Bootstrapping

Andrew Jin

September 30, 2017

Abstract

In this post, we will discuss bootstrapping and apply bootstrapping methods to discuss the relationship between Urban Sprawl and Passenger Transportation. To begin exploring, we use a jackknife like method to create a 95% confidence interval for linear regressions. This is useful for visualizing the confidence of our linear regression and constructing hypothetical tests when collecting more data may be unfeasible. By doing this, we can better understand an estimate for the relationship between the miles of road within a town and the vehicle miles traveled, which can help infer better understandings of our data.

Introduction

In this post, I will look at the relationship between urban sprawl and vehicles miles traveled. Most specifically, I will look at federally designated urban areas in the United States and how the relationship of their population-scaled road length compares to their population-scaled vehicle miles traveled.

This question is important in sustainability because of the major policy implications of urban growth management for sustainable transportation. Urban sprawl is a major American trend that is one reason for America's high car usage (http://www.slate.com/articles/arts/architecture/2005/11/suburban_despair.html). To better understand the sustainability question with regard to urban sprawl, we will look at the relationship between urban sprawl and vehicle miles traveled in federally designated urban areas.

Studies have found that trip distance and car use greatly increase on city fringes due to urban sprawl (http://www.sciencedirect.com/science/article/pii/S0197397509000757). Many models of urban sprawl and its relationship to transportation and simulated futures utilize major data sets(http://onlinelibrary.wiley.com/doi/10.1111/j.1541-0064.2008.00214.x/full). As cities change, it is important for regulators to understand the implications of the data.

Because transportation data is hard to find in complete, well defined areas, especially in uniform manners, we must use some proxy values to analyze the relationship between sprawl and vehicle travel. The federal highway administration uses federal urban zones to characterize the total number of roads in federally designated urban areas and the total daily vehicle miles traveled within those urban areas. (https://www.fhwa.dot.gov/policyinformation/statistics/2015/2_intro.cfm). We utilize the census data provided in the database to scale both values. This census data defines "urban areas" as areas of 50,000 or more people (https://www.census.gov/geo/reference/ua/urban-rural-2010.html). We thus make the assumption that a more "sprawled" city contains more roads per person than does one with fewer road miles per person.

Resampling methods, such as bootstrapping ,allow researchers to estimate the precision of sample statistics or validate models by using random subsets of initial data to create useful inferences (http://mathworld.wolfram.com/BootstrapMethods.html). Bootstrapping pulls random samples with replacement to calculate regressions. While after analysis our data looked roughly linear, bootstrapping allows us to do statistical inference when the assumptions of normality and/or constant variance are violated.

Methods

Our analysis requires the aforementioned data set from the United States Federal Highway administration. It also requires the R packages "boot" and "dplyr"

Other examples of bootstrapping methods similar to ours were utilized to inform our code. (https://www.statmethods.net/advstats/bootstrapping.html, https://www.sagepub.com/sites/default/files/upm-binaries/21122_Chapter_21.pdf ,http://homepage.divms.uiowa.edu/~rdecook/stat3200/notes/bootstrap_4pp.pdf)

We start by adjusting our data to create our two statistics of interest.

```
library(dplyr)
```

```
##
## Attaching package: 'dplyr'
```

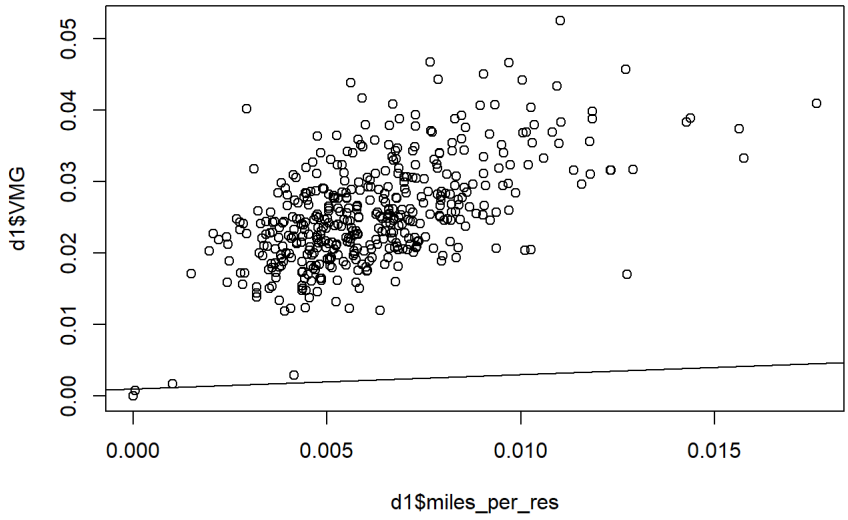
```
## The following objects are masked from 'package:stats':
##
##     filter, lag
```

```
## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

```
library(boot)
dat<- read.csv("C:/Users/Andrew S. Jin/Desktop/Stat 133/stat133-hws-fall17/post01/fhwa-urban-data.csv",  header =
TRUE, colClasses=c("character",rep("numeric",17)))
d1<- select(dat,FEDERAL.AID,CENSUS, TOTAL, VMT..TOTAL) #Limits the data set to create a smaller data frame with Ar
ea neames, Census Data, Total Number of Roads, and Total Vehicle Miles Traveled
d1<- mutate(d1, miles_per_res=  TOTAL / CENSUS) # Calculates the Miles of Road per Resident.
d1<- mutate(d1, VMG=  VMT..TOTAL / CENSUS) #Calculates the daily vehicle miles traveled per resident.
```

At this point we could perform a standard linear regression that would look something like this.

```
linearregression<-lm(miles_per_res~VMG,data=d1)
plot(d1$miles_per_res, d1$VMG)
abline(linearregression)
```



However, this value gives us no sense of variability or uncertainty. Thus we then create a bootstrapper function that allows us to perform a linear regression function that can be passed through the boot() function with given indices that vary depending on which points are sampled.

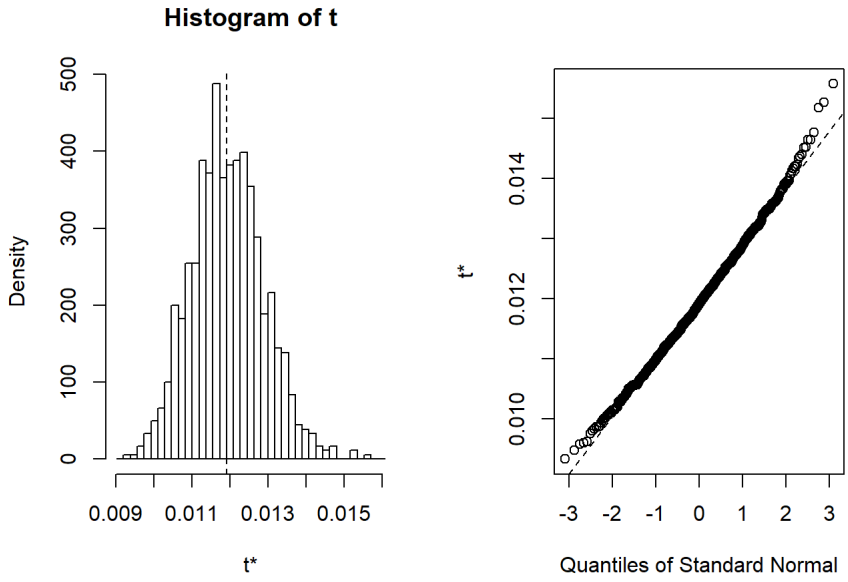
```
bootstrapper<- function(formula, data, indices){
  d<-data[indices,]
  fit<-lm(formula,data=d)
  return(coef(fit))
}
```

We can now utilize the boot function to pass through a large number of simulations

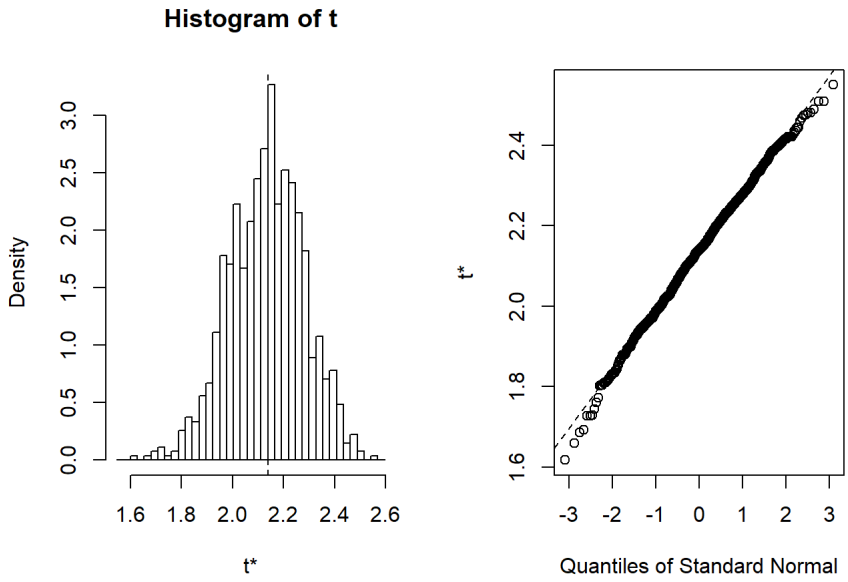
```
sims=1000
results <- boot(data=d1, statistic=bootstrapper,R=sims, formula=VMG~miles_per_res)
```

We can display the results of this analysis by looking at the variability of this data.

```
## plot of index 1 (Intercept)
plot(results, index=1)
```




```
## plot of index 2 (Slope)
plot(results, index=2)
```



We can even see the values for results of the data such that:

```
## Index1 (Intecept Statistics)
boot.ci(boot.out=results,type='perc',index=1)
```

```
## BOOTSTRAP CONFIDENCE INTERVAL CALCULATIONS
## Based on 1000 bootstrap replicates
##
## CALL :
## boot.ci(boot.out = results, type = "perc", index = 1)
##
## Intervals :
## Level     Percentile
## 95%   ( 0.0101,  0.0139 )
## Calculations and Intervals on Original Scale
```

```
## Index 2 ( Slope Statistics)
boot.ci(boot.out=results,type='perc',index=2)
```

```
## BOOTSTRAP CONFIDENCE INTERVAL CALCULATIONS
## Based on 1000 bootstrap replicates
##
## CALL :
## boot.ci(boot.out = results, type = "perc", index = 2)
##
## Intervals :
## Level     Percentile
## 95%   ( 1.834,  2.413 )
## Calculations and Intervals on Original Scale
```