Section 8: Classification with R

(UC Davis, Spring 2026) — STA 141A: Fundamentals of Statistical Data Science

Instructor: Akira Horiguchi

Overview

Based on Chapter 4 of ISL book James et al. (2021). For more R code examples, see R Markdown files in book website.

Why not linear regression?

Binary classification

- Logistic regression
- Errors in binary classification

Classification with more than two classes

- Multinomial logistic regression
- Alternatives to logistic regression
- Linear discriminant analysis for $p = 1$
- Naive Bayes
- Comparison of classification methods

Example (two categories)

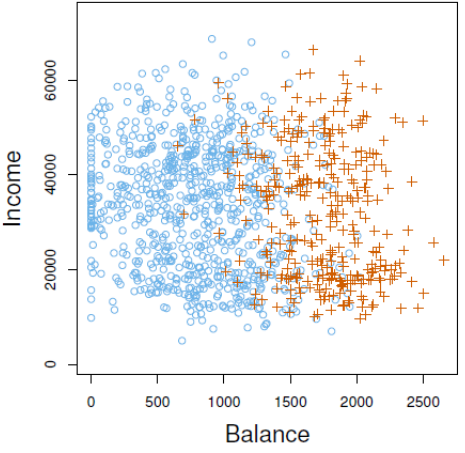


Figure: Image by James et al. (2021), based on the `Default` data set in `R`. The annual incomes and monthly credit card balances of a number of individuals. The individuals who defaulted on their credit card payments are shown as orange +s, and those who did not are shown as blue ∘s.

What are the predictors and responses in each example?

1. A person arrives at the emergency room with a set of symptoms that could possibly be attributed to one of three medical conditions. Which of these medical conditions does the person have based on the symptoms given?

2. An online banking service must be able to determine whether or not a transaction being performed on the site is fraudulent, on the basis of the user's IP address, past transaction history, and so forth.

3. Using DNA sequence data for a number of patients with and without a given disease, one would like to figure out which DNA mutations are disease-causing and which are not.

The concept

Classification: the task of predicting *qualitative*/*categorical* responses

- Each response y_i is one of finitely many predetermined categories.
- *Classifying* an observation: assigning/predicting that observation to a certain category/class.
- In contrast, regression deals with "continuous" numeric response values.

As in regression, in the classification setting

- We have a set of training observations $(x_1, y_1), \ldots, (x_n, y_n)$ that we can use to build a classifier.
- We want our classifier to perform well not only on the training data, but also on test observations that were not used to train the classifier.

Why not linear regression?

Binary classification

- Logistic regression
- Errors in binary classification

Classification with more than two classes

- Multinomial logistic regression
- Alternatives to logistic regression
- Linear discriminant analysis for $p = 1$
- Naive Bayes
- Comparison of classification methods

No natural ordering

In example 1 above, a person arrives at the emergency room with a set of symptoms. We'd like to treat the person based on one of three reasonable medical conditions: `Appendicitis`, `Food poisoning`, `Gastritis`.

- We could code each medical condition Y as:

$$Y = \begin{cases} 1, & \text{if } \texttt{Appendicitis} \text{ ,} \\ 2, & \text{if } \texttt{Food poisoning}, \\ 3, & \text{if } \texttt{Gastritis}. \end{cases}$$

  This coding implies an ordering on the outcomes, insisting that the difference between `Appendicitis` and `Food poisoning` is the same as the difference between `Food poisoning` and `Gastritis`.

- We could also code:

$$Y = \begin{cases} 1, & \text{if } \texttt{Gastritis}, \\ 2, & \text{if } \texttt{Appendicitis}, \\ 3, & \text{if } \texttt{Food poisoning}. \end{cases}$$

  Equally reasonable, but would lead to very different predictions on test observations.

Natural ordering

What if categories had a natural ordering, such as `mild`, `moderate`, and `severe`?

- Issue: the distance between *ordinal* categories is generally unknown.
- In general there is no natural way to convert a qualitative response variable with *more than two levels* into a quantitative response that is ready for linear regression.

Only two levels: linear regression

Can we use linear regression for a *binary* (two levels) response?

- In the Default data set, the two response values can be coded as

$$Y = \begin{cases} 1, & \text{if } \texttt{Default}, \\ 0, & \text{if } \texttt{Not default}. \end{cases}$$

- We could then fit a linear regression to this binary response:

$$\hat{Y} = \hat{\beta}_0 + \hat{\beta}_1 \times \texttt{Balance} + \hat{\beta}_2 \times \texttt{Income}$$

  and then predict `Default` if $\hat{Y} > 0.5$ and `Not default` otherwise.

- What if we also want to estimate e.g.,

$$P(\texttt{Default} | \texttt{Balance} = 4000, \texttt{Income} = 80000)$$

  i.e., the probability of defaulting given certain values of `Balance` and `Income`?

- Issue: $\hat{Y}$ can be smaller than zero or larger than one.

Only two levels: linear regression vs logistic regression

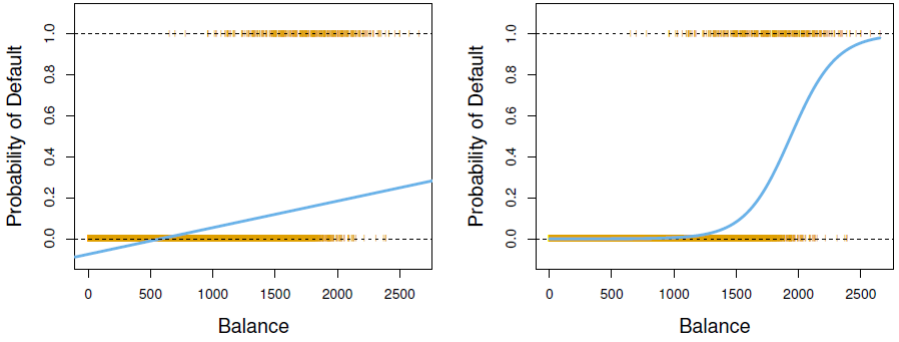


Figure: Image by James et al. (2021), based on the `Default` data set in `R`. Left: The estimated probability of default using *linear regression*, where the orange ticks indicate the values "0" for `No`, and "1" for `Yes`. Right: Predicted probabilities of default using *logistic regression*, where all predicted values lie between 0 and 1.

- Let's explore logistic regression, where interpreting coefficients is easier if we use *log odds*.

Log odds: another way to express probability

Let $P(A)$ be the *probability* that event A occurs. Then $P(A) \in [0, 1]$. Map to $(-\infty, \infty)$?

- For this course, assume log is the natural logarithm, i.e., log with base e.
- For any $t \in (0, 1)$, define the function

$$\text{logit}(t) \coloneqq \log\left(\frac{t}{1-t}\right), \tag{1}$$

  which maps $(0, 1)$ to $(-\infty, \infty)$.
- Because this function is continuous and strictly increasing, it has a well-defined inverse.
- The inverse transformation is the *logistic* function

$$\text{logistic}(s) \coloneqq \frac{1}{1+e^{-s}}, \tag{2}$$

  which maps $(-\infty, \infty)$ to $(0, 1)$.

Log odds: another way to express probability

Let $P(A)$ be the *probability* that event A occurs. Then $P(A) \in [0, 1]$. Map to $(-\infty, \infty)$?

- The *log odds* of A occurring is defined by plugging $P(A)$ into the logit function (1):

$$\text{logit}\Big(P(A)\Big) \;=\; \log\left(\frac{P(A)}{1-P(A)}\right). \tag{3}$$

- Log odds enables easier interpretion of the coefficients of a logistic regression model.
- Notice: we can always go back and forth between log odds and probability:

$$\text{logistic}\Big(\text{log odds of } A\Big) \;=\; P(A). \tag{4}$$

- StatQuest: Odds and Log(Odds), Clearly Explained!!! (11:30)

Binary classification

Each response belongs to one of two classes, coded as 0 and 1 (e.g., `No` and `Yes`).

- Classification: compute/estimate conditional prob. $P(Y = k|X)$ for each class k.
- If only two classes, we only need to compute/estimate $P(Y = 1|X)$. (Why?)

- For notational simplicity, define $p(X) := P(Y = 1|X)$. (Remember that X can be a p-variate quantity.)

Logistic regression

The *logistic regression* approach is to model the *log odds* using linear regression:

$$\text{logit}\Big(p(X)\Big) \;=\; \beta_0 + \beta_1 X_1 + \cdots + \beta_p X_p. \tag{5}$$

- Interpretation: increasing X_1 by one unit changes the log odds $\text{logit}\Big(p(X)\Big)$ by

$$\begin{aligned}&\text{logit}\Big(p(X_1+1, X_2, X_3, \ldots, X_p)\Big) \;-\; \text{logit}\Big(p(X_1, X_2, X_3, \ldots, X_p)\Big)\\ &= (\beta_0 + \beta_1(X_1+1) + \cdots + \beta_p X_p) \;-\; (\beta_0 + \beta_1 X_1 + \cdots + \beta_p X_p)\end{aligned}$$

  which is just β_1.

We could have instead modeled the conditional probability directly:

$$p(X) \;=\; \frac{e^{\beta_0+\beta_1 X_1+\cdots+\beta_p X_p}}{1+e^{\beta_0+\beta_1 X_1+\cdots+\beta_p X_p}} \;=\; \frac{1}{1+e^{-(\beta_0+\beta_1 X_1+\cdots+\beta_p X_p)}}, \tag{6}$$

- Interpretation: increasing X_1 by one unit changes $p(X)$ by

$$p(X_1+1, X_2, X_3, \ldots, X_p) \;-\; p(X_1, X_2, X_3, \ldots, X_p)$$

  which depends on all $p-1$ coefficient values and the current predictor values. (messy!)

Estimating the regression coefficients $\beta_0, \beta_1, \ldots, \beta_p$

Usually use the method of *maximum likelihood estimation*.

- Details outside scope of this class; we will just use R to compute these estimates.

To estimate log odds (5) or conditional probability (6), just plug in estimates $\hat{\beta}_0, \hat{\beta}_1, \ldots, \hat{\beta}_p$:

- We can estimate the log odds (5) at X by

$$\hat{\beta}_0 + \hat{\beta}_1 X_1 + \cdots + \hat{\beta}_p X_p. \tag{7}$$

- We can estimate the conditional probability (6) at X by

$$\hat{p}(X) \coloneqq \text{logistic}\left(\hat{\beta}_0 + \hat{\beta}_1 X_1 + \cdots + \hat{\beta}_p X_p\right) \tag{8}$$

Example of logistic regression

If $\hat{\beta}_0 = -9.9$ and $\hat{\beta}_1 = 0.005$

We estimate the log odds of `default` for individuals with balance $X = \$1{,}000$ and $X = \$2{,}000$

$$\begin{aligned} X = 1,000 &\implies \hat{\beta}_0 + \hat{\beta}_1 X = -9.9 + 0.005 \cdot 1000 = -4.9, \\ X = 2,000 &\implies \hat{\beta}_0 + \hat{\beta}_1 X = -9.9 + 0.005 \cdot 2000 = 0.1. \end{aligned}$$

We estimate the corresponding probabilities as

$$\begin{aligned} \hat{p}(X = 1,000) &= \text{logistic}\left(\hat{\beta}_0 + \hat{\beta}_1 X\right) = \text{logistic}(-9.9 + 0.005 \cdot 1000) \approx 0.007, \\ \hat{p}(X = 2,000) &= \text{logistic}\left(\hat{\beta}_0 + \hat{\beta}_1 X\right) = \text{logistic}(-9.9 + 0.005 \cdot 2000) \approx 0.525. \end{aligned}$$

Mental math tricks (deducible from (1) and (2)):

- Log odds is positive $\iff$ corresponding probability is larger than 0.5.
- Log odds is negative $\iff$ corresponding probability is smaller than 0.5.
- Log odds is zero $\iff$ corresponding probability is exactly 0.5.

Prediction

Suppose we have computed/estimated probability $P(Y = 1|X)$ for a given value of predictor X. What class (0 or 1) should be assigned to X?

- A *default decision rule* for predictor value X is to assign:

$$\begin{cases} 1 & \text{if } P(Y = 1|X) > 0.5; \\ 0 & \text{if } P(Y = 1|X) \leq 0.5. \end{cases} \tag{9}$$

  (What are the equivalent conditions using log odds?)

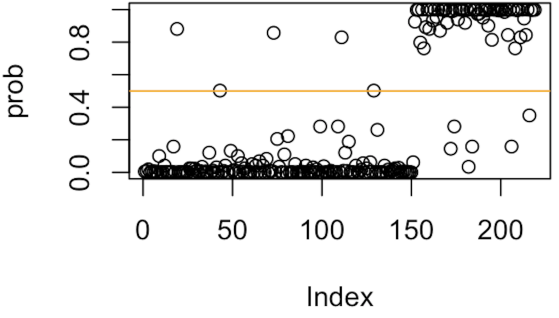
$$\begin{cases} P(Y = 1|X) > 0.5 & \iff \\ P(Y = 1|X) \leq 0.5 & \iff \end{cases}$$

- If we don't know $P(Y = 1|X)$, replace it with estimate (6).

We use `glm()` for logistic regression (*generalized linear model* (glm)).

- Must put `family=binomial` to specify a binary response.

```
# We work with only Adelie and Chinstrap species (we exclude Gentoo).
peng_binary <- na.omit(penguins[penguins$species != 'Gentoo', ])
logreg <- glm(species ~ bill_length_mm, data=peng_binary, family=binomial)
prob <- predict(logreg, type='response')  # 'link' also possible
predicted <- ifelse(prob<.5, 'Adelie', 'Chinstrap')
plot(prob)
abline(a=0.5, b=0, col='orange')
```



Another exposition on logistic regression

- StatQuest: Logistic Regression (8:47)
- StatQuest: Logistic Regression Details Pt1: Coefficients (19:01)

Errors in binary classification

In classification, observations can be assigned to the wrong class.

> It is possible to commit no mistakes and still lose. That is not a weakness; that is life.
> — Jean Luc Picard

> It is possible to use the data perfectly and still mispredict. That is not a weakness; that is life's randomness and uncertainty.
> — Me

Confusion matrix

Consider binary classification: e.g., not default vs default, cancer vs no cancer, spam vs not spam.

- Two mistakes are: *false positives* and *false negatives*.
- A *confusion matrix* displays both error types.

```
# Using peng_binary and predicted from earlier slide
pb_species <- factor(peng_binary$species, levels=c('Adelie', 'Chinstrap'))
table(pb_species, predicted)
```

```
> table(pb_species, predicted)
           predicted
pb_species  Adelie Chinstrap
  Adelie       141         5
  Chinstrap      6        62
```

```
# pb_species line is not necessary, but what happens if we instead did:
table(peng_binary$species, predicted)
```

Confusion matrix: more generally

		True class		
		− or Null	+ or Non-null	Total
Predicted class	− or Null	True Neg. (TN)	False Neg. (FN)	N^*
	+ or Non-null	False Pos. (FP)	True Pos. (TP)	P^*
	Total	N	P	

Figure: Table by James et al. (2021). Confusion matrix described using general terminology.

Name	Definition	Synonyms
False Pos. rate	FP/N	Type I error, 1−Specificity
True Pos. rate	TP/P	1−Type II error, power, sensitivity, recall
Pos. Pred. value	TP/P^*	Precision, 1−false discovery proportion
Neg. Pred. value	TN/N^*	

Figure: Table by James et al. (2021). Additional common terminology derivable from top table.

Decision rule threshold

Recall the earlier "default" decision rule (9) for binary responses.

- This rule weights both types of mistakes (FN and FP) the same.
- Is threshold of 0.5 appropriate if a false positive is worse than a false negative? E.g., is it worse to mark an innocent person as `guilty`, or mark a guilty person as `not guilty`? May want to change the decision rule to assign:

$$\begin{cases} \texttt{guilty} & \text{if } P(Y = 1|X) > 0.8; \\ \texttt{not guilty} & \text{if } P(Y = 1|X) \leq 0.8. \end{cases}$$

- But sometimes we care more about lowering false negatives. E.g., a credit card company trying to detect a fraudulent charge.
  May want to change the decision rule to assign:

$$\begin{cases} \texttt{fraudulent charge} & \text{if } P(Y = 1|X) > 0.2; \\ \texttt{not fraudulent charge} & \text{if } P(Y = 1|X) \leq 0.2. \end{cases}$$

- What happens to TP rate and FP rate as threshold decreases?

Decision rule threshold

For some threshold value $c \in [0, 1]$, consider the decision rule

$$\begin{cases} \texttt{positive} & \text{if } P(Y = 1|X) > c; \\ \texttt{negative} & \text{if } P(Y = 1|X) \leq c. \end{cases}$$

What happens to TP rate and FP rate as threshold c increases from 0 to 1?

ROC curve

The *ROC curve* simultaneously displays both types of errors for all thresholds.

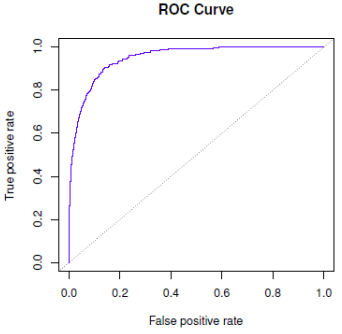


Figure: Image by James et al. (2021). An *ROC curve* for LDA classifier on `Default` data. Dotted line represents "no information" classifier, i.e., one that doesn't use predictors.

- ROC curve is parameterized by the possible threshold values.
- Overall performance of a classifier, summarized over all possible thresholds, is given by the *area under the ROC curve (AUC)*. The larger the AUC, the better the classifier.
- StatQuest: ROC and AUC, Clearly Explained! (16:16)

Classification with more than two classes

We sometimes wish to classify *a response variable that has more than two classes.*

- Extend the two-class logistic regression approach to the setting of $K > 2$ classes.
- Many ideas are the same as in binary classification, but notation can be heavier.

Multinomial logistic regression model

Model "something like log-odds" using a linear function of the predictors.

1. Select a class to serve as the baseline; WLOG, select the Kth class for this role.
2. Define

$$\alpha_k(x) \coloneqq \beta_{k,0} + \beta_{k,1}x_1 + \cdots + \beta_{k,p}x_p \qquad \text{for } k = 1, \ldots, K-1. \tag{10}$$

Replace the binary log-odds model (5) with the model

$$\log\left(\frac{P(Y=k \mid X=x)}{P(Y=k \mid X=x)}\right) = \begin{cases} \alpha_k(x) & \text{for } k = 1, \ldots, K-1, \\ 0 & \text{for } k = K. \end{cases} \tag{11}$$

The resulting conditional probability is then

$$P(Y=k \mid X=x) = \begin{cases} \frac{e^{\alpha_k(x)}}{1+\sum_{l=1}^{K-1} e^{\alpha_l(x)}} & \text{for } k = 1, \ldots, K-1, \\ \frac{1}{1+\sum_{l=1}^{K-1} e^{\alpha_l(x)}} & \text{for } k = K. \end{cases} \tag{12}$$

Note:

- The denominators in (12) are defined to ensure that $\sum_{k=1}^{K} P(Y=k \mid X=x)$ equals 1.
- Model (5) is a special case of model (11);
  Model (6) is a special case of model (12).

Interpretation

Each of the first $K-1$ classes has its own set of separate regression coefficients.

- E.g., consider conditions `Appendicitis`, `Food poisoning`, and `Gastritis`.
  For $j = 1, \ldots, p$, consider $x_j =$ `Severity of symptom` j (e.g. how much does head hurt?).
- For `Appendicitis`, we want to define $\beta_{\texttt{Appendicitis},\,j}$ for $j = 0, 1, \ldots, p$.
- For `Food poisoning`, we want to define $\beta_{\texttt{Food poisoning},\,j}$ for $j = 0, 1, \ldots, p$.
- We don't need regression coefficients for baseline class `Gastritis`.

Consider classifying ER visits into `Appendicitis`, `Food poisoning`, `Gastritis`.

- Suppose we set `Gastritis` as the baseline.
- If X_j increases by one unit, then the log odds

$$\log\left(\frac{P(Y = \texttt{Food poisoning} \mid X = x)}{P(Y = \texttt{Gastritis} \mid X = x)}\right)$$

  increases by $\beta_{\texttt{Food poisoning},\,j}$.
- If X_j increases by one unit, then the probability

$$P(Y = \texttt{Food poisoning} \mid X = x)$$

  increases by a messy function of coefficients and current predictor values.

Code walkthrough

Here is a code walkthrough for multinomial logistic regression (the ISLR2 textbook doesn't have one).

Alternative coding: softmax coding

Softmax coding (used extensively in some areas of machine learning) treats all K classes symmetrically:

$$P(Y=k \mid X=x) \quad = \quad \frac{e^{\alpha_k(x)}}{\sum_{l=1}^{K} e^{\alpha_l(x)}} \qquad \text{for } k = 1, \ldots, K$$

- Thus, we estimate coefficients for all K classes (rather than for just $K-1$ classes).
- The log odds ratio between the kth and lth classes equals

$$\begin{aligned} \log\left(\frac{P(Y=k \mid X=x)}{P(Y=l \mid X=x)}\right) &= \alpha_k(x) - \alpha_l(x) \\ &= (\beta_{k,0} - \beta_{l,0}) + (\beta_{k,1} - \beta_{l,1})x_1 + \cdots + (\beta_{k,p} - \beta_{l,p})x_p. \end{aligned}$$

Interpretation: if X_j increases by one unit, then the log odds

$$\log\left(\frac{P(Y = \texttt{Food poisoning} \mid X=x)}{P(Y = \texttt{Appendicitis} \mid X=x)}\right)$$

increases by $(\beta_{\texttt{Food poisoning},j} - \beta_{\texttt{Appendicitis},j})$.

Motivation

Recall: logistic regression directly models $P(Y = k \mid X = x)$ for binary responses by using the logistic function.

Pros:

- We can see impact of e.g., a unit increase in X_j on log odds.
- Assumptions are relatively loose: independence of errors, a linear relationship between the logit and predictor variables, and absence of severe multicollinearity among predictor variables.

Some issues:

- If the sample size is small, other approaches can be more accurate than logistic regression.

Bayes classifier

Recall: *Bayes classifier* assigns observation with predictor x to the class

$$\underset{k \in \{1,2,\ldots,K\}}{\arg\max}\ p_k(x) \tag{13}$$

What medical condition will the Bayes classifier pick for someone with headache pain level x?

$p_k(x)$	$k =$Appendicitis	$k =$Food poisoning	$k =$Gastritis	$\arg\max_k p_k(x)$
$x = 0$	0.2	0.25	0.55	
...	...	...	...	
$x = 4$	0.4	0.25	0.35	
...	...	...	...	
$x = 9$	0.7	0.25	0.05	

Bayes classifier...

- ...performs better than any other classifier *if all misclassifications are weighted the same.*
- ...requires knowing $p_k(x)$ for all k, which we usually don't. Let's try estimating!

Using Bayes' theorem to decompose the *posterior* probability $p_k(x)$...

...into more manageable quantities that we can try to estimate.

$$p_k(x) \;\coloneqq\; P(Y=k \mid X=x) \;=\; \frac{\pi_k f_k(x)}{\sum_{\ell=1}^{K} \pi_\ell f_\ell(x)}, \tag{14}$$

- $\pi_k \coloneqq P(Y=k)$ is the overall or *prior* probability that a randomly chosen observation comes from the kth class.
  - Here π is just a variable name — not the same as $\pi = 3.14159\ldots$
- f_k is the predictor's PMF/PDF given that the response is from the kth class.
  - PMF case: $f_k(x) = P(X=x \mid Y=k)$.

We typically don't know either π_k or f_k.

- Can estimate π_k by the proportion of observed elements in the kth class. E.g.,

k	1	2	3
n_k	3	2	5
$\hat{\pi}_k$	3/10	2/10	5/10

- Estimating f_k is more challenging — typically requires a huge amount of data unless *strong simplifying assumptions* are made.
  - Bias-variance tradeoff: assumptions introduce some bias in order to reduce variance.
  - Two approaches (LDA and Naive Bayes) will be discussed in the next few slides.

Assumptions of f_k in LDA for $p = 1$

The *linear discriminant analysis (LDA)* approach estimates f_k by assuming:

1. f_k is a normal/Gaussian PDF, i.e. for all x holds

$$f_k(x) = \frac{1}{\sqrt{2\pi}\sigma_k} \exp\left\{ -\frac{1}{2\sigma_k^2}(x - \mu_k)^2 \right\}. \tag{15}$$

   - This class will focus on $p = 1$.
   - But if $p > 1$, replace μ_k with p-tuple and replace σ_k with $p \times p$ covariance matrix Σ_k.

2. Same variance parameter across all K classes: $\sigma_1^2 = \sigma_2^2 = \cdots = \sigma_K^2 = \sigma^2$.
   - The *quadratic discriminant analysis (QDA)* relaxes this *equal-variance* assumption, but we won't discuss QDA in this class.

StatQuest: Linear Discriminant Analysis (LDA) clearly explained. (15:11)

Bayes decision boundary

With these assumptions, we plug the PDF (15) into the posterior probability (14) to get

$$p_k(x) = \frac{\pi_k \frac{1}{\sqrt{2\pi}\sigma} \exp\left\{-\frac{1}{2\sigma^2}(x-\mu_k)^2\right\}}{\sum_{\ell=1}^{K} \pi_\ell \frac{1}{\sqrt{2\pi}\sigma} \exp\left\{-\frac{1}{2\sigma^2}(x-\mu_\ell)^2\right\}}. \tag{16}$$

- The class assigned by the Bayes classifier is equivalent to the class

$$\underset{k\in\{1,2,\ldots,K\}}{\arg\max}\ \delta_k(x) \qquad \text{where } \delta_k(x) \coloneqq x \cdot \frac{\mu_k}{\sigma^2} - \frac{\mu_k^2}{2\sigma^2} + \log(\pi_k). \tag{17}$$

  (See next slide for proof.) Here δ_k is called a *discriminant function*.
- If $K = 2$, classifier assigns x to class 1 if $\delta_1(x) > \delta_2(x)$, to class 2 otherwise.
- If also $p = 1$, the *Bayes decision boundary* is the value x for which $\delta_1(x) = \delta_2(x)$.
  - What does this inequality $\delta_1(x) > \delta_2(x)$ and boundary simplify to if $\pi_1 = \pi_2$?

Bayes decision boundary

We claim that $\boxed{\underset{k\in\{1,2,\ldots,K\}}{\arg\max}\ p_k(x) = \underset{k\in\{1,2,\ldots,K\}}{\arg\max}\ \delta_k(x)}$. Proof:

$$\underset{k\in\{1,2,\ldots,K\}}{\arg\max}\ p_k(x) = \underset{k\in\{1,2,\ldots,K\}}{\arg\max}\ \frac{\pi_k \frac{1}{\sqrt{2\pi}\sigma}\exp\left\{-\frac{1}{2\sigma^2}(x-\mu_k)^2\right\}}{\sum_{\ell=1}^{K}\pi_\ell \frac{1}{\sqrt{2\pi}\sigma}\exp\left\{-\frac{1}{2\sigma^2}(x-\mu_\ell)^2\right\}} \tag{18}$$

$$= \underset{k\in\{1,2,\ldots,K\}}{\arg\max}\ \pi_k \exp\left\{-\frac{1}{2\sigma^2}(x-\mu_k)^2\right\} \tag{19}$$

$$= \underset{k\in\{1,2,\ldots,K\}}{\arg\max}\ \log\left(\pi_k \exp\left\{-\frac{1}{2\sigma^2}(x-\mu_k)^2\right\}\right) \tag{20}$$

$$= \underset{k\in\{1,2,\ldots,K\}}{\arg\max}\ \log(\pi_k) - \frac{1}{2\sigma^2}(x-\mu_k)^2 \tag{21}$$

$$= \underset{k\in\{1,2,\ldots,K\}}{\arg\max}\ \log(\pi_k) - \frac{1}{2\sigma^2}(x^2 - 2x\mu_k + \mu_k^2) \tag{22}$$

$$= \underset{k\in\{1,2,\ldots,K\}}{\arg\max}\ \log(\pi_k) - \frac{1}{2\sigma^2}(-2x\mu_k + \mu_k^2) \tag{23}$$

$$= \underset{k\in\{1,2,\ldots,K\}}{\arg\max}\ \delta_k(x). \tag{24}$$

Example for decision boundary

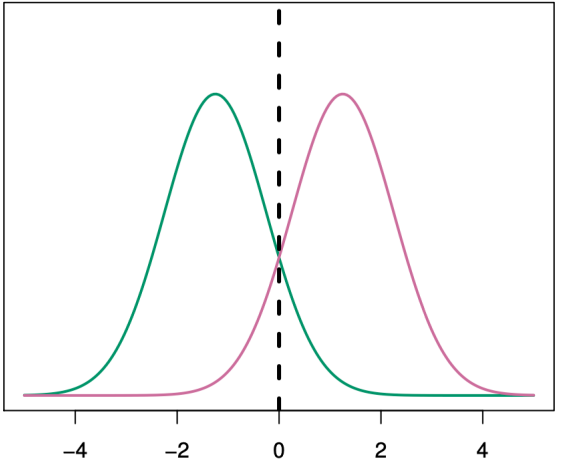


Figure: Image by James et al. (2021). Two PDFs of normal distributions with means $\mu_1 = -1.25$ and $\mu_2 = 1.25$, and variance $\sigma^2 = 1$. The dashed vertical line represents the Bayes decision boundary, so we assign the observation to class 1 if x is left of the line, and to class 2 otherwise.

The LDA method for $p = 1$

In the plot above, we can calculate the Bayes classifier because we know values for all parameters $\pi_1, \ldots, \pi_K, \mu_1, \ldots, \mu_K, \sigma^2$.

- In practice, we must estimate these parameters to apply the Bayes classifier.
- Consider the estimates

$$\hat{\pi}_k = \frac{n_k}{n}, \qquad \hat{\mu}_k = \frac{1}{n_k} \sum_{i:y_i=k} x_i, \qquad \hat{\sigma}^2 = \frac{1}{n-K} \sum_{k=1}^{K} \sum_{i:y_i=k} (x_i - \hat{\mu}_k)^2,$$

  - $\hat{\pi}_k$ is the proportion of all training observations from kth class.
  - $\hat{\mu}_k$ is the average of all training observations from kth class.
  - $\hat{\sigma}^2$ is a weighted average of sample variances for each class.
- Plug these estimates into the definition of $\delta_k(x)$ in (17) to get the *LDA discriminant function*

$$\hat{\delta}_k(x) \;=\; x \cdot \frac{\hat{\mu}_k}{\hat{\sigma}^2} - \frac{\hat{\mu}_k^2}{2\hat{\sigma}^2} + \log(\hat{\pi}_k), \tag{25}$$

which is linear in x, hence the "linear" in LDA.
(The QDA approach will produce a discriminant function that is quadratic in x.)

Example for decision boundary

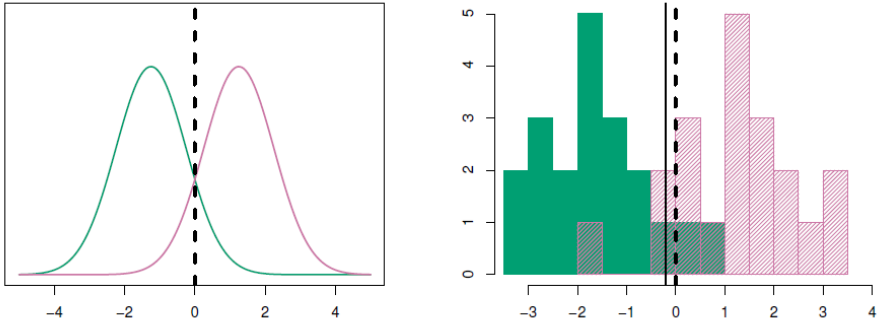


Figure: Image by James et al. (2021). Left: Two PDFs of normal distributions with means $\mu_1 = -1.25$ and $\mu_2 = 1.25$,, and variance $\sigma^2 = 1$. The dashed vertical line represents the Bayes decision boundary, so we assign the observation to class 1 if $x < 0$ and class 2 otherwise. Right: 20 observations were drawn from each of the two classes, and are shown as histograms. The Bayes decision boundary is shown as a dashed vertical line, and the solid vertical line represents the LDA decision boundary estimated from the training data.

Calculation example

$K = 2$ classes (class "0" and "1")

Calculate the LDA discrimination function (25) for $k = 0$ and $k = 1$ given the five data points

$$(9, 1), (8, 0), (6, 0), (7, 1), (4, 0).$$

.

Naive Bayes classifier

The *naive Bayes classifier* assumes for each class $k = 1, \dots, K$ that:

Within the kth class, the p predictors are independent.

Mathematically, this means that for each class k:

$$f_k(x) \;=\; f_{k,1}(x_1) \times f_{k,2}(x_2) \times \cdots \times f_{k,p}(x_p) \tag{26}$$

where $f_{k,j}$ is the PDF/PMF of the jth predictor for observations in the kth class.

- Plug (26) into (14) to get the posterior probability

$$p_k(x) \quad = \quad P(Y = k \mid X = x) \quad = \quad \frac{f_{k,1}(x_1) \times f_{k,2}(x_2) \times \cdots \times f_{k,p}(x_p) \times \pi_k}{\sum_{\ell=1}^{K} f_{\ell,1}(x_1) \times \cdots \times f_{\ell,2}(x_2) \times f_{\ell,p}(x_p) \times \pi_\ell}. \tag{27}$$

- The *naive Bayes classifier* will assign an observation with predictor x to the class that maximizes posterior probability (27).
- The independence assumption, though often unrealistic, produces decent results, especially when n is too small to effectively estimate the joint distribution f_k.

Naive Bayes allows both quantitative and qualitative predictors

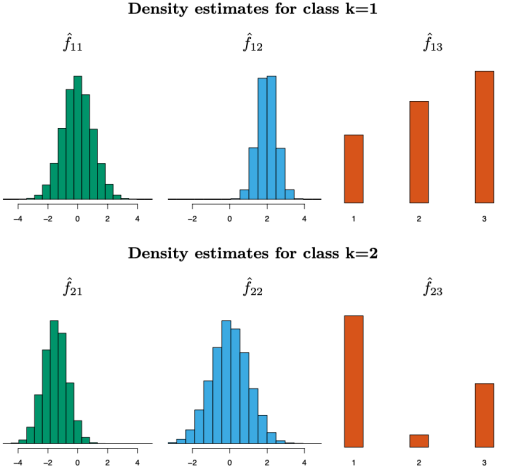


FIGURE 4.10. *In the toy example in Section 4.4.4, we generate data with $p = 3$ predictors and $K = 2$ classes. The first two predictors are quantitative, and the third predictor is qualitative with three levels. In each class, the estimated density for each of the three predictors is displayed. If the prior probabilities for the two classes are equal, then the observation $x^* = (0.4, 1.5, 1)^T$ has a 94.4% posterior probability of belonging to the first class.*

Figure: Image by James et al. (2021).

Estimation approach

Estimating the posterior probability (27) requires estimating the univariate density functions $f_{k,j}$ for all classes $k = 1, \dots, K$ and all predictors $j = 1, \dots, p$. Some options:

- If X_j is quantitative, we can assume $X_j | Y = k \sim \mathcal{N}(\mu_{k,j}, \sigma^2_{k,j})$ (as in LDA).
- If X_j is qualitative, we could count the proportion of training observations for the jth predictor corresponding to each class k.
  - E.g. suppose we want to predict whether a student studies more than 10 hours per week based on their major (so $p = 1$). We survey 100 people:

	Math major	Art major	Poli Sci major
Study $>$ 10 hr /wk	20	15	5
Study $\leq$ 10 hr /wk	15	25	20

We use proportions (i.e., divide each cell by row sum) to estimate the "true" PMFs $f_{k,j}$ (each row is a PMF):

	Math major	Art major	Poli Sci major
Study $>$ 10 hr /wk	20/40	15/40	5/40
Study $\leq$ 10 hr /wk	15/60	25/60	20/60

Another exposition on Naive Bayes

StatQuest: Naive Bayes, Clearly Explained!!! (15:11)

Comparison

Analytical (or mathematical) comparison:

- No method uniformly dominates others: The appropriate model depends on the predictor's distribution in each class as well as n and p.
- K-nearest neighbors (KNN) is a flexible approach (if K is small); it can dominate LDA and naive Bayes when the true decision boundary is highly non-linear. However, KNN requires many observations relative to the number of predictors to perform well.

For an *empirical* (or data-based) comparison, see Section 4.5.2 of ISLR2 textbook.

More

We have just scratched the surface

- For more, see STA 138 (Analysis of Categorical Data).