

Assignment $\xrightarrow{\text{ML}}$

Q.1 what is Parameter.

Ans It is a configuration variable that is internal to the model & whose value is estimated from the data. eg:- linear regression.
the slope & intercept are parameters.

Q.2 what is correlation?

Correlation measures statistical relationship b/w two variable.

It tells you how one variable change with respect to another. Value range from -1 to 1.

- +1 Perfect positive correlation
- 0 No correlation
- -1 Perfect negative correlation

Q.3 What does -ve correlation mean?

Ans A -ve correlation means that as one variable increases, the other decreases. eg:-

temp↑, the sale of winter jacket may decrease.

Q.4 what is machine learning. what are the main components in machine learning?

Ans ML is subset of AI that enables system to learn from data and improve over time without being explicitly programmed.

- Dataset (Input data)
- Features (Independent variables)

11

Q.5 How does loss value help in determining whether the model is good or not?

Ans Loss value indicates the difference b/w the predicted output & the actual value. A low loss means the model's prediction are close to the actual values.

A High loss shows poor model performance.

Q.6 What are continuous & categorical variables.

Ans

- Continuous variables: Numeric values that can take any value (e.g. age, height, income)

- Categorical variables:

Represent categories or groups
e.g. gender, color, country.

Q.7 How do we handle categorical variables in Machine learning?
What are common techniques.

Ans • Label encoding:

Convert categories into numeric codes

- One-Hot encoding: Create binary column for each category.
- Ordinal encoding: For categorical data with an inherent order.

Q.8 What is `sklearn.preprocessing`?

- `sklearn.preprocessing` is a module in Scikit-learn that provides preprocessing functions like:
 - Scaling (`StandardScaler`, `MinMaxScaler`)
 - Encoding (`LabelEncoder`, `OneHotEncoder`)
 - Imputation (`SimpleImputer`)

Q.9 How do we split data for model fitting (training & testing) in python

from `sklearn.model_selection import train_test_split`

$x_train, x_test, y_train, y_test = \text{train_test_split}(X, y, test_size=0.2, random_state=42)$

Q.10 How do you approach a machine learning problem?

- Define the problem.
- Collect data
- Perform EDA
- Preprocess data.
- Split data into training / testing.
- Choose a model.
- Train the model.
- Evaluate the model.
- Fine tune
- Deploy.

Q.11 Why do we have perform EDA before fitting a model to the data?

- EDA (Exploratory Data Analysis) →
 - Understand the data distribution
 - Detect outliers
 - Identify missing values
 - Discover patterns & relationships
 - choose appropriate feature

Q.12 How can you find correlation b/w variables in python?

Q.10 How do you approach a machine learning problem?

- Define the problem.
- Collect data
- Perform EDA
- Preprocess data.
- Split data into training / testing.
- Choose a model.
- Train the model.
- Evaluate the model.
- Fine tune
- Deploy.

Q.11 Why do we have perform EDA before fitting a model to the data?

- EDA (Exploratory Data Analysis) →
 - Understand the data distribution
 - Detect outliers
 - Identify missing values
 - Discover patterns & relationships
 - choose appropriate feature

Q.12 How can you find correlation b/w variables in python?

Import pandas as pd.

df.corr()

(Q1)

Import Seaborn as sns

sns.heatmap (df.corr().annot = True)

Q.13 what is causation? Explain difference b/w correlation and causation with an example.

- Causation: One variable directly affect another
- Correlation: Two variables are related, but not necessarily cause effect

e.g.: Ice-Cream Sales ↑ with Summer

↳ e.g.: More heat causes for ice cream sale

Q.14 What is an optimizer?

Ans

An optimizer is an algorithm used in Machine Learning and Deep Learning, to adjust Model parameters (like weights & bias) to minimize the loss function.

Q.18 Types of optimizers

Aw i) Gradient Descent.

(ii) Stochastic Gradient Descent (SGD)

(iii) Mini-batch Gradient Descent

(iv) Adam

(v) RMSprop.

(vi) Adagrad

Q.16 What is Sklearn linear model?

Ay Sklearn.linear_model is a module in Scikit-learn that provides linear models for regression and classification.

→ LinearRegression

→ LogisticRegression

→ Ridge, Lasso

→ SGDClassifier, SGDRegressor

Q.17 What does model.fit() do?

→ model.fit(x,y) trains the model using the feature X and target values Y.

Arguments

$X \rightarrow$ 2D array of shape (n_samples, n_features)
 $y \rightarrow$ 1D array of target values

`model.fit(X_train, y_train)`

Q.18 What does `model.predict()` do?

→ `model.predict(X)` returns predicted values for given input data X based on the trained model.

$y_{pred} = \text{model.predict}(X_{test})$

Q.17 What is feature Scaling? Why is it important?

→ Feature Scaling transforms input features to the same scale or range

Why -
• prevent feature with larger scale from dominating

- Improves convergence of gradient descent
- Necessary for distance-based model (KNN, SVM).

Q.2 How do we perform Scaling in Python?

Ans: Using `sklearn.preprocessing`

Standard Scaling (mean 0, std=1)

```
from sklearn.preprocessing import StandardScaler
scaler = StandardScaler()
X_scaled = scaler.fit_transform(X)
```

Q.21 what is `sklearn.preprocessing`?

A module in scikit-learn provide functions to preprocess data —

- StandardScaler — Standardize feature
- MinMaxScaler — Normalize feature
- LabelEncoder — Encode labels
- OneHotEncoder — One-hot encode categorical feature
- PolynomialFeatures — Generate Polynomial features

Q.22 How to split Data for Model Fitting (Train / Test Split).

→ `from sklearn.model_selection import
train_test_split`

`x_train, x_test, y_train, y_test =`

`train_test_split(x, y, test_size=0.2, random_state=42)`

Q.23 what is Data Encoding?

→ Data encoding transform categorical variables into a numeric format.

Type :-

- Label Encoding: Assigns numeric labels

(eg Male → 0, Female → 1)

- One-Hot encoding: Create binary columns for each category

Eg:-

```
from sklearn.preprocessing import  
OneHotEncoder
```

```
encoder = OneHotEncoder(sparse=False)
```

```
X_encoded = encoder.fit_transform(X[[category]])
```