Imbalanced Classes

FinTech

Lesson 11.3

Class Objectives

By the end of today's class, you will be able to:

- Define model evaluation metrics and understand the pros and cons of each metric as applied to different classification problems.
- Define class imbalance and understand why it presents a problem for classification models.
- Demonstrate the ability to `under-` and `oversample` data with imbalanced classes.
- Demonstrate the ability to plot a `precision-recall` curve and use it to compare different models.

Review of Classification Metrics

Confusion Matrix

This is a special kind of table for the visualization and quantification of model performance.

A *confusion matrix* is a way to tally and visualize the errors.

	Predicted: No (0)	**Predicted: Yes (1)**
Actual=No (0)	True Negatives (TN)	False Positive (FP)
Actual=Yes (1)	False Negative (FN)	True Positives (TP)

Confusion Matrix

We can use our confusion matrix to calculate the model's overall *accuracy.*

- *Accuracy* is the proportion of correct calls
- It is calculated as Acc = (TP+TN)/(TP+TN+FP+FN)
- Treats FP and FNs equally—an issue for unbalanced data

	Predicted: No (0)	**Predicted: Yes (1)**
Actual=No (0)	True Negatives (TN)	False Positive (FP)
Actual=Yes (1)	False Negative (FN)	True Positives (TP)

Confusion Matrix

We can use our confusion matrix to calculate the model's ***precision***.

- *Precision* is the proportion of positive calls that were correct.
- It is calculated as Precision = TP/(TP+FP), using the first column of the confusion matrix.
- A model with no FPs has perfect precision. All of its positive calls are correct!

If FPs are very undesirable, you want a high precision

	Predicted: No (0)	**Predicted: Yes (1)**
Actual=No (0)	True Negatives (TN)	False Positive (FP)
Actual=Yes (1)	False Negative (FN)	True Positives (TP)

Confusion Matrix

We can use our confusion matrix to calculate the model's ***recall***.

- Recall is the proportion of actually positive samples that were correct
- It is calculated as Recall = TP/(TP+FN), using the first row of the confusion matrix
- Recall is a critical metric for optimizing a model with unbalanced data
- A model with no FNs has perfect recall. All of the positive samples are correctly identified!
- Recall is sometimes called sensitivity

If FNs are very undesirable, you want a model with high recall

	Predicted: No (0)	**Predicted: Yes (1)**
Actual=No (0)	True Negatives (TN)	False Positive (FP)
Actual=Yes (1)	False Negative (FN)	True Positives (TP)

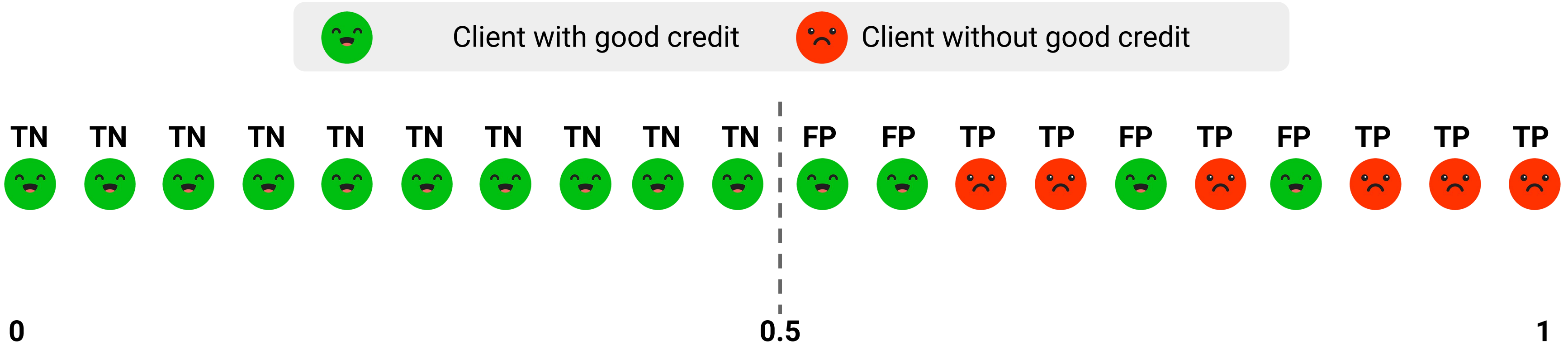
It is very rare that you will develop a model with high precision and recall. The two metrics are often in opposition–when one goes up, the other frequently goes down.

Confusion Matrix

When evaluating a model, report both precision and recall.

Consider the credit example. The model predicts that an individual has good credit if its probability (i.e., from a logistic regression or SVM) of good credit is ≥ 0.5.

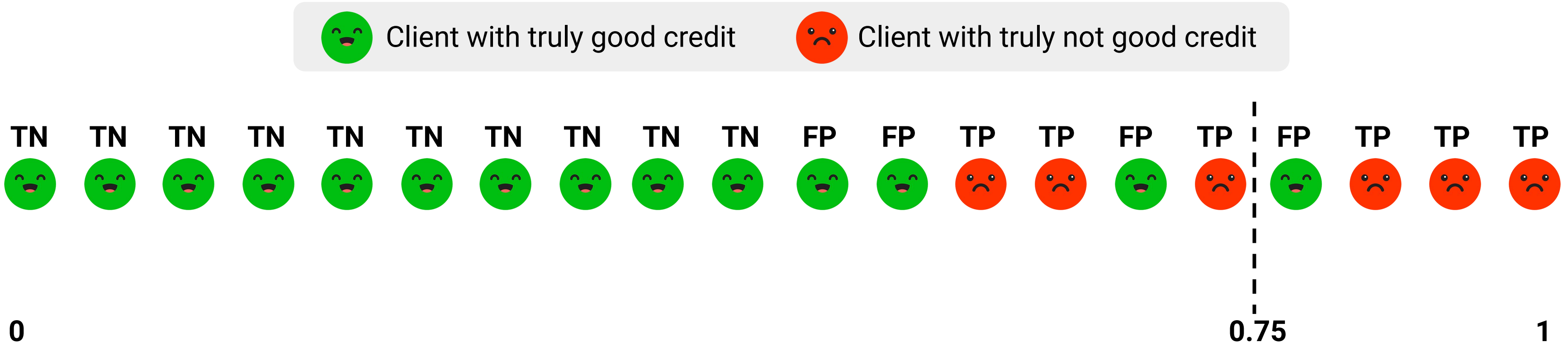
Probability that a client has good credit



Confusion Matrix

If we shift this probability threshold from 0.5 to 0.75, we may increase our FNs, but decrease the FPs.

Probability that a client has good credit



Confusion Matrix

We can use our confusion matrix to calculate the model's specificity.

- Specificity is the proportion of actually negative samples that were correct.
- It is calculated as Specificity = TN/(TN+FP), using the second row of the confusion matrix.
- A model with no FPs has perfect specificity. All of the negative samples are correctly identified!
- Well-performing models with lots of TNs (>10,000) will often have very high specificity (>0.99).

💎 **If FPs are very undesirable, you want a highly specific model.**

	Predicted True	Predicted False
Actually True	TP	FN
Actually False	FP	TN

Confusion Matrix

We can use our confusion matrix to calculate the model's ***F1-score***.

- The *F1-score* (or F-measure) is another overall accuracy measure equivalent to the harmonic mean of the precision and recall
- It is calculated as F1 = 2* (Precision * Recall)/(Precision + Recall), using the first row and column of the confusion matrix.
- A model with perfect precision and recall has an F1-score of 1.0. The F1-score gives equal weight to precision and recall. Note that if either are 0, the score is 0 too.
- It is a good summary metric for comparing one model's performance to another.

	Predicted: No (0)	**Predicted: Yes (1)**
Actual=No (0)	True Negatives (TN)	False Positive (FP)
Actual=Yes (1)	False Negative (FN)	True Positives (TP)

Activity:
Hypothetical Models

Suggested Time:
15 minutes

Time’s Up! Let’s Review.

Activity: Hypothetical Models

01

A company wants to block phishing messages:
Predict whether an email is spam or not spam.

02

Doctors want an objective second opinion on imaging results:
Predict whether or not an MRI shows cancerous growth.

03

A study looks at gender differences in writing:
Predict whether a student is a boy or a girl, based on their essays.

04

Improve weather forecasts:
Predict whether or not it will rain the next day.

05

A venture capital firm wants to optimize its investments:
Predict whether a company will file an IPO in the next 12 months.

Imbalanced Data

What’s wrong with imbalanced classes?

Models are biased toward the majority class.
Evaluation metrics, such as accuracy, are misleading.

Dealing with imbalanced classes

Potential strategies:

- Oversampling and undersampling
- Use the right performance metrics for evaluation
- Change your model

Oversampling

Potential strategies:

Add additional samples of the minority class until

---

instances of minority = instances of majority.

Random oversampling: Randomly choose minority class instances (with replacement).

SMOTE (synthetic minority oversampling technique): Creates synthetic data from minority samples through K-nearest neighbors.

Activity: More Loans

In this activity, you will practice using random and SMOTE oversampling, in combination with logistic regression, to predict the likelihood of someone defaulting on their credit card loans in a given month.

Suggested Time:
15 minutes

Time's Up! Let's Review.

Undersampling

Potential strategies:

Take out instances of the majority class until

----

instances of minority = instances of majority.

Random undersampling: Randomly choose majority class instances to take out of the training set.

Cluster centroid: Undersampling: first create N clusters, when N is the number of minority class training instances; then take the centroids from those clusters as the majority class training data.

Activity: Undersampling

In this activity, you will research and practice undersampling with the `imbalanced-learn` library.

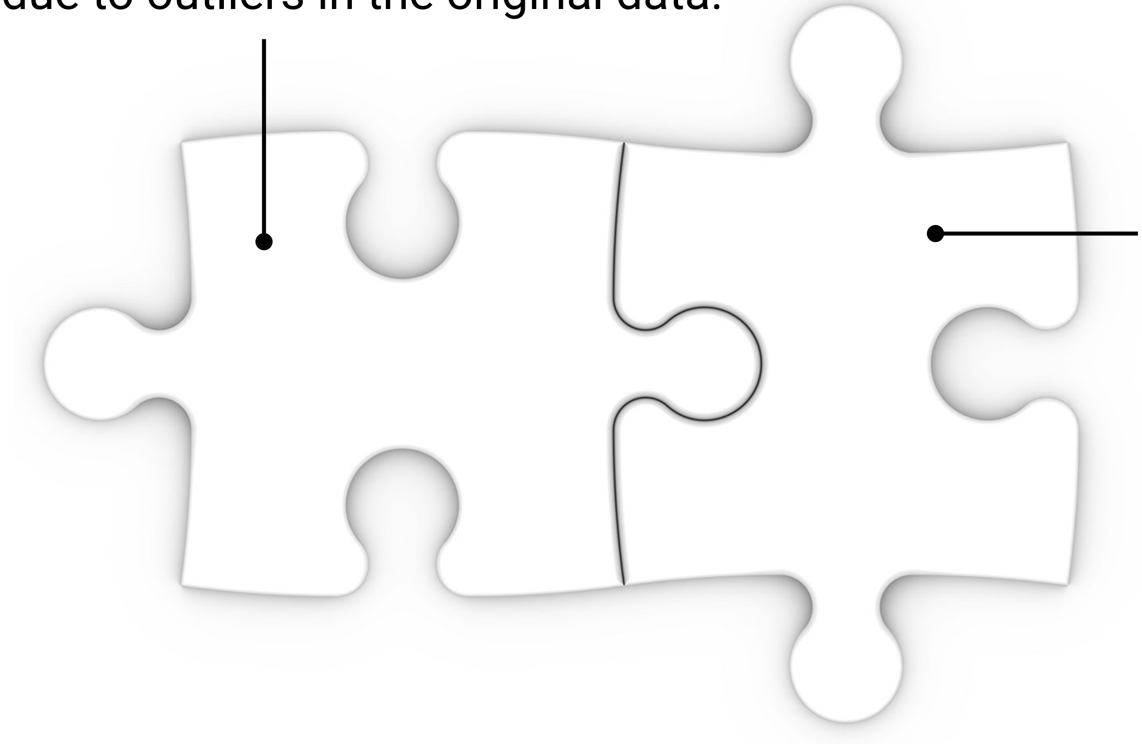
Suggested Time:
15 minutes

Time's Up! Let's Review.


Break

Combination Sampling

Oversampling with `SMOTE` can result in noisy data due to outliers in the original data.



Undersampling is not always realistic due to limited data.

Combination Sampling

`SMOTEENN` “combines” the two methodologies.

01 **oversampling**

First, we oversample the minority class.

02 **undersampling**

Next, we “clean” the resulting data using an undersampling strategy: If a data points’ two nearest neighbors are in a different class, then we drop that data point.

Activity: Combination Sampling

In this activity, you will practice combination sampling with the `imbalanced-learn` library.

Suggested Time:
15 minutes

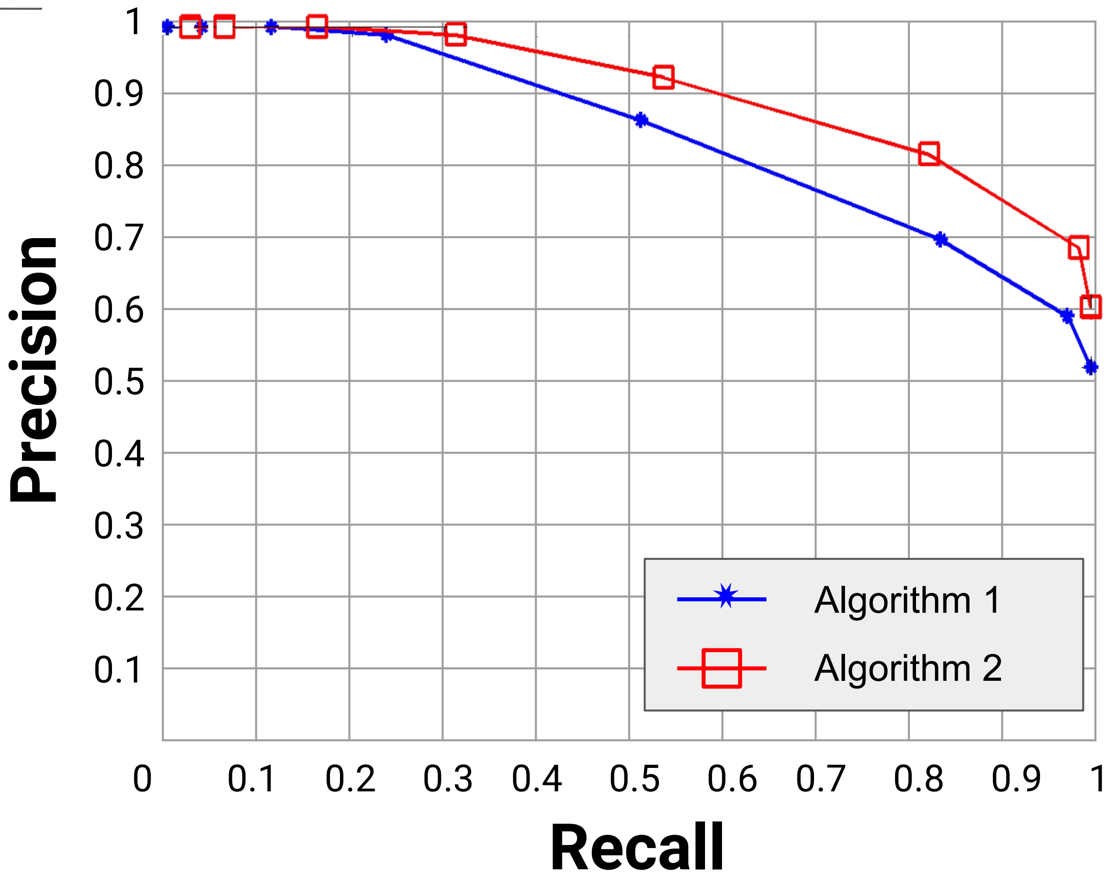
Time's Up! Let's Review.

Precision-Recall Curve

The balance between precision and recall can be visualized with a `precision-recall` curve.

Precision-Recall Curve

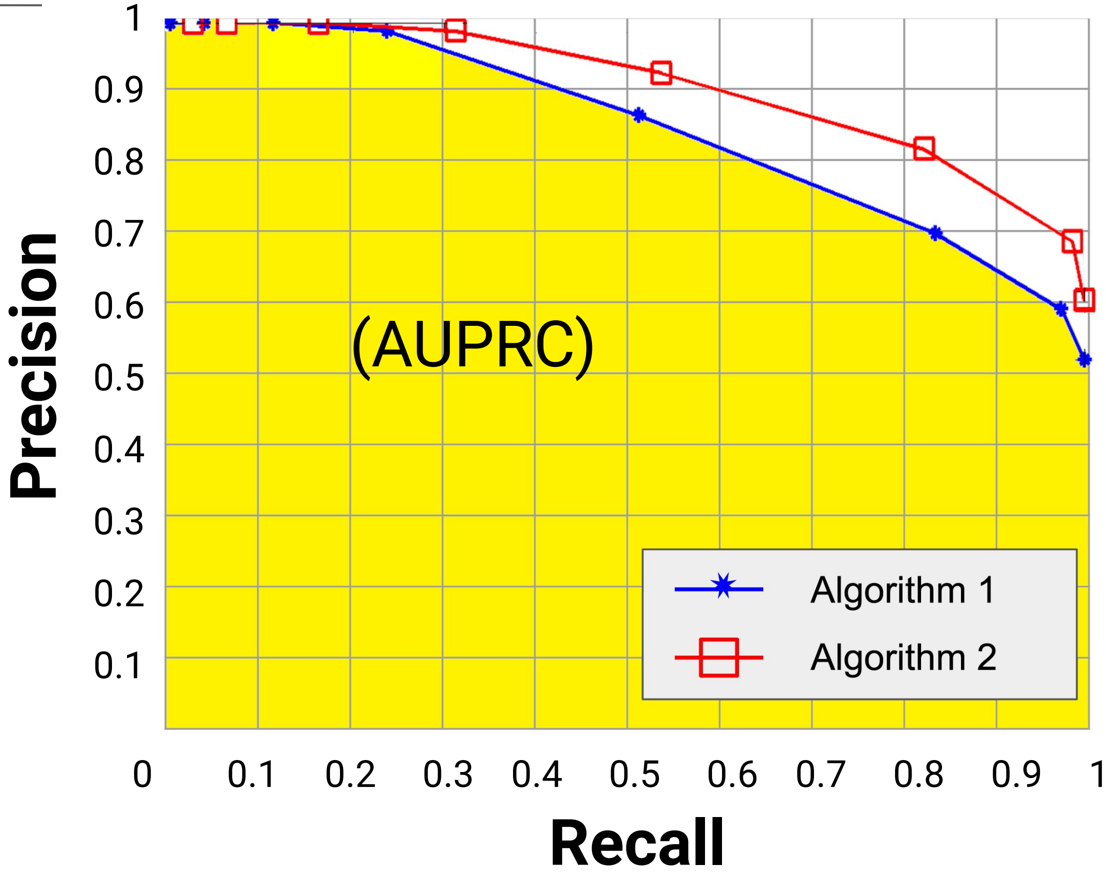
A PR curve plots recall (x-axis) versus precision (y-axis) at various classification thresholds to help visualize the balance between these metrics.



Precision-Recall Curve

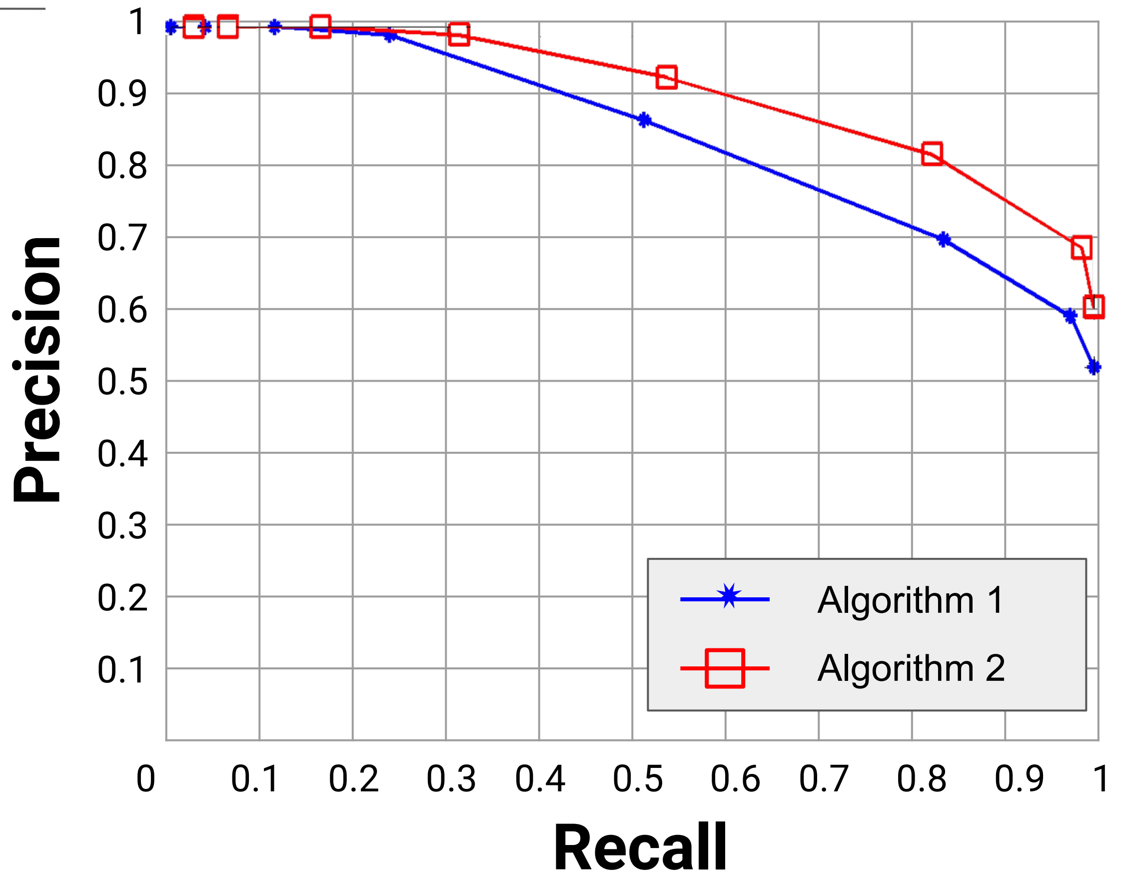
The area under the PR curve (AUPRC) is a metric for how good the model is in absolute terms.

It can be valuable when comparing one model to another.



Precision-Recall Curve

Note how, in general, as the recall increases (right) the precision decreases (down).



Activity:

Credit Card Fraud

In this activity, you will practice resampling techniques and use different models to classify credit card transactions as fraudulent or not fraudulent.

Suggested Time:

15 minutes

Time's Up! Let's Review.

Questions?