

Data Science School at DKE

Summer 2019

Christof Seiler

Assistant Professor

Department of Data Science and Knowledge Engineering (DKE)

<http://christofseiler.github.io>

Maastricht University, June 2019

About Me – My Background

- Assistant Professor (DKE, Maastricht University)
 - Teaching:
 - Statistics and Software Engineering courses
 - Research:
 - Statistical modeling of complex data
 - Omics (CyTOF and RNA-seq) and imaging data (2d and 3d)
 - Uncertainty quantification
 - Convergence of computer simulations
- Postdoc in **Statistics** (Stanford University)
- PhD in
 - Computer Science** (Inria, France) and
 - Biomedical Engineering** (University of Bern, Switzerland)

Topics of Today

1. What is Data Science?
2. Computer Simulations
3. The Bootstrap
4. Regularized Regression

What is Data Science?

What?

- Sciences are primarily defined by their **questions** not their tools

What?

- Sciences are primarily defined by their **questions** not their tools
- Example: **Astrophysics** is the discipline that learns the composition of the stars, not the discipline that uses the spectroscope

What?

- Sciences are primarily defined by their **questions** not their tools
- Example: **Astrophysics** is the discipline that learns the composition of the stars, not the discipline that uses the spectroscope
- Definition: **Data science** is the discipline that **describes**, **predicts**, and **makes causal inferences**, not the discipline that uses machine learning algorithms or other technical tools

Classification of Tasks

Data Science Task			
	Description	Prediction	Causal inference
Example of scientific question	How can women aged 60–80 years with stroke history be partitioned in classes defined by their characteristics?	What is the probability of having a stroke next year for women with certain characteristics?	Will starting a statin reduce, on average, the risk of stroke in women with certain characteristics?
Data	<ul style="list-style-type: none">• Eligibility criteria• Features (symptoms, clinical parameters ...)	<ul style="list-style-type: none">• Eligibility criteria• Output (diagnosis of stroke over the next year)• Inputs (age, blood pressure, history of stroke, diabetes at baseline)	<ul style="list-style-type: none">• Eligibility criteria• Outcome (diagnosis of stroke over the next year)• Treatment (initiation of statins at baseline)• Confounders• Effect modifiers (optional)
Examples of analytics	Cluster analysis ...	Regression Decision trees Random forests Support vector machines Neural networks ...	Regression Matching Inverse probability weighting G-formula G-estimation Instrumental variable estimation ...

Source: Hernán, Hsu, and Healy (2019)

Why Now?

- **More data**
- **Cheaper computers**
- The field itself has existed for 50 years already (Donoho 2017)
- Two cultures (Breiman 2001)
 - Prediction: To be able to predict what the responses are going to be to future input variables
 - Inference: To [infer] how nature is associating the response variables to the input variables
- The predictive culture is currently winning because of **The Common Task Framework**

The Common Task Framework

Donoho (2017):

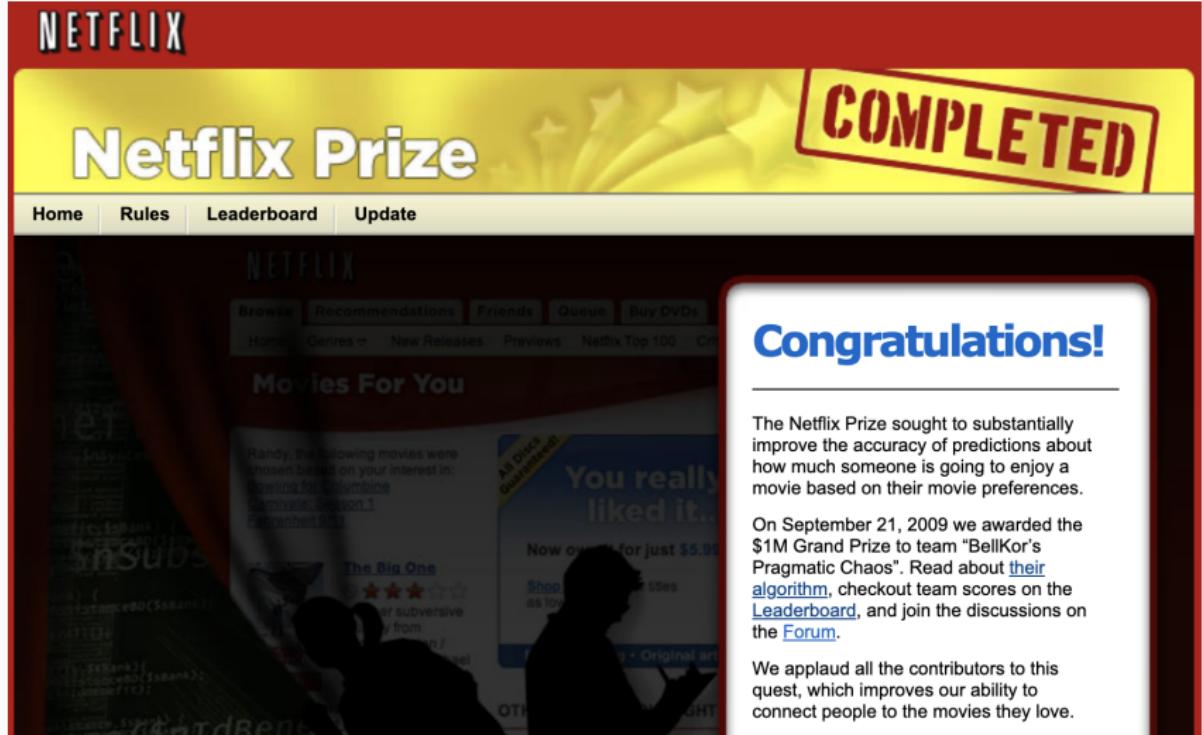
- (a) A **publicly available training dataset** involving, for each observation, a list of (possibly many) feature measurements, and a class label for that observation.
- (b) A set of enrolled **competitors** whose common task is to infer a class prediction rule from the training data.
- (c) A **scoring referee**, to which competitors can submit their prediction rule. The referee runs the prediction rule against a testing dataset which is sequestered behind a Chinese wall. The referee objectively and automatically reports the score (prediction accuracy) achieved by the submitted rule.

The Common Task Framework: Netflix Prize

The screenshot shows the Netflix Prize Leaderboard page. At the top, the Netflix logo is on the left, and a large red stamp on the right reads "COMPLETED". Below the stamp, the word "Leaderboard" is displayed in a large blue font. A horizontal menu bar at the top includes links for Home, Rules, Leaderboard, and Update. The main content area is titled "Leaderboard" and displays a table of results. The table has columns for Rank, Team Name, Best Test Score, % Improvement, and Best Submit Time. The top row highlights the Grand Prize winner: BellKor's Pragmatic Chaos with an RMSE of 0.8567. The table lists 12 teams, all of which have submitted their best scores by July 26, 2009.

Rank	Team Name	Best Test Score	% Improvement	Best Submit Time
Grand Prize - RMSE = 0.8567 - Winning Team: BellKor's Pragmatic Chaos				
1	BellKor's Pragmatic Chaos	0.8567	10.06	2009-07-26 18:18:28
2	The Ensemble	0.8567	10.06	2009-07-26 18:38:22
3	Grand Prize Team	0.8582	9.90	2009-07-10 21:24:40
4	Opera Solutions and Vandelay United	0.8588	9.84	2009-07-10 01:12:31
5	Vandelay Industries I	0.8591	9.81	2009-07-10 00:32:20
6	PragmaticTheory	0.8594	9.77	2009-06-24 12:06:56
7	BellKor in BigChaos	0.8601	9.70	2009-05-13 08:14:09
8	Dace	0.8612	9.59	2009-07-24 17:18:43
9	Feeds2	0.8622	9.48	2009-07-12 13:11:51
10	BigChaos	0.8623	9.47	2009-04-07 12:33:59
11	Opera Solutions	0.8623	9.47	2009-07-24 00:34:07
12	BellKor	0.8624	9.46	2009-07-26 17:19:11

The Common Task Framework: Netflix Prize



The screenshot shows the official Netflix Prize website. At the top, the Netflix logo is on the left, and a large yellow banner features the words "Netflix Prize" in white. To the right of the banner is a red "COMPLETED" stamp. Below the banner is a navigation bar with links for "Home", "Rules", "Leaderboard", and "Update". The main content area has a dark background with a blurred image of a movie screen showing a couple. A "NETFLIX" logo is centered above a menu bar with options like "Browse", "Recommendations", "Friends", "Queue", and "Buy DVDs". Below the menu, a section titled "Movies For You" lists recommended movies. A prominent blue box on the right contains the text "Congratulations!" in large blue letters. To the right of this box is a paragraph about the Netflix Prize's goal of improving movie predictions. Below that is another paragraph about the award ceremony and links to team details, the leaderboard, and forum discussions. At the bottom right of the page is the number "9".

COMPLETED

Congratulations!

The Netflix Prize sought to substantially improve the accuracy of predictions about how much someone is going to enjoy a movie based on their movie preferences.

On September 21, 2009 we awarded the \$1M Grand Prize to team "BellKor's Pragmatic Chaos". Read about [their algorithm](#), checkout team scores on the [Leaderboard](#), and join the discussions on the [Forum](#).

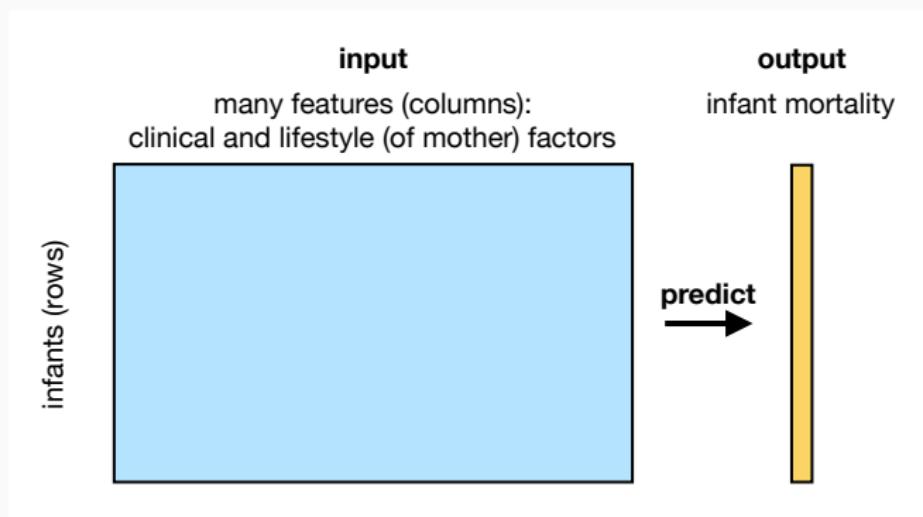
We applaud all the contributors to this quest, which improves our ability to connect people to the movies they love.

Prediction

- Google's algorithm to **diagnose** diabetic retinopathy (after 54 ophthalmologists classified more than 120,000 images)
- Microsoft's algorithm to **predict** pancreatic cancer months before its usual diagnosis (using the online search histories of 3,000 users who were later diagnosed with cancer), and
- Facebook's algorithm to **detect** users who may be suicidal (based on posts and live videos)

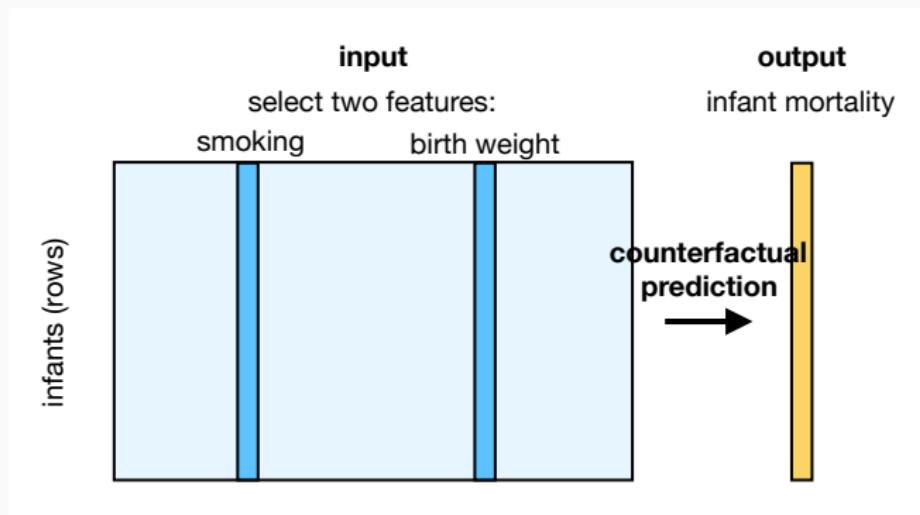
Prediction vs. Causal Inference

- Prediction: Large health records database to predict infant mortality from clinical and lifestyle factors

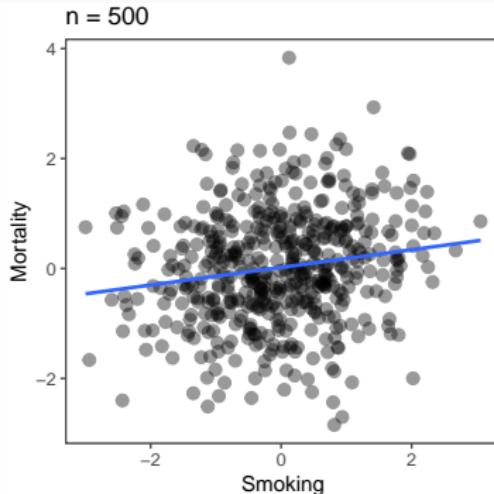
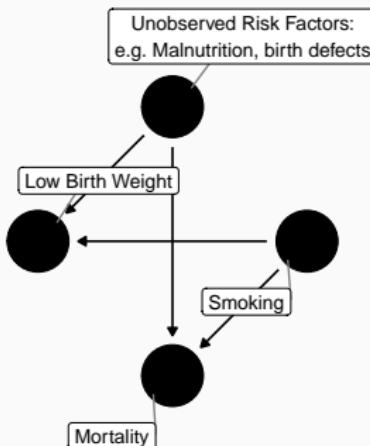


Prediction vs. Causal Inference: Birth Weight Paradox

- Causality: Answer what if statements, e.g. if a mother stops smoking during pregnancy does this reduce infant mortality



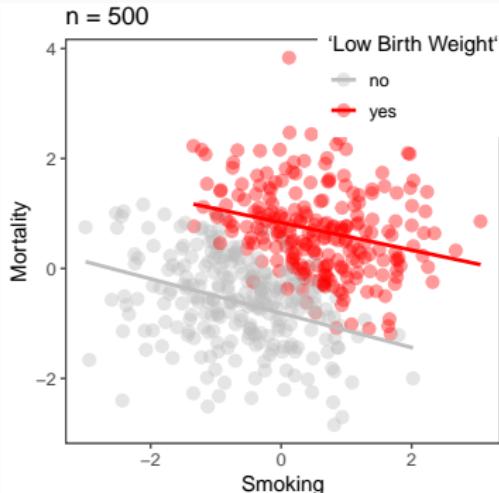
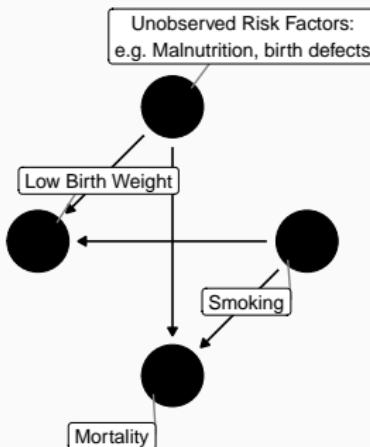
Prediction vs. Causal Inference: Birth Weight Paradox



```
lm(formula = Mortality ~ Smoking,  
  data = health_records) %>% tidy
```

```
## # A tibble: 2 x 5  
##   term      estimate std.error statistic p.value  
##   <chr>      <dbl>     <dbl>     <dbl>    <dbl>  
## 1 (Intercept)  0.0208    0.0453    0.460  0.646  
## 2 Smoking       0.161     0.0429    3.75   0.000200
```

Prediction vs. Causal Inference: Birth Weight Paradox



```
lm(formula = Mortality ~ Smoking + `Low Birth Weight` ,  
  data = health_records) %>% tidy
```

```
## # A tibble: 3 x 5  
##   term            estimate std.error statistic p.value  
##   <chr>          <dbl>     <dbl>      <dbl>    <dbl>  
## 1 (Intercept) -0.795     0.0526    -15.1  1.01e-42  
## 2 Smoking      -0.284     0.0387    -7.34  8.57e-13  
## 3 `Low Birth Weight`yes  1.65      0.0818   20.2  1.61e-66
```

Prediction vs. Causal Inference: Birth Weight Paradox

- Birth weight is strongly associated with both maternal smoking and infant mortality
- Adjustment for it **induces bias**
- This bias is often referred to as the “**birth weight paradox**”:
 - Low birth weight babies from mothers who smoked during pregnancy have a lower mortality than those from mothers who did not smoke during pregnancy (Hernández-Díaz, Schisterman, and Hernán 2006)

Computer Simulations

Computer Age



Books freely available:

- Efron and Hastie (2016): [website](#)
- Holmes and Huber (2019): [website](#)

To quote Andrew Gelman ([source](#)):

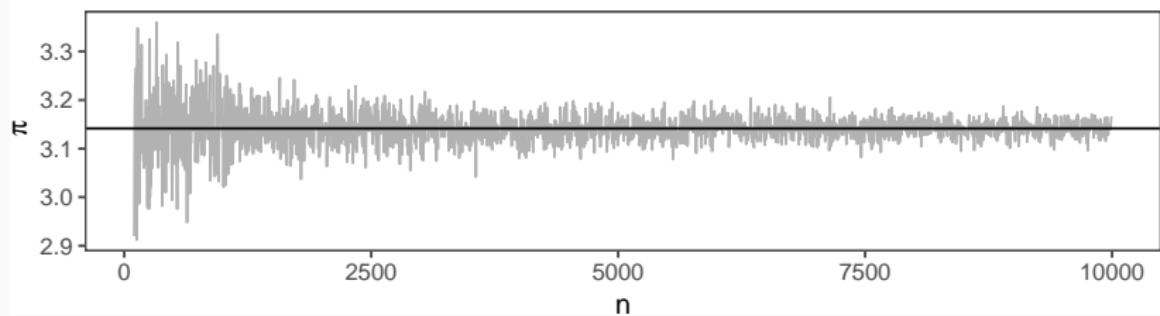
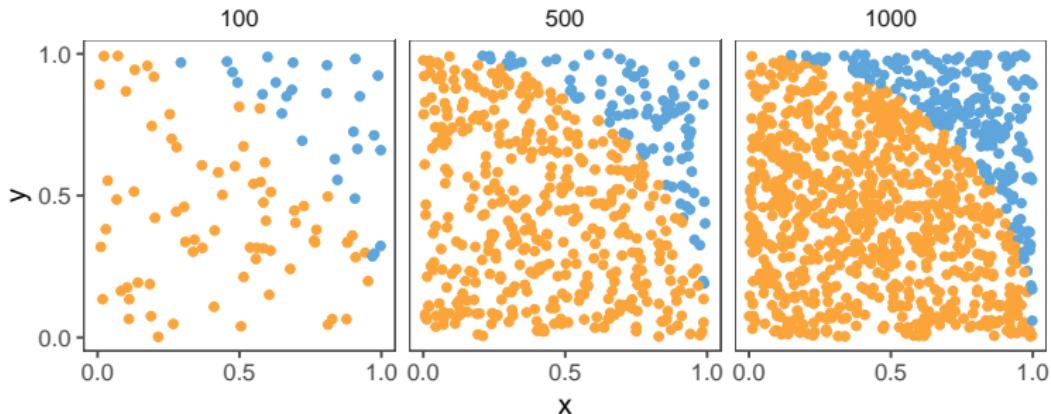
"If you wanted to do foundational research in statistics in the mid-twentieth century, you had to be bit of a mathematician, ... if you want to do statistical research at the turn of the twenty-first century, you have to be a computer programmer."



- RStudio Cloud for labs: <https://rstudio.cloud/project/350555>

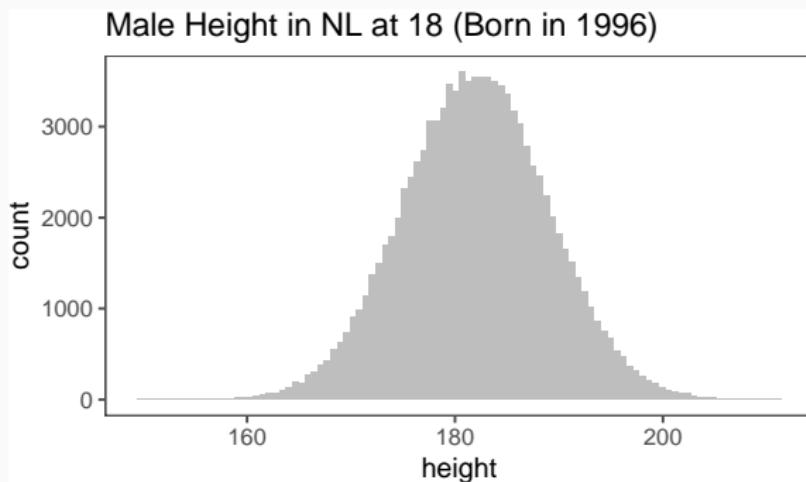
Simulations Example

Approximating π



The Bootstrap

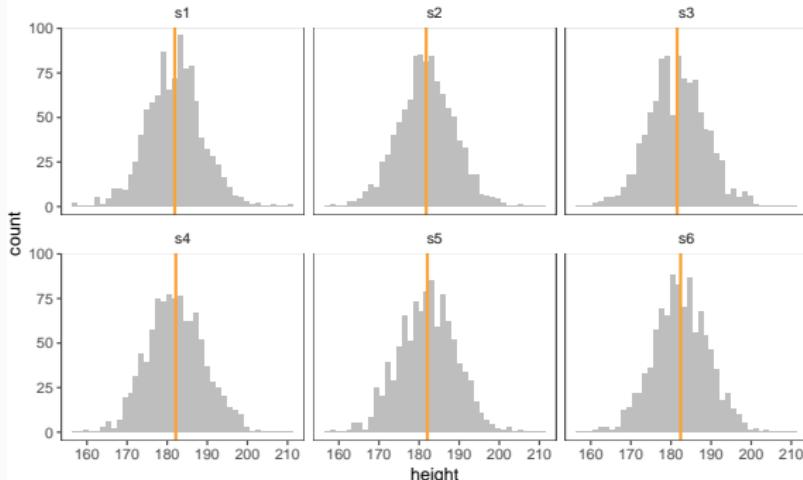
Height Example



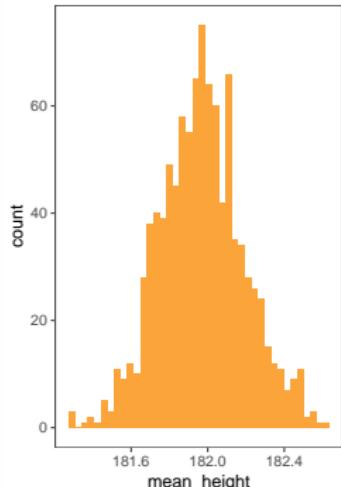
- Let's assume that we measured all 18 year old Dutch male born in 1996

Height Example

Six Samples of 1000 Male Height in NL at 18 (Born in 1996)

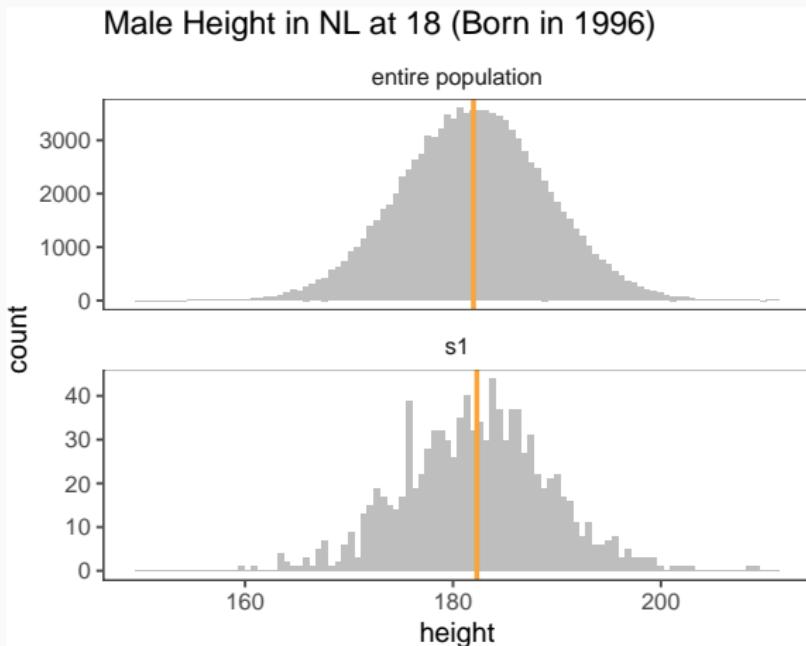


Sampling Distribution



- In real life, too expensive
- We can only take samples from the population
- Sample surveys: **s1, s2, s3, s4, s5, and s6**

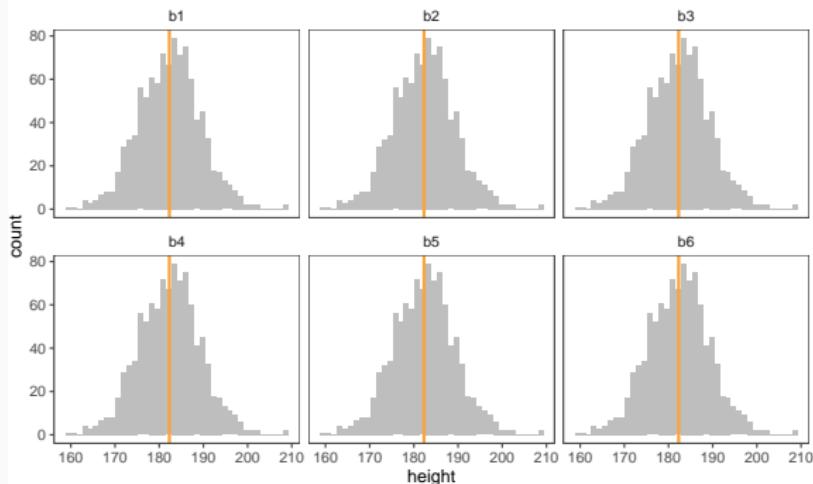
Height Example



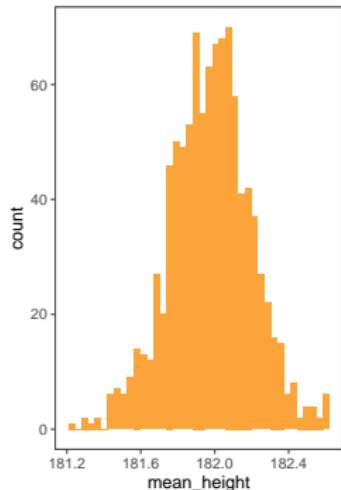
- Compare **population** with sample **s1**

Height Example

Six Samples of 1000 Male Height in NL at 18 (Born in 1996)

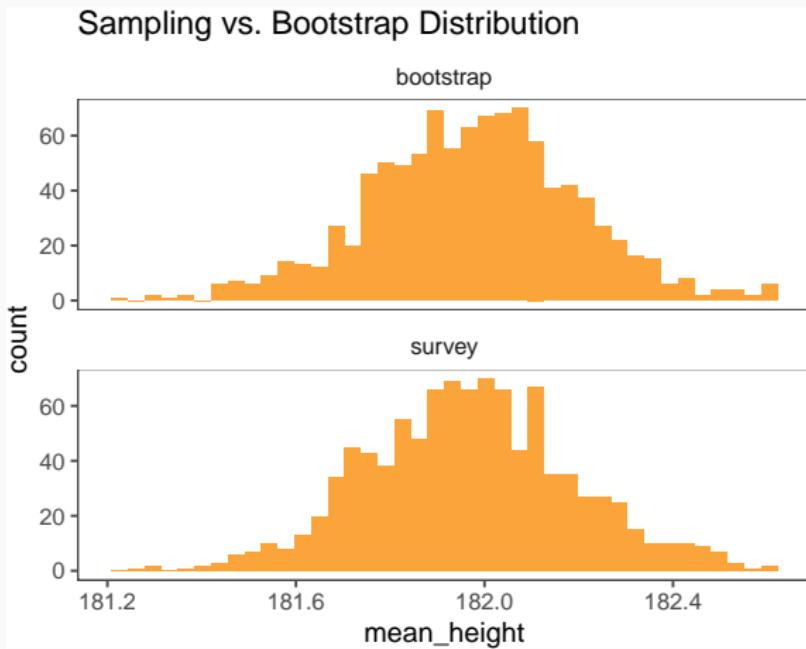


Bootstrap Distribution



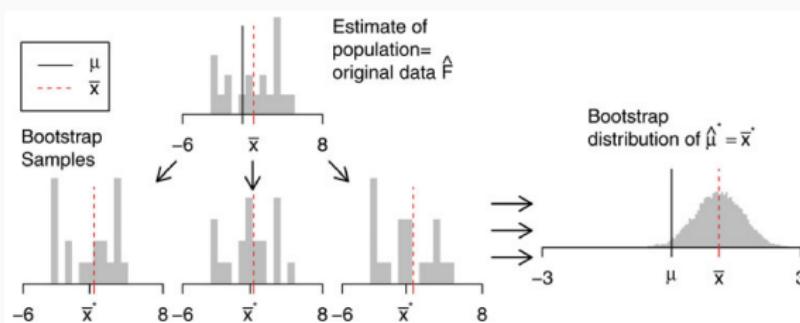
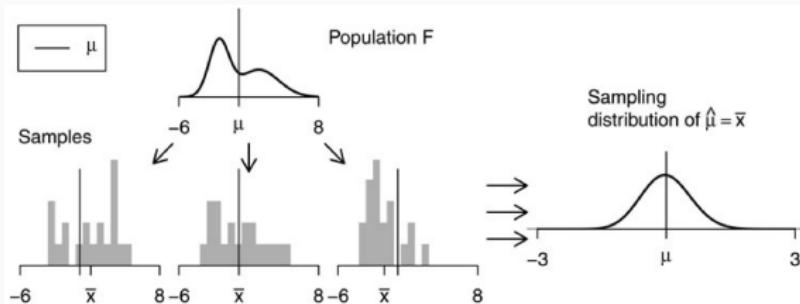
- Bootstrap samples: **b1, b2, b3, b4, b5, and b6**

Height Example



- Compare distributions

Main Idea

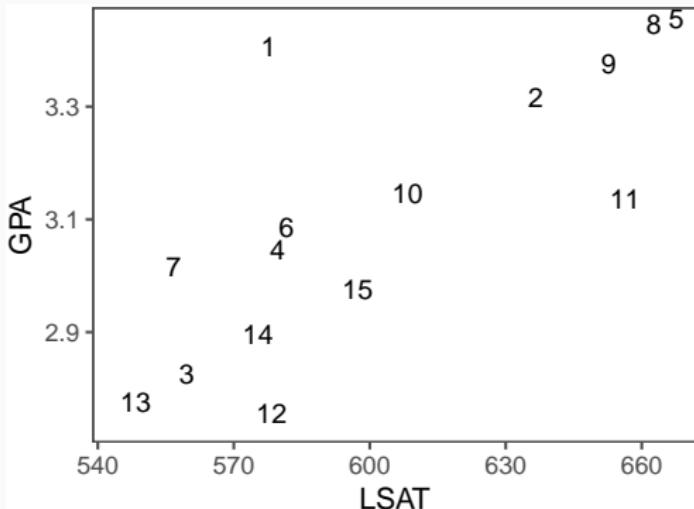


(Hesterberg 2015)

Monte Carlo Simulations

- If number of observations is **small**,
then we can do **exhaustive bootstrap**
- If number of observations is **large**,
then we can do **Monte Carlo simulations**

Law Schools Example



- Sample correlation coefficient:

```
theta_hat = cor(law$LSAT, law$GPA)  
theta_hat
```

```
## [1] 0.7763745
```

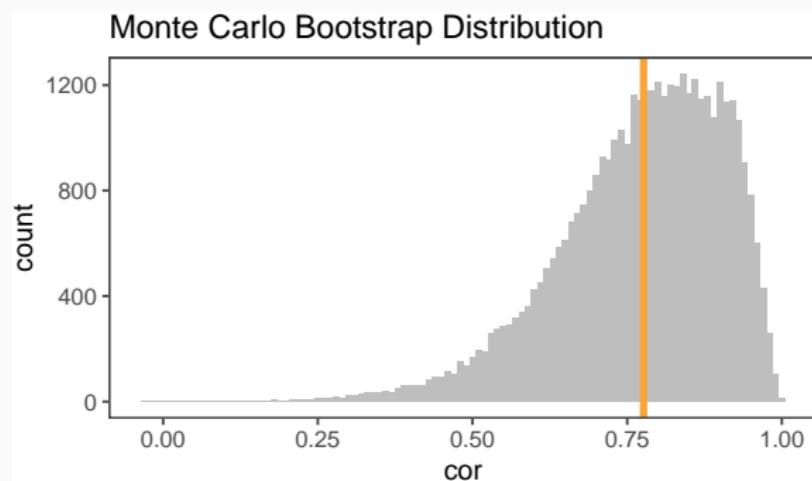
Law Schools Example

- How accurate is this estimate?
- Let's look at the bootstrap distribution:

```
draw_bootstrap_sample = function() {  
  n = dim(law)[1]  
  ind = sample(n, replace = TRUE)  
  return(cor(law[ind,]$LSAT, law[ind,]$GPA))  
}  
B = 40000  
theta_star = replicate(B, draw_bootstrap_sample())
```

Law Schools Example

- Evaluate the correlation coefficient using a Monte Carlo simulation:



Law Schools Example

- Create matrix of all

$$\binom{2n-1}{n-1}$$

enumerations

- Using R package partitions:

```
n = 15
```

```
allCompositions = compositions(n,n)
```

- Each bootstrap sample has weight according to

$$\text{Multinomial} \left(\# \text{trials} = n, \text{probabilities} = \frac{1}{n}, \dots, \frac{1}{n} \right)$$

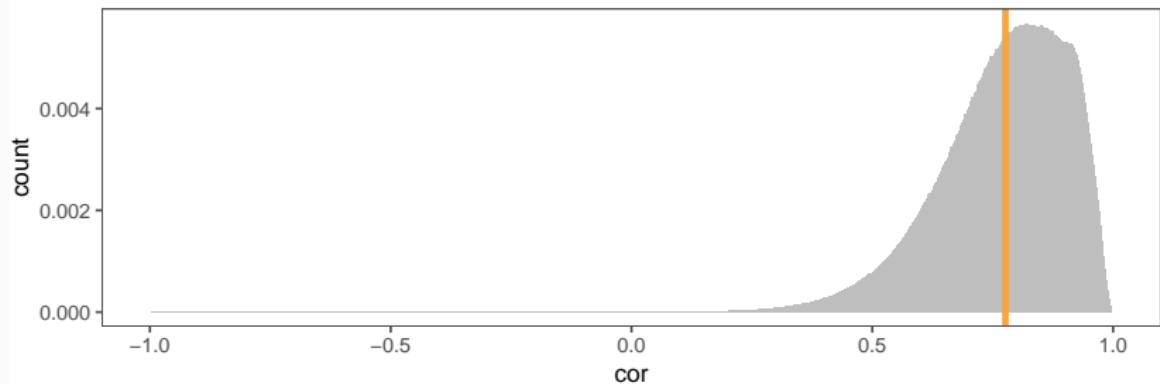
- For more details and background: Diaconis and Holmes (1994)

```
allCompositions[,1:10]
```

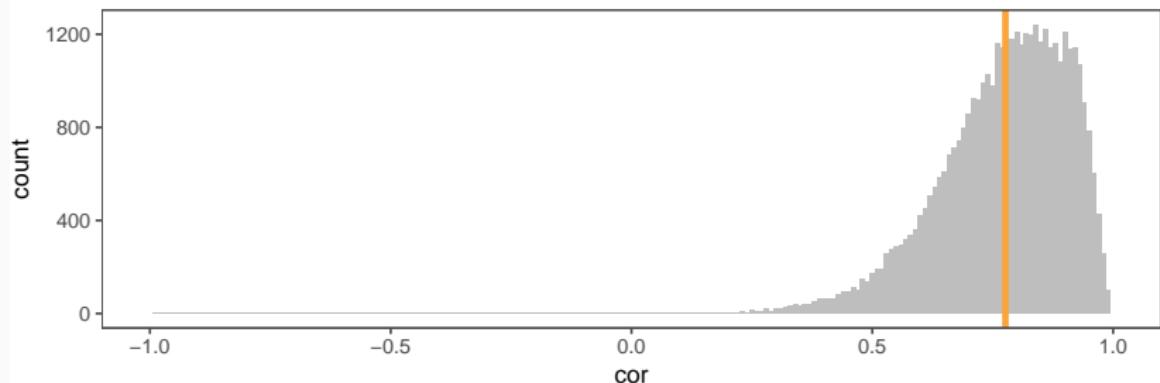
##	[,1]	[,2]	[,3]	[,4]	[,5]	[,6]	[,7]	[,8]	[,9]	[,10]
## [1,]	15	14	13	12	11	10	9	8	7	6
## [2,]	0	1	2	3	4	5	6	7	8	9
## [3,]	0	0	0	0	0	0	0	0	0	0
## [4,]	0	0	0	0	0	0	0	0	0	0
## [5,]	0	0	0	0	0	0	0	0	0	0
## [6,]	0	0	0	0	0	0	0	0	0	0
## [7,]	0	0	0	0	0	0	0	0	0	0
## [8,]	0	0	0	0	0	0	0	0	0	0
## [9,]	0	0	0	0	0	0	0	0	0	0
## [10,]	0	0	0	0	0	0	0	0	0	0
## [11,]	0	0	0	0	0	0	0	0	0	0
## [12,]	0	0	0	0	0	0	0	0	0	0
## [13,]	0	0	0	0	0	0	0	0	0	0
## [14,]	0	0	0	0	0	0	0	0	0	0
## [15,]	0	0	0	0	0	0	0	0	0	0

Law Schools Example

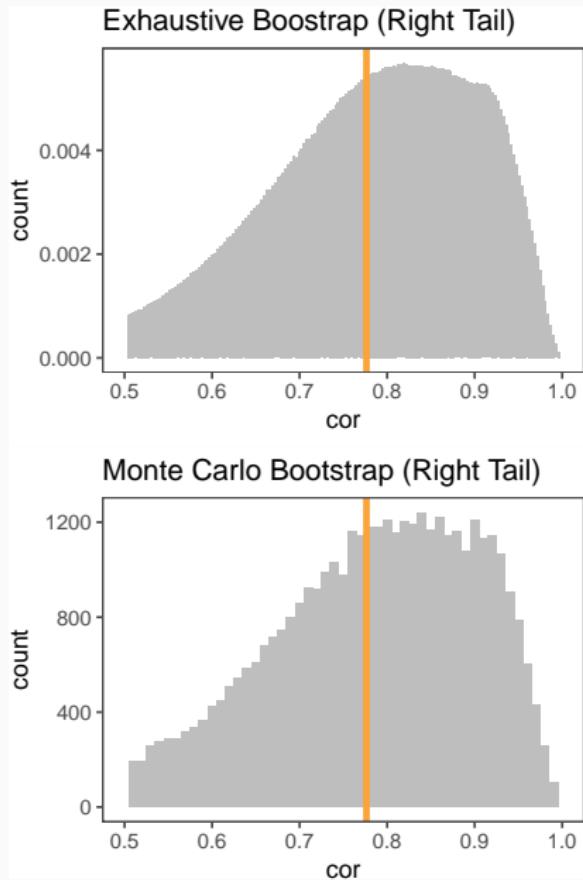
Exhaustive Bootstrap (Right Tail)



Monte Carlo Bootstrap Distribution



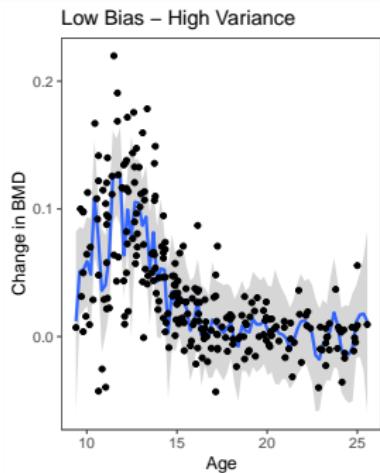
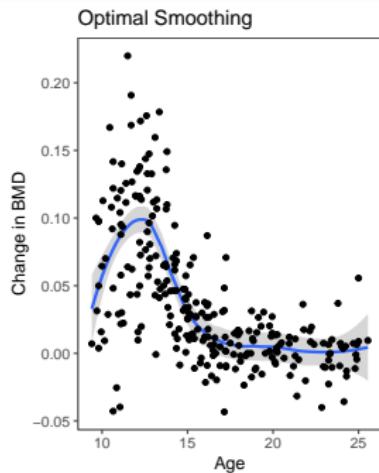
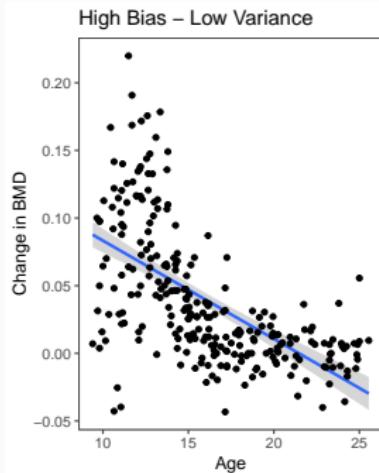
Law Schools Example



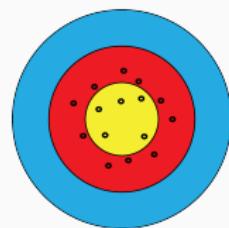
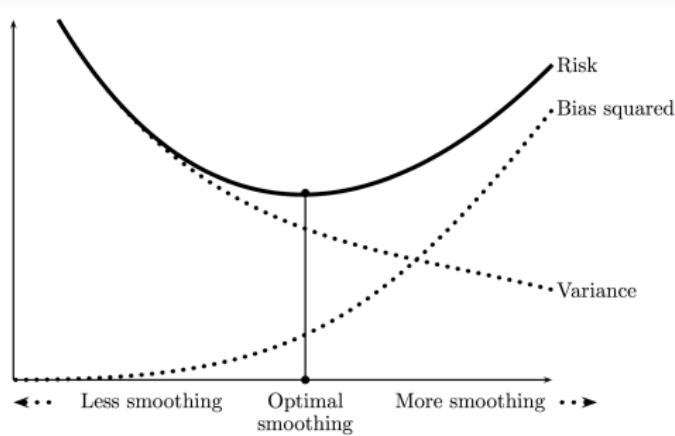
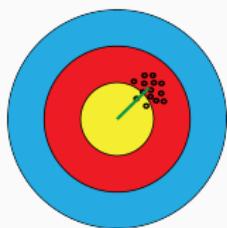
Regularized Regression

Smoothing

Bone Mineral Density Data

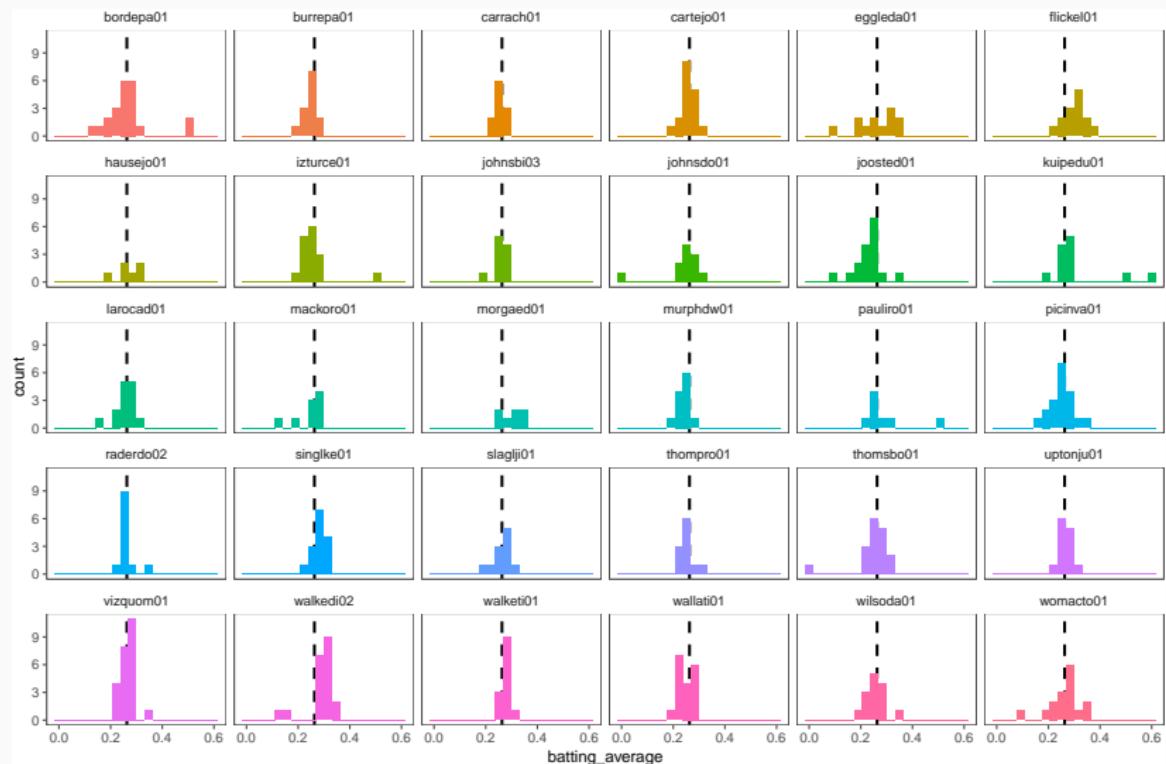


The Variance–Bias Tradeoff



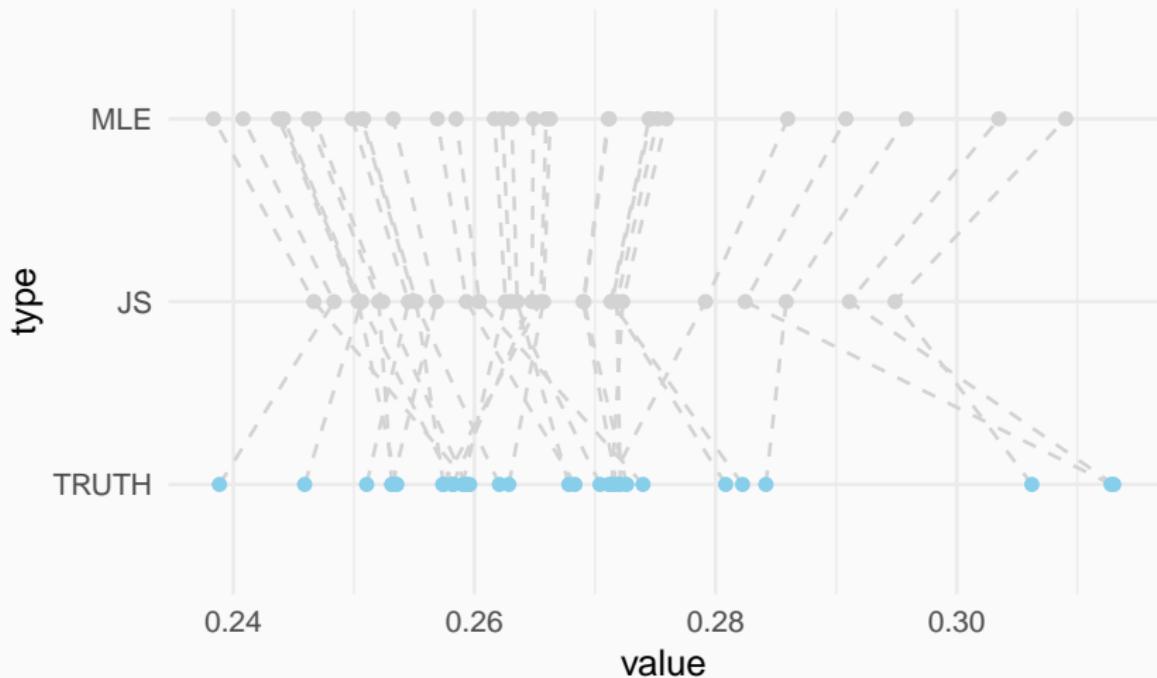
(Wasserman 2006)

Batting Average of Baseball Players



Batting Average of Baseball Players

- Here Maximum Likelihood Estimate (MLE) is the sample mean
- The James-Stein Estimator (JS) Shrinks the MLE



James-Stein Theorem

- For $d \geq 3$, the James-Stein estimator dominates the MLE in terms of expected total squared error; that is

$$E \left[\|\hat{\mu}^{\text{JS}} - \mu\|^2 \right] < E \left[\|\hat{\mu}^{\text{MLE}} - \mu\|^2 \right]$$

where $x_i | \mu_i$ is drawn from a distribution as follows

$$x_i | \mu_i \sim \text{Normal}(\mu_i, 1).$$

James-Stein Theorem

- For $d \geq 3$, the James-Stein estimator dominates the MLE in terms of expected total squared error; that is

$$E \left[\|\hat{\mu}^{\text{JS}} - \mu\|^2 \right] < E \left[\|\hat{\mu}^{\text{MLE}} - \mu\|^2 \right]$$

where $x_i | \mu_i$ is drawn from a distribution as follows

$$x_i | \mu_i \sim \text{Normal}(\mu_i, 1).$$

- In our baseball example:

$$3.0 \times 10^{-3} < 4.4 \times 10^{-3}$$

Thus a 31% improvement!

James-Stein Theorem

- For $d \geq 3$, the James-Stein estimator dominates the MLE in terms of expected total squared error; that is

$$E \left[\|\hat{\mu}^{\text{JS}} - \mu\|^2 \right] < E \left[\|\hat{\mu}^{\text{MLE}} - \mu\|^2 \right]$$

where $x_i | \mu_i$ is drawn from a distribution as follows

$$x_i | \mu_i \sim \text{Normal}(\mu_i, 1).$$

- In our baseball example:

$$3.0 \times 10^{-3} < 4.4 \times 10^{-3}$$

Thus a 31% improvement!

- For more R simulations:

<https://bookdown.org/content/922/james-stein.html>

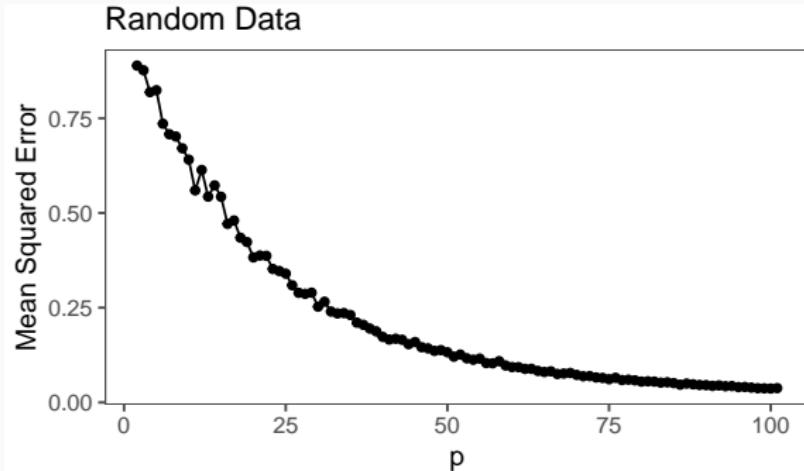
Ridge Regression

- Linear model

$$\mathbf{y} = \mathbf{X}\boldsymbol{\beta} + \boldsymbol{\epsilon}$$

- We observe or define \mathbf{y} and \mathbf{X}
- Goal: Estimate $\hat{\boldsymbol{\beta}}$
- Idea: **shrink the coefficients $\hat{\boldsymbol{\beta}}$ to zero** (similarly to the Baseball example where we shrank the individual batting averages)
- The amount of shrinkage is controlled by a **tuning parameter λ** (thus estimate depends on it: $\hat{\boldsymbol{\beta}}_{\lambda}$)

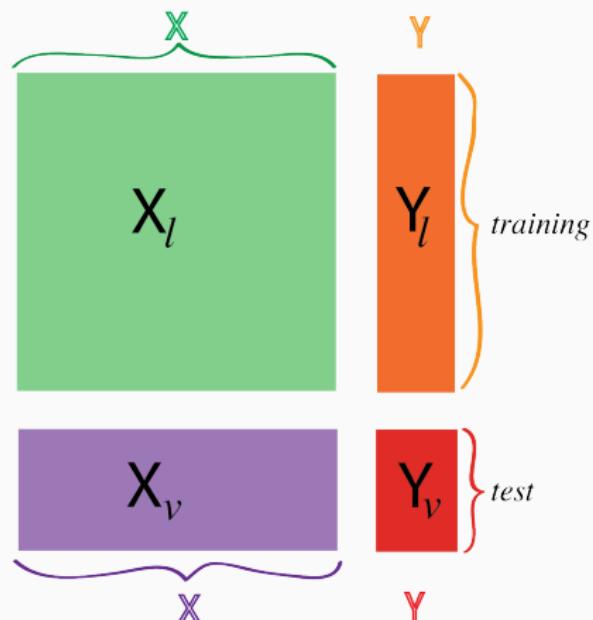
Ridge Regression: Computer Experiment



- Simulation setup:
 - Tuning parameter $\lambda = 1$
 - Number of observations $n = 20$
 - Let number of predictors p grow from 2 to 101
 - Both \mathbf{y} and \mathbf{X} are random (no relationship)

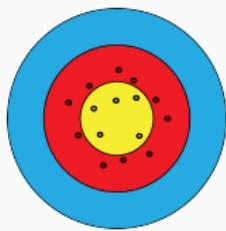
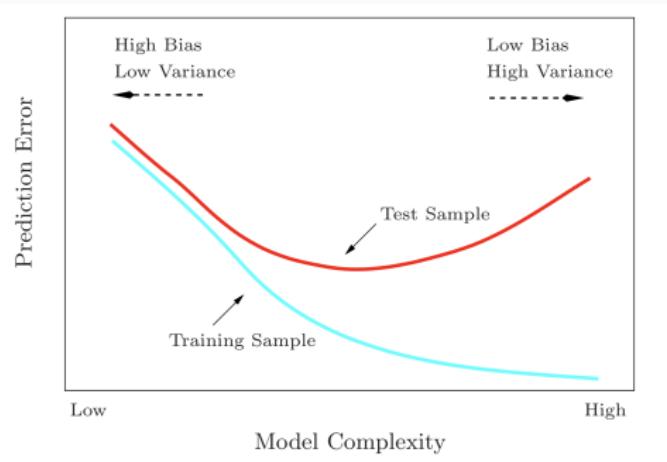
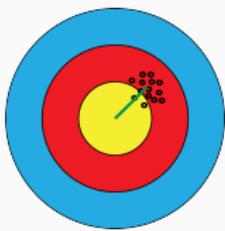
Cross-Validation

- **Problem:** model performance evaluated on training data
- **Solution:** train and evaluate on different data



(Holmes and Huber 2019)

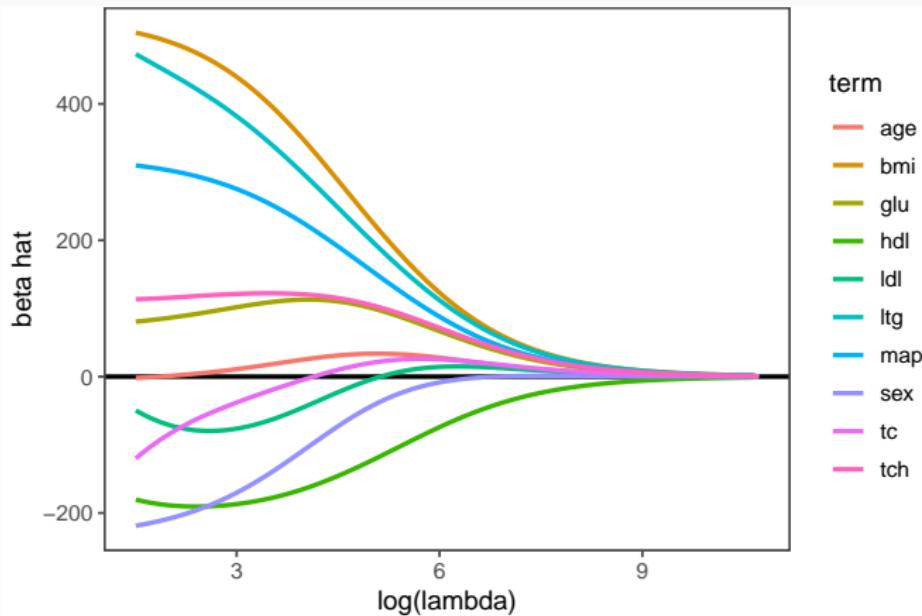
The Variance–Bias Tradeoff



(Hastie, Tibshirani, and Friedman 2009)

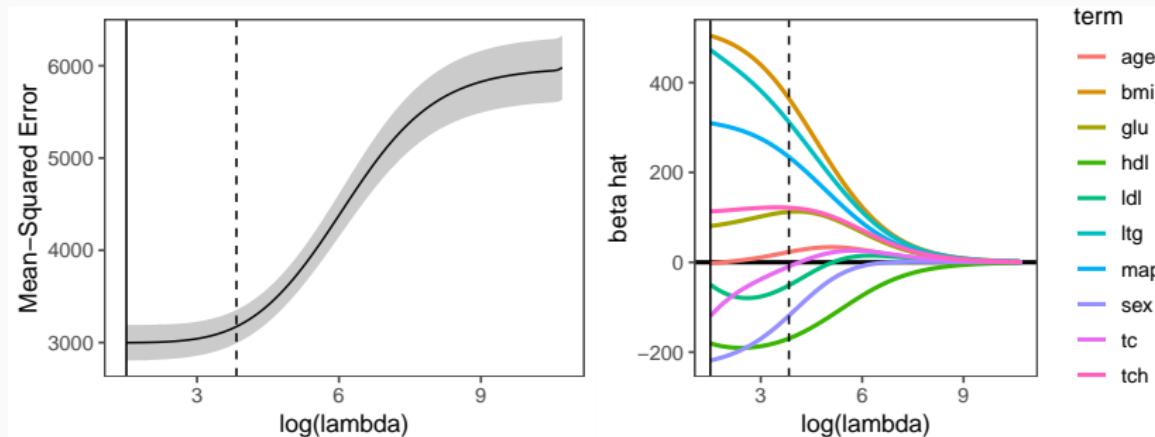
Ridge Regression: Example

- How to pick λ ?



Ridge Regression: Example

- Use cross-validation:
 1. Split data in folds
 2. Fit model on all but one fold
 3. Calculate error on the left-out fold



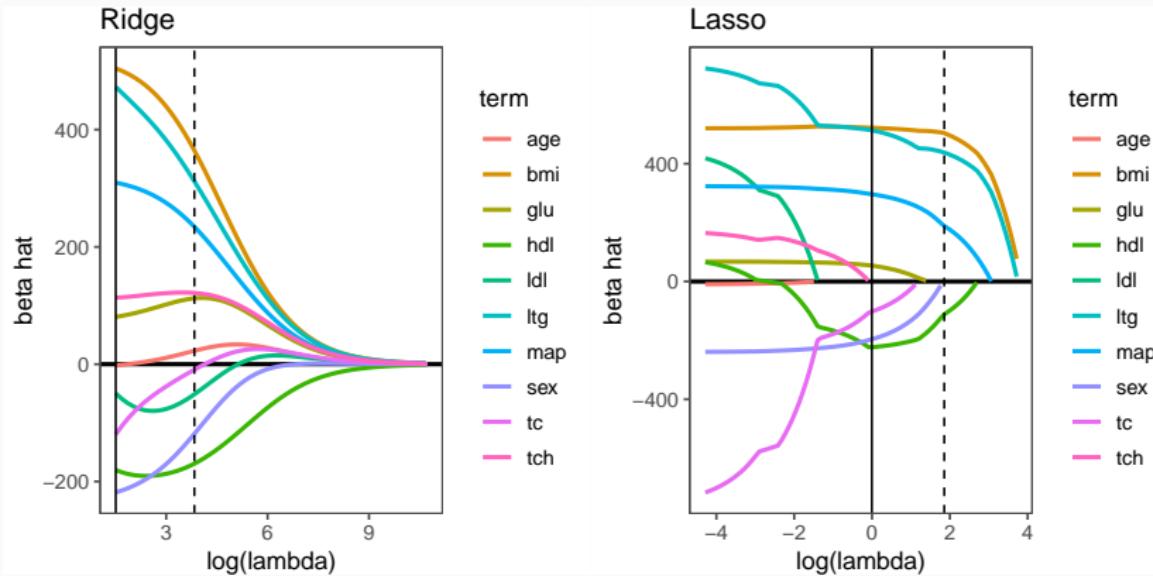
Lasso Regression

- Linear model

$$\mathbf{y} = \mathbf{X}\boldsymbol{\beta} + \boldsymbol{\epsilon}$$

- Idea: shrink the coefficients $\boldsymbol{\beta}$ to zero **and set some of them to zero completely** (similarly to the Baseball example where we shrank the individual batting averages)
- The amount of shrinkage is controlled by a **tuning parameter** λ (thus estimate depends on it: $\hat{\boldsymbol{\beta}}_\lambda$)

Lasso Regression: Example



Lab Session

- During the lab, we will get familiar with the following concepts:
 1. R and R markdown
 2. Plotting in R using package `ggplot2`
 3. Computer simulations in R using function `replicate`
 4. Construct bootstrap confidence intervals using function `quantile`
 5. Fitting regularized regression models in R using package `glmfit`
- We will use RStudio Cloud:
<https://rstudio.cloud/project/350555>

References i

- Breiman, Leo. 2001. "Statistical Modeling: The Two Cultures." *Statistical Science* 16 (3): 199–231.
- Diaconis, Persi, and Susan Holmes. 1994. "Gray Codes for Randomization Procedures." *Statistics and Computing* 4 (4): 287–302.
- Donoho, David. 2017. "50 Years of Data Science." *Journal of Computational and Graphical Statistics* 26 (4): 745–66.
- Efron, Bradley, and Trevor Hastie. 2016. *Computer Age Statistical Inference*. Vol. 5. Cambridge University Press.

References ii

- Hastie, Trevor, Robert Tibshirani, and Jerome Friedman. 2009. "The Elements of Statistical Learning: Data Mining, Inference, and Prediction." Springer New York.
- Hernán, Miguel A, John Hsu, and Brian Healy. 2019. "A Second Chance to Get Causal Inference Right: A Classification of Data Science Tasks." *CHANCE* 32 (1): 42–49.
- Hernández-Díaz, Sonia, Enrique F Schisterman, and Miguel A Hernán. 2006. "The Birth Weight 'Paradox' Uncovered?" *American Journal of Epidemiology* 164 (11): 1115–20.

- Hesterberg, Tim C. 2015. "What Teachers Should Know About the Bootstrap: Resampling in the Undergraduate Statistics Curriculum." *The American Statistician* 69 (4): 371–86.
- Holmes, Susan, and Wolfgang Huber. 2019. *Modern Statistics for Modern Biology*. Cambridge University Press.
- Wasserman, Larry. 2006. *All of Nonparametric Statistics*. Springer Science & Business Media.