Statistical Graphics

Distribution of 1 Variable:

```
histogram( ~ wage, data=CPS85)
densityplot( ~ wage, data=CPS85)
freqpolygon( ~ wage, data=CPS85)
```

Scatter plot:

```
xyplot(wage~educ, data=CPS85)
```

Compare distribution by group:

```
bwplot(wage ~ sex, data=CPS85)
```

Can use `groups=sex` as an argument to `xyplot()`, `densityplot()`, or `freqpolygon()`

RMarkdown documents

```
---
title: "Homework #3"
author: "Abby Seedlef"
date: "January 7, 2015"
output: pdf_document
---

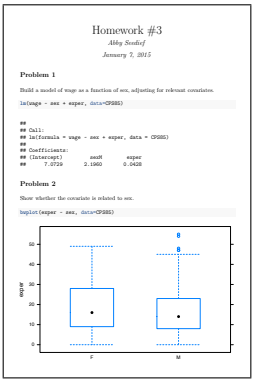
```{r include=FALSE}
require(mosaic)
require(mosaicData)
```

Problem 1

Build a model of wage as a function of sex,
adjusting for relevant covariates.
```{r}
lm(wage ~ sex + exper, data=CPS85)
```

Problem 2

Show whether the covariate is related to sex.
```{r}
bwplot(exper ~ sex, data=CPS85)
```
```


Homework #3

Abby Seedlef

January 7, 2015

Problem 1

Build a model of wage as a function of sex, adjusting for relevant covariates.

```
lm(wage ~ sex + exper, data=CPS85)

##
## Call:
## lm(formula = wage ~ sex + exper, data = CPS85)
##
## Coefficients:
## (Intercept)         sexM        exper
##      7.0729       2.1960       0.0428
```

Problem 2

Show whether the covariate is related to sex.

```
bwplot(exper ~ sex, data=CPS85)
```


Compile to any of HTML, PDF, or Word.

See `mosaic plain` template through RStudio menu:
File/New File/RMarkdown/From Template

Basic Statistical Tests

Difference between two means

```
result<-t.test(wage ~ sex, data=CPS85, mu=1.50)
```

Difference between two proportions

```
result<-prop.test(sex~homeless, data=HELPrct)
```

For either, a verbose report is printed. Use `pval(result)` or `confint(result)`

Randomization and Iteration

Resample/Bootstrap:

```
do(100)*mean(wage ~ sex, data=resample(CPS85))
```

Random Permutations:

```
do(100)*mean(wage ~ shuffle(sex), data=CPS85)
```

1000 trials of flipping 6 coins, count heads

```
flips <- do(1000) * rflip(6)
tally( ~ heads, data=flips)
```

10000 trials of adding three dice

```
scores <- do(10000)*sum(resample(1:6,size=3))
freqpolygon(~ result, data=scores)
```