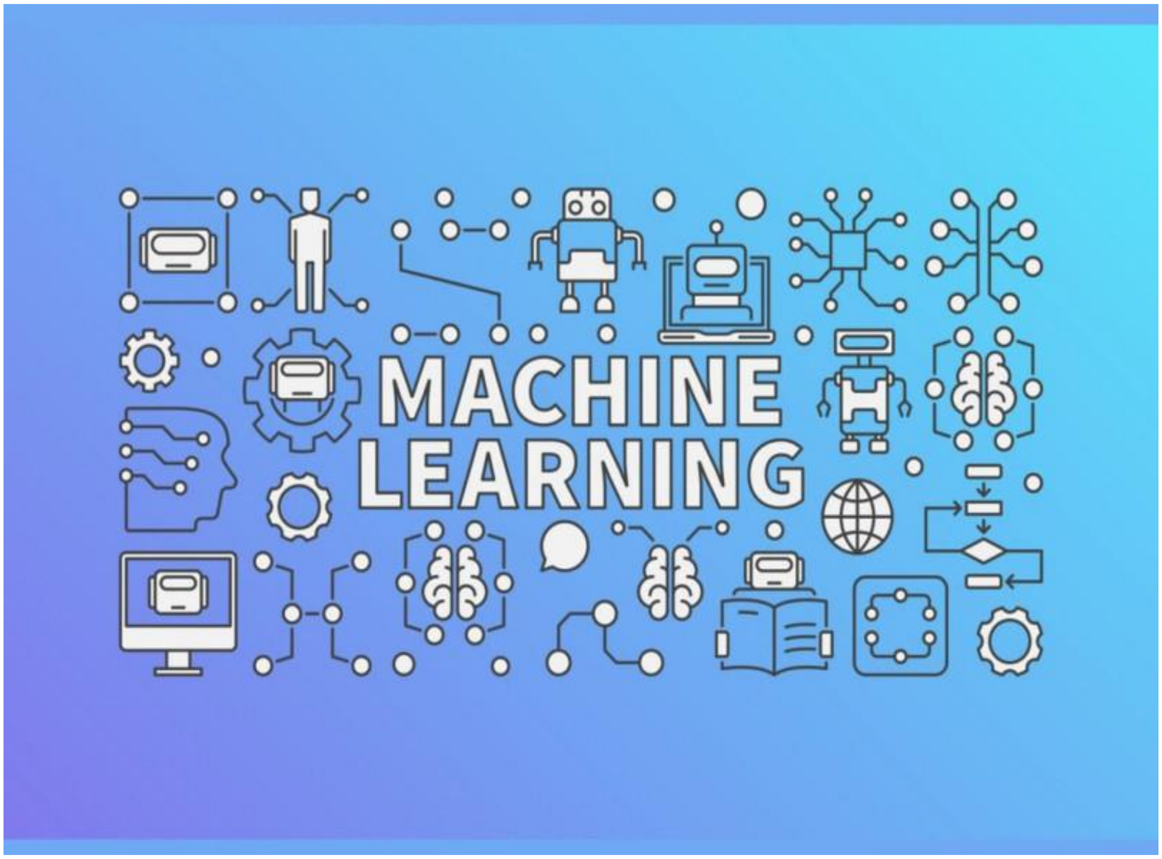
Machine learning models

Name	ID
Hala Khaled	42020007
Hagar Galal	42010084
Hadeer Elkady	42010436
Verina Gamal	42010044
Zeyad Mohamed	42010488



First project (Regression project)

- ## Introduction

  This project involves the analysis and modeling of property prices using a dataset containing various features related to housing in a specific region. The goal is to create predictive models for property prices and explore the relationships between different variables. The code is written in Python and utilizes various libraries such as Pandas, NumPy, Seaborn, Scikit-learn, Matplotlib, Pycaret, and XGBoost.

- ## Data Loading and Cleaning

  The initial steps involve loading the dataset and cleaning it to prepare for analysis.
  The dataset is loaded from a CSV file, and preliminary exploration is conducted. Duplicate values are removed, and missing values in the 'mrtdist' column are addressed by dropping the entire column. Additionally, several irrelevant columns are dropped to focus on relevant features. A scatter plot is created to visualize the geographical distribution of property prices.

- ## Exploratory Data Analysis (EDA)

  The exploratory data analysis section involves visualizing the data to gain insights into its distribution and relationships. Histograms, pair plots, correlation heatmaps, and bar plots are used to analyze numerical and categorical features. These visualizations provide a better understanding of the dataset's characteristics and help identify potential patterns.

- ## Data Transformation

  To prepare the data for modeling, log transformation is applied to the resale price column. This transformation is beneficial for linear models, enhancing their predictive performance.

- ## Model Building and Evaluation

  The main focus of the project is to build regression models to predict property prices. The Pycaret library is utilized for an automated setup of the regression task. Several regression algorithms, including Random Forest, Extra Trees, Gradient Boosting, Linear Regression, K-Nearest Neighbors, and XGBoost, are employed. The models are evaluated using various metrics such as R-squared, Mean Absolute Error (MAE), Mean Squared Error (MSE), and Root Mean Squared Error (RMSE).

- ## Hyperparameter Tuning

  Hyperparameter tuning is performed on selected models, specifically Random Forest and XGBoost, using GridSearchCV. This process involves searching for the best combination of hyperparameters to optimize the model's performance.

- ## Model Pipelines

  Pipeline structures are implemented for Random Forest, Extra Trees, Decision Tree, Linear Regression, K-Nearest Neighbors, and XGBoost models. These pipelines include data preprocessing steps such as label encoding and standard scaling, followed by model training and evaluation.

- ## Feature Importance

  Feature importance is analyzed for the best-performing model using Random Forest. The importance of each feature is visualized to understand which variables contribute the most to predicting property prices.

Model Tuning with Ridge Regression

Ridge Regression, a linear model with regularization, is employed with hyperparameter tuning using GridSearchCV. This step involves standardizing the features and searching for the best alpha value to optimize the model's performance.

Final Model Evaluation

The final models are evaluated on the test set using various metrics to assess their predictive capabilities. The R-squared value, Mean Absolute Error, Mean Squared Error, and Root Mean Squared Error are reported for each model.

Conclusion

The documentation concludes with a summary of the entire process, including the initial data exploration, cleaning, feature engineering, model building, hyperparameter tuning, and final evaluation. Recommendations for future improvements or additional analyses may be suggested based on the outcomes of the project.

Second project (Classification Problem)

- ## Introduction

  This project aims to develop a predictive model to classify breast cancer tumors as malignant (M) or benign (B) based on various features. The dataset used in this analysis is the Breast Cancer Wisconsin (Diagnostic) dataset.

- ## Data Exploration and Cleaning

  Explored the dataset's shape and found it to be (569, 33).

  Checked for duplicate rows and found none.

  Explored summary statistics for each column.

  Checked for missing values and found none.

  Dropped the 'id' and 'Unnamed: 32' columns as they are not informative.

  Encoded the target variable 'diagnosis' using Label Encoder.

- ## Data Visualization

  Utilized various visualization techniques to understand the data distribution and relationships.

  Plotted count plots, bar plots, box plots, histograms, and a pie chart.

  Created a correlation heatmap to visualize feature correlations.

Building Models

Trained and evaluated the performance of different classification models:

  - Logistic Regression
  - Decision Trees
  - Random Forest
  - Support Vector Machines (SVM)
  - K-Nearest Neighbors (KNN)
- Naive Bayes

Model Evaluation

Utilized metrics such as accuracy, precision, recall, and F1-score for model evaluation.

Plotted confusion matrices for each model to visualize true positive, true negative, false positive, and false negative predictions.

Model Pipelines

Created pipelines for each model, including data preprocessing steps such as scaling.

Hyperparameter Tuning

Performed hyperparameter tuning using GridSearchCV for the following models:

- Logistic Regression
- Decision Trees
- Random Forest
- Support Vector Machines (SVM)
- K-Nearest Neighbors (KNN)

- Gradient Boosting
- Naive Bayes

Model Comparison

Evaluated and compared the accuracy of each model after hyperparameter tuning.

Conclusion

Achieved the best accuracy with the Gradient Boosting model after hyperparameter tuning. The Gradient Boosting model had the highest accuracy among the models considered.

The project provides a comprehensive overview of data preprocessing, model building, and evaluation techniques in the context of breast cancer classification.

Third project (Unsupervised Learning)

Introduction

This project aims to perform customer segmentation using the KMeans clustering algorithm on a dataset containing information about customers. The dataset includes features such as age, income, gender, marital status, occupation, and settlement size. The goal is to group customers based on age and income to identify distinct segments within the customer base.

Data Exploration and Cleaning

Loaded the dataset and conducted initial exploration.

Checked the dataset's shape and dropped the 'ID' column.

Explored summary statistics, data types, missing values, and duplicates.

Utilized visualizations to gain insights into the distribution of age, income, gender, marital status, and occupation.

Data Visualization

Plotted histograms to visualize the distribution of age and income.

Used count plots to display the distribution of gender and marital status.

Created a scatter plot to explore the relationship between age and income.

Utilized box plots to understand income distribution across different occupations.

Constructed a correlation heatmap to examine correlations between numerical features.

- ## KMeans Clustering Mode

Applied KMeans clustering to segment customers based on age and income.

Utilized a pipeline for preprocessing and model building.

Defined a parameter grid for hyperparameter tuning using GridSearchCV.

Evaluated the model using Silhouette Score and Calinski-Harabasz Index.

- ## Model Evaluation

Achieved the best model from the pipeline using GridSearchCV.

Evaluated the model using Silhouette Score and Calinski-Harabasz Index to assess clustering quality.

- ## Results and Visualizations

Visualized the identified clusters and centroids on a scatter plot.

Presented the clusters in different colors and marked centroids with black stars.

- ## Conclusion

Successfully implemented customer segmentation using the KMeans clustering algorithm.

Identified distinct customer segments based on age and income.

Evaluated the model using silhouette and calinski-harabasz indices to measure clustering quality.

Fourth project (Image project)

Introduction

This project focuses on image classification using different machine learning and deep learning models. The goal is to classify images into three categories: pretty sunflower, rugby ball leather, and ice cream cone.

Data Collection

The data was collected using the Bing Image Downloader Python library, with a total of 60 images for each category. The images were downloaded and stored in separate folders for each category.

Data Preprocessing

Images were resized to a consistent shape (150x150x3) to ensure uniformity across the dataset. The pixel values were flattened to create a feature vector, and the target labels were assigned based on the categories.

Model Training and Evaluation

1. Decision Tree Classifier

- Decision Tree Classifier was trained on the dataset.

- Accuracy and confusion matrix were used to evaluate the model's performance.

2. Naive Bayes Classifier

- Gaussian Naive Bayes was employed for classification.

- Accuracy and confusion matrix were used for evaluation.

3. Random Forest Classifier

- Random Forest Classifier was trained using an ensemble of decision trees.

- Cross-validation scores and confusion matrix were analyzed.

4. Logistic Regression

- Logistic Regression was used with hyperparameter tuning.

- Cross-validation scores, accuracy, and confusion matrix were examined.

5. Support Vector Machine (SVM)

- SVM with a grid search for hyperparameter tuning was implemented.

- Accuracy, confusion matrix, and classification report were assessed.

• Deep Learning Model (TensorFlow/Keras)

- A neural network with a sequential model was created.

- The model was trained and evaluated using the TensorFlow/Keras library.

• Model Deployment

The best-performing model was saved using the Pickle library for future use. The model can be loaded and used for making predictions.

- ## Prediction Using the Deployed Model

  The deployed model was tested with new images fetched from URLs. The user can input an image URL, and the model will provide predictions for the given image.

- ## Conclusion

  The project explores various machine learning and deep learning models for image classification. Each model's performance is assessed, and the best-performing model is saved for future use.