Math 459 Lecture 17

Todd Kuffner

Linear Regression Model

$$Y_i = \beta_0 + \beta_1 X_{i1} + \beta_2 X_{i2} + \cdots + \beta_{k-1} X_{i,k-1} + \varepsilon_i$$

- response variable Y related to predictors/covariates/explanatory variables $X_1, \ldots, X_{k-1}$
- observe n predictor-response pairs $\{X_{ij}, Y_i\}$, $i = 1, \ldots, n$, $j = 0, 1, \ldots, k-1$
- $\varepsilon_i \overset{iid}{\sim} g(\cdot)$; usually $E(\varepsilon_i) = 0$ and $\mathrm{Var}(\varepsilon_i) = \sigma^2$

Model in Matrix Form

$$Y = X\beta + \varepsilon$$

with

$$Y = \begin{pmatrix} Y_1 \\ \vdots \\ Y_n \end{pmatrix}, \quad X = \begin{pmatrix} 1 & X_{11} & \cdots & X_{1,k-1} \\ 1 & X_{21} & \cdots & X_{2,k-1} \\ \vdots & \vdots & \ddots & \vdots \\ 1 & X_{n1} & \cdots & X_{n,k-1} \end{pmatrix},$$

$$\beta = \begin{pmatrix} \beta_0 \\ \beta_1 \\ \vdots \\ \beta_{k-1} \end{pmatrix}, \quad \varepsilon = \begin{pmatrix} \varepsilon_1 \\ \vdots \\ \varepsilon_n \end{pmatrix}$$

What this actually means

The random variable Y is a linear combination of the random variables $X_1, \ldots, X_{k-1}, \varepsilon$.

- often the goal is prediction of $Y \Rightarrow$ (conditional) mean of Y is best predictor *under squared-error loss*
- the conditional distribution $Y|X$ has

$$E(Y|X) = \beta_0 + \sum_{j=1}^{k-1} \beta_j X_j + E(\epsilon|X), \ \mathrm{Var}(\epsilon|X) = \sigma^2 I_{n\times n}$$

- each unknown parameter β_j represents the expected change in Y per unit change in X_j, when all other predictors are held fixed
- i.e. β_j is the partial derivative of the conditional mean $E(Y|X)$ w.r.t. X_j, $\beta_j = \partial E(Y|X)/\partial X_j$
- 'linear' means the *parameters* enter linearly

Frequentist Estimation by Least Squares

The least squares estimator is the solution to

$$\min_{\beta}(Y - X\beta)^T(Y - X\beta)$$

and this is equal to

$$\hat{\beta} = (X^T X)^{-1} X^T Y, \;\; \hat{\sigma}^2 = \frac{(Y - X\hat{\beta})^T(Y - X\hat{\beta})}{n-k}$$

provided $(X^T X)^{-1}$ exists.

- if we assume $\varepsilon \sim \mathcal{N}(0, \sigma^2 I_{n\times n})$, the likelihood function is

$$L(\beta, \sigma^2; Y, X) = \frac{1}{(2\pi)^{n/2}\sigma^n} \exp\left(-\frac{1}{2\sigma^2}(Y - X\beta)^T(Y - X\beta)\right)$$

- maximizing this is equivalent to the least squares problem

Towards a Bayesian Analysis

When a Bayesian encounters this model, she sees X and Y and immediately thinks there must be some sampling model for each of them:

$$p(X|\psi), \quad p(Y|\theta)$$

but in fact we have a joint density $f(x, y|\psi, \theta)$ and hence a joint likelihood $L(\psi, \theta)$.

- we need a joint prior $p(\psi, \theta)$
- Bayesians like to assume the distribution of X, $p(X|\psi)$, and hence the parameter ψ, provides no information about $p(Y|X, \theta)$
- i.e. prior independence of ψ and θ, $p(\psi, \theta) = p(\psi)p(\theta)$

If we assume that $p(\psi, \theta) = p(\psi)p(\theta)$ and, if the likelihood factors, then the posterior distribution factors

$$p(\psi, \theta|X, y) = p(\psi|X)p(\theta|X, y)$$

and the second factor (i.e. the regression model) can be studied by itself without information loss:

$$p(\theta|X, y) \propto p(\theta)p(y|X, \theta)$$

In a fixed design, the Xs are not random, and then $p(X)$ is known (there are no parameters ψ).

Bayesian Linear Regression with Noninformative Prior

Common to use uniform prior on $(\beta, \log \sigma)$, which is the same as

$$p(\beta, \sigma^2 | X) \propto \sigma^{-2}$$

Based on the above justification, and to facilitate the use of Gibbs sampling, we factor the joint posterior as

$$p(\beta, \sigma^2 | X, y) = p(\beta | \sigma^2, X, y) p(\sigma^2 | X, y)$$

Start by finding the posterior for β, conditional on σ, then find marginal distribution of σ^2.

Conditional posterior $\beta|\sigma$

$\beta|\sigma$ is the exponential of a quadratic form in β $\Rightarrow$ conditional posterior

$$\beta|\sigma, X, y \sim \mathcal{N}(\hat{\beta}, (X^T X)^{-1}\sigma^2)$$

which can be seen noting that $\hat{\beta} = (X^T X)^{-1} X^T Y$

Marginal posterior of σ^2

Written as

$$p(\sigma^2|y) = \frac{p(\beta, \sigma^2|X, y)}{p(\beta|\sigma^2, X, y)}$$

which is a scaled inverse-χ^2 so that

$$\sigma^2 \sim \text{Inv-}\chi^2(n-k, s^2)$$

with

$$s^2 = \frac{1}{n-k}(Y - X\hat{\beta})^T(Y - X\hat{\beta})$$

- marginal posterior of $\beta|X, y$, found by *averaging over* σ, is multivariate t with $(n-k)$ degrees of freedom (and some covariance matrix)
- in practice we use MCMC by drawing from σ and then drawing $\beta|\sigma$, so we don't really need to use $\beta|X, y$ explicitly

Is the posterior proper?

The joint posterior $p(\beta, \sigma^2 | X, y)$ is proper provided that

1. $n > k$
2. $\text{rank}(X) = k$

Posterior Predictive Density

Common use of regression modeling is prediction of future observation $\tilde{y}$ corresponding to a covariate vector x^*.

- from above we know that $\tilde{y}$, conditional on the parameters, has distribution

$$\tilde{y}|\beta, \sigma^2, x^* \sim \mathcal{N}(x^*\beta, \sigma)$$

- posterior predictive density of $\tilde{y}$, denoted $p(\tilde{y}|y)$, represented as a *mixture* of these sampling densities $p(\tilde{y}|\beta, \sigma^2)$, averaged over the posterior of β, σ^2:

$$p(\tilde{y}|y) = \int p(\tilde{y}|\beta, \sigma^2) p(\beta, \sigma^2|y) d\beta d\sigma^2$$

Sampling from the posterior

To sample from $p(\beta, \sigma^2 | X, y)$,

1. Compute $\hat{\beta}$ and $(X^T X)^{-1}$.
2. Compute s^2.
3. Draw σ^2 from scaled inverse-χ^2 distribution.
4. Draw β from multivariate normal distribution above.

Gibbs Sampling

If we know the full conditional distributions $p(\beta|\sigma^2, X, Y)$ and $p(\sigma^2|\beta, X, Y)$, we can sample from the joint posterior $p(\beta, \sigma^2|X, Y)$ using the Gibbs sampler:

Initialize $\beta_{(1)}$, $\sigma^2_{(1)}$

For $t = 1 : T$

$$\beta_{(t+1)} \sim p(\beta_{(t)}|\sigma^2_{(t)}, X, Y)$$
$$\sigma_{(t+1)} \sim p(\sigma^2_{(t)}|\beta_{(t+1)}, X, Y)$$

END

The `puffin` data from `LearnBayes` package

Measurements on breedings of the common puffin on different habits at Great Island, Newfoundland. A data frame with 38 observations on the following 5 variables.

`Nest`	nesting frequency (burrows per 9 square meters)
`Grass`	grass cover (percentage)
`Soil`	mean soil depth (in centimeters)
`Angle`	angle of slope (in degrees)
`Distance`	distance from cliff edge (in meters)

Frequentist Fit

```r
library(LearnBayes); library(MASS)
 fit <- lm(Nest ~ Grass + Soil + Angle
           + Distance, data = puffin)
```

```r
summary(fit)
```

```
Call:
lm(formula = Nest ~ Grass + Soil + Angle + Distance, data = puffin)

Residuals:
    Min      1Q  Median      3Q     Max
-4.0166 -2.1088  0.2293  1.2505  6.9881

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept) 10.117840   3.185028   3.177  0.00323 **
Grass       -0.007408   0.019459  -0.381  0.70586
Soil         0.209211   0.077238   2.709  0.01062 *
Angle        0.082389   0.077796   1.059  0.29727
Distance    -0.366571   0.057473  -6.378 3.18e-07 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 2.647 on 33 degrees of freedom
Multiple R-squared:  0.8792,    Adjusted R-squared:  0.8645
```

Minor detour (the **graph** package is not on CRAN)

```
source("http://bioconductor.org/biocLite.R")
biocLite(c("graph", "RBGL", "Rgraphviz"))
install.packages("gRain", dependencies=TRUE)
```

Bayesian Fit with MCMCpack

```
library(MCMCpack)
```

```
Warning: package 'MCMCpack' was built under R version 3.2.4
```

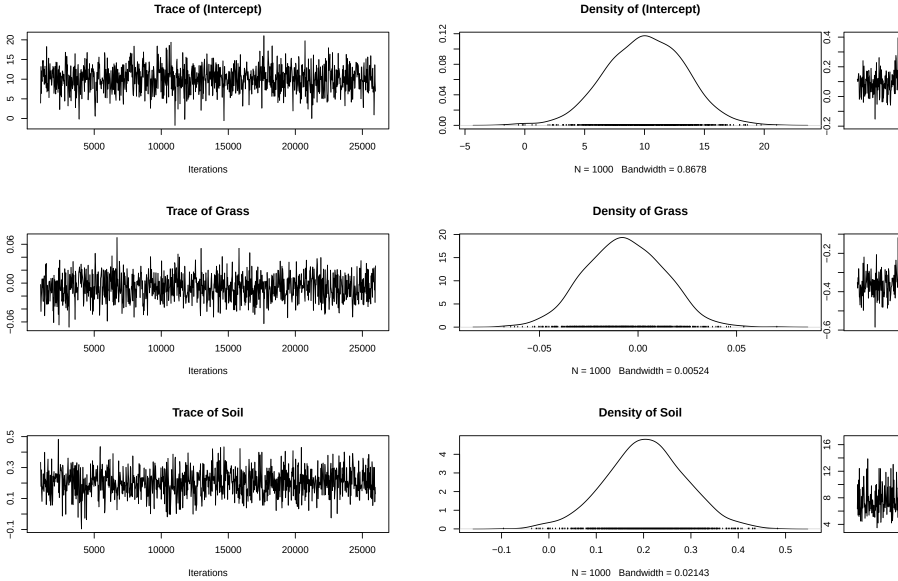
```
Bfit <- MCMCregress(Nest ~ Grass + Soil + Angle
          + Distance, data = puffin,
          burnin = 1000, mcmc = 25000, thin = 25)
```

```r
summary(Bfit)
```

```
Iterations = 1001:25976
Thinning interval = 25
Number of chains = 1
Sample size per chain = 1000

1. Empirical mean and standard deviation for each variable,
   plus standard error of the mean:

                 Mean      SD  Naive SE Time-series SE
(Intercept) 10.083975 3.25933 0.1030689      0.1081777
Grass       -0.006773 0.01968 0.0006224      0.0006627
Soil         0.204445 0.08408 0.0026587      0.0026587
Angle        0.087999 0.08374 0.0026481      0.0026481
Distance    -0.363486 0.06300 0.0019924      0.0021090
sigma2       7.409313 1.92603 0.0609063      0.0609063

2. Quantiles for each variable:

                2.5%      25%       50%       75%    97.5%
(Intercept)  3.86041  7.79296 10.094169 12.364227 16.37203
Grass       -0.04390 -0.02070 -0.007058  0.006583  0.03036
```

Posterior Density Plot

```r
plot(Bfit)
```



What `MCMCregress` does

Simulates from posterior using Gibbs sampling.

$$y_i = x_i'\beta + \epsilon_i, \;\; \epsilon_i \overset{iid}{\sim} \mathcal{N}(0, \sigma^2)$$

$$\beta \sim \mathcal{N}(b_0, B_0^{-1}), \;\; \sigma^{-2} \sim \text{Gamma}(\frac{c_0}{2}, \frac{d_0}{2})$$

and β, σ^{-2} assumed to be *a priori* independent

- b_0 prior mean for β; if scalar then all means the same
- default prior precision of β is $B0 = 0$, equivalent to improper uniform prior on β; if scalar then it is value times identity matrix
- $c_0/2$: shape parameter for inverse gamma prior on σ^2
- $d_0/2$: scale parameter for inverse gamma prior

More details

- multivariate normal draw for β
- inverse gamma draw for conditional error variance $\sigma^2|\beta$
- output is `mcmc` object (can use with `coda`)
- conditional error variance is the first block in the sampler, so only β is initialized

Example using `LearnBayes` package

Source: N.B. Al-Majed and M.R. Preston (2000). "Factors Influencing the Total Mercury and Methyl Mercury in the Hair of Fishermen in Kuwait," Environmental Pollution, Vol. 109, pp. 239-250 Description: Factors related to mercury levels among fishermen and a control group of non-fishermen.

Fisherman	indicator (fisherman)
Age	in years (age)
Residence	Time in years (restime)
Height	in cm (height)
Weight	in kg (weight)
Fish	meals per week ((fishmlwk)

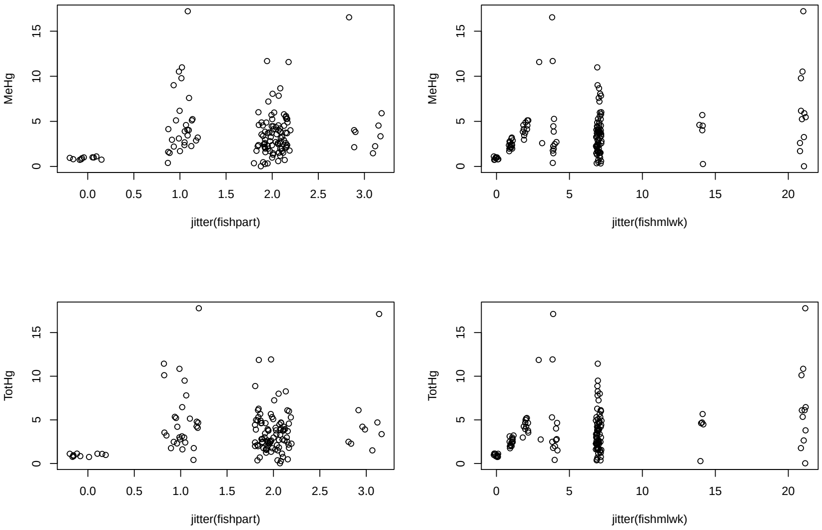
Parts of fish consumed: 0=none, 1=muscle tissue only, 2=mt and sometimes whole fish, 3=whole fish (fishpart)

Methyl	Mercury in mg/g (MeHg)
Total	Mercury in mg/g (TotHg)

```r
data <- read.csv(file="fishermen_mercury.csv")
attach(data)
head(data)
```

```
  fisherman age restime height weight fishmlwk fishpart   MeHg  TotHg
1         1  45       6    175     70       14        2  4.011  4.484
2         1  38      13    173     73        7        1  4.026  4.789
3         1  24       2    168     66        7        2  3.578  3.856
4         1  41       2    183     80        7        1 10.988 11.435
5         1  43      11    175     78       21        1 10.520 10.849
6         1  58       2    176     75       21        1  6.169  6.457
```

```r
par(mfrow = c(2, 2))
plot(jitter(fishpart), MeHg)
plot(jitter(fishmlwk), MeHg)
plot(jitter(fishpart), TotHg)
plot(jitter(fishmlwk), TotHg)
```



MeHg
0
5
10
15
0.0
0.5
1.0
1.5
2.0
2.5
3.0
jitter(fishpart)
0
5
10
15
20
jitter(fishmlwk)
TotHg

Linear model for total mercury against height, weight, age, restime, fishmlwk and fishpart

```
Call:
lm(formula = TotHg ~ age + height + weight + restime + fishmlwk +
    fishpart, x = TRUE, y = TRUE)

Residuals:
    Min      1Q  Median      3Q     Max
-4.7487 -1.3973 -0.3171  0.7519 11.0089

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept) -17.92133    5.82703  -3.076  0.00257 **
age           0.04986    0.03456   1.443  0.15157
height        0.03701    0.03420   1.082  0.28118
weight        0.16790    0.03511   4.782 4.69e-06 ***
restime      -0.05791    0.05324  -1.088  0.27874
fishmlwk      0.14380    0.04407   3.263  0.00141 **
fishpart      0.35247    0.32244   1.093  0.27638
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 2.558 on 128 degrees of freedom
Multiple R-squared:  0.2761,    Adjusted R-squared:  0.2422
F-statistic: 8.138 on 6 and 128 DF,  p-value: 1.848e-07
```

Using the `blinreg` function in `LearnBayes`

```r
theta.sample = blinreg(fit$y, fit$x, 5000)
```

Samples from joint posterior, taking the response and design matrix defined in the linear model fit above.

Posterior Summaries

```r
apply(theta.sample$beta, 2, quantile, c(0.025, 0.5, 0.975))
```

```
      X(Intercept)        Xage     Xheight    Xweight    Xrestime
2.5%    -29.600940 -0.01720058 -0.03139177 0.09859833 -0.16622606
50%     -17.886906  0.05036845  0.03668394 0.16746942 -0.05804612
97.5%    -6.308901  0.12037359  0.10586953 0.23673457  0.04725259
      Xfishmlwk  Xfishpart
2.5%  0.0564391 -0.2776243
50%   0.1436992  0.3575589
97.5% 0.2287748  0.9904738
```

```r
quantile(theta.sample$sigma, c(0.025, 0.5, 0.975))
```

```
    2.5%      50%    97.5%
2.279657 2.562344 2.925803
```