Lecture 25 – Logistic Regression and Maximum Likelihood Estimation

DSC 40A, Fall 2022 @ UC San Diego

Dr. Truong Son Hy, with help from **many others**

Announcements

- Look at the readings linked on the course website!
- We will have the Thanksgiving break, so there is no class on Friday this week.
- The final is coming, so there will be a review session next week.

Agenda

- Text classification by Naive Bayes classifier (continued).
- Logistic Regression.
- Maximum Likelihood Estimation.

Text classification by Naive Bayes classifier (continued)

Concrete example

Dictionary: “prince”, “money”, “free”, and “xxx”.
Dataset of 5 emails (red are spam, green are ham):

“I am the prince of UCSD and I demand money.”
“Tapioca Express: redeem your free Thai Iced Tea!”
“DSC 40A: free points if you fill out CAPEs!”
“Click here to make a tax-free donation to the IRS.”
“Free COVID-19 tests at Prince Center.”

	prince	money	free	xxx	**Label**
Sentence 1	1	1	0	0	**spam**
Sentence 2	0	0	1	0	**ham**
Sentence 3	0	0	1	0	**ham**
Sentence 4	0	0	1	0	**spam**
Sentence 5	1	0	1	0	**ham**

Concrete example

	prince	money	free	xxx	**Label**
Sentence 1	1	1	0	0	**spam**
Sentence 2	0	0	1	0	**ham**
Sentence 3	0	0	1	0	**ham**
Sentence 4	0	0	1	0	**spam**
Sentence 5	1	0	1	0	**ham**

$x^{(1)}$ = prince, $x^{(2)}$ = money, $x^{(3)}$ = free, $x^{(4)}$ = xxx

Prior:

$$P(\text{spam}) = \frac{2}{5}$$

$$P(\text{ham}) = \frac{3}{5}$$

Concrete example

	prince	money	free	xxx	**Label**
Sentence 1	1	1	0	0	**spam**
Sentence 2	0	0	1	0	**ham**
Sentence 3	0	0	1	0	**ham**
Sentence 4	0	0	1	0	**spam**
Sentence 5	1	0	1	0	**ham**

$x^{(1)}$ = prince, $x^{(2)}$ = money, $x^{(3)}$ = free, $x^{(4)}$ = xxx

Conditional probability on spam:

$$P(x^{(1)} = 0|\text{spam}) = \frac{1}{2}, \quad P(x^{(1)} = 1|\text{spam}) = \frac{1}{2},$$

$$P(x^{(2)} = 0|\text{spam}) = \frac{1}{2}, \quad P(x^{(2)} = 1|\text{spam}) = \frac{1}{2},$$

$$P(x^{(3)} = 0|\text{spam}) = \frac{1}{2}, \quad P(x^{(3)} = 1|\text{spam}) = \frac{1}{2},$$

$$P(x^{(4)} = 0|\text{spam}) = 1, \quad P(x^{(4)} = 1|\text{spam}) = 0.$$

Concrete example

	prince	money	free	xxx	**Label**
Sentence 1	1	1	0	0	**spam**
Sentence 2	0	0	1	0	**ham**
Sentence 3	0	0	1	0	**ham**
Sentence 4	0	0	1	0	**spam**
Sentence 5	1	0	1	0	**ham**

$x^{(1)}$ = prince, $x^{(2)}$ = money, $x^{(3)}$ = free, $x^{(4)}$ = xxx

Conditional probability on ham:

$$P(x^{(1)} = 0|\text{ham}) = \frac{2}{3}, \quad P(x^{(1)} = 1|\text{ham}) = \frac{1}{3},$$

$$P(x^{(2)} = 0|\text{ham}) = 1, \quad P(x^{(2)} = 1|\text{ham}) = 0,$$

$$P(x^{(3)} = 0|\text{ham}) = 0, \quad P(x^{(3)} = 1|\text{ham}) = 1,$$

$$P(x^{(4)} = 0|\text{ham}) = 1, \quad P(x^{(4)} = 1|\text{ham}) = 0.$$

Concrete example

- New email to classify: “Download a free copy of the Prince of Persia.”‘

Concrete example

- New email to classify: “Download a free copy of the Prince of Persia.”‘

prince	money	free	xxx
1	0	1	0

To compute the probability of the text being **spam**, we have: $P(\text{features}|\text{spam})$

$= P(x^{(1)} = 1|\text{spam})P(x^{(2)} = 0|\text{spam})P(x^{(3)} = 1|\text{spam})P(x^{(4)} = 0|\text{spam})$

$= \frac{1}{2} \cdot \frac{1}{2} \cdot \frac{1}{2} \cdot 1 = \frac{1}{8}$

Thus:

$$P(\text{spam}|\text{features}) \propto P(\text{features}|\text{spam}) \cdot P(\text{spam}) = \frac{1}{8} \cdot \frac{2}{5} = \frac{1}{20}$$

Concrete example

- New email to classify: "Download a free copy of the Prince of Persia."'

prince	money	free	xxx
1	0	1	0

To compute the probability of the text being **ham**, we have:
$P(\text{features}|\text{ham})$

$$= P(x^{(1)} = 1|\text{ham})P(x^{(2)} = 0|\text{ham})P(x^{(3)} = 1|\text{ham})P(x^{(4)} = 0|\text{ham})$$

$= \frac{1}{3} \cdot 1 \cdot 1 \cdot 1 = \frac{1}{3}$
Thus:

$$P(\text{ham}|\text{features}) \propto P(\text{features}|\text{ham}) \cdot P(\text{ham}) = \frac{1}{3} \cdot \frac{3}{5} = \frac{1}{5}$$

Concrete example

- New email to classify: “Download a free copy of the Prince of Persia.”‘

prince	money	free	xxx
1	0	1	0

Because

$$P(\text{ham}|\text{features}) = \frac{1}{5} > P(\text{spam}|\text{features}) = \frac{1}{20},$$

this sentence is classified as **ham**.

Uh oh...

- What happens if we try to classify the email “xxx what’s your price, prince”?

Uh oh...

- What happens if we try to classify the email “xxx what’s your price, prince”?

prince	money	free	xxx
1	0	0	1

There is a keyword “xxx” and the sentence is likely **spam**. But:

$$P(x^{(4)} = 1|\text{spam}) = 0$$

Thus:

$$P(\text{features}|\text{spam}) = 0$$

Then, it will be classified as **ham** with absolute certainty.

Smoothing

- **Without** smoothing:

$$P(x^{(i)} = 1 \mid \text{spam}) \approx \frac{\text{\# spam containing word } i}{\text{\# spam containing word } i + \text{\# spam not containing word } i}$$

- **With** smoothing:

$$P(x^{(i)} = 1 \mid \text{spam}) \approx \frac{(\text{\# spam containing word } i) + 1}{(\text{\# spam containing word } i) + 1 + (\text{\# spam not containing word } i) + 1}$$

- When smoothing, we add 1 to the count of every group whenever we're estimating a conditional probability.
  - **Don't** smooth the estimates of unconditional probabilities (e.g. $P(\text{spam})$).

Concrete example with smoothing

	prince	money	free	xxx	**Label**
Sentence 1	1	1	0	0	**spam**
Sentence 2	0	0	1	0	**ham**
Sentence 3	0	0	1	0	**ham**
Sentence 4	0	0	1	0	**spam**
Sentence 5	1	0	1	0	**ham**

$x^{(1)}$ = prince, $x^{(2)}$ = money, $x^{(3)}$ = free, $x^{(4)}$ = xxx

Conditional probability on spam:

$$P(x^{(1)} = 0|\text{spam}) = \frac{1}{2}, \quad P(x^{(1)} = 1|\text{spam}) = \frac{1}{2},$$

$$P(x^{(2)} = 0|\text{spam}) = \frac{1}{2}, \quad P(x^{(2)} = 1|\text{spam}) = \frac{1}{2},$$

$$P(x^{(3)} = 0|\text{spam}) = \frac{1}{2}, \quad P(x^{(3)} = 1|\text{spam}) = \frac{1}{2},$$

$$P(x^{(4)} = 0|\text{spam}) = \frac{2}{3}, \quad P(x^{(4)} = 1|\text{spam}) = \frac{1}{3}.$$

Concrete example with smoothing

	prince	money	free	xxx	**Label**
Sentence 1	1	1	0	0	**spam**
Sentence 2	0	0	1	0	**ham**
Sentence 3	0	0	1	0	**ham**
Sentence 4	0	0	1	0	**spam**
Sentence 5	1	0	1	0	**ham**

$x^{(1)}$ = prince, $x^{(2)}$ = money, $x^{(3)}$ = free, $x^{(4)}$ = xxx

Conditional probability on ham:

$$P(x^{(1)} = 0|\text{ham}) = \frac{3}{5}, \quad P(x^{(1)} = 1|\text{ham}) = \frac{2}{5},$$

$$P(x^{(2)} = 0|\text{ham}) = \frac{2}{3}, \quad P(x^{(2)} = 1|\text{ham}) = \frac{1}{3},$$

$$P(x^{(3)} = 0|\text{ham}) = \frac{1}{3}, \quad P(x^{(3)} = 1|\text{ham}) = \frac{2}{3},$$

$$P(x^{(4)} = 0|\text{ham}) = \frac{2}{3}, \quad P(x^{(4)} = 1|\text{ham}) = \frac{1}{3}.$$

Concrete example with smoothing

- What happens if we try to classify the email “xxx what’s your price, prince”?

prince	money	free	xxx
1	0	0	1

Probability of **spam**:

$$P(\text{features}|\text{spam})$$

$$= P(x^{(1)} = 1|\text{spam})P(x^{(2)} = 0|\text{spam})P(x^{(3)} = 0|\text{spam})P(x^{(4)} = 1|\text{spam})$$

$$= \frac{1}{2} \cdot \frac{1}{2} \cdot \frac{1}{2} \cdot \frac{1}{3} = \frac{1}{24}$$

Thus:

$$P(\text{spam}|\text{features}) \propto P(\text{features}|\text{spam}) \cdot P(\text{spam}) = \frac{1}{24} \cdot \frac{2}{5} = \frac{1}{60} \approx 0.0166$$

Concrete example with smoothing

- What happens if we try to classify the email "xxx what's your price, prince"?

prince	money	free	xxx
1	0	0	1

Probability of **ham**:

$$P(\text{features}|\text{ham})$$

$$= P(x^{(1)} = 1|\text{ham})P(x^{(2)} = 0|\text{ham})P(x^{(3)} = 0|\text{ham})P(x^{(4)} = 1|\text{ham})$$

$$= \frac{2}{5} \cdot \frac{2}{3} \cdot \frac{1}{3} \cdot \frac{1}{3} = \frac{4}{135}$$

Thus:

$$P(\text{ham}|\text{features}) \propto P(\text{features}|\text{ham}) \cdot P(\text{ham}) = \frac{4}{135} \cdot \frac{3}{5} \approx 0.0177$$

Concrete example with smoothing

- What happens if we try to classify the email "xxx what's your price, prince"?

We have:

$$P(\text{spam}|\text{features}) \approx 0.0166$$

$$P(\text{ham}|\text{features}) \approx 0.0177$$

Probability of **spam**: 48.3%
Probability of **ham**: 51.7%
This is a confusing case for Naive Bayes classifier. We need more data!

Practical demo (see code for Lecture 24)

More realistic example

My source code in Java (it is easier to do in Python):

`https://github.com/HyTruongSon/Spambase-filtering`

Data:

`https://archive.ics.uci.edu/ml/datasets/Spambase`

Classifiers: Linear/RBF Support Vector Machine, Logistic Regression and Multilayer Perceptron.

Logistic Regression & Maximum Likelihood Estimation

Introduction

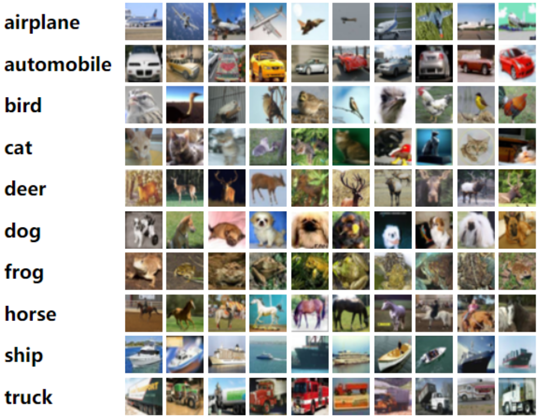
- Classification methods of supervised machine learning have many successful applications in vision, speech, medicine, finance, etc.

- **Setup:** We need to map $\vec{x} \in X$ to a label $y \in Y$.

- Examples:

Digit images (MNIST dataset): $\vec{x} \in R^{28\times 28}$, $y \in \{0, 1, .., 9\}$.

Introduction



The CIFAR-10 dataset consists of 60,000 32 × 32 colour images in 10 classes, with 6,000 images per class. There are 50,000 training images and 10,000 test images.

Classification as regression?

- Suppose we have a binary problem: $y \in \{-1, +1\}$.
- **Idea:** Treat it as regression, with squared loss.
- Assuming the model $y = f(\vec{x}; \vec{w}, w_0) = \vec{x} \cdot \vec{w} + w_0$, and solving with least squares, we get $\vec{w}^*$ and w_0^*.
- This corresponds to squared loss as a measure of classification performance! Does this make sense?
- How do we decide on the label based on $f(\vec{x}; \vec{w}^*, w_0^*)$?

Classification as regression?

- Model:

$$f(\vec{x}; \vec{w}^*, w_0^*) = \vec{w}^* \cdot \vec{x} + w_0^*$$

- Cannot just take $\hat{y} = f(\vec{x}; \vec{w}^*, w_0^*)$ since it won't be a valid label.

- A reasonable **decision rule**:

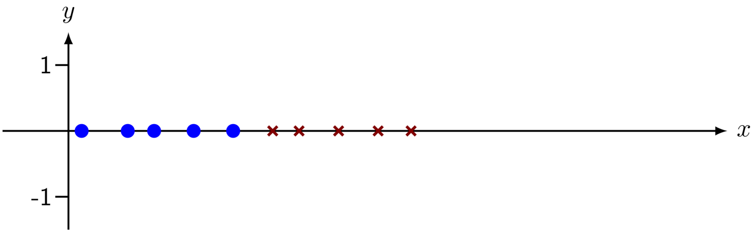
$$\hat{y} = \text{sign}(\vec{w}^* \cdot \vec{x} + w_0^*)$$

  If $f(\vec{x}; \vec{w}^*, w_0^*) \geq 0$ then $\hat{y} = 1$, otherwise $\hat{y} = -1$.

- This specifies a **linear classifier**: The linear **decision boundary** (hyperplane) given by the equation $\vec{w}^* \cdot \vec{x} + w_0^* = 0$ separates the space into two "half-spaces".

Example on 1D

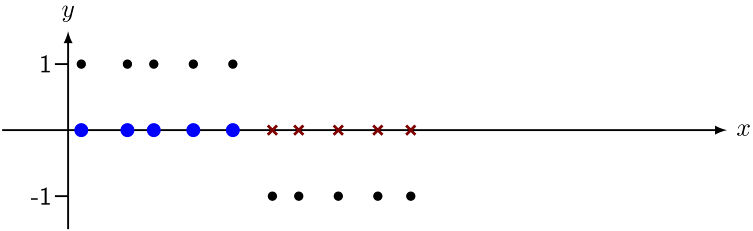
Let's consider the following data on 1-dimensional space. We can easily separate the blue dots from the red crosses.



But can the **linear classifier** successfully classify this data with 100% accuracy?

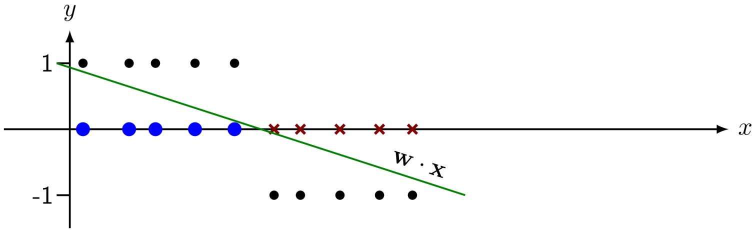
Example on 1D

The value for blue dots is +1. The value for red crosses is -1. Let's try our linear regression!



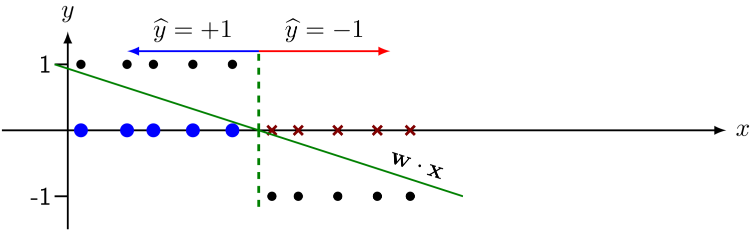
Example on 1D

The green line is our decision boundary / hyperplane. Let's classify the points!



Example on 1D

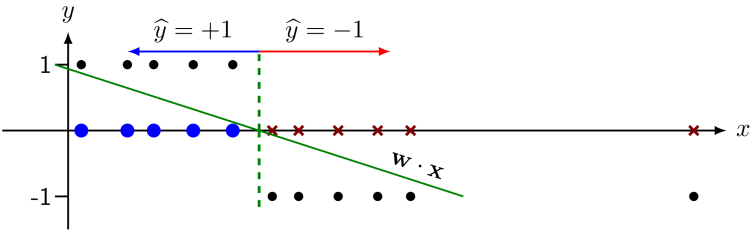
Our **linear classifier** can classify this data with 100% accuracy.



But let's add one more point to the data!

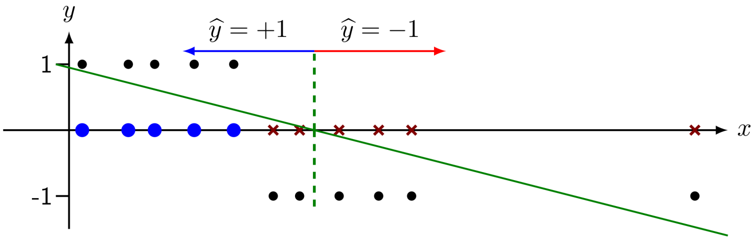
Example 1D

We add one outlier to the right. By a simple threshold, we can easily classify this data. But let's see how this outlier affects our linear regression and decision boundary!



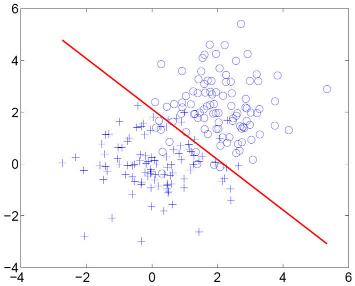
Example 1D

The linear regression is sensitive to the outlier. As the consequence, our linear classifier can no longer classify this simple data with 100% accuracy!

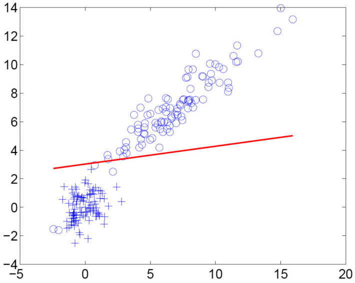


Example 2D

Let's consider some data on 2-dimensional space!



Seems to work well here



but not so well here

In conclusion, we should **not** use the squared loss.

Linear classifier

- Hypothesis:
$$\hat{y} = h(\vec{x}) = \mathrm{sign}(\vec{x} \cdot \vec{w} + w_0)$$
- Classifying using a linear decision boundary effectively reduces the data dimension to 1.
- We need to find the direction $\vec{w}$ and location w_0 of the boundary.
- We want to minimize the expected **zero/one** loss for classifier $h : X \to Y$, which for $(\vec{x}, y)$ is:
$$L(h(\vec{x}), y) = \begin{cases} 0 & \text{if } h(\vec{x}) = y, \\ 1 & \text{if } h(\vec{x}) \neq y. \end{cases}$$

Empirical Risk Minimization

- The risk (expected loss) of a C-way classifier $h(\vec{x})$ (i.e. C is the number of classes):

$$R(h) = E_{p(\vec{x},y)}[L(h(\vec{x}), y)],$$

where E denotes the expectation and $p(\vec{x}, y)$ denotes the joint probability distribution of our data $(\vec{x}, y)$. Our data is considered as samples drawn from p.

- We can write the risk in intergral form:

$$R(h) = \int_{\vec{x}} \sum_{c=1}^{C} L(h(\vec{x}), c)p(\vec{x}, y = c)d\vec{x}$$

Empirical Risk Minimization

- We can further write the risk as:

$$R(h) = \int_{\vec{x}} \left[\sum_{c=1}^{C} L(h(\vec{x}), c) p(y = c | \vec{x}) \right] p(\vec{x}) d\vec{x}$$

- Clearly, it is enough to minimize the **conditional risk** for any $\vec{x}$:

$$R(h|\vec{x}) = \sum_{c=1}^{C} L(h(\vec{x}), c) p(y = c | \vec{x})$$

- **Next time:** We will continue learning about how to find the hypothesis h via the ERM framework and derive to Logistic Regression.