S&DS 265 / 565

Introductory Machine Learning

Classification

September 16

Yale

Notes

- Assn 1 posted; due 1 week from today (midnight)
- Please join us at office hours!
- Use Ed Discussion
- Some notes at an appropriate level:
  - Background concepts:
    `http://www.mit.edu/~6.s085/notes/lecture1.pdf`
  - Linear regression:
    `http://www.mit.edu/~6.s085/notes/lecture3.pdf`

Outline—Next two classes

- Some important concepts
- Logistic regression
- Generative vs. discriminative
- Gaussian discriminant analysis
- Examples: Supernovae and political blogs
- Regularization
- Algorithms for fitting the models

Working example: Fisher's Iris data

Outline—today

- Some important concepts
- Logistic regression
- Examples in Jupyter: Mushrooms and flowers

Classification tasks

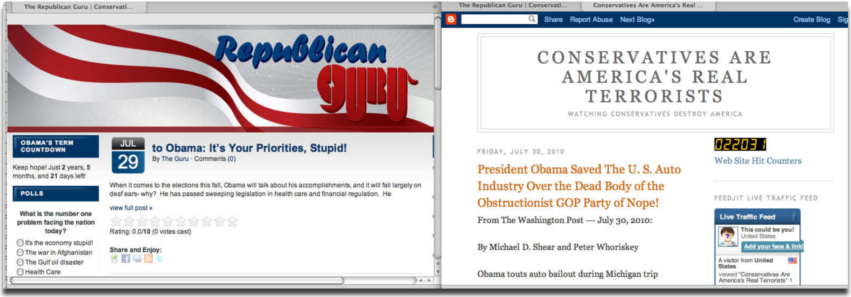
- The Coronary Risk-Factor Study (CORIS). Data: 462 males between ages of 15 and 64 from three rural areas in South Africa.

  Outcome Y is presence ($Y = 1$) or absence ($Y = 0$) of coronary heart disease

  9 covariates: systolic blood pressure, cumulative tobacco (kg), ldl (low density lipoprotein cholesterol), adiposity, famhist (family history of heart disease), typea (type-A behavior), obesity, alcohol (current alcohol consumption), and age.

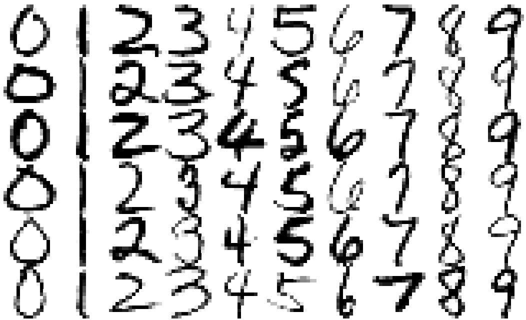
Classification tasks

- Political Blog Classification. A collection of 403 political blogs were collected during two months before the 2004 presidential election. The goal is to predict whether a blog is *liberal* ($Y = 0$) or *conservative* ($Y = 1$) given the content of the blog.



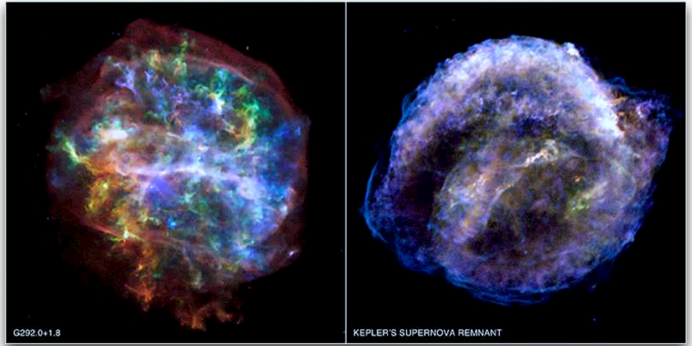
Classification tasks

- Handwriting Digit Recognition. Here each Y is one of the ten digits from 0 to 9. There are 256 covariates $X_1, \ldots, X_{256}$ corresponding to the intensity values of the pixels in a 16×16 image.



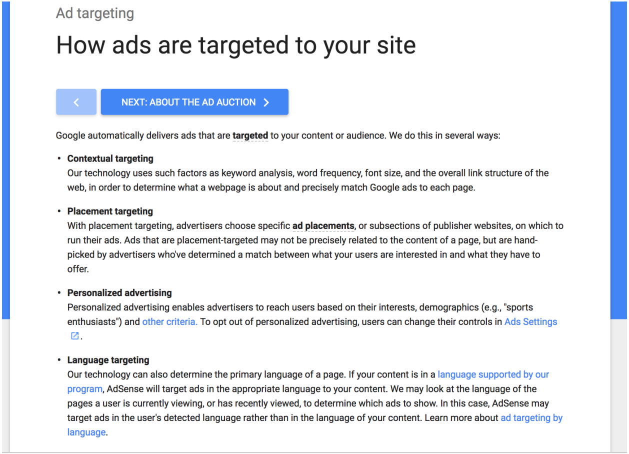
Classification tasks

- A supernova is an exploding star. Type Ia supernovae are a special class of supernovae that are very useful in astrophysics research. These supernovae have a characteristic *light curve*, which is a plot of the luminosity of the supernova versus time.



Classification tasks

- Ad click-through prediction. Predict whether or not a user will click on an ad presented. Used for ranking ads and setting prices.


Ad targeting

How ads are targeted to your site

NEXT: ABOUT THE AD AUCTION

Google automatically delivers ads that are **targeted** to your content or audience. We do this in several ways:

- **Contextual targeting**
  Our technology uses such factors as keyword analysis, word frequency, font size, and the overall link structure of the web, in order to determine what a webpage is about and precisely match Google ads to each page.
- **Placement targeting**
  With placement targeting, advertisers choose specific **ad placements**, or subsections of publisher websites, on which to run their ads. Ads that are placement-targeted may not be precisely related to the content of a page, but are hand-picked by advertisers who've determined a match between what your users are interested in and what they have to offer.
- **Personalized advertising**
  Personalized advertising enables advertisers to reach users based on their interests, demographics (e.g., "sports enthusiasts") and other criteria. To opt out of personalized advertising, users can change their controls in Ads Settings.
- **Language targeting**
  Our technology can also determine the primary language of a page. If your content is in a language supported by our program, AdSense will target ads in the appropriate language to your content. We may look at the language of the pages a user is currently viewing, or has recently viewed, to determine which ads to show. In this case, AdSense may target ads in the user's detected language rather than in the language of your content. Learn more about ad targeting by language.

Classification tasks

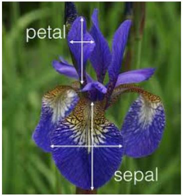
- The Iris Flower study. The data are 50 samples from each of three species of Iris flowers, *Iris setosa*, *Iris virginica* and *Iris versicolor* The length and width of the sepal and petal are measured for each specimen, and the task is to predict the species of a new Iris flower based on these features.
- App for wildflowers



Iris setosa (Left), *Iris versicolor* (Middle), and *Iris virginica* (Right).

Fisher's iris classification

Iris setosa (Left), *Iris versicolor* (Middle), and *Iris virginica* (Right).



Important concepts

Binary classifier h: function from $\mathcal{X}$ to $\{0, 1\}$.

Linear if exists a function $H(x) = \beta_0 + \beta^T x$ such that $h(x) = 1$ if $H(x) > 0$; 0 otherwise.

$H(x)$ also called a *linear discriminant function*. Decision boundary: set $\left\{x \in \mathbb{R}^d \;:\; H(x) = 0\right\}$

Classification risk, or *error rate*, of h:

$$R(h) = \mathbb{P}\big(Y \neq h(X)\big)$$

and the *empirical classification error* or *training error* is

$$\widehat{R}(h) = \frac{1}{n} \sum_{i=1}^{n} I\big(h(x_i) \neq y_i\big).$$

Optimal classification rule

Theorem. The rule h that minimizes $R(h)$ is

$$h^*(x) = \begin{cases} 1 & \text{if } m(x) > \frac{1}{2} \\ 0 & \text{otherwise} \end{cases}$$

where $m(x) = \mathbb{E}(Y \mid X = x) = \mathbb{P}(Y = 1 \mid X = x)$ denotes the regression function.

The rule h^* is called the *Bayes rule*. The risk $R^* = R(h^*)$ of the Bayes rule is called the *Bayes risk*. The set $\{x \in \mathcal{X} : \; m(x) = 1/2\}$ is called the *Bayes decision boundary.*

The Bayes decision rule

From Bayes' theorem

$$\begin{aligned}\mathbb{P}(Y=1 \mid X=x) &= \frac{p(x \mid Y=1)\mathbb{P}(Y=1)}{p(x \mid Y=1)\mathbb{P}(Y=1)+p(x \mid Y=0)\mathbb{P}(Y=0)} \\ &= \frac{\pi_1 p_1(x)}{\pi_1 p_1(x)+(1-\pi_1)p_0(x)}.\end{aligned}$$

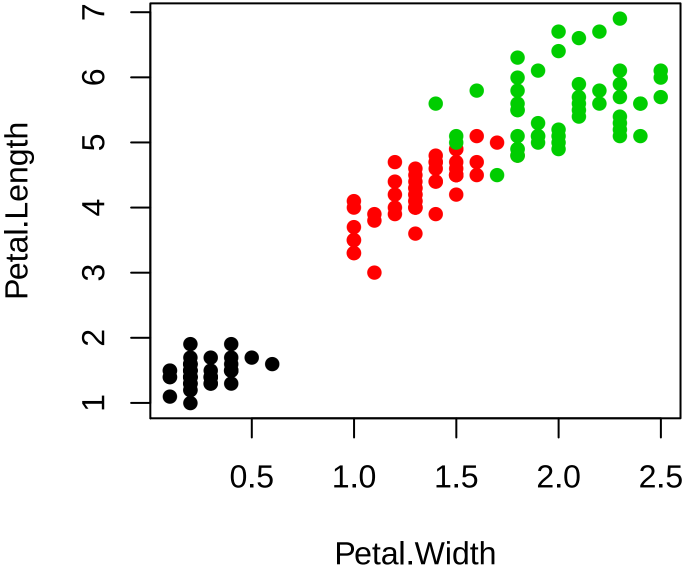
where $\pi_1 = \mathbb{P}(Y=1)$. So,

$$m(x) > \frac{1}{2} \quad \text{is equivalent to} \quad \frac{p_1(x)}{p_0(x)} > \frac{1-\pi_1}{\pi_1}.$$

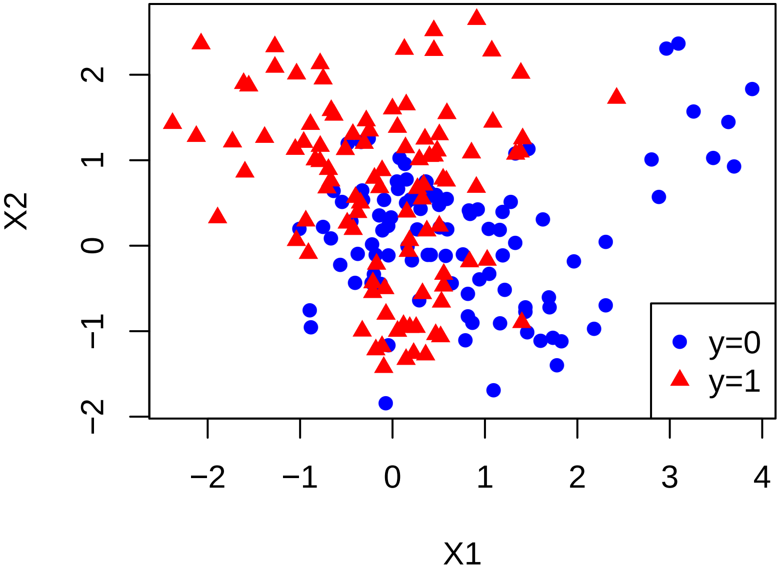
Thus the Bayes rule can be rewritten as

$$h^*(x) = \begin{cases} 1 & \text{if } \dfrac{p_1(x)}{p_0(x)} > \dfrac{1-\pi_1}{\pi_1} \\ 0 & \text{otherwise.} \end{cases}$$

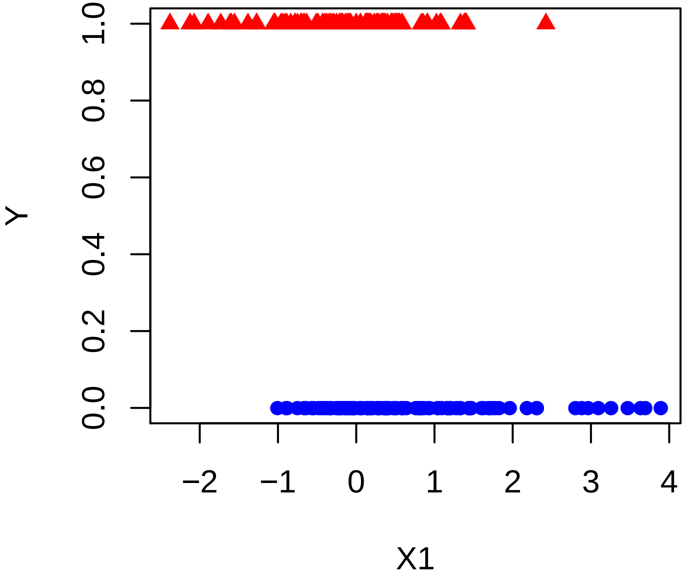
Small dataset example



Simulated data–two classes



Simplification–one predictor



Logistic regression

Conditional probabilities of the class:

$$\mathbb{P}(Y_i = 1 \,|\, X = x_i) = p(x_i)$$

$$\mathbb{P}(Y_i = 0 \,|\, X = x_i) = 1 - p(x_i)$$

Logistic regression

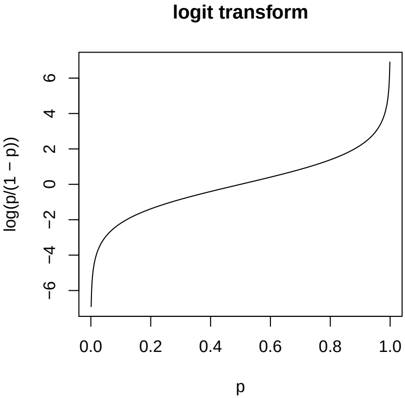
Conditional probabilities of the class:

$$\mathbb{P}(Y_i = 1 \mid X = x_i) = p(x_i)$$

$$\mathbb{P}(Y_i = 0 \mid X = x_i) = 1 - p(x_i)$$

We model the relationship between $p(x_i)$ and x_i.

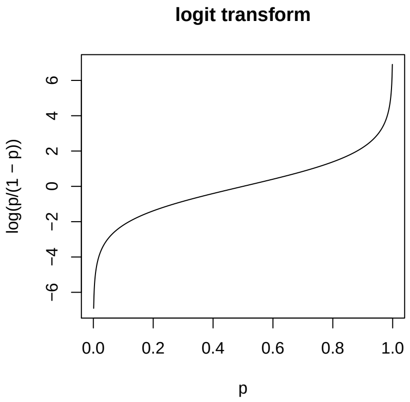
Logistic regression



The *logit* transform:

$$logit(p) = \log\left(\frac{p}{1-p}\right)$$

Logistic regression



The *logit* transform:

$$logit(p) = \log\left(\frac{p}{1-p}\right)$$

The logit transform

- is monotone
- maps the interval $[0, 1]$ to $(-\infty, \infty)$

Logistic regression

Logistic regression is a linear regression model of the log odds:

$$logit(\widehat{p}(x)) = \widehat{\beta}_0 + \widehat{\beta}_1 x$$

- p is a probability.
- $\frac{p}{1-p}$ is odds.
- $logit(p) = \log\left(\frac{p}{1-p}\right)$ is (natural) log odds.

Logistic regression

Logistic regression is a linear regression model of the log odds:

$$logit(\widehat{p}(x)) = \widehat{\beta}_0 + \widehat{\beta}_1 x$$

- p is a probability.
- $\frac{p}{1-p}$ is odds.
- $logit(p) = \log\left(\frac{p}{1-p}\right)$ is (natural) log odds.

Equivalent formulation:

$$\widehat{p}(x) = \frac{e^{\widehat{\beta}_0+\widehat{\beta}_1 x}}{1 + e^{\widehat{\beta}_0+\widehat{\beta}_1 x}} = logistic(x^T\widehat{\beta}) \equiv softmax(x^T\widehat{\beta})$$

LR decision boundary is linear

- When $\widehat{\beta}_0 + \widehat{\beta}_1 x = 0$, $\frac{\widehat{p}}{1-\widehat{p}} = 1$, so $\widehat{p} = 0.5$.

LR decision boundary is linear

- When $\widehat{\beta}_0 + \widehat{\beta}_1 x = 0$, $\frac{\widehat{p}}{1-\widehat{p}} = 1$, so $\widehat{p} = 0.5$.
- If our goal is to minimize the overall training error rate, then we use the rule:

$$\widehat{y} = \begin{cases} 1 & \widehat{p} \geq 0.5 \\ 0 & \widehat{p} < 0.5 \end{cases}$$

LR decision boundary is linear

- When $\widehat{\beta}_0 + \widehat{\beta}_1 x = 0$, $\frac{\widehat{p}}{1-\widehat{p}} = 1$, so $\widehat{p} = 0.5$.
- If our goal is to minimize the overall training error rate, then we use the rule:
$$\widehat{y} = \begin{cases} 1 & \widehat{p} \geq 0.5 \\ 0 & \widehat{p} < 0.5 \end{cases}$$
- Hence, the decision boundary is given by $\{x : x^T \widehat{\beta} = 0\}$.

LR decision boundary is linear

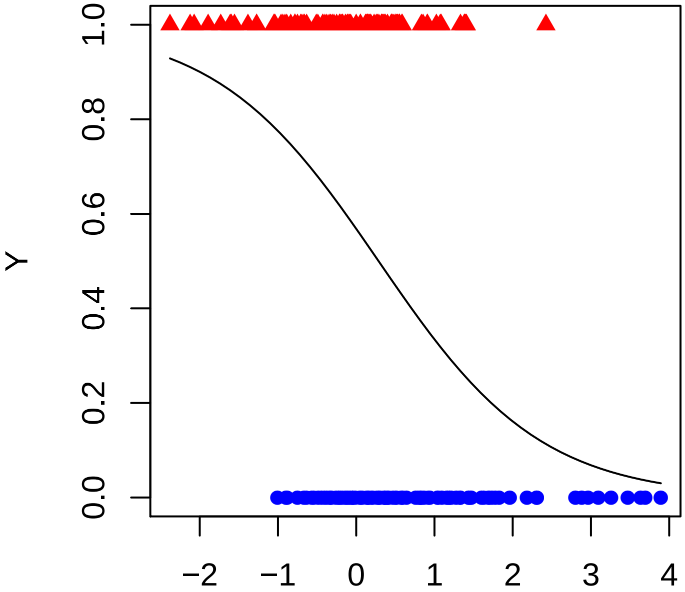
- When $\widehat{\beta}_0 + \widehat{\beta}_1 x = 0$, $\frac{\widehat{p}}{1-\widehat{p}} = 1$, so $\widehat{p} = 0.5$.
- If our goal is to minimize the overall training error rate, then we use the rule:

$$\widehat{y} = \begin{cases} 1 & \widehat{p} \geq 0.5 \\ 0 & \widehat{p} < 0.5 \end{cases}$$

- Hence, the decision boundary is given by $\{x : x^T \widehat{\beta} = 0\}$.

The decision boundary is linear in x!

Simulated data



Fitting a logistic regression

Traditionally, use maximum likelihood estimation (MLE).

- Likelihood of a single observation (x_i, y_i):

$$L_i(\beta) = p_i^{y_i} \cdot (1 - p_i)^{1-y_i} = \left(\frac{e^{x_i^T \beta}}{1 + e^{x_i^T \beta}} \right)^{y_i} \cdot \left(1 - \frac{e^{x_i^T \beta}}{1 + e^{x_i^T \beta}} \right)^{1-y_i}$$

Fitting a logistic regression

Traditionally, use maximum likelihood estimation (MLE).

- Likelihood of a single observation (x_i, y_i):

$$L_i(\beta) = p_i^{y_i} \cdot (1 - p_i)^{1-y_i} = \left(\frac{e^{x_i^T \beta}}{1 + e^{x_i^T \beta}} \right)^{y_i} \cdot \left(1 - \frac{e^{x_i^T \beta}}{1 + e^{x_i^T \beta}} \right)^{1-y_i}$$

- Log-likelihood of a single observation:

$$\begin{aligned} \ell_i(\beta) &= y_i \log \left(\frac{e^{x_i^T \beta}}{1 + e^{x_i^T \beta}} \right) + (1 - y_i) \log \left(\frac{1}{1 + e^{x_i^T \beta}} \right) \\ &= y_i x_i^T \beta - \log(1 + e^{x_i^T \beta}) \end{aligned}$$

Fitting a logistic regression

Traditionally, use maximum likelihood estimation (MLE).

- Likelihood of a single observation (x_i, y_i):

$$L_i(\beta) = p_i^{y_i} \cdot (1 - p_i)^{1-y_i} = \left(\frac{e^{x_i^T\beta}}{1 + e^{x_i^T\beta}} \right)^{y_i} \cdot \left(1 - \frac{e^{x_i^T\beta}}{1 + e^{x_i^T\beta}} \right)^{1-y_i}$$

- Log-likelihood of a single observation:

$$\begin{aligned} \ell_i(\beta) &= y_i \log \left(\frac{e^{x_i^T\beta}}{1 + e^{x_i^T\beta}} \right) + (1 - y_i) \log \left(\frac{1}{1 + e^{x_i^T\beta}} \right) \\ &= y_i x_i^T \beta - \log(1 + e^{x_i^T\beta}) \end{aligned}$$

- Aggregate log-likelihood:

$$\ell(\beta) = \sum \left(y_i x_i^T \beta - \log(1 + e^{x_i^T\beta}) \right)$$

Extension to more than 2 classes

Multinomial logistic regression extends the logistic regression model to $K \geq 2$ classes.

$$\log\left(\frac{P(Y = k \mid X = x)}{P(Y = 0|X = x)}\right) = x^T\beta_k, \qquad k = 1, 2, \ldots, K - 1$$

Extension to more than 2 classes

Multinomial logistic regression extends the logistic regression model to $K \geq 2$ classes.

$$\log\left(\frac{P(Y=k\,|\,X=x)}{P(Y=0|X=x)}\right) = x^T\beta_k, \qquad k = 1, 2, \ldots, K-1$$

$$P(Y=k\,|\,X=x) = \frac{\exp(x^T\beta_k)}{1+\sum_{l=1}^{K-1}\exp(x^T\beta_l)}, \qquad k = 1, 2, \ldots, K-1$$

Extension to more than 2 classes

Multinomial logistic regression extends the logistic regression model to $K \geq 2$ classes.

$$\log\left(\frac{P(Y=k \mid X=x)}{P(Y=0|X=x)}\right) = x^T\beta_k, \qquad k = 1, 2, \ldots, K-1$$

$$P(Y=k \mid X=x) = \frac{\exp(x^T\beta_k)}{1+\sum_{l=1}^{K-1}\exp(x^T\beta_l)}, \qquad k = 1, 2, \ldots, K-1$$

$$P(Y=\cdot \mid X=x) = \text{softmax}\left(1, \exp(x^T\beta_1), \ldots, \exp(x^T\beta_{K-1})\right)$$

Separable classes

An important consideration related to the likelihood...

Separable classes

An important consideration related to the likelihood...

Note: When we write $x^T\beta$ this will (usually) imply that we have included an intercept,

$$x^T\beta = \beta_0 + \beta_1 x_1 + \cdots \beta_p x_p$$

Recall that logistic regression fits a linear decision boundary:

$$\left\{x : x^T\widehat{\beta} = 0\right\}$$

Separable classes

An important consideration related to the likelihood...

Note: When we write $x^T\beta$ this will (usually) imply that we have included an intercept,

$$x^T\beta = \beta_0 + \beta_1 x_1 + \cdots \beta_p x_p$$

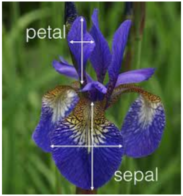
Recall that logistic regression fits a linear decision boundary:

$$\left\{x : x^T\widehat{\beta} = 0\right\}$$

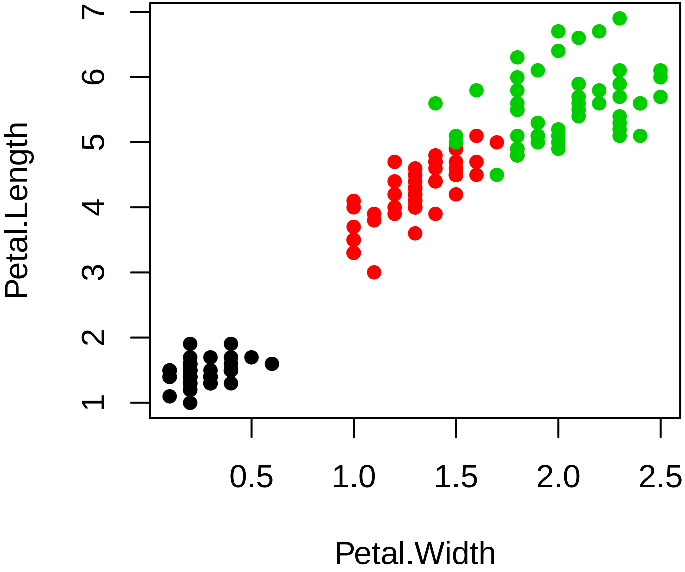
What happens when the two classes are *separable* (*i.e.*, a hyperplane can perfectly separate out the two classes)?

Fisher’s iris classification

Iris setosa (Left), *Iris versicolor* (Middle), and *Iris virginica* (Right).

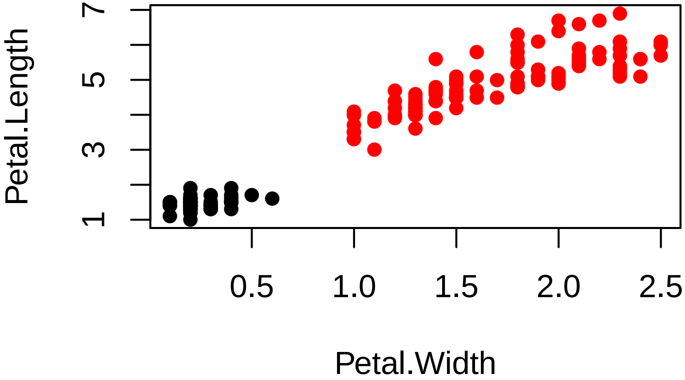


Separable classes

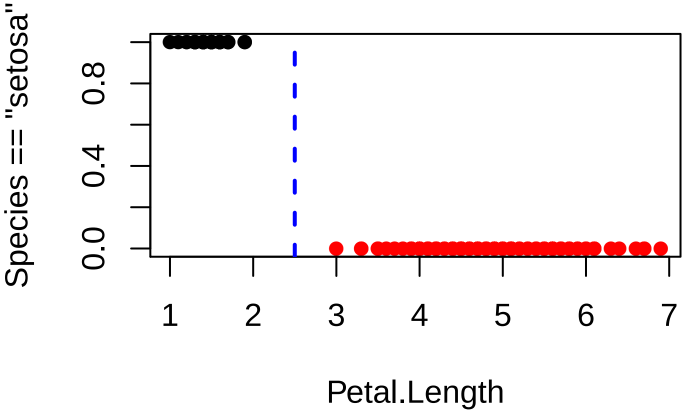


Separable classes

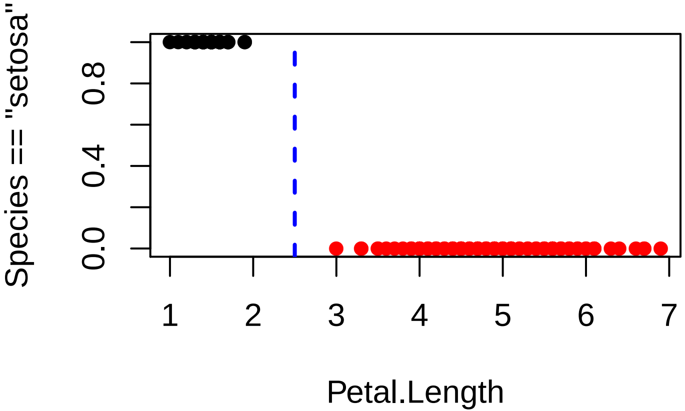
Pretend we only care for predicting setosas ($Y = 1$) vs. non-setosas ($Y = 0$):



Separable classes



Separable classes



Petal length of 2.5 can perfectly separate $Y = 1$ and $Y = 0$ groups.

Separable classes

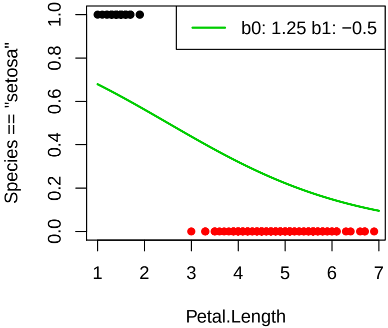
Decision boundary: $\widehat{\beta}_0 + \widehat{\beta}_1 x = 0$.

$\widehat{\beta}_1 = -\frac{\widehat{\beta}_0}{2.5}$ for $\widehat{\beta}_1 < 0$ will yield perfect fits.

Separable classes

Decision boundary: $\widehat{\beta}_0 + \widehat{\beta}_1 x = 0$.

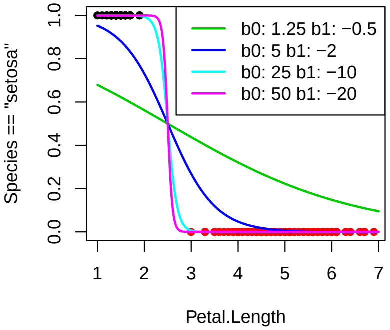
$\widehat{\beta}_1 = -\frac{\widehat{\beta}_0}{2.5}$ for $\widehat{\beta}_1 < 0$ will yield perfect fits.



Separable classes

Decision boundary: $\widehat{\beta}_0 + \widehat{\beta}_1 x = 0$.

$\widehat{\beta}_1 = -\frac{\widehat{\beta}_0}{2.5}$ for $\widehat{\beta}_1 < 0$ will yield perfect fits.

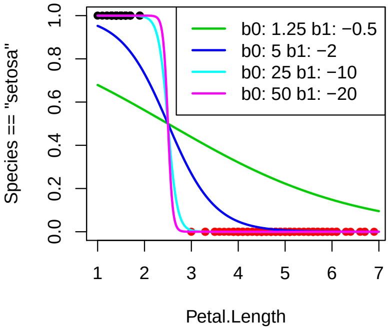


Int	Slope	Likelihood
1.25	-0.5	0.0000000
5.00	-2.0	0.0001696
25.00	-10.0	0.9846004
50.00	-20.0	0.9999415

Separable classes

Decision boundary: $\widehat{\beta}_0 + \widehat{\beta}_1 x = 0$.

$\widehat{\beta}_1 = -\frac{\widehat{\beta}_0}{2.5}$ for $\widehat{\beta}_1 < 0$ will yield perfect fits.



Int	Slope	Likelihood
1.25	-0.5	0.0000000
5.00	-2.0	0.0001696
25.00	-10.0	0.9846004
50.00	-20.0	0.9999415

As $\|\beta\|$ increases, likelihood approaches 1.

Separable classes

Problematic?

- Appears that predictor is not informative, but it is!

Separable classes

Problematic?

- Appears that predictor is not informative, but it is!
- Theoretically we obtained a perfect fit on the training data.

Separable classes

Problematic?

- Appears that predictor is not informative, but it is!
- Theoretically we obtained a perfect fit on the training data.
  - Overfitting is possible. Regularization can help.

Examples in Jupyter notebook

Lets work through some examples in a Jupyter notebook. Please open `classification-examples.ipynb` and run the notebook as we go through it.