Errata and Notes

Statistical Computing with R
Chapman & Hall/CRC 2008
Maria L. Rizzo

Errata

p. 30  Last line: \( x \geq 0, \nu = 1, 2, \ldots \).

p. 35  In the first two displayed equations: \( \lim_{n \to \infty} \).

p. 57  In se calculation (middle of page) \( \text{dbeta}(Q, 2, 2) \) should be \( \text{dbeta}(Q, 2, 2)^{-2} \).

p. 58  Item 4. \( Z_2 = \sqrt{-2 \log U \sin(2\pi V)} \).

p. 121 In step 2: \( \overline{g}(X) = \frac{1}{m} \sum_{i=1}^{m} g(X_i) \).

p. 132 In sentence above (5.9): \( \hat{\theta}_c = g(X) + c(f(X) - \mu) \).

p. 133 Line 4 from bottom: 100(0.2429355 - 0.003940175)/0.2429355.

p. 158 Table 6.1: \( n \hat{\sigma}_c \) for \( p = 0.95 \) and \( p = 0.90 \) should be different from \( p = 1 \) case. See note below.

p. 187 Line 1. Remove hat from the first ‘se’.

p. 200 Example 7.10. Misplaced right paren.; correction:

```r
#normal
print(boot.obj$t0 + qnorm(alpha) * sd(boot.obj$t))
```

p. 204

\[
\hat{a} = \frac{\sum_{i=1}^{n} (\theta_{(i)} - \theta_{(i)})^3}{6 \left( \sum_{i=1}^{n} (\theta_{(i)} - \theta_{(i)})^2 \right)^{3/2}}, \quad (7.11)
\]

p. 260 Last paragraph: delete the second sentence "Then an observed sample is generated."

p. 263 Example 9.7.

\[
E[X_2|x_1] = \mu_2 + \rho \frac{\sigma_2}{\sigma_1} (x_1 - \mu_1)
\]

p. 312 Example 10.15. Correction: The mean vectors are

\[
\mu_1 = \begin{bmatrix} 0 \\ 0 \end{bmatrix}, \quad \mu_2 = \begin{bmatrix} 1 \\ 3 \end{bmatrix}, \quad \mu_3 = \begin{bmatrix} 4 \\ -1 \end{bmatrix}.
\]
Notes

p. 56 Remark on Example 3.7. Although 6 is an upper bound, it is not the least upper bound. The generator is more efficient if \( c = 1.5 \), the maximum value of \( f(x)/g(x) \) for \( 0 \leq x \leq 1 \).

p. 57 Code above para. 2: #See Ch. 2

p. 71–72 Example 3.16, summary statistics. The \texttt{rmv.eigen} generator takes the covariance matrix \texttt{Sigma} as an argument, so in general one may want to display \texttt{cov(X)} for comparison with \texttt{Sigma} rather than the sample correlation matrix \texttt{cor(X)}. (Here our \texttt{Sigma} was a correlation matrix.)

p. 158 Table 6.1. With \texttt{set.seed(522)} the corresponding table should be:

<table>
<thead>
<tr>
<th>k</th>
<th>Normal ( \hat{MSE} )</th>
<th>( n \hat{\sigma} )</th>
<th>Normal ( \hat{MSE} )</th>
<th>( n \hat{\sigma} )</th>
<th>Normal ( \hat{MSE} )</th>
<th>( n \hat{\sigma} )</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0.976</td>
<td>0.140</td>
<td>6.229</td>
<td>0.353</td>
<td>11.485</td>
<td>0.479</td>
</tr>
<tr>
<td>1</td>
<td>1.019</td>
<td>0.143</td>
<td>1.954</td>
<td>0.198</td>
<td>4.126</td>
<td>0.287</td>
</tr>
<tr>
<td>2</td>
<td>1.009</td>
<td>0.142</td>
<td>1.304</td>
<td>0.161</td>
<td>1.956</td>
<td>0.198</td>
</tr>
<tr>
<td>3</td>
<td>1.081</td>
<td>0.147</td>
<td>1.168</td>
<td>0.153</td>
<td>1.578</td>
<td>0.178</td>
</tr>
<tr>
<td>4</td>
<td>1.048</td>
<td>0.145</td>
<td>1.280</td>
<td>0.160</td>
<td>1.453</td>
<td>0.170</td>
</tr>
<tr>
<td>5</td>
<td>1.103</td>
<td>0.149</td>
<td>1.395</td>
<td>0.167</td>
<td>1.423</td>
<td>0.169</td>
</tr>
<tr>
<td>6</td>
<td>1.316</td>
<td>0.162</td>
<td>1.349</td>
<td>0.164</td>
<td>1.574</td>
<td>0.177</td>
</tr>
<tr>
<td>7</td>
<td>1.377</td>
<td>0.166</td>
<td>1.503</td>
<td>0.173</td>
<td>1.734</td>
<td>0.186</td>
</tr>
<tr>
<td>8</td>
<td>1.382</td>
<td>0.166</td>
<td>1.525</td>
<td>0.175</td>
<td>1.694</td>
<td>0.184</td>
</tr>
<tr>
<td>9</td>
<td>1.491</td>
<td>0.172</td>
<td>1.646</td>
<td>0.181</td>
<td>1.843</td>
<td>0.192</td>
</tr>
</tbody>
</table>

p. 178–179 Examples 6.14–6.15. Although mathematically it is not an error, it is unnecessary to subtract the sample means in the expression \texttt{tests} of Example 6.14 because the sample means are subtracted in the function \texttt{count5test}. Mathematically, if \( Z_i = X_i - \bar{X} \), then \( Z = 0 \). The same is true in the expression for \texttt{alphahat} in Example 6.15. The \texttt{count5test} can be applied without centering the data first, as in Example 6.16.

p. 225–228 Examples 8.4-8.6. The \texttt{knnFinder} package with \texttt{nn} function for finding nearest neighbors has been withdrawn from CRAN. These examples have been revised using the \texttt{nn} function in the \texttt{yaImpute} package.

p. 323-333 Example 11.3. In \texttt{system.time} the timings are hardware dependent; however, the vectorized version should be faster on all platforms.

pp. 338–339 Example 11.11. The first code snippet will produce a graph similar to Figure 11.3 but with \( x \)-axis ranging from about 2 to 8. To produce Figure 11.3 as shown, replace 8 with 15 in \texttt{seq(2, 8, .001)}.

p. 341 Example 11.12. The histograms will of course vary slightly from Figure 11.4 because the data is generated at random. According to my notes,
**Programming Notes**

1. *curve* is convenient for plotting a function. It can replace *lines* in some examples; e.g., in Example 3.2 to add the density curve to the histogram, instead of *lines* we can use:
   ```r
curve(3x^2, add=TRUE)
```

2. Displaying a sequence of graphs: *par(ask=TRUE)* has the effect that the user is asked for input before each new figure is drawn. Follow it with *par(ask=FALSE)* to restore to default behavior. Alternately in the R GUI use “Recording” from the graph History menu.

3. *sapply* can be used instead of *apply* in some examples, which eliminates the need for MARGIN and the need for the argument to have a dimension attribute. See e.g. Example 5.4 on page 123. The lines:
   ```r
dim(x) <- length(x)
p <- apply(x, MARGIN=1, function(x, z) {mean(z < x)}, z=z)
```
   can be replaced with either version below:
   ```r
   p <- sapply(x, FUN=function(x,z) mean(z<x), z=z)
p <- sapply(x, function(x) mean(z<x))
```

4. A more elegant approach to the comparison of generators in Example 3.19 is to wrap the repeated statements in a function that takes the name of the generator (e.g. `rmvn.eigen`) as an argument. An example of a function that has a functional argument is *boot*; see Example 7.10 for a typical example.

**Acknowledgements**  Thanks to several readers who sent comments, corrections, and suggestions.