Machine Learning Projects

1.Book Recommendation System

2.Sonar -Rock vs Mine Prediction

3.House Price Prediction

4.Handwritten Digit Classification

Book Recommendation System

Overview

This project focuses on building a book recommendation system that suggests books to users based on their preferences and viewing history. The system uses collaborative filtering and content based filtering techniques to predict books. The dataset used for this project is sourced from the Kaggle.

Objective

1. To develop a system that can recommend books tailored to a user’s taste.
2. To compare different recommendation techniques.
3. To handle large dataset efficiently using Python libraries like: NumPy, Pandas, Scikit-Learn etc.
4. To improve user engagement by offering personalized movie suggestions

Technologies Used

1. Programming Languages:- Python
2. Libraries:- Pandas, NumPy, Scikit-learn, Matplotlib, Seaborn
3. Data Visualization:- Matplotlib, Seaborn
4. Data Sources:- Kaggle(Open Source Platform)

Recommendation Techniques

1. First we filtered the data according to the content
2. In second stage we filtered the data according to the users ratings and the book reviews .

Approach

1. Data cleaning and preprocessing
2. First I merge the data and create groups of book titles and ratings.
3. In second stage I handled the missing values and the duplicate values.
4. Filtering and Model Building.
5. Used Book-Title, Users and Ratings columns for filtering . in this first we find the average ratings of each book than we considered only those ratings which is given by actual book reader means who gave the ratings across the 50+ books and considered those books who have more than 200+ ratings.
6. We use the Scikit-learn metrics cosine similarities to find the similarities between these columns. After this we take the high similar books according to their similarities .

Sonar Rock vs Mine Prediction

Overview

This project involves building a machine learning model to classify sonar signals bounced off objects as either rock or mine. The dataset contains sonar returns obtained by bouncing sonar signals off different surfaces. The goal is to identify the type of object using these sonar signals.

Objective

The objective is to develop binary classification model that can distinguish between

R (Rock)

M(Mine)

Using sonar signal features. Each instance in the dataset represents a sonar signal

Technologies Used

1. Programming Languages:- Python
2. Libraries:- Pandas, NumPy, Scikit-learn, Matplotlib, Seaborn
3. Data Visualization:- Matplotlib, Seaborn
4. Data Sources:- Kaggle(Open Source Platform)

Prediction Techniques

1. Tried multiple classification techniques:
2. K-Nearest Neighbours (KNN)- KNN technique is used to fill the empty values in rows. By this technique the value is fill according to same rows data.
3. Logistic Regression- It is a statistical model that predicts the probability of binary outcome based on one or more input features.

Approach

1. Data cleaning and preprocessing
2. Loaded the dataset
3. Checked for missing/null values
4. Converted target labels (M,R) to binary (1,0)
5. Normalized the features using Standard Scaler

1. Filtering and Model Building.
2. Exploratory Data Analysis (EDA)- Visualized the distribution of classes
3. After done the EDA we performed the Logistic regression on train data and classify the signals. Our model is ready to predict the Rock vs Mine prediction.

House Price Prediction

Overview

The aim of this project is to build a machine learning model that can accurately predict house prices based on various features like the number of rooms, location, area, and more. Predicting real estate prices is a common regression problem and has practical applications in real estate, banking and property investment.

Objective

To predict the sales price of housing using a dataset with multiple numerical and categorical variables describing each property. The task is a **supervised regression problem.**

Technologies Used

1. Programming Languages:- Python
2. Libraries:- Pandas, NumPy, Scikit-learn, Matplotlib, Seaborn
3. Data Visualization:- Matplotlib, Seaborn
4. Data Sources:- Kaggle(Open Source Platform)

Prediction Techniques

1. Tried multiple classification techniques:
2. XGBoost Regression- decision tree is a type of regression techniques that used to predict the data. The accuracy of the decision tree is very high.
3. K-Nearest Neighbours (KNN)- KNN technique is used to fill the empty values in rows. By this technique the value is fill according to same rows data.

Approach

1. Data cleaning and preprocessing
2. Handled **missing values** using strategies

*Numerical: mean/median*

*Categorical: mode*

1. Converted categorical variables using One-Hot Encoding
2. Split data into training and test sets.
3. Filtering and Model Building.
4. Exploratory Data Analysis (EDA)- Visualized the distribution of classes
5. XGBoost Regression- Afte done the EDA we use the XGBoost for the prediction of house price.

Handwritten Digit Classification

Overview

This project aims to classify handwritten digits (0-9) using a machine learning model. We use image data where each image is a 28x28 grayscale picture of a digit. This task is a classic **multi-class classification** problem in computer vision and a common benchmark in machine learning.

Objective

To build a model that accurately identifies the digit present in a 28x28 pixel image using supervised learning algorithms. This model will classify each image into one of 10 classes (digits 0-9).

Technologies Used

1. Programming Languages:- Python
2. Libraries:- Pandas, NumPy, Scikit-learn, Matplotlib, Seaborn
3. Data Visualization:- Matplotlib, Seaborn
4. Data Sources:- Kaggle(Open Source Platform).

Prediction Techniques

1. Tried multiple classification techniques:
2. Convolutional Neural Network (CNN)- A CNN is a deep learning model designed specifically to process and recognize patterns in grid-like data
3. Activation Layer (ReLU)- Applies a non linear function to introduce non-linearity into the model.

Approach

1. Data cleaning and preprocessing
2. Loaded dataset using libraries like TensorFlow, Keras or scikit-learn
3. Normalized pixel values (from [0, 255] to [0, 1])
4. Flattened images (28x28 to 784) for non-CNN models
5. Split data into train and test sets.
6. Model Building
7. Logistic regression- We used the Logistic regression for the prediction of digits .