Statistical Computing

Hung Chen

hchen@math.ntu.edu.tw

Department of Mathematics

National Taiwan University

18th February 2004

Meet at NW 405 On Wednesday from 9:10 to 12.

Course Overview

- **Monte Carlo methods for statistical inference**
  1. **Pseudo-random deviate**
  2. **Non-uniform variate generation**
  3. **Variance reduction methods**
  4. **Jackknife and Bootstrap**
  5. **Gibbs sampling and MCMC**
- **Data partitioning and resampling (bootstrap)**
  1. **Simulation Methodology**
  2. **Sampling and Permutations (Bootstrap and permutation methods)**
- **Numerical methods in statistics**
  1. **Numerical linear algebra and linear regressions**

2. Application to regression and nonparametric regression
3. Integration and approximations
4. Optimization and root finding
5. Multivariate analysis such as principal component analysis

- Graphical methods in computational statistics
- Exploring data density and structure
- Statistical models and data fitting
- Computing Environment: Statistical software R ("GNU's S")
  - http://www.R-project.org/
  - Input/Output

- Structured Programming
- Interface with other systems

- **Prerequisite:**
  - Knowledge on regression analysis or multivariate analysis
  - Mathematical statistics/Probability theory; Statistics with formula derivation
  - Knowledge about statistical software and experience on programming languages such as Fortran, C and Pascal.

- **Text Books:**
  The course materials will be drawn from following recommended resources (some are available via Internet) and others that will be made available

through the handout.

- **Gentle, J.E. (2002)** *Elements of Computational Statistics*. **Springer.**
- **Hastie, T., Tibshirani, T. , Friedman, J.H. (2001)** *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*. **Springer-Verlag.**
- **Lange, K. (1999)** *Numerical analysis for statisticians*. **Springer-Verlag, New York**
- **Ripley, B.D. and Venables, V.W. and BD (2002) Modern applied statistics with S, 4th edition. Springer-Verlag, New York.**
  Refer to http://geostasto.eco.uniroma1.it/utenti/liseo/dispense_R.pdf for a note of this book.
- **Robert, C.P. and Casella, G. (1999).** *Monte Carlo*

Statistical Methods. **Springer Verlag.**

- **Stewart, G.W. (1996).** *Afternotes on Numerical Analysis*. **SIAM, Philadelphia.**
- *An Introduction to R* **by William N. Venables, David M. Smith (http://www.ats.ucla.edu/stat/books/ #DownloadableBooks)**

- **Grading: HW** 50%**, Project** 50%

Outline

- Statistical Learning
  - Handwritten Digit Recognition
  - Prostate Cancer
  - DNA Expression Microarrays
  - Language
- IRIS Data
  - Linear discriminant analysis
  - R-programming
  - Logistic regression
  - Risk minimization
  - Discriminant Analysis
- Optimization in Inference

- Estimation by Maximum Likelihood
- Optimization in R
- EM Methods
  - Algorithm
  - Genetics Example(Dempster, Laird, and Rubin; 1977)
- Normal Mixtures
- Issues

- EM Methods: Motivating example
  - ABO blood groups
  - EM algorithm
  - Nonlinear Regression Models
  - Robust Statistics

- **Probability statements in Statistical Inference**
- **Computing Environment: Statistical software R ("GNU's S")**
  - **http://www.R-project.org/**
  - **Input/Output**
  - **Structured Programming**
  - **Interface with other systems**

Introduction: Statistical Computing in Practice

Computationally intensive methods have become widely used both for statistical inference and for exploratory analysis of data.
The methods of computational statistics involve

- **resampling, partitioning, and multiple transformations of a data set**
- **make use of randomly generated artificial data**
- **function approximation**

Implementation of these methods often requires advanced techniques in numerical analysis, so there is a close connection between computational statistics and statistical computing.

In this course, we first address some areas of application of computationally intensive methods, such as

- **density estimation**
- **identification of structure in data**
- **model building**
- **optimization**

Handwritten Digit Recognition

In order to devise an automatic sorting procedure for envelopes, we consider the problem of recognizing the handwritten ZIP codes on envelopes from U.S. postal mail.
This is a so-called pattern recognition problem in literature.

- **Each image is a segment from a five digit ZIP code, isolating a single digit.**
- **The images are 16×16 eight-bit grayscale maps, with each pixel ranging in intensity from 0 to 255.**
- **The images have been normalized to have approximately the same size and orientation.**
- **The task is to predict, from the 16×16 matrix of pixel**

intensities, the identity of each image $(0, 1, \ldots, 9)$ **quickly and accurately.**
The dimensionality of x **is** 256**.**

Abstraction:

- **Consider space** X **as matrices with entries in the interval** $[0, 1]$**-each entry representing a pixel in a certain grey scale of a photo of the handwritten letter or some features extracted from the letters.**
- **Let** Y **to be**

$$Y = \left\{ y \in R^{10} \mid y = \sum_{i=1}^{10} p_i e_i \text{ s.t. } \sum_{i=1}^{10} p_i = 1 \right\}.$$

Here e_i **is the** i**th coordinate vector in** R^{10} **(each coordinate corresponding to a letter).**

- **If we only consider the set of points y with $0 \leq p_i \leq 1$, for $i = 1, \ldots, 10$, one can interpret in terms of a probability measure on the set $\{0, 1, 2, \ldots, 10\}$.**
- **The problem is to learn the ideal function $f : X \to Y$ which associates, to a given handwritten digit x, the point $\{Prob(x = 0), Prob(x = 1), \ldots, Prob(x = 9)\}$.**
- *Learning* f **means to find a sufficiently good approximation of f within a given prescribed class.**
  - **For a two-class problem, think of logistic regression in survival analysis.**
  - **Fisher discriminant analysis and SVM**

Further mathematical abstraction:

- **Consider a measure ρ on $X \times Y$ where Y is the label set and X is the set of handwritten letters.**

- **The pairs (x_i, y_i) are randomly drawn from $X \times Y$ according to the measure ρ.**
- **y_i is a sample for a given x_i.**
- **The function to be learned is the regression function of f_ρ.**
- **$f_\rho(x)$ is the average of the y values of $\{x\} \times Y$.**
- **Difficulty: The probability measure ρ governing the sampling which is not known in advance.**

How do we learn f?

Prostate Cancer

To identify the risk factors for prostate cancer, Stamey et al. (1989), they examined the correlation between the level of prostate specific antigen (PSA) and a number of clinical and demographic measures, in 97 men who were about to receive a radical prostatetectomy.

- **The goal is to predict the log of PSA (lpsa) from a number of measurements including log-cancer-volume (lcavol), log prostate weight lweight, age, log of benign prostatic hyperplasia amount lbph, seminal vesicle invasion svi, log of capuslar penetration lcp, Gleason score gleason, and percent of Gleason scores 4 or 5 pgg45.**
- **This is a supervised learning problem, known as a re-**

gression problem, **because the outcome measurement is quantitative.**

Let Y denote lpsa and X be those explanatory variables.

- **We have data in the form of $(x_1, y_1), \ldots, (x_n, y_n)$.**
- **Use the squared error loss as the criterion of choosing the best prediction function, i.e.,**

$$\min_{\theta} E[Y - \theta(X)]^2.$$

- **What is the $\theta(\cdot)$ which minimizes the above least-squares error in population version.**
  - **Find c to minimize $E(Y - c)^2$.**
  - **For every $x \in X$, let $E(Y \mid X = x)$ be the conditional expectation.**

- **Regression function:** $\theta(x) = E(Y \mid X = x)$
- **Write** Y **as the sum of** $\theta(X)$ **and** $Y - \theta(X)$**.**
  - **Conditional expectation:** $E(Y - \theta(X) \mid X) = 0$
  - **Conditional variance:** $Var(Y \mid X) = E[(Y - \theta(X))^2 \mid X)]$

- **ANOVA decomposition:**

$$Var(Y) = E[Var(Y \mid X)] + Var[E(Y \mid X)]$$

- **If we use** $E(Y)$ **to minimize prediction error** $E(Y - c)^2$**, its prediction error is** $Var(Y)$**.**
- **If** $E(Y \mid X)$ **is not a constant, the prediction error of** $\theta(x)$ **is smaller.**

- **Nonparametric regression: No functional form is assumed for** $\theta(\cdot)$

- **Suppose that $\theta(x)$ is of the form $\sum_{i=1}^{\infty} w_i \phi_i(x)$.**
  - **What is $\{\phi_i(x); i = 1, 2, \ldots\}$ and how to determine $\{w_i, i = 1, 2, \ldots\}$ with finite number of data?**
  - **Estimate $\theta(x)$ by k-nearest-neighbor method such as**

$$\hat{Y}(x) = \frac{1}{k} \sum_{x_i \in N_k(x)} y_i,$$

where $N_k(x)$ is the neighborhood of x defined by the k closest points x_i in the training sample. How do we choose k and $N_k(x)$?

- **Empirical error:**
  - **How do we measure the theoretical error $E[Y - f(X)]^2$?**
    - **Consider n examples, $\{(x_1, y_1), \ldots, (x_n, y_n)\}$, in-**

dependently drawn according to ρ.

* **Define the** *empirical error* **of f to be**

$$E_n(f) = \frac{1}{n}\sum_{i=1}^{n}(y_i - f(x_i))^2.$$

What is the empirical cdf?

* **Is $E_n(f)$ close to $E[Y - f(X)]^2$?**
* **Think of the following problem:**

Is sample variance a consistent estimate of populatio

Or the following holds in some sense

$$\frac{1}{n}\sum_{i}(y_i - \bar{y})^2 \to E(Y - E(Y))^2.$$

* **Can we claim that the minimizer $\hat{f}$ of $\min_{f\in\mathcal{F}}(y_i-$**

$f(x_i))^2$ **is close to the minimizer of** $\min_{f \in \mathcal{F}} E(Y - f(X))^2$**?**

- **Consider the problem of email spam.**
  - **Consider the data which consists of information from** 4601 **email messages.**
  - **The objective was to design an automatic spam detector that could filter out spam before clogging the users' mailboxes.**
  - **For all** 4601 **email messages, the true outcome (email type)** $email$ **or** $spam$ **is available, along with the relative frequencies of** 57 **of the most commonly occurring words and punctuation marks in the email message.**
  - **This is a** $classification$ **problem.**

DNA Expression Microarrays

DNA is the basic material that make up human chromosomes.

- **The regulation of gene expression in a cell begins at the level of transcription of DNA into mRNA.**
- **Although subsequent processes such as differential degradation of mRNA in the cytoplasm and differential translation also regulate the expression of genes, it is of great interest to estimate the relative quantities of mRNA species in populations of cells.**
- **The circumstances under which a particular gene is up- or down-regulated provide important clues about gene function.**
- **The simultaneous expression profiles of many genes**

can provide additional insights into physiological processes or disease etiology that is mediated by the coordinated action of sets of genes.

Spotted cDNA microarrays (Brown and Botstein 1999) are emerging as a powerful and cost-effective tool for large scale analysis of gene expression.

- **In the first step of the technique, samples of DNA clones with known sequence content are spotted and immobilized onto a glass slide or other substrate, the microarray.**
- **Next, pools of mRNA from the cell populations under study are purified, reverse-transcribed into cDNA, and labeled with one of two fluorescent dyes, which we will refer to as "red" and "green."**

- **Two pools of differentially labeled cDNA are combined and applied to a microarray.**
- **Labeled cDNA in the pool hybridizes to complementary sequences on the array and any unhybridized cDNA is washed off.**

The result is a few thousand numbers, typically ranging from say -6 to 6, measuring the expression level for each gene in the target relative to the reference sample. As an example, we have a data set with 64 samples (column) and 6830 genes (rows). The challenge is to understand how the genes and samples are organized. Typical questions are as follows:

- **Which samples are most similar to each other, in terms of their expression profiles across genes?**

- **Think of the samples as points in 6830-dimensional space, which we want to *cluster* together in some way.**

- **Which genes are most similar to each other, in terms of their expression profile across samples?**

- **Do certain genes show very high (or low) expression for certain cancer samples?**
  - **Feature selection problem.**

Statistical Language

For the examples we just described, they have several components in common.

- **For each there is a set of variables that might be denoted as** *inputs*, **which are measured or present. These have some influence on one or more outputs.**
- **For each example, the goal is to use the inputs to predict the values of the outputs. In machine learning language, this exercise is called** *supervised learning*.
- **In statistical literature the inputs are often called the** *predictors* **or more classically the** *independent variables*.
  The outputs are called *responses*, **or classically the**

dependent variables.

- **The output can be a** *quantitative* **measurement, where some measurements are bigger than others, and measurements close in value are close in nature.**
  - **EX 1. Consider the famous Iris discrimination examples (Fisher, 1936). In this data set, there are** 150 **cases with** 50 **cases per class. The output is** *qualitative* **(species of Iris) and assumes values in a finite set** $\mathcal{G} = \{$**Virginica, Setosa and Versicolor**$\}$**. There are four predictors: sepal length, sepal width, petal length, and petal width.**
  - **EX 2. In the handwritten digit example, the output is one of** 10 **different digit class.**
  - **In Ex1 and 2, there is no explicit ordering in the**

classes, and in fact often descriptive labels rather than numbers are used to denote the classes. Qualitative variables are often referred to as *categorical* **or** *discrete* **variables as well as** *factors*.

- **Ex 3. For given specific atmospheric measurements today and yesterday, we want to predict the ozone level tomorrow.**
  Given the grayscale values for the pixels of the digitized image of the handwritten digit, we want to predict its class labels.
- **For all three examples, we think of using the inputs to predict the output. The distinction in output type has led to a naming convention for the prediction tasks:** *regression* **when we predict quantitative outputs, and** *classification* **when we**

predict qualitative outputs.

- **For** *regression* **and** *classification*, **both can be viewed as a task in function approximation.**
- **For qualitative variables, they are typically represented numerically by codes.**

- **A third variable type is** *ordered categorical*, **such as** *small*, *medium* **and** *large*, **where there is an ordering between the values, but no metric notion is appropriate (the difference between medium and small need not be the same as that between large and medium).**

IRIS Data

First applied in 1935 by M. Barnard at the suggestion of R.A. Fisher (1936).

- **Fisher linear discriminant analysis (FLDA): It consists of**
  - **Find linear combination $\mathrm{x}^T\mathrm{a}$ of $\mathrm{x} = (x_1, \ldots, x_p)$ to maximize the the ratio of "between group" and "within group variances.**
  - **$\mathrm{x}^T\mathrm{a}$ is called discriminant variables.**
  - **Predicting the class of an observation x by the class whose mean vector is closest to x of the discriminant variables.**
  - **Represent Class k by (μ_k, Σ).**
  - **Define $B_0 = \sum_{k=1}^{3}(\mu_k - \bar{\mu})(\mu_k - \bar{\mu})^T$.**

- **Identify eigenvalues and eigenvectors of $\Sigma^{-1}B_0$.**

$$\max_{\mathbf{a}} \frac{\mathbf{a}^T B_0 \mathbf{a}}{\mathbf{a}^T \Sigma \mathbf{a}}.$$

- **The problem of learning is that of choosing from the given set of functions $\mathrm{x}^T\mathrm{a}$, the one which predicts the supervisor's response in the best possible way.**
  - **How do we quantify it?**
  - **How will the variability of $\hat{\mathrm{a}}$ affect the prediction?**

- **R-programming**
  - **data(iris)**
  - **attach(iris)**
  - **help.search("discriminant analysis")**

- **Linear Discriminant Analysis: lda**

- **Logistic regression**
  - **Model**

$$\log \frac{P(y=1 \mid x)}{1-P(y=1 \mid x)} = \alpha + \mathbf{x}^T \beta$$

  - **The coefficients α and β need to be estimated iteratively (writing down the likelihood and finding the MLE).**
    The "scoring method" or "iterative weighted least squares."
  - **Convert a classification problem to an estimation problem.**

- **Problem of risk minimization**

– **The** $loss$ **or discrepancy:** $L(y, f(\mathbf{x}|\alpha))$
Here y **is the supervisor's response to given input** x **and** $f(\mathrm{x}|\alpha)$ **is the response provided by the learning machine.**

– **The risk functional: the expected value of the loss or**

$$R(\alpha) = \int L(y, f(\mathbf{x}|\alpha)) d\rho(\mathbf{x}, y).$$

– **Goal: Find the function which minimizes the risk functional** $R(\alpha)$ **(over the class of functions** $f(\mathrm{x}|\alpha)$**,** $\alpha \in \mathcal{A}$**, in the situation where the joint probability distribution is unknown and the only available information is contained in the training set.**

– **Pattern recognition**

∗ **The supervisor's output** y **take on only two val-**

ues $\{0, 1\}$**.**

* $f(x|\alpha)$**,** $\alpha \in \mathcal{A}$**, are a set of** $indicator$ **functions the (functions which take on only two values zero and one).**
* **The loss-function:**

$$L(y, f(\mathbf{x}|\alpha)) = \begin{cases} 0 \text{ if } y = f(\mathbf{x}|\alpha), \\ 1 \text{ if } y \neq f(\mathbf{x}|\alpha), \end{cases}$$

* **For this loss function, the risk functional provides the probability of classification error (i.e., when the answers given by supervisor and the answers given by indicator function differ).**
* **The problem, therefore, is to find the function which minimizes the probability of classification errors when probability measure is unknown, but the data are given.**

- **Formulation of Discriminant Analysis**
  - **Objective: Distinguish two classes based on the observed covariates (and training data).**
  - **Data:** $\{(\mathbf{x}_1, y_1), \ldots, (\mathbf{x}_\ell, y_\ell)\}$**,** $y_i = 0$ **or** 1**.**
  - **Goal: Make decision on** Y **based on** $\mathbf{X}$**.**
    $(\mathbf{x}, y)$ **has an unknown probability distribution** $\rho(\mathbf{x}, y)$**.**
  - **The Bayes Solution: (minimum error rule)**

    $$P(Y = y \mid \mathbf{X}) = \frac{P(\mathbf{X} \mid Y = y)\pi(Y = y)}{P(\mathbf{X} \mid Y = 0)\pi(Y = 0) + P(\mathbf{X} \mid Y = 1)\pi(Y$$

    It assumes that the cost of different type of misclassification are the same. (Costs may not be the same)
  - **Practical problems: Prior? Distributions?**

Role of Optimization in Inference

Many important classes of estimators are defined as the point at which some function that involves the parameter and the random variable achieves an optimum.

- **Estimation by Maximum Likelihood:**
  - **Concept intuitively**
  - **Nice mathematical properties**
  - **Give a sample $y_1, \ldots, y_n$ from a distribution with pdf or pmf $p(y \mid \theta_*)$, MLE of θ is the value that maximizes the joint density or joint probability with variable θ at the observed sample value: $\prod_i p(y_i \mid \theta)$.**

- **Likelihood function**

$$L_n(\theta; \mathbf{y}) = \prod_{i=1}^{n} p(y_i \mid \theta).$$

  The value of θ for which L_n achieves its maximum value is the MLE of θ_*.

- **Critique: How to determine the form of the function $p(y \mid \theta)$?**

- **The data-that is, the realizations of the variables in pdf or pmf- are considered as fixed, and the parameters are considered as variables of the optimization problem,**

$$\max_{\theta} L_n(\theta; \mathbf{y}).$$

- **Questions:**

- Does the solution exist?
- Existence of local optima of the objective function.
- Constraints on possible parameter values.

• MLE may not be a good estimation scheme.

• Penalized maximum likelihood estimation

• Optimization in R

- nlm is the general function for "nonlinear minimization."
  * It can make use of a gradient and/or Hessian, and it can give an estimate of the Hessian at the minimum.
  * This function carries out a minimization of the function f using a Newton-type algorithm.

  - * **It needs starting parameter values for the minimization.**
  - * **See demo(nlm) for examples.**
- **Minimize function of a single variable**
  - * **optimize: It searches the interval from 'lower' to 'upper' for a minimum or maximum of the function f with respect to its first argument.**
  - * **uniroot: It searches the interval from 'lower' to 'upper' for a root of the function f with respect to its first argument.**
- **D and deriv do symbolic differentiation.**

- **Example 1: Consider $x \sim mult(n, p)$ where $p = ((2 + \theta)/4, (1 - \theta)/2, \theta/4)$.**
  - **Suppose we observe the data $x = (100, 94, 6)$.**

– **The negative log likelihood is**

$$\ln n! - (\sum_i \ln n_i!) + x_1 \ln((2+\theta)/4) + x_2 \ln((1-\theta)/2)$$
$$+x_3 \ln(\theta/4).$$

– **In R, it can be written as**

$$nll < -function(theta, x) - sum(x * log(c((2 + theta)/4$$
$$(1 - theta)/2, theta/4)))$$

– **Using the nlm function, we get**

$$nlm(nll, 0.1, \textbf{typsize} = 0.01, \textbf{hessian = TRUE},$$
$$x = c(100, 94, 6))$$

– **We have** $\hat{\theta} = 0.104$ **with an estimated SE** $\approx 1/\sqrt{689.7} \approx 0.04$**.**

EM Methods

- **Goals: Provide an iterative scheme for obtaining maximum likelihood estimates, replacing a hard problem by a sequence of simpler problems.**
- **Context: Though most apparent in the context of missing data, it is quite useful in other problems as well.**
  The key is to recognize a situation where, if you had more data, the optimization would be simplified.
- **Approach: By augmenting the observed data with some additional random variables, one can often convert a difficult maximum likelihood problem into one which can be solved simply, though requiring iteration.**

– **Treat the observed data $\mathbf{Y}$ as a function $\mathbf{Y} = \mathbf{Y}(\mathbf{X})$ of a larger set of unobserved** *complete* **data $\mathbf{X}$, in effect treating the density**

$$g(y;\theta) = \int_{\mathcal{X}(y)} f(x;\theta)dx.$$

The trick is to find the right f so that the resulting maximization is simple, since you will need to iterate the calculation.

- **Computational Procedure: The two steps of the calculation that give the algorithm its name are**

  1. **Estimate the sufficient statistics of the complete data $\mathbf{X}$ given the observed data $\mathbf{Y}$ and current parameter values,**
  2. **Maximize the X-likelihood associated with these**

estimated statistics.

- **Genetics Example (Dempster, Laird, and Rubin; 1977): Observe counts**

$$\begin{aligned}\mathbf{y} &= (y_1, y_2, y_3, y_4) = (125, 18, 20, 34) \\ &\sim Mult(1/2 + \theta/4, (1-\theta)/4, (1-\theta)/4, \theta/4)\end{aligned}$$

where $0 \leq \theta \leq 1$**.**

  - **Estimate** θ **by solving the score equation**

$$\frac{y_1}{2+\theta} - \frac{y_2 + y_3}{1-\theta} + \frac{x_4}{\theta} = 0.$$

    It is a quadratic equation in θ**.**

  - **Think of** $\mathbf{y}$ **as a collapsed version (**$y_1 = x_0 + x_1$**) of**

$$\mathbf{x} = (x_0, x_1, x_2, x_3, x_4) \sim Mult(1/2, \theta/4, (1-\theta)/4, (1-\theta)/4, \theta$$

Step 1. Estimate x_0 and x_1 given $y_1 = 125$ and an estimate $\theta^{(i)}$ implies that

$$x_0^{(i)} = 125\frac{1/2}{1/2 + \theta^{(i)}/4} \text{ and } x_1^{(i)} = 125\frac{\theta^{(i)}/4}{1/2 + \theta^{(i)}/4}.$$

Conditional distribution of X_1 given $X_0 + X_1 = 125$ is

$$Bin\left(125, \frac{\theta/4}{1/2 + \theta/4}\right).$$

Step 2. Maximize the resulting binomial problem, obtaining $\theta^{(i+1)} = (x_1^{(i)} + 34)/(x_1^{(i)} + 18 + 20 + 34)$.

Given the *complete data*, MLE of θ is

$$\hat{\theta} = \frac{x_1 + x_4}{x_1 + x_2 + x_3 + x_4}.$$

– **Starting from an initial value of 0.5, the algorithm moved for eight steps as following: $0,608247423$,**

$0,624321051$, 0.626488879, 0.626777323, 0.62677323, 0.626815
0.626820719, 0.626821395, 0.626821484.

- **If** $E-step$ **is hard,**
  - ∗ **replace it by Monte Carlo approach (MCEM)**
  - ∗ **Wei and Tanner (1990,** $JASA$**)**

- **Mixture models:**
  Suppose that the observed data $\mathbf{Y}$ **is a mixture of samples from** k **populations, but that the mixture indicators** $\mathbf{Y}_{miss}$ **are unknown.**
  Think of $\mathbf{Y}_{miss} = (0, 0, \ldots, 0)$ **as a** k**-vector with one position one and the rest zero.**
  - **The complete data is** $\mathbf{X} = (\mathbf{Y}, \mathbf{Y}_{miss})$**.**

Step 1. Estimate the group membership probability for each Y_i **given the current parameter estimates.**

Step 2. Maximize the resulting likelihood, finding in effect the weighted parameter estimates.

- **References**
  - **Dempster, A.P., N.M. Laird, and Rubin, D.R. (1977). Maximum likelihood from incomplete data via the EM algorithm. JRSS-B, 39, 1-38.**
  - **Little, R.J.A. and Rubin, D.B. (1987).** *Statistical Analysis with Missing Data*. **Wiley, New York.**
  - **Tanner, M.A. (1993).** *Tools for Statistical Inference*. **Springer, New York.**
- **Success?**
  Theory shows that the EM algorithm has some very appealing monotonicity properties, improving the likelihood at each iteration.

Though often slow to converge, it does get there!

EM Methods: Motivating example

Consider the example on ABO *blood groups*.

Genotype	Phenotype	Gen freq
AA	A	p_A^2
AO	A	$2p_Ap_O$
BB	B	p_B^2
BO	B	$2p_Bp_O$
OO	O	p_O^2
AB	AB	$2p_Ap_B$

- **The genotype frequencies above assume Hardy-Weinberg equilibrium.**
- **Imagine we sample** n **individuals (at random) and observe their phenotype (but not their genotype).**
- **We wish to obtain the MLES of the underlying allele**

frequencies p_A,p_B, and p_O.

- **Observe $n_A = n_{AA} + n_{AO}$, $n_B = n_{BB} + n_{BO}$, $n_O = n_{OO}$, and n_{AB}, the numbers of individuals with each of the four phenotypes.**
- **We could, of course, form the likelihood function and find its maximum. (There are two free parameters.) But long ago, RA Fisher (or others?) came up with the following (iterative) "allele counting" algorithm**

Allele counting algorithm:
Let n_{AA}, n_{AO}, n_{BB}, and n_{BO} be the (unobserved) numbers of individuals with genotypes AA, AO, BB, and BO, respectively.
Here's the algorithm:

1. **Start with initial estimates $\hat{p}^{(0)} = (\hat{p}_A^{(0)}, \hat{p}_B^{(0)}, \hat{p}_O^{(0)})$.**
2. **Calculate the expected numbers of individuals in each of the genotype classes, given the observed numbers of individuals in each phenotype class and given the current estimates of the allele frequencies. For example:**

$$\begin{aligned} n_{AA}^{(s)} &= E(n_{AA} \mid n_{AA}, \hat{p}^{(s-1)}) \\ &= n_A \hat{p}_A^{(s-1)} / (\hat{p}_A^{(s-1)} + 2\hat{p}_O^{(s-1)}). \end{aligned}$$

3. **Get new estimates of the allele frequencies, imagin-**

ing that the expected n's were actually observed.

$$\begin{aligned}\hat{p}_A^{(s)} &= (2n_{AA}^{(s)} + n_{AO}^{(s)} + n_{AB})/n \\ \hat{p}_B^{(s)} &= (2n_{BB}^{(s)} + n_{BO}^{(s)} + n_{AB})/n \\ \hat{p}_O^{(s)} &= (n_{AO}^{(s)} + n_{BO}^{(s)} + 2n_O)/n.\end{aligned}$$

4. **Repeat steps (2) and (3) until the estimate converges.**

EM algorithm

Consider $X \sim f(x \mid \theta)$ **where** $X = (X_{obs}, X_{miss})$ **and** $f(x_{obs} \mid \theta) = \int f(x_{obs}, x_{miss} \mid \theta) dx_{miss}$**.**

- **Observe** x_{obs} **but not** x_{miss}**.**
- **Wish to find the MLE** $\hat{\theta} = \arg\max_\theta f(x_{obs} \mid \theta)$**.**
- **In many cases, this can be quite difficult directly, but if we had observed** x_{obs}**, it would be easy to find**

$$\hat{\theta}^C = \arg\max_\theta f(x_{obs}, x_{miss} \mid \theta).$$

EM algorithm:

E step:

$$\ell^{(s)}(\theta) = E\{\log f(x_{obs}, x_{miss} \mid \theta) \mid x_{obs}, \hat{\theta}^{(s)}\}$$

M step:

$$\hat{\theta}^{(s+1)} = \arg\max_{\theta} \ell^{(s)}(\theta)$$

Remarks:

- **Nice property: The sequence $\ell[\hat{\theta}^{(s)}]$ is non-decreasing.**
- **Exponential family: $\ell(\theta \mid x) = T(x)^t \eta(\theta) - B(\theta)$.**
  - **$T(x)$ are the sufficient statistics.**
  - **Suppose $x = (y, z)$ where y is observed and z is missing.**
    E step: Calculate $W^{(s)} = E\{T(x) \mid y, \hat{\theta}^{(s-1)}\}$
    M step: Determine $\hat{\theta}^{(s)}$ solving $E\{T(x) \mid \theta\} = W^{(s)}$.
  - **Refer to Wu (1983, Ann Stat 11:95-103) on the convergence of EM algorithm.**

Normal Mixtures

Finite mixtures are a common modelling technique.

- **Suppose that an observable $\mathbf{y}$ is represented as n observations $\mathbf{y} = (y_1, \ldots, y_n)$.**
- **Suppose further that there exists a finite set of J states, and that each y_i is associated with an unobserved state.**
  Thus, there exists an unobserved vector $\mathbf{q} = (q_1, \ldots, q_J)$, where q_i is the indicator vector of length J whose components are all zero except for one equal to unity indicating the unobserved state associated with y_i.
- **Define the complete data to be $\mathbf{x} = (\mathbf{y}, \mathbf{q})$.**

A natural way to conceptualize mixture specifications is to think of the marginal distribution of the indicators

q, and then to specify the distribution of y given q.

- **Assume that the y_i given q_i are conditionally independent with densities $f(y_i \mid q_i)$.**
- **Consider $x_1, \ldots, x_n \overset{iid}{\sim} \sum_{j=1}^{J} p_j f(x_i \mid \mu_j, \sigma)$ where $f(\cdot \mid \mu, \sigma)$ is the normal density.**
  (Here we put the SD rather than the variance here.)

- **Let**

$$y_{ij} = \begin{cases} 1 \text{ if } x_i \text{ is drawn from } N(\mu_j, \sigma) \\ 0 \text{ otherwise} \end{cases}$$

  so that $\sum_j y_{ij} = 1$.
  (x_i) is the observed data; (x_i, y_i) is the complete data.

- **The following is the unobserved** *Complete data log likelihood*

$$\ell(\mu, \sigma, p \mid x, y) = \sum_i \sum_j y_{ij} \{\log p_j + \log f(x_i \mid \mu_j, \sigma)\}$$

  - **How do we estimate it?**
  - **Sufficient statistics**

$$S_{1j} = \sum_i y_{ij} \qquad S_{2j} = \sum_i y_{ij} x_i \qquad S_{3j} = \sum_i y_{ij} x_i^2$$

  - **E step**

$$\begin{aligned} w_{ij}^{(s)} &= E[y_{ij} \mid x_i, \hat{p}^{(s-1)}, \hat{\mu}^{(s-1)}, \hat{\sigma}^{(s-1)}] \\ &= Pr[y_{ij} = 1 \mid x_i, \hat{p}^{(s-1)}, \hat{\mu}^{(s-1)}, \hat{\sigma}^{(s-1)}] \\ &= \frac{\hat{p}_j^{(s-1)} f(x_i \mid \hat{\mu}_j^{(s-1)}, \hat{\sigma}^{(s-1)})}{\sum_j \hat{p}_j^{(s-1)} f(x_i \mid \hat{\mu}_j^{(s-1)}, \hat{\sigma}^{(s-1)})} \end{aligned}$$

Hence

$$S_{1j}^{(s)} = \sum_i w_{ij}^{(s)}, \quad S_{2j}^{(s)} = \sum_i w_{ij}^{(s)} x_i, \quad S_{3j}^{(s)} = \sum_i w_{ij}^{(s)} x_i^2.$$

– **M step**

$$\begin{aligned}
\hat{p}_j^{(s)} &= S_{1j}^{(s)}/n, \\
\hat{\mu}_j^{(s)} &= S_{2j}^{(s)}/S_{1j}^{(s)}, \\
\hat{\sigma}^{(s)} &= \sqrt{\sum_j \left\{ S_{3j}^{(s)} - [S_{2j}^{(s)}]^2 S_{1j}^{(s)} \right\} /n}
\end{aligned}$$

Issues

I. Stopping rules

1. $| \ell(\hat{\theta}^{(s+1)}) - \ell(\hat{\theta}^{(s)}) | < \epsilon$ **for** m **consecutive steps.**
   This is bad**!** ℓ **may not change much even when** θ **does.**
2. $\|\hat{\theta}^{(s+1)} - \hat{\theta}^{(s)}\| < \epsilon$ **for** m **consecutive steps.**
   This runs into problems when the components of θ **are of quite different magnitudes.**
3. $| \hat{\theta}_j^{(s+1)} - \hat{\theta}_j^{(s)} | < \epsilon_1(| \hat{\theta}_j^{(s)} | + \epsilon_2)$ **for** $j = 1, \ldots, p$
   In practice, take

$$\epsilon_1 = \sqrt{\textbf{machine } \epsilon} \approx 10^{-8} \quad \epsilon_2 = 10\epsilon_1 \text{ to } 100\epsilon_1$$

II. Local vs global max

– **There may be** $many$ **modes.**

- **EM may converge to a saddle point**
  Solution: Many starting points

III. Starting points

- **Use information from the context**
- **Use a crude method (such as the method of moments)**
- **Use an alternative model formulation**

IV. Slow convergence

The EM algorithm can be painfully slow to converge near the maximum.
Solution: Switch to another optimization algorithm when you get near the maximum.

V. Standard errors

- **Numerical approximation of the Fisher information (ie, the Hessian)**
- **Louis (1982), Meng and Rubin(1991)**

Probability statements in Statistical Inference

In hypothesis testing, the inferential methods depend on probabilities of two types of errors. In confidence intervals the decisions are associated with probability statements about coverage of the parameters. For both cases the probability statements are based on the distribution of a random sample, $Y_1, \ldots, Y_n$.

- **Simulation of data generating process. (Statistical experiment)**
- **Monte Carlo expectation**
- **Study the random process**