STA 314: Statistical Methods for Machine Learning I

Lecture 6 - Introdution to classification: the Bayes rule

Xin Bing

Department of Statistical Sciences
University of Toronto

Logistics

- Midterm on Wednesday, Sep 25th.
- No classes but only tutorials on the next Monday, Sep 30th.
- Course project
  - Group sign-up: self sign up on Quercus (due Nov 8th, 11:59pm)
  - Project document: available on Quercus from Sep 26th, 11:59pm.
  - Kaggle competition due: Dec 6th, 11:59pm.
  - Final report due: Dec 8th, 11:59pm.

  - No late submission allowed!

Introduction to classification problems

The response variable Y is qualitative, taking values in an unordered set C. Depending on the cardinality of C,

- binary classification: $|C| = 2$
  - email is $C = \{\text{spam}, \text{non-spam}\}$
  - the status of patient is C ={cancer, non-cancer}
- Multi-class classification: $|C| > 2$
  - digit is $C = \{0, 1, ..., 9\}$
  - eye color is C ={brown, blue, green}.

Classification

Given the training data: $\mathcal{D}^{train} = \{(x_1, y_1), \ldots, (x_n, y_n)\}$, with $y_i \in C$ and $x_i \in \mathbb{R}^p$, our goals are to:

- Build a classifier (a.k.a. a rule)

$$\hat{f} : \mathbb{R}^p \rightarrow C$$

  that assigns a future observation $x \in \mathbb{R}^p$ to a class label $\hat{f}(x) \in C$.

- Assess the accuracy of this classifier $\hat{f}$ (classification accuracy).
- Understand the roles of different features in $\hat{f}$ (estimation and interpretability).

The metric used in classification

Let (X, Y) be a random pair, independent of $\mathcal{D}^{train}$. Let us encode the labels as

$$C = \{0, 1, 2, \ldots, K-1\}.$$

For any classifier $\hat{f}$, we evaluate it based on the **expected error rate**

$$\mathbb{E}\left[1\{Y \neq \hat{f}(X)\}\right].$$

Question: what is the best classifier?

Draw analogy in the regression context

In regression context

$$Y = f^*(X) + \epsilon,$$

the regression function is the best predictor: for any $x \in \mathbb{R}^p$,

$$\begin{aligned} f^*(x) &= \mathbb{E}[Y \mid X = x] \\ &= \underset{\hat{f}(x)}{\operatorname{argmin}} \ \mathbb{E}\left[(Y - \hat{f}(X))^2 \mid X = x\right] \end{aligned}$$

Its MSE is the smallest (a.k.a. irreducible error)

$$\mathbb{E}\left[(Y - f^*(X))^2\right] = \mathrm{Var}(\epsilon) = \sigma^2.$$

The Bayes rule and the Bayes error

The Bayes classifier (rule) is a function: $f^*: \mathbb{R}^p \to C$, that minimizes the expected error rate as

$$f^*(x) = \underset{\hat{f}(x) \in C}{\operatorname{argmin}} \ \mathbb{E}\left[1\{Y \neq \hat{f}(X)\} \mid X = x\right], \qquad \forall x \in \mathbb{R}^p.$$

Correspondingly, its expected error rate

$$\mathbb{E}\left[1\{Y \neq f^*(X)\}\right]$$

is called the **Bayes error rate** which is the smallest.

The Bayes rule

For any $x \in \mathbb{R}^p$,

$$f^*(x) = \operatorname*{argmin}_{\hat{f}(x) \in C} \mathbb{E}\left[1\{Y \neq \hat{f}(X)\} \mid X = x\right]$$
$$= \operatorname*{argmin}_{\hat{f}(x) \in C} \mathbb{P}\left\{Y \neq \hat{f}(x) \mid X = x\right\}.$$

Intuitively, $f^*(x)$ assigns each x to its most probable class, that is,

$$f^*(x) = \arg\max_{k \in C} \mathbb{P}\{Y = k \mid X = x\}.$$

The Bayes classifier, f^*, is our target to estimate / learn in classification problems.

The Bayes Error Rate

The Bayes error rate at $X = x$ is

$$\begin{aligned}\mathbb{E}\left[1\{Y \neq f^*(X)\} \mid X = x\right] &= \mathbb{P}\left\{Y \neq f^*(X) \mid X = x\right\} \\ &= 1 - \mathbb{P}\left\{Y = f^*(X) \mid X = x\right\} \\ &= 1 - \max_{1 \leq j \leq K} \mathbb{P}\left\{Y = j \mid X = x\right\}.\end{aligned}$$

The Bayes error rate is:

- between 0 and 1.
- typically $\neq 0$.

Binary classification

In binary classification, $C = \{0, 1\}$ and the Bayes classifier is

$$f^*(x) = \begin{cases} 1, & \text{if } \mathbb{P}\{Y = 1 \mid X = x\} \geq 0.5; \\ 0, & \text{otherwise.} \end{cases}$$

Learning the Bayes classifier equals to estimating **the conditional probability**

$$p(x) \ := \ \mathbb{P}\{Y = 1 \mid X = x\}, \quad \forall x \in \mathbb{R}^p,$$

a function: $\mathbb{R}^p \to \{0, 1\}$.

Why Not Regression?

- In the binary case, $Y \in \{0,1\}$,

$$p(X) = \mathbb{P}\{Y = 1 \mid X\} = \mathbb{E}[Y \mid X].$$

  Recall the regression setting,

$$Y = f(X) + \epsilon = \mathbb{E}[Y \mid X] + \epsilon.$$

- Can we use the regression approach (such as OLS) to estimate $\mathbb{E}[Y \mid X]$?

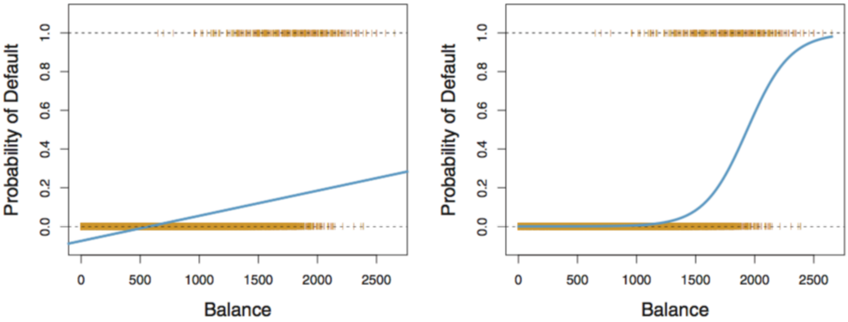
Using OLS to predict $p(X) = \mathbb{P}(Y = 1 \mid X)$

- Yes, we could (as commonly done in practice).
- However, OLS predict $p(X)$ by

$$\hat{\beta}_0 + \hat{\beta}_1 X_1 + \cdots + \hat{\beta}_p X_p,$$

  which could be less than zero or bigger than one.
- A more tailored approach is needed!

Linear Regression versus Logistic Regression in binary classification



- Left: Estimated probability of default using linear regression. Some estimated probabilities are negative! The orange points represents the 0/1 values coded for default (No or Yes).
- Right: Predicted probabilities of default using logistic regression. All probabilities lie between 0 and 1.

Classification approaches

How to estimate

$$p(x) = \mathbb{P}\{Y = 1 \mid X = x\}$$

or, more generally,

$$\mathbb{P}\{Y = j \mid X = x\}, \qquad \forall j \in C,$$

for any $x \in \mathbb{R}^p$?

- Parametric methods
  - Logistic regression
  - Discriminant analysis
- Non-parametric methods
  - Support vector machine
  - k-nn
  - Classification tree

How to select among a set of classifiers?

For a given classifier $\hat{f} : \mathbb{R}^p \to C$, we have

- **Training 0-1 error rate.**

$$\frac{1}{n}\sum_{i=1}^{n} 1\{y_i \neq \hat{f}(x_i)\}$$

- **Test 0-1 error rate** when we have the test data $\{(x_{T_1}, y_{T_1}), \ldots, (x_{T_m}, y_{T_m})\}$,

$$\frac{1}{m}\sum_{i=1}^{m} 1\left\{y_{T_i} \neq \hat{f}(x_{T_i})\right\}.$$

How to select among a set of classifiers?

- **Data-splitting based on 0-1 error rate** when we don't have the test data.
    - Validation-set approach
    - Cross-validation

- More metrics on binary classification.