Relatório de Previsão de Preços de Corridas

Integrantes:

Vinicius Miranda Andrade Piovesan RA: 23025544

Matheus de Medeiros Takaki RA: 23024683

Sérgio Ricardo Pedote Junior RA:23747441

Felipe Ribeiro Almeida RA: 23025143

1. Introdução

Com o avanço da mobilidade urbana e o crescimento exponencial de plataformas de transporte como Uber, 99 e táxis, torna-se cada vez mais importante a utilização de técnicas de inteligência artificial (IA) para a previsão de preços de corridas. Esses preços são influenciados por diversas variáveis, como distância percorrida, tempo de espera, demanda e tipo de serviço. A previsão eficiente desses valores pode otimizar os serviços e proporcionar uma experiência mais transparente para os usuários. Este relatório aborda a aplicação de um modelo de Regressão Linear para prever os preços das corridas com base em dados históricos extraídos de diferentes plataformas de transporte.

---

2. Objetivo

O principal objetivo deste trabalho é desenvolver um modelo preditivo capaz de estimar o preço das corridas com base em variáveis que influenciam diretamente esse valor. Utilizando o algoritmo de **Regressão Linear**, busca-se identificar os padrões subjacentes que determinam o preço de cada corrida. Este estudo visa fornecer uma ferramenta que possa ser aplicada para melhorar o planejamento e a gestão das plataformas de transporte, otimizando a precificação e aumentando a satisfação dos usuários.

---

3. Metodologia

A metodologia aplicada foi estruturada em várias etapas para garantir a construção de um modelo robusto e eficaz. Essas etapas incluem desde a coleta de dados até a avaliação do modelo preditivo. O processo foi dividido da seguinte forma:

- **Coleta de Dados**: Extração dos dados históricos de corridas das plataformas de mobilidade, como Uber, 99 e serviços de táxi. Estes dados incluem variáveis relevantes como tempo de espera, tipo de serviço, preço e identificadores das corridas e estimativas.
- **Limpeza e Tratamento dos Dados**: Realização de um processo de limpeza para remover inconsistências, valores nulos e transformações necessárias, como a normalização dos dados e codificação de variáveis categóricas.
- **Análise Exploratória de Dados (EDA)**: Estudo das variáveis para entender suas distribuições e identificar correlações importantes entre os dados, com o intuito de melhorar a compreensão do comportamento do preço.
- **Treinamento do Modelo**: Utilização do algoritmo de **Regressão Linear** para treinar o modelo com as variáveis independentes selecionadas e a variável dependente (preço).
- **Avaliação e Validação**: Após o treinamento do modelo, foi realizada a avaliação do desempenho utilizando métricas como **RMSE (Root Mean Squared Error)** e **R^2 (Coeficiente de Determinação)**, que indicam a acuracidade e a capacidade de previsão do modelo.

---

4. Aplicação do Algoritmo de Regressão Linear

A **Regressão Linear** foi escolhida devido à sua simplicidade e eficácia para modelar variáveis contínuas, como o preço das corridas. As variáveis consideradas para a modelagem foram:

- **Variáveis Numéricas**: Preço das corridas (Price), tempo de espera (WaitingTime).
- **Variáveis Categóricas**: Tipo de serviço (por exemplo, **UberX**, **Comfort**, **poupa99**), e identificadores únicos da corrida e estimativa.

O processo de treinamento envolveu a divisão dos dados em conjuntos de **treinamento** e **teste**, com 80% dos dados utilizados para treinamento e 20% para validação do modelo. Além disso, a normalização dos dados foi realizada para garantir que todas as variáveis tivessem a mesma escala, o que é importante para a regressão linear.

Foi aplicada a técnica de **normalização** utilizando o **StandardScaler** para ajustar os valores das variáveis de modo que todas tivessem média zero e desvio padrão igual a um, o que facilita a convergência do modelo.

A avaliação do modelo foi feita com as métricas **RMSE (Root Mean Squared Error)** e **R^2 (Coeficiente de Determinação)**:

- **RMSE**: A raiz quadrada do erro médio quadrático, que mede a magnitude média dos erros de previsão. Quanto menor o valor do RMSE, melhor a performance do modelo.
- **R^2**: Mede a proporção da variabilidade dos dados que é explicada pelo modelo. Quanto mais próximo de 1, melhor o modelo.

---

5. Resultados

Os resultados obtidos do modelo de regressão linear demonstram que ele possui boa capacidade de previsão. A **métrica RMSE** foi de aproximadamente **88,11**, o que indica que a média do erro nas previsões do preço das corridas está em torno desse valor. Isso sugere que o modelo consegue prever o preço com um nível razoável de acuracidade, embora haja espaço para melhorias.

Além disso, o **R^2** do modelo foi de aproximadamente **0,85**, o que significa que o modelo é capaz de explicar 85% da variação nos preços das corridas com base nas variáveis selecionadas. Esse valor de R^2 é considerado muito bom, pois indica uma forte relação entre as variáveis independentes e a variável dependente.

A análise gráfica também confirmou que a regressão linear está adequadamente capturando as relações entre as variáveis, embora haja algumas discrepâncias para os casos em que o preço real se distanciou significativamente da previsão.

6. Conclusão

A aplicação de algoritmos de **Regressão Linear** demonstrou ser uma abordagem eficiente para prever o preço das corridas em diferentes plataformas de mobilidade urbana. O modelo desenvolvido forneceu previsões com boa acuracidade, e as métricas de desempenho (RMSE e R^2) indicaram que ele é capaz de capturar a maior parte da variação nos preços das corridas.

Esse modelo pode ser uma ferramenta útil para otimização da precificação das corridas, auxiliando na melhoria da experiência do usuário e na estratégia das empresas de transporte. A continuidade deste trabalho pode envolver a experimentação com modelos mais complexos, como **regressão polinomial** ou **redes neurais**, para explorar maior complexidade nos dados e melhorar ainda mais a precisão das previsões.

O uso de variáveis adicionais, como condições climáticas ou eventos especiais na cidade, poderia ser integrado ao modelo para aumentar ainda mais a acuracidade das previsões.

Codigo:

```python
import pandas as pd
from sklearn.linear_model import LinearRegression
from sklearn.model_selection import train_test_split
from sklearn.metrics import mean_squared_error
import numpy as np

df_address = pd.read_csv('rideaddress_v1.csv', delimiter=';')
df_estimative = pd.read_csv('rideestimative_v3.csv', delimiter=';')

df_merged = pd.merge(df_address, df_estimative, on='RideID', how='inner')

df_selected = df_merged[['WaitingTime', 'Price', 'ProductID']]

df_selected = df_selected.dropna()

df_selected['ProductID'] = pd.factorize(df_selected['ProductID'])[0]

X = df_selected[['WaitingTime', 'ProductID']]
y = df_selected['Price']

X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)

model = LinearRegression()

model.fit(X_train, y_train)

y_pred = model.predict(X_test)

mse = mean_squared_error(y_test, y_pred)
rmse = np.sqrt(mse)

print(f"Root Mean Squared Error (RMSE): {rmse}")
```