Linear regression documentary

A screen shot of a computer program

Description automatically generated**Importlibraries**

---------------------------------------------------------------------------------------------------------------------------

this code loads Tesla stock data from a CSV file into a DataFrame, converts the 'Date' column to datetime objects, displays the minimum and maximum dates (concatenated without a separator, which might make it hard to read), and then shows the total number of days covered by the dataset.A screenshot of a computer code

Description automatically generated

---------------------------------------------------------------------------------------------------------------------------

This code computes the correlation matrix for the columns in the tesla DataFrame and then prints the resulting correlation matrix.



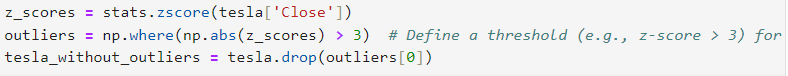
This code is used to drop specific columns from the tesla DataFrame using Pandas.

A white background with black and red letters

Description automatically generated

---------------------------------------------------------------------------------------------------------------------------

This code appears to be performing outlier detection and removal based on z-scores in the 'Close' column of the tesla DataFrame.



---------------------------------------------------------------------------------------------------------------------------

This code sets up a customized layout for a Plotly plot and then generates a visualization showing Tesla stock prices over time, excluding outliers from the 'Close' prices.

A screenshot of a computer program

Description automatically generated

this code calculates a moving average, defines features and target variables, splits the dataset for training and testing, and then normalizes the feature data to prepare it for use in a machine learning model, typically in a regression or predictive context.

A screenshot of a computer program

Description automatically generated

---------------------------------------------------------------------------------------------------------------------------

this code segment trains a Linear Regression model, generates predictions on the test set, computes R-squared and MSE as evaluation metrics, and then displays these metrics to assess the model's performance on the scaled test data.

A screenshot of a computer code

Description automatically generated

---------------------------------------------------------------------------------------------------------------------------

this code generates a Plotly visualization illustrating a comparison between the actual and predicted values from a Linear Regression model. It plots the actual values as markers and the predicted values as a line plot against the first feature of the scaled training data, allowing visual inspection of how well the model predictions align with the actual data.

A screenshot of a computer code

Description automatically generated

---------------------------------------------------------------------------------------------------------------------------

this code performs Ridge Regression, assesses its performance using evaluation metrics (MSE and R-squared), and presents a scatter plot visualizing how well the Ridge model's predictions align with the actual values.

A screenshot of a computer code

Description automatically generated

---------------------------------------------------------------------------------------------------------------------------

this code sets up a KNN regression model, trains it using the training data, generates predictions on the test data, and computes evaluation metrics (R-squared and MSE) to assess the model's predictive performance. Adjusting the number of neighbors (n\_neighbors) can impact the model's behavior and performance.

A screenshot of a computer program

Description automatically generated

---------------------------------------------------------------------------------------------------------------------------

this code snippet generates a scatter plot where the x-axis represents the actual values from the test dataset (y\_test), the y-axis displays the predicted values from the KNN model (predictions), and each point represents a data instance. This visualization allows for a visual assessment of how well the model's predictions align with the actual values.

A screenshot of a computer

Description automatically generated

---------------------------------------------------------------------------------------------------------------------------

Kmeansdocumentary

Import libraries

A screenshot of a computer program

Description automatically generated

---------------------------------------------------------------------------------------------------------------------------

this code loads images from a specified folder, preprocesses them by converting to grayscale, resizing, and flattening, then encodes the labels and stores the processed data in a CSV file for further use, like training a machine learning model on image data.

A screenshot of a computer program

Description automatically generated

this code segment reads the contents of a CSV file ('output\_data.csv') into a Pandas Data Frame (sheet) and then displays the top rows of this Data Frame to provide a preview of the loaded data.

A screenshot of a computer

Description automatically generated

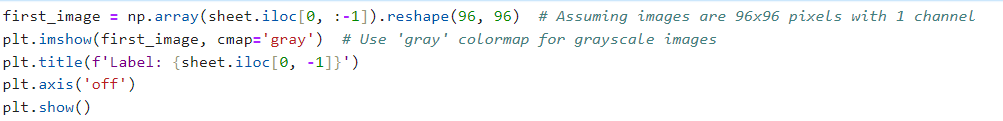
---------------------------------------------------------------------------------------------------------------------------

these methods offer a comprehensive overview of the dataset's structure, data types, presence of missing values, and statistical summaries of numerical columns. They are crucial initial steps in understanding and exploring the characteristics and content of the dataset.

A screenshot of a computer code

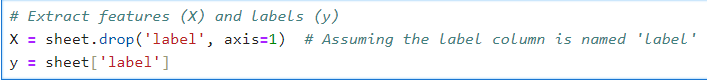
Description automatically generated

**this code snippet extracts pixel data for the first image in the dataset, reshapes it to a 96x96 grayscale image, and then uses Matplotlib to display the image with its associated label. This visualization helps to preview the appearance of the image and understand its content in relation to its label or class.**

****

---------------------------------------------------------------------------------------------------------------------------

**this code separates the features (X) from the labels (y) in the sheet Data Frame, assuming that the 'label' column holds the target variable or class labels, and prepares the data for use in a machine learning model where X represents the features and y represents the corresponding labels.**

****

**This code snippet performs feature normalization by calculating the mean and standard deviation for each column in the feature dataset (X), and then normalizing each column by subtracting the mean and dividing by the standard deviation. Here's a summary:**

**A screenshot of a computer program

Description automatically generated**

---------------------------------------------------------------------------------------------------------------------------

**this code performs K-means clustering with two clusters on the feature dataset X using scikit-learn'sKMeans algorithm. It assigns cluster labels to each data point and adds these labels as a new column ('cluster') in the X Data Frame for further analysis or segmentation based on the identified clusters.**

**A close up of a computer code

Description automatically generated**

**this code processes an image by resizing, normalizing, and flattening it. It then uses a pre-trained KMeans clustering model to predict the cluster label for the processed image and prints the result.**

**A screenshot of a computer program

Description automatically generated**

---------------------------------------------------------------------------------------------------------------------------