Applied Statistics  Problem Sheet 0  MT 2018

This is an introductory and revision sheet which you can practise on as the course begins. These exercises will not be covered in problems classes, mostly they should be revision. A couple of things below may be written in a different way to what you are used to, but the notation is explained.

1. Likelihood ratio tests.

Suppose $y_1, \ldots, y_n \overset{iid}{\sim} f(y; \theta)$ where $\theta$ is an element of the parameter space $\Theta$. Let $\Theta_0$ be a subset of $\Theta$, where $\dim \Theta = p$, $\dim \Theta_0 = q$ and $q < p$. Here $\dim \Theta$ denotes the dimension of $\Theta$, i.e. the number of free parameters in $\Theta$, and similarly for $\dim \Theta_0$.

Consider testing the null hypothesis $H_0: \theta \in \Theta_0$ against the general alternative $H_1: \theta \in \Theta$.

(a) What is the definition of the log-likelihood ratio statistic for testing $H_0$? Denote this statistic by $\Lambda(y)$.

(b) What is the approximate distribution of $\Lambda(y)$ under $H_0$? What can you say about the conditions required for this to be a good approximation?

2. Standard distributions.

Let $z_1, z_2, \ldots \overset{iid}{\sim} N(0, 1)$. Write down, in terms of $z_1, z_2, \ldots$, a random variable whose distribution is:

(a) The chi-squared distribution with $r$ degrees of freedom.

(b) The $t$ distribution with $r$ degrees of freedom.

(c) The $F$ distribution with $m$ and $n$ degrees of freedom (if you are not familiar with the $F$ distribution, look it up).

3. Linear regression models.

Consider the simple linear regression model

$$y_i = \beta_0 + \beta_1 x_i + \epsilon_i, \quad i = 1, \ldots, n$$

(1)

where $\epsilon_1, \ldots, \epsilon_n \overset{iid}{\sim} N(0, \sigma^2)$.

(a) Model (1) can also be written

$$y_i = \gamma_0 + \gamma_1 (x_i - \bar{x}) + \epsilon_i, \quad i = 1, \ldots, n$$

where $\bar{x} = \frac{1}{n} \sum_{i=1}^n x_i$. Write down expressions for $\beta_0$ and $\beta_1$ in terms of $\gamma_0$ and $\gamma_1$. How do the MLEs $\hat{\beta}_0$ and $\hat{\beta}_1$ relate to the MLEs $\hat{\gamma}_0$ and $\hat{\gamma}_1$?

(b) Let

$$y = \begin{pmatrix} y_1 \\ \vdots \\ y_n \end{pmatrix}, \quad \beta = \begin{pmatrix} \beta_0 \\ \beta_1 \end{pmatrix}, \quad e = \begin{pmatrix} \epsilon_1 \\ \vdots \\ \epsilon_n \end{pmatrix}.$$ 

Note that (1) represents $n$ equations: write down the appropriate matrix $X$ so that these equations, written in matrix form, are

$$y = X\beta + e.$$
(c) Write down the likelihood for model (1), and show that the log-likelihood
\[ \ell(\beta, \sigma^2; y) \] can be written
\[ \ell(\beta, \sigma^2; y) = -\frac{n}{2} \log \sigma^2 - \frac{1}{2\sigma^2} SS(\beta) + \text{constant} \]
\[ = -\frac{n}{2} \log \sigma^2 - \frac{1}{2\sigma^2} (y - X\beta)^T (y - X\beta) + \text{constant} \]
where \( SS(\beta) = \sum_{i=1}^{n}(y_i - \beta_0 - \beta_1 x_i)^2 \), and where a superscript of \( T \) denotes transpose.

(d) Consider the multiple regression model
\[ y_i = \beta_1 x_{i1} + \beta_2 x_{i2} + \cdots + \beta_p x_{ip} + \epsilon_i, \quad i = 1, \ldots, n. \]
If this model is written as \( y = X\beta + \epsilon \), what is \( X \)?

4. Mean vectors and covariance matrices.
Let \( y = (y_1, \ldots, y_n)^T \) be an \( n \times 1 \) vector of random variables. Recall that the \( n \times 1 \) mean vector \( \mu = (\mu_i) \) and the \( n \times n \) covariance matrix \( \Sigma = (\Sigma_{ij}) \) of \( y \) are defined by
\[ \mu_i = E(y_i), \quad i = 1, \ldots, n \]
\[ \Sigma_{ii} = \text{var}(y_i), \quad i = 1, \ldots, n \]
\[ \Sigma_{ij} = \text{cov}(y_i, y_j), \quad i \neq j = 1, \ldots, n. \]
We write \( E(y) = \mu \) and \( \text{var}(y) = \Sigma \).

If \( A \) is a matrix of constants with \( n \) columns, show that
(a) \( \text{var}(y) = E[(y - \mu)(y - \mu)^T] \)
(b) \( E(Ay) = AE(y) \)
(c) \( \text{var}(Ay) = A \text{var}(y) A^T \).

5. Fitting a linear regression in R.
Work through the following example in R and think about what this simple analysis is doing.

Old Faithful is a geyser in Yellowstone National Park, USA. The dataset faithful is built-in to R. To look at the first few rows of the dataset use:
`head(faithful)`

To see what the data represent, use `?faithful` and read the first few lines of the help page.

Note: the function `head()` shows only the first part of a data frame or vector. We use it here to avoid getting too much output. Type `faithful` if you want to see the whole data frame (too much information). To get summaries use:
`str(faithful)`
`summary(faithful)`

Consider using the duration of the current eruption to predict the length of time until the next eruption takes place. Plot the data, fit a simple linear regression and draw the regression line on the plot:
plot(waiting ~ eruptions, data = faithful, 
    xlab = "duration of current eruption (minutes)", 
    ylab = "time until next eruption (minutes)")
fit1 <- lm(waiting ~ eruptions, data = faithful)
abline(fit1, col = "blue")

Summarise the fitted model:

fit1
summary(fit1)

A couple of residual plots:

plot(resid(fit1) ~ fitted(fit1), main = "Residuals vs Fitted values")
qqnorm(resid(fit1), main = "Normal Q-Q plot of residuals")
qqline(resid(fit1))

Confidence and prediction intervals at durations of 1.6, 2.1, ..., 5.1 minutes:

new <- data.frame(eruptions = seq(1.6, 5.1, by = 0.5))
predict(fit1, newdata = new)
predict(fit1, newdata = new, interval = "confidence")
predict(fit1, newdata = new, interval = "prediction")

Add confidence and prediction intervals to the original plot:

new <- data.frame(eruptions = seq(1.6, 5.1, by = 0.5))
p.conf <- predict(fit1, newdata = new, interval = "confidence")
p.pred <- predict(fit1, newdata = new, interval = "prediction")
erup <- new$eruptions
plot(waiting ~ eruptions, data = faithful, 
    xlab = "duration of current eruption (minutes)", 
    ylab = "time until next eruption (minutes)")
lines(p.conf[, 1] ~ erup, col = "blue") # fitted values are 1st column of p.conf
lines(p.conf[, 2] ~ erup, lty = 2) # lwr conf values are 2nd column
lines(p.conf[, 3] ~ erup, lty = 2) # upr conf values are 3rd column
lines(p.pred[, 2] ~ erup, lty = 2, col = "red")
lines(p.pred[, 3] ~ erup, lty = 2, col = "red")
legend("topleft", 
c("fitted line", "confidence intervals", "prediction intervals"), 
lty = c(1, 2, 2), col = c("blue", 1, "red"))