empanque mecânico. Ainda assim, e após uma análise mais atenta, é possível concluir que o valor da pressão do sistema de selagem não permite delimitar de forma satisfatória um estado de falha de um estado de funcionamento normal. De facto, uma situação em que a pressão é 6 bar tendo diminuído 6 bar na última hora é muito mais crítica que uma situação em que a pressão é 6 bar quando no dia anterior era 7 bar. Assim, não é o valor da pressão do sistema de selagem que permite definir o instante de falha, mas a sua taxa de variação, ou seja, o declive do sinal de pressão.

Posto isto, importa definir o valor do declive crítico da pressão de selagem, ou seja, o valor da queda de pressão que permite determinar o instante de falha. Por razões operacionais, o declive crítico deve ser igual ou inferior a 8 bar/12h, ou seja, a 24 bar/dia, uma vez que este valor pressupõe a disponibilidade dos operadores para pressurizarem o empanque mecânico a cada mudança de turno. Na subsecção seguinte é discutida a escolha do declive crítico, salientando-se desde já que foi selecionado o valor de 6 bar/dia porque se entendeu que 24 bar/dia conduz a quedas de pressão mais abruptas do que, mantendo a forma de atuação atual, os operadores conseguem enfrentar.

Acrescenta-se ainda que uma queda de pressão de um determinado valor quando a pressão do óleo de selagem é elevada (maior que 8 bar) não é tão crítica quanto uma queda de pressão do mesmo valor a pressão mais baixa. De facto, observa-se que o declive tende a diminuir com a diminuição da pressão (o que faz sentido do ponto de vista físico). Assim, além de ser necessário um declive de 6 bar/dia, é também necessário que esse declive esteja associado a uma pressão inferior a 8 bar para que seja considerado que o empanque mecânico viu o seu estado alterado para falha.

Na implementação de uma função Matlab para identificar os instantes de falha teve-se ainda em conta um terceiro aspeto: o estado de funcionamento da bomba no instante de falha detetado. Isto prende-se com a incapacidade da função em determinar se uma dada queda abrupta de pressão está relacionada com uma intervenção da equipa de manutenção para, por exemplo, proceder à limpeza do circuito de selagem. Assim, caso a bomba esteja parada no instante de falha detetado, o utilizador tem de o validar. O processo de deteção de instantes de falha é, portanto, semi-automático. Para anular o efeito da presença de *outliers*, a função Matlab considera apenas declives que tenham sido determinados com três ou mais pontos.

A importância da definição de *falha* desenvolvida ao longo da presente secção e da metodologia de determinação dos instantes de falha não deve ser menosprezada. Em pri-

meiro lugar, porque é nesta definição que assentam muitas das conclusões apresentadas no Capítulo 6. Em segundo lugar e mais importante, porque a classificação dos dados introduzidos nos modelos de *Machine Learning* baseia-se, exclusivamente, nos instantes de falha determinados. Acresce a isto que sempre que se diz que um dado modelo de *Machine Learning* prevê que a falha do empanque mecânico vai ocorrer num determinado período de tempo, o que se está realmente a afirmar é que se prevê que o declive do sinal de pressão do sistema de selagem atinja e/ou ultrapasse o valor crítico nesse período de tempo.

4.2.1 Seleção do declive crítico

Um princípio que esteve sempre presente na procura do declive crítico mais adequado foi que este se deve basear, mais do que em considerações físicas, na forma de atuação dos operadores e na sua disponibilidade para executar pressurizações. Assim, é necessário definir métricas para avaliar um dado declive crítico. A métrica que se entende ser a mais adequada é o número de empanques que foram substituídos sem que tenha sido atingido o instante de falha. Se este valor for demasiado elevado, então o declive crítico está sobredimensionado: a equipa de manutenção considera que declives mais reduzidos são justificação para a substituição dos empanques mecânicos. Valores demasiado reduzidos significam que o tempo de vida útil dos empanques mecânicos está a ser subaproveitado. Deve ainda ter-se presente que, em algumas ocasiões, a substituição dos empanques pode ter sido efetuada de forma oportunista (por exemplo, aquando da paragem para substituição do empanque mecânico do lado oposto). Na Tabela 4.1 apresenta-se o número de unidades, para três declives críticos, que foram substituídas sem que o instante de falha tenha sido atingido.

Tabela 4.1: Contagem do número de unidades, por bomba e tipo de empanque mecânico, que, para diferentes declives críticos, foram substituídas sem ter sido atingido o instante de falha

		-6[1]	-12[1]	-24[1]
A	Acop.	1	4	4
	Livre	0	1	1
B	Acop.	0	1	1
	Livre	0	1	1
		1	7	7

[1] 1-bar d^{-1}

Como se referiu anteriormente, 24 bar/dia é o limite superior para o declive crítico. Como limite inferior, considerou-se 6 bar/dia, pois é razoável efetuar uma pressurização por dia.

A Tabela 4.1 demonstra que para 12 bar/dia e 24 bar/dia o número de unidades substituídas é muito elevado, o que demonstra que estes declives não são escolhas razoáveis para declive crítico. Em contrapartida, com o declive de 6 bar/dia verificou-se que apenas numa ocasião ocorreu a substituição da unidade sem que tenha sido atingido o instante de falha. Esta substituição foi atribuída, após análise dos registos SAP, a manutenção oportunista. É então razoável afirmar que o declive crítico é 6 bar/dia pois, salienta-se uma vez mais, é um declive que vai de encontro à forma de atuar dos operadores e das

equipas de manutenção.

Importa também referir que a observação do sinal de pressão corrobora as conclusões anteriores. Foi ainda observado que o instante de falha determinado com base no declive crítico define uma fronteira entre dois tipos de funcionamento que, de agora em diante, se denominam por estável e instável. Em funcionamento estável verifica-se que a pressão do empanque mecânico diminui de forma gradual no tempo (é intrínseco à tecnologia). Em oposição, funcionamento instável significa que a frequência com que ocorrem quedas severas de pressão (com declives próximos ou superiores ao crítico) é elevada. Na maioria dos casos, um funcionamento instável foi apenas corrigido com substituição das unidades ou intervenções da equipa de manutenção. Atendendo a que os empanques mecânicos do plano 53B são dimensionados para que as pressurizações necessitem apenas de ser efetuadas com um espaçamento de, pelo menos, 28 dias [28], é muito razoável afirmar que o empanque mecânico se encontra em funcionamento instável em períodos em que se requerem várias pressurizações por dia. Esta informação não foi usada na definição do declive crítico (podia ter-se considerado, por exemplo, 6 bar/28 dias) porque se entende que é demasiado otimista: são raras as situações em que o espaçamento entre pressurizações é inferior a 28 dias.

Faz-se ainda notar que em determinados períodos um empanque mecânico que se encontra em funcionamento instável (ou seja, já foi ultrapassado o instante de falha) apresenta um comportamento muito similar ao do funcionamento estável. Ainda assim, o que se observou na maioria dos casos é que após o momento em que a queda de pressão é superior ao declive crítico, o comportamento do empanque mecânico passa a ser mais imprevisível, verificando-se uma alternância, difícil de justificar, entre períodos de funcionamento aparentemente estável e períodos de funcionamento claramente instável. Por vezes, esta alternância pode ser atribuída ao estado de funcionamento da bomba: é comum a queda de pressão ser inaceitável com a bomba em funcionamento e reduzir drasticamente a partir do momento em que esta é desligada.

O que foi exposto nos parágrafos anteriores demonstra que a forma de determinação dos instantes de falha não é perfeita e dificilmente será consensual (até tendo em conta os diferentes objetivos das várias equipas que interagem num meio industrial). Ainda assim, considera-se que, face aos objetivos propostos, é muito razoável. Num eventual projeto posterior em que esta definição possa ser revista, entende-se que, mais do que variar o declive crítico, é importante introduzir outros requisitos que tenham de se verificar para que se possa considerar que o empanque mecânico falhou.

A desvantagem de considerar um declive crítico baixo é, tal como se afirmou anteriormente, o subaproveitamento da vida útil dos empanques mecânicos. Na Tabela 4.2 apresenta-se, para os três declives críticos considerados, a percentagem média dos dados que correspondem a períodos antes das falhas. É evidente que para declives superiores a percentagem média de aproveitamento é maior, dado que o instante de falha é atingido mais tarde.

4.2.2 Definição de *empanque virtual*

Após se proceder à divisão dos dados por empanque mecânico e à determinação dos instantes de falha verificou-se que, em algumas situações, os empanques mecânicos voltaram a funcionar, após um período de funcionamento instável, de forma estável durante períodos longos (próximos ou superiores ao período de funcionamento estável que antecedeu a falha). Embora algumas dessas situações possam ser atribuídas a intervenções da

Tabela 4.2: Percentagem média dos dados que correspondem a períodos antes das falhas, para diferentes declives críticos e por bomba e tipo de empanque mecânico (consideram-se apenas as unidades que atingiram a falha para todos os declives críticos)

	/%	-6[1]	-12[1]	-24[1]
A	Acop.	38.2	38.7	48.3
	Livre	56.6	61.9	62.4
B	Acop.	45.7	48.6	50.9
	Livre	28.4	35.6	40.8
		42.2	46.2	50.6

[1] 1-bar d^{-1}

equipa de manutenção (por exemplo, limpando o circuito de selagem), outras são muito difíceis de explicar. Por forma a contornar essas situações, definiu-se o conceito *empanque virtual*.

Imagine-se a situação em que é montado um determinado empanque mecânico que, após um período de funcionamento estável, falha. Admita-se que a equipa de manutenção opta por não intervir o empanque mecânico e, como este se encontra avariado, desliga a bomba. Nos dias que sucederam a falha volta a ligar a bomba e verifica que o empanque mecânico mantém o mesmo comportamento instável, o que demonstra que tem realmente algum problema. Opta então por, em vez de proceder imediatamente à substituição do dispositivo, limpar o circuito de selagem. Volta a ligar a bomba e verifica que o empanque mecânico "recuperou" do problema, apresentando agora um comportamento estável. Este comportamento mantém-se durante um longo período de tempo até que, eventualmente, o empanque mecânico volta a falhar. Do ponto de vista estatístico, e nomeadamente tendo em mente a aplicação de modelos de *Machine Learning*, faz sentido considerar que nos dois períodos de funcionamento estável foi o mesmo empanque mecânico que esteve montado? Mais, se se considerar que foi o mesmo empanque mecânico que esteve montado, faz sentido considerar que primeira falha ocorreu? Entende-se que o mais correto é considerar que estiveram montadas unidades diferentes em cada um dos períodos, cada uma delas partindo do pressuposto "as good as new." Isto equivale a dizer que entre o primeiro período de falha e a retoma de funcionamento estável ocorreu uma *substituição virtual*, denominando-se então a segunda unidade *empanque virtual*. O termo *virtual* é usado porque fisicamente o empanque mecânico é o mesmo nos dois períodos de funcionamento estável.

A necessidade de definir este conceito ficará mais clara na Secção 6.3.

5. Pré-processamento dos dados e derivação de *features*

5.1 Introdução

No Capítulo 2 afirmou-se que os dados da RTDB, embora altamente fiáveis, requerem um tratamento rigoroso, sobretudo para completar pontos em falta. Na Tabela 5.1 apresenta-se informação relativa à falta de dados. Salienta-se que as estatísticas apresentadas não se referem exclusivamente a falhas reais dos instrumentos de medição, mas também, como se mencionou na Secção 2.1, a incapacidade de armazenamento de dados por extração dos equipamentos de proteção ou falhas de comunicação.

Tabela 5.1: Número de falhas, percentagem do tempo em falha, tempo em falha, média do tempo em falha e máximo dos tempos em falha para os instrumentos de medição em função da bomba e, se aplicável, do empanque mecânico

		N° falhas		% falha		Tempo falha /h		Média /h		Máximo /h	
		Acop.	Livre	Acop.	Livre	Acop.	Livre	Acop.	Livre	Acop.	Livre
A	I	39		3.86		1746.17		44.77		648.33	
	TI	32	32	0.13	0.13	56.83	56.83	1.78	1.78	17.33	17.33
	PI	80	42	0.36	0.16	163.17	70.50	2.04	1.68	20.00	17.33
B	I	39		3.86		1746.17		44.77		648.33	
	TI	34	32	0.16	0.13	72.33	56.83	2.13	1.78	17.33	17.33
	PI	54	183	0.55	1.07	247.67	484.83	4.59	2.65	135.00	71.83

A Tabela 5.1 demonstra que a falta de dados é sobretudo gravosa para o caso da corrente, verificando-se que existem períodos de 648 horas de falta consecutiva de pontos e que em 3.86% do tempo o sinal de corrente não é armazenado. Em contrapartida, os sinais de pressão e temperatura têm muito menos pontos em falta e, com exceção do PI da bomba B, não foram armazenados, no máximo, durante 17 horas consecutivas.

As conclusões apresentadas no parágrafo anterior demonstram que a forma de encarar a reconstrução dos pontos em falta tem de ser diferente. Assim, para os sinais de pressão e temperatura, que têm menor tempo total de falta de pontos e tempos consecutivos de falta de pontos máximos mais reduzidos, opta-se por usar modelos auto-regressivos. Por sua vez, para o sinal de corrente, dada a elevada percentagem de pontos em falta e os elevados tempos consecutivos máximos de falta de dados, usa-se uma metodologia

alternativa, que recorre à segmentação do sinal de temperatura. Nas primeiras secções do presente capítulo apresenta-se cada uma das metodologias. Alguns autores [29] utilizam *Machine Learning* para reconstruir pontos em falta.

No presente capítulo apresentam-se ainda as variáveis que serão usadas nos modelos de *Machine Learning* e a forma como foram obtidas.

5.2 Reconstrução de pontos em falta

Existem muitas técnicas que podem ser usadas para reconstruir pontos em falta[1]. A escolha da mais adequada está fortemente dependente das características do sinal que se pretende completar. É importante ter em conta aspetos como a frequência de amostragem e o número de pontos em falta.

Se o número de pontos em falta for baixo pode, por exemplo, considerar-se simplesmente que os pontos em falta tomam o valor médio dos pontos adjacentes. Alternativamente, pode recorrer-se a interpolações lineares. Esta técnica é muito usada pela sua simplicidade e, se a frequência de amostragem for suficientemente alta, é uma boa solução [30]. A interpolação de *spline* cúbica é uma terceira alternativa e tem em conta que muitos sinais físicos são contínuos e têm derivadas contínuas (esta interpolação assegura que ambas as continuidades são verificadas para todos os pontos) [30].

As técnicas apresentadas no parágrafo anterior perdem eficácia quando aplicadas em situações onde o número de pontos em falta é elevado [30]. Em oposição, os modelos auto-regressivos apresentam-se como uma boa solução para estimar os pontos em falta de intervalos amplos [30]. Estes modelos permitem estimar o valor dos pontos em falta com base no comportamento do sinal na vizinhança do intervalo [30]. Nas subsecções seguintes apresenta-se a teoria em que os modelos auto-regressivos se baseiam e os métodos usados para modelar o sinal em torno do intervalo de pontos em falta.

5.2.1 Modelos auto-regressivos

Como se representa na Figura 5.1, um sinal aleatório pode ser considerado como sendo o *output* de um filtro linear causal e estável $B(z)$ cujo sinal de entrada é uma sequência estacionária não correlacionada (*white-noise*) [31]. Por causal entende-se que o seu valor de saída depende apenas do valor de entrada no mesmo instante ou em instantes passados [32]. Por sua vez, um filtro diz-se estável se entradas limitadas derem origem a saídas limitadas [32]. Dito de outra forma, um filtro é estável se todos os seus pólos (raízes do polinómio característico) estão contidos pelo círculo unitário no plano complexo [31; 33]. A estabilidade do filtro garante a estacionaridade do sinal de saída [31]. Num sinal estacionário a média não se altera com o tempo [31; 34]. A maioria dos métodos para lidar com sinais aleatórios é fortemente dependente da hipótese de estacionaridade, sendo que, caso esta hipótese não seja verificada, o sinal pode ser segmentado de modo a que cada segmento a verifique [31].

O inverso do acima exposto é também verdade: um conjunto de pontos $\{y_0, ..., y_{N-1}\}$ pode ser sujeito a um determinado filtro de forma a fornecer a sequência ϵ_n (Figura 5.2). Este filtro relaciona-se com o anterior através de [31]:

[1] Por *reconstruir pontos em falta* entende-se estimar o valor de uma determinada variável num instante onde esta não está definida.

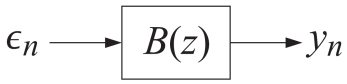


Figura 5.1: Filtro causal e estável $B\,(z)$ [31].

$$A\,(z) = \frac{1}{B\,(z)} \tag{5.1}$$

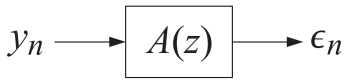


Figura 5.2: Filtro inverso $A\,(z)$ [31].

Se $A\,(z)$ é estável e causal, então pode expandir-se na seguinte forma [31]:

$$A\,(z) = \sum_{n=0}^{\infty} a_n z^{-n} \tag{5.2}$$

Que conduz a [31]:

$$\epsilon_n = \sum_{n=0}^{\infty} a_i y_{n-1} \tag{5.3}$$

Tendo presente a equação anterior e fazendo $\epsilon_n = y_n - \hat{y}_n$ obtém-se:

$$\hat{y}_n = -\left(a_1 y_{n-1} + \cdots + a_n y_0\right) \tag{5.4}$$

onde $\hat{y}_n$ é a melhor previsão linear de y_n baseada em todo o seu passado [31]. Num modelo auto-regressivo a previsão baseia-se apenas nos p pontos passados mais recentes, podendo então escrever-se [31]:

$$\hat{y}_n = -\left(a_1 y_{n-1} + \cdots + a_p y_{n-p}\right) \tag{5.5}$$

Assim, um conjunto de pontos $\{y_0, ..., y_{N-1}\}$ pode ser representado por um conjunto de parâmetros $\{a_{p1}, ..., a_{pp}; E_p\}$ [31]. E_p representa o erro quadrático médio de previsão [31]. A partir deste conjunto de parâmetros é possível prever qualquer y_n conhecidos p pontos passados.

5.2.2 Modelos auto-regressivos e pontos em falta

O problema da reconstrução de pontos em falta difere do problema da previsão apenas num aspeto: não existem apenas pontos passados, existem também pontos futuros. Uma forma de beneficiar deste aspeto é prever os pontos em falta usando ambos os conjuntos de pontos. Assim, o procedimento de reconstrução de pontos pode ser o seguinte [35]: 1) prever os pontos em falta usando os pontos passados; 2) inverter a ordem dos elementos do vetor dos pontos futuros (dado que se pretende prever para o passado) e prever os pontos em falta; 3) obter os pontos em falta a partir de uma média ponderada das duas

previsões efetuadas. Uma forma de ponderar a média é ter em conta a distância temporal entre os pontos conhecidos e os pontos previstos, dado que as previsões são tanto piores quanto maior for o afastamento temporal entre estes conjuntos de pontos.

Desta forma, o problema da reconstrução de pontos em falta resume-se agora a encontrar uma metodologia adequada para a extração dos parâmetros do modelo a partir de um bloco de sinal conhecido. Existem três métodos que são amplamente usados [31]: o método de Yule-Walker ou da autocorrelação, o método da covariância e o método de Burg ou da máxima entropia (MEM).

O método de Yule-Walker, apesar de estável e eficientemente implementável, necessita que o conjunto de pontos seja alargado (através da adição de zeros) em ambos os sentidos [31]. Se o número de pontos conhecidos for reduzido, então as previsões obtidas são negativamente afetadas [31].

O método da covariância, apesar não necessitar que o conjunto de pontos seja alargado, não tem garantida a sua estabilidade [31]. Ainda assim, tem sido usado com sucesso no processamento de voz [31].

Por sua vez, o método de Burg não requer o alargamento do conjunto de pontos e tem a estabilidade garantida [31]. Além disso, é o método usado pela função *Matlab fillgaps*, cujo objetivo é reconstruir pontos em falta [35]. Assim, no presente projeto optou-se pela utilização deste método.

5.2.3 Método de Burg (máxima entropia)

O método de Burg procura minimizar a soma dos erros de previsão de avanço, $e_p^+(n)$, e recuo, $e_p^-(n)$, (*forward and backward prediction errors*) [31]:

$$\varepsilon = \sum_{n=p}^{N-1} e_p^+(n)^2 + e_p^-(n)^2 \tag{5.6}$$

A minimização não é efetuada em relação aos coeficientes a_{pi}, mas de forma iterativa, por forma a evitar que o modelo de previsão obtido seja estável [31]. O processo iterativo é inicializado admitindo que [31]:

$$e_0^+(n) = e_0^-(n) = y_n, \text{ para } 0 \leq n \leq N-1 \tag{5.7a}$$

$$A_0(z) = 1 \tag{5.7b}$$

$$E_0 = \frac{1}{N} \sum_{n=0}^{N-1} y_n^2 \tag{5.7c}$$

Conhecido o modelo de ordem $p-1$, está-se em condições de determinar o modelo de ordem p. p deve ser menor que N, sendo N o número de pontos conhecidos. Para isso, começa por determinar-se o coeficiente de reflexão γ_p através de [31]:

$$\gamma_p = \frac{2\sum_{n=p}^{N-1} e_{p-1}^{+}(n) + e_{p-1}^{-}(n)}{\sum_{n=p}^{N-1} e_{p-1}^{+}(n)^2 + e_{p-1}^{-}(n)^2} \tag{5.8}$$

Após a determinação de γ_p pode obter-se os coeficientes do modelo de ordem p aplicando uma recursão de Levinson [31]:

$$\begin{bmatrix} 1 \\ a_{p1} \\ a_{p2} \\ \vdots \\ a_{p,p-1} \\ a_{pp} \end{bmatrix} = \begin{bmatrix} 1 \\ a_{p-1,1} \\ a_{p-1,2} \\ \vdots \\ a_{p-1,p-1} \\ 0 \end{bmatrix} - \gamma_p \begin{bmatrix} 0 \\ a_{p-1,p-1} \\ a_{p-1,p-2} \\ \vdots \\ a_{p-1,1} \\ 1 \end{bmatrix} \tag{5.9}$$

Com o intuito de se preparar a iteração seguinte pode determinar-se os erros de previsão de avanço e recuo através de [31]:

$$e_p^{+}(n) = e_{p-1}^{+}(n) - \gamma_p e_p^{+}(n-1) \tag{5.10a}$$

$$e_p^{-}(n) = e_{p-1}^{-}(n-1) - \gamma_p e_{p-1}^{+}(n) \tag{5.10b}$$

Por fim, pode ainda determinar-se o erro quadrático médio do modelo de ordem p [31]:

$$E_p = \left(1 - \gamma_p^2\right) E_{p-1} \tag{5.11}$$

Desta forma, efetuando o processo iterativo M vezes obtêm-se M modelos diferentes, cada um caracterizado por um conjunto de coeficientes e por um erro quadrático médio (para as M iterações: $\{E_0, ..., E_M\}$). Torna-se assim necessário definir uma métrica que permita escolher o modelo mais adequado de entre os determinados.

A escolha da ordem do modelo é um processo sobretudo empírico [31]. Existem numerosos critérios, tendo-se optado, pela sua simplicidade, pelo critério do erro final de previsão de Akaike (*Akaike's final prediction error*, FPE) [31; 36]. De acordo com este critério deve optar-se pelo modelo que minimiza a quantidade [31]:

$$E_M \frac{N+M+1}{N-M-1} \tag{5.12}$$

5.3 Deteção de pontos de mudança

Por ponto de mudança entende-se o instante em que uma determinada propriedade estatística de um dado sinal muda abruptamente [37–39]. Para simplificar, considera-se que o vetor dos pontos de mudança é um vetor com as posições dos elementos de um ve-

tor com os valores do sinal (ordenados de forma sequencial) onde ocorrem as mudanças abruptas.

O problema de deteção de pontos de mudança é simples caso seja conhecido à partida o número de pontos de mudança de um dado sinal e torna-se mais complexo se esse valor for desconhecido. As próximas subsecções abordam os procedimentos tipicamente adotados em ambas as situações.

5.3.1 Com número de pontos de mudança conhecido

Comece por considerar-se o caso em que um dado sinal pode ser dividido em dois segmentos (um ponto de mudança). Um procedimento que pode ser executado para determinar o ponto de mudança é [37]: 1) escolher um ponto e dividir o sinal em dois segmentos; 2) calcular a estatística desejada em cada segmento, isto é, a estatística para a qual se pretende detetar a mudança abrupta; 3) calcular os desvios de cada ponto em relação à estatística do seu segmento, de acordo com uma função de erro[2], e somá-los para determinar o erro total da divisão; 4) variar a localização do ponto de divisão do sinal; 5) repetir o procedimento anterior até se determinar o ponto de mudança que conduz ao erro total mínimo.

O procedimento anterior mantém-se exequível se o número de pontos de mudança for superior a um. Por exemplo, para dois pontos de mudança pode começar por encontrar-se o primeiro e, posteriormente, aplicar o procedimento a cada um dos novos segmentos por forma a determinar-se a melhor localização para o segundo.

O procedimento exposto no parágrafo anterior é o executado pelo método *Binary Segmentation* (BS) [39]. Este método, que é um dos mais usados na prática, é computacionalmente eficiente, mas não garante que se determine os pontos de mudança que conduzem a um erro global mínimo [39]. Por sua vez, o método *Segment Neighbour* (SN), que também é um dos mais usados, computa os erros residuais totais para todos os segmentos possíveis [39]. Ainda assim, esta procura exaustiva acarreta um elevado custo computacional [39].

Por forma a generalizar o atrás exposto, comece por considerar-se uma sequência ordenada de pontos $\boldsymbol{y_{1:n}} = (y_1, \ldots, y_n)$[3]. n é a dimensão da sequência. Admita-se que se conhece o número de pontos de mudança, m, e que $\boldsymbol{\tau_{1:m}} = (\tau_1, \ldots, \tau_m)$ é o vetor que os contém. Considere-se ainda que $\boldsymbol{\tau_{1:m}}$ é uma sequência ordenada ($\tau_i < \tau_j$, com $i < j$), $1 \leq \tau_i \leq n-1$, $\tau_0 = 0$ e $\tau_{m+1} = n$. A função que deve ser minimizada por forma a se encontrar a melhor divisão é então [38]:

$$J(\boldsymbol{\tau}, \boldsymbol{y}) = \frac{1}{n} \sum_{k=1}^{m+1} E_k \left(\boldsymbol{y}_{\boldsymbol{\tau}_{k-1}+\mathbf{1}:\boldsymbol{\tau}_k} \right) \tag{5.13}$$

E_k representa a função de erro do segmento k.

Se a estatística de interesse for a média, então a função de erro E_k do segmentos k pode ser definida, por exemplo, como:

[2] Adota-se a terminologia *função de erro* pois esta função procura quantificar o afastamento de um dado ponto em relação à estatística do segmento onde está inserido. Na literatura é comum o uso do termo *função de custo* porque o objetivo dos problemas onde estas funções são utilizadas é minimizar o custo de uma determinada ação, neste caso efetuar uma dada divisão em detrimento de outra [38; 39].

[3] $y_{1:n}$ significa que se consideram todos os pontos de $\boldsymbol{y}$ entre os elementos 1 e n.

$$E_k\left(\boldsymbol{y}_{\boldsymbol{\tau_{k-1}+1:\tau_k}}\right) = \sum_{i=\tau_{k-1}+1}^{\tau_k} \left(y_i - \bar{y}_k\right)^2 \tag{5.14}$$

onde $\bar{y}_k$ é dado por:

$$\bar{y}_k = \frac{1}{\tau_k - \tau_{k-1}} \sum_{i=\tau_{k-1}+1}^{\tau_k} y_i \tag{5.15}$$

Note-se que (5.14) pode ser vista como a soma dos quadrados dos resíduos para a aproximação de um conjunto de pontos por uma reta de ordenada igual à média desse conjunto.

Desta forma, o problema da deteção dos pontos de mudança para um número de pontos de mudança conhecido consiste em determinar o vetor $\boldsymbol{\tau_{1:m}}$ que minimiza (5.13).

5.3.2 Com número de pontos de mudança desconhecido

O desconhecimento do número de pontos de mudança (que, diga-se, é o caso mais comum na prática) dificulta a tarefa de determinação dos pontos de mudança porque a adição de pontos de mudança conduz sempre à diminuição do erro total e, portanto, resulta em *overfitting* [37]. Neste contexto, *overfitting* significa que o método considera que todos os pontos do vetor $\boldsymbol{y}$ são pontos de mudança. Desta forma é necessário penalizar o erro total sempre que se adiciona um novo ponto de mudança, ou seja, minimizar [38]:

$$H\left(\boldsymbol{\tau}, \boldsymbol{y}\right) = J\left(\boldsymbol{\tau}, \boldsymbol{y}\right) + \beta f\left(m\right) \tag{5.16}$$

$f\left(m\right)$ depende apenas da dimensão do vetor $\boldsymbol{\tau}$ (na sua versão mais simples: $f\left(m\right) = m$) e permite contrariar a tendência de *overfitting*. O coeficiente β ajusta o *trade-off* entre a minimização de $J\left(\boldsymbol{\tau}, \boldsymbol{y}\right)$ (obtida com uma maior dimensão de $\boldsymbol{\tau}$) e a minimização de $f\left(m\right)$ (obtida com a menor dimensão de $\boldsymbol{\tau}$) [38].

A função Matlab que é usada no presente projeto, *findchangepts*, considera que β é um coeficiente fixo definido pelo utilizador e que $f\left(m\right) = m$ [37]. O coeficiente β mais adequado é dependente do objetivo pretendido com a aplicação da função, ainda assim, valores em torno da variância do sinal são vistos como um bom ponto de partida para a obtenção de uma divisão adequada [40]. Mais considerações sobre a seleção de β serão efetuadas na Secção 5.4.

5.4 Variável *motor*

Na Secção 5.1 concluiu-se que a falta de dados era mais gravosa no caso da corrente. Assim, e mais do que perceber os valores de corrente em falta, torna-se necessário determinar o estado de funcionamento do motor nos instantes em que não se conhecem os valores de corrente.

Uma análise simultânea das variáveis *TI* e *I* permite verificar que uma mudança do estado de funcionamento do motor está associada, sem se verificar inércia térmica relevante, a um aumento substancial da temperatura. Visualiza-se também que, de forma simplificada, se pode considerar que a temperatura é baixa se o motor estiver desligado e elevada se o motor estiver ligado, ou seja, verifica-se que o uso de uma metodologia ade-

quada permite dividir os dados da temperatura em segmentos, cada um associado a um patamar de temperatura baixa (e motor desligado) ou temperatura alta (e motor ligado). Mais, se a divisão for bem efetuada, então dois segmentos consecutivos estão associados a estados de funcionamento do motor diferentes.

Na presente secção recorre-se aos conceitos apresentados na Secção 5.3 para efetuar a divisão idealizada. Como se verá, as divisões insatisfatórias obtidas obrigarão ao desenvolvimento de uma metodologia distinta que, ao invés de se basear na adição de pontos de mudança, se baseia na subtração. Uma métrica de avaliação das divisões efetuadas com a metodologia apresentada na Secção 5.3 será também desenvolvida. No final da secção todos os passos necessários à obtenção da variável *motor* terão sido apresentados. Volta a salientar-se a importância desta variável, que é fundamental na obtenção de informação a partir dos dados obtidos. As várias variáveis que podem ser obtidas a partir da variável *motor* são apresentadas na última subsecção.

5.4.1 Estatística de divisão

Na Tabela 5.2 apresenta-se as estatísticas que podem ser usadas na função Matlab *findchangepts* e os respetivos tipos de mudança que esta deteta em função dessa estatística [37].

Tabela 5.2: *Input* da função *findchangepts* e tipo de mudança detetado

Input	Tipo de mudança
mean	Alterações na média
rms	Alterações no *root-mean-square* (RMS)
std	Alterações no desvio-padrão
linear	Alterações na média e no declive

Com base nas opções e dado o objetivo de segmentar a temperatura em patamares bem definidos, o *RMS* apresenta-se como a melhor solução, visto que o impacto da introdução de um valor de um dado patamar de temperatura num conjunto de pontos do outro patamar é mais refletido nesta estatística do que na média (dado que é o quadrado do novo valor que é tido em conta na determinação do *RMS*). O *RMS* é dado por [41]:

$$RMS = \sqrt{\frac{1}{N}\sum_{n=1}^{N}|y_n|^2} \tag{5.17}$$

5.4.2 β ótimo

Selecionada a estatística mais ajustada ao objetivo estabelecido, é agora necessário selecionar um valor adequado para o coeficiente β , ou seja, que não conduza a *overfitting* (se demasiado baixo), nem proporcione uma divisão insatisfatória do sinal (se demasiado elevado). Tal é necessário porque não se sabe à partida qual é o número de pontos de mudança.

Como mencionado anteriormente, a variância do sinal é um valor inicial razoável para a seleção de β. Assim, uma metodologia adequada para a seleção deste parâmetro é preparar uma malha de pontos em torno da variância, proceder à divisão do sinal para

todos esses pontos e determinar a divisão ótima. A malha não deve ser muito refinada porque o tempo de computação da função para a estatística *RMS* é elevado.

Para que se possam comparar divisões, é necessário definir uma métrica que avalie cada divisão. Assim, defina-se o parâmetro p como sendo a razão entre a média do módulo das diferenças das médias de segmentos consecutivos e a raiz da média da variância dos segmentos. Atendendo a que a média $\bar{y}_k$ do segmentos k é dada por (5.15) e a variância σ_k^2 do segmento k é dada por [42]:

$$\sigma_k^2 = \frac{1}{\tau_k - \tau_{k-1} + 1} \sum_{i=\tau_{k-1}+1}^{\tau_k} (y_i - \bar{y}_k)^2 \tag{5.18}$$

pode escrever-se[4]:

$$p = \frac{\frac{1}{m} \sum_{k=1}^{m} |\bar{y}_{k+1} - \bar{y}_k|}{\left(\frac{1}{m+1} \sum_{k=1}^{m+1} \sigma_k^2 \right)^{\frac{1}{2}}} \tag{5.19}$$

O parâmetro p é adequado para avaliar as divisões, tendo presente o objetivo pretendido, porque: 1) o numerador aumenta com o aumento da diferença de médias de segmentos consecutivos. Logo, a exigência de diferenças elevadas entre dois segmentos consecutivos é satisfeita (o que significa que segmentos consecutivos estão associados a estados de funcionamento diferentes); 2) o denominador permite evitar que os segmentos obtidos contenham demasiados pontos dos dois patamares, penalizando a divisão se esta conduzir à obtenção de muitos segmentos de elevada variância. A raiz quadrada do denominador permite que o parâmetro seja adimensional.

Na Figura 5.3 apresenta-se o valor de p em função do coeficiente β normalizado pela variância σ^2 para os sinais de ambos os tipos de empanques mecânicos das duas bombas analisadas.

Pode observar-se que o valor máximo não ocorre quando β é igual à variância do sinal, mas quando este toma valores em torno dos 10% da variância. Note-se também que o parâmetro p diminui com o aumento de β. Isto acontece porque o aumento de β impede que sejam introduzidos pontos de mudança suficientes, sendo que quer o numerador (segmentos mais heterogéneos, logo menor diferença entre médias de segmentos consecutivos), quer o denominador (segmentos com pontos de ambos os patamares, logo com maior variância) do parâmetro p são muito penalizados. Em contrapartida, a diminuição excessiva de β também provoca a diminuição de p porque o sinal começa a ser dividido em demasia e surge a possibilidade de segmentos consecutivos poderem conter apenas pontos de um dado patamar e, portanto, terem médias próximas.

5.4.3 Pós-processamento da divisão

A aplicação da metodologia de divisão de um dado sinal apresentada na Secção 5.3 conduziu à obtenção de divisões que, embora razoáveis, não cumpriam alguns dos requisitos necessários para que possam ser consideradas satisfatórias, nomeadamente que dois

[4] Ver a Secção 5.3 para mais informação acerca do significado de cada variável.

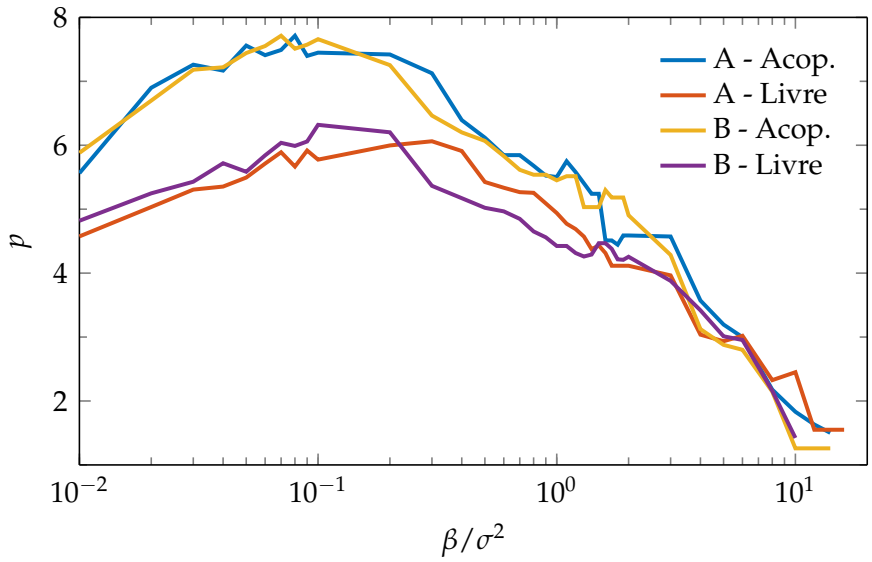


Figura 5.3: Valor do parâmetro p em função do coeficiente β normalizado pela variância σ^2 para os sinais de ambos os tipos de empanques mecânicos das bombas analisadas.

segmentos consecutivos se associassem a estados de funcionamento do motor distintos. Desta forma, optou-se por pós-processar as melhores divisões obtidas para cada caso.

O pós-processamento, neste contexto, consiste na eliminação dos pontos de mudança excessivos, isto é, que dividem segmentos consecutivos associados ao mesmo estado de funcionamento.

Para isso, começa por determinar-se a média de cada segmento sem *outliers* (como *outlier* entende-se um ponto mais afastado do que um dado desvio-padrão da média). A desconsideração dos *outliers* procura diminuir a influência de erros de medição nas divisões obtidas. De seguida, faz-se a subtração das médias calculadas para dois segmentos consecutivos e compara-se com um dado valor limite. Se a diferença das médias for inferior a esse limite, o ponto de mudança é eliminado. Note-se que o processo deve ser repetido várias vezes até que nenhuma eliminação seja efetuada (porque a eliminação de pontos conduz a novos segmentos que devem ser comparados com os segmentos vizinhos).

A determinação do valor ótimo da diferença que permite eliminar pontos de mudança efetua-se de forma iterativa: procura-se o valor mais baixo para o qual se verifica que, ao longo de todo o domínio, a diferença entre segmentos consecutivos altera sucessivamente de sinal. Por exemplo, se nos instantes iniciais a bomba está parada, a temperatura é baixa. Assim que é ligada, a temperatura aumenta. Se a divisão capta essa mudança, então a diferença da média das temperaturas desses dois segmentos é positiva (subtraindo o mais antigo ao mais recente). Assim que a bomba é desligada a temperatura volta a descer. A diferença entre a média das temperaturas dos dois segmentos mais recentes é agora negativa. O próximo segmento, espera-se, tem temperatura média superior e, portanto, a diferença volta a ser positiva. Desta forma, e admitindo que há uma valor ótimo que permite que a divisão seja efetuada da maneira idealizada, obtém-se a melhor divisão possível com o método apresentado.

Por forma a diminuir o tempo de computação necessário para encontrar o valor ótimo da diferença das médias, procedeu-se a uma implementação semelhante à utilizada no

método das bissecções [43]: 1) o utilizador define os valores mínimo e máximo aceitáveis; 2) a função começa por verificar se os valores mínimo e máximo permitem a divisão do sinal da forma pretendida; 3) se o máximo permitir, mas o mínimo não, então é garantido que existe um valor ótimo da diferença, que deve ser determinado; se o mínimo permitir, então esse é o valor ótimo do intervalo introduzido (o utilizador deverá, porventura, introduzir um valor mínimo mais baixo); 4) caso apenas o máximo permita, a função verifica se o valor médio do intervalo definido pelos dois valores introduzidos divide o sinal de forma ótima; 5) se dividir, a função avalia o valor médio do intervalo definido pelo valor mínimo e o valor avaliado; caso contrário, avalia o valor médio do intervalo definido pelo valor avaliado e o valor máximo; 6) o valor pretendido é encontrado assim que o extremo inferior do intervalo permitir a divisão do sinal de forma ótima.

5.4.4 Método da subtração

Ao longo das subsecções anteriores apresentou-se os passos para obter a melhor divisão possível do sinal usando o método apresentado na Secção 5.3, de agora em diante designado por *método da adição*, através da escolha da estatística adequada, da determinação do parâmetro β ótimo e do pós-processamento da divisão obtida. Ainda assim, e embora as divisões obtidas sejam as melhores que o *método da adição* permite, verificou-se que estas continuam a ser insatisfatórias, dado que algumas mudanças de estado não são detetadas.

Assim, surgiu a necessidade de desenvolver um novo método de divisão do sinal em patamares bem definidos. Tendo presente todos os passos do *método da adição*, a nova abordagem consistiu em, ao invés de usar a metodologia apresentada na Secção 5.3 para encontrar os pontos de mudança, considerar que todos os pontos são pontos de mudança e pós-processar a divisão obtida. Note-se que isto é semelhante a considerar que $\beta = 0$ e, portanto, que ocorre *overfitting*. Com este método, de agora em diante designado por *método da subtração*, espera ultrapassar-se a incapacidade do *método da adição* em detetar todas as mudanças de estado de funcionamento.

Enquanto que no pós-processamento efetuado no *método da adição* se procurou apenas determinar a diferença ótima entre as médias de segmentos consecutivos, no *método da subtração* procurou também determinar-se o desvio-padrão ótimo que separa um ponto normal de um *outlier*. Salienta-se, no entanto, que esta etapa tem pouca influência nas divisões obtidas, tendo sobretudo relevância nos casos em que a diferença das médias de segmentos consecutivos é próxima da diferença ótima. O impacto da seleção de um desvio-padrão ótimo no *método da adição* é ainda menor porque o pós-processamento é efetuado em segmentos com, tipicamente, um elevado número de pontos, ou seja, onde a influência dos *outliers* na média é muito reduzida.

O desvio-padrão ótimo é determinado através da comparação de uma variável *motor* definida com base nos segmentos determinados e uma variável *motor* obtida a partir dos dados de corrente conhecidos. A combinação diferença ótima-desvio-padrão ótimo que conduz a maior semelhança entre as variáveis é considerada a divisão ótima.

Verificou-se que o *método da subtração* é capaz de detetar mudanças de estado em situações em que o *método da adição* não o conseguia.

5.4.5 Comparação entre os métodos da *adição* e da *subtração*

Tendo sido apresentados os dois métodos de segmentação dos dados de temperatura e demonstrado que o *método da subtração* deteta pontos de mudança que escapam ao *método da adição*, importa agora demonstrar que a situação apresentada como exemplo não foi excecional. Na Tabela 5.3 apresenta-se o número de pontos de mudança obtidas usando cada um dos métodos por bomba e tipo de empanque mecânico.

Tabela 5.3: Número de pontos de mudança obtidos, por bomba e tipo de empanque mecânico, usando cada um dos métodos e comparação percentual dos valores

		Método		Comp. /%
		Adição	Subtração	
A	Acop.	128	178	39.1
	Livre	108	214	98.1
B	Acop.	133	231	73.7
	Livre	131	211	61.1

A análise da Tabela 5.3 permite concluir que o *método da subtração* conduz sempre à obtenção de um maior número de pontos de mudança (num dos casos levou à obtenção do dobro dos pontos de mudança). Demonstra-se assim que o *método da subtração* é, face aos objetivos pretendidos, o mais adequado.

5.4.6 Determinação da variável *motor*

Tendo sido apresentada a metodologia usada na divisão dos dados de temperatura em segmentos associados a estados de funcionamento do motor, importa agora tecer considerações acerca de como essas divisões podem ser utilizadas para obter a variável *motor* que, relembra-se, indica o estado de funcionamento do motor num dado instante.

Inicialmente, deve ser ponderado se os dados conhecidos de corrente devem ser utilizados. Se sim, a variável *motor* pode ser determinada para os instantes em que a corrente é conhecida e as variáveis *motor* determinadas com base nos dados de temperatura podem ser usadas para completar os instantes em falta. Se não, a variável *motor* deve ser obtida exclusivamente com base nos dados de temperatura. Embora a segunda opção seja viável, optou por usar-se os dados da temperatura apenas para completar os instantes em falta. Ainda assim, salienta-se que o sinal de corrente é redundante e podia não ser medido e armazenado caso tal se revelasse vantajoso do ponto de vista económico.

De seguida, e tendo em conta que para cada bomba são obtidas duas variáveis *motor* usando os sinais de temperatura (uma para cada tipo de empanque mecânico), importa ponderar de que modo estas podem ser selecionadas e/ou combinadas. A Tabela 5.4 demonstra que as variáveis obtidas são muito próximas (iguais em pelo menos 99% dos instantes temporais), pelo que se optou simplesmente por usar a variável que é mais semelhante, nos instantes conhecidos, à variável *motor* determinada com base no sinal de corrente do motor. Os resultados apresentados na Tabela 5.4 demonstram que o erro cometido ao completar os pontos em falta é muito reduzido, uma vez que nos instantes conhecidos há uma elevada semelhança entre as variáveis obtidas com os sinais de corrente e as variáveis usadas para completar os pontos em falta (>98%).

Tabela 5.4: Comparação entre variáveis *motor*: na primeira coluna comparam-se, entre si, as variáveis obtidas com o sinal de temperatura de cada tipo de empanque mecânico e nas restantes colunas estas são comparadas, em instantes conhecidos, com as variáveis *motor* obtidas a partir do sinal de corrente

	Comp. tipos	Comp. sinal corrente /%	
	empanques /%	Acop.	Livre
A	99.48	97.91	98.31
B	99.12	98.40	98.10

5.4.7 Informação extraída da variável *motor*

As subsecções anteriores demonstram que muita atenção foi dedicada à obtenção das variáveis *motor* (uma para cada bomba) robustas, tendo-se até experimentado e avaliado diferentes métodos de segmentação dos dados de temperatura. Tal deve-se à importância que a variável *motor* toma ao longo do projeto. De facto, esta variável é usada como ponto de partida de determinação de muitas outras (maioritariamente tempos) que, além de serem fundamentais nas conclusões obtidas nos Capítulos 3 e 6, são os *inputs* dos modelos de *Machine Learning*.

Número de arranques do motor

Conhecendo-se o estado de funcionamento da bomba em todos os instantes é possível, contando transições, obter o número de arranques do motor. Mais, é possível determinar a duração dos períodos de funcionamento associados a essas transições. O mesmo procedimento pode ser adotado para determinar o número de paragens e respetiva duração.

A informação acerca dos arranques e paragens é relevante porque estes são momentos críticos nos equipamentos dinâmicos. São-no ainda mais nos equipamentos dinâmicos que têm empanques mecânicos, dado que durante o arranque/paragem há alteração severa das condições de contacto dos conjuntos anel rotativo-anel estático, aumentando-se, por isso, a probabilidade de avaria.

Tempo desde o último arranque/paragem

Conhecendo-se as datas dos arranques/paragens pode determinar-se, num dado instante, o tempo desde o último arranque/paragem. Estas variáveis acrescentam informação ao número de arranques e podem revelar-se importantes em modelos que, como os modelos de *Machine Learning* utilizados, se baseiam na classificação do estado de funcionamento.

Tempo de funcionamento da bomba e tempo de calendário

Da mesma forma que é possível calcular o tempo desde um determinado acontecimento (por exemplo, arranque), pode determinar-se o tempo de funcionamento da bomba. Esta variável, além de permitir potenciar as técnicas tradicionais de análise de falhas (Secção 3.4) e permitir comparações entre equipamentos (Secção 6.4), pode ser útil para distinguir falhas associadas a processos de envelhecimento.

Aliado ao tempo de funcionamento, é também importante conhecer o tempo de calendário, definindo-se este como o tempo desde a instalação das bombas.

Tempo de funcionamento desde a mudança de empanque mecânico e tempo de vida

Conhecendo-se as datas de montagem dos empanques mecânicos é possível, combinando os procedimentos de cálculo de tempo desde um determinado acontecimento e de tempo de funcionamento, determinar o tempo de funcionamento da bomba desde a montagem de um dado empanque mecânico. Esta variável pode ser relevante se o envelhecimento (entenda-se, acumulação de desgaste) for a causa de falha de um dado empanque mecânico.

De forma similar, pode determinar-se o tempo de vida do empanque mecânico. Esta variável, além de ser relevante para os modelos de *Machine Learning*, é fundamental para avaliar se o tempo de vida dos empanques mecânicos está de acordo com o estabelecido pelas normas e fabricantes (Secção 6.3).

Tempo desde a última pressurização

Combinando os conceitos que serão apresentados na secção seguinte com os que foram apresentados na corrente secção pode determinar-se o tempo desde a última pressurização. A expectativa é que o tempo entre pressurizações diminua com a proximidade da falha, dado que é esperado que as fugas sejam maiores.

Tempo desde a última substituição do empanque mecânico do lado oposto

Um último tempo que se considera relevante determinar é o tempo desde a última substituição do empanque mecânico do lado oposto. Esta variável procura verificar se os trabalhos de substituição de um empanque mecânico influenciam a vida do empanque mecânico do lado oposto. Tal pode resultar de perturbações introduzidas na bomba centrífuga durante a realização da intervenção.

5.5 Pressurizações

5.5.1 Número de pressurizações

Do ponto de vista da análise de sinal, uma pressurização é um aumento abrupto da pressão. Esse aumento, geralmente de vários bar, ocorre tipicamente num período inferior a 10 minutos. Ainda assim, e tendo em conta que os dados da RTDB usados estão afastados temporalmente 10 minutos, há a possibilidade de uma pressurização aparentar ter uma duração superior (em casos excecionais a duração real pode aproximar-se dos 20 minutos).

Embora a definição acima apresentada já permita reduzir enormemente o número de instantes de pressurização possíveis, até porque aumentos abruptos de pressão por outras razões não são esperados, a definição de pressurização deve ser mais restrita. De facto, além dos aumentos de pressão devidos às pressurizações, ocorrem aumentos de pressão por outras duas razões: 1) a baixa pressão, por possível fuga de fluido bombeado para o circuito de selagem; 2) devido ao aumento da temperatura (verifica-se uma oscilação sazonal da pressão, justificada pela variação da temperatura ambiente ao longo do dia). A dependência da pressão com a temperatura, embora pequena, é justificada com o aumento do volume do gás do acumulador hidráulico (Secção 4.1). Desta forma, definiu-se um aumento de pressão mínimo de 1 bar para que um dado aumento de pressão possa ser considerado pressurização (as outras razões para o aumento da pressão não conduzem, tipicamente, a aumentos tão elevados).

Por fim, impõe-se ainda um valor mínimo de pressão final após aumento de pressão, para que esse aumento possa ser atribuído a uma pressurização. Após observação do sinal, considerou-se que a pressão deve aumentar para, pelo menos, 5 bar. Acrescenta-se que este valor foi definido tendo por base que se observam, em torno dos 4.5 bar, quedas abruptas de pressão. A importância destas quedas de pressão vai ser abordada na Secção 6.5, sendo também necessário determinar os instantes em que ocorrem.

Desta forma, é possível determinar os instantes das pressurizações (e, portanto, o número de pressurizações), o tempo entre pressurizações, o valor de pressão final e o aumento de pressão.

5.5.2 Declives do sinal de pressão

Dado que a definição de falha do empanque mecânico se baseia no declive do sinal de pressão (Secção 4.2), é também necessário definir um procedimento para a sua determinação.

Na Secção 5.3 introduziu-se uma metodologia de deteção de pontos de mudança. Por sua vez, na Subsecção 5.4.1 mencionou-se que a função Matlab que permite aplicar essa metodologia aceita como argumento de entrada a estatística *linear*, que indica à função para identificar pontos onde se verifiquem alterações na média e no declive. Conclui-se então que não é necessário desenvolver uma metodologia de cálculo de declives, mas antes usar uma metodologia que já se revelou útil para o cumprimento de outros objetivos.

Importa, por fim, mencionar que a função *findchangepts* não foi aplicada ao sinal de pressão completo, mas antes a segmentos de pressão limitados pelos instantes de pressurização. Desta forma, contorna-se a dificuldade provocada pelos aumentos abruptos de pressão e aumenta-se a qualidade da divisão, já que o parâmetro β utilizado em cada divisão é a variância de cada segmento (ao invés da variância de todo o sinal, que localmente tem menos significado).

5.6 Conclusões e resumo dos sinais medidos e variáveis obtidas

No presente capítulo foram apresentadas e aplicadas as metodologias que permitem reconstruir os dados em falta. Diferentes abordagens foram efetuadas para executar essa tarefa, dado que a quantidade de pontos em falta e, sobretudo, o tempo consecutivo de falta de dados são diferentes para os vários sensores. Uma metodologia de deteção de pontos de mudança através da remoção de pontos de mudança excessivos foi desenvolvida.

Ao longo do capítulo foram ainda apresentadas as diversas variáveis obtidas a partir dos sinais medidos e/ou da sua combinação com os dados do SAP (Tabela 5.5). Todas as variáveis apresentadas na Tabela 5.5 foram introduzidas no capítulo atual ou em anteriores.

Tabela 5.5: Resumo dos sinais medidos e das variáveis obtidas a partir dos sinais medidos e/ou da sua combinação com os dados do SAP

Variável	Descrição	Origem
I	Corrente do motor	Medido
TI	Temperatura do sistema de selagem de um dado tipo de empanque	Medido

PI	Pressão do sistema de selagem de um dado tipo de empanque	Medido
motor	Estado de funcionamento do motor	*I*, *TI*
narranques	Número de arranques	*motor*
tempoultarranque	Tempo desde o último arranque do motor	*motor*
tempoultparagem	Tempo desde a última paragem do motor	*motor*
tempofunc	Tempo de funcionamento do motor	*motor*
tempocal	Tempo de calendário	Data dos sinais medidos
tempofuncmudemp	Tempo de funcionamento desde a mudança de empanque	*motor*, SAP
tempovidaemp	Tempo de vida do empanque	SAP
npress	Número de pressurizações	*PI*
tempoultpress	Tempo desde a última pressurização	*motor*, *PI*
pressaoapospress	Pressão após pressurização	*PI*
aumentopressao	Aumento de pressão ocorrido numa dada pressurização	*PI*
nquedaspressao	Número de quedas de pressão abaixo de um determinado valor mínimo (4 bar)	*PI*
declives	Declive do sinal de pressão	*PI*
tempoultsuboposto	Tempo desde a última substituição do empaque do lado oposto	*motor*, SAP

6. Análise dos dados da base de dados RTDB

6.1 Introdução

No Capítulo 3 foi efetuado um estudo “tradicional" dos registos SAP, no Capítulo 4 foi apresentado o princípio de funcionamento dos empanques mecânicos e a definição quantitativa de falha e no Capítulo 5 foram tratados os sinais medidos em contínuo e desenvolvidas ferramentas que permitem obter informação a partir da utilização simultânea de registos SAP e desses sinais. O presente capítulo procura integrar todas as tarefas realizadas e demonstrar que as ferramentas Matlab desenvolvidas são uma mais valia para a tomada de decisão, visto permitirem obter informação útil que, até ao momento, não está a ser considerada.

Numa primeira fase, apresentam-se as datas de paragem da unidade e as datas de substituição dos empanques mecânicos. Posteriormente, utiliza-se a definição quantitativa de falha e determinam-se os instantes nos quais esta ocorre em cada um dos casos. Essa informação é então comparada com as datas definidas no SAP e considerações acerca dos tempos médios de reparação e dos tempos médios de vida são efetuadas. Ainda, é apresentada e analisada informação relativa ao tempo de funcionamento e arranques do motor e às pressurizações (número, pressurizações por dia da semana e quedas de pressão abaixo dos 4 bar).

6.2 Datas de paragem da unidade

O complexo industrial onde foi realizado o projeto é sujeito a paragens quadrienais para manutenção. Além disso, a unidade onde se localizam as bombas em análise pode parar, por exemplo, por avaria prolongada de um dado equipamento, em períodos que não os de paragem global. Assim, importa detetar os períodos de paragem e analisar o comportamento dos empanques mecânicos, por forma a concluir se os dados recolhidos nesses períodos podem ser usados nos modelos de *Machine Learning* ou se devem ser descaracterizados.

A identificação de paragens da unidade pode ser efetuada diretamente a partir dos dados da RTDB (ou, mais facilmente, a partir das variáveis *motor*). Será considerada paragem da unidade sempre que a unidade parar por um período igual ou superior a três dias. Dado que a unidade não opera com ambas as bombas desligadas, podem determinar-se as paragens da unidade através da identificação de períodos em que a variável *motor* é nula, para ambas as bombas, pelo menos três dias consecutivos. Na Tabela 6.1 apresentam-se as paragens identificadas, bem como as respetivas durações. Salienta-se que a RTDB per-

mite obter, praticamente ao minuto, as datas de início e fim de paragem. A Tabela 6.1 demonstra ainda a continuidade de funcionamento da unidade: entre 2012 e 2017 a unidade parou apenas seis vezes por períodos superiores a dois dias.

Tabela 6.1: Datas estimadas da paragem da unidade (por visualização gráfica)

	Início		Fim		Duração /d
	Data	Hora aprox.	Data	Hora aprox.	
1	2012-11-01	00:00	2012-11-13	20:30	12
2	2013-05-10	12:00	2013-05-22	18:00	12
3	2014-01-03	12:00	2014-01-19	16:00	16
4	2015-01-09	09:00	2015-01-26	06:00	17
5	2016-04-12	18:30	2016-05-18	16:00	36
6	2017-03-24	12:00	2017-04-01	16:00	8

A observação gráfica dos sinais medidos nos períodos de paragem da unidade permite concluir que os dados recolhidos nesses períodos podem ser usados nos modelos de *Machine Learning*. De facto, observa-se que na maioria das paragens não é efetuada qualquer intervenção nos empanques mecânicos. Quando são efetuadas intervenções, estas são apenas de limpeza do circuito de selagem ou de pressurização. Não ocorreram substituições de empanques mecânicos nesses períodos.

6.3 Datas de substituição dos empanques mecânicos

6.3.1 Registos SAP

Mais importantes do que as datas de paragem da unidade (que, como se viu na secção anterior, não são relevantes para a aplicação de modelos de *Machine Learning*) são as datas de intervenção dos empanques mecânicos. Estas datas, com apenas uma exceção, são datas de substituição destes componentes.

A correta definição das datas de substituição dos empanques mecânicos é fundamental, uma vez que a partir do conhecimento, para cada empanque mecânico, das datas em que este foi colocado e retirado de serviço e, atendendo ao que foi apresentado no Capítulo 4, da data em que falhou, é possível dividir os dados da RTDB em períodos que correspondem a modos de funcionamento normal, "pré-instável" (período que antecede a falha) e instável (período que sucede a falha e termina quando se efetua uma nova mudança de empanque mecânico), ou seja, classificar os períodos de funcionamento. Salienta-se ainda que os dados dos instantes de tempo em que o funcionamento é instável não são usados nos modelos de *Machine Learning*.

As datas de início e fim de avaria dos empanques mecânicos, a respetiva duração de parada, o número de empanques mecânicos utilizados e o empanque mecânico responsável pela abertura da nota (*bad actor*) são apresentados nas Tabelas 6.2 e 6.3 para as bombas A e B, respetivamente. Salienta-se que apenas num caso (A3) se verificou manutenção oportunista, ou seja, substituição de um dado empanque mecânico aquando da paragem para a substituição do empanque mecânico do lado oposto, sem que o primeiro estivesse em funcionamento instável. Esta conclusão está intrinsecamente ligada à definição de

falha (se tivesse sido escolhido outro declive crítico a conclusão podia ser diferente).

Tabela 6.2: Datas de intervenção nos empanques mecânicos da bomba A

Ref.	Datas		Duração	Nº emp.		*Bad Actor*[1]
	Início	Fim	/d	Acop.	Livre	
A1	2012-04-14	2012-04-17	3	1	0	1
A2	2012-11-18	2012-12-20	32	1	0	1
A3	2012-12-26	2013-11-02	311	1	1	2
A4	2013-10-28	2013-11-19	22	1	0	1
A5	2014-03-13	2014-03-14	1	1	0	1
A6	2014-06-30	2014-07-08	8	1	0	1
A7	2014-07-18	2014-08-22	35	1	0	1
A8	2014-09-01	2014-11-25	85	1	0	1
A9	2015-09-24	2015-10-05	11	1	1	2
A10	2016-05-19	2016-12-12	207	1	0	1

[1] 1-Acoplado; 2-Livre; 3-Ambos

A identificação do *bad actor* nas intervenções em que foram substituídos empanques mecânicos de ambos os lados pode ser efetuada apenas com auxílio dos dados da RTDB (daí que este campo não esteja definido em intervenções efetuadas em 2011 ou no início de 2012). A partir desses dados é possível observar o tipo de empanque mecânico que atingiu o estado de falha em primeiro lugar. Ainda assim, verificou-se que, com exceção da intervenção oportunista, os empanques mecânicos do lado oposto ao *bad actor* atingiram o estado de falha antes da intervenção para substituição do *bad actor*.

As Tabelas 6.2 e 6.3 ao detalharem os tempos de parada para cada intervenção demonstram que, tal como se afirmou no Capítulo 3, as datas de início de avaria registadas no SAP referem-se a datas de deteção de avaria que, mais do que identificar uma falha, procuram informar a equipa de manutenção da eventual necessidade de substituição de um dado empanque mecânico num futuro próximo (só assim se justificam as durações de parada mais elevadas). Ou seja, o equipamento pode continuar a funcionar após o que se considera ser o início da avaria, pelo que a duração da parada contempla também um período em que o empanque mecânico funcionou de forma instável, não contabilizando, portanto, apenas o período em que o equipamento esteve indisponível.

Deve também ser referido que, não tendo sido encontrada justificação para a utilização de quatro empanques mecânicos na intervenção B19 (podem, porventura, ter sido danificados durante a montagem), se assumiu que foram montados apenas dois.

A observação dos sinais de pressão do circuito de selagem em torno das datas consideradas no SAP como datas de início de avaria não permite identificar um padrão de abertura de nota. No que concerne às datas de fim de avaria, verificou-se que são próximas das reais, mas que, obviamente, não estão definidas ao minuto. Dado que tal é necessário para utilização destas datas em conjunto com os dados da RTDB, desenvolveu-se um procedimento semi-automático de identificação destas datas. O procedimento e os resultados obtidos são alvo de análise na subsecção seguinte.

Tabela 6.3: Datas de intervenção nos empanques mecânicos da bomba B

Ref.	Datas		Duração	Nº emp.		*Bad Actor*[1]
	Início	Fim	/d	Acop.	Livre	
B1	2011-10-17	2011-11-11	25	0	1	2
B2	2011-12-21	2011-12-23	2	1	1	5
B3	2012-06-12	2012-06-20	8	1	0	1
B4	2012-07-04	2012-11-14	133	1	1	5
B5	2012-11-28	2013-02-21	85	1	1	2
B6	2013-08-14	2013-09-03	20	1	1	2
B7	2014-01-28	2014-02-12	15	0	1	2
B8	2014-02-13	2014-02-18	5	1	0	1
B9	2014-03-04	2014-03-14	10	0	1	2
B10	2014-06-11	2014-06-13	2	0	0	4
B11	2014-07-17	2014-07-31	14	1	0	1
B12	2014-08-07	2014-10-02	56	1	0	1
B13	2014-10-16	2014-10-22	6	0	1	2
B14	2014-11-06	2014-12-22	46	0	1	2
B15	2015-03-02	2015-03-13	11	1	1	1
B16	2015-03-28	2015-12-18	265	1	1	2
B17	2015-12-29	2016-01-21	23	0	1	2
B18	2016-12-17	2017-01-09	23	0	1	2
B19	2017-09-28	2017-10-18	20	2	2	2
B20	2017-11-17	2017-12-04	17	1	1	1

[1] 1-Acoplado; 2-Livre; 3-Ambos; 4-Sem substituição; 5-Não definido

6.3.2 Datas reais de substituição dos empanques mecânicos e de início de avaria

A determinação das datas de substituição dos empanques mecânicos é feita, tal como mencionado na subsecção anterior, de forma semi-automática. Inicialmente, a função Matlab desenvolvida procura todos os instantes em que ocorre uma pressurização que parte de um valor de pressão inferior a um máximo introduzido (não se considera a partir de zero porque o sinal *PI* apresenta, nesses instantes, um valor não nulo, ainda que muito baixo) e atinge um valor de pressão superior a um mínimo introduzido. Além disso, a função deteta apenas as pressurizações que, além de cumprirem os requisitos anteriores, ocorrem em instantes em que a bomba correspondente está desligada. No entanto, observou-se que a função deteta datas que não correspondem a substituições e não capta algumas das que correspondem. Assim, o passo manual necessário consiste em identificar essas datas, eliminando as excessivas e introduzindo as que estão em falta. Este é o único procedimento de pré-processamento de dados que requer, obrigatoriamente, intervenção humana.

As datas de substituição obtidas com o procedimento acima apresentado (e que a partir de agora passam a designar-se por datas de montagem) são apresentadas, para cada bomba e por empanque mecânico, nas Tabelas 6.4 a 6.7. Nessas tabelas apresenta-se também a data real de falha (obtida após a definição quantitativa do modo de falha apresentada na Secção 4.2), o tempo de vida de um dado empanque mecânico, a duração da parada (que, neste contexto, é o tempo desde a data real de falha até à substituição do empanque mecânico), a referência da intervenção a que um dado empanque mecânico está associado (Tabelas 6.2 e 6.3) e a diferença entre as datas de falha e início de avaria do registo SAP correspondente. A nomenclatura adotada para cada unidade contém duas letras e um número. A primeira letra corresponde à bomba e a segunda corresponde ao tipo de empanque mecânico. Na nomenclatura dos *empanques virtuais* acrescenta-se a letra *V*.

Atente-se agora nos casos em que se considerou necessário definir *empanques virtuais* (AA8, AA9, AA10, BA8, BL2 e BL5). Comece por notar-se que em todos eles (com exceção do AA9 e do AA10 - este último porque ainda não tem um registo SAP associado) a diferença entre a data de falha determinada e a data de início de avaria do registo SAP correspondente é muito elevada (superior a 130 dias). No caso do empanque AA9, verifica-se que a data real de falha coincide com a data de início do registo SAP. No entanto, a duração da parada no SAP é muito elevada, o que demonstra que a equipa de manutenção demorou muito a intervir porque, eventualmente, o empanque mecânico voltou a funcionar corretamente. Quanto ao empanque AA8, é interessante notar que embora a data real de falha seja muito inferior à do registo SAP, a data de falha do *empanque virtual* é muito próxima. O mesmo sucede com os empanques BL2 e BL5. Já a falha do empanque BA8V1 ocorre cerca de três meses antes do registo SAP identificar o início da avaria.

Da mesma forma que se atribuíram *empanques virtuais* para os empanques mecânicos mencionados no parágrafo anterior, ponderou-se fazê-lo para os empanques AA3 e BL11. Ainda assim, a observação dos sinais medidos permitiu concluir que estes nunca voltaram verdadeiramente a um estado de funcionamento estável (a análise dos sinais medidos deve ter em conta, tal como se mencionou anteriormente, o estado de funcionamento da bomba, visto que por vezes os empanques mecânicos em funcionamento instável aparentam apresentar um comportamento normal quando a bomba está desligada).

Tabela 6.4: Datas de montagem e falha (definida quantitativamente), tempo de vida, duração da parada, referência ao registo SAP associado e diferença entre a data de falha real e a de início de avaria desse registo para o empanque mecânico do lado acoplado da bomba A

Un.	Data		Tempo vida		Dur. parada		Ref. int.	Dif.
	Montagem	Falha	/d		/d			/d
AA1	2012-11-06	2012-11-15	9		32		A2	-3
AA2	2012-12-17	-	-		0		A3	-
AA3	2013-05-09	2013-06-26	48		142		A4	-124
AA4	2013-11-15	2013-12-01	16		152		A5	-102
AA5	2014-05-02	2014-05-14	12		54		A6	-47
AA6	2014-07-07	2014-07-17	10		35		A7	-1
AA7	2014-08-21	2014-08-31	10		80		A8	-1
AA8	2014-11-19	2014-12-06	17	151	151	165	A9	-292
AA8V1	2015-05-06	2015-09-17	134		14			
AA9	2015-10-01	2016-05-19	231	341	21	97	A10	0
AA9V1	2016-06-09	2016-09-27	110		76			
AA10	2016-12-12	2017-05-14	153		3			
AA10V1	2017-05-17	2017-09-27	133	362	2	5	-	-
AA10V2	2017-09-29	2017-12-14	76		-			

Tabela 6.5: Datas de montagem e falha (definida quantitativamente), tempo de vida, duração da parada, referência ao registo SAP associado e diferença entre a data de falha real e a de início de avaria desse registo para o empanque mecânico do lado livre da bomba A

Un.	Data		Tempo vida	Dur. parada	Ref. int.	Dif.
	Montagem	Falha	/d	/d		/d
AL1	2012-11-06	2012-12-23	47	137	A3	-3
AL2	2013-05-09	2015-06-16	768	107	A9	-100
AL3	2015-10-01	2017-12-14	805	-	-	-

Tabela 6.6: Datas de montagem e falha (definida quantitativamente), tempo de vida, duração da parada, referência ao registo SAP associado e diferença entre a data de falha real e a de início de avaria desse registo para o empanque mecânico do lado acoplado da bomba B

Un.	Data		Tempo vida		Dur. parada		Ref. int.	Dif.
	Montagem	Falha	/d		/d			/d
BA1	2012-11-14	2013-01-22	69		24		B5	55
BA2	2013-02-15	2013-07-24	159		36		B6	-21
BA3	2013-08-29	2014-01-03	127		45		B8	-41
BA4	2014-02-17	2014-03-10	21		141		B11	-129
BA5	2014-07-29	2014-08-04	6		57		B12	-3
BA6	2014-09-30	2014-12-26	87		76		B15	-66
BA7	2015-03-12	2015-07-30	140		139		B16	124
BA8	2015-12-16	2016-03-17	92	355	294	330	B19	-560
BA8V1	2017-01-05	2017-09-25	263		36			
BA9	2017-10-31	2017-11-08	8		29		B20	-9
B10	2017-12-07	2017-12-27	20		-		-	-

Tabela 6.7: Datas de montagem e falha (definida quantitativamente), tempo de vida, duração da parada, referência ao registo SAP associado e diferença entre a data de falha real e a de início de avaria desse registo para o empanque mecânico do lado livre da bomba B

Un.	Data		Tempo vida		Dur. parada		Ref. int.	Dif.
	Montagem	Falha	/d		/d			/d
BL1	2012-11-14	2012-11-19	5		88		B5	-9
BL2	2013-02-15	2013-03-31	44	158	15	37	B6	-136
BL2V1	2013-04-15	2013-08-07	114		22			
BL3	2013-08-29	2013-10-20	52		115		B7	-100
BL4	2014-02-12	2014-02-22	10		20		B9	-10
BL5	2014-03-14	2014-04-02	19	69	85	152	B13	-197
BL5V1	2014-06-26	2014-08-15	50		67			
BL6	2014-10-21	2014-11-01	11		47		B14	-5
BL7	2014-12-18	2014-12-30	12		72		B15	-62
BL8	2015-03-12	2015-03-17	5		274		B16	-11
BL9	2015-12-16	2015-12-23	7		20		B17	-6
BL10	2016-01-12	2016-09-09	241		117		B18	-99
BL11	2017-01-04	2017-01-10	6		287		B19	-261
BL12	2017-10-24	2017-11-02	9		28		B20	-15
BL13	2017-11-30	-	-		-			-

Os parágrafos anteriores justificam a necessidade do desenvolvimento do conceito *empanque virtual*[1]. De facto, a introdução deste conceito permite que se usem mais de 850 dias de dados que, de outra forma, corresponderiam a funcionamento instável, ou seja, permitem o uso de mais 120000 pontos (atendendo a que os pontos usados distam 10 minutos), o que corresponde a cerca de 20% do total de pontos usados nos modelos de *Machine Learning*. Mais, seriam desconsiderados alguns dos empanques mecânicos com maior tempo de vida. Faz-se notar, no entanto, que o conceito deve ser definido, em trabalhos posteriores, de forma mais clara ou, idealmente, deve procurar perceber-se o comportamento físico dos empanques mecânicos de tal forma que não seja necessário a introdução do conceito e, mesmo assim, sejam considerados todos os pontos de funcionamento normal.

As Tabelas 6.4 a 6.7 permitem ainda concluir que as datas reais de falha são, tipicamente, inferiores às datas de início de avaria registadas no SAP. No entanto, observam-se diversas situações (sobretudo associadas a tempos de vidas reduzidos) em que a diferença é inferior a 10 dias. A grande variância das diferenças observadas, mais do que indiciar que o declive crítico foi definido erradamente, demonstra que, tal como se afirmou anteriormente, as datas de início de avaria dos registos SAP não se baseiam em critérios objetivos de definição de falha. Ainda assim, volta a frisar-se que, para garantir um maior aproveitamento do tempo de vida útil dos empanques mecânicos, deve ser dada mais atenção à definição de falha utilizada, uma vez que a atual conduz ao subaproveitamento do tempo de vida dos empanques mecânicos.

No que concerne às datas de início de avaria importa ainda mencionar que os registos SAP não permitem, atualmente, identificar o tipo de empanque mecânico responsável pela abertura da nota. A análise posterior efetuada usando os dados da RTDB ou a definição quantitativa de falha permite ultrapassar o problema. No entanto, sugere-se que esta informação passe a ser de introdução obrigatória no SAP por forma a ser possível, em análises subsequentes, identificar o que levou um determinado operador a considerar que um dado empanque mecânico estava em falha (poderá ser útil para afinar a definição de falha).

6.3.3 Tempos médios de reparação

A determinação das datas reais de falha permite ainda fazer uma avaliação dos tempos médios de reparação calculados no Capítulo 3. Relembra-se que se tinha concluído que o tempo médio de reparação do modo de falha *fuga empanque* da bomba B era cerca de metade (937.8 horas) do da bomba A (1712.6 horas). Na Tabela 6.8 apresentam-se os novos valores calculados. Salienta-se que se consideram apenas os empanques mecânicos que falham e que, nos cálculos efetuados, um empanque mecânico e os seus *empanques virtuais* totalizam apenas um empanque mecânico.

Antes de mais, importa salientar que a definição quantitativa de falha permite que se possa dividir o *MTTR* por tipo de empanque mecânico (com base nas datas de início é apenas possível dividir por bomba). Esta divisão permite concluir que, em média, os empanques mecânicos do lado livre demoram mais tempo a ser reparados (pode relacionar-se com diferenças nos procedimentos de manutenção). Já as diferenças entre equipamentos são mínimas (o que contraria as conclusões retiradas na análise dos registos SAP). Além disso, verifica-se que os tempos determinados são superiores aos anteriormente obtidos. Uma vez mais, é a falta de objetividade da definição das datas de início de

[1] Ver Secção 4.2.

Tabela 6.8: Tempo médio de reparação dos empanques mecânicos por bomba e tipo de empanque mecânico obtidos após definição quantitativa do modo de falha

/h	Acop.	Livre	Conj.
A	2271.0	2928.0	2402.4
B	2338.7	2514.0	2438.9
			2427.1

avaria que justifica a discrepância dos resultados.

6.3.4 Tempos médios de vida

Por fim, pode ainda analisar-se os tempos médios de vida dos empanques mecânicos. Partindo dos mesmos pressupostos usados no cálculo dos tempos médios de reparação, obtém-se os tempos médios de vida apresentados na Tabela 6.9.

Tabela 6.9: Tempo médio de vida dos empanques mecânicos por bomba e tipo de empanque mecânico obtidos após definição quantitativa do modo de falha

/d	Acop.	Livre	Conj.
A	107	540	215
B	99	49	72
			122

A Tabela 6.9 permite concluir que os tempos médios de vida dos empanques mecânicos da bomba A são superiores ao da bomba B. Tal deve-se, sobretudo, aos tempos médios de vida elevados (em comparação com os restantes casos) dos empanques mecânicos do lado livre da bomba A. Em contrapartida, os empanques mecânicos do lado livre da bomba B são os que apresentam tempos médios de vida mais baixos. Salienta-se que a norma ANSI/API 682 [25] apresenta indicações de seleção de empanques mecânicos que procuram conduzir à escolha de sistemas de selagem com “elevada probabilidade de operar 3 anos em serviço contínuo." Assim, verifica-se que em média os empanques mecânicos têm uma vida que corresponde apenas a cerca de 11% da vida prevista na norma. Mais, atentando na Figura 6.1, verifica-se que nenhum dos empanques mecânicos utilizados atingiu o tempo de vida previsto, sendo que o de maior durabilidade completou apenas 73% desse tempo.

A Figura 6.1 permite ainda concluir que cerca de 50% dos empanques mecânicos utilizados não ultrapassaram os 50 dias de tempo de vida e que apenas 15% ultrapassou a barreira dos 150 dias. Importa também notar que não existe qualquer falha registada com um tempo de vida entre os 300 e os 750 dias.

As semelhanças entre a Figura 6.1 e a curva da banheira, Figura 6.2, são evidentes. A elevada taxa de avarias verificada para os primeiros 50 dias diminui rapidamente e por volta dos 150 dias pode assumir-se que este período, muitas vezes designado por período de mortalidade infantil [16; 27], deu lugar a um período de taxa de avarias aproximadamente constante. De facto, entre os 150 dias e os 750 dias a taxa de avarias tende a estabilizar (verifica-se uma pequena diminuição com o tempo), sendo neste período que

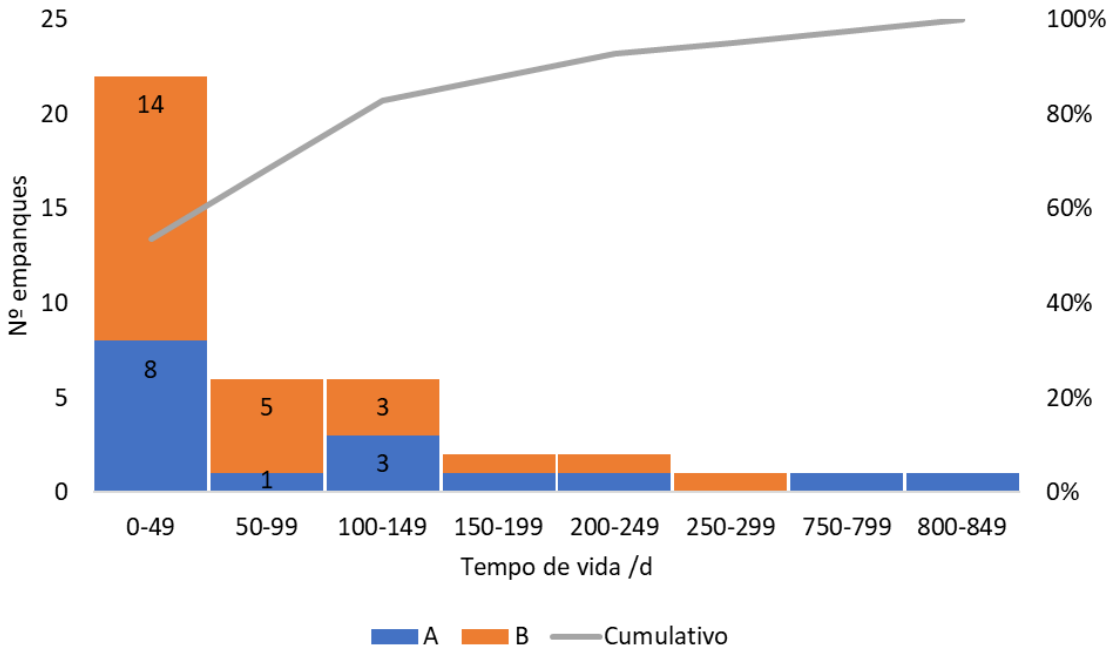


Figura 6.1: Tempo de vida dos empanques mecânicos.

é atingido o seu valor mínimo. Por fim, a partir dos 750 dias verifica-se que a taxa de avarias começa a aumentar, o que pode indiciar que este é o instante onde se inicia a zona final de vida do componente e onde as falhas estão intimamente ligadas ao tempo de operação [16].

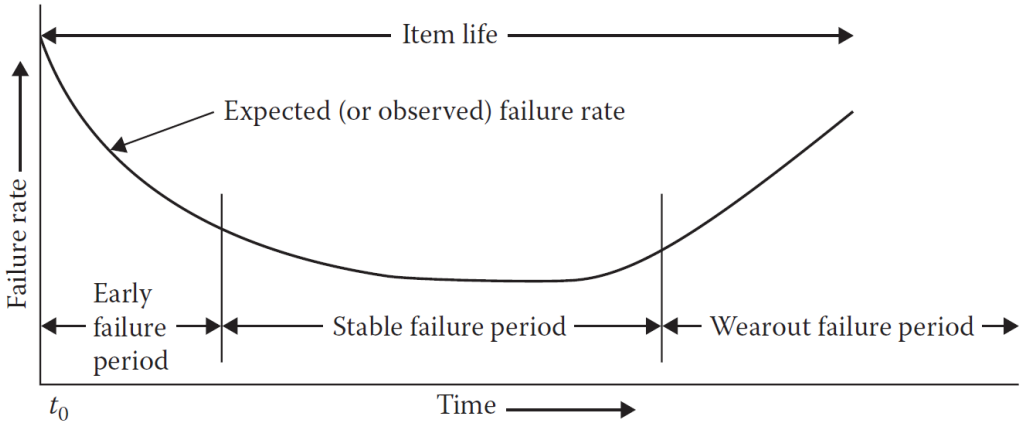


Figura 6.2: Curva da banheira para a taxa de avarias [16].

6.4 Tempo de funcionamento e arranques do motor

Além de informação relativa à falta de dados[2], às paragens da unidade[3] e às falhas e tempos de vida dos empanques mecânicos[4], os dados da RTDB, e nomeadamente as

[2]Ver Secção 5.1.
[3]Ver Secção 6.2.
[4]Ver Secção 6.3.

variáveis apresentadas no Capítulo 5, permitem extrair informação adicional acerca do funcionamento das bombas. Esta secção procura demonstrar o tipo de informação que pode ser obtida, nomeadamente em relação ao tempo de funcionamento das bombas e ao número de arranques e paragens do motor.

6.4.1 Tempos de funcionamento

Na Tabela 3.5 apresentou-se o tempo de funcionamento de cada bomba por ano. Na Figura 6.3 apresenta-se a evolução do tempo de funcionamento de ambas as bombas ao longo do tempo.

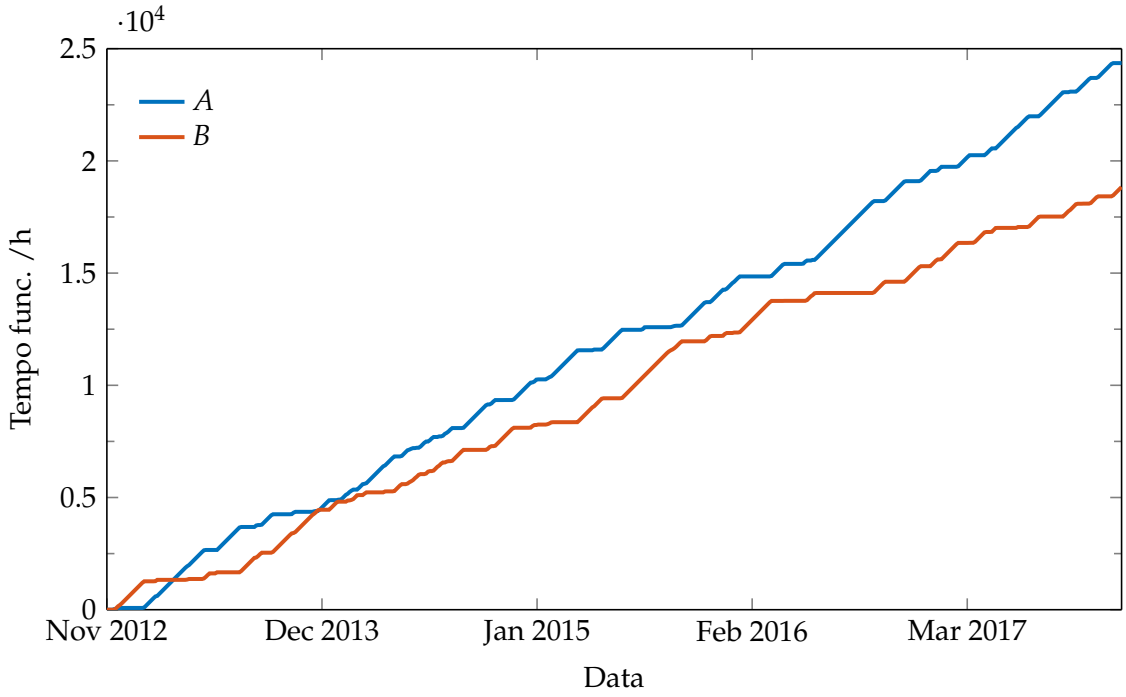


Figura 6.3: Evolução do tempo de funcionamento das bombas ao longo do tempo.

Num contexto industrial onde é esperado que os sistemas redundantes funcionem durante longos anos sem alterações tecnológicas significativas é boa prática procurar que o equipamento principal funcione o dobro do tempo do equipamento de redundância. Tal pode ser justificado, por exemplo, pela zona final da curva da banheira (Figura 6.2): perto do fim de vida, a probabilidade de avaria aumenta. Assim, com o funcionamento diferenciado evita-se que a probabilidade de avaria comece a aumentar em simultâneo em ambos os equipamentos. A análise da Figura 6.3 demonstra que esta política não tem sido seguida. De facto, embora se verifique uma tendência para a bomba A funcionar durante mais tempo, o tempo de funcionamento de ambas as bombas foi sempre muito similar durante os primeiros anos de operação.

Três aspetos podem justificar que a política de manutenção acima apresentada não seja adotada: 1) impossibilidade devido ao número elevado de avarias das bombas e consequente falta de disponibilidade; 2) a inexistência, até agora, de uma ferramenta que permitisse determinar a evolução dos tempos de funcionamento; 3) outras práticas são adotadas. Para justificar a possibilidade de adoção de outras práticas é necessário ter em conta a localização geográfica da refinaria (junto à costa marítima) e a elevada exposição dos seus equipamentos às condições atmosféricas. Devido a estes dois pontos verifica-se

que os equipamentos estão sujeitos a elevados níveis de corrosão. Dado que a corrosão pode contribuir para limitar a vida das bombas centrífugas, a utilização similar de ambas permite aproveitar ao máximo o tempo de vida útil do conjunto.

6.4.2 Arranques do motor

Os dados da RTDB permitem ainda extrair informação acerca do número de arranques e paragens do motor (e, consequentemente, da bomba). Além disso, pode obter-se a duração dos períodos de funcionamento associados a esses arranques. Assim, apresenta-se na Figura 6.4, para cada bomba, a distribuição das durações de funcionamento.

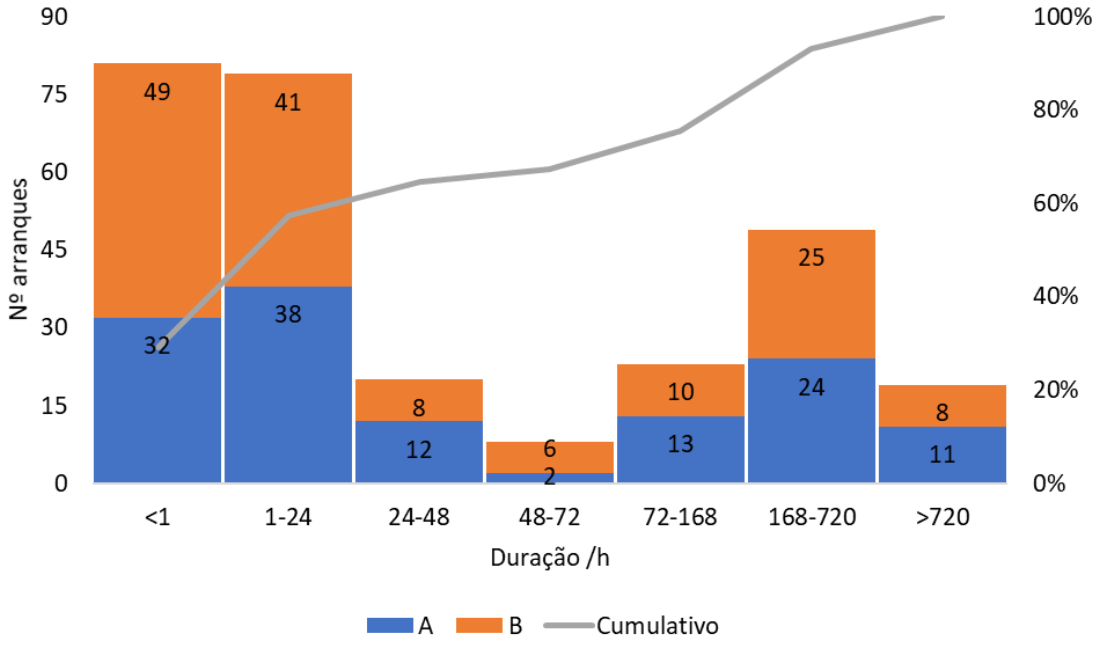


Figura 6.4: Distribuição das durações de funcionamento.

A análise da Figura 6.4 permite concluir que o número de arranques associados a tempos de funcionamento muito reduzidos é muito elevado. De facto, cerca de 30% dos arranques conduzem a períodos de funcionamento de duração inferior a uma hora. Além disso, mais de metade das durações de funcionamento são inferiores a um dia. Dado que os arranques e as paragens são momentos críticos nos equipamentos dinâmicos, poderá ser relevante, com vista a diminuir o número de avarias, procurar diminuir o número de arranques de curta duração.

Na Tabela 6.10 apresenta-se, para cada bomba, o número de arranques e a razão entre o tempo de funcionamento e o número de arranques. Embora a diferença entre o número de arranques não seja muito significativa entre bombas, a razão entre o tempo de funcionamento e o número de arranques é cerca de 30% mais baixa para a bomba B. Este aspeto justifica, em parte, que o número de avarias seja superior na bomba B, apesar desta apresentar um tempo de funcionamento inferior.

Salienta-se, por fim, que todos os resultados apresentados para os arranques e respetivas durações podem também ser obtidos para as paragens. Mais, as ferramentas desenvolvidas permitem determinar, ao minuto, as datas de paragem e arranque. Demonstra-se portanto, uma vez mais, que existe muito informação “oculta" nos dados da RTDB que

Tabela 6.10: Número de arranques e razão entre o tempo de funcionamento e o número de arranques por bomba

	Nº arranques	$\frac{\text{Tempo func.}}{\text{N° arranques}}$ /h
A	132	184.5
B	147	128.1

pode ser obtida utilizando as ferramentas adequadas.

6.5 Pressurizações

6.5.1 Número de pressurizações

Informação detalhada acerca das pressurizações pode também ser extraída da RTDB. Na Tabela 6.11 apresenta-se o número total de pressurizações por ano, bomba e tipo de empanque mecânico. Note-se que no total foram efetuadas cerca de 2000 pressurizações, sendo que não há diferenças significativas entre as bombas. Note-se ainda o número elevado de pressurizações em 2014. Relembra-se que esse ano foi atípico no que concerne substituições de empanques mecânicos[5], tendo sido efetuadas onze. Em 2012 foram efetuadas mais substituições de empanques mecânicos que em 2013, mas como os dados da RTDB são analisados apenas a partir de novembro desse ano, isso não vem refletido no número de pressurizações.

Tabela 6.11: Número de pressurizações por ano, bomba e tipo de empanque mecânico

		2012	2013	2014	2015	2016	2017		
A	Acop.	25	79	362	34	21	13	534	945
	Livre	26	335	7	26	9	8	411	
B	Acop.	4	31	114	64	49	33	295	1048
	Livre	210	70	251	127	19	76	753	
		265	515	734	251	98	130	1993	

Mais interessante que analisar o número de pressurizações total é considerar apenas as pressurizações que foram efetuadas nos períodos anteriores às falhas dos empanques mecânicos (Tabela 6.12). Note-se que apenas 10% das pressurizações foram efetuadas durante esses períodos. O número de pressurizações por ano, bomba e tipo de empanque mecânico é agora mais similar.

6.5.2 Pressurizações por dia da semana

A refinaria tem um contrato de manutenção preventiva com uma empresa subcontratada para pressurizar os empanques mecânicos todas as semanas. É também possível, com base nas ferramentas desenvolvidas, avaliar o cumprimento do contrato.

Comece por observar-se, na Tabela 6.13, a distribuição das pressurizações, nos períodos

[5] Ver Secção 3.4.

Tabela 6.12: Número de pressurizações por ano, bomba e tipo de empanque mecânico considerando apenas os períodos antes das falhas

		2012	2013	2014	2015	2016	2017		
A	Acop.	4	19	8	10	16	11	68	113
	Livre	3	7	7	12	9	7	45	
B	Acop.	4	10	9	5	11	18	57	90
	Livre	1	6	16	2	4	4	33	
		12	42	40	29	40	40	203	

anteriores às falhas, por dia da semana. Observa-se que o maior número de pressurizações é efetuado às quintas e sextas-feiras, o que sugere que as bombas em análise são, tipicamente, pressurizadas no final da semana (deve notar-se que os responsáveis pelas pressurizações têm uma lista extensa de bombas a pressurizar, ficando ao seu critério a ordem pela qual o fazem). Importa notar, contudo, que existem duas possibilidades para que uma pressurização seja efetuada: manutenção preventiva ou queda da pressão para valores inferiores a um dado alarme (6 bar). No segundo caso, um operador é responsável por repor a pressão para valores aceitáveis. Assim, importa também verificar que percentagem de pressurizações é efetuada a partir de valores abaixo do valor de alarme (Tabela 6.14).

Tabela 6.13: Número de pressurizações por dia da semana, bomba e tipo de empanque mecânico considerando apenas os períodos antes das falhas

		1[1]	2	3	4	5	6	7
A	Acop.	1	7	7	9	16	24	4
	Livre	1	6	3	3	9	22	1
B	Acop.	1	7	7	5	19	18	0
	Livre	2	2	6	5	12	5	1
		5	22	23	22	56	69	6

[1] Domingo

Tabela 6.14: Percentagem das pressurizações cuja pressurização de partida foi inferior a 6 bar por dia da semana, bomba e tipo de empanque mecânico considerando apenas os períodos antes das falhas

/%		1[1]	2	3	4	5	6	7
A	Acop.	0	43	43	33	44	29	25
	Livre	0	17	67	100	56	9	0
B	Acop.	0	71	71	80	32	17	0
	Livre	0	100	50	80	50	40	0
		0	50	56	64	43	20	17

[1] Domingo

A Tabela 6.14 demonstra que à sexta-feira apenas 20% das pressurizações foram efetuadas com a pressão inicial inferior à pressão de alarme (também se verificam valores baixos para as quintas-feiras). Assim, as tabelas anteriores permitem corroborar que no final da semana são efetuadas muitas pressurizações de prevenção, demonstrando-se que o contrato de manutenção preventiva está, pelo menos ocasionalmente, a ser cumprido.

Alternativamente, podia considerar-se que a avaliação do contrato pode ser efetuada através da razão entre o número de semanas em que os empanques mecânicos não estavam em falha e o número de pressurizações de prevenção. No entanto, verificou-se que há 622 semanas (a contagem é efetuada para cada bomba e tipo de empanque mecânico) em que os empanques mecânicos não estavam em falha (na contagem considerou-se o estado de funcionamento do empanque mecânico à meia-noite de sexta-feira), tendo-se observado apenas 203 pressurizações de prevenção. Demonstra-se assim que as pressurizações preventivas não são efetuadas todas as semanas.

Os factos apresentados no parágrafo anterior não devem ser entendidos como resultado do incumprimento do contrato. De facto, o que é possível concluir é que, eventualmente, a equipa de manutenção preventiva decide se a pressurização deve ser efetuada com base na pressão dos empanques mecânicos aquando das supostas datas de pressurização. Esta forma de atuação está de acordo com aconselhado por alguns autores [44]: as pressurizações devem ser apenas efetuadas quando a pressão é inferior ao alarme estabelecido, podendo o excesso de pressurizações em curtos intervalos diminuir a fiabilidade dos empanques mecânicos. Além disso, deve procurar-se que, tal como mencionado anteriormente, as pressurizações dos empanques mecânicos do plano 53B sejam espaçadas de, pelo menos, 28 dias [28], pelo que as distância temporal entre pressurizações estabelecida no contrato é curta e justifica a opção eventualmente tomada pelos responsáveis pelas pressurizações.

6.5.3 Queda de pressão abaixo dos 4 bar

Tal como se afirmou anteriormente, um dos aspetos que se observou durante a análise do sinal de pressão dos empanques mecânicos foi que a pressão cai abruptamente (para cerca de 2 bar) ao atingir valores em torno dos 4.5 bar. Isto pode ser justificado pela incapacidade de, a essa pressão, se manter o contacto entre os anéis estático e rotativo internos. Desta forma, o óleo de selagem escapa para o interior da bomba e, eventualmente, algum do fluido bombeado percorre o caminho inverso. Dado que este acontecimento é muito prejudicial ao funcionamento do empanque mecânico, apresenta-se na Tabela 6.15 informação relativa ao número de vezes em que a pressão caiu abaixo dos 4 bar em períodos anteriores às falhas.

Tabela 6.15: Número de vezes que a pressão cai abaixo de 4 bar antes do empanque falhar por ano, bomba e tipo de empanque mecânico

		2012	2013	2014	2015	2016	2017
A	Acop.	1	1	2	1	1	1
	Livre	1	1	0	4	2	3
B	Acop.	2	1	1	0	1	0
	Livre	3	0	2	0	0	0
		7	3	5	5	4	4

A informação apresentada na Tabela 6.15 permite concluir acerca da efetividade do operador a atuar nas situações em que a pressão é inferior ao valor de alarme. Demonstra ainda de que forma essa efetividade varia no tempo. Pode concluir-se que a sua forma de atuação se tem mantido estável.

6.6 Conclusões

No presente capítulo demonstrou-se que a utilização simultânea de dados de diferentes fontes permite uma análise mais detalhada e precisa das avarias dos equipamentos mecânicos.

Demonstrou-se também a importância da definição quantitativa de falha e a necessidade da definição do conceito de *empanque virtual*. O tempo médio de vida dos empanques mecânicos foi apresentado, tendo-se concluído que corresponde apenas a cerca de 11% do tempo previsto pela norma ANSI/API 682 [25] para estes dispositivos.

Demonstrou-se ainda que informação adicional e relevante pode ser retirada dos dados já existentes, nomeadamente no que concerne ao tempo de funcionamento e arranques do motor e pressurizações.

7. *Machine Learning*

7.1 O que é *Machine Learning*?

Machine Learning pode ser definido como "o campo de estudo que dá ao computador a habilidade de aprender sem ser explicitamente programado" [45]. Diz-se que um programa de computador está a "aprender da experiência E com respeito a alguma tarefa T e alguma métrica de *performance* P, se a sua *performance* em T, avaliada por P, melhora com a experiência E" [45]. De forma mais simples, pode dizer-se que "os algoritmos de *Machine Learning* usam métodos computacionais para aprender informação diretamente dos dados sem confiarem em equações pré-definidas como modelo" [46]. Ainda, tal como se mencionou no Capítulo 1 pode olhar-se para o *Machine Learning* como sendo o inverso de programar [4].

O *Machine Learning* é um campo de estudo contido num campo ainda maior, a Inteligência Artificial. A Inteligência Artificial é uma "área da ciência da computação que procura a criação de máquinas inteligentes que trabalham e reagem como os seres humanos" [47]. Não existindo pleno consenso quanto à definição de Inteligência Artificial [48], para o presente trabalho é suficiente entender que o grande objetivo desta área é "construir entidades inteligentes" [49] e que, para que tal seja possível, é fundamental que se consiga que essas entidades aprendam. Portanto, o *Machine Learning* desempenha um papel essencial na Inteligência Artificial. Isto é de tal forma verdade, que alguns autores [4] defendem que o crescimento do *Machine Learning* foi tal que eclipsou o seu campo de origem.

7.1.1 Porque usar *Machine Learning*?

Não se devendo olhar para o *Machine Learning* como a ferramenta que vai permitir resolver todos os problemas para os quais ainda não há resposta, também não se deve desprezar o seu potencial. De entre as várias tarefas para as quais há uma elevada probabilidade das ferramentas de *Machine Learning* serem usadas com sucesso, destacam-se:

1. Procurar soluções para problemas complexos para os quais as abordagens tradicionais não encontraram nenhuma ou, tendo encontrado, requeiram demasiadas afinações de parâmetros ou uma longa lista de regras [45].
2. Auxiliar o ganho de "intuições" acerca de grandes quantidades de dados [45].

Do ponto de vista da manutenção, tal como foi abordado no Capítulo 1, o *Machine Learning* permite integrar informação e aproveitar o potencial da grande quantidade de dados que, tipicamente, são armazenados.

7.2 Tipos de *Machine Learning*

Os sistemas de *Machine Learning* podem dividir-se em várias categorias, de acordo com vários aspetos [45]:

1. Se são ou não treinados com supervisão humana: *supervised*, *unsupervised*, *reinforcement* e *semisupervised learning*.
2. Se podem aprender de forma incremental: *batch* e *online learning*.
3. Se fazem previsões comparando os novos dados com pontos conhecidos ou se constroem modelos preditivos: *instance-based* e *model-based learning*.

7.2.1 *Supervised*, *unsupervised* e *reinforcement learning*

A primeira divisão estabelecida baseia-se na quantidade e tipo de supervisão que os sistemas recebem no treino [45]. Quando efetuada, esta supervisão ocorre antes do treino através da classificação dos dados.

Este aspeto permite a divisão dos sistemas *Machine Learning* em três categorias principais (*supervised*, *unsupervised* e *reinforcement learning*) e, ainda, numa quarta categoria que consiste simplesmente na combinação de modelos das categorias anteriores (*semisupervised learning*) [5; 45; 49].

Supervised learning

O modo de aprendizagem *supervised learning* divide-se em classificação e regressão. Neste, os algoritmos treinam sobre *inputs* e *outputs* conhecidos por forma a desenvolverem modelos preditivos capazes de gerar respostas razoáveis quando confrontados com novos dados [46; 50]. Dito de outra forma, estes "aprendem uma função que mapeia x em y, usando dados classificados (x, y) como exemplos de treino" [51].

Nas técnicas de classificação procura prever-se a resposta, ou seja, a classificação, a partir de um conjunto de valores de entrada [46]. Neste caso, as respostas são discretas (os modelos classificam os dados em categorias) e o conjunto de respostas possíveis é definido pelo programador. As técnicas de regressão prevêem respostas contínuas [46].

O exemplo clássico de uma tarefa de classificação é o filtro de *spam* [4; 45; 52]. Neste exemplo, o papel do modelo é classificar um dado e-mail como *spam* ou não *spam*. Para isso apoia-se em e-mails passados que, com base nas suas características (e.g. palavras presentes no e-mail [52]), foram categorizados de uma ou de outra forma. O sucesso do *Machine Learning* neste problema foi tão elevado que, atualmente, já existem algoritmos capazes não só de prever se um dado e-mail é *spam*, mas também de dividi-lo por várias categorias (problema de classificação multi-categoria, ao invés de classificação binária).

As flutuações de temperatura e procura de energia [46] e previsão do preço médio das casas de um dado distrito [45] são exemplos de problemas de regressão.

É ainda importante mencionar que as variáveis usadas num modelo de *Machine Learning* que, espera-se, contêm informação relevante acerca do problema e permitem ao modelo aprender com sucesso são denominadas por *feature*s (ou atributos). Uma *feature* pode ser diretamente medida (e.g. temperatura), calculada a partir dos sinais medidos (e.g. tempo de funcionamento) ou, simplesmente, gerada pelo programador (e.g. identificação do equipamento).

Os algoritmos mais populares de *supervised learning* são: *k-nearest neighbors* (kNN), *linear*

regression, *logistic regression*, *support vector machines* (SVMs), *decision trees*, *random forests*, *naive Bayes* e *neural networks*.

Unsupervised learning

Em *unsupervised learning* um algoritmo aprende padrões existentes nos *inputs* sem que *feedback* explícito (e.g. categorização dos dados) tenha sido fornecido [49]. Os sistemas de *unsupervised learning* procuram encontrar padrões escondidos e estruturas intrínsecas nos dados [46].

Quatro tarefas principais estão associadas a *unsupervised learning* [45]: *clustering*, visualização e redução de dimensionalidade, deteção de anomalias e *association rule learning*.

Os algoritmos de *clusterização* procuram dividir os exemplos em *clusters* de exemplos de natureza similar. A visualização está intrinsecamente ligada a esta tarefa (bem como à redução de dimensionalidade), mas distingui-se dela porque o *output* dos algoritmos de visualização são representações 2D ou 3D dos dados [45]. A redução de dimensionalidade procura simplificar os dados sem perder muita informação relevante [45]. Por sua vez, a deteção de anomalias é possível com algoritmos de *unsupervised learning* pois, treinando o modelo com exemplos normais, é imediato verificar se os novos exemplos se localizam fora do padrão (anomalias) [4; 45]. Por fim, a tarefa de *association rule learning* procura encontrar relações entre as *feature*s (e.g. verificar se, num supermercado, a compra de um dado produto se associa à de outro) [4; 45].

Na Tabela 7.1 apresentam-se exemplos de algoritmos de *unsupervised learning* em função das tarefas às quais estão associados.

Tabela 7.1: Exemplos de algoritmos de *unsupervised learning* em função das tarefas às quais estão associados [45]

Clustering
k-Means
Hierarchichal cluster analysis (HCA)
Expectation maximization
Visualização e redução de dimensionalidade
Principal component analysis (PCA)
Kernel PCA
Locally-linear embedding (LLE)
t-distributed stochastic neighbor embedding (t-SNE)
Association rule learning
Apriori
Eclat

Reinforcement learning

Em *reinforcement learning* um agente aprende a partir de uma série de reforços, sejam eles recompensas ou punições [49]. Note-se que se chamou *agente* ao sistema de aprendizagem [45]. O agente pode observar o ambiente, selecionar e cumprir tarefas [45]. Uma *política* (estratégia) define a ação que o agente deve executar em determinada situação

[45]. O objetivo de um sistema de aprendizagem de *unsupervised learning* é procurar as estratégias que dão mais recompensas a longo prazo [45]. Esta forma de atuação é claramente similar à dos seres humanos [53].

Este tipo de aprendizagem é usado, por exemplo, por *robots* para aprender a andar e no programa *AlphaGo*, da Google, que em 2016 venceu o campeão mundial do jogo *Go* [45] (embora o Deep Blue, da IBM, tenha vencido o campeão mundial de xadrez em 1996 [54], a maior complexidade do *Go* dificultou, até recentemente, que o mesmo sucedesse neste jogo).

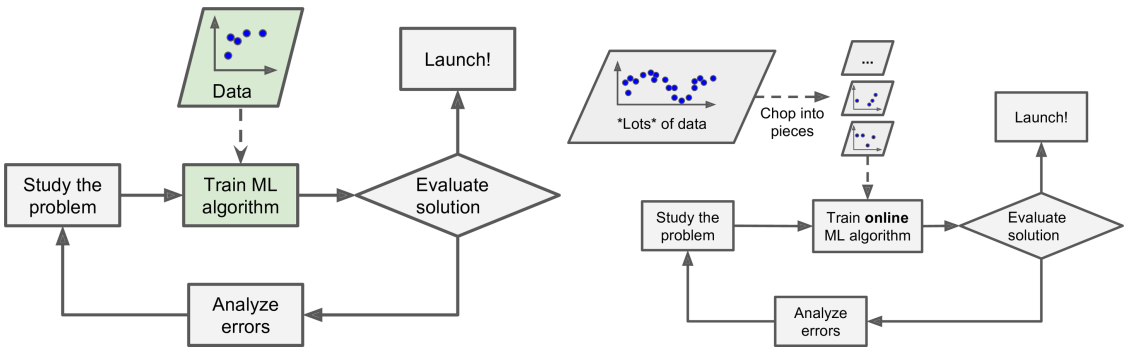
Semisupervised learning

Alguns algoritmos são capazes de lidar, em simultâneo, com uma combinação de dados de treino categorizados e não categorizados [45]. O *Google Photos* é um exemplo deste tipo de aprendizagem, onde a componente *unsupervised* do programa é capaz de reconhecer pessoas em fotos e, depois do utilizador identificar uma pessoa numa dada foto (cria uma categoria), compreender que a mesma pessoa, que foi "etiquetada", está presente em várias fotos [45]. Tipicamente estes algoritmos são combinações de algoritmos de *supervised* e *unsupervised learning*.

7.2.2 *Batch* e *online* learning

A segunda divisão que se efetuou baseia-se na capacidade dos sistemas de *Machine Learning* aprenderem de forma incremental. Os sistemas baseados em *batch learning* (também conhecido como *offline learning*) são incapazes de aprender instantaneamente com a introdução de novos dados, ou seja, para treinar requerem sempre acesso a todos os dados (Figura 7.1a) [45]. Deve notar-se, contudo, que estes sistemas continuam a ser atualizados no tempo. No entanto, tal obriga a que se volte a treinar, avaliar e lançar o sistema [45].

Alternativamente, *online learning* permite ao sistema treinar continuamente (*on the fly*) ao ser alimentado com exemplos (individuais ou em pequenos grupos; Figura 7.1b) [45]. Este tipo de aprendizagem permite também eliminar informação já usada no treino (que, caso os recursos computacionais sejam limitados, pode ser importante) e permitem *out-of-core learning* (necessário quando a memória ocupada pelos dados é superior à RAM disponível [55]) [45]. O *learning rate* é um parâmetro importante destes sistemas e refere-se à rapidez de adaptação do modelo a novos dados [45].

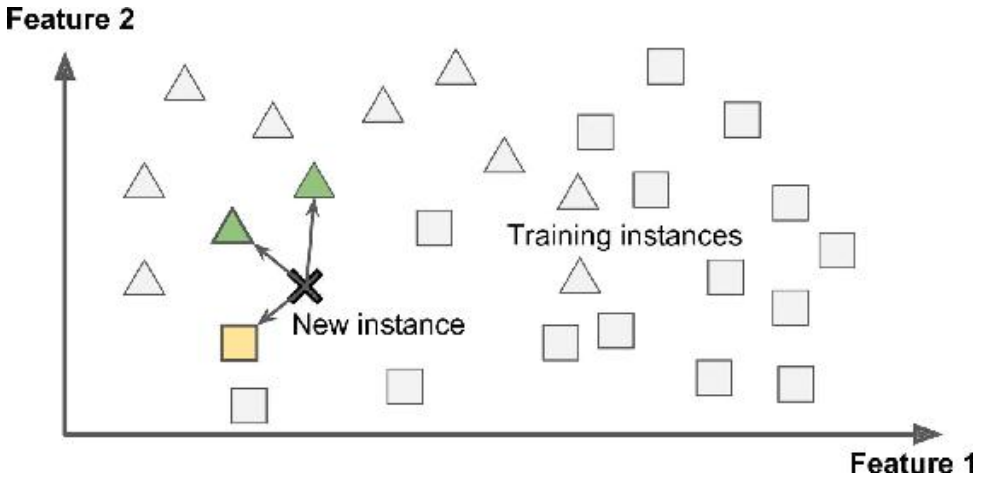


(a) *Batch learning* (b) *Online learning*

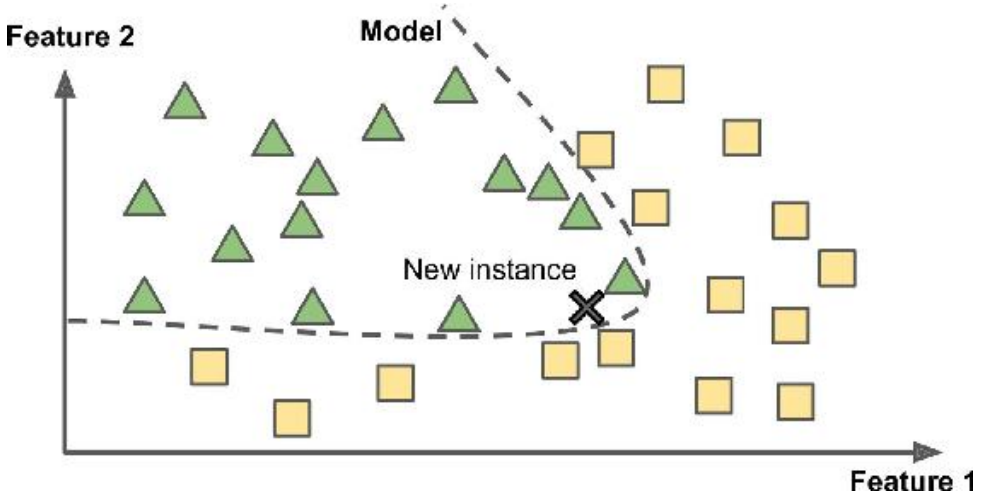
Figura 7.1: Divisão baseada na capacidade dos algoritmos aprenderem de forma incremental [45].

7.2.3 *Instance-based* e *model-based learning*

Uma última divisão dos sistemas de *Machine Learning* baseia-se na forma como as previsões são efetuadas, ou seja, na forma como os modelos generalizam[1] [45]. Dito de outra forma, se os sistemas se baseiam em medidas de similaridade (distância) aos exemplos conhecidos, então o seu modo de aprendizagem é *instance-based learning*, enquanto que se existe um modelo matemático associado a estes sistemas, é *model-based learning* [45]. Na Figura 7.2 demonstra-se o modo como a generalização é feita para estes modos de aprendizagem.



(a) *Instance-based learning*



(b) *Model-based learning*

Figura 7.2: Generalização para diferentes modos de aprendizagem [45].

Os modelos de *instance-based learning* não têm realmente uma fase de treino e requerem, para cada nova previsão, o acesso a todos os dados de treino [56].

7.2.4 Outras formas de divisão

Nas subsecções anteriores fez-se a divisão dos sistemas de *Machine Learning* em várias categorias considerando diferentes aspetos. Muitos outros aspetos podiam ter sidos em

[1] Ver Secção 7.6.

conta e, portanto, muitas divisões diferentes podiam ser apresentadas. Na Figura **??** apresenta-se outra forma de dividir os algoritmos de *Machine Learning*.

7.3 *Workflow* de um projeto de *Machine Learning*

A Mathworks sugere que um projeto de *Machine Learning* siga o *workflow* apresentado na Figura 7.4 [46]. Claro que seria demasiado ambicioso considerar que o *workflow* apresentado é o ideal para todos os projetos, dado que cada um tem as suas peculiaridades. No entanto, este é genérico e dá excelentes indicações das etapas que devem ser ultrapassadas para que um sistema de *Machine Learning* possa ser implementado. Ainda assim, considera-se que o *workflow* apresentado carece de duas etapas muito importantes:

1. A etapa inicial deve ser a formulação do problema e definição de objetivos [45]. Esta é, provavelmente, a etapa crucial de todo o projeto.
2. Entre as duas primeiras etapas e entre as etapas 3 e 4 da Figura 7.4 deve fazer-se uma exploração dos dados para se entender melhor o problema [45]. Histogramas [45], representações gráficas de combinações de duas *feature*s [45; 58] e *box plots* [59; 60] são excelentes apoios para perceber a natureza dos dados, detetar erros na definição das *feature*s, avaliar características destas (e.g. *skewness* [61]) e detetar a presença de *outliers*.

Ao longo da presente secção serão tecidas considerações acerca de cada uma das etapas.

7.3.1 Formulação do problema e definição dos objetivos

Um projeto de *Machine Learning* tem de começar, sempre, pelo estabelecimento rigoroso e claro dos objetivos (nesta fase admite-se que já foi identificado o problema e que se considera que as ferramentas de *Machine Learning* são as que têm mais potencial para o resolver). É necessário entender que um sistema de *Machine Learning* cumpre uma tarefa muito específica e a definição de objetivos muito vagos pode conduzir a que, após vários meses de trabalho e, eventualmente, elevadas quantias de dinheiro investidas, se conclua que o que o modelo de *Machine Learning* desenvolvido prevê não é exatamente o que se pretende.

Exemplifica-se, de seguida, o que se pretende transmitir com o parágrafo anterior com o que podia ter sido o processo de definição do objetivo do presente projeto.

Imagine-se que como objetivo inicial se estabelecia: *prever falhas*. Embora seja um ponto de partida interessante (de facto, já permite olhar para os modelos de *Machine Learning* do ponto de vista da manutenção preditiva), é ainda muito vago. Uma forma de o começar a especificar é, por exemplo, responder às perguntas: 1) pretende-se identificar modos de falha ou prever o tempo até à falha? 2) Pretende-se prever as falhas de todos os equipamentos do complexo industrial, de grupos de equipamentos ou de equipamentos específicos? 3) Faz sentido procurar prever todos os tipos de falha ou será mais sensato procurar prever as falhas associadas aos modos de falha mais relevantes (seja a relevância medida pela frequência de falhas, criticidade, ou custo)?

A resposta às três perguntas formuladas permitiu estabelecer, para o presente projeto, o objetivo: *prever o tempo até à falha dos empanques mecânicos de duas bombas centrífugas responsáveis pelo bombeamento de resíduo de vácuo da coluna de destilação a vácuo para a unidade de visbreaking*.

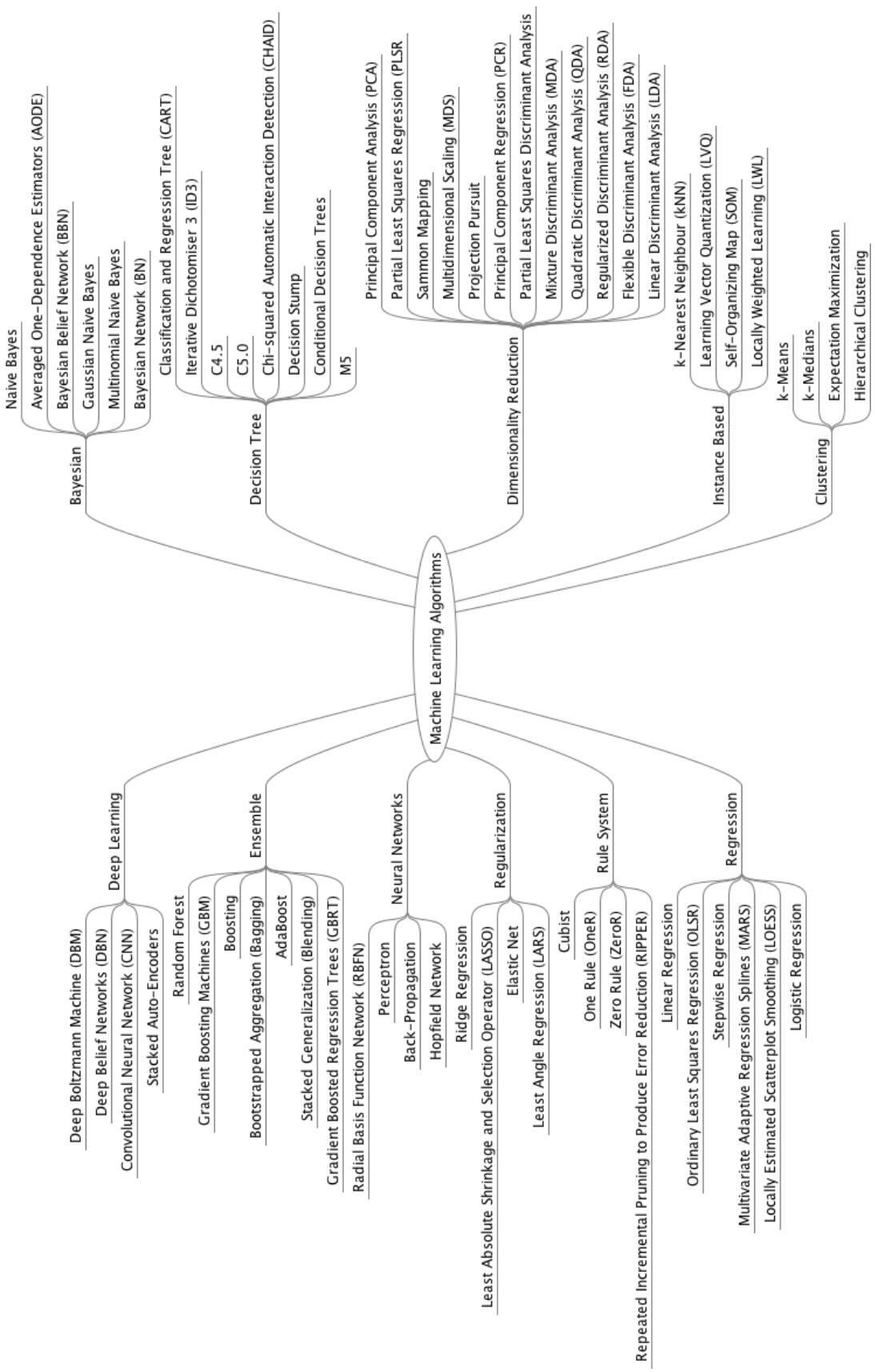


Figura 7.3: Divisão alternativa dos algoritmos de *Machine Learning* [57].

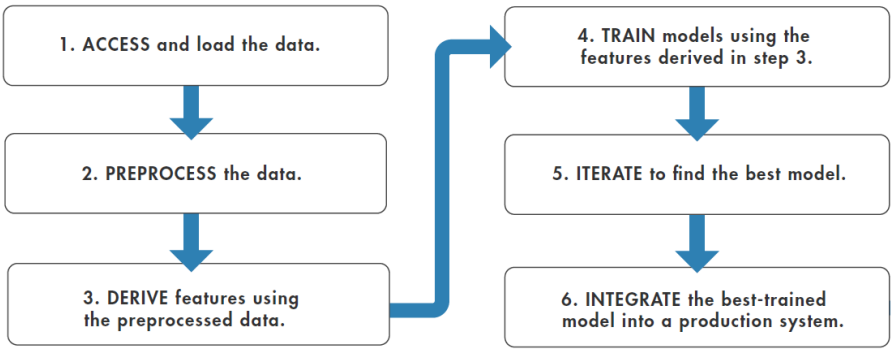


Figura 7.4: *Workflow* de um projeto de *Machine Learning* [46].

Assim, à primeira pergunta respondeu-se: *prever o tempo de falha*. Entende-se que num contexto industrial há mais potencial na utilização de modelos que procuram prever instantes de falha do que na de modelos que pretendem identificar modos de falha. Isto porque, tipicamente, mais do que se compreender *como* vai falhar um dado equipamento, pretende determinar-se *quando* este vai falhar, de modo a que a produção seja adaptada em função da possível indisponibilidade do equipamento. Num contexto académico e em situações em que o objetivo principal é compreender o fenómeno de falha de um dado componente, então a abordagem de identificação de modos de falha poderá revelar-se muito interessante porque pode permitir perceber como determinadas variáveis se combinam para originar um dado modo de falha. Ainda assim, a procura de modelos que, de alguma forma, combinem ambas as abordagens poderá fornecer uma solução com maior potencial e com a possibilidade de um uso mais generalizado.

À segunda pergunta respondeu-se: *equipamentos específicos*. No futuro, após as técnicas de *Machine Learning* atingirem uma maior maturidade e, sobretudo, se compreender melhor como estas podem ser aplicadas na manutenção, poderá almejar prever-se todas as falhas dos equipamentos de um dado complexo industrial (provavelmente nessa altura o projeto, mais do que desenvolver um modelo de *Machine Learning* específico, passará pela integração de vários modelos, cada um deles com o objetivo de prever falhas muito específicas). Para já, e até tendo em conta que ainda são poucos os casos de sucesso de aplicação destes sistemas em manutenção, faz mais sentido procurar prever falhas de equipamentos específicos. Também se entende que ainda não é o momento de procurar prever falhas em grupos de equipamentos (e.g. bombas centrífugas). Em primeiro lugar, porque embora os equipamentos de um dado grupo tenham funções idênticas (e.g. bombear um fluido), as condições de operação são, tipicamente, muito diferentes para cada equipamento (e.g. bombear água é muito diferente de bombear petróleo). Além disso, porque a frequência e tipo de falhas em equipamentos idênticos e que cumprem exatamente a mesma função não é a mesma (basta atentar nas conclusões do Capítulo 3). Ainda assim, se os equipamentos forem idênticos (mesma tecnologia) entende-se que é possível encontrar um modelo de *Machine Learning* que faça previsões para as falhas de todos os equipamentos em simultâneo, desde que se use a identificação do equipamento como *feature* (para que o modelo seja capaz de identificar as diferenças entre equipamentos).

À terceira pergunta respondeu-se: *prever as falhas associada ao modo de falha mais relevante*. No Capítulo 3 demonstrou-se que o modo de falha *fuga empanque* é claramente o modo

de falha mais crítico. Além disso, os dados disponíveis[2] são úteis para a previsão deste modo de falha. Assim, o mais sensato é começar por procurar um modelo para prever os instantes de falha a ele associados. A previsão de falhas associadas a outros modos de falha poderá requerer a instalação de outro tipo de sensores (e.g. a medição da vibração em locais de interesse poderá permitir a previsão de falhas de rolamentos, além de, possivelmente, melhorar a qualidade dos modelos de previsão de falhas de empanques mecânicos).

Antes ainda de se pensar nas especificidades do projeto de *Machine Learning*, é necessário verificar se já existe alguma ferramenta no complexo industrial que procure cumprir o objetivo que foi estabelecido. Se existir, é necessário perceber porque é que não funciona (se funcionasse não seria necessário começar um projeto *Machine Learning*) e tentar incorporar esse conhecimento no novo projeto. Deve ainda verificar-se se já existem em funcionamento sistemas de *Machine Learning* e, eventualmente, pensar o novo sistema de forma a que, no futuro, possa ser integrado com os sistemas já existentes. Dado que o presente projeto é pioneiro quer na tentativa de prever instantes de falha de empanques mecânicos, quer na utilização de sistemas de *Machine Learning*, avançou-se esta etapa.

Posteriormente, e tendo em conta que já se estabeleceu um objetivo particular e claro, pode começar a fazer-se perguntas mais específicas quanto ao *Machine Learning* propriamente dito: 1) deve optar-se por modelos de *supervised*, *unsupervised*, *reinforcement learning* ou, eventualmente, por combinações de modos de aprendizagem? 2) se *supervised learning*, classificação ou regressão? 3) pretende-se que os modelos treinem imediatamente à medida que novos dados são obtidos (*batch learning* ou *online learning*)? 4) há limitações computacionais (e.g. capacidade de armazenamento)?

Após análise do problema, e tendo presente o objetivo proposto, optou-se por *supervised learning* e, em particular, classificação. Para previsão de tempos de falha é evidente que os modelos de regressão são os mais adequados, já que, idealmente, se pretende um tempo de falha, não uma categoria/estado. Ainda assim, a simplicidade dos modelos de classificação justifica a sua utilização numa primeira abordagem. A simplicidade justifica também que se tenha optado por classificação binária (divide-se o estado de funcionamento em duas categorias: funcionamento estável e funcionamento "pré-instável"). Relativamente ao treino *on the fly* do sistema, considera-se que não é prioritário que se faça a atualização constante do modelo, visto que não é expectável que a natureza dos dados se altere a uma taxa tão elevada que conduza a uma rápida degradação do modelo (é suficiente voltar a treinar, avaliar e implementar um novo modelo, por exemplo, a cada seis meses - fazível se o processo for automático ou necessitar de intervenção humana mínima). As limitações computacionais não se consideram, numa primeira fase, relevantes, mas entende-se que alguns aspetos têm de ser melhorados antes da implementação efetiva do modelo desenvolvido, nomeadamente a facilidade e rapidez de acesso aos sinais medidos.

Note-se que parte da tomada de decisão efetuada ao longo da presente subsecção pode ser baseada na Figura 7.5. A árvore de decisão apresentada permite ainda avaliar se se deve optar pelo uso de ferramentas *Machine Learning* e, caso não, quais são as alternativas disponíveis.

Nesta fase, se todas as etapas apresentadas tiverem sido efetuadas, os objetivos e as linhas guiadoras de um projeto de *Machine Learning* estão definidos e pode começar a

[2] Ver Capítulo 2.

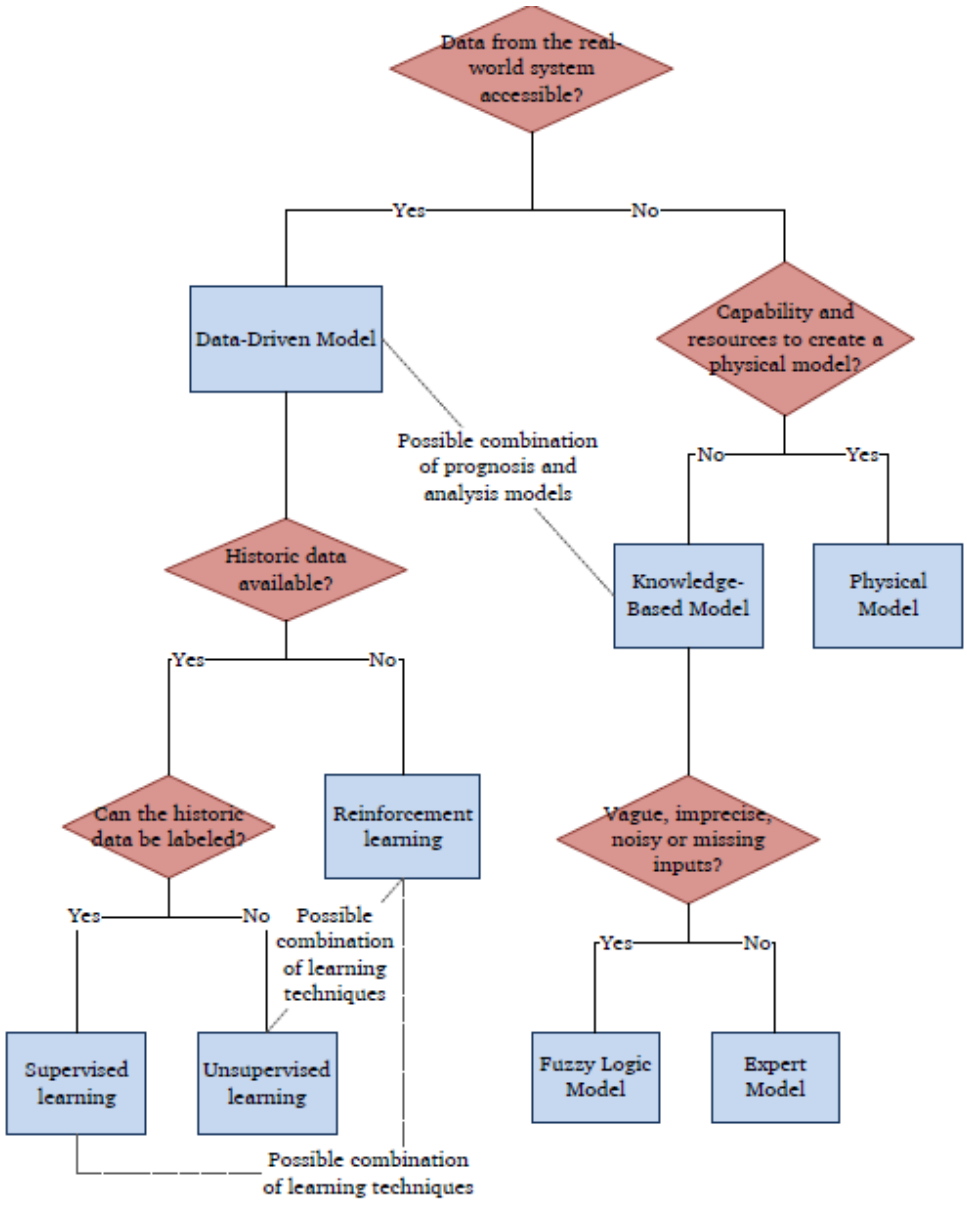


Figura 7.5: Árvore de decisão de apoio à seleção de uma ferramenta de manutenção preditiva [5].

dispensar-se atenção ao cumprimento das restantes etapas que conduzem à implementação do sistema.

7.3.2 Acesso aos dados, pré-processamento e derivação de *features*

No presente projeto, o acesso aos dados não levantou problemas de maior porque estes já se encontravam integrados na RTDB e são da mesma natureza (numéricos). Ainda assim, deve ter-se em conta que no caso geral os dados tomam diversas formas e tamanhos [46]. Assim, pode ser necessário combinar informação de natureza distinta, como sinais de sensores, texto e imagens [46]. Deve ainda esperar-se alguma desorganização e falta de dados [46].

Após o acesso aos dados, e antes de se iniciar o pré-processamento, deve proceder-se à exploração e visualização destes. A visualização dos dados, além de permitir entender melhor o problema, permite avaliá-los quanto aos aspetos abordados no parágrafo anterior. No presente projeto, todas as etapas até ao pré-processamento estão apresentadas nos Capítulos 2 a 4.

O pré-processamento é, em conjunto com a etapa 3 da Figura 7.4, parte integrante do *feature engineering* (Figura 7.6). Esta é uma das etapas que requer conhecimentos mais especializados (e.g. um algoritmo de detenção de objetos requer conhecimento especializado de processamento de imagem) [5; 46]. Tarefas típicas de pré-processamento são remover *outliers*[3] e tendências, completar pontos em falta e normalizar dados[4] [46].

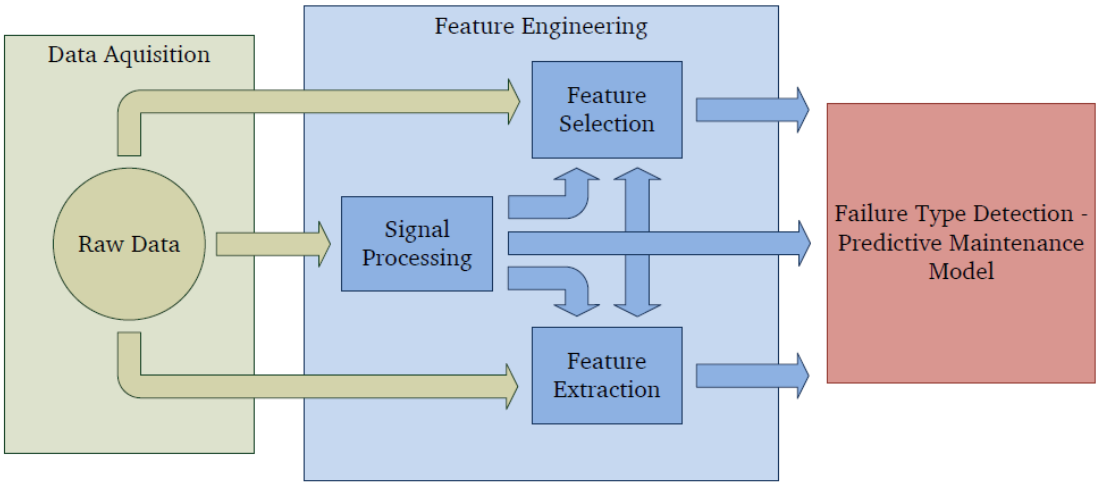


Figura 7.6: *Feature engineering* [5].

Salienta-se que, embora o *workflow* apresentado na Figura 7.6 permita compreender o conceito de *feature engineering* e as tarefas a ele associadas, se considera que a procura do modelo *Machine Learning* com melhor *performance* é mais dinâmica, não sendo o processo tão linear. O *workflow* apresentado na Figura 7.7 demonstra o dinamismo necessário para a obtenção dos melhores modelos. No presente projeto, todas as etapas de *feature engineering* (com exceção de *feature selection*) foram apresentadas no Capítulo 5. O *feature engineering* é fundamental para a obtenção de modelos adequados e mais atenção é-lhe dedicada na Secção 7.4.

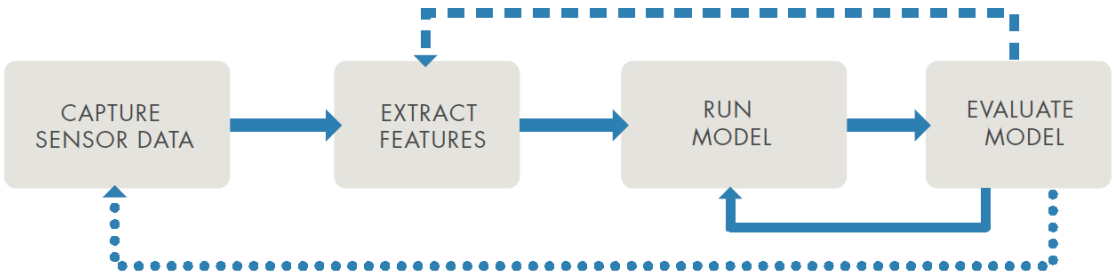


Figura 7.7: Processo iterativo para obtenção do modelo com melhor *performance* [50].

[3] Em muitas aplicações os *outliers* fornecem informação crucial e não devem ser removidos [46].
[4] Ver Secção 7.4.

Importa ainda salientar que, no presente projeto, as etapas desde o acesso aos dados até à derivação de *feature*s foram, de longe, as que consumiram mais tempo. Tal deve-se não só à especificidade, mencionada acima, das ferramentas de pré-processamento, mas também à necessidade de perceber a fundo o problema e os sinais medidos por forma a determinar que *feature*s podem ser importantes. O desenvolvimento de funções para as determinar é também demorado [62]. Acrescenta-se que tal era, à partida, esperado. De facto, Domingos [63] refere que é comum que nas primeiras vezes que as pessoas se envolvem em projetos de *Machine Learning* fiquem surpreendidas com a quantidade reduzida de tempo que é realmente despendida a fazer *Machine Learning* propriamente dito.

Após derivação das *feature*s os dados devem ser agrupados numa tabela (pode variar em função do *software* utilizado) cujas linhas são os exemplos de treino e as colunas os valores que as *feature*s tomam para esses exemplos (Tabela 7.2). Se se tratar de um problema de *supervised learning*, então os exemplos devem também estar classificados. Nesta altura recomenda-se uma nova exploração e representação gráfica dos dados.

Tabela 7.2: Exemplo de tabela usada como *input* dos modelos de *Machine Learning*.

$feature_1$	$feature_2$	...	Classificação
30.60	9.34	...	Negativo
59.260	11.14	...	Positivo
⋮	⋮	⋱	⋮
8.410	10.30	...	Negativo

7.3.3 Seleção e treino de modelos

Na subsecção anterior demonstrou-se que no final da etapa 3 da Figura 7.4 todos os exemplos estão agrupados numa tabela. Isto é útil porque, como passo seguinte, propõe-se o treino e avaliação de um conjunto diversificado de algoritmos [63], ou seja, não se recomenda que as tarefas do *feature engineering* sejam executadas exaustivamente imediatamente após a derivação das *feature*s. Por treinar um modelo entende-se usar os dados conhecidos (não todos, como se verá na Secção 7.5) para determinar os parâmetros de um dado modelo.

Tal abordagem deve-se ao facto de não ser possível à partida determinar qual é o algoritmo mais adequado para um dado problema (da mesma forma que não existe um algoritmo que funcione para todos os problemas - apesar de existir uma procura pelo algoritmo-mestre [4]) [46; 50]. Assim, a determinação do melhor modelo é um processo iterativo que se baseia numa metodologia de tentativa-erro. Importa ainda notar que embora haja uma infinidade de algoritmos de *Machine Learning*, é um pequeno conjunto deles que é responsável pela grande maioria das aplicações *Machine Learning* bem sucedidas [4].

Ao treinar-se e avaliar-se uma grande variedade de modelos logo numa fase inicial começa a perceber-se quais são os que têm mais potencial e, possivelmente, o *feature engineering* poderá ser adaptado em função dos primeiros resultados obtidos. Para que esta etapa seja bem sucedida, é necessário que as métricas de avaliação dos modelos tenham sido escolhidas em concordância com os objetivos estabelecidos. Nesta fase não deve

procurar afinar-se exaustivamente os hiperparâmetros[5].

Embora seja discutível, entende-se também que nesta fase não deve ser dada muita atenção ao modo como cada algoritmo treina e faz previsões de novos exemplos. De facto, atendendo à enorme quantidade de algoritmos disponíveis, quem se limitar apenas a aplicar aqueles que compreende corre o risco de não testar modelos com elevado potencial para a resolução do seu problema. Por outro lado, quem procurar perceber todos os algoritmos disponíveis, corre o risco de permanecer eternamente nesta etapa. Numa fase posterior, quando o número de algoritmos em avaliação tiver sido reduzido (com base na *performance*), então faz todo o sentido que se dispense atenção ao modo de funcionamento dos algoritmos. De facto, considera-se que isto é fundamental para perceber que hiperparâmetros ajustar e como os ajustar.

Faz-se ainda notar que compreender o princípio de funcionamento de um dado algoritmo é diferente de compreender quais são os passos que ele efetua entre a receção de um dado conjunto de valores e a emissão de um resultado (resposta): muitos algoritmos são caixas-negras. Quanto à utilização de algoritmos que são caixas negras entende-se que, embora desconfortável, é inevitável, dado que o potencial destes é, tipicamente, superior ao dos algoritmos para os quais é possível perceber o processo de decisão. Ainda assim, entende-se que a seleção de uma boa métrica de avaliação e, eventualmente, a demonstração prática da capacidade de previsão de um dado modelo podem permitir diminuir esse desconforto. Tornar o processo de decisão dos algoritmos de *Machine Learning* mais transparente é uma tarefa que não cabe aos utilizadores dos algoritmos.

Menciona-se ainda que embora no corrente projeto se avaliem diversos algoritmos de forma separada, a tendência atual é procurar uma combinação de modelos que conduza a um sistema de previsão de elevada *performance*: *model ensembles* [63].

Na Tabela 7.3 apresentam-se alguns dos algoritmos de classificação disponíveis na aplicação *Classification Learner* do Matlab e as suas características gerais. Parte destes algoritmos são usados no presente projeto para cumprir a etapa apresentada nesta subsecção.

As características relacionadas com a computação não devem ser desprezadas. Por exemplo, a rapidez de treino dificulta a afinação dos hiperparâmetros porque técnicas de procura exaustiva das melhores combinações de hiperparâmetros e/ou *feature*s não podem ser aplicadas. Este ponto é também afetado pela rapidez de previsão (e.g. os algoritmos *k-nearest neighbor* não requerem treino - *instance-based learning* -, mas o tempo de previsão dificulta a afinação). A memória pode determinar a viabilidade de determinado algoritmo ser implementado, por exemplo, num *smartphone*. A avaliação geral apresentada pode ser um ponto de partida para selecionar os modelos a treinar numa fase inicial.

7.3.4 Processo iterativo para encontrar o melhor modelo

Nesta fase, onde se espera que já tenha sido encontrado um grupo mais restrito de algoritmos (que apresentam *performance* razoável), o objetivo passa por afinar os hiperparâmetros e, sobretudo, recorrer ao *feature engineering* com o intuito de obter um modelo com elevada *performance* que, eventualmente, possa ser implementado.

No que concerne à melhoria da *performance* de um dado modelo, duas vias opostas podem ser seguidas: simplificar ou adicionar complexidade [46].

[5] Um hiperparâmetro é um parâmetro do algoritmo, não do modelo [45].

Tabela 7.3: Alguns dos algoritmos disponíveis na aplicação *Classification Learner* do Matlab e as suas características gerais [46; 58]

Algoritmo	Rapidez previsão	Rapidez treino	Uso memória	Afinação requerida
SVM (linear)[1]	Rápido	Rápido	Pequeno	Mínima
Decision trees[2]	Rápido	Rápido	Pequeno	Alguma
SVM (não linear)[3]	Lento	Lento	Médio	Alguma
Nearest Neighbor[4]	Moderado	Mínimo	Médio	Mínima
Naive Bayes[5]	Rápido	Rápido	Médio	Alguma
Ensembles[6]	Moderado	Lento	Varia	Alguma

[1] Bom para pequenos problemas com fronteiras de decisão lineares.
[2] Bom generalista, mas sensível a *overfitting*.
[3] Bom para muitos problemas binários e lida bem com elevada dimensionalidade.
[4] Baixa *accuracy*, mas fácil de usar e interpretar.
[5] Muito usado para texto, incluindo filtros de *spam*.
[6] Elevada *accuracy* e boa *performance* para pequenos a médios conjuntos de dados.

Simplificar

Embora simplificar não seja sempre a solução (daí que a outra alternativa apresentada seja aumentar a complexidade), é possível melhorar os modelos através, por exemplo, da seleção de um número limitado de *feature*s (idealmente as que têm maior poder preditivo) [46]. De facto, é preferível um modelo simples que é capaz de generalizar a um modelo complexo que, embora capaz de efetuar previsões com grande sucesso em dados conhecidos, prevê mal quando é confrontado com novos dados [46]. O princípio que a solução mais simples é, na maior parte das vezes, a melhor (navalha de Occam [4; 49]) deve ser apreciado com cautela [63].

A simplificação torna também o modelo mais fácil de compreender, mais robusto e computacionalmente mais eficiente [46].

A diminuição do número de *feature*s (através de *feature selection* e *feature extraction*) é muito importante porque logo depois do *overfitting*[6], o número de dimensões (igual ao número de *feature*s) é o maior problema em *Machine Learning* [63]. Este problema é conhecido como a maldição da dimensionalidade (*curse of dimensionality*) e resulta da observação que algoritmos que têm boa *performance* com poucas dimensões vêem o seu comportamento deteriorar-se com o aumento do número de variáveis [63].

Existem outras formas de simplificar os modelos, como *pruning* de ramos de uma *decision tree* [46; 49] (consiste em cortar ramos desnecessários; alternativamente, pode limitar-se o número máximo de divisões à partida), regularização [64] (consiste em adicionar um termo à função objetivo usada na determinação dos parâmetros do modelo - que tipicamente se pretende minimizar - com o intuito de diminuir o valor dos parâmetros) e remoção de algoritmos de um *ensemble*.

[6] Ver Secção 7.6.

Adicionar complexidade

A forma mais comum de adicionar complexidade é, como se afirmou anteriormente, recorrendo a *ensamble models* [46]. Alternativamente, pode adicionar-se mais fontes de informação (e, consequentemente, mais *features*) [46]. Com esta alternativa pretende-se encontrar, por exemplo, outras variáveis que sejam capazes de dividir os dados entre determinadas categorias que, até agora, eram indistinguíveis (e.g. num algoritmo para classificar a atividade física, a adição do sinal do giroscópio pode permitir distinguir corrida de dança, enquanto que com informação apenas do acelerómetro tal divisão não é possível [46]).

7.3.5 Implementação do modelo

Embora o presente projeto não tenha como objetivo a implementação do modelo, importa fazer algumas considerações acerca de como o trabalho efetuado pode ser transferido para o ambiente industrial.

Antes de mais, importa relembrar que o projeto foi desenvolvido em Matlab, um *software* que requer licença e não está disponível na refinaria. Trabalhar num *software* que não o de implementação pode parecer contra-intuitivo, mas alguns autores [65] indicam que o uso de uma linguagem de programação de alto-nível permite diminuir o tempo de projeto, mesmo com a necessidade de, na fase final, converter todo o código para uma linguagem de nível inferior. Tal prende-se com a facilidade de manipulação de variáveis e deteção de erros associada a linguagens de alto-nível. Acresce a isto que o Matlab possibilita a geração de código C/C^{++}, Python, Java ou .NET através da aplicação *Matlab Coder* [50].

Conclui-se assim que as dificuldades de implementação em ambiente industrial não estão nesta etapa. De facto, estas encontram-se logo na fase inicial de acesso aos dados, sendo necessário, como já se mencionou anteriormente, encontrar uma solução para aceder mais fácil e rapidamente aos dados da RTDB.

7.3.6 Proposta de *workflow*

Com base nas considerações tecidas ao longo da presente secção sugere-se que um projeto de *Machine Learning* se baseie no *workflow* apresentado na Figura 7.8.

7.4 *Feature engineering*

A Figura 7.9 apresenta a forma como, tipicamente, a *performance* de um algoritmo varia com o aumento da dimensionalidade. Embora intuitivamente se espere que o aumento do número de *features* melhore a *performance* de um dado modelo (dado que mais *features* contêm mais informação), o que se observa é que existe um número ótimo de *features* a partir do qual a *performance* se começa a degradar [66]. Este problema, que tinha sido mencionado na Subsecção 7.3.4, denomina-se maldição da dimensionalidade e está intrinsecamente ligado ao *overfitting*.

De forma simples, o problema surge porque os dados se tornam mais esparsos com o aumento do número de dimensões (assumindo que o número de exemplos é mantido), ou seja, a densidade espacial dos exemplos de treino diminui com o aumento dimensional [66].

A Figura 7.10 permite compreender a maldição da dimensionalidade e a forma como este

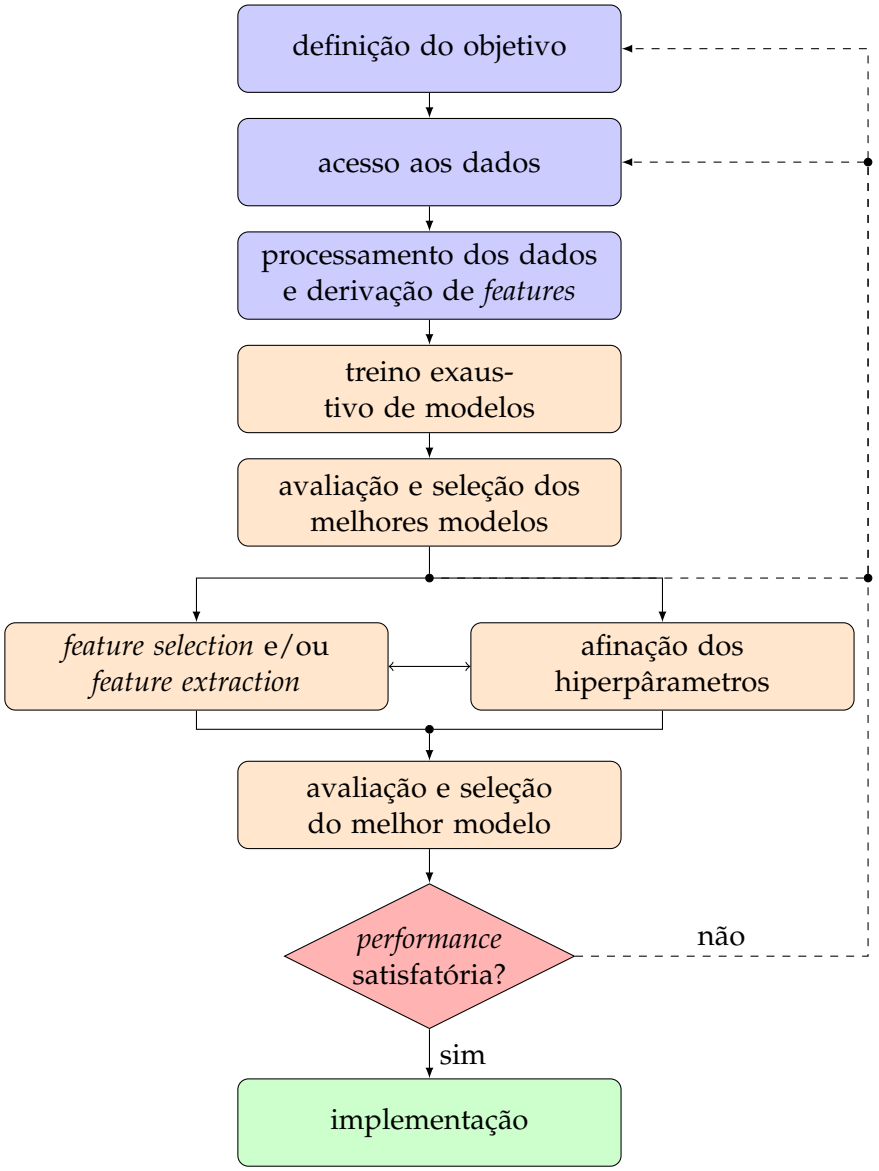


Figura 7.8: Proposta de *workflow* de um projeto de *Machine Learning*.

fenómeno está ligado ao *overfitting*. Imagine-se que se pretendia um modelo para separar linearmente exemplos de cães e de gatos. Com uma *feature* (Figura 7.10a) é provável que não se encontre uma fronteira que divida corretamente os exemplos. Com duas *feature*s (Figura 7.10b) já se pode obter uma divisão mais correta verificando-se, contudo, que existem exemplos localizados no lado errado da fronteira de decisão. Uma terceira *feature* (Figura 7.10c) pode ser, dada a maior dispersão dos dados, o suficiente para se obter uma divisão perfeita entre os exemplos das duas categorias. Note-se ainda que uma divisão linear numa dada dimensão é similar a usar um modelo com uma classificação não linear em dimensões inferiores (Figura 7.10d). O exemplo apresentado demonstra que é mais fácil dividir os dados em dimensões mais elevadas. Repare-se, contudo, que os dados divididos são os exemplos de treino, ou seja, o aumento do número de dimensões permite que as fronteiras de decisão se ajustem aos exemplos de treino. O problema é que, tipicamente, este ajuste conduz a que o modelo seja capaz de aprender as particularidades de um certo conjunto de dados, quando o que realmente se pretende é que este

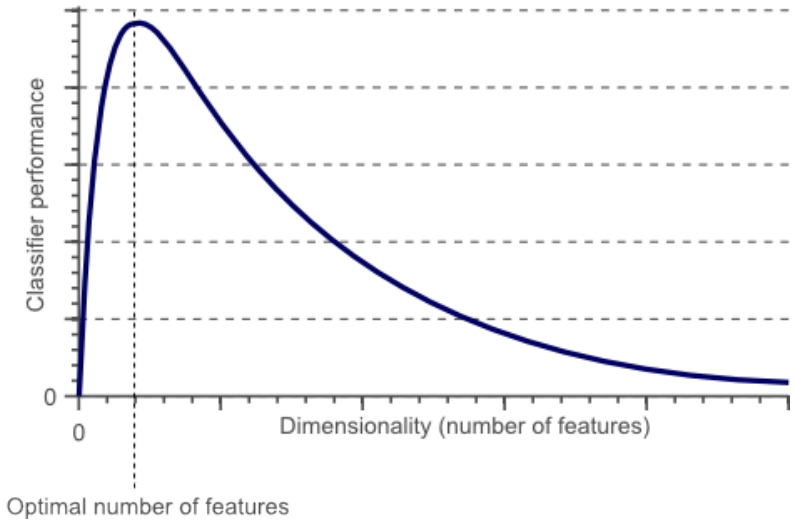
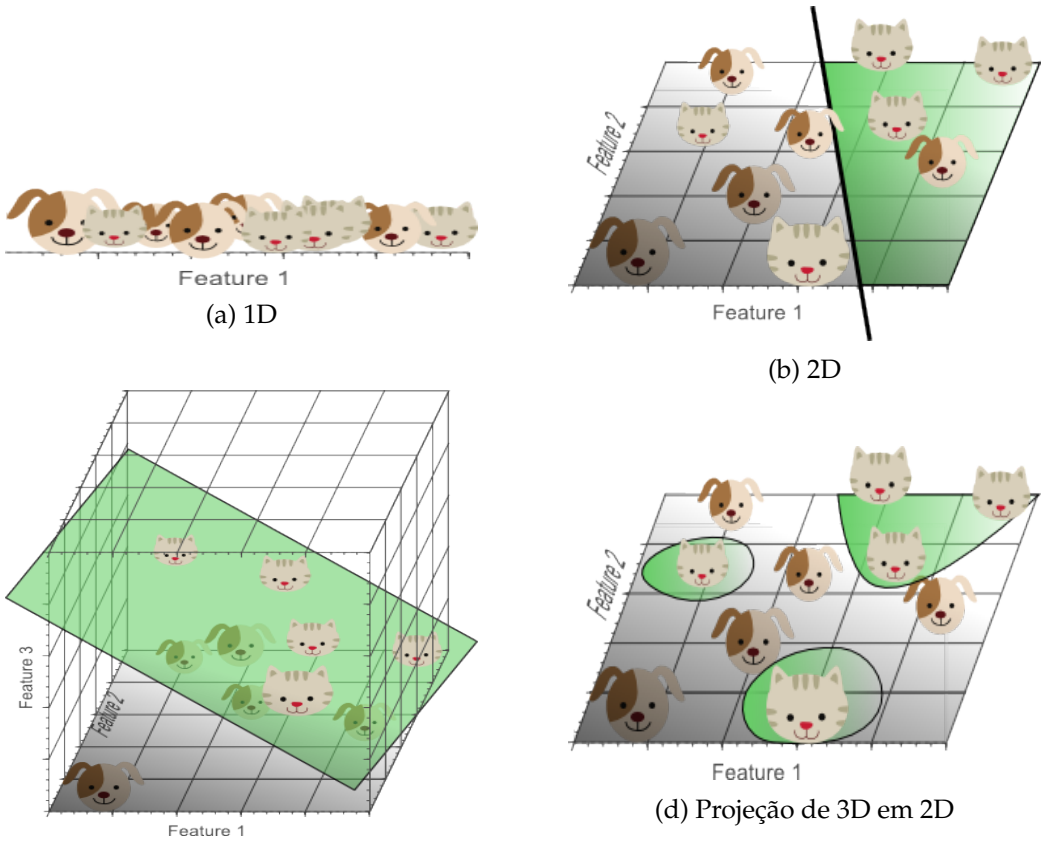


Figura 7.9: Evolução da *performance* com o aumento da dimensionalidade [66].

seja capaz de generalizar. Assim, modelos com elevado número de dimensões são mais propensos a *overfitting*.



(a) 1D

(b) 2D

(c) 3D

(d) Projeção de 3D em 2D

Figura 7.10: Impacto do aumento da dimensionalidade na divisão dos exemplos (adpatado de [66]).

Outras explicações para o problema da dimensionalidade podem ser encontradas em

[66].

Tendo-se reconhecido que a utilização de um número elevado de *features* não significa (bem pelo contrário) modelos com melhor *performance*, é necessário encontrar ferramentas que permitam diminuir o número de *features* sem que se perca informação relevante. É nesta fase que as tarefas de *feature selection* e *feature extraction* ganham relevância. De facto, Domingos [63] afirma que o *feature engineering* é a chave dos projetos de *Machine Learning* e que, não raras vezes, os sinais medidos não são adequados à aprendizagem, sendo necessário construir *features* a partir deles que o são. Deve notar-se que não é possível saber à partida qual é o número de *features* que conduz à melhor *performance* [66]. Este depende da quantidade de exemplos de treino disponíveis, da complexidade das fronteiras de decisão e do tipo de algoritmo utilizado [66].

Feature selection

Como *feature selection* entende-se a seleção, de entre as *features* existentes, das *features* mais representativas [5; 45; 67]. A forma mais simples de diminuir o número de *features* é avaliar a correlação entre elas e eliminar as altamente correlacionadas [5; 67]. Outros métodos como o *minimum-Redundancy Maximum-Relevance* (mRMR) [68] e *Neighbourhood Component Analysis* (NCA) [50; 69] revelam um elevado potencial de seleção de *features* representativas. Este último método fornece um vetor com a importância relativa das *features* (Figura 7.11) [50]. Deve ter-se presente, contudo, que a combinação de *features* boas individualmente não conduz necessariamente a um modelo com boa *performance* (não há garantia que funcionem em conjunto) [68]. Além disso, uma dada combinação de *features* pode ser a melhor para um dado modelo e conduzir a uma péssima *performance* noutro.

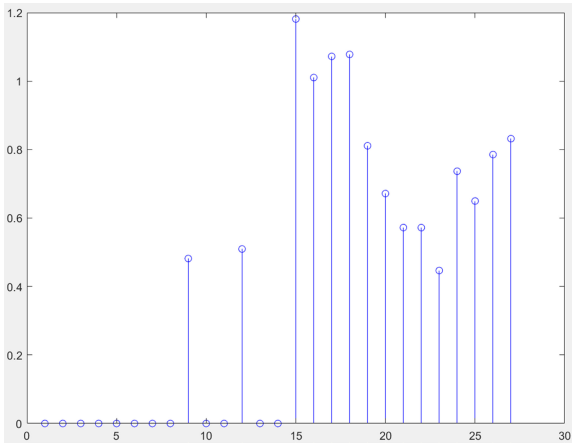


Figura 7.11: Exemplo de aplicação de NCA para seleção de *features* [50].

Uma outra forma de seleção de *features* é avaliar todas as combinações possíveis de um dado número de *features*. Esta metodologia, que foi aplicada no corrente projeto, é muito dependente da rapidez de treino e previsão dos modelos e, tipicamente, não pode ser utilizada.

Feature extraction

Por outro lado, *feature extraction* é a combinação de *features* existentes com o intuito de produzir *features* mais informativas [5; 45; 46]. Ou seja, consiste na transformação de

features de um espaço de elevada dimensionalidade para um espaço de dimensões mais reduzidas [64; 67]. O *feature extraction* pode também ter como motivação a compressão e visualização de dados [64]. O *Principal Component Analysis* (PCA) é o método de redução de dimensionalidade mais utilizado e baseia-se na procura das direções que preservam a máxima quantidade de variância (Figura 7.12) [45; 64]. *Factor Analysis* e *Nonnegative Matrix Factorization* são outros exemplos de técnicas de redução de dimensionalidade [46].

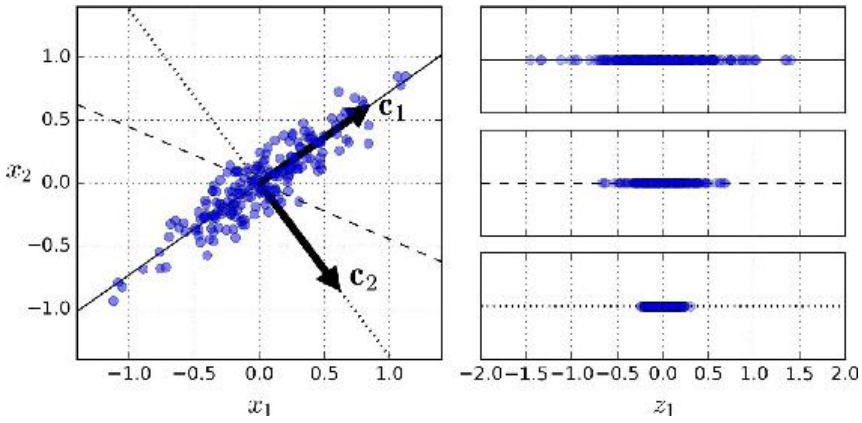


Figura 7.12: Exemplo de aplicação de PCA para redução de dimensionalidade de 2D para 1D [45].

Feature scaling

Muitas vezes as *features* apresentam diferentes escalas, isto é, variam entre valores muito distintos. Alguns algoritmos não funcionam adequadamente nestas condições [45] sendo portanto necessário reduzir todas as variáveis à mesma escala (*feature scaling*). As duas técnicas mais utilizadas são [45]: *min-max scaling* (*normalization*) e *standardization*.

Min-max scaling consiste em dividir um conjunto de valores ao qual foi subtraído o seu valor mínimo pela diferença entre os valores máximo e mínimo [45]. Desta forma, garante-se que os valores ficam compreendidos entre 0 e 1. Ainda assim, este método tem dificuldades em lidar com *outliers* [45].

Standardization consiste em dividir um conjunto de valores ao qual foi subtraída a sua média pela sua variância [45]. Assim, os novos valores têm média nula e variância unitária [45].

7.5 Treino e métricas de avaliação

Dado que, como mencionado anteriormente, é comum não se ter acesso ao processo de decisão de um dado modelo, a definição de métricas de avaliação adequadas é fundamental. Tão importante como as métricas de avaliação é saber em que dados estas devem ser aplicadas. Por exemplo, se o modelo for avaliado quanto à sua capacidade de classificar os dados de treino, é provável que os resultados obtidos sejam demasiado otimistas.

Técnicas de validação

Existem diversas técnicas de validação, sendo que duas das mais utilizadas são [70; 71]:

1. *k-fold* (Figura 7.13): os dados são divididos de forma aleatória em k subconjuntos mais pequenos com, aproximadamente, o mesmo tamanho e o modelo é treinado em cada subconjunto e avaliado nos restantes.
2. *holdout* (Figura 7.14): os dados são exatamente divididos em dois conjuntos com um rácio de divisão bem definido (que é função do número de observações), um conjunto de treino (*training set*) e um conjunto de teste (*test set*).

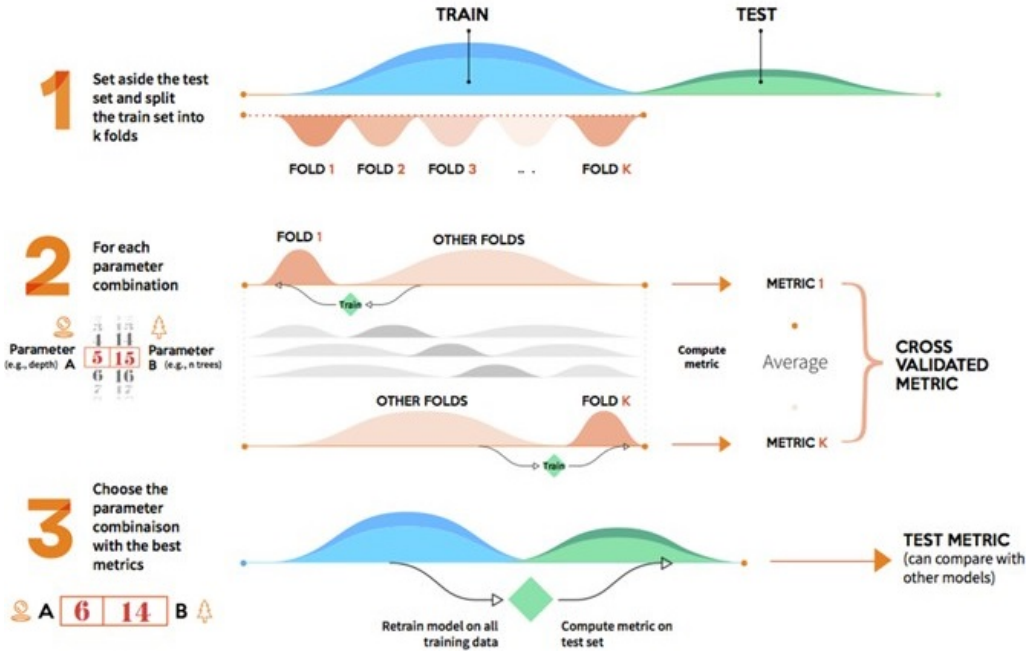


Figura 7.13: Técnica de validação *k-fold* [71].

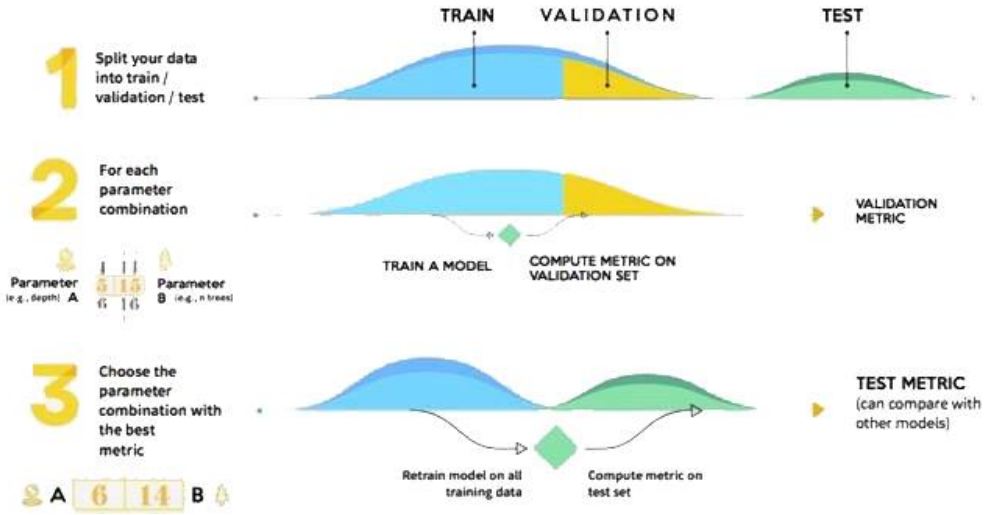


Figura 7.14: Técnica de validação *holdout* [71].

No presente projeto opta por utilizar-se, pela sua simplicidade e baixo custo computacional, a validação *holdout*. Com esta técnica de validação deve ainda fazer-se uma segunda divisão do conjunto de treino num subconjunto de treino e num subconjunto de desenvolvimento (*developer set*) [45; 51]. O primeiro subconjunto deve ser usado para o treino efetivo dos modelos, enquanto o segundo deve ser usado para avaliar os modelos e afinar

os hiperparâmetros. Se fosse considerada apenas a avaliação feita sobre o subconjunto de desenvolvimento corria-se o risco de se lançar um sistema de *Machine Learning* com uma *performance* inferior à esperada, já que os hiperparâmetros são afinados para melhorar as métricas de avaliação nesse subconjunto. Ou seja, o conjunto de teste deve ser utilizado apenas para avaliar o modelo que se considera mais adequado à resolução do problema após todos os hiperparâmetros terem sido definidos (Géron [45] defende mesmo que durante as etapas de processamento e visualização dos dados o conjunto de teste deve já estar definido e não deve ser observado, por forma a que as decisões efetuadas não sejam enviesadas pelos dados que são usados como avaliação). É comum usar-se o rácio 80-20% para dividir os dados em conjunto de teste e de avaliação [45].

Antes de se avaliar o modelo final com o conjunto de teste deve treinar-se o modelo escolhido com a totalidade dos dados de treino, por forma a diminuir a quantidade de dados desperdiçados, isto é, usados apenas em avaliação. De igual forma, antes do modelo ser implementado na prática deve ser treinado com todos os dados (conjuntos de treino e teste).

Tipicamente, a divisão dos dados pelos vários conjuntos é feita de forma aleatória. No presente projeto optou por dividir-se os dados com base no tempo, tendo-se usado os dados até 2017 como conjunto de treino e os dados de 2017 como conjunto de teste. A necessidade desta abordagem prende-se com a elevada similaridade entre dados temporalmente próximos (para a mesma bomba e tipo de empanque mecânico). Assim, a avaliação num conjunto de dados obtido de forma aleatória seria similar à avaliação do modelo nos dados de treino (resultados demasiado otimistas). A abordagem escolhida permite perceber como o modelo se irá comportar realmente na prática. De facto, avaliar o modelo com base nos dados de 2017 é similar a assumir-se que se implementou um modelo no final de 2016 que efetuou previsões durante o ano de 2017 (para as quais já se conhece as classificações reais).

Métricas de avaliação

Existem diversas métricas de avaliação, sendo que a mais adequada depende dos objetivos do sistema de *Machine Learning*. Para facilitar a otimização dos modelos, deve escolher-se uma única métrica de avaliação, dado que a melhoria de uma dada métrica está associada, normalmente, à degradação de outra [51].

A métrica mais simples é a *accuracy* e consiste simplesmente na razão entre o número total de previsões efetuadas com sucesso e o número total de previsões efetuadas [45]. Num problema de classificação binária esta métrica é, normalmente, uma boa escolha para métrica a otimizar. Ainda assim, nos casos em que uma das categorias tem muito mais exemplos que a outra (*skewed datasets*), pode não ser a mais adequada [45].

Por sua vez, as matrizes de confusão (*confusion matrices*) permitem perceber como o modelo se comporta na previsão de cada categoria. Cada linha da matriz de confusão representa a categoria real, enquanto que as colunas representam as categorias previstas pelo modelo (Tabela 7.4) [45]. Assim, na primeira linha são representados os verdadeiros-positivos (TP) e os falsos-negativos (FN) e na segunda linha são representados os falsos-positivos (FP) e os verdadeiros-negativos (TN). A *accuracy* pode ser vista, alternativamente, como a razão entre a soma dos TP e TN e o número de previsões total.

Apesar de a matriz de confusão fornecer mais informação do que a *accuracy*, é de difícil uso na otimização de um dado modelo porque contém pelo menos quatro componentes.

Tabela 7.4: Matriz de confusão para um problema de classificação binária

		Previsão	
		Positivo	Negativo
Real	Positivo	TP	FN
	Negativo	FP	TN

Métricas mais concisas e que se baseiam nela são a *precision* e a *recall* [45].

A *precision* define-se como a razão entre o número de verdadeiros-positivos e a soma de verdadeiros-positivos com falsos-positivos [45]. Assim, esta métrica mede a percentagem de previsões positivas que o são realmente.

A *recall* (também conhecida como *sensivity* e *true positive rate*) define-se como a razão entre o número de verdadeiros-positivos e a soma de verdadeiros-positivos com falsos negativos [45]. É uma métrica da capacidade do modelo em detetar os exemplos positivos.

A *precision* e a *recall* não são independentes: o aumento de uma implica a diminuição da outra [45; 72]. A Figura 7.15 apresenta um exemplo da variação das duas métricas com um dado hiperparâmetro de um modelo. O *trade-off* entre estas duas métricas é fácil de perceber: se for pretendido um algoritmo capaz de detetar todos os exemplos positivos, então a *precision* será baixa, porque para que ele possa detetar todos os positivos tem de estabelecer uma fronteira de decisão que, inevitavelmente, vai conter muitos elementos negativos; por outro lado, é possível ter elevada confiança que um dado exemplo é positivo quando o modelo prevê que é positivo (elevada *precision*) se o modelo "aliviar" a sua fronteira de decisão e permitir que alguns dos exemplos positivos não sejam detetados.

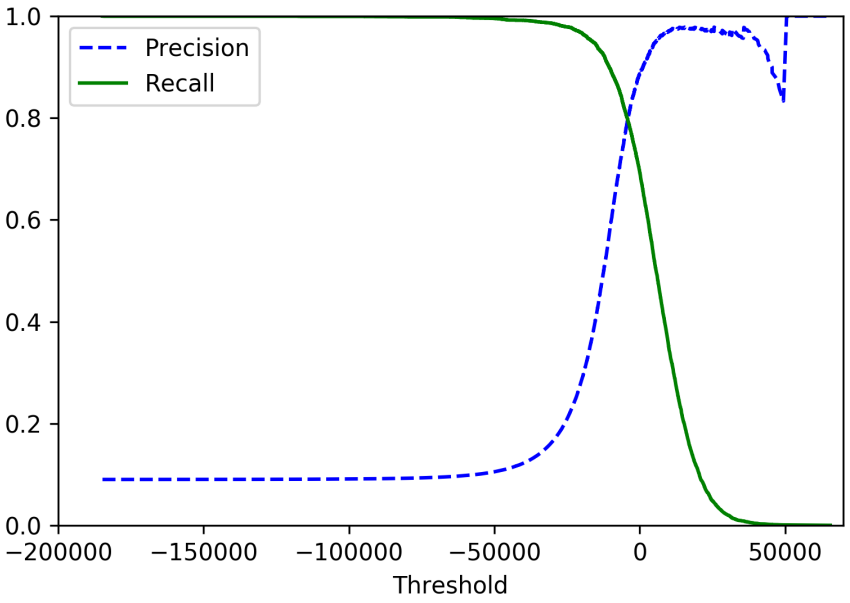


Figura 7.15: Exemplo do *trade-off* entre *precision* e *recall* [72].

Por vezes, não é claro se um sistema *Machine Learning* deve ser otimizado para *precision* ou *recall*. O F_1 *score* é uma média harmónica das duas métricas que, ao contrário da média normal, dá mais peso a valores mais baixos, pelo que um modelo com um elevado F_1 *score* tem *precision* e *recall* elevados [45].

7.6 Principais desafios do *Machine Learning*

Ao longo do presente capítulo foram indiretamente abordados os principais problemas de *Machine Learning* e apresentadas algumas soluções para os ultrapassar. Nesta secção apresentam-se alguns problemas ainda não abordados e aprofundam-se outros. De forma simplista, os problemas de *Machine Learning* têm dois responsáveis máximos [45]: os dados e os algoritmos.

7.6.1 Quantidade insuficiente de dados

Um aspeto fundamental do *Machine Learning* é que os algoritmos requerem grandes quantidades de dados para funcionarem efetivamente [45; 63; 73]. Assim, ao invés de procurar melhores algoritmos é preferível, se possível, recolher mais dados [63]. Acresce a isto que quando se possui uma quantidade muito elevada de dados se acabe por usar algoritmos mais simples porque os mais complexos demoram demasiado tempo a aprender [63]. Na Figura 7.16 apresenta-se a evolução da *performance* de vários algoritmos com o aumento da quantidade de dados. Note-se que a *performance* da maioria dos modelos para quantidades elevadas de dados é muito similar.

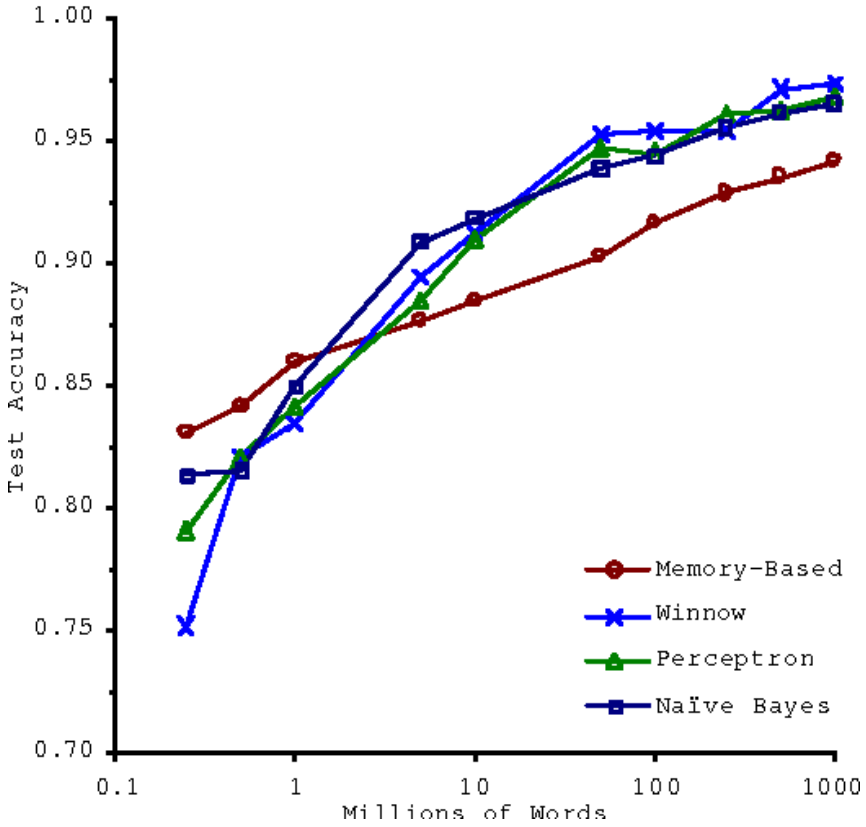


Figura 7.16: Importância da quantidade de dados [45].

Na Figura 7.17 procura demonstrar-se a similaridade de *performance* dos algoritmos com quantidades elevadas de dados: os diferentes algoritmos podem produzir fronteiras de decisão muito distintas enquanto continuam a fazer as mesmas previsões nas regiões de interesse (onde os exemplos de treino estão mais concentrados) [63], isto é, as grandes diferenças entre algoritmos verificam-se, sobretudo, nas regiões junto às fronteiras de decisão.

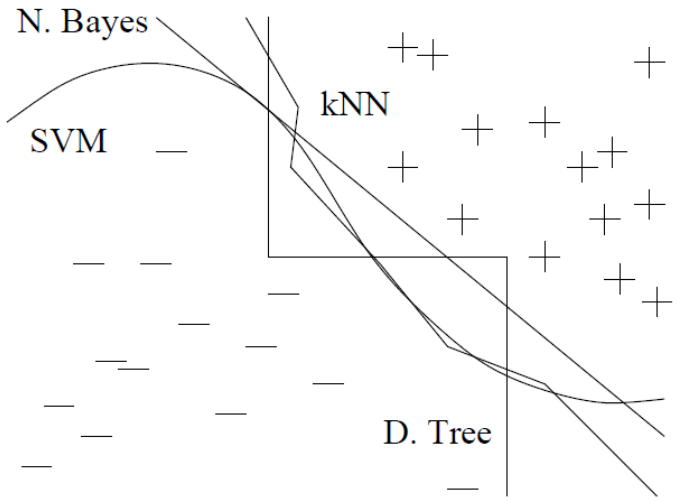


Figura 7.17: Diferentes fronteiras de decisão podem conduzir a classificações similares [63].

7.6.2 Dados de treino não representativos ou de baixa qualidade

Para que os algoritmos de *Machine Learning* tenham capacidade de generalizar é necessário que os dados de treino sejam representativos dos novos casos para os quais se pretendem efetuar previsões [45]. Por generalizar entende-se a capacidade do modelo fazer previsões corretas de exemplos que ainda não "viu" [70].

A não representatividade dos dados pode advir de [45]: i) amostra demasiado pequena (*sampling noise*); ii) amostras de tamanho suficiente que foram obtidas usando métodos de amostragem incorretos (*sampling bias*).

A baixa qualidade dos dados é também impeditiva da obtenção de modelos com elevadas *performances*. neste contexto, baixa qualidade significa elevada quantidade de erros (e.g. categorias mal atribuídas), *outliers* e ruído (e.g. baixa qualidade de medição de sinais) [45].

7.6.3 *Overfitting* e *underfitting*

Como mencionado anteriormente, o *overfitting* é o maior problema do *Machine Learning*. *Overfitting* significa que o modelo é capaz de explicar muito bem os dados de treino, mas não de generalizar [74]. Em oposição ao *overfitting*, existe o *underfitting* [4; 45; 63; 74]. Assim, a afinação de um dado modelo deve procurar um equilíbrio que permita que, por um lado, o modelo não "alucine" (*overfitting*) e, por outro, não seja "cego" (*underfitting*) [4]. Estes dois termos estão também ligados aos conceitos de *variance* e *bias*, respetivamente (Figura 7.18).

A distinção entre *bias* e *variance* é facilmente entendida usando um jogo de dardos como exemplo (Figura 7.19) [63]. *Bias* significa que o algoritmo tende a aprender consistentemente a mesma coisa errada [63]. *Variance* ocorre quando não existe um padrão nos erros de previsão [4].

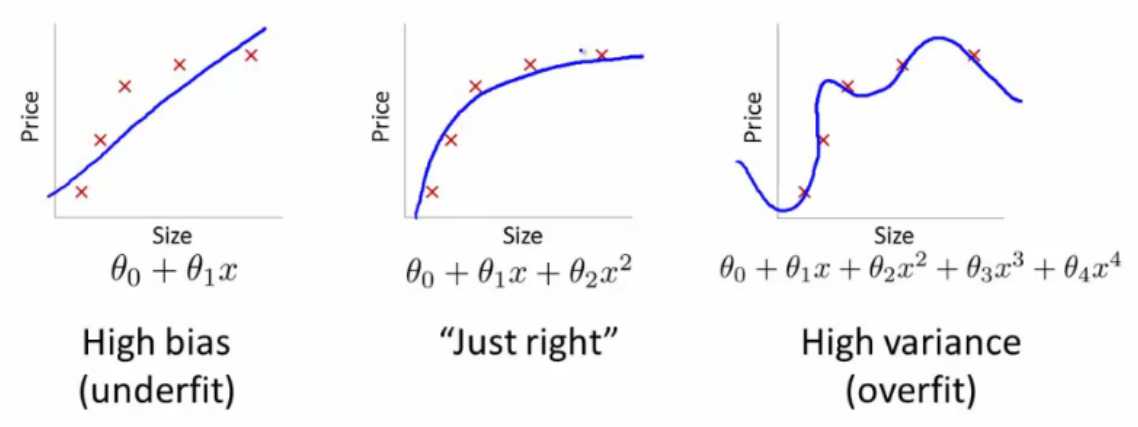


Figura 7.18: *Bias-variance trade-off* [74].

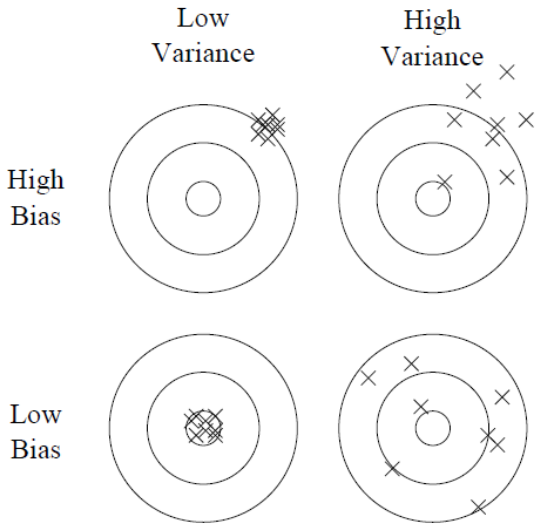


Figura 7.19: *Bias* e *variance* explicados com base num jogo de dardos [63].

7.7 Conclusões

Ao longo do presente capítulo introduziram-se os conceitos fundamentais de *Machine Learning*. A definição e os tipos de *Machine Learning* foram apresentados. Abordaram-se temas importantes como o *feature engineering*, o treino e as métricas de avaliação e os principais desafios do *Machine Learning*. O *workflow* de um projeto de *Machine Learning* foi apresentado e justificado.

No que concerne ao *workflow*, partiu-se de um geral, sugerido pela Mathworks, e fizeram-se diversas adaptações, tendo-se proposto um *workflow* que se entende que conduzirá à aplicação com sucesso das ferramentas de *Machine Learning* em manutenção (nomeadamente na previsão de falhas). O cumprimento das várias etapas foi apresentado (ou então foram indicados os capítulos onde tal foi efetuado). Apresentaram-se também alguns aspetos relevantes, tais como a necessidade de definir de forma clara o objetivo, a importância do *feature engineering*, a relevância dos dados, a falta de transparência do processo de decisão, o *overfitting* e a maldição da dimensionalidade.

8. Aplicação de modelos de *Machine Learning*

8.1 Introdução

Na Tabela 5.5 resumiram-se os sinais medidos e as variáveis obtidas a partir dos sinais medidos e/ou da sua combinação com os dados do SAP. Com exceção de *I*, cuja informação está armazenada na variável *motor*, todas as variáveis apresentadas são consideradas, numa fase inicial, como potenciais *features*. Acresce a essa lista a identificação da bomba centrífuga e do tipo de empanque mecânico, perfazendo-se um total de 19 *features*. Atendendo à maldição da dimensionalidade[1], começou por procurar-se, embora não exaustivamente, diminuir o número de *features*.

A diminuição inicial do número de *features* foi conseguida através de uma análise cuidada do significado das variáveis. O recurso à correlação entre variáveis resultou numa diminuição ainda mais significativa. De seguida, procedeu-se à classificação dos dados e à divisão destes pelos conjuntos de treino (subconjuntos de treino e desenvolvimento) e teste.

Após concluídas as etapas anteriores, procedeu-se à primeira aplicação de modelos de *Machine Learning*, onde uma variedade de algoritmos foi treinada e avaliada. Para os modelos de treino e previsão rápidos foi ainda efetuada uma procura exaustiva das *features* mais relevantes e avaliados os modelos mais promissores. O método NCA foi também aplicado para avaliar a importância das *features*.

Por fim, avaliou-se o efeito da limitação do tempo de vida do empanque mecânico na *performance* dos modelos e demonstrou-se o potencial do *Machine Learning* no aumento da compreensão de um fenómeno físico.

Todas as figuras apresentadas ao longo do presente capítulo que têm as *features* indicadas por um número são descodificadas no Anexo A.

8.2 Breve seleção de *features*

Uma análise cuidada da lista de *features* permite detetar duas que não devem ser utilizadas: *tempofunc* e *tempocal*. Embora seja evidente que fisicamente estas variáveis são importantes no fenómeno de falha, a sua utilização é limitada por se possuir informação relativa a apenas duas bombas centrífugas. De facto, como ambas as bombas têm tempos de funcionamento similares e o mesmo tempo de calendário, nos dados mais recentes essas variáveis tomam valores superiores aos dos dados de treino, pelo que os modelos

[1] Ver Secção 7.4.

não são capazes de decidir efetivamente com base nelas. A utilização destas variáveis faz sentido quando são considerados dados de várias bombas com tempos de início de funcionamento diferentes.

Em seguida, verificou-se a correlação linear entre variáveis (Figura 8.1). Deve notar-se que esta metodologia é capaz de determinar apenas se uma variável justifica a variação linear de outra: o componente a_{ij} da matriz indica a correlação das variáveis x_i e x_j, ou seja, valores absolutos mais elevados indicam elevada correlação. Encontraram-se dois grupos de variáveis altamente correlacionadas (correlação superior a 0.8): 1) *TI* e *motor*; 2) *narranques*, *tempofuncmudemp*, *tempovidaemp*, *npress* e *nequedaspress*. Do primeiro grupo optou por manter-se *TI*, por conter mais do que apenas informação binária, enquanto que do segundo grupo optou por manter-se *tempovidaemp*, dado que a sua influência no estado de funcionamento dos empanques mecânicos é de fácil compreensão.

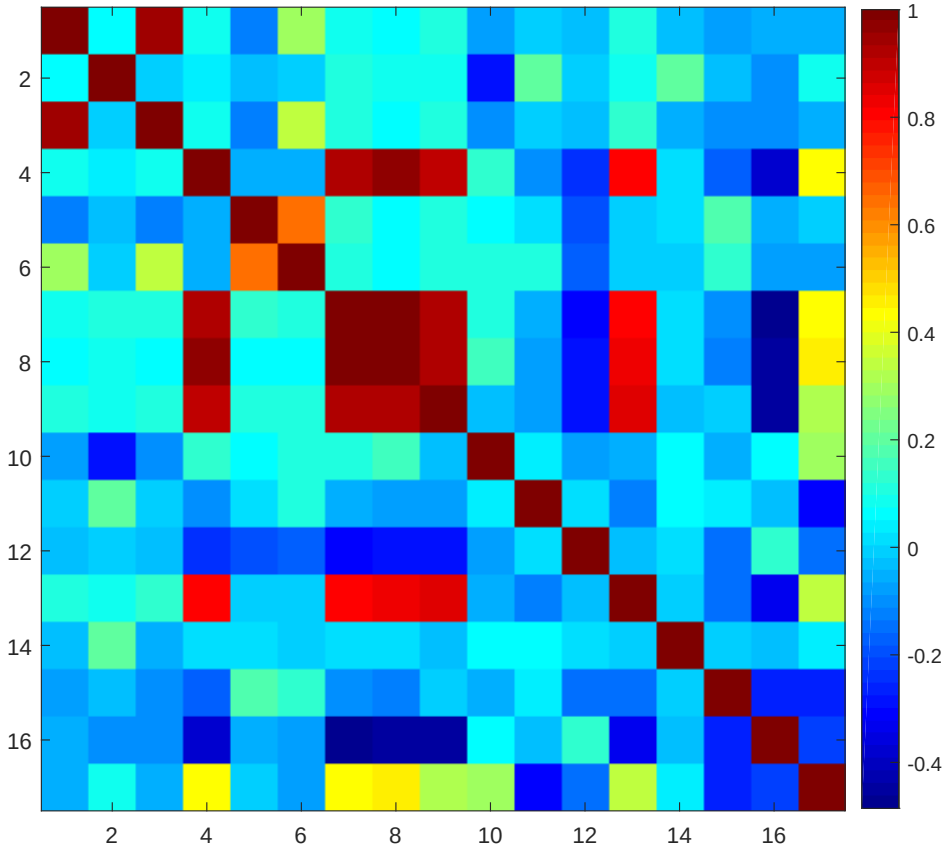


Figura 8.1: Correlação entre *features*.

Salienta-se que se desenvolveu um procedimento semi-automático de deteção de *features* linearmente correlacionadas onde o utilizador é questionado acerca da variável que pretende manter.

Assim, após uma primeira avaliação das *features* foi possível selecionar 12 variáveis que, se entende, contêm informação semelhante à contida pela lista inicial de 19 *features*. Antes de se proceder à aplicação de modelos de *Machine Learning*, é ainda necessário classificar os dados.

8.3 Classificação dos dados

Dado que se conhece o instante de falha de cada empanque mecânico[2] é possível classificar os dados com base no tempo até à falha. Um aspeto crucial é notar que todos os pontos usados nos modelos de *Machine Learning* são pontos de funcionamento normal, isto é, pontos onde ainda não se considera que tenha ocorrido falha. Desta forma, optou por dividir-se os dados em duas categorias de funcionamento:

1. Estável: pontos cuja distância temporal até ao instante de falha é superior a dois meses.
2. Pré-instável: pontos cuja distância temporal até ao instante de falha é menor ou igual a dois meses.

O tempo até à falha que delimita a fronteira entre as duas categorias foi escolhido por forma a permitir que a equipa de manutenção receba informação relativa a uma possível falha do equipamento atempadamente, podendo preparar conveniente a intervenção (e, eventualmente, adaptar a produção em função da possível indisponibilidade do equipamento). Ainda assim, sugere-se que em futuros projetos se verifique a influência deste parâmetro na *performance* dos modelos. Para facilitar a compreensão das métricas de avaliação aplicadas, apresentadas na Secção 7.5, considera-se que o principal objetivo dos modelos *Machine Learning* utilizados é determinar o instante a partir do qual faltam dois meses para a falha, pelo que, ao longo do presente capítulo, os pontos da categoria *Pré-instável* são por vezes classificados como *Positivo*.

8.4 Divisão dos dados

Restam apenas duas tarefas para que os dados estejam prontos a ser usados nos modelos de *Machine Learning*: filtrar os dados e dividi-los pelos três conjuntos (treino, desenvolvimento e teste).

Por filtrar os dados entende-se eliminar aqueles que não pertencem às categorias consideradas, têm pressão inferior à pressão mínima de funcionamento ou declive não razoável (eliminação de *outliers*).

A primeira eliminação é necessária porque além de pontos da categoria *Estável* e *Pré-instável* existem pontos de empanques mecânicos que não falharam e pontos posteriores aos instantes de falha. No presente projeto todos esses pontos foram descaracterizados. No futuro, para garantir um maior aproveitamento dos dados, devem usar-se os pontos dos empanques mecânicos que não falharam como pertencentes à categoria *Estável* (com exceção dos pontos dos últimos dois meses, que devem ser descaracterizados). As duas outras filtragens eliminaram um número reduzido de pontos e foram aplicadas apenas com o intuito de limitar a distância entre os valores mínimo e máximo dessas *features*.

Assim, de 1081376 pontos existentes à partida usaram-se 596432, o que corresponde a cerca de 55% dos pontos. Desses 55%, 36% corresponde a pontos classificados como *Pré-instável* e os restantes 64% a pontos classificados como *Estável*. Na Tabela 8.1 apresenta-se a divisão dos pontos utilizados por bomba e tipo de empanque mecânico.

[2] Ver Secção 6.3.

Tabela 8.1: Divisão dos dados usados nos modelos de *Machine Learning* por bomba e tipo de empanque mecânico

/%	Acop.	Livre	
A	23.2	38.8	61.9
B	24.0	14.1	38.1
	47.2	52.9	

8.4.1 Divisão em conjuntos

A divisão dos pontos pelos diversos conjuntos foi efetuada, tal como mencionado na Secção 7.5, em função das datas associadas aos pontos. Os pontos anteriores a 2016, os pontos de 2016 e os pontos de 2017 foram atribuídos aos conjuntos de treino, desenvolvimento e teste, respetivamente. Esta metodologia de divisão de dados, embora menos recorrente, é essencial para garantir que as avaliações obtidas correspondem à *performance* real que se deve esperar dos modelos.

Desta forma, apresenta-se na Tabela 8.2 a quantidade de dados que foi atribuída a cada um dos conjuntos e a percentagem que corresponde à categoria *Estável*. Note-se que 74% dos dados do conjunto de desenvolvimento (sobre o qual é calculada a maior parte das métricas de avaliação apresentadas no presente capítulo) correspondem à categoria *Estável*. Assim, um modelo tão simples (e sem qualquer utilidade, dado que admite que o funcionamento é sempre estável) como um modelo que prevê que um dado empanque mecânico está em funcionamento estável em todos os instantes tem uma *accuracy* de 74%.

Tabela 8.2: Quantidade de dados atribuída a cada um dos conjuntos e a percentagem correspondente à categoria *Estável*

	Quantidade /%	Estável /%
Treino	52.9	59.2
Dev.	23.1	74.0
Teste	24.1	64.8

8.5 Primeira aplicação de modelos

Como mencionado na Secção 7.3, sugere-se que antes de se selecionar e/ou extrair *features* de forma exaustiva se deve verificar como uma grande variedade de modelos se comporta com a primeira seleção de *features* efetuada.

Os tempos de treino e previsão, que são muito importantes numa fase mais avançada de desenvolvimento de modelos (porque limitam as combinações de variáveis e hiperparâmetros que podem ser testadas), não são características prevalecentes nesta etapa, visto que a afinação dos modelos deve ser mínima. Assim, de seguida treinam-se e avaliam-se modelos de treino e previsão rápidos e lentos.

8.5.1 Modelos de treino e previsão rápidos

Na Tabela 8.3 apresentam-se os resultados da aplicação de *decision trees* com diferentes números máximos de divisões permitidas (hiperparâmetro que procura evitar o *overfitting*). Uma conclusão é clara: o número máximo de divisões deve ser limitado sob o risco de o modelo determinado ser incapaz de generalizar. De facto, observa-se que para um máximo de 50 divisões o modelo é capaz de prever corretamente todos os pontos do conjunto de treino, mas tem uma *performance* sobre o conjunto de desenvolvimento inferior à de uma *decision tree* capaz de prever com sucesso apenas 89% dos dados de treino (o que demonstra claramente a existência de *overfitting*). Note-se ainda que todos os modelos têm *accuracy* inferior à do modelo que prevê todos os dados como *Estável*.

Tabela 8.3: *Accuracy* e F_1 *score* calculados para os conjuntos de treino e desenvolvimento de *decision trees* com diferentes números máximos de divisões permitidas

Max. Num.	Conj. dev.		Conj. treino	
Divisões	*acc* /%	F_1 *score* /%	*acc* /%	F_1 *score* /%
10	66.8	56.4	89.3	87.1
20	66.7	51.9	94.2	92.4
30	66.0	54.4	98.6	98.3
40	66.7	55.1	99.9	99.8
50	66.7	55.1	100.0	100.0

De igual forma, apresentam-se na Tabela 8.4 os resultados para *linear discriminant analysis* (LDA) e *naive Bayes*. Note-se que apesar destes modelos revelarem, à semelhança das *decision trees*, baixas *performances*, o problema do *overfitting* não se verificou. De facto, a *performance* dos modelos é similar nos dois conjuntos. Note-se ainda que embora a *accuracy* dos modelos continue inferior à percentagem de dados da categoria *Estável*, a utilidade dos modelos apresentados é superior porque estes são capazes, tal como demonstra o F_1 *score*, de distinguir entre categorias.

Tabela 8.4: *Accuracy* e F_1 *score* calculados para os conjuntos de treino e desenvolvimento de modelos LDA e *naive Bayes*

/%	Conj. dev.		Conj. treino	
	acc	F_1 *score*	*acc*	F_1 *score*
LDA	71.8	54.1	76.7	72.5
naive Bayes	73.3	61.1	75.7	72.7

8.5.2 Modelos de treino e previsão lentos

Nesta primeira fase obtiveram-se ainda modelos kNN e SVM. Para estes algoritmos, dado os seus elevados tempos de treino (SVM) e previsão (kNN e SVM), determinou-se a *performance* apenas no conjunto de desenvolvimento. Além dos resultados para as 12 *features*, apresentam-se, na Tabela 8.5, os resultados da aplicação dos algoritmos usando as *features* que o método de seleção de *features* NCA[3] indica como sendo as mais relevan-

[3] Ver Secção 7.4.

tes. A exploração destes modelos não foi intensiva, pretendendo-se apenas demonstrar o seu potencial.

Tabela 8.5: *Accuracy* e F_1 *score* calculados para os conjuntos de desenvolvimento de modelos kNN e SVM treinados com as 12 *features* iniciais e com as *features* selecionadas pelo NCA

/%	Sem NCA		Com NCA	
	acc	F_1 *score*	*acc*	F_1 *score*
kNN (k=100)	65.5	43.2	58.6	42.9
SVM (linear)	40.7	35.4	42.3	41.3
SVM (quad.)	74.0	0.0	76.4	47.5
SVM[1]	74.0	0.0	74.0	0

[1] Resultados idênticos para *kernels* gaussiano e polinomial de terceiro grau.

A Tabela 8.5 demonstra que, para as 12 *features* iniciais, os modelos SVM (com exceção do linear) prevêem que todos os dados são da categoria *Estável*. O comportamento mantém-se, com exceção do quadrático, com as *features* selecionadas pelo NCA. Quanto ao SVM quadrático, é relevante notar que a redução de *features* melhora a sua *performance*, sendo o modelo capaz de distinguir entre categorias no último caso. O SVM linear apresenta uma *performance* inaceitável em ambos as situações. Já o kNN vê a sua *performance* degradada com a diminuição do número de *features*, podendo concluir-se que, neste caso, o NCA não seleciona as *features* mais relevantes para este modelo.

Apesar de nenhum dos modelos apresentado até ao momento apresentar uma *performance* elevada, é necessário relembrar que, com exceção de parte dos resultados apresentados na Tabela 8.5, foi usada uma elevada quantidade de *features* na aplicação dos modelos. Assim, é agora necessário executar tarefas de extração e seleção de *features*.

8.6 Procura das *features* mais relevantes

No presente projeto não se efetuaram as tarefas de *feature engineering* de forma aprofundada. De facto, e atendendo ao número de *features* relativamente reduzido, optou apenas por procurar-se, a título demonstrativo e de forma exaustiva, para os algoritmos com treino e previsão rápidos, a combinação de *features* (até um máximo de 6) que conduz às melhores *performances* no conjunto de desenvolvimento. Os melhores resultados obtidos para cada modelo e número de *features* são apresentados na Figura 8.2.

Ao contrário do que sucedeu na aplicação de modelos com 12 *features*, a redução da dimensionalidade conduziu a *performances* muito razoáveis (o melhor modelo testado teve uma *accuracy* de 87%). A Figura 8.2 demonstra também que o algoritmo *naive Bayes* é o que conduz à obtenção de melhores resultados. É interessante notar que na manutenção tradicional a análise bayesiana é muito utilizada, podendo esse aspeto facilitar a introdução deste algoritmo no meio. Note-se também que os modelos de maior *accuracy* não são necessariamente os modelos de maior F_1 *score*. Para *decision trees* e LDA conseguiram-se *accuracies* máximas de cerca de 80%.

Na Tabela 8.6 apresentam-se as *features* que permitiram obter os modelos com melhor

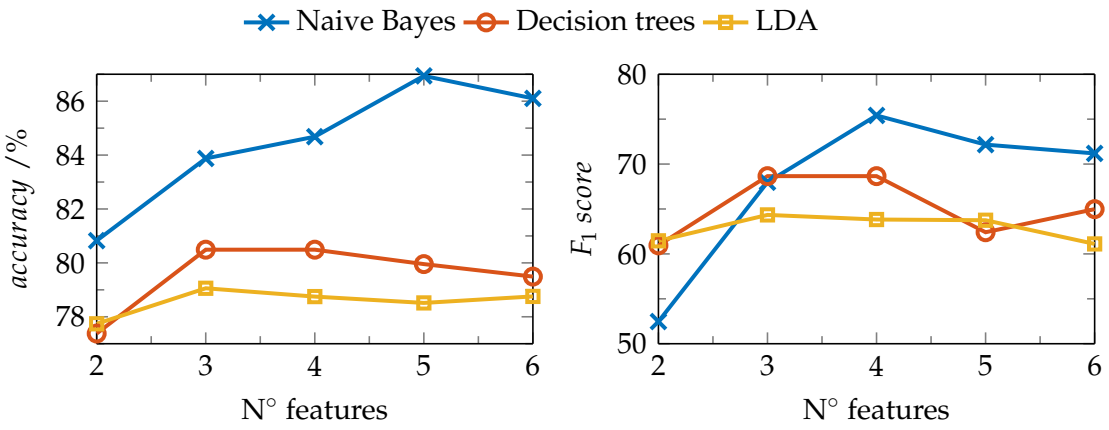


Figura 8.2: Melhores *performances* obtidas para diferentes modelos combinando exaustivamente um diferente número de *features*.

performance para os diferentes algoritmos.

Tabela 8.6: *Features* utilizadas pelos modelos com melhor *accuracy*

Naive Bayes	***Decision tree***	**LDA**
tipoemp	*tipoemp*	*tipoemp*
tempoultarranque	*tempoultparagem*	*aumentopressao*
pressaoapospress	*TI*	*PI*
tempoultsubsoposto	-	-
nobomba	-	-

Note-se que apenas uma das *features* (*tipoemp*) foi utilizada em simultâneo pelos melhores modelos dos três algoritmos testados, o que demonstra que as *features* mais relevantes para um dado algoritmo não são, necessariamente, as mais relevantes para o outro (pode ter implicações na escolha dos métodos de *feature extraction* e *selection*). O facto de *tipoemp* ser uma *feature* relevante para os três algoritmos pode sustentar a hipótese de a falha diferir entre tipos de empanques mecânicos. Ainda assim, os resultados devem ser analisados com cautela, até porque, como se verá, esta *feature* não é uma das consideradas mais relevantes pelo NCA. Deve também notar-se que nas *features* selecionadas pelos diferentes modelos existe informação acerca dos arranques, das pressurizações e das intervenções efetuadas.

8.6.1 Representação no tempo das previsões do melhor modelo

Um modelo que possa ser implementado na prática deve ser capaz de detetar a mudança de estado real de *Estável* para *Pré-instável* e, mais do que isso, manter as previsões de "pré-instabilidade" no tempo. Assim, na Figura 8.3 apresenta-se um exemplo demonstrativo da forma como as previsões variam no tempo: representa-se a evolução do estado de um dado empanque mecânico, através da contagem cumulativa do número de pontos da categoria *Estável*, e o comportamento do modelo *naive Bayes* com melhor *performance*, através da contagem cumulativa das previsões *Estável*.

Observa-se na Figura 8.3 que ao longo do tempo de vida do empanque mecânico, o mo-

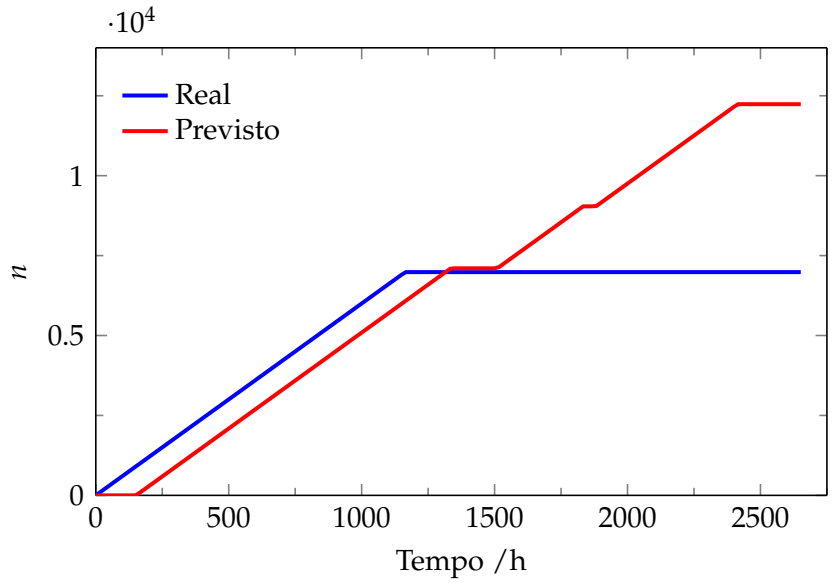


Figura 8.3: Exemplo da evolução do estado de um empanque mecânico e das previsões efetuadas pelo modelo *naive Bayes* com melhor *accuracy*.

delo faz previsões de *Pré-instável* em quatro ocasiões distintas (patamares). A de duração mais demorada é, efetivamente, a que antecede a falha. Ainda assim, não seria possível perceber que patamar corresponderia, realmente, à aproximação da falha. Noutros casos observaram-se situações onde o modelo é capaz de prever antecipadamente a falha (Figura 8.4a) e onde nem sequer se apercebe do início da instabilidade (Figura 8.4b). Entende-se que este tipo de representação e a duração dos patamares de previsão *Pré-instável* podem ser o ponto de partida para o desenvolvimento de modelos de *performance* satisfatória.

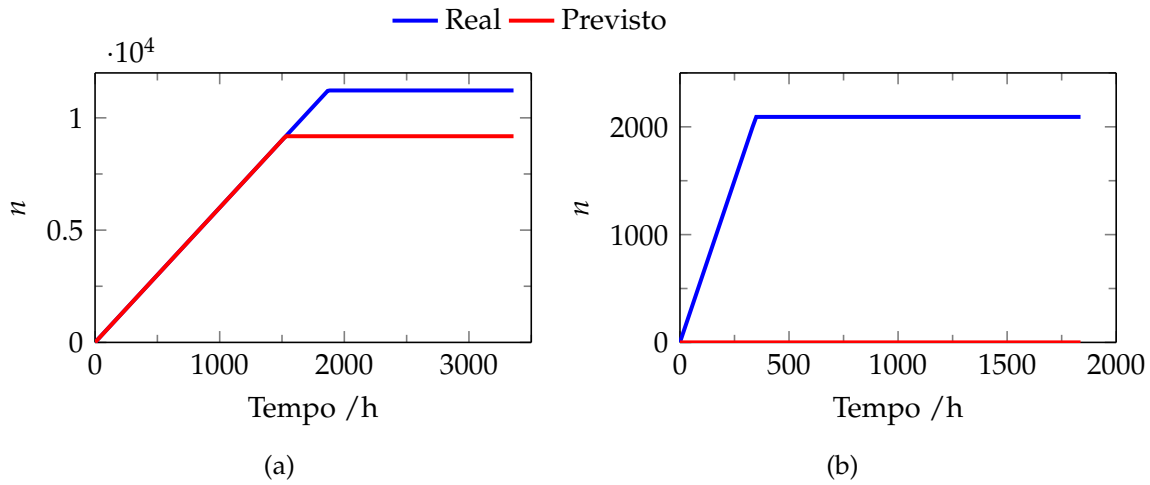


Figura 8.4: Exemplos da evolução do estado de um empanque mecânico e previsões efetuadas pelo modelo *naive Bayes* com melhor *accuracy*.

8.6.2 Comportamento no conjunto de teste

Identificado o modelo com *performance* mais aceitável (*naive Bayes* com 5 *features*), importa fazer a sua avaliação com os dados do conjunto de teste, visto que as avaliações efetuadas até ao momento basearam-se no conjunto de desenvolvimento e foram usadas na otimização de hiperparâmetros (número máximo e combinação de *features*). Apesar de, num caso real, ser aconselhado que se avalie apenas o modelo que se pretende implementar, na Tabela 8.7 apresentam-se as métricas de avaliação para os melhores modelos *naive Bayes* para diferente número de *features*.

Tabela 8.7: *Performance* dos melhores modelos *naive Bayes* com diferentes *features* nos conjuntos de desenvolvimento e teste

/%		Nº *features*				
		2	3	4	5	6
Conj. dev.	*acc*	80.8	83.9	84.7	86.9	86.1
	F_1 *score*	52.5	68.0	75.4	72.2	71.2
Conj. teste	*acc*	67.9	66.1	50.9	56.5	58.9
	F_1 *score*	53.9	60.1	59.3	56.1	57.5

Os resultados obtidos na previsão dos dados do conjunto de teste são muito piores do que a *performance* no conjunto de desenvolvimento perspetivava. De facto, o modelo com maior *accuracy* no conjunto de desenvolvimento tem uma *accuracy* de apenas 56.4% no conjunto de teste. Isto demonstra a necessidade de dividir os dados em três conjuntos. Relembre-se que, para o conjunto de teste, 64.8% são dados da categoria *Estável*, pelo que se observa que o modelo de 5 *features* apresenta *accuracy* inferior a um modelo que prevê todos os pontos como *Estável*. Ainda assim, tal não significa que este último modelo seja melhor. De facto, apesar de muito limitado o modelo de 5 *features* é capaz de identificar 78.9% dos instantes em que a classificação é *Pré-instável* (Tabela 8.8). O problema é que o número de falsos positivos é muito elevado (baixa *precision*).

Tabela 8.8: Matriz de confusão para o modelo *naive Bayes* com 5 *features*

/%		*Previsão*	
		Positivo	Negativo
Real	Positivo	78.91	21.09
	Negativo	55.65	44.35

Conclui-se assim que o melhor modelo encontrado não tem *performance* suficiente para que possa ser implementado com sucesso.

8.7 Seleção de *features* usando NCA

Na Secção 8.5 apresentaram-se resultados de modelos de treino e previsão lentos aplicados usando *features* selecionadas com recurso ao método NCA. Na presente secção apresentam-se as *features* selecionadas por esse método e a respetiva influência de cada uma na classificação.

Na Figura 8.5 apresenta-se o peso que o método atribui a cada *feature* (não importa tanto os valores absolutos dos pesos, mas antes a comparação entre os pesos de diferentes *features*). Pode observar-se que 5 *features* têm maior destaque. São elas: *TI*, *PI*, *tempoultparagem*, *tempoultpress* e *declives*. Estas foram as *features* usadas na Secção 8.5.

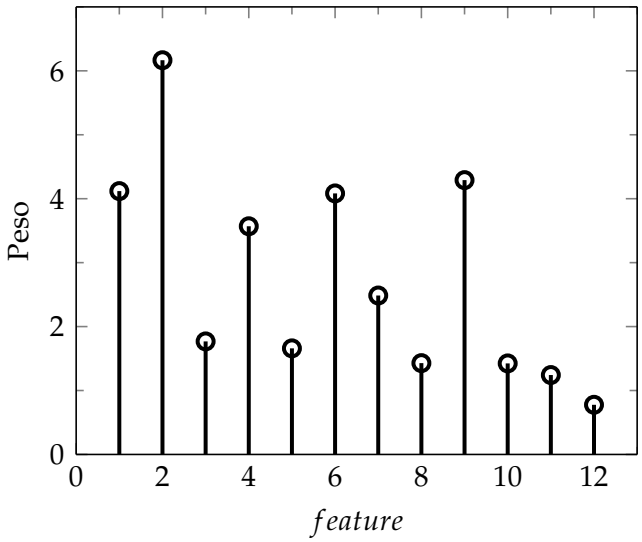


Figura 8.5: Pesos obtidos com a aplicação do método NCA.

Do ponto de vista físico, quatro das cinco variáveis destacadas pelo método são relevantes: *TI* e *PI* porque são os únicos sinais de variáveis físicas que se conhecem; *tempoultpress* e *declives* porque é esperado que o tempo entre pressurizações diminua e o declive aumente à medida que a falha se aproxima. A importância da variável *tempoultparagem* poderá, eventualmente, relacionar-se com os arranques e paragens da bomba e, portanto, ter também um papel importante na falha. No entanto, outras variáveis contêm informação sobre os arranques e paragens e não foram selecionadas pelo método.

Ainda, tendo presentes os resultados do método e atentando na Tabela 8.6, é possível concluir que as *features* mais relevantes variam muito em função do método de seleção utilizado. Assim, considera-se muito importante aprofundar, em futuros projetos, as metodologias de seleção e/ou extração de *features* (antes ainda de uma exploração mais cuidada dos algoritmos de *Machine Learning*).

8.8 Outros estudos

8.8.1 Limitação do tempo de vida

A análise da Figura 6.1 permitiu concluir que mais de 80% dos empanques mecânicos têm um tempo de vida inferior a 150 dias. Assim, tal como se demonstra na Figura 8.6, a maior parte dos pontos classificados como *Pré-instável* (Positivo) estão associados a tempos de vida dos empanques mecânicos reduzidos. Em contrapartida, nos empanques mecânicos com tempos de vida mais elevados, o estado *Pré-instável* está associado a tempos de vida dos empanques mecânicos elevados. Entre os dois extremos existem poucos (ou nenhuns) pontos da categoria *Pré-instável*. Esta observação levou a que se ponderasse limitar o tempo de vida dos dados introduzidos no modelo, por forma a evitar a possível influência negativa que os *outliers* do tempo de vida dos empanques mecânicos possam ter.

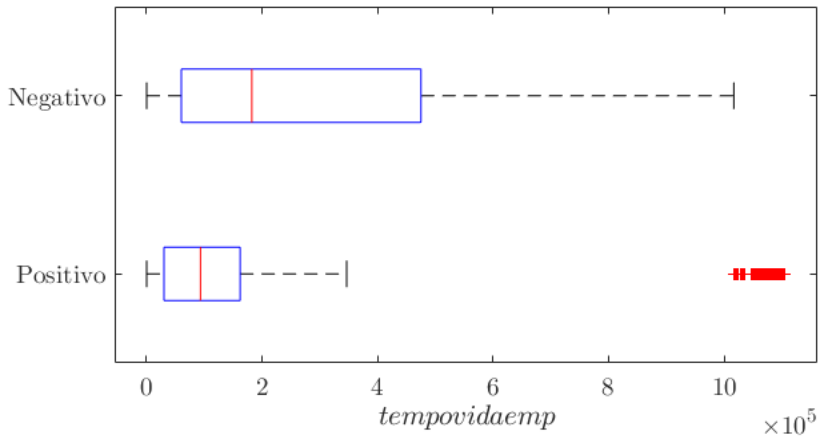


Figura 8.6: Distribuição dos pontos em função do tempo de vida do empanque mecânico e da classe.

Assim, treinaram-se modelos (com a mesma abordagem apresentada nas secções anteriores) usando apenas os dados associados a tempos de vida inferiores a 150 dias. A desvantagem desta abordagem é que estes modelos permitem fazer previsões apenas enquanto um dado empanque mecânico estiver montado há menos de 150 dias. Os resultados obtidos foram piores que os apresentados anteriormente. Uma justificação pode ser o facto de as falhas no início de vida terem um carácter mais aleatório. Acresce a isso que se diminuiu a quantidade de dados utilizados (a *performance* dos modelos de *Machine Learning* está intimamente ligada à quantidade de dados[4]).

8.8.2 Possível efeito de rodagem

A redução da dimensionalidade, além de melhorar a *performance* dos modelos, facilita a visualização gráfica dos resultados. Desta forma, procuraram-se modelos com combinações de duas variáveis que tivessem *performance* razoável. Verificou-se que um modelo *naive Bayes* baseado nas *features tempovidaemp* e *pressaoapospress* tem uma *accuracy* de cerca de 80% (*precision* $=$ 71% e *recall* $=$ 35%), ou seja, quando prevê que um conjunto de dados é da categoria *Pré-instável* acerta em cerca de 71% dos casos, deixando escapar, ainda assim, muitas das situações de pré-instabilidade. A fronteira de decisão do modelo é apresentada na Figura 8.7.

Embora as conclusões retiradas da Figura 8.7 devam ser olhadas com desconfiança, é interessante, antes de mais, atentar na forma como o modelo faz previsões (se um dado exemplo estiver fora da zona a verde é considerado *Pré-instável*). Mais interessante ainda é notar o modo como a zona verde expande ao longo do tempo. Tal expansão pode resultar de um efeito de rodagem, isto é, os componentes do empanque mecânico podem acomodar-se durante o funcionamento e, portanto, o empanque mecânico torna-se menos sensível ao valor da pressão após pressurização. Ainda assim, o intervalo de valores de pressurização “permitidos" para empanques mecânicos com tempos de vida reduzidos parece demasiado estreito.

[4] Ver Secção 7.6.

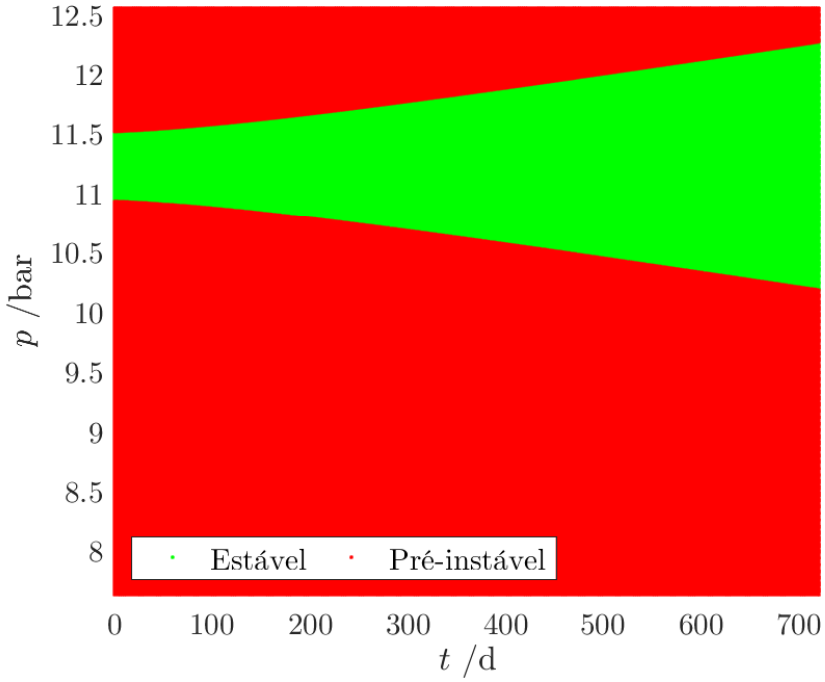


Figura 8.7: Fronteira de decisão de um modelo *naive Bayes* baseado nas *features* *tempovidaemp* e *pressaoapospress*.

8.9 Conclusões

No presente capítulo foram aplicados modelos de *Machine Learning* aos dados obtidos e tratados durante o projeto. Apesar de se ter obtido um modelo (*Naive Bayes* com 5 *features*) com *performance* muito satisfatória ($acc = 87\%$ quando avaliado no conjunto de desenvolvimento), não foi possível encontrar um modelo para implementar na prática porque a *performance* degradou-se seriamente quando a avaliação foi efetuada no conjunto de teste ($acc = 57\%$).

Foi demonstrado como se pode efetuar uma primeira seleção de *features* e como deve ser efetuada a classificação e divisão dos dados. Demonstrou-se ainda a importância do *feature engineering*: combinações de diferentes *features* conduzem a modelos com *performances* muito distintas. Além da seleção de *features* de forma exaustiva, determinando-se as mais relevantes através da avaliação de todas as combinações possíveis, procedeu-se à seleção de *features* pelo método NCA. Concluiu-se que diferentes métodos conduzem à seleção de diferentes *features*.

Por fim, avaliou-se o impacto de limitar o tempo de vida dos empanques mecânicos (obtêm-se *performances* inferiores) e demonstrou-se que as ferramentas de *Machine Learning* podem ser usadas para auxiliar a compreensão de fenómenos físicos.

9. Conclusões e trabalhos futuros

9.1 Conclusões

O presente projeto permitiu perceber de que modo o *Machine Learning* pode ser aplicado na manutenção e, mais importante, definir uma metodologia que pode ser seguida em projetos futuros. Demonstrou-se que o *Machine Learning* tem enorme potencial e deve ser usado nas "indústrias do futuro," sob pena de perda de competitividade. Ainda assim, não se deve esperar que o *Machine Learning* resolva todos os problemas da manutenção. Espera-se que a necessidade de um engenheiro mecânico conhecer estas ferramentas (ainda que do ponto de vista de utilizador) tenha sido evidenciada.

No que concerne aos registos de falha, foi possível concluir que embora os registos SAP estejam, por vezes, incompletos e apresentem alguns erros, a combinação desses registos com os dados obtidos em contínuo permite obter informação de elevada qualidade sobre as falhas (nomeadamente dos empanques mecânicos). Os pontos em falta não limitam a utilização dos dados obtidos em contínuo porque podem ser facilmente reconstruidos.

Demonstrou-se que a incorporação de informação obtida a partir dos dados em contínuo é uma mais valia para a análise fiabilística, dado que permite uma identificação mais assertiva dos equipamentos críticos (nomeadamente porque agrava a fiabilidade dos componentes que funcionam durante menos tempo). Os tempos de indisponibilidade podem também ser calculados de forma mais correta. Por sua vez, a informação acerca das pressurizações pode ser usada, por exemplo, para avaliar os contratos com empresas subcontratadas e, eventualmente, adaptar os procedimentos de manutenção.

Concluiu-se também que é crucial definir quantitativamente as falhas dos empanques mecânicos. Em primeiro lugar, porque só assim é possível identificar o problema de forma efetiva. Em segundo lugar, porque os modelos de *Machine Learning* requerem um estabelecimento claro e rigoroso do objetivo a que se propõem. Assim, se o objetivo dos sistemas de *Machine Learning* for determinar instantes de falha, a correta definição de falha é fundamental para o sucesso do projeto. Demonstrou-se ainda que a falha de um empanque mecânico pode ser identificada a partir do declive da pressão, mas que a forma como foi definida subestima o tempo de vida dos empanques mecânicos e obriga à utilização do conceito de *empanque virtual*.

Relativamente à aplicação de modelos de *Machine Learning* concluiu-se que, mais importantes que os modelos, são os dados com que estes são alimentados. O *feature engineering* é, de facto, a etapa mais importante de todo o projeto, sendo que as *features* usadas condicionam profundamente os resultados obtidos. A seleção do modelo mais adequado à resolução de um dado problema tem de ser feita através de uma metodologia de tentativa-

erro.

Concluiu-se ainda que a divisão dos dados em conjuntos de treino e teste é fundamental para que a *performance* real dos sistemas de *Machine Learning* não seja inferior às expectativas geradas por uma avaliação incorreta dos resultados dos modelos.

Demonstrou-se também que o *overfitting* é o maior obstáculo à aplicação de modelos de *Machine Learning*, pois as *performances* obtidas na previsão dos dados dos subconjuntos de treino e desenvolvimento (durante a afinação dos hiperparâmetros) são, tipicamente, muito superiores às *performance*s no conjunto de teste. Embora existam estratégias para atenuar o *overfitting*, é necessário afinar o modelo de modo a que, por um lado, não sofra de *overfitting* e, por outro, seja capaz de detetar os padrões e estruturas dos dados de modo a ser capaz de fazer previsões assertivas. O papel da maldição da dimensionalidade foi também destacado.

Algumas considerações foram tecidas acerca da falta de transparência do processo de decisão inerente a muitos dos modelos de *Machine Learning*. Concluiu-se que o uso desses modelos é inevitável e, portanto, o desconforto que possa advir desse aspeto tem de ser ultrapassado com, por exemplo, o uso de métricas de avaliação poderosas e demonstrações práticas de sucesso preditivo. O desconhecimento do processo de decisão não deve justificar o uso dos algoritmos sem compreensão do seu princípio de funcionamento (que é, aliás, fundamental aquando da afinação dos hiperparâmetros).

Importa ainda mencionar que os modelos *naive Bayes* foram os que apresentaram melhores *performances*. A larga utilização da análise bayesiana na manutenção poderá facilitar a utilização extensiva deste modelo no meio.

Demonstrou-se ainda que a resolução de um dado problema com recurso a *Machine Learning* é muito específica até à etapa de classificação dos dados, isto é, é muito dependente do tipo de problema que se pretende resolver. Dessa etapa em diante, é quase indiferente se o problema é de deteção de instantes de falha ou de filtragem de *spam*: a metodologia a aplicar é idêntica e os algoritmos de *Machine Learning* os mesmos. Assim, o eventual recurso a uma empresa externa que ofereça soluções gerais de *Machine Learning* deve ser visto com cautela porque se as etapas iniciais (específicas) não forem efetuadas, o risco de se implementar um sistema de baixa *performance* é elevado.

Por fim, salienta-se que o Matlab se apresenta como uma ferramenta extraordinária para a aplicação de todos os conceitos apresentados ao longo da dissertação. A facilidade com que permite manipular vetores e matrizes é uma mais valia durante o processamento dos dados. Ainda assim, entende-se que as ferramentas de *Machine Learning* ainda não estão suficientemente maduras no *software*. A linguagem de programação Python é uma alternativa a considerar na parte final do projeto (quando se pretende treinar e avaliar os modelos). A sugestão desta linguagem de programação prende-se com as numerosas bibliotecas disponíveis e a extensa bibliografia que demonstra como estas podem ser aplicadas em projetos de *Machine Learning*. Esta é, por exemplo, a linguagem de programação preferencial usada em competições de *Machine Learning* como as que decorrem na comunidade Kaggle [75].

9.2 Trabalhos futuros

No que concerne ao acesso aos dados, sugere-se o desenvolvimento de ferramentas que facilitem a sua exportação para ficheiros de onde possam ser facilmente importados para

os *softwares* de trabalho. A par do desenvolvimento dessas ferramentas, sugere-se a implementação das ferramentas desenvolvidas para a obtenção de informação a partir da combinação de dados de diferentes origens porque, como foi demonstrado, são uma mais valia para a análise fiabilística.

Sugere-se também a revisão da definição quantitativa de falha e a avaliação do seu impacto nos tempos de vida calculados. Entende-se que, mais do que encontrar um declive crítico mais adequado que o utilizado, é necessário considerar outros aspetos. A nova definição de falha deve ser tal que não seja necessário recorrer ao conceito de *empanque virtual*. O estudo físico dos empanques mecânicos poderá ser uma mais valia.

Para avaliar objetivamente a importância dos dados, sugere-se que sejam considerados dados de outras bombas centrífugas (com a mesma tecnologia e condições de operação semelhantes). Espera-se que as *performances* dos modelos tenham uma tendência similar às apresentadas na Figura 7.16, ou seja, melhorem com o aumento da quantidade de dados. Salienta-se que o aumento do número de bombas requer considerações adicionais no estabelecimento de *features*, nomeadamente transformar cada bomba introduzida numa variável binária, dado que a solução atual (considerar o número da bomba como *feature*) não deve ser utilizada quando o número de equipamentos é superior a dois.

Como se mencionou recorrentemente, o *feature engineering* é essencial para o sucesso da aplicação das ferramentas de *Machine Learning*. Assim, sugere-se que uma exploração profunda das técnicas de *feature engineering* seja efetuada, por forma a que diferentes métodos sejam comparados e o mais adequado (se é que existe) para a resolução de problemas idênticos ao proposto seja identificado. Entende-se que além de se selecionar *features*, se deve também combinar *features* de modo a que a informação perdida seja menor (em relação à perdida com a seleção de algumas *features* em detrimento de outras).

Entende-se também que é necessário ganhar uma maior perceção do impacto dos hiperparâmetros nos resultados, assim como explorar outros algoritmos. Considera-se relevante considerar o tempo de fronteira entre classes como hiperparâmetro, ou seja, não definir à partida qual deve ser a antecedência para a qual se pretende que o modelo seja capaz de fazer previsões. Depois destes objetivos serem atingidos, fará sentido considerar técnicas de regressão.

Um aspeto relevante que é demonstrado pela Figura 8.3 é a inexistência de acoplamento entre previsões consecutivas, isto é, o modelo pode voltar a prever *Estável* após ter previsto *Pré-instável*. Por forma a evitar este tipo de incoerências, poderá ser relevante procurar acoplar previsões consecutivas e, eventualmente, desenvolver uma nova métrica de avaliação (a otimizar) baseada nesse acoplamento.

A exploração de modelos mais complexos (redes neuronais e *deep learning*) e outras modos de aprendizagem deverá também ser considerada.

Por fim, sugere-se a aplicação da metodologia apresentada a outros equipamentos da refinaria, possivelmente com comportamentos mais facilmente explicáveis do que os empanques mecânicos, de modo a que a metodologia apresentada possa ser validada.

Referências

[1] Sandy Dunn. Big Data, Predictive Analytics and Maintenance, 2018. URL https://www.assetivity.com.au/article/maintenance-management/big-data-predictive-analytics-and-maintenance.html. Último acesso a: 2018-09-01.

[2] R.K. Mobley. *An Introduction to Predictive Maintenance*. Plant engineering series. Van Nostrand Reinhold, 1990. ISBN 9780442318284.

[3] Luís Ferreira. Desenvolvimento de modelos de previsão de falhas em equipamentos críticos. Proposta de projeto, 2018.

[4] P. Domingos. *The Master Algorithm: How the Quest for the Ultimate Learning Machine Will Remake Our World*. Penguin Books Limited, 2015. ISBN 9780241004555. URL https://books.google.pt/books?id=pjRkCQAAQBAJ.

[5] Patrick Janhke. Machine learning approaches for failure type detection and predictive maintenance. Master's thesis, Technische Universitat Darmstadt, 2015.

[6] P. Pina. Análise fiabilística de um sistema instrumentado de variáveis complexas na refinaria de Matosinhos. Master's thesis, Faculdade de Engenharia da Universidade do Porto, 2018.

[7] Galp. Nota interna da Galp. 2018.

[8] SAP. SAP Software Solutions, 2018. URL https://www.sap.com/index.html. Último acesso a: 2018-08-27.

[9] Wikipedia. SAP SE, 2018. URL https://en.wikipedia.org/wiki/SAP_SE. Último acesso a: 2018-08-27.

[10] J. Pinto. Análise fiabilística de um compressor alternativo na refinaria de Matosinhos. Master's thesis, Faculdade de Engenharia da Universidade do Porto, 2016.

[11] Jay Lee, Behrad Bagheri, and Hung-An Kao. A cyber-physical systems architecture for industry 4.0-based manufacturing systems. *Manufacturing Letters*, 3:18 – 23, 2015. ISSN 2213-8463. doi: https://doi.org/10.1016/j.mfglet.2014.12.001.

[12] Wikipedia. Industry 4.0, 2018. URL https://en.wikipedia.org/wiki/Industry_4.0. Último acesso a: 2018-08-27.

[13] Techopedia. What is a Macro?, 2018. URL https://www.techopedia.com/definition/3833/macro. Último acesso a: 2018-08-27.

[14] NP EN 113306:2007. Terminologia da manutenção, 2007.

[15] ISO 14224:2016. Petroleum, petrochemical and natural gas industries - collection and exchange of reliability and maintenance data for equipment, 2016.

[16] P.A. Tobias and D. Trindade. *Applied Reliability, Third Edition*. CRC Press, 2011. ISBN 9781439897249.

[17] P. O'Connor. *Practical Reliability Engineering*. Wiley, 2002. ISBN 9780470844632.

[18] Frank Rotello. Mechanical seal fundamentals, 2014. URL https://www.youtube.com/watch?v=GioZjBOcgWU. Último acesso a: 2018-09-01.

[19] M. Volk. *Pump Characteristics and Applications, Second Edition*. MECHANICAL ENGINEERING. Taylor & Francis, 2005. ISBN 9780824727550. URL https://books.google.pt/books?id=29XYpypoRywC.

[20] AESSEAL. What is a mechanical seal?, 2018. URL https://www.aesseal.com/en/resources/academy/what-is-a-mechanical-seal. Último acesso a: 2018-09-01.

[21] AESSEAL. Why use mechanical seals?, 2018. URL https://www.aesseal.com/en/resources/academy/why-use-a-mechanical-seal. Último acesso a: 2018-09-01.

[22] B. S. Nau. Mechanical seal face materials. *Proceedings of the Institution of Mechanical Engineers, Part J: Journal of Engineering Tribology*, 211(3):165–183, 1997. doi: 10.1243/1350650971542408.

[23] Vern Frahm. Mechanical seal fundamentals, 2014. URL https://blog.craneengineering.net/6-reasons-why-mechanical-seals-fail?hs_amp=true. Último acesso a: 2018-09-01.

[24] WSC Mechanical Seals. Mechanical seal stationary & rotary seal face materials, 2018. URL http://www.wscmechanicalseals.com/Stationary---Rotary-Face-Seal-Materials.html. Último acesso a: 2018-09-01.

[25] ANSI/API 682. Pumps - shaft sealing systems for centrifugal and rotary pumps, 2004.

[26] flowserve. Mechanical seal piping plans, 2016. URL https://www.flowserve.com/sites/default/files/2016-07/FTA157_Piping_Plan_Poster.pdf. Último acesso a: 2018-09-01.

[27] Luís Ferreira. Introdução à Metodologia RCM, 2010. Acção de formação.

[28] Tom Arnold and Chris Fone, editors. *Mechanical Seal Performance and Related Calculations*, 2010. Turbomachinery Laboratory, Texas A&M University.

[29] G. Kejela, R. M. Esteves, and C. Rong. Predictive analytics of sensor data using distributed machine learning techniques. In *2014 IEEE 6th International Conference on Cloud Computing Technology and Science*, pages 626–631, Dec 2014. doi: 10.1109/CloudCom.2014.44.

[30] Mathworks. Reconstructing missing data, 2018. URL https://www.mathworks.com/help/signal/examples/reconstructing-missing-data.html. Matlab *Help*. Último acesso a 2018-09-01.

[31] S.J. Orfanidis. *Optimum Signal Processing*. Sophocles Orfanidis, 2007. ISBN 9780979371318.

[32] Maria Carvalho. Sinais e Sistemas, 2007. Slides da unidade curricular *Sinais e Sistemas* do MIEIC. Faculdade de Engenharia da Universidade do Porto.

[33] Fernando Almeida. Transformadas de Laplace e Função de Tranferência, 2014. Slides da unidade curricular *Sistemas de Controlo* do MIEM. Faculdade de Engenharia da Universidade do Porto.

[34] Victor Barroso. Sinais Aleatórios em Tempo Contínuo. Parte II: Modelos de Fontes de Informação e de Ruído., 1999. Textos de apoio à unidade curricular *Fundamentos de Telecomunicações* da licenciatura em Engenharia Electrotécnica e de Computadores. Instituto Superior Técnico.

[35] Mathworks. Fill gaps using autoregressive modeling - Matlab fillgaps, 2018. URL https://www.mathworks.com/help/signal/ref/fillgaps.html. Matlab *Help*. Último acesso a: 2018-09-01.

[36] Hirotugu Akaike. Fitting autoregressive models for prediction. *Annals of the Institute of Statistical Mathematics*, 21:243–247, 12 1969.

[37] Mathworks. Find abrupt changes in signal - MATLAB findchangepts, 2018. URL https://www.mathworks.com/help/signal/ref/findchangepts.html. Matlab *Help*. Último acesso a: 2018-09-01.

[38] Marc Lavielle. Using penalized contrasts for the change-point problem. *Signal Processing*, 85(8):1501 – 1510, 2005. ISSN 0165-1684. doi: https://doi.org/10.1016/j.sigpro.2005.01.012.

[39] Rebecca Killick, Paul Fearnhead, and I.A. Eckley. Optimal detection of changepoints with a linear computational cost. *Journal of the American Statistical Association*, 107: 1590–1598, 12 2012.

[40] Greg Dionne. What does MinThreshold parameter explain in FindchangePts function (in matlab2017R)?, 2017. URL https://www.mathworks.com/matlabcentral/answers/351954-what-does-minthreshold-parameter-explain-in-findchangepts-function-in-matlab2017r. Último acesso a: 2018-09-01.

[41] Mathworks. Root-mean-square level - MATLAB rms, 2018. URL https://www.mathworks.com/help/signal/ref/rms.html. Matlab *Help*. Último acesso a: 2018-09-01.

[42] Mathworks. Variance - MATLAB var, 2018. URL https://www.mathworks.com/help/matlab/ref/var.html. Matlab *Help*. Último acesso a: 2018-09-01.

[43] Maria Raquel Valença. *Análise Numérica*. Universidade Aberta, 1996. ISBN 9789726741954.

[44] RELIABILITYWEB. Mechanical Seal Flush API Plan 53B: What Can Plant Operators Do to Help?, 2018. URL https://reliabilityweb.com/. Último acesso a: 2018-08-16.

[45] A. Géron. *Hands-on Machine Learning with Scikit-Learn and TensorFlow: Concepts, Tools, and Techniques to Build Intelligent Systems*. O'Reilly Media, Incorporated, 2017. ISBN 9781491962282. URL https://books.google.pt/books?id=f50otAEACAAJ.

[46] Mathworks. Machine Learning with MATLAB, 2018. URL https://www.mathworks.com/help/matlab/ref/var.html. Último acesso a: 2018-09-01.

[47] Techopedia. Artificial intelligence (ai), 2018. URL https://www.techopedia.com/definition/190/artificial. Último acesso a: 2018-09-01.

[48] Gary Lear. The struggle to define what Artificial Intelligence actually means, 2015. URL https://www.popsci.com/why-we-need-legal-definition-artificial-intelligence. Último acesso a: 2018-10-01.

[49] S. Russell and P. Norvig. *Artificial Intelligence: A Modern Approach*. Always learning. Pearson, 2016. ISBN 9781292153964. URL https://books.google.pt/books?id=XS9CjwEACAAJ.

[50] Mathworks. Mastering Machine Learning: a step-by-step guide withMATLAB, 2018. URL https://www.mathworks.com/campaigns/offers/mastering-machine-learning-with-matlab.html. Último acesso a: 2018-09-01.

[51] Andrew Ng. Machine Learning earning: Technical strategy for AI engineers, in the era of Deep Learning, 2017. URL http://www.mlyearning.org/. Último acesso a: 2018-09-01.

[52] Victor Lavrenko. Naive bayes classifier, 2015. URL https://www.youtube.com/playlist?list=PLBv09BD7ez_6CxkuiFTbL3jsn2Qd1IU7B. Último acesso a: 2018-09-01.

[53] Highbrow. B. F. Skinner, 2018. URL https://gohighbrow.com/b-f-skinner/. Último acesso a: 2018-09-01.

[54] Wikipedia. Deep Blue (chess computer), 2018. URL https://en.wikipedia.org/wiki/Deep_Blue_(chess_computer). Último acesso a: 2018-09-01.

[55] Quora. What is out-of-core learning?, 2018. URL https://www.quora.com/What-is-out-of-core-learning. Último acesso a: 2018-09-01.

[56] Victor Lavrenko. k-nearest neighbor algorithm, 2015. URL https://www.youtube.com/playlist?list=PLBv09BD7ez_48heon5Az-TsyoXVYOJtDZ. Último acesso a: 2018-09-01.

[57] Jason Brownlee. Machine Learning Algorithms, 2018. URL https://s3.amazonaws.com/MLMastery/MachineLearningAlgorithms.png?__s=jskdh1vsoduivip7es9d. Último acesso a: 2018-10-01.

[58] Mathworks. Classification Learner, 2018. URL https://www.mathworks.com/help/stats/classificationlearner-app.html. Matlab *Help*. Último acesso a: 2018-09-01.

[59] Mathworks. Box plot - MATLAB boxplot, 2018. URL https://www.mathworks.com/help/stats/boxplot.html. Matlab *Help*. Último acesso a: 2018-09-01.

[60] Mathworks. Principal component analysis of raw data - MATLAB pca, 2018. URL https://www.mathworks.com/help/stats/pca.html. Matlab *Help*. Último acesso a: 2018-09-01.

[61] Margaret Rouse. skewness, 2017. URL https://whatis.techtarget.com/definition/skewness. Último acesso a: 2018-09-01.

[62] Yevgeniy Brikman. The 10:1 rule of writing and programming, 2018. URL https://www.ybrikman.com/writing/2018/08/12/the-10-to-1-rule-of-writing-and-programming/. Último acesso a: 2018-10-01.

[63] Pedro Domingos. A few useful things to know about machine learning. *Commun. ACM*, 55(10):78–87, October 2012. ISSN 0001-0782. doi: 10.1145/2347736.2347755.

[64] Andrew Ng. Machine learning - Andrew Ng, Stanford University, 2017. URL https://www.youtube.com/playlist?list=PLLssT5z_DsK-h9vYZkQkYNWcItqhlRJLN. Último acesso a: 2018-09-01.

[65] Quora. Is it a good idea to learn machine learning in Matlab from Andrew Ng's course, or should one find another course in Python or R because that is how it will be used?, 2018. URL https://www.quora.com/Is-it-a-good-idea-to-learn-machine-learning-in-Matlab-from-Andrew-Ngs-course-or-should-one-find-another-course-in-Python-or-R-because-that-is-how-it-will-be-used. Último acesso a: 2018-09-01.

[66] Vincent Spruyt. The curse of dimensionality in classification, 2014. URL http://www.visiondummy.com/2014/04/curse-dimensionality-affect-classification/. Último acesso a: 2018-09-01.

[67] Mathworks. Selecting features for clasifying high-dimensional data, 2018. URL https://www.mathworks.com/help/stats/examples/selecting-features-for-classifying-high-dimensional-data.html. Matlab *Help*. Último acesso a: 2018-09-01.

[68] Hanchuan Peng, Fuhui Long, and C. Ding. Feature selection based on mutual information criteria of max-dependency, max-relevance, and min-redundancy. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 27(8):1226–1238, Aug 2005. ISSN 0162-8828. doi: 10.1109/TPAMI.2005.159.

[69] Wei Yang, Kuanquan Wang, and Wangmeng Zuo. Neighborhood component feature selection for high-dimensional data. 7:161–168, 01 2012.

[70] Mathworks. Machine learning challenges: Choosing the best model and avoiding overfitting, 2018. URL https://www.mathworks.com/campaigns/offers/common-machine-learning-challenges.html. Último acesso a: 2018-09-01.

[71] Robert Kelley. Making predictive models robust: holdout vs cross-validation, 2017. URL https://www.kdnuggets.com/2017/08/dataiku-predictive-model-holdout-cross-validation.html. Último acesso a: 2018-10-01.

[72] Sanyam Kapoor. Visualizing the confusion matrix, 2017. URL https://www.sanyamkapoor.com/machine-learning/confusion-matrix-visualization/. Último acesso a: 2018-10-01.

[73] A. Halevy, P. Norvig, and F. Pereira. The unreasonable effectiveness of data. *IEEE Intelligent Systems*, 24(2):8–12, March 2009. ISSN 1541-1672. doi: 10.1109/MIS.2009.36.

[74] Vishal Maini and Samer Sabri. Machine Learning for Humans, 2017. URL https://medium.com/machine-learning-for-humans/why-machine-learning-matters-6164faf1df12. Último acesso a: 2018-10-01.

[75] Kaggle. Kaggle is the place to do data science projects, 2018. URL https://www.kaggle.com/. Último acesso a: 2018-10-01.

Apêndice A: Descodificação das figuras do Capítulo 8

Na Tabela A.1 apresenta-se a descodificação das figuras apresentadas ao longo do Capítulo 8.

Tabela A.1: Descodificação das figuras apresentadas ao longo do Capítulo 8

feature	Figura	
	8.1	8.5
1	*TI*	*TI*
2	*PI*	*PI*
3	*motor*	*tempoultarranque*
4	*narranques*	*tempoultparagem*
5	*tempoultarranque*	*tempovidaemp*
6	*tempoultparagem*	*tempoultpress*
7	*tempofuncmudemp*	*pressaoapospress*
8	*tempovidaemp*	*aumentopressao*
9	*npress*	*declives*
10	*tempoultpress*	*tempoultsubsoposto*
11	*pressaoapospress*	*nobomba*
12	*aumentopressao*	*tipoemp*
13	*nquedaspressao*	-
14	*declives*	-
15	*tempoultsubsoposto*	-
16	*nobomba*	-
17	*tipoemp*	-