Lecture 11: Frequentist Regression and Bootstrap

Jacob Steinhardt

October 1, 2020

Recap

- Bayesian regression
  - Least squares = MLE
  - Ridge regression = MAP
- Overdispersion
  - Model mis-specification $\implies$ overly narrow uncertainty

Recap

- Bayesian regression
  - Least squares = MLE
  - Ridge regression = MAP
- Overdispersion
  - Model mis-specification $\implies$ overly narrow uncertainty

This time: frequentist uncertainty and bootstrap

Bayesian vs. frequentist uncertainty

Credible interval:

Confidence interval:

Bayesian vs. frequentist uncertainty

Credible interval: Posterior probability that θ lies in interval is at least *p*

Confidence interval: Conditional on θ, interval contains θ with probability *p*

Confidence/credible intervals for regression

Recall COVID-19 example: $\mathbb{E}[\text{Cases} \mid \text{Day}] = \exp(\beta_{\text{Day}} \cdot \text{Day} + \beta_{\text{Intercept}})]$

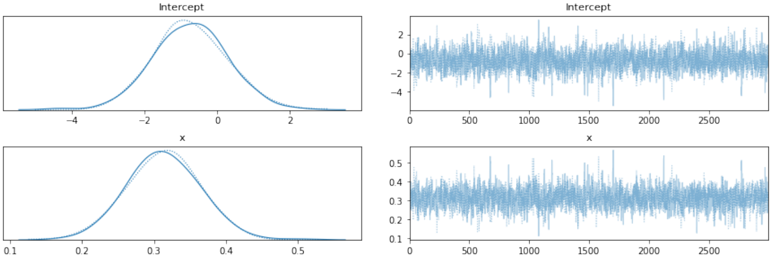
To understand growth rate, care about coefficient β_{Day}

Confidence/credible intervals for regression

Recall COVID-19 example: $\mathbb{E}[\text{Cases} \mid \text{Day}] = \exp(\beta_{\text{Day}} \cdot \text{Day} + \beta_{\text{Intercept}})]$

To understand growth rate, care about coefficient β_{Day}

Previous lecture: MCMC sampling gives us posterior distribution (and hence credible interval) for β_{Day}:



Confidence/credible intervals for regression

Recall COVID-19 example: $\mathbb{E}[\text{Cases} \mid \text{Day}] = \exp(\beta_{\text{Day}} \cdot \text{Day} + \beta_{\text{Intercept}})]$

To understand growth rate, care about coefficient β_{Day}

What about confidence interval? Can't use prior.

Confidence/credible intervals for regression

Recall COVID-19 example: $\mathbb{E}[\text{Cases} \mid \text{Day}] = \exp(\beta_{\text{Day}} \cdot \text{Day} + \beta_{\text{Intercept}})]$

To understand growth rate, care about coefficient β_{Day}

What about confidence interval? Can't use prior.

General recipe: use generalization of CLT called "asymptotic normality"

Confidence/credible intervals for regression

Recall COVID-19 example: $\mathbb{E}[\text{Cases} \mid \text{Day}] = \exp(\beta_{\text{Day}} \cdot \text{Day} + \beta_{\text{Intercept}})]$

To understand growth rate, care about coefficient β_{Day}

What about confidence interval? Can't use prior.

General recipe: use generalization of CLT called "asymptotic normality"

Beyond scope of this class, but `statsmodels` package will do it for us!

Confidence intervals with statsmodels

[Jupyter demo]

Escaping model mis-specification

Frequentist confidence intervals can be wrong if model is wrong

- Just like Bayesian case

We'll escape this with a **non-parametric** tool for producing frequentist CIs

Non-parametric $\implies$ doesn't rely on model $\implies$ more robust

Escaping model mis-specification

Frequentist confidence intervals can be wrong if model is wrong

- Just like Bayesian case

We'll escape this with a **non-parametric** tool for producing frequentist CIs

Non-parametric $\implies$ doesn't rely on model $\implies$ more robust

You've seen this before: the **bootstrap**

The Bootstrap

Idea for computing confidence intervals + uncertainty

Without bootstrap:

- Chi-square test, student-t test, . . .
- Lots of algebra, need different formula for each setting
- Often rely on model assumptions

With bootstrap:

- Single unified approach
- Computer simulation
- Fewer assumptions

Bootstrap: formal setting

Data: $x^{(1)}, \ldots, x^{(n)}$

Estimator: $\hat{\theta} = \hat{\theta}(x^{(1)}, \ldots, x^{(n)})$

- θ^*: population parameter (what $\hat{\theta}$ converges to as $n \to \infty$)

Question: How close is θ^* to $\hat{\theta}$?

Some concrete examples

Mean of 1-dimensional distribution:

- $x^{(1)}, \ldots, x^{(n)} \in \mathbb{R}$
- $\hat{\theta}(x^{(1)}, \ldots, x^{(n)}) = \frac{1}{n}(x_1 + \ldots + x_n)$

How close is estimate to the true mean?

Some concrete examples

Mean of 1-dimensional distribution:

- $x^{(1)}, \dots, x^{(n)} \in \mathbb{R}$
- $\hat{\theta}(x^{(1)}, \dots, x^{(n)}) = \frac{1}{n}(x_1 + \dots + x_n)$

How close is estimate to the true mean?

Regression:

- $(x^{(1)}, y^{(1)}), \dots, (x^{(n)}, y^{(n)}) \in \mathbb{R}^d \times \mathbb{R}$
- $\hat{\beta}((x^{(1)}, y^{(1)}), \dots, (x^{(n)}, y^{(n)})) = \operatorname{argmin}_\beta \sum_{i=1}^n (y^{(i)} - \beta^\top x^{(i)})^2$

How close is $\hat{\beta}$ to population parameters β^*?

More complex examples

Mixture models

Density estimation

Neural nets? (Actually not...)

The ideal hypothetical: re-sampling

Population distribution p^*

- $x^{(1)}, \ldots, x^{(n)} \sim p^*$

The ideal hypothetical: re-sampling

Population distribution p^*

- $x^{(1)}, \dots, x^{(n)} \sim p^*$

Noise in $\hat{\theta}$ due to randomness in $x^{(1)}, \dots, x^{(n)}$

The ideal hypothetical: re-sampling

Population distribution p^*

- $x^{(1)}, \dots, x^{(n)} \sim p^*$

Noise in $\hat{\theta}$ due to randomness in $x^{(1)}, \dots, x^{(n)}$

Imagine hypothetically sampling fresh data:

$$\begin{aligned}
x^{(1)}, \dots, x^{(n)} &\rightarrow \hat{\theta} \text{ (Original sample)} \\
x^{(1)\prime}, \dots, x^{(n)\prime} &\rightarrow \hat{\theta}' \text{ (Re-sample)} \\
x^{(1)\prime\prime}, \dots, x^{(n)\prime\prime} &\rightarrow \hat{\theta}'' \\
x^{(1)\prime\prime\prime}, \dots, x^{(n)\prime\prime\prime} &\rightarrow \hat{\theta}''' \\
&\vdots
\end{aligned}$$

The ideal hypothetical: re-sampling

Population distribution p^*

- $x^{(1)}, \ldots, x^{(n)} \sim p^*$

Noise in $\hat{\theta}$ due to randomness in $x^{(1)}, \ldots, x^{(n)}$

Imagine hypothetically sampling fresh data:

$$\begin{aligned} x^{(1)}, \ldots, x^{(n)} &\rightarrow \hat{\theta} \text{ (Original sample)} \\ x^{(1)\prime}, \ldots, x^{(n)\prime} &\rightarrow \hat{\theta}' \text{ (Re-sample)} \\ x^{(1)\prime\prime}, \ldots, x^{(n)\prime\prime} &\rightarrow \hat{\theta}'' \\ x^{(1)\prime\prime\prime}, \ldots, x^{(n)\prime\prime\prime} &\rightarrow \hat{\theta}''' \\ &\vdots \end{aligned}$$

Implicit commitment: distribution of $\hat{\theta}$ roughly centered on θ^* (low bias)

Counterexample

$\hat{\theta}(x_1, \ldots, x_n) = \max_{i=1}^{n} x_i$

n samples: always finite

∞ samples: infinite

The Boostrap

Want to approximate hypothetical samples $\hat{\theta}', \hat{\theta}'', \ldots$

But only have actual data $x^{(1)}, \ldots, x^{(n)} \to \hat{\theta}$

Idea: subsample data

- With replacement
- n points in each sample

Bootstrap: Pseudocode

B: number of bootstrap samples

For $b = 1, \ldots, B$:

- Sample $x^{(1)\prime}, \ldots, x^{(n)\prime}$ with replacement from $x^{(1)}, \ldots, x^{(n)}$
- Let $\hat{\theta}^{(b)} = \hat{\theta}(x^{(1)\prime}, \ldots, x^{(n)\prime})$

Output $\{\hat{\theta}^{(1)}, \ldots, \hat{\theta}^{(B)}\}$

Bootstrap in python

[Jupyter demos]

When does the bootstrap work?

Most parametric estimators are fine

- I.e. fixed number of parameters d and $d \ll n$

When does the bootstrap work?

Most parametric estimators are fine

- I.e. fixed number of parameters d and $d \ll n$

NOT parametric:

- Decision trees
- Neural nets
- Kernel regression

These “interpolate” data, sampling with replacement $\approx$ subsampling

When does the bootstrap work?

Most parametric estimators are fine

- I.e. fixed number of parameters d and $d \ll n$

NOT parametric:

- Decision trees
- Neural nets
- Kernel regression

These "interpolate" data, sampling with replacement $\approx$ subsampling

Other commitments:

- $\hat{\theta}$ approximately unbiased
- θ^* is a meaningful quantity

Summary

- Credible intervals vs. confidence intervals
- Confidence intervals in statsmodels
- Still depend on assumptions!
- Bootstrap more robust (and flexible)