

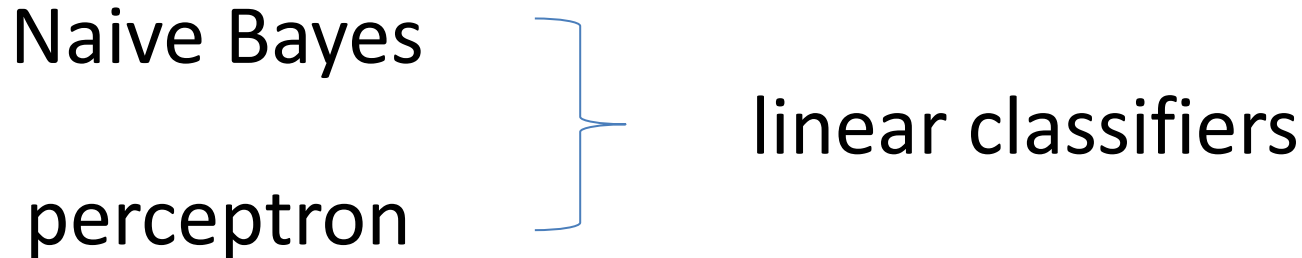
# Semi-Supervised Learning and Domain Adaptation in Natural Language Processing

## 2.4 PERCEPTRON

### 2.4.1 LARGE-MARGIN METHODS

吉田拓夢

# Linear classifiers



- algorithm ← very different
  - Naive Bayes
    - computes  $\mathbf{w}$  once and for all
  - perceptron
    - updates the weights in  $\mathbf{w}$  at a fixed rate in multiple passes over the data

# Perceptron:c

$$c(\mathbf{x}_j) = \begin{cases} 1 & \mathbf{w} \cdot \mathbf{x} + b > 0 \\ 0 & \text{other} \end{cases}$$

$\mathbf{w}$ :weight vector

$\alpha$ :learning rate

$b$ :bias term

In other words.....,



$\mathbf{x}_j \leftarrow \text{positive}$

$$\mathbf{x}_j \cdot \mathbf{w} > -b$$

- If the prediction is wrong,an update occurs:

$$\mathbf{w}^{i+1} \leftarrow \mathbf{w}^i + \alpha(y_j - \text{sign}(\mathbf{w}^i \cdot \mathbf{x}_j))\mathbf{x}_j$$

↑i is presented with a labeled data point  $\langle \mathbf{x}_j, y_j \rangle$

# Assessment of algorithm

Good

- computationally efficient
- conceptually simple
- very popular

Bad

- it does not prefer scoring functions with larger margins
- the final scoring function depends heavily on the data sample  
→ prone to overfitting data

# Averaged Perceptron

- averaging the weight vectors  $w^1, w^2, \dots$ 
  - one way overfitting can be reduced
- two options
  - to store all weight vectors (one for each update) and average them after learning
  - to average after every single update (early update)

# Passive-aggressive

Averaged Perceptron

Passive-aggressive



passive learning  
algorithms

do not update their weight vectors on instances that are classified correctly with the current model.

However....

the passive-aggressive is also aggressive in the sense that it forces  $w^{i+1}$  to classify the latest instance correctly, regardless of the step-size required.