Bootstrap and Linear regression

Max Turgeon

STAT 3150—Statistical Computing

Lecture Objectives

- Understand the difference between resampling cases vs residuals.

Motivation

- In the last two modules, we reviewed linear regression and discussed residual analysis.
    - We discussed the linear regression assumptions, and their relative importance.
- In this module, we will discuss how to use bootstrap in the context of linear regression.
- There are actually 2 different ways of using bootstrap, corresponding to 2 different sets of assumptions concerning the data generating mechanism.

Bootstrap and Linear regression i

- When the error term is normally distributed, we know the distribution of the estimator $\hat{\beta}$:

$$\hat{\beta} \sim N\left(\beta, \sigma^2(\mathbb{X}^T\mathbb{X})^{-1}\right).$$

  - This can be used to compute p-values and confidence intervals.
- But when we *don't know* the distribution, or if we *don't want* to assume it follows a normal distribution, we can use bootstrap to make valid inference.

Bootstrap and Linear regression ii

- As we will see, there are two different ways to use bootstrap:
  - **Resample cases**;
  - **Resample residuals**.
- The main difference is how many assumptions we want to retain:
  - To resample residuals, we need to assume additivity, linearity, and homoscedasticity.
- In both cases, we still need to assume **independence of the errors**.

Resampling cases

- This is the simplest form of bootstrap for linear regression.
    - It should also be familiar.
- For this form of bootstrap to be valid, we only need to assume the errors are independent.
- In fact, it can be shown that when resampling cases, the bootstrap estimate of the standard error is approximately equal to the Huber-White robust standard error.

Algorithm (Cases)

1. Sample with replacement from $(Y_1, \mathbf{X}_1), \ldots, (Y_n, \mathbf{X}_n)$.
2. Refit the linear model using the bootstrap sample and obtain bootstrap estimates $\hat{\beta}^{(b)}$.

First example i

```
library(DAAG)
# Recall
fit1 <- lm(magnetic ~ chemical, data = ironslag)

n <- nrow(ironslag)
boot_beta1 <- replicate(5000, {
  indices <- sample(n, n, replace = TRUE)
  fit_boot <- lm(magnetic ~ chemical,
                 data = ironslag[indices, ])
  coef(fit_boot)
})
```

First example ii

```
str(boot_beta1)
```

```
## num [1:2, 1:5000] 7.1357 0.6105 -0.0984 1.0029
0.8562 ...
## - attr(*, "dimnames")=List of 2
## ..$ : chr [1:2] "(Intercept)" "chemical"
## ..$ : NULL
```

First example iii

```
se_int <- sd(boot_beta1[1,])
se_slope <- sd(boot_beta1[2,])

cbind("Lower" = coef(fit1) - 1.96*c(se_int, se_slope),
      "Upper" = coef(fit1) + 1.96*c(se_int, se_slope))
```

```
##                   Lower    Upper
## (Intercept) -3.2731976 6.078392
## chemical     0.6725151 1.159025
```

First example iv

```
# Compare to MLE theory
confint(fit1)
```

```
##                   2.5 %   97.5 %
## (Intercept) -3.7856893 6.590884
## chemical     0.6768355 1.154704
```

- Our confidence interval for the intercept is a bit smaller, but it still includes 0.
- On the other hand, the confidence interval for `chemical` is comparable to the one from MLE theory.

Resampling residuals i

- As mentioned above, this approach requires more assumptions than resampling cases:
    - Additivity and linearity;
    - Homoscedasticity.
- But the trade-off is that we get smaller confidence intervals than if we resample cases.

Resampling residuals ii

Algorithm (Residuals)

First, compute residuals E_i and fitted values $\hat{Y}_i = \hat{\beta}^T \mathbf{X}_i$ for each observation $i = 1, \ldots, n$.

1. Sample with replacement from the residuals and obtain a bootstrap sample $E_1^{(b)}, \ldots, E_n^{(b)}$.
2. Add the bootstrapped residuals to the fitted values: $Y_i^{(b)} = \hat{Y}_i + E_i^{(b)}$.
3. Using these new outcomes $Y_i^{(b)}$ and the original covariates $\mathbf{X}_i$, fit a linear regression model and obtain bootstrap estimates $\hat{\beta}^{(b)}$.

Second example i

```
library(MASS)
# Recall
dataset <- transform(mammals,
                     log_body = log(body),
                     log_brain = log(brain))

# Fit model
fit2 <- lm(log_brain ~ log_body, data = dataset)
```

Second example ii

```r
# Compute residuals
resids <- resid(fit2)

n <- length(resids)
boot_beta2 <- replicate(5000, {
  indices <- sample(n, n, replace = TRUE)
  logbrain_boot <- fitted(fit2) + resids[indices]
  fit_boot <- lm(logbrain_boot ~ log(mammals$body))
  coef(fit_boot)
})
```

Second example iii

```
str(boot_beta2)
```

```
## num [1:2, 1:5000] 2.128 0.778 2.051 0.733
1.992 ...
## - attr(*, "dimnames")=List of 2
## ..$ : chr [1:2] "(Intercept)"
"log(mammals$body)"
## ..$ : NULL
```

Second example iv

```
se_int <- sd(boot_beta2[1,])
se_slope <- sd(boot_beta2[2,])

cbind("Lower" = coef(fit2) - 1.96*c(se_int, se_slope),
      "Upper" = coef(fit2) + 1.96*c(se_int, se_slope))
```

```
##                   Lower     Upper
## (Intercept) 1.9517078 2.3178695
## log_body    0.6964323 0.8069395
```

Second example v

```
# Compare to MLE theory
confint(fit2)
```

```
##                   2.5 %    97.5 %
## (Intercept) 1.9426733 2.3269041
## log_body    0.6947503 0.8086215
```

- This time, we can see that we get essentially the same result in both cases.
    - The bootstrap confidence intervals are slightly smaller.

Second example vi

- **Note**: Other types of residuals can be used for the bootstrap, e.g. to mitigate the effect of outliers.
    - But don't use standardized residuals! You want the residuals to retain approximately the same variance as in the original data.

Final remarks i

- We looked at two different ways to perform bootstrap in the context of linear regression.
    - Resample the **cases** or the **residuals**.
- Resampling the cases is valid more generally than resampling the residuals.
- But resampling the residuals can lead to smaller, more accurate confidence intervals.
- Deciding which approach to use is a question of how much you trust the model.

Final remarks ii

- **Importantly**, neither approach is valid when the errors are *correlated*.
    - E.g. clustered data, repeated measurements, time series.
    - Bootstrap can be adapted to these methods, but this is beyond the scope of STAT 3150.