Trabalho Prático - Machine Learning - Primeira Entrega

Bruno Rodrigues Faria
Pontifícia Universidade Católica De Minas Gerais
Belo Horizonte, Minas Gerais, Brasil

Guilherme Dantas Caldeira Fagundes
Pontifícia Universidade Católica De Minas Gerais
Belo Horizonte, Minas Gerais, Brasil

Laura Iara Silva Santos Xavier
Pontifícia Universidade Católica De Minas Gerais
Belo Horizonte, Minas Gerais, Brasil

Maria Luisa Tomich Raso
Pontifícia Universidade Católica De Minas Gerais
Belo Horizonte, Minas Gerais, Brasil

Matheus Rangel de Figueiredo
Pontifícia Universidade Católica De Minas Gerais
Belo Horizonte, Minas Gerais, Brasil

ABSTRACT

Este artigo irá tratar sobre a escolha do Dataset e suas mais características, além de mostrar resultados de uma aplicação de um algoritmo de Machine Learning. Que foi aplicado sem as devidas etapas de pré-processamento antes. Trazendo assim os resultados a serem analisados pelos estudantes e tratados na próxima etapa de desenvolvimento do trabalho.

CCS CONCEPTS

• **Computing methodologies** → **Machine learning approaches**.

KEYWORDS

dataset, atributos, supervisionado, machine learning, detecção de incêndio

ACM Reference Format:
Bruno Rodrigues Faria, Guilherme Dantas Caldeira Fagundes, Laura Iara Silva Santos Xavier, Maria Luisa Tomich Raso, and Matheus Rangel de Figueiredo. 2018. Trabalho Prático - Machine Learning - Primeira Entrega. In *Woodstock '18: ACM Symposium on Neural Gaze Detection, June 03–05, 2018, Woodstock, NY*. ACM, New York, NY, USA, 3 pages. https://doi.org/10.1145/1122445.1122456

PUC MINAS , September 17–10, 2022, Belo Horizonte, MG

ACM ISBN 978-1-4503-9999-9/18/06…$15.00
https://doi.org/10.1145/1122445.1122456

1 INTRODUÇÃO

O objetivo deste trabalho é analisar o dataset Smoke Detection Dataset retirado do Kaggle, e mostrar suas características através da aplicação de um algoritmo de aprendizado de máquina.

O arquivo apresenta cerca de 60.000 instâncias com dados de diversos ambientes passíveis de incêndio, viabilizando um diagnóstico sobre a a existência ou não de fumaça no local. Os resultados obtidos, podem ser utilizados no funcionamento de detectores de fumaça que captam essas variáveis.

2 DESCRIÇÃO DA BASE

O dataset é composto por dados numéricos distribuídos em 13 colunas, dentre elas, o atributo binário de classificação "Fire Alarm" reperesentando a presença de fumaça no local, em função dos outros 12 valores.

Cada um desses atributos possui um limite máximo e mínimo, e uma descrição da sua representação dentro do contexto:

- Temperature (°C): representa a temperatura do ar, possui como mínimo -22,01°C e máximo como 59,93°C.
- Humidity (%): representa a porcentagem de humidade do ar, possui como mínimo 10,74% e 75,2% como máximo.
- TVOC (ppb): valor numérico em partes por bilhão que representa o total de compostos orgânicos voláteis. O valor mínimo é 0 e o valor máximo é 60.000.
- eCO2 (ppm): valor numérico em partes por milhão que representa a concentração total de dióxido de carbono no ambiente. O mínimo é 400 e o máximo é 60.000.
- Raw H2 (valor numérico bruto): valor que representa a quantidade de moléculas de H2 no ambiente. O mínimo é 10.668 e o máximo é 13.803.
- Raw Ethanol (valor numérico bruto): valor que representa a quantidade de moléculas de gás etanol no ambiente. O mínimo é de 15.317 e o máximo de 21.410

- Pressure (hPA): valor em hectopascal (100 x pascal) do ambiente. Menor valor é de 930,852 e o maior de 939,861
- PM1.0 (valor numérico bruto): valor que representa a quantidade de partículas com menos de 1µm. Menor valor é 0 e maior é 14.333,69
- PM2.5 (valor numérico bruto): valor que representa a quantidade de partículas com menos de 2.5µm. Menor valor é 0 e maior é 45.432,26
- NC0.5 (valor numérico): valor que representa a concentração de partículas com menos de 0.5µm. Menor valor é 0 e maior é 61.482,03
- NC1.0 (valor numérico): valor que representa a concentração de partículas com menos de 1µm. Menor valor é 0 e maior é 51.914,68
- NC2.5 (valor numérico): valor que representa a concentração de partículas com menos de 2.5µm. Menor valor é 0 e maior é 30.026,438

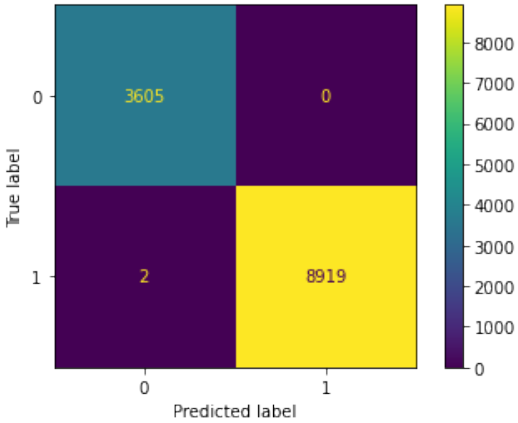
3 CÓDIGO FONTE

O código fonte pode ser obtido através do repositório no GitHub, para todas as etapas do relatório. Onde pode ser visto por etapa realizada ou também pode ser visto um código JupyterNotebook completo , cujo tem todas etapas concatenadas.

Resultados e Métricas

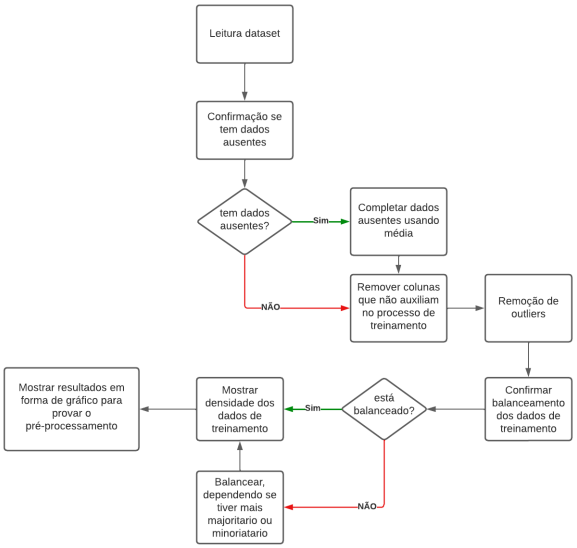
Após a finalização do algoritmo, estes foram os resultados encontrados. A matriz de confusão indicou um ótimo desempenho, ja que apenas 2 de 12.526 instâncias foram classificadas incorretamente. Portanto, as taxas VP e VN foram altíssimas.

As demais métricas também apresentaram resultados acima do esperado. A Precisão foi de 0,99 enquanto o recall, acurácia e F-measure foram 1.



Implementação do Primeiro Tratamento de Dados

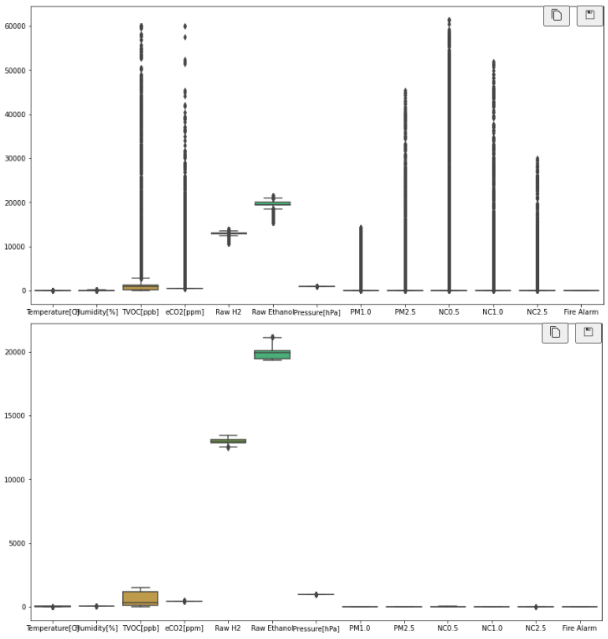
Nessa etapa foram feitos diversos tipos de tratamento como, por exemplo, verificaçào de dados nulos, remoção de outliers, balanceamento de dados e analise da estrutura interna dos dados e a variação dos dados da base através do algoritmo PCA, o passo a passo realizado para a pré-processamento da base pode ser exemplificado pelo seguinte workflow.



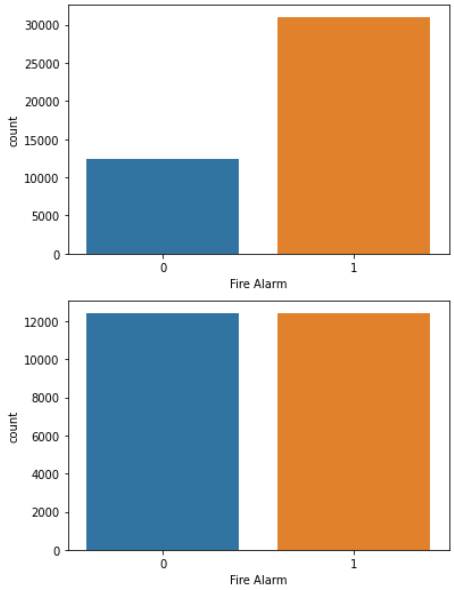
A análise sobre dados nulos e foi percebido que não existe nenhum dado nulo na base, e os tipos de dados presentes na base eram todos númericos não sendo necessário a aplicação de um algoritmo de transformação de dados por conta disso, nenhum tratamento nesse sentido foi necessário fazer.

```
 #   Column           Non-Null Count  Dtype
---  ------           --------------  -----
 0   Unnamed: 0       62630 non-null  int64
 1   UTC              62630 non-null  int64
 2   Temperature[C]   62630 non-null  float64
 3   Humidity[%]      62630 non-null  float64
 4   TVOC[ppb]        62630 non-null  int64
 5   eCO2[ppm]        62630 non-null  int64
 6   Raw H2           62630 non-null  int64
 7   Raw Ethanol      62630 non-null  int64
 8   Pressure[hPa]    62630 non-null  float64
 9   PM1.0            62630 non-null  float64
 10  PM2.5            62630 non-null  float64
 11  NC0.5            62630 non-null  float64
 12  NC1.0            62630 non-null  float64
 13  NC2.5            62630 non-null  float64
 14  CNT              62630 non-null  int64
 15  Fire Alarm       62630 non-null  int64
dtypes: float64(8), int64(8)
```

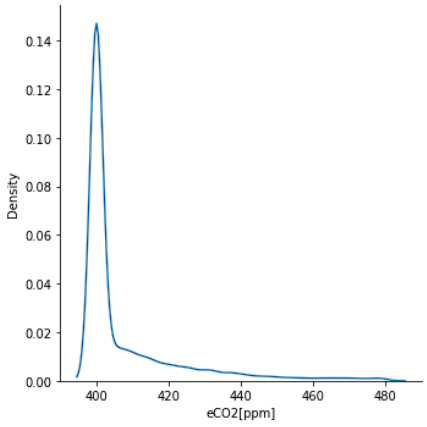
A partir disso o tratamento para outliers foi feito e, inicialmente, percebeu-se uma grande quantidade de outliers presentes na base. Foi necessário fazer o tratamento dos dados removendo-os como é mostrado nox bloxplots a seguir.

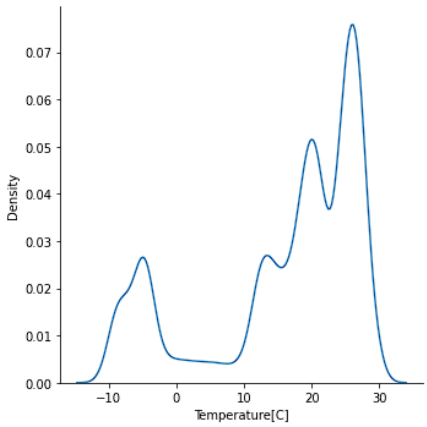


A seguir, verificou-se que existiam muito mais valores verdadeiros do que falsos no atributo de classificação e, para que um balanceamento adequado fosse alcançado, foi utilizada a técnica de undersampling.



Além disso, uma breve visualização da distribuição dos dados foi feita para maior compreensão do estado do dataset, como pode ser observado, a distribuição varia dependendo da natureza do dado dos atributos de teste, seguem alguns exemplos, pela tabela ter muitas colunas não foi possivel adicionar todos os gráficos mas todos os gráficos podem ser analisados no arquivo JupyterNotebook:





Por fim, com o intuito de analise da estrutura interna dos dados e a variação dos dados, foi aplicado o algoritmo PCA (principal component analisys). Esse algoritmo tem o foco em encontrar os principais componentes da base de dados, traçando correlações entre os atributos. Esse algoritmo é útil em casos que existem diversas colunas, ignorando as de menor significância e, por consequência, reduzindo a dimensão do dataset.

Por se fundamentar em uma combinação linear de diversos atributos, pode ser difícil para humanos extrair valores de maneira simples do resultado gerado. Entretanto, a aplicação desse conceito matemático facilita o processamento de outros algoritmos de machine learning.