ML General Questions

1. Why we need to use data preprocessing in machine learning ?

- Data preprocessing is a crucial step in machine learning because raw data is often messy, incomplete, inconsistent, and not directly usable by models.

- Improves Model Accuracy

- Handles Missing or Incomplete Data(remove missing rows/imputing the values)

- Feature Scaling(standardization/ normalization(min/max scaling)

- Encodes Categorical Variables(one-hot encoding/label encoding)

- Balances Data for Fair Learning (under-sampling ((700,300) -> (300,300), over-sampling(700,300) -> (700,700) adding duplicate rows randomly, smote(creating new data using old data))

- Removes Outliers(outliers are those which changes the statistical values very highly(ex: for 1,2,3,4,5,6 the mean is 3.5 and for 1,2,3,4,5,100 the mean can be 20 so 100 is outlier) methods to detect is z-score , IQR, to see visually we can use box plots)

2.Why Use Feature Transformation?

- Feature transformation is the process of converting or modifying features (input variables) in a dataset to improve the performance, accuracy, or interpretability of a machine learning model.

1. Make patterns more learnable for models.
2. Normalize skewed distributions (e.g., exponential, log-normal).
3. Handle nonlinear relationships.
4. Reduce the impact of outliers.

3. What is Feature Engineering?

- Feature engineering is the process of creating, selecting, modifying,removing or transforming features from raw data to improve the predictive performance of machine learning models
- Ex: customer_id is not wanted,
- For creating use any mathematical formulas if available or use correlation heatmap to get relation

4. What is a model?

- In machine learning, a model is a mathematical representation of a system that learns patterns from data and makes predictions or decisions without being explicitly programmed (ex: y = mx + c is a model)

5. Types of model?

- Supervised : Training the model by using labeled(for every input it has desired output) and structured data
- Unsupervised : Training the model by using unlabeled data. (Ex: pattern matching)
- Reinforcement : Reinforcement Learning (RL) is a type of machine learning where an agent learns to make decisions by interacting with an environment, receiving rewards or penalties, and improving its strategy based on those outcomes. (ex: mainly used for gaming bots).

6. Models in supervised?

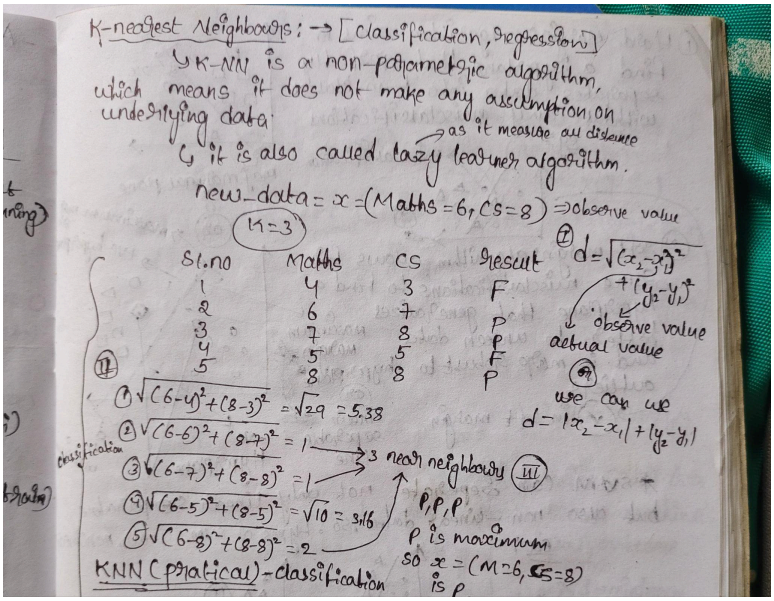
- Y(dependent variable) = mx(independent variable)+c

- Regression models(used when the output is continuous): simple linear regression(single input,single output), multi-linear regression(multi-input, single output)

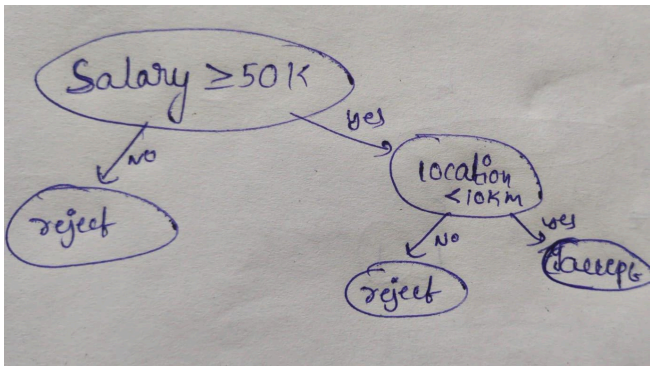
- Classification models(used when the output is categorical):
    - For Non-linear separable data: SVM(support vector machine), knn(k nearest neighbor), navis-base, decision tree.
    - For linear separable data: logistic regression , svm.

Classification models can also be used for regression data but vice-versa is not true

7. How knn works?



8. Working of decision tree?



9. advantages and disadvantages of simple linear regression(use linear equation) models?

Advantages:

- Easy to understand and interpret
- Fast to train and computationally efficient
- Works well with linearly related data
- Serves as a foundation for more complex models
- Useful for quick predictions and forecasting

Disadvantages:
- Assumes a linear relationship
- Sensitive to outliers
- Assumes independence between variables and errors
- Limited to one independent variable
- Requires constant variance in errors (homoscedasticity)

10. advantages and disadvantages of multi linear regression models?

Advantages:
- Can model relationships with multiple independent variables
- More accurate than simple linear regression when multiple factors influence the outcome
- Allows feature importance analysis
- Captures complex data patterns better than simple regression
- Scalable to large datasets with multiple features

Disadvantages:
- Assumes linear relationship between predictors and target
- Prone to multicollinearity if predictors are correlated
- Requires more data to train effectively
- Sensitive to outliers
- Interpretation becomes harder with many variables

11. advantages and disadvantages of svm(uses hyper-plane equation) models?

Advantages:
- Effective in high-dimensional spaces
- Works well for both linear and non-linear data (with kernel trick)
- Robust to overfitting, especially in high-dimensional space
- Memory efficient due to use of support vectors
- Versatile with different kernel functions (linear, RBF, polynomial, etc.)

Disadvantages:
- Slow training time for large datasets
- Requires careful tuning of parameters and kernel choice
- Less effective on noisy or overlapping data

- Hard to interpret and visualize compared to simpler models
- Not suitable for very large datasets due to computational complexity

12. advantages and disadvantages of knn(also known as lazy learning) models?

Advantages:
- Simple to understand and implement
- No training phase; good for real-time applications
- Works well with multi-class problems
- Makes no assumptions about data distribution (non-parametric)
- Naturally adapts to complex decision boundaries

Disadvantages:
- Slow prediction time for large datasets
- Sensitive to irrelevant or redundant features
- Requires feature scaling for accurate results
- Performance depends heavily on the choice of K and distance metric
- Memory-intensive as it stores the entire training dataset

13. What is naive-bayes and advantages and disadvantages of naive-bayes models?

Naive Bayes is a probabilistic classifier based on **Bayes' Theorem** with the assumption that all features are **independent** given the class. It calculates the probability of each class using the **prior probability** and the **likelihood of features**, selecting the class with the highest posterior probability.

Advantages:

- Simple and fast to train and predict

- Works well with high-dimensional data

- Effective for text classification problems (e.g., spam detection)

- Requires relatively small amount of training data

- Not sensitive to irrelevant features

Disadvantages:

- Assumes features are independent (rarely true in real data)

- Struggles with correlated features

- Can perform poorly if conditional independence assumption is violated

- Zero-frequency problem: assigns zero probability to unseen words (can be fixed with smoothing)

- Less flexible compared to more complex models

13. What is logistic regression and advantages and disadvantages?

Logistic Regression is a classification algorithm that uses the **sigmoid function** to model the probability that a given input belongs to a particular class. It optimizes the model using **maximum likelihood estimation** and outputs values between 0 and 1.

Advantages:
- Simple and easy to implement
- Works well for linearly separable data
- Provides probabilistic interpretation

Disadvantages:
- Assumes linear relationship between features and log-odds
- Struggles with non-linear data
- Can underperform with complex relationships or noisy data

14. What is decision tree and advantages and disadvantages?

Decision Tree is a non-linear model that splits data based on feature values using **if-else rules**, building a tree structure where each node represents a decision based on a feature that maximizes **information gain** (e.g., using Gini impurity or entropy).
Advantages:
- Easy to understand and interpret
- Handles both numerical and categorical data
- No need for feature scaling or normalization

Disadvantages:
- Prone to overfitting, especially with deep trees
- Unstable to small variations in data
- Can be biased if some classes dominate

15. Models in unsupervised?
- Clustering : Diving given data into cluster(groups)
    - Models; k-means clustering, hierarchical clustering
- Association : arrange the given data in layers(ex: supermarket)

16. What is model evaluation?
- Model evaluation is the process of assessing how well a machine learning model performs on data, especially unseen/test data. It helps you understand how accurate, reliable, or useful the model is for its intended task
- Regression model evaluation: mse, rmse(root mean square error), mae(mean absolute error), r-squared
- Classification model evaluation: confusion matrix(recal,presion,f1), roc, z-score
    - Pression = TP/(TP + FP)
    - Recall = TP/(TP + FN)
    - F1 = 2 * (precision * recall) / (precision + recall)

17. what is overfitting and undet-fitting?

Overfitting happens when a machine learning model learns the training data too well, including the noise and small details that don’t actually matter. As a result, it

performs very well on the training data but poorly on new, unseen data. It's like memorizing answers instead of understanding the material.

Underfitting happens when a model is too simple and doesn't learn enough from the training data. It performs poorly on both the training data and new data because it hasn't captured the important patterns. It's like not studying enough and failing all the tests.

18. What is hyper-parameter tuning?

Hyperparameter tuning is the process of finding the best set of **hyperparameters** for a machine learning model to improve its performance.

Model parameters: y = mx + c here m and c are model parameters

Hyper parameters: parameters used to set when we are training the model

Ex: dt = decisiontreeclasiifier(criterion = "enthalpy") here criterion is hyper parameter.

19. Types of hyperparameter tuning?

- Gridsearchcv, random search cv

Where we Use of Sklearn:

1.In preprocessing for Feature scaling:

```
From sklearn.preprocessing import MinMaxscaler

m=MinMaxscaler()

m.fit(dataset["coloum_name"])=m.transform("coloum_name")
```

2.In preprocessing for function transformation:

From sklearn.preprocessing import import funtiontransformer

ft=funtionTransformer(func=np.log1p)

ft.fit(dataset[“colom_name”])

ft.transfor(dataset[“colom_name”])

3. in train and test split:

From sklearn.model_selection import train_test_split
x_train,x_test,y_train,y_test=train_test_split(input_data(x),output_data(y))

4. in training model:

From sklearn.linear_model import linerRegression

lr=LinearRegression()

lr.fit(x_train,y_train)

lr.score(x_test,y_test)

Where we Use Seaborn and Matplotlib is for data visualization:

Import seaborn as sn

Import matplotlib.pyplot as plt

Ex:

sn.barplot(dataset[“colom_name”],dataset[“colm_name1”])

plt.show()