Universidade do Minho
Escola de Engenharia

Francisco Barros da Cunha

Análise Inteligente de Dados de Sistemas de Pesagem Ciber-Físicos

outubro de 2022

Análise Inteligente de Dados de Sistemas de Pesagem Ciber-Físicos
Francisco Barros da Cunha
UMinho | 2022

Universidade do Minho
Escola de Engenharia

Francisco Barros da Cunha

Análise Inteligente de Dados de Sistemas de Pesagem Ciber-Físicos

Dissertação de Mestrado
Mestrado Integrado em Engenharia e Gestão de Sistemas de Informação

Trabalho efetuado sob a orientação do
Professor Doutor Paulo Alexandre Ribeiro Cortez

outubro de 2022

DIREITOS DE AUTOR

AGRADECIMENTOS

No fim deste percurso, que me marcou e desenvolveu enquanto pessoa, gostaria de agradecer a todos aqueles que contribuíram e possibilitaram a concretização desta importante fase da minha vida, a todos vós transmito os meus mais sinceros agradecimentos.

Começo por agradecer ao meu orientador o Professor Doutor Paulo Cortez, por toda a disponibilidade, dedicação e apoio fornecido ao longo de todos este processo. Agradeço também todos os ensinamentos, conselhos e conhecimentos partilhados, os quais foram fundamentais para o desenvolvimento deste trabalho, bem como para a criação em mim de um grande interesse por esta área.

A nível profissional, um agradecimento especial à Doutora Graça Coelho e ao Engenheiro Cândido Martins pela oportunidade e pelos vários ensinamentos transmitidos, os quais levarei para resto da minha vida. Além deste agradecimento, um grande obrigado a todas as pessoas que fazem ou fizeram parte da Cachapuz, sendo obrigatório referenciar o Luiz, Peter, Marcelo, João, Rui, Jorge, Juliana, Natália, visto que me acompanharam todos os dias deste meu percurso na Cachapuz.

A nível académico, tenho de agradecer obrigatoriamente ao Júlio Barros e ao Bruno Miguel pelo papel fundamental que estes tiveram no desenvolvimento da dissertação e pelos vários ensinamentos e conhecimentos que estes me transmitiram.

Agradeço aos meus amigos tanto de Esposende como da Universidade por todos os bons momentos que passamos juntos ao longo destes anos.

É com enorme carinho que por fim agradeço à minha namorada, aos meus pais e à minha família por acreditarem sempre em mim, estarem comigo nos melhores e nos piores momentos e por toda a força que me dão dia após dia.

Muito obrigado a todos de coração.

This work is supported by European Structural and Investment Funds in the FEDER component, through the Operational Competitiveness and Internationalization Programme (COMPETE 2020) [Project nº 069716; Funding Reference: POCI-01-0247-FEDER-069716].

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

Análise Inteligente de Dados de Sistemas de Pesagem Ciber-físicos

A presente dissertação surge no âmbito do projeto *New Generation of Cyber-physical Weighing Systems* (NeWeSt), cujo consórcio é formado por: Cachapuz - Weighing & Logistics Systems, Lda, INL - *International Iberian Nanotechnology Laboratory*, DTx - Digital Transformation CoLab e a Universidade do Minho. Esta tem como foco o desenvolvimento de uma *framework* baseada em técnicas de *Machine Learning* para prever a ocorrência de anomalias em processos de carregamento de sacos de cimento, a partir de um micro-serviço em *cloud*. Técnicas de *Machine Learning* foram aplicadas com o intuito de se criar modelos preditivos de classificação, capazes de prever a ocorrência de desvios relativamente a processos de pesagem em sacos de cimento. Esta *framework* foi aplicada e avaliada numa organização multinacional de sistemas de pesagem do mundo real e após a exploração de vários algoritmos de classificação, o modelo *Random Forest* foi o adotado, visto que obteve os melhores resultados.

Palavras-chave: Aprendizagem supervisionada*,* Ciência de dados, *Machine Learning,* Previsão de Anomalias, Sistemas de Pesagem.

ABSTRACT

Intelligent Data Analysis of Cyber-physical Weighing Systems

This dissertation arises in the scope of the New Generation of Cyber-physical Weighing Systems (NeWeSt) project, whose consortium is composed of: Cachapuz - Weighing & Logistics Systems, Lda, INL - International Iberian Nanotechnology Laboratory, DTx - Digital Transformation CoLab and the University of Minho. This project focuses on the development of a framework based on Machine Learning techniques to predict the occurrence of anomalies in the process of loading cement bags through a cloud microservice. Machine Learning techniques were applied with the purpose of creating predictive classificatory models capable of predicting the occurrence of anomalies related to the loading process of bags of cement. The framework was implemented and evaluated by a weighing systems multinational organization in a real-world setting. After exploring various classification algorithms, we adopted the Random Forest model since it obtained the best overall results.

Keywords: Anomaly Prediction, Data Science, Machine Learning, Supervised Machine Learning, Weighing Systems.

ÍNDICE

LISTA DE FIGURAS

LISTA DE TABELAS

LISTA DE ABREVIATURAS

AI	*Artificial Intelligence*
ANN	*Artificial Neural Networks*
API	*Application Programming Interface*
AUC	*Area Under the Curve*
AutoML	*Automated Machine Learning*
BD	Base de Dados
CI/CD	*Continuous Integration/Continuous Delivery*
CPS	*Cyber-Physical Systems*
CRISP-DM	*CRoss Industry Standard Process for Data Mining*
CSV	*Comma-separated values*
DL	*Deep Learning*
DM	*Data Mining*
DT	*Decision Tree*
DTx	*Digital Transformation Colab*
DS	*Data Science*
ERP	*Enterprise Resource Planning*
FN	*False Negative*
FP	*False Positive*
FPR	*False Positive Rate*
GBT	*Gradient Boosting Tree*
GPS	*Global Positioning System*
IDE	*Integrated Development Environment*
INL	*International Iberian Nanotechnology Laboratory*
IoT	*Internet of Things*
KDD	*Knowledge Discovery in Databases*
ML	*Machine Learning*
MLP	*Multilayer Perceptron*
NBI	*National Bridge Inventory*
NeWeSt	*New Generation of Cyber-physical Weighting Systems*

REST	*RESTful API*
RF	*Random Forest*
ROC	*Receiver Operating Characteristic*
RW	*Rolling Window*
SHAP	*SHapley Additive exPlanations*
SI	Sistemas de Informação
SLV	Sistemas Logístico de Veículos
SQL	*Structured Query Language*
SVM	*Support Vector Machine*
TTC	*Truck Traffic Classification*
TN	*True Negative*
TP	*True Positive*
TPR	*True Positive Rate*
UM	Universidade do Minho
WIN	*Weight-In-Motion*
XAI	*Explainable Artificial Intelligence*
XGBoost	*eXtreme Gradient Boosting*

"The beautiful thing about learning is nobody
can take it away from you."

B. B. King

1. INTRODUÇÃO

Este capítulo inicia-se com o tópico "Enquadramento e Motivações", que visa dar a conhecer dar a conhecer as razões que levaram à execução dos trabalhos de dissertação. De seguida, é apresentada a "Formulação do Problema", seguida dos "Objetivos e Resultados Esperados" no âmbito do caso de estudo aprofundado, e posteriormente a "Organização do Documento" e por fim, é apresentada a "Abordagem Metodológica".

1.1 Enquadramento e Motivações

O surgimento novas tecnologias digitais tem mudado a forma como a sociedade e, em particular, a indústria têm desenvolvido e conduzido os seus domínios com o intuito de alcançar os benefícios referentes à implementação do paradigma denominado de Transformação Digital (Fitzgerald et al., 2013; Ross et al., 2016). Ao longo dos anos, são diversas as organizações que não têm alterado a sua forma de atuar e o modo como os seus processos organizacionais e de negócio são implementados, criando desta forma uma incapacidade de seguir as tendências e competitividade do mercado. Tal facto faz com que estas organizações comecem a extinguir-se e um exemplo notório foi a empresa de referência mundial na área do aluguer de vídeo, a *Blockbuster* (Hess et al., 2016), que acabou por fechar as portas em 23 de setembro de 2010 (Davis & Higgins, 2013).

Como anteriormente abordado, o paradigma da Transformação Digital abrange diversas áreas e domínios. Em termos industriais, o desenvolvimento e a integração de novas tecnologias levou ao aparecimento da designação Indústria 4.0 (Cachapuz - Weighing & Logistics Systems, 2019), denominada por muitos como a 4ª revolução industrial (Bitkom et al., 2016). Esta tem como base a unificação dos sistemas físicos e lógicos (Santos et al., 2018), ou seja, a criação de uma nova geração de sistemas computacionais integrados com o mundo físico, denominados de *Cyber-Physical Systems* (CPS) (Baheti & Gill, 2011), em português Sistemas Ciber-Físicos.

O setor de pesagem industrial é um mercado bastante competitivo, onde o aparecimento de elementos de mudança e de inovação representa um fator preponderante para as organizações se destacarem das demais. Assim sendo, com base no conceito de evolução tecnológica e diferenciação surge o projeto de R&D NeWeSt - AVISO No 17/SI/2019 - Nova geração de Sistemas de Pesagem Ciber-Físicos, onde se insere a presente dissertação. O projeto tem como consórcios a empresa Cachapuz -

Weighing & Logistics Systems, Lda, o INL - International Iberian Nanotechnology Laboratory, o DTx - Digital Transformation CoLab e a Universidade do Minho.

A base deste projeto está associada a uma alteração disruptiva do paradigma referente ao setor da pesagem industrial, com o objetivo de obter um ecossistema de dispositivos e de interações inseridos nos conceitos da Indústria 4.0 (Cachapuz - Weighing & Logistics Systems, 2019), criando assim, a possibilidade de uma mudança holística do modelo de negócio, bem como a exploração de novos mercados (Ebert & Duarte, 2018). O projeto NeWeSt assenta em vários pilares:

1. *Internet of Things* (IoT) – baseia-se na criação de redes inteligentes de dispositivos conectados entre si (Valter, et al., 2020);
2. *Cloud Computing* – apresentado pela I.B.M e pela Google em 2007 (Murchinson & Haikes, 2007; Lohr, 2007; Vouk, 2008), definido como um modelo que permite a partilha de recursos computacionais como serviços para diversas entidades (Mell & Grace, 2011);
3. *Machine Learning* (ML) – corresponde ao estudo e aplicação de algoritmos computacionais capazes de aprender e aperfeiçoar de forma autónoma através da utilização de dados (Mitchell, 1997), e;
4. *Big Data* – representa o repositórios de armazenamento de dados em quantidades massivas (Cumbley & Church, 2013).

Relativamente à presente investigação, foca-se essencialmente na componente de *Machine Learning* do projeto NeWeSt. Esta terá a participação e o apoio da entidade Cachapuz - Weighing & Logistics Systems, Lda na elaboração dos objetivos a serem alcançados, bem como das atividades a serem realizadas. A Cachapuz é uma empresa especialista em soluções de pesagem para a indústria, cujo foco passa pela criação de produtos e processos que podem ser aplicados aos vários setores de atividade em que a pesagem se encontra como um elemento crítico da cadeia de valor (Cachapuz - Weighing & Logistics Systems, 2019).

1.2 Formulação do Problema

Diariamente, são realizadas várias pesagens nas fábricas dos vários clientes que a organização Cachapuz detém. Estas pesagens abrangem um grande número de áreas industriais, contudo, uma das mais importantes para a Cachapuz, uma vez que se posiciona como especialista, é a pesagem e logística

associada à indústria do cimento. No que se refere às organizações pertencentes a este ramo industrial, existe um processo crítico da sua cadeia de valor, que corresponde ao carregamento de sacos e de paletes com cimento para posterior envio para os clientes.

No decorrer deste processo, subsiste um parâmetro que é constantemente analisado e avaliado pelos responsáveis, o desvio percentual entre o peso do material solicitado pelo cliente e o peso do material realmente enviado para o mesmo. O desvio do peso constitui um parâmetro muito importante, visto que este recaí sobre dois fatores cruciais: monetários e segurança. O fator monetário está associado ao pagamento e fornecimento do produto em quantidades distintas das expectáveis. Por outro lado, o fator segurança está associado ao transporte de material em quantidades superiores aos limites de segurança definidos para o bom funcionamento do veículo de transporte utilizado.

São diversos os fatores que levam ao aparecimento de desvios nas quantidades de matérias transacionadas, contudo um dos fatores primordiais corresponde às máquinas utilizadas para a realização do enchimento dos sacos de cimento. Estas máquinas são calibradas para encherem os sacos de cimento com um determinado peso, no entanto, as alterações climáticas influenciam as propriedades do produto carregado e, consequentemente, o peso real do saco difere do peso carregado pela máquina. Este problema conduz a que alguns dos veículos utilizados para carregar este tipo de material apresentem uma grande diferença entre o peso real do material relativamente ao peso da quantidade pedida pelo cliente, denominado por desvio. Isto resulta num vasto conjunto de consequências, como por exemplo, o bloqueio do veículo na fábrica e a respetiva repetição do processo, ou no pior dos cenários, a saída do veículo da fábrica com as quantidades transportadas fora do intervalo aceite, para que não haja perda de tempos logísticos.

Com base no problema enunciado, o trabalho realizado contribui então para o desenvolvimento de uma *framework* que com recurso a *Machine Learning* é capaz de realizar a predição de anomalias em sistemas de carregamento de sacos de cimento. Os resultados obtidos são de elevada relevância, visto permitirem fornecer uma visão futura de possíveis acontecimentos a ocorrer durante a realização do processo abordado, melhorando desse modo o *workflow* dos mesmos.

1.3 Objetivos e Resultados Esperados

Os objetivos desta dissertação encontram-se de acordo com o plano de ação do projeto *NeWeSt*, mais concretamente, com a componente de *Dispatch Workflow* associada ao mesmo. Assim sendo, os objetivos desta dissertação estão inseridos em dois tópicos distintos, sendo eles, a interpretação de

dados fornecidos pela plataforma SLV – Sistema Logístico de Veículos pertencente à Cachapuz, de forma a compreender o negócio associado à indústria cimenteira, e posterior desenvolvimento de modelos de *Machine Learning,* no contexto do caso de estudo definido para um dos vários objetivos inerentes ao projeto.

No que se refere à componente de *Machine Learning*, esta consiste em combinar o desenvolvimento de modelos de *Machine Learning* para a identificação de padrões e comportamentos no processo logístico de carregamento de sacos de cimento e o desenvolvimento de um micro-serviço em *cloud*. O primeiro tem como objetivo a previsão de ocorrência de anomalias relativamente ao peso do material pedido pelo cliente e o peso realmente transportado; o segundo permite a utilização do modelo selecionado após o *deployment* em produção através de uma *REST API* (chamadas HTTP), bem como a possibilidade de escalar o modelo de aprendizagem selecionado no sistema. Desse modo, o micro-serviço será responsável por realizar um alerta caso preveja um desvio percentual do peso superior a 2% ou inferior a -2%. Esta abordagem permite, então, à organização agir de forma proativa sobre os componentes e intervenientes do processo, possibilitando a redução de eventuais desvios percentuais do peso e, consequentemente, a otimização deste mesmo processo em termos de custos e tempos associados.

1.4 Organização do Documento

O documento desenvolvido encontra-se dividido em 4 capítulos e em cada um deles são abordados conteúdos relevantes para o estudo. O primeiro capítulo apresenta a contextualização do documento, as motivações, os objetivos, resultados esperados e a metodologia utilizada para a realização do projeto de dissertação. O segundo capítulo corresponde à revisão de literatura, onde primeiramente, é abordada a estratégia utilizada para a pesquisa de documentos capazes de fornecer informação pertinente, e posteriormente, são desenvolvidos os temas: *Data Science, Data Mining, Machine Learning,* Sistemas de pesagem e Anomalias. O terceiro capítulo tem como base o desenvolvimento do caso de estudo, seguindo a metodologia de investigação selecionada para dar resposta ao problema. Finalmente, é apresentado o quarto e último capítulo, que conclui todo o trabalho efetuado, bem como a discussão dos objetivos alcançados com o desenvolver do projeto, e possíveis trabalhos futuros a serem realizados.

1.5 Abordagem Metodológica

Tendo em consideração o contexto em que se insere a dissertação, a metodologia selecionada para dar resposta ao plano de ação é o CRISP-DM, uma vez que esta permite diminuir a complexidade associada aos projetos de *Data Mining* em termos da gestão dos passos a serem seguidos, tornando assim os desenvolvimentos e as implementações mais rápidas, confiáveis e exequíveis (Wirth & Hipp, 2000).

O *CRoss-Industry Standard Process for Data Mining*, abreviadamente CRISP-DM, é uma metodologia que foi criada em 1996 e conta com um fluxo de seis fases (Figura 1). Esta metodologia tem como foco projetos de *Data Mining*, sendo uma abordagem bastante prática, robusta e simples de se aplicar (Chapman et al., 2000). Além disso, as características que englobam esta metodologia, como por exemplo, o facto de poder ser utilizada em qualquer setor industrial, aliado à sua não dependência de ferramentas especificas, tornam a mesma uma das metodologias mais preponderantes para a elaboração de projetos de *Data Mining* (Shearer, 2000).

Como abordado anteriormente, a metodologia CRISP-DM é formada por seis fases, sendo elas, *"Business Understanding", "Data Understanding", "Data Preparation", "Modeling", "Evaluation"* e *"Deployment".* As fases enunciadas não ocorrem de forma isolada, ou seja, na maioria dos casos, a resolução de uma leva a alterações de outras fases já realizadas, sendo assim um processo iterativo e de desenvolvimento gradual ao longo do tempo do projeto.

De seguida as seis fases do CRISP-DM são apresentadas e descritas de uma forma pormenorizada:

1. ***Business Understanding*** **(Compreensão do Negócio)** – Corresponde à primeira fase da metodologia CRISP-DM e tem como foco o entendimento e perceção dos objetivos referentes ao problema de *Data Mining* que o projeto a desenvolver pretende dar resposta. Contudo, esta fase deverá partir de uma perspetiva de negócio, assim sendo, existe a necessidade de ser elaborado um plano de ação com base nos requisitos definidos para o projeto. Isto leva a que na maioria das vezes exista a necessidade de realizar alterações nas tarefas alocadas à compreensão do negócio, uma vez que, com o decorrer do projeto, existem sempre alterações no âmbito e nos requisitos.

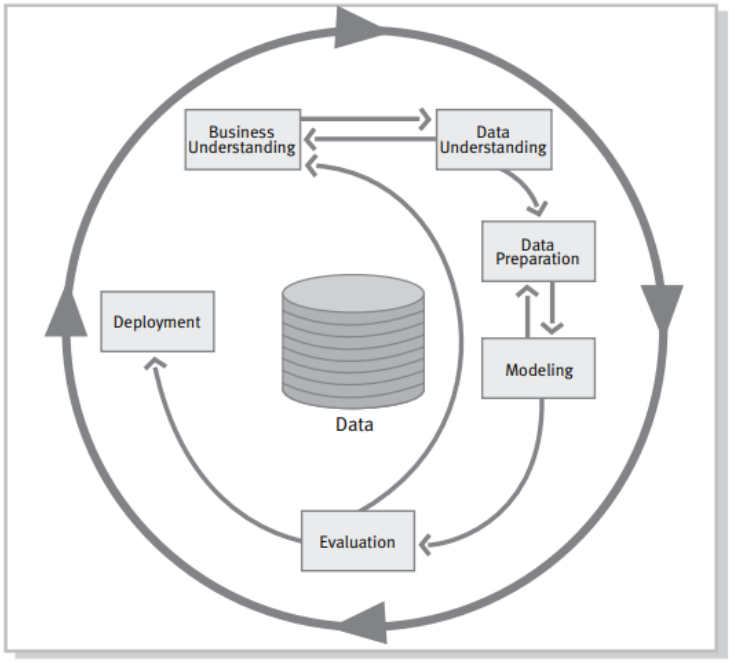

Figura 1 - Metodologia CRISP-DM (Retirado de Chapman et al., 2000).

2. ***Data Understanding*** **(Compreensão dos Dados)** – É a segunda fase desta metodologia e baseia-se na recolha, compreensão, familiarização e interpretação dos dados existentes e necessários para a realização do projeto de *Data Mining*. Além disso, nesta fase são identificados os problemas apresentados nos dados existentes, bem como as relações visíveis entre os mesmos.

3. ***Data Preparation*** **(Preparação dos Dados)** – Trata-se da terceira fase, cujo propósito é a seleção, limpeza, transformação e construção dos dados, ou seja, o processamento dos mesmos que, posteriormente, serão utilizados no processo de modelação.

4. ***Modeling*** **(Modelação)**– Sendo a quarta fase da metodologia, o *Modeling* corresponde à fase onde ocorre a seleção e aplicação de várias técnicas e modelos de *Machine Learning*, as quais serão depois avaliadas com base em métricas existentes e selecionadas para o efeito.

5. ***Evaluation*** **(Avaliação)** – É a quinta fase da metodologia. Aqui é preponderante proceder à avaliação do desempenho dos modelos elaborados com o intuito de averiguar aquele que apresenta melhores resultados, com base nas métricas de avaliação definidas. Deste modo, é escolhido o modelo que melhor se adequa aos objetivos estipulados e requisitos definidos.

6. ***Deployment*** **(Implementação)** – Representa a sexta e última fase metodologia e, neste momento, o modelo a ser utilizado encontra-se previamente selecionado. Esta etapa corresponde à operacionalização (em produção) do modelo selecionado, de forma a ser utilizado para auxiliar no processo de tomada de decisão.

2. REVISÃO DE LITERATURA

A revisão de literatura tem como objetivo apresentar a bibliografia existente em relação aos domínios abordados ao longo da dissertação. Este capítulo conta com o estudo de tópicos como: *Data Science, Data Mining, Machine Learning,* Sistemas de pesagem e Anomalias.

2.1 Estratégia de Pesquisa Bibliográfica

A revisão de literatura é uma etapa fundamental do desenvolvimento da dissertação, que se baseia em analisar e interpretar bibliografia referente a investigações realizadas na área de estudo em questão, com o intuito de averiguar a situação atual em termos dos conhecimentos existentes (Bento, 2012), bem como possibilitar a definição de respostas com um elevado valor científico aos questionamentos que surgem a partir do problema que se pretende solucionar (Snyder, 2019). Deste modo, é necessário estabelecer alguns requisitos para a seleção dos documentos a serem utilizados na dissertação. Esta necessidade surge do facto da bibliografia ser bastante extensa e o tema estar interligado às tecnologias de um setor com bastante inovação, onde os avanços são constantes e rápidos e, por essa razão, muitas vezes os conteúdos acabam por se tornar obsoletos.

O princípio base da estratégia de pesquisa bibliográfica tem como pressuposto, a recolha de referências de uma forma inteligente e que possibilite a análise e obtenção de informação fidedigna, criteriosa, credível, atual e com valor científico, sobre o domínio em estudo. Deste modo, foi selecionado um conjunto de plataformas como por exemplo, o *Google Scholar*, *Repositorium, Scopus, IEEE Xplore* e *Elsevier's Science Direct,* com o intuito de procurar por publicações científicas, tais como, livros, artigos e *papers,* dos quais se pudessem retirar informações preciosas e importantes para a compreensão do problema e possíveis resoluções para o mesmo. No entanto, a procura de referências não se baseou somente nas plataformas abordadas, sendo também realizadas algumas pesquisas em *websites,* normalmente associados à documentação das tecnologias utilizadas para solucionar o problema.

Outro fator de elevada importância está relacionado com a data de publicação dos elementos textuais utilizados como referência, visto que, a evolução das tecnológicos se procede a uma elevada velocidade, sendo fundamental procurar informações nas publicações mais recentes (dentro de um espetro temporal realista). Assim, optou-se por recolher bibliografia publicada a partir do ano de 2010, salvo determinadas exceções, referentes a publicações de grande relevância e de interesse científico.

Para além da componente temporal, o número de citações apresentado foi também um fator preponderante para a seleção de referências bibliográficas.

Para as pesquisas desenvolvidas, foi adotado um conjunto de palavras-chave associadas às temáticas abordadas nesta dissertação como, por exemplo, *Data Science, Data Mining, Machine Learning;* Indústria 4.0, Sistemas de pesagem, Sistemas ciber-físicos, e Anomalias. O uso dos termos enumerados possibilitou a recolha de bibliografia atual, tendo esta transmitido informações relativas a conceitos relevantes de cada uma das temáticas, bem como metodologias e ferramentas a serem utilizadas para a realização dos estudos.

2.2 Mapa Conceptual

O Mapa Conceitual ilustrado na Figura 2 consiste em dar a conhecer os conceitos que foram abordados ao longo da Revisão de Literatura apresentada, bem como demonstrar a forma como os vários temas se relacionam entre si.

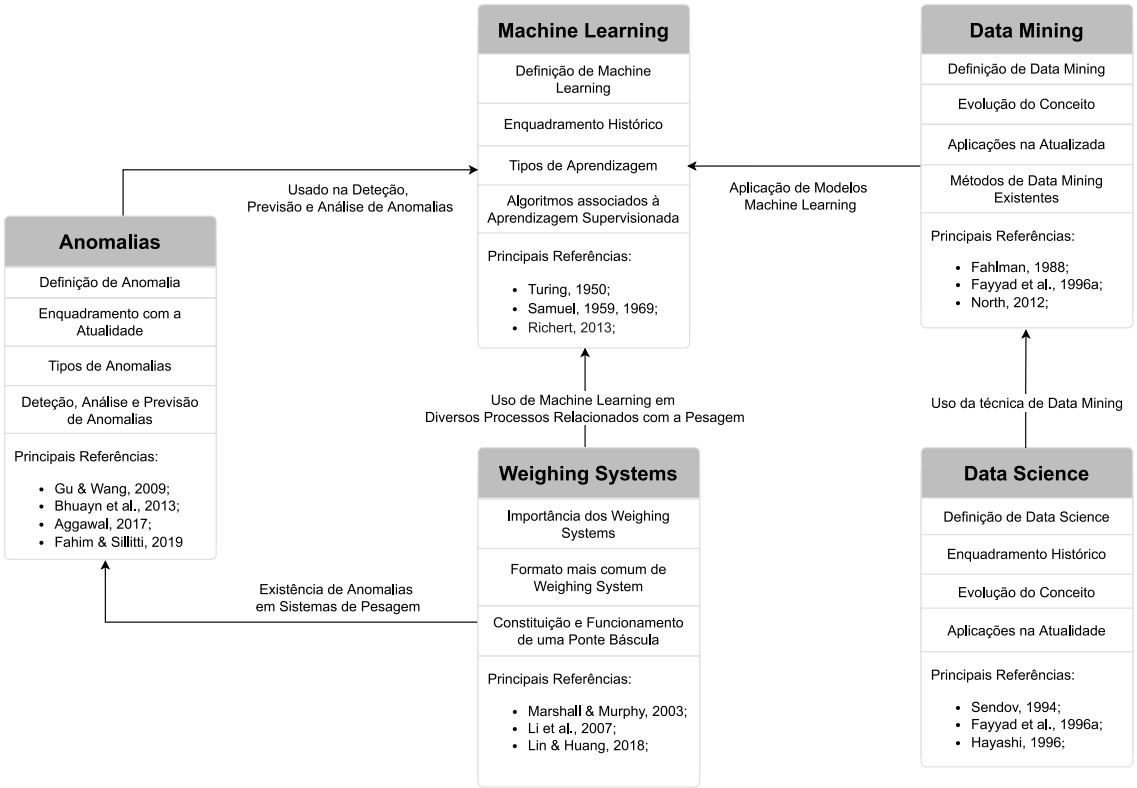

Figura 2 - Mapa Conceptual

2.3 Data Science

Atualmente, os dados são considerados uma enorme fonte de valor para a sociedade e para as organizações, uma vez que o conhecimento existente nos mesmos permite criar vantagem competitiva, inovação e diferenciação (Silwattananusarn & Tuamsuk, 2012). Tal facto, leva a que nesta "Era da Informação", os dados sejam recolhidos, armazenados, processados, partilhados e interpretados, com base num crescimento exponencial (Sendov, 1994; Fayyad et al., 1996a; Cortez & Santos, 2013; Lyman & Varian, 2003). Deste modo, devido à necessidade de usar os dados como uma fonte interminável de conhecimento, aliado ao crescente número de investigações relacionadas com este domínio, conceitos como o *Data Science* adquiriram um grande destaque e atenção (Emmert-Streib & Dehmer, 2019).

Por volta da década de 60, surgiu a necessidade de criar uma "ciência" cujo objeto de estudo e de interesse consistisse em extrair conhecimento e a aprendizagem com base em dados (Donoho, 2017). Todavia, somente no final do séc. XX e início do séc. XXI, através dos esforços realizados por John Chambers, Jeff Wu, William S. Cleveland e Leo Breiman, surge o conceito de *Data Science*. Esta denominação atribuída por William S. Cleveland provém da vontade de separar da estatística clássica, esta forma de extrair conhecimento a partir dos dados. (Donoho, 2017).

Desde então, são várias as definições atribuídas ao conceito de *Data Science.* Hayashi (1996) define *Data Science* não apenas como um simples conceito de análise de dados, técnicas e métodos utilizados, mas como um processo agregado com vista a obtenção de resultados. Smith (2006), afirma que *Data Science* é um estudo baseado na aquisição, análise, processamento e filtragem de dados, com o objetivo de se desenvolver conhecimento através dos relacionamentos existentes entre os mesmos. Finalmente, segundo Van der Aalst (2014), *Data Science* corresponde a um conceito agregador de técnicas, elementos e construções provenientes de variadíssimas áreas de conhecimento, como por exemplo, matemática, computação avançada e visualização de dados, com o único propósito de extrair informação, significado e conhecimento dos dados.

No presente, o conceito de *Data Science* apresenta um grande impacto devido a uma enorme necessidade de evidenciar diversos padrões na sociedade (Semeler et al., 2019). Para tal é fulcral e um fator diferenciador compreender o passado, o presente e prever o futuro, com o objetivo de se obter conhecimento e informação para a tomada de decisão (Salazar-Reyna, et al., 2020).

2.4 Data Mining

O conceito de *Data Mining* surgiu por volta do final da década de 80, correspondendo a uma das várias etapas do processo *Knowleadge Discovery in Databases* (KDD), o qual tinha como objetivo identificar padrões e comportamentos nos dados presentes em bases de dados (Fahlman, 1988). Com o avançar dos anos o termo KDD começou a perder relevância e foi sendo substituído por outros conceitos, contudo o mesmo não se verificou com o conceito de *Data Mining*, uma vez que é utilizado até ao presente (North, 2012). Neste momento a noção de *Data Mining* é aplicada em diversos contextos e domínios, sendo visível o uso deste processo em organizações ligadas a várias áreas do quotidiano, como por exemplo a saúde e indústria, de forma a captar informação atempada, útil e necessária à tomada de decisão (Fayyad et al., 1996b).

No que se refere ao *Data Mining,* existem dois objetivos primários sendo estes: a descrição, que corresponde à descoberta de padrões e ligações entre os dados, e a previsão, isto é, o uso de variáveis existentes nos dados, de forma a prever-se uma outra variável cujo valor é desconhecido (Silwattananusarn & Tuamsuk, 2012). A maneira de se atingir os objetivos relacionados com a descrição e previsão de elementos através da aplicação do processo de *Data Mining* passa pela implementação de um dos seguintes métodos (Fayyad et al., 1996a):

- ***Classification*** – tem como objetivo a avaliação e análise dos vários atributos que constituem um elemento, de maneira a ser capaz de inserir o mesmo numa ou mais classes definidas de forma prévia;
- ***Regression*** – método de aprendizagem baseado na identificação de padrões entre as variáveis, possibilitando a realização de previsões com base em valores numéricos;
- ***Clustering*** – técnica de aprendizagem analítica não supervisionada, capaz de identificar conjuntos de características ou de *clusters* que são, posteriormente, utilizados como forma de descrever os dados;
- ***Summarization*** – corresponde a um conceito baseado em técnicas, como por exemplo, regras de associação, capazes de descrever os dados por inteiro ou um subconjunto dos mesmos;
- ***Dependency Modeling*** – técnica que se baseia na procura de um modelo, cuja função passa por descrever as relações existentes entres os dados;
- ***Change and Deviation Detection*** – baseia-se na deteção de alterações significativas que ocorrem, com base em valores e medições previamente realizadas.

2.5 Machine Learning

Nas últimas décadas, os avanços tecnológicos existentes associados à necessidade de crescimento e inovação das organizações tecnológicas e industriais levou a um crescimento exponencial dos conceitos associados a *Artificial Intelligence* (AI). De uma forma geral, o conceito de *Artificial Intelligence*, refere-se à capacidade de tornar as máquinas aptas a funcionar como o cérebro Humano (Shinde & Shah, 2018). Esta corresponde ao estudo de *"Intelligent Agents"*, dispositivos capazes de entender o ambiente, aprender padrões, solucionando problemas e executar funções Humanas, de modo a atingir os objetivos para os quais foram definidos (Shinde & Shah, 2018; Fang et al., 2018).

Atualmente, existe um aumento gradual do espaço académico, científico e industrial reservado a esta área de conhecimento e respetivas tecnologias associadas (Figura 3), derivado do aparecimento de novos conceitos como a Indústria 4.0 e a Transformação Digital. Este facto, levou à criação de projetos maioritariamente de investigação, os quais, levam ao desenvolvimento de uma perceção cada vez mais aprofundada dos conceitos, o que permite mais facilmente perceber todo o conjunto de benefícios que as organizações podem retirar para se diferenciarem das restantes.

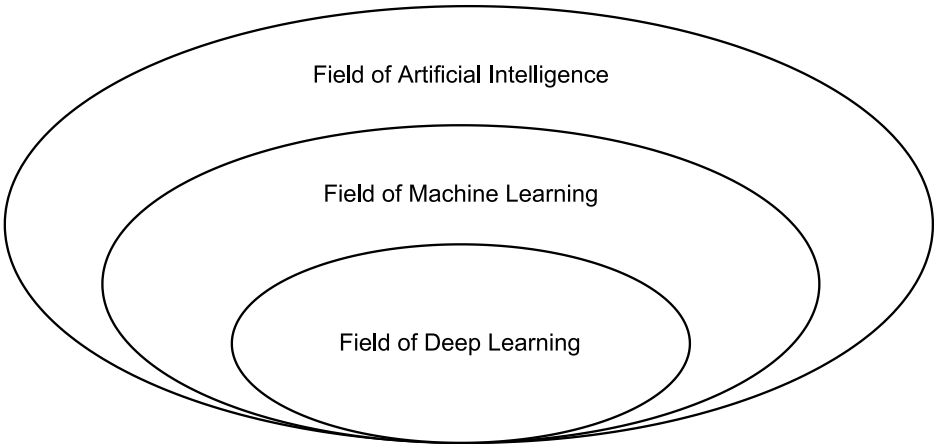

Figura 3 - Enquadramento das Tecnologias de AI (Baseado em Shinde & Shah, 2018).

Machine Learning tem a sua origem em 1950, quando Turing através da questão *"Can machines think?"* e do conceito "*Learning Machines*" dá início ao desenvolvimento de métodos, técnicas e algoritmos capazes de aprender e de melhorar por si mesmos, aproximando assim a inteligência da máquina à inteligência humana (Turing, 1950). Todavia, o termo *Machine Learning (ML)* foi apenas concebido em 1959 por um pioneiro na área de *computer gaming* e inteligência artificial denominado de Arthur Samuel (1959). Na sua obra "Some studies in machine learning using the game of checkers",

Arthur Samuel realizou uma revisão dos seus trabalhos, mais concretamente demonstrando a aplicação das suas técnicas de aprendizagem automáticas em termos do "game of checkers", ou em português, "jogo das damas" (Samuel, 1969).

Os estudos apresentados consistiam em mostrar como uma máquina, recorrendo apenas às regras do jogo e à inserção de certos parâmetros, foi capaz de desenvolver uma estratégia adequada e aprimorada para ganhar o jogo em questão (Samuel, 1969). No entanto, a máquina apresentou uma limitação que foi a incapacidade de vencer o melhor jogador humano do mundo. Contudo, tantos os conceitos técnicos, bem como os avanços apresentados marcaram a história, mostrando um mundo de possibilidades em aberto através do uso da capacidade de processamento de uma máquina (Samuel, 1969).

Nos dias de hoje, *Machine Learning* tem como principal objetivo ensinar máquinas a desempenhar uma dada tarefa, fornecendo às mesmas exemplos específicos de como a realizar (Richert, 2013). Além disso, uma das capacidades inerentes ao *Machine Learning* é a forma como o mesmo é capaz de detetar e identificar padrões nos dados históricos, com o intuito de desenvolver perspetivas de futuro, i.e., previsões (Vieira et al., 2019).

A capacidade de facilmente compreender e de interpretar os dados com base em exemplos específicos permite e potencia o uso de *Machine Learning* em diversos mecanismos do quotidiano como: Sistemas de Deteção de Fraudes; Manutenção Preditiva e *Ad Placement* (Domingos, 2012). Este facto permitiu o desenvolvimento de diversas técnicas e tipos distintos de *Machine Learning* devido aos conhecimentos e avanços tecnológicos (Kotsiantis et al., 2007): Aprendizagem Supervisionada; Aprendizagem Não-Supervisionada; Aprendizagem Semi-Supervisionada; Aprendizagem Ativa; e, Aprendizagem por Reforço.

2.5.1 Aprendizagem Supervisionada

Segundo Praveena e Jaiganesh (2017), a Aprendizagem Supervisionada corresponde a uma aprendizagem que tem como objetivo desenvolver a capacidade de atribuir significado ou rotular dados, através da aprendizagem de dados de treino que são usados como exemplos. De uma forma simplificada, existe um par *input-output*, e um conjunto de dados de treino representativos do par. Após o treino do algoritmo com os dados de exemplo, o mesmo torna-se capaz de prever ou classificar um *target (output)*, por meio dos dados fornecidos como *input* e dos padrões comportamentais dos mesmos.

Tendo em consideração o tipo e natureza dos dados utilizados como *target* do modelo treinado, é possível separar os problemas existentes e relacionados com Aprendizagem Supervisionada em dois tipos distintos: problemas de classificação e problemas de regressão (Praveena & Jaiganesh, 2017). O que difere o tipo de Aprendizagem Supervisionada a ser utilizada com o objetivo de dar resposta a um dado problema é a natureza do *output*. Em problemas associados à classificação, como por exemplo, se um movimento bancário corresponde a uma fraude ou não, a variável binária utilizada é de natureza discreta. Por outro lado, caso o problema seja associado a uma regressão, como por exemplo, prever o preço de um dado voo, o valor que se pretende obter como *output* é contínuo.

Apesar de existirem diferentes tipos de Aprendizagem Supervisionada, o objetivo de ambas passa por visualizar o que possivelmente irá ocorrer num dado evento futuro. Assim, existem diversos algoritmos que podem ser utilizados para implementar modelos com o objetivo de dar resposta aos problemas propostos como a seguir descritos:

- ***Decision Trees*** – é um algoritmo que pode ser utilizado em ambos os tipos de Aprendizagem Supervisionada, ou seja, tanto em termos de classificação como de regressão. Segundo estudos estatísticos, este algoritmo é o mais amplamente utilizado em projetos de *Data Mining* (Li & Zhang, 2010), devido à sua capacidade de transpor a *"black box"* presente em modelos de *Machine Learning*, permitindo assim extrair e exibir de uma forma fácil as regras de decisão elaboradas pelo algoritmo (Kamel & Selim, 1994). Além disso, uma outra característica que torna este algoritmo bastante utilizado é o seu baixo nível de complexidade e de processamento computacional. De uma forma sucinta, uma *Decision Tree* (Figura 4) consiste num classificador capaz de realizar repartições, recursivamente, das instâncias fornecidas ao algoritmo (Nasteski, 2017). Ou seja, o classificador é formado por *nodes*, que em conjunto formam a *root tree*, a qual não apresenta qualquer *edge* contrariamente aos restantes *nodes*, que se podem denominar de *test nodes* caso apresentem *outgoing edges* ou caso contário, *leaf nodes* (Nasteski, 2017). No decorrer do treino do algoritmo, os padrões existentes nos dados são descobertos, permitindo a criação dos vários *decision nodes* aos quais estão associados um determinado valor numérico, discreto ou contínuo,que corresponde ao *output* que dará resposta ao problema de classificação ou de regressão apresentado.

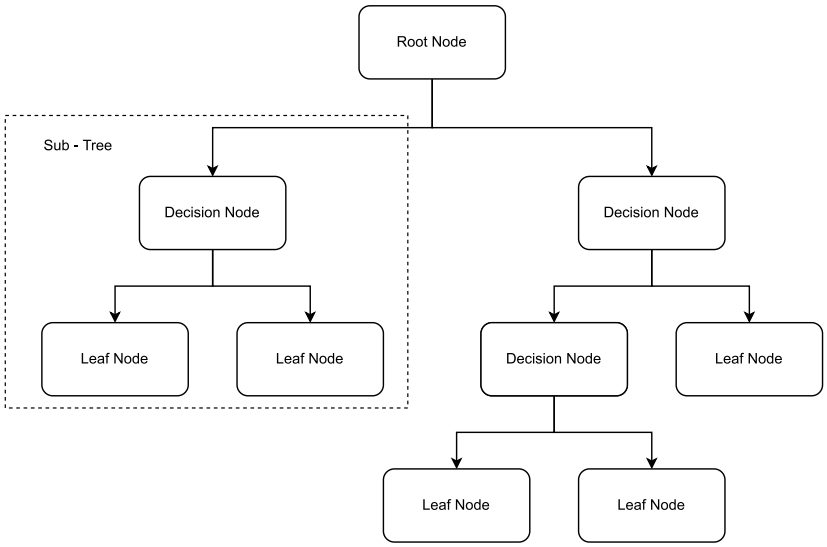

Figura 4 - Arquitetura Decision Tree (Baseado em Charbuty and Mohsin Abdulazeez, 2021).

- ***Random Forest*** – este método de aprendizagem foi proposto por Breiman (2001) para dar resposta a problemas de classificação e de regressão. Segundo Carvalho et al. (2019), o *Random Forest* (Figura 5) utiliza diferentes partes dos dados fornecidos para criar um *"ensemble"* de árvores de decisão aleatórias, que por sua vez irão realizar a previsão de um dado valor. Desse modo, para problemas de classificação o valor do *output* corresponde àquele que mais vezes aparecer como resultado de cada uma das várias árvores de decisão, e para problemas de regressão, é obtido o valor que corresponde à média dos *outputs* apresentados pelas várias árvores de decisão (Carvalho et al., 2019).

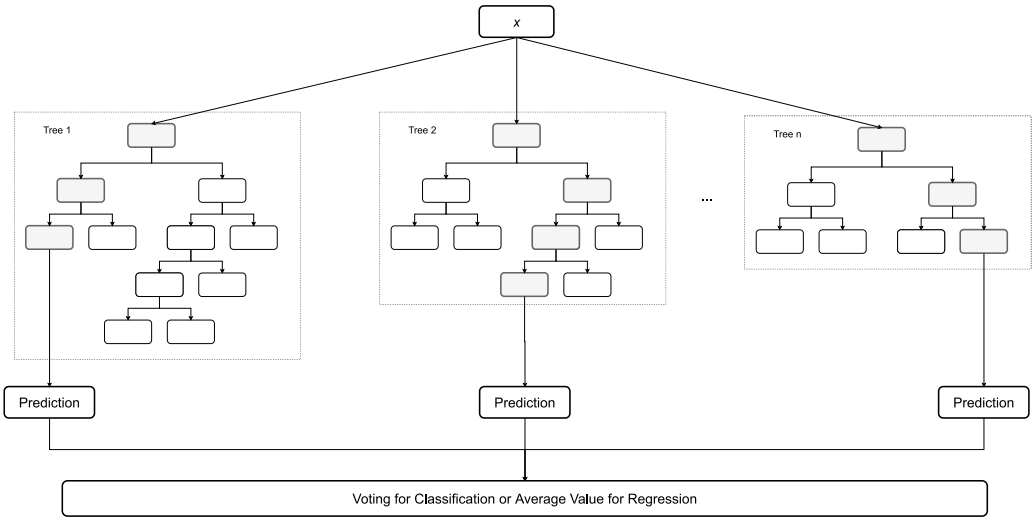

Figura 5 - Arquitetura Random Forest (Baseado em Belgiu & Drăguţ, 2016).

- ***Gradient Boosting Tree*** – surge com base na estratégia genérica e de aprimoramento do poder preditivo de modelos de aprendizagem denominada *Boosting*, podendo ser aplicado a problemas de classificação e de regressão (Freund et al., 1999; Friedman, 2001; Hastie et al., 2009). Segundo Shoaran et al. (2018), a par do Random Forest, o *Gradient Boosting* é um dos algoritmos mais versáteis e competitivos em termos da resolução de problemas de *Machine Learning*, especialmente em ambientes cuja existência de dados para treino é bastante reduzida. De forma sucinta, o algoritmo ilustrado na Figura 6 faz tipicamente a utilização de árvores de decisão que normalmente apresentam um tamanho fixo, e que atuam num espaço binário, denominadas de *"Weak Classifier"*, que ao funcionarem em conjunto produzem *"Strong Classifier"* (Shoaran et al., 2018; Hastie et al., 2009). Desse modo, cada iteração utiliza o *"Prediction Residual"* da iteração anterior como base da sua aprendizagem, visando reduzir o erro apresentado pelo algoritmo ao longo das várias iterações.

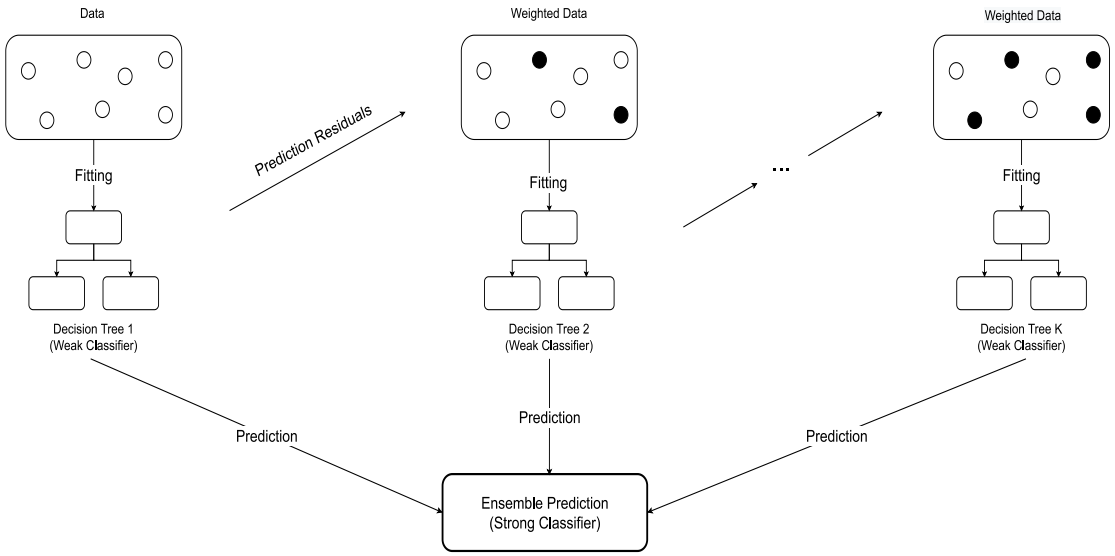

Figura 6 - Arquitetura Gradient Boosting Tree (Baseado em Deng, 2021).

- ***eXtreme Gradient Boosting (XGBoost)*** – o *XGBoost* é um algoritmo bastante adequado para a elaboração de modelos de classificação (Dhaliwal et al., 2018), uma vez que corresponde a uma implementação mais eficiente do *Gradient Boosting Tree* (Friedman et al., 2000). Este algoritmo baseia-se em múltiplas árvores de decisão e difere de outros, como o Random Forest, na forma como a aprendizagem ocorre de forma gradual ao longo das várias árvores de decisão (Wang et al., 2019), conforme ilustrado na Figura 7, uma árvore de decisão existente neste

algoritmo. O processo de aprendizagem tem por base os resultados obtidos a partir das restantes árvores de decisão. Deste modo, o resultado que o algoritmo proporciona corresponde ao somatório de todos os resultados apresentados ao longo das várias árvores de decisão presentes no mesmo (Wang et al., 2019).

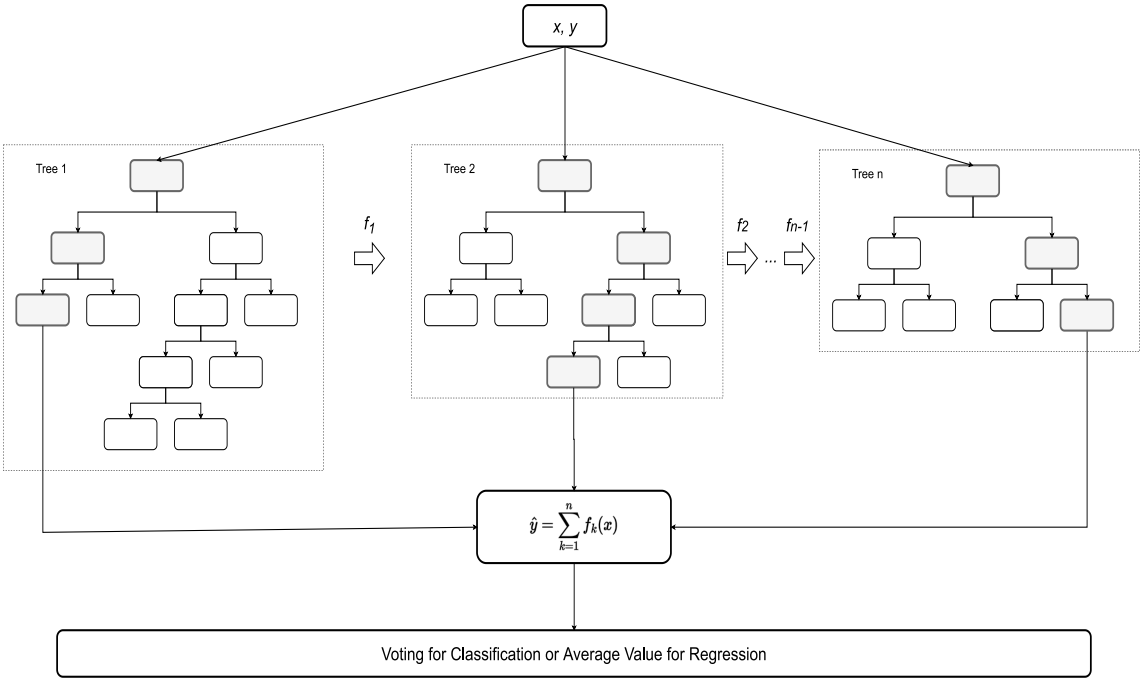

Figura 7 – Arquitetura XGBoost (Baseado em Wang et al., 2019).

- ***Support Vector Machines*** – são modelos de *Machine Learning* baseados nos modelos estatísticos desenvolvidos em Vapnik (1999), capazes de executar tarefas relacionadas com problemas de classificação e de regressão. No que se refere a estes modelos de *Machine Learning*, o seu objetivo passa por transformar as entradas para um dado espaço imaginário através de um método matemático denominado de *Kernel* (Lorena, & de Carvalho, 2007). Este processo possibilita que os pontos existentes sejam linearmente separados, encontrando assim um hiper-plano, como ilustrado na Figura 8. O hiper-plano consiste num espaço multidimensional que permite uma separação perfeita das classes existentes (Gandhi, 2018). Este algoritmo apresenta uma enorme capacidade de solucionar problemas complexos relacionados com os vários tipos de Aprendizagem Supervisionada, fazendo deste um dos algoritmos mais populares (Karatzoglou et al., 2006).

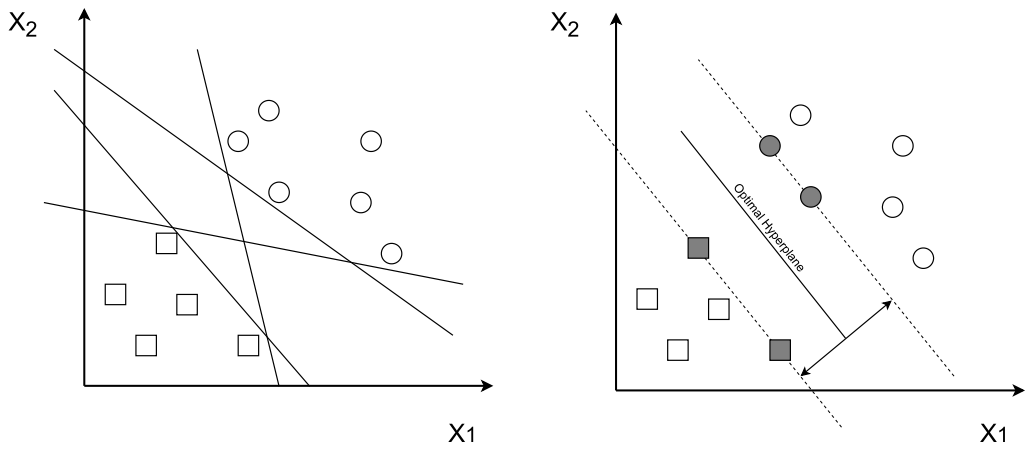

Figura 8 - Arquitetura Support Vector Machines (Baseado em Gandhi, 2018).

- ***Multilayer Perceptron*** – é um dos vários tipos distintos de redes neuronais e pertence à taxonomia de *Feed-Forward Networks* (Gardner & Dorling, 1998), sendo uma variação do *Perceptron Model* proposto por Rosenblatt (1958). Esta forma de rede neuronal ilustrada na Figura 9 consiste num sistema de neurónios, ou *nodes,* conectados entre si e separados por, no mínimo, 3 *layers* distintos *(input layer, hidden layer, output layer),* podendo este valor variar consoante o número de *hidden layers* definidos. O *Multilayer Perceptron* é um modelo capaz de responder a problemas de classificação e de regressão em diversas áreas do conhecimento (Ramchoun et al., 2016).

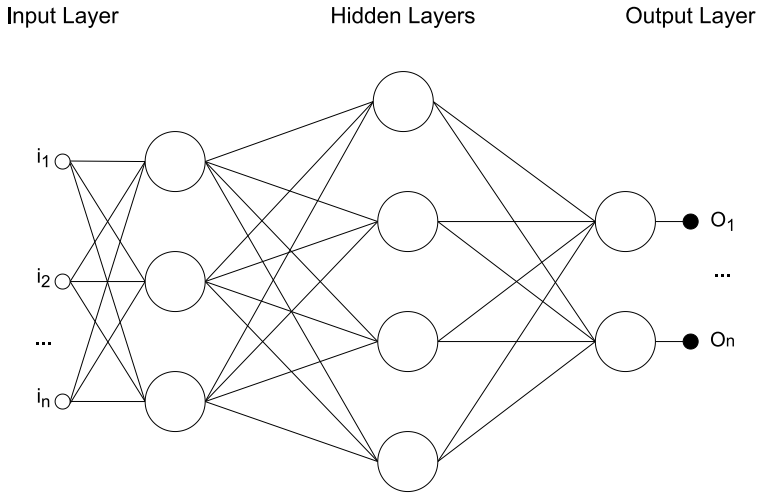

Figura 9 - Arquitetura MultiLayer Perceptron (Baseado em Gardner & Dorling, 1998).

2.5.2 Aprendizagem Não-Supervisionada

A Aprendizagem Não-Supervisionada, muitas vezes referenciada como *Clustering*, é um dos vários métodos de *Machine Learning* e baseia-se essencialmente na descoberta de classes ocultas por entre os padrões dos dados (Gentleman & Carey, 2008). Ao contrário da Aprendizagem Supervisionada, esta forma de aprendizagem não é constituída por nenhuma variável que possa ser supervisionada *(target)* (James et al., 2013).

Este método de aprendizagem é caracterizado por uma enorme capacidade de detetar relações e padrões existentes nos dados, possibilitando uma agregação e organização dos mesmos por classes ou grupos de dados com características semelhantes ou aproximadas, que permite a visualização e descrição dos dados em termos de regras e *clusters* (Kotsiantis et al., 2007; Nelson, 2020). Atualmente são vários os algoritmos de Aprendizagem Não-Supervisionada, contudo aquele que apresenta uma maior popularidade e utilização, devido às suas características e capacidades, é o algoritmo *K-Means* (Lloyd, 1957; MacQueen, 1967; Dhanachandra et al., 2015).

2.5.3 Aprendizagem Semi-Supervisionada

A Aprendizagem Semi-Supervisionada é um ramo de *Machine Learning* situado entre a Aprendizagem Supervisionada e a Aprendizagem Não-Supervisionada, que utiliza dados com e sem *label data*, para realizar tarefas associadas ao processo de aquisição de conhecimento (Van Engelen & Hoos, 2020). Assim sendo, é possível afirmar que a Aprendizagem Semi-Supervisionada corresponde a um cruzamento, no que se refere à realização de tarefas das duas formas de aprendizagem (Chapelle et al. 2006; Zhu 2008).

De uma forma geral, a aplicação de uma Aprendizagem Semi-Supervisionada está relacionada com problemas de classificação onde existe uma certa escassez de dados com *labels* (Van Engelen & Hoos, 2020). Contudo, existindo um vasto conjunto de dados sem *labels* à disposição, é possível utilizar estes dados e as suas suposições com o intuito de melhorar o classificador desenvolvido (Van Engelen & Hoos, 2020). Além disso, é possível utilizar o método de Aprendizagem Semi-Supervisionada em cenários onde não existe escassez de dados com *labels*, uma vez que, a utilização dos dados sem *labels* permite uma melhor generalização dos dados nunca vistos, aumentando assim a *performance* dos modelos desenvolvidos (Berthelot, et al., 2019).

2.5.4 Aprendizagem Ativa

No que se refere a previsões, mais concretamente em relação à *Accuracy* que envolve este tipo de aprendizagem, a escolha e utilização de determinados modelos de *Machine Learning* não é o único fator preponderante para os resultados obtidos, sendo também relevante a existência de uma quantidade assinalável de dados de treino, com uma distribuição adequada das classes (Gubaev et al., 2018). Desse modo, conceitos como Aprendizagem Ativa começaram a surgir e a desenvolver-se.

A Aprendizagem Ativa, por vezes denominada como *Query Learning* é um tipo de aprendizagem inerente a *Machine Learning* (Settles, 2009), que se baseia em fornecer ao *learner* (modelo de *Machine Learning)*, o controlo sobre os dados que vai utilizar para aprender (Olsson, 2009). Este tipo de processo de aprendizagem apresenta a capacidade de consultar um *oracle* ("professor" com grande conhecimento sobre o domino em que os dados se inserem) com o objetivo de atribuir significado e criar *labels* sobre dados que não os tenham (Olsson, 2009; Settles, 2009). Este método utiliza um conjunto de dados com e sem *labels* e tem como *output* um classificador capaz de elaborar previsões, bem como uma nova porção de dados com *label* (Olsson, 2009). Como tal, não há custos adicionais no que se refere ao processo de criação de *labels* nos dados, nem a necessidade de fornecer novos dados para melhorar o desempenho dos modelos (Olsson, 2009).

2.5.5 Aprendizagem por Reforço

Esta forma de aprendizagem tem por objetivo estudar diversos tipos de ambientes onde sistemas naturais ou artificiais aprendem com base em consequências das suas escolhas (Dayan & Niv, 2008). Segundo Kaelbling et al. (1996), esta forma de aprendizagem apoia-se na recompensa e na punição para que os agentes aprendam comportamentos através de um processo dinâmico de tentativa-erro. Este facto, permite que a atuação dos agentes não seja influenciada pela estrutura adjacente aos dados, mas sim pela bonificação ou punição da sua ação perante um dado comportamento. Assim sendo, esta aprendizagem é capaz de lidar facilmente com comportamentos que não estejam presentes no conjunto de dados de treino (Sutton & Barto, 2018).

Existem duas abordagens que podem ser adotadas para resolver problemas utilizando a aprendizagem por reforço (Kaelbling et al., 1996). A primeira tem como objetivo encontrar um comportamento que apresente um elevado desempenho dentro do espaço de procura. A segunda passa pela utilização de técnicas, mecanismos e modelos estatísticos, com o objetivo de estimar qual será a recompensa e utilidade da escolha de um dado comportamento.

2.6 Sistemas de Pesagem

Os sistemas de pesagem são parte fundamental do quotidiano tanto da sociedade, como das organizações (Marshall & Murphy, 2003), uma vez que é importante saber exatamente o peso dos produtos e objetos com os quais se está a lidar (Halimic & Balachandran, 1995), e consequentemente daí retirar informação útil com o objetivo de obter benefícios económicos e aprimorar a qualidade dos serviços prestados (Lin & Huang, 2018). O transporte de mercadorias através de veículos pesados é uma das atividades cuja eficiência do seu funcionamento está totalmente e diretamente dependente do processo de pesagem (Lin & Huang, 2018).

Os sistemas de pesagem associados ao setor dos transportes são uma prática bastante comum e que se tornou bastante popular ao longo dos anos (Marshall & Murphy, 2003). São vários os motivos que levam à necessidade de pesar uma viatura de mercadorias, como por exemplo, garantir que a quantidade de matéria transportada entre fornecedor e cliente é realmente a estipulada e expectável por ambas as partes e assegurar que o veículo que realiza o transporte da mercadoria não se encontra com excesso de peso, sendo este um dos principais fatores de acidentes rodoviários em veículos pesados de mercadorias (Li et al., 2007).

No que concerne estes sistemas, a pesagem de veículos pode adotar diversos formatos, mas o mais comum é denominado de Ponte Báscula (Marshall & Murphy, 2003). Esta "ponte" utilizada para a pesagem de veículos é constituída por uma plataforma de metal, um conjunto de células de carga, uma caixa de ligações, e, um terminal de leitura (Figura 10). De forma sucinta, quando um veículo se coloca na posição de pesagem em cima da plataforma de metal, a carga do mesmo é distribuída para cada uma das células de carga que dão suporte à plataforma (Marshall & Murphy, 2003). Em seguida, as células de carga enviam um sinal elétrico através da caixa de ligações, que é transformado numa leitura do peso a partir de um calculo matemático no terminal de pesagem (Halimic & Balachandran, 1995).

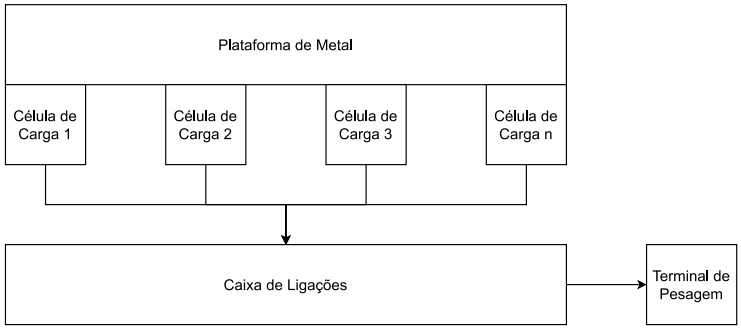

Figura 10 - Diagrama simplificado de conexões em WS (Baseado em Soc. Coop. Bilanciai Campogalliano, 2009).

2.7 Anomalias

O desenvolvimento tecnológico visível ao redor do mundo e assente em conceitos como a *Internet of Things* e a Transformação digital, têm proporcionado grandes quantidades de dados, os quais são constantemente armazenados e interpretados pelas organizações (Fahim & Sillitti, 2019). O contínuo aumento do volume de dados armazenados pelas organizações tem por consequência o aparecimento cada vez mais frequente de anomalias associadas a estes sistemas baseados em informação, as quais apresentam normalmente complicações em processos críticos da cadeia de valor das organizações (Fahim & Sillitti, 2019). Com base na previsão, compreensão e deteção destas anomalias, as complicações que estas apresentam para as organizações diminui, o que afeta positivamente a *performance* em termos de processos organizacionais (Fahim & Sillitti, 2019).

O estudo das anomalias, as quais podem também ser referenciadas como desvios, é um ramo importante e dinâmico de *Machine Learning* (Bhuyan et al., 2013), tendo o seu estudo e desenvolvimento vindo a ser aprimorado ao longo do tempo (Arning et al., 1996). Segundo Aggarwal, (2017), anomalias correspondem a anormalidades, desvios e *outliers* nos dados referentes a um dado processo específico. Desse modo, os algoritmos de *Machine Learning*, quando aplicados a este campo de estudo, têm como objetivo detetar ou prever comportamentos e padrões nos dados que apresentem uma discrepância entre o valor esperado e o valor real apresentado.

A ocorrência de anomalias está associada a diversos fatores, contudo estas também podem ocorrer aleatoriamente, devido a fatores externos e não compreendidos como, por exemplo, por exemplo, a inserção incorreta de valores acidentalmente ou até mesmo intencionalmente com o objetivo de obter algum beneficio de forma fraudulenta. No entanto, é preciso compreender que a presença de uma anomalia não significa necessariamente a existência de um erro ou problema, por exemplo, um maior fluxo de clientes num dado estabelecimento pode corresponder a um aumento súbito e isolado do faturamento, sendo tal considerado uma anomalia, mas não negativa.

Atualmente, projetos e estudos relacionados com anomalias podem ser observados em diversas áreas como, por exemplo, ciber-segurança e indústria (Zenati et al., 2018), este facto ocorre devido aos impactos e resultados positivos apresentados pela aplicação destes estudos no mundo real. Esta conceção cientificamente comprovada e situada num grau de entendimento elevado demonstra uma grande complexidade no desenvolvimento de métodos e estratégias que pretendam dar resposta a problemas de grande dimensão (Zenati et al., 2018).

Como abordado anteriormente, os projetos organizacionais relacionados com anomalias apresentam um papel fundamental e transversal a diversas áreas. As organizações precisam de ter consciência dos problemas que enfrentam, de forma a selecionar, desenvolver e aplicar um conjunto de ferramentas e tecnologias que permitam responder aos dilemas presentes (Fahim & Sillitti, 2019).

A análise a ser elaborada pelas organizações tende a seguir o fluxo de eventos ilustrado na Figura 11. Este tem como objetivo compreender os dados captados e a forma como esta atividade foi realizada, determinar os tipos de anomalias presentes, quais os dados que serão utilizados para treinar os modelos de *Machine Learning;* que modelos serão utilizados e qual o real objetivo associado às anomalias que se pretende atingir (Fahim & Sillitti, 2019). Assim, somente com uma correta implementação deste fluxo de eventos se consegue obter a determinação dos conjuntos de dados considerados normais ou alarmísticos, no processo organizacional estudado (Fahim & Sillitti, 2019).

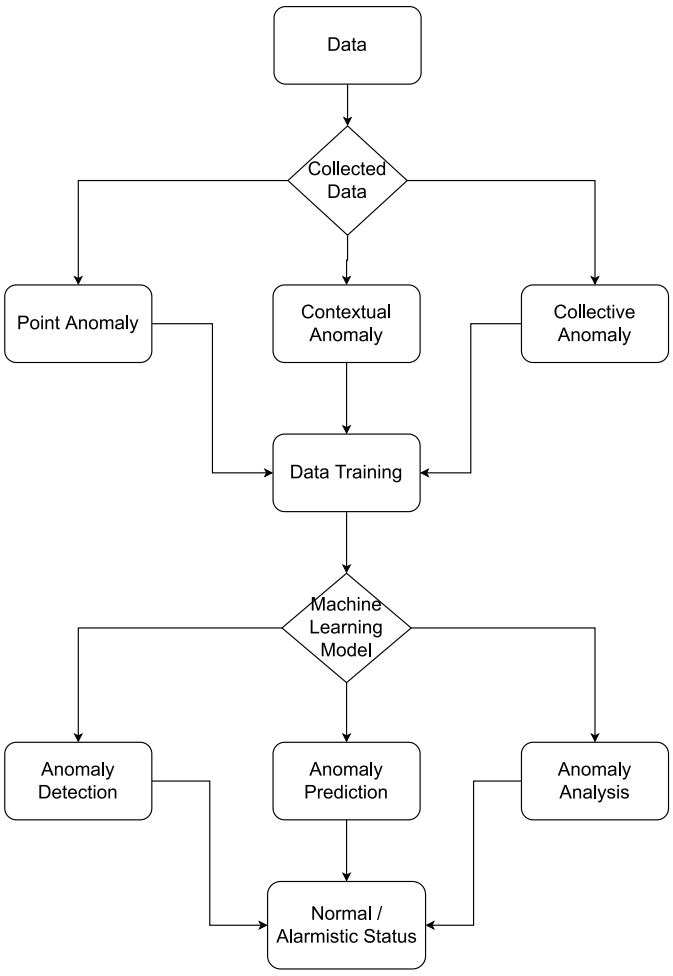

Figura 11 - Fluxo de um Sistema de Deteção de Anomalias (Baseado em Fahim & Sillitti, 2019).

2.7.1 Tipos de Anomalias

As anomalias presentes num conjunto de dados podem variar consoante o seu tipo. Atualmente, estão definidos três tipos distintos de anomalias dos dados: Anomalia pontual ou *Point Anomaly,* Anomalia contextual ou *Contextual Anomaly,* Anomalia coletiva ou *Collective Anomaly.*

- **Anomalia Pontual ou *Point Anomaly*** – também conhecido como *outlier*, é um tipo de anomalia (Figura 12), na qual um ponto específico nos dados apresenta uma elevada discrepância em relação ao conjunto dos restantes (Aggarwal, 2017).

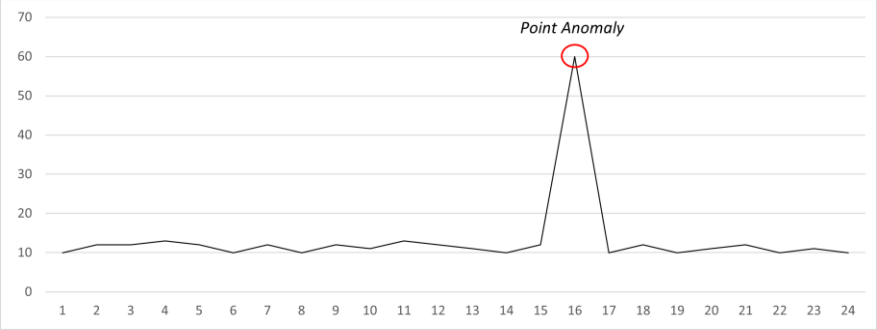

Figura 12 - Exemplo de uma Anomalia pontual ou Point Anomaly (Baseado em Fahim & Sillitti, 2019).

- **Anomalia contextual ou *Contextual Anomaly*** – é um tipo de anomalia determinada a partir do contexto em que o objeto em estudo. Neste tipo de anomalias, existe a necessidade de compreender a conjuntura em redor da discrepância dos valores apresentados, existindo a possibilidade de a variação ser considerada ou não uma anomalia (Song et al., 2007). A Figura 13 ilustra a ocorrência de variações nos dados e o papel que o contexto tem na avaliação da presença ou ausência de uma anomalia baseada em condições (Song et al., 2007).

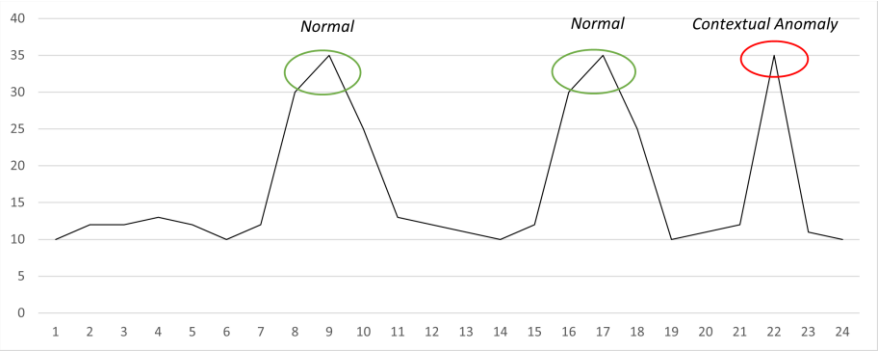

Figura 13 - Exemplo de uma Anomalia contextual ou Contextual Anomaly (Baseado em Fahim & Sillitti, 2019).

- **Anomalia coletiva ou *Collective Anomaly*** – é uma anomalia que é apenas confirmada perante a existência de uma sequência de dados discrepantes, os quais apontam um comportamento distinto da normalidade, como se pode confirmar na Figura 14. Segundo Ahmed et al. (2016b), a observação de um conjunto de dados completamente distinto dos restantes poderá indicar a presença de uma anomalia coletiva, assim, não é possível a identificação deste tipo de anomalias, caso exista somente um valor discrepante no intervalo sequencial nos dados disponibilizados.

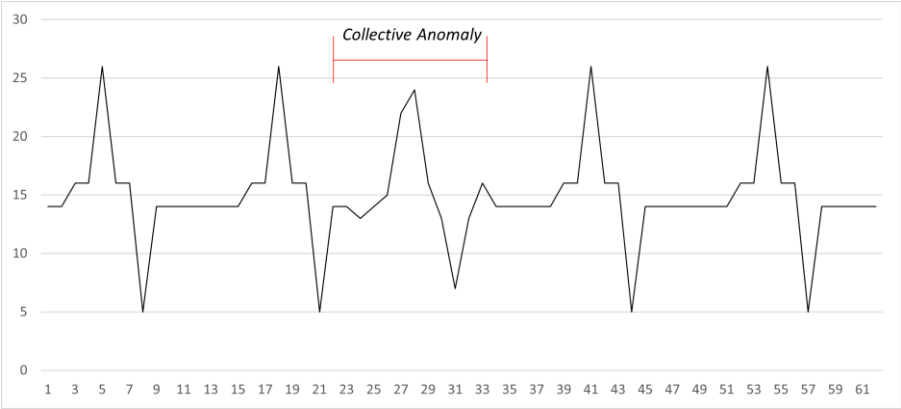

Figura 14 - Exemplo de uma Anomalia coletiva ou Collective Anomaly (Baseado em Fahim & Sillitti, 2019).

2.7.2 Deteção de Anomalias

De acordo com a Figura 11, a Deteção de Anomalias é um dos três tipos distintos de processos que se podem abordar em termos de anomalias. De forma sucinta, a Deteção de Anomalias é uma tarefa que tem como função detetar comportamentos ou padrões anormais num dado conjunto de dados (Ahmed et al., 2016a). Segundo Ahmed et al. (2014), a Deteção de Anomalias está fortemente associada à área de *Data Mining*, de modo que existem diversos estudos e projetos que implementaram modelos de *Machine Learning* e estatísticos.

2.7.3 Análise de Anomalias

O processo de análise de anomalias, muitas vezes referenciado como análise de *Outlier*, corresponde à utilização de métodos e técnicas capazes de descobrir e atribuir significado a comportamentos que não se enquadram nos parâmetros normais definidos (Modi & Oza, 2016). Deste modo, após a deteção de uma anomalia referente a um dado processo, é fundamental compreender tudo em redor da mesma, com o intuito de se redigir uma descrição adequada dos fatores que levaram

ao seu desenvolvimento. Este estudo do ambiente que rodeia o aparecimento da anomalia permite a compreensão de determinados indicadores relacionados a eventos que nunca foram reconhecidos, permitindo uma melhor *performance* na deteção, bem como na previsão do aparecimento de situações anómalas (Agarwal, 2013).

2.7.4 Previsão de Anomalias

Segundo Gu e Wang (2009), o aparecimento de processos operacionais cujo funcionamento decorre sem qualquer pausa, levou ao desenvolvimento de sistemas baseados na Previsão de Anomalias. Diferente da Deteção de Anomalias, que tem sido estudada e desenvolvida ao longo de décadas, o conceito de Previsão de Anomalias é recente e ainda pouco aprofundado pela comunidade científica (Fahim & Sillitti, 2019).

O conceito de Previsão de Anomalias baseia-se na capacidade de usar dados captados no presente, com o objetivo de prever um possível comportamento fora dos parâmetros normais, numa linha temporal futura. Assim sendo, é possível definir a Previsão de Anomalias como um conjunto de técnicas que, através da sua utilização, são capazes de demonstrar a probabilidade de algo anormal ocorrer, sem que o sistema em análise realmente entre num estado de alerta devido à presença de uma anomalia futura (Gu & Wang, 2009).

No que se refere ao conceito de Previsão de Anomalias, a capacidade de elaborar um sistema capaz de utilizar dados do passado para prever a existência futura de comportamentos anómalos, não é uma tarefa considerada fácil (Alonso et al., 2011). Contudo, existem diversas técnicas que quando aplicadas de forma adequada, podem ser utilizadas para dar resposta a este tipo de problemas. Uma técnica bastante utilizada na resolução de problemas fundados na Previsão de Anomalias é a aplicação de modelos de *Machine Learning* baseados em Aprendizagem Supervisionada. Estes modelos, são capazes de encontrar padrões no comportamento dos dados que permitem elaborar uma classificação preditiva baseada na ocorrência ou não de anomalias, o que faz com que as organizações disponham de um período de tempo significativo desde o momento em que se prevê a anomalia até ao momento que esta realmente ocorre. O referido período é fundamental porque permite às organizações a definição de planos de ação que contrariem o problema futuro.

2.8 Trabalhos Relacionados

Esta secção tem como objetivo analisar e explorar contribuições científicas associadas à aplicação de técnicas de *Machine Learning* ou de *Deep Learning* em contextos relacionados com a pesagem de veículos pesados de transporte de mercadorias e matérias-primas. A Tabela 1 apresenta de forma sucinta a revisão de literatura elaborada associada à área de aplicação dos trabalhos desenvolvidos, possibilitando a obtenção de informações relevantes em relação aos artigos analisados.

Assim, Kim et al. (2009) aplicaram o método de *Artificial Neural Networks (ANN),* com o objetivo de aprimorar sistemas *Weigh-In-Motion (WIN)* alojados em pontes*.* Os autores desenvolveram uma *framework* baseada em técnicas de *Machine Learning* que utiliza os dados provenientes de Sistemas de Pontes *Weigh-In-Motion*. Esta *framework* decompõe-se em duas etapas: a primeira consiste em calcular o peso bruto do veículo que circula pela ponte e na segunda é realizado o cálculo do peso por eixo. Posteriormente, a *framework* foi aplicada a diversos tipos de Pontes *Weigh-In-Motion* e os autores adotaram a métrica *Accuracy* para avaliar a capacidade preditiva de cada um dos modelos. Como resultado, obtiveram uma *Accuracy* de 94.7% relativamente ao modelo do cálculo do peso bruto do veículo e uma *Accuracy* de 93.9% para o modelo do cálculo do peso do eixo.

Anos mais tarde, Gungor et al. (2018) propuseram uma *framework* orientada por dados capaz de estimar a taxa de veículos com excesso de carga que circulam no tabuleiro de uma dada ponte utilizando dados *WIN*. Depois combinaram a informação estimada com os dados do *National Bridge Inventory* (NBI), com o intuito de desenvolver um modelo preditivo, cujo objetivo é prever o estado do tabuleiro[1] após a passagem de um veículo com excesso de peso. Os autores adotaram o modelo *Support Vector Machine* (SVM), para a realização de previsões, o qual apresenta uma *Accuracy* (métrica de avaliação) de 73.57%.

Por sua vez, Liu et al. (2019), propuseram uma *framework* baseada em técnicas de *Deep Learning* e visão por computador capaz de classificar veículos de transporte de matérias-primas como carregados com a carga total, ou como vazios. Assim sendo, foram utilizadas 2454 imagens de veículos pesados, as quais foram distribuídas por *sets* de treino, teste e validação no processo de modelação. Os resultados obtidos permitem concluir que o modelo com maior *Accuracy* foi o VGG16-FT, tendo este apresentado um valor de 98.0%.

No mesmo ano, Yan et al. (2019) desenvolveu uma *framework* para prever a probabilidade de uma possível falha relativamente a uma dada ponte, combinando os dados provenientes do estado de

[1] Por vezes denominado de sistema piso, o tabuleiro é um elemento estrutural de uma ponte que pode ser constituído por betão, aço ou concreto, tendo toda a restante estrutura a função de o suportar.

desgaste da mesma com os dados referentes ao tráfego que nela circula, mais concretamente veículos em sobrecarga. Os autores desenvolveram um modelo ANN que permite verificar como a probabilidade de falha na ponte aumenta com o passar do tempo perante diferentes sobrecargas aplicadas à mesma. Como espectável, os resultados apontam para uma maior probabilidade de falha ao longo do tempo quanto maior o valor da sobrecarga.

Num contexto logístico, Akter & Hernandez (2021) desenvolveram um método capaz de prever a que tipo de indústria pertence a mercadoria transportada por um dado veículo. O método desenvolvido é fundamental para o setor público, uma vez que as entidades privadas têm restrições de privacidade referentes aos dados partilhados, logo o setor industrial a que o material transportado pertence encontra-se anónimo. Assim sendo, com o objetivo de se obter um método capaz de prever o setor industrial de entre os seis definidos pelos investigadores, foram usados essencialmente dados de GPS para treinar e testar o modelo *Random Forest,* que apresentou uma *Accuracy* de 88%.

Park et al. (2021) realizou um estudo cujo objetivo passou pelo desenvolvimento de um sistema de diagnóstico do estado das operações referente ao transporte de minerais em minas subterrâneas. Desse modo, foram utilizados dados constituídos por informação detalhada sobre os tempos de transporte realizados por diversos veículos ao longo de várias operações, bem como os respetivos estados associados a este processo. Os autores exploraram vários modelos de *Machine Learning*, nomeadamente o GNB, kNN, SVM e CART, com o intuito de diagnosticar situações anómalas durante a realização do processo de transporte. O modelo CART foi o que demonstrou melhores resultados, com uma *Accuracy* de 94.6%*,* Precisão de 93.5%, *Recall* de 95.7% e *F1-Score* de 94.5%.

Através da aplicação de técnicas de Aprendizagem Não Supervisionada, Tahaei et al. (2021) realizaram um caso de estudo com o objetivo de estabelecer um número apropriado de *Truck Traffic Classification* (TTC), os quais são normalmente utilizados como referência para o desenvolvimento e *design* específico de pavimentos rodoviários. Portanto, fatores como o tipo de veículos que irão circular numa estrada especifica serão no momento de realizar o *design* de um novo pavimento. O número de TTC definidos para o local em análise correspondia a 17, contudo os esforços dos autores permitiram que, através da exploração do algoritmo *K-Means* e utilizando os dados rodoviários retirados de sistemas WIM*,* reduzido este número para somente 4 grupos distintos de TTC. Este valor deriva do elevado nível de correlação presente entres os vários TTC definidos até então.

Yao et. al, (2021) apresentaram um estudo que aborda a distribuição da carga em veículos de transporte de sedimentos. A forma como a carga é distribuída pelo veículo é fundamental para garantir uma vasta vida útil do mesmo. Os autores efetuaram os testes num ambiente controlado (num

laboratório), com pequenas quantidades de material e posteriormente aplicaram técnicas de *Deep Learning* para prever a forma como a carga foi distribuída nos veículos. O estudo foi concretizado através de dois passos: no primeiro foram utilizados modelos de *Deep Learning* para a classificação das imagens armazenadas, em uma de seis classes, sendo o peso do veículo o seu principal fator diferenciador. Após a classificação das imagens, na segunda etapa, foram aplicados modelos de *Deep Learning* baseados em regressão, com o objetivo de prever a distribuição do peso pelo veículo. Os modelos aplicados apontaram para um RMSE de 0.19 correspondente a 3.17% do peso total.

Ainda em 2021, Zhou et al., (2021) propuseram uma metodologia baseada em técnicas de *Deep Learning* capaz de identificar o peso de um veículo de transporte de mercadorias através da utilização de dados de acelerómetros extraídos das pontes. Segundo os autores, a deslocação de um veículo provoca uma dada resposta por parte da ponte, sendo possível extrair estes padrões visuais através da sua conversão para um espectrograma bidimensional em formato de imagem. Desse modo, foi treinado um modelo *DCNN* capaz de distinguir a informação do peso do veículo relativamente às respostas estruturais fornecidas pela ponte. Esta facto permitiu então identificar o peso do veículo que se encontrava em movimento na ponte sem a necessidade de se aplicar sistemas *WIM* em pontes, uma vez que, os custos associados a estes sistemas são elevadíssimos. O modelo proposto apresentou uma capacidade preditiva com uma *Accuracy* de 93.63%.

Recentemente, Deniz et al. (2022) realizaram uma investigação com o objetivo de obter um sistema capaz de estimar a quantidade de material que se encontrava a ser transportado por um veículo descoberto, utilizando técnicas de *Deep Learning*. No estudo em questão, foi utilizado o modelo pré-treinado VGG16, juntamente com um conjunto de 4884 imagens, das quais 3663 foram usadas para treino, 1221 para teste e as restantes para validação. Os trabalhos efetuados contaram com a realização de 17 cenários distintos, variando em cada um deles as percentagens de cargas transportadas pelo veículo. A avaliação efetuada apontou para 89.72% de *Accuracy* no pior dos cenários elaborados.

Tabela 1 - Resumo dos Trabalhos Relacionados

Autor	Objetivo	Modelo	Dados	Métrica
Kim et al., (2009)	Prever o peso total e o peso por eixo de um veículo numa ponte constituída por um sistema *Weigh-In-Motion* (WIM)	ANN	Dados retirados de uma ponte constituída por um sistema WIM.	*Accuracy*
Gungor et al., (2018)	Prever o estado do tabuleiro de uma ponte.	SVM	Dados retirados da base de dados *National Bridge Inventory* (NBI) e dados retirados de um sistema *Weight in Motion* (WIM).	*Accuracy, Recall, Precision*
Liu et al., (2019)	Classificar veículos de transporte como carregados ou vazios.	VGG16, VGG16-BF, VGG16-FT, *InceptionV3, InceptionV3-BF, InceptionV3-FT, Xception, Xception-BF, Xception-FT, Resnet50, Resnet50-BF, Resnet50-FT*	Imagens capturadas de veículos com a carga completa e sem carga.	*Accuracy*
Yan et al., (2019)	Prever a probabilidade de possíveis falhas devido ao desgaste provocado por veículos com sobrecarga que circulam em pontes.	ANN	Dados sobre o estado de desgaste da ponte e do tráfego.	*Mean Squared Erro (MSE)*
Akter & Hernandez, (2021)	Prever a que setor industrial pertence a mercadoria transportada por um dado veículo através da sua atividade diária	RF	Dados de GPS, dados rodoviários e dados do estabelecimento comercial.	*Precision, Recall F1-Score, ROC Curve*

Autor	Objetivo	Modelo	Dados	Métrica
Park et al., (2021)	Diagnosticar problemas relacionados com o processo de transporte de minerais em minas subterrâneas.	GNB, kNN, SVM, CART	Tempos de viagem dos veículos no interior das minas.	*Accuracy, F1-Score, Precision, Recall*
Tahaei et al., (2021)	Estabelecer um conjunto apropriado de *Truck Traffic Groups* com o objetivo de diminuir os *designs* de pavimentos.	*K-Means*	Dados retirados de sistemas WIM alojados em pontes.	*Root Mean Squared Derivation* (RMSE)
Yao et al., (2021)	Prever o estado do carregamento de material num dado veículo e a forma como a carga foi distribuída.	VGG19, VGG16, *InceptionV3, InceptionV2, Googlenet, Alexnet, Resnet18, Resnet101, Densenet201, Resnet50*	Imagens de veículos em processo de carregamento e dados de células de carga.	*Root Mean Squared Error* (RMSE)
Zhou et al., (2021)	Identificar o peso de veículos em movimento sobre pontes.	DCNN	Padrões capturados por acelerômetros e transformados em imagens correspondentes a espectrogramas bidimensionais por meio de análises de tempo-frequência.	*Accuracy, Precision, Recall*
Deniz et al., (2022)	Estimar o volume de material que se encontra a ser transportado por veículos descoberto.	VGG16	Imagens capturadas de um veículo de pequena escala com atributos semelhantes ao real.	*Accuracy*

2.9 Ferramentas Tecnológicas

Esta secção consiste em demonstrar todas as ferramentas que serão aplicadas e utilizadas no âmbito dos trabalhos realizados. Esta dissertação, uma vez inserida num projeto de investigação e desenvolvimento em contexto empresarial, segue a *stack* tecnológica definida e adotada no mesmo. A Tabela 2 descreve a *stack* definida no projeto. A linguagem de programação *Python* é a adotada para desenvolver os trabalhos pretendidos.

A linguagem de programação Python foi desenvolvida por Guido Van Rossum e tem o seu lançamento oficial em 1991. Com o passar do tempo esta linguagem começou a ganhar popularidade e destaque devido à sua versatilidade, começando a ser usada para o desenvolvimento de aplicações *Web*, educacionais e "numeric and scientific" (Brittain et al., 2018). São várias as características que fazem do Python a linguagem de programação escolhida para o desenvolvimento dos trabalhos necessários em termos do projeto NeWeSt.

As principais qualidades são a existência de mais de cem mil packages (Barlas et al., 2015), que podem ser utilizados para inúmeras tarefas como, por exemplo, a manipulação e modelação de dados (Vasconcelos, 2018), bem como a vasta lista de Integrated Development Environments (IDE) existentes para Data Science (exemplo: Jupyter Notebook, Spyder) (Brittain et al., 2018; Pérez et al., 2011).

Tabela 2 - Ferramentas Tecnológicas.

Função	Nome	Descrição	Versão
Distribuição	*Anaconda* [2]	É uma plataforma utilizada para o desenvolvimento de conteúdo relacionado com *Data Science.* Esta distribuição permite a implementação de computação a nível científico através do uso de linguagens de programação tais como *Python* e *R.* As principais características que levam ao uso desta plataforma de distribuição são os mais de 7500 packages *open-source* compatíveis com *Windows* e *Linux* e os *IDEs* que a constituem, os quais já se encontram preparados para desenvolvimento e implementação de projeto de *Data Science.*	4.10.1

2 https://www.anaconda.com/

Função	Nome	Descrição	Versão
Processamento e Modelação dos Dados	*Pandas* [3]	Biblioteca *open-source* desenvolvida para *Python*, cuja principal função é tornar rápida, fácil, intuitiva e flexível a manipulação e análise de dados tabulares.	1.1.5
	Pandas Profiling [4]	Biblioteca *open-source* que permite o desenvolvimento de análises exploratórias dados, além de fornecer um relatório interativo com as observações elaboradas.	1.4.2
	Scikit-Learn [5]	Biblioteca *open-source* desenvolvida para a linguagem *Python,* que fornece um conjunto de ferramentas capazes de aplicar de forma eficiente técnicas e algoritmos de *Machine Learning.*	1.1.0
	Hyperopt [6]	Ferramenta utilizada com o propósito de otimizar os parâmetros inerentes aos modelos de *Machine Learning*, através de uma procura dentro de valores selecionados e que melhor darão resposta à função de maximização ou minimização escolhida.	0.2.7
	SHAP [7]	SHAPley Additive exPlanations é uma ferramenta que permite compreender e explicar detalhadamente um modelo de *Machine Learning* através da visualização dos seus *outputs.* Assim sendo, com a utilização desta ferramenta é exequível averiguar a contribuição de cada um dos atributos para a formulação da previsão.	0.41.0
	NumPy [8]	Biblioteca *open-source* desenvolvida para a linguagem *Python* capaz de executar com elevado desempenho operações matemáticas em grandes volumes de dados pertencentes a *arrays* e matrizes multidimensionais.	1.22.3
	Xgboost [9]	Biblioteca *open-source* capaz de dar resposta a problemas de *Data Science* com uma elevada rapidez e eficácia, através da implementação de modelos de *Machine Learning* baseados na *framework Gradient Boosting.*	1.6.1

3 https://pandas.pydata.org/
4 https://pandas-profiling.github.io/pandas-profiling/
5 https://scikit-learn.org/ l
6 http://hyperopt.github.io/hyperopt/
7 https://shap.readthedocs.io/
8 https://numpy.org/
9 https://xgboost.readthedocs.io/

Função	Nome	Descrição	Versão
Processamento e Modelação dos Dados	*Imblearn* [10]	Biblioteca *open-source* que fornece ferramentas capazes de lidar com o desequilíbrio de classes que pode ocorrer em projetos cuja classificação é o principal objetivo.	0.9.1
Visualização dos Dados	*Matplotlib* [11]	Biblioteca *open-source* desenvolvida para a linguagem *Python* que permite compreender os dados através da criação de visualizações e gráficos de dados.	3.5.2
	Plotly [12]	Biblioteca *open-source* desenvolvida para as linguagens *Python*, *R* e *Julia code* que permite ao utilizador facilmente desenvolver complexas e interativas visualizações de dados.	5.7.0
Ferramenta de Visualização	*PowerBI* [13]	Serviço *Microsoft* que permite o desenvolvimento de relatórios e de *dashboards* interativos através do uso de dados, os quais podem estar armazenados em diferentes fontes e formatos.	2.100
IDE	*VSCode* [14]	O *Visual Studio Code* é um editor de código-fonte desenvolvido pela *Microsoft* e compatível com os sistemas operativos *Windows, Linux e macOS.*	1.69.0
Documentação	*Microsoft Word* [15]	É um dos vários serviços do *Microsoft Office* e tem como finalidade permitir o processamento de texto.	16.0

10 https://imbalanced-learn.org/
11 https://matplotlib.org/
12 https://plotly.com/
13 https://powerbi.microsoft.com/
14 https://code.visualstudio.com/
15 https://www.microsoft.com/pt-pt/microsoft-365/word

3. Caso de Estudo: Previsão de Desvios em Processos de Carregamento de Sacos de Cimento

Este capítulo descreve o desenvolvimento da *framework* utilizada nesta dissertação, bem como todos os passos realizados no âmbito do problema que deu origem a esta dissertação. Assim sendo, neste capítulo são abordadas as várias etapas referentes ao CRISP-DM, a metodologia selecionada para dar resposta às necessidades do projeto.

3.1 Contextualização

O caso de estudo é parte integrante do projeto NeWeSt – *New Generation of Cyber physical Weighting Systems*, cujo consórcio é formado por: Cachapuz - Weighing & Logistics Systems, Lda, INL - International Iberian Nanotechnology Laboratory, DTx - Digital Transformation CoLab e a Universidade do Minho. Este projeto tem como grande objetivo promover uma mudança holística do paradigma que rodeia os ecossistemas associados aos sistemas de pesagem industrial, quer em termos de tecnologias, conceitos e modelos de negócio utilizados.

A principal parte interessada do caso de estudo desenvolvido é a organização Cachapuz, uma empresa especialista em processos, equipamentos e soluções tecnológicas inerentes às pesagens industriais. Além da organização Cachapuz, o desenvolvimento deste caso de estudo teve o apoio da instituição *Digital Transformation Colab (DTx)*, que partilha dos mesmos objetivos. O caso de estudo presente na dissertação pertence à componente *Dispatch Workflow* do projeto NeWeSt. Esta componente tem como objetivo apresentar a construção de diversos serviços, que através de interfaces de computação são capazes de se comunicarem com as plataformas da Cachapuz, facilitando assim o desenvolvimento e implementação de futuras soluções, permitindo uma rápida e eficiente maturação até ao ponto de comercialização.

O serviço que se pretende obter com o caso de estudo, tem por base responder a um problema que ocorre durante o processo de carregamento sacos de cimento em empresas clientes da Cachapuz. De forma sucinta, neste processo ocorre uma variação entre o peso expectável (associado à quantidade encomendada pelo cliente) relativamente ao peso realmente transportado. Assim, de forma a mitigar esta variação de pesos conhecido por desvios (anomalia), desenvolveu-se uma *framework* com base em

técnicas de *Machine Learning* capaz de prever se o próximo processo de carregamento irá ou não apresentar um desvio anormal (ou seja, um desvio superior a 2% ou inferior a -2%).

3.2 Compreensão do Negócio

A etapa de Compreensão do Negócio presente no documento inclui as tarefas de conhecimento do processo *(Background),* Objetivo de Negócio e Objetivos de *Data Mining*, as quais permitem perceber todo o ambiente em volta do caso de estudo abordado, bem como compreender determinadas circunstâncias implicadas no processo de tomadas de decisão.

3.2.1 Conhecimento do Processo

No que se refere ao processo de carregamento de sacos de cimento, primitivamente o veículo responsável pelo transporte do material realiza uma pesagem inicial com o objetivo de obter o peso da tara (peso do veículo apenas). Em paralelo o SLV (plataforma que dá suporte a este processo) determina o número de sacos de cimento a serem carregados para o veículo, bem como o tipo de operação a ser realizado.

Após a pesagem da tara do veículo, este desloca-se para o local onde é efetuado o carregamento dos sacos de cimento pelas respetivas estações de carregamento, conhecido igualmente por *Packing Plant*. Posteriormente, após o cimento sofrer todas as transformações necessárias, é enviado para o *Packing Plant* de forma a ser empacotado em sacos de 50Kg e ser direcionado às várias estações de carregamento. Normalmente existem 3 tipos distintos de estações de carregamento: a estação de carregamento de sacos em paletes, a estação de carregamento automático de sacos, e a estação de carregamento manual de sacos.

Concluído o carregamento dos sacos de cimento para o veículo, este dirige-se para o local de saída da fábrica, onde mais uma vez é efetuado o processo de pesagem com o objetivo de verificar se a quantidade encomendada pelo cliente corresponde à quantidade a ser transportado pelo veículo, ou seja, determinar o desvio entre estes dois pesos. Neste momento, duas situações distintos podem ocorrer:

- O desvio apresentado encontra-se dentro do intervalo de valores aceite e, se tal ocorrer, o veículo tem permissão para sair da fábrica e o processo é dado como concluído;

- O desvio apresentado encontra-se fora do intervalo de valores aceites. Caso este evento ocorra, o veículo fica bloqueado nas instalações (durante um período indeterminado) até que sejam tomadas medidas para resolver o problema.

Os desvios que ocorrem durante o processo descrito apresentam diversas causas, algumas conhecidas e outras ainda não compreendidas pelas organizações (clientes da Cachapuz). As causas mais comuns estão associadas aos fatores ambientais, por exemplo, as variações no nível de humidade alteram as propriedades do cimento e consequentemente os pesos dos sacos tornam-se superiores ou inferiores ao estabelecido (peso de 50Kg). Para além dos fatores ambientais, os erros humanos também constituem uma das causas mais comuns para os desvios, visto que podem ser realizados carregamentos com um número incorreto de sacos ou até mesmo pesagens mal efetuadas.

3.2.2 Objetivo de Negócio

Os objetivos de negócio associados ao caso de estudo passam pelo desenvolvimento e implementação de uma *framework* que se baseia em técnicas de *Machine Learning* ou aprendizagem automática capaz de utilizar os dados recolhidos *a priori* da realização do processo de carregamento de sacos de cimento no veículo de transporte de mercadorias, para prever a ocorrência ou não do bloqueio do veículo na fábrica após a realização do carregamento dos sacos com base no histórico associado a este processo.

Como referido anteriormente, esta previsão constitui uma ferramenta muito importante para as organizações clientes da Cachapuz, visto que permite a tomada de decisão pró-ativa e consequentemente melhor fluxo de saída de veículos da fábrica, diminuição de recursos humanos para a resolução deste problema, bem como deixa de existir a necessidade de saírem veículos com excesso de peso, devido à falta de tempo para o corrigir.

3.2.3 Objetivos de *Data Mining*

O caso de estudo abordado tem como objetivo principal a previsão da ocorrência de bloqueios de veículos de transporte após o processo de carregamento de sacos de cimento resultantes de um desvio relativamente ao peso expectável (associado à quantidade encomenda pelo cliente) e o peso realmente transportando (desvio superior a 2% ou inferior a -2%). Para dar resposta ao objetivo principal do caso de estudo, foi necessário explorar vários modelos de *Machine Learning* capazes de responderem a tarefas

de previsão baseadas na classificação binária (Aprendizagem Supervisionada). Além disso, foram adotadas um conjunto de métricas associadas às tarefas de classificação como, por exemplo, o *Area Under the ROC Curve* (AUC) e *True Positve Rate* (TPR), para auxiliar na seleção do modelo a o *deployment* no serviço implementado *(cloud-based).*

3.3 Compreensão dos Dados

Na etapa de Compreensão dos Dados foram realizadas as tarefas de Recolha Inicial dos Dados, Descrição, Qualidade e Exploração dos Dados. A realização destas tarefas permitiu aprofundar os conhecimentos adjacentes aos dados que foram utilizados para o desenvolvimento do processo de modelação.

3.3.1 Recolha Inicial de Dados

Os dados à disposição para a concretização do caso de estudo são propriedade da Cachapuz - Weighing & Logistics Systems, Lda, e são concernentes ao processos de carregamento de sacos em fábricas de clientes onde a plataforma logística de pesagens SLV encontra-se em funcionamento. Os dados coletados estão armazenados numa base de dados *SQL*, contudo, de forma a não existir qualquer tipo de impacto com conexões à base de dados, a informação lá presente foi extraída para um ficheiro com formato *comma-separated values*.

3.3.2 Descrição de Dados e Qualidade de Dados

O *dataset* utilizado contém um total de 44 902 linhas, onde cada uma dessas linhas expressa um processo distinto que ocorreu numa fábrica cliente da Cachapuz. Também é composto por um total de 22 colunas, das quais sete correspondem a variáveis categóricas, oito a variáveis numéricas e sete são variáveis do tipo *date*. As 22 variáveis presentes no *dataset* são descritas na Tabela 3, Tabela 4, Tabela 5 e Tabela 6, de forma a compreender o que cada uma das colunas representa, bem como obter uma visão sobre as suas cardinalidades, correlações e valores omissos. No Apêndice I da dissertação encontram-se detalhados os resultados obtidos após a verificação da qualidade dos dados.

Tabela 3 - Descrição das variáveis do Dataset.

Atributo	Descrição	Tipo	Formato
TipoDoc	Tipo de documento desenvolvido	String	Ex: TP
TipoViatura	Configuração do veículo	String	Ex: Z002
CodProduto	Código do produto	Int	Ex: 5
DescProduto	Descrição do produto	String	Ex: CIMENT I 42,5 R SAC
estado	Estado do processo	String	Ex: F
Tara	Tara do veículo	Int	Ex: 18640
bruto	Peso bruto do veículo	Int	Ex: 53500
Liquido	Peso do material transportado	Int	Ex: 34860
QtdPedida	Quantidade solicitada em formato de peso	Int	Ex: 35000
percDiff	Diferença percentual entre a quantidade líquida e solicitada	Double	Ex: 0,266666667
PostoOperacao	Posto de operação onde ocorreu o processo de carregamento	String	Ex: SPEED1
DataCriacao	Data de criação do processo na base de dados	Date	(Ex: 2020-02-18 07:53:53.513)
Dataentrada	Data de entrada do veículo nas instalações	Date	(Ex: 2020-02-18 07:53:53.513)
TaraData	Data de pesagem da tara	Date	(Ex: 2020-02-18 07:53:53.513)
DataInicioOperacao	Data de início da operação de carregamento	Date	(Ex: 2020-02-18 07:53:53.513)
DataFimOperacao	Data de término da operação de carregamento	Date	(Ex: 2020-02-18 07:53:53.513)
BrutoData	Data da pesagem do peso bruto	Date	(Ex: 2020-02-18 07:53:53.513)
DataFecho	Data de conclusão do processo	Date	(Ex: 2020-02-18 07:53:53.513)
Matricula	Matrícula do veículo que realiza o processo	String	Ex: 7825TU127
CodEntidade	Código da entidade a quem o veículo pertence	Int	Ex: 400187
CodMotorista	Código do motorista que conduz o veículo	Int	Ex: 9900006167
NomeMotorista	Nome do motorista	String	Ex: Nome

Tabela 4 - Sumário da Correlação dos dados.

Atributos	Correlação
TipoDoc	Tara, bruto, qtdpedida, percDiff
TipoViatura	DescProduto, PostoOperacao
CodProduto	DescProduto, bruto, Liquido, qtdpedida
DescProduto	TipoViatura, CodProduto, bruto, Liquido, qtdpedida
estado	Process state
Tara	TipoDoc, bruto, Liquido, qtdpedida, percDiff

Tabela 5 - Sumário da Cardinalidade dos dados.

Atributos	Cardinalidade
Matrícula	Elevada Cardinalidade 1822 valores distintos
NomeMotorista	Elevada Cardinalidade: 2695 valores distintos

Tabela 6 - Sumário de Valores Omissos e Nulos.

Atributos	Valores Omissos e Nulos
PostoOperacao	16288 (36.0%) valores omissos
TaraData	26694 (59.0%) valores omissos
DataInicioOperacao	26694 (59.0%) valores omissos
DataFimOperacao	31522 (69.7%) valores omissos
CodMotorista	35326 (78.1%) valores omissos
percDiff	760 (1.7%) nulos

3.3.3 Exploração de Dados

Na tarefa de Exploração de dados foi realizada uma análise exploratória das diversas variáveis que constituem o *dataset* original, bem como das ligações e possíveis padrões que estas podem apresentar entre si. A análise realizada permitiu retirar diversas informações que foram fundamentais para as futuras etapas da metodologia CRISP-DM, uma vez que indicaram a necessidade ou não da aplicação de determinadas técnicas para a resolução de problemas e boas práticas para a implementação de modelos de *Machine Learning.* Nesta subsecção serão abordadas as análises consideradas mais relevantes e que demonstraram um maior impacto nos trabalhos desenvolvidos, as restantes encontram-se detalhadas no Apêndice II.

A análise corresponde à distribuição das classes da variável *target.* A Figura 15 ilustra uma discrepância consideravelmente elevada relativamente à distribuição da variável target, ou seja, estamos perante um *dataset* não balanceado. Este facto leva a que, em fases posteriores do desenvolvimento do caso de estudo, certas técnicas tivessem sido aplicadas de forma a balancear a distribuição das classes.

Para além da análise realizada às classes que constituem o *dataset* original foi fundamental efetuar a interpretação sobre o comportamento do atributo "percDiff". A Figura 16 ilustra duas análises concretizadas, sendo possível observar o comportamento do atributo "percDiff" pelos vários postos, tanto em termos de contabilização de casos anómalos, bem como a média de *percDiff* que cada posto apresentou. Assim sendo, foi exequível observar que a maioria dos casos anómalos que ocorrem têm a

sua origem na *"Unknown Station"*, este posto corresponde a uma ausência de informação alusiva ao posto de operação onde o processo ocorre. Contudo, o posto *ENV2* apresenta em média maior desvio.

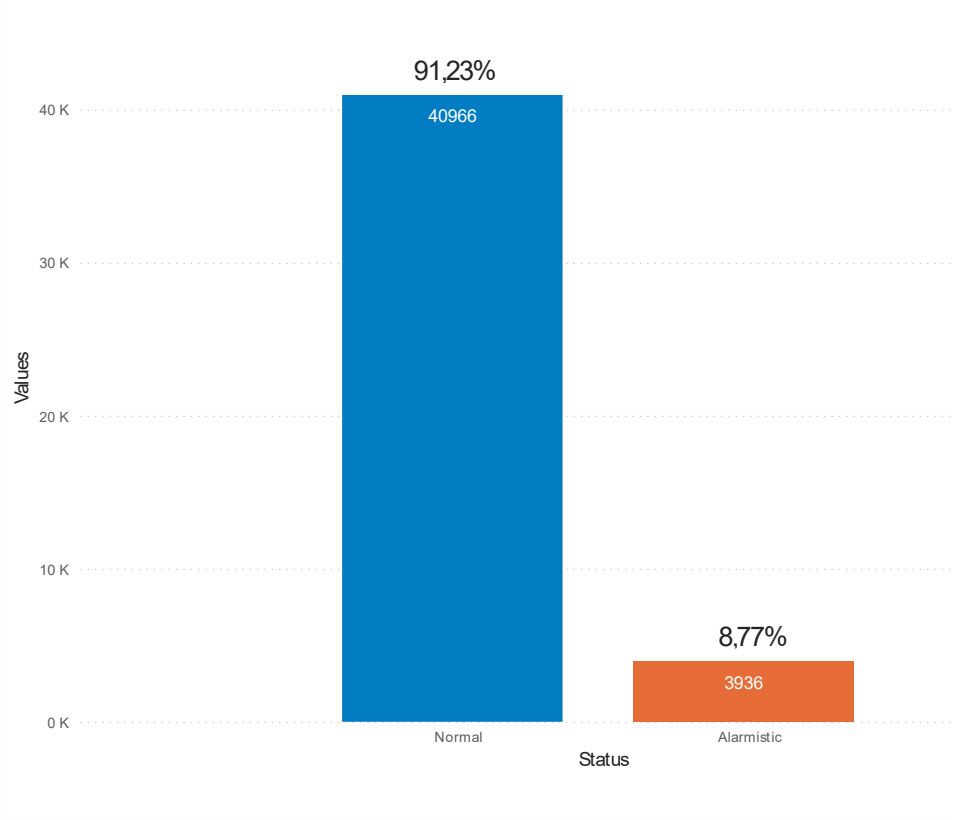

Figura 15 - Distribuição das classes.

Na subsecção de Exploração dos Dados, a realização da análise ilustrada na Figura 17, permitiu averiguar a existência de valores incorretos referentes aos campos "Liquido", "Tara" e "bruto". Após a análise dos gráficos é possível concluir que existem valores equivalentes a zero nestes atributos. Deste modo, em fases mais avançadas da metodologia estes valores tiveram de ser eliminados, visto não fazerem sentido para o processo representado no caso de estudo.

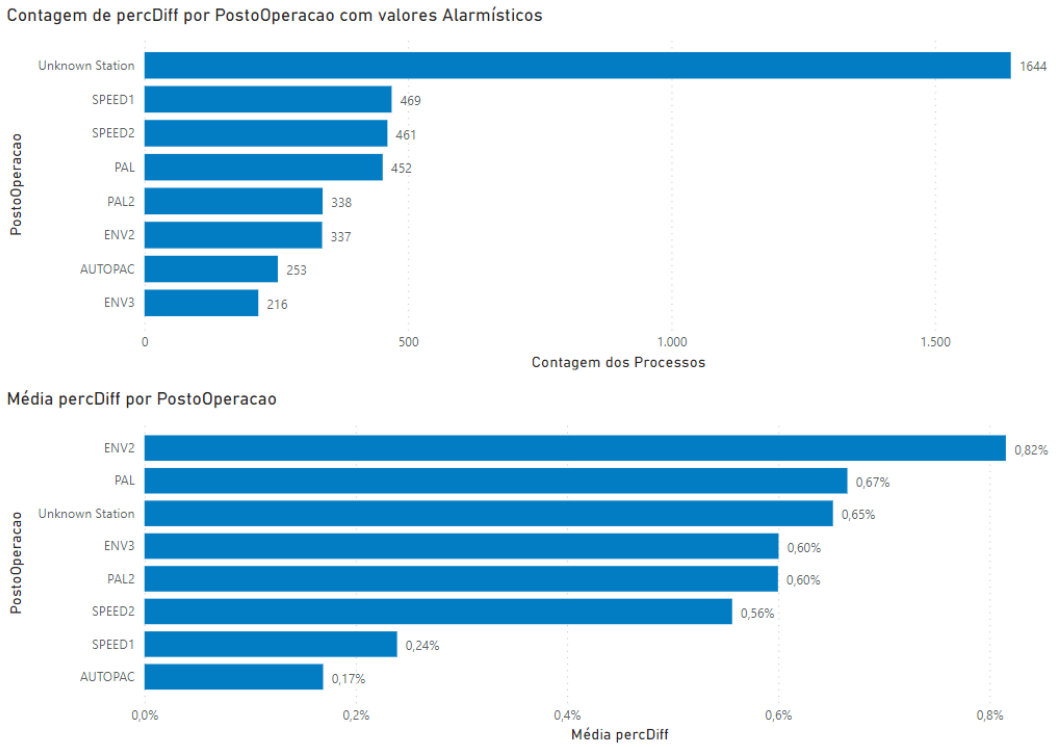

Figura 16 - Contagem e média do atributo percDiff pelos vários Postos de Operação.

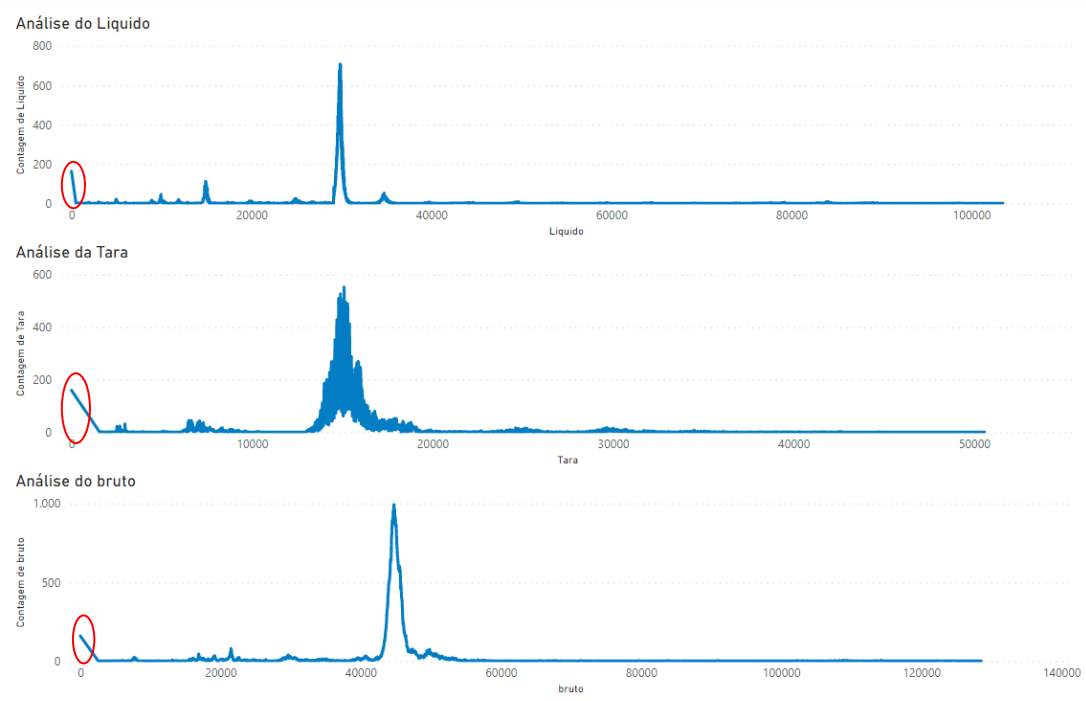

Figura 17 - Análise dos atributos Liquido, Tara e bruto.

Por último, é fundamental referir que foram efetuadas análises das componentes temporais, uma vez que as explorações das mesmas apontaram para a presença de variações significativas do "percDiff". A Figura 18 ilustra como o "percDiff" varia em média ao longo das 24 horas que compõem um dia e ao longo dos 12 meses que constituem um ano, além de demonstrar a média das contagens de "percDiff" anormais ao longo das horas e dos meses. Estes gráficos permitem concluir que o percDiff tem tendências de subida e de descida bastante peculiares, e através de questionamentos realizados a especialistas da área chegou-se à conclusão de que estas variação estão diretamente associadas a variações meteorológicas que ocorrem ao longo das horas do dia e ao longo dos meses do ano.

As alterações meteorológicas têm um forte impacto no desvio do peso, visto que as variações da temperatura e da humidade fazem com que as propriedades do cimento se alterem, levando a que a densidade do mesmo também se altere, e consequentemente o peso seja diferente. Assim sendo, um saco de cimento com determinadas condições de temperatura e humidade dificilmente apresenta o mesmo peso que o mesmo saco em condições completamente distintas. A principal forma que as organizações têm de contrariar esta problemática é a calibração das máquinas, de modo a encher os sacos com pesos distintos consoante o estado do clima apresentado no local, contudo esta calibração dificilmente é feita de maneira adequada, o que conduz ao aparecimento de outras anomalias.

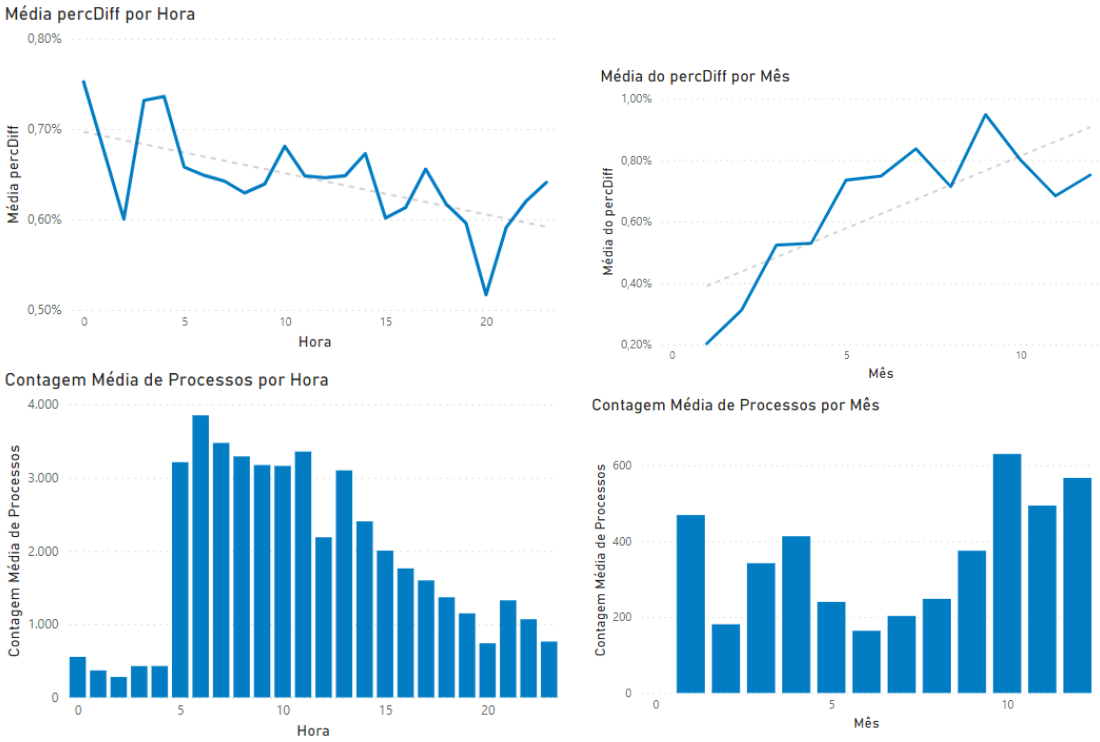

Figura 18 - Análise do percDiff ao longo das horas de um dia e ao longo dos meses.

3.4 Preparação dos Dados

A fase de Preparação dos Dados aporta as temáticas relativas à Seleção, Limpeza, Construção e Formatação dos Dados. No final desta etapa, será obtido um *dataset*, cujo propósito será ser empregue no processo de modelação.

3.4.1 Seleção dos Dados

Após uma profunda e criteriosa análise dos atributos do *dataset* original, foi efetuada uma filtragem que culminou na seleção dos originais que iriam ser aproveitados para o treino dos modelos de *Machine Learning.* Os atributos selecionados foram então acrescentados a um novo *dataset* que continha toda a informação que foi utilizada na etapa de Modelação. Dos restantes atributos, alguns foram utilizados na tarefa de Construção de Dados, com o objetivo de criar novas *features,* as quais também integraram o novo *dataset*.

Tabela 7 – Atributos selecionados a partir do dataset original.

Contexto	Atributo	Decisão
Veículo	TipoDoc	Descartar
	TipoViatura	Descartar
	Matricula	**Construção de Dados**
Produto	CodProduto	Descartar
	DescProduto	Descartar
	estado	Descartar
Pedido	QtdPedida	**Manter**
Pesagem	Tara	**Manter**
	bruto	Descartar
	Liquido	Descartar
Operação	PostoOperacao	**Construção de Dados**
Operação	DataCriacao	Descartar
	Dataentrada	Descartar

Contexto	Atributo	Decisão
Operação	TaraData	**Construção de Dados**
	DataInicioOperacao	Descartar
	DataFimOperacao	Descartar
	BrutoData	Descartar
	DataFecho	Descartar
Entidade e Motorista	CodEntidade	Descartar
	CodMotorista	Descartar
	nome Motorista	Descartar
Desvio	percDiff	**Construção de Dados**

3.4.2 Limpeza de Dados

A tarefa de Limpeza dos Dados tem como objetivo resolver problemas identificados durante o decorrer da fase de Qualidade dos Dados. Desta forma, foi necessário remover ou alterar valores omissos. Com base nos trabalhos desenvolvidos, somente o atributo "PostoOperacao" sofreu o processo de "limpeza". Neste atributo, os valores omissos que este continha foram convertidos em "Unknown Station", de forma a que fosse possível agrupar vários elementos durante o processo de construção das novas *features*. Para além disso, efetuou-se a remoção de todas as linhas cujos valores fossem iguais a zero nos atributos "Tara", "bruto" e "Liquido" (valores introduzidos na base de dados como testes pela Cachapuz). Os valores anteriormente referidos verificam-se devido a testes organizados pela Cachapuz, que levaram à inserção destes valores na base de dados.

3.4.3 Construção de Dados

Com o objetivo de enriquecer o *dataset* utilizado na etapa da Modelação, um conjunto de novos atributos foram criados com base em informações fornecidas pelo especialista do domínio, bem como através das análises exploratórias realizadas ao dados na etapa de Compreensão dos Dados. Desse modo, foram criados os seguintes atributos: "Inspection", "Block", "Average_Deviation_Station",

“Average_Station_Weekly”, “Average_Station_Hourly”, “Percentage_Blocks”, “Hour”, “Day”, “DayOfWeek” e “Month”.

O atributo "Inspection" consiste em determinar a existência de elementos administrativos no processo de carregamento de sacos de cimento. No período do dia compreendido entre as 06h-18h, o processo é inspecionado. Como tal, definimos este atributo como binário, querendo isto dizer que recebe o valor 0 para determinar que o processo de carregamento foi efetuado no período com inspeção e o valor 1 para o período sem inspeção.

A “Average_Deviation_Station” foi criada, tendo por base a média de desvio das últimas cinco pesagens (em que após reuniões de *braimstorming* com o especialista de domínio, foi sugerido o n=5) realizadas naquela estação de carregamento específica. Os atributos “Average_Station_Weekly” e “Average_Station_Hourly” têm a mesma lógica, contudo, estes expõem a média de desvio no contexto da última semana e da última hora. É necessário ter em consideração que cada um destes novos atributos está assente nas médias verificadas pelo posto de operação específico onde o processo se realizou.

Após a criação dos atributos anteriormente mencionados, foram estruturados os atributos referentes à componente temporal, assim sendo, foram concebidos os atributos “Hour”, “Day”, “DayOfWeek” e “Month”. As análises exploratórias dos dados mostram que determinadas horas do dia apresentam uma maior frequência de desvios do que outras. Além disso, o dia, dia da semana e mês são elementos a ter em conta, porque as análises efetuadas também indicam variações dos níveis de desvio consoante determinados valores que estes atributos apresentem.

De seguida, foi criado o atributo “Percentage_Blocks”, que corresponde à percentagem de bloqueios que um dado veículo teve ao longo do tempo. Este atributo baseia-se na conceção de que um veículo com uma elevada frequência de bloqueios, irá apresentar uma probabilidade elevada de ficar novamente bloqueado em processos futuros.

Por fim foi desenvolvido o atributo “Block”, correspondendo este à *label,* permitindo assim, através do seu desenvolvimento o uso de uma Aprendizagem Supervisionada. No que se refere a este atributo, foi utilizada a coluna “percDiff” do *dataset* original, e aplicado um dado conjunto de condições, com o intuito de se obterem duas classes. Deste modo, a classe 0 corresponde a processos em que o desvio apresentado é aceite pela organização, tendo como intervalo de valores $]-2;\ 2[$, enquanto a classe 1 representa desvios alarmísticos e é definida a partir dos valores dentro do intervalo $]-\infty;\ -2]\ U\ [2;\infty[$.

3.4.4 Formatação de Dados

No decorrer da tarefa de análise da qualidade dos dados verificou-se a necessidade de formatar determinados atributos. Assim sendo, foi aplicada a tarefa de formatação de dados dos atributos "QtdPedida" e "TaraData" relativos ao *dataset* original, o que permitiu alterar o formato que estes apresentavam. O atributo "QtdPedida" teve o seu formato alterado para *float,* enquanto o "TaraData" que inicialmente apresentava-se no formato texto*,* foi transformado em *datetime.*

3.5 Modelação

Com o término da fase de Processamento dos Dados é dado início à Modelação, a quarta fase da metodologia CRISP-DM. Nesta etapa da metodologia, é realizado todo um processo de seleção e aplicação de técnicas de *Machine Learning* capazes de fazerem uso do novo *dataset* elaborado na fase anterior. Assim sendo, o uso dos dados foi fundamental para a obtenção de modelos de classificação binária, capazes de prever a existência de uma anomalia futura relacionada com o desvio do peso para limites fora dos intervalos de aceitação definidos para o processo elaborado no caso de estudo.

3.5.1 Seleção de Técnicas de Modelação

Na etapa de Modelação foi utilizada a linguagem de programação Python que, através do uso das bibliotecas *scikit-learn* e *Xgboost,* possibilitou o desenvolvimento dos modelos preditivos. Tendo em conta o problema em questão, este representa uma classificação binária (Aprendizagem Supervisionada) e, assim, foram então selecionados os modelos *Decision Tree, Random Forest, Gradient-Boosted Tree, XGBoost, Support Vector Machine,* e *Multilayer Perceptron*, os quais tiveram a sua explicação e desenvolvimento teórico discorridos na Secção *2.5.1*.

3.5.2 *Design* de Testes

No que se refere aos projetos de *Machine Learning,* uma componente essencial dos mesmos é a forma como os dados à disposição são separados em partições de treino, teste e validação. Assim sendo, é necessária uma apreciação que permita definir que dados serão utilizados para treinar e consequentemente estimar os melhores parâmetros dos modelos implementados, e que dados de teste serão utilizados para avaliar a performance dos modelos implementados.

Com o objetivo de definir um plano de ação capaz de suportar a conceção das tarefas relacionadas com a Modelação, foi desenvolvida a *Pipeline de Machine Learning* ilustrada na Figura 19. Esta figura, apesar de apresentar componentes das etapas de Preparação dos Dados e Implementação, tem como principal foco a Modelação e a fase posterior de Avaliação. Nesta *pipeline* é observável a aplicação das técnicas *Rolling Window, Data Standardization* e *Data Augmentation.*

Rolling Window

De forma a obter um ambiente robusto e que se aproxime da realidade, foi aplicado o mecanismo de *Rolling Window* ilustrado na Figura 20. Este mecanismo tem por base a simulação de um ambiente real, onde ocorrem diversas iterações sequenciais ao longo do tempo, constituídas por dados de treino e de teste. Desse modo, em cada uma das iterações o algoritmo em análise é treinado e testado com a parcela de dados alocada àquela iteração, produzindo, assim, um conjunto de previsões que são posteriormente submetidas a métricas estatísticas de forma a avaliar os resultados ao longo do tempo.

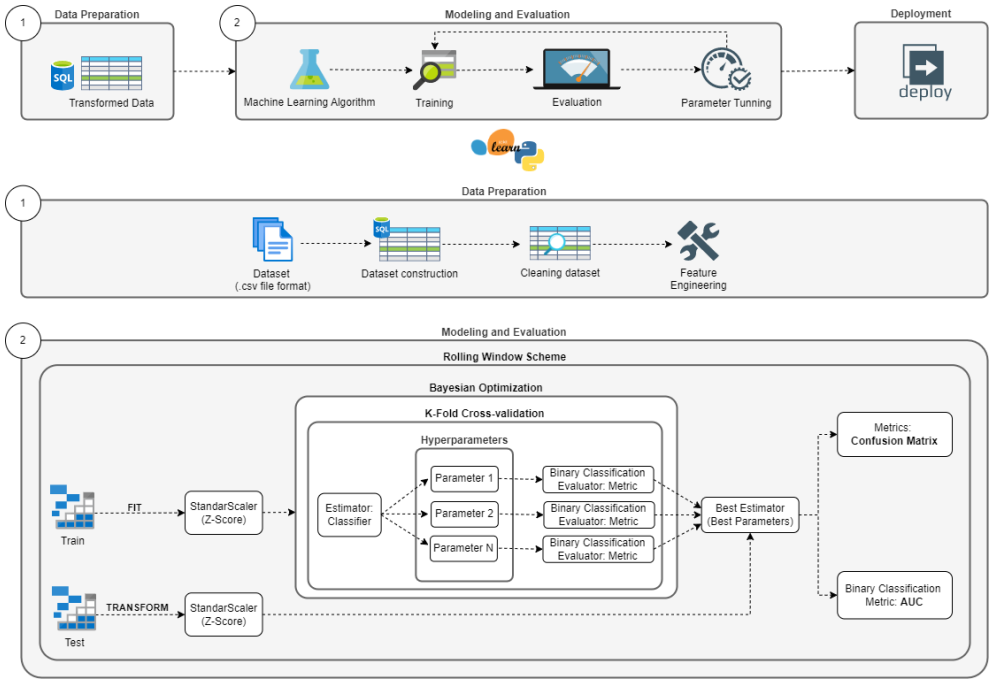

Figura 19 - Pipeline de Machine Learning.

No contexto do problema em estudo foi aplicado um mecanismo fixo de *Rolling Window,* em que cada iteração apresenta *W* intervalo de dados de treino, *T* intervalo de dados de teste e um espaço *S* de avanço. Numa primeira iteração, o mecanismo utiliza os intervalos iniciais de *W e T*, para testar e treinar

o modelo e, nas iterações posteriores, o intervalo *W e T*, avança *S* no tempo, sendo então utilizados dados mais recentes para treinar e testar os modelos.

Relativamente ao mecanismo *Rolling Window*, a fórmula seguinte permite determinar o número de iterações que este irá realizar, sendo esta fórmula constituída pelas variáveis:

$$U = (D - (W + T))/S$$

U – Número de iterações, tendo a fórmula devolvido um total de 20;

D – Tamanho do dataset utilizado (44902 linhas);

W – Tamanho da janela de treino (31500 linhas);

T – Tamanho da janela de teste (635 linhas), e

S – Tamanho da janela de avanço (635 linhas).

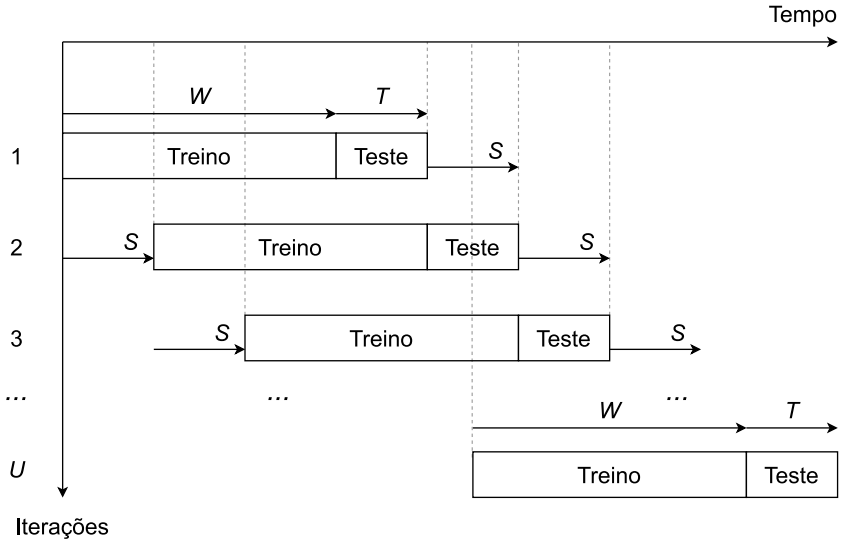

Figura 20 - Mecanismo de Rolling Window (Adaptado de Oliveira et al., 2017).

Data Augmentation

As análises exploratórias dos dados realizadas na Secção 0 permitiram concluir que os dados são constituídos por um desbalanceamento entre as classes definidas como *target* dos modelos. Este facto é problemático, porque os modelos perdem uma maior quantidade de tempo a aprender padrões relacionados com a classe de maiores dimensões, levando assim a uma menor aprendizagem e compreensão dos padrões relacionados com as restantes. Deste modo, um desbalanceamento das classes por vezes apresenta uma ilusão relacionada com as métricas de avaliação de modelos de

classificação, porque quanto maior a diferença entre as classes menor o impacto de uma previsão errada na classe com menores dimensões, o que faz com que o valor da métrica seja bastante elevado. Com o propósito de dar resposta à problemática do desbalanceamento das classes, de entre as diversas técnicas, a escolhida para o caso de uso foi a *Data Augmentation.*

O método de *Data Augmentation* cria pequenas alterações nos dados da classe que se apresenta em menor número. Deste modo, estes novos dados são adicionados ao *dataset,* o que leva a uma paridade do número de ambas as classes, sem existir repetição de dados na classe de menor valor, porque os novos dados adicionados são ligeiramente diferentes dos originais. O *SMOTE* foi o algoritmo escolhido para a criação dos novos dados sintéticos, tendo este sido aplicado através do uso da biblioteca *imblearn.*

Standardization

Muitos modelos de *Machine Learning* são sensíveis às escalas apresentadas pelos dados, estando este facto intrinsecamente ligado com a forma como estes irão operar. Desse modo, o processo de colocar todos os dados que serão utilizados para treinar os modelos numa mesma escala é importantíssimo. Assim sendo, no que se refere à colocação de todos os dados em escala, é possível recorrer-se a técnicas de *Standardization* e *Normalization.* Nos trabalhos elaborados, foi selecionada e implementada a técnica de *Standardization* denominada de *Z-Score.* Esta técnica visa criar uma transformação nos dados que torna a média dos valores de cada coluna igual a zero e o desvio-padrão igual a um.

3.5.3 Construção de Modelos

Um dos principais componentes a ter em conta no que se refere ao desenvolvimento e construção de modelos de *Machine Learning* é a otimização dos *hyperparameters* que os constituem. Assim sendo, foi adotada uma estratégia que passou pela implementação de *Bayesian Optimization*, através da biblioteca *HyperOpt* abordada na Secção 2.9, juntamente com o procedimento *K-Fold Cross Validation.*

O *K-Fold Cross Validation* ilustrado na Figura 21 é um procedimento que aplica técnicas associadas à criação de amostras nos dados, com o objetivo de se obter uma avaliação do modelo de *Machine Learning*. Este procedimento é constituído por somente um único parâmetro denominado de *K,*

o qual define o número de *Folds* que existiram durante a aplicação do procedimento. Nos trabalhos realizados foi utilizado um $K = 5$ e a métrica de avaliação AUC.

Em relação à ferramenta *HyperOpt,* primeiramente foi definido para cada um dos modelos de Machine Learning um espaço de procura referente ao *hyperparameters.* De seguida, tendo por base a função objetivo *fmin()* ilustrada na Figura 22 da ferramenta *HyperOpt,* foi definido que esta função iria ser minimizada ao longo de 10 iterações (número máximo de avaliações) do método *Tree of Parzen Estimator*. A cada iteração do *K-Fold Cross Validation* como visualizável na Figura 23, foi submetido o valor negativo da métrica AUC *(loss)* devolvida, de forma a existir uma busca no espaço de procura, pelos melhores parâmetros do algoritmo em análise.

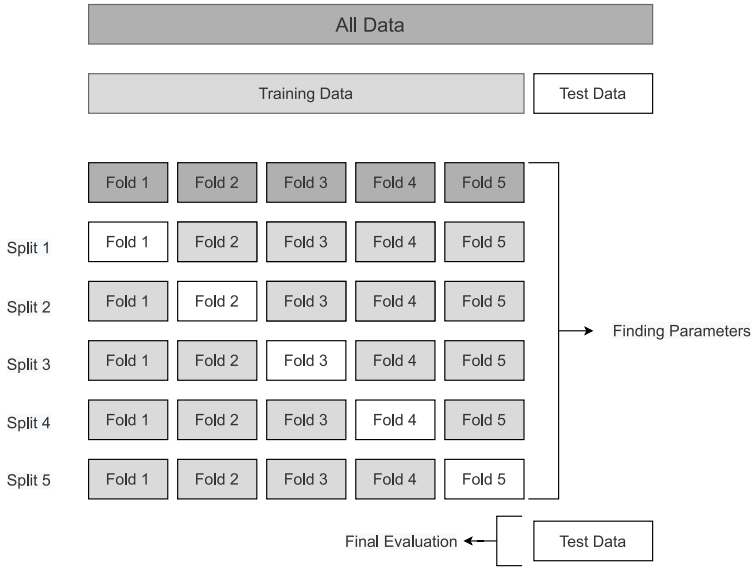

Figura 21 - Exemplificação do Procedimento K-Fold Cross Validation (Retirado de scikit-learn, 2012).

```
best_Parameters = fmin(
    fn=partial(
        mod_eval.fit_model_hyperopt,
            estimator=k,
            kfolds=kfolds,
            X_train=X_train_scaled,
            y_train=y_train
            ),  # function to optimize
        space=models[k][1],  # Defines space of hyperparameters
            # optimization algorithm, hyperotp will select its
            # parameters automatically (Search algorithm: Tree of Parzen Estimators, a Bayesian method)
        algo=tpe.suggest,
        max_evals=n_iter,  # maximum number of iterations
        trials=trials  # logging
        )

print("\nFit {}...".format(models[k][0]))
results = mod_eval.fit_model_h(params=best_Parameters,estimator=k, X_train=X_train_scaled, y_train=y_train,X_test=X_test_scaled)
predictions = results['y_pred']
model = results['model']
probability = results['probs']
time_elapsed1 = results['time_train']
time_elapsed2 = results['time_predict']
```

Figura 22 - Função fmin() da ferramenta HyperOpt.

```
cval = cross_val_score(model, X_train, y_train, scoring='roc_auc', cv=kfolds)

auc = cval.mean()
# Because fmin() tries to minimize the objective, this function must return the negative auc.
return {'loss': -auc, 'status': STATUS_OK}
```

Figura 23 - Devolução da métrica AUC negativa.

3.5.4 Configuração do Parâmetros

Como abordado anteriormente, muitos modelos possuem parâmetros únicos e para cada modelo que foi implementado verificou-se a necessidade de criar um intervalo de valores de procura para os seus parâmetros. Assim sendo, o modelo DT teve definido como espaço de procura dos *hyperparameters, max_depth* = {2, 5, 10, 20, 30} e *min_samples_split* = {2, 6, 10}. No caso do RF e do GBT foi definido *n_estimators* = 200 e os restantes parâmetros iguais aos do DT. Em relação aos intervalos de parâmetros do modelo *XGBoost* este apresentou eta = {0.0, 0.25, 0.5, 0.75, 1.0}, *max_depth* = {2, 5, 10, 20, 30} e *max_bin* = {10, 20, 40, 80, 100}. O modelo SVC teve como intervalos, *C* = {0.01, 0.1, 0.5, 1.0, 2.0}. Por último, para o modelo MLP foi definida uma rede com apenas uma *hidden layer* com *H* neurónios, cujo valor de *H* = *round(N/2),* onde 2 é o número de *inputs* do modelo. Além disso, foi definido um *learning_rate* = *{constant, invscaling, adaptive}.*

3.6 Avaliação

A fase de Avaliação é a quinta etapa da metodologia CRISP-DM, sendo nesta realizada toda a componente correspondente à avaliação dos modelos desenvolvidos, com o objetivo de apreciar o desempenho dos mesmos face ao caso de estudo e aos objetivos de negócio estabelecidos. Assim sendo, através de um conjunto de métricas é possível traduzir os resultados obtidos nas previsões realizadas pelos modelos de *Machine Learning*, de forma a compreender as mesmas e avaliar concretamente a performance associada a cada modelo. As métricas de avaliação utilizadas foram escolhidas com base no tipo de aprendizagem de *Machine Learning* empregue e, desse modo, foram aplicadas aquelas que permitem avaliar modelos baseados em Aprendizagem Supervisionada.

3.6.1 Métricas para a Avaliação dos Modelos

Como abordado anteriormente, cada vez mais a sociedade tem presente no seu quotidiano a aplicação de técnicas e projetos relacionados com *Machine Learning.* Contudo, a implementação de um

modelo ou algoritmo não é suficiente, este facto evidencia a necessidade de avaliar os modelos desenvolvidos através de métricas, com o intuito de averiguar o desempenho e até mesmo permitir a comparação entre diversos modelos, sendo este um "crucial step", no que diz respeito às diferentes tarefas de *Machine Learning* (Pathak, 2020).

A aplicação de diferentes tarefas de *Machine Learning* leva a que diferentes métricas tenham de ser usadas, ou seja, existe uma variação das métricas aplicadas com base no tipo de aprendizagem que se está a usar. Desse modo, e tendo em consideração os trabalhos desenvolvidos, foram aplicadas somente métricas relacionadas com a Aprendizagem Supervisionada, referentes à tarefa de classificação.

A principal métrica aplicada no decorrer dos trabalhos desenvolvidos foi a *Area Under Curve* (AUC), referente à análise da curva *Receiver Operating Characteristic* (ROC) (Fawcett, 2006). Num classificador, a probabilidade de decisão referente a uma determinada classe encontra-se definida pelo intervalo $p \in [0,1]$ e pelo *threshold* D, desse modo, a classe prevista será positiva se $p > D$ (Afsar et al., 2018). A aplicação de um *threshold* permite configurar o modelo de forma a que este realize previsões mais sensíveis (D, de baixo valor) ou mais especificas (D, de elevado valor) consoante as necessidades (Afsar et al., 2018). Além disso, o uso da métrica AUC, a qual tem a sua fórmula expressa na Figura 24, permite uma interpretação dos resultados sendo possível concluir que 50% é uma classificação randómica, 60% é razoável, 70% é boa, 80% é muito boa, 90% é excelente e, 100% é uma classificação perfeita (Matos et al., 2021).

$$AUC = \int_0^1 \frac{TP}{TP+FN} d\frac{FP}{FP+TN} \; d = \int_0^1 \frac{TP}{P} d\left(\frac{FP}{N}\right)$$

Figura 24 - Fórmula da métrica AUC

A AUC não foi a única métrica aplicada nos trabalhos desenvolvidos, ou seja, depois de se obter o modelo com melhores resultados, no que se refere à métrica previamente abordada, foi descoberto o melhor *threshold* associado a ele, com o objetivo de se criar a *Confusion Matrix* e consequentemente ser possível obter as métricas *True Positive Rate* (TPR) e *False Positive Rate* (FPR) (Larose 2005; Sun et al. 2009). A *Confusion Matrix* é fundamental para representar de forma descritiva a *performance* de um modelo (Markham, 2020). Esta matriz é constituída por 4 termos: *True Negative* – TN*; True Positive* – TP*; False Negative* – FN*;* e, *False Positive* – FP, como representado na Figura 25 (Mishra, 2020). Os termos formados na matriz correspondem à ligação entre o caso real e a previsão realizada pelo modelo.

Como abordado anteriormente, a criação da *Confusion Matrix* permite obter as restantes métricas associadas ao processo de classificação ao nível de uma abordagem de Aprendizagem Supervisionada. A Tabela 8 descreve as restantes métricas desenvolvidas de forma a avaliar os resultados obtidos nos vários modelos de *Machine Learning* aplicados.

A curva ROC corresponde a um gráfico bidimensional que representa técnicas para a visualização, organização e seleção de classificadores de *Machine Learning,* baseados na sua *performance*. De uma maneira geral, a curva *ROC* ilustra o *trade-off* entre o TPR (eixo y) e o FPR (eixo x) para diversos pontos de threshold (D), os quais contêm valores entre 0.0 e 1.0 (Dwivedi 2018; Fawcett 2006; Li & Zhang 2020; Sun et al. 2009). Além disso, a curva ROC é normalmente aplicada de forma a medir na globalidade a *Accuracy* de um modelo (Li & Zhang 2020).

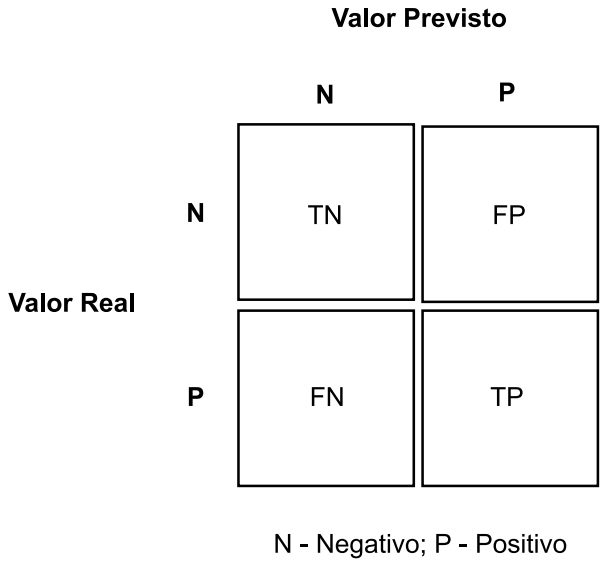

Figura 25 - Confusion Matrix (Baseado em Markham, 2020).

Tabela 8 - Métricas usadas em tarefas de Classificação (Baseado em Pathak, 2020).

Métrica	Fórmula
True Positive Rate (TPR)	$\frac{TP}{TP + FN}$
False Positive Rate (FPR)	$\frac{FP}{FP + TN}$

Por fim, foi aplicado o teste não paramétrico *Wilcoxon Test*, com o objetivo de verificar se existia algum tipo de significância estatística entre os vários modelos implementados (Hollander et al., 2013).

3.6.2 Avaliação do Modelos

Após a compreensão das métricas utilizadas para a Avaliação abordadas na subsecção 3.6.1 procedeu-se à realização efetiva da avaliação dos modelos de *Machine Learning* implementados. A Figura 26 ilustra as 20 iterações realizadas através dos mecanismos de *Rolling Window*. Assim sendo, a partir da análise da figura é possível interpretar o comportamento da métrica AUC ao longo das várias iterações para cada modelo implementado. Contudo, de forma a facilitar a avaliação dos modelos foi criada a Tabela 9, com o intuito de demonstrar os resultados em termos da mediana obtidos pela métrica AUC ao longo das iterações realizadas, bem como apresentar a média dos tempos despendidos para treino e teste de cada um dos modelos.

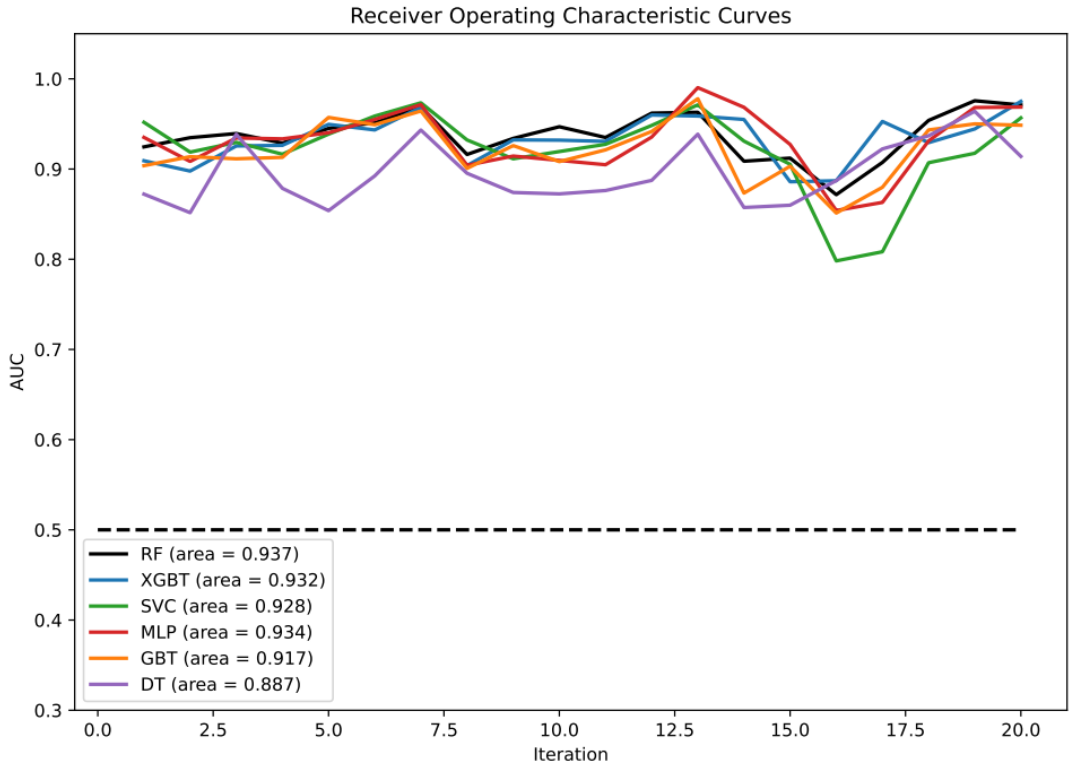

Figura 26 - Evolução do valor AUC ao longo das iterações do Rolling Window.

Tabela 9 - Comparação entre os vários modelos de Machine Learning (melhores valores a negrito).

Modelo	*AUC*	Tempo de Treino (s)	Tempo de Previsão (s)
Random Forest (RF)	**0.937***	14.71	0.031
Gradient-Boosted Tree (GBT)	0.917	177.71	0.015
eXtreme Gradient-Boosting (XGBoost)	0.932	10.89	0.001
Support Vector Machine (SVM)	0.928	318.48	0.374
Multilayer Perceptron (MLP)	0.933	15.18	**0.001**
Decision Tree (DT)	0.887	**0.18**	**0.001**

* *RF é estatisticamente significativo com GBT, SVM e DT.*

A Tabela 9 permite concluir que o modelo *Random Forest* é aquele que apresenta uma mediana da métrica AUC mais eleavada, sendo este valor de 0.937. A Tabela 8 também permite visualizar que o modelo *Decision Tree* é o que apresenta menor tempo de treino, demorando apenas 0.18s em média. No que se refere ao tempo de predição (s), os modelos com melhores resultados foram o *Multilayer Perceptron* e o *Decision Tree* tendo estes apresentado uma duração em média de 0.001s. Para além das análises anteriormente elaboradas, foi aplicado o teste estatistico abordado na subsecção 3.6.1, denominado *Wilcoxon Test*. O teste desenvolvido permitiu concluir que o modelo *Random Forest* é estatisticamente significativo com o modelo *Gradient-Boosted Tree, Support Vector Machine* e *Decision Tree.*

Com base nos resultados ilustrados anteriormente, o modelo selecionado para realizar as tarefas de previsão necessárias ao caso de estudo desenvolvido foi o modelo *Random Forest*. Assim sendo, e para demonstrar os resultados reais obtidos pelo modelo, foi necessário primeiramente encontrar o melhor *threshold* (D) para o modelo em questão. Desse modo, desenvolveu-se a aplicação de um mecanismo na penúltima iteração do *Rolling Window*, ou seja, na iteração número 19, de forma a obter o valor em concreto do *threshold.* A Figura 27 demonstra a curva ROC associada à iteração número 19, bem como o valor ideal encontrado para o *threshold*, que foi de 0.285. Após a obtenção do melhor *threshold* e de forma a validar os resultados obtidos, foi aplicada a *Confusion Matrix,* ilustrada na Figura 28, na iteração número 20 do mecanismo de *Rolling Window* permitindo extrair os valores necessários para a aplicação das métricas TPR e FPR, cujos resultados estão expostos na Tabela 10.

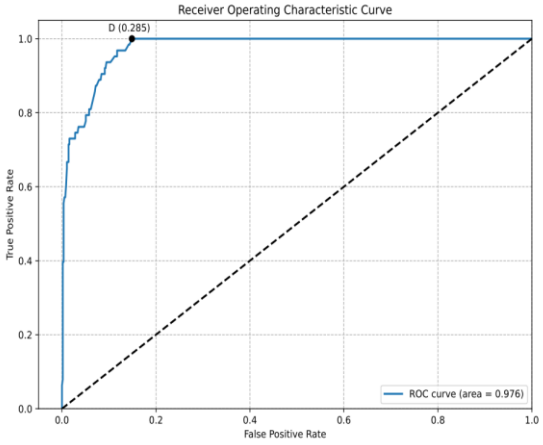

Figura 27 - Curva ROC de RF para U = 19 do RW e D = 0.285.

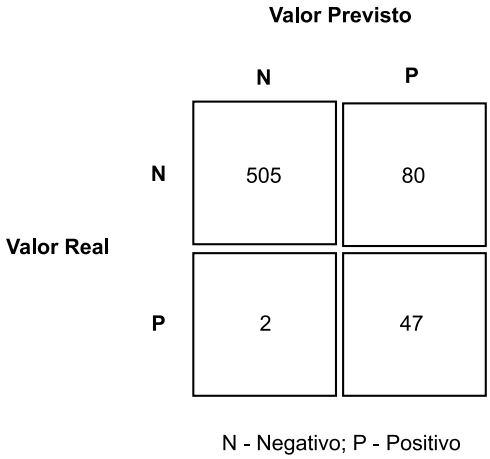

Figura 28 - CM de RF para U = 20 e D = 0.28

Tabela 10 - Resultados da Previsão de U = 20 e D = 0.285.

Threshold (D)	***TP***	***TN***	***FP***	***FN***	***TPR***	***FPR***
0.285	47	505	80	2	95.91%	4.08%

3.6.3 Explicação dos Resultados do Modelo

A aplicação de técnicas e modelos de *Machine Learning,* como anteriormente mencionados neste documento, tem ganho uma grande popularidade, contudo muitos dos modelos aplicados sofrem do problema "black box" (Ribeiro et al., 2016; Sahakyan et al., 2021). Ou seja, são fornecidos *inputs*, os quais irão resultar num *output* através da aplicação de um conjunto de decisões, tornando impossível os analistas conseguirem explicar o que leva o modelo a tomar essas determinadas decisões (Sahakyan et al., 2021). Este facto tem ainda mais importância quando os resultados obtidos por um modelo se apresentam como críticos para um dado processo. Como tal, este problema determinou um constante aumento dos estudos relacionado com *Explainable Artificial Intelligence* (XAI) (Sahakyan et al., 2021).

O conceito XAI refere-se a um vasto conjunto de técnicas que permitem que um ser humano compreenda e interprete as decisões tomadas pelo modelo relativamente à previsão elaborada por este (Sahakyan et al., 2021). A capacidade de poder compreender e interpretar os fatores inerentes à "black box" permite cujos resultados estão com os resultados obtidos pelo modelo de *Machine Learning* (Lundberg & Lee, 2017). Assim sendo, através do conhecimento aprofundado da forma como as predições são elaboradas, surgem diversos *insights* que conferem uma maior compreensão do modelo

e possibilitam o aprimoramento do mesmo, bem como consequentemente a obtenção de melhores resultados (Lundberg & Lee, 2017).

O surgimento de contextos de *Big Data,* onde subsiste uma grande quantidade e variedade de dados, leva à aplicação de modelos cada vez mais complexos, facto que torna a capacidade de os explicar e interpretar cada vez mais desafiante (Mitchell et al., 2022). Deste modo, ao longo do tempo, diversos métodos foram desenvolvidos com o objetivo de fornecer uma resposta concreta para o problema (Ribeiro et al., 2016), como por exemplo o *SHapley Additive exPlanations* [16] (SHAP) (Lundberg & Lee, 2017).

O SHAP foi o método escolhido para dar resposta à necessidade de aplicação de uma componente de XAI considerando o modelo RF selecionado. Este método tem como base os *Shapley Values* (Shapiro & Shapley, 1978), uma abordagem que está plenamente ligada à teoria dos jogos cooperativos permitindo este método explicar os outputs originados pelo modelo de Machine Learning através de visualizações (Lundberg & Lee, 2017). As Figura 29, Figura 30 e Figura 31 ilustram algumas das várias análises elaboradas, encontram-se nas restantes no Apêndice III.

A Figura 29 ilustra o impacto que os atributos apresentam para cada uma das classes *(outputs* do modelo de *Machine Learning* implementado). A figura mostra que o atributo "Average_Station_Hourly" é o que mais influencia o resultado das previsões do modelo, seguido pelo "Percentage_Blocks", "Average_Station_Weekly" e "Average_Station_Station", terminando no atributo "Day". De realçar que 8 dos 11 atributos que mais contribuem para o resultado das previsões foram criados no processo de *feature engineering* (ver Figura 29).

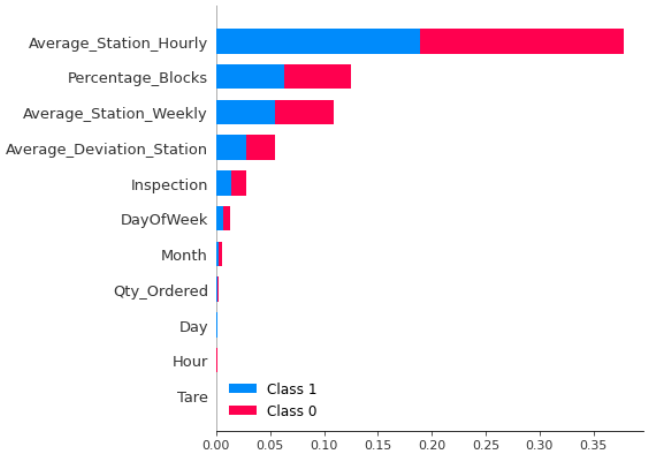

Figura 29 - Impacto dos atributos no output.

[16] https://shap.readthedocs.io/en/latest/index.html

Visando aprofundar a análise anteriormente realizada, foram aplicadas as visualizações ilustradas na Figura 30 e Figura 31, com objetivo de compreender as probabilidades face à classe 0. Estas mostram a influência dos vários atributos associados às previsões do modelo. O método *SHAP* foi aplicado a dois processos de pesagem distintos, no primeiro a previsão realizada pelo modelo *Random Forest,* devolveu um resultado pertencente à classe 0, ou seja, não existência de anomalia (Figura 30), enquanto no segundo processo, o resultado da previsão apontou para uma possível anomalia (Figura 31).

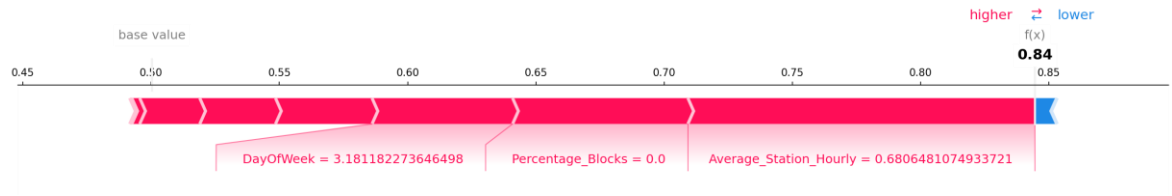

Figura 30 - Impacto dos atributos numa pesagem normal

A Figura 30 representa um *Force Plot* que permite observar que os atributos "Average_Station_Hourly" e "Percentage_Block" foram o que mais contribuíram para a previsão do output Classe 0 com uma probabilidade de acerto de 0.84.

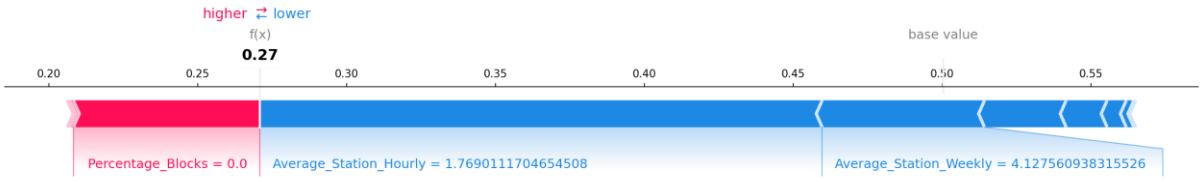

Figura 31 - Impacto dos atributos numa pesagem alarmística.

A Figura 31 permite visualizar que os atributos "Average_Station_Hourly" e "Average_Station_Weekly" foram os que mais impactaram a previsão do output Classe 1 do modelo com uma probabilidade de 0.70. Como tal, os restantes 0.30 referem-se à probabilidade para a Classe 0, cujo atributo que mais contribuiu foi o "Percentage_Block".

3.7 Implementação

A Implementação é a última fase da sequência presente na metodologia CRISP-DM, sendo nela feita a agregação de toda a informação e dos resultados obtidos ao longo da metodologia, com o objetivo de fornecer uma implementação que dê respostas ao problema que originou o projeto de *Data Mining* desenvolvido e aprofundado.

3.7.1 Plano de Implementação

De forma a dar resposta à problemática que deu origem ao caso de estudo abordado ao longo de todo o documento foi desenvolvido um micro-serviço capaz de utilizar dados provenientes da plataforma *SLV Cement* e realizar previsões de anomalias relacionadas com o desvio do peso durante o carregamento de sacos de cimento em veículos de transporte de mercadorias, a partir de um modelo ML previamente treinado A

Figura 32 ilustra a Arquitetura Tecnológica constituída por três grandes componentes, *Machine Learning, SLV Cement,* e *Visualization* que em conjunto formam o ambiente do processo realizado pelo micro-serviço.

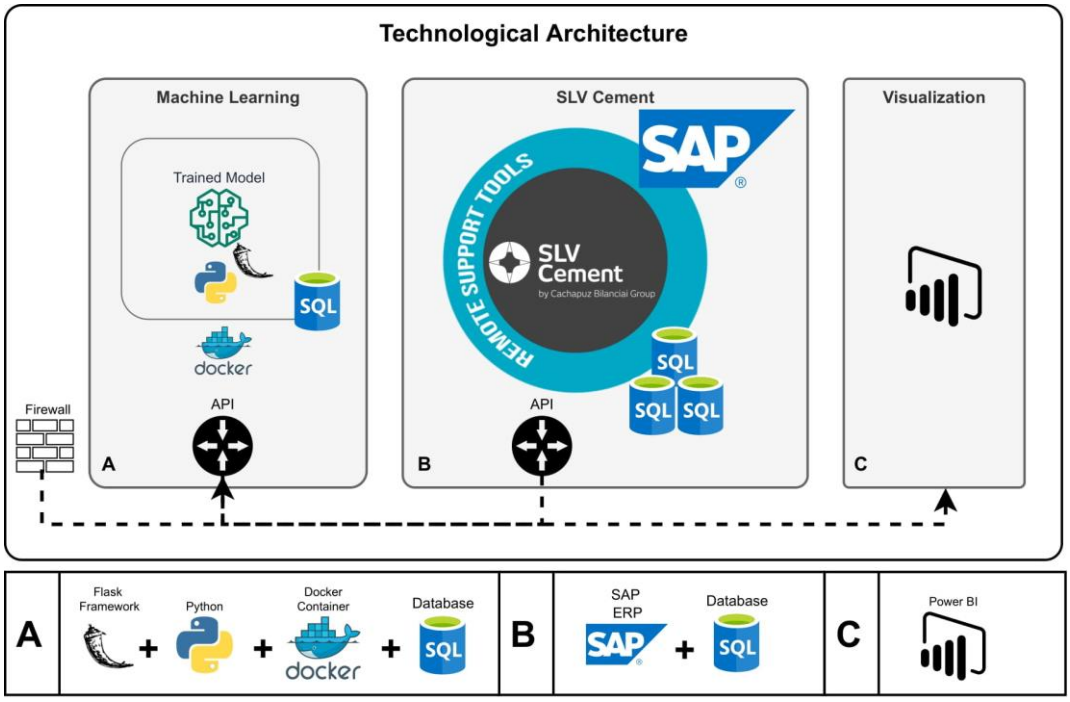

Figura 32 - Arquitetura Tecnológica da Implementação

A componente denominada de *Machine Learning* corresponde ao micro-serviço, o qual encontra-se distribuído através de *containers,* sendo estes geridos pela tecnologia *Docker.* O micro-serviço foi desenvolvido com base na *microframework Flask* [17]*,* cujo principal objetivo é permitir o desenvolvimento de APIs de forma rápida e fácil através da linguagem de programação *Python.* Esta componente está construída de forma a poder-se realizar o treino e previsões do modelo de *Machine Learning* escolhido, através da utilização dos dados alojados nas bases de dados SQL conectadas à componente *SLV Cement.*

A componente *SLV Cement* representa a plataforma SLV*,* a qual está ligada a diversas bases de dados SQL, bem como ao ERP SAP*.* Quando surge a necessidade de realizar uma previsão da possível existência de uma anomalia, a plataforma SLV, emite um pedido ao micro-serviço desenvolvido, enviando os dados necessários para o modelo de *Machine Learning* realizar a sua previsão. Como resposta será devolvido o valor 1 ou 0, que, respetivamente, representam a existência e a inexistência de uma anomalia no processo que se irá realizar. Por fim, a componente de *Visualization* é constituída por um *Dashboard* (desenvolvido utilizando a ferramenta PowerBI) como ilustrado na Figura 33.

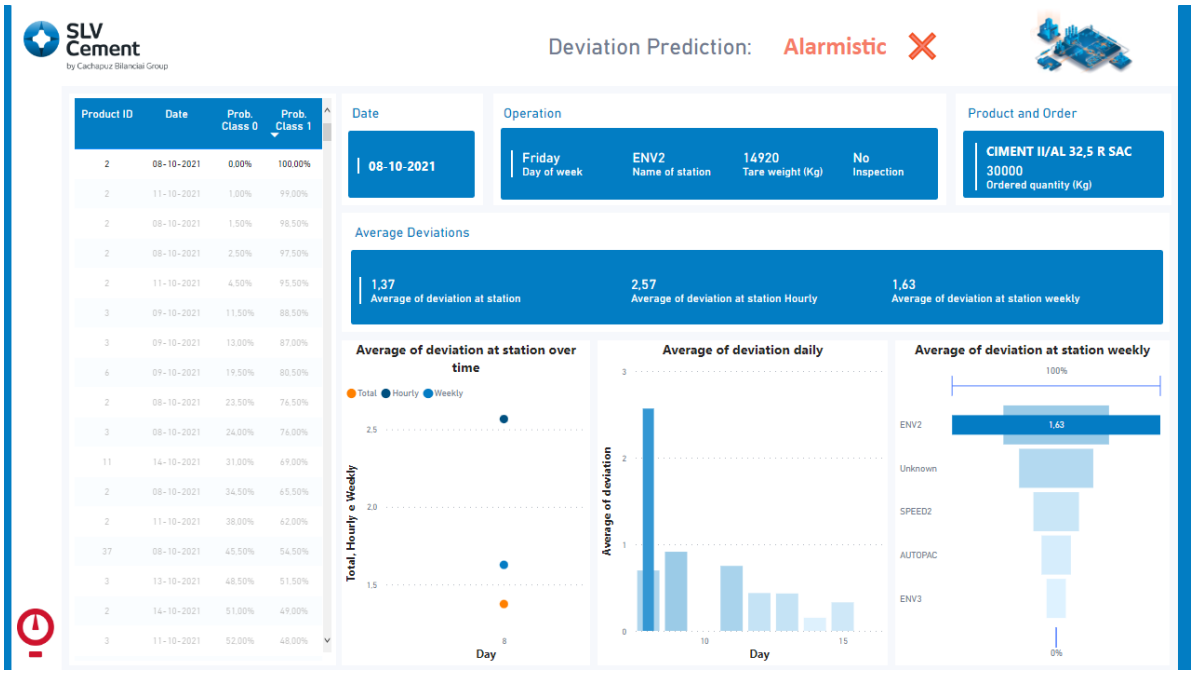

Figura 33 - Dashboard das Previsões de Desvios.

[17] https://flask.palletsprojects.com/en/2.1.x/

4. CONCLUSÃO

Este capítulo da presente dissertação tem como objetivo apresentar uma visão geral de todos os trabalhos realizados ao longo dos vários capítulos deste documento. Assim sendo, o capítulo Conclusão encontra-se dividido em três pontos elementares. Primeiramente, é abordada a síntese do trabalho efetuado, com o intuito de demonstrar de forma resumida todo o conteúdo abordado ao longo da dissertação. Depois, na discussão é levada a cabo uma avaliação dos resultados, a fundamentação das limitações do trabalho e, por fim, é elaborada uma lista dos trabalhos futuros.

4.1.1 Síntese do Trabalho Efetuado

Na Introdução, primeiro capítulo da presente dissertação, foi realizado todo um enquadramento referente ao projeto NeWeSt e ao impacto que a Transformação Digital têm nos vários setores da sociedade, e em especial na indústria. Além disso, neste primeiro capítulo foram aprofundadas as motivações que levaram à realização dos trabalhos desenvolvidos, bem como os objetivos, resultados esperados e a formulação do problema presente no caso de estudo. Também foi apresentada a metodologia CRISP-DM, que foi adota para conduzir o correto desenvolvimento e implementação do trabalho realizado.

Depois, no capítulo de Revisão de Literatura, foi dada a conhecer a estratégia utilizada para a pesquisa e obtenção de bibliografia referente a *Data Science, Data Mining, Machine Learning,* sistemas de pesagem e Anomalias, os quais foram posteriormente aprofundados. Por fim, o capítulo termina com a apresentação das várias ferramentas tecnológicas utilizadas no decorrer do caso de estudo.

O desenvolvimento do caso de estudo relativo à Previsão de Desvios em Processos de Carregamento de Sacos de Cimento é o terceiro capítulo da presente dissertação. Este tem por base a implementação metodologia CRISP-DM, ou seja, a elaboração de todas as tarefas que lhe são inerentes, com o objetivo de atingir os requisitos e solucionar os problemas que concernem este caso de estudo. A aplicação da metodologia CRISP-DM culminou no desenvolvimento e implementação de um micro-serviço que permite otimizar os processos de carregamento de sacos de cimento em organizações clientes da Cachapuz, alertando atempadamente para uma possível anomalia em termos do peso, no próximo processo a ocorrer.

4.1.2 Discussão

A conclusão dos trabalhos associados a esta dissertação resulta na necessidade de realizar um ponto de situação, que permita avaliar os resultados obtidos ao longo de todo o processo implementado face aos objetivos e requisitos expectáveis de serem alcançados com o desenvolvimento do projeto. Deste modo, uma apreciação global dos trabalhos realizados permite aferir que os objetivos definidos para o projeto foram alcançados, o que culminou no desenvolvimento de um micro-serviço capaz de realizar previsões referentes à existência ou não de anomalias relacionadas com o desvio do peso no processo a realizar.

No que se refere aos trabalhos desenvolvidos é de salientar a origem dos dados utilizados para treinar os modelos de *Machine Learning.* De uma forma geral, os dados normalmente utilizados em contextos de previsão de anomalias são, na maioria dos casos, dados provenientes de sensores alojados nos equipamentos associados aos processos. Contudo, no caso de estudo elaborado, na ausência de tais dados, foram utilizados dados referentes ao processo de carregamento, designado por *Dispatch workflow*, juntamente com atributos que foram desenvolvidos através do processo de *feature engineering*. Esta estratégia apresenta algumas vantagens, sendo a principal delas a possibilidade de implementar um sistema capaz de prever anomalias sem a necessidade de se realizarem grandes investimentos em equipamentos com sensores que permitam obter dados distintos, uma vez que os resultados obtidos com esta abordagem foram satisfatórios para os especialistas da área que avaliaram o artefacto desenvolvido.

O estudo realizado no âmbito da dissertação permitiu averiguar que o modelo *Random Forest* foi o que apresentou melhores resultados para a previsão de anomalias no decorrer do processo de carregamento de sacos de cimento em veículos de transporte. Este modelo RF, com uma mediana da métrica AUC de 0,937, foi o que demonstrou melhores resultados. Além disso, os resultados referentes à métrica AUC foram obtidos a partir da utilização do mecanismo de *Rolling Window*, o qual permitiu que estes valores fossem desenvolvidos num ambiente realista e robusto. Os resultados relacionados com o modelo selecionado foram então posteriormente analisados segundo a biblioteca SHAP, o que permitiu compreender a influência dos vários atributos para a realização de uma previsão por parte do modelo. Neste contexto de análise foi possível observar que "Average_Station_Houry" é o atributo que mais impacta o resultado devolvido pelo modelo *Random Forest.*

Para além do que já foi abordado, é fundamental referir que o setor da pesagem industrial não apresenta grandes estudos e desenvolvimentos em termos da aplicação de projetos de *Machine*

Learning. Este facto levou à dificuldade em determinar o melhor caminho a seguir, bem como as estratégias a serem utilizadas, com o objetivo de se obter os melhores resultados com a elaboração do projeto. No entanto, a conclusão e os resultados obtidos no âmbito do caso de estudo, permitiram abrir diversas portas para futuros projetos e desafios que possam vir a surgir neste setor tão competitivo.

4.1.3 Trabalho Futuro

Os resultados alcançados com o desenvolvimento dos trabalhos levados a cabo no âmbito da presente dissertação permitem alterar a visão face à aplicação de técnicas de *Machine Learning* em contexto industrial, mais propriamente no setor relacionado à pesagem. Em trabalho futuro, aspetos como a utilização de uma quantidade superior de dados associada a uma maior variedade de clientes da Cachapuz deverão ser considerados. Este facto, deriva das limitações atuais relativamente aos dados disponíveis para o desenvolvimento e implementação do caso de estudo. Além disso, e de forma a trazer uma maior profundidade para o projeto, a introdução e recolha de informação derivada de sensores nos equipamentos e de sensores climáticos, bem como do registo das calibrações executadas nas respetivas máquinas, seriam dados fundamentais e que possivelmente trariam um impacto positivo para os resultados obtidos pelos vários modelos de *Machine Learning* aplicados.

Os aspetos relacionados com os dados não são a única temática que merece atenção em iterações futuras do caso de estudo, a aplicação de diferentes modelos de *Machine Learning* e a introdução de ferramentas de *AutoML* são um passo a ter futuramente em consideração para efeitos de *Benchmarking*. Para além disso, e com a ambição de expandir e enaltecer o trabalho desenvolvido, poder-se-ia utilizar mecanismos automáticos de extração de dados e de *deployment* através de ferramentas de *Continuous Integration/Continuous Delivery* (CI/CD) como, por exemplo o *Jenkins* ou o *Azure Pipelines*

BIBLIOGRAFIA

Afsar, P., Cortez, P., & Santos, H. (2018). Automatic human trajectory destination prediction from video*. In 2018 Expert Systems with Applications*, 110, 41-51. https://doi.org/10.1016/j.eswa.2018.03.035

Agarwal, S. (2013). Data mining: concepts and techniques. *In 2013 international conference on machine intelligence and research advancement,* 203-207. https://doi.org/10.1109/ICMIRA.2013.45

Aggarwal, C. C. (2017). An introduction to outlier analysis. *In Outlier analysis*, 1-34. https://doi.org/10.1007/978-3-319-47578-3_1

Ahmed, M., Mahmood, A. N., & Hu, J. (2014). Outlier detection. *The state of the art in intrusion prevention and detection*, 3-21.

Ahmed, M., Mahmood, A. N., & Hu, J. (2016a). A survey of network anomaly detection techniques. *In 2016 Journal of Network and Computer Applications*, *60*, 19-31. https://doi.org/10.1016/j.jnca.2015.11.016

Ahmed, M., Mahmood, A. N., & Islam, M. R. (2016b). A survey of anomaly detection techniques in financial domain. *In 2016 Future Generation Computer Systems*, *55*, 278-288. https://doi.org/10.1016/j.future.2015.01.001

Akter, T., & Hernandez, S. (2021). Truck industry classification from anonymous mobile sensor data using machine learning. *In 2021 International Journal of Transportation Science and Technology*. https://doi.org/10.1016/j.ijtst.2021.07.001

Alonso, J., Belanche, L., & Avresky, D. R. (2011). Predicting software anomalies using machine learning techniques. *In 2011 IEEE 10th international symposium on network computing and applications*, 163-170. https://doi.org/10.1109/NCA.2011.29

Arning, A., Agrawal, R., & Raghavan, P. (1996). A Linear Method for Deviation Detection in Large Databases. *In 1996 Knowleadge Discover in Databases Proceedings*, *1141*(50), 972-981.

Baheti, R., & Gill, H. (2011). Cyber-physical systems. *The impact of control technology*, *12*(1), 161-166.

Barlas, P., Lanning, I., & Heavey, C. (2015). A survey of open-source data science tools. *In 2015 International Journal of Intelligent Computing and Cybernetics*, *8*(3), 232-261 https://doi.org/10.1108/IJICC-07-2014-0031

Belgiu, M., & Drăguţ, L. (2016). Random forest in remote sensing: A review of applications and future directions*. In 2016 ISPRS Journal of Photogrammetry and Remote Sensing*, *114*, 24-31. https://doi.org/10.1016/j.isprsjprs.2016.01.011

Bento, A. (2012). Como fazer uma revisão da literatura: Considerações teóricas e práticas. *Revista JA (Associação Académica da Universidade da Madeira)*, *7*(65), 42-44.

Berthelot, D., Carlini, N., Goodfellow, I., Papernot, N., Oliver, A., & Raffel, C. A. (2019). Mixmatch: A holistic approach to semi-supervised learning. *In 2019 Advances in neural information processing systems*, *32*(454), 5049-5059.

Bhuyan, M. H., Bhattacharyya, D. K., & Kalita, J. K. (2013). Network anomaly detection: methods, systems, and tools. *In 2014 IEEE Communications Surveys & Tutorials, 16*(1), 303-336. https://doi.org/ 10.1109/SURV.2013.052213.00046

Bitkom, VDMA, & ZVEI. (2016). *Implementation Strategy Industrie 4.0: Report on the Results of the Industrie 4.0 Plataform*.

Breiman, L. (2001). Random Forests. *In 2001 Machine Learning*, *45*, 5–32. https://doi.org/10.1023/A:1010933404324

Brittain, J., Cendón, M., Nizzi, J., & Pleis, J. (2018). Data Scientist's Analysis Toolbox: Comparison of Python, R, and SAS Performance. *In 2018 SMU Data Science Review*, *1*(2).

Carvalho, T. P., Soares, F. A., Vita, R., Francisco, R. D. P., Basto, J. P., & Alcalá, S. G. (2019). A systematic literature review of machine learning methods applied to predictive maintenance. *In 2019 Computers & Industrial Engineering*, *137*. https://doi.org/10.1016/j.cie.2019.106024

Charbuty, B. & Mohsin Abdulazeez, A. (2021). Classification based on decision tree algorithm for machine learning. In 2021 Journal of Applied Science and Technology Trends, *2*(1), 20–28. https://doi.org/10.38094/jastt20165

Dayan, P., & Niv, Y. (2008). Reinforcement learning: the good, the bad and the ugly. *In 2008 Current opinion in neurobiology*, *18*(2), 185-196. https://doi.org/10.1016/j.conb.2008.08.003

Deniz, M., Alam, F., Yuan, C., Ko, H. S., & Lee, H. F. (2022). Single Image based Volume Estimation for Dump Trucks in Earthmoving using Machine Learning Approach. *In 2022 EPiC Series in Built Environment*, *3*, 380-388. https://doi.org/10.29007/h7ct

Dhanachandra, N., Manglem, K., & Chanu, Y. J. (2015). Image segmentation using K-means clustering algorithm and subtractive clustering algorithm. *Procedia Computer Science*, *54*, 764-771. https://doi.org/10.1016/j.procs.2015.06.090

Ebert, C., & Duarte, C. H. C. (2018). Digital Transformation. *In IEEE Software*, *35*(4), 16–21. https://doi.org/10.1109/MS.2018.2801537

Fitzgerald, M., Kruschwitz, N., Bonnet, D., & Welch, M. (2013). *Embracing Digital Technology, A New Strategic Imperative*. MIT Sloan Managment Review. http://sloanreview.mit.edu/faq/

Cachapuz - Weighing & Logistics Systems, (2919). *Anexo Técnico da Proposta de Candidatura do projeto NeWeSt – New generation of cyberphysical Weighing Systems ao Sistema De Incentivos À Investigação E Desenvolvimento Tecnológico (Si I&DT).*

Chapelle, O., Schölkopf, B., & Zien, A. (2006). *Semi-supervised learning* (1st ed.). Cambridge: The MIT Press.

Chapman, P., Clinton, J., Kerber, R., Khabaza, T., Reinartz, T., Shearer, C. & Wirth, R. (2000). *CRISP-DM 1.0 Step-by-step data mining guide.*

Cortez, P., & Santos, M. F. (2013). Knowledge discovery and business intelligence.

Cumbley, R., & Church, P. (2013). Is Big Data creepy? *In 2013 Computer Law and Security Review*, *29*(5), 601–609. https://doi.org/10.1016/j.clsr.2013.07.007

Davis T. & Higgins, J. (2013). *A Blockbuster Failure: How an Outdated Business Model Destroyed a Giant*. Chapter 11 Bankruptcy Case Studies. 11. https://ir.law.utk.edu/utk_studlawbankruptcy/11

Deng, H., Zhou, Y., Wang, L., & Zhang, C. (2021). Ensemble learning for the early prediction of neonatal jaundice with genetic features. *In 2021 BMC Medical Informatics and Decision Making*, *21*(1), 1-11. https://doi.org/10.1186/s12911-021-01701-9

Dhaliwal, S. S., Nahid, A. A., & Abbas, R. (2018). Effective intrusion detection system using XGBoost. *In 2018 Information*, *9*(7), 149. https://doi.org/10.3390/info9070149

Domingos, P. (2012). A few useful things to know about machine learning. *In 2012 Communications of the ACM*, 55(10), 78-87. https://doi.org/10.1145/2347736.2347755

Donoho, D. (2017). 50 years of data science. *In 2017 Journal of Computational and Graphical* Statistics, *26*(4), 745-766. https://doi.org/10.1080/10618600.2017.1384734

Dwivedi, A. K. (2018). Performance evaluation of different machine learning techniques for prediction of heart disease. *In Neural Computing and Applications*, *29*(5), 685–693. https://doi.org/10.1007/s00521-016-2604-1

Emmert-Streib, F., & Dehmer, M. (2019). Defining data science by a data-driven quantification of the community. *In 2018 Machine Learning and Knowledge Extraction*, *1*(1), 235-251. https://doi.org/10.3390/make1010015

Fahim, M., & Sillitti, A. (2019). Anomaly Detection, Analysis, and Prediction Techniques in IoT Environment: A Systematic Literature Review. *In 2019 IEEE Access*, 7. https://doi.org/10.1109/ACCESS.2019.2921912

Fahlman S. (1988). *An Empirical Study of Learning Speed in Back-Propagation Networks.* In Technical Report CMU-CS-88-162. Carnegie-Mellon University.

Fang, J., Su, H., and Xiao, Y. (2018). Will artificial intelligence surpass human intelligence? *In 2018 SSRN Electronic Journal.* https://doi.org/10.2139/ssrn.3173876

Fawcett, Tom. (2006). An introduction to ROC analysis. *In Pattern Recognition Letters*, *27*(8), 861–874. https://doi.org/10.1016/j.patrec.2005.10.010

Fayyad, U. M., Piatetsky-Shapiro, G., & Smyth, P. (1996a). Knowledge Discovery and Data Mining: Towards a Unifying Framework. *In 1996 Knowledge Discovery in Databases Proceedings*, *96*, 82-88.

Fayyad, U., Piatetsky-Shapiro, G. & Smyth, P. (1996b). From Data Mining to Knowledge Discovery in Databases. *AI Magazine*, *17*(3), 37-54.

Freund, Y., Schapire, R., & Abe, N. (1999). A short introduction to boosting. *In 1999 Journal of Japanese Society for Artificial Intelligence*, *14*(5), 771-780.

Friedman, J., Hastie, T., & Tibshirani, R. (2000). Additive logistic regression: a statistical view of boosting (with discussion and a rejoinder by the authors). *In 2000 The Annals of statistics*, *28*(2), 337-407. https://doi.org/10.1214/aos/1016218223

Friedman, J. H. (2001). Greedy function approximation: a gradient boosting machine. *In 2001 The Annals of statistics*, *29*(5), 1189-1232. https://doi.org/10.1214/aos/1013203451

Gandhi, R. (2018). *Support Vector Machine – Introduction to Machine Learning Algorithms*. Towards Data Science. https://towardsdatascience.com/support-vector-machine-introduction-to-machine-learning-algorithms-934a444fca47

Gardner, M. W., & Dorling, S. R. (1998). Artificial neural networks (the multilayer perceptron)—a review of applications in the atmospheric sciences. *In 1998 Atmospheric Environment*, *32*(14-15), 2627-2636. https://doi.org/10.1016/S1352-2310(97)00447-0

Gentleman, R., & Carey, V. J. (2008). Unsupervised machine learning. *In Bioconductor case studies* (pp. 137-157). Springer, New York, NY.

Gu, X., & Wang, H. (2009). *Online anomaly prediction for robust cluster systems*. In 2009 IEEE 25th International Conference on Data Engineering (pp. 1000-1011). IEEE. https://doi.org/10.1109/ICDE.2009.128

Gubaev, K., Podryabinkin, E. V., & Shapeev, A. V. (2018). Machine learning of molecular properties: Locality and active learning. *In 2018 The Journal of chemical physics*, *148*(24), 24172. https://doi.org/10.1063/1.5005095

Gungor, O. E., Al-Qadi, I. L., & Mann, J. (2018). Detect and charge: Machine learning based fully data-driven framework for computing overweight vehicle fee for bridges. *In 2018 Automation in Construction*, *96*, 200-210. https://doi.org/10.1016/j.autcon.2018.09.007

Halimic, M., & Balachandran, W. (1995). Kalman filter for dynamic weighing system. *In 1995 Proceedings of the IEEE International Symposium on Industrial Electronics* (2nd ed., pp. 786-791). IEEE https://doi.org/10.1109/ISIE.1995.497286

Hastie, T., Tibshirani, R., Friedman, J. H., & Friedman, J. H. (2009). *The elements of statistical learning: data mining, inference, and prediction* (2nd ed., pp. 1-758). New York: Springer

Hayashi, C. (1998). What is Data Science? Fundamental Concepts and a Heuristic Example. In: Hayashi, C., Yajima, K., Bock, HH., Ohsumi, N., Tanaka, Y., Baba, Y. (eds) *Data Science, Classification, and Related Methods. Studies in Classification, Data Analysis, and Knowledge Organization* (pp. 40-51). Springer, Tokyo. https://doi.org/10.1007/978-4-431-65950-1_3

Hess, T., Matt, C., Benlian, A., & Wiesböck, F. (2016). Options for Formulating a Digital Transformation Strategy. *In 2016 MIS Quarterly Executive*. *15*(2), 123-139. https://aisel.aisnet.org/misqe/vol15/iss2/6

Hollander, M., Wolfe, D. A., & Chicken, E. (2013). *Nonparametric statistical methods*. John Wiley & Sons.

James, G., Witten, D., Hastie, T., & Tibshirani, R. (2013). *An introduction to statistical learning With Applications in R* (2nd ed.). New York: Springer

Kaelbling, L. P., Littman, M. L., & Moore, A. W. (1996). Reinforcement learning: A survey. *In 1996 Journal of artificial intelligence research*, *4*, 237-285. https://doi.org/10.1007/978-981-33-4859-2_29

Kamel, M. & Selim, S. Z. (1994) New algorithms for solving the fuzzy clustering problem. *In 1994 Pattern Recognition. 27*(3), 421-428. https://doi.org/10.1016/0031-3203(94)90118-X

Karatzoglou, A., Meyer, D., & Hornik, K. (2006). Support vector machines in R. *In 2006 Journal of statistical software*, *15*(9), 1-28. https://doi.org/10.18637/jss.v015.i09

Kim, S., Lee, J., Park, M. S., & Jo, B. W. (2009). Vehicle signal analysis using artificial neural networks for a bridge weigh-in-motion system. *In 2009 Sensors*, *9*(10), 7943-7956. https://doi.org/10.3390/s91007943

Kotsiantis, S. B., Zaharakis, I., & Pintelas, P. (2007). Supervised machine learning: A review of classification techniques. *Proceedings of the 2007 conference on Emerging Artificial Intelligence Applications in Computer Engineering: Real Word AI Systems with Applications in eHealth, HCI, Information Retrieval and Pervasive Technologies, 160*(1), 3-24.

Larose, D.T. (2005). *Discovering Knowledge in Data: An Introduction to Data Mining*. Wiley Blackwell

Li, L., & Zhang, X. (2010). Study of data mining algorithm based on decision tree. *In 2010 International Conference On Computer Design and Applications*, *1, 155-158.* https://doi.org/10.1109/ICCDA.2010.5541172

Li, Shenglong, & Zhang, Xiaojing. (2020). Research on orthopedic auxiliary classification and prediction model based on XGBoost algorithm. *In 2020 Neural Computing and Applications*, *32*(4), 1971–1979. https://doi.org/10.1007/s00521-019-04378-4

Lin, W., & Huang, L. (2018). Design of Port Unattended Weighbridge System Based on Internet of Things. *In 2018 3rd International Conference on Automation, Mechanical Control and Computational Engineering*, *166,* 194-198. https://doi.org/10.2991/amcce-18.2018.34

Liu, Q., Feng, C., Song, Z., Louis, J., & Zhou, J. (2019). Deep learning model comparison for vision-based classification of full/empty-load trucks in earthmoving operations. *In 2019 Applied Sciences*, *9*(22), 4871. https://doi.org/10.3390/app9224871

Lloyd, S. P. (1957). Least squares quantization in PCM. *In 1982 IEEE transactions on information theory*, *28*(2), 129-137.

Lohr, S. (2007). Google and I.B.M. Join in 'Cloud Computing' Research. The New York Times. https://www.nytimes.com/2007/10/08/technology/08cloud.html

Lorena, A. C., & de Carvalho, A. C. (2007). Uma introdução às support vector machines. *In 2007 Revista de Informática Teórica e Aplicada*, *14*(2), 43-67. https://doi.org/10.22456/2175-2745.5690

Lundberg, S. M., & Lee, S. I. (2017). A unified approach to interpreting model predictions. *In 2017 Advances in neural information processing systems*, *30*.

Lyman, P., & Varian, H, R. (2003). *How much information*, *2003*. Sims.Berkeley. http://www. sims. berkeley. edu/research/projects/how-much-info-2003/

MacQueen, J. B. (1967). Some methods for classification and analysis of multivariate observations. In 1967 5th Berkeley Symp. Math. Statist. Probability, *5*(1), 281–297.

Markham, K. (2020). *Simple guide to confusion matrix terminology*. Data School. https://www.dataschool.io/simple-guide-to-confusion-matrix-terminology/

Marshall, H., & Murphy, G. (2003). Factors affecting the accuracy of weighbridge systems. *In 2003 International Journal of Forest Engineering*, *14*(1), 67-79. https://doi.org/10.1080/14942119.2003.10702471

Matos, L. M., Domingues, A., Moreira, G., Cortez, P., & Pilastri, A. (2021). A comparison of machine learning approaches for predicting in-car display production quality. *In 2021 International Conference on Intelligent Data Engineering and Automated Learning* (pp. 3-11). Springer, Cham. https://doi.org/10.1007/978-3-030-91608-4_1

Mell, P., & Grace, T. (2011). The NIST definition of clouding computing recommendations national inst. of standards and technology. NIST Special Publication 800-145. http://csrc.nist.gov/publications/nistpubs/800-145/SP800-145.pdf

Mishra, A. (2020). *Metrics to Evaluate your Machine Learning Algorithm*. Medium. https://towardsdatascience.com/metrics-to-evaluate-your-machine-learning-algorithm-f10ba6e38234

Mitchell, R., Frank, E., & Holmes, G. (2022). GPUTreeShap: massively parallel exact calculation of SHAP scores for tree ensembles. *In 2022 PeerJ Computer Science*, *8*, e880. https://doi.org/10.7717/peerj-cs.880

Mitchell, T. M. (1997). Does machine learning really work?. *In 1997 AI Magazine*, *18*(3), 11. https://doi.org/10.1609/aimag.v18i3.1303

Modi, K., & Oza, B. (2016). Outlier analysis approaches in data mining. *In 2016 International Journal of Innovative Research in Technology*, *6*(7), 6-12.

Murchinson, J., & Haikes, C. (2007). *Google and IBM Announce University Initiative to Address Internet-Scale Computing Challenges*. Google Press. http://googlepress.blogspot.com/2007/10/google-and-ibm-announce-university_08.html

Nasteski, V. (2017). An overview of the supervised machine learning methods. In 2017 Horizons.B, *4,* 51–62. https://doi.org/10.20544/HORIZONS.B.04.1.17.P05

Nelson, D. (2020). *Supervised vs unsupervised learning*. Unite.AI, https://www.unite.ai/supervised-vs-unsupervised-learning/

North, M. (2012). *Data mining for the masses* (pp. 1-10). Global Text Project

Oliveira, N, Cortez, P, & Areal, N. (2017). The impact of microblogging data for stock market prediction: Using Twitter to predict returns, volatility, trading volume and survey sentiment indices. *In 2017 Expert Systems with Applications, 73*, 125–144. https://doi.org/10.1016/j.eswa.2016.12.036

Olsson, F. (2009). A literature survey of active machine learning in the context of natural language processing.

Park, S., Jung, D., Nguyen, H., & Choi, Y. (2021). Diagnosis of problems in truck ore transport operations in underground mines using various machine learning models and data collected by internet of things systems. *In 2021 Minerals*, *11*(10), 1128. https://doi.org/10.3390/min11101128

Pathak, M. (2020). *Evaluation Metrics For Machine Learning For Data Scientists*. Analytics Vidhya. https://www.analyticsvidhya.com/blog/2020/10/quick-guide-to-evaluation-metrics-for-supervised-and-unsupervised-machine-learning/

Pérez, F., Granger, B.E., & Hunter, J.D. (2011). Python: An Ecosystem for Scientific Computing. *In 2011 Computing in Science & Engineering*, *13*(2), 13-21. https://doi.org/10.1109/MCSE.2010.119

Praveena, M., & Jaiganesh, V. (2017). A literature review on supervised machine learning algorithms and boosting process. *In 2017 International Journal of Computer Applications, 169*(8), 32-35. https://doi.org/10.5120/ijca2017914816

Ramchoun, H., Ghanou, Y., Ettaouil, M., & Janati Idrissi, M. A. (2016). Multilayer perceptron: Architecture optimization and training. *In 2016 International Journal of Interactive Multimedia and Artificial Intelligence*, *4*(1). https://doi.org/10.9781/ijimai.2016.415

Ribeiro, M. T., Singh, S., & Guestrin, C. (2016). "Why should i trust you?" Explaining the predictions of any classifier. *In Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining* , 1135-1144. https://doi.org/10.1145/2939672.2939778

Richert, W. (2013). *Building machine learning systems with Python*. Packt Publishing Ltd.

Rosenblatt, F. (1958). The perceptron: a probabilistic model for information storage and organization in the brain. *Psychological review*, *65*(6), 386. https://doi.org/10.1037/h0042519

Ross, J., Sebastian, I., Beath, C., Scantlebury, S., Mocker, M., Fonstad, N., Kagan, M., Moloney, K., & Geraghty Krusel, S. (2016). Designing Digital Organizations. MIT Center for IS Research, *46*.

Sahakyan, M., Aung, Z., & Rahwan, T. (2021). Explainable artificial intelligence for tabular data: A survey. *In 2021 IEEE Access*, 9, 135392-135422. https://doi.org/10.1109/ACCESS.2021.3116481

Salazar-Reyna, R., Gonzalez-Aleu, F., Granda-Gutierrez, E. M., Diaz-Ramirez, J., Garza-Reyes, J. A., & Kumar, A. (2020). A systematic literature review of data science, data analytics and machine learning applied to healthcare engineering systems. *In 2020 Management Decision*. https://doi.org/10.1108/MD-01-2020-0035

Samuel, A. L. (1959). Some studies in machine learning using the game of checkers. *IBM Journal of research and development, 3*(3), 210–229. https://doi.org/10.1147/rd.33.0210.

Samuel, A. L. (1969). Some studies in machine learning using the game of checkers. II-Recent progress. In 1969 *Annual Review in Automatic Programming, 6*(1), 1–36. https://doi.org/10.1016/0066-4138(69)90004-4

Santos, B. P., Alberto, A., Lima, T. D. F. M., & Charrua-Santos, F. M. B. (2018). Indústria 4.0: Desafios e Oportunidades. *In 2018 Centro Federal de Educação Tecnológica Celso Suckow da Fonseca 4*(1), 111-124. https://doi.org/10.32358/rpd.2018.v4.316

scikit-learn. (2012). *3.1. Cross-validation: evaluating estimator performance.* Retrieved 2022, from https://scikit-learn.org/stable/modules/cross_validation.html

Semeler, A. R., Pinto, A. L., & Rozados, H. B. F. (2019). Data science in data librarianship: core competencies of a data librarian. *In 2019 Journal of Librarianship and Information Science*, *51*(3), 771-780. https://doi.org/10.1177/0961000617742465

Sendov, B. (1994). Entrando na era da informação. *In 1994 Estudos Avançados*, 8, 28-32. https://doi.org/10.1590/S0103-40141994000100008

Settles, B. (2009). Active learning literature survey.

Shapiro, N. Z., & Shapley, L. S. (1978). Values of large games, I: *A limit theorem. Mathematics of Operations Research*, *3*(1), 1-9. https://doi.org/10.1287/moor.3.1.1

Shearer, C. (2000). The CRISP-DM Model: The New Blueprint for Data Mining. *In 2000 Journal of Data Warehousing*, *5*(4), 13-22.

Shinde, P. P., & Shah, S. (2018). A review of machine learning and deep learning applications. *In 2018 Fourth international conference on computing communication control and automation.* 1-6. https://doi.org/ 10.1109/ICCUBEA.2018.8697857.

Shoaran, M., Haghi, B. A., Taghavi, M., Farivar, M., & Emami-Neyestanak, A. (2018). Energy-efficient classification for resource-constrained biomedical applications. *In 2018 IEEE Journal on Emerging and Selected Topics in Circuits and Systems*, *8*(4), 693-707. https://doi.org/ 10.1109/JETCAS.2018.2844733

Song, X., Wu, M., Jermaine, C., & Ranka, S. (2007). Conditional anomaly detection.*In 2007 IEEE Transactions on knowledge and Data Engineering*, *19*(5), 631-645. https://doi.org /10.1109/TKDE.2007.1009

Silwattananusarn, T., & Tuamsuk, K. (2012). Data mining and its applications for knowledge management: a literature review from 2007 to 2012. *In 2012 International Journal of Data Mining & Knowledge Management Process*, *2*(5), 13-24. https://doi.org /10.5121/ijdkp.2012.2502

Smith FJ (2006). Data science as an academic discipline. *In 2006 Data Science Journal*, *5,* 163–164. https://doi.org /10.2481/dsj.5.163

Snyder, H. (2019). Literature review as a research methodology: An overview and guidelines. *In 2019 Journal of Business Research*, *104*, 333–339. https://doi.org/10.1016/j.jbusres.2019.07.039

Soc. Coop. Bilanciai Campogalliano. (2009). Instructions for installing CPD cells.

Sun, Yanmin, Wong, Andrew K.C., & Kamel, Mohamed S. (2009). Classification of imbalanced data: A review. *In 2009 International Journal of Pattern Recognition and Artificial Intelligence*, *23*(4), 687–719. https://doi.org/10.1142/S0218001409007326

Sutton, R. S., & Barto, A. G. (2018). *Reinforcement learning: An introduction*. MIT press.

Tahaei, N., Yang, J. J., Chorzepa, M. G., Kim, S. S., & Durham, S. A. (2021). Machine learning of truck traffic classification groups from weigh-in-motion data. *In 2021 Machine Learning with Applications*, *6*, 100178. https://doi.org/10.1016/j.mlwa.2021.100178

Turing, A. M. (1950). Computing machinery and intelligence-am turing. In 1950 *Mind, 59*(236). https://doi.org/10.1093/mind/LIX.236.433

Valter, C., Pedro, M., Fares, J. A., Cristhina, M., Rocha, S., Cristina, N., Mercer, H., Sistema, O., Marilia De Souza, F., Valença, R., Braga De Andrade, R., Lucchesi, R. E., & Diretor-Superintendente, R. (2020). Sistemas das Indústria do Estado do Paraná. Sistema Fiep Confederação Nacional da Industria - CNI.

Van der Aalst, W. M. (2014). Data scientist: The engineer of the future. *Enterprise interoperability VI*, 13-26. https://doi.org/10.1007/978-3-319-04948-9_2

Van Engelen, J. E., & Hoos, H. H. (2020). A survey on semi-supervised learning. *In 2020 Machine Learning*, *109*(2), 373-440. https://doi.org/10.1007/s10994-019-05855-6

Vapnik, V. (1999). *The nature of statistical learning theory* (2nd ed.). Springer science & business media. https://doi.org/10.1007/978-1-4757-3264-1

Vasconcelos, P. (2018). *5 Top Python IDEs for Data Science - What is an IDE in Python?*. DataCamp Community. https://www.datacamp.com/community/tutorials/data-science-python-ide

Vieira, S., Lopez Pinaya, W. H., & Mechelli, A. (2019). Introduction to machine learning. *In 2020 Methods and Applications to Brain Disorders*, 1-20. https://doi.org/10.1016/B978-0-12-815739-8.00001-8

Vouk, M. A. (2008). Cloud computing - Issues, research and implementations. *In 2008 Journal of Computing and Information Technology*, *16*(4), 235–246. https://doi.org/10.1109/ITI.2008.4588381

Wang, Y., Pan, Z., Zheng, J., Qian, L., & Li, M. (2019). A hybrid ensemble method for pulsar candidate classification. *In 2019 Astrophysics and Space Science*, *364*(8), 1-13. https://doi.org/10.1007/s10509-019-3602-4

Wirth, R. & Hipp, J. (2000). Crisp-dm: Towards a standard process model for data mining. *In 2000 Proceedings of the 4th International Conference on the Practical Applications of Knowledge Discovery and Data Mining*, *1*, 29-39. https://doi.org/10.1.1.198.5133

Yan, W., Deng, L., Zhang, F., Li, T., & Li, S. (2019). Probabilistic machine learning approach to bridge fatigue failure analysis due to vehicular overloading. *In 2019 Engineering Structures*, *193*, 91-99. https://doi.org/10.1016/j.engstruct.2019.05.028

Yao, Z., Huang, Q., Ji, Z., Li, X., & Bi, Q. (2021). Deep learning-based prediction of piled-up status and payload distribution of bulk material. *In 2021 Automation in Construction*, *121*, 103424. https://doi.org/10.1016/j.autcon.2020.103424

Zenati, H., Romain, M., Foo, C. S., Lecouat, B., & Chandrasekhar, V. (2018). Adversarially learned anomaly detection. *In 2018 IEEE International conference on data mining*, 727-736. https://doi.org/10.1109/ICDM.2018.00088

Zhou, Y., Pei, Y., Zhou, S., Zhao, Y., Hu, J., & Yi, W. (2021). Novel methodology for identifying the weight of moving vehicles on bridges using structural response pattern extraction and deep learning algorithms. *In 2021 Measurement*, *168*, 108384. https://doi.org/10.1016/j.measurement.2020.108384

Zhu, X. (2008). Semi-supervised learning literature survey. *Technical Report, University of Wisconsin-Madi*

APÊNDICE I – RESULTADOS DA ANÁLISE QUALITATIVA UTILIZANDO A FERRAMENTA *PANDAS PROFILING*

TipoDoc: Categorical

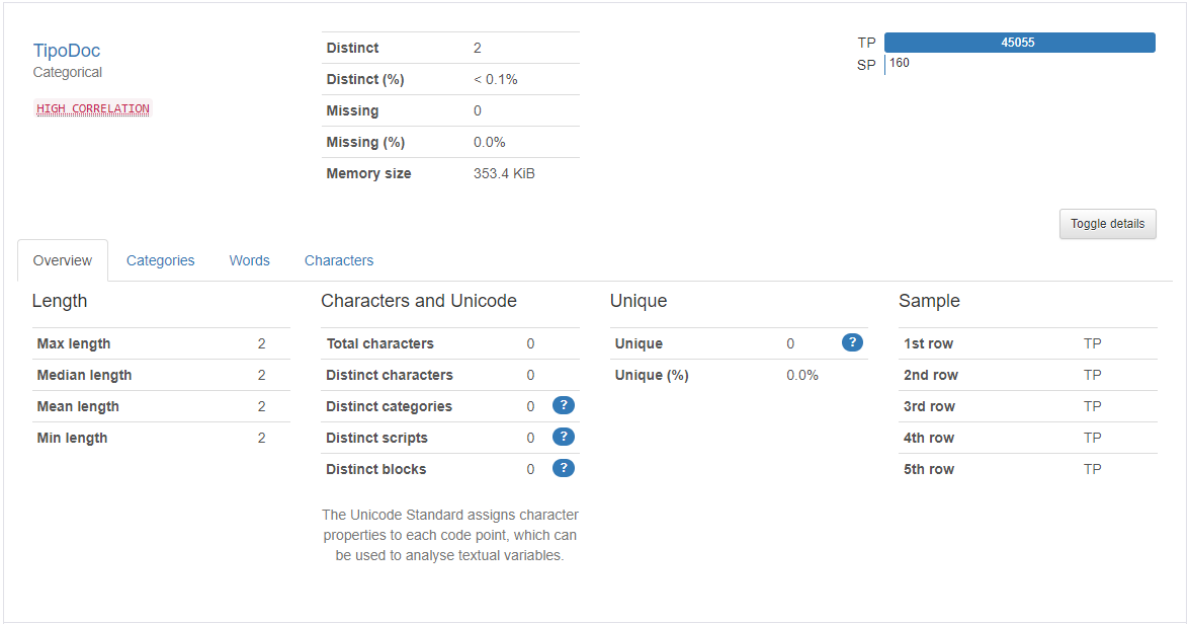

Figura 34 - Análise Qualitativa da Variável TipoDoc.

TipoViatura: Categorical

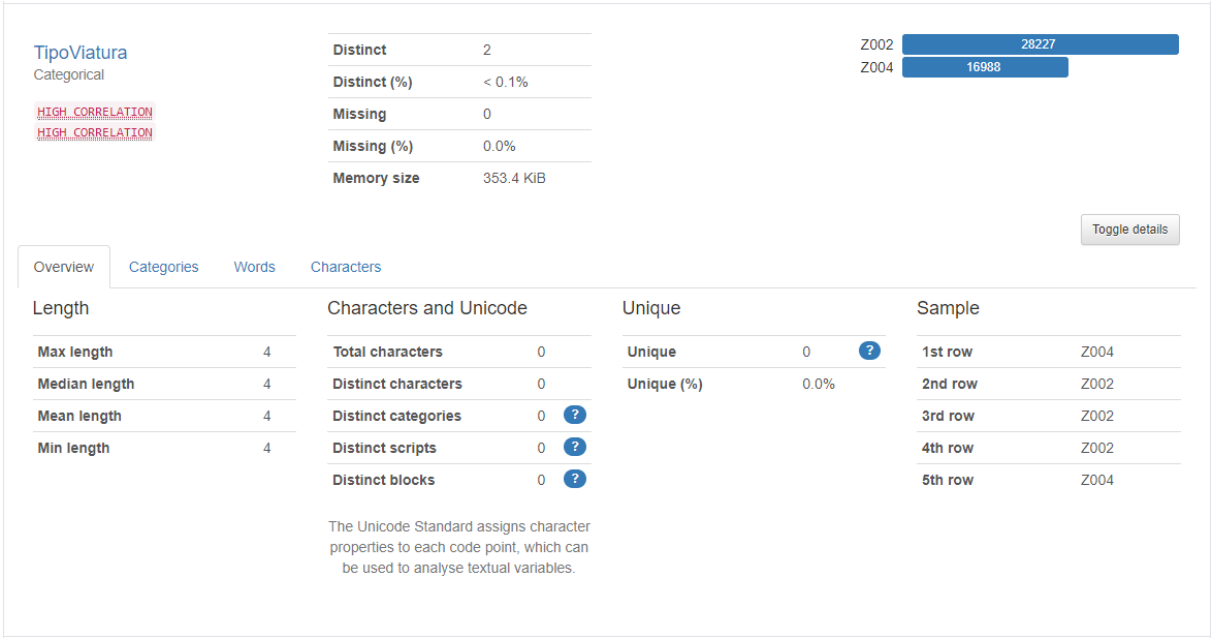

Figura 35 - Análise Qualitativa da Variável TipoViatura.

CodProduto: Numerical

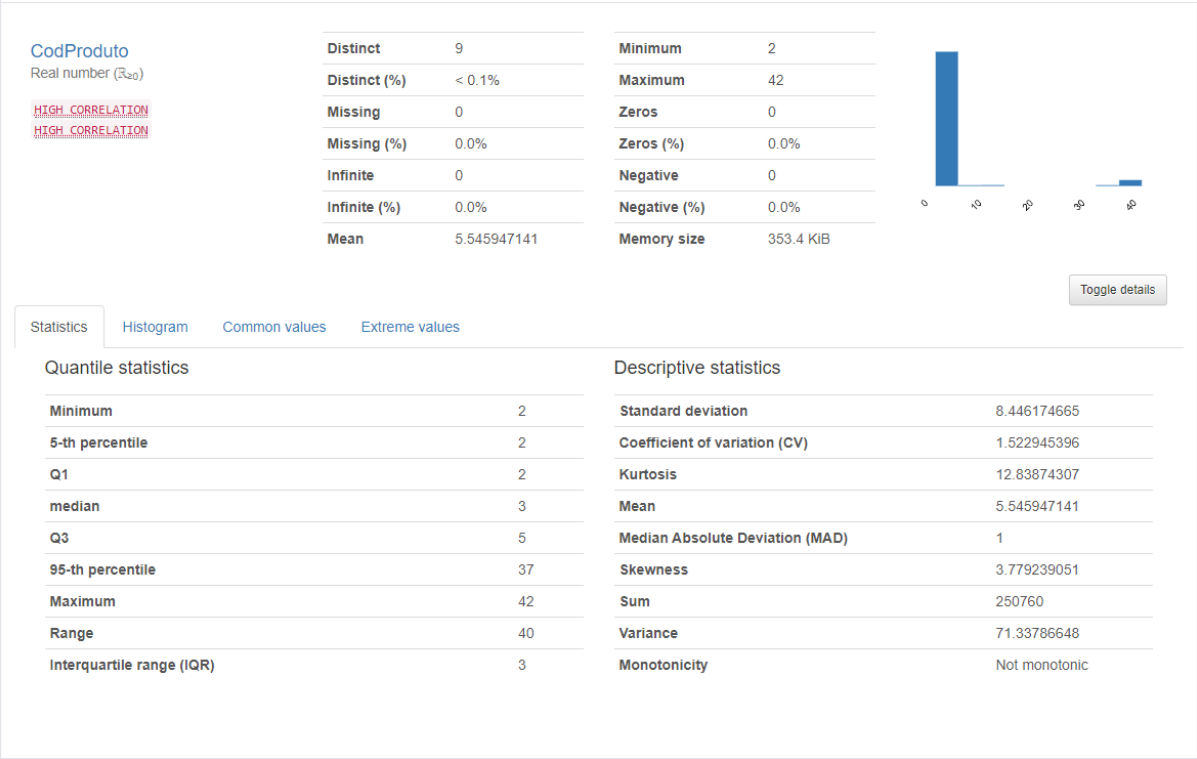

Figura 36 - Análise Qualitativa da Variável CodProduto.

DescProduto: Categorical

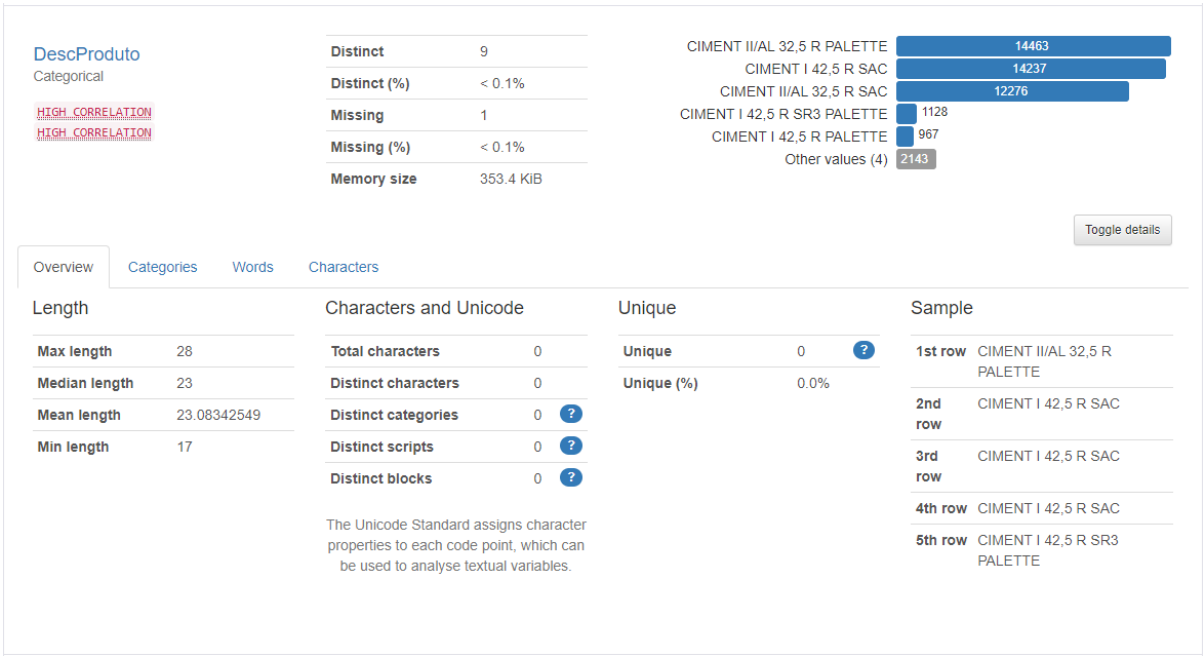

Figura 37 - Análise Qualitativa da Variável DescProduto.

estado: Categorical

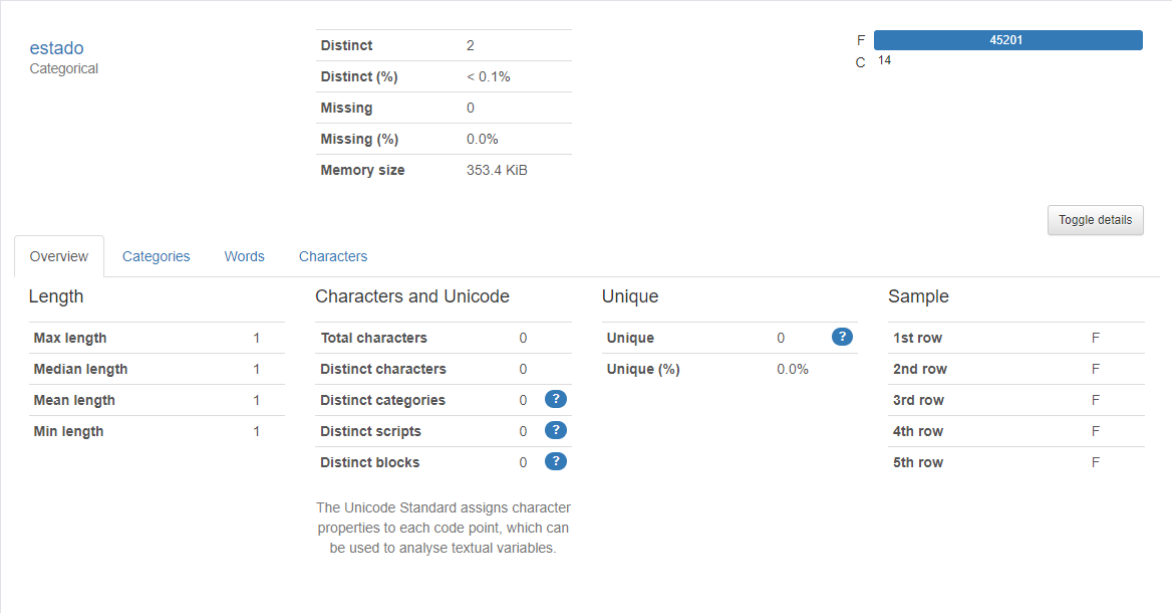

Figura 38 - Análise Qualitativa da Variável Estado.

Tara: Numerical

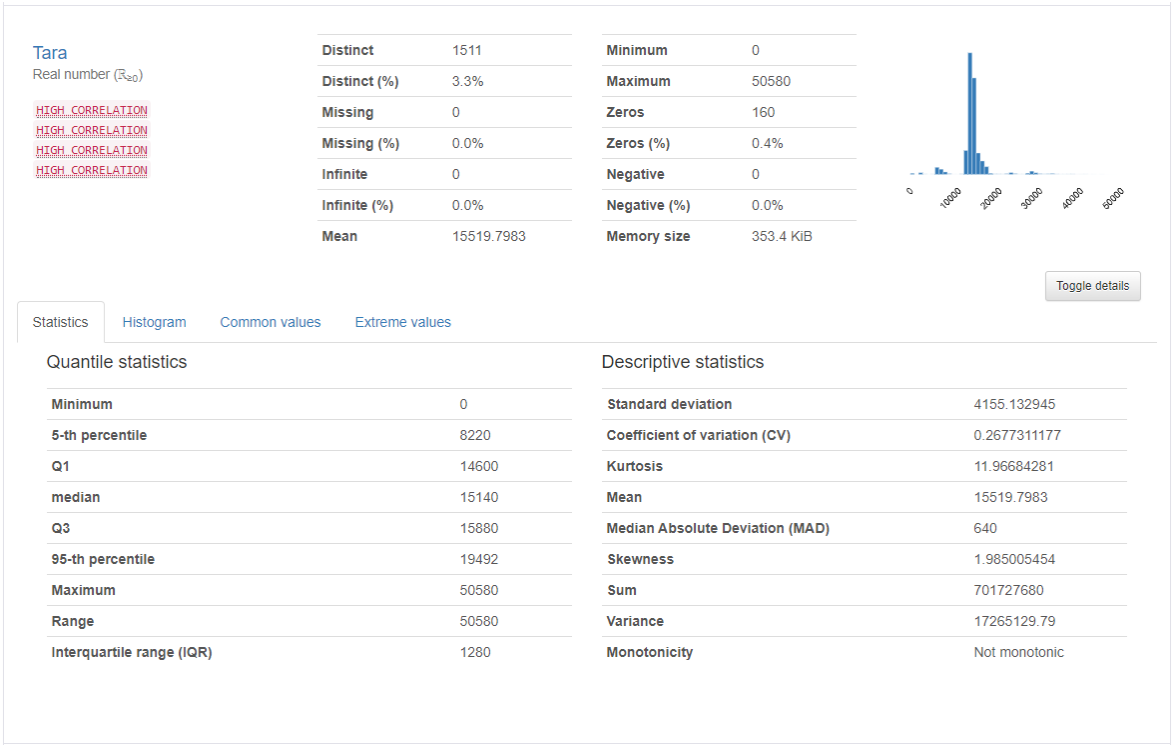

Figura 39 - Análise Qualitativa da Variável Tara.

bruto: Numerical

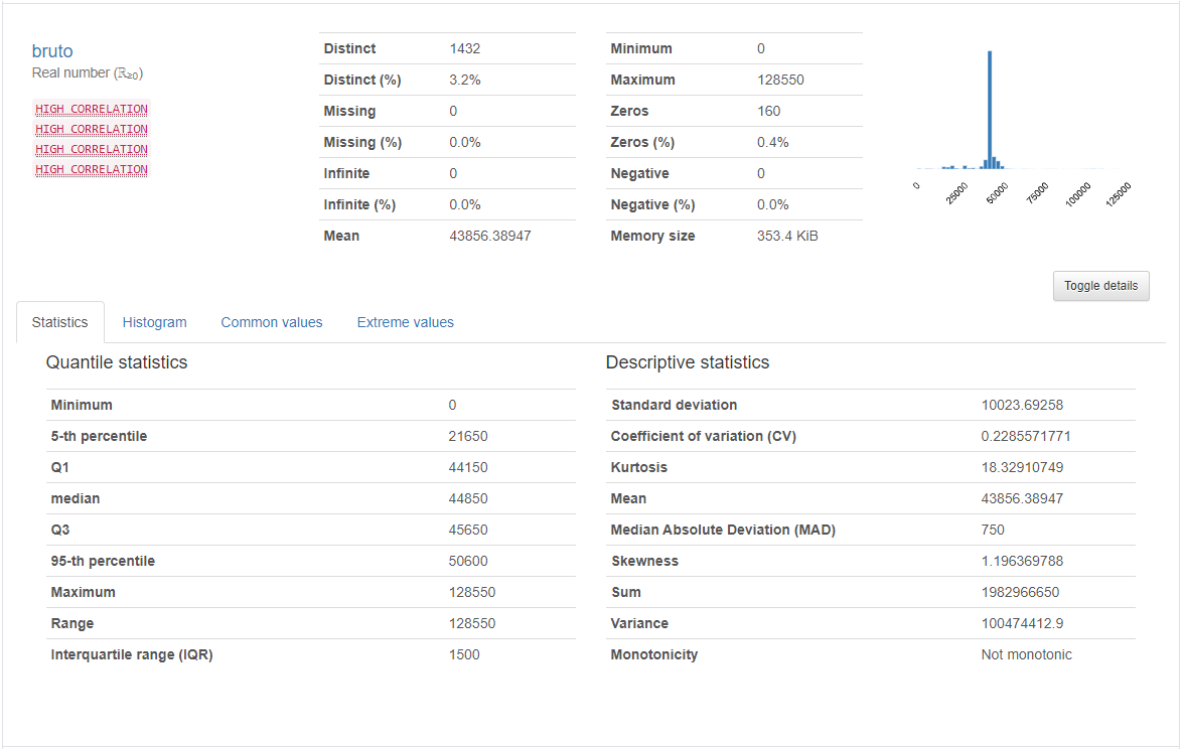

Figura 40 - Análise Qualitativa da Variável bruto.

PostoOperacao: Categorical

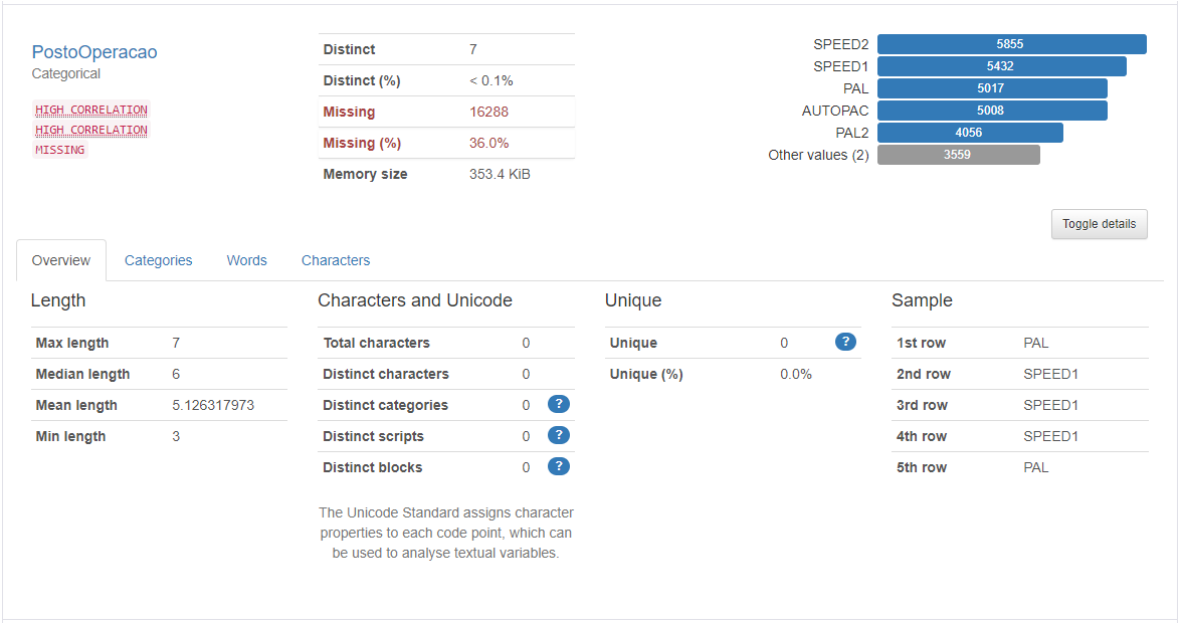

Figura 41 - Análise Qualitativa da Variável PostoOperacao.

Matricula: Categorical

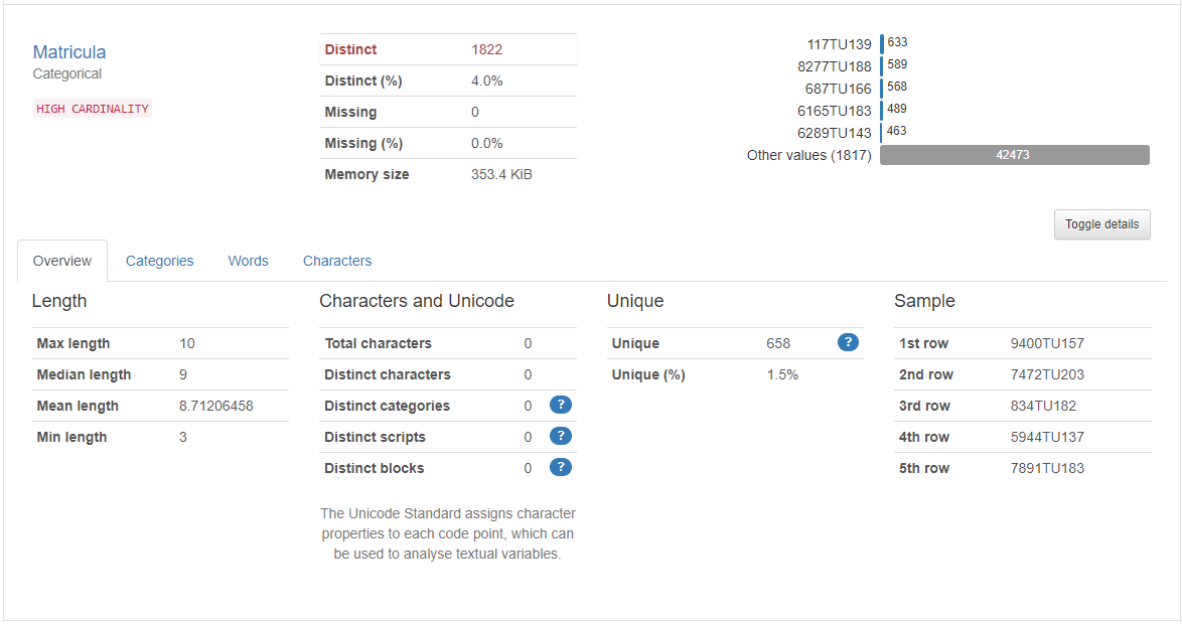

Figura 42 - Análise Qualitativa da Variável Matricula.

NomeMotorista: Categorical

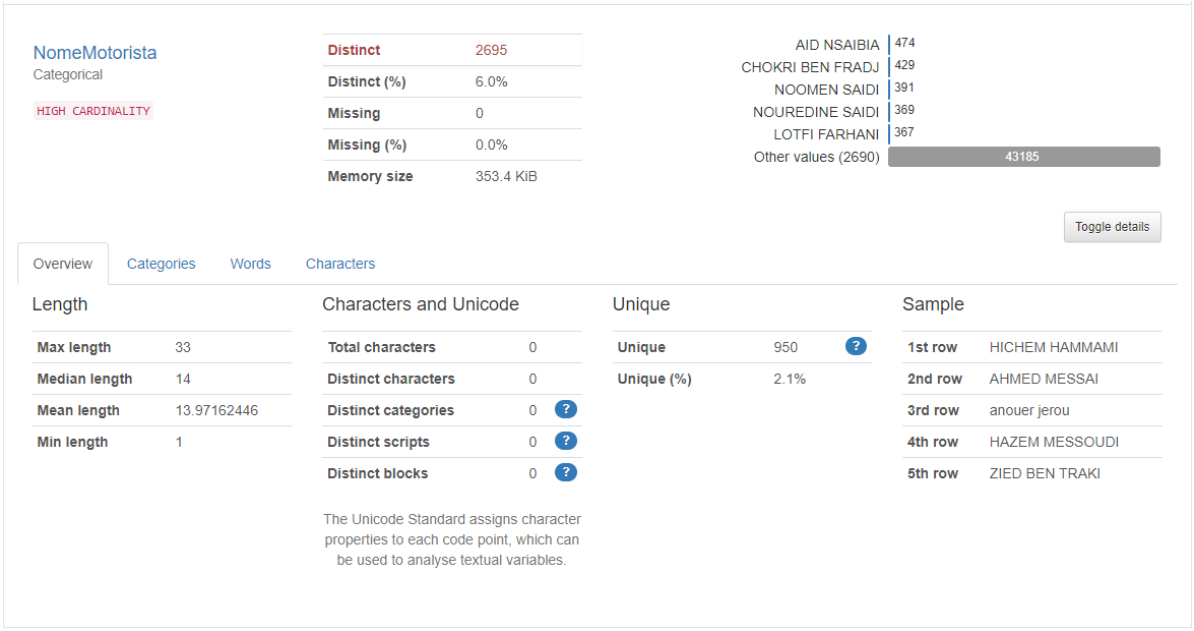

Figura 43 - Análise Qualitativa da Variável NomeMotorista.

Liquido: Numerical

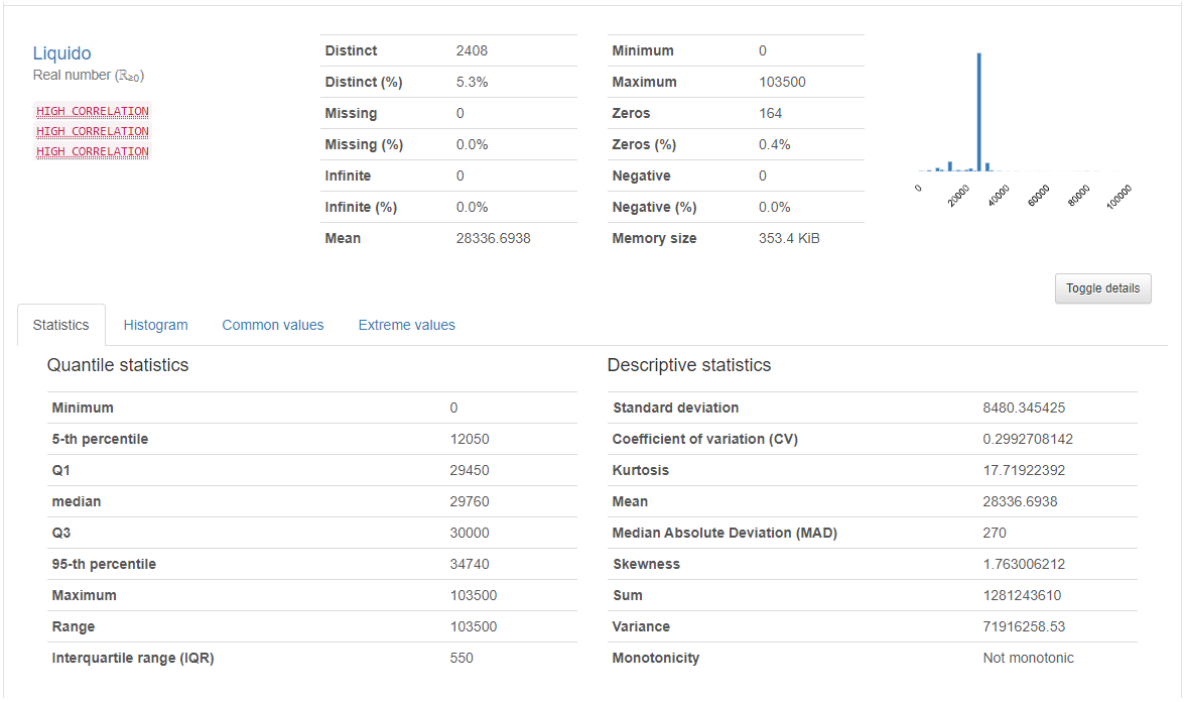

Figura 44 - Análise Qualitativa da Variável Liquido.

DataCriacao: DateTime

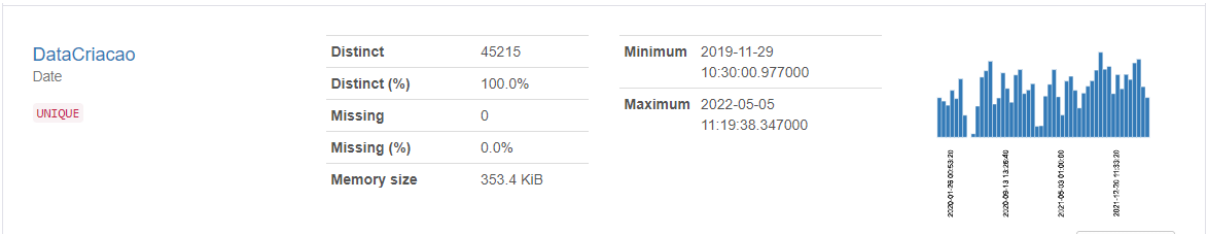

Figura 45 - Análise Qualitativa da Variável DataCriacao.

TaraData: DateTime

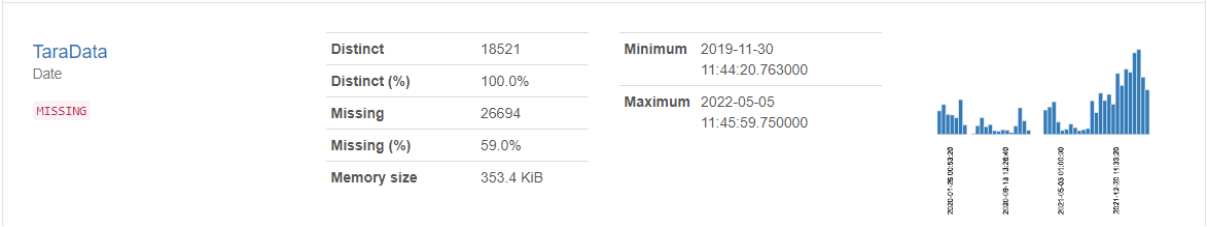

Figura 46 - Análise Qualitativa da Variável TaraData.

QtdPedida: Numerical

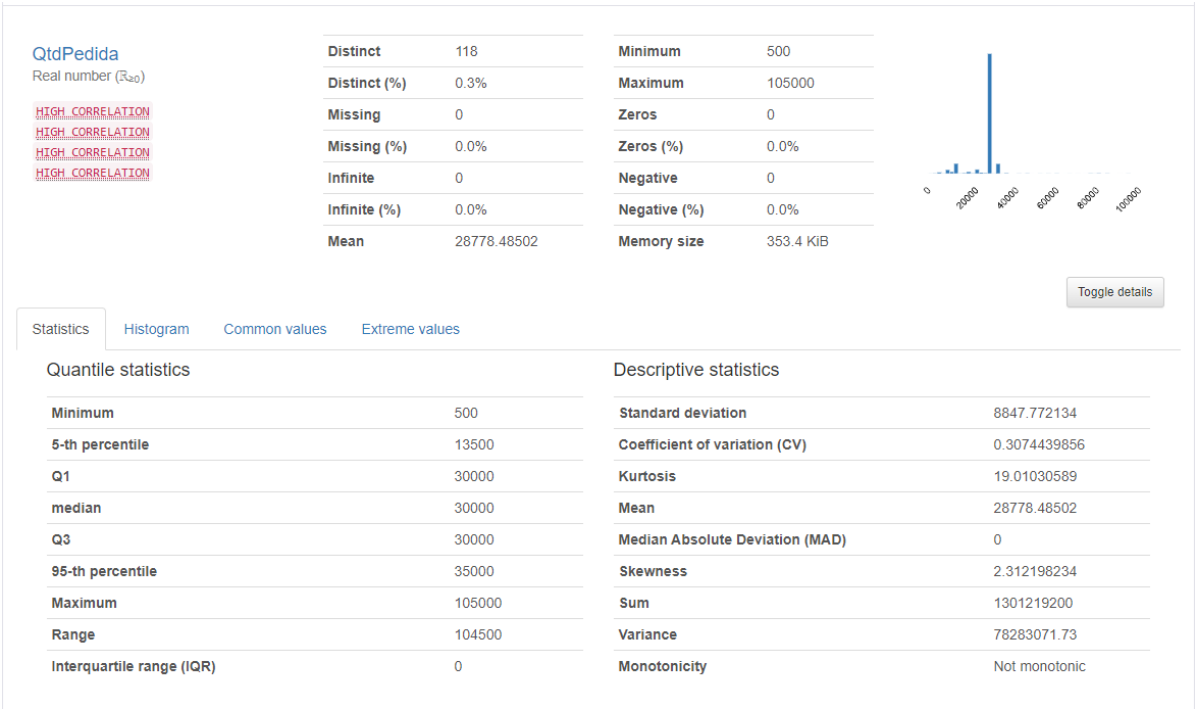

Figura 47 - Análise Qualitativa da Variável QtdPedida

Dataentrada: DateTime

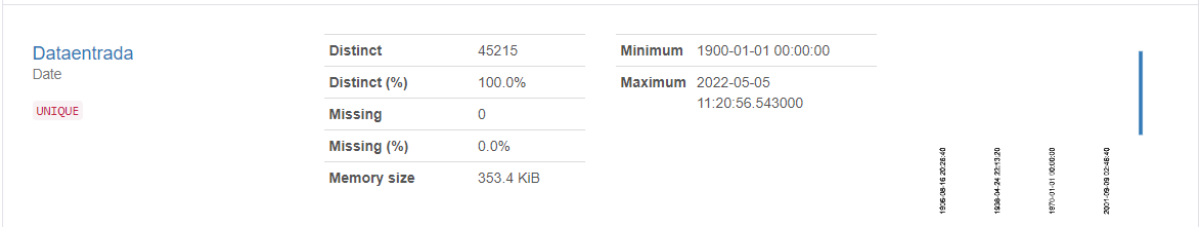

Figura 48 - Análise Qualitativa da Variável Dataentrada.

DataInicioOperacao: DateTime

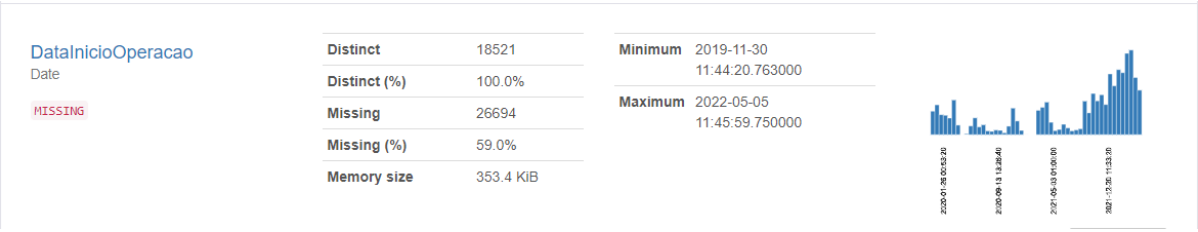

Figura 49 - Análise Qualitativa da Variável DataInicioOperacao.

precDiff: Numerical

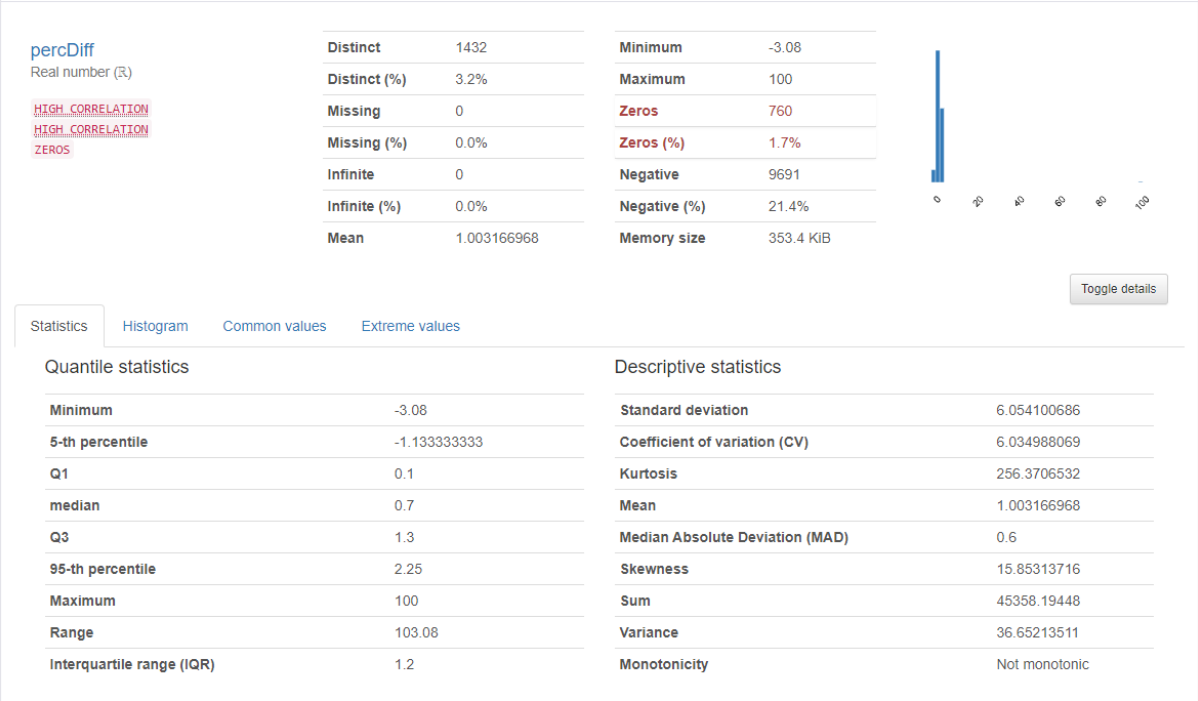

Figura 50 - Análise Qualitativa da Variável percDiff.

DataFimOperacao: DateTime

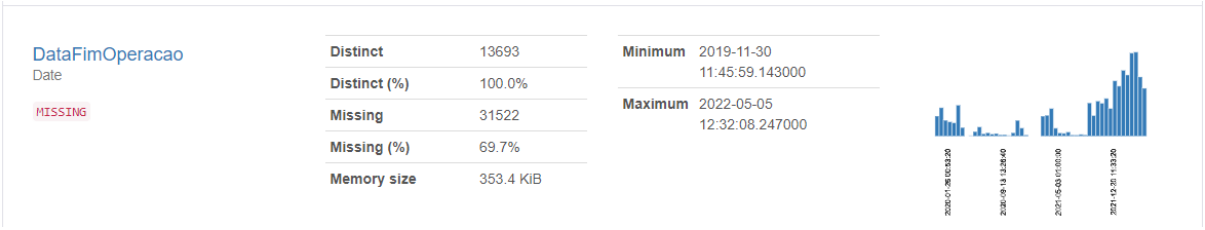

Figura 51 - Análise Qualitativa da Variável DataFimOperacao.

BrutoData: DateTime

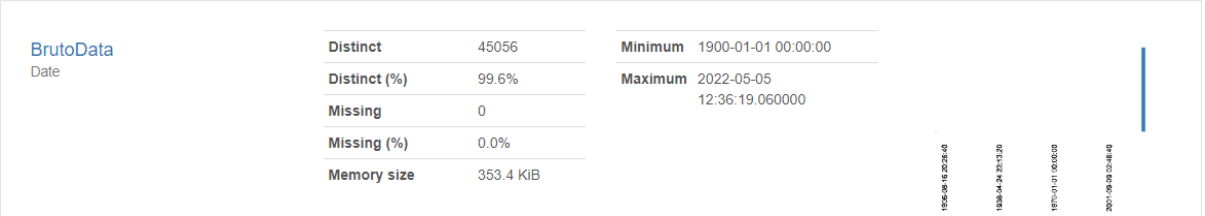

Figura 52 - Análise Qualitativa da Variável BrutoData.

DataFecho: DateTime

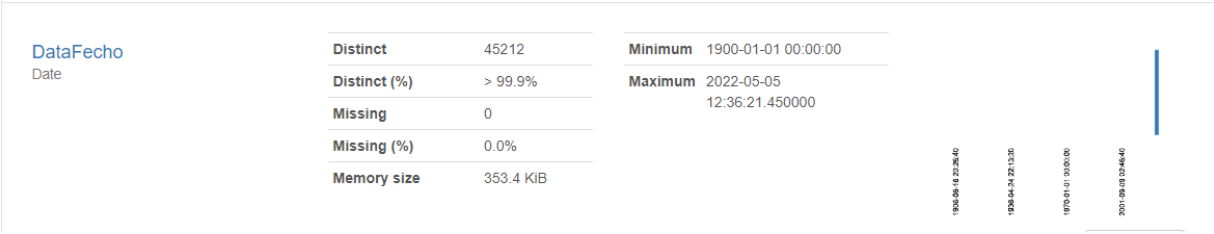

Figura 53 - Análise Qualitativa da Variável DataFecho.

Correlação das Variáveis

Spearman Correlation

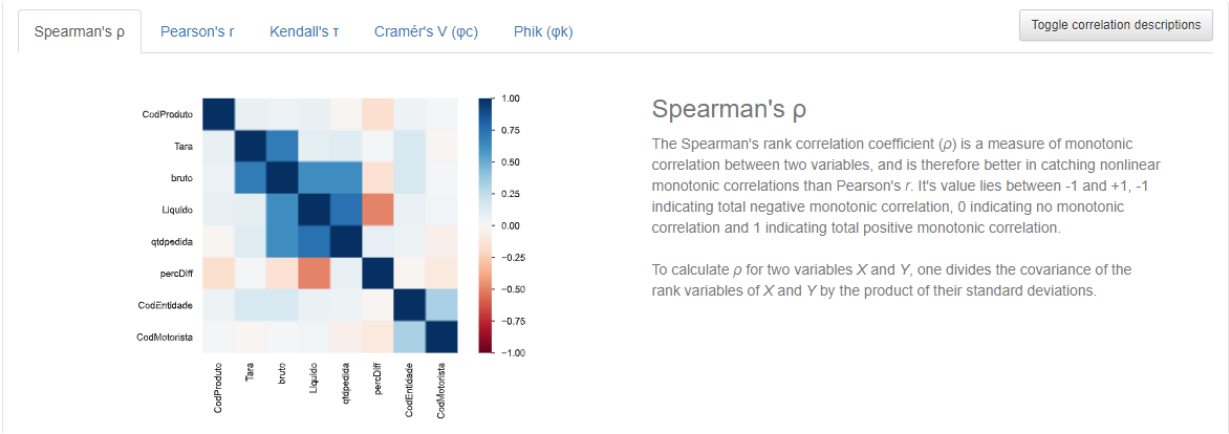

Figura 54 - Análise Qualitativa através da Spearman Correlation.

Pearson Correlation

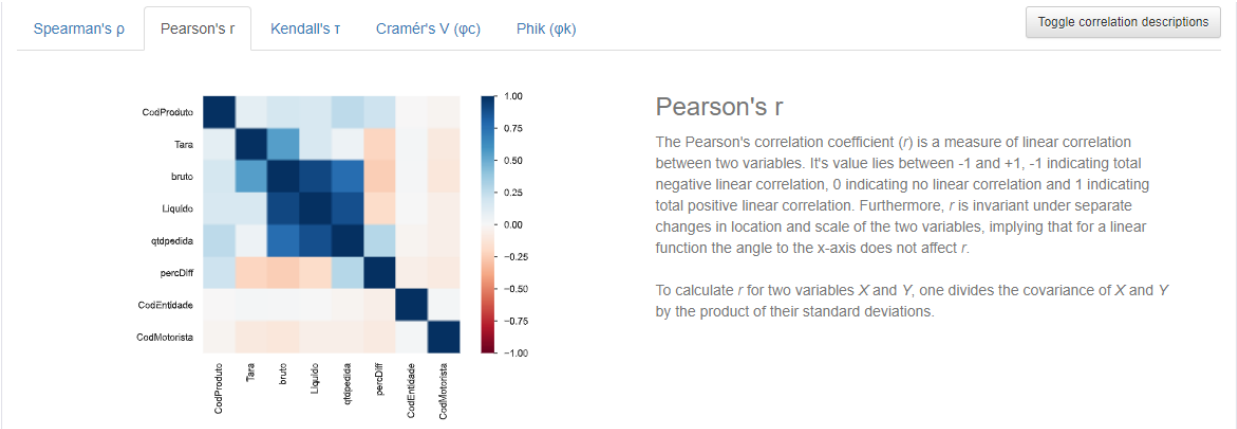

Figura 55 - Análise Qualitativa através da Pearson Correlation.

Valores Nulos

Contador da não existência de valores Nulos

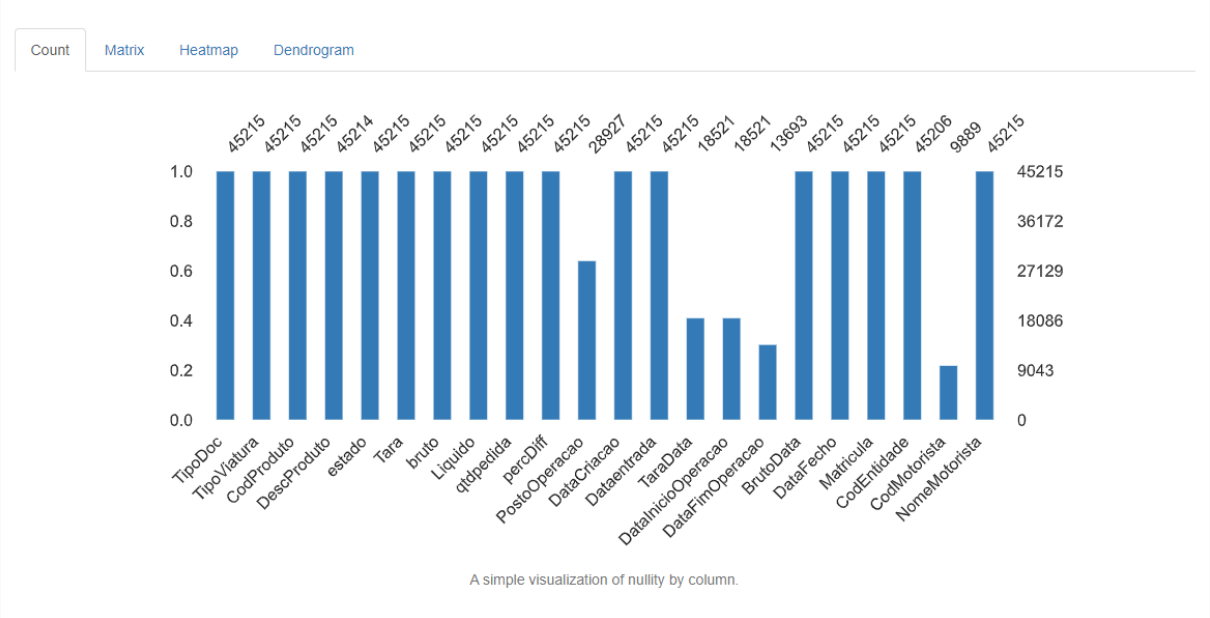

Figura 56 - Análise Qualitativa de valores nulos.

Heatmap

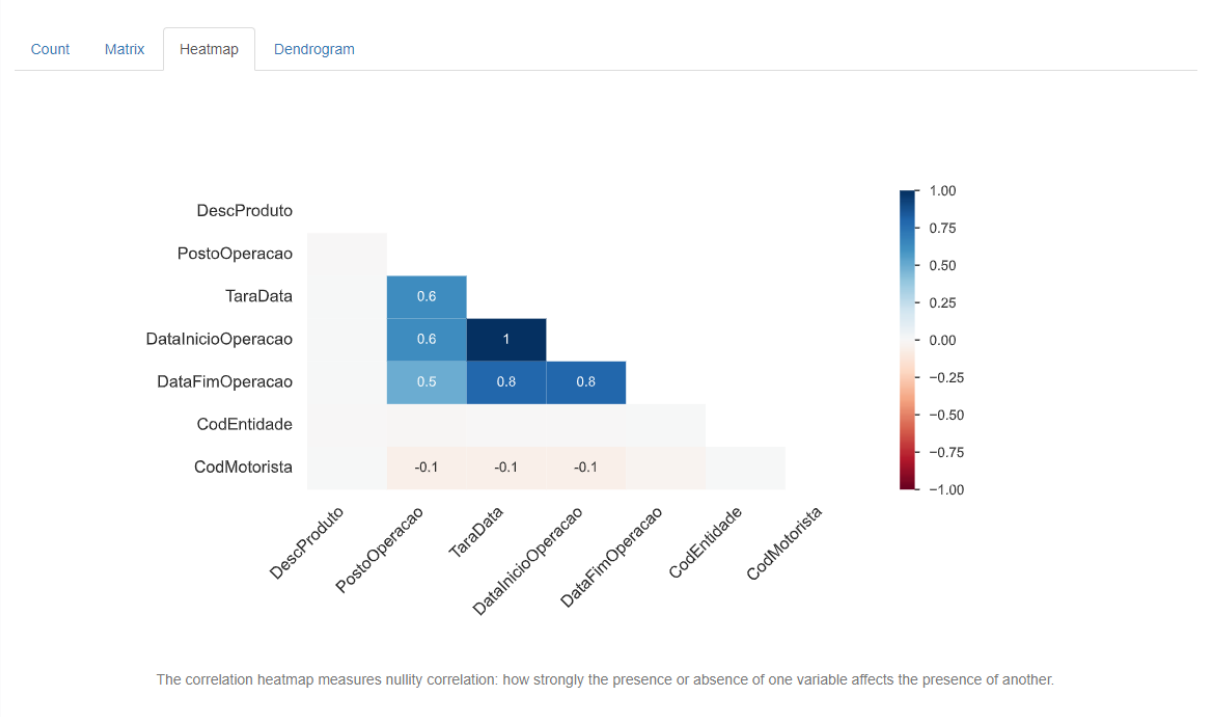

Figura 57 - Análise Qualitativa através do Heatmap de valores nulos.

APÊNDICE II – ANÁLISE EXPLORATÓRIA DOS DADOS

Desvios ao longo das Horas

A Figura 58 ilustra o total em termos de contagem, a média, a mediana e o desvio padrão da variável percDiff ao longo das horas do dia. O objetivo desta observação é identificar períodos horários que apresentem valores de desvio superiores aos restantes. A análise dos gráficos permite visualizar uma tendência de descida do desvio ao longo das horas do dia, sendo o valor mais baixo apresentado por volta das 20 horas.

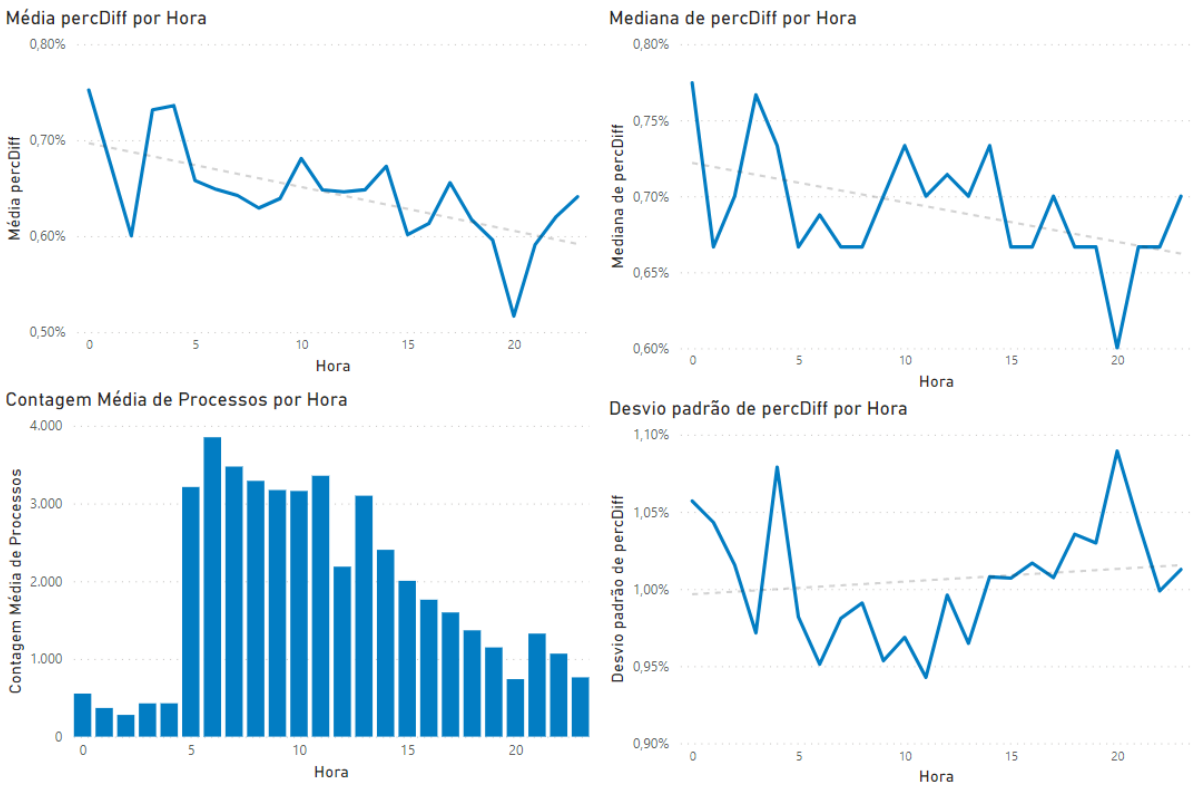

Figura 58 - Desvios ao longo das horas do dia.

Desvios ao longo dos Dias do Mês

A Figura 59 ilustra o total em termos de contagem, a média, a mediana e o desvio padrão da variável percDiff ao longo dos dias do mês. Esta análise foi desenvolvida de forma a verificar a existência ou não de possíveis padrões relacionados com o desvio e o seu comportamento ao longo dos dias do mês. Uma observação atenta dos gráficos desenvolvidos permite concluir que existe uma tendência de descida do percDiff nos primeiros 15 dias do mês, seguido de uma subida nos restantes 15.

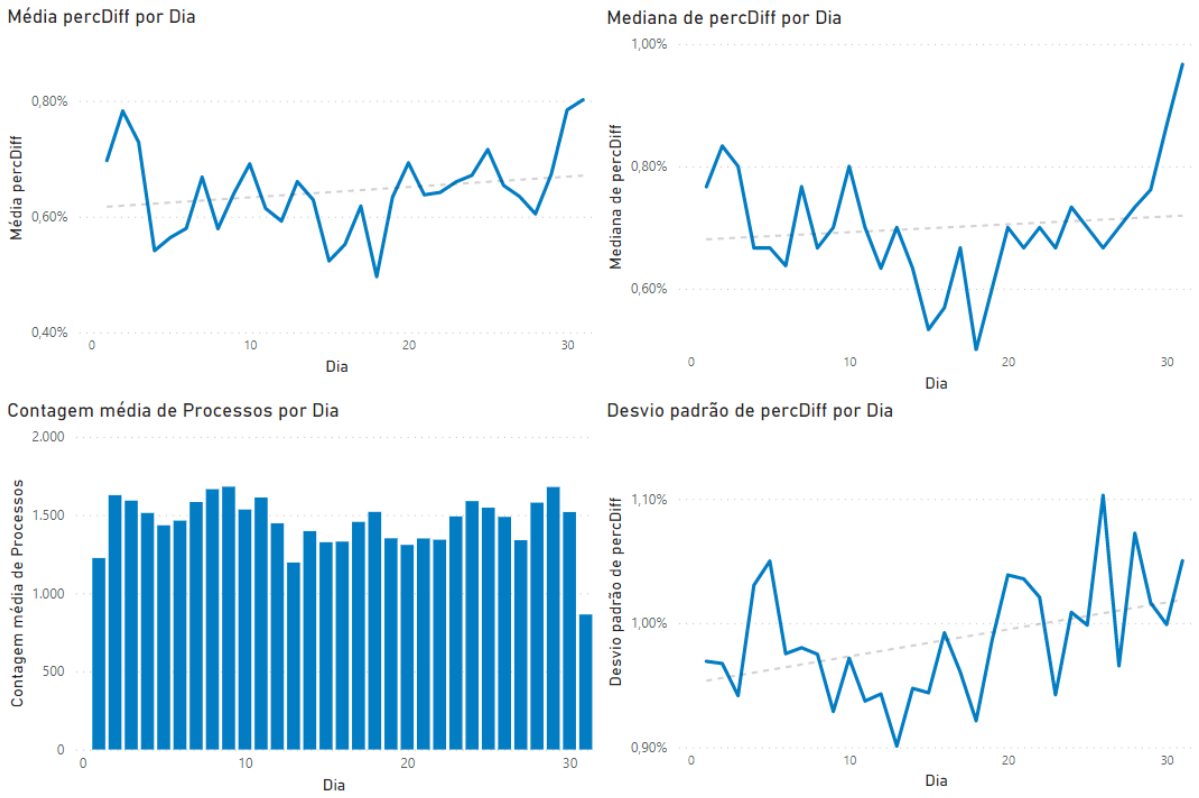

Figura 59 - Desvios ao longo dos dias do mês.

Desvios ao longo dos Meses

A Figura 60 foi concebida com o objetivo de proporcionar uma visão da contagem, da média, da mediana e do desvio padrão da variável percDiff ao longo dos meses. Uma observação atenta dos gráficos elaborados permite concluir que existe uma tendência de aumento da variável percDiff ao longo dos meses.

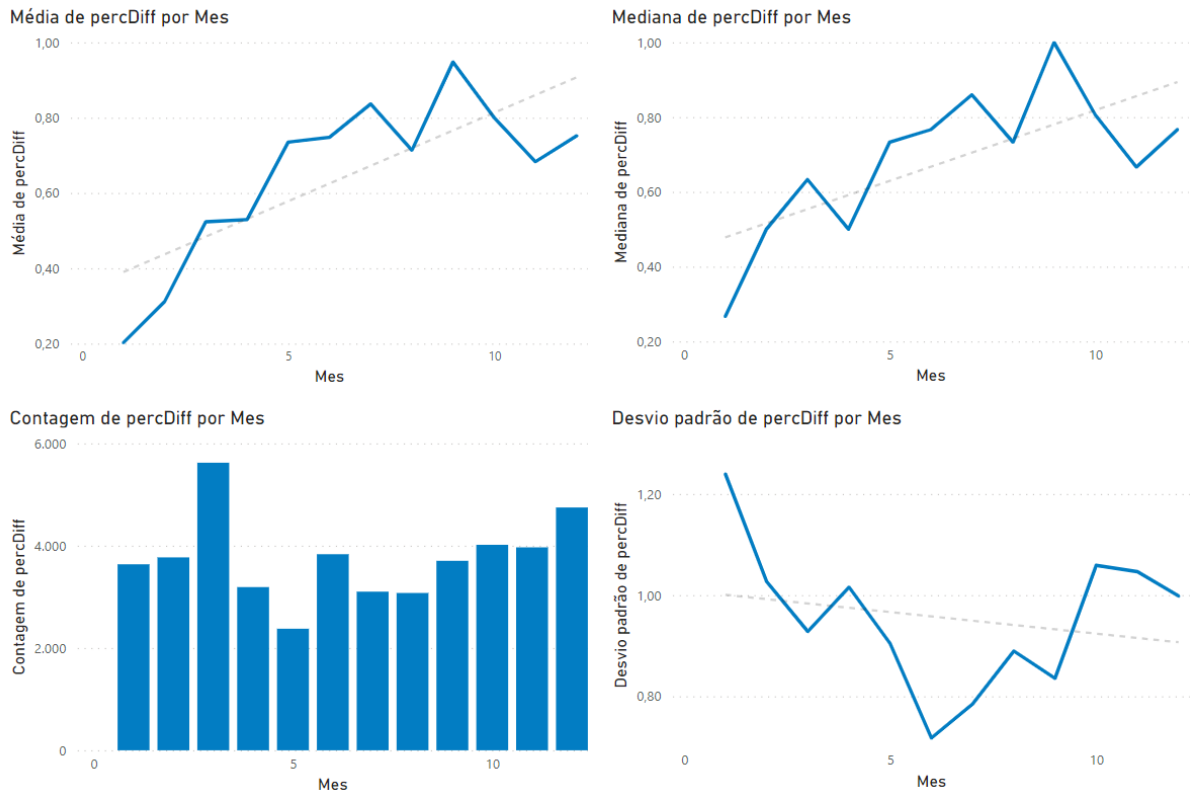

Figura 60 - Desvios ao longo dos meses.

Desvios Alarmísticos ao longo das Horas

Na Figura 61 são ilustrados os desvios alarmísticos (percDiff <= -2 U percDiff >= 2) em termos de contagem, da média, da mediana e do desvio padrão da variável percDiff ao longo das horas do dia. Uma observação completa do gráfico permite concluir que ao longo do dia existe uma tendência de descida dos valores alarmísticos da variável percDiff.

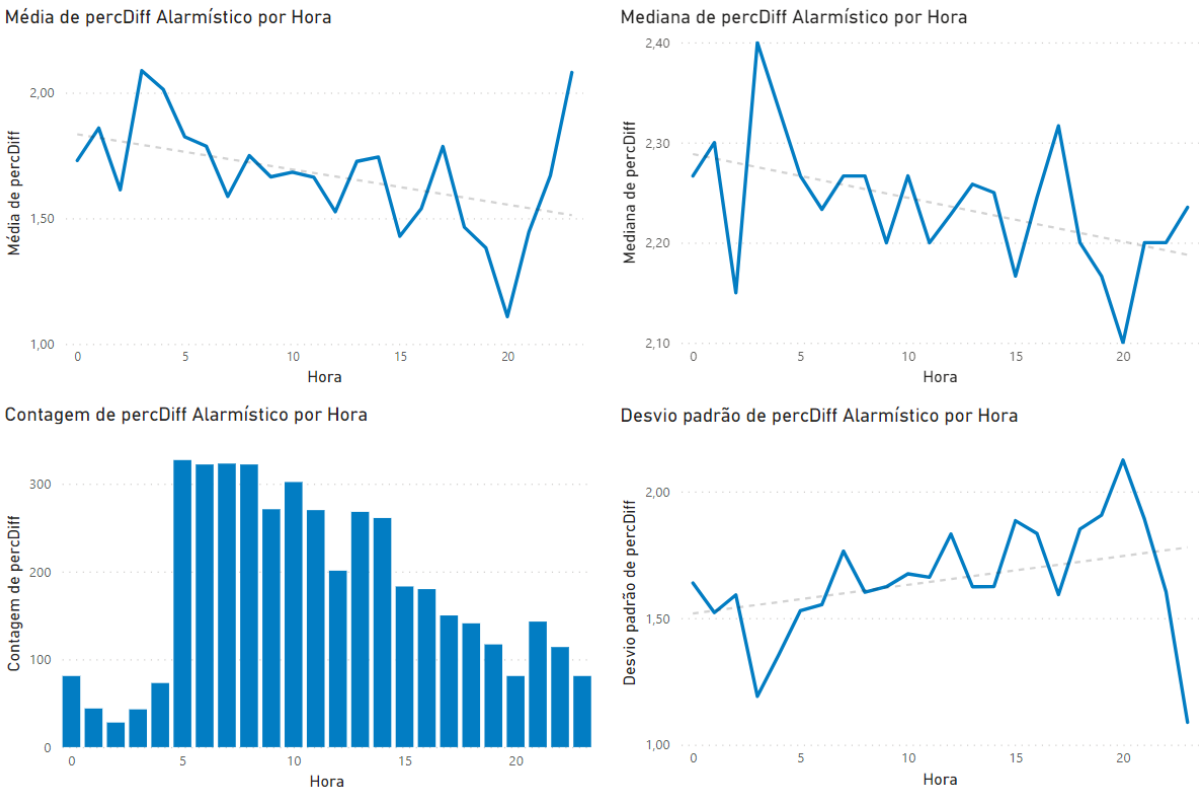

Figura 61 - Desvio alarmístico ao longo das horas.

Desvios Alarmísticos ao longo dos Dias

A Figura 62 ilustra os desvios alarmísticos (percDiff <= -2 U percDiff >= 2) em termos de contagem, da média, da mediana e do desvio padrão da variável percDiff ao longo dos dias de um mês. No que se refere às variações do percDiff em termos da média, da mediana e do desvio padrão, não é possível tirar qualquer conclusão objetiva, uma vez que as variações apresentam grandes variações ao longo dos dias. Contudo, uma análise da componente da contagem média de processos permite averiguar que, por volta dos dias que se encontram a meio do mês, existem menos processos a ser realizados que nas restantes alturas.

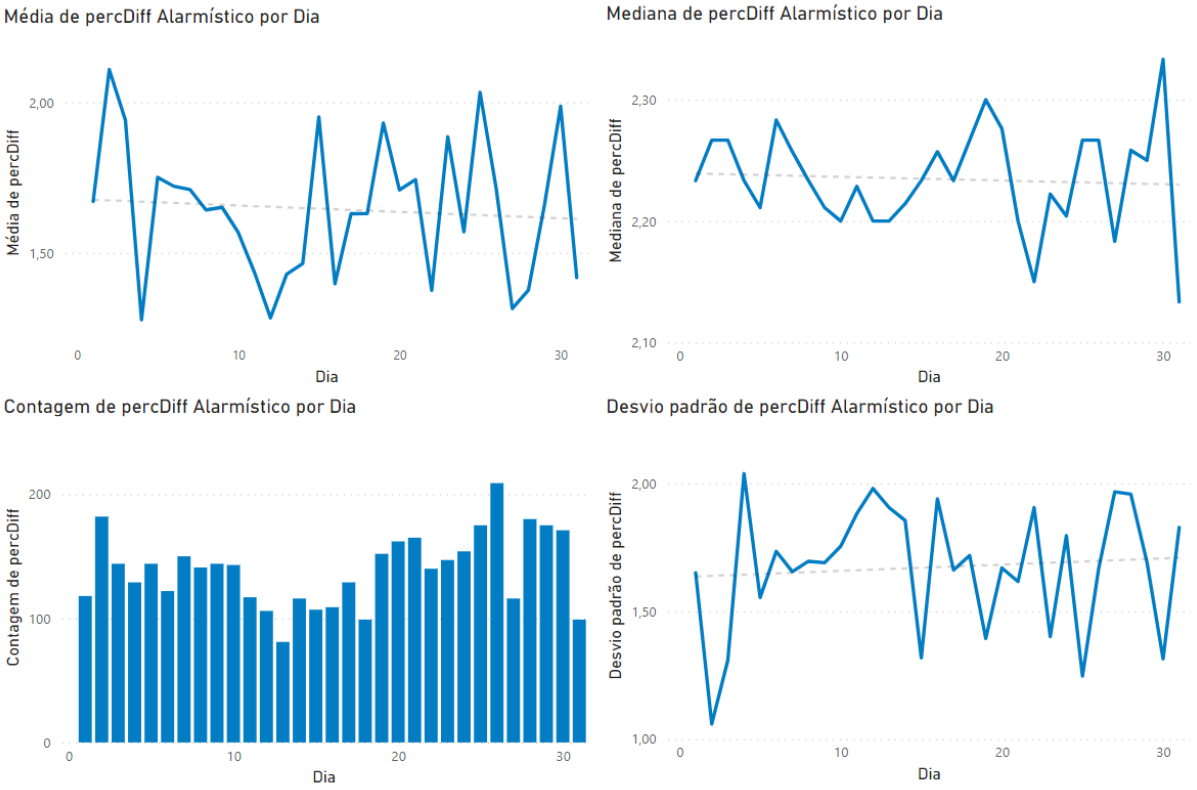

Figura 62 - Desvio alarmístico ao longo dos dias.

Desvios Alarmísticos ao longo dos Meses

A Figura 63 demonstra os desvios alarmísticos (percDiff <= -2 U percDiff >= 2) em termos de contagem, da média, da mediana e do desvio padrão da variável percDiff ao longo dos meses. Nos gráficos desenvolvidos é possível observar uma tendência de subida do percentual do percDiff Alarmístico ao longo dos meses do ano. Além disso, uma atenta observação em relação à média dos processos realizados permite concluir que os meses que representam o meio do ano são aqueles com menos processos alarmísticos a ocorrer, ao contrário do que acontece com o final e primeiro mês do ano.

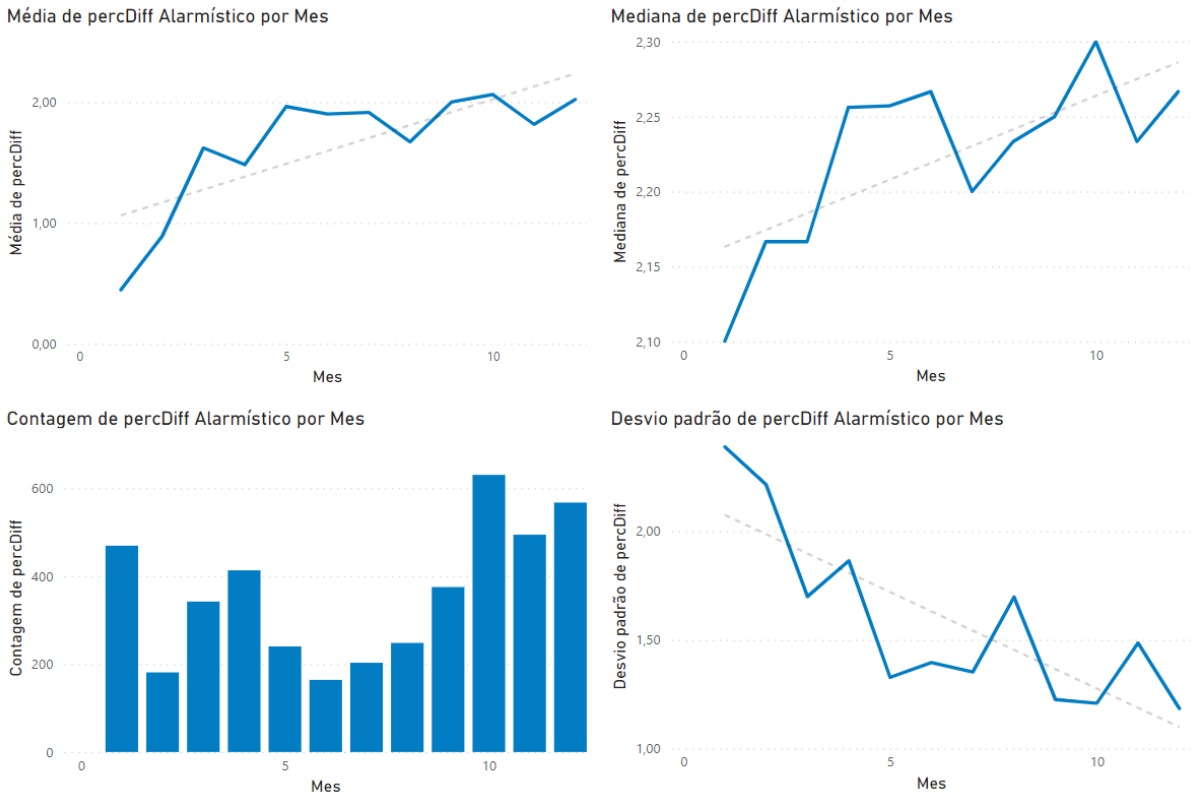

Figura 63 - Desvio alarmístico ao longo dos mês.

Desvios por Posto de Operação

A Figura 64 mostra o total de desvios realacionado com cada posto de operação. É importante ter em atenção que a maioria dos carregamentos de sacos de cimento são realizados no posto "Unknown Station". Contudo, embora o posto anteriormente referido seja aquele onde mais carregamentos são realizados, o posto que apresenta um maior desvio percentual no peso é o posto "ENV2".

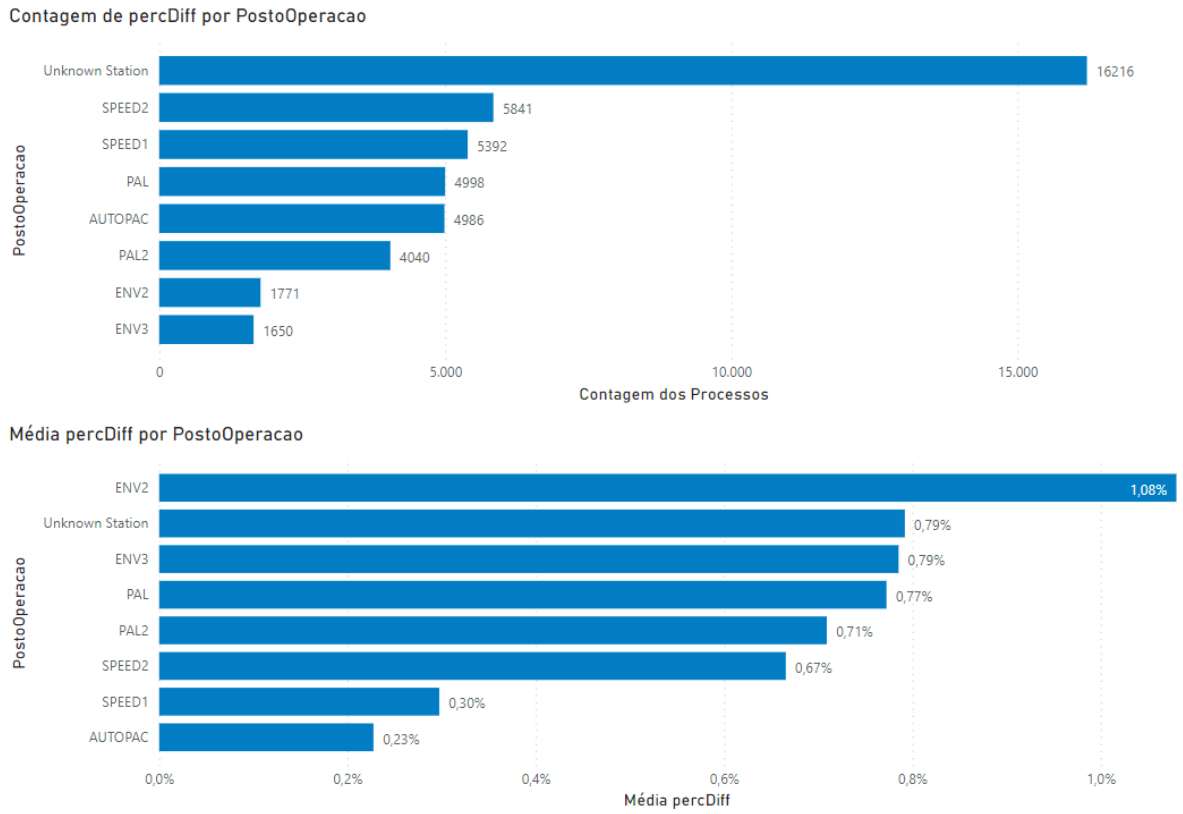

Figura 64 - percDiff por PostoOperacao.

Desvios Alarmísticos por Posto de Operação

Na Figura 65, é possivel visualizar os desvios alarmísticos (percDiff <= -2 U percDiff >= 2) por posto de operação. Tal como na Figura 64, o posto denominado de "Unknown Station" é aquele que apresenta uma maior ocorrência de processos alarmísticos, contudo não é o que apresenta uma média de desvios alarmisticos maior, esse registo pertence à estação ENV2

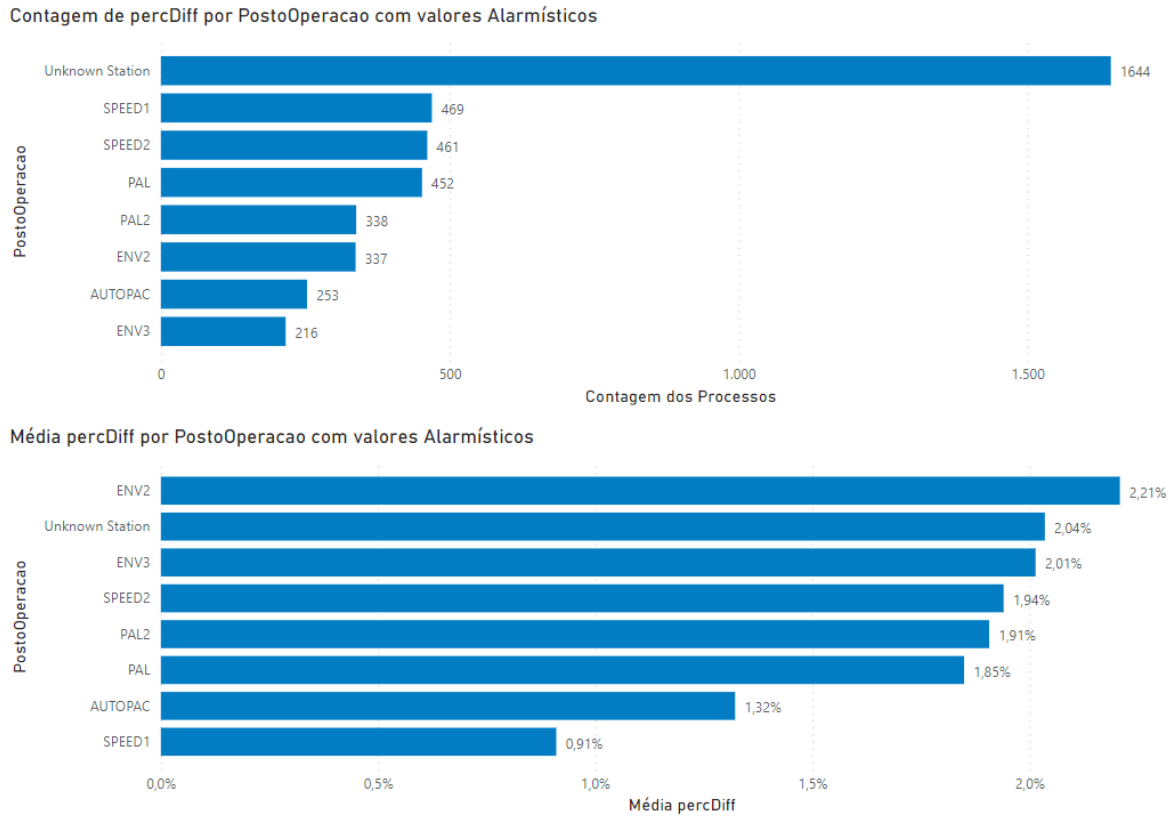

Figura 65 - percDiff com valores Alarmísticos por PostoOperacao.

Média do desvio ao longo das horas do dia por Tipo de Veículo e por Posto de Operação

A Figura 66 ilustra a média dos desvios ao longo do dia pelas variáveis “TipoVeiculo” e “PostoOperacao”.

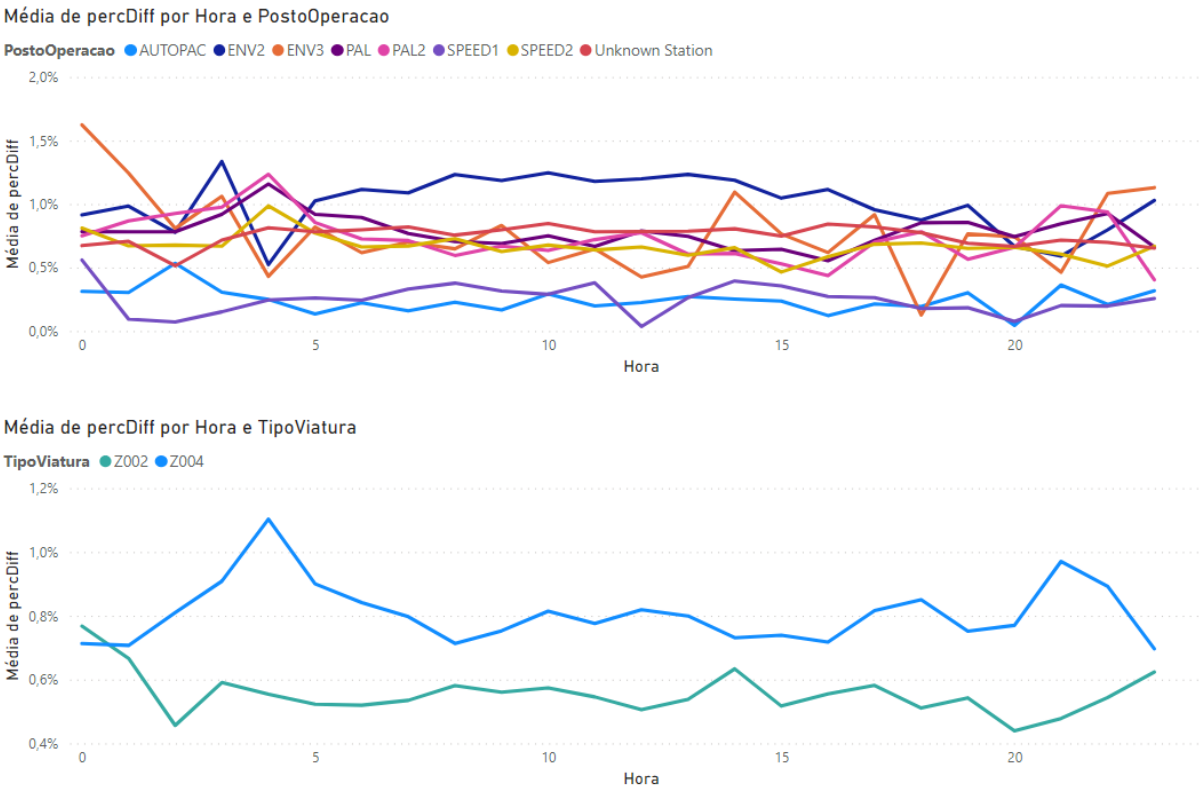

Figura 66 - Média de Desvio ao longo do dia por TipoVeiculo e PostoOperacao.

Média do desvio Alarmístico ao longo das horas do dia por Tipo de Veículo e por Posto de Operação

A Figura 67 ilustra os desvios alarmísticos (percDiff <= -2 U percDiff >= 2) por "TipoVeiculo" e por "PostoOperacao" ao longo das horas de um dia. É importante ter em atenção que o veículo Z002 é o tipo de veículo que apresenta o maior desvio em termos de processos alarmísticos.

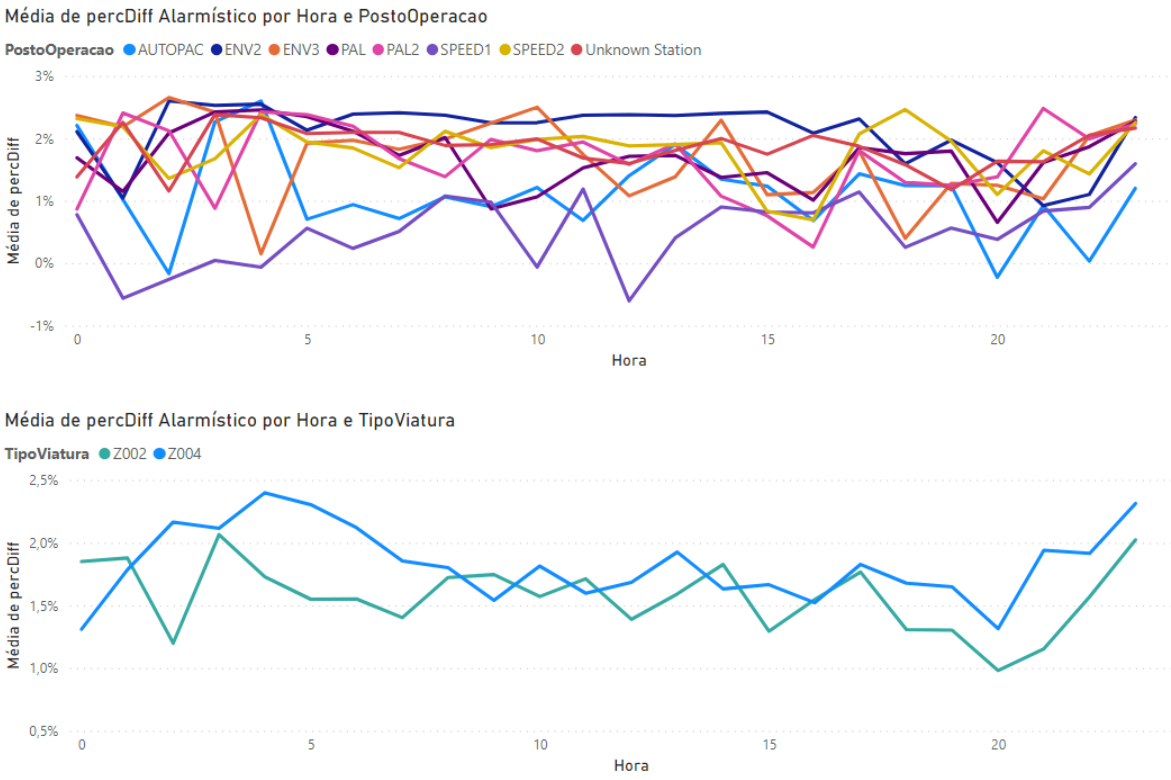

Figura 67 - Média de Desvio Alarmístico ao longo do dia por TipoVeiculo e PostoOperacao.

Média do desvio Alarmístico ao longo dos dias do mês por Tipo de Veículo e por Posto de Operação

A Figura 68 demonstra as variações alarmísticas ocorridas em média ao longo dos dias que representam um mês, para o "TipoVeiculo" e "PostoOperacao".

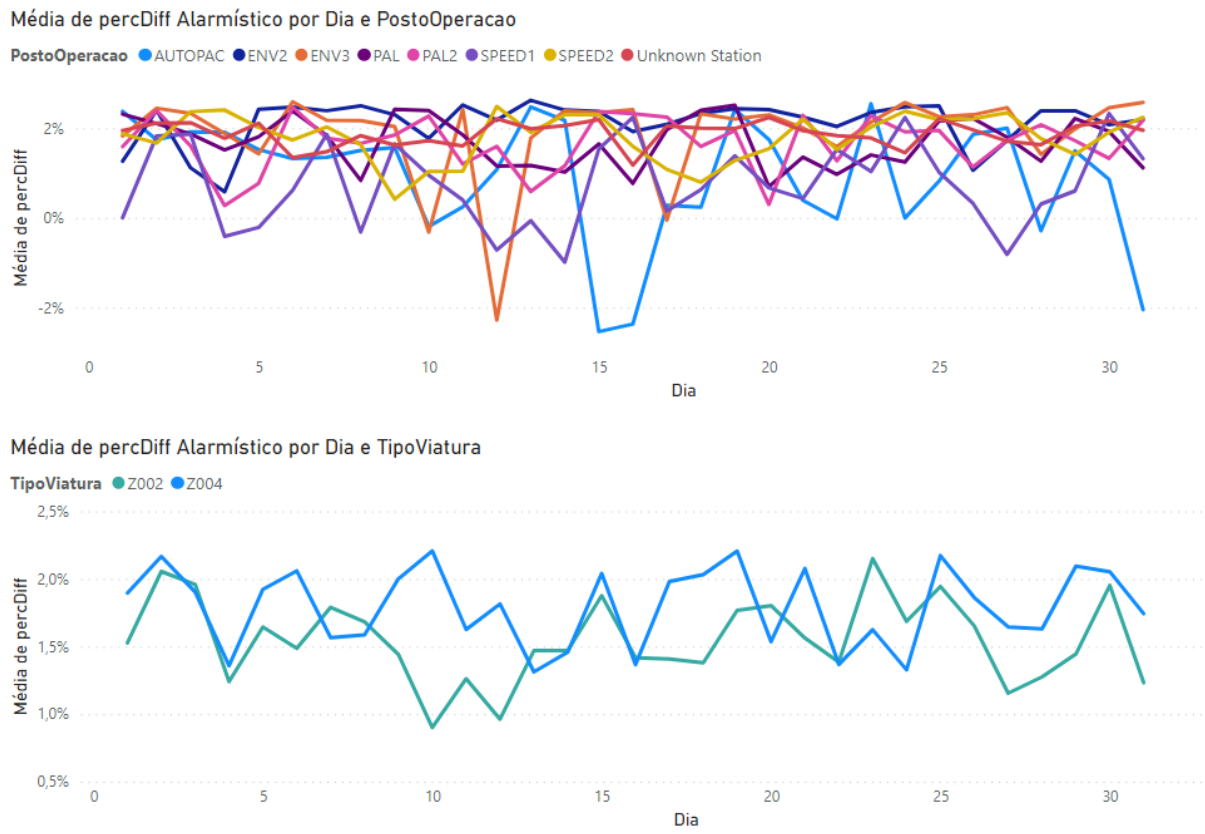

Figura 68 - Média de Desvio Alarmístico ao longo do mês por TipoVeiculo e PostoOperacao.

Contagem de Desvios por Posto de Operação e por Tipo de Veículo ao longo do tempo (Horas do Dia)

A Figura 69 mostra o total de desvios ocorridos por posto de operação e por tipo de veículo ao longo das várias horas do dia. Como abordado anteriormente, a maioria dos processos realizados não tem especificado, definido ou registado o posto de operação ("Unknown Station") e o tipo de veículo Z002, é aquele que performa um maior número de operações.

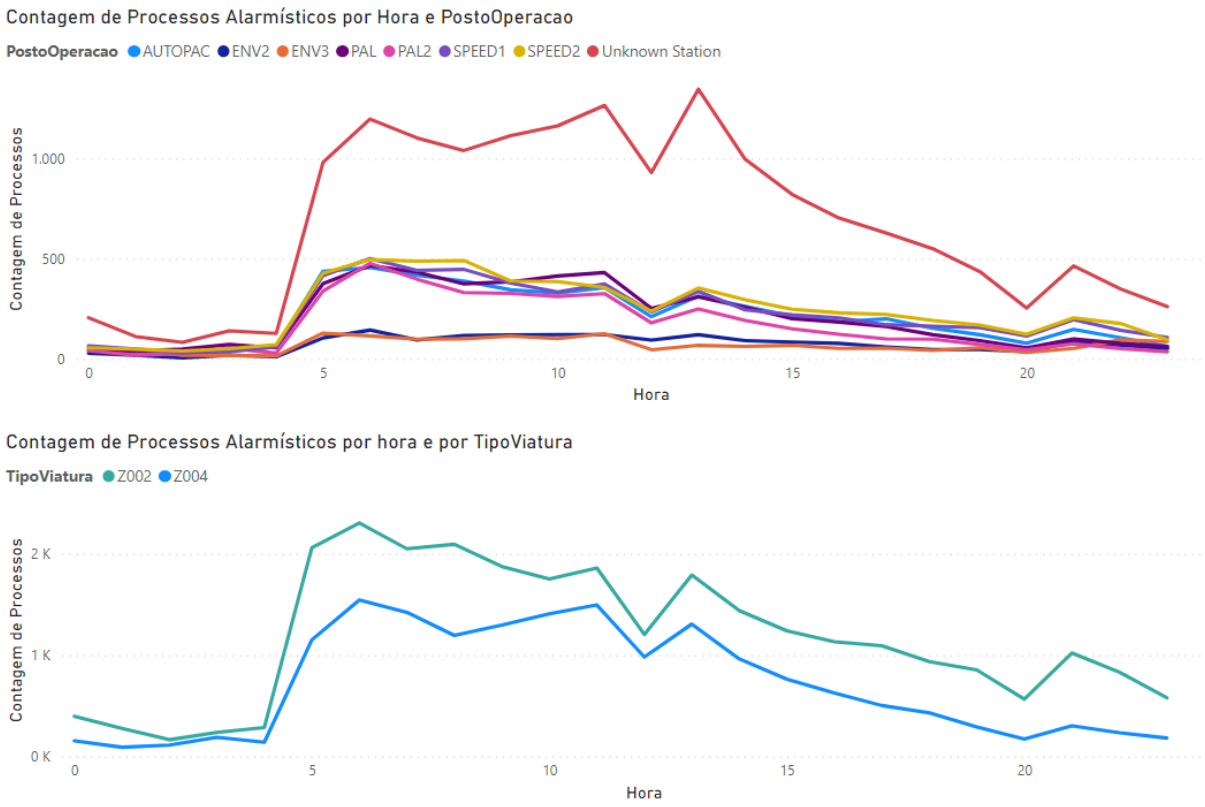

Figura 69 - Contagem de Processos por Hora e PostoOperacao / TipoViatura.

Contagem de Desvios Alarmísticos por Posto de Operação e por Tipo de Veículo ao longo do tempo (Horas do Dia)

A Figura 70 ilustra os desvios alarmísticos (percDiff <= -2 U percDiff >= 2) por posto de operação e por tipo de veiculo ao longo das várias horas que compõem um dia. A análise permite concluir que a maioria dos desvios ocorre no posto de operação "Unknown Station" e, dos dois tipos de veículos existentes, Z002 é o que apresenta um maior número de desvios.

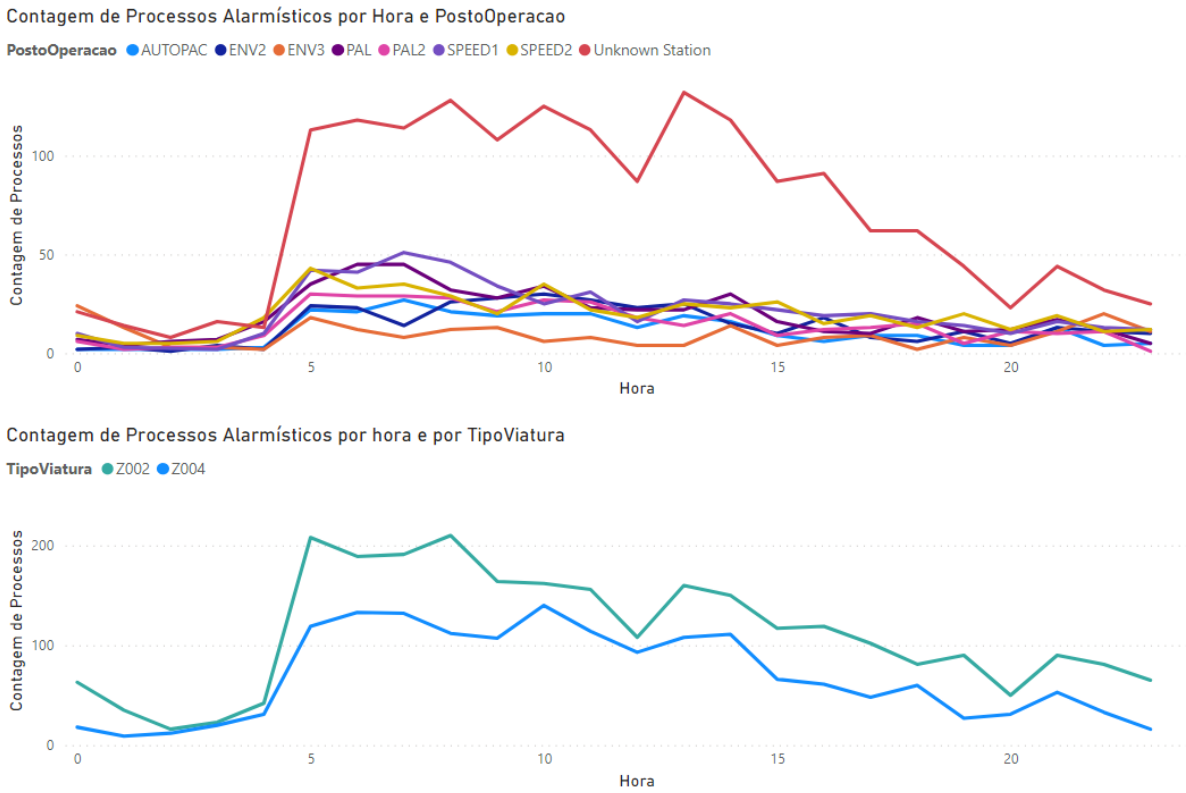

Figura 70 - Contagem de Processos Alarmísticos por Hora e PostoOperacao / TipoViatura.

Top 20 de veículos que apresentam mais desvios alarmísticos

A Figura 71 demonstra o top 20 de veículos com o registo de processos alarmístico o mais elevado

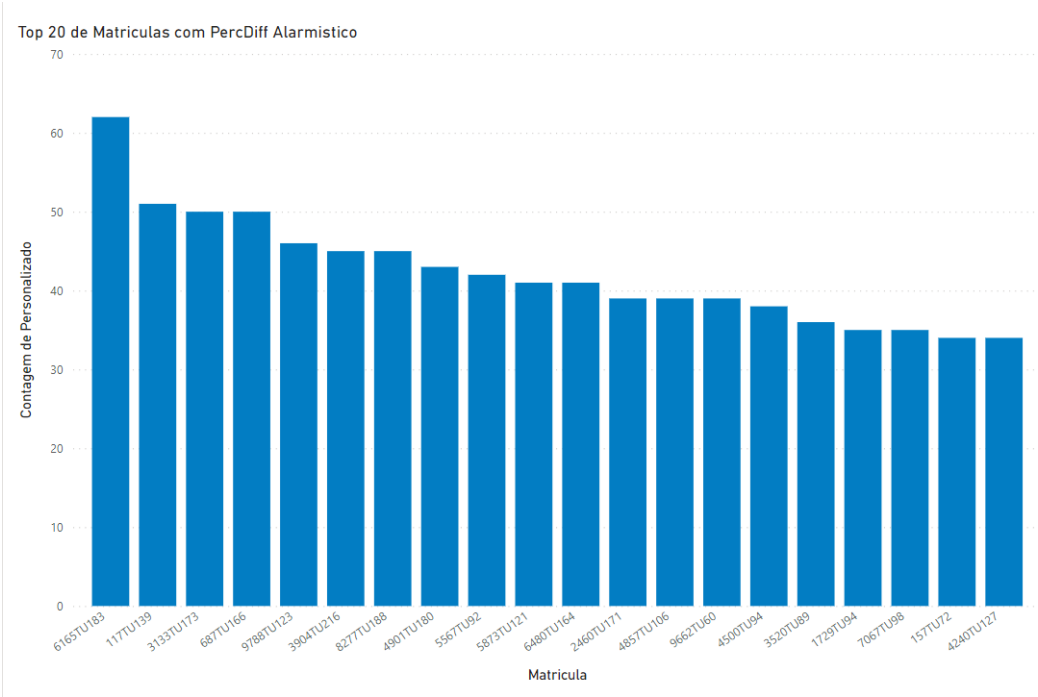

Figura 71 - Ranking de veículos com desvios alarmísticos.

Desvio apresentado ao longo dos processos realizados

A Figura 72 mostra as várias variações ocorridas ao longo do tempo.

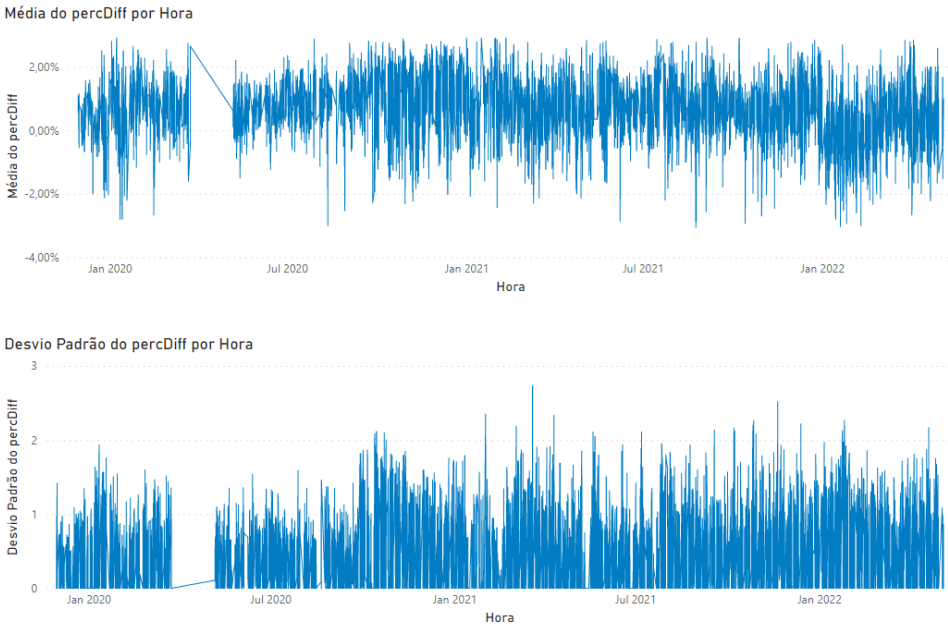

Figura 72 - Média e Desvio Padrão do percDiff por Hora.

APÊNDICE III – VISUALIZAÇÕES *SHAP*

As figuras ilustradas no Apêndice III permitem observar e compreender a forma como os vários valores de cada um dos atributos utilizados como *input* para o modelo *Random Forest* afetam a previsão realizada. Assim sendo, o gráfico presente em cada uma das figuras é constituído por diversos pontos e cada um deles representa o valor do atributo numa dada linha do *dataset.* Além disso, a cor que constitui o ponto representa a forma como os valores do atributo impactaram ou não o aumento da probabilidade no que se refere à classe 0. Os gráficos constituintes das figuras demonstram também qual é o atributo, de entre os existentes, que mais interage com o atributo em análise, de forma a elevar ou a diminuir o valor da probabilidade referente à classe em análise.

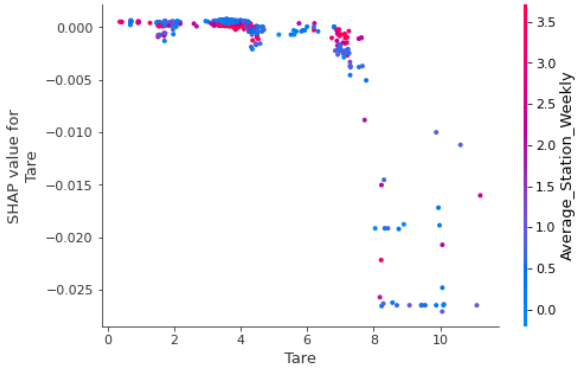

Figura 73 – Dependence Plot da tara do veículo (Tare) e a média de desvio semanal nas estações (Average_Station_Weekly)

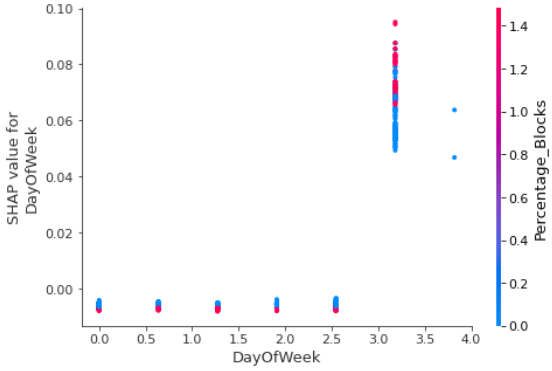

Figura 75 - Dependence Plot dos dias da semana (DayOfWeek) e a percentagem de bloqueios de um dado veículo (Percentage_Blocks)

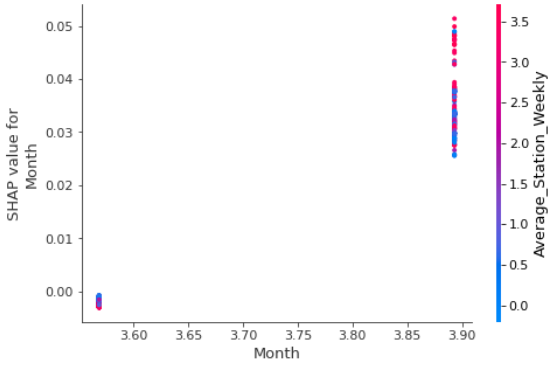

Figura 74 - Dependence Plot do mês em que se realiza o processo (Month) e a média de desvio semanal nas estações (Average_Station_Weekly)

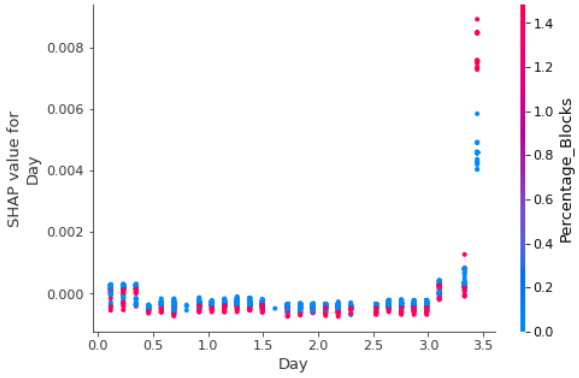

Figura 76 – Dependence Plot do dia em que se realiza o processo (Day) e a percentagem de bloqueios de um dado veículo (Percentage_Blocks)

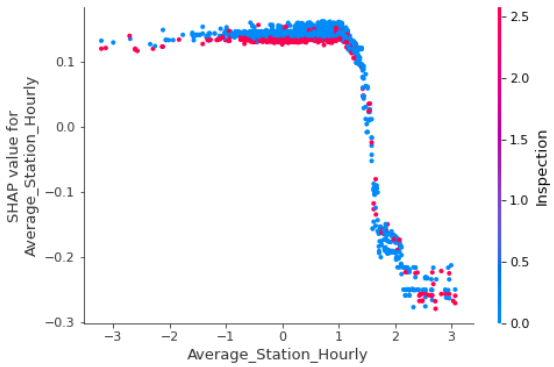

Figura 77 - Dependence Plot do período de inspeção no processo (Inspection) e a média de desvio da última hora nas estações (Average_Station_Hourly)

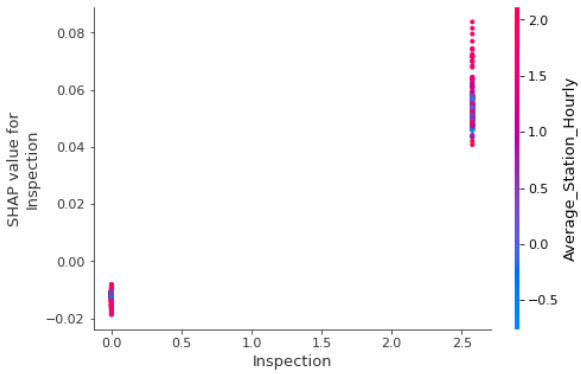

Figura 80 - Dependence Plot da média de desvio da última hora nas estações (Average_Station_Hourly) e o período de inspeção no processo (Inspection)

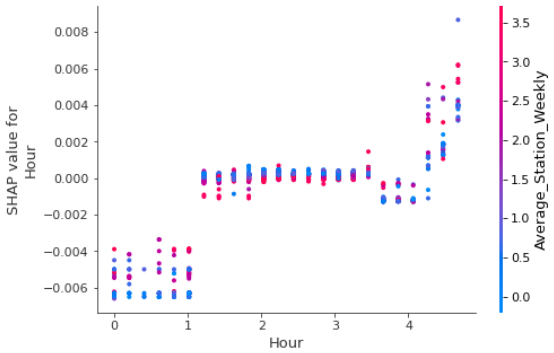

Figura 78 - Dependence Plot da hora em que se realizou o processo (Hour) e a média de desvio semanal nas estações (Average_Station_Weekly)

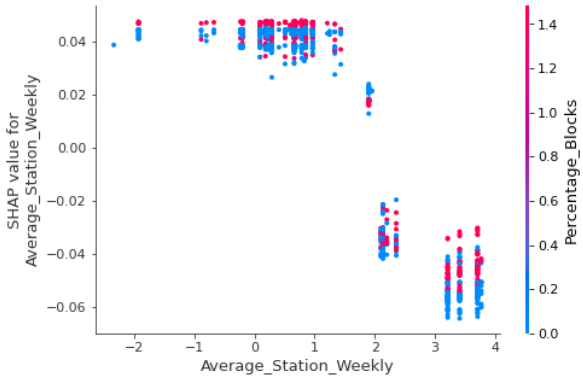

Figura 81 - Dependence Plot da média de desvio semanal nas estações (Average_Station_Weekly) e a percentagem de bloqueios de um dado veículo (Percentage_Blocks)

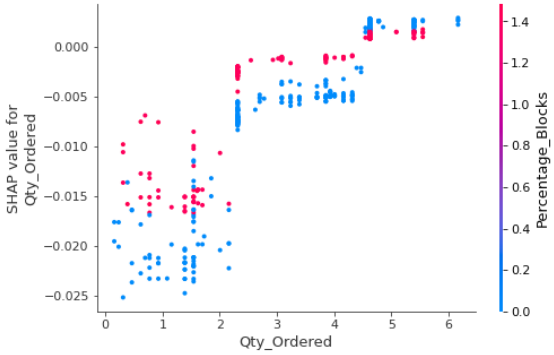

Figura 79 - Dependence Plot da quantidade solicitada (Qty_Ordered) e a percentagem de bloqueios de um dado veículo (Percentage_Blocks)

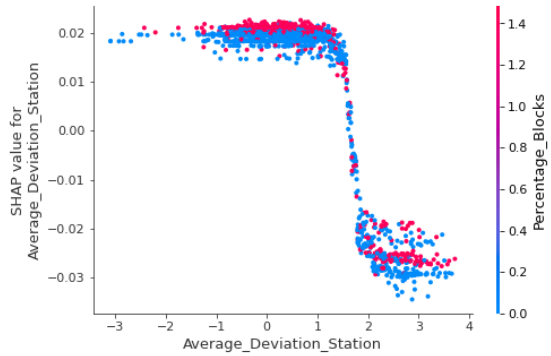

Figura 82 - Dependence Plot da média de desvio nas estações (Average_Deviation_Station) e a percentagem de bloqueios de um dado veículo (Percentage_Blocks)

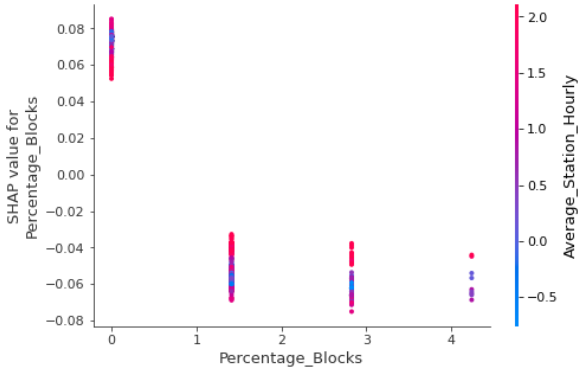

Figura 83 - Dependence Plot da percentagem de bloqueios de um dado veículo (Percentage_Blocks) e a média de desvio da última hora nas estações (Average_Station_Hourly)