Assignment 3

Upload today’s practical to your T-Drive

Deadline(Next Tuesday:23October)

**Practical5- Histograms, Boxplots and Stemleaf**

16/10/18

Show all R commands and outputs- type in the commands below and add new commands where requested. Commands are in black.

1) Histograms

a) Create data for the graph and make a histogram that you save.

set.seed(1) this will set it up so that results are reproducible.

v = rnorm(50,0,1) sample of 50 with mean 0 and standard deviation 1.

This is the default- i.e.rnorm(50) is the same as rnorm(50,0,1).

Give the histogram file a name (before calling the histogram function). Plots are usually named before actually doing them. You can set the command to name more than 1 plot, and you can choose the output format (pdf, jpg etc.).

png(file = 'histogram.png')

Create the histogram. Make the colour of the histogram yellow.

hist(v,col = 'yellow')

Save the file.

This command changes graph output back to the default (i.e. your graph window).

dev.off()

b) Make a new histogram as follows:

Redo the above histogram with these changes-

Give this new chart file the name 'histogram\_lim\_breaks.png'.

This time make the colour green, and give it a yellow border (using ‘border= yellow’).

Make the x limits -5 to 5; make the y limits 0 to 25.

Put 5 breaks in the histogram using the 'breaks=’ argument.

Remember to save the file.

c) Now type in set.seed(1) and then create a variable x which has 500 random values from the standard normal distribution. Make a boxplot like the one in histogram\_lim\_breaks.png with ylim changed so that the frequencies are all shown. Name it hist500.png and save it.

You can get the ylim that you need using trial and error - or by running the histogram without ylim and seeing what the y-axis is like. Also- FYI- you can get the actual counts for each breakpoint as follows:

hh=hist(x,breaks=5)

hh$counts

2)Boxplots

Read the ‘stations.csv’ dataset from the st1050 folder.

These are temperatures from are 4 weather stations. Each Station has three separate readings (so Stat11 is Station 1's first reading; Stat12 is the second reading there etc.)

Make a boxplot of the whole dataset.

boxplot(data)

Are all the names appearing on the x-axis?

Sometimes there isn’t room to put all the names horizontally, so add vertical names:

Use the argument las=2 to make the x labels vertical.

boxplot(data, las = 2)

(Can you see all the names?

This view sometimes can cause problems with the size of the plot.

If the names are cut off changing the margins will often fix the problem:

Try changing the margins as suggested here (whether you have a problem or not).

par(mar=c(10,3,10,3))

play with the numbers in that command until you get the

entire plot showing and you are happy with the spacing etc.

Be sure to have all names showing in your plot. This may vary from computer to computer- particularly on laptops.)

Add space between days for clarity by specifying where each boxplot is drawn. Use the

'at' argument, and leave out the numbers where you want spaces as is done here:

boxplot(data, las = 2, at =c(1,2,3,4, 6,7,8,9, 11,12,13,14))

add colour: colour each station differently - but keep the same colour for all 3 readings at that station.

boxplot(data, las = 2, at =c(1,2,3,4, 6,7,8,9, 11,12,13,14),col = rep(c(1:4),3))

3) Another Boxplot- Probably for Homework:

Using the ‘stations’ data make a new histogram that has the weather stations grouped together, with space in between each of them. i.e. There will be 4 groups on your boxplot - each group will have the three days of data for the one station.

So for Station 1 it will have 3 days grouped together- Stat11, Stat12, and Stat13.

Colour each weather station a different colour (4 colours altogether).

Hint: The easiest way to do this is probably to make a new dataframe which has the columns

in the correct order- data2=cbind(data[,1],data[,5] etc.