LOGO

BigData Engineering

11주차: Supervised Learning

강의 : 신경섭

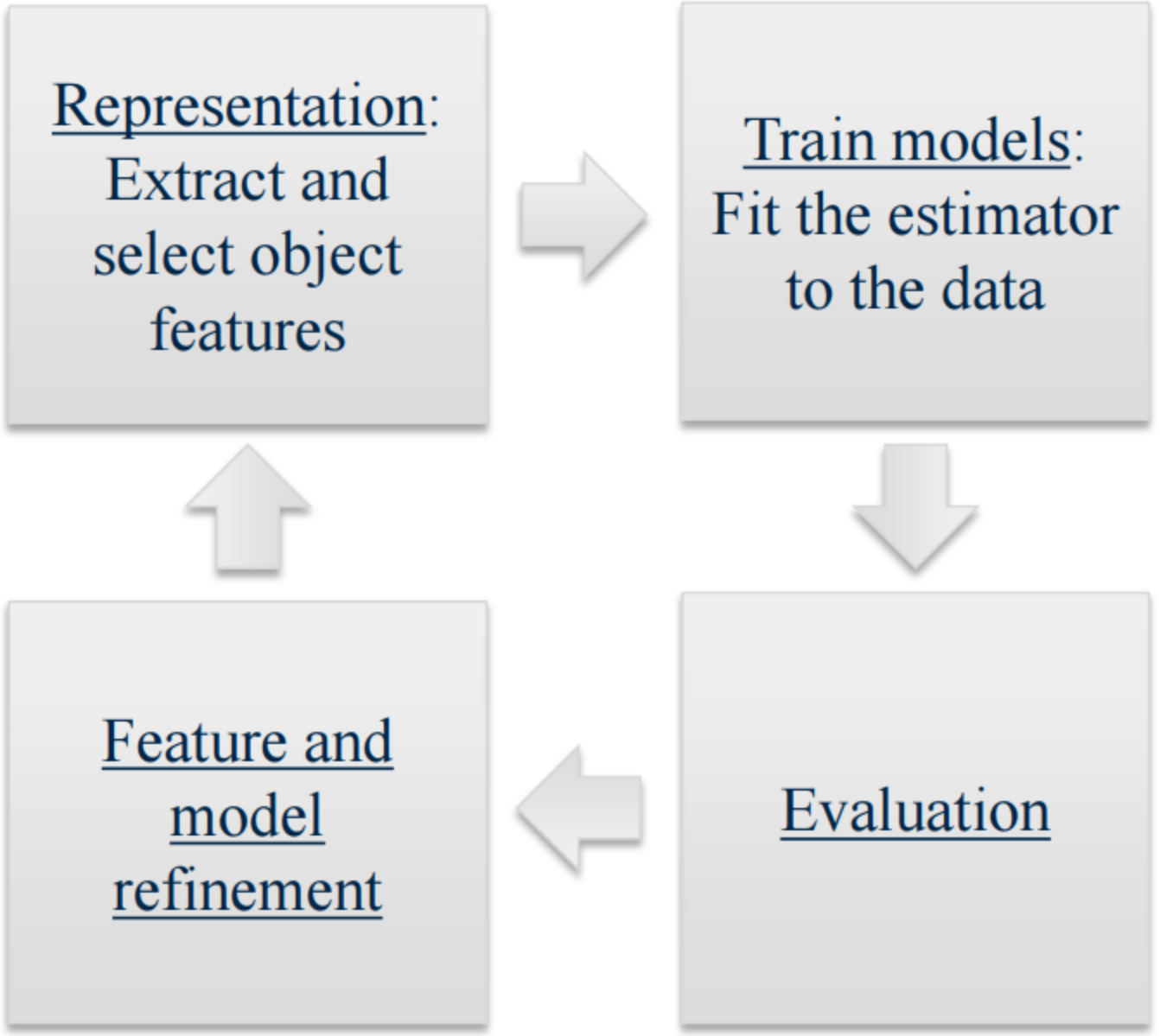
We covered…

- Data management/Visualization by python
  - Numpy, pandas, data acquisition
- Machine learning workflow with data
- EDA (Exploratory Data Analysis)
- Supervised learning
  - Examples – k-NN classifier

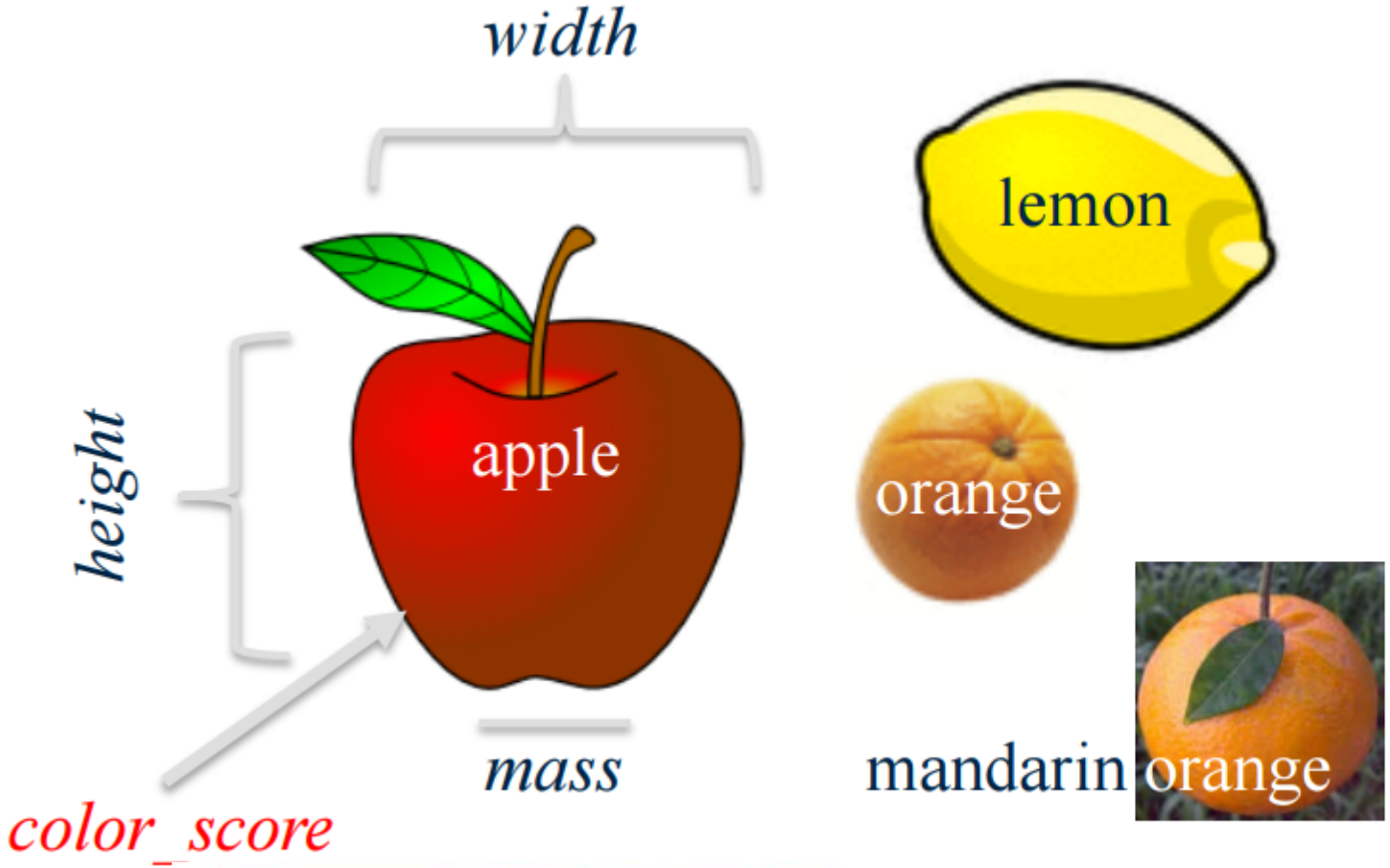
Today's Subjects

- Supervised learning
  - Regression : linear regression
  - Classification
    - logistic regression based binary classification
    - Support vector machine
    - Decision tree
    - Random Forest

Represent / Train / Evaluate / Refine Cycle



Fruit Dataset

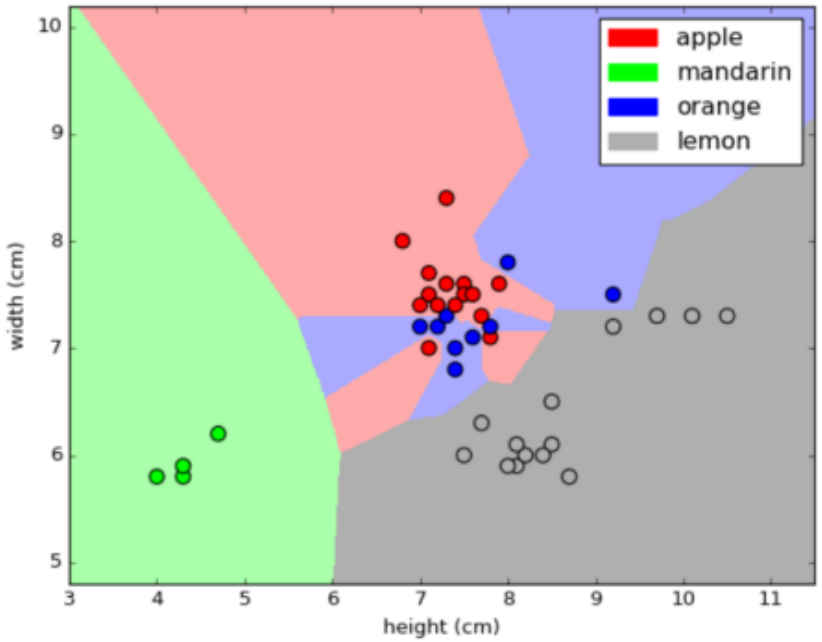


	fruit_label	fruit_name	fruit_subtype	mass	width	height	color_score
0	1	apple	granny_smith	192	8.4	7.3	0.55
1	1	apple	granny_smith	180	8.0	6.8	0.59
2	1	apple	granny_smith	176	7.4	7.2	0.60
3	2	mandarin	mandarin	86	6.2	4.7	0.80
4	2	mandarin	mandarin	84	6.0	4.6	0.79
5	2	mandarin	mandarin	80	5.8	4.3	0.77
6	2	mandarin	mandarin	80	5.9	4.3	0.81
7	2	mandarin	mandarin	76	5.8	4.0	0.81
8	1	apple	braeburn	178	7.1	7.8	0.92
9	1	apple	braeburn	172	7.4	7.0	0.89
10	1	apple	braeburn	166	6.9	7.3	0.93
11	1	apple	braeburn	172	7.1	7.6	0.92
12	1	apple	braeburn	154	7.0	7.1	0.88
13	1	apple	golden_delicious	164	7.3	7.7	0.70
14	1	apple	golden_delicious	152	7.6	7.3	0.69
15	1	apple	golden_delicious	156	7.7	7.1	0.69
16	1	apple	golden_delicious	156	7.6	7.5	0.67

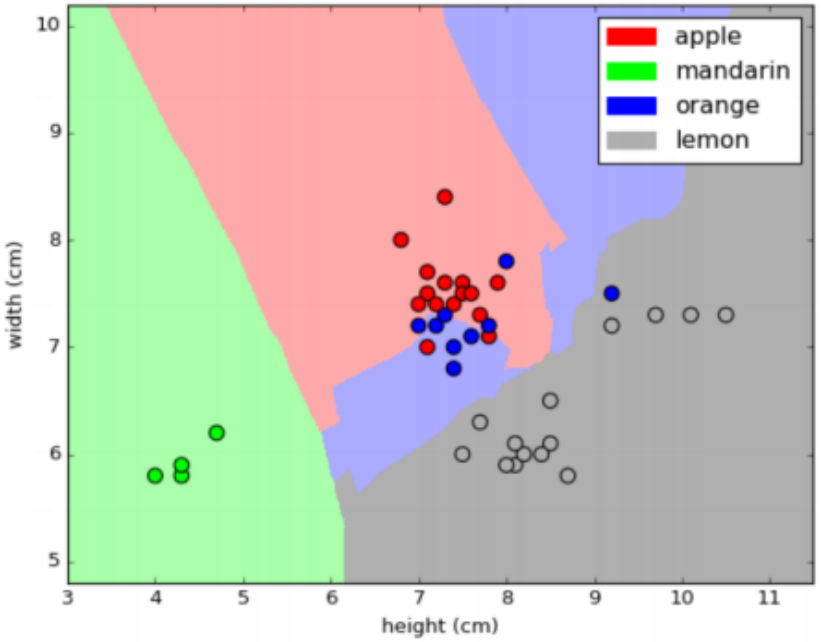
fruit_data_with_colors.txt

The k-Nearest Neighbor(k-NN) Algorithm

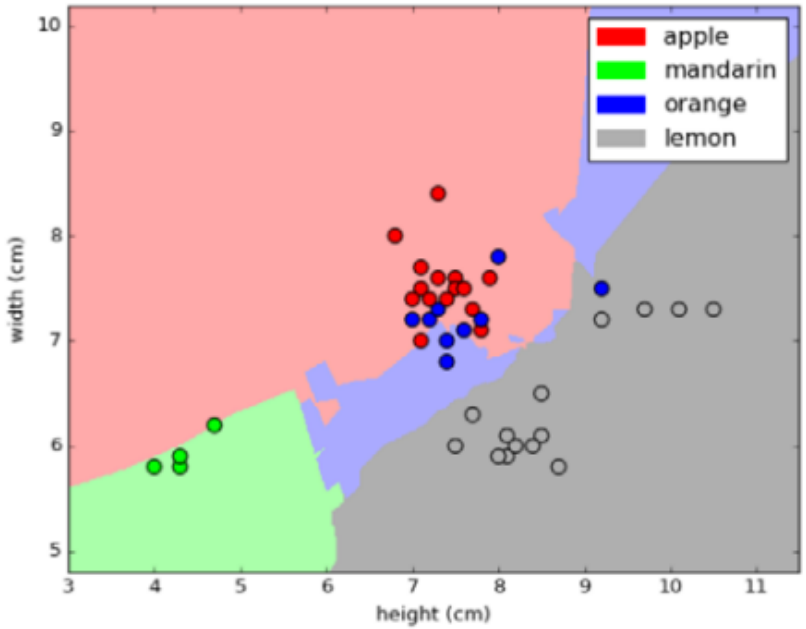
- **Find k nearest neighbor data to classification**



K=1



K=5



K=10

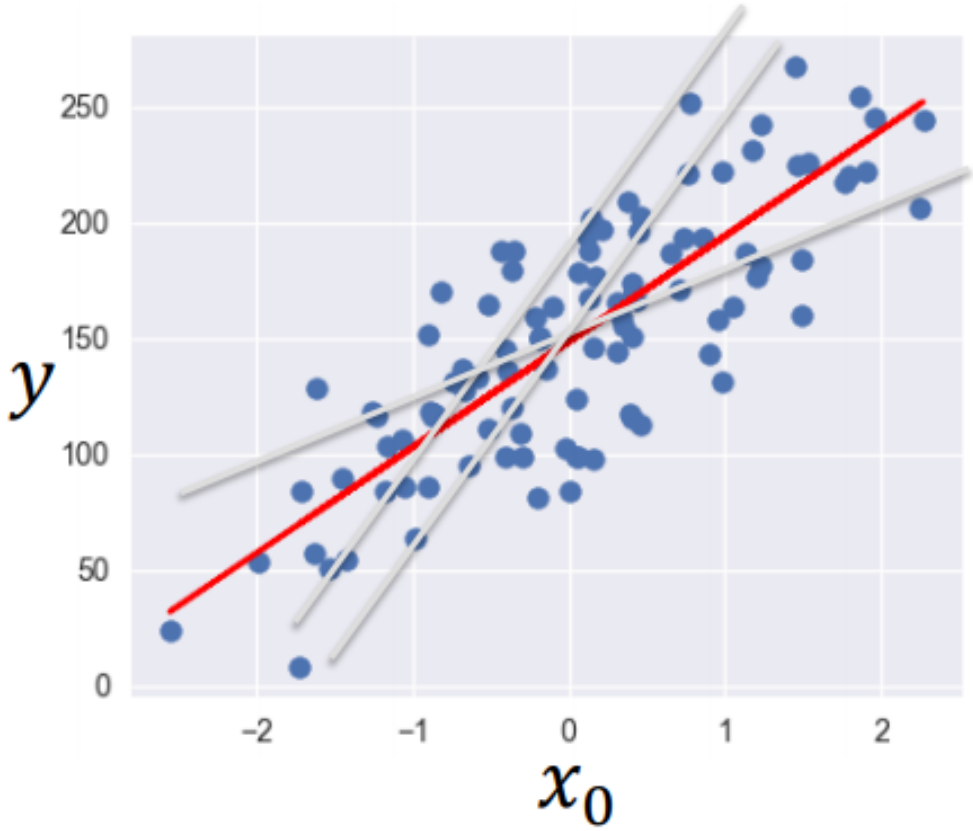
Linear regression

- Example : linear regression model with one variable

Input instance: $\boldsymbol{x} = (x_0)$

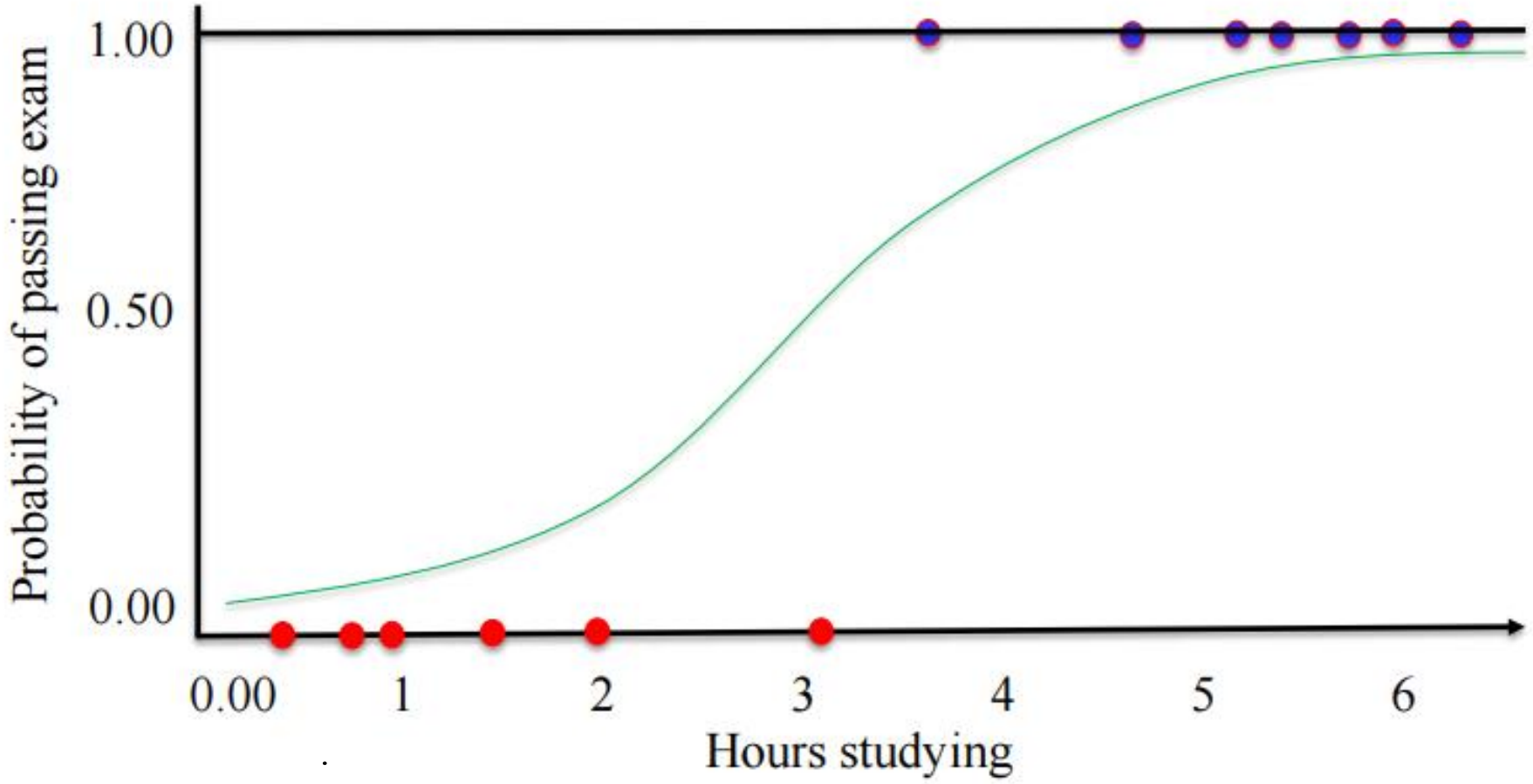
Predicted output: $\hat{y} = \widehat{w_0} x_0 + \hat{b}$

Parameters to estimate: $\widehat{w_0}$ (slope), $\hat{b}$ (y-intercept)



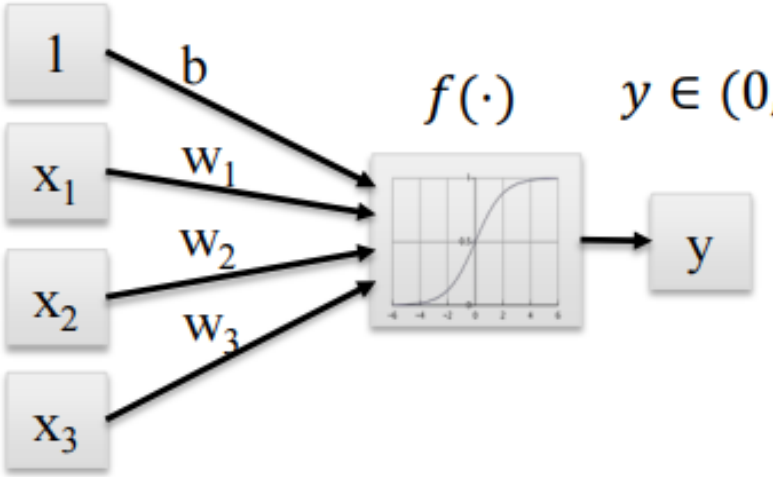
- Objective: minimize $RSS(\boldsymbol{w}, b) = \sum_{\{i=1\}}^{N} (\boldsymbol{y}_i - (\boldsymbol{w} \cdot \boldsymbol{x}_i + b))^2$

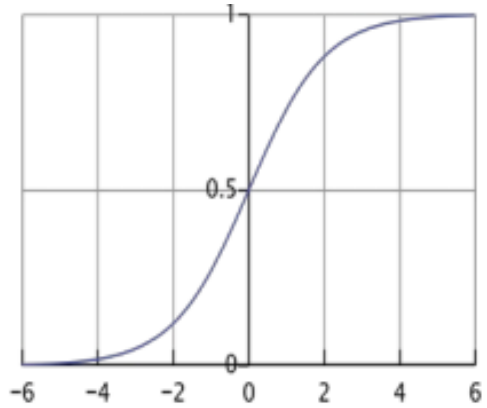
Linear Regression to Logistic Regression



Linear models for classification:

Logistic Regression





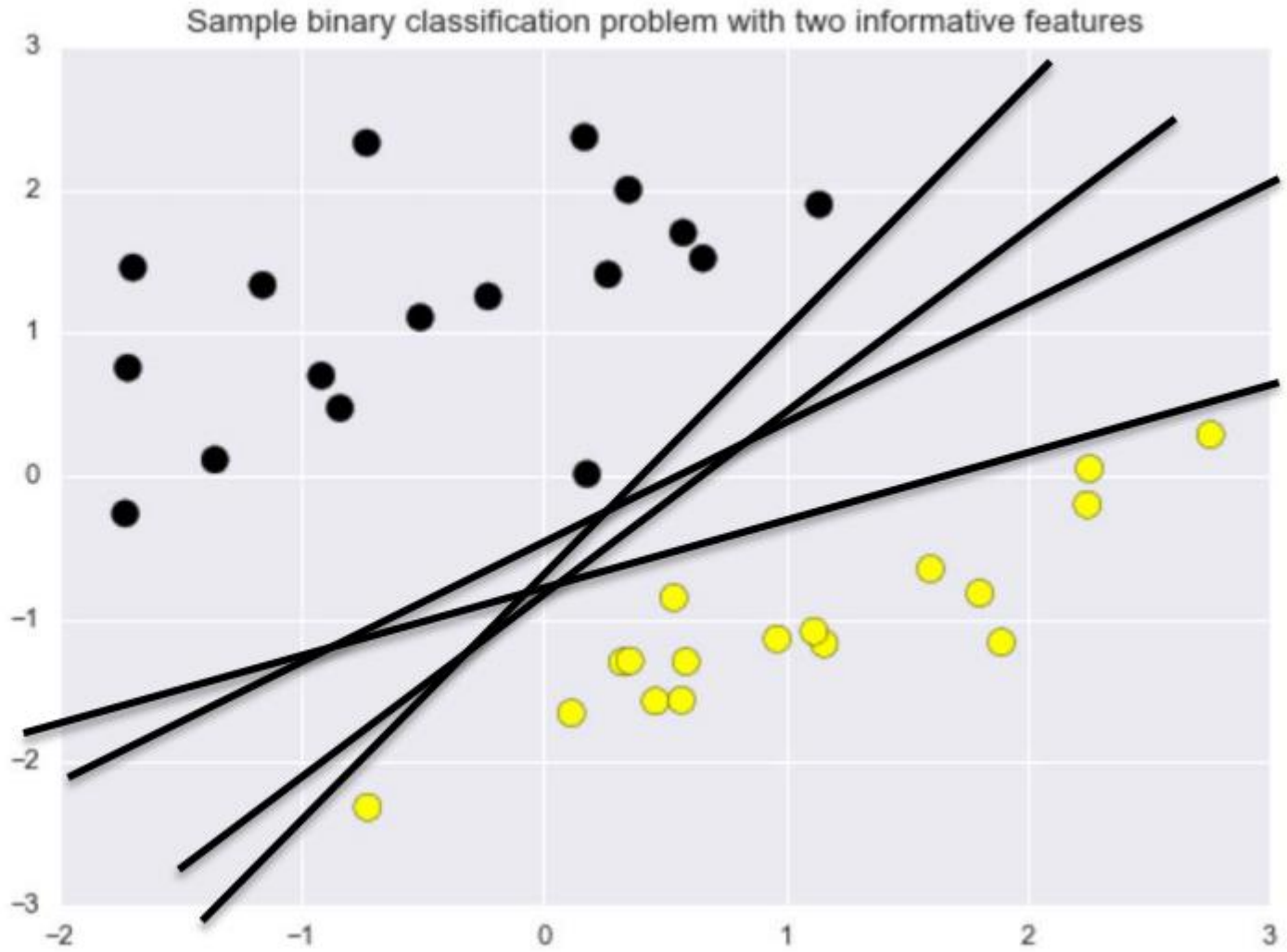
$$\hat{y} = \text{logistic}(\hat{b} + \hat{w}_1 \cdot x_1 + \cdots \hat{w}_n \cdot x_n)$$

$$= \frac{1}{1 + \exp\left[-(\hat{b} + \hat{w}_1 \cdot x_1 + \cdots \hat{w}_n \cdot x_n)\right]}$$

Linear models for classification: Linear Classifiers

$$f(x, w, b) = sign(w \circ x + b)$$

There are many possible linear classifiers that could separate the two classes.
Which one is best?

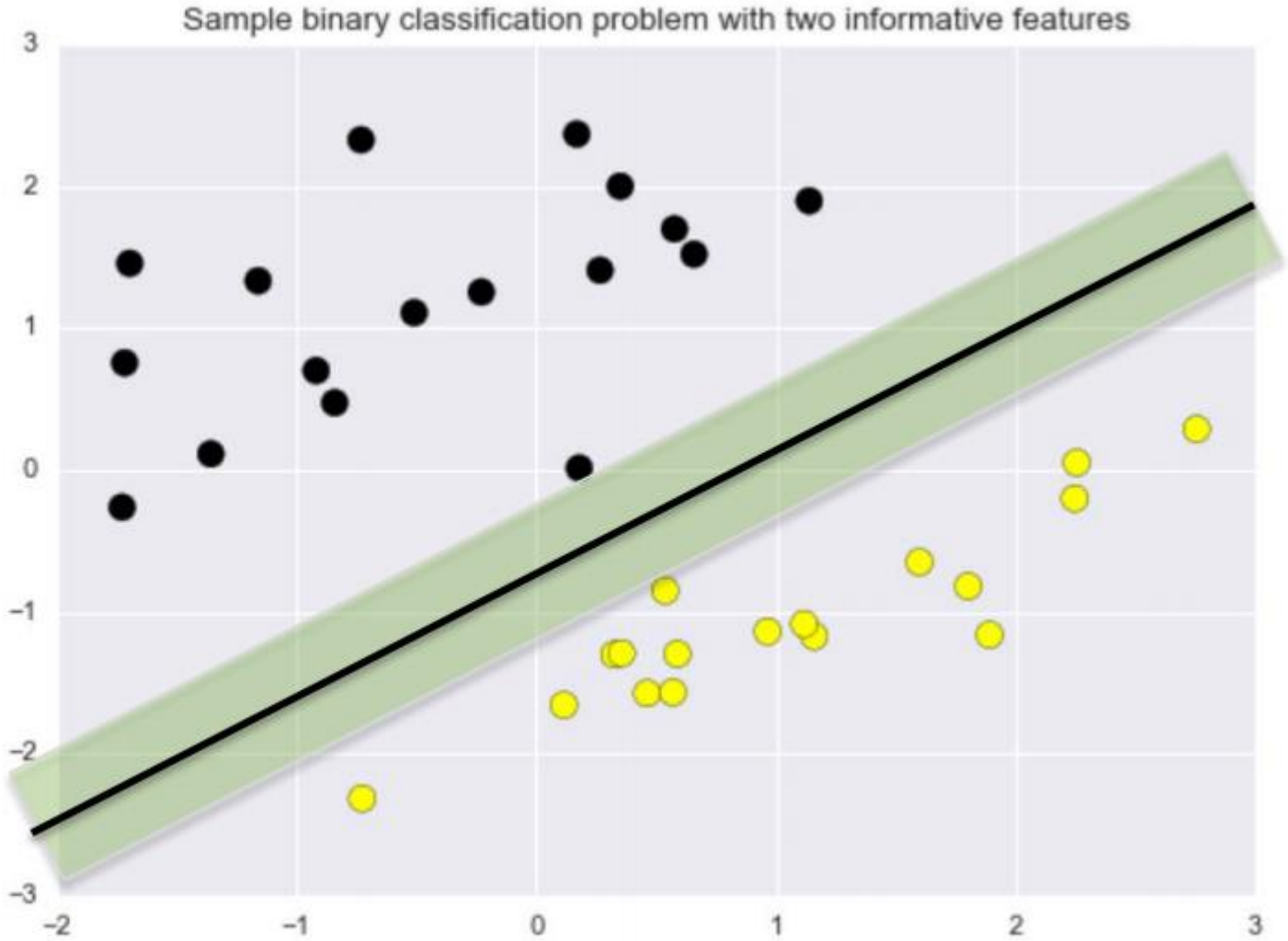


Linear models for classification: Support Vector Machine

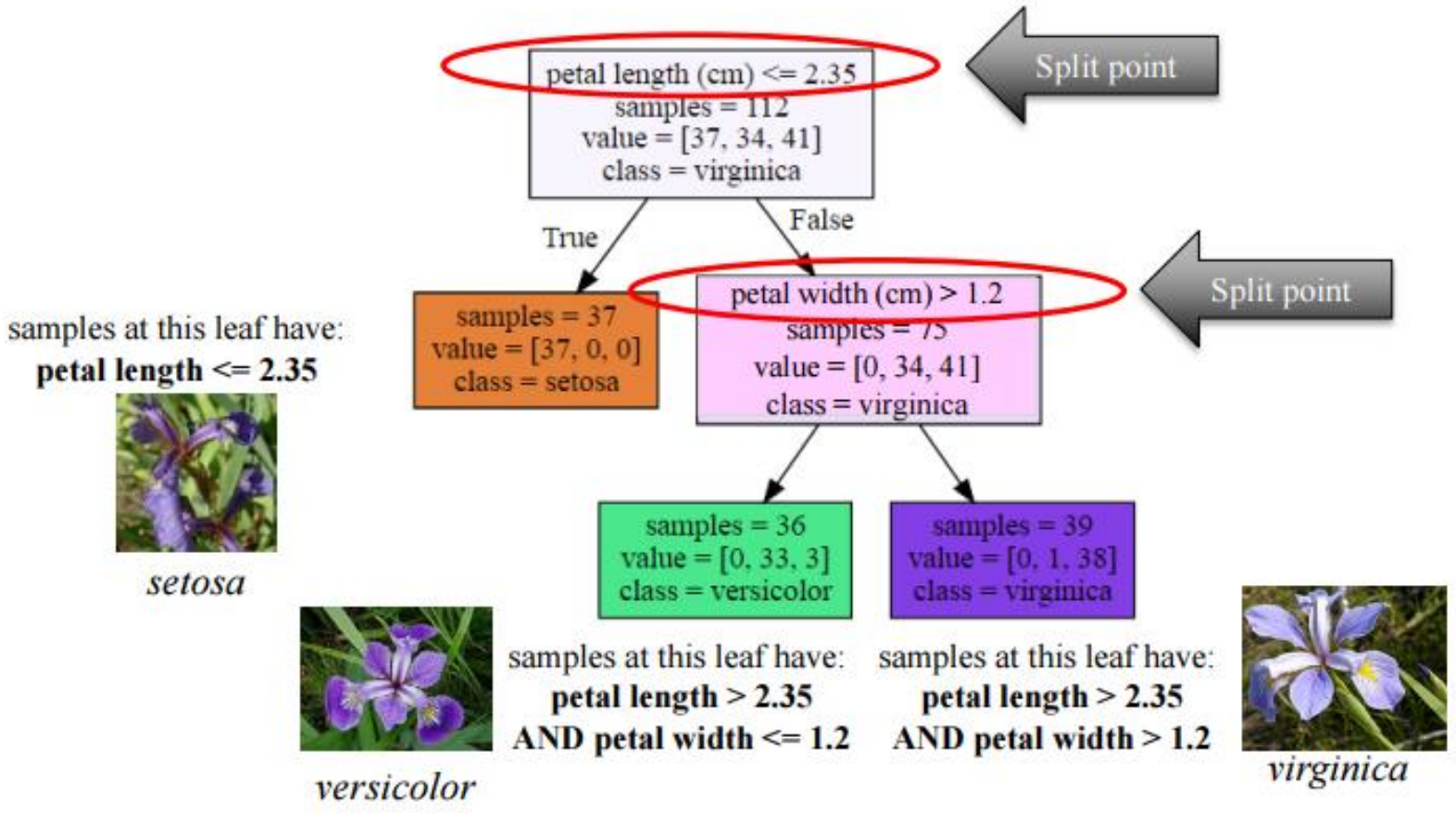
$$f(x, w, b) = sign(w \circ x + b)$$

Maximum margin classifier

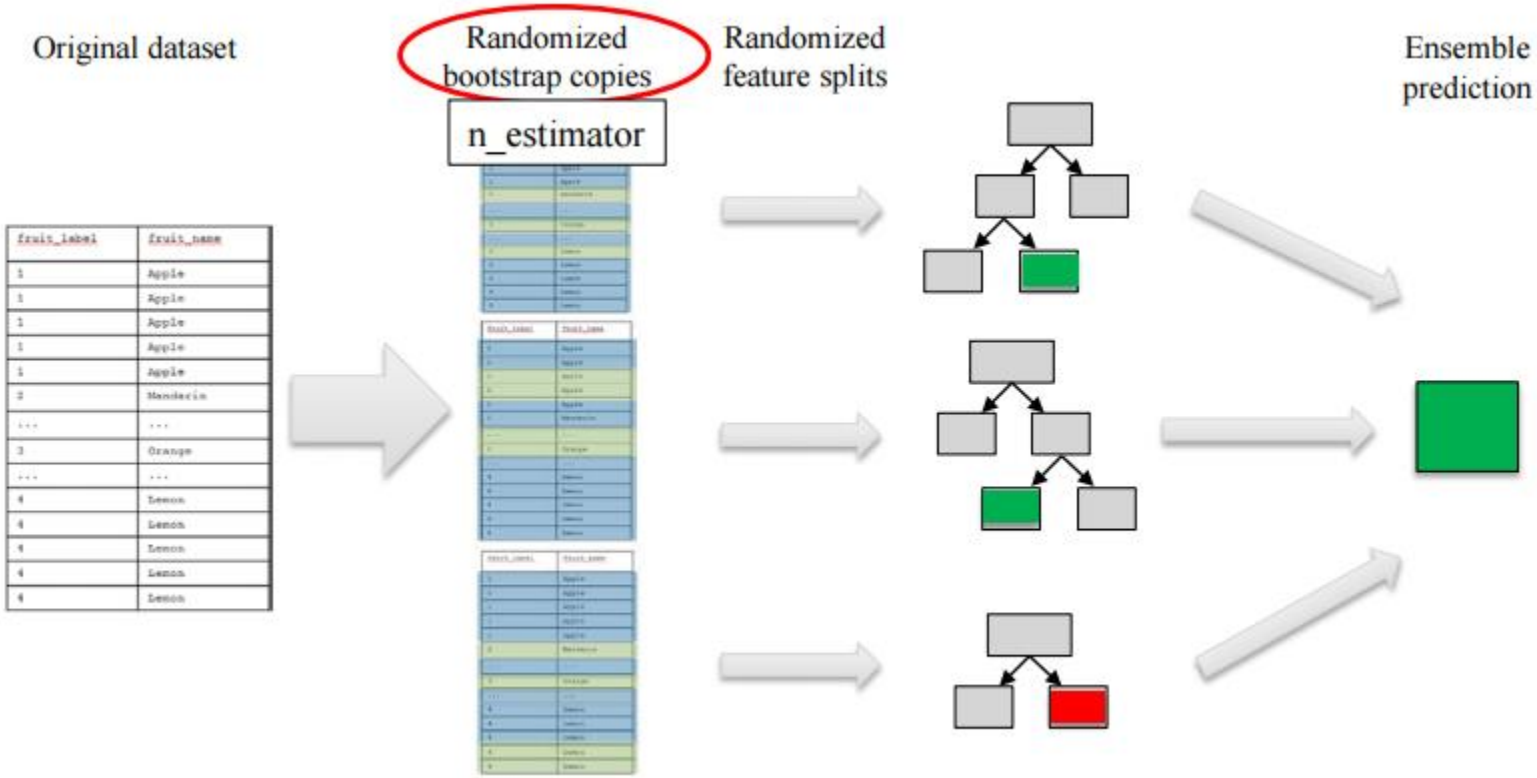
The linear classifier with maximum margin is a linear Support Vector Machine (LSVM).



Decision Tree



Random Forest Split



Multi-class classification with linear models

```
clf = LinearSVC(C=5, random_state = 67)
clf.fit(X_train, y_train)

print(clf.coef_)

[[-0.23401135  0.72246132]
 [-1.63231901  1.15222281]
 [ 0.0849835   0.31186707]
 [ 1.26189663 -1.68097   ]]

print(clf.intercept_)
[-3.31753728  1.19645936 -2.7468353  1.16107418]
```