Unit V Impact of Machine Learning in BI

5.1 Regression: Regression problems, Evaluation of regression models, Linear regression..

Regression: An Overview

Regression is a type of **supervised machine learning** used to model the relationship between a dependent (target) variable and one or more independent (predictor) variables. The goal of regression is to predict a continuous output based on input variables.

1. Regression Problems

Regression problems involve predicting a continuous output variable from a given set of input variables. These problems can be as simple as predicting house prices based on features like size and location, or more complex, such as predicting stock prices based on various economic indicators.

Common Types of Regression Problems:

- **Simple Linear Regression**: Predicting a continuous variable using one independent variable.
- **Multiple Linear Regression**: Predicting a continuous variable using multiple independent variables.
- **Polynomial Regression**: Modeling the relationship between the independent and dependent variables using polynomial equations (useful for capturing non-linear relationships).
- **Logistic Regression**: While technically a classification algorithm, logistic regression is often introduced in the context of regression since it predicts probabilities (continuous values between 0 and 1) based on input features.

- **Ridge & Lasso Regression**: Variants of linear regression that use regularization techniques to prevent overfitting.

2. Evaluation of Regression Models

The performance of regression models is assessed using several evaluation metrics that compare the predicted values to the actual values in the dataset. Here are the most common metrics used:

Common Evaluation Metrics:

- **Mean Absolute Error (MAE)**:
    - MAE measures the average of the absolute differences between the predicted and actual values. It is easy to understand and is useful for evaluating the overall accuracy of a regression model.
    - Formula:

MAE=1n∑i=1n|yi−yi^|MAE = \frac{1}{n} \sum_{i=1}^{n} |y_i - \hat{y_i}|MAE=n1i=1∑n|yi−yi^|

where yiy_iyi are the true values and yi^\hat{y_i}yi^ are the predicted values.

- **Mean Squared Error (MSE)**:
    - MSE calculates the average of the squared differences between the predicted and actual values. This metric penalizes larger errors more heavily than MAE, as errors are squared.
    - Formula:

MSE=1n∑i=1n(yi−yi^)2MSE = \frac{1}{n} \sum_{i=1}^{n} (y_i - \hat{y_i})^2MSE=n1i=1∑n(yi−yi^)2

- **Root Mean Squared Error (RMSE)**:

- RMSE is the square root of the MSE and provides the error in the same units as the target variable. It is more sensitive to outliers compared to MAE.
- Formula:

RMSE=MSERMSE = \sqrt{MSE}RMSE=MSE

- **R-Squared (R^2)**:
  - R-squared represents the proportion of the variance in the dependent variable that is predictable from the independent variables. It is a measure of how well the model fits the data, with values closer to 1 indicating a better fit.
  - Formula:

R2=1−∑i=1n(yi−yi^)2∑i=1n(yi−y¯)2R^2 = 1 - \frac{\sum_{i=1}^{n} (y_i - \hat{y_i})^2}{\sum_{i=1}^{n} (y_i - \bar{y})^2}R2=1−∑i=1n(yi−y¯)2∑i=1n(yi−yi^)2

where y¯\bar{y}y¯ is the mean of the true values.

- **Adjusted R-Squared**:
  - Adjusted R-squared is a modified version of R-squared that accounts for the number of predictors in the model. It penalizes excessive use of irrelevant features and is useful when comparing models with different numbers of independent variables.

When to use which metric?

- **MAE**: Useful when you care equally about all errors, regardless of their size.

- **MSE/RMSE**: Better when you want to heavily penalize larger errors (outliers).
- **R-Squared**: Useful for understanding how well your model explains the variability in the data, but can be misleading when there are many irrelevant features.

3. Linear Regression

Linear regression is one of the simplest and most widely used regression techniques. It assumes a linear relationship between the dependent variable and the independent variable(s). The objective of linear regression is to find the best-fitting straight line (or hyperplane in the case of multiple variables) that minimizes the error between the predicted and actual values.

Simple Linear Regression (One Independent Variable)

In simple linear regression, the model assumes the relationship between the dependent variable yyy and the independent variable xxx is linear and can be expressed as:

y=β0+β1x+ϵy = \beta_0 + \beta_1x + \epsilony=β0+β1x+ϵ

where:

- yyy = dependent variable (target)
- xxx = independent variable (predictor)
- β0\beta_0β0 = intercept
- β1\beta_1β1 = slope (coefficient)
- ϵ\epsilonϵ = error term (residuals)

Multiple Linear Regression (Multiple Independent Variables)

In multiple linear regression, the model involves two or more independent variables. The relationship can be expressed as:

y=β0+β1x1+β2x2+⋯+βnxn+ϵy = \beta_0 + \beta_1x_1 + \beta_2x_2 + \dots + \beta_nx_n + \epsilony=β0+β1x1+β2x2+⋯+βnxn+ϵ

where:

- x1,x2,…,xnx_1, x_2, \dots, x_nx1,x2,…,xn = multiple independent variables
- β0,β1,…,βn\beta_0, \beta_1, \dots, \beta_nβ0,β1,…,βn = coefficients (intercept and slopes for each predictor)
- ϵ\epsilonϵ = error term

Assumptions of Linear Regression:

1. **Linearity**: The relationship between the dependent and independent variables is linear.
2. **Independence**: The residuals (errors) are independent.
3. **Homoscedasticity**: The variance of the residuals is constant across all levels of the independent variable(s).
4. **Normality**: The residuals should be normally distributed (important for significance testing).

Estimating Coefficients (β-values):

The coefficients of the linear regression model (β0\beta_0β0, β1\beta_1β1, etc.) are typically estimated using the **Ordinary Least Squares (OLS)** method, which minimizes the sum of squared residuals (errors).

5.2 Classification: Classification problems, Evaluation of classification models, Bayesian methods, Logistic regression.

🔍 **Classification Overview**

Classification is a **supervised learning** technique used to predict **categorical outcomes** (like yes/no, spam/not spam, disease/no disease). It's used when the target variable is **discrete**.

📌 1. Classification Problems

In classification problems, the goal is to assign a **label** (class) to a new observation based on past data.

🔸 **Examples:**

- Email → Spam or Not Spam
- Medical Diagnosis → Disease or No Disease
- Image → Cat or Dog

🔸 **Types:**

- **Binary Classification**: Two classes (e.g., pass/fail, spam/ham).
- **Multi-class Classification**: More than two classes (e.g., classifying fruits as apple, orange, banana).
- **Multi-label Classification**: One instance can belong to multiple classes (e.g., a movie can be action + thriller).

📊 2. Evaluation of Classification Models

To check how well a classification model performs, we use **evaluation metrics**.

✅ **Common Metrics:**

- **Accuracy**:

Accuracy=Correct PredictionsTotal Predictions\text{Accuracy} = \frac{\text{Correct Predictions}}{\text{Total Predictions}}Accuracy=Total PredictionsCorrect Predictions

- **Confusion Matrix**: Shows true/false positives and negatives.
    - **TP**: True Positive (correctly predicted positive)
    - **TN**: True Negative (correctly predicted negative)
    - **FP**: False Positive (wrongly predicted positive)
    - **FN**: False Negative (wrongly predicted negative)
- **Precision**:

Precision=TPTP+FP\text{Precision} = \frac{TP}{TP + FP}Precision=TP+FPTP

(How many predicted positives are actually correct?)

- **Recall**:

Recall=TPTP+FN\text{Recall} = \frac{TP}{TP + FN}Recall=TP+FNTP

(How many actual positives were found?)

- **F1-Score**:

F1=2×Precision × RecallPrecision + RecallF1 = 2 \times \frac{\text{Precision × Recall}}{\text{Precision + Recall}}F1=2×Precision + RecallPrecision × Recall

(Harmonic mean of Precision and Recall)

- **ROC Curve & AUC**: A graph to show model performance across thresholds.

🤖 3. Bayesian Methods

Bayesian classification uses **probabilities** to classify data points.

📌 **Naive Bayes Classifier:**

- Based on **Bayes' Theorem**:

P(A|B)=P(B|A)×P(A)P(B)P(A|B) = \frac{P(B|A) \times P(A)}{P(B)}P(A|B)=P(B)P(B|A)×P(A)

- Called *naive* because it assumes features are **independent**.
- Very fast and effective for text classification (like spam detection).

➕ **4. Logistic Regression**

- Despite its name, **logistic regression is a classification algorithm**.
- It is used to **predict the probability** of a class label.

📌 **How it works:**

- Uses a **sigmoid (logistic) function** to squash output between 0 and 1:

σ(z)=11+e−z\sigma(z) = \frac{1}{1 + e^{-z}}σ(z)=1+e−z1

- If the output is > 0.5 → class 1
  If the output is ≤ 0.5 → class 0

🔸 **Used For:**

- Binary classification (e.g., yes/no decisions)

🔸 **Advantage:**

- Simple, interpretable, and efficient.

5.3 Clustering: Clustering methods, Partition methods, Hierarchical methods, Evaluation of clustering models. Association Rule: Structure of Association Rule, Apriori Algorithm

🔹 Clustering (Unsupervised Learning)

Clustering is a technique where **similar data points are grouped together**. There are **no labels**—the algorithm finds patterns on its own.

📌 1. Clustering Methods

There are several types of clustering methods:

✅ Partition Methods

- Divide the data into **k groups (clusters)**.
- Most popular: **K-Means Clustering**
    - Choose k (number of clusters)
    - Randomly assign centroids
    - Assign points to the nearest centroid
    - Recalculate centroids and repeat

✅ Hierarchical Methods

- Build a tree-like structure of clusters.
- Two types:
    - **Agglomerative (Bottom-up)**: Start with individual points and merge.
    - **Divisive (Top-down)**: Start with one big cluster and split.

📊 2. Evaluation of Clustering Models

Since there are no labels, we use **internal metrics**:

- **Silhouette Score**: Measures how similar a point is to its own cluster vs. other clusters. Closer to 1 = better.
- **Dunn Index**: Ratio of the smallest distance between clusters to the largest within a cluster.
- **Elbow Method**: Used to find the best value of k in K-means by plotting inertia (within-cluster sum of squares) vs. number of clusters.

🔷 Association Rule Mining

Association Rules are used to find **patterns or relationships** between items in large datasets. Often used in **market basket analysis**.

Example:
🛒 If a customer buys **milk** and **bread**, they might also buy **butter**.

📌 1. Structure of an Association Rule

A rule looks like this:

mathematica

CopyEdit

If {X} then {Y}

- X = Antecedent (what you have)
- Y = Consequent (what is likely to come next)

📊 Important Metrics:

- **Support**: How often the rule appears in the data

Support(X→Y)=Transactions with both X and YTotal transactions\text{Support}(X \rightarrow Y) = \frac{\text{Transactions with both X and Y}}{\text{Total transactions}}Support(X→Y)=Total transactionsTransactions with both X and Y

- **Confidence**: How often Y appears when X is present

Confidence(X→Y)=Support(X∪Y)Support(X)\text{Confidence}(X \rightarrow Y) = \frac{\text{Support}(X \cup Y)}{\text{Support}(X)}Confidence(X→Y)=Support(X)Support(X∪Y)

- **Lift**: How much more likely Y is when X is present compared to random chance

Lift(X→Y)=Confidence(X→Y)Support(Y)\text{Lift}(X \rightarrow Y) = \frac{\text{Confidence}(X \rightarrow Y)}{\text{Support}(Y)}Lift(X→Y)=Support(Y)Confidence(X→Y)

    - Lift > 1 = Positive association

📌 2. Apriori Algorithm

- A popular algorithm to **generate association rules** from transactional data.

🔸 Steps:

1. **Find frequent itemsets** (items that appear often together)
2. Use these itemsets to **generate rules**
3. Prune rules that don't meet **minimum support/confiden**ce