ASA Data Challenge Expo

Helping Communities During the COVID-19 Pandemic

Entry Details

- Event: 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 following questions through analysis of core and supplemental datasets:

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.

The intended audience are state/local governments, non-governmental organizations (NGOs) and volunteers which are able to provide aid and services to these communities.

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.
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.

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

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://www.latimes.com/projects/california-coronavirus-cases-tracking-outbreak/)

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. Join tables together into single dataframe

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 median household income.
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.
3. Identify communities with lowest median income and highest COVID-19 death counts by analyzing the 2019 ACS and NCHS COVID-19 mortality count data.

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. Carpool had a good fit in the car-based modes. Transit/remote work had good fits with transit-based modes. This finding implies that certain commute modes (i.e. carpool, transit and remote work) have a better relationship to income than other ones (i.e. walking).

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_transit <- lm(
    log(hh_median) ~ log(as.numeric(commute_transit)),
    data=acs19_nofactor
)
round(summary(fit_acs19_commute_transit)$coeff, 6)
```

```
##                                   Estimate Std. Error   t value Pr(>|t|)
## (Intercept)                      11.441476   0.144607 79.120982 0.000000
## log(as.numeric(commute_transit)) -0.079897   0.050044 -1.596517 0.118245
```

```
# 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)                      11.381655   0.160668 70.839554 0.000000
## log(as.numeric(commute_transit)) -0.112801   0.062959 -1.791662 0.080949
## log(as.numeric(commute_walk))     0.054523   0.062959  0.866015 0.391779
```

```r
# 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)                      11.700082   0.205122 57.039762 0.000000
## log(as.numeric(commute_transit)) -0.118576   0.059751 -1.984494 0.054456
## log(as.numeric(commute_walk))     0.055702   0.059702  0.932999 0.356711
## log(as.numeric(commute_remote))  -0.110482   0.047654 -2.318405 0.025906
```

```r
# variance analysis
anova(
    fit_acs19_commute_transit,
    fit_acs19_commute_transit_walk,
    fit_acs19_commute_transit_remote
)
```

```
## Analysis of Variance Table
##
## Model 1: log(hh_median) ~ log(as.numeric(commute_transit))
## 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 2.9166
## 2     39 2.8616  1   0.05503 0.8341 0.36684
## 3     38 2.5070  1   0.35461 5.3750 0.02591 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```r
# 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)                  11.492942   0.142192 80.826739 0.000000
## log(as.numeric(commute_car)) -0.098497   0.049209 -2.001614 0.052141
```

```r
# 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)                       11.721675   0.169333 69.222784 0.000000
## log(as.numeric(commute_car))      -0.072442   0.048289 -1.500162 0.141625
## log(as.numeric(commute_carpool)) -0.108719   0.048289 -2.251404 0.030069
```

```
# 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)                       11.920758   0.195793 60.884431 0.000000
## log(as.numeric(commute_car))      -0.059627   0.047322 -1.260027 0.215343
## log(as.numeric(commute_carpool)) -0.107633   0.046825 -2.298612 0.027119
## log(as.numeric(commute_other))   -0.085849   0.045997 -1.866393 0.069718
```

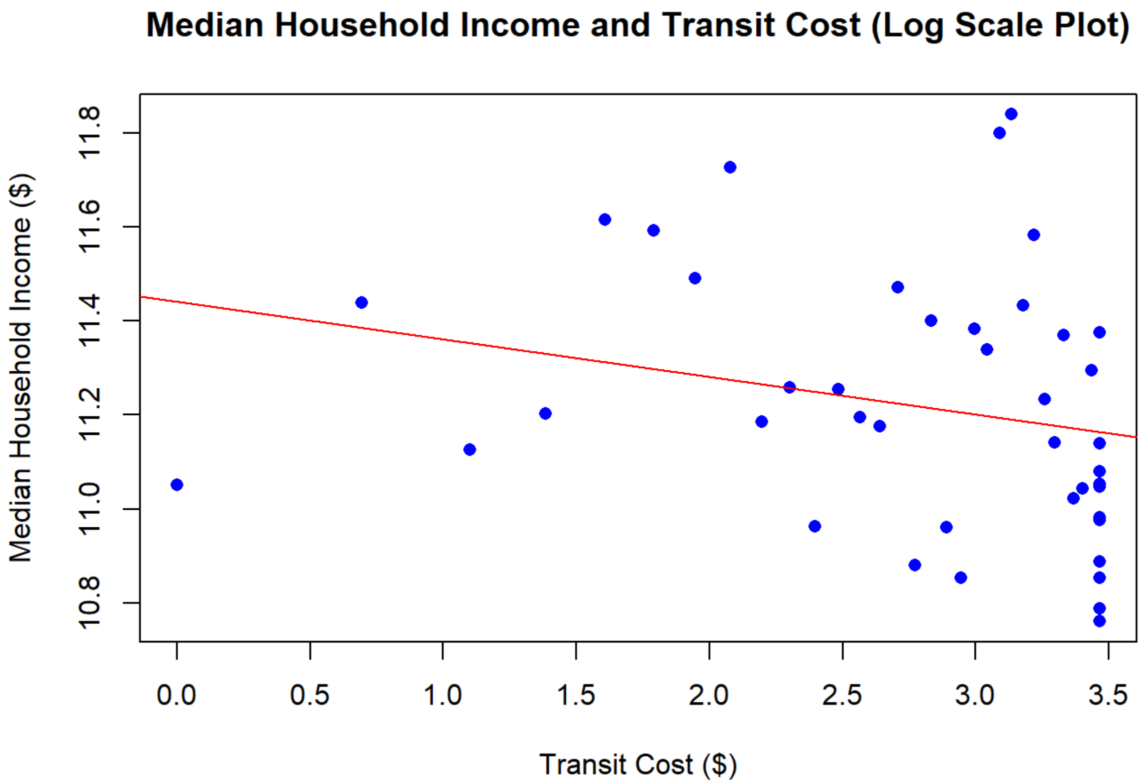
```
# variance analysis
anova(
    fit_acs19_commute_car,
    fit_acs19_commute_car_carpool,
    fit_acs19_commute_car_other
)
```

```
## 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 2.8200
## 2     39 2.4957  1   0.32436 5.3916 0.02569 *
## 3     38 2.2861  1   0.20956 3.4834 0.06972 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

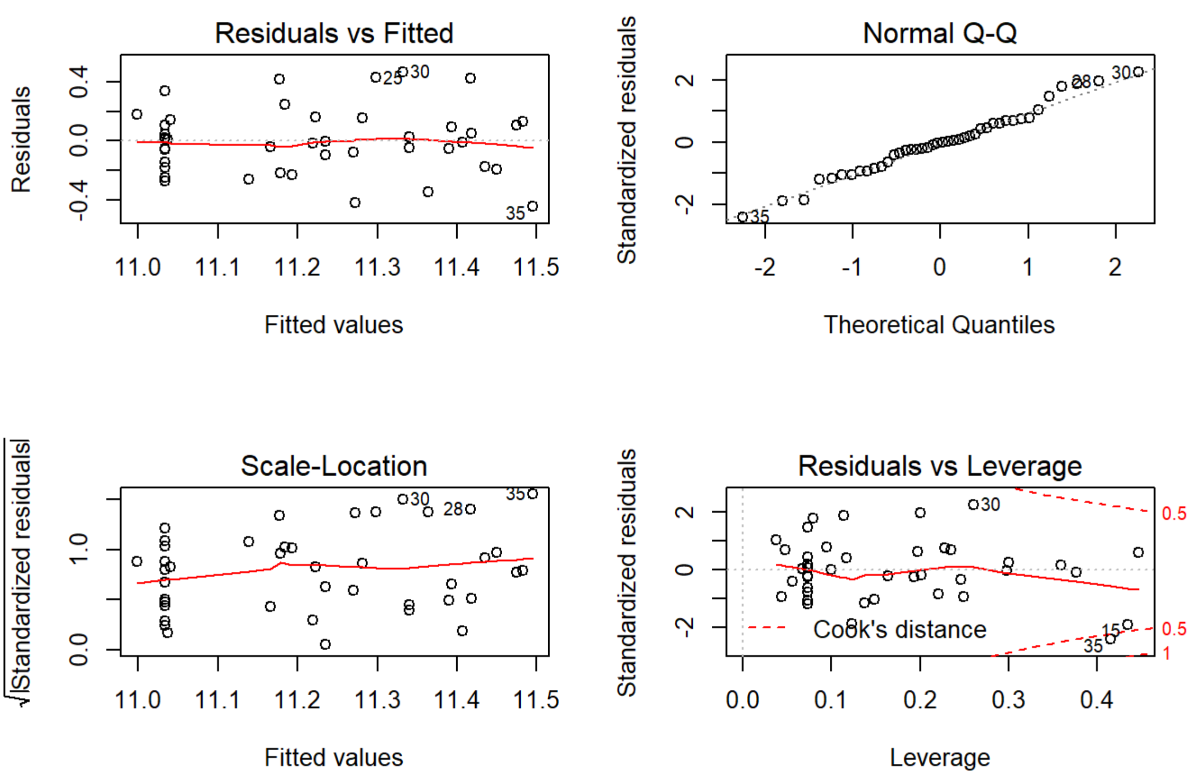
```
# visualize income and transit variables
# https://www.theanalysisfactor.com/linear-models-r-plotting-regression-lines/
plot(log(hh_median) ~ log(as.numeric(commute_transit)),
    main="Median Household Income and Transit Cost (Log Scale Plot)",
    xlab="Transit Cost ($)",
    ylab="Median Household Income ($)",
    pch=16,
    col="blue",
    data=acs19_nofactor
)

# 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
## -0.44367 -0.16962 -0.00412  0.12651  0.46676
##
## Coefficients:
##                                  Estimate Std. Error t value Pr(>|t|)
## (Intercept)                      11.95956    0.21533  55.540   <2e-16 ***
## log(as.numeric(commute_transit)) -0.07571    0.05825  -1.300   0.2022
## log(as.numeric(commute_walk))     0.09939    0.05847   1.700   0.0980 .
## log(as.numeric(commute_remote))  -0.05796    0.05276  -1.099   0.2794
## log(as.numeric(commute_car))     -0.03094    0.05075  -0.610   0.5461
## log(as.numeric(commute_carpool)) -0.10227    0.05071  -2.017   0.0514 .
## log(as.numeric(commute_other))   -0.09964    0.05080  -1.962   0.0578 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.2409 on 35 degrees of freedom
## Multiple R-squared:  0.3451, Adjusted R-squared:  0.2329
## F-statistic: 3.074 on 6 and 35 DF,  p-value: 0.01596
```

```
# trend line plot
abline(
    fit_acs19_commute_transit,
    col="red"
)
```




```
# 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. In particular, higher monthly housing costs were more closely associated with median household income.

This finding implies that certain higher housing cost is associated with median household income. However, lower housing cost features were not analyzed 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 the higher housing cost features were analyzed.

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)            10.308289   0.131590 78.336582        0
## log(housing_more_3000)  0.096496   0.013589  7.100991        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)            10.799505   0.146373 73.780543  0.0e+00
## log(housing_more_3000)  0.366723   0.056938  6.440710  0.0e+00
## log(housing_2500)      -0.330248   0.068302 -4.835088  2.1e-05
```

```
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)            11.305476   0.151144 74.799361 0.000000
## log(housing_more_3000)  0.287267   0.047057  6.104604 0.000000
## log(housing_2500)       0.037860   0.089612  0.422489 0.675049
## log(housing_2000)      -0.321007   0.062834 -5.108774 0.000009
```

```
# 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
## -0.18782 -0.09325 -0.02297  0.06727  0.26495
##
## Coefficients:
##                        Estimate Std. Error t value Pr(>|t|)
## (Intercept)            11.30548    0.15114  74.799  < 2e-16 ***
## log(housing_more_3000)  0.28727    0.04706   6.105 4.09e-07 ***
## log(housing_2500)       0.03786    0.08961   0.422    0.675
## log(housing_2000)      -0.32101    0.06283  -5.109 9.45e-06 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.1157 on 38 degrees of freedom
## Multiple R-squared:  0.836,  Adjusted R-squared:  0.8231
## F-statistic: 64.59 on 3 and 38 DF,  p-value: 5.54e-15
```

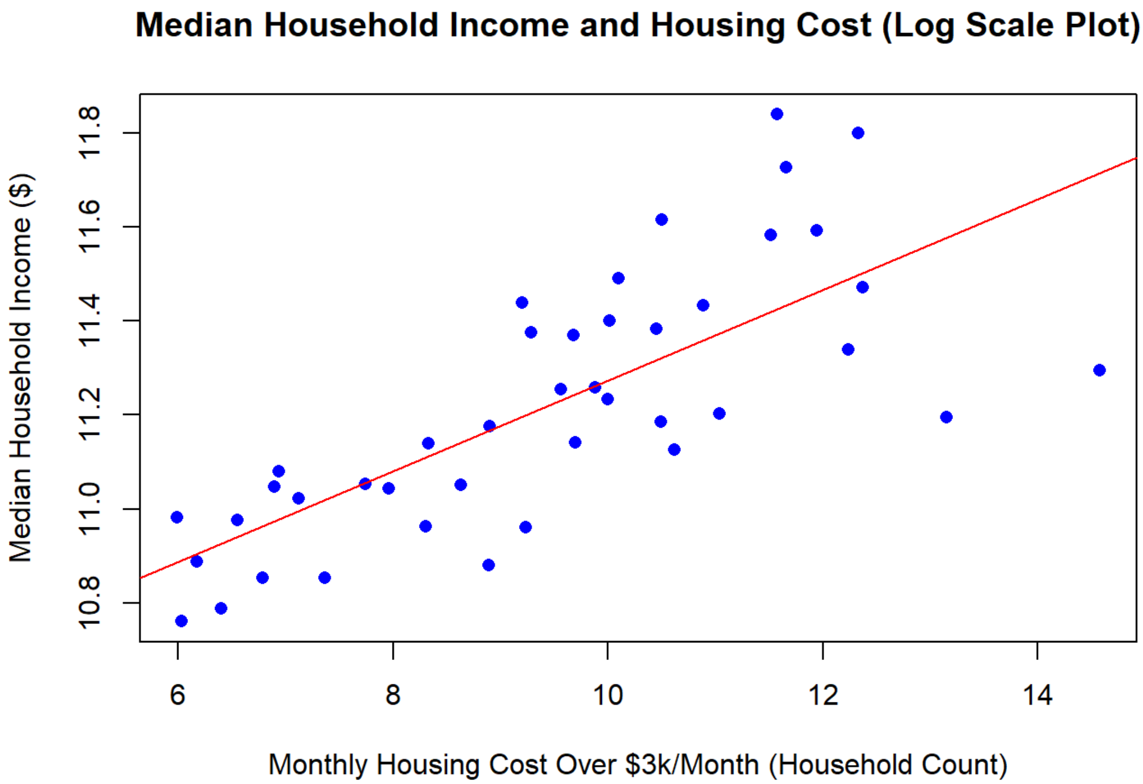
```
# anova analysis to identify change
anova(
    fit_acs19_housing_3000,
    fit_acs19_housing_2500,
    fit_acs19_housing_2000
)
```

```
## 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 1.37241
## 2     39 0.85806  1   0.51435 38.424 3.040e-07 ***
## 3     38 0.50868  1   0.34938 26.100 9.447e-06 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

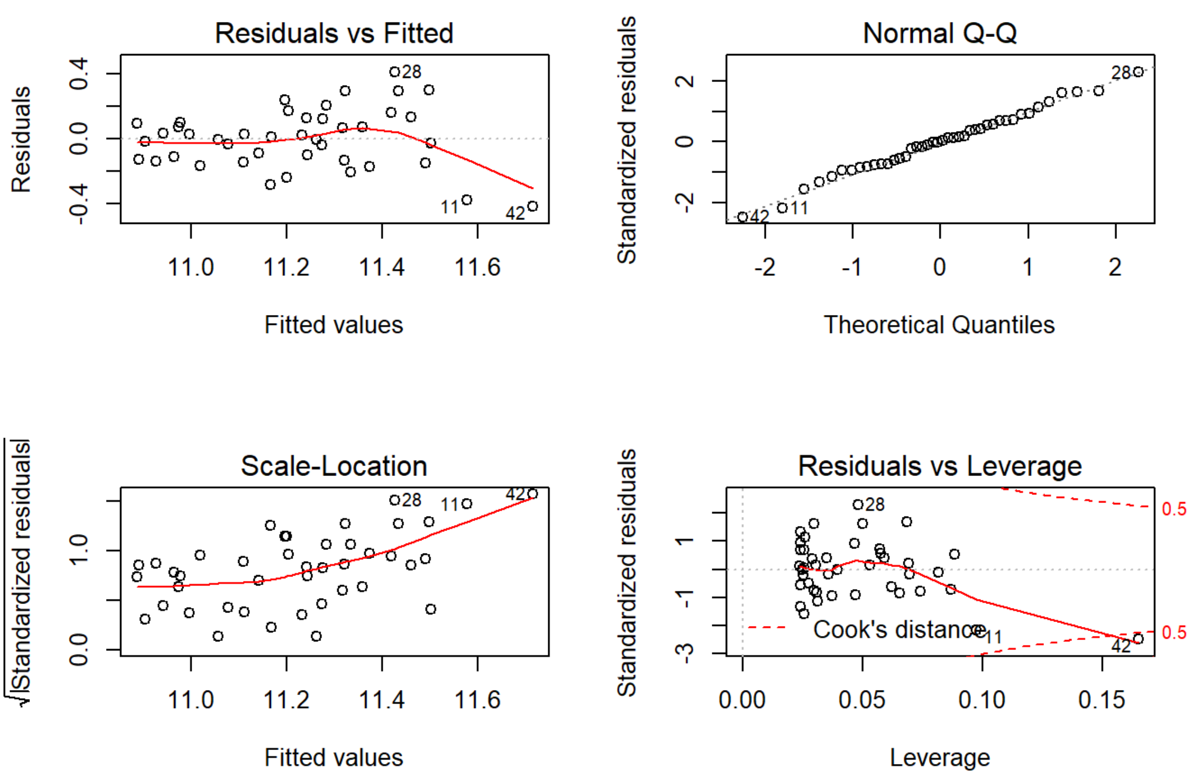
```
# calculate correlation
col_housing = c(
    "housing_more_3000",
    "housing_2500",
    "housing_2000"
)
cor(acs19_nofactor$hh_median, acs19_nofactor[col_housing])
```

```
##      housing_more_3000 housing_2500 housing_2000
## [1,]         0.1383946   0.09838385   0.07478075
```

```
# visualize income and transit variables
# https://www.theanalysisfactor.com/linear-models-r-plotting-regression-lines/
plot(log(hh_median) ~ log(as.numeric(housing_more_3000)),
    main="Median Household Income and Housing Cost (Log Scale Plot)",
    xlab="Monthly Housing Cost Over $3k/Month (Household Count)",
    ylab="Median Household Income ($)",
    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. In particular, education attainment including and beyond a bachelors degree was more closely associated with median household income.

This finding implies that certain higher education attainment is associated with median household income. When data features were added the model, most maintained relatively lower p-values. In particular, higher levels of education (e.g. doctorate degree) generally had a positive relationship with median household income.

```
# 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)        10.472282   0.135864 77.078918    0e+00
## log(edu_doctorate)  0.092182   0.016270  5.665822    1e-06
```

```
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)           10.487339   0.218387 48.021761 0.000000
## log(edu_doctorate)     0.097300   0.059954  1.622916 0.112665
## log(edu_professional) -0.006443   0.072572 -0.088778 0.929713
```

```
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)           10.629329   0.332720 31.946779 0.000000
## log(edu_doctorate)     0.115854   0.068697  1.686438 0.099906
## log(edu_professional)  0.046016   0.117667  0.391073 0.697928
## log(edu_master)       -0.075171   0.132001 -0.569472 0.572387
```

```
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)           11.463659   0.391705 29.266026 0.000000
## log(edu_doctorate)     0.136440   0.061653  2.213014 0.033145
## log(edu_professional)  0.086179   0.105766  0.814803 0.420399
## log(edu_master)        0.320078   0.168878  1.895317 0.065883
## log(edu_bachelor)     -0.483599   0.148002 -3.267511 0.002346
```

```
# 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
## -0.26914 -0.13845 -0.01585  0.14502  0.31568
##
## Coefficients:
##                       Estimate Std. Error t value Pr(>|t|)
## (Intercept)           11.46366    0.39171  29.266  < 2e-16 ***
## log(edu_doctorate)     0.13644    0.06165   2.213  0.03315 *
## log(edu_professional)  0.08618    0.10577   0.815  0.42040
## log(edu_master)        0.32008    0.16888   1.895  0.06588 .
## log(edu_bachelor)     -0.48360    0.14800  -3.268  0.00235 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.1892 on 37 degrees of freedom
## Multiple R-squared:  0.5732, Adjusted R-squared:  0.527
## F-statistic: 12.42 on 4 and 37 DF,  p-value: 1.675e-06
```

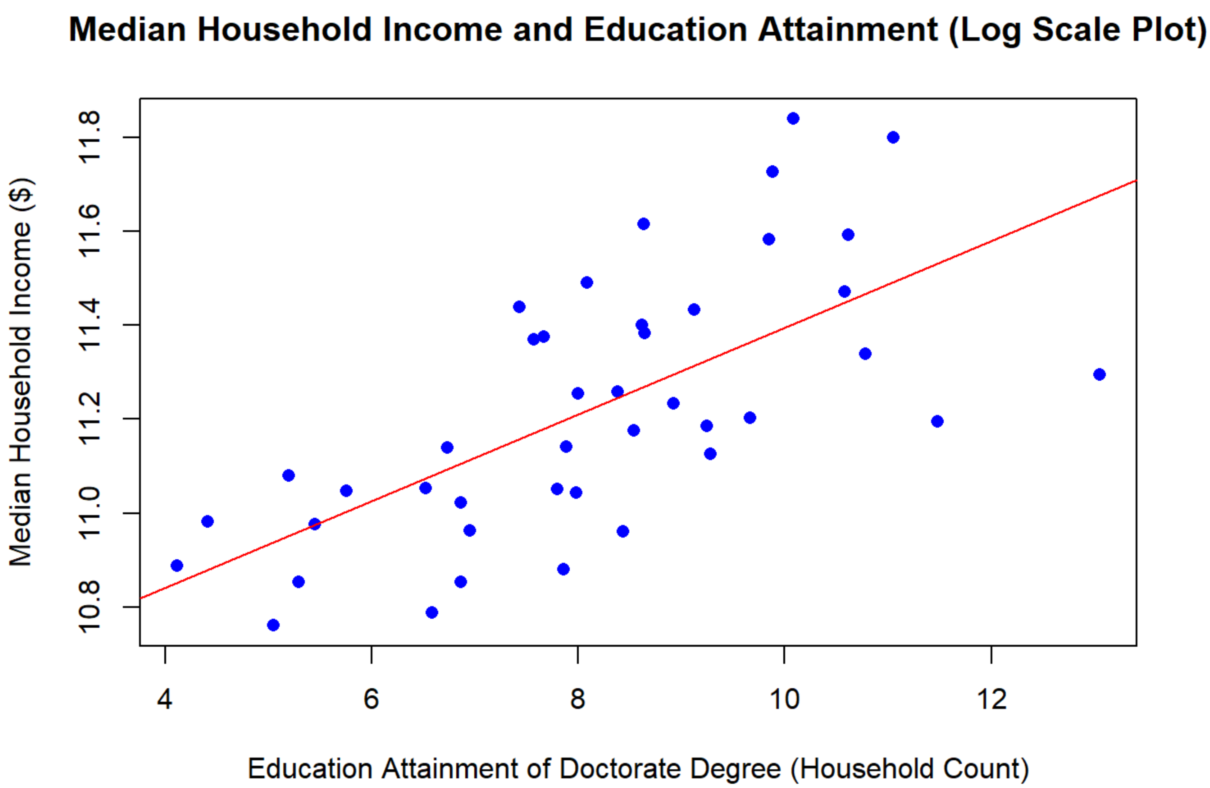
```
# anova analysis to identify change
anova(
    fit_acs19_edu_doctorate,
    fit_acs19_edu_professional,
    fit_acs19_edu_master,
    fit_acs19_edu_bachelor
)
```

```
## 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 1.7212
## 2     39 1.7208  1   0.00035  0.0097 0.922007
## 3     38 1.7063  1   0.01456  0.4069 0.527485
## 4     37 1.3242  1   0.38210 10.6766 0.002346 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

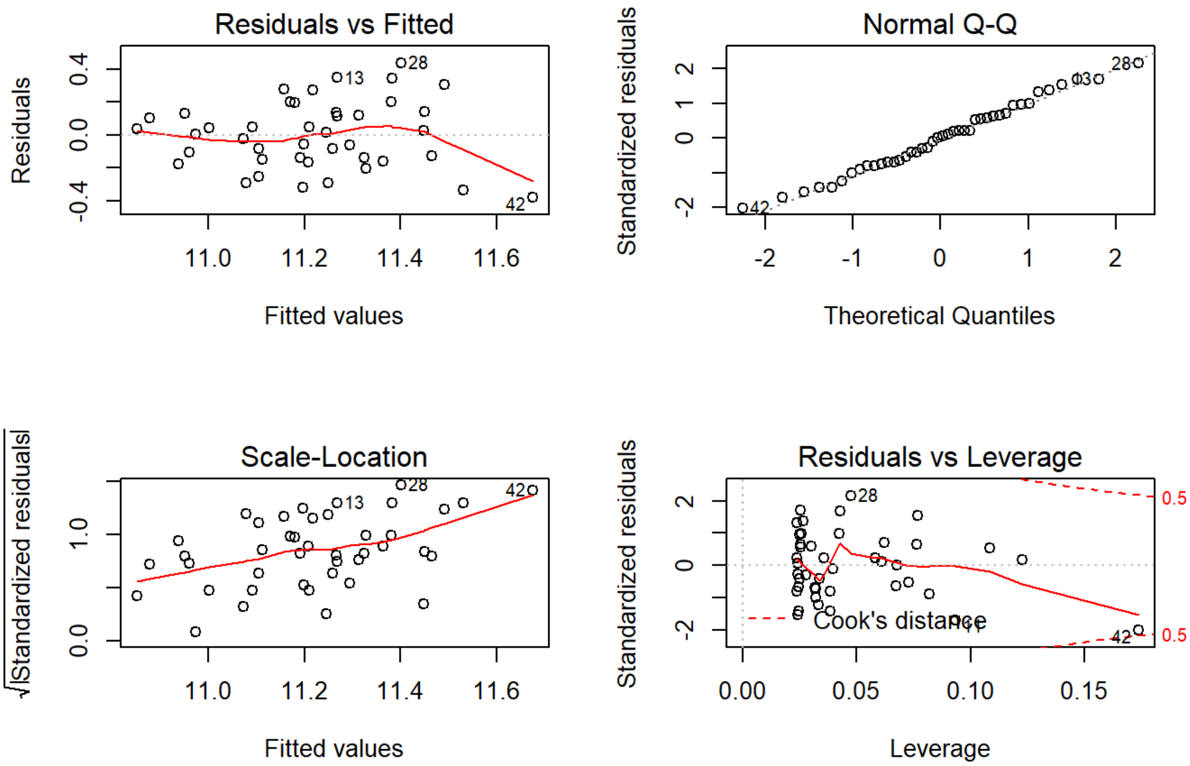
```
# calculate correlation
col_edu = c(
    "edu_doctorate",
    "edu_professional",
    "edu_master",
    "edu_bachelor"
)
cor(acs19_nofactor$hh_median, acs19_nofactor[col_edu])
```

```
##      edu_doctorate edu_professional edu_master edu_bachelor
## [1,]     0.1412332        0.1003801  0.1182039   0.09137493
```

```
# visualize income and transit variables
# https://www.theanalysisfactor.com/linear-models-r-plotting-regression-lines/
plot(log(hh_median ) ~ log(as.numeric(edu_doctorate)),
    main="Median Household Income and Education Attainment (Log Scale Plot)",
    xlab="Education Attainment of Doctorate Degree (Household Count)",
    ylab="Median Household Income ($)",
    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 “American Indian”, “Alaska Native alone” and “Asian alone” 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)                   10.191937   0.402524 25.320072 0.000000
## log(as.numeric(B02001_002E))  0.081089   0.031581  2.567625 0.014083
```

```
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)                   10.467179   0.584933 17.894673 0.000000
## log(as.numeric(B02001_002E))  0.035602   0.076595  0.464811 0.644651
## log(as.numeric(B02001_003E))  0.031754   0.048641  0.652821 0.517701
```

```
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)                    9.824854   0.495952 19.810088 0.000000
## log(as.numeric(B02001_002E))  0.251561   0.078132  3.219709 0.002628
## log(as.numeric(B02001_003E))  0.051680   0.039796  1.298597 0.201908
## log(as.numeric(B02001_004E)) -0.272604   0.059534 -4.578978 0.000049
```

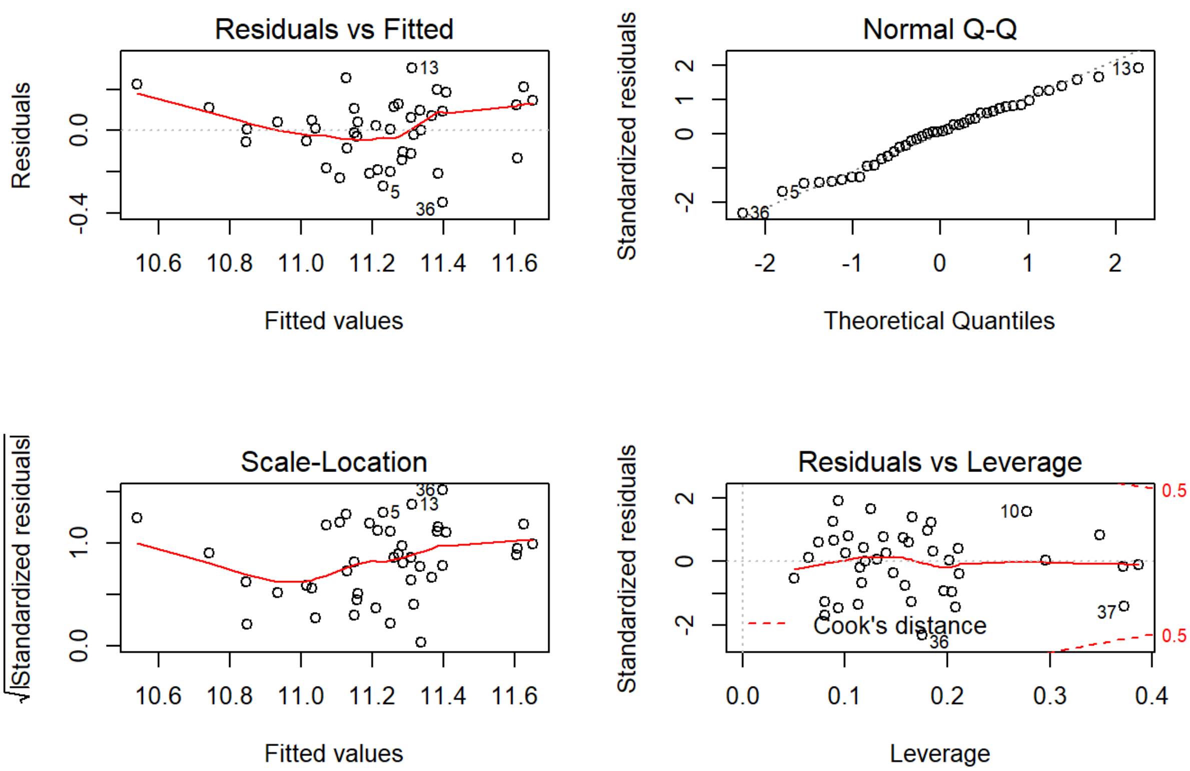
```
fit_acs19_race_all <- lm(
    log(hh_median) ~ log(as.numeric(B02001_002E)) +
    log(as.numeric(B02001_003E)) +
    log(as.numeric(B02001_004E)) +
    log(as.numeric(B02001_005E)) +

    # 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)                   10.631479   0.432333 24.590947 0.000000
## log(as.numeric(B02001_002E))  0.112424   0.090269  1.245438 0.221244
## log(as.numeric(B02001_003E)) -0.055093   0.048654 -1.132341 0.265189
## log(as.numeric(B02001_004E)) -0.150160   0.053787 -2.791762 0.008436
## log(as.numeric(B02001_005E))  0.228252   0.046353  4.924258 0.000020
## log(as.numeric(B02001_007E)) -0.013888   0.033885 -0.409862 0.684405
## log(as.numeric(B02001_008E)) -0.127890   0.093397 -1.369323 0.179621
```

```r
# 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
## -0.34967 -0.11142  0.00936  0.10986  0.30295
##
## Coefficients:
##                              Estimate Std. Error t value Pr(>|t|)
## (Intercept)                  10.63148    0.43233  24.591  < 2e-16 ***
## log(as.numeric(B02001_002E))  0.11242    0.09027   1.245  0.22124
## log(as.numeric(B02001_003E)) -0.05509    0.04865  -1.132  0.26519
## log(as.numeric(B02001_004E)) -0.15016    0.05379  -2.792  0.00844 **
## log(as.numeric(B02001_005E))  0.22825    0.04635   4.924 2.02e-05 ***
## log(as.numeric(B02001_007E)) -0.01389    0.03388  -0.410  0.68441
## log(as.numeric(B02001_008E)) -0.12789    0.09340  -1.369  0.17962
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.1668 on 35 degrees of freedom
## Multiple R-squared:  0.6862, Adjusted R-squared:  0.6324
## F-statistic: 12.76 on 6 and 35 DF,  p-value: 1.384e-07
```

```r
# 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. However when either of the two features are isolated, then their fit is poor so poverty is not particularly useful for the analysis.

The poor fit is likely due to the large size of each group (above or below poverty income level) which basically divides the dataset into two halves and results in poor fit due to large variance of each subset. Poverty level is intuitively a good indicator of median household income; however, the feature is not granular enough to use for additional analysis.

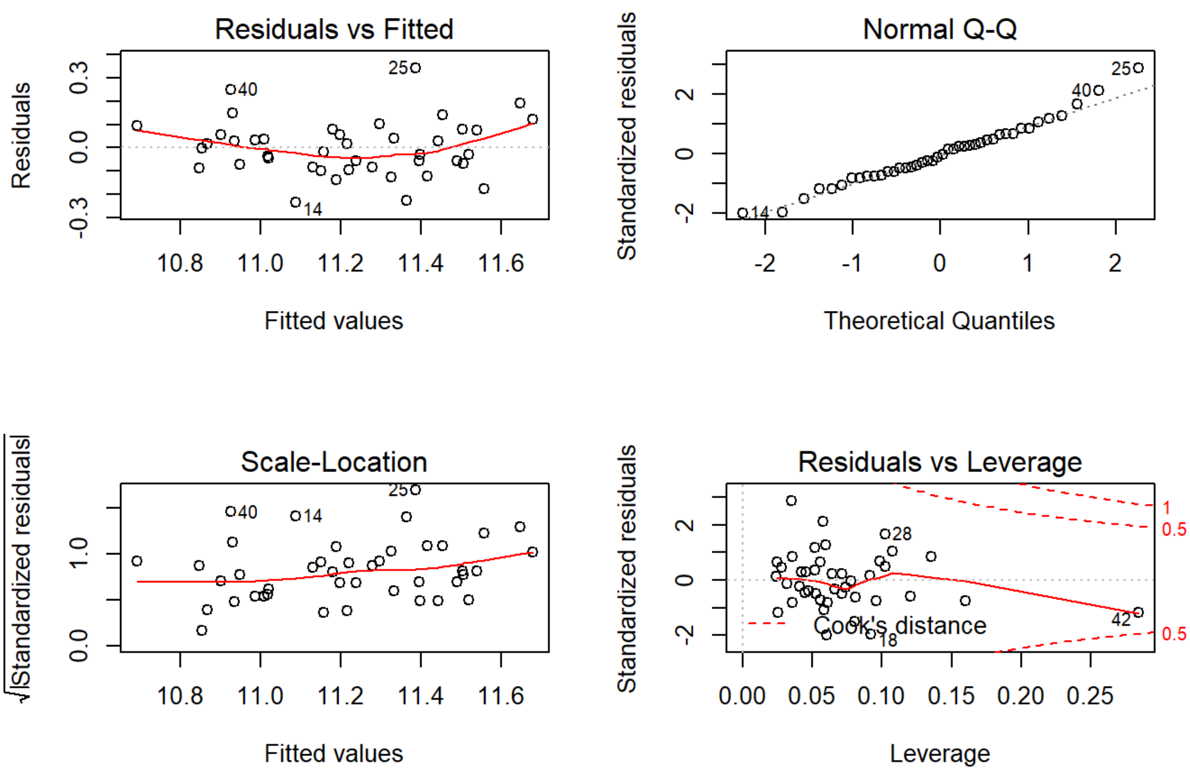
```
# features:
# B17001_002E: Income in the past 12 months below poverty level
# B17001_031E: Income in the past 12 months at or above poverty level

# 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
## -0.38885 -0.17788 -0.04469  0.15596  0.56923
##
## Coefficients:
##                    Estimate Std. Error t value Pr(>|t|)
## (Intercept)        10.08532    0.35860  28.124  < 2e-16 ***
## log(poverty_above)  0.08784    0.02759   3.184  0.00281 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.2488 on 40 degrees of freedom
## Multiple R-squared:  0.2022, Adjusted R-squared:  0.1822
## F-statistic: 10.14 on 1 and 40 DF,  p-value: 0.002815
```

```
# 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
## -0.23382 -0.08078 -0.00965  0.07180  0.34082
##
## Coefficients:
##                    Estimate Std. Error t value Pr(>|t|)
## (Intercept)         9.62080    0.17908   53.73  < 2e-16 ***
## log(poverty_below) -0.53143    0.04658  -11.41 5.40e-14 ***
## log(poverty_above)  0.57319    0.04461   12.85 1.35e-15 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.121 on 39 degrees of freedom
## Multiple R-squared:  0.816,  Adjusted R-squared:  0.8066
## F-statistic:  86.5 on 2 and 39 DF,  p-value: 4.593e-15
```

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



2019 ACS - Model Refinement

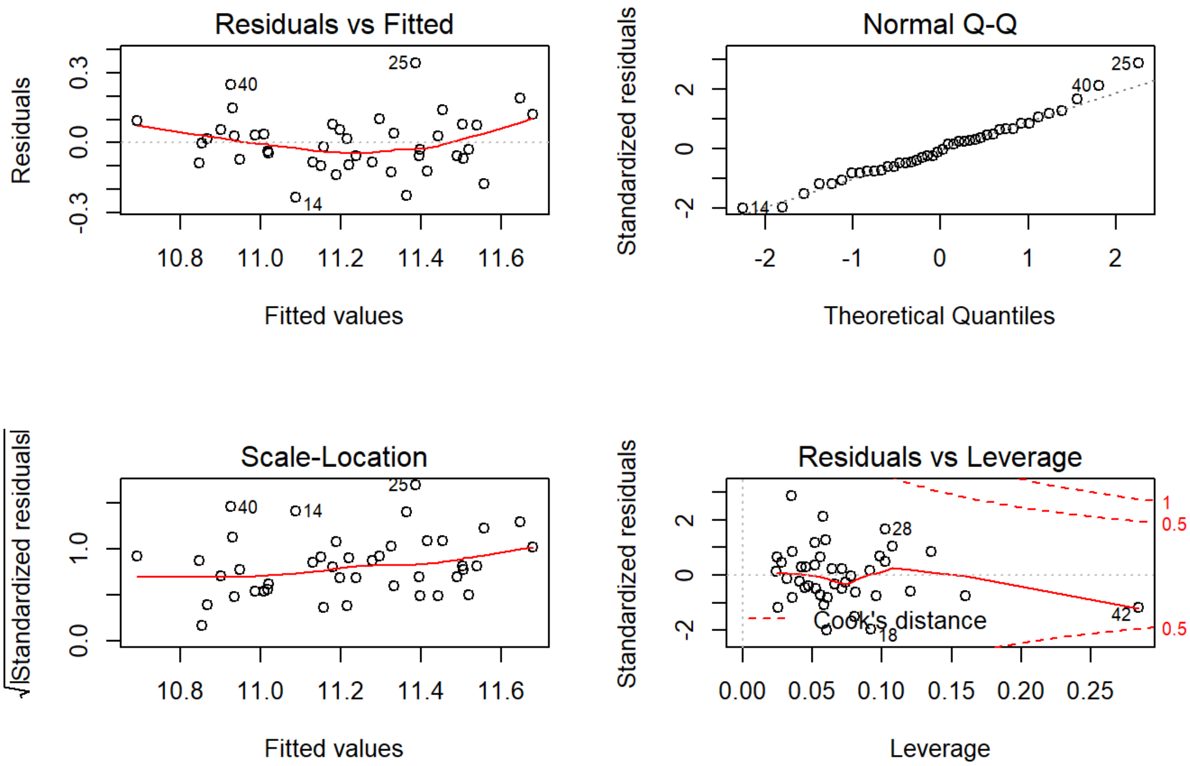
> Improve linear model with best features and analyze results.

The linear regression model was refined by fitting the best attributes from the data analysis 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_transit)) +
    log(as.numeric(commute_remote)) +
    log(as.numeric(commute_carpool)) +
    log(as.numeric(edu_doctorate)) +
    log(as.numeric(edu_professional)) +
    log(as.numeric(edu_master)) +
    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_transit)) + log(as.numeric(commute_remote)) +
##     log(as.numeric(commute_carpool)) + log(as.numeric(edu_doctorate)) +
##     log(as.numeric(edu_professional)) + log(as.numeric(edu_master)) +
##     log(as.numeric(edu_bachelor)), data = acs19_nofactor)
##
## Residuals:
##       Min        1Q    Median        3Q       Max
## -0.248800 -0.068944 -0.009409  0.083232  0.258076
##
## Coefficients:
##                                    Estimate Std. Error t value Pr(>|t|)
## (Intercept)                        11.42106    0.32394  35.256  < 2e-16 ***
## log(as.numeric(housing_more_3000))  0.37176    0.05926   6.273 4.33e-07 ***
## log(as.numeric(commute_transit))    0.02173    0.02796   0.777   0.4426
## log(as.numeric(commute_remote))     0.03987    0.03046   1.309   0.1997
## log(as.numeric(commute_carpool))   -0.03750    0.02837  -1.322   0.1953
## log(as.numeric(edu_doctorate))     -0.01144    0.04929  -0.232   0.8178
## log(as.numeric(edu_professional))  -0.08852    0.08079  -1.096   0.2811
## log(as.numeric(edu_master))        -0.05156    0.13484  -0.382   0.7046
## log(as.numeric(edu_bachelor))      -0.21784    0.11195  -1.946   0.0602 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.1324 on 33 degrees of freedom
## Multiple R-squared:  0.8137, Adjusted R-squared:  0.7685
## F-statistic: 18.01 on 8 and 33 DF,  p-value: 5.374e-10
```

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



2019 ACS - Disadvantaged Communities

> This section identifies disadvantaged communities with a subset of the ACS 2019 dataset of communities with the 20% lowest percentile of median household income.

The 20% quantile of median household income was selected as a threshold to identify disadvantaged communities based on the CA HUD [low income guidelines][3.00] 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.

California communities with the lowest median household income based the 2019 ACS dataset are listed below in rank order:

1. Tehama County, California ($25,310/year)
2. Lake County, California ($25,968/year)
3. Yuba County, California ($26,827/year)
4. Sutter County, California ($33,039/year)
5. Mendocino County, California ($34,478/year)
6. Nevada County, California ($39,365/year)
7. Madera County, California ($44,387/year)
8. Kings County, California ($44,761/year)
9. Napa County, California ($48,107/year)

```
# evaluate median household income
# hh_median_sw = median(acs19_nofactor$hh_median)
# hh_median_sw

# subset for 0.20 quantile per CA HUD guidelines
# https://www.hcd.ca.gov/grants-funding/income-limits/state-and-federal-income-limits/docs/income-limits-20
20.pdf
hh_median_qt20 = quantile(
    as.numeric(as.character(acs19_nofactor$hh_median)), 0.2
)
hh_median_qt20
```

```
##     20%
## 58515.4
```

```
# subset for 0.80 quantile per CA HUD guidelines
# https://www.hcd.ca.gov/grants-funding/income-limits/state-and-federal-income-limits/docs/income-limits-20
20.pdf
hh_median_qt80 = quantile(
    as.numeric(as.character(acs19_nofactor$hh_median)), 0.8
)
hh_median_qt80
```

```
##     80%
## 92662.4
```

```
# 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)))
```

```
##                           NAME B19001_001E
## 1    Tehama County, California       25310
## 2      Lake County, California       25968
## 3      Yuba County, California       26827
## 4    Sutter County, California       33039
## 5 Mendocino County, California       34478
## 6    Nevada County, California       39365
## 7    Madera County, California       44387
## 8     Kings County, California       44761
## 9      Napa County, California       48107
```

```
# 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)))
```

```
##                                NAME B19001_001E
## 1     Sacramento County, California      556752
## 2        Alameda County, California      585632
## 3    Santa Clara County, California      643637
## 4 San Bernardino County, California      644758
## 5      Riverside County, California      734948
## 6         Orange County, California     1044280
## 7      San Diego County, California     1132434
## 8    Los Angeles County, California     3328398
## 9                        California    13157873
```

```
# 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)
```

NCHS COVID-19 Data

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

The NHCS COVID-19 [mortality count data][3.00] 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://www.latimes.com/projects/california-coronavirus-cases-tracking-outbreak/) 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 leves 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, California (3328398)
2. San Diego County, California (1132434)
3. Orange County, California (1044280)
4. Riverside County, California (734948)
5. San Bernardino County, California (644758)
6. Santa Clara County, California (643637)
7. Alameda County, California (585632)
8. Sacramento County, California (556752)

```
# 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 above 0.80 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.COV
ID.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))
```

```
##   State      County.name Urban.Rural.Code Deaths.involving.COVID.19
## 1    CA    Amador County          Noncore                        29
## 2    CA      Napa County      Small metro                        23
## 3    CA      Lake County     Micropolitan                        19
## 4    CA      Yuba County      Small metro                        19
## 5    CA Mendocino County     Micropolitan                        18
## 6    CA    Tehama County     Micropolitan                        17
## 7    CA      Inyo County          Noncore                        14
## 8    CA    Nevada County     Micropolitan                        12
## 9    CA Calaveras County          Noncore                        11
```

```
# filter for mortality count above 0.20 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.COV
ID.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

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

Key findings are as follows:

1. 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.

Data analysis findings which may be applied to other U.S. regions are as follows: * Some features have a wide variance and outlier/high leverage points; as a result, log transformation is recommended to reduce their impact. * 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
2. Histogram to Evaluate Skew
3. Logistic Regression for Median Income and Location
4. Diagnostic Plot and Analysis of Logistic Regression