

# Linear Regression Models

## P8111

3/8

Lecture 09

10:00 am → 5:00 pm

3/10

Jeff Goldsmith  
February 18, 2016

Midterm!!

↓

3B!!!



THE DEPARTMENT OF  
**BIostatISTICS**



Columbia University  
MAILMAN SCHOOL  
OF PUBLIC HEALTH

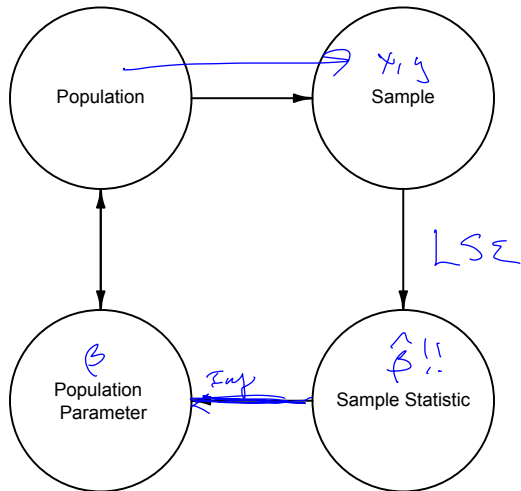
# Today's Lecture

$$\text{MLR: LS} \rightarrow \hat{\beta} = (X^T X)^{-1} X^T y$$

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

- Sampling distribution of  $\hat{\beta}$
- Hypothesis tests for individual coefficients
- Global tests

# Circle of Life

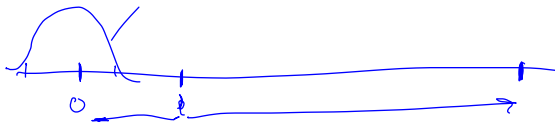


# Statistical inference

- We have LSEs  $\hat{\beta}_0, \hat{\beta}_1, \dots$ ; we want to know what this tells us about  $\beta_0, \beta_1, \dots$
- Two basic tools are confidence intervals and hypothesis tests
  - ▶ Confidence intervals provide a plausible range of values for the parameter of interest based on the observed data
  - ▶ Hypothesis tests ask how probable are the data we gathered under a null hypothesis about the data generating distribution

# A quick word about p-values

P-values ...



- Are not universally adored
  - ▶ Compares data vs null (usually no effect) rather than testing whether data are consistent with your real hypothesis
  - ▶ Often misinterpreted (“probability the null is true”)
- Can get people in trouble
  - ▶ Especially when misinterpreted ✓
- Are still the default tool for inference

*“why most published research findings are false”*

# Motivation

## Recall the MLB data:

```
> setwd("~/Desktop")
> download.file("http://www.openintro.org/stat/data/mlb11.RData", destfile = "mlb11.RData")
> load("mlb11.RData")
>
> mlb11 %>% tbl_df
Source: local data frame [30 x 12]
```

	team (fctr)	runs (int)	at_bats (int)	hits (int)	homeruns (int)	bat_avg (dbl)	strikeouts (int)	stolen_bases (int)	wins (int)	new
1	Texas Rangers	855	5659	1599	210	0.283	930	143	96	
2	Boston Red Sox	875	5710	1600	203	0.280	1108	102	90	
3	Detroit Tigers	787	5563	1540	169	0.277	1143	49	95	
4	Kansas City Royals	730	5672	1560	129	0.275	1006	153	71	
5	St. Louis Cardinals	762	5532	1513	162	0.273	978	57	90	
6	New York Mets	718	5600	1477	108	0.264	1085	130	77	
7	New York Yankees	867	5518	1452	222	0.263	1138	147	97	
8	Milwaukee Brewers	721	5447	1422	185	0.261	1083	94	96	
9	Colorado Rockies	735	5544	1429	163	0.258	1201	118	73	
10	Houston Astros	615	5598	1442	95	0.258	1164	118	56	
..	...	...	...	...	...	...	...	...	...	...

Variables not shown: new\_slug (dbl), new\_obs (dbl)

# Motivation

```
Call:
lm(formula = runs ~ at_bats + hits + homeruns + stolen_bases,
    data = mlb11)
```

```
...
Coefficients:
```

	Estimate
(Intercept)	581.2110
at_bats	-0.2023
hits	0.6974
homeruns	1.2535
stolen_bases	0.5230

```
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

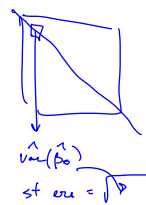
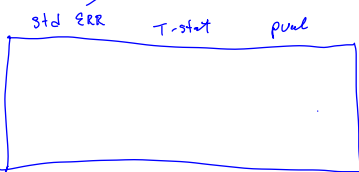
```
Residual standard error: 26.85 on 25 degrees of freedom
```

```
Multiple R-squared: 0.9087, Adjusted R-squared: 0.894
```

```
F-statistic: 62.17 on 4 and 25 DF, p-value: 1.26e-12
```

$$\hat{\beta} = (X^T X)^{-1} X^T y$$

$$\text{var}(\hat{\beta}) = \frac{1}{\sigma^2} (X^T X)^{-1}$$



# Motivation

$\hat{\beta}$

- Can we say anything about whether the effect of stolen\_bases is "significant" after adjusting for other variables?  $H_0: \hat{\beta}_{SB} = 0$
- Can we compare this model to a model with only hits and homeruns?

$$y_i = \beta_0 + \beta_1 \text{at} + \beta_2 \text{hits} + \beta_3 \text{HR} + \beta_4 \text{SB} + \epsilon$$

$v_3$

$$\hookrightarrow y_i = \beta_0 + \beta_1 \text{hits} + \beta_2 \text{HR} + \epsilon$$

$H_0: \beta_4 = 0$  then  $H_0: \beta_1 = 0$

$H_0: \beta_4 = \beta_1 = 0$



# Sampling distribution

$$\hat{\beta} = (X^T X)^{-1} X^T y \quad \hat{\beta} \sim (\beta, \sigma^2 (X^T X)^{-1})$$

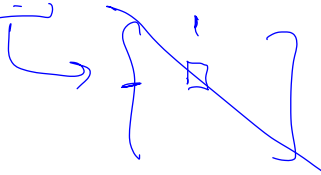
$\hookrightarrow N(X\beta, \sigma^2 I)$

If our usual assumptions are satisfied and  $\epsilon \stackrel{iid}{\sim} \underline{N}[0, \sigma^2]$  then

$$\hat{\beta} \sim N[\beta, \sigma^2 (X^T X)^{-1}]$$

$$\hat{\beta}_j \sim N[\beta_j, \sigma^2 (X^T X)^{-1}_{jj}]$$

- This will be used for inference.



# Asymptotic distribution

Assume that

- $E(\epsilon_i | \mathbf{x}_i) = 0 \forall i;$
- $Var(\epsilon_i | \mathbf{x}_i) = \sigma^2 \forall i;$
- $n \xrightarrow{\text{lim}} \infty \frac{\mathbf{X}^T \mathbf{X}}{n} \rightarrow Q$  where  $Q$  is a finite non-singular matrix.

Then

$$\sqrt{n}(\hat{\beta} - \beta) \rightarrow N \left[ 0, \sigma^2 Q^{-1} \right]$$

(This is essentially an extension of the central limit theorem)

# Simulations exploring distributions

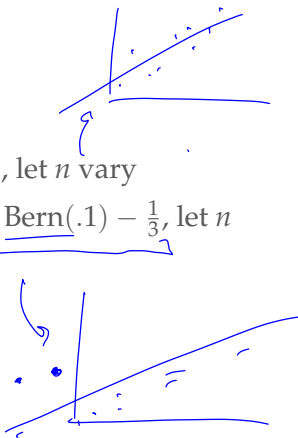
$$n \in \{10, 100, 1000\}$$

Look at SLR

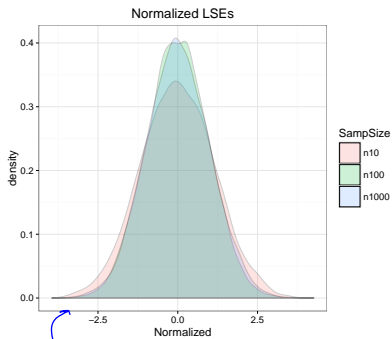
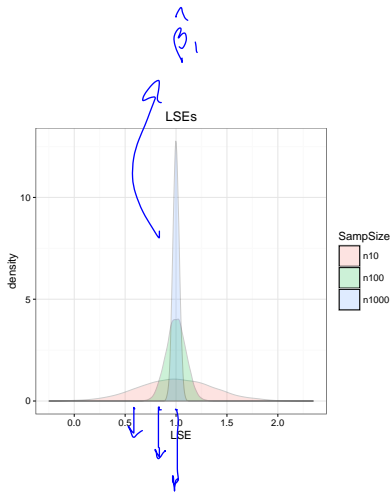
$$y_i = \underline{0} + \underline{1}x_i + \epsilon_i$$

under various conditions.

- First simulations: errors follow  $N[0, 1]$ , let  $n$  vary
- Second simulations: errors follow  $\frac{10}{3} * \text{Bern}(.1) - \frac{1}{3}$ , let  $n$  vary
- In both cases,  $\epsilon \sim \underline{(0, 1)}$

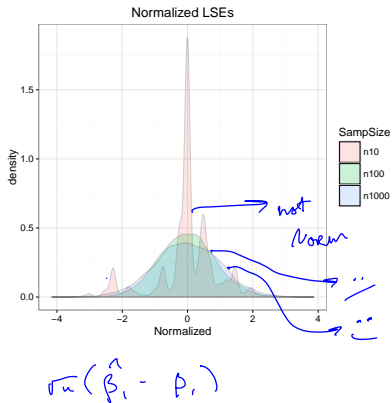
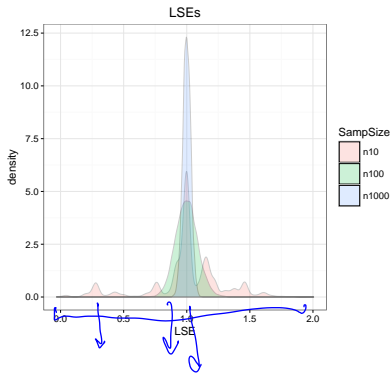


# Normal errors



$$\sqrt{n}(\hat{\beta}_1 - \beta_1)$$

# Non-normal errors



# Testing procedure

Calculate the probability of the observed data (or more extreme data) under a null hypothesis.

- Often  $H_0 : \beta_1 = 0$  and  $H_a : \beta_1 \neq 0$
- Set  $\alpha = P(\text{falsely rejecting a true null hypothesis})$  (type I error rate) *0.05*

■ Calculate a test statistic assuming the null hypothesis is true

- Compute a p-value =

$$P(\text{As or more extreme test statistic} | H_0)$$

- Reject or fail to reject  $H_0$

# Testing

$$\hat{\beta}_j \sim \mathcal{N}(\beta_j, \sigma^2 (X^T X)^{-1}_{jj})$$

For real data we have to estimate  $\sigma^2$  as well as  $\beta$ .

- Recall our estimate of the error variance is

$$\hat{\sigma}^2 = \frac{RSS}{n - p - 1} = \frac{\sum_i (y_i - \hat{y}_i)^2}{n - p - 1}$$

- With Normally distributed errors, it can be shown that

$$(n - p - 1) \frac{\hat{\sigma}^2}{\sigma^2} \sim \chi^2_{n-p-1}$$

Implication is that test statistics follow a t distribution rather than Normal with  $df = n - p - 1$

# Individual coefficients

For individual coefficients

- We can use the test statistic

$$T = \frac{\hat{\beta}_j - \beta_j}{\widehat{se}(\hat{\beta}_j)} = \frac{\hat{\beta}_j - \beta_j}{\sqrt{\hat{\sigma}^2(\mathbf{X}^T \mathbf{X})_{jj}^{-1}}} \sim t_{n-p-1}$$

- For a two-sided test of size  $\alpha$ , we reject if

$$|T| > t_{1-\alpha/2, n-p-1}$$

- The p-value gives  $P(t_{n-p-1} > T_{obs} | H_0)$

Note that  $t$  is a symmetric distribution that converges to a Normal as  $n - p - 1$  increases.



# Example revisited

```
Call:
lm(formula = runs ~ at_bats + hits + homeruns + stolen_bases,
    data = mlb11)
```

```
...
```

```
Coefficients:
```

	Estimate	Std. Error	t value	Pr(> t )	
(Intercept)	581.2110	526.4063	1.104	0.28006	
at_bats	-0.2023	0.1174	-1.724	0.09706	.
hits	0.6974	0.1131	6.164	1.91e-06	***
homeruns	1.2535	0.1593	7.868	3.18e-08	***
stolen_bases	0.5230	0.1686	3.101	0.00473	**

```
---
```

```
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
Residual standard error: 26.85 on 25 degrees of freedom
```

```
Multiple R-squared:  0.9087, Adjusted R-squared:  0.894
```

```
F-statistic: 62.17 on 4 and 25 DF,  p-value: 1.26e-12
```

# Inference for linear combinations

Sometimes we are interested in making claims about  $c^T \beta$  for some  $c$ .

- Define  $H_0 : c^T \beta = c^T \beta_0$  or  $H_0 : c^T \beta = 0$
- We can use the test statistic

$$T = \frac{c^T \hat{\beta} - c^T \beta_0}{\widehat{se}(c^T \hat{\beta})} = \frac{c^T \hat{\beta} - c^T \beta_0}{\sqrt{\hat{\sigma}^2 c^T (\mathbf{X}^T \mathbf{X})^{-1} c}}$$

- This test statistic is asymptotically Normally distributed
- For a two-sided test of size  $\alpha$ , we reject if

$$|T| > z_{1-\alpha/2}$$

# Inference about multiple coefficients

Our model contains multiple parameters; often we want to perform multiple tests:

$$H_{01} : \beta_1 = 0$$

$$H_{02} : \beta_2 = 0$$

$$\vdots = \vdots$$

$$H_{0k} : \beta_k = 0$$

where each test has a size of  $\alpha$

- For any individual test,  $P(\text{reject } H_{0i} | H_{0i}) = \alpha$

# Inference about multiple coefficients

What about

$$P(\text{reject at least one } H_{0i} | \text{all } H_{0i} \text{ are true}) = \alpha$$

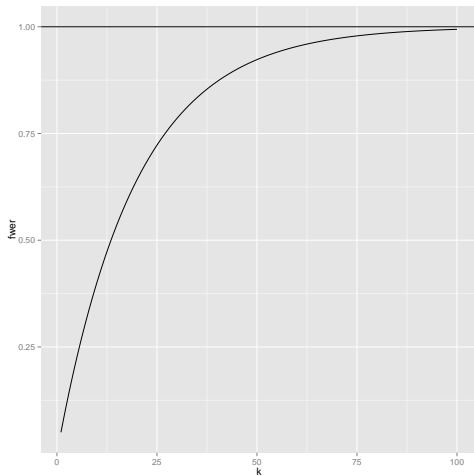
# Family-wise error rate

To calculate the FWER

- First note  $P(\text{no rejections}|\text{all } H_{0i} \text{ are true}) = (1 - \alpha)^k$
- It follows that  
 $P(\text{at least one rejection}|\text{all } H_{0i} \text{ are true}) = 1 - (1 - \alpha)^k$
- Further,

$$\begin{aligned} FWER = 1 - (1 - \alpha)^k &= 1 - \left(1 - \frac{k\alpha}{k}\right)^k \\ &\approx 1 - \exp(1 - k\alpha) \\ &\approx 1 - (1 - k\alpha) \\ &= k\alpha \end{aligned}$$

# Family-wise error rate



# Addressing multiple comparisons

## Three general approaches

- Do nothing in a reasonable way
  - ▶ Define comparisons and expectations ahead of time
  - ▶ Don't trust scientifically implausible results
  - ▶ Don't over-emphasize isolated findings
- Correct for multiple comparisons
  - ▶ Often, use the Bonferroni correction and use  $\alpha_i = \alpha/k$  for each test
  - ▶ Thanks to the Bonferroni inequality, this gives an overall  $FWER \leq \alpha$
  - ▶ Control false discovery rate
- Use a global test

# Global tests

Compare a smaller “null” model to a larger “alternative” model

- Smaller model must be nested in the larger model
- That is, the smaller model must be a special case of the larger model
- For both models, the  $RSS$  gives a general idea about how well the model is fitting
- In particular, something like

$$\frac{RSS_S - RSS_L}{RSS_L}$$

compares the relative  $RSS$  of the models



# Nested models

- These models are nested:

Smaller = Regression of  $Y$  on  $X_1$

Larger = Regression of  $Y$  on  $X_1, X_2, X_3, X_4$

- These models are not:

Smaller = Regression of  $Y$  on  $X_2$

Larger = Regression of  $Y$  on  $X_1, X_3$

# Global $F$ tests

- Compute the test statistic

$$F_{obs} = \frac{(RSS_S - RSS_L)/(df_S - df_L)}{RSS_L/df_L}$$

- If  $H_0$  (the null model) is true, then  $F_{obs} \sim F_{df_S - df_L, df_L}$
- Note  $df_S = n - p_S - 1$  and  $df_L = n - p_L - 1$
- We reject the null hypothesis if the p-value is above  $\alpha$ , where

$$\text{p-value} = P(F_{df_S - df_L, df_L} > F_{obs})$$

# Global $F$ tests

There are a couple of important special cases for the  $F$  test

- The null model contains the intercept only
  - ▶ When people say ANOVA, this is often what they mean (although all  $F$  tests are based on an analysis of variance)
- The null model and the alternative model differ only by one term
  - ▶ Gives a way of testing for a single coefficient
  - ▶ Turns out to be equivalent to a two-sided  $t$ -test:  $t_{df_L}^2 \sim F_{1,df_L}$

# MLB data

You can test multiple coefficient simultaneously using the  $F$  test

```
> linmod.null1 = lm(runs ~ hits + homeruns, data = mlb11)
> anova(linmod.null1, linmod)
Analysis of Variance Table
```

```
Model 1: runs ~ hits + homeruns
```

```
Model 2: runs ~ at_bats + hits + homeruns + stolen_bases
```

	Res.Df	RSS	Df	Sum of Sq	F	Pr(>F)
1	27	27128				
2	25	18020	2	9107.8	6.3178	0.006015 **

```
---
```

```
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

# MLB data

The  $F$  test is equivalent to the  $t$  test when there's only one parameter of interest

```
> linmod.null12 = lm(runs ~ at_bats + hits + homeruns, data = mlb11)
```

```
> anova(linmod.null12, linmod)
```

```
Analysis of Variance Table
```

```
Model 1: runs ~ at_bats + hits + homeruns
```

```
Model 2: runs ~ at_bats + hits + homeruns + stolen_bases
```

	Res.Df	RSS	Df	Sum of Sq	F	Pr(>F)
1	26	24953				
2	25	18020	1	6932.7	9.618	0.004728 **

```
---
```

```
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

# MLB data

By default, R's `summary` function compares to an intercept-only null model

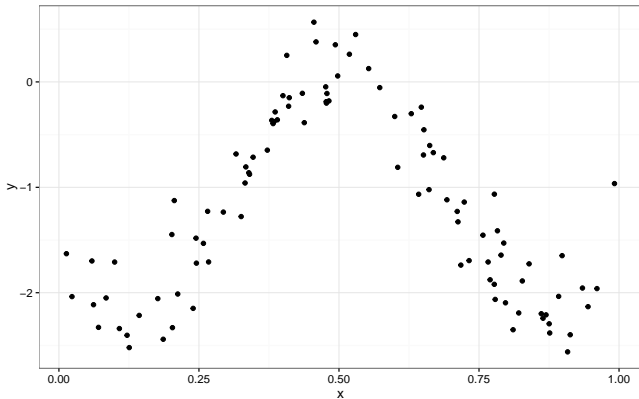
```
> linmod.null13 = lm(runs ~ 1, data = mlb11)
> anova(linmod.null13, linmod)
Analysis of Variance Table

Model 1: runs ~ 1
Model 2: runs ~ at_bats + hits + homeruns + stolen_bases
  Res.Df  RSS Df Sum of Sq    F    Pr(>F)
1      29 197281
2       25  18020  4    179261 62.174 1.26e-12 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

# Test for “linearity”

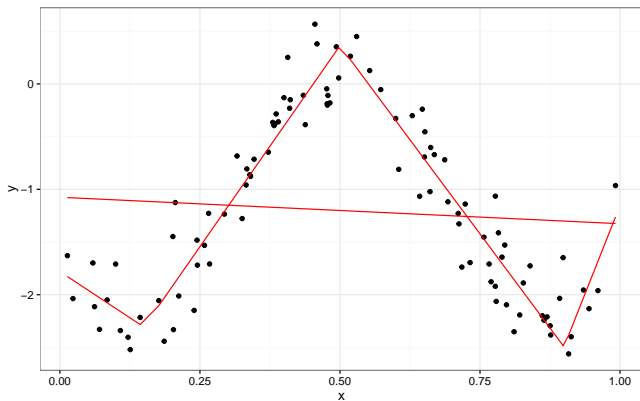
- To test more flexible vs less flexible approaches to non-linearity, we can often use global tests
  - ▶ Polynomials and piecewise linear models have “linear” associations as nested model; B-splines don’t
- Global  $F$  tests can be pretty useful here

# Testing for linearity





# Testing for linearity



# Testing linearity

```
> piecewise.underfit = lm(y ~ x, data = data.nonlin)
> piecewise.fit = lm(y ~ x + spline_15 + spline_5 + spline_9, data = data.nonlin)
> anova(piecewise.underfit, piecewise.fit)
```

Analysis of Variance Table

Model 1: y ~ x

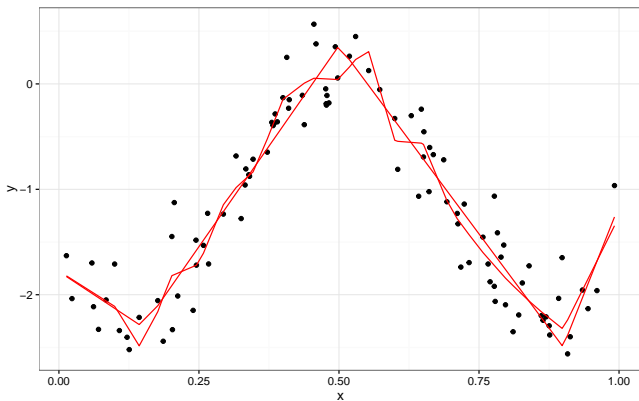
Model 2: y ~ x + spline\_15 + spline\_5 + spline\_9

	Res.Df	RSS	Df	Sum of Sq	F	Pr(>F)
1	98	73.444				
2	95	8.240	3	65.205	250.6	< 2.2e-16 ***

---

Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

# Testing comparing twenty polynomials to four



# Testing comparing twenty polynomials to four

```
> piecewise.overfit = lm(y ~ x + spline_1 + spline_15 + spline_2 + spline_25 + spline_3 + spl
+ spline_4 + spline_45 + spline_5 + spline_55 + spline_6 + spline_65
+ spline_7 + spline_75 + spline_8 + spline_85 + spline_9,
+ data = data.nonlin)
> anova(piecewise.fit, piecewise.overfit)
Analysis of Variance Table
```

Model 1:  $y \sim x + \text{spline}_{15} + \text{spline}_5 + \text{spline}_9$

Model 2:  $y \sim x + \text{spline}_1 + \text{spline}_{15} + \text{spline}_2 + \text{spline}_{25} + \text{spline}_3 +$   
 $\text{spline}_{35} + \text{spline}_4 + \text{spline}_{45} + \text{spline}_5 + \text{spline}_{55} +$   
 $\text{spline}_6 + \text{spline}_{65} + \text{spline}_7 + \text{spline}_{75} + \text{spline}_8 +$   
 $\text{spline}_{85} + \text{spline}_9$

	Res.Df	RSS	Df	Sum of Sq	F	Pr(>F)
1	95	8.2395				
2	81	6.7862	14	1.4533	1.239	0.2645

# Testing comparing twenty polynomials to four

```
> anova(piecewise.underfit, piecewise.fit, piecewise.overfit)
```

```
Analysis of Variance Table
```

```
Model 1: y ~ x
```

```
Model 2: y ~ x + spline_15 + spline_5 + spline_9
```

```
Model 3: y ~ x + spline_1 + spline_15 + spline_2 + spline_25 + spline_3 +  
spline_35 + spline_4 + spline_45 + spline_5 + spline_55 +  
spline_6 + spline_65 + spline_7 + spline_75 + spline_8 +  
spline_85 + spline_9
```

	Res.Df	RSS	Df	Sum of Sq	F	Pr(>F)
1	98	73.444				
2	95	8.240	3	65.205	259.427	<2e-16 ***
3	81	6.786	14	1.453	1.239	0.2645

```
---
```

```
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

# Today's big ideas

- Inference for multiple linear regression models
- 

- Suggested reading: Faraway Ch 3.1 - 3.3