Classification

Big Data y Machine Learning para Economía Aplicada

Ignacio Sarmiento-Barbieri

Universidad de los Andes

Agenda

Recap

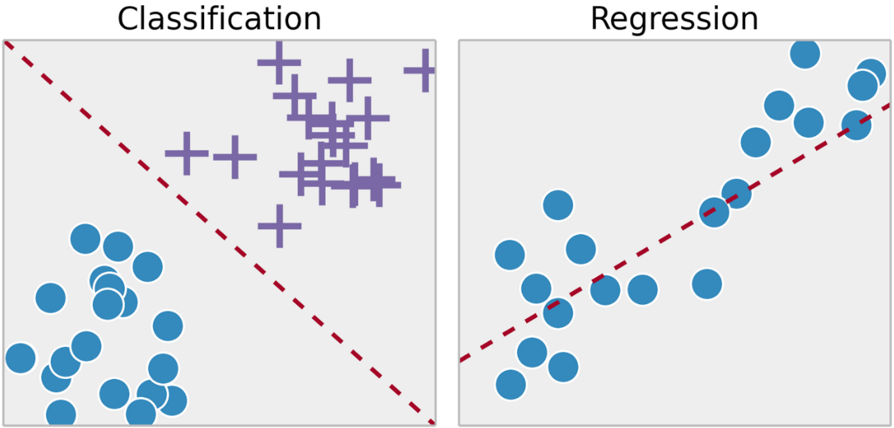
- Queremos predecir y en funcion de observables ($\mathbf{x}$)

$$y = f(\mathbf{x}) + u \tag{1}$$

- donde la estimación de f implica la que minimize el riesgo (prediga mejor fuera de muestra):

$$\hat{f} = \underset{f}{\text{argmin}} \left\{ E\left[L(y, f(\mathbf{x}; \Theta)) \right] \right\} \tag{2}$$

Classification



Classification: Motivation

- Many predictive questions are about classification
  - Email should go to the spam folder or not
  - A household is bellow the poverty line
  - Accept someone to a graduate program or no

Classification: Motivation

- Main difference is that y represents membership in a category: $y \in \{1, 2, ..., n\}$
  - Qualitative (e.g., spam, personal, social)
  - Not necessarily ordered

The prediction question is, given a new X,
what is our best guess at the response category $\hat{y}$

Agenda

Risk, Probability, and Classification

- Two states of nature $Y \to i \in \{0,1\}$
- Two actions $(\hat{Y}) \to j \in \{0,1\}$

		$\hat{Y}$ 0	$\hat{Y}$ 1
Y	0	True Negative	False Positive
	1	False Negative	True Positive

Risk, Probability, and Classification

- Two actions $\hat{Y} \rightarrow j \in \{0, 1\}$
- Two states of nature $Y \rightarrow i \in \{0, 1\}$
- Probabilities
  - $p = Pr(Y = 1|X)$
  - $1 - p = Pr(Y = 0|X)$

Risk, Probability, and Classification

- Actions have costs associated to them
- Loss: $L(i,j)$, penalizes being in bin i,j
  - We define $L(i,j)$

$$L(i,j) = \begin{cases} 1 & i \neq j \\ 0 & i = j \end{cases} \tag{3}$$

- Risk: expected loss of taking action j

Bayes classifier

$$R(1) \quad < R(0) \tag{4}$$

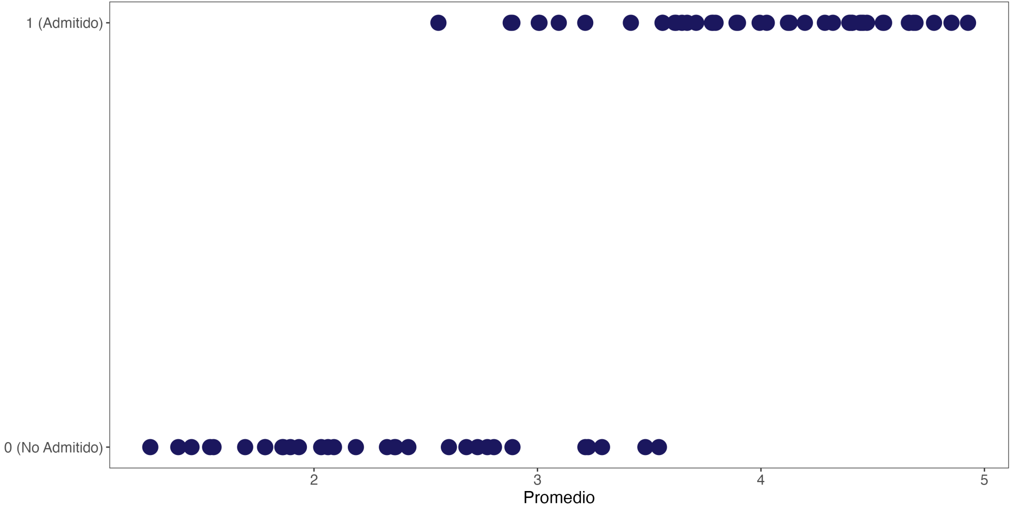
Agenda

Agenda

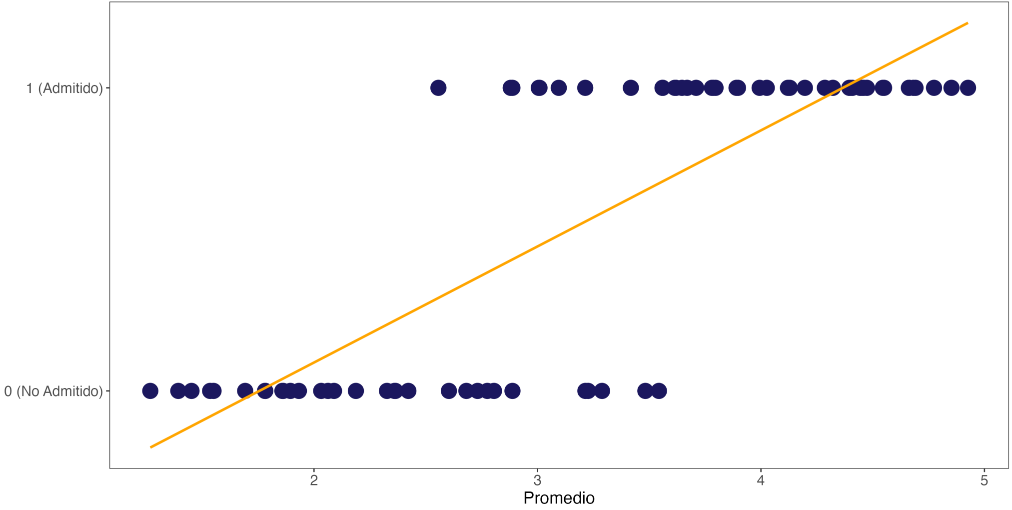
Setup

- Y is a binary random variable$\{0, 1\}$
- X is a vector of K predictors
- $p = Pr(Y = 1|X)$

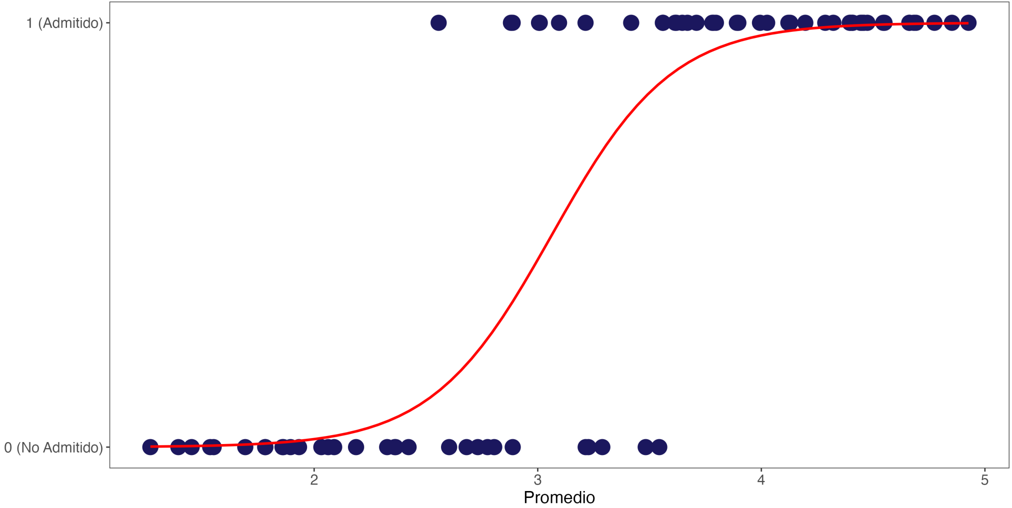
Logit



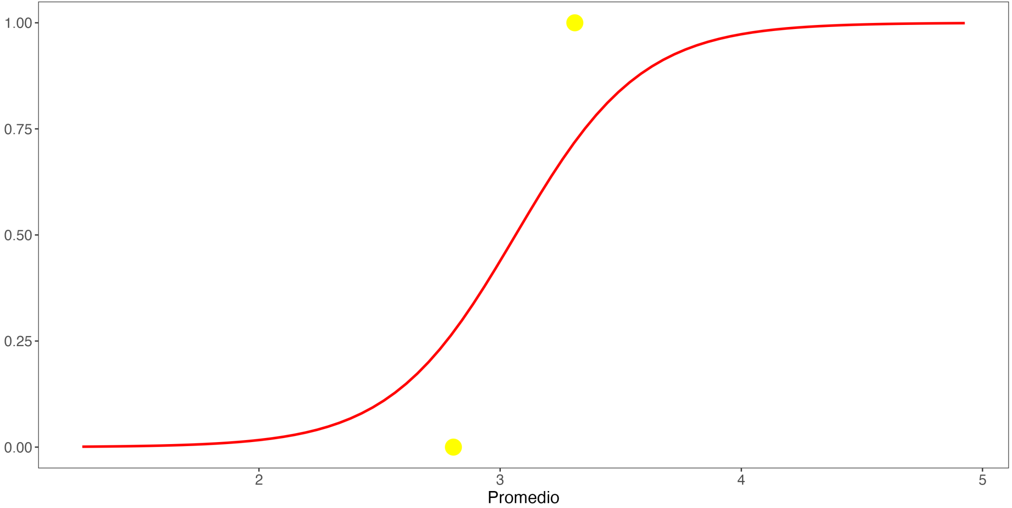
Logit



Logit



Logit



Logit

- Logit

$$p = \frac{e^{X\beta}}{1 + e^{X\beta}} \tag{5}$$

- Odds ratio

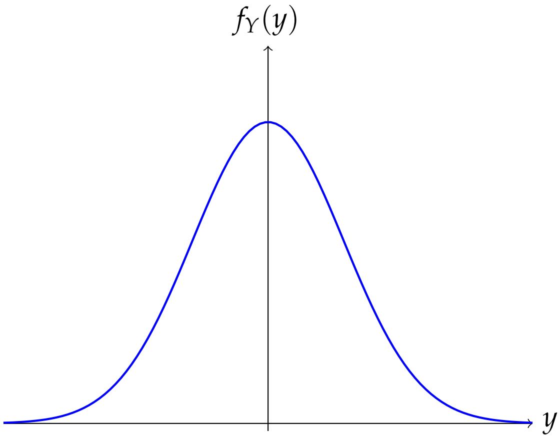
$$ln\left(\frac{p}{1-p}\right) = X\beta \tag{6}$$

Agenda

Aside: Maximum Likelihood Estimation

- Developed by Ronald A. Fisher (1890-1962)
- "If Fisher had lived in the era of "apps," maximum likelihood estimation might have made him a billionaire" (Efron and Tibshiriani, 2016)
- Why? MLE gives "automatically"
  - Consistent
  - Asymptotically normal
  - Asymptotically efficient

Aside: Maximum Likelihood Estimation



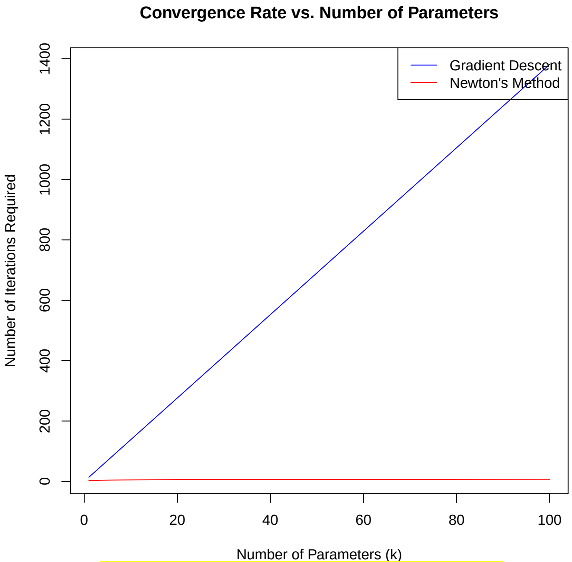
MLE Logit

- Imagine that we have a sample of iid observations $(y_i, x_i); i = 1, \ldots, n$, where $y_i \in \{0, 1\}$

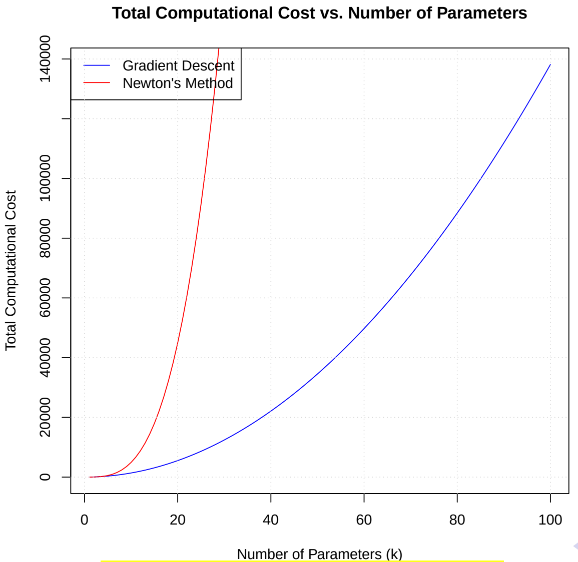
Agenda

Gradient Descent vs Newton's Method

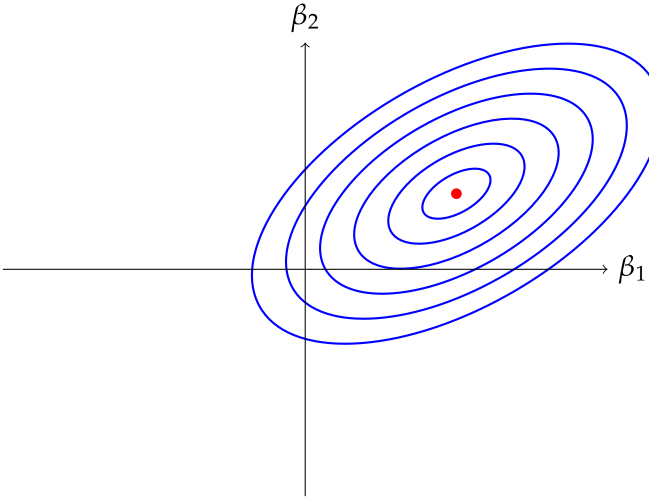
Gradient Descent vs Newton's Method



Gradient Descent vs Newton’s Method



Gradient Descent vs Newton's Method



Summary

- We observe (y_i, X_i) $i = 1, \ldots, n$
- Generate probabilities
  - Logit (example)

$$p_i = \frac{e^{X_i\beta}}{1 + e^{X_i\beta}} \tag{7}$$

  - Predict

$$\hat{p}_i = \frac{e^{X_i\hat{\beta}}}{1 + e^{X_i\hat{\beta}}} \tag{8}$$

- Classification

$$\hat{Y}_i = 1[\hat{p}_i > 0.5] \tag{9}$$

Accuracy

$$
\begin{array}{cccc}
 & & \hat{y}_i & \\
 & & 0 & 1 \\
\multirow{2}{*}{y_i} & 0 & \text{TN} & \text{FP} \\
 & 1 & \text{FN} & \text{TP}
\end{array}
$$

$$\frac{TP + TN}{TP + TN + FN + FP} \tag{10}$$

Example

photo from https://www.dailydot.com/parsec/batman-1966-labels-tumblr-twitter-vine/

Agenda

Probit

- $Pr(y = 1|X) = \Phi(X'\beta)$ where Φ is the standard normal cdf.
- In practice, the probit and logit models generally yield very similar predicted probabilities,
- There are practical reasons for favoring one or the other in some cases for mathematical convenience, in other computational convenience, but it is difficult to justify the choice of one distribution or another on theoretical grounds.

Probit

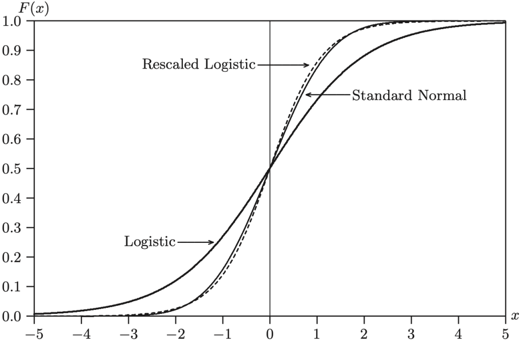


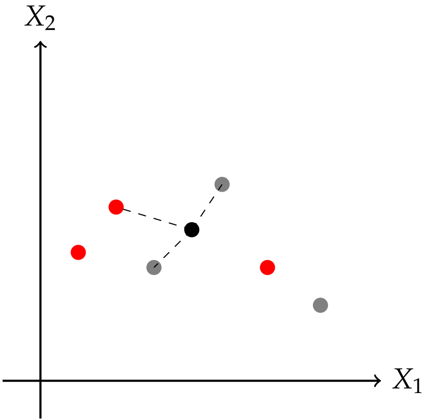
Figure 11.1 Alternative choices for $F(x)$

Agenda

Agenda

K-Nearest Neighbors

- K nearest neighbor (K-NN) algorithm predicts class $\hat{y}$ for x by asking
  What is the most common class for observations around x?



K-Nearest Neighbors

- K nearest neighbor (K-NN) algorithm predicts class $\hat{y}$ for x by asking *What is the most common class for observations around x?*
- Algorithm: given an input vector x_f where you would like to predict the class label
  - Find the K nearest neighbors in the dataset of labeled observations, $\{x_i, y_i\}_{i=1}^n$, the most common distance is the Euclidean distance:

$$d(x_i, x_f) = \sqrt{\sum_{j=1}^{p} (x_{ij} - x_{fj})^2} \tag{11}$$

  - This yields a set of the K nearest observations with labels:

$$[x_{i1}, y_{i1}], \ldots, [x_{iK}, y_{iK}] \tag{12}$$

  - The predicted class of x_f is the most common class in this set

$$\hat{y}_f = mode\{y_{i1}, \ldots, y_{iK}\} \tag{13}$$

Agenda

Agenda

Linear Discriminant Analysis

Reverend Bayes to the rescue: Bayes Theorem

$$Pr(Y = 1|X) \tag{14}$$

Example: Default

photo from https://www.dailydot.com/parsec/batman-1966-labels-tumblr-twitter-vine/

Linear Discriminant Analysis

- Why is it called linear?
- One predictor with $\sigma_0 = \sigma_1$ (equal variance)

Quadratic Discriminant Analysis

- QDA assumes diferent variances for the components

Agenda

Naive Bayes

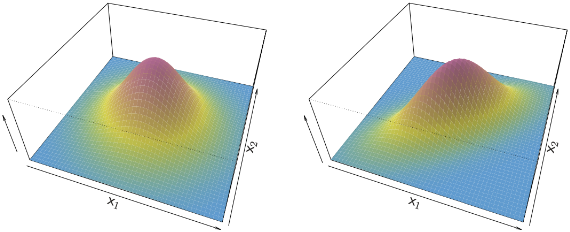
$$Pr(Y = 1|X) = \frac{f(X|Y = 1)\pi(Y = 1)}{f(X|Y = 1)\pi(Y = 1) + f(X|Y = 0)(1 - \pi(Y = 1))} \tag{15}$$

- $\pi(Y = 1)$
- $f(X|Y = 1)$

Naive Bayes

- NB assumes independence

$$f(X|Y=1) = f(x_1|Y=1) \times \cdots \times f(x_k|Y=1) \tag{16}$$



Naive Bayes

- NB assumes independence

$$f(X|Y=1) = f(x_1|Y=1) \times \cdots \times f(x_k|Y=1) \tag{17}$$