We acknowledge the Australian Aboriginal and Torres Strait Islander peoples as the traditional owners of the lands and waters where we live and work.

MA5810 - Introduction to Data Mining

Week 5: Unsupervised Learning – Outlier Detection & PCA

Dr Max Cao

Foundations for Data Science

- Week 1: Introduction
- Week 2: Supervised Learning – Bayes Classifiers
- Week 3: Supervised Learning kNN and Logistic Regression
- Week 4: Unsupervised Learning – Clustering
- Week 5: Unsupervised Learning – Outlier Detection and PCA
- Week 6: Notions of Basket Analysis and Recommender Systems

Reading

For further details on PCA

Unsupervised Learning, Chapter 10. section 10.2

An Introduction to Statistical Learning with Applications in R. by James, G., Witten, D., Hastie, T., and Tibshirani, R. (2013). New York, NY: Springer. (freely available online at http://wwwbcf.usc.edu/~gareth/ISL/)

Unsupervised Learning – Outlier Detection and PCA

- Unsupervised outlier learning
  - Parametric statistical approaches- estimating the parameters of the distributions
  - Non-parametric statistical approaches- KNN outlier
- Local outlier factor (LOF)
- Global-Local Outlier Scores from Hierarchies (GLOSH)
- Principal Component Analysis (PCA)

What is outlier/anomaly

- "Something that deviates from what is standard, normal, or expected" Oxford Dictionary.
- There are many approaches to determine anomalies in datasets. We will look at unsupervised methods.
- Basically, you run a clustering algorithm and ask yourself are those points close or far away from clusters.
- Usually, additional information or domain knowledge is required to determine outlier.
- May want to remove outliers or be interested in the outliers themselves.

What are sources of outlier

- Data entry errors (human errors)
- Measurement errors (instrument errors)
- Experimental errors (data extraction or planning/executing errors)
- Intentional (dummy outliers made to test detection methods)
- Data processing errors (data manipulation or unintended mutations)
- Sampling errors (extracting or mixing data from wrong)
- Natural (not an error, **novelties** in data)

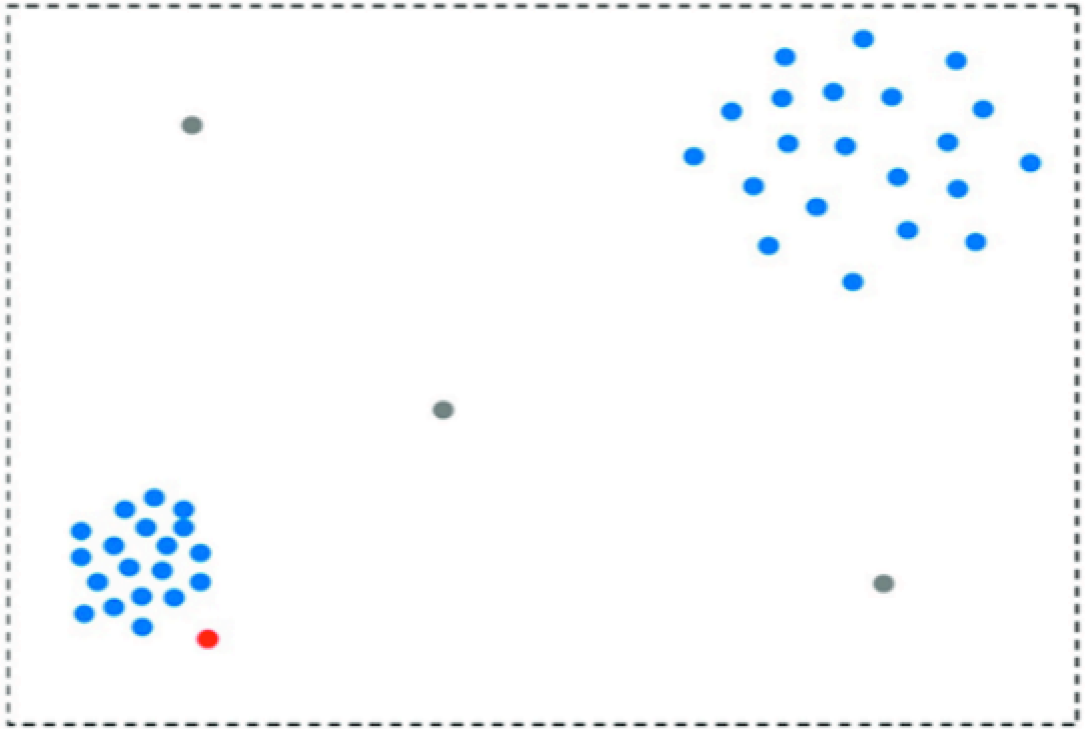
Use cases for unsupervised outlier detection

- Fraud Detection
- Network Security
- Manufacturing Quality Control
- Environmental Monitoring:
- Retail Inventory Management
- Predictive Maintenance
- Image and Video Analysis
- Social Media Analysis
- Energy Consumption Monitoring
- Transportation and Logistics
- Market Research

Should outlier be removed?

- Understand the Data
- Consider Data Quality
- Impact on Analysis
- Research Goals
- Consult Experts
- Robust Methods
- Document Decisions
- Sensitivity Analysis

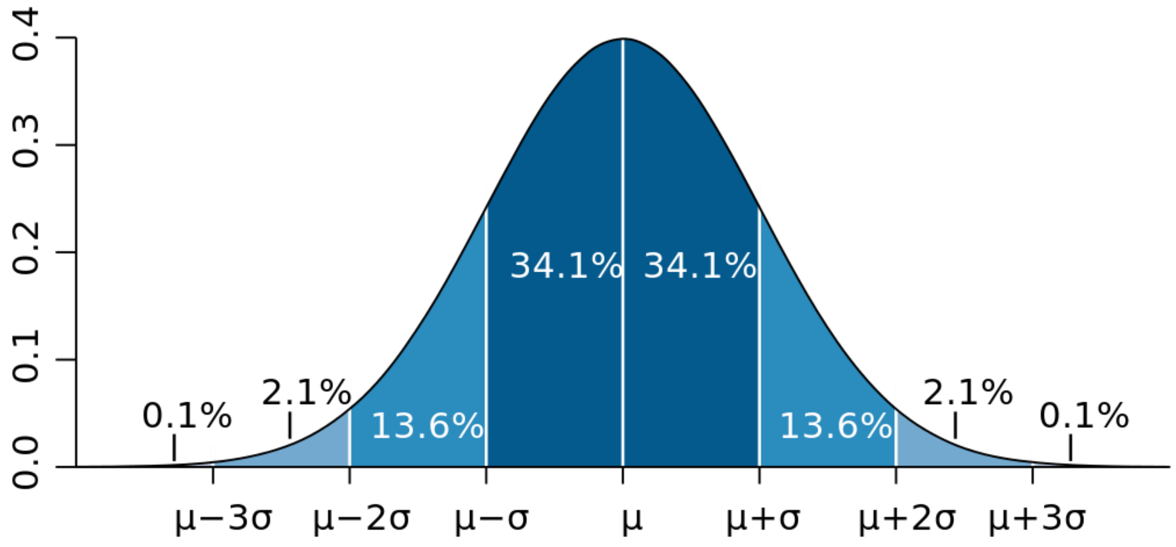
Local and global outliers?

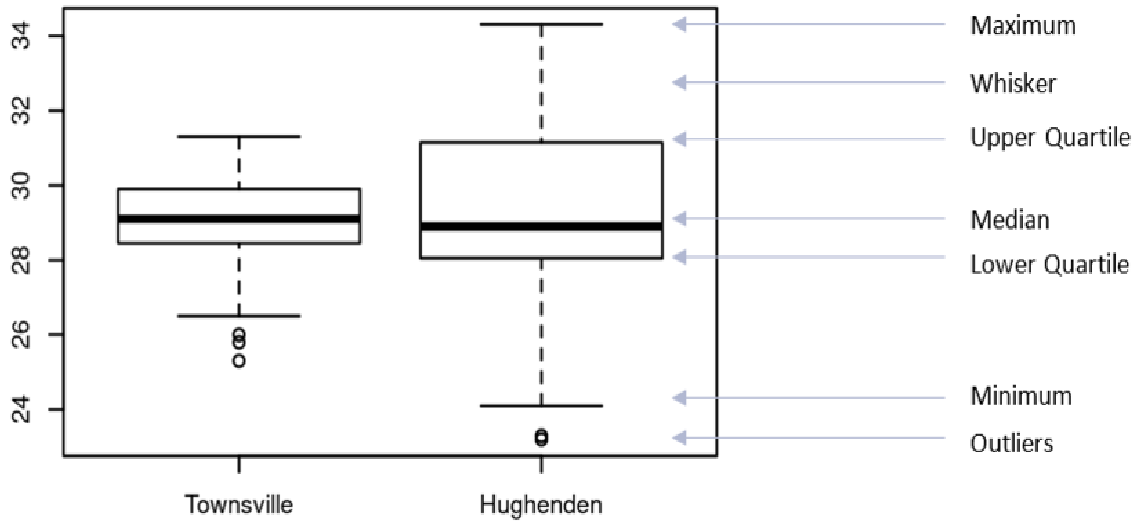


Source: Alghushairy, Omar & Alsini, Raed & Soule, Terence & Ma, Xiaogang. (2020). A Review of Local Outlier Factor Algorithms for Outlier Detection in Big Data Streams. Big Data and Cognitive Computing. 5. 1. 10.3390/bdcc5010001

Outlier Modeling Example – Parametric Statistical Model

- Fit a parametric distribution to the data by estimating its parameters.
- Assume normal observations follow a univariate Gaussian distribution.
- Parameters (mean and standard deviation) are estimated from the data.
- Observations in the tails of the distribution are flagged as outliers.
- Observations deviating more than three standard deviations from the mean (3σ rule) are considered outliers.


0.0
0.1
0.2
0.3
0.4
0.1%
2.1%
13.6%
34.1%
34.1%
13.6%
2.1%
0.1%
μ−3σ
μ−2σ
μ−σ
μ
μ+σ
μ+2σ
μ+3σ


34
32
30
28
26
24
Townsville
Hughenden
Maximum
Whisker
Upper Quartile
Median
Lower Quartile
Minimum
Outliers

Outlier techniques

The outliers' techniques used in this course can be grouped into two categories:

- Distance based methods: Outliers are data points most distant from other points. Try to determine which points are far away from all other points.

- Density based methods: Outliers are data points in low density regions. Try to determine which data points in your index are far away from the bulk of the data.

Distance -based approach: KNN outlier

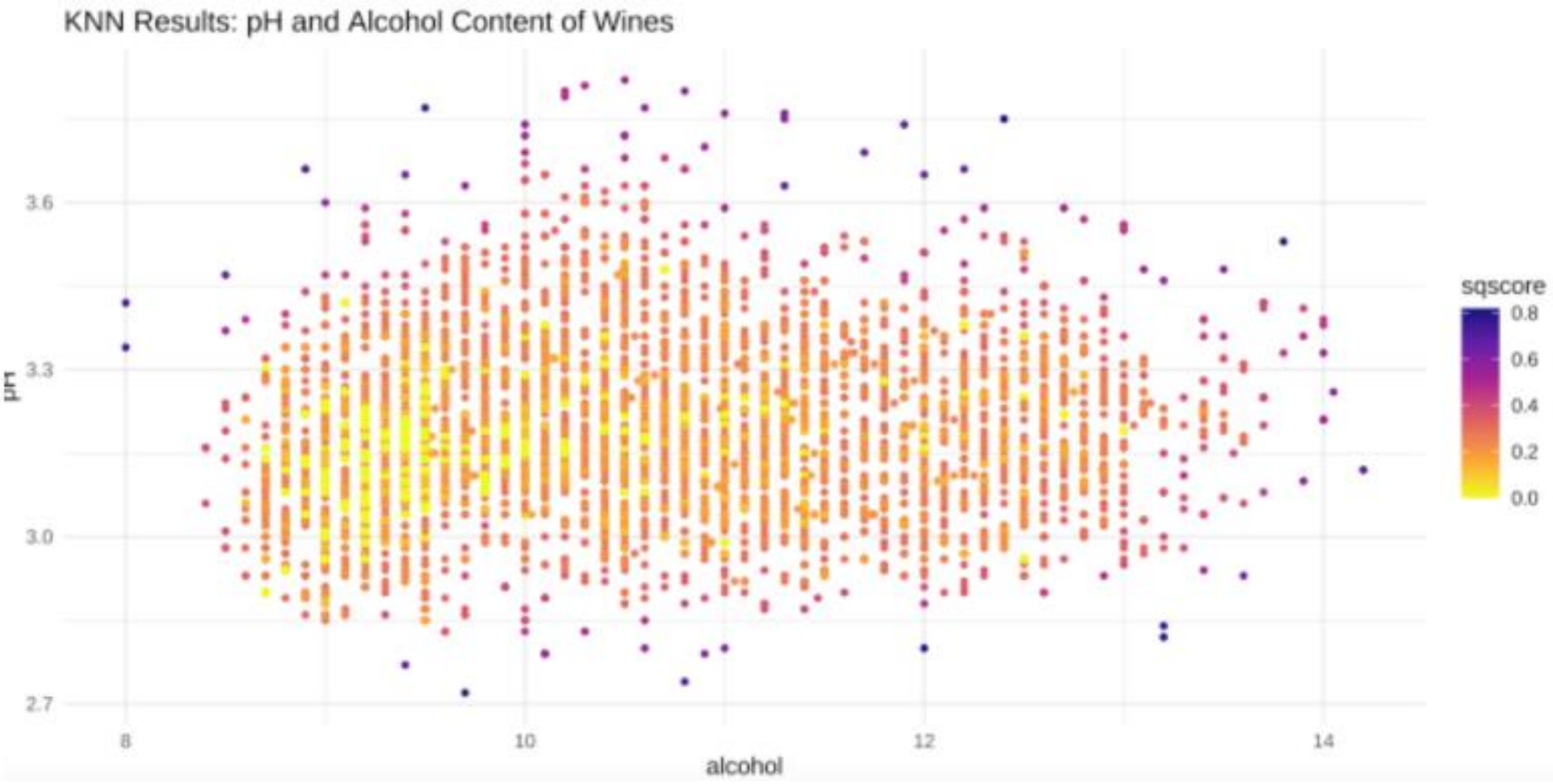
- Distance-based K-Nearest Neighbors (KNN) outlier analysis is a technique used to identify outliers within a dataset based on their distance to their K nearest neighbors.
- Outliers are data points that deviate significantly from the rest of the data and may represent anomalies, errors, or interesting events. In this method, each data point's "outlyingness" is determined by how far away it is from its K nearest neighbors.
- KNN outlier detection:
  1. For each point, compute the outlier score (the distance to its kth nearest neighbor)
  2. Sort these values.
  3. Choose the biggest values as outliers.

Weighted KNN outlier

- Weighted KNN outlier:
  - Computes the outlier score as the (Weighted) average distance to the k nearest neighbors.
  - Relies on distances to all k neighbors rather than just the kth nearest neighbor.
  - Provides a more balanced measure of an observation's deviation.

- Scores: Scoring techniques assign an outlier score to each data point depending on the degree to which that instance is considered an outlier. Thus, the output of such techniques is a ranked list of outliers. You may choose to either analyze the top few outliers or use a cut-of threshold to select the outliers.

Example – KNN Outlier



Example – Detecting anomalous heath readings

Suppose you work in a healthcare setting where you're responsible for monitoring patient health readings collected from various wearable devices. You want to identify patients with potentially anomalous health readings using distance-based KNN outlier analysis.

Data Collection: You gather a dataset of patient health readings, including metrics like heart rate, blood pressure, temperature, and respiratory rate. Each patient's readings are collected at regular intervals.

Data Preprocessing: After collecting the data, you preprocess it by normalizing the features to ensure that each feature contributes equally to the distance calculations. For example, you might scale all the features to have a mean of 0 and a standard deviation of 1.

Choosing K and Distance Metric: You decide to use K=5 for this analysis, meaning you will consider each patient's 5 nearest neighbors. For the distance metric, you opt for the Euclidean distance, which is commonly used for numerical data.

Example – Detecting anomalous heath readings

Calculating Distances: For each patient's health readings, calculate the Euclidean distances to the 5 nearest neighbors' health readings.

Ranking Distances: Rank the distances in ascending order for each patient's readings.

Setting a Threshold: Determine a threshold value for distances based on the distribution of distances. Readings with distances beyond this threshold are potential outliers.

Identifying Outliers: Compare each patient's readings' distances to the threshold. If the distance exceeds the threshold, the patient's readings are considered an outlier.

Results: You find that Patient Y's health readings have significantly higher distances to their nearest neighbors compared to other patients. This suggests that Patient Y's health readings are different from the norm, which could be due to a medical condition or sensor malfunction. You decide to investigate further and discover that Patient Y is indeed experiencing a health issue that requires immediate attention.

KNN outlier

- Both KNN outlier and Weighted KNN outlier are Global methods.
  - The degree of *outlyingness* is computed using the entire dataset as a reference.
  - Points in a large, sparse cluster can have higher outlier scores.
  - Inliers from sparse clusters may appear more "outlying" than local outliers near smaller, denser clusters.

- In this measures, larger values are more likely to indicate an outlier. In order to apply this method, data should first be scaled to avoid sensitivity to the scales of individual variables.

Local outlier factor (LOF)

- Local outlier factor (LOF) is a method for identifying outliers in a dataset by assessing the density-based behavior of data points in their local neighborhoods.
- Unlike global outlier detection methods that consider the entire dataset, LOF focuses on the relative density of points within their local cluster.
- It quantifies how much a data point's density deviates from the density of its neighbors, making it particularly effective at identifying local outliers.
- Inliers= local density will be similar to that of its neighbors.
- Outlier = its local density will be lower than that of its nearest neighbors. Hence the anomalous instance will get a higher LOF score.

Local outlier factor (LOF)

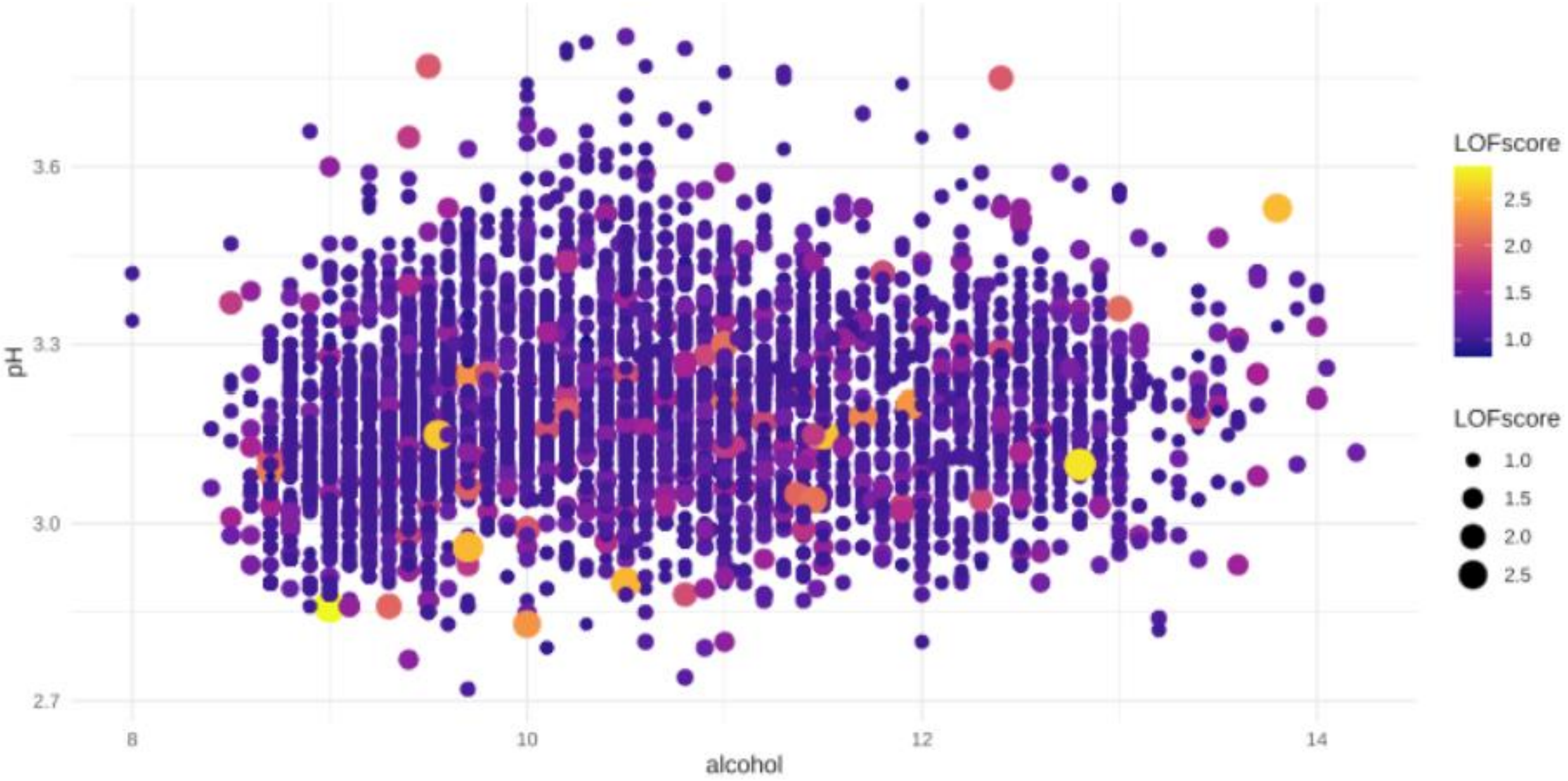
Basic procedure:

- k nearest neighbors are found for each observation
- the local density in the neighborhood of the observation is estimated.
- compare this local density with the ones for the nearest neighbors of the point. The resulting score is an average ratio of local densities.

Interpreting LOF - LOF is a ratio of densities.

- LOF ≈ 1: point is in a cluster (region with homogeneous density around the point and its neighbors)
- LOF(x) < 1 higher density than neighbor (inlier)
- LOF(x) > 1 lower density than neighbor (outlier)
- Large LOF values indicate more isolated point

Local outlier factor (LOF)



Example – Identifying anomalies in Taxi Rides

Suppose you are working with a dataset of taxi ride information in a city, and you want to use the Local Outlier Factor (LOF) algorithm to identify potentially anomalous taxi rides.

Anomalies in this context could represent fraudulent activities, unusual routes, or abnormal passenger behaviors.

Data Collection: You collect a dataset containing information about taxi rides, including start and end locations, distance, duration, fare, and passenger count.

Choosing k (Number of Neighbors): You decide to set the value of k, the number of neighbors to consider, as 15. This means that for each taxi ride, you will assess its density in relation to its 15 nearest neighbors.

Calculating Reachability Distance: For each taxi ride, calculate the reachability distance from itself to its 15 nearest neighbors based on relevant attributes like distance and fare.

Example – Identifying anomalies in Taxi Rides

Calculating Local Reachability Density (LRD): Compute the Local Reachability Density (LRD) for each taxi ride by averaging the inverse of the reachability distances from its 15 nearest neighbors.

Calculating Local Outlier Factor (LOF): Calculate the Local Outlier Factor (LOF) for each taxi ride by comparing its LRD to the average LRD of its 15 nearest neighbors.

Setting a Threshold: Determine a threshold value for LOF scores based on the distribution of LOF scores. Taxi rides with LOF scores exceeding this threshold are considered potential outliers.

Identifying Anomalies: Compare each taxi ride's LOF score to the threshold. If the LOF score is above the threshold, the taxi ride is flagged as a potential anomaly.

Results: You discover that a specific taxi ride, Ride X, has a high LOF score compared to its neighbors. Upon further investigation, you find that Ride X took an unusually long route, had an unusually low fare, and had more passengers than usual for that route. These factors suggest that Ride X might be an anomaly, possibly indicating a fraudulent or unusual taxi ride.

KNN Outlier vs LOF

- LOF is designed to address some of the limitations of KNN, particularly when it comes to handling varying data densities and local anomalies.
- LOF's focus on local density makes it a robust choice for detecting anomalies in complex datasets with varying patterns of data points.
- The choice of K (number of neighbors) is also important in LOF, but LOF is less sensitive to the specific value of K.
- KNN's results can be harder to interpret, especially when the dataset has complex or high-dimensional features. LOF provides a more intuitive interpretation by focusing on the relative density of points and their deviations from local patterns.
- KNN can be effective for global outlier detection when the density of data points is relatively uniform. LOF is well-suited for detecting local anomalies in datasets with varying densities, making it valuable for many real-world scenarios.
- However, the choice between the two methods should depend on the specific characteristics of your data and your outlier detection goals.

Pros & Cons of KNN based Outlier Techniques

Advantages:

- They are purely data driven.
- Adapting nearest neighbor based techniques to a di!erent data type is straight- forward, and primarily requires defining an appropriate distance measure for the given data.

Disadvantages :

- For unsupervised techniques, if the data has normal instances that do not have enough close neighbors or if the data has anomalies that have enough close neighbors, the technique fails to label them correctly, resulting in missed anomalies.
- The computational complexity
- Performance of a nearest neighbor based technique greatly relies on a distance measure, defined between a pair of data instances, that can effectively distinguish between normal and anomalous instances. Defining distance measures between instances can be challenging when the data is complex, e.g. graphs, sequences, etc.

Global-Local Outlier Scores from Hierarchies (GLOSH)

- Global-Local Outlier Scores from Hierarchies (GLOSH) is an approach for detecting outliers or anomalies within hierarchical data structures.
- This technique combines both global and local perspectives to assign outlier scores to individual data points within a hierarchy, taking into account their relationships within the hierarchy.
- GLOSH is particularly useful in scenarios where anomalies might not be easily detectable when looking at the entire dataset in isolation but become apparent when considering the hierarchical relationships.
- It's important to note that GLOSH is just one approach among many for detecting outliers in hierarchical data. The effectiveness of this technique depends on the specific characteristics of the data and the problem at hand.
- Domains - fraud detection, network security, supply chain management

Global-Local Outlier Scores from Hierarchies (GLOSH)

- GLOSH is an outlier score derived from the HDBSCAN* density-based clustering hierarchy.
- It calculates a ratio involving the density around each observation and the densest point in its closest cluster, considering density-based connectivity.
- For local outliers, GLOSH compares the observation to nearby density-based clusters in the hierarchy, making it locally sensitive.
- For global outliers, GLOSH compares the observation to the entire dataset (root of the hierarchy), making it globally sensitive.

Global-Local Outlier Scores from Hierarchies (GLOSH)

The GLOSH score ranges from 0 to 1:

- Values close to 0 indicate inliers.
- Values close to 1 indicate outliers.
- Global outliers are ranked higher than local outliers, which are ranked higher than inliers.

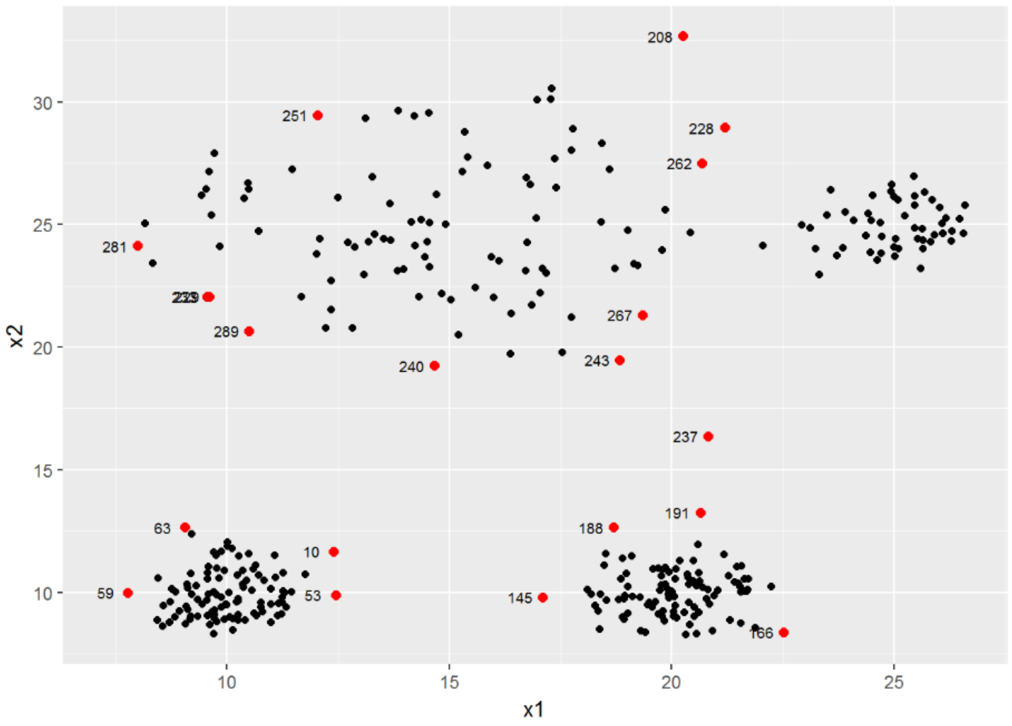


Figure 5.3.1. GLOSH ($MinPts = 4$): Red points are the top 20 outliers, with their IDs.

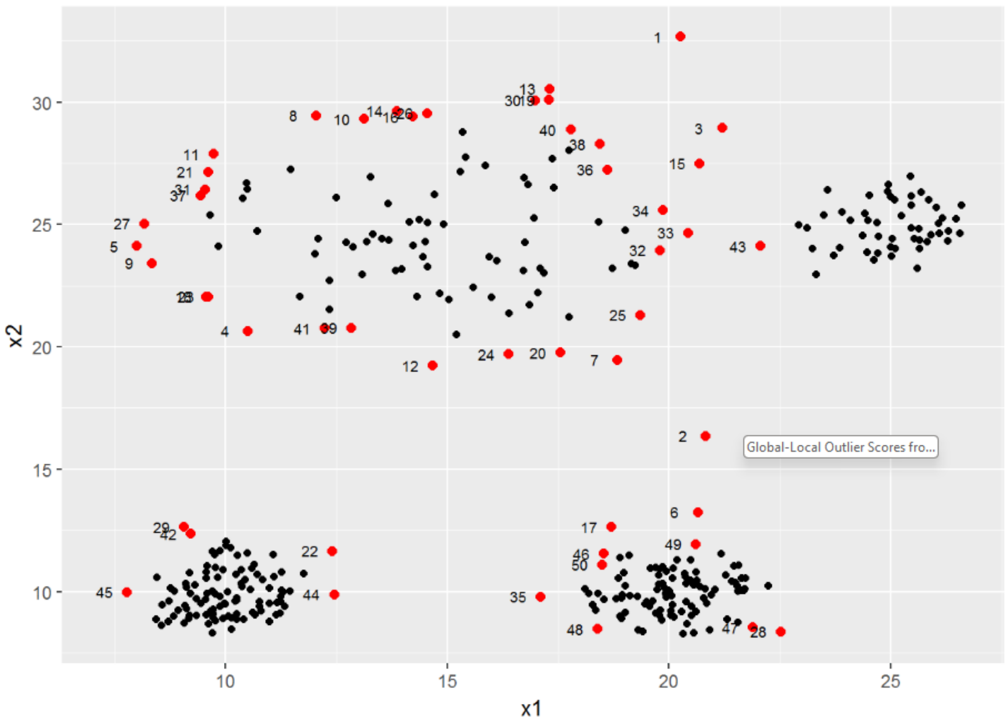


Figure 5.3.5. GLOSH ($MinPts = 15$): Red points are the top 50 outliers, with their outlier ranks.

Principal Component Analysis (PCA)

- PCA is a multivariate analysis technique, allows us to reduces a large set of correlated variables into a smaller set.
- PCA Creates representative variables called principal components, that captures most of the variability in the original dataset.
- Purposes:
    - Dimensionality reduction
    - Noise reduction
    - Data visualization
    - Addressing multicollinearity
- PCA is widely used in various data mining tasks, including machine learning, pattern recognition, and clustering, to preprocess data before feeding it into algorithms or for exploratory data analysis. It enables data scientists and analysts to handle large datasets more efficiently, reduce computational costs, and gain valuable insights from complex data.

The Curse of Dimensionality

- The **curse of dimensionality** refers to the challenges and problems that arise when dealing with high-dimensional data. It is a critical concept in data analysis, machine learning, and other fields where datasets have a large number of features/variables.
- On a high level, the curse of dimensionality is related to the fact that as dimensions (variables/features) are added to a data set, the average and minimum distance between points (observations) increase.
- The curse of dimensionality manifests in several ways:
  - Data sparsity
  - Increased computational complexity
  - Overfitting
  - Diminished discriminative power
  - Increased sample size requirements
  - Loss of intuition
- **The curse of dimensionality underscores the importance of dimensionality reduction techniques like Principal Component Analysis (PCA)**

Example: Disease Diagnosis using PCA in Healthcare

Scenario: Consider a dataset containing various health-related measurements for a group of patients, such as blood pressure, cholesterol levels, body mass index (BMI), and other clinical indicators. The goal is to use PCA to assist in disease diagnosis and prediction.

Steps:

Data Collection: Gather data from patients, including various health measurements and clinical indicators.

Data Preprocessing: Ensure that the data is cleaned, missing values are handled, and outliers are addressed. Standardize the data to have zero mean and unit variance, as this is often necessary for PCA.

PCA Calculation: Apply PCA to the standardized dataset. This will help identify patterns and relationships between the different health measurements. The principal components represent linear combinations of the original features that capture the most significant variance in the data.

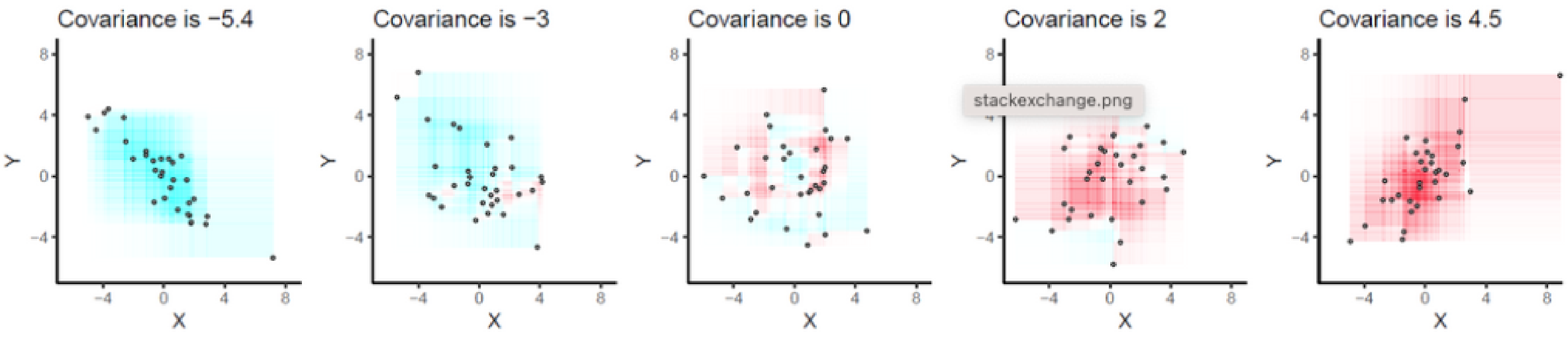
Explained Variance: Analyze the explained variance for each principal component. This helps determine how much of the total variance in the dataset is captured by each component. You can decide how many principal components to retain based on a desired level of explained variance (e.g., retaining components that capture 95% of the total variance).

Feature Interpretation: Examine the loadings of the original features on the retained principal components. Positive or negative loadings indicate how much each feature contributes to the particular principal component. This can provide insights into which health measurements are strongly associated with certain patterns or conditions.

Diagnosis and Prediction: Use the reduced-dimensional data obtained from PCA for disease diagnosis and prediction. You can employ various machine learning algorithms (such as logistic regression, Naive Bayes etc.) using the principal components as input features. These models can predict the presence of certain diseases or health conditions based on the reduced feature set.

How does PCA work?

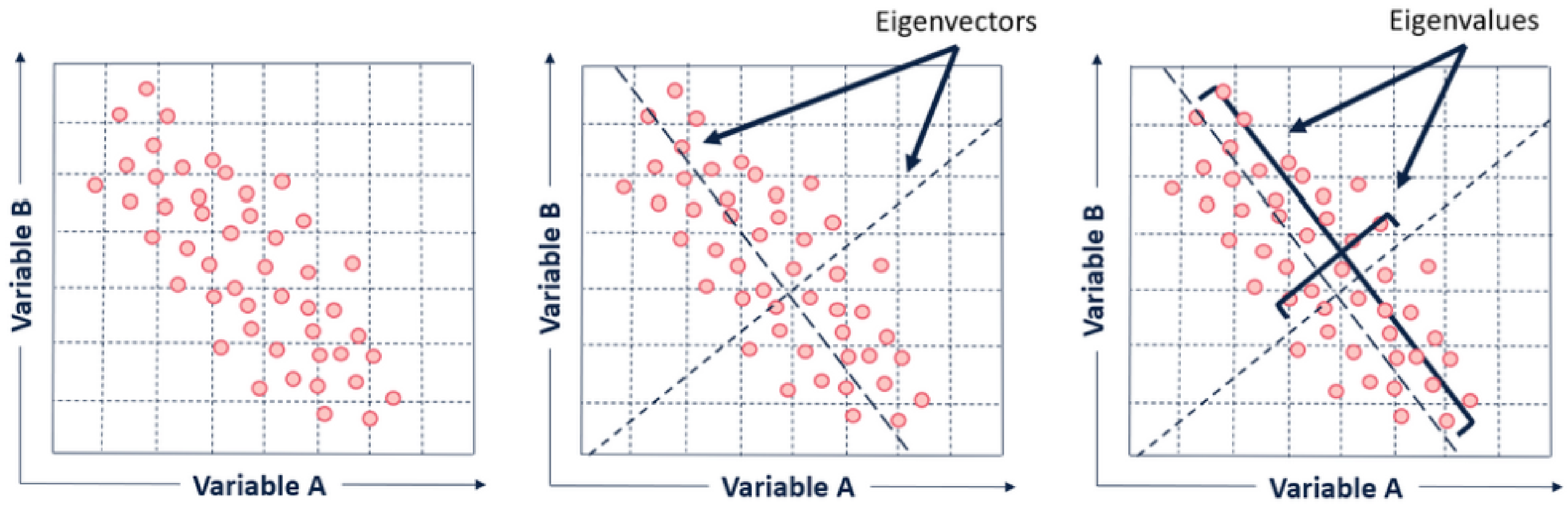
Covariance Matrix: The first step in PCA involves computing the covariance matrix of the original data. The covariance matrix summarizes the relationships between all pairs of features and their variances. It describes how related the values of the features are to one another. As the observed values of feature x increase is the same true for feature y? A large covariance value (positive or negative) indicates that the features have a strong linear relationship with one another. Covariance values close to 0 indicate a weak or non-existent linear relationship. In a nutshell, the covariance matrix in PCA helps us figure out how different pieces of data relate to each other and allows us to find the most important directions in the data.



Source:https://stats.stackexchange.com/questions/18058/how-would-you-explain-covariance-to-someone-who-understands-only-the-mean

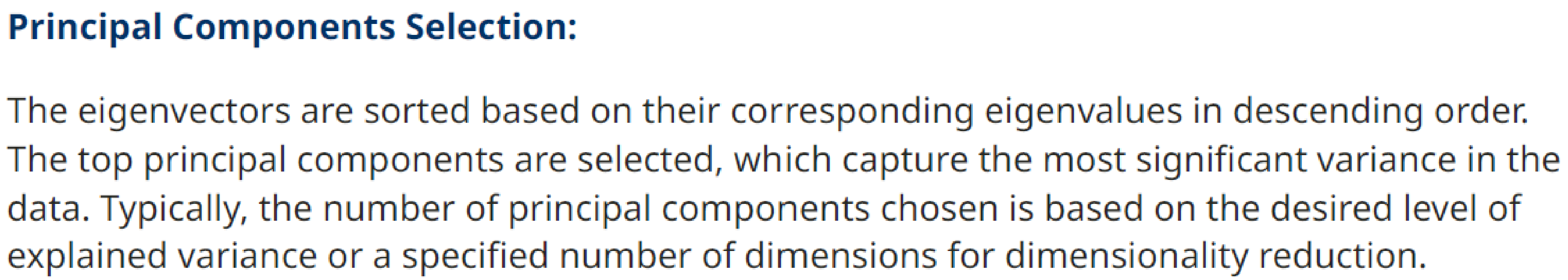
How does PCA work?

Eigenvectors and Eigenvalues: Next, PCA calculates the eigenvectors and eigenvalues of the covariance matrix. Eigenvectors represent the directions (principal components) along which the data varies the most, and eigenvalues indicate the variance explained by each eigenvector.



Source: https://community.alteryx.com/t5/Data-Science/Tidying-up-with-PCA-An-Introduction-to-Principal-Components/ba-p/382557

How does PCA work?



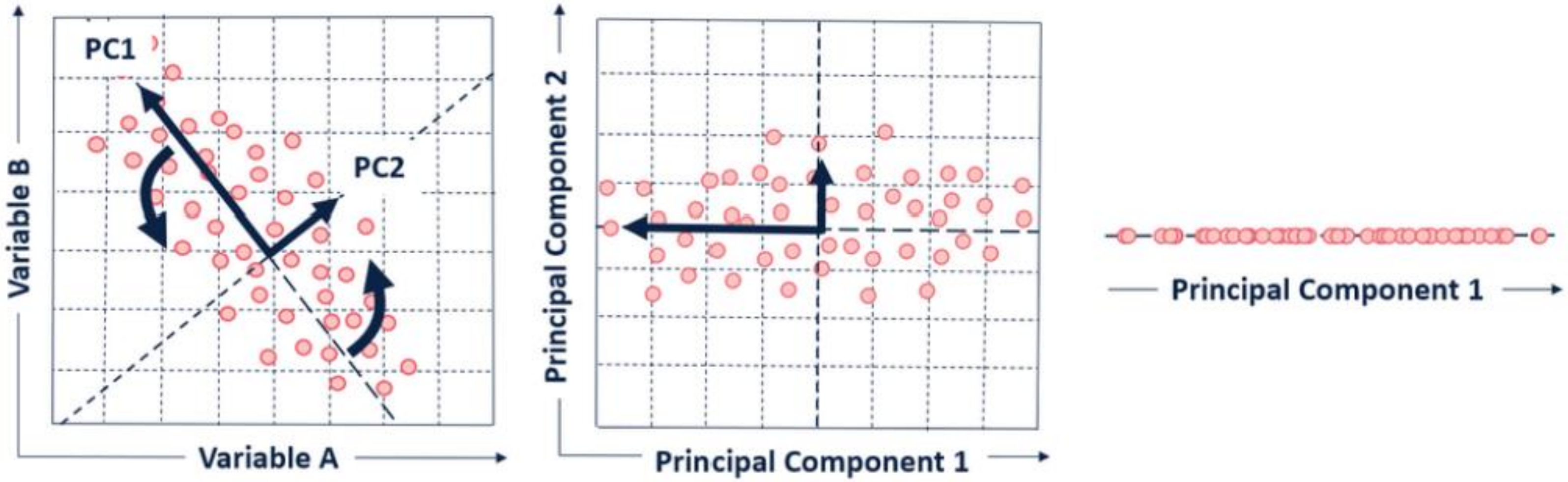
Principal Components Selection:

The eigenvectors are sorted based on their corresponding eigenvalues in descending order. The top principal components are selected, which capture the most significant variance in the data. Typically, the number of principal components chosen is based on the desired level of explained variance or a specified number of dimensions for dimensionality reduction.

How does PCA work?

Dimensionality Reduction:

Once the principal components are determined, the data is transformed by projecting it onto the new coordinate system represented by these components. This effectively reduces the number of dimensions to the selected principal components.



Source: https://community.alteryx.com/t5/Data-Science/Tidying-up-with-PCA-An-Introduction-to-Principal-Components/ba-p/382557

PCA - Desired level of explainability

The desired level of explainability in Principal Component Analysis (PCA) is generally determined by the ***trade-off between retaining as much information as possible from the original data while reducing its dimensionality***.

There's no fixed rule for deciding the exact level of explainability, as it depends on the specific goals and constraints of your analysis.

- Explained variance
- Scree plot
- Domain knowledge
- Validation
- Trail and Error

PCA – Applications

- PCA is widely used for dimensionality reduction, serving as a pre-processing step before applying other algorithms (including supervised methods), particularly in high-dimensional spaces.
- PCA aids in visualization for n-dimensional datasets ($n > 3$), enabling the creation of 2D or 3D plots using the first few components to approximate the data.

Enough chatter- Let’s get down to R!