Chapter 15

Stuff for Students

15.1 Tips for Doing Research

As a student or new researcher, you will probably encounter researchers who think that their method of doing research is the only correct way of doing research, but there are dozens of methods that have proven effective.

Familiarity with the literature is important since your research should be original. The field of high breakdown (HB) robust statistics has perhaps produced more literature in the past 40 years than any other field in statistics.

This text presents the author's applied research in multivariate analysis from 1997–2012, and a summary of the ideas that most influenced the development of this text follows. Gnanadesikan and Kettenring (1972) suggested an algorithm similar to concentration and suggested that robust covariance estimators could be formed by estimating the elements of the covariance matrix with robust scale estimators. Devlin, Gnanadesikan and Kettenring (1975, 1981) introduced the concentration technique. Rousseeuw (1984) extended the MCD location estimator to the MCD estimator of multivariate location and dispersion. Cook and Nachtsheim (1994) showed that robust Mahalanobis distances could be used to reduce the bias of 1D regression estimators. Rousseeuw and Van Driessen (1999) introduced the DD plot.

Much of the HB literature is not applied or consists of ad hoc methods. In far too many papers, the estimator actually used is an ad hoc inconsistent zero breakdown approximation of an estimator for which there is theory. The MCD, depth and MVE estimators are impractical to compute. The S estimators and projection estimators are currently impossible to compute for

$p > 2$. Unless there is a computational breakthrough, these estimators can rarely be used in practical problems. Similarly, two stage estimators need a good initial HB estimator, but no good initial HB estimator was available until Olive (2004a) and Olive and Hawkins (2007, 2008, 2010).

There are hundreds of papers on outlier detection. Most of these compare their method with an existing method on one or two outlier configurations where their method does better. However, the new method rarely outperforms the existing method (such as `lmsreg` or `cov.mcd`) if a broad class of outlier configurations is examined. In such a paper, check whether the new estimator is consistent and if the author has shown types of outlier configurations where the method fails. **Try to figure out how the method would perform for the cases of one and two predictors**.

Dozens of papers suggest that a classical method can be made robust by replacing a classical estimator with a robust estimator. Again inconsistent robust estimators are usually used. These methods can be very useful, but rely on perfect classification of the data into outliers and clean cases. Check whether these methods can find outliers that can not be found by the response plot, FCH DD plot and FMCD DD plot.

For example consider making a robust Hotelling's t–test. If the paper uses the FMCD `cov.mcd` algorithm, then the procedure is relying on the perfect classification paradigm. On the other hand, Srivastava and Mudholkar (2001) present an estimator that has large sample theory.

Beginners can have a hard time determining whether a robust algorithm estimator is consistent or not. As a rule of thumb, assume that the approximations (including those for depth, MCD, MVE, S, projection estimators and two stage estimators) are inconsistent unless the authors show that they understand Hawkins and Olive (2002) and Olive and Hawkins (2007, 2008, 2010). In particular, the elemental or basic resampling algorithms, concentration algorithms and algorithms based on random projections should be considered inconsistent until you can prove otherwise.

After finding a research topic, **paper trailing** is an important technique for finding related literature. To use this technique, find a paper on the topic, go to the bibliography of the paper, find one or more related papers and repeat. Often your university's library will have useful internet resources for finding literature. Usually a research university will subscribe to either *The Web of Knowledge* with a link to ISI Web of Science or to the *Current Index to Statistics*. Both of these resources allow you to search for literature by author,

eg Olive, or by topic, eg robust statistics. Both of these methods search for recent papers. With Web of Knowledge, find an article with *General Search*, click on the article and then click on the *Find Related Articles* icon to get a list of related articles. For papers before 1997, use the free *Current Index to Statistics* website (http://query.statindex.org/CIS/OldRecords/queryOld).

The search engines (www.google.com), (www.ask.com), (www.msn.com), (www.yahoo.com), (www.info.com) and (www.scirus.com) are also useful. The google search engine also has a useful link to "Google Scholar." When searching, enter a topic and the word *robust* or *outliers*. For example, enter the keywords *robust factor analysis* or *factor analysis and outliers*.

The STATLIB site (http://lib.stat.cmu.edu/) is useful for finding statistics departments, data sets and software. Statistical journals often have websites that make abstracts and preprints available. Two useful websites are given below.

```
(www.stat.ucla.edu/journals/ProbStatJournals/)
(www.statsci.org/jourlist.html)
```

Websites for researchers or research groups can be very useful. Below are websites for Dr. Rousseeuw's group, Dr. Rocke, Dr. Croux, Dr. Hubert's group and for the University of Minnesota.

```
(www.agoras.ua.ac.be/)
(http://handel.cipic.ucdavis.edu/~dmrocke/preprints.html)
(www.econ.kuleuven.ac.be/public/NDBAE06/)
(http://wis.kuleuven.be/stat/robust.html)
(www.stat.umn.edu)
```

The latter website has useful links to software. *Arc* and *R* can be downloaded from these links. **Familiarity with a high level programming language** such as FORTRAN or *R/Splus* is essential. A very useful *R* link is (www.r-project.org/#doc). See *R Development Core Team* (2011).

Finally, a Ph.D. student needs an advisor or **mentor** and most researchers will find collaboration valuable. Attending conferences and making your research available over the internet can lead to contacts.

Some references on research, including technical writing and presentations, include American Society of Civil Engineers (1950), Becker and Keller-McNulty (1996), Ehrenberg (1982), Freeman, Gonzalez, Hoaglin and Kilss (1983), Hamada and Sitter (2004), Rubin (2004) and Smith (1997).

15.2 R/Splus and Arc

R is the free version of *Splus*. The website (www.stat.umn.edu) has useful links for *Arc* which is the software developed by Cook and Weisberg (1999a). The website (www.stat.umn.edu) also has a link to **Cran** which gives *R* support. As of April 2012, the author's personal computer has Version 2.13.1 (July 8, 2011) of *R*, Splus–2000 (see Mathsoft 1999ab) and Version 1.06 (July 2004) of *Arc*. Many of the text *R/Splus* functions and figures were made in the 1990's using *Splus* on a workstation.

Downloading the book's data.lsp files into Arc

Many homework problems use data files for *Arc* contained in the book's website (www.math.siu.edu/olive/mbook.htm). As an example, open the *cbrain.lsp* file with *Notepad*. Then use the menu commands "File>Save As". A window appears. On the top "Save in" box change what is in the box to "Removable Disk (G:)" in order to save the file on flash drive G. Then in *Arc* activate the *cbrain.lsp* file with the menu commands "File > Load > Removable Disk (G:) > cbrain.lsp."

Alternatively, open *cbrain.lsp* file with *Notepad*. Then use the menu commands "File>Save As". A window appears. On the top "Save in" box change what is in the box to "My Documents". Then go to *Arc* and use the menu commands "File>Load". A window appears. Change "Arc" to "My Documents" and open *cbrain.lsp*.

Downloading the book's R/Splus functions *mpack.txt* into *R* or *Splus*:

Many of the homework problems use *R/Splus* functions contained in the book's website (www.math.siu.edu/olive/mbook.htm) under the file name *mpack.txt*. Suppose that you download *mpack.txt* onto flashdrive G. Enter *R* and wait for the cursor to appear. Then go to the *File* menu and drag down *Source R Code*. A window should appear. Navigate the *Look in* box until it says *Removable Disk (G:)*. In the *Files of type* box choose *All files(*.*)* and then select *mpack.txt*. The following line should appear in the main *R* window.

```
> source("G:/mpack.txt")
```

Type *ls()*. About 70 *R/Splus* functions from *mpack.txt* should appear.

When you finish your *R/Splus* session, enter the command *q()*. A window asking "*Save workspace image?*" will appear. Click on *No* if you do not want to save the programs in *R*. (If you do want to save the programs then click on *Yes*.)

If you use *Splus*, the command

```
> source("G:/mpack.txt")
```

will enter the functions into *Splus*. Creating a special workspace for the functions may be useful.

This section gives tips on using *R/Splus*, but is no replacement for books such as Becker, Chambers, and Wilks (1988), Braun and Murdoch (2007), Crawley (2005, 2007), or Venables and Ripley (2003). Also see Mathsoft (1999ab) and use the website (www.google.com) to search for useful websites. For example enter the search words *R documentation*.

The command *q()* gets you out of *R* or *Splus*.

Least squares regression is done with the function *lsfit*.

The commands *help(fn)* and *args(fn)* give information about the function fn, eg if fn = lsfit.

Type the following commands.

```
x <- matrix(rnorm(300),nrow=100,ncol=3)
y <- x%*%1:3 + rnorm(100)
out<- lsfit(x,y)
out$coef
ls.print(out)
```

The first line makes a 100 by 3 matrix x with N(0,1) entries. The second line makes $y[i] = 0 + 1 * x[i,1] + 2 * x[i,2] + 3 * x[i,2] + e$ where e is N(0,1). The term 1:3 creates the vector $(1,2,3)^T$ and the matrix multiplication operator is $\%*\%$. The function `lsfit` will automatically add the constant to the model. Typing "out" will give you a lot of irrelevant information, but *out$coef* and *out$resid* give the OLS coefficients and residuals respectively.

To make a residual plot, type the following commands.

```
fit <- y - out$resid
plot(fit,out$resid)
title("residual plot")
```

The first term in the plot command is always the horizontal axis while the second is on the vertical axis.

To put a graph in *Word,* hold down the *Ctrl* and *c* buttons simultaneously. Then select "paste" from the *Word* Edit menu.

To enter data, open a data set in *Notepad* or *Word*. You need to know the number of rows and the number of columns. Assume that each case is entered in a row. For example, assuming that the file *cyp.lsp* has been saved on your disk from the webpage for this book, open *cyp.lsp* in *Word.* It has 76 rows and 8 columns. In *R* or *Splus*, write the following command.

```
cyp <- matrix(scan(),nrow=76,ncol=8,byrow=T)
```

Then copy the data lines from *Word* and paste them in *R/Splus*. If a cursor does not appear, hit *enter.* The command *dim(cyp)* will show if you have entered the data correctly.

Enter the following commands

```
cypy <- cyp[,2]
cypx<- cyp[,-c(1,2)]
lsfit(cypx,cypy)$coef
```

to produce the output below.

```
   Intercept           X1           X2           X3           X4
205.40825985   0.94653718   0.17514405   0.23415181   0.75927197
 X5           X6
 -0.05318671  -0.30944144
```

To check that the data is entered correctly, fit LS in *Arc* with the response variable *height* and the predictors *sternal height, finger to ground, head length, nasal length, bigonal breadth,* and *cephalic index* (entered in that order). You should get the same coefficients given by *R* or *Splus*.

Making functions in R and Splus is easy.

For example, type the following commands.

```
mysquare <- function(x){
# this function squares x
r <- x^2
r }
```

The second line in the function shows how to put comments into functions.

Modifying your function is easy.

Use the fix command.

fix(mysquare)

This will open an editor such as *Notepad* and allow you to make changes.

In *Splus*, the command *Edit(mysquare)* may also be used to modify the function *mysquare*.

To save data or a function in *R*, when you exit, click on *Yes* when the "*Save worksheet image?*" window appears. When you reenter *R*, type *ls().* This will show you what is saved. You should rarely need to save anything for the material in the first thirteen chapters of this book. In *Splus*, data and functions are automatically saved. To remove unwanted items from the worksheet, eg x, type *rm(x)*,
pairs(x) makes a scatterplot matrix of the columns of x,
hist(y) makes a histogram of y,
boxplot(y) makes a boxplot of y,
stem(y) makes a stem and leaf plot of y,
scan(), source(), and *sink()* are useful on a *Unix* workstation.
To type a simple list, use $y <-$ *c(1,2,3.5).*
The commands *mean(y), median(y), var(y)* are self explanatory.

The following commands are useful for a scatterplot created by the command *plot(x,y).*
lines(x,y), lines(lowess(x,y,f=.2))
identify(x,y)
abline(out$coef), abline(0,1)

The usual arithmetic operators are $2 + 4$, $3 - 7$, $8 * 4$, $8/4$, and

```
2^{10}.
```

The ith element of vector y is $y[i]$ while the ij element of matrix x is $x[i, j]$. The second row of x is $x[2,]$ while the 4th column of x is $x[, 4]$. The transpose of x is *t(x).*

The command *apply(x,1,fn)* will compute the row means if fn = mean. The command *apply(x,2,fn)* will compute the column variances if fn = var.

The commands *cbind* and *rbind* combine column vectors or row vectors with an existing matrix or vector of the appropriate dimension.

Downloading the book's R/Splus data sets *robdata.txt* into *R* or *Splus* is done in the same way for downloading *rpack.txt*. Use the following command.

```
> source("G:/mrobdata.txt")
```

For example the command

```
> lsfit(belx,bely)
```

will perform the least squares regression for the Belgian telephone data.

Transferring Data to and from *Arc* and *R* or *Splus*.
For example, suppose that the Belgium telephone data (Rousseeuw and Leroy 1987, p. 26) has the predictor *year* stored in x and the response *number of calls* stored in y in *R* or *Splus*. Combine the data into a matrix z and then use the *write.table* command to display the data set as shown below. The

```
 sep=' '
```

separates the columns by two spaces.

```
> z <- cbind(x,y)
> write.table(data.frame(z),sep='   ')
row.names   z.1   y
 1   50   0.44
 2   51   0.47
 3   52   0.47
 4   53   0.59
 5   54   0.66
 6   55   0.73
 7   56   0.81
 8   57   0.88
 9   58   1.06
 10   59   1.2
 11   60   1.35
 12   61   1.49
 13   62   1.61
```

```
 14   63   2.12
 15   64   11.9
 16   65   12.4
 17   66   14.2
 18   67   15.9
 19   68   18.2
 20   69   21.2
 21   70   4.3
 22   71   2.4
 23   72   2.7073
 24   73   2.9
```

To enter a data set into *Arc*, use the following template *new.lsp*.

```
dataset=new
begin description
Artificial data.
Contributed by David Olive.
end description
begin variables
col 0 = x1
col 1 = x2
col 2 = x3
col 3 = y
end variables
begin data
```

Next open *new.lsp* in *Notepad*. (Or use the *vi* editor in Unix. Sophisticated editors like *Word* will often work, but they sometimes add things like page breaks that do not allow the statistics software to use the file.) Then copy the data lines from *R/Splus* and paste them below *new.lsp*. Then modify the file *new.lsp* and save it on a disk as the file *belg.lsp*. (Or save it in *mdata* where *mdata* is a data folder added within the *Arc data* folder.) The header of the new file *belg.lsp* is shown on the next page.

```
dataset=belgium
begin description
Belgium telephone data from
```

```
Rousseeuw and Leroy (1987, p. 26)
end description
begin variables
col 0 = case
col 1 = x = year
col 2 = y = number of calls in tens of millions
end variables
begin data
1 50 0.44
.  .  .
.  .  .
.  .  .
24 73 2.9
```

The file above also shows the first and last lines of data. The header file needs a data set name, description, variable list and a *begin data* command. Often the description can be copied and pasted from source of the data, eg from the STATLIB website. Note that the first variable starts with *Col 0*.

To transfer a data set from Arc to R or Splus, select the item "Display data" from the dataset's menu. Select the variables you want to save, and then push the button for "Save in R/Splus format." You will be prompted to give a file name. If you select *bodfat*, then two files *bodfat.txt* and *bodfat.Rd* will be created. The file *bodfat.txt* can be read into either *R* or *Splus* using the *read.table* command. The file *bodfat.Rd* saves the documentation about the data set in a standard format for *R*.

As an example, the following command was used to enter the body fat data into *Splus*. (The *mdata folder* does not come with *Arc*. The folder needs to be created and filled with files from the book's website. Then the file *bodfat.txt* can be stored in the mdata folder.)

```
bodfat <- read.table("C:\\ARC\\DATA\\MDATA\\BODFAT.TXT",header=T)
bodfat[,16] <- bodfat[,16]+1
```

The last column of the body fat data consists of the case numbers which start with 0 in *Arc*. The second line adds one to each case number.

As another example, use the menu commands
"File>Load>Data>Arcg>forbes.lsp" to activate the forbes data set. From the *Forbes* menu, select *Display Data*. A window will appear. Double click

on *Temp* and *Pressure*. Click on *Save Data in R/Splus Format* and save as *forbes.txt* in the folder *mdata*.

Enter *Splus* and type the following command.

```
forbes<-read.table("C:\\ARC\\DATA\\ARCG\\FORBES.TXT",header=T)
```

The command *forbes* will display the data set.

Getting information about a library in R

In *R*, a *library* is an add–on package of *R* code. The command *library()* lists all available libraries, and information about a specific library, such as `MASS` for robust estimators like `cov.mcd` or `ts` for time series estimation, can be found, eg, with the command *library(help=MASS)*.

Downloading a library into R

Many researchers have contributed a *library* of *R* code that can be downloaded for use. To see what is available, go to the website (http://cran.us.r-project.org/) and click on the Packages icon. Suppose you are interested the Weisberg (2002) dimension reduction library *dr*. Scroll down the screen and click on *dr*. Then click on the file corresponding to your type of computer, eg *dr 2.0.0.zip* for *Windows*. My unzipped files are stored in my directory

```
C:\unzipped.
```

The file

```
C:\unzipped\dr
```

contains a folder *dr* which is the *R library*. Cut this folder and paste it into the *R* library folder. (On my computer, I store the folder *rw1011* in the file

```
C:\R-Gui.
```

The folder

```
C:\R-Gui\rw1011\library
```

contains the library packages that came with *R*.) Open *R* and type the following command.

library(dr)

Next type *help(dr)* to make sure that the library is available for use.

Warning: R is free but not fool proof. If you have an old version of R and want to download a library, you may need to update your version of R. The libraries for robust statistics may be useful for outlier detection, but the methods have not been shown to be consistent or high breakdown. All software has some bugs. For example, Version 1.1.1 (August 15, 2000) of R had a random generator for the Poisson distribution that produced variates with too small of a mean θ for $\theta \geq 10$. Hence simulated 95% confidence intervals might contain θ 0% of the time. This bug seems to have been fixed in Version 2.4.1.

15.3 Projects

Straightforward Projects

- Read Bentler and Yuan (1998) and Cattell (1966). These papers use scree plots to determine how many eigenvalues of the covariance matrix are nonzero. This topic is very important for dimension reduction methods such as principal components.

- Remark 4.1 estimates the percentage of outliers that the FMCD algorithm can tolerate. In Section 4.5, data is generated such that the FMCD estimator works well for $p = 4$ but fails for $p = 8$. Generate similar data sets for $p = 8, 9, 10, 12, 15, 20, 25, 30, 35, 40, 45,$ and 50. For each value of p find the smallest integer valued percentage of outliers needed to cause the FMCD and FCH estimators to fail. Use the `mpack` function `concsim`. If `concsim` is too slow for large p, use `covsim2` which will only give counts for the fast FCH estimator. As a criterion, a count ≥ 16 is good. Compare these observed FMCD percentages with Remark 4.1 (use the `gamper2` function). Do not forget the *library(MASS)* command if you use R.

- DD plots: compare classical–FCH vs classical–cov.mcd DD plots on real and simulated data. Do problems 4.4, 5.2 and 5.3 but with a wider variety of data sets, n, p and gamma.

- Many papers substitute the latest MCD algorithm for the classical estimator and have titles like "Fast and Robust Factor Analysis." Find such a paper that analyzes a data set on

i) factor analysis,

ii) discriminant analysis,

iii) principal components,

iv) canonical correlation analysis,

v) Hotelling's t test, or

vi) principal component regression.

For the data, make a scatterplot matrix of the classical, RFCH and FMCD Mahalanobis distances. Delete any outliers and run the classical procedure on the undeleted data. Did the paper's procedure perform as well as this procedure?

- Examine the DD plot as a diagnostic for multivariate normality and elliptically contoured distributions. Use real and simulated data.

- Resistant regression: modify `tvreg` by using OLS–covfch instead of OLS–cov.mcd. (L_1–cov.mcd and L_1–covfch are also interesting.) Compare your function with `tvreg`. The `tvreg` and `covfch` functions are in *rpack.txt*.

- *Using ESP to Search for the Missing Link*: Compare `trimmed views` which uses OLS and `cov.mcd` with another regression–MLD combo. There are 8 possible projects: i) OLS–FCH, ii) OLS–Classical (use `ctrviews`), iii) SIR–cov.mcd (`sirviews`), iv) SIR–FCH, v) SIR–classical, vi) lmsreg–cov.mcd (`lmsviews`), vii) lmsreg–FCH, and viii) lmsreg –classical. Do Problem 14.3ac (but just copy and paste the best view instead of using the essp(nx,ncuby,M=40) command) with both your estimator and `trimmed views`. Try to see what types of functions work for both estimators, when `trimmed views` is better and when the procedure i)–viii) in better. If you can invent interesting 1D functions, do so. See Problem 14.4.

- Investigate using trimmed views to make various procedures such as sliced inverse regression resistant against the presence of nonlinearities. The functions *sirviews*, *drsim5*, *drsim6* and *drsim7* in *rpack.txt* may be useful.

- The DGK estimator with 66% coverage should be able to tolerate a cluster of about 30% extremely distant outliers. Compare the DGK estimators with 50% and 66% coverage for various outlier configurations.

Harder Projects

- Which estimator is better FCH, RFCH, CMBA or RCMBA?
- For large data sets, make the DD plot of the DGK estimator vs MB estimator and the DD plot of the classical estimator versus the MB estimator. Which DD plot is more useful? Does your answer depend on n and p? These two plots are among the fastest outlier diagnostics for multivariate data.
- *The Super Duper Outlier Scooper for Multivariate Location and Dispersion:* Consider the modified MBA estimator for multivariate location and dispersion given in Problem 4.7. This MBA estimator uses 8 starts using 0%, 50%, 60%, 70%, 80%, 90%, 95% and 98% trimming of the cases closest to the coordinatewise median in Euclidean distance. The estimator is $\sqrt{n}$ consistent on elliptically contoured distributions with nonsingular covariance matrix. For small data sets the *cmba2* function can fail because the covariance estimator applied to the closest 2% cases to the coordinatewise median is singular. Modify the function so that it works well on small data sets. Then consider the following proposal that may make the estimator asymptotically equivalent to the classical estimator when the data are from a multivariate normal (MVN) distribution. The attractor corresponding to 0% trimming is the DGK estimator $(\hat{\boldsymbol{\mu}}_0, \hat{\boldsymbol{\Sigma}}_0)$. Let $(\hat{\boldsymbol{\mu}}_T, \hat{\boldsymbol{\Sigma}}_T) = (\hat{\boldsymbol{\mu}}_0, \hat{\boldsymbol{\Sigma}}_0)$ if $det(\hat{\boldsymbol{\Sigma}}_0) \leq det(\hat{\boldsymbol{\Sigma}}_M)$ and $(\hat{\boldsymbol{\mu}}_T, \hat{\boldsymbol{\Sigma}}_T) = (\hat{\boldsymbol{\mu}}_M, \hat{\boldsymbol{\Sigma}}_M)$ otherwise where $(\hat{\boldsymbol{\mu}}_M, \hat{\boldsymbol{\Sigma}}_M)$ is the attractor corresponding to M% trimming. Then make the DD plot of the classical Mahalanobis distances versus the distances corresponding to $(\hat{\mu}_T, \hat{\boldsymbol{\Sigma}}_T)$ for $M = 50, 60, 70, 80, 90, 95$ and 98. If all seven DD plots "look good" then use the classical estimator. The resulting estimator will be asymptotically equivalent to the classical estimator if P(all seven DD plots "look good") goes to one as $n \to \infty$. We conjecture that all seven plots will look good because if n is large and the trimmed attractor "beats" the DGK estimator, then the plot will look good. Also if the data is MVN but not spherical, then the DGK estimator will almost always "beat" the trimmed estimator, so all 7 plots will be identical.

- The TV estimator for MLR has a good combination of resistance and theory. Consider the following modification to make the method asymptotically equivalent to OLS when the Gaussian model holds: if each trimmed view "looks good," use OLS. The method is asymptotically equivalent to OLS if the probability P(all 10 trimmed views look good) goes to one as $n \to \infty$. Rousseeuw and Leroy (1987, p. 128) shows that if the predictors are bounded, then the ith residual r_i converges in probability to the ith error e_i for $i = 1, ..., n$. Hence all 10 trimmed views will look like the OLS view with high probability if n is large.

- Compare outliers and missing values, especially missing and outlying at random. See Little and Rubin (2002).

- Suppose that the data set contains missing values. Code the missing value as $\pm 99999 +$ rnorm(1). Run a robust procedure on the data. The idea is that the case with the missing value will be given weight zero if the variable is important, and the variable will be given weight zero if the case is important. See Hawkins and Olive (1999b).

- Download the `dr` function for R, (contributed by Sanford Weisberg), and make PHD and SAVE trimmed views.

- Implement the Carroll and Pederson (1993) robust logistic regression estimator using the robust MLD estimator RFCH or RMVN and see how well the estimator works.

Research Ideas that have Confounded the Author

- If the attractor of a randomly selected elemental start is (in)consistent, then FMCD is (in)consistent. Hawkins and Olive (2002) showed that the attractor is inconsistent if k concentration steps are used. Suppose K elemental starts are used for an MCD concentration estimator and that the starts are iterated until convergence instead of for k steps. Prove or disprove the conjecture that the resulting estimator is inconsistent. (Intuitively, the elemental starts are inconsistent and hence are tilted away from the parameter of interest. Concentration may reduce but probably does not eliminate the tilt.)

- Prove or disprove Conjectures 4.1, 4.2, and 4.3.

- Prove or disprove Conjecture 5.1. Do elemental set and concentration algorithms for multivariate location and dispersion (MLD) give consistent estimators if the number of starts increases to ∞ with the sample size n? (Algorithms that use a fixed number of elemental sets along with the classical estimator and a biased but easily computed high breakdown estimator will be easier to compute and have better statistical properties. See Theorem 4.9 and Olive and Hawkins, 2007, 2008.)

  It is easy to create consistent algorithm estimators that use $O(n)$ randomly chosen elemental sets. He and Wang (1997) show that the all elemental subset approximation to S estimators for MLD is consistent for $(\boldsymbol{\mu}, c\boldsymbol{\Sigma})$. Hence an algorithm that randomly draws $g(n)$ elemental sets and searches all $C(g(n), p+1)$ elemental sets is also consistent if $g(n) \to \infty$ as $n \to \infty$. For example, $O(n)$ elemental sets are used if $g(n) \propto n^{1/(p+1)}$.

  When a fixed number of K elemental starts are used, the best attractor is inconsistent but gets close to $(\boldsymbol{\mu}, c_{MCD}\boldsymbol{\Sigma})$ if the data distribution is EC. (The estimator may be unbiased but the variability of the component estimators does not go to 0 as $n \to \infty$.) If $K \to \infty$, then the best attractor should approximate the highest density region arbitrarily closely and the algorithm should be consistent. However, the time for the algorithm greatly increases, the convergence rate is very poor (possibly between $K^{1/2p}$ and $K^{1/p}$), and the elemental concentration algorithm can not guarantee that the determinant is bounded when outliers are present.

- A promising two stage estimator is the "cross checking estimator" that uses a standard consistent estimator and an alternative consistent estimator with desirable properties such as a high breakdown value. The final estimator uses the standard estimator if it is "close" to the alternative estimator, and hence is asymptotically equivalent to the standard estimator for clean data. One important area of research for robust statistics is finding good computable consistent robust estimators to be used in plots and in the cross checking algorithm. The estimators given in Theorems 4.8 and 4.9 (see Olive 2004a and Olive and Hawkins 2007, 2008) finally make the cross checking estimator practical, but better estimators are surely possible. He and Wang (1996) suggested

the cross checking idea for multivariate location and dispersion.

15.4 Hints for Selected Problems

Chapter 1

1.1 a) $8.25 \pm 0.7007 = (6.020, 10.480)$

b) $8.75 \pm 1.1645 = (7.586, 9.914)$.

1.2 a) $\overline{Y} = 24/5 = 4.8$.

b)

$$S^2 = \frac{138 - 5(4.8)^2}{4} = 5.7$$

so $S = \sqrt{5.7} = 2.3875$.

c) The ordered data are 2,3,5,6,8 and $\text{MED}(n) = 5$.

d) The ordered $|Y_i - \text{MED}(n)|$ are 0,1,2,2,3 and $\text{MAD}(n) = 2$.

1.2 a) $\overline{Y} = 15.8/10 = 1.58$.

b)

$$S^2 = \frac{38.58 - 10(1.58)^2}{9} = 1.5129$$

so $S = \sqrt{1.5129} = 1.230$.

c) The ordered data set is 0.0,0.8,1.0,1.2,1.3,1.3,1.4,1.8,2.4,4.6 and $\text{MED}(n) = 1.3$.

d) The ordered $|Y_i - \text{MED}(n)|$ are 0,0,0.1,0.1,0.3,0.5,0.5,1.1,1.3,3.3 and $\text{MAD}(n) = 0.4$.

e) 4.6 is unusually large.

Chapter 2

Chapter 3

3.1 a) $X_2 \sim N(100, 6)$.

b)

$$\begin{pmatrix} X_1 \\ X_3 \end{pmatrix} \sim N_2 \left(\begin{pmatrix} 49 \\ 17 \end{pmatrix}, \begin{pmatrix} 3 & -1 \\ -1 & 4 \end{pmatrix} \right).$$

c) $X_1 \perp\!\!\!\perp X_4$ and $X_3 \perp\!\!\!\perp X_4$.

d)

$$\rho(X_1, X_2) = \frac{Cov(X_1, X_3)}{\sqrt{\text{VAR}(X_1)\text{VAR}(X_3)}} = \frac{-1}{\sqrt{3}\sqrt{4}} = -0.2887.$$

3.2 a) $Y|X \sim N(49, 16)$ since $Y \perp\!\!\!\perp X$. (Or use $E(Y|X) = \mu_Y + \Sigma_{12}\Sigma_{22}^{-1}(X - \mu_x) = 49 + 0(1/25)(X - 100) = 49$ and $\text{VAR}(Y|X) = \Sigma_{11} - \Sigma_{12}\Sigma_{22}^{-1}\Sigma_{21} = 16 - 0(1/25)0 = 16$.)

b) $E(Y|X) = \mu_Y + \Sigma_{12}\Sigma_{22}^{-1}(X - \mu_x) = 49 + 10(1/25)(X - 100) = 9 + 0.4X$.

c) $\text{VAR}(Y|X) = \Sigma_{11} - \Sigma_{12}\Sigma_{22}^{-1}\Sigma_{21} = 16 - 10(1/25)10 = 16 - 4 = 12$.

3.4 The proof is identical to that given in Example 3.2. (In addition, it is fairly simple to show that $M_1 = M_2 \equiv M$. That is, M depends on Σ but not on c or g.)

3.6 a) Sort each column, then find the median of each column. Then $\text{MED}(\boldsymbol{W}) = (1430, 180, 120)^T$.

b) The sample mean of $(X_1, X_2, X_3)^T$ is found by finding the sample mean of each column. Hence $\overline{\boldsymbol{x}} = (1232.8571, 168.00, 112.00)^T$.

3.11 $\boldsymbol{\Sigma B} = E[E(\boldsymbol{X}|\boldsymbol{B}^T\boldsymbol{X})\boldsymbol{X}^T\boldsymbol{B})] = E(\boldsymbol{M}_B\boldsymbol{B}^T\boldsymbol{X}\boldsymbol{X}^T\boldsymbol{B}) = \boldsymbol{M}_B\boldsymbol{B}^T\boldsymbol{\Sigma B}$. Hence $\boldsymbol{M}_B = \boldsymbol{\Sigma B}(\boldsymbol{B}^T\boldsymbol{\Sigma B})^{-1}$.

Chapter 4

4.4 The 4 plots should look nearly identical with the five cases 61–65 appearing as outliers.

4.5 Not only should none of the outliers be highlighted, but the highlighted cases should be ellipsoidal.

4.6 Answers will vary since this is simulated data, but should get gam near 0.4, 0.3, 0.2 and 0.1 as p increases from 2 to 20.

Chapter 5

5.2 b Ideally the answer to this problem and Problem 5.3b would be nearly the same, but students seem to want correlations to be very high and use n too high. Values of n around 20, 40 and 50 for $p = 2, 3$ and 4 should be enough.

5.3 b Values of n should be near 20, 40 and 50 for $p = 2, 3$ and 4.

5.4 This is simulated data, but for most plots the slope is near 2.

Chapter 6

6.1 Note that $o_P(1)O_P(1) = [(\hat{\mathbf{\Sigma}} - \hat{\lambda}_i) - c(\mathbf{\Sigma} - \lambda_i)]\hat{\boldsymbol{e}}_i = c(\mathbf{\Sigma} - \lambda_i)\hat{\boldsymbol{e}}_i \xrightarrow{P} \mathbf{0}$.

Chapter 7

Chapter 8

Chapter 9

Chapter 10

Chapter 11

Chapter 12

Chapter 13

Chapter 14

14.6 The identity line should NOT PASS through the cluster of outliers with Y near 0. The amount of trimming seems to vary some with the computer (which should not happen unless there is a bug in the `tvreg2` function or if the computers are using different versions of `cov.mcd`), but most students liked 70% or 80% trimming.

15.5 F Table

Tabled values are F(0.95,k,d) where $P(F < F(0.95, k, d)) = 0.95$.
00 stands for ∞. Entries produced with the `qf(.95,k,d)` command in *R*. The numerator degrees of freedom are k while the denominator degrees of freedom are d.

k	1	2	3	4	5	6	7	8	9	00
d										
1	161	200	216	225	230	234	237	239	241	254
2	18.5	19.0	19.2	19.3	19.3	19.3	19.4	19.4	19.4	19.5
3	10.1	9.55	9.28	9.12	9.01	8.94	8.89	8.85	8.81	8.53
4	7.71	6.94	6.59	6.39	6.26	6.16	6.09	6.04	6.00	5.63
5	6.61	5.79	5.41	5.19	5.05	4.95	4.88	4.82	4.77	4.37
6	5.99	5.14	4.76	4.53	4.39	4.28	4.21	4.15	4.10	3.67
7	5.59	4.74	4.35	4.12	3.97	3.87	3.79	3.73	3.68	3.23
8	5.32	4.46	4.07	3.84	3.69	3.58	3.50	3.44	3.39	2.93
9	5.12	4.26	3.86	3.63	3.48	3.37	3.29	3.23	3.18	2.71
10	4.96	4.10	3.71	3.48	3.33	3.22	3.14	3.07	3.02	2.54
11	4.84	3.98	3.59	3.36	3.20	3.09	3.01	2.95	2.90	2.41
12	4.75	3.89	3.49	3.26	3.11	3.00	2.91	2.85	2.80	2.30
13	4.67	3.81	3.41	3.18	3.03	2.92	2.83	2.77	2.71	2.21
14	4.60	3.74	3.34	3.11	2.96	2.85	2.76	2.70	2.65	2.13
15	4.54	3.68	3.29	3.06	2.90	2.79	2.71	2.64	2.59	2.07
16	4.49	3.63	3.24	3.01	2.85	2.74	2.66	2.59	2.54	2.01
17	4.45	3.59	3.20	2.96	2.81	2.70	2.61	2.55	2.49	1.96
18	4.41	3.55	3.16	2.93	2.77	2.66	2.58	2.51	2.46	1.92
19	4.38	3.52	3.13	2.90	2.74	2.63	2.54	2.48	2.42	1.88
20	4.35	3.49	3.10	2.87	2.71	2.60	2.51	2.45	2.39	1.84
25	4.24	3.39	2.99	2.76	2.60	2.49	2.40	2.34	2.28	1.71
30	4.17	3.32	2.92	2.69	2.53	2.42	2.33	2.27	2.21	1.62
00	3.84	3.00	2.61	2.37	2.21	2.10	2.01	1.94	1.88	1.00