

FIT2086 Lecture 7

Classification and logistic regression

Daniel F. Schmidt

Faculty of Information Technology, Monash University

September 6, 2022

1 Classifiers

- Directly Building Classifiers
- Logistic Regression

2 Assessing Classifiers

- How good is a classifier?

- Assignment updates
 - Assignment 1 marks being finalised
 - Assignment 2 due next week
- Last week we examined linear regression for continuous targets
- This week we extend these models to binary targets (“classification”)
- Learn how to assess the quality of predictions from such models
- Learning Outcomes:
 - produce models for predictive statistical analysis;
 - interpret results for a variety of models.

Revision from last week (1)

- Linear regression

$$\mathbb{E}[Y_i] = \beta_0 + \beta_1 x_{i,1} + \beta_2 x_{i,2} + \cdots + \beta_p x_{i,p}$$

- β_0 is the **intercept** (value of $\mathbb{E}[Y_i]$ when all predictors are zero)
- β_j is a **coefficient** (change in $\mathbb{E}[Y_i]$ per unit change in $x_{j,i}$)

- Residuals (errors)

$$e_i = y_i - \beta_0 - \beta_1 x_{i,1} - \beta_2 x_{i,2} - \cdots - \beta_p x_{i,p}$$

- Residual sum-of-squares

$$\text{RSS}(\beta_0, \beta_1, \dots, \beta_p) = \sum_{i=1}^n e_i^2$$

- Least-squares estimates linear model by finding $\beta_0, \beta_1, \dots, \beta_p$ that minimise the RSS

Revision from last week (2)

- R^2 goodness-of-fit

$$R^2 = 1 - \frac{\text{RSS}}{\text{TSS}}$$

where TSS is the sum of squared errors for the mean model

- Model fitting:
 - Overfitting = including unimportant predictors
 - Underfitting = excluding important predictors
- Hypothesis testing to determine if variable is important
 - Test $H_0 : \beta_j = 0$ vs $H_A : \beta_j \neq 0$
 - The smaller the p -value the stronger predictor j is associated with the target
- Model selection methods:
 - Add complexity penalty to the negative log-likelihood
- Finding good models – all subsets selection, stepwise selection

Today's Relevant Figure (N/A)



Bin Yu. Born in China. Studied mathematics at Peking university, and obtained a PhD from Berkeley. Has worked in the areas of genomics, machine learning and statistics with particularly contributions to model selection, complexity and the theoretical foundations of deep learning. Has recently proposed a framework for trustworthy data science.

1 Classifiers

- Directly Building Classifiers
- Logistic Regression

2 Assessing Classifiers

- How good is a classifier?

Supervised Learning – recap (1)

- In Week one and two we examined the use of random variables as a tool for modelling data
- From week three to week five we looked at parameter inference
 - In week 3 we examined **point estimation** using maximum likelihood
 - In week 4 we examined **interval estimation** using confidence intervals
 - In week 5 we examined **hypothesis testing**
- Last week we extended the normal distribution via linear regression
 - We allowed each individual to have a different mean depending on their attributes
 - We used the tools from Week 3 to 5 to learn these models
- This week we will do the same for the Bernoulli distribution

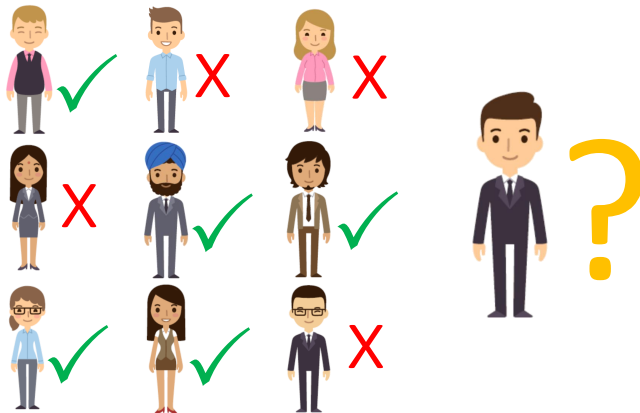
Supervised Learning – recap (2)

- Imagine we have measured $p + 1$ variables on n individuals (people, objects, things)
- We would like to predict one of the variables using the remaining p variables
- If the variable we are predicting is categorical, we are performing **classification**
 - Example: predicting if someone has diabetes from medical measurements.
- If the variable we are predicting is numerical, we are performing **regression**
 - Example: Predicting the quality of a wine from chemical and seasonal information.

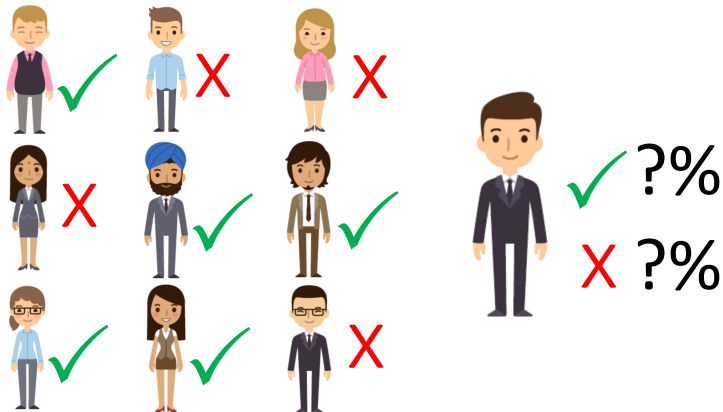
Supervised Learning – recap (3)

- The variable we are predicting is designated the “y” variable
 - We have (y_1, \dots, y_n)
- This variable is often called the:
 - target;
 - response;
 - outcome.
- The other variables are usually designated “X” variables
 - We have $(x_{i,1}, \dots, x_{i,p})$ for $i = 1, \dots, n$
- These variables are often called the
 - explanatory variables;
 - predictors;
 - covariates;
 - features;
 - exposures.
- Usually we assume the targets are random variables and the predictors are known without error

Classifiers



Probabilistic classifiers



Classification – Key Slide

- We begin by defining a classifier in terms of probability
- Imagine we have a **categorical** outcome variable Y
- We also have p predictor variables X_1, \dots, X_p
 \Rightarrow often called **features** in classification literature
- We can build a classifier for Y using our predictors, i.e., we want to find

$$\mathbb{P}(Y = y \mid X_1 = x_1, X_2 = x_2, \dots, X_p = x_p)$$

- This is the **conditional probability** of Y given X_1, \dots, X_p .
- Gives us the probability of an individual being in class $Y = y$, given the values of their predictors x_1, \dots, x_p

Classifiers (1)

- Let us now specialise our problem
- Assume that all the predictors are also **categorical**
- The formula for conditional probability is:

$$\mathbb{P}(Y = y \mid X_1 = x_1, \dots, X_p = x_p) = \frac{\mathbb{P}(Y = y, X_1 = x_1, \dots, X_p = x_p)}{\mathbb{P}(X_1 = x_1, \dots, X_p = x_p)}$$

where

- the numerator is the **joint probability** of $(Y = y, X_1 = x_1, \dots, X_p = x_p)$;
- the denominator is the **marginal probability** of $(X_1 = x_1, \dots, X_p = x_p)$.

Classifiers (2)

- So if we have the **joint probability** we can build a classifier
- Consider the following example:

	No Heart Disease ($H = 0$)	Heart Disease ($H = 1$)
No Mutation ($M = 0$)	0.35	0.30
Mutation ($M = 1$)	0.10	0.25

Population joint probabilities of heart disease/LDLR mutation.

- Then we have

$$P(H = 1 \mid M = 0) = \frac{P(H = 1, M = 0)}{P(M = 0)} = 0.4615$$

$$P(H = 1 \mid M = 1) = \frac{P(H = 1, M = 1)}{P(M = 1)} = 0.7143$$

Classifiers (3)

- In our example we got told the population joint probabilities
- But in reality we don't know these – we just have data
- We can try and estimate them from the data
- For our example:
 - our target is heart disease, $H \in \{0, 1\}$,
 - predictor is LDLR mutation, $M \in \{0, 1\}$

Classifiers (4)

- Imagine we had n realisations of this random variables, $\mathbf{m} = (m_1, \dots, m_n)$ and $\mathbf{h} = (h_1, \dots, h_n)$
- We could estimate joint probability by proportions

$$F(H = h, M = m) = \frac{1}{n} \sum_{i=1}^n I(h_i = h \text{ and } m_i = m)$$

- Weak law of large numbers guarantees this will converge on population proportions for large enough n
- We essentially did this in Assignment 1 (FC Barcelona performance)

Classifiers (5)

- Example, imagine we had
 - $\mathbf{m} = (1, 1, 0, 1, 1, 1, 0, 0)$ and
 - $\mathbf{h} = (1, 0, 1, 1, 0, 0, 1, 0)$
- Then estimated joint probabilities are

	No Heart Disease ($H = 0$)	Heart Disease ($H = 1$)
No Mutation ($M = 0$)	1/8	2/8
Mutation ($M = 1$)	3/8	2/8

Estimated joint probabilities of heart disease/LDLR mutation

- Now we can estimate $\mathbb{P}(H = h \mid M = m)$ using

$$\frac{F(H = h, M = m)}{F(H = 0, M = m) + F(H = 1, M = m)}$$

- For large n the proportions will be close to population probabilities

Classifiers (6)

- Simple enough – but there is a problem
- For our simple problem H and M were binary
⇒ only need to estimate $2 \times 2 = 4$ joint probabilities
- What if we had two binary genetic mutations, M_1 and M_2 ?
- Now have $2 \times 2 \times 2 = 8$ probabilities to estimate
- For p predictors, there are 2^{p+1} probabilities to estimate
⇒ **exponential growth** in p
- This rapidly outstrips our sample size n no matter how big n is

Classifiers (7)

- We need to constrain the problem
- Two simple approaches popular in literature
- **Naïve Bayes** classifiers
 - Make simplifying assumptions about joint probabilities
 - Easily handle categorical predictors
 - Easily handles multi-class targets
 - Popular in text mining and classification
 - We don't examine – but simple enough for you to learn yourself if interested
- **Logistic regression**
 - Adaptation of the linear model, widely used
 - Directly estimates conditional probabilities
 - Handles categorical and continuous predictors
 - More difficult to handle multi-class targets

Logistic Regression (1)

- The direct approach we examined used the joint probabilities to find the conditional probability of the targets
- This is (potentially) a round-about way of solving problem
- Logistic regression directly models the conditional probabilities
⇒ extends the **linear regression** model to binary data
- This approach also extends to other classification methods
 - Decision trees and forests (in two weeks time)
 - Support vector machines
 - Neural networks
 - And many more ...

Logistic Regression (2)

- Given predictors $x_{i,1}, \dots, x_{i,p}$ multiple linear regression predicts the target as

$$\mathbb{E}[Y_i] = \eta_i = \beta_0 + \sum_{j=1}^p \beta_j x_{i,j}$$

where η_i is shorthand for our **linear predictor** for individual i

- We find $\beta_0, \beta_1, \dots, \beta_j$ by least-squares
- If our target is binary, we *could* fit a linear model using least-squares and approximate

$$\mathbb{P}(Y_i = 1 \mid x_{i,1}, \dots, x_{i,p}) \approx \eta_i$$

- What is a potential serious problem with this approach?

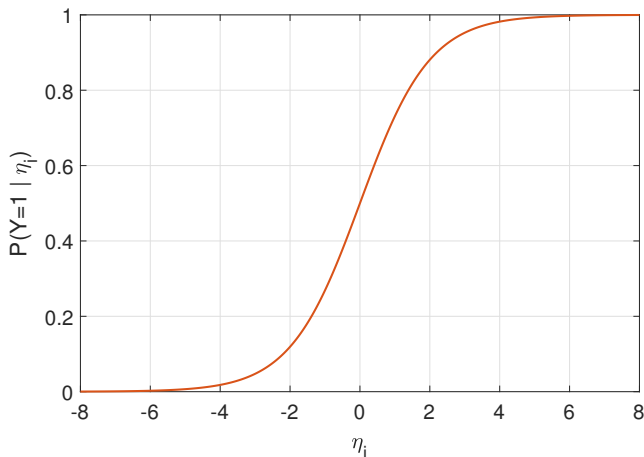
Logistic Regression (3)

- The simple approach has the problem that the predicted value η_i could be less than zero, or greater than one, for certain values of the features!
- One solution to fix this is to bound η_i to $(0,1)$
- There exist a lot of ways of bounding η_i
- Logistic regression chooses to use the **logistic function**

$$\mathbb{P}(Y_i = 1 \mid x_{i,1}, \dots, x_{i,p}) = \frac{1}{1 + \exp(-\eta_i)}$$

- This function **smoothly**
 - tends to 0 as $\eta_i \equiv \beta_0 + \sum_{j=1}^p \beta_j x_{i,j} \rightarrow -\infty$;
 - tends to 1 as $\eta_i \equiv \beta_0 + \sum_{j=1}^p \beta_j x_{i,j} \rightarrow \infty$.

Logistic Regression (4)



The logistic function. As $\eta_i \rightarrow -\infty$, then $\mathbb{P}(Y_i = 1 | \eta_i) \rightarrow 0$, and as $\eta_i \rightarrow \infty$, then $\mathbb{P}(Y_i = 1 | \eta_i) \rightarrow 1$.

Logistic Regression (5)

- We can interpret the logistic model in terms of **log-odds**
- Given $\mathbb{P}(Y = 1)$ and $\mathbb{P}(Y = 0)$, the odds for $Y = 1$ are

$$\mathbb{P}(Y = 1)/\mathbb{P}(Y = 0)$$

- They reflect how *many more times likely* the event $Y = 1$ is to occur than the event $Y = 0$
- Example: probability of a heads from coin toss is 0.75; then
 - the odds for seeing a head are $0.75/0.25 = 3$;
 - the odds for seeing a tail are $0.25/0.75 = 1/3$.
- The log-odds make this symmetric:
 - the log-odds for seeing a head are $\log(0.75/0.25) = \log 3$;
 - the log-odds for seeing a tail are $\log(0.25/0.75) = -\log 3$.

Logistic Regression (6) – Key Slide

- A logistic regression models the conditional log-odds as

$$\log \left(\frac{\mathbb{P}(Y_i = 1 \mid x_{i,1}, \dots, x_{i,p})}{\mathbb{P}(Y_i = 0 \mid x_{i,1}, \dots, x_{i,p})} \right) = \beta_0 + \sum_{j=1}^p \beta_j x_{i,j} \equiv \eta_i$$

- So the log-odds of a success, given the values of the predictors, is equal to the linear predictor η_i
 - the **intercept** β_0 is the log-odds when all the predictors are zero, i.e., $x_{i,1} = x_{i,2} = \dots = x_{i,p} = 0$;
 - the **coefficient** β_j is the increase in log-odds per unit change of predictor x_j
- The odds for $Y = 1$ are $\exp(\eta_i)$
 - when $\eta_j > 0$, $Y = 1$ is more likely than $Y = 0$, and $e^{\eta_i} > 1$
 - when $\eta_j < 0$, $Y = 0$ is more likely than $Y = 1$, and $e^{\eta_i} < 1$

Logistic Regression (7)

- To see that setting log-odds equal to η_i leads to logistic regression, write:

$$\log \left(\frac{\mathbb{P}(Y_i = 1 \mid x_{i,1}, \dots, x_{i,p})}{1 - \mathbb{P}(Y_i = 1 \mid x_{i,1}, \dots, x_{i,p})} \right) = \eta_i$$

- Now exponentiate both sides

$$\frac{\mathbb{P}(Y_i = 1 \mid x_{i,1}, \dots, x_{i,p})}{1 - \mathbb{P}(Y_i = 1 \mid x_{i,1}, \dots, x_{i,p})} = \exp(\eta_i)$$

- Solving for $\mathbb{P}(Y_i = 1 \mid \dots)$ yields

$$\mathbb{P}(Y_i = 1 \mid x_{i,1}, \dots, x_{i,p}) = \frac{\exp(\eta_i)}{1 + \exp(\eta_i)} = \frac{1}{1 + \exp(-\eta_i)}$$

(noting $1/e^a = e^{-a}$) which is the logistic function.

- **Question 1:** Imagine we have logistic regression

$$\text{LogOdds}(Y) = 0.5 + 3x$$

If $x = 0.2$, then

- what is the odds that $Y = 1$?
 - what is the probability that $Y = 1$?
-
- **Question 2:** How could we try and decide if an individual's class was 0 or 1 using our logistic regression model?
-
- **Question 3:** If we only had a matrix of predictors \mathbf{X} , and a vector of observed outcomes \mathbf{y} , how could we learn the coefficients of a logistic regression model?

- **Question 1:** Imagine we have logistic regression

$$\text{LogOdds}(Y) = 0.5 + 3x$$

If $x = 0.2$, then

- what is the odds that $Y = 1$?
 - what is the probability that $Y = 1$?
- **Question 2:** How could we try and decide if an individual's class was 0 or 1 using our logistic regression model?
- **Question 3:** If we only had a matrix of predictors \mathbf{X} , and a vector of observed outcomes \mathbf{y} , how could we learn the coefficients of a logistic regression model?

Fitting Logistic Regression Models (1)

- How to estimate the regression coefficients?
- Many packages use maximum likelihood
- Let $\beta = (\beta_1, \dots, \beta_p)$ be our regression coefficients
- Assume our targets are independent RVs;
- Each Y_i is then distributed as per a Bernoulli distribution

$$Y_i \sim \text{Be}(\theta_i(\beta_0, \beta))$$

where

$$\theta_i(\beta_0, \beta) = \frac{1}{1 + \exp\left(-\beta_0 - \sum_{j=1}^p \beta_j x_{i,j}\right)}$$

is the probability of success for individual i , given the predictors $x_{i,1}, \dots, x_{i,p}$ and the parameters $\beta_0, \beta_1, \dots, \beta_p$.

Fitting Logistic Regression Models (2)

- If $\mathbf{y} = (y_1, \dots, y_n)$ are binary targets, the likelihood for a logistic regression is then

$$\begin{aligned} p(\mathbf{y} \mid \beta_0, \boldsymbol{\beta}) &= \prod_{i=1}^n p(y_i \mid \beta_0, \boldsymbol{\beta}) \\ &= \prod_{i=1}^n \theta_i(\beta_0, \boldsymbol{\beta})^{y_i} (1 - \theta_i(\beta_0, \boldsymbol{\beta}))^{1-y_i} \end{aligned}$$

from the pdf of the Bernoulli distribution (Lecture 2).

- The negative log-likelihood is then

$$L(\mathbf{y} \mid \beta_0, \boldsymbol{\beta}) = - \sum_{i=1}^n [y_i \log \theta_i(\beta_0, \boldsymbol{\beta}) + (1 - y_i) \log (1 - \theta_i(\beta_0, \boldsymbol{\beta}))]$$

Fitting Logistic Regression Models (3)

- The negative log-likelihood is then

$$\begin{aligned} L(\mathbf{y} \mid \beta_0, \boldsymbol{\beta}) &= - \sum_{i=1}^n [y_i \log \theta_i(\beta_0, \boldsymbol{\beta}) + (1 - y_i) \log (1 - \theta_i(\beta_0, \boldsymbol{\beta}))] \\ &= \sum_{i=1}^n [-y_i \eta_i + \log (1 + e^{\eta_i})] \end{aligned}$$

where $\eta_i = \beta_0 + \sum_{j=1}^p \beta_j x_{i,j}$ is the log-odds (linear predictor)

- The values $\hat{\beta}_0$, $\hat{\boldsymbol{\beta}}$ of β_0 and $\boldsymbol{\beta}$ that minimise this quantity are the maximum likelihood estimates
- No closed form solution exists, must be found numerically
 \implies But luckily always only a single, global minimum
- Time complexity roughly cubic in number of predictors p

- The minimised negative log-likelihood

$$L(\mathbf{y} \mid \hat{\beta}_0, \hat{\beta})$$

is a measure of goodness-of-fit of a model.

- The difference in minimised negative log-likelihoods

$$L(\mathbf{y} \mid \hat{\beta}_0) - L(\mathbf{y} \mid \hat{\beta}_0, \hat{\beta})$$

is a measure similar to R^2 ; (bigger differences \Rightarrow better fit)

- Relative to model with an intercept only (no predictors); equivalent to assuming $P(Y = 1)$ is same for all individuals
- Sometimes maximised log-likelihood is reported instead, or two times (negative) log-likelihoods; depends on package

Predicting with a Logistic Regression

- Once we have found estimates $\hat{\beta}_0, \hat{\beta}$ it is easy to predict with a logistic regression
- For some new values of features x'_1, \dots, x'_p , we calculate

$$\hat{\eta} = \hat{\beta}_0 + \sum_{j=1}^p \hat{\beta}_j x'_j$$

- Then we can estimate the probability that $Y' = 1$ for these features:

$$\mathbb{P}(Y' = 1 | x'_1, \dots, x'_p) = \frac{1}{1 + \exp(-\hat{\eta})}$$

- If we need to guess at most likely class, choose value of Y' that maximises this probability

- A strength of logistic regression is that it builds on the tools used in linear regression
- Handle categorical predictors same as linear regression
⇒ form new indicator variables for each category
- They can also handle non-linearities the same way as linear regressions, e.g.,
 - logarithmic transformations of predictors;
 - polynomial transformations of predictors.

Finding logistic regression models

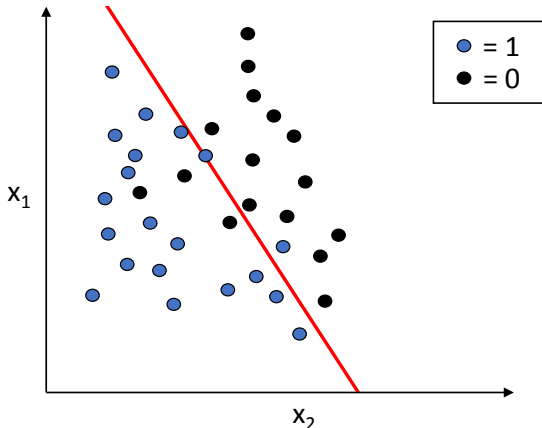
- Same approaches as for linear models
⇒ try to avoid under/over-fitting
- Hypothesis testing $H_0 : \beta_j = 0$ vs $H_A : \beta_j \neq 0$
 - Smaller p -value ⇒ more likely predictor j is important
- Model selection
 - Use penalized likelihood:

$$L(\mathbf{y} \mid \hat{\beta}_0, \hat{\beta}) + k\alpha_n$$

where:

- k is number of predictors in model;
 - $\alpha_n = 1$ for Akaike information criterion (AIC);
 - $\alpha_n = 3/2$ for Kullback information criterion (KIC);
 - $\alpha_n = (1/2) \log n$ for Bayesian information criterion (BIC).
- Sometimes two times these quantities are used (e.g., in R)
- Forward/backwards selection of predictors

Logistic regressions are linear



A logistic regression separates successes from failures by using a linear separation surface. The line is defined by the values of the two features x_1 and x_2 that satisfy $\mathbb{P}(Y = 1 \mid x_1, x_2) = 1/2$. For models with p features, this becomes a p -dimensional plane.

- 1 Classifiers
 - Directly Building Classifiers
 - Logistic Regression
- 2 Assessing Classifiers
 - How good is a classifier?

Performance Measures for Classifiers (1)

- Imagine we have trained a classifier on some data (logistic regression or something else)
- We now get a new body of data and want to test how well our classifier performs
- What measures of performance exist for classification problems?
 - Classification error
 - Sensitivity and specificity
 - Area-under-the-curve (AUC)
 - Logarithmic loss

Performance Measures for Classifiers (2)

- Let $\mathbf{y}' = (y'_1, \dots, y'_{n'})$ be a vector of new data to test on
 - For simplicity, let us assume y'_i is binary
- Let $\mathbf{x}'_j = (x'_{1,j}, \dots, x'_{n',j})$ be the vector for feature j
- For each of the new individuals, we can calculate our best guess at which class it belongs to using:

$$\hat{y}'_i = \arg \max_{y \in \{0,1\}} \left\{ \mathbb{P}(Y'_i = y \mid x'_{i,1}, \dots, x'_{i,p}) \right\}$$

where the probabilities are estimated using the model we have learned from our training data

Classification Accuracy (1) – Key Slide

- The most straightforward measure of performance is **classification accuracy**
- This is given by:

$$CA = \frac{1}{n'} \sum_{i=1}^{n'} I(y'_i = \hat{y}'_i)$$

where $I(\cdot)$ is one if the condition inside the parenthesis is met, and a zero otherwise

- The proportion of times our classifier correctly guesses the class of a new individual
- This ranges from 0 (perfectly incorrect), through to $1/2$ (only as good as random guessing) to 1 (perfectly correct)
- Realistically $1/2$ is worst accuracy – if $< 1/2$ we can swap our classification output

Classification Accuracy (2)

- More generally, we can form a **confusion matrix**

	$y_i = 0$	$y_i = 1$
$\hat{y}_i = 0$	True Negative (TN)	False Negative (FN)
$\hat{y}_i = 1$	False Positive (FP)	True Positive (TP)

- Classification accuracy is then

$$CA = \frac{TP + TN}{TP + TN + FP + FN}$$

Specificity and Sensitivity (1) – Key Slide

- Can form other useful information
- **Sensitivity** is the true positive rate:

$$\text{TPR} = \frac{\text{TP}}{\text{TP} + \text{FN}}$$

- **Specificity** is the true negative rate:

$$\text{TNR} = \frac{\text{TN}}{\text{TN} + \text{FP}}$$

- Sensitivity of 1 means we correctly classify all individuals for which $y'_i = 1$
 - Specificity of 1 means we correctly classify all individuals for which $y'_i = 0$
- High sensitivity can be achieved at expense of decreased specificity , and vice versa

Specificity and Sensitivity (2)

- Set a detection threshold $T \in (0, 1)$ for our classifier.
- For each of the new individuals, we say that $\hat{y}'_i = 1$ if

$$\mathbb{P}(Y'_i = 1 \mid x'_{i,1}, \dots, x'_{i,p}) \geq T,$$

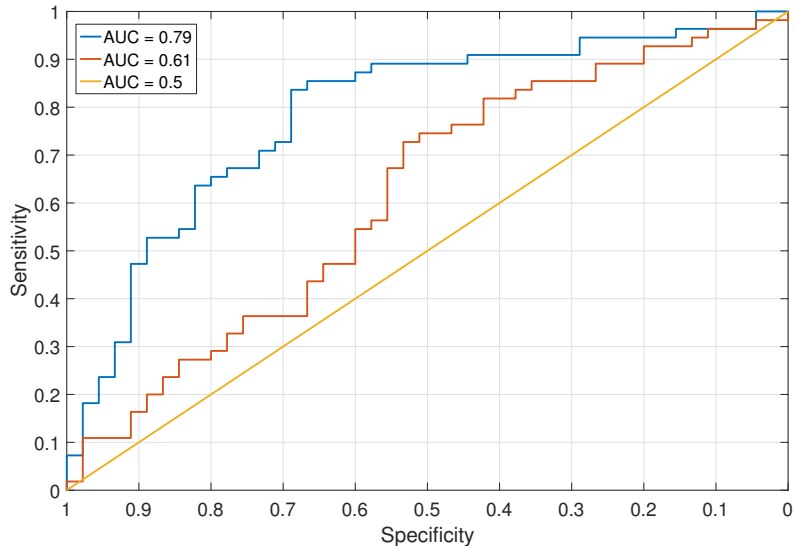
otherwise they are in class $\hat{y}'_i = 0$.

- Classifying an individual based on which class is more likely is equivalent to using $T = 1/2$
- Varying T produces different values of sensitivity, specificity
- Let $\text{TPR}(T)$, $\text{TNR}(T)$ be the sensitivity, specificity for threshold T
 - Small $T \Rightarrow$ increased sensitivity;
 - Large $T \Rightarrow$ increased specificity

Area-under-the-curve (AUC) (1)

- If we vary T from 0 to 1 we get a range of different classification rules
 \implies each will yield a different TPR and TPN
- We can plot these different values to get a “receiver operating curve”
- The area under this curve is called the **AUC**
- The bigger the area, the better the classifier

Area-under-the-curve (AUC) (2)



Area-under-the-curve (AUC) (2) – Key Slide

- AUC is always between 0 and 1
 - AUC of 1 means we can achieve perfect classification;
 - AUC of 0 means we can achieve perfect misclassification;
 - AUC of $1/2$ means we do no better than a random guess.
- How to interpret AUC?
- An AUC of p means that if we randomly sampled an individual i from our test group for whom $y'_i = 1$, and randomly sampled an individual k from our test group for whom $y'_k = 0$ then

$$\mathbb{P} \left[\mathbb{P}(Y'_i = 1 \mid x'_{i,1}, \dots, x'_{i,p}) > \mathbb{P}(Y'_k = 1 \mid x'_{k,1}, \dots, x'_{k,p}) \right] = p$$

that is, it is the probability that a random individual i sampled from class 1 will be rated more likely to be in class 1 than random individual k sampled from class 0.

Logarithmic Loss (1) – Key Slide

- The final performance measure we consider is logarithmic loss
- For each sample i in our test group, we score

$$L(y'_i) = \begin{cases} -\log \mathbb{P}(Y'_i = 1 \mid x'_{i,1}, \dots, x'_{i,p}) & \text{for } y'_i = 1 \\ -\log \mathbb{P}(Y'_i = 0 \mid x'_{i,1}, \dots, x'_{i,p}) & \text{for } y'_i = 0 \end{cases}.$$

This is the negative-log-probability of the test data point y'_i under our classification model

- The total logarithmic loss is then

$$L(\mathbf{y}') = \sum_{i=1}^n L(y'_i)$$

\implies the negative-log-likelihood of this new, future data

- Smaller the score, the better our classifier predicts this data
- Log-loss measures how well the model predicts the **probabilities** of an individual being in a class (calibration)

Logarithmic Loss (2)

- Classification accuracy measures how good our guesses at the most likely class are
- Log-loss measures how well the model predicts the **probabilities** of an individual being in a class
- Why is this important?
- It tells you how confident you should be in your predicted class
- **Example:** Both $P(Y = 1 | x_1, \dots, x_p) = 0.501$ and $P(Y = 1 | x_1, \dots, x_p) = 0.99$ would predict the most likely class for Y to be $Y = 1$.
 \implies we are much more confident about latter than the former
- Estimating conditional probabilities well lets us get better idea of how confident we should be in our predicted classes

- Terms you should know:
 - Odds, log-odds
 - Logistic regression
 - Classification accuracy
 - Specificity, sensitivity
 - Area-under-the-curve (AUC)
 - Logarithmic loss
- Next week we will be examined some more recent developments in fitting and estimating linear and logistic regression models.