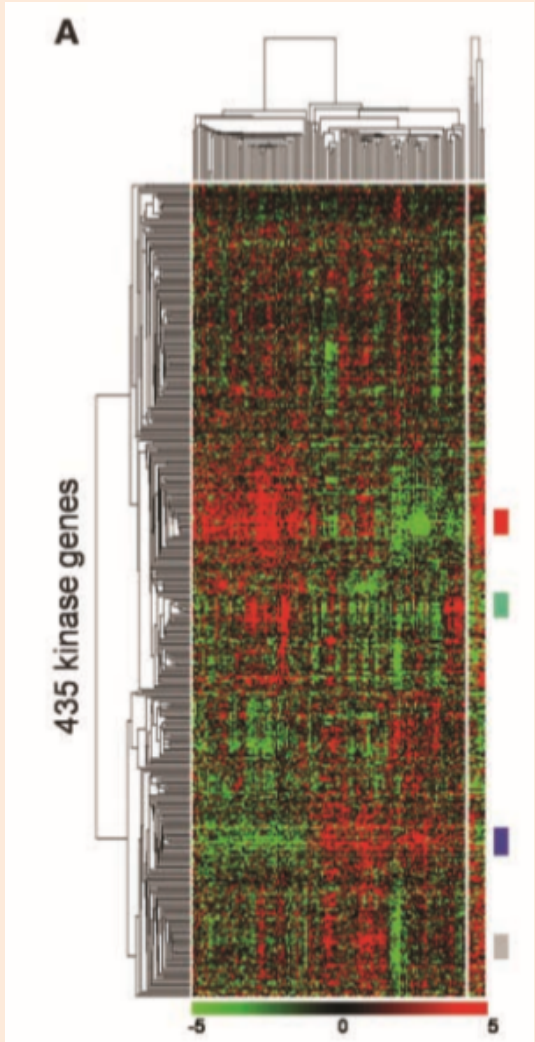
CPSC 340:
Machine Learning and Data Mining

Outlier Detection

Bonus slides

Application: Medical data

- Hierarchical clustering is very common in medical data analysis.
  - Biclustering different samples of breast cancer:



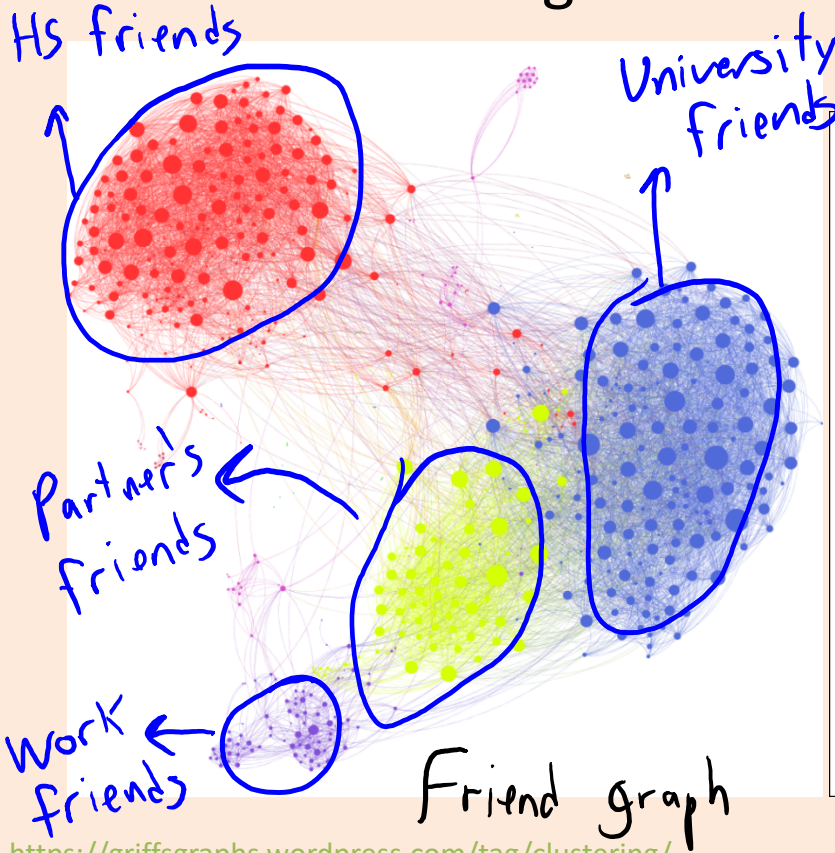
http://members.cbio.mines-paristech.fr/~jvert/svn/bibli/local/Finetti2008Sixteen-kinase.pdf

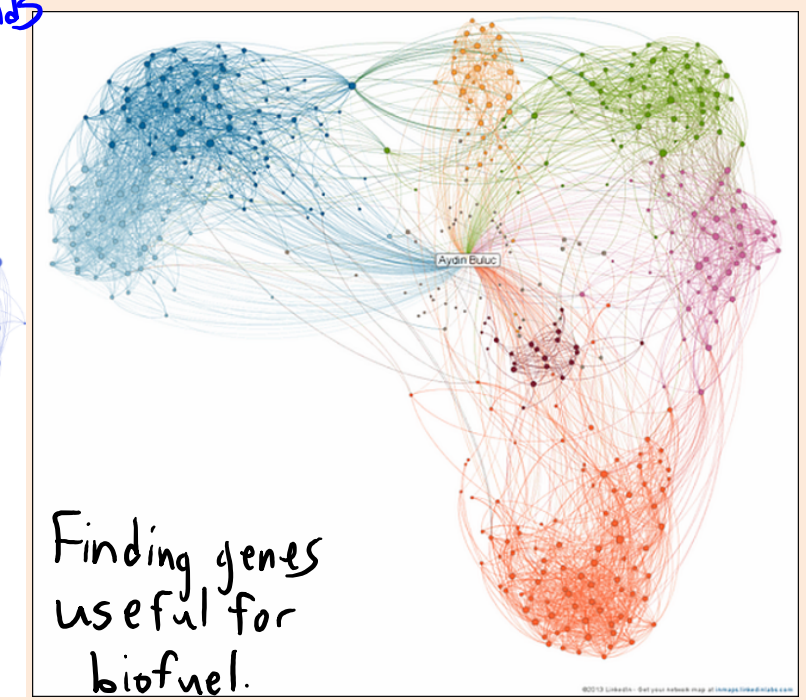
Other Clustering Methods

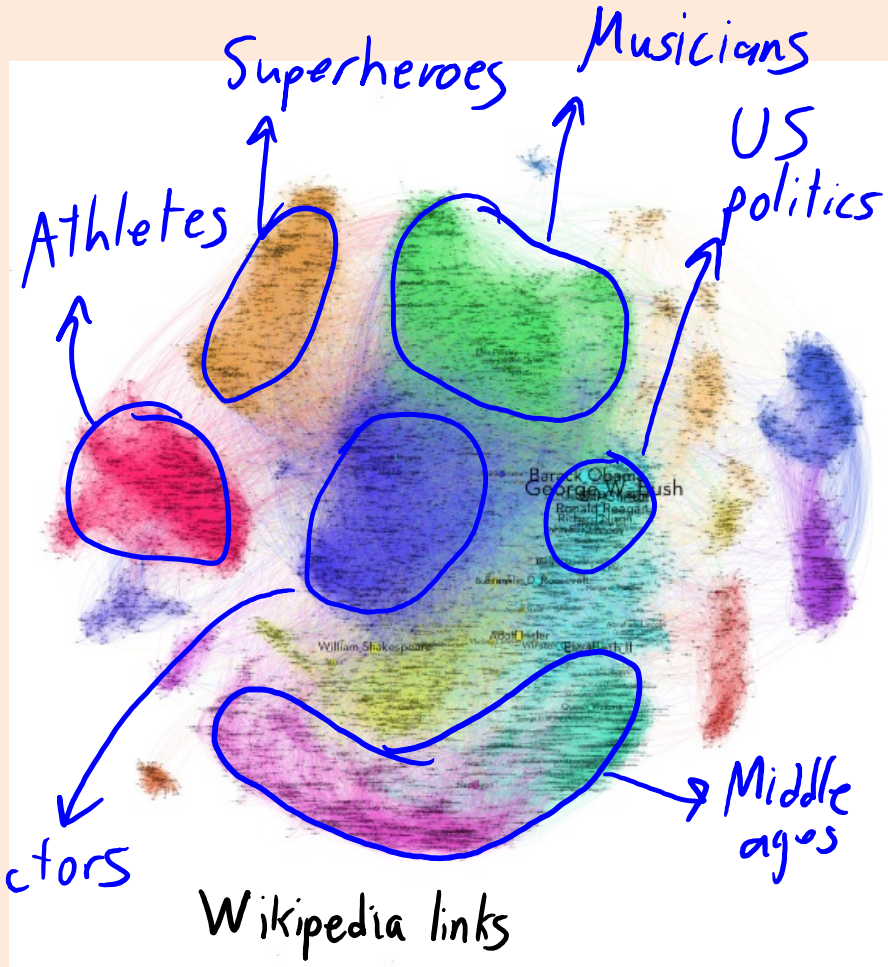
- Mixture models:
  - Probabilistic clustering.
- Mean-shift clustering:
  - Finds local “modes” in density of points.
  - Alternative approach to vector quantization.
- Bayesian clustering:
  - A variant on ensemble methods.
  - Averages over models/clusterings,
    weighted by “prior” belief in the model/clustering.

Graph-Based Clustering

- Spectral clustering and graph-based clustering:
  - Clustering of data described by graphs.



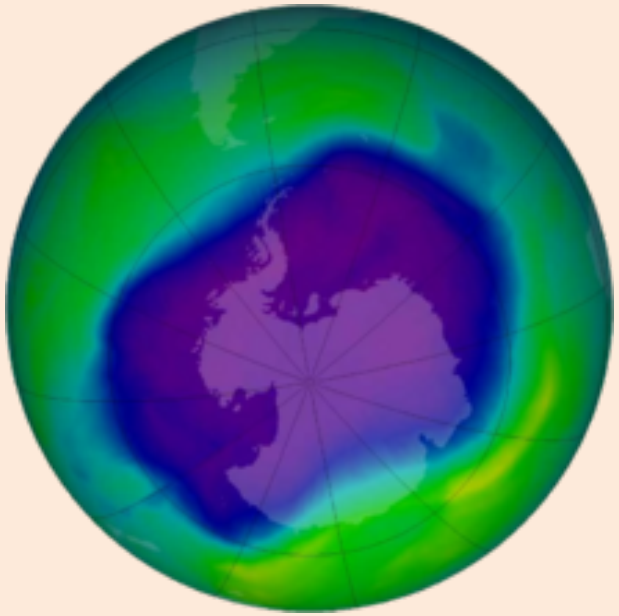




https://griffsgraphs.wordpress.com/tag/clustering/
http://ascr-discovery.science.doe.gov/2013/09/sifting-genomes/
https://www.hackdiary.com/2012/04/05/extracting-a-social-graph-from-wikipedia-people-pages/

Motivating Example: Finding Holes in Ozone Layer

- The huge Antarctic ozone hole was “discovered” in 1985.



- It had been in satellite data since 1976:
  - But it was flagged and filtered out by a quality-control algorithm.

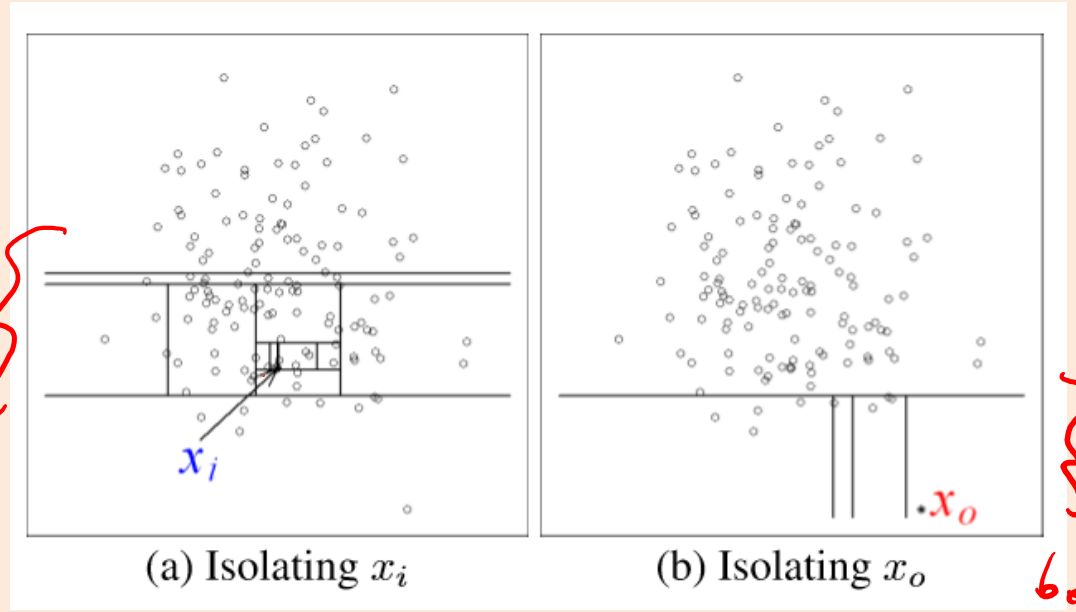
https://en.wikipedia.org/wiki/Ozone_depletion

Isolation Forests

- Recent method based on random trees is isolation forests.
  - Grow a tree where each stump uses a random feature and random split.
  - Stop when each example is "isolated" (each leaf has one example).
  - The "isolation score" is the depth before example gets isolated.
    - Outliers should be isolated quickly, inliers should need lots of rules to isolate.

Depth 12: - needed 12 rules to isolate so may be inlier.

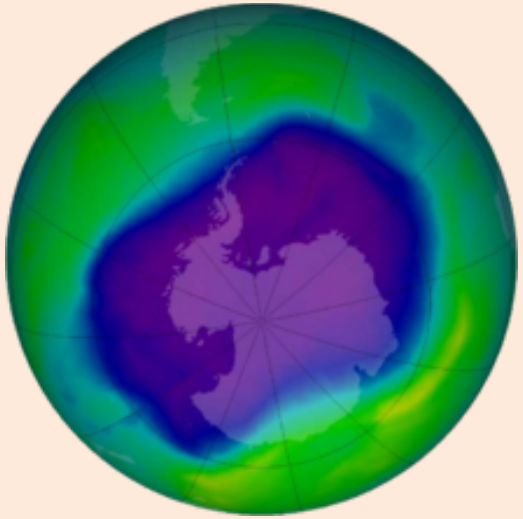
  - Repeat for different random trees, take average score.



https://cs.nju.edu.cn/zhouzh/zhouzh.files/publication/icdm08b.pdf

Problem with Unsupervised Outlier Detection

- Why wasn't the hole in the ozone layer discovered for 9 years?

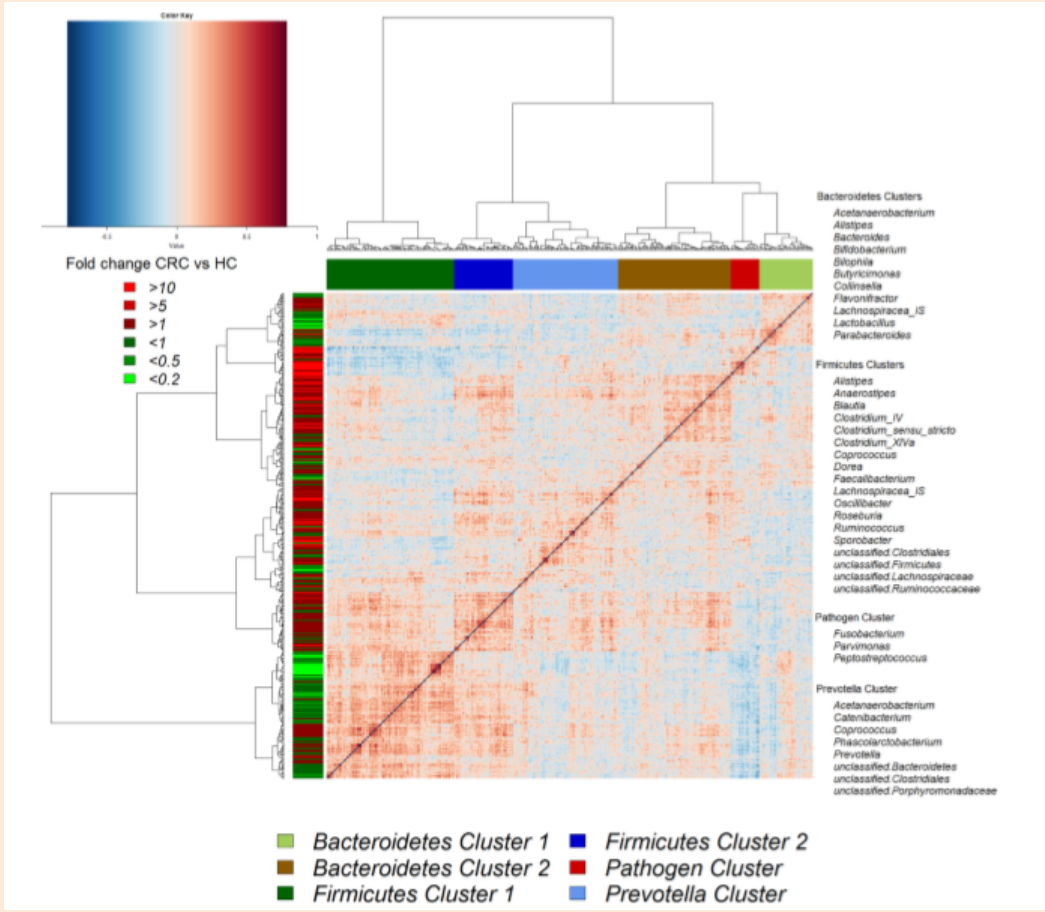


- Can be hard to decide when to report an outler:
  - If you report too many non-outliers, users will turn you off.
  - Most antivirus programs do not use ML methods (see "base-rate fallacy")

https://en.wikipedia.org/wiki/Ozone_depletion

Application: Medical data

- Hierarchical clustering is very common in medical data analysis.
  - Clustering different samples of colorectoral cancer:
  - This plot is different, it's not a biclustering:
    - The matrix is 'n' by 'n'.
    - Each matrix element gives correlation.
    - Clusters should look like "blocks" on diagonal.
    - Order of examples is reversed in columns.
      - This is why diagonal goes from bottom-to-top.
      - Please don't do this reversal, it's confusing to me.



https://gut.bmj.com/content/gutjnl/66/4/633.full.pdf

Issues with using z-scores for grades

I definitely sympathize with issues regarding baseline grades in different classes. The ideal solution is to encourage grades to have a standardized meaning across courses, and for courses to have a standardized difficulty, but obviously this is incredibly hard (and probably impossible).

The use of z-scores seems to be a nice solution, but I wanted to point out some potential issues:

1. Z-scores are quite sensitive to outliers. Basically, the mean will be pulled in the direction of outliers, and the variance will be made much larger by outliers. See Slide 8 here:
https://www.cs.ubc.ca/~schmidtm/Courses/540-W20/L6.pdf

The major way this manifests is if you have a relatively-small class, and one person just catastrophically fails the course. This has weird effects on the z-score compared to if that person was not in the class: since the average moves lower, people who are slightly below average will actually appear slightly above average. This isn't a big deal, but the more serious issue is that since the variance is made larger the people who are a bit below average will appear very-far below average. (And students well above average get pushed way above average.)

The effect is much smaller in big classes, unless you have a cluster of catastrophic fails and in that case the effect is the same.

There are easy solution to this issue by using statistics based on more-robust measures that allow outliers (for examples, see Slide 9 in that lecture).

2. Z-scores assume the distribution is unimodal. See Slide 10 here:
https://www.cs.ubc.ca/~schmidtm/Courses/540-W20/L6.pdf

If you have a group of "good" students and a group of "bad" students, it may reward the good group and punish the bad group more than their grade difference would justify. I think this is a less serious issue, and it's also harder to fix (you would probably need to use historic grade distribution data). In 340, I would expect the grade distribution to roughly look like this.

3. It doesn't address "skew" in the distribution. This could be the case if you have a lot of people at the very top and then the grades drop off slowly from there (another effect I've noticed in 340 grades). Similar to 2, I view this as a less-serious issue than 1 since the shifts probably aren't huge.

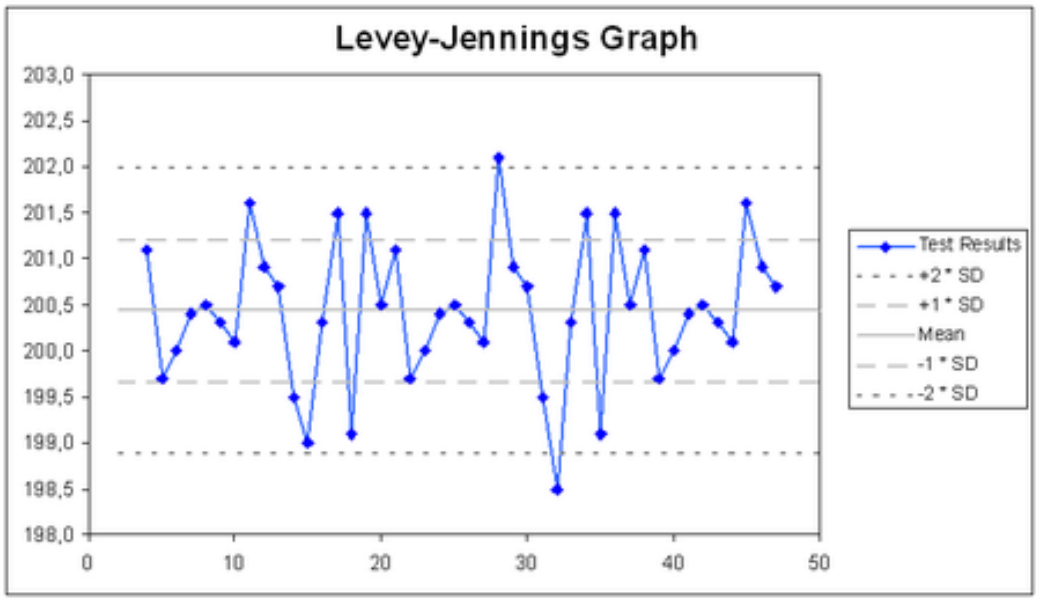
4. If you compare z-scores *across* classes, there is a confounding factor that the students may not come from the same distribution. E.g., one class may attract more strong students and one class may attract more weak students. In a simple setting where only top students take one class and only weak students take another class, the weaker "top" students will be hurt and the stronger "weak" students will be helped.

A simple approach that would address 1-3 is using quantiles. For example, just saying "student A ranked in the top 38% of grades" is simple and avoids some of the issues above. It's not perfect since it doesn't give the real spread (problematic if many students are really close, since it will push them apart). It also doesn't address issue 4, but I would be more comfortable making decisions with this than z-scores. Indeed, my criterion for whether I will write reference letters for students in class is based on ranking rather than absolute score. It's even-more informative to give the class size, like "student A ranked 14 out of 76", but that might be more-difficult to use in automated ways.

For addressing issue 4, you would really need data across classes and I would have to think about whether there is a simple/fair solution.

“Quality Control”: Outlier Detection in Time-Series

- A field primarily focusing on outlier detection is quality control.
- One of the main tools is plotting z-score thresholds over time:



- Usually don’t do tests like “$|z_i| > 3$”, since this happens normally.
- Instead, identify problems with tests like “$|z_i| > 2$ twice in a row”.

https://en.wikipedia.org/wiki/Laboratory_quality_control

Outlierness (Symbol Definition)

- Let $N_k(x_i)$ be the k-nearest neighbours of x_i.
- Let $D_k(x_i)$ be the average distance to k-nearest neighbours:

$$D_k(x_i) = \frac{1}{k} \sum_{j \in N_k(x_i)} \|x_i - x_j\|$$

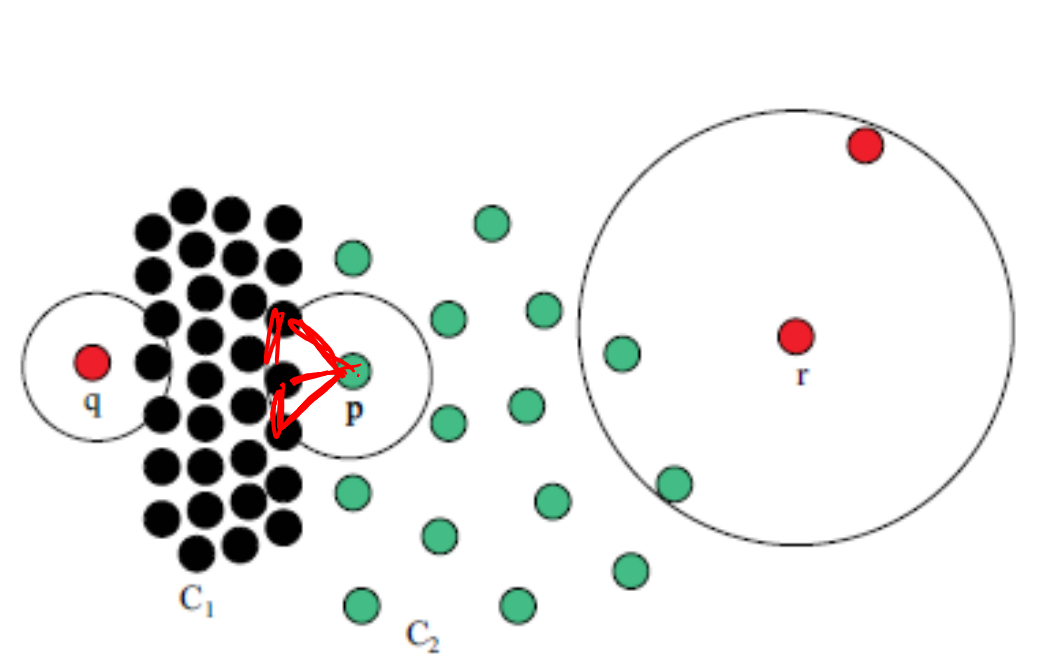
- Outlierness is ratio of $D_k(x_i)$ to average $D_k(x_j)$ for its neighbours 'j':

$$O_k(x_i) = \frac{D_k(x_i)}{\frac{1}{k} \sum_{j \in N_k(x_i)} D_k(x_j)}$$

- If outlierness > 1, x_i is further away from neighbours than expected.

Outlierness with Close Clusters

- If clusters are close, outlierness gives unintuitive results:

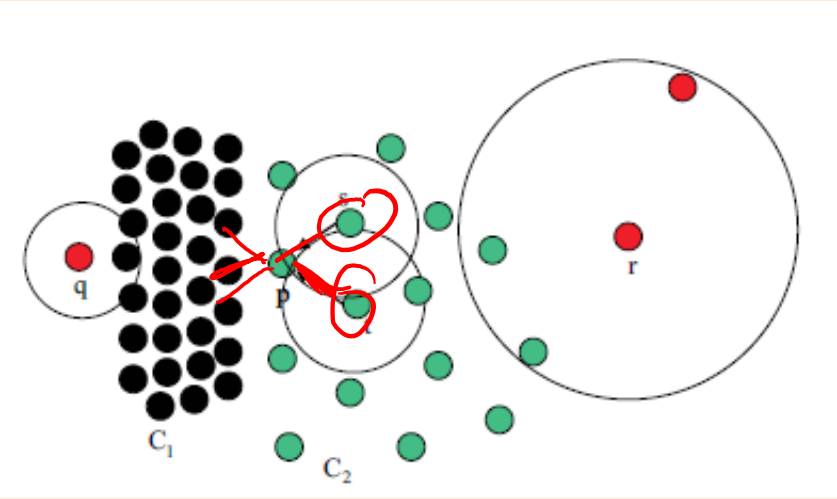


- In this example, ‘p’ has higher outlierness than ‘q’ and ‘r’:
  - The green points are not part of the KNN list of ‘p’ for small ‘k’.

http://www.comp.nus.edu.sg/~atung/publication/pakdd06_outlier.pdf

Outlierness with Close Clusters

- 'Influenced outlierness' (INFLO) ratio:
  - Include in denominator the 'reverse' k-nearest neighbours:
    - Points that have 'p' in KNN list.
  - Adds 's' and 't' from bigger cluster that includes 'p':



- But still has problems:
  - Dealing with hierarchical clusters.
  - Yields many false positives if you have "global" outliers.
  - Goldstein and Uchida [2016] recommend just using KNN.

http://www.comp.nus.edu.sg/~atung/publication/pakdd06_outlier.pdf

Training/Validation/Testing (Supervised)

- A typical supervised learning setup:
  - Train parameters on dataset D_1.
  - Validate hyper-parameters on dataset D_2.
  - Test error evaluated on dataset D_3.

- What should we choose for D_1, D_2, and D_3?

- Usual answer: should all be IID samples from data distribution D_s.

Training/Validation/Testing (Outlier Detection)

- A typical outlier detection setup:
  - Train parameters on dataset D_1 (there may be no "training" to do).
    - For example, find z-scores.
  - Validate hyper-parameters on dataset D_2 (for outlier detection).
    - For example, see which z-score threshold separates D_1 and D_2.
  - Test error evaluated on dataset D_3 (for outlier detection).
    - For example, check whether z-score recognizes D_3 as outliers.

- D_1 will still be samples from D_s (data distribution).
- D_2 could use IID samples from another distribution D_m.
  - D_m represents the "none" or "outlier" class.
  - Tune parameters so that D_m samples are outliers and D_s samples aren't.
    - Could just fit a binary classifier here.

Training/Validation/Testing (Outlier Detection)

- A typical outlier detection setup:
  - Train parameters on dataset D_1 (there may be no "training" to do).
    - For example, find z-scores.
  - Validate hyper-parameters on dataset D_2 (for outlier detection).
    - For example, see which z-score threshold separates D_1 and D_2.
  - Test error evaluated on dataset D_3 (for outlier detection).
    - For example, check whether z-score recognizes D_3 as outliers.

- D_1 will still be samples from D_s (data distribution).
- D_2 could use IID samples from another distribution D_m.
- D_3 could use IID samples from D_m.
  - How well do you do at recognizing "data" samples from "none" samples?

Training/Validation/Testing (Outlier Detection)

- Seems like a reasonable setup:
  - D_1 will still be samples from D_s (data distribution).
  - D_2 could use IID samples from another distribution D_m.
  - D_3 could use IID samples from D_m.

- What can go wrong?

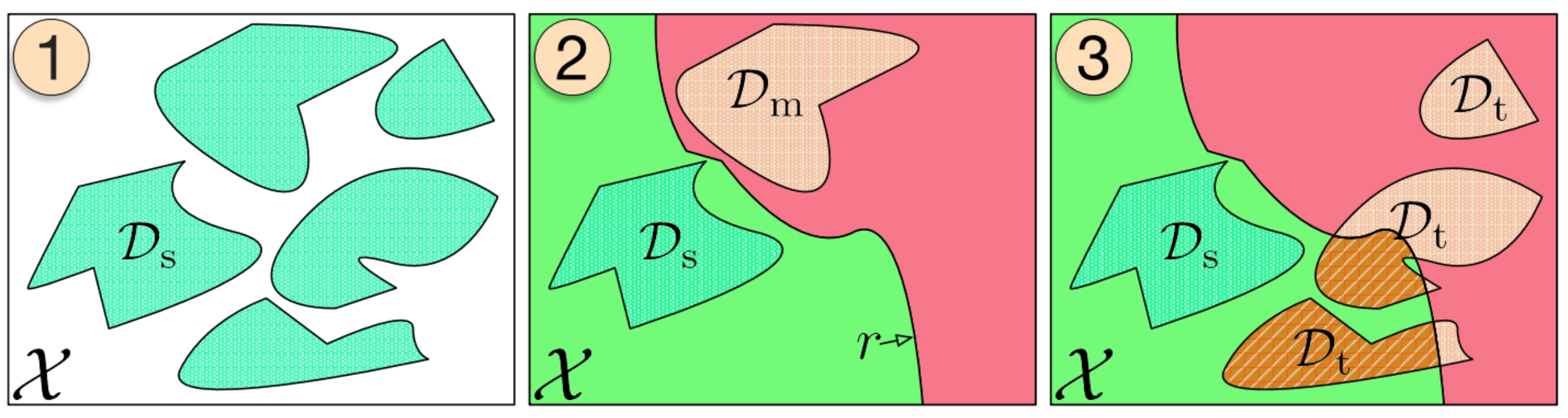
- You needed to pick a distribution D_m to represent "none".
  - But in the wild, your outliers might follow another "none" distribution.
  - This procedure can overfit to your D_m.
    - You can overestimate your ability to detect outliers.

OD-Test: a better way to evaluate outlier detections

- A reasonable setup:
  - D_1 will still be samples from D_s (data distribution).
  - D_2 could use IID samples from another distribution D_m.
  - ~~D_3 could use IID samples from D_m.~~
  - D_3 could use IID samples from yet-another distribution D_t.

- "How do you perform at detecting different types of outliers?"
  - Seems like a harder problem, but arguably closer to reality.

OD-Test: a better way to evaluate outlier detections



- “How do you perform at detecting different types of outliers?”

https://arxiv.org/pdf/1809.04729.pdf