# 비지도학습

김정은



Growth Hackers

# CONTENTS

# 비지도학습 ?

- **지도 학습 (Supervised Learning)**
  - 이미 라벨이 존재하는 데이터를 모델을 통해 학습
  → 새로운 데이터의 라벨을 예측

- **비지도 학습 (Unsupervised Learning)**
  - 데이터를 분류하는 라벨이 존재하지 않음
  - 데이터에 내재된 특성을 분석하여 유사한 데이터를 구별하거나 묶는 과정

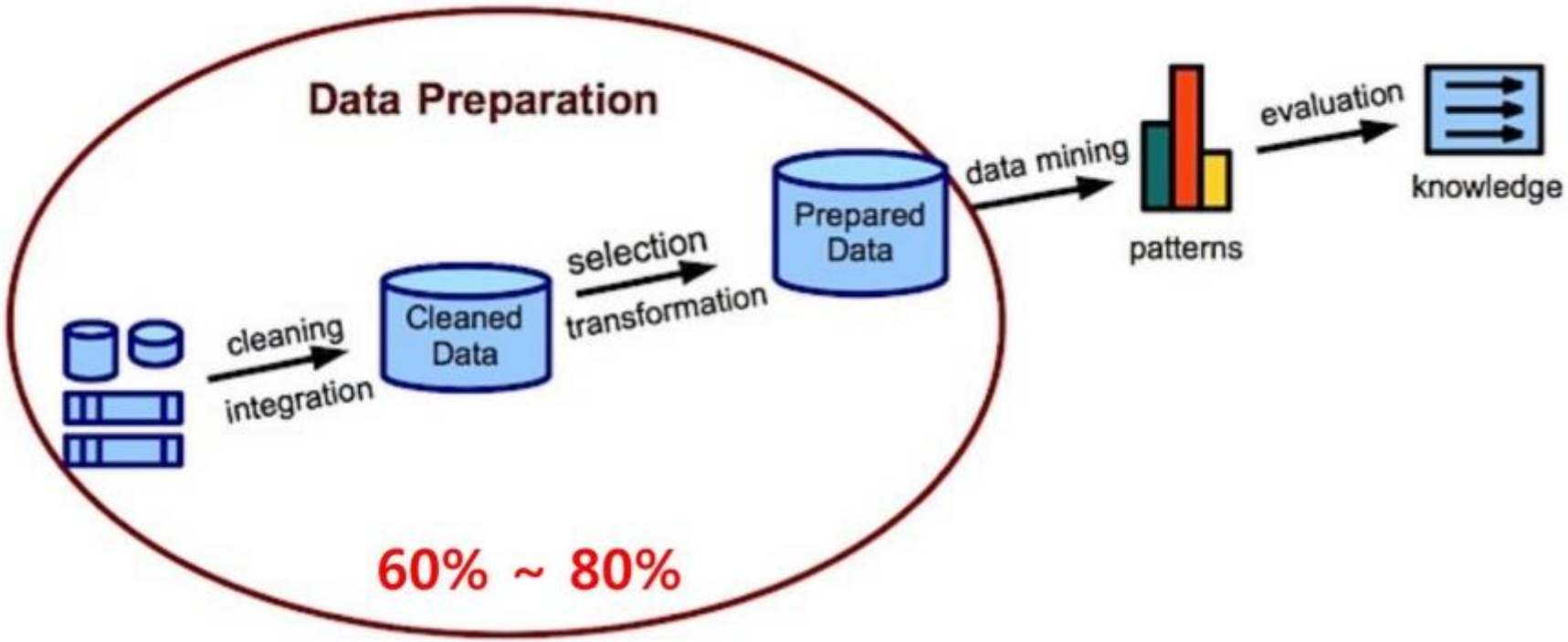
| | | |
|---|---|---|
| 지도학습 | Classification | kNN |
| | | Naïve Bayes |
| | | Support Vector machine |
| | | Decision Tree |
| | Regression | Linear regression |
| | | Locally weighted linear regression |
| | | Ridge |
| | | Lasso |
| 비지도학습 | | Clustering |
| | | K means |
| | | Density estimation |
| | | Expectation maximization |
| | | Pazen window |
| | | DBSCAN |



Growth Hackers

# 데이터 전처리

- 분석 및 처리에 적합한 형식으로 데이터를 조작하는 것






Growth Hackers

# 데이터 전처리

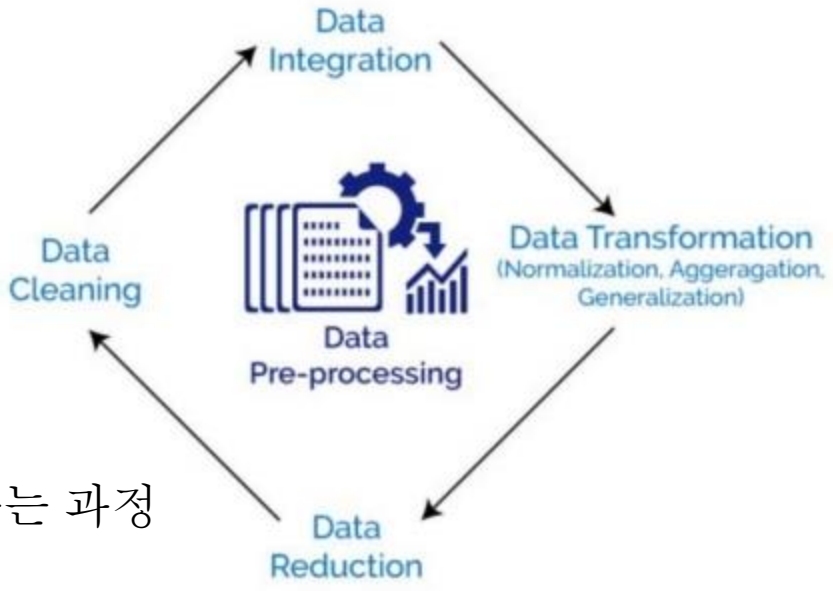
- 분석 및 처리에 적합한 형식으로 데이터를 조작하는 것

**데이터 정제**
- 이상치 , 결측값 검색 , 수정 및 제거
- 데이터의 신뢰도를 높이는 과정

**데이터 통합**
- 데이터 , 스키마 통합
- 여러 소스의 데이터를 통합하는 과정

**데이터 변환**
- 데이터 요약 , 집계 작업
- 노이즈 제거 , 새로운 속성 추가 등
- 데이터 정규화
- 효과적인 분석을 위해 데이터를 변환 및 변형하는 과정

**데이터 정리**
- 데이터 크기 축소






Growth Hackers

# 데이터 전처리

- Feature scaling

Scaling(min-max scaling)

$$x' = \frac{x - x_{min}}{x_{max} - x_{min}}$$

- 서로 다른 피처를 범주의 값을 같게함
- 데이터 포인트 간의 거리가 중요한 분석에서 주로 쓰임 (e.g. SVM, kNN)
- Outlier 에 주의해야 함

Standardization

$$x' = \frac{x - x_{mean}}{\sigma}$$

- 데이터의 피처를 평균이 0이고 분산이1인 가우시안 정규 분포를 가진 값으로 변환
- Linear regression, Logistic regression, Linear discriminate analysis.

# 데이터 전처리– 실습

Data Preprocessing Ex.ipynb



Growth Hackers

# 차원 축소

- 차원 축소 ?
  - 고차원의 데이터 → 데이터 간의 관계를 설명할 수 있는 중요한 차원으로 변환

- 차원 축소의 필요성
  - 차원 : 공간 내에 있는 점 등의 위치를 나타내기 위해 필요한 축의 개수
  - 변수의 수가 늘어난다 = 차원이 늘어난다 = 데이터 공간이 커진다

    **= 분석을 위해 필요한 최소한의 데이터 건수가 많아진다**

- 차원 축소의 효과
  - 1) 차원의 저주 탈피 ( 차원이 증가하면 데이터를 표현하기 위한 공간은 기하급수적으로 커지고 그로 인해 차원이 낮을 때 없었던 문제 ( 신뢰도 및 정확도 감소 , 러닝 타임 증가 ) 가 발생한다 )
  - 2) 시각화의 용이성



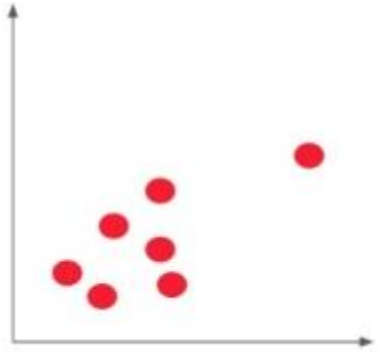
Growth Hackers

# 차원 축소

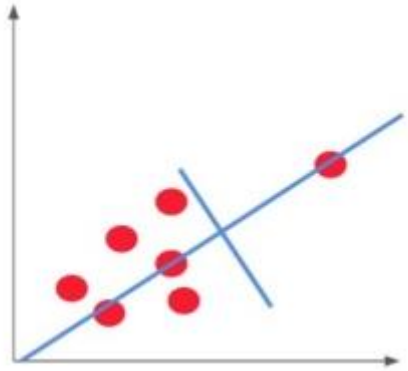
- 차원 축소 방법
  - 1) Feature Selection
    - 가지고 있는 여러 변수들 중 중요한 것을 고르기
    - 분석 주제 : 변수 간에 중첩이 있는가 ? 어떤 변수가 중요한가 ? 어떤 변수가 타깃에 큰 영향을 주는가 ?
    - 분석 방법 : 상관 분석 (Correlation) / VIF( 분산팽창지수 , Variance Inflation Factor) 분석 / Random Forest, XGBoost 등을 이용한 Variable importance 분석 / 등 ..
  - 2) Feature Extraction
    - 모든 변수를 조합하여 전체 데이터를 잘 표현할 수 있는 중요 성분을 가진 새로운 변수 추출
    - 분석 방법 : 주성분분석 (Principle Component Analysis) / TSNE (T-Distributed Stochastic Neighbor Embedding) / 비음수 행렬 분해 (NMF) / 등 ..

# 차원 축소

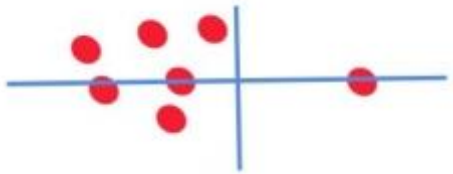
## 주성분분석 (Principle Component Analysis)



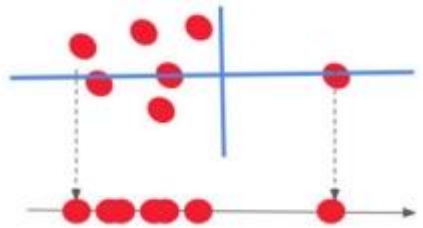
- 원본 데이터 !



- 데이터의 변화의 폭이 가장 큰 축 & 그것과 직교하는 축 찾기



- 축의 방향과 위치를 전환시켜 데이터를 균등하게 분포시키기



- 1 차원으로도 축소할 수 있음 !



Growth Hackers

# 차원 축소

## 주성분분석 (Principle Component Analysis)

1. 입력 데이터 세트의 공분산 행렬 생성
2. 공분산 행렬의 고유 벡터와 고유 값 계산
3. 고유값이 가장 큰 순으로 K개만큼 고유벡터 추출
4. 고유값이 가장 큰 순으로 추출된 고유벡터 이용하여 새롭게 입력 데이터 반환

# 차원 축소

## 주성분분석 (Principle Component Analysis)

PCA
- C : covariance matrix of x
- $C = P\Sigma P^T$ (P: orthogonal, Σ: diagonal)

$$C = \begin{pmatrix} e_1 & \cdots & e_n \end{pmatrix} \begin{pmatrix} \lambda_1 & & 0 \\ & \ddots & \\ 0 & & \lambda_n \end{pmatrix} \begin{pmatrix} e_1^T \\ \vdots \\ e_n^T \end{pmatrix}$$

- P : n×n orthogonal matrix
- Σ : n×n diagonal matrix
- $Ce_i = \lambda_i e_i$
  - $e_i$ : eigenvector of C, direction of variance
  - $\lambda_i$ : eigenvalue, $e_i$ 방향으로의 분산
  - $\lambda_1 \geq \ldots \geq \lambda_n \geq 0$
- $e_1$: 가장 분산이 큰 방향
- $e_2$: $e_1$에 수직이면서 다음으로 가장 분산이 큰 방향
- $e_k$: $e_1, \ldots, e_{k-1}$에 모두 수직이면서 가장 분산이 큰 방향

고유벡터 | 고유값 | T(고유벡터)

$$C = \begin{pmatrix} e_1 & \cdots & e_n \end{pmatrix} \begin{pmatrix} \lambda_1 & & 0 \\ & \ddots & \\ 0 & & \lambda_n \end{pmatrix} \begin{pmatrix} e_1^T \\ \vdots \\ e_n^T \end{pmatrix}$$



Growth Hackers

# 차원 축소

## 주성분분석 (Principle Component Analysis)

- K값 선택

$$\begin{bmatrix} 2.7596 & 0 & 0 \\ 0 & 0.1618 & 0 \\ 0 & 0 & 0.0786 \end{bmatrix}$$

$$\frac{\lambda_1}{\lambda_1 + \lambda_2 + \lambda_3} = \frac{2.7596}{2.7596 + 0.1618 + 0.0786} = 0.920$$

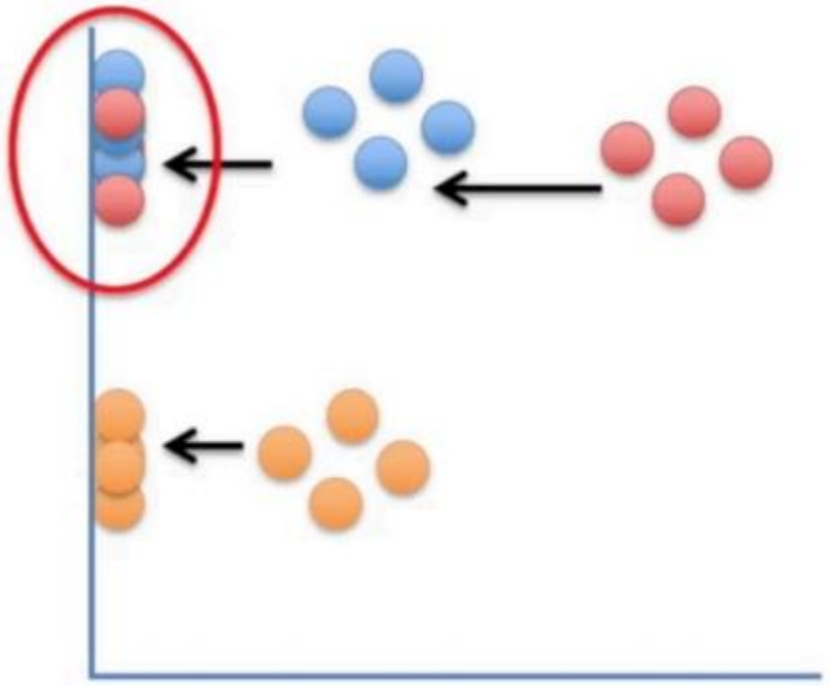


Growth Hackers

# 차원 축소

## 주성분분석 (Principle Component Analysis)

- 문제점





Growth Hackers

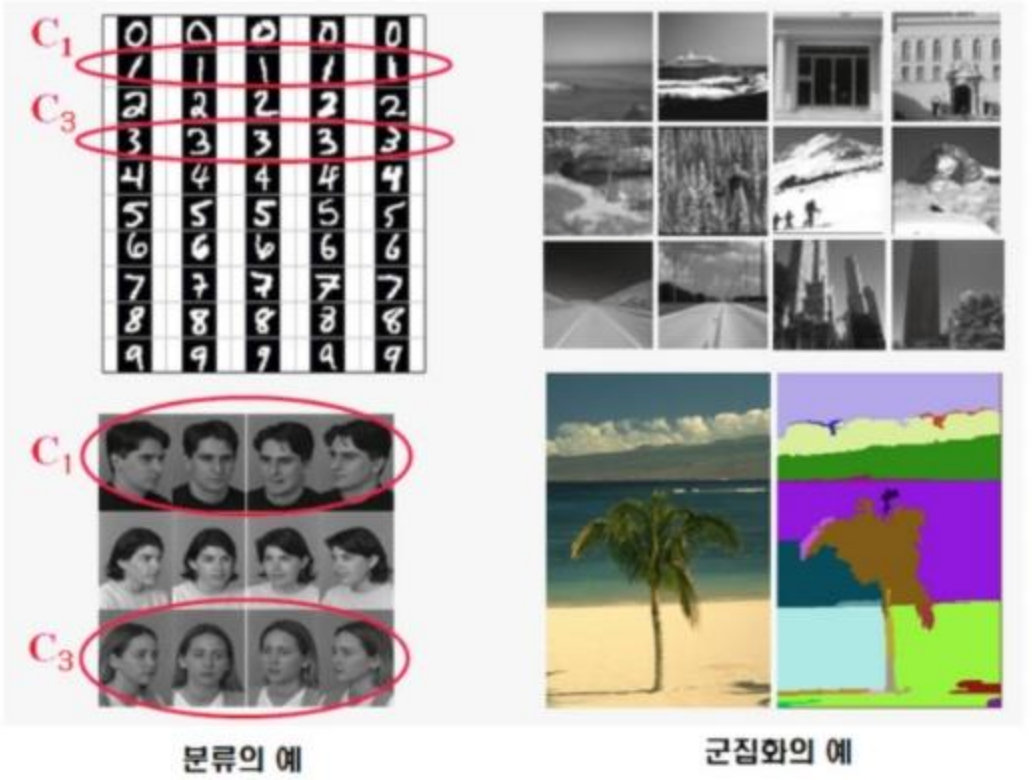
# 차원축소– 실습

PCA Ex.ipynb

# Classification & Clustering

Goal : 유사한 데이터를 같은 그룹으로 묶는 모델 생성,
새로운 instance의 그룹 예측

| 분류(classification) | 군집화(clustering) |
| --- | --- |
| 주어진 데이터 집합을 이미 정의된 몇 개의 클래스로 구분하는 문제 | 입력 데이터의 분포 특성(입력값의 유사성)을 분석하여 임의의 복수 개의 그룹으로 나누는 것 |
| 입력 데이터와 각 데이터의 클래스 라벨이 함께 제공 -> $\{x_i, y(x_i)\}$ | 클래스에 대한 정보 없이 단순히 입력값만 제공 ->$\{x_i\}$ |
| 숫자인식, 얼굴인식 등 | 영상분리, market segmentation |
| K-Nearest Neighbor<br>Support Vector Machine<br>Bayes Classifier | K-means clustering<br>Hierarchical clustering<br>Gaussian clustering |

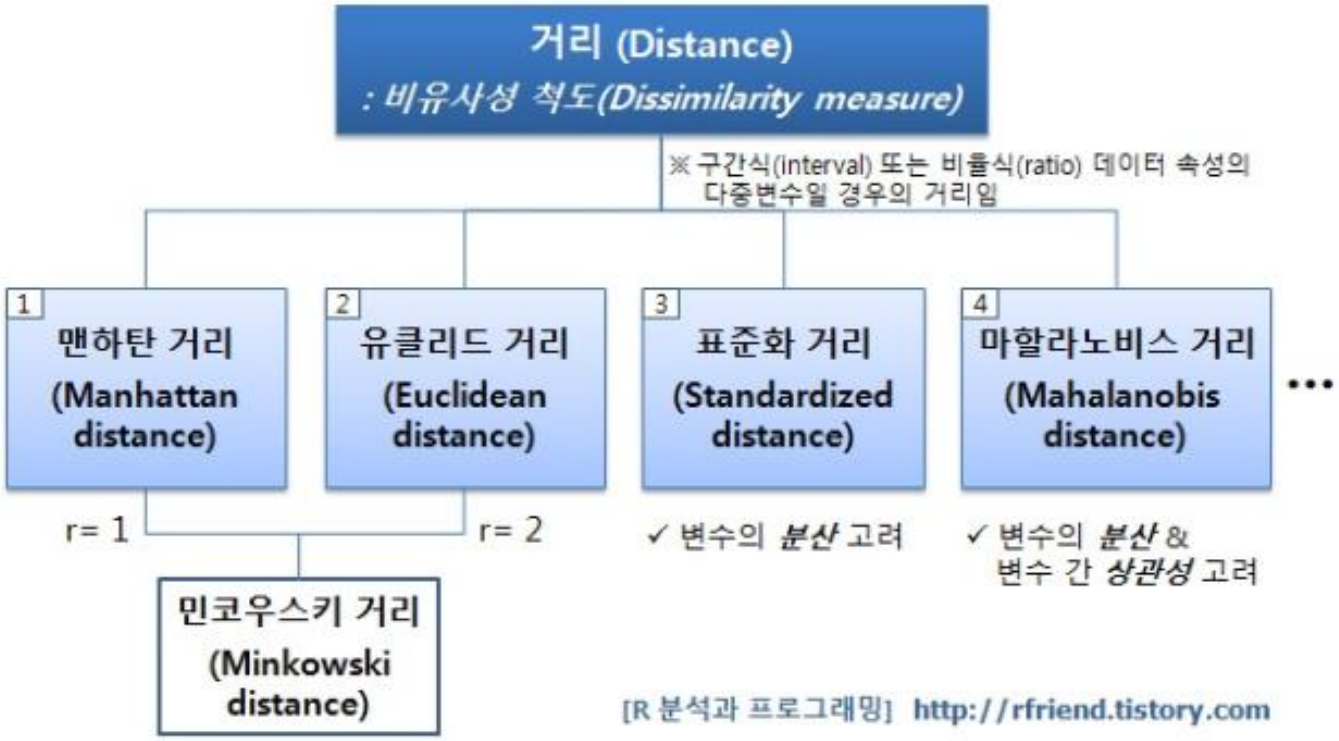



분류의 예 군집화의 예

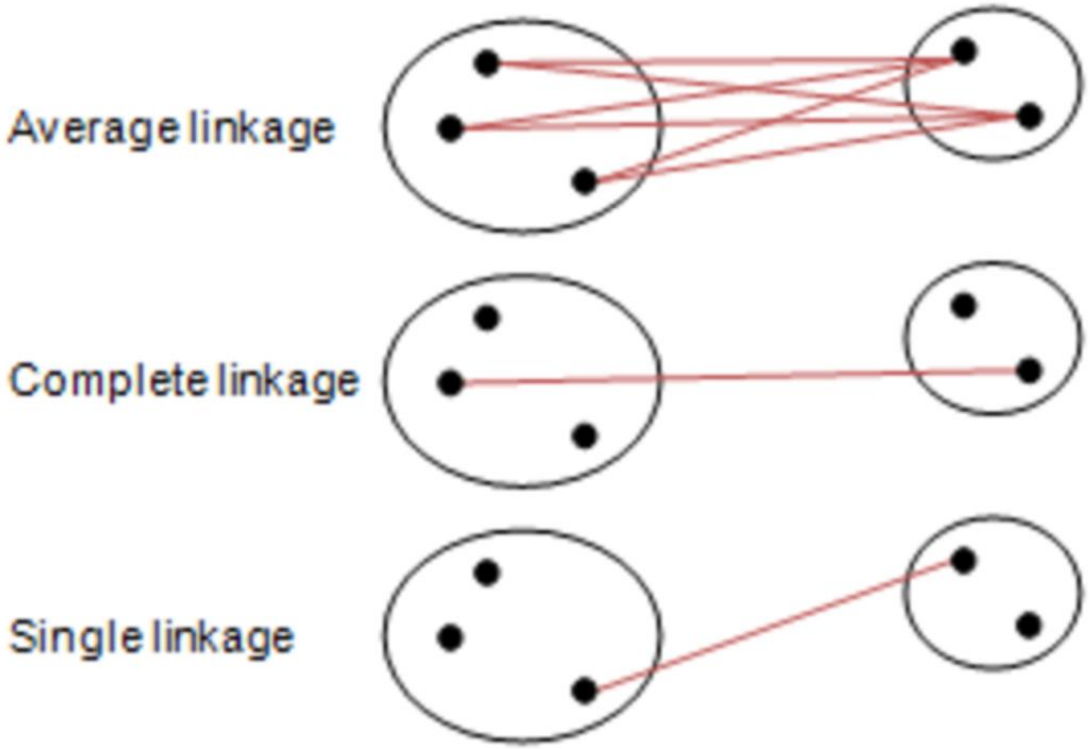


Growth Hackers

# Clustering_Hierarchical



# Clustering_Hierarchical

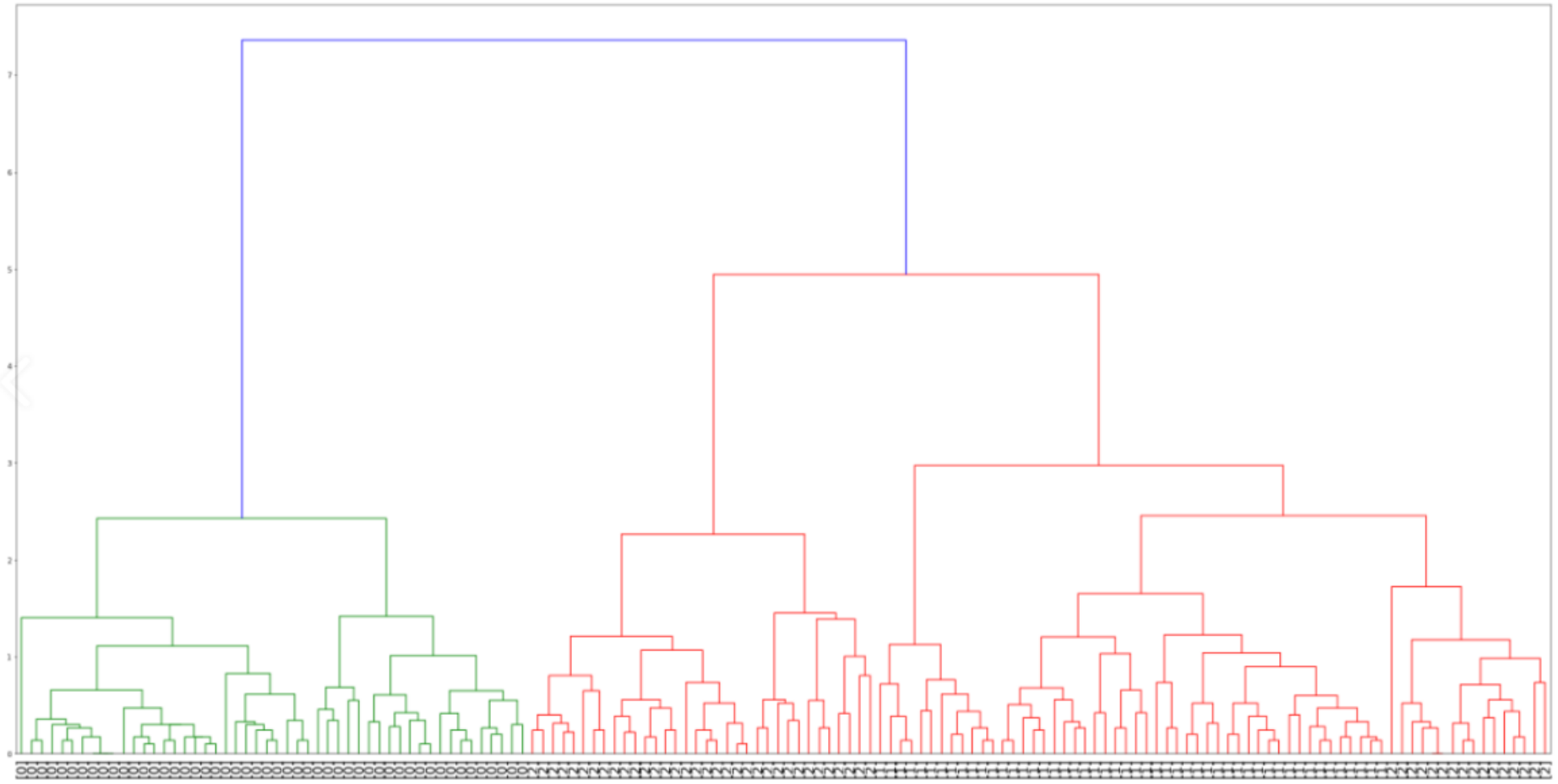



두 그룹간의 평균거리

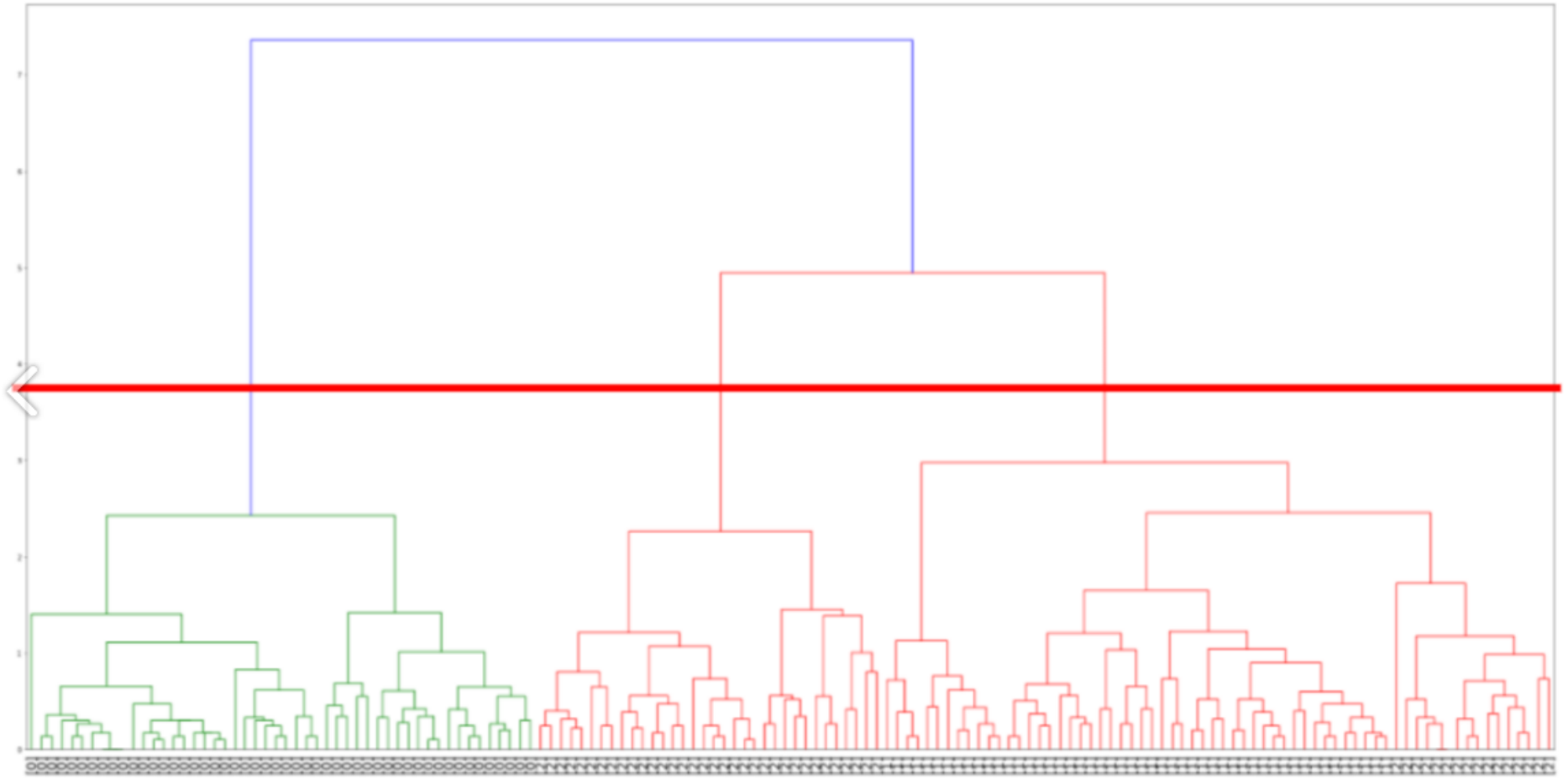
두 그룹간의 최대거리

두 그룹간의 최소거리

# Clustering_Hierarchical

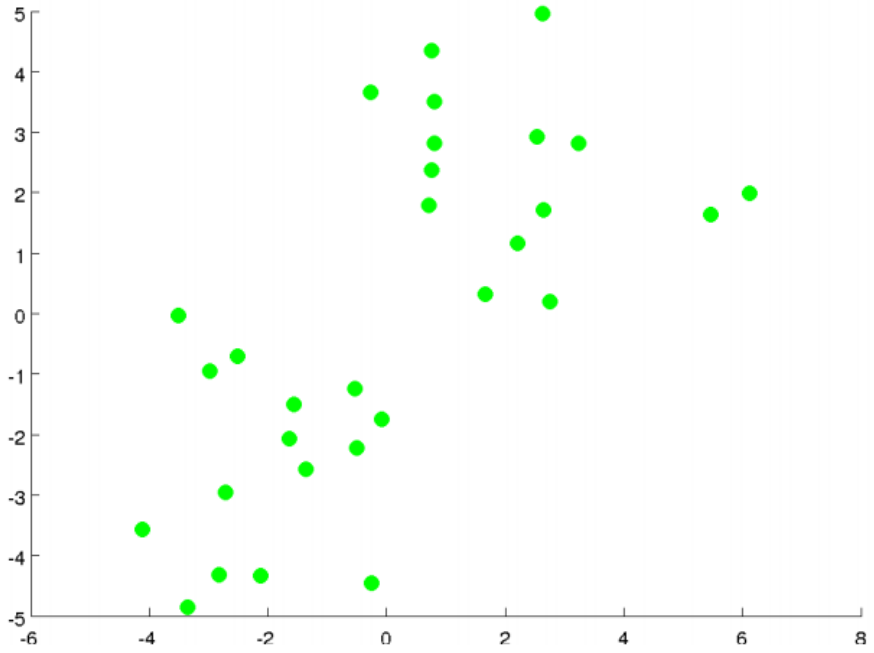


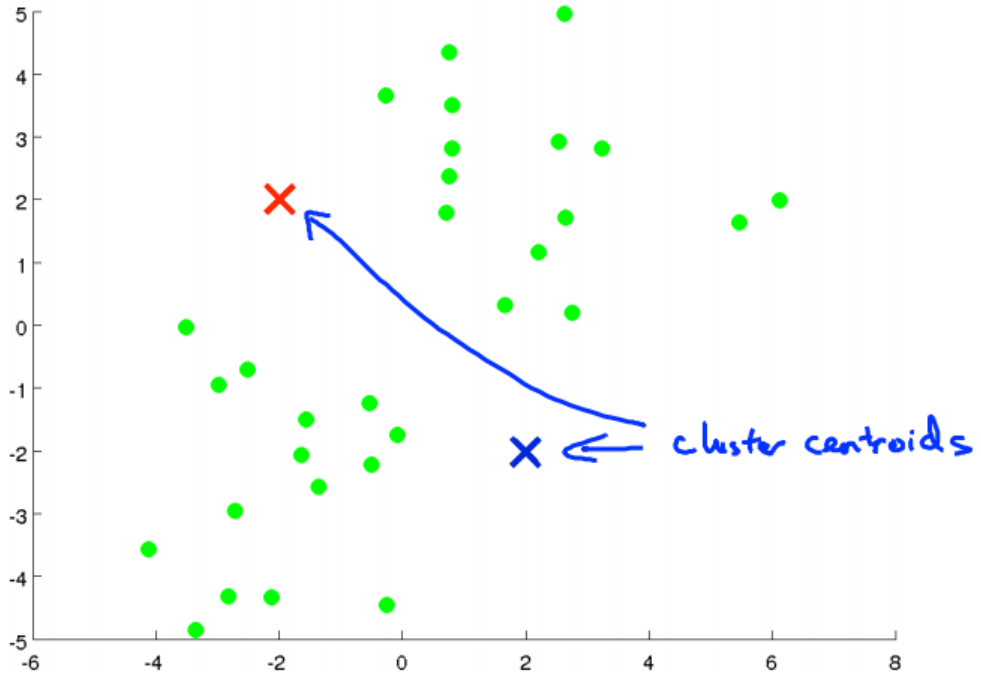
# Clustering_Hierarchical



# Clustering_ k-means

1. 임의로 점 k 개를 찍고, 각 군집의 중심점으로 잡는다.



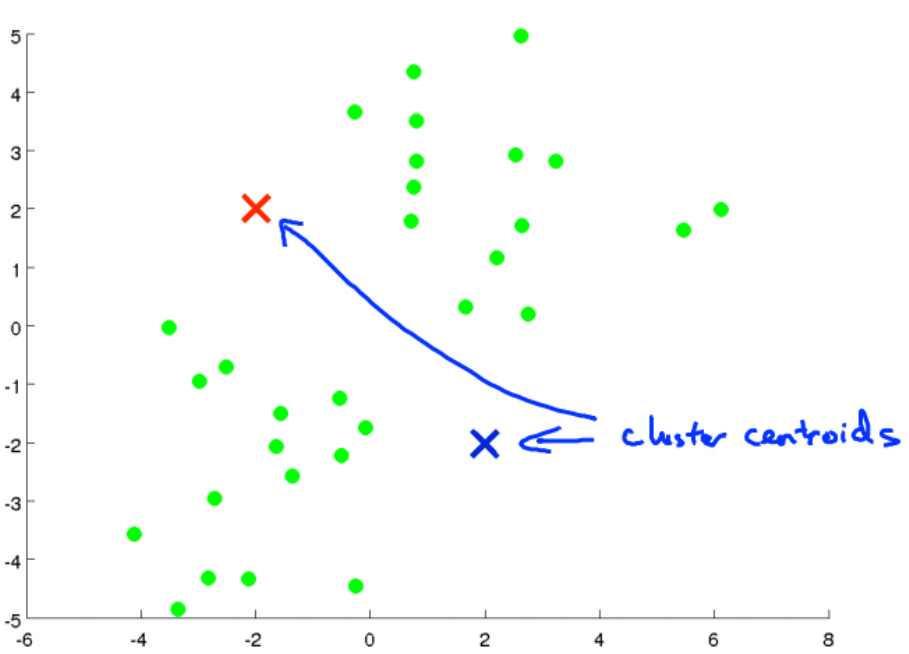



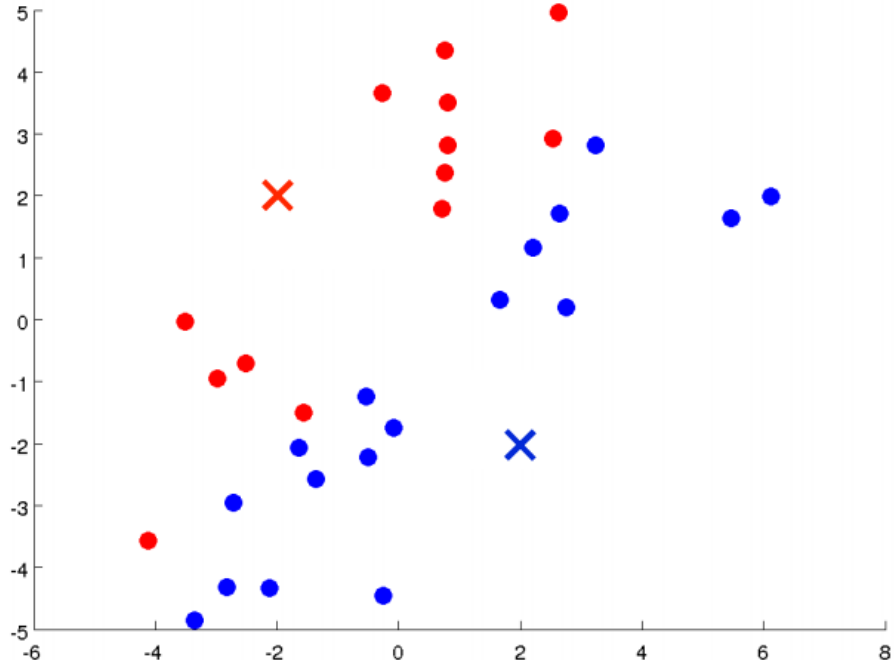



Growth Hackers

# Clustering_ k-means

2. 각 중심점과 데이터의 거리를 재서 가장 가까운 군집에 배정한다



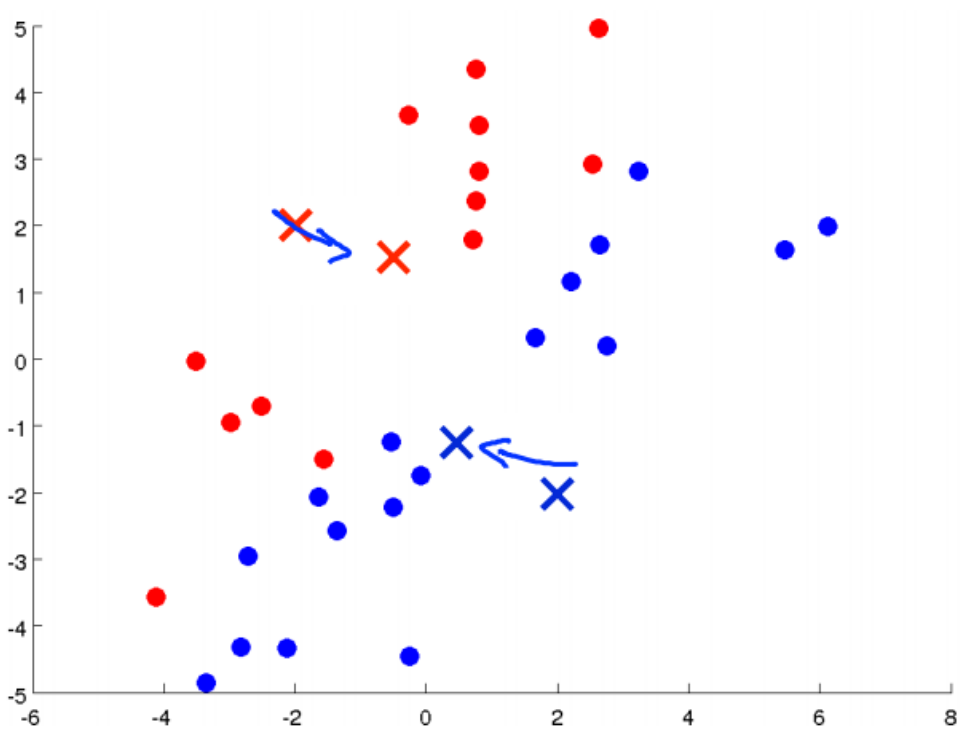



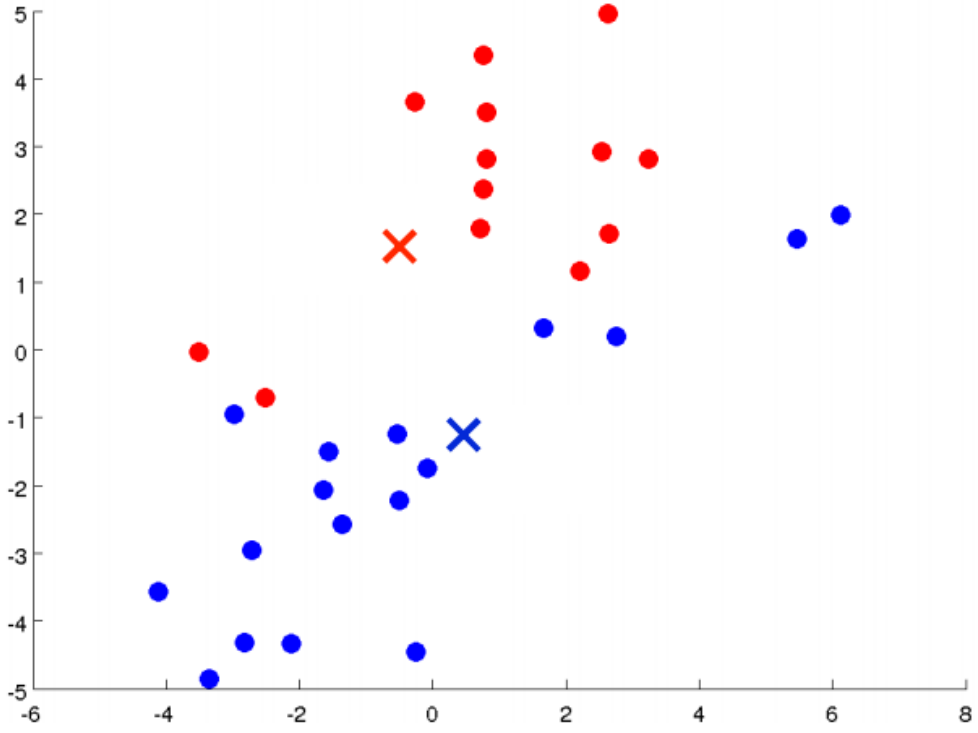


Growth Hackers

# Clustering_ k-means

3. 배정된 군집이 이전 배정과 하나라도 다르면 , 배정된 군집 내의 평균을 계산해서 새로운 중심점으로 잡는다 . → 2 단계



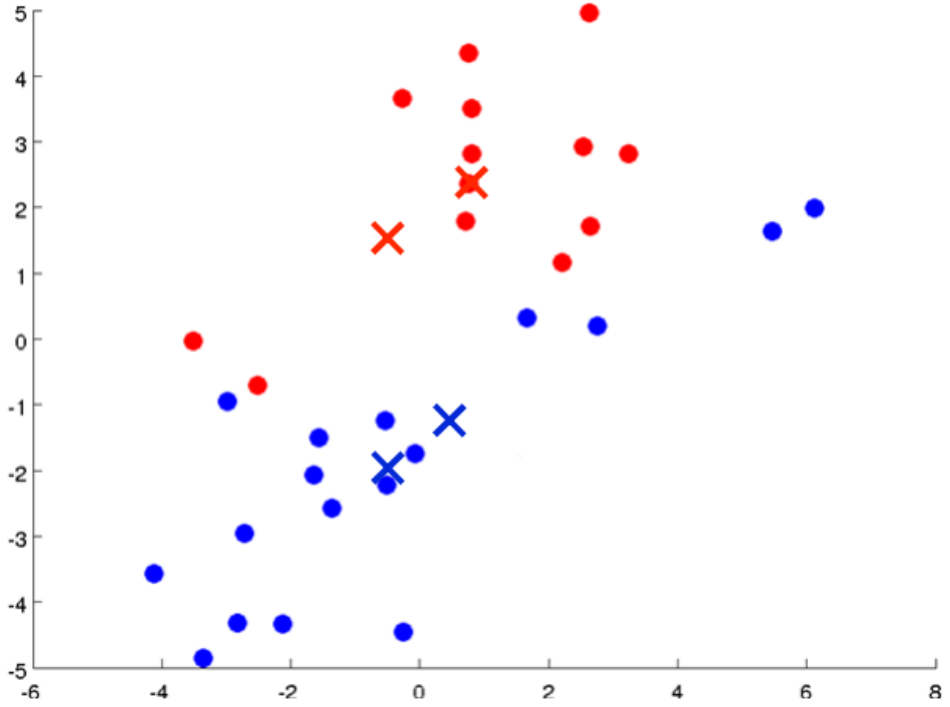



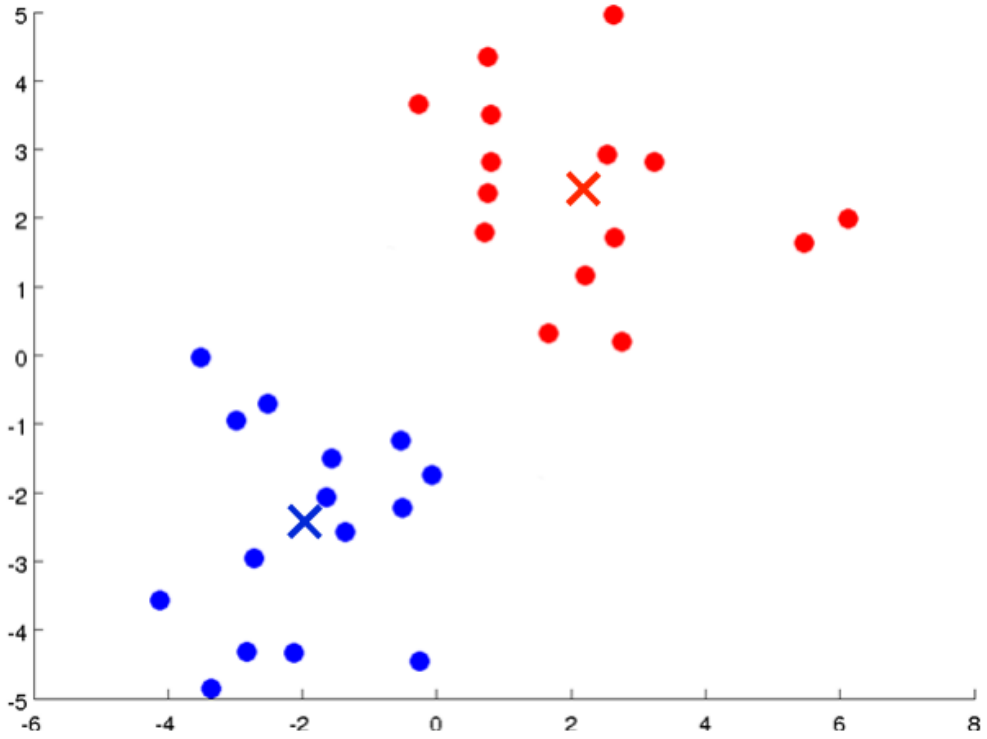



Growth Hackers

# Clustering_ k-means

4. 배정된 군집이 이전 배정과 같다면 , 알고리즘 종료

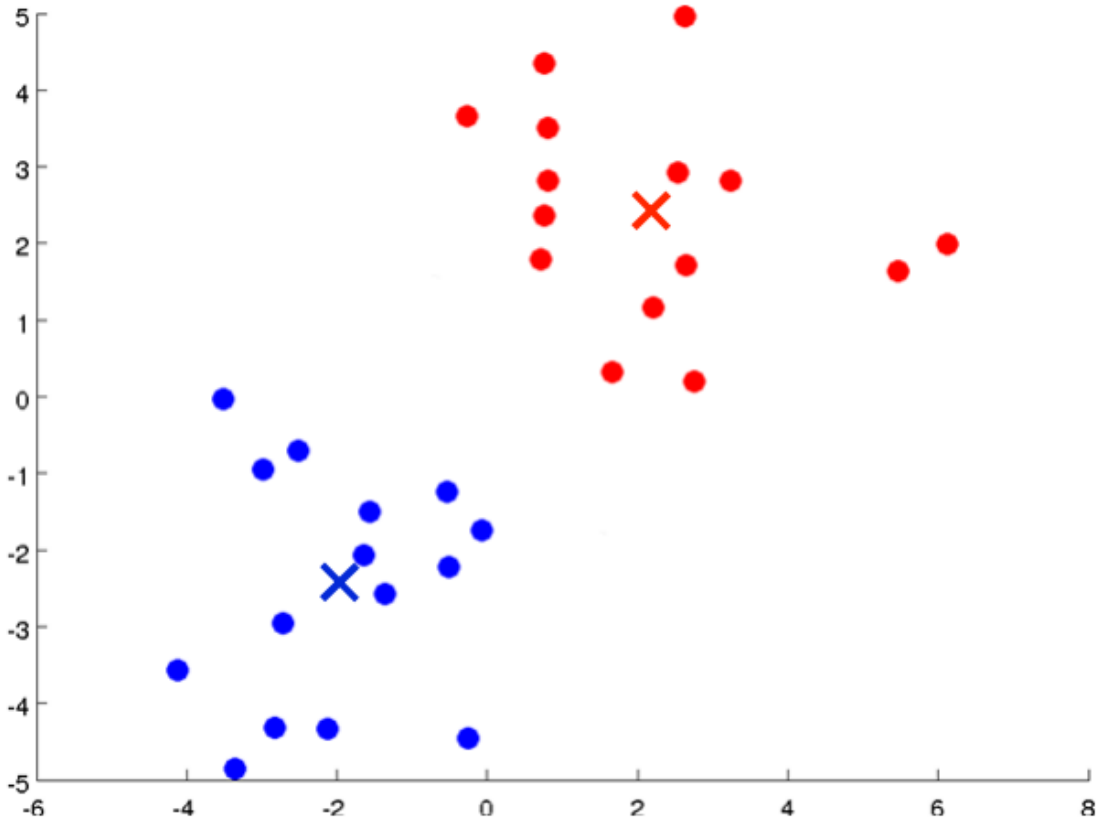





Growth Hackers

# Clustering_ k-means






Growth Hackers

# Clustering_ k-means

- Goal : k 개의 군집으로 데이터를 나누기

1. 임의로 점 k 개를 찍고 , 각 군집의 중심점으로 잡는다 .
2. 각 중심점과 데이터의 거리를 재서 가장 가까운 군집에 배정
3. 배정된 군집이 이전 배정과 같다면 , 알고리즘 종료
4. 배정된 군집이 이전 배정과 하나라도 다르면 , 배정된 군집 내의 평균을 계산해서 새로운 중심점으로 잡는다 . → 2 단계

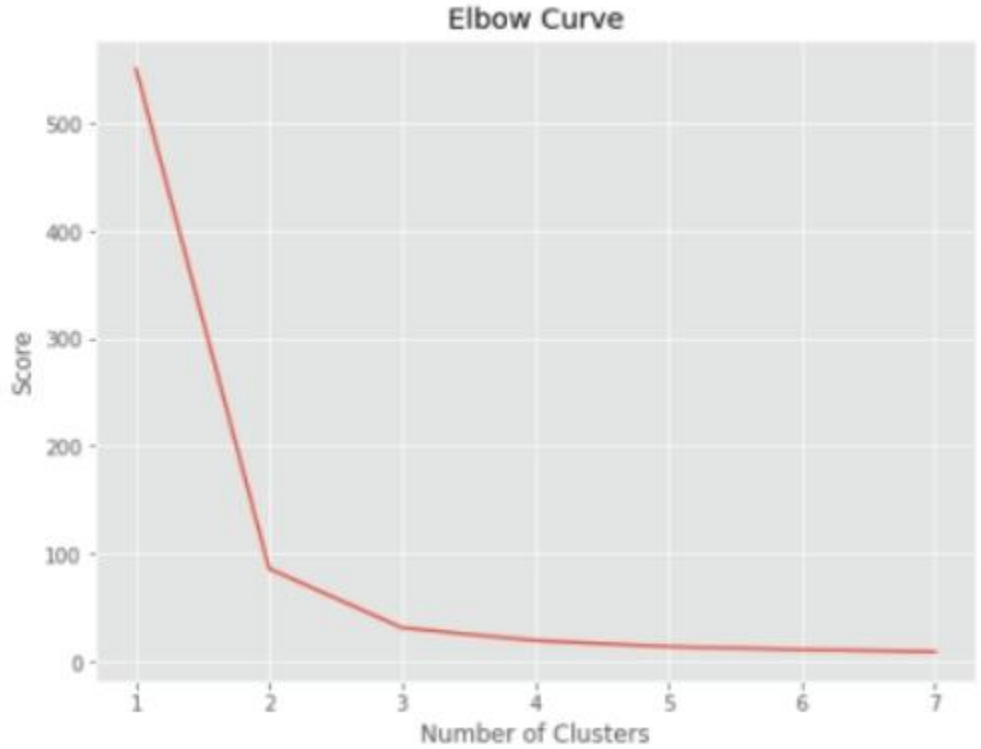


Growth Hackers

# Clustering_ k-means

- K 선택 : 몇 개의 군집으로 나눌 것인가

  ⇒ ( 중심점 ~ 군집 내 데이터 ) 의 제곱합을 작게 하는 'Elbow' 지점의 k 선택

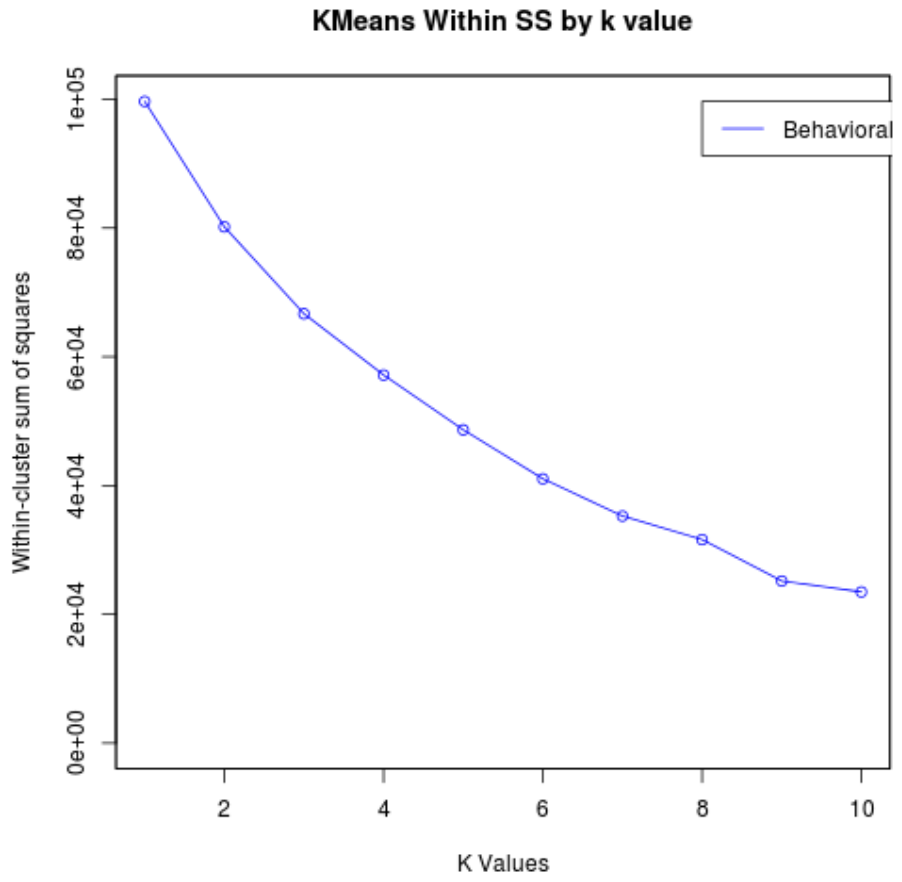





Growth Hackers

# Clustering_ k-means

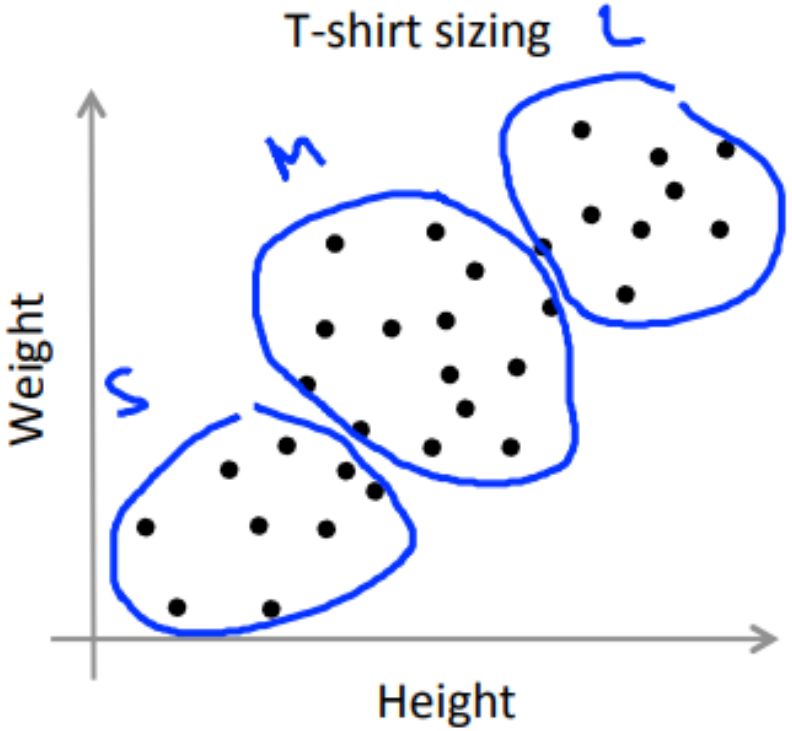
- K 선택 : Elbow 의 한계

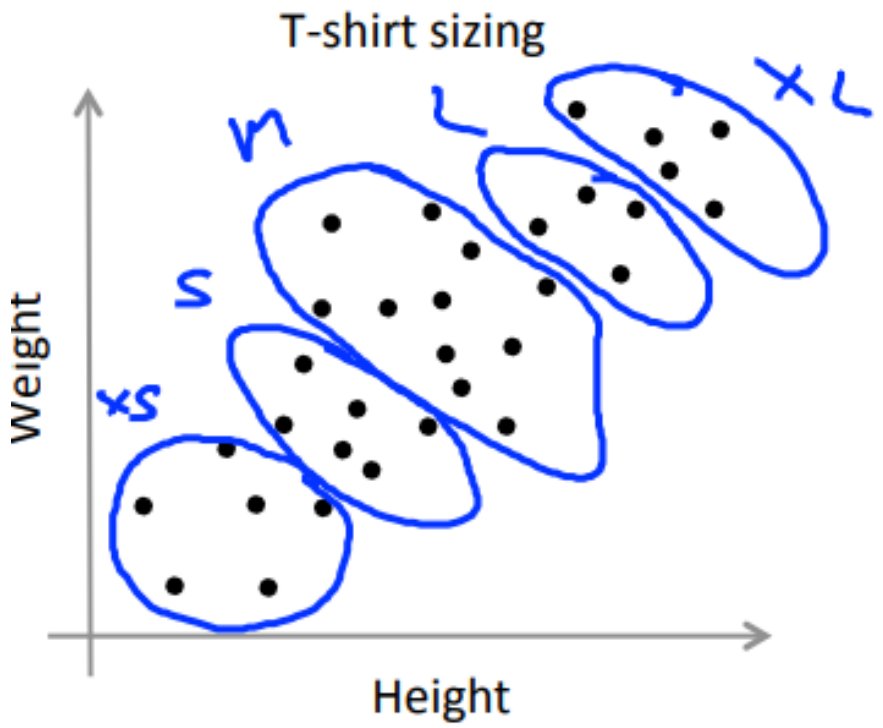



Growth Hackers

# Clustering

- K 선택 : 몇 개의 군집으로 나눌 것인가






Growth Hackers

# Clustering- 실습

Clustering Ex.ipynb

# 더 알아보기

SVD
:https://www.fun-coding.org/recommend_basic6.html

비지도_PCA, NMF, 매니폴드 학습(T-SNE)
:https://data-newbie.tistory.com/24

Clustering for mixed-type data
: https://towardsdatascience.com/clustering-on-mixed-type-data-8bbd0a2569c3

# Quest

- 1) 복습
- 2) 각 ipynb 당 코드 설명 주석 (이미 있는 주석 제외)
  10개씩 달아서 ipynb 파일 3개 압축하여 제출 해주세요
  작성하시는 주석은 기존 주석과 차이를 두기위해
  ## 을 매 주석마다 앞에 붙혀주세요

ex) ##k means clustering



Growth Hackers

# Reference


- Scaling VS Normalization ( https://kharshit.github.io/blog/2018/03/23/scaling-vs-normalization )
- Feature Scaling with scikit-learn( http://benalexkeen.com/feature-scaling-with-scikit-learn/ )
- PCA (https://sherry-data.tistory.com/1)
- PCA(https://datascienceschool.net/view-notebook/f10aad8a34a4489697933f77c5d58e3a/)
- kmeans clustering ( https://www.kaggle.com/vjchoudhary7/kmeans-clustering-in-customer-segmentation/notebook?login=true )
- 정혜선 (GH 4 기 ), Clustering 세션 자료
- 계층적 군집 분석 (https://github.com/bwcho75/dataanalyticsandML/blob/master/
- Clustering/3.%20Hierarchical%20clustering-IRIS%204%20feature.ipynb)
- 계층적 군집 분석 https://bcho.tistory.com/1204
- Andrew Ng, https://www.coursera.org/learn/machine-learning/
- 권철민, 파이선 머신러닝 완벽가이드, 위키북스
- 거리 척도 https://rfriend.tistory.com/199




Growth Hackers