Relatório 1.ª parte do projeto de Ciência de Dados

Docente: Fábio Mendonça

Bruno Carvalho
n.º 2120122

Leonardo Sousa
n.º 2123822

***Abstract*—Este relatório visa cobrir o processo realizado para a primeira fase do projeto prático da UC de Ciência de Dados. Este relatório irá cobrir a realização das primeiras três fases do ciclo de ciência de dados, nomeadamente: discovery, data pre-processing e model planning.**

I. Introdução

Tal como referido no abstrato, este relatório irá cobrir todo o trabalho realizado para a realização da primeira fase do projeto de Ciência de Dados. Em cada um dos capítulos vamos descrever as diferentes fases e processos realizados em detalhe justificando as nossas decisões no decorrer da implementação.

II. 1.ª Fase - Discovery

Na primeira fase do ciclo de vida de Ciência de Dados um dos nossos principais objetivos é enquadrar o problema de negócio como um desafio de análise. O dataset fornecido trata-se de dados referentes a viagens de taxi em Nova Iorque ao longo de um ano. Estas mesmas viagens têm vários fatores associados como a hora de início e fim, distância percorrida, número de passageiros e entre outros dados associados a cada viagem. Toda esta informação conclui no ”fare amount” que é o valor que foi necessário pagar pela viagem de taxi. O nosso objetivo no âmbito da ciência de dados é formular um modelo que consiga prever o valor do ”fare amount” com base nos dados associados a cada viagem.

III. 2.ª Fase - Data pre-processing

Uma vez que o problema estava definido, estava na hora de efetivamente extrair os dados e processar os mesmos. O que se segue é uma descrição de cada classe conforme as mesmas são invocadas:

A. TripDataAnalizer

Após dar load dos dados de todos os meses é feita uma análise inicial dos dados invocando a classe *TripDataAnalizer* na qual as features são separadas em dados: discretos, contínuos e categóricos. Adicionalmente são extraídas features a partir das colunas de tempo e hora da viagem através de uma conversão para segundos formatando as features ”tpep-pickup-datetime” e ”tpep-dropoff-datetime” no formato *datetime* para facilitar cálculos futuros.

Adicionalmente, colunas desnecessárias são deixadas de lado como por exmplo o ”store-and-fwd-flag” e as de data de pickup e dropoff uma vez que já foram decompostas anteriormente. É de se realçar que também damos ”drop” da coluna do *total-amount* uma vez que, ao fazer análise da importância de features o *total-amount* torna-se problemático uma vez que tem uma correlação demasiado forte com o *fare-amount* o que seria problemático para o nosso modelo, sendo deixado de parte logo no início da nossa análise.

Identify applicable funding agency here. If none, delete this.

B. DataPreprocessor

Após toda esta divisão de features, usamos o *DataPreprocessor* para normalizar features numéricas usando o *StandardScaler* para ter uma mediana igual a zero e *standard diviation* igual a 1. Adicionalmente é usado o Min-Max para ajustar a escala das features categoriais.

C. DataCleaning

A classe de DataCleaning é responsável por realizar todo o tipo de tratamento e limpeza dos dados, isto incluí a rremoção de outliers e valores em falta. A remoção de valores em falta pode ser resolvida com base na estratégia definida, podendo substituir os valores pela média, a mediana , a moda ou valores constantes. Esta classe também permite lidar com valores em falta fazendo *drop* dessa linha.

D. EDA

Após todo este processamento de dados, realizamos o *Exploratory Data Analysis (EDA)* para poder realizar a visualização das propriedades do dataset.

Nesta classe são realizados vários processos de descrição de dados tais como as informações do dataset, um sumário estatístico, e os valores em falta de cada uma das features. Adicionalmente é realizado a visualização de features numéricas usando histogramas:

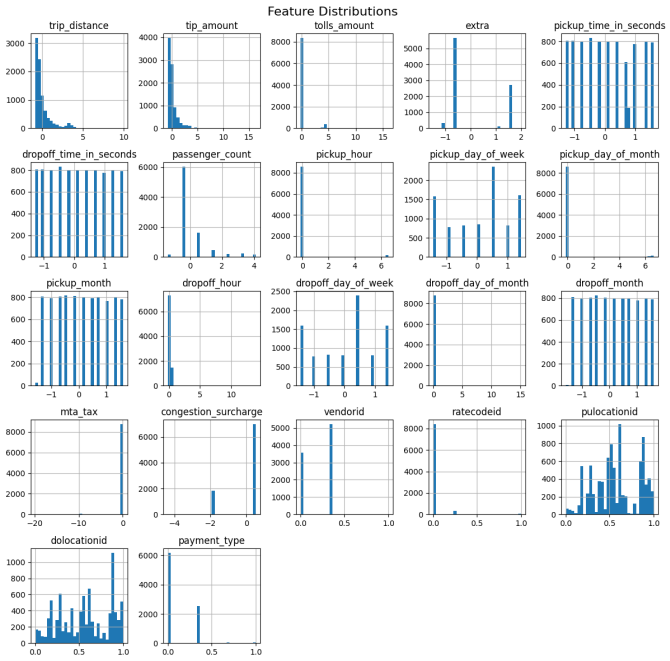

Fig. 1. Histogramas

São criados boxplots para features numéricas para facilitar a deteção de possíveis outliers:

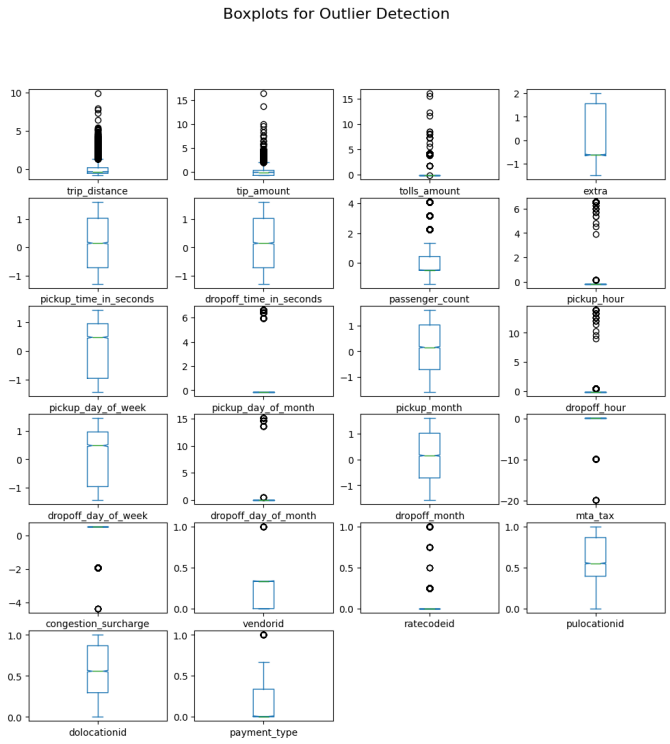

Fig. 2. Boxplots

É gerado um heatmap para demonstrar a correlação entre as diferentes features:

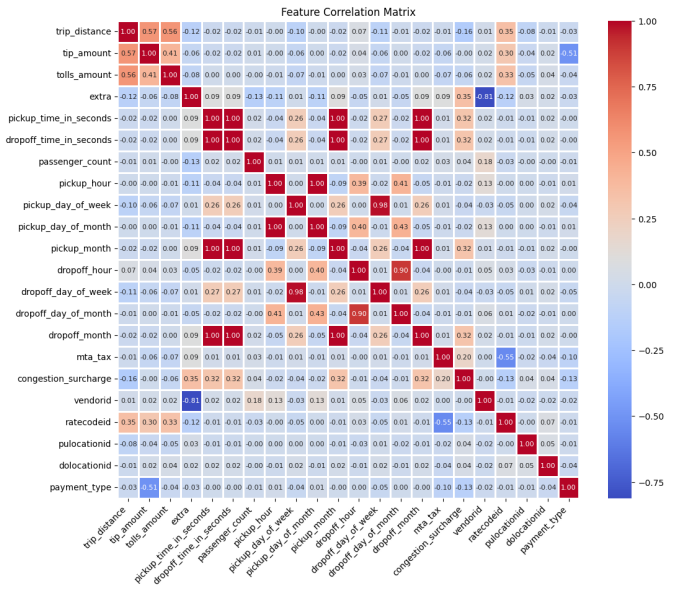

Fig. 3. Heatmap

E por último são realizados *pairplots* para ver a relação entre cada feature numérica:

O QUE FAZER EM RELAÇÃO Á IMAGEM???

E. FeatureAnalysis

De seguida é realizada uma análise das *features* que temos no momento, esta classe é responsável por realizar a computação e visualização da *feature importance* usando o modelo *Random Forest*:

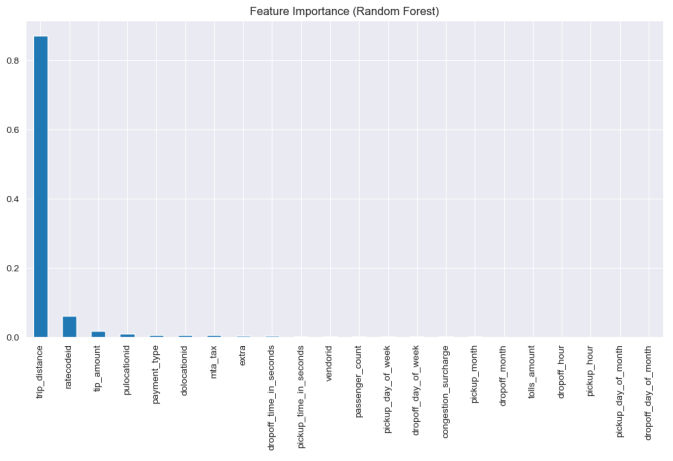

Fig. 4. Feature Importance (usando o modelo Random Forest)

Para além disso, esta classe também executa o algoritmo PCA para visualizar o dataset num espaço 2D:

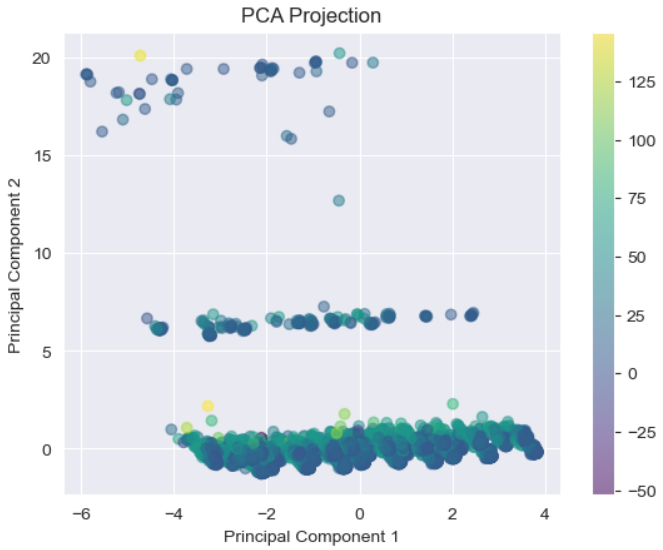

Fig. 5. PCA Projection

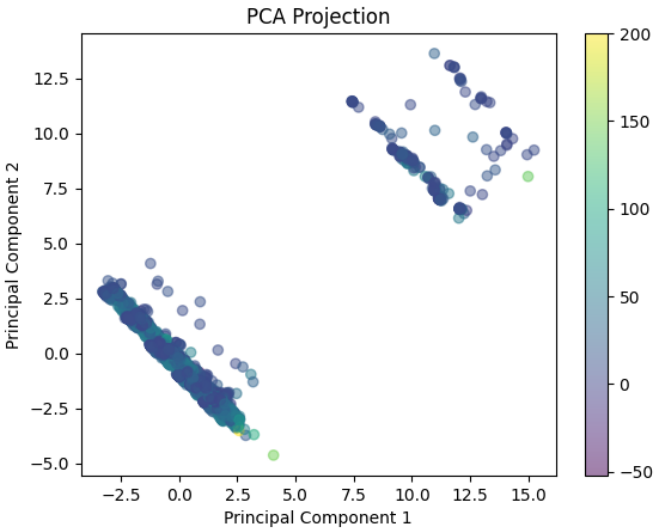

Fig. 7. PCA Projection após geração de novas features

F. Feature Generation

Esta classe é responsável por gerar novas features com base nas features já existentes, são criadas features como por exmplo a duração da viagem (trip-duration-min) e a velocidade média (average-speed-mph). Após a geração das novas features, o Feature Analysis é invocado novamente para avaliar o impacto das novas features:

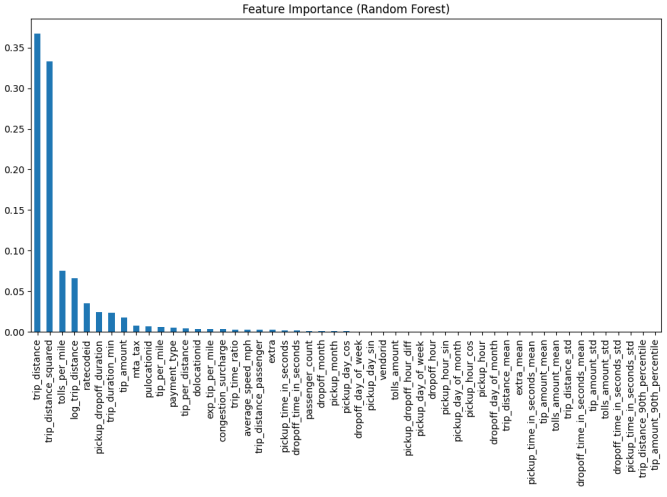

Fig. 6. Feature Importance após geração de novas features

G. DataVisualization

Esta classe é responsável por visualizar os dados, sendo realizados boxplots e ridgeplots para todas as features:

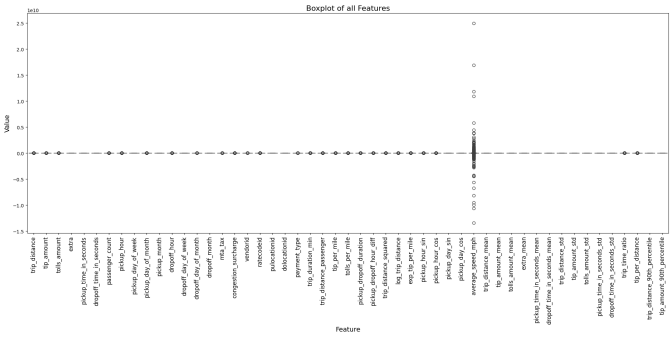

Fig. 8. Boxplot de todas as features

H. HypothesisTesting

Por último a classe de teste de hipóteses é responsável por realizar a análise da variância (ANOVA test), os resultados *Kruskal-Wallis* e resultados t-test.

IV. 3.ª Fase - Model Planning

Após toda a análise e tratamento de dados, foi possível chegar à conclusão de que o modelo mais apropriado seria o Random Forest uma vez que estamos na presença de relações não lineares entre as diferentes features. Adicionalmente permite computar *feature importance* o que provou ser um dado bastante valioso, é um modelo bastante resistente a outliers o que é algo muito prevalente nesta base de dados bem como ruído.

Para toda esta primeira fase do projeto nós seguimos o ciclo da ciência de dados, realizando todas as tarefas principais tais como a limpeza de dados e pre-processamento dos dados. Também foram realizados histogramas e boxplot para visualizar melhor o comportamento de cada feature antes e depois

de serem criadas novas features que consideramos relevantes. Por último foi realizado os testes de hipóteses para avaliar a significância de cada classe.