University of Colorado **Boulder**

CSCI 3022: Intro to Data Science

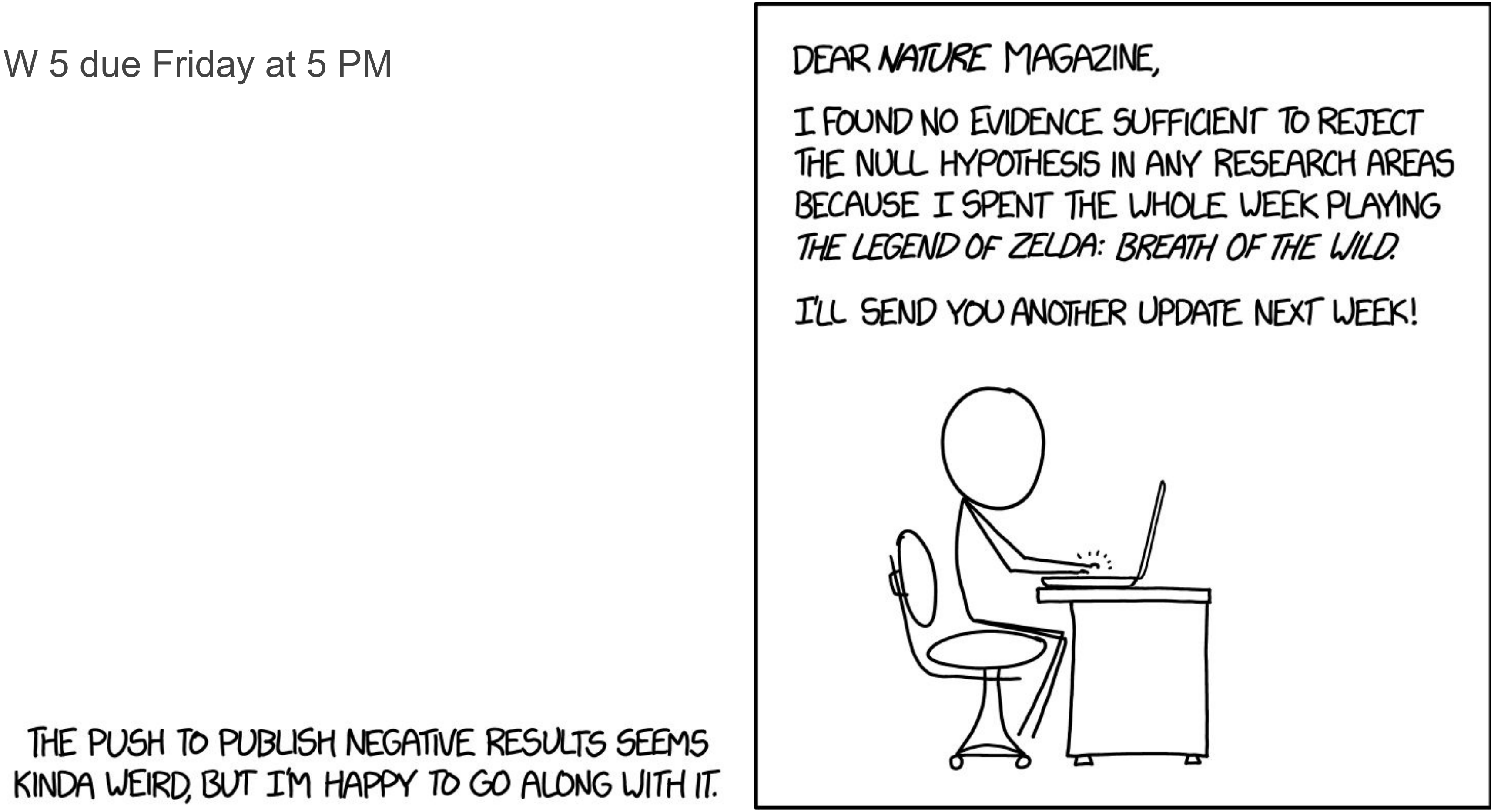
Spring 2019 Tony Wong

Lecture 23: Inference and Model Selection in Multiple Linear Regression

THE PUSH TO PUBLISH NEGATIVE RESULTS SEEMS KINDA WEIRD, BUT I'M HAPPY TO GO ALONG WITH IT.

Announcements and reminders

- HW 5 due Friday at 5 PM



Previously on CSCI 3022…

Given data (x_{i1}, x_{i2}, … , x_{ip}, y_i), for i = 1, 2, …, n, fit a MLR model of the form

$$y_i = \beta_0 + \beta_1 x_{i1} + \beta_2 x_{i2} + \ldots + \beta_p x_{ip} + \epsilon_i, \quad \text{where each } \epsilon_i \sim N(0, \sigma^2)$$

Estimate of the parameters are found by minimizing

$$SSE = \sum_{i=1}^{n} (y_i - (\hat{\beta}_0 + \hat{\beta}_1 x_1 + +\hat{\beta}_2 x_2 + \ldots + \hat{\beta}_p x_p))^2$$

The covariance and correlation coefficient for random variables X and Y are given by:

$$\mathrm{Cov}(X,Y) = E[(X - E[X])(Y - E[Y])] \quad \text{and} \quad \rho(X,Y) = \frac{\mathrm{Cov}(X,Y)}{\sqrt{\mathrm{Var}(X)\ \mathrm{Var}(Y)}}$$

Recap of advertising budget example

SLR

```
SLR for tv vs sales
----------------------
intercept = 7.0326
slope = 0.0475
p-value = 1.4673897001948012e-42


SLR for radio vs sales
----------------------
intercept = 9.3116
slope = 0.2025
p-value = 4.354966001766976e-19


SLR for news vs sales
----------------------
intercept = 12.3514
slope = 0.0547
p-value = 0.0011481958688882112
```

MLR

`sales = 2.94 + 0.046×TV + 0.189×radio - 0.001×news`

- SLR: Each advertising medium shows sig. slope
- MLR: The coefficient for newspaper ads disappears

Recap of advertising budget example

SLR

```
SLR for tv vs sales
----------------------
intercept = 7.0326
slope = 0.0475
p-value = 1.467389700194801 2e-42
```

```
SLR for radio vs sales
----------------------
intercept = 9.3116
slope = 0.2025
p-value = 4.354966001766976e-19
```

```
SLR for news vs sales
----------------------
intercept = 12.3514
slope = 0.0547
p-value = 0.0011481958688882112
```

MLR

`sales = 2.94 + 0.046×TV + 0.189×radio - 0.001×news`

- SLR: Each advertising medium shows sig. slope
- MLR: The coefficient for newspaper ads disappears
- This was because in SLR, news was a surrogate for radio, which we learned by looking at pairwise correlation coefficients:

	tv	radio	news
tv	1.000000	0.054809	0.056648
radio	0.054809	1.000000	0.354104
news	0.056648	0.354104	1.000000

Inference in MLR

Questions we would like to answer**:**

- Is at least one of the features useful in predicting the response?
- Do all of the feature help to explain the response? Or can we reduce to just a few?
- How well does the model fit the data? How well does just a subset of features do?

Is at least one feature important?

SLR : $Y = \beta_0 + \beta_1 x$

$H_0 : \beta_1 = 0$

$H_1 : \beta_1 \neq 0$

- In the SLR setting, we can do a hypothesis test to determine if $\beta_1 = 0$
- In the MLR setting with *p* features, we need to check whether ALL coefficients are 0:

$$H_0: \beta_1 = \beta_2 = \dots = \beta_p = 0$$

$$H_1: \beta_k \neq 0 \text{ for at least one value of k in } 1, 2, \dots, p$$

Is at least one feature important?

SST − SSE = how much better they are.

The F-test:

We test the hypothesis via the **F-statistic:**

$$F = \frac{(SST - SSE)/p}{SSE/(n-p-1)}$$

Fond memories:

$$SSE = \sum_{i=1}^{n}(y_i - \hat{y})^2$$

$$\underbrace{SST = \sum_{i=1}^{n}(y_i - \bar{y})^2}$$

$p = dof_{SST} - dof_{SSE}$

$= (n-1) - (n-p-1)$

$= p$

SST is the SSE we would have if we used $\hat{y} = \bar{y}$

SST is from a model w/out any feature: $\hat{y} = \beta_0 \rightarrow$ slop

So: The F-statistic is a measure of **how much better our model is than just using the mean**

Is at least one feature important?

The **F-test:** $$F = \frac{(SST - SSE)/p}{SSE/(n-p-1)}$$ $$SSE = \sum_{i=1}^{n}(y_i - \hat{y})^2 \qquad SST = \sum_{i=1}^{n}(y_i - \bar{y})^2$$

S'pose H_0 were true. What would F be?

→ $\beta_1 = \beta_2 = \ldots \beta_p = 0$ → $F \approx 1$

S'pose H_1 were true. What would F be?

→ Better explained data → Lower SSE → SST - SSE is higher → F is **higher**

Hypothesis testing:

→ If $F \geq F_{\alpha,\, p,\, n-p-1}$, then **reject H_0** and conclude at least one feature is important

→ p-value = 1 - stats.f.cdf(F, p, n-p-1)

Is at least one feature important?

The **F-test:** $$F = \frac{(SST - SSE)/p}{SSE/(n-p-1)} \qquad SSE = \sum_{i=1}^{n}(y_i - \hat{y})^2 \qquad SST = \sum_{i=1}^{n}(y_i - \bar{y})^2$$

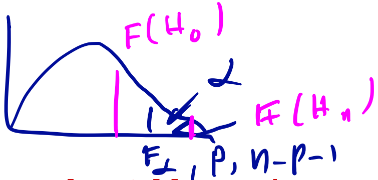
S'pose H_0 were true. What would F be?

→ $\beta_1 = \beta_2 = \ldots \beta_p = 0 \rightarrow F \approx 1$ SST ≈ SSE low value of F test statistic ⟺ support for H_0.

S'pose H_1 were true. What would F be?

→ Better explained data → Lower SSE → SST - SSE is higher → F is **higher** ⟺ support for H_1

Hypothesis testing:



→ If $F \geq F_{\alpha, p, n-p-1}$, then **reject H_0** and conclude at least one feature is important

→ p-value = 1 - stats.f.cdf(F, $\underbrace{p}_{dfn}$, $\underbrace{n-p-1}_{dfd}$)

Is subset of features important?

- **Full model:** $y = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \beta_3 x_3 + \beta_4 x_4$ (p=4 features in full model)
- **Reduced model:** $y = \beta_0 + \beta_2 x_2 + \beta_4 x_4$ (k=2 features in reduced model)

Question: Are the missing features important, or are we okay going with the reduced model?

SST use the model $\hat{y} = \bar{y} = \beta_0$

Answer: Partial F-test!

- H_0: $\beta_1 = \beta_3 = 0$

Since the features in the reduced model are also in the full model, we expect the full model to perform *at least* as well as the reduced model.

Strategy: Fit the **full** and **reduced** models. Determine if the difference in performance is **real** or just **due to chance**

Is subset of features important?

- **Full model:** $y = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \beta_3 x_3 + \beta_4 x_4$ (p=4 features in full model)
- **Reduced model:** $y = \beta_0 + \beta_2 x_2 + \beta_4 x_4$ (k=2 features in reduced model)

Question: Are the missing features important, or are we okay going with the reduced model?

Full : $H_0 : \beta_1 = \beta_2 = \ldots = \beta_p = 0$

H_1 : at least one of β_k's $\neq 0$

Answer: Partial F-test!

- H_0: $\beta_1 = \beta_3 = 0$

H_1: at least one of $\beta_0, \beta_2 \neq 0$

Since the features in the reduced model are also in the full model, we expect the full model to perform *at least* as well as the reduced model.

Strategy: Fit the **full** and **reduced** models. Determine if the difference in performance is **real** or just **due to chance**

Is subset of features important?

$$F = \frac{(SST - SSE)/p}{SSE/(n-p-1)}$$

- SSE_{full} = variation unexplained by the full model

$\hat{Y}_{red,i}$

- $SSE_{reduced}$ = variation unexplained by the reduced model $= \sum (y_i - \overbrace{(\hat{\beta}_0 + \hat{\beta}_2 x_2 + \hat{\beta}_k x_k)}^{\hat{Y}_{red,i}})^2$

Intuitively, if SSE_{full} is much smaller than $SSE_{reduced}$, the full model fits the data much better than the reduced model.

The appropriate test statistic should depend on the difference $SSE_{reduced} - SSE_{full}$ in unexplained variation.

Test statistic: $F = \dfrac{(SSE_{reduced} - SSE_{full}) / \overbrace{(df_{SSE_{redu}} - df_{SSE_{full}})}^{p-k}}{SSE_{full}/(n-p-1)}$

Rejection region:

$(n-k-1) - (n-p-1) = p-k$

Is subset of features important?

- SSE_{full} = variation unexplained by the full model
- $SSE_{reduced}$ = variation unexplained by the reduced model

Intuitively, if SSE_{full} is much smaller than $SSE_{reduced}$, the full model fits the data much better than the reduced model.

The appropriate test statistic should depend on the difference $SSE_{reduced}$ - SSE_{full} in unexplained variation.

Test statistic: $$F = \frac{(SSE_{red} - SSE_{full})/(p-k)}{SSE_{full}/(n-p-1)} \sim F_{p-k,n-p-1}$$

Rejection region: $F \geq F_{\alpha,p-k,n-p-1}$

http://homepage.divms.uiowa.edu/~mbognar/applets/f.html

Why use the F-tests?

CI for β_k: $\hat{\beta}_k \pm t_{\alpha/2}$ ST ERROR

Question: Why compute the p-value for the F-statistic when we could compute p-values for each of the feature slopes?

- **Why, Part A:** If we do this, we're testing p different hypotheses instead of a single hypothesis
- **Why, Part B:** At α = 0.05, how many p-values do we expect to be significant if the null hypothesis is in fact true?
  - If we had 100 parameters, about 5 would be significant **just by chance**
  - **Problem of Multiple Comparisons**

In [6]:
```
model.summary()
```

Out[6]:

OLS Regression Results

Dep. Variable:	sales	**R-squared:**	0.897
Model:	OLS	**Adj. R-squared:**	0.896
Method:	Least Squares	**F-statistic:**	570.3
Date:	Tue, 10 Jul 2018	**Prob (F-statistic):**	1.58e-96
Time:	18:04:05	**Log-Likelihood:**	-386.18
No. Observations:	200	**AIC:**	780.4
Df Residuals:	196	**BIC:**	793.6
Df Model:	3		
Covariance Type:	nonrobust		

pvalue

	coef	**std err**	**t**	**P>|t|**	**[0.025**	**0.975]**
const	2.9389	0.312	9.422	0.000	2.324	3.554
tv	0.0458	0.001	32.809	0.000	0.043	0.049
radio	0.1885	0.009	21.893	0.000	0.172	0.206
news	-0.0010	0.006	-0.177	0.860	-0.013	0.011

Why use the F-tests?

Question: Why compute the p-value for the F-statistic when we could compute p-values for each of the feature slopes?

- **Why, Part A:** If we do this, we're testing p different hypotheses instead of a single hypothesis
- **Why, Part B:** At α = 0.05, how many p-values do we expect to be significant if the null hypothesis is in fact true?
  - If we had 100 parameters, about 5 would be significant **just by chance**
  - **Problem of Multiple Comparisons**

```
In [6]: model.summary()
```

Out[6]:

OLS Regression Results

Dep. Variable:	sales	**R-squared:**	0.897
Model:	OLS	**Adj. R-squared:**	0.896
Method:	Least Squares	**F-statistic:**	570.3
Date:	Tue, 10 Jul 2018	**Prob (F-statistic):**	1.58e-96
Time:	18:04:05	**Log-Likelihood:**	-386.18
No. Observations:	200	**AIC:**	780.4
Df Residuals:	196	**BIC:**	793.6
Df Model:	3		
Covariance Type:	nonrobust		

	coef	**std err**	**t**	**P>|t|**	**[0.025**	**0.975]**
const	2.9389	0.312	9.422	0.000	2.324	3.554
tv	0.0458	0.001	32.809	0.000	0.043	0.049
radio	0.1885	0.009	21.893	0.000	0.172	0.206
news	-0.0010	0.006	-0.177	0.860	-0.013	0.011

Why use the F-tests?

Question: Why compute the p-value for the F-statistic when we could compute p-values for each of the feature slopes?

- **Why, Part A:** If we do this, we're testing p different hypotheses instead of a single hypothesis
- **Why, Part B:** At α = 0.05, how many p-values do we expect to be significant if the null hypothesis is in fact true?

$P(\text{at least 1 false positive}) = 1 - P(\text{no false +})$

$= 1 - (.95)^3$

$= 1 - .86 \approx .14$

  - If we had 100 parameters, about 5 would be significant **just by chance**
  - **Problem of Multiple Comparisons**

In [6]:
```
model.summary()
```

Out[6]:

OLS Regression Results

Dep. Variable:	sales	**R-squared:**	0.897
Model:	OLS	**Adj. R-squared:**	0.896
Method:	Least Squares	**F-statistic:**	570.3
Date:	Tue, 10 Jul 2018	**Prob (F-statistic):**	1.58e-96
Time:	18:04:05	**Log-Likelihood:**	-386.18
No. Observations:	200	**AIC:**	780.4
Df Residuals:	196	**BIC:**	793.6
Df Model:	3		
Covariance Type:	nonrobust		

	coef	std err	t	P>|t|	[0.025	0.975]
const	2.9389	0.312	9.422	0.000	2.324	3.554
tv	0.0458	0.001	32.809	0.000	0.043	0.049
radio	0.1885	0.009	21.893	0.000	0.172	0.206
news	-0.0010	0.006	-0.177	0.860	-0.013	0.011

Why use the F-tests?

Question: Why compute the p-value for the F-statistic when we could compute p-values for each of the feature slopes?

- **Why, Part A:** If we do this, we're testing p different hypotheses instead of a single hypothesis
- **Why, Part B:** At $\alpha = 0.05$, how many p-values do we expect to be significant if the null hypothesis is in fact true?
  - If we had 100 parameters, about 5 would be significant **just by chance**
  - **Problem of Multiple Comparisons**

```
In [6]: model.summary()
```

Out[6]:

OLS Regression Results

Dep. Variable:	sales	**R-squared:**	0.897
Model:	OLS	**Adj. R-squared:**	0.896
Method:	Least Squares	**F-statistic:**	570.3
Date:	Tue, 10 Jul 2018	**Prob (F-statistic):**	1.58e-96
Time:	18:04:05	**Log-Likelihood:**	-386.18
No. Observations:	200	**AIC:**	780.4
Df Residuals:	196	**BIC:**	793.6
Df Model:	3		
Covariance Type:	nonrobust		

	coef	**std err**	**t**	**P>|t|**	**[0.025**	**0.975]**
const	2.9389	0.312	9.422	0.000	2.324	3.554
tv	0.0458	0.001	32.809	0.000	0.043	0.049
radio	0.1885	0.009	21.893	0.000	0.172	0.206
news	-0.0010	0.006	-0.177	0.860	-0.013	0.011

Problem of Multiple Comparisons

“In 2018, a Yale economics professor and a graduate student calculated correlations between daily changes in Bitcoin prices and hundreds of other financial variables. They found that Bitcoin prices were positively correlated with stock returns in the consumer goods and health care industries, and that they were negatively correlated with stock returns in the fabricated products and metal mining industries. “We don’t give explanations," the professor said, "we just document this behavior.” In other words, **they may as well have looked at correlations of Bitcoin prices with hundreds of lists of telephone numbers** and reported the highest correlations.”

Original article here: https://www.wired.com/story/the-exaggerated-promise-of-data-mining/

Quantifying model goodness-of-fit

Like in SLR, the MLR **sum of squared errors, SSE**, is:

$$SSE = \sum_{i=1}^{n} (y_i - \hat{y}_i)^2 = \sum (y_i - (\hat{\beta}_0 + \hat{\beta}_1 x_1 + \dots + \hat{\beta}_p x_p))^2$$

Like in SLR, the MLR **total sum of squares, SST**, is:

$$SST = \sum_{i=1}^{n} (y_i - \bar{y})^2$$

→ The **coefficient of determination, R^2**, is:

$$R^2 = 1 - \frac{SSE}{SST}$$

← how much unexplained

R^2 interpretation: The fraction of variation that IS explained by the model.

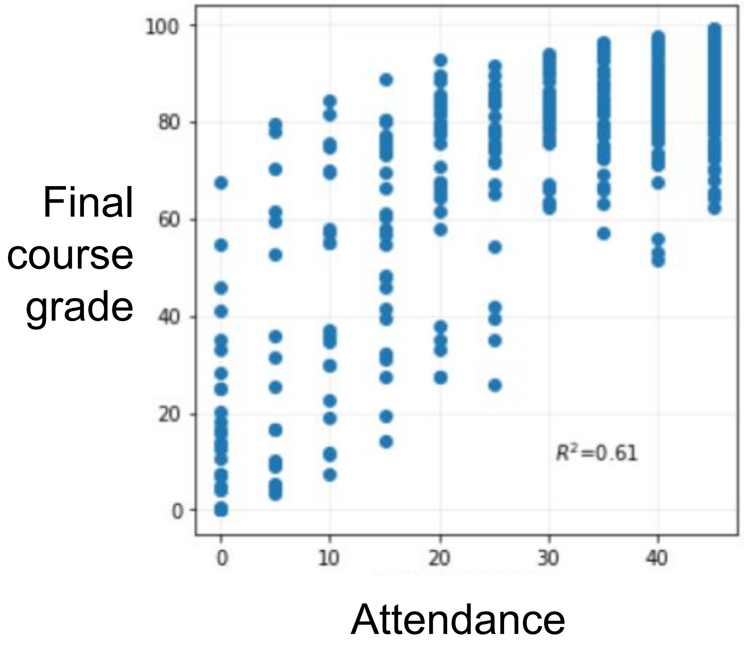
Quantifying model goodness-of-fit

R^2 interpretation: The fraction of variation that IS explained by the model.

So what does **THIS** mean? →

Expectations: Attendance

$R^2 = 0.61$



Quantifying model goodness-of-fit

Problem: The standard R^2 value can be artificially inflated by adding lots and lots of frivolous features. (You can fit **anything** with a polynomial of high enough degree!)

Example: S'pose that y represents the sale price of a house. Reasonable features associated with the sale price might include:

- x_1 -- the interior size of the house
- x_2 -- the size of the lot
- x_3 -- the number of bedrooms
- x_4 -- the number of bathrooms
- x_5 -- the age of the house

But s'pose we also add:

- x_6 -- the diameter of the doorknob on the coat closet
- x_7 -- the thickness of the cutting board in the kitchen

Quantifying model goodness-of-fit

Problem: The standard R^2 value can be artificially inflated by adding lots and lots of frivolous features. (You can fit **anything** with a polynomial of high enough degree!)

Example: S'pose that y represents the sale price of a house. Reasonable features associated with the sale price might include:

- x_1 -- the interior size of the house
- x_2 -- the size of the lot
- x_3 -- the number of bedrooms
- x_4 -- the number of bathrooms
- x_5 -- the age of the house

But s'pose we also add:

- x_6 -- the diameter of the doorknob on the coat closet
- x_7 -- the thickness of the cutting board in the kitchen

Quantifying model goodness-of-fit

The objective of MLR is not simply to explain the most variation in the data, but to do so with a model with relatively few features that are easily interpreted.

→ **principle of parsimony**

It is thus desirable to adjust R^2 to account for the size of the model (i.e., # features)

→ $R^2 = 1 - SSE/SST$, but let's *adjust* each of SSE and SST by their degrees of freedom

→ $df_{SSE} = n-p-1$ and $df_{SST} = n-1$

Definition: The **adjusted R^2** value is

$$R_a^2 = 1 - \frac{SSE/df_{SSE}}{SST/df_{SST}} = 1 - \frac{SSE/(n-p-1)}{SST/(n-1)}$$

Quantifying model goodness-of-fit

The objective of MLR is not simply to explain the most variation in the data, but to do so with a model with relatively few features that are easily interpreted.

→ **principle of parsimony**

It is thus desirable to adjust R^2 to account for the size of the model (i.e., # features)

→ $R^2 = 1 - SSE/SST$, but let's *adjust* each of SSE and SST by their degrees of freedom

→ $df_{SSE} = n-p-1$ and $df_{SST} = n-1$

adjust p, how many feature are in model

Penalizes for having too many features that doesn't reduce SSE

Definition: The **adjusted R^2** value is

$$R_a^2 = 1 - \frac{SSE/df_{SSE}}{SST/df_{SST}} = 1 - \frac{SSE/(n-p-1)}{SST/(n-1)}$$

Quantifying model goodness-of-fit

$R^2_{adj} \le R^2$ → haven't over fit our model to nonsense feature

The objective of MLR is not simply to explain the mo
model with relatively few features that are easily inte

if R^2_{adj} ↑

→ principle of parsimony

It is thus desirable to adjust R^2 to account for the siz

→ $R^2 = 1 - SSE/SST$, but let's *adjust* each of S

→ $df_{SSE} = n-p-1$ and $df_{SST} = n-1$

```
In [6]: model.summary()
```

Out[6]:

OLS Regression Results

$R^2_{adj} \le R^2$

Dep. Variable:	sales	**R-squared:**	0.897
Model:	OLS	**Adj. R-squared:**	0.896
Method:	Least Squares	**F-statistic:**	570.3
Date:	Tue, 10 Jul 2018	**Prob (F-statistic):**	1.58e-96
Time:	18:04:05	**Log-Likelihood:**	-386.18
No. Observations:	200	**AIC:**	780.4
Df Residuals:	196	**BIC:**	793.6
Df Model:	3		
Covariance Type:	nonrobust		

	coef	std err	t	P>|t|	[0.025	0.975]
const	2.9389	0.312	9.422	0.000	2.324	3.554
tv	0.0458	0.001	32.809	0.000	0.043	0.049
radio	0.1885	0.009	21.893	0.000	0.172	0.206
news	-0.0010	0.006	-0.177	0.860	-0.013	0.011

Definition: The **adjusted R^2** value is

$$R^2_a = 1 - \frac{SSE/df_{SSE}}{SST/df_{SST}} = 1 - \frac{SSE/(n-p-1)}{SST/(n-1)}$$

Quantifying model goodness-of-fit

→ is a good that our model fit or not

Definition: The **adjusted R^2** value is

$$R_a^2 = 1 - \frac{SSE/df_{SSE}}{SST/df_{SST}} = 1 - \frac{SSE/(n-p-1)}{SST/(n-1)}$$

There are LOTS of other measures of goodness of fit that also account for over-parameterization. Examples close to my heart:

- Akaike information criterion
- Bayesian information criterion
- Deviance information criterion

These require **likelihood functions**. Be aware they exist, and could be useful later in life.

Model selection: which features should we keep?

I have a great idea! Try all possible combinations of p features, and choose the best combo!

```
sales = 2.94 + 0.046×TV + 0.189×radio - 0.001×news
```

Model selection: which features should we keep?

I have a ~~great~~ terrible idea! Try all possible combinations of p features, and choose the best combo!

→ there are 2^p possible models

→ with p = 30, that's about $2^{30} \approx$ 1,000,000,000 models to test…

```
sales = 2.94 + 0.046×TV + 0.189×radio - 0.001×news
```

Model selection: which features should we keep?

4 selection, that start
with zero feature

Forward selection: A greedy algorithm for adding features

1) Fit model with an intercept but no slopes

2) Fit *p* individual SLR models -- 1 for each possible feature.
   Add the one that improves the performance the most based on some measure.
   (e.g., decreases SSE the most, or increases F-statistic the most)

3) Fit *p*-1 MLR models -- 1 for each of the remaining features, adding to the feature you added in Step 2.
   Add the one that improves model performance the most.

4) Repeat until some stopping criterion is reached.
   (e.g., some threshold SSE, or some fixed number of features)

Model selection: which features should we keep?

Backward selection: A greedy algorithm for removing features

1) Fit model with all available features
2) Remove the feature with the largest p-value (i.e., the least significant feature)

   or smallest increase in SSE, or other goodness of fit.

3) Repeat until some stopping criterion is reached. (e.g., some threshold SSE, or some fixed number of features)