Classification

Topics covered so far

1. Intro: Classification
2. Gaussian Models
3. Logistic Regression
4. Performance Assessments
5. K-Nearest Neighbors

Discussion questions

1. Why do we use logistic regression?
2. What is a confusion matrix and how can you interpret it?
3. Why is accuracy not always a good performance measure?
4. How to choose the threshold using the Precision-Recall curve?
5. Is there a performance measure that can cover both Precision and Recall?
6. How does the K-NN algorithm work? How to identify K in this algorithm?

Why do we use logistic regression?

- Logistic Regression is a supervised learning algorithm that is used for classification problems, i.e., where the dependent variable is categorical.
- In logistic regression, we use the Sigmoid function to calculate the probability of the dependent variable.
- The real-life applications of logistic regression are churn prediction, spam detection, etc.
- The below image shows how logistic regression is different from linear regression in fitting the model.

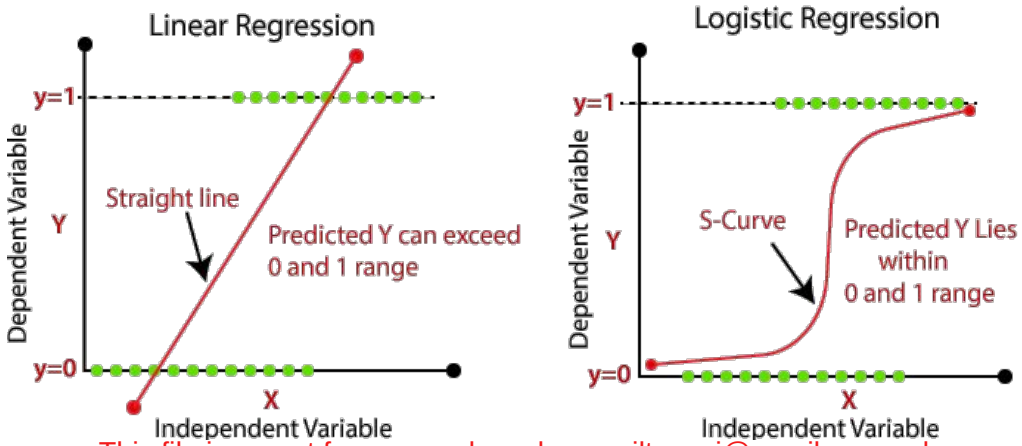


Image Source

Confusion matrix

It is used to measure the performance of a classification algorithm. It can be used to calculate the following metrics:

1. **Accuracy:** Proportion of correctly predicted results among the total number of observations

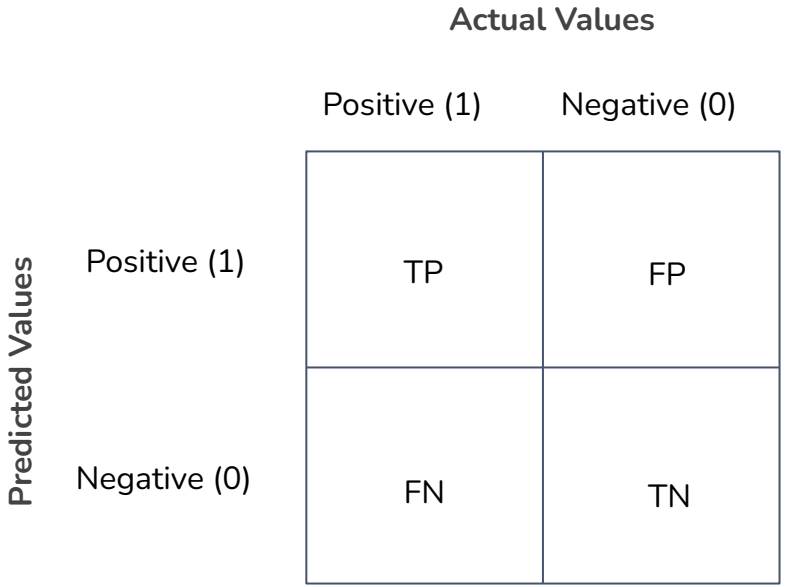
   Accuracy = (TP+TN)/(TP+FP+FN+TN)

2. **Precision:** Proportion of true positives to all the predicted positives, i.e., how valid the predictions are

   Precision = (TP)/(TP+FP)

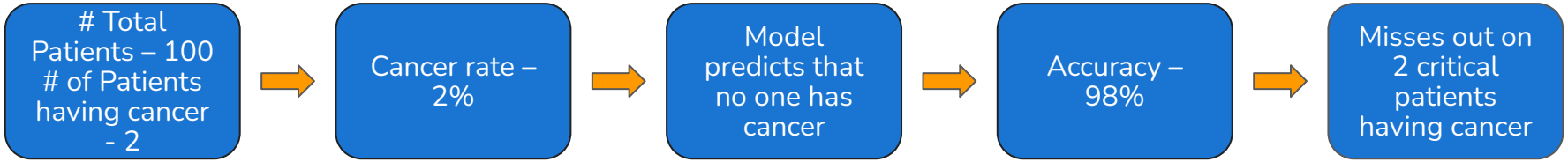
3. **Recall:** Proportion of true positives to all the actual positives, i.e., how complete the predictions are

   Recall = (TP)/(TP+FN)



Why accuracy is not always a good performance measure

Accuracy is simply the overall % of correct predictions and can be high even for very useless models.

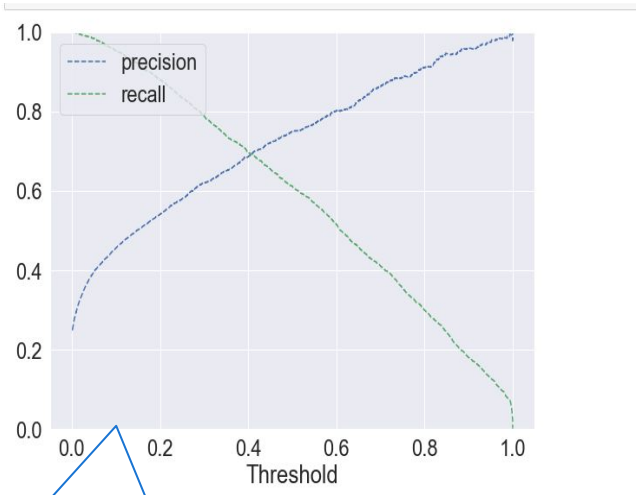


- Here, accuracy will be 98%, even if we simply predict that every patient does not have cancer.
- In this case, Recall should be used as a measure of model performance; high recall imply fewer false negatives.
- Fewer false negatives implies a lower chance of 'missing' a cancer patient, i.e., predicting a cancer patient as one not having cancer.
- This is where we need other metrics to evaluate model performance.

- The other important metrics are Recall and Precision:
  - Recall - What % of actuals 1s did the model capture in prediction?
  - Precision - What % of predicted 1s are actual 1s?
- There is a tradeoff - as you try to increase the Recall, the Precision will reduce and vice versa.
- This tradeoff can be used to figure out the right threshold to use for the model.

How to chose thresholds using the Precision-Recall curve?

- The Precision-Recall curve is a useful measure of the success of prediction when the classes are imbalanced.
- The curve shows the tradeoff between the precision and the recall for different thresholds.
- It can be used to select an optimal threshold as required to improve the model performance.
- Here, as we can see, the precision and the recall are almost equal when the threshold is around 0.4.
- If we want a higher precision, we can increase the threshold.
- If we want a higher recall, we can decrease the threshold.



Choosing different thresholds can completely change the model's performance.
It is important to think about what constitutes the 'sweet spot'.

Is there a performance measure that can cover both Precision and Recall?

- F1 Score is a measure that takes into account both Precision and Recall.
- The F1 Score is the harmonic mean of Precision and Recall. Therefore, this score takes both false positives and false negatives into account.

$$F1\ Score = \frac{2 \times Precision \times Recall}{Precision + Recall}$$

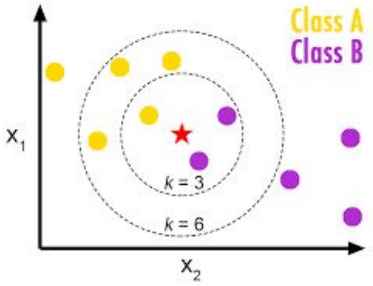
- The highest possible value of the F1 score is 1, indicating perfect precision and recall, and the lowest possible value is 0.

K-Nearest Neighbours (K-NN) algorithm

This algorithm uses features from the training data to predict the values of new data points, which means the new data point will be assigned a value based on how similar it is to the data points in the training set. We can define its working in the following steps:

- Step 1: We need to choose the value of K, i.e., the number of nearest data points to consider. K can be any positive integer.
- Step 2: For each point in the test data do the following:
  - Calculate the distance between the test point and each training point with the help of any of the distance methods, namely: Euclidean, Manhattan, etc. The most commonly used method to calculate the distance is the Euclidean method.
  - Now, based on the distance value, sort them in ascending order.
  - Next, choose the top K rows from the sorted array.
  - Now, assign a class to the test point based on the most frequent class.
- Step 3: Repeat this process until all the test points are classified in a particular class.

We try different values of K and plot them against the test error. The lower the value of the test error, the better the value of K.



Case Study

Appendix

LDA vs QDA

Linear Discriminant Analysis	Quadratic Discriminant Analysis
It is a linear classifier but much less flexible than QDA	It is a non-linear classifier but more flexible than LDA
It assumes a common covariance matrix for all the classes	It assumes that each class has its covariance matrix
It is preferred when the training set only has a few observations	It is preferred when the training set is very large
It can be used as a dimensionality reduction technique	It cannot be used as a dimensionality reduction technique

Great Learning

Happy Learning !