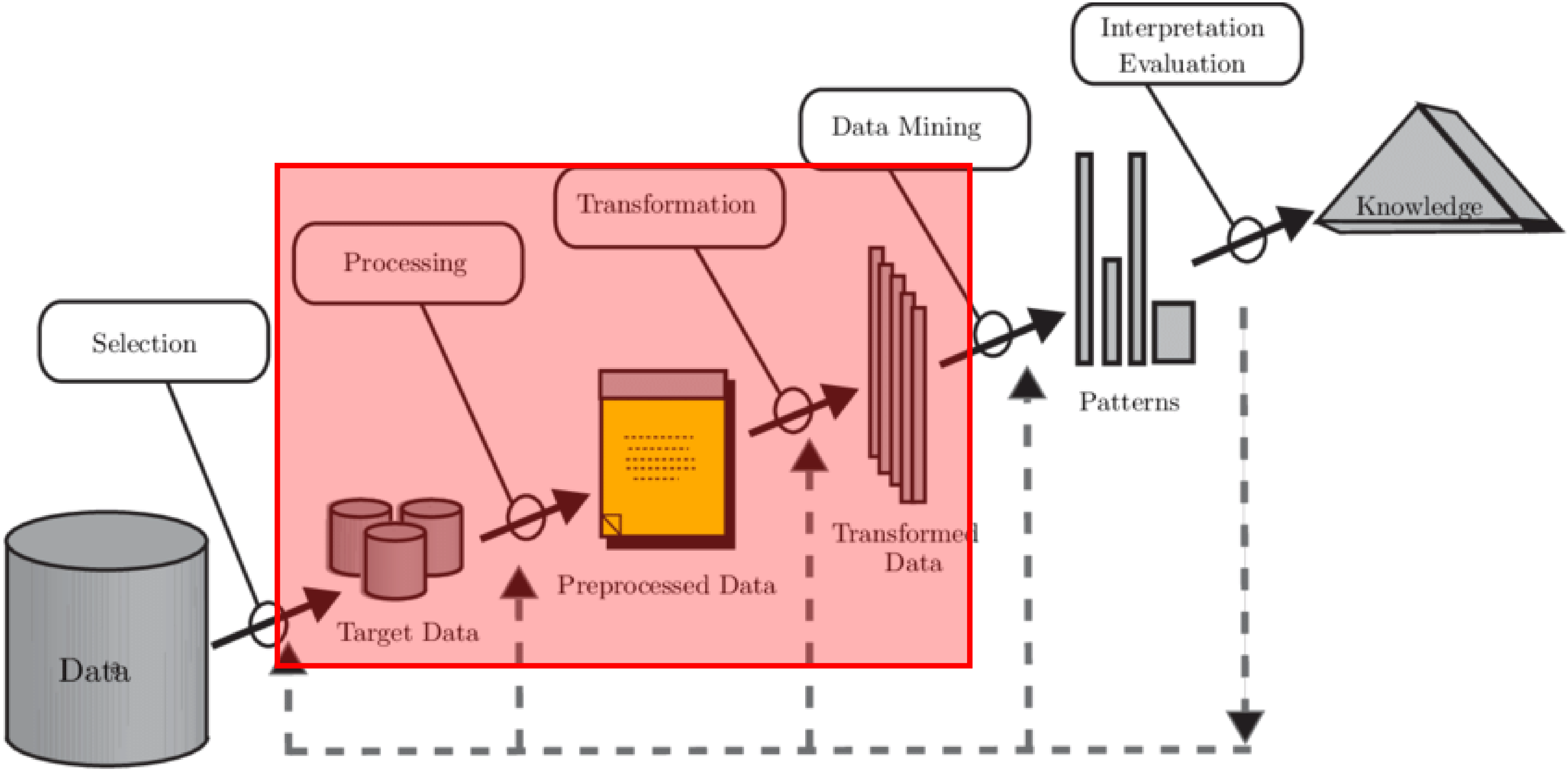
Data Preprocessing

Lecture CS1AC16

Dr Varun Ojha
University of Reading

02/03/2022

Module Outline



Data Preprocessing

- **Approx. 80% of effort in data science is data preprocessing**

- **Dirty to clean data**
  - Clean Data -> proper representation
  - Clean Data -> high quality of data

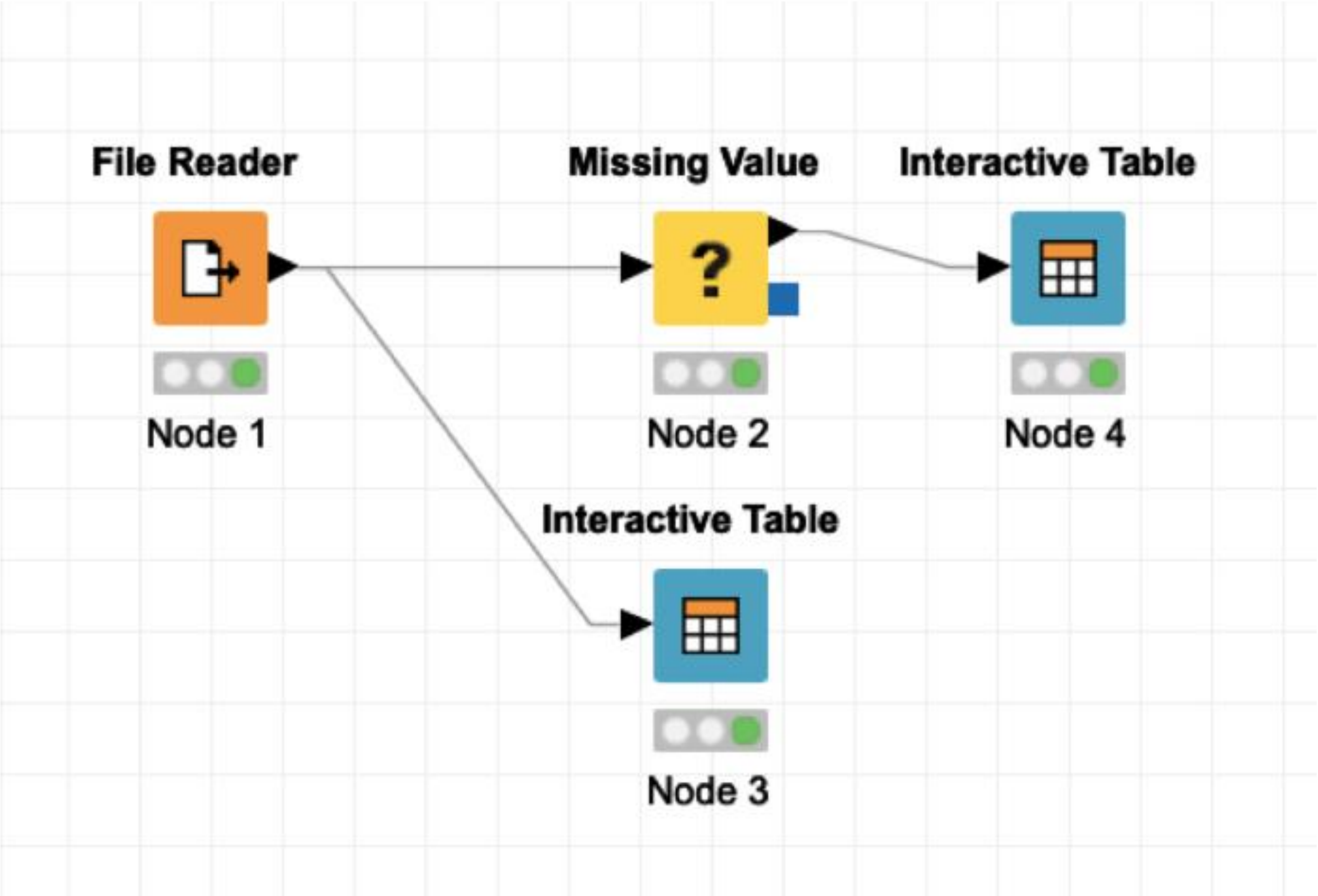
- **Steps to clean data**
  - Dealing with Missing values
  - Attribute Transformation (Binarization and Discretizatio
  - Sampling
  - Feature Selection
  - Dimensionality Reduction (Feature Engineering)

Missing Values

- **Reasons for missing values**
  - Information is not collected
  - Attributes may not be relevant
  - Data discarded as an outlier

- **Methods to handle missing values**
  - Remove all objects that have missing data
  - Estimate Missing Values (Data Imputation)

Workflow for Missing Values in KNIME



Iris Data (Multivariate and Unorder Data)

- Missing data is shown as "?" in an interactive table in KNIME workflow

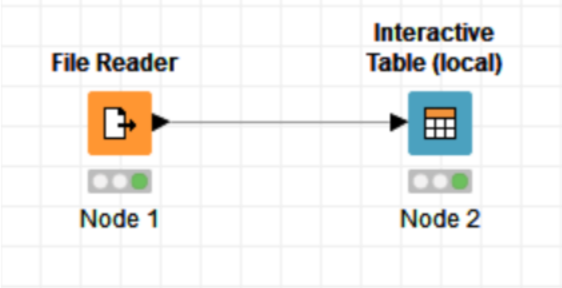


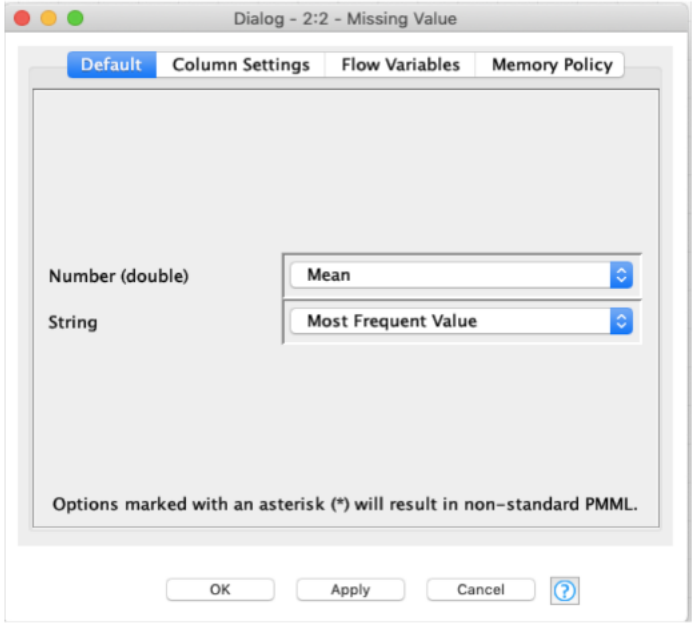
Table View - 2:3 - Interactive Table

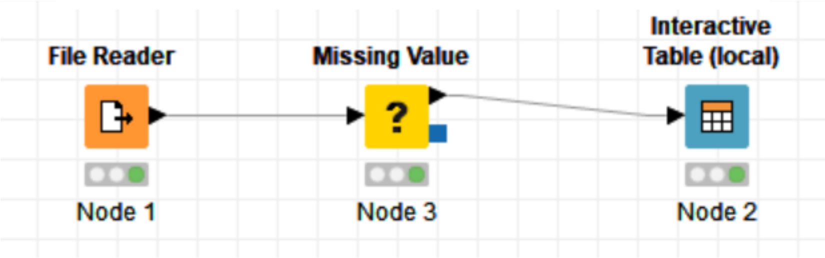
File Hilite Navigation View Output

Row ID	D Col0	D Col1	D Col2	D Col3	S Col4
Row0	5.1	3.5	1.4	0.2	Iris-setosa
Row1	4.9	3	1.4	0.2	Iris-setosa
Row2	4.7	3.2	1.3	0.2	Iris-setosa
Row3	4.6	3.1	1.5	0.2	Iris-setosa
Row4	5	3.6	1.4	0.2	Iris-setosa
Row5	5.4	3.9	1.7	0.4	Iris-setosa
Row6	4.6	3.4	?	0.3	Iris-setosa
Row7	5	3.4	1.5	0.2	Iris-setosa
Row8	4.4	2.9	1.4	0.2	Iris-setosa
Row9	4.9	3.1	1.5	0.1	?
Row10	5.4	3.7	1.5	0.2	Iris-setosa
Row11	4.8	3.4	1.6	0.2	Iris-setosa
Row12	4.8	3	1.4	0.1	Iris-setosa
Row13	4.3	3	1.1	?	Iris-setosa
Row14	5.8	4	1.2	?	Iris-setosa
Row15	5.7	4.4	1.5	0.4	Iris-setosa
Row16	5.4	3.9	1.3	0.4	Iris-setosa
Row17	5.1	3.5	1.4	0.3	Iris-setosa
Row18	?	?	1.7	0.3	Iris-setosa
Row19	5.1	3.8	1.5	0.3	Iris-setosa
Row20	5.4	3.4	1.7	0.2	Iris-setosa
Row21	5.1	3.7	1.5	0.4	Iris-setosa

Configure Missing Value Node

- Unorder data:
  - Use **mean** and **mode** (Most Frequent Value)





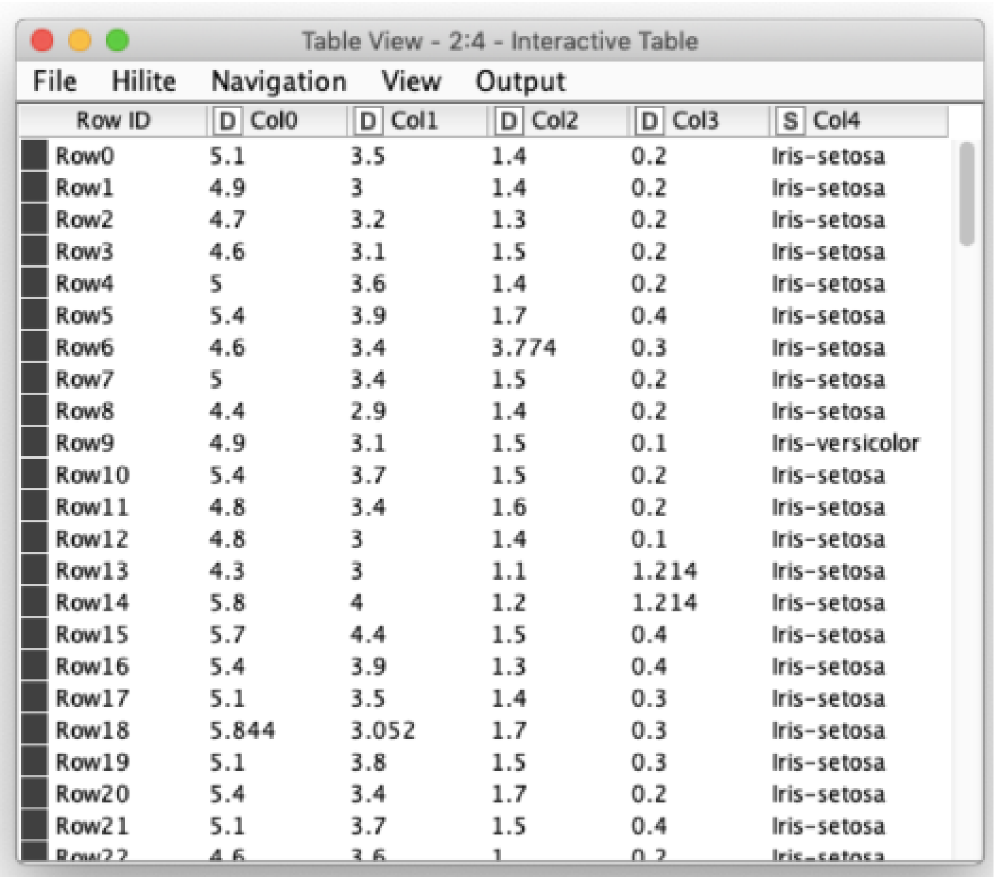


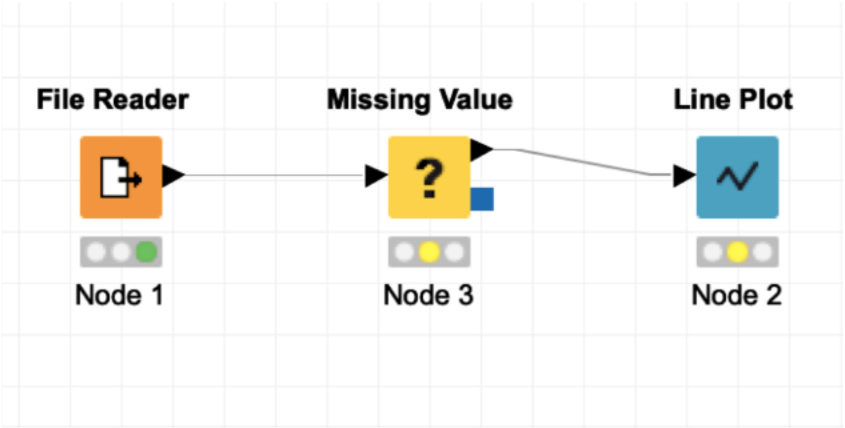
Table View - 2:4 - Interactive Table

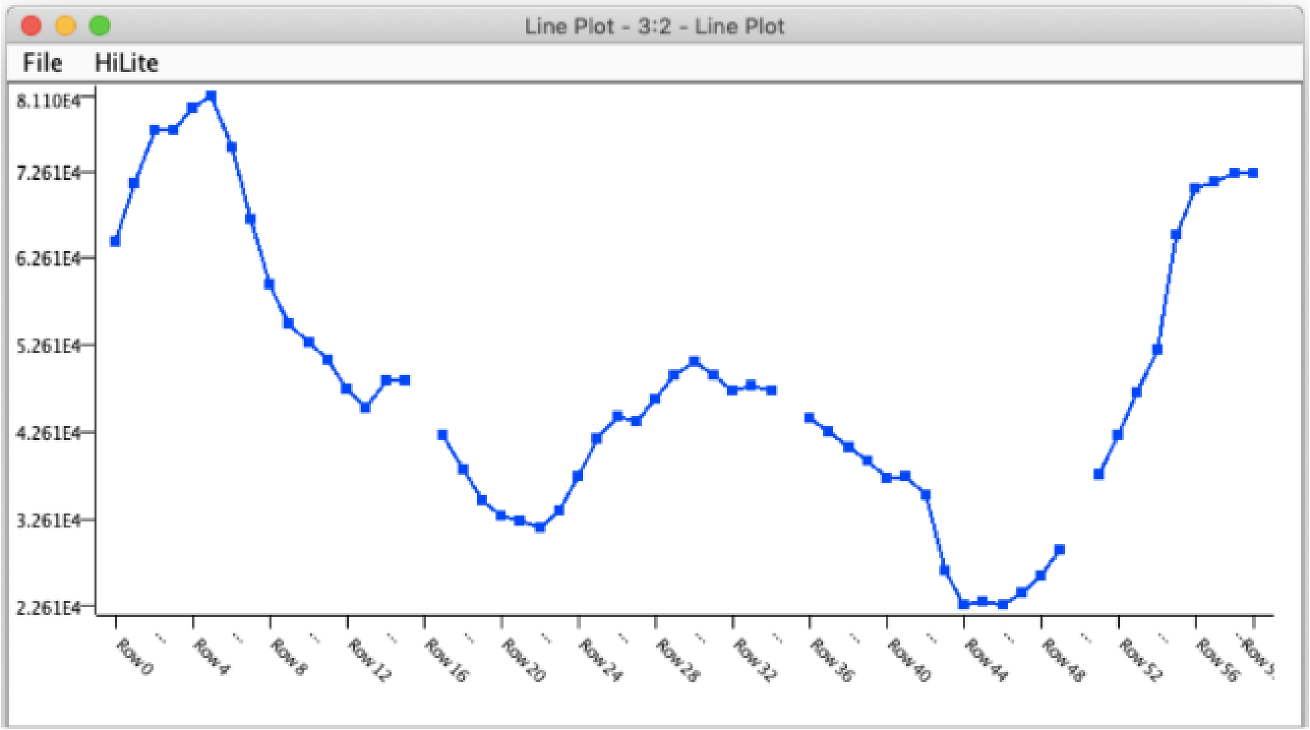
File Hilite Navigation View Output

Row ID	D Col0	D Col1	D Col2	D Col3	S Col4
Row0	5.1	3.5	1.4	0.2	Iris-setosa
Row1	4.9	3	1.4	0.2	Iris-setosa
Row2	4.7	3.2	1.3	0.2	Iris-setosa
Row3	4.6	3.1	1.5	0.2	Iris-setosa
Row4	5	3.6	1.4	0.2	Iris-setosa
Row5	5.4	3.9	1.7	0.4	Iris-setosa
Row6	4.6	3.4	3.774	0.3	Iris-setosa
Row7	5	3.4	1.5	0.2	Iris-setosa
Row8	4.4	2.9	1.4	0.2	Iris-setosa
Row9	4.9	3.1	1.5	0.1	Iris-versicolor
Row10	5.4	3.7	1.5	0.2	Iris-setosa
Row11	4.8	3.4	1.6	0.2	Iris-setosa
Row12	4.8	3	1.4	0.1	Iris-setosa
Row13	4.3	3	1.1	1.214	Iris-setosa
Row14	5.8	4	1.2	1.214	Iris-setosa
Row15	5.7	4.4	1.5	0.4	Iris-setosa
Row16	5.4	3.9	1.3	0.4	Iris-setosa
Row17	5.1	3.5	1.4	0.3	Iris-setosa
Row18	5.844	3.052	1.7	0.3	Iris-setosa
Row19	5.1	3.8	1.5	0.3	Iris-setosa
Row20	5.4	3.4	1.7	0.2	Iris-setosa
Row21	5.1	3.7	1.5	0.4	Iris-setosa
Row22	4.6	3.6	1	0.2	Iris-setosa

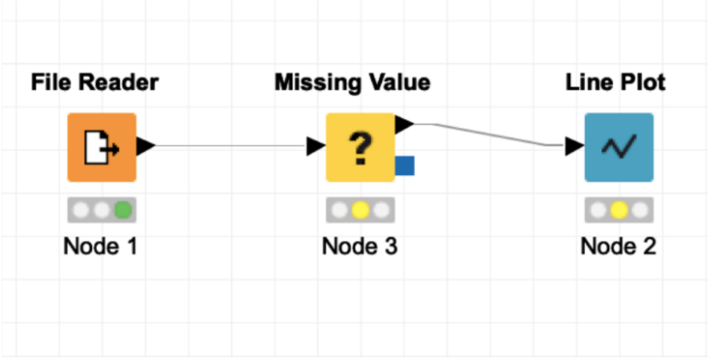
Internet traffic Data (Time Series Data)

- Order data:
  - internet traffic volumes in the UK academic network backbone collected hourly.



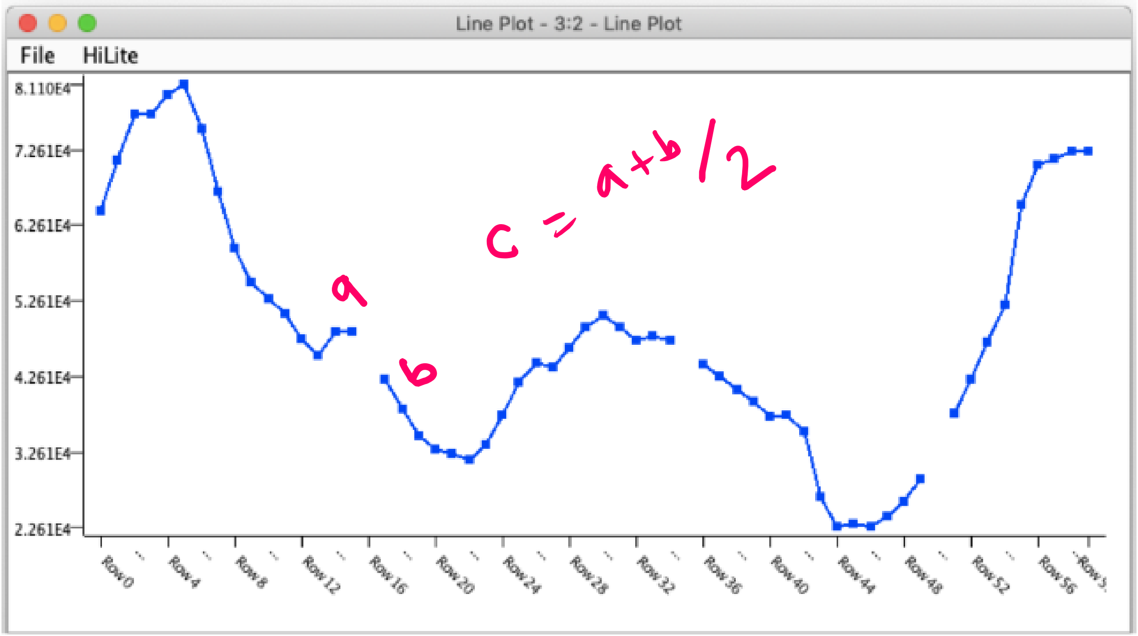


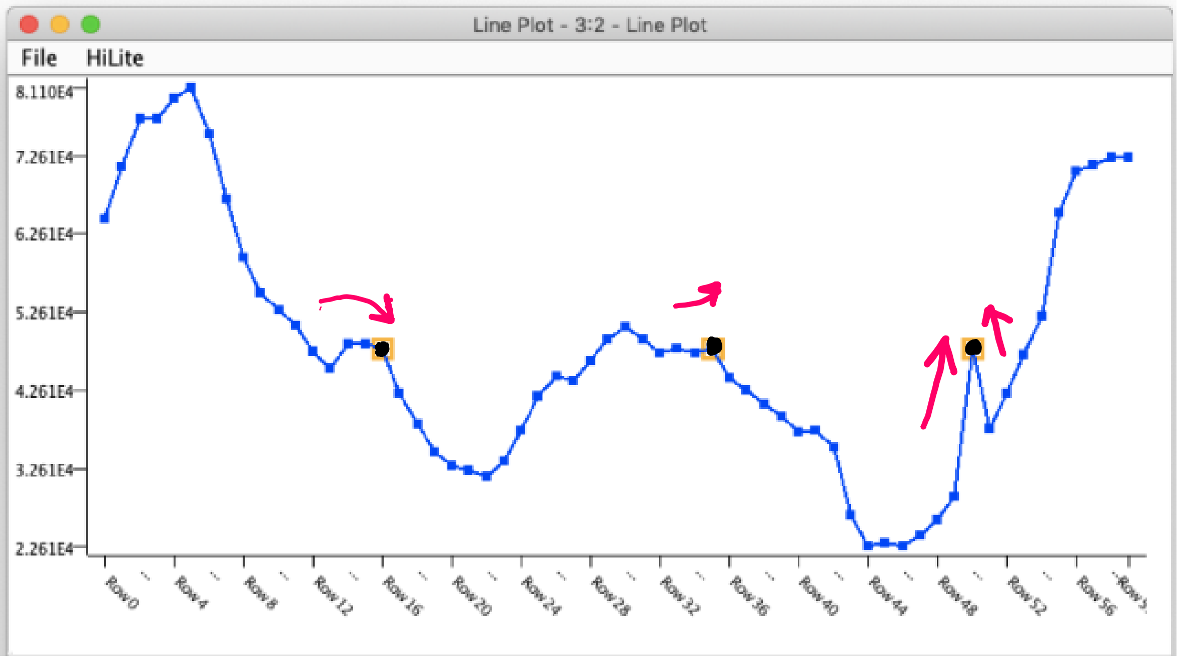
Mean value



missing data

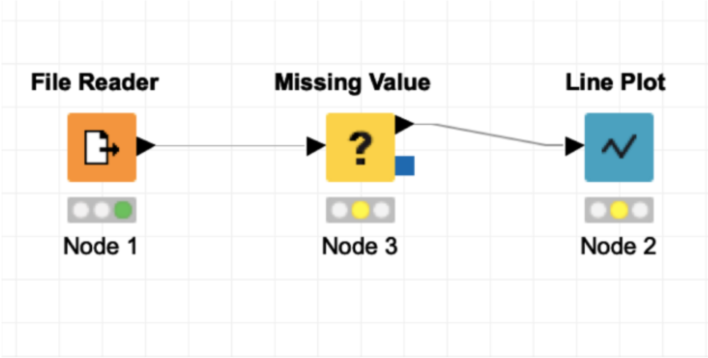
mean to impute data



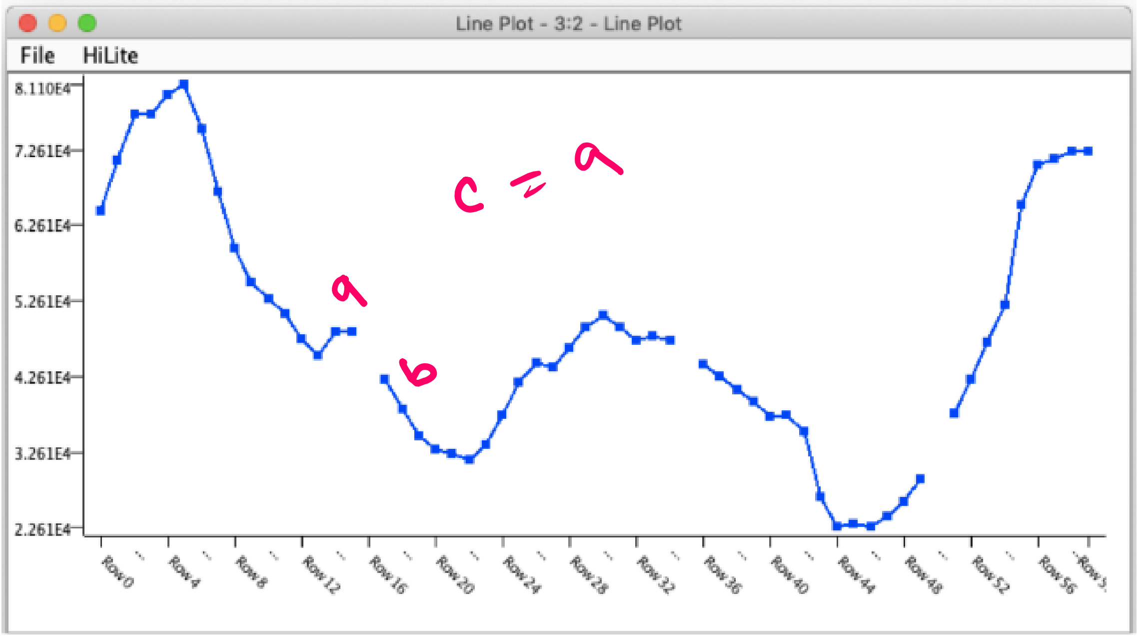


Carry over value

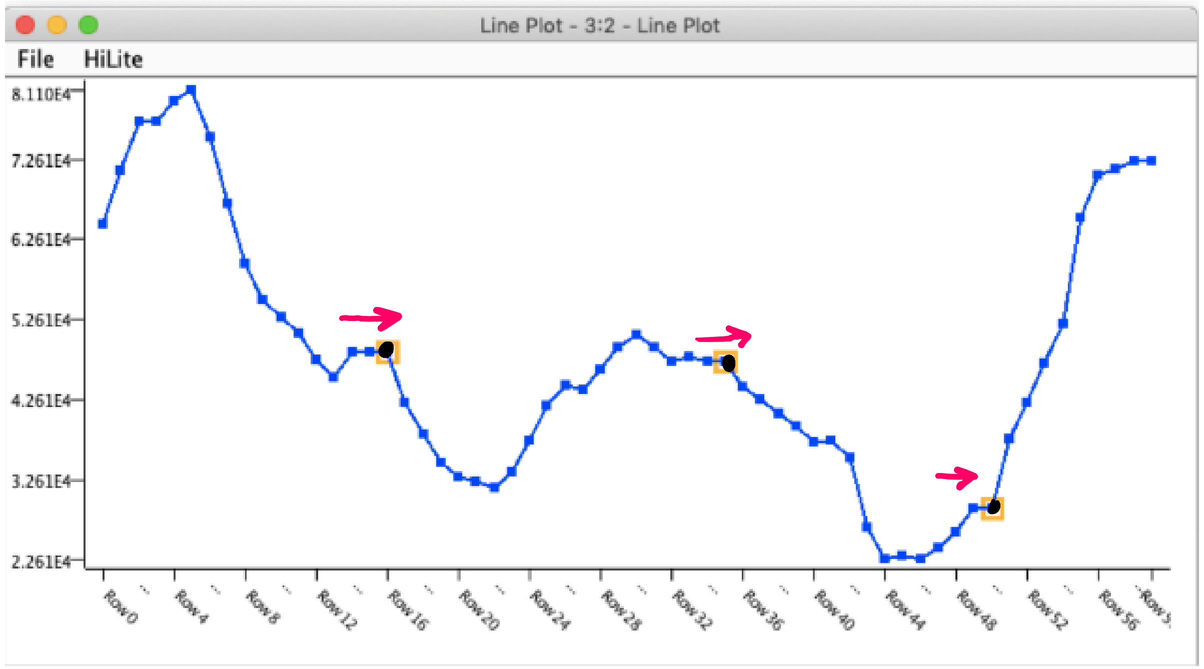
(only makes sense for such ordered data)



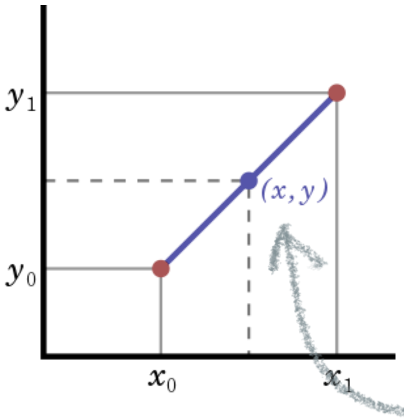
missing data

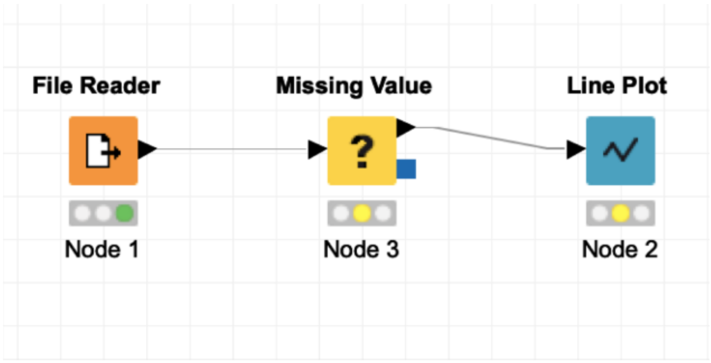


(carry over) the **previous value to impute**



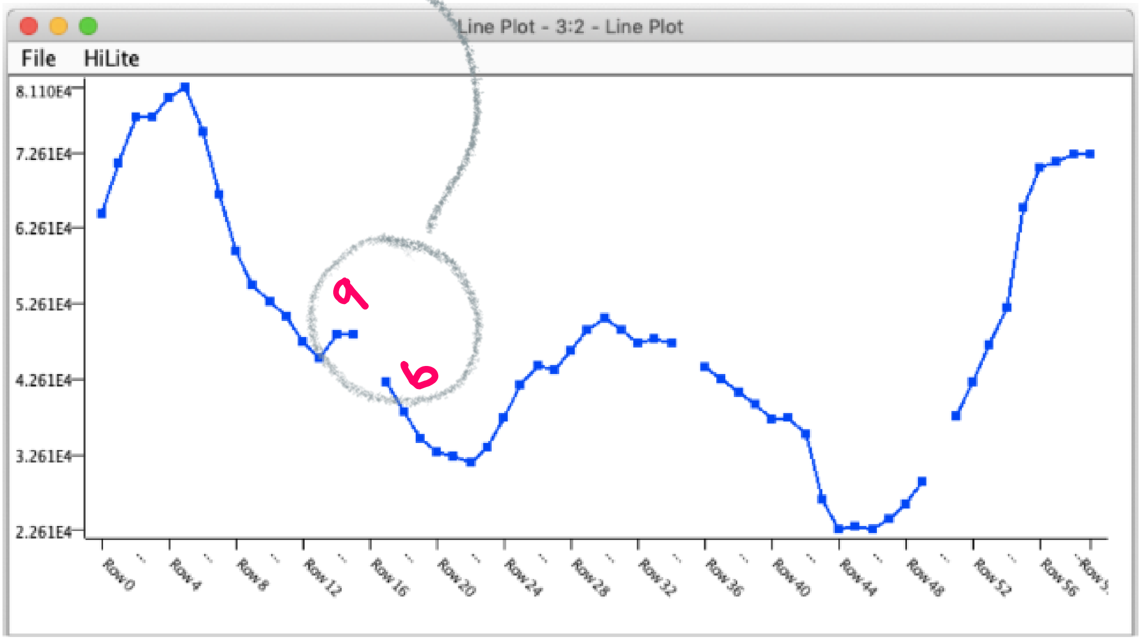
Interpolated value

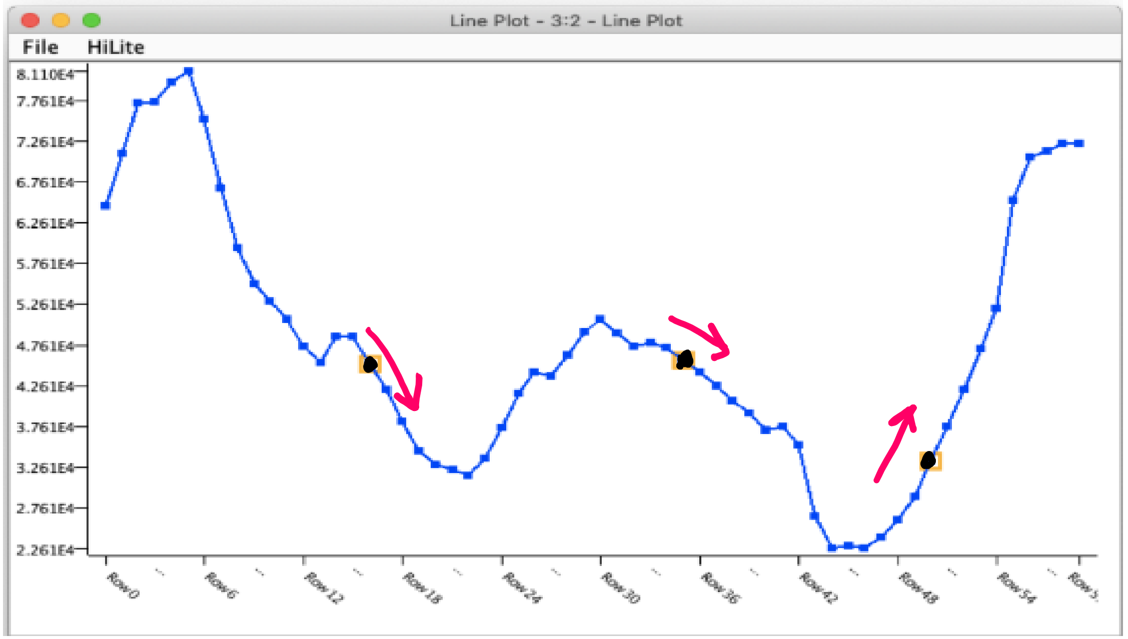




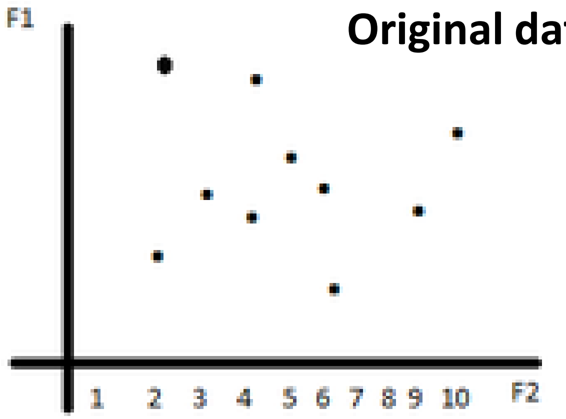
missing data

linear interpolation to impute

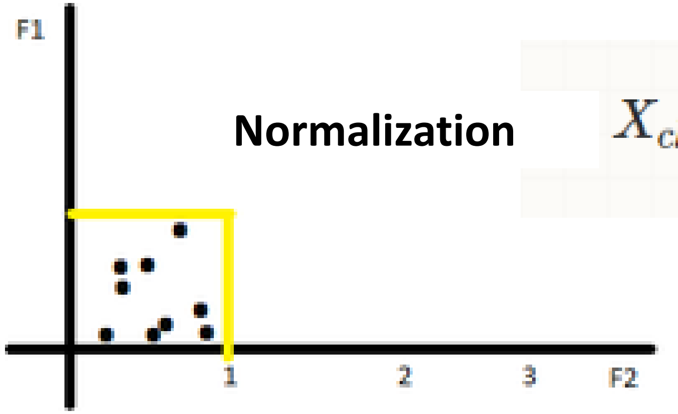




Attribute Transformation

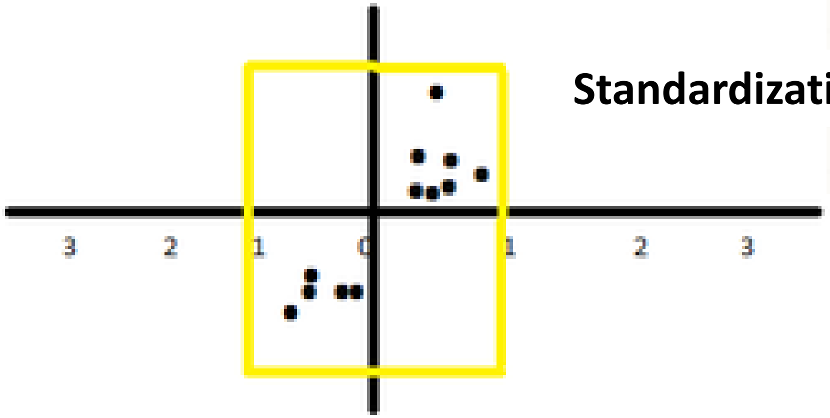


Original data



Normalization

$$X_{changed} = \frac{X - X_{min}}{X_{max} - X_{min}}$$



Standardization

$$X_{changed} = \frac{X - \mu}{\sigma}$$

Discretization and Binarization

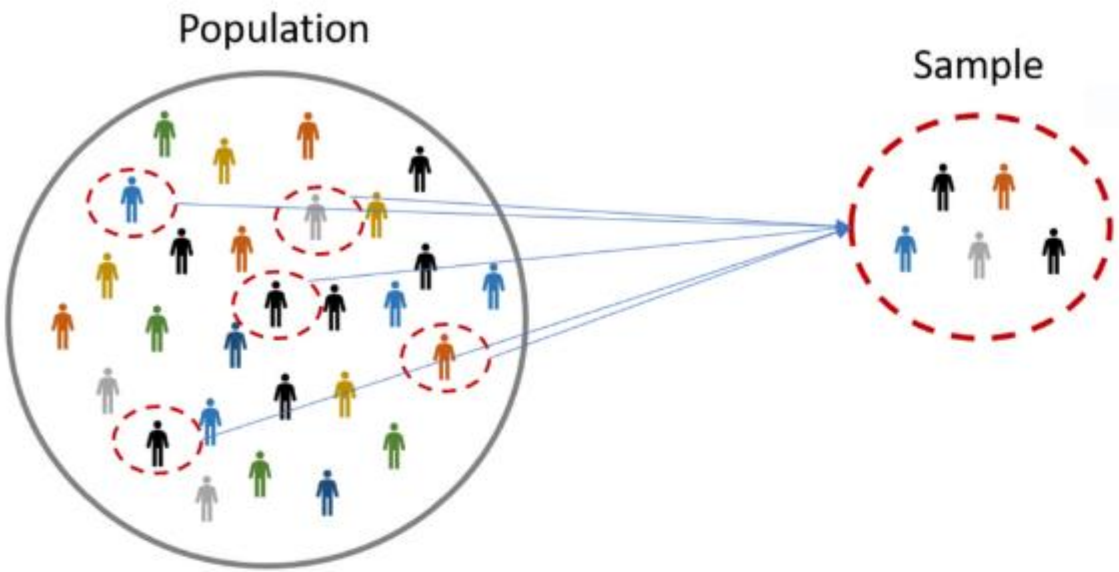
Attribute (e.g., age)	Discretization (age group)	Binarization (can vote?)
10	Kid	0
12	Kid	0
15	Young	0
20	Young	1
55	Old	1
60	Old	1

Sampling

- Sampling is used for **data selection**
- Statisticians sample because **obtaining the entire set of data of interest is too expensive or time-consuming**
- Sampling is used in data mining because **processing the entire set of data of interest is too expensive or time-consuming**

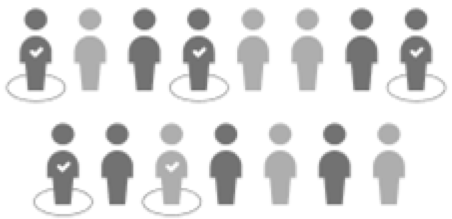
Sampling: Principles

- **Sampling assumes** that using a sample will work almost as well as using the entire data sets if the sample is representative
- A sample is **representative if it has approximately the same property** (of interest) as the original set

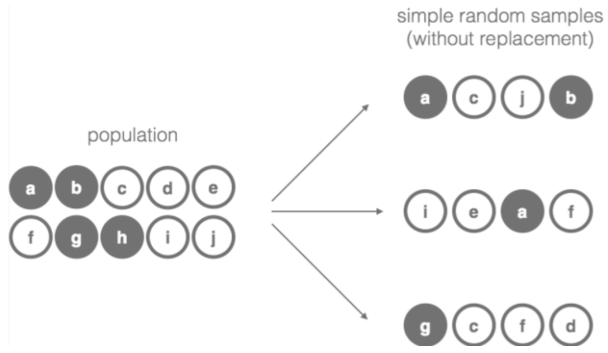


Types of Sampling

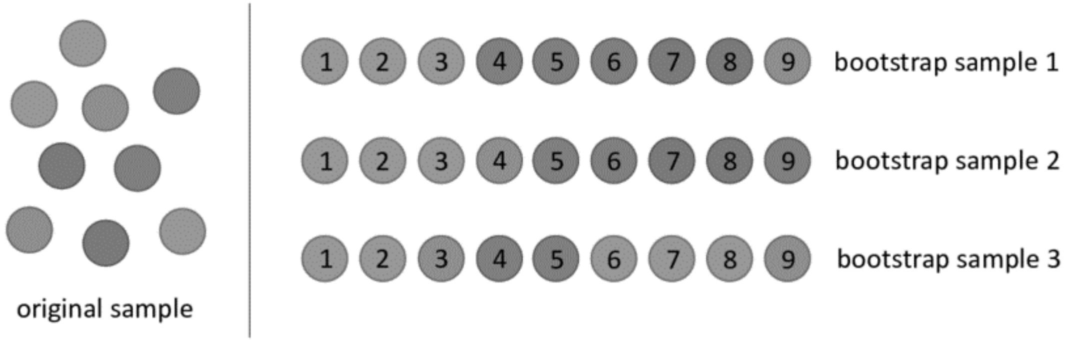
- Simple Random Sampling



- Sampling without replacement

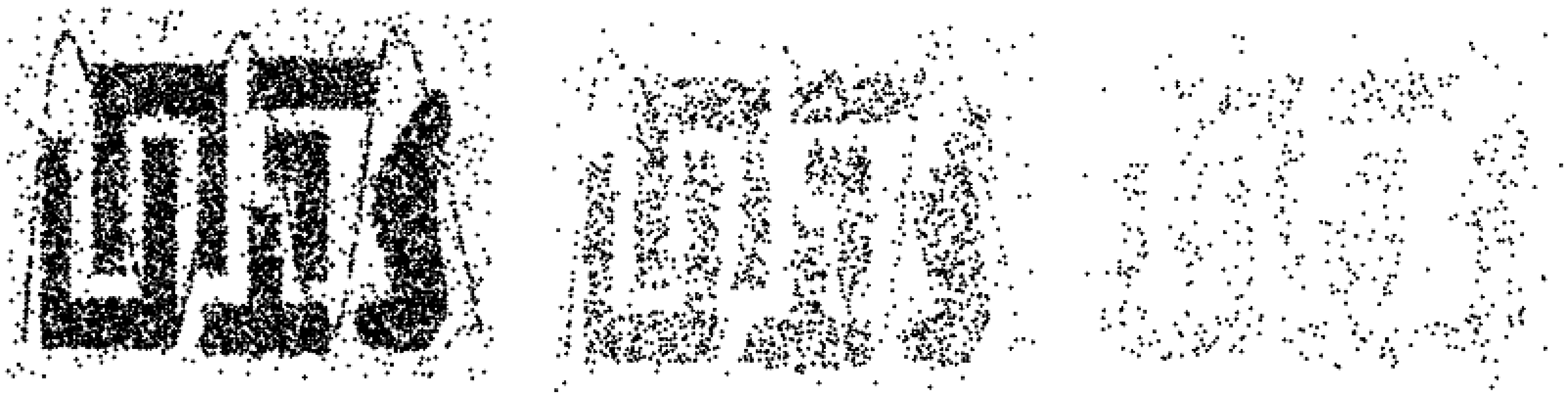


- Sampling with replacement

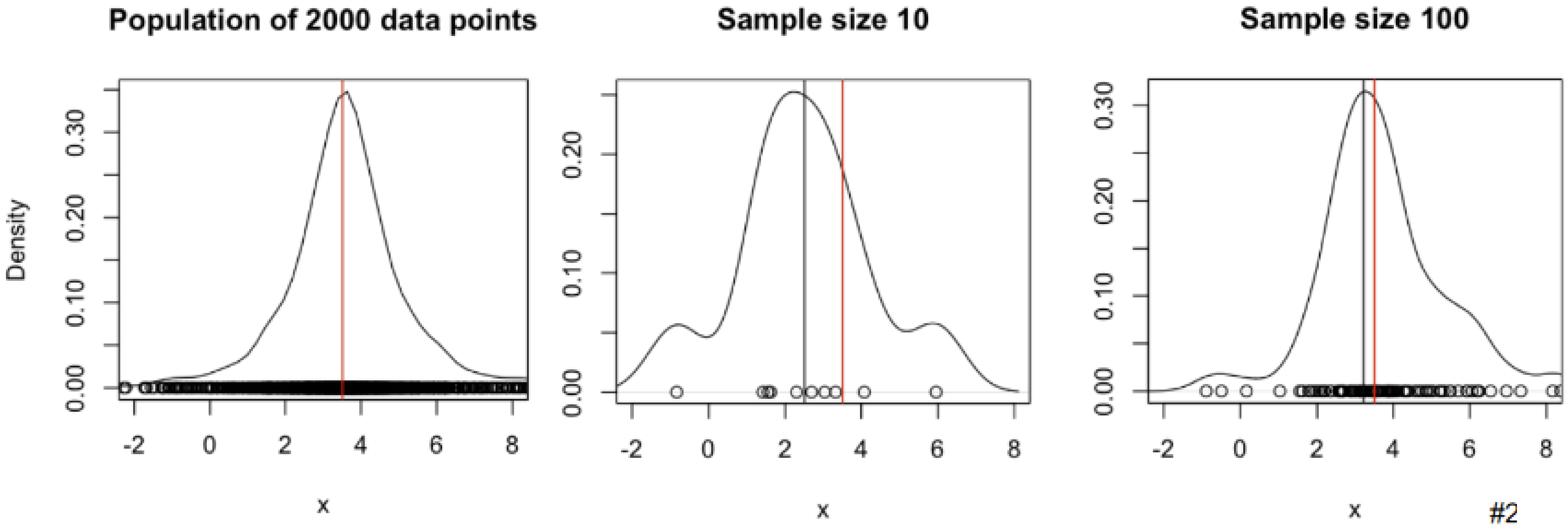


- Stratified sampling

The Effect of Sample Size



8000 points 2000 Points 500 Points



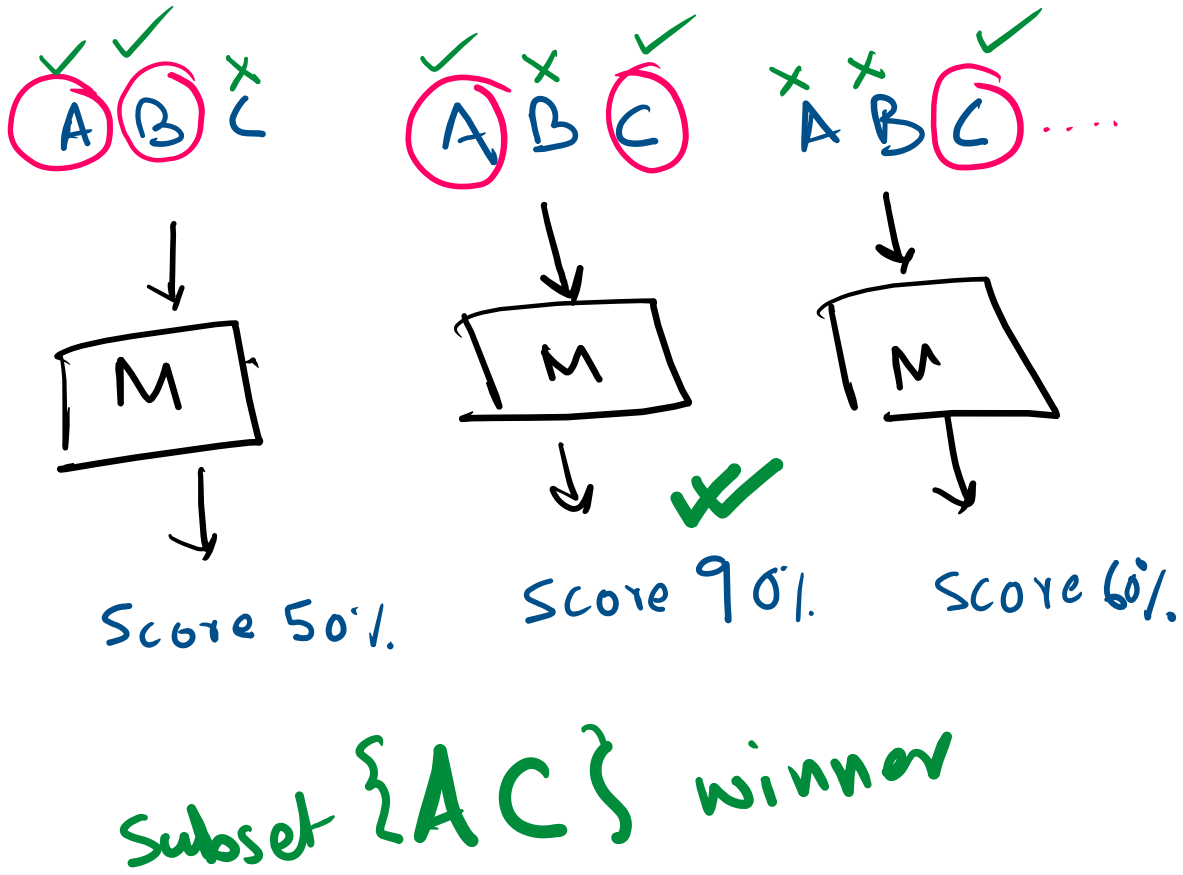
#2

Feature Subset Selection (**Benefits**)

- **Discard Redundant and irrelevant features**
- Speed up Training time
- Improve model interpretability
- Improve model generalisation (by reducing overfitting)

Feature Subset Selection (**Techniques**)

- **Brute-force approach:**
  - Try all possible feature subsets
- **Filter approaches:**
  - Features are selected before the run
- **Wrapper approaches:**
  - Use the data mining algorithm as a black box to find the best subset



Aggregation (Feature Engineering)

- **Combining two or more attributes into a single attribute**
- Combining two or more objects (examples) into a single object

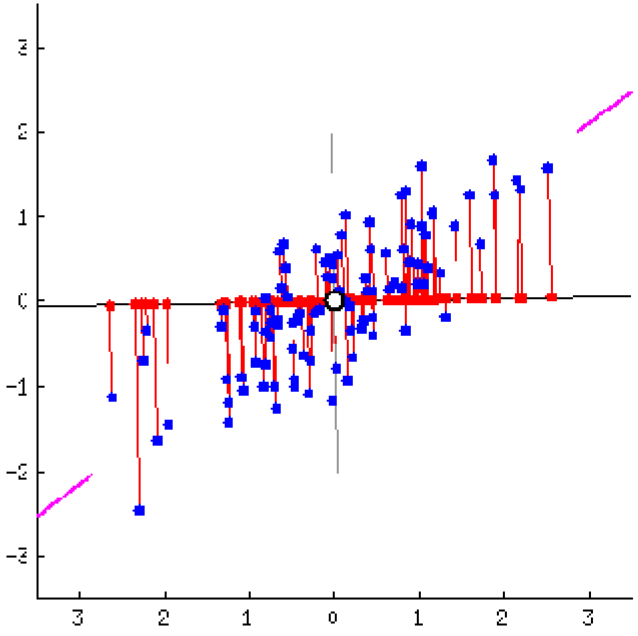
- The purpose of aggregation:
  - Data reduction
    - Reduce the number of attributes or objects
  - Change of scale
    - Cities aggregated into regions, states, countries, etc
  - More “stable” data
    - Aggregated data tends to have less variability

Dimensionality Reduction Principles

- Purpose:
  - **Reduce the amount of time and memory required** by data mining algorithms
  - Allow data to be more easily visualized
  - May help to eliminate irrelevant features or reduce noise
- Techniques
  - Principle Component Analysis
  - Singular Value Decomposition
  - Others: supervised and non-linear techniques

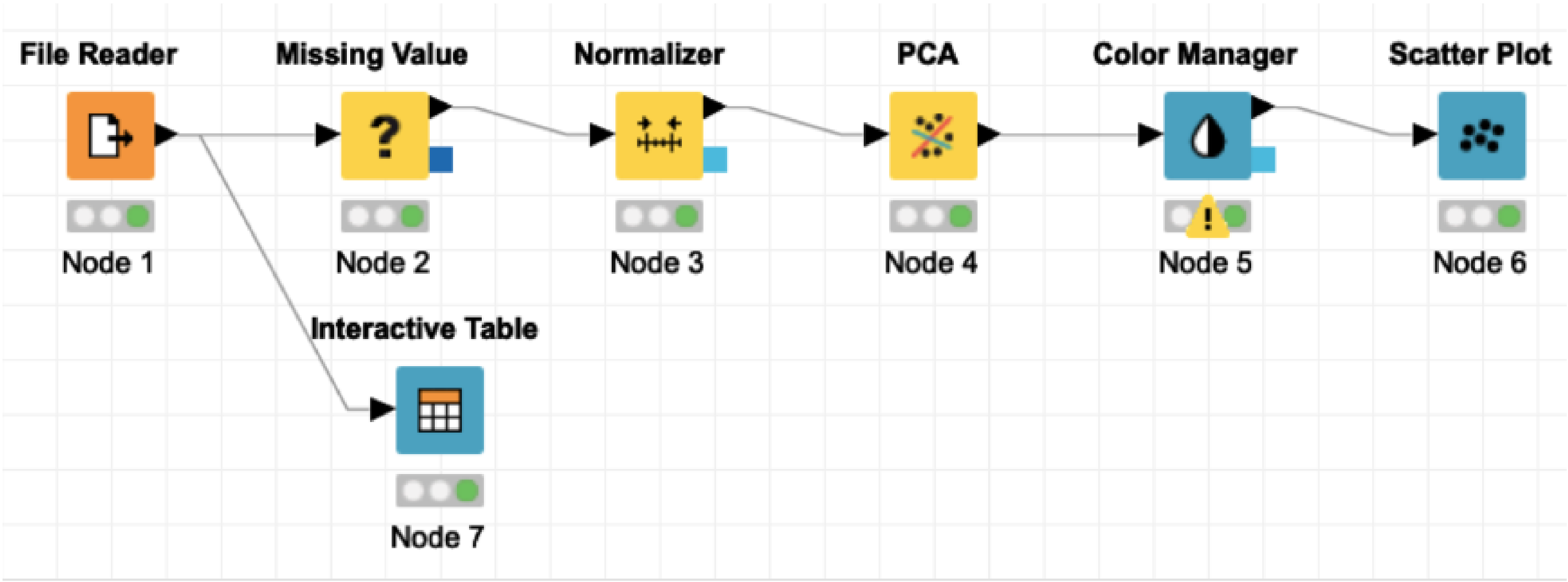
Dimensionality Reduction with PCA

- Principle Component Analysis (PCA):
  - Goal is to find a projection that captures the largest amount of variation in data



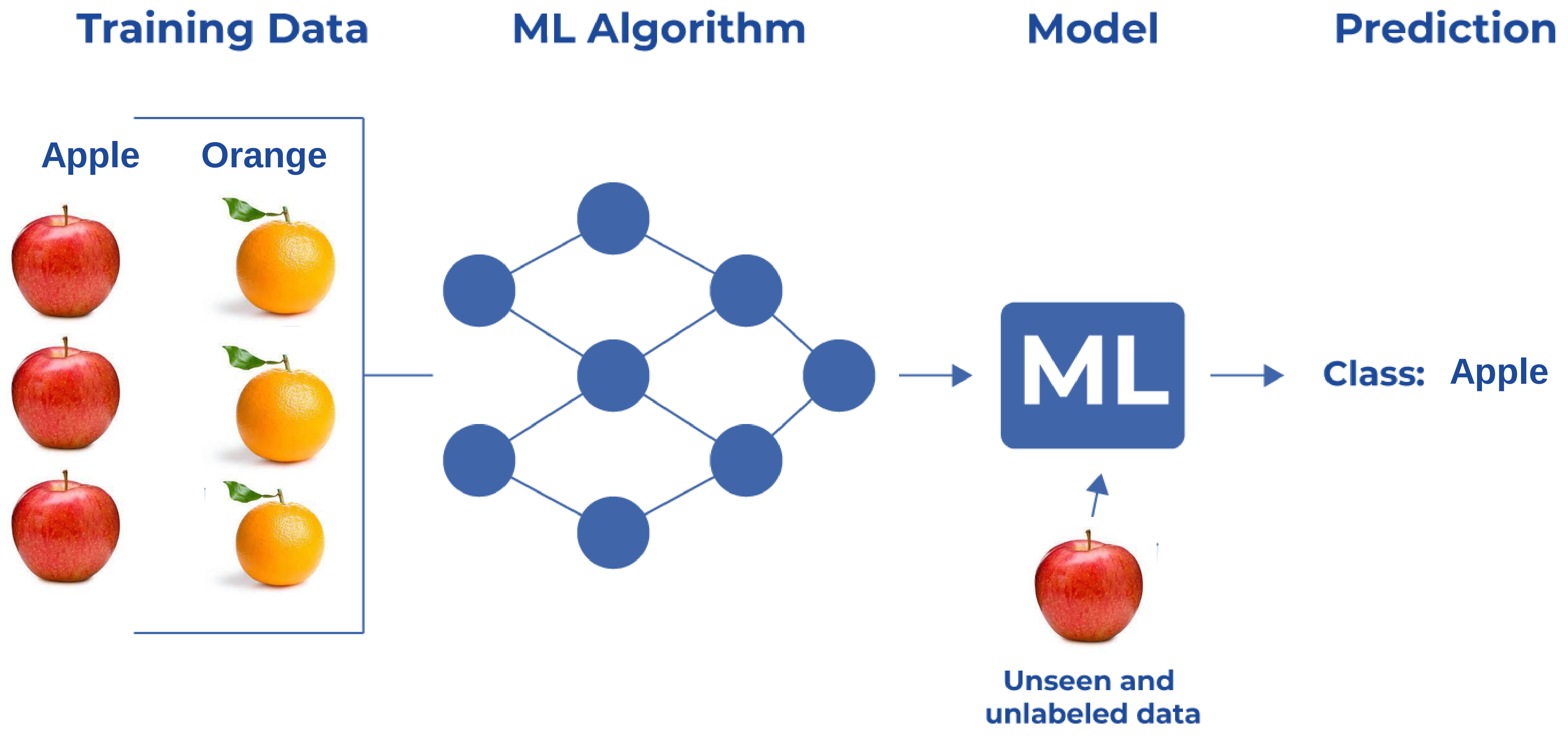
Source: https://www.r-bloggers.com/2017/01/principal-component-analysis-in-r/

Dimensionality Reduction in KNIME



Modelling

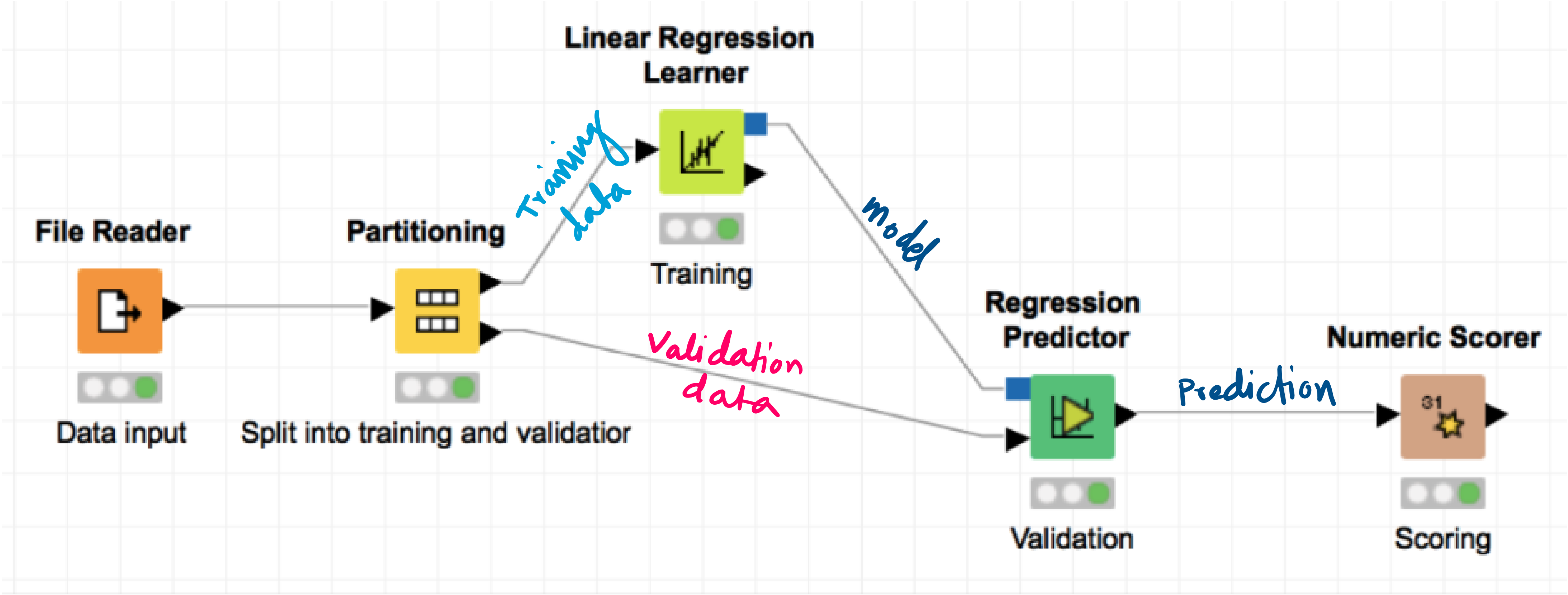
Supervised Learning (Classification)



Holdout Validation

- **Train on a subset of the data**
- Validate on ‘**unseen**’ data
- **Calculate score** of the model

Partitioning data in KNIME

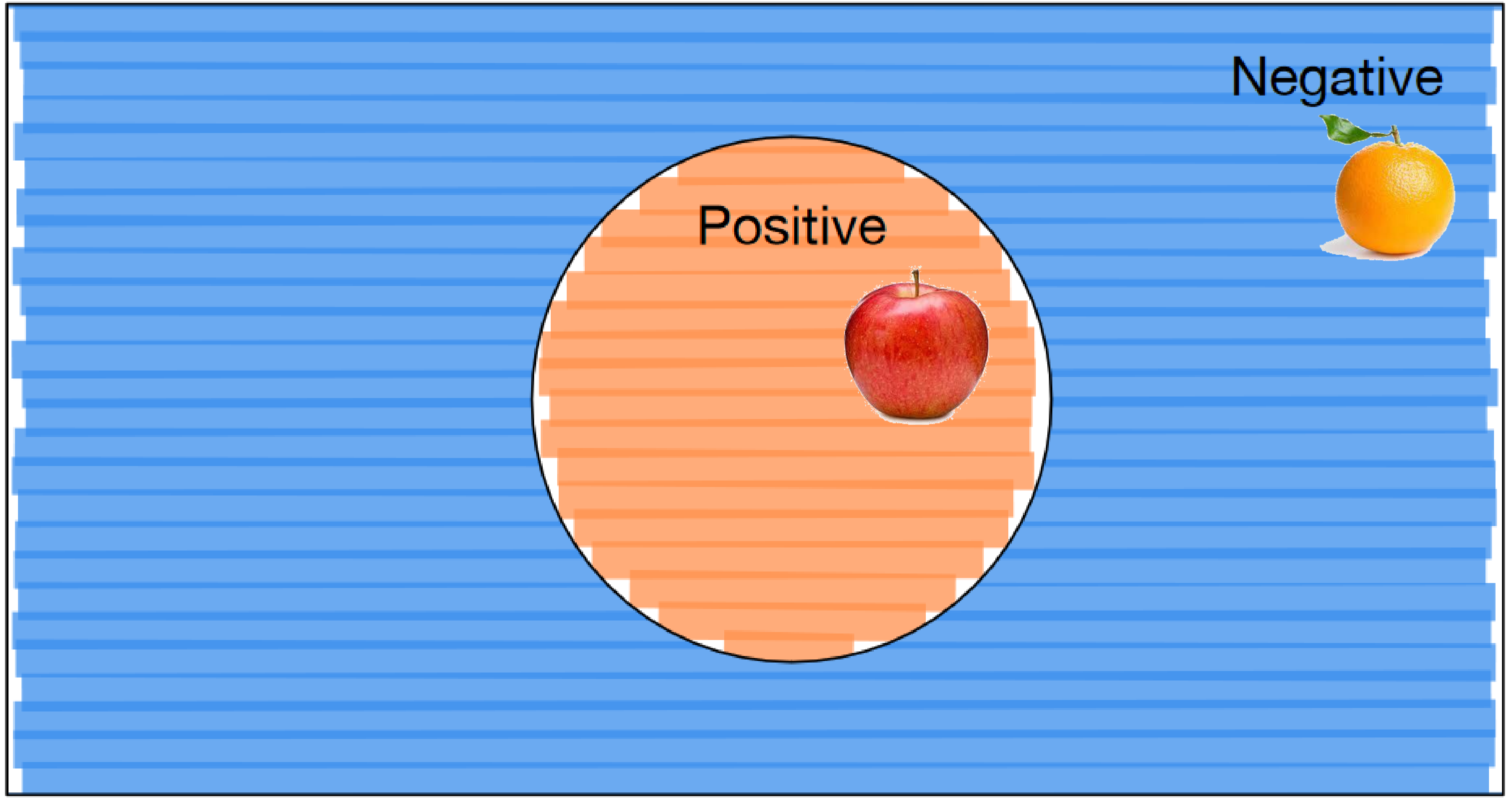


Scoring: Evaluating a Classification Model

- **Classification models are evaluated depending on the correct number of predictions**

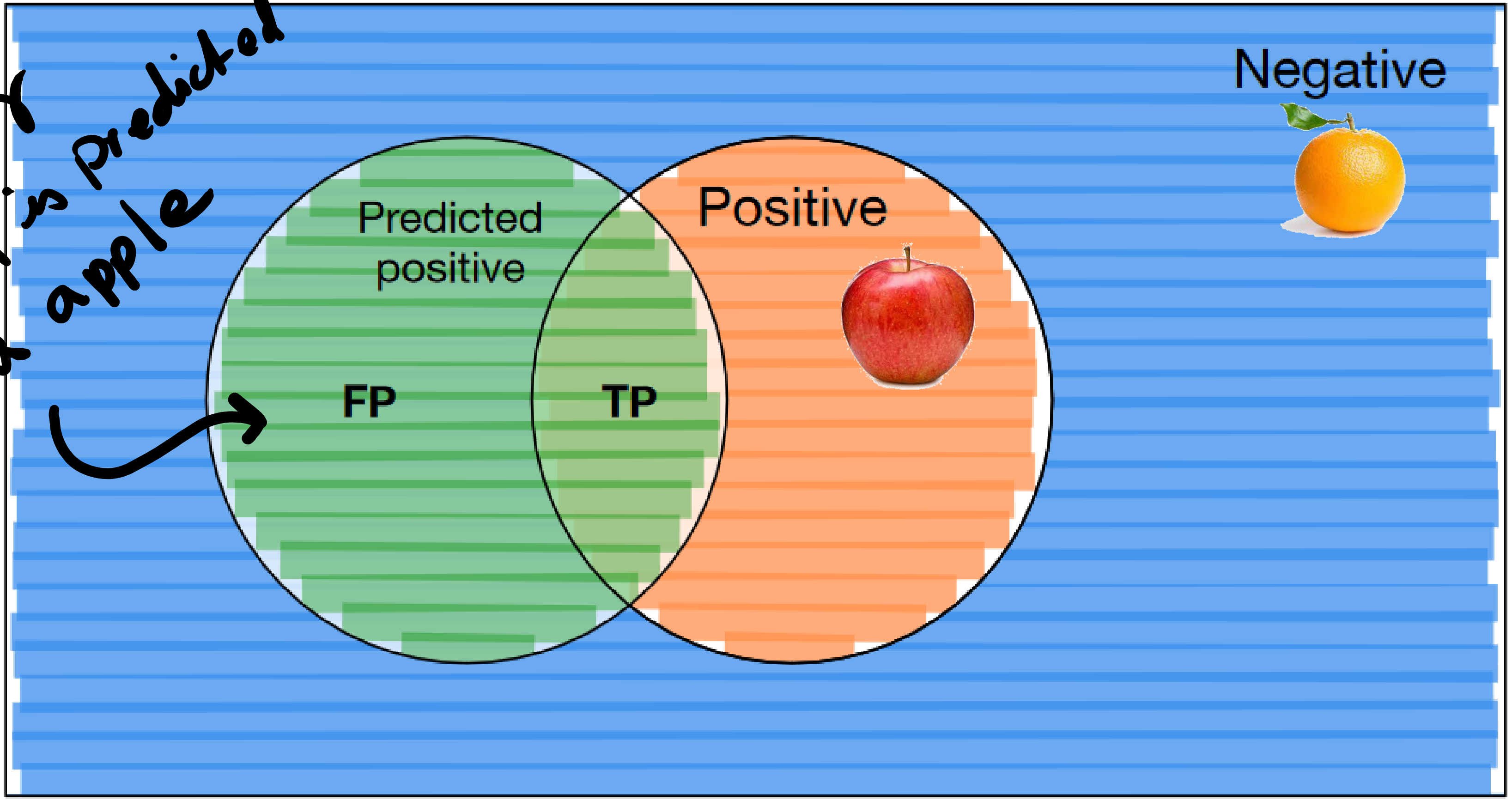
- There are **2 basic categories of classification problems**
  - Binary classification problem
    - – A classification model with only 2 classes
  - Multiclass classification problem
    - – A classification model with more than 2 classes

A Binary Classification Problem

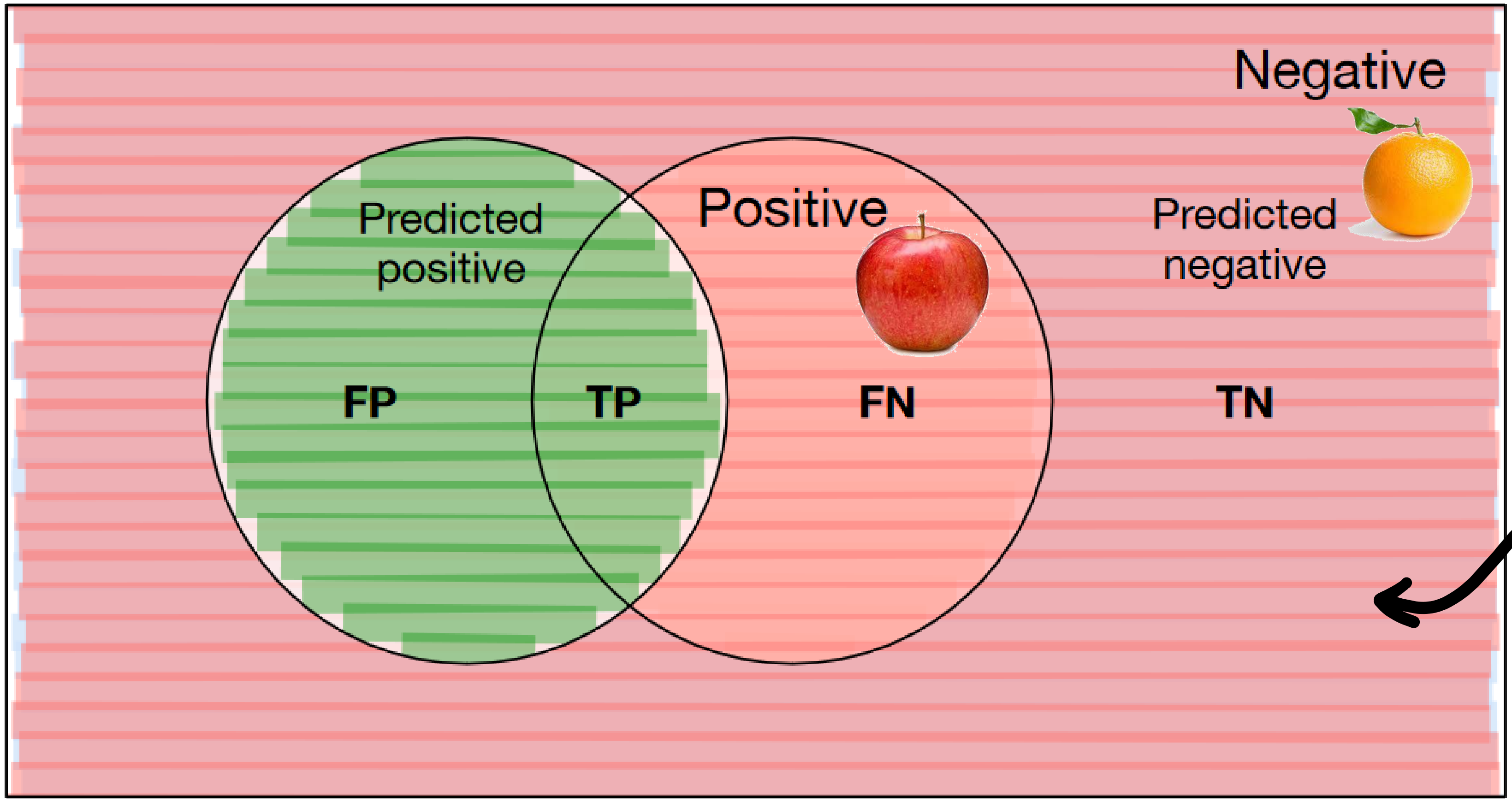


False Positives and True Positives

Everything green is predicted as apple



False Negatives and True Negatives



Everything red is predicted as orange

Scoring (Classification)

- **How many correct predictions were made on the test dataset**
  - The higher percentage is better
- A better way to display results of predictions is as a **confusion matrix**
  - A confusion matrix of actual data vs predicted data
  - Different metrics can be generated according to values in the confusion matrix

The Confusion Matrix

(A matrix of actual data vs predicted data)

	Predicted Class 0	**Predicted Class 1**
Actual Class 0	True Positive	False negative
Actual Class 1	False Positive	True Negative

- From the confusion matrix the total accuracy of the model can be generated:

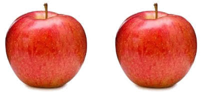
$$\text{Accuracy} = \frac{True\ Positive + True\ Negative}{True\ Positive + True\ Negative + False\ Positve + False\ Negative}$$

- The accuracy states how many predictions were correct

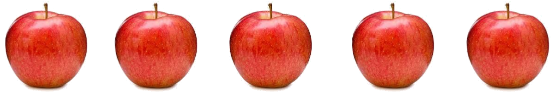
Evaluation Metrics: Precision

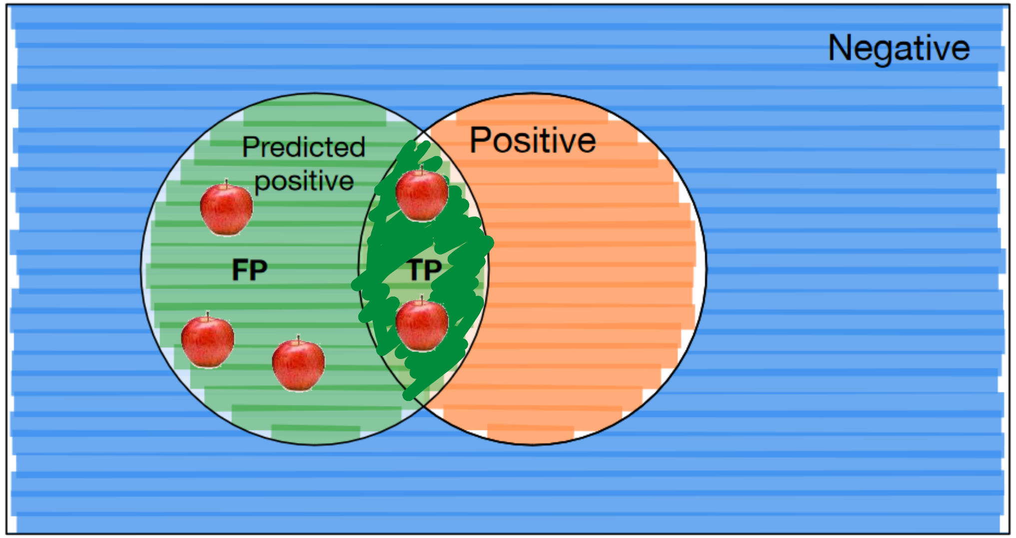
- Precision is the fraction of positive predictions for the respective class that are correct: *How well you guess*

$$\text{Precision} = \frac{True\ Positive}{True\ Positive + False\ Positive}$$



$$= \frac{\text{Total } \textit{apples} \text{ correctly predicted as } \textit{apples}}{\text{Total number of postive prediction in the data}}$$

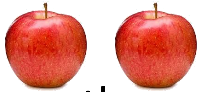




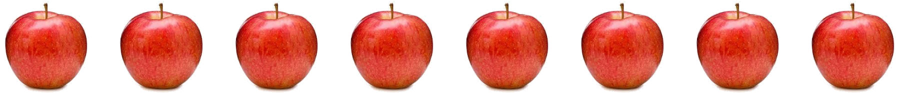
Evaluation Metrics: Recall

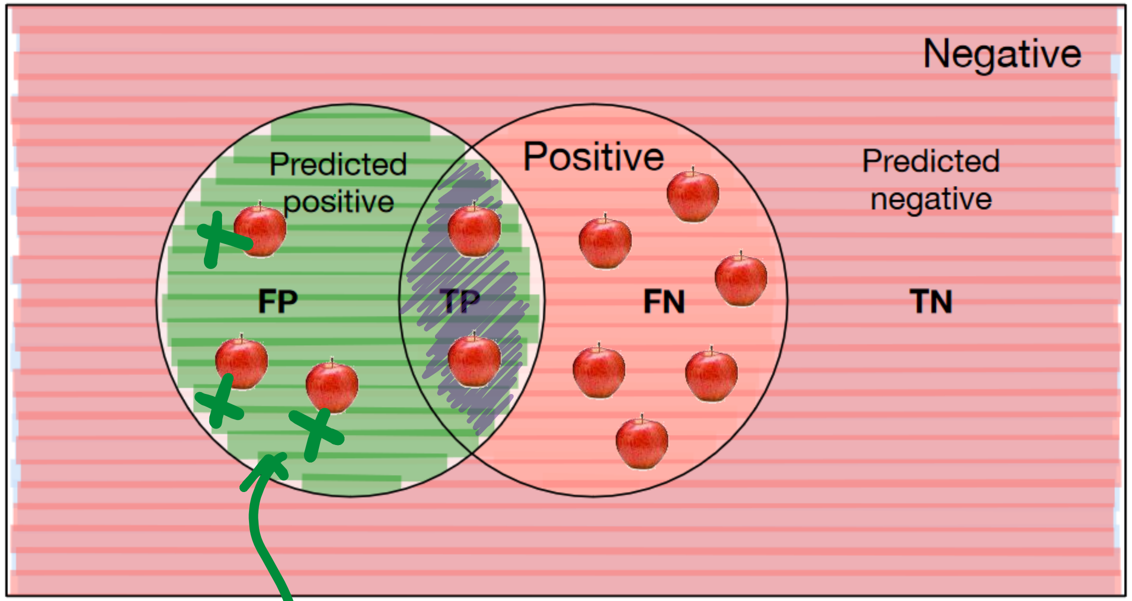
- Recall is the fraction of positive values in the data that we correctly predict: *How complete is the prediction?*

$$\text{Recall} = \frac{\text{True Positive}}{\text{True Positive} + \text{False Negative}}$$



$$= \frac{\text{Total } \textit{apples} \text{ correctly predicted as } \textit{apples}}{\text{Total } \textit{apples} \text{ in the data}}$$





These are wrongly predicted as apples

Trade off between Precision and Recall

- When dealing with real world data you will never see 100% precision and recall at the same time

- For example: **if all the samples are predicted as a positive class in a dataset you will get 100 % recall but a low precision statistic**

- Instead, the F1 score is used to generate a single metric that balances the precision and recall:

$$\text{F1 Score} = 2 * \frac{Precision * Recall}{Precision + Recall}$$

Summary

1. Data Imputation (Data Cleaning)
2. Sampling
3. Feature Engineering (Feature Selection)
4. Dimensionality Reduction
5. Feature Transformation
6. Modelling
7. Validation and Scoring