Travis Darby Linear Models Project (all steps)

```
#library imports
library(tidyverse)
```

```
## -- Attaching core tidyverse packages ------------------------ tidyverse 2.0.0 --
## v dplyr     1.1.2     v readr     2.1.4
## v forcats   1.0.0     v stringr   1.5.0
## v ggplot2   3.4.3     v tibble    3.2.1
## v lubridate 1.9.2     v tidyr     1.3.0
## v purrr     1.0.2
## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
## i Use the conflicted package (<http://conflicted.r-lib.org/>) to force all conflicts to become error
```

```
#importing the data
df <- read_csv("../Project_Data/DS64510_Project_Data.csv")
```

```
## Rows: 3047 Columns: 30
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr  (2): binnedInc, Geography
## dbl (28): avgAnnCount, avgDeathsPerYear, TARGET_deathRate, incidenceRate, me...
##
## i Use `spec()` to retrieve the full column specification for this data.
## i Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

```
#viewing the data
glimpse(df)
```

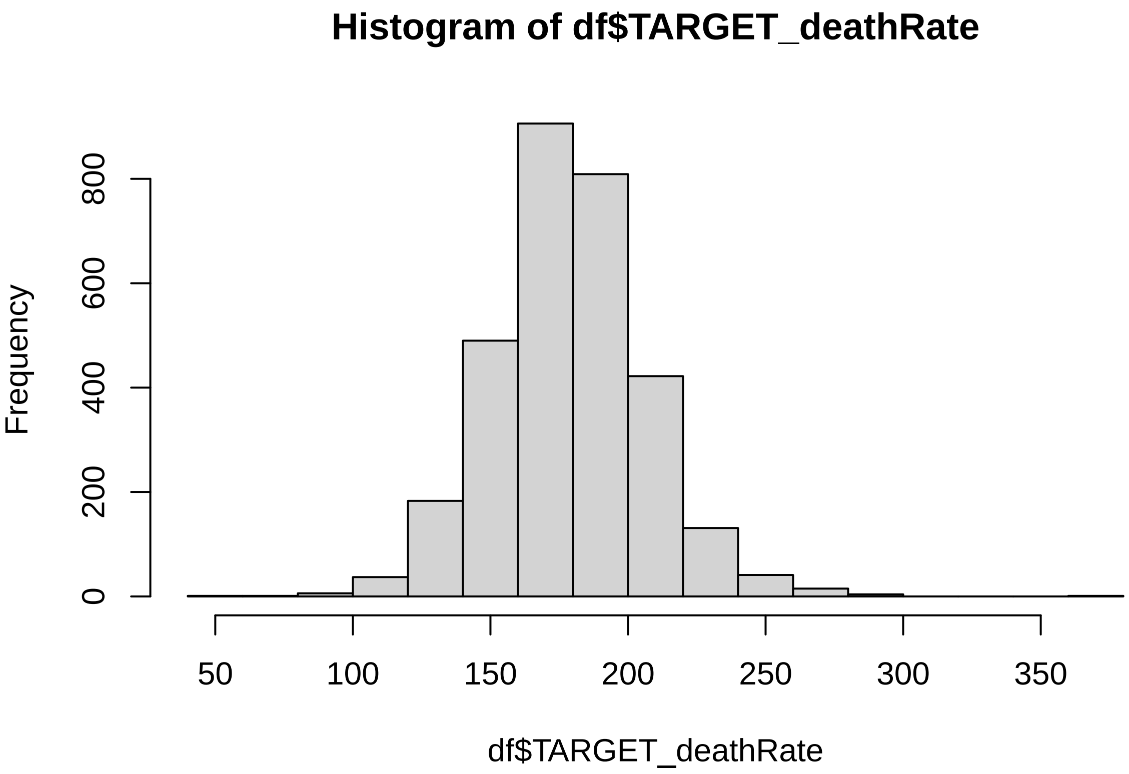
```
## Rows: 3,047
## Columns: 30
## $ avgAnnCount             <dbl> 1397, 173, 102, 427, 57, 428, 250, 146, 88, 40~
## $ avgDeathsPerYear        <dbl> 469, 70, 50, 202, 26, 152, 97, 71, 36, 1380, 3~
## $ TARGET_deathRate        <dbl> 164.9, 161.3, 174.7, 194.8, 144.4, 176.0, 175.~
## $ incidenceRate           <dbl> 489.8, 411.6, 349.7, 430.4, 350.1, 505.4, 461.~
## $ medIncome               <dbl> 61898, 48127, 49348, 44243, 49955, 52313, 3778~
## $ popEst2015              <dbl> 260131, 43269, 21026, 75882, 10321, 61023, 415~
## $ povertyPercent          <dbl> 11.2, 18.6, 14.6, 17.1, 12.5, 15.6, 23.2, 17.8~
## $ studyPerCap             <dbl> 499.74820, 23.11123, 47.56016, 342.63725, 0.00~
## $ binnedInc               <chr> "(61494.5, 125635]", "(48021.6, 51046.4]", "(4~
## $ MedianAge               <dbl> 39.3, 33.0, 45.0, 42.8, 48.3, 45.4, 42.6, 51.7~
## $ MedianAgeMale           <dbl> 36.9, 32.2, 44.0, 42.2, 47.8, 43.5, 42.2, 50.8~
## $ MedianAgeFemale         <dbl> 41.7, 33.7, 45.8, 43.4, 48.9, 48.0, 43.5, 52.5~
## $ Geography               <chr> "Kitsap County, Washington", "Kittitas County,~
## $ AvgHouseholdSize        <dbl> 2.5400, 2.3400, 2.6200, 2.5200, 2.3400, 2.5800~
## $ PercentMarried          <dbl> 52.5, 44.5, 54.2, 52.7, 57.8, 50.4, 54.1, 52.7~
## $ PctNoHS18_24            <dbl> 11.5, 6.1, 24.0, 20.2, 14.9, 29.9, 26.1, 27.3,~
## $ PctHS18_24              <dbl> 39.5, 22.4, 36.6, 41.2, 43.0, 35.1, 41.4, 33.9~
## $ PctSomeCol18_24         <dbl> 42.1, 64.0, NA, 36.1, 40.0, NA, NA, 36.5, NA, ~
## $ PctBachDeg18_24         <dbl> 6.9, 7.5, 9.5, 2.5, 2.0, 4.5, 5.8, 2.2, 1.4, 7~
## $ PctHS25_Over            <dbl> 23.2, 26.0, 29.0, 31.6, 33.4, 30.4, 29.8, 31.6~
## $ PctBachDeg25_Over       <dbl> 19.6, 22.7, 16.0, 9.3, 15.0, 11.9, 11.9, 11.3,~
## $ PctEmployed16_Over      <dbl> 51.9, 55.9, 45.9, 48.3, 48.2, 44.1, 51.8, 40.9~
## $ PctUnemployed16_Over    <dbl> 8.0, 7.8, 7.0, 12.1, 4.8, 12.9, 8.9, 8.9, 10.3~
## $ PctPrivateCoverage      <dbl> 75.1, 70.2, 63.7, 58.4, 61.6, 60.0, 49.5, 55.8~
## $ PctPrivateCoverageAlone <dbl> NA, 53.8, 43.5, 40.3, 43.9, 38.8, 35.0, 33.1, ~
## $ PctEmpPrivCoverage      <dbl> 41.6, 43.6, 34.9, 35.0, 35.1, 32.6, 28.3, 25.9~
## $ PctPublicCoverage       <dbl> 32.9, 31.1, 42.1, 45.3, 44.0, 43.2, 46.4, 50.9~
## $ PctPublicCoverageAlone  <dbl> 14.0, 15.3, 21.1, 25.0, 22.7, 20.2, 28.7, 24.1~
## $ PctMarriedHouseholds    <dbl> 52.85608, 45.37250, 54.44487, 51.02151, 54.027~
## $ BirthRate               <dbl> 6.1188310, 4.3330956, 3.7294878, 4.6038408, 6.~
```

Step 1

```
#determining which variables have missing data and how much
sapply(df, function(x) sum(is.na(x)))
```

```
##             avgAnnCount        avgDeathsPerYear        TARGET_deathRate 
##                       0                       0                       0 
##           incidenceRate               medIncome              popEst2015 
##                       0                       0                       0 
##          povertyPercent             studyPerCap               binnedInc 
##                       0                       0                       0 
##               MedianAge           MedianAgeMale         MedianAgeFemale 
##                       0                       0                       0 
##               Geography        AvgHouseholdSize          PercentMarried 
##                       0                       0                       0 
##            PctNoHS18_24              PctHS18_24         PctSomeCol18_24 
##                       0                       0                    2285 
##         PctBachDeg18_24            PctHS25_Over       PctBachDeg25_Over 
##                       0                       0                       0 
##      PctEmployed16_Over    PctUnemployed16_Over      PctPrivateCoverage 
##                     152                       0                       0 
## PctPrivateCoverageAlone      PctEmpPrivCoverage       PctPublicCoverage 
##                     609                       0                       0 
##  PctPublicCoverageAlone    PctMarriedHouseholds               BirthRate 
##                       0                       0                       0
```

```
#viewing the distribution of the Target variable
hist(df$TARGET_deathRate)
```



Varaible selection:

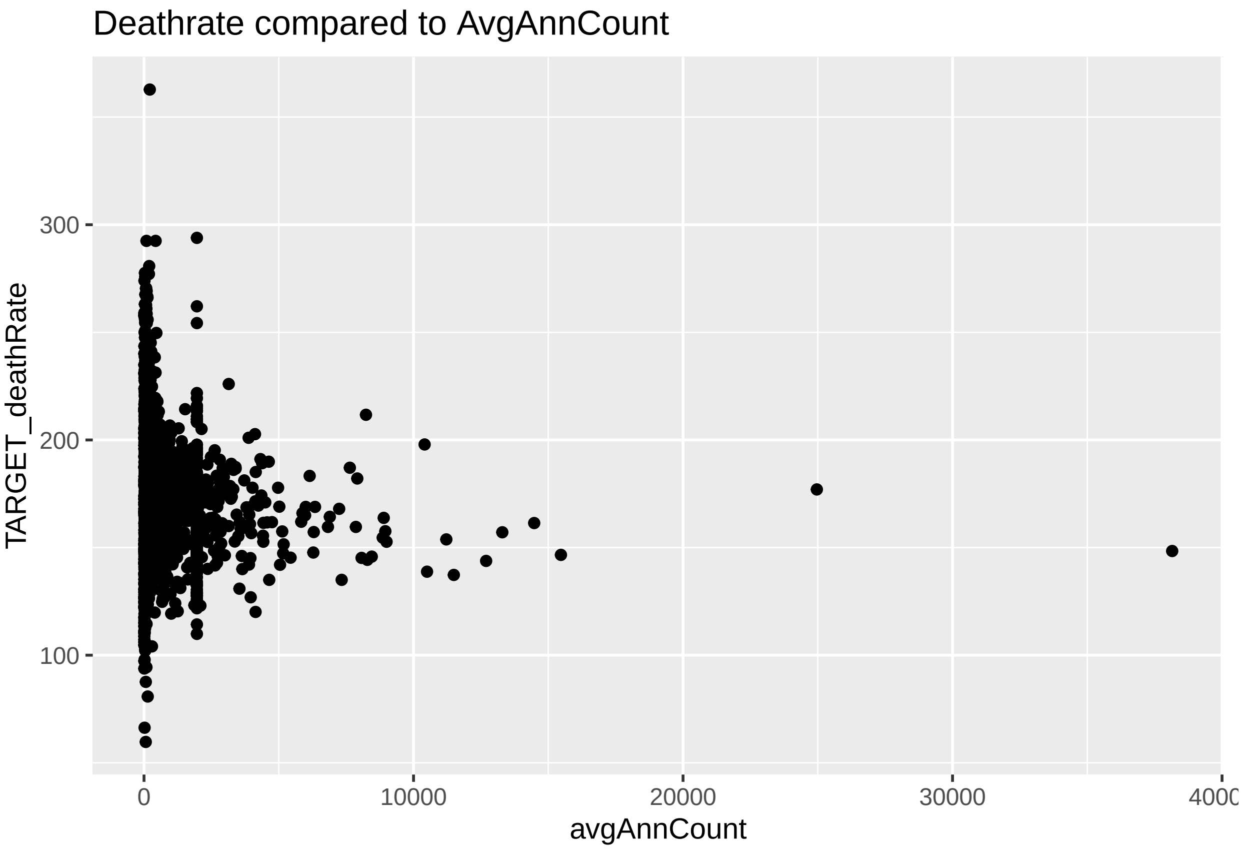
Initially, I will explore the following variables in order to put them in to a model in future steps:

```
- avgAnnCount
- IncidenceRate
- MedianIncome
- MedianAge
- studyPerCap
- PctHS25_Over
- PctPrivateCoverage
- PctPublicCoverage
```

AvgAnnCount

Mean number of reported cases of cancer diagnosed annually

```r
df %>% ggplot(aes(x=avgAnnCount, y=TARGET_deathRate)) + geom_point() +
  ggtitle("Deathrate compared to AvgAnnCount")
```



There does not appear to be a positive or negative relationship, the data looks to be centered around 200 for the deathrate with values ± 100 on the y-axis.

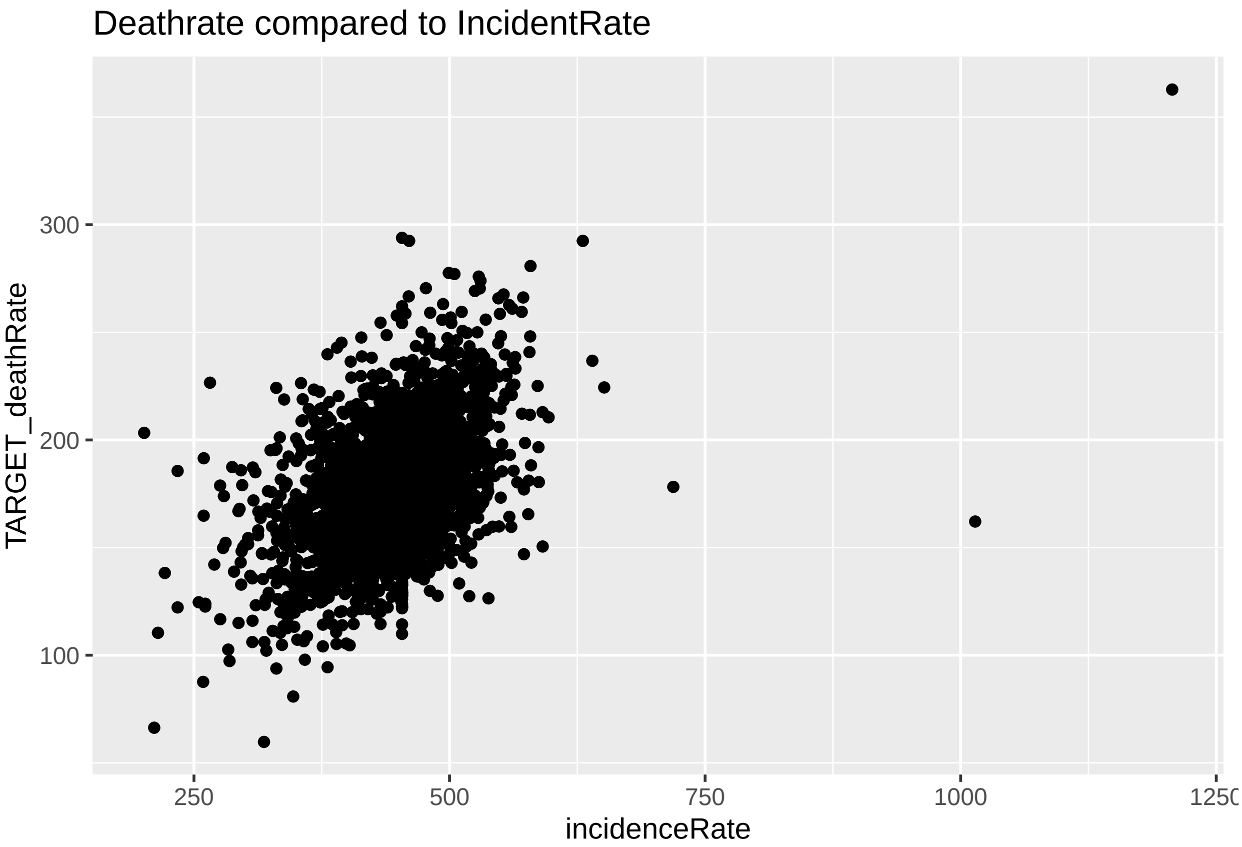
```
#overall summary of the the variable
summary(df$avgAnnCount)
```

```
##     Min.  1st Qu.   Median     Mean  3rd Qu.     Max. 
##      6.0     76.0    171.0    606.3    518.0  38150.0
```

Incidence Rate

Mean per capita (100,000) cancer diagnoses

```
df %>% ggplot(aes(x=incidenceRate, y= TARGET_deathRate)) + geom_point() +
  ggtitle("Deathrate compared to IncidentRate")
```



There appears to be a strong positive relationship between IncidenceRate and Deathrate

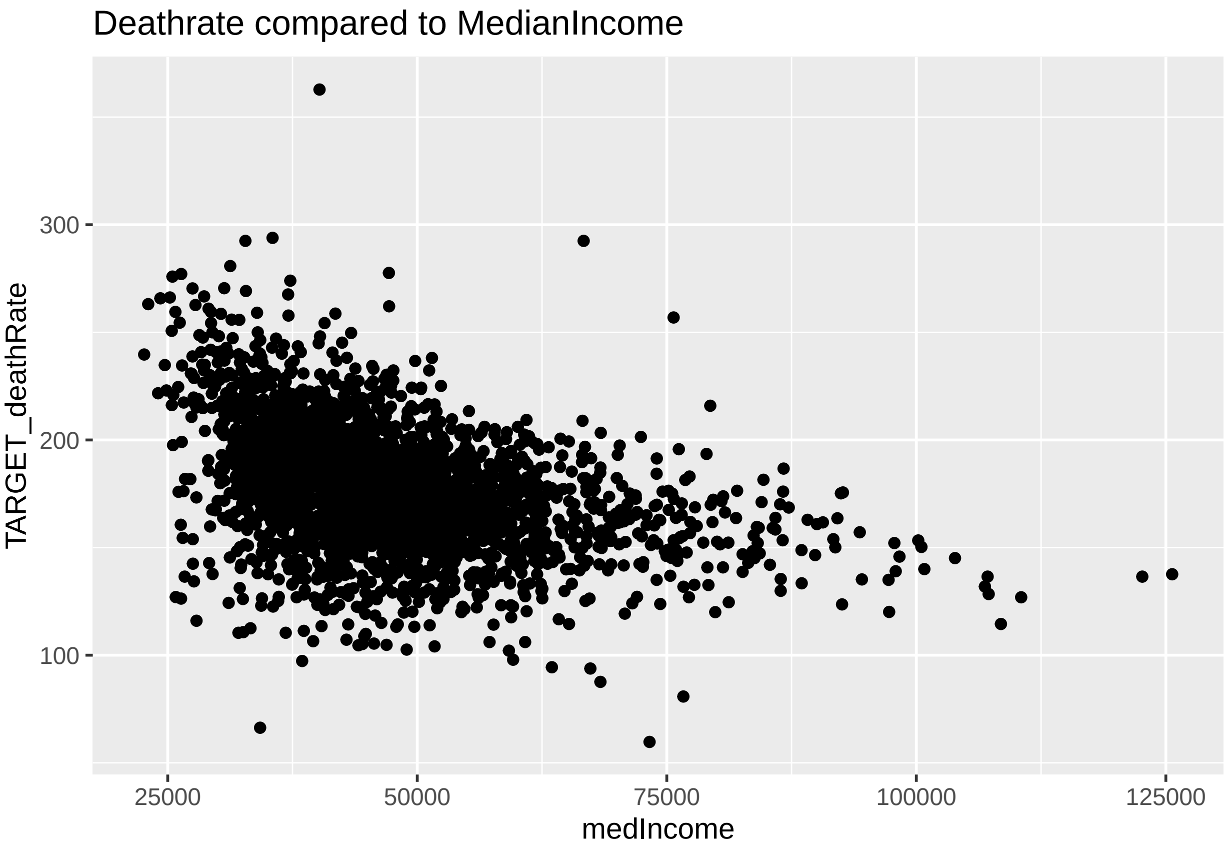
```r
summary(df$incidenceRate)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max. 
##   201.3   420.3   453.5   448.3   480.9  1206.9
```

Median Income

Median income per county

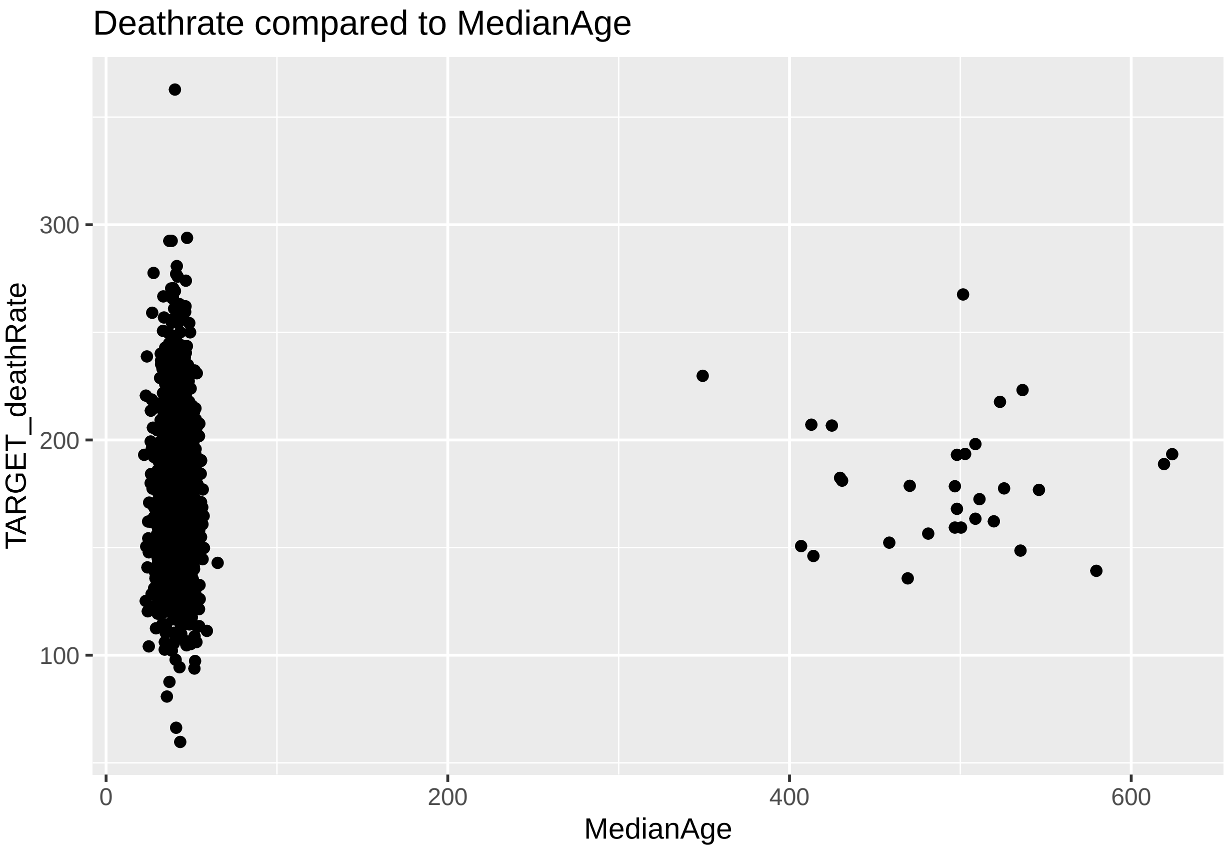
```r
df %>% ggplot(aes(x=medIncome, y=TARGET_deathRate)) + geom_point() +
  ggtitle("Deathrate compared to MedianIncome")
```



There looks to be a moderate negative relationship btween MedianIncome and Deathrate.

Median age

```r
df %>% ggplot(aes(x=MedianAge, y=TARGET_deathRate)) + geom_point() +
  ggtitle("Deathrate compared to MedianAge")
```

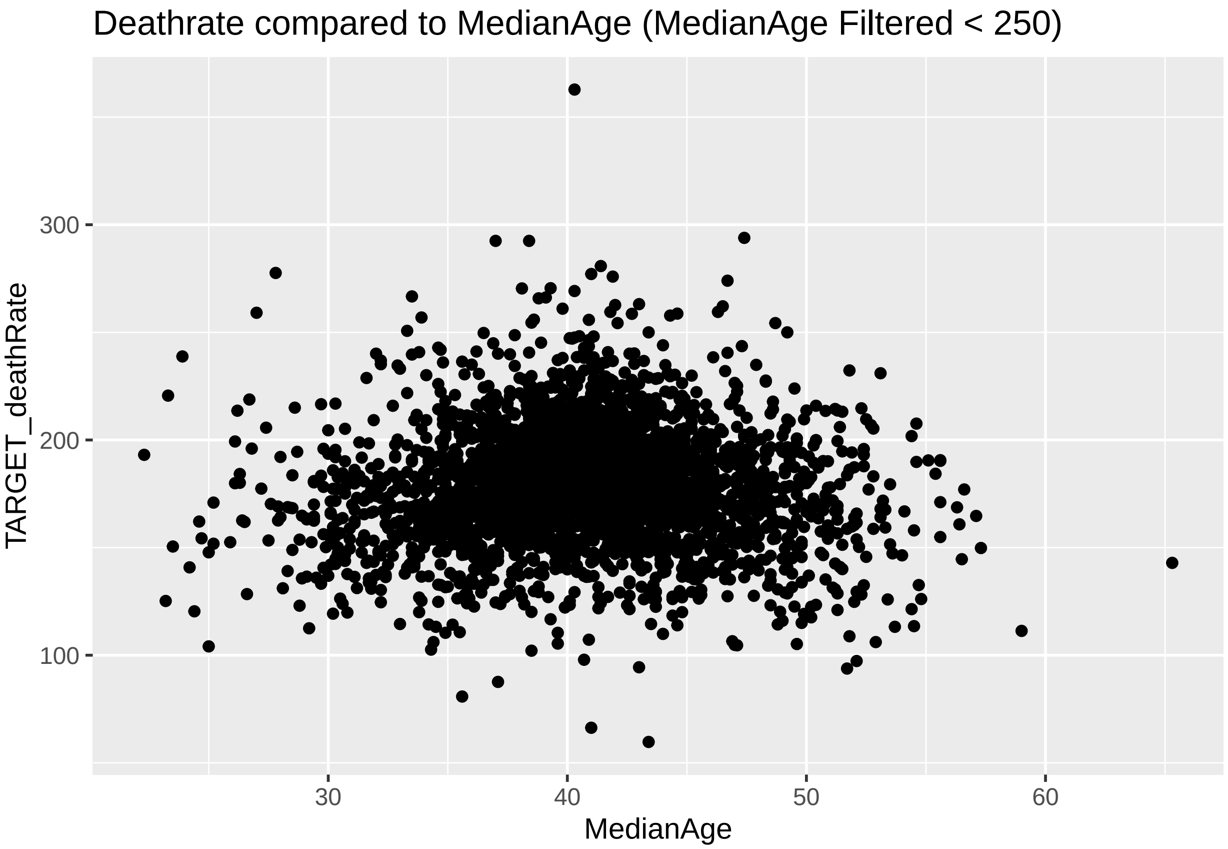


It is difficult to see any relationship between MedianAge and Deathrate with the values in Median Age, as there are 20 or more that seem to be incorrect (median ages of individuals 350 years+)

```
summary(df$MedianAge)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   22.30   37.70   41.00   45.27   44.00  624.00
```

```
#plotting MedianAge and Deathrate filtering out the values that appear to be incorrectly entered
df %>% filter(MedianAge < 250) %>%
  ggplot(aes(x=MedianAge, y=TARGET_deathRate)) + geom_point() +
  ggtitle("Deathrate compared to MedianAge (MedianAge Filtered < 250)")
```



With a better look at the data between MedianAge and Deathrate, there does not appear to be a relationship.

```
glimpse(df)
```

```
## Rows: 3,047
## Columns: 30
## $ avgAnnCount             <dbl> 1397, 173, 102, 427, 57, 428, 250, 146, 88, 40~
## $ avgDeathsPerYear        <dbl> 469, 70, 50, 202, 26, 152, 97, 71, 36, 1380, 3~
## $ TARGET_deathRate        <dbl> 164.9, 161.3, 174.7, 194.8, 144.4, 176.0, 175.~
## $ incidenceRate           <dbl> 489.8, 411.6, 349.7, 430.4, 350.1, 505.4, 461.~
## $ medIncome               <dbl> 61898, 48127, 49348, 44243, 49955, 52313, 3778~
## $ popEst2015              <dbl> 260131, 43269, 21026, 75882, 10321, 61023, 415~
## $ povertyPercent          <dbl> 11.2, 18.6, 14.6, 17.1, 12.5, 15.6, 23.2, 17.8~
## $ studyPerCap             <dbl> 499.74820, 23.11123, 47.56016, 342.63725, 0.00~
## $ binnedInc               <chr> "(61494.5, 125635]", "(48021.6, 51046.4]", "(4~
## $ MedianAge               <dbl> 39.3, 33.0, 45.0, 42.8, 48.3, 45.4, 42.6, 51.7~
## $ MedianAgeMale           <dbl> 36.9, 32.2, 44.0, 42.2, 47.8, 43.5, 42.2, 50.8~
## $ MedianAgeFemale         <dbl> 41.7, 33.7, 45.8, 43.4, 48.9, 48.0, 43.5, 52.5~
## $ Geography               <chr> "Kitsap County, Washington", "Kittitas County,~
## $ AvgHouseholdSize        <dbl> 2.5400, 2.3400, 2.6200, 2.5200, 2.3400, 2.5800~
## $ PercentMarried          <dbl> 52.5, 44.5, 54.2, 52.7, 57.8, 50.4, 54.1, 52.7~
## $ PctNoHS18_24            <dbl> 11.5, 6.1, 24.0, 20.2, 14.9, 29.9, 26.1, 27.3,~
## $ PctHS18_24              <dbl> 39.5, 22.4, 36.6, 41.2, 43.0, 35.1, 41.4, 33.9~
## $ PctSomeCol18_24         <dbl> 42.1, 64.0, NA, 36.1, 40.0, NA, NA, 36.5, NA, ~
## $ PctBachDeg18_24         <dbl> 6.9, 7.5, 9.5, 2.5, 2.0, 4.5, 5.8, 2.2, 1.4, 7~
## $ PctHS25_Over            <dbl> 23.2, 26.0, 29.0, 31.6, 33.4, 30.4, 29.8, 31.6~
## $ PctBachDeg25_Over       <dbl> 19.6, 22.7, 16.0, 9.3, 15.0, 11.9, 11.9, 11.3,~
## $ PctEmployed16_Over      <dbl> 51.9, 55.9, 45.9, 48.3, 48.2, 44.1, 51.8, 40.9~
## $ PctUnemployed16_Over    <dbl> 8.0, 7.8, 7.0, 12.1, 4.8, 12.9, 8.9, 8.9, 10.3~
## $ PctPrivateCoverage      <dbl> 75.1, 70.2, 63.7, 58.4, 61.6, 60.0, 49.5, 55.8~
## $ PctPrivateCoverageAlone <dbl> NA, 53.8, 43.5, 40.3, 43.9, 38.8, 35.0, 33.1, ~
## $ PctEmpPrivCoverage      <dbl> 41.6, 43.6, 34.9, 35.0, 35.1, 32.6, 28.3, 25.9~
## $ PctPublicCoverage       <dbl> 32.9, 31.1, 42.1, 45.3, 44.0, 43.2, 46.4, 50.9~
## $ PctPublicCoverageAlone  <dbl> 14.0, 15.3, 21.1, 25.0, 22.7, 20.2, 28.7, 24.1~
## $ PctMarriedHouseholds    <dbl> 52.85608, 45.37250, 54.44487, 51.02151, 54.027~
## $ BirthRate               <dbl> 6.1188310, 4.3330956, 3.7294878, 4.6038408, 6.~
```

Step 2

1 - Fit a linear model with y as the response and the variables chosen in Step One as the predictors.

Variables going in to model:

```
- avgAnnCount
- IncidenceRate
- MedianIncome
- MedianAge
- studyPerCap
- PctHS25_Over
- PctPrivateCoverage
- PctPublicCoverage
```

```r
model1 <- lm(TARGET_deathRate~avgAnnCount+incidenceRate+medIncome+MedianAge+
                      studyPerCap+PctHS25_Over+PctPrivateCoverage+PctPublicCoverage, data = df)

summary(model1)
```

model1 summary

```
##
## Call:
## lm(formula = TARGET_deathRate ~ avgAnnCount + incidenceRate +
##     medIncome + MedianAge + studyPerCap + PctHS25_Over + PctPrivateCoverage +
##     PctPublicCoverage, data = df)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -117.082  -11.968    0.441   11.655  140.421
##
## Coefficients:
##                      Estimate Std. Error t value Pr(>|t|)
## (Intercept)         1.122e+02  6.283e+00  17.855  < 2e-16 ***
## avgAnnCount        -8.341e-04  2.804e-04  -2.975 0.002957 **
## incidenceRate       2.346e-01  7.000e-03  33.515  < 2e-16 ***
## medIncome          -1.991e-04  5.483e-05  -3.630 0.000288 ***
## MedianAge          -7.115e-03  8.179e-03  -0.870 0.384405
## studyPerCap        -7.880e-05  7.070e-04  -0.111 0.911259
## PctHS25_Over        9.206e-01  6.427e-02  14.325  < 2e-16 ***
## PctPrivateCoverage -8.782e-01  5.767e-02 -15.227  < 2e-16 ***
## PctPublicCoverage  -1.106e-01  8.054e-02  -1.373 0.169935
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 20.38 on 3038 degrees of freedom
## Multiple R-squared:  0.4623, Adjusted R-squared:  0.4609
## F-statistic: 326.5 on 8 and 3038 DF,  p-value: < 2.2e-16
```

2 - Which predictors are statistically significant? What is the R squared value? Does it match your intuition?

- The following variables, using an alpha of .05, are statistically significant:
  - avgAnnCount
  - incidenceRate
  - medIncome
  - PctHS25_Over
  - PctPrivateCoverage
- The initial model has an R^2 value of .4623.
- 5/8 of the variables selected were statistically significant. The most statistically insignificant variable was studyPerCap. I assumed this might be the most significant as it could indicate counties were research was being done for high levels of cancer rates and help the model with its predictions.

I also assumed the R^2 value would be higher with the number of variables selected.

Step 3

1 - Perform model selection via automated selection on the model with all your chosen predictors (NOT all the predictors in the data set):

```
- Apply `fastbw()` to the data in R.

- Apply `stepAIC()` to the data in R.

- For each procedure, submit your comment on the variables that the procedure removed from or retained

    - Does it match your intuition?

    - How do the automatically selected models compare to your model from Step 2?

    - Which model will you choose to proceed with?
```

- Variables I anticipate the automated selection process will remove:
  - MedianAge
  - studyPerCap
  - PctPublicCoverage

fastbw()

```
#library for
library(rms)
```

```
## Warning: package 'rms' was built under R version 4.3.3
```

```
model_ols <- ols(TARGET_deathRate~avgAnnCount+incidenceRate+medIncome+MedianAge+
                      studyPerCap+PctHS25_Over+PctPrivateCoverage+PctPublicCoverage, data = df)
```

```r
fastbw(model_ols, rule='p', sls = 0.05)
```

```
##
##  Deleted           Chi-Sq d.f. P      Residual d.f. P      AIC   R2
##  studyPerCap       0.01   1    0.9113 0.01     1    0.9113 -1.99 0.462
##  MedianAge         0.75   1    0.3858 0.76     2    0.6823 -3.24 0.462
##  PctPublicCoverage 2.07   1    0.1507 2.83     3    0.4186 -3.17 0.462
##
## Approximate Estimates after Deleting Factors
##
##                          Coef      S.E.   Wald Z         P
## Intercept           1.056e+02 4.262e+00  24.785 0.0000000
## avgAnnCount        -8.499e-04 2.797e-04  -3.039 0.0023734
## incidenceRate       2.334e-01 6.939e-03  33.633 0.0000000
## medIncome          -1.703e-04 5.075e-05  -3.356 0.0007901
## PctHS25_Over        9.010e-01 6.254e-02  14.407 0.0000000
## PctPrivateCoverage -8.455e-01 5.185e-02 -16.307 0.0000000
##
## Factors in Final Model
##
## [1] avgAnnCount        incidenceRate      medIncome          PctHS25_Over
## [5] PctPrivateCoverage
```

fastbw() model summary Fitting model with suggested predictors from `fastbw()`

```r
model_fastbw_selection <- lm(TARGET_deathRate~avgAnnCount+incidenceRate+medIncome+
                     PctHS25_Over+PctPrivateCoverage,data=df)
summary(model_fastbw_selection)
```

```
##
## Call:
## lm(formula = TARGET_deathRate ~ avgAnnCount + incidenceRate +
##     medIncome + PctHS25_Over + PctPrivateCoverage, data = df)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -115.655  -12.173    0.442   11.849  140.284
##
## Coefficients:
##                      Estimate Std. Error t value Pr(>|t|)
## (Intercept)         1.056e+02  4.262e+00  24.786  < 2e-16 ***
## avgAnnCount        -8.499e-04  2.796e-04  -3.039  0.00239 **
## incidenceRate       2.334e-01  6.939e-03  33.634  < 2e-16 ***
## medIncome          -1.703e-04  5.075e-05  -3.356  0.00080 ***
## PctHS25_Over        9.010e-01  6.254e-02  14.407  < 2e-16 ***
## PctPrivateCoverage -8.455e-01  5.185e-02 -16.308  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 20.38 on 3041 degrees of freedom
## Multiple R-squared:  0.4618, Adjusted R-squared:  0.4609
## F-statistic: 521.8 on 5 and 3041 DF,  p-value: < 2.2e-16
```

The `fastbw()` model selection performed how I anticipated. It removed all of the predictors that were shown to be statistically insignificant in the original model.

- MedianAge
- studyPerCap
- PctPublicCoverage

This is because I set the selection criteria to take everything out above an α of .05.

The `fastbw()` model has an R^2 value of just .0005 smaller than the original model, with 3 less variables. My originally selected model and the `fastbw()` model have identical R_a^2 values

AIC selection

```r
library(MASS)

model_aic <- stepAIC(model1)
```

```
## Start:  AIC=18378.78
## TARGET_deathRate ~ avgAnnCount + incidenceRate + medIncome +
##     MedianAge + studyPerCap + PctHS25_Over + PctPrivateCoverage +
##     PctPublicCoverage
##
##                      Df Sum of Sq     RSS   AIC
## - studyPerCap         1         5 1261448 18377
## - MedianAge           1       314 1261757 18378
## - PctPublicCoverage   1       782 1262225 18379
## <none>                            1261443 18379
## - avgAnnCount         1      3674 1265117 18386
## - medIncome           1      5472 1266915 18390
## - PctHS25_Over        1     85203 1346646 18576
## - PctPrivateCoverage  1     96280 1357723 18601
## - incidenceRate       1    466413 1727856 19335
##
## Step:  AIC=18376.79
## TARGET_deathRate ~ avgAnnCount + incidenceRate + medIncome +
##     MedianAge + PctHS25_Over + PctPrivateCoverage + PctPublicCoverage
##
##                      Df Sum of Sq     RSS   AIC
## - MedianAge           1       312 1261760 18376
## - PctPublicCoverage   1       785 1262233 18377
## <none>                            1261448 18377
## - avgAnnCount         1      3700 1265148 18384
## - medIncome           1      5473 1266921 18388
## - PctHS25_Over        1     85978 1347426 18576
## - PctPrivateCoverage  1     97325 1358773 18601
## - incidenceRate       1    468296 1729744 19337
##
## Step:  AIC=18375.54
## TARGET_deathRate ~ avgAnnCount + incidenceRate + medIncome +
##     PctHS25_Over + PctPrivateCoverage + PctPublicCoverage
##
##                      Df Sum of Sq     RSS   AIC
## <none>                            1261760 18376
## - PctPublicCoverage   1       857 1262618 18376
## - avgAnnCount         1      3663 1265423 18382
## - medIncome           1      5532 1267292 18387
## - PctHS25_Over        1     85879 1347639 18574
## - PctPrivateCoverage  1     97913 1359673 18601
## - incidenceRate       1    468174 1729934 19335
```

```r
model_aic$anova
```

```
## Stepwise Model Path 
## Analysis of Deviance Table
## 
## Initial Model:
## TARGET_deathRate ~ avgAnnCount + incidenceRate + medIncome + 
##     MedianAge + studyPerCap + PctHS25_Over + PctPrivateCoverage + 
##     PctPublicCoverage
## 
## Final Model:
## TARGET_deathRate ~ avgAnnCount + incidenceRate + medIncome + 
##     PctHS25_Over + PctPrivateCoverage + PctPublicCoverage
## 
## 
##            Step Df   Deviance Resid. Df Resid. Dev      AIC
## 1                                  3038    1261443 18378.78
## 2 - studyPerCap  1   5.158395      3039    1261448 18376.79
## 3   - MedianAge  1 312.355207      3040    1261760 18375.54
```

```r
model_aic_selection <- lm(TARGET_deathRate~avgAnnCount+incidenceRate+medIncome+
                            PctHS25_Over+PctPrivateCoverage+PctPublicCoverage, data=df)
summary(model_aic_selection)
```

```
## 
## Call:
## lm(formula = TARGET_deathRate ~ avgAnnCount + incidenceRate + 
##     medIncome + PctHS25_Over + PctPrivateCoverage + PctPublicCoverage, 
##     data = df)
## 
## Residuals:
##      Min       1Q   Median       3Q      Max 
## -116.893  -12.011    0.467   11.702  140.486 
## 
## Coefficients:
##                      Estimate Std. Error t value Pr(>|t|)    
## (Intercept)         1.123e+02  6.280e+00  17.876  < 2e-16 ***
## avgAnnCount        -8.315e-04  2.799e-04  -2.971 0.002995 ** 
## incidenceRate       2.345e-01  6.983e-03  33.586  < 2e-16 ***
## medIncome          -1.997e-04  5.470e-05  -3.651 0.000266 ***
## PctHS25_Over        9.207e-01  6.400e-02  14.384  < 2e-16 ***
## PctPrivateCoverage -8.807e-01  5.734e-02 -15.359  < 2e-16 ***
## PctPublicCoverage  -1.155e-01  8.033e-02  -1.437 0.150727    
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
## 
## Residual standard error: 20.37 on 3040 degrees of freedom
## Multiple R-squared:  0.4621, Adjusted R-squared:  0.4611 
## F-statistic: 435.3 on 6 and 3040 DF,  p-value: < 2.2e-16
```

The `stepAIC()` selection removed two of the three variables I thought would be removed. Median Age and

Study per cap were removed both had extremely high p-values. The `stepAIC()` process did not remove the Pct Public Coverage variable. It had a p-value of .16 in the original model, but with the other two removed it droped to .15, still 3x higher than the selected α value. I assumed Everything that was statisically insignificant would also lead to a large enough AIC reduction when removed from the model.

The `stepAIC()` model has a .0002 R^2 value compared to the original model, but it has a higher R_a^2 of .0002.

Moving forward I don't believe leaving Pct Public Coverage variable in the model adds much. I will use the `fastbw()` selected model going forward, which has the following variables:

- avgAnnCount
- incidenceRate
- medIncome
- PctHS25_Over
- PctPrivateCoverage

Step 4

1 - Perform diagnostics on the model chosen in Step Three to check the mathematical assumptions of the model.

```
library(lmtest)

#running the bptest() to look at the constant variance
bptest(model_fastbw_selection)
```

```
##
##  studentized Breusch-Pagan test
##
## data:  model_fastbw_selection
## BP = 70.308, df = 5, p-value = 8.841e-14
```

Looking just at the bptest() we would conclude that the model assumptions are not met and we can reject the null and state there is not constant variance.

```
#running the shapiro.test() to look at if there is normality within the residuals
shapiro.test(model_fastbw_selection$residuals)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  model_fastbw_selection$residuals
## W = 0.98365, p-value < 2.2e-16
```

The shapiro.test() p-vale would indicate that we have evidence to reject the null and state there is not normality in the residuals

```
#running the dwtest() to look at if there is independence within the residuals
dwtest(model_fastbw_selection)
```

```
##
##  Durbin-Watson test
##
## data:  model_fastbw_selection
## DW = 1.6812, p-value < 2.2e-16
## alternative hypothesis: true autocorrelation is greater than 0
```

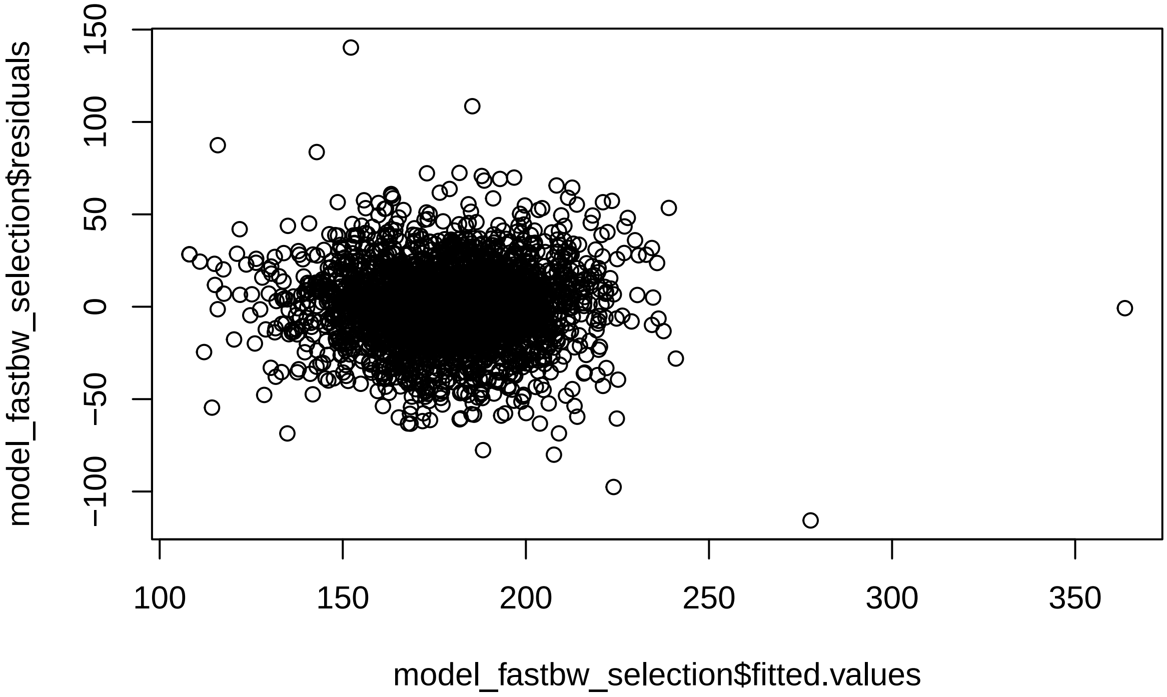
The dwtest() has given us a p-value to say we can reject the null and state there is not independence within the residuals.

With all of these statistical test we might assume that the assumptions are not upheld.

2 - Produce the following three (3) plots:

- Fitted values vs. residuals plot,

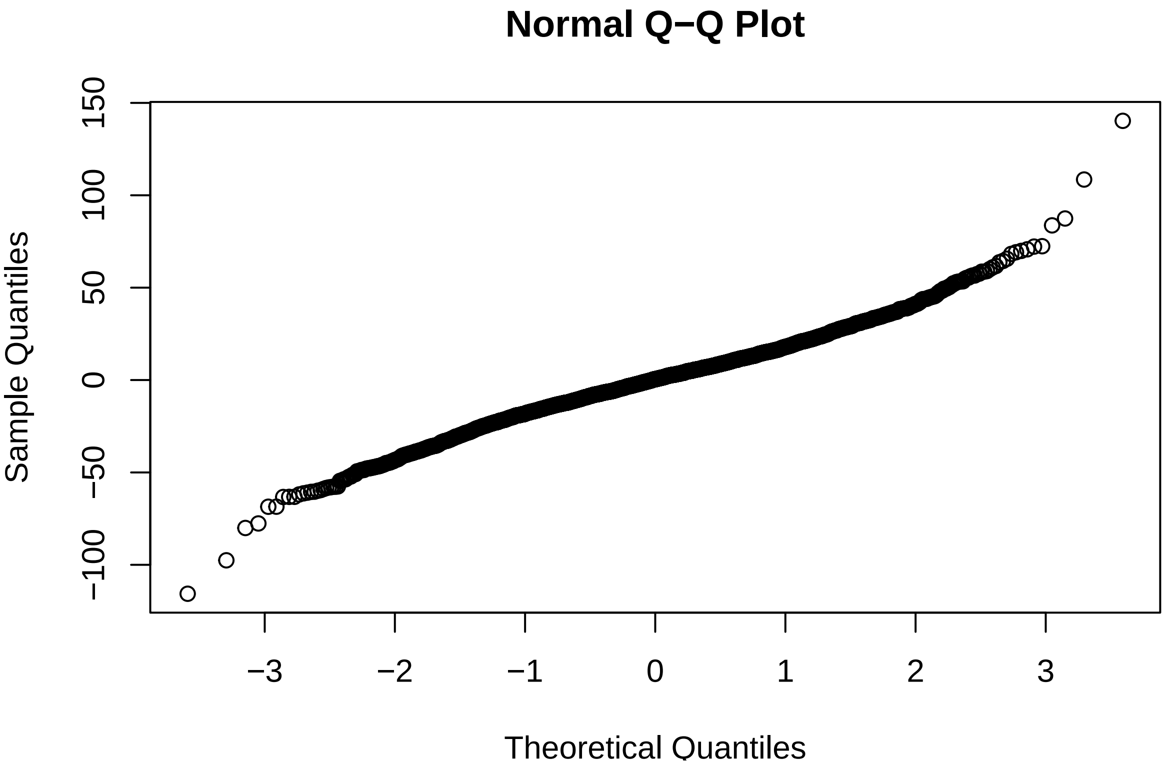
```
plot(model_fastbw_selection$fitted.values, model_fastbw_selection$residuals)
```



```
- The Fitted values vs. residuals plot looks to be circular and no real trends appearing in the plot, tl
```

- Q-Q plot

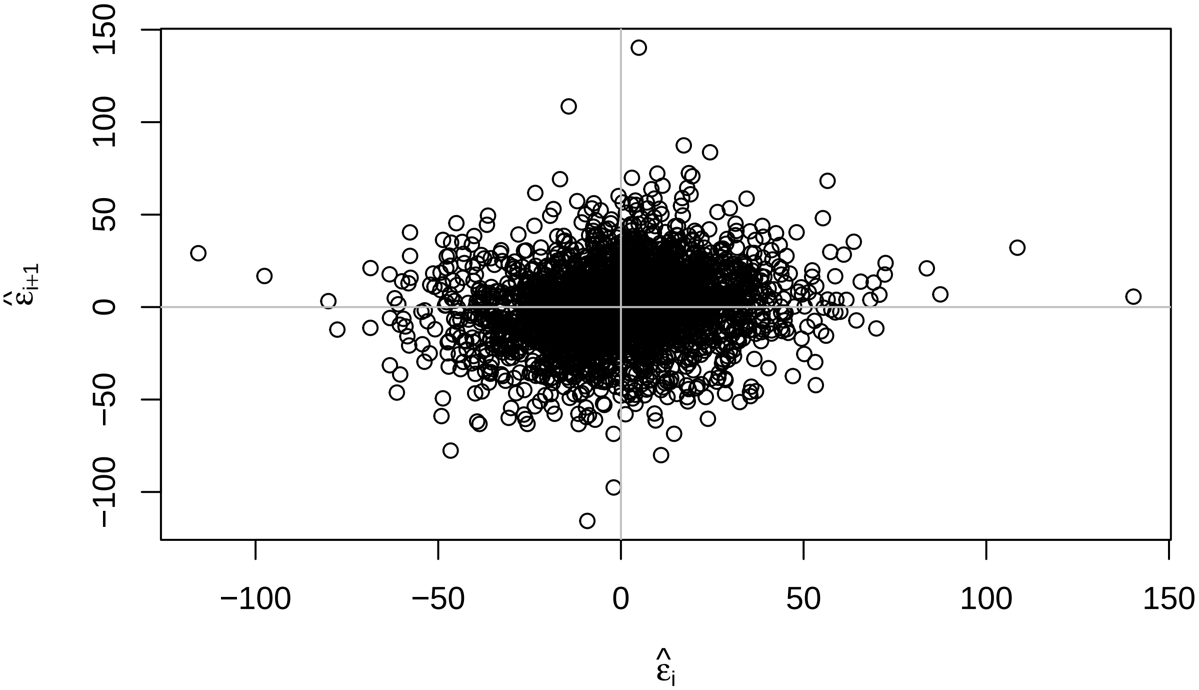
```
qqnorm(model_fastbw_selection$residuals)
```



```
- The Q-Q plot does not have any gross derivations from normality within the residuals and the model as
```

- Lagged residual plot (i.e., right hand plot in Figure 6.7 in the text).

```
n <- length(residuals(model_fastbw_selection))
plot(tail(residuals(model_fastbw_selection),n-1) ~ head(residuals(model_fastbw_selection),n-1),
     xlab=expression(hat(epsilon)[i]),ylab=expression(hat(epsilon)[i+1]))
abline(h=0,v=0,col=grey(0.75))
```



```
- Lagged residual plot does not have any positive or negative trends, indicating there is not be any se
```

3 - For each, provide one (1) sentence to explain whether they indicate if the model assumptions are upheld. If the assumptions do not appear to be upheld, then hang tight, as this is addressed in a later step.

Step 5

1 - Check for outliers and influential observations.

2 - Write up brief explanations as to what you are seeing.

- Your code used to calculate standardized residuals and Cook's Distance;
- Your code used to identify outliers and influential observations (don't print/paste all individual values); and
- Your brief explanations as to what you are seeing.

```r
#creating dataframe with standardized residuals
standard_resid <- data.frame(round(rstandard(model_fastbw_selection),4))

#looking at first 5
head(standard_resid)
```

```
##   round.rstandard.model_fastbw_selection...4.
## 1                                     -0.0352
## 2                                      0.1907
## 3                                      1.1630
## 4                                      0.8604
## 5                                     -0.6085
## 6                                     -0.7351
```

```r
#looking to see if there are values that would be considered outliers (greater than 3)
which(standard_resid$round.rstandard.model_fastbw_selection...4. > 3)
```

```
##  [1]  116  122  254  522  775 1221 1366 1497 2176 2549 2600 2637 2714 2727
```

```r
#seeing what those values are
standard_resid[which(standard_resid$round.rstandard.model_fastbw_selection...4. > 3),]
```

```
##  [1] 3.3981 3.5565 4.1182 3.1296 3.5457 6.8959 5.3300 3.4729 3.1660 3.4358
## [11] 3.2229 3.0294 4.3117 3.3617
```

There are 14 total points that would be considered outliers, or .46% of the total dataframe. Some of these values have sizable standardized residual values. But we should look to see with cooks distance if any of them have leverage and affect the model.

cooks distance

```r
#finding n and p
n <- dim(model.matrix(model_fastbw_selection))[1]
p <- dim(model.matrix(model_fastbw_selection))[2]
#creating the fraction
num_df <- p
den_df <- n-p
#creating the 50th percentile threshold
F_thresh <- qf(0.5,num_df,den_df)
F_thresh
```

```
## [1] 0.891551
```

```r
#calculaing the cooks distance values for the residuals
cooks_distance_values <- cooks.distance(model_fastbw_selection)

#seeing which value is the largest
which.max(cooks_distance_values)
```

```
## 282
## 282
```

```r
#looking at what the largest index's value is
cooks_distance_values[282]
```

```
##       282
## 0.2661295
```

A rule of thumb of leverage would be if the cooks distance is over 1. The largest value of the distances is no where near 1.

```r
#double checking to make sure the above statement is true
which(cooks.distance(model_fastbw_selection)>1)
```

```
## named integer(0)
```

```r
#checking to see if there are any values above the f_threshold
which(cooks.distance(model_fastbw_selection)>F_thresh)
```

```
## named integer(0)
```

Since there are no values above the 50th percentile threshold of the F Distribution, we can say that while there are some values that could be considered outliers, there are none that appear to have any influence/leverage and are affecting the model.

Step 6

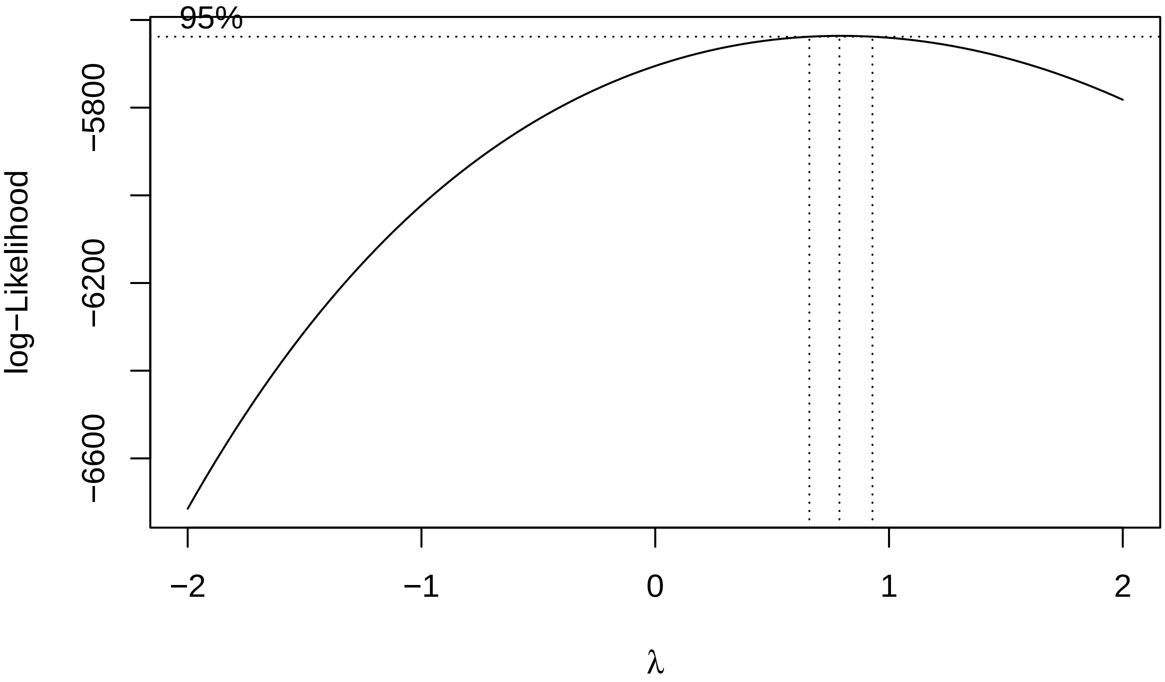
Correct the model if mathematical assumptions of the model were not met in Step Four.

- If a transformation is needed, indicate what measures you are taking (i.e., Box-Cox Transformation, polynomial regression, or some method that wasn't covered if you're feeling adventurous).

- If a transformation is not needed, run the Box-Cox method anyway to see if the result agrees with your intuitive assessment from Step Four.

```r
#library import for boxcox
require(MASS)

#applying boxcox to the fastbw_model_selection
bc_fastbw_results <- boxcox(model_fastbw_selection, data=df, plotit = T)
```



The max vlue looks to be close to .75, but not close enough to 1 to say there would definitively not be any benifit from a trasnformation.

```
#finding the max value of lambda
lambda <- bc_fastbw_results$x[which.max(bc_fastbw_results$y)]
lambda
```

```
## [1] 0.7878788
```

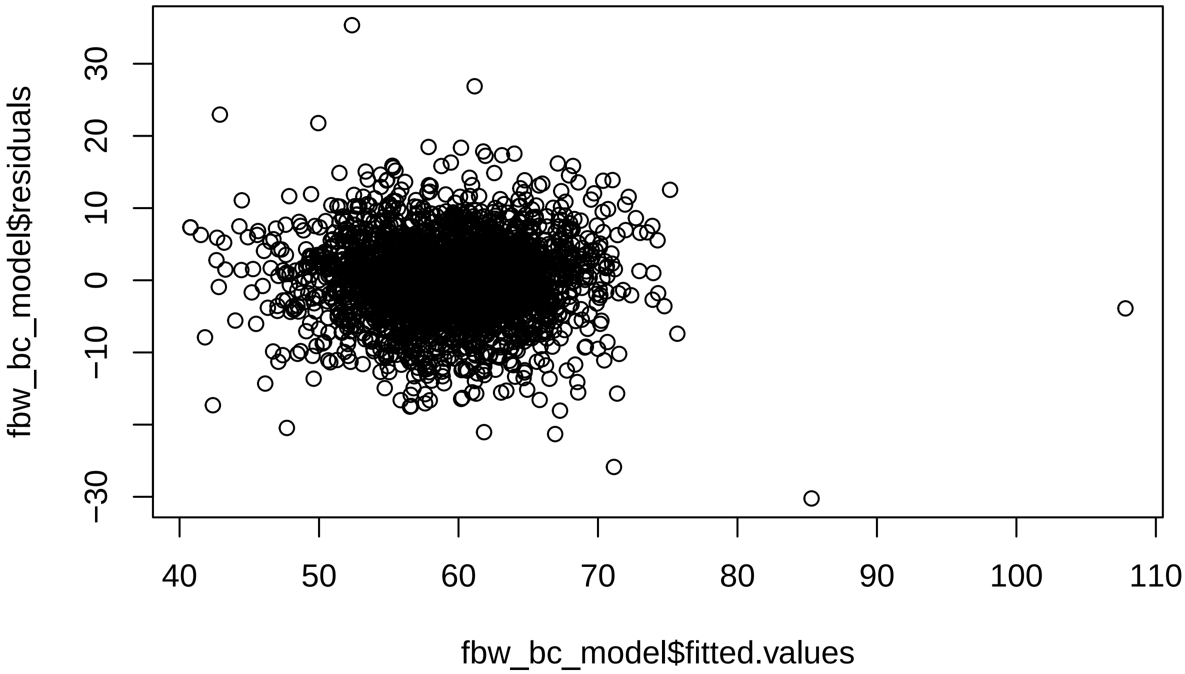
Boxcox method has suggested a value of .7879 for λ

```
#fitting a new model with the value of lambda transforming the response variable
fbw_bc_model <- lm((TARGET_deathRate)^lambda~avgAnnCount+incidenceRate+medIncome+
                     PctHS25_Over+PctPrivateCoverage,data=df)

summary(fbw_bc_model)
```

```
##
## Call:
## lm(formula = (TARGET_deathRate)^lambda ~ avgAnnCount + incidenceRate +
##     medIncome + PctHS25_Over + PctPrivateCoverage, data = df)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -30.226  -3.122   0.197   3.160  35.341
##
## Coefficients:
##                      Estimate Std. Error t value Pr(>|t|)
## (Intercept)         3.985e+01  1.119e+00  35.615  < 2e-16 ***
## avgAnnCount        -2.103e-04  7.343e-05  -2.863 0.004221 **
## incidenceRate       6.117e-02  1.822e-03  33.573  < 2e-16 ***
## medIncome          -4.568e-05  1.333e-05  -3.428 0.000616 ***
## PctHS25_Over        2.404e-01  1.642e-02  14.641  < 2e-16 ***
## PctPrivateCoverage -2.177e-01  1.361e-02 -15.994  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 5.351 on 3041 degrees of freedom
## Multiple R-squared:  0.4607, Adjusted R-squared:  0.4599
## F-statistic: 519.7 on 5 and 3041 DF,  p-value: < 2.2e-16
```

```
#plotting the diagnostic plot
plot(fbw_bc_model$fitted.values, fbw_bc_model$residuals)
```



There does not seem to by any change in the diagnostic plot from the original `fastbw_model_selection` and the transformed `fbw_bc` model. This appears to prove the model did already met the mathematical assumptions of a linear model.

Step 7

Report your final model and use it to perform inferences.

- First, report the parameter estimates and p-values for your final model in a table similar to the one below:

```
#showing the model summary for the fast_bw selected model
summary(model_fastbw_selection)
```

```
##
## Call:
## lm(formula = TARGET_deathRate ~ avgAnnCount + incidenceRate +
##     medIncome + PctHS25_Over + PctPrivateCoverage, data = df)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -115.655  -12.173    0.442   11.849  140.284
##
## Coefficients:
```

```
##                        Estimate Std. Error t value Pr(>|t|)    
## (Intercept)           1.056e+02  4.262e+00  24.786  < 2e-16 ***
## avgAnnCount          -8.499e-04  2.796e-04  -3.039  0.00239 ** 
## incidenceRate         2.334e-01  6.939e-03  33.634  < 2e-16 ***
## medIncome            -1.703e-04  5.075e-05  -3.356  0.00080 ***
## PctHS25_Over          9.010e-01  6.254e-02  14.407  < 2e-16 ***
## PctPrivateCoverage -8.455e-01  5.185e-02 -16.308  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
## 
## Residual standard error: 20.38 on 3041 degrees of freedom
## Multiple R-squared:  0.4618, Adjusted R-squared:  0.4609 
## F-statistic: 521.8 on 5 and 3041 DF,  p-value: < 2.2e-16
```

```r
#loading library that shows the desired table in the Rstudio IDE viewer window
library(sjPlot)
```

```
## Warning: package 'sjPlot' was built under R version 4.3.3
```

```
## #refugeeswelcome
```

```r
#showcasing the models parameters estimates and the p-values for each estimate
tab_model(model_fastbw_selection, show.ci = F)
```

TARGET_deathRate

Predictors

Estimates

p

(Intercept)

105.63

<0.001

avgAnnCount

-0.00

0.002

incidenceRate

0.23

<0.001

medIncome

-0.00

0.001

PctHS25 Over

0.90

<0.001

PctPrivateCoverage

-0.85

<0.001

Observations

3047

R2 / R2 adjusted

0.462 / 0.461

- Report the R^2 for the model.

```
#selecting the fast_bw's model R^2 value
summary(model_fastbw_selection)$r.squared
```

```
## [1] 0.461769
```

- Compute and report a 95% confidence interval for the slope of whichever predictor you feel is most important.

```
#computing a 95% confidence interval for the PctHS25Over variable
0.9010 * c(-1,1) + qt(.975, 3041) * 0.06254
```

```
## [1] -0.778375  1.023625
```

We are 95% confident that the slope for `PctHS25Over` is between -.778375 and 1.023625.

- Compute and report a 95% confidence interval for a prediction. In other words, choose particular values of your predictors that are meaningful (say, perhaps the median of each) and compute a 95% confidence interval for the predicted value of y at those values.

```
#finding the median values for all predictor variables in the model
median(df$avgAnnCount)
```

```
## [1] 171
```

```
median(df$incidenceRate)
```

```
## [1] 453.5494
```

```
median(df$medIncome)
```

```
## [1] 45207
```

```
median(df$PctHS25_Over)
```

```
## [1] 35.3
```

```
median(df$PctPrivateCoverage)
```

```
## [1] 65.1
```

```
#taking a 95% confidence interval for the median values of all predictor variables
predict(model_fastbw_selection, newdata = data.frame(avgAnnCount = median(df$avgAnnCount),
                                                     incidenceRate = median(df$incidenceRate),
                                                     medIncome = median(df$medIncome),
                                                     PctHS25_Over = median(df$PctHS25_Over),
                                                     PctPrivateCoverage = median(df$PctPrivateCoverage)
        interval = "confidence")
```

```
##        fit      lwr      upr
## 1 180.3991 179.6135 181.1846
```

We are 95% confident that a county's target_Deathrate who has an median avgAnnCount of 171, a median incidenceRate of 453.55, a median medIncome of 45207, a median PctHS25_Over of 35.3, and a median PctPrivateCoverage of 65.1 will lie in the range between 179.61 and 181.1846.

- Compute and report a 95% prediction interval for a particular observation. Again, you'll choose particular values of your predictors and compute the prediction interval for those values.

```
#selecting random row from dataframe
set.seed(1842)
df %>% dplyr::select(TARGET_deathRate, avgAnnCount, incidenceRate, medIncome, PctHS25_Over, PctPrivateC
```

```
## # A tibble: 1 x 6
##   TARGET_deathRate avgAnnCount incidenceRate medIncome PctHS25_Over
##              <dbl>       <dbl>         <dbl>     <dbl>        <dbl>
## 1             204.         155          467.     39303         39.8
## # i 1 more variable: PctPrivateCoverage <dbl>
```

```
#95 prediction interval for a specific observation in the dataset
predict(model_fastbw_selection, newdata = data.frame(avgAnnCount = 155,
                                                     incidenceRate = 467.1,
                                                     medIncome = 39303,
                                                     PctHS25_Over = 39.8,
                                                     PctPrivateCoverage = 59.8),
        interval = "prediction", level = 0.95)
```

```
##        fit      lwr      upr
## 1 193.1166 153.1524 233.0809
```

There is a 95% probability that the target_deathrate of a county with avgAnnCount of 155, incidenceRate of 467.1, medIncome of 39303, PctHS25_Over of 39.8, and PctPrivateCoverage of 59.8 will lie in the range between 153.15 and 233.08.