DATA 180:
Intro to Data Science

Week 13: Classification: Logistic regression

Agenda

Intro to probability

Logistic regression

- Intro
- Estimating coefficients

Linear Discriminant Analysis (LDA)

- Bayes' Theorem
- Extension: Quadratic Discriminant Analysis (QDA)

Evaluation of classification results

Types of response variables

Regression refers to the type of supervised learning models with a non-binary response variable, for example:

- Credit card balance of customers.
- Students' grade from a class.

Classification refers to the type of supervised learning models with a binary response variable, for example:

- Is this email a spam or not?
- Is this patient diagnosed with cancer or not?
- Is this picture a cat or not?

Types of response variables

Regression refers to the type of supervised learning models with a non-binary response variable, for example:

- Credit card balance of customers.
- Students' grade from a class.

Classification refers to the type of supervised learning models with a binary response variable, for example:

- Is this email a spam or not?
- Is this patient diagnosed with cancer or not?
- Is this picture a cat or not?

However sometimes we cannot answer these questions with certainty, but we are "xx percent sure"..

Types of response variables

Regression refers to the type of supervised learning models with a non-binary response variable, for example:

- Credit card balance of customers.
- Students' grade from a class.

Classification refers to the type of supervised learning models with a binary response variable, for example:

- Is this email a spam or not?
- Is this patient diagnosed with cancer or not?
- Is this picture a cat or not?

However sometimes we cannot answer these questions with certainty, but we are "xx percent sure".. (probability)

Toy example: flip a coin

Flip a fair coin: what is the chance that I get heads? What is the chance that I get tails? Are the chances of getting tails or heads equal?

Toy example: flip a coin

Flip a fair coin: what is the chance that I get heads? What is the chance that I get tails?

- 50% chance of getting heads, 50% chance of getting tails
- Chances are equal because it is a fair coin

- Does that mean I flip a coin twice and will get 1 head and 1 tail?
- Does that mean I flip two coins at the same time, I will get 1 head and 1 tail?
- Does that mean I flip 1000 coins at the same time, I will get 500 heads and 500 tails?

Toy example: flip a coin

Flip a fair coin: what is the chance that I get heads? What is the chance that I get tails?

- 50% chance of getting heads, 50% chance of getting tails
- Chances are equal because it is a fair coin

- ~~Does that mean I flip a coin twice and will get 1 head and 1 tail?~~
- ~~Does that mean I flip two coins at the same time, I will get 1 head and 1 tail?~~
- ~~Does that mean I flip 1000 coins at the same time, I will get 500 heads and 500 tails?~~

Definitions

Definition: A ***probability experiment*** is an act or process of observation that leads to a single outcome that cannot be predicted with certainty.

Definition: The ***probability*** of an event is the proportion of times the event occurs over the long run, as a probability experiment is repeated over and over again, *i.e.,* it is the relative frequency with which that outcome occurs.

Notation for Probabilities

<u>Definition</u>: A ***probability model*** for a probability experiment consists of a sample space along with a probability for each event.

- P denotes probability.
- Events are typically denoted with capital letters, e.g., A, B, and C denote specific events.
- If A is an event, the probability of the event A is denoted $P(A)$.

How does this apply in classification?

We tried to predict if a movie is a good movie (rating > 7) or not (rating < 7) based on its features:

rating	Critic_rating	Trailer_views	X3D_available	Time_taken	Twitter_hastags	Genre	Avg_age_actors	Num_multiplex	Collection	class
7.995	7.94	527367	YES	109.60	223.840	Thriller	23	494	48000	1
7.470	7.44	494055	NO	146.64	243.456	Drama	42	462	43200	1
7.515	7.44	547051	NO	147.88	2022.400	Comedy	38	458	69400	1
7.020	8.26	516279	YES	185.36	225.344	Drama	45	472	66800	1
7.070	8.26	531448	NO	176.48	225.792	Drama	55	395	72400	1
7.005	7.26	498425	YES	143.48	284.592	Comedy	53	460	57400	1
7.400	8.96	459241	YES	139.16	243.664	Thriller	41	522	45800	1
7.170	7.96	400821	NO	116.84	243.536	Drama	56	571	44200	1
7.000	7.96	295168	YES	118.60	242.640	Comedy	55	564	33000	1
6.855	7.96	412012	YES	189.56	283.024	Thriller	45	508	37800	1
7.060	8.96	369595	NO	120.00	222.400	Drama	29	578	30000	1

How does this apply in classification?

We tried to predict if a movie is a good movie (rating > 7) or not (rating < 7) based on its features:

rating	Critic_rating	Trailer_views	X3D_available	Time_taken	Twitter_hastags	Genre	Avg_age_actors	Num_multiplex	Collection	class
7.995	7.94	527367	YES	109.60	223.840	Thriller	23	494	48000	1
7.470	7.44	494055	NO	146.64	243.456	Drama	42	462	43200	1
7.515	7.44	547051	NO	147.88	2022.400	Comedy	38	458	69400	1
7.020	8.26	516279	YES	185.36	225.344	Drama	45	472	66800	1
7.070	8.26	531448	NO	176.48	225.792	Drama	55	395	72400	1
7.005	7.26	498425	YES	143.48	284.592	Comedy	53	460	57400	1
7.400	8.96	459241	YES	139.16	243.664	Thriller	41	522	45800	1
7.170	7.96	400821	NO	116.84	243.536	Drama	56	571	44200	1
7.000	7.96	295168	YES	118.60	242.640	Comedy	55	564	33000	1
6.855	7.96	412012	YES	189.56	283.024	Thriller	45	508	37800	1
7.060	8.96	369595	NO	120.00	222.400	Drama	29	578	30000	1

Using the language of probability, we are predicting the **probability of the ith movie being a good movie,** given the predictors:

$$Prob(y_i = 1|x_i) = Prob(class_i = 1|x_i)$$

Basic rules of probability

If you roll a fair dice, is it possible to get a 10? What is the chance of getting a 10?

- It is impossible to get a 10.
  - Probability = 0 means impossible

Basic rules of probability

- If you roll a fair dice, is it possible to get a 10? What is the chance of getting a 10?
  - It is impossible to get a 10.
    - Probability = 0 means impossible

- If you roll a fair dice, is it possible to have a number in {1, 2, 3, 4, 5, 6}? What is the chance of getting a number from {1, 2, 3, 4, 5, 6}?

Basic rules of probability

- If you roll a fair dice, is it possible to get a 10? What is the chance of getting a 10?
  - It is impossible to get a 10.
    - Probability = 0 means *impossible to happen*

- If you roll a fair dice, is it possible to have a number in {1, 2, 3, 4, 5, 6}? What is the chance of getting a number from {1, 2, 3, 4, 5, 6}?
  - You are absolutely going to get a number from {1, 2, 3, 4, 5, 6}.
    - Probability = 1 means *absolutely going to happen*

Basic Rules of Probability

- Any probability is always a numerical value between zero and one. The probability is zero if the event cannot occur. The probability is one if the event is a sure thing, i.e., it occurs every time:

$$0 \leq P(A) \leq 1$$

- Note that a probability close to zero indicates that the event is *unlikely* to occur, while a probability close to one indicates that the event is *likely* to occur.

How does this apply in classification?

We tried to predict if a movie is a good movie (rating > 7) or not (rating < 7) based on its features:

rating	Critic_rating	Trailer_views	X3D_available	Time_taken	Twitter_hastags	Genre	Avg_age_actors	Num_multiplex	Collection	class
7.995	7.94	527367	YES	109.60	223.840	Thriller	23	494	48000	1
7.470	7.44	494055	NO	146.64	243.456	Drama	42	462	43200	1
7.515	7.44	547051	NO	147.88	2022.400	Comedy	38	458	69400	1
7.020	8.26	516279	YES	185.36	225.344	Drama	45	472	66800	1
7.070	8.26	531448	NO	176.48	225.792	Drama	55	395	72400	1
7.005	7.26	498425	YES	143.48	284.592	Comedy	53	460	57400	1
7.400	8.96	459241	YES	139.16	243.664	Thriller	41	522	45800	1
7.170	7.96	400821	NO	116.84	243.536	Drama	56	571	44200	1
7.000	7.96	295168	YES	118.60	242.640	Comedy	55	564	33000	1
6.855	7.96	412012	YES	189.56	283.024	Thriller	45	508	37800	1
7.060	8.96	369595	NO	120.00	222.400	Drama	29	578	30000	1

Using the language of probability, we are predicting the **probability of the ith movie being a good movie**:

$$Prob(class_i = 1|x_i)$$

Using this notation, what is the probability of the ith movie being a bad movie?

How does this apply in classification?

We tried to predict if a movie is a good movie (rating > 7) or not (rating < 7) based on its features:

rating	Critic_rating	Trailer_views	X3D_available	Time_taken	Twitter_hastags	Genre	Avg_age_actors	Num_multiplex	Collection	class
7.995	7.94	527367	YES	109.60	223.840	Thriller	23	494	48000	1
7.470	7.44	494055	NO	146.64	243.456	Drama	42	462	43200	1
7.515	7.44	547051	NO	147.88	2022.400	Comedy	38	458	69400	1
7.020	8.26	516279	YES	185.36	225.344	Drama	45	472	66800	1
7.070	8.26	531448	NO	176.48	225.792	Drama	55	395	72400	1
7.005	7.26	498425	YES	143.48	284.592	Comedy	53	460	57400	1
7.400	8.96	459241	YES	139.16	243.664	Thriller	41	522	45800	1
7.170	7.96	400821	NO	116.84	243.536	Drama	56	571	44200	1
7.000	7.96	295168	YES	118.60	242.640	Comedy	55	564	33000	1
6.855	7.96	412012	YES	189.56	283.024	Thriller	45	508	37800	1
7.060	8.96	369595	NO	120.00	222.400	Drama	29	578	30000	1

Using the language of probability, we are predicting the **probability of the ith movie being a good movie**:

$$Prob(class_i = 1|x_i)$$

Using this notation, what is the probability of the ith movie being a bad movie?

$$Prob(y_i = 0) = 1 - Prob(class_i = 1|x_i)$$

The complement rule in Probability

If event A and event $\bar{A}$ cannot happen together (e.g., being a good movie and a bad movie), then event A and event $\bar{A}$ are called *complementary events,* and their probabilities add to 1:

$$Prob(A) + Prob(\bar{A}) = 1$$

The complement rule in Probability

If event A and event $\bar{A}$ cannot happen together (e.g., being a good movie and a bad movie), then event A and event $\bar{A}$ are called *complementary events,* and their probabilities add to 1:

$$Prob(A) + Prob(\bar{A}) = 1$$

Therefore, in classification, we only need to calculate:

$$Prob(y_i = 1|x_i),$$

which is the probability of the response variable equaling to 1, given the values of the predictors.

Agenda

Intro to probability

Logistic regression

- Intro
- Estimating coefficients

Linear Discriminant Analysis (LDA)

- Bayes' Theorem
- Extension: Quadratic Discriminant Analysis (QDA)

Evaluation of classification results

From linear regression to logistic regression

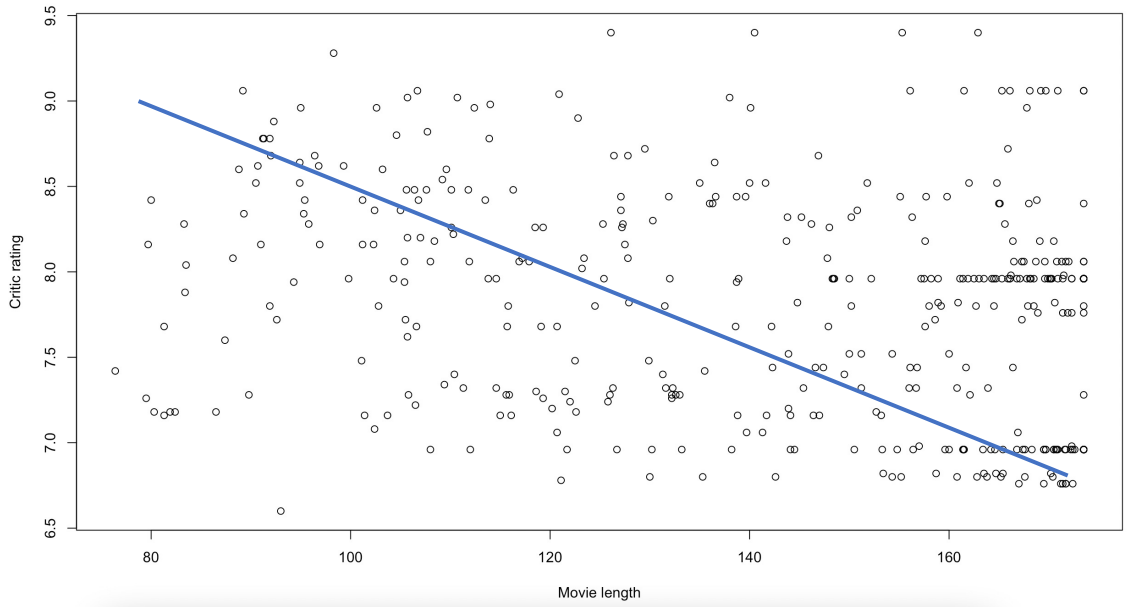
Linear regression assumes a linear relationship between **predictors $\boldsymbol{X}$ and the response variable $\boldsymbol{y}$, where the response variable y is NOT binary**:

$$y = b_0 + b_1 x_1 + b_2 x_2 + \ \dots + b_D x_D,$$

where $\{b_0, b_1, \dots, b_D\}$ is the parameter set that the OLS method could solve for.

Intuition behind logistic regression

Because a linear regression line takes the form:



You cannot form a line if the response variable is only either 0 or 1.

Intuition behind logistic regression

Therefore, logistic regression takes the following form:

$$Prob(y = 1) = b_0 + b_1 x_1 + b_2 x_2 + \ldots + b_D x_D,$$

In other words, the predictors are NOT used to predict the binary value of the response variable y, BUT to predict the probability of it occurring.

Intuition behind logistic regression

Therefore, logistic regression takes the following form:

$$Prob(y = 1) = b_0 + b_1 x_1 + b_2 x_2 + \ldots + b_D x_D,$$

In other words, the predictors are NOT used to predict the binary value of the response variable y, BUT to predict the probability of it occurring.

Something is wrong with this equation.. What is it?

Intuition behind logistic regression

Therefore, logistic regression takes the following form:

$$Prob(y = 1) = b_0 + b_1 x_1 + b_2 x_2 + \ldots + b_D x_D,$$

In other words, the predictors are NOT used to predict the binary value of the response variable y, BUT to predict the probability of it occurring.

Something is wrong with this equation.. What is it?

- $Prob(y = 1)$ being a probability, it needs to be between 0 and 1.
- But $b_0 + b_1 x_1 + b_2 x_2 + \ldots + b_D x_D$ could be ANY value, and thus might give an invalid probability.

Logistic regression

Therefore, to make sure all the predicted values are valid probabilities, logistic regression takes the following form:

$$Prob(y = 1) = \frac{\exp(b_0 + b_1 x_1 + b_2 x_2 + \ \ldots + b_D x_D)}{1 + \exp(b_0 + b_1 x_1 + b_2 x_2 + \ \ldots + b_D x_D)}$$

where

- the exp() function forces the predicted value to be positive,
- and because the denominator is always tiny bit larger than the numerator, the calculated value will always be less than 1.

Therefore, we can always expect a valid probability as the output.

Why is logistic regression still be considered a generalized linear model?

Therefore, to make sure all the predicted values are valid probabilities, logistic regression takes the following form:

$$Prob(y = 1|X) = \frac{\exp(b_0 + b_1 x_1 + b_2 x_2 + \ \dots + b_D x_D)}{1 + \exp(b_0 + b_1 x_1 + b_2 x_2 + \ \dots + b_D x_D)}$$

Denote $Prob(y = 1|X) = p(X)$. Then this model is equivalent to:

$$\frac{p(X)}{1-p(X)} = \exp(b_0 + b_1 x_1 + b_2 x_2 + \ \dots + b_D x_D).$$

Why is logistic regression still be considered a generalized linear model?

Therefore, to make sure all the predicted values are valid probabilities, logistic regression takes the following form:

$$Prob(y = 1|X) = \frac{\exp(b_0 + b_1 x_1 + b_2 x_2 + \ \ldots + b_D x_D)}{1 + \exp(b_0 + b_1 x_1 + b_2 x_2 + \ \ldots + b_D x_D)}$$

Denote $Prob(y = 1|X) = p(X)$. Then this model is equivalent to:

$$\frac{p(X)}{1-p(X)} = \exp(b_0 + b_1 x_1 + b_2 x_2 + \ \ldots + b_D x_D).$$

$\frac{p(X)}{1-p(X)}$ is the so-called *odds,* and can take any value between 0 to positive infinity.

Why is logistic regression still be considered a generalized linear model?

Therefore, to make sure all the predicted values are valid probabilities, logistic regression takes the following form:

$$Prob(y = 1|X) = \frac{\exp(b_0 + b_1x_1 + b_2x_2 + \ \dots + b_Dx_D)}{1 + \exp(b_0 + b_1x_1 + b_2x_2 + \ \dots + b_Dx_D)}$$

Denote $Prob(y = 1|X) = p(X)$. Then this model is equivalent to:

$$\frac{p(X)}{1-p(X)} = \exp(b_0 + b_1x_1 + b_2x_2 + \ \dots + b_Dx_D).$$

$\frac{p(X)}{1-p(X)}$ is the so-called *odds,* and can take any value between 0 to positive infinity.

Take the log on both sides, we obtain

$$\log\left(\frac{p(X)}{1-p(X)}\right) = b_0 + b_1x_1 + b_2x_2 + \ \dots + b_Dx_D.$$

Why is logistic regression still be considered a generalized linear model?

Therefore, to make sure all the predicted values are valid probabilities, logistic regression takes the following form:

$$Prob(y = 1|X) = \frac{\exp(b_0 + b_1x_1 + b_2x_2 + \ \dots + b_Dx_D)}{1 + \exp(b_0 + b_1x_1 + b_2x_2 + \ \dots + b_Dx_D)}$$

Denote $Prob(y = 1|X) = p(X)$. Then this model is equivalent to:

$$\frac{p(X)}{1-p(X)} = \exp(b_0 + b_1x_1 + b_2x_2 + \ \dots + b_Dx_D).$$

$\frac{p(X)}{1-p(X)}$ is the so-called *odds,* and can take any value between 0 to positive infinity.

Take the log on both sides, we obtain

$$\log\left(\frac{p(X)}{1-p(X)}\right) = b_0 + b_1x_1 + b_2x_2 + \ \dots + b_Dx_D.$$

$\log\left(\frac{p(X)}{1-p(X)}\right)$ is thus the log-odds (or called *logit*). And this *logit* is linear in X.

Interpretation of coefficients

Therefore, to make sure all the predicted values are valid probabilities, logistic regression takes the following form:

$$Prob(y = 1|X) = \frac{\exp(b_0 + b_1x_1 + b_2x_2 + \ \dots + b_Dx_D)}{1 + \exp(b_0 + b_1x_1 + b_2x_2 + \ \dots + b_Dx_D)}$$

Take the log on both sides, we obtain

$$\log\left(\frac{p(X)}{1-p(X)}\right) = b_0 + b_1x_1 + b_2x_2 + \ \dots + b_Dx_D.$$

$\log\left(\frac{p(X)}{1-p(X)}\right)$ is thus the log-odds (or called *logit*). And this *logit* is linear in X.

Thus, the interpretation of b_1 in logistic regression is:

- Every unit change in x_1, the log-odds change by b_1.

Agenda

Intro to probability

Logistic regression

- Intro
- Estimating coefficients

Linear Discriminant Analysis (LDA)

- Bayes' Theorem
- Extension: Quadratic Discriminant Analysis (QDA)

Evaluation of classification results

Logistic regression

Logistic regression is to build a generalized linear relationship between the predictors X and the response variable y, when y is binary:

$$Prob(y = 1) = \frac{\exp(b_0 + b_1 x_1 + b_2 x_2 + \ldots + b_D x_D)}{1 + \exp(b_0 + b_1 x_1 + b_2 x_2 + \ldots + b_D x_D)},$$

where $\{b_0, b_1, \ldots, b_D\}$ is the parameter set that needs to be solved.

The objective of solving a logistic regression

If the true response variable equal to 0, do we want

$$Prob(y = 1) = \frac{\exp(b_0 + b_1 x_1 + b_2 x_2 + \ \ldots + b_D x_D)}{1 + \exp(b_0 + b_1 x_1 + b_2 x_2 + \ \ldots + b_D x_D)}$$

to be small or large?

The objective of solving a logistic regression

If the true response variable equals to 0, do we want

$$Prob(y = 1) = \frac{\exp(b_0 + b_1 x_1 + b_2 x_2 + \ \ldots + b_D x_D)}{1 + \exp(b_0 + b_1 x_1 + b_2 x_2 + \ \ldots + b_D x_D)}$$

to be small or large?

- Small, because Prob(y=1) being small is equivalent to a large $Prob(y = 0)$:

$$Prob(y = 0) = \ 1 - Prob(y = 1)$$

The objective of solving a logistic regression

If the true response variable equals to 0, do we want

$$Prob(y = 1) = \frac{\exp(b_0 + b_1 x_1 + b_2 x_2 + \ \ldots + b_D x_D)}{1 + \exp(b_0 + b_1 x_1 + b_2 x_2 + \ \ldots + b_D x_D)}$$

to be small or large?

- Small, because Prob(y=1) being small is equivalent to a large $Prob(y = 0)$:

$$Prob(y = 0) = \ 1 - Prob(y = 1)$$

If the true response variable equals to 1, do we want

$$Prob(y = 1) = \frac{\exp(b_0 + b_1 x_1 + b_2 x_2 + \ \ldots + b_D x_D)}{1 + \exp(b_0 + b_1 x_1 + b_2 x_2 + \ \ldots + b_D x_D)}$$

to be small or large?

The objective of solving a logistic regression

If the true response variable equals to 0, do we want

$$Prob(y = 1) = \frac{\exp(b_0 + b_1x_1 + b_2x_2 + \ \dots + b_Dx_D)}{1 + \exp(b_0 + b_1x_1 + b_2x_2 + \ \dots + b_Dx_D)}$$

to be small or large?

- Small, because Prob(y=1) being small is equivalent to a large $Prob(y = 0)$:

$$Prob(y = 0) = \ 1 - Prob(y = 1)$$

If the true response variable equals to 1, do we want

$$Prob(y = 1) = \frac{\exp(b_0 + b_1x_1 + b_2x_2 + \ \dots + b_Dx_D)}{1 + \exp(b_0 + b_1x_1 + b_2x_2 + \ \dots + b_Dx_D)}$$

to be small or large?

- Large

The objective of solving a logistic regression

Therefore, in logistic regression, the goal is to find a parameter that maximizes:

$$L(b_0, b_1, \dots, b_D) = \prod_{i:y_i=1} Prob(y_i = 1) \prod_{i:y_i=0} (1 - Prob(y_i = 1))$$

For the sample units that indeed have a response variable equal to 1, we want $Prob(y_i = 1)$ to be large

For the sample units that indeed have a response variable equal to 0, we want $Prob(y_i = 1)$ to be small, and thus $1 - Prob(y_i = 1)$ to be large

The objective of solving a logistic regression

Therefore, in logistic regression, the goal is to find a parameter set that maximizes:

$$L(b_0, b_1, \dots, b_D) = \prod_{i:y_i=1} Prob(y_i = 1) \prod_{i:y_i=0} (1 - Prob(y_i = 1))$$

For the sample units that indeed have a response variable equal to 1, we want $Prob(y_i = 1)$ to be large

For the sample units that indeed have a response variable equal to 0, we want $Prob(y_i = 1)$ to be small, and thus $1 - Prob(y_i = 1)$ to be large

This is called **Maximize Likelihood Estimation** (MLE), the objective function $L(b_0, b_1, \dots, b_D)$ is called the likelihood function.

The objective of solving a logistic regression

Therefore, in logistic regression, the goal is to find a parameter set that maximizes:

$$L(b_0, b_1, \dots, b_D) = \prod_{i:y_i=1} Prob(y_i = 1) \prod_{i:y_i=0} (1 - Prob(y_i = 1))$$

For the sample units that indeed have a response variable equal to 1, we want $Prob(y_i = 1)$ to be large

For the sample units that indeed have a response variable equal to 0, we want $Prob(y_i = 1)$ to be small, and thus $1 - Prob(y_i = 1)$ to be large

This is called **Maximize Likelihood Estimation** (MLE), the objective function $L(b_0, b_1, \dots, b_D)$ is called the likelihood function.

MLE is one of the most common approaches to fit non-linear models. However, it often does not have an analytical solution.

Agenda

Intro to probability

Logistic regression

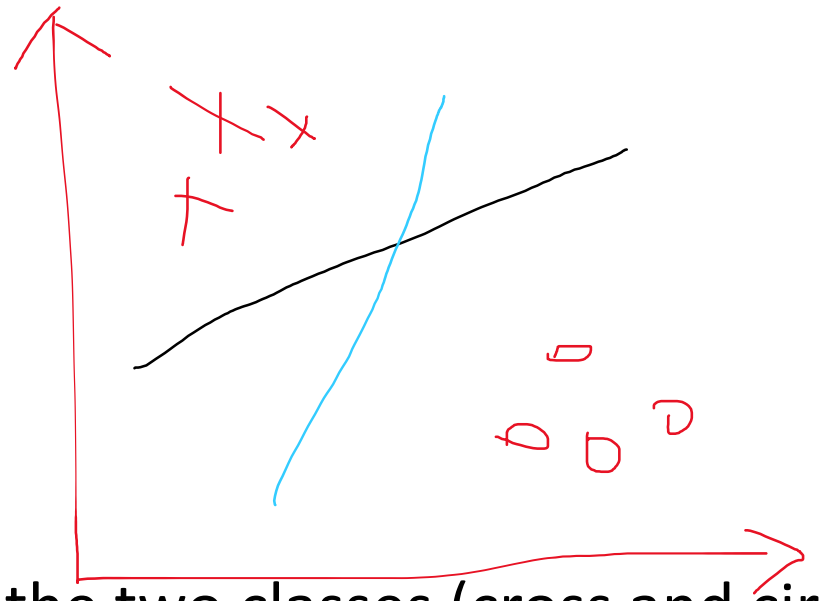
- Intro
- Estimating coefficients

Linear Discriminant Analysis (LDA)

- Bayes' Theorem
- Extension: Quadratic Discriminant Analysis (QDA)

Evaluation of classification results

What happens to the coefficients in logistic regression if the two classes are easy to separate?



When the two classes (cross and circle) is easy to separate, there might be multiple ways to separate. In other words, the estimation might change every time we do an MLE estimation.

Linear Discriminant Analysis

Linear Discriminant Analysis (LDA) is based on Bayes' theorem. Consider the following univariate setting with K classes:

$$\Pr(Y = k|X = x) = \frac{\Pr(X = x|Y = k)\Pr(Y = k)}{\sum_{l=1}^{K}\Pr(X = x|Y = l)\Pr(Y = l)}$$

where $\Pr(Y = k)$ is called the ***prior*** distribution of classes, i.e., the proportion of class k among all sample units.

$\Pr(X = x|Y = k)$ is often denoted as $f_k(x)$, which is the density function for observations that are in class k.

Linear Discriminant Analysis

Linear Discriminant Analysis (LDA) is based on Bayes' theorem. Consider the following univariate setting with K classes:

$$\Pr(Y = k|X = x) = \frac{\Pr(X = x|Y = k)\Pr(Y = k)}{\sum_{l=1}^{K}\Pr(X = x|Y = l)\Pr(Y = l)}$$

where $\Pr(Y = k)$ is called the ***prior*** distribution of classes, i.e., the proportion of class k among all sample units.

$\Pr(X = x|Y = k)$ is often denoted as $f_k(x)$, which is the density function for observations that are in class k.

We calculate $\Pr(Y = k|X = x)$ for $k = 1, \dots, K, ,$ and each sample unit is thus assigned to class *k* where $\Pr(Y = k|X = x)$ is the largest among all K classes.

Linear Discriminant Analysis

Linear Discriminant Analysis (LDA) is based on Bayes' theorem. Consider the following univariate setting with K classes:

$$\Pr(Y = k|X = x) = \frac{\Pr(X = x|Y = k)\Pr(Y = k)}{\sum_{l=1}^{K}\Pr(X = x|Y = l)\Pr(Y = l)}$$

where $\Pr(Y = k)$ is called the ***prior*** distribution of classes, i.e., the proportion of class k among all sample units.

$\Pr(X = x|Y = k)$ is often denoted as $f_k(x)$, which is the density function for observations that are in class k.

How do we calculate $\Pr(Y = k)$ and $\Pr(X = x|Y = k)$?

Linear Discriminant Analysis

Linear Discriminant Analysis (LDA) is based on Bayes' theorem. Consider the following univariate setting with K classes:

$$\Pr(Y = k|X = x) = \frac{\Pr(X = x|Y = k)\Pr(Y = k)}{\sum_{l=1}^{K}\Pr(X = x|Y = l)\Pr(Y = l)}$$

where $\Pr(Y = k)$ is called the ***prior*** distribution of classes, i.e., the proportion of class k among all sample units.

- Obtain from the dataset. $\Pr(Y = k) = \frac{\#\ of\ units\ in\ class\ k}{\#\ of\ sample\ units}$

$\Pr(X = x|Y = k)$ is often denoted as $f_k(x)$, which is the density function for observations that are in class k.

- Make assumptions on the function.

Linear Discriminant Analysis for univariate dataset (p = 1)

LDA assumes that $f_k(x)$ is a normal distribution:

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

i.e., in each class k, the observations are normally distributed with its respective μ_k, σ_k.

Linear Discriminant Analysis for univariate dataset (p = 1)

LDA assumes that $f_k(x)$ is a normal distribution:

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

i.e., in each class k, the observations are normally distributed with its respective μ_k. We further assume all classes have the same variance $\sigma_1^2 = \cdots = \sigma_K^2$.

Plug it back to the LDA model, we obtain:

$$p_k(x) = \frac{\pi_k \frac{1}{\sqrt{2\pi}\sigma} \exp\left(-\frac{1}{2\sigma^2}(x-\mu_k)^2\right)}{\sum_{l=1}^{K} \pi_l \frac{1}{\sqrt{2\pi}\sigma} \exp\left(-\frac{1}{2\sigma^2}(x-\mu_l)^2\right)}$$

where $\pi_k = \Pr(Y = k)$ is the proportion of class k.

Linear Discriminant Analysis for univariate dataset (p = 1)

Plug it back to the LDA model, we obtain:

$$p_k(x) = \frac{\pi_k \frac{1}{\sqrt{2\pi}\sigma} \exp\left(-\frac{1}{2\sigma^2}(x-\mu_k)^2\right)}{\sum_{l=1}^{K} \pi_l \frac{1}{\sqrt{2\pi}\sigma} \exp\left(-\frac{1}{2\sigma^2}(x-\mu_l)^2\right)}$$

where $\pi_k = \Pr(Y = k)$ is the proportion of class k.

Take the log on both sides, LDA is equivalent to assign each observation to the class with the largest δ_k:

$$\delta_k(x) = x \cdot \frac{\mu_k}{\sigma^2} - \frac{\mu_k^2}{2\sigma^2} + \log(\pi_k)$$

Assume binary classification, when would $\delta_1 > \delta_2$?

Take the log on both sides, LDA is equivalent to assign each observation to the class with the largest δ_k:

$$\delta_k(x) = x \cdot \frac{\mu_k}{\sigma^2} - \frac{\mu_k^2}{2\sigma^2} + \log(\pi_k)$$

Assume binary classification, when would $\delta_1 > \delta_2$?

Take the log on both sides, LDA is equivalent to assign each observation to the class with the largest δ_k:

$$\delta_k(x) = x \cdot \frac{\mu_k}{\sigma^2} - \frac{\mu_k^2}{2\sigma^2} + \log(\pi_k)$$

For any observation, if

$$2x\,(\mu_1 - \mu_2) > \mu_1^2 - \mu_2^2,$$

then this observation has a larger δ_1, and it is thus assigned to class 1.

Assume binary classification, when would $\delta_1 > \delta_2$?

Take the log on both sides, LDA is equivalent to assign each observation to the class with the largest δ_k:

$$\delta_k(x) = x \cdot \frac{\mu_k}{\sigma^2} - \frac{\mu_k^2}{2\sigma^2} + \log(\pi_k)$$

For any observation, if

$$2x\,(\mu_1 - \mu_2) > \mu_1^2 - \mu_2^2,$$

then this observation has a larger δ_1, and it is thus assigned to class 1.

$x = \frac{\mu_1^2 - \mu_2^2}{2(\mu_1 - \mu_2)} = \frac{\mu_1 + \mu_2}{2}$ is thus called a Bayes decision boundary, because each side of this boundary corresponds to a different class.

Linear Discriminant Analysis with a single predictor

Thus, LDA is to calculate the probability

$$\Pr(Y = k|X = x) = \frac{\Pr(X = x|Y = k)\Pr(Y = k)}{\sum_{l=1}^{K}\Pr(X = x|Y = l)\Pr(X = x)}$$

for each class k, and a sample unit is assigned to the class with the highest $\Pr(Y = k|X = x)$.

For binary classification, we can simplify it to a sample unit x is assigned to class 1 if

$$2x\left(\mu_1 - \mu_2\right) > \mu_1^2 - \mu_2^2.$$

Linear Discriminant Analysis with a single predictor

Thus, LDA is to calculate the probability

$$\Pr(Y = k|X = x) = \frac{\Pr(X = x|Y = k)\Pr(Y = k)}{\sum_{l=1}^{K}\Pr(X = x|Y = l)\Pr(X = x)}$$

for each class k, and a sample unit is assigned to the class with the highest $\Pr(Y = k|X = x)$.

For binary classification, we can simplify it to a sample unit x is assigned to class 1 if

$$2x\left(\mu_1 - \mu_2\right) > \mu_1^2 - \mu_2^2.$$

What is unknown here?

Linear Discriminant Analysis with a single predictor

Thus, LDA is to calculate the probability

$$\Pr(Y = k|X = x) = \frac{\Pr(X = x|Y = k)\Pr(Y = k)}{\sum_{l=1}^{K}\Pr(X = x|Y = l)\Pr(X = x)}$$

for each class k, and a sample unit is assigned to the class with the highest $\Pr(Y = k|X = x)$.

For binary classification, we can simplify it to a sample unit x is assigned to class 1 if

$$2x\left(\mu_1 - \mu_2\right) > \mu_1^2 - \mu_2^2.$$

We further estimate:

$$\hat{\mu}_k = \frac{1}{n_k}\sum_{i:y_i=k} x_i$$

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

Linear Discriminant Analysis with multiple predictors

For single predictor, LDA assumes that $f_k(x)$ is a normal distribution:

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

i.e., in each class k, the observations are normally distributed with its respective μ_k, σ_k.

Linear Discriminant Analysis with multiple predictors

For single predictor, LDA assumes that $f_k(x)$ is a normal distribution:

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

i.e., in each class k, the observations are normally distributed with its respective μ_k, σ_k.

With multiple predictors, LDA assumes that this is a multivariate normal distribution:

$$f(x) = \frac{1}{(2\pi)^{p/2}|\mathbf{\Sigma}|^{1/2}} \exp\left(-\frac{1}{2}(x-\mu)^T\mathbf{\Sigma}^{-1}(x-\mu)\right)$$

Linear Discriminant Analysis with multiple predictors

For single predictor, LDA assumes that $f_k(x)$ is a normal distribution:

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

i.e., in each class k, the observations are normally distributed with its respective μ_k, σ_k.

With multiple predictors, LDA assumes that this is a multivariate normal distribution:

$$f(x) = \frac{1}{(2\pi)^{p/2}|\mathbf{\Sigma}|^{1/2}} \exp\left(-\frac{1}{2}(x-\mu)^T\mathbf{\Sigma}^{-1}(x-\mu)\right)$$

where μ is a length-P vector, each element is the mean for variable p. μ is different for each class.

Σ is a p*p variance-covariance matrix, where Σ_{ij} is the covariance between two variables. Σ is the same for all the classes.

Agenda

Intro to probability

Logistic regression

- Intro
- Estimating coefficients

Linear Discriminant Analysis (LDA)

- Bayes' Theorem
- Extension: Quadratic Discriminant Analysis (QDA)

Evaluation of classification results

Quadratic Discriminant Analysis

With multiple predictors, LDA assumes that this is a multivariate normal distribution:

$$f(x) = \frac{1}{(2\pi)^{p/2}|\mathbf{\Sigma}|^{1/2}} \exp\left(-\frac{1}{2}(x-\mu)^T \mathbf{\Sigma}^{-1}(x-\mu)\right)$$

where μ is a length-P vector, each element is the mean for variable p. μ is different for each class.

Σ is a p*p variance-covariance matrix, where Σ_{ij} is the covariance between two variables. Σ is the same for all the classes.

QDA is the same as LDA, except it allows Σ to change for each class.

Agenda

Intro to probability

Logistic regression

- Intro
- Estimating coefficients

Linear Discriminant Analysis (LDA)

- Bayes' Theorem
- Extension: Quadratic Discriminant Analysis (QDA)

Evaluation of classification results

The output of a logistic regression

```
> head(movie_log_pred)
        1        30        36        39        40        41 
0.9592492 0.8650059 0.8219064 0.9565668 0.9715923 0.9876120 
```

How to read this output?

The output of a logistic regression

```
> head(movie_log_pred)
        1        30        36        39        40        41 
0.9592492 0.8650059 0.8219064 0.9565668 0.9715923 0.9876120 
```

Logistic regression predicts for the probability instead of the actual class, in other words:

- The model predicts that movie 1 has a 95.9% chance of being a good movie,
- Movie 30 has an 86.5% chance of being a good movie,
- etc.

The output of a logistic regression

```
> head(movie_log_pred)
        1        30        36        39        40        41 
0.9592492 0.8650059 0.8219064 0.9565668 0.9715923 0.9876120 
```

Logistic regression predicts for the probability instead of the actual class, in other words:

- The model predicts that movie 1 has a 95.9% chance of being a good movie,
- Movie 30 has an 86.5% chance of being a good movie,
- etc.

If your audience is impatient and only wants to know if this is a good movie or not (instead of the percentage), what can you do?

Determining the *threshold* for logistic regression

Common practice:

$$y_i = \begin{cases} 1, & if\ Prob(y_i = 1) > 0.5 \\ 0 & , \quad otherwise \end{cases}$$

0.5 is the common threshold to use.

For example, if this movie is predicted to be a good movie with a higher than 50% chance, then this movie is predicted to be a good movie.

In R: Determining the *threshold* for logistic regression

```
> movie_log <- glm(class~., trainingset, family = "binomial")
> movie_log_pred <- predict(movie_log, validationset, type = "response")
> head(movie_log_pred)
        3         4         7         8        11        15
0.9169647 0.9723311 0.9217470 0.9197476 0.8804556 0.8312608
>
> movie_pred_class <- ifelse(movie_log_pred >= 0.5, 1, 0)
> table(validationset$class, movie_pred_class)
   movie_pred_class
     0  1
  0  3 25
  1  5 66
```

Determining the *threshold* for logistic regression

Common practice:

$$y_i = \begin{cases} 1, & if\ Prob(y_i = 1) > 0.5 \\ 0 & , \quad otherwise \end{cases}$$

0.5 is the common threshold to use.

For example, if this movie is predicted to be a good movie with a higher than 50% chance, then this movie is predicted to be a good movie.

But obviously this is judgmental.. Think about a situation where 0.5 seems like a bad choice.

Determining the *threshold* for logistic regression

Common practice:

$$y_i = \begin{cases} 1, & if\ Prob(y_i = 1) > 0.5 \\ 0 & , \quad otherwise \end{cases}$$

0.5 is the common threshold to use.

For example, if this movie is predicted to be a good movie with a higher than 50% chance, then this movie is predicted to be a good movie.

But obviously this is judgmental.. Think about a situation where 0.5 seems like a bad choice:

- Follow this logic, if a model predicts a patient might have a 40% chance of having cancer, do we want to notify the patient or not?
- This **threshold** is very situation-dependent, and you should choose one that aligns with your task.

What if you do not know how to choose this threshold?

We can try different thresholds and see which threshold gives us the best performance on the holdout set.

Assume there is a threshold c, where

$$y_i = \begin{cases} 1, & if\ Prob(y_i = 1) > c \\ 0 & , \ otherwise \end{cases}$$

We first look at the evaluation of the results.

Confusion matrix

A confusion matrix, also called a 2*2 contingency table, compares the predicted class with the actual class.

Confusion Matrix

	Actually Positive (1)	Actually Negative (0)
Predicted Positive (1)	True Positives (TPs)	False Positives (FPs)
Predicted Negative (0)	False Negatives (FNs)	True Negatives (TNs)

Confusion matrix in R: the table() function

The first input will be the rows on the table.

The second input will be the columns on the table.

```
> table(validationset$class, knn_pred)
   knn_pred
     0  1
  0  2 18
  1  3 76
```

In other words, in the validation set:

- there are 2+18 = 20 worse movies, and,
- 3+76 = 79 good movies.

Explain 2, 18, 3, 76?

Confusion matrix in R: the table() function

The first input will be the rows on the table.

The second input will be the columns on the table.

```
> table(validationset$class, knn_pred)
   knn_pred
     0  1
  0  2 18
  1  3 76
```

In other words, in the validation set:

- there are 2+18 = 20 worse movies,
  - 2 of these 20 were accurately predicted by the model
- 3+76 = 79 good movies.
  - 76 of these 79 were accurately predicted by the model.

How well did this model do?

Quality of a classification model

There is **a lot of different ways** to measure the quality of a classification model:

- Accuracy: percentage of correctly identified sample units.

```
> table(validationset$class, knn_pred)
   knn_pred
     0  1
  0  2 18
  1  3 76
```

Accuracy = (2+76)/(2+18+3+76) = 78.8%

Quality of a classification model

There is **a lot of different ways** to measure the quality of a classification model:

- Accuracy: percentage of correctly identified sample units.

```
> table(validationset$class, knn_pred)
   knn_pred
     0  1
  0  2 18
  1  3 76
```

Accuracy = (2+76)/(2+18+3+76) = 78.8%

- What are the other ways?

Quality of classification models

Assume the response variable is *whether a patient has cancer or not*, with y=1 being yes and y=0 being no. Which of the following model has better results?

		Predicted response (Model 1)	
		Yes, this person is predicted to have cancer.	No, this person is predicted to NOT have cancer.
Actual response	Yes, this person indeed has cancer.	10	20
	No, this person does not have cancer.	5	10

		Predicted response (Model 2)	
		Yes, this person is predicted to have cancer.	No, this person is predicted to NOT have cancer.
Actual respons e	Yes, this person indeed has cancer.	10	5
	No, this person does not have cancer.	20	10

Quality of classification models

Assume the response variable is *whether a patient has cancer or not*, with y=1 being yes and y=0 being no. In this confusion matrix:

		Predicted response	
		Yes, this person is predicted to have cancer.	No, this person is predicted to NOT have cancer.
Actual response	Yes, this person indeed has cancer.		
	No, this person does not have cancer.		

What is the worse mistake to make in this case?

Quality of classification models

Assume the response variable is *whether a patient has cancer or not*, with y=1 being yes and y=0 being no. In this confusion matrix:

		Predicted response	
		Yes, this person is predicted to have cancer.	No, this person is predicted to NOT have cancer.
Actual response	Yes, this person indeed has cancer.		
	No, this person does not have cancer.		

What is the worse mistake to make in this case?

- If a person indeed has cancer, but the model predicts otherwise.
- This would be the time we want a model that **can accurately identify patients with cancer** (instead of healthy patients)

Quality of a classification model

There is **a lot of different ways** to measure the quality of a classification model:

- Accuracy: percentage of correctly identified sample units.

```
> table(validationset$class, knn_pred)
   knn_pred
     0  1
  0  2 18
  1  3 76
```

Accuracy = (2+76)/(2+18+3+76) = 78.8%

- Recall (also called sensitivity or True positive rate): percentage of correctly identified positive cases.

Recall = 76/(3+76) = 96.2%

Quality of classification models

		Predicted condition		
	Total population $= P + N$	**Positive (PP)**	**Negative (PN)**	Informedness, bookmaker informedness (BM) $= TPR + TNR - 1$
Actual condition	**Positive (P)**	**True positive (TP),** **hit**	**False negative (FN),** **type II error, miss,** **underestimation**	True positive rate (TPR), recall, sensitivity (SEN), probability of detection, hit rate, power $= \frac{TP}{P} = 1 - FNR$
	Negative (N)	**False positive (FP),** **type I error, false alarm,** **overestimation**	**True negative (TN),** **correct rejection**	False positive rate (FPR), probability of false alarm, fall-out $= \frac{FP}{N} = 1 - TNR$
	Prevalence $= \frac{P}{P + N}$	Positive predictive value (PPV), precision $= \frac{TP}{PP} = 1 - FDR$	False omission rate (FOR) $= \frac{FN}{PN} = 1 - NPV$	Positive likelihood ratio (LR+) $= \frac{TPR}{FPR}$
	Accuracy (ACC) $= \frac{TP + TN}{P + N}$	False discovery rate (FDR) $= \frac{FP}{PP} = 1 - PPV$	Negative predictive value (NPV) $= \frac{TN}{PN} = 1 - FOR$	Markedness (MK), deltaP (Δp) $= PPV + NPV - 1$

Wikipedia link: https://en.wikipedia.org/wiki/Precision_and_recall

What if you do not know how to choose this threshold?

Assume there is a threshold c, where

$$y_i = \begin{cases} 1, & if\ Prob(y_i = 1) > c \\ 0 & , \ otherwise \end{cases}$$

In other words, we are trying to find a threshold c such that all the different accuracies measures are well-balanced on the validation set.

What if you do not know how to choose this threshold?

Assume there is a threshold c, where

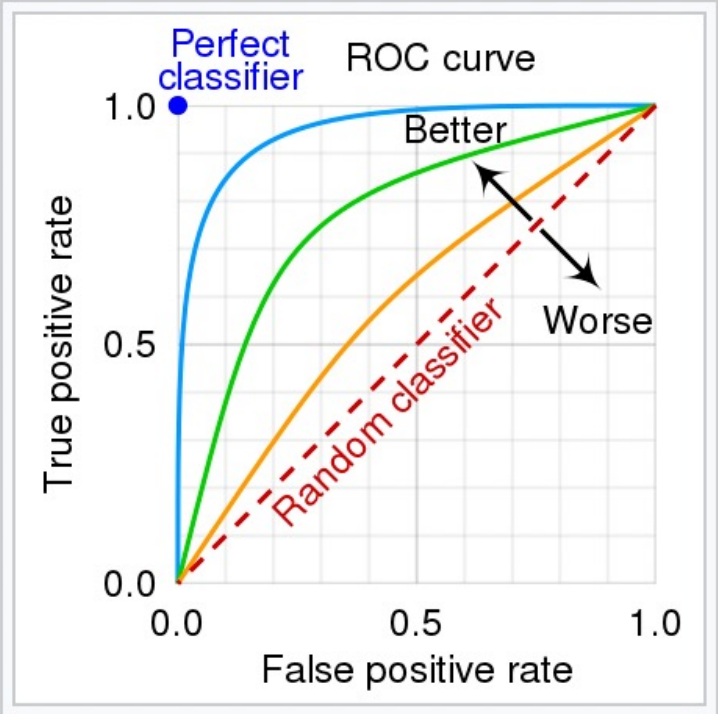
$$y_i = \begin{cases} 1, & if\ Prob(y_i = 1) > c \\ 0 & , \ otherwise \end{cases}$$

In other words, we are trying to find a threshold c such that all the different accuracies measures are well-balanced on the validation set.

How about comparing different models? E.g., LDA, QDA, logistic regression?

ROC curve and AUC value

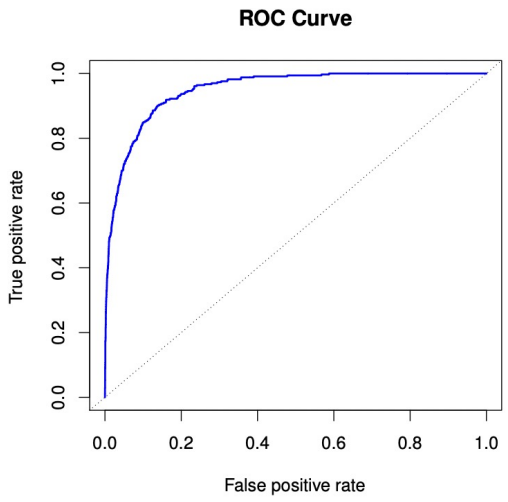
A ROC (*Receiver Operating Characteristics*) curve is to measure the overall performance of a classification model under different thresholds.



ROC curve and AUC value

A ROC (*Receiver Operating Characteristics*) curve is to measure the overall performance of a classification model under different thresholds.

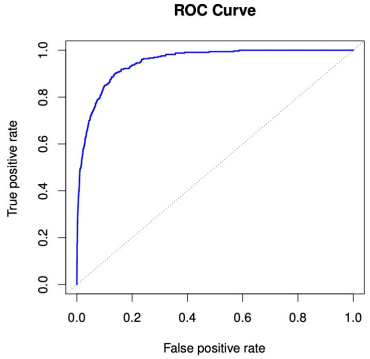
We want the classifier with the ROC curve closer to the left-top corner.
Why?



ROC curve and AUC value

A ROC (*Receiver Operating Characteristics*) curve is to measure the overall performance of a classification model under different thresholds.

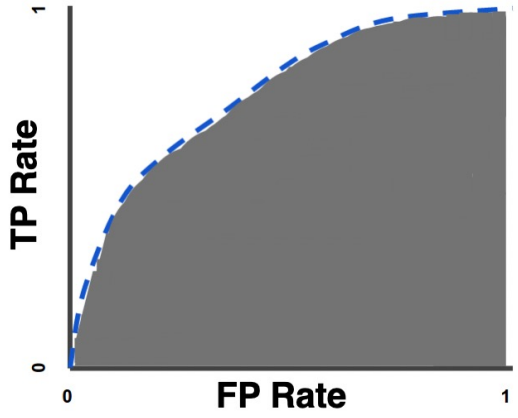
We want the classifier with the ROC curve closer to the left-top corner. Why?



This signifies a more efficient classifier, where we are able to find a threshold that gives a higher true positive but low false positive.

ROC curve and AUC value

We want the classifier with the ROC curve closer to the left-top corner.



AUC *(Area Under the Curve)* value then measures the size of the area under the ROC curve. A ROC curve on the left-top corner corresponds to a higher AUC alue.

How can you tell the quality of validation accuracy?

Assuming you are taking an exam on a subject that **you know nothing about**.

- The exam is all multiple-choice questions, and
- you know that 25% of the exam questions are A, 25% are B, 25% are C, and 25% are D.
- What is your best strategy?

How can you tell the quality of validation accuracy?

Assuming you are taking an exam on a subject that **you know nothing about**.

- The exam is all multiple-choice questions, and
- you know that 25% of the exam questions are A, 25% are B, 25% are C, and 25% are D.
- What is your best strategy?
  - Put the same answer for all questions.

Benchmark in classification problems

Assuming there are 100 sample units in the validation set, and you know that there are 50% of them being in class 1 and the other 50% being in class 2.

- Without running any model, what is your best classification strategy?

Benchmark in classification problems

Assuming there are 100 sample units in the validation set, and you know that there are 50% of them being in class 1 and the other 50% being in class 2.

- Without running any model, what is your best classification strategy?
  - Assign all sample units to one of the classes, and you will at least have 50% accuracy.
  - This is called ***random guessing***, which is a common benchmark in classification.

Benchmark in classification problems

Therefore, whenever you calculated an accuracy on the validation set, you ALWAYS want to compare it with random guessing, i.e.,

- look at the distribution of class 0 and class 1 in the validation set.

```
table(validation$class)
```

- The accuracy you obtained from a model MUST be better than the accuracy obtained from random guessing.