ISyE 6416: Computational Statistics
Spring 2017

Lecture 4: Gradient Decent and Newton's Method

Prof. Yao Xie

H. Milton Stewart School of Industrial and Systems Engineering
Georgia Institute of Technology

Multiple linear regression

- set-up

$$y_i = \beta_1 x_{i1} + \ldots \beta_p x_{ip} + \beta_0 + \epsilon_i, \quad i = 1, \ldots, n$$

p variables: $\beta = [\beta_0, \beta_1, \cdots, \beta_p]^\mathsf{T}$
n samples

$$\min_{\beta} \sum_{i=1}^{n} (y_i - (\beta_1 x_{i1} + \ldots \beta_p x_{ip} + \beta_0))^2$$

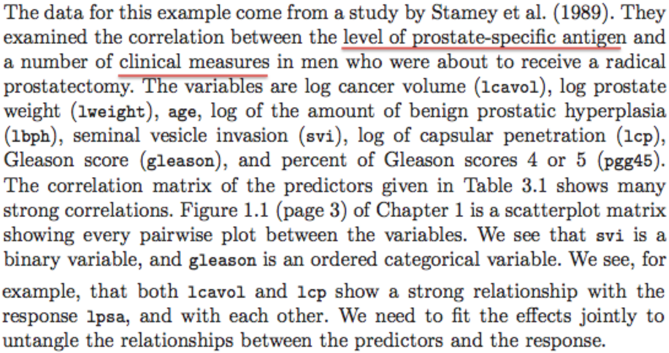
- matrix-vector form

$$y = A\beta + \epsilon, \quad A = \begin{bmatrix} 1 & x_{11} & \cdots & x_{p1} \\ \vdots & \vdots & & \vdots \\ 1 & x_{1n} & \cdots & x_{pn} \end{bmatrix} \in \mathbb{R}^{n \times (p+1)}$$

- parameter estimation

$$\min_{\beta} \|y - A\beta\|_2^2$$

Example: prostate cancer



The data for this example come from a study by Stamey et al. (1989). They examined the correlation between the level of prostate-specific antigen and a number of clinical measures in men who were about to receive a radical prostatectomy. The variables are log cancer volume (`lcavol`), log prostate weight (`lweight`), `age`, log of the amount of benign prostatic hyperplasia (`lbph`), seminal vesicle invasion (`svi`), log of capsular penetration (`lcp`), Gleason score (`gleason`), and percent of Gleason scores 4 or 5 (`pgg45`). The correlation matrix of the predictors given in Table 3.1 shows many strong correlations. Figure 1.1 (page 3) of Chapter 1 is a scatterplot matrix showing every pairwise plot between the variables. We see that `svi` is a binary variable, and `gleason` is an ordered categorical variable. We see, for example, that both `lcavol` and `lcp` show a strong relationship with the response `lpsa`, and with each other. We need to fit the effects jointly to untangle the relationships between the predictors and the response.

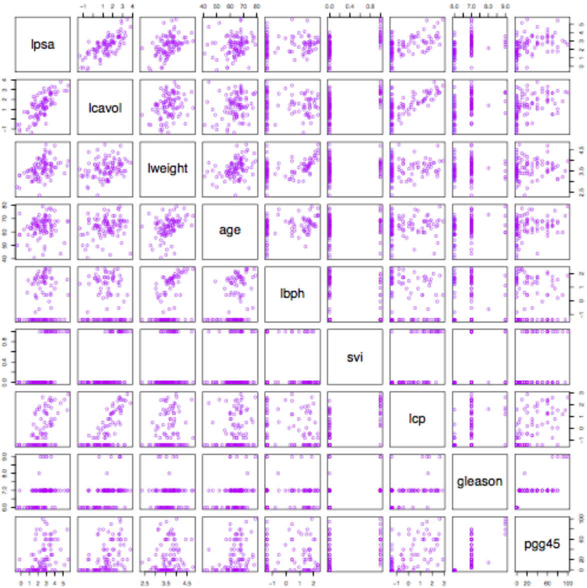
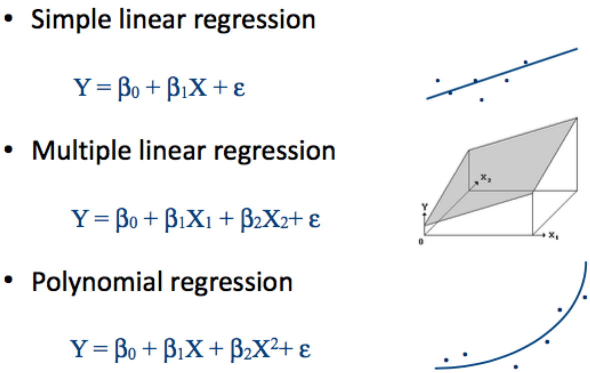


FIGURE 1.1. *Scatterplot matrix of the prostate cancer data. The first row shows the response against each of the predictors in turn. Two of the predictors,* `svi` *and* `gleason`*, are categorical.*

Regression



- Simple linear regression

$$Y = \beta_0 + \beta_1 X + \varepsilon$$

- Multiple linear regression

$$Y = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \varepsilon$$

- Polynomial regression

$$Y = \beta_0 + \beta_1 X + \beta_2 X^2 + \varepsilon$$

Variable selection: for multiple linear regression, select the "most important" variables that are responsible for the output.

Method of least squares

- Linear regression model

$$y_i = \beta_0 + \beta_1 x_i + \epsilon_i, \quad i = 1, 2, \ldots, n$$

- To estimate (β_0, β_1), we find values that minimize the risk (square error)

$$\min_{\beta_0,\beta_1} R(\beta_0, \beta_1) = \min_{\beta_0,\beta_1} \sum_{i=1}^{n} (y_i - \beta_0 - \beta_1 x_i)^2$$

$$\hat{\beta}_1 = S_{xy}/S_{xx}$$

$$\hat{\beta}_0 = \bar{y} - \hat{\beta}_1 \bar{x}$$

$S_{xy} = \sum_{i=1}^{n}(y_i - \bar{y})(x_i - \bar{x})$, $S_{xx} = \sum_{i=1}^{n}(x_i - \bar{x})^2$

Logistic regression

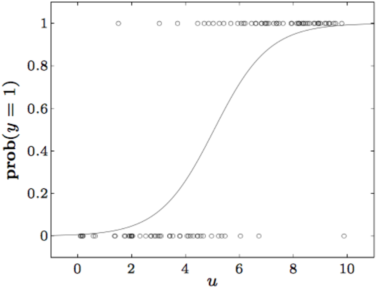
- random variable $y\{0,1\}$ with distribution

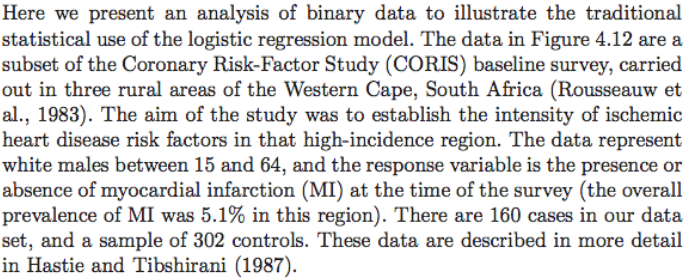
$$h(x;a,b) = \mathbb{P}(y=1) = \frac{\exp(a^\intercal x + b)}{1+\exp(a^\intercal x + b)}$$

  Sigmoid function $s(x) = \frac{1}{1+e^{-x}}$

- maximum likelihood

$$\max_{a,b} \sum_{i=1}^{n} \{y_i \log h(x_i;a,b) + (1-y_i)\log(1-h(x_i;a,b))\}$$





Here we present an analysis of binary data to illustrate the traditional statistical use of the logistic regression model. The data in Figure 4.12 are a subset of the Coronary Risk-Factor Study (CORIS) baseline survey, carried out in three rural areas of the Western Cape, South Africa (Rousseauw et al., 1983). The aim of the study was to establish the intensity of ischemic heart disease risk factors in that high-incidence region. The data represent white males between 15 and 64, and the response variable is the presence or absence of myocardial infarction (MI) at the time of the survey (the overall prevalence of MI was 5.1% in this region). There are 160 cases in our data set, and a sample of 302 controls. These data are described in more detail in Hastie and Tibshirani (1987).

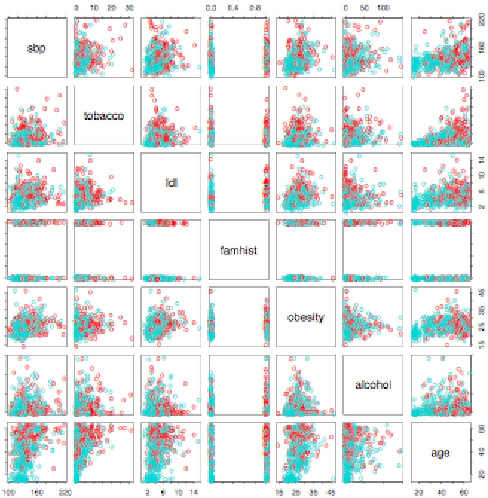
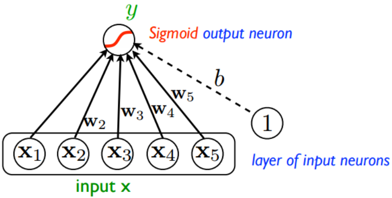
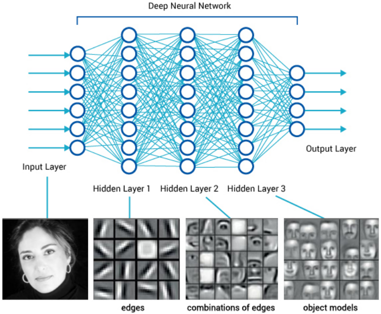


FIGURE 4.12. *A scatterplot matrix of the South African heart disease data. Each plot shows a pair of risk factors, and the cases and controls are color coded (red is a case). The variable* family history of heart disease (`famhist`) *is binary (yes or no).*

Deep learning and neural networks





Solving optimization problem

- solve optimization problem

$$\min_x f(x)$$

- produce sequence of points $x^{(k)}$, $k = 0, 1, 2, \ldots$ with

$$f(x^{(k)}) \to p^*$$

- can be interpreted as iterative methods for solving optimality condition

$$\nabla f(x^*) = 0$$

Gradient decent

$$x^{(k+1)} = x^{(k)} - t^{(k)} \nabla f(x^{(k)})$$

$t^{(k)}$: step-size for the kth iteration
$\nabla f(x)$: gradient vector

- can be viewed as function iteration for function $h(x) = x - t\nabla f(x)$
- for convex optimization it gives the global optimum under fairly general conditions.
- for nonconvex optimization it arrives at local optimum

Example: solving multiple linear regression

$$\min_{\beta} \|y - A\beta\|_2^2, \quad A \in \mathbb{R}^{n\times(p+1)}$$

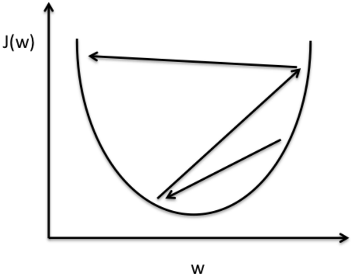
- $f(\beta) = \|y - A\beta\|_2^2$
- Gradient $\nabla f(\beta) = 2A^\intercal(A\beta - y)$
- Hessian $H[f](\beta) = 2A^\intercal A$
- Exact solution $\hat{\beta} = (A^\intercal A)^{-1}A^\intercal y$, issue: complexity $\mathcal{O}(p^3)$
- Gradient descent

$$\beta^{(k+1)} = \beta^{(k)} - 2t^{(k)}A^\intercal(A\beta^{(k)} - y)$$

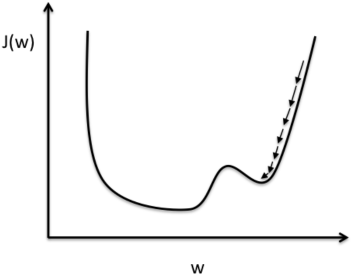
complexity $\mathcal{O}(n^2p)$

Exercise: gradient descent for logistic regression

Choice of step-size



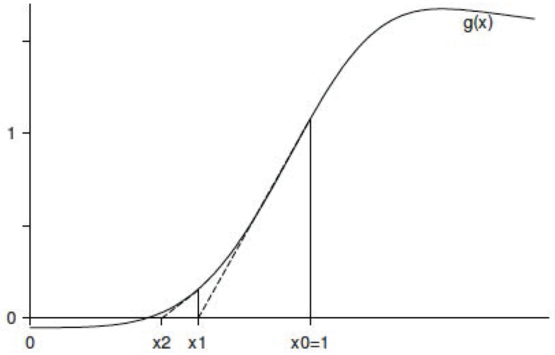
Large learning rate: Overshooting.



Small learning rate: Many iterations until convergence and trapping in local minima.

Newton's method for finding function root

- solve $g(x) = 0$
- iterative method: $x_n = x_{n-1} - \frac{g(x_{n-1})}{g'(x_{n-1})}$
- functional iteration $x_n = f(x_{n-1})$ with $f(x) = x - \frac{g(x)}{g'(x)}$



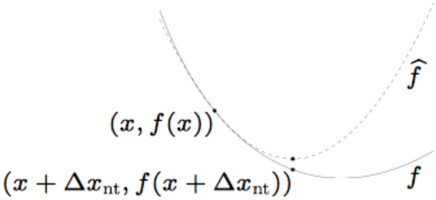
Newton's method for optimization

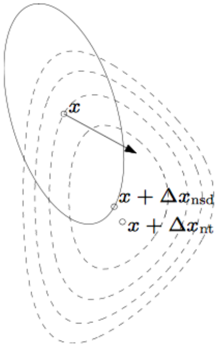
$$x^{(k+1)} = x^{(k)} - t^{(k)}[H\{f(x^{(k)})\}]^{-1}\nabla f(x^{(k)})$$

$t^{(k)}$: step-size for the kth iteration

- interpretation $x + \Delta x$ minimizes the second order approximation of the function

$$f(x+v) \approx f(x) + \nabla f(x)^\intercal v + \frac{1}{2} v^\intercal H\{f(x)\} v$$





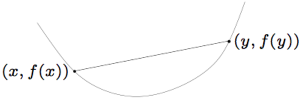
dashed lines are contour lines of f; ellipse is $\{x + v \mid v^T \nabla^2 f(x) v = 1\}$

arrow shows $-\nabla f(x)$

Convex function

A function f is convex if

$$f(\theta x + (1-\theta)y) \leq \theta f(x) + (1-\theta) f(y)$$



convex functions

- affine: $ax + b$
- exponential e^{ax}
- powers $|x|^{\alpha}$ for $p \geq 1$

concave:

- affine: $ax + b$
- log: $\log x$
- powers x^{α} for $0 \leq \alpha \leq 1$

Strong convexity and implications

- f is strongly convex on domain S if there exists an $m > 0$ such that
$$H\{f(x)\} \geq mI, \quad \text{for all } x \in S.$$
- **implications**
  - for $x, y \in S$
$$f(y) \geq f(x) + \nabla f(x)^{\mathsf{T}}(y - x) + \frac{m}{2}\|x - y\|_2^2$$
  - for $x \in S$
$$f(x) - p^* \leq \frac{1}{2m}\|\nabla f(x)\|_2^2$$

useful as a stoping criterion

Convergence results

- **Gradient descent**: for strongly convex f with constant m number of iterations until $f(x) - p^* \leq \epsilon$ is bounded above by

  $$f(x^{(k)}) - p^* \leq c^k(f(x^{(0)}) - p^*)$$

  $c \in (0, 1)$ is a constant depend on $x^{(0)}$, step-size, m etc. Very simple, but converges very slow

- **Newton's method**: for strongly convex f with constant m number of iterations until $f(x) - p^* \leq \epsilon$ is bounded above by

  $$\frac{f(x^{(0)}) - p^*}{\gamma} + \log\log_2(\epsilon_0/\epsilon)$$

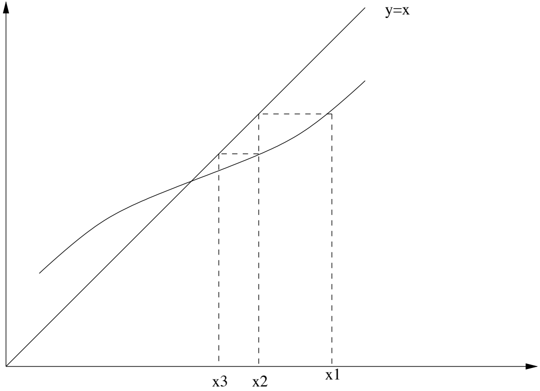
  constants γ, ϵ_0 depends on ϵ_0, m, L (Lipschitz constant for the Hessian).

Quadratic convergence of Newton's method

- let $e_n = |x_n - x_\infty|$
- quadratic convergence: $\lim_{n\to\infty} \frac{e_n}{e_{n-1}^2} = \frac{1}{2} f''(x_\infty)$
- linear convergence definition: if $\lim_{n\to\infty} \frac{e_n}{e_{n-1}} = f'(x_\infty)$, f: iteration function, $0 < |f'(x_\infty)| < 1$

Functional iteration

- find a root of the equation $g(x) = 0$
- introduce $f(x) = g(x) + x$, $g(x) = 0 \Rightarrow f(x) = x$
- in many examples, iterates $x_n = f(x_{n-1})$ convergens to $x^* = f(x^*)$, x^* called fixed point of $f(x)$



Convergence

Theorem

Suppose the function $f(x)$ defined on a closed interval I satisfies the conditions:

1. $f(x) \in I$ *whenever* $x \in I$
2. $|f(y) - f(x)| \leq \lambda |y - x|$ *for any two points* x *and* y *in* I*.*

Then, provided the Lipschiz constant λ is in $[0, 1)$, $f(x)$ has a unique fixed point $x^* \in I$, and $x_n = f(x_{n-1})$ converges to x^* regardless of starting point $x_0 \in I$. Furthermore, we have

$$|x_n - x_\infty| \leq \frac{\lambda^n}{1 - \lambda} |x_1 - x_0|.$$

First consider

$$|x_{k+1} - x_k| = |f(x_k) - f(x_{k-1})| \leq \lambda |x_k - x_{k-1}| \cdots \leq \lambda^k |x_1 - x_0|$$

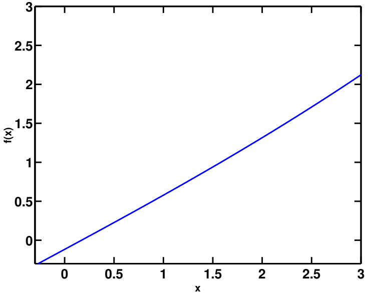
then for some $m > n$,

$$|x_m - x_n| \leq \sum_{k=n}^{m-1} |x_{k+1} - x_k| \leq \sum_{k=n}^{m-1} \lambda^k |x_1 - x_0| \leq \frac{\lambda^n}{1-\lambda} |x_1 - x_0|$$

So $\{x_n\}$ forms a Cauchy sequence. Since $I \in \mathbb{R}$ is closed and bounded, it is compact.

Example

- $g(x) = -\sin(\frac{x+e^{-1}}{\pi})$, find x such that $g(x) = 0$
- $f(x) = g(x) + x$, $f'(x) = -\frac{1}{\pi}\cos(\frac{x+e^{-1}}{\pi}) + 1$
- $|f'(x)| < 1$ for $[-e^{-1}, \frac{\pi^2}{2} - e^{-1}] = [-0.3679, 4.5669]$ (so we can apply function iteration to $g(x)$
- let $I = [-0.3, 3]$, then $f(x) \in I$ whenever $x \in I$, and $\lambda < 1$ for $f(x)$ on this range



Computation for maximum likelihood

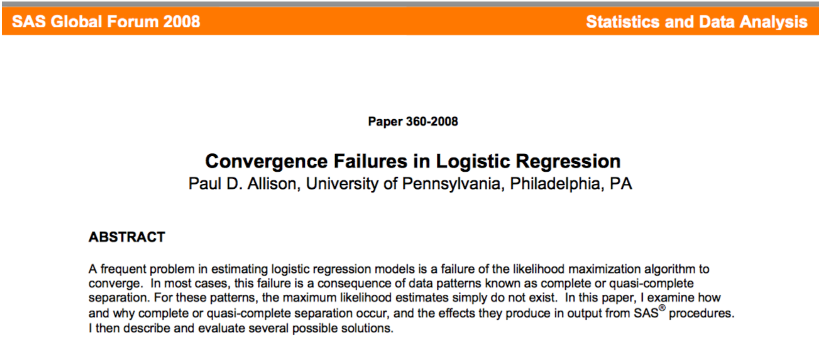
- Given PDF, we form the likelihood or log-likelihood function, estimate parameter by maximum likelihood

$$\max_{\theta} \ell(\theta|x)$$

- Newton method $f = -\ell$

$$\theta^{(k+1)} = \theta^{(k)} - t^{(k)}[H\{-\ell(\theta^{(k)})\}]^{-1}\nabla[-\ell(\theta^{(k)})]$$

- Drawback: most of time, $H\{\ell(\theta^{(k)})\}$ is complex, and may not be positive definite (no guarantee to be invertible)



SAS Global Forum 2008 Statistics and Data Analysis

Paper 360-2008

Convergence Failures in Logistic Regression

Paul D. Allison, University of Pennsylvania, Philadelphia, PA

ABSTRACT

A frequent problem in estimating logistic regression models is a failure of the likelihood maximization algorithm to converge. In most cases, this failure is a consequence of data patterns known as complete or quasi-complete separation. For these patterns, the maximum likelihood estimates simply do not exist. In this paper, I examine how and why complete or quasi-complete separation occur, and the effects they produce in output from SAS® procedures. I then describe and evaluate several possible solutions.

Fish scoring

- Remove observations, replace Hessian by the expected Hessian

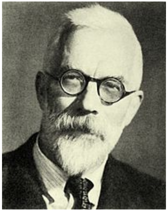
$$J(\theta) = \mathbb{E}\{H[-\log \ell(\theta|x)]\} = -\mathbb{E}\{\frac{\partial^2 \log \ell(\theta)}{\partial\theta\partial\theta^\mathsf{T}}\}$$

Fisher information (FI)

- Under mild regularity conditions

$$J(\theta) = \mathbb{E}\{H[-\log \ell(\theta|x)]\} = \mathbb{E}\{(\frac{\partial}{\partial\theta}\log \ell(\theta|x))^2\}$$

(Recall CRB)



Advantage of Fisher Scoring

- $J(\theta)$ usually has simpler close form
- $J(\theta)$ is nonnegative definite (invertible)
- if $\hat{\theta}$ is the maximum likelihood estimator (MLE), then $J^{-1}(\hat{\theta})$ is the variance/covariance of the MLE $\hat{\theta}$
- Example: FI for Geometric distribution

$$J(p) = \frac{n}{p^2(1-p)}$$

increasing as p moves away from 2/3 towards 0 or 1.

Exponential family

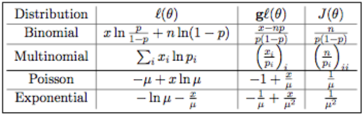
- Exponential family: provides a general framework to parameterize distributions

$$f(x|\theta) = g(x)e^{\beta(\theta)+h^\intercal(x)v(\theta)}$$

- Example: $\exp(\lambda)$, $\mathcal{N}(\mu, \sigma^2)$
- **Sufficient statistic**: $h(x)$
- Fisher information for general exponential family

$$\Sigma(\theta) = \mathsf{Var}[h(x)]$$

$$J(\theta) = \nabla v^\intercal(\theta)\Sigma(\theta)\nabla v(\theta)$$

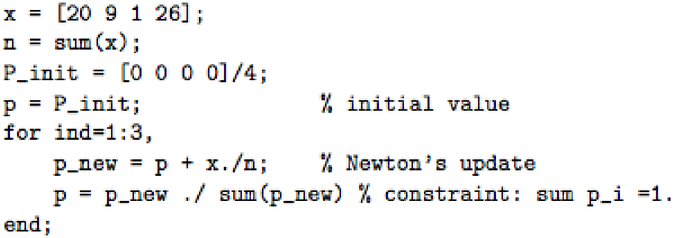


Distribution	$\ell(\theta)$	$\mathbf{g}\ell(\theta)$	$J(\theta)$
Binomial	$x \ln \frac{p}{1-p} + n \ln(1-p)$	$\frac{x-np}{p(1-p)}$	$\frac{n}{p(1-p)}$
Multinomial	$\sum_i x_i \ln p_i$	$\left(\frac{x_i}{p_i}\right)_i$	$\left(\frac{n}{p_i}\right)_{ii}$
Poisson	$-\mu + x \ln \mu$	$-1 + \frac{x}{\mu}$	$\frac{1}{\mu}$
Exponential	$-\ln \mu - \frac{x}{\mu}$	$-\frac{1}{\mu} + \frac{x}{\mu^2}$	$\frac{1}{\mu^2}$

Example: Newton's method using Fisher scoring

- Consider multinomial distribution Multi$(n; p_1, \dots, p_4)$
- Assume $n = 56$, $x_1 = 20$, $x_2 = 9$, $x_3 = 1$ and $x_4 = 26$.
- Find MLE for p_1, p_2, p_3 and p_4.
- FI $J(\theta)$ is diagonal matrix with diagonal entries (n/p_i)
- needed for Newton step: $J^{-1}(\theta)\nabla[-\ell(\theta)]$ is diagonal matrix with entries $-\frac{x_i/p_i}{n/p_i} = -\frac{x_i}{n}$
- Newton's updating rule

$$p_i^{(k+1)} = p_i^{(k)} + \mu \frac{x_i}{n}$$



```
x = [20 9 1 26];
n = sum(x);
P_init = [0 0 0 0]/4;
p = P_init;              % initial value
for ind=1:3,
    p_new = p + x./n;    % Newton's update
    p = p_new ./ sum(p_new) % constraint: sum p_i =1.
end;
```

Some limitations

- ignores the condition $\sum_{i=1}^{4} p_i = 1$ (constrained optimization; need to include Lagrangian multiplier)
- sensitive to the initial value

Stochastic gradient descent

- Stochastic gradient descent method uses noisy unbiased subgradients

$$x^{(k+1)} = x^{(k)} - \alpha_k \tilde{g}^{(k)}$$

- $\tilde{g}^{(k)}$ is any noisy unbiased gradient of a function at $x^{(k)}$

$$\mathbb{E}[\tilde{g}^{(k)}] = g^{(k)}$$

- Convergence

$$\min_{i=1,\dots,k} (\mathbb{E} f(x^{(i)}) - p^*) \leq \frac{R^2 + G^2 \|\alpha\|_2^2}{2 \sum_{i=1}^{k} \alpha_i}$$

$\mathbb{E}\|g^{(k)}\|_2^2 \leq G^2$, $\mathbb{E}\|x^{(1)} - x^*\| \leq R^2$

Email SPAM classification via SVMs

- Support Vector Machine (SVM) is the name in machine learning for a linear classifier
- Suppose we wish to train the classifier to classify emails as spam/nonspam.
- Each email is represented using a vector that gives the frequencies of various words in it ("bag of words" model).

- $x_1, \dots, x_n$ emails, label $y_i \in \{-1, 1\}$
- If spam were perfectly identifiable by a linear classifier, there would be a function such that $w^\intercal x_i \geq 1$ if X_i is spam, and $w^\intercal x_i \leq -1$ otherwise. Hence,

  $$1 - y_i w^\intercal x_i \leq 0$$

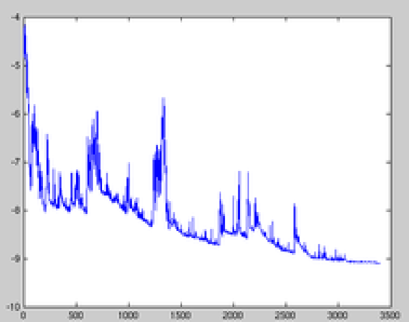
  Solve

  $$\min \sum_i \mathsf{Loss}(1 - y_i w^\intercal x_i) + \lambda \|w\|_2^2 \quad (*)$$

  Hinge loss: $\mathsf{Loss}(x) = \max\{0, x\}$
- Loss(.) a function that penalizes unsatisfied constraints according to the amount by which they are unsatisfied

- Note that (*) is a sum of many similar terms
- If we randomly pick a single term (data for one email) and compute just its gradient, the expectation of this gradient is just the true gradient

$$w^{(1)} \xrightarrow{\tilde{g}(x_1)} w^{(2)} \xrightarrow{\tilde{g}(x_2)} w^{(3)} \ldots$$


-4
-5
-6
-7
-8
-9
-10
0
500
1000
1500
2000
2500
3000
3500