

Análise e modelagem preditiva de dados

Estudo sobre enchentes em áreas urbanas



Carlos Henrique Hannas de Carvalho, nº USP: 11965988;

Carlos Nery Ribeiro, nº USP: 12547698;

Erik Melges, nº USP: 12547399;

Gabriel Ribeiro Rodrigues Dessotti, nº USP: 12547228;

Gustavo Barbosa Sanchez, nº USP: 11802440;

Pedro Antonio Bruno Grando, nº USP: 12547166;

Pedro Manicardi Soares, nº USP: 12547621.

SCC 0630 - Inteligência Artificial - Professora: Solange Oliveira Rezende

Identificação do Problema

○ Brasil enfrenta inúmeros desastres ambientais

- Prejuízos financeiros e óbitos ao Estado
- Rompimento das barragens de Mariana (2015) e Brumadinho (2019), e a enchente no Rio Grande do Sul (2024)

Objetivo: Usar técnicas de **machine learning** para **estimar probabilidades de ocorrências de enchente em áreas urbanas.**



Dataset

Fonte dos dados


- Kaggle - "Regression with a flood Prediction Dataset".

Características do Dataset

- 1.117.957 exemplares e 21 features
- **Variável alvo: "Probabilidade de Enchente"**

Regression with a Flood Prediction Dataset

Playground Series - Season 4, Episode 5



[Overview](#) [Data](#) [Code](#) [Models](#) [Discussion](#) [Leaderboard](#) [Rules](#) [Team](#) [Submissions](#)

Dataset Description

The dataset for this competition (both train and test) was generated from a deep learning model trained on the [Flood Prediction Factors](#) dataset. Feature distributions are close to, but not exactly the same, as the original. Feel free to use the original dataset as part of this competition, both to explore differences as well as to see whether incorporating the original in training improves model performance.

Note: This dataset is particularly well suited for visualizations, clustering, and general EDA. Show off your skills!

Files

- **train.csv** - the training dataset; `FloodProbability` is the target
- **test.csv** - the test dataset; your objective is to predict the `FloodProbability` for each row
- **sample_submission.csv** - a sample submission file in the correct format

Files

3 files

Size

104.17 MB

Type

CSV

License

[Attribution 4.0 International \(CC BY\)](#)

<https://www.kaggle.com/c/playground-series-s4e5/data>

Pré-processamento dos dados

1. Extração e Integração

- Todos os dados foram extraídos do **arquivo "train.csv"**

2. Transformação

- Apenas dados numéricos

3. Divisão dos dados em conjuntos de treino e teste

- Utilização da função `train-test-split` para divisão dos dados

4. Limpeza

- Sem dados ausentes

Extração de Padrões

Escolha da atividade

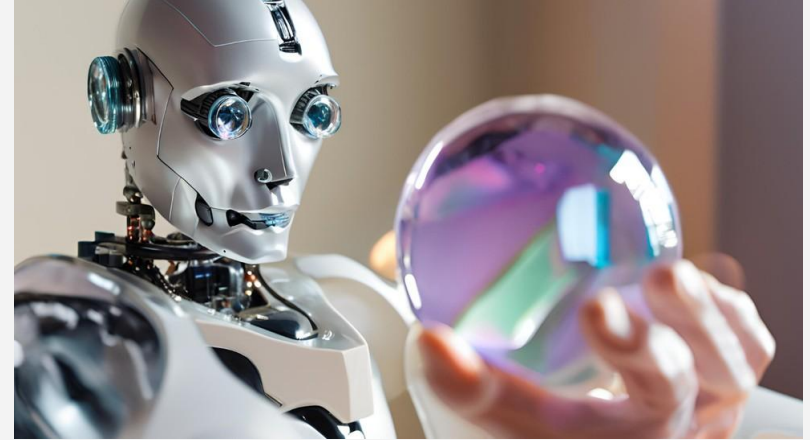


Atividade preditiva

Tipo de aprendizado



Aprendizado supervisionado



[https://s2-techtudo.glbimg.com/yQuA8jGKHeYkwca128vzYZcMGEs=/0x0:1260x700/984x0/smart/filters:strip_icc\(\)/i.s3.glbimg.com/v1/AUTH_08fbf48bc0524877943fe86e43087e7a/internal_photos/bs/2024/D/f/MiCD68SXKI4BEnVtJS6Q/design-sem-nome-12-.png](https://s2-techtudo.glbimg.com/yQuA8jGKHeYkwca128vzYZcMGEs=/0x0:1260x700/984x0/smart/filters:strip_icc()/i.s3.glbimg.com/v1/AUTH_08fbf48bc0524877943fe86e43087e7a/internal_photos/bs/2024/D/f/MiCD68SXKI4BEnVtJS6Q/design-sem-nome-12-.png)

Bibliotecas e leitura dos dados

Bibliotecas Utilizadas

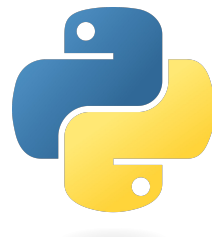
pandas —————> manipulação de dados

scikit-learn —————> pré-processamento/extração de padrões

Leitura do Dataset

Leitura do arquivo CSV —————> *pandas*

Manipulação dos dados —————> *dataframe* com *pandas*



Seleção de *Features* e modelagem inicial

Seleção de *Features*

SelectKBest e Chi-squared (chi2):

Uso do SelectKBest para selecionar as melhores *features* com base no teste qui-quadrado.

Importância da Seleção de *Features*:

Redução da dimensionalidade e melhoria do desempenho do modelo.

Modelagem Inicial:

RandomForestRegressor

Algoritmo *Random Forest*:

Combina múltiplas árvores de decisão para melhorar a precisão e evitar *overfitting*.

Treinamento do Modelo:

Treinamento com os dados de treino e avaliação com o MSE.

Avaliação de Diferentes Modelos

1. Linear Regression:

- Simplicidade e eficiência.
- MSE: 0.00036516456815355775

2. K-Nearest Neighbors Regressor:

- Facilidade de entendimento e aplicação.
- MSE: 0.00119586600000000003

3. Ridge Regression:

- Adição de regularização para evitar overfitting.
- MSE: 0.00036516639323572333

4. Support Vector Regressor (SVR):

- Uso para capturar relações não lineares.
- MSE: 0.001239588093034845

Validação cruzada e otimização de Hiperparâmetros

Validação Cruzada

Importância:

- Avaliação robusta do desempenho do modelo.

Resultados:

- Aplicação da validação cruzada para o modelo RandomForestRegressor.

Otimização de Hiperparâmetros

GridSearchCV:

- Utilização do **GridSearchCV** para encontrar os melhores hiperparâmetros.

Comparação do Desempenho:

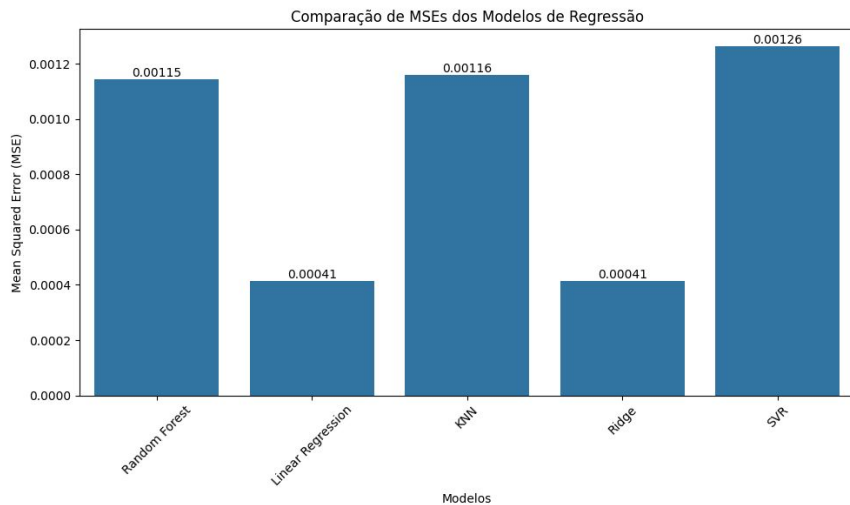
- Avaliação do desempenho do modelo antes e depois da otimização.

Resultados

Valores de MSE para cada modelo

RandomForestRegressor:

- MSE inicial: 0.001141
- MSE variando o número de features:
 - K = 1: 0.002385
 - K = 10: 0.001802
 - K = 20: 0.001145



Linear Regression:

- MSE: 0.0004149486536292868

K-Nearest Neighbors Regressor:

- MSE: 0.0011598595000000002

Ridge Regression:

- MSE: 0.00041494680626900487

Support Vector Regressor (SVR):

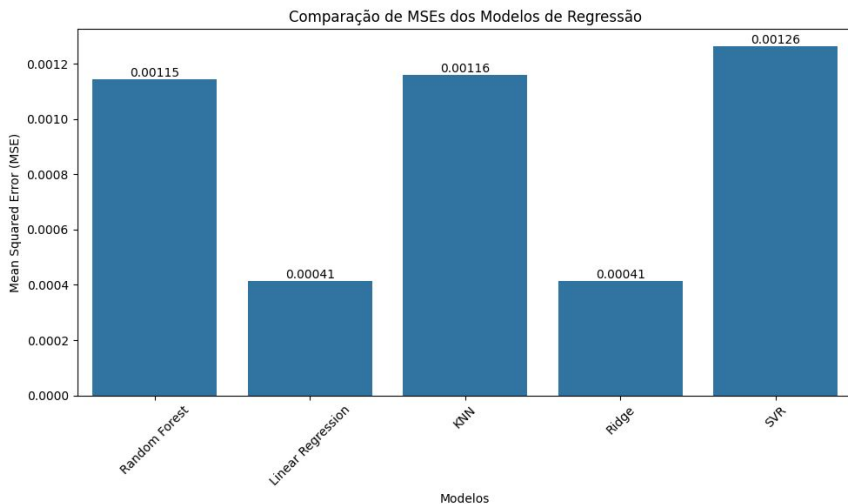
- MSE: 0.0012627002738536458

Resultados

Discussão sobre a adequação dos modelos

RandomForestRegressor:

- **Vantagens:** Robusto e eficaz com a seleção de features adequada.
- **Desvantagens:** Sensível à seleção de features e pode ser computacionalmente intensivo.



Linear Regression:

- **Vantagens:** Simplicidade e eficiência.
- **Desvantagens:** Pode não capturar relações complexas.

K-Nearest Neighbors Regressor:

- **Vantagens:** Fácil de entender.
- **Desvantagens:** Sensível à dimensionalidade dos dados.

Ridge Regression:

- **Vantagens:** Regularização ajuda a evitar *overfitting*.
- **Desvantagens:** Pode não capturar relações complexas.

Support Vector Regressor (SVR):

- **Vantagens:** Modela relações não lineares.
- **Desvantagens:** Complexo de ajustar.

Importância da seleção de *Features* e *Tuning* de Hiperparâmetros

Seleção de Features:

- Impacto significativo no desempenho, especialmente para RandomForestRegressor.
- Melhor MSE com $K = 20$.

Tuning de Hiperparâmetros:

- Essencial para extrair o melhor desempenho.
- A falta de tuning pode levar à subutilização do potencial do modelo.

Conclusão

Resumo dos Principais Achados

- **Melhores Modelos:** Linear Regression e Ridge Regression com MSE de 0.00041495.
- **Seleção de Features:** Crucial para o desempenho do RandomForestRegressor.

Considerações Finais sobre a Performance dos Modelos

- **Eficiência:** Linear e Ridge são simples e eficientes.
- **Complexidade:** Modelos como RandomForest e SVR necessitam de *tuning* e seleção de *features*.

Sugestões para Trabalhos Futuros

- **Novos Algoritmos:** Testar XGBoost e LightGBM.
- **Importância das Features:** Analisar as features mais impactantes.
- **Engenharia de Features:** Criar novas features para melhorar a previsão.

Obrigado!