# WEEK 7

**Question1: Explain the evaluation matrix for classification as follows.**

**1) confusion matrix**
**2) Accuracy**
**3) F1 – score**
**4) AUC_ROC**
**5) Precision and Recall**

## 1. Confusion Matrix

Confusion Matrix is a performance measurement for the machine learning classification problems where the output can be two or more classes. It is a table with combinations of predicted and actual values.

**A confusion matrix is defined as the table that is often used to describe the performance of a classification model on a set of the test data for which the true values are known**.

| Predicted Values | | Actual Values: Positive (1) | Actual Values: Negative (0) |
|---|---|---|---|
| | Positive (1) | TP | FP |
| | Negative (0) | FN | TN |

It is extremely useful for measuring the Recall, Precision, Accuracy, and AUC-ROC curves. **True Positive:** We predicted positive and it's true. In the image, we predicted that a woman ispregnant and she actually is.

**True Negative:** We predicted negative and it's true. In the image, we predicted that a man isnot pregnant and he actually is not.

**False Positive (Type 1 Error)-** We predicted positive and it's false. In the image, we predictedthat a man is pregnant but he actually is not.

**False Negative (Type 2 Error)-** We predicted negative and it's false. In the image, wepredicted that a woman is not pregnant but she actually is.

## 2. Accuracy

Accuracy simply measures how often the classifier correctly predicts. We can define accuracyas the ratio of the number of correct predictions and the total number of predictions.

$$\textit{Accuracy} = \frac{TP+TN}{TP+TN+FP+FN} = \frac{\textit{No of Correct Predictions}}{\textit{Total no of predictions}}$$

- When any model gives an accuracy rate of 99%, you might think that model is performing very good but this is not always true and can be misleading in some situations.
- Accuracy is useful when the target class is ***well balanced*** but is not a good choice for the unbalanced classes.

- Imagine the scenario where we had 99 images of the dog and only 1 image of a cat present in our training data. Then our model would always predict the dog, and therefore we got 99% accuracy.
- In reality, Data is always imbalanced for example Spam email, credit card fraud, and medical diagnosis.
- Hence, if we want to do a better model evaluation and have a full picture of the model evaluation, other metrics such as recall and precision should also be considered

3. **F1 Score**

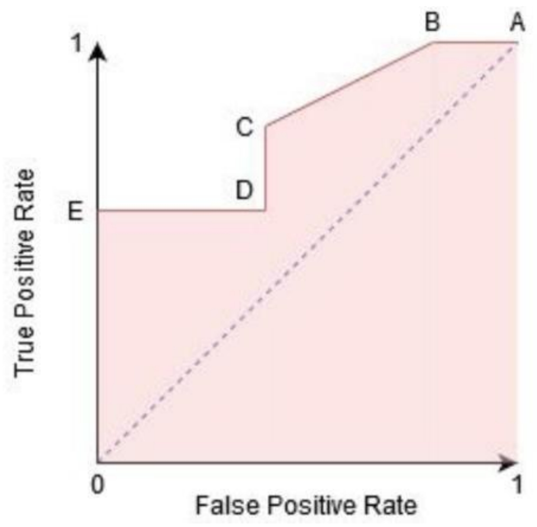
- It gives a combined idea about Precision and Recall metrics. It is maximum whenPrecision is equal to Recall.
- **F1 Score is the harmonic mean of precision and recall.**

$$F1 - score = 2 * \frac{Precision * Recall}{Precision + Recall}$$

- The F1 score punishes extreme values more. F1 Score could be an effective evaluationmetric in the following cases:
  - When FP and FN are equally costly.
  - Adding more data doesn’t effectively change the outcome
  - True Negative is high

4. **AUC-ROC**

- The Receiver Operator Characteristic (ROC) is a probability curve that plots the TPR(True Positive Rate) against the FPR(False Positive Rate) at various threshold values and separatesthe ‘signal’ from the ‘noise’.
- The **Area Under the Curve (AUC)** is the measure of the ability of a classifier to distinguishbetween classes. From the graph, we simply say the area of the curve ABDE and the X andY-axis.
- From the graph shown below, the greater the AUC, the better is the performance of the model at different threshold points between positive and negative classes.
- This simply means that When AUC is equal to 1, the classifier is able to perfectly distinguish between all Positive and Negative class points.
- When AUC is equal to 0, the classifier would be predicting all Negatives as Positives and vice versa.
- When AUC is 0.5, the classifier is not able to distinguish between the Positive and Negativeclasses.



5. **Precision**

- Precision explains how many of the correctly predicted cases actually turned out to bepositive.
- Precision is useful in the cases where False Positive is a higher concern than False Negatives.
- The importance of Precision is in music or video recommendation systems, e-commerce websites, etc. where wrong results could lead to customer churn and this could be harmful to the business.
- **Precision for a label is defined as the number of true positives divided by the number of predicted positives.**

$$Precision = \frac{TruePositive}{TruePositive + FalsePositive}$$

6. **Recall (Sensitivity)**

- Recall explains how many of the actual positive cases we were able to predict correctly with our model.
- It is a useful metric in cases where False Negative is of higher concern than False Positive. It is important in medical cases where it doesn't matter whether we raise a false alarm butthe actual positive cases should not go undetected.
- **Recall for a label is defined as the number of true positives divided by the total number of actual positives.**

$$Recall = \frac{TruePositive}{TruePositive + FalseNegative}$$

**Question: 2. Build a data set and predict the heart disease based on BP, Sugar, Age, Gender and Cholesterol by using relevant operations.**

**ANSWER:**

```
import pandas as pd
from sklearn.model_selection import train_test_split
from sklearn.linear_model import LogisticRegression
from sklearn.metrics import accuracy_score

# Sample data
data = {'BP': [120, 140, 130, 150, 110],
 'Sugar': [5, 6, 7, 8, 9],
 'Age': [40, 50, 35, 45, 55],
 'Gender': ['M', 'F', 'M', 'M', 'F'],
 'Cholesterol': [180, 200, 220, 240, 260],
 'Heart_Disease': [0, 1, 1, 1, 0]}
df = pd.DataFrame(data)

# Split the data into training and test sets
X = df[['BP', 'Sugar', 'Age', 'Gender', 'Cholesterol']]
y = df['Heart_Disease']
```

```
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2,random_state=42)

# Train a logistic regression model
model = LogisticRegression()
model.fit(X_train, y_train)

# Make predictions on the test set
y_pred = model.predict(X_test)

# Evaluate the model's performance
accuracy = accuracy_score(y_test, y_pred)
print("Accuracy:", accuracy)
```

In this example, we're using a logistic regression model to predict heart disease based on the five features: BP, Sugar, Age, Gender, and Cholesterol. The data is split into a training set and a test set, and the model is trained on the training set and evaluated on the test set. The accuracy of the model is printed as the evaluation metric It's worth noting that this is a simple example and real-world data would likely require more preprocessing and feature engineering. Also, this is one of the many ways to model the problem, the model's performance might change using different algorithms.

**Question3: The confusion matrix for a model is as shown below. Evaluate accuracy, precision, recall, Specificity and F1-Score, AUC-ROC**

| | | Actual | |
|---|---|---|---|
| | | 1 | 0 |
| Predicted | 1 | 397 | 103 |
| | 0 | 126 | 142 |

**ANSWER:**

| | |
|---|---|
| 397 | 103 |
| 126 | 142 |

| | |
|---|---|
| TP | FP |
| FN | TN |

$$\text{accuracy} \Rightarrow \frac{TP + TN}{TP + TN + FP + FN}$$

$$\Rightarrow \frac{397 + 142}{397 + 103 + 126 + 142}$$

$$\Rightarrow \frac{539}{768}$$

$$\boxed{\text{accuracy} \Rightarrow 0.7018229167}$$

$$\text{Precision} \Rightarrow \frac{TP}{TP + FP}$$

$$\Rightarrow \frac{397}{397 + 103}$$

$$\Rightarrow \frac{397}{500}$$

$$\boxed{\text{Precision} \Rightarrow 0.794}$$

$$\text{Recall} \Rightarrow \frac{TP}{TP + FN}$$

$$\Rightarrow \frac{397}{397 + 126}$$

$$\Rightarrow 397$$

Specificity :- $\frac{TN}{TN + FP}$

$$\Rightarrow \frac{142}{142 + 103}$$

$$\Rightarrow \frac{142}{245}$$

$\boxed{\text{Specificity} \Rightarrow 0.5795918367}$

F1-Score $\Rightarrow 2\left(\frac{\text{Precision} * \text{Recall}}{\text{Precision} + \text{Recall}}\right)$

$$2\left(\frac{0.794 * 0.759}{0.794 + 0.759}\right)$$

$$2\left(\frac{0.602646}{1.553}\right)$$

$$2\left(0.388052801\right)$$

$\boxed{\text{F1-Score} \Rightarrow 0.7761056021}$

AUC-ROC $\Rightarrow$

**Question 4: Given a dataset perform comparative analysis using decision tree and SVM algorithm and check accuracy**.

**ANSWER:** To perform a comparative analysis using decision tree and SVM algorithms on a given dataset, you would need to follow these steps:

- Prepare the dataset: Clean and preprocess the dataset, if necessary, to ensure that it is in a format that can be used by the algorithms.
- Split the dataset: Divide the dataset into a training set and a test set. The training set will be used to train the algorithms, and the test set will be used to evaluate the performance of the algorithms.
- Train the algorithms: Use the training set to train the decision tree and SVM algorithms. You can use a library such as scikit-learn in Python to implement the algorithms.
- Test the algorithms: Use the test set to evaluate the performance of the algorithms by making predictions and comparing them to the actual values. You can use metrics such as accuracy, precision, and recall to measure the performance of the algorithms.
- Compare the performance: Compare the performance of the two algorithms by looking at the metrics you have used to evaluate them. You can also use visualizations such as confusion matrices to help you understand the performance of the algorithms.
- Select the best model: Based on the performance of the two algorithms, select the one that performed the best on the dataset.

Please note that the accuracy is not the only metric to consider, other performance metrics such as precision, recall, F1-Score, AUC-ROC should be considered based on the problem scenario.

**Question 5: A machine learning model was built to classify people based on whether they speak English or Hindi. The confusion matrix for the model is as shown below. Compute accuracy, precision, recall, F1-Score and Specificity.**

**Actual**

**Predicted**

| | English Speaker | Hindi Speaker |
|---|---|---|
| English Speaker | 109 | 11 |
| Hindi Speaker | 8 | 56 |

**ANSWER**:

Actual

5) Predicted

| | English Speaker | Hindi Speaker |
|---|---|---|
| E/S | 109 | 11 |
| H/S | 8 | 56 |

| | |
|---|---|
| TP | FP |
| FN | FN |

$$\text{Precision} = \frac{TP}{TP + FP}$$

$$\Rightarrow \frac{109}{109 + 11}$$

$$\Rightarrow \frac{109}{120}$$

Percision => 0.90833333

$$\text{accuracy} = \frac{TP + TN}{TP + FP + FN + TN}$$

$$\Rightarrow \frac{109 + 56}{109 + 11 + 56 + 8}$$

$$\frac{165}{184}$$

accuracy => 0.8967391304

$$\text{Recall} \Rightarrow \frac{TP}{TP + FN}$$

$$\Rightarrow \frac{109}{109 + 8}$$

$$\Rightarrow \frac{109}{117}$$

$\Rightarrow$

$$\text{Recall} \Rightarrow 0.9316239316$$

$$\text{F1-score} \Rightarrow 2\left(\frac{\text{Precision} * \text{Recall}}{\text{Precision} + \text{Recall}}\right)$$

$$2\left(\frac{0.90 * 0.93}{0.90 + 0.93}\right)$$

$$2\left(\frac{0.837}{1.83}\right)$$

$$2\,(0.4573770492)$$

$$\text{F1-score} \Rightarrow 0.9147540984$$

$$\text{Specificity} \Rightarrow \frac{TN}{TN + FP}$$

$$\Rightarrow \frac{56}{56 + 11}$$

$$\Rightarrow \frac{56}{67}$$

$$\text{Specificity} \Rightarrow 0.8358208955$$