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

AMLD2020
Unsupervised fraud detection

Giulio Ghirardo
Ernst Oldenhof
Alessandro Scarpato
Steffen Terhaar

Fraud detection

- Global costs of fraud ~ 7% of total company expenditures ($5x10^{12}$ 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

Anomaly detection: some definitions

There are two types of anomaly detection:

Novelty detection: finding points that are different from a base population

Outlier detection: finding points that “seem to be generated by a different mechanism” than the other points

The outliers/novelties are typically called positives

Novelty detection is semi-supervised (we have a collection of points we know to be negatives)

The definition of an outlier is not universal

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

Outlier detection algorithms (for tabular data)

- Deviation of a specified normal-form (regression, mahalanobis)
- Distance- or density-based (kNN, LOF)
- Reconstruction error (PCA, Autoencoder)
- Clustering (GMM, DBSCAN) [combined with distance metric]

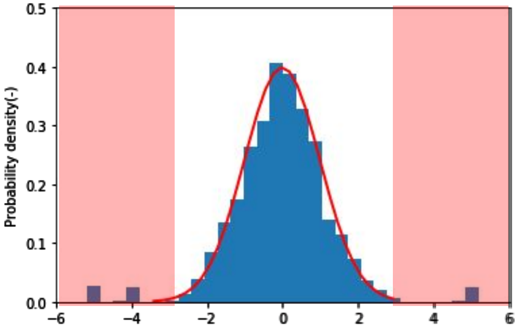
Special cases:

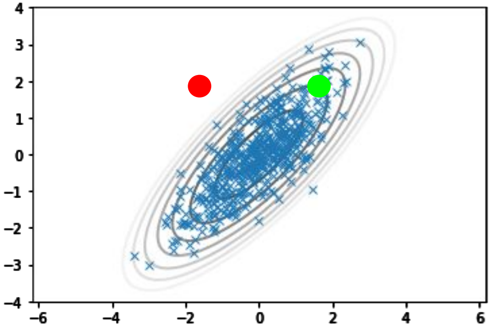
- One-class SVM
- Isolation Forest

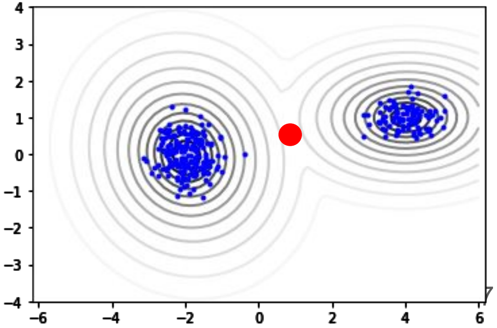
Note that these algorithms are based on different ideas of what an outlier means (far away from other points, easily splittable, not part of a large cluster, …)

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 of components is hyperparameter





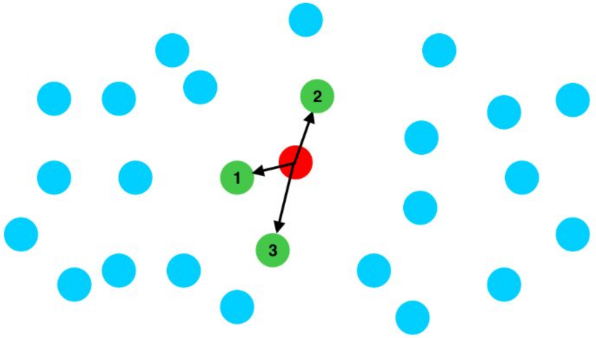


k-Nearest Neighbour (KNN)

- Distance-based outlier score
- Various distance measures may be used (L1, L2, cosine-distance, …)
- Various aggregation possibilities (distance to neighbour k, median of distances 1..k, ….)

Search algorithms:

- 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

LOF

- Similarity to KNN: neighbour search
- Compares “reachability-density” of a point to that of its nearest neighbours
- Homogeneous cluster → score = 1
- Outlier → score >> 1
- (Recommended K > 10)

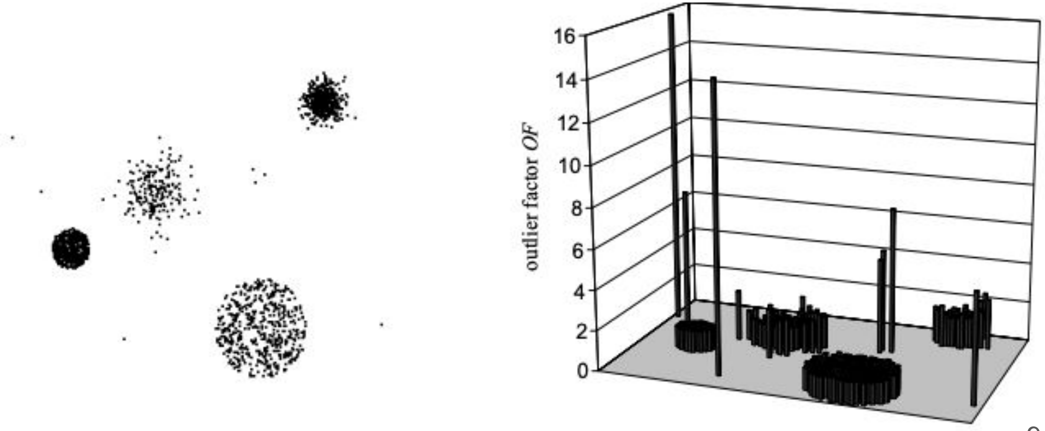
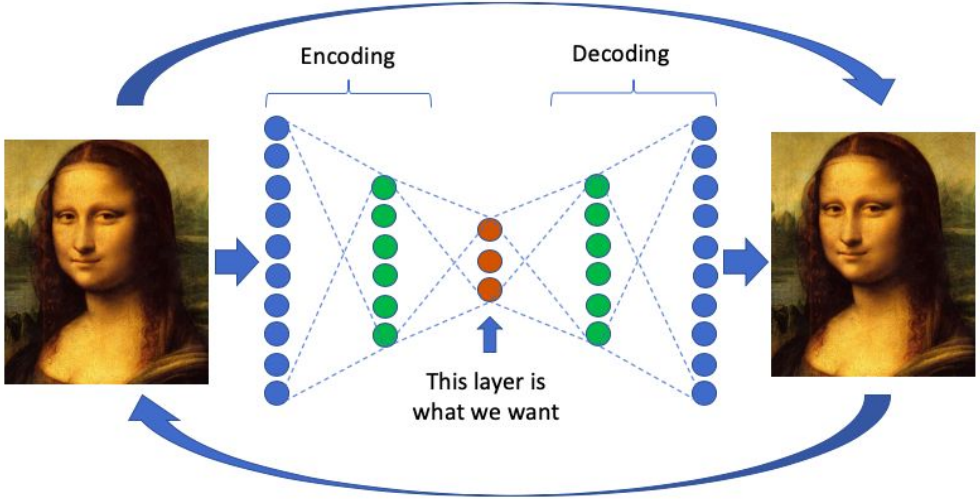


Figure 9: Outlier-factors for points in a sample dataset (*MinPts*=40)

Autoencoders / PCA

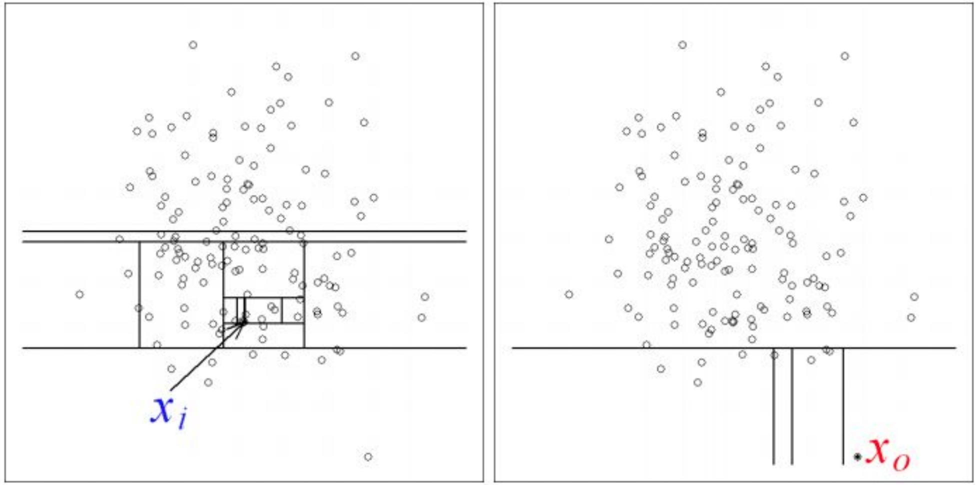
- Dimensionality reduction, and reconstruction
- Reconstruction error is globally minimized by PCA and Autoencoder
- Individual large reconstruction error may indicate outlier
- How to choose the layer sizes?

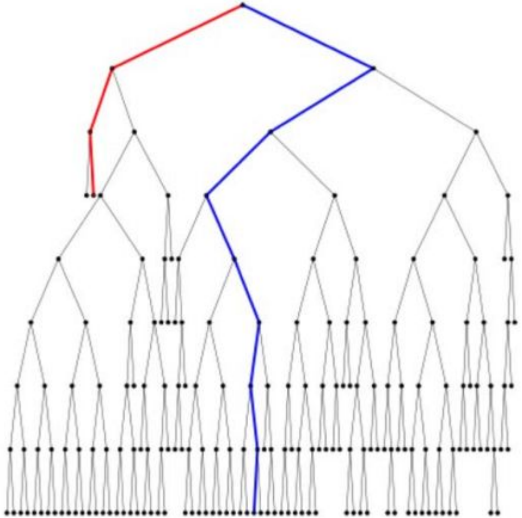


https://towardsdatascience.com/anomaly-detection-with-autoencoder-b4cdce4866a6

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



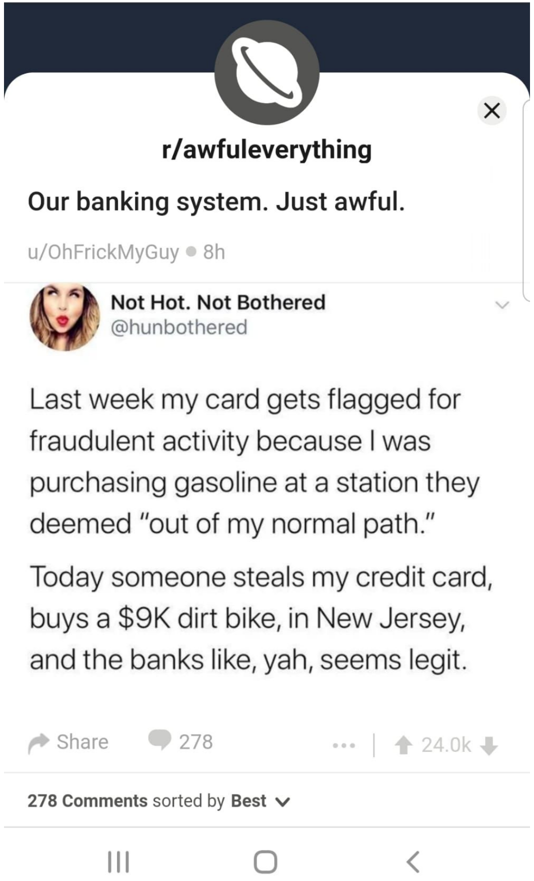


Liu, F. T., Ting, K. M., & Zhou, Z. H. (2008, December). Isolation forest. In *Data Mining, 2008. ICDM'08. Eighth IEEE International Conference on* (pp. 413–422). IEEE.

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

Metrics for outlier classification accuracy

- False Positive: Incorrectly predicting a positive (false alert)
- False Negative: Missing a positive
- Metrics for binary predictions versus for scores
- Metrics for binary predictions: accuracy, F1-score, Matthew’s correlation
  - All attempts to summarize a confusion matrix (2x2) into a single number
  - Accuracy is the right measure when the cost of a FP = FN
    (! Don’t believe everything you read in Medium articles. Accuracy is not necessarily bad for unbalanced data!)
- Metrics for outlier scores
  - Leave the choice of a decision threshold open
  - Aggregate (AUC-ROC, AUC-PR) or make (arbitrary) choices (N in precision@N, or recall in precision@recall)



r/awfuleverything

Our banking system. Just awful.

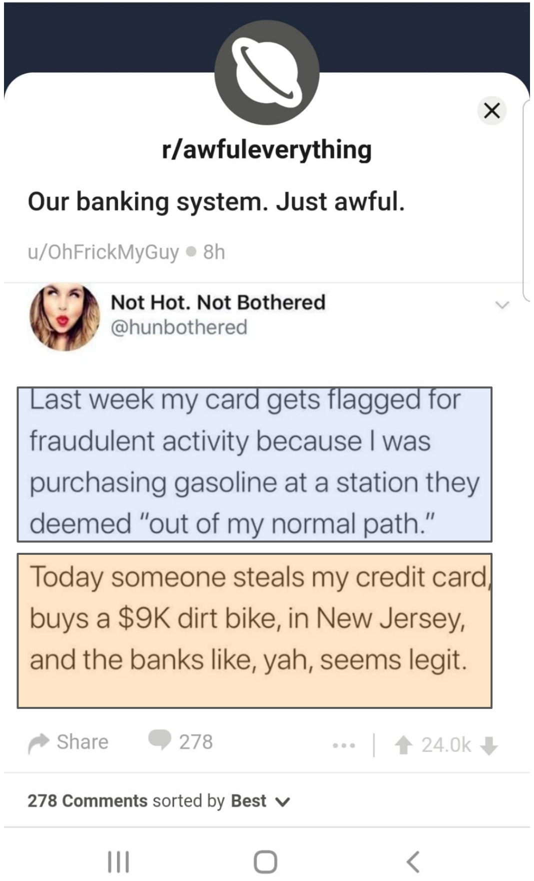
u/OhFrickMyGuy • 8h

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."

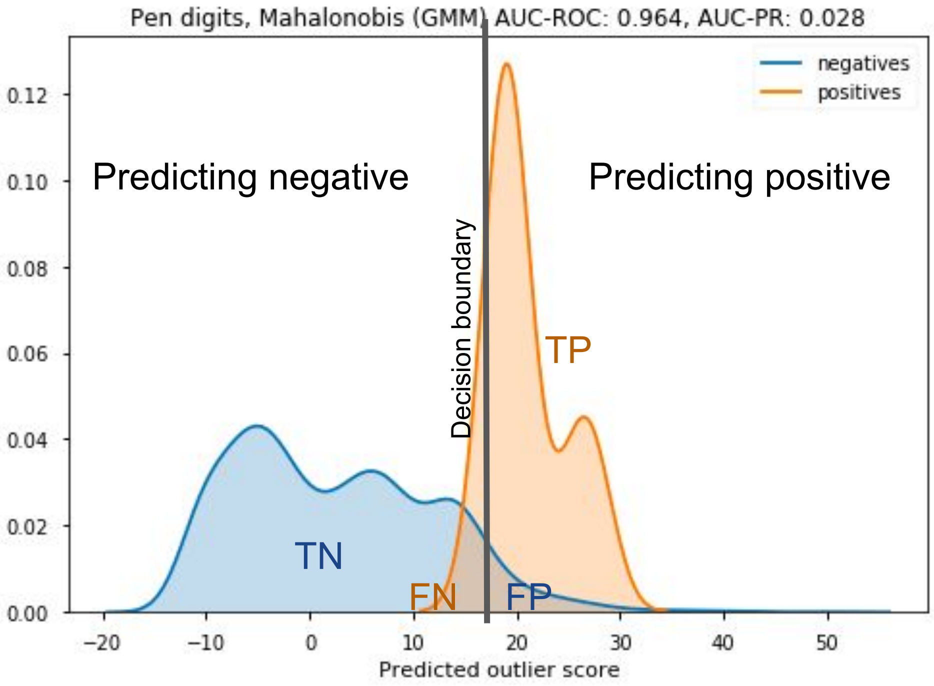
Today someone steals my credit card, buys a $9K dirt bike, in New Jersey, and the banks like, yah, seems legit.

278 Comments sorted by Best

False Positive: annoying

False Negative:
annoying + expensive

Visualizing scoring: conditional score curves



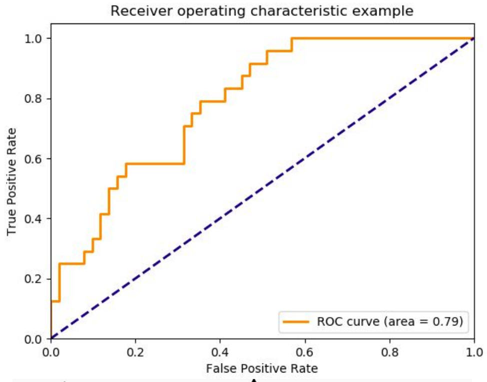
Recall (TPR): what fraction of the true positives were correctly identified?
TP / P

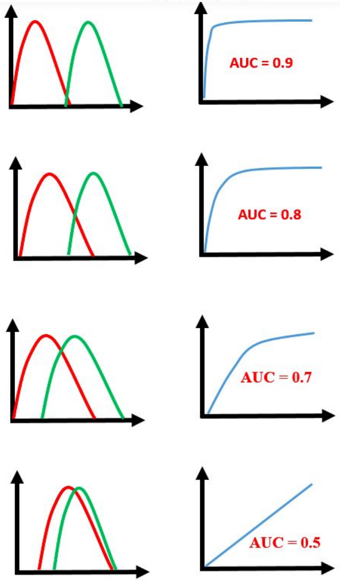
False Positive Rate (FPR): what fraction of actual negatives was falsely predicted positive?
FP / N

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

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



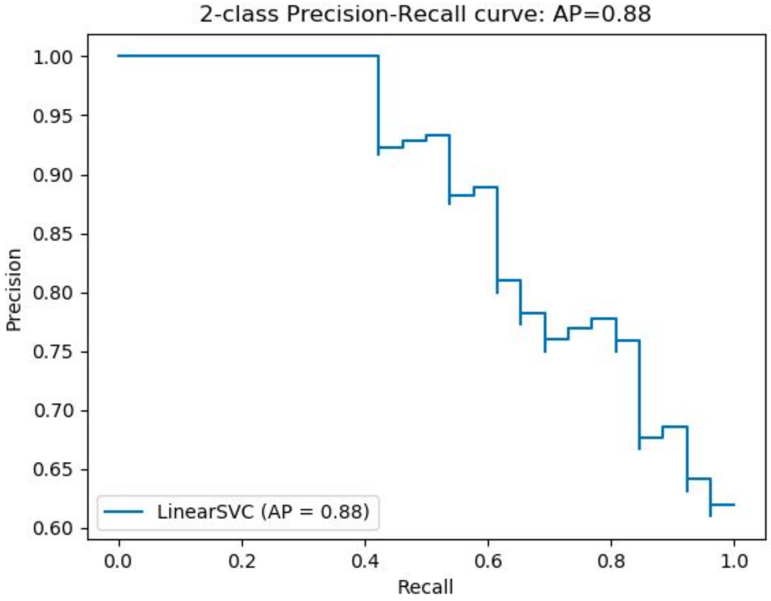


https://scikit-learn.org/stable/auto_examples/plot_roc_curve_visualization_api.html#sphx-glr-auto-examples-plot-roc-curve-visualization-api-py

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

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

Outlier detection in Python

Scikit-Learn has a lot (KNN, GMM, LOF, iForest, one-class SVM, Covariance)

PyOD provides a nice wrapper around those, with a consistent API

- .fit()
- .decision_scores_

And makes the construction of e.g. Autoencoders very easy (uses keras + tensorflow)

https://github.com/yzhao062/Pyod

Exercise

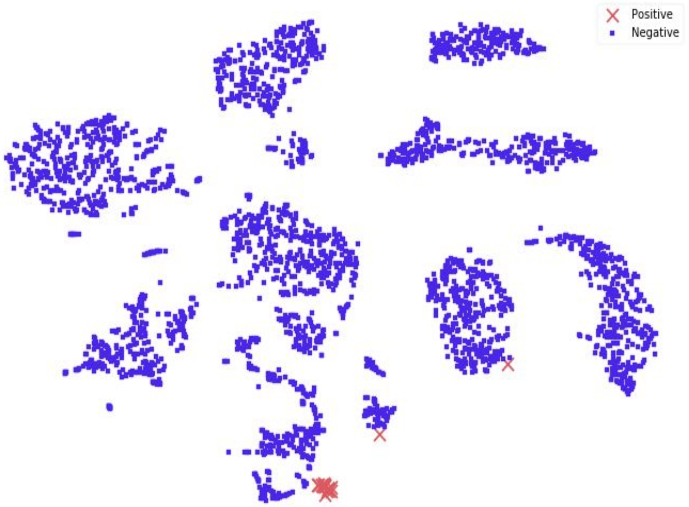
We use the (small and simple) pendigits dataset, to get familiar with pyod and several of its implemented algorithms (that are often scikit-learn).

The function `plot_outlier_scores()` shows the conditional score curves, the AUC-ROC and the AUC-AP

The function `plot_top_N()` shows the true labels of the top-N predictions, and the precision@N

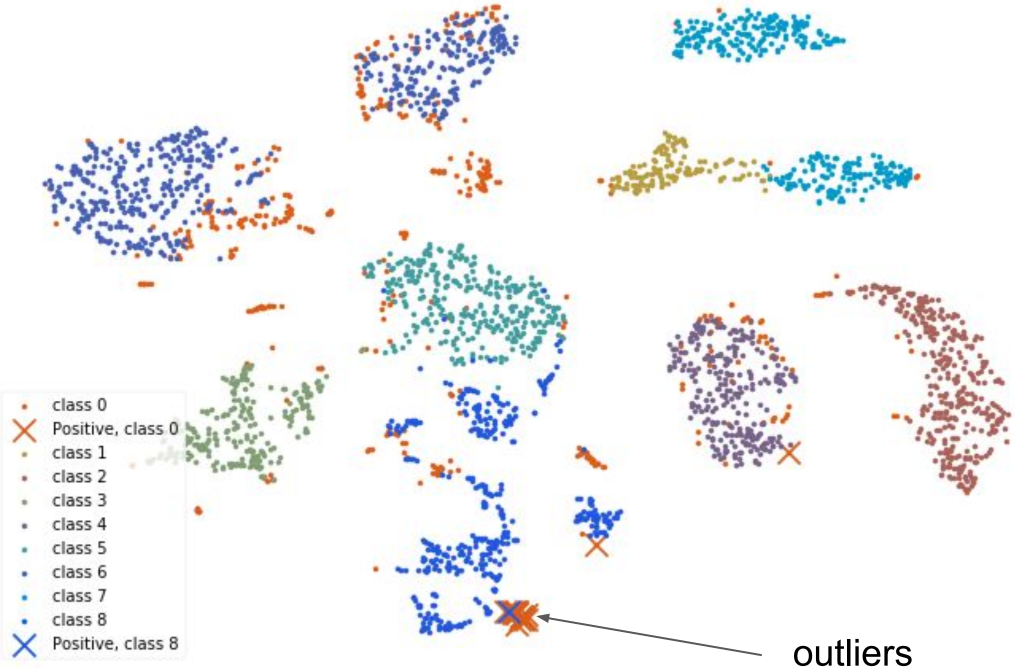
Exercise - data structure

- 10k points, 0.2% is positive
- t-SNE: positives form their own cluster (note: they are all 4’s → underrepresented class)
- 90% of the neighbours of positives is indeed a positive
- We see ~ 10 clusters, which corresponds to our data understanding
- GMM and t-SNE agree on the data structure (see next slide)



Exercise - data structure

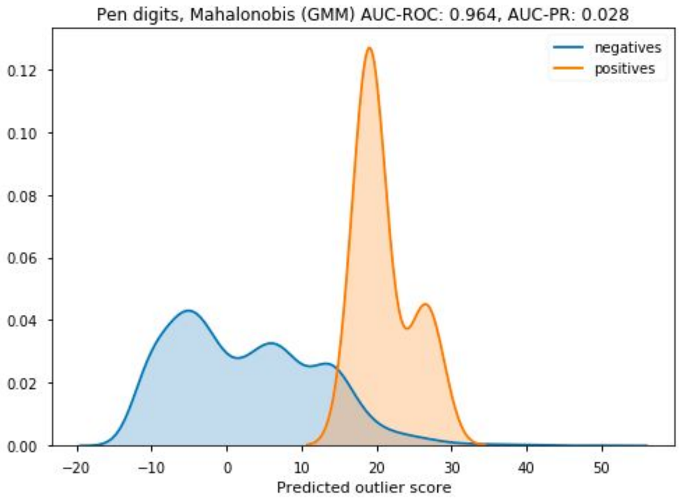
GMM results: colours from GMM labels, coordinates from t-SNE



outliers

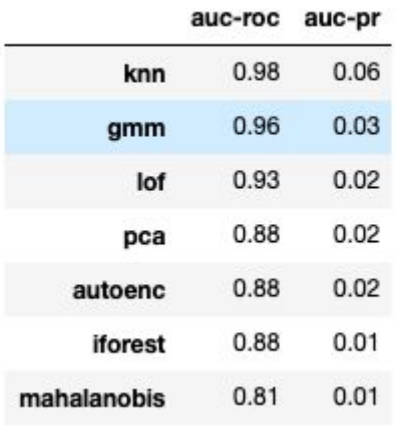
Exercise - algorithm performance

- 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%)



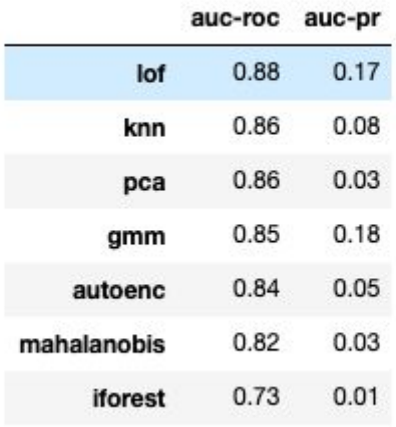
Exercise - algorithm comparison

Original Outliers



	auc-roc	auc-pr
knn	0.98	0.06
gmm	0.96	0.03
lof	0.93	0.02
pca	0.88	0.02
autoenc	0.88	0.02
iforest	0.88	0.01
mahalanobis	0.81	0.01

Synthetic Outliers



	auc-roc	auc-pr
lof	0.88	0.17
knn	0.86	0.08
pca	0.86	0.03
gmm	0.85	0.18
autoenc	0.84	0.05
mahalanobis	0.82	0.03
iforest	0.73	0.01

Exercise - “Tuning complexity”

- Mahalanobis: parameter-free
- Isolation Forest: practically parameter-free
- KNN, GMM, LOF: single tuning parameter. May be “guesstimated”
- PCA and Autoencoder: difficult (heuristic: middle layer N/2 +1)

Workshop challenge

KDD Cup data: dataset with 38 numerical and 3 categorical variables, 48k rows

What to do with categorical data?

- Ignore it
- **One-hot/dummy encoding**
- Binary encoding
- Frequency encoding
- Embeddings (this is implicitly achieved in an autoencoder)

Hint: use pd.get_dummies, and MinMaxScaler

Workshop challenge

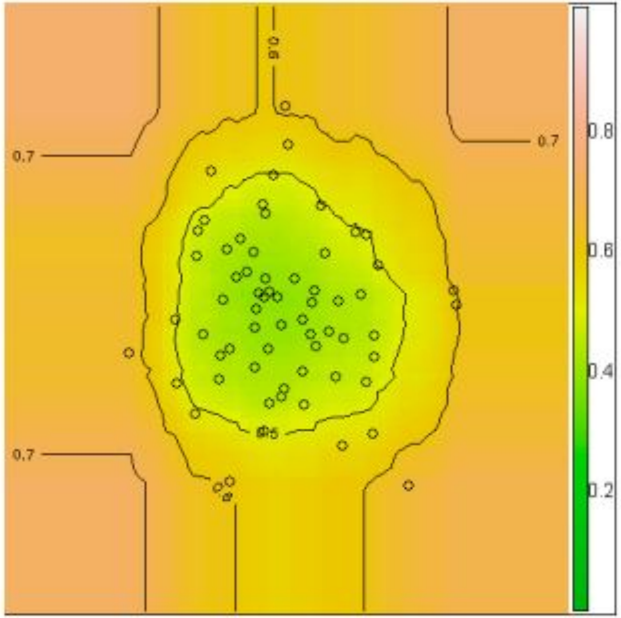
There is an API on AWS (see link in notebook) to submit predictions, and a “front end” in Python that helps you submit (LabelSubmitter object)

- The `LabelSubmitter` class has two relevant functions:
  - .post_predictions(idx=[np.array], endpoint=’kdd’) → post predictions (positives)
  - .get_labels(endpoint=’kdd’) → get the true labels to all points you have submitted
- Once a prediction is made, it can’t be changed (submitting again is okay, but won’t change anything)
- A true positive gets you 500 points, false positive costs you 25 points
- Hints:
  - don’t submit too much, but also don’t be afraid to submit something
  - The positive class is < 1% of the data (don’t guess too much randomly)

Links

Extra: Isolation Forest

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


0.6
0.7
0.7
0.7
0.5
0.8
0.6
0.4
0.2