**Resumo Projeto - Ciência de dados**

**Disciplina:** Tópicos de projetos 4

**Aluno:** Alan Firmino da Silva

**Professor:** Francisco Fernando Roberto Pereira

**Introdução**

O desafio proposto pelo professor consiste em reproduzir praticamente um pouco do que foi visto em aula e durante as atividades semanais. Este MVP precisa de alguns requisitos para ser completo, e a partir do notebook criado, o professor avaliará toda a sua lógica e meu entendimento sobre o assunto.

[Requisitos para o projeto.](https://github.com/AlanFirmino2024/projeto_cd/blob/main/Instru%C3%A7%C3%B5es%20para%20projeto%20de%20ci%C3%AAncia%20de%20dados.pdf)

# Tópico 1 – Descrição do problema

Neste tópico explicamos a origem do dataset, quais atributos que contém nele e o que cada um é, e também importamos as bibliotecas que iremos utilizar durante o desenvolvimento do MVP.

# Tópico 2 – Carga de dados

Aqui fazemos o carregamento do meu dataset no notebook, e conferimos em que formato está nosso dataset.

Após conferir, foi verificada a necessidade de normalizar o formato dos dados de cada bloco do nosso dataset separadamento, tudo isso utilizando a biblioteca *pandas.* E, posteriormente juntando os dataframes gerados anteriormente.

# Tópico 3 – Análise dos dados

Neste tópico, faremos uma análise descritiva do dataset para assim, temos uma melhor visão de como os dados estão dispostos.

Começamos vendo as dimensões e as informações do dataset para temos uma noção do tamanho e os tipos de dados que vamos trabalhar, e após analisarmos vemos que alguns atributos estão com o tipo errado, ou seja, precisamos trata-los.

Após realizar a troca do tipo de alguns atributos, geramos o relatório estatístico, para ter uma melhor noção de todos eles no nosso dataset.

# Tópico 4 – Preparação dos dados

Neste tópico, iremos preparar o nosso dataset da melhor forma e fazendo algumas considerações, para que possamos ter uma modelagem de qualidade.

Primeiro, aplicamos um filtro, onde definimos que só queremos trabalhar com imóveis residenciais na cidade do Rio de Janeiro (Esse filtro contempla mais de 95% do dataset).

Após o filtro, chegamos e vemos que o filtro não chegou a apagar os dados que não queremos. Por isso, houve a necessidade de resetarmos o índice para deletar esses dados não usuais.

Após isso, exibimos cada atributo individualmente para verificação dos dados vazios e nulos, e posteriormente tratamos estes problemas.

Os itens vazios perteciam apenas ao atributo de zona, e como a zona é um atributo essencial preferimos não deletá-los. Em contrapartida, criamos um dicionário onde cada bairro era referente à uma zona. Depois utilizamos este dicionário para preencher os valores vazios no atributo zona a partir destes bairros.

Quanto as nulidades a serem tratadas, só existiam nas colunas de taxa de iptu e taxa de condomínio. Porém, aqui é mais complicados pois sabemos que há imóveis que não pagam iptu e nem condomínio. Com isso, assumimos que os valores nulos dessas colunas se referem ao não pagamento dessas taxa.

Posteriormente, renomeamos os atributos para melhor entendimento, e criamos um gradiente de histogramas para verificar o comportamento de cada atributo numérico. A partir dele, vemos que alguns o dataset possui alguns *outliers* e que alguns atributos possuem comportamento exponencial, o que pode afetar nosso modelo ao tentar prever, por exemplo, um valor de casa alto, uma vez que a maior parte dos casas são de valor baixo.

Após essa conclusão, decidimos que era necessário realizar uma transformação para normalizar nossos dados, mas antes foi necessário tratar os outliers. Para tratar esses outliers assumimos alguns requisitos, e se eles não fossem atendidos eles seriam excluídos. Foi o que houve com alguns valores de vagas de estacionamento, taxa de condomínio, taxa de iptu e número de andares do imóvel.

Partindo assim, para o tratamento dos dados para normalizá-los, onde utilizou-se uma transformação logarítmica, depois verificamos a correlação dos atributos entre si, utilizando um *Heat Map*.

E para finalizar este tópico, utilizamos a função *dummies,* do *pandas,* para converter as variável categóricas em variáveis booleanas, e assim conseguirmos utilizar esses atributos para melhorar o modelo. Fazendo posteriormente, o treino e teste do conjunto de dados.

# Tópico 5 – Modelagem

Neste tópico, utilizou-se 6 modelos diferentes para realizar a modelagem dos dados e a partir deles avaliar, a partir das métricas de desempenho, qual o melhor modelo de regressão e quais hiperparâmetros são os melhores referente a ele.

Os modelos são: Regressão linear, Árvore de decisão, Floresta aleatória, KNeighbostRegressor, AdaBoostRegressor e GradientBoostingRegressor.

Para cada modelo, fizemos os seguintes procedimentos:

1. Treinamos utilizando apenas o modelo.
2. Otimizamos o modelo com a validação cruzada.
3. Otimizamos o modelo com o *RandomSearchCV* com os dados de treino.
4. Testamos o modelo com o *RandomSearchCV* com os dados de teste*.*
5. Verificamos as métricas de desempenho para regressão.

Durante a modelagem, foram vistos alguns pontos. O modelo KNN não se adequou tão bem em relação ao outros (*é possível a existência de underfitting*). E as causas para otimizar os hiperparâmetros foram para evitar problemas como possíveis *overfittings* e *underfittings* e para melhorar o desempenho do modelo.

Feitas as modelagens, vamos para o próximo tópico que é o de comparação dos modelos.

# Tópico 6 – Comparação dos modelos

Neste tópicos, utilizamos duas ferramentas gráficas (Erros de predição e Gráficos de resíduos) para analisar o desempenho de cada modelo e utilizamos as métricas de desempenho para escolher qual modelo e os respectivos hiperparâmetros iremos utilizar.

As ferramentas gráficas nos mostraram que o modelo de regressão linear deu algo não esperado: O valor do coeficiente de determinação do conjunto de teste deu maior que o do conjunto de treino. Isso é algo não esperado, e algumas causas podem ser o *overfitting* no conjunto de treino, por diferenças de amostragem, ou por uma variabilidade aleatória que o conjunto de teste seja mais fácil de prever. Enfim, decidimos aceitas essas causas, pois a diferença entre esses valores é muito pequeno.

Para finalizar o tópico, avaliamos e ranqueamos os melhores modelos com os melhores hiperparâmetros, ou seja, ordenamos pelo maior valor do coeficiente de determinação. Assim, chegamos ao modelo que apresentou melhor desempenho para o nosso conjunto de dados.

# Tópico 7 – Seleção, salvamento e carregamento do modelo

Aqui é onde pegamos o modelo, salvamos num arquivo pickle e carregamos ele numa variável.

# Tópico 8 - Aquisição de dados e previsão

Este é o último tópico e consiste em captar os dados do imóvel que queremos saber o valor e realizar a previsão a partir do modelo escolhido e destes dados captados.