Title

Comparing the Distributions of Home Rental Prices in Los Angeles County between Craigslist and the American Community Survey

Author

Billah, Moody

Publication Date

2022

Peer reviewed|Thesis/dissertation

UNIVERSITY OF CALIFORNIA
Los Angeles

Comparing the Distributions of Home Rental Prices in Los Angeles County between Craigslist and the American Community Survey

A thesis submitted in partial satisfaction
of the requirements for the degree
Master of Applied Statistics

by

Muhtasham Billah

2022

ABSTRACT OF THE THESIS

Comparing the Distributions of Home Rental Prices
in Los Angeles County between
Craigslist and the American Community Survey

by

Muhtasham Billah
Master of Applied Statistics
University of California, Los Angeles, 2022
Professor Mark Stephen Handcock, Chair

Prior research in the area of housing policy has shown that rental prices collected from the US government surveys do not reflect the market rental prices that are paid by new rent seekers. This paper is motivated by previous studies, where researchers have particularly criticized the American Community Survey (ACS) conducted by the US Census Bureau for heavily underestimating the market price of rental housing units. It collects rental data for Los Angeles County web scraped from Craigslist, a popular online marketplace, as an estimate for the current rental market. It then compares the distribution of that data with rental prices collected by the ACS. The results confirm previous findings since rental prices collected from the ACS are significantly lower than Craigslist. In addition, the rental data from Craigslist show a clear effect caused by the COVID-19 pandemic, which is not present in the ACS data.

The thesis of Muhtasham Billah is approved.

Nicolas Christou

Frederic R. Paik Schoenberg

Mark Stephen Handcock, Committee Chair

University of California, Los Angeles

2022

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

CHAPTER 1

Introduction

When the US Census Bureau conducts the American Community Survey (ACS), a question that is asked to renting households is how much money they spend on rent every month. This methodology is good for understanding how much rent is paid by current residents. However, it does not reflect the market rental price a new resident will pay for the same units. Due to various economic and political factors, such as rent control and tenant protections, the rental price for new tenants is often substantially higher compared to existing tenants. As a result, rental market information cited from the ACS is often misleading.

There has been extensive research in the area of housing policy that illustrates this phenomenon, leading to a consensus among researchers that the ACS data will not provide an accurate idea of the current housing rental market. A study published in 2017 by Boeing and Waddell web scraped rental listings from Craigslist for a number of metropolitan areas in the US, then compared them to the ACS data. Their results strongly suggest that the ACS underestimates the current market rental prices. A follow-up study published in 2020 by Boeing, Wegmann, and Jiao updated the data from Craigslist and the ACS, but it lead to the same conclusion. A separate study by Myers and Park published in 2019 also criticized affordability metrics calculated from the ACS data, showing that they are unreliable due to the discrepancies in current market rental prices.

Motivated by the previous research, this paper will web scrape rental listings from Craigslist and compare them to the ACS, but the geographic scope will be limited to Los Angeles County. In addition to the overall effects between the two data sources, this analysis

will cover effects from the COVID-19 pandemic and the number of bedrooms. Simulations will be run to generate data in cases where individual observations do not exist. Comparisons will be done based on the distribution of the data so that differences in variance are accounted for in the results.

CHAPTER 2

Data

2.1 ACS

The ACS collects and publishes data annually for each county in the US, which are readily available on their website. For the purpose of this study, the dataset from 2019 is the indicator for pre-pandemic, while the dataset from 2020 is post-pandemic. The datasets published do not have any individual observations, but give the share of observations within specified ranges. The following tables show the particular ACS datasets that were used for this study.

Gross Rent	**Estimate**	**Percent**
Less than $500	79,829	4.6%
$500 to $999	257,120	14.7%
$1,000 to $1,499	584,645	33.4%
$1,500 to $1,999	425,144	24.3%
$2,000 to $2,499	214,039	12.2%
$2,500 to $2,999	99,969	5.7%
$3,000 or more	90,965	5.2%

Table 2.1: 2019 ACS Gross Rent

Bedrooms	Estimate	Percent
No bedroom	250,626	7.1%
1 bedroom	678,129	19.1%
2 bedrooms	1,060,590	29.9%
3 bedrooms	1,003,706	28.3%
4 bedrooms	434,082	12.3%
5 or more bedrooms	115,667	3.3%

Table 2.2: 2019 ACS Bedrooms

Gross Rent	Estimate	Percent
Less than $500	78,101	4.5%
$500 to $999	223,123	12.7%
$1,000 to $1,499	544,078	31.1%
$1,500 to $1,999	443,379	25.3%
$2,000 to $2,499	236,205	13.5%
$2,500 to $2,999	114,397	6.5%
$3,000 or more	112,156	6.4%

Table 2.3: 2020 ACS Gross Rent

Bedrooms	Estimate	Percent
No bedroom	254,783	7.2%
1 bedroom	683,893	19.2%
2 bedrooms	1,061,722	29.8%
3 bedrooms	1,001,007	28.1%
4 bedrooms	437,634	12.3%
5 or more bedrooms	120,751	3.4%

Table 2.4: 2020 ACS Bedrooms

2.2 Craigslist

The rental listings web scraped from Craigslist is sourced using WayBack Machine, an open-source digital archive of historical web pages. This allows for the retrieval of historical listings rather than just the current ones. A sample of 7 Craigslist web pages are chosen with dates ranging between 2017 and 2021. Web pages dated before March 11, 2020 are indicated as pre-pandemic, while the ones after that date are indicated as post-pandemic. The final dataset contains a total of 719 observations with 3 variables, which are rental price, number of bedrooms, and square footage. A drawback of this data collection method is that the samples are not entirely random, since they are influenced by WayBack Machine's archiving methodology. However, with a high number of observations across a long time span, it is assumed that this data is random enough for the purpose of this analysis.

CHAPTER 3

Methodology

3.1 Exploring Craigslist Data

The first step of this analysis to explore the Craigslist data in order to make reasonable assumptions, which will help draw better conclusions when comparing with the ACS data. The overall rent distribution, the distribution of rent by pandemic, and the distribution rent by bedrooms are investigated. All of these distributions are mostly normal, which means that standard statistical methods can be applied with sufficient validity. A preliminary linear model with the rental price as the response variable shows that the number of bedrooms and square footage have high multicollinearity, with a VIF (Variance Inflation Factor) score 3 times higher than the standard threshold. Since the square footage variable has some missing values as well, it is removed from further analysis. Afterwards, an ANOVA (Analysis of Variance) shows that there is a significant difference in rental prices at the 5% level for both pandemic status and number of bedrooms, but their interaction term is not significant. This leads to the key assumption that the ACS rental prices will also have a similar relationship with pandemic status and number of bedrooms, which will greatly simplify the comparison. The following figures illustrate various distributions of the Craigslist data.

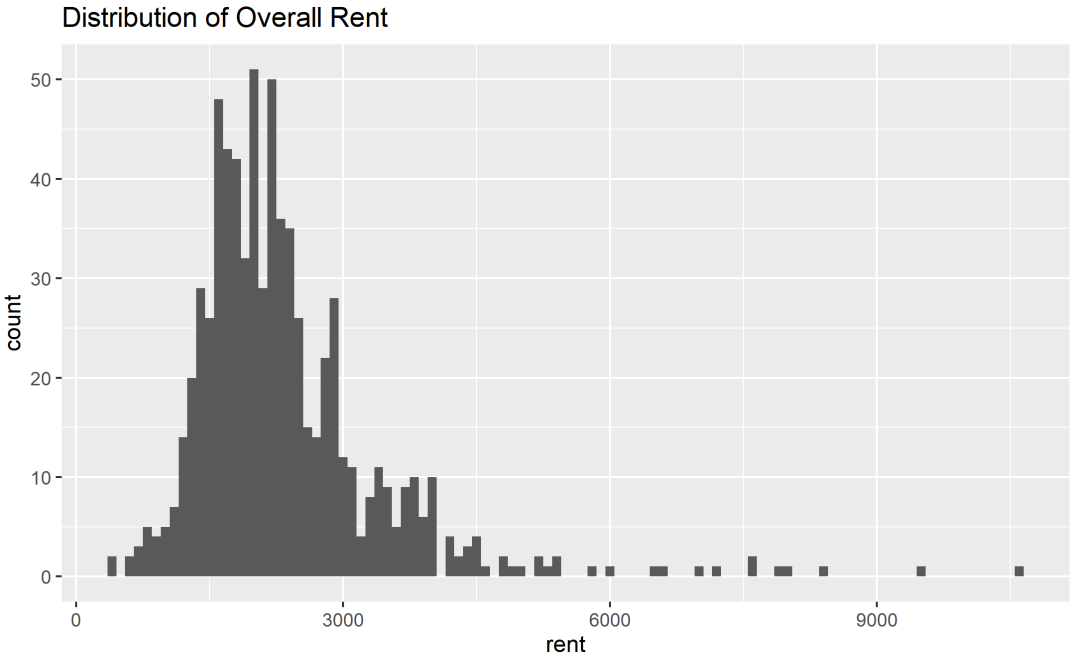

Figure 3.1: Craigslist Distribution of Overall Rent

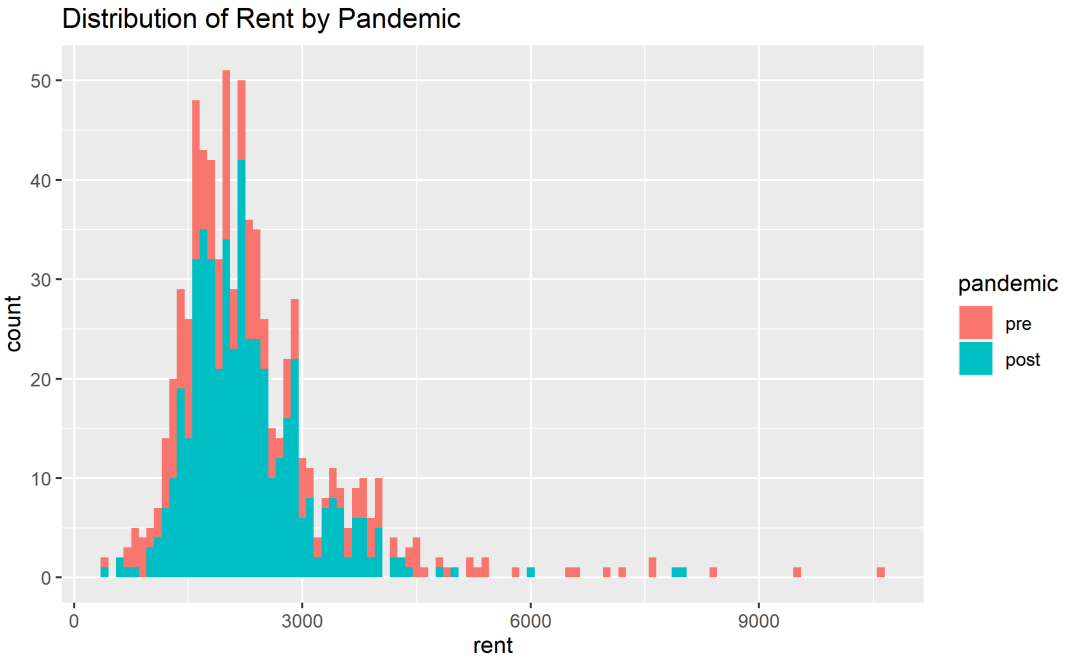

Figure 3.2: Craigslist Distribution of Rent by Pandemic

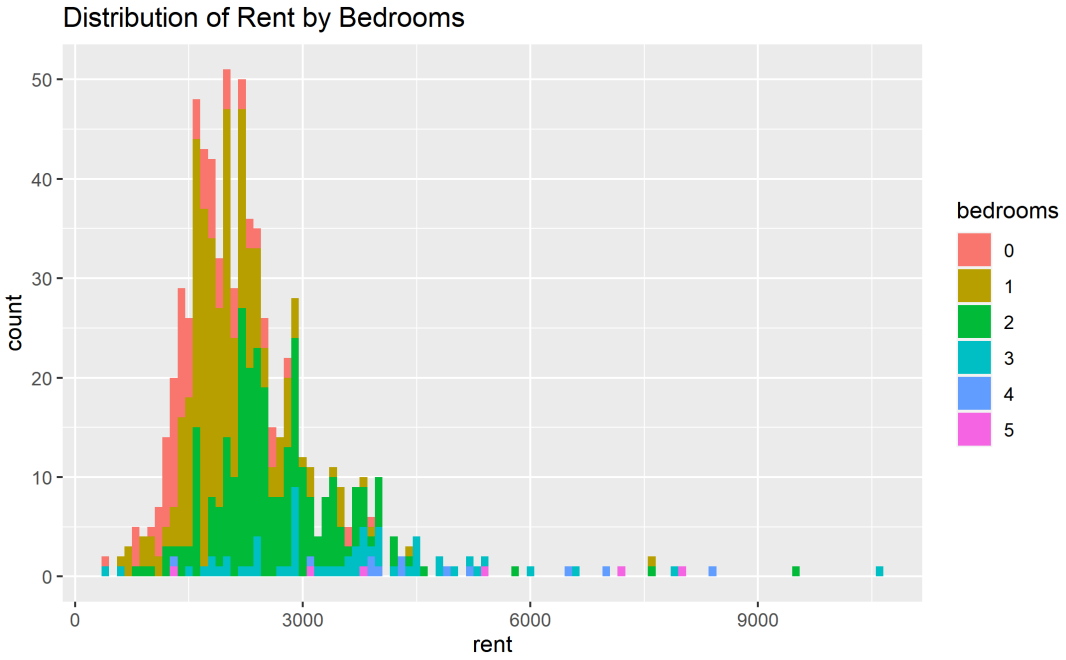

Figure 3.3: Craigslist Distribution of Rent by Bedrooms

3.2 Generating ACS Rent Observations

Since the ACS dataset does not provide individual rent observations, they need to be computer generated. For each specified range in the dataset, a random uniform sample of rent values are simulated. Then, bootstrapping (re-sampling with replacement) is done by combining all the ranges, with the probability distribution equal to the proportion of observations in each range. This provides a set of rent values with a distribution similar to the one published by the ACS. The entire sampling process is repeated twice due to the assumption of significant difference between pandemic statuses. The pre-pandemic observations are sampled from the 2019 dataset, while the post-pandemic observations are sampled from the 2020 dataset. The following figures illustrate the distribution of the computer generated ACS rental prices.

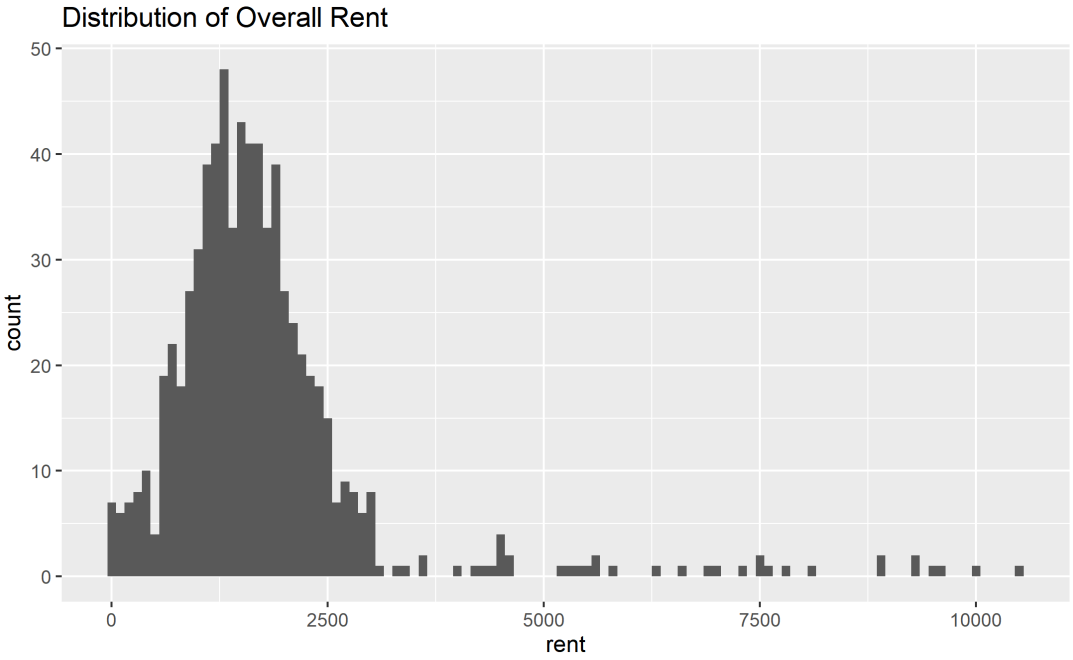

Figure 3.4: ACS Distribution of Overall Rent

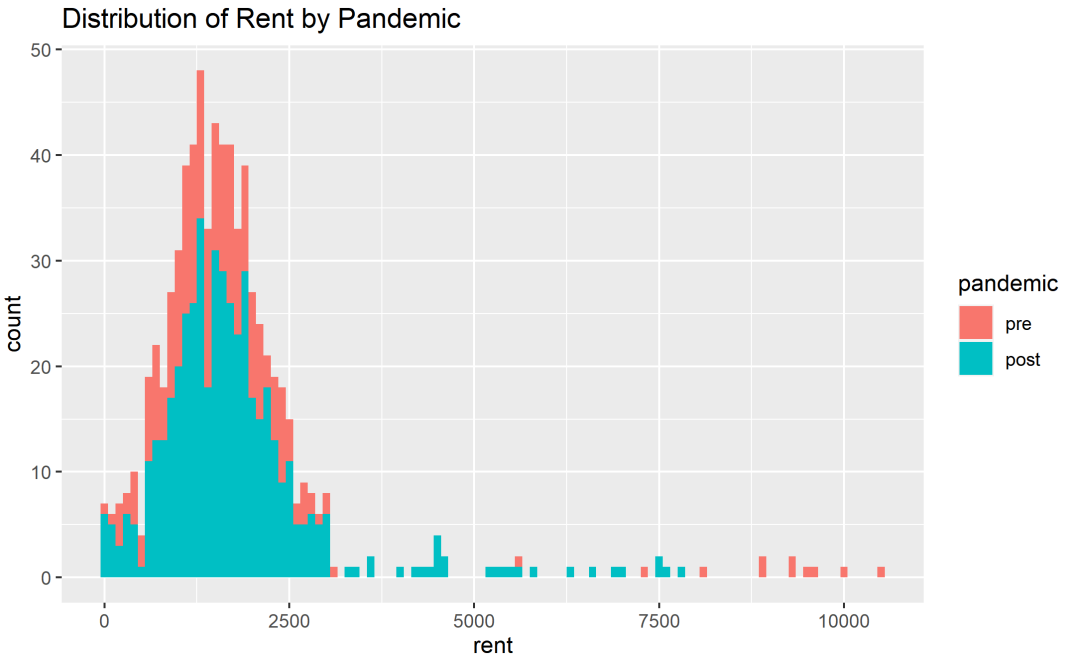

Figure 3.5: ACS Distribution of Rent by Pandemic

3.3 Generating ACS Bedroom Observations

The ACS bedroom observations are computer generated using the same sampling process as the rent observations shown previously. However, the set of bedroom observations are not useful for this analysis unless they are associated with a corresponding rent value. In order to match the rent and bedroom observations with each other, the Gibbs sampling method in the MCMC (Monte Carlo Markov Chain) algorithm is used. This method is generally used to generate a sample for a multivariate target distribution by iterating over samples of conditional univariate distributions. A mathematical overview of the Gibbs sampling method is shown below.

$$Assume\ Pr(Rent|Bedrooms) = Pr(Bedrooms|Rent)$$

$$Let\ \pi(X) = \pi(x_1, x_2, ..., x_n)\ be\ the\ target\ distribution$$

$$Let\ X^{(t)} = (x_1^{(t)}, x_2^{(t)}, ..., x_n^{(t)})\ be\ the\ sample\ at\ time\ t$$

$$Generate\ x_i^{(t+1)} \sim \pi(x_i|x_1^{(t+1)}, x_2^{(t+1)}, ..., x_{i-1}^{(t+1)}, x_{i+1}^{(t)}, ..., x_n^{(t)})$$

$$For\ i = 1, 2, ..., n$$

In this particular case, the target is the joint distribution of the bedroom and rent values. For each bedroom observation, a rent value is assigned by sampling from a subset of the rent observations. The subset is equivalent to the conditional distribution in Gibbs sampling, and is determined by a percentile range based on the given bedroom value. The percentile range is validated by the assumption that the underlying relationship between rental price and number of bedrooms is similar for both the ACS and Craigslist data. First, the 95% confidence interval of Craigslist rent for each subset of bedroom values is calculated. Next, those confidence intervals are converted into percentile ranges. Then, those percentile ranges are translated over to the ACS rent observations to create the subsets corresponding to the bedroom values. The 95% confidence interval provides a good balance of capturing enough

variation, but not skewing the subset due to influence from outliers. The entire sampling and updating process is iterated 100 times in order to obtain a sufficiently good final sample. An alternative methodology that would give similar results is post-stratification survey sampling, where the ACS bedroom observations are simulated from a post-stratified sample based on survey weights from the Craigslist data. The following figure illustrates the joint distribution of the computer generated ACS rent and bedroom values.

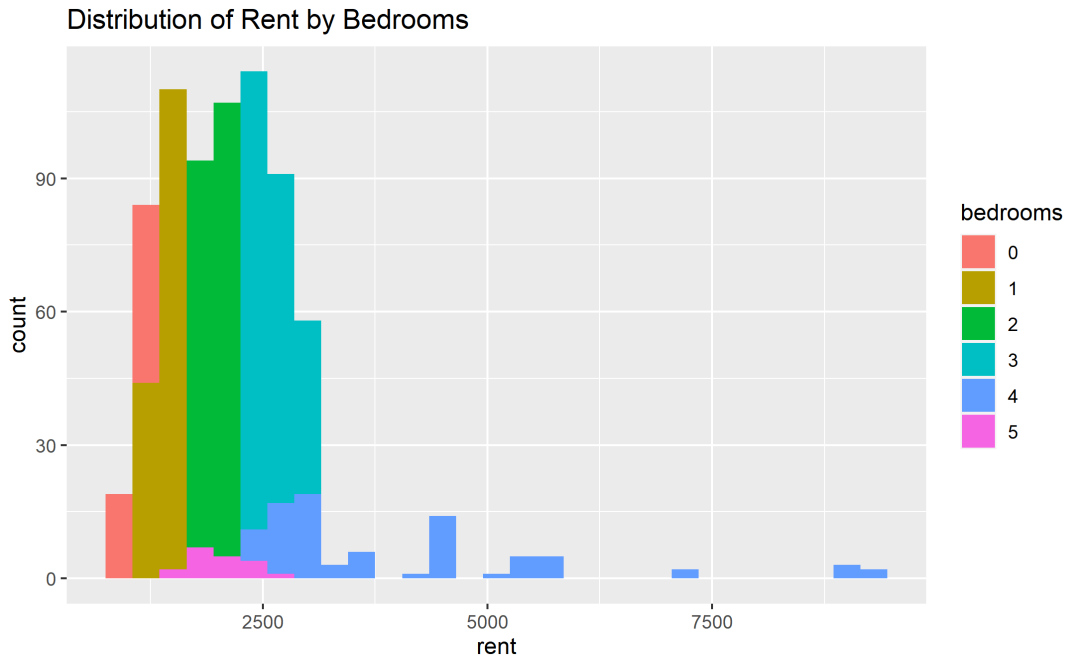

Figure 3.6: ACS Distribution of Rent by Bedrooms

CHAPTER 4

Results

4.1 Overall Effects

In aggregate, the rental prices from the Craigslist data are significantly higher than the ACS data. A 2-sample t-test confirms this at a 5% significance level (p-value $\approx$ 0.0000), where the mean Craigslist rent is \$2,361 and the mean ACS rent is \$1,766. The following figure compares the overall distributions of the ACS and Craigslist rental prices.

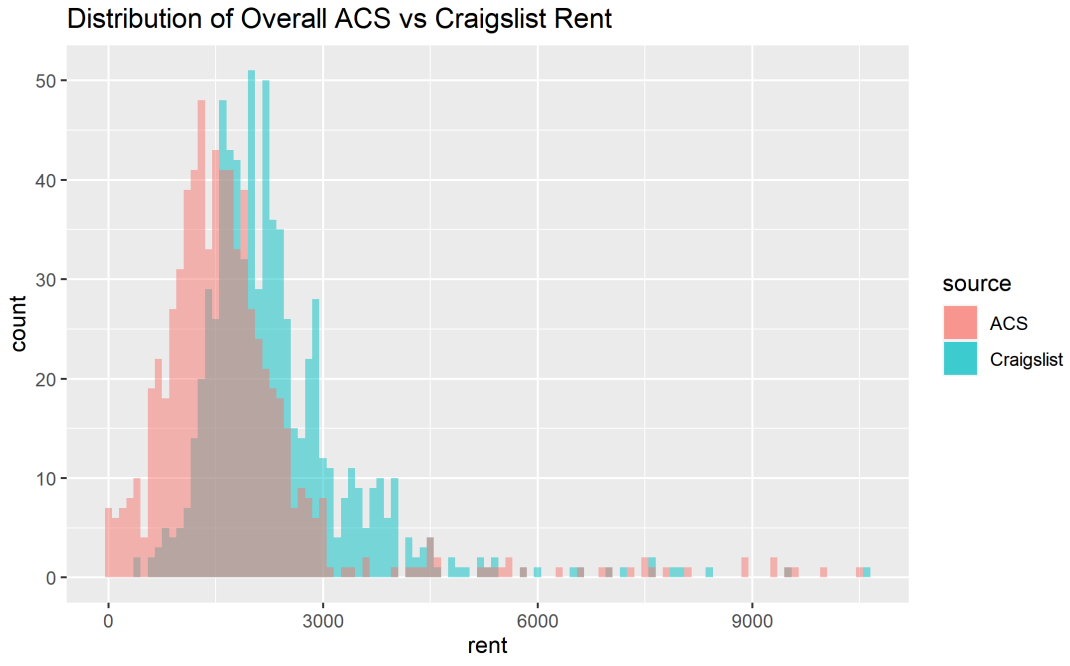

Figure 4.1: ACS vs Craigslist Distribution of Overall Rent

4.2 Pandemic Effects

As a result of the COVID-19 pandemic, rental prices dropped for both the Craigslist and ACS data. However, the ANOVA conducted shows that the drop in the Craigslist rent is significant at the 5% level, while the drop in ACS rent is not significant. This indicates a pandemic effect present in the Craigslist data that is not seen in the ACS data. Nevertheless, 2-sample t-tests at a 5% significance level show that rental prices on Craigslist are significantly higher than the ACS for both pre and post pandemic. The following table and figures compare the pre and post pandemic rental prices between the ACS and Craigslist.

	ACS Mean Rent	**Craigslist Mean Rent**	**p-value (t-test)**
Pre-Pandemic	\$1,806	\$2,541	0.0000***
Post-Pandemic	\$1,745	\$2,270	0.0000***
p-value (ANOVA)	0.57	0.0018**	

Table 4.1: ACS vs Craigslist Pandemic Effects

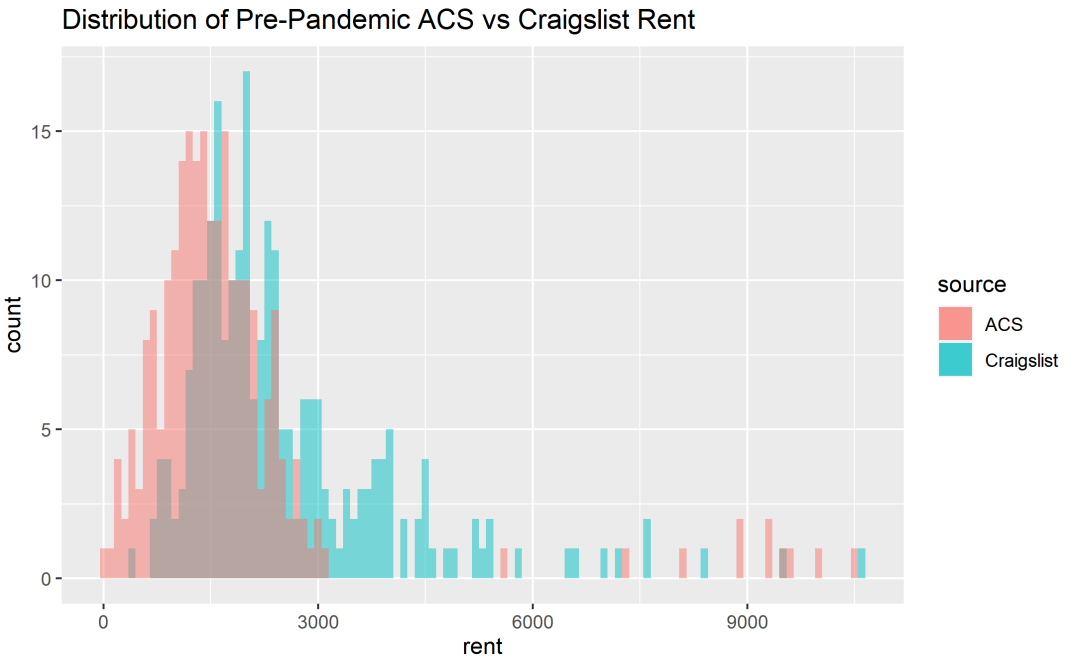

Figure 4.2: ACS vs Craigslist Distribution of Pre-Pandemic Rent

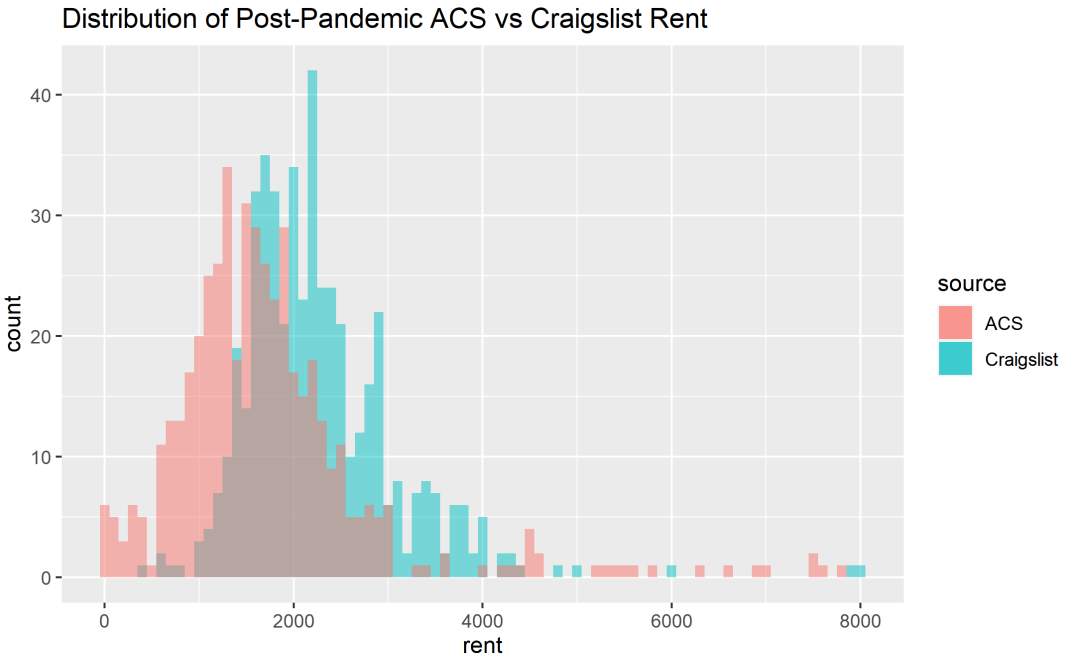

Figure 4.3: ACS vs Craigslist Distribution of Post-Pandemic Rent

4.3 Bedroom Effects

Breaking down by the number of bedrooms, Craigslist rental prices are significantly higher than the ACS for all bedroom values. This is confirmed by conducting multiple 2-sample t-tests at the 5% significance level. Number of bedrooms equal to 4 and up are grouped together because there are not enough observations above 4 bedrooms to make any meaningful conclusions. The following table compares the rental prices by number of bedrooms between the ACS and Craigslist.

Bedrooms	**ACS Mean Rent**	**Craigslist Mean Rent**	**p-value (t-test)**
0	\$1,103	\$1,701	0.0000***
1	\$1,396	\$1,974	0.0000***
2	\$1,955	\$2,612	0.0000***
3	\$2,622	\$3,576	0.0000***
4+	\$3,642	\$4,756	0.0414*

Table 4.2: ACS vs Craigslist Bedroom Effects

CHAPTER 5

Discussion

In conclusion, this study confirms the trend seen in prior housing policy research within the scope of Los Angeles County. The ACS severely underestimates housing rental prices and does not represent the true price paid by new rent seekers. By using Craigslist rental listings as an estimator for the current rental market, this study presents evidence for significantly higher market rent for both before and after the COVID-19 pandemic, as well as for any number of bedrooms. Furthermore, the ACS is unable to account for external shocks in the rental market as seen by the lack of any significant effect due to the pandemic. On the other hand, the data from Craigslist did show a significant drop in rental prices after the pandemic, which is more realistic to the change in economic conditions.

Although this study presents some clear evidence, there are a few shortcomings. The rental listings that are web scraped from Craigslist are not completely random, so they may contain biases from the archiving methods of WayBack Machine. Adding additional data from other online marketplaces could help reduce this bias. Another shortcoming is that individual records of the ACS are not readily available, so the data needs to be computer generated from summary tables and a set of assumptions. In order to properly run the simulations, this study assumed that the ACS and Craigslist rent prices follow similar patterns in relation to other factors. This is a very strong assumption which may not be realistic, so individual observations from the ACS are required to make the comparison more valid and reliable.

Moving forward, data collection methodologies in the area of housing policy need to

substantially improve. As shown by this study as well as prior studies, housing surveys conducted by the US Census Bureau have clear gaps in terms of accurately assessing economic factors, such as the cost of housing. This is especially true for large and diverse urban regions like Los Angeles County, where there is a constant inflow and outflow of rental residents. By collecting data using market oriented methodologies, governments across all levels would be able to initiate policies that are more appropriate for the current housing situation.

APPENDIX A

Web Scraping in Python

```
#Importing libraries


import itertools
import requests
from sqlalchemy import create_engine
import pandas as pd
import numpy as np
from bs4 import BeautifulSoup

# Database connection


conn = create_engine(
                    'mysql://root:root@localhost/craigslist_web_scraping'
                    )


#Getting the webpages from the database


q1 = 'select page_ID from webpages'
q2 = 'select page_url from webpages'
page_id = pd.read_sql(q1, conn)['page_ID']
page_url = pd.read_sql(q2, conn)['page_url']
```

```
#Defining web scraping function

def web_scraping(index, page_url, page_id):

    #Initiating the web scraper for the url

    url = page_url[index]
    page = requests.get(url)
    parser = BeautifulSoup(page.content, "html.parser")

    #Getting all the listings data into a dataframe

    listings = pd.DataFrame(parser.find_all(class_="result-meta"),
                            dtype=object)

    #Defining fuction to parse separate sections of the listings data

    def parse_listings(listings, col_name, class_name):
        var = list(itertools.repeat(0, len(listings)))
        for i in range(len(listings)):
            var[i] = listings.iloc[i,0].find(class_=class_name)
            i += 1
        var = pd.DataFrame(var, columns=[col_name])
        return var

    #Getting and cleaning all the rental price data into a dataframe

    price = parse_listings(listings, "rent", "result-price")
```

```
price["rent"] = price["rent"].str.replace("\$", "")
price["rent"] = price["rent"].str.replace(",", "")
price = price.astype(int)

#Getting and cleaning all the property size data into a dataframe

size = parse_listings(listings, "size", "housing")
size = size.astype("str")
size["size"] = size["size"].str.replace(
            '<span class="housing">\n                    ', ''
            )
size["size"] = size["size"].str.replace(
            '<sup>2</sup> -\n                </span>', ''
            )
size["size"] = size["size"].str.replace(
            ' -\n                </span>', ''
            )
size["size"] = size["size"].str.split(
            ' -\n                    '
            )

#Separating and cleaning bedrooms and square feet data

beds = pd.DataFrame(index=range(len(size)),
                    columns=["bedrooms"], dtype="str")
sq_ft = pd.DataFrame(index=range(len(size)),
                    columns=["sq_feet"], dtype="str")

for i in range(len(size)):
```

```
    if len(size["size"][i]) == 2:
        beds["bedrooms"][i] = size["size"][i][0]
        sq_ft["sq_feet"][i] = size["size"][i][1]
    elif "br" in size["size"][i][0]:
        beds["bedrooms"][i] = size["size"][i][0]
        sq_ft["sq_feet"][i] = -1
    elif "ft" in size["size"][i][0]:
        beds["bedrooms"][i] = -1
        sq_ft["sq_feet"][i] = size["size"][i][0]
    else:
        beds["bedrooms"][i] = -1
        sq_ft["sq_feet"][i] = -1
    i += 1

beds["bedrooms"] = beds["bedrooms"].str.replace("br", "")
sq_ft["sq_feet"] = sq_ft["sq_feet"].str.replace("ft", "")
beds = beds.fillna(-1).astype(int)
sq_ft = sq_ft.fillna(-1).astype(int)

 #Getting and cleaning all the neighborhood data into a dataframe

hood = parse_listings(listings, "location", "result-hood")
hood = hood.astype("str")
hood["location"] = hood["location"].str.replace(
            '<span class="result-hood"> ', ''
            )
hood["location"] = hood["location"].str.replace('</span>', '')
hood["location"] = hood["location"].str.replace('nan', '')
hood["location"] = hood["location"].str.replace('(', '')
```

```
    hood["location"] = hood["location"].str.replace(')', '')

    #Generating the IDs for the final dataset

    pid = pd.DataFrame(itertools.repeat(page_id[index], len(listings)),
                       columns=["page_id"])

    #Creating the final dataset and and adding it to the database

    dataset = pd.concat([pid, price, beds, sq_ft, hood], axis=1)
    dataset.to_sql(name='listings', con=conn,
                   schema='craigslist_web_scraping',
                   index=False, if_exists='append'
                   )

#Running web scraping function

for i in range(len(page_url)):
    web_scraping(i, page_url, page_id)
    i += 1
```

APPENDIX B

Analysis in R

```
library(DBI)
library(RMySQL)
library(sqldf)
library(tidyverse)
library(car)
library(Rmisc)

#Reading and cleaning web scraped data with 719 observations

con = dbConnect(MySQL(), user='root', password='root',
                dbname='craigslist_web_scraping', host='localhost')
query = "select rent, bedrooms, sq_feet, page_date
         from listings l join webpages w
         on l.page_ID = w.page_ID where rent > 0"
craigslist_df = dbGetQuery(con, query)

craigslist_df = mutate(craigslist_df,
                bedrooms = ifelse(bedrooms < 0, 0, bedrooms))
craigslist_df = mutate(craigslist_df,
                bedrooms = ifelse(bedrooms > 5, 5, bedrooms))
craigslist_df = mutate(craigslist_df,
```

```
                sq_feet = ifelse(sq_feet < 0, NA, sq_feet))
craigslist_df$bedrooms = as.factor(craigslist_df$bedrooms)

dim(craigslist_df)
head(craigslist_df)

#Creating pandemic variable with the cutoff at date at 2020-03-11

craigslist_df$page_date = as.Date(
                    craigslist_df$page_date, format="%Y-%m-%d"
                    )
craigslist_df$pandemic = as.factor(
                    ifelse(craigslist_df$page_date >= "2020-03-11", 1, 0)
                    )

#Overall distribution

ggplot(craigslist_df, aes(x=rent)) + geom_histogram(binwidth=100) +
       ggtitle("Distribution of Overall Rent")
ggsave("cl_rent_overall.png")

#Distributions by pandemic

ggplot(craigslist_df, aes(x=rent, fill=pandemic)) +
       geom_histogram(binwidth=100) +
       ggtitle("Distribution of Rent by Pandemic") +
       scale_fill_hue(labels=c("pre","post"))
ggsave("cl_rent_pandemic.png")
```

```
#Significant difference pre and post pandemic (ANOVA assumptions met)

aov_pandemic = aov(rent ~ pandemic, craigslist_df)
summary(aov_pandemic)

plot(aov_pandemic, which=c(2,3))

#Distribution by bedrooms

ggplot(craigslist_df, aes(x=rent, fill=bedrooms)) +
       geom_histogram(binwidth=100) +
       ggtitle("Distribution of Rent by Bedrooms")
ggsave("cl_rent_bedrooms.png")

#Significant difference between bedrooms (ANOVA assumptions met)

aov_bedrooms = aov(rent ~ bedrooms, craigslist_df)
summary(aov_bedrooms)

plot(aov_bedrooms, which=c(2,3))

#Distribution by pandemic and bedrooms

pre_pandemic = filter(craigslist_df, pandemic == 0)
post_pandemic = filter(craigslist_df, pandemic == 1)

ggplot(pre_pandemic, aes(x=rent, fill=bedrooms)) +
       geom_histogram(binwidth=100) +
       ggtitle("Distribution of Rent by Bedrooms Pre Pandemic")
```

```
ggplot(post_pandemic, aes(x=rent, fill=bedrooms)) +
       geom_histogram(binwidth=100) +
       ggtitle("Distribution of Rent by Bedrooms Post Pandemic")

#No significant interaction between pandemic and bedrooms
#(ANOVA assumptions met)

aov_pan_bed = aov(rent ~ pandemic*bedrooms, craigslist_df)
summary(aov_pan_bed)

plot(aov_pan_bed, which=c(2,3))

#Square feet and bedrooms have high multicollinearity
#square feet will be ignored from analysis
#(linear regression assumptions met)

lm_sq_bed = lm(rent ~ sq_feet + bedrooms + pandemic, craigslist_df)
summary(lm_sq_bed)
vif(lm_sq_bed)

plot(lm_sq_bed, which=c(1,2))

#Generating ACS bedroom data pre pandemic

set.seed(100)
acs_beds_prepan = sample(c(0,1,2,3,4,5), length(pre_pandemic$bedrooms),
                  replace=T,
                  prob=c(0.071,0.191,0.299,0.283,0.123,0.033)
```

```
                  )

#Generating ACS bedroom data post pandemic

set.seed(100)
acs_beds_postpan = sample(c(0,1,2,3,4,5), length(post_pandemic$bedrooms),
                   replace=T,
                   prob=c(0.072,0.192,0.298,0.281,0.123,0.034)
                   )

#Creating function to generate ACS rent data

acs_generate_rent = function(rindex, max_rent, n){
  acs_rent = rep(0, n)
  for (i in 1:n){
  acs_rent[i] = case_when(
    rindex[i] == 1 ~ runif(1, 0, 499),
    rindex[i] == 2 ~ runif(1, 500, 999),
    rindex[i] == 3 ~ runif(1, 1000, 1499),
    rindex[i] == 4 ~ runif(1, 1500, 1999),
    rindex[i] == 5 ~ runif(1, 2000, 2499),
    rindex[i] == 6 ~ runif(1, 2500, 2999),
    rindex[i] == 7 ~ runif(1, 3000, max_rent)
    )
  }
  return(acs_rent)
}

#Generating ACS rent data pre pandemic
```

```
set.seed(100)
rindex_prepan = sample(c(1,2,3,4,5,6,7), length(pre_pandemic$bedrooms),
                replace=T,
                prob=c(0.046,0.147,0.334,0.243,0.122,0.057,0.052)
                )

acs_rent_prepan = acs_generate_rent(rindex_prepan, max(pre_pandemic$rent),
                                    length(pre_pandemic$rent))

#Generating ACS rent data post pandemic

set.seed(100)
rindex_postpan = sample(c(1,2,3,4,5,6,7), length(post_pandemic$bedrooms),
                 replace=T,
                 prob=c(0.045,0.127,0.311,0.253,0.135,0.065,0.064)
                 )

acs_rent_postpan = acs_generate_rent(rindex_postpan,
                   max(post_pandemic$rent),
                   length(post_pandemic$rent)
                   )

#Distribution of ACS rent

acs_df = rbind(data.frame(rent=acs_rent_prepan, pandemic=as.factor(0)),
               data.frame(rent=acs_rent_postpan, pandemic=as.factor(1)))

ggplot(acs_df, aes(x=rent)) + geom_histogram(binwidth=100) +
```

```
       ggtitle("Distribution of Overall Rent")
ggsave("acs_rent_overall.png")

#Distribution of ACS rent by pandemic

ggplot(acs_df, aes(x=rent, fill=pandemic)) +
       geom_histogram(binwidth=100) +
       ggtitle("Distribution of Rent by Pandemic") +
       scale_fill_hue(labels=c("pre","post"))
ggsave("acs_rent_pandemic.png")

#No significant difference on ACS pre and post pandemic
#(normality assumption skewed right)

aov_acs_pandemic = aov(rent ~ pandemic, acs_df)
summary(aov_acs_pandemic)

plot(aov_acs_pandemic, which=c(2,3))

#Significant difference between ACS and Craigslist rent overall

t.test(craigslist_df$rent, acs_df$rent)

ggplot() + geom_histogram(data=craigslist_df,
                          aes(x=rent, fill="Craigslist"),
                          binwidth=100, alpha=0.5) +
  geom_histogram(data=acs_df, aes(x=rent, fill="ACS"),
                 binwidth=100, alpha=0.5) +
  ggtitle("Distribution of Overall ACS vs Craigslist Rent") +
```

```
  scale_fill_discrete(name="source")
ggsave("cl_acs_overall.png")

#Significant difference between ACS and Craigslist rent pre pandemic

t.test(pre_pandemic$rent, acs_rent_prepan)

prepan_acs = filter(acs_df, pandemic == 0)

ggplot() + geom_histogram(data=pre_pandemic,
                          aes(x=rent, fill="Craigslist"),
                          binwidth=100, alpha=0.5) +
  geom_histogram(data=prepan_acs, aes(x=rent, fill="ACS"),
                 binwidth=100, alpha=0.5) +
  ggtitle("Distribution of Pre-Pandemic ACS vs Craigslist Rent") +
  scale_fill_discrete(name="source")
ggsave("cl_acs_prepan.png")

#Significant difference between ACS and Craigslist rent post pandemic

t.test(post_pandemic$rent, acs_rent_postpan)

postpan_acs = filter(acs_df, pandemic == 1)

ggplot() + geom_histogram(data=post_pandemic,
                          aes(x=rent, fill="Craigslist"),
                          binwidth=100, alpha=0.5) +
  geom_histogram(data=postpan_acs, aes(x=rent, fill="ACS"),
                 binwidth=100, alpha=0.5) +
```

```
  ggtitle("Distribution of Post-Pandemic ACS vs Craigslist Rent") +
  scale_fill_discrete(name="source")
ggsave("cl_acs_postpan.png")

#Getting 95% confidence interval of Craigslist rent by bedroom

beds = sort(unique(craigslist_df$bedrooms))
upper = mean = lower = rep(0, length(beds))

for (i in 1:length(beds)){
  bed_rent = craigslist_df %>% filter(bedrooms == i-1) %>% select(rent)
  upper[i] = CI(bed_rent$rent)[1]
  mean[i] = CI(bed_rent$rent)[2]
  lower[i] = CI(bed_rent$rent)[3]
}

beds_confint = cbind.data.frame(beds, lower, mean, upper)
beds_confint

#Getting the percentiles from overall rent by bedroom using
#95% confidence interval bounds

beds_confint$lower_percent = ecdf(craigslist_df$rent)(beds_confint$lower)
beds_confint$upper_percent = ecdf(craigslist_df$rent)(beds_confint$upper)
beds_confint

#Creating function to associated ACS rent with bedrooms
#Using Gibbs sampling
```

```
lower_pct = beds_confint$lower_percent
upper_pct = beds_confint$upper_percent

gibbs_sampler = function(acs_beds, acs_rent, lower_pct, upper_pct){
  acs_rentbed = rep(0, length(acs_beds))
  t = 1
  while (t <= 100){
    for (i in 1:length(acs_beds)){
      beds_val = acs_beds[i]
      lower_cut = quantile(acs_rent, lower_pct[beds_val+1])
      upper_cut = quantile(acs_rent, upper_pct[beds_val+1])
      cut_df = data.frame("rent" = acs_rent) %>%
               filter(rent >= lower_cut, rent <= upper_cut)
      acs_rentbed[i] = sample(cut_df$rent, 1)
    }
    t = t + 1
  }
  return(acs_rentbed)
}

#Simulating Gibbs sample for pre pandemic ACS rent

set.seed(100)
acs_rentbed_prepan = gibbs_sampler(acs_beds_prepan, acs_rent_prepan,
                                   lower_pct, upper_pct)

#Simulating Gibbs sample for post pandemic ACS rent

set.seed(100)
```

```
acs_rentbed_postpan = gibbs_sampler(acs_beds_postpan, acs_rent_postpan,
                                    lower_pct, upper_pct)

#Distribution of ACS rent by bedroom pre pandemic

acs_sim_prepan = data.frame("bedrooms" = as.factor(acs_beds_prepan),
                            "rent" = acs_rentbed_prepan)

ggplot(acs_sim_prepan, aes(x=rent, fill=bedrooms)) +
geom_histogram(binwidth=300) +
ggtitle("Distribution of ACS Rent by Bedrooms Pre Pandemic")

#Distribution of ACS rent by bedroom post pandemic

acs_sim_postpan = data.frame("bedrooms" = as.factor(acs_beds_postpan),
                             "rent" = acs_rentbed_postpan)

ggplot(acs_sim_postpan, aes(x=rent, fill=bedrooms)) +
geom_histogram(binwidth=300) +
ggtitle("Distribution of ACS Rent by Bedrooms Post Pandemic")

#Distribution of ACS rent by bedroom overall

acs_sim_df = rbind.data.frame(acs_sim_prepan, acs_sim_postpan)

ggplot(acs_sim_df, aes(x=rent, fill=bedrooms)) +
geom_histogram(binwidth=300) +
ggtitle("Distribution of Rent by Bedrooms")
ggsave("acs_rent_bedrooms.png")
```

```
#Significant difference between Craigslist and ACS for 0 bedroom

cl_b0 = craigslist_df %>% filter(bedrooms == 0) %>% select(rent)
acs_b0 = acs_sim_df %>% filter(bedrooms == 0) %>% select(rent)

t.test(cl_b0$rent, acs_b0$rent)

ggplot() + geom_histogram(data=cl_b0, aes(x=rent, fill="Craigslist"),
                          binwidth=100, alpha=0.5) +
  geom_histogram(data=acs_b0, aes(x=rent, fill="ACS"),
                 binwidth=100, alpha=0.5) +
  ggtitle("Distribution of 0 Bedroom ACS vs Craigslist Rent") +
  scale_fill_discrete(name="source")

#Significant difference between Craigslist and ACS for 1 bedroom

cl_b1 = craigslist_df %>% filter(bedrooms == 1) %>% select(rent)
acs_b1 = acs_sim_df %>% filter(bedrooms == 1) %>% select(rent)

t.test(cl_b1$rent, acs_b1$rent)

ggplot() + geom_histogram(data=cl_b1, aes(x=rent, fill="Craigslist"),
                          binwidth=100, alpha=0.5) +
  geom_histogram(data=acs_b1, aes(x=rent, fill="ACS"),
                 binwidth=100, alpha=0.5) +
  ggtitle("Distribution of 1 Bedroom ACS vs Craigslist Rent") +
  scale_fill_discrete(name="source")
```

```
#Significant difference between Craigslist and ACS for 2 bedroom

cl_b2 = craigslist_df %>% filter(bedrooms == 2) %>% select(rent)
acs_b2 = acs_sim_df %>% filter(bedrooms == 2) %>% select(rent)

t.test(cl_b2$rent, acs_b2$rent)

ggplot() + geom_histogram(data=cl_b2, aes(x=rent, fill="Craigslist"),
                          binwidth=100, alpha=0.5) +
  geom_histogram(data=acs_b2, aes(x=rent, fill="ACS"),
                 binwidth=100, alpha=0.5) +
  ggtitle("Distribution of 2 Bedroom ACS vs Craigslist Rent") +
  scale_fill_discrete(name="source")

#Significant difference between Craigslist and ACS for 3 bedroom

cl_b3 = craigslist_df %>% filter(bedrooms == 3) %>% select(rent)
acs_b3 = acs_sim_df %>% filter(bedrooms == 3) %>% select(rent)

t.test(cl_b3$rent, acs_b3$rent)

ggplot() + geom_histogram(data=cl_b3, aes(x=rent, fill="Craigslist"),
                          binwidth=100, alpha=0.5) +
  geom_histogram(data=acs_b3, aes(x=rent, fill="ACS"),
                 binwidth=100, alpha=0.5) +
  ggtitle("Distribution of 3 Bedroom ACS vs Craigslist Rent") +
  scale_fill_discrete(name="source")

#Significant difference between Craigslist and ACS for 4+ bedroom
```

```
cl_b4 = craigslist_df %>% filter(bedrooms == 4 | bedrooms == 5)
        %>% select(rent)
acs_b4 = acs_sim_df %>% filter(bedrooms == 4 | bedrooms == 5)
        %>% select(rent)

t.test(cl_b4$rent, acs_b4$rent)

ggplot() + geom_histogram(data=cl_b4, aes(x=rent, fill="Craigslist"),
                          binwidth=100, alpha=0.5) +
  geom_histogram(data=acs_b4, aes(x=rent, fill="ACS"),
                 binwidth=100, alpha=0.5) +
  ggtitle("Distribution of 4+ Bedroom ACS vs Craigslist Rent") +
  scale_fill_discrete(name="source")
```

REFERENCES

[1] Boeing, G., & Waddell, P. (2017). New insights into rental housing markets across the United States: Web scraping and analyzing Craigslist rental listings. Journal of Planning Education and Research, 37(4), 457–476.
https://doi.org/10.1177/0739456X16664789

[2] Boeing, G., Wegmann, J., & Jiao, J. (2020). Rental housing spot markets: How online information exchanges can supplement transacted-rents data. Journal of Planning Education and Research.
https://doi.org/10.1177/0739456X20904435

[3] Explore census data. (2019). U.S. Census Bureau.
https://data.census.gov/cedsci/table?g=0500000US06037&d=ACS%205-Year%20Estimates%20Data%20Profiles&tid=ACSDP5Y2019.DP04&moe=false

[4] Explore census data. (2020). U.S. Census Bureau.
https://data.census.gov/cedsci/table?g=0500000US06037&d=ACS%205-Year%20Estimates%20Data%20Profiles&tid=ACSDP5Y2020.DP04&moe=false

[5] Myers, D., & Park, J. (2019). A constant quartile mismatch indicator of changing rental affordability in U.S. metropolitan areas, 2000 to 2016. Cityscape: A Journal of Policy Development and Research, 21(1), 139–176.
https://www.proquest.com/scholarly-journals/constant-quartile-mismatch-indicator-changing/docview/2261181754/se-2?accountid=14512

[6] Wayback machine. (n.d.). Internet Archive. Retrieved February 7, 2022, from
https://web.archive.org/web/*/https://losangeles.craigslist.org/search/apa