Statistics 601 Assignment 2

Naiqing Cai

ncai5@wisc.edu

1. Let X1, … , Xn be an i.i.d sample from a uniform distribution U(0,θ), where θ > 0.

(a) Find the method-of-moments estimator of θ

$$E_\theta(X) = \int_0^\theta x\frac{1}{\theta}dx = \frac{\theta}{2} = \frac{1}{n}\sum_{i=1}^n X_i$$

The method − of − moments estimator of θ is : $\hat{\theta}_n = \frac{2}{n}\sum_{i=1}^n X_i$

(b) Find the MLE of θ

The p.d.f of each observation has the following form:

$$f(x|\theta) = \frac{1}{\theta}, 0 \leq x \leq \theta$$

otherwise, $f(x|\theta) = 0$

Therefore, the likelihood function has the form

$$L(\theta) = \frac{1}{\theta^n}, 0 \leq x_i \leq \theta(i = 1,2,...,n)$$

otherwise, $L(\theta) = 0$

It can be seen that the MLE of θ must be a value of θ for which θ ≥ xi for i = 1, … , n and which maximizes L(θ) among all such values.

Since L(θ) is a decreasing function of θ, the estimate will be the smallest possible value of θ such that θ ≥ xi for i = 1, … , n

MLE of θ is : $\hat{\theta} = \max(X_1,...,X_n)$

(c) Suppose we observe x1 = 3, x2 = 5, x3 = 6, and x4 = 18. Find the moment-of-moments estimate and MLE of θ. Which estimate is better?

The method − of − moments estimator of θ is : $\hat{\theta}_n = \frac{2}{n}\sum_{i=1}^n X_i = 16$

MLE of θ is : $\hat{\theta} = \max(X_1,...,X_n) = 18$

I think the two estimates both have pros and cons.

The moment-of-moments estimate is better because it is unbiased. But sometimes outside the valid range. The MLE is better for it is efficient and asymptotically unbiased but it can be highly biased for small samples.

2. An experimenter was interested in the relationship between temperature and heart rate in the common grass frog.
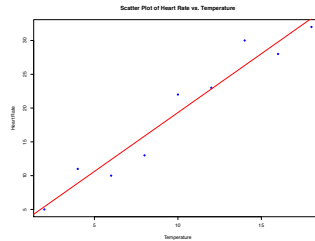
(a) Scatter Plot

figure 2.1 Scatter Plot of Heart Rate vs. Temperature

(b) Indicate the model underlying a simple linear regression of heart rate on temperature. Interpret the terms in the context of the study.

A simple linear regression model is:

$Y_i = \beta_0 + \beta_1 X_i + \varepsilon_i$, $i = 1,...,n$

Interpretation:

Yi represents the heart rate in the common grass frog, serves as a response variable.

Xi represents the temperature, serves as an explanatory variable.

ε_i is random error.

β0, β1 are fixed unknown regression coefficients

β0: intercept

β1: slope, interpreted as the change in the mean of Y (heart rate in the common grass frog) per unit increase in x (in the common grass frog).

(c) For the model in (b), what is the underlying population or the set of underlying populations?

The underlying population is the total temperature and heart rate in the common grass frog.

(d) Compute the least squares estimates of slope and intercept viewing heart rate as the response variable and temperature as the explanatory variable. Interpret the results in the context of the study.

According to the estimators:

$\hat{\beta}_1 = \dfrac{\sum_{i=1}^{n}(X_i - \bar{X})(Y_i - \bar{Y})}{\sum_{i=1}^{n}(X_i - \bar{X})^2}$, $\quad \hat{\beta}_0 = \dfrac{1}{n}(\sum_{i=1}^{n}Y_i - \hat{\beta}_1\sum_{i=1}^{n}X_i) = \bar{Y} - \hat{\beta}_1\bar{X}$

2

We can calculate that:

$$\bar{X} = 10, \bar{Y} = \frac{58}{3} \Rightarrow \hat{\beta}_0 = \frac{23}{12}, \hat{\beta}_1 = \frac{209}{120}$$

Interpretation:

The results represent that the intercept is 23/12, and the slope is 209/120. With the increase of temperature, the heart rate in the common grass frog accordingly increases. Thus, the simple linear regression model is $Y_i = \frac{23}{12} + \frac{209}{120} X_i + \varepsilon_i$

(e) At temperature 9 C, estimate the population mean heart rate corresponding to this temperature. Interpret the results in the context of the study.

When temperature equals to 9, the population mean heart rate equals to **17.59**.

Interpretation:

It means that when temperature is 9 C, the population mean heart rate of common grass frog is 17.59 in beats per minute.

(f) At temperature –2 C, estimate the population mean heart rate corresponding to this temperature. Interpret the results in the context of the study.

When temperature equals to -2, the population mean heart rate equals to **-1.57**.

Interpretation:

It means that when temperature is -2 C, the population mean heart rate of common grass frog is -1.57 in beats per minute, which is below 0. That indicates when temperature is -2 C, common grass frog may die.

(g) Provide a suitable estimate of the population error variance in the model. Interpret the results in the context of the study.

Estimate of the population error variance:

$$\varepsilon_i \sim N(0, \sigma^2)$$

$$\hat{\sigma}^2 = \frac{\sum_{i=1}^{n}(Y_i - \hat{Y}_i)^2}{n-2}$$

$$\hat{\sigma}^2 = \frac{2639}{60} / 7 = 6.283$$

Interpretation:

It means that the population error variance of heart rate of common grass frog is 6.283.

3. Prove the following results on page 36 of slides set #6.

(a) Proof:

$$e_i = Y_i - \hat{Y}_i$$

$$\min \sum_i e_i^2 = \min \sum_i (Y_i - \hat{\beta}_0 - \hat{\beta}_1 X_i)^2$$

$$\frac{\partial \sum_i e_i^2}{\partial \hat{\beta}_0} = -2 \sum_i (y_i - \hat{\beta}_0 - \hat{\beta}_1 x_i) = -2 \sum_i e_i = 0 \Rightarrow \sum_i e_i$$

(b) Proof:

$$e_i = Y_i - \hat{Y}_i$$

$$\min \sum_i e_i^2 = \min \sum_i (Y_i - \hat{\beta}_0 - \hat{\beta}_1 X_i)^2$$

$$\frac{\partial \sum_i e_i^2}{\partial \hat{\beta}_1} = -2 \sum_i (Y_i - \hat{\beta}_0 - \hat{\beta}_1 X_i) X_i = -2 \sum_i e_i X_i = 0 \Rightarrow \sum_i e_i X_i = 0$$

(c) Proof:

$$\sum_i Y_i = \sum_i (\hat{Y}_i + e_i) = \sum_i \hat{Y}_i + \sum_i e_i = \sum_i \hat{Y}_i$$

(d) Proof:

$$\sum_i \hat{Y}_i e_i = \sum_i (\hat{\beta}_0 + \hat{\beta}_1 X_i) e_i = \hat{\beta}_0 \sum_i e_i + \hat{\beta}_1 \sum_i e_i X_i = 0$$

(e) Proof:

$$\bar{y} = \frac{\sum y_i}{n} = \frac{\sum (\hat{y}_i + e_i)}{n} = \frac{\sum (\hat{\beta}_0 + \hat{\beta}_1 x_i + e_i)}{n} = \hat{\beta}_0 + \hat{\beta}_1 \bar{x}$$

$$\bar{Y} = \hat{\beta}_0 + \hat{\beta}_1 \bar{X}$$

Thus, the regression line goes through $(\bar{X}, \bar{Y})$

4. Conduct a simulation study to evaluate the sampling distributions of $\hat\beta_0, \hat\beta_1$, MSE

(i.e., residual mean square, $\hat\sigma^2$) using the least squares method.

(a) Simple linear regression analysis

$Y_i = \beta_0 + \beta_1 X_i + \varepsilon_i$

Find estimated regression coefficients:

$$\hat\beta_1 = \frac{\sum_{i=1}^{n}(X_i - \bar X)(Y_i - \bar Y)}{\sum_{i=1}^{n}(X_i - \bar X)^2} \qquad \hat\beta_0 = \frac{1}{n}\left(\sum_{i=1}^{n} Y_i - \hat\beta_1 \sum_{i=1}^{n} X_i\right) = \bar Y - \hat\beta_1 \bar X \qquad \hat\sigma^2 = \frac{\sum_{i=1}^{n}(Y_i - \hat Y_i)^2}{n-2}$$

With r code, the results are:

$\hat\beta_0 = 8.6968 \qquad \hat\beta_1 = 4.1559 \qquad \hat\sigma^2 = 1.347^2 = 1.814$
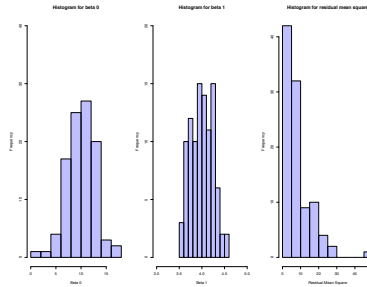
i. Draw one histogram

figure 4.1 Histogram of beta 0 & beta 1 & residual mean square

What observations can you make about these histograms? How consistent are they with the theoretical

results?

From these histograms, I found that $\hat\beta_0, \hat\beta_1$ are approximate normal distribution, and $\hat\sigma^2$ is

approximate chi-squared distribution. The observations are nearly consistent with the theoretical

results.

ii. 95% confidence interval for E(Yh)

I compute a 95% confidence interval for E(Yh) when Xh = 7 and repeat this 100 times.

I found that the proportion of the 100 confidence intervals for E(Yh) when Xh = 7 cover E(Yh|Xh) is

about 0.96. It indicates that about 96% probability that the confidence intervals for E(Yh) will cover

E(Yh|Xh).

(b) Repeat (a) but this time with {Xi} = {2, 4, 6, 8, 10, 12, 14, 16, 18, 20}. How do the

result compare with those in (a)?

Find estimated regression coefficients:

$\hat\beta_0 = 8.415 \qquad \hat\beta_1 = 4.215 \qquad \hat\sigma^2 = 3.905^2 = 15.25$
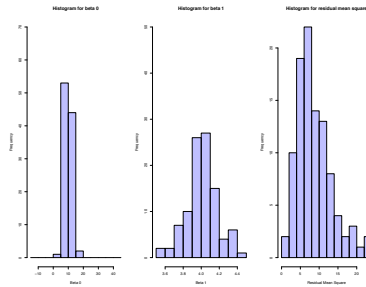
i. Draw one histogram

figure 4.2 Histogram of beta 0 & beta 1 & residual mean square

Compared with result in (a), the residual mean square is larger than that in (a). β^0, β^1 are approximate the same.

ii. 95% confidence interval for E(Yh)

I compute a 95% confidence interval for E(Yh) when Xh = 7 and repeat this 100 times.

I found that the proportion of the 100 confidence intervals for E(Yh) when Xh = 7 cover E(Yh|Xh) is about 0.93. It indicates that about 93% probability that the confidence intervals for E(Yh) will cover E(Yh|Xh).

It is almost the same as (a).

(c) Repeat (a) but this time with $\{Xi\} = \{2, 6, 10, 14, 18, 22, 26, 30, 34, 38\}$. How do the result com- pare with those in (a) and (b)?

Find estimated regression coefficients:

$$\hat{\beta}_0 = 8.6293 \qquad \hat{\beta}_1 = 4.1074 \qquad \hat{\sigma}^2 = 3.905^2 = 15.25$$
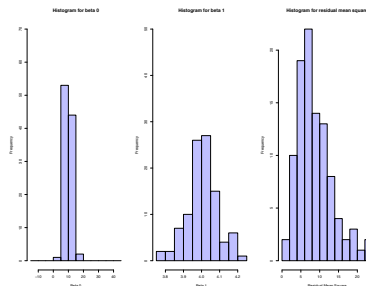
i. Draw one histogram

figure 4.3 Histogram of beta 0 & beta 1 & residual mean square

Compared with result in (a), the residual mean square is larger than that in (a). β^0, β^1 are approximate the same.

Compared with result in (b), the distributions are the same. And the estimated regression coefficients are approximate the same.

ii. 95% confidence interval for E(Yh)

I compute a 95% confidence interval for E(Yh) when Xh = 7 and repeat this 100 times.

I found that the proportion of the 100 confidence intervals for E(Yh) when Xh = 7 cover E(Yh|Xh) is about 0.94. It indicates that about 94% probability that the confidence intervals for E(Yh) will cover E(Yh|Xh).

It is almost the same as (a), (b).

(d) Repeat (a) but this time assume that the model E(Y|X) = 10 + 4X + 0.1X^2. How do the result compare with those in (a)?

Find estimated regression coefficients:

$$\hat{\beta}_0 = 1.897 \qquad \hat{\beta}_1 = 6.156 \qquad \hat{\sigma}^2 = 4.617^2 = 21.35$$

i. Draw one histogram
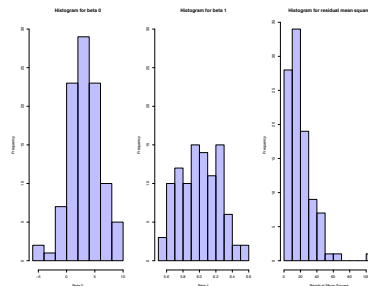

figure 4.4 Histogram of beta 0 & beta 1 & residual mean square

Compared with result in (a), the residual mean square is larger than that in (a) and β^0 is greatly smaller than that in (a). β^1 is a bit larger.

ii. 95% confidence interval for E(Yh)

I compute a 95% confidence interval for E(Yh) when Xh = 7 and repeat this 100 times.

I found that the proportion of the 100 confidence intervals for E(Yh) when Xh = 7 cover E(Yh|Xh) is about 0.97. It indicates that about 97% probability that the confidence intervals for E(Yh) will cover E(Yh|Xh).

It is almost the same as that in (a).

(e) Repeat (a) but this time assume that the model $E(Y|X) = 10 + 4X + 0.2X^2$. How do the result compare with those in (a) and (d)?

Find estimated regression coefficients:

$$\hat{\beta}_0 = -4.903 \qquad \hat{\beta}_1 = 8.156 \qquad \hat{\sigma}^2 = 8.045^2 = 64.72$$
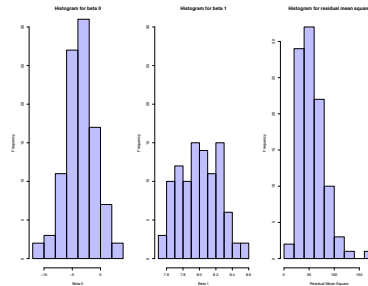
i. Draw one histogram

figure 4.5 Histogram of beta 0 & beta 1 & residual mean square

Compared with result in (a), the residual mean square is greatly larger than that in (a) and β^0 is smaller than that in (a). β^1 is larger.

Compared with result in (d), the residual mean square is larger than that in (a) and β^0 is negative while in (d) that is positive. β^1 is a bit larger.

ii. 95% confidence interval for E(Yh)

I compute a 95% confidence interval for E(Yh) when Xh = 7 and repeat this 100 times.

I found that the proportion of the 100 confidence intervals for E(Yh) when Xh = 7 cover E(Yh|Xh) is about 1. It indicates that about 100% probability that the confidence intervals for E(Yh) will cover E(Yh|Xh).

It is even larger than that in (a).

5. Continue to work on the frog data in Problem 2. Let β0 and β1 denote the intercept and the slope in the simple linear regression model $Y_i = \beta_0 + \beta_1 X_i + \varepsilon_i$ where $\varepsilon_i \sim$ iid $N(0, \sigma^2)$ for $I = 1, 2, \ldots, n$.

(a) T test for β1

Hypothesis:

$$H_0 : \beta_1 = 0$$

$$H_1 : \beta_1 \neq 0$$

Test statistics:

$$T = \frac{(\hat{\beta}_1 - \beta_1)}{s\hat{e}(\hat{\beta}_1)} = \frac{\hat{\beta}_1}{\sqrt{\dfrac{\hat{\sigma}^2}{\sum_{i=1}^{n}(X_i - \bar{X})^2}}}$$

t = 10.764

P-value:

Under the null hypothesis: $T \sim T(n-2)$

p-value= 2*P(T(n-2)>|t|)

p-value = 1.31e-05 <0.05, so we should reject the null hypothesis, there is strong evidence that there is linear relationship between temperature and the heart rate in the common grass frog.

Confidence Interval:

$$\beta_1 \in [\hat{\beta}_1 \pm t_{n-2, \frac{\alpha}{2}} s\hat{e}(\hat{\beta}_1)]$$

95% confidence interval for β1 is [1.359, 2.124]

(b) A power analysis

Hand calculation:

Power = 1-pr(z-b/s*sqrt(Sxx))+pr(-z-b/s*sqrt(Sxx))

β1 = 0, power = 0.05

β1 = ±0.5, power = 0.8725282

Use computer:

β1 = ±1.0, power = 0.9999887

β1 = ±1.5, power = 1

(c) Power curves

(i)

x<-c(2, 4, 6, 8, 10, 12, 14, 16, 18)

y<-c(5, 11, 10, 13, 22, 23, 30, 28, 32)
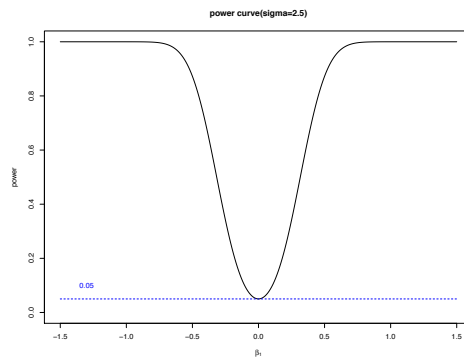
figure 5.1 Power Curve for (b)

(ii)

x<-c(4, 8, 12, 16)

y<-c(11, 13, 23, 28)

$\beta 1 = 0$, power $= 0.05$

$\beta 1 = \pm 0.5$, power $= 0.5433681$

$\beta 1 = \pm 1.0$, power $= 0.9852846$

$\beta 1 = \pm 1.5$, power $= 0.9999891$

figure 5.2 Power Curve for (c)

Comparison:

Compare two plots, I found that when the data set becomes larger, the range of the plot will be narrower, making the plot steeper, like the figure 5.1. Also, when the data set becomes larger, at the same $\beta 1$, the power value will be larger.

(d) T test for β0

Hypothesis:

$$H_0 : \beta_0 = 0$$

$$H_1 : \beta_0 \neq 0$$

Test statistics:

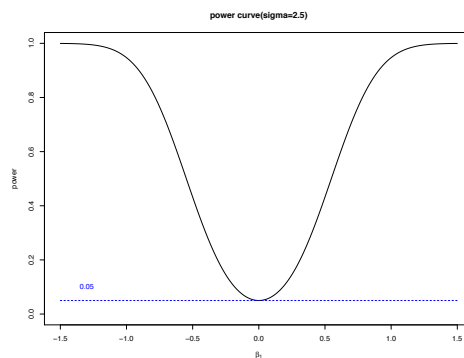$$T = \frac{(\hat{\beta}_0 - \beta_0)}{s\hat{e}(\hat{\beta}_0)} = \frac{\hat{\beta}_0}{\sqrt{\hat{\sigma}^2 (\frac{1}{n} + \frac{\overline{X}^2}{\sum_{i=1}^{n}(X_i - \overline{X})^2})}}$$

t = 1.053

P-value:

Under the null hypothesis: $T \sim T(n-2)$

p-value= $2 * P(T(n-2) > |t|)$

p-value = 0.328 >0.05, so we should accept the null hypothesis, there is an evidence that intercept can be zero.

Confidence Interval:

$$\beta_0 \in [\hat{\beta}_0 \pm t_{n-2,\frac{\alpha}{2}} s\hat{e}(\hat{\beta}_0)]$$

95% confidence interval for β0 is [-2.389, 6.223]

(e) Confidence interval for the error variance

95% confidence interval (two sided)

$$\frac{(n-2)\hat{\sigma}^2}{\sigma^2} \sim \chi^2_{n-2}$$

$$\sigma^2 \in [\frac{(n-2)\hat{\sigma}^2}{\chi^2_{n-2,1-\frac{\alpha}{2}}}, \frac{(n-2)\hat{\sigma}^2}{\chi^2_{n-2,\frac{\alpha}{2}}}]$$

$$\hat{\sigma}^2 = \frac{2639}{60} / 7 = 6.283$$

$$\chi^2_{n-2,1-\frac{\alpha}{2}} = 16.01, \chi^2_{n-2,\frac{\alpha}{2}} = 1.69$$

$$\sigma^2 \in [2.747, 26.024]$$

(f) Provide an appropriate 95% interval for the population mean heart rate at temperature 9 C.

$$\mu_h \in [\hat{\mu}_h - t_{n-2,\frac{\alpha}{2}}\sqrt{\hat{V}ar(\hat{\mu}_h)}, \hat{\mu}_h + t_{n-2,\frac{\alpha}{2}}\sqrt{\hat{V}ar(\hat{\mu}_h)}]$$

$$\hat{\mu}_h = 17.59$$

$$\hat{\sigma}^2 = \frac{2639}{60}/7 = 6.283$$

$$\hat{V}ar(\hat{\mu}_h) = \hat{\sigma}^2\left\{\frac{1}{n} + \frac{(X_h - \bar{X})^2}{\sum(X_i - \bar{X})^2}\right\} = 0.724$$

$$t_{n-2,\frac{\alpha}{2}} = 2.365$$

$$\Rightarrow \mu_h \in [15.58, 19.60]$$

(g) Provide an appropriate 95% interval for the heart rate of a frog to be drawn at random such that the temperature will be manipulated to be –2 C.

$$\mu_h \in [\hat{\mu}_h - t_{n-2,\frac{\alpha}{2}}\sqrt{\hat{V}ar(\hat{\mu}_h)}, \hat{\mu}_h + t_{n-2,\frac{\alpha}{2}}\sqrt{\hat{V}ar(\hat{\mu}_h)}]$$

$$\hat{\mu}_h = -1.57$$

$$\hat{\sigma}^2 = \frac{2639}{60}/7 = 6.283$$

$$\hat{V}ar(\hat{\mu}_h) = \hat{\sigma}^2\left\{\frac{1}{n} + \frac{(X_h - \bar{X})^2}{\sum(X_i - \bar{X})^2}\right\} = 4.468$$

$$t_{n-2,\frac{\alpha}{2}} = 2.365$$

$$\Rightarrow \mu_h \in [-6.57, 3.43]$$

(h) Provide the coefficient of determination R^2

$$R^2 = 1 - \frac{\sum_{i=1}^{n}(Y_i - \hat{Y}_i)^2}{\sum_{i=1}^{n}(Y_i - \bar{Y}_i)^2}$$

Multiple R-squared: 0.943

(i) Sample correlation coefficient $\hat{\varrho}$

Sample correlation coefficient:

$$\hat{\rho} = \frac{\sum_{i=1}^{n}(x_i - \bar{x}_i)(y_i - \bar{y}_i)}{\sqrt{\sum_{i=1}^{n}(x_i - \bar{x}_i)^2}\sqrt{\sum_{i=1}^{n}(y_i - \bar{y}_i)^2}}$$

$$\hat{\rho} = 0.97$$

95% confidence interval for ϱ:

$$\lambda(\hat{\rho}) = \frac{1}{2}\ln(\frac{1+\hat{\rho}}{1-\hat{\rho}}) = 2.0923$$

$$\lambda(\hat{\rho}) \pm z_{\frac{\alpha}{2}}\sqrt{\frac{1}{n-3}} = [\hat{\lambda}_1, \hat{\lambda}_2] = [1.2921, 2.8925]$$

$$\frac{e^{2\hat{\lambda}_1} - 1}{e^{2\hat{\lambda}_1} + 1} \le \rho \le \frac{e^{2\hat{\lambda}_2} - 1}{e^{2\hat{\lambda}_2} + 1} \Rightarrow \rho \in [0.8597, 0.9939]$$

Describe the model:

ϱ is a measure of linear relationship between X and Y.

$\hat{\rho} = 0.97$ indicates that there is strong positive correlation between the temperature and the frog heart rate. That means with the increase of temperature, the frog heart rate will increase as well.

6. Bivariate Normal Distribution

(a) Probability density function for a bivariate normal distribution

In scalar terms:

$$f_{X,Y}(x,y) = \frac{1}{2\pi\sigma_X\sigma_Y\sqrt{1-\rho^2}}\exp[-\frac{z}{2(1-\rho^2)}]$$

$$z = \frac{(x-\mu_X)^2}{\sigma_X^2} - \frac{2\rho(x-\mu_X)(y-\mu_Y)}{\sigma_X\sigma_Y} + \frac{(y-\mu_Y)^2}{\sigma_Y^2}$$

$$\rho = cor(x,y) = \frac{\sigma_{XY}}{\sigma_X\sigma_Y}$$

In matrix forms:

$$Z = \begin{pmatrix} X \\ Y \end{pmatrix} \quad \mu = \begin{pmatrix} \mu_X \\ \mu_Y \end{pmatrix} \quad \Sigma = \begin{pmatrix} \sigma_X^2 & \rho\sigma_X\sigma_Y \\ \rho\sigma_X\sigma_Y & \sigma_Y^2 \end{pmatrix}$$

$$f_Z(z) = \frac{1}{2\pi}(\det\Sigma)^{-\frac{1}{2}}\exp[-\frac{1}{2}(z-\mu)^T\Sigma^{-1}(z-\mu)]$$

(b) Marginal distributions

$$f_X(x) = \int_{-\infty}^{+\infty} f_Z(z)dy = \frac{1}{\sigma_X\sqrt{2\pi}}e^{-\frac{(x-\mu_X)^2}{2\sigma_X^2}}$$

$$\Rightarrow X \sim N(\mu_X,\sigma_X^2)$$

$$f_Y(y) = \int_{-\infty}^{+\infty} f_Z(z)dx = \frac{1}{\sigma_Y\sqrt{2\pi}}e^{-\frac{(y-\mu_Y)^2}{2\sigma_Y^2}}$$

$$\Rightarrow Y \sim N(\mu_Y,\sigma_Y^2)$$

(C) Conditional distribution of Y given X

$$X = \mu_X + \sigma_X Z_1$$

$$Y = \mu_Y + \sigma_Y Z_2 = \mu_Y + \sigma_Y(\rho Z_1 + \sqrt{1-\rho^2}Z_2)$$

$$Z_1, Z_2 \sim N(0,1)$$

$$E[Y \mid X = x] = E[\mu_Y + \sigma_Y(\rho Z_1 + \sqrt{1-\rho^2}Z_2) \mid X = x]$$

$$= E[\mu_Y + \sigma_Y(\rho\frac{X - \mu_X}{\sigma_X} + \sqrt{1-\rho^2}Z_2) \mid X = x]$$

$$= \mu_Y + \sigma_Y(\rho\frac{X - \mu_X}{\sigma_X} + \sqrt{1-\rho^2}E[Z_2 \mid X = x])$$

$$= \mu_Y + \sigma_Y\rho(\frac{X - \mu_X}{\sigma_X})$$

$$Var[Y \mid X = x] = Var[\mu_Y + \sigma_Y(\rho Z_1 + \sqrt{1-\rho^2}Z_2) \mid X = x]$$

$$= Var[\mu_Y + \sigma_Y(\rho\frac{X - \mu_X}{\sigma_X} + \sqrt{1-\rho^2}Z_2) \mid X = x]$$

$$= Var[\sigma_Y(1-\rho^2)Z_2) \mid X = x]$$

$$= \sigma_Y^2(1-\rho^2)$$

$$\Rightarrow Y \mid X = x \sim N(\alpha + \beta x, \sigma_{Y|x}^2)$$

(d) Conduct a research

The counterparts of dnorm: dmvnorm

First, it is a function for dmvnorm, by this function, we can change any data to get the density of bivariate normal distribution.

```
dmvnorm <- function (x, mu, Sigma, log = FALSE, tol = 1e-06) {
    if (is.vector(x))
        x = t(as.matrix(x))
    n = length(mu)
    if (is.vector(mu)) {
        p <- length(mu)
        if (is.matrix(x)) {
            mu <- matrix(rep(mu, nrow(x)), ncol = p, byrow = TRUE)
        }
    }
    else {
        p <- ncol(mu)
    }
    if (!all(dim(Sigma) == c(p, p)) || nrow(x) != nrow(mu))
        stop("incompatible arguments")
    eS <- eigen(Sigma, symmetric = TRUE)
    ev <- eS$values
    if (!all(ev >= -tol * abs(ev[1])))
```

```
      stop("Sigma is not positive definite")
   z = t(x - mu)
   logdetS = try(determinant(Sigma, logarithm = TRUE)$modulus)
   attributes(logdetS) <- NULL
   iS = try(solve(Sigma))
   if (class(iS) == "try-error" || class(logdetS) == "try-error") {
      warning("difficulty inverting/taking determinant of Var-Cov matrix")
      return(NA)
   }
   ssq = diag(t(z) %*% iS %*% z)
   loglik = -(n * (log(2*pi)) +    logdetS + ssq)/2
   if (log) loglik else exp(loglik)
}
```

Make an example: we can create several sets of data like these and get the results by using this function.

```
M = matrix(c(1,0.5,0.5,0.5,1,0.5,0.5,0.5,1),nrow=3)
dmvnorm(1:3,mu=1:3,Sigma=M,log=TRUE)
```
The result is:
```
[1] -2.410242
dmvnorm(matrix(1:6,nrow=2),mu=1:3,Sigma=M,log=TRUE)
```
The result is:
```
[1] -5.160242 -7.410242
dmvnorm(matrix(1:6,nrow=2),mu=matrix(1:6,nrow=2),Sigma=M,log=TRUE)\
```
The result is:
```
[1] -5.167058 -5.167058
```

The counterparts of rnorm: mvrnorm

To set up for the simulations this first block of code defines N, the number of random samples to simulate, the means of the random variables, and and the covariance matrix. It also provides a small function for drawing confidence ellipses on the simulated data.

```
library(mixtools)
N <- 10 # Number of random samples
set.seed(123)
# Target parameters for univariate normal distributions
rho <- -0.6
mu1 <- 1; s1 <- 2
mu2 <- 1; s2 <- 8
# Parameters for bivariate normal distribution
mu <- c(mu1, mu2) # Mean
```

```
sigma <- matrix(c(s1^2, s1*s2*rho, s1*s2*rho, s2^2),2) # Covariance matrix
```

The first method is to use the mvrnorm() function from the MASS package. The results are as follows:

```
library(MASS)
bvn1 <- mvrnorm(N, mu = mu, Sigma = sigma ) # from MASS package
colnames(bvn1) <- c("bvn1_X1","bvn1_X2")
```

```
> bvn2
        bvn2_X1      bvn2_X2
 [1,] -0.1209513    2.2171472
 [2,]  4.1174166   -6.0305462
 [3,]  1.2585755   11.3558348
 [4,]  1.9218324   -9.3087897
 [5,] -0.3737057    1.4446571
 [6,]  3.4481636   -2.5727841
 [7,]  1.8015429   -0.2153336
 [8,] -0.1116823   15.1042815
 [9,]  1.9957010  -13.9760321
[10,]  2.4027118   -5.3923733
```

For the second method, generate bivariate Normal random varieties with the Cholesky decomposition. Cholesky decomposition of sigma (a positive definite matrix) yields a matrix M such that M times its transpose gives sigma back again. Multiplying M by a matrix of standard random Normal varieties and adding the desired mean gives a matrix of the desired random samples. The results are the same.

```
M <- t(chol(sigma))
# M %*% t(M)
Z <- matrix(rnorm(2*N),2,N) # 2 rows, N/2 columns
bvn2 <- t(M %*% Z) + matrix(rep(mu,N), byrow=TRUE,ncol=2)
colnames(bvn2) <- c("bvn2_X1","bvn2_X2")
```
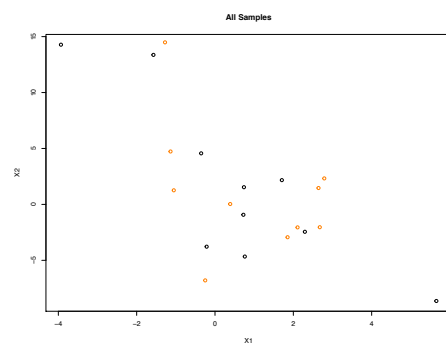
Also, I draw a plot for these samples.

figure 6.1 Samples from bivariate normal distribution

17

Appendix

```
# 2(a)
x<-c(2,4,6,8,10,12,14,16,18) # x represents temperature
y<-c(5,11,10,13,22,23,30,28,32) # y represents heart rate
p <- plot(x,y, xlab="Temperature" ,ylab="Heart Rate", pch=16,col="blue",main="Scatter Plot of
Heart Rate vs. Temperature")
abline(lm(y ~ x), col="red", lwd=2, lty=1)
```

```
# 4(a)
S=100 # the number of simulations
n=5
set.seed(2017)
b0=rep(NA,S)
b1=rep(NA,S)
MSe=rep(NA,S)
for (i in 1:S){
    x=c(2,6,10,14,18)
    error=rnorm(n,0,3)
    y=10+x*4+error
    lsfit=lm(y ~ x)
    b0[i]=lsfit$coefficients[1]
    b1[i]=lsfit$coefficients[2]
    b0[i];b1[i];summary(lsfit);
    Sxx=sum((x-mean(x))^2)
    Sxy=sum((x-mean(x))*(y-mean(y)))
    e=lsfit$residuals
    SSe=sum(e^2)
    MSe[i]=SSe/(n-2)
    varb1=MSe[i]/Sxx
    varb0=MSe[i]*(1/n+mean(x)^2/Sxx)
    confint(lsfit)
}
# make graphs
par(mfrow=c(1,3))
hist(b0, ylim=c(0,40),xlab="Beta 0",col=rgb(0,0,1,1/4), main="Histogram for beta 0")
hist(b1, xlim=c(3,5), ylim=c(0,20),xlab="Beta 1",
```

```r
      col=rgb(0,0,1,1/4), main="Histogram for beta 1")
hist(MSe, xlab="Residual Mean Square",col=rgb(0,0,1,1/4), main="Histogram for residual mean
square")
summary(lsfit)
# CI
x=c(2,6,10,14,18)
predict100=replicate(100,predict(lm(10+4*x+rnorm(5,0,3)~x),newdata=data.frame(x=7),interval =
"confidence")[-1])
xh=7
EY=10+4*xh
n=0
for (i in 1:100){
   if (EY>predict100[1,i]&EY<predict100[2,i]){
      n=n+1
   }
}
n
probability=n/100
probability
```

```r
# 4(b)
S=100 # the number of simulations
n=10
set.seed(2017)
b0=rep(NA,S)
b1=rep(NA,S)
MSe=rep(NA,S)
for (i in 1:S){
   x=c(2, 4, 6, 8, 10, 12, 14, 16, 18, 20)
   error=rnorm(n,0,3)
   y=10+x*4+error
   lsfit=lm(y ~ x)
   b0[i]=lsfit$coefficients[1]
   b1[i]=lsfit$coefficients[2]
   b0[i];b1[i];summary(lsfit);
   Sxx=sum((x-mean(x))^2)
```

```
    Sxy=sum((x-mean(x))*(y-mean(y)))
    e=lsfit$residuals
    SSe=sum(e^2)
    MSe[i]=SSe/(n-2)
    varb1=MSe[i]/Sxx
    varb0=MSe[i]*(1/n+mean(x)^2/Sxx)
    confint(lsfit)
}
# make graphs
par(mfrow=c(1,3))
hist(b0, ylim=c(0,70),xlab="Beta 0",col=rgb(0,0,1,1/4), main="Histogram for beta 0",breaks =
seq(-15,45,5))
hist(b1, ylim=c(0,50),xlab="Beta 1",col=rgb(0,0,1,1/4), main="Histogram for beta 1")
hist(MSe, xlab="Residual Mean Square",col=rgb(0,0,1,1/4), main="Histogram for residual mean
square")
summary(lsfit)
# CI
x=c(2, 4, 6, 8, 10, 12, 14, 16, 18, 20)
predict100=replicate(100,predict(lm(10+4*x+rnorm(10,0,3)~x),newdata=data.frame(x=7),interval
= "confidence")[-1])
xh=7
EY=10+4*xh
n=0
for (i in 1:100){
    if (EY>predict100[1,i]&EY<predict100[2,i]){
        n=n+1
    }
}
n
probability=n/100
probability
# 4(c)
S=100 # the number of simulations
n=10
set.seed(2017)
```

```r
b0=rep(NA,S)
b1=rep(NA,S)
MSe=rep(NA,S)
for (i in 1:S){
   x=c(2, 6, 10, 14, 18, 22, 26, 30, 34, 38)
   error=rnorm(n,0,3)
   y=10+x*4+error
   lsfit=lm(y ~ x)
   b0[i]=lsfit$coefficients[1]
   b1[i]=lsfit$coefficients[2]
   b0[i];b1[i];summary(lsfit);
   Sxx=sum((x-mean(x))^2)
   Sxy=sum((x-mean(x))*(y-mean(y)))
   e=lsfit$residuals
   SSe=sum(e^2)
   MSe[i]=SSe/(n-2)
   varb1=MSe[i]/Sxx
   varb0=MSe[i]*(1/n+mean(x)^2/Sxx)
   confint(lsfit)
}
# make graphs
par(mfrow=c(1,3))
hist(b0, ylim=c(0,70),xlab="Beta 0",col=rgb(0,0,1,1/4), main="Histogram for beta 0",breaks =
seq(-15,45,5))
hist(b1, ylim=c(0,50),xlab="Beta 1",col=rgb(0,0,1,1/4), main="Histogram for beta 1")
hist(MSe, xlab="Residual Mean Square",col=rgb(0,0,1,1/4), main="Histogram for residual mean
square")
summary(lsfit)
# CI
x=c(2, 6, 10, 14, 18, 22, 26, 30, 34, 38)
predict100=replicate(100,predict(lm(10+4*x+rnorm(10,0,3)~x),newdata=data.frame(x=7),interval
= "confidence")[-1])
xh=7
EY=10+4*xh
n=0
```

```
for (i in 1:100){
   if (EY>predict100[1,i]&EY<predict100[2,i]){
      n=n+1
   }
}
n
probability=n/100
probability
```

```
# 4(d)
S=100 # the number of simulations
n=5
set.seed(2017)
b0=rep(NA,S)
b1=rep(NA,S)
MSe=rep(NA,S)
for (i in 1:S){
   x=c(2,6,10,14,18)
   error=rnorm(n,0,3)
   y=10+x*4+0.1*x^2+error
   lsfit=lm(y ~ x)
   b0[i]=lsfit$coefficients[1]
   b1[i]=lsfit$coefficients[2]
   b0[i];b1[i];summary(lsfit);
   Sxx=sum((x-mean(x))^2)
   Sxy=sum((x-mean(x))*(y-mean(y)))
   e=lsfit$residuals
   SSe=sum(e^2)
   MSe[i]=SSe/(n-2)
   varb1=MSe[i]/Sxx
   varb0=MSe[i]*(1/n+mean(x)^2/Sxx)
   confint(lsfit)
}
# make graphs
par(mfrow=c(1,3))
hist(b0,ylim=c(0,30),xlab="Beta 0",col=rgb(0,0,1,1/4), main="Histogram for beta 0")
```

```
hist(b1,ylim=c(0,30),xlab="Beta 1",col=rgb(0,0,1,1/4), main="Histogram for beta 1")

hist(MSe, xlab="Residual Mean Square",col=rgb(0,0,1,1/4), main="Histogram for residual mean

square")

summary(lsfit)

# CI

x=c(2, 6, 10, 14, 18)

predict100=replicate(100,predict(lm(10+4*x+0.1*x^2+rnorm(5,0,3)~x),newdata=data.frame(x=7),i

nterval = "confidence")[-1])

xh=7

EY=10+4*xh+0.1*xh^2

n=0

for (i in 1:100){

    if (EY>predict100[1,i]&EY<predict100[2,i]){

        n=n+1

    }

}

n

probability=n/100

probability
```

```
# 4(e)

S=100 # the number of simulations

n=5

set.seed(2017)

b0=rep(NA,S)

b1=rep(NA,S)

MSe=rep(NA,S)

for (i in 1:S){

    x=c(2,6,10,14,18)

    error=rnorm(n,0,3)

    y=10+x*4+0.2*x^2+error

    lsfit=lm(y ~ x)

    b0[i]=lsfit$coefficients[1]

    b1[i]=lsfit$coefficients[2]

    b0[i];b1[i];summary(lsfit);

    Sxx=sum((x-mean(x))^2)
```

```
    Sxy=sum((x-mean(x))*(y-mean(y)))

    e=lsfit$residuals

    SSe=sum(e^2)

    MSe[i]=SSe/(n-2)

    varb1=MSe[i]/Sxx

    varb0=MSe[i]*(1/n+mean(x)^2/Sxx)

    confint(lsfit)

}

# make graphs

par(mfrow=c(1,3))

hist(b0,ylim=c(0,30),xlab="Beta 0",col=rgb(0,0,1,1/4), main="Histogram for beta 0")

hist(b1,ylim=c(0,30),xlab="Beta 1",col=rgb(0,0,1,1/4), main="Histogram for beta 1")

hist(MSe, xlab="Residual Mean Square",col=rgb(0,0,1,1/4), main="Histogram for residual mean
square")

summary(lsfit)

# CI

x=c(2, 6, 10, 14, 18)

predict100=replicate(100,predict(lm(10+4*x+0.2*x^2+rnorm(5,0,3)~x),newdata=data.frame(x=7),i
nterval = "confidence")[-1])

xh=7

EY=10+4*xh+0.2*xh^2

n=0

for (i in 1:100){

    if (EY>predict100[1,i]&EY<predict100[2,i]){

        n=n+1

    }

}

n

probability=n/100

probability

# 5(b)

# calculating power

x<-c(2, 4, 6, 8, 10, 12, 14, 16, 18)

y<-c(5, 11, 10, 13, 22, 23, 30, 28, 32)

Sxx=sum((x-mean(x))^2)
```

```
Sxy=sum((x-mean(x))*(y-mean(y)))
power=function(b,s,a,Sxx){
   z=qnorm(1-a/2)
   print(z)
   print(z-b/s*sqrt(Sxx))
   print(-z-b/s*sqrt(Sxx))
   print(pnorm(z-b/s*sqrt(Sxx)))
   print(pnorm(-z-b/s*sqrt(Sxx)))
   print(1-pnorm(z-b/s*sqrt(Sxx))+pnorm(-z-b/s*sqrt(Sxx)))
}
power(0,2.5,0.05,Sxx) #power when beta1 is 0, sigma is 2.5 and alpha =0.05
power(0.5,2.5,0.05,Sxx) #power when beta1 is 0, sigma is 2.5 and alpha =0.05
power(-0.5,2.5,0.05,Sxx) #power when beta1 is 0, sigma is 2.5 and alpha =0.05
power(1,2.5,0.05,Sxx) #power when beta1 is 1, sigma is 2.5 and alpha =0.05
power(-1,2.5,0.05,Sxx) #power when beta1 is -1, sigma is 2.5 and alpha =0.05
power(1.5,2.5,0.05,Sxx) #power when beta1 is 1.5, sigma is 2.5 and alpha =0.05
power(-1.5,2.5,0.05,Sxx) #power when beta1 is -1.5, sigma is 2.5 and alpha =0.05
```

```
# 5(c)
# make graphs for (b)
x<-c(2, 4, 6, 8, 10, 12, 14, 16, 18)
y<-c(5, 11, 10, 13, 22, 23, 30, 28, 32)
Sxx=sum((x-mean(x))^2)
power.plot=function(b,s,a,Sxx){
   z=qnorm(1-a/2)
   (1-pnorm(z-b/s*sqrt(Sxx))+pnorm(-z-b/s*sqrt(Sxx)))
}

b.seq=seq(-1.5,1.5,0.01)
b.power=power.plot(b.seq,2.5,0.05,Sxx)
b.power
plot(b.seq,b.power,type="l",xlab=expression(beta[1]),ylab="power"
      ,main="power curve(sigma=2.5)",xlim=c(-1.5,1.5),ylim=c(0,1))
segments(-1.5,0.05,1.5,0.05,col="blue",lty=2)
text(-1.3,0.1,"0.05",col="blue")
```

```
# make graphs for (c)
x<-c(4, 8, 12, 16)
y<-c(11, 13, 23, 28)
Sxx=sum((x-mean(x))^2)
power.plot=function(b,s,a,Sxx){
   z=qnorm(1-a/2)
   (1-pnorm(z-b/s*sqrt(Sxx))+pnorm(-z-b/s*sqrt(Sxx)))
}


b.seq=seq(-1.5,1.5,0.01)
b.power=power.plot(b.seq,2.5,0.05,Sxx)
b.power
plot(b.seq,b.power,type="l",xlab=expression(beta[1]),ylab="power"
      ,main="power curve(sigma=2.5)",xlim=c(-1.5,1.5),ylim=c(0,1))
segments(-1.5,0.05,1.5,0.05,col="blue",lty=2)
text(-1.3,0.1,"0.05",col="blue")
```

```
# 6(d)
# mvrnorm
library(mixtools)    #for ellipse
N <- 10 # Number of random samples
set.seed(123)
# Target parameters for univariate normal distributions
rho <- -0.6
mu1 <- 1; s1 <- 2
mu2 <- 1; s2 <- 8
# Parameters for bivariate normal distribution
mu <- c(mu1,mu2) # Mean
sigma <- matrix(c(s1^2, s1*s2*rho, s1*s2*rho, s2^2),2) # Covariance matrix
# method 1
library(MASS)
bvn1 <- mvrnorm(N, mu = mu, Sigma = sigma ) # from MASS package
colnames(bvn1) <- c("bvn1_X1","bvn1_X2")
bvn1
# method 2
M <- t(chol(sigma))
```

```r
# M %*% t(M)
Z <- matrix(rnorm(2*N),2,N) # 2 rows, N/2 columns
bvn2 <- t(M %*% Z) + matrix(rep(mu,N), byrow=TRUE,ncol=2)
colnames(bvn2) <- c("bvn2_X1","bvn2_X2")
bvn2
# make graphs
par(mfrow=c(1,1))
bvn <- list(bvn1,bvn2)
plot(bvn1, xlab="X1",ylab="X2",main= "All Samples")
for(i in 2:2){
    points(bvn[[i]],col=i)
}


# dmvnorm
dmvnorm <- function (x, mu, Sigma, log = FALSE, tol = 1e-06) {
    if (is.vector(x))
        x = t(as.matrix(x))
    n = length(mu)
    if (is.vector(mu)) {
        p <- length(mu)
        if (is.matrix(x)) {
            mu <- matrix(rep(mu, nrow(x)), ncol = p, byrow = TRUE)
        }
    }
    else {
        p <- ncol(mu)
    }
    if (!all(dim(Sigma) == c(p, p)) || nrow(x) != nrow(mu))
        stop("incompatible arguments")
    eS <- eigen(Sigma, symmetric = TRUE)
    ev <- eS$values
    if (!all(ev >= -tol * abs(ev[1])))
        stop("Sigma is not positive definite")
    z = t(x - mu)
    logdetS = try(determinant(Sigma, logarithm = TRUE)$modulus)
```

```
   attributes(logdetS) <- NULL

   iS = try(solve(Sigma))

   if (class(iS) == "try-error" || class(logdetS) == "try-error") {

      warning("difficulty inverting/taking determinant of Var-Cov matrix")

      return(NA)

   }

   ssq = diag(t(z) %*% iS %*% z)

   loglik = -(n * (log(2*pi)) +    logdetS + ssq)/2

   if (log) loglik else exp(loglik)

}
# make an example
M = matrix(c(1,0.5,0.5,0.5,1,0.5,0.5,0.5,1),nrow=3)
dmvnorm(1:3,mu=1:3,Sigma=M,log=TRUE)
dmvnorm(matrix(1:6,nrow=2),mu=1:3,Sigma=M,log=TRUE)
dmvnorm(matrix(1:6,nrow=2),mu=matrix(1:6,nrow=2),Sigma=M,log=TRUE)
```