Scenario based ML Answer

6.Problem Type:Sentiment Analysis or Text Classification-goal is to determine the sentiment of customer reviews as either positive or negative.

Step1: Data Collection-Gather customer review data,including:

Review text

Product information(optional)

Rating(if available)

Step2: Data preprocessing:

Text Preprocessing:

Tokenization:Split the review text into individual words or tokens

Stopword removal:Remove common words like ‘the” “and” etc.that donot add much value

To the sentiment.

Stemming or Lemmatization: Reduce words to their bas form

Remove punctuation and special characters.

Step3:Feature Extraction

Bag- of- words(BoW):Represent the review textas a bag of words.

TermFrequency-Inverse Document Frequency(TF-IDF):

Weight the importance of words based on their frequency and rarity.

Sentiment lexicons:Use pre_trained dictionaries that map words to sentiments scores

Step4:Model Selection

NavieBayes

LogisticRegression

SVMs

Random Forest

Deep learning models like Convolution NeuralNetworks(CNNs)or Recurrent Neural

Networks*RNNs)

Step5:Model Training and Evaluation-

Split the dataset-Train set(80%) and Test set(20%)

Train the model

Evaluate the model using-

Accuracy

Precision

Recall

F1-score

ROC-AUC

Step6:Model Deployment-Deploy the trained model in a production where it can analyse customer reviews and determine the sentiment.

7.Customer claim driving insurance-binary Classification problem

Goal is to predict whether a customer is likely to file a claim in the next year (yes/no) based on their driving history and demographics.

Step1:Data Collection-

Driving-history (accidents,tickets,claims)

Demograpics(age,location,income)

Policy details(coverage,deductible)

Step2: Data Preprocessing-

Handle missing values :

Encode Categorical Variables:

Scale/normalize:

Step3:Featuring Engineering

Number of accidents in the past 5 years

Number of tickets in the past 3 years

Age and driving experience

Location-based features(e.g., urbanvs.rural)

Step4:Model Selection

Logistic Regression

Decision Trees

Random Forest

SVMs

Gradient Boosting

Neural Networks

Step5: Model Training and Evaluation

Split the dataset

Train the Model

Evaluate the model:

Accuracy

Precision

Recall

F1-score

ROC-AUC

Step6:Hyperparameter Tuning the model to optimize its performanace.

Step7:Model Deployment-it can predict the likelihood of a customer filling a claim

8.Recommendation System or Clustering-goal is to recommend movies to users

Based on their viewing preferences and watch history

Step1:Data ollection:

Watch history(movies watched, genres,ratings)

User demographics(age,location,etc)

User ratings or feedback(explicit or implicit)

Step2:Data preprocessing

Handle missing values

Encode Categorical variables

Scale/normalize

Step3:Feature Engineering

Extract relevant genres

Users favorite genres

Average rating given by the user

Time spent watching movies in different genres

Step4:Clustering;Choose suitable

K-means

Hierarchical clustering

DBSCAN

Determine the optimal number of clustering using techniques like the elbow method or silhouette score

Step5:Recommendation

Within each cluster,recommend movies that are popular or highly rated by users in that cluster

Use techniques like,

Collaborative filtering (user-based or item-based)

Content-based filtering

Hybrid approaches

Step6: Model Evaluation

Precision

Recall

F1-score

Mean Average Deployment

Normalized Discounted Cumulative Gain

Step7:Model Deployment

Deploy the recommendation system In a production

9.Predict the recovery time of patients-goal is to predict a continuous output variable(recovery time),based on input features(age,medicalhistory,and lifestyle habits)

Regression problem

Step1;Data Collection

Patient data,Age,

Medical history (pre-existing coditions ,previous surgery)

Lifestyle habits (smoking,exercise,diet)

Surgery details(type,complexity)

Recovery time(outcome variable)

Step2: Data preprocessing

Handle missing

Encode Categorical Variable

Scale/Normalize

Step3:Feature Engineering

Extract relevant features

Comorbidity index(measure of pre-existing conditions)

Lifestyle score(measure of healthy habits

Surgery complexity score

Step4:Model Selection

Linear Regression

Decision Trees

Random Forest

SVMs

Gradient Boosting

Step5:Model Training and Evaluation

Split the dataset

Train the model

Evaluate the model using metrics like

Mean Absolute ErrorMean Squared Error

R-squared

Step6:Hyperparameter Tuning

Step7:Model Deplioyment-it can predict recovery time

10. Students final exam score

Regression problem

Goal is to predict a continuous output variable(final exam score)

Based on input features(study hours,attendance,and,past academic performance)

Step1: Data Collection

Study hours(per week or per topic)

Attendance(percentage or number of classes attended)

Past academic performance(previous exam scores,grades)

Final exam score (output)

Step2:Data preprocessing

Handle missing values

Encodde categorical variables

Scale/Normalize

Step3: Feature Engineering

Average study hourse per week

Attendence rate

Past academic performance metrics(GPA,average grade)

Step4:Model selection

Linear Regression

Decision Trees

Random Forest

SVMs

Gradient Boosting

Step5:Model Training and Evaluation

Split the data

Train the model

Evaluate the model:

Mean Absolute Error

MeanSquared Error

R-squared

Step6:Hyperparameter Tuning-Tune the model

Step7:Model Deployment-Deploy the trained model

It can predict final exam scores for students

,