**Corrections to Ledolter: Data Mining and Business Analytics with R. Wiley 2013 (as of April 21, 2016)**

Page 157, line 5 from bottom. It should be: `zqua=qda(Sp~.,Iris,prior=c(1,1,1)/3)`

Page 192, Section 14.4. Better introductory discussions of Support Vector Machines can be found in other texts. See, for example, F. Provost and T. Fawcett: Data Science for Business, O'Reilly Media, 2013.

Page 196, line 4: better to write: …. so as to minimize the sum of the within-cluster sums of squares,

Pages 213-216 (Table 15.1 and Table 15.2, Figure 15.1): The entries in the 5x5 "distance" matrix are not distances as they do not satisfy the triangle inequality. For example, d({a},{e}) = 11 is not smaller than d({a,},{c}) + d({c},{e}) = 3 + 2. The updating calculations for distances under single-linkage and complete-linkage are correct, however the clustering results in Figure 15.1 exaggerate the differences between single-and complete linkage clustering (because the entries in the 5x5 matrix are not distances). Replace the bold 2 in the 5x5matrix by 9, check that the resulting matrix is indeed a distance matrix, and repeat the calculations.

Page 218, line 1 of program should be changed to

```
foodaggcomp=agnes(food[,-1],diss=FALSE,metric="euclidian")
```

This also results in a change of the dendrogram.

Page 221, last line: Hence, one screens for combinations that result in a good lift and a reasonable large support for the consequent.

Page 223, line 22: We look for artists (or groups of artists) who give to another artist confidence larger than 0.50 (50%) and support larger than 0.01 (1%).

Page 237, line 21: Lengths of the loading vectors are always constrained to one.

Page 238, line 12: Replace $v_2$ with $v_1$.

Page 277, network graph: The printer failed to add the connection between ""Strozzi" and "Castellani". You can plot the paths as follows:

```
## calculate and plot the shortest paths
V(marriage)$color <- 8
E(marriage)$color <- 8
PtoA <- get.shortest.paths(marriage, from="Peruzzi", to="Acciaiuoli")
E(marriage, path=PtoA$vpath[[1]])$color <- "magenta"
V(marriage)[PtoA$vpath[[1]] ]$color <- "magenta"
GtoS <- get.shortest.paths(marriage, from="Ginori", to="Strozzi")
E(marriage, path=GtoS$vpath[[1]])$color <- "green"
```

```
V(marriage)[ GtoS$vpath[[1]] ]$color <- "green"
V(marriage)[ "Medici" ]$color <- "cyan"

plot(marriage,  layout=layout.fruchterman.reingold,
vertex.label=V(marriage)$name,vertex.label.color="black",
vertex.frame.color=0, vertex.label.cex=1.5)
```

Pages 7, 31, 39, 78, 82, and 340: Montgomery, A.L.: Creating micro-marketing pricing strategies using supermarket scanner data. *Marketing Science*, Vol. 16 (**1997**), 315–337.

**Comments on Chapters 7 (Logistic Regression) with LASSO constraints**

**library(lars) cannot be used for logistic regression. Instead, use library(glmnet).**

Example: Data set 4 (LoanData.csv) on page 295 of Ledolter (2013). Classify loans into just two groups: good = current and bad = [late, default].

```
library(lars)
library(glmnet)
loan <- read.csv("C:/DataMining/Data/LoanData.csv")
loan[1:3,]
levels(loan$Status) = c("Good","Bad","Bad") #Classifies loans into 2 groups
Xloan = model.matrix(Status~., data = loan) [,-1]
Xloan[1:10,]
Y=loan$Status
Y
fit=glmnet(x=Xloan,y=Y,family="binomial")
fit
plot(fit)
cv.fit1=cv.glmnet(x=Xloan,y=Y,family="binomial",type.measure="deviance")
plot(cv.fit1)
cv.fit1$lambda.1se
cv.fit2=cv.glmnet(x=Xloan, y=Y,family="binomial",type.measure="class")
plot(cv.fit2)
cv.fit2$lambda.1se
cv.fit3=cv.glmnet(x=Xloan, y=Y,family="binomial",type.measure="auc")
plot(cv.fit3)
cv.fit3$lambda.1se
cv.fit4=cv.glmnet(x=Xloan,y=Y, family="binomial")
cv.fit4$lambda.1se
coef(cv.fit4)
coef(cv.fit4,s="lambda.1se")
coef(cv.fit4,s="lambda.min")
predict(cv.fit4,newx=Xloan[1:10,],type="response")    ## uses optimal lambda
predict(cv.fit4,newx=Xloan[1:10,],s="lambda.1se",type="response")
## uses optimal lambda”
predict(cv.fit4,newx=Xloan[1:10,],s="lambda.min",type="response")
## uses standard logistic regression estimates
```

**Note:** Get a listing of the response Y. You find that the response is categorical (a factor) with factor levels listed as: Good Bad. The second label (here “Bad”) becomes 1 = success, and the first label “Good” becomes 0 = failure. In this example, success in the logistic regression means a “Bad” loan. We are modeling the probability of a bad loan. Keep that in mind when interpreting the regression coefficients.