Streams Processing

Imbalanced learning for Streams

Imbalanced datasets: why are they a problem?

- Class imbalance and concept drift can significantly hinder predictive performance
- drift detection algorithms based on the traditional classification error may be sensitive to the imbalanced ratio and become less effective
- class imbalance techniques need to be adaptive to changing imbalance rates; otherwise, the class receiving the preferential treatment may not be the correct minority class at the current moment

Metrics

- Precision
- Recall
- F1
- Geometric mean
- …

Why not use accuracy?
Example with binary classification

$$\text{Accuracy} = \frac{\#\text{Correct Predictions}}{\#\text{Predictions}}$$

- Consider a credit card fraud dataset where Fraud happens in 0.01% of the examples;

- Lets use as predictive model the constant predictor

- Q: what is the best constant predictor for this data in terms of accuracy? Compute the exact accuracy of both in our dataset

$$f_0(x) = \text{Not Fraud} \qquad\qquad f_1(x) = \text{Fraud}$$

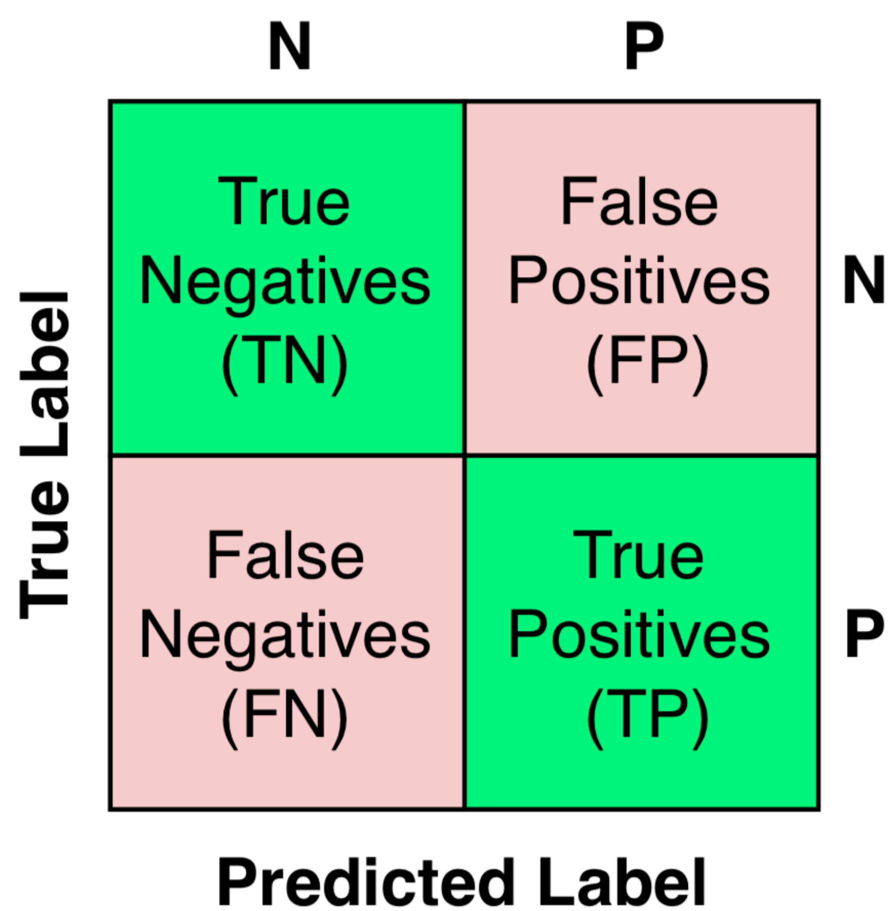
Why not use accuracy? Example with binary classification

$$\text{Accuracy} = \frac{\#\text{Correct Predictions}}{\#\text{Predictions}}$$

$$f_0(x) = \text{Not Fraud} \qquad\qquad f_1(x) = \text{Fraud}$$

- Why is this a bad predictor?
- What can we do about it?

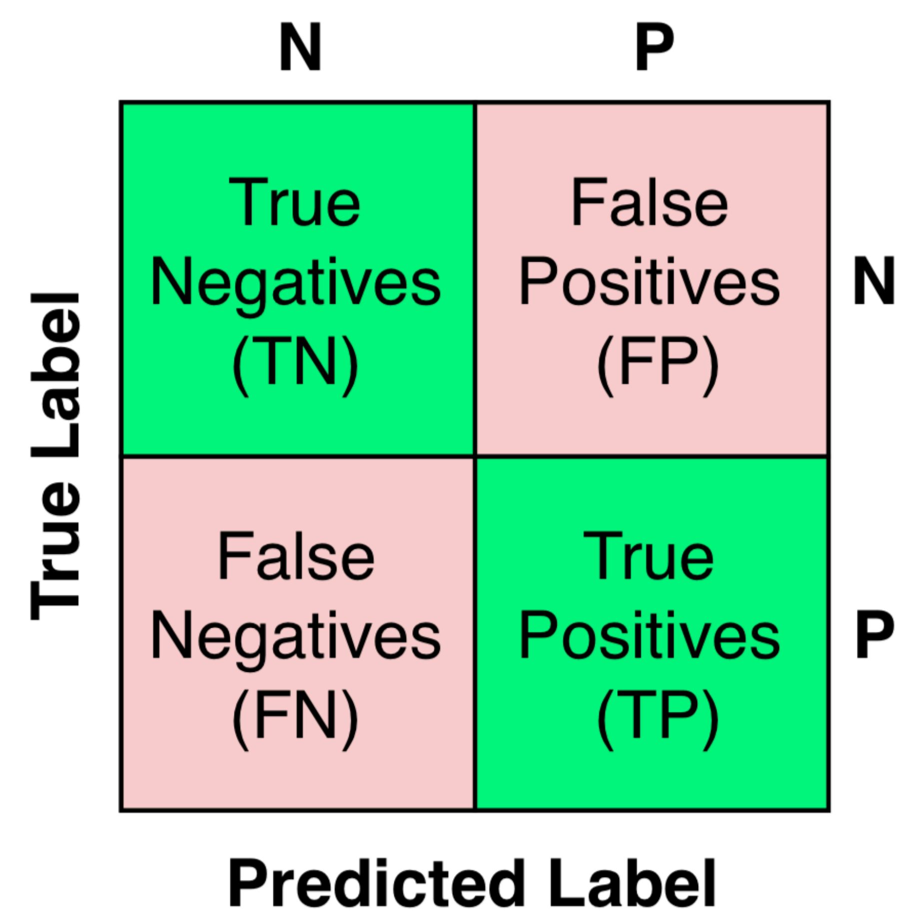
Confusion matrix



Binary case

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

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



Binary case

$$F1 = \frac{2TP}{2TP + FP + FN}$$

(Harmonic mean of P and R)

True Label \ Predicted Label	N	P
N	True Negatives (TN)	False Positives (FP)
P	False Negatives (FN)	True Positives (TP)

Matthews Correlation Coefficient (MCC)

$$MCC = \frac{TP \cdot TN - FP \cdot FN}{\sqrt{(TP + FP)(TP + FN)(TN + FP)(TN + FN)}}$$

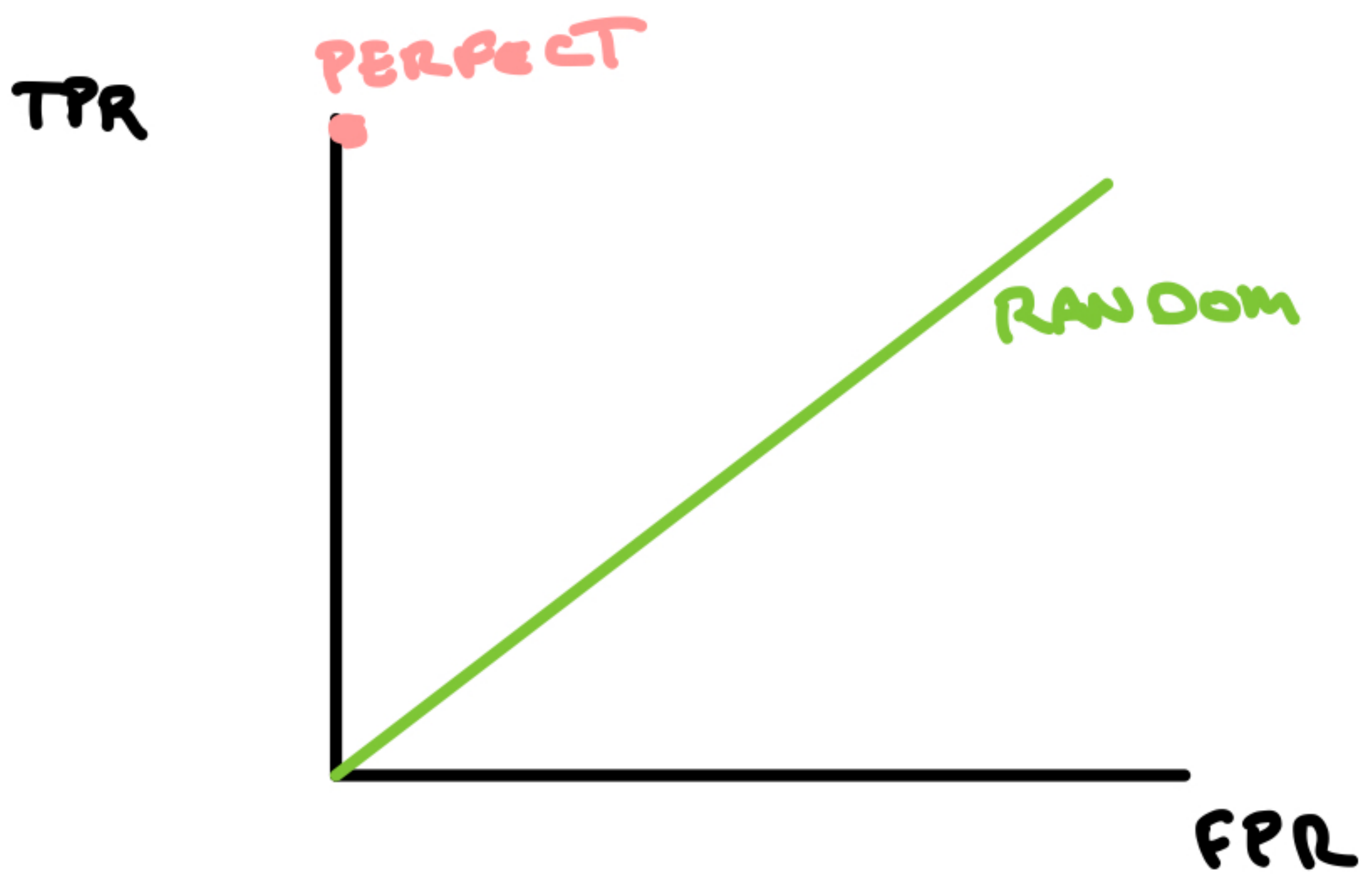
ROC

- Receiver Operating Characteristic
- Instead of training a classifier at a specific imbalance ratio, the classifier is trained over all possible imbalance ratios
- For all imbalance levels we measure the
    - True Positive Rate (TPR): proportion of positive examples assigned as positive. AKA sensitivity and Recall
    - False Positive Rate (FPR):

$$\mathrm{TPR} = \frac{\mathrm{TP}}{\mathrm{TP} + \mathrm{FN}}$$

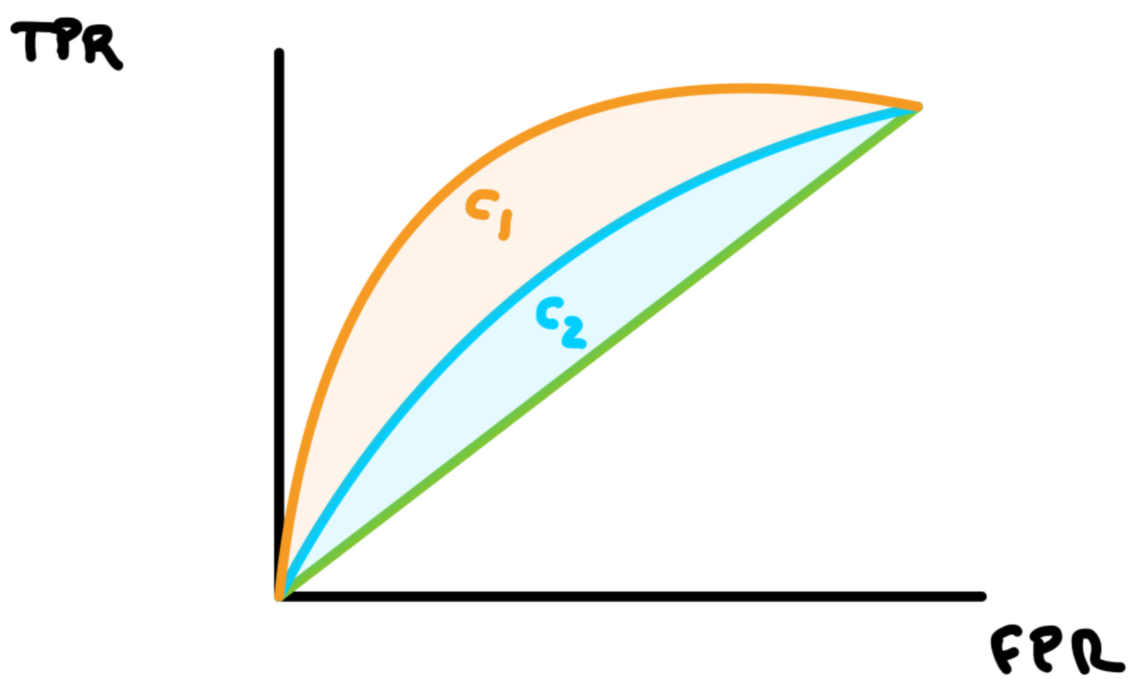
$$\mathrm{FPR} = \frac{\mathrm{FP}}{\mathrm{FP} + \mathrm{TN}}$$

ROC curve evaluation: the perfect and the random classifiers



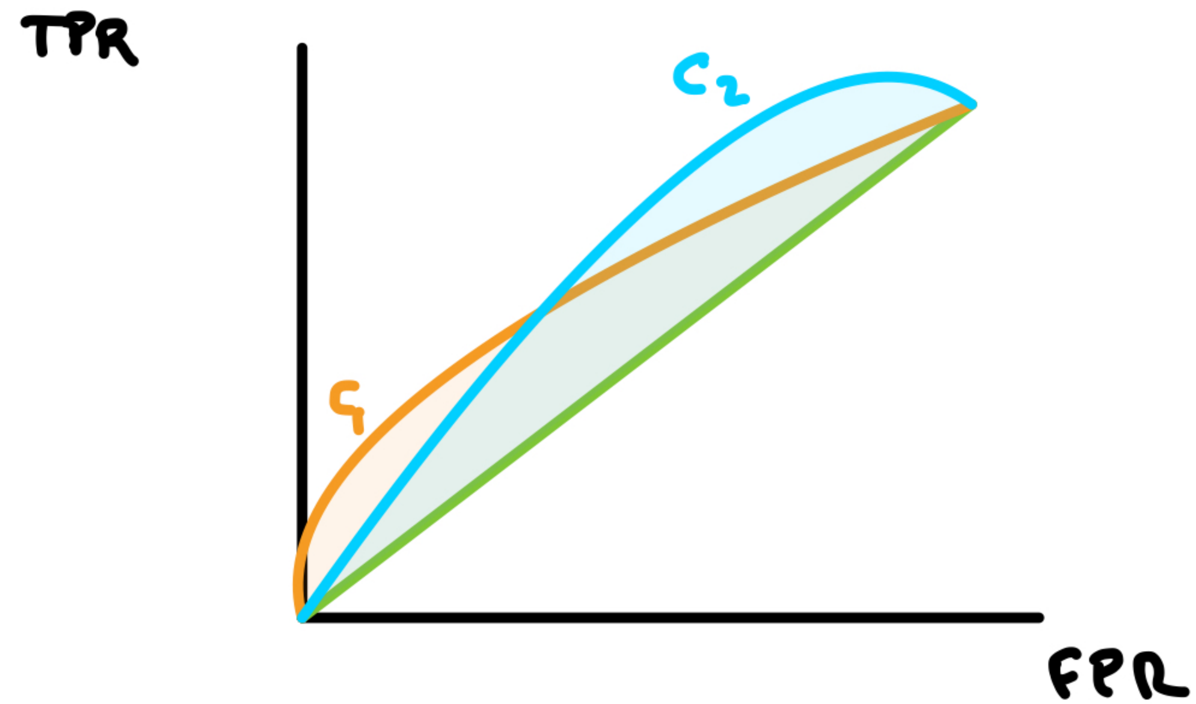
ROC curve evaluation

Sometimes



C1 is always better than C2

Frequently



For some imbalance ratios,
C1 is better than C2

How to address class imbalance?

Data level		Algorithm level	
Data sampling	Feature selection		
Oversampling		Cost sensitive	Ensemble methods
Under-sampling	Data augmentation		

Changing the data

Random over-sampling (ROS)

Random under-sampling (RUS)

Synthetic Minority Over-Sampling Technique (SMOTE)

Generative models:

- Sample-based: Generative Adversarial Network (GAN)
- Model-based: GMM

Random over-sampling

- Acts on the minority class
- Samples the minority class randomly, with reposition, until classes are balanced
- May lead to overfitting to the existing data

Random under-sampling

- Acts on the majority class
- Randomly removes samples from the majority class until both classes are balanced
- Increases variance on the estimator because information is lost

SMOTE

- Acts on the minority class
- Computes k nearest neighbors in the minority class, for each minority example
- Generates artificial data points in the line segments to all or a few of the nearest neighbors
- Can create wrong data

Imbalance-aware algorithms

- Increased cost for misclassification of the minority class
- Ensemble techniques
- Hybrid with data modification

Cost sensitive learning

- Weight differently residuals from minority and majority class
- Use imbalance ratio
- Equivalent to oversampling of the minority class
- In practice very effective

Ensemble methods

- Random forest
- Bayesian optimization

Random forest

- Bag of weak learners
- Each learner is trained with the same number of majority and minority data points
- Inference done by majority vote of all learners opinions

Technological aspects

MapReduce systems tend to increase the imbalance problem

Spark-based systems allow for imbalance mitigation