CSCI E-82
Advanced Machine Learning, Data Mining & Artificial Intelligence
Lecture 10

Outlier Analysis

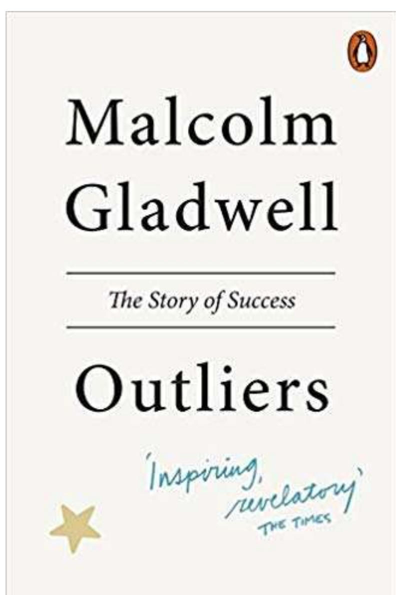
Peter V. Henstock
Fall 2018

Rest of the semester…

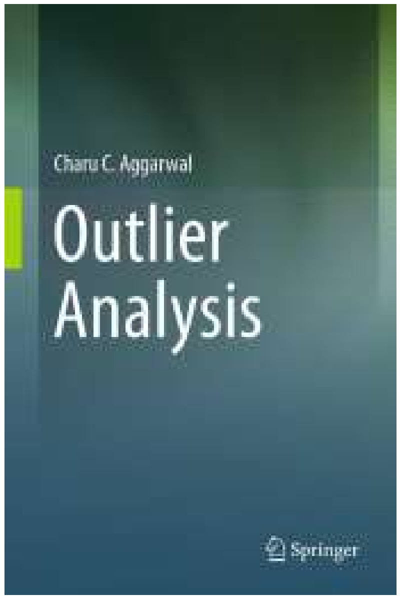
- Homework 5 CNN due next week
- Exam the next weekend
- Project proposal 2 lines
- Paper ~~presentations~~ → (1-2 paragraphs or YouTube 5 min)
  - Paper reviews
- HW 6 on Shakespeare (reduced)
- Final project

Outliers

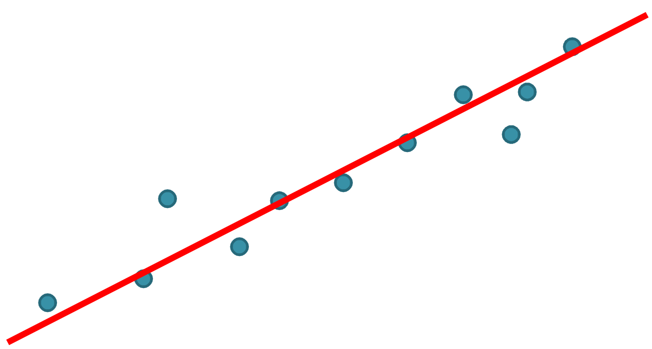
Popular Literature



The Outlier Analysis Bible



Regression Outlier



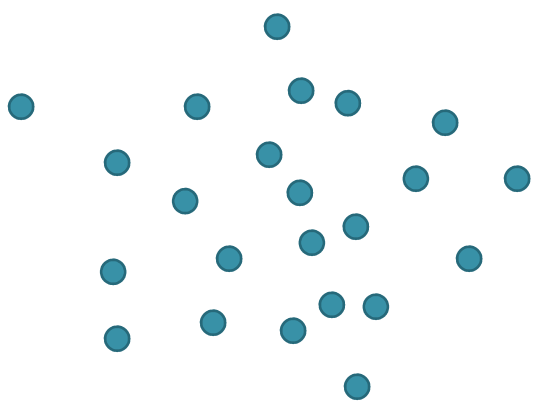
- $Y = WX + E$
- Assumption of residuals E
- What is an outlier?

Outlier

"An observation which deviates so much from the other observations as to arouse suspicions that it was generated by a different mechanism"

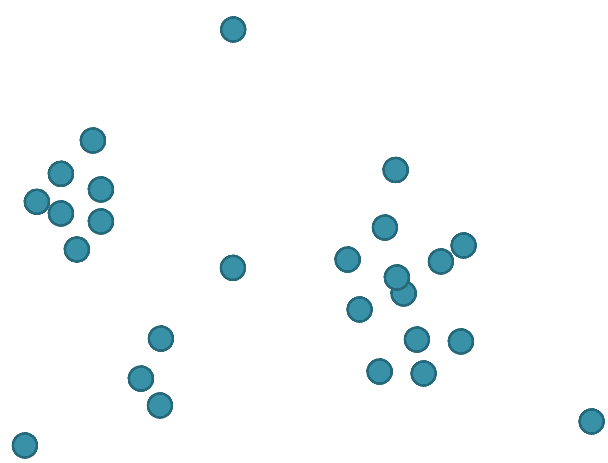
--Hawkins

How would you determine outliers?

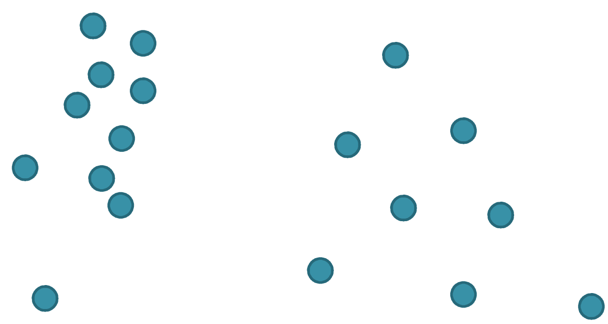


How would you determine outliers?

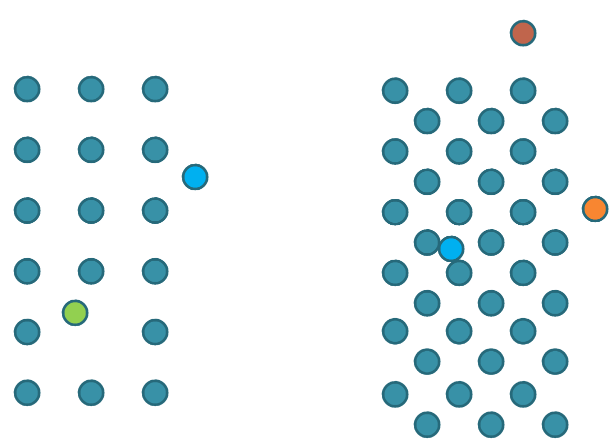
How would you determine outliers?



How would you determine outliers?



How would you determine outliers?



General Categories

- Outlier score
  - Use algorithm to score to each observation
  - Threshold scores as outliers
- Binary yes/no
  - May generate label from outlier score
- Noise issue
  - Adds variance to measurements
  - Requires separation of noise (‘weak’) & ‘strong’ outliers

Types of Anomalies

- Point
  - Single instance or small group
- Contextual
  - Outlier is not typically an extreme value
  - Outlier relative to a standard behavior
  - Time-domain, spatial domain, etc.
  - Snow in July. Snow in Florida
- Collective
  - Irregular pattern such as missing heartbeat
  - Strong pattern

Types of Anomaly Detection

- Supervised: labels for normal & outliers
- Semi-supervised: Labels for normal only
- Unsupervised: no labels
  - Assume that outliers are rare

Challenges

- High dimensional data
- Sparse data
- Heterogeneous data
- Categorical or ordered data
- Noise
- Contextual outliers (networks)

Statistical Approaches

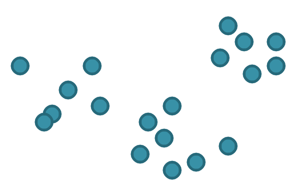
Central Limit Theorem (Strong)

- Sum of large number N of iid random variables with mean μ and stdev σ
- Sum converges to $N(\mu N, \sigma / \mathrm{sqrt}(N))$
- So if have a sum variable, we can compute the probabilities of outliers based on a normal distribution
- Applicable to customer store visits
- Applicable to sports statistics

1D outlier

- $Z_i = |X_i – \text{mean}| / \text{stdev}$
- If X ~ N(mean, stdev) then Z_i ~ Zipf
- Outliers assumed $Z_i >= 3$ perhaps
- What if don't have enough points to come up with a good estimate of mean and standard deviation?
  - Use Student t-distribution instead

Dependence on model

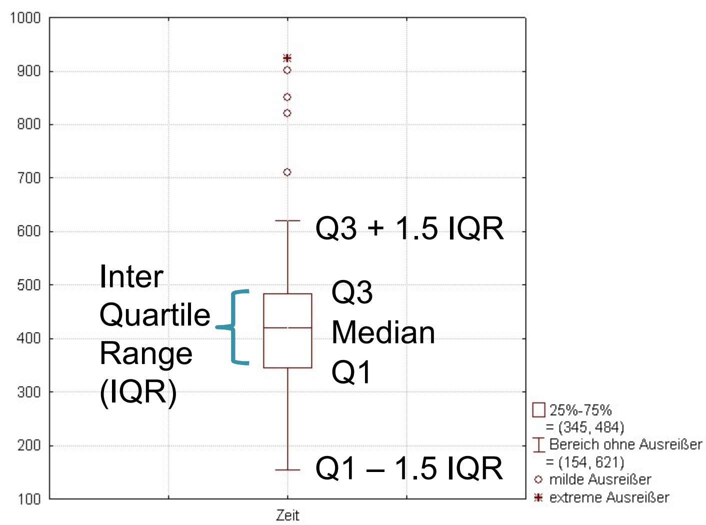


- Could model as Gaussian in 1D or 2D
- Could model as 3 clusters
- Depends on understanding of the natural underlying patterns inherent to the domain

Thresholds for Outliers

- Markov Inequality (Weak)
  - Let X be random variable s.t. X>=0
  - For α satisfying $E[X] < \alpha$ then
    - $P(X > \alpha) <= E[X] / \alpha$
- Chebychev Inequality (Weak)
  - Let X be random variable (no restrictions)
  - $P(|X - E[X]| > \alpha) <= Var[X] / \alpha^2$

Box-Whisker View



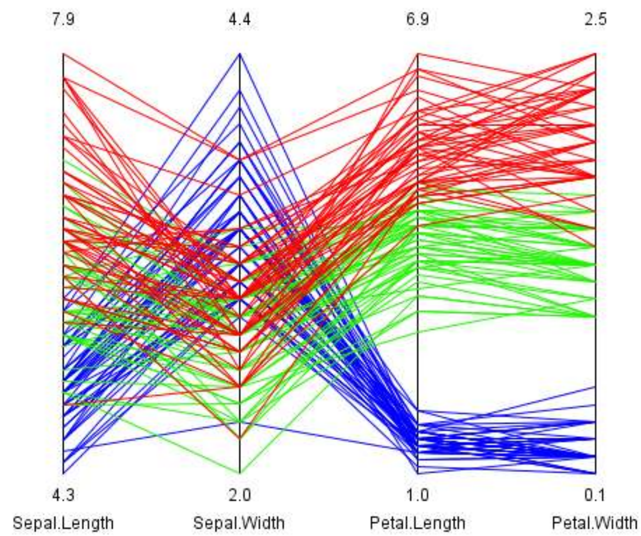
Box-Whisker

- How does this compare to N() dist?
- Median ~ Mean if normal distribution
- Q1 ~ 0.667 stdev
- So IQR = 2*0.667 = 1.349
- Threshold = 0.667 + 1.349 = 2.7
- Corresponds to probability of 0.9965
- Note: there are other box-whisker types or conventions

Going from 1-D to N-D

- What if you have an k-D set of data?
- For a single dimension, apply Z-value
- But, want to compute outlier score across multiple dimensions or z's
- Outlier score could be ΣZ_i^2
- Distribution of ΣZ_i^2 ~ chi-sq(d)
  - d = degrees of freedom or d = "k"

Parallel Coordinates Visualization



https://www.wanderinformatiker.at/unipages/general/iris_en.html

SmartSifter

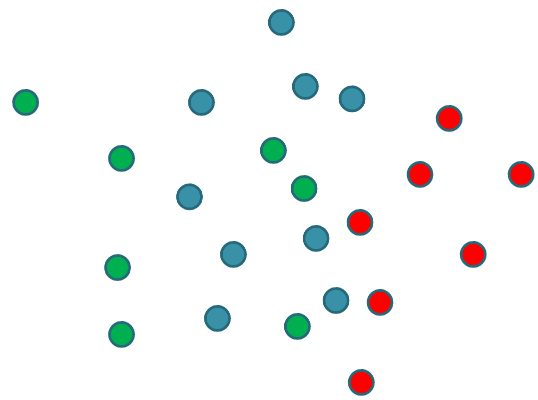
- Yamanishi 2000
- Online unsupervised outlier detection using finite mixtures with discounting learning algorithms
- Combination of data types:
  - Handles categorical with histogram
  - Handles continuous with mixture model
- Approach
  - Model full data set
  - For each point
    - Leave-one-out statistical model
    - Compute |p(full)-p(l-o-o)| as outlier value

Limitations of Probability Modeling

- Assumptions to distributions
- Number of parameters
- Fitting & over-fitting (cluster or EM)

Classification (Supervised) Approaches

Active Learning concept



Supervised Methods

- Whenever you can, use a supervised method for outlier detection
  - More accurate
  - Gives insight on the class of outlier in some cases (like intrusion detection)
- Challenges:
  - Class imbalance
  - Contaminated labels
    - Undetected spam may exist in a data set
  - Partial training available

Adaptive Re-sampling

- Sample training data to favor the outliers
  - Either with or without replacement or both
- Optimize weighted cost
  - Σ_i classError$_i$ * cost$_i$
- Adaptive part: Sample proportional to size
  - Might take 2% of normal + all 1% of outlier
  - Variation: "Sequential Ensemble" correct predictions excluded in later iterations

Under/Over-Sampling

- Undersampling option for normal class:
  - Smaller training sets are faster to train
  - Normal class is proportionally reduced
  - Faster training → more sets
- SMOTE (Synthetic Over-Sampling)
  - Far less common but interesting
  - Replicating outlier class→over-training
  - Create rare class samples for training
    - Sample from k-NN of each outlier class sample
    - Sample on line segment between point & neighbor
  - "SMOTEBoost" algorithm using boosting

How to Balance Data

- Increase number of outliers [Ling98]
  - Duplicate outliers until equal size
  - Changes only cost of misclassification
- Under-sample the non-outliers [Kubat97]
  - Emphasize the points closest to outliers
  - Under-sample distant points
- Create fake outliers
  - SMOTE (Synthetic Minority Over-sampling TEchnique) within outlier zone [Chawla 2002]
  - Use active learning to create outliers [Abe06]

Active Learning

Choose points with 2 criteria:

1) Low likelihood
   - Fit the models poorly, perhaps in tails
   - Special case for outlier detection
2) High uncertainty
   - Unclear which class they belong to
   - Standard practice for classification

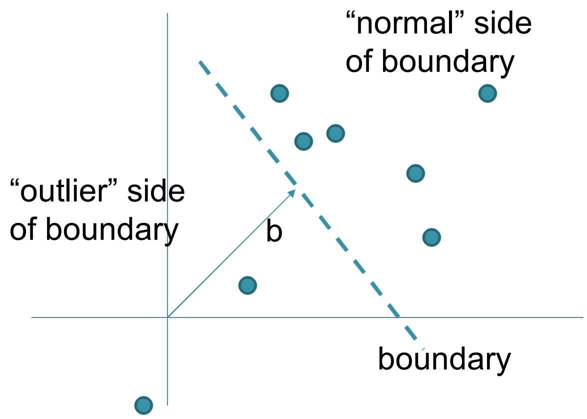
Problems with one-class model

- Data is used for training and scoring
  - Outlier affects the model
  - As remove outliers, model changes
- No separate training/testing models
  - How to do prevent overfitting?
- How could you fix this?
  - Cross-validation

1-class SVM = linear method

- Model data as 1 class
- Apply kernel transform
- Require assumption:
  - Origin (i.e. zero) of kernel-transformed data belongs to outlier class
- Create margin to the origin
  - Avoids overfitting
- Approach
  - Take X data → $\Phi(X)$ transform
  - $W \bullet \Phi(X) - b = 0$

SVM Idea



- Penalize outliers for being on wrong side
- ~Trade-off penalty for of linear separability
  - Penalize training samples on wrong side
  - Good model fit

Ensemble Methods

- Very useful for outlier detection
- Outlier version different from traditional ensembles
  - No labels for outlier class typically
- Main benefits are still the bias/variance
  - Bias is more difficult to reduce w/o "outlier" label

Ensemble Types

- Approach to variation
  - Model-centric: multiple methods/parameters
  - Data-centric: multiple data samples
- Independence of variation
  - Independent (bagging)
  - Sequential (boosting)
- Score normalization:
  - Range norm: Output x→(x-min)/(max-min)
  - Standardization: Output → (x-mean) / stdev
- Scoring:
  - Average of outputs
  - Max of outputs

Distance or Geometric Approaches

Distance-Based Methods

- Compute k-NN for all points
- Take the largest 1% perhaps that have the largest distances
- Challenged to find outliers in high dimensions
- Assumes the density is equivalent

Distance Methods

- Essentially k-NN
  - Score(Xi) = kth smallest dist to rest of X
- Highly granular: identify local outlier
- Assumes density is equal globally
- Can identify smaller outlier clusters of m
  - Need to se k > m (some say k>=m)
- Requires N^2 calculations if need score
- Faster approaches reduce computation if only need outlier/non-outlier decision

Distance Based

- Knorr & Ng 1997
- “Distance-based outliers: algorithms & applications”
- DB(D,p) = Outliers
- Object O in T is a DB(p,D) outlier if:
  - At least fraction p of objects in T lies greater than D distance from O
  - e.g. 90% of objects are at least 5 away
- $O(kN^2)$ for k dimensions, N points
- $O(c^k + N)$ for small c constant using cells

Clustering for Outlier Detection

- Advantages
  - Much faster than nearest neighbor
  - Optimizations readily available
  - Applicable to multiple types of variables
  - Intuitive
- Issues
  - Small data sets
  - Parameter choices change results
  - Noise vs. true outlier

Distance to Cluster

- Mahalanobis distance to cluster is useful metric for outliers $d(\vec{x},\vec{y}) = \sqrt{(\vec{x}-\vec{y})^T S^{-1}(\vec{x}-\vec{y})}$
  - Uses Euclidian-like distance
  - Weights features by variance
- What if the data is a spiral manifold or other non-Gaussian blob?
  - Nonlinear PCA works for global view
  - Problem is we don't necessarily want a global model for local outlier detection

Mahalanobis Distance

- Recall multivariate Gaussian distribution

$$f_{\mathbf{x}}(x_1,\ldots,x_k) = \frac{\exp\left(-\frac{1}{2}(\mathbf{x}-\boldsymbol{\mu})^{\mathrm{T}}\boldsymbol{\Sigma}^{-1}(\mathbf{x}-\boldsymbol{\mu})\right)}{\sqrt{(2\pi)^k|\boldsymbol{\Sigma}|}}$$

- Mahalanobis(X, μ, Σ) = $\sqrt{(\mathbf{x}-\boldsymbol{\mu})^{\mathrm{T}}\boldsymbol{\Sigma}^{-1}(\mathbf{x}-\boldsymbol{\mu})}$
- What if covariance is not invertible?
  - Take inverse of $\Sigma + \lambda I$ for small λ
  - Considered a form of regularization

Characteristics of Mahalanobis

- Similar to Euclidian distance but utilizes correlations between features to normalize the results
- Similar to a PCA by taking the strength of the various axes into account
- No parameters!
- Computationally reasonable
  - $O(k^2)$ for the inverse where k=#dimensions
  - O(N) for number of points

Geometric Method

- Eskin 2002 "Geometric Framework for unsupervised anomaly detection"
- Apply leader clustering with radius r
- Outliers = clusters with fewest members

Proximity-Based

- Multiple categories
  - Cluster-based
  - Distance-based (K-NN)
  - Density based
- Differ in performance
  - Cluster uses summarized representation that is potentially robust
  - Distance-based uses individual point with high granularity but $O(N^2)$

Curse of Dimensionality

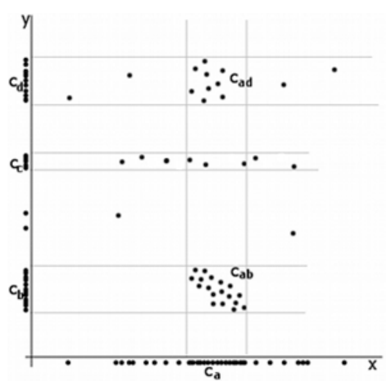
- Affects many fields
- Outlier may occur in a 2D plot but when other variables are viewed, not an outlier
- Outliers will be apparent in very of many 2D plots for N-dimensional space
  - Apparent means visually or computationally
- Noise of d-dimensional features may drown out the outlier on a few dimensions

High Dimensional Data

- Data become increasingly sparse
- Inter-point distances become somewhat equivalent

Solution for the Curse

- Find the relevant subspaces
- But, number of possible projections to subspaces is exponential



https://en.wikipedia.org/wiki/Clustering_high-dimensional_data

Grid-Based Hi-Dim search

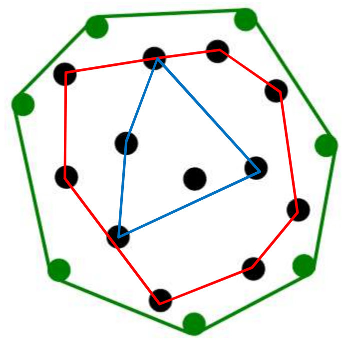
- Divide up each variable into bins of equal number of data points
- Let f = fraction of points in each bin
- If we assume independence, then
  - $N*f^k$ = expected # in any N-dim bin
  - Sqrt($N*f^k$ ($1-f^k$) = stdev of points in N-dim bin
- For large N, we can assume a normal distribution rather than Bernoulli

Genetic Algorithm on Grids

- Search is for subspace with rare combinations
- Encode bins across features "3" or "*" if don't care
- Fitness function is rare combinations

Convex Hull

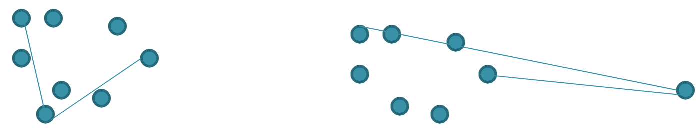
- Compute a geometric "depth" as #rings
- Avoid distribution requirements
- Do until no points:
  - Find convex hull such that all lines connecting points are within the "hull"
  - Remove hull points
  - Increase depth by 1



Computation of Depth

- Computing the convex hull can be fairly slow even for 2D or 3D yet alone ND
- Faster methods exist for this
  - ISODEPTH Ruts & Rousseeuw,1996
    - Computes depth contours efficiently for 2D
    - Scales poorly < 5000 points
  - FDC Johnson, Kwok, NG 1998
    - Computes first k contours for 2D space
    - Scales to 100K points at least
  - Quickhull for ND   Barber et al. 1996
    - Divide & conquer approach using extreme points

Angle-Based Method



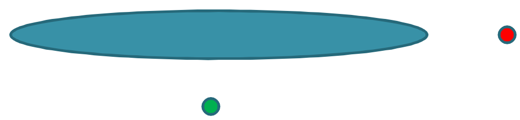
- Concept is for any triple of points
  - Angles for outlier to other points are similar
  - Angles for non-outlier vary widely
- Weighted cosine distance
  - Wcos(YX, ZX) = (YX dot ZX) / $|YX|^2$ $|ZX|^2$
- Angle-Based outlier factor(X)
  - ABOF(X) = $Var_{\{all\ Y,Z\}}$ Wcos(YX, ZX)

Computation of ABOF

- For all points is a lot of calculations
  - Could sample from the space
- Points with largest impact to ABOF(X) are closest points or the K-NN of X
  - Wcos(YX, ZX) = (YX dot ZX) / [$||YX||^2$ $||ZX||^2$]
  - Basically due to the denominator

Issues with Angle Based

- Approach useful for boundary outliers but not outliers in the middle of blob
- Which points is a greater outlier?



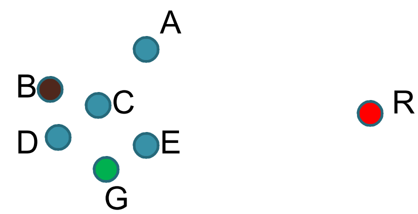
- High dimensional data
  - Initially believed angles would be better
  - Reality is they have inherent distance basis
  - All triples turn into equilateral triangles

ODIN: Reverse RNN

- K-NN alternative: use #reverse KNN
- p is reverse KNN of q iff q is among k-NN of p
- Outliers: reverse KNN < threshold
- Score = #reverse KNN
- Outlier Detection using In-degree Number (ODIN)
  - $O(N^2)$ algorithm

Reverse KNN

- p is reverse KNN of q iff q is among k-NN of p



- KNN(C) = BDE C revKNN D? Yes
- KNN(D) = BCG
- KNN(B) = CDG
- KNN(R) = AEG R revKNN anything?
- KNN(G) = DCE G revKNN C,D,E

Model Approaches

Linear Modeling

1) Regression
2) Principal Component Analysis

- Assumption:
  - Data fits a linear model
  - Data fits a lower dimensional space
  - Normal distribution about model
- What's the main difference between 1 & 2? What does that imply?

Regression

- $y = \Sigma_d \, w_i \, x_i + w_{d+1}$
- $Y = DW^T + \text{error}$
- Solution $W^T = (D^TD)^{-1}D^Ty$
  - If the parens terms is not invertible, can use regularization $W^T = (D^TD + \alpha I)^{-1}D^Ty$
- Different choices of variables will produce different fits and outliers
  - Normal residual assumption provides distribution for the outliers

Ensemble Methods

- Regression has circular logic
  - Fit the data to identify the outlier
  - Outlier can significantly change the fit
- Ensemble methods avoid this issue
  - Sample part of the data and assess fits
  - Repeat many times to score all points
  - Average the predicted outliers
- Concept applies to unsupervised
  - Treat arbitrary variable as dependent

PCA

- Related to regression
- Finds optimal k-dimensional hyperplane that minimizes the squared projection error over the remaining d-k dimensions
- PCA minimizing projection error:
  - Outliers are deviations from the principal component axes
- Score(X) = $\Sigma_j [(X - \mu)^* e_j]^2 / \lambda_j$
  - X = point, μ = centroid, e_j = eigenvector
  - Note smaller eigenvalues weighed more
  - ~Mahalanobis distance except for e_j / λ_j

PCA vs. Regression approaches

- PCA is more stable with few outliers
  - Focuses on optimal hyperplane
  - Regression focuses on optimizing against a single variable
- What if many outliers?
  - May need to run several rounds
    - Identify large outliers and remove them
    - Rerun the methods and identify mid-range outliers

PCA Methods

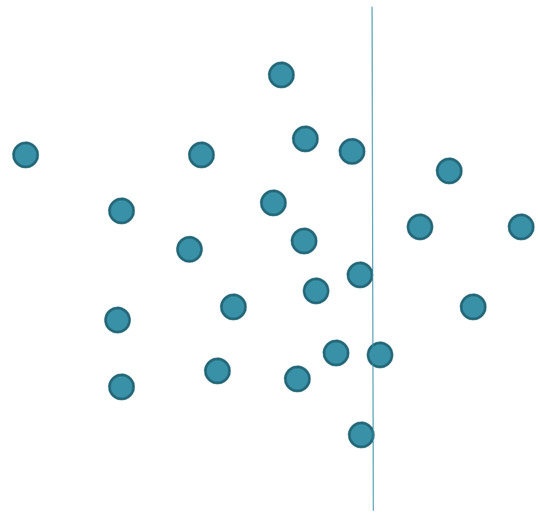
- PCA captures most variation
- Ideally, outliers won't be captured by the reduction in variance
- In reality, outliers really distort PCA
- Robust PCA methods
  - Optimize projection for <=50% non-outliers
  - Identify which points are outliers
- Alternative:
  - Find points governing low eigenvalue eigenvectors (opposite of PCA)

Tree Approaches

Create Isolation Tree

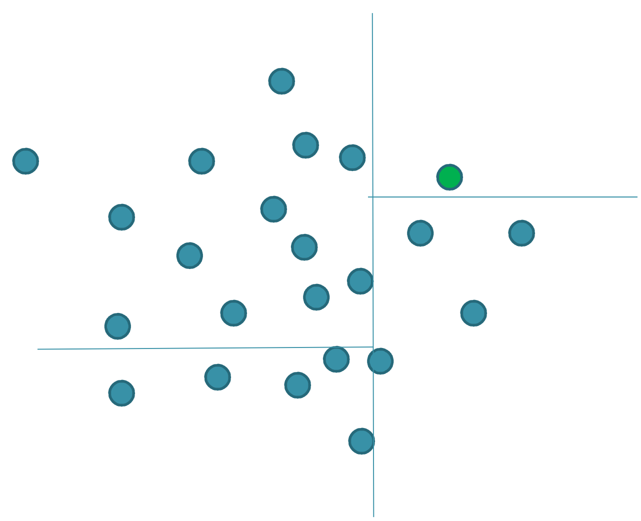
- Candidate list = nodes to split initialized by root node
- Repeat until empty:
  - R = randomly select node from candidate list
  - Select random attribute
  - Choose random threshold using uniform distribution on attribute from min to max
  - Split data at threshold into R1 & R2 as children of R
  - Add R1 and R2 into candidate list if > 1 point

Isolation Tree



- Continue until 1 record per leaf

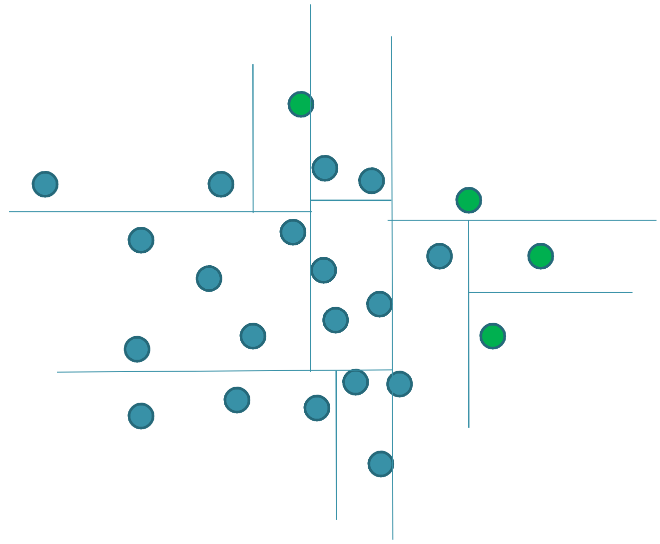
Isolation Tree



- Continue until 1 record per leaf

© 2018 Peter V. Henstock

Isolation Tree: Binary tree

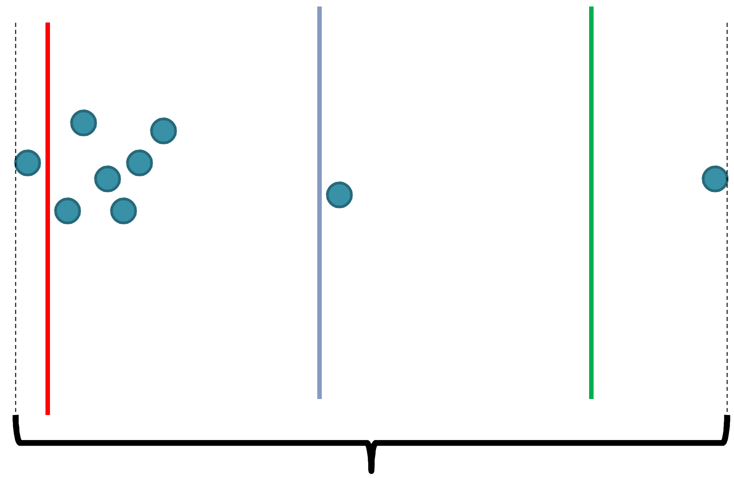


- Continue until 1 record per leaf

Isolation Forests ~ Random Forest

- Isolation forest =
  - Ensemble of isolation trees
- Isolation tree =
  - Axis-parallel cuts chosen at random to partition data across randomly selected attributes until node has one data point
- Score per ensemble: depth of tree
  - Outliers are in sparse regions so are in less deep nodes of the trees
- Average path depth across ensemble = full score
- Liu et al. ICDM 2008 “Isolation Forest”

Outlier closest to root of tree



Randomly make a vertical cut over range

Isolation Tree

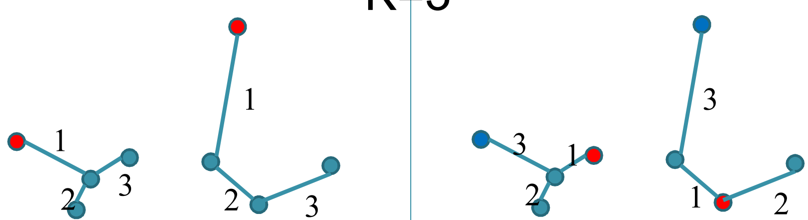
- Relatively fast to compute
  - O(N) per tree
- No parameters!
- Can have a training & test phase by dividing data into two pieces randomly
  - Better diversity
  - Better computational efficiency
  - Average path lengths across trees

Connectivity Outlier Factor

- Local sparsity assessment around neighborhoods
- Find set based nearest path
  - Found ← {point a}
  - Repeat until have r points
    - Find closest point p to any in Found with dist edgeDist
    - Add p to Found and record sequence of edgeDist
  - Chain = distances to points added
  - ACDist = $\sum_{i=1}^{r} edgeDist_i \frac{r(r-i)}{r(r-1)}$ low ACDist = denser
- COF(p) = $\frac{ACDist(p)}{\frac{1}{k}\sum_{o \in kNN(p)} ACDist(o)}$ normalized ACDist
- Comparing density of points against neighbors
  - If less dense, then more likely an outlier

Connectivity Factor

K=3



- Chains formed in order from each red point
- Chains will vary based on starting point
- Record distances *in order* of addition
- Weigh the first points more than later
- Outliers are relative to neighborhood