Introduction to Data Science

Lecture 2; April 4th, 2016

Ernst Henle
ErnstHe@UW.edu
Skype: ernst-henle

Agenda

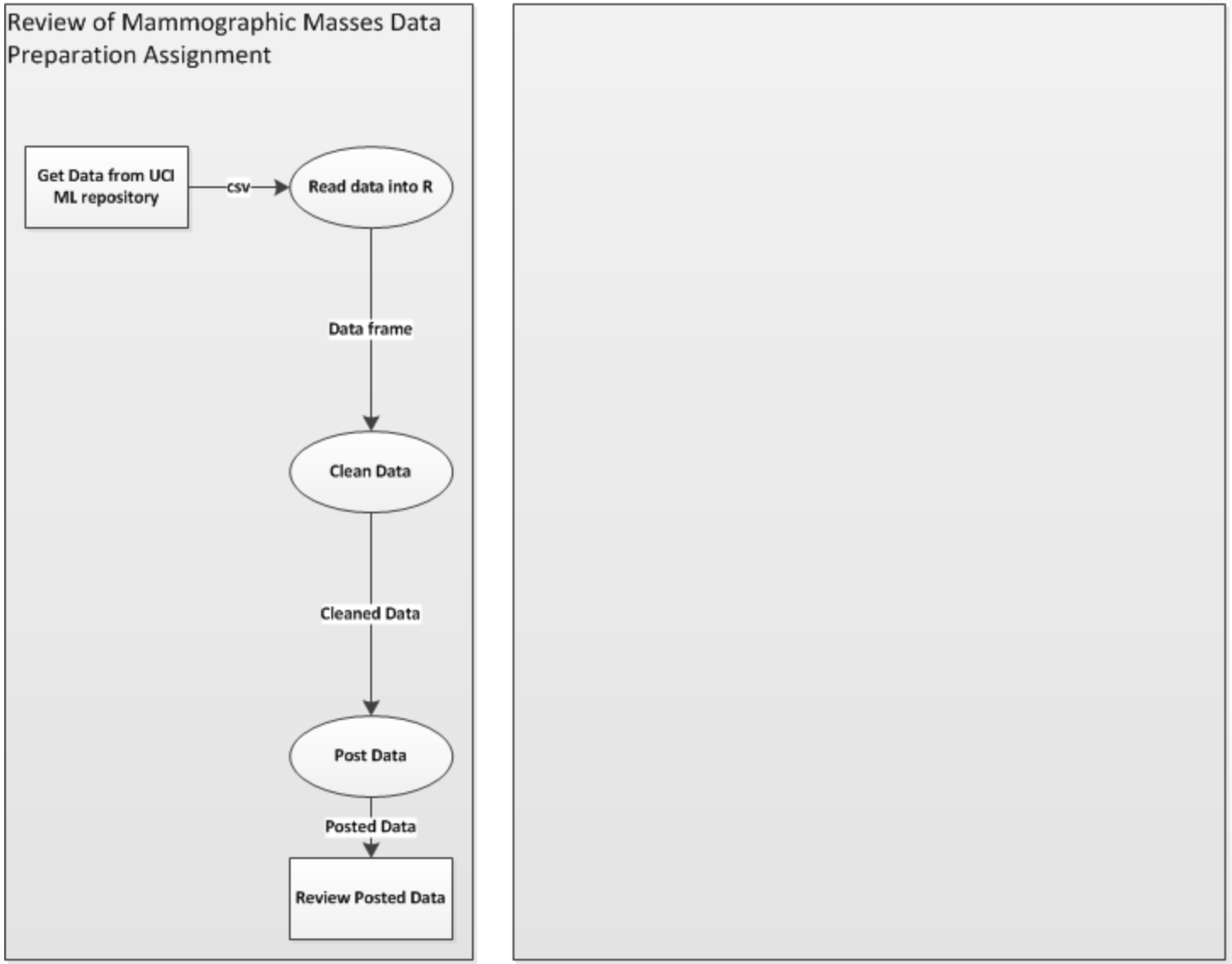
- Announcements
  - LinkedIn Social aspect of this course
  - Proposed Guest Lectures in May
    - Business side of Data Science by Marius Marcu
    - Data Visualization by Tatyana Yakushev
    - Building a Data Science Group by TBD
- Review
  - Optional class on programming in R
  - Class Structure: Canvas Module and preview quiz in preview section
  - Data Preparation DFD
- Quiz 02 (Data Preparation)
- Introduction to K-means Clustering
- Break
- In-Class Exercise and Homework Assignment
- Break
- Dimensions in Clustering
- Normalization (Clustering vs Linear Regression)
- Assignment

Data Preparation Review

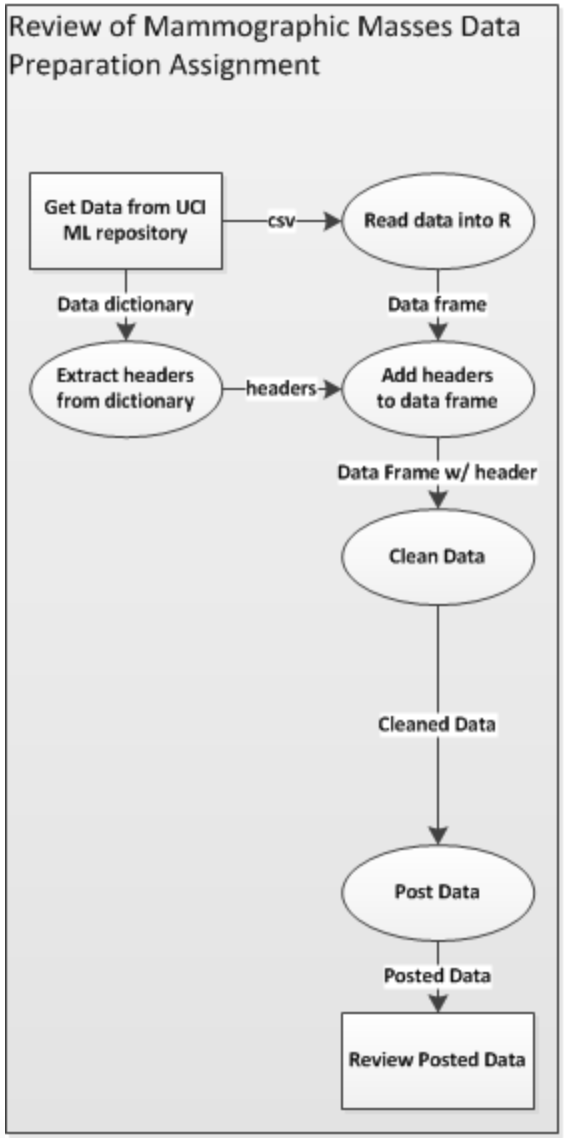
Data Preparation Review (0)

- Find in Canvas: DataScience01Homework.R

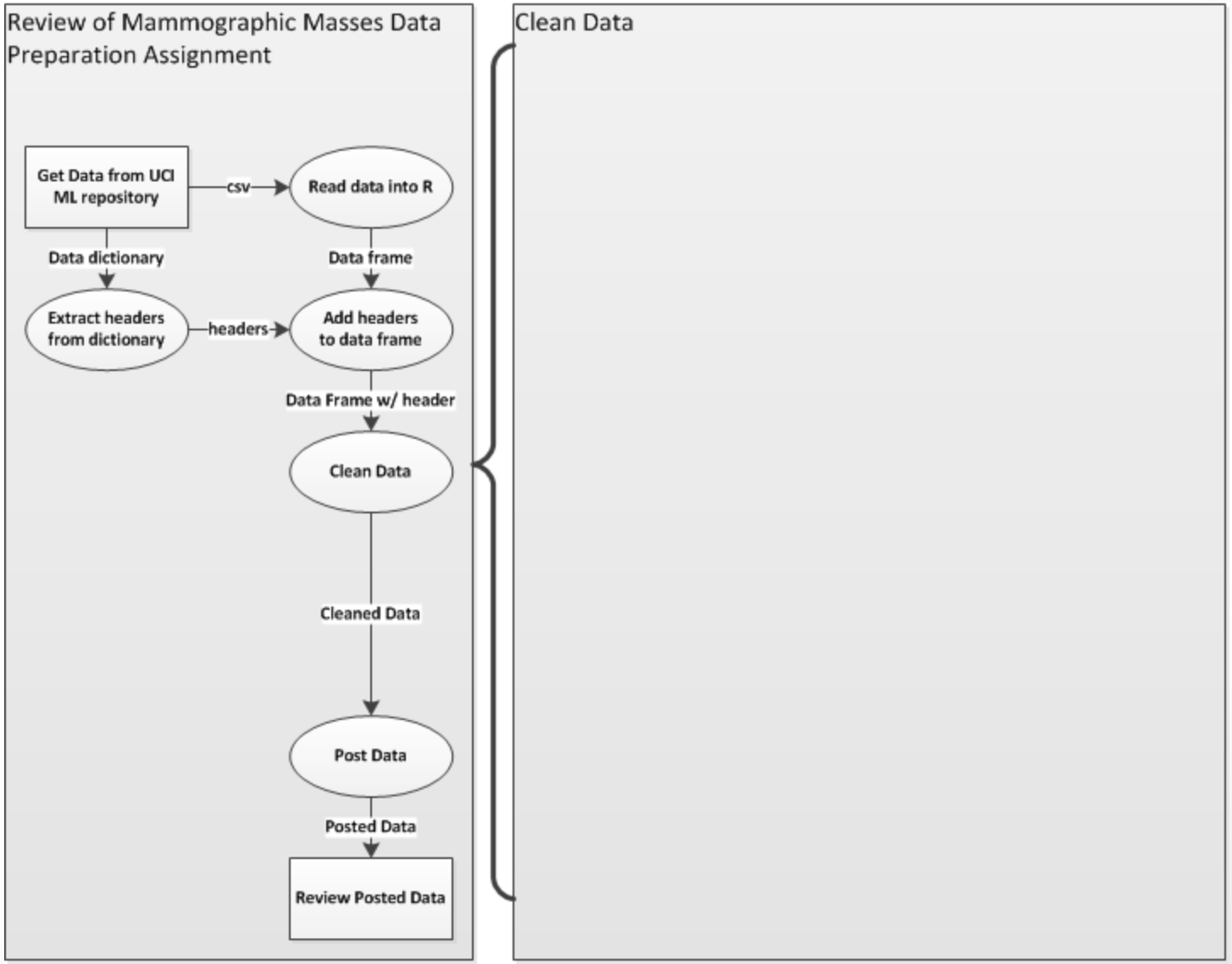
Data Preparation Review (1)



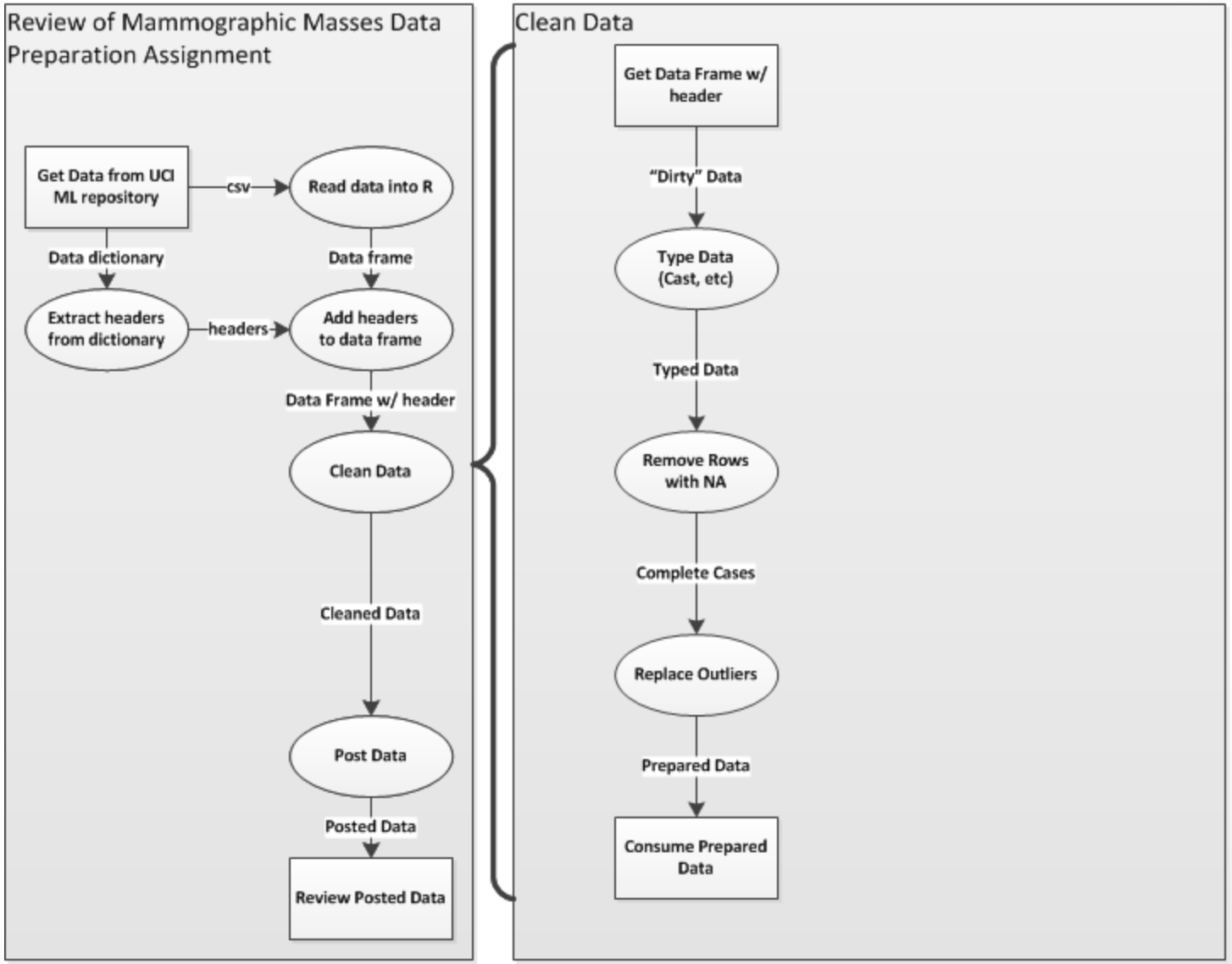
Data Preparation Review (2)



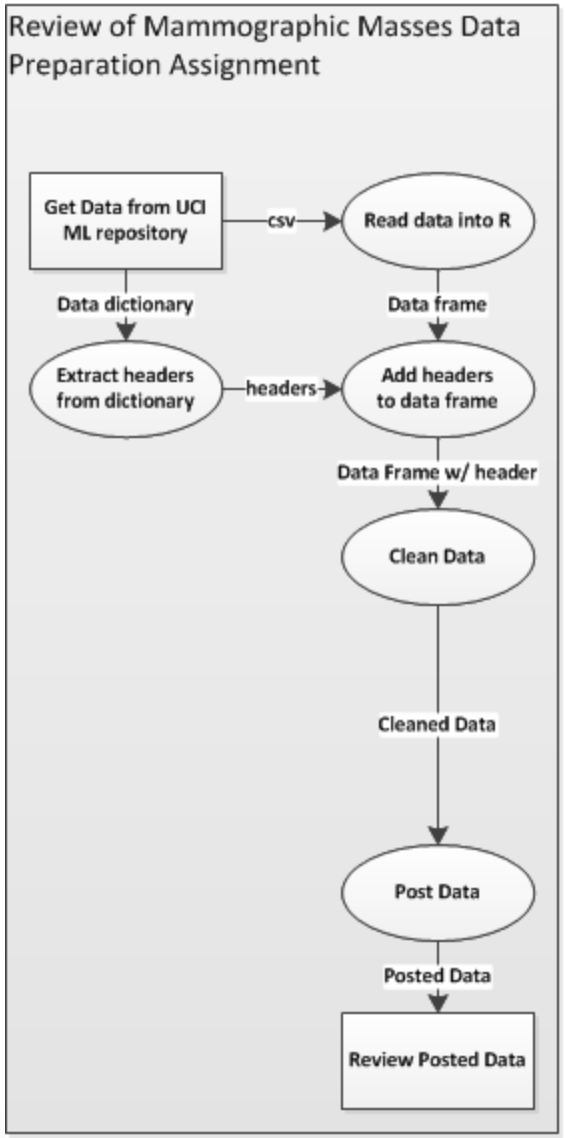
Data Preparation Review (3)

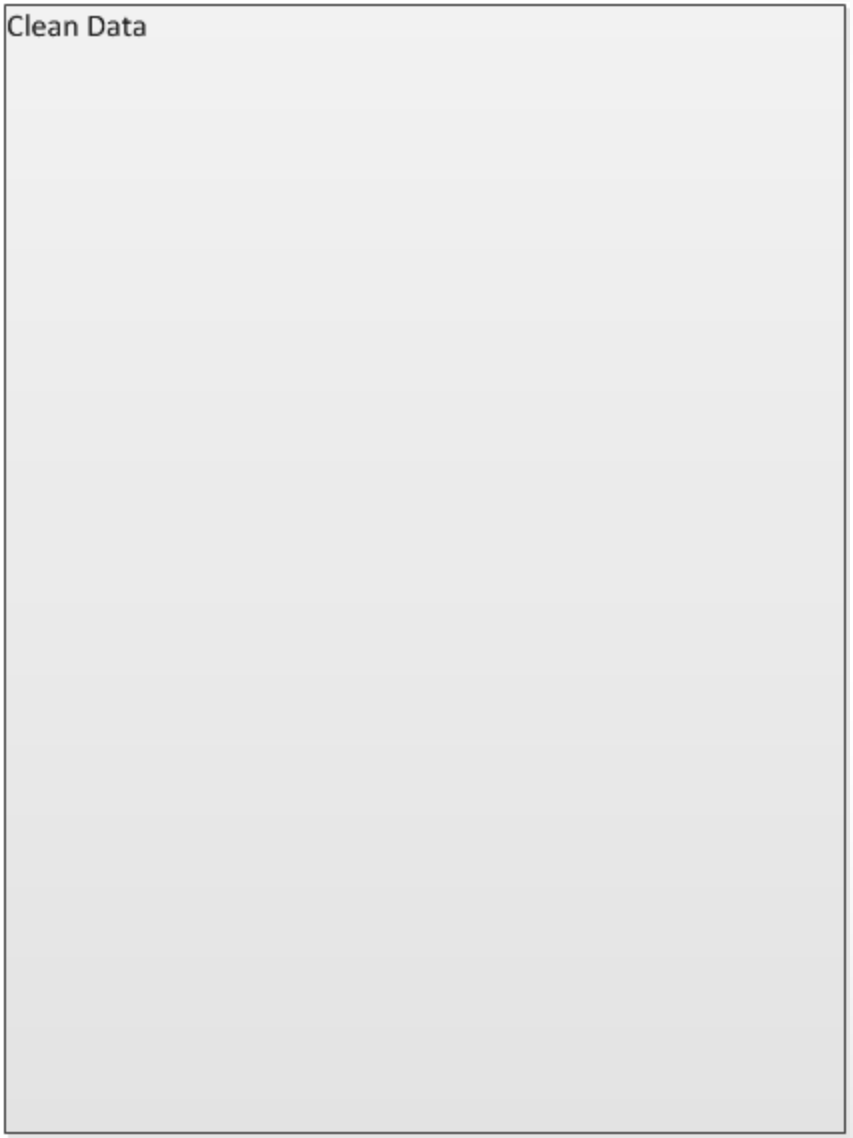


Data Preparation Review (4)

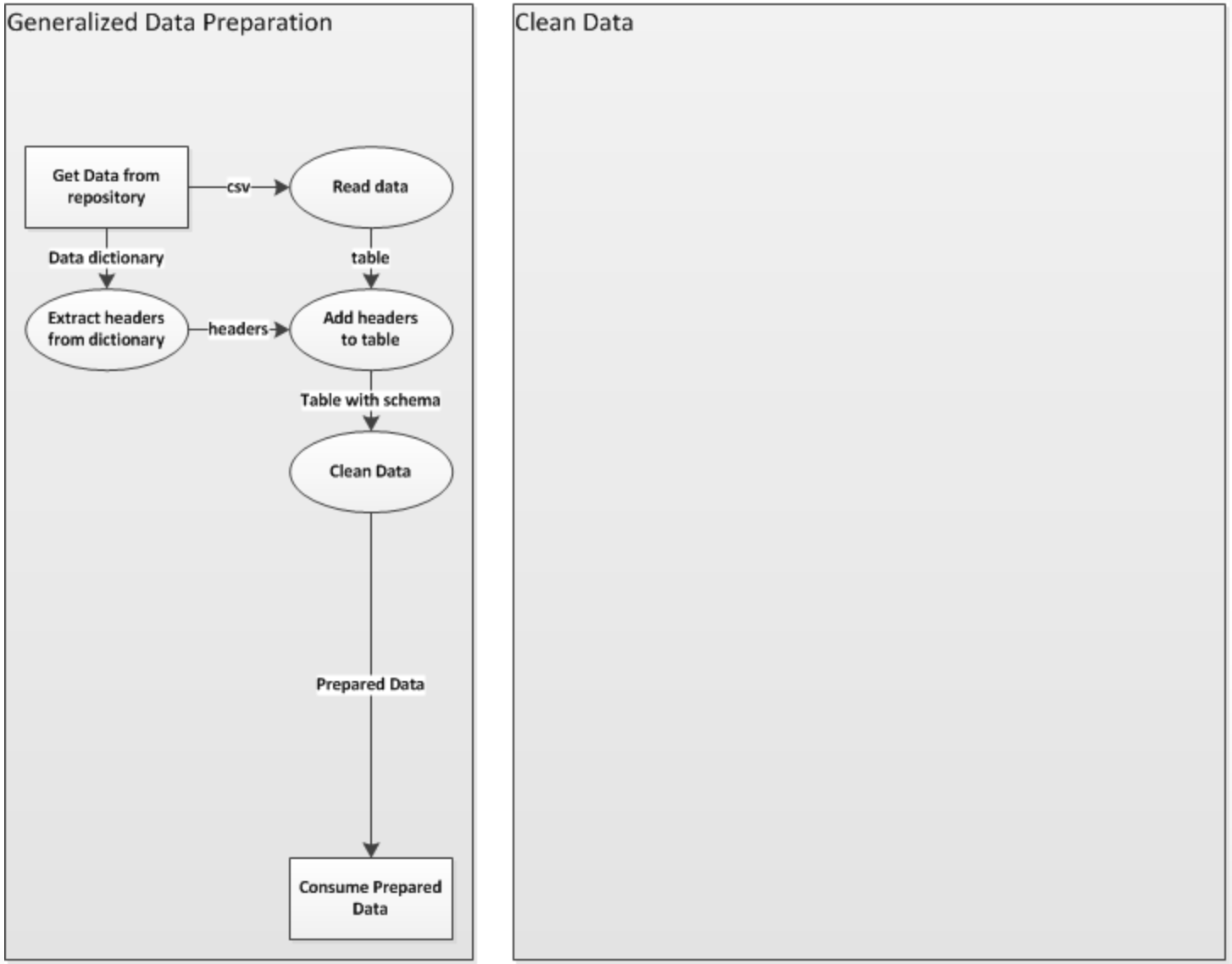


Data Preparation Review (5)

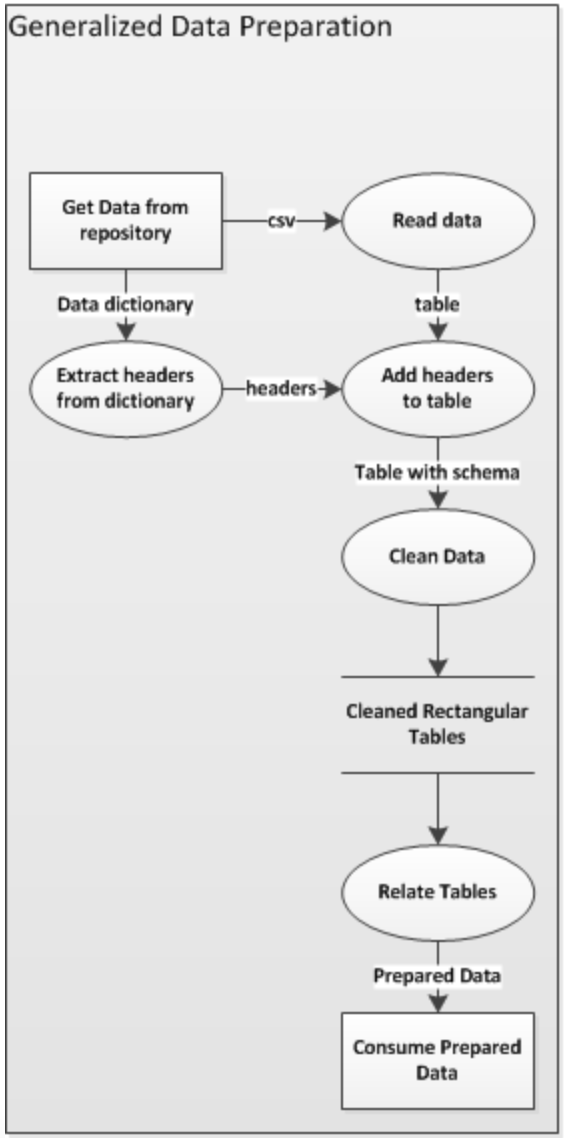


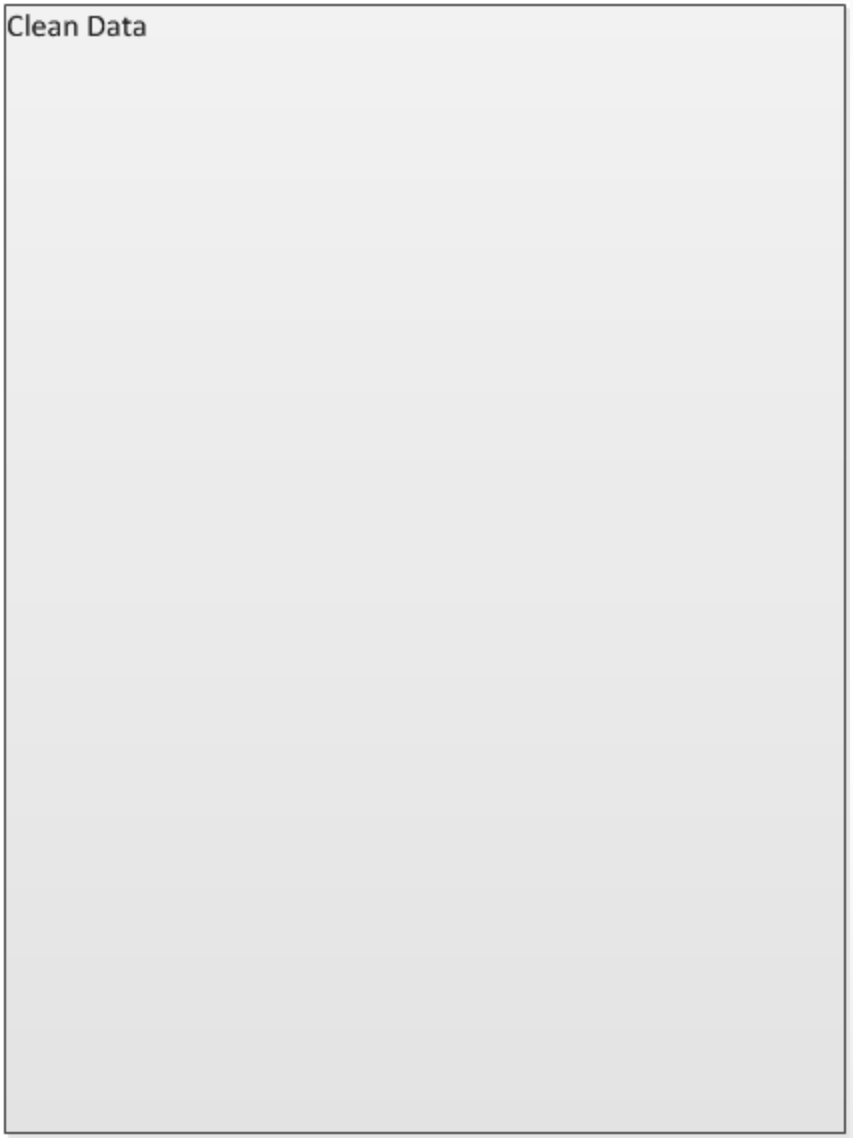


Data Preparation Review (6)

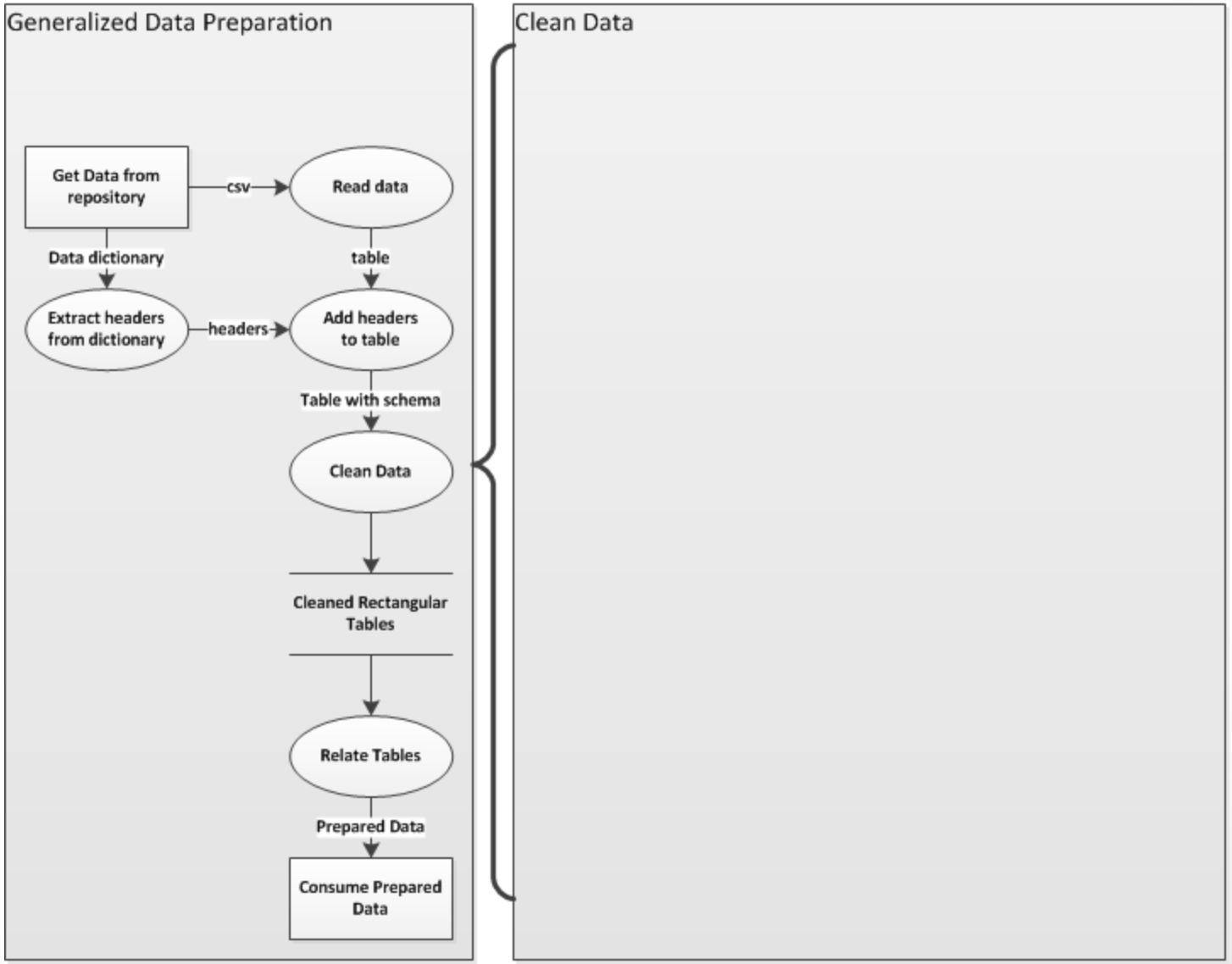


Data Preparation Review (7)

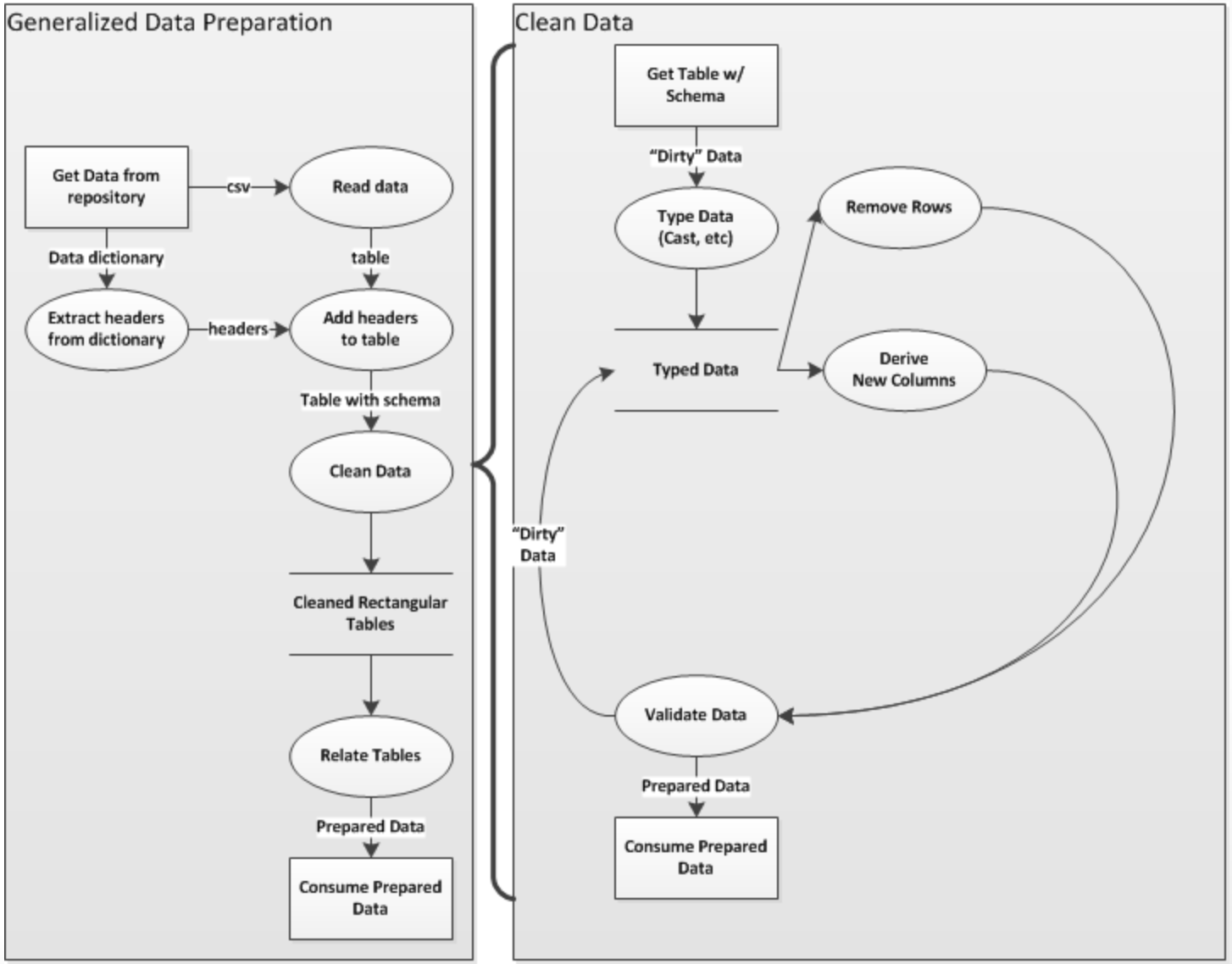




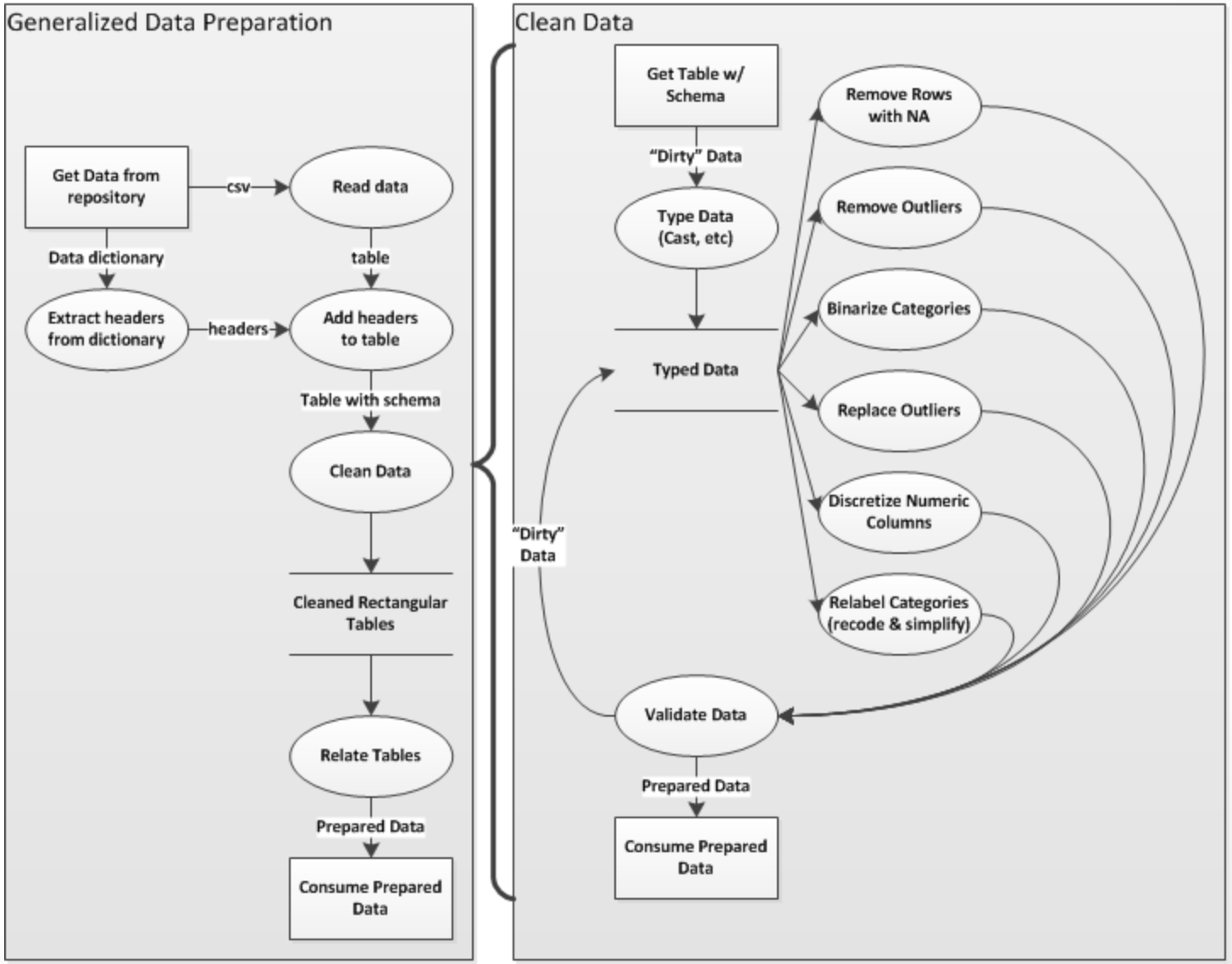
Data Preparation Review (8)



Data Preparation Review (9)



Data Preparation Review (10)



Data Preparation Review

Quiz 02

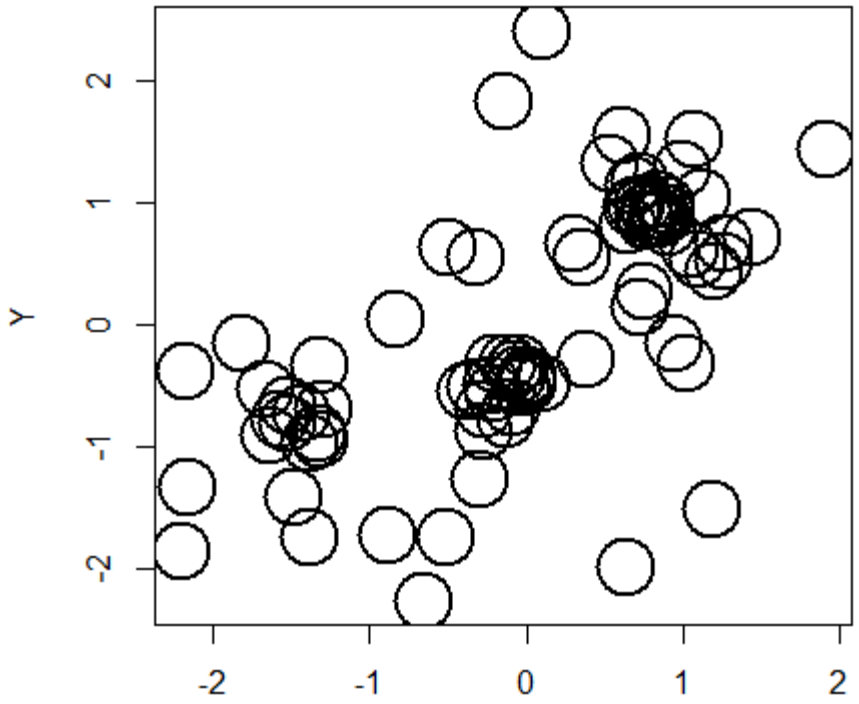
- Quiz available in Canvas

Introduction to K-means Clustering

K-means clustering: Algorithm

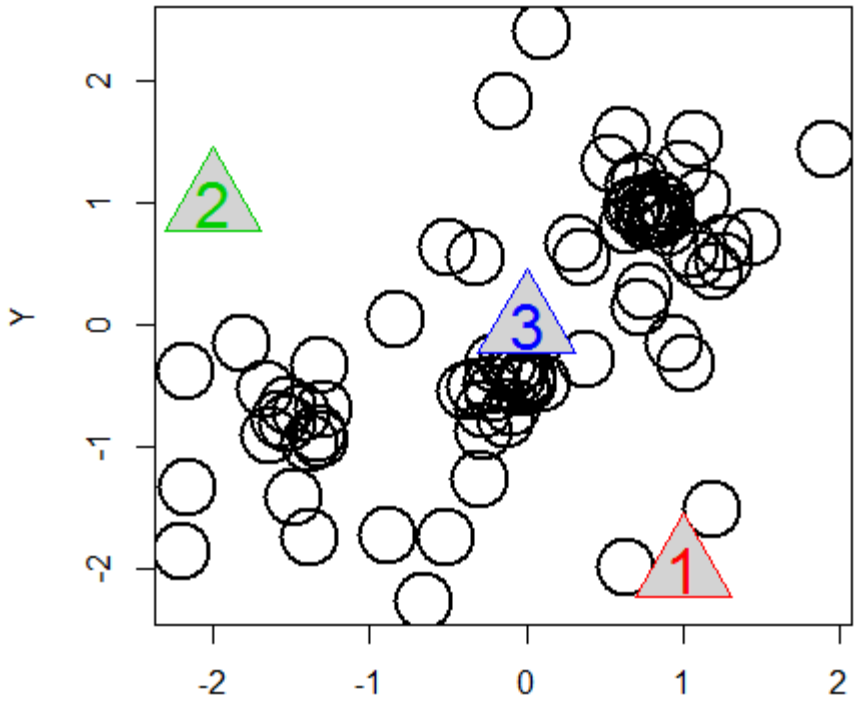
- Pre-requisites
  1. Get points in multi-dimensional space.
     - table, matrix, rectangular dataset
  2. Specify the number of clusters
     - Weakest point in algorithm (makes algorithm non-deterministic)
  3. Get a random center for each cluster
     - Another weak point in the algorithm
- Repeat until convergence:
  1. For each point, determine its closest cluster center and assign that point to that cluster
  2. Determine the centroid (mean) for each cluster of points

K-Means Clustering (0)



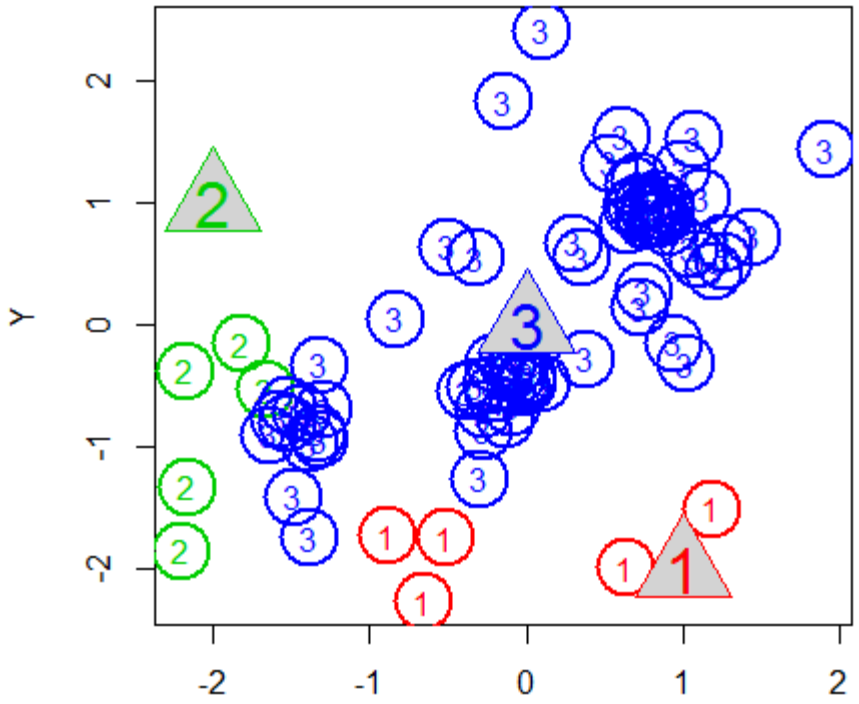
- Clustering starts by getting the data and representing the data as points in space. In this example the space is 2-dimensional.
- Each point describes an observation. An observation is an individual item.
- The dimensions are attributes that describe the item.

K-Means Clustering (1)



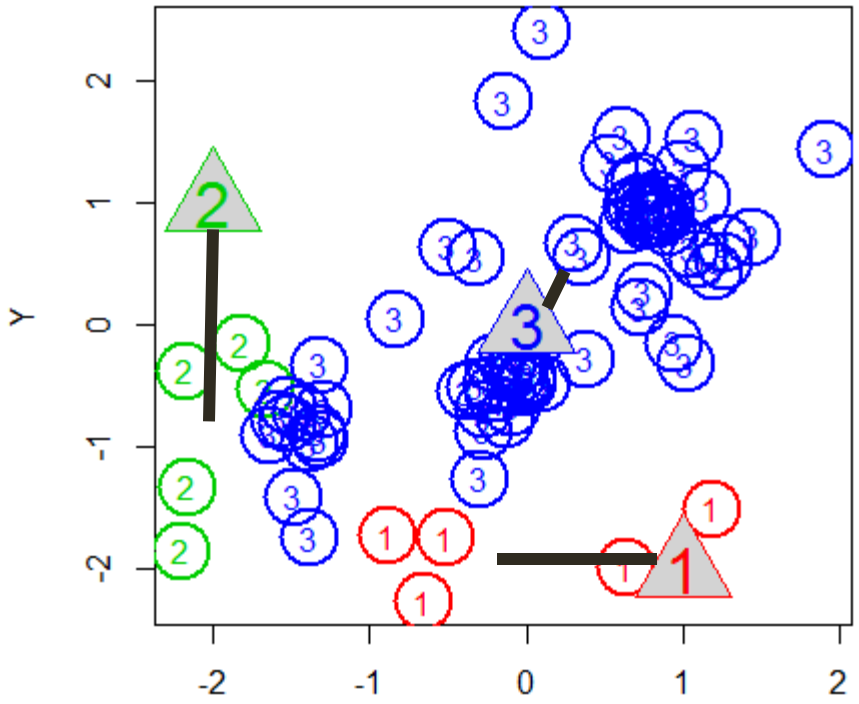
- Clustering continues by guessing, presuming, or specifying a number of clusters.
- Each centroid represents a cluster.
- The centroid positions are determined randomly. The centroids should be within the bounds of the points.

K-Means Clustering (2)



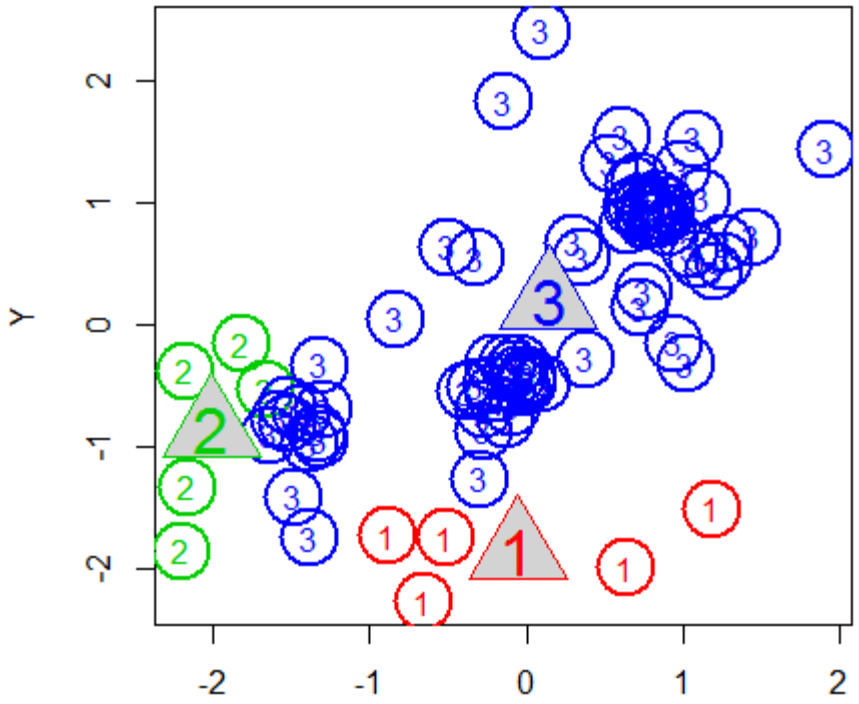
- Clustering continues by assigning each point to a cluster.
- For each point, the algorithm measures the distance to each centroid.
- For each point, the smallest distance to a centroid indicates the assignment.

K-Means Clustering (2)



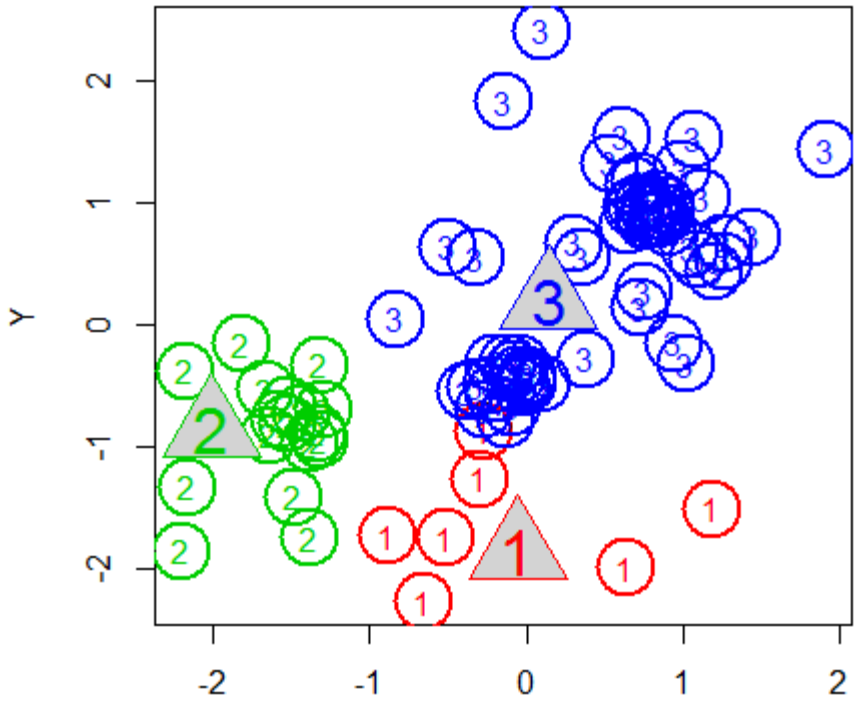
- Clustering continues by moving each centroid to the center of its cluster.

K-Means Clustering (3)



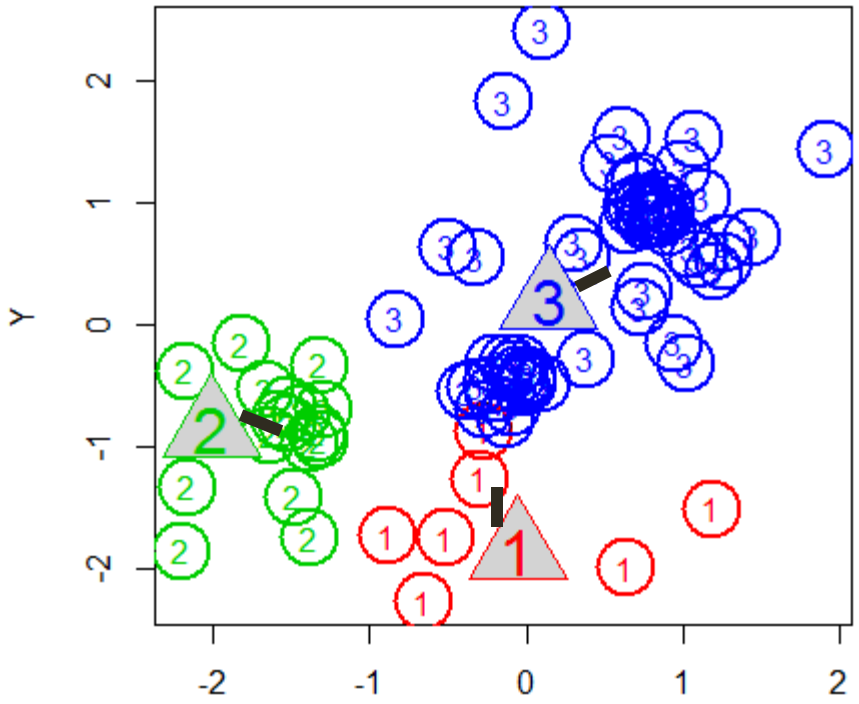
- Clustering continues by moving each centroid to the center of its cluster.

K-Means Clustering (4)



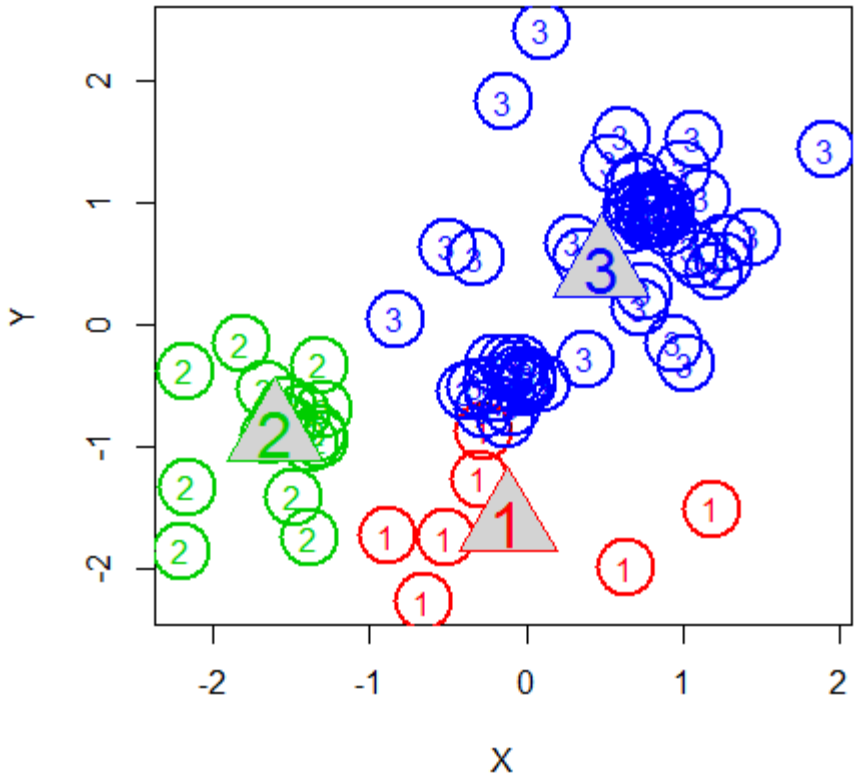
- Clustering continues by assigning each point to a cluster.
- For each point, the algorithm measures the distance to each centroid.
- For each point, the smallest distance to a centroid indicates the assignment.

K-Means Clustering (4)

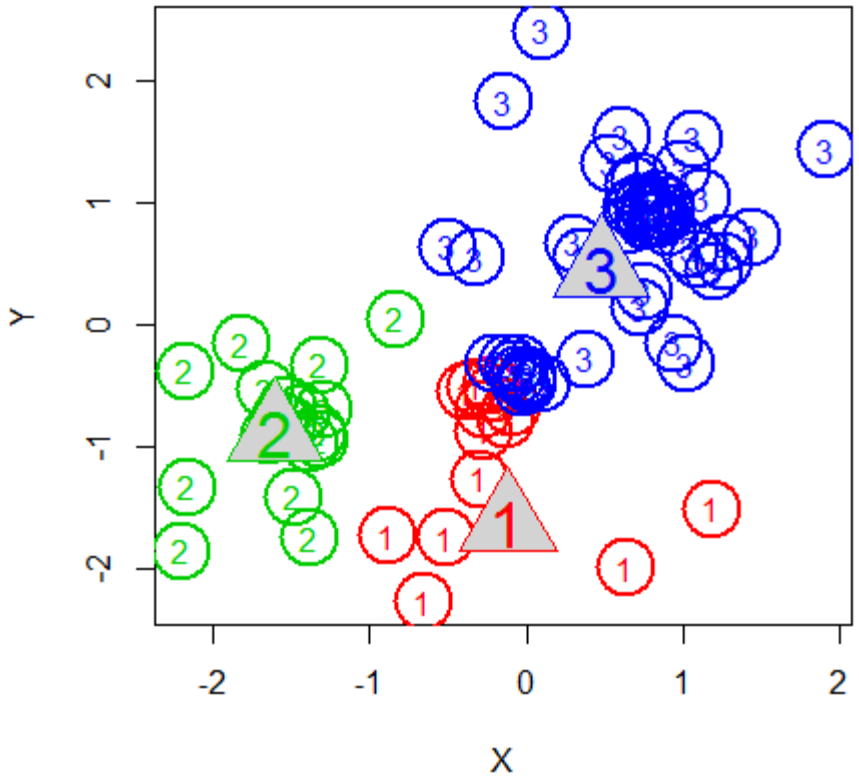


- Clustering continues by assigning each point to a cluster.
- For each point, the algorithm measures the distance to each centroid.
- For each point, the smallest distance to a centroid indicates the assignment.

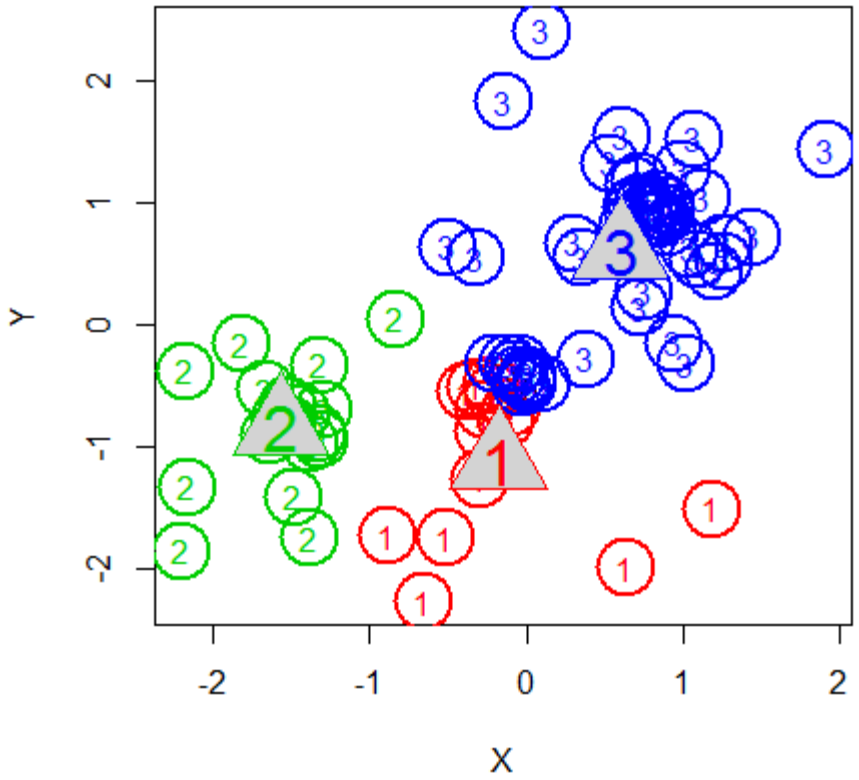
K-Means Clustering (5)



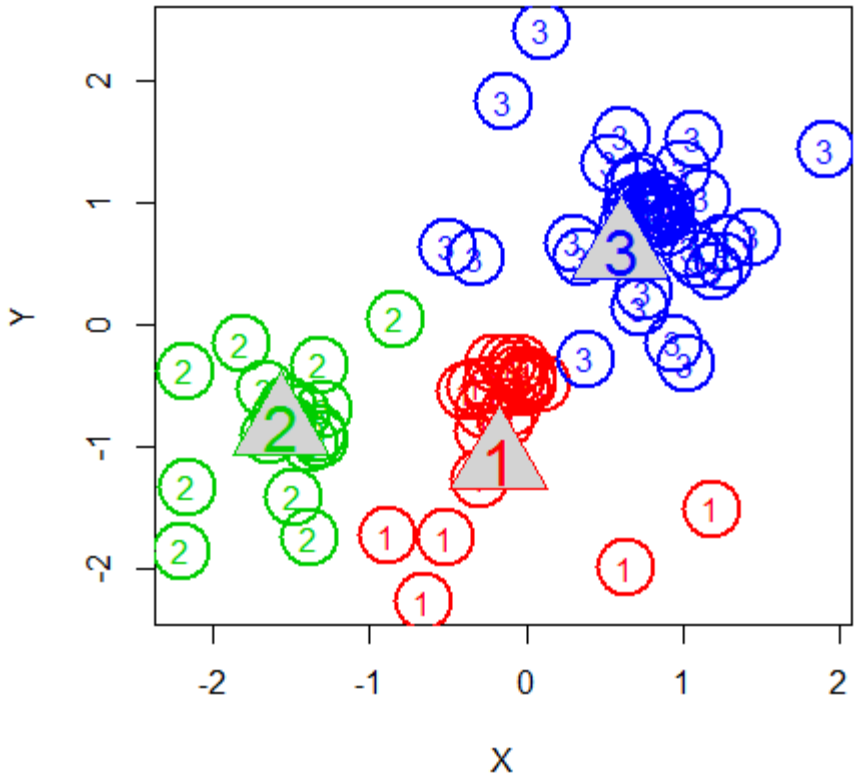
K-Means Clustering (6)



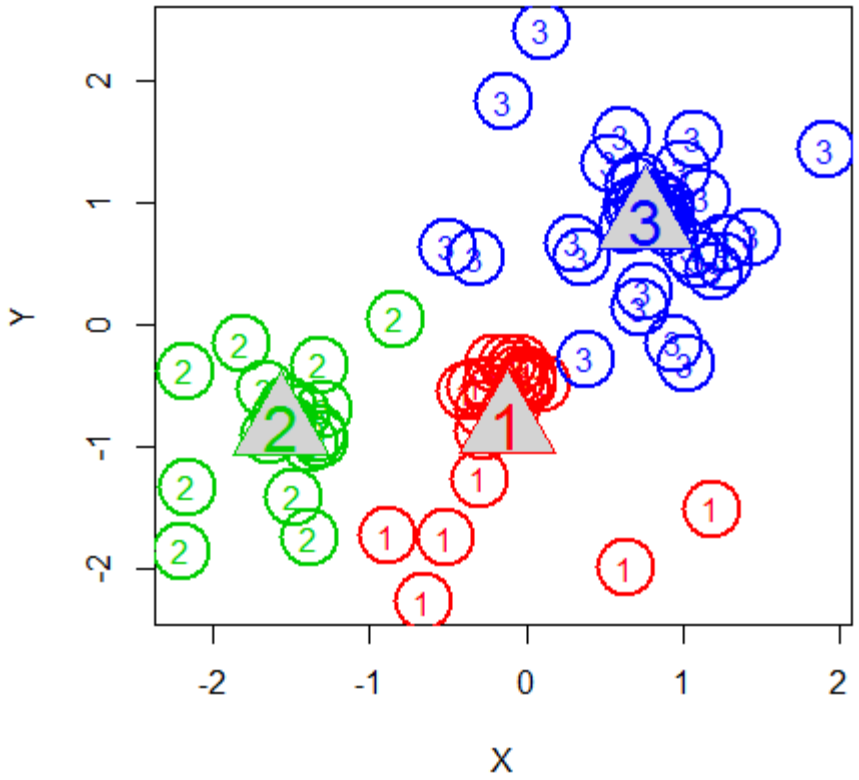
K-Means Clustering (7)



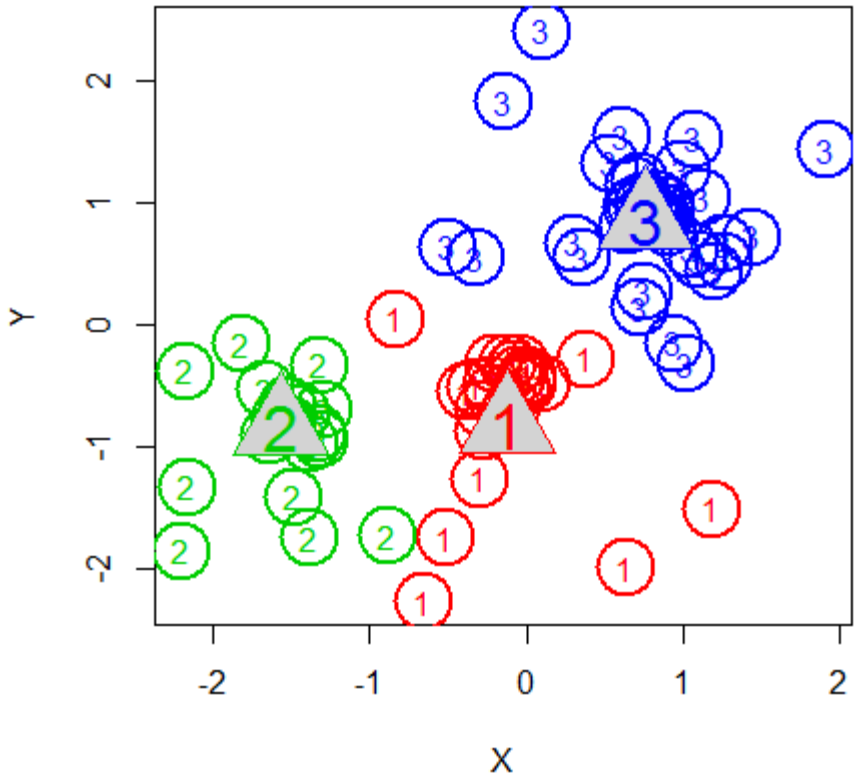
K-Means Clustering (8)



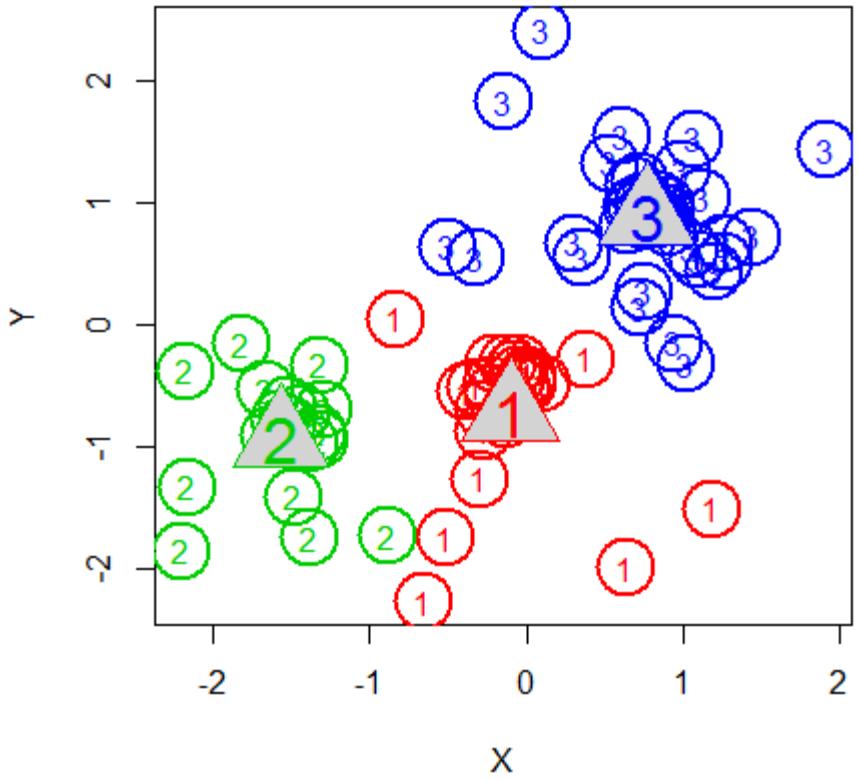
K-Means Clustering (9)



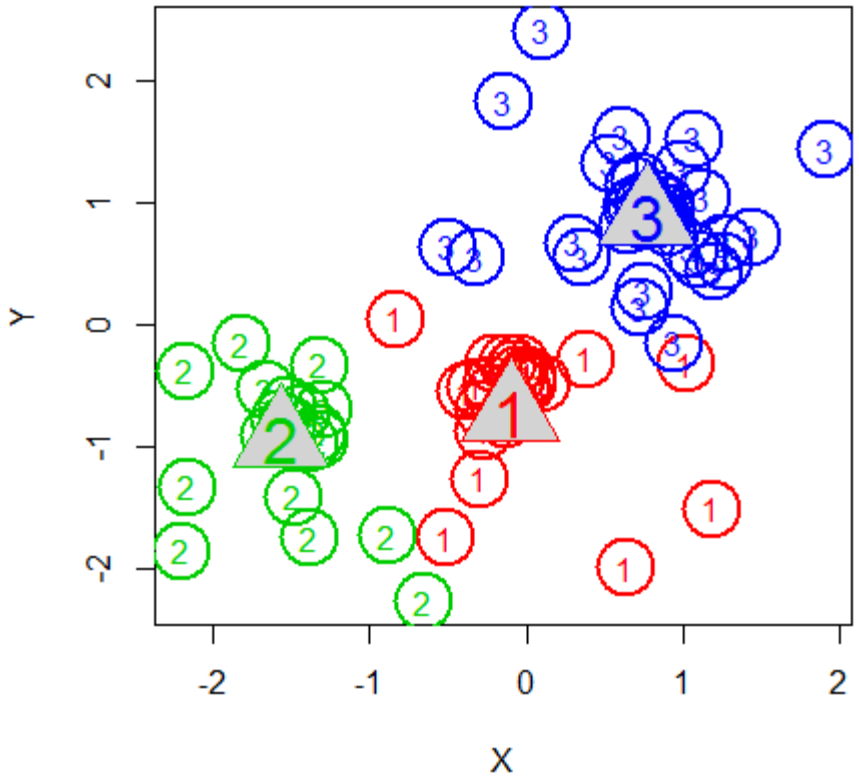
K-Means Clustering (10)



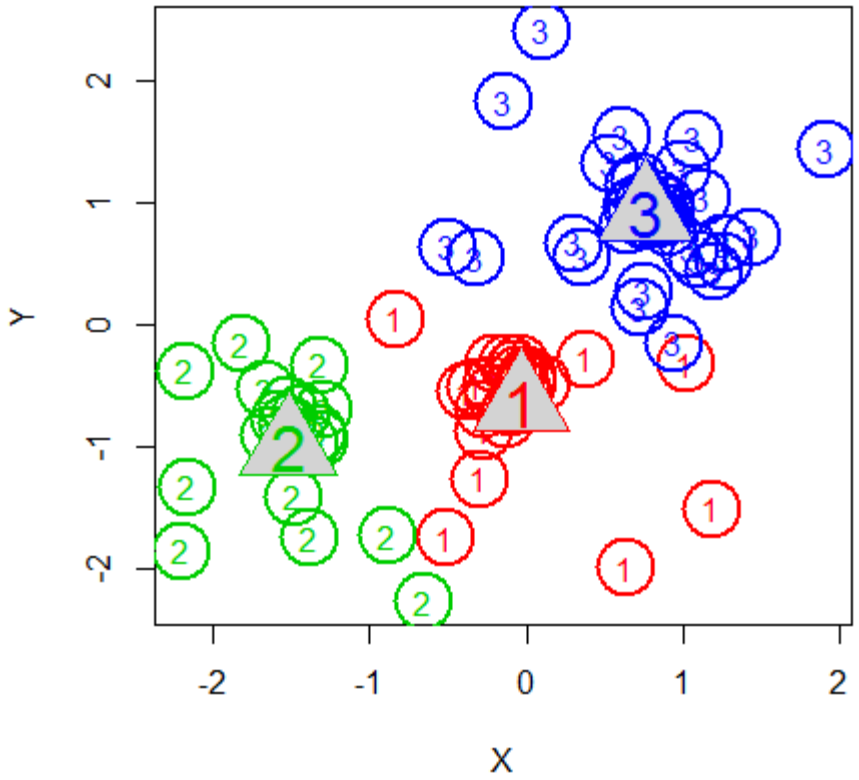
K-Means Clustering (11)



K-Means Clustering (12)



K-Means Clustering (13)



K-means Demo

- K-Means Clustering (Kmeans.R)
- Clustering of Patients (Segmentation of a Population)

K-means

- Some Points:
  - Normalizations are important to put data on equal terms
  - Initial centroid number and placement is an art.
  - Categorical Data must be binarized
  - K-means is unsupervised because we do not tell the algorithm what outcome was observed or what outcome is desired.

Break

In-Class Exercise and Homework Assignment

Write K-Means in R: Kmeans_Skeleton.R

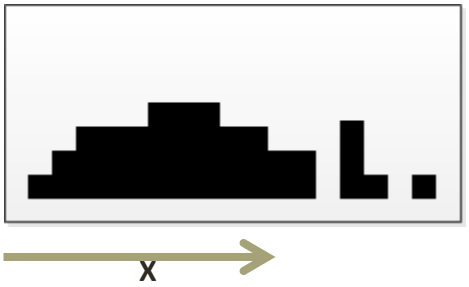
- Write a version of K-Means in R and name the file KMeans.R. The function signatures should be the same as those in Kmeans_Skeleton.R, Specifically, implement
  - **KMeans <- function(observations = sampleObservations, clusterCenters = centersGuess)**
  - **findLabelOfClosestCluster <- function(observations = sampleObservations, clusterCenters=centersGuess)**
  - **calculateClusterCenters <- function(observations=sampleObservations, clusterLabels=labelsRandom)**
- You can use Kmeans_Skeleton.R as a template and replace all lines that say: "**Put code in place of this line**". Execute the built in tests and verify that your code works:
  - **ClusterPlot()**
  - **findLabelOfClosestCluster()**
  - **calculateClusterCenters()**
  - **KMeans()**

Break

Introduction to K-means Clustering

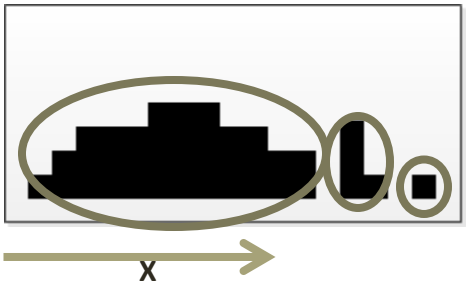
Dimensions in Clustering

Clustering: Dimensions (1)



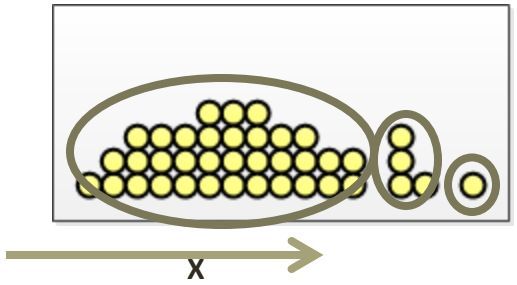
Where are the three clusters?

Clustering: Dimensions (2)



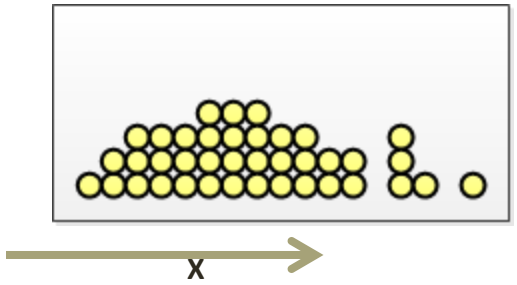
Simple assignment based on a 1D distribution

Clustering: Dimensions (3)



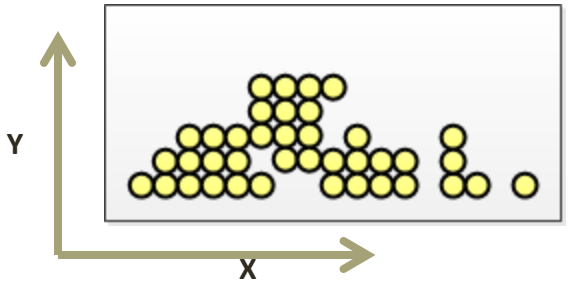
Simple assignment based on a 1D distribution

Clustering: Dimensions (4)



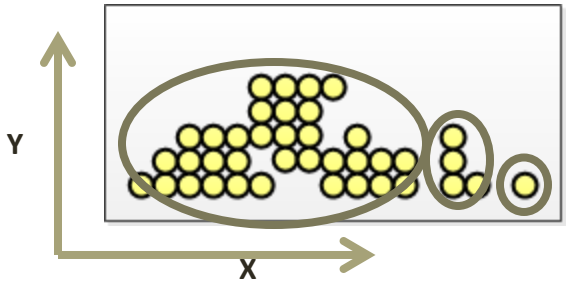
What if this was not a 1D distribution?

Clustering: Dimensions (5)



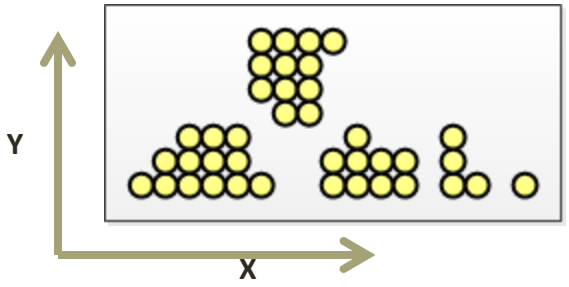
The distribution is in 2D. Some points differ in the 2nd D

Clustering: Dimensions (6)



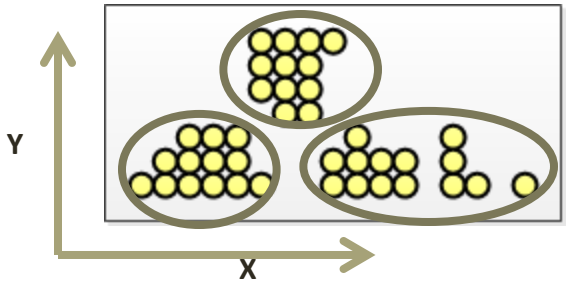
If the difference is minor, we still get the same clusters

Clustering: Dimensions (7)



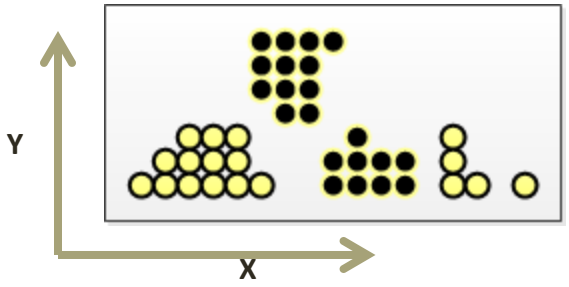
The difference could be significant

Clustering: Dimensions (8)



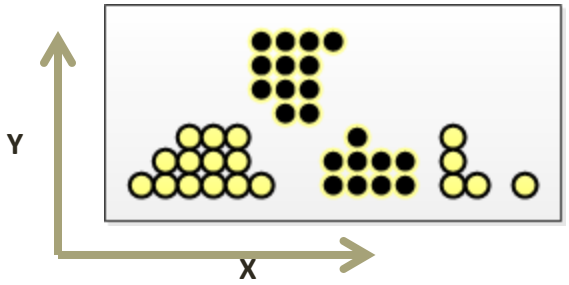
A big difference in the 2nd D can lead to different clusters

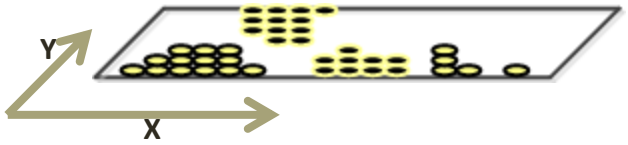
Clustering: Dimensions (9)



We can introduce another D by color coding. This is a Boolean Dimension

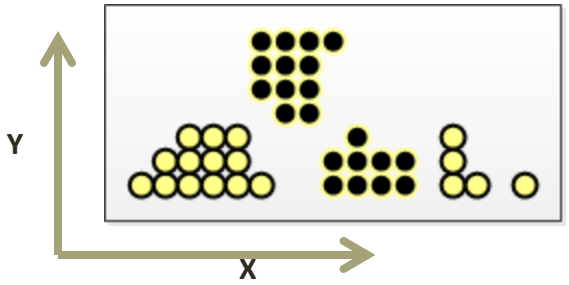
Clustering: Dimensions (10)

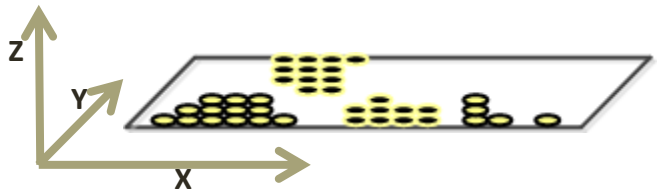




Create a 3rd Dimansion

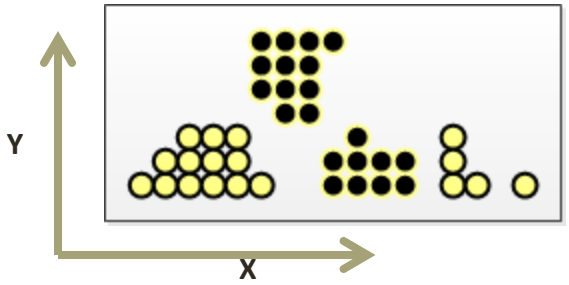
Clustering: Dimensions (11)

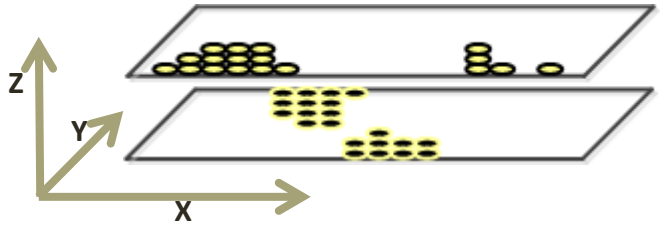




Create a 3rd Dimansion

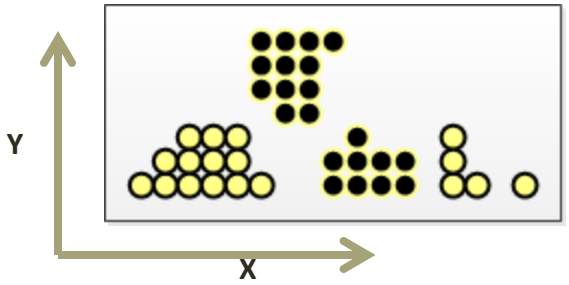
Clustering: Dimensions (12)

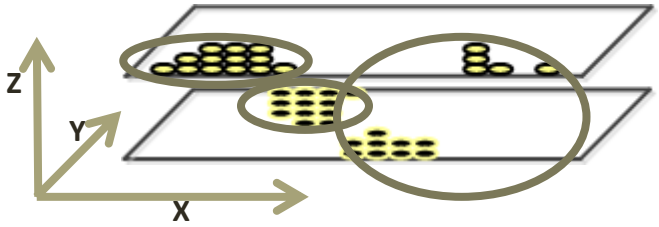


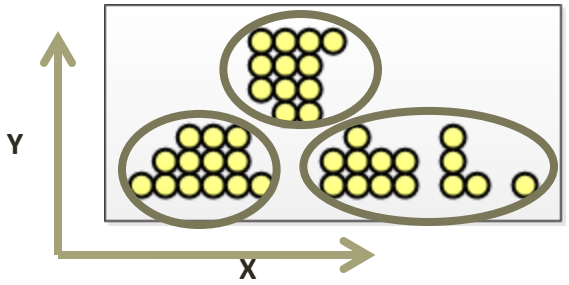


Where are the 3 clusters now?

Clustering: Dimensions (13)

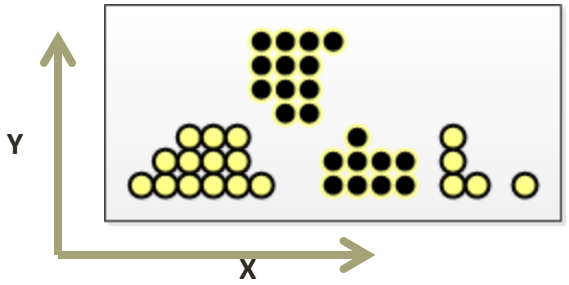


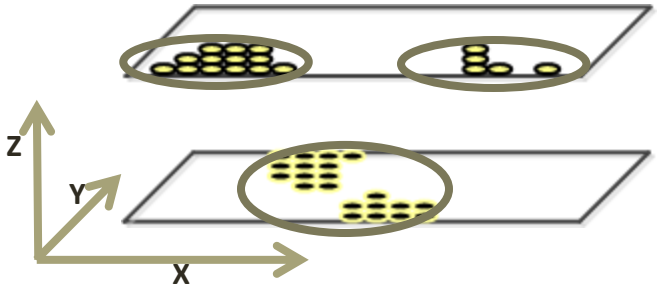


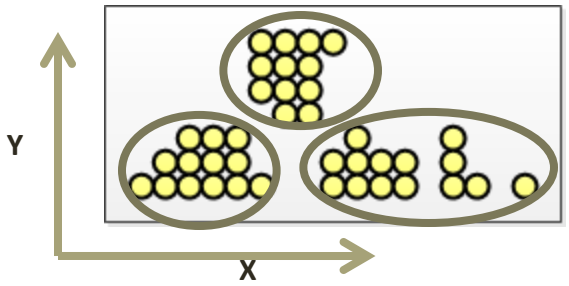


If the 3rd is small, then the clustering is the same as in 2D

Clustering: Dimensions (14)







If the 3rd is big, then the clustering differs from 2D

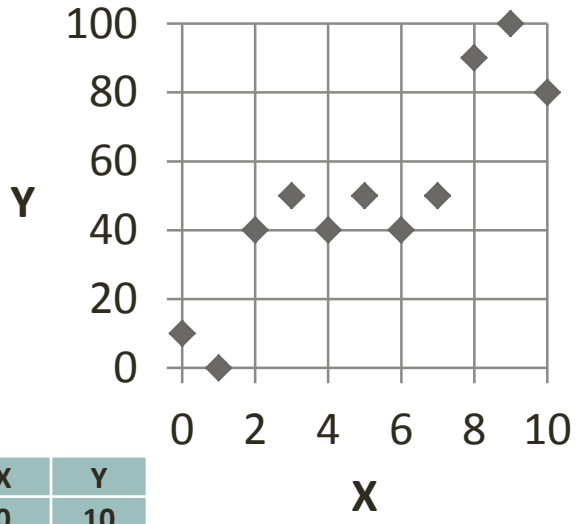
Dimensions in Clustering

Normalization in Clustering

Normalization of a linear relationship (1)

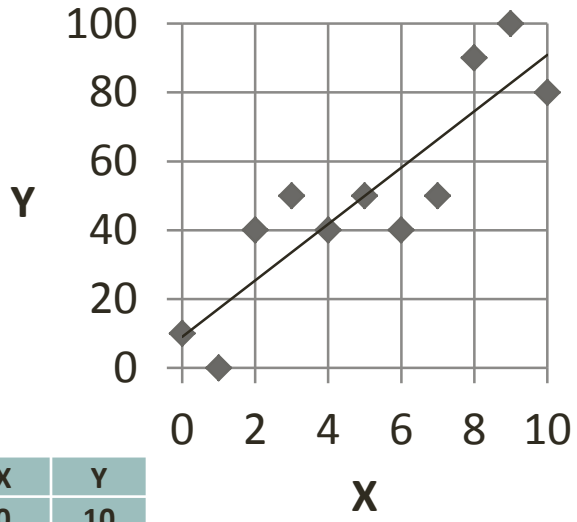
X	Y
0	10
1	0
2	40
3	50
4	40
5	50
6	40
7	50
8	90
9	100
10	80

Normalization of a linear relationship (2)



X	Y
0	10
1	0
2	40
3	50
4	40
5	50
6	40
7	50
8	90
9	100
10	80

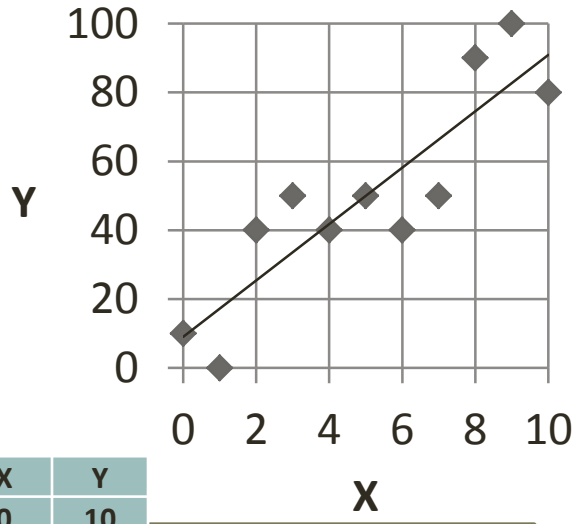
Normalization of a linear relationship (3)



X	Y
0	10
1	0
2	40
3	50
4	40
5	50
6	40
7	50
8	90
9	100
10	80

$Y = 10 + 8*X$

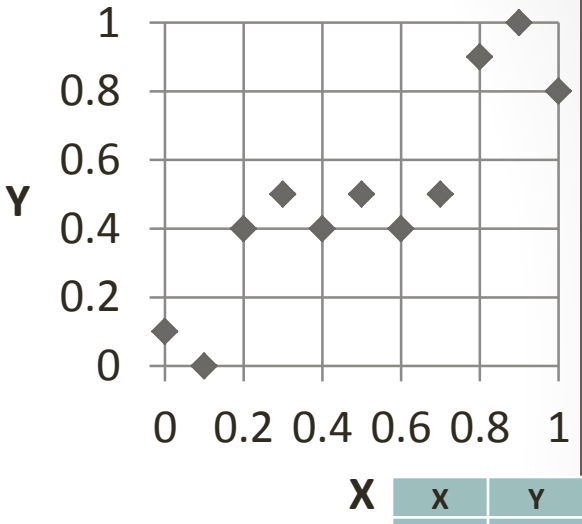
Normalization of a linear relationship (4)



X	Y
0	10
1	0
2	40
3	50
4	40
5	50
6	40
7	50
8	90
9	100
10	80

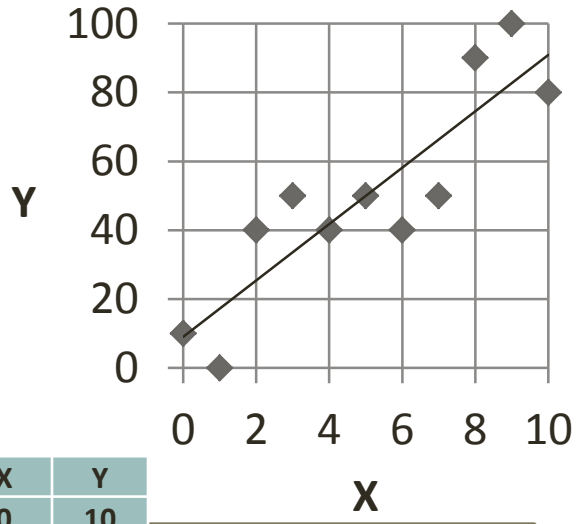
Y = 10 + 8*X

Normalize



X	Y
0	0.1
0.1	0
0.2	0.4
0.3	0.5
0.4	0.4
0.5	0.5
0.6	0.4
0.7	0.5
0.8	0.9
0.9	1
1	0.8

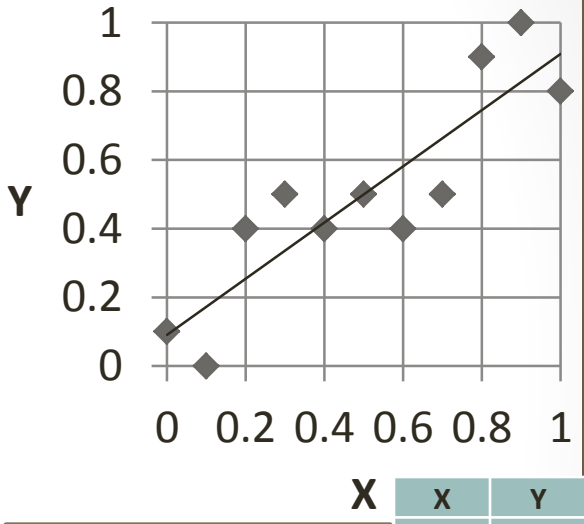
Normalization of a linear relationship (5)



X	Y
0	10
1	0
2	40
3	50
4	40
5	50
6	40
7	50
8	90
9	100
10	80

Y = 10 + 8*X

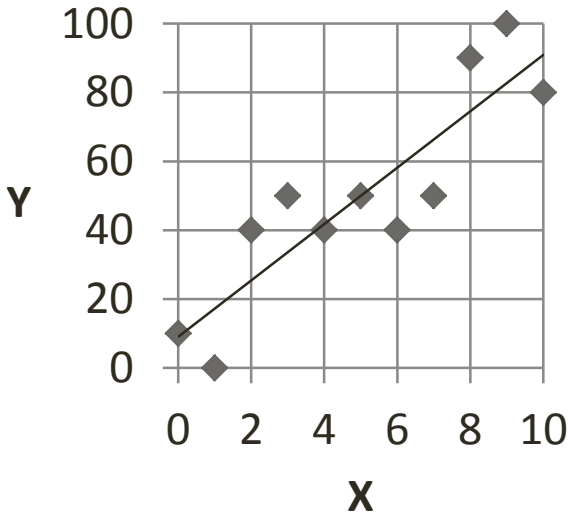
Normalize



Y = 0.1 + 0.8*X

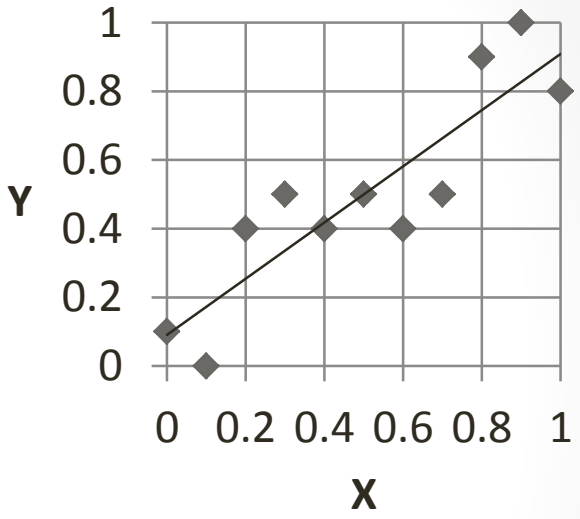
X	Y
0	0.1
0.1	0
0.2	0.4
0.3	0.5
0.4	0.4
0.5	0.5
0.6	0.4
0.7	0.5
0.8	0.9
0.9	1
1	0.8

Normalization of a linear relationship (6)



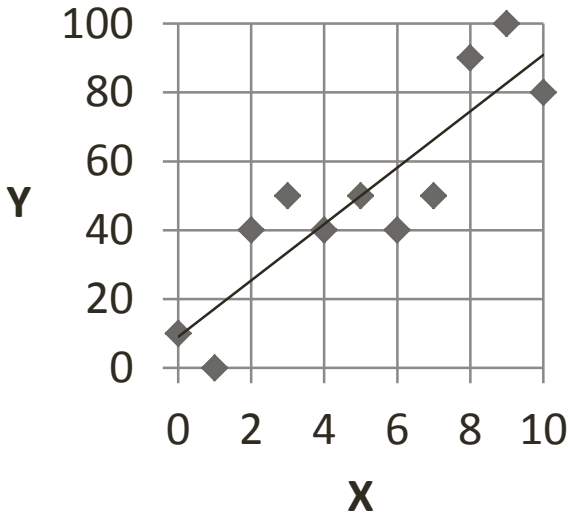
Y = 10 + 8*X

Normalize



Y = 0.1 + 0.8*X

Normalization of a linear relationship (7)



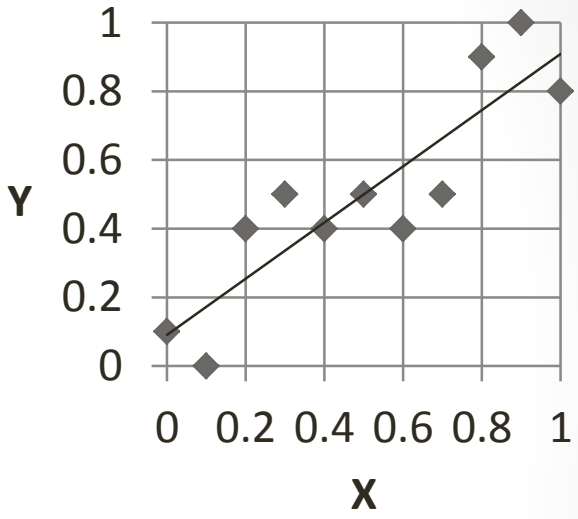
Y = 10 + 8*X

Normalize

Normalize Input
X = 2 -> X' = 0.2

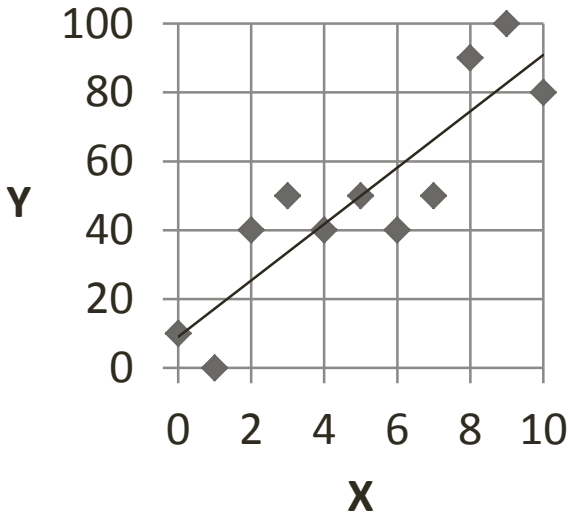
Predict Output
X' = 0.2 -> Y'= 0.26

Denormalize Output
Y'= 0.26 -> Y = 26



Y = 0.1 + 0.8*X

Normalization of a linear relationship (8)



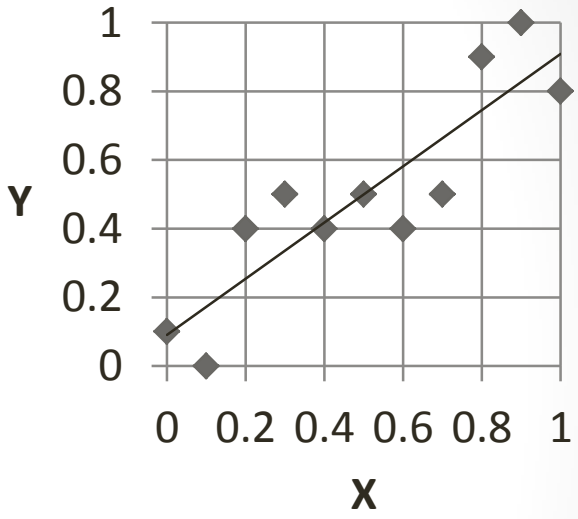
Y = 10 + 8*X

Normalize

Normalize Input
X = 2 -> X' = 0.2

Predict Output
X' = 0.2 -> Y'= 0.26

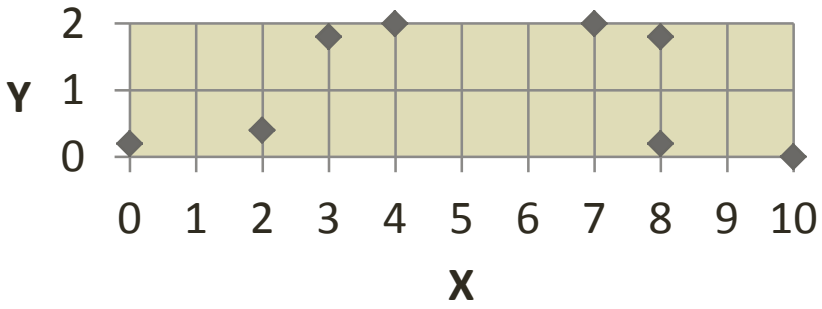
Denormalize Output
Y'= 0.26 -> Y = 26



Y = 0.1 + 0.8*X

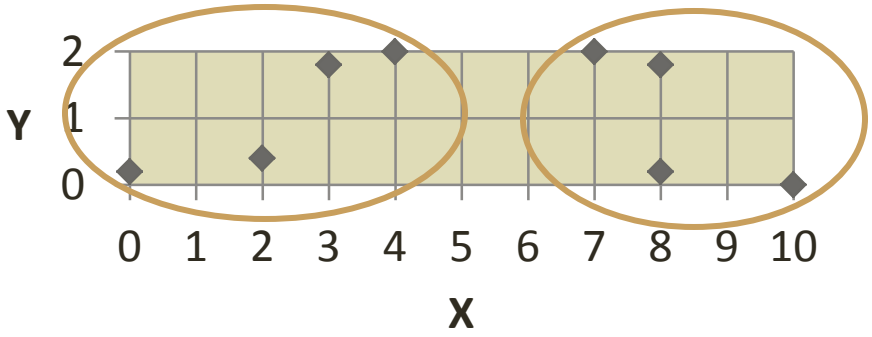
Prediction in Original Space:
X = 2 -> Y = 26

Normalization of a non-linear relationship (1)



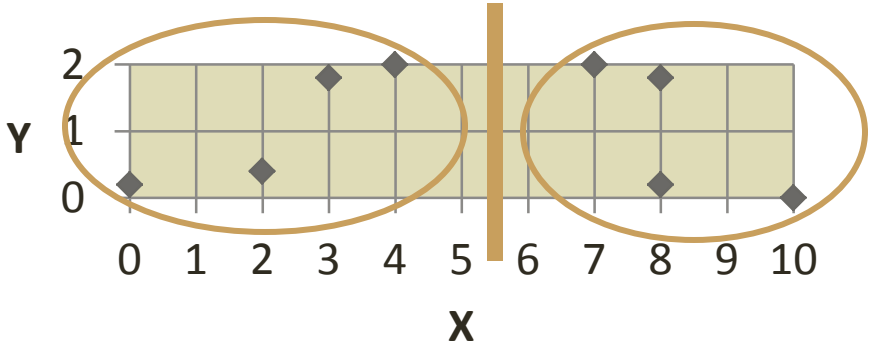
Original data in 2D:
Find 2 clusters

Normalization of a non-linear relationship (2)



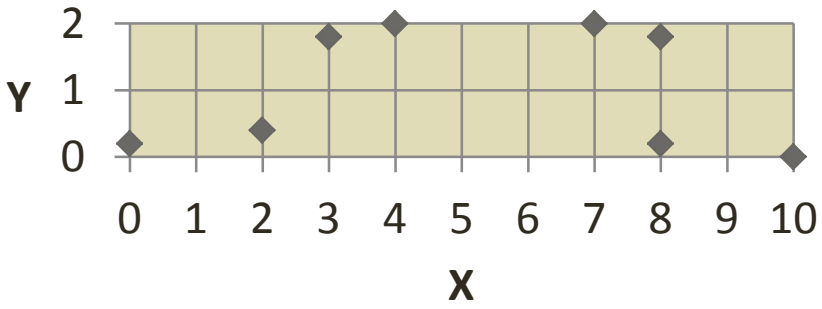
Found 2 Clusters

Normalization of a non-linear relationship (3)



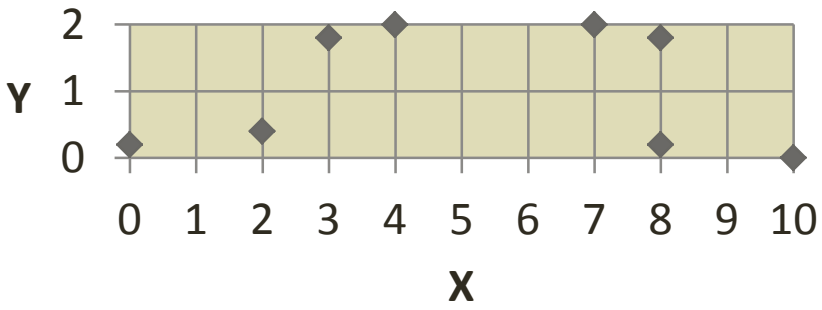
Clusters segment the image

Normalization of a non-linear relationship (4)



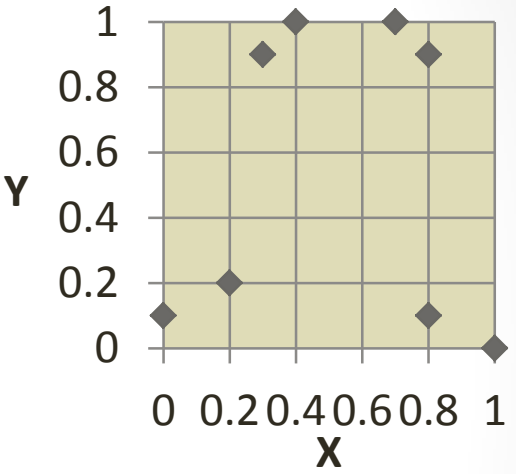
Non-normalized 2D data

Normalization of a non-linear relationship (5)



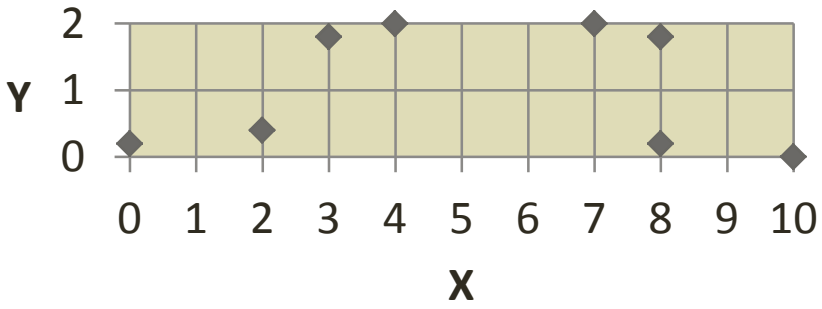
Non-normalized 2D data

Normalize

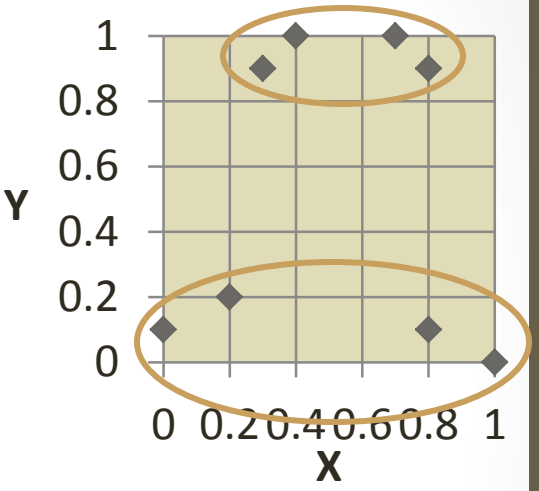


Normalize the data:
Search for 2 Clusters

Normalization of a non-linear relationship (6)



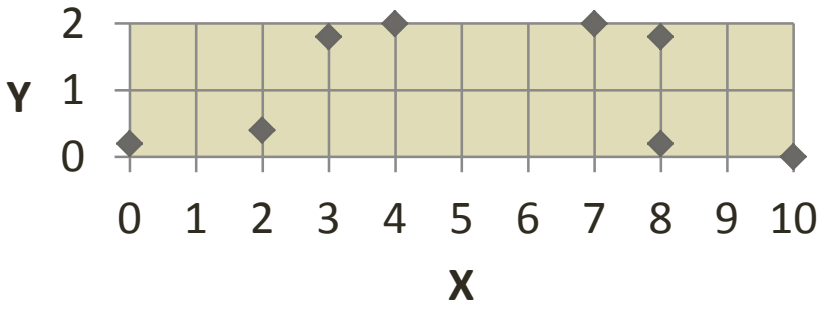
Normalize



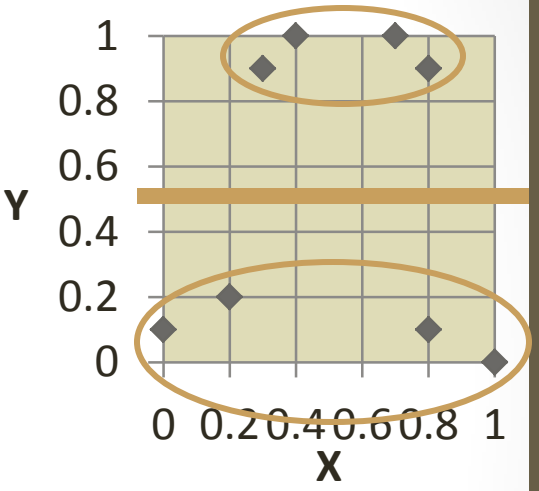
Non-normalized 2D data

Found 2 Clusters in the normalized data

Normalization of a non-linear relationship (6)



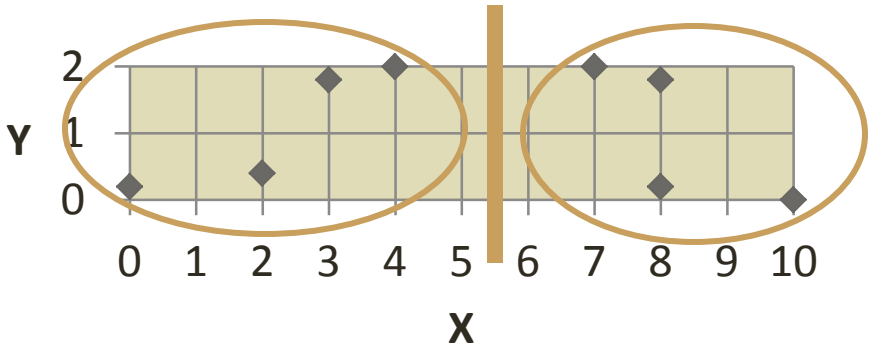
Normalize



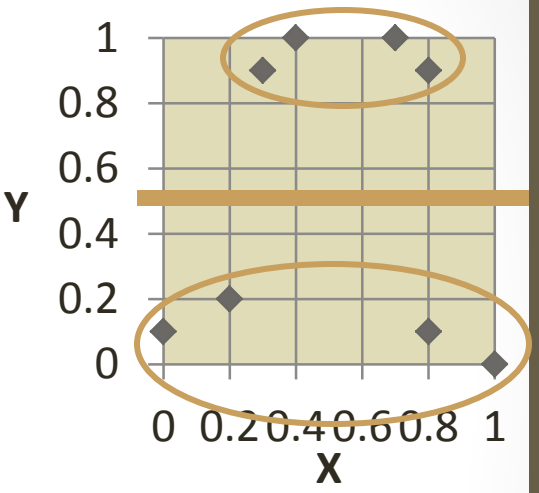
Non-normalized 2D data

Clusters Segment the Image

Normalization of a non-linear relationship (7)



Normalize



Clustering before normalization

Clustering after normalization

Normalization of Linear and Non-Linear Outcomes

- Non-linear (Normalization can change outcome):
- K-Means
- Neural Net

- Linear (Normalization should not change outcome):
- Logistic Regression
- Linear Regression
- Mixture of Gaussians

Normalization in Clustering

Assignment (0)

1. Start a discussion, or make a comment on an existing discussion in the LinkedIn group.
2. Review the slide: "In-Class Exercise and Homework Assignment" . Copy Kmeans_Skeleton.R to Kmeans.R. Complete Kmeans.R. Make sure you get the test results. I will test your code with other centers and points.
3. Submit your R script titled Kmeans.R from item 2. Submit to the Homework Submission site on Canvas for this module by the due date. If you cannot submit the assignment on time, please notify me before the deadline at ErnstHe@UW.edu.
4. Take a look at the part of DataScience01a.R that is titled: A glimpse into what we will do in future lessons (Do not submit anything for this item -- just play with it)
5. Reading Assignment
   - Look through Preview section of Lesson02 Overview
   - AFewUsefulThingsToKnowAboutMachineLearning.pdf
   - http://homes.cs.washington.edu/~pedrod/papers/cacm12.pdf
   - http://en.wikipedia.org/wiki/Supervised_learning
   - http://en.wikipedia.org/wiki/Unsupervised_learning

Introduction to Data Science