Summary of Modern Methods in Applied Statistics

Jinhong Du

May 14, 2020

Contents

1 The Bayes Rules for Model Comparison **5**

1.1 Likelihood 5

1.1.1 Likelihood And Likelihood Ratio 5

1.1.2 Connection between LRs of Discrete And Continuous Models 5

1.1.3 The Inverse Likelihood Ratio And the Log-likelihood Ratio 6

1.1.4 Bayes Factor 6

1.2 Decision Theory 7

1.2.1 The Prediction Problem 7

1.2.2 Decision Rule 7

1.2.3 Loss Functions And Expected Loss/Integrated Risk 7

1.2.4 The Optimal Decision Rule 7

2 Supervised Learning **9**

2.1 Introduction 9

2.1.1 Setting 9

2.1.2 Procedures 9

2.1.3 Strategies 9

2.2 Generative Learning 9

2.2.1 Linear discriminant analysis 10

2.2.2 Naive Bayesian 10

2.3 Discriminative Learning 10

2.3.1 Logistic Regression 10

2.3.2 K-Nearest Neighbor 10

2.3.3 Pros And Cons 10

3 Complexity of A Learning Rule **11**

3.1 Overfitting 11

3.2 Regularization 11

3.2.1 Computation 11

3.2.2 Orthonormal Cases 12
3.3 Model Validation 12
3.3.1 Leave-One-Out Validation 12
3.3.2 k-Fold Cross Validation 12

4 Empirical Bayes 13
4.1 Empirical Bayes 13
4.1.1 Bayes Conjugate 13
4.1.2 Empirical Bayes 13
4.1.3 Connection between Empirical Bayes Shrinkage and Penalized Likelihood . . 13
4.1.4 Normal Means Model And Orthonormal Regression 14
4.2 Wavelet Shrinkage 14
4.2.1 Wavelet Smoothing 14
4.2.2 Trend Filtering 15

5 Density Estimation 17
5.1 Histogram 17
5.2 Kernel Density Estimates 18

6 Mixture Models 19
6.1 Mixture Models 19
6.2 EM Algorithm 19
6.2.1 Algorithm 19
6.2.2 Variational View 20
6.2.3 Majorization-Minimization View 20

7 MCMC 21
7.1 Metropolis Hastings Algorithm 21
7.2 Gibbs Sampling 21
7.2.1 Directed Acyclic Graphical Models (DAGs) 21

8 Hypothesis Testing 23
8.1 Introduction 23
8.2 Difficulty of Calibrating p Values 24
8.3 p Values and Bayes Factors 24
8.4 Multiple Testing 24

Chapter 1

The Bayes Rules for Model Comparison

1.1 Likelihood

1.1.1 Likelihood And Likelihood Ratio

Given data x, the *likelihood* for a fully-specified discrete (or continuous) model is the probability mass function (or the probability density function) of x under the model:

$$L(M) := p(x|M)$$

where $p(\cdot|M)$ denotes the probability mass function (or the probability density function) for model M. And the *likelihood ratio* comparing two fully-specified models M_1 versus M_0 is defined as

$$LR(M_1, M_0) := \frac{L(M_1; x)}{L(M_0; x)}.$$

Large values of $LR(M_1, M_0)$ indicate that the data are much more probable under model M_1 than under model M_0, and so indicate support for M_1. Conversely, small values of LR indicate support for model M_0.

Individual likelihood values are mostly irrelevant: it is likelihood ratios that matter. If the likelihood ratio for model 1 vs model 2 is c, then this means the data favour model 1 by a factor of c. Or, if $c < 1$ then it means the data favour model 2 by a factor of $\frac{1}{c}$.

1.1.2 Connection between LRs of Discrete And Continuous Models

Suppose we assume that measurement precision is ϵ. So the "observation" $X = x$ really means $X \in [x-\epsilon, x+\epsilon]$. Then the likelihood for a model M, given this observation, is $\mathbb{P}(X \in [x-\epsilon, x+\epsilon]|M)$. Provided that the density $p(x|M)$ is approximately constant in the region within radius ϵ around

x, then this probability is approximately $2\epsilon p(x|M)$. Thus the LR for two models M_1 vs M_0, is given by

$$LR = \frac{\mathbb{P}(X \in [x-\epsilon, x+\epsilon]|M_1)}{\mathbb{P}(X \in [x-\epsilon, x+\epsilon]|M_0)} \approx \frac{2\epsilon p(x|M_1)}{2\epsilon p(x|M_0)} = \frac{p(x|M_1)}{p(x|M_0)}.$$

In most cases, the LR for model M_1 vs model M_0 for a continuous random variable X, given observation $X = x$, can be well approximated by the ratio of the model densities of X, evaluated at x. This approximation comes from assuming that the model density functions are approximately constant within the neighborhood of x that has radius equal to the measurement precision.

1.1.3 The Inverse Likelihood Ratio And the Log-likelihood Ratio

From the definition of the LR, the LR for M_0 versus M_1 is the just inverse of the LR for M_1 versus M_0. That is

$$LR(M_0, M_1; x) = \frac{1}{LR(M_1, M_0; x)}.$$

For many reasons, it is common to work with the log likelihood ratio,

$$LLR := \log(LR).$$

Usually mathematicians work with base logarithms base e, and we will use that convention unless otherwise stated. Although the usual convention is to use log base e, it can sometimes be useful to work with logarithms base 10 to make the inverse logarithm operation easier for human calculation.

1.1.4 Bayes Factor

For fully specified models, the likelihood ratio is also known as the *Bayes Factor (BF)*, so we could also define the Bayes Factor for M_1 vs M_0 as

$$BF(M_1, M_0) := \frac{p(x|M_1)}{p(x|M_0)}.$$

When comparing fully specified models the LR and BF are just two different names for the same thing.

Let $Z_i \in \{0, 1\}$ denote whether x_i was generared from model M_0 or M_1. From Bayes Theorem, we have

$$\mathbb{P}(Z_i = 1|x_i) = \frac{\mathbb{P}(x_i|Z_i = 1)\mathbb{P}(Z_i = 1)}{\mathbb{P}(x_i)}$$
$$\mathbb{P}(Z_i = 0|x_i) = \frac{\mathbb{P}(x_i|Z_i = 0)\mathbb{P}(Z_i = 0)}{\mathbb{P}(x_i)}.$$

Taking the ratio of these gives

$$\underbrace{\frac{\mathbb{P}(Z_i = 1|x_i)}{\mathbb{P}(Z_i = 0|x_i)}}_{\text{posterior odds}} = \underbrace{\frac{\mathbb{P}(Z_i = 1)}{\mathbb{P}(Z_i = 0)}}_{\text{prior odds}} \times \underbrace{\frac{\mathbb{P}(x_i|Z_i = 1)}{\mathbb{P}(x_i|Z_i = 0)}}_{\text{Bayes factor}}.$$

1.2 Decision Theory

1.2.1 The Prediction Problem

Consider the problem of predicting an outcome Y on the basis of inputs (or “features”or “predictors”) X. Typically Y might be a one-dimensional outcome (discrete or continuous) and X a multi-dimensional input. If Y is discrete then this is often referred to as a *classification problem*; if Y is continuous then this is often referred to as a *regression problem.*

1.2.2 Decision Rule

A *decision rule* is simply a way of predicting Y from X. That is it is a function $\hat{Y}(X)$, which for any given X produces a predicted value $\hat{Y}$ for Y.

1.2.3 Loss Functions And Expected Loss/Integrated Risk

The *loss function* is a function $L : \mathcal{Y} \times \mathcal{Y} \mapsto \mathbb{R}$ such that $L(\hat{Y}, Y)$ measures how wrong the prediction is.

If Y is continuous and real-valued, then the loss function can be

- Squared loss: $L(\hat{Y}, Y) = (\hat{Y} - Y)^2$.
- Absolute loss: $L(\hat{Y}, Y) = |\hat{Y} - Y|$.

If Y is discrete, then the loss function can be the 0-1 loss: $L(\hat{Y}, Y) = \mathbb{1}_{\{\hat{Y} \neq Y\}}$.

For a decision rule, the *expected loss* or *integrated risk* is defined as

$$r(\hat{Y}) = \int \int L(\hat{Y}(X), Y) \mathrm{d}X \mathrm{d}Y.$$

1.2.4 The Optimal Decision Rule

The optimal decision rule $\hat{Y}_{opt}(X)$ is a decision rule that minimizes the expected loss r, or equivalently one that minimizes the conditional expected loss:

$$\hat{Y}_{opt}(Y) = \arg\min_{a} \int L(a, Y) p(Y|X) \mathrm{d}Y.$$

Since

$$\begin{aligned} r(\hat{Y}) &= \int \left[\int L(\hat{Y}(X), Y) p(Y|X) \mathrm{d}Y \right] p(X) \mathrm{d}X \\ &\geq \int \left[\int L(\hat{Y}_{opt}(X), Y) p(Y|X) \mathrm{d}Y \right] p(X) \mathrm{d}X = r(\hat{Y}_{opt}) \end{aligned}$$

The conditional distribution $Y|X$ is sometimes be referred to as the *posterior distribution of Y given data X*, and computing this distribution is sometimes referred to as *performing Bayesian inference for Y.*

Thus, the above result can be thought of as characterizing the optimality of Bayesian inference in terms of a “frequentist”measure (r) which measures long-run performance across many samples (X, Y) from $p(X, Y)$. For example, predicting Y by its posterior mean, $\mathbb{E}(Y|X)$, is optimal in terms of expected squared loss (with expectation taken across $p(X, Y)$).

Because of this connection with Bayesian inference, the optimal value $r(\hat{Y}_{opt})$ is sometimes referred to as the *Bayes risk*, and $\hat{Y}_{opt}$ is referred to as a *Bayes decision rule*.

In practice, Bayesian inference is optimal, on average, if both the prior distribution $p(Y)$ and likelihood $p(X|Y)$ are correct.

Chapter 2

Supervised Learning

2.1 Introduction

2.1.1 Setting

In supervised learning, we have a feature population $X \in \mathcal{X} \subset \mathbb{R}^p$ as well as its corresponding label population $Y \in \mathcal{Y} \subset \mathbb{R}$, and we can observe the iid samples $\{(X_i, Y_i) : i = 1, \ldots, n\}$. We assume there is an underlying map from $\mathcal{X}$ to $\mathcal{Y}$, and the goal of supervised learning is to predict outcome Y from some predictors X.

2.1.2 Procedures

1. A training set $\{(X_i, Y_i) : i = 1, \ldots, n\}$ is revealed.
2. Learn a decision rule to predict Y from X from the training set. Some procedures may provide just point estimates; others may give intervals, or a full distribution $p(Y|X)$.
3. Apply the rule for future examples (usually from the same or similar population).

2.1.3 Strategies

If we know $p(X, Y)$, then we can find the optimal decision rule as shown in the previous chapter. And the optimal decision rule will depends on the conditional distribution $p(Y|X)$. There are two strategies to learn the conditional distribution, generative and discriminative learning.

2.2 Generative Learning

In *generative learning*, we use training data to learn a model $\hat{p}(X, Y)$. Then compute $\hat{p}(Y|X)$ by Bayes Theorem.

If Y is discrete, then learning $\hat{p}(Y)$ is easy, and $\hat{p}(X|Y)$ is the hard part.

2.2.1 Linear discriminant analysis

Consider the Gaussian models within classes $X|Y = k \sim \mathcal{N}_p(\mu_k, \Sigma)$. The log posterior odds ratio for class k_1 and k_2 has simple form,

$$\begin{aligned}\log \frac{\mathbb{P}(Y = k_1|X = x)}{\mathbb{P}(Y = k_2|X = x)} &= \log \frac{\mathbb{P}(Y = k_1)\mathbb{P}(X = x|Y = k_2)}{\mathbb{P}(Y = k_2)\mathbb{P}(X = x|Y = k_1)} \\ &= \log \frac{\mathbb{P}(Y = k_1)}{\mathbb{P}(Y = k_2)} - \frac{1}{2}(\mu_{k_1} + \mu_{k_2})^\top \Sigma^{-1}(\mu_{k_1} - \mu_{k_2}) + x^\top \Sigma^{-1}(\mu_{k_1} + \mu_{k_2})\end{aligned}$$

which is linear in x. So we can estimate μ and Σ by maximum likelihood from training set. Also we can estimate the prior distribution $\mathbb{P}(Y = k)$ by maximum likelihood from training set. Then we can plug in these estimates to compute $\mathbb{P}(Y = k|X = x)$.

2.2.2 Naive Bayesian

When X is a vector, we can assume that its elements are independent given Y, i.e., $p(X|Y) = \prod_{j=1}^{p} p(X_j|Y)$. This simplifies problem to modelling univariate distributions, but it tends to underestimate uncertainty of classifications. This method could work poorly in presence of strong dependence.

2.3 Discriminative Learning

In *discriminative learning*, we use training data to directly learn $p(Y|X)$ (or even just $\mathbb{E}(Y|X)$) directly.

2.3.1 Logistic Regression

2.3.2 K-Nearest Neighbor

2.3.3 Pros And Cons

- In discriminative learning, we don't need to model X (potentially more robust).
- In discriminative learning, we don't get to model X (less efficient if modeling assumptions are good).
- The generative models can incorporate unlabelled data.

Chapter 3

Complexity of A Learning Rule

Some supervised learning procedures are more complex/flexible than others

Defining complexity/flexibility is not easy, but sometimes corresponds to ”number of parameters” in parametric cases. Vapnik–Chervonenkis (VC) Dimension provides a general approach; we won’t cover it here though.

3.1 Overfitting

As the complexity of the model increases, the flexibility of learning rule increases, and its test error eventually starts to increase (even though fit to training data improves). Can think of it as ”fitting the noise” instead of ”fitting the signal”.

3.2 Regularization

One way to deal with overfitting is to use regularization. Regularization refers to modifying or constraining a training procedure to make it less flexible, usually with the goal of avoiding (or reducing) overfitting and improving generalization performance.

3.2.1 Computation

L_0 norm is non-convex and computationally hard. Until recently, intractable for modest p; heuristics used (most simply, forward stepwise regression) More recently, work using sophisticated optimization algorithms can lead to provably correct solutions for even quite big problems in minutes/hours, eg: https://arxiv.org/abs/1803.01454 and software https://github.com/hazimehh/L0Learn.

L_1 is a convex optimization problem and can be solved efficiently for any λ. We can get the whole ”solution path” as λ varies.

L_2 is convex, and analytically tractable, but computation of this scales with np^2 or pn^2 (whichever is smaller) for Ridge Regression.

3.2.2 Orthonormal Cases

When X has orthonormal columns, then the estimated coefficients for linear regression satisfy

$$\begin{aligned}
\hat{\beta}_j^0 &= \hat{\beta}_j \mathbb{1}_{\{|\hat{\beta}_j| \geq |\hat{\beta}_{(1)}|\}} \mathbb{1}_{\{|\hat{\beta}_j| \geq |\hat{\beta}_{(M)}|\}} \\
\hat{\beta}_j^1 &= \operatorname{sign}(\hat{\beta}_j)(|\hat{\beta}_j| - \lambda)_+ \\
\hat{\beta}_j^2 &= \frac{\hat{\beta}_j}{1+\lambda}
\end{aligned}$$

where $\hat{\beta}_j$ is the OLS estimator and $\hat{\beta}_{(m)}$ is mth estimated coefficient chosen from subset selection with size M.

3.3 Model Validation

3.3.1 Leave-One-Out Validation

3.3.2 k-Fold Cross Validation

Chapter 4

Empirical Bayes

4.1 Empirical Bayes

4.1.1 Bayes Conjugate

To compute the posterior density of a continuous parameter, up to a normalizing constant, we multiply the likelihood by the prior density,

$$p(q|X) \propto p(X|\theta)p(\theta).$$

In simple cases we may find that the result is the density of a distribution we recognize. If so, we can often use known properties of that distribution to compute quantities of interest.

In cases when we do not recognize the posterior distribution, we may need to use computational methods (like Importance Sampling or Markov chain Monte Carlo) to compute quantities of interest.

If $X|q \sim \text{Binomial}(n, q)$ and $q \sim \text{Beta}(a, b)$, then $q|X \sim \text{Beta}(X + a, n - X + b)$.

If $X|\mu \sim \text{Poisson}(\mu)$ and $\mu \sim \text{Gamma}(n, \lambda)$, then $\mu|X \sim \text{Gamma}(a + X, b + 1)$.

4.1.2 Empirical Bayes

We assume that $X|\theta \sim F(\cdot; \theta)$ and the parameters have a prior $\theta \sim G(\cdot; \alpha)$. The empirical bayes

1. Compute $\hat{\alpha}$, the MLE estimate of prior parameters α from observed data $X_1, \dots, X_n$.
2. Compute $\hat{\theta}$, the estimate of θ by posterior mean.
3. Make inference about data $X_1, \dots, X_n$ based on $\hat{\theta}$.

4.1.3 Connection between Empirical Bayes Shrinkage and Penalized Likelihood

Using the penalized likelihood is equivalent to using the posterior mode (instead of posterior mean).

Since $p(\theta|X) \propto L(\theta; X) \times p(\theta)$,

$$\text{Posterior Mode} = \arg\max L(\theta)p(\theta) = \arg\max l(\theta; X) + \log[p(\theta)]$$

In this view, penalty can be interpreted as the log prior term, e.g. normal prior for L_2, double-exponential prior for L_1.

However, the posterior mode is often very different from the posterior mean and there is little theoretical support when estimating continuous parameters (0-1 loss makes no sense). For example, Bayesian inference (posterior mean or median) with Laplace prior can be very different from inference with LASSO. The former is not sparse while the latter is often sparse.

For L_2 penalty, the prior is normal, so posterior is normal (with normal likelihood). For normal, the posterior mean, mode and median are the same. So Ridge regression is equivalent to Bayesian posterior mean with independent normal priors on regression coefficients.

4.1.4 Normal Means Model And Orthonormal Regression

Normal means model is closely connected to regression with orthonormal covariates/predictors.

For regression, $Y = X\beta + e$ with $e \sim N(0, (1/\tau)I_n)$. Suppose the columns of X are orthonormal, i.e., $X^\top X = I_p$. So $\hat{\beta} = (X^\top X)^{-1} X^\top Y = X^\top Y$. The log-likelihood is

$$\begin{aligned} l(\beta) &= -0.5\tau (Y - X\beta)^\top (Y - X\beta) + C \\ &= -0.5\tau[\beta^\top (X^\top X)\beta - 2Y^\top X\beta] + C \\ &= -0.5\tau[\beta^\top \beta - 2Y^\top X\beta] + C \\ &= -0.5\tau[\beta^\top \beta - 2\hat{\beta}^\top \beta] + C \end{aligned}$$

For normal means model, assume data $\beta_j \sim N(\theta_j, s_j^2 = 1/\tau)$. Then the log-likelihood is

$$\begin{aligned} l(\theta) &= -0.5\tau \sum_j (\beta_j - \theta_j)^2 + C \\ &= -0.5\tau(\beta_j - \theta)^\top (\beta_j - \theta) + C \\ &= -0.5\tau[\theta^\top \theta - 2\beta_j^\top \theta] + C' \end{aligned}$$

So doing inference for β in regression case is equivalent to normal means. Similarly if we assume X to be orthogonal and scaled to have unit variance ($X^\top X = nI_p$ instead of I_p) then similar algebra shows inference for b in regression case is equivalent to normal means with $s_j^2 = \frac{1}{n\tau}$. So for orthogonal X, we can do Empirical Bayes shrinkage in regression using Empirical Bayes normal means (if we can estimate the residual variance). For non-orthogonal X it is much harder.

4.2 Wavelet Shrinkage

4.2.1 Wavelet Smoothing

Suppose we want to fit model $\boldsymbol{Y} = \boldsymbol{\mu} + \boldsymbol{\epsilon}$, with $\mu_1, \dots, \mu_n$ ”smooth” or ”spatially structured”. For example, $|\mu_t - \mu_s|$ is usually small when $|t - s|$ is small and we may want to allow for occasional big changes in $\boldsymbol{\mu}$.

We can transform $\boldsymbol{Y}$ into parts that capture changes at different resolutions by Wavelet transformation. There are different types of wavelet transforms and much theory, but we just focus on simplest case of *Haar wavelets*.

The wavelet coefficients (and the mean/sum) can be written as $\boldsymbol{WY}$ for a matrix $\boldsymbol{W}$ that has orthonormal rows, $\boldsymbol{W}\boldsymbol{W}^\top = \boldsymbol{I}$. There is a very efficient algorithm (”pyramid algorithm” for ”discrete wavelet transform”) for computing $\boldsymbol{WY}$.

Wavelet Transform of Non-Parameteric Regression

$\boldsymbol{Y} = \boldsymbol{\mu} + \boldsymbol{\epsilon}$ with $\boldsymbol{\epsilon} \sim \mathcal{N}(\boldsymbol{0}, \sigma^2 \boldsymbol{I})$ becomes $\boldsymbol{WY} = \boldsymbol{W\mu} + \boldsymbol{W\epsilon}$ with $\boldsymbol{W}\boldsymbol{W}^\top = \boldsymbol{I}$. This is equivalent to $\tilde{\boldsymbol{Y}} = \tilde{\boldsymbol{\mu}} + \tilde{\boldsymbol{\epsilon}}$ with $\tilde{\boldsymbol{\epsilon}} \sim \mathcal{N}(0, \sigma^2 \boldsymbol{I})$ and $\tilde{\boldsymbol{\mu}}$ is “sparse”.

This is just the normal means problem. So we can apply EB shrinkage to shrink $\tilde{\boldsymbol{\mu}}$.

4.2.2 Trend Filtering

We consider the simplest case. Suppose $\boldsymbol{Y} = \boldsymbol{\mu} + \boldsymbol{\epsilon}$ where $|\mu_k - \mu_{k+1}|$ is sparse or at least many elements are near 0. This can be implemented via L_1 penalized regression:

$$\arg\min \|\boldsymbol{Y} - \boldsymbol{X\beta}\|_2 + \lambda \|\boldsymbol{\beta}\|_1,$$

where columns of $\boldsymbol{X}$ is given by $\boldsymbol{X}_i = \sum_{j=1}^{i} \boldsymbol{e}_j$. Since $\boldsymbol{\mu} = \boldsymbol{X\beta}$, we have $\mu_k = \sum_{j=1}^{k} \beta_j$ and $\beta_k = \mu_k - \mu_{k-1}$.

Chapter 5

Density Estimation

5.1 Histogram

We can use piecewise constant estimate of density.

1. Divide range up into J discrete bins, b_j, $j = 1, \ldots, J$.
2. Let $p_j = \mathbb{P}(x_1 \in b_j)$. Then the maximum likelihood estimate is

$$\begin{aligned} \hat{p}_j &= \frac{1}{n} \sum_{i=1}^{n} \mathbb{1}_{\{i : x_i \in b_j\}} \\ \mathbb{E}(\hat{p}_j) &= p_j \\ \operatorname{Var}(\hat{p}_j) &= \frac{1}{n} p_j (1 - p_j) \end{aligned}$$

3. The corresponding estimate of density in b_j is $\hat{d}_j = \frac{\hat{p}_j}{|b_j|}$ and $\mathbb{E}(\hat{d}_j) = \frac{p_j}{|b_j|}$.

   CV for bin size Loss function: Kullback–Leibler divergence

   if you estimate

$$q$$

and truth is

$$p$$

$KL(p \| q) = L(p, q) = -\int p(x) \log \left(\frac{q(x)}{p(x)} \right) \mathrm{md}x$ Also written

$$KL(p \| q)$$

to emphasise direction Properties:

1. $KL(p \| p) = 0$;
2. $\min_q KL(p \| q) = KL(p \| p)$.
3. $KL(p \| q) \neq KL(q \| p)$ in general.

We do not know p, but given a sample from p, we can estimate the loss $L(x_1, \ldots, x_n; q) = -\sum_j \log(q(x_j)) + \text{const}$. Also, we must avoid overfitting by using CV, e.g., train on 90% and test on 10%.

5.2 Kernel Density Estimates

$\hat{p}(x) = \frac{1}{n} \sum\limits_j k(x - x_j)$ where k is a kernel (density), e.g. the density of $N(0, \sigma^2)$.

The width/sd of kernel (σ here) is called *bandwidth*. Small bandwidth is analogous to small bin size in histogram, which gives lower bias of density estimate but at expense of higher variance.

Chapter 6

Mixture Models

6.1 Mixture Models

Mixture models are used in clustering and density estimation.

Suppose that we have data $X = (X_1, \ldots, X_n)^\top$ from the mixture density $p(x) = \sum_k \pi_k f_k(x; \theta)$. We can introduce indicators $Z_{ik} = 1$ if observation i came from component k. That is $\mathbb{P}(Z_{ik} = 1) = \pi_k$ and $Z_i \sim Mult(1, \pi)$.

Terminology:

- Z is sometimes referred to as "missing data" or "latent variables".
- (X, Z) is referred to as "complete data" or "augmented data" with parameters π, θ. The complete data likelihood is

$$p(X, Z|\theta) = \prod_i \prod_k [\pi_k f_k(X_i; \theta)]^{Z_{ik}}$$
$$\log p(X, Z|\theta) = \sum_{i,k} Z_{ik}[\log(\pi_k) + \log f_k(X_i; \theta)].$$

6.2 EM Algorithm

6.2.1 Algorithm

1. Initialize $\theta^{(0)}$, and iterate the following steps for $i = 0, 1, 2, \ldots$.
2. E step: form the expected complete data log-likelihood (Q function)

$$Q(\theta, \theta^{(i)}) = E_{Z|D,\theta^{(i)}}[\log p(D, Z|\theta)].$$

3. M step: maximise the Q function $\theta^{(i+1)} = \arg\max_\theta Q(\theta, \theta^{(i)})$

Every iteration is guaranteed to increase (or at least, not decrease) the log-likelihood $l(\theta) = \log p(D|\theta)$.

6.2.2 Variational View

Let q denote any distribution on the latent variables Z. Let $H(q)$ denote the entropy of q: $H(q) = -\mathbb{E}_q \log(q(Z))$. Let θ denote any parameter values. Define $F(\theta, q) := \mathbb{E}_q[\log p(D, Z|\theta)] + H(q)$

Lemma 1. *For fixed θ, the optimal $\hat{q}_\theta := \arg\max F(\theta, q)$ is the conditional distribution $\hat{q}_\theta(Z) = p(Z|D, \theta)$.*

We can think of the EM algorithm as maximizing $F(\theta, q)$ by iteratively maximizing over θ and q:

$$q^{(i+1)} = \arg\max_q F(\theta^{(i)}, q)$$
$$\theta^{(i+1)} = \arg\max_\theta F(\theta, q^{(i+1)})$$

By the lemma, the first step is accomplished by the conditional distribution of $p(Z|\theta, D)$ that is computed in the E step. And the second step above is then seen to be the ”M-step” of the EM algorithm.

6.2.3 Majorization-Minimization View

Recall that $KL(q\|p) = -\mathbb{E}_q\left[\log\left(\frac{p(x)}{q(x)}\right)\right]$ is the Kullback–Leibler divergence from q to p. It is non-negative, and takes its minimum value 0 at $q = p$.

Since

$$p(D|\theta) = \frac{p(D, Z|\theta)}{p(Z|D, \theta)}$$
$$\log p(D|\theta) = \log p(D, Z|\theta) - \log p(Z|D, \theta),$$

for any distribution q_0 on Z we have

$$\begin{aligned}\log p(D|\theta) &= \mathbb{E}_{q_0} \log p(D, Z|\theta) - E_{q_0} \log p(Z|D, \theta)\\ &= \mathbb{E}_{q_0} \log p(D, Z|\theta) - \mathbb{E}_{q_0}[\log(p(Z|D, \theta)/q_0(Z))] - \mathbb{E}_{q_0}[\log q_0(Z)]\\ &= \mathbb{E}_{q_0} \log p(D, Z|\theta) + KL(q_0\|q_\theta) - \mathbb{E}_{q_0}[\log q_0(Z)].\end{aligned}$$

So, setting $q_0 = q_{\theta_0}$ yields:

$$l(\theta) = Q(\theta, \theta_0) + KL(q_{\theta_0}\|q_\theta) + const.$$

Then maximizing $l(\theta)$ is equivalent to maximizing $M(\theta; \theta_0) := Q(\theta, \theta_0) + KL(q_{\theta_0}\|q_\theta)$ for any θ_0. Note that $M(\theta; \theta_0) \geq Q(\theta; \theta_0)$ with equality at $\theta = \theta_0$. We say Q minorizes M. So maximizing $Q(\theta; \theta_0)$ over θ is guaranteed to increase $M(\theta; \theta_0)$ and so increase $l(\theta)$.

Chapter 7

MCMC

7.1 Metropolis Hastings Algorithm

1. Initialize, $x_1 = x$ say.
2. For $t = 1, 2, \dots$
   (a) sample y from the *transition kernel* or *proposal distribution* $Q(y|x_t)$. Think of y as a proposed value for x_{t+1}.

   (b) Compute the *acceptance probabilty*,

   $$A = \min\left\{1, \frac{\pi(y)Q(x_t|y)}{\pi(x_t)Q(y|x_t)}\right\},$$

   where π is the target distribution that is known only up to a constant and want to approximate.

   (c) With probability A accept the proposed value, and set $x_{t+1} = y$. Otherwise set $x_{t+1} = x_t$.

When the transition kernel is symmetric, i.e. $Q(y|x) = Q(x|y)$ for all x, y, the algorithm is called Metropolis algorithm. A continuous kernel can be

$$Q(y|x) = \frac{1}{\sqrt{2\pi}} e^{-\frac{1}{2}(x-y)^2}.$$

This kind of kernel, which adds some random number to the current position x to obtain y, is often used in practice and is called a “random walk” kernel.

7.2 Gibbs Sampling

7.2.1 Directed Acyclic Graphical Models (DAGs)

Chapter 8

Hypothesis Testing

8.1 Introduction

When performing a statistical hypothesis test, like comparing two models, if the hypotheses completely specify the probability distributions, these hypotheses are called *simple hypotheses.* If it specifies a set of distributions, e.g. $\mu > \mu_0$, then it is therefore a *composite hypothesis.*

Given the observed data $X_1, \ldots, X_n$, we can measure the relative plausibility of H_1 to H_0 by the log-likelihood ratio:

$$\log \frac{f(X_1, \ldots, X_n | H_1)}{f(X_1, \ldots, X_n | H_0)}.$$

The *generalized log-likelihood ratio* is given by

$$\Lambda^* = \log \frac{\max\limits_{\theta_1 \in \Theta_1} f(X_1, \ldots, X_n | \theta_1)}{\max\limits_{\theta_0 \in \Theta_0} f(X_1, \ldots, X_n | \theta_0)}.$$

For technical reasons, it is preferable to use the following related quantity:

$$\Lambda_n = \log \frac{\max\limits_{\theta \in \Omega} f(X_1, \ldots, X_n | \theta)}{\max\limits_{\theta_0 \in \Theta_0} f(X_1, \ldots, X_n | \theta_0)},$$

where $\Omega = \Theta_0 \cup \Theta_1$.

Theorem 1. (Wilks's Theorem) *Suppose that the dimension of Ω is v and the dimension of Θ_0 is r. Under regularity conditions and assuming H_0 is true, the distribution of Λ_n tends to a chi-squared distribution with degrees of freedom equal to $v - r$ as the sample size tends to infinity.*

With this theorem in hand (and for n large), we can compare the value of our log-likehood ratio to the expected values from a χ^2_{v-r} distribution.

8.2 Difficulty of Calibrating p Values

A key problem with p values, when testing null hypotheses, is that they can be difficult to calibrate. That is, it is hard to answer the question "If I get a p-value of 0.01 (or any other number) how strong is the evidence against the null hypothesis?"

8.3 p Values and Bayes Factors

Consider Bayes Factor (BF) for testing whether normal mean is zero or not

$$H_{0i} : \theta_i = 0, \qquad H_{1i} : \theta_i \sim \mathcal{N}(0, \sigma^2).$$

In addition we will assume that we have data (e.g. the results of a drug trial) that give us imperfect information about . Specifically we assume $X_i|\theta_i \sim \mathcal{N}(\theta_i, 1)$. This implies that:

$$X_i \overset{H_{0i}}{\sim} \mathcal{N}(0,1), \qquad X_i \overset{H_{1i}}{\sim} \mathcal{N}(0, \sigma^2 + 1).$$

Consequently the Bayes Factor (BF) comparing H_1 vs H_0 can be computed as follows:

$$BF(X_i) = \frac{p(X_i|H_{0i})}{p(X_i|H_{1i})}.$$

which depends both on the data X_i and the choice of σ (effect size).

Note some key features:

- Answer of what proportion of tests near 0.05 are actually null has to depend on prior proportion (prior odds), and effect sizes! (BF). That is, this is a fundamental issue, not just a question of whether wants to take a Bayesian approach or not.
- As σ increases from 0 the BF is initially 1, rises to a maximum, and then gradually decays. This behavior, which occurs for any x, perhaps helps provide intuition into why it is even possible to derive an upper bound.
- In the limit as $\sigma \infty$ it is easy to show that (for any x) the BF goes to 0. This is an example of "Bartlett's paradox", and illustrates why you should not just use a "very flat" prior for θ under H_1: the Bayes Factor will depend on how flat you mean by "very flat", and in the limit will always favor H_0.

8.4 Multiple Testing