ECE 295: Lecture 05 Supervised Learning

Spring 2018

Prof Stanley Chan

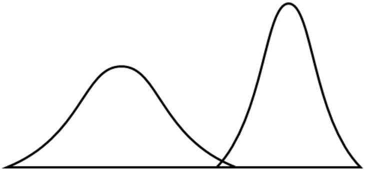
School of Electrical and Computer Engineering
Purdue University

Motivation

Escalator Problem:

- You study the escalator problem for two airports
- Repeat the measurements for N days
- You have two distributions of the sample means
- I give you a new data point
- Which class does it belong to?

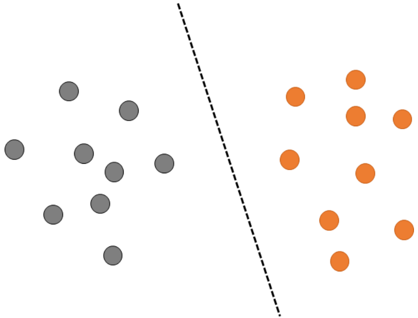
	Indianapolis	Chicago O'Hare
Day 1	$\overline{X}_1 = 10$	$\overline{Y}_1 = 100$
Day 2	$\overline{X}_2 = 11$	$\overline{Y}_2 = 98$
Day 3	$\overline{X}_3 = 10$	$\overline{Y}_3 = 99$
$\vdots$	$\vdots$	$\vdots$
Day N	$\overline{X}_N = 12$	$\overline{Y}_N = 103$



Supervised Learning

Why call **supervised** learning?

- Labeled ground truth available
- Build a classifier based on the training data
- Given a new data point, tell which class does it belong to
- Can be high-dimensional data points



Supervised Learning Methods

We will talk about two methods

Naive Bayes

- Requires a model, e.g., Gaussian.
- Do classification by estimating the likelihood.
- High training cost (depending on choice model).
- Low testing cost. Likelihood is usually not expensive.

K Nearest Neighbor

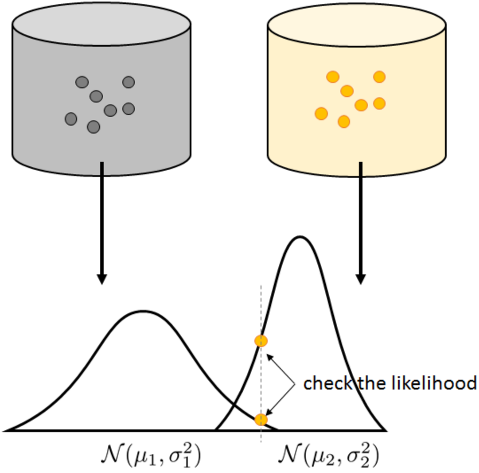
- Does not require a model.
- Do classification by measuring distance.
- No training cost.
- High testing cost. You need to measure distance for every testing data point.

There are other methods:

- Support Vector Machine
- Neural Networks

Naive Bayes

- Pick a model. Let's say Gaussian.
- From the data, estimate the parameters. For Gaussian, estimate μ and σ



Naive Bayes

Recall **Bayes Theorem**: The **posterior** distribution is

$$f_{C \mid \boldsymbol{X}}(c \mid \boldsymbol{x}) = \frac{f_{\boldsymbol{X} \mid C}(\boldsymbol{x} \mid c) f_C(c)}{f_{\boldsymbol{X}}(\boldsymbol{x})}. \tag{1}$$

- $f_{\boldsymbol{X} \mid C}(\boldsymbol{x} \mid C)$: The likelihood of having $\boldsymbol{X} = \boldsymbol{x}$ given class $C = c$.
- $f_C(c)$: The probability of getting class $C = c$.

The Naive Bayes is also called the Maximum-a-Posteriori (MAP) decision. In a two-class classification problem, MAP states that

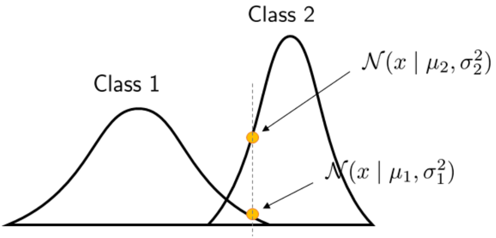
$$f_{C \mid \boldsymbol{X}}(1 \mid \boldsymbol{x}) \quad \gtrless^{\text{class } 1}_{\text{class } 0} \quad f_{C \mid \boldsymbol{X}}(0 \mid \boldsymbol{x}). \tag{2}$$

Example: Single-variable Gaussian

Recall Gaussian:

$$f_{X\,|\,C}(x\,|\,0) = \frac{1}{\sqrt{2\pi\sigma_0^2}} \exp\left\{-\frac{(x-\mu_0)^2}{2\sigma_0^2}\right\} \overset{\text{def}}{=} \mathcal{N}(x\,|\,\mu_0, \sigma_0^2)$$

$$f_{X\,|\,C}(x\,|\,1) = \frac{1}{\sqrt{2\pi\sigma_1^2}} \exp\left\{-\frac{(x-\mu_1)^2}{2\sigma_1^2}\right\} \overset{\text{def}}{=} \mathcal{N}(x\,|\,\mu_1, \sigma_1^2)$$



Example: Single-variable Gaussian

- Given a testing data point x
- Compute likelihoods $\mathcal{N}(x \mid \mu_1, \sigma_1^2)$ and $\mathcal{N}(x \mid \mu_0, \sigma_0^2)$
- The MAP decision rule is

$$\frac{f_{X \mid C}(x \mid 1) f_C(1)}{f_X(x)} \gtrless_{\text{class } 0}^{\text{class } 1} \frac{f_{X \mid C}(x \mid 0) f_C(0)}{f_X(x)}$$

We can cancel out the denominators:

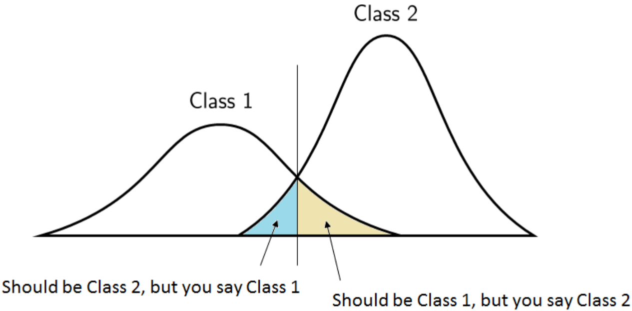
$$f_{X \mid C}(x \mid 1) f_C(1) \gtrless_{\text{class } 0}^{\text{class } 1} f_{X \mid C}(x \mid 0) f_C(0)$$

Write out the terms explicitly:

$$\mathcal{N}(x \mid \mu_1, \sigma_1^2) \mathbb{P}[\text{Class } 1] \gtrless_{\text{class } 0}^{\text{class } 1} \mathcal{N}(x \mid \mu_0, \sigma_0^2) \mathbb{P}[\text{Class } 0]$$

Naive Bayes

- $\mathbb{P}$[Class 1] is the probability that Class 1 shows up
- $\mathbb{P}$[Class 1] usually requires some prior knowledge
- Naive Bayes tells you "soft-decisions"
- They are the probabilities that x should belong to Class 1 or 2.
- Cut off appears when the two Gaussian intersects
- There are two types of error



Multi-Dimensional Gaussian

What if the data is high-dimensional?

Definition (High-dimensional Gaussian)

A d-dimensional **Gaussian** has a PDF

$$\mathcal{N}(\boldsymbol{x} \mid \boldsymbol{\mu}, \boldsymbol{\Sigma}) = \frac{1}{\sqrt{(2\pi)^d |\boldsymbol{\Sigma}|}} \exp\left\{ -\frac{1}{2}(\boldsymbol{x} - \boldsymbol{\mu})^T \boldsymbol{\Sigma}^{-1} (\boldsymbol{x} - \boldsymbol{\mu}) \right\}.$$

- $\boldsymbol{\mu} \in \mathbb{R}^d$ is the mean vector
- $\boldsymbol{\Sigma} \in \mathbb{R}^{d \times d}$ is the covariance matrix

The mean vector and the covariance matrix can be computed using commands

`mean, cov`

Caution: Be careful about the transpose of the data matrix.

Classification

- Given a testing dataset $\boldsymbol{y}_1, \ldots, \boldsymbol{y}_N$.
- Assume $\mathbb{P}[\text{Class 1}] = \mathbb{P}[\text{Class 0}] = \frac{1}{2}$.

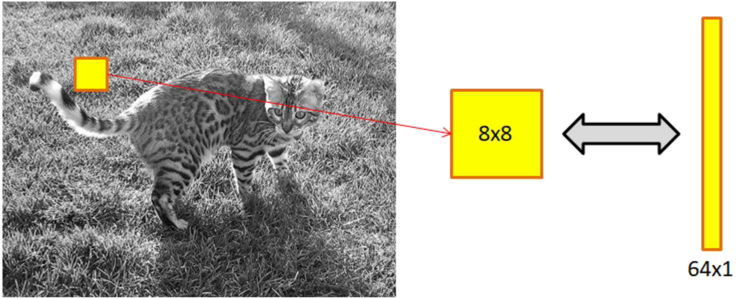
The Naive Bayes (i.e., the MAP) decision is

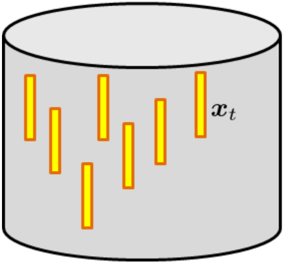
$$\mathcal{N}(\boldsymbol{y}_i \mid \boldsymbol{\mu}_1, \boldsymbol{\Sigma}_1) \gtrless_{\text{class 0}}^{\text{class 1}} \mathcal{N}(\boldsymbol{y}_i \mid \boldsymbol{\mu}_0, \boldsymbol{\Sigma}_0),$$

This is equivalent to

$$\frac{1}{\sqrt{(2\pi)^d|\boldsymbol{\Sigma}_1|}} \exp\left\{-\frac{1}{2}(\boldsymbol{x}-\boldsymbol{\mu}_1)^T\boldsymbol{\Sigma}_1^{-1}(\boldsymbol{x}-\boldsymbol{\mu}_1)\right\}$$
$$\gtrless_{\text{class 0}}^{\text{class 1}} \frac{1}{\sqrt{(2\pi)^d|\boldsymbol{\Sigma}_0|}} \exp\left\{-\frac{1}{2}(\boldsymbol{x}-\boldsymbol{\mu}_0)^T\boldsymbol{\Sigma}_0^{-1}(\boldsymbol{x}-\boldsymbol{\mu}_0)\right\} \quad (3)$$

Homework 5





$$\boldsymbol{\mu} = \frac{1}{T}\sum_{t=1}^{T} \boldsymbol{x}_t \qquad \in \mathbb{R}^{64\times 1}$$

$$\boldsymbol{\Sigma} = \frac{1}{T}\sum_{t=1}^{T} (\boldsymbol{x}_t - \boldsymbol{\mu})(\boldsymbol{x}_t - \boldsymbol{\mu})^T \qquad \in \mathbb{R}^{64\times 64}$$

Naive Bayes

Pros:

- You have a model
- More interpretable
- Usually cheap to compute the likelihood
- Robust against outliers
- Good for missing data

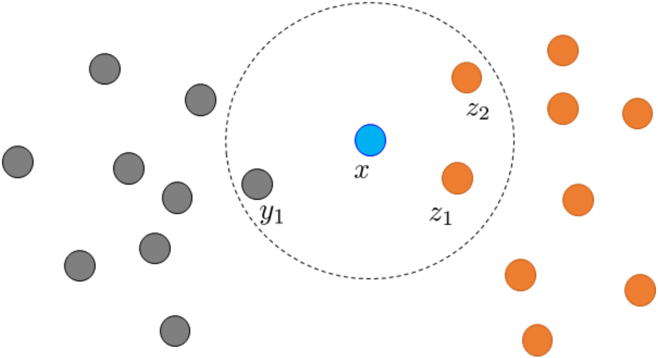
Cons:

- You need to choose a model
- Your model may not work — It may not describe the data accurately
- Decision boundary could be over-simplified
- You need to have prior knowledge

k-NN

k-Nearest Neighbor

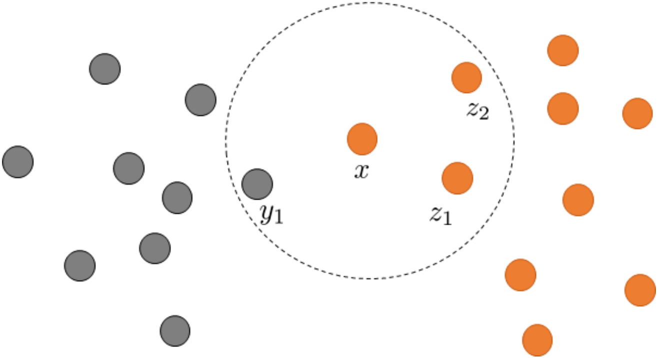
- Start with two labeled clusters
- Give me a new data point x
- Draw a circle around x



k-NN

k-Nearest Neighbor

- Grow the circle until you find k data points, e.g., $k = 3$
- Count how many are in Class 1 and Class 2
- If Class 1 is more, then assign x to Class 1



k-NN

Distance

- For 1D data points, we can set

$$D(x, y) = (x - y)^2, \quad \text{or} \quad D(x, y) = |x - y|$$

- For high dimensional data points, we can set

$$D(\boldsymbol{x}, \boldsymbol{y}) = \sum_{i=1}^{d} (x_i - y_i)^2, \quad \text{or} \quad D(\boldsymbol{x}, \boldsymbol{y}) = \sum_{i=1}^{d} |x_i - y_i|$$

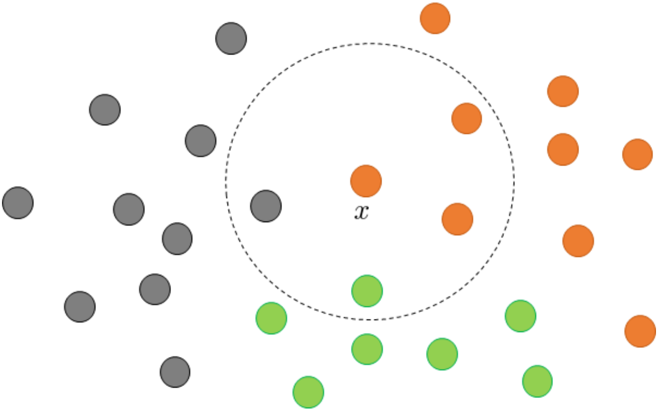
How large K should be?

- Depends on your problem
- Small datasets, k should be small

k-NN

Multiple Classes

- Say there are M classes
- Then k cannot be a multiple of M
- Otherwise there will be tie



kNN

Pros:

- No need to have a model
- Can have arbitrary decision boundary
- Could be efficient if dataset is small

Cons:

- Very expensive if dataset if large
- Not as interpretable as Naive Bayes
- Need to tune k
- Doesn't handle missing data

Summary

- Supervised learning: Ground truth available
- Two methods in this lecture
- Naive Bayes: Require a model, but fast and interpretable
- kNN: Does not require a model, but slow
- There are many other supervised learning methods