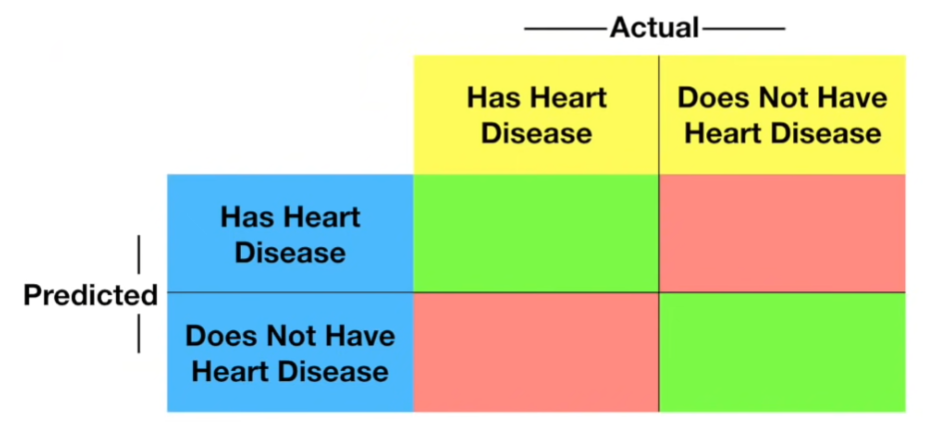
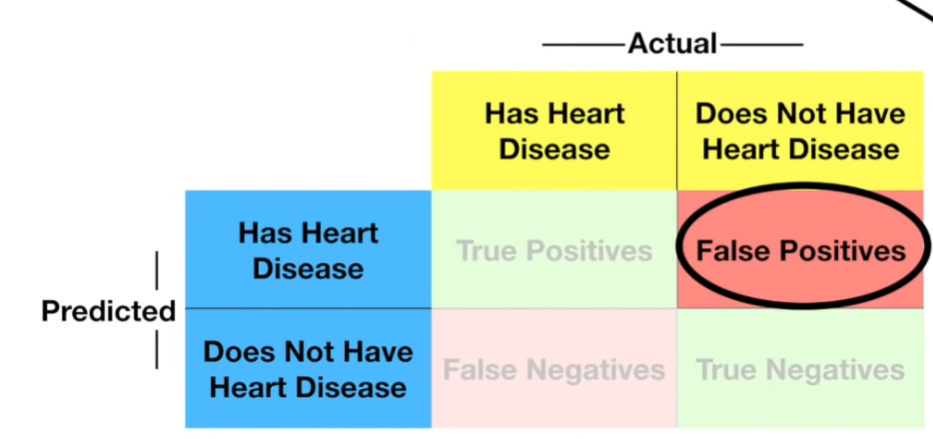
Confusion Matrix

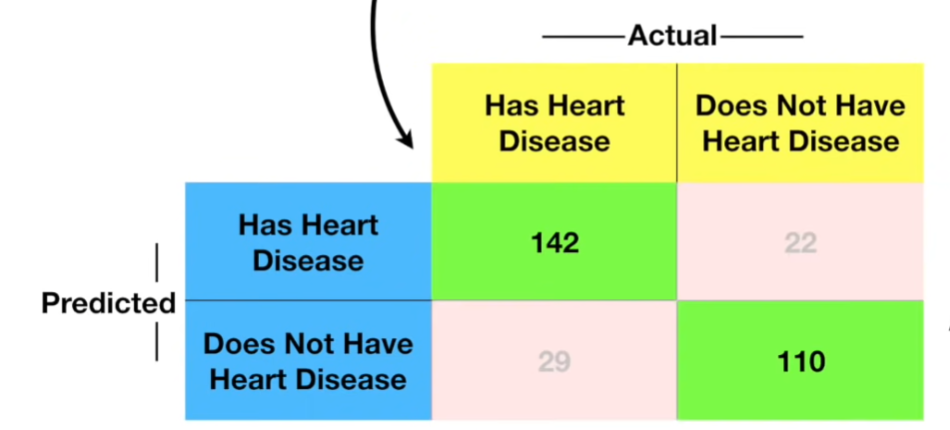
Confusion matrix checks how well a model performs on testing data.



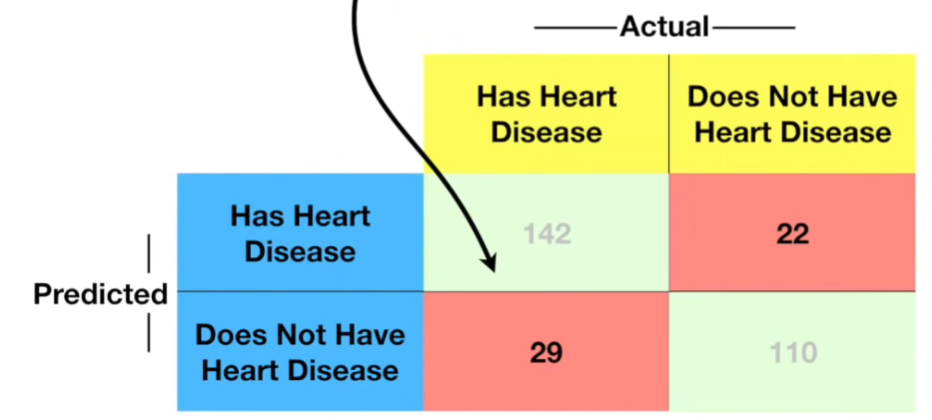
* The rows in the confusion matrix corresponds to what machine learning has predicted (predicted values)
* And the column corresponds to the known truth (actual values)



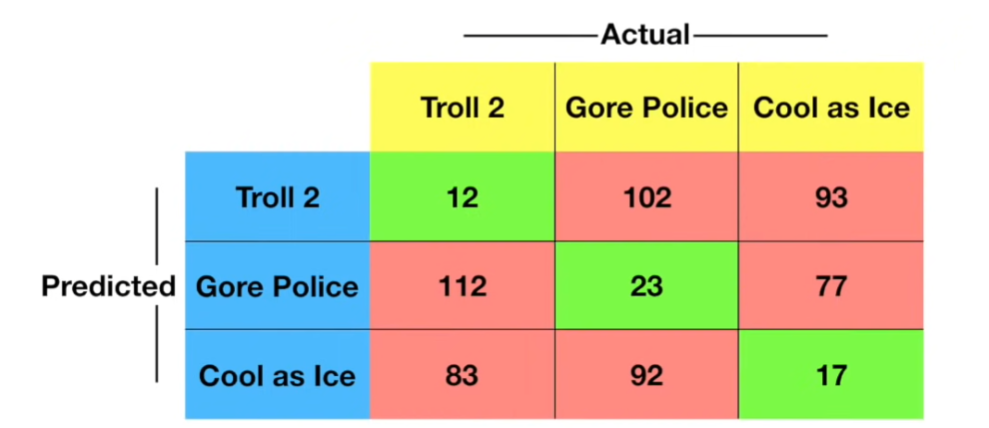
* True positive and True Negatives are the predictions that are correctly made by the algorithm. The numbers on the green diagonal indicate all correct predicted.



* False positive: Predicted positive but actual negative. False negative: Predicted negative but actually positive. The red diagonal indicates misclassification



For multiclass classification (3 categories here) the confusion matrix will look like,



* Here the green boxes are correctly predicted. And the red boxes are misclassified.

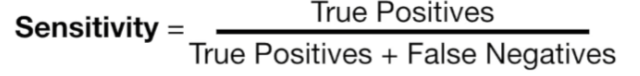
Therefore, for a classification problem with,

* For 2 classes we have 2 X 2 matrix
* For 3 classes we have 3 X 3 matrix
* For 4 classes we have 4 X 4 matrix and so on.

In Summary a confusion matrix tells what our machine learning algorithm did right and what it did wrong.

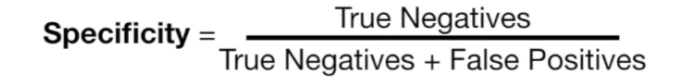
Sensitivity and specificity:

Sensitivity:



* Sensitivity tells us about the percentage of positive predictions.
* In other words, sensitivity is the percentage of correctly identifying positives.

Specificity:



* Specificity tells us about the percentage of negative predictions.
* In other words, specificity is the percentage of correctly identifying negatives.

When the matrix is more than 2 X 2, then there we can’t just pick a number from the matrix and do sensitivity and specificity calculation. So, here we have found sensitivity and specificity for each category.

In summary, if we want to choose a model where we more focus on positive values go for the model with more sensitivity, if we want a model where the more focus is on negative values then we go for specificity.

Here is the summary of all performance metrics.

Confusion matrix:

|  |  |  |  |
| --- | --- | --- | --- |
| Predicted |  | Actual | |
| Positive | Negative |
| Positive | TP | FP |
| Negative | FN | TN |

True Positives (TP) – The model correctly predicts a positive class.

False positive (FP) – Predicted positive but actually negative. (Type I error)

False negative (FN) – Predicted negative but actually positive (Type II error)

True negative (TN) – The model correctly predicts the negative class.

1. Accuracy

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

* Checks the correctness of the model, works well with balanced datasets (when the positive and negative classes are the same)
* Disadvantage: Doesn’t work well with imbalanced dataset

1. Precision

Precision = TP/TP+FP

* Precision tells that out of all positive predictions how many are actually positive.
* Disadvantage: Doesn’t account for false negatives

1. Recall (Sensitivity, True positive rate)

Recall = TP/TP+FN

* This measures how the true positive cases are correctly predicted.
* Doesn’t account for false positives.

1. F1-Score (Harmonic mean of precision and recall)

F1-Score = 2 X (Precision \* Recall) / (Precision + Recall)

* Balances precision and recall for imbalance dataset
* F1-Score is used when the precision and recall are equally important.

1. Specificity (True Negative rate)

Specificity = TN / (TN + FP)

* This measure how many true negative cases are correctly predicted.