# ***Machine Learning Project***

**Numerical Dataset Documentation**

|  |  |
| --- | --- |
| **ID** | **Name** |
| **20210065** | **Ahmed Tarek Abdelwali Ahmed** |
| **20210080** | **Ahmed Alaa Abassi** |
| **20210058** | **Ahmed Said Mohamed Said** |
| **20210044** | **Ahmed Khaled Amin Kadi** |
| **20210119** | **Ahmed Medhat Abobakr** |
| **20210590** | **Omar Ahmed Elsaid Ahmed** |

# **Numerical dataset :**

General information about dataset

|  |  |
| --- | --- |
| Name | House Prediction |
| No. of classes |  |
| Total no. of samples | 20,433 |
| No. of samples in training / validation | 18000 / 933 |
| No. of samples in testing | 1500 |

The provided code performs various tasks related to housing data analysis and linear regression modeling. The main steps include data loading, exploration, visualization, preprocessing, model training, and evaluation.

**Importing Libraries :**

The code begins by importing necessary libraries for data manipulation, visualization, and machine learning tasks.

A screen shot of a computer code

Description automatically generated

### Data Loading and Exploration :

Loads a dataset ('housing.csv') into a Pandas DataFrame and displays the first few rows of the dataset. The dataset likely contains information about housing.

A close-up of a computer code

Description automatically generated

A screenshot of a computer

Description automatically generated

### Geographical Visualization

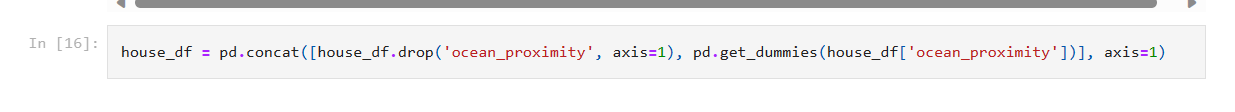
Creates a scatter plot with geographical coordinates, population size, and median house value. The color of the points represents the median house value, and a California map is overlaid for context.

A map of the state of california

Description automatically generated

### Data Preprocessing :

One-hot encodes the categorical variable ('ocean\_proximity') and drops the original column.



Checks for missing values in the dataset and drops rows containing missing values.

A screenshot of a computer

Description automatically generated

### Data Splitting

Splits the dataset into training, testing, and validation sets.

Further divides the data into features (X) and target values (y) for training, validation, and testing sets.

A screenshot of a computer code

Description automatically generated

### Scaling

Standardizes the feature values using StandardScaler to ensure that all features have the same scale.

A screenshot of a computer

Description automatically generated

### Linear Regression Model Training and Evaluation

Trains a linear regression model using the training data and evaluates its performance on the scaled test data.

A screenshot of a cell phone

Description automatically generated

Calculates and prints mean squared error and R-squared as evaluation metrics for the linear regression model.

A screenshot of a computer error

Description automatically generated

A graph showing a line of dots

Description automatically generated with medium confidence

### K-Nearest Neighbors (KNN) Regression

Implements K-Nearest Neighbors regression with n\_neighbors=20 and evaluates its performance on the test set. Then Predicts values using the KNN model on the test set and creates a DataFrame with actual and predicted values for comparison. Then Calculates and prints mean squared error and R-squared for the KNN model's predictions.

A screenshot of a computer program

Description automatically generated

Performs K-Nearest Neighbors regression with varying numbers of neighbors and plots the training and testing accuracies to identify an optimal number of neighbors.

A graph of a number of neighbors

Description automatically generated

### Cross-Validation with Linear Regression

Performs k-fold cross-validation (k=6) on the linear regression model and prints the cross-validation results.

A computer code on a white background

Description automatically generated

### Model Prediction Visualization

Plots the actual and predicted values for the first 50 samples in the test set, providing a visual representation of how well the model predictions align with the true values. The legend indicates the distinction between actual and predicted values. This type of plot is useful for a quick assessment of model performance and identifying any patterns or discrepancies between the predicted and actual values.A graph with blue and orange lines

Description automatically generated

### Random Forest & Grid Search

### Here we trains model using random forest algorithm , evaluates its performance, performs a grid search to find the best hyperparameters, trains a new model with the best hyperparameters, and evaluates its performance on the test set. Replace X and y with your actual feature matrix and target variable

A screenshot of a chat

Description automatically generated

A screenshot of a computer program

Description automatically generated

### Model Comparison and Visualization

Creates boxplots to compare the predicted values from the Linear Regression and KNN models.

A screenshot of a graph

Description automatically generated

Plots the actual values against the predicted values from both the KNN and Linear Regression models for a visual comparison.

A graph showing a number of data

Description automatically generated with medium confidence