

Scatterplots, Box Plots, Bar Plots, and Cross Tabulation Tables

Paul E. Johnson¹ ²

¹Department of Political Science

²Center for Research Methods and Data Analysis, University of Kansas

Outline

- 1 Terminology
- 2 Scatterplot
- 3 Boxplot
- 4 Cross Tabulation
- 5 Barplot
- 6 Practice Problems

Outline

- 1 Terminology
- 2 Scatterplot
- 3 Boxplot
- 4 Cross Tabulation
- 5 Barplot
- 6 Practice Problems

Levels of Measurement

- Measurement Defined: Observations are associated with “letters” or “numbers” with which we remember them.
- Numeric variables
 - real numbers: continuous, (respondent’s weight, in kilograms)
 - discrete valued: 0,1,2,3 (how many arms does respondent have?)
- Hopefully, proportional re-scaling ($k_1 + k_2 \times X_i$) does not alter the meaning of measurements.
- Natural science abounds with variables that are truly numeric, such as velocity, mass, volume, thickness, etc.
- Social science has fewer “real valued” numeric variables, but we often pretend that discrete variables or ordinal scales are numeric (feeling thermometer variables, left/right ideological placements, etc).

Categorical Measurements

- Categorical variables: Observations are drawn from a discrete list of possible observations
 - Nominal. Ordering of levels is completely unimportant (e.g., candidate names)
 - Ordinal. There is substantive significance to the ordering of the levels (conservative, moderate, liberal)

- The measurement process will often keep a categorical variable in 2 formats

religion	label	numeric score
	Catholic	1
	Protestant	2
	Jewish	3

- In some software (SPSS), the numeric score is the primary referent, while the label is incorporated in presentations.
- In R, the label is the primary referent, the user is discouraged from acting as though the numeric score is meaningful (except for differentiating observations).

Unordered: Category Order does not matter

- Many equivalent numeric scores can be used

religion	label	numeric score	numeric score	numeric score
	Catholic	1	2	-1
	Protestant	2	1	0
	Jewish	3	3	1

- All of the above have the same information
- Any analysis which purports to gain “power” or “statistical significance” by choosing one over the other is wrong.

Ordinal

- A truly ordinal variable has the same meaning as long as the numbers representing it retain the order.

Party Ideology	X	Y	Z
Democrat	1	-10,000	399
Republican	2	0	653
Libertarian	3	1	1,000,000

- X, Y, Z are equivalent numeric scores.
- Reminder: if conclusions differ from analysis of X, Y, or Z, the procedure is probably wrong.

Factor = R Term For Categorical Variable

- In R, categorical variables are called factors (see functions `factor()`, `ordered()`, `levels()`)
- Data values as levels, terms like “male”, “female”
- Most R statistical procedures try to automatically handle the “behind the scenes” conversion into numeric variables.
- Examples, $sex_i \in \{Male, Female\}$. When R analyzes that variable, many procedures will report a result for a numeric variable that R constructs automatically, $sexFemale$. (see next slide)

How to make a Categorical variable into Numeric Variables

- A “dummy variable” is usually coded 0 or 1, to mean that a quality is present (or not).
- A variable “sex” may be *Male* or *Female*, but we often focus on a 0, 1 numeric representation, $sexFemale_i \in \{0, 1\}$ (remembering 0 is for *Male* and 1 is for *Female*).
- $sexFemale_i$ often called an “indicator variable,” or “binary variable”, or “dichotomous variable”, or “dummy variable”.
- Note the variable sex_i can beget 2 indicators, $sexMale_i = \{0, 1\}$ or $sexFemale_i \in \{0, 1\}$.

Foreshadow multi-Category treatment

- A multi-category variable like religion may be used to create several separate indicators

religion	numeric score	Cath	Prot	Jewish
Catholic	1	1	0	0
Protestant	2	0	1	0
Jewish	3	0	0	1

- Expect the “too many dummy variables” problem in regression.

Summarizing “categorical” variables (“factors”) is an art form.

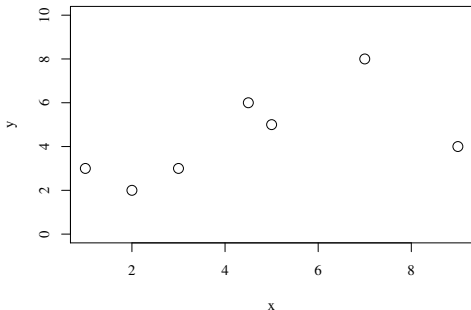
- Only the “mode” appears to be universally accepted as one indicator of central tendency.
- No universally accepted index of “diversity”
- A table of frequencies (either raw counts or proportions) may be best.
- `rockchalk::summarize()` tries to handle that.

Outline

- 1 Terminology
- 2 Scatterplot**
- 3 Boxplot
- 4 Cross Tabulation
- 5 Barplot
- 6 Practice Problems

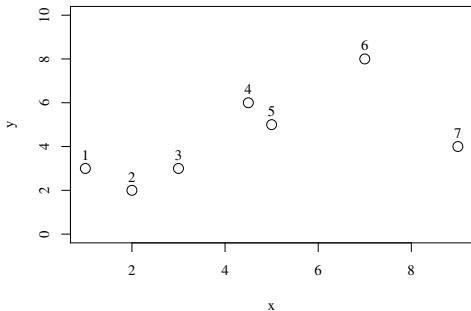
Scatterplot 2 Numeric Variables

- Each observation is one “point”
- x and y appear to positively related
 - they “go together”, but not perfectly

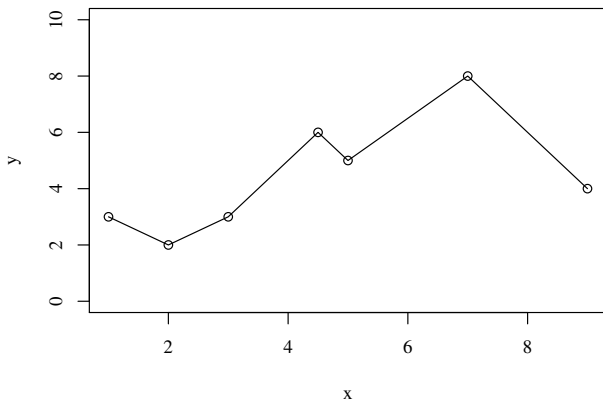


Perhaps you want to number the points

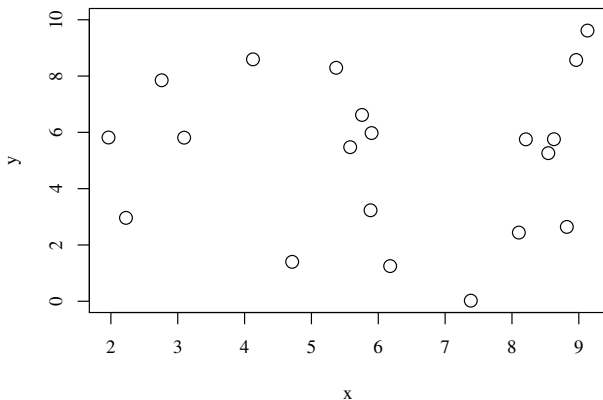
- I used row numbers for the points



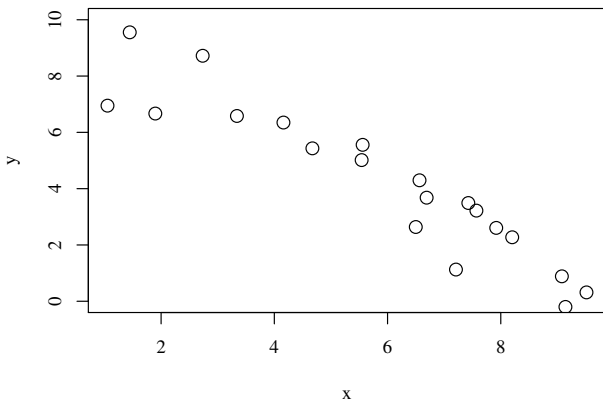
Line graph: Connect the dots



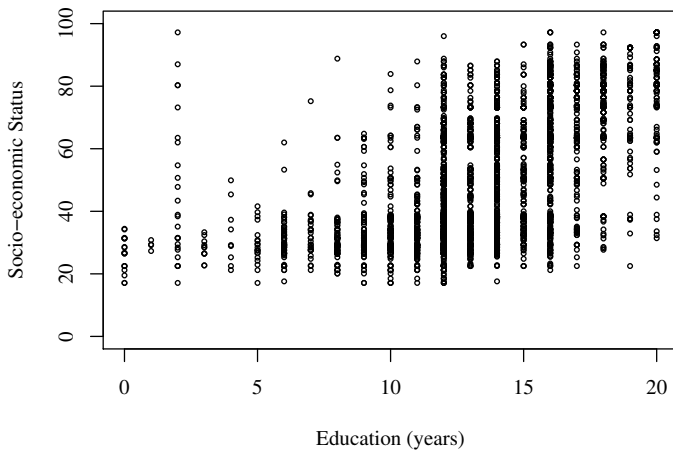
Another Scatterplot: No relationship



Another Scatterplot: Negative relationship



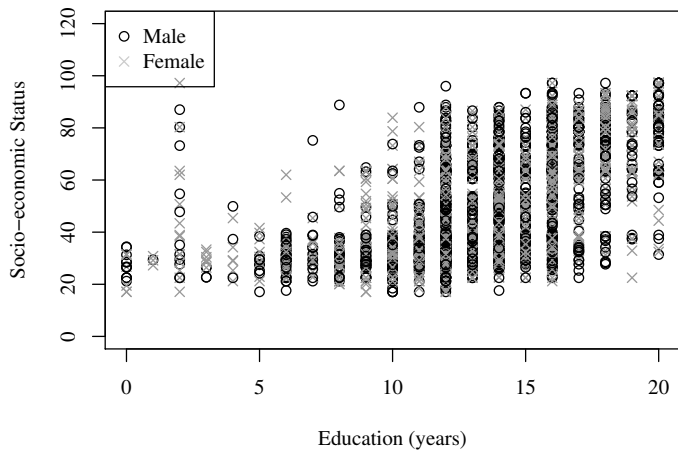
2 Variables from GSS



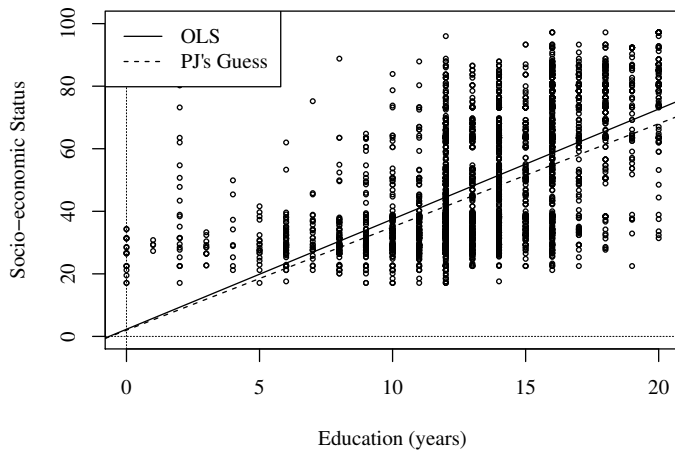
R plot tools

- R's `plot()` function is a rather interesting case: it doesn't actually draw plots, it re-organizes the input and sends it to other plot functions (for scatterplots, barplots, and so forth)
- Numeric variable plots are created by the function `plot.default()`, which we could use directly instead if we wanted to.
- plot functions share a common syntax, we specify (optionally) `xlim`, `ylim`, `col`, `type`, and so forth.
- Once plot is created, add details with "points", "text", "lines", "polygon", "legend" and so forth.

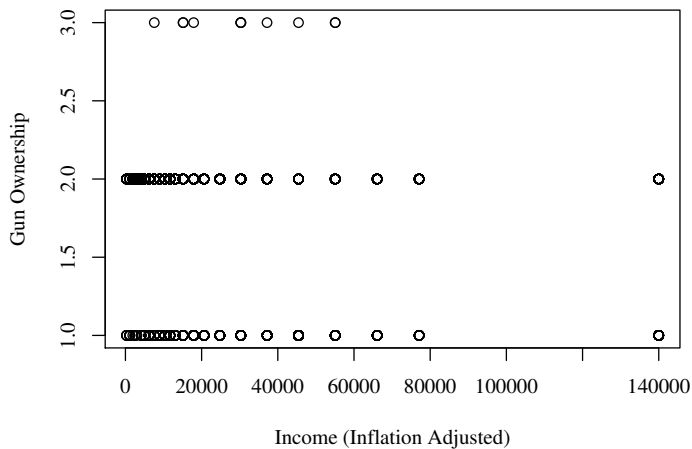
Color Coded Men and Women



Can Add Predictive Lines

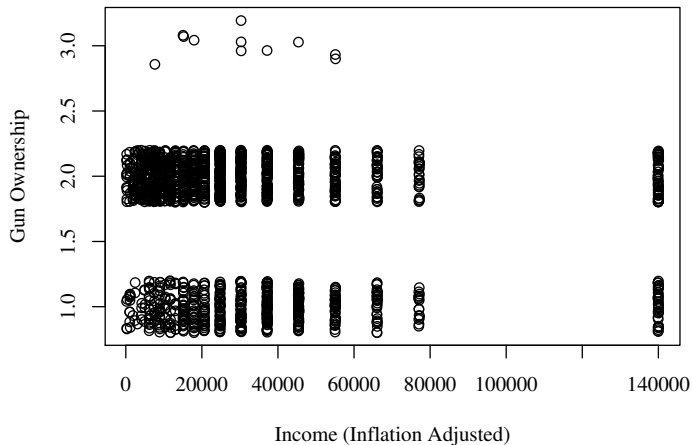


Discrete Variable Problem: Piled Up Observations



Jitter Observations

The jitter function adds random noise to scores, so they don't overlap anymore



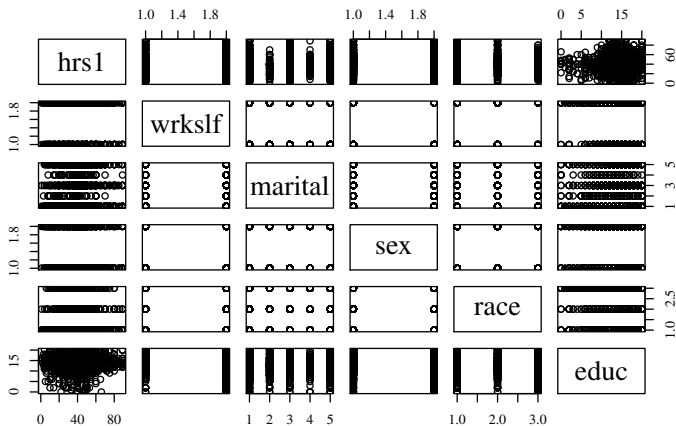
Is Jittering Good? Or Dubious!

- If variables are truly numeric (discrete), jittering may help.
- But if variables are categorical, it may be deceptive.
- Giving numbers 1, 2, 3 to gun ownership does not mean the numbers are meaningful!
 - 1 : Yes
 - 2 : No
 - 3 : Refused to Answer
- Better to use methods intended for truly categorical outcomes
- Nevertheless, common “solution” is to add numerical random noise to 0, 1 in order to make a better looking scatterplot

Scatterplot Matrix

- Some programs offer a quick way to see a lot of scatterplots in a single picture.
- Usually doesn't help me too much.

S.M. for 6 variables

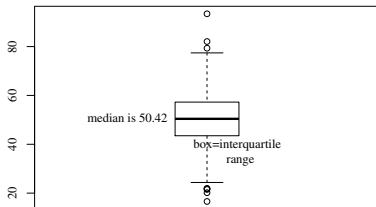


Outline

- 1 Terminology
- 2 Scatterplot
- 3 Boxplot**
- 4 Cross Tabulation
- 5 Barplot
- 6 Practice Problems

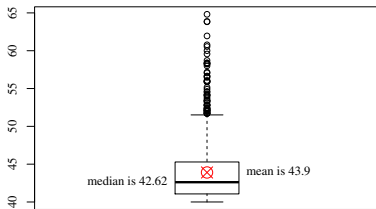
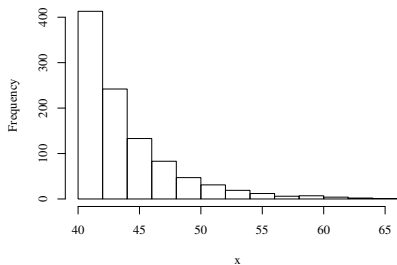
Boxplot: Like a Histogram Turned on its Side

- A boxplot is
 - with weird shaped bars and funny markings.
- Dark line at Median
- Box has 25% of cases above and below
- “Whiskers” default to reach out 1.5*interquartile range
- Dots represent extreme cases.



This variable is symmetric, with mean near median of 50.

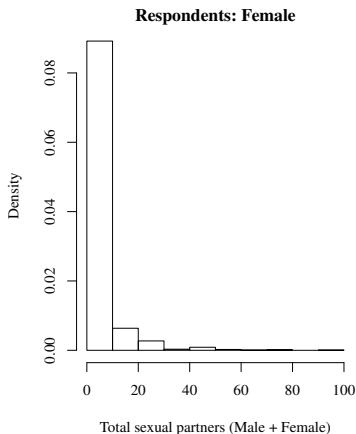
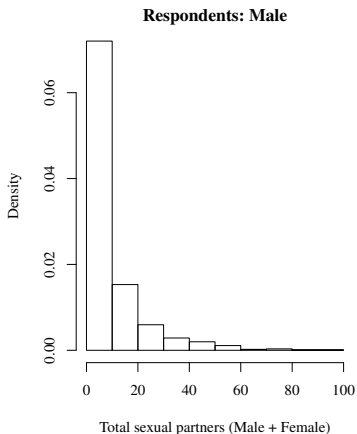
Boxplot: For a Nonsymmetric Variable



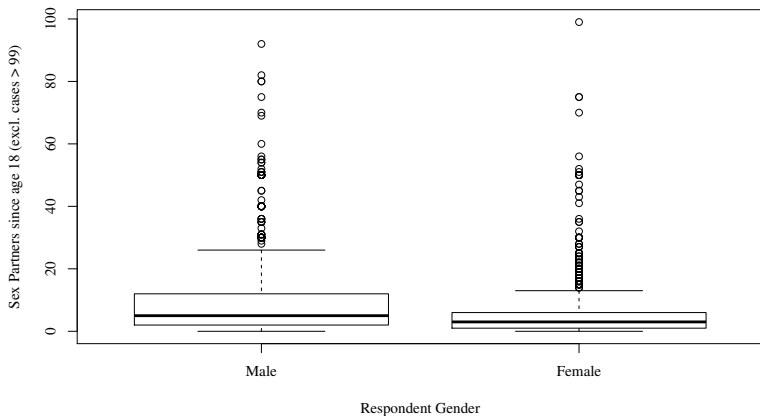
Boxplot: View Several Subsets

- A histogram can display only one group of respondents
- If you get used to boxplots, you gain the benefit that more groups can be fit into a single display.

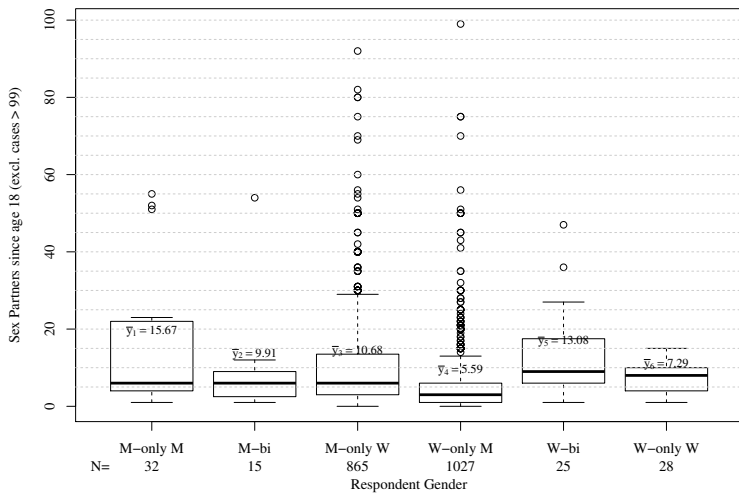
Histograms for Number of Sexual Partners(GSS 2006)



Use a Box Plot Instead



My Most Informative Box Plot



Outline

- 1 Terminology
- 2 Scatterplot
- 3 Boxplot
- 4 Cross Tabulation**
- 5 Barplot
- 6 Practice Problems

Frequencies

- A Frequency Count is a simple table that lists the number of observations within each category

Hair Color Among POLS110 Students				
Brown	Black	Blond	Red	Unknown
155	67	48	10	63

- The MODE is the most frequently occurring value. (Here, Brown)

Cross-Tabulation Table

- A Cross-Tabulation table counts the number of observations in cross-classified variables.

Hair Color Among POLS110 Students					
	Brown	Black	Blond	Red	Unknown
From Kansas	98	40	18	5	10
Not	57	27	30	5	53
	155	67	48	10	63

- This simple descriptive table does not necessarily have a dependent or independent variable.
- There have been many efforts to summarize the observed relationship numerically (Google for correlation coefficients like phi, Gamma or Kendall's tau)

The Iron Laws of Crosstabs

For analytical purposes, Gerry Lowenberg taught me to always follow 3 rules. You should too.

- ① IV on top, DV on left
- ② Convert to percentages (or proportions) on the columns
- ③ Compare the across, find if columns are distributed differently

The FX Network is	Column	Percentages
	Respondent	Sex
	male	female
really horrible	25%	80%
OK	50%	18%
really great	25%	2%
N	343	288

Example: 1980 US Pres. Election

The 1980 American presidential election (ANES):

	Party Identification		
	Republican	Independent	Democrat
Percentages voting for			
Reagan	86%	55	26
Carter	9	30	67
Anderson	4	12	6
N	544	355	721

Example: Patriotism and Voting

This was compiled from the American National Election Study of 1988.
How does seeing the American Flag make you feel? *

	Extremely good	Very good	Somewhat good	Not very good
Percentages voting for				
Bush	60.7%	50%	30%	8%
Dukakis	39.3%	50%	70%	92%
N	403	666	299	28

*Excludes voters who did not select either Bush or Dukakis

There Are Standards for Tables

- There is a literature on the correct format for tables in publications
- See Simon Fear, “Publication quality tables in \LaTeX ” (in the \LaTeX booktabs package)
- Advice: Avoid vertical lines

Example: GSS 2006 Gun Data

Table : Opinions on Gun Registration Laws

Stance on Gun Registration	Does Respondent Own a Gun?		
	Yes	No	Refused To Say
Favor	71%	85	57
Oppose	29	15	43
Number of Cases	600	1128	23

- Reminder how I did that. Using gss data “dat” (from above)

```
library(memisc)
guntab <- with(dat, genTable(percent(gunlaw) ~ owngun))
toLatex(guntab)
```

Making tables in R. PITA

- `table(rowvar, colvar)` gives the raw counts, completely unbeautified

```
load("/home/pauljohn/ps/SVN-guides/stat/DataSets/GSS/  
gss-subset2.Rda")  
with(dat, table(gunlaw, owngun) )
```

gunlaw	owngun		
	YES	NO	REFUSED
FAVOR	464	1085	17
OPPOSE	192	193	10

Get column percentages with prop.table()

- `prop.table(table(rowvar, colvar), margin=2)` converts the table into column percentages

```
t1 <- with(dat, table(gunlaw, owngun, exclude=NULL) )
t1.prop <- prop.table(t1, margin=2)
t1.prop
```

gunlaw	owngun			
	YES	NO	REFUSED	<NA>
FAVOR	0.69879518	0.83014537	0.56666667	0.00079713
OPPOSE	0.28915663	0.14766641	0.33333333	0.00000000
<NA>	0.01204819	0.02218822	0.10000000	0.99920287

Ach. No column totals

Add Column totals

```
t1.marg <- margin.table(t1, margin=2)
t1.result <- rbind(t1.prop, t1.marg)
t1.result
```

	YES	NO	REFUSED	<NA>
FAVOR	0.698795	0.830145	0.56667	0.00079713
OPPOSE	0.289157	0.147666	0.33333	0.00000000
<NA>	0.012048	0.022188	0.10000	0.99920287
t1.marg	664.000000	1307.000000	30.00000	2509.00000000

Use supplementary packages

- gmodels: the original CrossTable function
- memisc: My favorite for making tables (can make \LaTeX)
- “vcd” & “vcdExtra” (VCD=“visualize categorical data”)
- “descr”

```
library(gmodels); with(dat, CrossTable(gunlaw, owngun))
```

Cell Contents				
				N
Chi-square contribution				
	N / Row Total			
	N / Col Total			
	N / Table Total			
Total Observations in Table: 1961				
gunlaw	owngun			Row Total
	YES	NO	REFUSED	
FAVOR	464	1085	17	1566
	6.841	4.067	0.965	
	0.296	0.693	0.011	0.799
	0.707	0.849	0.630	
	0.237	0.553	0.009	
OPPOSE	192	193	10	395
	27.121	16.123	3.826	
	0.486	0.489	0.025	0.201
	0.293	0.151	0.370	
	0.098	0.098	0.005	
Column Total	656	1278	27	1961
	0.335	0.652	0.014	

descr CrossTable similar

```
library(descr);
descrCT <- with(dat , descr::CrossTable(gunlaw , owngun))
descrCT
```

Cell Contents				
				N
				Chi-square contribution
				N / Row Total
				N / Col Total
				N / Table Total
gunlaw	owngun			
	YES	NO	REFUSED	Total
FAVOR	464	1085	17	1566
	6.841	4.067	0.965	
	0.296	0.693	0.011	0.799
	0.707	0.849	0.630	
	0.237	0.553	0.009	
OPPOSE	192	193	10	395
	27.121	16.123	3.826	
	0.486	0.489	0.025	0.201
	0.293	0.151	0.370	
	0.098	0.098	0.005	
Total	656	1278	27	1961
	0.335	0.652	0.014	

Outline

- 1 Terminology
- 2 Scatterplot
- 3 Boxplot
- 4 Cross Tabulation
- 5 Barplot**
- 6 Practice Problems

Barplot

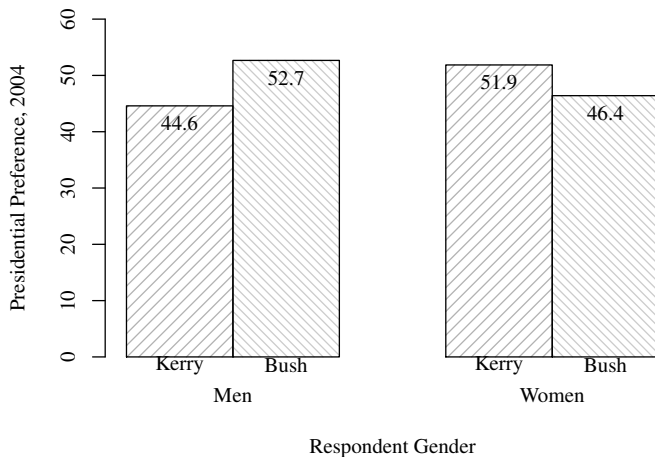
- Barplot: Make a graphic presentation of a cross tabulation table
- Use
 - Any discrete variable that can classify respondents
 - Any summary score (mean, proportion, count) calculated from the subgroups of respondents
- The width of the bar has no “substantive” meaning
- Unlike a histogram, where the width \times height represents the area

Table Demonstrating the Gender Gap in 2004

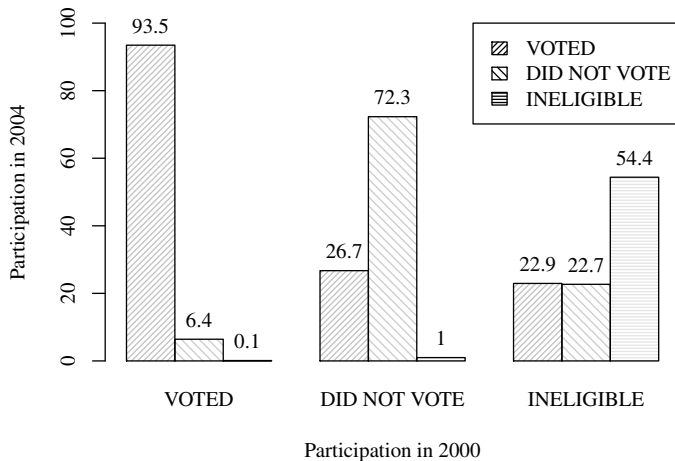
Presidential Choice	Respondent Gender	
	Male	Female
Kerry	45%	52
Bush	53	47
Nader	2	1
Didn't Vote*	1	1
Number of Cases	1137	1487

* Respondent voted, but did not cast vote in Presidential contest

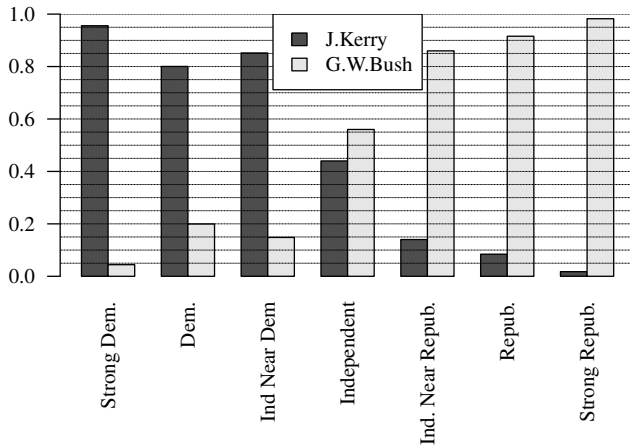
Barplot Representing Gender Gap in 2004



Barplot: Voter Participation Dynamics



Barplot: Partisanship in 2004



To Find Out More

- Check Rcourse for lectures plot-1, plot-2, and plot-3
- In <http://pj.freefaculty.org/R/WorkingExamples>, see plot-barplot*.R examples. They have very detailed step-by-step instructions.

NB: Many Other Types of Plots

- “spinogram” is a barplot that scales the widths of the bars according to the numbers of observations
- dot plot replaces the “big boxy bars” with smaller dots to mark the tops of the bars.
- pie charts are awful, every reasonable person would agree they should never be used for anything. (my definition of reasonable is based on your answer: “do you hate pie charts?”).

Outline

- 1 Terminology
- 2 Scatterplot
- 3 Boxplot
- 4 Cross Tabulation
- 5 Barplot
- 6 Practice Problems**

Problems

- 1 Better learn to make some scatterplots. I used to have a bunch of scatterplot exercises. Where did they go? Oh, well. Lets try this. First, make some data that is easy to understand. Create 5 observations for x and y

```
x <- c( 3, 5, 7, 9, 11)
y <- c( 1, 8, 3, 5, 7)
z <- c( "M", "F", "F", "M", "F")
z <- factor(z, levels=c("M", "F"), labels=c("
  Male", "Female"))
```

Then put that into a data frame called dat and remove the stray variables

```
dat <- data.frame(myx=x, myy=y, myz=z)
rm(x, y, z)
```

Note that you get the same plot result, whether you use the x, y format for the plot command. This uses the raw format, and if you want documentation on it, you should run `?plot.default` because that's where plot sends the data to get the work done.

Problems ...

```
plot(x=dat$myx, y=dat$myy)
## which you could abbreviate to
plot(dat$myx, dat$myy)
## because R does "positional matching"
    automatically
## Or try this more elegant way
with(dat, plot(myx, myy))
## which saves you the trouble of repeatedly
    typing dat$
```

Or use the “formula” format, which I prefer:

```
plot(myy ~ myx, data=dat)
```

When you use the formula interface, the plot command sends the work to the function `plot.formula`, for which there is a help page.

Problems ...

- ② Better make some box plots. I better make some exercises on that. But I don't have much in mind right now.

Oh, here. In R, try this. This creates a numeric variable `xn`, and a categorical (or "factor") variable that represents the same thing.

```
xf <- gl(4,250, labels=c("A","B","C","D"))
xn <- as.numeric(xf)
y <- rnorm(1000, m = xn, s=2)
par(mfcol=c(1,2))
plot(xf, y, main="x is a factor variable")
plot(xn, y, main="x is a categorical variable"
      )
par(mfcol=c(1,1))
```

Explain why the 2 plots differ!

Problems ...

- 3 Make some cross tabulation tables. For some reason, I have never really understood how to use the `xtabs` function in R, but I get by with `table` from base R and the `CrossTable` function from the `gmodels` package. Oh, also vital are some functions from the `memisc` package. The `memisc` package can be used to create the table in the ordinary R format, and then it can produce the publication quality latex table as well. That's written up in *Stuff Worth Knowing*, there's a chapter on `crosstabs`.

```
x <- c("M", "M", "M", "F", "M", "F", "F")
xf <- factor(x)
y <- c("Y", "N", "Y", "Y", "Y", "Y", "Y")
yf <- factor(y)
dat <- data.frame(x, xf, y, yf)
rm(x, y, xf, yf)
t1 <- with(dat, table(xf, yf))
t1
t2 <- with(dat, prop.table(t1, 2))
t2
library(gmodels)
```

Problems ...

```
with(dat, CrossTable(xf, yf))
library(memisc)
mytable <- genTable(percent(yf) ~ xf, data=dat
)
mytable
toLatex(mytable)
```

- 4 Try to make a barplot. I suppose it would be good to try to make a barplot of t2 in the previous exercise. Practice up by doing these exercises. Remember that barplot does not “digest” the data for you. It is necessary to run a table-making command first, and then the barplot function wants that. For example

```
x <- rnorm(100)
xc1 <- cut(x, c(-1000, -1.5, 0, .6, 2,100),
  labels=c("none", "some", "occasional", "often",
  "abundant"))
xt1 <- table(xc1)
barplot(xt1)
```

Problems ...

cut is a handy way to “create” a categorical variable by dividing a numeric variable into ranges.

After that, try this additional code (we still need x and xc1 from previous in the workspace)

```
y <- 0.5*x + rnorm(100)
yc1 <- cut(y, c(-1000, -1, 1.1, 1.7, 2.3, 100)
          , labels=c("frozen", "cold", "chilly", "warm",
                    "boiling"))
xyt1 <- prop.table(table(yc1, xc1), 2)
barplot(xyt1, ylab="Proportions Of Respondents
")
```

Who likes stacked barplots? Not me, usually. So try:

```
barplot(xyt1, beside=T, ylab="Proportions")
```

Does that plot make you feel as though your head is chopped off?

```
barplot(xyt1, beside=T, ylim=c(0,0.85))
```

Problems ...

Need a legend. We did not specify colors with our barplot, so that means the barplot colors are coming from the default color assignment function, which I believe is "gray.colors". Lets see if this works:

```
barplot(xyt1, beside=T, ylim=c(0,.85))  
legend("topright", legend=levels(yc1), fill=  
      gray.colors(5))
```

We could avoid the guessing game of "what were those colors" if we assigned colors explicitly in our code. That's not difficult. Try, for example,

```
mycolors <- rainbow(5)  
barplot(xyt1, beside=T, ylim=c(0,.85), col=  
      mycolors)  
legend("topright", legend=levels(yc1), fill=  
      mycolors)
```

Problems ...

I would never use those rainbow colors, but it is good to know we could make a barplot for the school children if we needed to.

There are many options that can be brought to bear to customize the appearance of a barplot. I urge you to try my R WorkingExamples called `barplot.R` and `barbplot-2.R`.