Lecture 5: Components of Classification

UNIVERSITY OF SAN FRANCISCO

James D. Wilson

MATH 373

Plan for this Lecture

- The classification problem
- Why not regression?
- Assessing model accuracy
  - Mean squared error and accuracy
  - Receiver Operating Curves (ROCs)

Reference: ISL Sections 2.2.3; 4.1; 4.2; 4.4.3

The Classification Setting

Data: Consisting of n observations $(x_1, y_1), \ldots, (x_n, y_n)$ with

- $x_i \in \mathcal{X}$ space of predictors (often $\subseteq \mathbb{R}^p$)
- $y_i \in \mathcal{C}$: response or class label
  - Binary classification: $\mathcal{C} = \{-1, +1\}$ (or equivalently, $\{0, 1\}$)
  - Multi-class classification: $\mathcal{C} = \{0, 1, \ldots, m\}$

Unlike regression, the observed labels are *categorical* or *qualitative*.

Classification

Goal: Given an unlabeled vector x, assign it to class $c \in \mathcal{C}$.

Prediction Rule / Classifier

A prediction rule or classifier is a map

$$\phi : \mathcal{X} \to \mathcal{C}$$

$$\phi(x) = c \in \mathcal{C}$$

Regard $\phi(x) = c \in \mathcal{C}$ as a prediction of the class label associated with the predictor x.

Motivation:

- Predictors readily available: relatively inexpensive and/or fast to obtain
- Response not readily available: relatively expensive and/or slow to obtain
- Understanding and modeling the relationship between the predictors and the response is of scientific interest.

Examples

Medical Tests:

- $x \in \mathbb{R}^p$ contains the (numerical) results of p diagnostic tests
- y = illness / condition

Object Recognition:

- $x \in \mathbb{R}^p$ contains the pixel intensities from a satellite image
- $y = +1$ if image contains a man-made object, $y = -1$ otherwise

Examples

Automatic Spam Recognition:

- x = vector of features extracted from text of email, e.g.,
  - presence of keywords (“cheap”, “cash”, “medicine”)
  - presence of key phrases (“Dear Sir/Madam”)
  - use of words in all-caps (“VIAGRA”)
  - point of origin of email
- $y = +1$ if email is spam, $y = -1$ otherwise

Credit Card Default

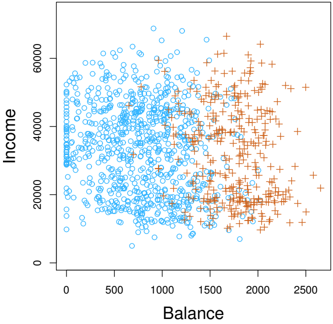


Figure: The annual incomes and monthly credit card balances of a group of individuals. Orange: defaulted on credit card payments; Blue: did not default.

Key Considerations

1. Why not use regression?
2. Measuring the loss/error of a prediction
3. Assessing the overall performance of a prediction rule
4. Identifying the optimal prediction rule

Why Not Use Regression?

Consider a simple example where Doctors are trying to predict the medical condition of a patient. Here,

$$y = \begin{cases} 1 & \text{if stroke} \\ 2 & \text{if drug overdose} \\ 3 & \text{if epileptic seizure} \end{cases}$$

- Regression assumes that there is a meaning behind the *ordering* of y and that a change in levels above suggest the *same* change.
- Typically, however, categorical variables have no natural order and there is no way to quantify a "jump" from one level to another.

Why Not Use Regression?

Consider a simple example where Doctors are trying to predict the medical condition of a patient. Here,

$$y = \begin{cases} 1 & \text{if stroke} \\ 2 & \text{if drug overdose} \\ 3 & \text{if epileptic seizure} \end{cases}$$

- Regression models y *directly* – therefore, estimates will be continuous values in $(-\infty, \infty)$
- Prediction rules are often concerned with *the probability* of each value of y

Measuring the Loss of a Prediction

Let $\phi : \mathcal{X} \to \mathcal{C}$ be a prediction/classification rule of interest

Question: Given a pair (x, y), how do we compare $\phi(x)$ and y? Namely, how do we measure the accuracy of $\phi(x)$?

Common to use the Zero-One Loss Function $\ell(\phi(x), y)$:

$$\ell(\phi(x), y) = \begin{cases} 1 & \text{if } \phi(x) \neq y \\ 0 & \text{if } \phi(x) = y \end{cases}$$

Note: Two types of errors $\phi(x) = 1, y = 0$ and $\phi(x) = 0, y = 1$ given equal weight

Expected Loss

Given: Zero-one loss of prediction rule $\phi : \mathcal{X} \to \mathcal{C}$ given by

$$\ell(\phi(x), y) = \mathbb{I}(\phi(x) \neq y)$$

We typically measure performance of ϕ by its expected loss (risk)

$$R(\phi) = \mathbb{E}[\ell(\phi(x), y)]$$

Important: Note that

$$R(\phi) = \mathbb{E}[\mathbb{I}(\phi(x) \neq y)] = \mathbb{P}(\phi(x) \neq y)$$

is just the probability that ϕ misclassifies a sample.

Measuring Accuracy

Accuracy

The accuracy of a classifier $\phi(x)$ is:

$$1 - R(\phi) = \mathbb{P}(\phi(x) = y)$$

Important Notes:

- In practice, we measure the empirical probability of misclassification over a data set with n observations using:

$$\frac{1}{n}\sum_{i=1}^{n}\mathbb{I}(y_i \neq \phi(x_i))$$

- If $y \in \{0, 1\}$, the empirical misclassification rate = MSE(ϕ).
- Training and test set evaluations still apply!

Issues with Measuring Accuracy Only

Example:

Paypal claims that its fraud rate is less than 0.5%. Suppose that you are hired to create a classifier that distinguishes fraudulent transactions from non-fraudulent transactions. How might you classify new transactions?

Issues with Measuring Accuracy Only

Example:

Paypal claims that its fraud rate is less than 0.5%. Suppose that you are hired to create a classifier that distinguishes fraudulent transactions from non-fraudulent transactions. How might you classify new transactions?

Let $y_i = -1$ if the transaction is fraudulent and $y_i = +1$ otherwise. A great classifier (perhaps the best) according to MSE / accuracy is choosing $\phi(x_i) = +1$ for all i. Indeed, your MSE would be ~ 0.005.

Result: You never detect any of the fraudulent transactions!

The above is a typical example of unbalanced data.

Informative Model Assessment

Let $y_i \in \{-1, +1\}$ (binary classification). ϕ = proposed classifier.

- True positives (TP):

$$\sum_{i=1}^{n} \mathbb{I}(y_i = \phi(x_i) = +1)$$

- False positives (FP):

$$\sum_{i=1}^{n} \mathbb{I}(y_i = -1; \phi(x_i) = +1)$$

- True negatives (TN):

$$\sum_{i=1}^{n} \mathbb{I}(y_i = \phi(x_i) = -1)$$

- False negatives (FN):

$$\sum_{i=1}^{n} \mathbb{I}(y_i = +1; \phi(x_i) = -1)$$

Model Assessment Relationships

- Accuracy $= \dfrac{\text{TP} + \text{TN}}{n} \in [0, 1]$
- The sensitivity (or recall) of ϕ is:

$$\frac{\text{TP}}{\text{TP} + \text{FN}} = \frac{\text{TP}}{\sum_{i=1}^{n} \mathbb{I}(y_i = +1)} \in [0, 1]$$

- The specificity of ϕ is:

$$\frac{\text{TN}}{\text{TN} + \text{FP}} = \frac{\text{TN}}{\sum_{i=1}^{n} \mathbb{I}(y_i = -1)} \in [0, 1]$$

- The precision of ϕ is:

$$\frac{\text{TP}}{\text{TP} + \text{FP}} = \frac{\text{TP}}{\sum_{i=1}^{n} \mathbb{I}(\phi(x_i) = +1)} \in [0, 1]$$

Model Assessment

To understand the performance of a classifier, we can use a confusion matrix which portrays the FN, TN, FP, TP rates.

		True condition	
	Total population	Condition positive	Condition negative
Predicted condition	Predicted condition positive	True positive	False positive (Type I error)
	Predicted condition negative	False negative (Type II error)	True negative

Figure: From Wikipedia.org

Model Choice

Choice of Model: depends on the context and constraints

Back to the Paypal problem: Suppose there are 100K transactions

	$y_i = +1$	$y_i = -1$
$\phi(x_i) = +1$	99500	500
$\phi(x_i) = -1$	0	0

Summary: TN = FN = 0; TP = 99500; FP = 500

Accuracy = precision = 0.995; sensitivity = 1; specificity = 0

Result: If we are concerned with identifying fraud, we want specificity to be close to 1. In this case, our model performs terribly.

Decision Regions and Decision Boundary

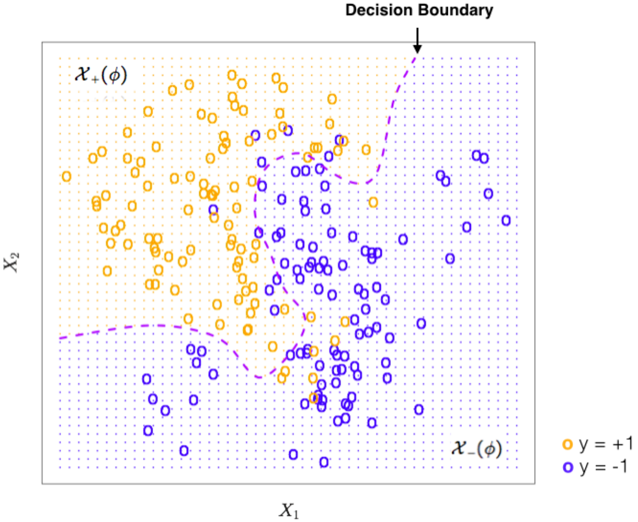
Every decision rule $\phi : \mathcal{X} \to \{-1, +1\}$ partitions the predictor space into two sets called decision regions

$$\begin{aligned} \mathcal{X}_+(\phi) &= \{x \in \ : \phi(x) = +1\} \\ &= \text{points } x \text{ assigned by } \phi \text{ to } +1 \end{aligned}$$

$$\begin{aligned} \mathcal{X}_-(\phi) &= \{x \in \ : \phi(x) = -1\} \\ &= \text{points } x \text{ assigned by } \phi \text{ to } -1 \end{aligned}$$

The boundary between $\mathcal{X}_-(\phi)$ and $\mathcal{X}_+(\phi)$ is called the decision boundary of ϕ.

Decision Regions and Decision Boundary



Idea:

- Regard given sample $(x_1, y_1), \ldots, (x_n, y_n) \in \mathcal{X} \times \{-1, +1\}$ as a set of labeled points in , with x_i having label y_i.
- Look for a simple prediction rule (equivalently, a partition of into two sets) that separates the -1s from the $+1$s.
- This idea extends to multi-class classification as well. In this case, we'll need multiple decision regions.

Next Up

- Classification Algorithms
  - k Nearest Neighbors
  - Bayes Classifiers
  - Linear Discriminant Analysis
- Logistic Regression
- Comparison of Classification Methods