ML 알고리즘

강사: 김남범(nbumkim@gmail.com)

특성공학과 규제 (Feature engineering and regularization)

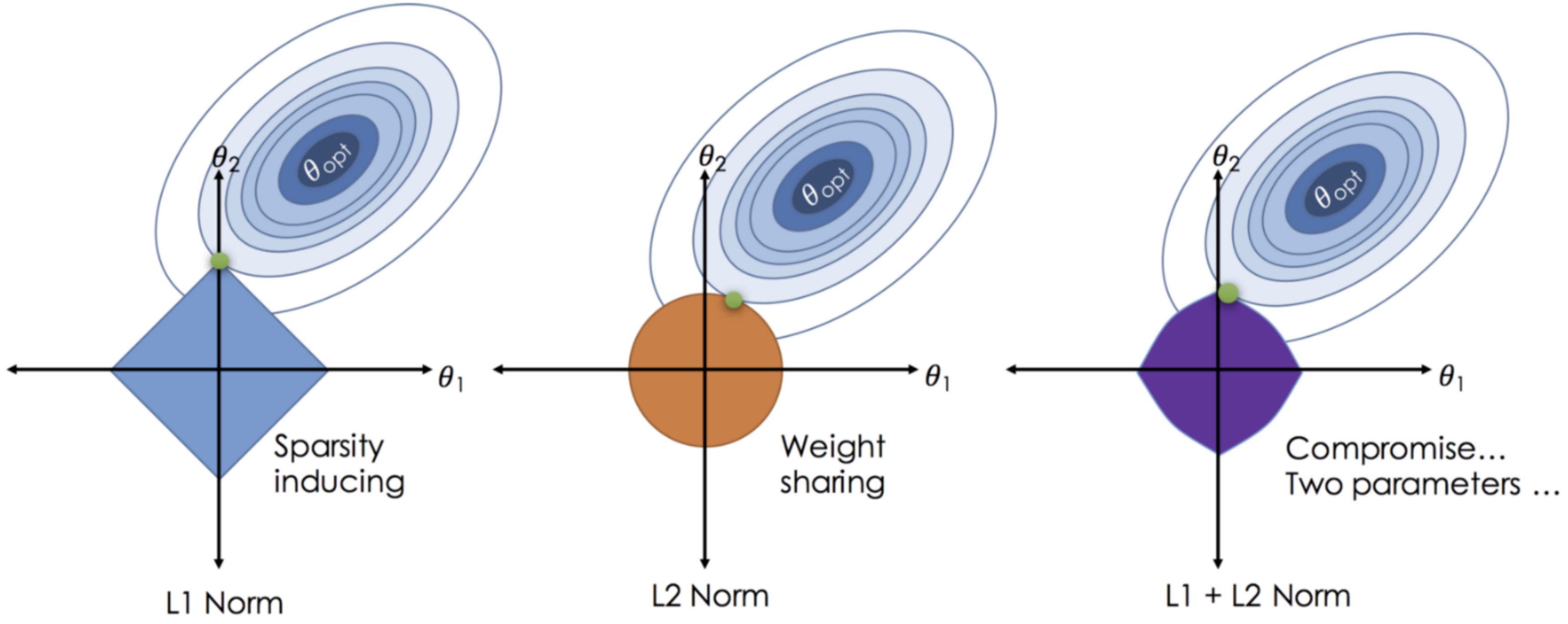
Regularization 목적

- 과적합 방지
- 모델 파라메터 수를 조정 (단순한 모델)

주로 사용하는 penalty parameters

- Ridge (L2 Norm)
- LASSO (L1 Norm, Least absolute shrinkage and selection operator)
- Elastic net (L1 + L2 Norm)

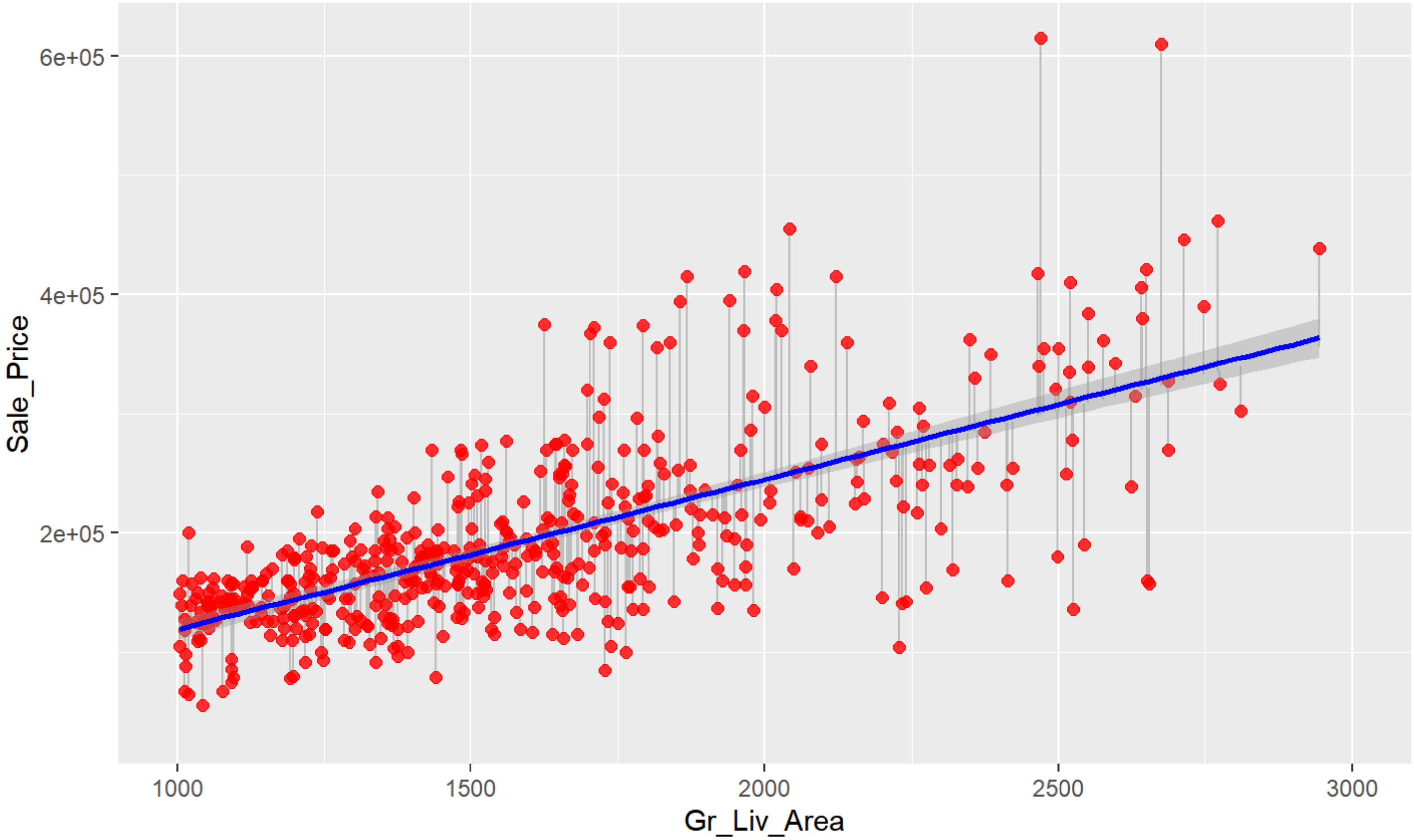
How to effect on parameter estimation



https://soobarkbar.tistory.com/30

Overview of OLS

$$\min SSE = \sum_{i=1}^{n}(y_i - \hat{y}_i)^2$$



Ridge penalty

$J(\beta) = min(SSE \ + \lambda \sum_{j=1}^{p} \beta_j^2)$

- When $\lambda = 0$, there is no effect, equals to the normal OLS regression objective function
- As $\lambda \to \infty$, the penalty becomes large and forces the coefficients toward zero.

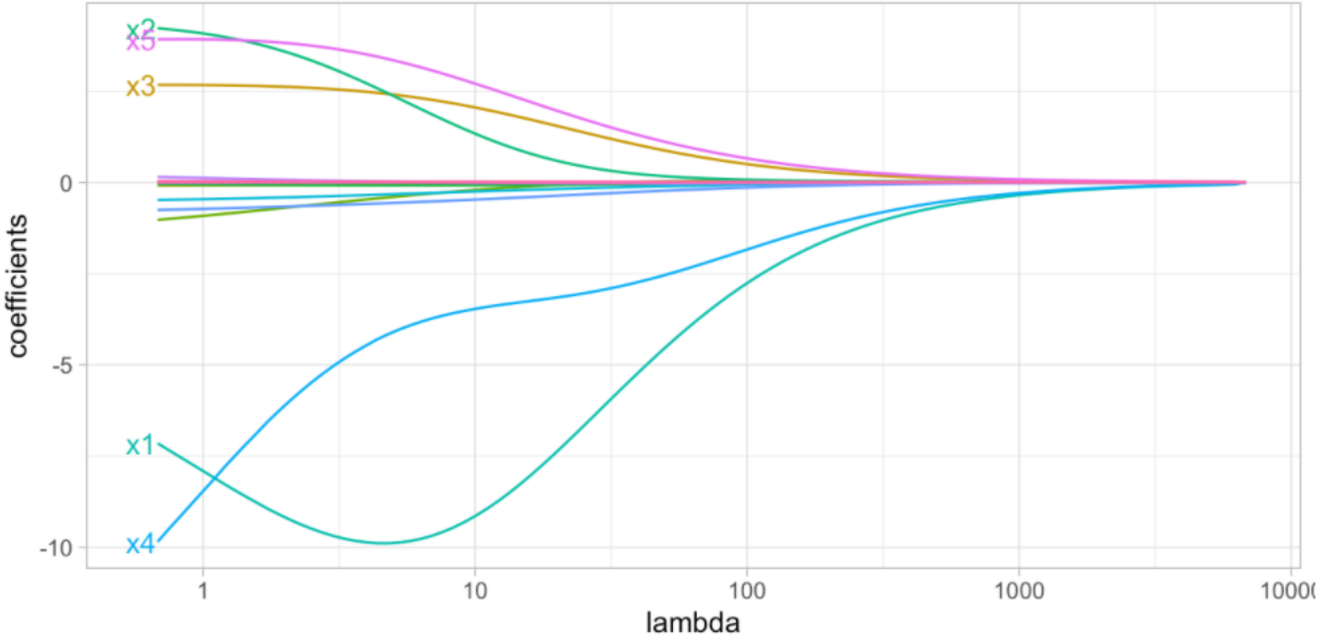


Figure 6.2: Ridge regression coefficients for 15 exemplar predictor variables as λ grows from $0 \to \infty$. As λ grows larger, our coefficient magnitudes are more constrained.

https://bradleyboehmke.github.io/HOML

LASSO penalty

$$J(\beta) = min(SSE \ + \lambda \sum_{j=1}^{p} |\beta_j|)$$

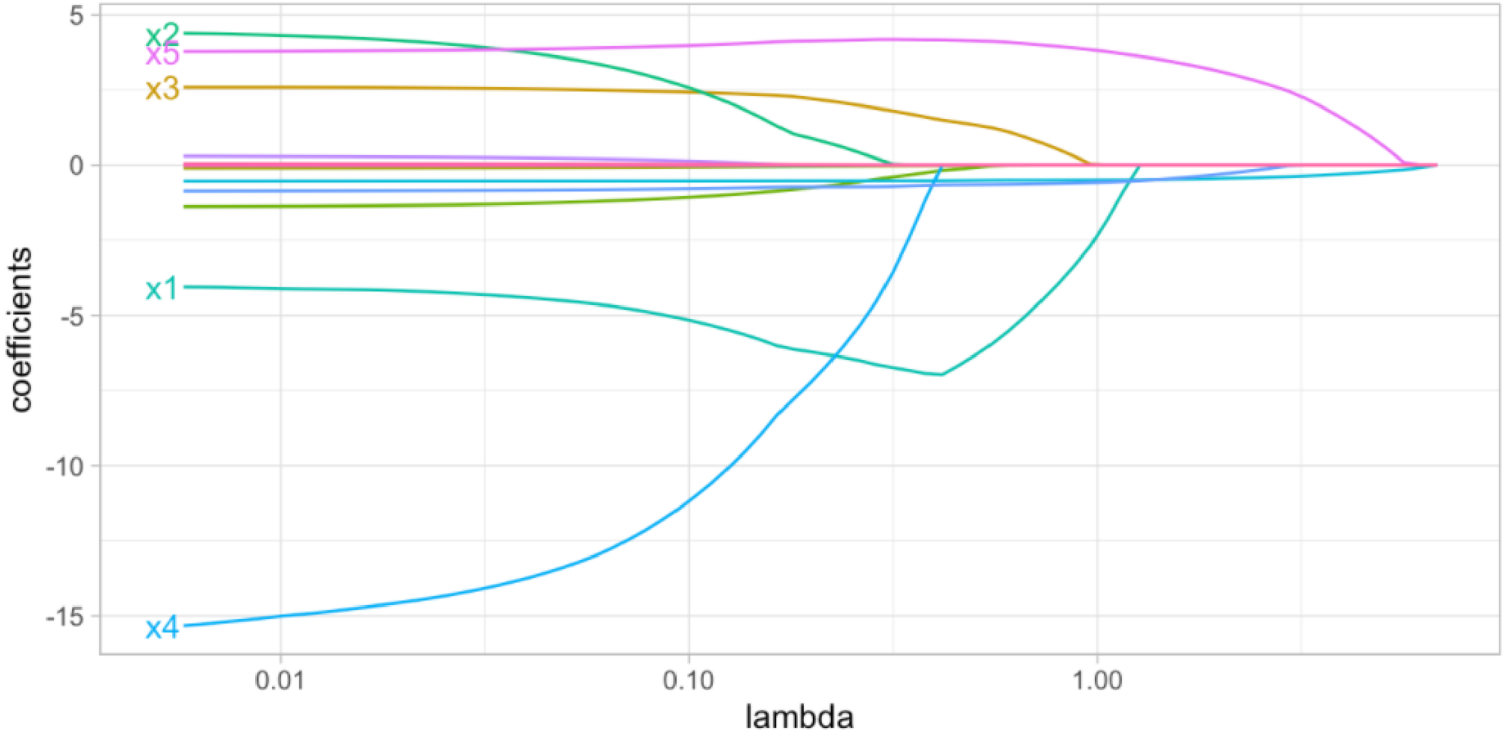


Figure 6.3: Lasso regression coefficients as λ grows from $0 \to \infty$.

https://bradleyboehmke.github.io/HOML

로지스틱 회귀 (Logistic regression)

Why logistic regression

곡선이 확률

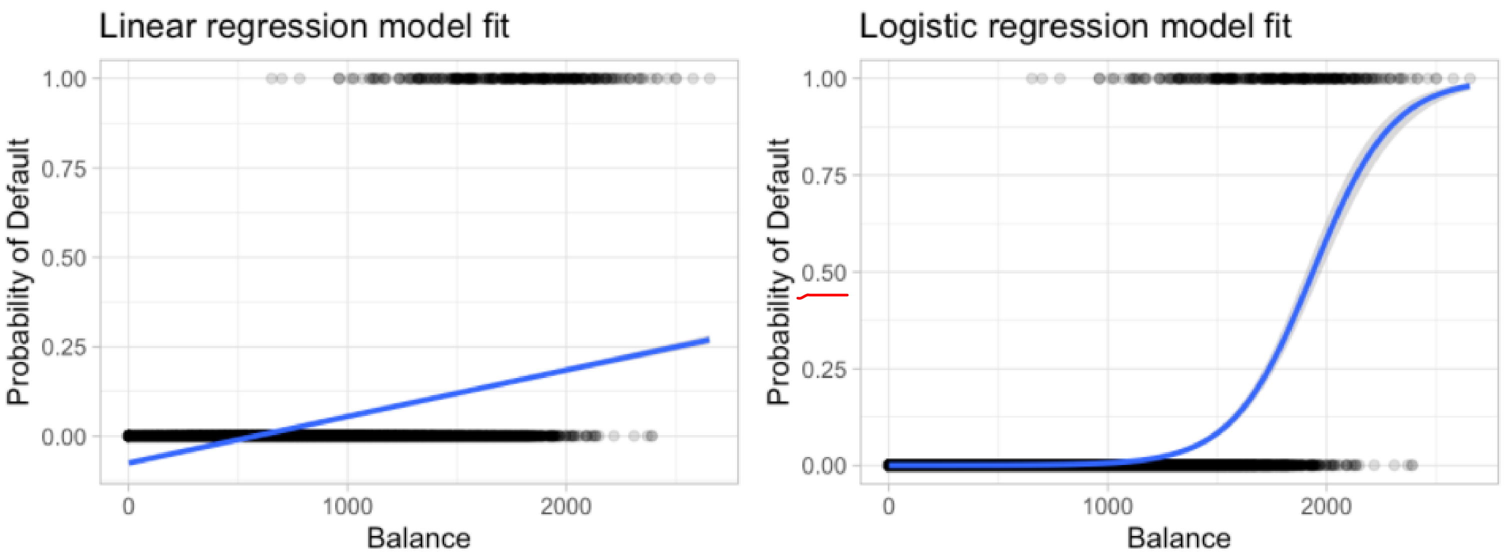


Figure 5.1: Comparing the predicted probabilities of linear regression (left) to logistic regression (right). Predicted probabilities using linear regression results in flawed logic whereas predicted values from logistic regression will always lie between 0 and 1.

https://bradleyboehmke.github.io/HOML

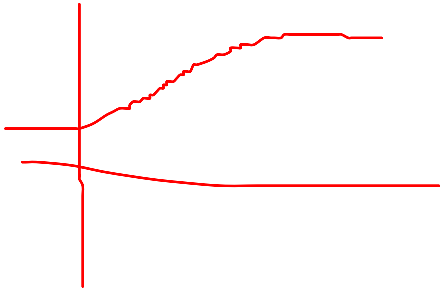
Multiple logistic regression

$p(x) \geq 0.5 \rightarrow 1$

- Formula extend

$$p(X) = \frac{e^{\beta_0+\beta_1X_1+\ldots+\beta_pX_p}}{1+e^{\beta_0+\beta_1X_1+\ldots+\beta_pX_p}} = \frac{1}{1+e^{-(\beta_0+\beta_1X_1+\ldots+\beta_pX_p)}}$$

0과 1을 나타내기 위해

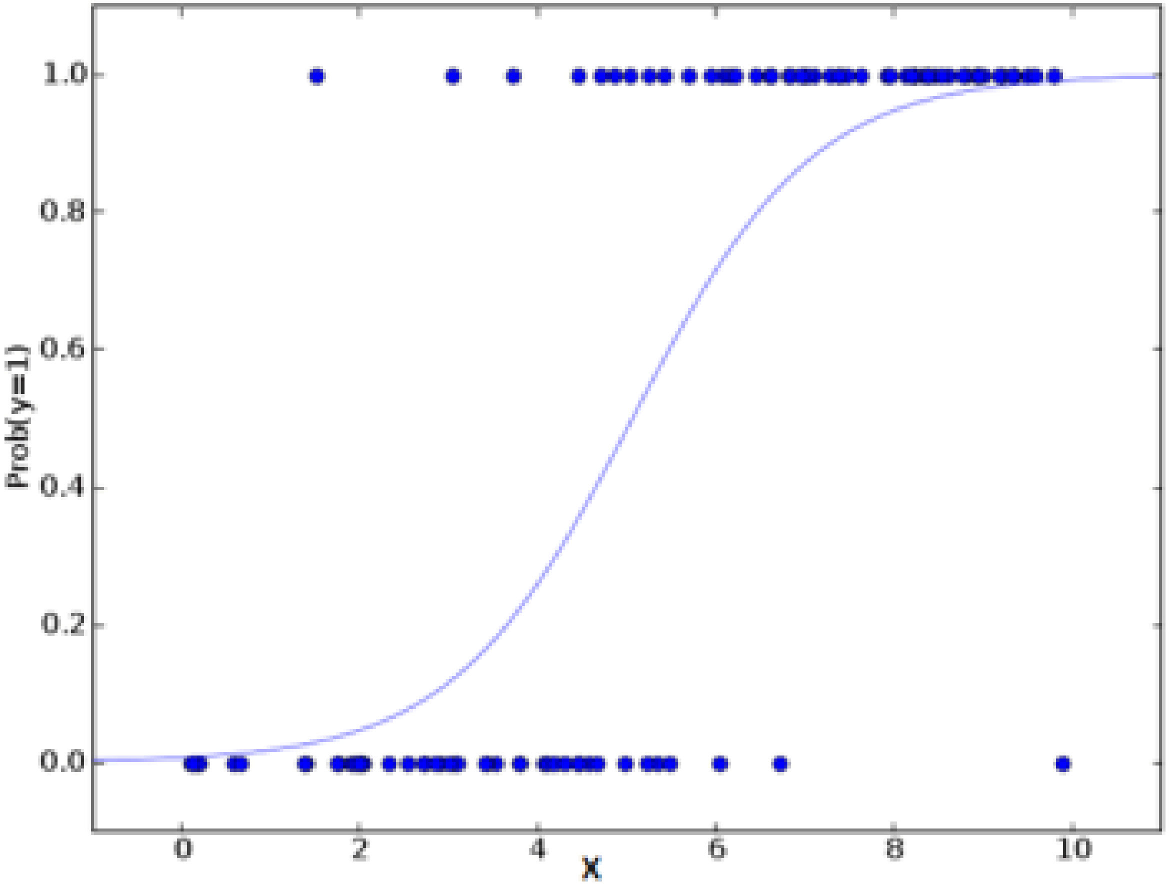


- Logit transformation

$$g(X) = ln[\frac{p(x)}{1-p(x)}] = \beta_0 + \beta_1X + \ldots + \beta_pX_p$$

확률밀도함수로부터 유도

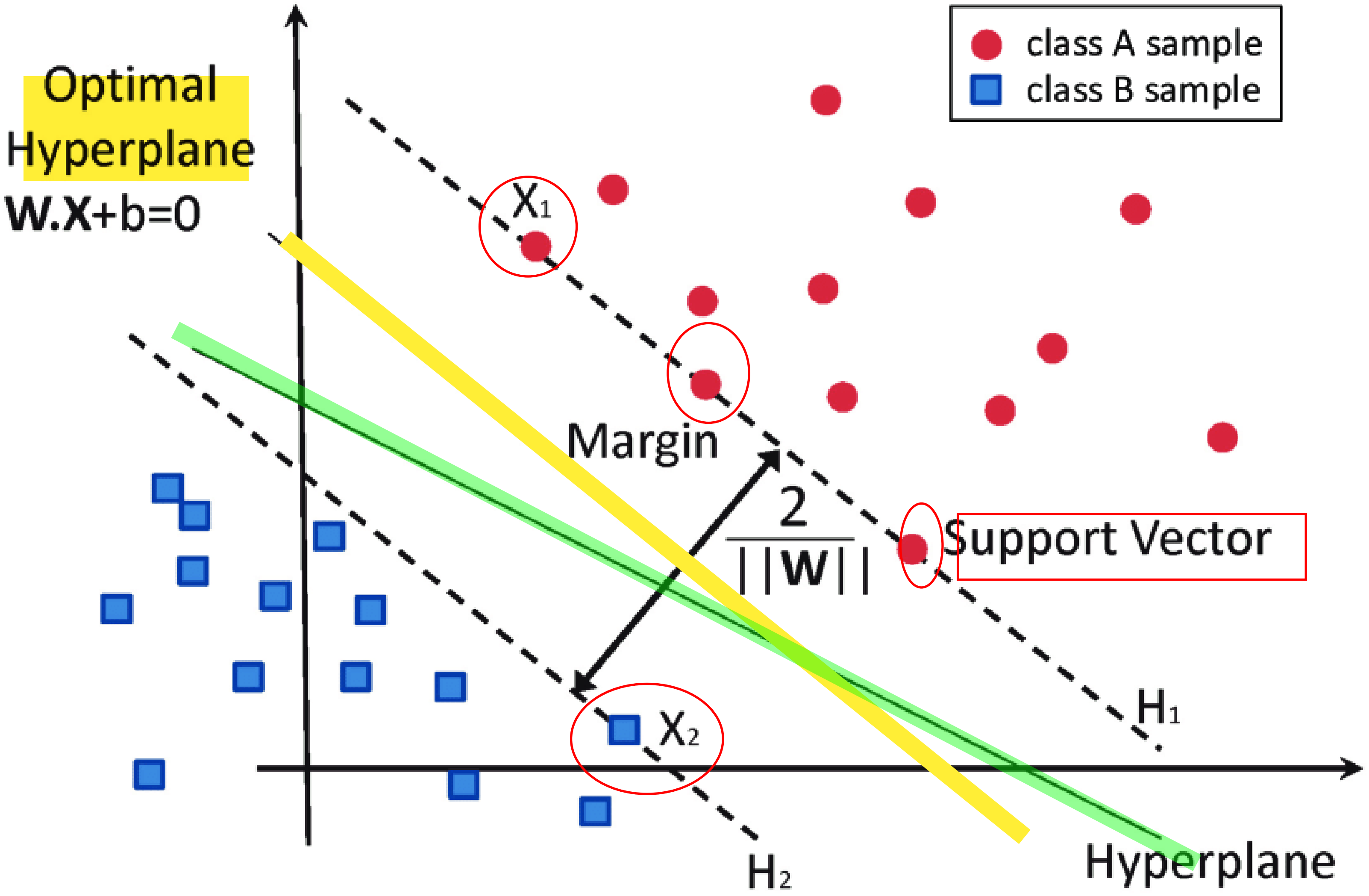
Logistic estimation



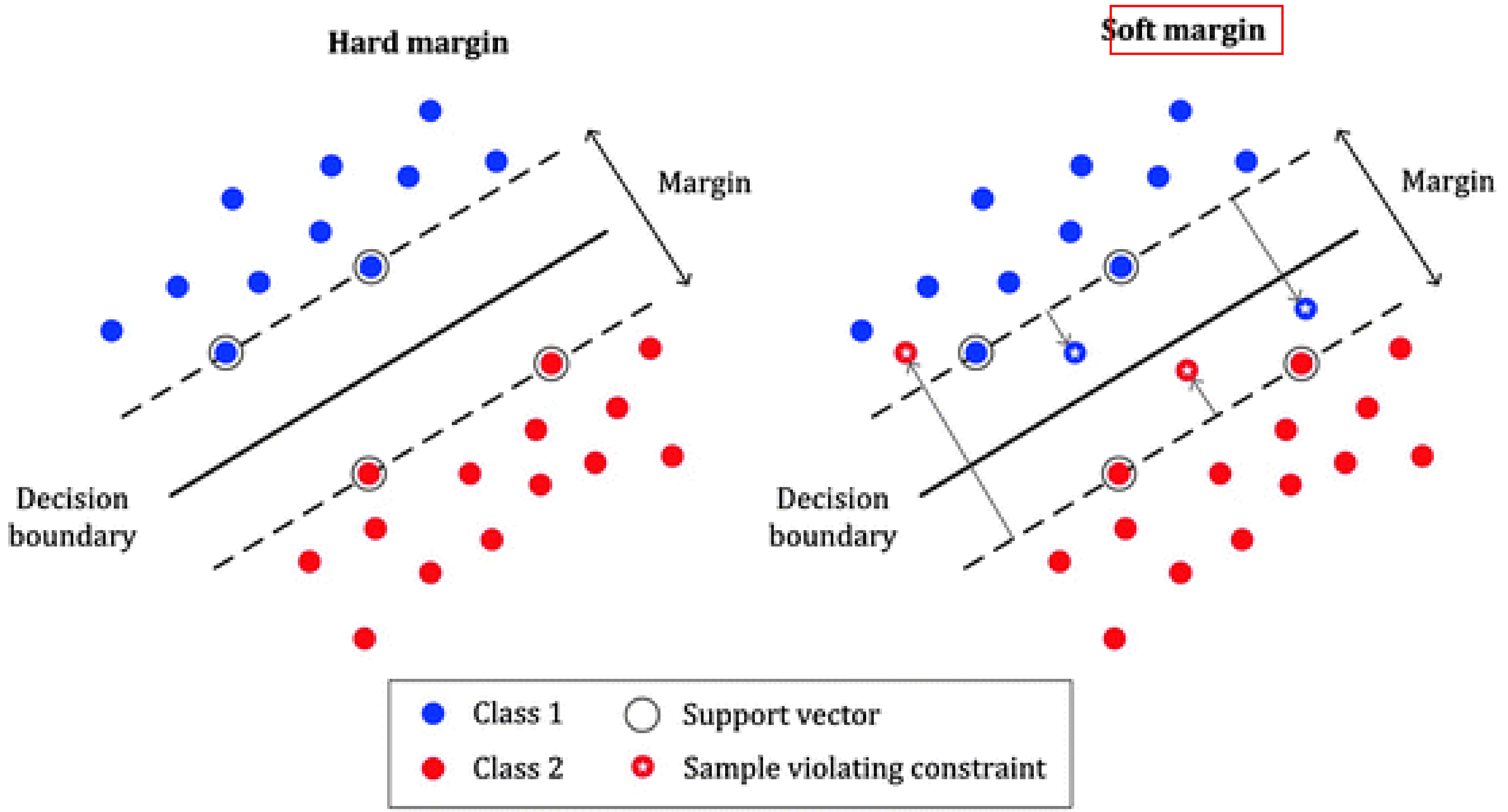
https://www.analyticsvidhya.com/blog/2015/11/beginners-guide-on-logistic-regression-in-r/

SVM (Support Vector Machine)

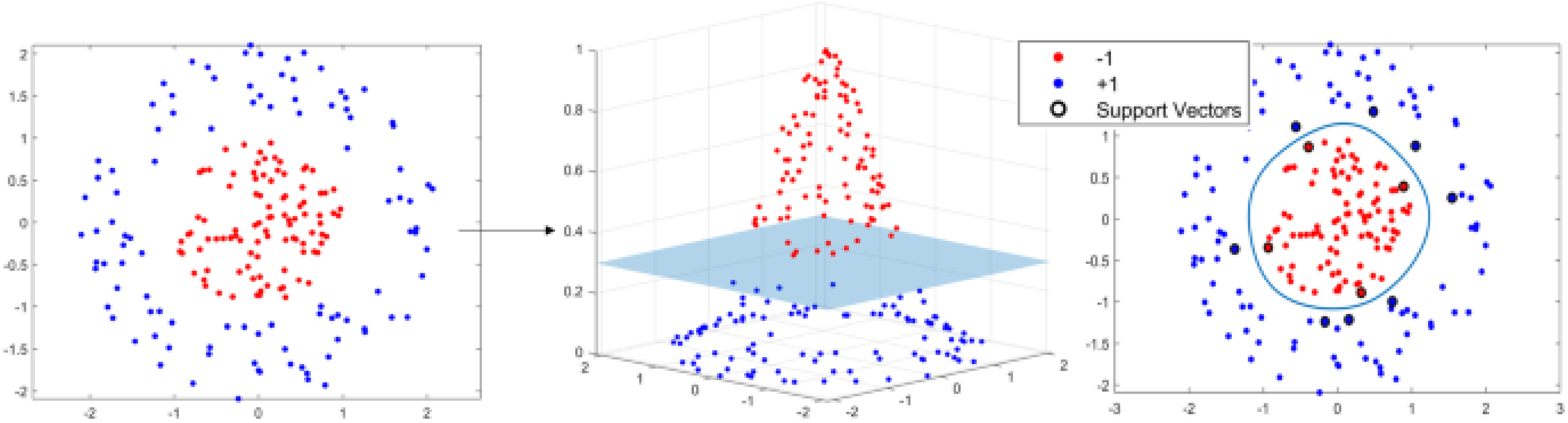
SVM (Support vector machine)



SVM with softmargin



SVM with kernel



(a) 2D non-linear training data

(b) 3D mapped data points using the Gaussian kernel and separating hyperplane

(c) non-linear SVM with the Gaussian kernel training result

$$K(x,y) = e^{-}(\gamma||x-y||^2)$$

SVM + softmargin + Kernel

Kernel SVM

확률적 경사 하강법 (Stochastic gradient descent)

Derivatives of cost function of logistic regression

Let's see the cases of cost function of logistic regression
This equation does not have a closed-form solution

$$J(\theta) = -\frac{1}{m}\sum_{i=1}^{m}[y_i log(h_\theta(x_i)) + (1 - y_i) log(1 - h_\theta(x_i))$$

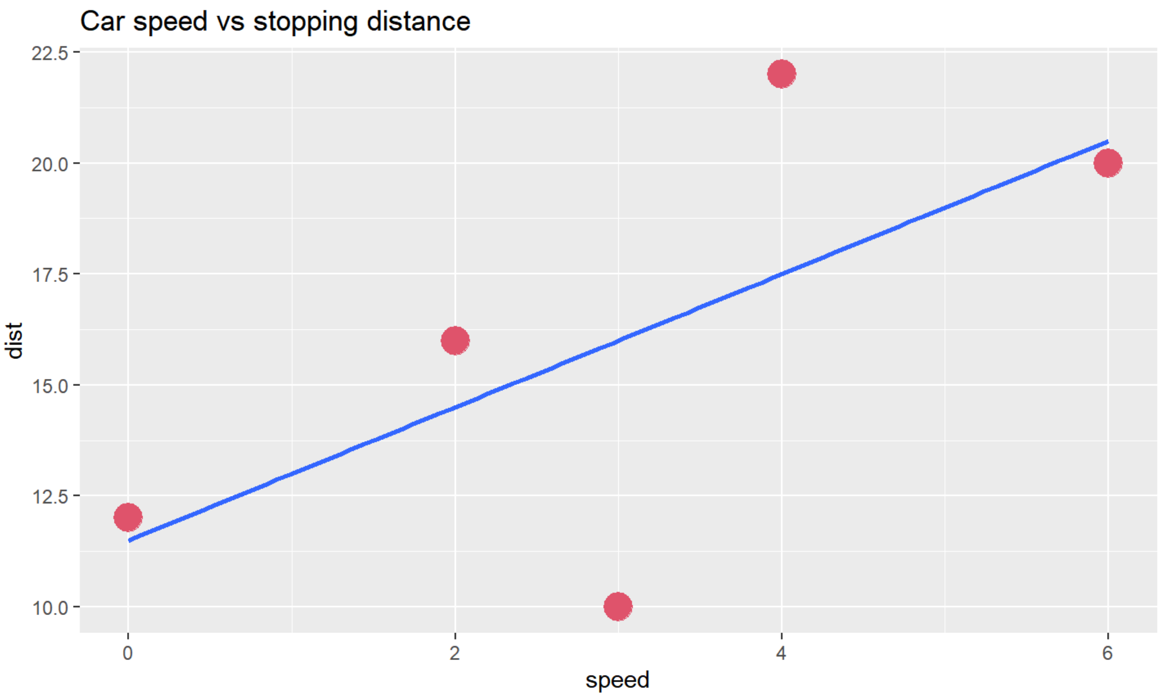
where, $h_\theta(x_i) = \frac{1}{1+e^{-\theta x}}, \quad y \in 0, 1$

The concept of gradient descent (GD) algorithm

```
# A tibble: 5 x 2
  speed  dist
  <dbl> <dbl>
1     0    12
2     2    16
3     3    10
4     6    20
5     4    22
```

```
(Intercept)       speed
       11.5         1.5
```

```
[1] "Sum of squard error =  59"
```



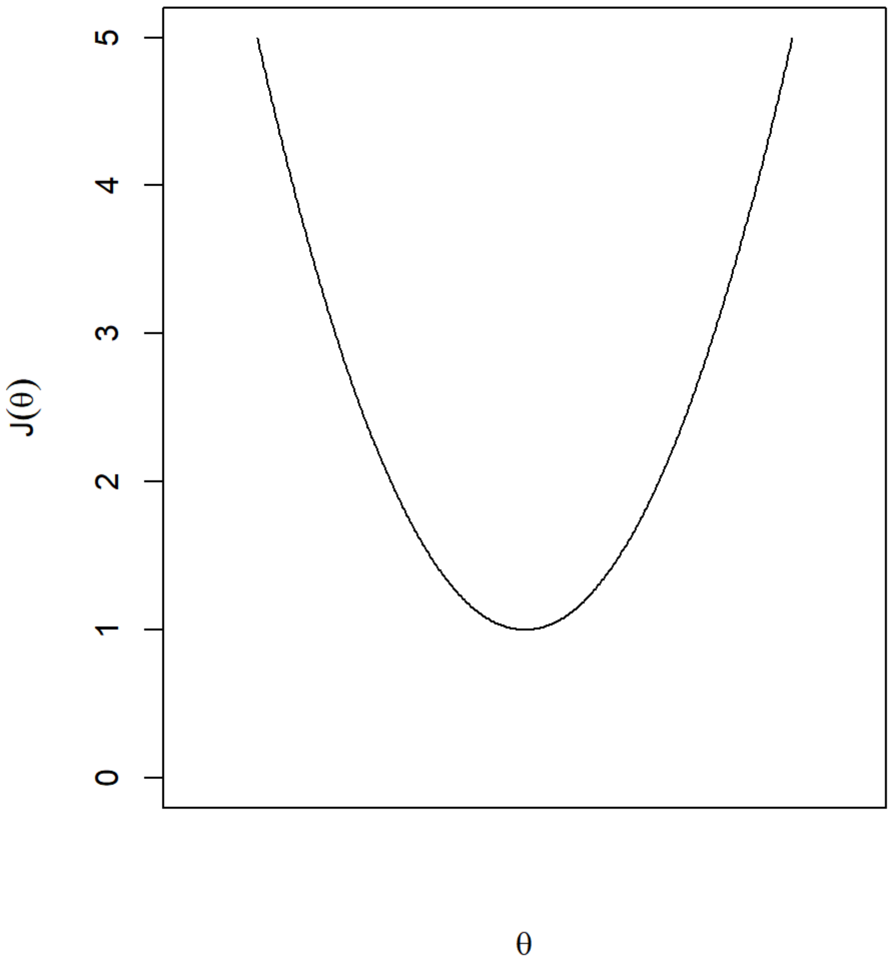
Gradient descent (GD) algorithm

- Objective (cost) function =
  $J(\theta) = \frac{1}{2m}\sum_{i=1}^{m}(h_\theta(x_i) - y_i)^2$
  $= \frac{1}{2m}\sum_{i=1}^{m}(y_i - h_\theta(x_i))^2$

- Parameter update :
  Repeat until convergence {

  $\theta_j^{(n+1)} = \theta_j^{(n)} - \gamma\frac{\partial}{\partial\theta_j}J(\theta^{(n)})$

  }



K-means clustering

K-means clustering animation 1

https://www.youtube.com/watch?v=5I3Ei69I40s&ab_channel=StatQuestwithJoshStarmer

https://www.youtube.com/watch?v=nXY6PxAaOk0&ab_channel=JohanHagelb%C3%A4ck

Geometric distance measures

- Euclidean: $d(\vec{x}, \vec{y}) = \sqrt{\sum_{i=1}^{p} (x_i - y_i)^2}, \quad \vec{x}, \vec{y} \in p$
- Manhattan: $d(\vec{x}, \vec{y}) = \sum_{i=1}^{p} |x_i - y_i|$
- Minkowski: $d(\vec{x}, \vec{y}) = (\sum_{i=1}^{p} |x_i - y_i|^q)^{\frac{1}{q}}$
- Gower: Manhattan (Continuous) + Dice coefficient (Nominal)

Euclidean vs Manhattan

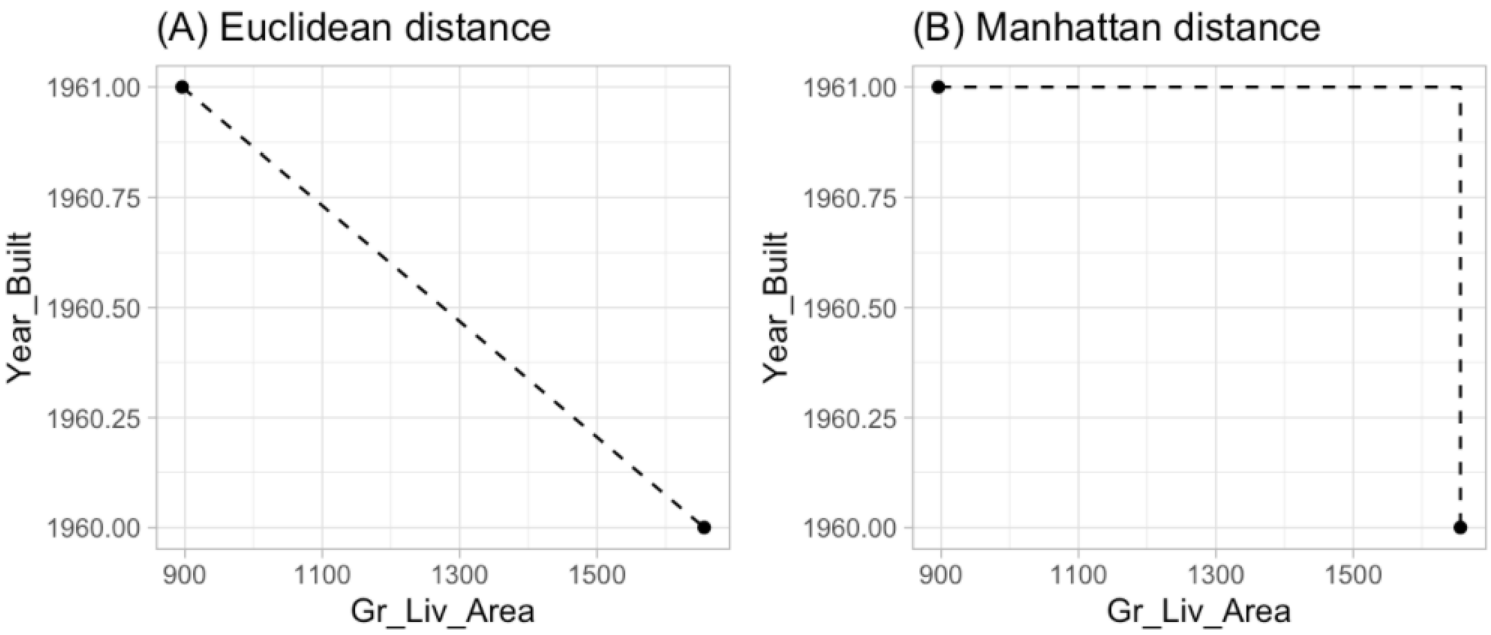


Figure 8.2: Euclidean (A) versus Manhattan (B) distance.

https://bradleyboehmke.github.io/HOML

Correlation distance measures

- Pearson: $d(\vec{x},\vec{y}) = \frac{\sum_i^p (x_i - \overline{x})(y_i - \overline{y})}{\sqrt{\sum_i^p (x_i - \overline{x})^2}\sqrt{\sum_i^n (y_i - \overline{y})^2}}$
- Mahalanobis: $d(\vec{x},\vec{y}) = \sqrt{(\vec{x} - \vec{y})^T S^{-1} (\vec{x} - \vec{y})}, S = Covariance\ Matrix$

Correlation-based distance measure

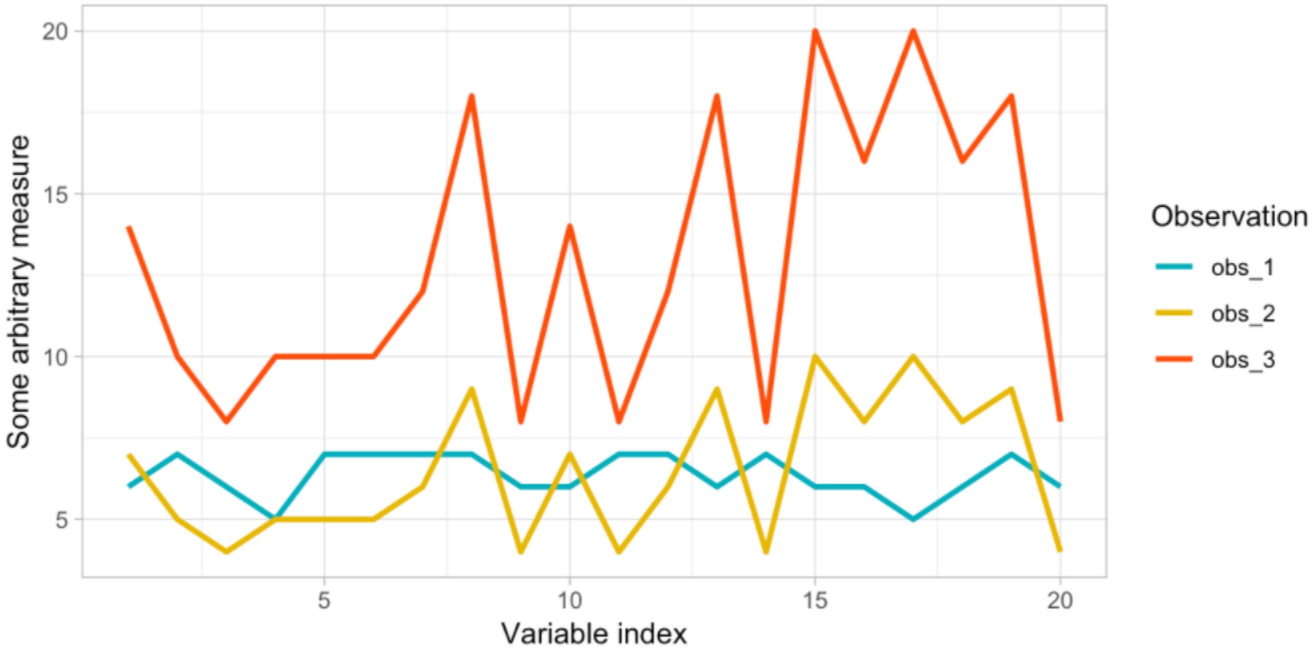


Figure 20.1: Correlation-based distance measures will capture the correlation between two observations better than a non-correlation-based distance measure; regardless of magnitude differences.

https://bradleyboehmke.github.io/HOML

Defining clusters

The basic idea behind k-means clustering is constructing clusters so that the total within-cluster variation (SS_w) is minimized

- $SS_w = \sum_{i=1}^{k} W(C_i)$, k = cluster number

- where $W(C_i) = \sum_{j \in C_i} (x_j - u_i)^2$

The compactness of SSw

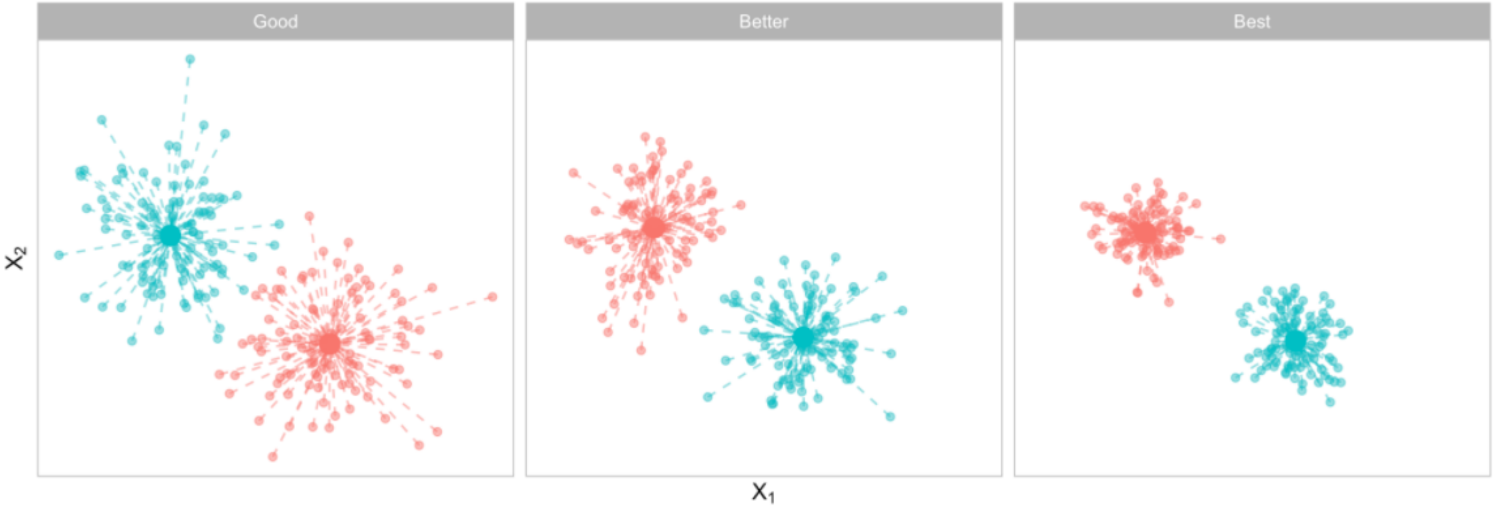


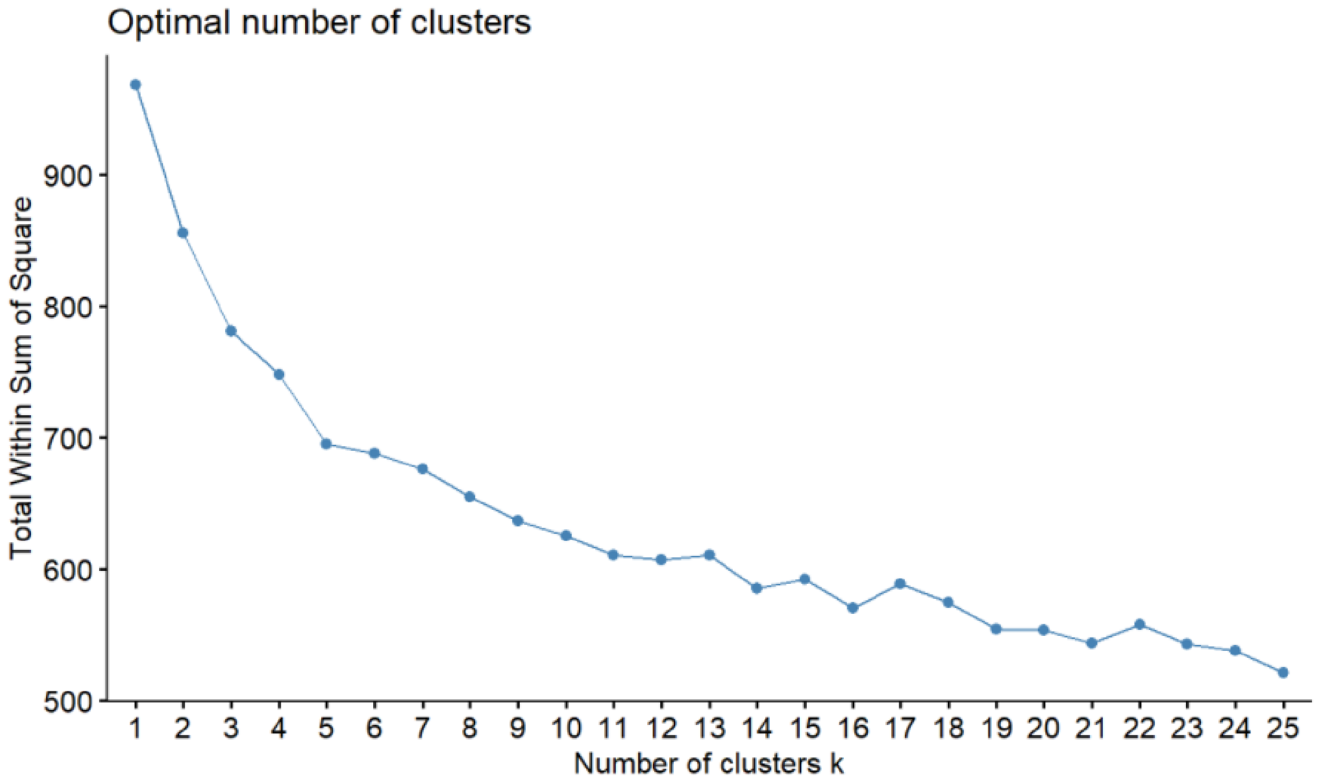
Figure 20.2: Total within-cluster variation captures the total distances between a cluster's centroid and the individual observations assigned to that cluster. The more compact the these distances, the more defined and isolated the clusters are.

https://bradleyboehmke.github.io/HOML

k−means algorithm

1. Specify the number of clusters (k)
2. Select k observations at random from the data set to use as the initial cluster centroids.
3. Assign each observation to their closest centroid based on the distance measure selected
4. For each of the k clusters update the cluster centroid by calculating the new mean values of all the data points in the cluster
5. Iterate steps 3 - 4 until minimizing SS_w

Choosing cluster numbers: Elbow method



https://bradleyboehmke.github.io/HOML

06_3. PCA

- A method for finding low-dimensional representations of a data set
- A smaller number of dimensions that are as interesting as possible
- Each of the new dimensions is a linear combination of the original p features
- Numeric data should be standardized (e.g., centered and scaled) to make features comparable.

What is Principal component analysis (PCA)

Data example

F1	F2
2.5	2.4
0.5	0.7
2.2	2.9
1.9	2.2
3.1	3.0
2.3	2.7

Illustration of PCA

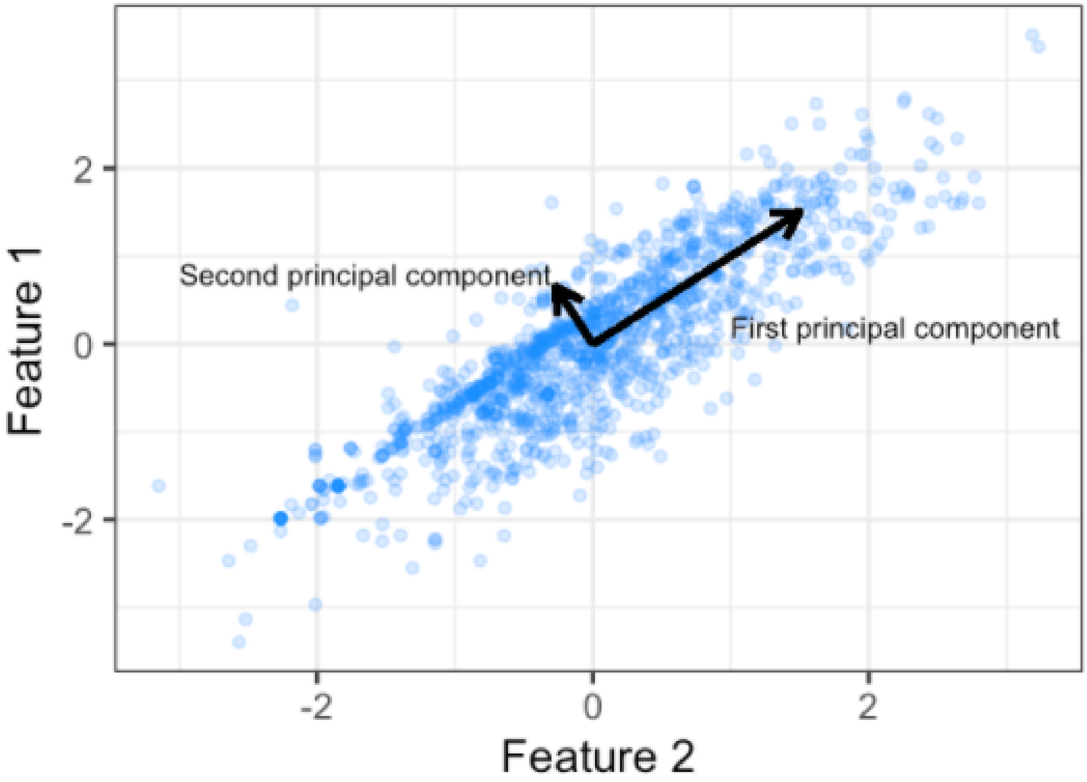


Figure 17.1: Principal components of two features that have 0.56 correlation.

https://bradleyboehmke.github.io/HOML

Formular of PC component

- First PC

  $$w_1 = argmax(\frac{||Xw||^2}{w^T w}) = argmax(\frac{w^T X^T X w}{w^T w})$$

- $k - th$ component

  $$\hat{X}_k = X - \sum_{s=1}^{k-1} X w_s w_s^T$$

  $$w_k = argmax(\frac{||\hat{X}_k w||^2}{w^T w}) = argmax(\frac{w^T \hat{X}_k^T \hat{X}_k w}{w^T w})$$