Documentation on using ioslides is available here: http://rmarkdown.rstudio.com/ioslides_presentation_format.html Some slides are adopted (or copied) from OpenIntro: https://www.openintro.org/

Meetup Presentations

- Hector Santana (7.23,7.39) http://rpubs.com/HSantana/presentationdata606
- Joshua Bentley (7.25) http://rpubs.com/ajbentley/cuny_msds_606_725answer
- Alexander Niculescu (7.29) http://rpubs.com/aniculescu/DATA606_PRESENTATION_CHAPTER_7_PROBLEM_29
- Eunice Ok (7.31) http://rpubs.com/eko226/443201

Weight of Books

```
data(allbacks, package='DAAG')
head(allbacks)
```

```
##   volume area weight cover
## 1    885  382    800    hb
## 2   1016  468    950    hb
## 3   1125  387   1050    hb
## 4    239  371    350    hb
## 5    701  371    750    hb
## 6    641  367    600    hb
```

From: Maindonald, J.H. & Braun, W.J. (2007).

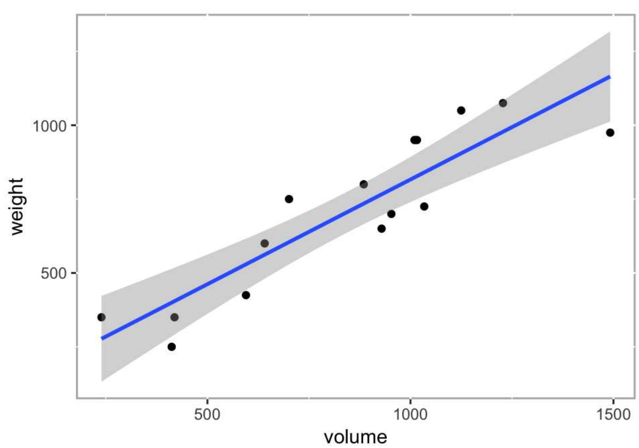
3/28

Weights of Books (cont)

```
lm.out <- lm(weight ~ volume, data=allbacks)
```

$$\hat{weight} = 108 + 0.71 volume$$

$$R^2 = 80\%$$



4/28

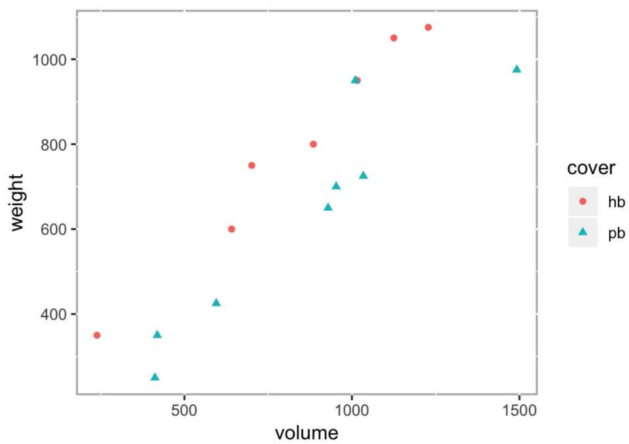
Modeling weights of books using volume

```
summary(lm.out)
```

```
##
## Call:
## lm(formula = weight ~ volume, data = allbacks)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -189.97 -109.86   38.08  109.73  145.57
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) 107.67931   88.37758   1.218    0.245
## volume        0.70864    0.09746   7.271 6.26e-06 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 123.9 on 13 degrees of freedom
## Multiple R-squared:  0.8026, Adjusted R-squared:  0.7875
## F-statistic: 52.87 on 1 and 13 DF,  p-value: 6.262e-06
```

Weights of hardcover and paperback books

- Can you identify a trend in the relationship between volume and weight of hardcover and paperback books?



- Paperbacks generally weigh less than hardcover books after controlling for book's volume.

Modeling weights of books using volume and cover type

```
lm.out2 <- lm(weight ~ volume + cover, data=allbacks)
summary(lm.out2)
```

```
##
## Call:
## lm(formula = weight ~ volume + cover, data = allbacks)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -110.10  -32.32  -16.10   28.93  210.95
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept)  197.96284   59.19274   3.344 0.005841 **
## volume         0.71795    0.06153  11.669  6.6e-08 ***
## coverpb     -184.04727   40.49420  -4.545 0.000672 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 78.2 on 12 degrees of freedom
## Multiple R-squared:  0.9275, Adjusted R-squared:  0.9154
## F-statistic: 76.73 on 2 and 12 DF,  p-value: 1.455e-07
```

Linear Model

$$\hat{weight} = 198 + 0.72 volume - 184 coverpb$$

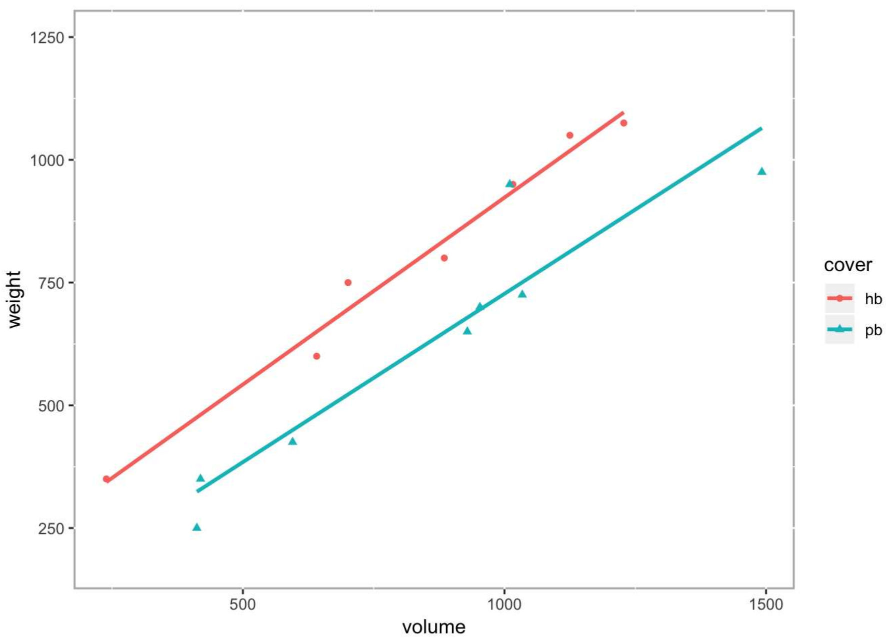
1. For **hardcover** books: plug in for cover.

$$\hat{weight} = 197.96 + 0.72 volume - 184.05 \times 0 = 197.96 + 0.72 volume$$

2. For **paperback** books: plut in 1 for cover.

$$\hat{weight} = 197.96 + 0.72 volume - 184.05 \times 1$$

Visualising the linear model



Interpretation of the regression coefficients

	Estimate	Std. Error	t value	Pr(>|t|)
(Intercept)	197.9628	59.1927	3.34	0.0058
volume	0.7180	0.0615	11.67	0.0000
coverpb	-184.0473	40.4942	-4.55	0.0007

- **Slope of volume**: All else held constant, books that are 1 more cubic centimeter in volume tend to weigh about 0.72 grams more.
- **Slope of cover**: All else held constant, the model predicts that paperback books weigh 184 grams lower than hardcover books.
- **Intercept**: Hardcover books with no volume are expected on average to weigh 198 grams.
  - Obviously, the intercept does not make sense in context. It only serves to adjust the height of the line.

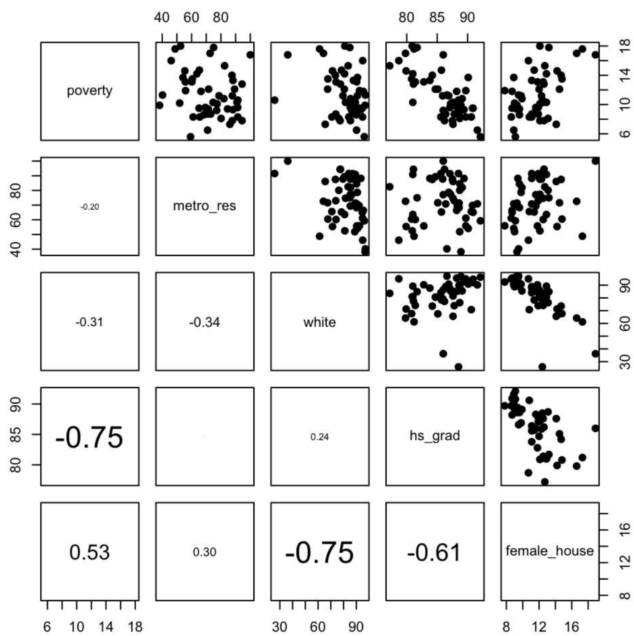
Modeling Poverty

```
poverty <- read.table("../Data/poverty.txt", h = T, sep = "\t")
names(poverty) <- c("state", "metro_res", "white", "hs_grad", "poverty", "female_house")
poverty <- poverty[,c(1,5,2,3,4,6)]
head(poverty)
```

```
##        state poverty metro_res white hs_grad female_house
## 1    Alabama    14.6      55.4  71.3    79.9         14.2
## 2     Alaska     8.3      65.6  70.8    90.6         10.8
## 3    Arizona    13.3      88.2  87.7    83.8         11.1
## 4   Arkansas    18.0      52.5  81.0    80.9         12.1
## 5 California    12.8      94.4  77.5    81.1         12.6
## 6   Colorado     9.4      84.5  90.2    88.7          9.6
```

From: Gelman, H. (2007). Cambridge University Press.

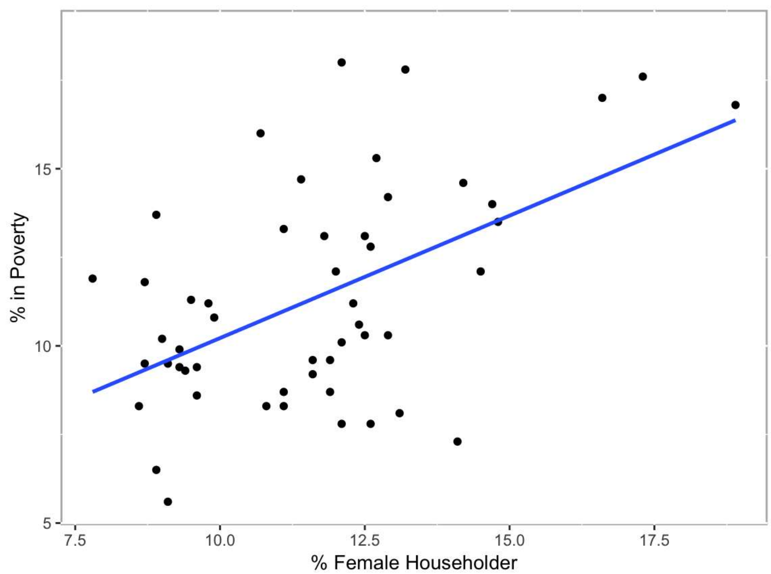
Modeling Poverty



Predicting Poverty using Percent Female Householder

```
lm.poverty <- lm(poverty ~ female_house, data=poverty)
summary(lm.poverty)
```

```
##
## Call:
## lm(formula = poverty ~ female_house, data = poverty)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -5.7537 -1.8252 -0.0375  1.5565  6.3285
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)    3.3094     1.8970   1.745   0.0873 .
## female_house   0.6911     0.1599   4.322 7.53e-05 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 2.664 on 49 degrees of freedom
## Multiple R-squared:  0.276,  Adjusted R-squared:  0.2613
## F-statistic: 18.68 on 1 and 49 DF,  p-value: 7.534e-05
```



% in Poverty
15
10
5
7.5
10.0
12.5
15.0
17.5
% Female Householder

Another look at R^2

R^2 can be calculated in three ways:

1. square the correlation coefficient of x and y (how we have been calculating it)
2. square the correlation coefficient of y and $\hat{y}$
3. based on definition:

$$R^2 = \frac{explained \quad variability \quad in \quad y}{total \quad variability \quad in \quad y}$$

Using ANOVA we can calculate the explained variability and total variability in y.

Sum of Squares

```r
anova.poverty <- anova(lm.poverty)
print(xtable(anova.poverty, digits = 2), type='html')
```

	Df	Sum Sq	Mean Sq	F value	Pr(>F)
female_house	1.00	132.57	132.57	18.68	0.00
Residuals	49.00	347.68	7.10		

Sum of squares of : $SS_{Total} = \sum (y - \bar{y})^2 = 480.25 \rightarrow$ **total variability**

Sum of squares of residuals: $SS_{Error} = \sum e_i^2 = 347.68 \rightarrow$ **unexplained variability**

Sum of squares of : $SS_{Model} = SS_{Total} - SS_{Error} = 132.57 \rightarrow$ **explained variability**

$$R^2 = \frac{explained \quad variability \quad in \quad y}{total \quad variability \quad in \quad y} = \frac{132.57}{480.25} = 0.28$$

16/28

Why bother?

- For single-predictor linear regerssion, having three ways to calculate the same value may seem like overkill.
- However, in multiple linear regression, we can't calculate R^2 as the square of the correlation between and because we have multiple s.
- And next we'll learn another measure of explained variability, R^2, that requires the use of the third approach, ratio of explained and unexplained variability.

Predicting poverty using % female household and % white

```
lm.poverty2 <- lm(poverty ~ female_house + white, data=poverty)
print(xtable(lm.poverty2), type='html')
```

	Estimate	Std. Error	t value	Pr(>|t|)
(Intercept)	-2.5789	5.7849	-0.45	0.6577
female_house	0.8869	0.2419	3.67	0.0006
white	0.0442	0.0410	1.08	0.2868

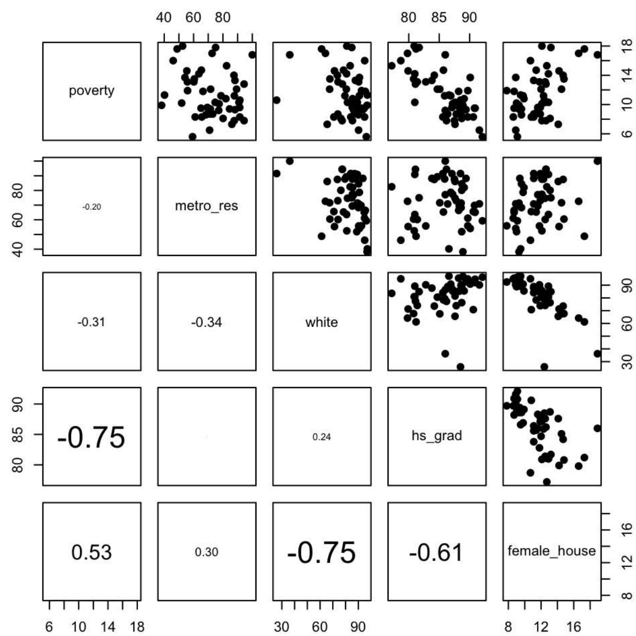
```
anova.poverty2 <- anova(lm.poverty2)
print(xtable(anova.poverty2, digits = 2), type='html')
```

	Df	Sum Sq	Mean Sq	F value	Pr(>F)
female_house	1.00	132.57	132.57	18.74	0.00
white	1.00	8.21	8.21	1.16	0.29
Residuals	48.00	339.47	7.07		

$$R^2 = \frac{explained \quad variability \quad in \quad y}{total \quad variability \quad in \quad y} = \frac{132.57 + 8.21}{480.25} = 0.29$$

18/28

Does adding the variable `white` to the model add valuable information that wasn't provided by `female_house`?



Collinearity between explanatory variables

poverty vs % female head of household

	Estimate	Std. Error	t value	Pr(>|t|)
(Intercept)	3.3094	1.8970	1.74	0.0873
female_house	0.6911	0.1599	4.32	0.0001

poverty vs % female head of household and % female household

	Estimate	Std. Error	t value	Pr(>|t|)
(Intercept)	-2.5789	5.7849	-0.45	0.6577
female_house	0.8869	0.2419	3.67	0.0006
white	0.0442	0.0410	1.08	0.2868

Note the difference in the estimate for `female_house`.

Collinearity between explanatory variables

- Two predictor variables are said to be collinear when they are correlated, and this collinearity complicates model estimation.
  Remember: Predictors are also called explanatory or independent variables. Ideally, they would be independent of each other.
- We don't like adding predictors that are associated with each other to the model, because often times the addition of such variable brings nothing to the table. Instead, we prefer the simplest best model, i.e. model.
- While it's impossible to avoid collinearity from arising in observational data, experiments are usually designed to prevent correlation among predictors

R^2 vs. adjusted R^2

Model	R^2	Adjusted R^2
Model 1 (Single-pridictor)	0.28	0.26
Model 2 (Multiple)	0.29	0.26

- When any variable is added to the model R^2 increases.
- But if the added variable doesn't really provide any new information, or is completely unrelated, adjusted R^2 does not increase.

Adjusted R^2

$$R^2_{adj} = 1 - \left(\frac{SS_{error}}{SS_{total}} \times \frac{n-1}{n-p-1} \right)$$

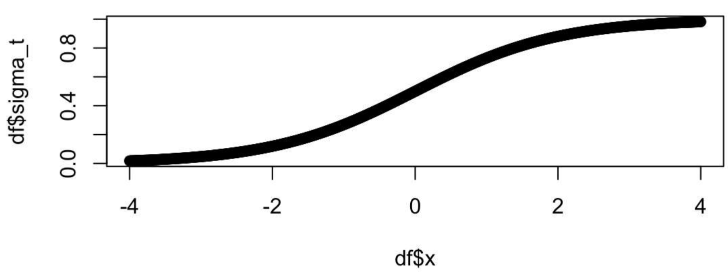
where is the number of cases and is the number of predictors (explanatory variables) in the model.

- Because is never negative, R^2_{adj} will always be smaller than R^2.
- R^2_{adj} applies a penalty for the number of predictors included in the model.
- Therefore, we choose models with higher R^2_{adj} over others.

The Logistic Function

$$\sigma(t) = \frac{e^t}{e^t + 1} = \frac{1}{1 + e^{-t}}$$

```
logistic <- function(t) { return(1 / (1 + exp(-t))) }
df <- data.frame(x=seq(-4, 4, by=0.01))
df$sigma_t <- logistic(df$x)
plot(df$x, df$sigma_t)
```



as a Linear Function

$$t = \beta_0 + \beta_1 x$$

The logistic function can now be rewritten as

$$F(x) = \frac{1}{1 + e^{-(\beta_0 + \beta_1 x)}}$$

We use the same ordinary least squares procedure we used before. That is, we wish to minimize the sum of squared residuals.

Example: Hours Studying Predicting Passing

```
study <- data.frame(
    Hours=c(0.50,0.75,1.00,1.25,1.50,1.75,1.75,2.00,2.25,2.50,2.75,3.00,3.25,3.50,4.00,4.25,4.50,4.75,5.00,5.50),
    Pass=c(0,0,0,0,0,0,1,0,1,0,1,0,1,0,1,1,1,1,1,1)
)
lr.out <- glm(Pass ~ Hours, data=study, family=binomial)
lr.out
```

```
##
## Call:  glm(formula = Pass ~ Hours, family = binomial, data = study)
##
## Coefficients:
## (Intercept)        Hours
##      -4.078        1.505
##
## Degrees of Freedom: 19 Total (i.e. Null);  18 Residual
## Null Deviance:       27.73
## Residual Deviance: 16.06     AIC: 20.06
```

Fitted Values

```
study[1,]
```

```
##   Hours Pass
## 1   0.5    0
```

```
logistic <- function(x, b0, b1) {
    return(1 / (1 + exp(-1 * (b0 + b1 * x)) ))
}
logistic(.5, b0=-4.078, b1=1.505)
```

```
## [1] 0.03470667
```

Of course, the `fitted` function will do the same:

```
study$fitted <- fitted(lr.out)
study[1,]
```

```
##   Hours Pass     fitted
## 1   0.5    0 0.03471034
```

Plotting the Results

```r
binomial_smooth <- function(...) {
    geom_smooth(method = "glm", method.args = list(family = "binomial"), ...)
}

ggplot(study, aes(x=Hours, y=Pass)) + geom_point() +
    binomial_smooth(se=FALSE)
```