BTRY6030/STSCI4110/ILRST4110: Spring 2012

Statistical Methods III: Categorical Data

Instructor: Ping Li

Department of Statistical Science

Cornell University

General Information

- **Lectures**: Tue, Thu 11:40am - 12:55pm, Caldwell Hall 100
- **Instructor**: Ping Li, pingli@cornell.edu,
  Office Hours: Wed. 3pm - 4:10 pm, Comstock Hall 1192.
- **TA**: No TA for this course
- **Prerequisite**: BTRY6010/BTRY6020 Or equivalent
- **Textbook**: Alan Agresti, An Introduction to Categorical Data Analysis

- **Homework**
  - About 5-8 homework assignments.
  - Please turn in your homework either in class or to BSCB front desk (Comstock Hall, 1198).
  - No late homework will be accepted.
  - Before computing your overall homework grade, the assignment with the lowest grade (if $\geq 25\%$) will be dropped, the one with the second lowest grade (if $\geq 50\%$) will also be dropped.
  - It is the students' responsibility to keep copies of the submitted homework.

- **Course grading**:

  1. Homework: 35%
  2. Prelim I: 15% or 20%
  3. Prelim II: 15% or 20%
  4. Final: 30%

  The lower Prelim score will be counted 15% and the higher Prelim score will be counted 20%

- **Course letter grade:**

  A $= 90\%$ (in the absolute scale)

  C $= 60\%$ (in the absolute scale)

In borderline cases, class participation will be used as a determining factor.

Course Description

- **Material:** Logistic regression, Support vector machines (SVM), Clustering, Log-linear models, Stratified tables, matched pairs analysis, polytomous response, and ordinal data. Applications in biomedical, social science, and computer science. Recent techniques for dealing with massive data will also be introduced.

- **Matlab:** Basic programming in Matlab will be taught in the class. Some programming assignments will require coding in Matlab.

- **R**: The R package will also be used. It is available free from the Comprehensive R Archive Network (CRAN): http://www.r-project.org/.

Textbook

Wiley Online Library kindly offers the online version of the textbook:

`http://onlinelibrary.wiley.com/doi/10.1002/9780470114759.ch1/pdf`

Replace “ch1” with “ch2” etc for other chapters.

We will cover selected topics from chapters 1 to 10. The major emphasis of this course is about contingency tables and logistic regression (and related techniques in classification and clustering).

Calculus Review: Derivatives

Simple derivatives:

$$[\log x]' = \frac{1}{x}, \quad [x^n]' = nx^{n-1}, \quad [e^x]' = e^x, \quad [a^x]' = a^x \log a$$

Chain rule:

$$[f(h(x))]' = f'(h(x))\, h'(x)$$

$$\left[\log\left(ax^2 + e^{2x}\right)\right]' = \frac{1}{(ax^2 + e^{2x})}\left[ax^2 + e^{2x}\right]' = \frac{2ax + 2e^{2x}}{(ax^2 + e^x)}$$

Multivariate derivatives:

$$f(x, y) = a^x + x^n y + cy^2,$$

$$\frac{\partial f(x,y)}{\partial x} = a^x \log a + nx^{n-1}y, \qquad \frac{\partial f(x,y)}{\partial y} = x^n + 2cy$$

Contingency Table Estimations

Original Contingency Table

N_{11}	N_{12}
N_{21}	N_{22}

Sample Contingency Table

n_{11}	n_{12}
n_{21}	n_{22}

Suppose we only observe the sample contingency table, how can we estimate the original table, if $N = N_{11} + N_{12} + N_{21} + N_{22}$ is known?

(Almost) equivalently, how can we estimate $\pi_{ij} = \frac{N_{ij}}{N}$?

An Example of Contingency Table

The task is to estimate how many times two words (e.g., Cornell and University) co-occur in all the Web pages (over 10 billion).

	Word 2	No Word 2
Word 1	N_{11}	N_{12}
No Word 1	N_{21}	N_{22}

N_{11}: number of documents containing both word 1 and word 2.

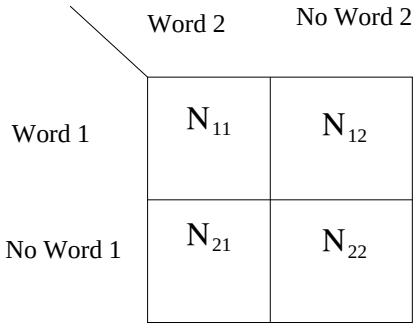
N_{22}: number of documents containing neither word 1 nor word 2.

Google Pagehits

Google tells user the number of Web pages containing the input query word(s).

Pagehits by typing the following queries in Google (numbers can change):

- a : 25,270,000,000 pages (a surrogate for N, the total # of pages).
- Cornell : 99,600,000 pages. ($N_{11} + N_{12}$)
- University : 2,700,000,000 pages. ($N_{11} + N_{21}$)
- Cornell University : 31,800,000 pages. (N_{11})



How much do we believe these numbers?

Suppose there are in total $n = 10^7$ word items.

It is easy to store 10^7 numbers (how many documents each word occurs in), but it would be difficult to store a matrix of $10^7 \times 10^7$ numbers (how many documents a pair of words co-occur in).

Even if storing $10^7 \times 10^7$ is not a problem (it is Google), it is much more difficult to store $10^7 \times 10^7 \times 10^7$ numbers, for 3-way co-occurrences (e.g., Cornell, University, Statistics).

Even if we can store 3-way or 4-way co-occurrences, most of the items will be so rare that they will almost never be used.

Therefore, it is realistic to believe that the counts for individual words are exact, but the numbers of co-occurrences may be estimated, eg, from some samples.

Estimating Contingency Tables by MLE of Multinomial Sampling

Original Contingency Table

π_{11}	π_{12}
π_{21}	π_{22}

Sample Contingency Table

n_{11}	n_{12}
n_{21}	n_{22}

Observations: $(n_{11}, n_{12}, n_{21}, n_{22})$, $\quad n = n_{11} + n_{12} + n_{21} + n_{22}.$

Parameters $(\pi_{11}, \pi_{12}, \pi_{21}, \pi_{22})$, $\quad (\pi_{11} + \pi_{12} + \pi_{21} + \pi_{22} = 1)$

The likelihood

$$\frac{n!}{n_{11}!n_{12}!n_{21}!n_{22}!}\pi_{11}^{n_{11}}\pi_{12}^{n_{12}}\pi_{21}^{n_{21}}\pi_{22}^{n_{22}}$$

The log likelihood

$$l = \log\frac{n!}{n_{11}!n_{12}!n_{21}!n_{22}!} \qquad \text{(which is not important, why?)}$$
$$+ n_{11}\log\pi_{11} + n_{12}\log\pi_{12} + n_{21}\log\pi_{21} + n_{22}\log\pi_{22}$$

We can choose to write $\pi_{22} = 1 - \pi_{11} - \pi_{12} - \pi_{21}$.

Finding the maximum (setting first derivatives to be zero)

$$\frac{\partial l}{\pi_{11}} = \frac{n_{11}}{\pi_{11}} + \frac{-n_{22}}{1-\pi_{11}-\pi_{12}-\pi_{21}} = 0,$$

$$\Longrightarrow \frac{n_{11}}{\pi_{11}} = \frac{n_{22}}{\pi_{22}} \text{ or written as } \frac{\pi_{11}}{\pi_{22}} = \frac{n_{11}}{n_{22}}$$

Similarly

$$\frac{n_{11}}{\pi_{11}} = \frac{n_{12}}{\pi_{12}} = \frac{n_{21}}{\pi_{21}} = \frac{n_{22}}{\pi_{22}}.$$

Therefore

$$\pi_{11} = n_{11}\lambda, \;\; \pi_{12} = n_{12}\lambda, \;\; \pi_{21} = n_{21}\lambda, \;\; \pi_{22} = n_{22}\lambda,$$

Summing up all the terms

$$1 = \pi_{11} + \pi_{12} + \pi_{21} + \pi_{22} = \left[n_{11} + n_{12} + n_{21} + n_{22}\right]\lambda = n\lambda$$

yields $\lambda = \frac{1}{n}$.

The MLE solution is

$$\hat{\pi}_{11} = \frac{n_{11}}{n}, \quad \hat{\pi}_{12} = \frac{n_{12}}{n}, \quad \hat{\pi}_{21} = \frac{n_{21}}{n}, \quad \hat{\pi}_{22} = \frac{n_{22}}{n}.$$

Finding the MLE Solution by Lagrange Multiplier

MLE as a constrained optimization:

$$\underset{\pi_{11},\pi_{12},\pi_{21},\pi_{22}}{\text{argmax}} \quad n_{11}\log\pi_{11} + n_{12}\log\pi_{12} + n_{21}\log\pi_{21} + n_{22}\log\pi_{22}$$

$$\text{subject to :} \quad \pi_{11} + \pi_{12} + \pi_{21} + \pi_{22} = 1$$

The unconstrained optimization problem:

$$\underset{\pi_{11},\pi_{12},\pi_{21},\pi_{22}}{\text{argmax}} \quad L = n_{11}\log\pi_{11} + n_{12}\log\pi_{12} + n_{21}\log\pi_{21} + n_{22}\log\pi_{22} - \lambda\left(\pi_{11} + \pi_{12} + \pi_{21} + \pi_{22} - 1\right)$$

Finding the optimum: $\frac{\partial L}{\partial z} = 0, \;\; z \in \{\pi_{11}, \pi_{12}, \pi_{21}, \pi_{22}, \lambda\}$

$$\frac{n_{11}}{\pi_{11}} - \lambda = 0, \qquad \frac{n_{12}}{\pi_{12}} = \frac{n_{21}}{\pi_{21}} = \frac{n_{22}}{\pi_{22}} = \lambda, \qquad \pi_{11} + \pi_{12} + \pi_{21} + \pi_{22} = 1$$

Quick Review of Numerical Optimization

Slides 16 - 29 are for reviewing some basic stuff about numerical optimization, which is essential in modern applications.

Maximum Likelihood Estimation (MLE)

Observations x_i, $i = 1$ to n, are i.i.d. samples from a distribution with probability density function $f_X(x; \theta_1, \theta_2, ..., \theta_k)$,
where θ_j, $j = 1$ to k, are parameters to be estimated.

The maximum likelihood estimator seeks the θ to maximize the joint likelihood

$$\hat{\theta} = \underset{\theta}{\text{argmax}} \prod_{i=1}^{n} f_X(x_i; \theta)$$

Or, equivalently, to maximize the log joint likelihood

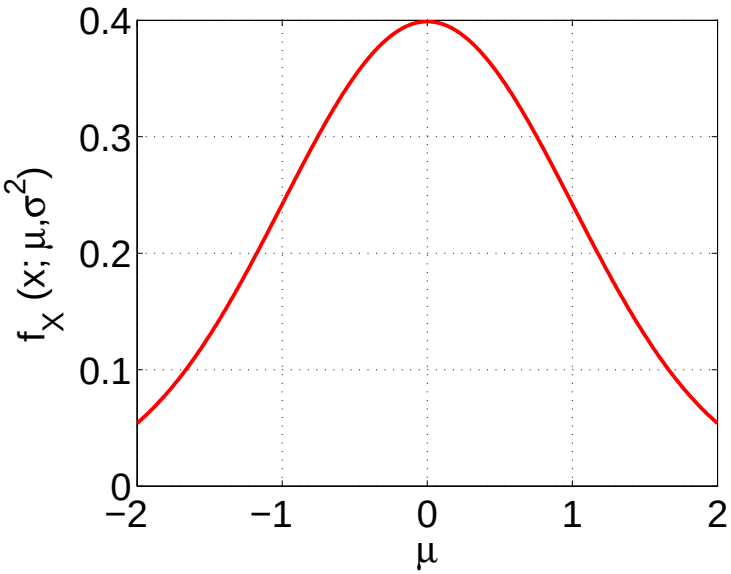
$$\hat{\theta} = \underset{\theta}{\text{argmax}} \sum_{i=1}^{n} \log f_X(x_i; \theta)$$

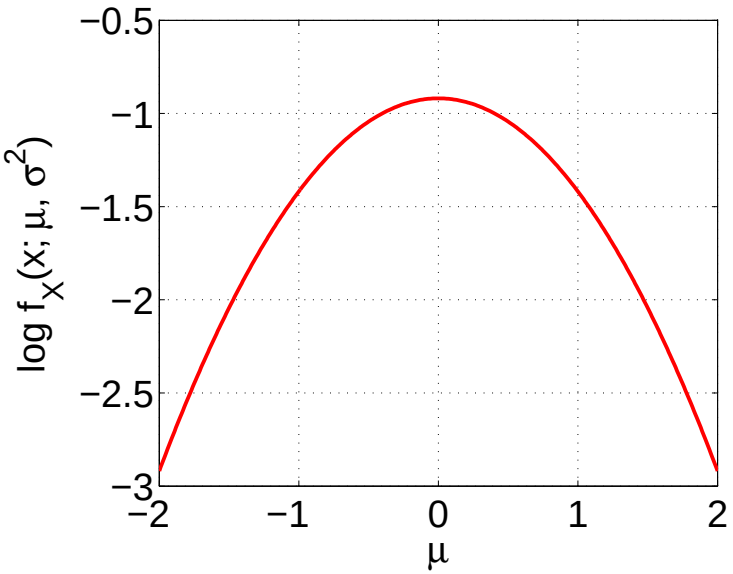
This is a **convex** optimization if f_X is concave or -log-convex.

An Example: Normal Distribution

If $X \sim N\left(\mu, \sigma^2\right)$, then $f_X\left(x; \mu, \sigma^2\right) = \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{(x-\mu)^2}{2\sigma^2}}$

Fix $\sigma^2 = 1, x = 0$. $f_X\left(x; \mu, \sigma^2\right)$ $\log f_X\left(x; \mu, \sigma^2\right)$





It is Not concave, but it is a -log convex, i.e., a unique MLE solution exists.

Another Example of Exact MLE Solution

Given n i.i.d. samples, $x_i \sim N(\mu, \sigma^2)$, $i = 1$ to n.

$$l\left(x_1, x_2, ..., x_n; \mu, \sigma^2\right) = \sum_{i=1}^{n} \log f_X(x_i; \mu, \sigma^2)$$

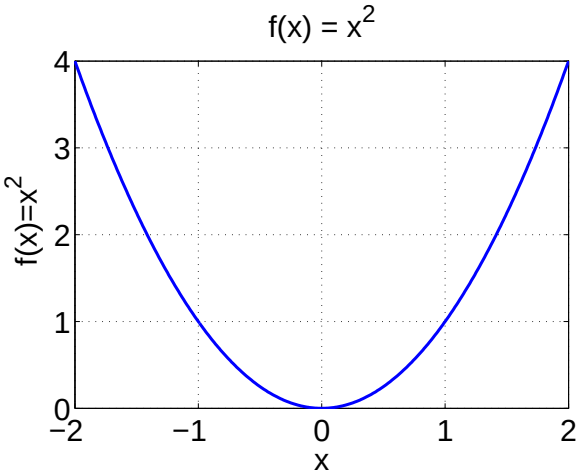
$$= -\frac{1}{2\sigma^2} \sum_{i=1}^{n} (x_i - \mu)^2 - \frac{1}{2} n \log(2\pi\sigma^2)$$

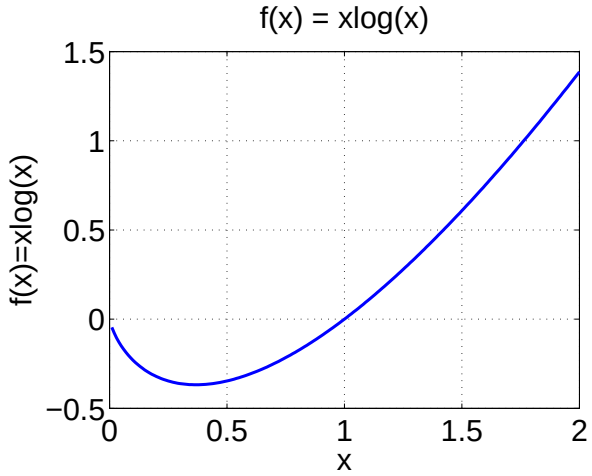
$$\frac{\partial l}{\partial \mu} = \frac{1}{2\sigma^2} 2 \sum_{i=1}^{n} (x_i - \mu) = 0 \Longrightarrow \hat{\mu} = \frac{1}{n} \sum_{i=1}^{n} x_i$$

$$\frac{\partial l}{\partial \sigma^2} = \frac{1}{2\sigma^4} \sum_{i=1}^{n} (x_i - \mu)^2 - \frac{n}{2\sigma^2} = 0 \Longrightarrow \hat{\sigma^2} = \frac{1}{n} \sum_{i=1}^{n} (x_i - \hat{\mu})^2.$$

Convex Functions

A function $f(x)$ is convex if the second derivative $f''(x) \geq 0$.





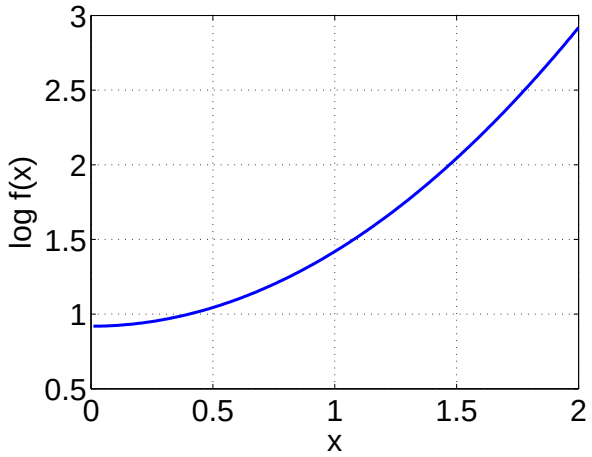
$f(x) = x^2 \Longrightarrow f'' = 2 > 0$, i.e., $f(x) = x^2$ is convex for all x.

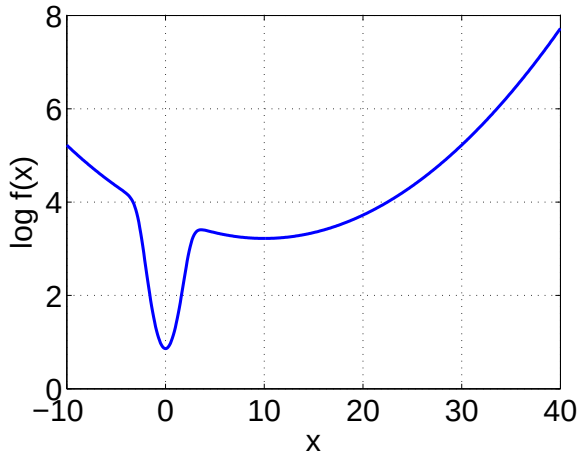
$f(x) = x \log x \Longrightarrow f'' = \frac{1}{x}$, i.e., $f(x) = x \log x$ is convex if $x > 0$.

Both are widely used in statistics and data mining as loss functions,

$\Longrightarrow$ computationally tractable algorithms: least square, logistic regression.

Left panel: $f(x) = \frac{1}{\sqrt{2\pi}} e^{-\frac{x^2}{2}}$ is -log convex, $\frac{\partial^2[-\log f(x)]}{\partial x^2} = 1 > 0.$





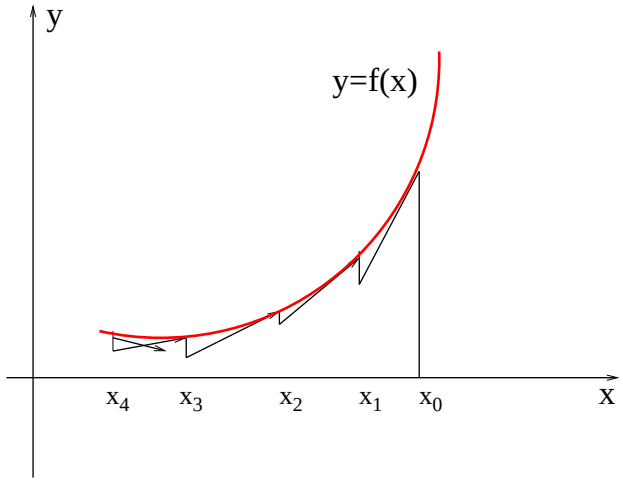
Right panel: a mixture of normals is not -log convex

$$f(x) = \frac{1}{\sqrt{2\pi}} e^{-\frac{x^2}{2}} + \frac{1}{\sqrt{2\pi}10} e^{-\frac{(x-10)^2}{200}}$$

The mixture of normals is an extremely useful model in statistics.

In general, only a local minimum can be obtained.

Steepest Descent



Procedure:

Start with an initial guess x_0.

Compute $x_1 = x_0 - \Delta f'(x_0)$, where Δ is the step size.

Continue the process $x_{t+1} = x_t - \Delta f'(x_t)$.

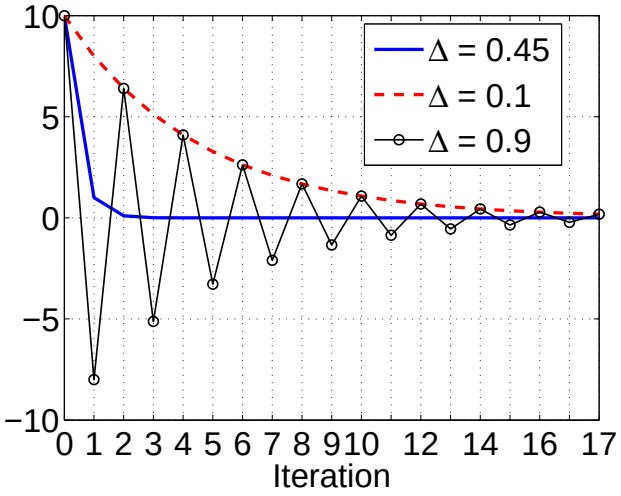
Until some criterion is met, e.g., $f(x_{t+1}) \approx f(x_t)$

The meaning of "steepest" is more clear in the two-dimensional situation.

An Example of Steepest Descent: $f(x) = x^2$

$f(x) = x^2$. The minimum is attained at $x = 0$, $f'(x) = 2x$.

The steepest descent iteration formula $x_{t+1} = x_t - \Delta f'(x_t) = x_t - 2\Delta x_t$.



Choosing the step size Δ is important (even when $f(x)$ is convex).

Too small $\Delta \Longrightarrow$ slow convergence, i.e., many iterations,

Too large $\Delta \Longrightarrow$ oscillations, i.e., also many iterations.

Steepest Descent in Practice

Steepest descent is one of the most widely techniques in real world

- It is extremely simple
- It only requires knowing the first derivative
- It is numerically stable (for above reasons)
- For real applications, it is often affordable to use very small Δ
- In machine learning, Δ is often called learning rate
- It is used in Neural Nets and Gradient Boosting (MART)

Newton's Method

Recall the goal is to find the x^* to minimize $f(x)$.

If $f(x)$ is convex, it is equivalent to finding the x^* such that $f'(x^*) = 0$.

Let $h(x) = f'(x)$. Take Taylor expansion about the optimum solution x^*:

$$h(x^*) = h(x) + (x^* - x)h'(x) + \text{“negligible” higher order terms}$$

Because $h(x^*) = f'(x^*) = 0$, we know approximately

$$0 \approx h(x) + (x^* - x)h'(x) \Longrightarrow x^* \approx x - \frac{h(x)}{h'(x)}$$

The procedure of Newton's Method

Start with an initial guess x_0

Update $x_1 = x_0 - \frac{f'(x_0)}{f''(x_0)}$

Repeat $x_{t+1} = x_t - \frac{f'(x_t)}{f''(x_t)}$

Until some stopping criterion is reached, e.g., $x_{t+1} \approx x_t$.

An example: $f(x) = (x-c)^2$. $f'(x) = 2(x-c)$, $f''(x) = 2$.

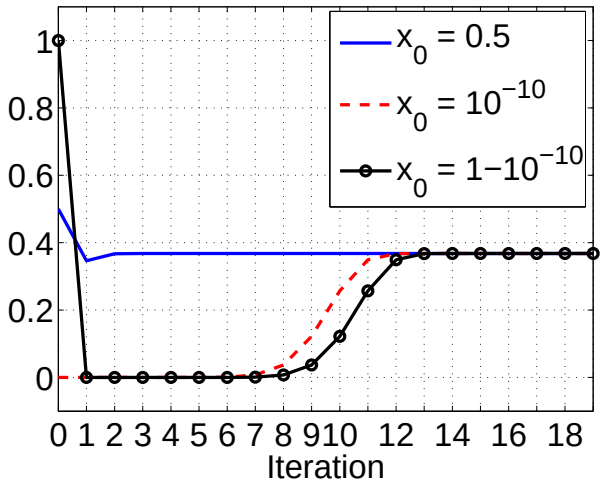
$$x_1 = x_0 - \frac{f'(x_0)}{f''(x_0)} \Longrightarrow x_1 = x_0 - \frac{2(x_0-c)}{2} = c$$

But we already know that $x = c$ minimizes $f(x) = (x-c)^2$.

Newton's method may find the minimum solution using only one step.

An Example of Newton's Method: $f(x) = x \log x$

$$f'(x) = \log x + 1, \qquad f''(x) = \frac{1}{x}. \qquad x_{t+1} = x_t - \frac{\log x_t + 1}{1/x_t}$$



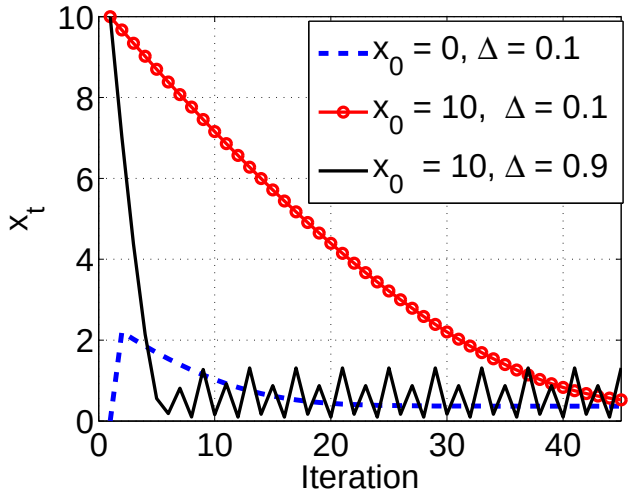
When x_0 is close to optimum solution, the convergence is very fast

When x_0 is far from the optimum, the convergence is slow initially

When x_0 is badly chosen, no convergence. This example requires $0 < x_0 < 1$.

Steepest Descent for $f(x) = x \log x$

$$f'(x) = \log x + 1, \qquad x_{t+1} = x_t - \Delta(\log x_t + 1)$$



Regardless of x_0, convergence is guaranteed if $f(x)$ is convex.

May be oscillating if step size Δ is too large

Convergence is slow near the optimum solution.

General Comments on Numerical Optimization

Numerical Optimization is tricky!, even for convex problems.

Multivariate optimization is much trickier!

Whenever possible, try to avoid intensive numerical optimization,
even maybe at the cost of losing some accuracy.

An example:
Iterative Proportional Scaling for contingency table with known margins

Contingency Table with Margin Constraints

Original Contingency Table

N_{11}	N_{12}
N_{21}	N_{22}

Sample Contingency Table

n_{11}	n_{12}
n_{21}	n_{22}

Margins: $M_1 = N_{11} + N_{12}, \qquad M_2 = N_{11} + N_{21}.$

Margins are much easier to be counted exactly than interactions.

An Example of Contingency Tables with Known Margins

Term-by-Document matrix $n = 10^6$ words and $m = 10^{10}$ (Web) documents.

Cell $x_{ij} = 1$ if word i appears in document j. $x_{ij} = 0$ otherwise.

	Doc 1	Doc 2					Doc m
Word 1	1	0	0	1	0	0	0	1
Word 2	0	1	0	1	0	0	1	0
Word 3								
Word 4								
Word n								

	Word 2	No Word 2
Word 1	N_{11}	N_{12}
No Word 1	N_{21}	N_{22}

N_{11}: number of documents containing both word 1 and word 2.

N_{22}: number of documents containing neither word 1 nor word 2.

Margins ($M_1 = N_{11} + N_{12}$, $M_2 = N_{11} + N_{21}$) for all rows costs nm, easy!

Interactions (N_{11}, N_{12}, N_{21}, N_{22}) for all pairs costs $n(n-1)m/2$, difficult!.

To avoid storing all pairwise contingency tables ($n(n-1)/2$ pairs in total), one strategy is to sample a fraction (k) of the columns of the original (term-doc) data matrix and and build sample contingency tables on demand, from which one can estimate the original contingency tables.

However, we observe that the margins (the total number of ones in each row) can be easily counted. This naturally leads to the conjecture that one might (often considerably) improves the estimation accuracy by taking advantage of the known margins. The next question is how.

Two approaches:

1. Maximum likelihood estimator (MLE) accurate but fairly complicated.
2. Iterative proportional scaling (IPS) simple but usually not as accurate.

An Example of IPS for 2 by 2 Tables

n_{11}	n_{12}
n_{21}	n_{22}

The steps of IPS

(1) Modify the counts to satisfy the row margins.

(2) Modify the counts to satisfy the column margins.

(3) Iterate until some stopping criterion is met.

An example: $n_{11} = 30, n_{12} = 5, n_{21} = 10, n_{22} = 10,\ D = 600.$

$M_1 = N_{11} + N_{12} = 400, \qquad M_2 = N_{11} + N_{21} = 300.$

In the first iteration: $N_{11} \leftarrow \frac{M_1}{n_{11}+n_{12}} n_{11} = \frac{400}{35} 30 = 342.8571.$

```
Iteration 1
  342.8571   57.1429
  100.0000  100.0000

  232.2581  109.0909
   67.7419  190.9091
Iteration 2
  272.1649  127.8351
   52.3810  147.6190

  251.5807  139.2265
   48.4193  160.7735
Iteration 3
  257.4985  142.5015
   46.2916  153.7084

  254.2860  144.3248
   45.7140  155.6752
```

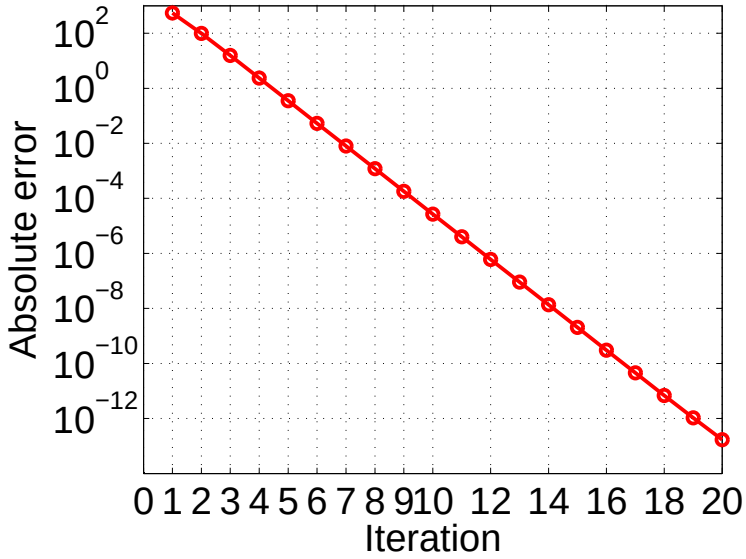
```
Iteration 4
  255.1722  144.8278
   45.3987  154.6013

  254.6875  145.1039
   45.3125  154.8961
Iteration 5
  254.8204  145.1796
   45.2653  154.7347

  254.7477  145.2211
   45.2523  154.7789
Iteration 6
 254.7676  145.2324
   45.2453  154.7547

  254.7567  145.2386
   45.2433  154.7614
```

Error = $\left|\text{current step - previous step counts}\right|$, sum over four cells.



IPS converges fast and it always converges.

But how good are the estimates?: My general observation is that it is very good for 2 by 2 tables and the accuracy decreases (compared to the MLE) as the table size increases.

The MLE for 2 by 2 Table with Known Margins

Total samples : $n = n_{11} + n_{12} + n_{21} + n_{22}$

Total original counts : $N = N_{11} + N_{12} + N_{21} + N_{22}$, i.e., $\pi_{ij} = N_{ij}/N$.

Sample Contingency Table

n_{11}	n_{12}
n_{21}	n_{22}

Original Contingency Table

N_{11}	N_{12}
N_{21}	N_{22}

Margins: $M_1 = N_{11} + N_{12}$, $M_2 = N_{11} + N_{21}$.

If margins M_1 and M_2 are known, then only need to estimate N_{11}.

The likelihood

$$\propto \left(\frac{N_{11}}{N}\right)^{n_{11}} \left(\frac{N_{12}}{N}\right)^{n_{12}} \left(\frac{N_{21}}{N}\right)^{n_{21}} \left(\frac{N_{22}}{N}\right)^{n_{22}}$$

The log likelihood

$$n_{11}\log\left(\frac{N_{11}}{N}\right) + n_{12}\log\left(\frac{N_{12}}{N}\right) + n_{21}\log\left(\frac{N_{21}}{N}\right) + n_{22}\log\left(\frac{N_{22}}{N}\right)$$

$$= n_{11}\log\left(\frac{N_{11}}{N}\right) + n_{12}\log\left(\frac{M_1 - N_{11}}{N}\right) + n_{21}\log\left(\frac{M_2 - N_{11}}{N}\right)$$

$$+ n_{22}\log\left(\frac{N - M_1 - M_2 + N_{11}}{N}\right)$$

The MLE equation

$$\frac{n_{11}}{N_{11}} - \frac{n_{12}}{M_1 - N_{11}} - \frac{n_{21}}{M_2 - N_{11}} + \frac{n_{22}}{N - M_1 - M_2 + N_{11}} = 0.$$

which is a cubic equation and can be solved either analytically or numerically.

Error Analysis

To assess the quality of the estimator $\hat{\theta}$ of θ, it is common to use bias, variance, and MSE (mean square error):

Bias : $E(\hat{\theta}) - \theta$

Var : $E\left(\hat{\theta} - E(\hat{\theta})\right)^2 = E(\hat{\theta}^2) - E^2(\hat{\theta})$

MSE : $E\left(\hat{\theta} - \theta\right)^2 = Var + Bias^2$

The last equality is known as the bias variance trade-off. For unbiased estimators, it is desirable to have smaller variance as possible. As the sample size increases, the MLE (under certain conditions) becomes unbiased and achieves the smallest variance. Therefore, the MLE is often a desirable estimator. However, in some cases, biased estimators may achieve smaller MSE than the MLE.

The Expectations and Variances of Common Distributions

The derivations of variances are not required in this course. Nevertheless, it is useful to know the expectations and variances of common distributions.

- **Binomial**: $X \sim binomail(n, p)$, $E(X) = np$, $Var(X) = np(1-p)$.
- **Normal**: $X \sim N(\mu, \sigma^2)$, $E(X) = \mu$, $Var(X) = \sigma^2$.
- **Chi-square**: $X \sim \chi^2(k)$, $E(X) = k$, $Var(X) = 2k$.
- **Exponential**: $X \sim exp(\lambda)$, $E(X) = \frac{1}{\lambda}$, $Var(X) = \frac{1}{\lambda^2}$.
- **Poisson**: $X \sim Pois(\lambda)$, $E(X) = \lambda$, $Var(\lambda)$.

Multinomial Distribution

The multinomial is a natural extension to the binomial distribution. For example, the 2 by 2 contingency table often assumes to follow the multinomial distribution.

Consider c cells and denote the observations by $(n_1, n_2, ..., n_c)$, which follow a c-cell multinomial distribution with the underlying probabilities $(\pi_1, \pi_2, ..., \pi_c)$ (with $\sum_{i=1}^{c} \pi_i = 1$). Denote $n = \sum_{i=1}^{c} n_i$. We write

$$(n_1, n_2, ..., n_c) \sim Multinomial\,(n, \pi_1, \pi_2, ..., \pi_c)$$

The expectations are (for $i = 1$ to c and $i \neq j$)

$$E\,(n_i) = n\pi_i, \quad Var\,(n_i) = n\pi_i(1 - \pi_i), \quad Cov\,(n_i n_j) = -n\pi_i\pi_j.$$

Note that the cells are negatively correlated (why?).

Variances of the 2 by 2 Contingency Table Estimates

Using previous notation, the MLE estimator of N_{11} is

$$\hat{N}_{11} = \frac{n_{11}}{n} N, \quad (n_{11}, n_{12}, n_{21}, n_{22}) \sim Multinomial(n, \pi_{11}, \pi_{12}, \pi_{21}, \pi_{22})$$

Using the general equalities about the expectations:

$$E(aX) = aE(X), \qquad Var(aX) = a^2 Var(X)$$

we know

$$E\left(\hat{N}_{11}\right) = \frac{N}{n} E(n_{11}) = \frac{N}{n} n\pi_{11} = N\pi_{11} = N_{11}$$

$$Var\left(\hat{N}_{11}\right) = \frac{N^2}{n^2} Var(n_{11}) = \frac{N^2}{n^2} n\pi_{11}(1-\pi_{11}) = \frac{N^2}{n}\pi_{11}(1-\pi_{11})$$

The Asymptotic Variance of the MLE Using Margins

When the margins are known: $M_1 = N_{11} + N_{12}, \qquad M_2 = N_{12} + N_{21}$

The MLE equation

$$\frac{n_{11}}{N_{11}} - \frac{n_{12}}{M_1 - N_{11}} - \frac{n_{21}}{M_2 - N_{11}} + \frac{n_{22}}{N - M_1 - M_2 + N_{11}} = 0.$$

The asymptotic variance of the solution, denoted by $\hat{N}_{11,M}$, can be shown to be

$$Var\left(\hat{N}_{11,M}\right) = \frac{N}{n} \frac{1}{\frac{1}{N_{11}} + \frac{1}{N_{12}} + \frac{1}{N_{21}} + \frac{1}{N_{22}}}$$

which is smaller than the variance of the MLE without using margins.

———-

What about the variance of IPS? : No closed-form answer and the estimates are usually biased.

Statistical Testing of Independence of Contingency Tables

In general, a contingency table can be described by its underlying probabilities for random variables X and Y:

$$\pi_{ij} = \mathbf{Pr}\left(X = i, Y = j\right), \qquad \sum_{i,j} \pi_{ij} = 1$$

The observations n_{ij} from a sample follows the multinomial distribution if $\sum_{ij} n_{ij} = n$ is fixed; we usually denote the sample proportion as

$$\hat{\pi}_{ij} = \frac{n_{ij}}{n}, \qquad \text{and hence } \sum_{i,j} \hat{\pi}_{ij} = 1$$

And we know from previous lectures that $\hat{\pi}_{ij}$ is an unbiased estimator of π_{ij} and it is the MLE. Now the question is whether the counts n_{ij}'s are purely due to random chance or due to the dependence between X and Y.

Marginal Probabilities, Marginal Observations, Independence

Marginal Probabilities:

$$\pi_{i+} = \mathbf{Pr}\left(X = i\right) = \sum_j \mathbf{Pr}\left(X = i, Y = j\right) = \sum_j \pi_{ij}$$

$$\pi_{+j} = \mathbf{Pr}\left(Y = j\right) = \sum_i \mathbf{Pr}\left(X = i, Y = j\right) = \sum_i \pi_{ij}$$

Marginal Observations:

$$n_{i+} = \sum_j n_{ij}, \qquad n_{+j} = \sum_i n_{ij}$$

Independence: If

$$\mathbf{Pr}\left(X = i | Y = j\right) = \mathbf{Pr}\left(X = i\right)$$

then we say X and Y are independent.

Consequence of Independence

If X and Y are independent, then basic fact is that

$$\mathbf{Pr}(X = i, Y = j) = \mathbf{Pr}(X = i) \times \mathbf{Pr}(Y = j)$$

i.e., $\pi_{ij} = \pi_{i+} \times \pi_{+j}$

In general, if X and Y are independent, then $\mathbf{Pr}(X = i, Y = j) \neq 0$.

An interesting consequence of the independence assumption.

If the margins the original tables are known, for example, M_1, M_2. Then we can estimate the counts without using samples, for example, $\hat{N}_{11,IND} = \frac{M_1 M_2}{N}$.

This is widely used in practice due to its simplicity.

Hypothesis Testing of Independence

Consider a contingency table: π_{ij}, $i = 1$ to I, and $j = 1$ to J.

We observe: n_{ij}, $i = 1$ to I, and $j = 1$ to J.

Assuming independence, then $\pi_{ij} = \pi_{i+}\pi_{+j}$ and we expect that $n_{ij} = n\hat{\pi}_{i+}\hat{\pi}_{+j}$. We use a special notation $\hat{\mu}_{ij}$ = $n\hat{\pi}_{i+}\hat{\pi}_{+j}$.

The task is to test the null hypothesis:

$$H_0 : \quad \pi_{ij} = \pi_{i+}\pi_{+j}, \qquad \text{for all } i \text{ and } j.$$

The fundamental tool is the likelihood ratio statistic.

The Likelihood Ratio Statistic

It is defined as

$$-2\log\left(\frac{\text{Maximum likelihood under } H_0}{\text{Maximum likelihood with no restriction}}\right)$$

which is asymptotically distributed as χ^2_k with k determined by the degree of freedom (red df):

$$df = \text{number of parameters to be estimated without restrictions}$$
$$- \quad \text{number of parameters to be estimated under } H_0$$

This result can be derived by large-sample theory.

The Likelihood Ratio Statistic For Contingency Tables

Maximum likelihood under H_0:

$$\prod_{i,j}\left[\frac{\hat{\mu}_{ij}}{n}\right]^{n_{ij}}$$

Maximum likelihood without restrictions:

$$\prod_{i,j}\left[\frac{n_{ij}}{n}\right]^{n_{ij}}$$

Likelihood Ratio Statistic:

$$\begin{aligned}-2\log\frac{\prod_{i,j}\left[\frac{\hat{\mu}_{ij}}{n}\right]^{n_{ij}}}{\prod_{i,j}\left[\frac{n_{ij}}{n}\right]^{n_{ij}}} &= 2\sum_{i,j} n_{ij}\log\frac{n_{ij}}{n} - 2\sum_{i,j} n_{ij}\log\frac{\hat{\mu}_{ij}}{n} \\ &= 2\sum_{i,j} n_{ij}\log\frac{n_{ij}}{\hat{\mu}_{ij}} = G^2.\end{aligned}$$

Degree of freedom:

$$df = [I \times J - 1] - [(I-1) + (J-1)] = (I-1)(J-1)$$

The Chi-Square Statistic:

$$X^2 = \sum_{i,j} \frac{(n_{ij} - \hat{\mu}_{ij})^2}{\hat{\mu}_{ij}}$$

which is asymptotically equivalent to G^2 and can be derived by a Taylor expansion of G^2.

Both statistics are very popular and their numerical values are usually very close.

An Example of Testing of Independence: Book 2.4.4

Cross Classification of Party Identification by Gender

Gender	Democrat	Independent	Republican	Total
Female	762	327	468	1557
Male	484	239	477	1200
Total	1246	566	945	2757

A 2×3 contingency table with $I = 2$ and $J = 3$. The sample margins are

$$n_{1+} = 762 + 327 + 468 = 1557, \quad n_{2+} = 484 + 239 + 477 = 1200$$

$$n_{+1} = 762 + 484 = 1246, \quad n_{+2} = 327 + 239 = 566, \quad n_{+3} = 945$$

Task: Test whether the cells are independent.

Expected counts under H_0: independence: $\hat{\mu}_{ij} = n_{i+} \times n_{+j}/n$

Gender	Democrat	Independent	Republican	Total
Female	762 (703.7)	327 (319.6)	468 (533.7)	1557
Male	484 (542.3)	239 (246.4)	477 (411.3)	1200
Total	1246	566	945	2757

G^2 test statistic: $G^2 = 2\sum_{ij} n_{ij} \log \frac{n_{ij}}{\hat{\mu}_{ij}} = 30.0$

X^2 test statistic: $X^2 = \sum_{ij} \frac{(n_{ij}-\hat{\mu}_{ij})^2}{\hat{\mu}_{ij}} = 30.1$

Degree of freedom: $df = (I-1)(J-1) = 2.$

Accept or Reject H_0?

Both G^2 and X^2 are asymptotically $\chi^2_{df} = \chi^2_2$. Thus, we check the cumulative probability of χ^2 to compute the p-value:

$$\mathbf{Pr}\left(G^2 > 30.0\right) = 3.059 \times 10^{-7}$$

$$\mathbf{Pr}\left(X^2 > 30.1\right) = 2.910 \times 10^{-7}$$

Both p-values are extremely small $\ll 0.05$. Therefore, we reject the null hypothesis H_0 that the cells are independent.

A Review of Testing Hypothesis

Suppose you have a coin which is possibly biased. You want to test whether the coin is indeed biased (i.e., $p \neq 0.5$), by tossing the coin $n = 10$ times.

Suppose you observe $k = 8$ heads (out of $n = 10$ tosses). It is reasonable to guess that this coin is indeed biased. But how to make a precise statement?

Are $n = 10$ tosses enough? How about $n = 100$? $n = 1000$? What is the principled approach?

Terminology

Null hypothesis $H_0 : p = 0.5$

Alternative hypothesis $H_A : p \neq 0.5$

Type I error Rejecting H_0 when it is true

Significance level $P(\text{Type I error}) = P(\text{Reject } H_0 | H_0) = \alpha$

Type II error Accepting H_0 when it is false

$P(\text{Type II error}) = P(\text{Accept } H_0 | H_A) = \beta$

Power $1 - \beta$

Goal: Low α and high $1 - \beta$.

Example: Let $X_1, X_2, ..., X_n$ be an i.i.d. sample from a normal with known variance σ^2 and unknown mean μ. Consider two **simple hypotheses**:

$$H_0 : \mu = \mu_0$$
$$H_A : \mu = \mu_1 \qquad (\mu_1 > \mu_0)$$

Under H_0, the **null distribution** likelihood is

$$f_0 \propto \prod_{i=1}^{n} \exp\left[-\frac{1}{2\sigma^2}(X_i - \mu_0)^2\right] = \exp\left[-\frac{1}{2\sigma^2}\sum_{i=1}^{n}(X_i - \mu_0)^2\right]$$

Under H_A, the likelihood is

$$f_1 \propto \exp\left[-\frac{1}{2\sigma^2}\sum_{i=1}^{n}(X_i - \mu_1)^2\right]$$

Which hypothesis is more likely?

Neyman-Pearson Lemma: Among all possible tests achieving significance level $\leq \alpha$, the test based on likelihood ratio maximizes the power.

Likelihood ratio test: $\frac{f_0}{f_1} \leq c \Longrightarrow$ Reject H_0.

$$\frac{f_0}{f_1} = \exp\left[\frac{n}{2\sigma^2}\left[2\bar{X}(\mu_0 - \mu_1) + \mu_1^2 - \mu_0^2\right]\right] \leq c$$

$$\alpha = P(\text{reject } H_0 | H_0) = P(f_0 \leq c f_1 | H_0)$$

Equivalently, reject H_0 if the sample mean $\bar{X}$ is too large:

Reject H_0 if $\bar{X} \geq x_0$, and

$$P(\bar{X} \geq x_0 | H_0) = \alpha.$$

Under H_0: $\bar{X} \sim N\left(\mu_0, \sigma^2/n\right)$

$$\begin{aligned} \alpha =& P(\bar{X} \geq x_0 | H_0) \\ \Longrightarrow x_0 =& \mu_0 + z_\alpha \frac{\sigma}{\sqrt{n}} \end{aligned}$$

z_α is the upper α point of the standard normal:

$P(Z \geq z_\alpha) = \alpha$, where $Z \sim N(0,1)$. $z_{0.05} = 1.645,\ z_{0.025} = 1.960$

Therefore, the test rejects H_0 if $\bar{X} \geq \mu_0 + z_\alpha \frac{\sigma}{\sqrt{n}}$.

P-Value

Definition: The p-value is the smallest significance level at which the null hypothesis would be rejected.

The smaller the p-value, the stronger the evidence against the null hypothesis.

In a sense, calculating the p-value is more sensible than specifying (often arbitrarily) the level of significance α.

Composite Test

Neyman-Pearson Lemma requires that both hypotheses be simple. However, most real-situations require composite hypothesis.

Examples:

$$H_0 : \mu = \mu_0$$
$$H_1 : \mu > \mu_0$$

$$H_0 : \mu < \mu_0$$
$$H_1 : \mu > \mu_0$$

$$H_0 : \mu = \mu_0$$
$$H_1 : \mu \neq \mu_0$$

Generalized Likelihood Ratio Test

Likelihood ratio test:

A simple hypothesis versus a simple hypothesis. Optimal. Very limited use.

Generalized likelihood ratio test: The one we have used

Composite hypotheses. Sub-optimal and widely-used.

Play the same role as MLE in parameter estimation.

Assume a sample X_1, ... ,X_n from a distribution with unknown parameter θ.

$$H_0 : \theta \in \omega_0$$

$$H_A : \theta \in \omega_1$$

Let $\Omega = \omega_0 \cup \omega_1$. **The test statistic**

$$\Lambda = \frac{\max\limits_{\theta \in \omega_0} lik(\theta)}{\max\limits_{\theta \in \Omega} lik(\theta)}$$

Reject H_0 if $\Lambda \leq \lambda_0$, such that

$$P(\Lambda \leq \lambda_0 | H_0) = \alpha$$

Theorem: Under some smoothness conditions on the probability density of mass functions, the null distribution of $-2\log\Lambda$ tends to a chi-square distribution with degrees of freedom equal to dimΩ− dimω_0, as the sample size tends to infinity.

dimΩ = number of free parameters under Ω

dimω_0 = number of free parameters under ω_0.

The Odds and Odds Ratio

Consider a 2×2 table: $\pi_{ij}, i, j \in \{1, 2\}$. Define

$$\textbf{Odds} = \theta = \frac{\text{Odds}_1}{\text{Odds}_2} = \frac{\pi_{11}/\pi_{12}}{\pi_{21}/\pi_{22}}$$

It is called "odds" because we (by convention) treat π_{11} and π_{21} as "success" probabilities.

The ratio θ provides another measure of the association of the contingency table. When the counts are independent, then we would expect that

$$\text{odds}_1 = \text{odds}_2, \qquad i.e., \qquad \theta = 1$$

A natural estimate of θ is just

$$\hat{\theta} = \frac{n_{11}/n_{12}}{n_{21}/n_{22}}$$

An Example of Odds Ratio

Cross Classification of Aspirin Use and Myocardinal Infarction

Group	Infarction YES	Infarction NO	Total
Placebo	189	10845	11034
Aspirin	104	10933	11037

Empirical estimates:

$$\text{Odds}_1 = 189/10845 = 0.0174, \qquad \text{Odds}_2 = 104/10933 = 0.0095,$$

$$\text{Odds Ratio} = 0.0174/0.0095 = 1.832$$

Should we reject the null hypothesis of independence? Need to do a test, either the (generalized) likelihood ratio test (for large samples) or Fisher's exact test (for small samples).

Small Sample Test: Fisher's Exact Test

Both G^2 test and X^2 test make the large-sample assumption because they rely on the asymptotic result. When the sample size is small, one might be able to conduct the "exact" test.

Consider a 2×2 table: $n_{ij}, i, j \in \{1, 2\}$.

Under the null hypothesis H_0 that the cell counts are independent, the probability is

$$\mathbf{Pr}\left(n_{11}\right) = \frac{\binom{n_{1+}}{n_{11}}\binom{n_{2+}}{n_{+1}-n_{11}}}{\binom{n}{n_{+1}}}, \quad 0 \leq n_{11} \leq \min(n_{1+}, n_{+1})$$

How do we understand this probability?

Fisher's Tea Taster

Fisher's Tea Tasting Experiment

Poured First	Milk	Tea	Total
Milk	3	1	4
Tea	1	3	4
Total	4	4	

To test a colleague's claim that she could distinguish whether milk or tea was added to the cup first, Fisher designed this test by asking her to drink 8 cups of teas, four cups had milk added first and the other four had tea added first.

Fisher's Tea Taster

In this example, the test is one-sided (for positive association)

$$H_0: \quad \theta = 1$$

$$H_A: \quad \theta > 1$$

Under H_0, the probability of n_{11} is

$$\mathbf{Pr}\left(n_{11}\right) = \frac{\binom{4}{n_{11}}\binom{4}{4-n_{11}}}{\binom{8}{4}}$$

and the corresponding (one-sided) p-value is

$$\sum_{i=n_{11}}^{4} \mathbf{Pr}\left(i\right)$$

Fisher's Tea Taster

Fisher's Tea Tasting Experiment

n_{11}	$\mathbf{Pr}(n_{11})$	one-sided p-value	X^2
0	0.014	1.000	8.0
1	0.229	0.986	2.0
2	0.514	0.757	0.0
3	0.229	0.243	2.0
4	0.014	0.014	8.0

One potential issue is that the values are very much discontinuous, the nature of the small sample problem. However, when the sample size is large, another issue arises. What is it?

More General Contingency Table Problems

It is actually much more common that the underlying probabilities of the cells are functions of some parameters.

For example, a multinomial distribution

$$(n_1, n_2, ..., n_c) = \text{multinomial}\,(p_1, p_2, ..., p_n)$$

where

$$p_i = p_i(\theta), \quad i = 1, 2, ..., c, \qquad \text{and} \qquad \sum_{i=1}^{c} p_i(\theta) = 1$$

θ can be one scalar parameter, or a vector of parameters.

Hardy-Weinberg Equilibrium

If gene frequencies are in equilibrium, the genotypes AA, Aa, and aa occur in a population with frequencies:

$$\pi_1 = (1-\theta)^2, \qquad \pi_2 = 2\theta(1-\theta), \qquad \pi_3 = \theta^2,$$

respectively. Suppose we observe sample counts n_1, n_2, and n_3, with total $= n$.

The task is to estimate θ (e.g., using MLE).

The MLE solution: The log likelihood can be written as

$$\begin{aligned} l(\theta) &= \sum_{i=1}^{3} n_i \log \pi_i \\ &= n_1 \log(1-\theta)^2 + n_2 \log 2\theta(1-\theta) + n_3 \log \theta^2 \\ &\propto 2n_1 \log(1-\theta) + n_2 \log \theta + n_2 \log(1-\theta) + 2n_3 \log \theta \\ &= (2n_1 + n_2) \log(1-\theta) + (n_2 + 2n_3) \log \theta \end{aligned}$$

Taking the first derivative

$$\frac{\partial l(\theta)}{\partial \theta} = -\frac{2n_1 + n_2}{1-\theta} + \frac{n_2 + 2n_3}{\theta} = 0$$

$$\Longrightarrow \hat{\theta} = \frac{2n_3 + n_2}{2n}$$

It can be shown by large-sample theory that $Var(\hat{\theta}) = \frac{\theta(1-\theta)}{2n}$.

Testing The Hardy-Weinberg Equilibrium Model

In an experiment, the cell counts are 342, 500, and 187 ($n = 1029$).

Using MLE, we estimate $\hat{\theta} = \frac{2n_3+n_2}{2n} = 0.4246842$.

The expected (estimated) counts are 340.6, 502.8, and 185.6, respectively.

Now we want to test H_0: the data follow the Hardy-Weinberg Model.

Generalized Likelihood Ratio Tests for Multinomial Distribution

Goodness of fit: Assume the multinomial probabilities p_i are specified by

$$H_0 : \ p = p(\theta), \qquad \theta \in \omega_0$$

where θ is a (vector of) parameter(s) to be estimated.

We need to know whether the model $p(\theta)$ is good or not, according to the observed data (cell counts).

We also need an alternative hypothesis. A common choice of Ω would be

$$\Omega = \{p_i, i = 1, 2, ..., m | p_i \geq 0, \ \sum_{i=1}^{m} p_i = 1\}$$

$$\Lambda = \frac{\max_{p\in\omega_0} lik(p)}{\max_{p\in\Omega} lik(p)}$$

$$= \frac{\binom{n}{x_1,x_2,\ldots,x_m} p_1(\hat{\theta})^{x_1} \ldots p_m(\hat{\theta})^{x_m}}{\binom{n}{x_1,x_2,\ldots,x_m} \hat{p}_1^{x_1} \ldots \hat{p}_m^{x_m}}$$

$$= \prod_{i=1}^{m} \left(\frac{p_i(\hat{\theta})}{\hat{p}_i} \right)^{x_i}$$

$\hat{\theta}$: the MLE under ω_0 $\qquad$ $\hat{p}_i = \frac{x_i}{n}$: the MLE under Ω.

$$\Lambda = \prod_{i=1}^{m} \left(\frac{p_i(\hat{\theta})}{\hat{p}_i} \right)^{n\hat{p}_i}, \qquad -2\log\Lambda = -2n \sum_{i=1}^{m} \hat{p}_i \log \left(\frac{p_i(\hat{\theta})}{\hat{p}_i} \right)$$

$$
\begin{aligned}
-2\log\Lambda =& -2n\sum_{i=1}^{m}\hat{p}_i\log\left(\frac{p_i(\hat{\theta})}{\hat{p}_i}\right)\\
=& 2\sum_{i=1}^{m} n\hat{p}_i\log\left(\frac{n\hat{p}_i}{np_i(\hat{\theta})}\right)\\
=& 2\sum_{i=1}^{m} O_i\log\frac{O_i}{E_i}
\end{aligned}
$$

$O_i = n\hat{p}_i = x_i$: the observed counts,

$E_i = np_i(\hat{\theta})$: the expected counts

$-2\log\Lambda$ is asymptotically χ^2_s.

The degrees of freedom $s = \dim\Omega - \dim\omega_0 = (m-1) - k$.

k = length of the vector θ = number of parameters in the model.

Continue the Example of Testing the Hardy-Weinberg Model

Using the count data, we can compute two test statistics to be

$$G^2 = 0.032499, \qquad X^2 = 0.0325041$$

Both G^2 and X^2 are asymptotically χ^2_s where

$$s = (m-1) - k = (3-1) - 1 = 1$$

p-values

For G^2, p-value = 0.85694. For X^2, p-value = 0.85682

Very large p-values indicate that we should not reject H_0.

In other words, the model is very good.

Why Modeling the Data?

- **Scientific purposes.**
- **Smaller number of parameters.**
- **Smaller errors (variance) if the model is correct (or close to be correct).**

The Hardy-Weinberg (3-cell) model was directly drived from science. In many cases, we have to derive the models from the observations. **Logistic regression** is a popular and flexible model for categorical responses.

Logistic Regression

Logistic regression is one of the most widely used statistical tools for predicting cateogrical outcomes.

General setup for binary logistic regression

n observations: $\{x_i, y_i\}, i = 1$ to n. x_i can be a vector.

$y_i \in \{0, 1\}$. For example, "1" = "YES" and "0" = "NO".

Define

$$p(x_i) = \mathbf{Pr}\left(y_i = 1 | x_i\right) = \pi(x_i)$$

$$i.e., \ \mathbf{Pr}\left(y_i = 0 | x_i\right) = 1 - p(x_i).$$

The major assumption of logistic regression

$$\log \frac{p(x_i)}{1-p(x_i)} = \beta_0 + \beta_1 x_{i,1} + ... + \beta_p x_{i,p} = \sum_{j=0}^{p} \beta_j x_{i,j}.$$

Here, we treat $x_{i,0} = 1$. We can also use vector notation to write

$$\log \frac{p(x_i;\beta)}{1-p(x_i;\beta)} = x_i \beta.$$

Here, we view x_i as a row-vector and β as a column-vector.

The model in vector notation

$$p(x_i;\beta) = \frac{e^{x_i\beta}}{1+e^{x_i\beta}}, \qquad 1-p(x_i;\beta) = \frac{1}{1+e^{x_i\beta}},$$

Log likelihood for the ith observation:

$$\begin{aligned} l_i(\beta|x_i) =& (1-y_i)\log\left[1-p(x_i;\beta)\right] + y_i \log p(x_i;\beta) \\ =& \begin{cases} \log p(x_i;\beta) & \text{if } y_i = 1 \\ \log\left[1-p(x_i;\beta)\right] & \text{if } y_i = 0 \end{cases} \end{aligned}$$

To understand this, consider binomial with only one sample $binomial(1, p(x_i))$ (i.e., Bernouli). When $y_i = 1$, the log likelihood is $\log p(x_i)$ and when $y_i = 0$, the log likelihood is $\log\left(1-p(x_i)\right)$. These two formulas can be written into one.

Joint log likelihood for n observations:

$$l(\beta|x_1, ..., x_n) = \sum_{i=1}^{n} l_i(\beta|x_i)$$

$$= \sum_{i=1}^{n} (1 - y_i) \log [1 - p(x_i; \beta)] + y_i \log p(x_i; \beta)$$

$$= \sum_{i=1}^{n} y_i \log \frac{p(x_i; \beta)}{1 - p(x_i; \beta)} + \log [1 - p(x_i; \beta)]$$

$$= \sum_{i=1}^{n} y_i x_i \beta - \log \left(1 + e^{x_i \beta}\right)$$

The remaining task is to solve the optimization problem by MLE.

The plan

- Solve logistic regression with only variable (one or two coefficients).
- Data examples of logistic regression.
- Intepret results of logistic regression.
- Solve general logistic regression.
- Logistic regression with regularization.

Logistic Regression with Only One Variable

Basic assumption

$$\text{logit}(\pi(x_i)) = \log \frac{p(x_i;\beta)}{1-p(x_i;\beta)} = \beta_0 + \beta_1 x_i$$

Joint Log likelihood

$$l(\beta|x_1,...,x_n) = \sum_{i=1}^{n} \left[y_i x_i \beta - \log\left(1 + e^{\beta_0 + x_i\beta_1}\right)\right]$$

Next, we solve the optmization problem for maximizing the joint likelihood, given the data.

First derivatives

$$\frac{\partial l(\beta)}{\partial \beta_0} = \sum_{i=1}^{n} y_i - p(x_i), \qquad \frac{\partial l(\beta)}{\partial \beta_1} = \sum_{i=1}^{n} x_i \left(y_i - p(x_i)\right),$$

Second derivatives

$$\frac{\partial^2 l(\beta)}{\partial \beta_0^2} = -\sum_{i=1}^{n} p(x_i)\left(1 - p(x_i)\right),$$

$$\frac{\partial^2 l(\beta)}{\partial \beta_1^2} = -\sum_{i=1}^{n} x_i^2 p(x_i)\left(1 - p(x_i)\right),$$

$$\frac{\partial^2 l(\beta)}{\partial \beta_0 \beta_1} = -\sum_{i=1}^{n} x_i p(x_i)\left(1 - p(x_i)\right)$$

Solve the MLE by Newton's Method or steepest descent (two-dim problem).

Logistic Regression without Intercept ($\beta_0 = 0$)

The simplified model

$$\text{logit}(\pi(x_i)) = \log \frac{p(x_i)}{1 - p(x_i)} = \beta x_i$$

Equivalently,

$$p(x_i) = \frac{e^{\beta x_i}}{1 + e^{\beta x_i}} = \pi(x_i), \qquad 1 - p(x_i) = \frac{1}{1 + e^{\beta x_i}},$$

Joint log likelihood for n observations:

$$l(\beta | x_1, ..., x_n) = \sum_{i=1}^{n} x_i y_i \beta - \log \left(1 + e^{\beta x_i}\right)$$

First derivative

$$l'(\beta) = \sum_{i=1}^{n} x_i \left(y_i - p(x_i)\right),$$

Second derivative

$$l''(\beta) = -\sum_{i=1}^{n} x_i^2 p(x_i)\left(1 - p(x_i)\right),$$

Newton's Method updating formula

$$\beta_{t+1} = \beta_t - \frac{l'(\beta_t)}{l''(\beta_t)}$$

Steepest descent (in fact ascent) updating formula

$$\beta_{t+1} = \beta_t + \Delta l'(\beta_t)$$

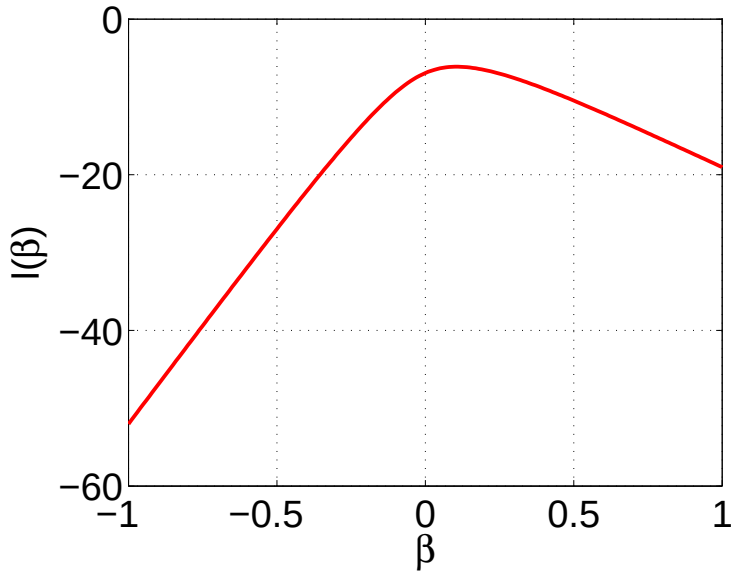
A Numerical Example of Logistic Regression

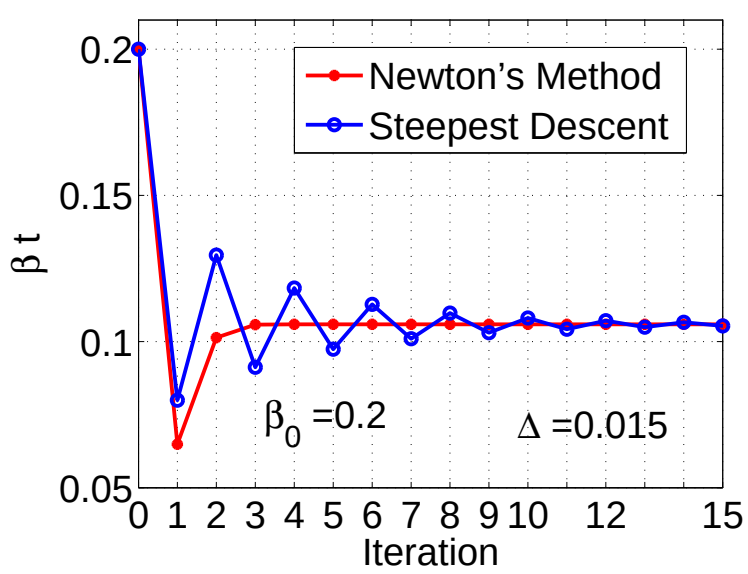
Data

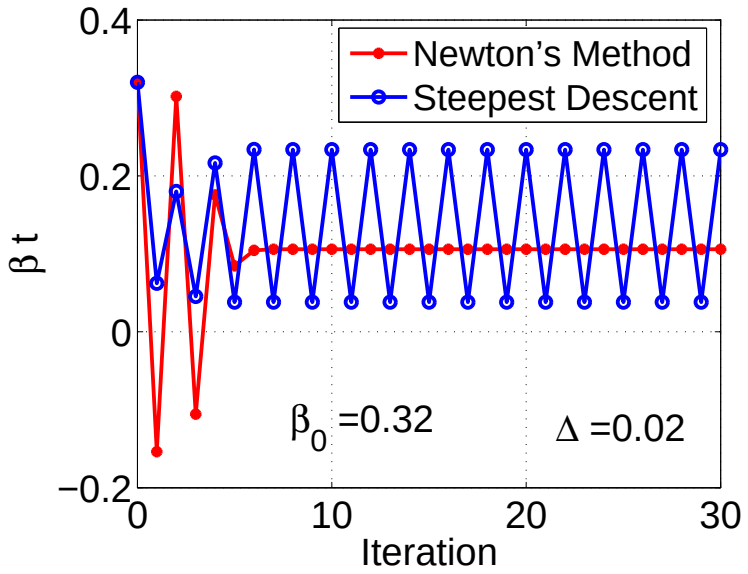
$x = \{8, 14, -7, 6, 5, 6, -5, 1, 0, -17\}$

$y = \{1, 1, 0, 0, 1, 0, 1, 0, 0, 0\}$

Log likelihood function

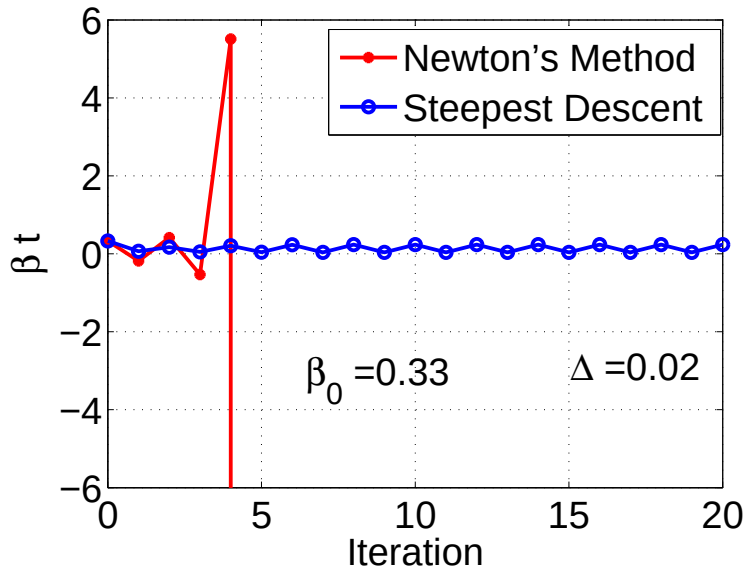


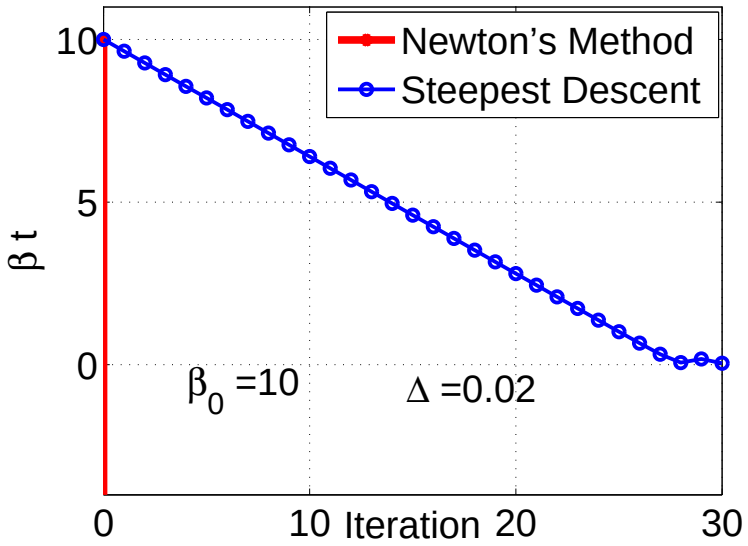




Steepest descent is quite sensitive to the step size Δ.

Too large Δ leads to oscillation.





Newton's Method is sensitive to the starting point β_0. May not converge at all.

The starting point (mostly) only affects computing time of steepest descent.

In general, with multiple variables, we need to use the matrix formulation, which in fact is easier to implement in matlab.

Newon's Method for Logistic Regression with β_0 and β_1

Analogous to the one variable case, the Newton's update formula is

$$\beta^{\mathbf{new}} = \beta^{\mathbf{old}} - \left[\left(\frac{\partial^2 \mathbf{l}(\beta)}{\partial \beta \partial \beta^\mathsf{T}} \right)^{-1} \frac{\partial \mathbf{l}(\beta)}{\partial \beta} \right]_{\beta^{\mathbf{old}}}$$

where $\beta = \begin{bmatrix} \beta_0 \\ \beta_1 \end{bmatrix}$,

$$\frac{\partial \mathbf{l}(\beta)}{\partial \beta} =$$

$$\begin{bmatrix} \sum_{i=1}^n y_i - p(x_i) \\ \sum_{i=1}^n x_i \left(y_i - p(x_i) \right) \end{bmatrix} = \begin{bmatrix} 1 & x_1 \\ 1 & x_2 \\ \dots & \\ 1 & x_n \end{bmatrix}^\mathsf{T} \begin{bmatrix} y_1 - p(x_1) \\ y_2 - p(x_2) \\ \dots \\ y_n - p(x_n) \end{bmatrix} = \mathbf{X}^\mathsf{T} (\mathbf{y} - \mathbf{p})$$

$$\left(\frac{\partial^{\mathbf{2}}\mathbf{l}(\beta)}{\partial\beta\partial\beta^{\mathsf{T}}}\right)$$

$$= \begin{bmatrix} -\sum_{i=1}^{n} p(x_i)\left(1-p(x_i)\right) & -\sum_{i=1}^{n} x_i p(x_i)\left(1-p(x_i)\right) \\ -\sum_{i=1}^{n} x_i p(x_i)\left(1-p(x_i)\right) & -\sum_{i=1}^{n} x_i^2 p(x_i)\left(1-p(x_i)\right) \end{bmatrix}$$

$$= -\mathbf{X}^{\mathsf{T}}\mathbf{W}\mathbf{X}$$

$$\mathbf{W} = \begin{bmatrix} p(x_1)(1-p(x_1)) & 0 & 0\ldots & 0 \\ 0 & p(x_2)(1-p(x_2)) & 0\ldots & 0 \\ \ldots & & & \\ 0 & 0 & 0\ldots & p(x_n)(1-p(x_n)) \end{bmatrix}$$

Multivariate Logistic Regression Solution in Matrix Form

Newton' update formula

$$\beta^{new} = \beta^{old} - \left[\left(\frac{\partial^2 l(\beta)}{\partial \beta \partial \beta^{\mathsf{T}}} \right)^{-1} \frac{\partial l(\beta)}{\partial \beta} \right]_{\beta^{old}}$$

where, in a matrix form

$$\frac{\partial l(\beta)}{\partial \beta} = \sum_{i=1}^{n} x_i \left(y_i - p(x_i; \beta) \right) = \mathbf{X^T (y - p)}$$

$$\frac{\partial^2 l(\beta)}{\partial \beta \partial \beta^{\mathsf{T}}} = -\sum_{i=1}^{n} x_i^{\mathsf{T}} x_i p(x_i; \beta) \left(1 - p(x_i; \beta) \right) = -\mathbf{X^T W X}$$

We can write the update formula in a matrix form

$$\beta^{\mathbf{new}} = \left[\mathbf{X^T W X} \right]^{-1} \mathbf{X^T W z},$$

$$\mathbf{z} = \mathbf{X} \beta^{\mathbf{old}} + \mathbf{W}^{-1} (\mathbf{y} - \mathbf{p})$$

$$\mathbf{X} = \begin{bmatrix} 1 & x_{1,1} & x_{1,2} & ... & x_{1,p} \\ 1 & x_{2,1} & x_{2,2} & ... & x_{2,p} \\ ... & & & & \\ 1 & x_{n,1} & x_{n,2} & ... & x_{n,p} \end{bmatrix} \in \mathbb{R}^{n \times (p+1)}$$

$$\mathbf{W} = \begin{bmatrix} p_1(1-p_1) & 0 & 0 & ... & 0 \\ 0 & p_2(1-p_2) & 0 & ... & 0 \\ ... & & & & \\ 0 & 0 & 0 & ... & p_n(1-p_n) \end{bmatrix} \in \mathbb{R}^{n \times n}$$

where $p_i = p(x_i; \beta^{old})$.

Derivation

$$\begin{aligned}
\beta^{\mathbf{new}} =& \beta^{\mathbf{old}} - \left[\left(\frac{\partial^2 l(\beta)}{\partial\beta\partial\beta^{\mathsf{T}}}\right)^{-1}\frac{\partial l(\beta)}{\partial\beta}\right]_{\beta^{old}} \\
=& \beta^{\mathbf{old}} + \left[\mathbf{X^T W X}\right]^{-1}\mathbf{X^T}(\mathbf{y}-\mathbf{p}) \\
=& \left[\mathbf{X^T W X}\right]^{-1}\left[\mathbf{X^T W X}\right]\beta^{\mathbf{old}} + \left[\mathbf{X^T W X}\right]^{-1}\mathbf{X^T}(\mathbf{y}-\mathbf{p}) \\
=& \left[\mathbf{X^T W X}\right]^{-1}\mathbf{X^T W}\left(\mathbf{X}\beta^{\mathbf{old}} + \mathbf{W^{-1}}(\mathbf{y}-\mathbf{p})\right) \\
=& \left[\mathbf{X^T W X}\right]^{-1}\mathbf{X^T W z}
\end{aligned}$$

Note that $\left[\mathbf{X^T W X}\right]^{-1}\mathbf{X^T W z}$ looks a lot like (weighted) least square.

Two major practical issues:

- The inverse may not (usually does not) exist, especially with large datasets.
- Newton update steps may be too agressive and lead to divergence.

Fitting Logistic Regression with a Learning Rate

At time t, update each coefficient vector:

$$\beta^{\mathbf{t}} = \beta^{(\mathbf{t-1})} + \nu \left[\mathbf{X^T W X}\right]^{-1} \mathbf{X^T}(\mathbf{y} - \mathbf{p})\Big|_{\mathbf{t-1}}$$

where

$$\mathbf{W} = \text{diag}\left[p_i(1-p_i)\right]_{i=1}^{n}$$

The magic parameter ν can be viewed as the learning rate to help make sure that the procedure converges. Practically, it is often set to be $\nu = 0.1$.

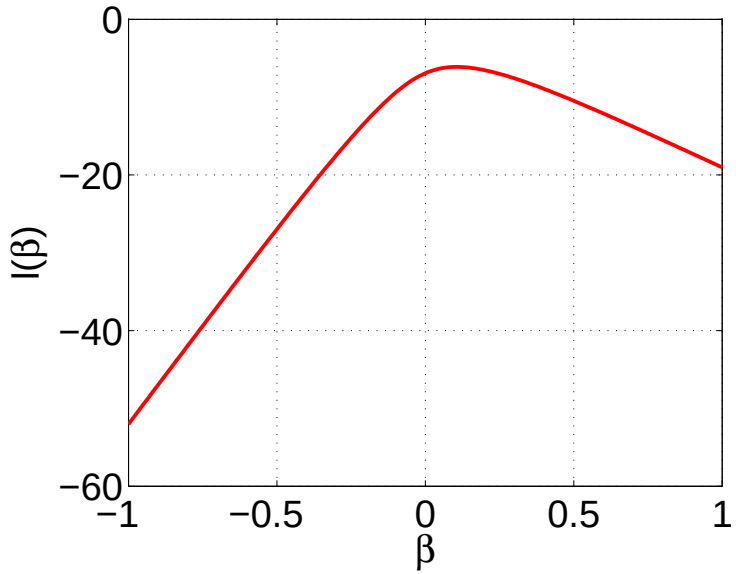
Revisit The Simple Example with Only One β

Data

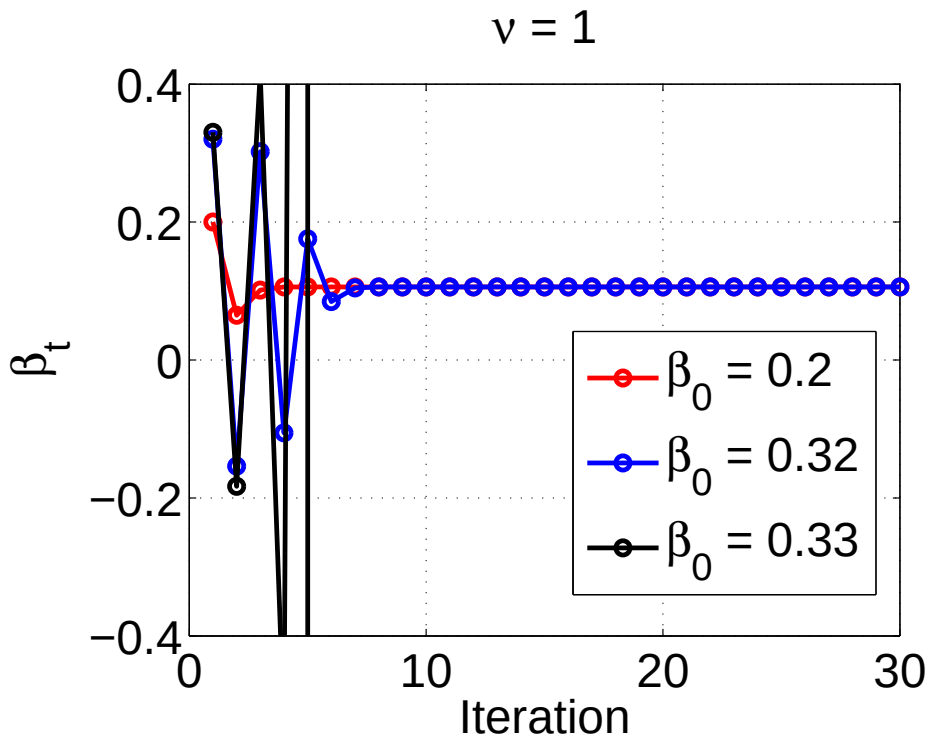
$x = \{8, 14, -7, 6, 5, 6, -5, 1, 0, -17\}$

$y = \{1, 1, 0, 0, 1, 0, 1, 0, 0, 0\}$

Log likelihood function

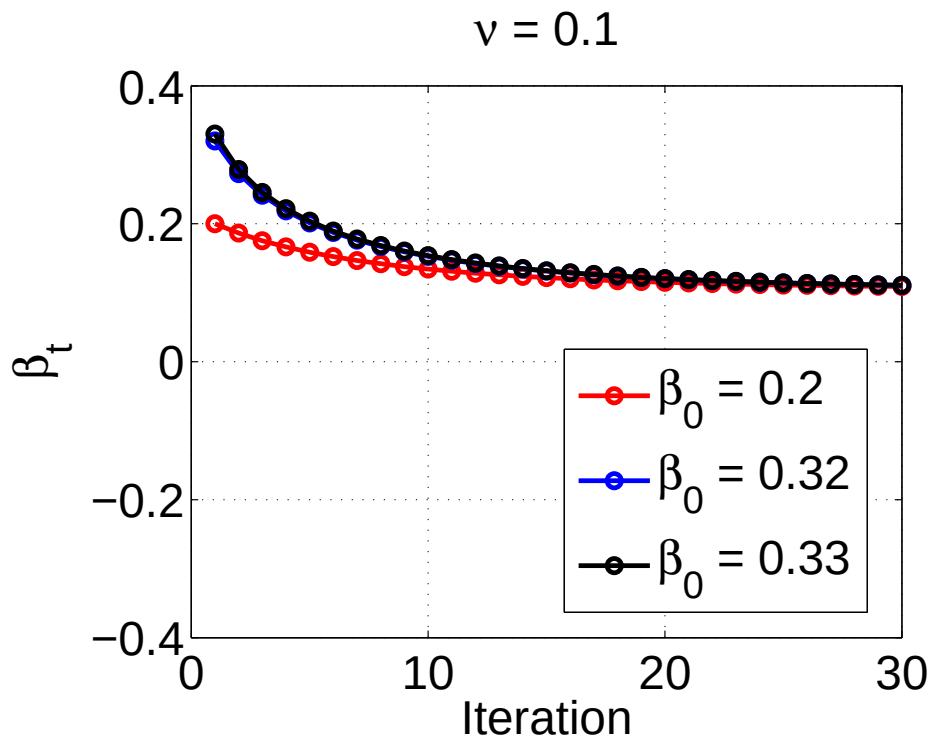


Newton's Method with Learning Rate $\nu = 1$



When initial $\beta_0 = 0.32$, the method coverges. When $\beta_0 = 0.33$, it does not converge.

Newton's Method with Learning Rate $\nu = 0.1$



Fitting Logistic Regression With Regularization

The almost correct update formula:

$$\beta^{\mathbf{t}} = \beta^{(\mathbf{t-1})} + \nu \left[\mathbf{X^T W X} + \lambda \mathbf{I}\right]^{-1} \mathbf{X^T}(\mathbf{y} - \mathbf{p})\Big|_{\mathbf{t-1}}$$

Adding the regularization parameter λ usually improves the numerical stability and some times may even result in better test errors.

There are also good statsitical interpretations.

Fitting Logistic Regression With Regularization

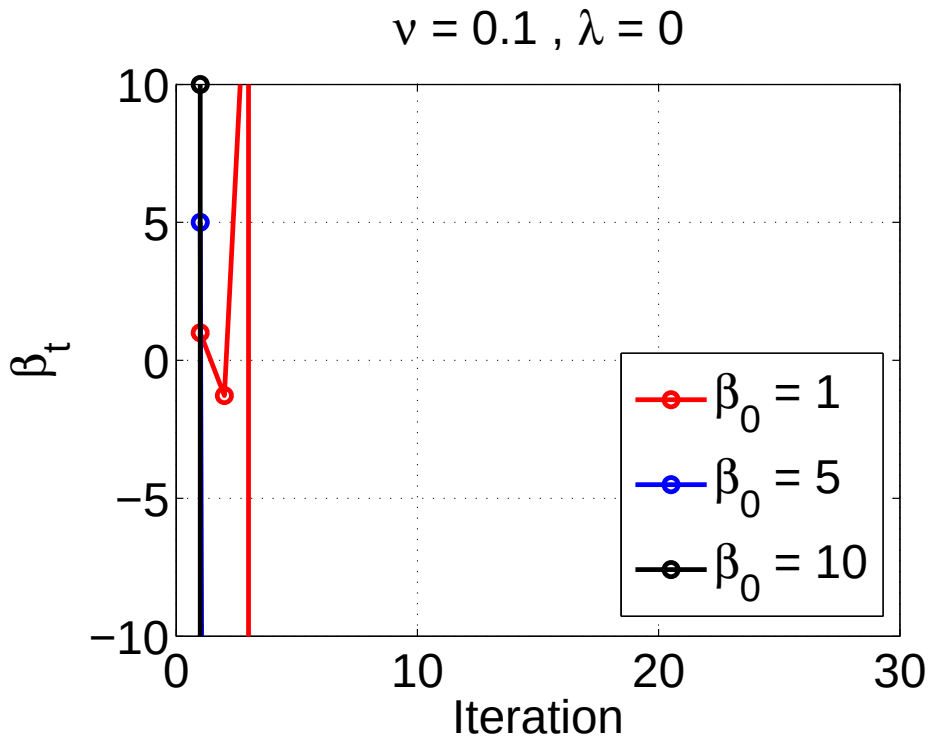
The update formula:

$$\beta^{\mathbf{t}} = \beta^{(\mathbf{t-1})} + \nu \left[\mathbf{X^T W X} + \lambda \mathbf{I}\right]^{-1} \left[\mathbf{X^T}(\mathbf{y} - \mathbf{p}) - \lambda\beta\right]\Big|_{\mathbf{t-1}}$$

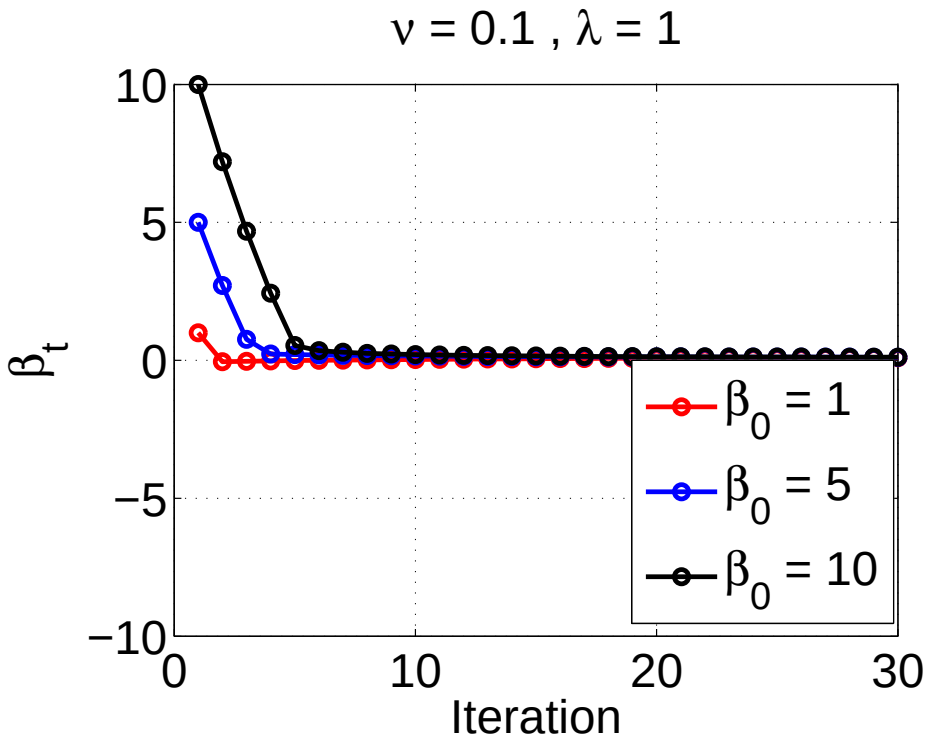
To understand the formula, consider the following modified (regularized) likelihood function:

$$l(\beta) = \sum_{i=1}^{n} \{y_i \log p_i + (1 - y_i) \log(1 - p_i)\} - \frac{\lambda}{2} \sum_{j=0}^{p} \beta_j^2$$

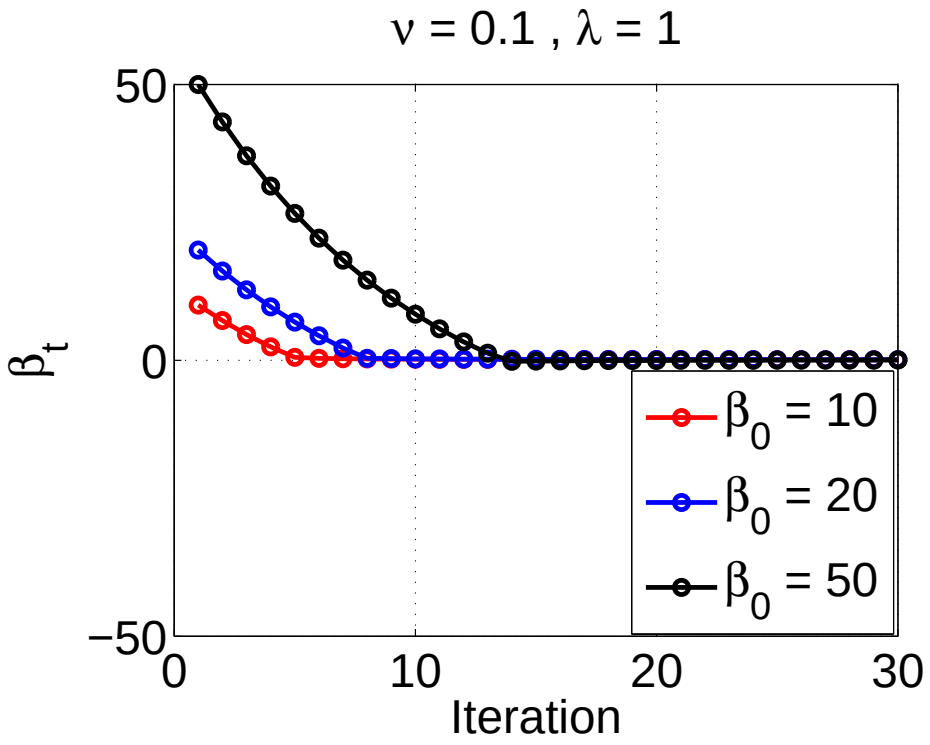
Newton's Method with No Regularization $\lambda = 0$ ($\nu = 0.1$)



Newton's Method with Regularization $\lambda = 1$ ($\nu = 0.1$)



Newton's Method with Regularization $\lambda = 1$ ($\nu = 0.1$)



Crab Data Analysis (Table 3.2)

Color (C)	Spine (S)	Width (W, cm)	Weight (Wt, Kg)	# Saterlites (Sa)
2	3	28.3	3.05	8
3	3	22.5	1.55	0
1	1	26.0	2.30	9
3	3	24.8	2.10	0
3	3	26.0	2.60	4
2	3	23.8	2.10	0
1	1	26.5	2.35	0
3	2	24.7	1.90	0

It is natural to view color as (norminal) cateogrical variable and weight and width as numerical variables. The distinction, however, is often not clear in practice.

Logistic regression for Sa classification using width only

$y = 1$ if $Sa > 0$, $y = 0$ if $Sa = 0$. Only one variable $x = W$. The task is to compute $\mathbf{Pr}\left(y = 1 \middle| x\right)$ and classify the data using a simple classification rule:

$$\hat{y}_i = 1, \qquad \text{if} \quad \hat{p}_i > 0.5$$

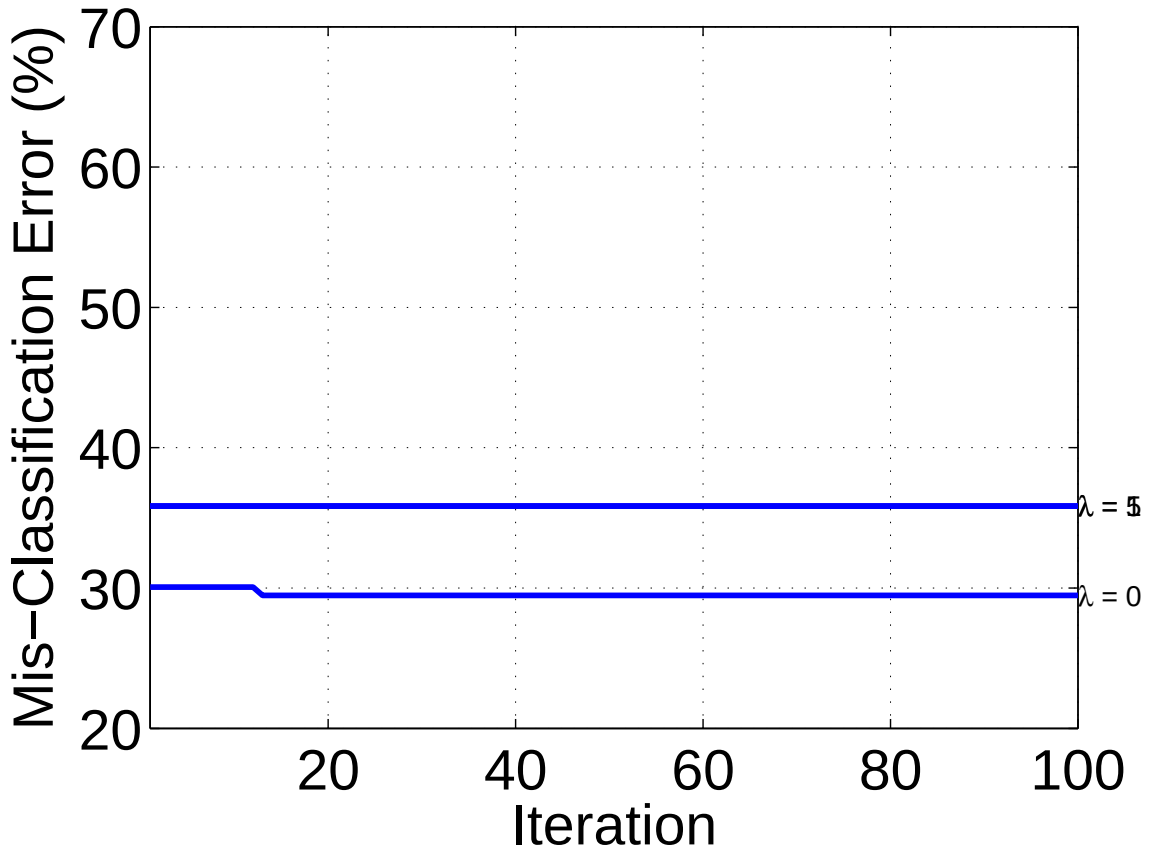
Using own matlab code, the fitted model is

$$\hat{p}(x_i) = \frac{e^{-12.3108+0.497x_i}}{1 + e^{-12.3108+0.497x_i}}$$

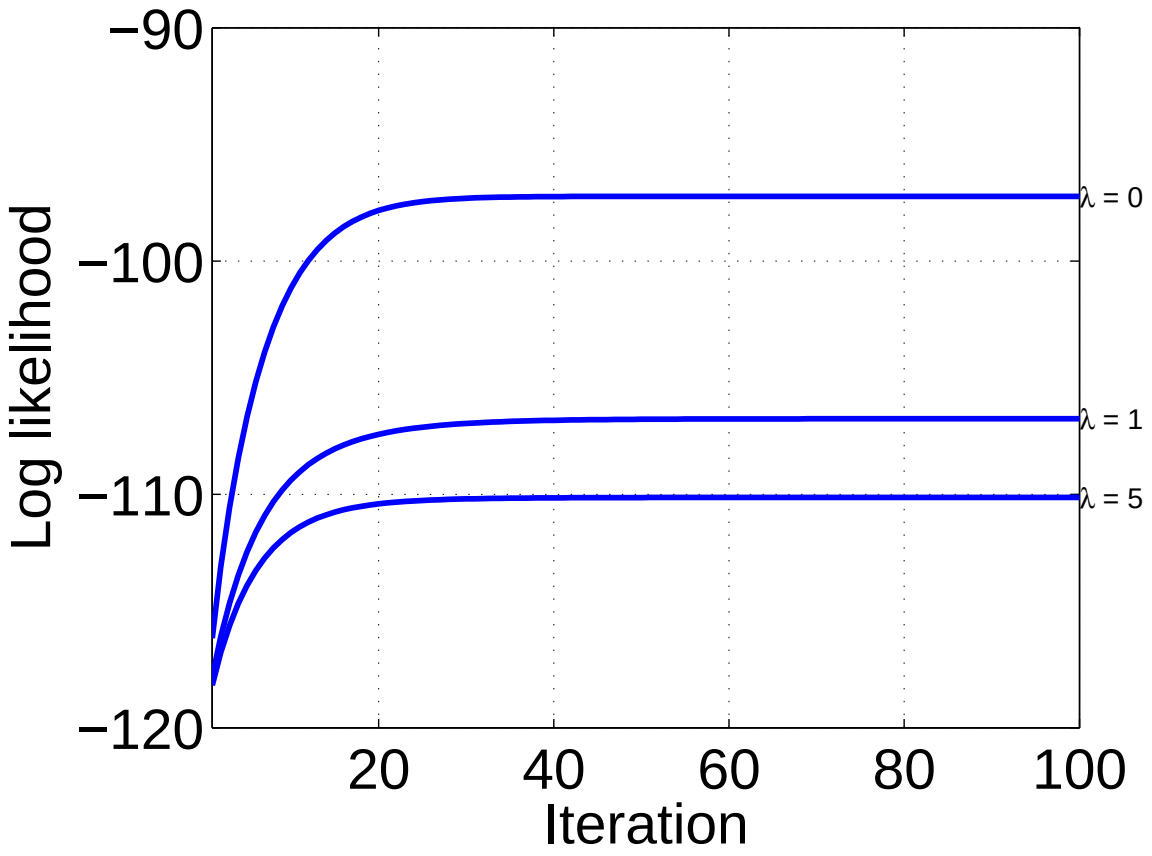
If we choose not to include the intercept term, the fitted model becomes

$$\hat{p}(x_i) = \frac{e^{0.02458x_i}}{1 + e^{0.02458x_i}}$$

Training mis-classification errors



Training log likelihood

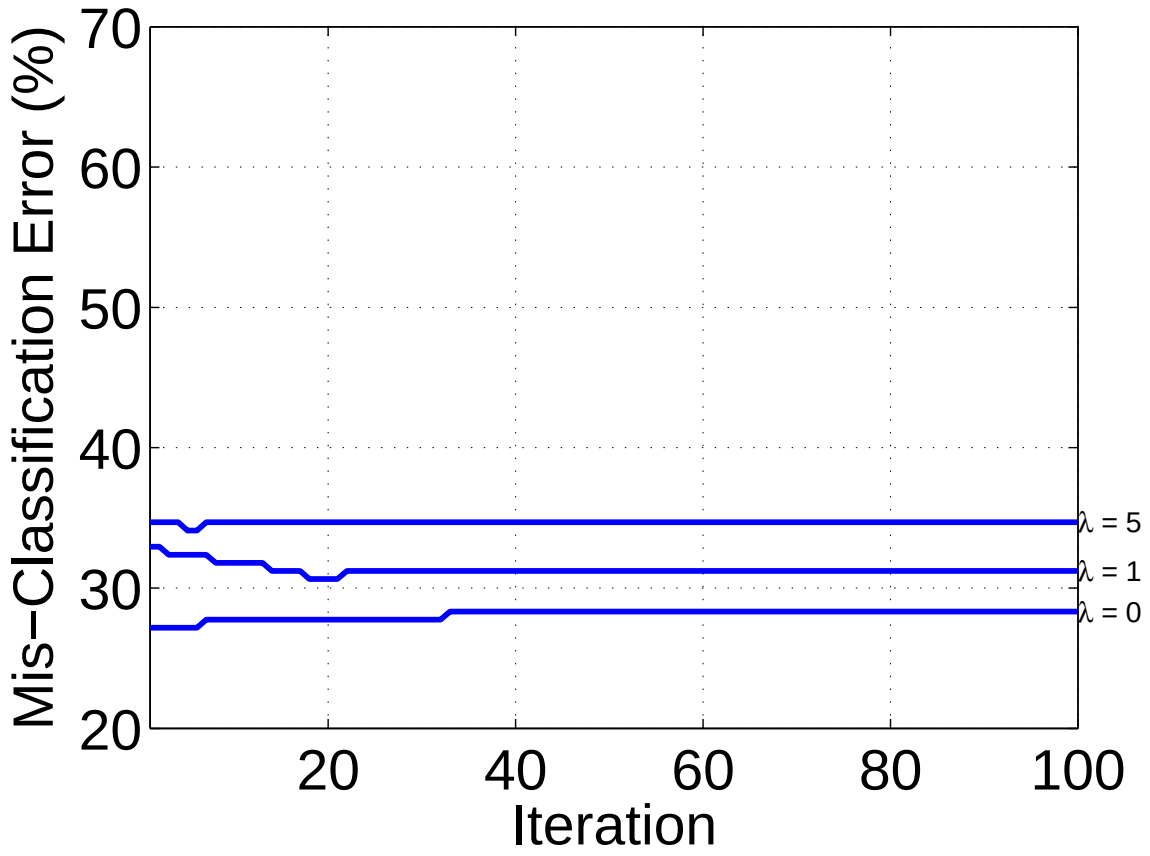


Logistic regression for Sa classification using S, W, and Wt

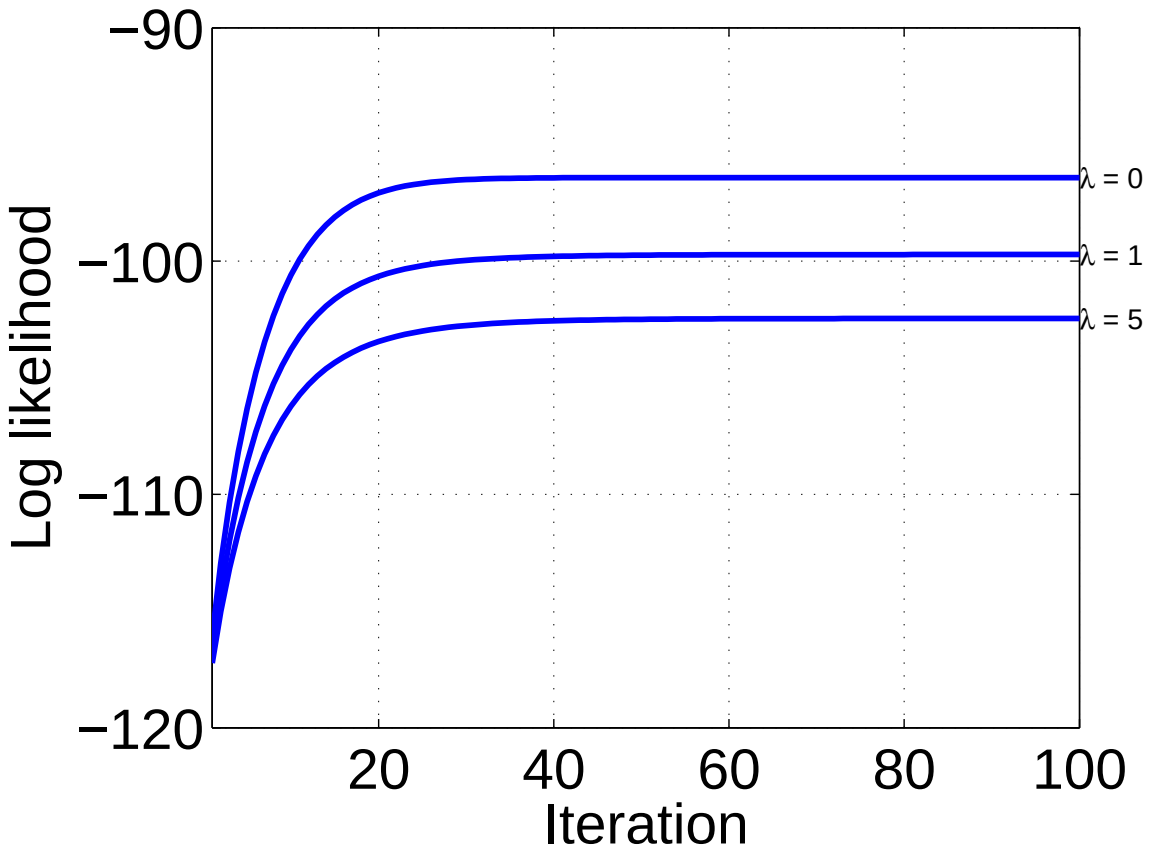
Using own matlab code, the fitted model is

$$\hat{p}(S, W, Wt) = \frac{e^{-9.4684+0.0495S+0.3054W+0.8447Wt}}{1 + e^{-9.4684+0.0495S+0.3054W+0.8447Wt}}$$

Training mis-classification errors



Training log likelihood



Multi-Class Logistic Regression

Data: $\{x_i, y_i\}_{i=1}^n$, $x_i \in \mathbb{R}^{n \times p}$, $y_i \in \{0, 1, 2, ..., K-1\}$.

Probability model

$$p_{i,k} = \mathbf{Pr}\left\{y_i = k | x_i\right\}, \qquad k = 0, 1, ..., K-1,$$

$$\sum_{k=0}^{K-1} p_k = 1, \qquad (\text{only } K-1 \text{ degrees of freedom}).$$

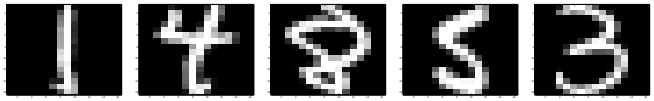
Label assignment

$$\hat{y}_i \big| x_i = \underset{k}{\text{argmax}} \ \hat{p}_{i,k}$$

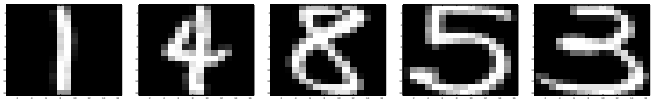
Multi-Class Example: USPS ZipCode Recognition

Person 1:

Person 2:



Person 3:



This task can be cast (simplified) as a $K = 10$-class classification problem.

Multinomial Logit Probability Model

$$p_{i,k} = \frac{e^{F_{i,k}}}{\sum_{s=0}^{K-1} e^{F_{i,s}}}$$

where $F_{i,k} = F_k(x_i)$ is the function to be learned from the data.

Linear logistic regression: $F_{i,k} = F_k(x_i) = x_i\beta_k$

Note that, $\beta_k = [\beta_{k,0}, \beta_{k,1}, ..., \beta_{k,p}]^{\mathsf{T}}$

Multinomial Maximum Likelihood

Mutlinomial likelihood: Suppose $y_i = k$,

$$Lik \propto p_{i,0}^0 \times ... \times p_{i,k}{}^1 \times ... \times p_{i,K-1}^0 = p_{i,k}$$

log likelihood:

$$l_i = \log p_{i,k}, \qquad \text{if } y_i = k$$

Total log-likelihood in a double summation form:

$$l(\beta) = \sum_{i=1}^{n} l_i = \sum_{i=1}^{n} \left\{ \sum_{k=0}^{K-1} r_{i,k} \log p_{i,k} \right\}$$

$$r_{i,k} = \begin{cases} 1 & \text{if } y_i = k \\ 0 & \text{otherwise} \end{cases}$$

Derivatives of Multi-Class Log-likelihood

The first derivative:

$$\frac{\partial l_i}{\partial F_{i,k}} = (r_{i,k} - p_{i,k})$$

Proof:

$$\frac{\partial p_{i,k}}{\partial F_{i,k}} = \frac{\left[\sum_{s=0}^{K-1} e^{F_{i,s}}\right] e^{F_{i,k}} - e^{2F_{i,k}}}{\left[\sum_{s=0}^{K-1} e^{F_{i,s}}\right]^2} = p_{i,k}\left(1 - p_{i,k}\right)$$

$$\frac{\partial p_{i,k}}{\partial F_{i,t}} = \frac{-e^{F_{i,k}} e^{F_{i,t}}}{\left[\sum_{s=0}^{K-1} e^{F_{i,s}}\right]^2} = -p_{i,k} p_{i,t}$$

$$\frac{\partial l_i}{\partial F_{i,k}} = \sum_{s=0}^{K-1} r_{i,s} \frac{1}{p_{i,s}} \frac{\partial p_{i,s}}{\partial F_{i,k}} = r_{i,k} \frac{1}{p_{i,k}} p_{i,k}(1 - p_{i,k}) + \sum_{s \neq k} r_{i,s} \frac{1}{p_{i,s}} \frac{\partial p_{i,s}}{\partial F_{i,k}}$$

$$= r_{i,k}(1 - p_{i,k}) - \sum_{s \neq k} r_{i,s} p_{i,k} = r_{i,k} - \sum_{s=0}^{K-1} r_{i,s} p_{i,k} = r_{i,k} - p_{i,k} \square$$

The second derivatives:

$$\frac{\partial^2 l_i}{\partial F_{i,k}^2} = -p_{i,k}\left(1 - p_{i,k}\right),$$

$$\frac{\partial^2 l_i}{\partial F_{i,k} F_{i,s}} = -p_{i,k} p_{i,s}$$

Multi-class logistic regression can be fairly complicated. Here, we introduce a simpler approach, which does not seem to explicitly appear in common textbooks.

Conceptually, we fit K binary classification problems (one vs rest) at each iteration. That is, at each iteration, we update β_k seperately for each class. At the end of each iteration, we jointly update the probabilities $p_{i,k} = \frac{e^{x_i \beta_k}}{\sum_{s=0}^{K-1} e^{x_i \beta_s}}$.

A Simple Implementation for Multi-Class Logistic Regression

At time t, update each coefficient vector:

$$\beta_{\mathbf{k}}^{\mathbf{t}} = \beta_{\mathbf{k}}^{(\mathbf{t-1})} + \nu \left[\mathbf{X^T W_k X}\right]^{-1} \mathbf{X^T}(\mathbf{r_k} - \mathbf{p_k})\Big|_{\mathbf{t-1}}$$

where

$$\mathbf{r_k} = \left[r_{1,k}, r_{2,k}, ..., r_{n,k}\right]^{\mathsf{T}}$$

$$\mathbf{p_k} = \left[p_{1,k}, p_{2,k}, ..., p_{n,k}\right]^{\mathsf{T}}$$

$$\mathbf{W_k} = \text{diag}\left[p_{i,k}(1 - p_{i,k})\right]_{i=1}^{n}$$

Then update $\mathbf{p_k}$, $\mathbf{W_k}$ for the next iteration.

Again, the magic parameter ν can be viewed as the learning rate to help make sure that the procedure converges. Practically, it is often set to be $\nu = 0.1$.

Logistic Regression With L_2 Regularization

Total log-likelihood in a double summation form:

$$l(\beta) = \sum_{i=1}^{n} \left\{ \sum_{k=0}^{K-1} r_{i,k} \log p_{i,k} \right\} - \frac{\lambda}{2} \sum_{k=0}^{K-1} \sum_{j=0}^{d} \beta_{k,j}^2$$

$$r_{i,k} = \begin{cases} 1 & \text{if } y_i = k \\ 0 & \text{otherwise} \end{cases}$$

Let $g(\beta) = \frac{\lambda}{2} \sum_{k=0}^{K-1} \sum_{j=0}^{d} \beta_{k,j}^2$, then

$$\frac{\partial g(\beta)}{\beta_{k,j}} = \beta_{k,j} \lambda, \qquad \frac{\partial^2 g(\beta)}{\beta_{k,j}^2} = \lambda$$

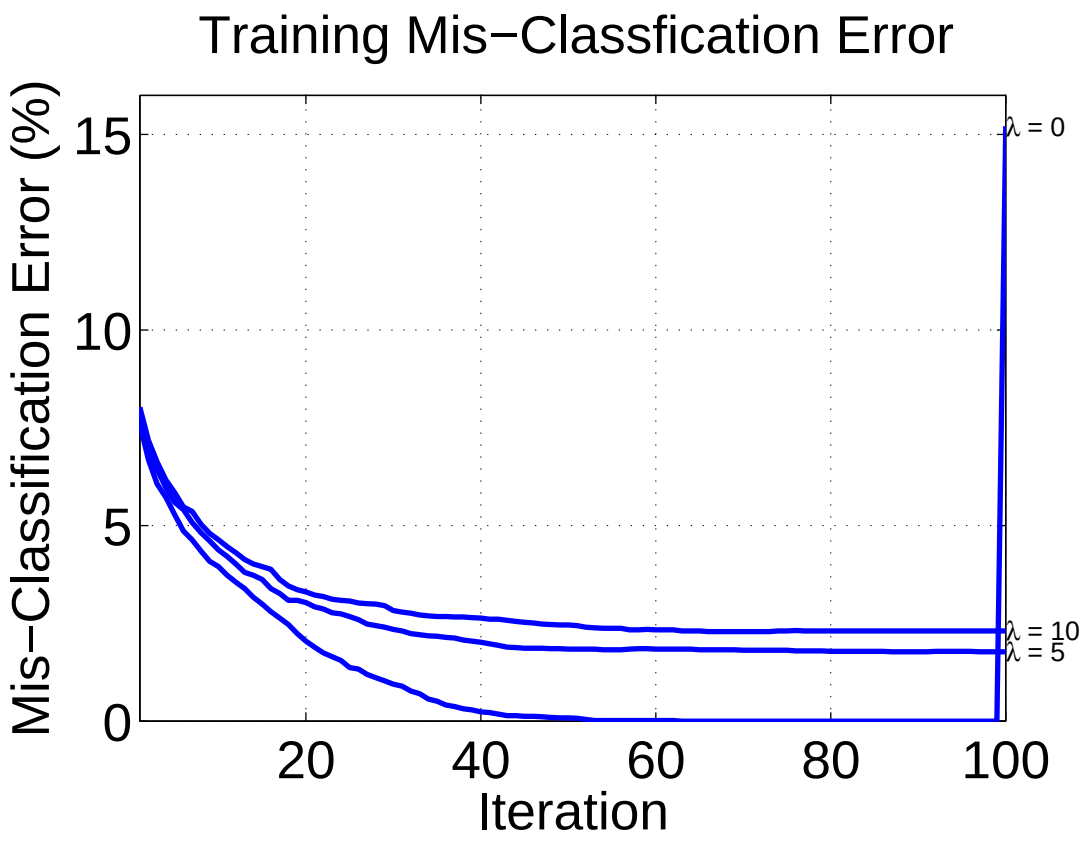
At time t, the updating formula becomes

$$\beta_{\mathbf{k}}^{\mathbf{t}} = \beta_{\mathbf{k}}^{(\mathbf{t-1})} + \nu \left[\mathbf{X^T W_k X} + \lambda \mathbf{I}\right]^{-1} \left[\mathbf{X^T}(\mathbf{r_k} - \mathbf{p_k}) - \lambda \beta_{\mathbf{k}}\right]\Big|_{\mathbf{t-1}}$$

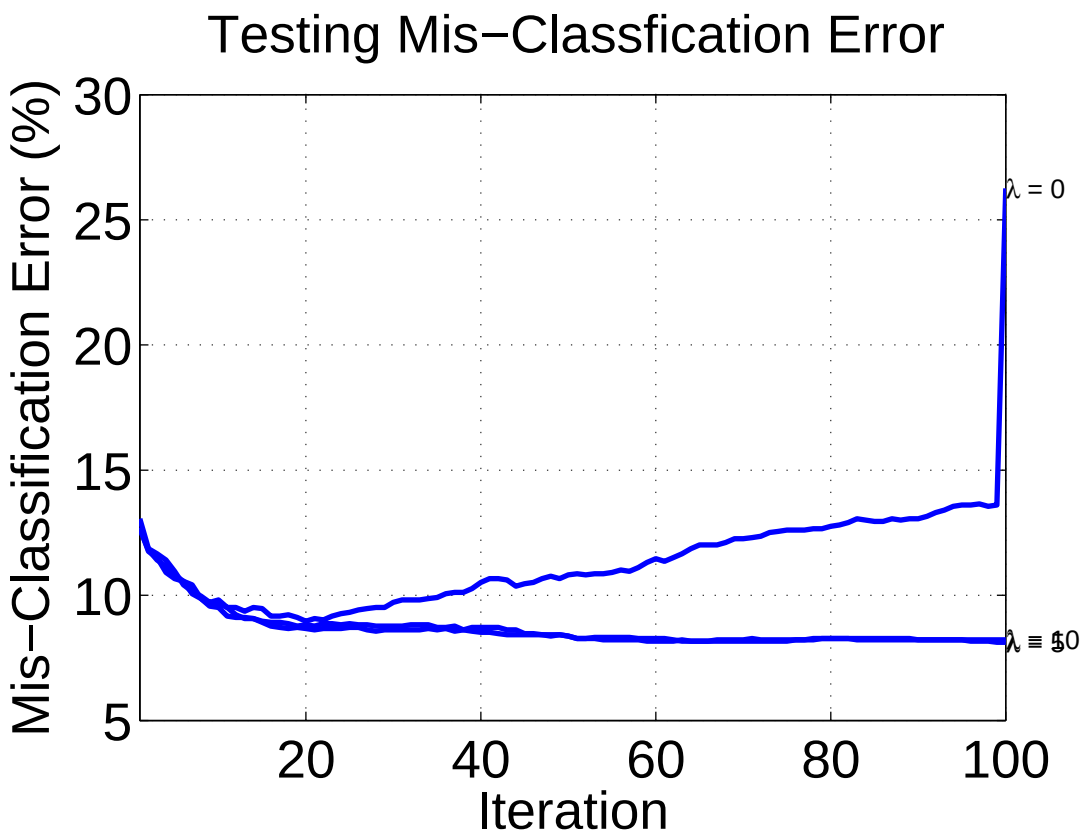
L_2 regularization sometimes improves the numerical stability and some times may even result in better test errors.

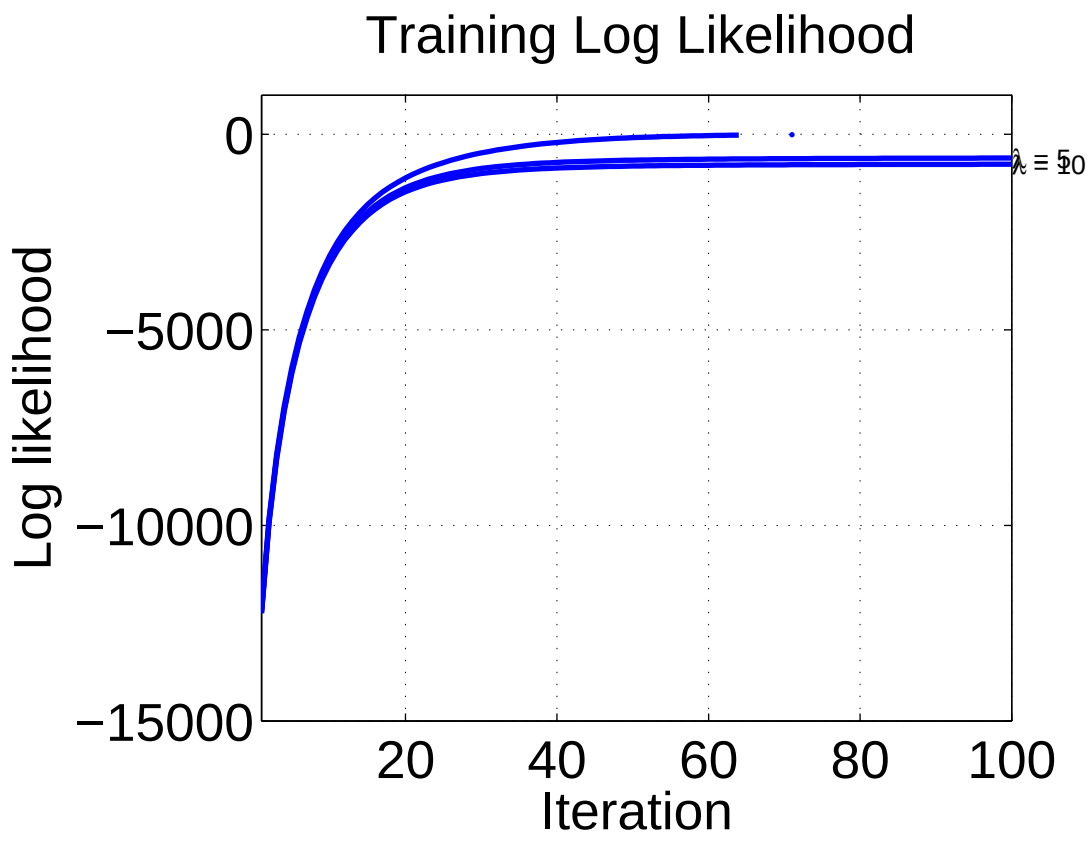
Logistic Regression Results on Zip Code Data

Zip code data: 7291 training examples in 256 dimensions. 2007 test examples.



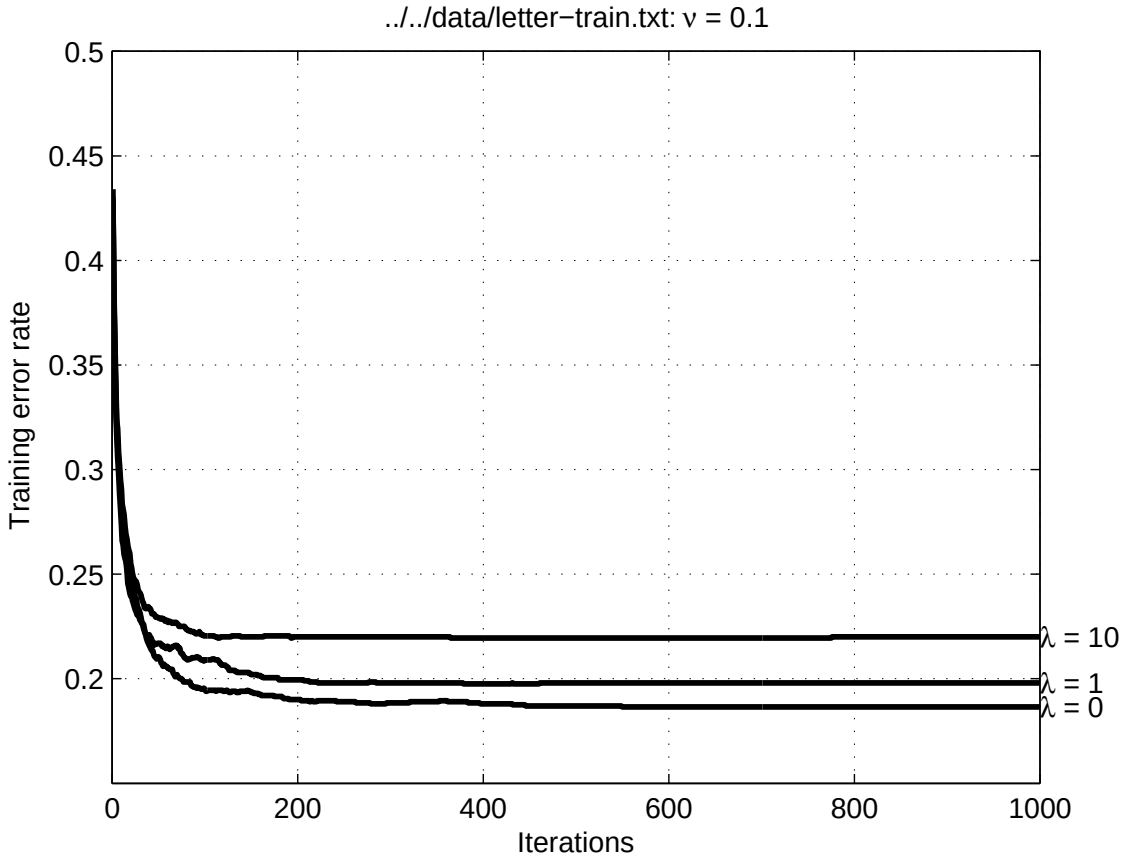
With no regularization ($\lambda = 0$), numerical problems may occur.

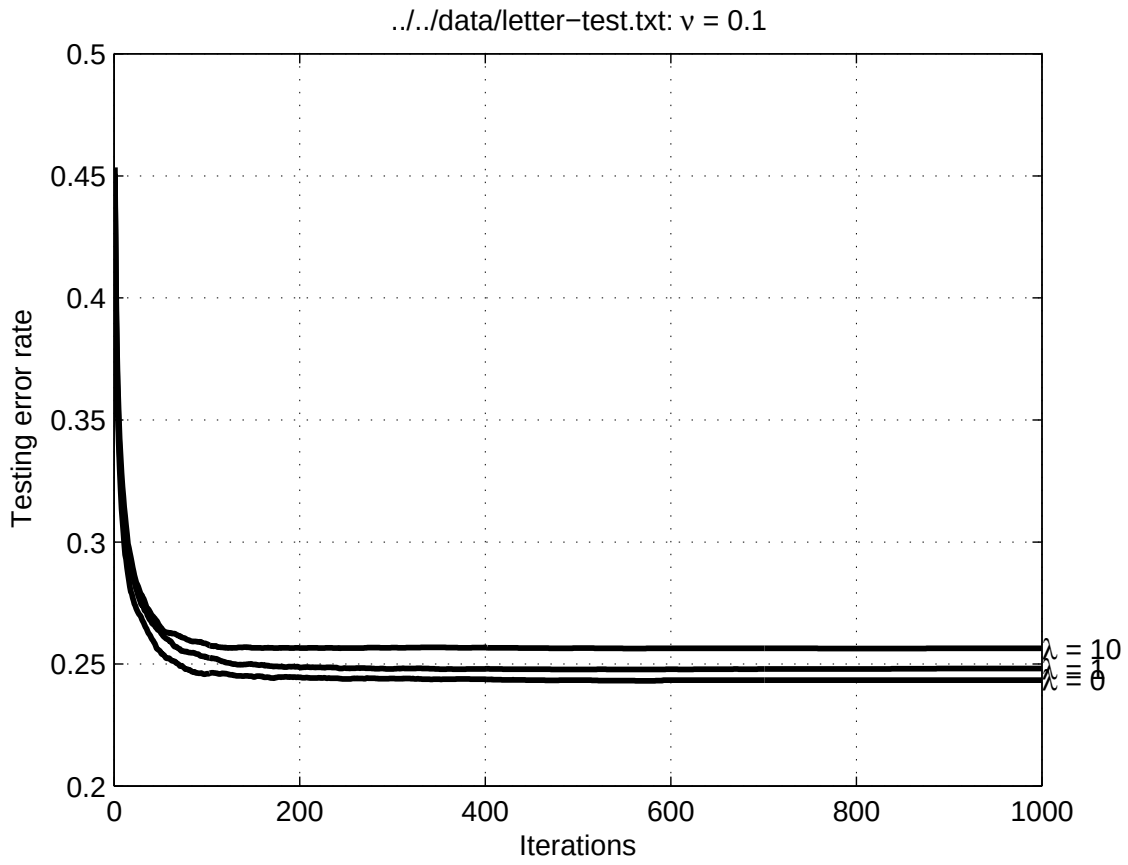

Testing Mis−Classfication Error
Mis−Classification Error (%)
30
25
20
15
10
5
20
40
60
80
100
Iteration
λ = 0
λ ≡ 1
λ ≡ 10

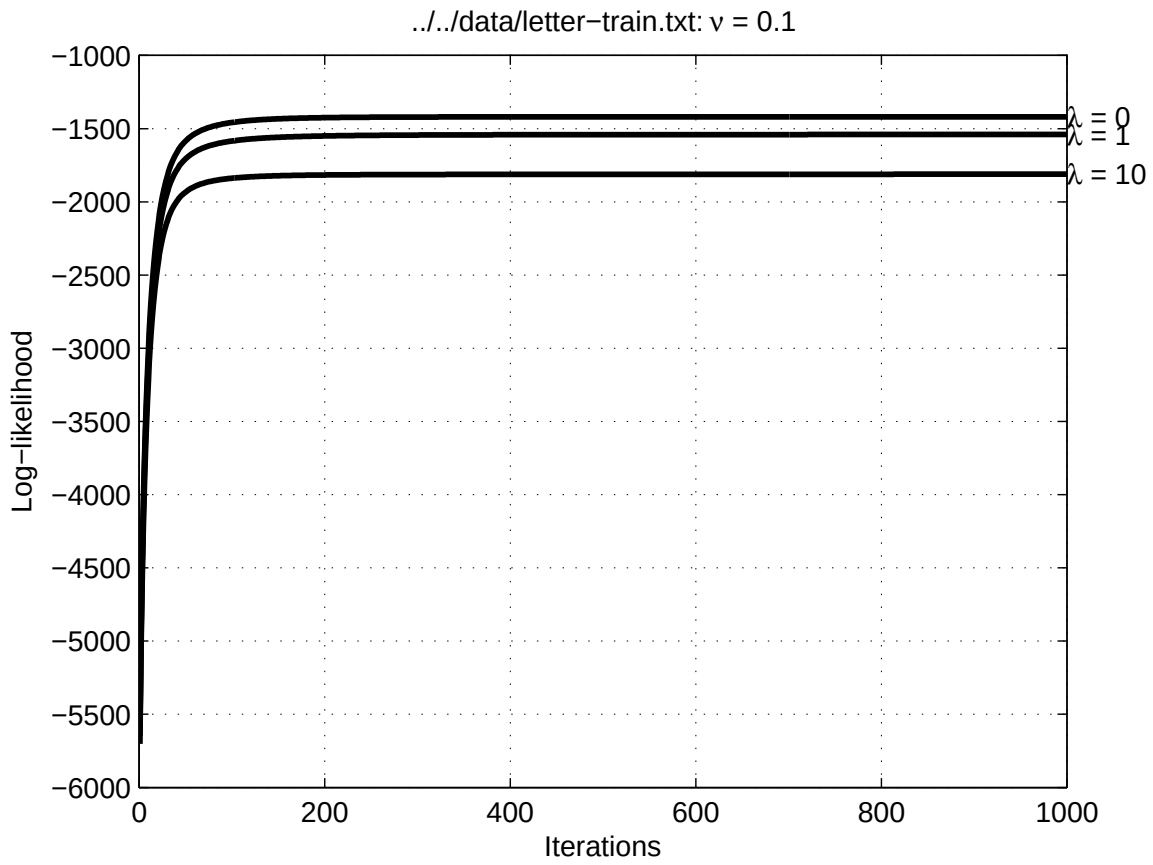

Training Log Likelihood
Log likelihood
0
−5000
−10000
−15000
20
40
60
80
100
Iteration
λ = 5
λ = 10

Another Example on Letter ($K = 26$) Recognition

Letter dataset: 2000 training samples in 16 dimensions. 18000 testing samples.

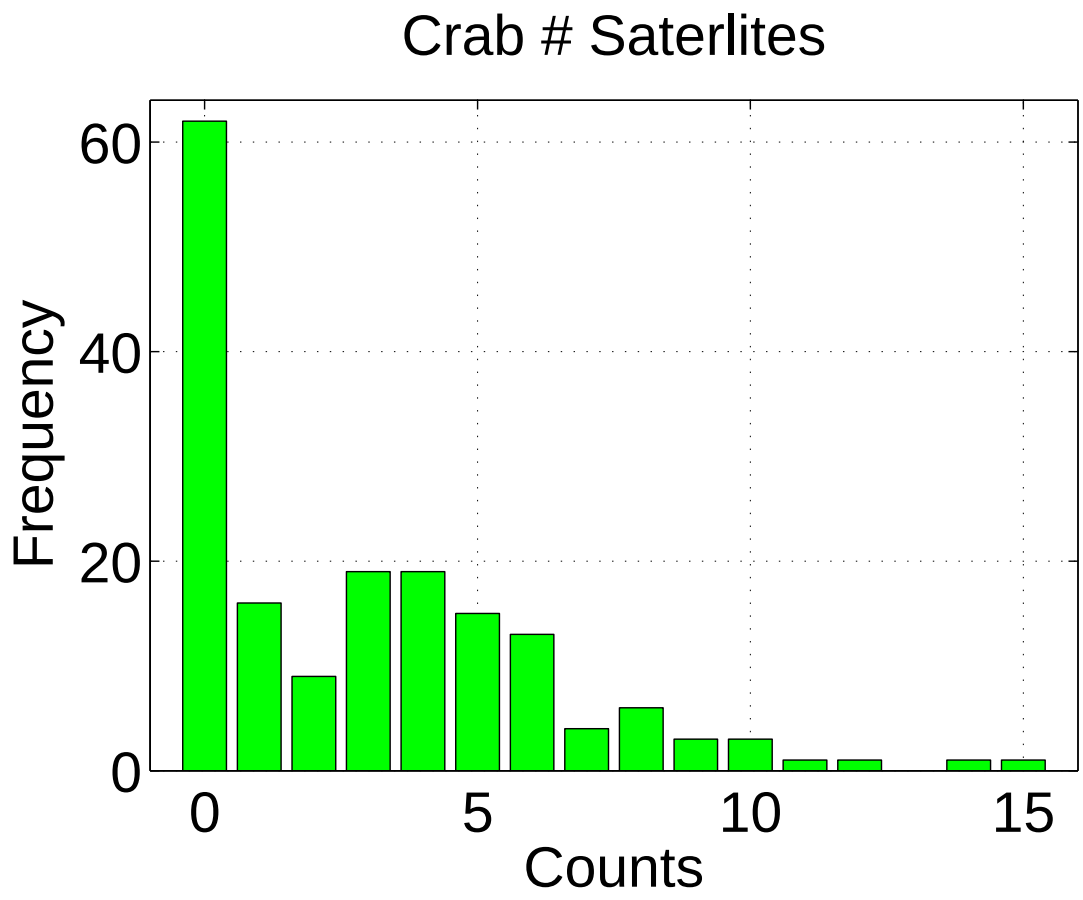



../../data/letter−test.txt: ν = 0.1
Testing error rate
Iterations
0.5
0.45
0.4
0.35
0.3
0.25
0.2
0
200
400
600
800
1000
λ = 10
λ ≡ 1
λ = 0


../../data/letter−train.txt: ν = 0.1
−1000
−1500
−2000
−2500
−3000
−3500
−4000
−4500
−5000
−5500
−6000
Log−likelihood
0
200
400
600
800
1000
Iterations
λ = 0
λ = 1
λ = 10

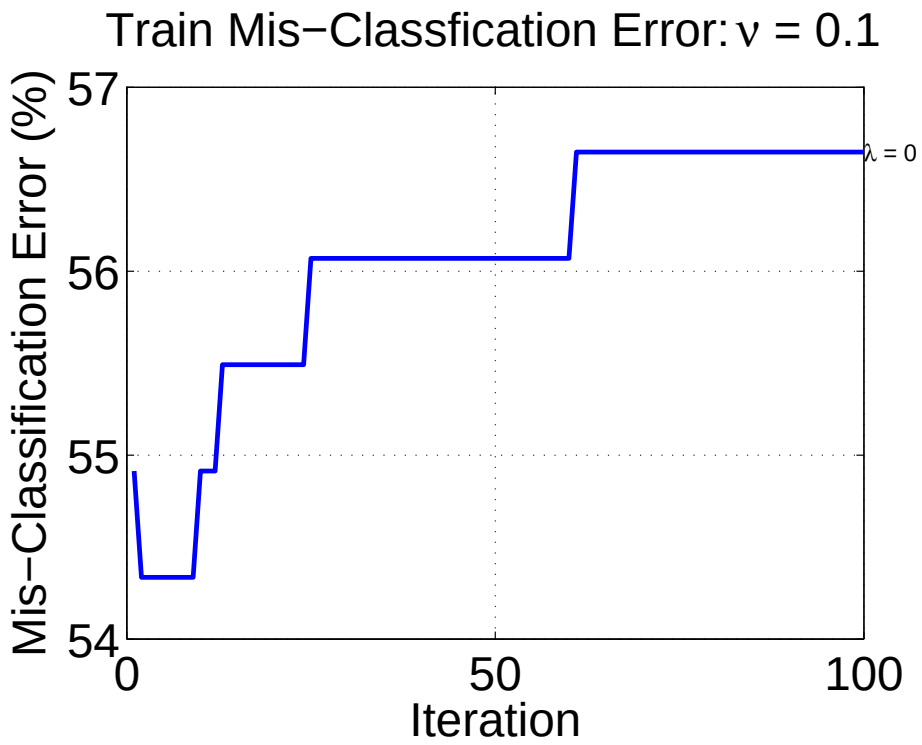
Revisit Crab Data as a Multi-Class Problem

Color (C)	Spine (S)	Width (W, cm)	Weight (Wt, Kg)	# Saterlites (Sa)
2	3	28.3	3.05	8
3	3	22.5	1.55	0
1	1	26.0	2.30	9
3	3	24.8	2.10	0
3	3	26.0	2.60	4
2	3	23.8	2.10	0
1	1	26.5	2.35	0
3	2	24.7	1.90	0



It appears reasonable to treat this as a binary classification problem, given the counts distribution and # samples. Nevertheless, it might be still interesting to consider it as a multi-class problem.

We consider a 6-class (0 to 5) classification problem by grouping all samples with counts ≥ 5 as class 5. Use 3 variales (S, W, Wt).



Compared to the binary-classification problem, it seems the mis-classification error is much higher. Why?

Some thoughts

- Multi-class problems are usually (but not always) more difficult.
- For binary-classifiction, an error rate of 50% is very bad because a random guess can achieve that. For K-class problem, the error rate of random guessing would be $1 - 1/K$ ($5/6$ in this example). So the results may be actually not too bad.
- Multi-class models are more complex (in that they require more parameters) and need more data samples. The crab dataset is very small.
- This problem may be actually ordinal classification instead of nomial, for biological reaons.

Dealing with Nominal Categorical Variables

It might be reasonable to consider "Color (C)" as a nominal cateogrical variable. Then how can we include it in our logistic regression model?

The trick is simple. Suppose the color variable take four different values. We add four binary variable (i.e., only taking values in $\{0,\ 1\}$. For one particular sample, only one of the four variables will take value 1.

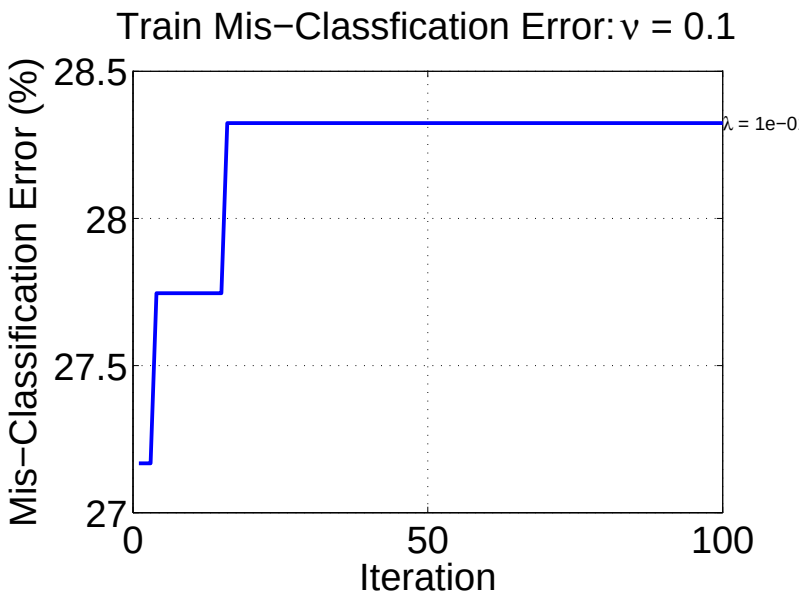
This is basically the same trick as we expand the y in multi-class logistic regression.

Adding Color as Four Binary Variables

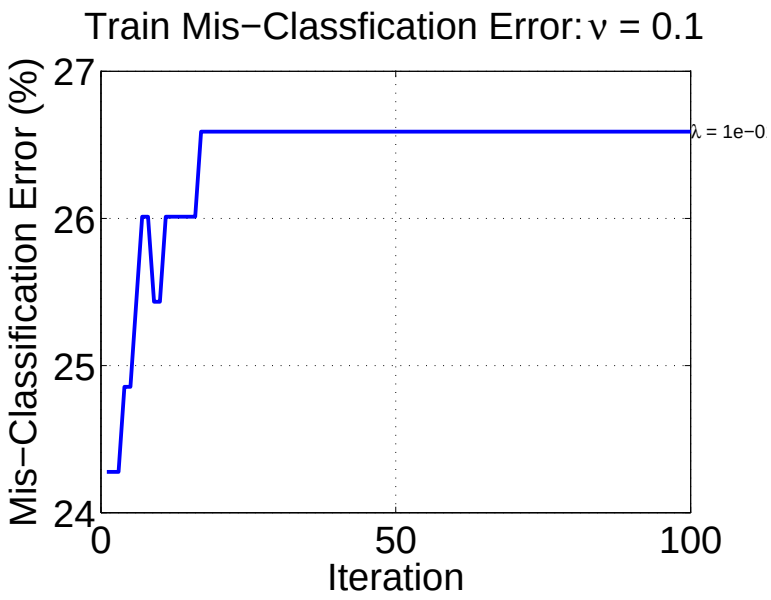
C1	C2	C3	C4	S	W	Wt	Sa
0	1	0	0	3	28.3	3.05	8
0	0	1	0	3	22.5	1.55	0
1	0	0	0	1	26.0	2.30	9
0	0	1	0	3	24.8	2.10	0
0	0	1	0	3	26.0	2.60	4
0	1	0	0	3	23.8	2.10	0
1	0	0	0	1	26.5	2.35	0
0	0	1	0	2	24.7	1.90	0

Adding the color variable noticeably reduced the (binary) classification error.

Colors Not Included



Colors Included

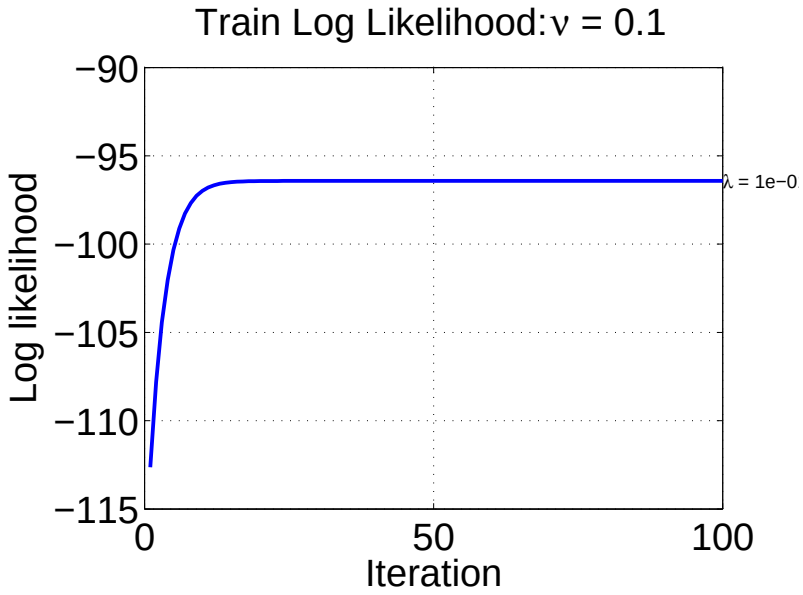


Here to minimize the effect of regularization, only $\lambda = 10^{-10}$ is used, just enough to ensure numerical stability.

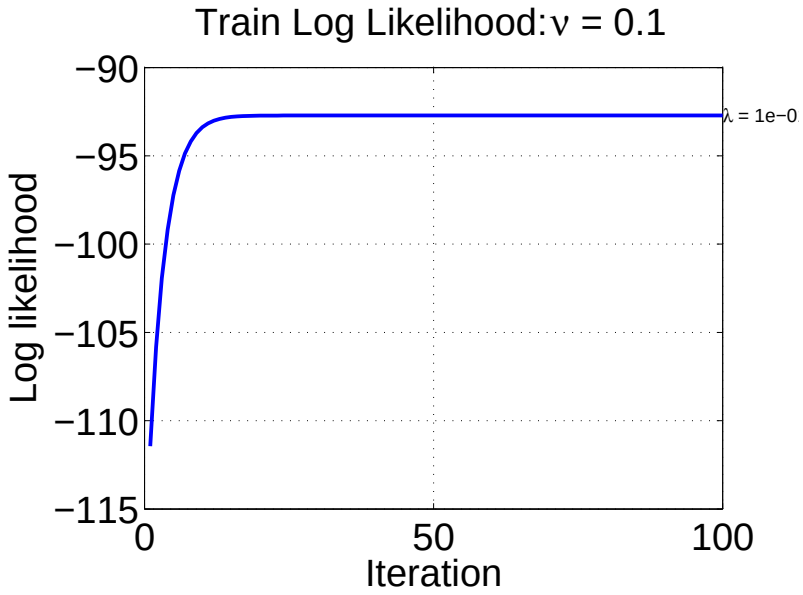
Logistic regression does not directly minimize mis-classification errors. The log likelihood probably better illustrates the effect of adding the color variable.

Adding the color variable noticeably improved the log likelihood.

Colors Not Included



Colors Included



Adding Pairwise (Interaction) Variables

Feature expansion is a common trick to boost the performance. For example,

$$(x_1, x_2, x_3, ..., x_p) \Longrightarrow$$

$$(x_1, x_2, x_3, ..., x_p, x_1^2, x_1x_2, ..., x_1x_p, x_2^2, x_2x_3, ..., x_2x_p, ..., x_p^2)$$

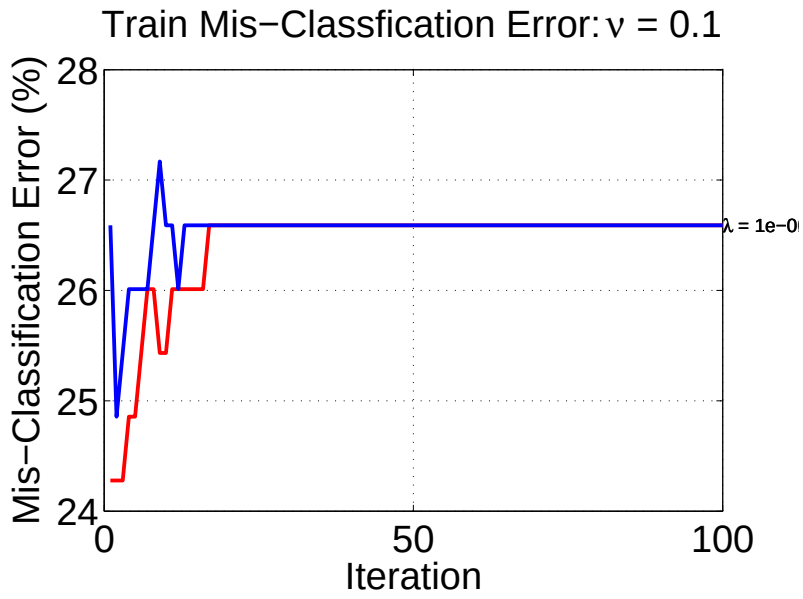
In other words, the original p variables can be expanded to be

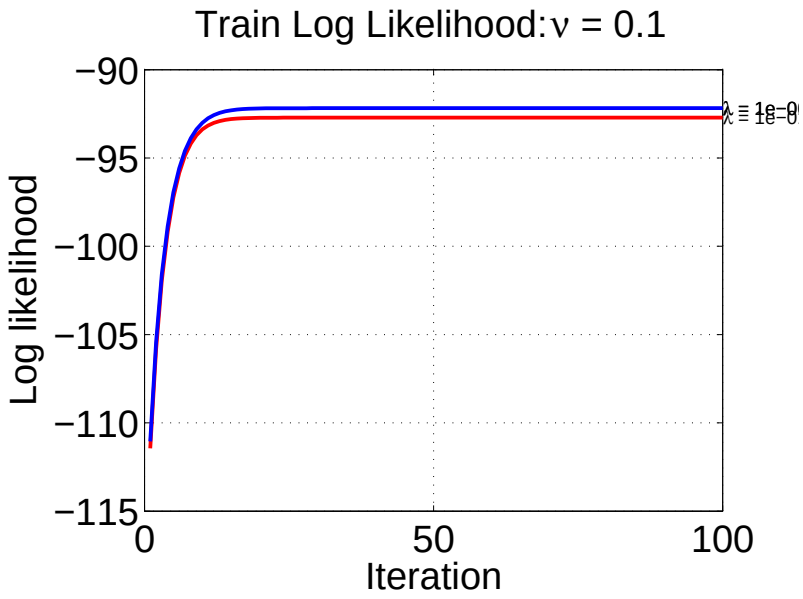
$$p + \frac{p(p+1)}{2} \quad \text{variables}$$

The expansion often helps, but not always. In general, when the number of examples n is large, feature expansion usually helps.

Adding Pairwise Interactions on Crab Data

Adding all pairwise (interaction) variables only help slightly in terms of the log likelihood (red denotes using only the original variables).





Simplify Label Assignments

Recall **label assignment** in logistic regression:

$$\hat{y}_i | x_i = \underset{k}{\text{argmax}} \ \hat{p}_{i,k}$$

and the **probability model** of logistic regression:

$$p_{i,k} = \frac{e^{x_i \beta_k}}{\sum_{s=0}^{K-1} e^{x_i \beta_s}}$$

It is equivalent to assign labels directly by

$$\hat{y}_i | x_i = \underset{k}{\text{argmax}} \ x_i \hat{\beta}_k$$

This raises an interesting question: maybe we don't need a probability model for the purpose of classification? For example, a linear regression may be sufficient?

Linear Regression and Its Applications in Classification

Both linear regression and logistic regression are examples of **Generalized Linear Models (GLM)**.

We first review linear regression and then discuss how to use it for (multi-class) classification.

Review Linear Regression

Given data $\{x_i, y_i\}_{i=1}^n$, where x_i is a p-dimensional vector and y_i is a scalar (not limited to be categories).

We again construct the data matrix

$$\mathbf{X} = \begin{bmatrix} 1 & x_{1,1} & x_{1,2} & \dots & x_{1,p} \\ 1 & x_{2,1} & x_{2,2} & \dots & x_{2,p} \\ \dots & & & & \\ 1 & x_{n,1} & x_{n,2} & \dots & x_{n,p} \end{bmatrix}, \quad \mathbf{y} = \begin{bmatrix} y_1 \\ y_2 \\ \dots \\ y_n \end{bmatrix}$$

The data model is

$$\mathbf{y} = \mathbf{X} \times \beta$$

β (a vector of length $p+1$) is obtained by minimizing the mean square errors (equivalent to maximizing the joint likelihood under the normal distribution model).

Linear Regression Estimation by Least Square

The idea is to minimize the mean square errors

$$MSE(\beta) = \sum_{i=1}^{n} |y_i - x_i\beta|^2 = (\mathbf{Y} - \mathbf{X}\beta)^\mathsf{T} (\mathbf{Y} - \mathbf{X}\beta)$$

We can find the optimal β by setting the first derivative to be zero

$$\frac{\partial MSE(\beta)}{\beta} = \mathbf{X}^\mathsf{T} (\mathbf{Y} - \mathbf{X}\beta) = 0$$

$$\Longrightarrow \mathbf{X}^\mathsf{T}\mathbf{Y} = \mathbf{X}^\mathsf{T}\mathbf{X}\beta$$

$$\Longrightarrow \beta = (\mathbf{X}^\mathsf{T}\mathbf{X})^{-1}\mathbf{X}^\mathsf{T}\mathbf{Y}$$

Don't worry much about how to do matrix derivatives. The trick is to view this simply as a scalar derivative but we need to manipulate the order (and add transposes) to get the dimensions correct.

Ridge Regression

Similar to l_2-regularized logistic regression, we can add a regularization parameter

$$\beta = (\mathbf{X}^\mathsf{T}\mathbf{X} + \lambda\mathbf{I})^{-1}\mathbf{X}^\mathsf{T}\mathbf{Y}$$

which is known as **ridge regression**.

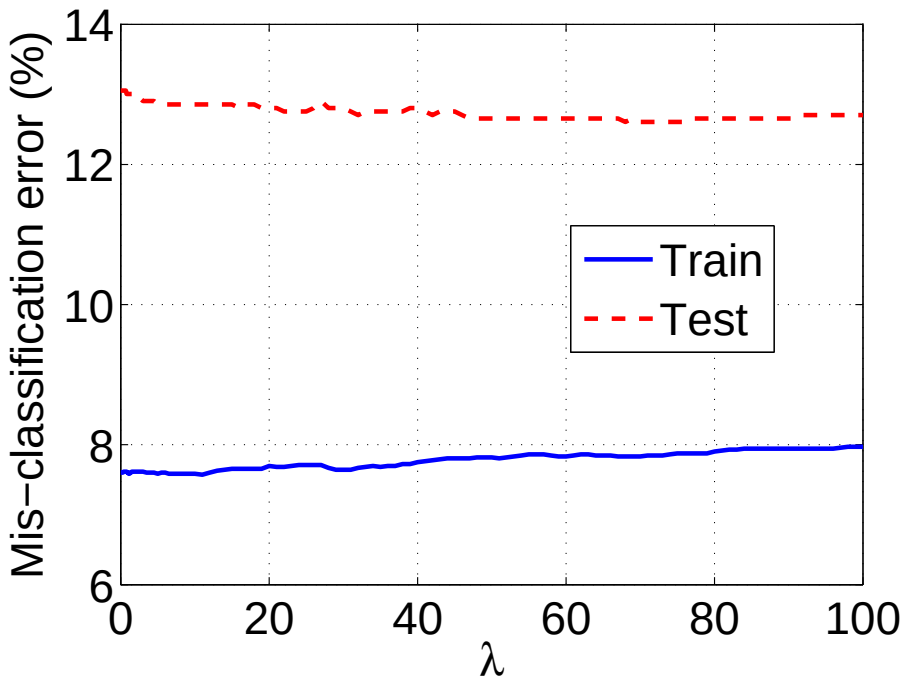
Adding regularization not only improves the numerical stability but also often increases the test accuracy.

Linear Regression for Classification

For binary classification, i.e., $y_i \in \{0, 1\}$, we can simply treat y_i as numerical response and fit a linear regression. To obtain the classification result, we can simply use $\hat{y} = 0.5$ as the classification threshold.

Multi-class classification (with K classes) is more interesting. We can use exactly the same trick as in multi-class logistic regression by first expanding the y_i into a vector of length K with only one entry being 1 and then fitting K binary linear regressions simultaneously and using the location of the maximum fitted value as the class label prediction. Since you have completed the homework in multi-class logistic regression, this idea should be straightforward now. Also see sample code.

Mis-Classification Errors on Zipcode Data



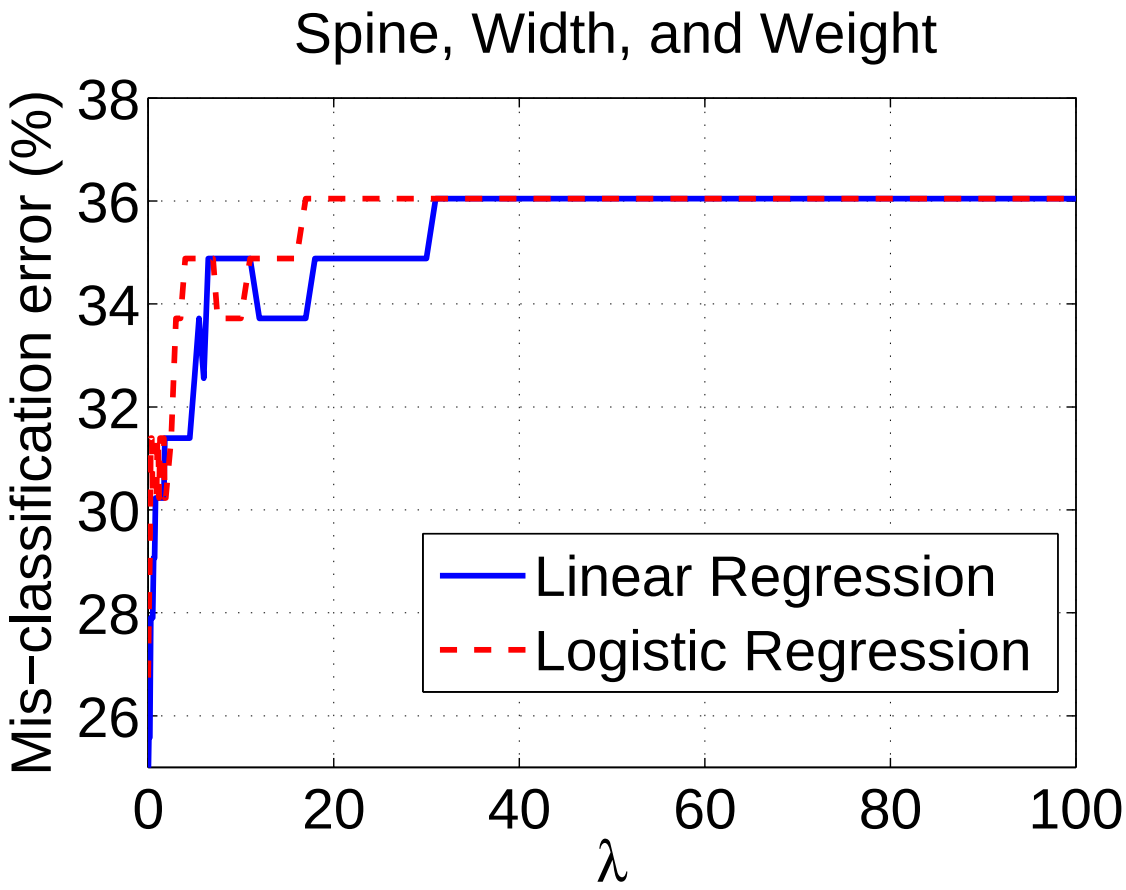
- This is essentially the first iteration of multi-class logistic regression. Clearly, the results are not as good as logistic regression with many iterations.
- Adding regularization (λ) slightly increases the training errors but decreases the testing errors at certain range.

Linear Regression Classification on Crab Data

Binary classification. 50% of the data points are used for training and the rest for testing. Three models are compared:

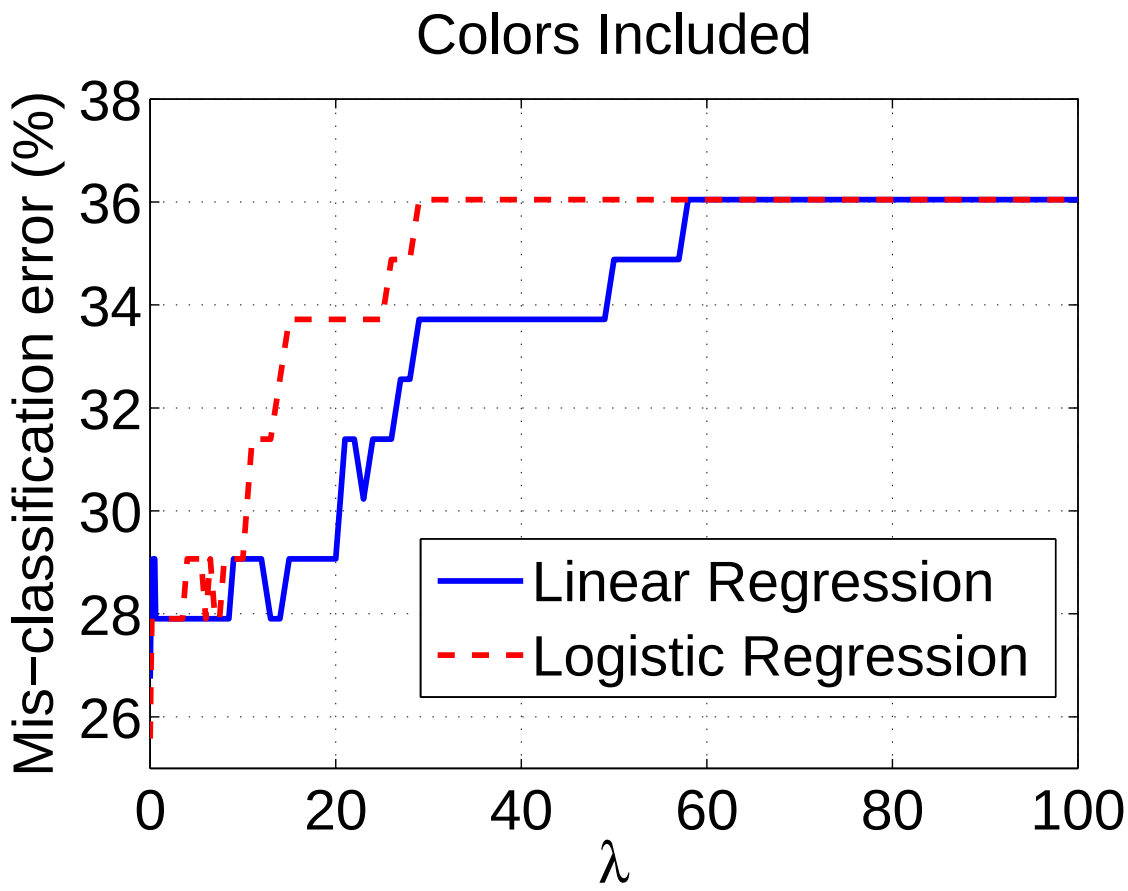
- Model using S, W, and Wt.
- Model using the above three as well as colors.
- Model using all four plus all pairwise interactions.

Both linear regression and logistic regressions are experimented. For logistic regression, we use $\nu = 0.1$ and only report the errors at the 100th iterations



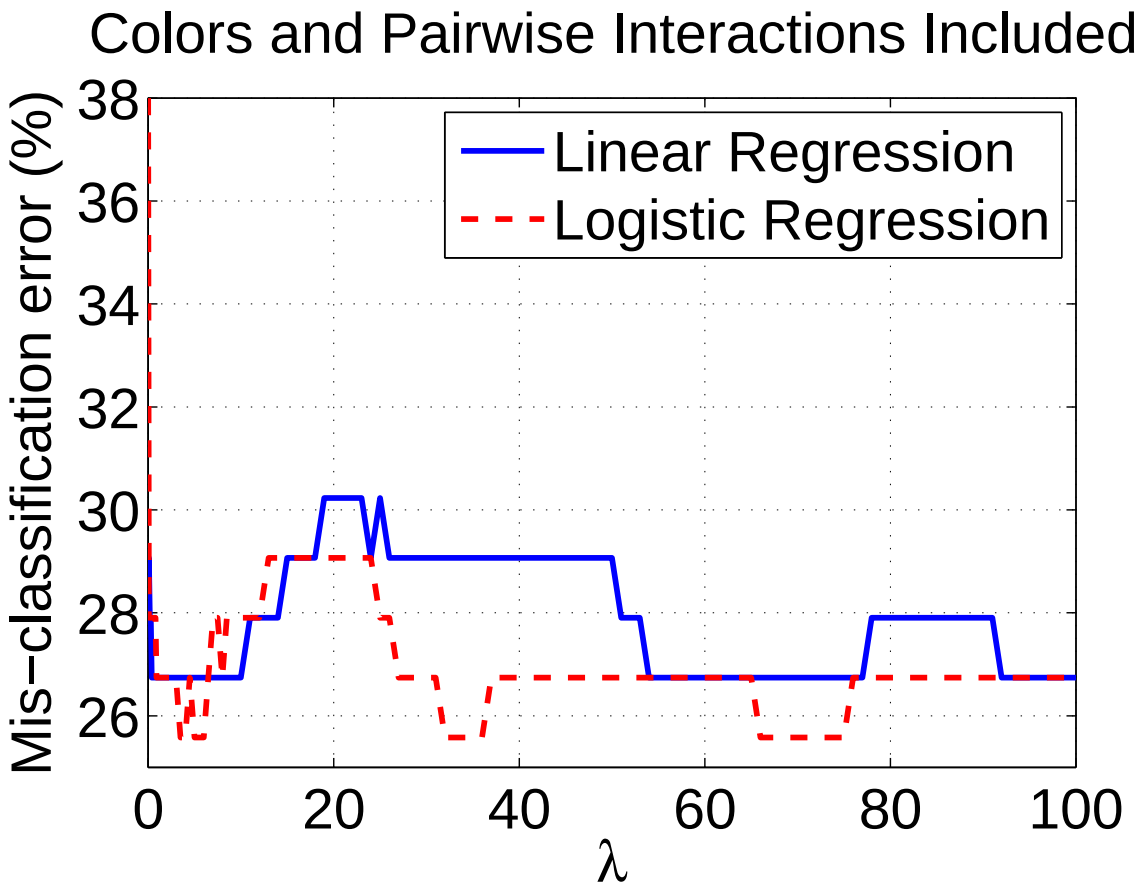
Linear regression and logistic regression produce almost the same results.

Regularization does not appear to be helpful in this example.



Linear regression seems to be even slightly better

Regularization still does not appear to be helpful.



Now logistic regression seems to be slightly better

Regularization really helps. (Why?)

Limitations of Using Linear Regression for Classification

- For many datasets, the classification accuracies of using linear regressions are actually quite similar to using logistic regressions, especially when the datasets are "not so good."
- However, for many "good" datasets (such as zip code data), logistic regressions may have some noticeable advantages.
- Linear regression does not (directly) provide an probabilistic interpretations of the classification results, which may be needed in many applications, for example, learning to rank using classification.

Poisson Log-Linear Model

Revisit the crab data. It appears very natural to model the Sa counts as a **poisson** random variable, which may be parameterized by a linear model.

Color (C)	Spine (S)	Width (W, cm)	Weight (Wt, Kg)	# Saterlites (Sa)
2	3	28.3	3.05	8
3	3	22.5	1.55	0
1	1	26.0	2.30	9
3	3	24.8	2.10	0
3	3	26.0	2.60	4
2	3	23.8	2.10	0
1	1	26.5	2.35	0
3	2	24.7	1.90	0

Poisson Distribution

Denote $Y \sim Poisson(\mu)$. The probability mass function (PMF) is

$$\mathbf{Pr}\,(Y = y) = \frac{e^{-\mu} u^y}{y!}, \qquad y = 0, 1, 2, ...$$

$$E(Y) = \mu, \qquad Var(Y) = \mu$$

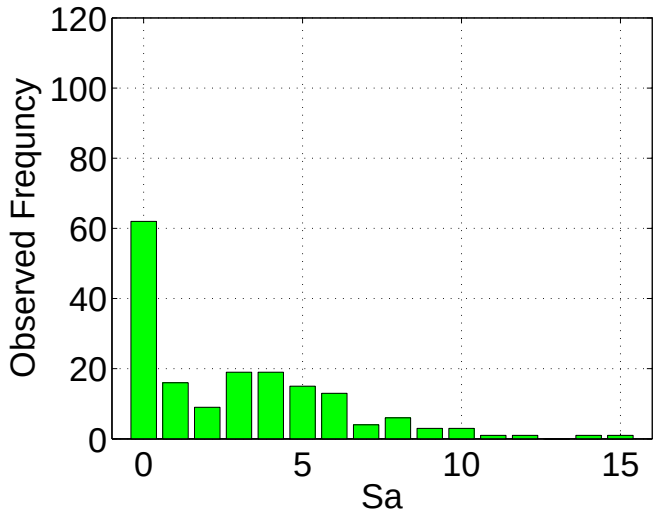
One drawback of the Poisson model is that its variance is the same as the mean which often contradicts real data observations.

Fitting Poisson Distribution

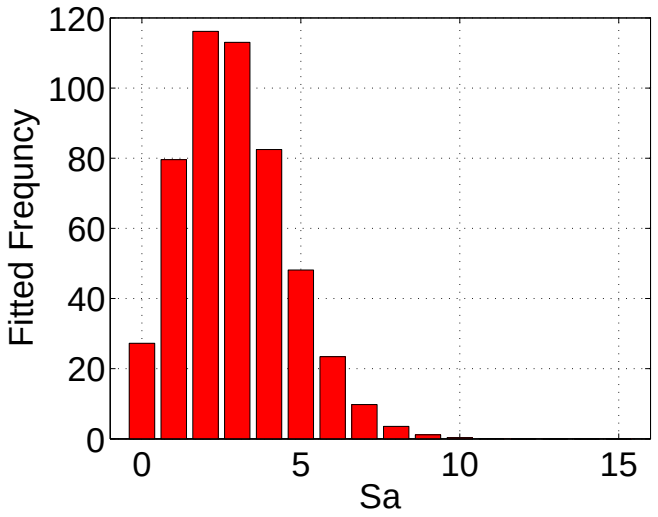
Given n observations, y_i, $i = 1$ to n, the MLE of μ is simply the sample mean:

$$\hat{\mu} = \frac{1}{n} \sum_{i=1}^{n} y_i$$

Observed counts



Fitted counts



No need to perform any test. It is obviously not a good fit.

Linear Regression for Predicting Counts

Maybe we can simply model

$$y_i \sim N(\mu_i, \sigma^2)$$
$$\mu_i = x_i\beta = \beta_0 + x_{i,1}\beta_1 + ... + x_{i,p}\beta_p$$

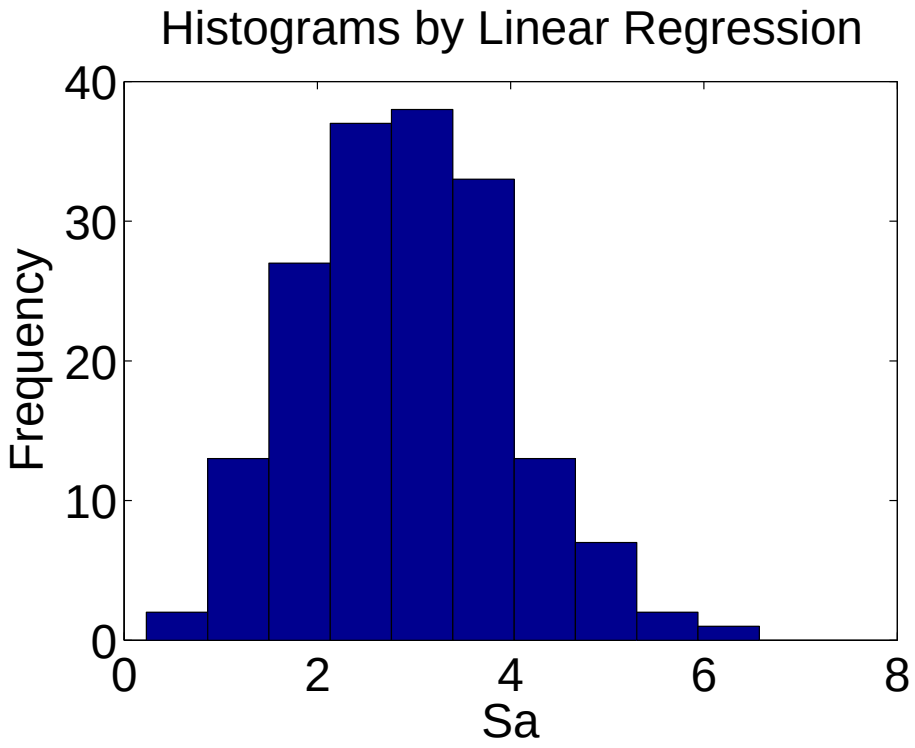
i.e., μ_i is the mean of a normal distribution $N(\mu_i, \sigma^2)$.

This way, we can easily predict the counts by

$$\hat{\beta} = \left(\mathbf{X}^\mathsf{T}\mathbf{X}\right)^{-1}\mathbf{X}^\mathsf{T}\mathbf{y}$$
$$\hat{\mathbf{y}} = \mathbf{X}\hat{\beta}$$

Histograms of the predictions of counts by using linear regression using only the width. The sum of square error (SE) is

$$SE = \sum_{i=1}^{n} (\hat{y}_i - y_i)^2 = 1.5079 \times 10^3$$



Clearly, linear regression can not possibly be the best approach.

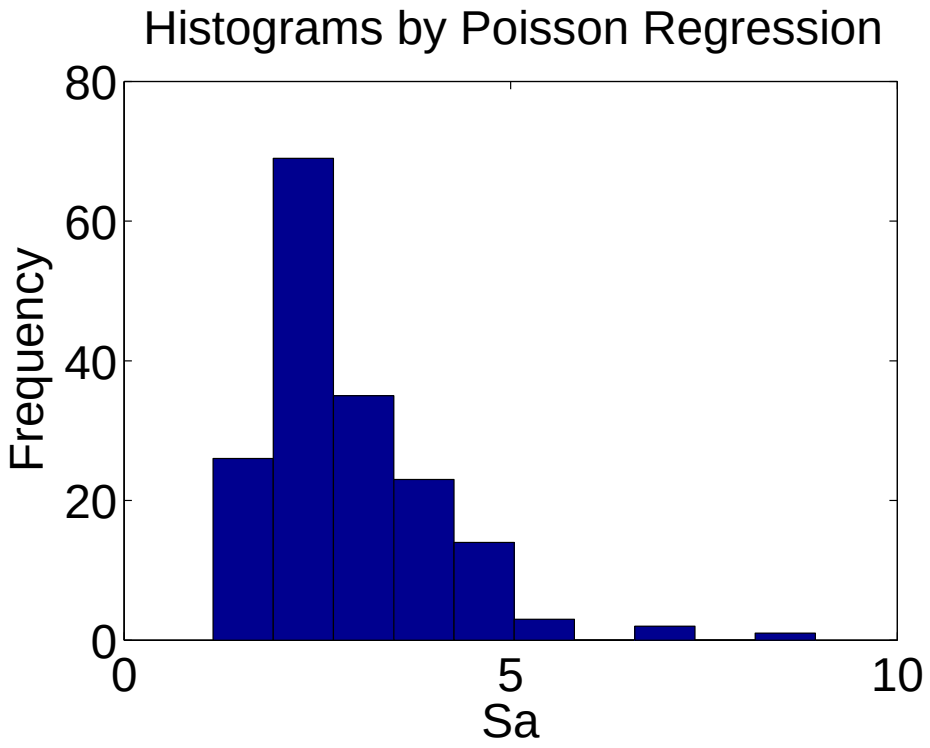
Poisson Regression Model

Assumption:

$$y_i \sim Poisson(\mu_i)$$

$$\log \mu_i = x_i\beta = \beta_0 + x_{i,1}\beta_1 + ... + x_{i,p}\beta_p$$

Note that this is very different from assuming that the logarithms of the counts follow a linear regression model. Why?



Clearly, this looks better than the histogram from linear regression.

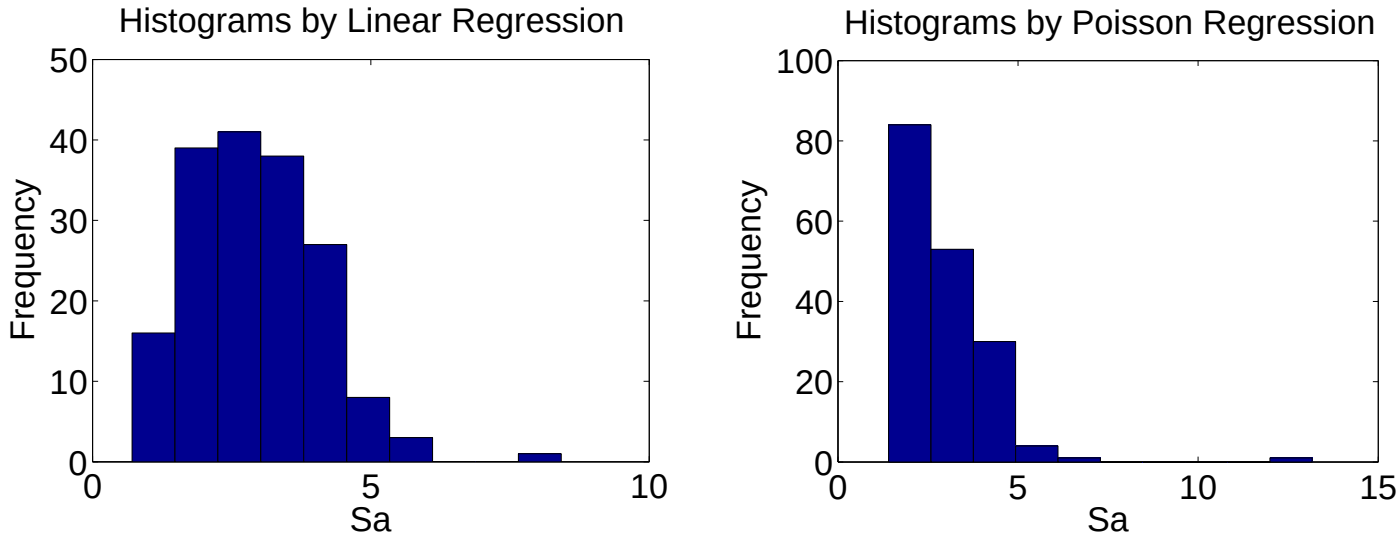
However, the square error $SE = \sum_{i=1}^{n} (\hat{y}_i - y_i)^2 = 1.5373 \times 10^3$ is actually larger than the SE from linear regression. Why is it not too surprising?

Comparing Fitted Counts

y	$\hat{y}$ Linear	$\hat{y}$ Poisson
8.0000	3.9344	3.8103
0	0.9916	1.4714
9.0000	2.7674	2.6127
0	2.1586	2.1459
4.0000	2.7674	2.6127
0	1.6512	1.8212
0	3.0211	2.8361
0	2.1079	2.1110
0	1.6005	1.7916
0	2.5645	2.4468

Need to see more rows to understand the differences...

Now we use 3 variables, S, W, Wt, to fit linear regression and Poisson regression.



Clearly, Poisson regression looks better, although SE values are 1.4696×10^3 and 1.5343×10^3, respectively, for linear regression and Poisson regression.

Fitting Poisson Regression

Log Likelihood:

$$l_i = -\mu_i + y_i \log \mu_i = -e^{x_i \beta} + y_i x_i \beta$$

First Derivatives:

$$\frac{\partial l_i}{\partial \beta} = (y_i - \mu_i)\, x_i^{\mathsf{T}}$$

Given n observations, the log likelihood is $l = \sum_{i=1}^{n} l_i$.

First Derivatives (matrix form) :

$$\frac{\partial \mathbf{l}}{\partial \beta} = \mathbf{X}^{\mathsf{T}} (\mathbf{y} - \mu)$$

Second Derivatives (matrix form) :

$$\frac{\partial^{\mathbf{2}} \mathbf{l}}{\partial \beta \beta^{\mathsf{T}}} = -\mathbf{X}^{\mathsf{T}} \mathbf{W} \mathbf{X}$$

where W is the diagonal matrix of μ.

They look very similar to logistic regression.

Newton's Method for Solving Poisson Regression Model

$$\beta^{new} = \beta^{old} - \left[\left(\frac{\partial^2 l(\beta)}{\partial \beta \partial \beta^{\mathsf{T}}} \right)^{-1} \frac{\partial l(\beta)}{\partial \beta} \right]_{\beta^{old}}$$

$$\beta^{\mathbf{t}} = \beta^{(\mathbf{t-1})} + \nu \left[\mathbf{X^T W X} \right]^{-\mathbf{1}} \left[\mathbf{X^T} (\mathbf{y} - \mu) \right] \Big|_{\mathbf{t-1}}$$

where again ν (e.g., 0.1) is a shrinkage parameter which helps the numerical stability.

Why "Log Linear"?

Poisson Model Without Log:

$$y_i \sim Poisson(\mu_i)$$
$$\mu_i = x_i\beta = \beta_0 + x_{i,1}\beta_1 + ... + x_{i,p}\beta_p$$

Its log likelihood and first derivative (assuming only one β) are:

$$l_i = -\mu_i + y_i \log \mu_i = -x_i\beta + y_i \log (x_i\beta)$$
$$\frac{\partial l_i}{\partial \beta} = -x_i + \frac{y_i x_i}{x_i\beta}$$

Considering the second derivatives and more than one β, using this model is almost like "looking for troubles." There is also another obvious issue with this model. What is it?

The reason why "Log Linear" will be more clear under the GLM framework.

Summary of Models

Given a dataset $\{x_i, y_i\}_{i=1}^n$, so far, we have seen three different models:

- **Linear Regression** $-\infty < y_i < \infty$,

$$y_i \sim N\left(\mu_i, \sigma^2\right), \qquad \mu_i = x_i\beta$$

- **Poisson Regression** $y_i \in \{0, 1, 2, ..., \}$,

$$y_i \sim Poisson\left(\mu_i\right), \qquad \log \mu_i = x_i\beta$$

- **Binary Logistic Regression** $y_i \in \{0, 1\}$,

$$y_i \sim Binomial\left(p_i\right), \qquad \log \frac{p_i}{1-p_i} = x_i\beta$$

Quotes from George E. P. Box

- **Essentially, all models are wrong, but some are useful.**

- **Remember that all models are wrong; the practical question is how wrong do they have to be to not be useful.**

Generalized Linear Models (GLM)

All the models we have seen so far belong to the family of generalized linear models (GLM). In general, a GLM consists of three components:

- **The random component** $y_i \sim f(y_i; \theta_i)$.

$$f(y_i; \theta_i) = a(\theta_i) b(y_i) e^{y_i Q(\theta_i)}$$

- **The systematic component** $\eta_i = x_i \beta = \sum_{j=0}^{p} x_{i,j} \beta_j$.

  (This may be replaced by a more flexible model.)

- **The link function** $\eta_i = g(u_i)$, where $u_i = E(y_i)$.

  $g(u)$ is a monotonic function. If $g(u) = u$, it is called "identity link".

Revisit Poisson Log Linear Model Under GLM

For GLM,

$$y_i \sim f(y_i;\theta_i) = a(\theta_i) \times b(y_i) \times e^{y_i Q(\theta_i)}$$

In this case, $\theta_i = u_i$,

$$f(y_i) = \frac{e^{-u_i} u_i^{y_i}}{y_i!} = \left[e^{-u_i}\right] \left[\frac{1}{y_i}\right] \left[e^{y_i \log u_i}\right]$$

Therefore,

$$a(\mu_i) = e^{-u_i}, \qquad b(y_i) = \frac{1}{y_i!}, \qquad Q(\mu_i) = \log u_i$$

And the link function

$$g(u_i) = Q(\theta_i) = \log u_i = x_i\beta$$

This is called canonical link.

Revisit Binary Logistic Model Under GLM

For GLM,

$$y_i \sim f(y_i;\theta_i) = a(\theta_i) \times b(y_i) \times e^{y_i Q(\theta_i)}$$

In this case, $\theta_i = p_i$,

$$f(y_i) = p_i^{y_i}(1-p_i)^{1-y_i} = [(1-p_i)]\,[1]\left[e^{y_i \log \frac{p_i}{1-p_i}}\right]$$

Therefore,

$$a(p_i) = 1-p_i, \qquad b(y_i) = 1, \qquad Q(p_i) = \log\frac{p_i}{1-p_i}$$

And the link function

$$g(p_i) = Q(\theta_i) = \log\frac{p_i}{1-p_i} = x_i\beta$$

This is again a canonical link.

Revisit Linear Regression Model Under GLM (with $\sigma^2 = 1$)

For GLM,

$$y_i \sim f(y_i; \theta_i) = a(\theta_i) \times b(y_i) \times e^{y_i Q(\theta_i)}$$

In this case, $\theta_i = \mu_i$ (and $\sigma^2 = 1$ by assumption)

$$f(y_i) = e^{-\frac{(y_i - \mu_i)^2}{2}} = \left[e^{-\frac{\mu_i^2}{2}}\right] \left[e^{-\frac{y_i^2}{2}}\right] \left[e^{y_i \mu_i}\right]$$

Therefore,

$$a(\mu_i) = e^{-\frac{\mu_i^2}{2}}, \qquad b(y_i) = e^{-\frac{y_i^2}{2}}, \qquad Q(\mu_i) = \mu_i$$

And the link function

$$g(u_i) = Q(\theta_i) = \mu_i = x_i \beta$$

This is again a canonical link and is in fact an identity link.

Statistical Inference

After we have fitted a GLM (e.g., logistic regression) and estimated the coefficients $\hat{\beta}$, we can ask many questions, such as

- Which β_j is more important?
- Is β_j significantly different from 0?
- What is the (joint) distribution of β?

To understand these questions, it is crucial to learn some theory of the MLE, because fitting a GLM is finding the MLE for a particular distribution.

Revisit the Maximum Likelihood Estimation (MLE)

Observations x_i, $i = 1$ to n, are i.i.d. samples from a distribution with probability density function $f_X(x; \theta_1, \theta_2, ..., \theta_k)$,
where θ_j, $j = 1$ to k, are parameters to be estimated.

The maximum likelihood estimator seeks the θ to maximize the joint likelihood

$$\hat{\theta} = \underset{\theta}{\text{argmax}} \prod_{i=1}^{n} f_X(x_i; \theta)$$

Or, equivalently, to maximize the log joint likelihood

$$\hat{\theta} = \underset{\theta}{\text{argmax}} \sum_{i=1}^{n} \log f_X(x_i; \theta) = \underset{\theta}{\text{argmax}}\ l(\theta; x)$$

where $l(\theta; x) = \sum_{i=1}^{n} \log f_X(x_i; \theta)$ is the joint log likelihood function.

Large Sample Theory for MLE

Large sample theory says, as $n \to \infty$, $\hat{\theta}$ is asymptotically unbiased and normal.

$$\hat{\theta} \sim N\left(\theta, \frac{1}{nI(\theta)}\right), \qquad \text{approximately}$$

$I(\theta)$ is the Fisher Information of θ:

$$I(\theta) = -E\left[\frac{\partial^2}{\partial\theta^2}\log f(X|\theta)\right] = -E\left(l''(\theta)\right)$$

Note that it is also true that

$$I(\theta) = E\left(l'(\theta)\right)^2$$

but you don't have to worry about the proof.

Intuition About the Asymptotic Distributions & Variances of MLE

The MLE $\hat{\theta}$ is the solution to the MLE equation $l'(\hat{\theta}) = 0$.

The Taylor expansion around the true θ

$$l'(\hat{\theta}) \approx l'(\theta) + (\hat{\theta} - \theta) l''(\theta)$$

Let $l'(\hat{\theta}) = 0$ (because $\hat{\theta}$ is the MLE solution)

$$(\hat{\theta} - \theta) \approx -\frac{l'(\theta)}{l''(\theta)}$$

We know that

$$E(-l''(\theta)) = nI(\theta) = E(l'(\theta))^2,$$

$E(l'(\theta)) = 0.$ (Read the next slide if interested in the proof)

(Don't worry about this slide if you are not interested.)

$$l'(\theta) = \sum_{i=1}^{n} \frac{\partial \log f(x_i)}{\partial \theta} = \sum_{i=1}^{n} \frac{\frac{\partial f(x_i)}{\partial \theta}}{f(x_i)}$$

$$E\left(l'(\theta)\right) = \sum_{i=1}^{n} E\left(\frac{\partial \log f(x_i)}{\partial \theta}\right) = nE\left(\frac{\frac{\partial f(x)}{\partial \theta}}{f(x)}\right) = 0$$

because

$$E\left(\frac{\frac{\partial f(x)}{\partial \theta}}{f(x)}\right) = \int \frac{\frac{\partial f(x)}{\partial \theta}}{f(x)} f(x)dx = \int \frac{\partial f(x)}{\partial \theta} dx = \frac{\partial}{\partial \theta} \int f(x)dx = 0$$

The heuristic trick is to approximate

$$\hat{\theta} - \theta \approx \frac{l'(\theta)}{-l''(\theta)} \approx \frac{l'(\theta)}{E(-l''(\theta))} = \frac{l'(\theta)}{nI(\theta)}$$

Therefore,

$$E(\hat{\theta} - \theta) \approx \frac{E(l'(\theta))}{nI(\theta)} = 0$$

$$Var(\hat{\theta}) \approx E(\hat{\theta} - \theta)^2 \approx E\left(\frac{l'(\theta)}{nI(\theta)}\right)^2 = \frac{nI(\theta)}{n^2 I^2(\theta)} = \frac{1}{nI(\theta)}$$

This is why intuitively, we know that $\hat{\theta} \sim N\left(\theta, \frac{1}{nI(\theta)}\right)$.

Example: Normal Distribution

Given n i.i.d. samples, $x_i \sim N(\mu, \sigma^2)$, $i = 1$ to n.

$$\log f_X(x; \mu, \sigma^2) = -\frac{1}{2\sigma^2}(x-\mu)^2 - \frac{1}{2}\log(2\pi\sigma^2)$$

$$\frac{\partial^2 \log f_X(x; \mu, \sigma^2)}{\partial \mu^2} = -\frac{1}{\sigma^2} \Longrightarrow \mathrm{I}(\mu) = \frac{1}{\sigma^2}$$

Therefore, the MLE $\hat{\mu}$ will have asymptotic variance $\frac{1}{nI(\mu)} = \frac{\sigma^2}{n}$. But in this case, we already know that

$$\hat{\mu} = \frac{1}{n}\sum_{i=1}^{n} x_i \sim N\left(\mu, \frac{\sigma^2}{n}\right)$$

In other words, the "asymptotic" variance of the MLE is in fact exact in this case.

Example: Binomial Distribution

$x \sim$ Binomial(p, n): $\mathbf{Pr}(x = k) = \binom{n}{k} p^k (1-p)^{n-k}$

Log likelihood and Fisher Information:

$$l(p) = k \log p + (n-k) \log(1-p)$$

$$l'(p) = \frac{k}{p} - \frac{n-k}{1-p} \Longrightarrow \text{MLE} \quad \hat{p} = \frac{k}{n}$$

$$l''(p) = -\frac{k}{p^2} - \frac{n-k}{(1-p)^2}$$

$$\mathsf{I}(p) = -\mathsf{E}\left(l''(p)\right) = \frac{np}{p^2} + \frac{n-np}{(1-p)^2} = \frac{n}{p(1-p)}$$

That is, the asymptotic variance of the MLE $\hat{p}$ is $\frac{p(1-p)}{n}$, which is in fact again the exact variance.

Example: Contingency Table with Known Margins

$n = n_{11} + n_{12} + n_{21} + n_{22}$

n_{11}	n_{12}
n_{21}	n_{22}

$N = N_{11} + N_{12} + N_{21} + N_{22}$

N_{11}	N_{12}
N_{21}	N_{22}

Margins: $M_1 = N_{11} + N_{12}$, $M_2 = N_{11} + N_{21}$, are known.

The (asymptotic) variance of the MLE (for N_{11}) is

$$\text{Var}\left(\hat{N}_{11,MLE}\right) = \frac{N/n}{\frac{1}{N_{11}} + \frac{1}{M_1 - N_{11}} + \frac{1}{M_2 - N_{11}} + \frac{1}{N - M_1 - M_2 + N_{11}}}$$

Derivation: The log likelihood is

$$\begin{aligned} l(N_{11}) =& n_{11} \log \frac{N_{11}}{N} + n_{12} \log \frac{M_1 - N_{11}}{N} \\ & + n_{21} \log \frac{M_2 - N_{11}}{N} + n_{22} \log \frac{N - M_1 - M_2 + N_{11}}{N} \end{aligned}$$

The MLE solution is

$$l'(N_{11}) = \frac{n_{11}}{N_{11}} - \frac{n_{12}}{M_1 - N_{11}} - \frac{n_{21}}{M_2 - N_{11}} + \frac{n_{22}}{N - M_1 - M_2 + N_{11}} = 0$$

The second derivative is

$$l''(N_{11}) = - \frac{n_{11}}{N_{11}^2} - \frac{n_{12}}{N_{12}^2} - \frac{n_{21}}{N_{21}^2} - \frac{n_{22}}{N_{22}^2}$$

The Fisher Information is thus

$$\begin{aligned} I(N_{11}) =& E(-l''(N_{11})) = \frac{E(n_{11})}{N_{11}^2} + \frac{E(n_{12})}{N_{12}^2} + \frac{E(n_{21})}{N_{21}^2} + \frac{E(n_{22})}{N_{22}^2} \\ =& \frac{n}{N}\left[\frac{1}{N_{11}} + \frac{1}{N_{12}} + \frac{1}{N_{21}} + \frac{1}{N_{22}}\right] \end{aligned}$$

Recall

$$E(n_{11}) = n\frac{N_{11}}{N}, \qquad E(n_{12}) = n\frac{N_{12}}{N},$$
$$E(n_{21}) = n\frac{N_{21}}{N}, \qquad E(n_{22}) = n\frac{N_{22}}{N},$$

Asymptotic Covariance Matrix

More generally, suppose there are more than one parameters $\theta = \{\theta_1,\ \theta_2,\ \theta_3,\ \theta_p\}$. The Fisher Information Matrix is defined as

$$I(\theta) = E\left(-\frac{\partial^2 l(\theta)}{\partial\theta_i \partial\theta_j}\right)$$

And the asymptotic covariance matrix is

$$Cov(\hat{\theta}) = I^{-1}(\theta)$$

Review Binary Logistic Regression Derivatives

Newton' update formula

$$\beta^{new} = \beta^{old} - \left[\left(\frac{\partial^2 l(\beta)}{\partial \beta \partial \beta^{\mathsf{T}}} \right)^{-1} \frac{\partial l(\beta)}{\partial \beta} \right]_{\beta^{old}}$$

where, in a matrix form

$$\frac{\partial l(\beta)}{\partial \beta} = \sum_{i=1}^{n} x_i \left(y_i - p(x_i; \beta) \right) = \mathbf{X^T}(\mathbf{y} - \mathbf{p})$$

$$\frac{\partial^2 l(\beta)}{\partial \beta \partial \beta^{\mathsf{T}}} = -\sum_{i=1}^{n} x_i^{\mathsf{T}} x_i p(x_i; \beta) \left(1 - p(x_i; \beta) \right) = -\mathbf{X^T W X}$$

where $W = diag\{p(x_i)(1 - p(x_i))\}$.

Fisher Information and Covariance for Logistic Regression

Suppose the Newton's iteration has reached the optimal solution (very important), then

$$\mathbf{I}(\beta) = \mathbf{E}\left(\mathbf{X^T W X}\right) = \mathbf{X^T W X}$$

And the asymptotic covariance matrix is

$$\mathbf{Cov}(\hat{\beta}) = \mathbf{I^{-1}}(\beta) = \left[\mathbf{X^T W X}\right]^{-1}$$

In other words, the MLE estimates $\hat{\beta}$ of the binary logistic regression parameters are asymptotically jointly normal

$$N\left(\beta, \left[\mathbf{X^T W X}\right]^{-1}\right)$$

A Simple Test for Logistic Regression Coefficients

At convergence, the coefficients of logistic regression

$$\beta \sim N\left(\beta, \left[\mathbf{X^T W X}\right]^{-1}\right)$$

We can just test each coefficient separately because, asymptotically

$$\beta_j \sim N\left(\beta_j, \left[\mathbf{X^T W X}\right]^{-1}_{jj}\right)$$

which allows us to use normal probability functions to compute the p-values.

Two caveats: (1) We need the "true" W, which is replaced by the estimated W at the last iteration. (2) We still have to specify the true β_j for the test. In general, it makes sense to test $H_0 : \beta_j = 0$.

GLM with R

```
> data= read.table("d:\\class\\6030Spring12\\fig\\crab.txt");
> model = glm((data[,5]==0)~data$V2+data$V3+data$V4,family='binomial');
> summary(model)
Call:
glm(formula = (data[, 5] == 0) ~ data$V2 + data$V3 + data$V4,
    family = "binomial")

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-1.7120  -0.8948  -0.5242   1.0431   2.0833

Coefficients:
            Estimate Std. Error z value Pr(>|z|)
(Intercept)  9.46885    3.56974   2.653  0.00799 **
data$V2     -0.04952    0.22094  -0.224  0.82267
data$V3     -0.30540    0.18220  -1.676  0.09370 .
data$V4     -0.84479    0.67369  -1.254  0.20985
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for binomial family taken to be 1)
```

```
    Null deviance: 225.76  on 172  degrees of freedom
Residual deviance: 192.84  on 169  degrees of freedom
AIC: 200.84

Number of Fisher Scoring iterations: 4
```

R Resources

Download R executable from

`http://www.r-project.org/`

After launching R, type “glm.help()” for the helper screen.

Validating the Asymptotic Theory Using Crab Data

We use 3 variables (Width, Weight, Spine, plus the intercept) from the crab data for building the binary logistic regression model for predicting $\mathbf{Pr}(Sa > 0)$. Instead of using the original labels, we generate the "true" β and sample the labels from the generated β.

```
function TestLogitCrab;


load crab.txt;
X = crab(:,1:end-1); X(:,1)=1;
be_true = [-10,0.05,0.3,0.8]' + randn(4,1)*0.1;
```

The true β is fixed once generated. Once β is known, we can easily compute

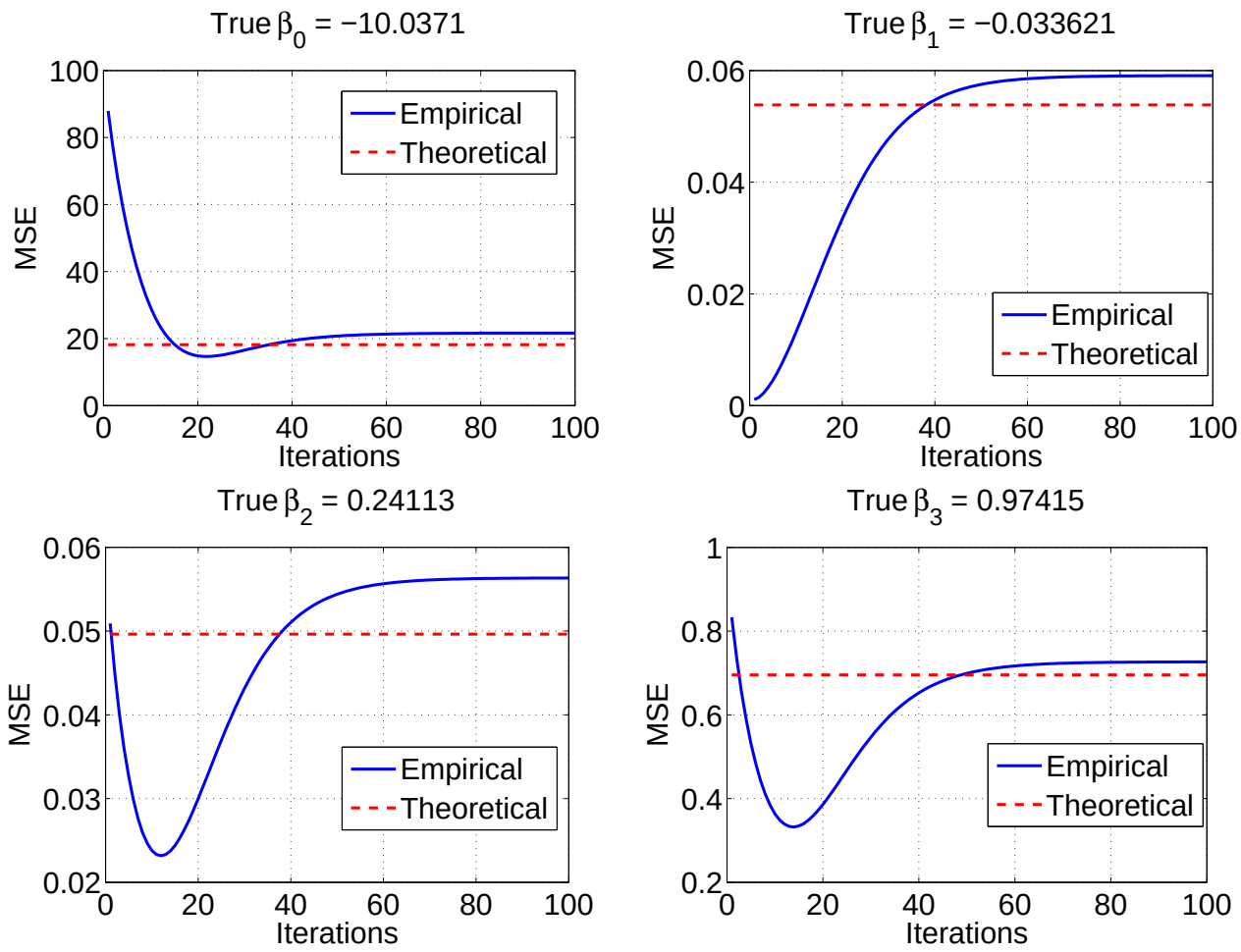
$$p(x_i) = \mathbf{Pr}(y_i = 1) = \frac{e^{x_i\beta}}{1 + e^{x_i\beta}}$$

Once β is fixed, we can compute p and sample the labels from $Bounoulli(p_{(x_i)})$ for each x_i.

We then fit the binary logistic regression using the original x_i and the generated y_i to obtain $\hat{\beta}$, which will be quite close to but not identical to the "true" β.

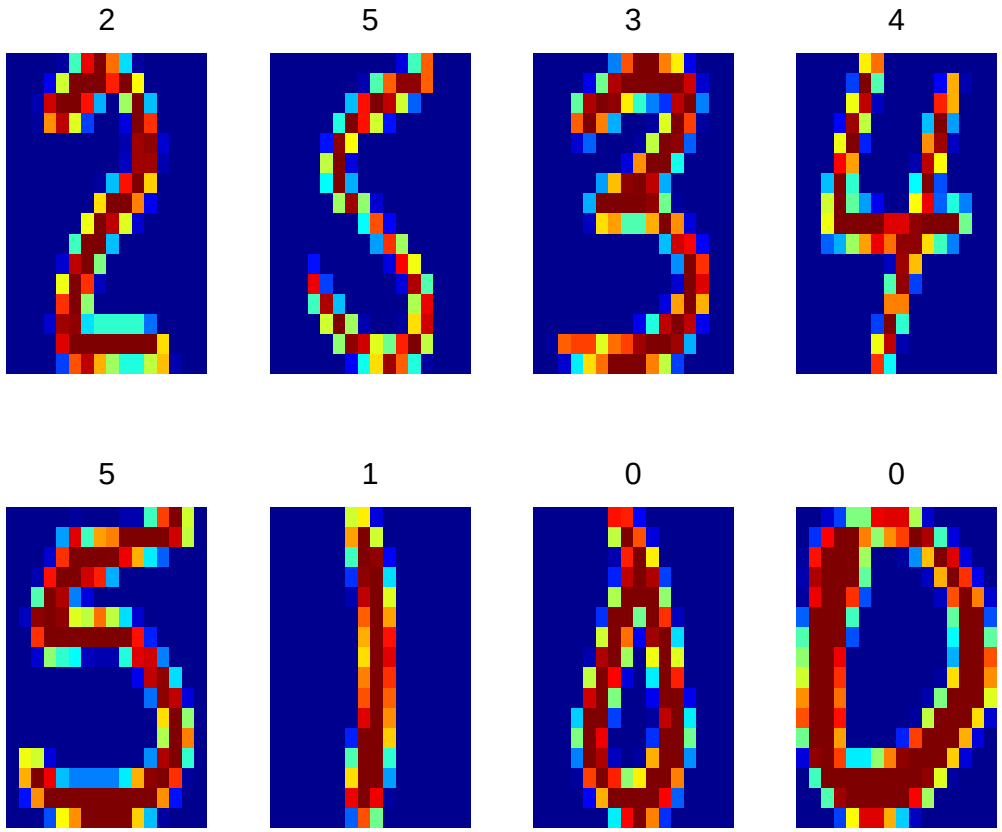
We then repeat the sampling procedure to create another set of labels and $\hat{\beta}$.

By repeating this procedure 1000 times, we will be able to assess the distribution of $\hat{\beta}$.



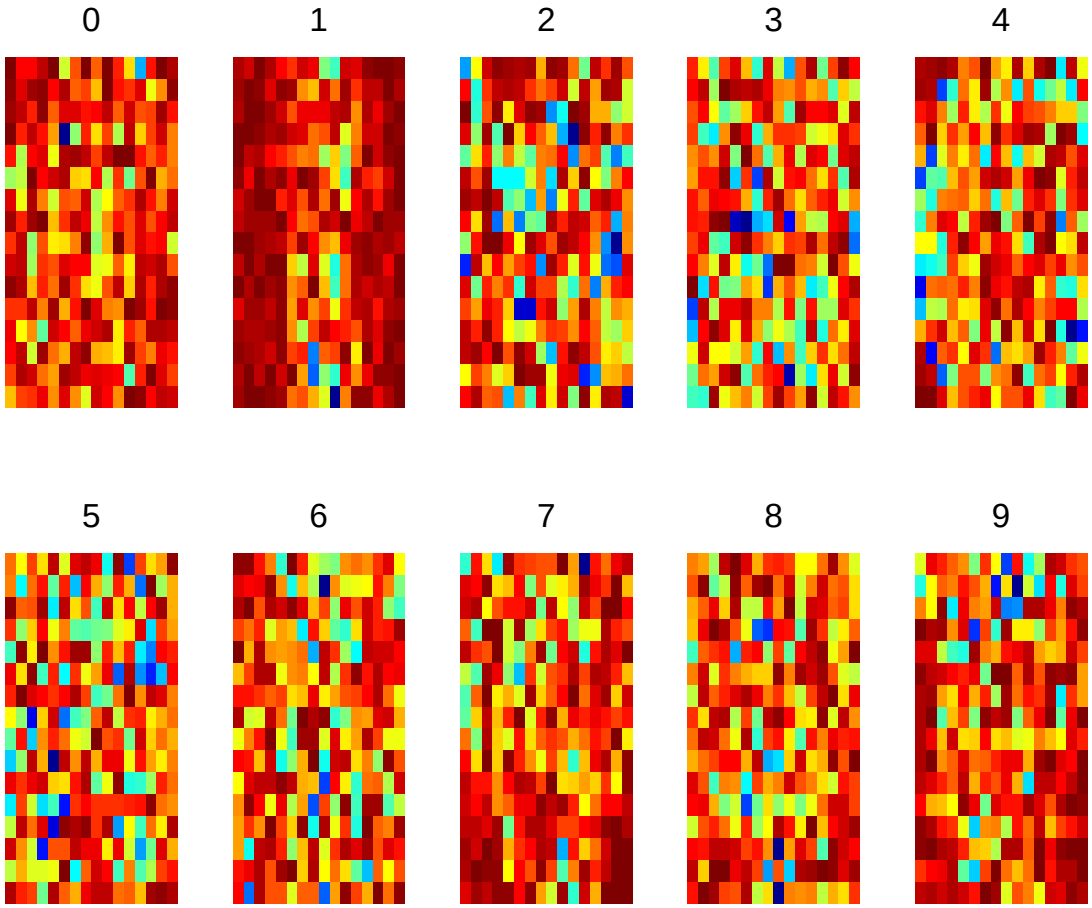
The MSEs for all β_j MSEs converge with increasing iterations. However, they deviate from the "true" variances predicted by $[X^{\mathsf{T}}WX]^{-1}$, most likely because our sample size $n = 173$ is too small for the large-sample theory to be accurate.

Experiments on the Zipcode data

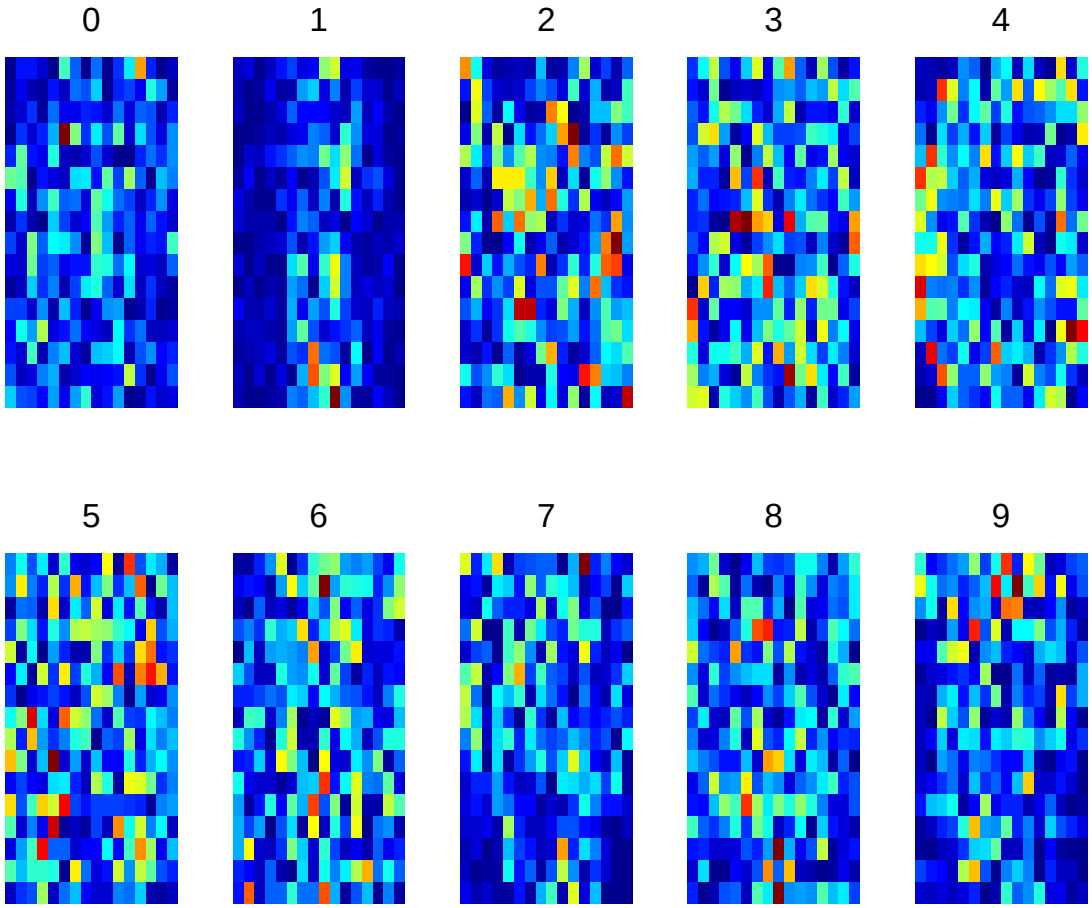


Conjecture: If we display the p-values from the z-test, we might be able to see some images similar to digits.

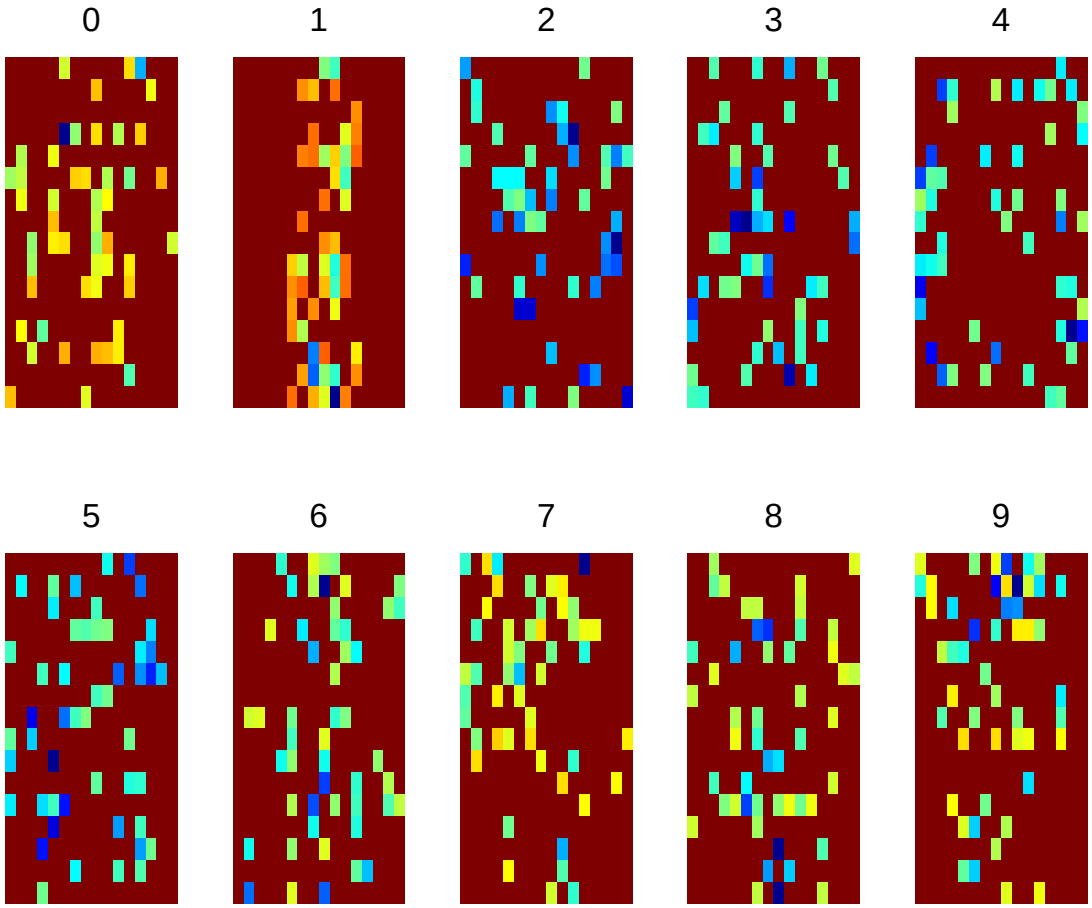
Displaying the p-values as images



Displaying 1- p-values as images



Displaying only top (smallest) 50 p-values as images



Plausible Interpretations: The asymptotic theory says

$$\beta \sim N\left(\beta, \left[\mathbf{X^T W X}\right]^{-1}\right)$$

Using only the marginal (diagonal) information

$$\beta_j \sim N\left(\beta_j, \left[\mathbf{X^T W X}\right]^{-1}_{jj}\right)$$

may result in serious loss of information. In particular, when the variables are highly correlated as in this dataset, it is not realistic to expect that only the marginal information will be sufficient.

In other words, for zipcode data, many pixels "work together" to provide strong discriminating powers. This is the power of **team work**.

Testing Logistic Regression Using Residuals

Recall the (generalized) log-likelihood test

$$-2\log\left(\frac{\text{Maximum likelihood under } H_0}{\text{Maximum likelihood with no restriction}}\right)$$

which is asymptotically distributed as χ^2_k with k determined by the degree of freedom (red df):

$$df = \text{number of parameters to be estimated without restrictions} - \text{number of parameters to be estimated under } H_0$$

This, called **deviance** in the context of logistic regression and GLM, can be used for (often more accurate) testing of the fitted models.

Deviance Residuals for Binomial Logistic Regression

$$D\left(y;\hat{p}\right) = -2\left(l(\hat{p};y) - l(y;y)\right)$$

where $l(\hat{p};y)$ denotes the log-likelihood of the fitted model and $l(y;y)$ denotes the log-likelihood of the **saturated model**:

$$l(\hat{p};y) = \sum_{i=1}^{n}(1-y_i)\log(1-\hat{p}(x_i)) + y_i \log \hat{p}(x_i)$$

$$l(y;y) = \sum_{i=1}^{n}(1-y_i)\log(1-y_i) + y_i \log y_i \qquad \text{(Note } 0\log 0 = 0\text{)}$$

$$D(\hat{p};y) = -2\sum_{i=1}^{n}(1-y_i)\log\frac{1-\hat{p}(x_i)}{1-y_i} - 2\sum_{i=1}^{n} y_i \log\frac{\hat{p}(x_i)}{y_i}$$

Deviance Residual for Un-Grouped (Crab) Data

It is easy to see that $l(y; y) = 0$ always. Therefore, with ungrouped data, we always have

$$D(\hat{p}; y) = -2 \times l(\hat{p}; y)$$

$$= -2 \sum_{i=1}^{n} (1 - y_i) \log(1 - \hat{p}(x_i)) - 2 \sum_{i=1}^{n} y_i \log \hat{p}(x_i)$$

For the crab data, the value is $D(\hat{p}; y) = 192.84$ and $df = 173 - 4 = 169$. The p-value is 0.1009, which is an indication that this model may not be very good.

Multi-Class Ordinal Logistic Regression

For zip-code recognition, it is natural to treat each class (0 to 9) equally, because in general there are indeed no orders among them (unless you are doing specific studies in which the zip code information reveals physical locations.).

In many applications, however, there are natural orders among the class labels. For example, in the crab data, it might be reasonable to consider # SA is ordinal because it reflects the growth process. Also, it variable “Spine condition” may be also ordinal.

Another example is the Webpage relevance ranking. A page with a rank of “perfect” (4) is certainly more important than a page of “bad” (0).

Practical Strategies

- For binary classification, it does not matter.
- In many cases, we can just ignore the orders.
- We can fit K binary logistic regressions by grouping the data according to whether the labels are smaller or larger than L:

  $$\mathbf{Pr}\left(Label > L\right)$$

  from which one can compute the individual class probabilities:

  $$\mathbf{Pr}\left(Label = L\right) = \mathbf{Pr}\left(Label \leq L+1\right) - \mathbf{Pr}\left(Label \leq L\right)$$

  One drawback is that for some data points, the fitted class probabilities may be smaller than 0 after subtraction. But if you have lots of data, this method is often quite effective in practice, for example, in our previous work on ranking webpages. Do read the slides on ranking if you are interested.
- More sophisticated models...

Have a Wonderful Summer!