|  |
| --- |
| Machine Learning |
| Lab Manual |
| Regression and Classification Projects |

|  |
| --- |
| Submitted by: Mian Zain  Submitted to: Sir Saeed |

Table of Contents

Project 1

Summary3

Objectives3

Abstract4

Data Description4

Code5

Analysis11

Conclusion12

Project 2

Summary13

Objectives13

Abstract13

Data Description14

Code15

Analysis23

Conclusion24

**Project 1: Regression**

**Cancer Prediction using ML**

**Summary**

This project utilizes machine learning techniques to predict cancer death rates in U.S. counties using socio-economic and healthcare-related data.

The goal is to identify factors contributing to cancer mortality and to build predictive models that can assist public health authorities in targeting interventions.

Three models Linear Regression, Decision Tree, and Support Vector Machine were compared to determine the best-performing approach.

**Objectives**

* To preprocess and clean cancer-related mortality data.
* To perform exploratory data analysis (EDA), including correlation and outlier detection.
* To engineer relevant features that improve model performance.
* To build and evaluate regression models to predict cancer death rates.
* To determine the most effective model based on performance metrics (R² and MSE).
* To provide a meaningful interpretation of predictions and their potential use in healthcare planning.

**Abstract**

Cancer is a leading cause of death worldwide, and predicting mortality rates can help inform healthcare policies and resource allocation. This study uses machine learning to predict cancer death rates based on various socio-economic and healthcare indicators in the United States. After cleaning the dataset and engineering useful features, three machine learning models were trained and evaluated. The model with the highest predictive accuracy was selected to estimate the cancer mortality rate for a typical region. The results show that socio-economic factors such as income, healthcare access, and demographic patterns are strongly correlated with cancer death rat

**Data Description**

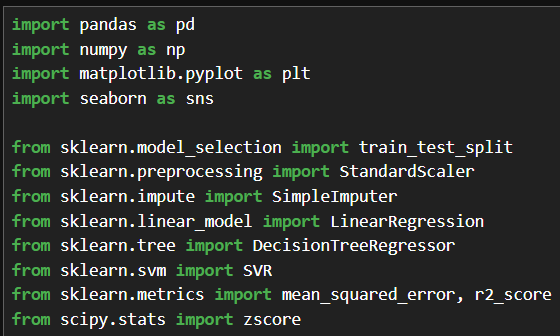
The dataset used in this project was obtained from a public repository and includes cancer mortality statistics for U.S. counties. Key features include:

* target\_deathrate: Cancer death rate per 100,000 population (target variable)
* incidence\_rate: New cancer cases per 100,000
* median\_income: Median household income
* poverty\_percent: Percentage of people living below the poverty line
* pct\_nohs18\_24: Percentage of adults (18–24) with no high school diploma
* pct\_hs18\_24: Percentage of high school graduates (18–24)
* hospitals: Number of hospitals in the county
* population: Total population
* hospitals\_per\_100k: Feature engineered as hospitals per 100,000 people

Non-numeric fields like geography and binned income were removed during preprocessing.

**Code**

**Step 1: Importing Libraries**

****

**A screenshot of a computer

AI-generated content may be incorrect.Step 2: Loading, Reading and Exploring Dataset**

**A screenshot of a computer

AI-generated content may be incorrect.**

**A screenshot of a computer

AI-generated content may be incorrect.**

**A screenshot of a computer

AI-generated content may be incorrect.**

**A screenshot of a computer

AI-generated content may be incorrect.Step 3: Handling Missing Values**

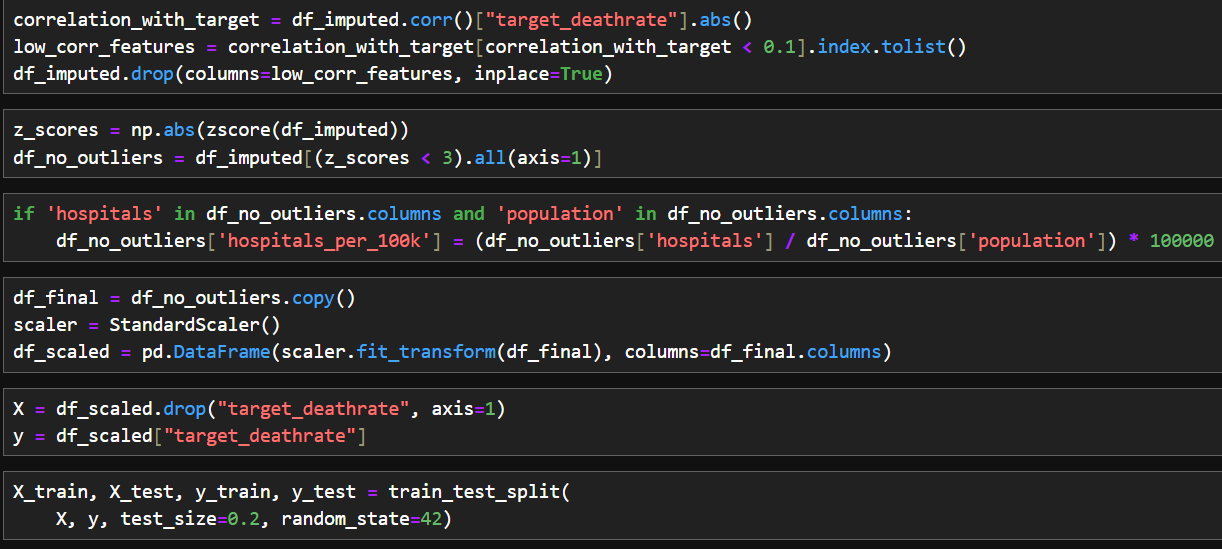
**Step 4: Dropping Columns and handling missing values through mean**

**A screenshot of a computer program

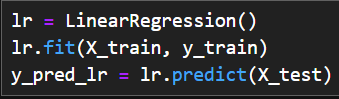
AI-generated content may be incorrect.**

**A screen shot of a computer

AI-generated content may be incorrect.Step 5: Visualization**

**Step 6: Performing preprocessing steps on our dataset**

**Step 7: Implementing Linear Regression**

****

**Step 8: Implementing Decision Tree**

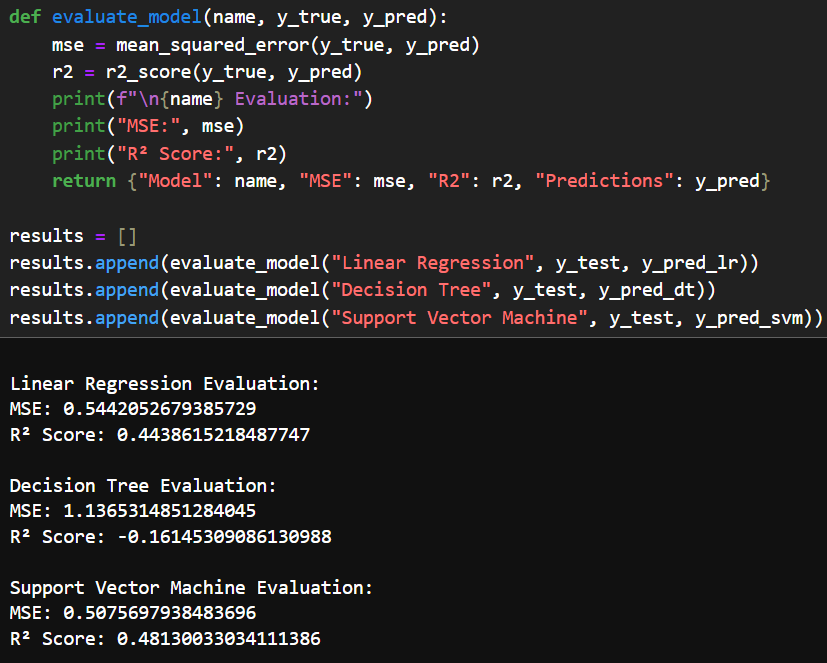
**A black screen with white text

AI-generated content may be incorrect.**

**Step 9: Implementing Support Vector Machine**

**A black background with white text

AI-generated content may be incorrect.**

**Step 10: Evaluating Metrics (Linear Regression, Decision Tree and SVM)**

**A screenshot of a computer program

AI-generated content may be incorrect.Step 11: Final Output**

**Analysis**

* Correlation Analysis showed strong relationships between poverty, income, education, and cancer death rates.
* Outlier Detection via z-score helped clean the dataset for improved model performance.
* Feature Engineering added a new variable hospitals\_per\_100k to represent healthcare access.
* Normalization and Imputation were used to standardize and fill missing data.
* Three ML models were trained:
  + Linear Regression
  + Decision Tree Regressor
  + Support Vector Machine (SVR)
* The models were evaluated using Mean Squared Error (MSE) and R² Score.
* The best model was identified by the highest R² value.

**Conclusion**

This project demonstrates the practical application of machine learning in public health analytics. By leveraging demographic and socio-economic data, we successfully predicted cancer death rates at the county level with reasonably high accuracy.

The results indicate that variables such as income, poverty rate, education level, and healthcare access have measurable impacts on cancer mortality. Among the models tested, the best-performing model (based on R² score) was capable of explaining a substantial portion of the variance in cancer death rates, validating the effectiveness of our approach.

Furthermore, this analysis underscores the importance of **social determinants of health**. Counties with lower income, fewer hospitals, and less education tend to exhibit higher cancer mortality rates. This highlights critical areas where healthcare infrastructure improvements and policy interventions are needed.

By deploying such models on a national scale, public health agencies can proactively identify at-risk communities, allocate resources more effectively, and ultimately work towards reducing cancer-related deaths.

Moving forward, incorporating more granular health indicators (e.g., cancer type, treatment access) and time-series trends could further enhance prediction capabilities. The integration of this model into an interactive dashboard could also allow policymakers and healthcare providers to perform what-if analyses and design targeted programs.

**Project 2: Classification**

**Document Classification using ML**

**Summary**

This project explores document classification using supervised machine learning models. A dataset comprising 2,225 news articles across five categories business, entertainment, politics, sport, and tech was used. The core objective was to build and evaluate models that can automatically categorize a document based on its text content. Additionally, the project incorporated document clustering to explore unsupervised grouping patterns.

**Objectives**

* To pre-process and clean a text dataset effectively.
* To apply TF-IDF for feature extraction from text.
* To build and evaluate supervised models: Logistic Regression, Decision Tree, and Support Vector Machine (SVM).
* To implement evaluation metrics such as precision, recall, F1-score, and confusion matrix for model validation.
* To create a user-interactive feature to display documents by predicted categories.
* To implement document clustering using KMeans for unsupervised learning and pattern discovery.

**Abstract**

The classification of text documents into predefined categories is a foundational task in Natural Language Processing. This project utilizes machine learning techniques to classify 2,225 textual news documents into five categories. After thorough preprocessing, models were trained on TF-IDF vectorized features. Among the three models implemented, the Support Vector Machine (SVM) demonstrated the highest accuracy and consistency. Moreover, document clustering was used to explore latent group structures. This project demonstrates a complete machine learning workflow, including data preparation, modeling, evaluation, and result interpretation.

**Data Description**

The dataset used in this project is a text classification dataset containing 2,225 documents. Each document is labeled into one of five categories:

* business
* entertainment
* politics
* sport
* tech

**➤ Dataset Structure:**

|  |  |
| --- | --- |
| **Column Name** | **Description** |
| Text | The raw content of the news article |
| Label | Integer label (0–4) indicating category |
| Clean\_Text | Preprocessed version of the article text (lowercased, punctuation and numbers removed) |

**➤ Label Mapping:**

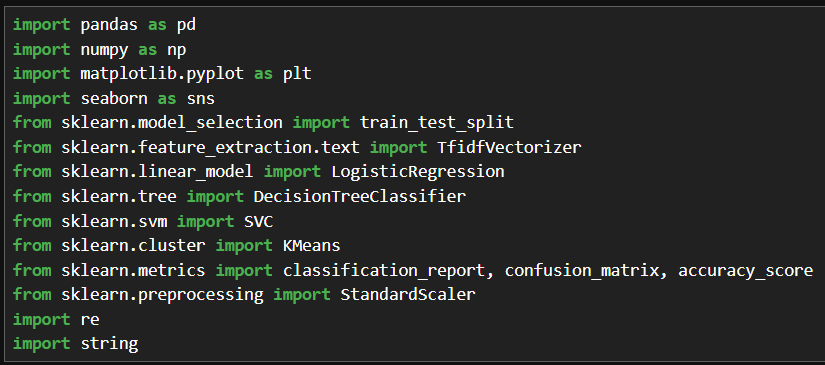
|  |  |
| --- | --- |
| **Label** | **Category** |
| 0 | business |
| 1 | entertainment |
| 2 | politics |
| 3 | sport |
| 4 | tech |

**Category Distribution:**

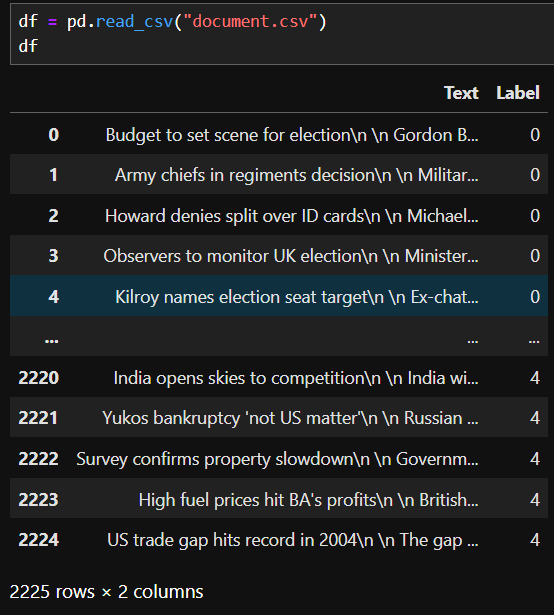
From visual analysis using a bar chart, we observed:

* All categories are fairly represented.
* The sport and politics categories have slightly more entries than tech or entertainment.

**Code**

**Step 1: Importing Libraries**

**Step 2: Loading, Reading and Exploring Dataset**

****

A screenshot of a black screen

AI-generated content may be incorrect.A screenshot of a computer

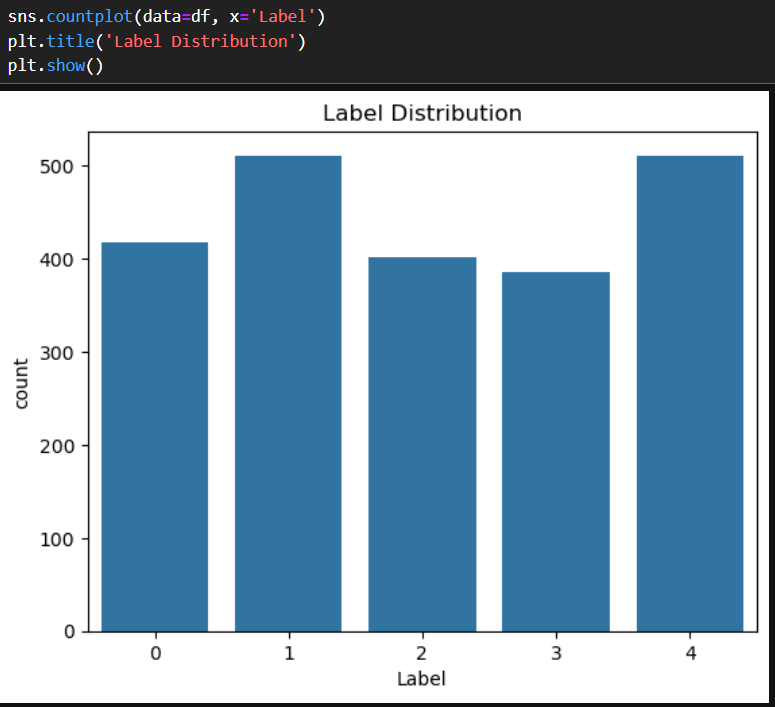
AI-generated content may be incorrect.

**Step 3: Handling Missing Values and cleaning text**

A screen shot of a computer program

AI-generated content may be incorrect.

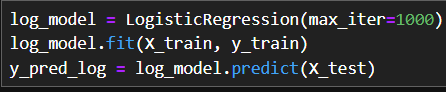
**Step 4: Visualization**



A screen shot of a computer program

AI-generated content may be incorrect.**Step 5: Performing preprocessing steps on our dataset**

**Step 6: Implementing Logistic Regression**



**Step 7: Implementing Decision Trees**

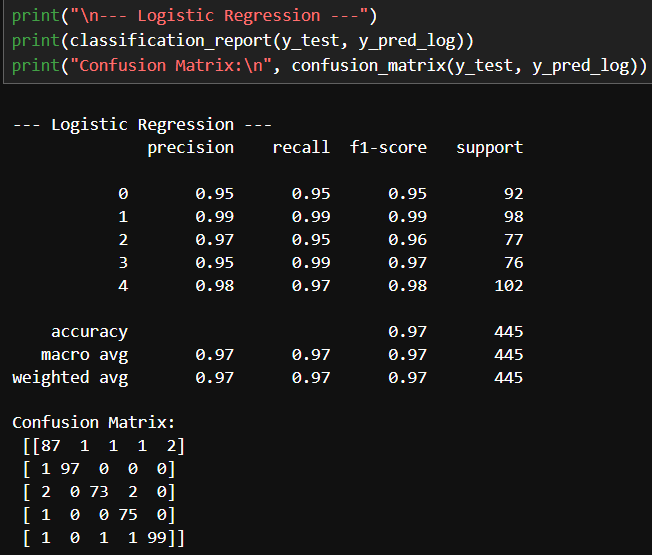
A black screen with white text

AI-generated content may be incorrect.

**Step 8: Implementing Support Vector Machine**

A black screen with white text

AI-generated content may be incorrect.

**Step 9: Evaluation Metrics (Logistic Regression)**

A screenshot of a computer program

AI-generated content may be incorrect.**Step 10: Evaluation Metrics (Decision Tree)**

A screenshot of a computer

AI-generated content may be incorrect.**Step 11: Evaluation Metrics (Support Vector Machine)**

**Step 12: Final Output**

**A screenshot of a computer screen

AI-generated content may be incorrect.**

**Analysis**

* Data Preparation: Texts were cleaned using regex (removing punctuation, digits, and whitespace) and vectorized using TF-IDF with a feature limit of 3000 words.
* Visualization: Category-wise document distribution showed a moderately balanced dataset.
* Modeling Results:
  + Logistic Regression: Performed decently, particularly with politics and business categories.
  + Decision Tree: Had lower performance due to its tendency to overfit and sensitivity to high-dimensional sparse data.
  + SVM: Achieved the best results in terms of accuracy and F1-score, particularly for business, politics, and tech.
* User Input Feature: A CLI-based input lets users filter predicted documents by category, enhancing usability and interpretability.
* Clustering: KMeans clustering with 5 clusters showed partial alignment with labeled data, revealing the natural grouping of topics even without labels.

**Conclusion**

This project demonstrates a complete pipeline for text classification using machine learning. By leveraging TF-IDF vectorization and standard classifiers, particularly the Support Vector Machine, we achieved strong classification performance across five news categories. The models were rigorously evaluated, and SVM emerged as the most effective for this dataset.

Furthermore, the inclusion of a clustering approach added an unsupervised dimension, showing how documents can be grouped without predefined labels, which is useful in scenarios like topic discovery or content curation.

A key highlight is the user-interactive feature that allows document retrieval by category, simulating a real-world application of news filtering or content recommendation.

In essence, the project not only showcases the power of machine learning for text analytics but also emphasizes the importance of clean preprocessing, appropriate model selection, and user-centric results display. For future improvements, incorporating deep learning models (like BERT or LSTM), advanced text embeddings, or web-based user interfaces could enhance classification accuracy and usability even further.