Please open Canvas and take the test

If you get the following question

> Using trees.csv, print the difference between the median and the first quartile. Enter your answer in the box (accurate to first decimal place)

it refers to the “Girth” variable (this info is missing in the question)

Test opens at 5:30 pm and due by 6:00 pm

If you finish early, you may leave the class or stay. We won’t start lecture until 6:00 pm today

Session 1

49-781 Data Analytics for Product Managers

Spring 2018

Linear Regression

Linear Regression

Simple Linear Regression models describe the effect that a particular variable might have on the value of a continuous outcome variable.

Explanatory/Independent Variable: the variable which affects the other

Response/Outcome/Dependent Variable: the outcome variable

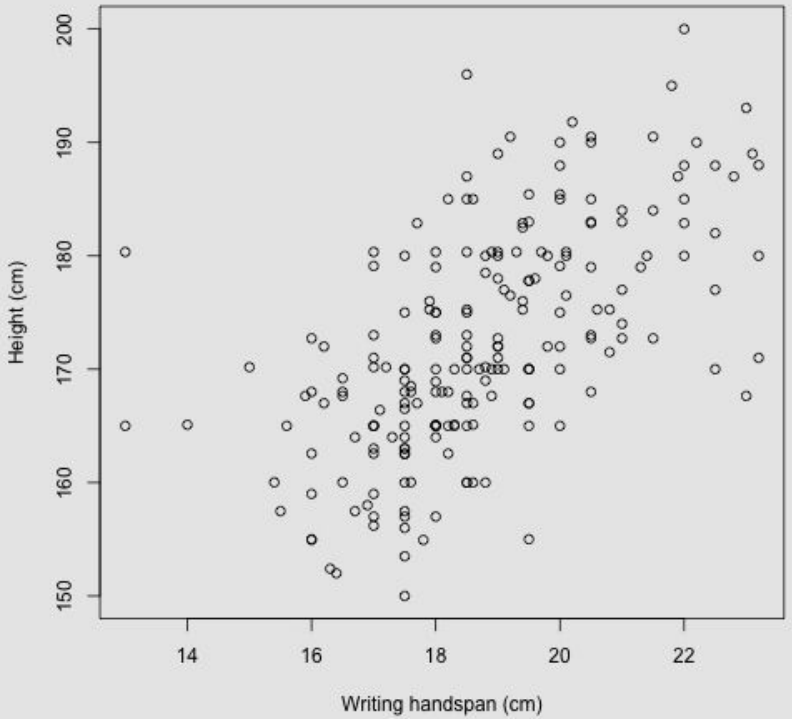
Variables can be continuous, discrete or categorical.

Height vs Writing handspan Data

A scatterplot of height against writing handspan for a sample of first-year statistics students

There seems to be a positive association between the two

Correlation Coefficient: 0.6009909



Linear Regression

The purpose **linear regression** model is to come up with a function that estimates the mean of one variable given the value of another variables.

What is the expected height of a student given their handspan?

The answer may not predict the exact height of a student, but gives the mean of a number of students whose handspan equals the given value.

Handspan is the **Explanatory Variable**

Height is the **Response Variable**

Simple Linear Regression Model

$$Y|X = \beta_0 + \beta_1 X + \epsilon$$

Y | X reads as “the value of Y conditional upon the value of X.”

- ϵ represents standard error indicates the response value is linear but subject to some random residual variation
- β_0 is the intercept - the expected value of response when the predictor is 0
- β_1 is the slope - the change in mean response for each each one-unit increase in the predictor

When the slope is positive, the regression line increases from right to left, when it's negative, it decreases from left to right; When the slope is 0, the predictor has no effect on the value of the response; β_0 and β_1 are also called regression parameters.

Estimating Slope and Intercept

Our goal is to estimate the regression parameters for a given data.

$\hat{y}$ is mean of the response

$$\hat{y} = \hat{\beta}_0 + \hat{\beta}_1 x$$

You can calculate the regression parameters as

$$\hat{\beta}_1 = \rho_{xy} \frac{s_y}{s_x} \quad \text{and} \quad \hat{\beta}_0 = \bar{y} - \hat{\beta}_1 \bar{x}$$

$\bar{x}$ and $\bar{y}$ are sample means of X and Y

s_x and s_y are the sample standard deviations of X and Y

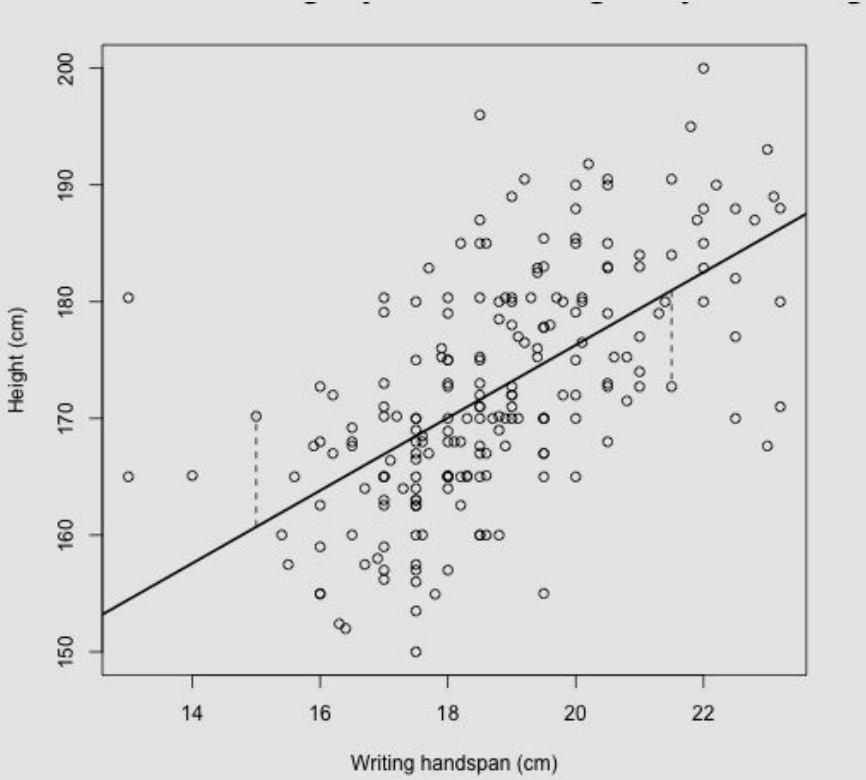
ρ_{xy} is the estimate of correlation between X and Y based on data

This method is called least squares regression

Least Squares Regression

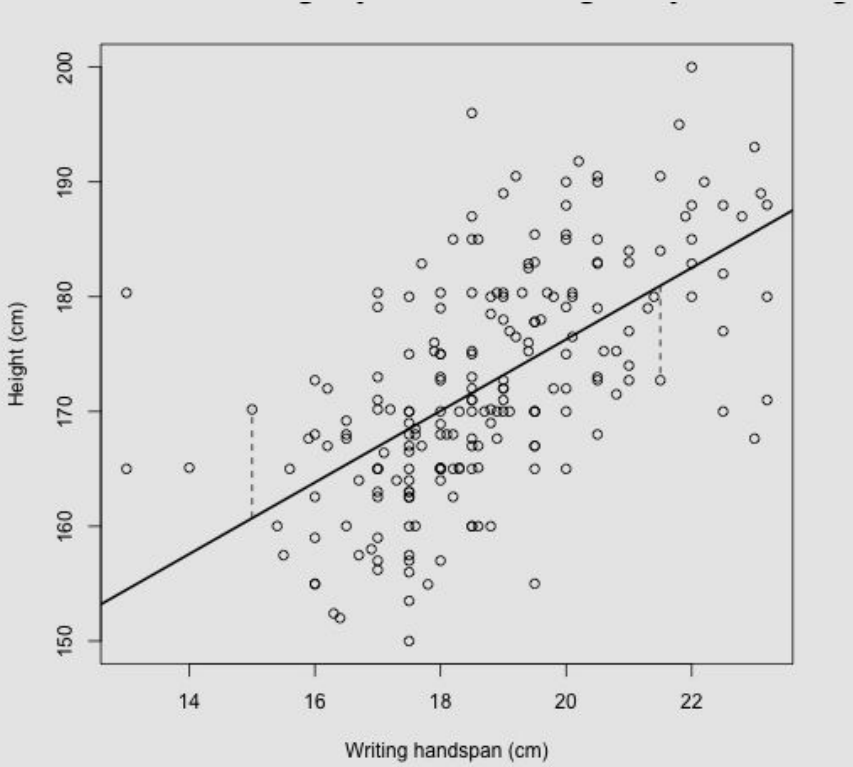
Minimizes the average squared distance between the observed data and itself

Residual is the distance between the observation and the fitted line



Fitting a Regression Line

$\hat{y} = 113.9536235 + 3.1166166\,x$



Prediction

Now that you have a model, you can use it to predict the outcome variable.

$\hat{y} = 113.9536235 + 3.1166166x$

For handspan of 14.5, the you can expect the mean height to be 159.1445638

For handspan of 24, you can expect the mean height to be 188.7524212

Confidence & Prediction Intervals

Confidence Interval for Mean Response

Account for model uncertainty with CI for a Mean Response. Let's take two values as an example.

```
##   Wr.Hnd
## 1   14.5
## 2   24.0
```

Let's calculate a 95% Confidence Interval

```
##        fit      lwr      upr
## 1 159.1446 156.4956 161.7936
## 2 188.7524 185.5726 191.9323
```

We have 95 percent confidence that *the mean height of a student* with a handspan of 14.5 cm lies somewhere between 156.5 cm and 161.8 cm

Confidence Interval for Mean Response

For 95%

```
##        fit      lwr      upr
## 1 159.1446 156.4956 161.7936
## 2 188.7524 185.5726 191.9323
```

For 99%,

```
##        fit      lwr      upr
## 1 159.1446 155.6513 162.6379
## 2 188.7524 184.5591 192.9457
```

Prediction Interval

```
##        fit      lwr      upr
## 1 159.1446 143.3286 174.9605
## 2 188.7524 172.8390 204.6659
```

A prediction interval (PI) for an observed response is different from the confidence interval in terms of context.

CIs are used to describe the variability of the mean response, a PI is used to provide the *possible range of values that an individual realization* of the response variable might take, given x.

Prediction Interval

This distinction is subtle but important: the CI corresponds to a mean, and the PI corresponds to an individual observation.

Contrast with CI below

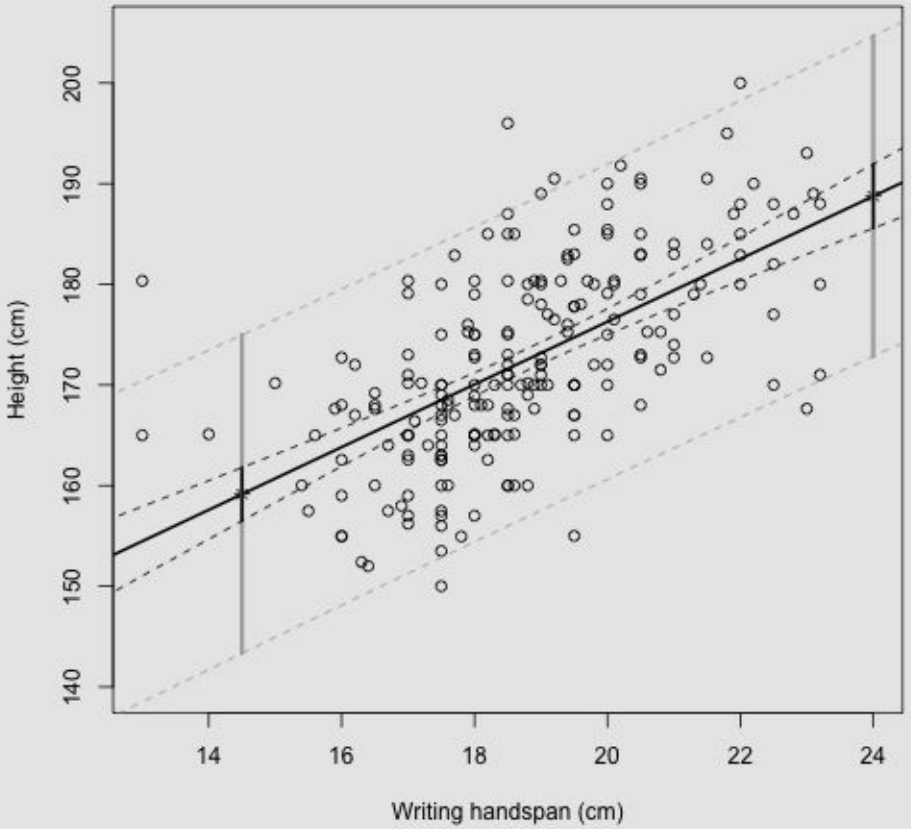
PI

```
##        fit      lwr      upr
## 1 159.1446 143.3286 174.9605
## 2 188.7524 172.8390 204.6659
```

CI

```
##        fit      lwr      upr
## 1 159.1446 156.4956 161.7936
## 2 188.7524 185.5726 191.9323
```

Intervals on a Plot

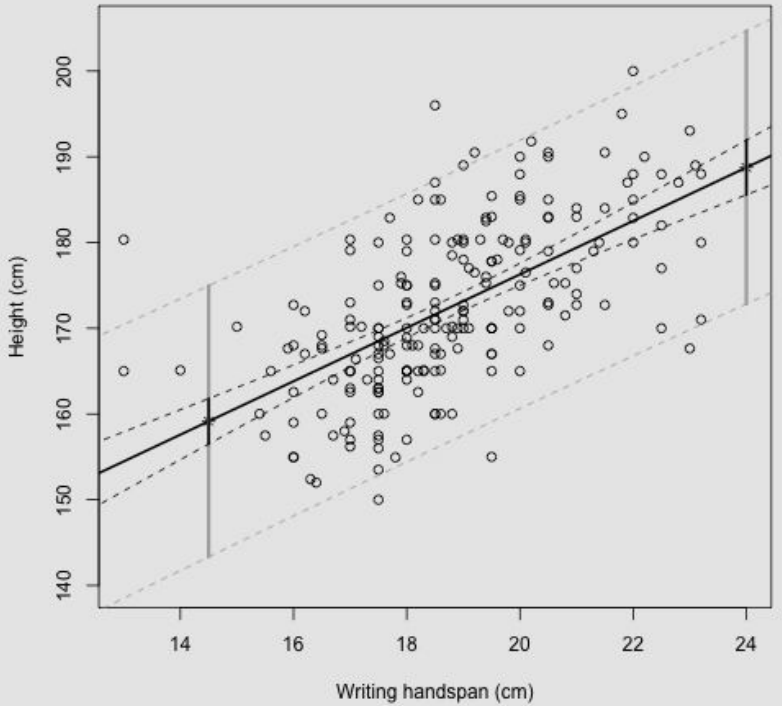


Interpolation vs Extrapolation

A prediction is an interpolation if the x value you specify falls within the range of your observed data

A prediction is an extrapolation if the x value lies outside the range of your observed data.

Does extrapolation always make sense? (x=24 is already an extrapolation. What about 0, or 50?)



Categorical Predictors

Categorical Predictors

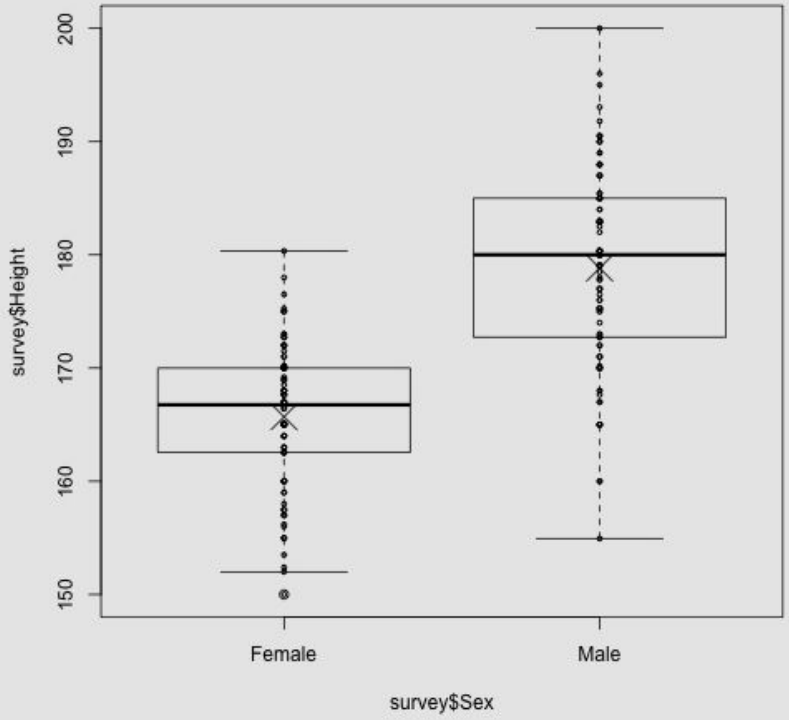
For a categorical variable with 2 values (true/false, male/female), you can write as

$\hat{y} = \hat{\beta}_0 + \hat{\beta}_1 x$

with $x = 0$ for for one value and 1 for the second value

Thus you define the coefficients such that $\hat{y} = \hat{\beta}_0$ for $x=0$ and $\hat{y} = \hat{\beta}_0 + \hat{\beta}_1$ for $x = 1$

Example



Linear Fit

```
## (Intercept)     SexMale
##   165.68667    13.13937
```

Let's predict

```
## [1] Female Male   Male   Male   Female
## Levels: Female Male
```

```
##        fit      lwr      upr
## 1 165.6867 164.2475 167.1258
## 2 178.8260 177.4143 180.2378
## 3 178.8260 177.4143 180.2378
## 4 178.8260 177.4143 180.2378
## 5 165.6867 164.2475 167.1258
```

Multilevel Categorical Variables

In general, categories cannot be related to each other in the same numeric sense as continuous variables.

$X=1,2,3,4,5,\ldots,k$

becomes

$X_{(1)}=0,1;X_{(2)}=0,1;X_{(3)}=0,1;X_{(4)}=0,1;X_{(5)}=0,1;\ldots;X_{(k)}=0,1;$

$\hat{y} = \beta_0\hat{} + \beta_1\hat{}X_{(2)} + \beta_2\hat{}X_{(3)} + \ldots + \beta_{k-1}\hat{}X_{(k)}$

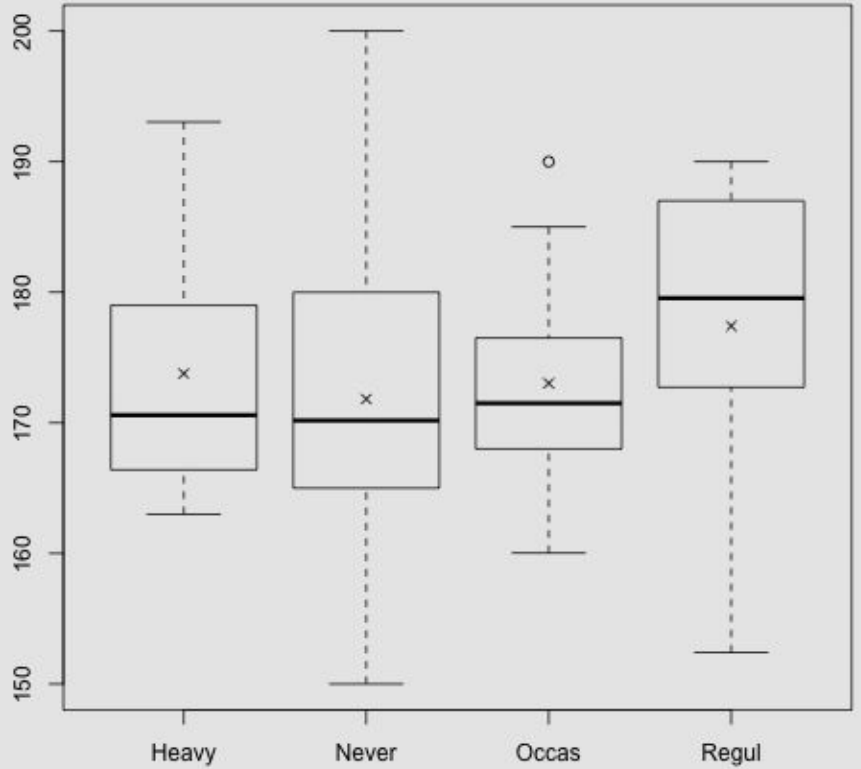
We can choose any value as the reference level

Example - Smoke Variable

Count

Heavy Never Occas Regul

11 189 19 17



Linear Model

Reference level is Heavy.

$$\hat{y} = 173.772 + (-1.952)X_{never} + (-0.74325)X_{occas.} + (3.6451429)X_{regul}$$

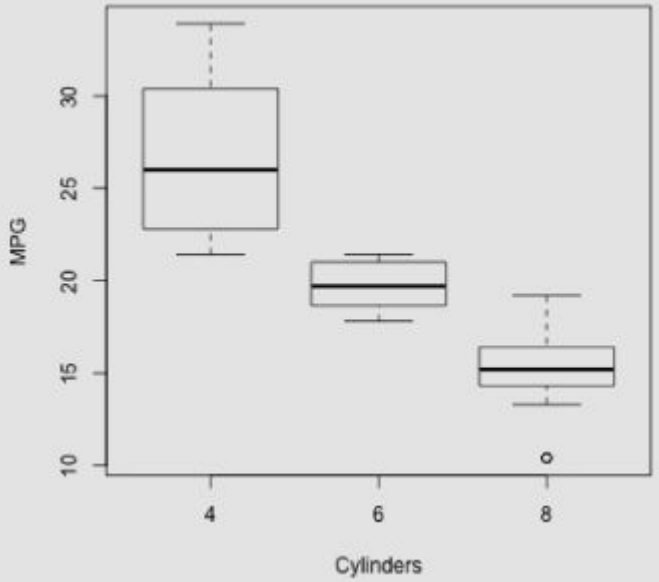
Treating Categorical Variables as Numeric

Motor Trend - MPG & Number of Cylinders

```
##  [1] 21.0 21.0 22.8 21.4 18.7 18.1 14.3 24.4 22.8 19.2 17.8 16.4 17.3 15.2
## [15] 10.4 10.4 14.7 32.4 30.4 33.9 21.5 15.5 15.2 13.3 19.2 27.3 26.0 30.4
## [29] 15.8 19.7 15.0 21.4
##  [1] 6 6 4 6 8 6 8 4 4 6 6 8 8 8 8 8 8 4 4 4 4 8 8 8 8 4 4 4 8 6 8 4
```

Though intended as categorical, treating cyl as numeric discrete variable makes sense.

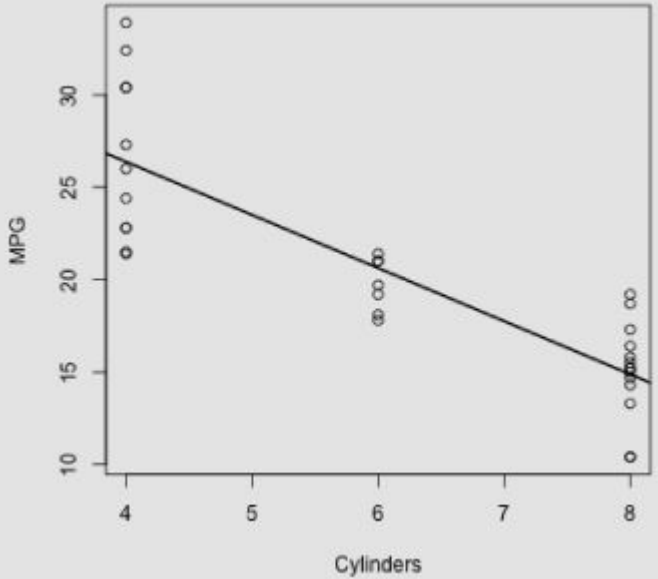
Box Plot of MPG vs Cyl



Plotting the Linear Fit

```
Mpg = 37.8846 - 2.8758 * cyl
```

You can do interesting things like predicting the MPG of a 5-cylinder car.. or even "5.5" cylinders??



Multiple Linear Regression

Multiple Linear Regression

Simple linear regression lets you control for one source of influence.

Multiple linear regression lets you control for multiple sources of influence. It has multiple explanatory variables. You are attempting to model the joint effect of several variables.

Formula

Given p independent variables $X_1, X_2, \ldots, X_p$, the model is

$$Y = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \ldots + \beta_p X_p + \epsilon$$

where $\beta_0, \beta_1, \ldots, \beta_p$ are the regression coefficients and ϵ is a normally distributed residuals around the mean.

Goal

In simple linear regression, goal is find the line of best fit with least-squares approximation. The “line” of best fit with multiple variables follows the same pattern but in a multi-dimensional plane or surface. It minimizes the overall squared distance between itself and the raw response data.

If $x_{j,i}$ is the observed value for observation i for explanatory variable X_j and y_i is the response value, we try to minimize

$$\sum_{i=1}^{n} \{y_i - (\beta_0 + \beta_1 x_{1,i} + \beta_2 x_{2,i} + \ldots + \hat{\beta}_p x_{p,i})\}^2$$

Matrix Representation

$$\boldsymbol{Y} = \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{bmatrix} \quad \boldsymbol{\epsilon} = \begin{bmatrix} \epsilon_1 \\ \epsilon_2 \\ \vdots \\ \epsilon_n \end{bmatrix} \quad \boldsymbol{\beta} = \begin{bmatrix} \beta_0 \\ \beta_1 \\ \vdots \\ \beta_p \end{bmatrix} \quad \boldsymbol{X} = \begin{bmatrix} 1 & x_{1,1} & \dots & x_{p,1} \\ 1 & x_{1,2} & \dots & x_{p,2} \\ \vdots & \vdots & \ddots & \vdots \\ 1 & x_{1,n} & \dots & x_{p,n} \end{bmatrix}$$

We can find the coefficients with this formula

$$\hat{\boldsymbol{\beta}} = \begin{bmatrix} \hat{\beta}_0 \\ \hat{\beta}_1 \\ \vdots \\ \hat{\beta}_p \end{bmatrix} = (\boldsymbol{X}^\top \cdot \boldsymbol{X})^{-1} \cdot \boldsymbol{X}^\top \cdot \boldsymbol{Y}$$

Example

y <dbl>	x1 <dbl>	x2 <dbl>
1.55	1.13	1
0.42	-0.73	0
1.29	0.12	1
0.73	0.52	1
0.76	-0.54	0
-1.09	-1.15	1
1.41	0.20	0
-0.32	-1.09	1

```
[ 1.    1.13  1.  ]
[ 1.   -0.73  0.  ]
[ 1.    0.12  1.  ]
[ 1.    0.52  1.  ]
[ 1.   -0.54  0.  ]
[ 1.   -1.15  1.  ]
[ 1.    0.2   0.  ]
[ 1.   -1.09  1.  ]
```

```
[ 1.22545715]
[ 1.01530042]
[-0.69801891]
```

You have solved

β_0=1.2254572,
β_1=1.0153004 and
β_2=−0.6980189

Important to note that each numeric-continuous variable has a slope coefficient that provides a per unit change in the outcome variable. Any k-group categorical variables provide $k-1$ intercepts.

Fit the Model

We now have the formula

$$\hat{y} = 1.2254572 + (1.0153004)\, x_1 + (-0.6980189)\, x_2$$

```
[1.67472772] [1.55]
[0.48428784] [0.42]
[0.64927429] [1.29]
[1.05539446] [0.73]
[0.67719492] [0.76]
[-0.64015725] [-1.09]
[1.42851723] [1.41]
[-0.57923922] [-0.32]
```

Clock Prices at Auction

```
Age,Bidders,Price
127,13,1235
115,12,1080
127,7,845
150,9,1522
156,6,1047
182,11,1979
156,12,1822
132,10,1253
137,9,1297
113,9,946
137,15,1713
117,11,1024
137,8,1147
153,6,1092
117,13,1152
126,10,1336
170,14,2131
```

Linear Regression Model

Dependent: Price ~ Independent: Age

	coef	std err	t	P>|t|	[0.025	0.975]
Intercept	-191.6576	263.887	-0.726	0.473	-730.586	347.271
Age	10.4791	1.790	5.854	0.000	6.823	14.135

$\hat{p} = -191 + 10.48\,a$

Linear Regression Model

Dependent: Price ~ Independent: Bidders

	coef	std err	t	P>|t|	[0.025	0.975]
Intercept	806.4049	230.685	3.496	0.001	335.284	1277.526
Bidders	54.6362	23.225	2.352	0.025	7.204	102.068

$\hat{p} = 806 + 54.64\, b$

Linear Regression Model

Dependent: Price ~ Independent: Bidders+Age

	coef	std err	t	P>|t|	[0.025	0.975]
Intercept	-1336.7221	173.356	-7.711	0.000	-1691.275	-982.169
Bidders	85.8151	8.706	9.857	0.000	68.010	103.620
Age	12.7362	0.902	14.114	0.000	10.891	14.582

$\hat{p} = -191 + 10.48\, a$

$\hat{p} = 806 + 54.64\, b$

$\hat{p} = -1336.72 + 85.81\, b + 12.74\, a$

Example: Student Survey

Student Survey

	Sex	Wr.Hnd	NW.Hnd	W.Hnd	Fold	Pulse	Clap	Exer	Smoke	Height	M.I	Age
1	Female	18.5	18.0	Right	R on L	92	Left	Some	Never	173.00	Metric	18.250
2	Male	19.5	20.5	Left	R on L	104	Left	None	Regul	177.80	Imperial	17.583
3	Male	18.0	13.3	Right	L on R	87	Neither	None	Occas	NA	<NA>	16.917
4	Male	18.8	18.9	Right	R on L	NA	Neither	None	Never	160.00	Metric	20.333
5	Male	20.0	20.0	Right	Neither	35	Right	Some	Never	165.00	Metric	23.667
6	Female	18.0	17.7	Right	L on R	64	Right	Some	Never	172.72	Imperial	21.000

Linear Regression Model

Dependent: Height ~ Independent: W.Hnd

	coef	std err	t	P>|t|	[0.025	0.975]
Intercept	113.9536	5.442	20.941	0.000	103.225	124.682
WrHnd	3.1166	0.289	10.792	0.000	2.547	3.686

$\hat{h} = 114 + 3.12\, w$

Linear Regression Model

Dependent: Height ~ Independent: Sex

	coef	std err	t	P>|t|	[0.025	0.975]
Intercept	165.6867	0.730	226.978	0.000	164.248	167.126
Sex[T.Male]	13.1394	1.023	12.850	0.000	11.123	15.155

$\hat{h} = 165.69 + 13.14\, s$ (1 if Male, 0 if Female)

Being male added about 13.14cm in height when compared to the mean of height for females (which is 166cm).

Linear Regression Model

Dependent: Height ~ Independent: W.Hand + Sex

	coef	std err	t	P>|t|	[0.025	0.975]
Intercept	137.6870	5.713	24.100	0.000	126.423	148.951
Sex[T.Male]	9.4898	1.229	7.724	0.000	7.067	11.912
WrHnd	1.5944	0.323	4.937	0.000	0.958	2.231

$\hat{h} = 114 + 3.12\, w$

$\hat{h} = 165.69 + 13.14\, s$

$\hat{h} = 137.69 + 1.59\, w + 9.49\, s$ (1 if Male, 0 if Female)

$\hat{h} = 114 + 3.12\, w$

$\hat{h} = 165.69 + 13.14\, s$

$\hat{h} = 137.69 + 9.49\, w + 1.59\, s$ (1 if Male, 0 if Female)

The effect of hand span is now 1.59 (about half 3.12) but it's still highly statistically significant even in the presence of sex.

The coefficient of sex has reduced as well, but that is also still significant.

Model Parameters

Linear Regression Model

Dep. Variable:	Height	R-squared:	0.506
Model:	OLS	Adj. R-squared:	0.501
Method:	Least Squares	F-statistic:	104.6
Date:	Thu, 29 Mar 2018	Prob (F-statistic):	5.47e-32
Time:	15:41:56	Log-Likelihood:	-694.63
No. Observations:	207	AIC:	1395.
Df Residuals:	204	BIC:	1405.
Df Model:	2		
Covariance Type:	nonrobust		

	coef	std err	t	P>|t|	[0.025	0.975]
Intercept	137.6870	5.713	24.100	0.000	126.423	148.951
Sex[T.Male]	9.4898	1.229	7.724	0.000	7.067	11.912
WrHnd	1.5944	0.323	4.937	0.000	0.958	2.231

R-Squared

Residual Standard Error is 6.987 which indicates the standard deviation of the residuals (Root mean square of the residuals)

R-squared is a statistical measure of how close the data are to the fitted regression line. It is also known as the coefficient of multiple determination for multiple regression.

R-squared = Explained variation / Total variation

Let's now add one more explanatory variable. Previously, smoking frequency had no statistically significant impact on height prediction. How about now?

Since it's a categoric variable with four values, it's dummy coded with one reference level and three variables.

Two Categorical Variables

Dependent: Height ~ Independent: W.Hand + Sex + Smoke

	coef	std err	t	P>|t|	[0.025	0.975]
Intercept	137.4056	6.544	20.996	0.000	124.501	150.310
Sex[T.Male]	9.3979	1.245	7.547	0.000	6.943	11.853
Smoke[T.Never]	-0.0442	2.314	-0.019	0.985	-4.606	4.518
Smoke[T.Occas]	1.5267	2.869	0.532	0.595	-4.131	7.185
Smoke[T.Regul]	0.9211	2.929	0.314	0.753	-4.854	6.697
WrHnd	1.6042	0.330	4.860	0.000	0.953	2.255

Writing handspan and sex give low p-values smoking frequency suggests no evidence against the hypothesis of zero coefficients.

The coefficients for handspan and sex have not been materially impacted by adding another explanatory variable.

We'll look later on at how we can use this summary to help decide whether to keep a new explanatory variable in a model or not.

Lab

Linear Regression - Imports Needed

```
import matplotlib.pyplot as plt
import numpy as np
import pandas as pd
import statsmodels.formula.api as sm
import statsmodels.tools.eval_measures as em
```

Linear Fit - Single Variable

```
y = np.array([[1.55],[0.42],[1.29],[0.73],[0.76],[-1.09],[1.41],[-0.32]])
x1 = np.array([[1.13],[-0.73],[0.12],[0.52],[-0.54],[-1.15],[0.20],[-1.09]])
```

$$\boldsymbol{X} = \begin{bmatrix} 1 & x_{1,1} & \dots & x_{p,1} \\ 1 & x_{1,2} & \dots & x_{p,2} \\ \vdots & \vdots & \ddots & \vdots \\ 1 & x_{1,n} & \dots & x_{p,n} \end{bmatrix} \qquad \hat{\boldsymbol{\beta}} = \begin{bmatrix} \hat{\beta}_0 \\ \hat{\beta}_1 \\ \vdots \\ \hat{\beta}_p \end{bmatrix} = (\boldsymbol{X}^\top \cdot \boldsymbol{X})^{-1} \cdot \boldsymbol{X}^\top \cdot \boldsymbol{Y}$$

```
# Let's use the formula first
id = np.ones((8,1))
x = np.hstack((id,x1))
beta=(np.dot(np.dot(np.linalg.inv(np.dot(x.transpose(),x)),x.transpose()),y))
print(beta)
yp1 = beta[0]+beta[1]*x1
print(np.hstack((x1,y,yp1)))
```

Manual Calculation - Compare Prediction

X1	Y	Y_{pred}
[1.13	1.55	1.83850911]
[-0.73	0.42	0.08784601]
[0.12	1.29	0.88788022]
[0.52	0.73	1.26436691]
[-0.54	0.76	0.26667719]
[-1.15	-1.09	-0.30746501]
[0.2	1.41	0.96317756]
[-1.09	-0.32	-0.250992]

Statsmodel Linear Regression

```
# Use statsmodels Linear Regression

d = pd.DataFrame(np.hstack((x1,y)))
d.columns = ["x1","y"]
print("dataframe", d)
res = sm.ols(formula="y ~ x1",data=d).fit()
print("Summary", res.summary())
yp2 = res.predict(x1).values.reshape((8,1))
print(np.hstack((x1,y,yp2)))

# Predict for new X
x1new =
pd.DataFrame(np.hstack(np.array([[1],[0],[-0.12],[0.
52]])))
x1new.columns=["x1"]
yp2new = res.predict(x1new).values.reshape((4,1))
print(np.hstack((x1new,yp2new)))
```

```
   X          y       y (manual)   y (model)
[ 1.13        1.55        1.83850911  1.83850911]
[-0.73        0.42        0.08784601  0.08784601]
[ 0.12        1.29        0.88788022  0.88788022]
[ 0.52        0.73        1.26436691  1.26436691]
[-0.54        0.76        0.26667719  0.26667719]
[-1.15       -1.09       -0.30746501 -0.30746501]
[ 0.2         1.41        0.96317756  0.96317756]
[-1.09       -0.32       -0.250992   -0.250992  ]
```

```
[ 1.          1.71615094]
[ 0.          0.77493422]
[-0.12        0.66198821]
[ 0.52        1.26436691]
```

Plotting Example & RMS Deviation

```
plt.scatter(x1,y)
plt.plot(x1,yp2, color="blue")
plt.show()
```

Multiple Regression - Manual

```
#Data
y = np.array([[1.55],[0.42],[1.29],[0.73],[0.76],[-1.09],[1.41],[-0.32]])
x1 = np.array([[1.13],[-0.73],[0.12],[0.52],[-0.54],[-1.15],[0.20],[-1.09]])
x2 = np.array([[1],[0],[1],[1],[0],[1],[0],[1]])

#Manual
id = np.ones((8,1))
x = np.hstack((id,x1,x2))
print(x)
beta=(np.dot(np.dot(np.linalg.inv(np.dot(x.transpose(),x)),x.transpose()),y))
print(beta)
yp1 = beta[0]+beta[1]*x1+beta[2]*x2
print(np.hstack((x,y,yp1)))
```

Multiple Regression - Statsmodels

```
# Statsmodels
d = pd.DataFrame(np.hstack((x1,x2,y)))
d.columns = ["x1","x2","y"]
print("dataframe", d)
res = sm.ols(formula="y ~ x1+x2",data=d).fit()

yp2 = res.predict(np.hstack((x1,x2))).values.reshape((8,1))
print(np.hstack((x1,x2,y,yp1,yp2)))

# Predict for new X
x1new =
pd.DataFrame(np.hstack((np.array([[1],[0],[-0.12],[0.52]]),
np.array([[1],[-1],[2],[0.77]]))))
x1new.columns=["x1","x2"]
yp2new = res.predict(x1new).values.reshape((4,1))
print(np.hstack((x1new,yp2new)))
```

```
dataframe      X1    x2     y  y(manual)  y(model)
0  1.13   1.0  1.55   1.674728  1.674728
1 -0.73   0.0  0.42   0.484288  0.484288
2  0.12   1.0  1.29   0.649274  0.649274
3  0.52   1.0  0.73   1.055394  1.055394
4 -0.54   0.0  0.76   0.677195  0.677195
5 -1.15   1.0 -1.09  -0.640157 -0.640157
6  0.20   0.0  1.41   1.428517  1.428517
7 -1.09   1.0 -0.32  -0.579239 -0.579239
[[ 1.5   1.    1.54]
 [ 0.   -1.    1.92]
 [-0.12  2.   -0.29]
 [ 0.52  0.77  1.22]]
```

Reading from a File (Survey)

```
d=pd.read_csv("survey.csv")
d=d.rename(index=str,columns={"Wr.Hnd":"WrHnd"})
print("dataframe", d)
res = sm.ols(formula="Height ~ WrHnd+Sex+Smoke",data=d).fit()
print(res.summary())
```

Reading from a File (Clocks)

```
d=pd.read_csv("clock.csv")
print("dataframe", d)
res = sm.ols(formula="Price ~ Bidders+Age",data=d).fit()
print(res.summary())
```