LECTURE 21

Inference for Modeling

Re-visiting many of the ideas in this class with an inferential perspective.

Agenda

- Inference.
  - Estimators and their bias and variance.
- Bootstrap resampling and confidence intervals.
- The regression model.
  - We were making a set of assumptions when doing regression before.
- Bootstrapping model parameters.
  - What do estimators and bootstrapping have anything to do with regression?
  - Should tie a lot of ideas in this class together.
- Multicollinearity.

Inference

Inference?

Inferences are steps in reasoning, moving from premises to logical consequences; etymologically, the word infer means to "carry forward". Inference is theoretically traditionally divided into deduction and induction, a distinction that in Europe dates at least to Aristotle (300s BCE). Deduction is inference deriving logical conclusions from premises known or assumed to be true, with the laws of valid inference being studied in logic. Induction is inference from particular premises to a universal conclusion.

Wikipedia.

Statistical Inference?

Statistical inference is the process of using data analysis to deduce properties of an underlying distribution of probability.

Oxford Dictionary of Statistics, Upton & Cook, 2008

Statistical inference, or "learning" as it is called in computer science, is the process of using data to infer the distribution that generated the data.

All of Statistics, L. Wasserman, 2004

The goal of empirical research is--or should be--to increase our understanding of the phenomena, rather than displaying our mastery of the technique.

Statistical Models, D. Freedman, 2009

Prediction vs. inference

Prediction is the task of using our model to make predictions for the response of unseen data.

Inference is the task of using our model to draw conclusions about the underlying true relationship(s) between our features and response.

For example, suppose we are interested in studying the relationship between the value of a home and crime rates, a view of a river, school districts, size, income level of community, etc.

- **Prediction**: Given the attributes of some house, how much is it worth?
  - Care more about making accurate predictions, don't care so much about how.
- **Inference**: How much extra will a house be worth if it has a view of the river?
  - Care more about having model parameters that are interpretable and meaningful.

What is statistical inference?

- There is some fact we want to know about the population. This is called a population **parameter**.
  - Formally, a parameter is a numerical function of a population.
- Accessing the entire population is infeasible (too expensive, too time-consuming), so we collect a random sample of the population.
- We can compute a **statistic** of the random sample.
  - A statistic is a numerical function of a sample.
- However, that sample could have come out differently.
  - For example: when we estimate the population mean given the sample mean, our guess is almost always going to be somewhat wrong.
- **Inference is all about drawing conclusions about population parameters, given only a random sample**.

Terminology

Useful terminology:

- **Parameter:** Some function of a population (or “data generating process”).
  - Denoted with θ^* .
  - Example: Population mean.
- **Estimator:** Some function of a sample, whose goal is to estimate a population parameter.
  - Denoted with $\hat{\theta}$.
  - Remember, random variables were functions of random samples.
  - Hence, since we sample at random, **estimators are random variables**.
  - Example: sample mean.
- **Sampling distribution:** The distribution of estimator values, across all possible samples.
  - This is unknown: we don’t have access to the population, so we don’t know what all possible samples look like.

Bias and variance of an estimator

Bias of an estimator: the difference between the estimator's expected value and the true value of the parameter being estimated.

- Zero bias (unbiased): on average, our estimate is correct.
- Non-zero bias: on average, our estimate is consistently too large / too small.

$$\mathbf{Bias}[\hat{\theta}, \theta^*] = E[\hat{\theta}] - \theta^*$$

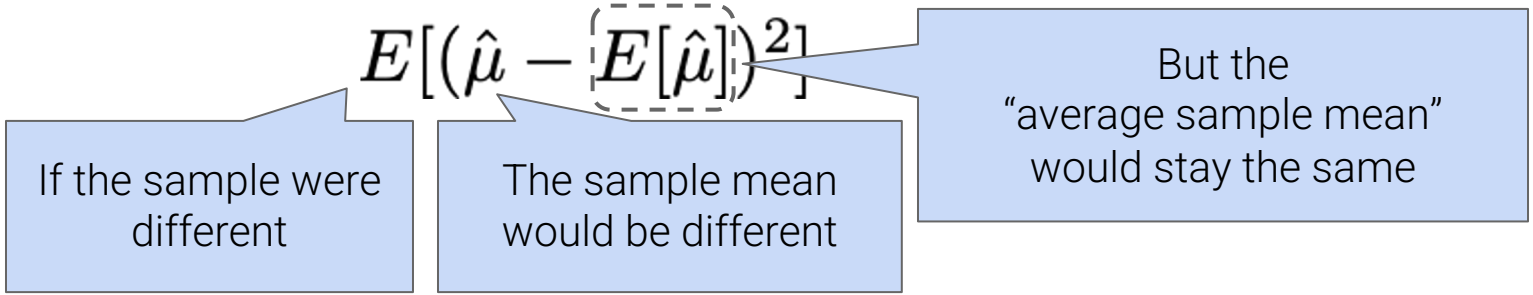
Variance of an estimator: the expected squared deviation of an estimator from its mean.

- The larger the variance of an estimator is, the more it varies from its own average.

$$\mathbf{Var}[\hat{\theta}] = E[(\hat{\theta} - E[\hat{\theta}])^2]$$

Example: sample mean estimator

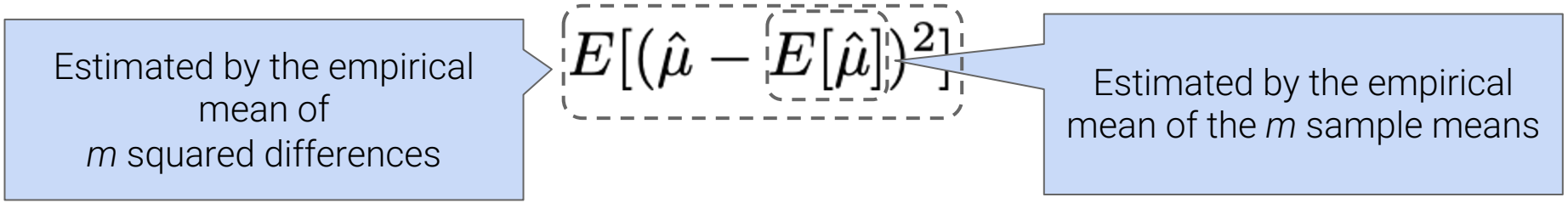
What's the variance of the **sample mean estimator**?



- Note: We use hats to denote estimates.
- Where's the population mean, μ^* ?
  - Variance isn't about the population parameter, it's about the estimator itself.
- We can't usually compute the expected value or variance of an estimator exactly!
  - We'd need access to the true population.

Example: estimating an estimator's variance

What's the variance of the **sample mean estimator**?

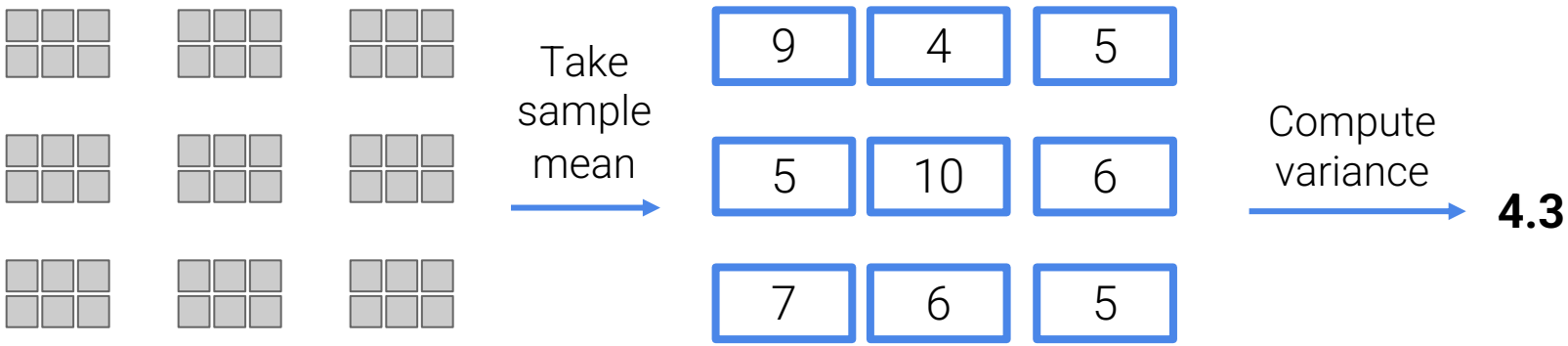


Impractical approach that would work:

- Draw m different random samples of size n from the population.
- For each sample, apply the estimator (i.e., compute the sample mean).
- Estimate the variance of the estimator using the empirical variance of these estimates.

Example: estimating an estimator's variance

Why is this so impractical?



(demo)

Bootstrapping

A Story

- A data scientist is using the data in a random sample to estimate an unknown parameter. She uses the sample to calculate the value of a statistic that she will use as her estimate.

- Once she has calculated the observed value of her statistic, she could just present it as her estimate. But she's a data scientist. She knows that her random sample is just one of numerous possible random samples, and thus her estimate is just one of numerous plausible estimates.

- By how much could those estimates vary? To answer this, it appears as though she needs to draw another sample from the population, and compute a new estimate based on the new sample. But she doesn't have the resources to go back to the population and draw another sample.

It looks as though the data scientist is stuck.

- Fortunately, a brilliant idea called *the bootstrap* can help her out. Since it is not feasible to generate new samples from the population, the bootstrap generates new random samples by a method called *resampling*: the new samples are drawn at random *from the original sample*.

Bootstrap resampling

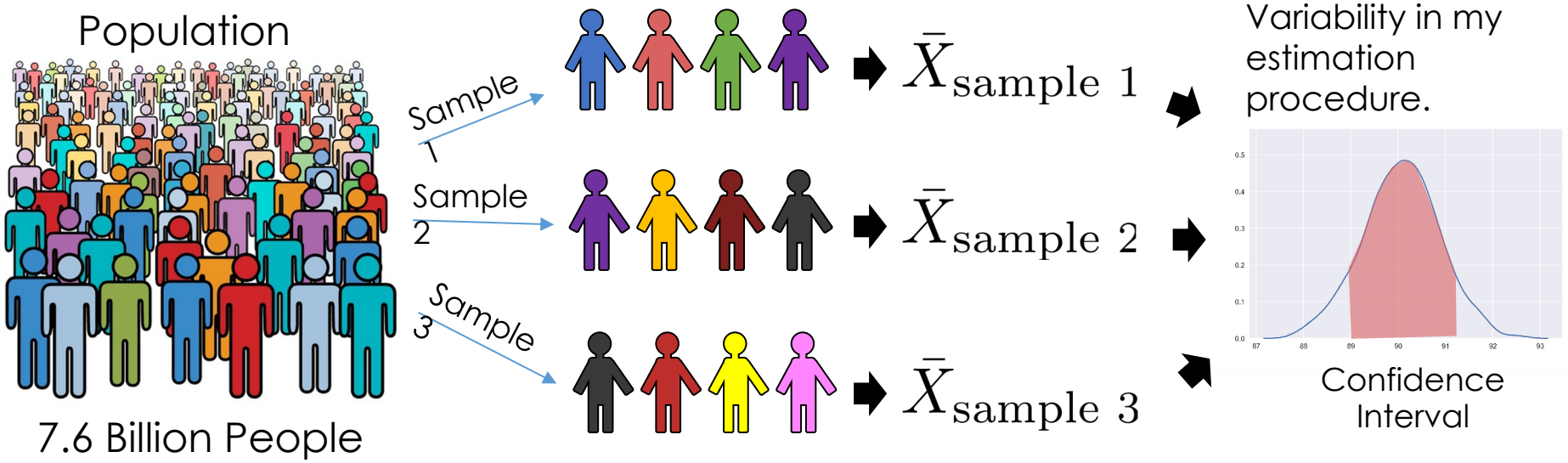
- To determine the properties (e.g. variance) of the sampling distribution of an estimator, we'd need to have access to the population.
    - We would have to consider all possible samples, and compute an estimate for each sample.
- But we don't, we only have one random sample from the population.

Idea: Treat our random sample as a "population", and resample from it.

- Intuition: a random sample resembles the population, so a random resample resembles a random sample.

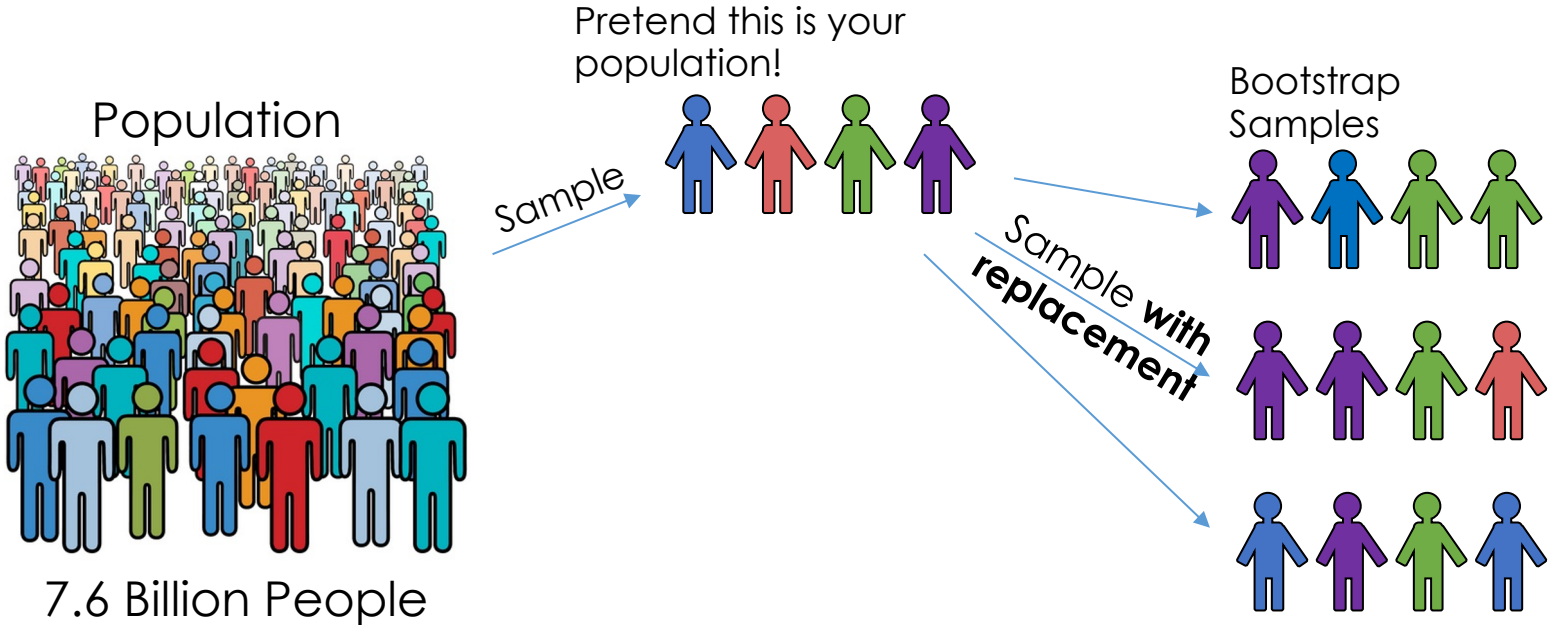
The Distribution of an Estimator

Resampling the population to estimate the sample distribution.



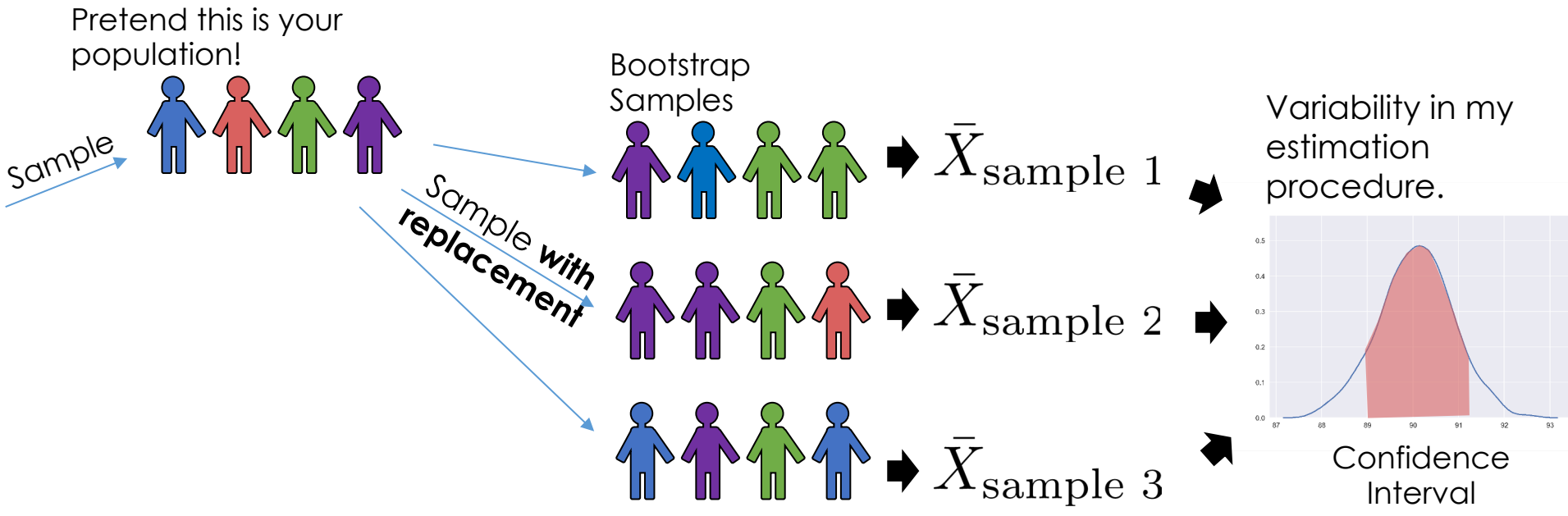
Bootstrap the Distribution of an Estimator

Simulation method to estimate the sample distribution.



Bootstrap the Distribution of an Estimator

Simulation method to estimate the sample distribution.



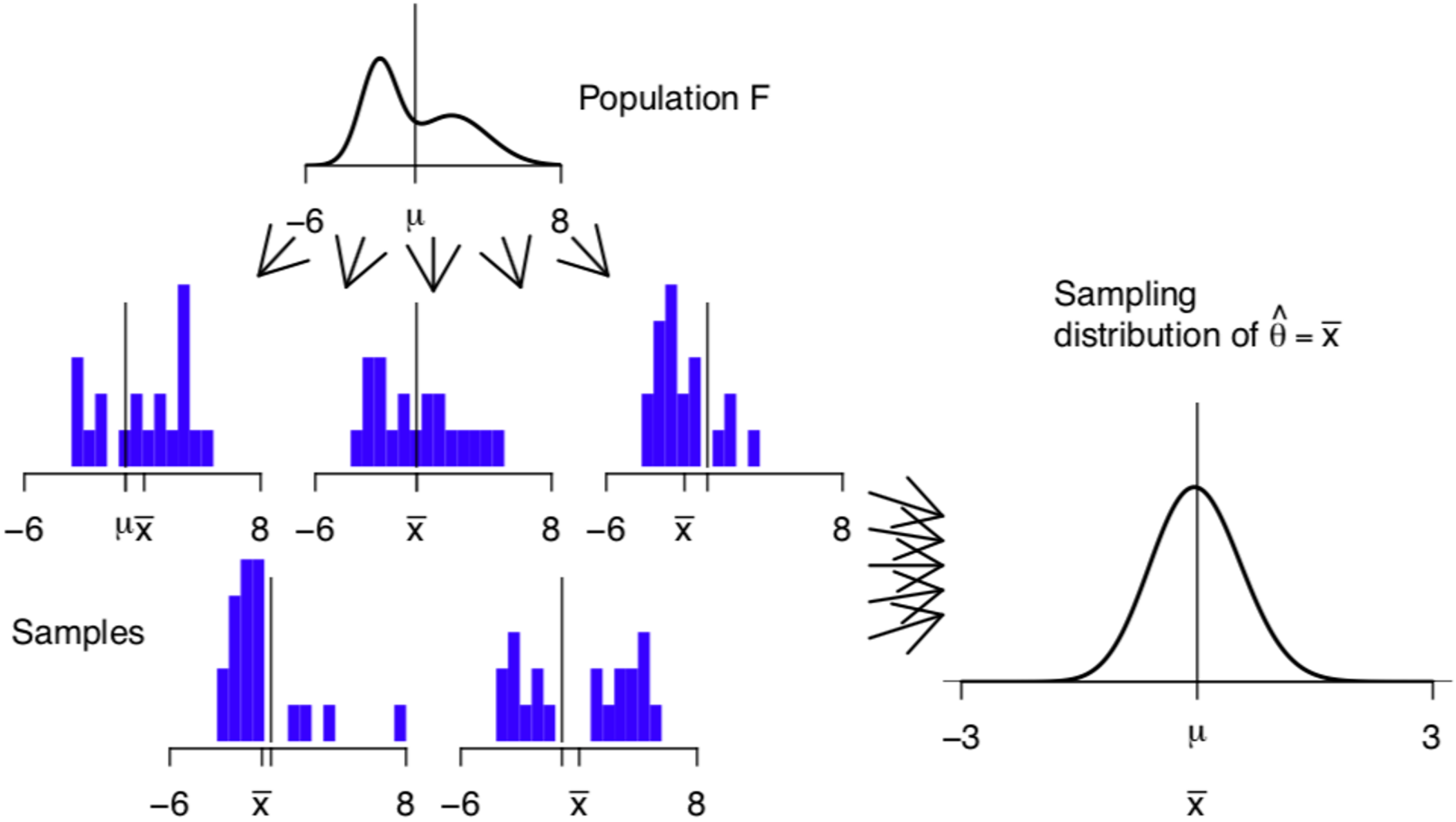
Bootstrapping pseudocode

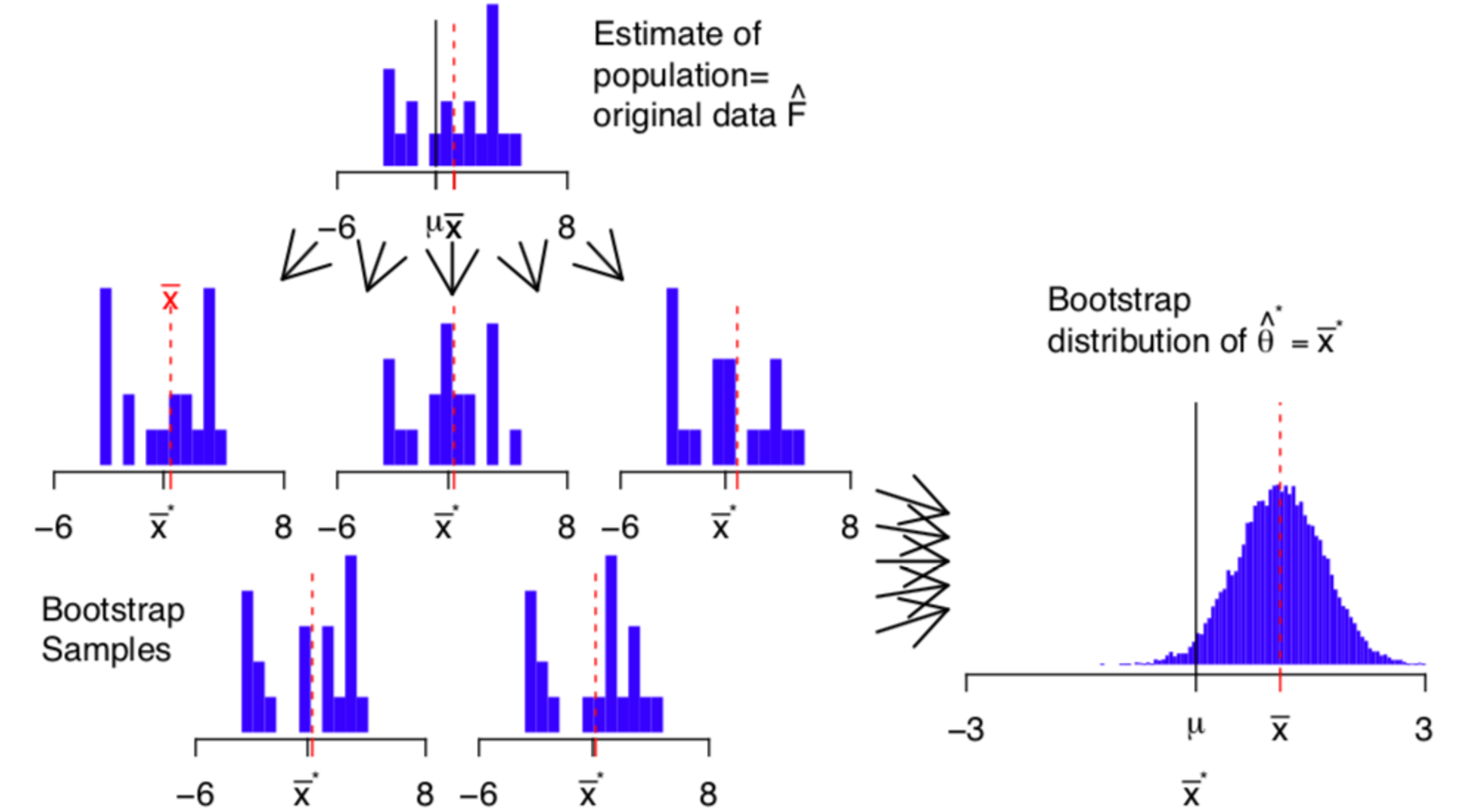
```
collect random sample of size n (called the bootstrap population)
initiate list of estimates
repeat 10,000 times:
        resample with replacement n times from bootstrap population
    apply estimator f to resample
    store in list
list of estimates is the bootstrapped sampling distribution of f
```

Why **must** we resample **with replacement**?

Bootstrap discussion

- The **bootstrapped sampling distribution of an estimator** does not exactly match the **sampling distribution of that estimator**.
  - The center and spread are both wrong (but often close).
- The center of the bootstrapped distribution is the estimator applied to our original sample.
  - We have no way of recovering the estimator's true expected value.
- The variance of the bootstrapped distribution is often close to the true variance of the estimator.
- The quality of our bootstrapped distribution depends on the quality of our original sample.
  - If our original sample was not representative of the population, bootstrap is next to useless.

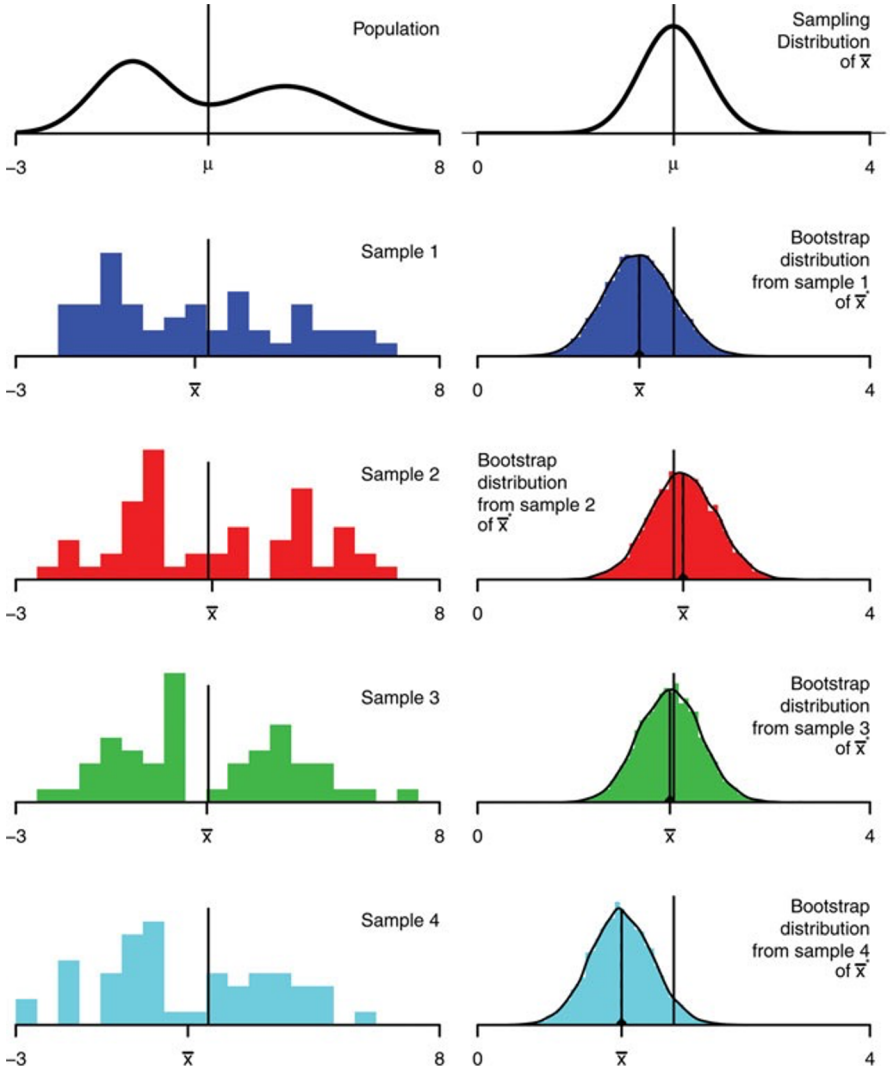

Population F
−6
μ
8
Samples
−6
μ
x̄
8
−6
x̄
8
−6
x̄
8
−6
x̄
8
−6
x̄
8
Sampling
distribution of θ̂ = x̄
−3
μ
3
x̄


Estimate of
population=
original data $\hat{F}$
−6
$\mu_{\bar{x}}$
8
$\bar{x}$
−6
$\bar{x}^*$
8
−6
$\bar{x}^*$
8
−6
$\bar{x}^*$
8
Bootstrap
Samples
−6
$\bar{x}^*$
8
−6
$\bar{x}^*$
8
Bootstrap
distribution of $\hat{\theta}^* = \bar{x}^*$
−3
μ
$\bar{x}$
3
$\bar{x}^*$

What Teachers Should Know About the Bootstrap: Resampling in the Undergraduate Statistics Curriculum

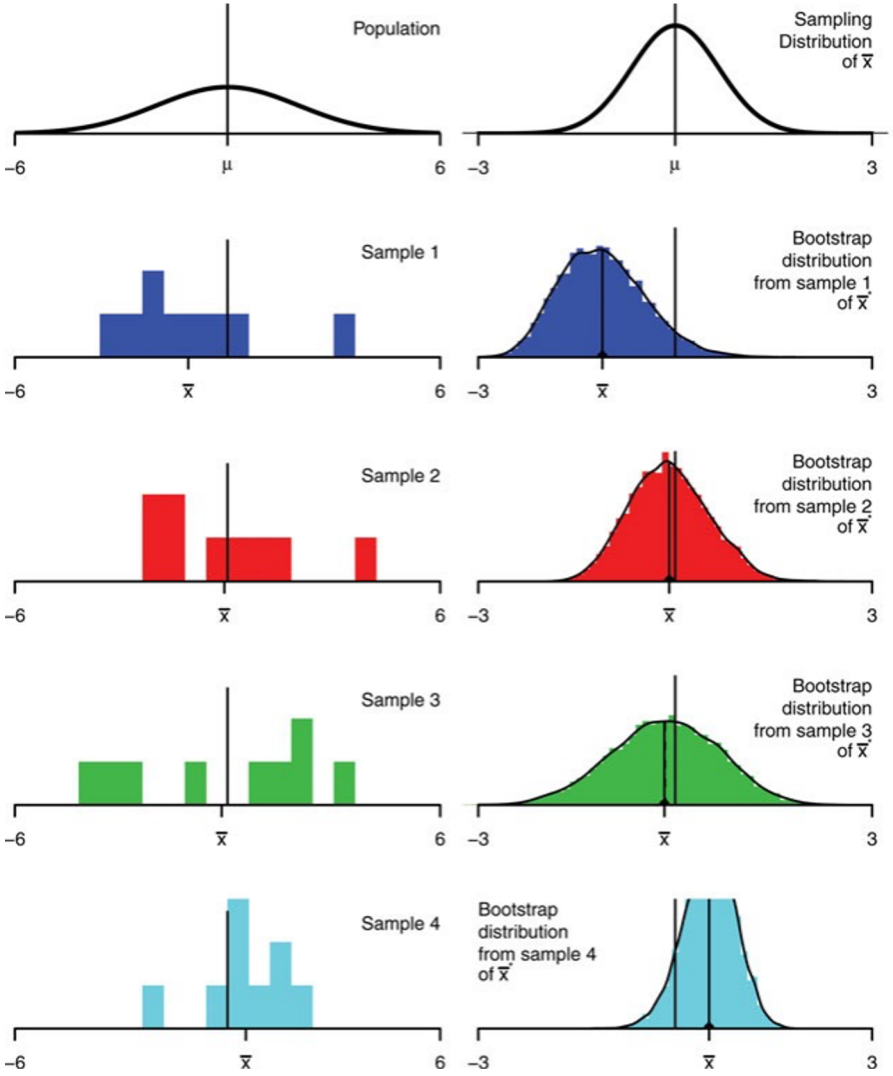
Tim C. Hesterberg (2015)

- The bootstrap is based on the *plug-in principle*—if something is unknown, we substitute an estimate for it.
- Instead of plugging in an estimate for a single parameter, we plug in an estimate for the whole population.
- *The bootstrap distribution is centered at the observed statistic, not the population parameter*, for example, at $\bar{x}$ not μ.
- For example, we cannot use the bootstrap to improve on $\bar{x}$; no matter how many bootstrap samples we take, they are centered at $\bar{x}$, not μ. Instead we use the bootstrap to tell how accurate the original estimate is.

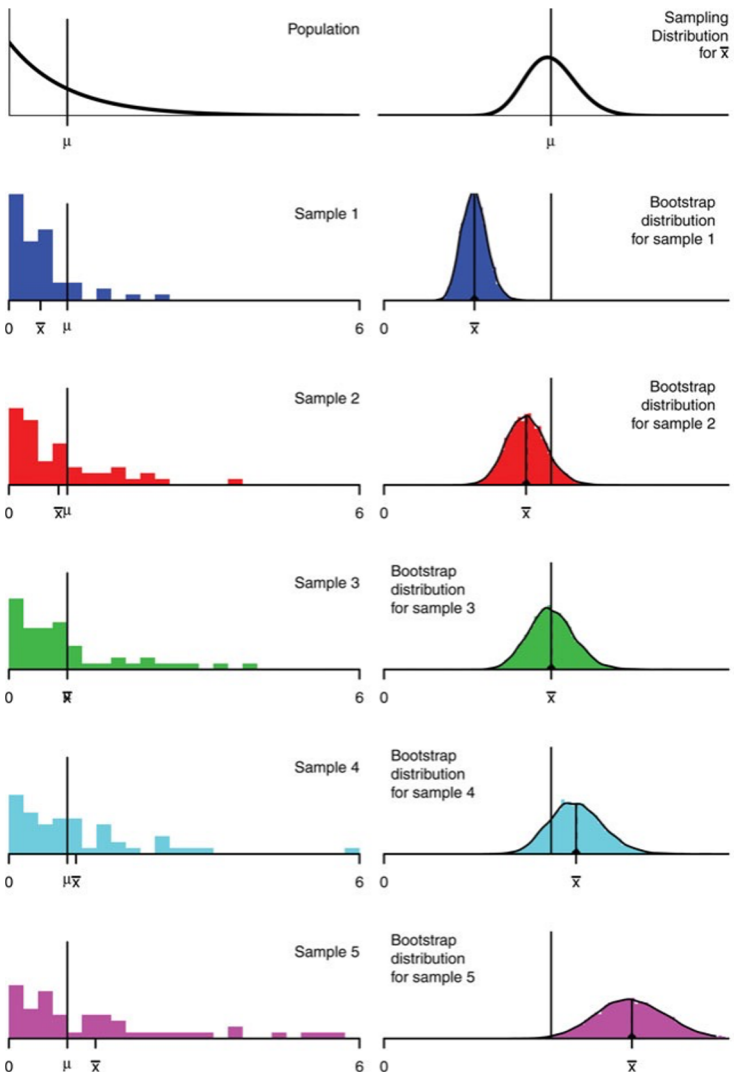


Bootstrap for the mean, n=50

From Tim C. Hesterberg (2015)



Bootstrap distributions for the mean, n = 9



Bootstrap distributions for the mean, n = 50, exponential population.

Some more lessons from Hesterberg

- The ordinary bootstrap tends not to work well for statistics such as the median or other quantiles in small samples that depend heavily on a small number of observations out of a larger sample. The bootstrap depends on the sample accurately reflecting what matters about the population, and those few observations cannot do that.
- *Bootstrapping does not overcome the weakness of small samples as a basis for inference*. Indeed, for the very smallest samples, it may be better to make additional assumptions such as a parametric family.

Bootstrap confidence intervals

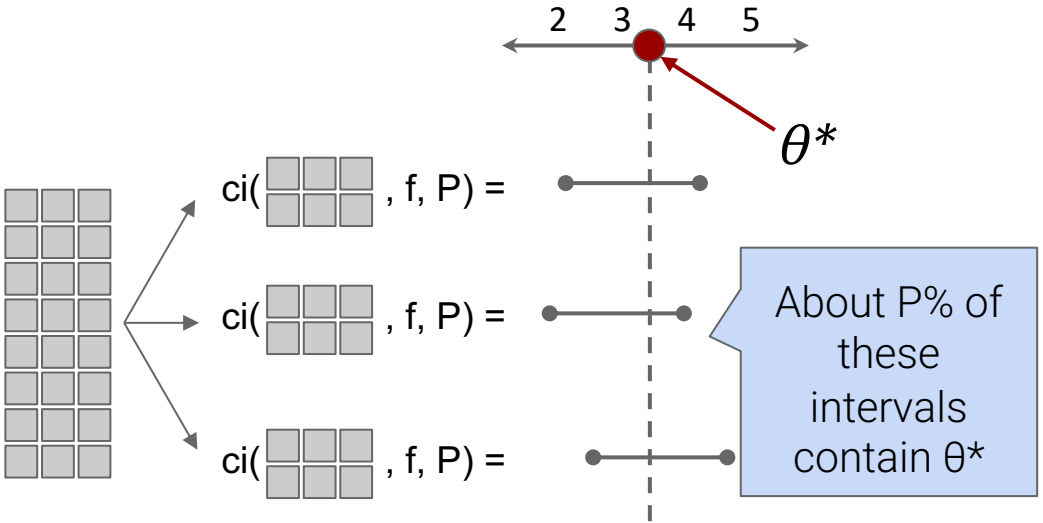
Confidence intervals

- Intuition: Estimate an interval where we think the population parameter is, based on the center and variance of the estimator.
- **What does a P% confidence interval mean?**
  - Imagine the following procedure:
    - Take a sample from the population.
    - Compute P% confidence interval for the true population parameter, **somehow**.
  - If we repeat this procedure many times, the population parameter will be in our interval P% of the time, in the long run.

Confidence intervals

An estimator *f* exists in order to guess the value of an unknown parameter θ^*.

An estimator *ci* for a P% confidence interval for *f* is a function that takes a sample and returns an interval. This interval will (ideally) contain θ^* for P% of samples.

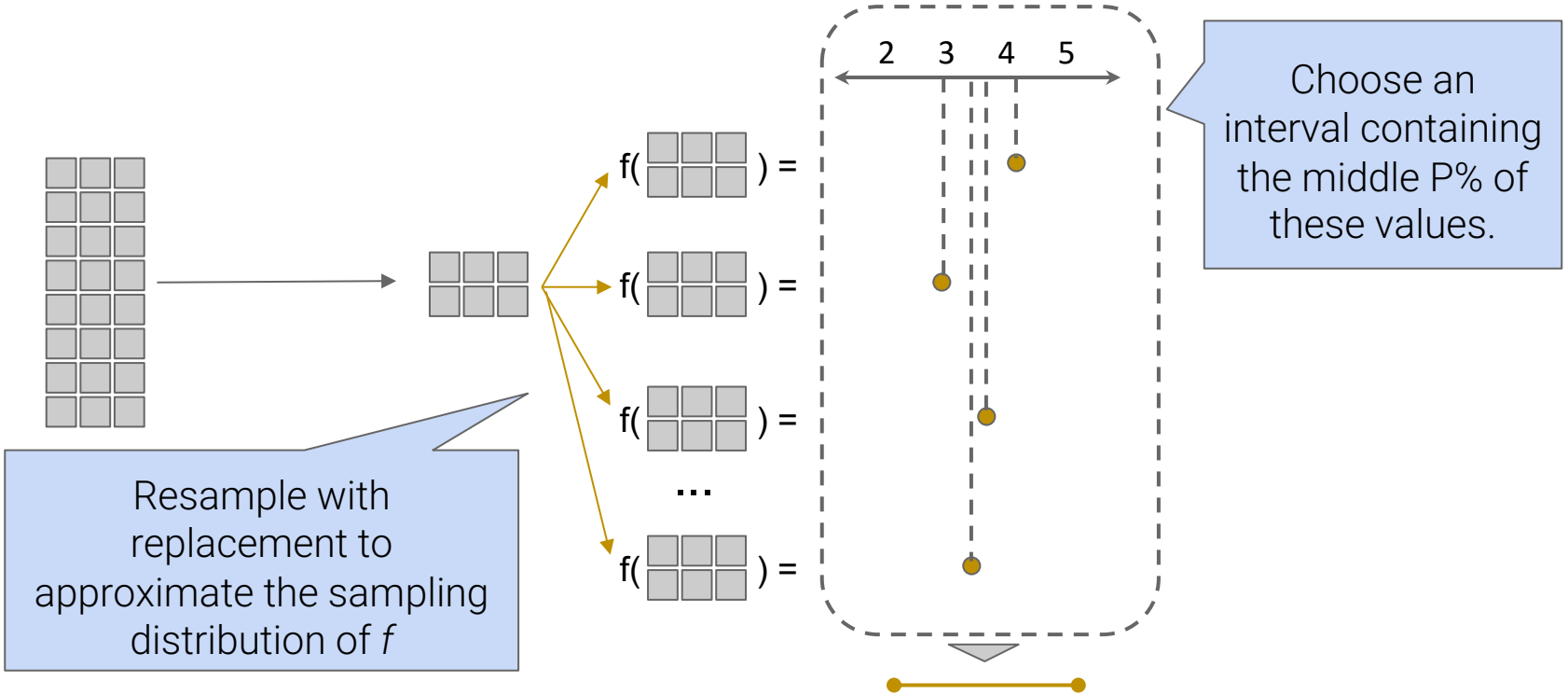


How do we compute `ci(s,f,P)`?

- Approximate the sampling distribution of *f* using the sample *s*.
- Choose the middle P% of samples from this approximate distribution.

Bootstrap confidence intervals

An estimator *ci* for a P% confidence interval for *f* is a function that takes a sample and returns an interval. This interval will (ideally) contain θ^* for P% of samples.



(Demo)

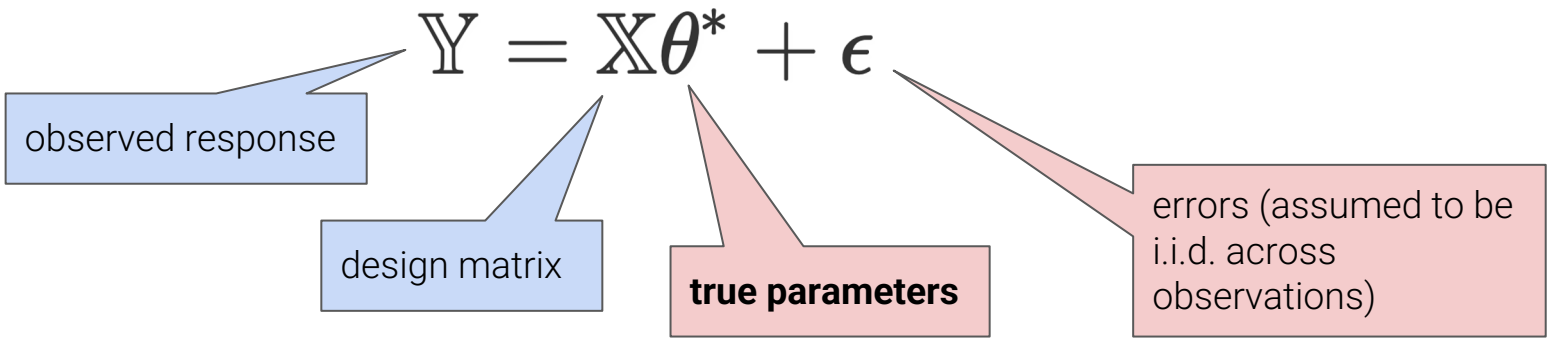
Confidence intervals

- The confidence level is a statement about the procedure used to create our interval.
- It is **not** the case that a 95% confidence interval means “there is a 95% chance that the population parameter is in our interval”.
  - The population parameter is fixed.
  - Our interval is fixed. The population parameter is either in it, or it isn’t.
  - Nothing random here!
- The confidence intervals we’ve created are sometimes called **percentile bootstrap confidence intervals**.
  - There are other methods of creating confidence intervals.
    - E.g. assume that the sampling distribution of your estimator is normal.

The regression model

The regression model

- When fitting a linear regression model, we are assuming there is some underlying true relationship between our features and response.
- But, **we never get to see the true relationship.**
- We instead see a noisy version of it.



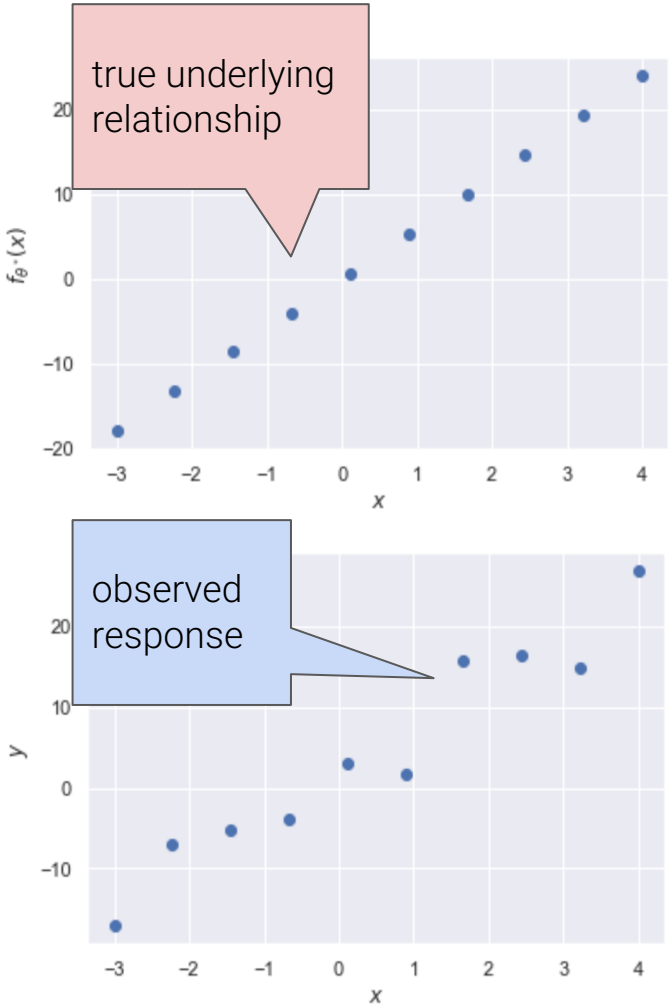
Example: simple linear regression

$$\hat{y}_i = \underbrace{\theta_0^* + \theta_1^* x_i}_{f_{\theta^*}(x_i)} + \epsilon_i$$

$f_{\theta^*}(x_i)$: true linear relationship

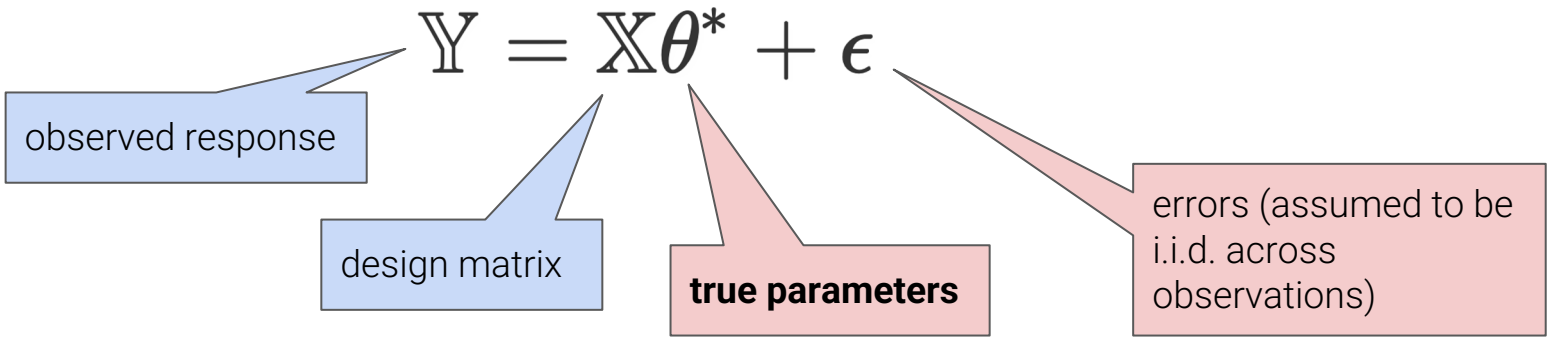
ϵ_i: random noise

$$E[\epsilon_i] = 0, \mathrm{Var}[\epsilon_i] = \sigma^2$$



The regression model

- When fitting a linear regression model, we are assuming there is some underlying true relationship between our features and response.
- But, **we never get to see the true relationship.**
- We instead see a noisy version of it.



We can **observe** the quantities in **blue**. The quantities in **red** are **unobservable**.

Our goal is to **estimate** θ^* .

Least squares estimation

- We called $\hat{\theta}$ the optimal model parameter, because it minimized MSE for our training data.
- $\hat{\theta}$ is an estimate of θ^*. Specifically, it is the one that minimizes the training MSE (or some regularized version of it).

$$\hat{\theta} = \left(\mathbb{X}^T\mathbb{X}\right)^{-1}\mathbb{X}^T\mathbb{Y}$$

Estimator! Takes in a sample, returns an estimate for a population parameter.

- This is nothing new – just a more rigorous statistical treatment of what we've already seen.
- Still make predictions as $\hat{\mathbb{Y}} = \mathbb{X}\hat{\theta}$.

Bootstrapping model parameters

Parameter estimates

- Our estimate for θ^* depends on what our training data was.
  - Different training data, different $\hat{\theta}$!
- We want to think about all of the different ways that our training data, and hence our parameter estimate, could have come out.
- Easy!
  - Bootstrap our training data.
  - Fit a linear model to each resample.
  - Look at the resulting distribution of bootstrapped parameter estimates.

$$\hat{\theta} = \left(\mathbb{X}^T\mathbb{X}\right)^{-1}\mathbb{X}^T\mathbb{Y}$$

Assessing the quality of our model

- Suppose we fit a linear regression model with p features, plus an intercept term.

$$y = f_{\theta^*}(x) + \epsilon = \theta_0^* + \sum_{j=1}^{p} \theta_j^* x_j + \epsilon$$

assumed underlying model

$$\hat{y} = f_{\hat{\theta}}(x) = \hat{\theta}_0 + \sum_{j=1}^{p} \hat{\theta}_j x_j$$

how we make predictions

- If the true θ_1^* is 0, then the feature x_1 has no effect on the response.
- How can we test whether or not $\theta_1^* = 0$?

Confidence interval for true slope

- We want to test whether θ_1^* is 0.
- We get one estimate $\hat{\theta}_1$ from our sample.
- But we must imagine all the other ways the random sample could have come out.
- If the sample is large – **bootstrap it!**
  - Estimate θ_1^* each time.
  - Make a confidence interval for θ_1^* and see if 0 is in the interval.
    - If yes: θ_1^* is not significantly different than 0.
    - If no: θ_1^* is significantly different than 0.
    - Can formalize with the language of hypothesis testing.
  - Works for linear (and logistic!) regression models with any number of features.

(demo)

Multicollinearity

The meaning of “slope”

Consider the equation

$$y = a_0 + a_1x_1 + a_2x_2 + \ldots + a_px_p$$

- The slope a_1 measures the change in y per unit change in x_1, **assuming all other variables are held constant.**
- We use an equation of the above form for linear regression.
- But what if we can’t hold all other variables constant?

Multicollinearity

- If features are related to each other, it might not be possible to have a change in one of them *while holding the others constant*.
  - Then, the individual slopes will have no meaning.
- **Multicollinearity:** when a feature can be predicted **fairly accurately** by a linear combination of other features.
  - Slopes can't be interpreted.
  - Small changes in the data can lead to big changes in the slopes.
  - Doesn't impact the predictive capability of our model – only impacts interpretability.
- **Perfect** multicollinearity: one feature can be written **exactly** as a linear combination of other features.
  - Design matrix isn't full rank! Can't find unique $\hat{\theta}$.
    - For instance, one-hot encoding with an intercept term.

Case study in multicollinearity

(demo)

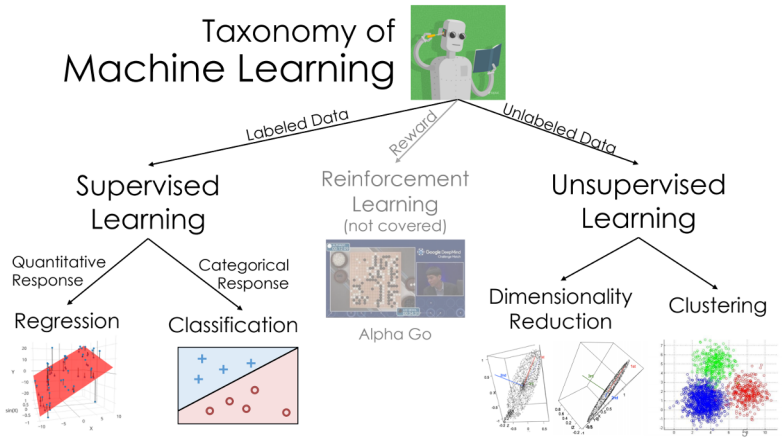
Summary

Summary

- Estimators are functions that provide estimates of true population parameters.
- We can bootstrap to estimate the sampling distribution of an estimator.
- Using this bootstrapped sampling distribution, we can compute a confidence interval for our estimator.
    - This gives us a rough idea of how uncertain we are about the true population parameter.
    - Only valid if the original random sample is representative.
- The assumption when performing linear regression is that there is some true parameter theta that defines a linear relationship between features X and response y.
    - We can use the bootstrap to determine whether or not an individual feature is significant.
- Multicollinearity arises when features are correlated with one another.

What's next

- This lecture was a (brief) diversion from the "ML" perspective.
  - ML: Make accurate predictions.
  - Statistics: Infer about a population.
- So far, we've covered **supervised learning** in great detail.
  - Linear regression, logistic regression, decision trees / random forests.
  - SVM, Boosting (On the way)
- We spend some time talking about **unsupervised learning.**
  - Dimensionality reduction, PCA.
  - Clustering.



from Joseph Gonzalez