National University of Computer & Emerging Sciences, Karachi

FALL 2024, LAB MANUAL – 08

LINEAR REGRESSION

COURSE CODE :	**AL3002**
INSTRUCTOR :	**Usama Bin Umar**

OBJECTIVE

1. Implementation of Linear Regression
2. Types of LR
3. Linear Regression Coefficients

REGRESSION:

Unlike classification, regression has continuous values in labels/class.
The term regression is used when you try to find the relationship between variables. In Machine Learning, and in statistical modeling, that relationship is used to predict the outcome of future events.

LINEAR REGRESSION

Linear regression is a type of statistical analysis used to predict the relationship between two variables. It assumes a linear relationship between the independent variable and the dependent variable, and aims to find the best-fitting line that describes the relationship. The line is determined by minimizing the sum of the squared differences between the predicted values and the actual values.

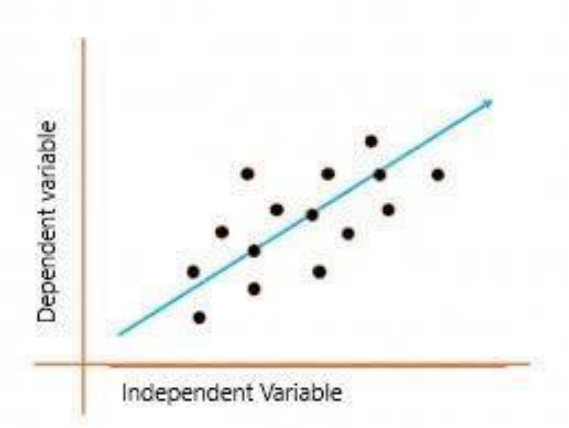
TYPES OF REGRESSION

- Linear Regression
- Multiple Linear Regression
- Multinomial Linear Regression
- Ordinal Regression
- Logistic Regression (it handles classification problems)
- Ridge Regression
- Lasso Regression

SIMPLE LINEAR REGRESSION

In a simple linear regression, there is one independent variable and one dependent variable. The model estimates the slope and intercept of the line of best fit, which represents the relationship between the variables. The slope represents the change in the dependent variable for each unit change in the independent variable, while the intercept represents the predicted value of the dependent variable when the independent variable is zero.

Linear regression is a quiet and the simplest statistical regression method used for predictive analysis in machine learning. Linear regression shows the linear relationship between the independent(predictor) variable i.e. X-axis and the dependent(output) variable i.e. Y-axis, called linear regression. If there is a single input variable X (independent variable), such linear regression is called **simple linear regression.**



The graph above presents the linear relationship between the output(y) and predictor(X) variables. The blue line is referred to as the best-fit straight line. Based on the given data points, we attempt to plot a line that fits the points the best.

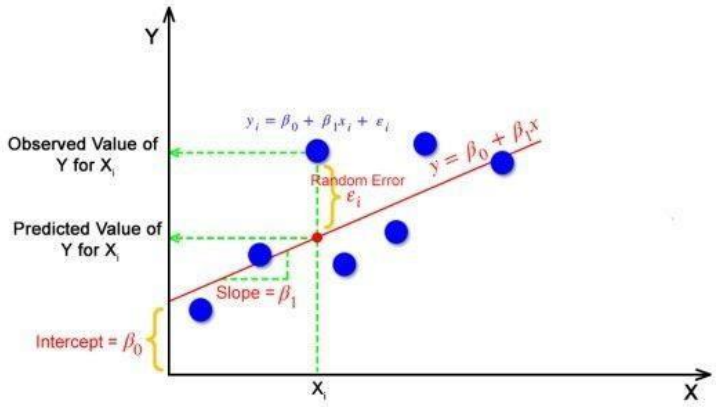
To calculate best-fit line linear regression uses a traditional slope-intercept form which is given below,

$Y_i = \beta_0 + \beta_1 X_i$

where Y_i = Dependent variable, β_0 = constant/Intercept, β_1 = Slope/Intercept, X_i = Independent variable.

This algorithm explains the linear relationship between the dependent(output) variable y and the independent(predictor) variable X using a straight line

Y= B0 + B1 X.



The goal of the linear regression algorithm is to get the best values for B0 and B1 to find the best fit line. The best fit line is a line that has the least error which means the error between predicted values and actual values should be minimum.

HOW TO CALCULATE ERROR IN REGRESSION?

Simple Linear Regression

```
from sklearn.linear_model import LinearRegression
LR = LinearRegression()
ModelLR = LR.fit(x_train,y_train)
```

```
PredictionLR = ModelLR.predict(x_test)
print(PredictionLR)
```

	YearsExperience	Salary
0	1.1	39343.0
1	1.3	46205.0
2	1.5	37731.0
3	2.0	43525.0
4	2.2	39891.0

Based on Year of Experience we are predicting the salary of a person.

Actual

```
====================ACtual Answers=================
[112635.  67938. 113812.  83088.  64445.  57189. 122391. 109431.  56957.]
```

Prediction

```
[115573.62288352  71679.93878159 102498.90847018  75415.57147111
  55803.4998511   60473.04071301 122110.98009019 107168.44933209
  63274.76523015]
```

Calculating Testing Accuracy

Random Error (Residuals)

In regression, the difference between the observed value of the dependent variable(yi) and the predicted value(predicted) is called the residuals.

εi = ypredicted – yi

where ypredicted = B0 + B1 Xi

Testing Accuracy

```
from sklearn.metrics import r2_score
print("====================LR Testing Accuracy============")
teacLR=r2_score(y_test,PredictionLR)
testingAccLR=teacLR*100
print(testingAccLR)
```

MULTIPLE LINEAR REGRESSION

Multiple linear regression is a technique to understand the relationship between a single dependent variable and multiple independent variables.

The formulation for multiple linear regression is also similar to simple linear regression with

The small change that instead of having one beta variable, you will now have betas for all the variables used. The formula is given as:

$Y = B0 + B1X1 + B2X2 + \ldots + BpXp + \varepsilon$

Considerations of Multiple Linear Regression

Overfitting: When more and more variables are added to a model, the model may become far too complex and usually ends up memorizing all the data points in the training set. This phenomenon is known as the overfitting of a model. This usually leads to high training accuracy and very low test accuracy.
Multi collinearity: It is the phenomenon where a model with several independent variables, may have some variables interrelated.
Feature Selection: With more variables present, selecting the optimal set of predictors from the pool of given features (many of which might be redundant) becomes an important task for building a relevant and better model.

Consider **student performance** dataset, multiple linear regression can be used to predict a student's academic performance based on multiple factors. For instance, consider a dataset that includes variables such as the number of hours studied per week, previous exam scores, and the amount of sleep hours per night etc. Using multiple linear regression, one can create a predictive model that takes into account all these variables simultaneously. The model might find that a student's performance based on these attributes

Hours Studied	Previous Scores	Extracurricular Activities	Sleep Hours	Sample Question Papers Practiced	Performance Index
7	99	Yes	9	1	91.0
4	82	No	4	2	65.0
8	51	Yes	7	2	45.0
5	52	Yes	5	2	36.0
7	75	No	8	5	66.0

Multiple Linear Regression

```
from sklearn.linear_model import LinearRegression
LR = LinearRegression()
ModelLR = LR.fit(x_train,y_train)
```

```
PredictionLR = ModelLR.predict(x_test)
print(PredictionLR)
```

LINEAR REGRESSION COEFFICIENTS

```
slope = ModelLR2.coef_
intercept = ModelLR2.intercept_

# Print the coefficients
print("Slope (Coefficient):", slope)
print("Intercept:", intercept)
```

In the context of linear regression, the equation for a straight line is represented as:
$y=mx+b$

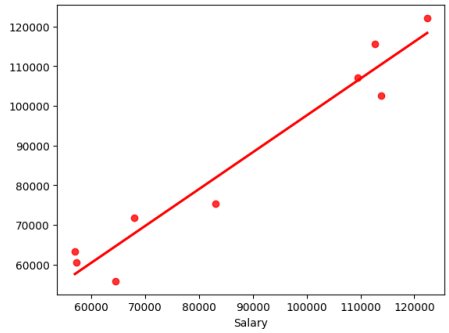
Where:

- **y** is the dependent variable (the variable you are trying to predict),
- **x** is the independent variable (the variable you are using to make predictions),
- **m** is the slope of the line (also known as the coefficient of the independent variable), and
- **b** is the y-intercept (the value of y when x is 0).

REGRESSION PLOT

Regplot is particularly useful when you want to visualize the relationship between two variables and understand how well a linear regression line fits the data. It provides a quick and easy way to assess the linear relationship between the variables.

```
sns.regplot(x=y_test,y=PredictionLR,ci=None,color ='red')
```



LINEAR REGRESSION ASSUMPTIONS

Linear regression relies on several assumptions to provide accurate and reliable results. Understanding and validating these assumptions are crucial when using linear regression models. Here are the key assumptions of linear regression:

LINEARITY:

Assumption: There is a linear relationship between the independent variables (features) and the dependent variable (target).

Why it's Important: Linear regression models assume that the relationship between variables is linear. Non-linear relationships might require different modeling approaches.

INDEPENDENCE:

Assumption: Observations are independent of each other. The value of the dependent variable for one observation should not depend on the values of the independent variables for other observations.

Why it's Important: Independence ensures that the model coefficients are not biased due to interdependencies between observations.

HOMOSCEDASTICITY (CONSTANT VARIANCE):

Assumption: The variance of the residuals (the differences between observed and predicted values) should remain constant across all levels of the independent variables.

Why it's Important: Homoscedasticity ensures that the variability in the residuals is consistent, indicating that the model's predictions are equally accurate for all levels of the independent variables.

NORMALITY OF RESIDUALS:

Assumption: The residuals should be approximately normally distributed.

Why it's Important: Normality of residuals is essential for hypothesis testing and constructing confidence intervals. If the residuals are not normally distributed, it might affect the accuracy of statistical inferences.

NO PERFECT MULTICOLLINEARITY

Assumption: The independent variables should not have a perfect linear relationship with each other.

Why it's Important: Perfect multicollinearity makes it impossible to determine the individual effect of each variable on the dependent variable. High multicollinearity can lead to unstable coefficients and reduced model interpretability.

NO AUTOCORRELATION:

Assumption: The residuals should not exhibit autocorrelation, meaning there should be no pattern in the residuals over time or across observations.

Why it's Important: Autocorrelation indicates that the errors are not independent, violating the assumption of independence. It's crucial for time-series data or any data with a temporal component.

ADDITIVITY:

Assumption: The effect of a change in an independent variable on the dependent variable is consistent regardless of the values of other variables.

Why it's Important: Additivity ensures that the total effect of multiple independent variables on the dependent variable is the sum of their individual effects.

Validating these assumptions is essential to ensure the reliability and interpretability of a linear regression model. Various diagnostic tools, such as residual plots and statistical tests, can help assess whether these assumptions hold for a given dataset.

POLYNOMIAL LINEAR REGRESSION

Polynomial regression is a type of linear regression where the relationship between the independent variable \(x\) and the dependent variable \(y\) is modeled as an \(n\)th degree polynomial. In other words, polynomial regression fits a curved line to the data instead of a straight line. This allows capturing more complex patterns in the data. The equation of polynomial regression of degree \(n\) can be expressed as:

[y = b_0 + b_1x + b_2x^2 + \ldots + b_nx^n + \varepsilon \]

Where:

(y \) is the dependent variable (target).

(x \) is the independent variable.

(b_0, b_1, \ldots, b_n \) are the coefficients to be estimated from the data.

(\varepsilon \) represents the error term.

KEY POINTS ABOUT POLYNOMIAL REGRESSION:

Degree of the Polynomial (\(n\))

The degree of the polynomial determines the complexity of the model. Higher degrees can fit the data more accurately but can lead to overfitting.

It's important to choose an appropriate degree based on the complexity of the underlying relationship in the data. This is often done using techniques like cross-validation.

Non-Linear Relationship:

Polynomial regression can capture non-linear relationships between variables. For example, in a quadratic regression (\(n=2\)), the relationship is parabolic.

Overfitting:

Using a very high degree polynomial can lead to overfitting, where the model fits the noise in the data rather than the underlying pattern. Regularization techniques can be applied to mitigate overfitting.

Data Transformation

Before applying polynomial regression, it's crucial to understand the nature of the data. Sometimes, transforming the variables (like taking logarithms) can make the relationship more linear, and simple linear regression might be sufficient.

Interpretability:

Interpretability becomes challenging with higher degree polynomials, as the relationship becomes more complex and difficult to visualize.

Advantages of Polynomial Regression:

Flexibility: Can capture a wider range of relationships.

Higher Accuracy: Can provide a more accurate fit for certain types of data.

Disadvantages of Polynomial Regression:

Overfitting: Prone to overfitting, especially with higher degree polynomials.

Interpretability: Harder to interpret and explain compared to linear regression.

Polynomial regression is a valuable tool when dealing with non-linear relationships, but careful consideration of the degree and potential overfitting is essential to its effective application.

Polynomial Linear Regression

```
from sklearn.preprocessing import PolynomialFeatures
from sklearn.linear_model import LinearRegression

poly = PolynomialFeatures(degree=3, include_bias=True)
x_train_trans = poly.fit_transform(x_train)
x_test_trans = poly.transform(x_test)
#include bias parameter
lr = LinearRegression()
lr.fit(x_train_trans, y_train)
y_pred = lr.predict(x_test_trans)
print(r2_score(y_test, y_pred))
```

TASKS

TASK 1: You are working for a car dealership company aiming to enhance its pricing strategy. The company has collected a vast dataset containing various attributes of cars, including features such as mileage, horsepower, number of doors, brand, model year, and more. Your task is to develop a robust machine learning model to predict car prices accurately based on these features.

Download the dataset

- Perform necessary EDA and Data Wrangling
- Implement Linear Regression
- Evaluate LR using different metrics
- Plot training and testing results

TASK 2:

Scenario:

You are provided with a dataset and your task is to determine whether applying linear regression is appropriate. Remember the fundamental assumptions of linear regression while making your decision.

Dataset Description:

The dataset contains numerical values representing various factors related to monthly electricity consumption in households. The features include the number of residents, average income, and the age of the house. The target variable is the monthly electricity consumption in kilowatt-hours.

Task Instructions:

Review the Dataset:

Familiarize yourself with the dataset, noting the features and the target variable.

Assumption Consideration:

Reflect on the key assumptions of linear regression: linearity, independence, homoscedasticity, normality of residuals, and absence of perfect multicollinearity.

Think about how these assumptions relate to the dataset. Consider whether the data might meet these assumptions or present challenges.

Decision Making:

Based solely on your understanding of the assumptions and by doing analysis, decide whether linear regression is suitable for this dataset. Provide a brief rationale for your decision.

Justification:

If you decided that linear regression is appropriate, outline why you believe the dataset meets the assumptions. If you decided against linear regression, briefly explain which assumptions you think the dataset violates.

TASK 3

Write an understanding of assumption in a word file of every classifier that you have learned in previous labs.