ASA Data Challenge Expo

Helping Communities During the COVID-19 Pandemic

Entry Details

- Event: 2021 ASA Data Challenge Expo (https://community.amstat.org/dataexpo/home)
- Name: Walter Yu
- Organization: Code for America (https://www.codeforamerica.org/)
- Section: Sacramento Brigade (https://codeforsacramento.org/)
- Code Repository: Github (https://github.com/walteryu/asa-2021)

Executive Summary

This project aims to help disadvantaged communities during the COVID-19 pandemic by answering the questions listed below through analysis of core and supplemental datasets. The intended audience are state/local governments, non-governmental organizations (NGOs) and volunteers which are able to provide aid and services to these communities.

1. Explore the relationship between socioeconomic features of the U.S. population and disadvantaged communities.
2. Identify disadvantaged communities based on their median household income. These communities are likely be more impacted by the COVID-19 pandemic and in need of public services.
3. Provide recommendations on helping these communities based on data analysis results.

Scope

This entry focuses on California communities to control its scope since several questions are being considered, and data analysis of all U.S. communities would expand the scope and length of this report. This limited scope provides for more detail and attention to be paid to analysis, documentation and recommendations.

Part 1: Overview

Methodology

This project and its analysis are designed to be interpretable, so it organizes data analysis steps into the following modules:

1. Overview: Outline approach, assumptions and data sources
2. Data Processing: Data preparation for analysis
3. Data Analysis: Model fit, coefficient interpretation and diagnostics
4. Recommendations: Document key findings from data analysis
5. Future Improvements: Possible improvements upon completing analysis

Assumptions

This entry makes the following assumptions:

1. Although the scope is limited to California communities, the methodology may be applied to other states since it is based on data extracted from the U.S. Census for the state/county level and do not contain any characteristics specific to California.
2. State and federal guidelines (https://www.hud.gov/topics/rental_assistance/phprog) typically define disadvantaged communities as being low-income, so median household income was used to identify such communities. In addition, state and federal guidelines typically define low income as 20% of median household income.
3. Data analysis was documented to be clear and easily interpretable, so linear regression and the Law of Parsimony (https://en.wikipedia.org/wiki/Occam%27s_razor) were applied whenever possible. The linear models are improved incrementally upon for interpretability.

Data Summary

This entry analyzes core and supplemental datasets from the data challenge problem statement (https://opportunity.census.gov/assets/files/covid-19-top-asa-problem-statement.pdf) as follows:

- Core Dataset: 2019 American Community Survey (ACS) Single-Year Estimates
- Supplemental Dataset: COVID-19 Data from the National Center for Health Statistics

Data was downloaded from portal websites as follows:

1. U.S. Census Website: Advanced search feature (https://data.census.gov/cedsci/advanced) was used to filter data in the following order: Surveys > Years > Geography > Topics.
2. U.S. Census COVID-19 Website: CA state data was downloaded from the categorical dataset search page (https://covid19.census.gov/).
3. National Center for Health Statistics (NCHS) Website: Death counts by county and race downloaded from their data portal (https://www.cdc.gov/nchs/covid19/index.htm).

Core Datasets

Datasets of interest were identified from the U.S. Census data portal and extracted using the advanced search tool. Table ID numbers are listed for reference.

1. 2019 American Community Survey (ACS) Single-Year Estimates - Language Spoken

- Description: PLACE OF BIRTH BY LANGUAGE SPOKEN AT HOME AND ABILITY TO SPEAK ENGLISH IN THE UNITED STATES
- Survey/Program: American Community Survey
- Years: 2019
- Table: B06007 (https://data.census.gov/cedsci/table?q=B06007&tid=ACSDT1Y2019.B06007)

2. 2019 American Community Survey (ACS) Single-Year Estimates - Household Income

- Description: HOUSEHOLD INCOME IN THE PAST 12 MONTHS (IN 2019 INFLATION-ADJUSTED DOLLARS)
- Survey/Program: American Community Survey
- Years: 2019
- Table: B19001 (https://data.census.gov/cedsci/table?text=B19001&tid=ACSDT1Y2019.B19001)

3. 2019 American Community Survey (ACS) Single-Year Estimates - Median Household Income

- Description: MEDIAN HOUSEHOLD INCOME IN THE PAST 12 MONTHS (IN 2019 INFLATION-ADJUSTED DOLLARS)
- Survey/Program: American Community Survey
- Years: 2019
- Table: B19013 (https://data.census.gov/cedsci/table?text=B19013&tid=ACSDT1Y2019.B19013)

4. 2019 American Community Survey (ACS) Single-Year Estimates - Poverty Status

- Description: POVERTY STATUS IN THE PAST 12 MONTHS BY SEX BY AGE
- Survey/Program: American Community Survey
- Years: 2019
- Table: B17001 (https://data.census.gov/cedsci/table?text=B17001&tid=ACSDT1Y2019.B17001)

5. 2019 American Community Survey (ACS) Single-Year Estimates - Housing Cost

- Description: MONTHLY HOUSING COSTS
- Survey/Program: American Community Survey
- Years: 2019
- Table: B25104 (https://data.census.gov/cedsci/table?text=B25104&tid=ACSDT1Y2019.B25104)

6. 2019 American Community Survey (ACS) Single-Year Estimates - Education Attainment

- Description: EDUCATIONAL ATTAINMENT FOR THE POPULATION 25 YEARS AND OVER
- Survey/Program: American Community Survey
- Years: 2019

- Table: B15003 (https://data.census.gov/cedsci/table?text=B15003&tid=ACSDT1Y2019.B15003)

7. 2019 American Community Survey (ACS) Single-Year Estimates - Commute Mode

- Description: MEANS OF TRANSPORTATION TO WORK BY AGE
- Survey/Program: American Community Survey
- Years: 2019
- Table: B08101 (https://data.census.gov/cedsci/table?text=B08101&tid=ACSDT1Y2019.B08101)

8. 2019 American Community Survey (ACS) Single-Year Estimates - Race

- Description: RACE
- Survey/Program: American Community Survey
- Years: 2019
- Table: B02001 (https://data.census.gov/cedsci/table?text=B02001&tid=ACSDT1Y2019.B02001)

Supplemental Datasets (U.S. Census)

Datasets of interest were identified from the U.S. Census COVID-19 data portal under the categorical dataset section.

1. U.S. Census - COVID-19 Demographic and Economic Resources

- Dataset: California Counties DP02 Social (https://covid19.census.gov/datasets/california-counties-dp02-social)

2. U.S. Census - COVID-19 Demographic and Economic Resources

- Dataset: California Counties DP03 Economic (https://covid19.census.gov/datasets/california-counties-dp03-economic)

3. U.S. Census - COVID-19 Demographic and Economic Resources

- Dataset: California Counties DP04 Housing (https://covid19.census.gov/datasets/california-counties-dp04-housing)

4. U.S. Census - COVID-19 Demographic and Economic Resources

- Dataset: California Counties DP05 Demographic (https://covid19.census.gov/datasets/california-counties-dp05-demographic)

5. U.S. Census - COVID-19 Demographic and Economic Resources

- Dataset: Household Pulse Survey Public Use File (https://www.census.gov/programs-surveys/household-pulse-survey/datasets.html)

6. U.S. Census - COVID-19 Demographic and Economic Resources

- Dataset: COVID-19 Case Surveillance Public Use Data (https://data.cdc.gov/Case-Surveillance/COVID-19-Case-Surveillance-Public-Use-Data/vbim-akqf)

Supplemental Datasets (NCHS)

Datasets of interest were identified from the National Center for Health Statistics (NCHS) data portal for COVID-related mortality count by California county to evaluate impacts by the pandemic.

1. NCHS - COVID-19 Data from the National Center for Health Statistics

- Dataset: Provisional COVID-19 Death Counts by County and Race (https://data.cdc.gov/NCHS/Provisional-COVID-19-Death-Counts-by-County-and-Ra/k8wy-p9cg)

2. NCHS - COVID-19 Data from the National Center for Health Statistics

- Dataset: Provisional COVID-19 Death Counts in the United States by County (https://data.cdc.gov/NCHS/Provisional-COVID-19-Death-Counts-in-the-United-St/kn79-hsxy)

Geospatial Datasets

Datasets of interest were identified from the U.S. Census COVID-19 data portal under the categorical dataset section. They were not used during the analysis due to time and scope constraints so are documented for future use.

1. U.S. Census - COVID-19 Demographic and Economic Resources

- Description: American Community Survey (ACS) about household income ranges and cutoffs and Poverty Status.
- These are 5-year estimates shown by state and county boundaries.
- Link: Dataset (https://uscensus.maps.arcgis.com/home/item.html?id=b2ba19b4cce04a9796d9cdeecaba2f18)

2. U.S. Census - COVID-19 Demographic and Economic Resources

- Description: American Community Survey (ACS) about household income ranges and cutoffs.
- These are 5-year estimates shown by county, and state boundaries.
- Link: Dataset (https://uscensus.maps.arcgis.com/home/item.html?id=0fbb1571e5b6458f941580d1d64a6693)

3. U.S. Census - COVID-19 Demographic and Economic Resources

- Description: American Community Survey (ACS) about total population count by sex and age group.
- These are 5-year estimates shown by state and county boundaries.
- Link: Dataset (https://uscensus.maps.arcgis.com/home/item.html?id=eab0f44ba5184c609175caa7ae317f0c)

Part 2: Data Processing

Methodology

> Core and supplemental datasets of interested were processed and joined as listed below prior to model fit and data visualization.

This module completes the tasks listed below; however, all text output, warnings and messages are silenced to minimize report length so please check the code repository for details about implementation.

1. Import csv files as dataframes
2. Remove first record from each dataframe (header data)
3. Import selected columns from full dataset
4. Relabel selected columns from full dataset
5. Total population below 20% of median household income
6. Join tables together into single dataframe

```
## [1] "80% of Non-Metropolitan County Median Income (2020 CA HUD Guidelines):"
```

```
## [1] 56560
```

Part 3: Data Analysis

Methodology

> Processed data was fit a linear regression model to identify key attributes associated with disadvantaged communities and those communities most impacted by the pandemic.

Data analysis was conducted as follows:

1. Fit 2019 ACS data tables into separate linear regression models to identify which had the best fit and association with population count below 80% median household income. Model summary was evaluated to identify model fit, p-values and r-squared values.
2. Interpret model fit, coefficient interpretation and model diagnostics to refine model by selecting the best data features and combining into its own linear regression model. Diagnostic plots were reviewed for possible outlier points with high leverage.

3. Identify counties with most residents below 80% median household income and highest COVID-19 death counts by analyzing the 2019 ACS and NCHS COVID-19 mortality count data. Counties with highest count of residents in both datasets were identified and compared.

2019 ACS - Commute Mode

> Commute mode has a positive relationship with median household income as verified with the model fit and low p-value. Features were split into two groups for analysis: car-based modes and transit-based modes.

Features within each group were fit individually into their own model. The car-only mode had the best fit in the car-based modes. Walking had the best good fit with transit-based modes. This finding implies that certain commute modes (i.e. car-only and walking) have a better relationship to income than other ones (i.e. carpool and remote work).

The plot shows count of residents below 80% of median income and residents who walks to work. The regression line indicates a negative relationship, so counties with higher low income population had higher population count which walks to work.

The pandemic may impact transit and resulted in additional remote work for the work force. One possible area for analysis would be to determine the level of reliance of residents in disadvantage communities on these modes to evaluate their true impacts.

```
# features:
# B08101_009E: Car, truck, or van - drove alone
# B08101_017E: Car, truck, or van - carpooled
# B08101_025E: Public transportation (excluding taxicab)
# B08101_033E: Walked
# B08101_041E: Taxicab, motorcycle, bicycle, or other means
# B08101_049E: Worked from home

# model fit for transit-based commute modes
# linear regression - transit

# note: fit to log scale
# source: https://stats.stackexchange.com/questions/176595/simple-log-regression-model-in-r
fit_acs19_commute_walk <- lm(
    log(hh_median) ~ log(as.numeric(commute_walk)),
    data=acs19_nofactor
)
round(summary(fit_acs19_commute_walk)$coeff, 6)
```

```
##                               Estimate Std. Error   t value Pr(>|t|)
## (Intercept)                  13.017058   0.609248 21.365794 0.000000
## log(as.numeric(commute_walk)) -0.694916   0.210843 -3.295893 0.002062
```

```
# linear regression - transit + walk
fit_acs19_commute_transit_walk <- lm(
    log(hh_median) ~ log(as.numeric(commute_transit)) +
    log(as.numeric(commute_walk)),
    data=acs19_nofactor
)
round(summary(fit_acs19_commute_transit_walk)$coeff, 6)
```

```
##                                 Estimate Std. Error   t value Pr(>|t|)
## (Intercept)                    12.841897   0.680270 18.877654 0.000000
## log(as.numeric(commute_transit))  0.159651   0.266569  0.598910 0.552696
## log(as.numeric(commute_walk))    -0.791264   0.266569 -2.968327 0.005097
```

```
# linear regression - transit + walk + remote
fit_acs19_commute_transit_remote <- lm(
    log(hh_median) ~ log(as.numeric(commute_transit)) +
    log(as.numeric(commute_walk)) +
    log(as.numeric(commute_remote)),
    data=acs19_nofactor
)
round(summary(fit_acs19_commute_transit_remote)$coeff, 6)
```

```
##                                  Estimate Std. Error   t value Pr(>|t|)
## (Intercept)                     14.333538   0.854612 16.771975 0.000000
## log(as.numeric(commute_transit))  0.132598   0.248947  0.532636 0.597387
## log(as.numeric(commute_walk))    -0.785745   0.248739 -3.158911 0.003101
## log(as.numeric(commute_remote))  -0.517542   0.198546 -2.606665 0.012989
```

```
# variance analysis
anova(
    fit_acs19_commute_walk,
    fit_acs19_commute_transit_walk,
    fit_acs19_commute_transit_remote
)
```

```
## Analysis of Variance Table
##
## Model 1: log(hh_median) ~ log(as.numeric(commute_walk))
## Model 2: log(hh_median) ~ log(as.numeric(commute_transit)) + log(as.numeric(commute_walk))
## Model 3: log(hh_median) ~ log(as.numeric(commute_transit)) + log(as.numeric(commute_walk)) +
##     log(as.numeric(commute_remote))
##   Res.Df    RSS Df Sum of Sq      F  Pr(>F)
## 1     40 51.771
## 2     39 51.299  1    0.4718 0.4120 0.52482
## 3     38 43.518  1    7.7814 6.7947 0.01299 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
# model fit for car-based commute modes
# linear regression - car
fit_acs19_commute_car <- lm(
    log(hh_median) ~ log(as.numeric(commute_car)),
    data=acs19_nofactor
)
round(summary(fit_acs19_commute_car)$coeff, 6)
```

```
##                               Estimate Std. Error   t value Pr(>|t|)
## (Intercept)                  13.370954   0.575033 23.252496 0.000000
## log(as.numeric(commute_car)) -0.822813   0.199002 -4.134691 0.000177
```

```
# linear regression - car + carpool
fit_acs19_commute_car_carpool <- lm(
    log(hh_median) ~ log(as.numeric(commute_car)) +
    log(as.numeric(commute_carpool)),
    data=acs19_nofactor
)
round(summary(fit_acs19_commute_car_carpool)$coeff, 6)
```

```
##                                  Estimate Std. Error   t value Pr(>|t|)
## (Intercept)                     13.820132   0.717992 19.248301 0.000000
## log(as.numeric(commute_car))     -0.771647   0.204753 -3.768680 0.000543
## log(as.numeric(commute_carpool)) -0.213499   0.204753 -1.042714 0.303501
```

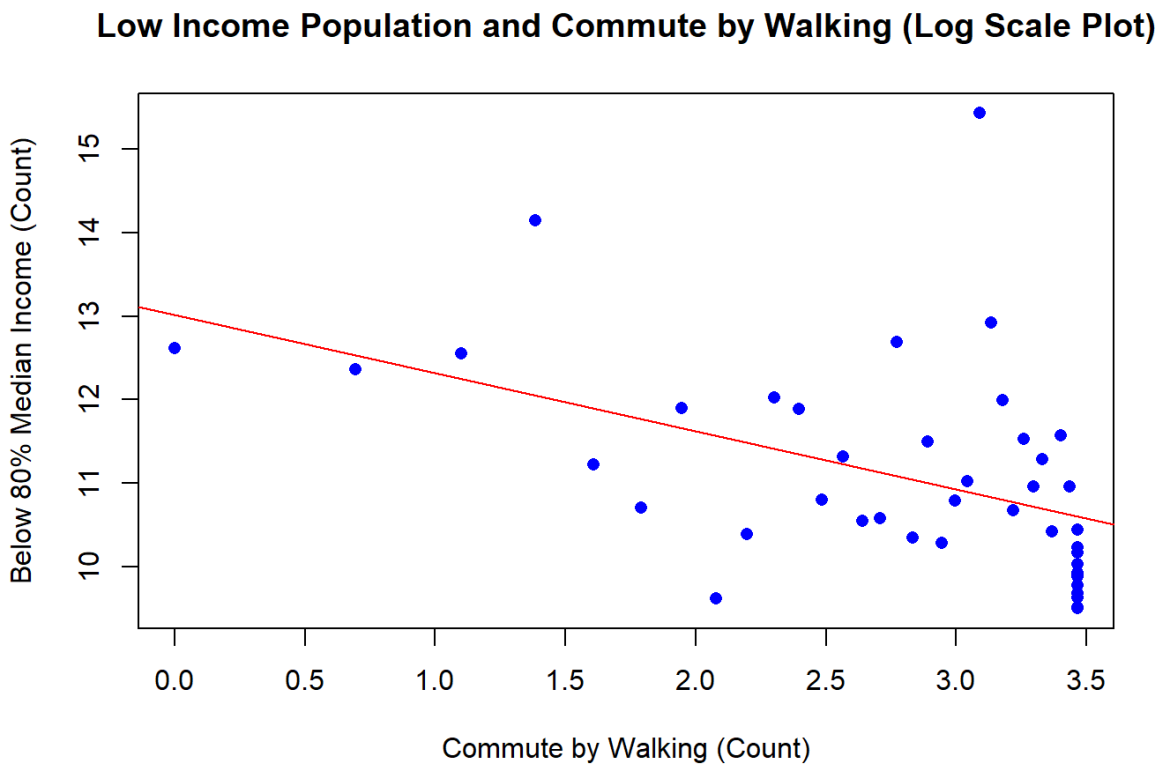
```
# linear regression - car + carpool + other
fit_acs19_commute_car_other <- lm(
    log(hh_median) ~ log(as.numeric(commute_car)) +
    log(as.numeric(commute_carpool)) +
    log(as.numeric(commute_other)),
    data=acs19_nofactor
)
round(summary(fit_acs19_commute_car_other)$coeff, 6)
```

```
##                                  Estimate Std. Error   t value Pr(>|t|)
## (Intercept)                     14.795451   0.817344 18.101857 0.000000
## log(as.numeric(commute_car))     -0.708869   0.197548 -3.588339 0.000937
## log(as.numeric(commute_carpool)) -0.208179   0.195473 -1.065004 0.293595
## log(as.numeric(commute_other))   -0.420576   0.192016 -2.190313 0.034708
```

```
# variance analysis
anova(
    fit_acs19_commute_car,
    fit_acs19_commute_car_carpool,
    fit_acs19_commute_car_other
)
```

```
## Analysis of Variance Table
##
## Model 1: log(hh_median) ~ log(as.numeric(commute_car))
## Model 2: log(hh_median) ~ log(as.numeric(commute_car)) + log(as.numeric(commute_carpool))
## Model 3: log(hh_median) ~ log(as.numeric(commute_car)) + log(as.numeric(commute_carpool)) +
##     log(as.numeric(commute_other))
##   Res.Df    RSS Df Sum of Sq      F  Pr(>F)
## 1     40 46.120
## 2     39 44.869  1    1.2509 1.1931 0.28158
## 3     38 39.839  1    5.0297 4.7975 0.03471 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

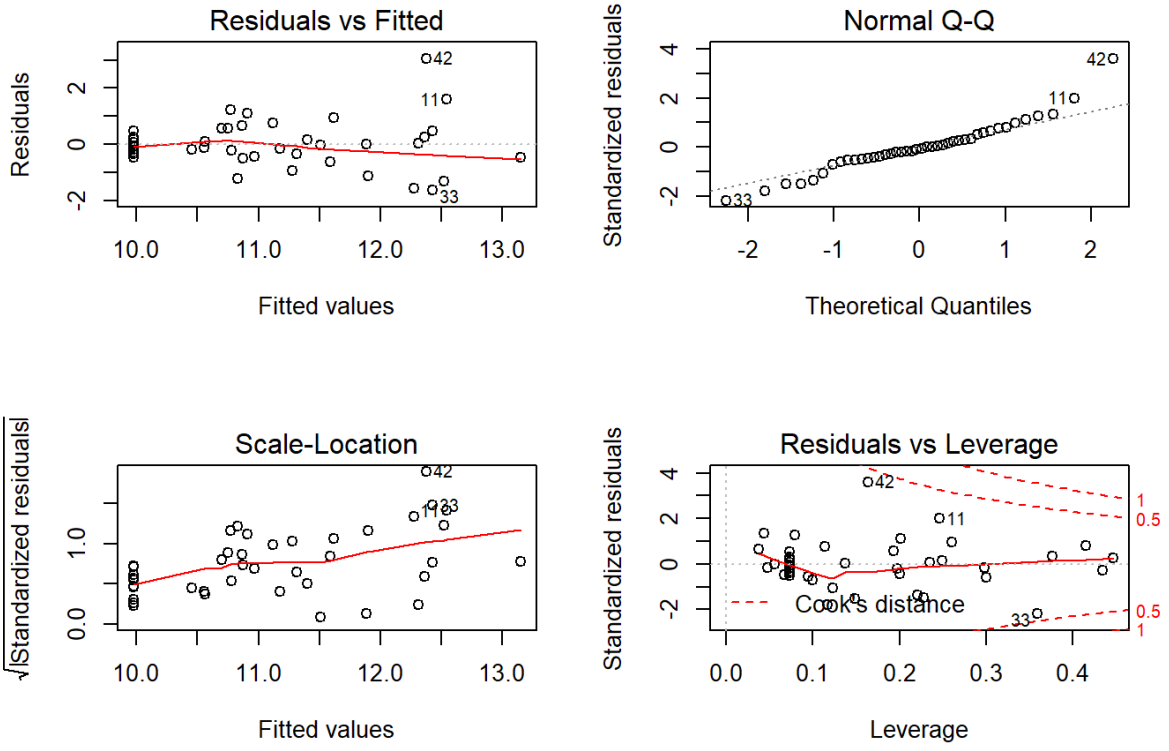
```
# visualize income and transit variables
# https://www.theanalysisfactor.com/linear-models-r-plotting-regression-lines/
plot(log(hh_median) ~ log(as.numeric(commute_walk)),
    main="Low Income Population and Commute by Walking (Log Scale Plot)",
    xlab="Commute by Walking (Count)",
    ylab="Below 80% Median Income (Count)",
    pch=16,
    col="blue",
    data=acs19_nofactor
)
# trend line plot
abline(
    fit_acs19_commute_walk,
    col="red"
)
```




```
# linear regression - all
fit_acs19_commute_all <- lm(
    log(hh_median) ~ log(as.numeric(commute_transit)) +
    log(as.numeric(commute_walk)) +
    log(as.numeric(commute_remote)) +
    log(as.numeric(commute_car)) +
    log(as.numeric(commute_carpool)) +
    log(as.numeric(commute_other)),
    data=acs19_nofactor
)
summary(fit_acs19_commute_all)
```

```
##
## Call:
## lm(formula = log(hh_median) ~ log(as.numeric(commute_transit)) +
##     log(as.numeric(commute_walk)) + log(as.numeric(commute_remote)) +
##     log(as.numeric(commute_car)) + log(as.numeric(commute_carpool)) +
##     log(as.numeric(commute_other)), data = acs19_nofactor)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -1.62281 -0.44430 -0.06592  0.41940  3.04833
##
## Coefficients:
##                                 Estimate Std. Error t value Pr(>|t|)
## (Intercept)                     15.53029    0.82891  18.736  < 2e-16 ***
## log(as.numeric(commute_transit))  0.30254    0.22424   1.349  0.18594
## log(as.numeric(commute_walk))    -0.72943    0.22508  -3.241  0.00262 **
## log(as.numeric(commute_remote))  -0.20677    0.20308  -1.018  0.31557
## log(as.numeric(commute_car))     -0.67763    0.19537  -3.468  0.00141 **
## log(as.numeric(commute_carpool)) -0.02396    0.19520  -0.123  0.90301
## log(as.numeric(commute_other))   -0.26794    0.19554  -1.370  0.17934
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.9275 on 35 degrees of freedom
## Multiple R-squared:  0.5427, Adjusted R-squared:  0.4643
## F-statistic: 6.922 on 6 and 35 DF,  p-value: 6.588e-05
```

```
# diagnostic plot
par(mfrow=c(2,2))
plot(fit_acs19_commute_all)
```



2019 ACS - Monthly Housing Cost

> Monthly housing cost has a positive relationship with median household income as verified with the model fit and low p-value. The model fit revealed an unexpected result that counties with larger low income populations also had larger populations with high housing cost (above $3k/month). However, this finding may be accounted by counties which are large metro areas tend to have large populations with both categories.

The model fit demonstrated that monthly housing cost is associated with median household income. The model did not include lower housing cost features due to their large number, values with little or not rent and not being adjusted for cost of living (i.e. some counties may have significantly lower rents). As a result, only higher housing cost features were included in the model.

The plot shows count of residents below 80% of median income and residents whose monthly housing income is above $3k per month. The regression line indicates a positive relationship, so counties with higher low income population also had higher population count with high monthly housing cost. This finding is an unexpected but may be accounted by counties which are large metro areas tend to have large populations with both categories.

The pandemic may impact the ability of residents within disadvantaged communities to cover their monthly housing costs. One possible area for analysis would be to determine the ability of residents in disadvantage communities to cover these costs to evaluate their true impacts.

```
# features:
# B25104_002E: Less than $100
# B25104_003E: 100 to $199
# B25104_004E: 200 to $299
# B25104_005E: 300 to $399
# B25104_006E: 400 to $499
# B25104_007E: 500 to $599
# B25104_008E: 600 to $699
# B25104_009E: 700 to $799
# B25104_010E: 800 to $899
# B25104_011E: 900 to $999
# B25104_012E: 1,000 to $1,499
# B25104_013E: 1,500 to $1,999
# B25104_014E: 2,000 to $2,499
# B25104_015E: 2,500 to $2,999
# B25104_016E: 3,000 or more
# B25104_017E: No cash rent

# linear regression model fit - housing cost
# note: fit to log scale
# source: https://stats.stackexchange.com/questions/176595/simple-log-regression-model-in-r
fit_acs19_housing_3000 <- lm(
    log(hh_median) ~ log(housing_more_3000),
    data=acs19_nofactor
)
round(summary(fit_acs19_housing_3000)$coeff, 6)
```

```
##                       Estimate Std. Error  t value Pr(>|t|)
## (Intercept)           6.486015   0.522898 12.40399        0
## log(housing_more_3000) 0.487520   0.053999  9.02833        0
```

```
fit_acs19_housing_2500 <- lm(
    log(hh_median) ~ log(housing_more_3000) +
    log(housing_2500),
    data=acs19_nofactor
)
round(summary(fit_acs19_housing_2500)$coeff, 6)
```

```
##                          Estimate Std. Error   t value Pr(>|t|)
## (Intercept)              4.313968   0.538512  8.010899 0.000000
## log(housing_more_3000) -0.707364   0.209478 -3.376794 0.001672
## log(housing_2500)       1.460282   0.251287  5.811221 0.000001
```

```
fit_acs19_housing_2000 <- lm(
    log(hh_median) ~ log(housing_more_3000) +
    log(housing_2500) +
    log(housing_2000),
    data=acs19_nofactor
)
round(summary(fit_acs19_housing_2000)$coeff, 6)
```

```
##                          Estimate Std. Error   t value Pr(>|t|)
## (Intercept)              1.921405   0.413208  4.649968 0.000039
## log(housing_more_3000) -0.331642   0.128649 -2.577886 0.013943
## log(housing_2500)      -0.280374   0.244986 -1.144446 0.259596
## log(housing_2000)       1.517933   0.171781  8.836425 0.000000
```

```
# model fit summary
summary(fit_acs19_housing_2000)
```

```
##
## Call:
## lm(formula = log(hh_median) ~ log(housing_more_3000) + log(housing_2500) +
##     log(housing_2000), data = acs19_nofactor)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.63096 -0.14359 -0.01924  0.18881  0.63194
##
## Coefficients:
##                         Estimate Std. Error t value Pr(>|t|)
## (Intercept)               1.9214     0.4132   4.650 3.94e-05 ***
## log(housing_more_3000)  -0.3316     0.1286  -2.578   0.0139 *
## log(housing_2500)       -0.2804     0.2450  -1.144   0.2596
## log(housing_2000)        1.5179     0.1718   8.836 9.45e-11 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.3163 on 38 degrees of freedom
## Multiple R-squared:  0.9422, Adjusted R-squared:  0.9377
## F-statistic: 206.7 on 3 and 38 DF,  p-value: < 2.2e-16
```

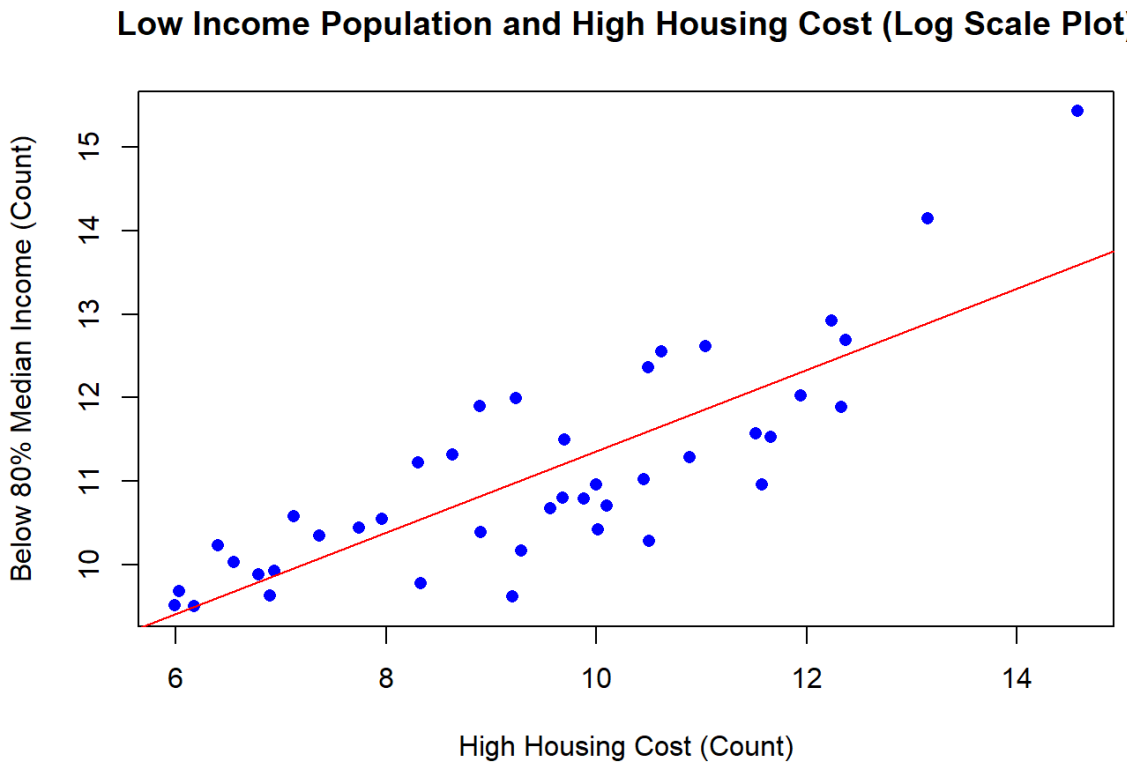
```
# anova analysis to identify change
anova(
    fit_acs19_housing_3000,
    fit_acs19_housing_2500,
    fit_acs19_housing_2000
)
```

```
## Analysis of Variance Table
##
## Model 1: log(hh_median) ~ log(housing_more_3000)
## Model 2: log(hh_median) ~ log(housing_more_3000) + log(housing_2500)
## Model 3: log(hh_median) ~ log(housing_more_3000) + log(housing_2500) +
##     log(housing_2000)
##   Res.Df     RSS Df Sum of Sq       F    Pr(>F)
## 1     40 21.6708
## 2     39 11.6141  1   10.0567 100.516 3.178e-12 ***
## 3     38  3.8019  1    7.8122  78.082 9.451e-11 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

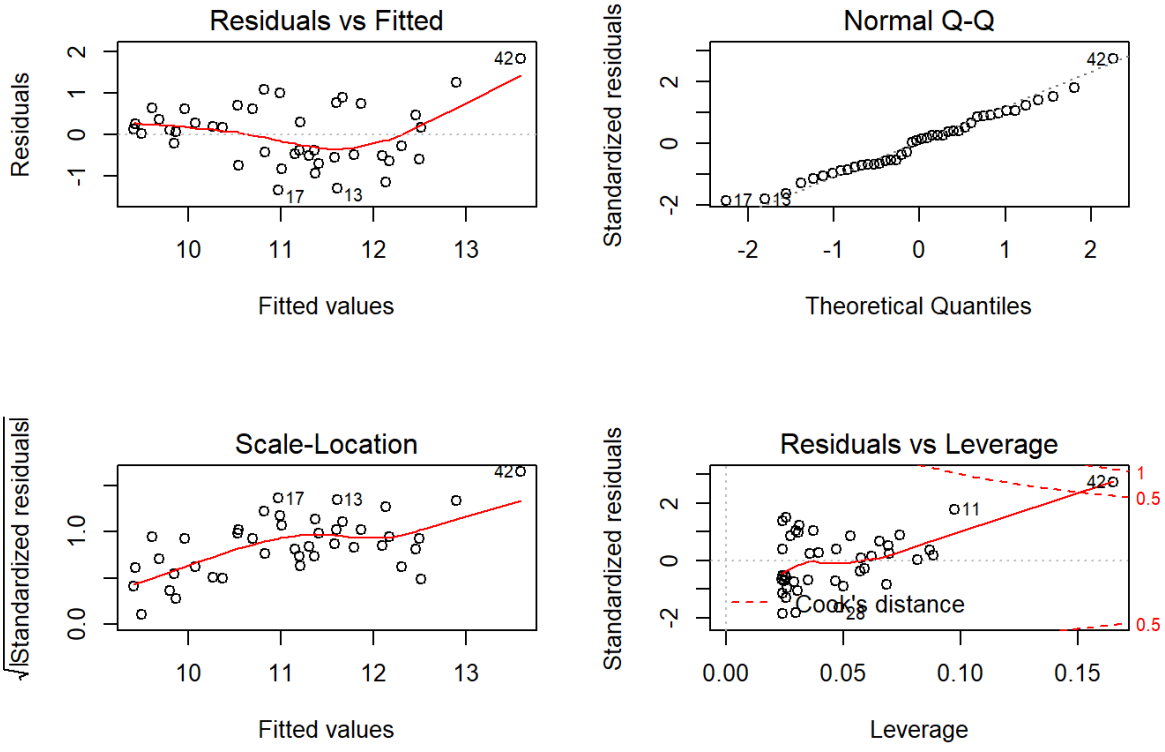
```
# calculate correlation
col_housing = c(
    "housing_more_3000",
    "housing_2500",
    "housing_2000"
)
cor(acs19_nofactor$hh_median, acs19_nofactor[col_housing])
```

```
##      housing_more_3000 housing_2500 housing_2000
## [1,]           0.99009    0.9957526    0.9978423
```

```
# visualize income and housing variables
# https://www.theanalysisfactor.com/linear-models-r-plotting-regression-lines/
plot(log(hh_median) ~ log(as.numeric(housing_more_3000)),
    main="Low Income Population and High Housing Cost (Log Scale Plot)",
    xlab="High Housing Cost (Count)",
    ylab="Below 80% Median Income (Count)",
    pch=16,
    col="blue",
    data=acs19_nofactor
)
# trend line plot
abline(
    fit_acs19_housing_3000,
    col="red"
)
```




```
# diagnostic plot
par(mfrow=c(2,2))
plot(fit_acs19_housing_3000)
```



2019 ACS - Education Attainment

> Education attainment has a positive relationship with median household income as verified with the model fit and low p-value. The model fit revealed an unexpected result that counties with larger low income populations also had larger populations with high education attainment (doctorate degree). However, this finding may be accounted by counties which are large metro areas tend to have large populations with both categories.

The model fit demonstrated that education attainment is associated with median household income. The model did not include all education attainment features due to their large number (i.e. all grade levels). As a result, only higher education attainment features were included in the model.

The plot shows count of residents below 80% of median income and residents with education attainment of a doctorate degree. The regression line indicates a positive relationship, so counties with higher low income population also had higher population count with doctorate degrees. This finding is an unexpected but may be accounted by counties which are large metro areas tend to have large populations with both categories.

```
# features:
# B15003_002E: No schooling completed
# B15003_003E: Nursery school
# B15003_004E: Kindergarten
# B15003_005E: 1st grade
# B15003_006E: 2nd grade
# B15003_007E: 3rd grade
# B15003_008E: 4th grade
# B15003_009E: 5th grade
# B15003_010E: 6th grade
# B15003_011E: 7th grade
# B15003_012E: 8th grade
# B15003_013E: 9th grade
# B15003_014E: 10th grade
# B15003_015E: 11th grade
# B15003_016E: 12th grade, no diploma
# B15003_017E: Regular high school diploma
# B15003_018E: GED or alternative credential
# B15003_019E: Some college, less than 1 year
# B15003_020E: Some college, 1 or more years, no degree
# B15003_021E: Associate's degree
# B15003_022E: Bachelor's degree
# B15003_023E: Master's degree
# B15003_024E: Professional school degree
# B15003_025E: Doctorate degree

# linear regression model fit - education attainment
# note: fit to log scale
# source: https://stats.stackexchange.com/questions/176595/simple-log-regression-model-in-r
fit_acs19_edu_doctorate <- lm(
    log(hh_median) ~ log(edu_doctorate),
    data=acs19_nofactor
)
round(summary(fit_acs19_edu_doctorate)$coeff, 6)
```

```
##                   Estimate Std. Error  t value Pr(>|t|)
## (Intercept)       6.715289   0.445508 15.07334        0
## log(edu_doctorate) 0.539567   0.053350 10.11371        0
```

```
fit_acs19_edu_professional <- lm(
    log(hh_median) ~ log(edu_doctorate) +
    log(edu_professional),
    data=acs19_nofactor
)
round(summary(fit_acs19_edu_professional)$coeff, 6)
```

```
##                       Estimate Std. Error   t value Pr(>|t|)
## (Intercept)            5.016929   0.624724  8.030631 0.000000
## log(edu_doctorate)    -0.037691   0.171506 -0.219763 0.827202
## log(edu_professional)  0.726732   0.207603  3.500592 0.001179
```

```
fit_acs19_edu_master <- lm(
    log(hh_median) ~ log(edu_doctorate) +
    log(edu_professional) +
    log(edu_master),
    data=acs19_nofactor
)
round(summary(fit_acs19_edu_master)$coeff, 6)
```

```
##                       Estimate Std. Error   t value Pr(>|t|)
## (Intercept)            3.047394   0.855486  3.562178 0.001010
## log(edu_doctorate)    -0.295044   0.176634 -1.670374 0.103063
## log(edu_professional) -0.000925   0.302544 -0.003058 0.997576
## log(edu_master)        1.042691   0.339400  3.072159 0.003917
```

```
fit_acs19_edu_bachelor <- lm(
    log(hh_median) ~ log(edu_doctorate) +
    log(edu_professional) +
    log(edu_master) +
    log(edu_bachelor),
    data=acs19_nofactor
)
round(summary(fit_acs19_edu_bachelor)$coeff, 6)
```

```
##                       Estimate Std. Error   t value Pr(>|t|)
## (Intercept)           -0.064401   0.831364 -0.077464 0.938672
## log(edu_doctorate)    -0.371825   0.130855 -2.841514 0.007262
## log(edu_professional) -0.150718   0.224480 -0.671408 0.506131
## log(edu_master)       -0.431466   0.358431 -1.203763 0.236326
## log(edu_bachelor)      1.803676   0.314123  5.741935 0.000001
```

```
# model fit summary
summary(fit_acs19_edu_bachelor)
```

```
##
## Call:
## lm(formula = log(hh_median) ~ log(edu_doctorate) + log(edu_professional) +
##     log(edu_master) + log(edu_bachelor), data = acs19_nofactor)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.75742 -0.26971 -0.00101  0.29885  0.71670
##
## Coefficients:
##                       Estimate Std. Error t value Pr(>|t|)
## (Intercept)            -0.0644     0.8314  -0.077  0.93867
## log(edu_doctorate)     -0.3718     0.1308  -2.842  0.00726 **
## log(edu_professional)  -0.1507     0.2245  -0.671  0.50613
## log(edu_master)        -0.4315     0.3584  -1.204  0.23633
## log(edu_bachelor)       1.8037     0.3141   5.742  1.4e-06 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.4015 on 37 degrees of freedom
## Multiple R-squared:  0.9094, Adjusted R-squared:  0.8996
## F-statistic: 92.84 on 4 and 37 DF,  p-value: < 2.2e-16
```

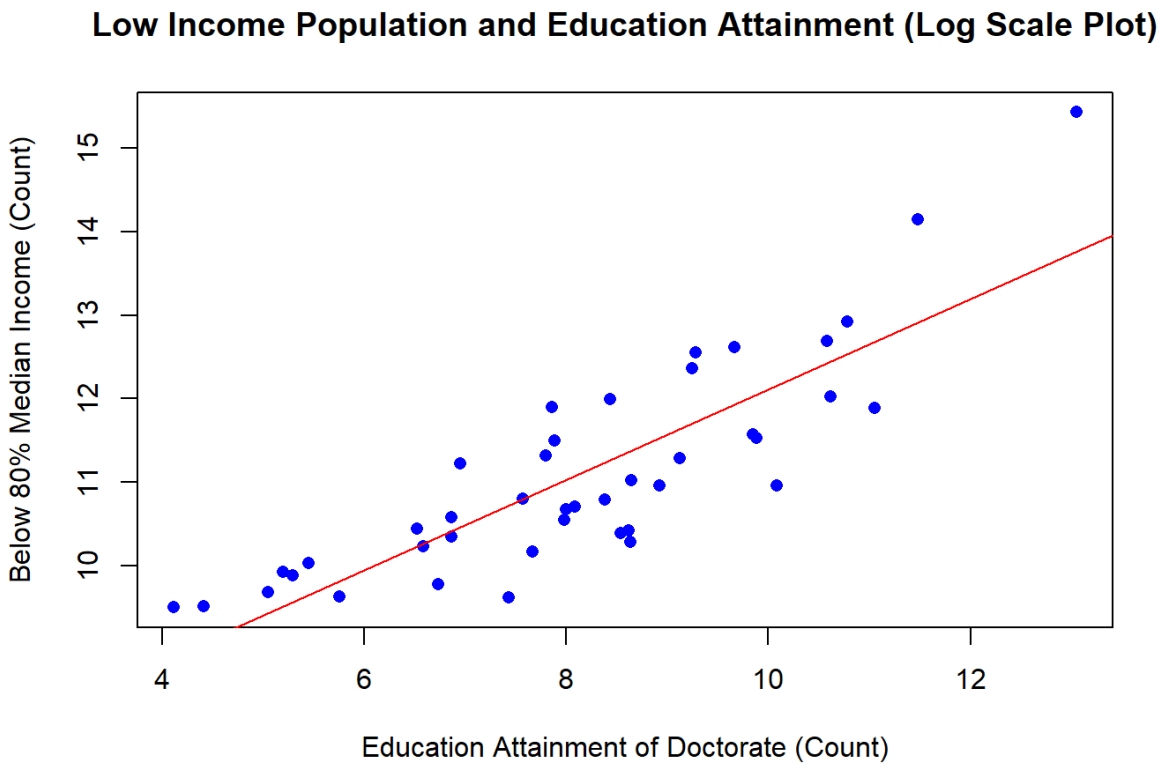
```
# anova analysis to identify change
anova(
    fit_acs19_edu_doctorate,
    fit_acs19_edu_professional,
    fit_acs19_edu_master,
    fit_acs19_edu_bachelor
)
```

```
## Analysis of Variance Table
##
## Model 1: log(hh_median) ~ log(edu_doctorate)
## Model 2: log(hh_median) ~ log(edu_doctorate) + log(edu_professional)
## Model 3: log(hh_median) ~ log(edu_doctorate) + log(edu_professional) +
##     log(edu_master)
## Model 4: log(hh_median) ~ log(edu_doctorate) + log(edu_professional) +
##     log(edu_master) + log(edu_bachelor)
##   Res.Df     RSS Df Sum of Sq      F    Pr(>F)
## 1     40 18.5064
## 2     39 14.0818  1    4.4246 27.446 6.729e-06 ***
## 3     38 11.2801  1    2.8017 17.379 0.0001769 ***
## 4     37  5.9649  1    5.3152 32.970 1.404e-06 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

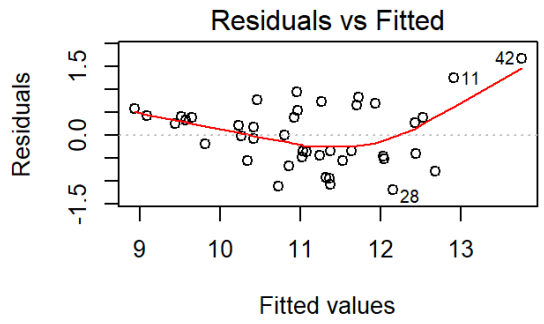
```
# calculate correlation
col_edu = c(
    "edu_doctorate",
    "edu_professional",
    "edu_master",
    "edu_bachelor"
)
cor(acs19_nofactor$hh_median, acs19_nofactor[col_edu])
```

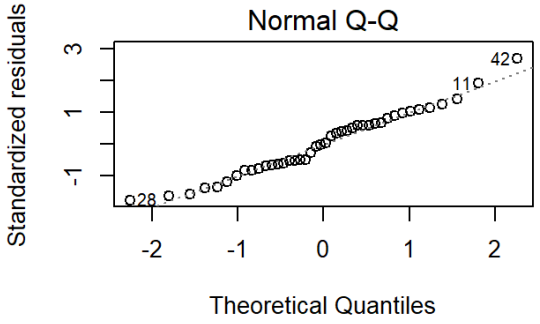
```
##      edu_doctorate edu_professional edu_master edu_bachelor
## [1,]       0.98663        0.9956973  0.9923332    0.9969078
```

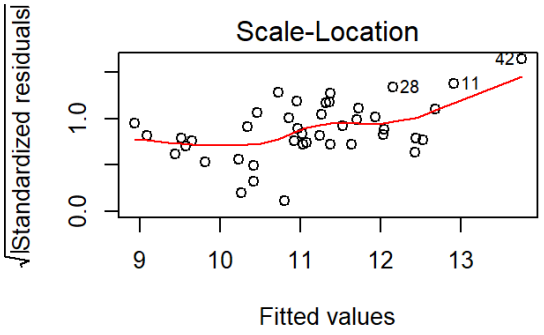
```
# visualize income and education variables
# https://www.theanalysisfactor.com/linear-models-r-plotting-regression-lines/
plot(log(hh_median ) ~ log(as.numeric(edu_doctorate)),
    main="Low Income Population and Education Attainment (Log Scale Plot)",
    xlab="Education Attainment of Doctorate (Count)",
    ylab="Below 80% Median Income (Count)",
    pch=16,
    col="blue",
    data=acs19_nofactor
)
# trend line plot
abline(
    fit_acs19_edu_doctorate,
    col="red"
)
```

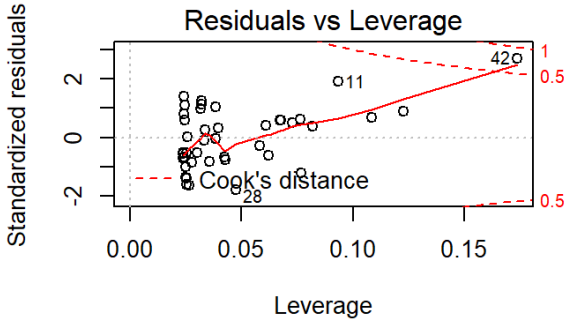



```
# diagnostic plot
par(mfrow=c(2,2))
plot(fit_acs19_edu_doctorate)
```









2019 ACS - Race

Race has a positive relationship with median household income as verified with the model fit and low p-value. In particular, residents which indicated “White alone”, “American Indian and Alaska Native alone” and “Two or more races” as their race were more closely associated with median household income.

```
# features:
# B02001_002E: White alone
# B02001_003E: Black or African American alone
# B02001_004E: American Indian and Alaska Native alone
# B02001_005E: Asian alone
# B02001_006E: Native Hawaiian and Other Pacific Islander alone
# B02001_007E: Some other race alone
# B02001_008E: Two or more races

# linear regression
# note: fit to log scale
# source: https://stats.stackexchange.com/questions/176595/simple-log-regression-model-in-r
fit_acs19_race1 <- lm(
    log(hh_median) ~ log(as.numeric(B02001_002E)),
    data=acs19_nofactor
)
round(summary(fit_acs19_race1)$coeff, 6)
```

```
##                              Estimate Std. Error   t value Pr(>|t|)
## (Intercept)                 -1.301261   0.354226 -3.673534 0.000701
## log(as.numeric(B02001_002E))  0.977316   0.027792 35.165567 0.000000
```

```
fit_acs19_race2 <- lm(
    log(hh_median) ~ log(as.numeric(B02001_002E)) +
    log(as.numeric(B02001_003E)),
    data=acs19_nofactor
)
round(summary(fit_acs19_race2)$coeff, 6)
```

```
##                               Estimate Std. Error   t value Pr(>|t|)
## (Intercept)                  -0.549873   0.489895 -1.122431 0.268542
## log(as.numeric(B02001_002E))  0.853141   0.064150 13.299167 0.000000
## log(as.numeric(B02001_003E))  0.086686   0.040738  2.127879 0.039723
```

```
fit_acs19_race3 <- lm(
    log(hh_median ) ~ log(as.numeric(B02001_002E)) +
    log(as.numeric(B02001_003E)) +
    log(as.numeric(B02001_004E)),
    data=acs19_nofactor
)
round(summary(fit_acs19_race3)$coeff, 6)
```

```
##                              Estimate Std. Error   t value Pr(>|t|)
## (Intercept)                  0.010691   0.405421  0.026369 0.979101
## log(as.numeric(B02001_002E)) 0.664671   0.063870 10.406689 0.000000
## log(as.numeric(B02001_003E)) 0.069297   0.032532  2.130108 0.039697
## log(as.numeric(B02001_004E)) 0.237904   0.048666  4.888460 0.000019
```

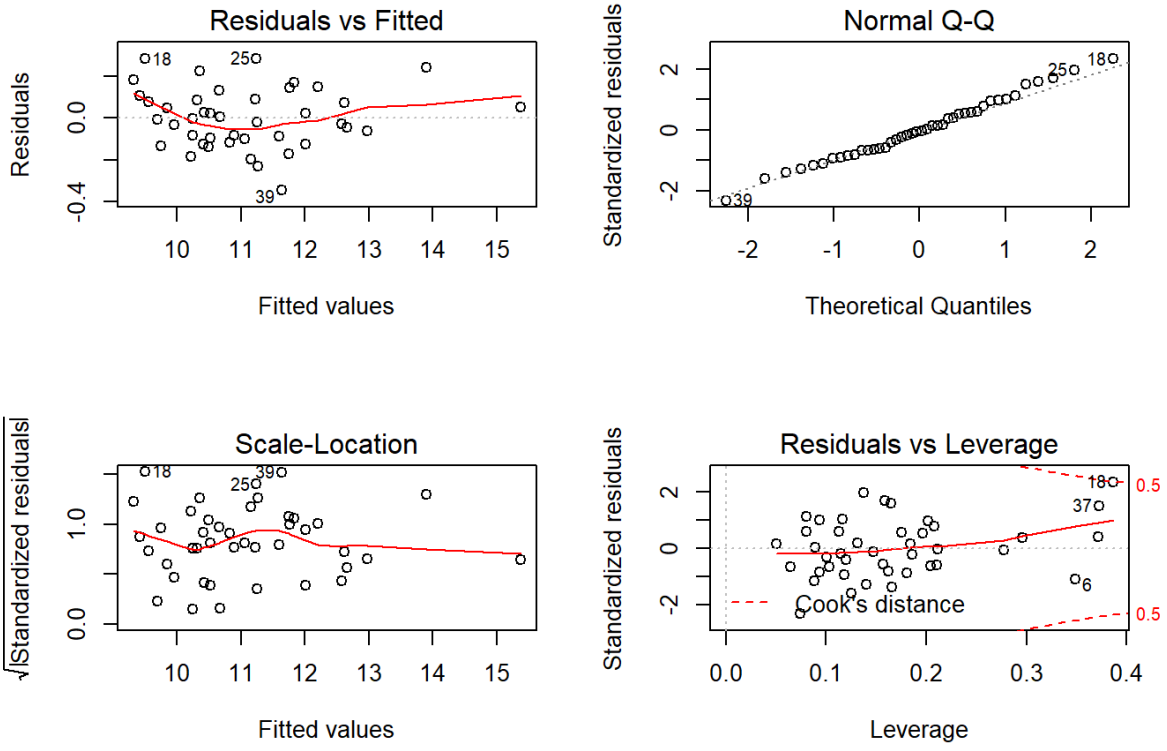
```
fit_acs19_race_all <- lm(
    log(hh_median) ~ log(as.numeric(B02001_002E)) +
    log(as.numeric(B02001_003E)) +
    log(as.numeric(B02001_004E)) +
    log(as.numeric(B02001_005E)) +

    # leave out 06e due to log error
    # (as.numeric(B02001_006E)) +
    log(as.numeric(B02001_007E)) +
    log(as.numeric(B02001_008E)),
    data=acs19_nofactor
)
round(summary(fit_acs19_race_all)$coeff, 6)
```

```
##                               Estimate Std. Error   t value Pr(>|t|)
## (Intercept)                   0.146095   0.402050  0.363376 0.718510
## log(as.numeric(B02001_002E))  0.503415   0.083946  5.996916 0.000001
## log(as.numeric(B02001_003E))  0.010125   0.045246  0.223771 0.824237
## log(as.numeric(B02001_004E))  0.221352   0.050019  4.425317 0.000090
## log(as.numeric(B02001_005E)) -0.078830   0.043106 -1.828757 0.075968
## log(as.numeric(B02001_007E))  0.038855   0.031511  1.233040 0.225779
## log(as.numeric(B02001_008E))  0.300439   0.086855  3.459099 0.001443
```

```
# model fit summary
summary(fit_acs19_race_all)
```

```
##
## Call:
## lm(formula = log(hh_median) ~ log(as.numeric(B02001_002E)) +
##     log(as.numeric(B02001_003E)) + log(as.numeric(B02001_004E)) +
##     log(as.numeric(B02001_005E)) + log(as.numeric(B02001_007E)) +
##     log(as.numeric(B02001_008E)), data = acs19_nofactor)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.34579 -0.09869 -0.00498  0.08764  0.28382
##
## Coefficients:
##                               Estimate Std. Error t value Pr(>|t|)
## (Intercept)                    0.14610    0.40205   0.363  0.71851
## log(as.numeric(B02001_002E))   0.50342    0.08395   5.997 7.79e-07 ***
## log(as.numeric(B02001_003E))   0.01012    0.04525   0.224  0.82424
## log(as.numeric(B02001_004E))   0.22135    0.05002   4.425 8.97e-05 ***
## log(as.numeric(B02001_005E)) -0.07883    0.04311  -1.829  0.07597 .
## log(as.numeric(B02001_007E))   0.03885    0.03151   1.233  0.22578
## log(as.numeric(B02001_008E))   0.30044    0.08685   3.459  0.00144 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.1551 on 35 degrees of freedom
## Multiple R-squared:  0.9872, Adjusted R-squared:  0.985
## F-statistic: 450.3 on 6 and 35 DF,  p-value: < 2.2e-16
```

```
# diagnostic plot
par(mfrow=c(2,2))
plot(fit_acs19_race_all)
```



2019 ACS - Poverty Level

> Poverty level has a positive relationship with median household income as verified with the model fit and low p-value. This model was not used for additional analysis since the dataset consists of two halves so did not have sufficient granularity for such use.

Although the model fit was excellent, it did not have sufficient granularity to warrant additional analysis. In addition, poverty level is determined based on median income so the features are highly correlated and not provide useful results.

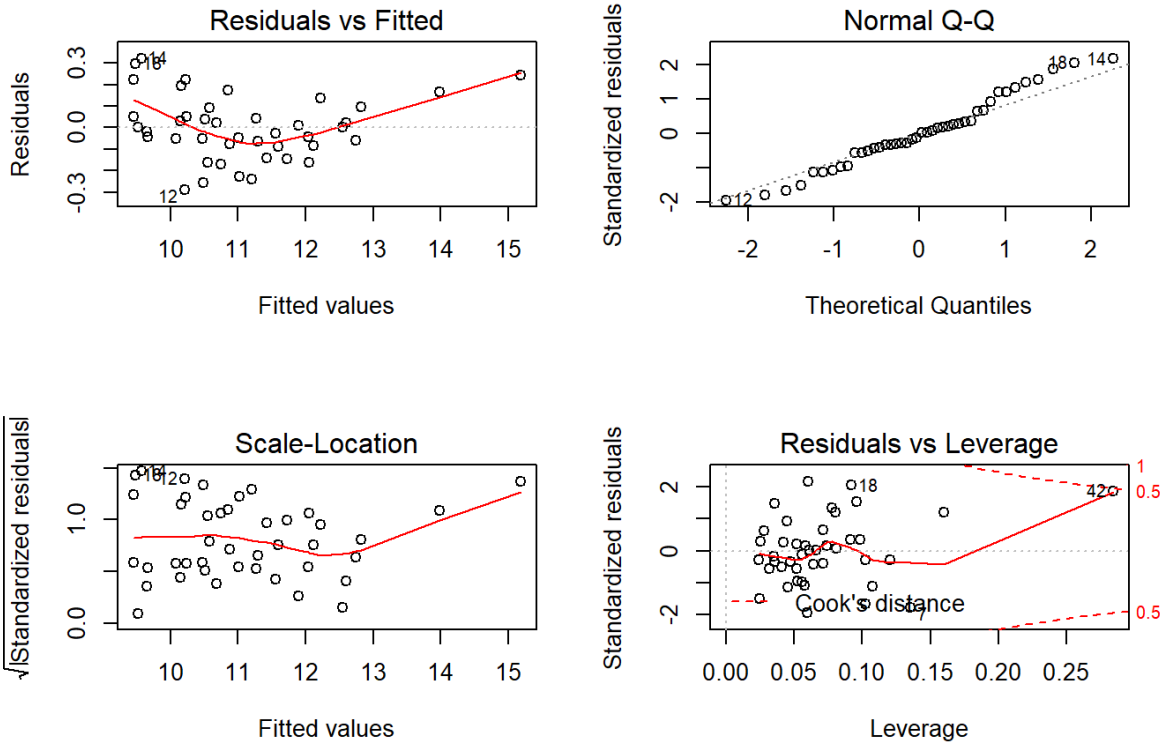
```
# features:
# B17001_002E: Income in the past 12 months below poverty level
# B17001_031E: Income in the past 12 months at or above poverty level

# linear regression model fit - poverty below
# note: fit to log scale
# source: https://stats.stackexchange.com/questions/176595/simple-log-regression-model-in-r
fit_acs19_poverty_above <- lm(
    log(hh_median) ~ log(poverty_above),
    data=acs19_nofactor
)
summary(fit_acs19_poverty_above)
```

```
##
## Call:
## lm(formula = log(hh_median) ~ log(poverty_above), data = acs19_nofactor)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.62167 -0.17409  0.03059  0.19942  0.44331
##
## Coefficients:
##                    Estimate Std. Error t value Pr(>|t|)
## (Intercept)        -0.28019    0.38481  -0.728    0.471
## log(poverty_above)  0.88018    0.02961  29.729   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.2669 on 40 degrees of freedom
## Multiple R-squared:  0.9567, Adjusted R-squared:  0.9556
## F-statistic: 883.8 on 1 and 40 DF,  p-value: < 2.2e-16
```

```
# linear regression model fit - poverty below + above
fit_acs19_poverty_all <- lm(
    log(hh_median) ~ log(poverty_below) +
    log(poverty_above),
    data=acs19_nofactor
)
summary(fit_acs19_poverty_all)
```

```
##
## Call:
## lm(formula = log(hh_median) ~ log(poverty_below) + log(poverty_above),
##     data = acs19_nofactor)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.28958 -0.08332 -0.00899  0.08327  0.32185
##
## Coefficients:
##                   Estimate Std. Error t value Pr(>|t|)
## (Intercept)         0.18759    0.22723   0.826    0.414
## log(poverty_below)  0.53516    0.05911   9.054 3.95e-11 ***
## log(poverty_above)  0.39141    0.05661   6.915 2.81e-08 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.1535 on 39 degrees of freedom
## Multiple R-squared:  0.986,  Adjusted R-squared:  0.9853
## F-statistic:  1377 on 2 and 39 DF,  p-value: < 2.2e-16
```

```
# diagnostic plot
par(mfrow=c(2,2))
plot(fit_acs19_poverty_all)
```



2019 ACS - Model Refinement

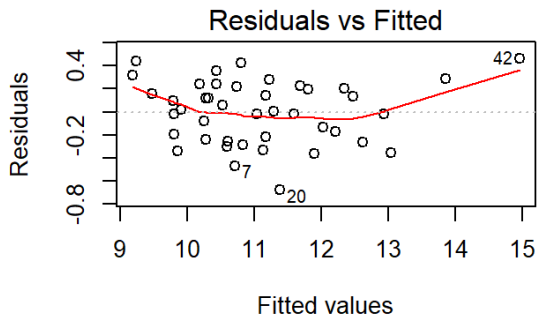
> Improve single models by selecting best features and including into single model.

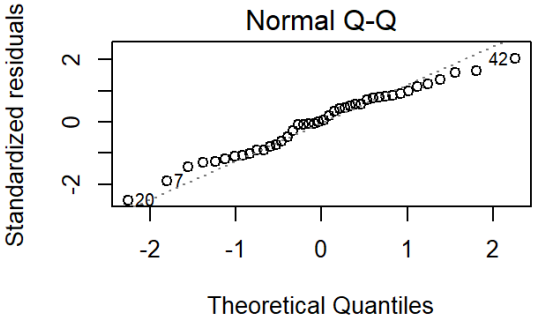
The linear regression model was refined by fitting the best attributes from each single linear model which improved the model as verified with a lower p-value and higher r-squared value than as compared to the other models using individual data tables.

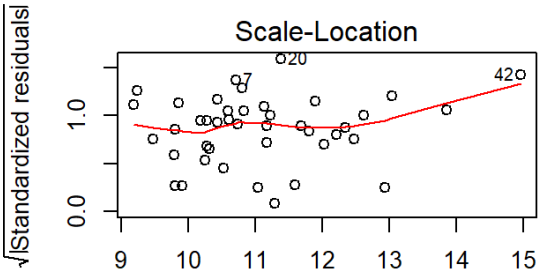
```
# fit best features to evaluate their importance
fit_acs19_best_features = lm(
    log(hh_median) ~ log(as.numeric(housing_more_3000)) +
    log(as.numeric(commute_car)) +
    log(as.numeric(commute_walk)) +
    log(as.numeric(edu_doctorate)) +
    log(as.numeric(edu_bachelor)),
    data=acs19_nofactor
)
summary(fit_acs19_best_features)
```

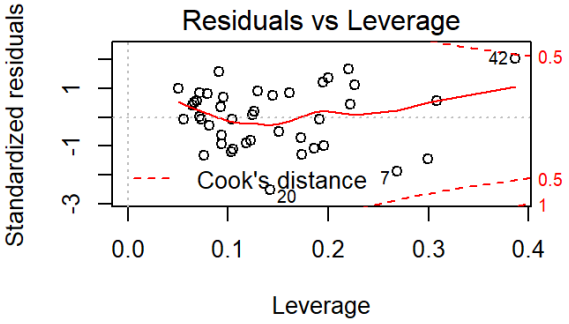
```
##
## Call:
## lm(formula = log(hh_median) ~ log(as.numeric(housing_more_3000)) +
##     log(as.numeric(commute_car)) + log(as.numeric(commute_walk)) +
##     log(as.numeric(edu_doctorate)) + log(as.numeric(edu_bachelor)),
##     data = acs19_nofactor)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.68062 -0.23713  0.01061  0.21814  0.46480
##
## Coefficients:
##                                  Estimate Std. Error t value Pr(>|t|)
## (Intercept)                       1.03065    0.83491   1.234   0.2250
## log(as.numeric(housing_more_3000)) -0.54663    0.09685  -5.644 2.08e-06 ***
## log(as.numeric(commute_car))       -0.12776    0.06731  -1.898   0.0657 .
## log(as.numeric(commute_walk))      -0.12168    0.06091  -1.998   0.0534 .
## log(as.numeric(edu_doctorate))     -0.10887    0.10591  -1.028   0.3108
## log(as.numeric(edu_bachelor))       1.52892    0.12788  11.956 4.28e-14 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.2918 on 36 degrees of freedom
## Multiple R-squared:  0.9534, Adjusted R-squared:  0.947
## F-statistic: 147.4 on 5 and 36 DF,  p-value: < 2.2e-16
```

```
# diagnostic plot
par(mfrow=c(2,2))
plot(fit_acs19_best_features)
```









2019 ACS - Disadvantaged Communities

> This section identifies disadvantaged communities by subset of the ACS 2019 dataset of communities with highest populations below 80% of median household income.

80% of median household income was selected as a threshold to identify disadvantaged communities based on the CA HUD [low income guidelines][4.00] which indicates a similar metric. In general, income level is typically used to evaluate communities and residents in need of government assistance so it was used for this portion of data analysis.

The California communities with the highest count of residents below 80% median household income are listed below in rank order:

1. Los Angeles County: 1,389,928
2. San Diego County: 407,303
3. Orange County: 323,340
4. Riverside County: 302,372
5. San Bernardino County: 283,845
6. Sacramento County: 232,810
7. Alameda County: 167,663
8. Fresno County: 161,743

```
# filter for median income above 0.80 quantile
# https://stackoverflow.com/questions/6253837/subset-data-frame-based-on-percentage
# https://faculty.nps.edu/sebuttre/home/R/factors.html
# hh_median_20 = subset(
#     acs19_hh_income,
#     as.numeric(as.character(B19001_001E)) <= quantile(as.numeric(as.character(B19001_001E)), 0.2),
#     select=c(NAME, B19001_001E)
# )

# sort in descending order
# https://dplyr.tidyverse.org/reference/desc.html
# hh_median_20 %>% arrange(desc(as.numeric(B19001_001E)))

# sort by median household income
# https://dplyr.tidyverse.org/reference/arrange.html
# dim(hh_median_20)
# arrange(hh_median_20, as.numeric(as.character(B19001_001E)))
# arrange(acs19, as.numeric(as.character(hh_median)))

# filter for median income below 0.20 quantile
# https://stackoverflow.com/questions/6253837/subset-data-frame-based-on-percentage
# https://faculty.nps.edu/sebuttre/home/R/factors.html
# hh_median_80 = subset(
#     acs19_hh_income,
#     as.numeric(as.character(B19001_001E)) >= quantile(as.numeric(as.character(B19001_001E)), 0.8),
#     select=c(NAME, B19001_001E)
# )

# sort in descending order
# https://dplyr.tidyverse.org/reference/desc.html
# hh_median_80 %>% arrange(desc(as.numeric(B19001_001E)))

# sort by median household income
# https://dplyr.tidyverse.org/reference/arrange.html
# https://faculty.nps.edu/sebuttre/home/R/factors.html
# dim(hh_median_80)
# arrange(hh_median_80, as.numeric(as.character(B19001_001E)))

# boxplot for median income below 0.80 quantile
# https://www.biostars.org/p/344165/
# https://stackoverflow.com/questions/14872783/how-do-i-show-all-boxplot-labels
# par(mar=c(10,2,2,1))
# boxplot(as.numeric(B19001_001E) ~ NAME,
#     data=hh_median_80,
#     las=2,
#     cex.axis=0.5,
#     main="Median Household Income (Low Income: 80% Quantile)",
#     ylab="Median Household Income ($)",
#     xlab=""
# )

# verify dataset size after subset
# dim(acs19_hh_income)
# unique(acs19_hh_income$NAME)
# dim(hh_median_20)
# unique(hh_median_20$NAME)
# dim(hh_median_80)
# unique(hh_median_80$NAME)

# names(acs19)
# names(acs19_nofactor)
```

```
# names(acs19_hh_income)
# names(acs19_hh_income_median)

# join tables for county name and median income
acs19_hh_income_name = inner_join(
    acs19,
    acs19_hh_income_median,
    by="GEO_ID"
)
# names(acs19_hh_income_name)

# subset and sort by median income
# income_name_vars = c("NAME", "hh_median")
# acs19_hh_income_name_only = acs19_hh_income_name[income_name_vars]
acs19_hh_income_name_only = subset(
    acs19_hh_income_name,
    as.numeric(as.character(hh_median)) >= quantile(as.numeric(as.character(hh_median)), 0.8),
    select=c(NAME, hh_median)
)
# arrange(acs19_hh_income_name_only, as.numeric(as.character(hh_median)))
acs19_hh_income_name_only %>% arrange(desc(hh_median))
```

```
##                                NAME hh_median
## 1                        California   5000438
## 2    Los Angeles County, California   1389928
## 3      San Diego County, California    407303
## 4         Orange County, California    323340
## 5      Riverside County, California    302372
## 6 San Bernardino County, California    283845
## 7     Sacramento County, California    232810
## 8        Alameda County, California    167663
## 9         Fresno County, California    161743
```

NCHS COVID-19 Data

> This section identifies impacts on disadvantaged communities with a subset of the NHCS COVID dataset of communities with population count above 20% of COVID-related mortality count.

The NHCS COVID-19 [mortality count data][4.01] reported by county was used to measure the impact of the pandemic on communities within California. The results showed that larger metro areas had higher total mortality count as compared with those in smaller, rural communities. This trend is validated in findings from sources such as the Los Angeles Times COVID-19 Dashboard (https://uscensus.maps.arcgis.com/home/item.html?id=0fbb1571e5b6458f941580d1d64a6693) which provides data visualizations of highest case counts within California.

One possible area for analysis would be to identify disadvantage communities within each California county to differentiate their levels of need since the current rank of counties indicate pandemic impacts primarily based on metro type and total population count.

California communities with the highest COVID-related mortality count based the NCHS dataset are listed below in rank order:

1. Los Angeles County (Large central metro): 6,624
2. Orange County (Large central metro): 1,452
3. Riverside County (Large central metro): 1,448
4. San Bernardino County (Large fringe metro): 1,311
5. San Diego County (Large central metro): 1,150
6. Stanislaus County (Medium metro): 537
7. Sacramento County (Large central metro): 525

8. San Joaquin County (Medium metro): 484
9. Fresno County (Medium metro): 473

```
# data source: U.S. Census Data Portal
nchs_covid_county <- read.csv(
    # "https://data.census.gov/cedsci/table?q=B06007&tid=ACSDT1Y2019.B06007.csv"
    "data/Provisional_COVID-19_Death_Counts_in_the_United_States_by_County.csv"
)

# show table dim
# print("Show table dimensions below:")
# dim(nchs_covid_county)

# preview table rows
# print("Preview table rows below:")
# head(nchs_covid_county)

# subset by state
nchs_covid_county_ca <- subset(
    nchs_covid_county,
    State=='CA',
    select=c(State, County.name, Urban.Rural.Code, Deaths.involving.COVID.19)
)

# filter for mortality count below 0.20 quantile
# https://stackoverflow.com/questions/6253837/subset-data-frame-based-on-percentage
# nchs_covid_county_20 = subset(
#     nchs_covid_county_ca, as.numeric(Deaths.involving.COVID.19) <= quantile(as.numeric(Deaths.involving.COVI
D.19), 0.2)
# )

# sort in descending order
# https://dplyr.tidyverse.org/reference/desc.html
# names(nchs_covid_county_20)
# dim(nchs_covid_county_20)
# nchs_covid_county_20 %>% arrange(desc(Deaths.involving.COVID.19))

# filter for mortality count above 0.80 quantile
# https://stackoverflow.com/questions/6253837/subset-data-frame-based-on-percentage
nchs_covid_county_80 = subset(
    nchs_covid_county_ca, as.numeric(Deaths.involving.COVID.19) >= quantile(as.numeric(Deaths.involving.COVID.
19), 0.8)
)

# sort in descending order
# https://dplyr.tidyverse.org/reference/desc.html
# names(nchs_covid_county_80)
# dim(nchs_covid_county_80)
nchs_covid_county_80 %>% arrange(desc(Deaths.involving.COVID.19))
```

```
##   State           County.name    Urban.Rural.Code Deaths.involving.COVID.19
## 1    CA    Los Angeles County Large central metro                      6624
## 2    CA         Orange County Large central metro                      1452
## 3    CA      Riverside County Large central metro                      1448
## 4    CA San Bernardino County  Large fringe metro                      1311
## 5    CA      San Diego County Large central metro                      1150
## 6    CA     Stanislaus County        Medium metro                       537
## 7    CA     Sacramento County Large central metro                       525
## 8    CA    San Joaquin County        Medium metro                       484
## 9    CA         Fresno County        Medium metro                       473
```

Part 4: Recommendations

Key Findings

> This section provides recommendations based on data analysis results to help disadvantage communities impacted by the pandemic.

Key findings are as follows:

1. Best Features: Education attainment, commute mode and housing cost had the best relationship with median household income among the features which were compared with median income. As a result, disadvantaged communities would most benefit from assistance with access to education, transit and affordable housing.
2. Population Count: The U.S. Census ACS and NCHS COVID-19 datasets were all provided by population count, so relationships were established between counties with high population count below 80% median income and other data features. Additional analysis is needed to differentiate specific impacts on disadvantaged communities since one county typically contains multiple communities.
3. Model Improvement: The linear regression models generally had good fits, and combining data features into the same model generated a better fit than the individual models. Data features were isolated to identify their impact to the model as they were added incrementally.
4. Pandemic Impact: COVID-related mortality count is higher in large metro areas, so additional analysis is needed within these metro areas to differentiate impacts to disadvantaged communities. For example, identifying the proportion of disadvantaged communities within counties with the highest COVID-related mortality count.

Data analysis findings which may be applied to other U.S. regions are as follows:

- Log Transformation: Some features have a wide variance and outlier/high leverage points; as a result, log transformation is recommended to reduce their impact. In addition, some data features vary in their magnitude, so the log transformation helped minimize their impacts.
- Linear Model: Combining best features resulted in better model fit, lower p-value and higher adjusted r-squared value, so it is recommended. Features were added incrementally to evaluate their individual impacts.

Part 5: Future Improvements

> This section provides possible areas for additional analysis.

1. PCA and Variance Plot: Identify principal components to validate findings from this project, and use plots to identify additional features which may contribute to variance. In addition, data features can be used to refine the linear model.
2. Histogram to Evaluate Skew: Plot distribution of each data table to evaluate their distribution and possible skew which may exist within the datasets. A few high leverage data points were observed during the analysis, so the plot will help identify their impacts.
3. Logistic Regression: Analyze median household income with location to evaluate whether location has a positive relationship with median income. All datasets are based on location, so their relationship with each data feature is of interest.