# WHO IS READY TO RETIRE: YOUR AVERAGE LIFE EXPECTANCY AND THE SAVINGS NEEDED TO SUPPORT YOU

by

Kaitlyn E. Burkett

Honors Thesis

Appalachian State University

Submitted to the Department of Mathematical Sciences
in partial fulfillment of the requirements for the degree of

Bachelor of Science

May, 2019

Approved by:

_____

Lasanthi Watagoda, Ph.D., Thesis Director

_____

Hasthika Rupasinghe, Ph.D., Second Reader

_____

William J. Cook, Ph.D., Honors Director

_____

Eric S. Marland, Ph.D., Chair

# Abstract

The following work helps predict the life expectancy of an individual and how much they will need to save each month in order to live to that age after retirement. People often forget about life after retirement until it is too late to start saving properly. This project is hoping to bring this idea to people's attention earlier and to make the process easier. The work presented first finds a model to predict that life expectancy and then presents a calculator to give the minimum monthly contribution to savings needed to live comfortably in retirement assuming death at the expected age. Statistical calculations and this document were prepared in an R Studio sweave document [11].

# Acknowledgements

I would like to thank my family for their support and the help to bring me to college to give me the opportunity to write this thesis. I would also like to thank the Department of Mathematical Sciences at Appalachian State for teaching me the concepts in this thesis. Finally, I would like to thank Dr. Lasanthi Watagoda for all of her wonderful support as my Thesis Director.

# Contents

# 1 PROJECT AND DATA DESCRIPTION

Retirement seems like it will never come. Individuals tend to keep it at the back of their minds and forget to work in savings for retirement into their monthly budgets. Many factors contribute to a lack in savings. These factors include means to save being too low and the lack of knowledge on how and what to save for the future. The project's goal is to educate individuals on how to save for retirement so that they may live comfortably during that time. First, the project will be looking at different models to find the most appropriate fit in order to find the average life expectancy. The models will include the variables of sex, race, and the birth year. After finding the appropriate model to find the average life expectancy, a calculator will then be created that will find the amount of money that needs to be saved each month in order to live comfortably in retirement assuming death occurs at the expected age. The calculator will be including other saving methods that will help in the future, such as 401(k) and social security (if still in use). Expenses for the future include the current cost of living brought forward to the years of retirement.

The data set being used to predict the average life expectancy was produced by the NCHS (National Center for Health Statistics) [5]. The dataset was obtained from the Department of Health and Human Services. It was published by the Center for Disease Control and Prevention in collaboration with the National Center for Health Statistics. The data itself contained enough variables for the needs of this thesis. The data set shows the U.S. mortality trends since 1900 until 2015. The factors in the data set are year, sex, race, average life expectancy, and age-adjusted death rate, or death rate for short for further reference. The death rate is referring to deaths per 100,000 people. Due to the changes in categories of race used in publications, data for the black population of the dataset are not consistent before 1968, and non-existent before 1960. The data in the table is mostly predicted based on very few records of data. In the data set categories, Year records the years since early 1900 until 2015 for each combination of race and sex. Race records the data of All Races, Black, or White. Sex records the data of Both Sexes, Female, or Male. Average Life Expectancy, referred to as ALE in the code, records

the average life expectancy of the person born in the corresponding year, race, and sex. Age-adjusted Death Rate, referred to DR in the code, records the death rate in the corresponding year with the related race and sex. [5] In this dataset, the focus will be on producing an ALE based off of the year the user was born, the user's sex, and the user's race.

# 2 THEORY

Multiple Linear Regression

Simple linear regression is useful for creating a model using only one predictor variable. Multiple linear regression, however, is useful when there are more than one predictor variable. This gives an easier way to create a model with all predictor variables instead of having to create individual models for each predictor. This is done by giving a separate slope coefficient in a single model. [6]

Interaction Models

Interaction models are similar to Multiple Linear Regression, except there is an interaction factor in the model. It is best to look at a graph of the dataset to see how the factors interact. For example, if Race and Sex seem to be crossing in the graphed data, it would be good to add a factor that has those two factors interact. [6]

Generalized Linear Model

Generalized linear models (GLM) are a flexible generalization of ordinary linear regression that allows for response variables that have error distribution models other than a normal distribution. It allows the linear model to be related to the response variable is a link function and by allowing the magnitude of the variance of each measurement to be a function of its predicted value. [6]

Regression Tree

A regression tree is built through a process known as binary recursive partitioning, which is an iterative process that splits the data into partitions or branches, and then continues splitting

each partition into smaller groups as the method moves up each branch. [3]

Initially, all records in the Training Set (pre-classified records that are used to determine the structure of the tree) are grouped into the same partition. The algorithm then begins allocating the data into the first two partitions or branches, using every possible binary split on every field. The algorithm selects the split that minimizes the sum of the squared deviations from the mean in the two separate partitions. This splitting rule is then applied to each of the new branches. This process continues until each node reaches a user-specified minimum node size and becomes a terminal node. (If the sum of squared deviations from the mean in a node is zero, then that node is considered a terminal node even if it has not reached the minimum size.) [3]

Akaike Information Criterion

Akaike Information Criterion (AIC) is an estimator of the relative quality of statistical models for a given set of data. AIC estimates the quality of each model, relative to each of the other models. It provides a means for model selection. The number given is the estimate of the relative amount of information lost by a given model; the less information a model loses, the higher the quality of the model. In other words, the lower the AIC, the better the quality of the model. AIC also deals with the risks of over fitting and under fitting. [6]

Bayesian Information Criterion

Bayesian Information Criterion (BIC) is closely related to AIC. It is a criterion for model selection among a finite set of models. Adding parameters to a model can increase the likelihood of fit, but can result in over fitting. BIC resolves this problem by introducing a penalty term for the number of parameters in the model, the penalty term being larger in BIC than in AIC. [6]

Mallows $C_p$

Mallows $C_p$ ($C_p$) is used to help choose between multiple regression models. It compares the precision and bias of the full model to models with a subset of the predictors. $C_p$ is meant to be small and close to the number of predictors in the model. A small value indicates that the

model is relatively precise - has a small variance - in estimating the true regression coefficients and predicting future responses. A $C_p$ that is close to the number of predictors plus the constant indicates that the model is relatively unbiased in estimating the true regression coefficients and predicting future responses. Models with a lack of fit and bias have values of $C_p$ larger than p. [6]

Cross Validation

Cross Validation is a technique which involves reserving a particular sample of a dataset that is not trained for the model. Later, the model is tested on the sample before finalizing it. Reserving a sample data set first, the rest of the data set is used to train the model. The reserved data is then used to test, or validate, the model. If this produces a positive result, then the model is a quality one. [6]

# 3 CLEANING AND EXPLORING THE DATA

This document uses rpart by Therneau and Atkinson (2019) [3], ggplot2 by Wickham, Chang, et al. (2019) [7], plotly by Sievert et al. (2019) [4], dplyr by Wickham, Franois, et al. (2019) [8], knitr by Xie (2019) [12], and boot by Canty and Ripley (2019) [1].

## 3.1 Cleaning the data set

The first step in creating the model is to attach the dataset. The dataset is renamed to LE for Life Expectancy. In order to remove the blank data entries in the dataset, na.omit() was used. However, the only blank sets of data were in the year of 2015, meaning the data set was shortened by one year.

The following code is loading the data set, assigning the column names, and omitting the non-available data.

```
LE <-

  read_csv("NCHSALEatBirth.csv")

#View(NCHS_Death_rates_and_life_expectancy_at_birth_2_)

cn <- c("Year", "Race", "Sex", "ALE", "DR")

colnames(LE) <- cn

LE = na.omit(LE)
```

## 3.2 Exploratory data analysis

To get the summary information, the use of the summary() function was implemented.

```
summary(LE)

      Year          Race                Sex                  ALE
 Min.   :1900   Length:1035        Length:1035        Min.   :29.10
 1st Qu.:1928   Class :character   Class :character   1st Qu.:56.60
 Median :1957   Mode  :character   Mode  :character   Median :66.60
 Mean   :1957                                         Mean   :64.12
 3rd Qu.:1986                                         3rd Qu.:73.60
 Max.   :2014                                         Max.   :81.40
       DR
 Min.   : 616.7
 1st Qu.:1052.1
 Median :1547.3
 Mean   :1621.3
 3rd Qu.:2078.6
 Max.   :3845.7
```

The summary above produces, for all quantitative variables, the minimum, maximum, median, mean, first quartile, and third quartile. For the categorical variables, length, class, and mode are produced.

Year is a quantitative variable. There is no missing data points in the data set. The starting year is 1900 and the ending year is 2014.

Race and Gender are the two categorical variables in the dataset.

ALE is a quantitative variable. It has a mean of 64.12, meaning the average age for all groups is about 64 years. ALE has a standard deviation of 11.79. The shortest ALE has is 29.1, first quartile value 56.6, second quartile value of 66.6, third quartile value of 73.6, and the longest ALE is 81.4.

DR is a quantitative random variable. It has a mean of DR is 1612.30 (per 100,000 people). DR has a standard deviation of 676.39. The minimum DR recorded is 616.70, first quartile value of 1052.1, second quartile value of 1547.30, third quartile value of 2078.55, and the highest DR recorded in the dataset is 3845.7.

The following code is used to create a scatter plot of ALE per year by Sex and Race.

```
library(ggplot2)

p<-(ggplot(data =LE, aes(x= Year, y= ALE, col = Sex,

                         shape = Race))

+geom_point()+theme_bw())


print(p)
```

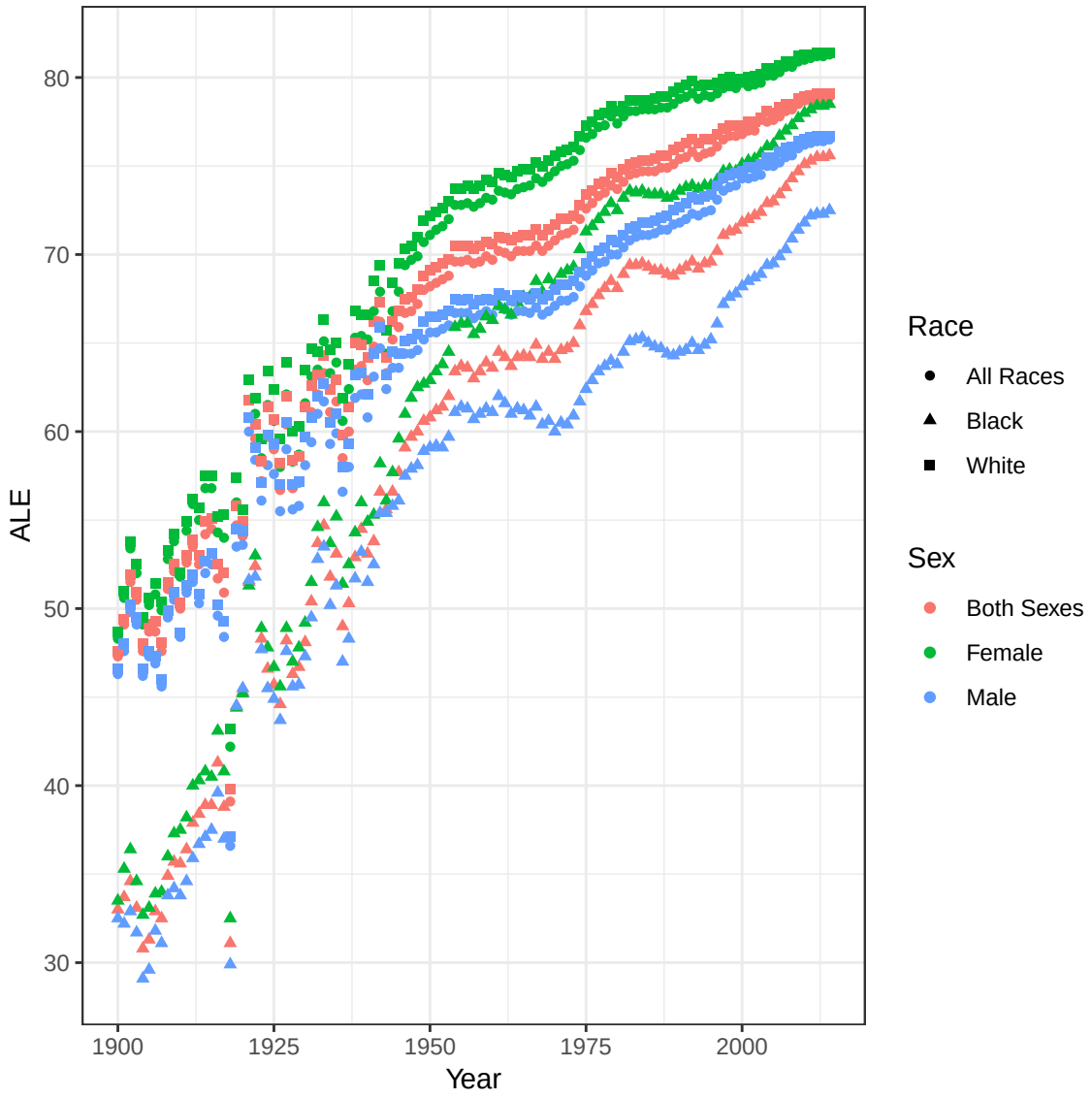



Figure 1: Average Life Expectancy per Year by Race and Sex

The $x$ axis of Figure 1 represents the year from 1900 to 2014. The $y$ axis of Figure 1 represents the average life expectancy. The data points are categorized into individual sex and race variables. For race, all races is represented by circles, black is represented by triangles, and white is represented by squares. For sex, both sexes is represented by red, female is represented by green, and male is represented by blue.

For all variables, there is a steady increase in ALE over the years. There is a noticeable

dip for those who were black and male in 1960s and 1980s. Black men have the lowest life expectancy compared to the other groups, and it is noticeable that white women have a greater life expectancy than any other group.

It is obvious that in the years before 1950 in Figure 1 that there is no cohesion and plenty of randomness in the variables, especially for black individuals. This is most likely due to a lack of proper data inputs for black men and women. Even though the data points before 1950 are less random for whites than for blacks, there is a noticeable difference between that data points before and after 1950 in Figure 1.

The following code is used to give side by side box plots of ALE by Sex and Race.

```
(ggplot(data =LE, aes(y= ALE, x = Race, fill = Race))

 +geom_boxplot() +facet_wrap(~Sex) + theme_bw())
```

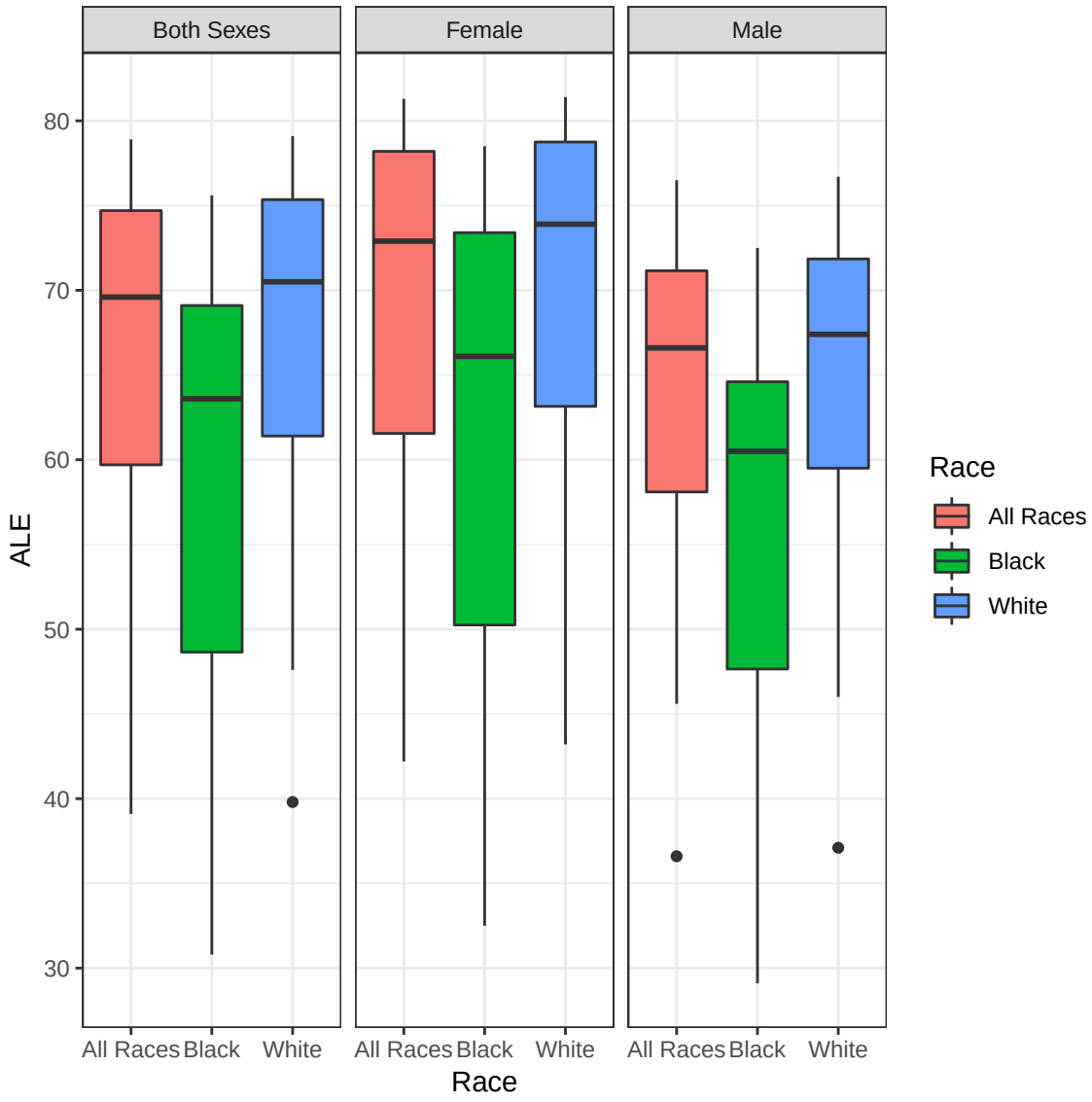



Figure 2: Box Plots of ALE by Sex and Race

Presented in Figure 2 are the side by side box plots of the different sexes based on race. Each box is separated by each sex category; Both Sexes, Female, and Male. Race is separated at the bottom and also by color. All Races is represented by red, Black by green, and White by blue. Female in general has a higher life expectancy than the other sex categories, but not significant enough due to the cross overs between plots. There are a few outliers that can be seen in White Males, All Males, and All Whites. All Whites have an outlier at age 40. All

Males have an outlier at age 36-37, and White Males an outlier around age 38. The box plots presented in 2 are skewed to the left for all box plots. This means that all means for each combination of race and sex will be higher than their corresponding medians.

The following code is used to produce a scatter plot of ALE vs. DR by Sex and Race.

```
(ggplot(data = LE, aes(x = DR, y = ALE, col = Sex,
                       shape = Race))
 + geom_point() + theme_bw())
```

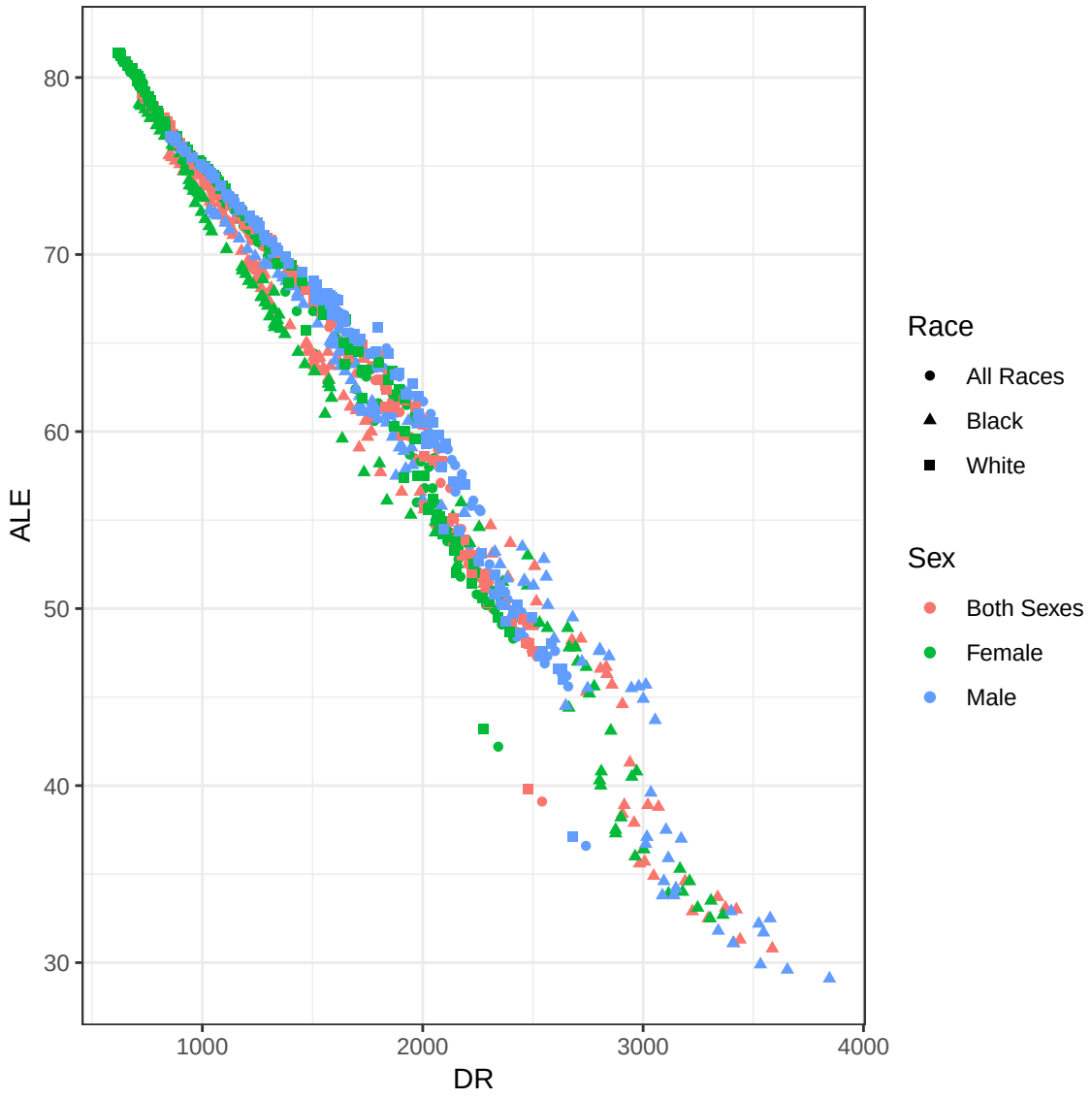



Figure 3: ALE vs. DR by Sex and Race

The plot given in Figure 3 shows ALE compared to the DR. The $x$ axis is represented by the death rate per 100,000 people. The $y$ axis is represented by the ALE. Much like in Figure 1, Figure 3 has data points represented by a combination of Sex and Race. Sex is represented by color, with Both Sexes being represented by red, Female by green, and Male by blue. Race is represented by shape, with All Races being represented by circles, Black by triangles, and White by squares. With this it is seen that the higher the ALE, the lower the DR, and the lower

the ALE, the higher the DR. This means that ALE and DR have a negative correlation. Even though the line of best fit falls apart around a death rate of 3000, there is still a downward trend and the negative correlation holds.

Table 1 and Table 2 are tables of the summaries of Sex and Race, respectively, with ALE. This produces the mean, median, minimum, maximum, quartile one and three, and the amount of data points per category in respect to ALE.

The following is used to produce a table of the summary data of ALE based on Sex.

```
Res <- (LE %>%
  group_by(Sex) %>%
  summarize(Mean = mean(ALE), Median = median(ALE),
            Minimum = min(ALE), Maximum = max(ALE),
            Q1 = quantile(ALE, .25),
            Q3 = quantile(ALE, .75), n = n()))
print(xtable(data.frame(Res),
               caption="Summary of Sex Data",
               label = "Table:Sex"))
```

| | Sex | Mean | Median | Minimum | Maximum | Q1 | Q3 | n |
|---|---|---|---|---|---|---|---|---|
| 1 | Both Sexes | 64.08 | 67.50 | 30.80 | 79.10 | 56.60 | 73.30 | 345 |
| 2 | Female | 66.74 | 70.70 | 32.50 | 81.40 | 58.00 | 76.80 | 345 |
| 3 | Male | 61.54 | 64.40 | 29.10 | 76.70 | 55.50 | 69.50 | 345 |

Table 1: Summary of Sex Data

ALE for Both Sexes has a mean of 64.08, a median of 67.5, the smallest and the largest ALE are 30.8 and 79.10 respectively. Due to the mean being smaller than the median, the data set will be skewed to the right. There are a total of 345 data points in Both Sexes, represented by n in Table 1. The first quartile has an ALE of 56.6 and the third quartile has an ALE of 73.3.

ALE for Female has a mean of 66.74, a median of 70.7, the smallest and the largest ALE for Female is 32.5 and 81.4 respectively. Due to the mean being smaller than the median, the data set will be skewed to the right. There are again a total of 345 data points in the Female category, represented by n in Table 1. The first quartile has an ALE of 58 and the third quartile has an ALE of 76.8.

ALE for Male has a mean of 61.54, a median of 64.4, the smallest ALE and the largest ALE for Male is 29.1 and 76.7 respectively. Again, the mean is smaller than the median in this data, meaning it is skewed to the right. The total data points is again 345. The first quartile has an ALE of 55.5 and the third quartile has an ALE of 69.5.

Out of the Sex variable presented by Table 1, the Male category has the smallest ALE on all counts. The Female category has the largest ALE on all counts. And the Both Sexes category, being that it contains both sexes, is the average of the two other categories.

The following code is used to produce a table of ALE based on Race

```
Res1 <- (LE %>%
 group_by(Race) %>%
 summarize(Mean = mean(ALE), Median = median(ALE),
           Minimum = min(ALE), Maximum = max(ALE),
           Q1 = quantile(ALE, .25),
           Q3 = quantile(ALE, .75), n = n()))


print(xtable(data.frame(Res1),
              caption="Summary of Race Data",
              label = "Table:Race"))
```

| | Race | Mean | Median | Minimum | Maximum | Q1 | Q3 | n |
|---|---|---|---|---|---|---|---|---|
| 1 | All Races | 66.59 | 68.40 | 36.60 | 81.30 | 59.60 | 74.70 | 345 |
| 2 | Black | 58.37 | 61.60 | 29.10 | 78.50 | 48.90 | 68.90 | 345 |
| 3 | White | 67.40 | 69.50 | 37.10 | 81.40 | 60.80 | 75.30 | 345 |

Table 2: Summary of Race Data

ALE for All Races has a mean of 66.59, a median of 68.4, the smallest and the largest ALE of 36.6 and 81.3 respectively. The mean here is lower than the median, meaning that the data will be skewed to the right. As shown in Table 2, All Races has a total of 345 data points. The first quartile has an ALE of 59.6 and a third quartile of 74.7.

ALE for Black has a mean of 58.37, a median of 61.6, the smallest and largest ALE of 29.1 and 78.5 respectively. The mean here is lower than the median, meaning that the data will be skewed to the right. As shown in Table 2, All Races has a total of 345 data points. The first quartile has an ALE of 48.9 and a third quartile of 68.9.

ALE for White has a mean of 67.4, a median of 69.5, the smallest and largest ALE of 37.1 and 81.4 respectively. The mean here is lower than the median, meaning that the data will be skewed to the right. As shown in Table 2, All Races has a total of 345 data points. The first quartile has an ALE of 60.8 and a third quartile of 75.3.

Out of the Race variable presented by Table 2, White has the highest ALE on all counts. Black has the lowest ALE on all counts. However, unlike Both Sexes before, All Races seems to be skewed by the white category. Comparing Table 2 with Table 1, it is noticeable that Black and Male, respectively, have the same minimum ALE of 29.1. This means that the lowest ALE is given by a black male data point with year being unclear. The categories of White and Female from each respective table has a maximum ALE of 81.4. This means that the highest ALE is given by a white female data point with year being unclear.

As the data description explains, the data for ALE and DR for the black race before 1960 was nonexistent, and therefore guessed upon using scattered data found [5]. Due to this, the data was made into a data set that only includes the data points for the years following 1960.

The following code is used to filter the data set to only include the years after 1960.

```
After1960 <- LE %>%
  filter(Year > 1960)
```

In order to remove the repetition of the data, the removal of All Races and Both Sexes in their corresponding variables of Race and Sex was done. The removal of the extra categories was applied to the data set after 1960 and renamed WARS (Without All Races and Sexes). The dimensions of WARS shows that there are a total of 216 rows and 5 columns after the removal of All Races and Sexes, a decrease from the 345 data points in each category before shown in Table 2 and Table 1.

The following code is used to remove All Races and Both Sexes from the dataset that represents the data after 1960.

```
NoAllRacesAfter <- (After1960 %>%
                      filter(Race == "Black"|Race == "White"))
WARS <- (NoAllRacesAfter %>%
           filter(Sex == "Male"|Sex == "Female"))
dim(WARS)

[1] 216   5
```

## 3.3 Spliting the data into Train and Test

In order to properly fit a model to the data, the data was divided into training and testing [6]. The data was split in a way that 80 percent is now in a training set and 20 percent is now in a testing set. The training includes 172 of the 216 observations. The testing includes 44 of the 216 observations. The training data will then be used in cross validation in order to find the best model of fit. A seed was set in order to keep the data constant due to the training and testing sets being randomly selected.

The following code is used to separate the data into training and testing.

```
set.seed(1)

WARSTrain <- WARS %>%

  sample_n(n()*0.8, replace = FALSE)

dim(WARSTrain)


[1] 172   5


# This is the test data set

WARSTest  <- anti_join(WARS, WARSTrain)

dim(WARSTest)


[1] 44  5
```

The following code is used to produce a new scatter plot of ALE vs. Year by Sex and Race with the training dataset.

```
(ggplot(WARSTrain, aes(x=Year, y= ALE,

                       color = Sex, shape = Race))

 + geom_point() + geom_smooth(method = "lm"))
```

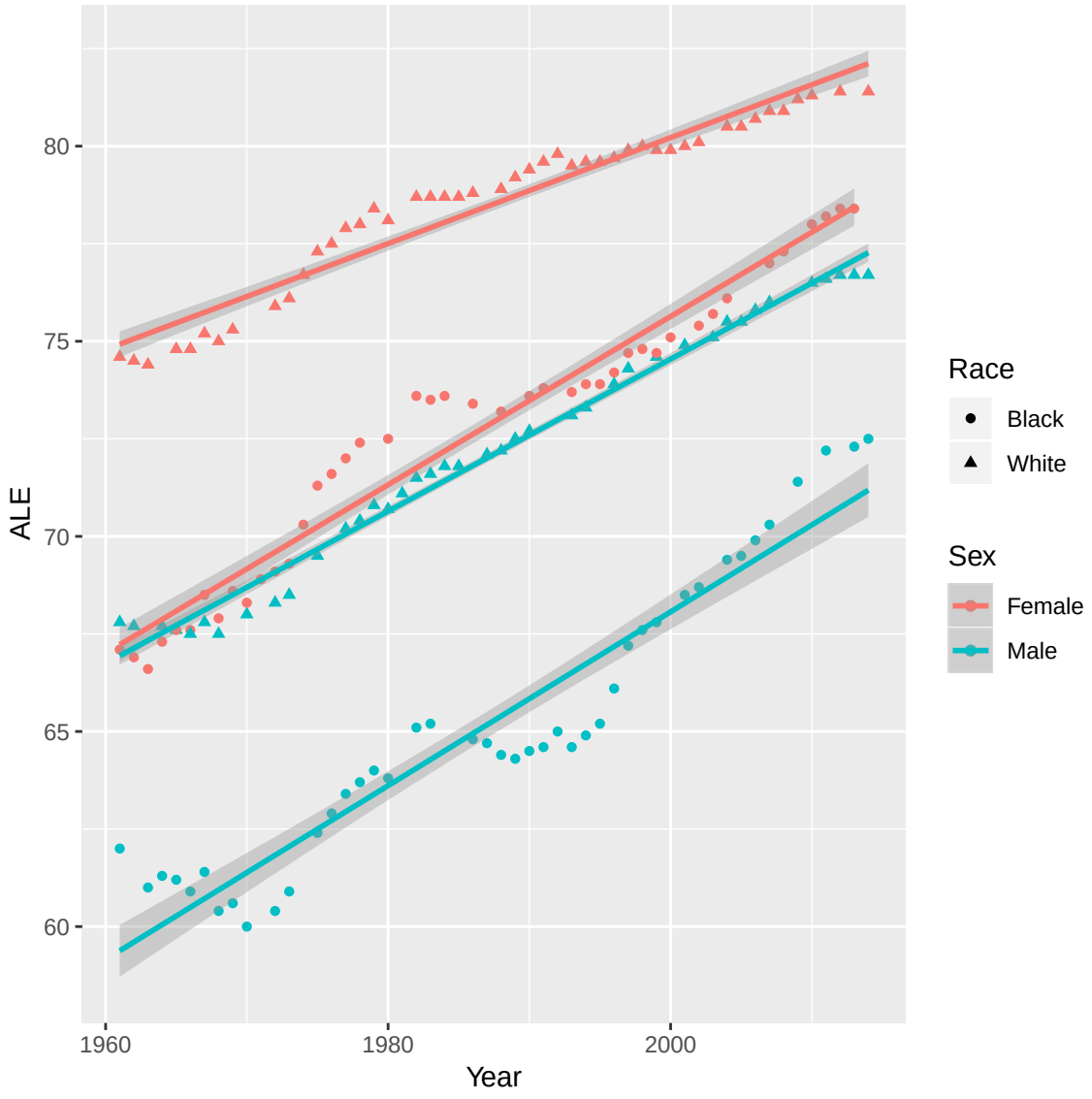



Figure 4: Training ALE vs. Year by Sex and Race

With the training dataset, a new plot of ALE and Year is made. In Figure 4, a major change has taken place since Figure 1. The x axis contains the $x$ variable year. The years now range only from 1960 to 2014. The $y$ variable represents the ALE of the following years based on race and sex. All Races has now been removed from the dataset presented, leaving the Race variable with Black, represented by circles, and White, represented by triangles. Both Sexes has been removed from the dataset presented as well, leaving the Sex variable with Female,

represented by red, and Male, represented by blue. White Females still have the highest ALE over the years with no cross over. Black Males still have the lowest ALE over the years with no cross over. However, Black Females and White Males overlap each other in some places, meaning that there is interaction between Race and Sex in multiple different Years.

## 3.4 Two Sample T-tests

### 3.4.1 Two sample T-test for the Average Life Expectancy for Male and Female

In order to see if there is a significant difference between Male and Female average life expectancy, a Two Sample T-test [6] is performed in order to compare the means of Male and Female. Before this test can be implemented, Male and Female variance need to be compared and shown that they are statistically the same. The use of the ansari.test is implemented to see if Female and Male have equal variances.

The null and alternative hypotheses are presented below for testing the variances of Male and Female:

$$H_0 : \sigma^2_{male} = \sigma^2_{female}$$

$$H_a : \sigma^2_{male} \neq \sigma^2_{female}$$

The following code is used to find the variances of Male and Female and then to test to see if those variances are equal.

```
# Finding variances
aggregate(ALE~Sex, data =WARSTrain, var)

     Sex      ALE
1 Female 17.46149
2   Male 23.52770

# Test for equal variances
ansari.test(ALE~Sex, WARSTrain)



Ansari-Bradley test


data:  ALE by Sex
AB = 3834, p-value = 0.8182
alternative hypothesis: true ratio of scales is not equal to 1
```

In order to reject the null and accept the alternative, a p-value of 0.05 or less will have to be presented [6]. According to the ansari test, the p-value is 0.8182, much greater than the 0.05 needed to reject the null. Therefore, there is failure to reject the null and a conclusion that the variances of Male and Female are statistically the same. Now the t-test can be implemented.

The null and alternative hypotheses are presented below for testing the means of Male and Female through T-testing:

$$H_0 : \mu_{male} = \mu_{female}$$

$$H_a : \mu_{male} \neq \mu_{female}$$

The following code is used to produce a t-test to see if Male and Female means are equal.

```
t.test(ALE~Sex, WARSTrain, var.equal = TRUE)



Two Sample t-test


data:  ALE by Sex

t = 10.366, df = 170, p-value < 2.2e-16

alternative hypothesis: true difference in means is not equal to 0

95 percent confidence interval:

 5.781997 8.502151

sample estimates:

mean in group Female   mean in group Male

            75.61798             68.47590
```

In order to reject the null and accept the alternative, a p-value of 0.05 or less will have to be presented [6]. According to the T-test that was implemented, the p-value is $2.2 * 10^{-16}$, much smaller than 0.05. Due to this, there is a rejection of the null hypothesis and there is enough evidence to conclude that the means of Male and Female are not equal, making them proper to use for modeling.

### 3.4.2 Two sample T test for the Average Life Expectancy for Black and White

In order to see if there is a significant difference between Black and White average life expectancy, a Two Sample T-test [6] is performed in order to compare the means of Black and White. Before this test can be implemented, Black and White variance need to be compared and shown that they have equal variances. The use of the ansari.test is implemented to see if Black and White have equal variances.

The null and alternative hypotheses are presented below for testing the variances of Black and White:

$$H_0 : \sigma^2_{black} = \sigma^2_{white}$$

$$H_a : \sigma^2_{black} \neq \sigma^2_{white}$$

The following code is used to find the variances of Black and White and then to test to see if those variances are equal.

```
# Finding variances
aggregate(ALE~Race, data =WARSTrain, var)

   Race      ALE
1 Black 26.37110
2 White 17.52334

# Test for normality of ALE distribution
ansari.test(ALE~Race, WARSTrain)



Ansari-Bradley test


data:  ALE by Race
AB = 3779.5, p-value = 0.8135
alternative hypothesis: true ratio of scales is not equal to 1
```

In order to reject the null hypothesis, a p-value of 0.05 or less will have to be presented [6]. According to the ansari test, the p-value is 0.8135, much greater than the 0.05 needed to reject the null hypothesis. Therefore, there is failure to reject the null and a conclusion that the variances of Black and White are statistically the same. Now the t-test can be implemented.

The null and alternative hypotheses are presented below for testing the means of Black and White through T-testing:

$$H_0 : \mu_{black} = \mu_{white}$$

$$H_a : \mu_{black} \neq \mu_{white}$$

The following code is used to produce a t-test to see if Black and White means are equal.

```
t.test(ALE~Race, WARSTrain, var.equal = TRUE)



Two Sample t-test


data:  ALE by Race

t = -9.3668, df = 170, p-value < 2.2e-16

alternative hypothesis: true difference in means is not equal to 0

95 percent confidence interval:

 -8.102144 -5.281577

sample estimates:

mean in group Black mean in group White

           68.82558            75.51744
```

In order to reject the null hypothesis, a p-value of 0.05 or less will have to be presented [6]. According to the T-test that was implemented, the p-value is $2.2 * 10^{-16}$, much smaller than 0.05. Due to this, there is a rejection of the null hypothesis and can make the decision that there is enough evidence to conclude that the means of Black and White are not equal, making them proper to use for modeling.

## 3.5 Fitting the Models

### 3.5.1 Generalized Linear Models

The Generalized Linear Model was used to fit the multiple linear models [6]. glm() is also used in order to present a $CV_{Error}$ later. The training data set was used to fit the models.

The following code gives the first model fit with basic variables.

```
Model1 <- glm(ALE ~ Sex + Race + Year, data = WARSTrain)

summary(Model1)



Call:

glm(formula = ALE ~ Sex + Race + Year, data = WARSTrain)


Deviance Residuals:

    Min       1Q   Median       3Q      Max

-2.4810  -0.5227   0.1290   0.5692   2.1336


Coefficients:

              Estimate Std. Error t value Pr(>|t|)

(Intercept) -3.081e+02  9.417e+00  -32.72   <2e-16 ***

SexMale     -7.025e+00  1.475e-01  -47.63   <2e-16 ***

RaceWhite    6.215e+00  1.475e-01   42.14   <2e-16 ***

Year         1.916e-01  4.741e-03   40.40   <2e-16 ***

---
```

```
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for gaussian family taken to be 0.9326088)

    Null deviance: 5656.61  on 171  degrees of freedom
Residual deviance:  156.68  on 168  degrees of freedom
AIC: 482.07

Number of Fisher Scoring iterations: 2
```

An equation was written in order to represent the model:

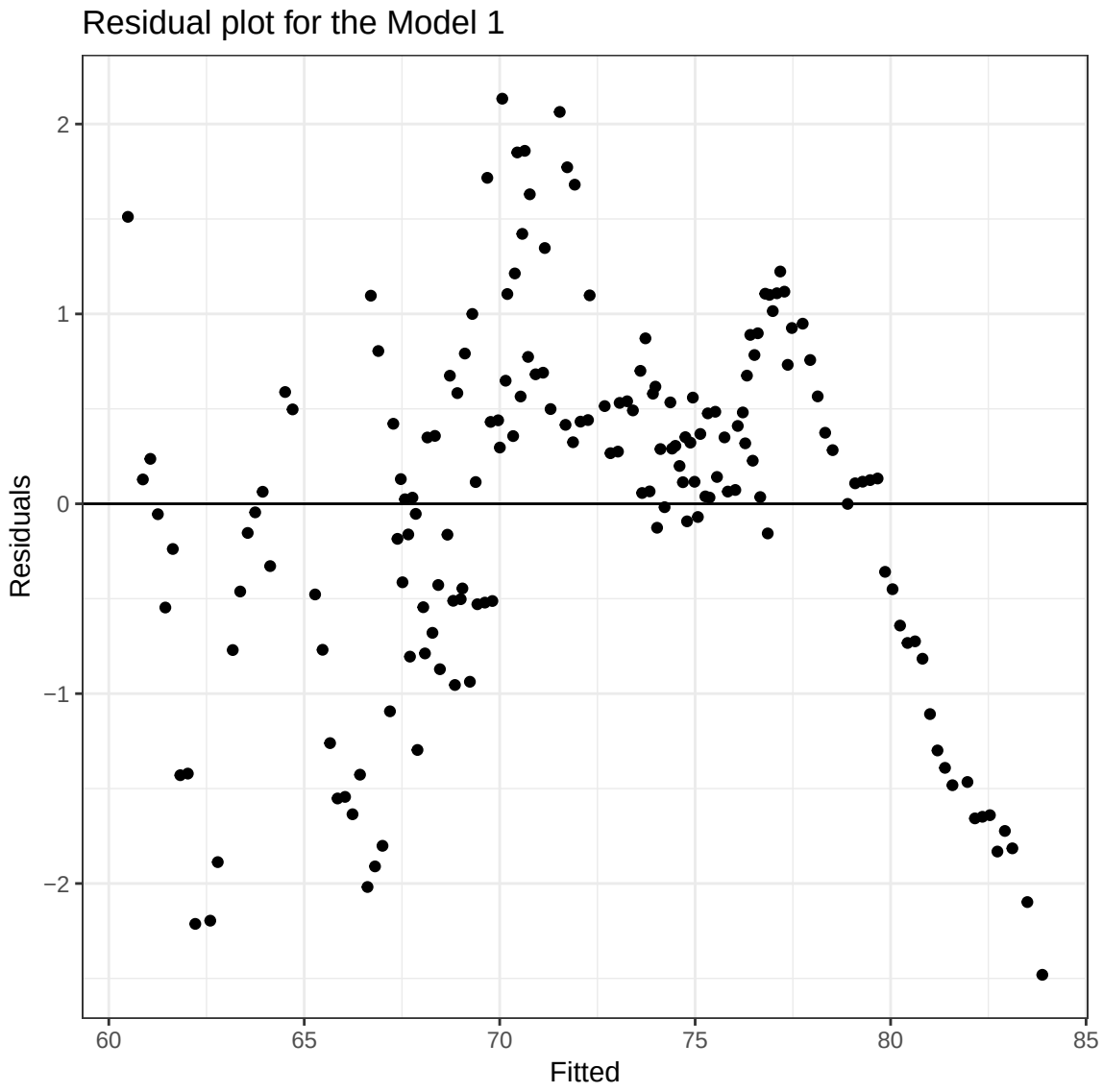
$$\hat{ALE} = -308.1226 - 7.0248 \cdot 1_{male}(x) + 6.2152 \cdot 1_{white}(x) +$$

$$0.1916 \cdot Year$$

The AIC of this model is 482.07. Due to no other models being made in order to compare, there is no way yet to see if this model is a good fit to the data [6]. Models will continue to be made to work with and compare. For now, Model 1 is the best model.

The following code produces a residual plot of the first model.

```
(ggplot(Model1, aes(x = .fitted, y = .resid))
  + geom_point() + geom_abline(intercept = 0, slope = 0)
  + labs(x = "Fitted", y = "Residuals",
         title = "Residual plot for the Model 1")
  + theme_bw())
```




The residuals are then plotted of Model 1. The residuals of the model are fairly random, indicating that the residuals of the model fit best to a linear model. The glm code will continue to be used.

In Figure 4, as mentioned before, there is an interaction between the Race and Sex variables between White Male and Black Female. An interaction GLM model was then made.

The following code produces the second model with basic variables and an interaction variable between Year and Race.

```
Model2 <- (glm(ALE ~ Sex + Race + Year + Year*Race,
               data = WARSTrain))
summary(Model2)
```

```
Call:
glm(formula = ALE ~ Sex + Race + Year + Year * Race, data = WARSTrain)

Deviance Residuals:
     Min        1Q    Median        3Q       Max
-2.18962  -0.46346   0.09682   0.52865   2.19351

Coefficients:
                Estimate Std. Error t value Pr(>|t|)
(Intercept)    -3.611e+02  1.199e+01 -30.111  < 2e-16 ***
SexMale        -7.046e+00  1.332e-01 -52.914  < 2e-16 ***
RaceWhite       1.127e+02  1.701e+01   6.626 4.54e-10 ***
Year            2.182e-01  6.038e-03  36.140  < 2e-16 ***
RaceWhite:Year -5.360e-02  8.561e-03  -6.261 3.12e-09 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
(Dispersion parameter for gaussian family taken to be 0.7598397)

    Null deviance: 5656.61  on 171  degrees of freedom
Residual deviance:  126.89  on 167  degrees of freedom
AIC: 447.8

Number of Fisher Scoring iterations: 2
```

An equation was fit to Model 2:

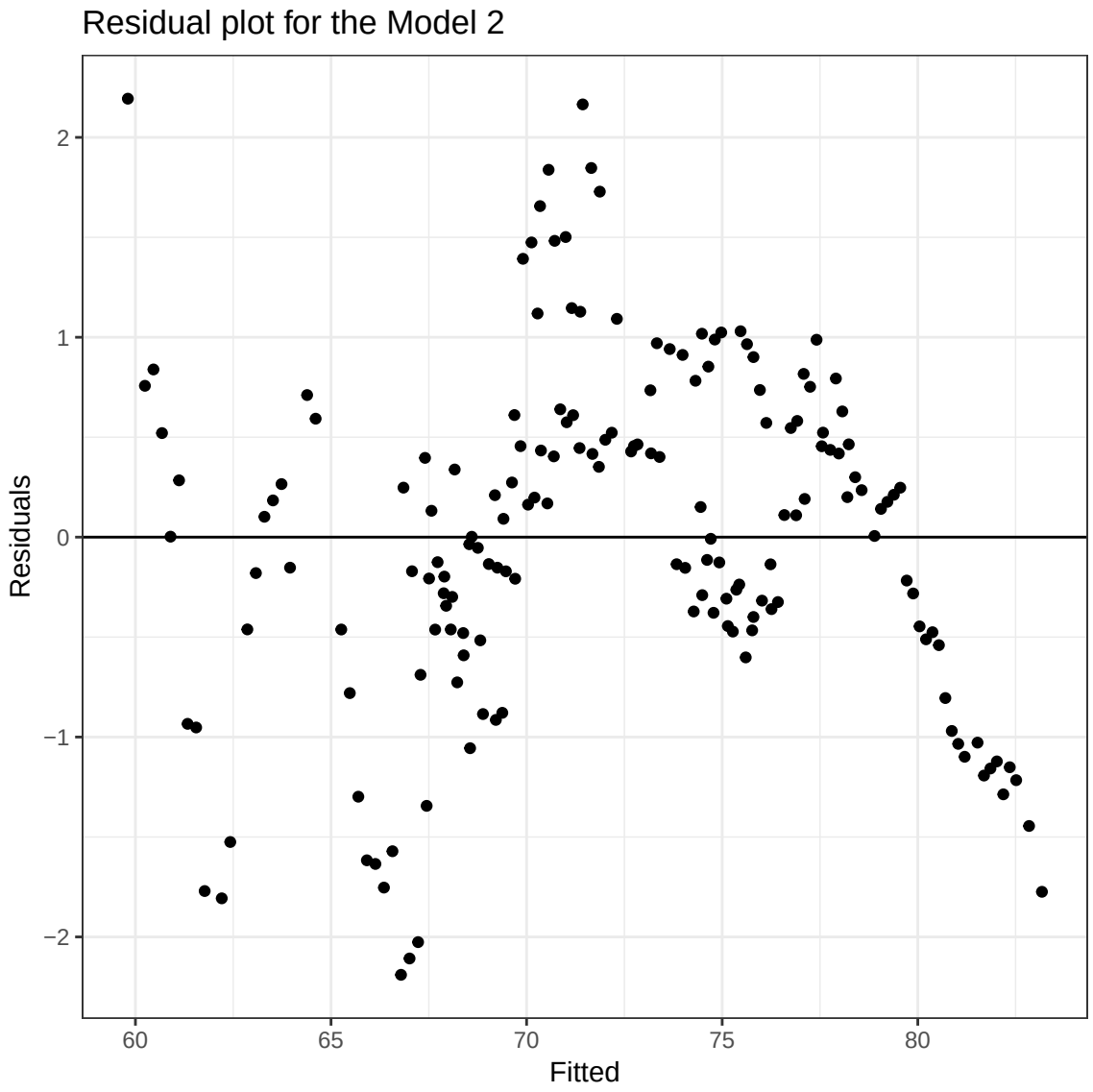
$$\hat{ALE} = -361.0831 - 7.0458 \cdot 1_{male}(x) + 112.7060 \cdot 1_{white}(x) +$$

$$0.2182 \cdot Year - 0.0536 \cdot 1_{white}(x) \cdot Year$$

The AIC of this model is 447.8. Model 2 has a better AIC than Model 1, therefore making Model 2 a better fit to the data. Models will continue to be made in order to see if more interactions will give a better fit.

The following code produces a residual plot of the second model.

```
(ggplot(Model2, aes(x = .fitted, y = .resid))
  + geom_point() + geom_abline(intercept = 0, slope = 0)
  + labs(x = "Fitted", y = "Residuals",
         title = "Residual plot for the Model 2")
  + theme_bw())
```




The residuals are then plotted of Model 2. The residuals of the model are fairly random, indicating that the residuals of the model fit best to a linear model. The glm code will continue to be used.

The following code produces the third model with basic variables and an interaction variable between Race and Sex.

```
Model3 <- (glm(ALE ~ Sex + Race + Year + Race*Sex,
               data = WARSTrain))
summary(Model3)



Call:
glm(formula = ALE ~ Sex + Race + Year + Race * Sex, data = WARSTrain)


Deviance Residuals:
    Min       1Q   Median       3Q      Max
-2.2110  -0.5797   0.1068   0.5706   2.4245


Coefficients:
                   Estimate Std. Error t value Pr(>|t|)
(Intercept)       -3.088e+02  8.957e+00 -34.477  < 2e-16 ***
SexMale           -7.631e+00  1.982e-01 -38.510  < 2e-16 ***
RaceWhite          5.628e+00  1.950e-01  28.860  < 2e-16 ***
Year               1.920e-01  4.511e-03  42.578  < 2e-16 ***
SexMale:RaceWhite  1.215e+00  2.805e-01   4.331 2.55e-05 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
(Dispersion parameter for gaussian family taken to be 0.8434429)

    Null deviance: 5656.61  on 171  degrees of freedom
Residual deviance:  140.85  on 167  degrees of freedom
AIC: 465.76

Number of Fisher Scoring iterations: 2
```

An equation fit to Model 3:

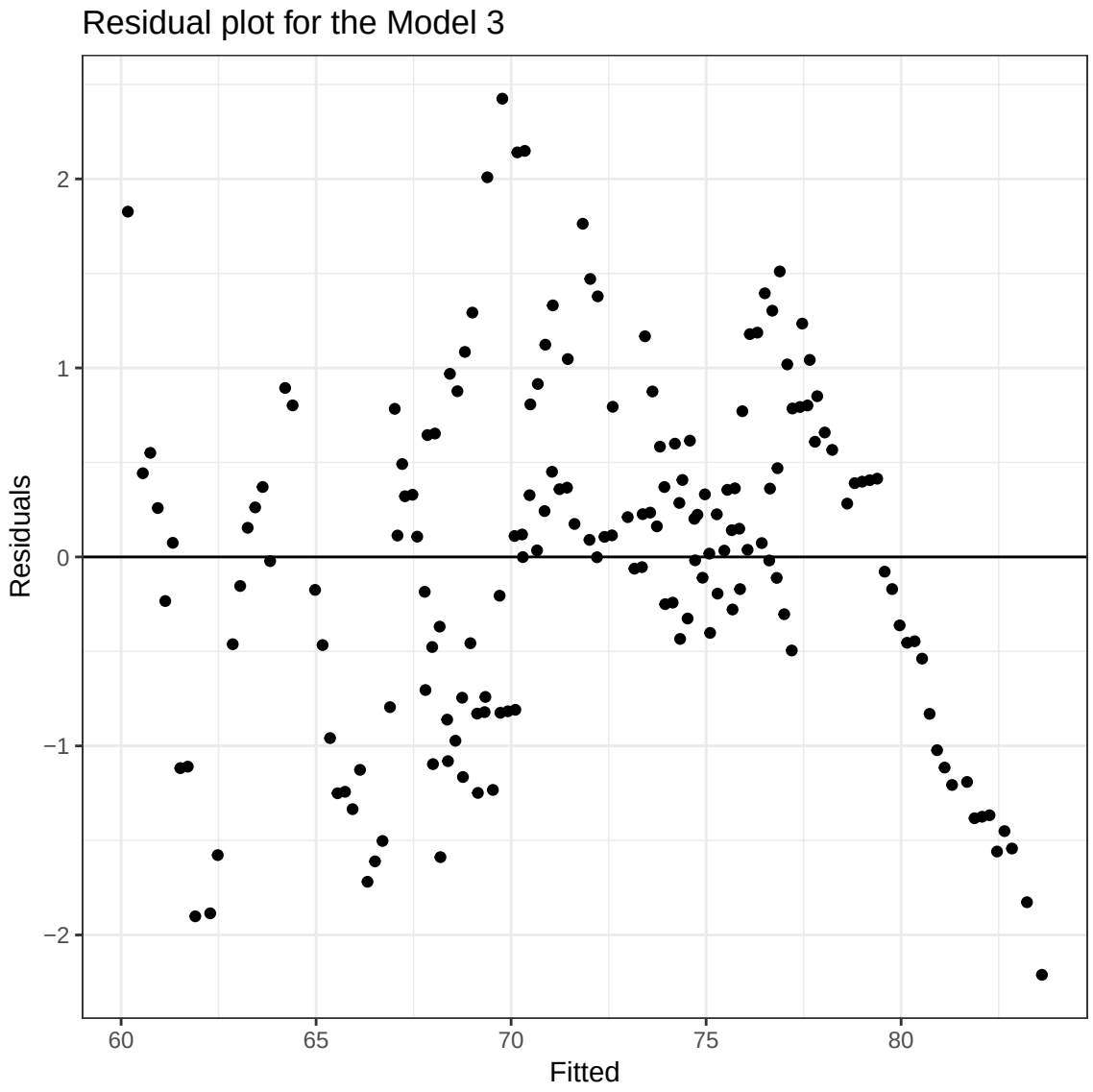
$$\hat{ALE} = -308.8012 - 7.6311 \cdot 1_{male}(x) + 5.6282 \cdot 1_{white}(x) +$$

$$0.1920 \cdot Year + 1.2151 \cdot 1_{male}(x) \cdot 1_{white}(x)$$

The AIC for Model 3 is 465.76. Due to Model 2 having a lower AIC than Model 3, Model 2 is still the model of best fit.

The following code produces a residual plot of the third model.

```
(ggplot(Model3, aes(x = .fitted, y = .resid))
  + geom_point() + geom_abline(intercept = 0, slope = 0)
  + labs(x = "Fitted", y = "Residuals",
         title = "Residual plot for the Model 3")
  + theme_bw())
```




The residuals are then plotted of Model 3. The residuals of the model are fairly random, indicating that the residuals of the model fit best to a linear model. The glm code will continue to be used.

The following code produces the fourth model with basic variables and an interaction variable between Year and Sex.

```
Model4 <- (glm(ALE ~ Sex + Race + Year + Year*Sex,
               data = WARSTrain))
summary(Model4)
```

```
Call:
glm(formula = ALE ~ Sex + Race + Year + Year * Sex, data = WARSTrain)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-2.1155  -0.5652   0.1433   0.5918   1.9962

Coefficients:
               Estimate Std. Error t value Pr(>|t|)
(Intercept)  -2.755e+02  1.265e+01 -21.766  < 2e-16 ***
SexMale      -7.434e+01  1.817e+01  -4.093 6.63e-05 ***
RaceWhite     6.229e+00  1.422e-01  43.792  < 2e-16 ***
Year          1.751e-01  6.372e-03  27.481  < 2e-16 ***
SexMale:Year  3.388e-02  9.143e-03   3.706 0.000286 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
(Dispersion parameter for gaussian family taken to be 0.8668991)

    Null deviance: 5656.61  on 171  degrees of freedom
Residual deviance:  144.77  on 167  degrees of freedom
AIC: 470.47

Number of Fisher Scoring iterations: 2
```

An equation fit to Model 4:

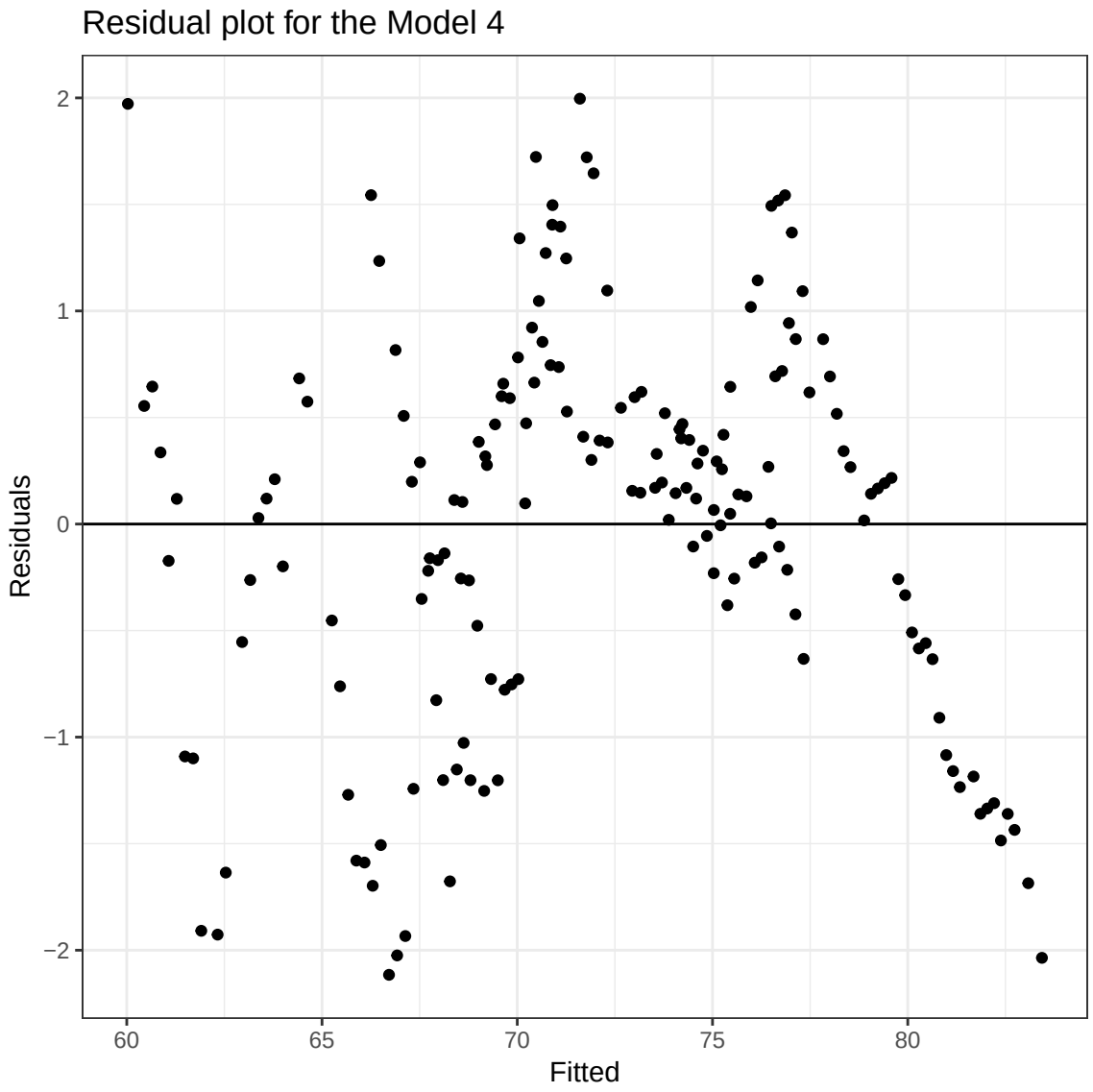
$$\hat{ALE} = -275.4517 - 74.3419 \cdot 1_{male}(x) + 6.2286 \cdot 1_{white}(x) +$$

$$0.1751 \cdot Year + 0.0339 \cdot 1_{male}(x) \cdot Year$$

The AIC for Model 4 is 470.47. This AIC is larger than the AIC presented in Model 2. Therefore, Model 2 is still the best model of fit.

The following code produces a residual plot of the fourth model.

```
(ggplot(Model4, aes(x = .fitted, y = .resid))
  + geom_point() + geom_abline(intercept = 0, slope = 0)
  + labs(x = "Fitted", y = "Residuals",
         title = "Residual plot for the Model 4")
  + theme_bw())
```




The residuals are then plotted of Model 4. The residuals of the model are fairly random, indicating that the residuals of the model fit best to a linear model. The glm code will continue to be used.

The following code produces the fifth model with basic variables and two interaction variables: Year and Sex, Year and Race.

```
Model5 <- (glm(ALE ~ Sex + Race + Year + Year*Sex + Year*Race,
               data = WARSTrain))
summary(Model5)
```

```
Call:
glm(formula = ALE ~ Sex + Race + Year + Year * Sex + Year * Race,
    data = WARSTrain)

Deviance Residuals:
     Min        1Q    Median        3Q       Max
-2.28960  -0.57743   0.02435   0.55551   2.66716

Coefficients:
                Estimate Std. Error t value Pr(>|t|)
(Intercept)    -3.282e+02  1.382e+01 -23.742  < 2e-16 ***
SexMale        -7.554e+01  1.621e+01  -4.660 6.45e-06 ***
RaceWhite       1.135e+02  1.621e+01   6.999 6.06e-11 ***
Year            2.017e-01  6.961e-03  28.972  < 2e-16 ***
SexMale:Year    3.447e-02  8.158e-03   4.226 3.92e-05 ***
RaceWhite:Year -5.398e-02  8.160e-03  -6.615 4.87e-10 ***
```

```
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for gaussian family taken to be 0.6901729)

    Null deviance: 5656.61  on 171  degrees of freedom
Residual deviance:  114.57  on 166  degrees of freedom
AIC: 432.23

Number of Fisher Scoring iterations: 2
```

An equation fit to Model 5:

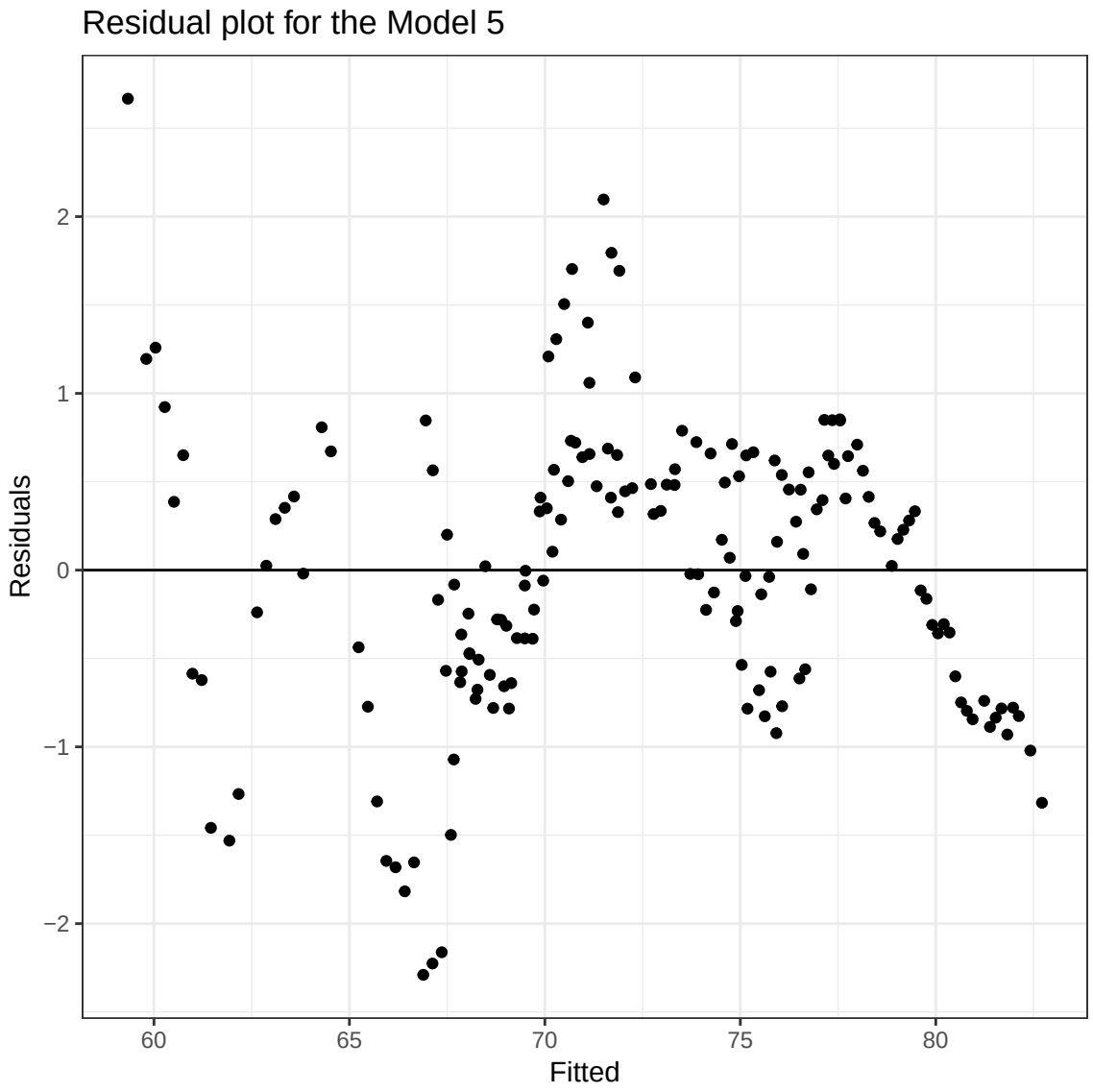
$$\hat{ALE} = -328.2151 - 75.5398 \cdot 1_{male}(x) + 113.4714 \cdot 1_{white}(x) +$$

$$0.2017 \cdot Year + 0.0345 \cdot 1_{male}(x) \cdot Year - 0.05398 \cdot 1_{white}(x) \cdot Year$$

The AIC of Model 5 is 432.23. Model 5 has a lower AIC than all other models presented so far. Therefore, it is currently the model of best fit.

The code for a residual plot of the fifth model follows.

```
(ggplot(Model5, aes(x = .fitted, y = .resid))
  + geom_point() + geom_abline(intercept = 0, slope = 0)
  + labs(x = "Fitted", y = "Residuals",
         title = "Residual plot for the Model 5")
  + theme_bw())
```




The residuals are then plotted of Model 5. The residuals of the model are fairly random, indicating that the residuals of the model fit best to a linear model. The glm code will continue to be used.

The following code produces the sixth model with basic variables and three interaction variables: Year and Sex, Year and Race, Sex and Race.

```
Model6 <- (glm(ALE ~ Sex + Race + Year + Year*Sex + Year*Race
               + Sex*Race, data = WARSTrain))
summary(Model6)
```

```
Call:
glm(formula = ALE ~ Sex + Race + Year + Year * Sex + Year * Race +
    Sex * Race, data = WARSTrain)

Deviance Residuals:
     Min        1Q    Median        3Q       Max
-1.98985  -0.49052   0.01572   0.45951   2.96937

Coefficients:
                   Estimate Std. Error t value Pr(>|t|)
(Intercept)       -3.310e+02  1.290e+01 -25.656  < 2e-16 ***
SexMale           -7.324e+01  1.512e+01  -4.844 2.91e-06 ***
RaceWhite          1.142e+02  1.512e+01   7.558 2.69e-12 ***
Year               2.032e-01  6.497e-03  31.278  < 2e-16 ***
SexMale:Year       3.301e-02  7.612e-03   4.337 2.51e-05 ***
RaceWhite:Year    -5.466e-02  7.609e-03  -7.184 2.22e-11 ***
```

```
SexMale:RaceWhite  1.207e+00  2.368e-01   5.096 9.42e-07 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for gaussian family taken to be 0.5999446)

    Null deviance: 5656.610  on 171  degrees of freedom
Residual deviance:   98.991  on 165  degrees of freedom
AIC: 409.09

Number of Fisher Scoring iterations: 2
```

An equation for Model 6:

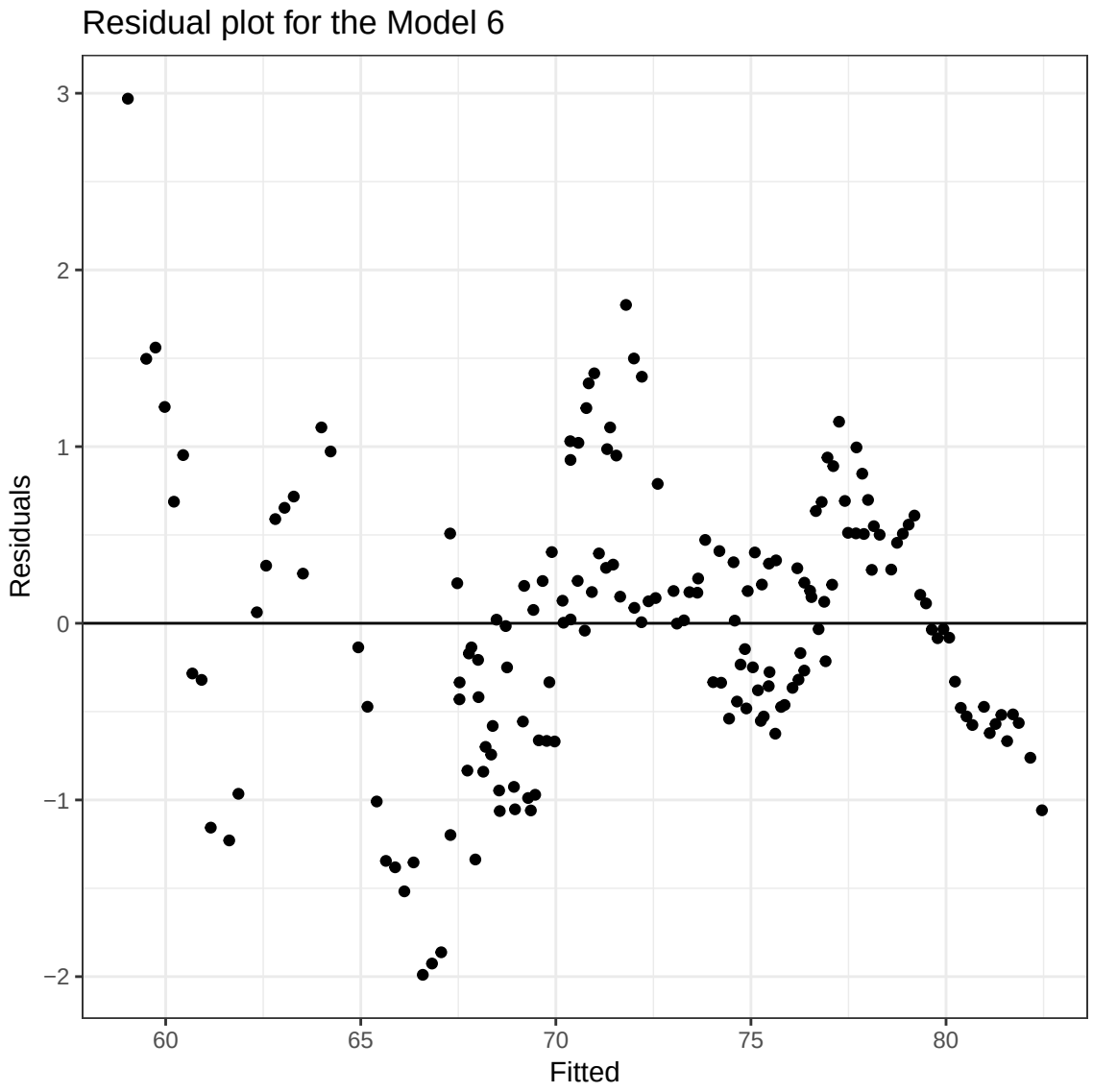
$$\hat{ALE} = -330.9723 - 73.2353 \cdot 1_{male}(x) + 114.2401 \cdot 1_{white}(x) +$$

$$0.2032 \cdot Year + 0.03301 \cdot 1_{male}(x) \cdot Year - 0.0547 \cdot 1_{white}(x) \cdot Year + 1.2067 \cdot 1_{male}(x) \cdot 1_{white}(x)$$

The AIC for Model 6 is 409.09, making it the smallest AIC of all other models made. Therefore, the model of best fit is Model 6.

A residual plot of the sixth model follows.

```
(ggplot(Model6, aes(x = .fitted, y = .resid))
  + geom_point() + geom_abline(intercept = 0, slope = 0)
  + labs(x = "Fitted", y = "Residuals",
         title = "Residual plot for the Model 6") + theme_bw())
```




The residuals are then plotted of Model 6. The residuals of the model are fairly random, indicating that the residuals of the model fit best to a linear model. The residuals for Model 6 seem to be the most scattered throughout compared to the rest of the models presented.

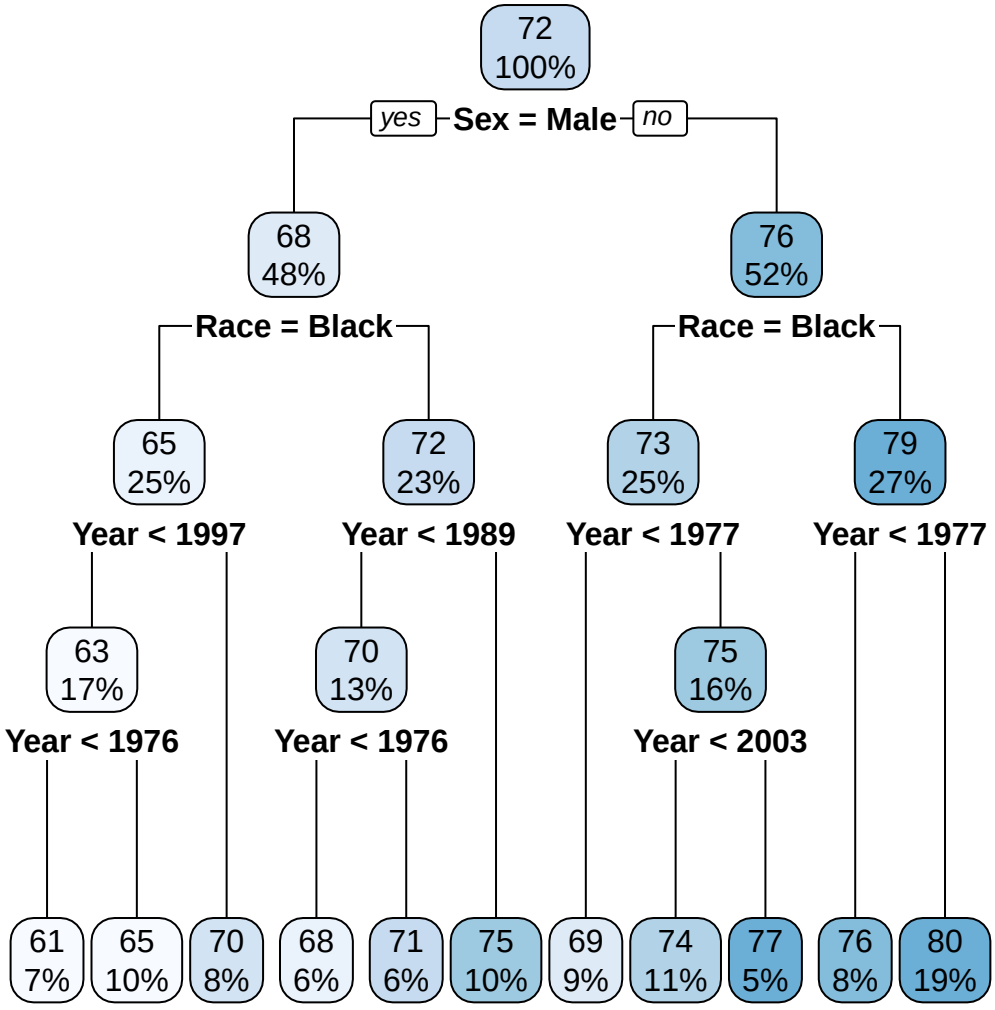
### 3.5.2 Regression Tree

In order to have some variation in models, a regression tree was made. This is another way to fit a model, but can seem complicated. While trees are easy to interpret and fit data nicely, they tend to be unstable with high variance due to over-fitting. A slight change in the training data can cause a significant difference in the model. [3]

The training dataset is used here as well to make a regression tree.

The following code is used to produce a regression tree of the training data.

```
library(rpart)

WARSTree <- rpart(ALE ~ Year + Sex + Race, data=WARSTrain)

dim(WARSTrain)


[1] 172   5


library(rpart.plot)

rpart.plot(WARSTree)
```




The tree starts off with a mean ALE of 72 for 100 percent of the data. The first branch separates based off if the Sex is Male. If yes, following the left branch gives 48 percent of the data having a mean ALE of 68. That 48 percent of the data represents All Males. The next branch asks if the Race is Black. If yes, then following the left branch gives 25 percent of the data having a mean ALE of 65. That 25 percent of the data is Black Males. The next question asks if the Year is less than 1997. If yes, following the left branch gives 17 percent of the data having a mean ALE of 63. That 17 percent represents Black Males born before 1997. The final question for this branch asks if the Year is less than 1976. If yes, following the left branch gives 7 percent of the data having a mean ALE of 61. That 7 percent represents Black Males

born before 1976.

Going back one branch, where a Black Male was born before 1997, the question asked again is if the Year is less than 1976. If not, following the right branch gives 10 percent of the data having a mean ALE of 65. That 10 percent represents Black Males born before 1997 but after 1976. Going back to the branch representing only Black Males, the question is asked again if the year is before 1997. If not, following the right branch gives 8 percent of the data with a mean ALE of 70. That 8 percent represents Black Males born after 1997.

Now going back to the question if the Race is Black. If not, following the right branch gives 23 percent of the data with a mean ALE of 72. That 23 percent represents White Men. The next question is if the Year is before 1989. If yes, following the left branch gives 13 percent of the data with a mean ALE of 70. This 13 percent represents White Males born before 1989. The next question is if the Year is before 1976. If yes, following the left branch gives 6 percent of the data with a mean ALE of 68. That 6 percent represents White Males born before 1976.

Going back one branch, where a White Male was born before 1989, the question asked again is if the Year is less than 1976. If not, following the right branch gives 6 percent of the data having a mean ALE of 71. This 6 percent represents White Males born before 1989 but after 1976. Going back to the branch representing White Males, the question is asked again if the Year is before 1989. If not, following the right branch gives 10 percent of the data with a mean ALE of 75. This 10 percent represents White Males born after 1989. With this, the left side of the tree is finished.

Starting back at the top of the tree, the question is asked again if the Sex is Male. If not, following the right branch gives 52 percent of the data with a mean ALE of 76. This 52 percent of the data represents All Females. The next question asks if the Race is Black. If yes, following the left branch gives 25 percent of the data with a mean ALE of 73. This 25 percent represents Black Females. The next question is if the Year is before 1977. If yes, following the left branch gives 9 percent of the data with a mean ALE of 69. This 9 percent represents Black Females born before the year 1977.

Going back to the branch that represents Black Females, the question is asked if the Year is before 1977. If no, following the right branch gives 16 percent of the data with a mean ALE of 75. The next question asked is if the year is before 2003. If yes, following the left branch gives 11 percent of the data with a mean ALE of 74. This 11 percent represents Black Females born before the year 2003 but after 1977. Going back to Black Females born after 1977. The question asked again is if the year is before 2003. If not, following the right branch gives 5 percent of the data with a mean ALE of 77. This 5 percent represents Black Females born after 2003.

Going back to the branch that represents Females, the question is asked if Race is Black. If not, following the right branch gives 27 percent of the data with a mean ALE of 79. The next question asked is if the year is before 1977. If yes, following the left branch gives 8 percent of the data with a mean ALE of 76. This 8 percent represents White Females born before the year 1977. If not, following the right branch gives 19 percent of the data with a mean ALE of 80. This 19 percent represents White Females born after the year 1977.

## 3.6 Validating the models using 5-Fold Cross Validation

Five Fold Cross Validation was used to select the best model amount the six fitted models [6].

The following code is used to produce a table of the models with their corresponding cross validation error and AIC.

```
library(boot)

#Run the CV with 5 folds

model2_CV <- cv.glm(WARSTrain, Model1, K=5)

#model2_CV$delta

model3_CV <- cv.glm(WARSTrain, Model2, K=5)

#model3_CV$delta

model4_CV <- cv.glm(WARSTrain, Model3, K=5)

#model4_CV$delta

model5_CV <- cv.glm(WARSTrain, Model4, K=5)

#model5_CV$delta

model6_CV <- cv.glm(WARSTrain, Model5, K=5)

#model6_CV$delta

model7_CV <- cv.glm(WARSTrain, Model6, K=5)

#model7_CV$delta

Model_Name <- (c("Model 1", "Model 2", "Model 3", "Model 4",

                 "Model 5", "Model 6"))
```

```
CV_Error <- (round(c(model2_CV$delta[1], model3_CV$delta[1],
                     model4_CV$delta[1], model5_CV$delta[1],
                     model6_CV$delta[1], model7_CV$delta[1]), 4))
AIC <- (round(c(summary(Model1)$aic, summary(Model2)$aic,
                summary(Model3)$aic, summary(Model4)$aic,
                summary(Model5)$aic, summary(Model6)$aic), 4))
print(xtable(data.frame(cbind(Model_Name, CV_Error, AIC)),
              caption="Error and AIC Summary",
             label = "Table:AIC"))
```

| | Model_Name | CV_Error | AIC |
|---|---|---|---|
| 1 | Model 1 | 0.9594 | 482.0673 |
| 2 | Model 2 | 0.7794 | 447.8013 |
| 3 | Model 3 | 0.8551 | 465.7555 |
| 4 | Model 4 | 0.9138 | 470.4735 |
| 5 | Model 5 | 0.7056 | 432.2279 |
| 6 | Model 6 | 0.6167 | 409.0905 |

Table 3: Error and AIC Summary

A low valued $CV_{Error}$ is wanted, along with a low AIC [6]. As presented in Table 3, Model 6 has the lowest $CV_{Error}$ and lowest AIC, making it the model of best fit compared to the other models.

## 3.7 Predictions

The Test Square Prediction Error (TSPE), as defined in the following equation, was used to compare the predictive powers among the models using the test data. The smallest TSPE indicates the tightest fit of the model to the data. [6]

$$TSPE = \frac{\sum (\mathbf{y_{test}} - \hat{\mathbf{y}}_{\mathbf{test}})^2}{n_{test}}$$

The following code is used to produce a table of the models and the regression tree with their corresponding Test Square Prediction Error (TSPE).

```
fittedValues2 <- predict(Model1, newdata = WARSTest)

obsValues <- WARSTest$ALE

TSPE2 <- mean((obsValues - fittedValues2)^2)

fittedValues3 <- predict(Model2, newdata = WARSTest)

TSPE3 <- mean((obsValues - fittedValues3)^2)

fittedValues4 <- predict(Model3, newdata = WARSTest)

TSPE4 <- mean((obsValues - fittedValues4)^2)

fittedValues5 <- predict(Model4, newdata = WARSTest)

TSPE5 <- mean((obsValues - fittedValues5)^2)

fittedValues6 <- predict(Model5, newdata = WARSTest)

TSPE6 <- mean((obsValues - fittedValues6)^2)

fittedValues7 <- predict(Model6, newdata = WARSTest)

TSPE7 <- mean((obsValues - fittedValues7)^2)

fittedValuesTree <- predict(WARSTree, newdata = WARSTest)
```

```
TSPE8 <- mean((obsValues - fittedValuesTree)^2)

Model_Name <- (c("Model 1", "Model 2", "Model 3", "Model 4",

                 "Model 5", "Model 6", "Tree Model"))

TSPE <- round(c(TSPE2, TSPE3, TSPE4, TSPE5, TSPE6, TSPE7, TSPE8), 4)

print(xtable(data.frame(cbind(Model_Name, TSPE)),

             caption="TSPE Summary", label = "Table:TSPE"))
```

| | Model_Name | TSPE |
|---|---|---|
| 1 | Model 1 | 0.9965 |
| 2 | Model 2 | 0.8117 |
| 3 | Model 3 | 0.9069 |
| 4 | Model 4 | 0.8249 |
| 5 | Model 5 | 0.6448 |
| 6 | Model 6 | 0.5548 |
| 7 | Tree Model | 1.1492 |

Table 4: TSPE Summary

Table 4 is the summary of the TSPE for all models, including the Tree Model. As seen in Table 4, Model 6 has the smallest TSPE. Model 6 has passed all tests in order to see if it is the best fit for the model. Therefore, the equation for the calculator is as follows:

$$\hat{ALE} = -330.9723 - 73.2353 \cdot 1_{male}(x) + 114.2401 \cdot 1_{white}(x) +$$

$$0.2032 \cdot Year + 0.03301 \cdot 1_{male}(x) \cdot Year - 0.0547 \cdot 1_{white}(x) \cdot Year + 1.2067 \cdot 1_{male}(x) \cdot 1_{white}(x)$$

# 4 Calculator Description

Now that Model 6 has been chosen, the calculator can be made. The calculator was implemented in Excel.

The first sheet of the Excel sheet is the calculator itself, where it asks for inputs and shows the needed outputs. There are several questions asked in order to properly calculate what is needed to figure out the Average Life Expectancy (ALE) for the user and how much the user needs to save each month until the age of retirement. The year the user was born, the user's race, and the user's sex is used in our model to find the ALE. There are also several questions that help us figure out the total expenses and savings the user will have in retirement. The current income before taxes and how long someone has been working helps figure out the amount of social security the user currently has. The current income before taxes and the questions about the user's 401(k) help us figure out how much money the user currently has and how much money the user will have in their 401(k) at the time of retirement.

The calculations page is the backbone of the calculator. This is where the calculations are done.

The cost of living by age (2013) is the first table shown [2]. This table helps disclose the cost of living later on in life. The later sets of age, 65-74 and greater than 75 are being brought forward to the age of retirement from 2013. This is done by using the rate of inflation plus 1 to the power of the years from 2013 until retirement multiplied by the monthly cost of living for 65-74 year old in the year of 2013. The same happens for the age range greater than 75, except that the years will be from the year 2013 to the age of 74 because it is end of year. All monthly payments made in the future will be brought back to the age of retirement.

The Rates table holds the rates used to either find the present value or future value of certain calculations. These will be described later when talking about the actual calculations.

The Variables table holds the betas and data variables needed to find out the output of ALE. The Data variable section comes from the calculator's inputs. The only output variable given from this section is the ALE, which is presented on the calculator page in the outputs table.

The current contribution section produces the current savings from Social Security, Savings, and 401(k). While Savings and 401(k) are given to us by the user of the calculator, Social Security is not. This is calculated by using the taxes taken out of the users paycheck each month and matched by the individual's employer for social security per year [10]. This is given in the rates table. The amount of time someone has been working helps give the total amount that the user has contributed to social security per year based off of the current income [10].

Finally the TVM (Time Value of Money) calculations. The total cost of living from retirement to age 74 is found by getting the present value of the monthly cost of living payments brought back to the age of retirement, using the cost of living each month that was calculated in the Retirement Cost of Living per month table using the inflation rate. If the ALE does not go past the age of 65, then they will not need savings for retirement due to the minimum age of retirement being 65. The total cost of living to age 74 to the ALE is found by taking the present value of all monthly cost of living payments from 74 until death, using the RCL table and the inflation rate, and bringing them back to the age of retirement as our time 0. Total 401(k) contributions are found by finding the future value at the age of retirement of the total payments made each month to the user's 401(k) made by them and their employer with a present value of their current 401(k) using the interest rate, both given by the calculator inputs. Social Security contribution is found by finding the future value at the age of retirement of the total payments made to social security with a present value of what is currently there using the inflation as the interest rate. All of these values found are then added up to show how much money the user needs to save up to to have a proper amount of funds at the age of retirement. The ability to add them up comes from having them all at the same year, the age of retirement. Finally, the payment per month needed is calculated by using the current savings as the present value and the money needed at retirement as the future value using the interest rate given by the rates table for the average savings account [9]. This value is then presented on the calculator page as an output. All calculations are made using TVM functions in Excel, such as PMT, FV, and PV which find payments needed, the future value, and the present value respectively.

## 4.1 Calculator Visual

The following pictures inserted are the inputs, outputs, and calculations used for the calculator. The inputs and outputs are presented on the first page of the Excel sheet called Calculator. The calculations and tables are presented on the second page of the Excel sheet called Calculations. The information is based off of a 23 year old white female.

| Questions | Inputs |
|---|---|
| What year were you born? | 1996 |
| What is the current year? | 2019 |
| What age do you plan to retire? | 67 |
| Which race describes you best? (Black or White) | White |
| What sex were you assigned at birth? (Male or Female) | Female |
| What is your current income? (Before Taxes) | $47,500 |
| What is your current 401(k)? (In dollars) | $0 |
| How much do you plan to contribute each month to your 401(k)? (No 401(k) = 0%) (% of Income Before Taxes) | 8.00% |
| Does your employer match your 401(k)? (If so, list percentage. If not, put 0%) | 5.00% |
| How much interest does your 401(k) earn per year? (If NA, =0%) | 5.00% |
| How long have you been working? (In years) | 0 |
| What is in your current Savings Account? | $2,000 |

| Answers | Outputs |
|---|---|
| Average Life Expectancy | 79.67 |
| What you should save each month until Retirement | $279.55 |
| | If negative, no need to save |

| Cost of Living by age in 2013 | | |
|---|---|---|
| **Age** | **Per year** | **Monthly** |
| <25 | $30,373.00 | $2,531.08 |
| 25-34 | $48,087.00 | $4,007.25 |
| 35-44 | $58,784.00 | $4,898.67 |
| 45-54 | $60,524.00 | $5,043.67 |
| 55-64 | $55,892.00 | $4,657.67 |
| 65-74 | $46,757.00 | $3,896.42 |
| >75 | $34,382.00 | $2,865.17 |

| | **Retirement to 74** | **74 to ALE** |
|---|---|---|
| Retirement Cost of Living per month | $10,487.55 | $7,866.09 |

| **X Variables** | **Betas** | **Data Variable** |
|---|---|---|
| Sex | -73.2353 | 0 |
| Race | 114.2401 | 1 |
| Birth Year | 0.2032 | 1996 |
| Sex*Year | 0.03301 | 0 |
| Race*Year | -0.0547 | 1996 |
| Race*Sex | 1.2067 | 0 |
| Y-Intercept | -330.9723 | - |

| **Y Variables** | **Outputs** |
|---|---|
| ALE | 79.6738 |

| **Current Contributions** | **Funds** |
|---|---|
| Social Security | $0.00 |
| Savings | $2,000 |
| 401(k) | $0 |

| | Rates | |
|---|---|---|
| | Annual Inflation Rate | 2.00% |
| | Contribute to Social Security | 12.40% |
| | Savings Interest Rate | 0.09% |
| | Effective Social Security | 13.13% |
| | **From Inputs** | |
| | Current Age | 23 |
| | Years until Retiremnt | 44 |
| | Your 401(k) Contribution | 380 |
| | Employer 401(k) Contribution | 19 |
| | Social Security per year | $6,236.55 |
| | | |
| + | $883,767.95 | CoL Retirement to age 74 |
| + | $466,862.36 | COL age 74 to ALE |
| - | $764,534.20 | 401(k) Contributions |
| - | $433,456.87 | Social Security Contributions |
| = | $152,639.24 | Money needed at Retirement |
| | ($279.55) | PMT/MO. |

# References


[1] Angelo Canty and Brian Ripley, *Boot: Bootstrap Functions*: *(Originally by Angelo Canty for S)* (2019), https://CRAN.R-project.org/package=boot.

[2] Ann Foster, *Consumer Expenditures Vary by Age*, https://www.bls.gov/opub/btn/volume-4/pdf/consumer-expenditures-vary-by-age.pdf. Accessed April 29, 2019.

[3] Beth Atkinson and Terry Therneau, *Rpart: Recursive Partitioning and Regression Trees* (2019), https://CRAN.R-project.org/package=rpart.

[4] Carson Sievert, Chris Parmer, Toby Hocking, Scott Chamberlain, Karthik Ram, Marianne Corvellec, and Pedro Despony, *Plotly: Create Interactive Web Graphics via Plotly.js* (2019), https://CRAN.R-project.org/package=plotly.

[5] CDC and NCHS, *NCHS - Death Rates and Life Expectancy at Birth*, https://catalog.data.gov/dataset/age-adjusted-death-rates-and-life-expectancy-at-birth-all-races-both-sexes- Accessed April 29, 2019.

[6] Daniela Witten, Gareth James, Trevor Hastie, and Robert Tibshirani, *An Introduction to Statistical Learning*, Springer New York Heidelberg Dordrecht London, 2013.

[7] Hadley Wickham, Winston Chang, Lionel Henry, Thomas Lin Pedersen, Kohske Takahashi, Claus Wilke, and Kara Woo, *Ggplot2: Create Elegant Data Visualisations Using the Grammar of Graphics* (2019), https://CRAN.R-project.org/package=ggplot2.

[8] Hadley Wickham, Romain Francois, Lionel Henry, and Kirill Muller, *Dplyr: A Grammar of Data Manipulation* (2019), https://CRAN.R-project.org/package=dplyr.

[9] Lauren Perez, *What is the Average Interest Rate for Savings Accounts*, https://smartasset.com/checking-account/average-savings-account-interest. Accessed April 29, 2019.

[10] Steven Melendez, *How Much Social Security Tax Gets Taken Out of my Paycheck?*, https://pocketsense.com/much-social-security-tax-gets-taken-out-paycheck-2911.html. Accessed April 29, 2019.

[11] R Development Core Team, *R: A Language and Environment for Statistical Computing* (2006), http://www.R-project.org. ISBN 3-900051-07-0.

[12] Yihui Xie, *Knitr: A General-Purpose Package for Dynamic Report Generation in R* (2019), https://CRAN.R-project.org/package=knitr.