HW1-OLS_Regression

2023-10-19

$$SSE = \sum (y_i - \hat{y_i})^2 = \sum [y_i - (\beta_0 +$$

Introduction

Methods

Data Cleaning

The data set used in our analysis contains information from the 2000 US Census for Philadelphia, with neighborhood characteristic variables included for 1,720 block groups. Our analysis incorporates the following variables:

- POLY_ID: Census Block Group ID
- MEDHVAL: Median value of all owner occupied housing units
- PCBACHMORE: Proportion of residents in Block Group with at least a bachelor's degree
- PCTVACANT: Proportion of housing units that are vacant
- PCTSINGLES: Percent of housing units that are detached single family houses
- NBELPOV100: Number of households with incomes below 100% poverty level (i.e., number of households living in poverty)
- MEDHHINC: Median household income

The original data set had 1,816 block groups and was cleaned using the following methods, which reduced to the total number of observations to 1,720:

- Block groups where population < 40
- Block groups where there are no housing units
- Block groups where the median house value is lower than $10,000
- One North Philadelphia block group which had a very high median house value (over $800,000) and a very low median household income (less than $8,000)

In this analysis, we will examine the relationships between our dependent variable, MEDHVAL, and the predictors PCBACHMORE, NBELPOV100, PCTVACANT, and PCTSINGLES.

Exploratory Data Analysis

We will examine the summary statistics and distributions of the data set's variables, including the mean and standard deviation of our dependent variable and predictor variables.

As part of our exploratory data analysis, we will examine the Pearson correlations between the predictors. A Pearson correlation, denoted by "r", is a standardized measurement of the strength and direction of the linear relationship between two variables. The correlation between two variables is calculated using the following equation:

$$r = \frac{1}{n-1}\sum_{i=1}^{n}\left(\frac{x_i - x}{s_x}\right)\left(\frac{y_i - y}{s_x}\right)$$

The Pearson correlation value ranges between -1 to 1, with no units of measurement attached, and the observed variables are interchangeable between the x axis and y axis. A value of -1 represents a perfect negative linear relationship and a value of 1 represents a perfect positive linear relationship - in either case, points on a graph would appear in a straight line with either a negative or positive slope, respectively.

A Pearson correlation value of 0 indicates that there is no linear relationship between two variables. However, a different type of relationship can exist, such as an exponential or quadratic relationship, that the Pearson correlation does not measure.

Multiple Regression Analysis

Additional Analyses

Software

This report used the open source software R to conduct statistical analyses.

Results

Exploratory Results

```
data <- read.csv("Data/RegressionData.csv")
```

```
summary_stats_mean <- data %>%
  summarise(HEDVAL = mean(MEDHVAL),
            PCTBACHMOR = mean(PCTBACHMOR),
            NBELPOV100 = mean(NBELPOV100),
            PCTVACANT = mean(PCTVACANT),
            PCTSINGLE = mean(PCTSINGLES)) %>%
  gather(key = "variable", value = "mean")

summary_stats_sd <- data %>%
  summarise(HEDVAL = sd(MEDHVAL),
            PCTBACHMOR = sd(PCTBACHMOR),
            NBELPOV100 = sd(NBELPOV100),
            PCTVACANT = sd(PCTVACANT),
            PCTSINGLE = sd(PCTSINGLES)) %>%
  gather(key = "variable", value = "sd") %>%
  mutate(row_names = c('Median Houme Value of all occupied housing units','% of Individuals with Bachel
```

Variable	Mean	Standard Deviation
Median Houme Value of all occupied housing units	66287.733139	60006.075990
% of Individuals with Bachelor Degrees or Higher	16.081372	17.769558
# Households Living in Poverty	189.770930	164.318480
% of Vacant Houses	11.288529	9.628472
% of Single House Units	9.226473	13.249250

```
left_join(summary_stats_mean, summary_stats_sd, by='variable') %>%
  dplyr::select('row_names','mean','sd') %>%
  kbl(col.names = c('Variable','Mean','Standard Deviation')) %>%
  kable_classic()
```

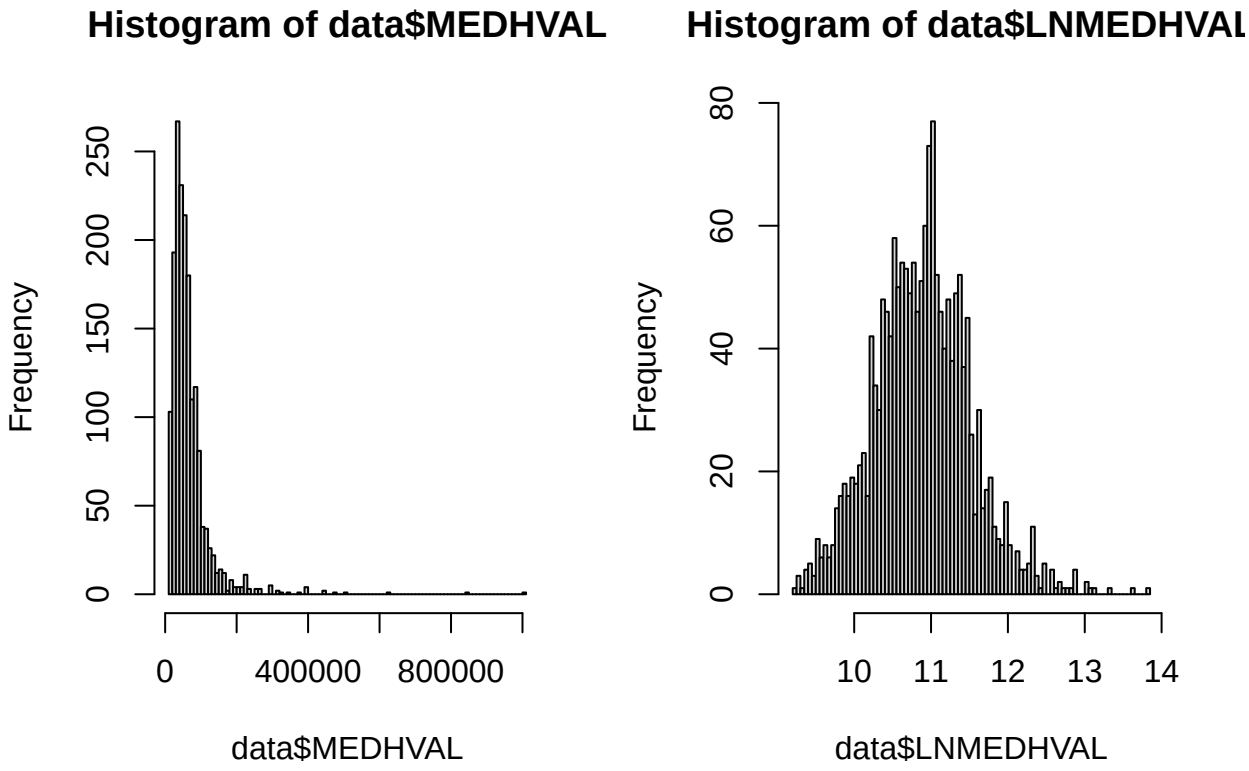
quick code that checks which variables have 0 values for logarithmic transformation

```
zero_columns <- apply(data, 2, function(col) any(col == 0))

variables_with_zero_values <- names(zero_columns[zero_columns])

cat("Columns with 0 values:", paste(variables_with_zero_values, collapse = ", "))
```

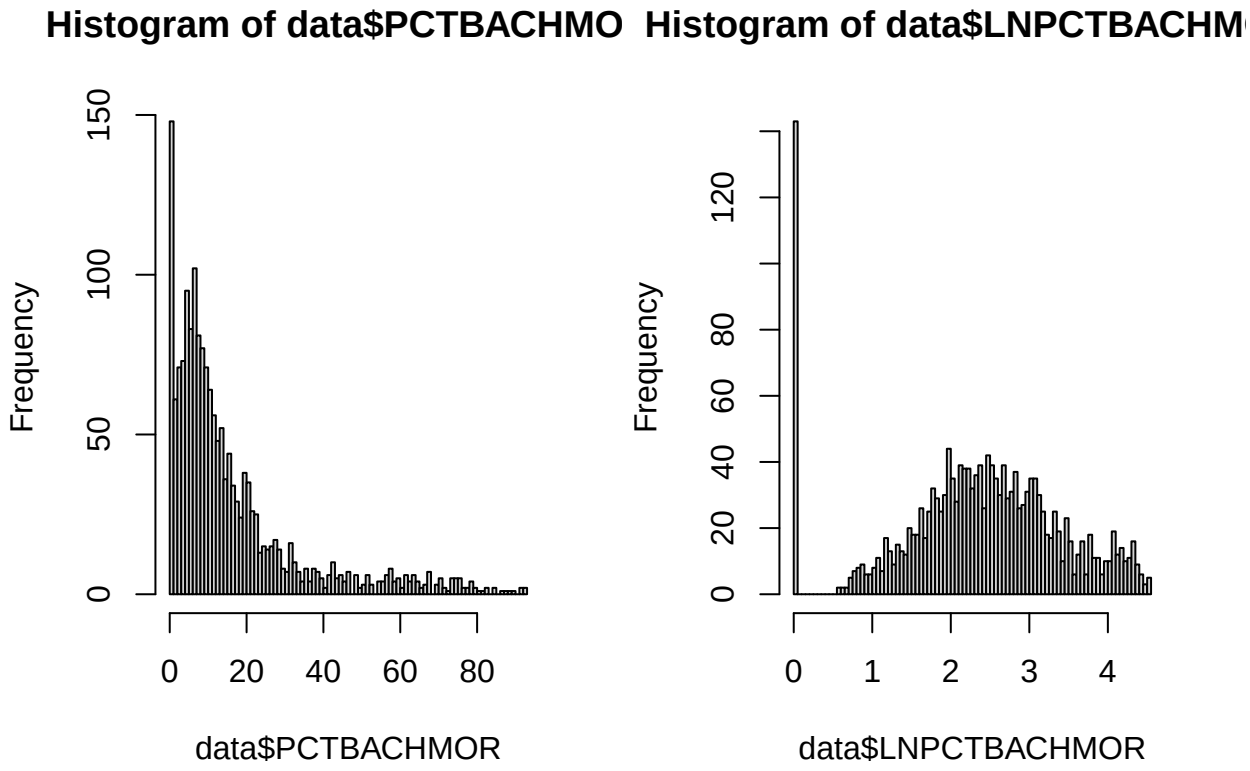
```
## Columns with 0 values: PCTBACHMOR, PCTVACANT, PCTSINGLES, NBELPOV100
```

```
data$LNMEDHVAL <- log(data$MEDHVAL)
data$LNPCTBACHMOR <- log(1 + data$PCTBACHMOR)
data$LNBELPOV100 <- log(1 + data$NBELPOV100) #Rename and add N to match original name?
data$LNPCTVACANT <- log(1 + data$PCTVACANT)
data$LNPCTSINGLES <- log(1 + data$PCTSINGLES)
```

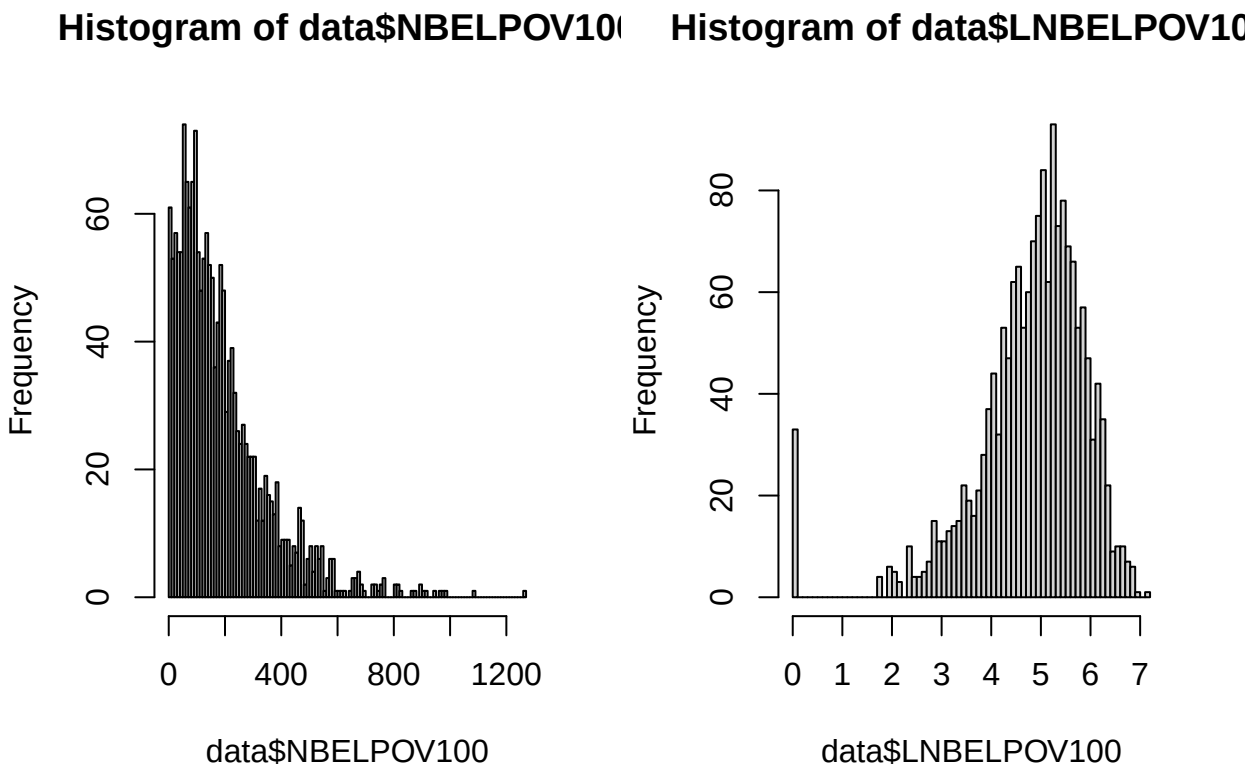
```
par(mfrow=c(1,2))
hist(data$MEDHVAL,breaks=100)
hist(data$LNMEDHVAL,breaks=100)
```



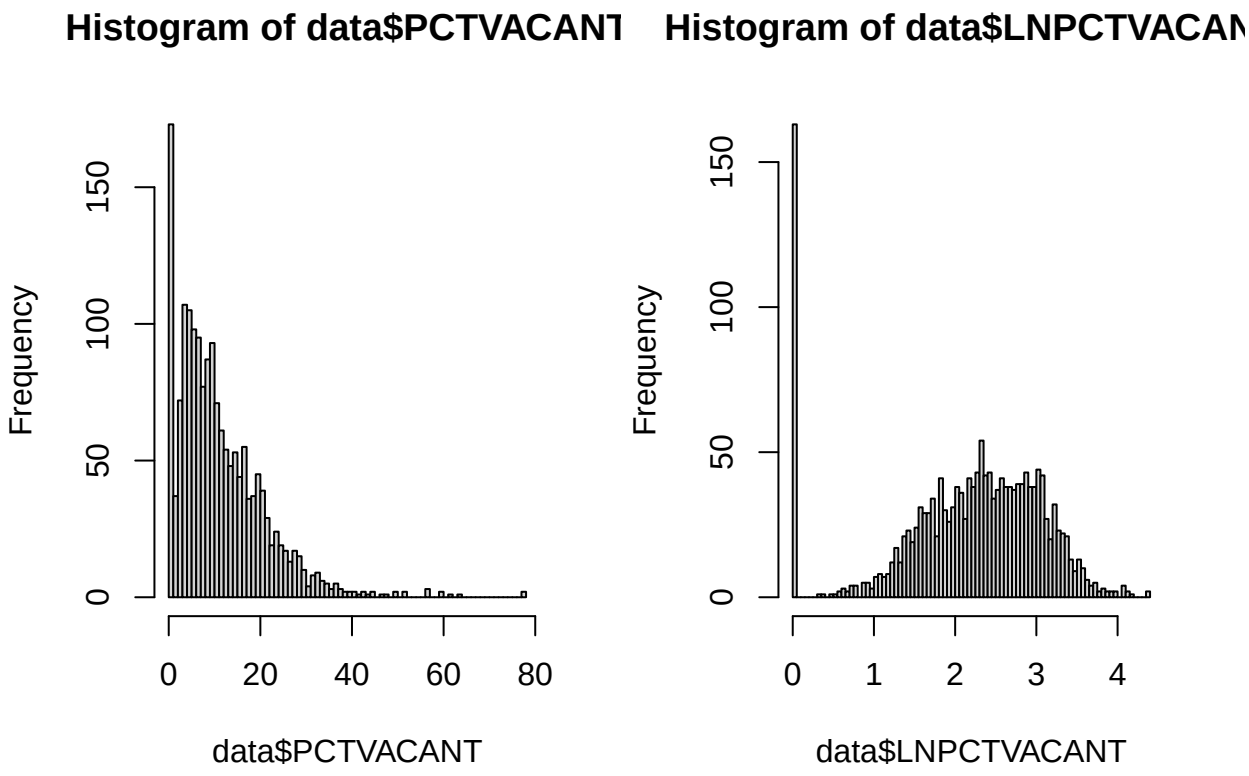

```
par(mfrow=c(1,2))
hist(data$PCTBACHMOR,breaks=100)
hist(data$LNPCTBACHMOR,breaks=100)
```



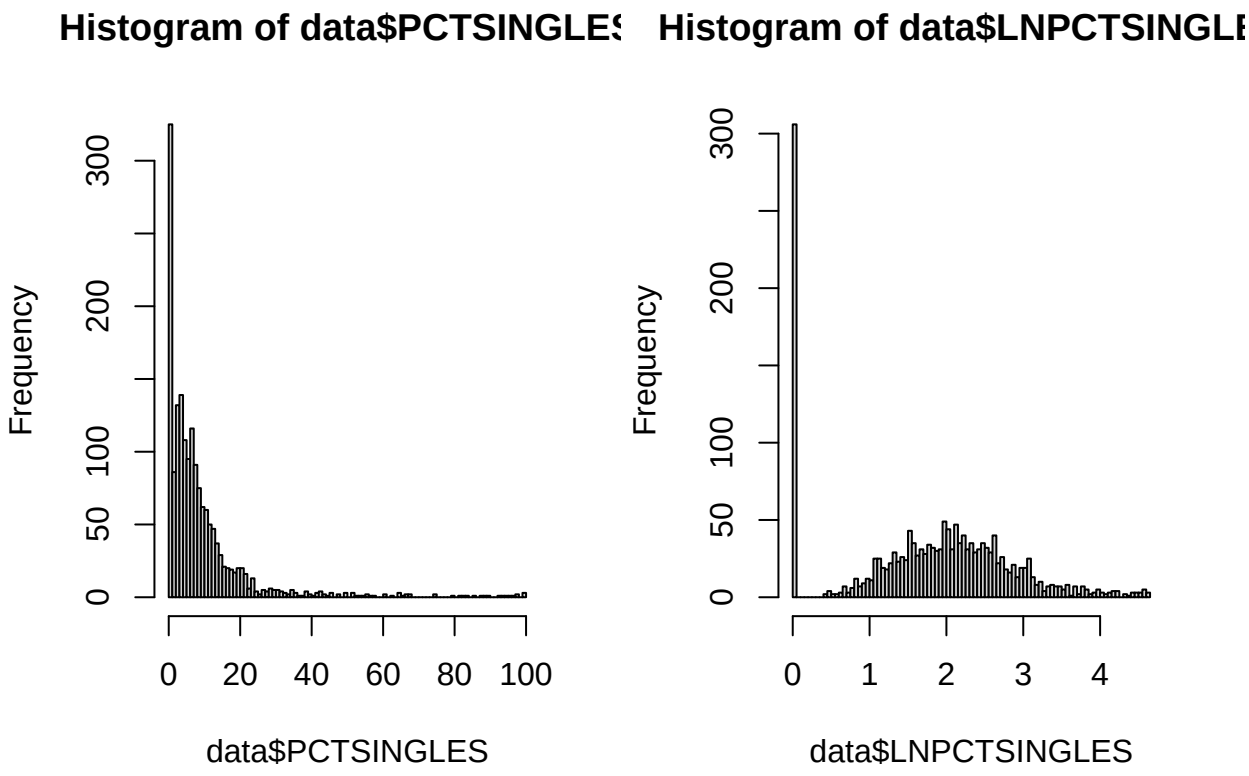

```
par(mfrow=c(1,2))
hist(data$NBELPOV100,breaks=100)
hist(data$LNBELPOV100,breaks=100)
```




```
par(mfrow=c(1,2))
hist(data$PCTVACANT,breaks=100)
hist(data$LNPCTVACANT,breaks=100)
```




```
par(mfrow=c(1,2))
hist(data$PCTSINGLES,breaks=100)
hist(data$LNPCTSINGLES,breaks=100)
```




```
# Change design
map <- st_read("Data/RegressionData.shp")
```

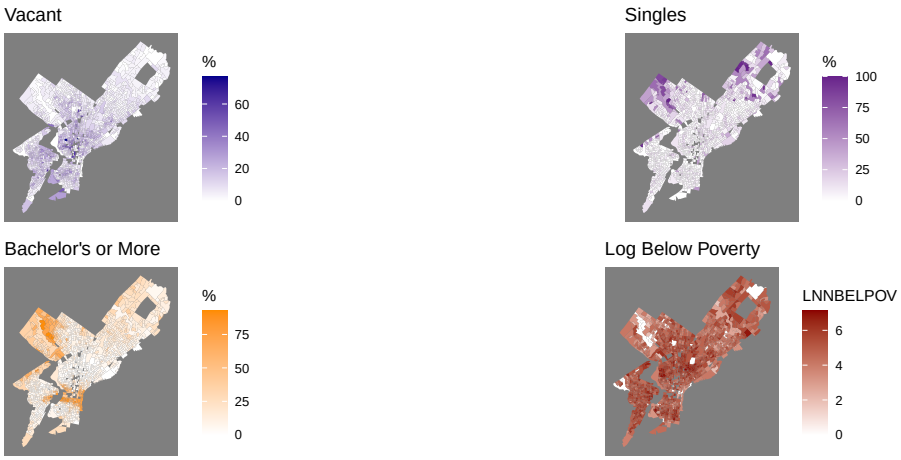
```
## Reading layer `RegressionData' from data source
##   `D:\Penn\Stats\MUSA500-HW1\Data\RegressionData.shp' using driver `ESRI Shapefile'
## Simple feature collection with 1720 features and 13 fields
## Geometry type: POLYGON
## Dimension:     XY
## Bounding box:  xmin: 2660605 ymin: 207610.6 xmax: 2750171 ymax: 304858.8
## CRS:           NA
```

```
ggplot() +
  geom_sf(data = map, aes(fill = LNMEDHVAL), color = NA) +
  scale_fill_gradient(low = "white",high = "darkseagreen4") +
  labs(title = "Log Median Home Value") +
  theme_dark() +
  theme(
    panel.grid.major = element_blank(),
    panel.grid.minor = element_blank(),
    axis.ticks = element_blank(),
    axis.text = element_blank()
    )
```

Log Median Home Value


```
# Change designs
pctvacant_map <- ggplot() +
  geom_sf(data = map, aes(fill = PCTVACANT), color = NA) +
  scale_fill_gradient(low = "white",high = "darkblue") +
  labs(title = "Vacant",
       fill = "%")+
  theme_dark() +
  theme(
    panel.grid.major = element_blank(),
    panel.grid.minor = element_blank(),
    axis.ticks = element_blank(),
    axis.text = element_blank()
    )

pctsingles_map <- ggplot() +
  geom_sf(data = map, aes(fill = PCTSINGLES), color = NA) +
  scale_fill_gradient(low = "white",high = "darkorchid4") +
  labs(title = "Singles",
       fill = "%")+
  theme_dark() +
  theme(
    panel.grid.major = element_blank(),
    panel.grid.minor = element_blank(),
    axis.ticks = element_blank(),
    axis.text = element_blank()
    )
```

```
pctbachmor_map <- ggplot() +
  geom_sf(data = map, aes(fill = PCTBACHMOR), color = NA) +
  scale_fill_gradient(low = "white",high = "darkorange") +
  labs(title = "Bachelor's or More",
       fill = "%")+
  theme_dark() +
  theme(
    panel.grid.major = element_blank(),
    panel.grid.minor = element_blank(),
    axis.ticks = element_blank(),
    axis.text = element_blank()
    )

lnnbelpov100_map <- ggplot() +
  geom_sf(data = map, aes(fill = LNNBELPOV), color = NA) +
  scale_fill_gradient(low = "white",high = "darkred") +
  labs(title = "Log Below Poverty")+
  theme_dark() +
  theme(
    panel.grid.major = element_blank(),
    panel.grid.minor = element_blank(),
    axis.ticks = element_blank(),
    axis.text = element_blank()
    )

grid.arrange(pctvacant_map, pctsingles_map, pctbachmor_map, lnnbelpov100_map)
```

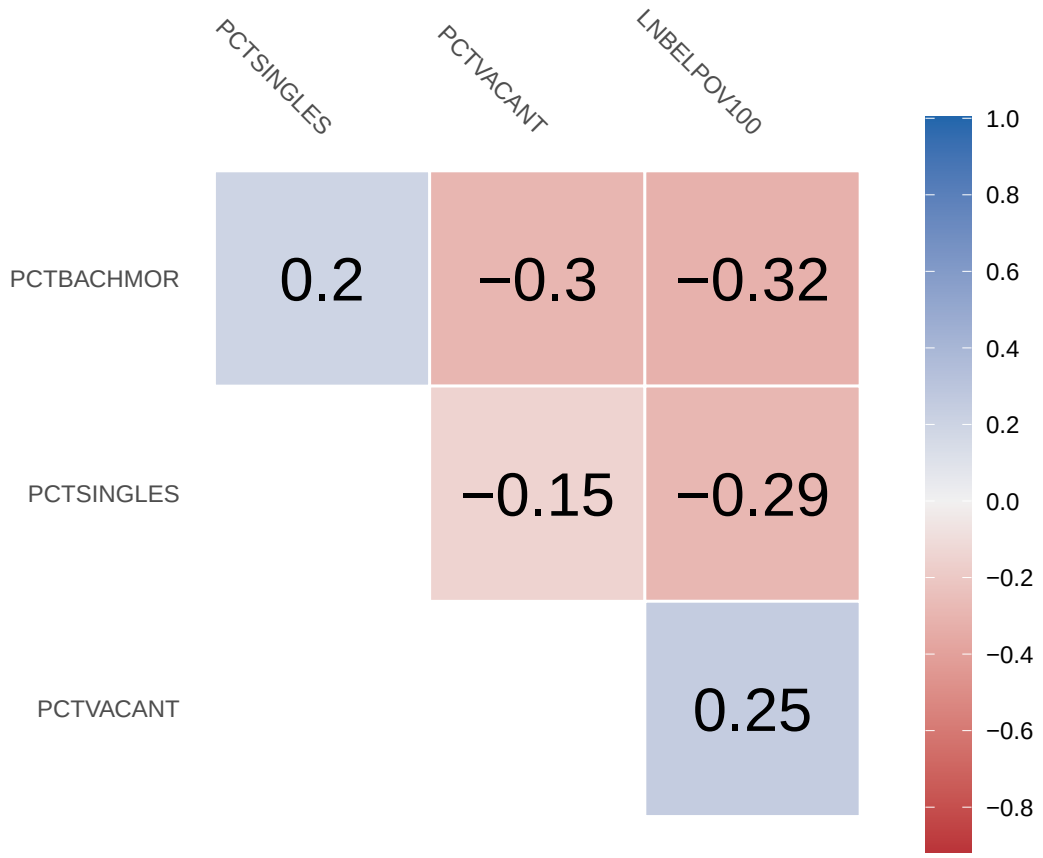


Pearson correlations

```
predictors <- data %>% dplyr::select(PCTBACHMOR, PCTVACANT, PCTSINGLES, LNBELPOV100)

predictors %>%
  correlate() %>%
```

```
  autoplot() +
  geom_text(aes(label = round(r,digits=2)),size = 8)
```

```
## Correlation computed with
## * Method: 'pearson'
## * Missing treated using: 'pairwise.complete.obs'
## Registered S3 methods overwritten by 'registry':
##   method               from
##   print.registry_field proxy
##   print.registry_entry proxy
```



Regression Analysis

```
## Regression Results

fit <-lm(LNMEDHVAL ~  PCTVACANT + PCTSINGLES + PCTBACHMOR + LNBELPOV100, data=data)

summary(fit)
```

```
##
## Call:
```

```
## lm(formula = LNMEDHVAL ~ PCTVACANT + PCTSINGLES + PCTBACHMOR +
##     LNBELPOV100, data = data)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -2.25825 -0.20391  0.03822  0.21744  2.24347
##
## Coefficients:
##               Estimate Std. Error t value             Pr(>|t|)
## (Intercept) 11.1137661  0.0465330 238.836 < 0.0000000000000002 ***
## PCTVACANT   -0.0191569  0.0009779 -19.590 < 0.0000000000000002 ***
## PCTSINGLES   0.0029769  0.0007032   4.234            0.0000242 ***
## PCTBACHMOR   0.0209098  0.0005432  38.494 < 0.0000000000000002 ***
## LNBELPOV100 -0.0789054  0.0084569  -9.330 < 0.0000000000000002 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.3665 on 1715 degrees of freedom
## Multiple R-squared:  0.6623, Adjusted R-squared:  0.6615
## F-statistic: 840.9 on 4 and 1715 DF,  p-value: < 0.00000000000000022
```

```
anova(fit)
```

```
## Analysis of Variance Table
##
## Response: LNMEDHVAL
##               Df  Sum Sq Mean Sq  F value                Pr(>F)
## PCTVACANT      1 180.392 180.392 1343.087 < 0.00000000000000022 ***
## PCTSINGLES     1  24.543  24.543  182.734 < 0.00000000000000022 ***
## PCTBACHMOR     1 235.118 235.118 1750.551 < 0.00000000000000022 ***
## LNBELPOV100    1  11.692  11.692   87.054 < 0.00000000000000022 ***
## Residuals   1715 230.344   0.134
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Regression Assumptions Checks

In this section, we will discuss testing model assumptions. We have already examined the variable distributions in a prior section.

Scatter Plots - Linear Relationships Between Variables

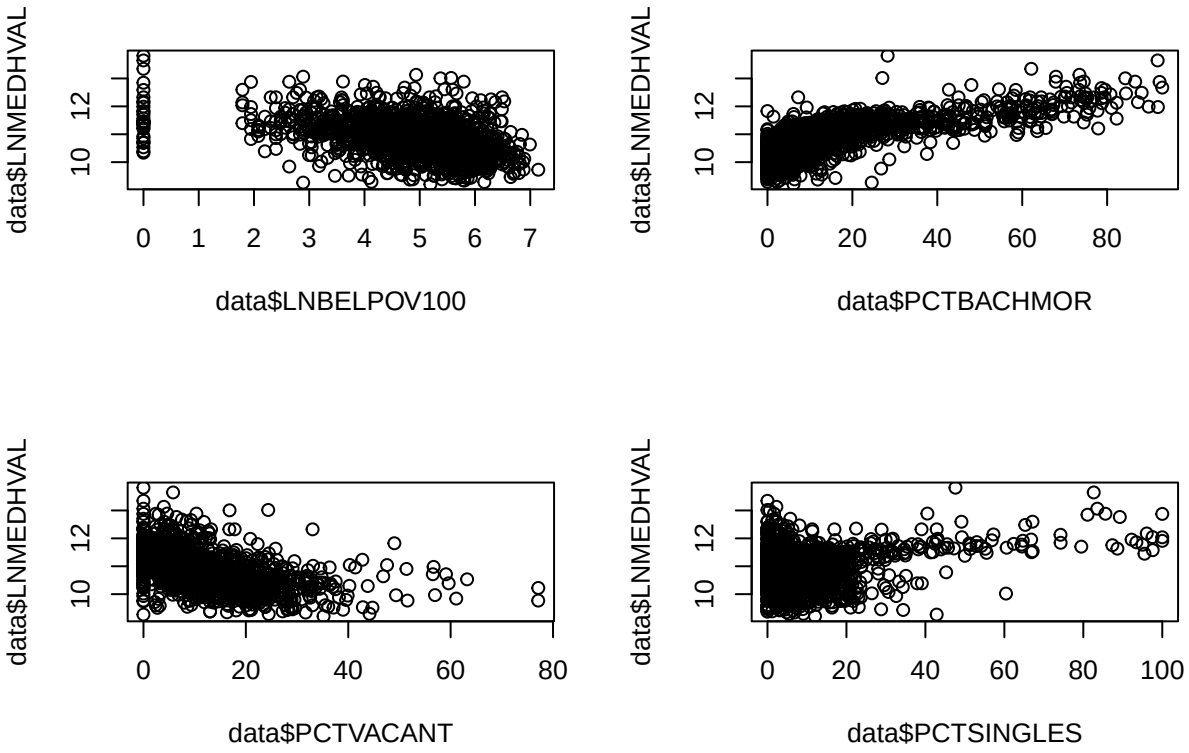
When running linear regressions, a core assumption is that there is a linear relationship between the dependent variable and each of the predictor variables. To check this assumption, we plot the dependent variable with each of the predictor variables in a scatter plot.

In cases where this assumption is not met, log transformations are often used. Based on the results of our variable distributions, we have already conducted log transformations for the dependent variable for median home value, now LNMEDHVAL, and for the predictor variable for number of households below poverty, now LNNBELPOV100.

The following scatter plots show the relationship between the dependent variable, LNMEDHVAL, and each the predictor variables LNNBELPOV100, PCTBACHMOR, PCTVACANT, and PCTSINGLES. There does

not appear to be a linear relationship between LNMEDHVAL and the predictor variables, even with log transformations used. The variables all appear heavily skewed - the relationship between LNNBELPOV and LNBELPOV100 appears to be negatively skewed, and the individual relationship between the other three predictors and LNMEDHVAL appears to be heavily positively skewed.

```
par(mfrow=c(2,2))
plot(data$LNBELPOV100, data$LNMEDHVAL)
plot(data$PCTBACHMOR,data$LNMEDHVAL)
plot(data$PCTVACANT, data$LNMEDHVAL)
plot(data$PCTSINGLES, data$LNMEDHVAL)
```



###Histogram of the standardized residuals

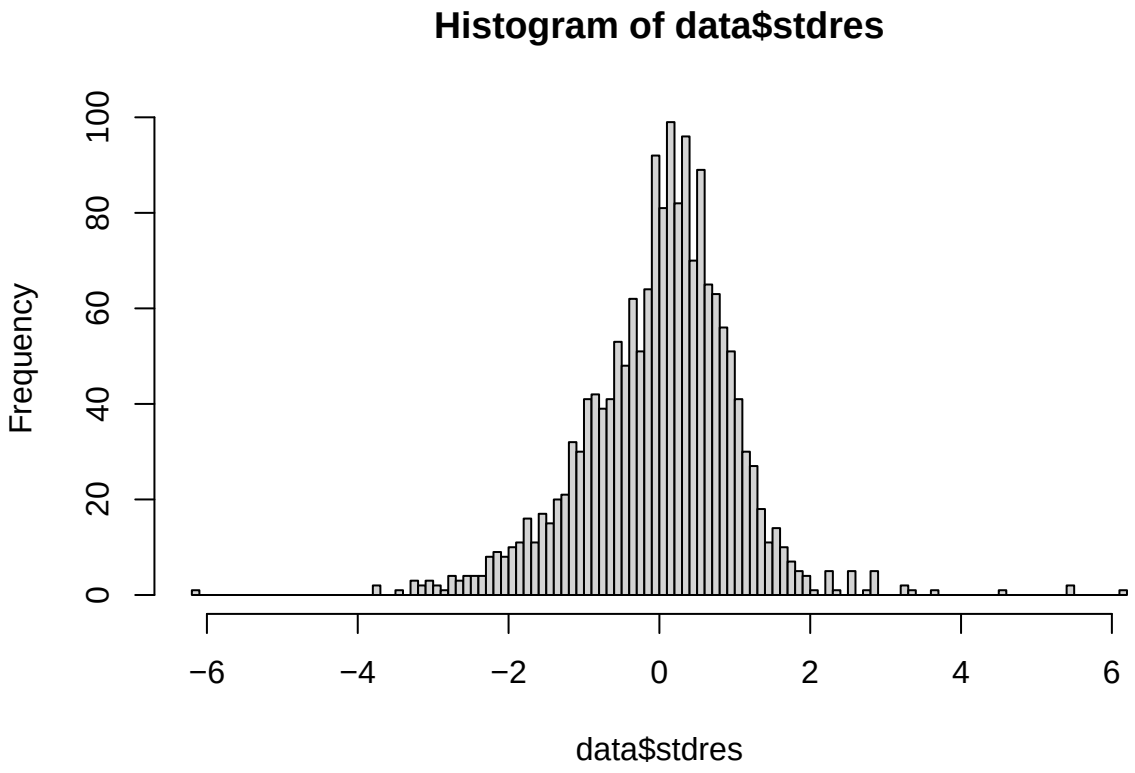
Another assumption when running linear regression is that regression residuals are distributed normally. However, this assumption of normality is not considered critical in a regression, especially for data sets with a large number of observations.

In order to compare residuals for different observations, we standardize the residuals through dividing a residual by its standard error. Standardizing allows us to observe how many standard deviations a residual is from our model's estimate

The following histogram of standardized residuals shows that residuals appear normally distributed.

```
#predicted values, residuals and standardized residuals

#Predicted values (y-hats)
data$predvals <- fitted(fit)
```

```
#Residuals
data$resids <- residuals(fit)
#Standardized Residuals
data$stdres <- rstandard(fit)

hist(data$stdres, breaks=100)
```



###Scatter Plot - Standardized Residual by Predicted Value

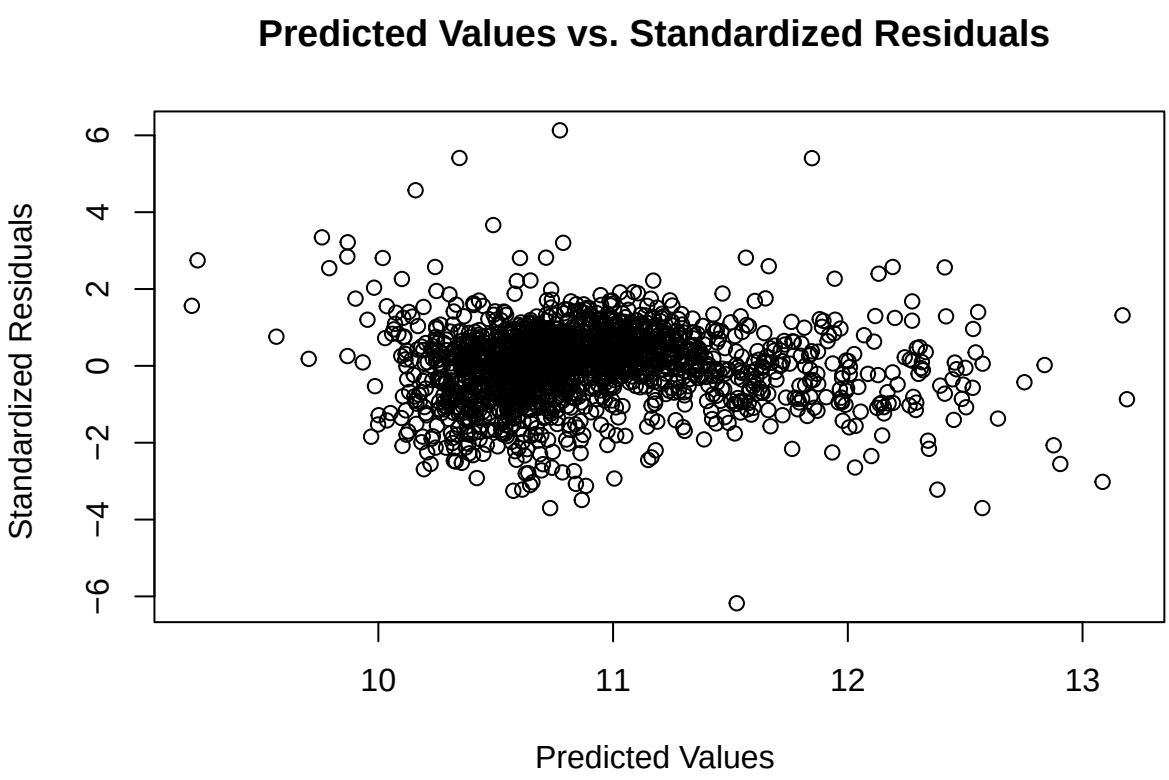
An additional core assumption of linear regression is that there is constant variance in residuals compared to the predicted values of the model - this relationship is referred to as homoscedastic. If non-constant variance is observed, the relationship is heteroscedastic.

Given that there multiple predictors, we can plot the standardized residuals of the model by our predicted values of LNMEDHVAL. The scatter plot of standardized residuals appears to show a slight heteroscedastic relationship, based on a small “bow-tie” shape present around the predicted value of 11.5.

Outliers also appear to be present based on our scatter plot - there are several positive standardized residuals above 4 standard deviations above 0 and at least one standardized residual beyond -6 standard deviation below 0.

The standardized residuals also appear to be heavily clustered around between the predicted values of about 10.5 to 11.5.

```
plot(data$predvals, data$stdres, xlab = "Predicted Values ", ylab = "Standardized Residuals ", main = "
```

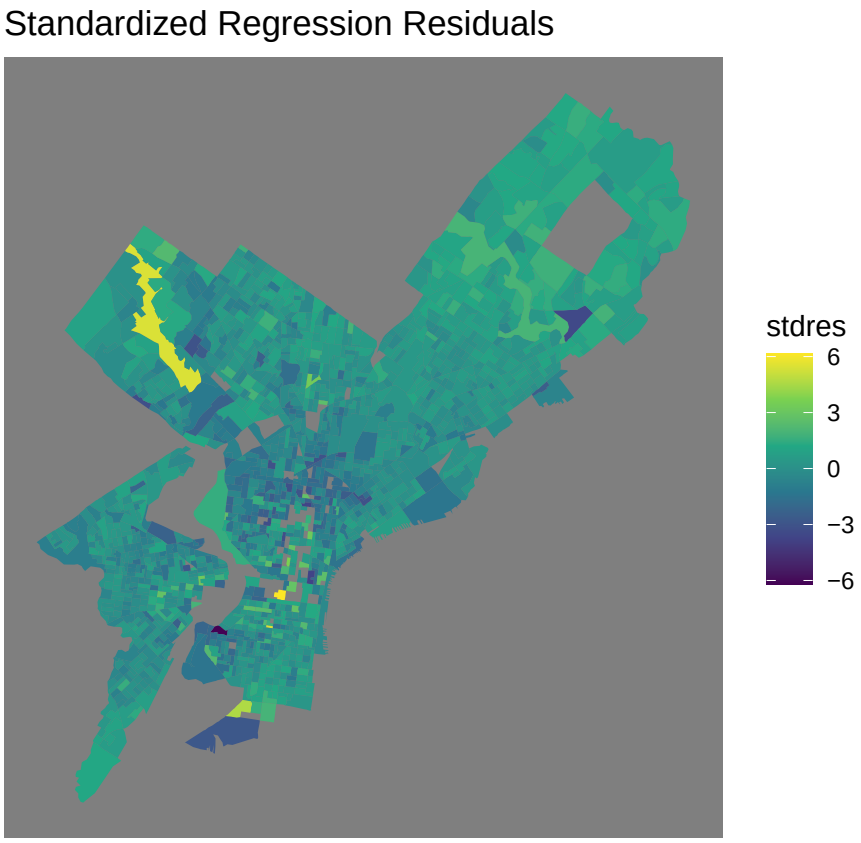


###Spatial Autocorrelation of Variables

Based on the maps of the dependent variable LNMEDHVAL and the predictor variables, we can estimate whether observations of each variable appear to show spatial autocorrelation - defined as observing the degree to which similar values cluster near each other. Our variables may appear to spatial autocorrelation

###Choropleth map of the standardized regression residuals

```
map2 <- cbind(map, data %>% dplyr::select(stdres))

ggplot()+
  geom_sf(data=map2, aes(fill = stdres), color = NA)+
  scale_fill_viridis_c()+
  labs(title = "Standardized Regression Residuals") +
  theme_dark() +
  theme(
    panel.grid.major = element_blank(),
    panel.grid.minor = element_blank(),
    axis.ticks = element_blank(),
    axis.text = element_blank()
    )
```



Additional Models

###Stepwise Regression

```
step <- stepAIC(fit, direction="both")
```

```
## Start:  AIC=-3448.07
## LNMEDHVAL ~ PCTVACANT + PCTSINGLES + PCTBACHMOR + LNBELPOV100
##
##                Df Sum of Sq    RSS     AIC
## <none>                      230.34 -3448.1
## - PCTSINGLES    1     2.407 232.75 -3432.2
## - LNBELPOV100   1    11.692 242.04 -3364.9
## - PCTVACANT     1    51.546 281.89 -3102.7
## - PCTBACHMOR    1   199.020 429.36 -2379.0
```

```
# display results
step$anova
```

```
## Stepwise Model Path
## Analysis of Deviance Table
##
## Initial Model:
```

```
## LNMEDHVAL ~ PCTVACANT + PCTSINGLES + PCTBACHMOR + LNBELPOV100
##
## Final Model:
## LNMEDHVAL ~ PCTVACANT + PCTSINGLES + PCTBACHMOR + LNBELPOV100
##
##
##   Step Df Deviance Resid. Df Resid. Dev       AIC
## 1                       1715   230.3435 -3448.073
```

###Cross-Validation

```
fit1 <- lm(LNMEDHVAL ~ PCTVACANT + PCTSINGLES + PCTBACHMOR + LNBELPOV100, data=data)
cv1 <- CVlm(data=data, fit1, m=5)
```

```
## Warning in CVlm(data = data, fit1, m = 5):
##
##  As there is >1 explanatory variable, cross-validation
##  predicted values for a fold are not a linear function
##  of corresponding overall predicted values.  Lines that
##  are shown for the different folds are approximate
```

```
mse1 <- attr(cv1, "ms")
rmse1 <- sqrt(mse1)                      #Obtaining RMSE for model 1
rmse1

fit2 <- lm(LNMEDHVAL ~ PCTVACANT + MEDHHINC, data=data)
cv2 <- CVlm(data=data, fit2, m=5)
```

```
## Warning in CVlm(data = data, fit2, m = 5):
##
##  As there is >1 explanatory variable, cross-validation
##  predicted values for a fold are not a linear function
##  of corresponding overall predicted values.  Lines that
##  are shown for the different folds are approximate
```

```
mse2 <- attr(cv2, "ms")
rmse2 <- sqrt(mse2)                      #Obtaining RMSE for model 2
rmse2

rmse_both <- cbind(rmse1, rmse2)

rmse_both %>% kbl() %>% kable_minimal(full_width = FALSE)
```

Additional Models

Discussion and Limitations