KUBIG
Data Science and Machine Learning

Week 5. Support Vector Machine

Review

- Least Square Regression solves

$$\min_{\boldsymbol{\beta}} (\mathbf{Y} - \mathbf{X}\boldsymbol{\beta})^T (\mathbf{Y} - \mathbf{X}\boldsymbol{\beta})$$

- OLS estimator is an Unbiased Estimator, MLE, and UMVUE.

$$\widehat{\boldsymbol{\beta}}_{OLS} = (\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\mathbf{Y}$$

$$E\left[\widehat{\boldsymbol{\beta}}_{OLS}\right] = (\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T E[\mathbf{Y}] = (\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\mathbf{X}\boldsymbol{\beta} = \boldsymbol{\beta}$$

Review

- Expected Prediction Error

$$E\left[(Y_0 - \hat{Y}_0)^2\right] = \underbrace{\sigma^2}_{\text{Irreducible error}} + \underbrace{E\left[(\mu_0 - \hat{Y}_0)^2\right]}_{\text{Model error}}$$

where $Y_0 = \mu_0 + \epsilon_0 = \mathbf{X}_0^T \boldsymbol{\beta} + \epsilon_0$

and $\hat{Y}_0 = \mathbf{X}_0^T \widehat{\boldsymbol{\beta}}$

Review

- Model Error

$$E\left[(\mu_0 - \hat{Y}_0)^2\right] = E\left[(\mu_0 - E[\hat{Y}_0] + E[\hat{Y}_0] - \hat{Y}_0)^2\right]$$

$$= \underbrace{(\mu_0 - E[\hat{Y}_0])^2}_{\text{Bias}^2} + \underbrace{Var[\hat{Y}_0]}_{\text{variance}}$$

- $\hat{\boldsymbol{\beta}}_{OLS}$ has the smallest variance among all unbiased estimators.

Review

- Ridge Regression solves

$$\min_{\boldsymbol{\beta}} (\mathbf{Y} - \mathbf{X}\boldsymbol{\beta})^T(\mathbf{Y} - \mathbf{X}\boldsymbol{\beta}) + \lambda||\boldsymbol{\beta}||_2^2 \qquad (L2\ penalty)$$

- LASSO Regression solves

$$\min_{\boldsymbol{\beta}} (\mathbf{Y} - \mathbf{X}\boldsymbol{\beta})^T(\mathbf{Y} - \mathbf{X}\boldsymbol{\beta}) + \lambda||\boldsymbol{\beta}||_1 \qquad (L1\ penalty)$$

Review

- Primal Problem

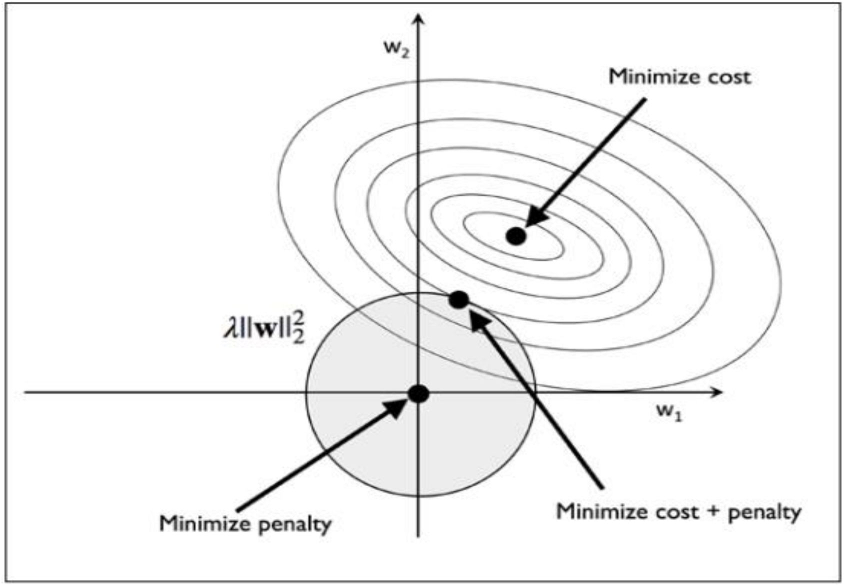
$$\min_{\boldsymbol{\beta}} (\mathbf{Y} - \mathbf{X}\boldsymbol{\beta})^T(\mathbf{Y} - \mathbf{X}\boldsymbol{\beta})$$

subject to $||\boldsymbol{\beta}||_{\mathrm{p}}^{\mathrm{p}} - C \leq 0$

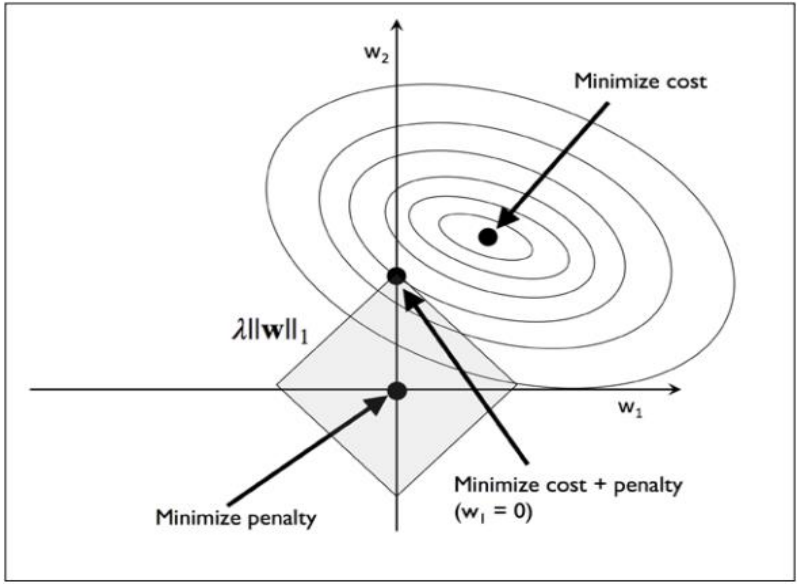
- Dual Problem

$$\min_{\boldsymbol{\beta}} (\mathbf{Y} - \mathbf{X}\boldsymbol{\beta})^T(\mathbf{Y} - \mathbf{X}\boldsymbol{\beta}) + \lambda(||\boldsymbol{\beta}||_{\mathrm{p}}^{\mathrm{p}} - C\ )$$

Review

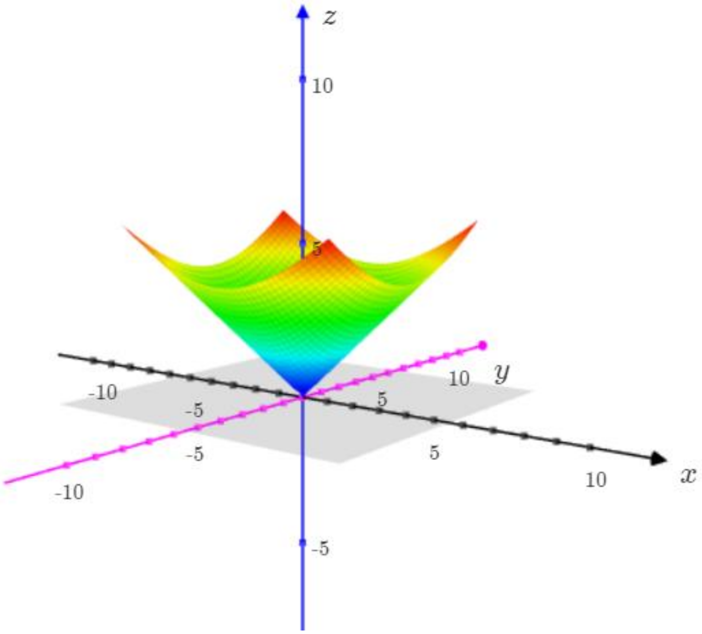


w_2
Minimize cost
$\lambda\|\mathbf{w}\|_2^2$
w_1
Minimize penalty
Minimize cost + penalty



w_2
Minimize cost
$\lambda\|\mathbf{w}\|_1$
w_1
Minimize cost + penalty
(w_1 = 0)
Minimize penalty

Review



Review

$$\widehat{\boldsymbol{\beta}}^{\lambda,p} = \underset{\boldsymbol{\beta}}{argmin}\ (\mathbf{Y}-\mathbf{X}\boldsymbol{\beta})^T(\mathbf{Y}-\mathbf{X}\boldsymbol{\beta}) + \lambda(||\boldsymbol{\beta}||_{\mathrm{p}}^{\mathrm{p}} - C\ )$$

$$\Leftrightarrow \underset{\boldsymbol{\beta}}{argmin}\ (\mathbf{Y}-\mathbf{X}\boldsymbol{\beta})^T(\mathbf{Y}-\mathbf{X}\boldsymbol{\beta}) + \lambda||\boldsymbol{\beta}||_{\mathrm{p}}^{\mathrm{p}}$$

- Although $\widehat{\boldsymbol{\beta}}^{\lambda,p}$ is biased, it can achieve smaller variance so that its model error (MSE) is smaller than $\widehat{\boldsymbol{\beta}}_{OLS}$ with a carefully selected λ.

Review

- Regularized Logistic Regression solves

$$\min_{\boldsymbol{\beta}} \; -\sum_{i=1}^{n}[\,y_i(\boldsymbol{\beta}^T\mathbf{X}_i) - \log(1+\exp(\boldsymbol{\beta}^T\mathbf{X}_i))] + \lambda||\boldsymbol{\beta}||_p^p$$

```
# Logistic regression
from sklearn.linear_model import LogisticRegression
Logit = LogisticRegression(C=1e2, random_state=1023)  # C = 1/λ. 디폴트: L2, One-versus-Rest.
Logit.fit(X_train_std, y_train)
```

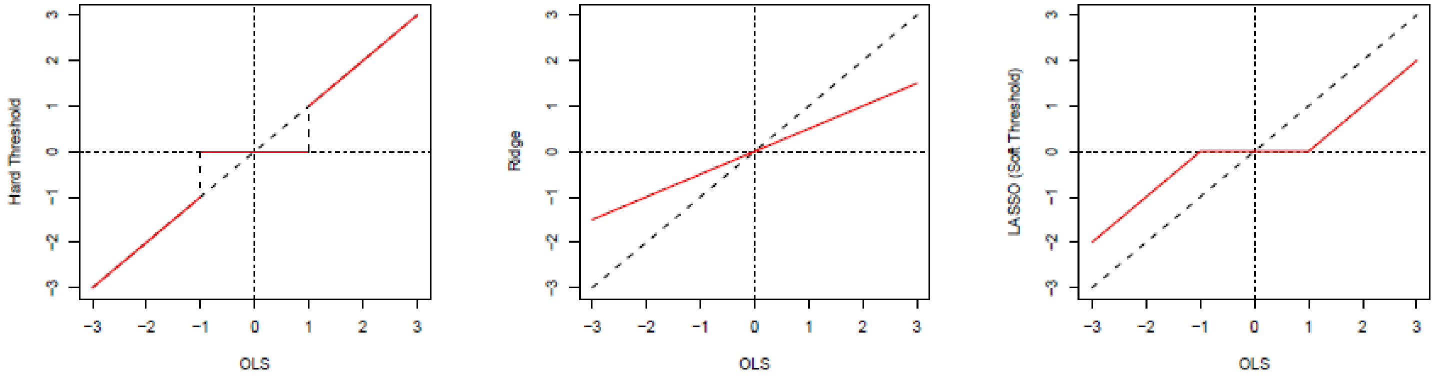
Review

- One-dimensional Solution

$$\hat{\beta}_{OLS} = \frac{1}{n}\sum x_i y_i \qquad \hat{\beta}_{Ridge} = \frac{\hat{\beta}_{OLS}}{1+\lambda} \qquad \hat{\beta}_{LASSO} = S_\lambda(\hat{\beta}_{OLS})$$

※ $S_\lambda(x) = sign(x)\ (|x| - \lambda)_+$

Review



(a) Hard Thresh. (b) Ridge Regression (c) Lasso (Soft Thresh.)

Review

- Elastic Net solves

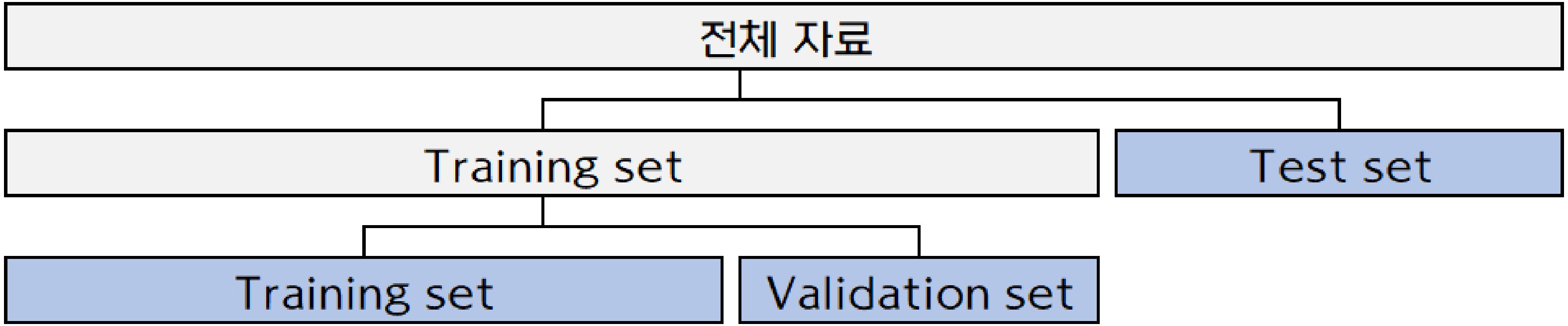
$$\min_{\boldsymbol{\beta}} (\mathbf{Y}-\mathbf{X}\boldsymbol{\beta})^T(\mathbf{Y}-\mathbf{X}\boldsymbol{\beta}) + \lambda\left[\alpha||\boldsymbol{\beta}||_1 + \frac{1}{2}(1-\alpha)||\boldsymbol{\beta}||_2^2\right]$$

- One-dimensional Case

$$\hat{\beta}_{\text{Elastic net}} = \frac{S_\lambda(\hat{\beta}_{OLS})}{1+\lambda(1-\alpha)}$$

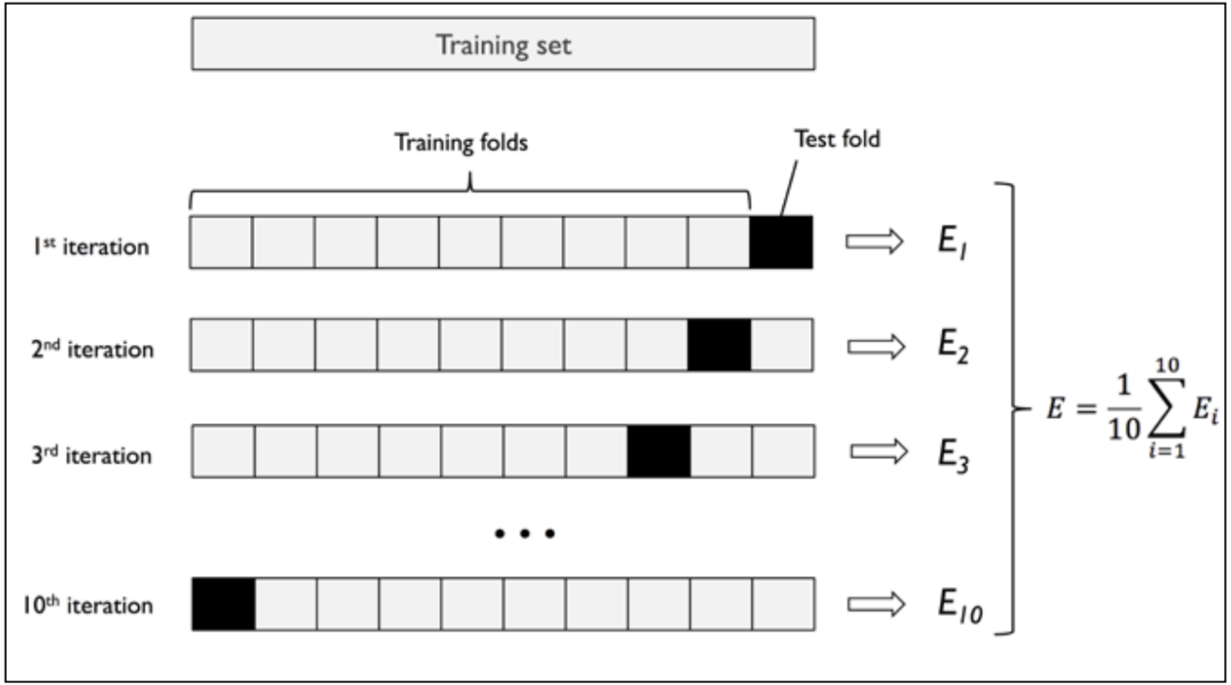
Review

- Cross-Validation



Review

- K = 10

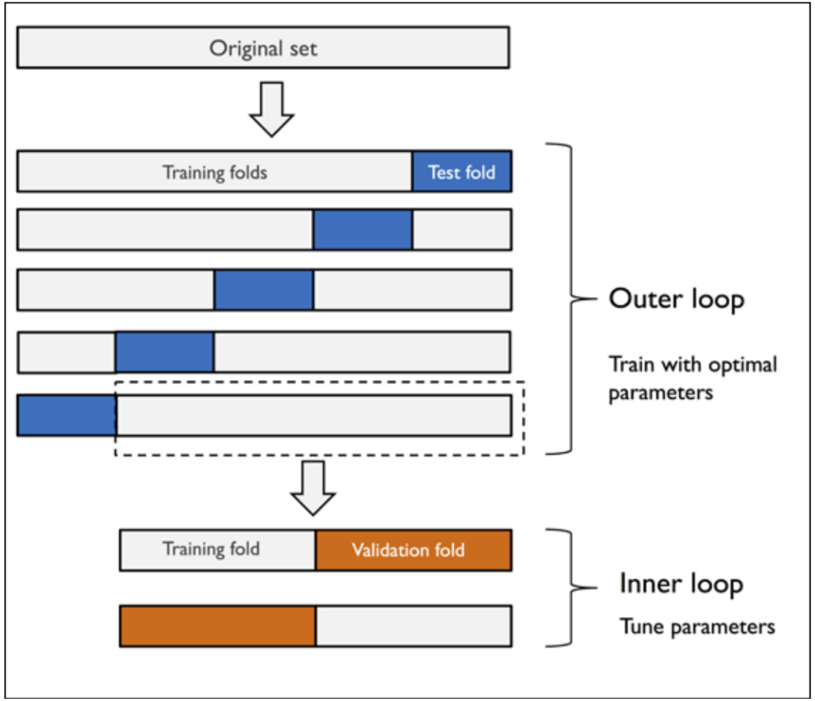


Training set
Training folds
Test fold
1st iteration
2nd iteration
3rd iteration
10th iteration
E_1
E_2
E_3
E_{10}
$E = \frac{1}{10}\sum_{i=1}^{10} E_i$

Review

- $K_1 = 5$

  $K_2 = 2$



Review

```
[ ]  # Decision tree
     from sklearn.tree import DecisionTreeClassifier
     from sklearn.model_selection import GridSearchCV
     from sklearn.model_selection import KFold
     inner_cv=KFold(n_splits=3, shuffle=True, random_state=0)
     outer_cv=KFold(n_splits=5, shuffle=True, random_state=0)
     gs = GridSearchCV(estimator=DecisionTreeClassifier(random_state=0),
                       param_grid=[{'max_depth': [1, 2, 3, 4, 5, 6, 7, None]}],
                       scoring='accuracy', cv=inner_cv)
     scores = cross_val_score(gs, X, y, scoring='accuracy', cv=outer_cv)
     print('CV accuracy: %.3f +/- %.3f' % (np.mean(scores), np.std(scores)))
```

```
CV accuracy: 0.942 +/- 0.012
```

Review

cv.glmnet {glmnet} R Documentation

Cross-validation for glmnet

Description

Does k-fold cross-validation for glmnet, produces a plot, and returns a value for `lambda` (and `gamma` if `relax=TRUE`)

Usage

```
cv.glmnet(x, y, weights = NULL, offset = NULL, lambda = NULL,
  type.measure = c("default", "mse", "deviance", "class", "auc", "mae",
  "C"), nfolds = 10, foldid = NULL, alignment = c("lambda",
  "fraction"), grouped = TRUE, keep = FALSE, parallel = FALSE,
  gamma = c(0, 0.25, 0.5, 0.75, 1), relax = FALSE, trace.it = 0, ...)
```

Lagrange Multiplier Theorem

- Primal Problem

$$\min_{\mathbf{x}} f(\mathbf{x})$$

$$\text{subject to} \quad g_i(\mathbf{x}) \leq 0, \quad \text{for} \quad i = 1, \cdots, m$$

$$h_j(\mathbf{x}) = 0, \quad \text{for} \quad j = 1, \cdots, \mathrm{k}$$

Lagrange Multiplier Theorem

- Dual Problem

$$\min_{\mathbf{x}} \quad f(\mathbf{x}) + \sum_{i}^{m} \alpha_i\, g_i(\mathbf{x}) + \sum_{j}^{\mathrm{k}} \gamma_j\, h_i(\mathbf{x})$$

$$\alpha_i \geq 0, \quad \text{for} \quad i = 1, \cdots, m$$

$$\gamma_j \geq 0, \quad \text{for} \quad j = 1, \cdots, \mathrm{k}$$

Karush-Kuhn-Tucker Conditions

1. $\nabla f(\mathbf{x}) + \sum_i^m \alpha_i \nabla g_i(\mathbf{x}) + \sum_j^{\mathrm{k}} \gamma_j \nabla h_i(\mathbf{x}) = 0$ (Stationary)

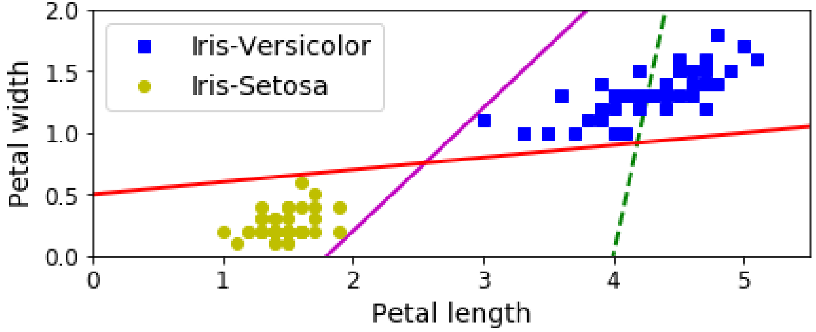
2. $\alpha_i g_i(\mathbf{x}) = 0, \quad$ for $\quad i = 1, \cdots, m$ (Complementary Slackness)

3. $g_i(\mathbf{x}) \leq 0, \quad$ for $\quad i = 1, \cdots, m \quad$ *and* (Primal Feasibility)

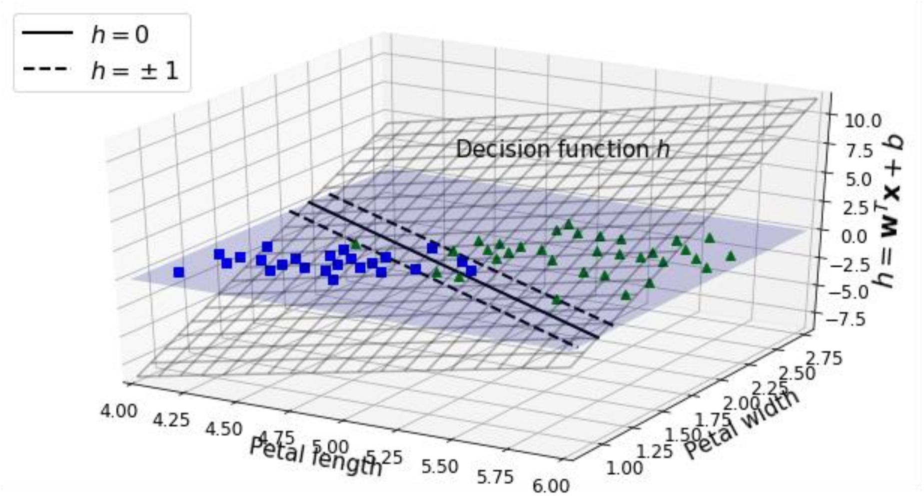
   $h_j(\mathbf{x}) = 0, \quad$ for $\quad j = 1, \cdots, \mathrm{k}$

4. $\alpha_i \geq 0, \quad$ for $\quad i = 1, \cdots, m$ (Dual Feasibility)

Hyperplane

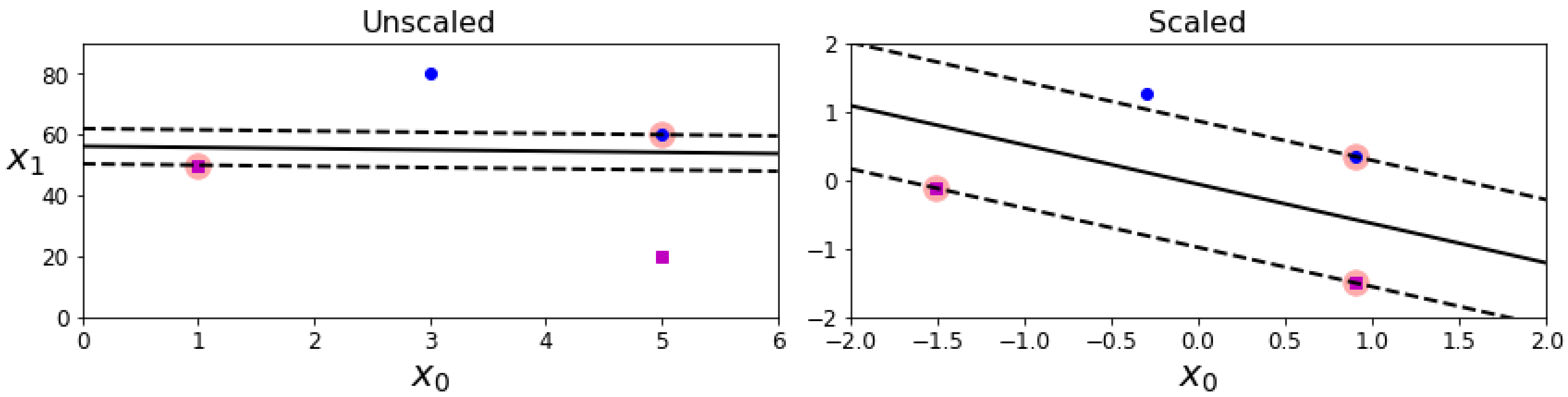


Iris-Versicolor
Iris-Setosa
Petal width
Petal length
2.0
1.5
1.0
0.5
0.0
0
1
2
3
4
5

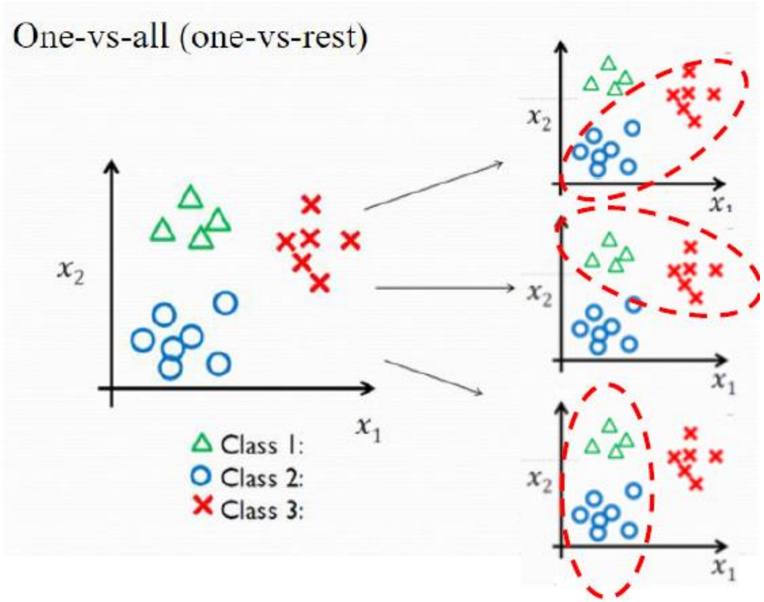


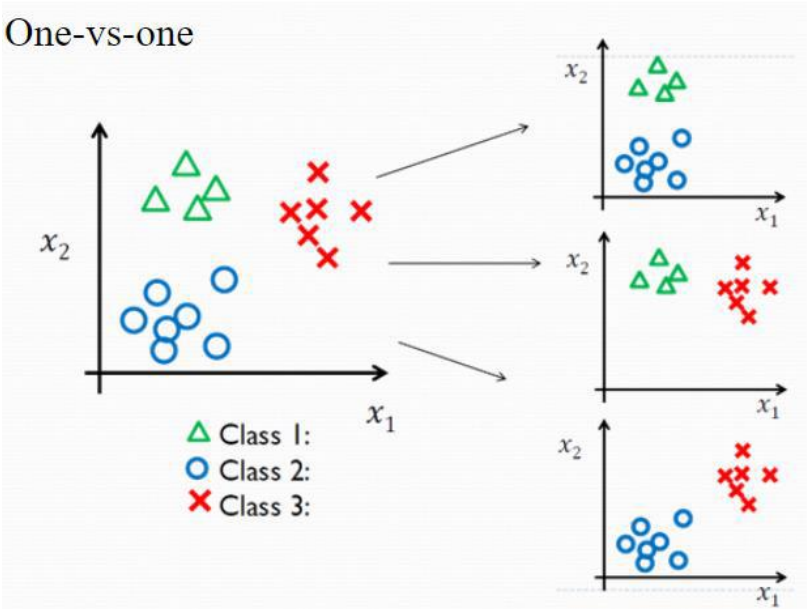
$h = 0$
$h = \pm 1$
Decision function h
$h = \mathbf{w}^T\mathbf{x} + b$
Petal length
Petal width
10.0
7.5
5.0
2.5
0.0
−2.5
−5.0
−7.5
4.00
4.25
4.50
4.75
5.00
5.25
5.50
5.75
6.00
1.00
1.25
1.50
1.75
2.00
2.25
2.50
2.75

Scaled? Unscaled?

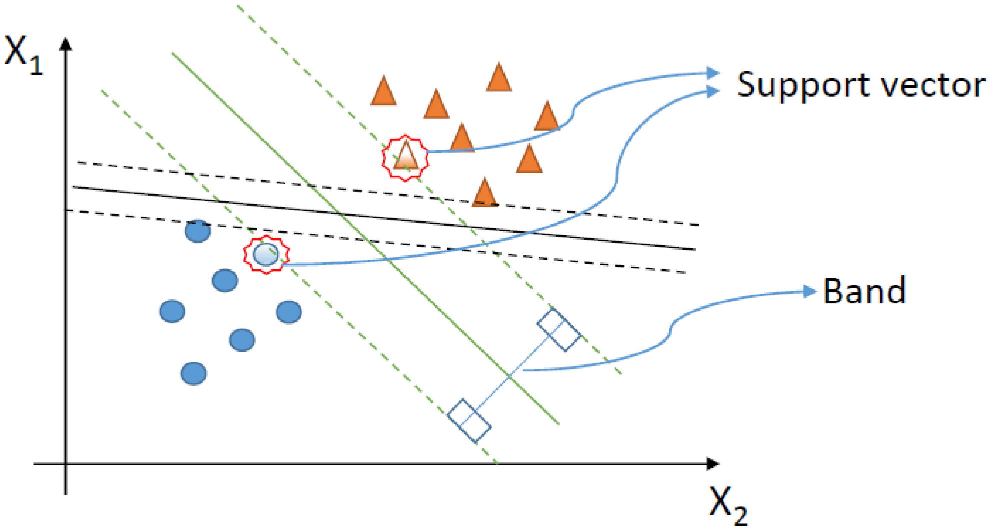


OVR and OVO


One-vs-all (one-vs-rest)
x_2
x_1
Class 1:
Class 2:
Class 3:

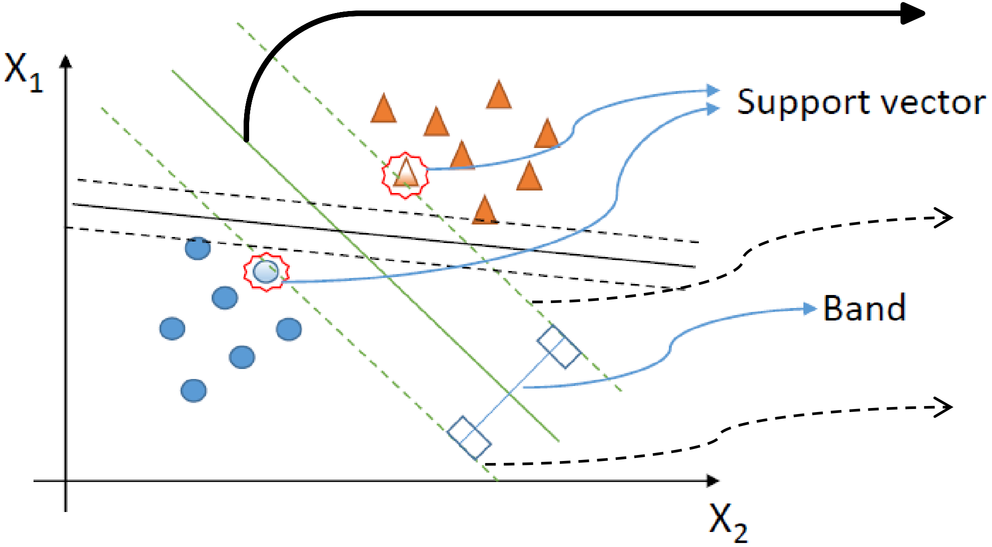

One-vs-one
x_2
x_1
Class 1:
Class 2:
Class 3:

Linear Support Vector Machine



$$y = \{-1, 1\}$$

Linear Support Vector Machine

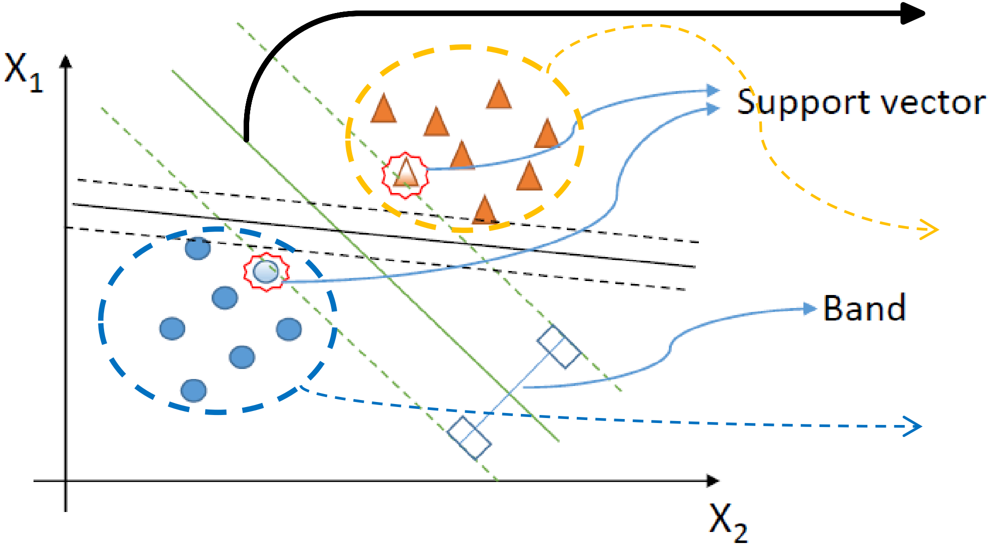


$$f(\mathbf{x}) = \beta_0 + \boldsymbol{\beta}^T \mathbf{X}_i = 0$$

$$\beta_0 + \boldsymbol{\beta}^T \mathbf{X}_+ = M$$

$$\beta_0 + \boldsymbol{\beta}^T \mathbf{X}_- = -M$$

Linear Support Vector Machine



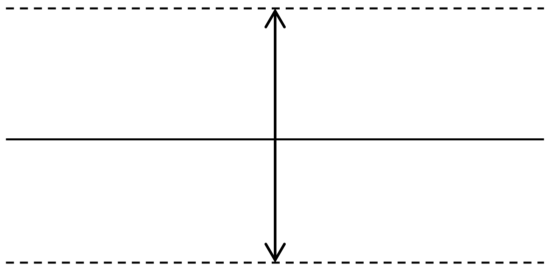
$$f(\mathbf{x}) = \beta_0 + \boldsymbol{\beta}^T \mathbf{X}_i = 0$$

$$\beta_0 + \boldsymbol{\beta}^T \mathbf{X}_i \geq M \qquad \text{if } \; y_i = 1$$

$$\beta_0 + \boldsymbol{\beta}^T \mathbf{X}_i \leq -M \qquad \text{if } \; y_i = -1$$

Linear Support Vector Machine

- We want to maximize the width of the band.



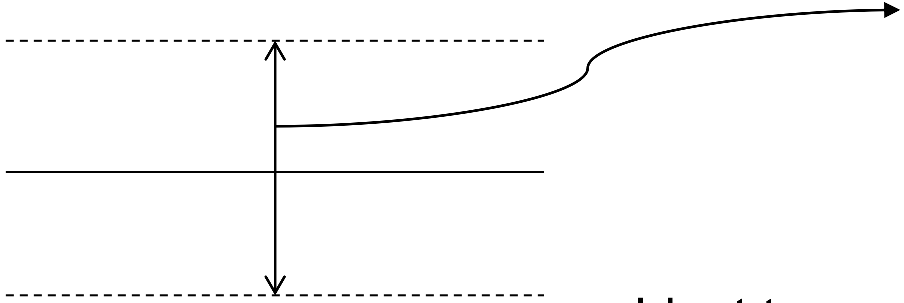
$$\beta_0 + \boldsymbol{\beta}^T \mathbf{X}_+ = M$$

$$\beta_0 + \boldsymbol{\beta}^T \mathbf{X}_i = 0$$

$$\beta_0 + \boldsymbol{\beta}^T \mathbf{X}_- = -M$$

Linear Support Vector Machine

- We want to maximize the width of the band.



$$\boldsymbol{\beta}^T(\mathbf{X}_+ - \mathbf{X}_-) = 2M$$

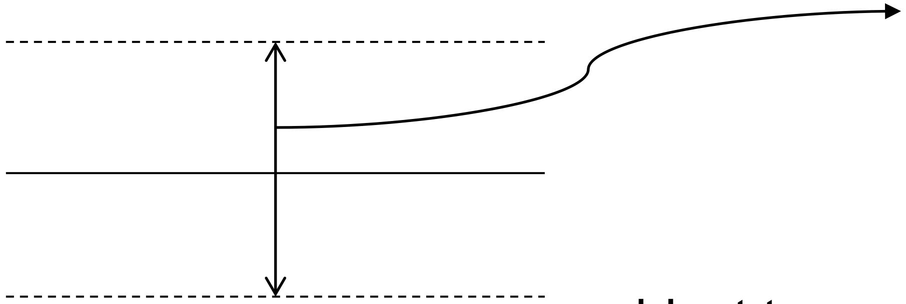
$$\max_{\beta_0,\, \boldsymbol{\beta}} M$$

subject to $y_i(\beta_0 + \boldsymbol{\beta}^T\mathbf{X}_i) \geq \mathrm{M}$, for $i = 1, \cdots, \mathrm{n}$

$$||\boldsymbol{\beta}|| = 1$$

Linear Support Vector Machine

- We want to maximize the width of the band.



$$\frac{\boldsymbol{\beta}^T}{||\boldsymbol{\beta}||}(\mathbf{X}_+ - \mathbf{X}_-) = \frac{2M}{||\boldsymbol{\beta}||}$$

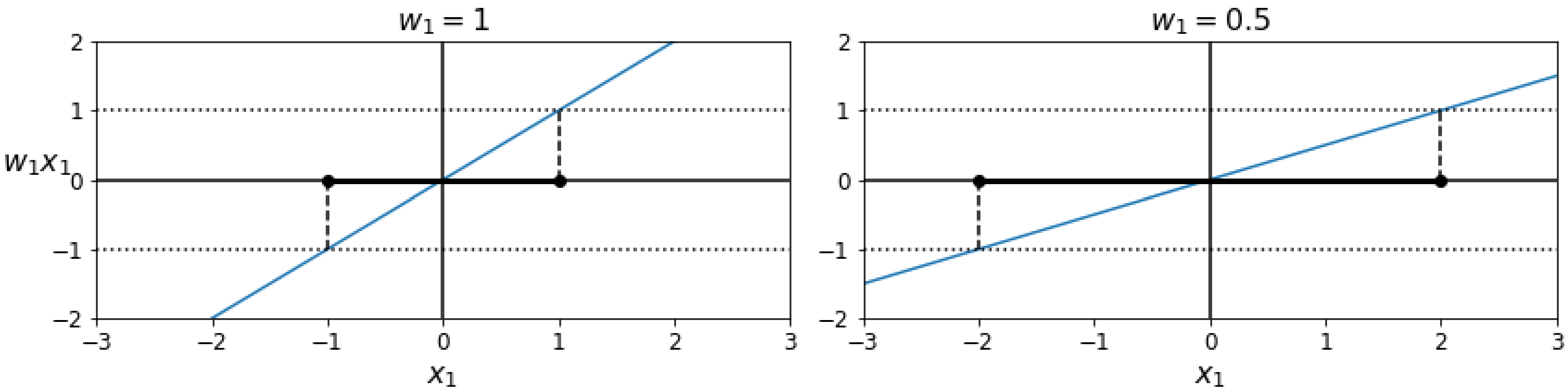
$$\max_{\beta_0,\,\boldsymbol{\beta}} M$$

subject to $y_i(\beta_0 + \boldsymbol{\beta}^T\mathbf{X}_i) \geq \mathrm{M}$, for $i = 1, \cdots, \mathrm{n}$

$$||\boldsymbol{\beta}|| = 1$$

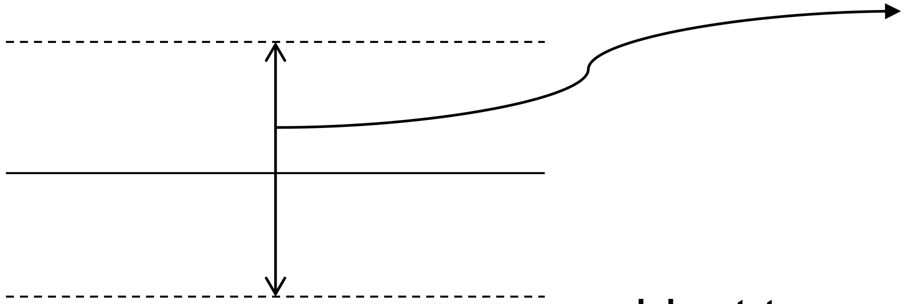
Linear Support Vector Machine

- A smaller weight vector results in a larger margin



Linear Support Vector Machine

- We want to maximize the width of the band.



$$\frac{\boldsymbol{\beta}^T}{||\boldsymbol{\beta}||}(\mathbf{X}_+ - \mathbf{X}_-) = \frac{2M}{||\boldsymbol{\beta}||}$$

$$\min_{\beta_0, \boldsymbol{\beta}} ||\boldsymbol{\beta}||$$

subject to $y_i(\beta_0 + \boldsymbol{\beta}^T \mathbf{X}_i) \geq \mathrm{M}$, for $i = 1, \cdots, \mathrm{n}$

$$\mathrm{M} = ?$$

Linear Support Vector Machine

- We want to maximize the width of the band.

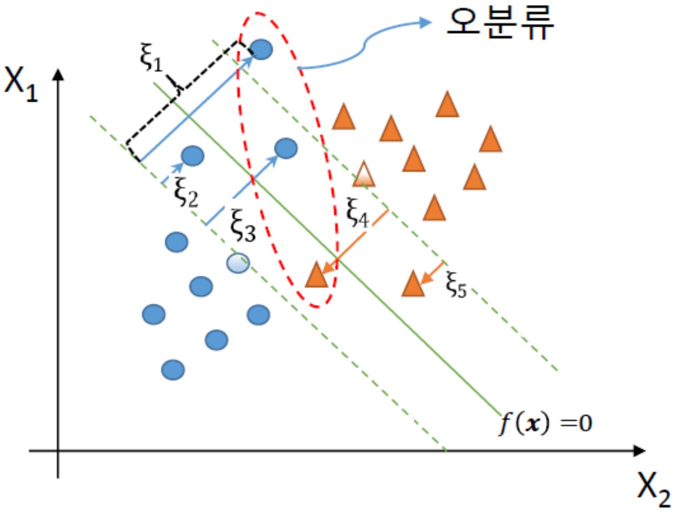
$$\max_{\beta_0, \boldsymbol{\beta}} M \quad \Longleftrightarrow \quad \min_{\beta_0, \boldsymbol{\beta}} ||\boldsymbol{\beta}||^2$$

subject to $y_i(\beta_0 + \boldsymbol{\beta}^T \mathbf{X}_i) \geq \mathrm{M}$, for $i = 1, \cdots, \mathrm{n}$ subject to $y_i(\beta_0 + \boldsymbol{\beta}^T \mathbf{X}_i) \geq 1$, for $i = 1, \cdots, \mathrm{n}$

and $||\boldsymbol{\beta}|| = 1$

Linear Support Vector Machine

- If the data are not perfectly separable, no solution exists.



Linear Support Vector Machine

- Hard Margin Classifier

$$\min_{\beta_0, \boldsymbol{\beta}} ||\boldsymbol{\beta}||^2$$

subject to $y_i(\beta_0 + \boldsymbol{\beta}^T \mathbf{X}_i) \geq 1$, for $i = 1, \cdots, \mathrm{n}$

$\Longrightarrow$

- Soft Margin Classifier

$$\min_{\beta_0, \boldsymbol{\beta}} ||\boldsymbol{\beta}||^2$$

subject to $y_i(\beta_0 + \boldsymbol{\beta}^T \mathbf{X}_i) \geq 1 - \zeta_i$

and $\zeta_i \geq 0,$

and $\sum_{\mathrm{i}}^{\mathrm{n}} \zeta_i \leq \tilde{C},$ for $i = 1, \cdots, \mathrm{n}$

Linear Support Vector Machine

- Primal Problem

$$\min_{\beta_0,\, \boldsymbol{\beta}} ||\boldsymbol{\beta}||^2$$

subject to $y_i(\beta_0 + \boldsymbol{\beta}^T \mathbf{X}_i) \geq 1 - \zeta_i$

and $\zeta_i \geq 0,$

and $\sum_{\mathrm{i}}^{\mathrm{n}} \zeta_i \leq \tilde{C},$ for $i = 1, \cdots, \mathrm{n}$

$$\Longleftrightarrow$$

- Dual Problem

$$\min_{\beta_0,\, \boldsymbol{\beta},\, \zeta_i} \quad ||\boldsymbol{\beta}||^2 + \mathrm{C} \sum_{\mathrm{i}}^{\mathrm{n}} \zeta_i$$

subject to $y_i(\beta_0 + \boldsymbol{\beta}^T \mathbf{X}_i) \geq 1 - \zeta_i$

and $\zeta_i \geq 0,$ for $i = 1, \cdots, \mathrm{n}$

C is not a Lagrange multiplier

Linear Support Vector Machine

- Primal Problem

$$\min_{\beta_0, \boldsymbol{\beta}, \zeta_i} \quad ||\boldsymbol{\beta}||^2 + \mathrm{C}\sum_{\mathrm{i}}^{\mathrm{n}} \zeta_i$$

subject to $\quad y_i(\beta_0 + \boldsymbol{\beta}^T \mathbf{X}_i) \geq 1 - \zeta_i$

and $\quad \zeta_i \geq 0, \quad$ for $i = 1, \cdots, \mathrm{n}$

$$\Longleftrightarrow$$

- Dual Problem

$$\min_{\beta_0, \boldsymbol{\beta}, \zeta_i} \quad ||\boldsymbol{\beta}||^2 + \mathrm{C}\sum_{\mathrm{i}}^{\mathrm{n}} \zeta_i - \sum_{\mathrm{i}}^{\mathrm{n}} \gamma_i \zeta_i$$

subject to $\quad y_i(\beta_0 + \boldsymbol{\beta}^T \mathbf{X}_i) \geq 1 - \zeta_i$

for $i = 1, \cdots, \mathrm{n}$

Linear Support Vector Machine

- Primal Problem

$$\min_{\beta_0, \boldsymbol{\beta}, \zeta_i} \quad ||\boldsymbol{\beta}||^2 + \mathrm{C}\sum_{\mathrm{i}}^{\mathrm{n}} \zeta_i - \sum_{\mathrm{i}}^{\mathrm{n}} \gamma_i \zeta_i$$

subject to $\quad y_i(\beta_0 + \boldsymbol{\beta}^T \mathbf{X}_i) \geq 1 - \zeta_i, \qquad$ for $i = 1, \cdots, \mathrm{n}$

Linear Support Vector Machine

- Dual Problem

$$\min_{\beta_0, \boldsymbol{\beta}, \zeta_i} \quad ||\boldsymbol{\beta}||^2 + \mathrm{C}\sum_{\mathrm{i}}^{\mathrm{n}} \zeta_i - \sum_{\mathrm{i}}^{\mathrm{n}} \gamma_i \zeta_i - \sum_{\mathrm{i}}^{\mathrm{n}} \alpha_i \left[y_i(\beta_0 + \boldsymbol{\beta}^T \mathbf{X}_i) - (1 - \zeta_i) \right]$$

- Taking derivative w.r.t β_0, $\boldsymbol{\beta}$, ζ_i

(Stationary)

Linear Support Vector Machine

$$\min_{\beta_0, \boldsymbol{\beta}, \zeta_i} \quad ||\boldsymbol{\beta}||^2 + \mathrm{C}\sum_{\mathrm{i}}^{\mathrm{n}} \zeta_i - \sum_{\mathrm{i}}^{\mathrm{n}} \gamma_i \zeta_i - \sum_{\mathrm{i}}^{\mathrm{n}} \alpha_i \left[y_i(\beta_0 + \boldsymbol{\beta}^T \mathbf{X}_i) - (1 - \zeta_i) \right]$$

(Stationary)

$$\begin{cases} \dfrac{\partial}{\partial \beta_0} \mathcal{L}_p: & \displaystyle\sum_i^n \alpha_i y_i = 0 \\ \dfrac{\partial}{\partial \boldsymbol{\beta}} \mathcal{L}_p: & \boldsymbol{\beta} = \displaystyle\sum_i^n \alpha_i y_i \mathbf{x}_i \\ \dfrac{\partial}{\partial \zeta_i} \mathcal{L}_p: & \alpha_i = C - \gamma_i \end{cases}$$

(Complementary Slackness)

$$\begin{cases} \alpha_i [y_i f(\mathbf{x}_i) - (1 - \zeta_i)] = 0 \\ \gamma_i \zeta_i = 0 \end{cases}$$

Linear Support Vector Machine

$$\min_{\beta_0, \boldsymbol{\beta}, \zeta_i} \quad ||\boldsymbol{\beta}||^2 + \mathrm{C}\sum_{\mathrm{i}}^{\mathrm{n}} \zeta_i - \sum_{\mathrm{i}}^{\mathrm{n}} \gamma_i \, \zeta_i - \sum_{\mathrm{i}}^{\mathrm{n}} \alpha_i \left[\, y_i(\beta_0 + \boldsymbol{\beta}^T \mathbf{X}_i) - (1 - \zeta_i)\right]$$

$$\Longleftrightarrow \qquad \max_{\alpha_i} \quad \sum_{\mathrm{i}}^{\mathrm{n}} \alpha_i + \frac{1}{2} \sum_{\mathrm{i}}^{\mathrm{n}} \sum_{\mathrm{j}}^{\mathrm{n}} \alpha_i \, \alpha_j \, y_i y_{\mathrm{j}} \mathbf{x}_i{}^T \mathbf{x}_{\mathrm{j}}$$

QP

subject to $\quad 0 \le \alpha_i \le C$

and $\quad \displaystyle\sum_{\mathrm{i}}^{\mathrm{n}} \alpha_i \, y_i = 0,\qquad$ for $i = 1, \cdots, \mathrm{n}$

Linear Support Vector Machine

$$\min_{\beta_0, \boldsymbol{\beta}, \zeta_i} \quad ||\boldsymbol{\beta}||^2 + \mathrm{C}\sum_{\mathrm{i}}^{\mathrm{n}} \zeta_i - \sum_{\mathrm{i}}^{\mathrm{n}} \gamma_i \zeta_i - \sum_{\mathrm{i}}^{\mathrm{n}} \alpha_i \left[y_i(\beta_0 + \boldsymbol{\beta}^T \mathbf{X}_i) - (1 - \zeta_i) \right]$$

$\Rightarrow$

$$\widehat{\boldsymbol{\beta}} = \sum_{i}^{n} \widehat{\alpha_i}\, y_i \mathbf{x}_i$$

$\widehat{\beta_0} = y_i - \widehat{\boldsymbol{\beta}}^T \mathbf{x}_\mathrm{k}$ for any support vector $\mathbf{x}_\mathrm{k}$

$$\widehat{f(\mathbf{x}_i)} = \widehat{\beta_0} + \widehat{\boldsymbol{\beta}}^T \mathbf{x}_\mathrm{k}$$

Linear Support Vector Machine

$$\min_{\beta_0, \boldsymbol{\beta}, \zeta_i} \quad ||\boldsymbol{\beta}||^2 + C\sum_{i}^{n} \zeta_i - \sum_{i}^{n} \gamma_i \zeta_i - \sum_{i}^{n} \alpha_i \left[y_i(\beta_0 + \boldsymbol{\beta}^T \mathbf{x}_i) - (1 - \zeta_i) \right]$$

$$\Longleftrightarrow \qquad \max_{\alpha_i} \quad \sum_{i}^{n} \alpha_i + \frac{1}{2} \sum_{i}^{n} \sum_{j}^{n} \alpha_i \, \alpha_j \, y_i y_j \mathbf{x}_i{}^T \mathbf{x}_j \qquad \text{QP}$$

subject to $\quad 0 \leq \alpha_i \leq C$

and $\quad \sum_{i}^{n} \alpha_i \, y_i = 0, \qquad$ for $i = 1, \cdots, n$

Kernel Support Vector Machine

$$\min_{\beta_0, \boldsymbol{\beta}, \zeta_i} \quad ||\boldsymbol{\beta}||^2 + C\sum_{i}^{n} \zeta_i - \sum_{i}^{n} \gamma_i \zeta_i - \sum_{i}^{n} \alpha_i \left[y_i(\beta_0 + \boldsymbol{\beta}^T \mathbf{X}_i) - (1 - \zeta_i) \right]$$

$$\Longleftrightarrow \qquad \max_{\alpha_i} \quad \sum_{i}^{n} \alpha_i + \frac{1}{2}\sum_{i}^{n}\sum_{j}^{n} \alpha_i \, \alpha_j \, y_i y_j K(\mathbf{x}_i, \mathbf{x}_j)$$

subject to $\quad 0 \le \alpha_i \le C$

and $\quad \sum_{i}^{n} \alpha_i \, y_i = 0, \qquad$ for $i = 1, \cdots, n$

Kernel Support Vector Machine

- Kernel function

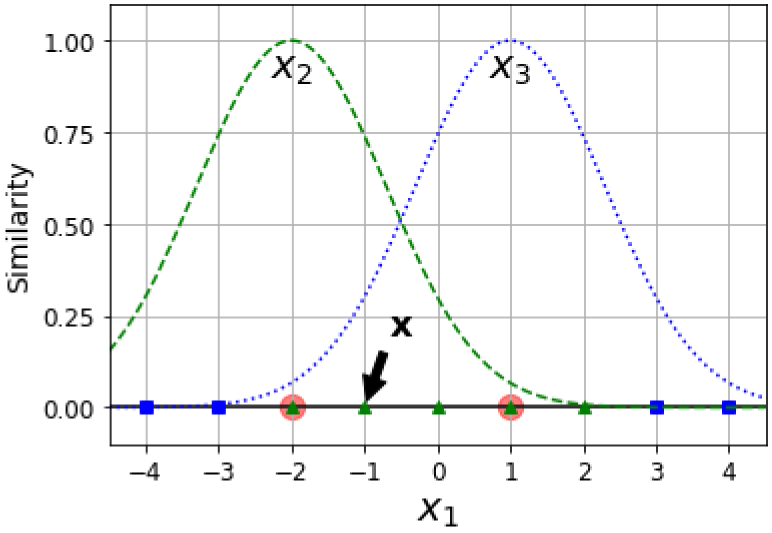
$$K(\mathbf{x}_i, \mathbf{x}_j) = \mathbf{x}_i^T \mathbf{x}_j$$ *Linear Kernel*

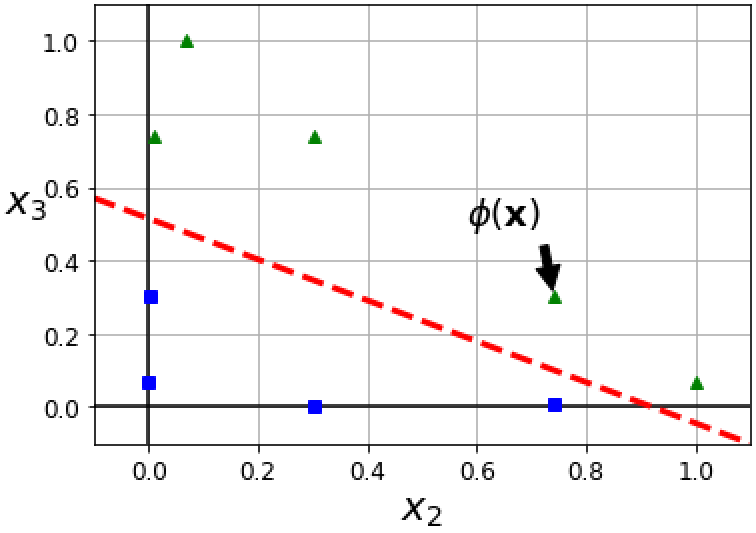
$$K(\mathbf{x}_i, \mathbf{x}_j) = \exp(-\gamma(\mathbf{x}_i - \mathbf{x}_j)^T(\mathbf{x}_i - \mathbf{x}_j))$$ *Gaussian Kernel (Radial Basis function)*

$$K(\mathbf{x}_i, \mathbf{x}_j) = (\gamma + \gamma\, \mathbf{x}_i^T \mathbf{x}_j)^p$$ *polynomial Kernel*

$$K(\mathbf{x}_i, \mathbf{x}_j) = \tanh(k_1 \mathbf{x}_i^T \mathbf{x}_j + k_2)$$ *Sigmoid Kernel*

Kernel Support Vector Machine

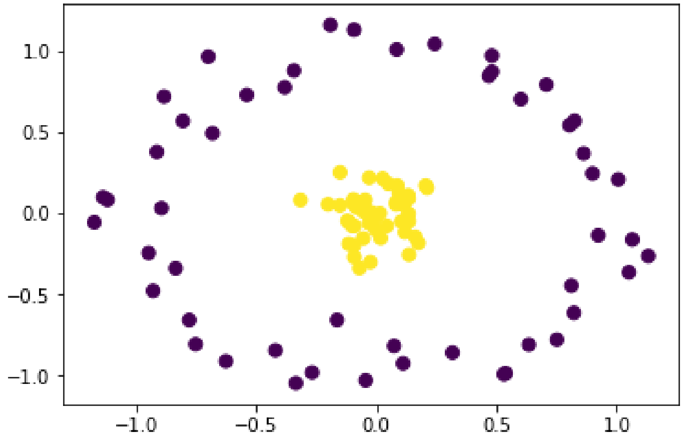




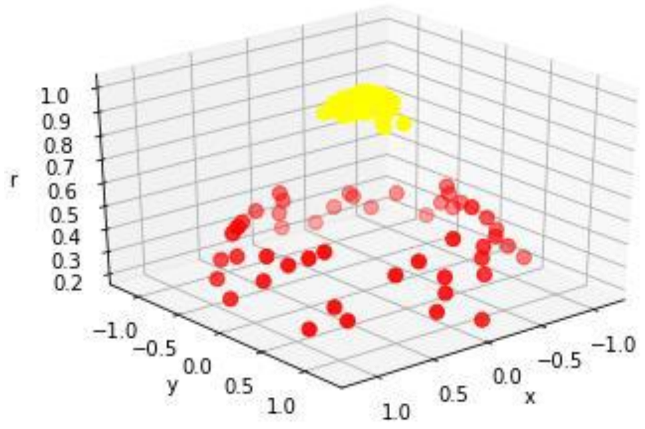
x_1

$$x_2 = \exp(-(x_1 - 2)^2)$$
$$x_3 = \exp(-(x_1 + 1)^2)$$

Kernel Support Vector Machine



(x_1, x_2)



$(x_1, x_2, \exp(-(x_1{}^2 + x_2{}^2)))$

Kernel Support Vector Machine

$$\min_{\beta_0, \boldsymbol{\beta}, \zeta_i} \; ||\boldsymbol{\beta}||^2 + C\sum_{i}^{n} \zeta_i - \sum_{i}^{n} \gamma_i \zeta_i - \sum_{i}^{n} \alpha_i \left[y_i(\beta_0 + \boldsymbol{\beta}^T \mathbf{X}_i) - (1 - \zeta_i) \right]$$

$$\Longleftrightarrow \quad \max_{\alpha_i} \; \sum_{i}^{n} \alpha_i + \frac{1}{2} \sum_{i}^{n} \sum_{j}^{n} \alpha_i \alpha_j y_i y_j \mathrm{h}(\mathbf{x}_i)^T h(\mathbf{x}_j)$$

subject to $0 \le \alpha_i \le C$

and $\sum_{i}^{n} \alpha_i y_i = 0$, for $i = 1, \cdots, n$

Kernel Trick

$$\min_{\beta_0, \boldsymbol{\beta}, \zeta_i} \quad ||\boldsymbol{\beta}||^2 + \mathrm{C}\sum_{\mathrm{i}}^{\mathrm{n}} \zeta_i - \sum_{\mathrm{i}}^{\mathrm{n}} \gamma_i \zeta_i - \sum_{\mathrm{i}}^{\mathrm{n}} \alpha_i \left[y_i(\beta_0 + \boldsymbol{\beta}^T \mathbf{X}_i) - (1 - \zeta_i) \right]$$

$$\Longleftrightarrow \qquad \max_{\alpha_i} \quad \sum_{\mathrm{i}}^{\mathrm{n}} \alpha_i + \frac{1}{2} \sum_{\mathrm{i}}^{\mathrm{n}} \sum_{\mathrm{j}}^{\mathrm{n}} \alpha_i \, \alpha_j \, y_i y_{\mathrm{j}} K(\mathbf{x}_i, \mathbf{x}_j)$$

subject to $0 \le \alpha_i \le C$

and $\sum_{\mathrm{i}}^{\mathrm{n}} \alpha_i \, y_i = 0,$ for $i = 1, \cdots, \mathrm{n}$

Kernel trick

- 특성함수의 생성 어려움 + 고차원 확장시 차원의 저주 문제 발생.
- 2차 다항커널 : 입력변수 x_1과 x_2이고 i번째 관측치와 j번째 관측치일때,

$$
\begin{aligned}
K(\boldsymbol{x}_i, \boldsymbol{x}_j) &= (1 + \boldsymbol{x}_i^T \boldsymbol{x}_j)^2 \\
&= (1 + x_{i,1}x_{j,1} + x_{i,2}x_{j,2})^2 \\
&= 1 + 2x_{i,1}x_{j,1} + 2x_{i,2}x_{j,2} + (x_{i,1}x_{j,1})^2 + (x_{i,2}x_{j,2})^2 + 2x_{i,1}x_{j,1}x_{i,2}x_{j,2}
\end{aligned}
\tag{7.11}
$$

- 이때 다음과 같이 정의하면,

$h_1(x_1,x_2) = 1$, $h_2(x_1,x_2) = \sqrt{2}x_1$, $h_3(x_1,x_2) = \sqrt{2}x_2$, $h_4(x_1,x_2) = x_1^2$, $h_5(x_1,x_2) = x_2^2$, $h_6(x_1,x_2) = \sqrt{2}x_1x_2$

$$\boldsymbol{h}(x_1,x_2) = (h_1(x_1,x_2), h_2(x_1,x_2), \cdots, h_6(x_1,x_2))^T$$

- 식 (7.11)은 $K(\boldsymbol{x}_i, \boldsymbol{x}_j) = (1 + \boldsymbol{x}_i^T \boldsymbol{x}_j)^2 = \boldsymbol{h}(\boldsymbol{x}_i)^T \boldsymbol{h}(\boldsymbol{x}_j)$ 로 변형 가능.
- 특성함수를 정의하지 않고 커널 함수를 이용.
- 즉, $\hat{\beta}$이 $\boldsymbol{h}(\boldsymbol{x}_i)^T \boldsymbol{h}(\boldsymbol{x}_j)$의 형태이면, $K(\boldsymbol{x}_i, \boldsymbol{x}_j)$를 직접 이용하여 추정.

β_0와 β의 추정 – *by kernel trick*

- 특성변수 x로 부터 basis함수 $\boldsymbol{h}(\boldsymbol{x})$로 차원을 증대시키면 커널 SVM 목적함수.

$$L_k = \sum_{i=1}^{n} \alpha_i - \frac{1}{2} \sum_{i=1}^{n} \sum_{i=1}^{n} \alpha_i \alpha_j y_i y_j \boldsymbol{h}(\boldsymbol{x}_i)^T \boldsymbol{h}(\boldsymbol{x}_j) \qquad (7.12)$$

- 선형 SVM 식 (7.11)은 $\widehat{f}(\boldsymbol{x}) = \widehat{\beta}_0 + \widehat{\boldsymbol{\beta}}^T \boldsymbol{x} = \widehat{\beta}_0 + \sum_{i=1}^{n} \widehat{\alpha}_i y_i \boldsymbol{x}_i^T \boldsymbol{x}$ 로 변형 가능.

- L_k 최소화한 모수 추정치를 $\widehat{\beta}_0^*$와 $\widehat{\boldsymbol{\beta}}^*$ 라 할 때 커널 SVM의 예측치

$$\widehat{f}(\boldsymbol{x}) = \widehat{\beta}_0^* + \sum_{i=1}^{n} \widehat{\alpha}_i^* y_i \boldsymbol{h}(\boldsymbol{x}_i)^T \boldsymbol{h}(\boldsymbol{x}) \qquad (7.13)$$

- 식(7.12)와 식(7.13) 모두 $\boldsymbol{h}(\boldsymbol{x}_i)^T \boldsymbol{h}(\boldsymbol{x}_j)$의 형태임.

- 식(7.12)에 $\boldsymbol{h}(\boldsymbol{x}_i)^T \boldsymbol{h}(\boldsymbol{x}_j)$ 대신 커널 함수 $K(\boldsymbol{x}_i, \boldsymbol{x})$를 대체하여 $\widehat{\beta}_0^*$와 $\widehat{\boldsymbol{\beta}}^*$ 를 추정.

- 식(7.13)도 $\boldsymbol{h}(\boldsymbol{x}_i)^T \boldsymbol{h}(\boldsymbol{x}_j)$를 이용하여 동일한 커널 SVM을 구함.

Hinge Loss

$$\min_{\beta_0, \boldsymbol{\beta}, \zeta_i} \quad ||\boldsymbol{\beta}||^2 + \mathrm{C}\sum_{\mathrm{i}}^{\mathrm{n}} \zeta_i - \sum_{\mathrm{i}}^{\mathrm{n}} \gamma_i \zeta_i - \sum_{\mathrm{i}}^{\mathrm{n}} \alpha_i \left[y_i(\beta_0 + \boldsymbol{\beta}^T \mathbf{X}_i) - (1 - \zeta_i) \right]$$

$$\Longleftrightarrow \quad \min_{\beta_0, \boldsymbol{\beta}, \zeta_i} \quad ||\boldsymbol{\beta}||^2 + \mathrm{C}\sum_{\mathrm{i}}^{\mathrm{n}} \zeta_i$$

subject to $y_i(\beta_0 + \boldsymbol{\beta}^T \mathbf{X}_i) \geq 1 - \zeta_i$

and $\zeta_i \geq 0, \quad$ for $i = 1, \cdots, \mathrm{n}$

Hinge Loss

$$\min_{\beta_0, \boldsymbol{\beta}, \zeta_i} \quad ||\boldsymbol{\beta}||^2 + \mathrm{C}\sum_{\mathrm{i}}^{\mathrm{n}} \zeta_i - \sum_{\mathrm{i}}^{\mathrm{n}} \gamma_i \zeta_i - \sum_{\mathrm{i}}^{\mathrm{n}} \alpha_i \left[y_i(\beta_0 + \boldsymbol{\beta}^T \mathbf{X}_i) - (1 - \zeta_i) \right]$$

$$\Longleftrightarrow \qquad \min_{\beta_0, \boldsymbol{\beta}, \zeta_i} \quad ||\boldsymbol{\beta}||^2 + \mathrm{C}\sum_{\mathrm{i}}^{\mathrm{n}} \zeta_i$$

subject to $\zeta_i \geq 1 - y_i(\beta_0 + \boldsymbol{\beta}^T \mathbf{X}_i)$

and $\zeta_i \geq 0, \quad$ for $i = 1, \cdots, \mathrm{n}$

Hinge Loss

$$\min_{\beta_0,\,\boldsymbol{\beta},\,\zeta_i} \quad ||\boldsymbol{\beta}||^2 + \mathrm{C}\sum_{\mathrm{i}}^{\mathrm{n}} \zeta_i - \sum_{\mathrm{i}}^{\mathrm{n}} \gamma_i \zeta_i - \sum_{\mathrm{i}}^{\mathrm{n}} \alpha_i \left[y_i(\beta_0 + \boldsymbol{\beta}^T \mathbf{X}_i) - (1 - \zeta_i) \right]$$

$$\Longleftrightarrow \quad \min_{\beta_0,\,\boldsymbol{\beta},\,\zeta_i} \quad ||\boldsymbol{\beta}||^2 + \mathrm{C}\sum_{\mathrm{i}}^{\mathrm{n}} \zeta_i$$

subject to $\zeta_i \geq [1 - y_i(\beta_0 + \boldsymbol{\beta}^T \mathbf{X}_i)]_+$ for $i = 1, \cdots, \mathrm{n}$

Hinge Loss

$$\min_{\beta_0, \boldsymbol{\beta}, \zeta_i} \quad ||\boldsymbol{\beta}||^2 + \mathrm{C}\sum_{\mathrm{i}}^{\mathrm{n}} \zeta_i - \sum_{\mathrm{i}}^{\mathrm{n}} \gamma_i \zeta_i - \sum_{\mathrm{i}}^{\mathrm{n}} \alpha_i \left[y_i(\beta_0 + \boldsymbol{\beta}^T \mathbf{X}_i) - (1 - \zeta_i) \right]$$

$$\Longleftrightarrow \qquad \min_{\beta_0, \boldsymbol{\beta}} \quad ||\boldsymbol{\beta}||^2 + \mathrm{C}\sum_{\mathrm{i}}^{\mathrm{n}} [1 - y_i(\beta_0 + \boldsymbol{\beta}^T \mathbf{X}_i)]_+$$

Hinge Loss

$$\min_{\beta_0, \boldsymbol{\beta}, \zeta_i} \quad ||\boldsymbol{\beta}||^2 + C\sum_{i}^{n} \zeta_i - \sum_{i}^{n} \gamma_i \zeta_i - \sum_{i}^{n} \alpha_i \left[y_i(\beta_0 + \boldsymbol{\beta}^T \mathbf{X}_i) - (1 - \zeta_i) \right]$$

$$\Longleftrightarrow \quad \min_{\boldsymbol{\beta}} \quad \frac{1}{C}||\boldsymbol{\beta}||^2 + \sum_{i}^{n} [1 - y_i f(\mathbf{x}_i)]_+$$

Hinge Loss

$$\min_{\beta_0, \boldsymbol{\beta}, \zeta_i} \quad ||\boldsymbol{\beta}||^2 + \mathrm{C}\sum_{\mathrm{i}}^{\mathrm{n}} \zeta_i - \sum_{\mathrm{i}}^{\mathrm{n}} \gamma_i \zeta_i - \sum_{\mathrm{i}}^{\mathrm{n}} \alpha_i \left[y_i(\beta_0 + \boldsymbol{\beta}^T \mathbf{X}_i) - (1 - \zeta_i) \right]$$

$$\Longleftrightarrow \quad \min_{\boldsymbol{\beta}} \quad \lambda||\boldsymbol{\beta}||^2 + \sum_{\mathbf{i}}^{\mathbf{n}} [1 - y_i f(\mathbf{x}_i)]_+$$

$$\frac{1}{\mathrm{C}} = \lambda$$

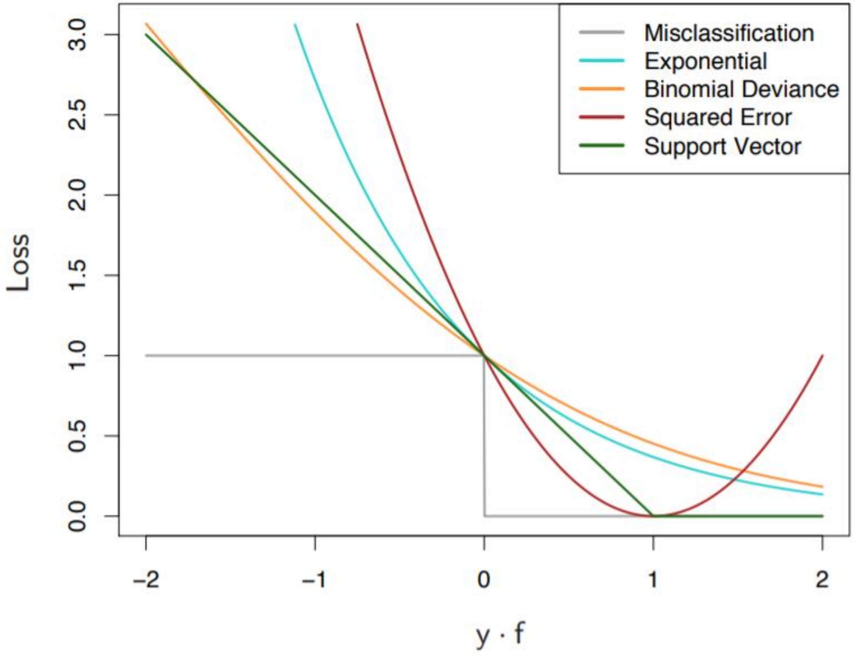
Hinge Loss

$$\min_{\beta_0, \boldsymbol{\beta}, \zeta_i} \; ||\boldsymbol{\beta}||^2 + C\sum_{i}^{n} \zeta_i - \sum_{i}^{n} \gamma_i \zeta_i - \sum_{i}^{n} \alpha_i \left[y_i(\beta_0 + \boldsymbol{\beta}^T \mathbf{X}_i) - (1 - \zeta_i) \right]$$

$\Longleftrightarrow$ $$\min_{\boldsymbol{\beta}} \; \sum_{i}^{n} [1 - y_i f(\mathbf{x}_i)]_+ + \lambda ||\boldsymbol{\beta}||^2$$

$\Longrightarrow$ Expression of " Loss + Penalty "

Hinge Loss



Misclassification
Exponential
Binomial Deviance
Squared Error
Support Vector
Loss
0.0
0.5
1.0
1.5
2.0
2.5
3.0
−2
−1
0
1
2
y · f

Grid Search for SVM

```
### Grid search에 의한 초모수 결정 (SVM) ###
from sklearn.model_selection import GridSearchCV
from sklearn.svm import SVC
pipe_svc = make_pipeline(StandardScaler(), SVC(random_state=1))
param_range = [0.0001, 0.001, 0.01, 0.1, 1.0, 10.0, 100.0, 1000.0]
param_grid = [{'svc__C': param_range, 'svc__kernel': ['linear']},
              {'svc__C': param_range, 'svc__gamma': param_range,
               'svc__kernel': ['rbf']},
              {'svc__C': param_range, 'svc__degree': [2,3,4,5],
               'svc__kernel': ['poly']}]
gs = GridSearchCV(estimator=pipe_svc, param_grid=param_grid,
                  scoring='accuracy', cv=10)
gs = gs.fit(X_train, y_train)
print(gs.best_score_)
print(gs.best_params_)

clf = gs.best_estimator_
clf.fit(X_train, y_train)
clf.score(X_train,y_train)
clf.score(X_test, y_test)
```

Support Vector Regression

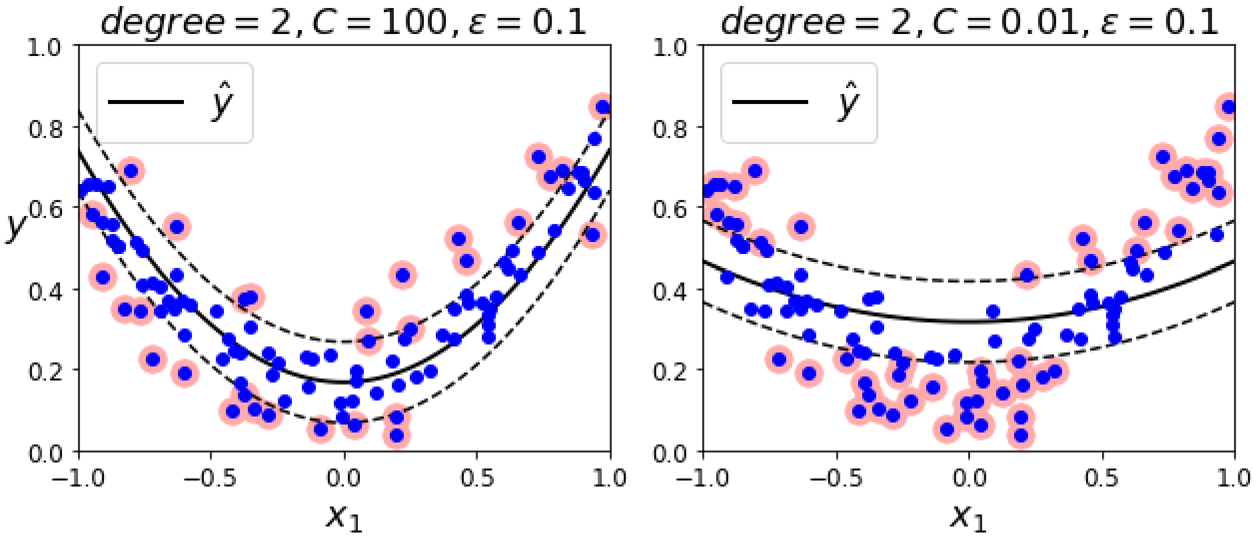
$$\min_{\boldsymbol{\beta}} \sum_{i}^{n} \mathrm{L}_{\epsilon}[y_i - f(\mathbf{x}_i)] + \lambda \boldsymbol{\beta}^{\mathrm{T}} \boldsymbol{\beta}$$

Linear SVR

$$\min_{\boldsymbol{\beta}} \sum_{i}^{n} \mathrm{L}_{\epsilon}[y_i - f(\mathbf{x}_i)] + \lambda \boldsymbol{\beta}^{\mathrm{T}} \mathbf{K}\, \boldsymbol{\beta}$$

Kernel SVR

Support Vector Regression



Support Vector Regression

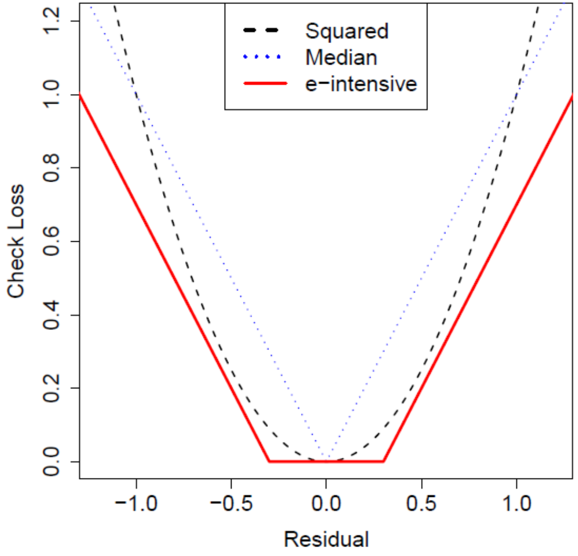


Figure: ϵ-intensive loss for SVR.

reference

자료

19-2 STAT424 통계적 머신러닝 - 박유성 교수님

교재

파이썬을 이용한 통계적 머신러닝 (2020) – 박유성

ISLR (2013) - G. James, D. Witten, T. Hastie, R. Tibshirani

The elements of Statistical Learning (2001) - J. Friedman, T. Hastie, R. Tibshirani

Hands on Machine Learning (2017) - Aurelien Geron