Quiz #4

To help illustrate the tools presented in this chapter, we apply many of the entries to the HELP data. First read in the HELP CSV file from Bb

Scatterplot with multiple axes

The following example creates a single figure that displays the relationship between CESD and the variables `indtot` (Inventory of Drug Abuse Consequences, InDUC) and `mcs` (Mental Component Score) for a subset of female alcohol-involved subjects. We specify two different y axes for the figure.

```
axis1 minor=none;
axis2 minor=none order=(5 to 60 by 13.625);
axis3 minor=none order=(20, 40, 60);
symbol1 i=sm65s v=circle color=black l=1 w=5;
symbol2 i=sm65s v=triangle color=black l=2 w=5;
proc gplot data=ds;
     where female eq 1 and substance eq 'alcohol';
     plot indtot*cesd / vaxis=axis1 haxis=axis3;
     plot2 mcs*cesd / vaxis = axis2;
run;
quit;
```

In the SAS code above, the symbol and axis statements are used to control the output and to add lines through the data. Note that three axes are specified and are associated with the various axes in the plot in the vaxis and haxis options to the plot and plot2 statements. The axis statements can be omitted for a simpler graphic.

In R, a considerable amount of housekeeping is needed. The second y variable must be rescaled to the range of the original, and the axis labels and tick marks added on the right. To accomplish this, we write a function `plottwoy()`, which first makes the plot of the first (left axis) y against x, adds a lowess curve through that data, then calls a second function, `addsecondy()`.

```
> plottwoy = function(x, y1, y2, xname="X", y1name="Y1", y2name="Y2"){
     plot(x, y1, ylab=y1name, xlab=xname)
     lines(lowess(x, y1), lwd=3)
     addsecondy(x, y2, y1, yname=y2name)
}
```

The function `addsecondy()` does the work of rescaling the range of the second variable to that of the first, adds the right axis, and plots a lowess curve through the data for the rescaled y2 variable.

```
> addsecondy = function(x, y, origy, yname="Y2") {
     prevlimits = range(origy)
     axislimits = range(y)
     axis(side=4, at=prevlimits[1] + diff(prevlimits)*c(0:5)/5,
          labels=round(axislimits[1] + diff(axislimits)*c(0:5)/5, 1))
     mtext(yname, side=4)
     newy = (yaxislimits[1])/(diff(axislimits)/diff(prevlimits)) +
          prevlimits[1]
     points(x, newy, pch=2)
     lines(lowess(x, newy), lty=2, lwd=3)
 }
```

Finally, the newly defined functions can be run

```
 > with(ds, plottwoy(cesd[female==1&substance=="alcohol"],
     indtot[female==1&substance=="alcohol"],
     mcs[female==1&substance=="alcohol"], xname="cesd",
     y1name="indtot", y2name="mcs"))
```

Note that the two graphics are not identical due to different y axes. In SAS it is difficult to select axis ranges exactly conforming to the range of the data, while our R function uses more of the space for data display

Conditioning plot

Let's create a conditioning plot with the association between MCS and CESD stratified by substance and report of suicidal thoughts (g1b).

```
proc sgpanel data=ds;
     panelby g1b substance / layout=lattice;
     pbspline x=cesd y=mcs;
run; quit;
```

For R, ensure that the necessary packages are loaded (B.6.1).

```
> library(lattice)
```

Then we can set up and generate the plot.

```
> ds = transform(ds, suicidal.thoughts = ifelse(g1b==1, "Y", "N"))
> coplot(mcs ~ cesd | suicidal.thoughts*substance, panel=panel.smooth,
data=ds)
```

There is a similar association between CESD and MCS for each of the substance groups. Subjects with suicidal thoughts tended to have higher CESD scores, and the association between CESD and MCS was somewhat less pronounced than for those without suicidal thoughts.

Scatterplot with marginal histograms

We can assess the univariate as well as bivariate distribution of the MCS and CESD scores using a scatterplot with a marginal histogram.

```
%include "c:\sta4133\scatterhist.sas"
proc sgrender data=ds template=scatterhist;
     dynamic YVAR="PCS" XVAR="MCS";
run;
```

The R implementation utilizes the layout() function to create the graphic.

```
> scatterhist = function(x, y, xlab="x label", ylab="y label"){
      zones=matrix(c(3,1,2,4), ncol=2, byrow=TRUE)
      layout(zones, widths=c(4/5,1/5), heights=c(1/5,4/5))
      par(mar=c(0,0,0,0))
      plot(type="n",x=1, y =1, bty="n",xaxt="n", yaxt="n")
      text(x=1,y=1,paste0("nobs = ",min(length(x), length(y))), cex
=1.8)
```

```
     xhist = hist(x, plot=FALSE)
     yhist = hist(y, plot=FALSE)
     top = max(c(xhist$counts, yhist$counts))
     par(mar=c(3,3,1,1))
     plot(x,y)
     par(mar=c(0,3,1,1))
     barplot(xhist$counts, axes=FALSE, ylim=c(0, top), space=0)
     par(mar=c(3,0,1,1))
     barplot(yhist$counts, axes=FALSE, xlim=c(0, top), space=0,
horiz=TRUE)
     par(oma=c(3,3,0,0))
     mtext(xlab, side=1, line=1, outer=TRUE, adj=0,
          at=.8 * (mean(x) - min(x))/(max(x)-min(x)))
     mtext(ylab, side=2, line=1, outer=TRUE, adj=0,
          at=(.8 * (mean(y) - min(y))/(max(y) - min(y)))) }
> with(ds, scatterhist(mcs, pcs, xlab="MCS", ylab="PCS"))
```

Kaplan–Meier plot

The main outcome of the HELP study was time to linkage to primary care, as a function of randomization group. This can be displayed using a Kaplan–Meier plot. For SAS, detailed information regarding the Kaplan–Meier estimator at each time point can be found by omitting the `ods select` statement

```
ods select censoredsummary survivalplot;
proc lifetest data=ds plots=s(test);
      time dayslink*linkstatus(0);
      strata treat;
run;
```

For R by using `summary(survobj).`

```
> plot(survobj, lty=1:2, lwd=2, col=c(4,2))
> title("Product-Limit Survival Estimates")
> legend(250, .75, legend=c("Control", "Treatment"), lty=c(1,2),
      lwd=2, col=c(4,2), cex=1.4)
```

There is a highly statistically significant effect of treatment, with approximately 55% of clinic subjects linking to primary care, as opposed to only 15% of control subjects.

ROC curve

Receiver operating characteristic (ROC) curves are used for diagnostic agreement as well as assessing goodness of fit for logistic regression. In SAS, they can be created using `proc logistic`.

```
ods graphics on;
ods select roccurve;
      proc logistic data=ds descending plots(only)=roc;
      model g1b = cesd;
run;
ods graphics off;
```

The `descending` option changes the behavior of `proc logistic` to model the probability that the outcome is 1; the default models the probability that the outcome is 0.

Using R, we first load the `ROCR` package, create a prediction object, and retrieve the area under the curve (AUC).

```
> library(ROCR)
> pred = with(ds, prediction(cesd, g1b))
> auc = slot(performance(pred, "auc"), "y.values")[[1]]
```

We can then plot the ROC curve, adding a display of cutoffs for particular CESD values ranging from 20 to 50. These values are offset from the ROC curve using the text.adj option. If the continuous variable (in this case cesd) is replaced by the predicted probability from a logistic regression model, multiple predictors can be included.

```
> plot(performance(pred, "tpr", "fpr"), print.cutoffs.at=seq(from=20,
to=50, by=5), text.adj=c(1, -.5), lwd=2)
> lines(c(0, 1), c(0, 1)) > text(.6, .2, paste("AUC=", round(auc,3),
sep=""), cex=1.4)
> title("ROC Curve for Model")
```

Pairs plot

We can qualitatively assess the associations between some of the continuous measures of mental health, physical health, and alcohol consumption using a pairsplot or scatterplot matrix. To make the results clearer, we display only the female subjects.

For SAS, the sgscatter procedure provides a simple way to produce this.

```
proc sgscatter data=ds;
     where female eq 1;
     matrix cesd mcs pcs i1 / diagonal=(histogram kernel);
run; quit;
```

If fitted curves in the pairwise scatterplots are required, the following code will produce a similar matrix, with LOESS curves in each cell and less helpful graphs in the diagonals.

```
proc sgscatter data=ds;
     where female eq 1;
     compare x = (cesd mcs pcs i1)
             y = (cesd mcs pcs i1) / loess;
run; quit;
```

For complete control of the figure, the `sgscatter` procedure will not suffice and more complex coding is necessary; we would begin with SAS macros written by Michael Friendly and available from his web site at York University (http://www.datavis.ca).

For R, a simple version with only the scatterplots could be generated easily with the pairs() function:

```
> pairs(ds[c("pcs", "mcs", "cesd", "i1")])
```

There is an indication that CESD, MCS, and PCS are interrelated, while I1 appears to have modest associations with the other variables

Visualize correlation matrix

One visual analysis which might be helpful to display would be the pairwise correlations. We approximate this in SAS by plotting a confidence ellipse for the observed data. This approach allows an assessment of whether the linear correlation is an appropriate statistic to consider.

In the code below, we demonstrate some options for the sgscatter procedure. The ellipse option draws confidence ellipses at the requested α-level, here chosen arbitrarily to mimic R. The start option also mimics R by making the diagonal begin in the lower left; the top left is the default. The markerattrs option controls aspects of the appearance of plots generated with the sgscatter, sgpanel, and sgplot procedures.

```
proc sgscatter data=ds;
     matrix mcs pcs pss_fr drugrisk cesd indtot i1 sexrisk /
     ellipse=(alpha=.25) start=bottomleft
     markerattrs=(symbol=circlefilled size=4);
run; quit;
```

In R, the `corrplot` package is a graphical display of a correlation matrix, confidence interval. It also contains some algorithms to do matrix reordering. In addition, `corrplot` is good at details, including choosing color, text labels, color labels, layout, etc. Check it out here: https://cran.r-project.org/web/packages/corrplot/vignettes/corrplot-intro.html

```
> myvars<-
c("mcs","pcs","pss_fr","drugrisk","cesd","indtot","i1","sexrisk")
> help.sub<-help[myvars]
> M<-cor(help.sub)
> corrplot(M,method="ellipse")
> # you can go crazy
> corrplot(M,method="ellipse", type="upper", order="AOE")
```

The plots suggests that some of these linear correlations might not be useful measures of association, there is a consistent frame of reference for the many correlations.