Universidade do Minho
Escola de Ciências

João Rodolfo Cardoso Alves

End-User Analytics: Comportamento de Máquinas e seus Utilizadores

Novembro de 2020

***End-User Analytics*: Comportamento de Máquinas e seus Utilizadores**
João Rodolfo Cardoso Alves
UMinho | 2020

Universidade do Minho
Escola de Ciências

João Rodolfo Cardoso Alves

End-User Analytics: Comportamento de Máquinas e seus Utilizadores

Dissertação de Mestrado
em Matemática e Computação

Trabalho efetuado sob a orientação da
Professora Doutora Ana Paula Costa Conceição Amorim

Novembro de 2020

Direitos de Autor e Condições de Utilização do Trabalho Por Terceiros

Este é um trabalho académico que pode ser utilizado por terceiros desde que respeitadas as regras e boas práticas internacionalmente aceites, no que concerne aos direitos de autor e direitos conexos. Assim, o presente trabalho pode ser utilizado nos termos previstos na licença abaixo indicada. Caso o utilizador necessite de permissão para poder fazer um uso do trabalho em condições não previstas no licenciamento indicado, deverá contactar o autor, através do RepositóriUM da Universidade do Minho.

Atribuição-NãoComercial-CompartilhaIgual
CC BY-NC-SA
https://creativecommons.org/licenses/by-nc-sa/4.0/

Agradecimentos

Em primeiro lugar, gostaria de expressar a minha gratidão em especial à minha orientadora, a Professora Doutora Ana Paula Amorim, por ter aceitado supervisionar o meu trabalho, além de, gentilmente, sempre se ter mostrado disponível para me ir ajudando no processo da elaboração desta dissertação, com as dificuldades acrescidas desta situação de pandemia.

Além disso, queria agradecer à *Fujitsu*, não só pela oportunidade, mas também pela persistência com que me ajudaram, principalmente, com a questão dos dados. Um agradecimento especial ao meu tutor na empresa José João e também ao Pedro, por me terem acompanhado no processo de obtenção dos dados e na contribuição para a sua análise, mesmo sabendo que andavam atarefados com outros assuntos da empresa, não podendo esquecer também a Ana Margarida.

Queria agradecer também aos meus colegas de grupo, a Cecília e o Fernando, que me acompanharam no decorrer do mestrado e que se tornaram, sem dúvida, dois bons amigos. Em especial à Cecília, por me ter ajudado nas formatações da dissertação.

Um agradecimento à minha Tia Ló, por me ter dado umas dicas na escrita da dissertação em bom português.

Finalmente, queria expressar o meu profundo agradecimento à minha mãe, pelos sacrifícios que teve que fazer no decorrer da minha formação, especialmente durante este Mestrado.

Um agradecimento extra a todas as outras pessoas que estiveram envolvidas no meu percurso e que não poderei colocar explicitamente nesta secção.

Declaração de Integridade

Declaro ter atuado com integridade na elaboração do presente trabalho académico e confirmo que não recorri à prática de plágio nem a qualquer forma de utilização indevida ou falsificação de informações ou resultados em nenhuma das etapas conducente à sua elaboração. Mais declaro que conheço e que respeitei o Código de Conduta Ética da Universidade do Minho.

Resumo

End-User Analytics centra-se na análise dos dados do comportamento dos utilizadores e da interação com as respetivas máquinas.

A Fujitsu tem vindo a investir bastante em *End-User Analytics* com o objetivo de ajudar as empresas a transitar para soluções mais digitais, através da análise de dados do comportamento operacional obtidos nas máquinas dos seus clientes.

Para tal, a *Fujitsu* disponibiliza numa única plataforma a recolha de dados de várias fontes em tempo-real, incluindo dados sobre licenças, sobre os utilizadores e as aplicações envolvidas. Este *software* consegue detetar erros e paragens de aplicações, distinguir problemas de utilizadores únicos de problemas mais globais, assim como identificar comportamentos ou aplicações de risco.

Os pontos descritos anteriormente, em conjunto com o potencial atual de recolha de enormes quantidades de dados através das máquinas das empresas, impulsionaram a *Fujitsu* a expandir o tema de *End-User Analytics* a outras áreas, utilizando abordagens e métodos mais automatizados como, por exemplo, a utilização de técnicas de *Machine Learning*.

Nesta dissertação, foram exploradas abordagens analíticas de forma a encontrar padrões que potenciem a experiência dos utilizadores e respetivas máquinas, juntamente com as métricas associadas aos dados, incluindo métricas que permitam avaliar o desempenho dessa experiência.

A análise também incidiu sobre um conjunto de dados exemplo proveniente da plataforma da empresa *Nexthink*. Estes dados não permitiram uma análise direta da experiência dos utilizadores mas, mesmo assim, foi explorada uma abordagem indireta. Os resultados obtidos não permitiram uma boa análise preditiva dos eventos associados às máquinas. No entanto, foi realizada uma abordagem de monitorização com base nos dados, que sugere conjuntos de máquinas (ou outros atributos destas) no qual a empresa em questão se deve focar de forma a isolar a maioria problemas encontrados. Esta abordagem deu ainda origem a uma aplicação de monitorização.

Abstract

End-User Analytics focuses on the analysis of user behavior data and their interaction with the respective machines. Fujitsu has invested heavily in End-User Analytics to help companies move towards more digital solutions by analyzing the operational behavior data obtained on their customers' machines.

To this end, Fujitsu provides a single platform for the gathering of data from various sources in real-time, including data on licenses, users and the applications involved. This software is able to detect errors and application issues, distinguish single user problems from more global ones, as well as identify risky behaviors or applications.

The points described above, together with the current potential to collect huge amounts of data through company machines, have driven Fujitsu to expand the subject of End-User Analytics to other areas, using more automated approaches and methods such as the use of Machine Learning techniques.

In this dissertation, analytical approaches were explored in order to find patterns that enhance the experience of users and their machines, together with the metrics associated with the data, including metrics that allow the performance of that experience to be evaluated.

The analysis also focused on a set of example data from the Nexthink platform. This data did not allow for a direct analysis of the user experience, but nevertheless an indirect analysis was explored. The results obtained did not lead to a good predictive analysis of the events associated with the machines. However, a monitoring approach based on the data was carried out, which suggests sets of machines (or other attributes of these) on which the company in question should focus on in order to isolate the majority of problems encountered. This approach has also led to a monitoring application.

Lista de Figuras

1.1 Logótipo da empresa *Fujitsu*. 20

2.1 Exemplo de corredor de um Centro de Dados. 26
2.2 Distribuição do consumo energético de um CPU com 4 *cores*. 28
2.3 *Layout* da *interface* da ferramenta *Google Analytics*. 34
2.4 Representação dos caminhos seguidos por um utilizador num dado *website*. 41
2.5 Exemplo de uma árvore de decisão. 43
2.6 Visualização gráfica das curvas de entropia (a vermelho) e índice de *Gini* (a verde). 46
2.7 Visualização gráfica da estrutura de uma *Random Forest* e do método de previsão. 48
2.8 Exemplo de clustering aplicado a um conjunto de dados a duas dimensões. 50
2.9 Procedimento algorítmico de *K-means clustering*. 51

4.1 Gráfico de barras com os erros de *device* por dia para cada tipo de *erro*. 71
4.2 Gráfico de barras com os *warnings* de *device* por dia para cada tipo de *warning*. 71
4.3 Gráfico de barras com os erros de *device* para os 14 *devices* com mais erros, por tipo de erro. 72
4.4 Gráfico de barras com os *warnings* de *device* para os 16 *devices* com mais *warnings*, por tipo de *warning*. 73
4.5 Gráfico de barras com os erros de *execution* por dia, por tipo de erro. 74
4.6 Gráfico de barras com os *warnings* de *execution* por dia, por tipo de *warning*. 74
4.7 Gráfico de barras com os erros de *execution* por *application*, por tipo de *erro*, para as 20 *applications* com mais erros. 75

4.8 Gráfico de barras com os *warnings* de *execution* por *application*, por tipo de *warning*, para as 20 *applications* com mais warnings. 75
4.9 Gráfico de barras com os erros de *execution* por *user*, por tipo de erro, para os 20 *users* com mais erros. 76
4.10 Gráfico de barras com os *warnings* de *execution* por *user*, por tipo de *warning*, para os 20 *users* com mais warnings. 76
4.11 Gráfico de barras com os erros de *execution* por *device*, por tipo de erro, para os 20 *devices* com mais erros. 77
4.12 Gráfico de barras com os *warnings* de *execution* por *device*, por tipo de *warning*, para os 20 *devices* com mais warnings. 77
4.13 Gráfico de Erros vs *Warnings* de *device* em escala logarítmica. 78
4.14 Gráfico de Erros vs *Warnings* de *execution* em escala logarítmica. . . 79
4.15 Representação 3D (por PCA) do *clustering* aplicado aos *users*. 81
4.16 Representação 3D (por PCA) do *clustering* aplicado aos *devices*. . . . 81
4.17 Resultados da Previsão de Erros de *Device* por dias ativos para o algoritmo de *Random Forest*. 83
4.18 Resultados da Previsão de Erros de *Device* por dias ativos para o algoritmo de *Gradient Boost*. 84
4.19 Árvore binária que descreve os resultados para um processo probabilístico de 4 categorias com probabilidades $0.5, 0.25, 0.125, 0.125$, respetivamente. 86
4.20 Janela de apresentação inicial da GUI. 94
4.21 Procedimento para carregar os dados para a memória, especificando o caminho para a base de dados. 95
4.22 Procedimento de escolha das datas iniciais e finais de cálculo, assim como do fator de γ (*"gamma"*). 95
4.23 Interatividade da GUI agora toda disponível para o utilizador. 96
4.24 Imagem ilustrativa da GUI apresentando os gráficos de barras da entropia modificada para cada tipo de evento/objeto/categoria, ordenados pelo valor da entropia modificada. 97
4.25 Figura demonstrativa da interactividade com os gráficos da GUI. . . . 98
4.26 Figura demonstrativa da opção de visualizar a tabela correspondente a um evento/objeto/categoria. 99

Lista de Tabelas

3.1 Descrição da Tabela *"User"*. 57
3.2 Descrição da Tabela *"Device"*. 58
3.3 Descrição da Tabela *"Application"*. 58
3.4 Descrição da Tabela *"Execution"*. 59
3.5 Descrição da Tabela *"Device_error"*. 59
3.6 Descrição da Tabela *"Device_warning"*. 59
3.7 Descrição da Tabela *"Execution_error"*. 60
3.8 Descrição da Tabela *"Execution_warning"*. 60
3.9 Descrição da Tabela *"Execution_relations"*. 60
3.10 Descrição da Tabela *"Device_error_relations"*. 61
3.11 Descrição da Tabela *"Device_warning_relations"*. 61
3.12 Descrição da Tabela *"Execution_error_relations"*. 61
3.13 Descrição da Tabela *"Execution_warning_relations"*. 61

4.1 Estatística da Tabela *"Application"*. 66
4.2 Estatística da Tabela *"Device"*. 67
4.3 Estatística da Tabela *"User"*. 67
4.4 Estatística da Tabela *"Device_error"*. 68
4.5 Estatística da Tabela *"Device_error_relations"*. 68
4.6 Estatística da Tabela *"Device_warning"*. 68
4.7 Estatística da Tabela *"Device_warning_relations"*. 68
4.8 Estatística da Tabela *"Execution_error"*. 68
4.9 Estatística da Tabela *"Execution_error_relations"*. 69
4.10 Estatística da Tabela *"Execution_warning"*. 69
4.11 Estatística da Tabela *"Execution_warning_relations"*. 69
4.12 Estatística da Tabela *"Execution"*. 69
4.13 Estatística da Tabela *"Execution_relations"*. 70

4.14 Tabela com a Entropia Modificada ($H_m(4.4)$) ordenada crescente para os diversos atributos de cada objeto para erros de *device*. 89
4.15 Tabela com a Entropia Modificada ($H_m(4.4)$) ordenada crescente para os diversos atributos de cada objeto para *warnings* de *device*. 90
4.16 Tabela dos Erros de *device* por Sistema Operativo para os objetos *device*. 90
4.17 Tabela com a Entropia Modificada ordenada crescente para os diversos atributos de cada objeto para erros de *execution*. 91
4.18 Tabela com a Entropia Modificada ordenada crescente para os diversos atributos de cada objeto para *warnings* de *execution*. 92
4.19 Tabela dos Erros de *execution* por *Department* para os *Users*. 92
4.20 Tabela dos Erros de *execution* por *Department* para os *Users*, com γ de 1.7. 93

Lista de Abreviações

ICT *Information and Communications Technology*
UX *User-Experience*
GUI *Graphical User Interface*
SLAs *Service Level Agreements*
IT *Information Technology*
E.U.A Estados Unidos da América
KWh *Kilowatt-hour*
CPU *Central Processing Unit*
CDs Centros de Dados
HTML *HyperText Markup Language*
IP *Internet Protocol*
URL *Uniform Resource Locator*
CPC *cost per click*
ID3 *Iterative Dichotomiser 3*
CART *Classification and Regression Tree*
NXQL *Nexthink Query Language*
SQL *Structure Query Language*
API *Application Programming Interface*
SMART *Self-Monitoring, Analysis and Reporting Technology*
PCA *Principal Component Analysis*

Conteúdo

Agradecimentos iii
Resumo vii
Abstract ix
Lista de Figuras xii
Lista de Tabelas xiv
Lista de Abreviações xv

1 Introdução e Estrutura da Tese **19**
1.1 Introdução 20
1.1.1 Contextualização da Dissertação 20
1.1.2 Definição de *End-User Analytics* 21
1.1.3 Estado Atual da *Fujistu* em *End-User Analytics* 22
1.1.4 Objetivo da Dissertação 22
1.2 Estrutura da Tese 24

2 Estado de Arte **25**
2.1 Otimização Energética 26
2.1.1 Centros de Dados 26
2.1.2 Otimização da Rede 27
2.1.3 Otimização da Planta do Armazém 27
2.1.4 Gestão do Agendamento 28
2.1.5 Análise aplicada à Otimização de CDs 29
2.2 *Web-Analytics* 32
2.2.1 Adquisição de Informação sobre o *Website* 32
2.2.2 Métricas e Atributos 34
2.2.3 Objetivos 36
2.2.4 *Web-Analytics* aplicado aos utilizadores 37
2.3 Algoritmos 42

2.3.1 Árvores de Decisão 42
2.3.2 *Random Forest* 47
2.3.3 *Gradient Boosting* 48
2.3.4 Análise de *Clustering* 49

3 Descrição de Dados **55**
3.1 *Nexthink* 56
3.2 Dados Utilizados na Análise 56
3.3 Conexão do Estado de Arte com os Dados 62
3.3.1 Otimização energética 62
3.3.2 *Web-Analytics* 62

4 Análise de Dados e Resultados **65**
4.1 Estatística Descritiva 66
4.2 Visualização dos Dados 70
4.2.1 Eventos de *Device* 71
4.2.2 Eventos de *Execution* 74
4.2.3 Erros vs *Warnings* 78
4.3 Análise de *Clustering* 80
4.4 Regressão de *Random Forest* e *Gradient Boost* 83
4.5 Entropia Modificada 86
4.5.1 Introdução 86
4.5.2 Entropia Modificada 87
4.5.3 Aplicação GUI de Monitorização 94

5 Conclusões e Trabalho Futuro **101**

Capítulo 1

Introdução e Estrutura da Tese

1.1 Introdução

1.1.1 Contextualização da Dissertação

Esta dissertação integra-se na unidade curricular de Dissertação do curso de Mestrado em Matemática e Computação da Universidade do Minho. É realizado em conjunto com o Centro de Competências da empresa *Fujitsu* (Figura 1.1 mostra o logótipo da empresa) de Braga, com o tema "*End-User Analytics*: Comportamento de Máquinas e seus Utilizadores".

Figura 1.1: Logótipo da empresa *Fujitsu*.

O que é a *Fujitsu*?

A *Fujitsu* é a empresa líder japonesa em tecnologia de informação e comunicação (ICT) e a 7ª no mundo em termos de fornecimento destes serviços. Está entre os 10 primeiros fornecedores de servidores a nível mundial e oferece uma vasta gama de produtos, serviços e soluções tecnológicas. Nestas incluem-se *Client Computing Devices, Peripherical devices, Integrated Systems, Servers, Data Storage, Infrastructure Management e Security* [1] [2]. Empregando cerca de 132 mil colaboradores em mais de 100 países, a *Fujitsu* apresentou uma receita consolidada de cerca de 36 mil milhões de dólares no ano fiscal de 2019.

Fujitsu em Portugal

Em Portugal, a *Fujitsu* é o maior empregador japonês que conta com mais de 1900 colaboradores. Com sede em Lisboa e operações no Porto e em Braga, apresenta como principais setores de atividade no país os de Administração Pública, onde assegura a Gestão Documental em 80% dos Ministérios e o balcão de atendimento responde a mais de 10 milhões de chamadas por ano; Retalho, apresentando-se como líderes, sendo responsáveis por mais de 400 milhões de transações por ano em mais de 500 supermercados e hipermercados; na Banca, marcando presença em mais de 1700 balcões, prestando suporte técnico a 20 mil utilizadores; e nos Transportes, onde,

através de sistemas de bilheteira inteligente, emitem mais de 20 milhões de bilhetes por ano e representam 250 mil horas de voo por ano [3]. Encontra-se atualmente em expansão, sendo que a sede em Lisboa foi inaugurada em 2008, o centro de operações em Braga em 2016 e foi anunciada a assinatura de um protocolo em 2019 para a instalação em Viseu [4] [5].

***Fujitsu* em Braga**

O centro da *Fujitsu* em Braga, localizado no Pólo de Negócios de Braga e inaugurado em 2016, é uma extensão do *Global Delivery Center* de Lisboa e fornece atualmente suporte técnico a milhares de utilizadores da empresa [6].

1.1.2 Definição de *End-User Analytics*

End-User Analytics refere-se à análise de dados, dados estes recolhidos de dispositivos dos utilizadores ou de sensores que recolhem métricas geradas pelo comportamento desses utilizadores, fazendo uso de métodos computacionais de forma a obter conhecimento sobre padrões do decorrer dos processos, para assim fornecer uma visualização mais simplificada ou certas indicações por forma a melhorar estas interações entre utilizador e o ambiente que o rodeia.
Seguindo as próprias definições de *End-User* e de *Analytics*, fornecidas pelo dicionário de *Cambridge* [7] [8]:

End-User:

> *"the person or organization that uses a product or service."*

Analytics:

> *"a process in which a computer examines information using mathematical methods in order to find useful patterns."*

Partindo das definições anteriores, podemos definir então *End-User Analytics* como *O processo segundo o qual se analisa computacionalmente informação usando métodos matemáticos, de forma a encontrar padrões úteis acerca de determinado utilizador de um certo produto ou serviço.*

No caso específico da *Fujitsu*, esta abordagem é aplicada ao ambiente laboral e tecnológico de uma determinada empresa, num ambiente de consultoria.

1.1.3 Estado Atual da *Fujistu* em *End-User Analytics*

Atualmente, a *Fujitsu* utiliza a abordagem de *End-User Analytics*, por exemplo, na monitorização de licenciamentos de *software* pelos utilizadores (*License Management*), para que deste modo, haja um acompanhamento sobre as licenças que estão no fim de prazo e necessitem, ou não, de ser renovadas [9]. Outros exemplos de áreas onde também aplicam esta abordagem incluem-se *Power Management*, *Application issues*, *Build version issues*, *OS issues*, *Network issues - connection/destination analysis*, *Hardware fail prevention*, entre outros.

1.1.4 Objetivo da Dissertação

O objetivo desta dissertação é explorar de que forma a experiência dos utilizadores (*User-Experience*) poderá ser melhorada no contexto de *End-User Analytics* desenvolvida pela *Fujitsu*. Desta forma, esta dissertação tem uma forte componente exploratória que pretende criar valor no campo da *User-Experience*, através da identificação de padrões, no caso hipotético em que um conjunto enorme de dados sobre as rotinas, dispositivos e eventos dos elementos pertencentes a uma determinada empresa tecnológica esteja disponível. Fará parte do objetivo, portanto, a descoberta científica e empresarial de métricas importantes a levar a cabo na recolha dos dados das diferentes abordagens existentes na literatura para a análise de dados propriamente dita, dos objetivos específicos que podem ser conseguidos com esta abordagem e da definição de métricas alvo que permitam medir os objetivos definidos, no âmbito de *User-Experience*.

Além disso, terá também como objetivo aplicar algumas das métricas, abordagens e métodos que provêm da exploração científica a um conjunto de dados exemplo, tendo em conta o objetivo final de melhorar a experiência dos utilizadores.

Tipicamente, profissionais de *User-Experience* (*UX*) querem entender o comportamento dos utilizadores e, principalmente, o porquê de os terem. A análise de dados permitirá revelar o que esses mesmos utilizadores fazem (*"what?"*) e como o

fazem (*"how?"*) e testar teorias do porquê de o fazerem, já que os dados em si não permitem inferir diretamente respostas para as perguntas do tipo *"porquê?"*.

1.2 Estrutura da Tese

A presente dissertação encontra-se dividida em 5 capítulos.

No **capítulo 1**, faz-se uma introdução à dissertação, contextualizando-a e descrevendo o seu objetivo. Além disso, são dispostas algumas definições no contexto de *User-Experience* (*UX*), e o seu estado atual na empresa *Fujitsu*.

No **capítulo 2**, são exploradas e apresentadas métricas, abordagens analíticas e objetivos recolhidos da literatura relativamente à análise de dados para a melhoria da *User-Experience*. Mais especificamente, esta recolha focou-se em Otimização Energética e em *Web-Analytics* como formas de atingir essa melhoria. São também descritas as bases teóricas dos diversos algoritmos usados na análise de dados.

No **capítulo 3**, contextualiza-se os dados analisados na dissertação, descrevendo-os mais detalhadamente. Expõe-se também a conexão entre as abordagens exploradas no estado de arte com os dados utilizados na análise específica deste projeto. Para além disso, é explicado que tipos de dados fariam sentido para se atingir certos objetivos propostos na literatura.

No **capítulo 4**, apresenta-se a descrição da análise de dados e os seus resultados. Numa primeira parte, expõe-se a estatística descritiva desses mesmos dados e a sua visualização gráfica. Numa segunda parte, descreve-se a abordagem analítica tomada nesta dissertação, assim como os resultados obtidos. Por último, é descrito o desenvolvimento de uma aplicação GUI de forma a realizar manipulação dos dados, no contexto da sua monitorização.

E finalmente, o **capítulo 5** é dedicado à apresentação das principais conclusões do estudo desenvolvido, assim como sugestões para trabalhos futuros.

Capítulo 2

Estado de Arte

2.1 Otimização Energética

Um dos principais métodos encontrados na literatura, no contexto da análise de dados de uma organização, aborda a otimização energética. Esta otimização demonstrou ser rentável no quadro específico de centro de dados e infraestruturas IT [10].

2.1.1 Centros de Dados

Centro de Dados são espaços dedicados ao armazenamento de recursos computacionais por forma a armazenar e processar uma grande quantidade de dados. Possuem toda uma infraestrutura de servidores e de rede que dão suporte computacional a uma ou várias organizações. A Figura 2.1 mostra um exemplo de um Centro de Dados [11].

Figura 2.1: Exemplo de corredor de um Centro de Dados.

Com a recente explosão na disponibilidade de obtenção de informação digital por parte das organizações tecnológicas, os centros de dados colocam-se como as infraestruturas centrais no suporte desta tendência. Sendo assim, cada vez mais atenção é dada à otimização de tais infraestruturas [11].

De acordo com estatísticas referidas em [12], o consumo energético dos centros de dados dos E.U.A. em 2006 eram de cerca de 61 mil milhões de KWh, sendo que em 2014 eram de 70 mil milhões de KWh. Estes números representam entre 1.5% a 1.8% de toda a energia consumida nos E.U.A. Quanto à perspetiva global,

o consumo de energia pelos centros de dados são responsáveis por 1.1% a 1.5% do consumo de energia, dados de 2011 [13].

O consumo energético dos centros de dados podem dividir-se em duas categorias: recursos computacionais e recursos físicos [11]. De acordo com [14], os recursos computacionais representam cerca de 50% da energia total consumida pelo centros de dados, sendo que 40% se referem à computação propriamente dita, 5% às comunicações entre dispositivos e os outros 5% ao armazenamento. Dos restantes, 40% referem-se aos custos dos sistemas de refrigeração e outros 10% aos sistemas de fornecimento de energia e outros factores distintos. Sendo assim, focar esforços na redução do consumo energético dos servidores e do sistema de refrigeração coloca-se como a chave para a otimização do consumo nos centros de dados [11].

O consumo energético dos centros de dados tem levantado preocupações tanto na esfera académica como na industrial, seja pelo custo elevado da eletricidade associada ou pela pegada de carbono no âmbito ambiental [15].

2.1.2 Otimização da Rede

A constante necessidade de expansão da infraestrutura física dos centros de dados coloca pressão na infraestrutura interna de rede, onde se incluem *routers*, *switches*, etc. A investigação em relação à eficiência energética desta rede tem-se tornado um tópico popular [16]. Estudos sobre a estrutura topológica da rede têm produzido resultados na redução e otimização energética. Em particular, através da utilização de tecnologia de agente verde (*green agent technology*) que permite acordar os nódos que necessitam de funcionar através da utilização de um mecanismo de dormência, sem que isto afete o desempenho da rede [17] [18] [19]. Também referem o estudo de topologia distribuída e algoritmos cooperativos na análise topológia com o intuito de conservar energia.

2.1.3 Otimização da Planta do Armazém

Um dos grandes utilizadores de energia nos centros de dados é o sistema de refrigeração. Este tipicamente consome cerca de 40% da energia utilizada pelo centro, podendo inclusive chegar aos 50% [20]. Neste campo, há duas formas descritas na literatura para reduzir o consumo energético. Estas são a arquitetura do edifício e o sistema de ventilação. Tipicamente, sistemas de ar-condicionado criam o que se chama por *hot island effect*, já que o ar-condicionado cria um micro ambiente

externo que requer um progressivo gasto energético. Assim, a escolha do local para alojar o centro de dados é de significativa importância. A Google vai investir 200 milhões de euros na construção de um centro de dados no Norte da Europa [21] e o Facebook vai construir centros de dados na Suécia.

Quanto à estrutura dos edifícios, foram analisados ainda a influência da altura do teto do edifício na condução de ar pelo armazém [22], a construção de aberturas no chão [23], entre outros.

2.1.4 Gestão do Agendamento

Como foi anteriormente referido, os recursos computacionais representam cerca de 50% dos gastos energéticos dos centros de dados. A taxa de utilização dos servidores atinge apenas cerca de 20% do estabelecido [24] [25]. É reportado em [26] que o agendamento e a otimização de clusters de servidores pode reduzir efetivamente a sua taxa de inatividade e, assim, o consumo energético [26]. De acordo com [27] [28] [29], a energia média consumida por um servidor livre (inativo) é de cerca de 70% do consumo energético desse mesmo servidor quando está a funcionar a capacidade máxima. Pode-se ver pela Figura 2.2, a distribuição da potência usada por um 4-CPU com diferentes cargas de utilização [10].

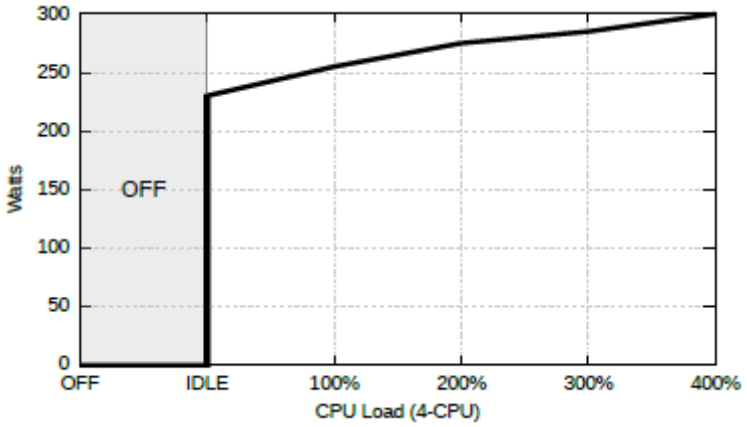


Figura 2.2: Distribuição do consumo energético de um CPU com 4 *cores*.

Sendo assim, conclui-se que se pode fazer uso de métodos analíticos de previsão e otimização da atribuição de tarefas computacionais aos servidores, de forma a reduzir os servidores inativos e assim poupar uma quantidade substancial de energia no processo.

Esta vertente será explorada em maior detalhe no restante desta secção sobre otimização energética, explorando diversos métodos encontrados na literatura, que pretendem, através de métodos analíticos, otimizar a atribuição de tarefas de modo a fazer uso da utilização eficiente dos servidores, mantendo os parâmetros de qualidade de serviço fornecido.

2.1.5 Análise aplicada à Otimização de CDs

A comunidade científica tem-se deparado com diversos desafios no que concerne à redefinição dos Centros de Dados. A eficiência energética surge como mais um, onde se incluem também a disponibilidade, confiabilidade e desempenho. No caso particular dos métodos de poupança de energia, vários têm sido explorados, sendo que, muitas vezes, estes se focam apenas em cenários muito particulares e restritivos, estratégias simples ou até com dados sintéticos em algumas partes do processo [10].

Dentro destes métodos, os dois mais representativos são a consolidação da carga de trabalho e o desligar servidores inativos. A consolidação da carga de trabalho implica uma estratégia de agregação de tarefas de várias máquinas e aplicações num número reduzido de sistemas. Esta abordagem permite diminuir a disponibilidade de *hardware*, o consumo energético ao nível dos servidores e do sistema de refrigeração e ainda reduzir o espaço físico necessário. Uma gestão inteligente das tarefas permite desligar os servidores inativos levando, obviamente, a um ganho energético, mantendo os níveis de desempenho [10].

Os níveis de desempenho são tipicamente avaliados sob *Service-Level Agreements* (SLA's). SLA's são acordos estabelecidos por contrato, onde se especificam garantias de disponibilidade e/ou desempenho por parte do prestador do serviço. Aqui incluem-se largura de banda, disponibilidade de disco ou CPU, tempo de resposta, objetivos temporais, e outros.

Em seguida, será descrita uma abordagem em particular, aplicada a três cenários distintos dentro da otimização analítica dos Centros de Dados [10].

Num primeiro cenário, aplicam-se métodos analíticos ao agendamento em Centros de Dados. Em [10] são aplicadas estratégias de agendamento de modo a reduzir o número de máquinas inativas, tendo em conta a carga de processamento necessário a cada momento, e decidir a colocação de tarefas e a sua realocação de forma a compactar as tarefas na menor quantidade de máquinas, sem reduzir as metas de SLA's.

Basicamente, a abordagem faz uso de algoritmos de preenchimento das tarefas nos nódos de processamento. Nestes algoritmos, incluem-se um algoritmo *Random*, sendo as tarefas atribuídas aleatoriamente, levando em conta se o nódo cabe na máquina; *Round Robin*, que consiste em atribuir as tarefas aos nódos existentes, maximizando assim os recursos para a tarefa em causa, mas fazendo um uso desadequado dos recursos; *Backfilling*, que tenta preencher o maior número possível de tarefas numa máquina até a preencher; e *Dynamic Backfilling* que permite mover tarefas de modo a otimizar o preenchimento e assim obter uma maior consolidação. Devido ao facto deste último algoritmo ter um elevado custo computacional, os autores apontam o uso de *Reinforcement learning* no futuro, como forma de modelar este processo.

Dynamic Backfiling apresenta um bom desempenho quando são conhecidos os recursos necessários para cada tarefa com precisão. Este não é sempre o caso, seja por não ser conhecido ou pela informação fornecida não ser a mais precisa. Deste modo, surge a necessidade de prever que recursos serão necessários na execução das tarefas requeridas. É aqui que surge então a necessidade de aplicar métodos de *Machine Learning* e integrar no algoritmo de *Dynamic Backfiling*. Neste artigo foi utilizada a regressão linear para prever o uso de CPU [30], e o algoritmo M5P (tree based) na previsão do gasto de energético, já que esta se apresenta como mais complexa e com uma relação bastante não linear em relação ao uso de CPU.

O correspondente SLA usado nesta abordagem foi o limite de tempo. Este verifica se a tarefa foi concretizada dentro do tempo indicado pelo SLA.

Daqui foi então possível concluir que o algoritmo *Dynamic Backfiling* incorporado com *Machine Learning* deu piores resultados no enquadramento do SLA quando a utilização de CPU era elevada. É dado como justificação o facto deste algoritmo não ter acesso prévio à informação fornecida pelo utilizador sobre o gasto de CPU previsto, pondo-o então numa posição de desvantagem em relação ao *Dynamic Backfiling* simples. No entanto, demonstrou resultados significativamente melhores quando comparado com o *Random* e o *Backfiling* [10].

Num artigo posterior, os mesmos autores numa outra abordagem, fazem uso do mesmo método aplicado a um contexto ligeiramente diferente. Desta vez, incluem dados sobre a infraestrutura do Centro de Dados.

Foram capazes de mostrar que também neste projeto conseguiram integrar métodos de *Machine Learning* de forma a melhorar os resultados obtidos e demons-

trar que é possível obter uma poupança energética considerável, fazendo uso de métodos analíticos num contexto de alocação de tarefas e consequente consolidação, permitindo assim desligar servidores inativos de forma mais eficiente [31].

Num terceiro cenário, os mesmos autores utilizaram uma abordagem similar à descrita anteriormente, mas desta vez a uma composição de vários Centros de Dados, ou seja, Multi-Centro de Dados. Desta vez, incluiram informação sobre a localização do Centro de Dados, bem como os preços da energia nos respetivos locais, de modo a aplicar uma otimização mais global. Segundo os autores, mais uma vez, o uso de métodos analíticos, onde se incluem a integração de algoritmos de *Machine Learning*, permitiu trazer ganhos energéticos para a rede de Centros de Dados [32].

2.2 *Web-Analytics*

A internet é uma fonte enorme de dados relativos à interação de uma grande quantidade de utilizadores com páginas *web*. *Web-Analytics* surge como uma forma de retirar conhecimento sobre como os utilizadores interagem com determinados websites e aplicações móveis, através da recolha automática de aspetos do comportamento desses mesmos utilizadores nos websites, e tratar e transformar esse comportamento em dados que possam ser analisados. A informação mais fundamental e útil em *Web-Analytics* refere-se ao conjunto de páginas que o utilizador percorre e à sua ordem.

Neste sentido, faz-se uso desta abordagem para obter conhecimento acerca de utilizadores e das suas interações com os websites, e assim colocar em prática uma série de padrões de *design* destes, por forma a potencializar tanto a satisfação dos utilizadores, como a sua produtividade [33] [34].

A maioria das ferramentas de *Web-Analytics* são aplicadas no ramo de marketing online, em que as diversas marcas das empresas são introduzidas ao consumidor, na tentativa de os seduzir a comprarem os seus produtos. Neste ramo, *Web-Analytics* permite efetivamente medir a eficácia das estratégias utilizadas, fazendo uso de um determinado conjunto de métricas bem definidas e estabelecidas. Métricas como o número de pessoas que chegam ao *website* ou que realmente compram algum produto, permitem comparar o tempo e dinheiro gasto a adquirir tais vendas.

No entanto, os profissionais de *Web-Analytics*, fazendo uso das ferramentas e código certos, conseguem ter ao seu dispor dados bastante mais diversos acerca de como os utilizadores navegam no *website* em questão. Dados como a forma como os utilizadores chegaram ao *website*, se pela utilização de um *search engine* ou através de um link de um outro *website* ou até dados mais técnicos sobre o sistema operativo dos utilizadores ou a resolução de ecrã, permitem alargar ou melhorar as inferências feitas sobre o comportamento desses mesmos utilizadores [33].

O restante desta secção será dedicado à exposição de diversas abordagens e ferramentas ao dispor de profissionais de *User-Experience* aplicado a *Web-Analytics*.

2.2.1 Adquisição de Informação sobre o *Website*

O primeiro passo na análise de um determinado *website* é, na verdade, não fazer uso de nenhuma ferramenta específica de *web-analytics*. Explorar o *website*

é a melhor forma de o conhecer, desde navegar pelos links indicados até estudar a disposição das diferentes páginas, o que ajuda o profissional a entender o significado dos dados a serem retirados e analisados, além de dar uma ideia das dificuldades na navegação.

O passo seguinte será então obter os dados sobre a navegação dos utilizadores em determinado *website*. Há dois métodos simples de obtenção desta informação: *log files* e *page tagging*.

Log files monitorizam que páginas são abertas para cada *webpage request*. Estes permitem manter informação aprofundada sobre a atividade desenvolvida e assim formar um *dataset* mais extenso. No entanto, a análise de *log files* pode ser desafiante, tanto na utilização como na implementação, embora as ferramentas sejam bastante mais dispendiosos. Nas ferramentas que permitem a utilização da análise de *log files* incluem-se *AWStats* e *Sawmill*.

Por outro lado, *Page tagging* funciona inserindo um pequeno pedaço de código *Javascript* em todas as páginas do *website* que se queira monitorizar para assim recolher os dados de cada vez que o utilizador carregar a página HTML, ou então através da utilização de cookies colocados no dispositivo do utilizador. Deste modo, apenas a informação que o profissional implemente será recolhida, permitindo a formação de um dataset mais rico, em comparação com a análise de *log files*. Além disso, *page tagging* é bastante mais simples de usar e mais acessível do ponto de vista comercial. *Google Analytics*, *Omniture* ou *Webtrends* são ferramentas para este último tipo de análise. A Figura 2.3 mostra o *layout* da *interface* da ferramenta *Google Analytics* [33].

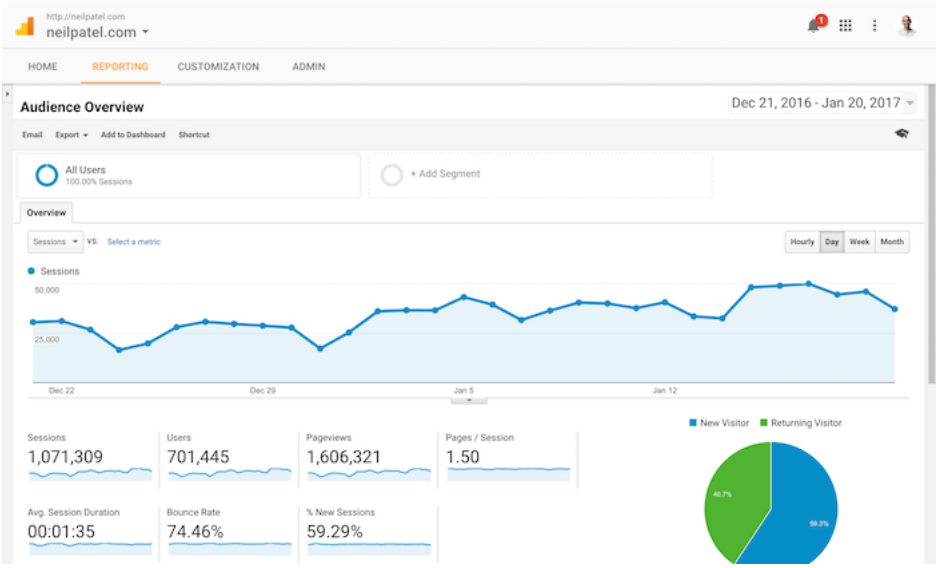


Figura 2.3: *Layout* da *interface* da ferramenta *Google Analytics*.

2.2.2 Métricas e Atributos

Dois dos principais conceitos em *web-analytics* são as métricas e os atributos. Métricas definem-se como medidas quantitativas sobre vários aspetos do comportamento dos utilizadores podendo, por exemplo, ser expressas sob a forma de uma soma, de uma média ou de uma taxa. Atributos, por outro lado e neste contexto, são categorias em que os dados podem ser agrupados, definindo-se mais como características qualitativas sobre os utilizadores, os seus dispositivos ou sobre partes do website que os utilizadores visitam. Em *Analytics*, faz-se uso da conjugação entre as métricas e os atributos. Em seguida, são exemplificadas alguns tipos de métricas tipicamente analisadas no contexto de *web-analytics*.

Visitas

Quando o utilizador entra num *website*, vai clicando, vê algumas páginas e depois sai, isso corresponde a uma visita. Por si, a visita não é muito interessante, mas a análise do número de visitas permite a segmentação de utilizadores e do seu comportamento no *website*.

Visitantes Únicos

Começar uma nova sessão num *website* conta como uma nova visita. Avaliar os visitantes únicos num *website* em particular permite revelar quantos utilizadores

utilizaram efetivamente o *website*. Este processo é conseguido através da presença de cookies, podendo portanto não ser exato.

Visualizações de Página

Uma visualização de página corresponde a cada vez que um determinado utilizador acede a uma determinada página. Se o utilizador entrar várias vezes na mesma página na mesma sessão, isso corresponde a várias visualizações de página atribuídas àquela sessão.

Páginas/Visita

Esta métrica corresponde à quantidades de páginas que são visualizadas por cada visita. Normalmente, um número mais elevado de páginas por visita é um indicador positivo, já que reflete utilizadores que passam tempo no *website* a explorá-lo, podendo ser um indicador de interesse no *website*.

Duração da Visita

Esta métrica corresponde à quantidade de tempo que um determinado utilizador navegou pelo *website*. Normalmente é utilizado algum tipo de agregação desta métrica, tipicamente a média.

Bounce Rate

O *Bounce Rate* de uma página corresponde ao quociente entre o número de utilizadores que entraram no *website* numa determinada página e saíram sem visitar qualquer outra página, e o número de utilizadores que entraram no *website*. Um *Bounce Rate* baixo é geralmente um bom indicador, embora para qualquer indicador, incluindo os anteriores, isso possa significar algo diferente conforme a situação.

% Novas Visitas

A % de novas visitas corresponde, em percentagem, ao quociente entre o número de utilizadores que entraram, mas que nunca estiveram no *website*, e o número de utilizadores total.

Esta métrica pode ser útil para avaliar o sucesso em trazer de volta utilizadores para novas visitas.

Isoladamente, estas métricas não respondem a quaisquer questões particularmente interessantes. O objetivo será atingir uma perceção mais profunda sobre o comportamento dos utilizadores, fazendo uso de agregações das métricas e dos atributos.

2.2.3 Objetivos

No contexto de *Web-Analytics*, um objetivo é determinada ação ou conjunto de ações dos utilizadores no *website* que se encontram em linha com o que é pretendido com este, e que permite reconhecê-lo como uma tarefa bem sucedida.

O processo de definição de objetivos em *web-analytics* é parte essencial na medição da qualidade de determinado *website*. Neste sentido, deve ter-se em atenção que os objetivos definidos para o *website* devem estar alinhados com o objetivo da empresa. E ainda, as métricas definidas devem permitir discriminar se os objetivos foram ou não atingidos. Portanto, para um dado conjunto de objetivos do *website*, a escolha das métricas para medir o seu sucesso é parte fundamental.

Conversões

Em *web-analytics*, uma ação que se pretende que o utilizador tome no *website* é chamada de "conversão". Contactar a equipa de vendas, comprar diretamente algum produto ou fazer download de um artigo são exemplos de "conversões", neste contexto.

Escolhendo um determinado objetivo, ou seja, um ponto final concreto, pode-se definir o comportamento do utilizador como "convertido", caso este o tenha atingido.

Portanto, a taxa de conversão é uma métrica definida como a porção de utilizadores selecionados que tomam determinada ação no *website*.

Pode ainda definir-se um conjunto de ações que constituem um caminho que leva a que um determinado objetivo seja cumprido e serem definidas etapas que indicam até que ponto desse caminho o utilizador chegou em termos de completação da tarefa pretendida.

Em termos de definição dos objetivos propriamente ditos, encontram-se quatro questões centrais que deverão ser atentamente respondidas [33].

1. Qual o propósito da empresa/organização?

2. Como encaixa o *website* nesse propósito?

3. O que pretende a empresa que os utilizadores façam no *website*?

4. Que comportamento específico mostra que os utilizadores serviram esse propósito?

Em relação à primeira e segunda questões, o próposito não deve ser exposto como "aumentar receitas"ou "fazer dinheiro". Em vez disso, deve-se enumerar quais os problemas que a empresa/organização pretende resolver e de que forma o *website* pode ajudar nessa resolução.

No que concerne a terceira questão, procura-se encontrar de que forma os objetivos dos utilizadores e os da empresa/organização se alinham. Este ponto é bastante importante para a definição dos objetivos do *website*.

Finalmente, a última questão foca-se nas métricas necessárias para avaliar a qualidade do *website* na aproximação dos objetivos da empresa/organização com os dos utilizadores.

2.2.4 *Web-Analytics* aplicado aos utilizadores

Em *Web-Analytics*, os dados recolhidos e disponíveis vão definir uma série de abordagens possíveis de serem efetuadas. Basicamente, estas fazem uso de características dos próprios utilizadores do *website* ou então do seu comportamento de interação com o mesmo *website*.

Os utilizadores de determinado *website* possuem um conjunto de características que lhes são inerentes. Estas características podem ser reunidas de forma a categorizar os utilizadores, a que se dá o nome em *Web-Analytics* de *personnas*. *Personnas* são definidos como utilizadores abstratos que representam grupos de utilizadores. Quando se pretende fazer corresponder um grupo de utilizadores a determinado comportamento, neste contexto, refere-se às tais *personnas*. Referir, por exemplo, *"Os portugueses gostam de praia"*, faz-se atribuir uma *personna* (neste caso, uma agregação por nacionalidade) a determinado comportamento (neste caso, gostar de praia).

Em *Web-Analytics*, muitas vezes, procura-se estabelecer este tipo de relação, para assim retirar informação relevante e condensada da grande quantidade de dados ao dispor. Em seguida, são apresentados os diferentes tópicos de recolha de

informação sobre os utilizadores, nomeadamente a nível das características do utilizador, análise de tráfego, análise de conteúdo e análise de caminho de cliques.

Características do Utilizador

No caso particular das características dos utilizadores, a ferramenta *Google Analytics* apresenta relatórios sobre:

- A demografia dos utilizadores: através do endereço IP, é obtida a localização do utilizador e apresentada por país, região e até cidade.
- O novo vs De regresso: é apresentada a percentagem de novos utizadores em relação ao total. Este relatório permite comparar os novos utilizadores e os utilizadores de regresso relativamente a outros atributos como taxa de conversão, páginas visitadas por utilizador, etc.
- A frequência vs Recente: é mostrada a frequência com que os utilizadores usam o *website* e também quão recente foi a última visita.
- O compromisso: esta secção contém informação acerca dos tempos de visitas e da profundidade de páginas (quantas páginas foram visitadas por visita). Algumas agregações destas métricas são também disponibilizadas.
- A tecnologia: descreve o tipo de browser, sistema operativo e tipo de dispositivo utilizado nas visitas ao *website*.

Análise de Tráfego

Outra componente que também é analisada consiste em categorizar utilizadores pelas fontes e meios que os levaram ao website. A esta abordagem dá-se o nome de análise de Tráfego e este faz uso de local específico que levou o utilizador ao *website*. Apesar de na ferramenta de *Google Analytics* serem descritas as URL ou os links que levaram os utilizadores ao *website*, normalmente este tipo de análise não é muito frequente, a não ser em casos específicos em que se queira analisar URLs em particular. O que é mostrado como tendo mais utilidade é a análise do Meio. O Meio deriva das fontes, mas em vez de descrever o local específico de onde os utilizadores vieram, recolhe informação sobre a categoria da fonte. No caso de *Google Analytics*, há 4 categorias apresentadas:

1. Orgânico: quem usa um motor de busca para chegar ao *website*;
2. Referencial: quem clica num link de outro *website*;
3. *Nenhum*: quem entra diretamente escrevendo o URL no browser;
4. CPC: referindo-se a pessoas que entram no *website* através de um link de publicidade. CPC refere-se a *"cost per click"*.

Análise de Conteúdo

No contexto da análise da interação e comportamento dos utilizadores no *website*, esta interação pode fornecer um conjunto de dados que se podem revelar interessantes. Por exemplo:

- Analisar quais as páginas do *website* que foram mais ou menos visitadas pode estar relacionado com o interesse despertado na página nos utilizadores, ou simplesmente porque é de fácil acesso a partir da página principal.
- Um *ratio* elevado entre páginas visitadas e páginas únicas visitadas poderá indicar intencionalidade, como voltar a uma página que é frequentemente atualizada ou então, poderá indicar um problema de navegação dentro do *website*. Esta interpretação depende do contexto, e deve ser avaliada para determinado caso específico.
- Tempo médio baixo para determinada página poderá indicar que o conteúdo não corresponde ao que o utilizador pretendia, está mal redigido ou não existe muito conteúdo na página. Poderá também indicar que o conteúdo da página está bem organizado e permite que os utilizadores satisfaçam os seus objetivos rapidamente, ou então que a esta é apenas um ponto de passagem para outras páginas.
- Pelo contrário, um tempo médio elevado numa determinada página tanto pode significar que os utilizadores estão a ler conteúdo extenso ou assistir a vídeos, esperar que determinado ficheiro seja descarregado ou ainda preencher um formulário extenso. Podem também estar a fazer uso de alguma funcionalidade interativa da página. No entanto, pode haver indicadores explicitamente negativos, como estarem a esforçar-se para

entender como usar a página. Este indicador deverá ser usado mais uma vez dentro de um contexto.

- Um *bounce rate* elevado (porção de pessoas que saem do *website* sem visitarem mais nenhuma página) poderá significar que os utilizadores não estão satisfeitos com a página, ou encontram dificuldades em atingir os seus objetivos com a página. Poderá indicar também que o motor de busca forneceu o *website* como uma má referência para o que as pessoas pretendiam, entre outros. Tipicamente, um *bounce rate* baixo raramente evidencia um problema em si.

- A percentagem de saídas refere-se à porção de visitas que corresponderam à última vez que o utilizador viu o *website*. Um valor elevado para este indicador tanto poderá mostrar que os utilizadores estão frustrados na página e portanto saem, como poderá implicar que os utilizadores atingiram o seu objetivo e portanto, saíram.

- Por último, encontra-se a velocidade a que as páginas carregam. Este indicador está diretamente relacionado com a qualidade da experiência do utilizador.

Análise de Caminho de Cliques

Embora seja quase impraticável para um típico *website* fazer uma caracterização de caminhos comuns de navegação entre páginas dos utilizadores (apenas praticável para *websites* bastante pequenos ou bastante lineares), a análise do caminho da navegação surge como uma abordagem poderosa para entender como os utilizadores navegam no *website*. A Figura 2.4 demonstra o caminho para apenas um utilizador e sugere a complexidade em que a análise de caminhos se pode tornar com o aumento do número de utilizadores do *website*.

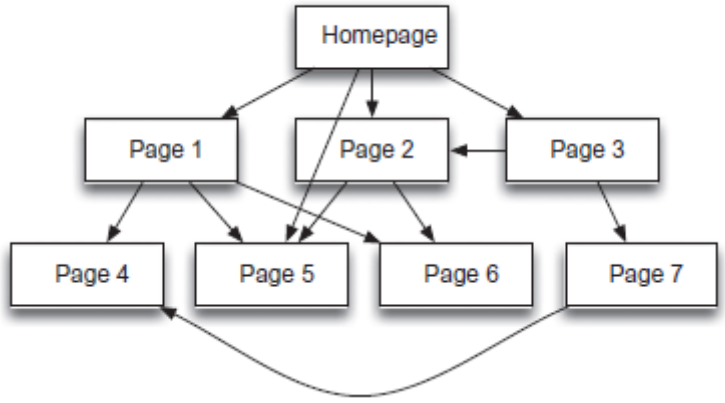


Figura 2.4: Representação dos caminhos seguidos por um utilizador num dado *website*.

Assim, o foco deve ser direcionado para, página por página, a relação entre as páginas. Ter atenção às páginas que levaram à página a ser analisada, ou às páginas que os utilizadores prosseguiram a partir da página analisada, o que se revela de bastante importância.

Concluindo, todos os atributos e métricas, assim como as diferentes abordagens, têm como objetivo auxiliar no processo de segmentação de utilizadores, quer pelos seus atributos, quer pelo seu comportamento no *website*. Este processo de segmentação permite estabelecer *personnas* específicos e/ou interessantes e concentrar os esforços para adaptar o *website* para que os utilizadores atingam os objetivos pretendidos.

2.3 Algoritmos

2.3.1 Árvores de Decisão

Uma árvore de decisão é um classificador (ou regressor) construído a partir da partição recursiva do espaço de dados. Consiste em nodos organizados numa estrutura de dados em árvore, tipicamente árvore binária, que separa o espaço de dados de acordo com uma função discreta dos atributos dos dados de entrada. O nodo principal, sem arestas de entrada, é chamado *root*. Todos os outros nodos têm uma aresta de entrada. Os nodos sem arestas de saída são chamados folhas e todos os outros são nodos internos ou de teste.

No caso mais simples e frequente, cada teste considera uma decisão por atributo, de forma a particionar o espaço de dados de entrada de acordo com os valores desse mesmo atributo, nomeadamente atributos qualitativos. No caso de atributos numéricos, a condição de teste refere-se a uma margem.

A cada folha é atribuída uma classe representando tipicamente a classe maioritária. De forma alternativa, pode-se atribuir um vetor de probabilidades, onde a cada valor do atributo alvo é atribuída uma probabilidade.

Cada caso é classificado a partir do nodo *root*, navegando pela árvore até chegar a um nodo folha, de acordo com as condições de cada nodo no caminho seguido. No nodo folha é, então, atribuída uma classe (caso de classificação) ou um valor (caso de regressão) a esse mesmo caso [35] [36].

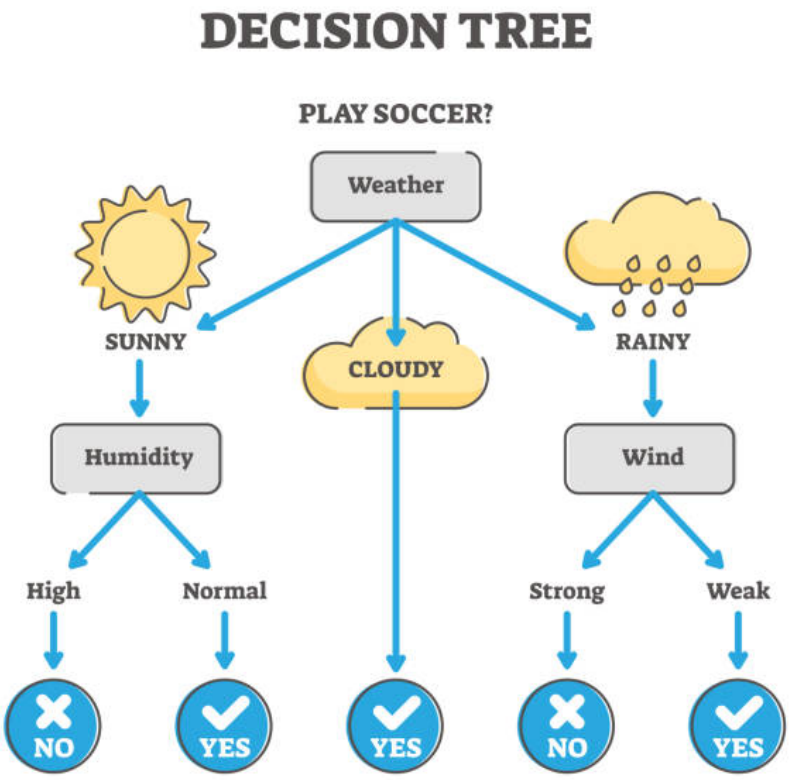


Figura 2.5: Exemplo de uma árvore de decisão.

A Figura 2.5 representa uma árvore de decisão onde se decide se em determinado dia se joga futebol, consoante alguns atributos de entrada sobre as condições climatéricas, sendo eles o estado do tempo, a humidade e a força do vento.

As caixas a cinzento representam os nodos, enquanto os círculos azuis representam as atribuições de valor para cada folha.

Dado este classificador, um analista pode prever a existência de jogo de futebol em determinado dia, seguindo o caminho dado pela árvore, ou pode então analisar o comportamento de jogo de futebol, relacionando-o com todo o domínio de atributos de entrada (interpretabilidade) [35] [36].

O desafio de um dado algoritmo para a construção de árvores de decisão torna-se, então, encontrar a árvore de decisão ótima, que minimize o erro, ou seja, uma medida de discrepância entre o observado e o previsto. No entanto, outros fatores podem ser levados em conta, como a minimização do número de nodos ou a mini-

mização da profundidade (número máximo de nodos entre o nodo raíz da árvore e as suas folhas) da árvore, por forma a reduzir a sua complexidade [35].

Obter a árvore de decisão ótima através de determinado conjunto de dados é considerada uma tarefa de complexidade computacional *NP-hard* [37] [38] [39]. Desta forma, torna-se necessário utilizar métodos heurísticos para resolver o problema. Estes métodos podem ser divididos em dois grupos: *top-down* e *bottom-up*. A abordagem *top-down* é claramente a preferível na literatura.

Algoritmos de *top-down* incluem ID3 [40], C4.5 [41], CART [42].

A construção algorítmica das árvores de decisão do tipo *top-down* é *greedy* (dividir o problema em subproblemas e escolher a opção óptima para cada um deles) por natureza, fazendo uso da recursividade (estratégia *divide-and-conquer*). Por forma a resolver o problema de seleção do atributo em cada nodo da árvore, é utilizado um critério de impureza, onde se pretende reduzir a impureza de cada partição dos dados, ou seja, pretende-se ter mais certeza sobre os valores atribuídos a cada decisão. Os investigadores usam tipicamente estratégias como o cálculo do ganho de informação ou o índice de *Gini* como critério de impureza. Estes serão explicados mais à frente.

Como demonstração, o procedimento do algoritmo ID3 (*Inductive Decision Tree*) para a construção de árvore de decisão segue os seguintes passos, onde S são os dados de treino e A são os atributos de entrada:

1. Começa com os dados S como o nodo *root*;
2. A cada iteração do algoritmo, itera sobre o subconjunto de atributos A que não foram até agora usados, onde calcula a entropia(E) e o ganho de informação (GI) para estes atributos.
3. Seleciona o atributo que maximiza o ganho de informação.
4. Os dados de entrada são separados de acordo com o atributo selecionado para produzir uma partição dos dados.
5. O algoritmo continua a recursão em cada partição dos dados, considerando apenas atributos não utilizados anteriormente.
6. O algoritmo pára quando o nodo for puro (só contiver dados correspondentes a uma classe).

Ganho de Informação

Entropia é uma medida da aleatoriedade da informação a ser processada. Quanto maior a entropia, mais difícil é retirar conclusões sobre os dados, utilizando o atributo em causa [43].

Matematicamente, entropia (E) é definida pela equação 2.1.

$$E(T) = \sum_{i=1}^{r} -p_i \, log_2 \, p_i \tag{2.1}$$

onde T é o estado atual, p_i é a probabilidade do evento i do estado T, com r eventos.

O ganho de informação (GI) é um critério que utiliza a entropia como medida da impureza [40]. Pode-se definir impureza, no contexto de árvores de decisão, como uma medida do quão frequente um elemento escolhido aleatoriamente de um dado conjunto seria erradamente classificado se este fosse classificado seguindo a distribuição de classificações dos subconjuntos. O ganho de informação é definido como:

$$GI = E(T) - E(T|X) \tag{2.2}$$

onde T é o estado atual, X é o atributo selecionado e $T|X$ representa o estado atual condicionado pelo atributo X.

Índice de *Gini*

O índice de *Gini* (G) é um outro critério baseado no cálculo da impureza.

Matematicamente, o índice de *Gini* é definido por:

$$G(T) = 1 - \sum_{i=1}^{r} p_i^2 \quad , \quad i \in \{1, ..., r\} \tag{2.3}$$

onde T é o estado atual e p_i é a probabilidade do evento i do estado T.

Consequentemente, o ganho de *Gini* (GG) é o critério de avaliação para selecionar o atributo correspondente ao nodo e é definido por:

$$GG(T, X) = G(T) - G(T|X) \tag{2.4}$$

onde T é o estado atual, X é o atributo selecionado e $T|X$ representa o estado atual condicionado pelo atributo X.

Na Figura 2.6 pode-se visualizar a representação gráfica das funções de Entropia e de Índice de *Gini* para uma decisão binária, onde a impureza se encontra normalizada entre 0 e 1.

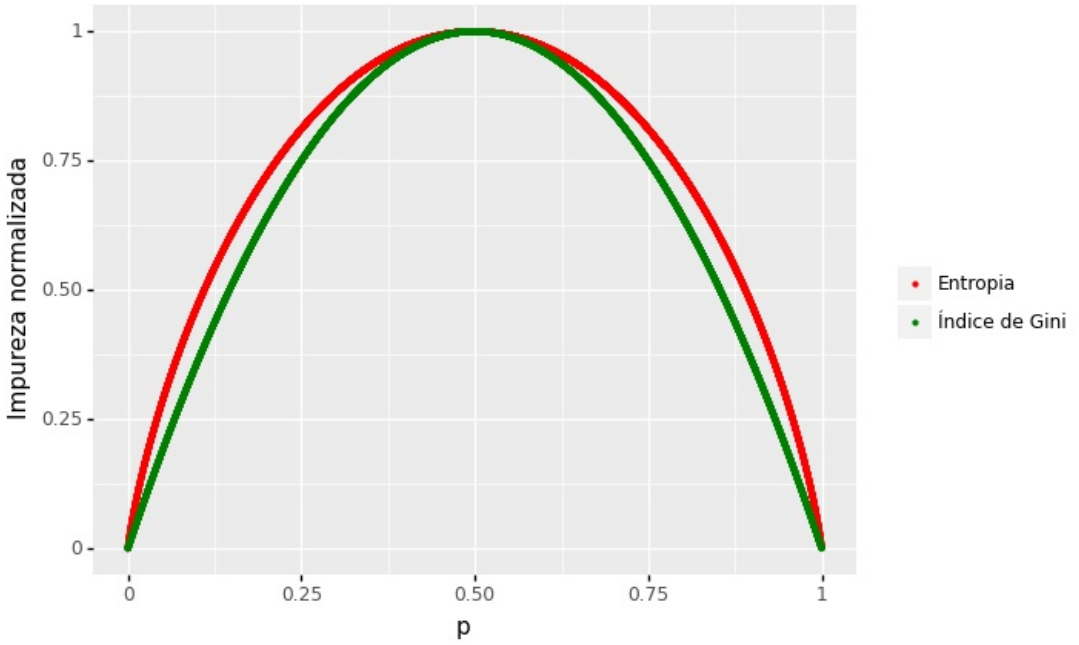


Figura 2.6: Visualização gráfica das curvas de entropia (a vermelho) e índice de *Gini* (a verde).

As árvores de decisão que são construídas até ao máximo da sua profundidade, tendem a gerar árvores grandes e complexas, sendo que tendem também a ter problemas de *overfitting* aos dados de treino, não conseguindo generalizar. Deste modo, vários métodos foram sendo propostos para abordar esta questão. Um dos mais interessantes refere-se à aplicação de critérios de paragem, realizando cortes em determinados ramos das árvores (*Pruning*). Foi mostrado que esta técnica reduz a complexidade da árvore ao mesmo tempo que melhora a sua capacidade de generalização [42].

Vários algoritmos foram sendo desenvolvidos com diferentes critérios de *pruning*, sendo alguns deles o *Cost-Complexity Pruning*, *Reduced Error Pruning*, *Minimum Error Pruning*, *Pessimistic Pruning*, entre outros [35] [36].

2.3.2 *Random Forest*

Surgindo como uma extensão do uso de árvores de decisão, as *Random Forests* fazem uso da aplicação do método de *bootstrap* agregado para através da utilização de classificadores mais fracos (árvores de decisão), construir uma classificador mais robusto, originalmente concebido como uma agregação de várias árvores CART [42].

Apesar do uso extenso de *Random Forests* na prática, o *framework* matemático que justifique o seu sucesso ainda não é bem entendido, como referido em [44]. Na verdade, o trabalho teórico inicial baseava-se bastante em intuição e heurísticas matemáticas, sendo que apenas foi formalizado rigorosamente em 2012, por Biau [45].

Random Forests são obtidas a partir da construção de diversas árvores isoladas. Estas são treinadas de forma independente e com acesso diferente ao conjunto de dados de treino (*bootstrapping*). No caso das *Forests*, as árvores não são submetidas a *pruning* [44].

Quanto à agregação dos diversos *outputs*, tipicamente as previsões das *Forests* são obtidas pela média das previsões das árvores no caso do problema de regressão, ou então através do voto maioritário (moda) em problemas de classificação.

A Figura 2.7 representa o processo de classificação de uma *Random Forest*, onde se pode visualizar a composição de várias árvores de decisão, sendo que cada uma fornece a sua classificação dos novos casos e, no final, é feita uma agregação maioritária para obter a classificação final da *Random Forest*.

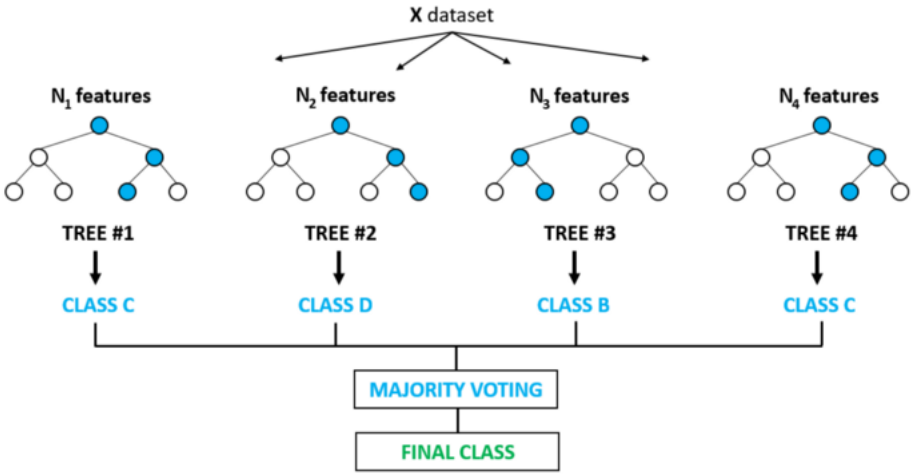


Figura 2.7: Visualização gráfica da estrutura de uma *Random Forest* e do método de previsão.

No caso da utilização das *Random Forest* para os problemas de regressão, estas apresentam limitações ao nível de extrapolação, ou seja, as *Forests* não conseguem generalizar para casos fora da margem de valores dos dados em que treinaram [46] [47].

2.3.3 *Gradient Boosting*

Gradient Boosting é um outro método de *Machine Learning* que faz uso da agregação de árvores de decisão. No entanto, é um pouco diferente das *Random Forests* descritas anteriormente. Neste caso, o algoritmo é formado pela combinação aditiva das contribuições das diferentes árvores, ou seja, a próxima árvore tenta prever os resíduos das previsões das árvores anteriores e assim sucessivamente. Uma das desvantagens aqui evidenciada é que, como cada árvore precisa da árvore anterior para ser treinada, estas não podem ser treinadas em paralelo, ao contrário das *Random Forest*. Pode-se descrever matematicamente a função de *Gradient Boosting* pela equação 2.5.

$$\hat{y}_i = \sum_{k=1}^{K} f_k(\mathbf{x}_i) \quad , \quad i \in \{1, ..., m\} \tag{2.5}$$

para um determinado *dataset* $D = \{(\mathbf{x}_i, y_i), i \in \{1, ..., m\}\}$, em que $\mathbf{x}_i$ são os

dados explicativos para determinado registo i, y_i são os labels associados a esse $\mathbf{x}_i$, e $\hat{y}_i$ são as previsões do modelo associados a esse $\mathbf{x}_i$, sendo m o número de registos total, K o número de árvores de regressão utilizadas e $f_k \in F$, com F o espaço de todas as árvores de regressão possíveis (CART) [48].

Desta forma, e pela equação 2.5, observa-se que cada árvore vai tentando corrigir os erros das árvores anteriores e o resultado final é a contribuição aditiva de todas as árvores. O termo *gradient* provém do facto do algoritmo utilizar a técnica de *gradient descent* para encontrar os parâmetros que minimizam a função erro a cada passo [49].

2.3.4 Análise de *Clustering*

Análise de *Cluster* é um tipo de análise de dados que procura agrupar objetos com base nas relações entre estes, estimando uma estrutura a partir de um conjunto de características selecionadas dos objetos. É usado como um nome genérico para um grupo de técnicas de análise de dados multivariada, tendo como representante de cada agrupamento (*cluster*) um ponto no espaço dos dados.

Na Figura 2.8, pode-se ver representado um exemplo da técnica de *clustering* aplicado a um conjunto de objetos (pontos) num espaço de duas dimensões. As diferentes cores referem-se às diferentes agregações, neste caso 3 agregações. Os pontos negros referem-se aos *clusters* (representação matemática do grupo) [50].

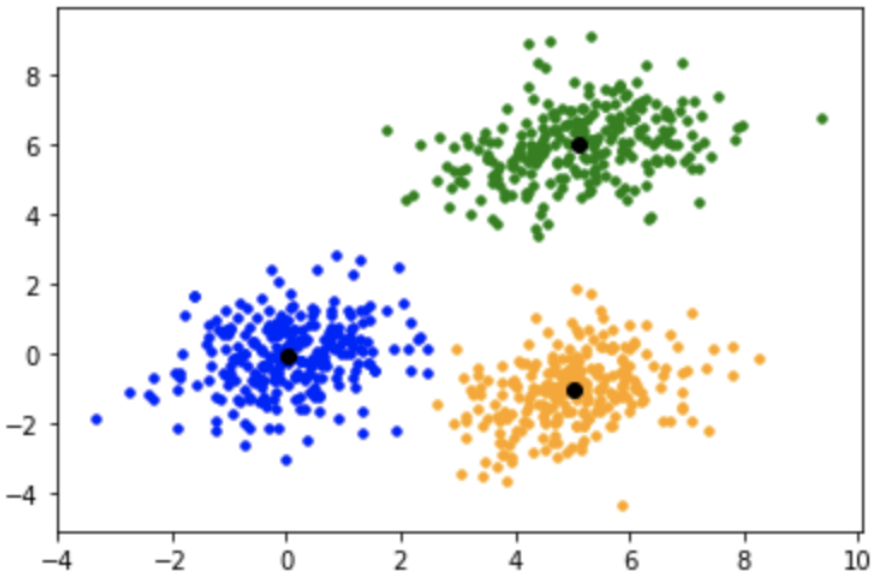


Figura 2.8: Exemplo de clustering aplicado a um conjunto de dados a duas dimensões.

A análise de *clusters* pretende imitar a tarefa bastante humana de reconhecer padrões num conjunto de dados, sendo particularmente útil em situações onde estão presentes enormes quantidades de dados, sendo que esta enormidade dificulta a tarefa a um ser humano. Fazendo uso de técnicas de *clustering*, a quantidade de dados pode ser reduzida para um número simbólico, mas ainda assim representativo dos dados, permitindo depois formular hipóteses sobre a estrutura do problema em questão. A seguir são apresentadas breves descrições de dois algoritmos de *clustering*, nomeadamente o *k-means* e o *k-medoids*.

K-Means

Em *clustering*, *k-means* é um algoritmo usado frequentemente para agrupar um dataset em k grupos, que encontrando as partições de forma a que o erro quadrático (E^*) entre a média empírica do *cluster* e os pontos do *cluster* é minimizada [51] [52]:

$$E^*(C) = \sum_{k=1}^{K} \sum_{x_i \in c_k} ||x_i - \mu_k||^2 \tag{2.6}$$

com $X = \{x_i\}, i = 1, ..., n$ o *set* de n pontos de dimensão d a serem agrupados num *set* de K *clusters*, $C = \{c_k, k = 1, ..., K\}$ e $E^*(C)$ a soma do quadrado dos erros sobre todos os *clusters* K.

O algoritmo de *k-means* necessita de 3 parâmetros iniciais: número de *clusters* K, inicialização dos *clusters* e a métrica de distância. Tipicamente, a métrica de distância utilizada é a métrica euclidiana, sendo que outras podem também ser usadas, como a distância de *Manhattan* [53] [54].

O procedimento segue os seguintes passos, cuja representação se encontra inclusive na Figura 2.9 [52]:

1. Selecionar uma partição inicial para os *clusters* K.
2. Repetir até que os *clusters* estabilizem:
   (a) Gerar uma nova partição, atribuindo cada elemento ao seu *cluster* mais próximo.
   (b) Calcular novos centros dos *clusters* (*centroids* - posição média de todos os pontos).

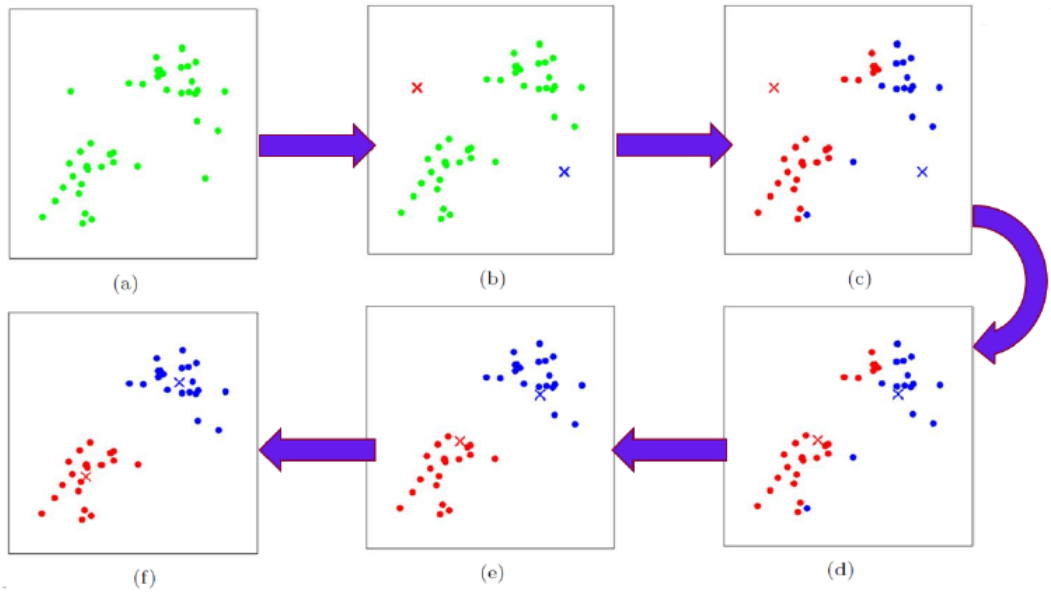


Figura 2.9: Procedimento algorítmico de *K-means clustering*.

Na Figura 2.9, começa-se com a distribuição dos dados em *a)*, a inicialização dos 2 *clusters* em *b)*, atribuição a cada elemento dos dados o seu *cluster* mais próximo em *c)*, calcular o ponto médio de cada conjunto de dados para cada partição em *d)*, aplicar recursivamente os passos *c) e d)* e concluir em *f* quando os *clusters* estabilizarem, ou seja, não haver alterações nos *clusters* em relação à iteração anterior.

K-Medoids

Seguindo uma lógica similar à do algoritmo *k-means*, o algoritmo *k-medoids* pretende segmentar os dados em *k clusters*, sendo que neste caso, os representantes dos *clusters* não são *centroids*, mas *medoids*, ou seja, os representantes dos *clusters* têm de pertencer ao conjunto de dados [55].
O algoritmo segue os seguintes passos:

1. Inicializar k pontos do conjunto de dados como *medoids* para reduzir o erro.
2. Associar cada ponto ao *medoid* mais próximo.
3. Repetir enquanto o erro da configuração decresce:
   (a) Para cada *medoid* m e para cada ponto não-*medoid* o:
      i. Considerar a troca de m por o, e calcular o erro da mudança.
      ii. Se o erro da mudança é o melhor encontrado, guardar este m e a combinação de o.
   (b) Executar a mudança do melhor m e o melhor o, se isso diminuir a função erro. De outra forma, o algoritmo termina.

Capítulo 3

Descrição de Dados

3.1 *Nexthink*

A *Nexthink* é uma empresa Suíça que desenvolveu uma plataforma de experiência digital, que combina monitorização, envolvimento do utilizador, processos analíticos e automação, tudo numa perspetiva do empregador.

O processo começa com a instalação de um leve coletor nos dispositivos dos utilizadores. Este coleta informação de atividades e métricas sobre questões de performance, utilização de *software*, browser requests, quebras de política, conexões de rede, execução de programas, entre centenas de outros parâmetros que juntos, pretendem contar a história da utilização e experiência do utilizador [56]. A própria *Nexthink* proporciona suporte para analisar estes dados recolhidos.

Nesta dissertação, foram utilizados os dados de exemplo proporcionados pela Nexthink, de forma a analisar abordagens possíveis e retirar informação relevante que possa ajudar a melhorar a experiência dos utilizadores.

Quanto aos dados brutos propriamente ditos, foram adquiridos 39 ficheiros em formato .csv, organizados por tabelas de dados. Dentro destes 39, 13 referiam-se a dados sobre os objetos, 13 sobre os eventos ocorridos e outros 13 referentes aos objetos associados a cada evento. Estes últimos continham informação que relacionava os id's dos eventos com os id's dos objetos envolvidos nos eventos.

As tabelas dos dados foram adquiridas de *queries* realizadas através do sistema NXQL *Data Model*. A *Nexthink Query Language* (NXQL) é uma linguagem desenhada para realizar queries à base de dados *in-memory* do Motor da *Nexthink* via *Web API V2*. A linguagem é baseada em SQL, mas otimizada de modo a melhorar a rapidez, a complexidade e o controlo das queries a serem realizadas [57]. Este sistema é similar ao sistema SQL para armazenamento de dados e, portanto, assim que as tabelas se tornaram disponíveis para tratamento de dados, estas foram armazenadas numa Base de Dados SQLite3, com as respetivas relações respeitadas.

3.2 Dados Utilizados na Análise

Quanto aos dados utilizados no tratamento e análise, as diferentes abordagens utilizaram apenas parte das tabelas adquiridas. A descrição destas tabelas de dados encontra-se nas tabelas seguintes de 3.1 a 3.13 [58]. As tabelas de 3.1 a 3.3 descrevem as tabelas de dados referentes à informação sobre os objetos envolvidos na recolha de dados. As tabelas de 3.4 a 3.8 descrevem as tabelas de dados referentes à informação

sobre os eventos envolvidos na recolha de dados. Enquanto as tabelas de 3.9 a 3.13 descrevem as tabelas de dados que se referem à informação que permite relacionar os eventos com os objetos envolvidos nos respetivos eventos.

Os eventos podem ser divididos em:

- *execution*, com informação sobre cada execução de determinada *application*, por determinado *user* em determinado *device*;
- *execution_error* e *execution_warning*, com informação acerca dos erros e *warnings* durante determinada execução numa dada *application*;
- *device_error* e *device_warning*, com informação acerca dos erros e *warnings* referentes a determinado *device*.

Tabela 3.1: Descrição da Tabela *"User"*.

Tabela: *User*		
A user is an object that represents an individual account in a device (local user) or in a group of devices (domain user). The account may identify a physical user or a system user.		
Atributo	**Tipo**	**Descrição**
Id	*Identifier*	*Unique application identifier*
Full_name	*String*	*Full user name as listed in active directory*
Job_title	*String*	*Job title as listed in active directory*
Department	*String*	*User department as listed in active directory*
Total_active_days	*Day*	*Total number of days the user was active*

Tabela 3.2: Descrição da Tabela *"Device"*.

Tabela: *Device*		
A device is Windows physical or virtual machine monitored by a Nexthink Collector.		
Atributo	**Tipo**	**Descrição**
Id	*Identifier*	*Unique device identifier*
Entity	*String*	*Entity*
Device_type	*Enum*	*Type of device*
Cpu_frequency	*Mhz*	*CPU frequency*
Cpu_model	*String*	*CPU model*
Hard_disks	*String*	*List of all hard disks*
Device_manufacturer	*String*	*Indicates the device manufacturer*
Os_architecture	*Enum*	*Architecture of device operating system (x86/x64)*
Total_active_days	*Day*	*Total number of days the device was active*
Os_version_and_architecture	*String*	*Indicates name, version and architecture (when applicable) of the operating system*
Total_ram	*Byte*	*Total amount of RAM*
System_drive_free_space	*Byte*	*Total available free space on system drive*
System_drive_capacity	*Byte*	*Total capacity of system drive*
System_drive_usage	*Percent*	*Use percentage of system drive*
platform	*Enum*	*Indicates the platform of the device*

Tabela 3.3: Descrição da Tabela *"Application"*.

Tabela: *Application*		
An application is a set of executables e.g. 'Microsoft Office'.		
Atributo	**Tipo**	**Descrição**
Id	*Identifier*	*Unique application identifier*
Platform	*String*	*The platform on which the application is running.*
Company	*String*	*Company producing the application*
Name	*String*	*Application name*
Total_active_days	*Day*	*Total number of days the application was active*

Tabela 3.4: Descrição da Tabela *"Execution"*.

Tabela: *Execution*		
An execution is a process executing on a device. Serveral executions of the same process are merged when in close succession.		
Atributo	**Tipo**	**Descrição**
Id	*Identifier*	*Unique execution identifier*
Total_cpu_time	*Millisecond*	*Total CPU time*
Average_memory_usage	*Byte*	*Average memory usage*
Cardinality	*Integer*	*Number of underlying processes, consolidated over time*
Duration	*Millisecond*	*Total execution duration*
Start_time	*Datetime*	*Execution start time*
End_time	*Datetime*	*Execution end time*
Privilege_level	*Enum*	*Privilege level of the execution*

Tabela 3.5: Descrição da Tabela *"Device_error"*.

Tabela: *Device_error*		
A device_error is a critical system errors (system crash, hard reset, or disk error).		
Atributo	**Tipo**	**Descrição**
Id	*Identifier*	*Problem identifier*
Start_time	*Datetime*	*Time of error*
Type	*Enum*	*Indicates the device error type*
Error_label	*String*	*Error label*

Tabela 3.6: Descrição da Tabela *"Device_warning"*.

Tabela: *Device_warning*		
A device_warning is a peak in device resource usage (CPU, memory or I/O).		
Atributo	**Tipo**	**Descrição**
Id	*Identifier*	*Unique performance event identifier*
Duration	*Millisecond*	*Performance event duration*
Start_time	*Datetime*	*Performance event start time*
End_time	*Datetime*	*Performance event end time*
Type	*Enum*	*Type of the device warning*

Tabela 3.7: Descrição da Tabela *"Execution_error"*.

Tabela: *Execution_error*		
An execution_error is application errors (crash or not responding).		
Atributo	**Tipo**	**Descrição**
Id	*Identifier*	*Error identifier*
Type	*Enum*	*Type of the execution error*
Time	*Datetime*	*Time of error*

Tabela 3.8: Descrição da Tabela *"Execution_warning"*.

Tabela: *Execution_warning*		
An execution_warning is a peak in application resource usage (CPU or memory).		
Atributo	**Tipo**	**Descrição**
Id	*Identifier*	*Unique performance event identifier*
Start_time	*Datetime*	*Performance event start time*
End_time	*Datetime*	*Performance event end time*
Warning_duration	*Millisecond*	*Indicates the duration of the warning*
Type	*Enum*	*Type of the execution warning*

Tabela 3.9: Descrição da Tabela *"Execution_relations"*.

Tabela: *Execution_relations*	
Atributo	**Tipo**
Execution_id	*Identifier*
Device_id	*Identifier*
User_id	*Identifier*
Application_id	*Identifier*

Tabela 3.10: Descrição da Tabela *"Device_error_relations"*.

Tabela: *Device_error_relations*	
Atributo	**Tipo**
Device_error_id	*Identifier*
Device_id	*Identifier*

Tabela 3.11: Descrição da Tabela *"Device_warning_relations"*.

Tabela: *Device_warning_relations*	
Atributo	**Tipo**
Device_warning_id	*Identifier*
Device_id	*Identifier*

Tabela 3.12: Descrição da Tabela *"Execution_error_relations"*.

Tabela: *Execution_error_relations*	
Atributo	**Tipo**
Execution_error_id	*Identifier*
Device_id	*Identifier*
User_id	*Identifier*
Application_id	*Identifier*

Tabela 3.13: Descrição da Tabela *"Execution_warning_relations"*.

Tabela: *Execution_warning_relations*	
Atributo	**Tipo**
Execution_warning_id	*Identifier*
Device_id	*Identifier*
User_id	*Identifier*
Application_id	*Identifier*

3.3 Conexão do Estado de Arte com os Dados

3.3.1 Otimização energética

Em relação à exploração da otimização energética, foi estudado na secção do Estado de Arte a sua aplicabilidade a Centro de Dados e a infraestruturas IT de larga escala. Entretanto, como descrito na secção introdutória, a *Fujitsu* tem uma posição forte no mercado português, tanto no setor público de Administração como nos setores privados no Retalho, na Banca e nos Transportes. Deste modo, poderia ser interessante uma abordagem neste sentido às infraestruturas da *Fujitsu*, fazendo uso das suas infraestruturas para o tratamento desta enormidade de dados. No entanto, não foi possível implementar tal análise com os dados fornecidos da *Nexthink*, pois estes não possuem dados relativos à informação energética ou custo computacional dos comportamentos associado às execuções e/ou aos utilizadores. Fica inclusive a dúvida se uma análise a infraestruturas de média ou pequena dimensão, ao nível da energia, teria a escalabilidade suficiente para obter resultados interessantes neste domínio.

Ainda assim, dados energéticos sobre os diferentes elementos envolvidos no processo de determinada empresa, integrados com as métricas dos próprios processos (utilização de cpu, memória computacional, tempo dos processos, quão bem foram atingidos os objetivos, assim como o impacto na experiência utilizador), poderia permitir uma análise de modo a discernir os processos eficientes dos ineficientes e resultar em sugestões sobre mudanças de comportamentos ou mesmo nas infraestruturas. Esta abordagem, partindo do princípio que obteria resultados conclusivos, permitiria realizar uma melhor gestão das tarefas computacionalmente mais exigentes, abrindo espaço para uma maior disponibilidade de recursos para os utilizadores, ao mesmo tempo que restringiria os custos para a empresa.

Em relação à *User-Experience* propriamente dita, seria necessário fazer a ponte entre a informação energética de uma organização e o seu impacto na experiência dos utilizadores, de modo a validar esta associação.

3.3.2 *Web-Analytics*

No caso da abordagem de *Web-Analytics*, e dentro do âmbito de consultoria que a *Fujitsu* estabelece, poder-se-ia aplicar esta abordagem num contexto em que

há interação direta entre os dispositivos e os utilizadores. Um exemplo óbvio seria a aplicação de *web-analytics* numa empresa que fizesse uso de compras *online*, onde se poderia obter informação acerca dos clientes e dos seus comportamentos.

Como outro exemplo, podemos incluir a análise a um conjunto de lojas do *Burger King* ou *McDonalds*, onde existe uma interação entre os empregados da receção e os dispositivos de registo de pedidos ou então os próprios dispositivos *touch* de registo das encomendas, desta vez realizados pelos próprios clientes. Neste último caso, a abordagem que tipicamente é realizada para segmentação de clientes *online* e o seu comportamento nas páginas *web*, poderia ser aplicada aos clientes que interagem com os ecrãs de encomenda, já que nestes é estabelecida uma *interface* de interação, e onde é possível guardar os dados referentes a esta interação.

Dentro desta abordagem de *web-analytics*, métricas que foram identificadas como importantes inclui-se a informação relativa aos elementos do *website*, como o número de visitas, visitantes, os visitantes únicos, as *pageviews*, duração da visita e duração das páginas abertas. Inclui-se também informação referente aos utilizadores como a sua demografia, frequência de utilização, duração das visitas, profundidade da visita na árvore das páginas, *browser* e *OS* e páginas de saída. Estas métricas podem ser obtidas também dentro de um registo mais profundo de dados em que realmente são armazenados os dados referentes a cada interação realizada. Ainda dentro das métricas, pode-se enumerar algumas métricas de *alvo*, que permitem estabelecer quão bom foi a concretização de objetivos específicos, onde se incluem métricas como a conversão (se determinado produto foi comprado, por exemplo), ou sobre a própria avaliação direta da experiência do utilizador.

Esta abordagem traria valor ao permitir discernir a eficácia das *interfaces* nos objetivos, a disposição dos elementos na *interface*, os produtos com mais sucesso ou sugestões para diferentes tipos de clientes com base numa segmentação.

No contexto dos dados da *Nexthink* analisados nesta dissertação, a abordagem *web-analytics* é utilizada na segmentação dos utilizadores ou dispositivos (criação de *personnas*) com base nas aplicações que foram sendo executadas (o *dataset* não contém informação direta sobre as aplicações instaladas nos dispositivos). Para além disso, os dados fornecem informação acerca dos eventos que vão ocorrendo em relação à sua utilização, como os erros de dispositivo ou de execução (bem como os *warnings*), permitindo assim agrupar os vários elementos envolvidos de forma a identificar as fontes de tais problemas (de erros e *warnings*) e assim, melhorar a experiência dos utilizadores e reduzir os custos para a empresa envolvida.

Capítulo 4

Análise de Dados e Resultados

4.1 Estatística Descritiva

Nesta secção, descrever-se-ão os dados de um ponto de vista estatístico. São exibidas várias agregações acerca dos dados (média, mediana, mínimo e máximo), bem como o número de registos, o número de dados em falta e ainda o número de campos únicos dentro de cada categoria.

Da Tabela 4.1 à 4.3 são apresentadas as estatísticas descritivas referentes aos objetos dos dados. As Tabelas 4.4, 4.6, 4.8, 4.10 e 4.12 contêm as estatísticas dos dados dos eventos, e as Tabelas 4.5, 4.7, 4.9, 4.11 e 4.13 representam as estatísticas para os dados que permitem estabelecer as relações entre as tabelas dos objetos e as tabelas dos eventos.

Tabela 4.1: Estatística da Tabela *"Application"*.

Application	Registos	*Missing*	Únicos	Média	Mínimo	Mediana	Máximo
id	1684	0	nan	nan	nan	nan	nan
name	1684	0	1588	nan	nan	nan	nan
company	1684	0	693	nan	nan	nan	nan
platform	1684	0	2	nan	nan	nan	nan
total_active_days	1684	0	nan	15.27	0	14	34

Tabela 4.2: Estatística da Tabela *"Device"*.

Device	Registos	Únicos	Mínimo	Mediana	Máximo
id	1327	nan	nan	nan	nan
entity	1327	28	nan	nan	nan
device_type	1327	3	nan	nan	nan
cpu_frequency	1327	nan	1100 Mhz	3401 Mhz	4200 Mhz
cpu_model	1327	60	nan	nan	nan
hard_disks	1348	125	nan	nan	nan
device_manufacturer	1327	9	nan	nan	nan
os_architecture	1327	2	nan	nan	nan
total_active_days	1327	nan	0	6	34
os_version_and_architecture	1327	26	nan	nan	nan
total_ram	1327	nan	2.00 Gb	8.00 Gb	64.00 Gb
system_drive_free_space	1327	nan	637.28 Mb	316.21 Gb	877.05 Gb
system_drive_capacity	1327	nan	31.90 Gb	372.53 Gb	1044.12 Gb
system_drive_usage	1327	nan	0.06	0.15	1.0
platform	1327	2	nan	nan	nan

Tabela 4.3: Estatística da Tabela *"User"*.

User	Registos	*Missing*	Únicos	Média	Mínimo	Mediana	Máximo
id	5038	0	nan	nan	nan	nan	nan
full_name	5038	36	4607	nan	nan	nan	nan
job_title	5038	4994	14	nan	nan	nan	nan
department	5038	4994	3	nan	nan	nan	nan
total_active_days	5038	0	nan	15.44	0	16	34

As Tabelas 4.1, 4.2 e 4.3, referentes aos dados sobre os objetos, permitem observar que existem 1684 *applications*, 1327 *devices* e 5038 *users* em registo. Pela coluna "Únicos", pode-se verificar que existem atributos que se repetem para várias *applications*, *devices* e *users*, como por exemplo para a Tabela de *Application*, o nome, a empresa e a plataforma, que não têm *missing values*.

Tabela 4.4: Estatística da Tabela *"Device_error"*.

Device_error	Registos	Únicos	Mínimo	Mediana	Máximo
id	881	nan	nan	nan	nan
start_time	881	nan	2019-08-04	2019-08-29	2019-09-07
type	881	3	nan	nan	nan
error_label	881	7	nan	nan	nan

Tabela 4.5: Estatística da Tabela *"Device_error_relations"*.

Device_error_relations	Registos	Únicos
device_error_id	814	814
device_id	814	182

Tabela 4.6: Estatística da Tabela *"Device_warning"*.

Device_warning	Registos	Únicos	Média	Mínimo	Mediana	Máximo
id	178527	nan	nan	nan	nan	nan
duration	178527	nan	9.95 min	29.99 s	59.99 s	23.58 hrs
start_time	178527	nan	nan	2019-08-04	2019-08-24	2019-09-07
end_time	178527	nan	nan	2019-08-04	2019-08-24	2019-09-07
type	178527	5	nan	nan	nan	nan

Tabela 4.7: Estatística da Tabela *"Device_warning_relations"*.

Device_warning_relations	Registos	Únicos
device_warning_id	178527	178527
device_id	178527	642

Tabela 4.8: Estatística da Tabela *"Execution_error"*.

Execution_error	Registos	Únicos	Mínimo	Mediana	Máximo
id	7204	nan	nan	nan	nan
type	7204	2	nan	nan	nan
time	7204	nan	2019-08-04	2019-08-23	2019-09-07

Tabela 4.9: Estatística da Tabela *"Execution_error_relations"*.

Execution_error_relations	Registos	Únicos
execution_error_id	7204	7204
application_id	7204	160
device_id	7204	551
user_id	7204	521

Tabela 4.10: Estatística da Tabela *"Execution_warning"*.

Execution_warning	Registos	Únicos	Média	Mínimo	Mediana	Máximo
id	170138	nan	nan	nan	nan	nan
start_time	170138	nan	nan	2019-08-04	2019-08-23	2019-09-07
end_time	170138	nan	nan	2019-08-04	2019-08-23	2019-09-07
warning_duration	170138	nan	20.93 min	0 ms	89.99 s	23.58 hrs
type	170138	2	nan	nan	nan	nan

Tabela 4.11: Estatística da Tabela *"Execution_warning_relations"*.

Execution_warning_relations	Registos	Únicos
execution_warning_id	170138	170138
application_id	170138	364
device_id	170138	658
user_id	170138	836

Tabela 4.12: Estatística da Tabela *"Execution"*.

Execution	Registos	Únicos	Média	Mínimo	Mediana	Máximo
id	6805700	nan	nan	nan	nan	nan
total_cpu_time	6805700	nan	6.4 s	0 ms	0 ms	5.29 days
average_memory_usage	6805700	nan	8.18 Mb	0.00 bytes	0.00 bytes	15.97 Gb
cardinality	6805700	nan	5.43	1.0	1.0	64433.0
duration	6805700	nan	99.26 min	0 ms	7.05 min	17.72 days
start_time	6805700	nan	nan	2019-08-04	2019-08-24	2019-09-07
end_time	6805700	nan	nan	2019-08-04	2019-08-24	2019-09-07
privilege_level	6805700	4	nan	nan	nan	nan

Tabela 4.13: Estatística da Tabela *"Execution_relations"*.

Execution_relations	Registos	Únicos
execution_id	4173229	4173229
application_id	4173229	1417
device_id	4173229	1204
user_id	4173229	1855

Em relação às tabelas dos eventos (4.4, 4.6, 4.8, 4.10 e 4.12), os eventos registados encontram-se entre o dia 2019-08-04 e 2019-09-07 (Tabela 4.12), sendo que neste período, houve 881 erros de *device*, 178527 *warnings* de *device*, 7204 erros de *execution* e 170138 *warnings* de *execution*, num total de 6805700 *executions*.

No caso de erros de *device*, a Tabela 4.5 é a tabela que permite relacionar esses erros com o *device* em que se deu cada erro. Comparando as Tabelas 4.4 e 4.5, pode-se verificar que estas não contêm o mesmo número de registos. Isto significa definitivamente que faltam registos de dados, violando, portanto, a consistência destes mesmos dados. O mesmo acontece para as Tabelas 4.12 e 4.13, em que a tabela *Execution_relations* tem menos de dois terços dos registos encontrados na tabela *Execution*. Esta observação implica uma limitação razoável em relação à análise que poderá ser realizada, especialmente à sua validade. No entanto, optou-se por excluir os registos de eventos, que apesar de se encontrarem nas tabelas de eventos, não se encontram na tabela de relações. Quanto às restantes tabelas, apesar de conterem o mesmo número de eventos, não foi possível confirmar se contêm a totalidade dos registos.

4.2 Visualização dos Dados

Nesta secção, serão apresentados vários gráficos que representam a descrição dos erros e dos *warnings* de *device* e *execution* agregados em tempo, *device*, *application* e *user*.

4.2.1 Eventos de *Device*

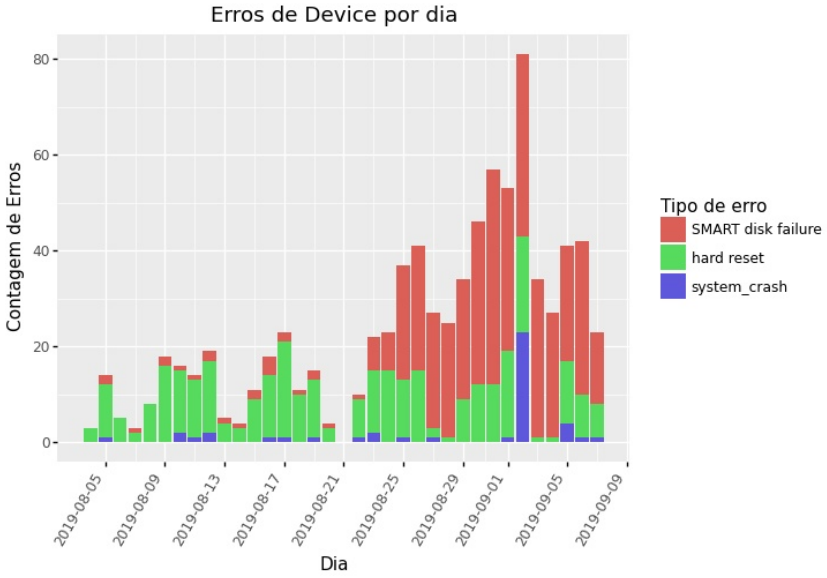


Figura 4.1: Gráfico de barras com os erros de *device* por dia para cada tipo de *erro*.

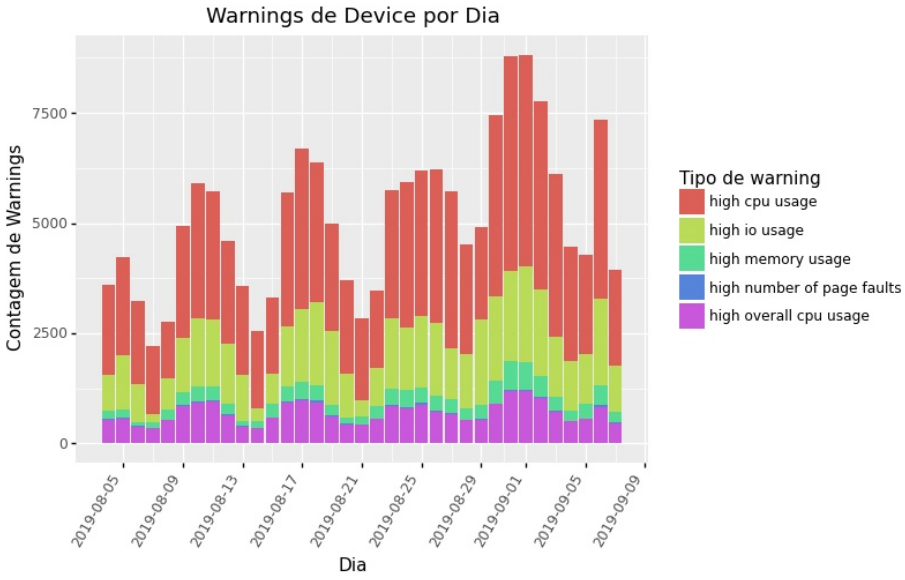


Figura 4.2: Gráfico de barras com os *warnings* de *device* por dia para cada tipo de *warning*.

Nos gráficos das Figuras 4.1 e 4.2, pode-se observar o número de erros e *warnings* de *device* em cada dia, por tipo (de erro e *warning*).

Em ambas as figuras é possível ver que aos fins de semana, o número de erros e *warning* tende a diminuir, sendo que este comportamento é natural numa empresa, e segue o esperado. No caso dos *warnings*, há uma tendência para ligeira subida do número de casos ao longo do tempo. Por outro lado, no caso dos erros da Figura 4.1, há uma subida acentuada de erros a partir do dia 2019-08-23, em particular, de erros *SMART disk failure*. Também se observa uma grande quantidade de erros do tipo *system_crash* no dia 2019-09-02.

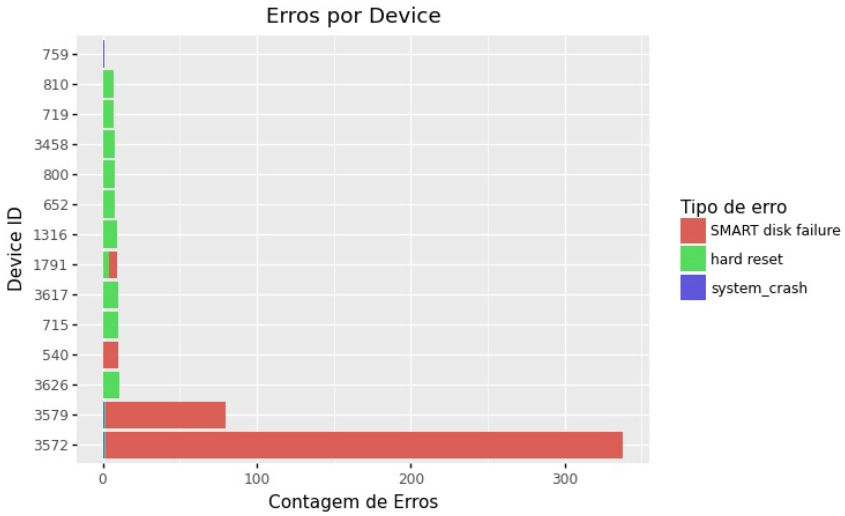


Figura 4.3: Gráfico de barras com os erros de *device* para os 14 *devices* com mais erros, por tipo de erro.

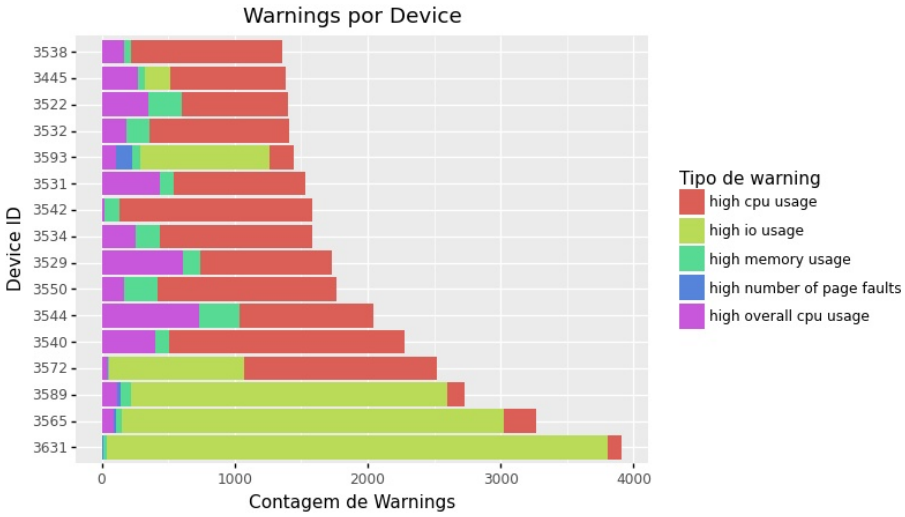


Figura 4.4: Gráfico de barras com os *warnings* de *device* para os 16 *devices* com mais *warnings*, por tipo de *warning*.

Nas Figuras 4.3 e 4.4 estão representados os gráficos de barras para os erros e *warnings* de *device* para os 16 *devices* com mais eventos, por tipo de evento. Pela Figura 4.3, pode-se ver que 2 *devices* contêm a larga maioria dos erros, sendo que estes se incluem nos erros de *SMART disk failure*. De facto, depois de uma análise mais rigorosa, chegou-se à conclusão que estes dois *devices* (ids: 3579 e 3572) são os responsáveis pelos erros de *SMART disk failure* dos dias da segunda metade do tempo analisado na Figura 4.1. Sendo assim, esta observação poderá implicar que os *hard disks* destes dois dispositivos poderão estar danificados e a necessitar de ser substituídos, embora também se tenha de ter em atenção o tempo de execução de cada um dos dispositivos.

Quanto à Figura 4.4, a distribuição de *warnings* apresenta-se relativamente uniforme nos *devices* com mais *warnings*. Nota-se também que o tipo de *warnings* mais habitual para os 3 *devices* com mais *warnings* são *warnings* de input/output (io), sendo que contam como a quase totalidade dos *warnings* para estes *devices*, sendo que este gráfico não permite estabelecer a relação entre *warnings* de *io* e de *memory*, o que poderia indicar um problema com os *hard_disks*, já que quando há uma utilização intensiva de memória, o *device* tende a fazer uso de memória virtual, o que implica ler e escrever no disco consistentemente.

4.2.2 Eventos de *Execution*

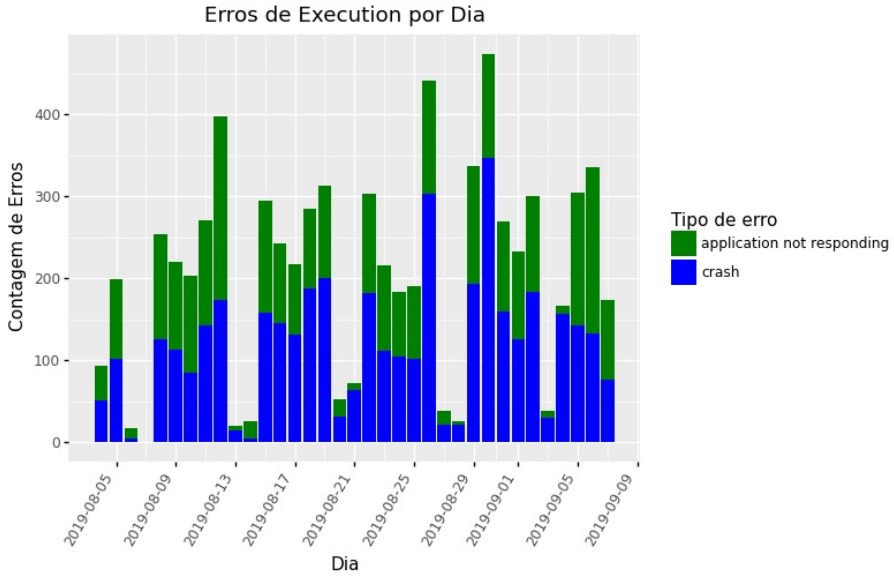


Figura 4.5: Gráfico de barras com os erros de *execution* por dia, por tipo de erro.

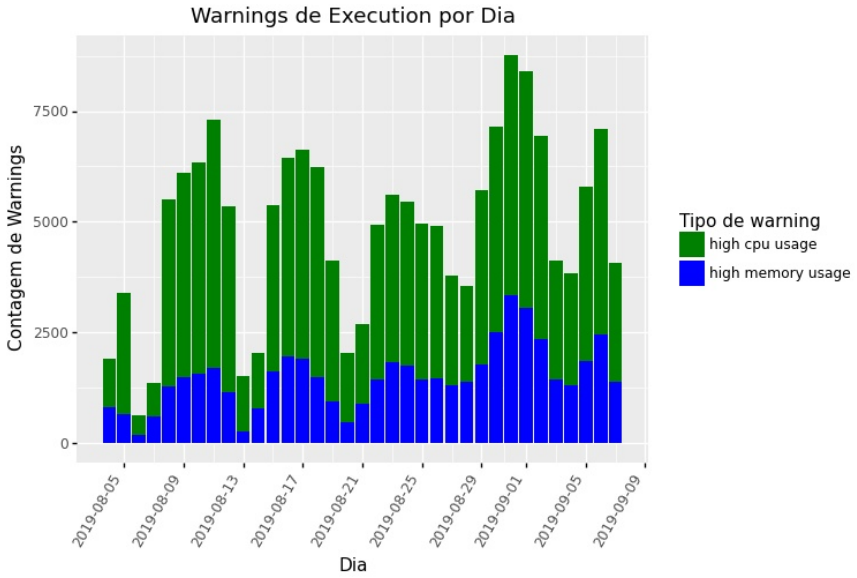


Figura 4.6: Gráfico de barras com os *warnings* de *execution* por dia, por tipo de *warning*.

Nas Figuras 4.5 e 4.6 são apresentados os gráficos de barras para os erros e os *warnings* de *execution* para cada dia. Mais uma vez, pode-se ver a variação dos

eventos para os dias de fim de semana, sendo que para os eventos de *execution*, e retirando a sazonalidade semanal, há uma distribuição bastante uniforme durante o tempo.

Figura 4.7: Gráfico de barras com os erros de *execution* por *application*, por tipo de *erro*, para as 20 *applications* com mais erros.

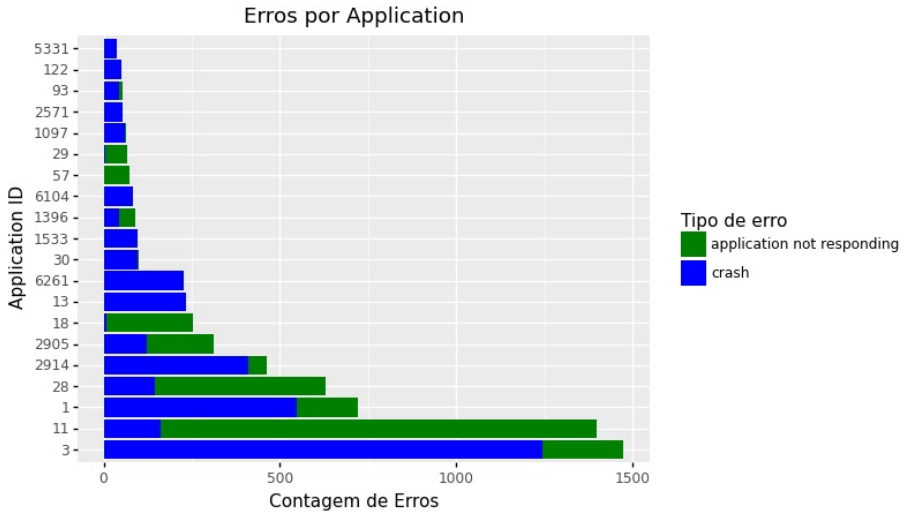
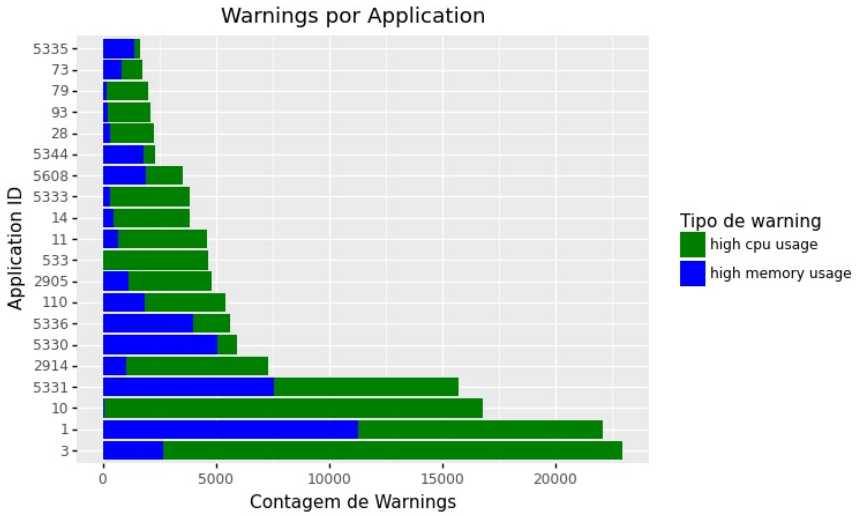


Figura 4.8: Gráfico de barras com os *warnings* de *execution* por *application*, por tipo de *warning*, para as 20 *applications* com mais warnings.



Nas Figuras 4.7 e 4.8 são apresentados os gráficos de barras para os eventos de *execution* por *application*, por tipo de *warning*.

Figura 4.9: Gráfico de barras com os erros de *execution* por *user*, por tipo de erro, para os 20 *users* com mais erros.

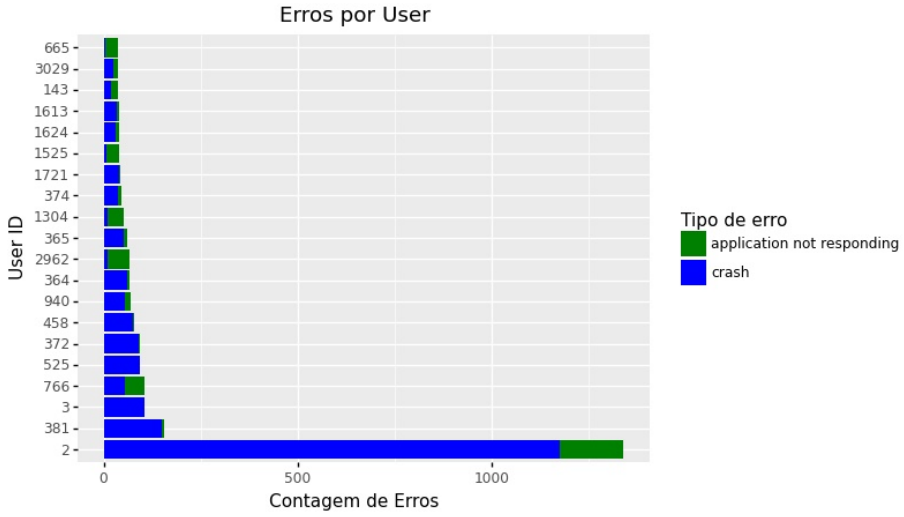
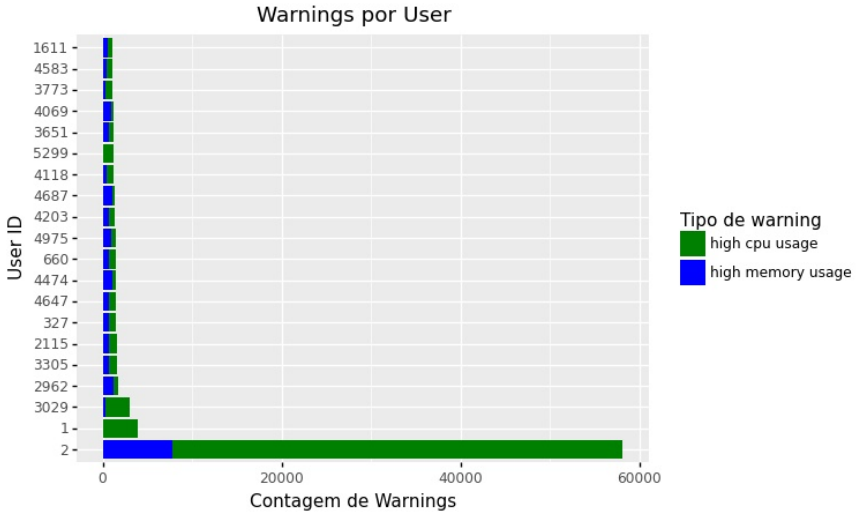


Figura 4.10: Gráfico de barras com os *warnings* de *execution* por *user*, por tipo de *warning*, para os 20 *users* com mais warnings.



Nas Figuras 4.9 e 4.10 são apresentados os gráficos de barras para os eventos de *execution* por *user*, por tipo de *warning*.

Neste caso, pode-se observar que um dos *users* é responsável pela maioria dos eventos (erros e *warnings*). Esta visualização permite focar a atenção da empresa neste *user* em particular de modo a entender o que se passa especificamente com

ele e, deste modo, resolver a maioria dos eventos. Portanto, esta visualização traria bastante valor à empresa.

Figura 4.11: Gráfico de barras com os erros de *execution* por *device*, por tipo de erro, para os 20 *devices* com mais erros.

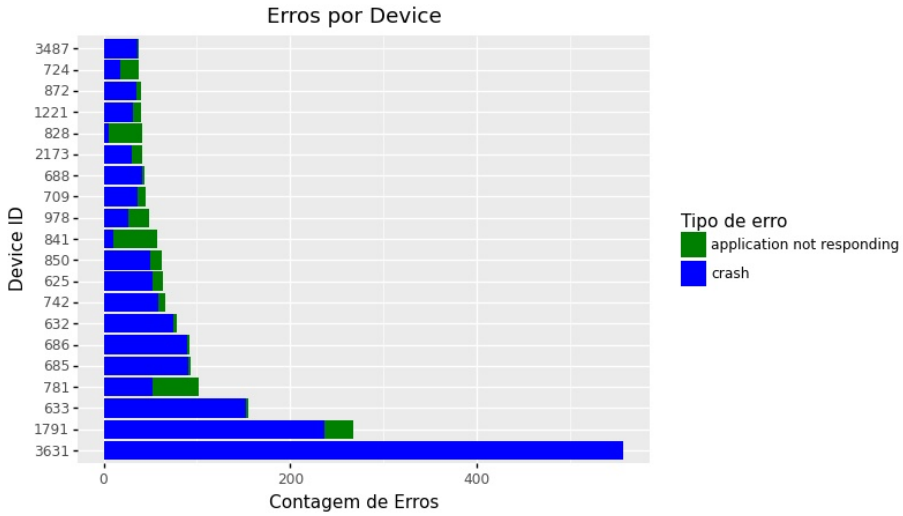
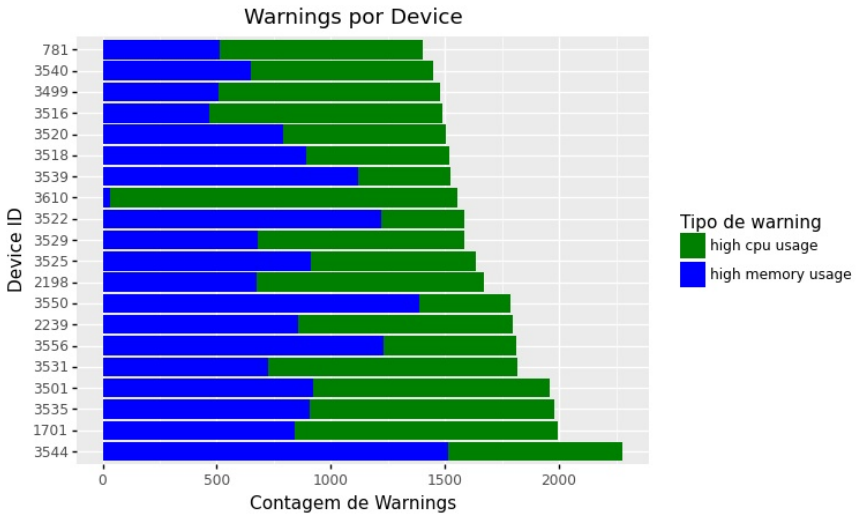


Figura 4.12: Gráfico de barras com os *warnings* de *execution* por *device*, por tipo de *warning*, para os 20 *devices* com mais warnings.



Nas Figuras 4.11 e 4.12, estão representados os gráficos de barras para os eventos de execução por *device*, para os 20 *devices* com mais eventos.

Não sendo o caso para a Figura 4.12, que contém uma distribuição mais uniforme dos eventos, a Figura 4.11 apresenta uma situação similar à das Figuras 4.9 e 4.10, onde é possível isolar 1 ou 2 *devices* de modo a resolver a grande maioria dos erros. No entanto, a visualização dos gráficos de barras referentes aos eventos dos *user* é mais revelador.

4.2.3 Erros vs *Warnings*

Nesta secção serão apresentados alguns gráficos em que se demonstra a relação entre o número de erros e o número de *warnings*.

Figura 4.13: Gráfico de Erros vs *Warnings* de *device* em escala logarítmica.

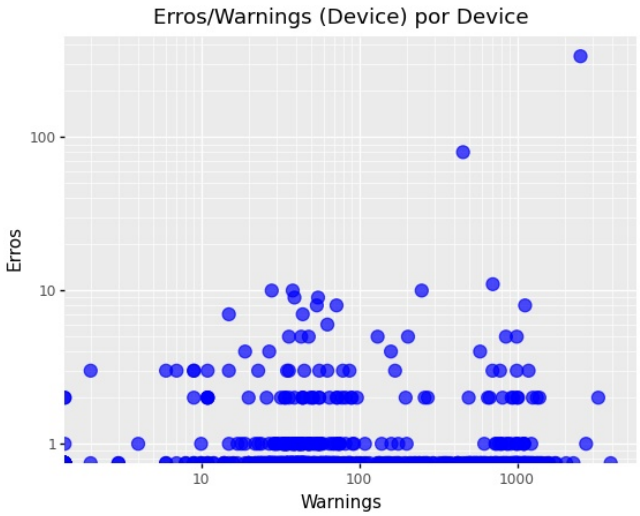
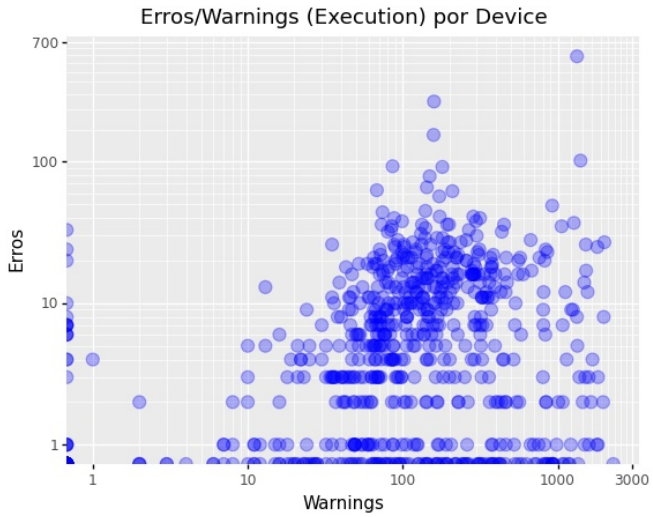


Figura 4.14: Gráfico de Erros vs *Warnings* de *execution* em escala logarítmica.



Nas Figuras 4.13 e 4.14 encontram-se os gráficos de Erros vs *Warnings* em escala logarítmica. Como se pode observar, não se consegue retirar uma relação entre a existência de um número elevado de erros e de *warnings*. Este resultado não causa espanto, já que os tipos de *warnings* e tipos de erros utilizados não estão necessariamente relacionados. Por exemplo, a existência de *warnings* de utilização elevada de memória ou processamento não se reflete, necessariamente, num posterior erro (*crash* ou *application not responding*) no dispositivo em questão.

4.3 Análise de *Clustering*

Uma das abordagens aplicadas nesta dissertação refere-se à segmentação de *devices* e *users* de forma a criar as tais *personnas* referidas na secção de *Web-Analytics* do Estado de Arte. Estas *personnas* referem-se a uma abstração ou representação de elementos que condensam um grupo de objetos com as mesmas características.

Esta abordagem utilizou os dados das *applications* executadas pelos *users* ou nos *devices* de forma a agrupar *users* ou *devices* com execuções de *applications* semelhantes. Grupos com execuções de *applications* semelhantes deverão desempenhar funções semelhantes, e portanto, podem-se reduzir ou eliminar redundâncias nas *applications* para desempenhar determinada função, isto é, a empresa estudada poderia tentar uniformizar as *applications* que utiliza para determinada função, reduzindo por exemplo, no custo das licenças. Por outro lado, e de maior importância para este trabalho, permitiria averiguar mais a fundo cada *cluster* ou *personna* de forma a entender que tipo de utilizador está representado na empresa, precisamente pela seu padrão de utilização dos recursos digitais.

Os dados foram tratados de forma a criar uma tabela que inclua como indíce cada um dos *devices*/*users* e as colunas com o *id* de cada *application*. Cada valor toma o valor de 0 ou 1, consoante a *application* foi ou não executada pela *user* ou executada no *device* durante o intervalo de tempo dos dados recolhidos.

Foram realizados 2 tipos de análises, tanto para os *users*, como para os *devices*. Um deles refere-se à aplicação do algoritmo *k-medoids*, com 4 *clusters*. O número de clusters foi obtido por visualização da curva dos erros quadráticos e também confirmado através da visualização gráfica posterior. Os resultados encontram-se nas Figuras 4.15 e 4.16.

Figura 4.15: Representação 3D (por PCA) do *clustering* aplicado aos *users*.

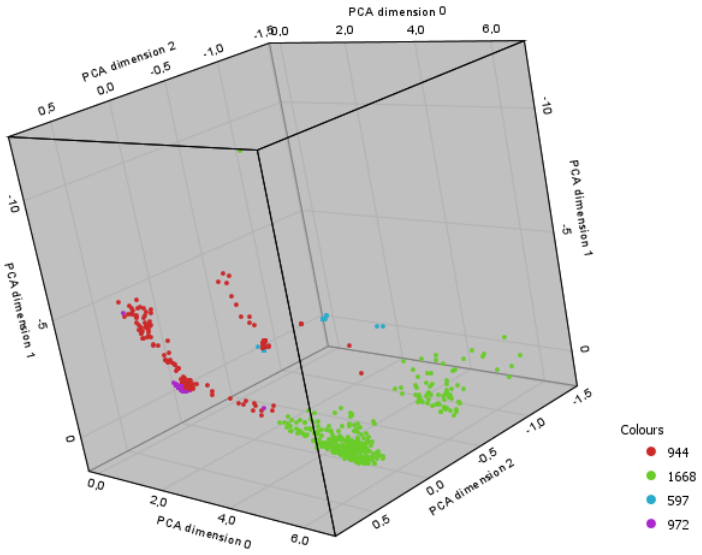
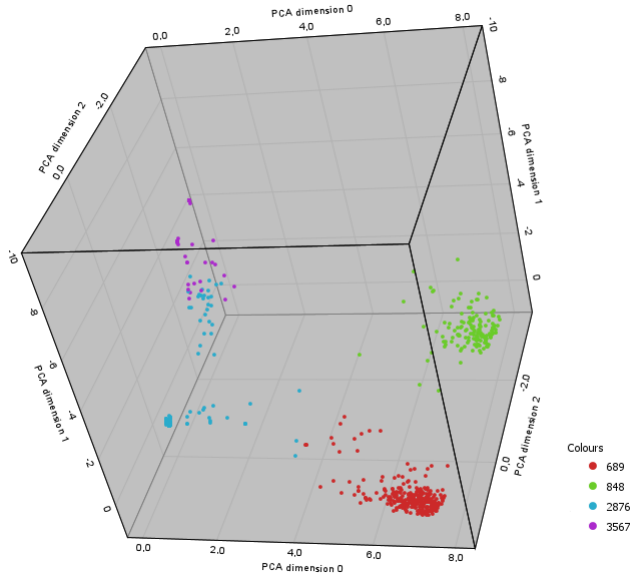


Figura 4.16: Representação 3D (por PCA) do *clustering* aplicado aos *devices*.



As representações tridimensionais das análises descritas anteriormente foram obtidas aplicando Análise de Componentes Principais por forma a reduzir a dimensão do espaço de *applications*, de modo a ser possível a sua visualização.

Nas Figuras 4.15 e 4.16, pode-se observar os 4 clusters. Apresentam-se também

os *ids* dos objetos (*users* e *devices*) no canto inferior direito, que representam as *personnas* (representantes) referidas anteriormente. Para o caso dos *users*, os representantes obtidos foram os *users* com *id* 944, 1668, 597 e 972. Por outro lado, para o caso dos *devices*, os representantes obtidos foram os *devices* com *id* 689, 848, 2876 e 3567. Estes representantes são o resultados da minimização das distâncias euclidianas dentro do *cluster*.

Verifica-se pelas duas Figuras 4.15 e 4.16, que o *clustering* por *devices* parece formar *clusters* mais consistentes e bem definidos. No total, foram encontrados 4 representantes que podem ser utilizados como tipos de *devices* existentes nos dados. Uma análise mais detalhada sobre o tipo de aplicações executadas por cada representante poderia fornecer mais informações acerca destes.

4.4 Regressão de *Random Forest* e *Gradient Boost*

Uma outra abordagem tomada nesta dissertação foi tentar prever o número de eventos (erros e *warnings*) para determinado tipo de objeto (registos de *devices*, *users* e *application*). Para isso, os eventos foram agregados por objeto e aplicava-se então um algoritmo de regressão com o intuito de prever os eventos (erros e *warnings*) associados aos registos de cada tipo de objeto. Os algoritmos usados nesta análise foram a *Random Forest* e o *Gradient Boost* de Regressão, ambos baseados em *ensembles* de árvores de decisão.

Parte dos resultados obtidos encontram-se nas Figuras 4.17 e 4.18. Estes gráficos apresentam o número médio de erros diário encontrado para cada objeto. No eixo horizontal, encontram-se a média de erros diária observada, enquanto que no eixo vertical se encontra a média de erros diária prevista, respetivamente.

Figura 4.17: Resultados da Previsão de Erros de *Device* por dias ativos para o algoritmo de *Random Forest*.

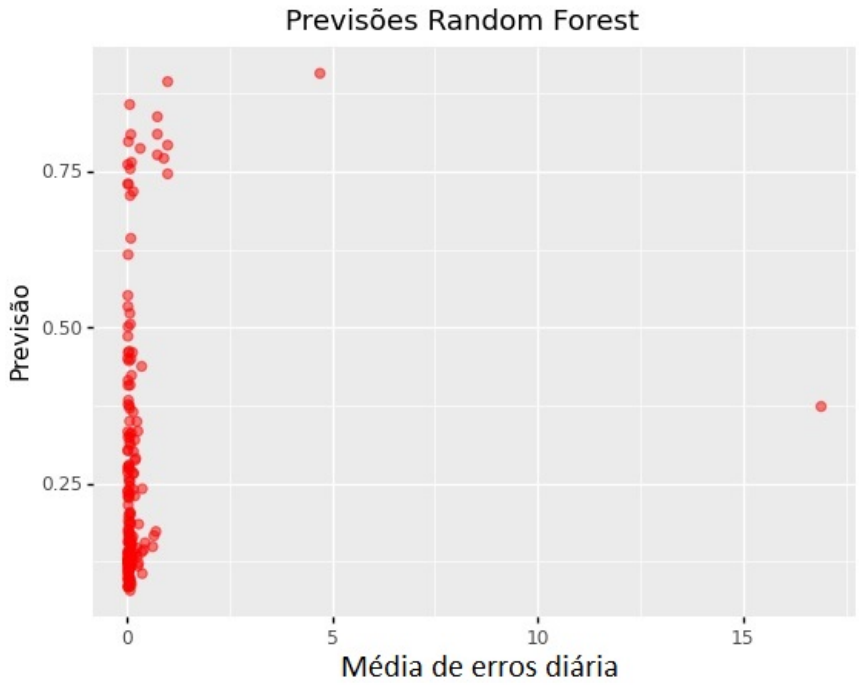
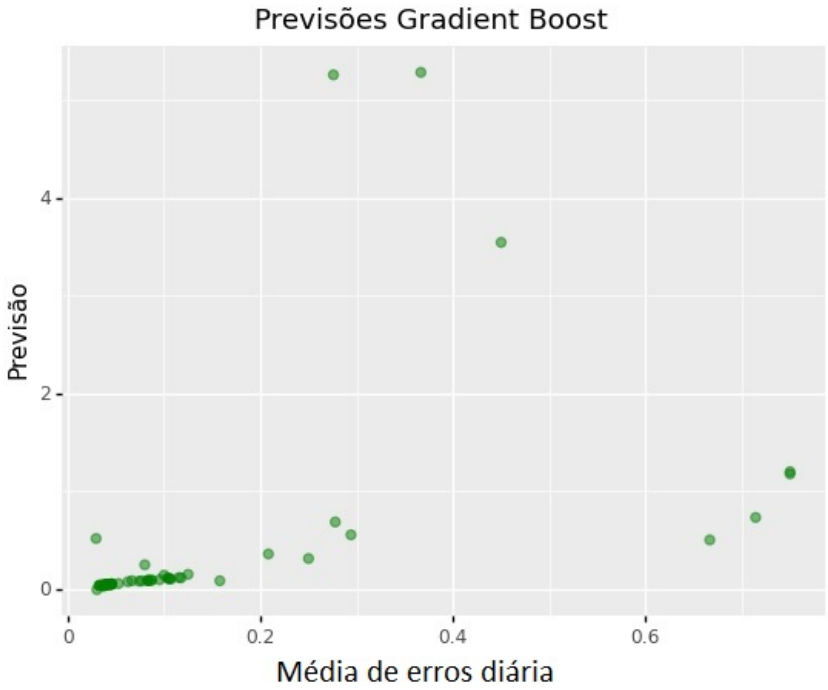


Figura 4.18: Resultados da Previsão de Erros de *Device* por dias ativos para o algoritmo de *Gradient Boost*.



Na Figura 4.17 pode-se observar que o algoritmo não tem qualquer capacidade aceitável de previsão da média dos erros diários, principalmente para casos com médias mais elevadas, sendo que estas seriam as mais interessantes de prever. Por exemplo, no caso extremo da média de erros diários observada de 17, o algoritmo prevê uma média de erros diários de 0.37. Esta qualidade de previsão é refletida na medida de ajuste de R^2 a seguir definida:

$$R^2 = 1 - \frac{\sum_i (y_i - \hat{y}_i)^2}{\sum_i (y_i - \overline{y})^2} \tag{4.1}$$

onde y_i são os valores observados, $\hat{y}_i$ são as previsões do modelo, $\overline{y}$ é a média dos valores observados e $i \in \{1, ..., n\}$, sendo n o número de observações.

De facto, o algoritmo apresenta um R^2 de 0.02 (equação 4.1), confirmando essa incapacidade. Além disso, analisando com mais detalhe os atributos mais importantes nas árvores de decisão presentes na *Random Forest*, chega-se à conclusão

de que o atributo mais importante é o *id* do próprio *device*, ou seja, o algoritmo não está a obter conhecimento com o agrupamento dos dados. Em vez disso, atua mais como um detetor de outliers, isolando *ids* em particular. Mesmo assim, não apresenta uma capacidade de previsão aceitável.

No caso da Figura 4.18, pode-se observar as previsões do algoritmo *Gradient Boost*, onde este modelo apresenta um R^2 de -3.7, pelo que se coloca num patamar pior do que o próprio algoritmo de *Random Forest*.

4.5 Entropia Modificada

4.5.1 Introdução

A equação 2.1 estabelece a forma de cálculo da entropia. Para um dado vetor de probabilidades de determinado evento, esta estabelece a quantidade média de questões binárias (0 ou 1) necessárias para obter a informação sobre determinado evento.

Por exemplo, para um evento com 4 categorias possíveis, cujo vetor de probabilidades é $[0.5, 0.25, 0.125, 0.125]$, a árvore binária (a entropia é definida pelo número de questões binárias) é representada na Figura 4.19. Nesta Figura, está representado o processo com os várias resultados, onde os nodos neutros (sem número) correspondem às questões binárias, os nodos numerados aos resultados correspondentes aos indíces do vetor de probabilidades. Em cada aresta encontra-se a probabilidade do processo terminar no nodo.

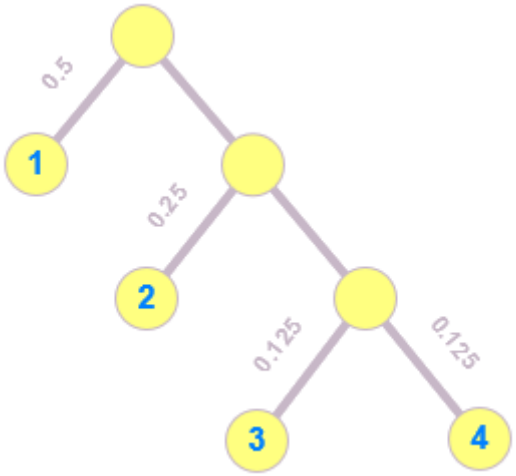


Figura 4.19: Árvore binária que descreve os resultados para um processo probabilístico de 4 categorias com probabilidades $0.5, 0.25, 0.125, 0.125$, respetivamente.

Sendo assim, e como se pode acompanhar pela Figura 4.19, o número de nodos para uma árvore binária na profundidade P é 2^P. Este número de nodos depende do inverso da probabilidade de determinado nodo. Por exemplo, para o nodo 4, a sua probabilidade é de 0.125, sendo que para uma árvore binária, à profundidade de 3, esta contém 8 nodos (2^3). Sendo assim, para se obter a profundidade da árvore

(número de questões) para a probabilidade de determinado resultado (p), faz-se uso da equação 4.2.

$$P = log_2(\frac{1}{p}) \tag{4.2}$$

Então, para se obter o número médio de questões binárias necessárias (profundidade P) para obter o resultado, calcula-se a entropia pela equação 2.1. No caso da árvore descrita pela Figura 4.19, a entropia calcula-se:

$$E = 0.5 \cdot log_2(\frac{1}{0.5}) + 0.25 \cdot log_2(\frac{1}{0.25}) + 2 \cdot 0.125 \cdot log_2(\frac{1}{0.125}) = 1.75\, bits$$

Pela expressão anterior, conclui-se que para o processo representado na Figura 4.19 são precisos, em média, 1.75 bits de informação, ou seja, é necessário responder em média a 1.75 questões binárias.

Voltemos à Figura 4.10. Esta figura é bastante interessante, pois revela bastante informação útil em relação aos *warnings* de *execution* para os *users*, onde apenas visualizando o gráfico de barras, é possível verificar que os *warnings* estão bastante concentrados em apenas um *user*. A atenção da empresa podia então ser voltada para resolver a questão deste *user* em particular, resolvendo assim a grande maioria dos *warnings* que foram aparecendo, usando uma quantidade limitada de recursos para o conseguir.

4.5.2 Entropia Modificada

A análise da entropia permite então medir o desequílibro de um determinado vetor, medindo a informação contida neste.

No caso dos dados analisados nesta dissertação, o objetivo desta abordagem é comparar o desequílibro dos vetores que contêm os erros/*warnings* para cada atributo de cada tipo de objeto (por exemplo, *company* dentro da tabela das *applications*). Naturalmente, estes vetores terão tamanhos diferentes consoante o número de registos únicos dentro de cada atributo. Por forma a poder estabelecer-se uma comparação entre o desequílibro destes vetores, é preciso aplicar uma normalização no cálculo da entropia.

Seguindo a mesma lógica demonstrada na introdução desta secção, vai modificar-se o cálculo da entropia da forma descrita na equação 4.3.

$$H_m = -\sum_{i}^{n} p_i \, log_n(p_i) \tag{4.3}$$

sendo H_m a entropia modificada, n a dimensão do vetor e p_i a probabilidade do próximo evento estar associado ao registo i na tabela de atributos.

O resultado da equação 4.3, com a base n em vez da base 2, deve-se ao facto da questão deixar de ser binária e passar a ser da dimensão do vetor. Assim, é definida uma métrica entre 0 e 1 que permite comparar o desequílibro dos vetores de tamanhos diferentes. Esta métrica será responsável por definir quais os atributos a que o analista dentro da empresa se deve focar de forma a resolver o maior número de erros/*warnings*, com o mínimo de atenção exigida.

Na análise, as probabilidades foram obtidas usando a contagem dos eventos (erros ou *warnings*) por dias de execução do objeto em questão (p.e. *device*), para cada vetor de atributos.

É introduzida então a equação explícita da entropia modificada:

$$H_m = -\sum_{i=1}^{n} \frac{N_{e_i}}{t_i^{\left(\frac{1}{\gamma}\right)}} \, log_n \left(\frac{N_{e_i}}{t_i^{\left(\frac{1}{\gamma}\right)}} \right) \tag{4.4}$$

sendo H_m a entropia modificada, N_{e_i} a frequência relativa de eventos, t_i o tempo de execução em dias, n a dimensão do vetor, γ como fator de correção e o i correspondente ao registo no vetor.

Supondo que temos um vetor $[\frac{1}{10}, \frac{10}{100}, \frac{100}{1000}]$, onde os numeradores são o número de erros (p.e.) e os denominadores são o número de dias de execução do respetivo objeto durante o intervalo de tempo de funcionamento da empresa em análise. Para o primeiro elemento deste vetor, houve 1 erro em 10 dias de execução, para o segundo, 10 erros em 100 dias e para o terceiro, 100 erros em 1000 dias. Ora, calculando a frequência de erros por dia, temos que para qualquer dos elementos corresponde uma frequência de 0.1 erros por dia, ou seja, é esperado que o próximo erro que aconteça, para tempos de execução iguais, seja igualmente provável para cada um

dos objetos. No entanto, para uma empresa, apesar da probabilidade associada aos registos ser a mesma, o terceiro registo (de $\frac{100}{1000}$) tem mais impacto na empresa do que os restantes, pois corresponde a cerca de 90% de todos os erros que ocorrem.

Para corrigir o processo descrito no parágrafo anterior, foi introduzido um fator extra designado por γ na medição da entropia, como mostrado na equação 4.4. Este fator aplica uma transformação à variável t (tempo de execução) de forma a valorizar registos com tempos de execução mais elevados, já que estes têm um maior impacto na empresa. O fator γ assume valores iguais ou maiores do que 1 e deve ser determinado conforme o impacto que a quantidade absoluta de erros tem na empresa.

Quando o fator γ toma o valor de 1, a equação 4.4 torna-se então na equação 4.3, com $p_i = \frac{N_{e_i}}{t_i}$, e portanto p_i é a frequência relativa de eventos por unidade de tempo para o tipo de evento i.

Por outro lado, quando $\gamma \to \infty$, $t_i \to 1$, a equação 4.3 resulta na equação 4.5, em que N_{e_i} é a frequência relativa de eventos para o tipo de evento i:

$$H_m = -\sum_{i}^{n} N_{e_i}\, log_n(N_{e_i}) \tag{4.5}$$

Neste caso, o tempo de execução é ignorado e a entropia modificada é calculada apenas com as frequências relativas dos erros.

Nas Tabelas 4.14, 4.15, 4.17 e 4.18 apresentam-se os resultados para dois casos particulares, onde foi utilizado um γ de 1 (neutro).

Tabela 4.14: Tabela com a Entropia Modificada ($H_m(4.4)$) ordenada crescente para os diversos atributos de cada objeto para erros de *device*.

Evento: Erros de *Device*

Objeto (atributo)	$H_m(\gamma = 1)$
Device (os_architecture)	0.00
Device (cpu_model)	0.32
Device (hard_disks)	0.46
Device (os_version_and_architecture)	0.47
Device (device_manufacturer)	0.49
Device (device_id)	0.54
Device (device_type)	0.64
Device (entity)	0.72

Tabela 4.15: Tabela com a Entropia Modificada ($H_m(4.4)$) ordenada crescente para os diversos atributos de cada objeto para *warnings* de *device*.

Evento: *Warnings* de *Device*

Objeto (atributo)	$H_m(\gamma = 1)$
Device (os_architecture)	0.07
Device (hard_disks)	0.71
Device (device_id)	0.71
Device (entity)	0.75
Device (device_type)	0.76
Device (device_manufacturer)	0.78
Device (cpu_model)	0.80
Device (os_version_and_architecture)	0.81

Nas Tabelas 4.14 e 4.15, pode-se observar a medição da entropia modificada para cada atributo de cada objeto (neste caso de eventos de *device*, apenas há atributos de *device*), para os erros e *warnings* respetivamente. Quanto menor a entropia, mais desequilibrado está o vetor das distribuições de erros. Portanto, podemos concentrar a atenção nestes atributos com menor entropia.

Na Tabela 4.14, repara-se que o valor da entropia modificada para os erros de *device* ordenados por *os_architecture*, têm uma entropia de 0, ou seja, o correspondente vetor encontra-se perfeitamente desequilibrado. A Tabela 4.16 contém a informação relacionada com este vetor.

Tabela 4.16: Tabela dos Erros de *device* por Sistema Operativo para os objetos *device*.

Erros de *Device* (*Device*)

os_architecture	Erros	Tempo (dias)	Execuções	Probabilidade
64 bits	814	5078	580	1.0
32 bits	0	443	624	0.0

Observando mais de perto a Tabela 4.16, valida-se então que o vetor está perfeitamente desequilibrado, ou seja, os *devices* cujo sistema operativo é 64 *bits* recolheram todos os erros existentes. Desta forma, pode-se então fazer uma análise dentro do contexto da empresa de forma a encontrar os motivos por que o sistema operativo de 64 *bits* é o que reune todos os erros.

Tabela 4.17: Tabela com a Entropia Modificada ordenada crescente para os diversos atributos de cada objeto para erros de *execution*.

Evento: Erros de *Execution*

Objeto (atributo)	$H_m(\gamma = 1)$
Device (os_architecture)	0.32
Application (company)	0.46
Application (name)	0.50
Application (application_id)	0.50
Device (os_version_and_architecture)	0.54
Device (cpu_model)	0.61
Application (platform)	0.72
Device (hard_disks)	0.75
User (user_id)	0.76
User (full_name)	0.77
Device (device_id)	0.79
Device (device_manufacturer)	0.80
Device (device_type)	0.84
Device (entity)	0.90
User (job_title)	0.95
User (department)	0.95

Tabela 4.18: Tabela com a Entropia Modificada ordenada crescente para os diversos atributos de cada objeto para *warnings* de *execution*.

Evento: *Warnings* de *Execution*

Objeto (atributo)	$H_m(\gamma = 1)$
Application (platform)	0.08
Device (os_architecture)	0.11
Application (company)	0.51
Application (application_id)	0.54
Application (name)	0.59
User (user_id)	0.66
User (full_name)	0.75
Device (device_id)	0.84
Device (os_version_and_architecture)	0.86
User (job_title)	0.87
Device (device_type)	0.88
Device (entity)	0.90
Device (device_manufacturer)	0.92
Device (cpu_model)	0.93
Device (hard_disks)	0.94
User (department)	0.95

Como exemplo contrário, temos nas Tabelas 4.17 e 4.18 o vetor dos *departments*, no tipo de objeto *User*, que apresenta um valor de entropia modificada de 0.95, ou seja, neste caso, o vetor está bastante balanceado. A Tabela 4.19 apresenta os atributos do vetor, com os erros, tempo, número de execuções e probabilidade para os vários *departments* relativos aos *Users*.

Tabela 4.19: Tabela dos Erros de *execution* por *Department* para os *Users*.

Erros de *User* (*department*)

department	Erros	Tempo (dias)	Execuções	Probabilidade
Finance	63	33	5	0.48
Sales	93	89	13	0.27
Marketing	229	233	26	0.25
Total	385	355	44	1

Como se verifica pela Tabela 4.19 o vetor de Probabilidades encontra-se bastante balanceado.

No entanto, neste caso, os *users* correspondentes ao departamento de *Marketing* tiveram mais erros e mais tempo de execução que os restantes. Assim, por exemplo, uma aplicação de um γ de 1.7, resultaria na Tabela 4.20, onde desta vez, o termo de probabilidade para o departamento de *Marketing* tem mais importância. Quer isto dizer que, usando o $\gamma = 1.7$, o algoritmo atribui uma probabilidade maior ao departamento de *Marketing* e, assim, reconhece que mais atenção deve ser dirigida a este departamento, pois recolhe mais erros e execuções em termos absolutos do que os restantes. O valor da entropia para este vetor de User (*department*) torna-se então de 0.99, ou seja, bastante mais balanceado o vetor, como aliás se pode observar pela coluna "Probabilidade Corrigida com γ"na Tabela 4.20.

Tabela 4.20: Tabela dos Erros de *execution* por *Department* para os *Users*, com γ de 1.7.

Erros de *User* (*department*)

department	Erros	Tempo (dias)	Execuções	Probabilidade Corrigida com $\gamma = 1.7$
Marketing	229	233	26	0.38
Finance	63	33	5	0.34
Sales	93	89	13	0.28
Total	385	355	44	1

4.5.3 Aplicação GUI de Monitorização

De forma a colocar em prática a abordagem desenvolvida na secção anterior da entropia modificada, foi desenvolvida uma aplicação *Graphical User Interface* (GUI) como protótipo de apresentação de dados e resultados dessa mesma abordagem.

A aplicação foi desenvolvida utilizando principalmente as bibliotecas *PyQt5*, *pyqtgraph*, *pandas*, *threading* e *SQLAlchemy* da linguagem de programação *Python*.

Em seguida, serão apresentadas as diferentes etapas de visualização e monitorização da análise de entropia modificada aos dados na aplicação GUI.

Começa, então, com a Janela de apresentação inicial da GUI na Figura 4.20. Nesta, apesar de serem apresentadas as funcionalidades que serão utilizadas, estas estão desabilitadas até que se carreguem os dados para a memória.

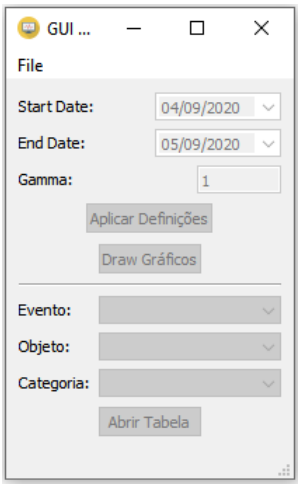


Figura 4.20: Janela de apresentação inicial da GUI.

Para carregar os dados para a memória, é necessário especificar o caminho da base de dados (.db), utilizando o *menu "File"*, em que se tem a opção de abrir a base de dados, como mostra na Figura 4.21.

Depois de especificado o caminho, o programa começará então a importar os dados, dando-se a indicação de *Loading* na barra de *status* até finalizar a importação.

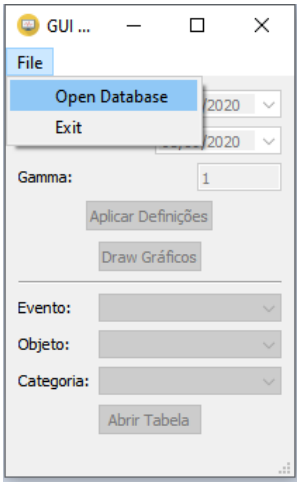


Figura 4.21: Procedimento para carregar os dados para a memória, especificando o caminho para a base de dados.

Finalizada a importação dos dados, algumas opções na janela principal estarão desta vez disponíveis, como se mostra na Figura 4.22. Desta vez, estão disponíveis as opções de escolha de datas iniciais e finais, o fator γ e ainda a opção de "Aplicar Definições". Esta opção, então, executa as restrições necessárias aos dados, assim como calcula os dados necessários para a próxima fase.

Deve-se ter em conta ainda que estes procedimentos demorados são realizados numa segunda *thread*, de forma a não interferir com a interatividade da GUI enquanto carrega e calcula os dados.

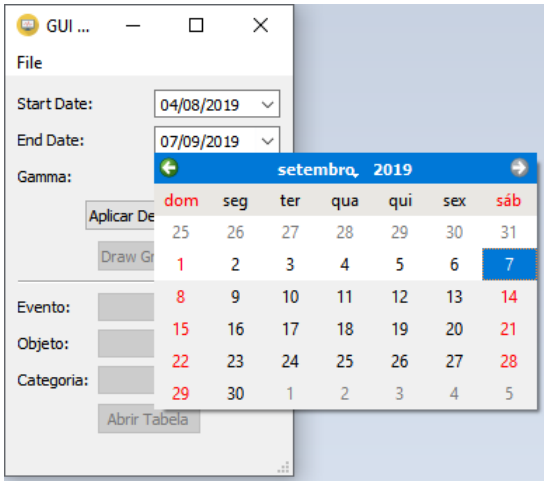


Figura 4.22: Procedimento de escolha das datas iniciais e finais de cálculo, assim como do fator de γ (*"gamma"*).

Após serem efetuados os cálculos necessários à próxima fase, a GUI volta a habilitar as outras opções dos passos seguintes da abordagem, como se mostra na Figura 4.23. Desta vez, é possível fazer o desenho dos gráficos com os "Objeto (atributo)"ordenados por valor crescente de entropia, assim como mostrar a tabela para cada "Objeto (atributo) de cada tipo de evento, ou seja, uma tabela muito similar ao que deu origem às Tabelas 4.16, 4.19 e 4.20.

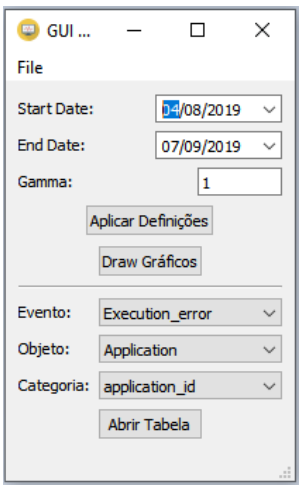


Figura 4.23: Interatividade da GUI agora toda disponível para o utilizador.

Em seguida, na Figura 4.24, é mostrada a apresentação da GUI quando se carregou na opção *"Draw Gráficos"*. Foi desenhado o gráfico de barras para os 4 tipos de eventos (erros de *device*, *warnings* de device, erros de *execution* e *warnings* de *execution*). Os gráficos contêm o valor de cada categoria para cada objeto, apresentados de forma ordenada por valor de entropia, como se pode, aliás, observar.

O eixo de baixo dos gráficos de barras apresenta-se inicialmente com os nomes sobrepostos. Isto deve-se ao facto de ainda não estar implementado em *pyqtgraph* a funcionalidade de os colocar na vertical.

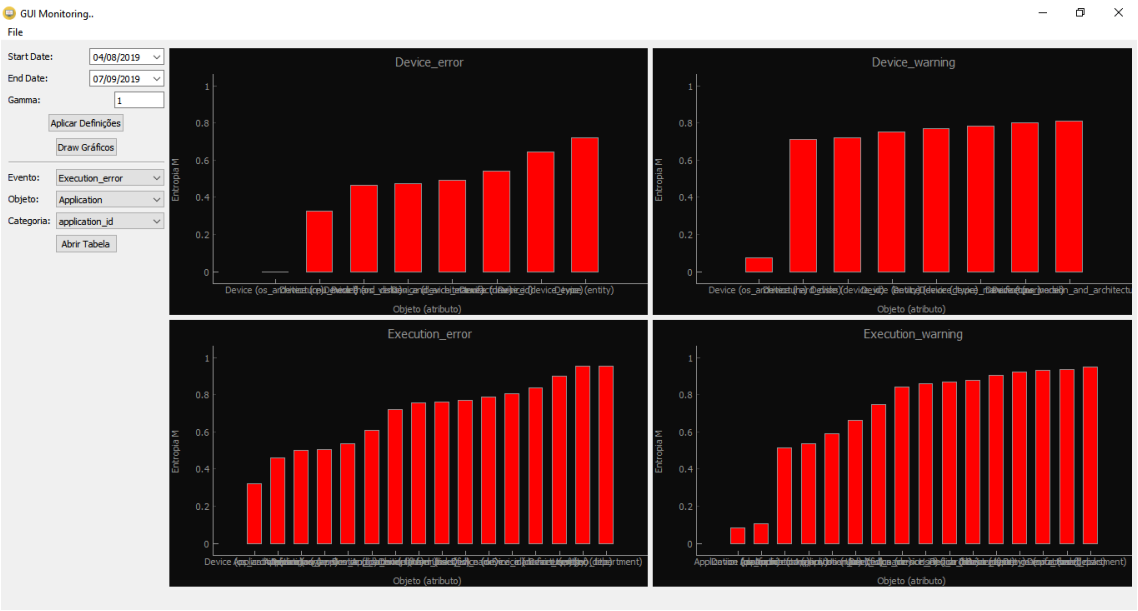


Figura 4.24: Imagem ilustrativa da GUI apresentando os gráficos de barras da entropia modificada para cada tipo de evento/objeto/categoria, ordenados pelo valor da entropia modificada.

No entanto, os gráficos são interativos e pode-se facilmente fazer zoom nas barras à escolha. A interatividade permite ainda aplicar transformações, mudar cores, exportar as imagens, entre outras funcionalidades. Algumas delas, encontram-se expostas na Figura 4.25.

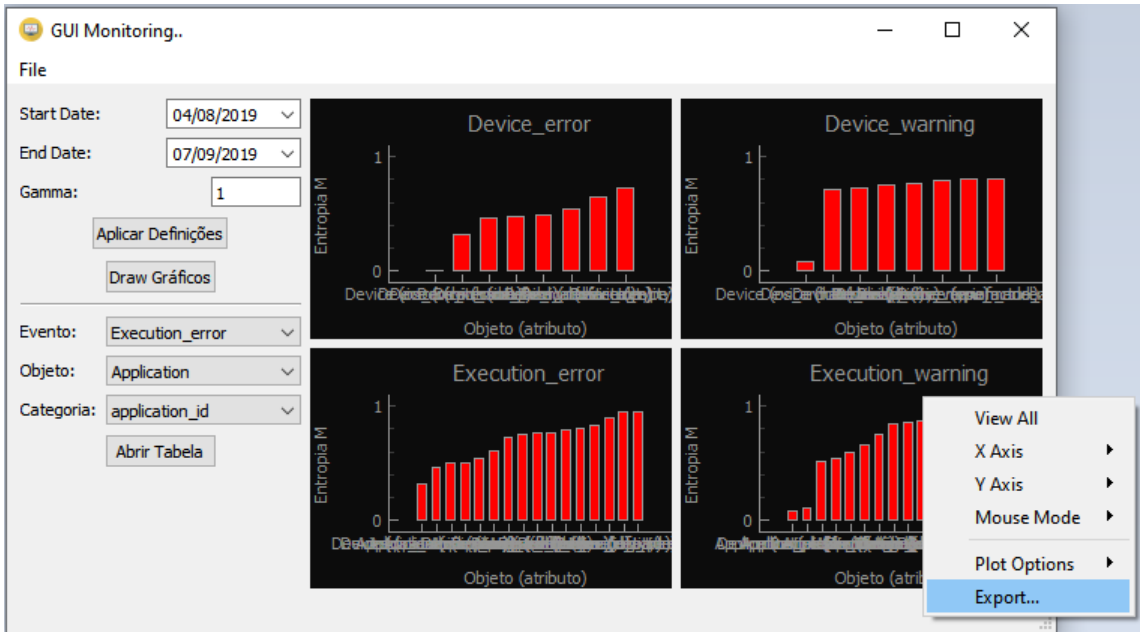


Figura 4.25: Figura demonstrativa da interactividade com os gráficos da GUI.

Por outro lado, há disponível ainda a opção de visualizar a tabela com a informação sobre as categorias dos objetos para cada tipo de evento, fazendo uso da parte inferior da janela principal da GUI.

Na Figura 4.26, pode-se então observar uma dessas tabelas. São apresentados no título da janela o tipo de evento, assim como o objeto a que corresponde a análise. Na primeira coluna está contida a categoria pelo qual foi feita a agregação. No caso desta tabela, foi selecionada a categoria *application_id*, em que se pode analisar as diferenças entre as diferentes *applications*, denotadas pelo seu identificador. Nas restantes colunas, é mostrada a contagem de eventos (*count*), o tempo de execução em dias (*timespan*), o impacto (*impact*) da *application_id* em específico, ou seja, quantos objetos contém aquele identificador (neste caso, são todos únicos, portanto tomam todos o valor de 1) e ainda a probabilidade (*prob*), que é estimada pela razão do número de eventos pelo tempo de execução e pelo total de eventos, normalizada ao vetor inteiro. Esta última, tenta responder à questão "Contando que se colocam todos os dispositivos a executar durante um período de tempo igual, qual a probabilidade do próximo evento ser atribuído ao *application_id* referente a essa linha?"

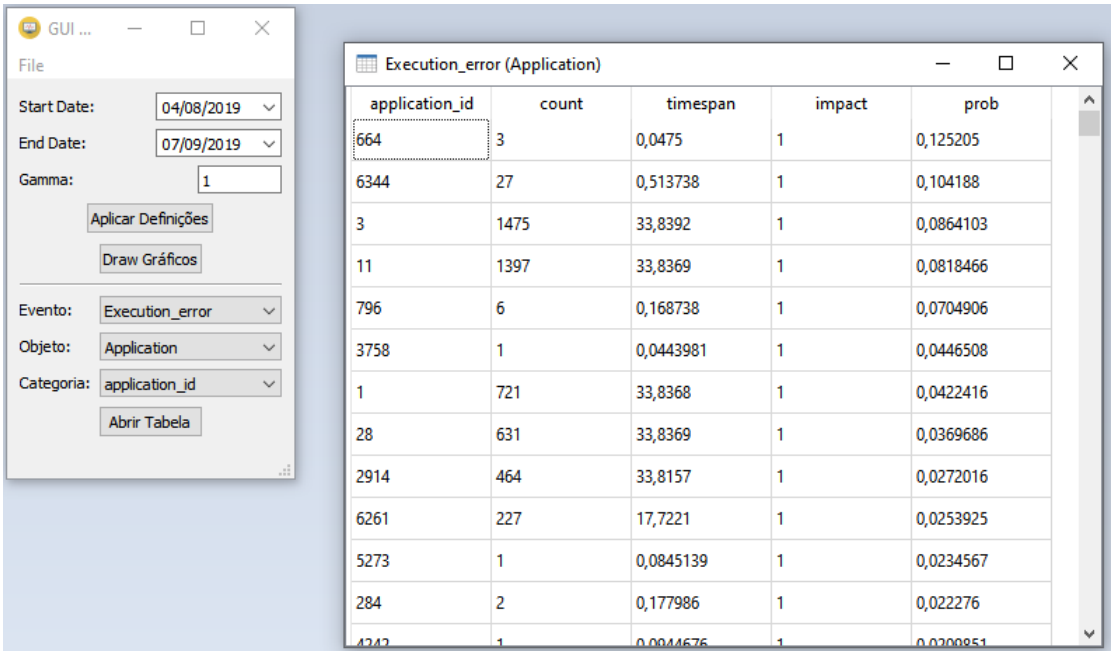


Figura 4.26: Figura demonstrativa da opção de visualizar a tabela correspondente a um evento/objeto/categoria.

Finalmente, foi criado um executável com a aplicação GUI, de forma a ser possível executá-la facilmente.

Capítulo 5

Conclusões e Trabalho Futuro

Esta dissertação abordou a exploração de métricas, objetivos e abordagens para a melhoria da experiência dos utilizadores. A primeira fase teve uma componente principalmente exploratória, em que se pretendeu conhecer que métricas sobre os dados se devem recolher, fazendo uso do reportado atualmente a nível científico e empresarial. Além disso, foram também explorados os objetivos a que se propõem nos últimos anos os vários grupos que se debruçam neste tipo de estudo e se de facto estes são atingíveis.

Sendo assim, esta dissertação fornece informações valiosas sobre as métricas que a *Fujitsu* deverá procurar em termos de dados, bem como os objetivos a que se poderá e deverá propor atingir, seja na própria empresa, ou na atividade de consultoria.

Foram explorados e expostos ainda algumas abordagens com as diferentes métricas e objetivos delineados no Estado de Arte.

Quanto à análise de dados propriamente dita, foram encontradas algumas limitações. Em primeiro lugar, foram encontrados graves problemas com a consistência dos dados, nomeadamente a existência de registos em algumas tabelas que deveriam ter o seu correspondente noutras, sendo que não se encontram nestas últimas. Esta inconsistência não impediu que se realizasse a análise, mas questionaria a validade das extrapolações, caso os resultados da análise fossem aplicados à empresa a que se referem os dados recolhidos. Por outro lado, as limitações também se verificaram no facto de não apenas os dados não conterem as métricas necessárias para aplicar as abordagens e análises dispostas na literatura, como por exemplo medidas dos recursos disponíveis aos utilizadores durante as suas tarefas ou medidas relacionadas com a navegação dos utilizadores na elaboração dessas mesmas tarefas, como também não estarem disponíveis nos dados, métricas explícitas de avaliação da experiência dos utilizadores, sendo que estas métricas eram essenciais para atingir o objetivo primordial da dissertação. Ainda assim, tentou-se encontrar formas indiretas de melhorar a experiência dos utilizadores.

Uma delas foi a segmentação dos utilizadores e dos dispositivos que interagiam com estes, de forma a ser possível personalizar a análise. Como se observa na Figura 4.16, foram determinados dois grupos de dispositivos cujos utilizadores apresentam elevado nível de similaridade, com representantes 689 e 848, e onde se podiam estabelecer o que se chama na literatura de *personna*, e ainda outros 2 grupos mais difuso, com representantes 2876 e 3567. Uma interpretação mais detalhada desta análise não foi realizada por escassez de informação acerca dos identificadores das

aplicações utilizadas.

Uma outra forma indireta debruçou-se na tentativa de prever ou de alguma forma entender os erros e *warnings* que foram registados. A lógica seria, diminuindo os erros e *warnings* a que se expunham os utilizadores, a sua experiência melhoraria. Fazendo uso das métricas recolhidas dos objetos, tentou-se prever o número de eventos ao longo do tempo com uma série de algoritmos preditivos do tipo *Emsemble* com árvores de decisão. Esta acabou por se revelar não ter sido bem sucedida, com as regressões, usando os tais algoritmos de *Ensemble*, a terem resultados nada satisfatórios. Este facto sugere que estes eventos dependem de outras características dos objetos, para além das disponíveis nos dados, sendo que conhecer o contexto interno da empresa poderia ter um grande impacto.

Sendo assim, a abordagem transitou para desenvolver estratégias de isolamento de objetos interessantes no sentido de influenciarem com uma quantidade de eventos (erros e *warnings*) elevada. Desta forma, poder-se-iam fornecer dicas ao analista de dados da empresa, tendo acesso a informações internas sobre o contexto dos processos na empresa, conseguindo, portanto, relacionar os desequílibros demonstrados por certos objetos, de forma a avaliar se estes fazem ou não sentido nos processos da empresa. A última análise (entropia modificada e aplicação de monitorização) focou-se precisamente neste ponto, tentando encontrar e isolar certos objetos que criam desequilibros nos eventos, resultando numa entropia baixa no vetor em questão. Isso permite que se realcem estes objetos problemáticos, focando a atenção e recursos da empresa para os resolver, tendo assim um impacto elevado nos processos da empresa em questão.

Esta análise de entropia modificada demonstrou bons resultados e criou expectativas de sucesso a nível empresarial. Com a aplicação desta técnica, foi possível isolar certos *devices*, e em particular apontar grandes desequílibros, sendo que na Tabela 4.16 se pode verificar que os erros todos surgem dos sistemas operativos de 64*bits* e na pela Tabela 4.18, onde a *platform* das *applications* ou de novo o sistema operativo dos *devices* são apontados como pontos que requerem atenção e onde podem existir objetos que estão a criar problemas.

Foi ainda desenvolvida uma aplicação GUI que permite auxiliar o processo descrito no parágrafo anterior.

O trabalho futuro deverá então focar-se no melhoramento das limitações demonstradas nesta dissertação, nomeadamente, deve ter-se atenção à programação da recolha de métricas com valor, ao estabelecimento de objetivos concretos, fazendo-

se auxiliar pelo que foi desenvolvido. Ainda neste ponto, convém referir que, estabelecendo-se o objetivo de melhorar a experiência dos utilizadores, se deve planear formas diretas de executar esta medição, de modo a ser possível estabelecer métricas da avaliação dos utilizadores, bem como das abordagens e algoritmos em específico.

Bibliografia

[1] *Fujitsu at a Glance*. Set-2020. URL: https://www.fujitsu.com/global/about/corporate/inf.

[2] *IT solutions for a digital world*. Set-2020. URL: https://www.fujitsu.com/global/products/.

[3] *Fujitsu Portugal - Linkedin*. Set-2020. URL: https://pt.linkedin.com/company/fujitsuportugal.

[4] *Portugal Fujitsu inaugura o novo Centro de Competências de Lisboa*. Set-2020. URL: https://www.fujitsu.com/pt/about/resources/news/press-releases/2008/Portugal_Fujitsu_inaugura_o_novo_Centro_de_Compet_ncia_de_Lisboa.html.

[5] *Fujitsu assina protocolo com Município de Viseu e estende as suas competências digitais*. 2019. URL: https://www.fujitsu.com/pt/about/resources/news/press-releases/2019/fujitsu-assina-protocolo-com-munic-pio-de-viseu-e-estende.html.

[6] *Novo Centro de Competências da Fujitsu inaugurado em Braga cria 300 postos de trabalho*. 2016. URL: https://www.fujitsu.com/pt/about/resources/news/press-releases/2016/novo-centro-de-compet-ncias-da-fujitsu-inaugurado-em-braga.html.

[7] *End-user*. Set-2020. URL: https://dictionary.cambridge.org/dictionary/english/end-user.

[8] *Analytics*. Set-2020. URL: https://dictionary.cambridge.org/dictionary/english/analytics.

[9] Fujitsu. *Whitepaper Fujitsu End User Analytics*. Rel. téc. Fujitsu, 2016, pp. 1–4.

[10] Josep Ll Berral et al. «Towards energy-aware scheduling in data centers using machine learning». Em: *Proceedings of the e-Energy 2010 - 1st Int'l Conf. on Energy-Efficient Computing and Networking* April (2010), pp. 215–224. DOI: 10.1145/1791314.1791349.

[11] Huigui Rong et al. «Optimizing energy consumption for data centers». Em: *Renewable and Sustainable Energy Reviews* 58 (2016), pp. 674–691. ISSN: 18790690. DOI: 10.1016/j.rser.2015.12.283. URL: http://dx.doi.org/10.1016/j.rser.2015.12.283.

[12] Richard Brown et al. *Report to Congress on Server and Data Center Energy Efficiency : Public Law 109-431 Environmental Energy Technologies Division Alliance to Save Energy ICF Incorporated*. Rel. téc. August. 2008. URL: http://eetd.lbl.gov/publications/report-to-congress-on-server-and-data.

[13] Jonathan G. Koomey. «Worldwide electricity used in data centers». Em: *Environmental Research Letters* 3.3 (2008). ISSN: 17489326. DOI: 10.1088/1748-9326/3/3/034008.

[14] Peter Johnson e Tony Marker. *Data centre energy efficiency product profile*. Rel. téc. 2009.

[15] Emily Farnworth e Juan Carlos Castilla-rubio. *SMART 2020 : Enabling the low carbon economy in the information age*. Rel. téc. 2020.

[16] Albert Greenberg et al. «The cost of a cloud: research problems in data center networks». Em: *ACM SIGCOMM Computer Communication Review* 39.1 (2008), pp. 68–73. ISSN: 0146-4833. DOI: 10.1145/1496091.1496103.

[17] Miguel Jimeno, Ken Christensen e Bruce Nordman. «A network connection proxy to enable hosts to sleep and save energy». Em: *Conference Proceedings of the IEEE International Performance, Computing, and Communications Conference* (2008), pp. 101–110. DOI: 10.1109/PCCC.2008.4745133.

[18] R. Wattenhofer et al. «Distributed topology control for power efficient operation in multihop wireless ad hoc networks». Em: *Twentieth Annual Joint Conference of the IEEE Computer and Communications Society INFOCOM 2001.* 3 (2001), pp. 1388–1397. DOI:

10.1109/INFCOM.2001.916634. URL: http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=916634.

[19] Benjie Chen et al. «Span: An energy-efficient coordination algorithm for topology maintenance in ad hoc wireless networks». Em: *Wireless Networks* 8.5 (2002), pp. 481–494. ISSN: 10220038. DOI: 10.1023/A:1016542229220.

[20] Hainan Zhang et al. «Free cooling of data centers: A review». Em: *Renewable and Sustainable Energy Reviews* 35 (2014), pp. 171–182. ISSN: 13640321. DOI: 10.1016/j.rser.2014.04.017.

[21] T. Brunschwiler et al. «Toward zero-emission data centers through direct reuse of thermal energy». Em: *IBM Journal of Research and Development* 53.3 (2009), pp. 1–13. ISSN: 00188646. DOI: 10.1147/JRD.2009.5429024.

[22] Sorell, V. and Abougabal. «An Analysis of the Effects of Ceiling Height on Air Distribution in Data Centers». Em: *ASHRAE Transactions* 112 (2006), pp. 623–631.

[23] Kailash C. Karki e Suhas V. Patankar. «Airflow distribution through perforated tiles in raised-floor data centers». Em: *Building and Environment* 41.6 (2006), pp. 734–744. ISSN: 03601323. DOI: 10.1016/j.buildenv.2005.03.005.

[24] Luiz André Barroso e Urs Hölzle. «The case for energy-proportional computing». Em: *Computer* 40.12 (2007), pp. 33–37. ISSN: 00189162. DOI: 10.1109/MC.2007.443.

[25] Xiaobo Fan, Wolf-Dietrich Weber e Luiz Andre Barroso. «Power provisioning for a warehouse-sized computer». Em: *ACM SIGARCH Computer Architecture News* 35.2 (2007), p. 13. ISSN: 01635964. DOI: 10.1145/1273440.1250665.

[26] Grant Wu et al. «Energy-efficient virtual machine placement in data centers by genetic algorithm». Em: *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)* 7665 LNCS.PART 3 (2012), pp. 315–323. ISSN: 03029743. DOI: 10.1007/978-3-642-34487-9_39.

[27] Ramya Raghavendra et al. «No "power"struggles: Coordinated multi-level power management for the data center». Em: *Operating Systems Review (ACM)* 42.2 (2008), pp. 48–59. ISSN: 01635980. DOI: 10.1145/1346281.1346289.

[28] Anshul Gandhi et al. «Optimal power allocation in server farms». Em: *SIGMETRICS/Performance'09 - Proceedings of the 11th International Joint Conference on Measurement and Modeling of Computer Systems* 37.1 (2009), pp. 157–168. ISSN: 0163-5999. DOI: 10.1145/1555349.1555368.

[29] Dara Kusic et al. «Power and performance management of virtualized computing environments via lookahead control». Em: *5th International Conference on Autonomic Computing, ICAC 2008* (2008), pp. 3–12. DOI: 10.1109/ICAC.2008.31.

[30] Ian H Witten e Eibe Frank. *Data Mining: Practical Machine Learning Tools and Techniques with Java Implementations*. 2000, p. 416. ISBN: 0080890369.

[31] Josep Ll Berral, Ricard Gavaldà e Jordi Torres. «Adaptive scheduling on power-aware managed data-centers using machine learning». Em: *Proceedings - 2011 12th IEEE/ACM International Conference on Grid Computing, Grid 2011* Ml (2011), pp. 66–73. DOI: 10.1109/Grid.2011.18.

[32] Josep Ll Berral, Ricard Gavaldà e Jordi Torres. «Power-Aware Multi-DataCenter management using machine learning». Em: *Proceedings of the International Conference on Parallel Processing* (2013), pp. 858–867. ISSN: 01903918. DOI: 10.1109/ICPP.2013.102.

[33] Practical Web Analytics e User Experience. *Practical Web Analytics for User Experience*. 2013, p. 234. ISBN: 9780124046191. DOI: 10.1016/c2012-0-01162-3.

[34] Domingos P. «A Few Useful Things to Know About Machine Learning». Em: *Communications of the ACM* 55.10 (2012). URL: https://dl.acm.org/citation.cfm?id=2347755.

[35] E. G. Galitskaya e E. B. Galitskkiy. «Classification trees». Em: *Sotsiologicheskie Issledovaniya*. 3. 2013, pp. 165–192. DOI: 10.4018/978-1-60960-557-5.ch006.

[36] Carl Kingsford e Steven L. Salzberg. «What are decision trees?» Em: *Nature Biotechnology* 26.9 (2008), pp. 1011–1013. ISSN: 10870156. DOI: 10.1038/nbt0908-1011.

[37] Thomas Hancock et al. «Lower bounds on learning decision lists and trees». Em: *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)* 900.0040 (1995), pp. 527–538. ISSN: 16113349. DOI: 10.1007/3-540-59042-0_102.

[38] Hans Zantema e Hans L Bodlaender. «Finding Small Equivalent Decision Trees is Hard». Em: *International Journal of Foundations of Computer Science* 11.2 (2000), pp. 343–354.

[39] G. E. Naumov. «NP-completeness of problems of construction of optimal decision trees». Em: *Soviet Physics Doklady* 36 (1991), p. 270.

[40] J. R. Quinlan. «Induction of decision trees». Em: *Machine Learning* 1.1 (1986), pp. 81–106. ISSN: 0885-6125. DOI: 10.1007/bf00116251.

[41] J. Ross Quinlan. *C4.5: Programs for Machine Learning*. Morgan Kaufmann, 1993, p. 302.

[42] Leo Breiman, Jerome Friedman, Charles J. Stone. *Classification and Regression Trees*. Taylor Francis, 1984, p. 368.

[43] *Decision Tree Algorithm - Explained*. 2019. URL: https://towardsdatascience.com/decision-tree-algorithm-explained-83beb6e78ef4.

[44] Misha Denil, David Matheson e Nando De Freitas. «Narrowing the Gap: Random Forests In Theory and In Practice». Em: *Proceedings of The 31st International Conference on Machine Learning* 1998 (2014), pp. 665–673. ISSN: 9781634393973. arXiv: arXiv:1310.1415v1. URL: http://jmlr.org/proceedings/papers/v32/denil14.html.

[45] Gérard Biau. «Analysis of a random forests model». Em: *Journal of Machine Learning Research* 13 (2012), pp. 1063–1095. ISSN: 15324435. arXiv: 1005.0208.

[46] Haozhe Zhang, Dan Nettleton e Zhengyuan Zhu. «Regression-Enhanced Random Forests». Em: *Section on Statistical Learning and Data Science* 1 (2019). arXiv: 1904.10416. URL: http://arxiv.org/abs/1904.10416.

[47] Tomislav Hengl et al. «Random forest as a generic framework for predictive modeling of spatial and spatio-temporal variables». Em: *PeerJ* 8 (2018). ISSN: 21678359. DOI: 10.7717/peerj.5518.

[48] Tianqi Chen e Carlos Guestrin. «XGBoost: A scalable tree boosting system». Em: *Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* 13-17-Augu (2016), pp. 785–794. DOI: 10.1145/2939672.2939785. arXiv: 1603.02754.

[49] Shirin Glader. *Machine Learning Basics - Gradient Boosting & XGBoost*. 2020. URL: https://www.shirin-glander.de/2018/11/ml%7B%5C_%7Dbasics%7B%5C_%7Dgbm/.

[50] Duncan Greaves. «Making Sense of Big Data Using Cluster Analysis». Em: *Impact* 2019.1 (2019), pp. 25–29. ISSN: 2058-802X. DOI: 10.1080/2058802x.2019.1571299.

[51] Kiri Wagstaff et al. «Constrained K-means Clustering with Background Knowledge». Em: *International Conference on Machine Learning ICML* pages (2001), pp. 577–584. URL: http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.90.4624%7B%5C&%7Drep=rep1%7B%5C&%7Dtype=pdf.

[52] Anil K. Jain. «Data clustering: 50 years beyond K-means». Em: *Pattern Recognition Letters* 31.8 (2010), pp. 651–666. ISSN: 01678655. DOI: 10.1016/j.patrec.2009.09.011. URL: http://dx.doi.org/10.1016/j.patrec.2009.09.011.

[53] Hisashi Kashima et al. «K-means clustering of proportional data using L1 distance». Em: *Proceedings - International Conference on Pattern Recognition* (2009), pp. 1–4. ISSN: 10514651. DOI: 10.1109/icpr.2008.4760982.

[54] Jianchang Mao e A.K. Jain. «A self-organizing network for hyperellipsoidal clustering (HEC)». Em: *IEEE Transactions on Neural Networks* I.1 (1996), pp. 16–29.

[55] Hae Sang Park e Chi Hyuck Jun. «A simple and fast algorithm for K-medoids clustering». Em: *Expert Systems with Applications* 36.2 PART 2 (2009), pp. 3336–3341. ISSN: 09574174. DOI: 10.1016/j.eswa.2008.01.039. URL: http://dx.doi.org/10.1016/j.eswa.2008.01.039.

[56] Linda Musthaler. *Nexthink's digital experience management platform quickly solves performance problems*. 2020. URL: https://www.networkworld.com/article/3285646/nexthinks-digital-experience-management-platform-quickly-solves-performance-problems.html.

[57] *NXQL Tutorial*. 2020. URL: https://doc.nexthink.com/Documentation/Nexthink/latest/APIAndIntegrations/NXQLTutorial.

[58] *NXQL Data Model*. 2020. URL: https://doc.nexthink.com/Documentation/Nexthink/latest/APIAndIntegrations/NXQLDataModel.