## Wifi connection

Network: Free_STCC

User ID : 2317791957

Password: 5197

# AMLD2021
# Unsupervised fraud detection

Giulio Ghirardo
Doodle

Alessandro Scarpato
Julius Baer

Ernst Oldenhof
Julius Baer

Steffen Terhaar
D|ONE

## Workshop session contents

- (70 min) Presentation and exercises
- (90 min) Challenge: unsupervised prediction
- (10 minutes): Presentation best solution, wrap-up

## Fraud detection

- Global costs of fraud ~ 7% of total company expenditures ($5 \times 10^{12}$ USD)
- Examples: CEO fraud, faked identity, credit card theft, internal fraud

Fraud detection systems often combine business rules, network analysis and machine learning

Often there is little data to learn from (exception: credit card fraud)

Both supervised and unsupervised approaches are therefore used

https://www.crowe.com/global/news/fraud-costs-the-global-economy-over-us$5-trillion

## Keep in mind, unsupervised means:

No cross-validation / hyperparameter tuning

Very little general “best-practices”

Which algorithm performs well is strongly dependent on the data

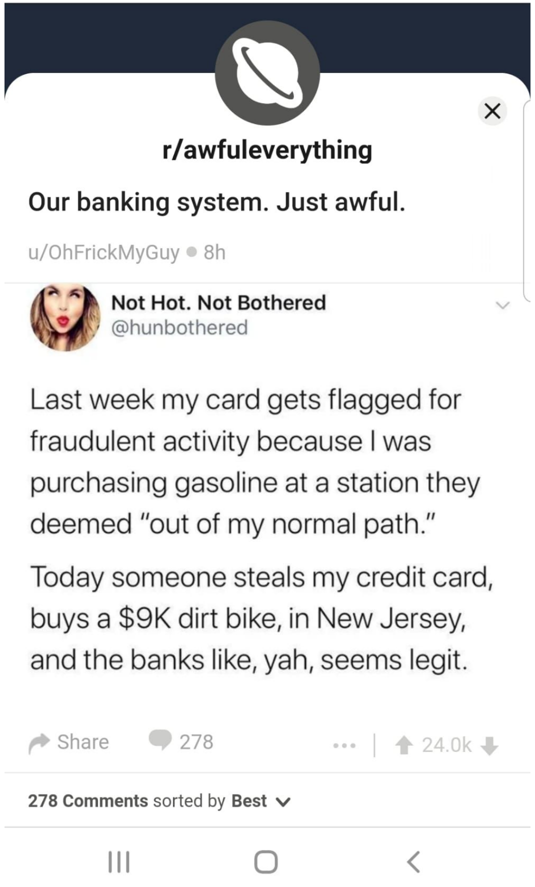
Curse of dimensionality / Noisy features

## Metrics for binary classification performance

Binary classification results (binary labels & predictions) can be summarized in a confusion matrix:

| | | **Actual** | |
|---|---|---|---|
| | | P | N |
| **Prediction** | P | True Positive | False Positive (“type-I”, false alarm) |
| | N | False Negative (“type-II”, oversight) | True Negative |

Ernst



r/awfuleverything

Our banking system. Just awful.

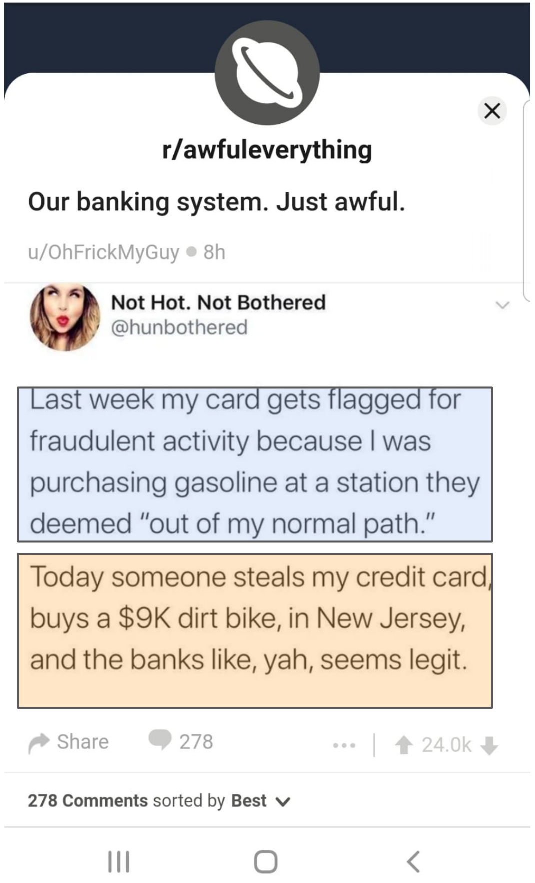
u/OhFrickMyGuy • 8h

Not Hot. Not Bothered
@hunbothered

Last week my card gets flagged for fraudulent activity because I was purchasing gasoline at a station they deemed "out of my normal path."

Today someone steals my credit card, buys a $9K dirt bike, in New Jersey, and the banks like, yah, seems legit.

**278 Comments** sorted by **Best**



r/awfuleverything

Our banking system. Just awful.

u/OhFrickMyGuy • 8h

Not Hot. Not Bothered
@hunbothered

Last week my card gets flagged for fraudulent activity because I was purchasing gasoline at a station they deemed "out of my normal path."

Today someone steals my credit card, buys a $9K dirt bike, in New Jersey, and the banks like, yah, seems legit.

False Positive: annoying

False Negative:
annoying + expensive

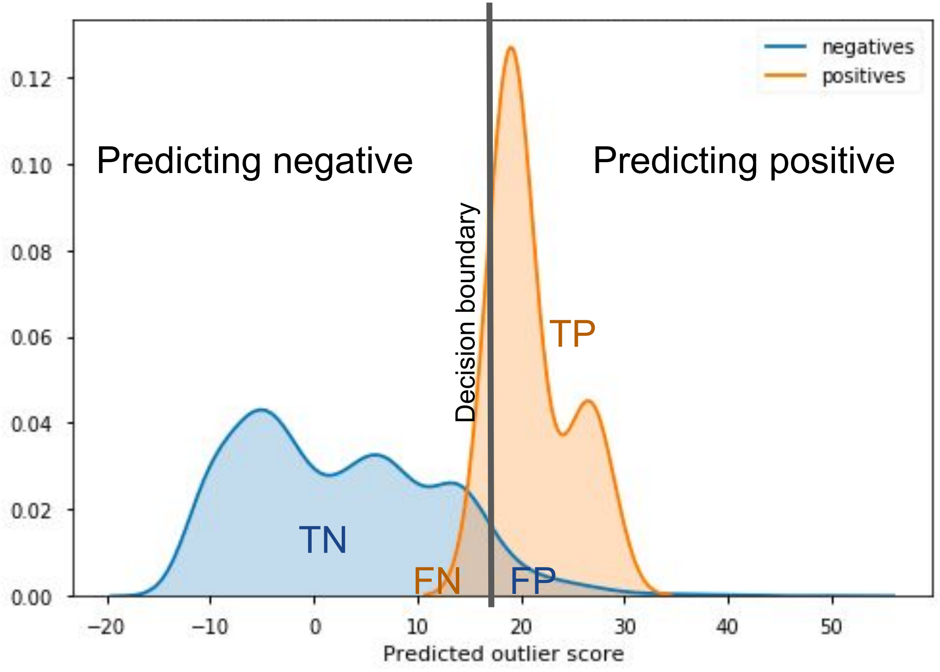
## Binary prediction metrics

Accuracy, F1-score, Matthew's correlation, ...

- All compress a confusion matrix (2x2) into a single number
- Note that accuracy is the correct measure when
  the cost of a False Positive = cost of a False Negative **(!)**

True Positive Rate, False Positive rate, Precision: consider only part (single row or column) of the confusion matrix

# Visualizing scoring: conditional score curves

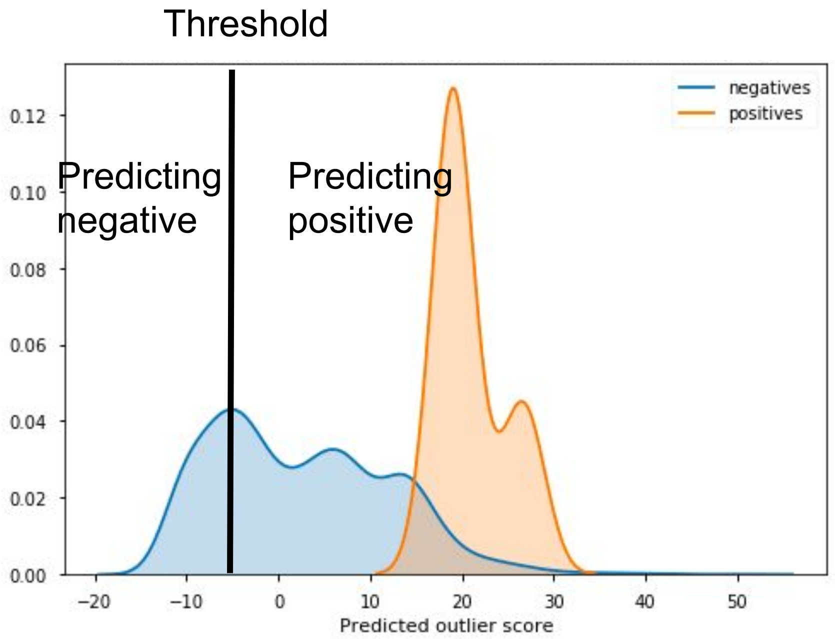


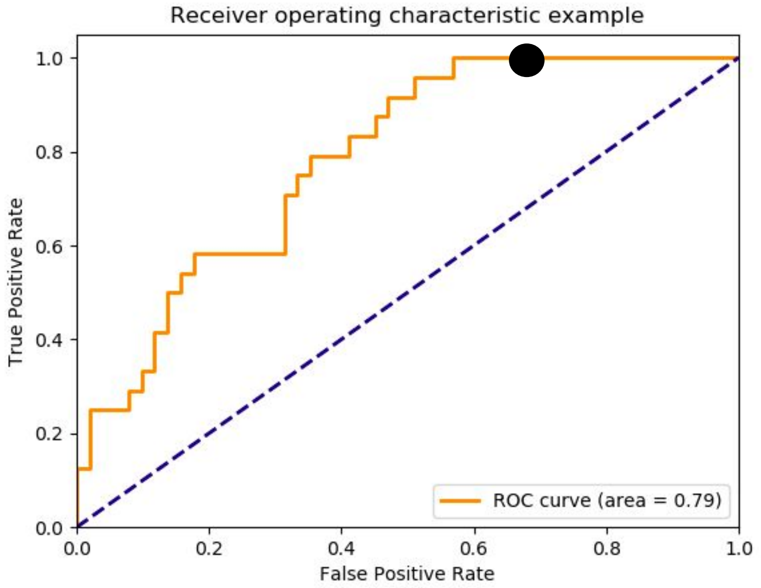
Recall (TPR): what fraction of the actual positives were correctly identified?
**TP / P**

False Positive Rate (FPR): what fraction of actual negatives was falsely predicted positive?
**FP / N**

Precision: what fraction of the predicted positives is actually positive?
**TP / (TP + FP)**

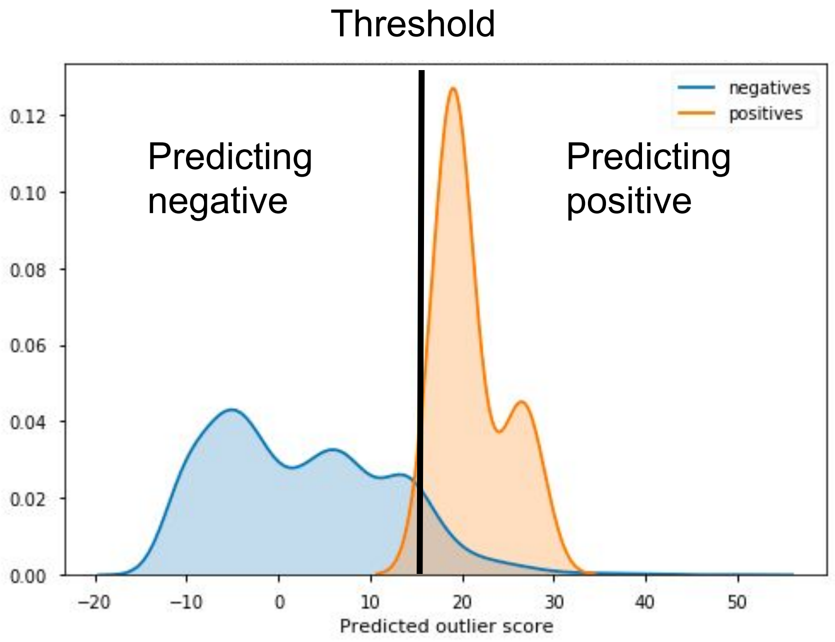
## Classification metrics for scores: AUC-ROC

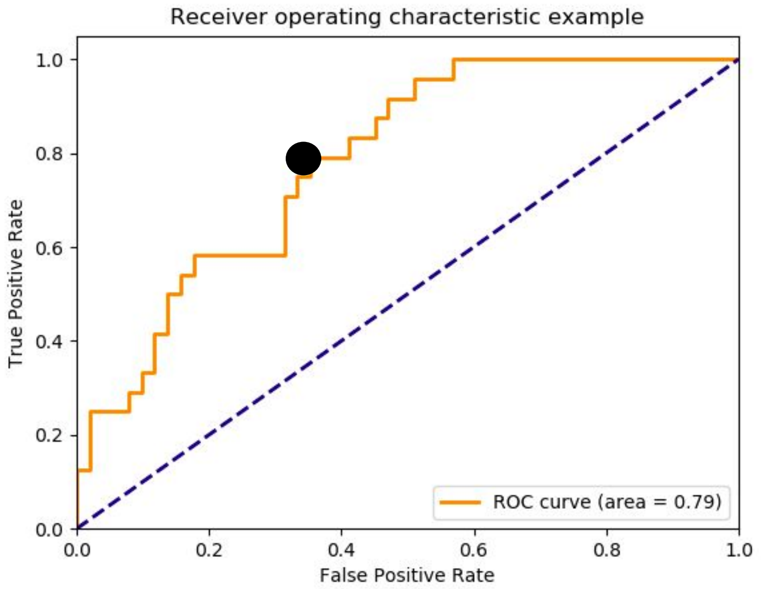




*Too low threshold: perfect recall (TPR = 1) but high FPR. False alerts!*

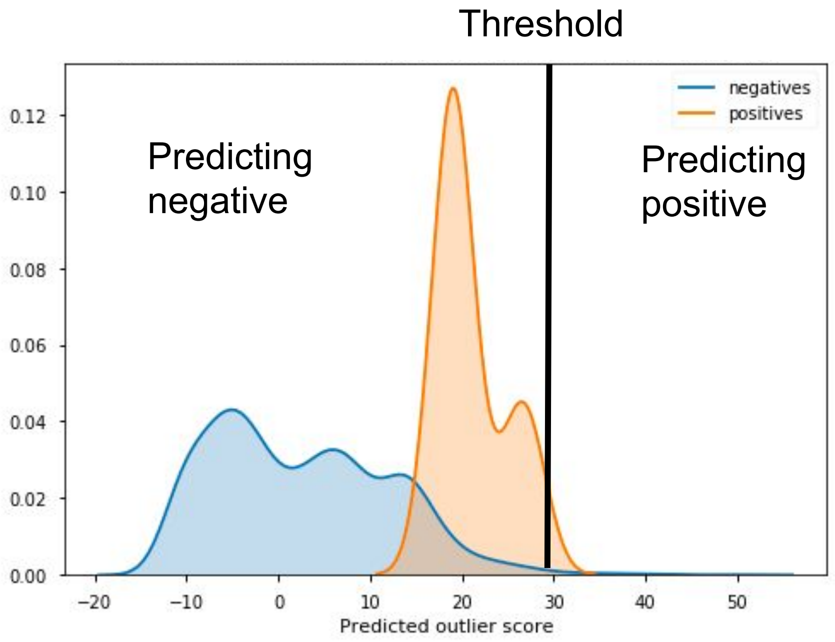
# Classification metrics for scores: AUC-ROC

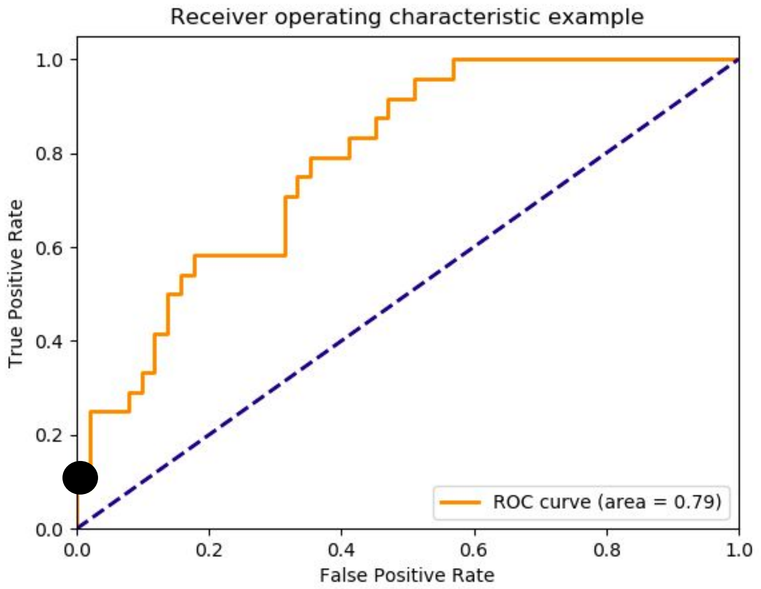




*Optimal threshold: high recall/TPR, low FPR*

# Classification metrics for scores: AUC-ROC





*Too high threshold: perfect FPR, but very low recall/TPR (false negatives!)*

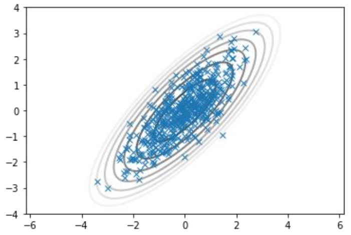
## Classification metrics for scores: AUC-ROC

- ROC-AUC: does not need the specification of a threshold
- Has a nice statistical interpretation: the probability that a randomly chosen actual positive-actual negative pair is correctly ranked
- Independent of class sizes (because of TPR/FPR)
- Baseline: 0.5 (random guessing)

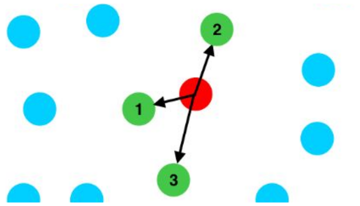
- A variation on the ROC-AUC: average precision-recall, AUC-PR
  *Note that AUC-PR has no statistical interpretation*

# Outlier detection algorithms (for tabular data)

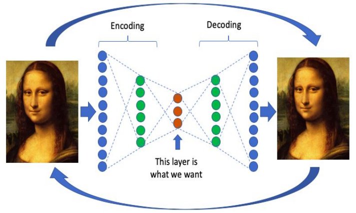
Mahalanobis



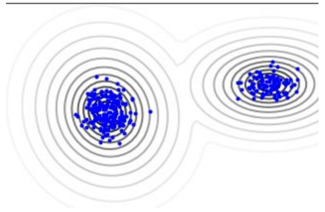
Distance/Density



Reconstruction



Clustering



- Deviation of a specified normal-form (Mahalanobis, regression, GPR)
- Distance- or density-based (kNN, LOF)
- Reconstruction error (PCA, Autoencoder)
- Clustering (GMM, DBSCAN)

*Special cases: One-class SVM, Isolation Forest*

Note that these algorithms are based on different ideas of what an outlier means (far away from other points, easily splittable, not part of a large cluster, …)

## Outlier detection in Python

Scikit-Learn

- Has a lot of algorithms (KNN, GMM, LOF, iForest, one-class SVM, Covariance)

PyOD

- User friendly unified API for 30+ algorithms (based on Scikit-Learn, Tensorflow,...)
    - `.fit()`
    - `.decision_scores_`

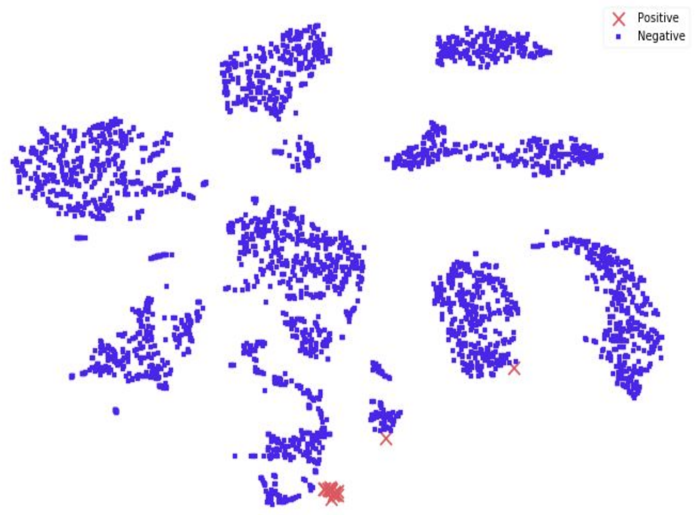
https://github.com/yzhao062/Pyod

## Exercise - data structure

- We use a (small and simple) pendigits dataset
- 10k points, 0.2% are outliers (4's are underrepresented class)

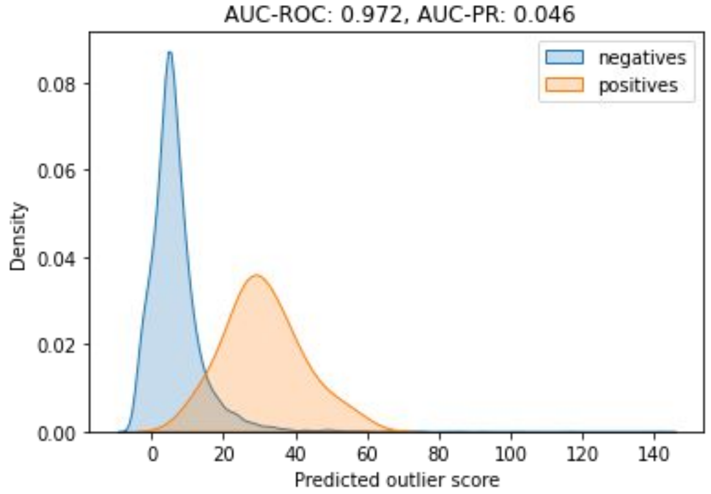
t-SNE 2D representation

- ~10 clusters, which corresponds to our data understanding
- positives (i.e., 4s) form their own cluster
- 90% of the neighbours of positives is indeed a positive



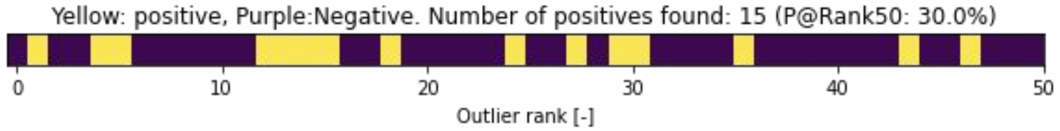
# Exercises - helper functions

`plot_outlier_scores()`



Conditional outlier score curves, AUC-ROC and AUC-AP

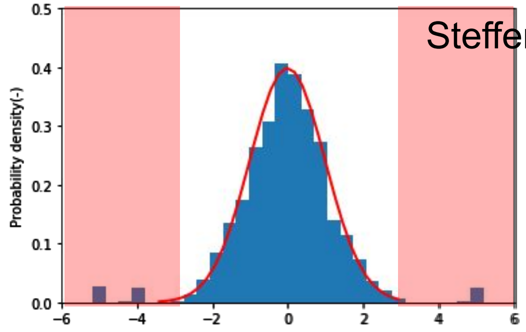
`plot_top_N()`

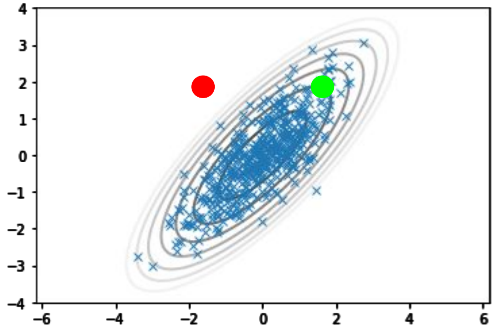


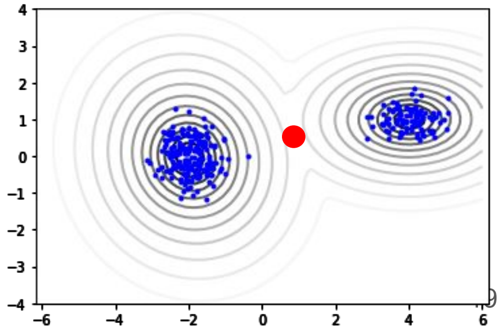
True labels of the top-N predictions, Precision@N

# Mahalanobis distance and GMM

- Univariate
  - z-score
- Multivariate
  - Features are uncorrelated and standardized (e.g., PCA coefficients) → Euclidean distance
  - Features are correlated→ Mahalanobis distance
- Multi-modal
  - Gaussian Mixture Model (GMM)
  - Number (and shape) of components is hyperparameter







## Work on your own - 10 min

Exercise: Mahalanobis distance and GMM

After 10 min we wrap up and show the solution

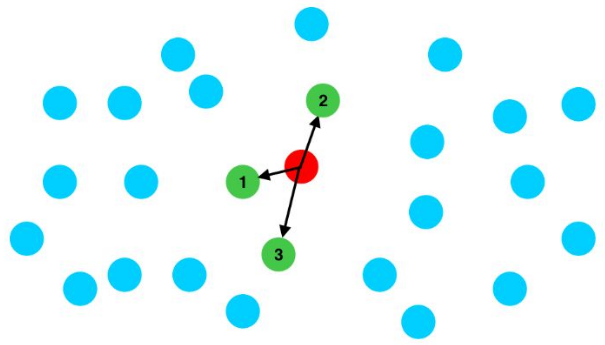
Github
https://github.com/DonErnesto/amld2021-unsupervised

Exercise (Colab)
https://colab.research.google.com/github/DonErnesto/amld2021-unsupervised/blob/master/notebooks/exercises.ipynb

## k-Nearest Neighbour (KNN)

How far is one point from:

- its closest neighbor?
- Its k-th closest neighbor?
- All k-th closest neighbors?



This distance defines the score: the higher the distance the further the point -> the higher the chance that it is an outlier. One can choose Euclidean or any other metric for the distance formula

No training phase, but prediction time scales with dataset size N:

- Brute search: time complexity N^2 (for N points)
- K-D/Ball trees: best case N log(N)

https://blog.mapbox.com/a-dive-into-spatial-search-algorithms-ebd0c5e39d2a

Giulio

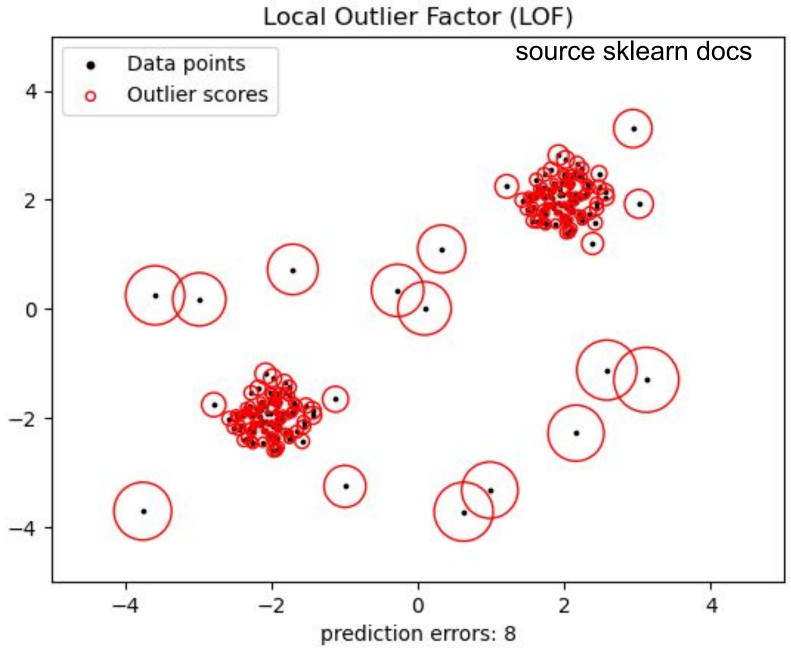
# LOF (Local Outlier Factor)

Largest distance of a point to its closest k points (*) → Reachability density ~ 1/avg. distance on the k-nearest points → Local Outlier Factor: how lower is the reachability density of a point with respect to all its k neighbours (avg. of ratios)

It estimates the degree of isolation of an object from its surrounding neighborhood of closest k objects:

- Clear clusters → LOF = 1
- Outliers → LOF >> 1

Recommended to use K > 10



Paper link https://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.1078.3580&rep=rep1&type=pdf

Giulio

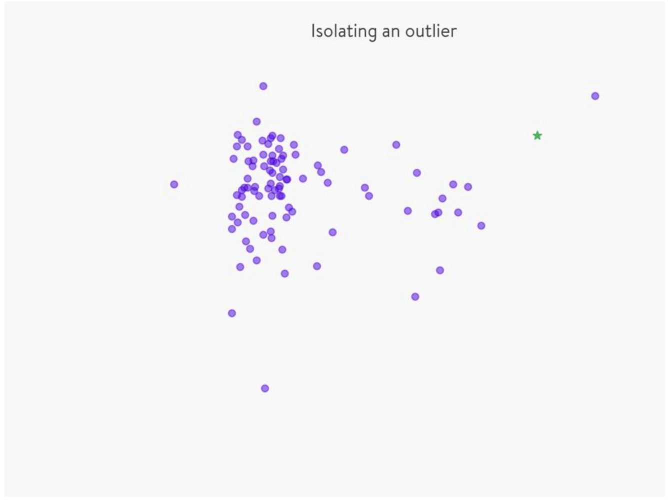
## Work on your own - 10 min

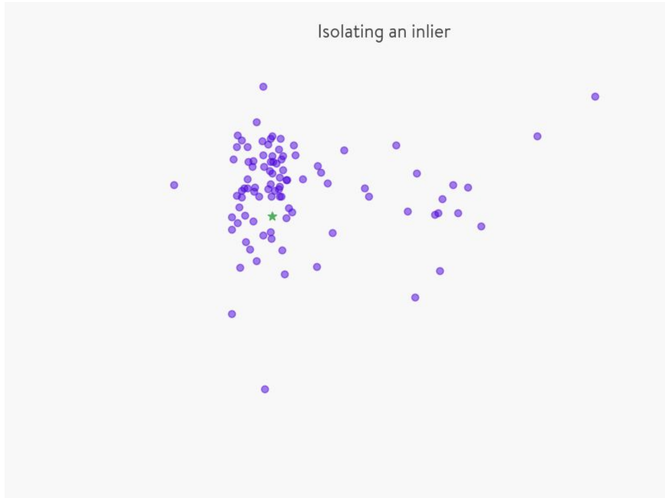
Only KNN, LOF

after 10min+- wrap up and show solutions

# Isolation Forest

- Based on outlier “isolation” compared to normal data
- Low memory requirement and fast execution (linear time complexity)
- Outliers are identified thanks to the average path length (they are closer to the tree root)
- Parameter to tune: sub-sampling size





Liu, F. T., Ting, K. M., & Zhou, Z. H. (2008, December). Isolation forest. In *Data Mining, 2008. ICDM’08. Eighth IEEE International Conference on* (pp. 413–422). IEEE.

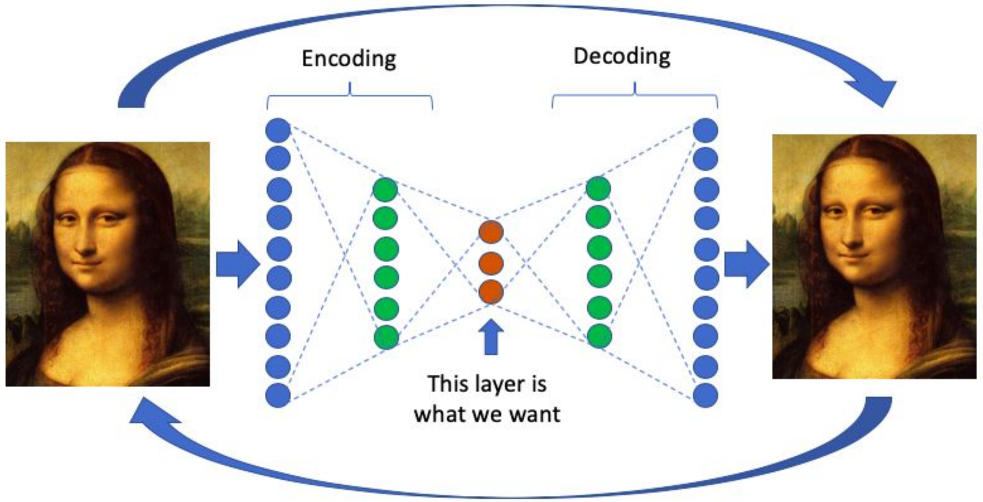
Hariri, S. Isolation Forest for Anomaly Detection. *LSST Workshop 2018, June 21, NCSA, UIUC.*

## Work on your own - 10 min

Only iForest

## Reconstruction Errors: Autoencoders / PCA

- Dimensionality reduction, and reconstruction
- Reconstruction error is globally minimized by PCA and Autoencoder
- Individual large reconstruction error may indicate outlier
- Hyperparameters
  - PCA: number of components
  - Autoencoder: Layer sizes, ...)



https://towardsdatascience.com/anomaly-detection-with-autoencoder-b4cdce4866a6

Alessandro

## Work on your own - 10 min

# Pen-digits exercise: wrap-up

| Algorithm | AUC-ROC | Tuning Complexity | Scalability | Comments |
|---|---|---|---|---|
| kNN | 0.98 | Single tuning parameter | OK with N, poor with dimensionality | Good fit with data characteristics |
| GMM | 0.96 | Single tuning parameter | Poor, too many parameters to fit for high dim data | Good fit with data characteristics |
| LOF | 0.93 | Single tuning parameter | OK with N, poor with dimensionality | Works poorly with N<20 |
| PCA | 0.88 | Difficult (heuristic: number of components N/2+1) | There are fast algos for high dim data | Single tuning parameter but less intuitive |
| Autoencoder | 0.88 | Difficult (heuristic: layer size N/2+1) | Scales well | Needs some assumptions on architecture, layer size, activation functions |
| iForest | 0.88 | Practically parameter-free | Linear | |
| Mahalanobis | 0.81 | Parameter-free | Curse of dimensionality | |

## Workshop challenge - Colab link here

- the original dataset comes from the KDD 1999 Cup with ~48k rows
- we provide a cleaned dataset for you to work with (min-max scaled, one-hot encoded, ...)

Your goal is to find the outliers in the dataset and report them to our server:

1) You tell the server which rows of the dataset you think are outliers

   for example, you tell the server that rows number 512, 10’156, 35’054 and 78’988 are outliers

2) The server will reply to you on which rows you were right, and keep track of your score: +500 for a correct prediction (true positive) and -25 for a wrong prediction (false positive)

   for example, the server will tell you that on row number 10’156 you were right, but the other 3 are inliers and you were wrong. Your total score after submission is 500 - 25*3 = 425

3) May the team with the highest score win!

**Live demo of a submission**

## Challenge

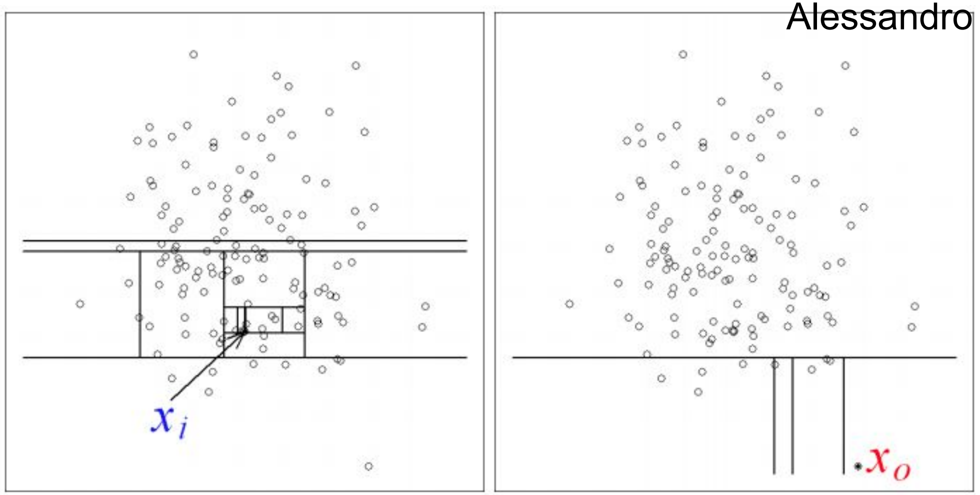
- 15 min
- 30 min
- ...

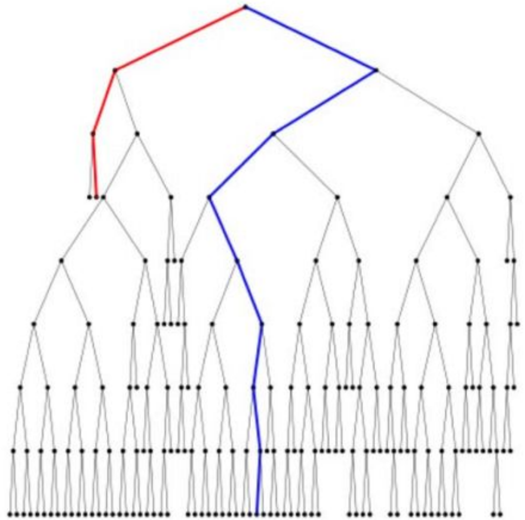
## Links

## Backup

## Isolation Forest

- Based on outlier "isolation" compared to normal data
- Low memory requirement and fast execution (linear time complexity)
- Outliers are identified thanks to the average path length (they are closer to the tree root)
- Parameter to tune: sub-sampling size



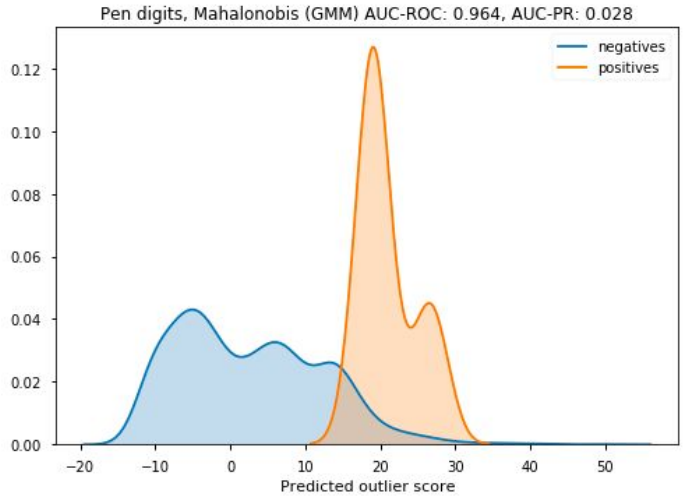


Liu, F. T., Ting, K. M., & Zhou, Z. H. (2008, December). Isolation forest. In *Data Mining, 2008. ICDM'08. Eighth IEEE International Conference on* (pp. 413–422). IEEE.

Hariri, S. Isolation Forest for Anomaly Detection. *LSST Workshop 2018, June 21, NCSA, UIUC.*

# Exercise - algorithm performance

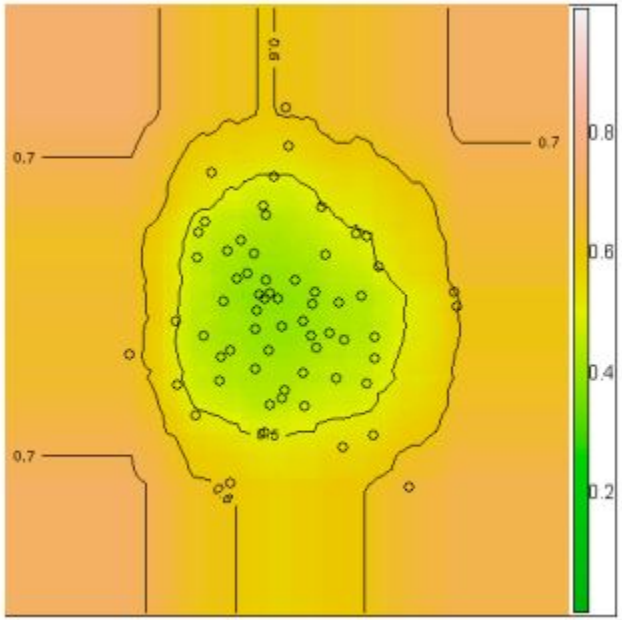
- AUC-ROC scores for KNN and GMM very high ( > 0.96)
- These fit the data characteristics
- LOF works very poorly with small N (<20) → makes sense, given data and algorithm
- AUC-PR low (3%-6%) due to very low fraction of positives. Still significantly exceeding random prediction (0.2%)



## Backups/Archive

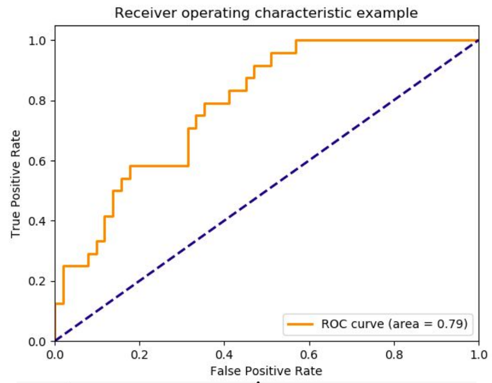
# Extra: Isolation Forest

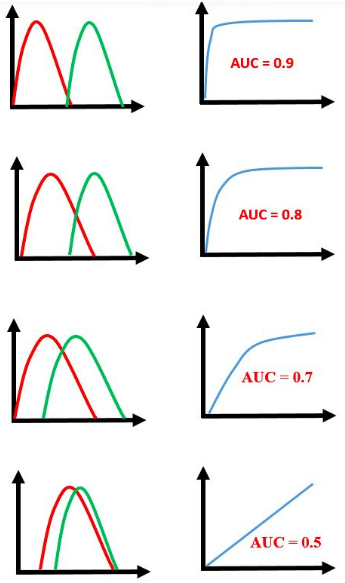
Artefacts of the Isolation Forest algorithm: contours following the coordinate axes



## Aggregated metrics for outlier scoring

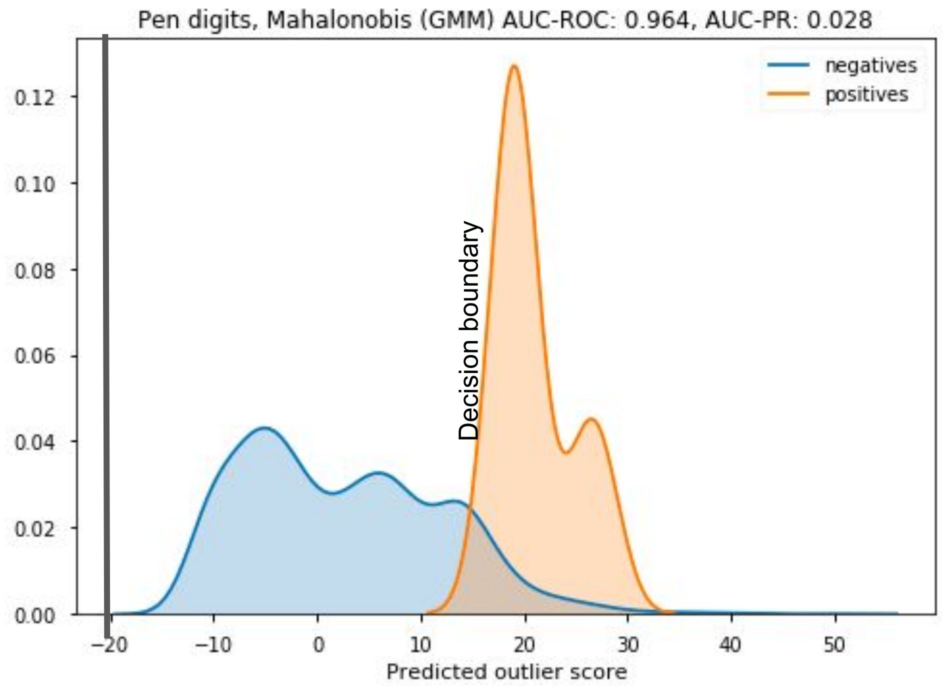
- AUC-ROC (Area under the ROC curve)
  - Most popular for binary prediction
  - Independent of class balances, true measure for classifier performance
  - Probabilistic interpretation: Probability that a randomly chosen (Positive, Negative) pair is correctly scored
  - Baseline: 0.5 (random guessing)
  - But: Equal weight is given to the negative class

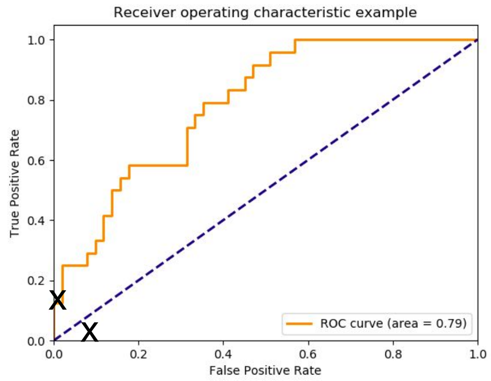


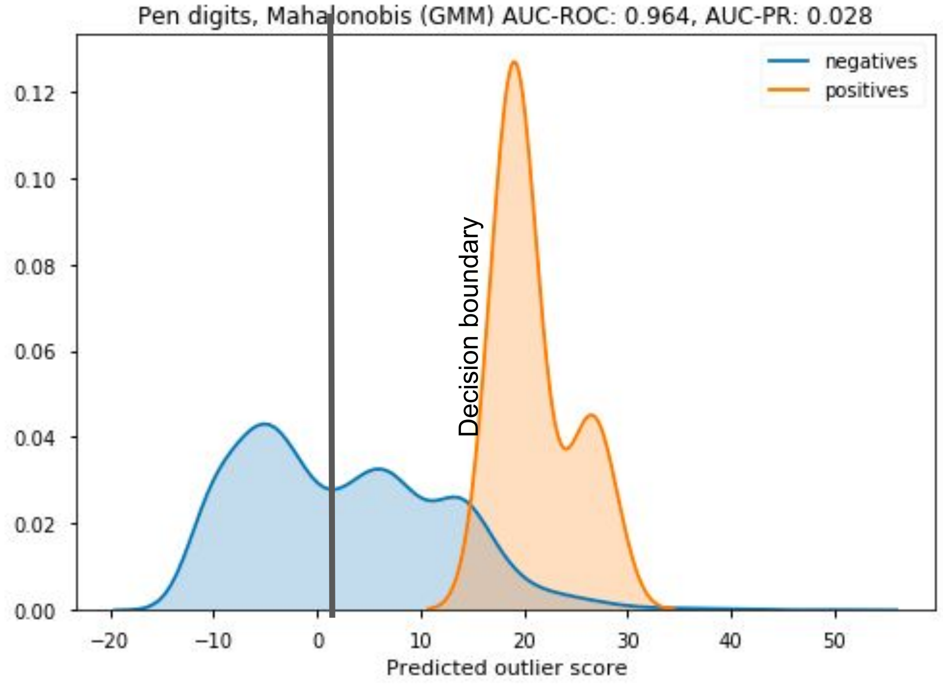


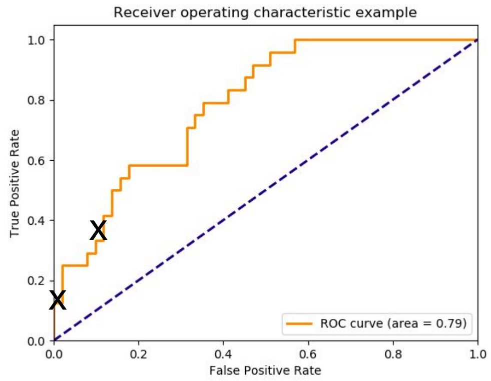
https://scikit-learn.org/stable/auto_examples/plot_roc_curve_visualization_api.html#sphx-glr-auto-examples-plot-roc-curve-visualization-api-py

https://medium.com/greyatom/lets-learn-about-auc-roc-curve-4a94b4d88152


Pen digits, Mahalonobis (GMM) AUC-ROC: 0.964, AUC-PR: 0.028
negatives
positives
Decision boundary
Predicted outlier score
0.12
0.10
0.08
0.06
0.04
0.02
0.00
−20
−10
0
10
20
30
40
50

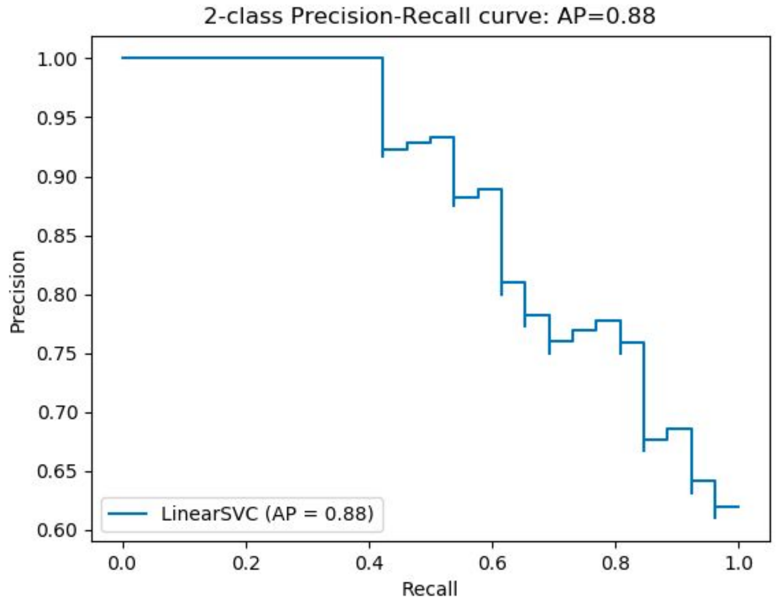

Receiver operating characteristic example
True Positive Rate
False Positive Rate
ROC curve (area = 0.79)
X
X
0.0
0.2
0.4
0.6
0.8
1.0


Pen digits, Mahalonobis (GMM) AUC-ROC: 0.964, AUC-PR: 0.028
negatives
positives
Decision boundary
0.12
0.10
0.08
0.06
0.04
0.02
0.00
−20
−10
0
10
20
30
40
50
Predicted outlier score


Receiver operating characteristic example
True Positive Rate
False Positive Rate
ROC curve (area = 0.79)
X
X
0.0
0.2
0.4
0.6
0.8
1.0

## Aggregated metrics for outlier scoring

- AUC-AP (Area under the Precision-Reca curve)
  - Recall and precision are both “business-relevant”
  - More sensitive to the positive class than the AUC-ROC
  - Baseline: fraction of positives (random guessing)
  - But: class-balance dependent, not universal



## Point-Metrics for outlier scoring

- Precision at Recall
  - Suitable business metric, because
    - Recall → effectiveness (captures cost of False Negative)
    - Precision → efficiency (captures cost of False Positive)
  - Less suitable for academic settings, no universal recall
- Precision at N
  - Popular in document retrieval
  - What N to choose?
- Minimal Cost
  - Most suitable when cost of FP and cost of FN are known (this is often not the case)
  - Note: iso-cost contour is a diagonal in ROC-curve

## Workshop contents

- (20 + 10 min) Presentation + Questions/buffer
- (10 min) Setup and discussion of Exercise
- (40 minutes) Exercise
- (5 minutes) Solutions to the exercise
- (90 min) Workshop challenge: unsupervised prediction
- (10 minutes): presentation best solution, wrap-up