Machine Learning

Lecture 04. Logistic Regression

Kien C Nguyen

August 26, 2024

Classification

- Recall from Lecture 1, in a classification problem, the goal is to learn a mapping from inputs $\mathbf{x}$ to outputs y, where $y \in \{0, \ldots, C-1\}$, with C being the number of classes.
- If $C = 2$, this is called binary classification ($y \in \{0, 1\}$
- if $C > 2$, this is called multiclass classification.

Examples of Binary Classification

- Classify an email as Not Spam / Spam
- In credit scoring, classify a customer as Good / Bad
- In network intrusion detection, classify a connection as Normal / Attack
- Detect the gender (Male / Female) using profile pictures
- Here y can take on only two values, 0 and 1.
- For instance, if we are building an spam classifier for email, then $\{\mathbf{x}_i\}_{i=1}^{N}$ may be features of an email, and y may be 1 if it is a spam email, and 0 otherwise. 0 is also called the negative class, and 1 the positive class.

Logistic Regression

- Recall that in linear regression, $\hat{y} = \mathbf{w}^T \mathbf{x}$ (Here we let $x_0 = 1$ to simplify the notation).
- Note that $\hat{y}$ is neither upper-bounded nor lower-bounded.
- In logistic regression, we predict the probability that $y = 1$.
  - Probability that an email is spam
  - Probability that a customer is a Bad customer.

Probability of passing an exam versus hours of study

- A group of 20 students spend between 0 and 6 hours studying for an exam. How does the number of hours spent studying affect the probability that the student will pass the exam?
- We predict the probability that a student passes the exam ($y = 1$) using the number of hours that student spent.

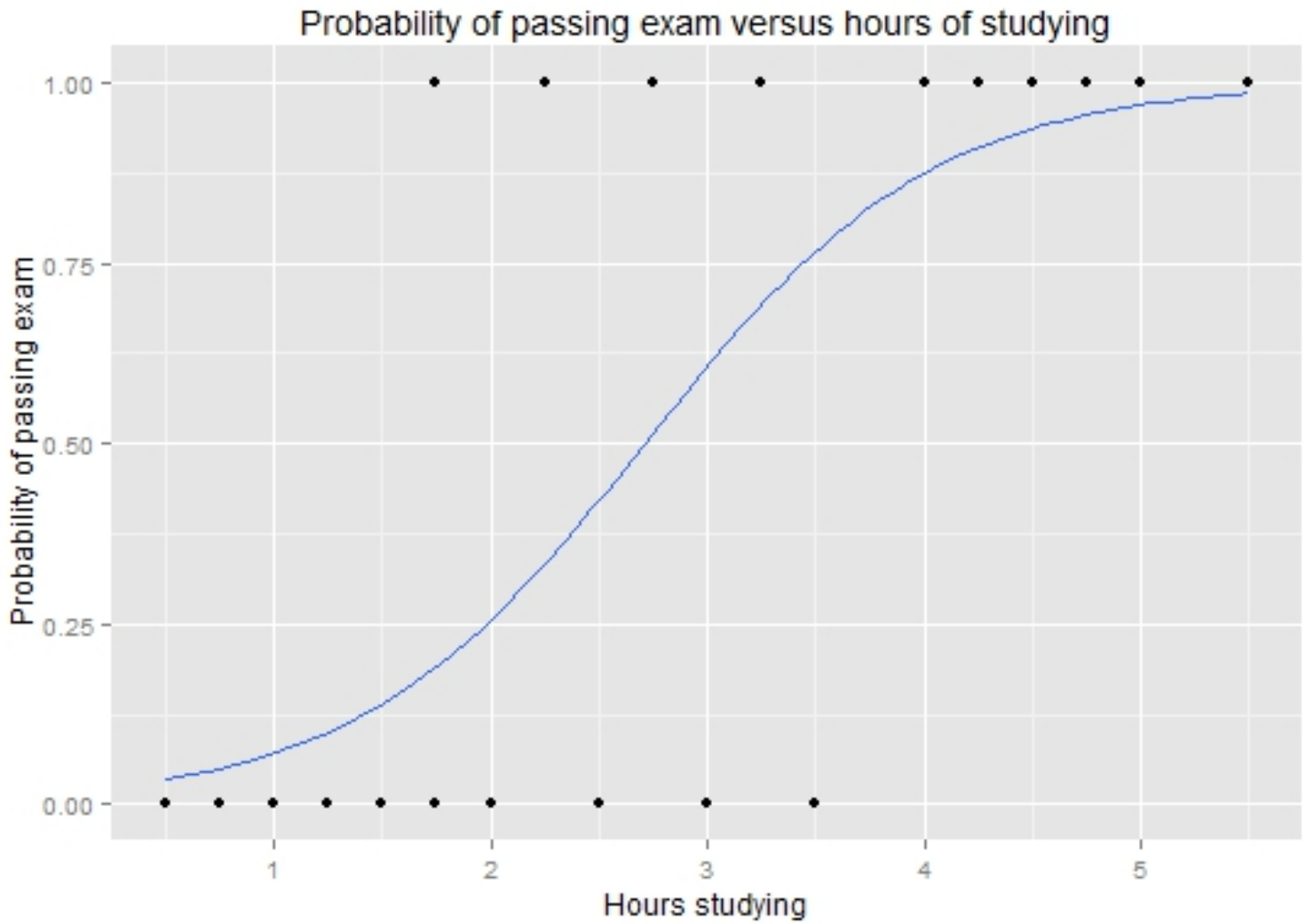
Source: https://en.wikipedia.org/wiki/Logistic_regression

Probability of passing an exam versus hours of study

Hours	Pass	Hours	Pass
.5	0	2.75	1
.75	0	3	0
1	0	3.25	1
1.25	0	3.5	0
1.5	0	4	1
1.75	0	4.25	1
1.75	1	4.5	1
2	0	4.75	1
2.25	1	5	1
2.5	0	5.5	1

Source: https://machinelearningcoban.com/2017/01/27/logisticregression/

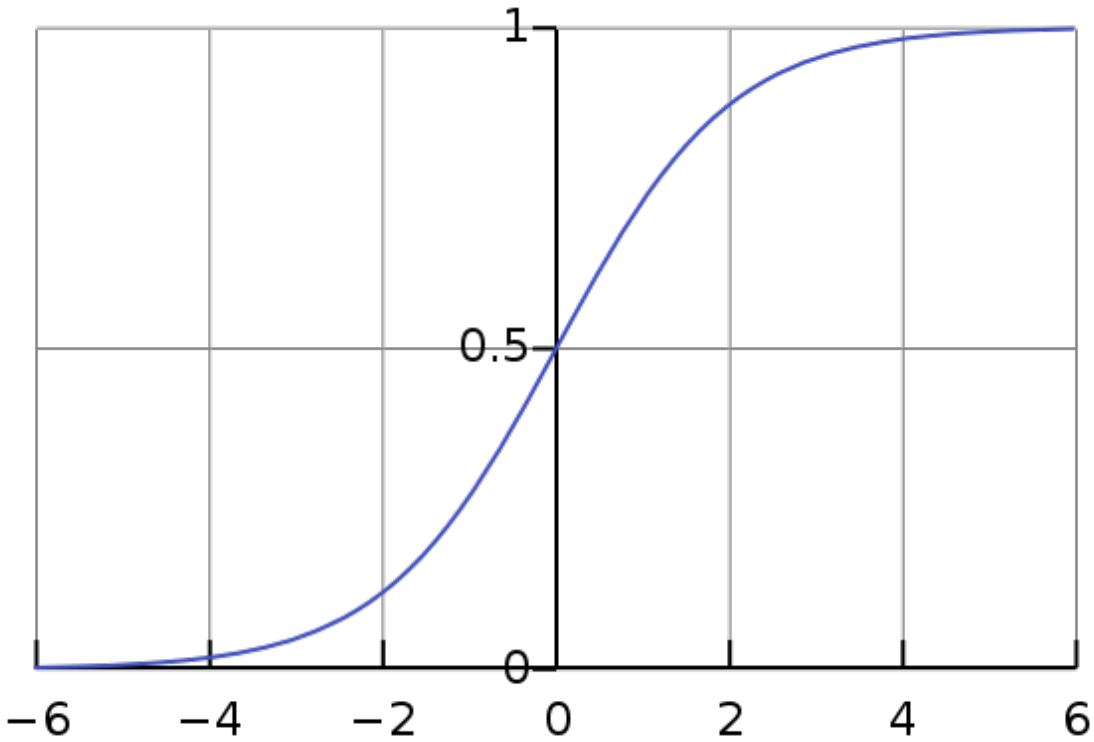
Probability of passing an exam versus hours of study



Source: https://machinelearningcoban.com/2017/01/27/logisticregression/

- Use a function $\Phi(\mathbf{w}^T\mathbf{x})$
- As this is a probability, we want $0 \leq \Phi(\mathbf{w}^T\mathbf{x}) \leq 1$
- Sigmoid function (Logistic function)

$$\sigma(z) = \frac{1}{1 + e^{-z}}$$

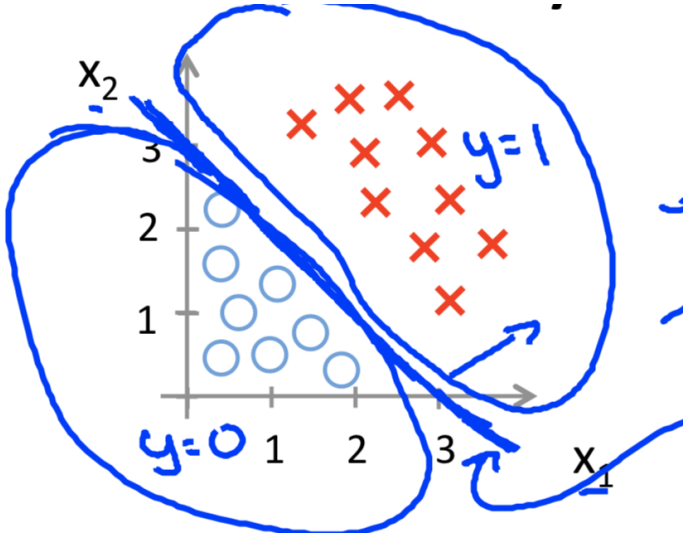


Source: https://en.wikipedia.org/wiki/Logistic_regression

Suppose we predict $y = 1$ if $P\{y = 1\} \geq 0.5$.

$$\sigma(z) \geq 0 \iff \mathbf{w}^T\mathbf{x} \geq 0$$

Predict $y = 0$ if $P\{y = 1\} < 0.5 \iff \mathbf{w}^T\mathbf{x} < 0$



Source: Andrew Ng – Machine Learning (Coursera)

Contents

Recall that the training set is $\left(\left(\mathbf{x}^{(1)}, y^{(1)}\right), \left(\mathbf{x}^{(2)}, y^{(2)}\right), \ldots \left(\mathbf{x}^{(N)}, y^{(N)}\right)\right)$.

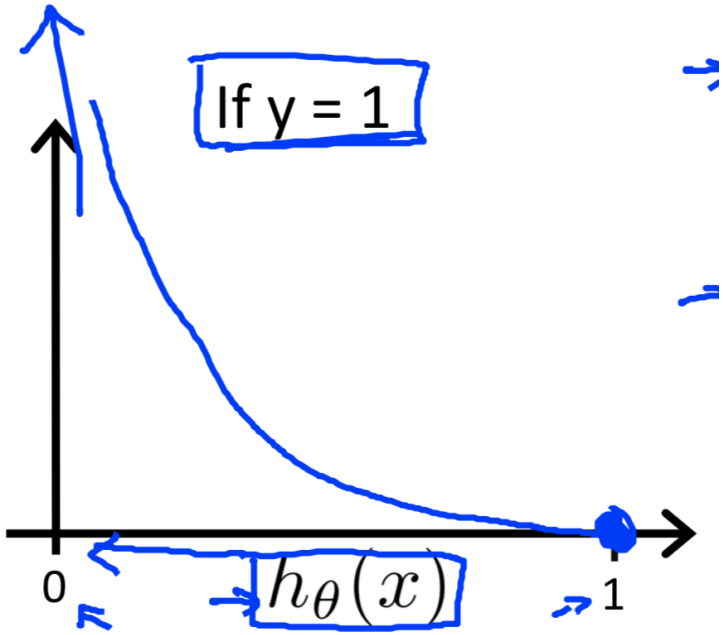
where $\mathbf{x}^{(i)}$ is given by $\mathbf{x}^{(i)} = \begin{bmatrix} x_0^{(i)} \\ x_1^{(i)} \\ \ldots \\ x_D^{(i)} \end{bmatrix}$

$x_0^{(i)} = 1$, $y^{(i)} \in \{0, 1\}$

Loss for each training example

$$
\begin{aligned}
L(\hat{y}, y) &= \begin{cases} -log(\Phi(\mathbf{w}^T\mathbf{x})) & if \quad y = 1 \\ -log(1 - \Phi(\mathbf{w}^T\mathbf{x})) & if \quad y = 0 \end{cases} \\
&= -ylog(\Phi(\mathbf{w}^T\mathbf{x})) - (1 - y)log(1 - \Phi(\mathbf{w}^T\mathbf{x}))
\end{aligned}
$$

Loss for each training example: $y = 1$

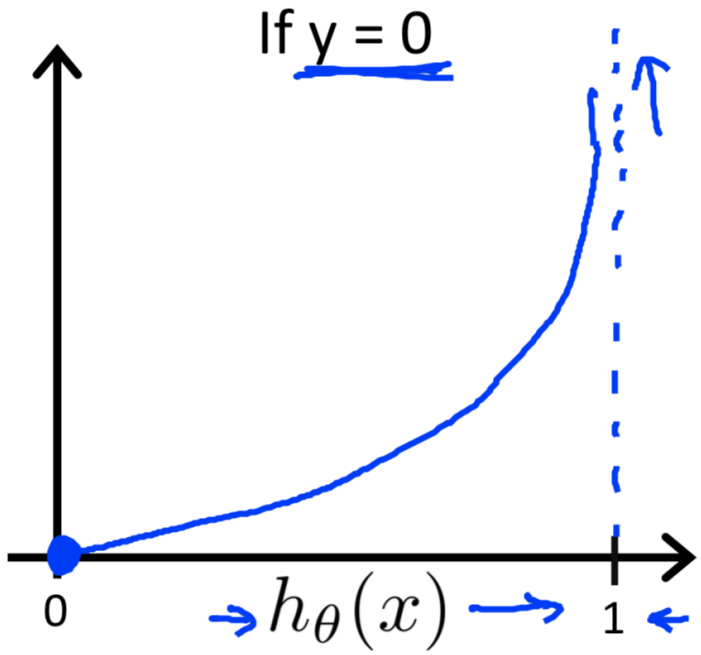


$L = 0$ when $\Phi(\mathbf{w}^T\mathbf{x}) = 1$

$L \to \infty$ as $\Phi(\mathbf{w}^T\mathbf{x}) \to 0$

Source: Andrew Ng – Machine Learning (Coursera)

Loss for each training example: $y = 0$



$L = 0$ when $\Phi(\mathbf{w}^T\mathbf{x}) = 0$

$L \to \infty$ as $\Phi(\mathbf{w}^T\mathbf{x}) \to 1$

Source: Andrew Ng – Machine Learning (Coursera)

Loss function

$$
\begin{aligned}
L(\mathbf{w}) &= \frac{1}{N}\sum_{i=1}^{N} -y^{(i)} log(\Phi(\mathbf{w}^T\mathbf{x}^{(i)})) - (1-y^{(i)}) log(1-\Phi(\mathbf{w}^T\mathbf{x}^{(i)})) \\
&= -\frac{1}{N}\sum_{i=1}^{N} y^{(i)} log(\Phi(\mathbf{w}^T\mathbf{x}^{(i)})) + (1-y^{(i)}) log(1-\Phi(\mathbf{w}^T\mathbf{x}^{(i)}))
\end{aligned}
$$

We find the $\hat{\mathbf{w}}$ such that

$$\hat{\mathbf{w}} = argmin_{\mathbf{w}} L(\mathbf{w}) \tag{1}$$

Once we have $\hat{\mathbf{w}}$, the prediction for a new $\mathbf{x}$ is

$$P\{\hat{y}=1\} = \Phi(\mathbf{w}^T\mathbf{x}) \tag{2}$$

Contents

Gradient Descent

Initialize $\mathbf{w} = [0, ..., 0]$;
Repeat $\mathbf{w} \leftarrow \mathbf{w} - \eta \nabla_{\mathbf{w}} L(\mathbf{w})$

- η: step size
- $\nabla_{\mathbf{w}} L(\mathbf{w})$: gradient

The update for each w_j:

$$w_j = w_j - \eta \sum_{i=1}^{N} \left(\Phi(\mathbf{w}^T \mathbf{x}^{(i)}) - y^{(i)} \right) x_j^{(i)} \tag{3}$$

Maximum Likelihood Estimation

- Consider a box with only two types of tickets:
  - one has '1' written on it
  - another has '0' written on it.
- Let p be the fraction of the '1' tickets in the box.
- The value of p is unknown.
- Suppose that 100 tickets are drawn from the box and 20 of the tickets are '1'.
- What is the best estimate for the value of p?
- We imagine there are many tickets in the box, so it doesn't matter whether the tickets are drawn with or without replacement.
- In the context of MLE, p is the parameter in the model we are trying to estimate.

Maximum Likelihood Estimation

- We can ask the question: given p , what is the probability that we get 20 tickets with '1' from 100 draws?
- The probability that we get a '1' ticket in each draw is p, and the probability that we get a '0' ticket is $(1 - p)$.
- So the probability that we get 20 '1' tickets and 80 '0' tickets in 100 draws is

$$\mathrm{L(p)} = \mathrm{P}(20 \mid \mathrm{p}) = \binom{100}{20} \mathrm{p}^{20}(1 - \mathrm{p})^{80}$$

Maximum Likelihood Estimation

- This is a function of the unknown parameter p, called the likelihood function.
- It is the probability of getting the data if the parameter is equal to p.
- The maximum likelihood estimate for the parameter is the value of p that maximizes the likelihood function.
- Instead of working with the likelihood function $\mathrm{L(p)}$, it is more convenient to work with the logarithm of L :

$$\ln L(p) = 20 \ln p + 80 \ln(1 - p)$$

Maximum Likelihood Estimation

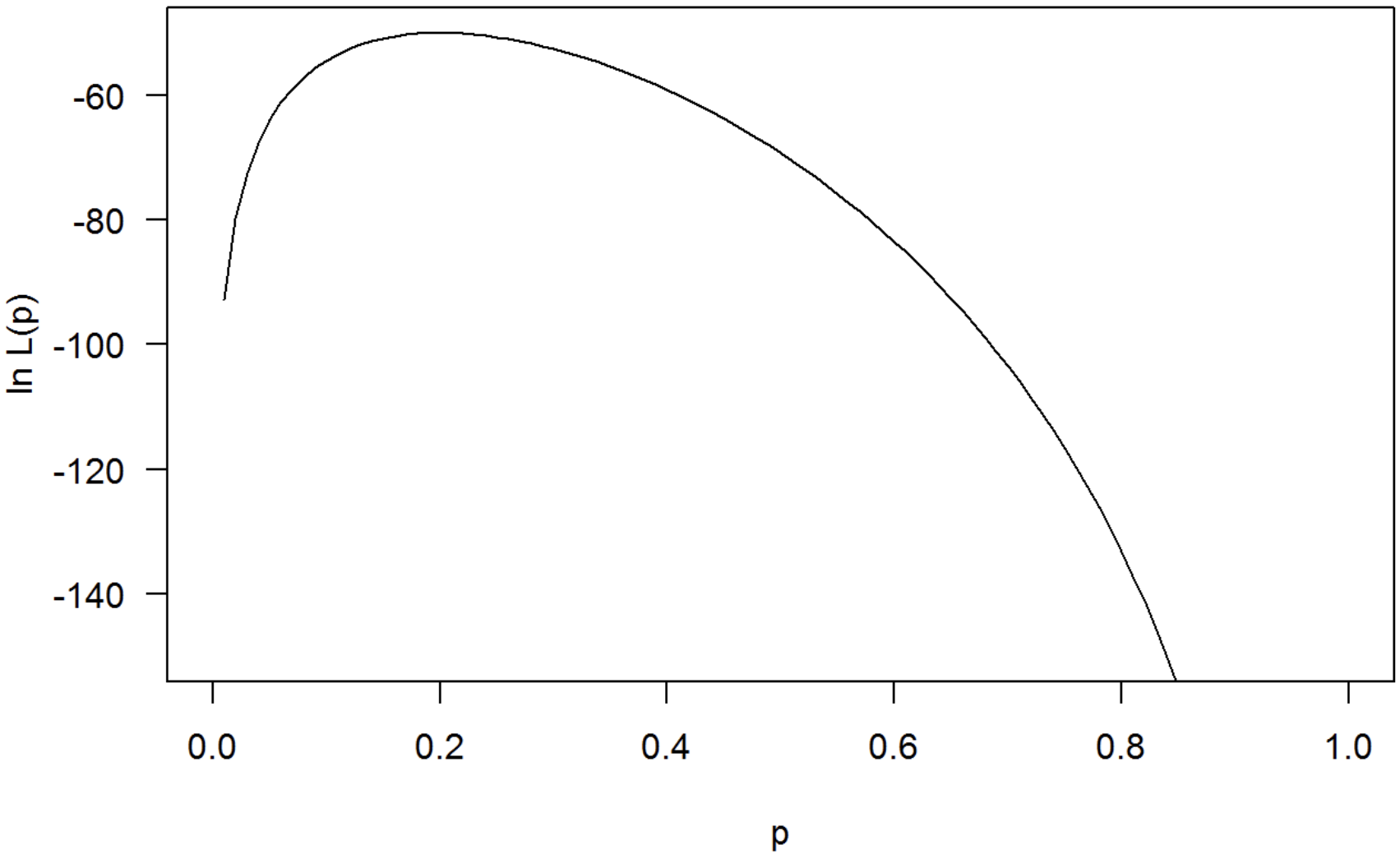


Figure: ln(L) vs p

Maximum Likelihood Estimation

Suppose y_i is a variable encoding the result of the ith draw. We set $y_i = 1$ if the ticket in the ith draw is '1'. We set $y_i = 0$ if the ticket in the ith draw is '0'. After N draws, we have the variables $\mathrm{y_1, y_2, \cdots, y_N}$. The number of '1' tickets in N draws is

$$\mathrm{n_1} = \sum_{\mathrm{i=1}}^{\mathrm{N}} \mathrm{y_i}$$

and so the maximum likelihood estimate for p is

$$\mathrm{p} = \frac{\mathrm{n_1}}{\mathrm{N}} = \frac{1}{\mathrm{N}} \sum_{\mathrm{i=1}}^{\mathrm{N}} \mathrm{y_i} = \overline{\mathrm{y}}$$

In other words, the maximum likelihood estimate for p is the mean of the y variable from the N draws.

Maximum Likelihood Estimation

$$L(p) = P\left(y_1, y_2, \cdots, y_N \mid p\right) = \left[p^{y_1}(1-p)^{1-y_1}\right]\left[p^{y_2}(1-p)^{1-y_2}\right]\cdots\left[p^{y_N}(1-\right.$$

Using the product notation, we can write

$$L(p) = \prod_{i=1}^{N} p^{y_i}(1-p)^{1-y_i}$$

The log-likelihood is given by

$$\ln L(p) = \sum_{i=1}^{N} \left[y_i \ln p + (1-y_i)\ln(1-p)\right]$$

References

[1] Bishop, C. M. (2013). Pattern Recognition and Machine Learning. Journal of Chemical Information and Modeling (Vol. 53).
[2] Wikipedia – Logistic Regression – https://en.wikipedia.org/wiki/Logistic_regression
[3] Vu Huu Tiep – Machine Learning Co Ban – https://machinelearningcoban.com/2017/01/27/logisticregression/
[4] Andrew Ng – Machine Learning (Coursera) – https://www.coursera.org/learn/machine-learning
[5] University of Illionois - Maximum Likelihood and Logistic Regression http://courses.atlas.illinois.edu/spring2016/STAT/STAT200/RProgramming/Maximum_Likelihood.html