

Statistical Modeling 2

Exercise 1

February 1, 2017

1 Bayesian inference in simple conjugate families

A

$$\begin{aligned}
 p(w \mid x_1, \dots, x_N) &\propto p(x_1, \dots, x_N \mid w)p(w) && \text{(Bayes rule)} \\
 &\propto \prod_{i=1}^N p(x_i \mid w) w^{a-1} (1-w)^{b-1} && \text{(independence)} \\
 &\propto w^s (1-w)^{N-s} w^{a-1} (1-w)^{b-1} && \text{(let } s = \sum_{i=1}^N x_i) \\
 &= w^{s+a-1} (1-w)^{N-s+b-1} \\
 &\propto \text{Beta}(s+a, N-s+b)
 \end{aligned}$$

B

Let $f(x_1, x_2) = (y_1, y_2) = (x_1/(x_1 + x_2), x_1 + x_2)$, we have:

$$f^{-1}(y_1, y_2) = (x_1, x_2) = (y_1 y_2, y_2 - y_1 y_2)$$

We then calculate the Jacobian of f^{-1} :

$$\begin{aligned}
 \partial x_1 / \partial y_1 &= y_2 \\
 \partial x_1 / \partial y_2 &= y_1 \\
 \partial x_2 / \partial y_1 &= -y_2 \\
 \partial x_2 / \partial y_2 &= 1 - y_1
 \end{aligned}$$

Therefore,

$$\begin{aligned}
 |J(f^{-1})| &= \begin{vmatrix} y_2 & y_1 \\ -y_2 & 1 - y_1 \end{vmatrix} \\
 &= y_2(1 - y_1) + y_1 y_2 \\
 &= y_2
 \end{aligned}$$

Let p_X be the joint density of (x_1, x_2) . We have the joint density of y_1 and y_2 :

$$\begin{aligned}
p(y_1, y_2) &= p_X(f^{-1}(y_1, y_2)) |J(f^{-1}(y_1, y_2))| \\
&= \text{Ga}(y_1 y_2; a_1, 1) \text{Ga}(y_2 - y_1 y_2; a_2, 1) y_2 \\
&= \frac{(y_1 y_2)^{a_1-1} \exp(-y_1 y_2)}{\Gamma(a_1)} \frac{((1 - y_1) y_2)^{a_2-1} \exp(y_1 y_2 - y_2)}{\Gamma(a_2)} y_2 \\
&= \frac{y_1^{a_1-1} y_2^{a_1+a_2-1} (1 - y_1)^{a_2-1} \exp(-y_2)}{\Gamma(a_1) \Gamma(a_2)}
\end{aligned}$$

The marginals are:

$$\begin{aligned}
p(y_1) &= \int_0^\infty p(y_1, y_2) dy_2 \\
&= \frac{y_1^{a_1-1} (1 - y_1)^{a_2-1}}{\Gamma(a_1) \Gamma(a_2)} \int_0^\infty y_2^{a_1+a_2-1} \exp(-y_2) dy_2 \\
&= \frac{y_1^{a_1-1} (1 - y_1)^{a_2-1} \Gamma(a_1 + a_2)}{\Gamma(a_1) \Gamma(a_2)}
\end{aligned}$$

and

$$\begin{aligned}
p(y_2) &= \int_0^\infty p(y_1, y_2) dy_1 \\
&= \frac{y_2^{a_1+a_2-1} \exp(-y_2)}{\Gamma(a_1) \Gamma(a_2)} \int_0^\infty y_1^{a_1-1} (1 - y_1)^{a_2-1} dy_1 \\
&= \frac{y_2^{a_1+a_2-1} \exp(-y_2)}{\Gamma(a_1) \Gamma(a_2)} \text{Beta}(a_1, a_2)
\end{aligned}$$

We can simulate a Beta random variable by taking two Gamma random variable as x_1 and x_2 and evaluate y_1 .

C

The posterior is:

$$\begin{aligned}
p(\theta|x_1, \dots, x_N) &\propto p(x_1, \dots, x_N|\theta)p(\theta) \\
&= \prod_{i=1}^N N(x_i; \theta, \sigma^2) N(\theta; m, v) \\
&\propto \prod_{i=1}^N \exp\left(-\frac{(x_i - \theta)^2}{2\sigma^2}\right) \exp\left(-\frac{(\theta - m)^2}{2v}\right) \\
&= \prod_{i=1}^N \exp\left(-\frac{x_i^2 - 2x_i\theta + \theta^2}{2\sigma^2}\right) \exp\left(-\frac{\theta^2 - 2\theta m + m^2}{2v}\right) \\
&= \exp\left(-\frac{\sum_i x_i^2 + 2\sum_i x_i\theta - N\theta^2}{2\sigma^2}\right) \exp\left(-\frac{-\theta^2 + 2\theta m - m^2}{2v}\right) \\
&= \exp\left(-\theta^2\left(\frac{N}{2\sigma^2} + \frac{1}{2v}\right) + \theta\left(\frac{\sum_i x_i}{\sigma^2} + \frac{m}{v}\right) - \frac{\sum_i x_i^2}{2\sigma^2} - \frac{m^2}{2v}\right)
\end{aligned}$$

We then complete the square by setting the posterior to:

$$\begin{aligned}
&= \exp\left[-a(\theta^2 - 2b\theta + b^2)\right] \\
&= \exp\left[-a(\theta - b)^2\right] \\
&= \exp\left[-\frac{(\theta - b)^2}{2(1/(2a))}\right]
\end{aligned}$$

We calculate a, b by matching coefficients:

$$\begin{aligned}
a &= \frac{N}{2\sigma^2} + \frac{1}{2v} = \frac{Nv + \sigma^2}{2\sigma^2 v} \\
2ab &= \frac{\sum_i x_i}{\sigma^2} + \frac{m}{v} \\
\implies b &= \frac{v \sum_i x_i + m\sigma^2}{v\sigma^2} \frac{\sigma^2 v}{Nv + \sigma^2} \\
&= \frac{v \sum_i x_i + m\sigma^2}{Nv + \sigma^2}
\end{aligned}$$

The posterior is then:

$$\begin{aligned}
&N(b, 1/(2a)) \\
&= N\left(\frac{v \sum_i x_i + m\sigma^2}{Nv + \sigma^2}, \frac{\sigma^2 v}{Nv + \sigma^2}\right)
\end{aligned}$$

D

$$\begin{aligned}
p(\omega \mid x_1, \dots, x_N) &\propto \prod_{i=1}^N p(x_i \mid \theta, \omega) p(\omega) \\
&\propto \prod_{i=1}^N \omega^{1/2} \exp \left[-\frac{\omega}{2} (x_i - \theta)^2 \right] \frac{b^a}{\Gamma(a)} \omega^{a-1} \exp(-b\omega) \\
&\propto \omega^{N/2+a-1} \exp \left[-\omega \left(b + \frac{\sum_i (x_i - \theta)^2}{2} \right) \right] \\
&\propto \text{Ga} \left(a + \frac{N}{2}, b + \frac{\sum_i (x_i - \theta)^2}{2} \right)
\end{aligned}$$

We have the posterior of the variance:

$$p(\sigma^2 \mid x_1, \dots, x_N) = \text{IG} \left(a + \frac{N}{2}, b + \frac{\sum_i (x_i - \theta)^2}{2} \right)$$

E

The posterior is:

$$\begin{aligned}
p(\theta \mid x_1, \dots, x_N) &\propto p(x_1, \dots, x_N \mid \theta) p(\theta) \\
&= \prod_{i=1}^N \text{N}(x_i; \theta, \sigma_i^2) \text{N}(\theta; m, v) \\
&\propto \prod_{i=1}^N \exp \left(-\frac{(x_i - \theta)^2}{2\sigma_i^2} \right) \exp \left(-\frac{(\theta - m)^2}{2v} \right) \\
&= \exp \left(-\sum_{i=1}^n \frac{(x_i - \theta)^2}{2\sigma_i^2} - \frac{(\theta - m)^2}{2v} \right) \\
&= \exp \left[-\frac{1}{2} \left(\sum_{i=1}^n \frac{(x_i - \theta)^2}{\sigma_i^2} + \frac{(\theta - m)^2}{v} \right) \right] \\
&= \exp \left[-\frac{1}{2} \left(\sum_{i=1}^n \frac{x_i^2}{\sigma_i^2} + \sum_{i=1}^n \frac{-2x_i\theta}{\sigma_i^2} + \sum_{i=1}^n \frac{\theta^2}{\sigma_i^2} + \frac{\theta^2 - 2\theta m + m^2}{v} \right) \right] \\
&= \exp \left\{ -\frac{1}{2} \left[\theta^2 \left(\sum_{i=1}^n \frac{1}{\sigma_i^2} + \frac{1}{v} \right) - 2\theta \left(\sum_{i=1}^n \frac{x_i}{\sigma_i^2} + \frac{m}{v} \right) + \sum_{i=1}^n \frac{x_i^2}{\sigma_i^2} + \frac{m^2}{v} \right] \right\} \\
\text{set } &= \exp \left\{ -\frac{1}{2} [a(\theta^2 - 2\theta b + b^2)] \right\} \\
&= \exp \left\{ -\frac{1}{2} \left[\frac{(\theta - b)^2}{1/a} \right] \right\}
\end{aligned}$$

Matching the coefficients, we have:

$$a = \sum_{i=1}^n \frac{1}{\sigma_i^2} + \frac{1}{v}$$

$$b = \left(\sum_{i=1}^N \frac{x_i}{\sigma_i^2} + \frac{m}{v} \right) / \left(\sum_{i=1}^n \frac{1}{\sigma_i^2} + \frac{1}{v} \right)$$

The posterior is:

$$N(b, 1/a)$$

F

$$\begin{aligned} p(x) &= \int_0^\infty p(x \mid \sigma^2) p(\sigma^2) d\sigma^2 \\ &= \int_0^\infty p(x \mid \omega) p(\omega) d\omega \\ &= \int_0^\infty \left(\frac{\omega}{2\pi} \right)^{1/2} \exp\left(-\frac{\omega}{2} x^2\right) \frac{b^a}{\Gamma(a)} \omega^{a-1} \exp(-b\omega) d\omega \\ &= \frac{b^a}{(2\pi)^{1/2} \Gamma(a)} \int_0^\infty \omega^{1/2+a-1} \exp\left(-\omega \left(\frac{x^2}{2} + b\right)\right) d\omega \\ &= \frac{b^a}{(2\pi)^{1/2} \Gamma(a)} \frac{\Gamma(a+1/2)}{(b + x^2/2)^{a+1/2}} \quad (\text{Gamma integral}) \\ &= \frac{\Gamma(a+1/2)}{(2\pi b)^{1/2} \Gamma(a) (1 + \frac{x^2}{2b})^{a+1/2}} \end{aligned}$$

Let $\nu = 2a$ and $\lambda = a/b$, we have:

$$p(x) = \frac{\Gamma(\frac{\nu+1}{2})}{\Gamma(\nu/2)} \left(\frac{\lambda}{\pi\nu} \right)^{1/2} \left(1 + \frac{\lambda x^2}{2} \right)^{-\frac{\nu+1}{2}}$$

This is the Student t distribution with ν degree of freedom and ‘precision’ λ .

2 The multivariate normal distribution

A

$$\begin{aligned} \text{cov}(x) &= E\{(x - \mu)(x - \mu)^T\} \\ &= E\{xx^T - x\mu^T - \mu x^T + \mu\mu^T\} \\ &= E(xx^T) - E(x)\mu^T - \mu E(x)^T + \mu\mu^T \\ &= E(xx^T) - \mu\mu^T \end{aligned}$$

We have:

$$E(Ax + b) = AE(x) + b = A\mu + b$$

then

$$\begin{aligned} \text{cov}(Ax + b) &= E\{[(Ax + b) - (A\mu + b)][(Ax + b) - (A\mu + b)]^T\} \\ &= E\{(Ax - A\mu)(Ax - A\mu)^T\} \\ &= E\{A(x - \mu)(x - \mu)^T A^T\} \\ &= AE\{(x - \mu)(x - \mu)^T\}A^T \\ &= A\text{cov}(x)A^T \end{aligned}$$

B

$$\begin{aligned} p(z) &= \prod_{i=1}^p p(z_i) \\ &= \frac{1}{(\sqrt{2\pi})^p} \exp\left(-\sum_{i=1}^p \frac{z_i^2}{2}\right) \\ &= \frac{1}{(\sqrt{2\pi})^p} \exp\left(-\frac{z^T z}{2}\right) \end{aligned}$$

The MGF of z is:

$$\begin{aligned} E(\exp(t^T z)) &= E\left[\exp\left(\sum_{i=1}^p t_i z_i\right)\right] \\ &= E\left[\prod_{i=1}^p \exp(t_i z_i)\right] \\ &= \prod_{i=1}^p E[\exp(t_i z_i)] \\ &= \prod_{i=1}^p \exp(t_i^2/2) \\ &= \exp\left[\sum_{i=1}^p t_i^2/2\right] \\ &= \exp(t^T t/2) \end{aligned}$$

C

We need to prove that for all vector a not identically zero, the scalar quantity $z = a^T x$ is normally distributed if and only if

$$E[\exp(t^T x)] = \exp(t^T \mu + t^T \Sigma t/2)$$

(only if) We have that $z = a^T x$ is normally distributed:

$$\text{MGF}_z(s) = E[\exp(sa^T x)] = \exp(ms + vs^2/2)$$

Consider:

D

We have $z \sim N(0, I)$ and $x = Lz + \mu$.

The MGF of x is:

$$E(\exp(t^T x)) = E[\exp(t^T Lz + t^T \mu)]$$

The expectation is with respect to z , $t^T \mu$ is a constant, we then look at:

$$\begin{aligned} E[\exp(t^T Lz)] &= E \left[\exp \left(\sum_{i=1}^p \sum_{j=1}^p t_i L_{ij} z_j \right) \right] \\ &= E \left[\prod_{j=1}^p \exp \left(\sum_{i=1}^p t_i L_{ij} z_j \right) \right] \\ &= \prod_{j=1}^p E \left[\exp \left(\sum_{i=1}^p t_i L_{ij} z_j \right) \right] \quad (\text{independence}) \\ &= \prod_{j=1}^p \text{MGF}_{z_j} \left(\sum_{i=1}^p t_i L_{ij} \right) \\ &= \prod_{j=1}^p \exp \left(\frac{1}{2} (t^T L_j)^2 \right) \\ &= \prod_{j=1}^p \exp \left(\frac{1}{2} t^T L_j L_j^T t \right) \\ &= \exp \left(\frac{1}{2} \sum_{j=1}^p t^T L_j L_j^T t \right) \\ &= \exp \left(\frac{1}{2} t^T L L^T t \right) \end{aligned}$$

Come back to the MGF of x :

$$E(\exp(t^T x)) = \exp \left(t^T \mu + \frac{1}{2} t^T L L^T t \right)$$

Therefore, $x \sim N(\mu, L L^T)$.

E

We have that x has a multivariate normal distribution: $x \sim N(\mu, \Sigma)$. The covariance matrix Σ is symmetric positive definite and has a Cholesky decomposition:

$$\Sigma = LL^T$$

where L is a lower triangular matrix with positive diagonal entries and therefore invertible. Let

$$z = L^{-1}(x - \mu)$$

Consider the MGF of z :

$$\begin{aligned} E[\exp(t^T z)] &= E[\exp(t^T L^{-1}(x - \mu))] \\ &= E[\exp(t^T L^{-1}x) \cdot \exp(-t^T L^{-1}\mu)] \\ &= \text{MGF}_x(t^T L^{-1}) \cdot \exp(-t^T L^{-1}\mu) \\ &= \exp(t^T L^{-1}\mu + t^T L^{-1}\Sigma L^{-T}t/2) \cdot \exp(-t^T L^{-1}\mu) \\ &= \exp(t^T L^{-1}\Sigma L^{-T}t/2) \\ &= \exp(t^T L^{-1}(LL^T)L^{-T}t/2) \\ &= \exp(t^T t/2) \end{aligned}$$

We conclude that z has standard multivariate normal distribution and that x can be written as an affine transformation of standard normal distribution.

F

Let z be standard multivariate Normal:

$$p_Z(z) \propto \exp\left(-\frac{z^T z}{2}\right)$$

By the previous result, we have that $x = Lz + \mu$ has multivariate Normal distribution. Since L is full rank, it is invertible, let $z = f(x) = L^{-1}(x - \mu)$

The pdf of x is:

$$\begin{aligned} p_X(x) &= p_Z(f(x))|J_f(x)| \\ &\propto \exp\left(-\frac{(x - \mu)^T L^{-T} L^{-1}(x - \mu)}{2}\right) |L^{-1}| \\ &\propto \exp(-Q(x - \mu)/2) \end{aligned}$$

G

By the previous results, x_1 and x_2 are affine transformation of independent standard Normal distribution. Let $z \sim N(0, I)$

$$\begin{aligned}x_1 &= L_1 z + \mu_1 \\x_2 &= L_2 z + \mu_2\end{aligned}$$

We have:

$$\begin{aligned}y &= Ax_1 + Bx_2 = AL_1 z + A\mu_1 + BL_2 z + B\mu_2 \\&= (AL_1 + BL_2)z + A\mu_1 + B\mu_2\end{aligned}$$

We see that y is an affine transformation of independent standard Normal variables and therefore is multivariate Normal with mean $A\mu_1 + B\mu_2$ and variance

$$\begin{aligned}(AL_1 + BL_2)(AL_1 + BL_2)^T &= AL_1 L_1^T A^T + AL_1 L_2^T B^T + BL_2 L_1^T A^T + BL_2 L_2^T B^T \\&= A\Sigma_1 A^T + AL_1 L_2^T B^T + BL_2 L_1^T A^T + B\Sigma_2 B^T\end{aligned}$$

3 Conditionals and marginals

A

Decompose the covariance matrix Σ and partition L into L_1 and L_2 where L_1 has k elements and corresponds to x_1 .

$$\begin{aligned}\Sigma &= LL^T \\&= \begin{pmatrix} L_1 \\ L_2 \end{pmatrix} \begin{pmatrix} L_1^T & L_2^T \end{pmatrix} \\&= \begin{pmatrix} L_1 L_1^T & L_1 L_2^T \\ L_2 L_1^T & L_2 L_2^T \end{pmatrix}\end{aligned}$$

We have that $\Sigma_{11} = L_1 L_1^T$. By the previous results, $x = Lz + \mu$ where z is a vector of independent standard Normal variables. Take the first k element, we have:

$$x_1 = L_1 z_1 + \mu_1$$

where z_1 is also a vector of independent standard Normal variables. Therefore, x_1 also has multivariate Normal distribution with mean μ_1 and variance $L_1 L_1^T = \Sigma_{11}$.

B

$$\begin{aligned} \begin{pmatrix} \Sigma_{11} & \Sigma_{12} \\ \Sigma_{21} & \Sigma_{22} \end{pmatrix}^{-1} &= \begin{pmatrix} (\Sigma_{11} - \Sigma_{12}\Sigma_{22}^{-1}\Sigma_{21})^{-1} & -\Sigma_{11}^{-1}\Sigma_{12}(\Sigma_{22} - \Sigma_{21}\Sigma_{11}^{-1}\Sigma_{12})^{-1} \\ -\Sigma_{22}^{-1}\Sigma_{21}(\Sigma_{11} - \Sigma_{12}\Sigma_{22}^{-1}\Sigma_{21})^{-1} & (\Sigma_{22} - \Sigma_{21}\Sigma_{11}^{-1}\Sigma_{12})^{-1} \end{pmatrix} \\ &= \begin{pmatrix} \Omega_{11} & \Omega_{12} \\ \Omega_{12}^T & \Omega_{22} \end{pmatrix} \end{aligned}$$

C

$$\begin{aligned} \log p(x_1|x_2) &= \log p(x_1, x_2) - \log(x_2) \\ &= \text{const} - \frac{(x - \mu)^T \Sigma^{-1} (x - \mu)}{2} \\ &= \text{const} - \frac{1}{2} \left\{ \begin{pmatrix} x_1 - \mu_1 \\ x_2 - \mu_2 \end{pmatrix}^T \begin{pmatrix} \Omega_{11} & \Omega_{12} \\ \Omega_{12}^T & \Omega_{22} \end{pmatrix} \begin{pmatrix} x_1 - \mu_1 \\ x_2 - \mu_2 \end{pmatrix} \right\} \\ &= \text{const} - \frac{1}{2} \{ (x_1 - \mu_1)^T \Omega_{11} (x_1 - \mu_1) + (x_2 - \mu_2)^T \Omega_{12}^T (x_1 - \mu_1) \\ &\quad + (x_1 - \mu_1)^T \Omega_{12} (x_2 - \mu_2) + (x_2 - \mu_2)^T \Omega_{22} (x_2 - \mu_2) \} \\ &= \text{const} - \frac{1}{2} \{ (x_1 - \mu_1)^T \Omega_{11} (x_1 - \mu_1) + 2(x_1 - \mu_1)^T \Omega_{12} (x_2 - \mu_2) \} \\ &= \text{const} - \frac{1}{2} \{ x_1^T \Omega_{11} x_1 - x_1^T \Omega_{11} \mu_1 - \mu_1^T \Omega_{11} x_1 + 2x_1^T \Omega_{12} x_2 - 2x_1^T \Omega_{12} \mu_2 \} \\ &= \text{const} - \frac{1}{2} \{ x_1^T \Omega_{11} x_1 + x_1^T (-\Omega_{11} \mu_1 - \Omega_{11}^T \mu_1 + 2\Omega_{12} x_2 - 2\Omega_{12} \mu_2) \} \\ &= \text{const} - \frac{1}{2} \{ x_1^T \Omega_{11} x_1 - 2x_1^T \Omega_{11} (\mu_1 - \Omega_{11}^{-1} \Omega_{12} x_2 + \Omega_{11}^{-1} \Omega_{12} \mu_2) \} \\ &= \text{const} - \frac{1}{2} \{ (x_1 - \mu_{1|2})^T \Omega_{11} (x_1 - \mu_{1|2}) \} \end{aligned}$$

where:

$$\mu_{1|2} = \mu_1 - \Omega_{11}^{-1} \Omega_{12} x_2 + \Omega_{11}^{-1} \Omega_{12} \mu_2$$

We also have:

$$\begin{aligned} \Omega_{11}^{-1} \Omega_{12} &= (\Sigma_{11} - \Sigma_{12}\Sigma_{22}^{-1}\Sigma_{12}^T)(-\Sigma_{11}^{-1})\Sigma_{12}(\Sigma_{22} - \Sigma_{12}^T\Sigma_{11}^{-1}\Sigma_{12})^{-1} \\ &= (-\Sigma_{12} + \Sigma_{12}\Sigma_{22}^{-1}\Sigma_{12}^T\Sigma_{11}^{-1}\Sigma_{12})(\Sigma_{22} - \Sigma_{12}^T\Sigma_{11}^{-1}\Sigma_{12})^{-1} \\ &= (-\Sigma_{12}\Sigma_{22}^{-1}\Sigma_{22} + \Sigma_{12}\Sigma_{22}^{-1}\Sigma_{12}^T\Sigma_{11}^{-1}\Sigma_{12})(\Sigma_{22} - \Sigma_{12}^T\Sigma_{11}^{-1}\Sigma_{12})^{-1} \\ &= -\Sigma_{12}\Sigma_{22}^{-1}(\Sigma_{22} - \Sigma_{12}^T\Sigma_{11}^{-1}\Sigma_{12})(\Sigma_{22} - \Sigma_{12}^T\Sigma_{11}^{-1}\Sigma_{12})^{-1} \\ &= -\Sigma_{12}\Sigma_{22}^{-1} \end{aligned}$$

Therefore,

$$\mu_{1|2} = \mu_1 + \Sigma_{12}\Sigma_{22}^{-1}(x_2 - \mu_2)$$

We conclude that $p(x_1|x_2)$ has Normal distribution with mean $\mu_{1|2}$ given above and variance $\Omega_{11}^{-1} = \Sigma_{11} - \Sigma_{12}\Sigma_{22}^{-1}\Sigma_{12}^T$

4 Multiple regression: three classical principles for inference

4.1 A

In the least square estimate, we minimize:

$$\begin{aligned} L(\beta) &= \sum_{i=1}^n (y_i - x_i^T \beta)^2 \\ &= (y - X\beta)^T (y - X\beta) \\ &= -y^T y - y^T X\beta - \beta^T X^T y + \beta^T X^T X\beta \end{aligned}$$

where we define $y = (y_1, \dots, y_n)^T$ as a $n \times 1$ vector and $X = (x_1^T; \dots; x_n^T)$ as a $n \times p$ matrix. We take the derivative of L and set to 0:

$$\begin{aligned} \frac{\partial L}{\partial \beta} &= -X^T y - X^T y + X^T X\beta + X^T X\beta \\ &= 2X^T (X\beta - y) \\ \text{set } &= 0 \\ \implies X^T X\beta &= X^T y \\ \beta &= (X^T X)^{-1} X^T y \end{aligned}$$

In the maximum likelihood estimate, we maximize:

$$\begin{aligned} L(\beta) &= \prod_{i=1}^n p(y_i | \beta, \sigma^2) \\ &= N(y | X\beta, \sigma^2 I) \\ &\propto \exp \left(-\frac{1}{2} (y - X\beta)^T \frac{1}{\sigma^2} I (y - X\beta) \right) \end{aligned}$$

which is equivalent to minimize:

$$\begin{aligned} &(y - X\beta)^T \frac{1}{\sigma^2} I (y - X\beta) \\ &\propto (y - X\beta)^T (y - X\beta) \end{aligned}$$

which is the same as the least square objective function.

In the method of moment estimate, we set:

$$\text{cov}(y - X\beta, X_j) = 0$$

where X_j is the column j of X for $j = 1, \dots, p$. We have:

$$\begin{aligned} \text{cov}(y - X\beta, X_j) &= 0 \quad \forall j \\ \iff (y - X\beta)^T X_j &= 0 \quad \forall j \\ \iff (y - X\beta)^T X &= 0 \\ \iff X^T(X\beta - y) &= 0 \end{aligned}$$

This is the same as the equation that we solve in least square.

B

The weighted sum of squared error is:

$$L(\beta) = \sum_{i=1}^n w_i (y_i - x_i^T \beta)^2$$

We first take the derivative with respect to β :

$$\frac{\partial L}{\partial \beta} = \sum_{i=1}^n w_i (2)(y_i - x_i^T \beta)(-x_i)$$

Setting the derivative to 0, we have:

$$\begin{aligned} \sum_{i=1}^n w_i x_i (y_i - x_i^T \beta) &= 0 \\ \iff \sum_{i=1}^n w_i x_i y_i - \sum_{i=1}^n w_i x_i x_i^T \beta &= 0 \\ \iff \sum_{i=1}^n w_i x_{ik} y_i - \sum_{i=1}^n w_i x_{ik} x_i^T \beta_k &= 0 \quad \forall k = 1, \dots, p \\ \iff \beta_k &= \frac{\sum_{i=1}^n w_i x_{ik} y_i}{\sum_{i=1}^n w_i x_{ik} x_i^T \beta_k} \quad \forall k = 1, \dots, p \end{aligned}$$

The maximum likelihood under heteroskedastic Gaussian error is:

$$\begin{aligned} L(\beta) &= \prod_{i=1}^n p(y_i \mid \beta, \sigma_i^2) \\ &= N(y \mid X\beta, C) \end{aligned}$$

where the covariance matrix is C is a diagonal matrix such that $C_{ii} = \sigma_i^2$. We have that $C^{-1} = D$ is a diagonal matrix such that $D_{ii} = 1/\sigma_i^2$. The maximum likelihood is then:

$$L(\beta) = N(y \mid X\beta, C) \propto \exp\left(-\frac{1}{2}(y - X\beta)^T D(y - X\beta)\right)$$

Maximizing the that likelihood is the same as minimizing:

$$\begin{aligned}(y - X\beta)^T D(y - X\beta) &= \sum_{i=1}^n (y_i - x_i^T \beta) \frac{1}{\sigma_i^2} (y_i - x_i^T \beta) \\ &= \sum_{i=1}^n w_i (y_i - x_i^T \beta)^2\end{aligned}$$

where $w_i = \frac{1}{\sigma_i^2}$.

This is the same objective as the weighted least square.

5 Quantifying uncertainty: some basic frequentist ideas

In linear regression

A

The estimate is:

$$\hat{\beta} = (X^T X)^{-1} X^T y$$

which is a linear combination of y . Since y has Normal distribution, $\hat{\beta}$ also has Normal distribution.

$$\begin{aligned}E(\hat{\beta}) &= (X^T X)^{-1} X^T E(y) \\ &= (X^T X)^{-1} X^T X \beta \\ &= \beta\end{aligned}$$

$$\begin{aligned}\text{cov}(\hat{\beta}) &= (X^T X)^{-1} X^T \text{cov}(y) X (X^T X)^{-T} \\ &= (X^T X)^{-1} X^T (\sigma^2 I) X (X^T X)^{-T} \\ &= \sigma^2 (X^T X)^{-1} X^T X (X^T X)^{-T} \\ &= \sigma^2 (X^T X)^{-T}\end{aligned}$$

B

We can estimate σ^2 by:

$$\hat{\sigma}^2 = \frac{\sum_{i=1}^n \hat{\epsilon}_i^2}{n-p} = \frac{\sum_{i=1}^n (x_i \hat{\beta} - y_i)^2}{n-p} = \frac{(X \hat{\beta} - y)^T (X \hat{\beta} - y)}{n-p}$$

and plug that into the equation for $\text{cov}(\hat{\beta})$. This estimate produces the same result as the one from R linear model.

```
# Load the library
# you might have to install this the first time
library(mlbench)

# Load the data
ozone = data(Ozone, package='mlbench')

# Look at the help file for details
?Ozone

# Scrub the missing values
# Extract the relevant columns
ozone = na.omit(Ozone)[,4:13]

y = ozone[,1]
x = as.matrix(ozone[,2:10])

# add an intercept
x = cbind(1,x)

# compute the estimator
betahat = solve(t(x) %*% x) %*% t(x) %*% y

# Fill in the blank
sshat = t(x %*% betahat - y) %*% (x %*% betahat - y)
sshat = sshat[1,1] / (dim(x)[1] - dim(x)[2])
betacov = sshat * t(solve(t(x) %*% x) )

# Now compare to lm
# the 'minus 1' notation says not to fit an intercept (we've already hard-coded
lm1 = lm(y~x-1)

summary(lm1)
betacovlm = vcov(lm1)
print("cov_of_Beta_from_LM")
print(sqrt(diag(betacovlm)))
print("cov_of_Beta_from_my_calculation")
```

```
print(sqrt(diag(betacov)))
```

OUTPUT:

```
[1] "cov of Beta from LM"
      x      xV5      xV6      xV7      xV8
3.832869e+01 7.251104e-03 1.741433e-01 2.376936e-02 6.929942e-02
      xV9      xV10      xV11      xV12      xV13
1.247136e-01 3.943697e-04 1.477719e-02 1.192917e-01 4.896254e-03
```

```
[1] "cov of Beta from my calculation"
      V5      V6      V7      V8
3.832869e+01 7.251104e-03 1.741433e-01 2.376936e-02 6.929942e-02
      V9      V10      V11      V12      V13
1.247136e-01 3.943697e-04 1.477719e-02 1.192917e-01 4.896254e-03
```

Propagating uncertainty

A

$$\begin{aligned}\text{var}(f) &= \text{var}(\theta_1 + \theta_2) \\ &= \text{var}(\theta_1) + \text{var}(\theta_2) + 2\text{cov}(\theta_1, \theta_2) \\ &= \hat{\Sigma}_{11} + \hat{\Sigma}_{22} + 2\hat{\Sigma}_{12}\end{aligned}$$

In the general case, when f is the sum of p components of θ , we have:

$$\begin{aligned}\text{var}(f) &= \text{var}(\theta_1 + \dots + \theta_p) \\ &= \text{var}(\theta_1) + \dots + \text{var}(\theta_p) + \sum_{i \neq j} \text{cov}(\theta_i, \theta_j) \\ &= \sum_{i,j=1}^p \hat{\Sigma}_{ij}\end{aligned}$$

B

We first approximate f by a first order Taylor approximation:

$$\begin{aligned}f(\hat{\theta}) &\approx f(\theta) + f'(\theta)^T(\hat{\theta} - \theta) \\ &= f(\theta) + \sum_{i=1}^p f'(\theta_i)(\hat{\theta}_i - \theta_i)\end{aligned}$$

We then approximate the variance of f by the variance of the approximation:

$$\begin{aligned}\text{var}[f(\hat{\theta})] &\approx \text{var}[f(\theta) + \sum_{i=1}^p f'(\theta_i)(\hat{\theta}_i - \theta_i)] \\ &= \sum_{i=1}^p f'(\theta_i)^2 \text{var}(\hat{\theta}_i) + \sum_{i \neq j} f'(\theta_i) f'(\theta_j) \text{cov}(\hat{\theta}_i, \hat{\theta}_j) \\ &\approx \sum_{i=1}^p f'(\hat{\theta}_i)^2 \Sigma_{ii} + \sum_{i \neq j} f'(\hat{\theta}_i) f'(\hat{\theta}_j) \Sigma_{ij}\end{aligned}$$

To use this, the function f must be differentiable. For this to be a good approximation, f should be approximately linear around θ so that the Taylor approximation is close. We have also approximated $f'(\theta)$ by $f'(\hat{\theta})$, so we need $\hat{\theta}$ to be ‘close’ to θ . A caveat is that with two approximations, the errors may amplify.

The bootstrap

A

```
# Load the library
# you might have to install this the first time
nboots = 100000
library(mlbench)

# Load the data
ozone = data(Ozone, package='mlbench')

# Look at the help file for details
?Ozone

# Scrub the missing values
# Extract the relevant columns
ozone = na.omit(Ozone)[,4:13]

y = ozone[,1]
x = as.matrix(ozone[,2:10])

# add an intercept
x = cbind(1,x)

boot = matrix(0, ncol = 10, nrow = nboots)
for(i in 1:nboots)
{
```



```

idx = sample(1:nrow(x), nrow(x), replace=TRUE)
bx = x[idx,]
by = y[idx]
betahat = solve(t(bx) %*% bx) %*% t(bx) %*% by
boot[i, ] = betahat
}

bcov = cov(boot)
sqrt(diag(bcov))

# compute the estimator

# Fill in the blank
# betacov = ?

# Now compare to lm
# the 'minus 1' notation says not to fit an intercept (we've already hard-coded
lm1 = lm(y~x-1)

summary(lm1)
betacovlm = vcov(lm1)
sqrt(diag(betacovlm))

round(betacovlm, 3) # theory estimate
round(bcov, 3) # bootstrap estimate

> round(betacovlm, 3) # theory estimate
      x      xV5      xV6      xV7      xV8      xV9      xV10      xV11      xV12      xV13
x    1469.089 -0.277 -2.062 -0.153  0.359  1.591  0.001 -0.041  0.419 -0.003
xV5   -0.277  0.000  0.000  0.000  0.000  0.000  0.000  0.000  0.000  0.000
xV6   -2.062  0.000  0.030 -0.001 -0.001 -0.002  0.000  0.000  0.000  0.000
xV7   -0.153  0.000 -0.001  0.001  0.000  0.000  0.000  0.000  0.000  0.000
xV8    0.359  0.000 -0.001  0.000  0.005 -0.004  0.000 -0.001 -0.002  0.000
xV9    1.591  0.000 -0.002  0.000 -0.004  0.016  0.000  0.000 -0.009  0.000
xV10   0.001  0.000  0.000  0.000  0.000  0.000  0.000  0.000  0.000  0.000
xV11  -0.041  0.000  0.000  0.000 -0.001  0.000  0.000  0.000  0.001  0.000
xV12   0.419  0.000  0.000  0.000 -0.002 -0.009  0.000  0.001  0.014  0.000
xV13  -0.003  0.000  0.000  0.000  0.000  0.000  0.000  0.000  0.000  0.000

> round(bcov, 3) # bootstrap estimate
      [,1] [,2] [,3] [,4] [,5] [,6] [,7] [,8] [,9] [,10]
[1,] 1266.641 -0.24 -2.367 -0.034  0.453  1.526  0.001 -0.078  0.163 -0.003
[2,] -0.240  0.00  0.000  0.000  0.000  0.000  0.000  0.000  0.000  0.000
[3,] -2.367  0.00  0.026  0.000 -0.002 -0.003  0.000  0.000  0.000  0.000
[4,] -0.034  0.00  0.000  0.001  0.000 -0.001  0.000  0.000  0.000  0.000

```

[5,]	0.453	0.00	-0.002	0.000	0.004	-0.002	0.000	-0.001	-0.002	0.000
[6,]	1.526	0.00	-0.003	-0.001	-0.002	0.013	0.000	0.000	-0.009	0.000
[7,]	0.001	0.00	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
[8,]	-0.078	0.00	0.000	0.000	-0.001	0.000	0.000	0.000	0.000	0.000
[9,]	0.163	0.00	0.000	0.000	-0.002	-0.009	0.000	0.000	0.014	0.000
[10,]	-0.003	0.00	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000

B

```
library(MASS)

sim_mvn <- function(n, mu, Sigma){
  return (rmvnorm(n = n, mu = mu, Sigma = Sigma))
}

mle_mvn <- function(X){
  mu = colMeans(X)
  n = nrow(X)
  Sigma = cov(X)*(n-1)/n
  return(list(mu, Sigma))
}

boot_mle <- function(bn, X){
  m = ncol(X)
  boot_mu = matrix(0, ncol = m, nrow = bn)
  boot_cov = matrix(0, ncol = m*m, nrow = bn)
  for(i in 1:bn)
  {
    idx = sample(1:nrow(X), nrow(X), replace=TRUE)
    bx = X[idx,]
    temp = mle_mvn(bx)
    boot_mu[i,] = temp[[1]]
    boot_cov[i,] = temp[[2]]
  }
  return (list(boot_mu, boot_cov))
}

mu <- c(0,0)
Sigma <- matrix(c(10,3,3,2),2,2)

list_n = c(10, 100, 1000, 10000)
for(i in 1:4){
  n = list_n[i]
  X = sim_mvn(n, mu, Sigma)
```

```

temp = mle_mvn(X)
mu_hat = temp[[1]]
cov_hat = temp[[2]]
print(paste("N= ", n, " samples:"))
print("MLE of cov")
print(cov_hat)
temp = boot_mle(10000, X)
boot_cov = temp[[2]]
v = sqrt(diag(cov(boot_cov)))
print("bootstrap standard error")
print(matrix(v, nrow = 2, ncol = 2))
}

```

```

> Sigma # True covariance matrix
      [,1] [,2]
[1,]    10    3
[2,]     3    2

```

```

[1] "N= 10 samples:"
[1] "MLE of cov"
      [,1] [,2]
[1,] 11.744577 3.207579
[2,]  3.207579 1.962002
[1] "bootstrap standard error"
      [,1] [,2]
[1,]  4.292160 1.2707615
[2,]  1.270761 0.4323074

```

```

[1] "N= 100 samples:"
[1] "MLE of cov"
      [,1] [,2]
[1,] 11.023129 3.265027
[2,]  3.265027 2.153419
[1] "bootstrap standard error"
      [,1] [,2]
[1,]  1.4178240 0.5215858
[2,]  0.5215858 0.2647075

```

```

[1] "N= 1000 samples:"
[1] "MLE of cov"
      [,1] [,2]
[1,]  9.873546 2.873105
[2,]  2.873105 1.942224

```

```

[1] "bootstrap standard error"
      [,1]      [,2]
[1,] 0.4600641 0.17240442
[2,] 0.1724044 0.08604483

```

```

[1] "N= 10000 samples:"
[1] "MLE of cov"
      [,1]      [,2]
[1,] 10.087604 3.068838
[2,] 3.068838 2.040076
[1] "bootstrap standard error"
      [,1]      [,2]
[1,] 0.14409462 0.05525678
[2,] 0.05525678 0.02893835

```