MINOR PROJECT 2 (COGNITIVE APPLICATION)

THEORY QUESTIONS

1. **When should you use classification over regression?**
   Classification produces discrete values and dataset to strict categories, while regression gives you continuous results that allow you to better distinguish differences between individual points, like cat or dog, cancer: malignant or benign, etc.

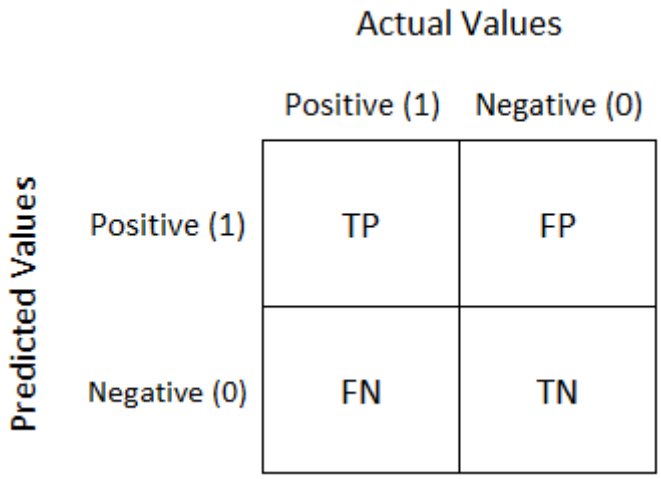
2. **How do you deal with the class imbalance in a classification problem?**
   Dealing with imbalanced datasets entails strategies such as improving classification algorithms or balancing classes in the training data (data pre-processing) before providing the data as input to the machine learning algorithm. The later technique is preferred as it has wider application.

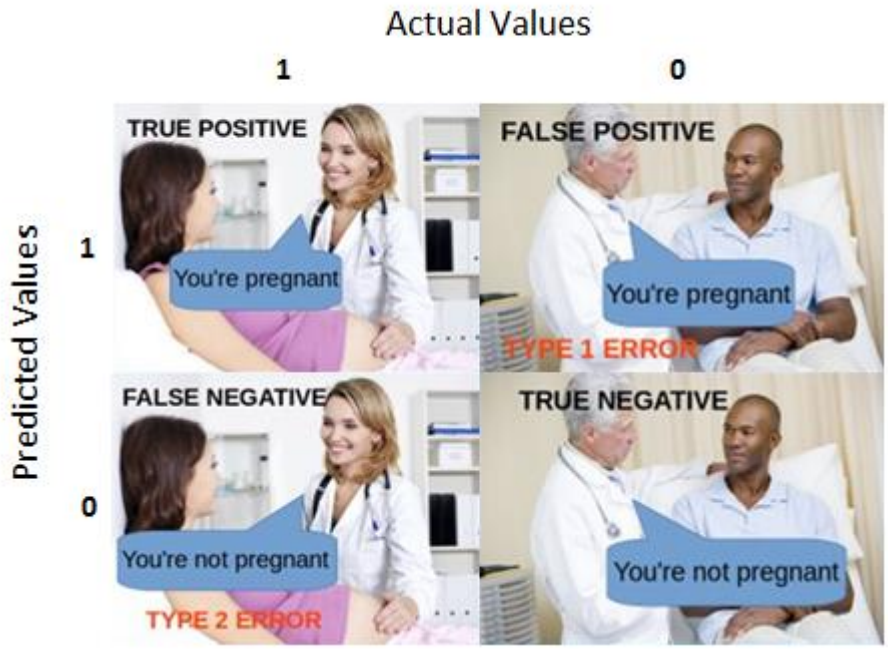
   The main objective of balancing classes is to either increasing the frequency of the minority class or decreasing the frequency of the majority class. This is done in order to obtain approximately the same number of instances for both the classes. A few methods to deal with class imbalance is:

   - Random Undersampling: It aims to balance class distribution by randomly eliminating majority class examples. This is done until the majority and minority class instances are balanced out.
   - Random Oversampling: Over-Sampling increases the number of instances in the minority class by randomly replicating them in order to present a higher representation of the minority class in the sample.
   - Using algorithmic ensemble techniques like bagging, boosting, etc.
   - Using recall and precision as metrics to give higher weightage to class with less samples.

3. **What is a confusion matrix and why do you need it?**

A confusion matrix is a table that is often used to describe the performance of a classification model on a set of test data for which the true values are known. It is an N x N matrix where N is the number of categorical variables, with the columns giving the actual values and rows giving the predicted values. For example for a binary classification problem, the confusion matrix will be as follows:





True Positive (TP)

- The predicted value matches the actual value
- The actual value was positive and the model predicted a positive value

True Negative (TN)

- The predicted value matches the actual value

- The actual value was negative and the model predicted a negative value

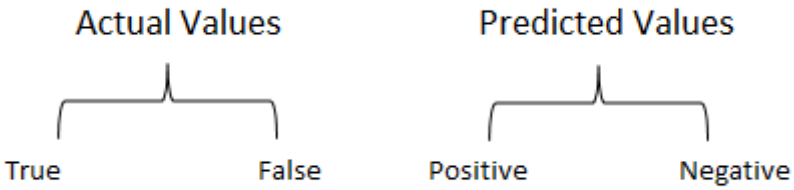
False Positive (FP) – Type 1 error

- The predicted value was falsely predicted
- The actual value was negative but the model predicted a positive value

False Negative (FN) – Type 2 error

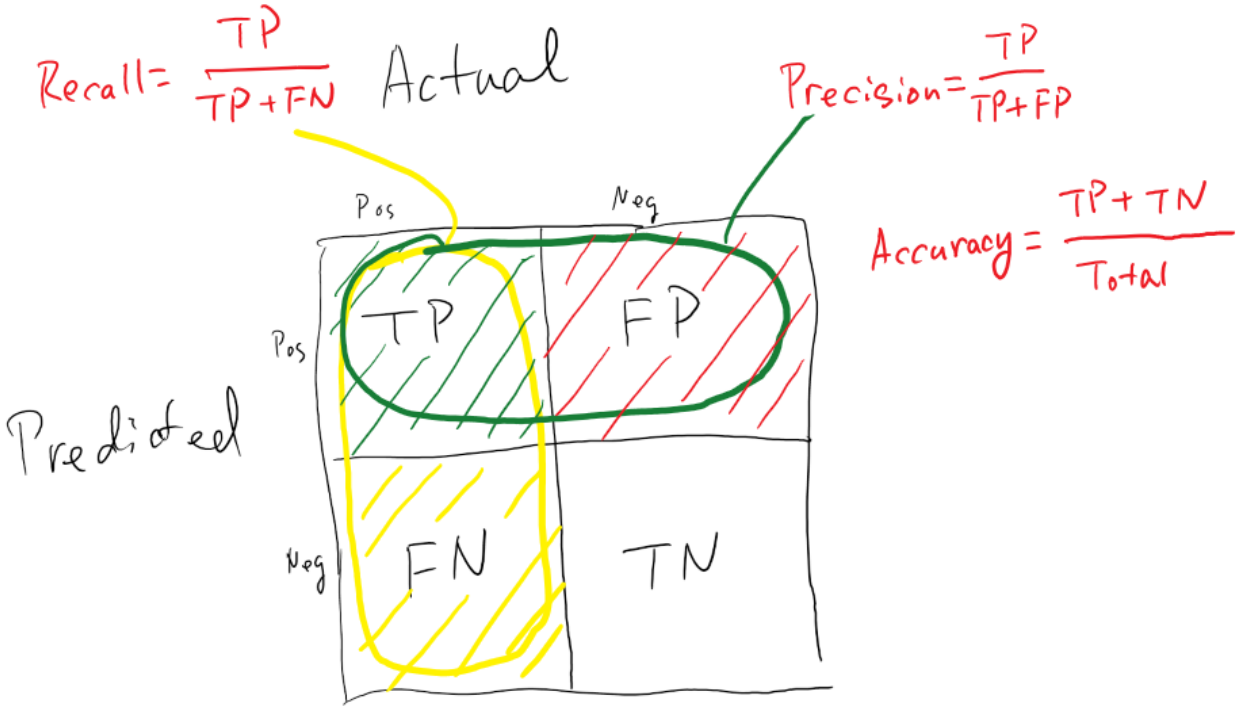
- The predicted value was falsely predicted
- The actual value was positive but the model predicted a negative value

Just Remember, We describe predicted values as Positive and Negative and actual values as True and False.



Confusion matrices are used to visualize important predictive analytics like accuracy, recall and precision, which are used to test the performance of classification model in different scenarios. The following parameters are defined as follows:

Let's understand confusion matrix through math.



Accuracy: Accuracy is a good measure when both the classes have approximately equal occurrences like predicting whether the image is a cat or dog.

- $$Accuracy = \frac{TP + TN}{TP + FP + TN + FN}$$

- Recall: Recall tells us how many of the actual positive cases we were able to predict correctly with our model. Recall is important in medical cases where it doesn't matter whether we raise a false alarm but the actual positive cases should not go undetected.

$$Recall = \frac{TP}{TP + FN}$$

- Precision: Precision tells us how many of the correctly predicted cases actually turned out to be positive. Precision is important in music or video recommendation systems, e-commerce websites, etc. Wrong results could lead to customer churn and be harmful to the business.

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

4. What is the difference between sigmoid and softmax function?

The sigmoid function is used for two class logistic regression whereas softmax function is used for multiclass logistic regression.

In the two-class logistic regression, the predicted probabilities are as follows, using the sigmoid function:

$$\Pr(Y_i = 0) = \frac{e^{-\boldsymbol{\beta}\cdot\mathbf{X}_i}}{1 + e^{-\boldsymbol{\beta}\cdot\mathbf{X}_i}}$$
$$\Pr(Y_i = 1) = 1 - \Pr(Y_i = 0) = \frac{1}{1 + e^{-\boldsymbol{\beta}\cdot\mathbf{X}_i}}$$

In the multiclass logistic regression, with K classes, the predicted probabilities are as follows, using the softmax function:

$$\Pr(Y_i = k) = \frac{e^{\boldsymbol{\beta}_k\cdot\mathbf{X}_i}}{\sum_{0\le c\le K} e^{\boldsymbol{\beta}_c\cdot\mathbf{X}_i}}$$

This is main reason why the Softmax is cool. It makes sure that the sum of all our output probabilities is equal to one.

*Characteristics of a **Sigmoid Activation Function***

- Used for Binary Classification in the Logistic Regression model
- The probabilities sum does not need to be 1
- Used as an Activation Function while building a Neural Network

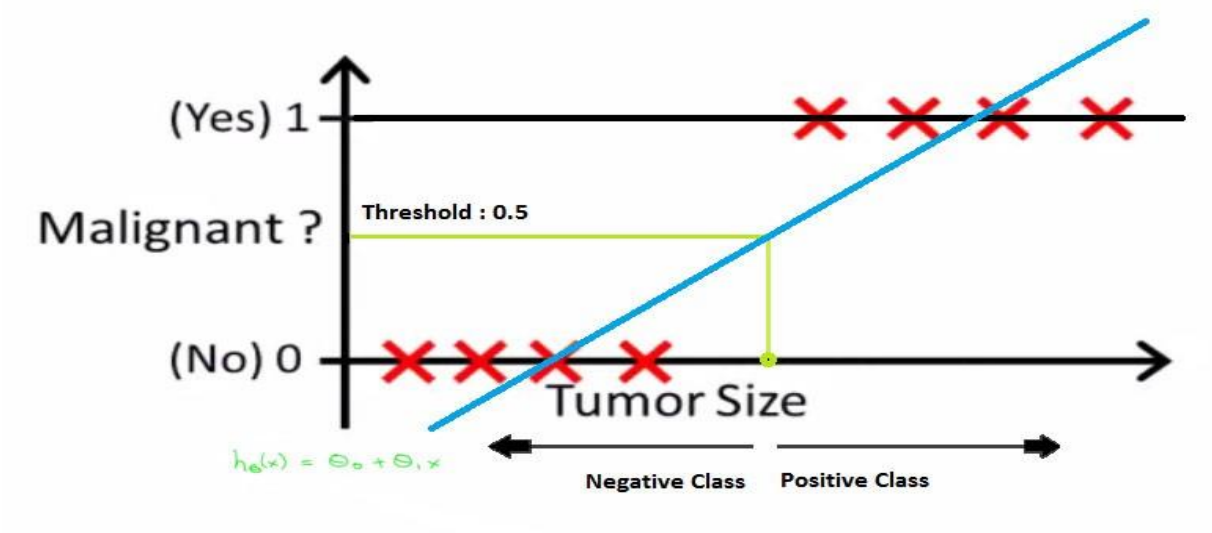
*Characteristics of a **Softmax Activation Function***

- Used for Multi-classification in the Logistics Regression model
- The probabilities sum will be 1
- Used in the different layers of Neural Networks

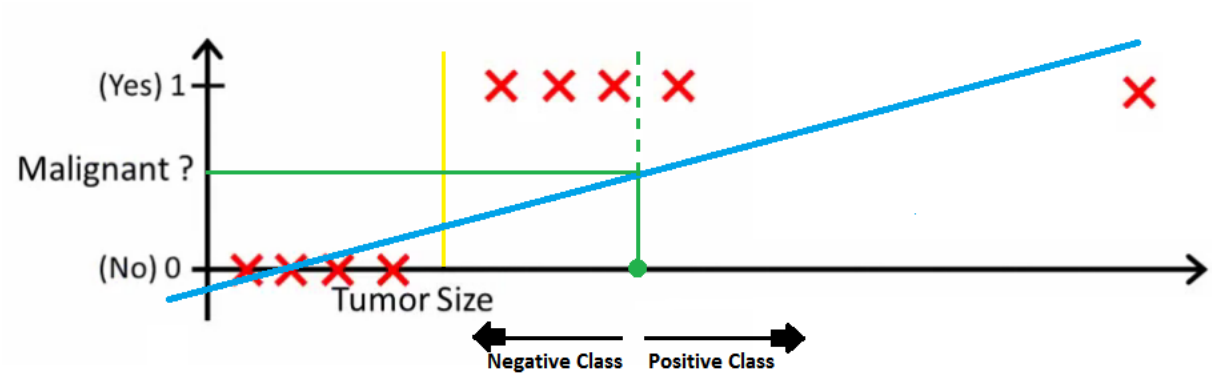
5. **Why is logistic regression a type of classification technique and not a regression? Name the function it is derived from?**

   Logistic model is used to model probabilities of classes like cat/dog, pass/fail, etc. It is therefore a type of classification technique. It widely uses the sigmoid function.

   Because it's derived from linear regression. Both are almost the same. Suppose you try to fit linear regression for a classification task, you get the following graph:



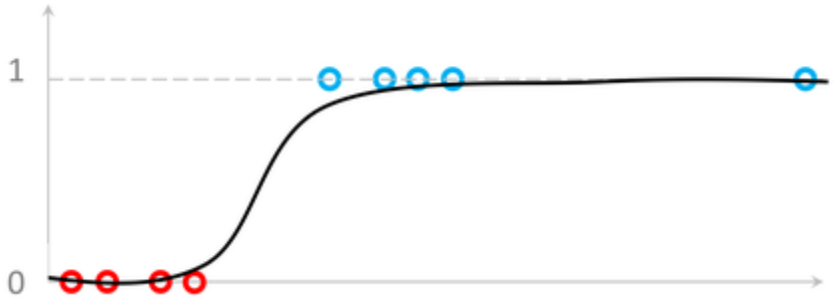
Let's say that all the points which lie to the left of the line represent No(0) and all the points which lie to the right of the line represent Yes(1). It works fairly well, right? But then this situation occurs:



An outlier appears! We need to fit a different equation to accommodate the outlier. Do some mathe-magic, and enter the Logistic equation!

$$p(x) = \frac{1}{1 + e^{-(\beta_0 + \beta_1 x)}}$$

Let's fit it, and take care of the outlier!



The outlier doesn't worry us any more! This is the story of how and why logistic regression came into being. Hope you enjoyed it! And they all lived happily ever after, even the outlier!