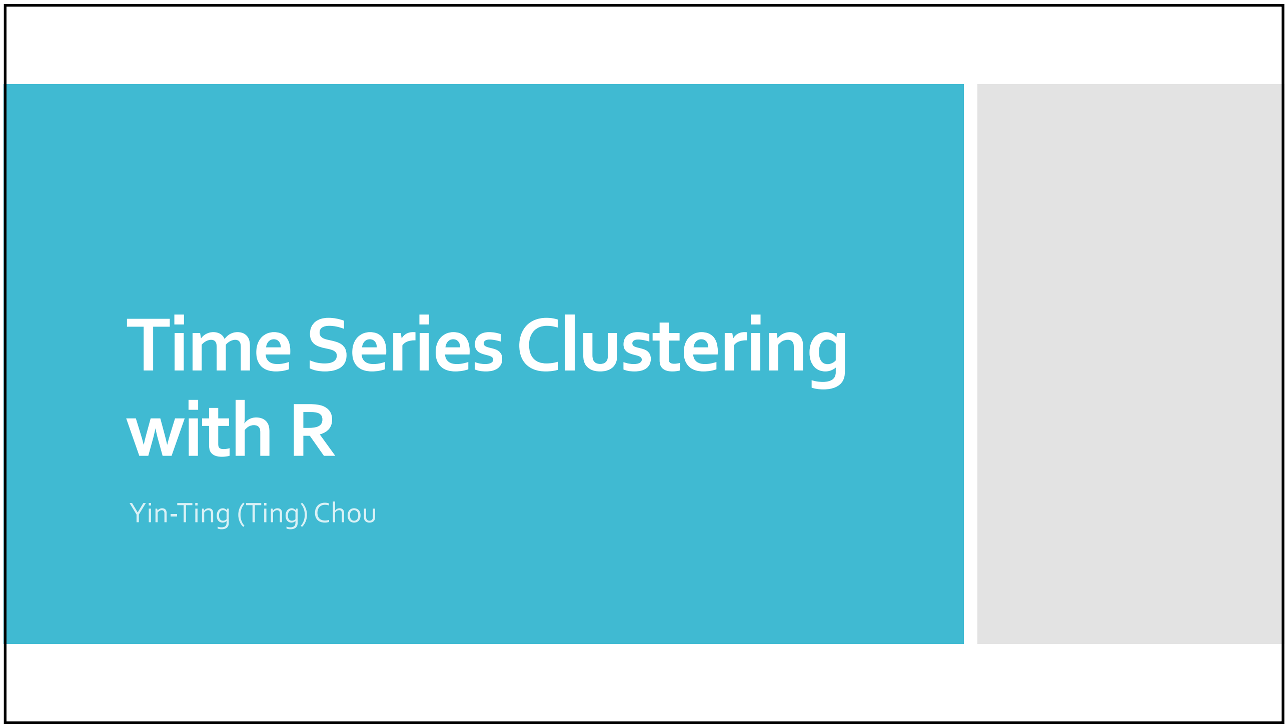
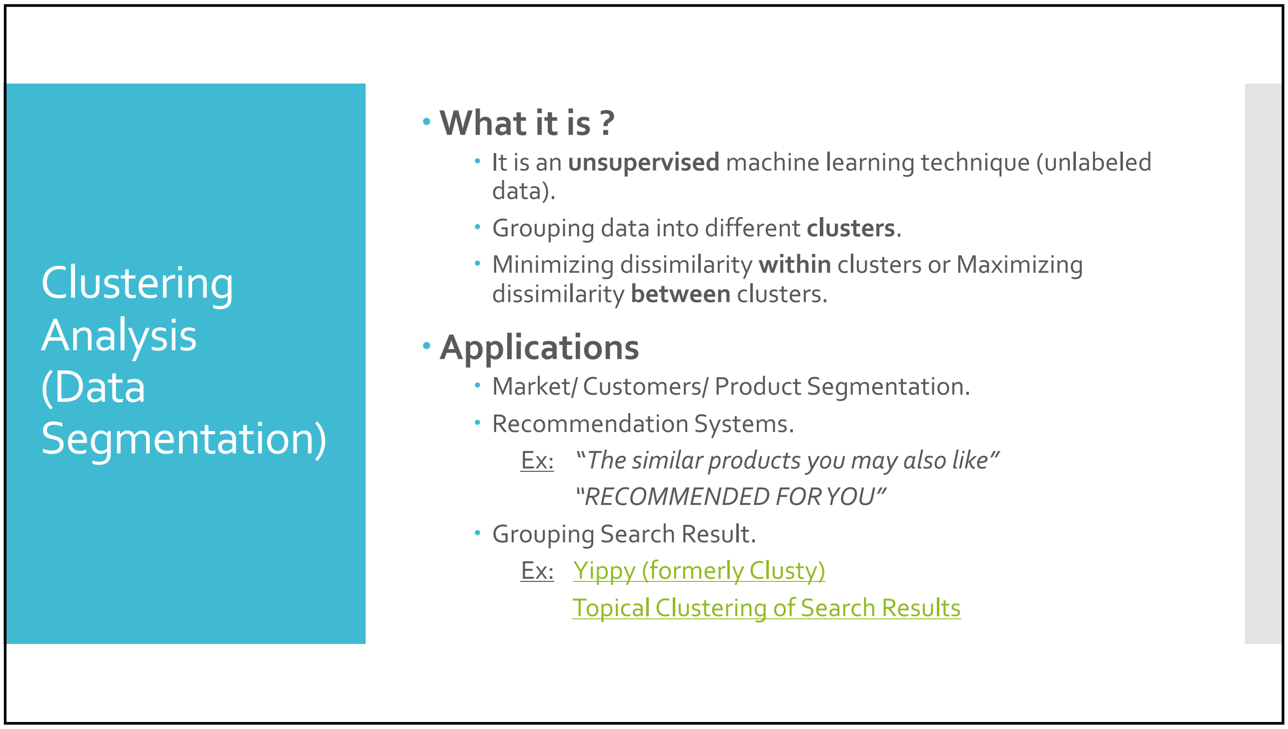


Time Series Clustering with R

Yin-Ting (Ting) Chou



Clustering Analysis (Data Segmentation)

- **What it is ?**
  - It is an **unsupervised** machine learning technique (unlabeled data).
  - Grouping data into different **clusters**.
  - Minimizing dissimilarity **within** clusters or Maximizing dissimilarity **between** clusters.
- **Applications**
  - Market/ Customers/ Product Segmentation.
  - Recommendation Systems.

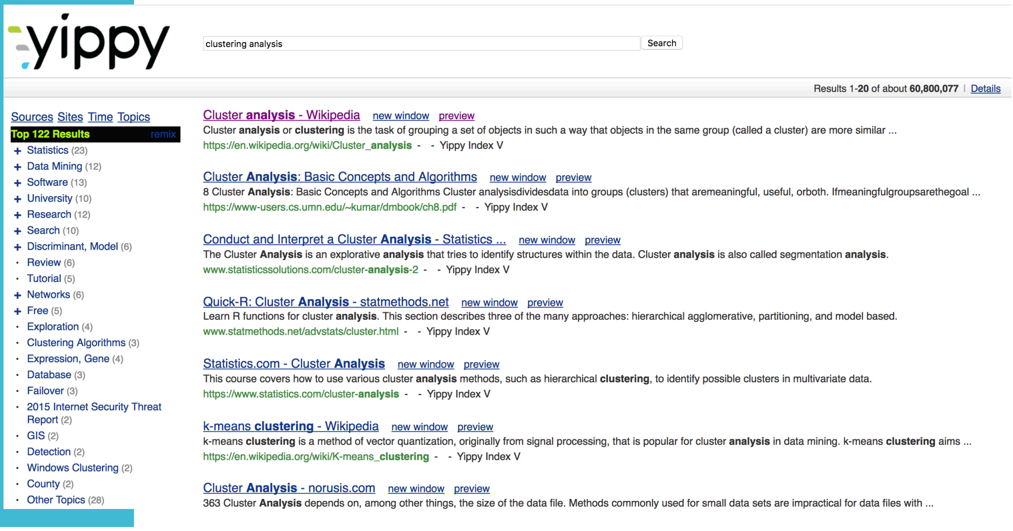
    Ex: *"The similar products you may also like"*

    "RECOMMENDED FOR YOU"
  - Grouping Search Result.

    Ex: Yippy (formerly Clusty)

    Topical Clustering of Search Results

Yippy



Results 1-**20** of about **60,800,077** | Details

Cluster **analysis** - Wikipedia new window preview
Cluster **analysis** or **clustering** is the task of grouping a set of objects in such a way that objects in the same group (called a cluster) are more similar ...
https://en.wikipedia.org/wiki/Cluster_**analysis** - - Yippy Index V

Cluster **Analysis**: Basic Concepts and Algorithms new window preview
8 Cluster **Analysis**: Basic Concepts and Algorithms Cluster analysisdividesdata into groups (clusters) that aremeaningful, useful, orboth. Ifmeaningfulgroupsarethegoal ...
https://www-users.cs.umn.edu/~kumar/dmbook/ch8.pdf - - Yippy Index V

Conduct and Interpret a Cluster **Analysis** - Statistics ... new window preview
The Cluster **Analysis** is an explorative **analysis** that tries to identify structures within the data. Cluster **analysis** is also called segmentation **analysis**.
www.statisticssolutions.com/cluster-**analysis**-2 - - Yippy Index V

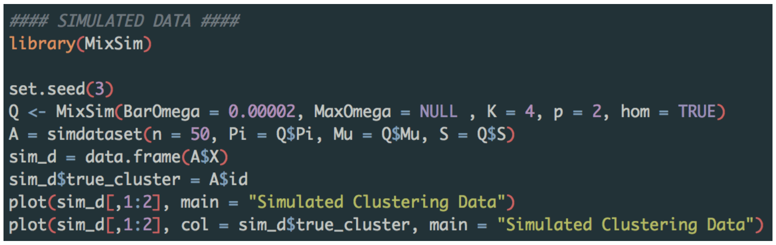
Quick-R: Cluster **Analysis** - statmethods.net new window preview
Learn R functions for cluster **analysis**. This section describes three of the many approaches: hierarchical agglomerative, partitioning, and model based.
www.statmethods.net/advstats/cluster.html - - Yippy Index V

Statistics.com - Cluster **Analysis** new window preview
This course covers how to use various cluster **analysis** methods, such as hierarchical **clustering**, to identify possible clusters in multivariate data.
https://www.statistics.com/cluster-**analysis** - - Yippy Index V

k-means **clustering** - Wikipedia new window preview
k-means **clustering** is a method of vector quantization, originally from signal processing, that is popular for cluster **analysis** in data mining. k-means **clustering** aims ...
https://en.wikipedia.org/wiki/K-means_**clustering** - - Yippy Index V

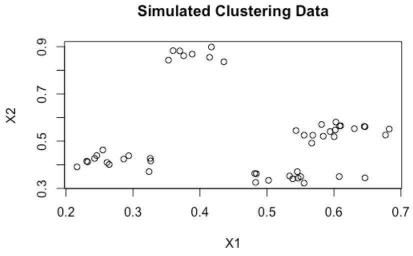
Cluster **Analysis** - norusis.com new window preview
363 Cluster **Analysis** depends on, among other things, the size of the data file. Methods commonly used for small data sets are impractical for data files with ...

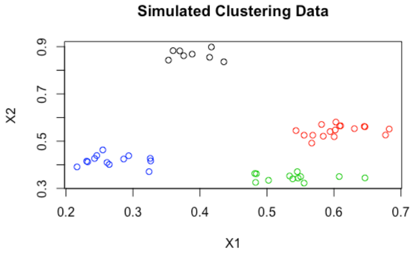
Example Data

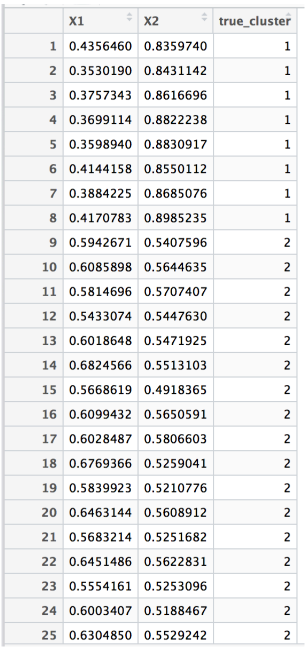


```
#### SIMULATED DATA ####
library(MixSim)

set.seed(3)
Q <- MixSim(BarOmega = 0.00002, MaxOmega = NULL , K = 4, p = 2, hom = TRUE)
A = simdataset(n = 50, Pi = Q$Pi, Mu = Q$Mu, S = Q$S)
sim_d = data.frame(A$X)
sim_d$true_cluster = A$id
plot(sim_d[,1:2], main = "Simulated Clustering Data")
plot(sim_d[,1:2], col = sim_d$true_cluster, main = "Simulated Clustering Data")
```

Example Data





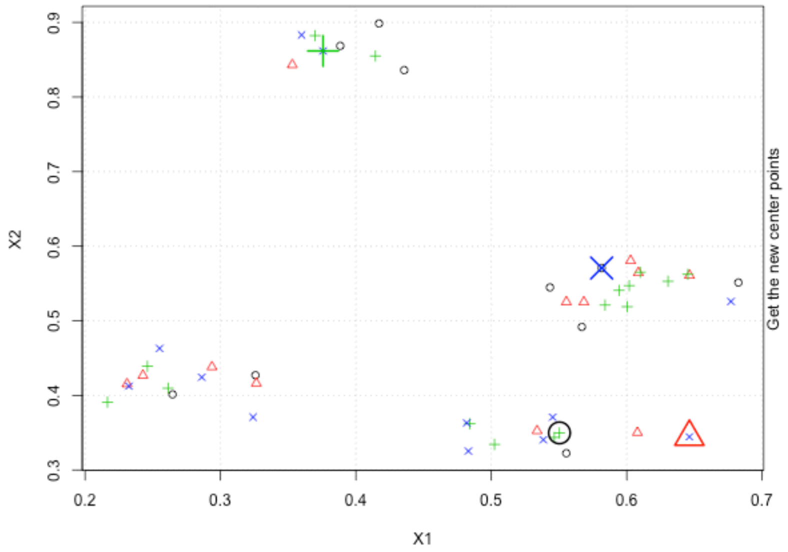


	X1	X2	true_cluster
1	0.4356460	0.8359740	1
2	0.3530190	0.8431142	1
3	0.3757343	0.8616696	1
4	0.3699114	0.8822238	1
5	0.3598940	0.8830917	1
6	0.4144158	0.8550112	1
7	0.3884225	0.8685076	1
8	0.4170783	0.8985235	1
9	0.5942671	0.5407596	2
10	0.6085898	0.5644635	2
11	0.5814696	0.5707407	2
12	0.5433074	0.5447630	2
13	0.6018648	0.5471925	2
14	0.6824566	0.5513103	2
15	0.5668619	0.4918365	2
16	0.6099432	0.5650591	2
17	0.6028487	0.5806603	2
18	0.6769366	0.5259041	2
19	0.5839923	0.5210776	2
20	0.6463144	0.5608912	2
21	0.5683214	0.5251682	2
22	0.6451486	0.5622831	2
23	0.5554161	0.5253096	2
24	0.6003407	0.5188467	2
25	0.6304850	0.5529242	2

Well-Known Methods

- **K-means Clustering**
  1. Specify the number of clusters (K).
  2. Define what is **similar** or **dissimilar**
     → **Distance!**
  3. Algorithm
     1. Random assign all the points to a unique cluster (k), where k = 1,2,3,…K.
     2. Calculate the mean center point for each cluster (m_1, m_2, …, m_K).
     3. Reassign the point to the cluster that it has closest distance to those mean center points.
     4. Repeat 2. ~ 3., until the assignment stop changing.

K-means Animation!



Definition for (Dis)similarity

- Suppose we have a measured point, x_{ij}, where

  Points: $i = 1, 2, 3, \dots, \mathrm{N}$

  Attributes: $j = 1, 2, 3, \dots, \mathrm{p}$

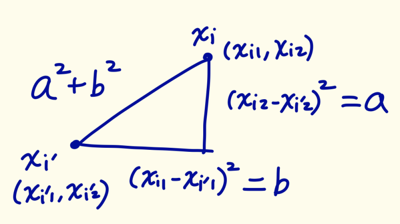
- Dissimilarity between point i andi'

$$D(x_i, x_{i'}) = \sum_{j=1}^{p} d_j(x_{ij}, x_{i'j})$$

- Euclidean Distance

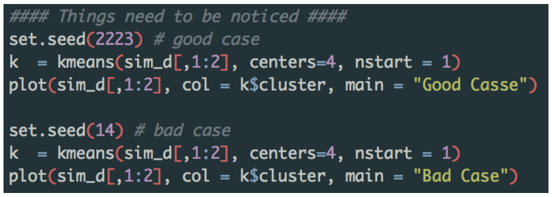
$$d_j(x_{ij}, x_{i'j}) = (x_{ij} - x_{i'j})^2.$$

$$d(x_i, x_{i'}) = \sum_{j=1}^{p} (x_{ij} - x_{i'j})^2 = ||x_i - x_{i'}||^2$$

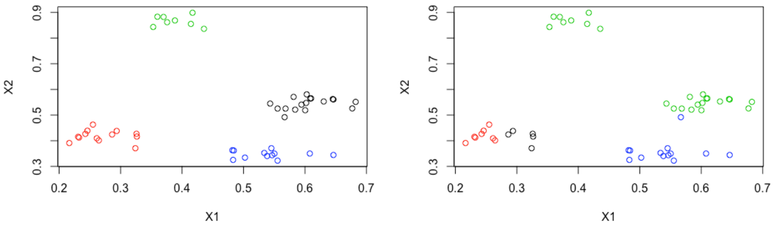


K-means Clustering

- **Things Need To Be Noticed**
  1. You won’t always get the same results. So, do it more times!

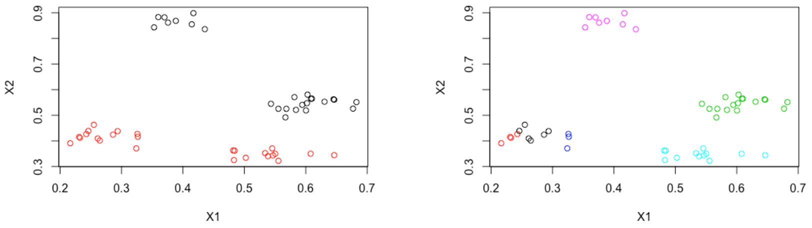


```
#### Things need to be noticed ####
set.seed(2223) # good case
k  = kmeans(sim_d[,1:2], centers=4, nstart = 1)
plot(sim_d[,1:2], col = k$cluster, main = "Good Casse")

set.seed(14) # bad case
k  = kmeans(sim_d[,1:2], centers=4, nstart = 1)
plot(sim_d[,1:2], col = k$cluster, main = "Bad Case")
```



K-means Clustering

- **Things Need To Be Noticed**
  2. How to choose the correct number of clusters ?

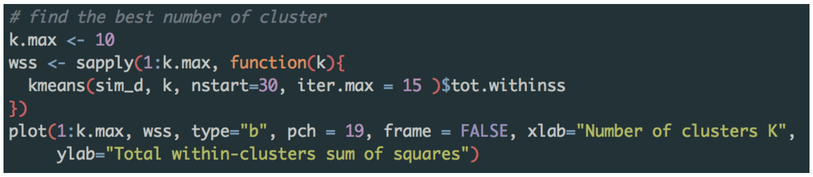


$$\sum(x_i - \bar{x})^2 = \sum(x_i - \bar{x_k})^2 + \sum(\bar{x_k} - \bar{x})^2$$

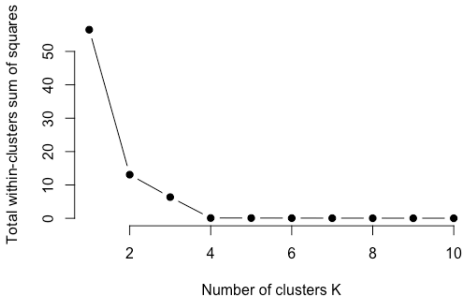
TOTAL Sum of Squares = **WITHIN** Groups SS + **BETWEEN** Groups SS
(**TOTAL** variance = **WITHIN** variance + **BETWEEN** variance)

K-means Clustering

- An Intuitive Method for finding the best k

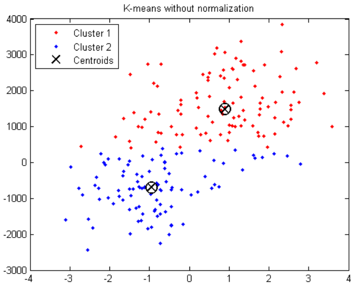


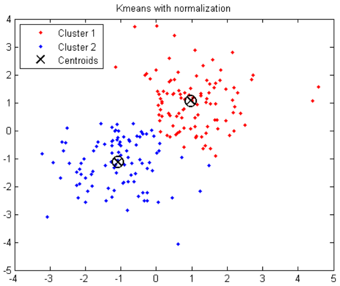
```
# find the best number of cluster
k.max <- 10
wss <- sapply(1:k.max, function(k){
  kmeans(sim_d, k, nstart=30, iter.max = 15 )$tot.withinss
})
plot(1:k.max, wss, type="b", pch = 19, frame = FALSE, xlab="Number of clusters K",
     ylab="Total within-clusters sum of squares")
```



K-means Clustering

- **Things Need To Be Noticed**
  3. Standardize the data or not?

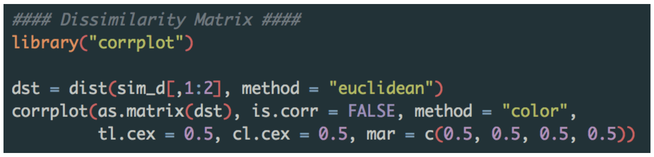




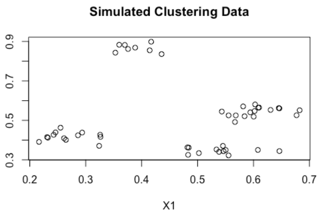
The source of pics : here

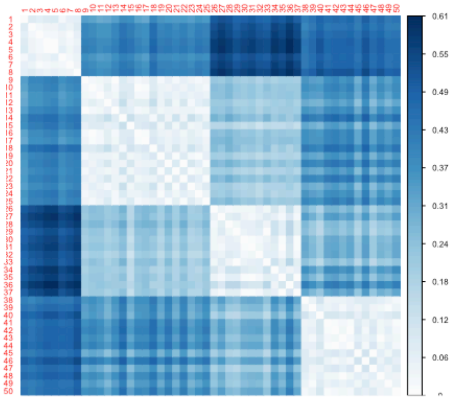
Hierarchical Clustering

- Dissimilarity Matrix



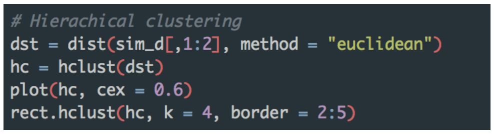
```
#### Dissimilarity Matrix ####
library("corrplot")

dst = dist(sim_d[,1:2], method = "euclidean")
corrplot(as.matrix(dst), is.corr = FALSE, method = "color",
         tl.cex = 0.5, cl.cex = 0.5, mar = c(0.5, 0.5, 0.5, 0.5))
```



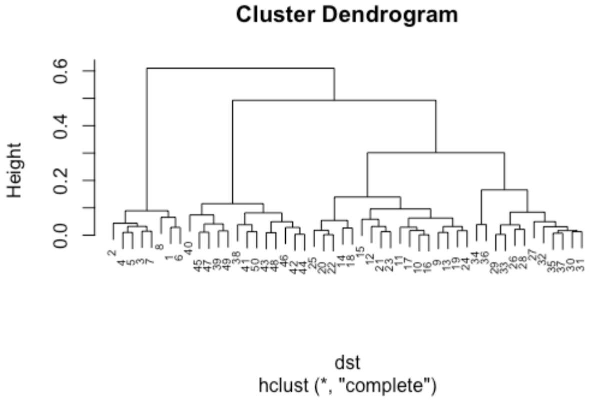


Hierarchical Clustering

- Algorithm

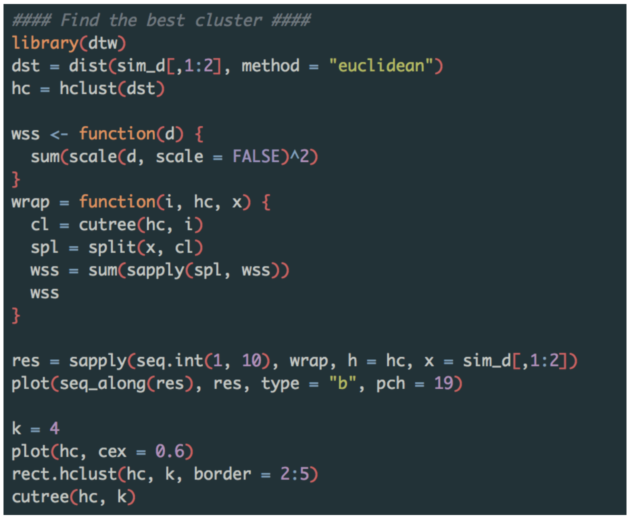


```
# Hierachical clustering
dst = dist(sim_d[,1:2], method = "euclidean")
hc = hclust(dst)
plot(hc, cex = 0.6)
rect.hclust(hc, k = 4, border = 2:5)
```



Hierarchical Clustering

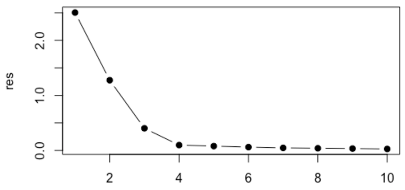
- Find the Best k

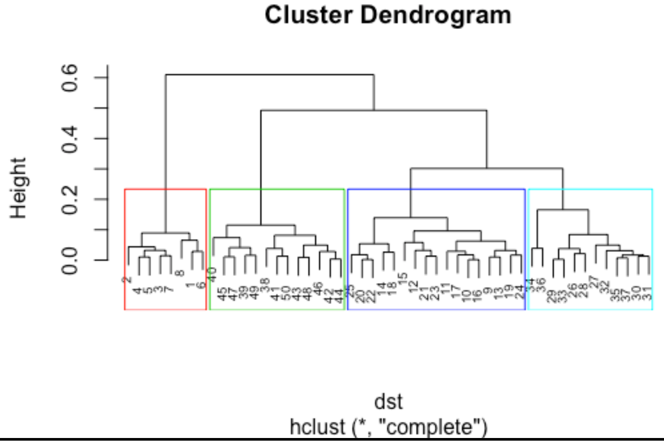


```
#### Find the best cluster ####
library(dtw)
dst = dist(sim_d[,1:2], method = "euclidean")
hc = hclust(dst)

wss <- function(d) {
  sum(scale(d, scale = FALSE)^2)
}
wrap = function(i, hc, x) {
  cl = cutree(hc, i)
  spl = split(x, cl)
  wss = sum(sapply(spl, wss))
  wss
}

res = sapply(seq.int(1, 10), wrap, h = hc, x = sim_d[,1:2])
plot(seq_along(res), res, type = "b", pch = 19)

k = 4
plot(hc, cex = 0.6)
rect.hclust(hc, k, border = 2:5)
cutree(hc, k)
```

Hierarchical Clustering

- Find the Best k





Hierarchical Clustering

- **Things Need To Know**

1. No need to specify the number of clusters. One Dendrogram can show many clusters.

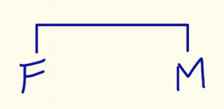
2. It assumes that all the clusters are nested with a hierarchical structure, but it may not be the case.

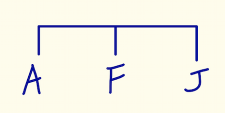
   Ex:

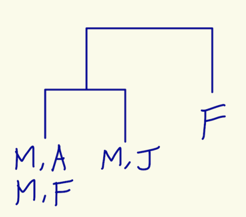
   x1 = { Male, Female }, x2 = { Americans, Japanese, French }

   Best 2 clusters : by Gender

   Best 3 clusters : by Nationality

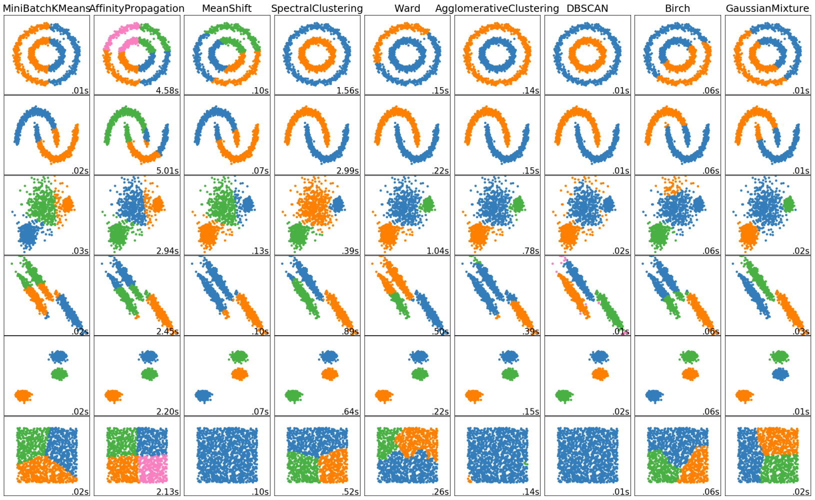






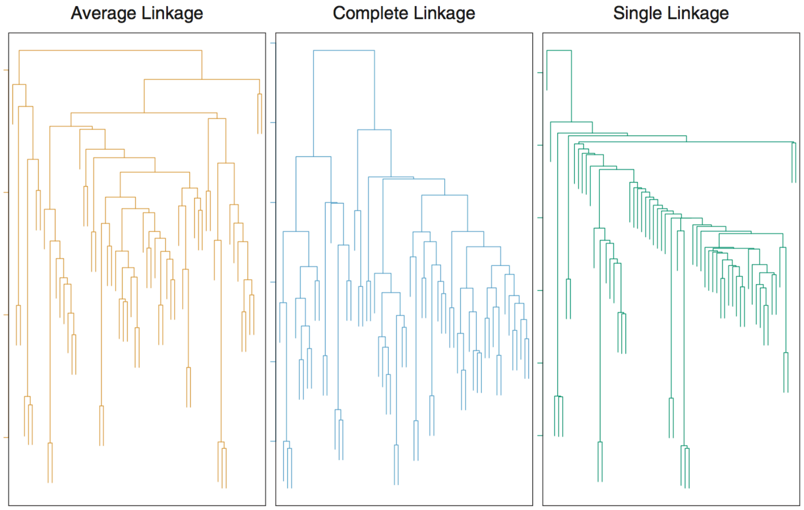
Further Topics

The source of the graph : here



Further Topics

The source of the graph : here



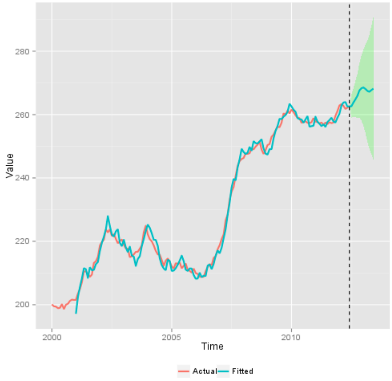
Time Series Data

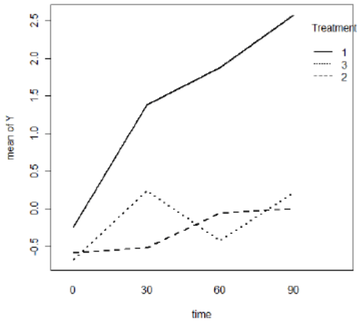
- **General Definition**

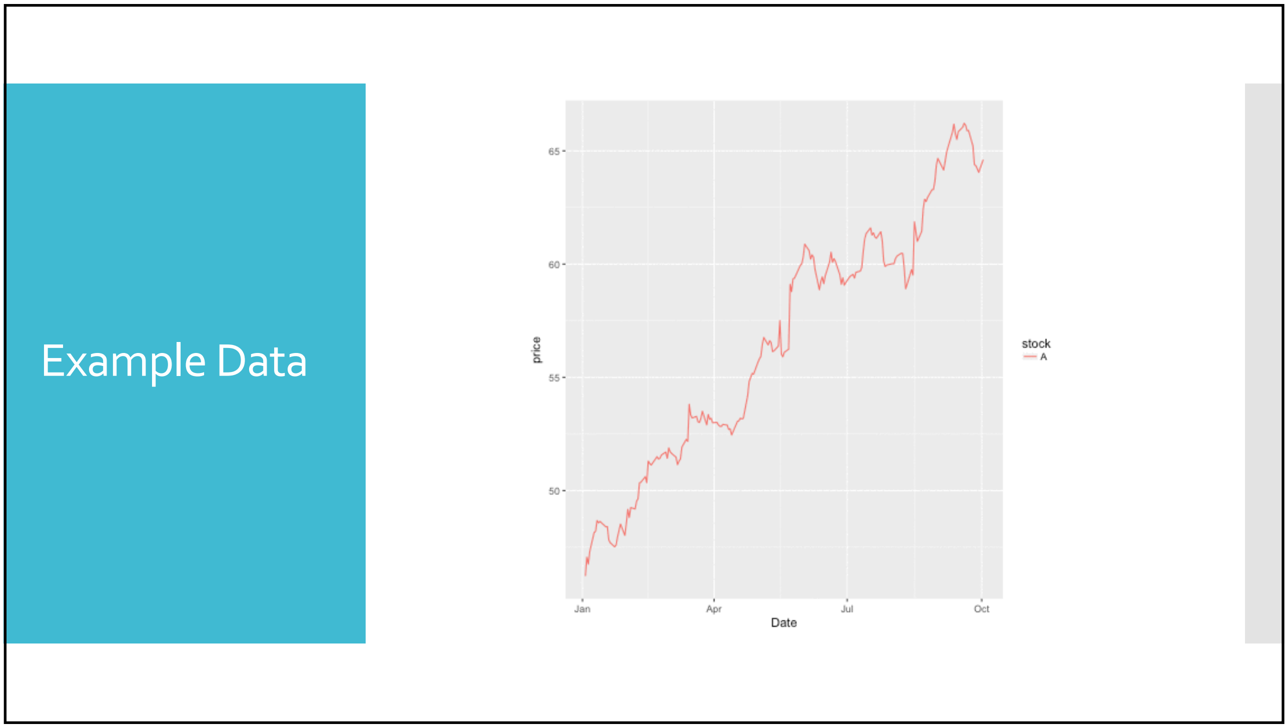
  The data is taken by time and believe that time will be an influential factor to the data.

  Ex : Sales data ,

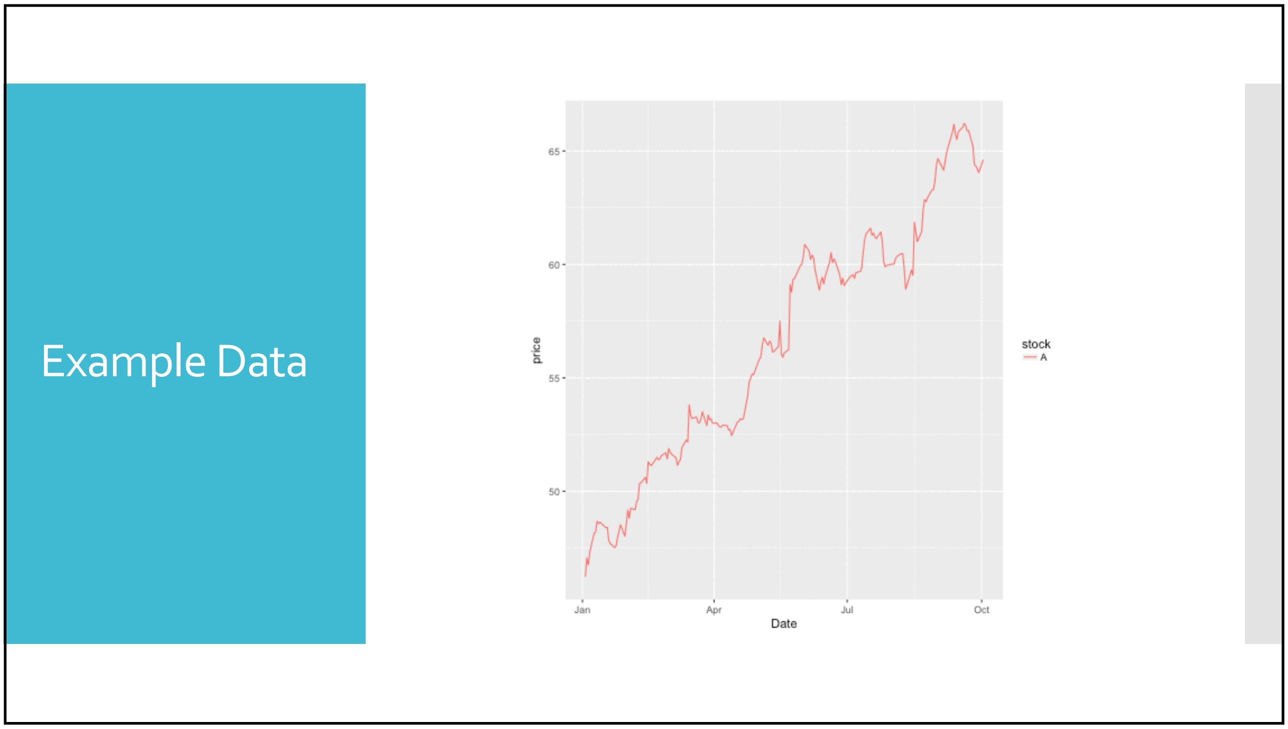
  Difference in temperature under different treatments





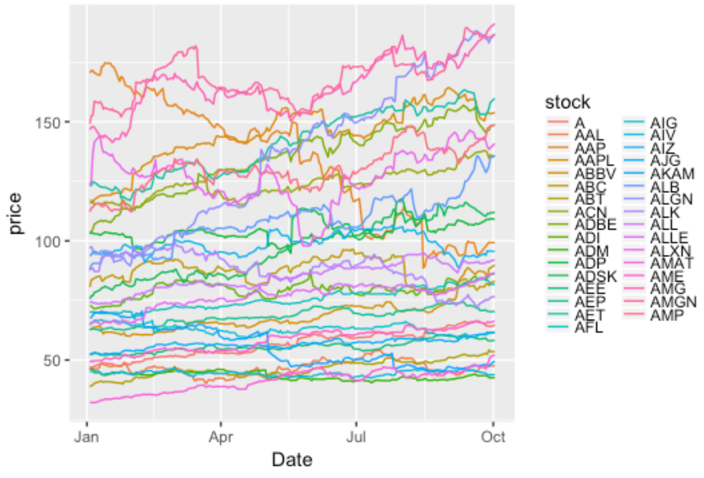


Example Data



Example Data

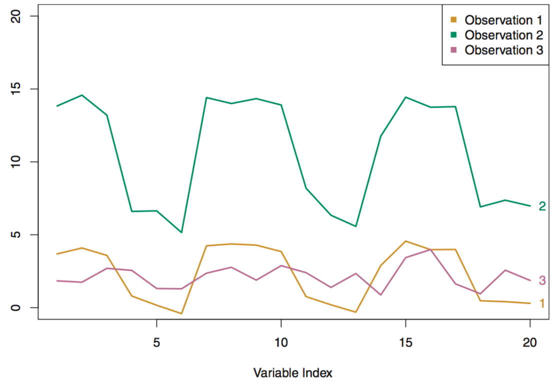
Example Data



Distance between Time Series

- **Euclidean Distance may not be what we want.**

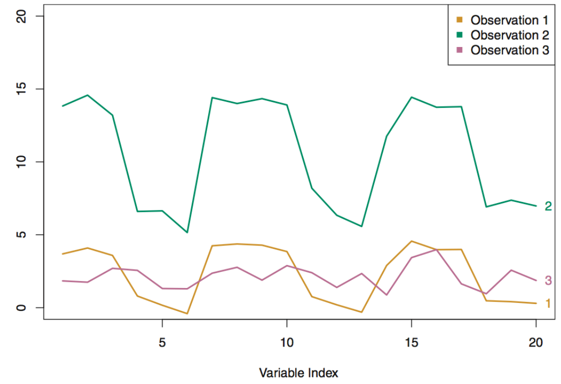
(The source of this pic : here)



Distance between Time Series

- **Think about Correlation**

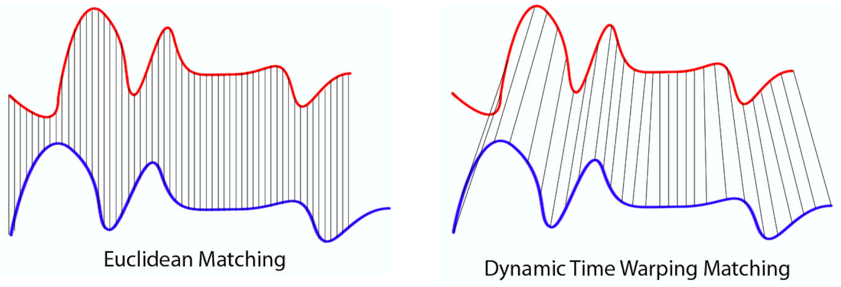
$$\rho(x_i, x_{i'}) = \frac{\sum_j (x_{ij} - \bar{x}_i)(x_{i'j} - \bar{x}_{i'})}{\sqrt{\sum_j (x_{ij} - \bar{x}_i)^2 \sum_j (x_{i'j} - \bar{x}_{i'})^2}}$$



Distance between Time Series

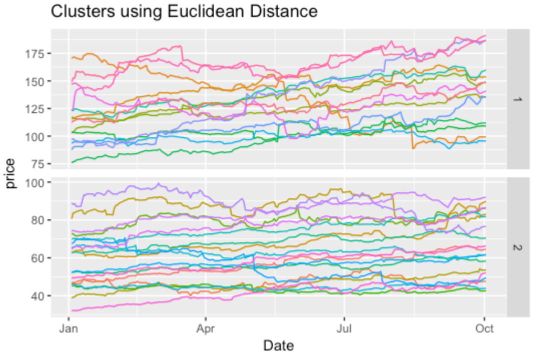
- **Think about Dynamic Time Warping (DTW) Distance**

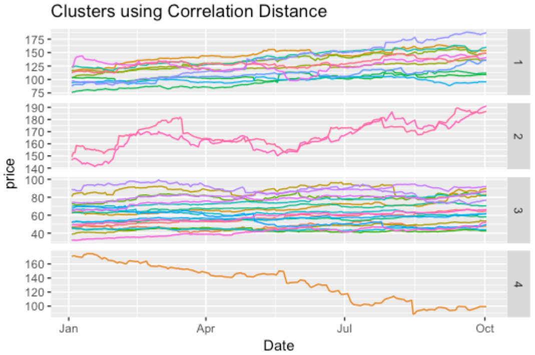
(The source of this pic : here)

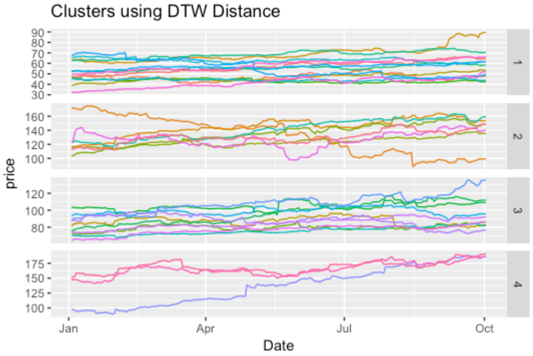

Euclidean Matching

Dynamic Time Warping Matching

Results

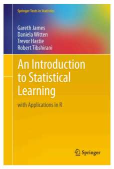
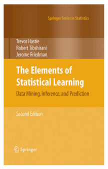

Clusters using Euclidean Distance


Clusters using Correlation Distance

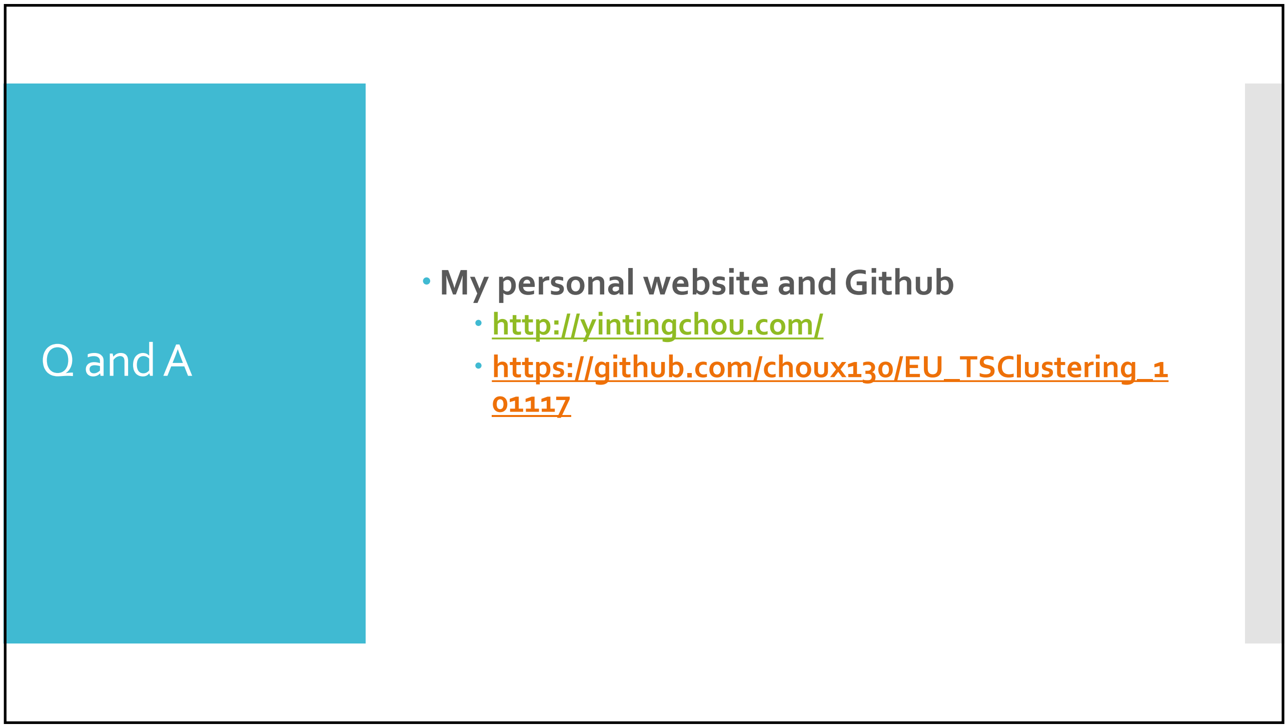

Clusters using DTW Distance

References and Resources

- **Books**
  - An Introduction to Statistical Learning with Applications in R
  - The Elements of Statistical Learning: Data Mining, Inference and Prediction

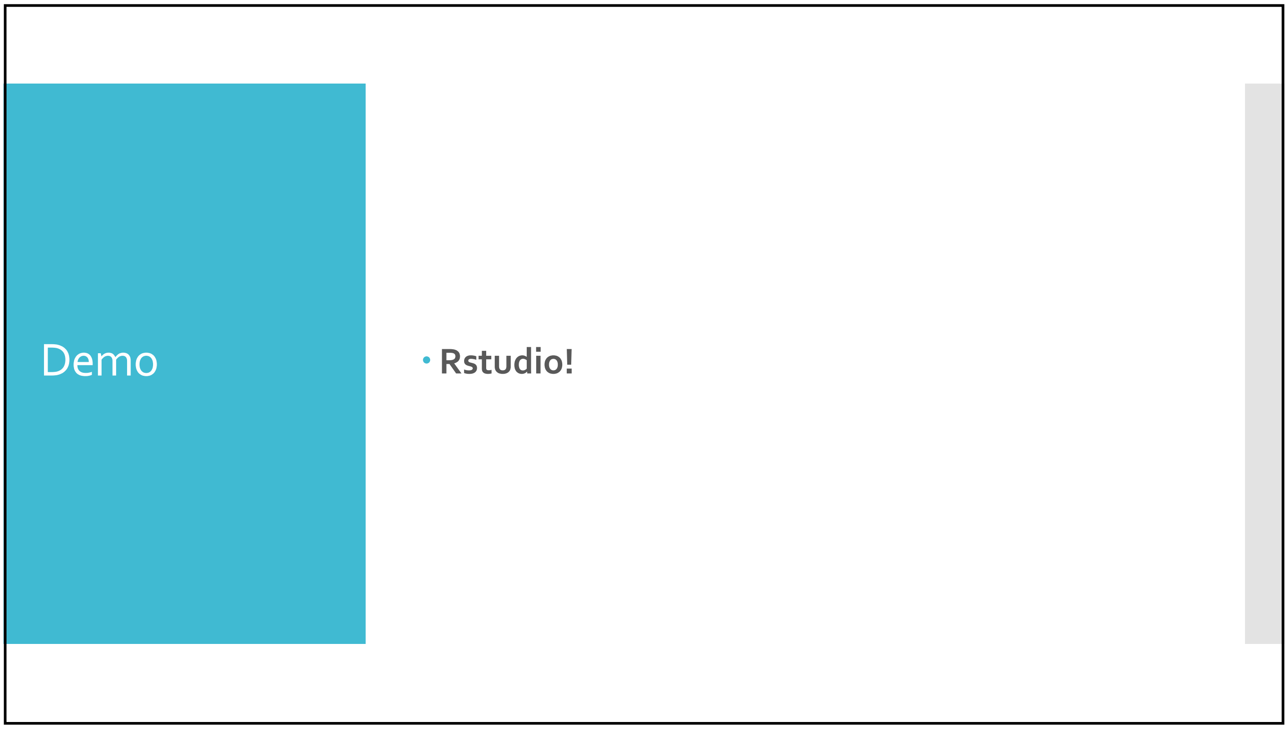



- **R Packages**
  - MixSim – Generate clustering data
  - corrplot – Plot the dissimilarity matrix
  - tidyverse – Data Manipulation
  - ggplot2 – Data Visualization
  - amap – K-means with Correlation Distance



Q and A

- My personal website and Github
  - http://yintingchou.com/
  - https://github.com/choux130/EU_TSClustering_101117



Demo

- Rstudio!