Linear Regression

Intilaq Data Science Bootcamp

Introduction to ML & Supervised learning

Machine learning Intro

A branch of artificial intelligence, concerned with the design and development of algorithms that allow computers to **evolve behaviors based on empirical data.**

As intelligence requires knowledge, it is necessary for the computers to acquire knowledge.

The success of machine learning system also depends on the algorithms.

The algorithms control the search to find and build the knowledge structures.

The learning algorithms should extract useful information from training examples.

Data and types of Data

- A gathered body of facts.
- Data is the central thread of any activity.
- Understanding the nature of data is most fundamental for proper and effective use of statistical skills.
- Data types:
  - **Categorical** (Qualitative). Could be:

    Ordinal : e.g. Education: Undergrad/ Graduate/ Postgrad/ Doctorate.

    Nominal: e.g. Subject: Physics/ Math/ Chemistry/ Biology.
  - Numerical (Quantitative)
    - Continuous: e.g. temprerature, gas volume in tank,
    - Discrete: # of days freezing

Supervised and Unsupervised Learning

Supervised learning is the machine learning task of learning a function that maps an input to an output based on example input-output pairs.[1]

Examples of addressed problems: Regression, Classification problems.

Unsupervised learning is the machine learning task of inferring a function that describes the structure of "unlabeled" data.

Examples of addressed problems: Clustering problems, NLP problems.

Reinforcement learning is the area of machine learning concerned with how software agents ought to take actions in an environment so as to maximize some notion of cumulative reward.

Application fields: Robotics, AI.

Intro to Regression Analysis

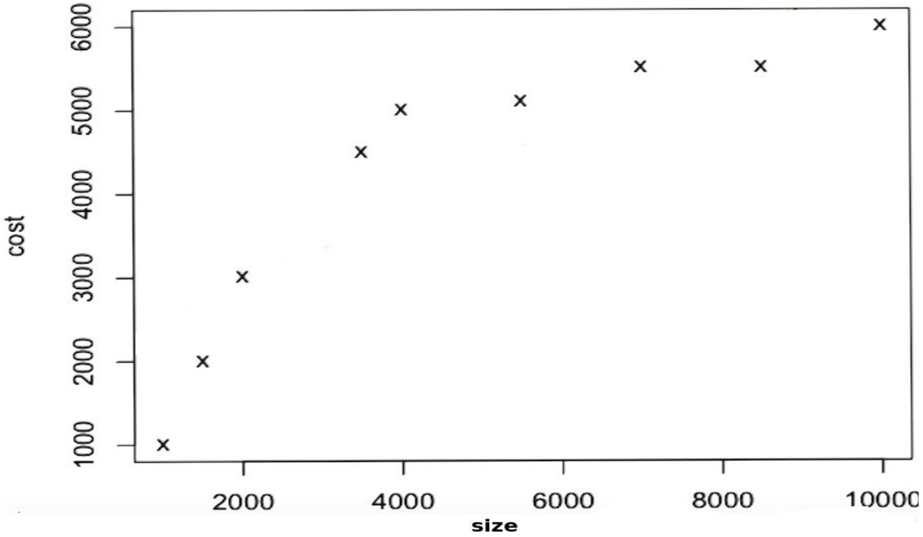
Regression Analysis

Regression analysis is a tool for building statistical models that characterize relationships among a dependent variable and one or more independent variables, all of which are numerical.

Output type: numerical/ categorical.

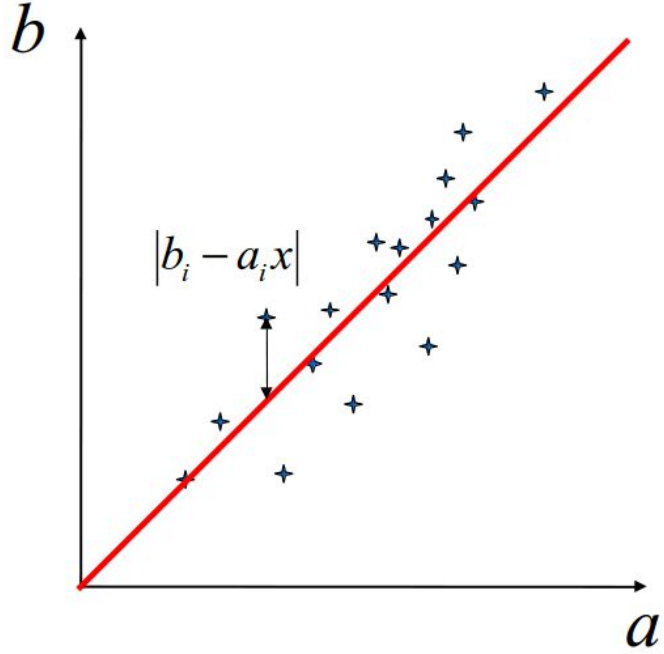
Output type: numerical

Example: Predict the housing prices given the property size in square ft.



Simple Linear Regression

- Problem: the data does not go through a line.
- Errors: $e_i = b_i - a_i x$
- We need to find the line that minimizes the sum of squared errors: $\sum_i (b_i - a_i x)^2$



Simple Linear Regression

The equation that describes how y is related to x and an error term is called the regression model:

$$y = \beta_0 + \beta_1 x + \varepsilon$$

Where :

- b0 and b1 are called parameters of the model,
- e is a random variable called the error term.

The simple linear regression equation is:

$$E(y) = \beta_0 + \beta_1 x$$

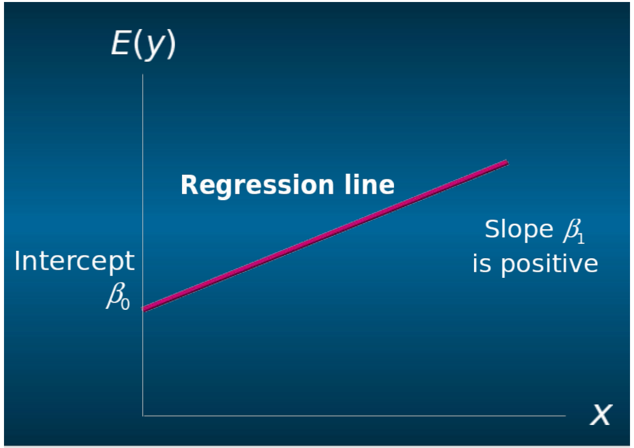
Where:

The graph of the regression equation is a straight line.

b0 is the y intercept of the regression line.

b1 is the slope of the regression line.

E(y) is the expected value of y for a given x value.



Simple Linear Regression

The estimated simple linear regression equation:

$$\hat{y} = b_0 + b_1 x$$

Where:

- The graph is called the estimated regression line.
- b0 is the y intercept of the line.
- b1 is the slope of the line.
- ŷ is the estimated value of y for a given x value.

Ordinary Least Squares (OLS)

Supervised and Unsupervised Learning

Least squares criterion: $\min \sum (y_i - \hat{y}_i)^2$

where:

yi = observed value of the dependent variable for the ith observation.

ŷi = estimated value of the dependent variable for the ith observation.

When substituting the value of ŷi, and imposing the derivative of the above quantity to 0, we get:

$$b_1 = \frac{\sum (x_i - \bar{x})(y_i - \bar{y})}{\sum (x_i - \bar{x})^2}$$

and

$$b_0 = \bar{y} - b_1 \bar{x}$$

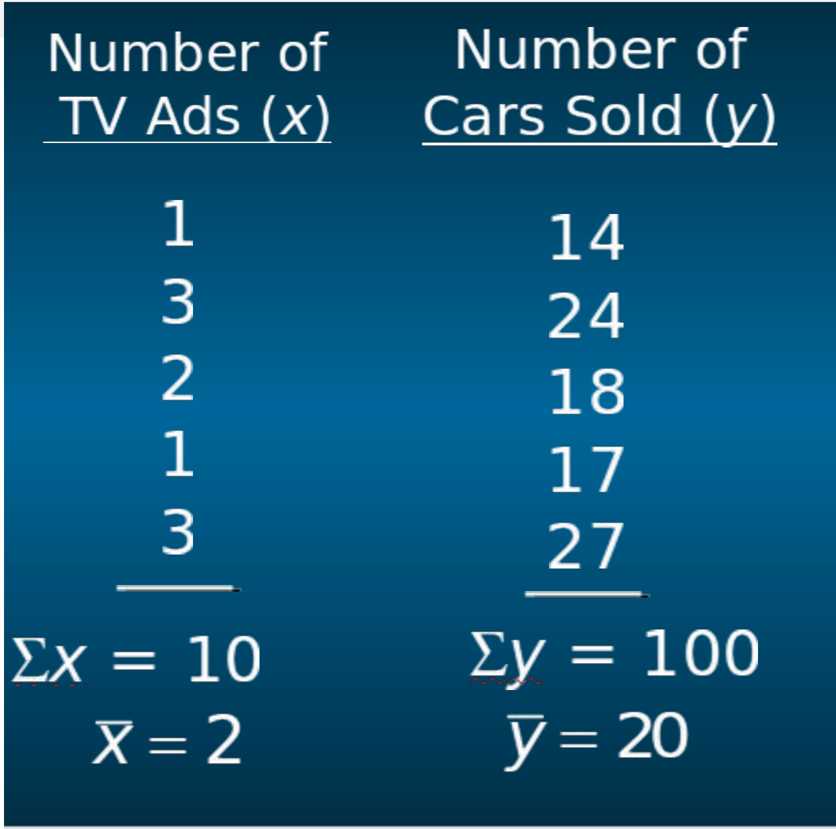
_ where:

x = mean value for independent variable.

y= mean value for dependent variable.

Example

Reed Auto periodically has a special week-long sale. As part of the advertising campaign Reed runs one or more television commercials during the weekend preceding the sale. Data from a sample of 5 previous sales are shown on the table:



Number of TV Ads (x)	Number of Cars Sold (y)
1	14
3	24
2	18
1	17
3	27
$\Sigma x = 10$	$\Sigma y = 100$
$\bar{x} = 2$	$\bar{y} = 20$

Multivariate Regression Models

Multivariate Regression Models

In the “real world” one explanatory variable is not enough

The general multivariate regression model with K independent variables is:

$$Y_i = \beta_0 + \beta_1 X_{1i} + \beta_2 X_{2i} + \ldots + \beta_K X_{Ki} + \varepsilon_i \quad (i = 1,2,\ldots,N)$$

Matrix representation is used to simplify the modeling.

The principle of OLS still stands with multivariate regression.

Total, Explained, and Residual Sums of Squares

$$\mathrm{TSS} = \sum_{i=1}^{N} (Y_i - \bar{Y})^2$$

$$\sum_i (Y_i - \bar{Y})^2 = \sum_i (\hat{Y}_i - \bar{Y})^2 + \sum_i e_i^2$$

TSS = ESS + RSS : This is usually called the decomposition of variance.

Model Evaluation

$$R^2 = \frac{\text{ESS}}{\text{TSS}} = 1 - \frac{\text{RSS}}{\text{TSS}} = 1 - \frac{\sum e_i^2}{\sum (Y_i - \bar{Y})^2}$$

OLS in Matrix form

OLS in Matrix form

Let X be an n × k matrix where we have observations on k independent variables for n observations.

Let y be an n × 1 vector of observations on the dependent variable.

Let E be an n × 1 vector of disturbances or errors.

Let β be an k × 1 vector of unknown population parameters that we want to estimate.

$$\begin{bmatrix} Y_1 \\ Y_2 \\ \vdots \\ \vdots \\ Y_n \end{bmatrix}_{n\times 1} = \begin{bmatrix} 1 & X_{11} & X_{21} & \dots & X_{k1} \\ 1 & X_{12} & X_{22} & \dots & X_{k2} \\ \vdots & \vdots & \vdots & \dots & \vdots \\ \vdots & \vdots & \vdots & \dots & \vdots \\ 1 & X_{1n} & X_{2n} & \dots & X_{kn} \end{bmatrix}_{n\times k} \begin{bmatrix} \beta_1 \\ \beta_2 \\ \vdots \\ \vdots \\ \beta_n \end{bmatrix}_{k\times 1} + \begin{bmatrix} \epsilon_1 \\ \epsilon_2 \\ \vdots \\ \vdots \\ \epsilon_n \end{bmatrix}_{n\times 1}$$

OLS in Matrix form

$$\begin{bmatrix} Y_1 \\ Y_2 \\ \vdots \\ \vdots \\ Y_n \end{bmatrix}_{n\times 1} = \begin{bmatrix} 1 & X_{11} & X_{21} & \dots & X_{k1} \\ 1 & X_{12} & X_{22} & \dots & X_{k2} \\ \vdots & \vdots & \vdots & \dots & \vdots \\ \vdots & \vdots & \vdots & \dots & \vdots \\ 1 & X_{1n} & X_{2n} & \dots & X_{kn} \end{bmatrix}_{n\times k} \begin{bmatrix} \beta_1 \\ \beta_2 \\ \vdots \\ \vdots \\ \beta_n \end{bmatrix}_{k\times 1} + \begin{bmatrix} \epsilon_1 \\ \epsilon_2 \\ \vdots \\ \vdots \\ \epsilon_n \end{bmatrix}_{n\times 1}$$

This can be rewritten more sim~~ply as~~:

$$y = X\beta + \epsilon$$

OLS in Matrix form

The vector of residuals e is given by: $e = y - X\hat{\beta}$

The sum of squared residuals (RSS) is $e'e.^2$

we can write the sum of squared residuals as:

$$\begin{aligned} e'e &= (y - X\hat{\beta})'(y - X\hat{\beta}) \\ &= y'y - \hat{\beta}'X'y - y'X\hat{\beta} + \hat{\beta}'X'X\hat{\beta} \\ &= y'y - 2\hat{\beta}'X'y + \hat{\beta}'X'X\hat{\beta} \end{aligned}$$

where this development uses the fact that the transpose of a scalar is the scalar:

$$y'X\hat{\beta} = (y'X\hat{\beta})' = \hat{\beta}'X'y.$$

OLS in Matrix form

To find the βˆ that minimizes the sum of squared residuals, we need to take the derivative of the last equation with respect to βˆ. This gives us the following equation:

$$\frac{\partial e'e}{\partial \hat{\beta}} = -2X'y + 2X'X\hat{\beta} = 0$$

Side note: For matrix differentiation:

when a and b are K×1 vectors. $\frac{\partial a'b}{\partial b} = \frac{\partial b'a}{\partial b} = a$

when a and b are K×1 vectors: $\frac{\partial b'Ab}{\partial b} = 2Ab = 2b'A$

when A is any symmetric matrix. Note that you can write the derivative as either 2Ab or 2b’A

$$\frac{\partial 2\beta' X' y}{\partial b} = \frac{\partial 2\beta'(X'y)}{\partial b} = 2X'y$$

and

$$\frac{\partial \beta' X' X \beta}{\partial b} = \frac{\partial \beta' A \beta}{\partial b} = 2A\beta = 2X'X\beta$$

OLS in Matrix form

we get what are called the ‘normal equations’: $(X'X)\hat{\beta} = X'y$

If the inverse of (X’X) exists: $(X'X)^{-1}(X'X)\hat{\beta} = (X'X)^{-1}X'y$

And we get:

$$\begin{aligned} I\hat{\beta} &= (X'X)^{-1}X'y \\ \hat{\beta} &= (X'X)^{-1}X'y \end{aligned}$$

Assumptions of Linear Regression

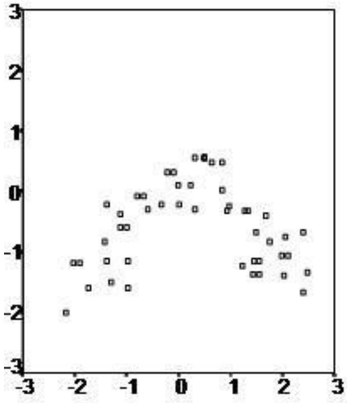
Assumptions of Linear Regression

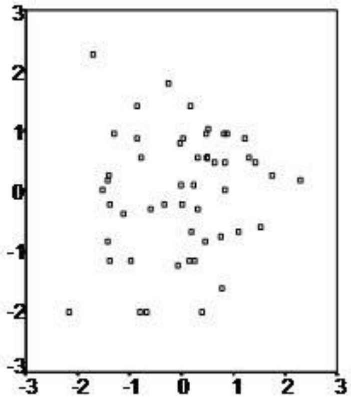
Linear Regression has 5 key assumptions:

- Linear relationship
- Multivariate normality
- No or little multicollinearity
- No auto-correlation
- Homoscedasticity

Linear relationship

- Linear regression needs the relationship between the independent and dependent variables to be linear.
- Linearity assumption can best be tested with scatter plots, the following two examples depict two cases, where no and little linearity is present.

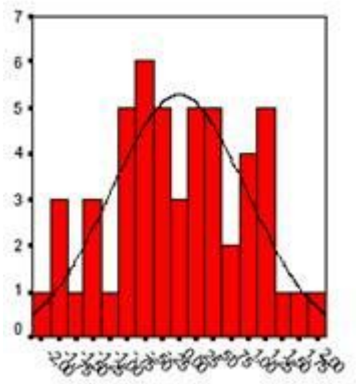


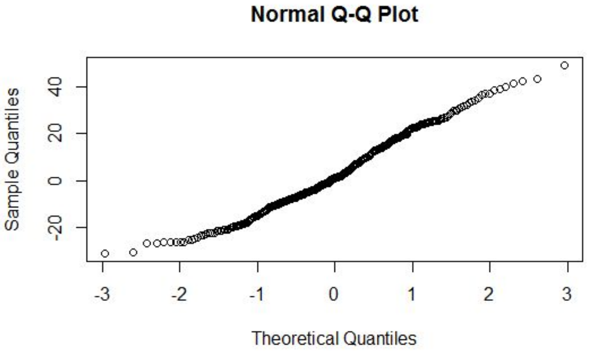


Important: Check for outliers since linear regression is sensitive to outlier effects.

Multivariate Normality

- Linear regression analysis requires all variables to be multivariate normal. This assumption can best be checked with a histogram or a Q-Q-Plot.
- Normality can be checked with a goodness of fit test, e.g., the Kolmogorov-Smirnov test.





Important: When the data is not normally distributed a non-linear transformation (e.g., log-transformation) might fix this issue.

Multicollinearity

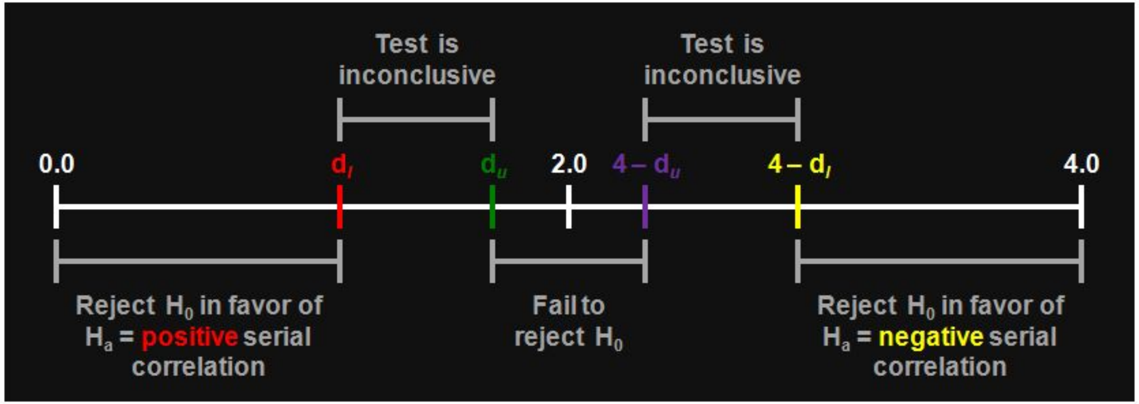
Linear regression assumes that there is little or no multicollinearity in the data. Multicollinearity occurs when the independent variables are too highly correlated with each other. It can be tested with three central criteria:

- Correlation matrix: when computing the matrix of Pearson's Bivariate Correlation among all independent variables the correlation coefficients need to be smaller than 1.
- Tolerance: is defined as $T = 1 – R^2$ for these first step regression analysis. With $T < 0.1$ there might be multicollinearity in the data and with $T < 0.01$ there certainly is.
- Variance Inflation Factor (VIF): the variance inflation factor of the linear regression is defined as VIF = 1/T.

Important: If multicollinearity is found in the data, centering the data (that is deducting the mean of the variable from each score) might help to solve the problem. However, the simplest way to address the problem is to remove independent variables with high VIF values.

Autocorrelation

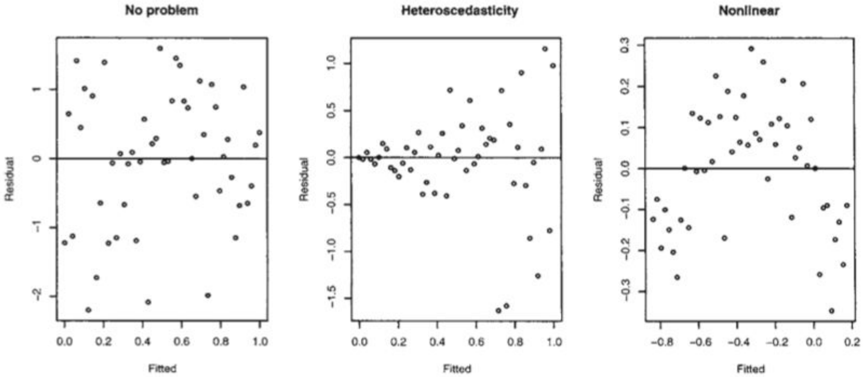
- Linear regression analysis requires that there is little or no autocorrelation in the data.
- Autocorrelation occurs when the residuals are not independent from each other. In other words when the value of y(x+1) is not independent from the value of y(x).
- Linear regression model can be tested for autocorrelation with the Durbin-Watson test.
- Durbin-Watson's d tests the null hypothesis that the residuals are not linearly auto-correlated.


Test is inconclusive
Test is inconclusive
0.0
d_l
d_u
2.0
$4 - d_u$
$4 - d_l$
4.0
Reject H_0 in favor of H_a = positive serial correlation
Fail to reject H_0
Reject H_0 in favor of H_a = negative serial correlation

Homoscedasticity

- The last assumption of the linear regression analysis is homoscedasticity. Homoscedasticity (or homogeneity of variance) is present if all random variables in the sequence or vector have the same finite variance.

- The scatter plot is good way to check whether the data are homoscedastic (meaning the residuals are equal across the regression line):

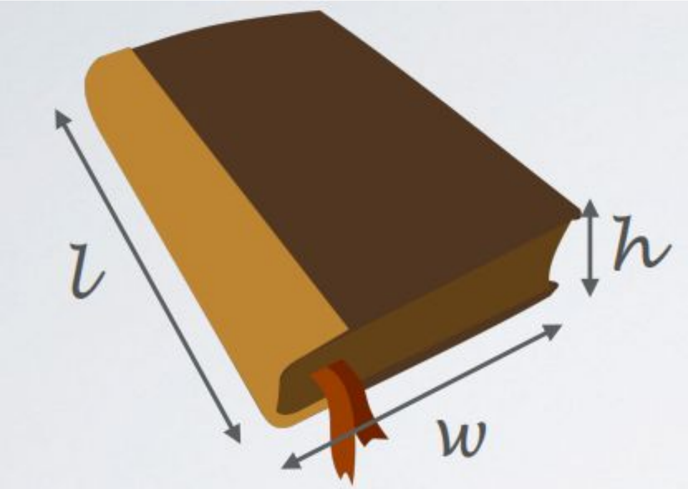


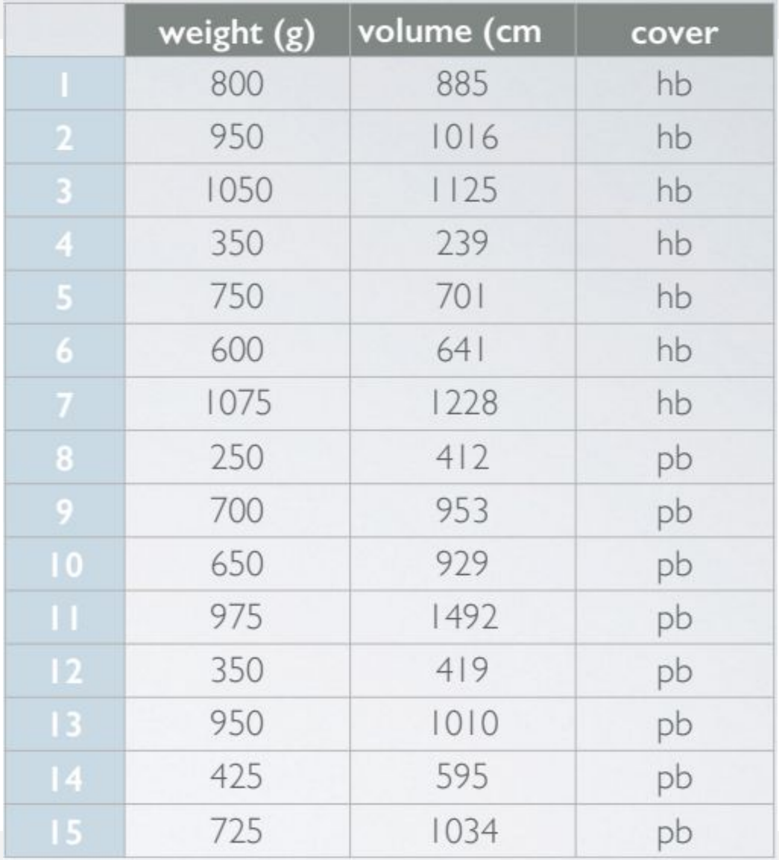
- The Goldfeld-Quandt Test can also be used to test for heteroscedasticity. The test splits the data into two groups and tests to see if the variances of the residuals are similar across the groups. If homoscedasticity is present, a non-linear correction might fix the problem.

Multiple predictors

Predicting weight of books example:

Target variable: Weights of books

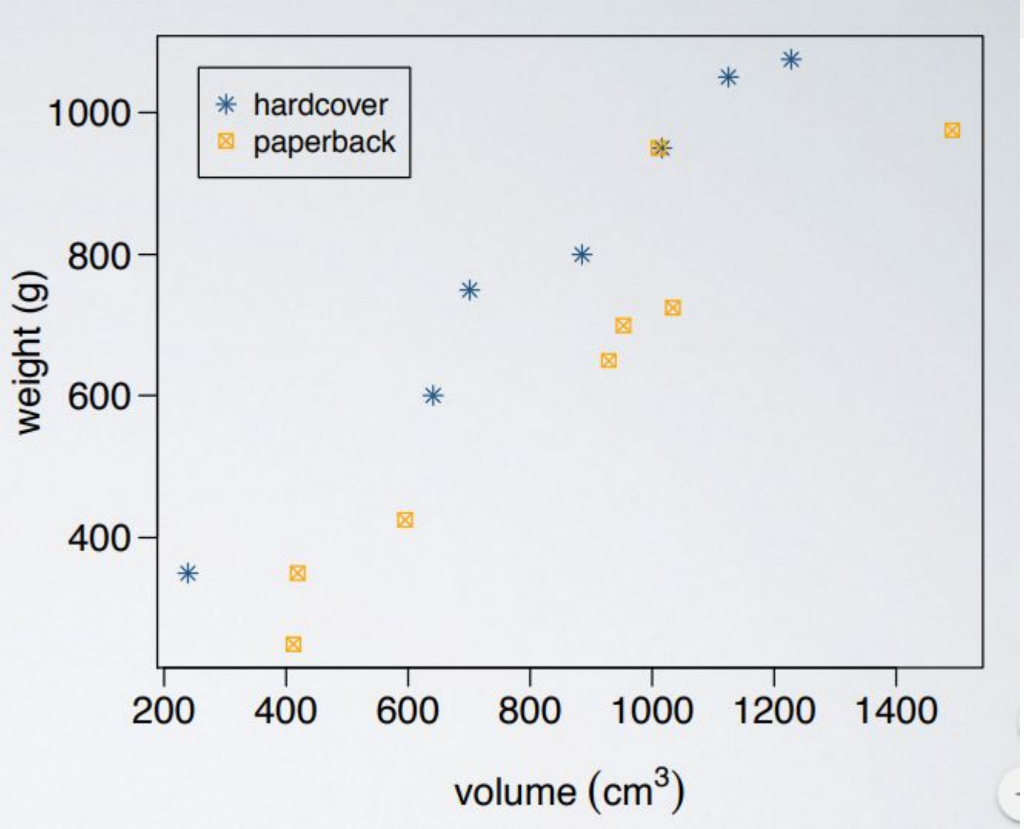




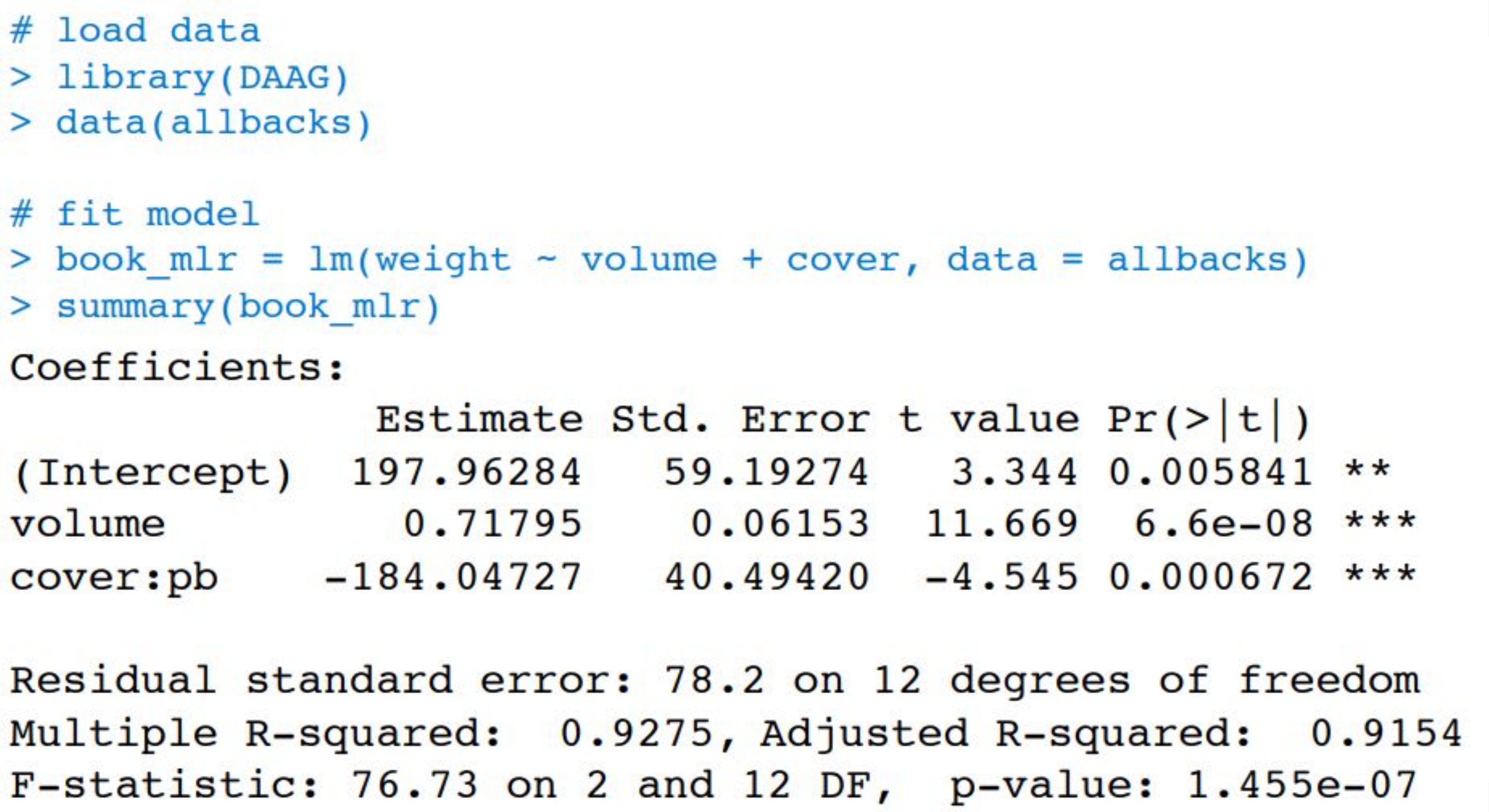
	weight (g)	volume (cm	cover
1	800	885	hb
2	950	1016	hb
3	1050	1125	hb
4	350	239	hb
5	750	701	hb
6	600	641	hb
7	1075	1228	hb
8	250	412	pb
9	700	953	pb
10	650	929	pb
11	975	1492	pb
12	350	419	pb
13	950	1010	pb
14	425	595	pb
15	725	1034	pb

Predicting weight of books example:

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

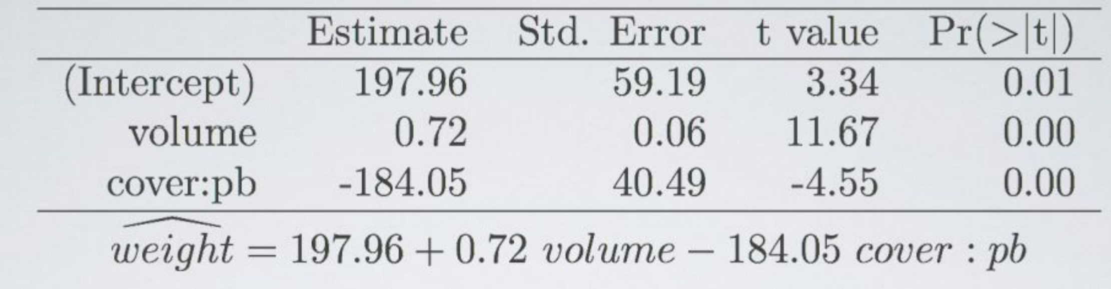


Predicting weight of books example:



```
# load data
> library(DAAG)
> data(allbacks)

# fit model
> book_mlr = lm(weight ~ volume + cover, data = allbacks)
> summary(book_mlr)
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 ***
cover:pb    -184.04727   40.49420  -4.545 0.000672 ***

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
```

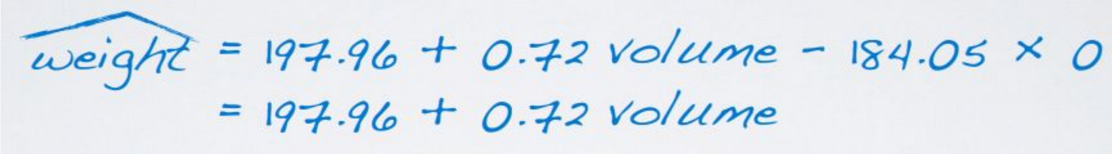
Predicting weight of books example:



	Estimate	Std. Error	t value	Pr(>|t|)
(Intercept)	197.96	59.19	3.34	0.01
volume	0.72	0.06	11.67	0.00
cover:pb	-184.05	40.49	-4.55	0.00

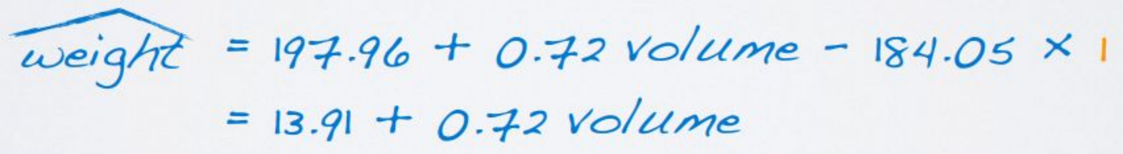
$$\widehat{weight} = 197.96 + 0.72\ volume - 184.05\ cover : pb$$

For hardcover books, plug in 0 for cover:



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

For paperback books, plug in 1 for cover:

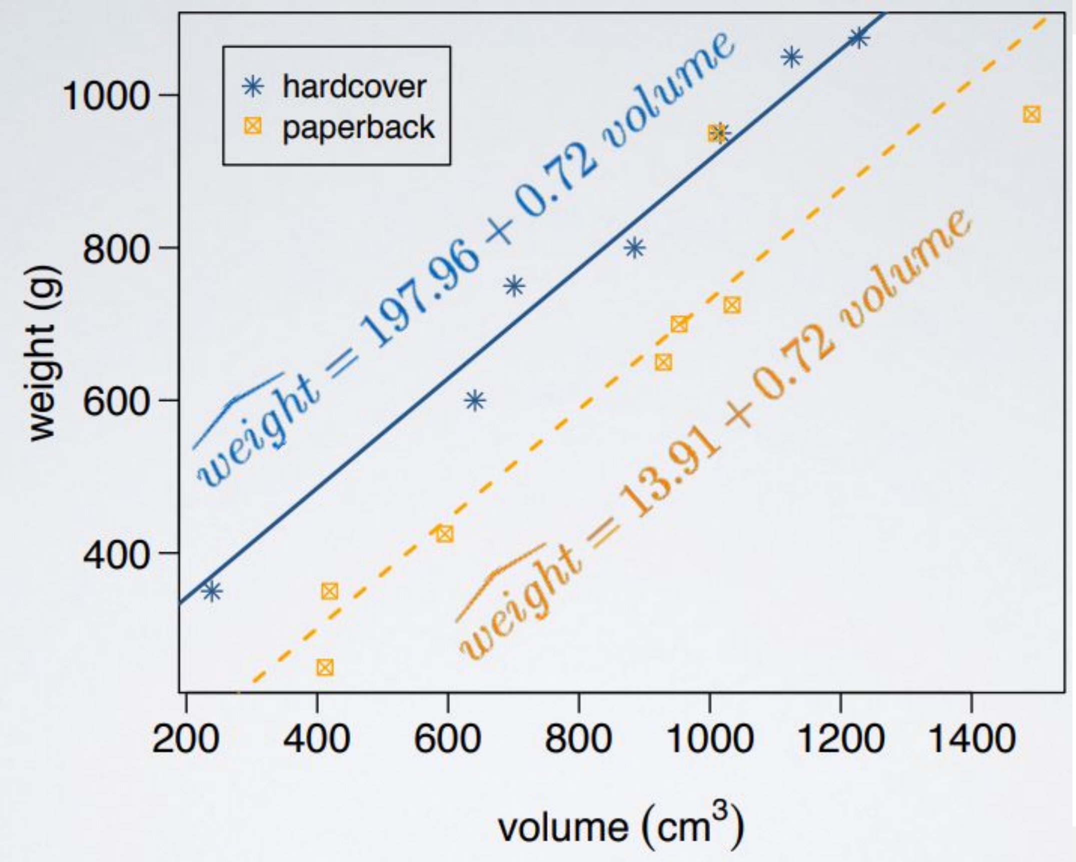


$$\widehat{weight} = 197.96 + 0.72\ volume - 184.05 \times 1$$
$$= 13.91 + 0.72\ volume$$

Interpretation

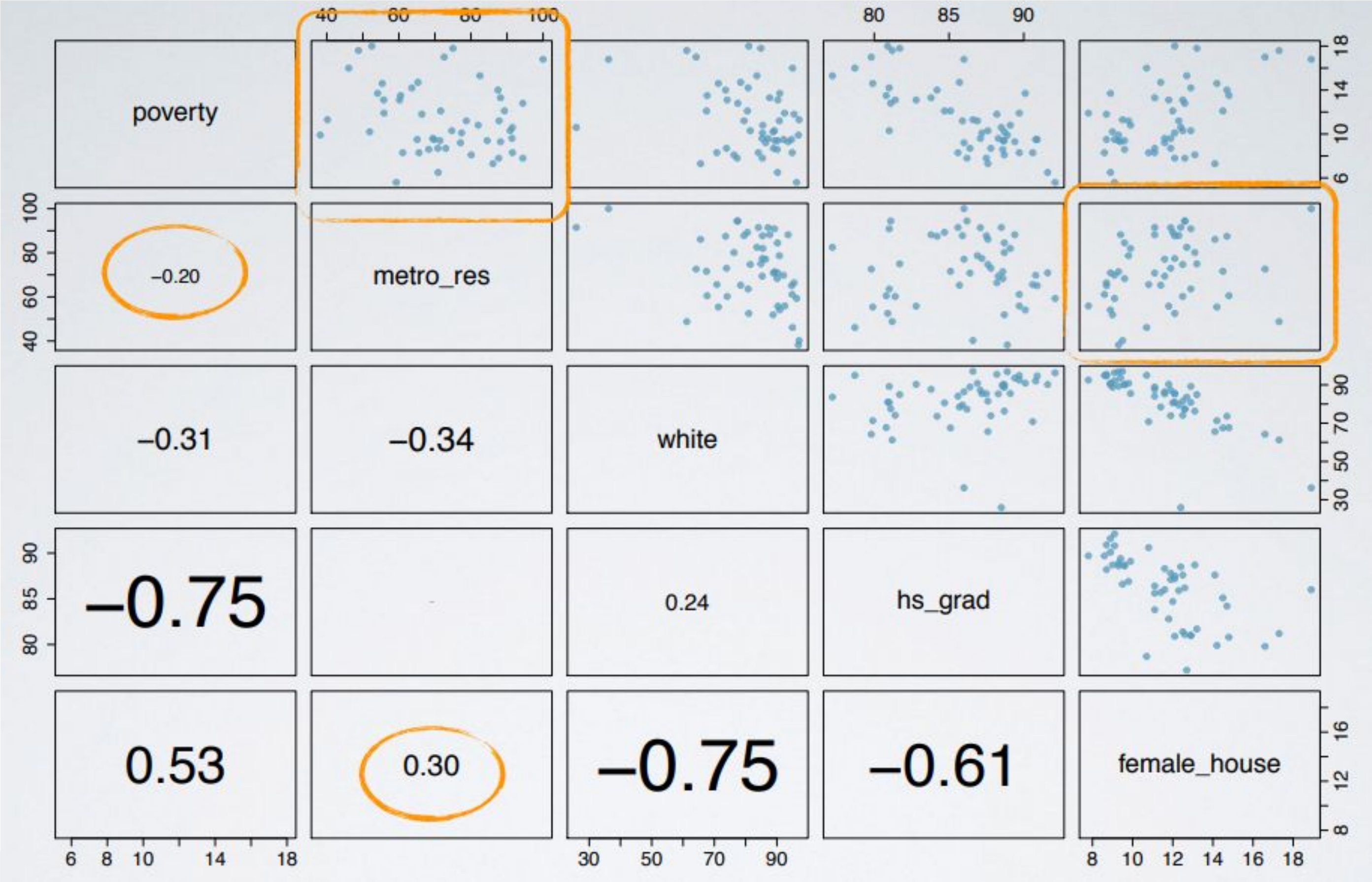
Slope of volume: All else held constant, for each 1 cm3 increase in volume the model predicts the books to be heavier on average by 0.72 grams.

Slope of cover: All else held constant, the model predicts that paperback books weigh 184.05 grams lower than hardcover books, on average

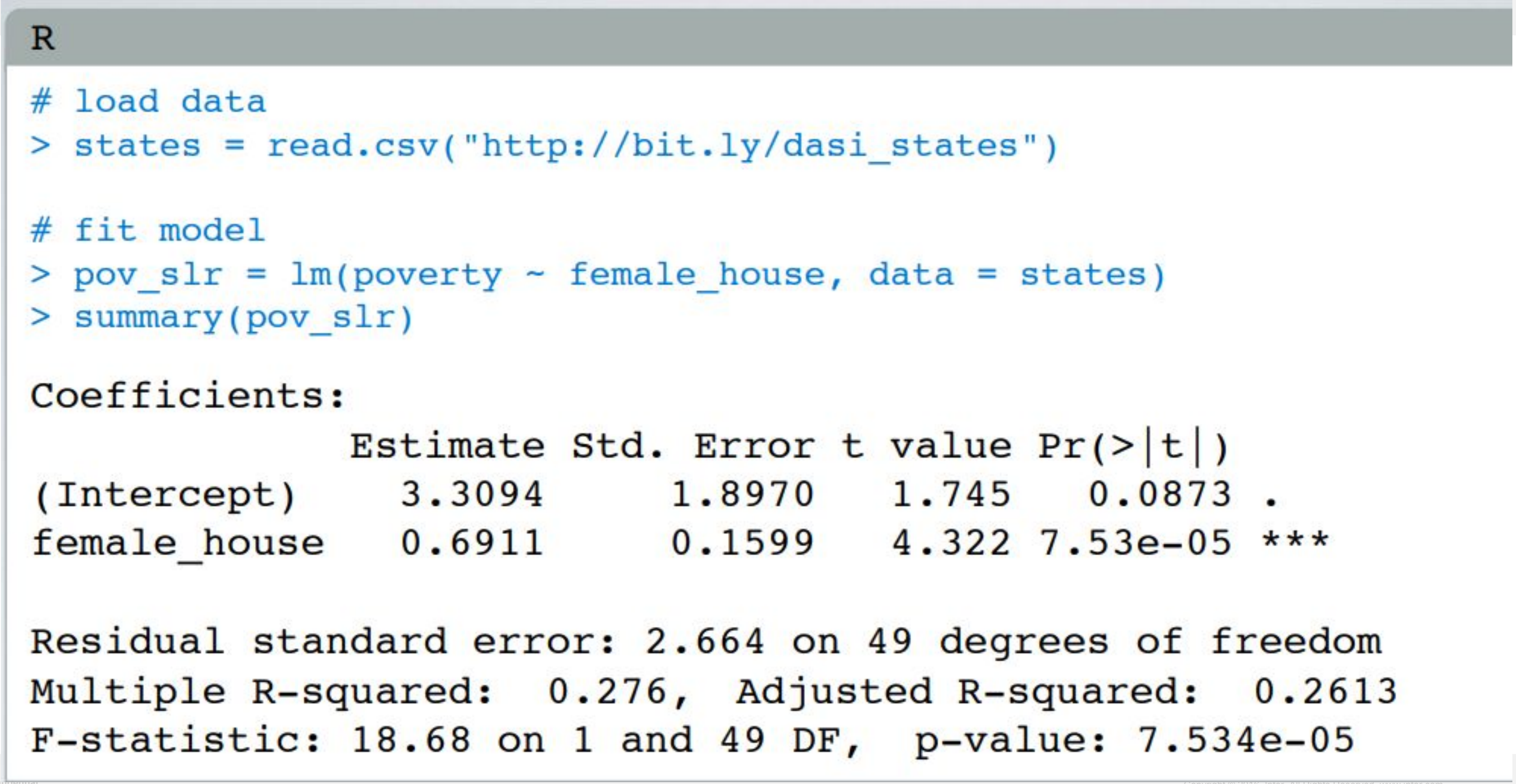


Adjusted R^2

Predicting %Poverty in a state (Pairwise Scatter plot)



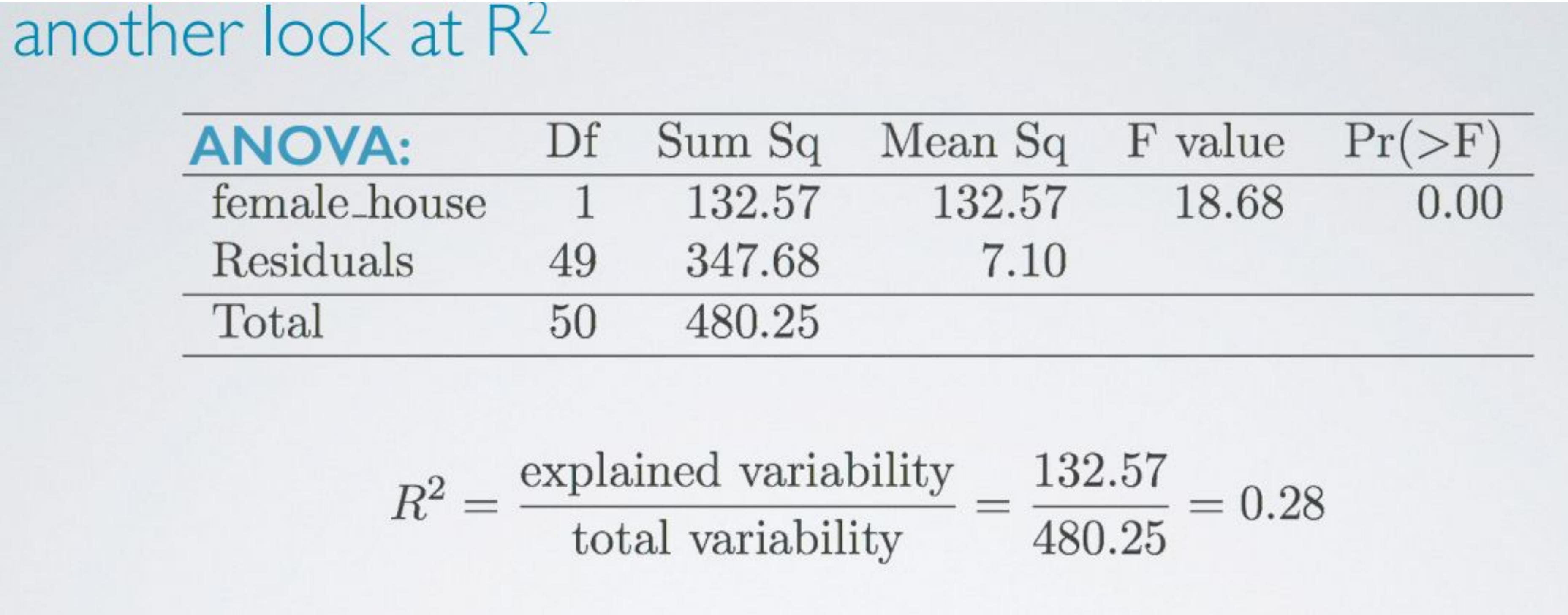
Predicting %Poverty in a state



R

```
# load data
> states = read.csv("http://bit.ly/dasi_states")

# fit model
> pov_slr = lm(poverty ~ female_house, data = states)
> summary(pov_slr)

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 ***

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
```

Predicting %Poverty in a state

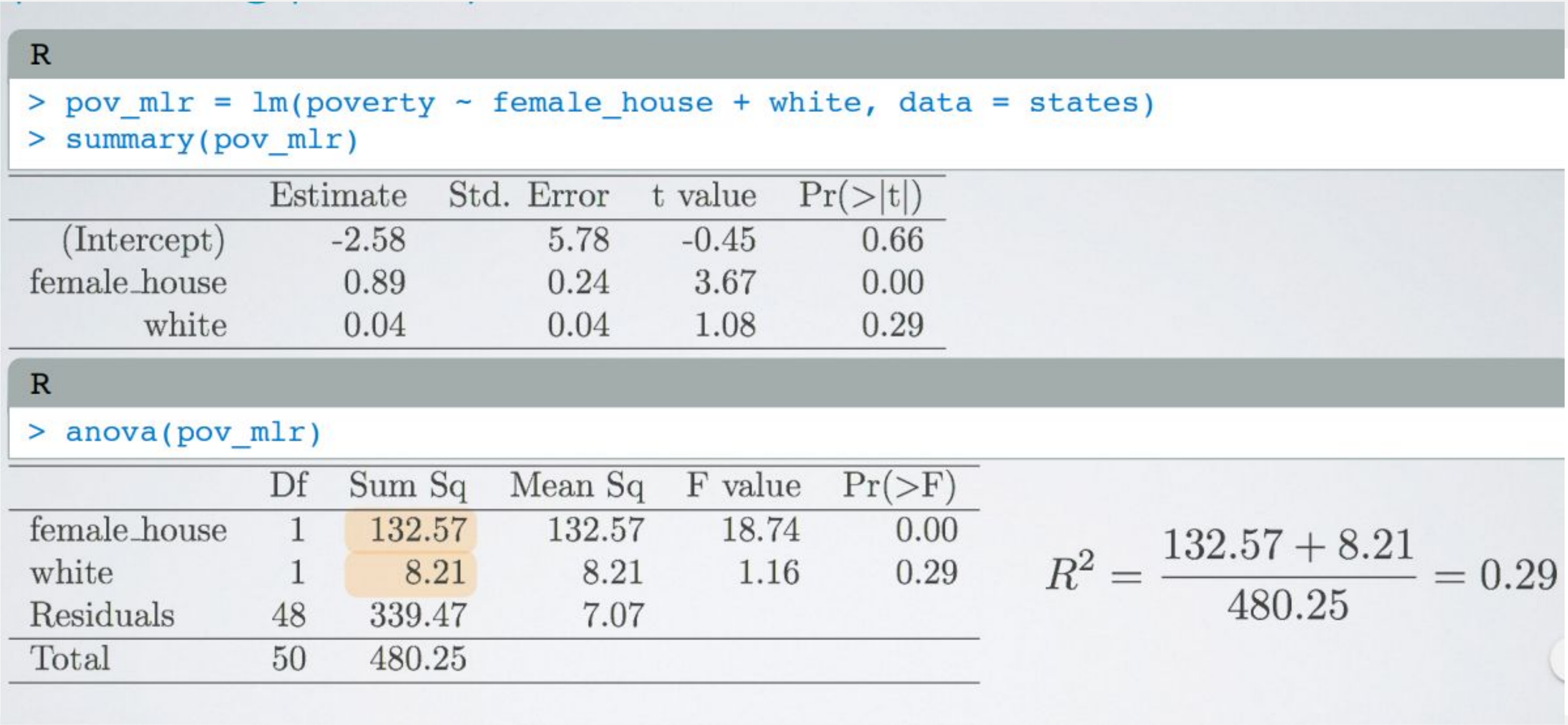


another look at R^2

ANOVA:	Df	Sum Sq	Mean Sq	F value	Pr(>F)
female_house	1	132.57	132.57	18.68	0.00
Residuals	49	347.68	7.10		
Total	50	480.25			

$$R^2 = \frac{\text{explained variability}}{\text{total variability}} = \frac{132.57}{480.25} = 0.28$$

Predicting %Poverty in a state



R

```
> pov_mlr = lm(poverty ~ female_house + white, data = states)
> summary(pov_mlr)
```

	Estimate	Std. Error	t value	Pr(>|t|)
(Intercept)	-2.58	5.78	-0.45	0.66
female_house	0.89	0.24	3.67	0.00
white	0.04	0.04	1.08	0.29

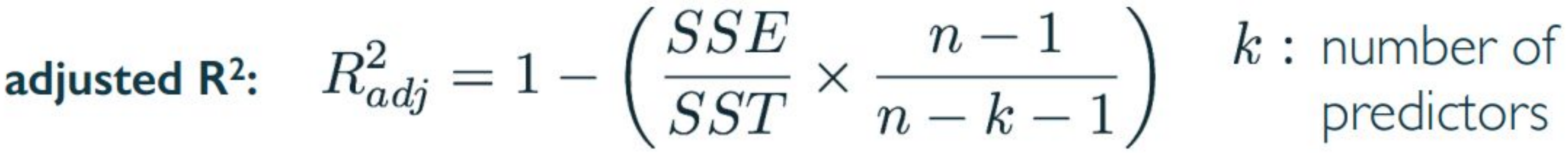
R

```
> anova(pov_mlr)
```

	Df	Sum Sq	Mean Sq	F value	Pr(>F)
female_house	1	132.57	132.57	18.74	0.00
white	1	8.21	8.21	1.16	0.29
Residuals	48	339.47	7.07		
Total	50	480.25			

$$R^2 = \frac{132.57 + 8.21}{480.25} = 0.29$$

Predicting %Poverty in a state

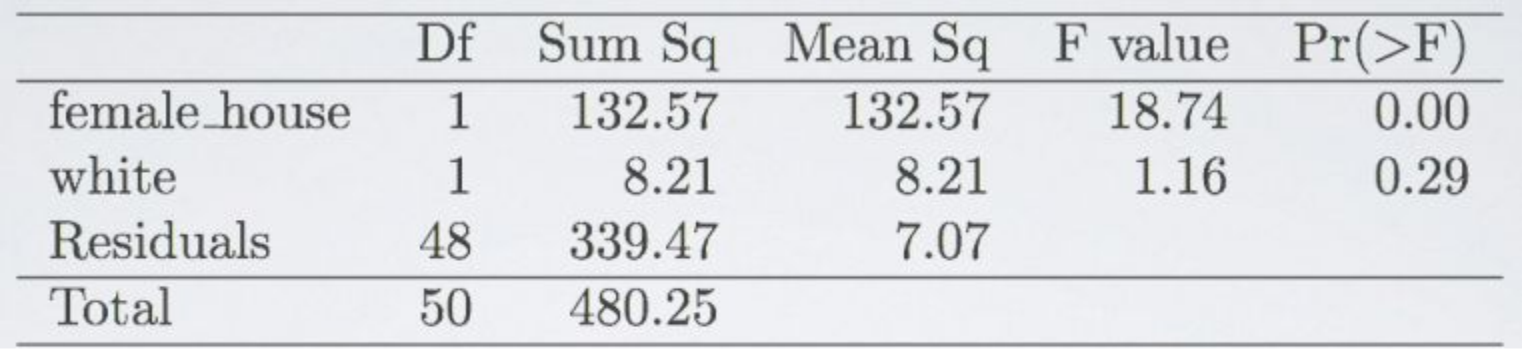


adjusted R²: $$R^2_{adj} = 1 - \left(\frac{SSE}{SST} \times \frac{n-1}{n-k-1} \right)$$

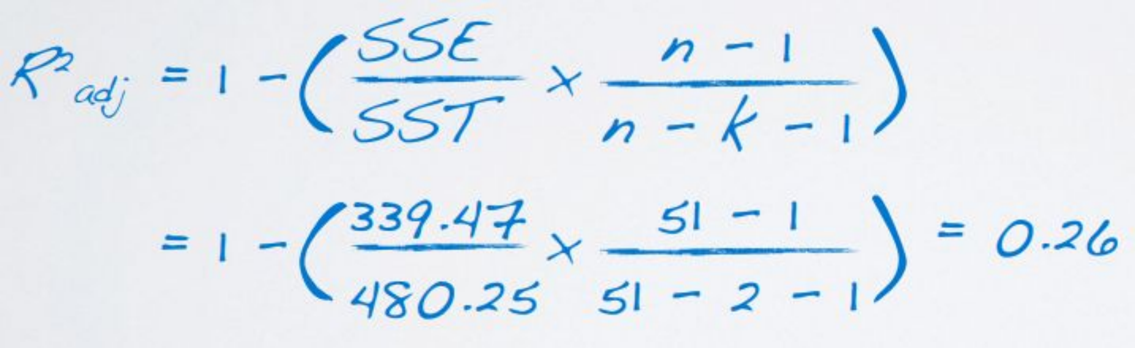
k : number of predictors

n = 51 (50 states + DC).

Predicting %Poverty in a state



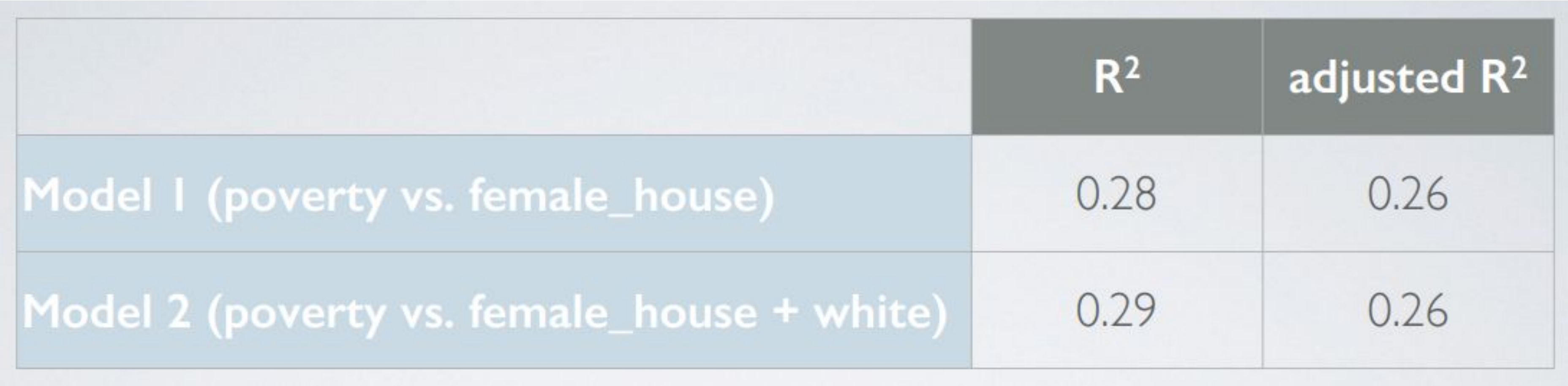
	Df	Sum Sq	Mean Sq	F value	Pr(>F)
female_house	1	132.57	132.57	18.74	0.00
white	1	8.21	8.21	1.16	0.29
Residuals	48	339.47	7.07		
Total	50	480.25			



$$R^2_{adj} = 1 - \left(\frac{SSE}{SST} \times \frac{n-1}{n-k-1} \right)$$

$$= 1 - \left(\frac{339.47}{480.25} \times \frac{51-1}{51-2-1} \right) = 0.26$$

Predicting %Poverty in a state



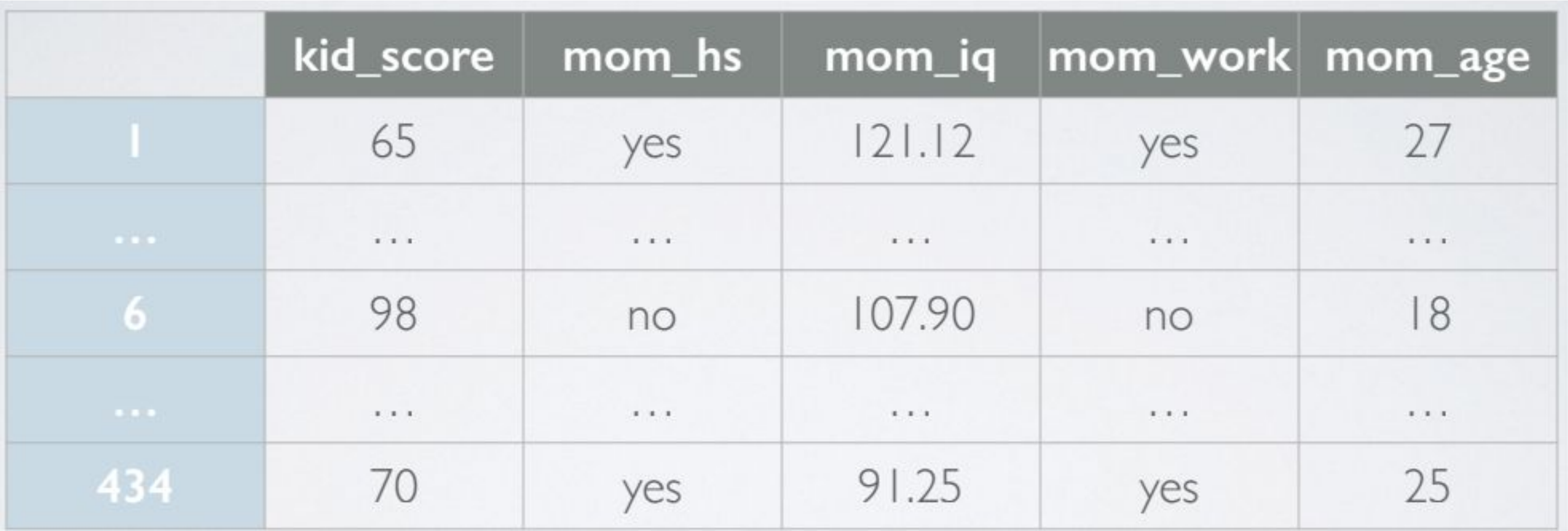
	R^2	adjusted R^2
Model 1 (poverty vs. female_house)	0.28	0.26
Model 2 (poverty vs. female_house + white)	0.29	0.26

- When any variable is added to the model R2 increases.
- But if the added variable doesn’t really provide any new information, or is completely unrelated, adjusted R2 does not increase.

Inference for a Multiple Linear Model

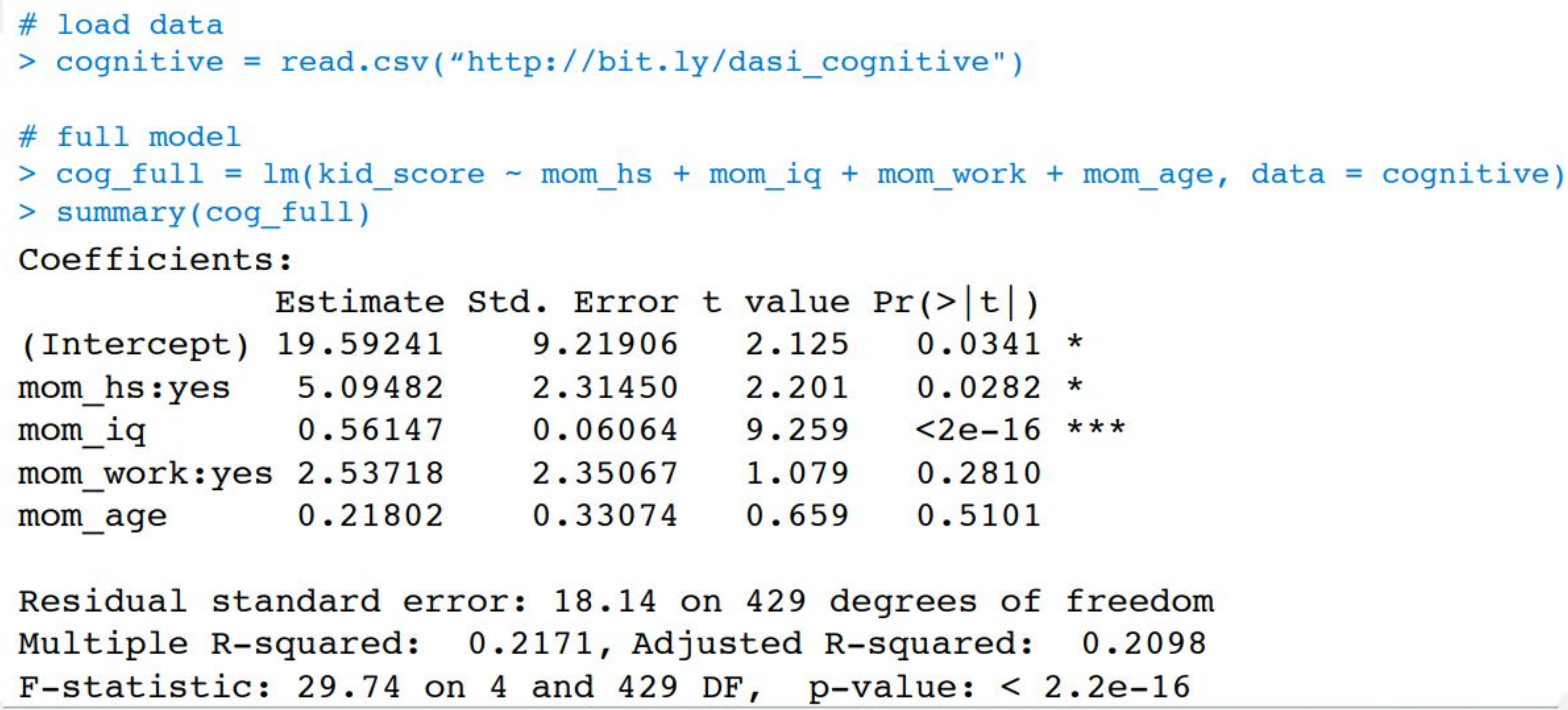
Inference for a Multiple Linear Model

Data: Cognitive test scores of three- and four-year-old children and characteristics of their mothers (from a subsample from the National Longitudinal Survey of Youth):



	kid_score	mom_hs	mom_iq	mom_work	mom_age
1	65	yes	121.12	yes	27
...	...	...	...	...	...
6	98	no	107.90	no	18
...	...	...	...	...	...
434	70	yes	91.25	yes	25

Inference for a Multiple Linear Model

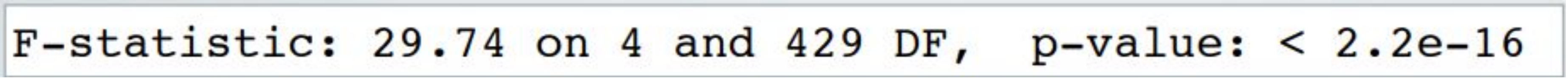


```
# load data
> cognitive = read.csv(“http://bit.ly/dasi_cognitive")

# full model
> cog_full = lm(kid_score ~ mom_hs + mom_iq + mom_work + mom_age, data = cognitive)
> summary(cog_full)
```

```
Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept) 19.59241    9.21906   2.125   0.0341 *
mom_hs:yes   5.09482    2.31450   2.201   0.0282 *
mom_iq       0.56147    0.06064   9.259   <2e-16 ***
mom_work:yes 2.53718    2.35067   1.079   0.2810
mom_age      0.21802    0.33074   0.659   0.5101

Residual standard error: 18.14 on 429 degrees of freedom
Multiple R-squared:  0.2171, Adjusted R-squared:  0.2098
F-statistic: 29.74 on 4 and 429 DF,  p-value: < 2.2e-16
```

Inference of a model as a whole

H0: $\beta1 = \beta1 = \ldots = \beta k= 0$ HA: At least one βi is different than 0.



```
F-statistic: 29.74 on 4 and 429 DF,  p-value: < 2.2e-16
```

Since p-value < 0.05, the model as a whole is significant.

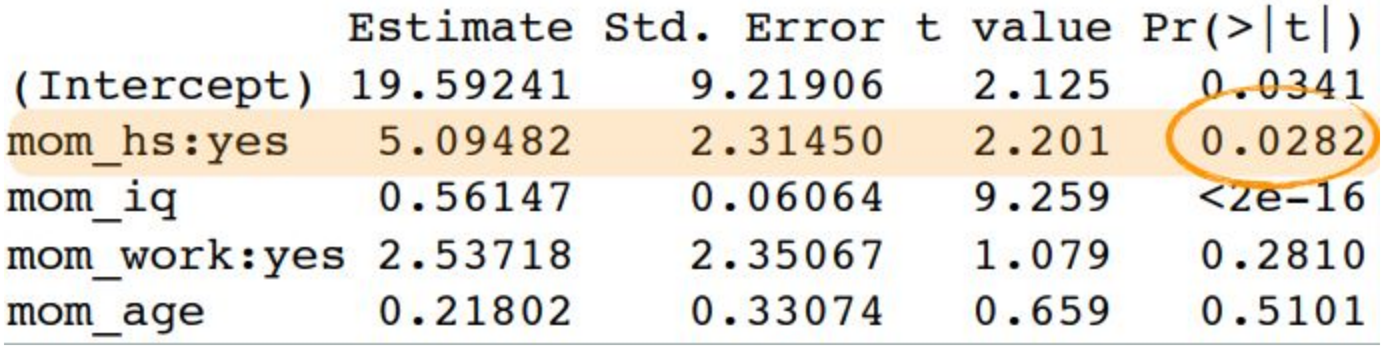
- The F test yielding a significant result doesn’t mean the model fits the data well, it just means at least one of the βs is non-zero.

- The F test not yielding a significant result doesn’t mean individuals variables included in the model are not good predictors of y, it just means that the combination of these variables doesn’t yield a good model.

hypothesis testing for slopes

Is whether or not the mother went to high school a significant predictor of the cognitive test scores of children, given all other variables in the model?

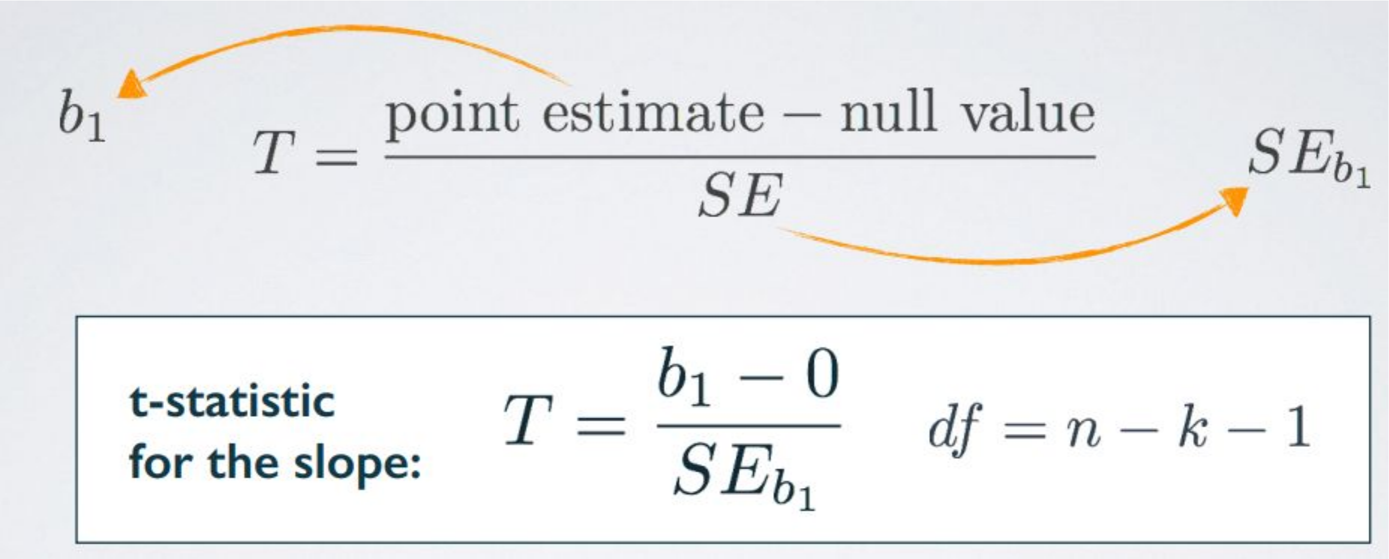
H0: β1 = 0, when all other variables are included in the model
HA: β1 ≠ 0, when all other variables are included in the model



```
              Estimate Std. Error t value Pr(>|t|)
(Intercept)   19.59241    9.21906   2.125   0.0341
mom_hs:yes     5.09482    2.31450   2.201   0.0282
mom_iq         0.56147    0.06064   9.259  <2e-16
mom_work:yes   2.53718    2.35067   1.079   0.2810
mom_age        0.21802    0.33074   0.659   0.5101
```

→ **Whether or not mom went to high school is a significant predictor of the cognitive test scores of children, given all other variables in the model.**

testing for the slope - mechanics



b_1

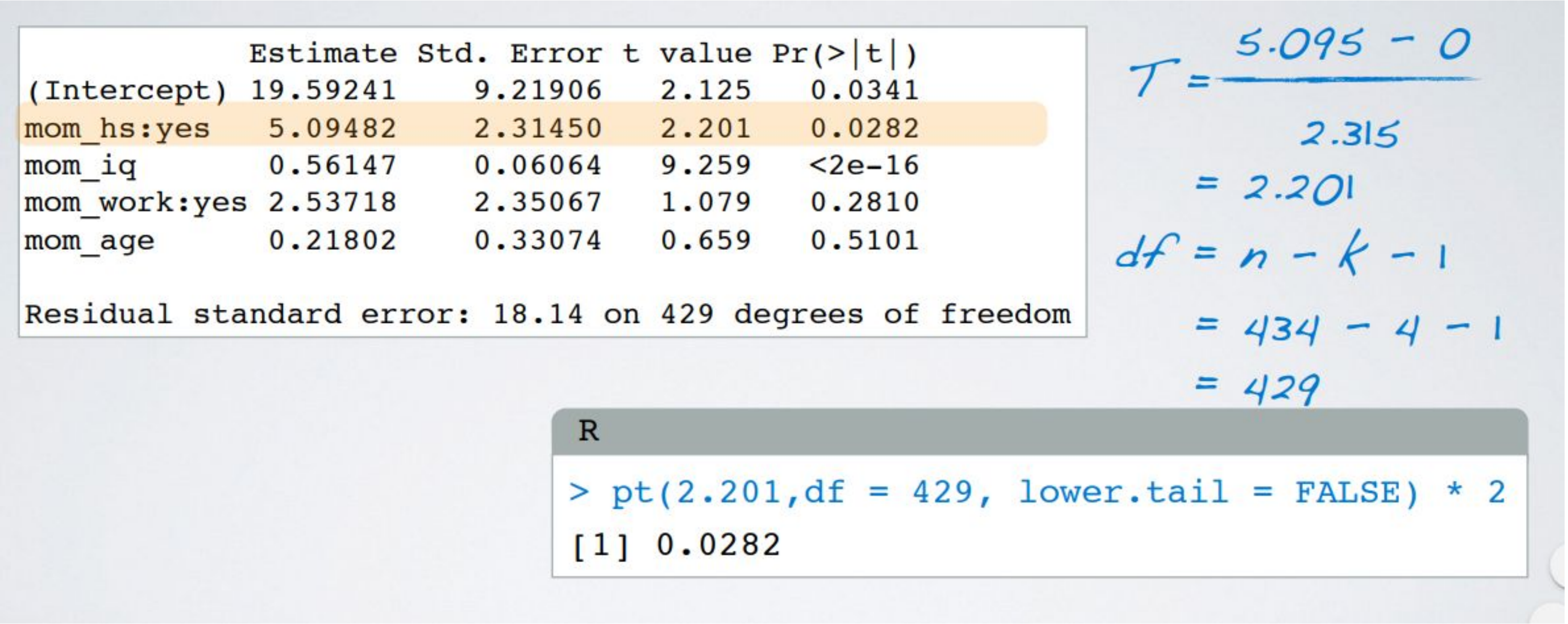
$$T = \frac{\text{point estimate} - \text{null value}}{SE}$$

SE_{b_1}

t-statistic for the slope:

$$T = \frac{b_1 - 0}{SE_{b_1}} \qquad df = n - k - 1$$

Verify the T score and the p-value for the slope of mom_hs.



```
              Estimate Std. Error t value Pr(>|t|)
(Intercept) 19.59241    9.21906   2.125   0.0341
mom_hs:yes   5.09482    2.31450   2.201   0.0282
mom_iq       0.56147    0.06064   9.259   <2e-16
mom_work:yes 2.53718    2.35067   1.079   0.2810
mom_age      0.21802    0.33074   0.659   0.5101

Residual standard error: 18.14 on 429 degrees of freedom
```

$$T = \frac{5.095 - 0}{2.315}$$

$$= 2.201$$

$$df = n - k - 1$$

$$= 434 - 4 - 1$$

$$= 429$$

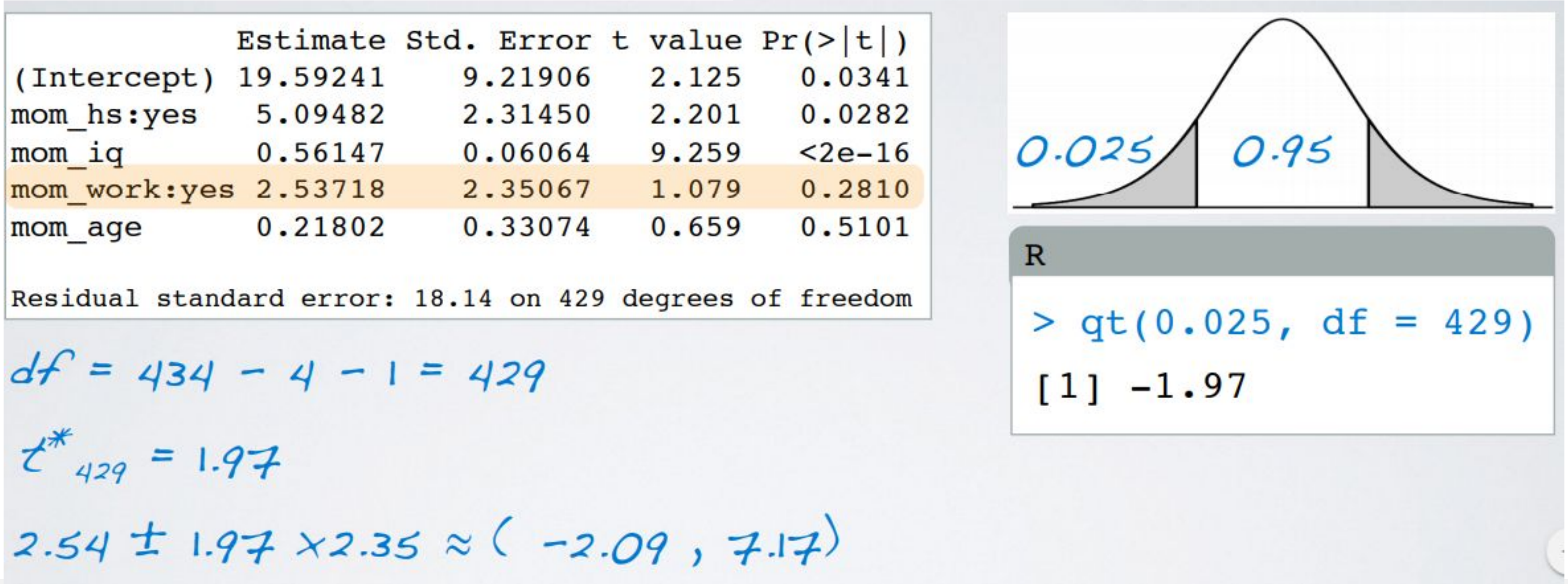
R

```
> pt(2.201,df = 429, lower.tail = FALSE) * 2
[1] 0.0282
```

Confidence Intervals for slopes

point estimate ± margin of error

$$b_1 \pm t^\star_{df} SE_{b_1}$$



```
               Estimate Std. Error t value Pr(>|t|)
(Intercept)    19.59241    9.21906   2.125   0.0341
mom_hs:yes      5.09482    2.31450   2.201   0.0282
mom_iq          0.56147    0.06064   9.259   <2e-16
mom_work:yes    2.53718    2.35067   1.079   0.2810
mom_age         0.21802    0.33074   0.659   0.5101

Residual standard error: 18.14 on 429 degrees of freedom
```

R

```
> qt(0.025, df = 429)
[1] -1.97
```

$df = 434 - 4 - 1 = 429$

$t^*_{429} = 1.97$

$2.54 \pm 1.97 \times 2.35 \approx (-2.09 , 7.17)$

Interpretation of Confidence Intervals for slopes

Interpret the 95% confidence interval for the slope of mom_work. CI: (-2.09, 7.17)

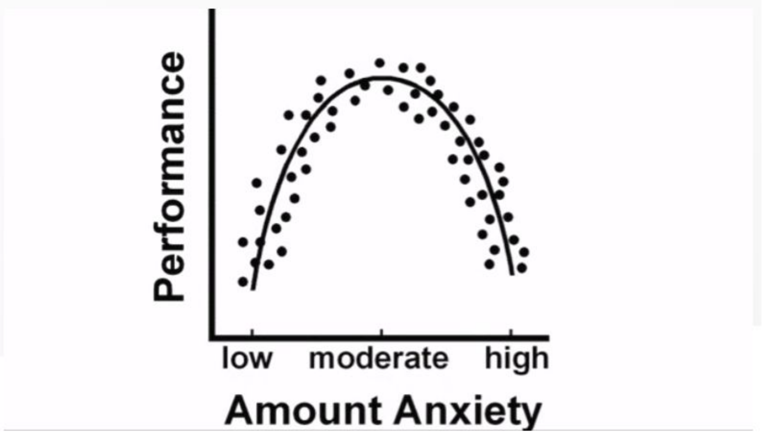
We are 95% confident that, all else being equal, the model predicts that children whose moms worked during the first three years of their lives score 2.09 points lower to 7.17 points higher than those whose moms did not work.

Polynomial Regression

Polynomial Regression

The goal of regression analysis is to model the expected value of a dependent variable y in terms of the value of an independent variable (or vector of independent variables) x. In simple linear regression, the model:

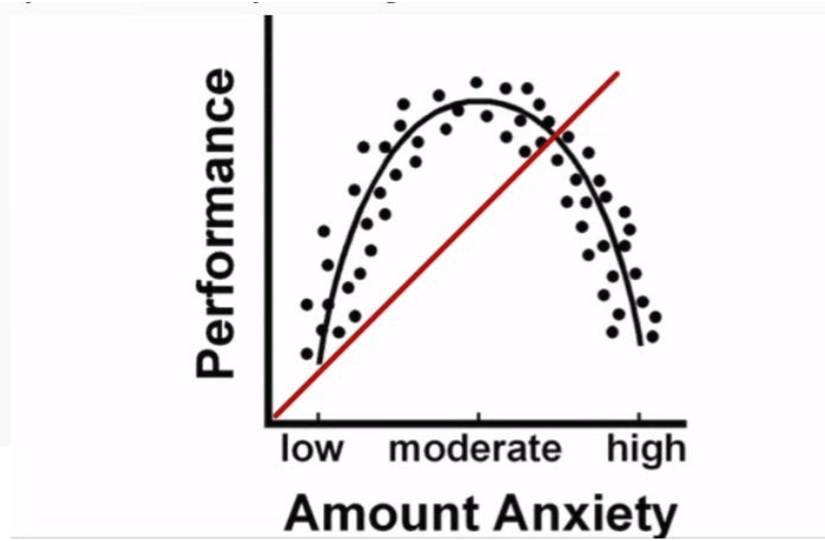
$$y = \beta_0 + \beta_1 x + \varepsilon,$$



Polynomial Regression

In some cases of real life problems, the relation between explanatory and independent variable is not linear, e.g.:

- Performance under stress levels.
- The yield of a chemical synthesis in terms of the temperature
- Case of spread of diseases in function of days passed…



We might propose a polynomial model of the form:

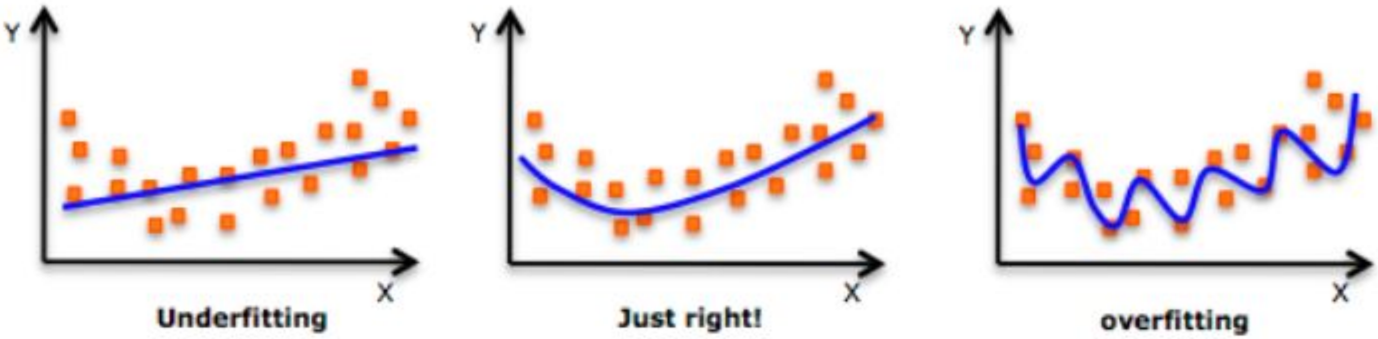
$$y = \beta_0 + \beta_1 x + \beta_2 x^2 + \beta_3 x^3 + \cdots + \beta_n x^n + \varepsilon.$$

Shall we always use higher order polynomials to fit the data set?

- Sadly, no. Basically, we have created a model that fits our training data well but fails to estimate the real relationship among variables beyond the training set.
- Bad performance on the test data. (This problem is called as over-fitting).

→ the model has high variance and low bias.

- Similarly, we have another problem called underfitting, it occurs when our model neither fits the training data nor generalizes on the new data.



Model build strategy

Forward Selection:

Start with linear model:

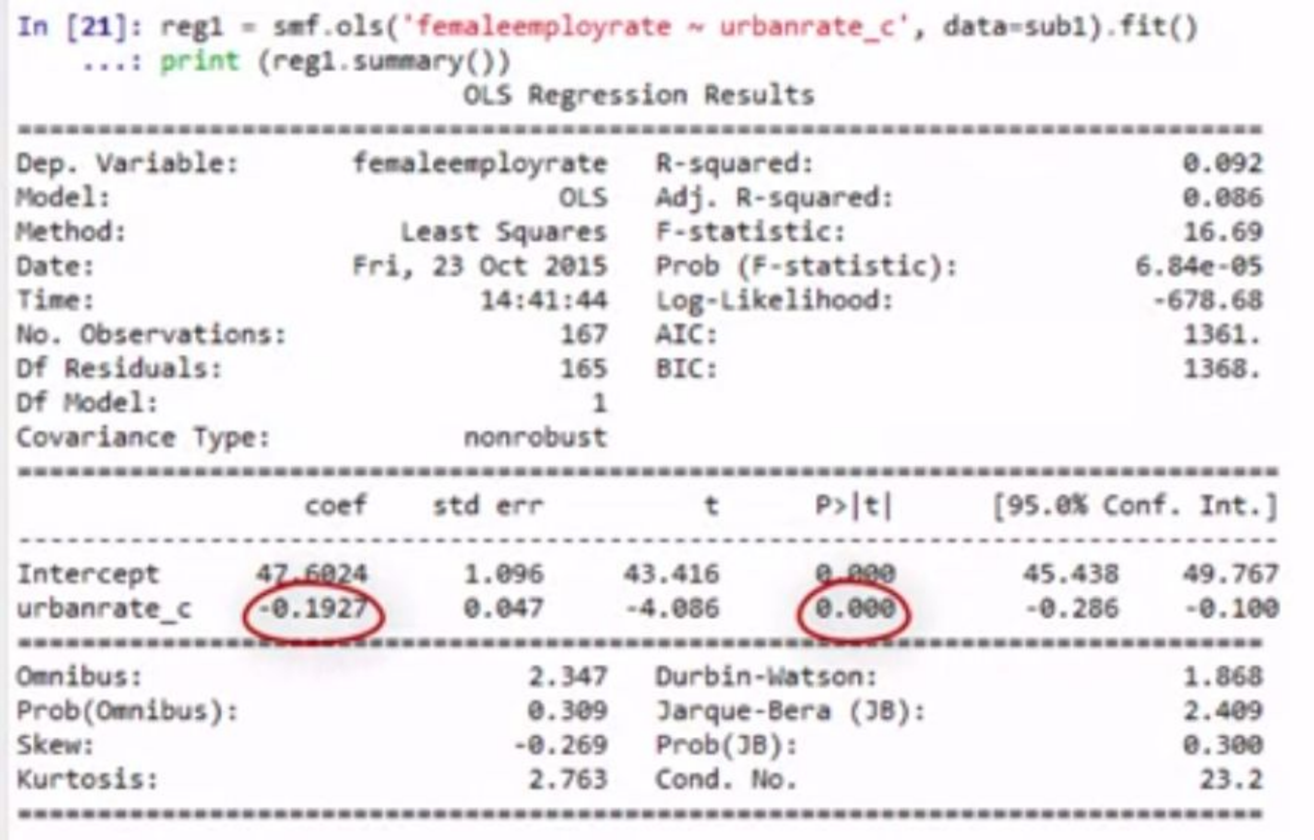
$$y = \beta_0 + \beta_1 x + \varepsilon$$

$$y = \beta_0 + \beta_1 x + \beta_2 x^2 + \varepsilon.$$

$$y = \beta_0 + \beta_1 x + \beta_2 x^2 + \beta_3 x^3 + \cdots + \beta_n x^n + \varepsilon$$

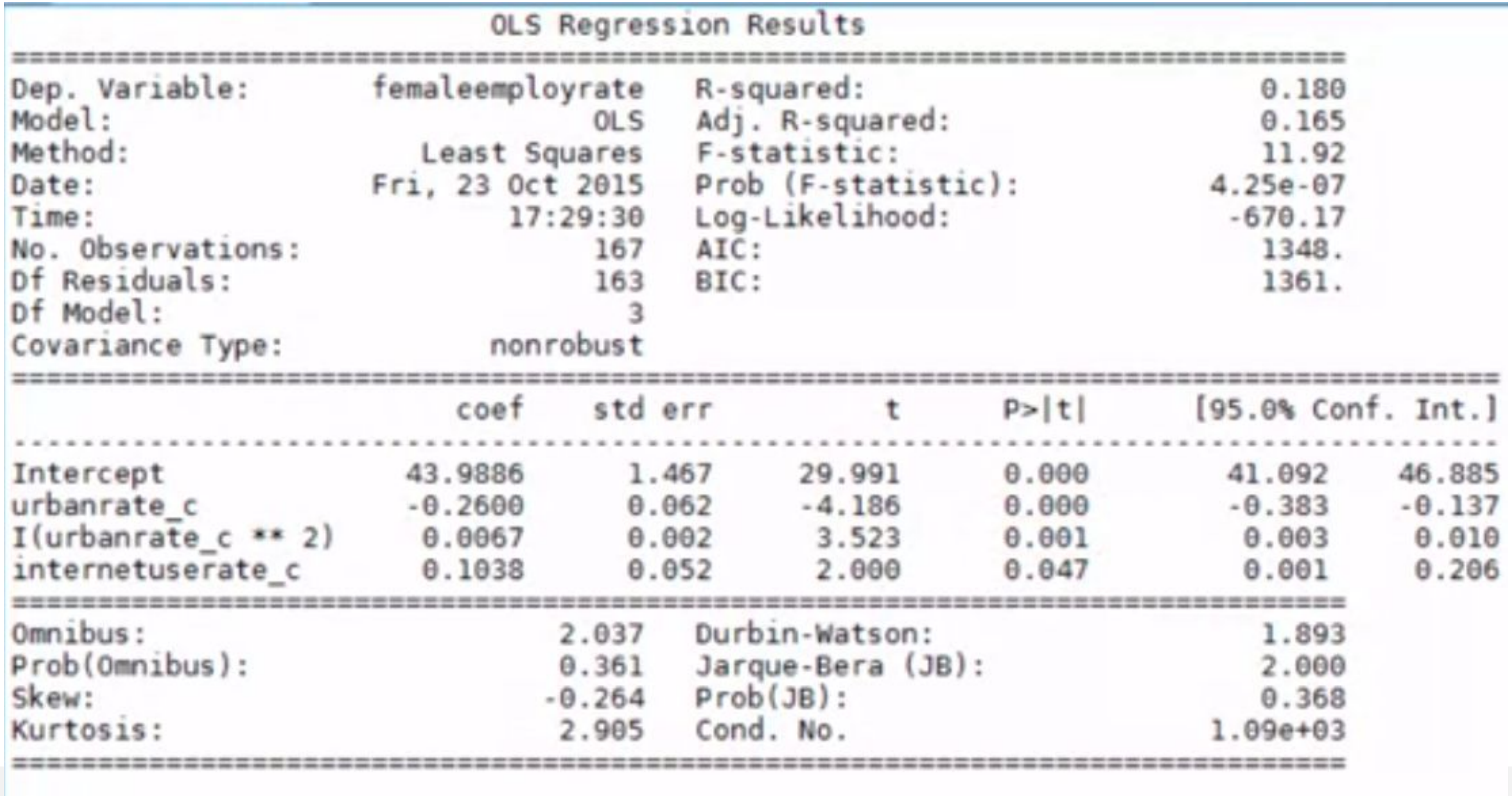
Successively fit model of increasing order until the t-test for the highest order term is not significant.

Example



```
In [21]: reg1 = smf.ols('femaleemployrate ~ urbanrate_c', data=sub1).fit()
    ...: print (reg1.summary())
                            OLS Regression Results
==============================================================================
Dep. Variable:      femaleemployrate   R-squared:                       0.092
Model:                           OLS   Adj. R-squared:                  0.086
Method:                Least Squares   F-statistic:                     16.69
Date:               Fri, 23 Oct 2015   Prob (F-statistic):           6.84e-05
Time:                       14:41:44   Log-Likelihood:                -678.68
No. Observations:                167   AIC:                             1361.
Df Residuals:                    165   BIC:                             1368.
Df Model:                          1
Covariance Type:           nonrobust
===============================================================================
                 coef    std err          t      P>|t|      [95.0% Conf. Int.]
-------------------------------------------------------------------------------
Intercept     47.6024      1.096     43.416      0.000        45.438    49.767
urbanrate_c   -0.1927      0.047     -4.086      0.000        -0.286    -0.100
==============================================================================
Omnibus:                       2.347   Durbin-Watson:                   1.868
Prob(Omnibus):                 0.309   Jarque-Bera (JB):                2.409
Skew:                         -0.269   Prob(JB):                        0.300
Kurtosis:                      2.763   Cond. No.                         23.2
==============================================================================
```

Example



```
                            OLS Regression Results
==============================================================================
Dep. Variable:        femaleemployrate   R-squared:                       0.180
Model:                             OLS   Adj. R-squared:                  0.165
Method:                  Least Squares   F-statistic:                     11.92
Date:                 Fri, 23 Oct 2015   Prob (F-statistic):           4.25e-07
Time:                         17:29:30   Log-Likelihood:                -670.17
No. Observations:                  167   AIC:                             1348.
Df Residuals:                      163   BIC:                             1361.
Df Model:                            3
Covariance Type:             nonrobust
=======================================================================================
                          coef    std err          t      P>|t|      [95.0% Conf. Int.]
---------------------------------------------------------------------------------------
Intercept              43.9886      1.467     29.991      0.000        41.092    46.885
urbanrate_c            -0.2600      0.062     -4.186      0.000        -0.383    -0.137
I(urbanrate_c ** 2)     0.0067      0.002      3.523      0.001         0.003     0.010
internetuserate_c       0.1038      0.052      2.000      0.047         0.001     0.206
==============================================================================
Omnibus:                         2.037   Durbin-Watson:                   1.893
Prob(Omnibus):                   0.361   Jarque-Bera (JB):                2.000
Skew:                           -0.264   Prob(JB):                        0.368
Kurtosis:                        2.905   Cond. No.                     1.09e+03
==============================================================================
```

Centering the dependent variable

The standard (for example quadratic) polynomial models look like this:

$$y = \beta_0 + \beta_1 x + \beta_2 x^2 + \varepsilon.$$

More terms are included with the higher order equations. There are two problems with polynomial fits:

1. When the X values are large, and start well above zero
2. Even when the X values are not large, the parameters of the model are intertwined, so have high collinearity.

Centering the dependent variable

- **Both problems go away (or are much less significant) when the X values are centered.**
- Subtract the mean X from all X values before fitting the model. This can be done as part of nonlinear regression, using this model:

XC = X - Xmean

Y= B0 + B1*XC +B2*XC^2

No free lunch?

Good!

- Fitting the centered model leads to exactly the same curve (unless the regular approach led to math errors).
- Accordingly, the sum-of-squares is the same, as are results of model comparisons.

However!

- The centered equation has reparameterized the model. The parameters have different meanings,

→ different best-fit values (except the first parameter which is the same), different standard errors and confidence intervals, smaller covariances and dependencies, and tighter confidence/ prediction bands.

Regularization, Ridge and Lasso Regression

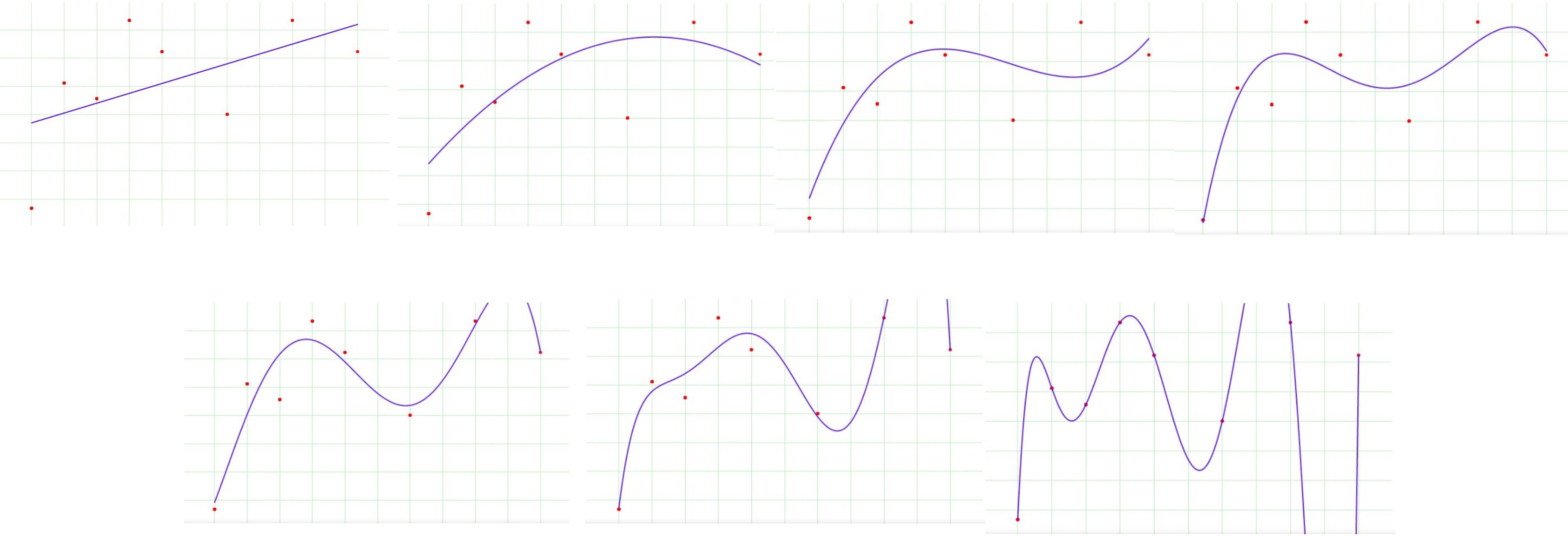
Overfitting

In supervised machine learning, models are trained on a subset of data aka training data. The goal is to compute the target of each training example from the training data.

Overfitting happens when model learns signal as well as noise in the training data and wouldn’t perform well on new data on which model wasn’t trained on.

In the example (next slide), we can see underfitting in first few charts and overfitting in last few:

Overfitting



Regularization

There are few ways you can avoid overfitting your model on training data like cross-validation sampling, reducing number of features, pruning, regularization etc.

Regularization basically adds the penalty as model complexity increases. Regularization parameter (lambda) penalizes all the parameters except intercept so that model generalizes the data and won't overfit:

As the complexity is increasing, regularization will add the penalty for higher terms. This will decrease the importance given to higher terms and will bring the model towards less complex equation.

$$J(\theta) = \frac{1}{2m}\left[\sum_{i=1}^{m}(h_\theta(x^{(i)}) - y^{(i)})^2 + \boxed{\lambda \sum_{j=1}^{n} \theta_j^2}\right]$$

$$\min_{\theta} J(\theta)$$

L2 Regularization

A regression model that uses L1 regularization technique is called Lasso Regression and model which uses L2 is called Ridge Regression.

The key difference between these two is the penalty term.

Ridge regression adds “squared magnitude” of coefficient as penalty term to the loss function. Here the highlighted part represents L2 regularization element:

$$\sum_{i=1}^{n}(y_i - \sum_{j=1}^{p} x_{ij}\beta_j)^2 + \lambda \sum_{j=1}^{p} \beta_j^2$$

L1 Regularization

- **Lasso Regression** (Least Absolute Shrinkage and Selection Operator) adds “absolute value of magnitude” of coefficient as penalty term to the loss function.

- Again, if lambda is zero then we will get back OLS whereas very large value will make coefficients zero hence it will under-fit.

- The key difference between these techniques is that Lasso shrinks the less important feature’s coefficient to zero thus, removing some feature altogether. So, this works well for feature selection in case we have a huge number of features.

- Traditional methods like cross-validation, stepwise regression to handle overfitting and perform feature selection work well with a small set of features but these techniques are a great alternative when we are dealing with a large set of features.

$$\sum_{i=1}^{n}(Y_i - \sum_{j=1}^{p} X_{ij}\beta_j)^2 + \lambda \sum_{j=1}^{p} |\beta_j|$$

Looking up Lambda

Mini lab

Logistic Regression

Why use logistic regression?

- For example: voting, morbidity or mortality, and participation data is not continuous or distributed normally.
- Binary logistic regression is a type of regression analysis where the dependent variable is a dummy variable: coded 0 (did not vote) or 1(did vote)

$$Y|X = B_0 + B_1X_1 + \varepsilon$$

$$E(\hat{Y}|X) = B_0 + B_1X_1$$

an expected value is a mean, so

$$(\hat{Y} = \hat{\pi}) = P_{Y=1}|X$$

The predicted value equals the proportion of observations for which Y|X = 1; P is the probability of Y = 1(A SUCCESS) given X, and Q = 1- P (A FAILURE) given X.

Why use logistic regression?

For any value of X, only two errors (Y-Y^) are possible, 1-Pi^ AND 0-Pi^. Which occur at rates P|X AND Q|X and with variance (P|X)(Q|X)

In the OLS regression:

Y = b0 +b1 X + e ; where Y = (0, 1)

- The error terms are heteroskedastic
- e is not normally distributed because Y takes on only two values
- The predicted probabilities can be greater than 1 or less than 0

Why use logistic regression?

The "logit" model solves these problems:

ln[p/(1-p)] = b0+ b1X + e

- p is the probability that the event Y occurs, p(Y=1)
- p/(1-p) is the "odds ratio"
- ln[p/(1-p)] is the log odds ratio, or "logit"

** Odds ratios range from 0 to positive infinity
** Odds ratio: P/Q is an odds ratio; less than 1 = less than .50 probability, greater than 1 means greater than .50 probability

Why use logistic regression?

More:

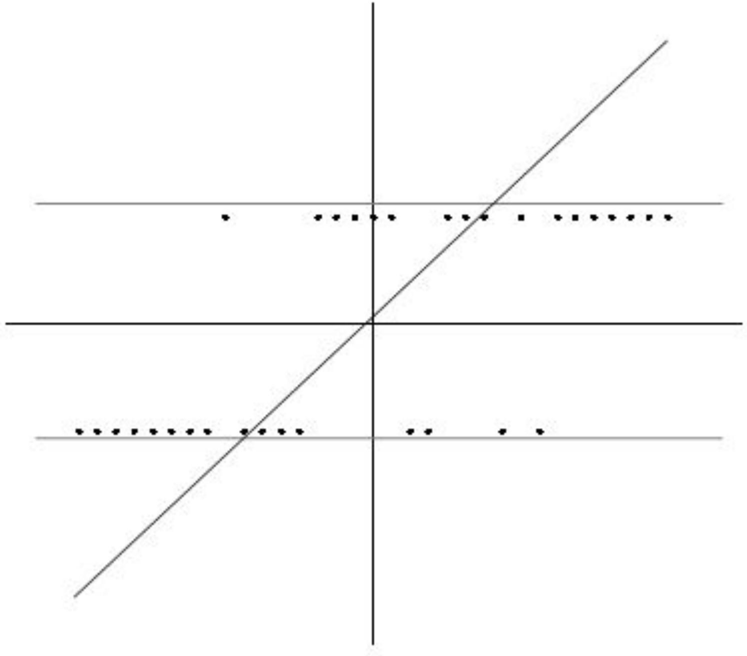
The logistic distribution constrains the estimated probabilities to lie between 0 and 1

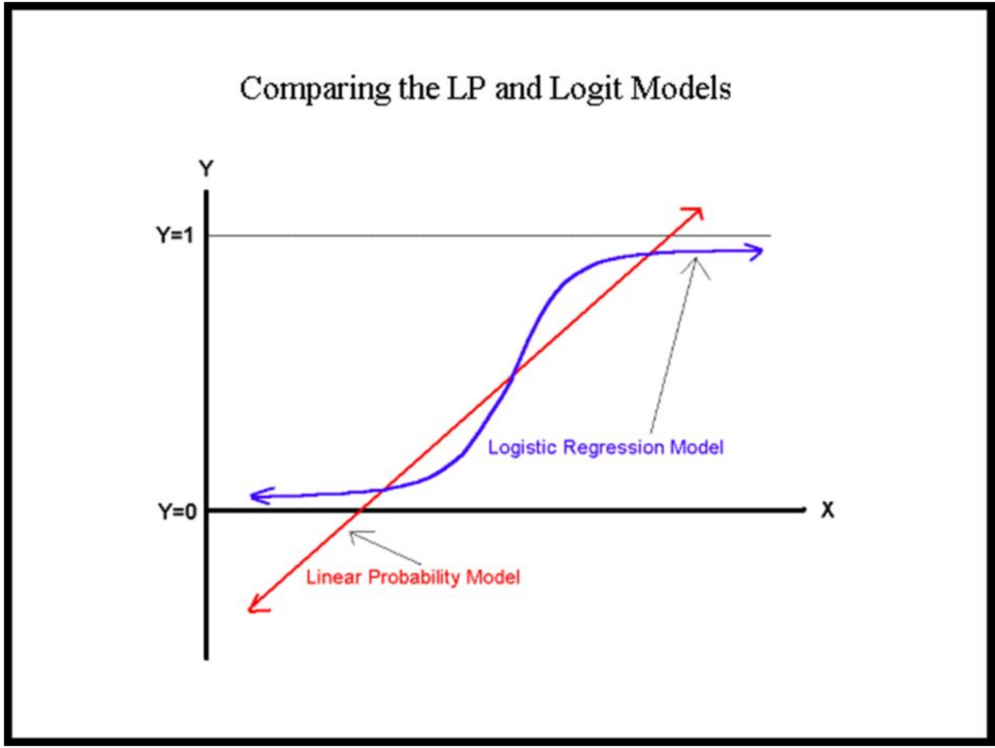
The estimated probability is:

$$p = 1/[1 + \exp(-b0 - b1\ X)]$$

if you let $b0+ b1\ X =0$, then $p = .50$
as $b0 + b1\ X$ gets really big, p approaches 1
as $b0 + b1\ X$ gets really small, p approaches 0

Linear vs Logistic regression





Coeffs interpretation

Since:

$\ln[p/(1-p)] = b0 + b1X + e$

- The slope coefficient () is interpreted as the rate of change in the "log odds" as X changes .

Multiple Logistic Regression Analysis

$$\hat{p} = \frac{\exp(b_0 + b_1X_1 + b_2X_2 + \ldots + b_pX_p)}{1 + \exp(b_0 + b_1X_1 + b_2X_2 + \ldots + b_pX_p)}$$

Outcome: **Preeclampsia**	**Regression Coefficient**	**Chi-square**	**P-value**	**Odds Ratio (95% CI)**
Intercept	-3.066	4.518	0.0335	-
Black race	2.191	12.640	0.0004	8.948 (2.673, 29.949)
Hispanic race	-0.1053	0.0325	0.8570	0.900 (0.286, 2.829)
Other race	0.0586	0.0021	0.9046	1.060 (0.104, 3.698)
Mother's age, yrs.	-0.0252	0.3574	0.5500	0.975 (0.898, 1.059)

Poisson Regression

Key Ideas

Many data take the form of a count:

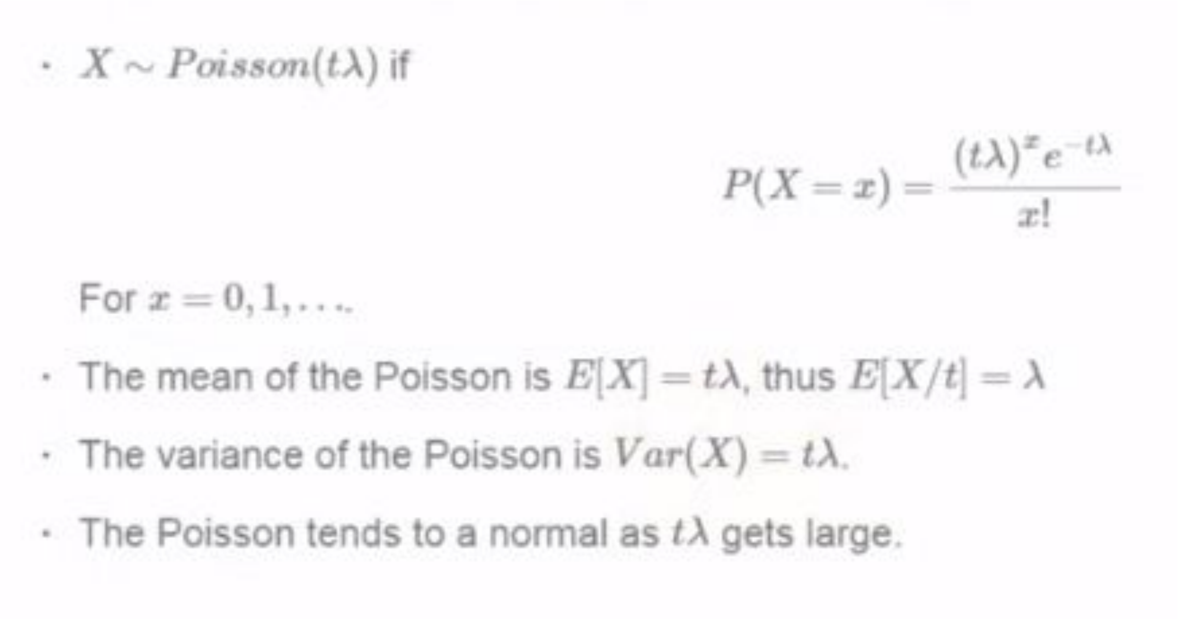
- Calls to a call center
- Number of Flu cases in an area
- Number of cars that cross a bridge

Data may also be in the form of rates:

- Percent of students passing a test
- Percent of hits to a website for a country

Linear regression with transformation is an option.

Poisson distribution



- $X \sim Poisson(t\lambda)$ if

$$P(X = x) = \frac{(t\lambda)^x e^{-t\lambda}}{x!}$$

  For $x = 0, 1, \ldots$.

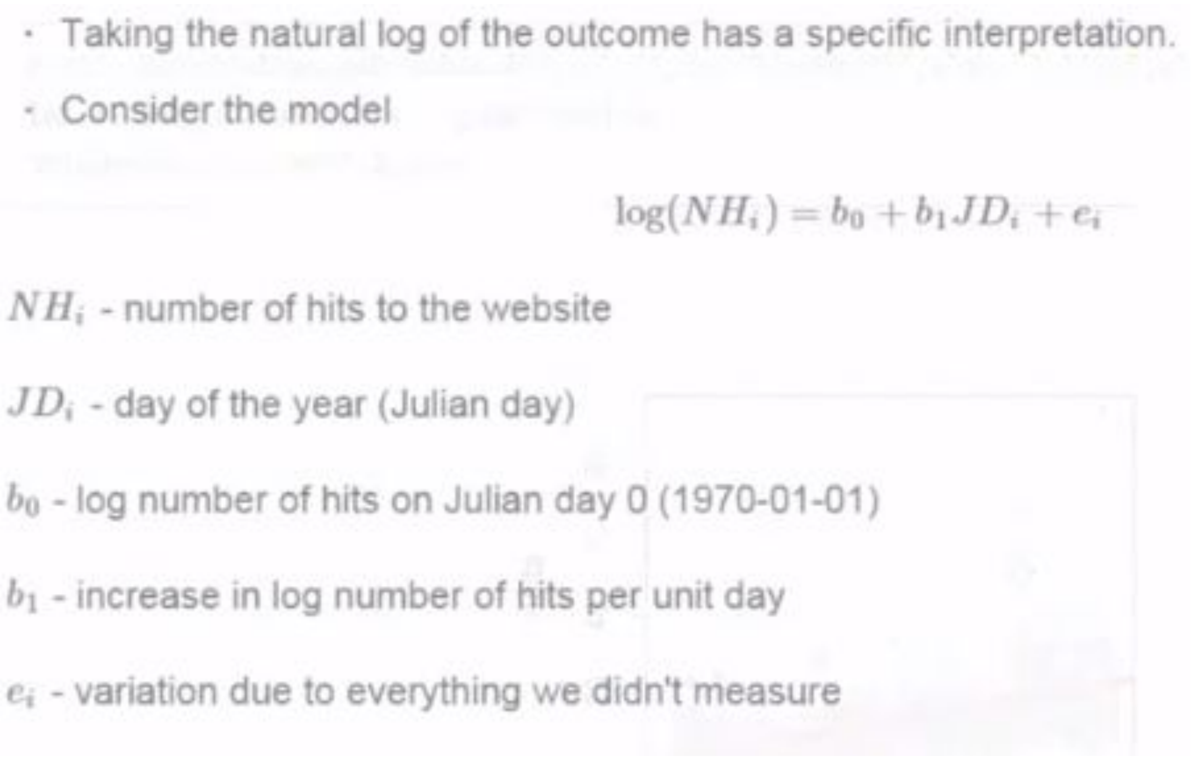
- The mean of the Poisson is $E[X] = t\lambda$, thus $E[X/t] = \lambda$
- The variance of the Poisson is $Var(X) = t\lambda$.
- The Poisson tends to a normal as $t\lambda$ gets large.

When to use

It’s best used for rare events, as these tend to follow a Poisson distribution (as opposed to more common events which tend to be normally distributed). For example:

- Number of colds contracted on airplanes.
- Number of bacteria found in a petri dish.
- Counts of catastrophic computer failures at a large tech firm in a calendar year.
- Number of 911 calls that end in the death of a suspect.

Model



- Taking the natural log of the outcome has a specific interpretation.
- Consider the model

$$\log(NH_i) = b_0 + b_1 JD_i + e_i$$

NH_i - number of hits to the website

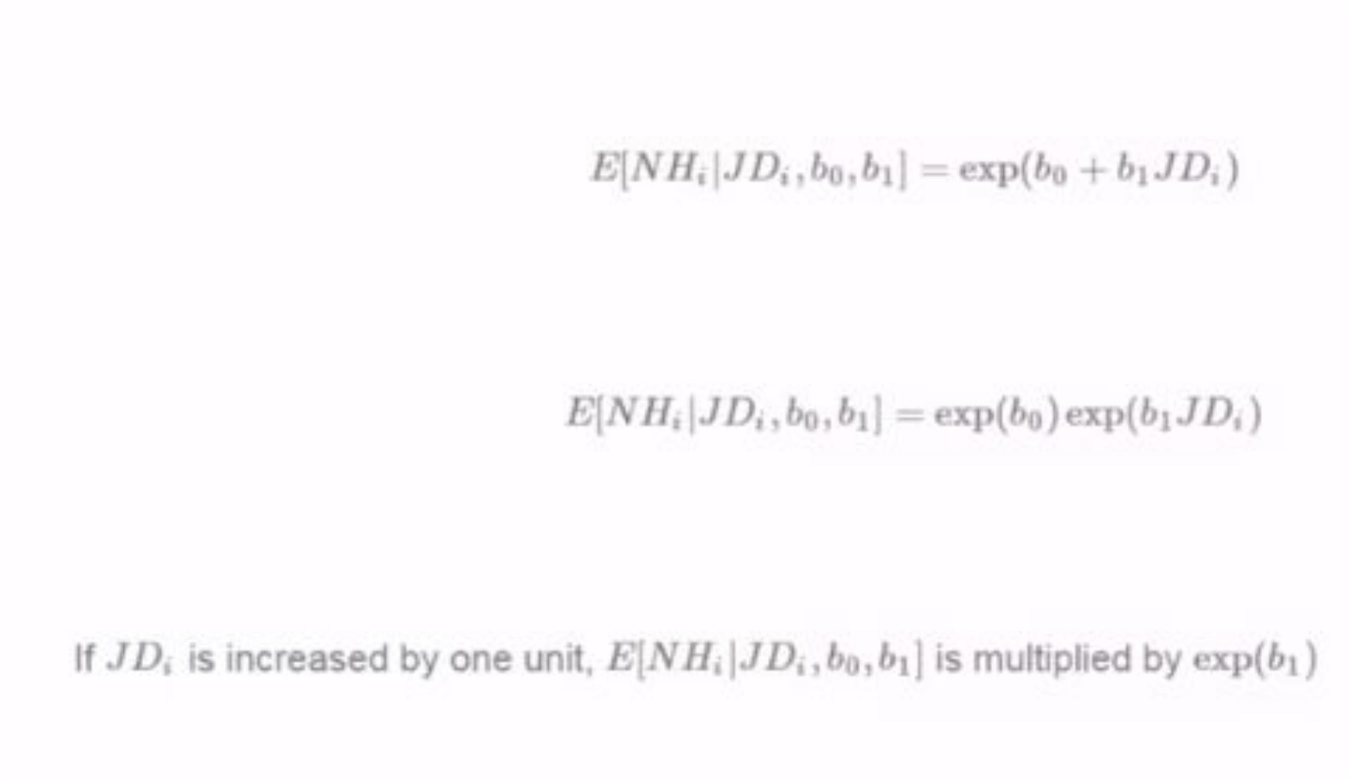
JD_i - day of the year (Julian day)

b_0 - log number of hits on Julian day 0 (1970-01-01)

b_1 - increase in log number of hits per unit day

e_i - variation due to everything we didn't measure

Multiplicative difference



$$E[NH_i|JD_i, b_0, b_1] = \exp(b_0 + b_1 JD_i)$$

$$E[NH_i|JD_i, b_0, b_1] = \exp(b_0)\exp(b_1 JD_i)$$

If JD_i is increased by one unit, $E[NH_i|JD_i, b_0, b_1]$ is multiplied by $\exp(b_1)$

Thank you for your attention!