9 – classification metrics

Accuracy is misleading when we have class imbalance (one class happens much more frequent than other classes) (However, if both classes are equally important then sometimes class imbalance are acceptable)

9.2 - Confusion Matrix

A common way to look at the errors made by your machine learning algorithm, gives us information about

1.false positive

2.false negative

3.true positive

4.true negative

9.3 - Precision, Recall, F1 score

1.Precision: Among the positive examples you identified, how many were actually positive? precision = TP / (TP + FP)

2.Recall: Among all positive examples, how many did you identify?, which is equal to TP / (TP + FN) = TP / #positives

3.F1 score: F1-score combines precision and recall to give one score, which could be used in hyperparam optimization.

f1 = 2 \* (precision \* recall) / (precision + recall)

Macro average

* You give equal importance to all classes and average over all classes.
* For instance, in the example above, recall for non-fraud is 1.0 and fraud is 0.63, and so macro average is 0.81.
* More relevant in case of multi-class problems.

Weighted average

* Weighted by the number of samples in each class.
* Divide by the total number of samples.

Macro and weighted averages are different ways to average the scores of a classifier. An average can be desirable since scores, other than accuracy, change according to which class is considered positive, and looking at the average allows us to assess the overall behaviour. A macro average is a simple arithmetic average of the scores and does not consider the number of samples in each class (that is, it gives equal importance to all classes). A weighted average instead is weighted by the number of samples in each class, so the scores obtained on classes with more samples will be given more importance (**undesirable if the most important class to identify is the minority class)**.

\*\*note that what you considered "positive" is important to the scores above

9.4 - Addressing Class Imbalance

E.g. the credit card fraud dataset is highly imbalanced (with many examples of non-fraudulent transactions than fraudulent transactions)

We want to spot all the fraudulent transactions so minimising false negatives is absolutely important. Then types of errors play a very important role.

Two scenarios to consider:

1.If it is really that one class is much more than the others, then it's ok to just ignore the class imbalance issues

2.It it is because of data collection methods, that means your test and training data come from different distributions! Then that's is a big problem because your training data cannot represent the population

Solutions:

1. change the threshold (so that your metrics changed, say by lowering the threshold to increase your recall)

2. change the data (undersampling/oversampling data) - Not in syllabus

3. change the training procedure - "class\_weight: dict or 'balanced'", which allows us to specify which class is more important to use.

• One way to do this is by computing the area under the PR curve.

• This is called average precision (AP score)

• AP score has a value between 0 (worst) and 1 (best).

AP (Area under the PR curve) vs. F1-score¶

It is very important to note this distinction:

F1 score is for a given threshold and measures the quality of predict.

AP score is a summary across thresholds and measures the quality of predict\_proba.

**AUC (Area under the curve - Receiver Operating Characteristic (ROC) curve which is FPR-x vs TPR-y)**

**AUC of 0.5 means random chance.**

**AUC can be interpreted as evaluating the ranking of positive examples.**

**What’s the probability that a randomly picked positive point has a higher score according to the classifier than a randomly picked point from the negative class.**

**AUC of 1.0 means all positive points have a higher score than all negative points.**