

R and Rcmdr : Basic Functions for Managing Data

Key issues in using R for a data analysis:

- Difference between numeric variables and factors in R/Rcmdr
- Load data either by entering manually, or by importing from another format (SPSS,Excel,SAS, Stata, ascii etc....)
- It is possible to have multiple data sets open at any time. As a result, each data set must be given a unique name by which it can be referred to and identified with.
- You can switch between loaded datasets
- You can generate summary statistics and graphics quickly and easily, saving time for thinking about what these numbers and graphics tell you about patterns in the data.

Manual input of data using the R spreadsheet (only for the smallest sets of data--otherwise use something like Excel and import the Excel spreadsheet)

1. Select the Data menu

2. Select the New data set... submenu

The New Data Set dialog box will appear.

3. Enter a name for the data set

4. Click the OK button

The R DataEntryWindow (R's graphical spreadsheet) window will appear within RGui. Switch control to Rgui using either Alt-tab or the Windows navigation buttons.

5. Clicking on a column heading and selecting Change Name from the resulting pop-up menu enables variable names to be customized.

6. Data are added by entering values in the cells

7. Close the R DataEntryWindow window and the dataset will be created. You will notice that the Data set panel now displays the name of the newly generated dataset.

Importing data form another format

1. Select the Data menu

2. Select the Import data.. submenu

3. Select the from Excel data set.. or from SPSS data set, etc.... submenu

The Import Excel data set or Import SPSS data set, etc... dialog box will appear.

4. Enter a unique name to be assigned to the imported data set. Remember that while this can be any name (and doesn't necessarily need to be the same as the name of the imported file), a name that describes the data set is recommended.

5. Keep any other default options and click the OK button

6. Locate the file you wish to import and click the OK button.

The data should now be ready to use.

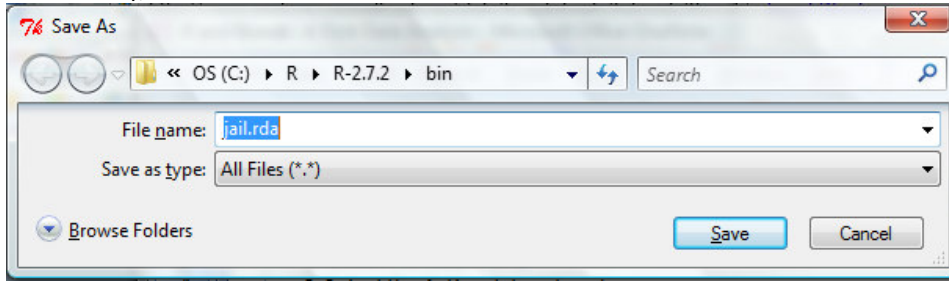
7. To view the data set, click on the View data set button from the main Rgui window.

Saving a data file (in R format)

1. Select the Data menu

2. Select the Active data set.. submenu

3. Select the Save Active Dataset.. Submenu
4. Select a name and location for the R-formatted dataset (example of dialog box below), and select Save.

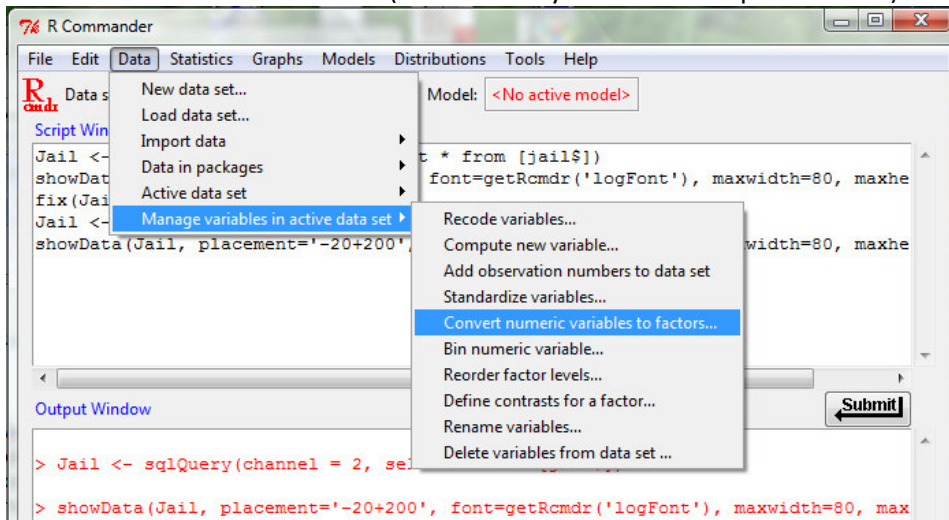


Converting numeric variable to a factor

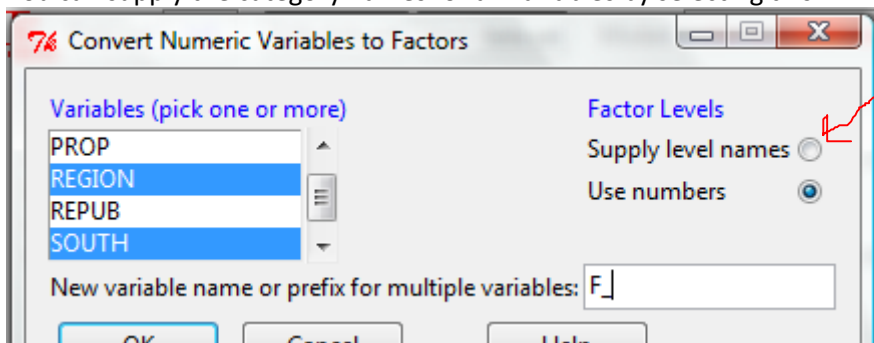
Generally, factors (categories) are entered as words. When this is the case R automatically recognizes the variable as a factor and therefore a categorical (rather than quantitative) variable. However, occasionally the levels of a categorical variable may be numbers. For example, you might have a coded race with numbers rather than names. In this case, your factor levels are 1,2,3,4 for White, black, Hispanic and other respectively. However, as these are numbers (rather than words), R will not automatically consider the variable as a category. It is possible, however, to convert such a numeric variable into a factor variable, and you should do so before you begin analyses.

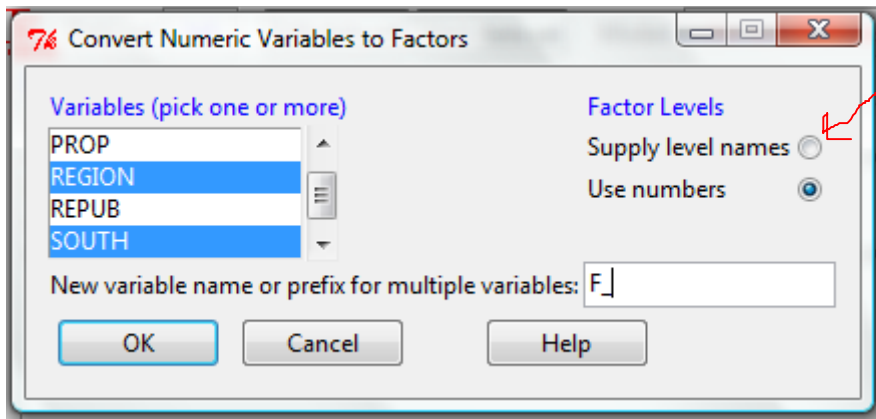
To do this:

1. Select the Data menu
2. Select the Manage variables.. Submenu
3. Select Convert numeric variables to factors
4. Select one or more variables (hold CTRL key to select multiple variables).



You can supply the category names for all variables by selecting this





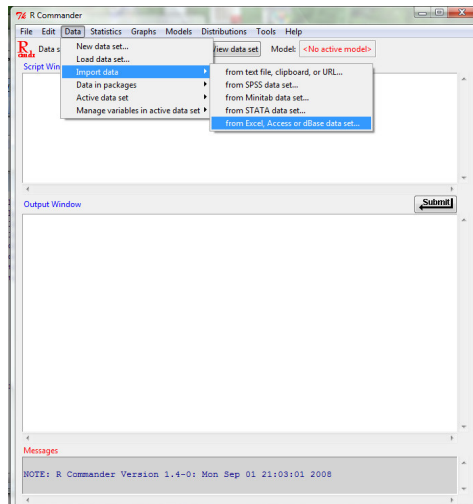
Here, I add a prefix of "F_" to the two selected variables, rather than overwrite the original names. It's always a good idea as a new data analyst to NOT overwrite anything. You can always delete extra variables once you confirm the recoding is correct.

Switching between different loaded data sets

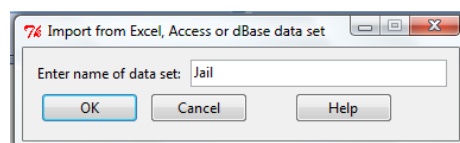
1. Click on the Data set display panel in the Rgui window
The Select Data Set dialog box will be displayed
2. Select the required data set
3. Click OK

R Basic Univariate Summary Techniques: The "Jail" Example

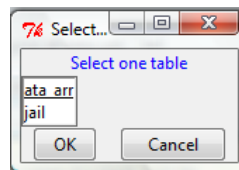
Importing data from an Excel spreadsheet:
Select Data-Import data-from Excel...



Name your dataset with an informative name.
R can have multiple datasets open at the same time, so consider a name that will easily identify a "working" dataset.



An Excel workbook can have multiple sheets-Rcmdr asks you to select which sheet in the Workbook you want. Here, we select "jail".



NOTE: You can select the "View data set" button just below the menu toolbar to see the actual rectangular dataset.

STEP 1: Look at the data: Note the NAs at the bottom of the dataset. There are numerous "observations" with all NAs (missing data)--we should delete these b/c they are errors in the import process.

INC	AFDC	PROP	UNEMP	VIOL	HOMI	REPUB	BLACK	POVERTY	SOUTH	REGION	STATE	
22	148	278.7204	3.7580	7.1	0.2570	4.3	63.0400	0.7591	10.70	0	4	WY
23	149	401.4144	4.5500	5.7	0.5450	5.4	45.8300	12.9847	9.10	0	4	NM
24	149	139.7233	3.6920	8.0	0.4740	9.0	39.4300	15.8818	18.25	1	2	TN
25	156	394.3957	6.1030	8.1	0.4250	5.2	55.1000	2.8227	10.80	0	4	WA
26	161	298.9165	4.5970	9.0	0.7030	8.0	50.8400	14.8514	13.00	0	3	IN
27	165	364.5563	6.1790	8.8	0.5510	4.7	45.0000	1.5126	10.35	0	4	OR
28	175	223.7917	3.6050	7.9	0.3090	5.8	65.3300	7.6746	12.40	0	3	IA
29	181	485.7995	5.7530	7.2	0.7650	10.5	40.8300	7.5527	12.45	0	4	CA
30	192	291.4502	4.0190	5.0	0.3560	4.9	76.8000	5.5500	9.85	0	3	KS
31	194	236.4950	3.8630	6.4	0.5040	8.1	31.4700	10.5768	13.20	0	2	MS
32	194	277.1772	4.3758	8.9	0.3820	5.2	47.7200	10.3070	10.65	0	3	OH
33	195	437.8222	4.6590	6.5	0.9300	9.5	46.6600	14.7852	14.05	0	3	ND
34	195	160.6710	5.4510	8.7	0.3480	7.9	5.9300	16.1302	20.55	1	4	AZ
35	196	407.4815	5.6320	9.9	0.7340	11.2	40.5400	13.4177	13.60	0	3	MI
36	204	231.2316	3.4840	5.6	0.2950	7.1	30.0000	18.8302	11.75	1	1	VT
37	226	129.2624	6.0190	7.0	0.5500	13.0	23.2000	11.9589	15.80	1	2	TX
38	237	99.3047	2.9950	10.3	0.2710	10.6	4.5900	35.3722	25.00	1	3	MO
39	247	206.8014	6.6330	6.0	0.9410	11.4	32.5000	13.6893	15.55	1	2	FL
40	250	258.4320	5.0030	7.1	0.4220	7.7	26.1700	7.1034	14.75	0	2	OK
41	251	182.0811	4.6030	6.5	0.5070	10.4	11.8600	26.8934	14.85	1	2	GA
42	254	190.1802	3.7010	5.4	0.4210	2.9	20.0000	22.1922	14.00	1	3	NE
43	256	208.1074	6.5140	6.5	0.6030	8.0	65.5500	2.8897	13.25	0	2	AR
44	267	110.7415	3.4850	8.9	0.4580	9.8	2.8500	25.4189	20.20	1	4	AK
45	279	279.8950	4.5380	4.6	0.4330	7.9	11.7000	23.8038	9.70	1	2	MD
46	281	249.4489	4.5280	5.3	0.4330	4.8	54.8300	16.4858	9.35	0	2	DE
47	288	466.8834	5.2950	9.7	0.5820	9.8	41.6600	3.7413	10.50	0	2	AL
48	294	142.5859	4.2100	6.8	0.6310	9.1	12.9400	30.1086	16.50	1	2	SC
49	308	153.6926	4.8690	11.5	0.6940	10.9	6.9400	30.1071	21.95	1	2	LA
50	397	231.9572	5.9080	8.0	0.6670	10.3	31.6600	6.4725	9.05	0	2	NC
51	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	<NA>
52	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	<NA>
53	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	<NA>
54	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	<NA>
55	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	<NA>
56	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	<NA>
57	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	<NA>
58	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	<NA>

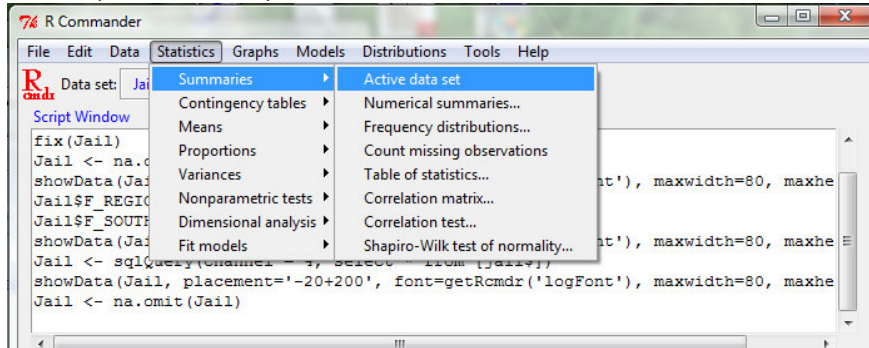
To delete these extraneous observations:

Data-Active dataset..-Remove cases with missing data.

Here, I select the variable STATE, because with these data, any observation missing a value on STATE is an error because all other variables are also missing when STATE is missing. We could legitimately be missing a few values on other measures, but we would not want to delete those entire observations. R can handle these cases.

NUMERICAL SUMMARIES:

Let's do a quick run of summary statistics for this dataset. Select Statistics-Summaries-Active data set



We get:

```
> summary(Jail)
      INC      AFDC      PROP      UNEMP
Min.   : 55.0   Min.   : 99.3   Min.   : 2.087   Min.   : 3.900
1st Qu.:114.5   1st Qu.:210.3   1st Qu.: 3.678   1st Qu.: 5.600
Median :158.5   Median :277.9   Median : 4.322   Median : 6.900
Mean   :174.1   Mean   :285.8   Mean   : 4.376   Mean   : 7.084
3rd Qu.:234.2   3rd Qu.:382.2   3rd Qu.: 4.997   3rd Qu.: 8.075
Max.   :397.0   Max.   :485.8   Max.   : 6.633   Max.   :13.000

      VIOL      HOMI      REPUB      BLACK
Min.   : 0.0470   Min.   : 1.000   Min.   : 2.85    Min.   : 0.2521
1st Qu.: 0.2605   1st Qu.: 3.650   1st Qu.:23.38   1st Qu.: 1.9585
Median : 0.4215   Median : 5.800   Median :40.69   Median : 6.7874
Mean   : 0.4232   Mean   : 6.344   Mean   :39.83   Mean   : 9.3367
3rd Qu.: 0.5487   3rd Qu.: 9.075   3rd Qu.:56.27   3rd Qu.:14.5112
Max.   : 0.9410   Max.   :13.000   Max.   :76.92   Max.   :35.3722

      POVERTY      SOUTH      REGION      STATE
Min.   : 6.65    Min.   : 0.00    Min.   : 1.00    AK      : 1
1st Qu.:10.35   1st Qu.: 0.00   1st Qu.: 2.00   AL      : 1
Median :12.43   Median : 0.00   Median : 2.50   AR      : 1
Mean   :13.14   Mean   : 0.26   Mean   : 2.58   AZ      : 1
3rd Qu.:14.84   3rd Qu.: 0.75   3rd Qu.: 3.75   CA      : 1
Max.   :25.00   Max.   : 1.00   Max.   : 4.00   CO      : 1
```

What's weird about the results for SOUTH and REGION?

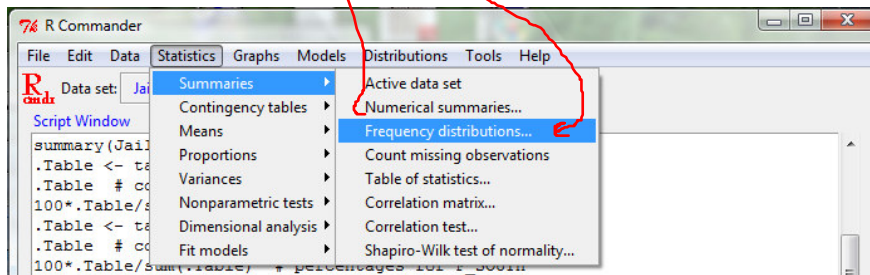
We need R to understand that the values for SOUTH and REGION represent categories, not real numerical quantities. To do this we must convert numeric variables to factors (R calls categorical variables factors) (see the other sheet for details on this procedure). Go to Data-Manage variables...- Convert numeric variables to factors...

Now, re-run the summary statistics as above. What do you get? The below is added; it reflects a frequency distribution of the two variables, now renamed with an F_ prefix (to reflect a "factorized" version of the variable).

```
F_REGION F_SOUTH
1: 9      0:37
2:16     1:13
3:12
4:13
```

How else can we generate numerical summaries?

- Statistics--Summaries
- Numerical summaries
- Frequency distributions



Play around here and see what you can do:

Numerical summaries: Select one or more quantitative variables, and select the quantities of interest. Note, you can select to summarize BY GROUP. This means you can generate separate summary measures by category for any single categorical variable in your dataset.

I can do this by "SOUTH" to see the difference in mean % black in Southern and Non-Southern states:

```
> numSummary(Jail[, "BLACK"], groups=Jail$F_SOUTH, statistics=c("mean", "sd",
"quantiles"), quantiles=c( 0, .25, .5, .75, 1 ))
      mean      sd      0%      25%      50%      75%      100%  n
0  5.117457  4.779509  0.2521  1.5126  3.4184  7.6560  16.4858  37
1  21.345300  8.253334  7.1023  15.8818  22.1922  26.8934  35.3722  13
```

Frequency distribution:

This only applies to categorical variables, so we can get the freq. dist and relative freq. dist for REGION

```
> .Table <- table(Jail$F_REGION)
> .Table # counts for F_REGION
 1  2  3  4
 9 16 12 13
> 100*.Table/sum(.Table) # percentages for F_REGION
 1  2  3  4
18 32 24 26
> remove(.Table)
```

Note that we get the frequency table as in earlier summaries, and now also the % breakout, or relative frequency distribution.

For now, this is about the end of the summary measures we can calculate, but note there are many more menu options for calculating statistics of interest under the Statistics menu.

GRAPHICS:

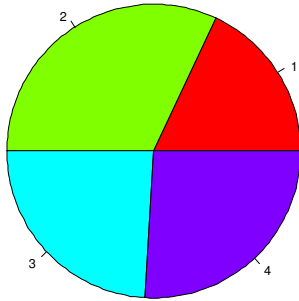
We now turn to the Graphs menu. We know about histograms, stem and leaf displays, boxplots, bar charts and pie charts so far...

If you select the Graphs menu, you'll see many familiar graph types. A hint here: The graphics will pop up in a separate window in the R console (the original program's window). You may need to maximize that window to see your graph pop up. These can easily be saved and inserted into a MS Word document (or

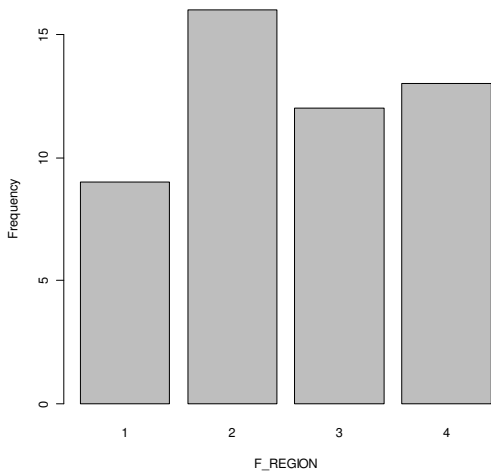
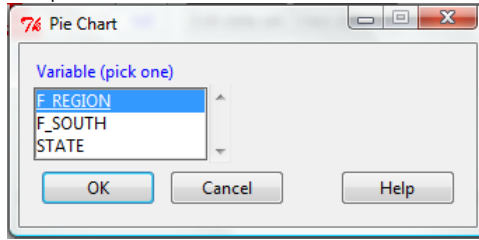
other documents....)

Okay-some quick examples:

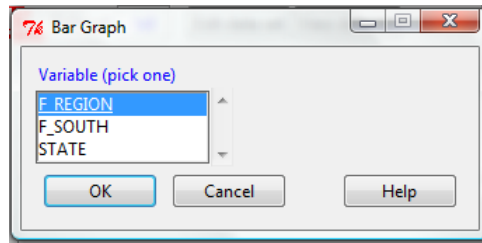
F_REGION



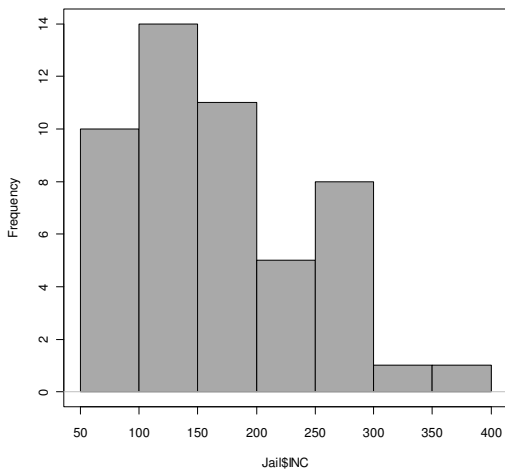
A Pie chart only works with categorical variables: Here, I use REGION as the example variable I can right click on the graphics window and select "Copy as metafile" to copy to the clipboard, then past into a Word document.



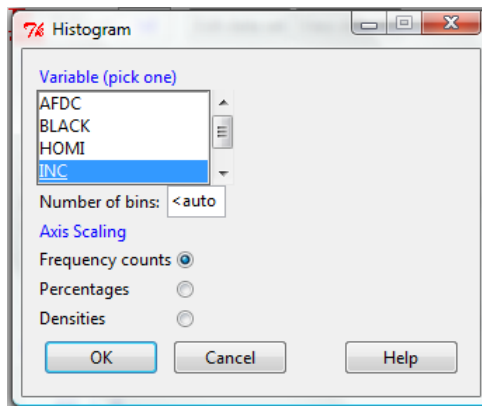
A bar chart--note the "gaps" between the bars indicating it's NOT a histogram.



Now, onto graphs for quantitative variables:



The fully automatic histogram of INC (incarceration rate) with defaults chosen. As shown below:



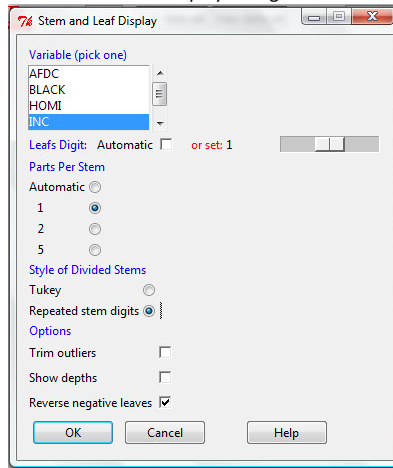
```
> stem.leaf(Jai$INC, style="bare", unit=1, trim.outliers=FALSE, depths=FALSE,
na.rm=TRUE)
1 | 2: represents 12
leaf unit: 1
n: 50
```

```

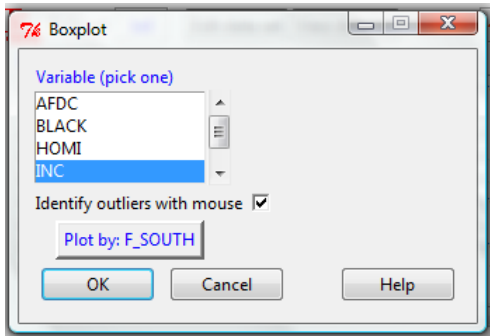
5 | 56
6 | 8
7 |
8 | 2389
9 | 889
10 | 38
11 | 39
12 | 7
13 | 3466
14 | 46899
15 | 6
16 | 15
17 | 5
18 | 1
19 | 244556
20 | 4
21 |
22 | 6
23 | 7
24 | 7
25 | 0146
26 | 7
27 | 9
28 | 18
29 | 4
30 | 8
31 |
32 |
33 |
34 |
35 |
36 |
37 |
38 |
39 | 7

```

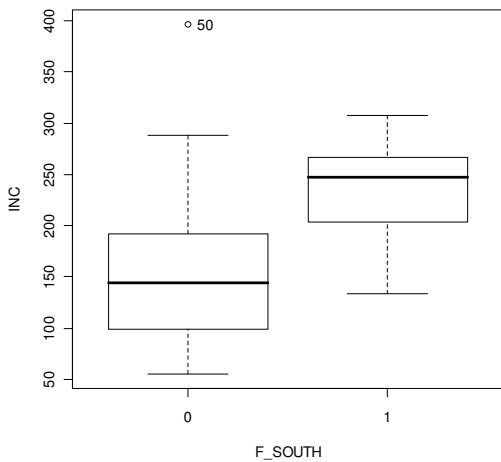
To get stem-and-leaf displays that look like those in Moore, press the Repeated stem digits radio button (under Style of Divided Stems) and de-select the Trim outliers and Show depths check boxes (under Options) in the Stem and Leaf Display dialog box.



Making boxplots: You can make single boxplots, or cooler still, make a side-by-side boxplot to compare across levels of a categorical variable.



The result:



Here, side-by-side boxplots show a high outlier, obs. 50 (North Carolina)