Consumer Behavior Data Analysis

Madison Wozniak

12/14/2021

I entered college wanting to study and eventually have a career in advertising, because I have always held an interest in people's behavior. This interest translated into advertising, when looking at consumer psychology, specifically why people purchase products based on the product itself, and the ads they come across. I decided having a career strictly based on advertising was not the path I wanted to take, but I still want to apply my new college major to the parts of consumer behavior that I am interested in. That is why I chose to analyze a data-set that provided demographic information on consumers as well as their purchasing behavior over the pan of two years. Many organizations use what is called "segmentation" to divide consumers into groups off four main categories: Demographics, Geographics, Psychographics, and Behavior. I want to use this data to test the degree to which demographic information is relevant to determining whether or not a consumer will buy your product.

The original structure of my data is as follows:

```r
library(tidyverse)
```

```
## -- Attaching packages --------------------------------------- tidyverse 1.3.1 --

## v ggplot2 3.3.5     v purrr   0.3.4
## v tibble  3.1.4     v dplyr   1.0.7
## v tidyr   1.1.3     v stringr 1.4.0
## v readr   2.0.1     v forcats 0.5.1

## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
```

```r
Marketing<-read_table("marketing_campaign.csv")
```

```
##
## -- Column specification --------------------------------------------------------
## cols(
##   .default = col_double(),
##   Education = col_character(),
##   Marital_Status = col_character(),
##   Income = col_character(),
##   Teenhome = col_character(),
##   Dt_Customer = col_character(),
##   Recency = col_character()
## )
## i Use `spec()` for the full column specifications.

## Warning: 221 parsing failures.
## row col   expected     actual                     file
##  11  -- 29 columns 28 columns 'marketing_campaign.csv'
##  20  -- 29 columns 30 columns 'marketing_campaign.csv'
##  28  -- 29 columns 28 columns 'marketing_campaign.csv'
##  38  -- 29 columns 30 columns 'marketing_campaign.csv'
##  44  -- 29 columns 28 columns 'marketing_campaign.csv'
## ... ... .......... .......... ......................
## See problems(...) for more details.
```

```
str(Marketing)
```

```
## spec_tbl_df [2,240 x 29] (S3: spec_tbl_df/tbl_df/tbl/data.frame)
##  $ ID                 : num [1:2240] 5524 2174 4141 6182 5324 ...
##  $ Year_Birth         : num [1:2240] 1957 1954 1965 1984 1981 ...
##  $ Education          : chr [1:2240] "Graduation" "Graduation" "Graduation" "Graduation" ...
##  $ Marital_Status     : chr [1:2240] "Single" "Single" "Together" "Together" ...
##  $ Income             : chr [1:2240] "58138" "46344" "71613" "26646" ...
##  $ Kidhome            : num [1:2240] 0 1 0 1 1 0 0 1 1 1 ...
##  $ Teenhome           : chr [1:2240] "0" "1" "0" "0" ...
##  $ Dt_Customer        : chr [1:2240] "04-09-2012" "08-03-2014" "21-08-2013" "10-02-2014" ...
##  $ Recency            : chr [1:2240] "58" "38" "26" "26" ...
##  $ MntWines           : num [1:2240] 635 11 426 11 173 520 235 76 14 28 ...
##  $ MntFruits          : num [1:2240] 88 1 49 4 43 42 65 10 0 0 ...
##  $ MntMeatProducts    : num [1:2240] 546 6 127 20 118 98 164 56 24 6 ...
##  $ MntFishProducts    : num [1:2240] 172 2 111 10 46 0 50 3 3 1 ...
##  $ MntSweetProducts   : num [1:2240] 88 1 21 3 27 42 49 1 3 1 ...
##  $ MntGoldProds       : num [1:2240] 88 6 42 5 15 14 27 23 2 13 ...
##  $ NumDealsPurchases  : num [1:2240] 3 2 1 2 5 2 4 2 1 1 ...
##  $ NumWebPurchases    : num [1:2240] 8 1 8 2 5 6 7 4 3 1 ...
##  $ NumCatalogPurchases: num [1:2240] 10 1 2 0 3 4 3 0 0 0 ...
##  $ NumStorePurchases  : num [1:2240] 4 2 10 4 6 10 7 4 2 0 ...
##  $ NumWebVisitsMonth  : num [1:2240] 7 5 4 6 5 6 6 8 9 20 ...
##  $ AcceptedCmp3       : num [1:2240] 0 0 0 0 0 0 0 0 0 1 ...
##  $ AcceptedCmp4       : num [1:2240] 0 0 0 0 0 0 0 0 0 0 ...
##  $ AcceptedCmp5       : num [1:2240] 0 0 0 0 0 0 0 0 0 0 ...
##  $ AcceptedCmp1       : num [1:2240] 0 0 0 0 0 0 0 0 0 0 ...
##  $ AcceptedCmp2       : num [1:2240] 0 0 0 0 0 0 0 0 0 0 ...
##  $ Complain           : num [1:2240] 0 0 0 0 0 0 0 0 0 0 ...
##  $ Z_CostContact      : num [1:2240] 3 3 3 3 3 3 3 3 3 3 ...
##  $ Z_Revenue          : num [1:2240] 11 11 11 11 11 11 11 11 11 11 ...
##  $ Response           : num [1:2240] 1 0 0 0 0 0 0 0 1 0 ...
##  - attr(*, "problems")= tibble [221 x 5] (S3: tbl_df/tbl/data.frame)
##   ..$ row     : int [1:221] 11 20 28 38 44 47 49 59 68 79 ...
##   ..$ col     : chr [1:221] NA NA NA NA ...
##   ..$ expected: chr [1:221] "29 columns" "29 columns" "29 columns" "29 columns" ...
##   ..$ actual  : chr [1:221] "28 columns" "30 columns" "28 columns" "30 columns" ...
##   ..$ file    : chr [1:221] "'marketing_campaign.csv'" "'marketing_campaign.csv'" "'marketing_campaig
##  - attr(*, "spec")=
##   .. cols(
##   ..   ID = col_double(),
##   ..   Year_Birth = col_double(),
##   ..   Education = col_character(),
##   ..   Marital_Status = col_character(),
##   ..   Income = col_character(),
##   ..   Kidhome = col_double(),
##   ..   Teenhome = col_character(),
##   ..   Dt_Customer = col_character(),
##   ..   Recency = col_character(),
##   ..   MntWines = col_double(),
##   ..   MntFruits = col_double(),
##   ..   MntMeatProducts = col_double(),
##   ..   MntFishProducts = col_double(),
##   ..   MntSweetProducts = col_double(),
##   ..   MntGoldProds = col_double(),
##   ..   NumDealsPurchases = col_double(),
##   ..   NumWebPurchases = col_double(),
##   ..   NumCatalogPurchases = col_double(),
##   ..   NumStorePurchases = col_double(),
##   ..   NumWebVisitsMonth = col_double(),
##   ..   AcceptedCmp3 = col_double(),
##   ..   AcceptedCmp4 = col_double(),
##   ..   AcceptedCmp5 = col_double(),
##   ..   AcceptedCmp1 = col_double(),
##   ..   AcceptedCmp2 = col_double(),
##   ..   Complain = col_double(),
##   ..   Z_CostContact = col_double(),
##   ..   Z_Revenue = col_double(),
##   ..   Response = col_double()
##   .. )
```

In order to use the data to test my prediction, I needed to manipulate certain variables to clean up the data.

First, I renamed some variables so they would be easier to quickly read

```
marketing<-rename(Marketing, "wine"="MntWines","fruits"="MntFruits","meat"="MntMeatProducts",
                  "fish"="MntFishProducts","sweets"="MntSweetProducts","gold"="MntGoldProds")
```

Second, I changed the class of the 'Income' variable from character to numeric so I could use the values in my calculations and plots later on.

```
marketing$Income<-as.numeric(as.character(marketing$Income))
```

```
## Warning: NAs introduced by coercion
```

```
sapply(marketing,class)
```

```
##                  ID          Year_Birth           Education      Marital_Status
##           "numeric"           "numeric"         "character"         "character"
##              Income             Kidhome            Teenhome         Dt_Customer
##           "numeric"           "numeric"         "character"         "character"
##             Recency                wine              fruits                meat
##         "character"           "numeric"           "numeric"           "numeric"
##                fish              sweets                gold   NumDealsPurchases
##           "numeric"           "numeric"           "numeric"           "numeric"
##     NumWebPurchases NumCatalogPurchases   NumStorePurchases   NumWebVisitsMonth
##           "numeric"           "numeric"           "numeric"           "numeric"
##        AcceptedCmp3        AcceptedCmp4        AcceptedCmp5        AcceptedCmp1
##           "numeric"           "numeric"           "numeric"           "numeric"
##        AcceptedCmp2            Complain       Z_CostContact           Z_Revenue
##           "numeric"           "numeric"           "numeric"           "numeric"
##            Response
##           "numeric"
```

```
marketing<-marketing%>%
  rename(income=Income)
```

```
str(marketing$income)
```

```
##  num [1:2240] 58138 46344 71613 26646 58293 ...
```

Third, I removed the columns that were not necessary to the goal of this analysis, and selected only the observations I need.

```
names(marketing)
```

```
##  [1] "ID"                  "Year_Birth"          "Education"
##  [4] "Marital_Status"      "income"              "Kidhome"
##  [7] "Teenhome"            "Dt_Customer"         "Recency"
## [10] "wine"                "fruits"              "meat"
## [13] "fish"                "sweets"              "gold"
## [16] "NumDealsPurchases"   "NumWebPurchases"     "NumCatalogPurchases"
## [19] "NumStorePurchases"   "NumWebVisitsMonth"   "AcceptedCmp3"
## [22] "AcceptedCmp4"        "AcceptedCmp5"        "AcceptedCmp1"
## [25] "AcceptedCmp2"        "Complain"            "Z_CostContact"
## [28] "Z_Revenue"           "Response"
```

```
marketing<-marketing%>%
  select(
    Year_Birth,Education,Marital_Status,Kidhome,Teenhome,
    wine,fruits,meat,fish,sweets,gold,income
  )
```

Finally, I changed the 'Year_Birth' variable to 'age' which represented the age of the participant as of 2021 to make calculations simpler. I also changed the 'Education' observations to factors with levels that represent the different levels of education that participants have completed.

```
marketing<-marketing%>%
  mutate(age=2021-Year_Birth)%>%
  select(-Year_Birth)

str(marketing$Education)
```

```
##  chr [1:2240] "Graduation" "Graduation" "Graduation" "Graduation" "PhD" ...
```

```
new_education<-c("Graduation","PhD","Master","Basic","2n")
class(new_education)
```

```
## [1] "character"
```

```
sort(new_education)
```

```
## [1] "2n"         "Basic"      "Graduation" "Master"     "PhD"
```

```
factor_education<-factor(new_education,levels=c("2nd","Basic","Graduation",
                                                "Master","PhD"))
factor_education
```

```
## [1] Graduation PhD        Master     Basic      <NA>
## Levels: 2nd Basic Graduation Master PhD
```

```
class(factor_education)
```

```
## [1] "factor"
```

```
sort(factor_education)
```

```
## [1] Basic      Graduation Master     PhD
## Levels: 2nd Basic Graduation Master PhD
```

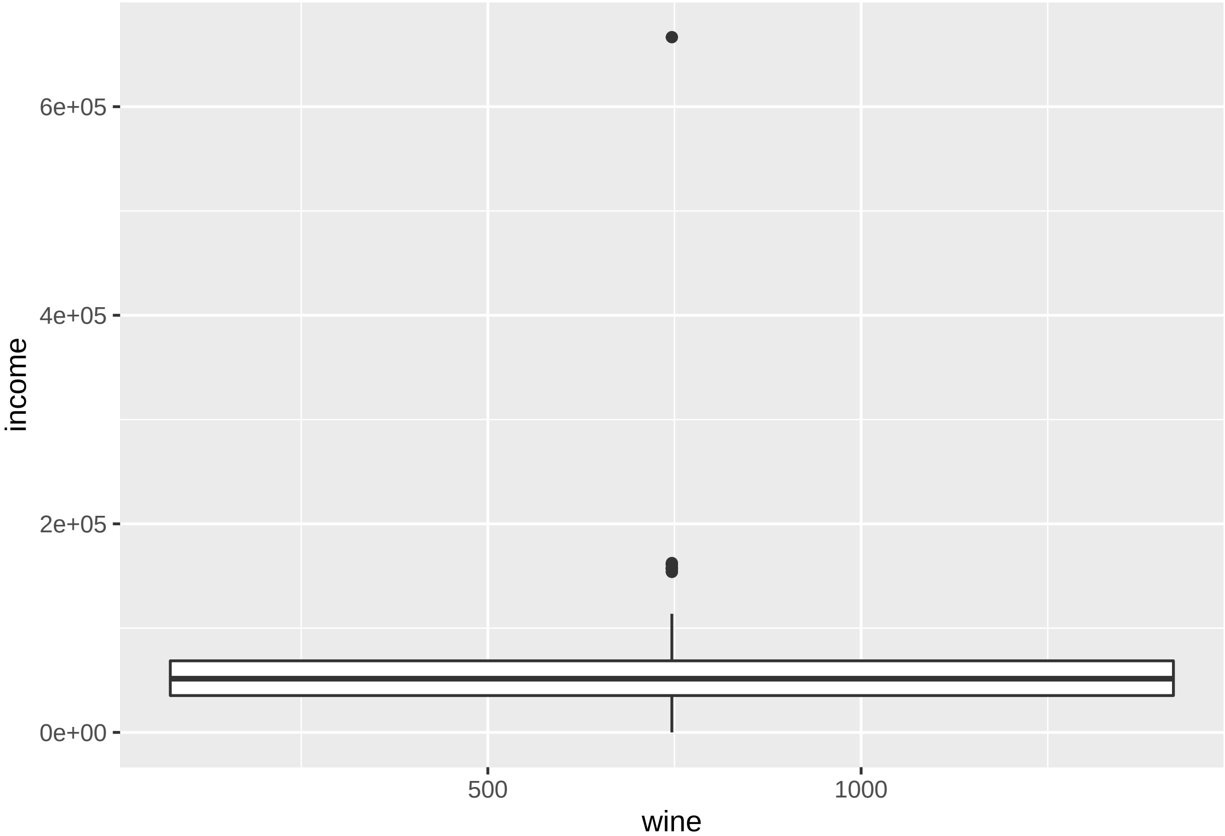
Now that all necessary manipulations to the data have been made, I can start the process of solving the key issue I have chosen to explore for this analysis. In order to answer whether or not demographic information is relevant to purchasing decisions, I wanted to start with the demographic variable 'income' and compare that to each item (wine, gold, meat, fruits, fish, and sweets) to visualize any potential relationships.

First, I focused on income and wine. I created boxplots that would tell me what outliers each variable contained.

```
wine_income_boxplot<-marketing%>%
  ggplot(aes(wine,income))+
  geom_boxplot()
wine_income_boxplot
```

```
## Warning: Continuous x aesthetic -- did you forget aes(group=...)?
```

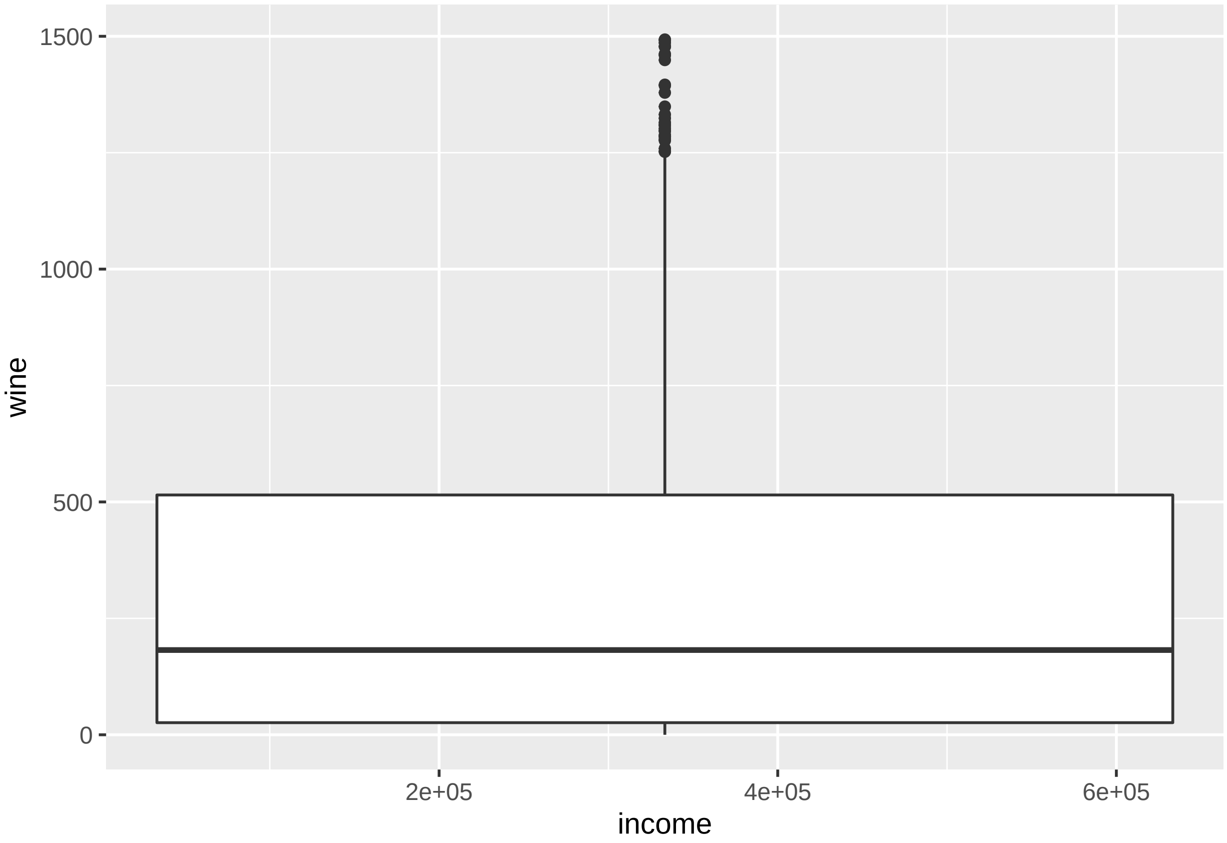
```
## Warning: Removed 203 rows containing non-finite values (stat_boxplot).
```



```
income_wine_boxplot<-marketing%>%
  ggplot(aes(income,wine))+
  geom_boxplot()
income_wine_boxplot
```

```
## Warning: Continuous x aesthetic -- did you forget aes(group=...)?
```

```
## Warning: Removed 203 rows containing missing values (stat_boxplot).
```

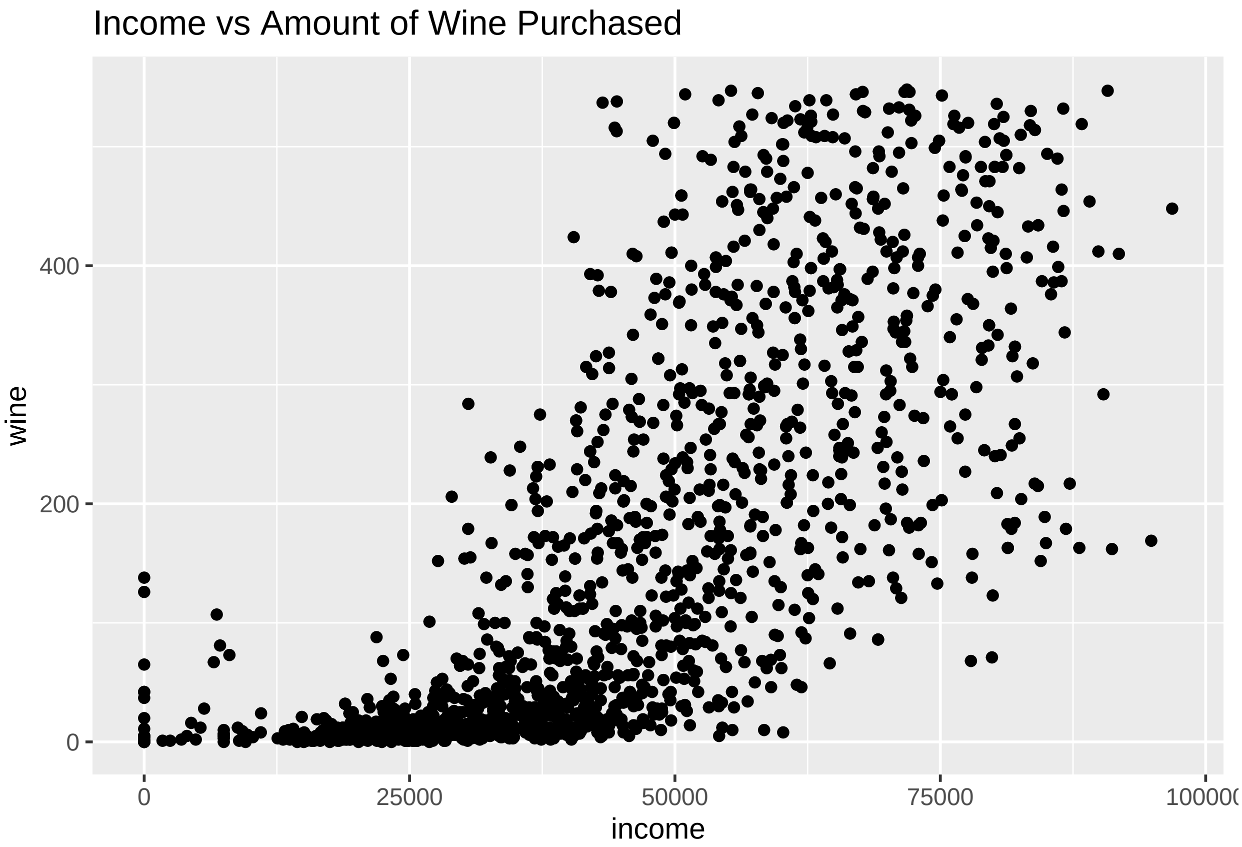


I can use this information to filter out the outliers that were shown.

```
wine_filter1<-marketing%>%
  filter(wine<550,income<1e+05)
```

With the outliers filtered, I can create a simple scatterplot that will give me an idea of what type of relationship the two variables might have with one another.

```
income_wine_plain<-wine_filter1%>%
  ggplot(aes(income,wine))+
  geom_point()+
  ggtitle("Income vs Amount of Wine Purchased")
income_wine_plain
```



Now, I can add a linear model on top of the data and report the variables' correlation and a summary of the linear model.

```
cor(wine_filter1$income,wine_filter1$wine)
```

```
## [1] 0.7429525
```

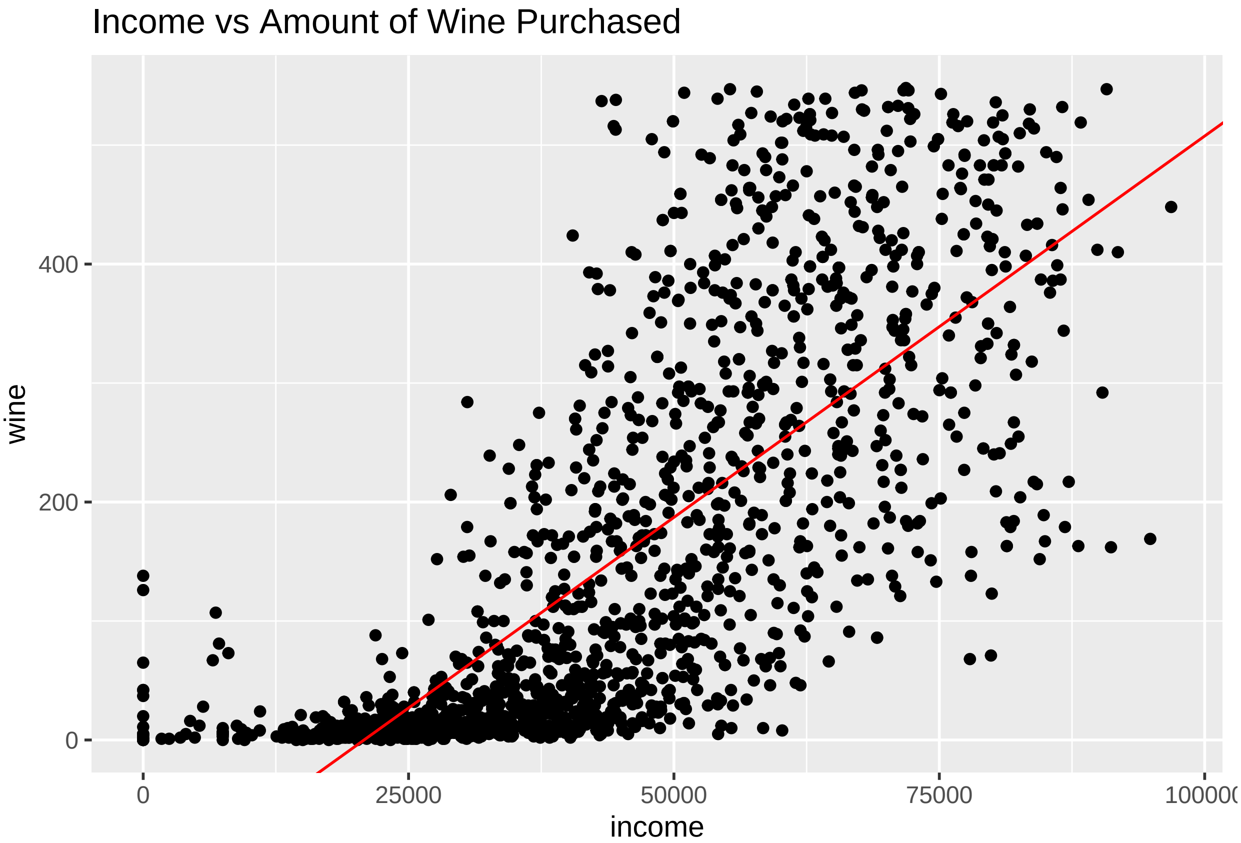
```
lm_wine<-lm(wine~income,wine_filter1)
lm_wine
```

```
##
## Call:
## lm(formula = wine ~ income, data = wine_filter1)
##
## Coefficients:
## (Intercept)       income
##  -133.35789      0.00641
```

```
summary(lm_wine)
```

```
##
## Call:
## lm(formula = wine ~ income, data = wine_filter1)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -307.56  -74.40  -17.04   61.75  393.55
##
## Coefficients:
##                Estimate Std. Error t value Pr(>|t|)
## (Intercept) -1.334e+02  7.173e+00  -18.59   <2e-16 ***
## income       6.410e-03  1.459e-04   43.94   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 110.7 on 1567 degrees of freedom
## Multiple R-squared:  0.552,  Adjusted R-squared:  0.5517
## F-statistic:  1931 on 1 and 1567 DF,  p-value: < 2.2e-16
```

```
income_wine<-wine_filter1%>%
  ggplot(aes(income,wine))+
  geom_point()+
  ggtitle("Income vs Amount of Wine Purchased")+
  geom_abline(slope=0.00641,intercept=-133.35789,color="red")
income_wine
```



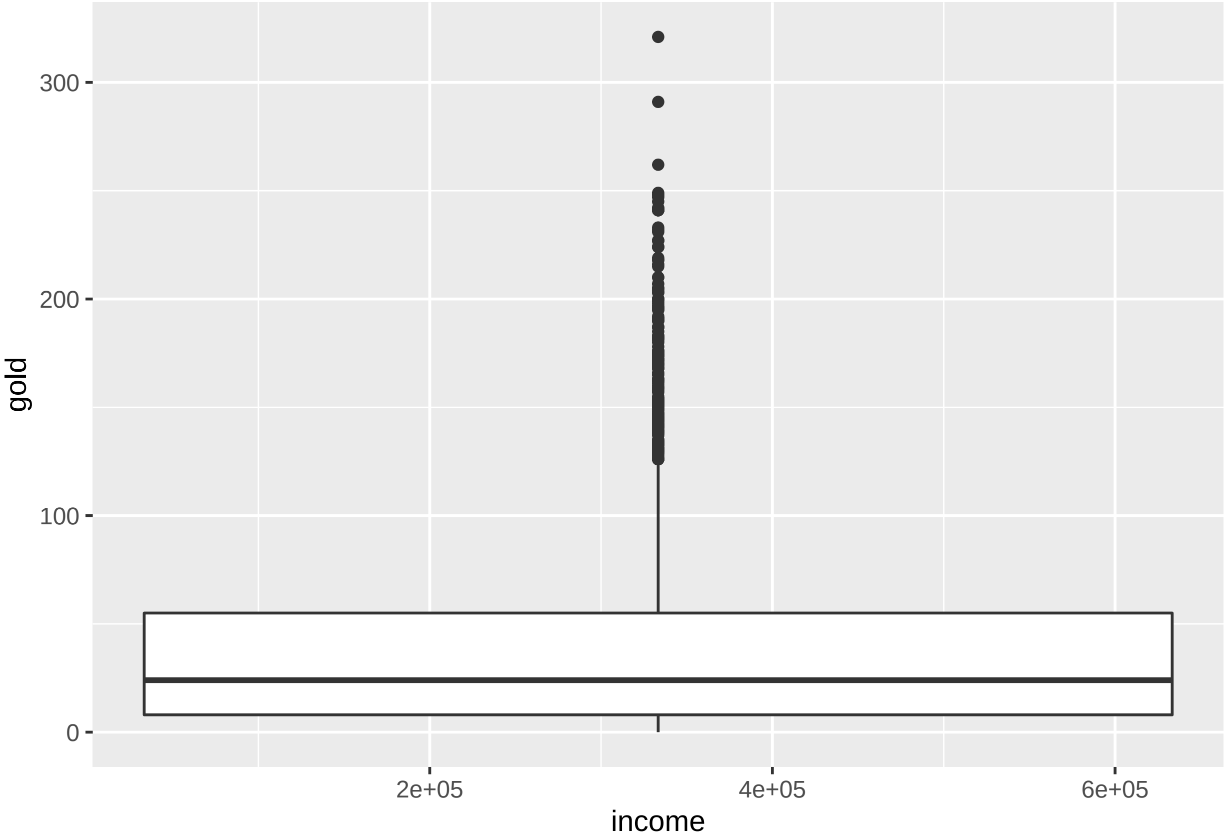
The summary of the linear model is valuable because it provides a Multiple R-squared value = 0.552 which tells me the data has a moderate positive correlation, and additionally the asterisks next to the Pr(>|t|) also known as the ‘p’ values, are a group of three. Generally, the more asterisks there are, the more significant our values are, so this means ‘income’ is a significant value to test against wine.

This process will be repeated for each of the items.

Finding the outliers and removing them for gold. Since I found teh outliers for the ‘income’ variable, that process does not need to be repeated, and the values found will carry-over into each model.

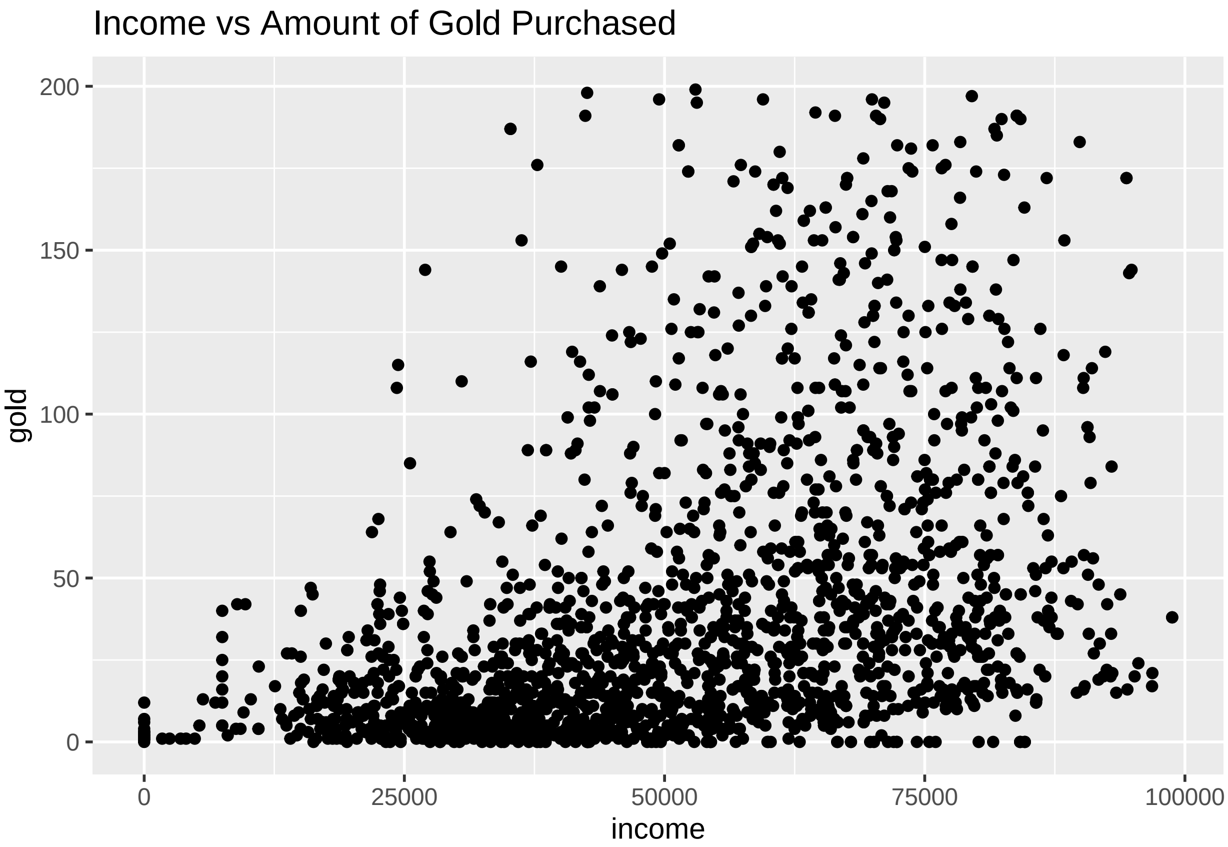
```
gold_boxplot<-marketing%>%
  ggplot(aes(income,gold))+
  geom_boxplot()
gold_boxplot
```

```
## Warning: Continuous x aesthetic -- did you forget aes(group=...)?

## Warning: Removed 203 rows containing missing values (stat_boxplot).
```



```
gold_filter<-marketing%>%
  filter(gold<200,income<1e+05)
```

```
income_gold_plain<-gold_filter%>%
  ggplot(aes(income,gold))+
  geom_point()+
ggtitle("Income vs Amount of Gold Purchased")
income_gold_plain
```



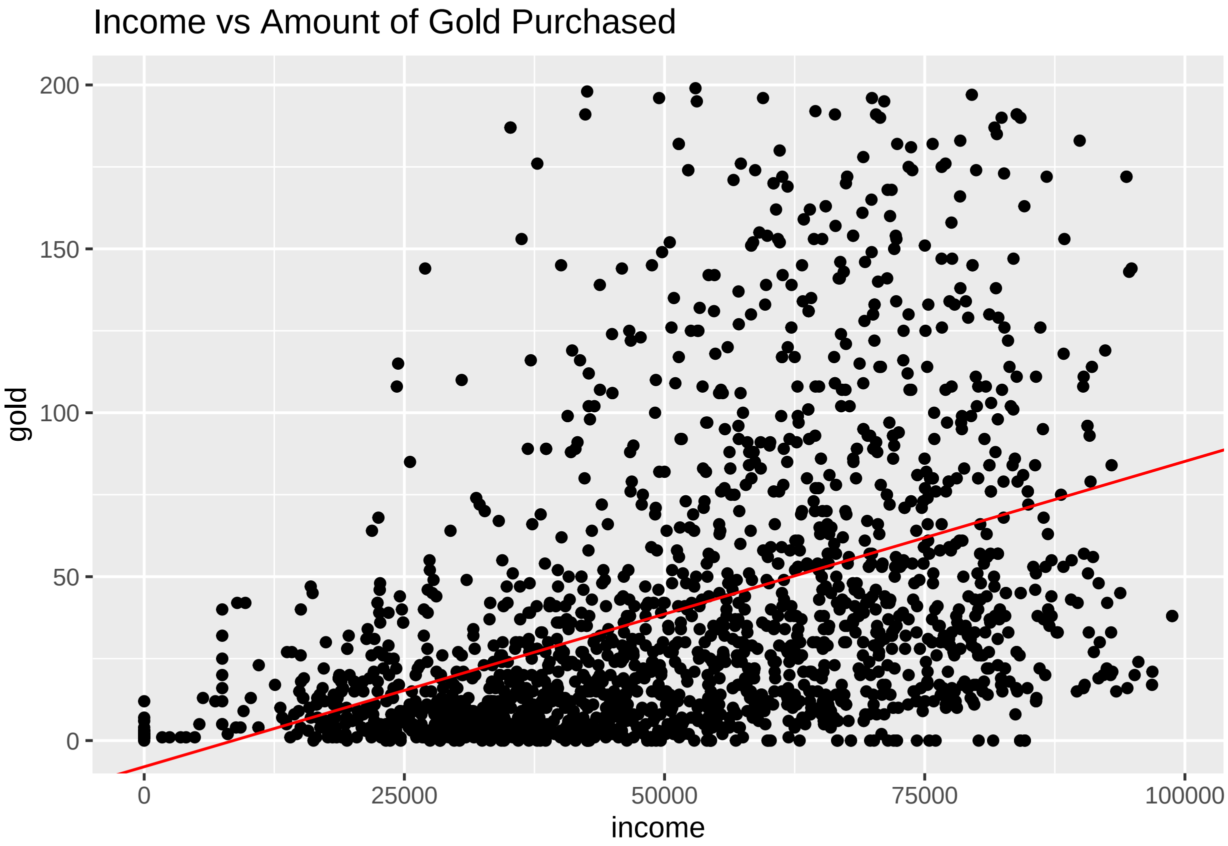
```
cor(gold_filter$income,gold_filter$gold)
```

```
## [1] 0.4351882
```

```
lm_gold<-lm(gold~income,gold_filter)
summary(lm_gold)
```

```
##
## Call:
## lm(formula = gold ~ income, data = gold_filter)
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -70.84 -24.96 -10.13  11.29 166.33
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept) -7.983e+00  2.385e+00  -3.347 0.000833 ***
## income       9.315e-04  4.328e-05  21.524  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 40.14 on 1983 degrees of freedom
## Multiple R-squared:  0.1894, Adjusted R-squared:  0.189
## F-statistic: 463.3 on 1 and 1983 DF,  p-value: < 2.2e-16
```

```
income_gold<-gold_filter%>%
  ggplot(aes(income,gold))+
  geom_point()+
  ggtitle("Income vs Amount of Gold Purchased")+
  geom_abline(slope=0.0009315,intercept=-7.9834186,color="red")
income_gold
```

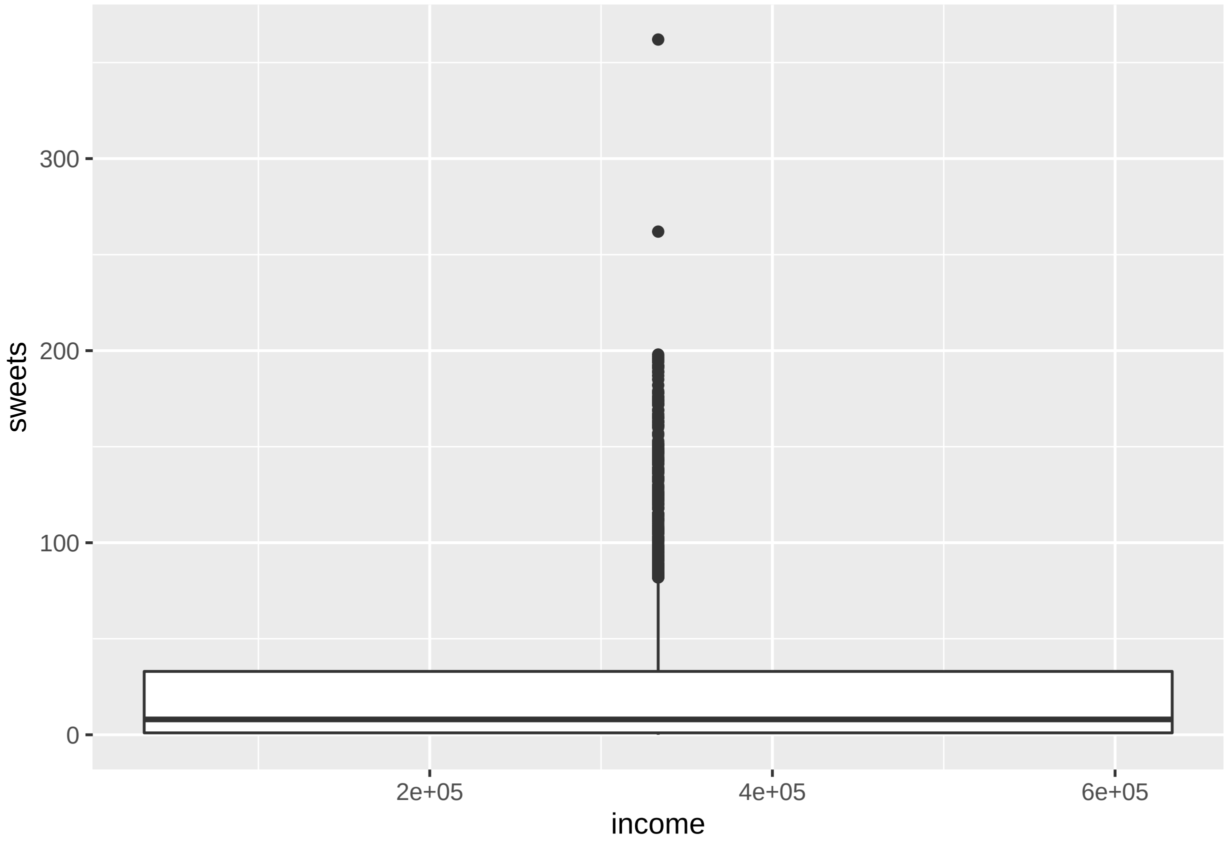


The summary for the linear regression model of ‘income’ and ‘gold’ tells us there is a Multiple R-squared value of 0.1894 which is very low, meaning there is a weak but positive relationship between the data. However, ‘income’ has three asterisks for its p value again, which indicated it is a good variable to use, and we potentially should use a nonlinear model to make a prediction.

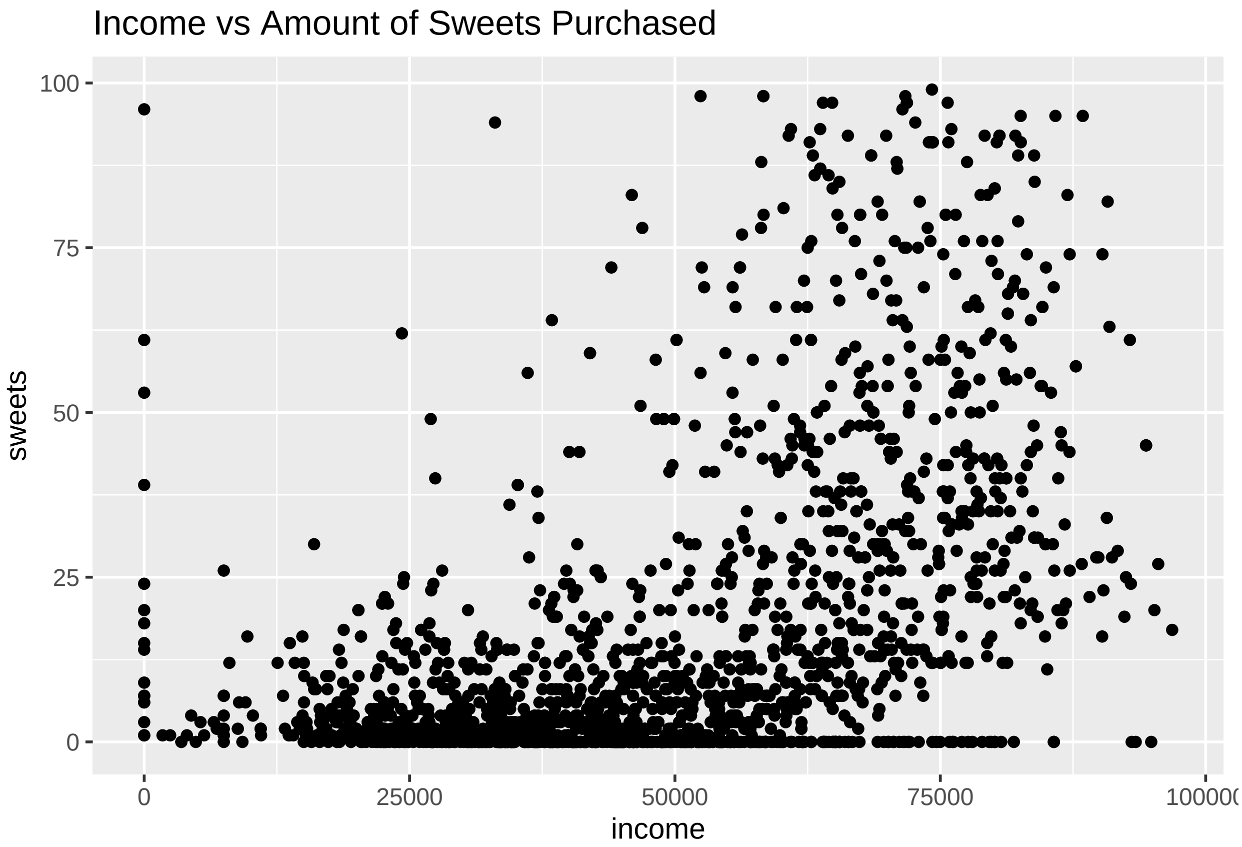
For sweets I took the same steps to find and filter outliers, plotted the data as a scatterplot, then found the correlation coefficient and linear model, which was then plotted on the original scatterplot.

```
sweets_boxplot<-marketing%>%
  ggplot(aes(income,sweets))+
  geom_boxplot()
sweets_boxplot
```

```
## Warning: Continuous x aesthetic -- did you forget aes(group=...)?

## Warning: Removed 203 rows containing missing values (stat_boxplot).
```



```
sweets_filter<-marketing%>%
  filter(sweets<100,income<1e+05)

income_sweets_plain<-sweets_filter%>%
  ggplot(aes(income,sweets))+
  geom_point()+
ggtitle("Income vs Amount of Sweets Purchased")
income_sweets_plain
```



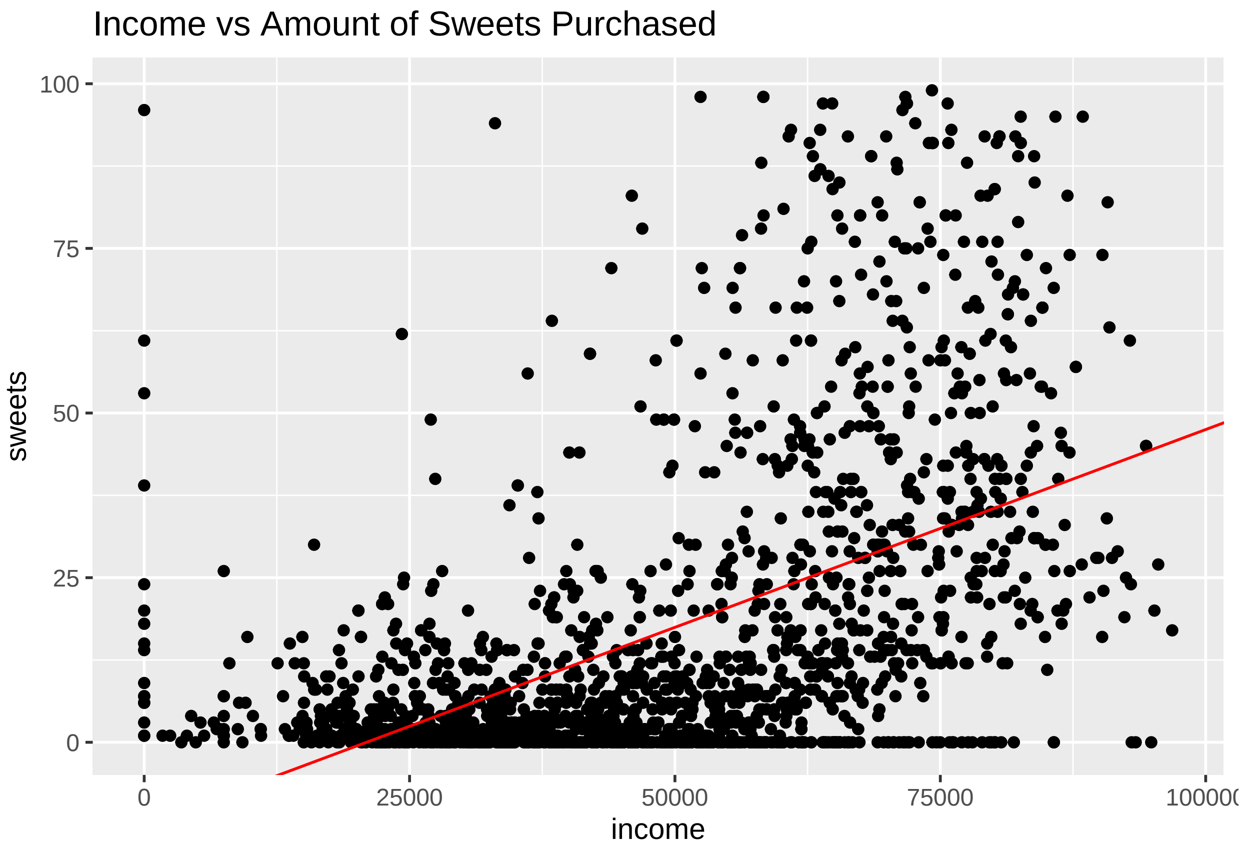
```
cor(sweets_filter$income,sweets_filter$sweets)
```

```
## [1] 0.5210026
```

```
lm_sweets<-lm(sweets~income,sweets_filter)
summary(lm_sweets)
```

```
##
## Call:
## lm(formula = sweets ~ income, data = sweets_filter)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -44.412 -12.925  -4.681   6.456 108.578
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept) -1.258e+01  1.217e+00  -10.34   <2e-16 ***
## income       6.007e-04  2.275e-05   26.40   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 19.85 on 1871 degrees of freedom
## Multiple R-squared:  0.2714, Adjusted R-squared:  0.2711
## F-statistic: 697.1 on 1 and 1871 DF,  p-value: < 2.2e-16
```

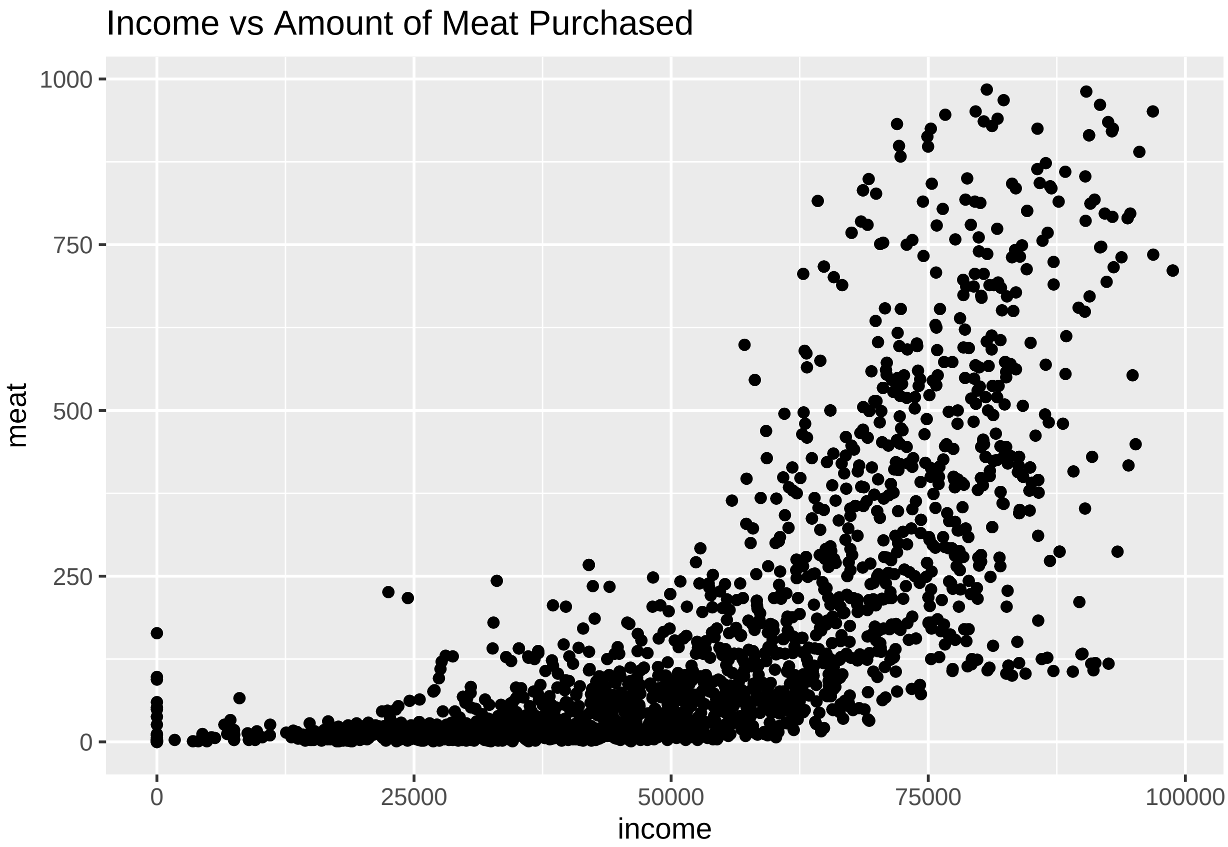
```
income_sweets<-sweets_filter%>%
  ggplot(aes(income,sweets))+
  geom_point()+
  ggtitle("Income vs Amount of Sweets Purchased")+
  geom_abline(slope=6.007e-04,intercept=-1.258e+01,color="red")
income_sweets
```



The summary of the sweets linear model tells us it has a Multiple R-squared of 0.2714 and income's p value still have three asterisks next to it. The last thing I want to note is the correlation coefficient of sweets and income, which came out to 0.52, meaning there is a very moderate correlation between the variables.

The next variable I am looking at is amount spent on meats vs income. I took the same steps as in the prior variables.

```
meat_boxplot<-marketing%>%
  ggplot(aes(income,meat))+
  geom_boxplot()

meat_filter<-marketing%>%
  filter(meat<1500,income<1e+05)

income_meat_plain<-meat_filter%>%
  ggplot(aes(income,meat))+
  geom_point()+
ggtitle("Income vs Amount of Meat Purchased")
income_meat_plain
```



```
cor(meat_filter$income,meat_filter$meat)
```

```
## [1] 0.7193946
```

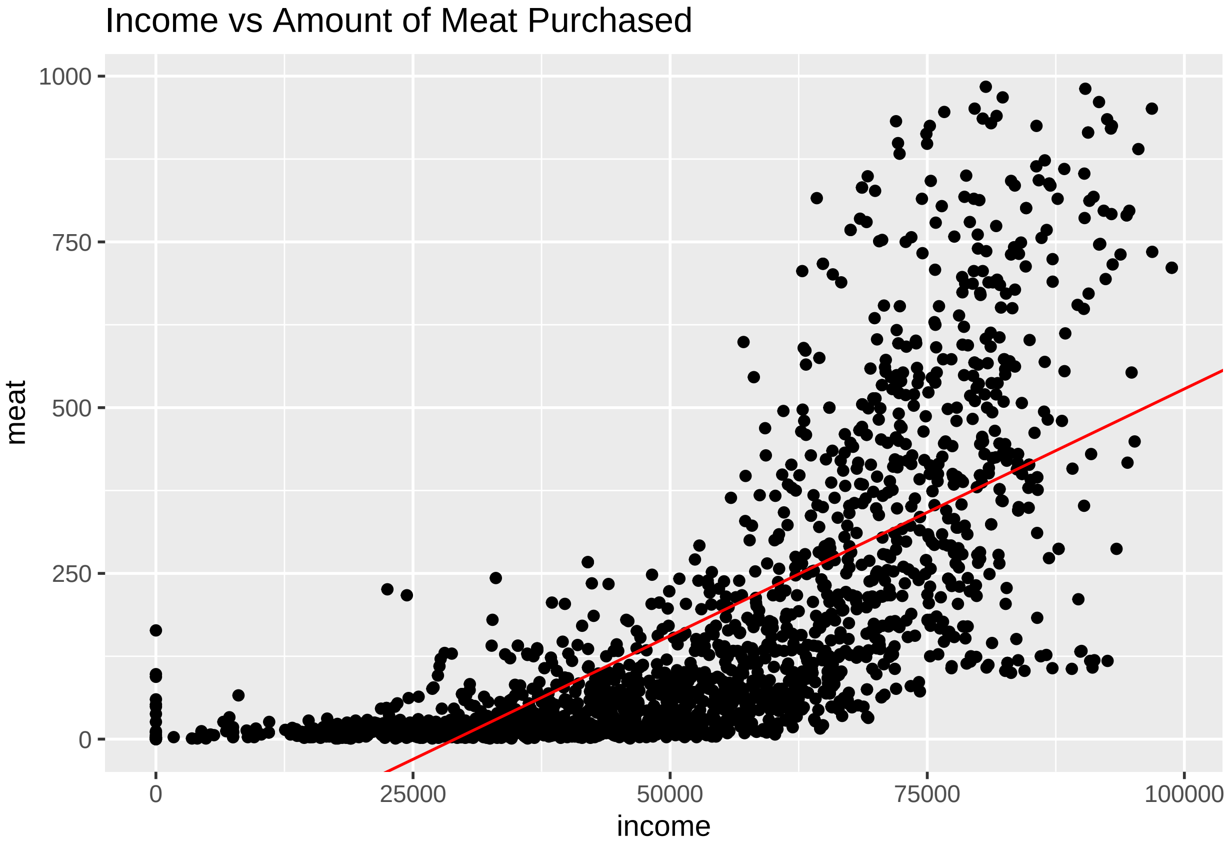
The correlation coefficient of the two variables is pretty high at 0.719 so with that information, we know they are pretty relevant to one another for a linear regression model to be applied.

Next I will find the linear model for this relationship and plot it on the scatterplot created above.

```
lm_meat<-lm(meat~income,meat_filter)
summary(lm_meat)
```

```
##
## Call:
## lm(formula = meat ~ income, data = meat_filter)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -354.69  -97.37  -23.71   65.05  612.53
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept) -2.166e+02  8.876e+00  -24.40   <2e-16 ***
## income       7.449e-03  1.600e-04   46.56   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 150.5 on 2021 degrees of freedom
## Multiple R-squared:  0.5175, Adjusted R-squared:  0.5173
## F-statistic:  2168 on 1 and 2021 DF,  p-value: < 2.2e-16
```

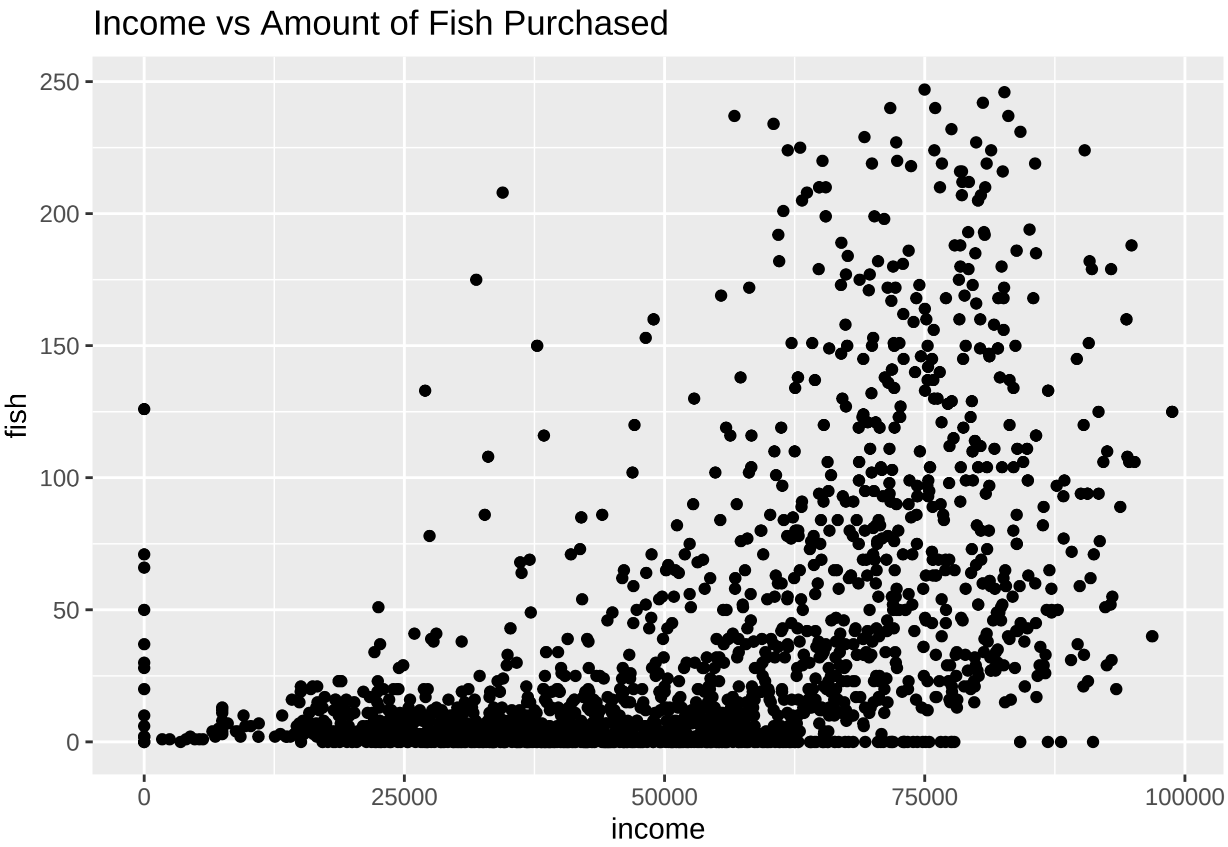
```
income_meat<-meat_filter%>%
  filter(meat<1500,income<1e+05)%>%
  ggplot(aes(income,meat))+
  geom_point()+
  ggtitle("Income vs Amount of Meat Purchased")+
  geom_abline(slope=7.449e-03,intercept=-2.166e+02,color="red")
income_meat
```



What can be recorded from the summary of the linear model, are the R-squared value and p value. The multiple R-square value is 0.51 which is very moderate, but the income p value has three astericks so we know it is a relevant value.

The next product I looked at was fish.

```
fish_boxplot<-marketing%>%
  ggplot(aes(income,fish))+
  geom_boxplot()

fish_filter<-marketing%>%
  filter(fish<250,income<1e+05)

income_fish_plain<-fish_filter%>%
  ggplot(aes(income,fish))+
  geom_point()+
 ggtitle("Income vs Amount of Fish Purchased")
income_fish_plain
```



```
cor(fish_filter$income,fish_filter$fish)
```

```
## [1] 0.5426942
```

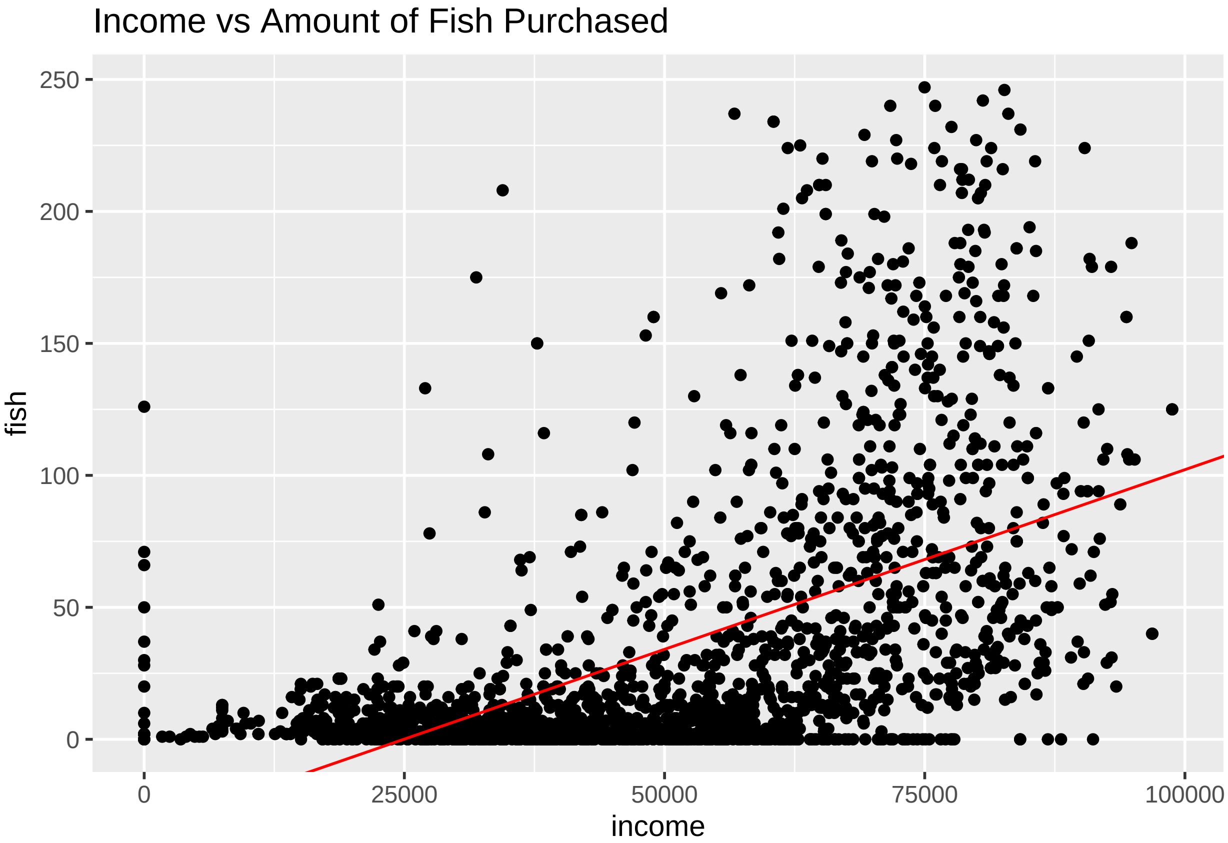
After plotting and filtering out the outliers, I calculated a correlation coefficient of 0.54. This is a bit weaker than the meats correlation, but it still signifies a moderately important relationship.

Next I applied a linear model and plotted that on the scatterplot.

```
lm_fish<-lm(fish~income,fish_filter)
summary(lm_fish)
```

```
##
## Call:
## lm(formula = fish ~ income, data = fish_filter)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -90.140 -27.815  -9.621  14.638 195.068
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept) -3.395e+01  2.599e+00  -13.06   <2e-16 ***
## income       1.361e-03  4.693e-05   29.00   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 44 on 2015 degrees of freedom
## Multiple R-squared:  0.2945, Adjusted R-squared:  0.2942
## F-statistic: 841.2 on 1 and 2015 DF,  p-value: < 2.2e-16
```

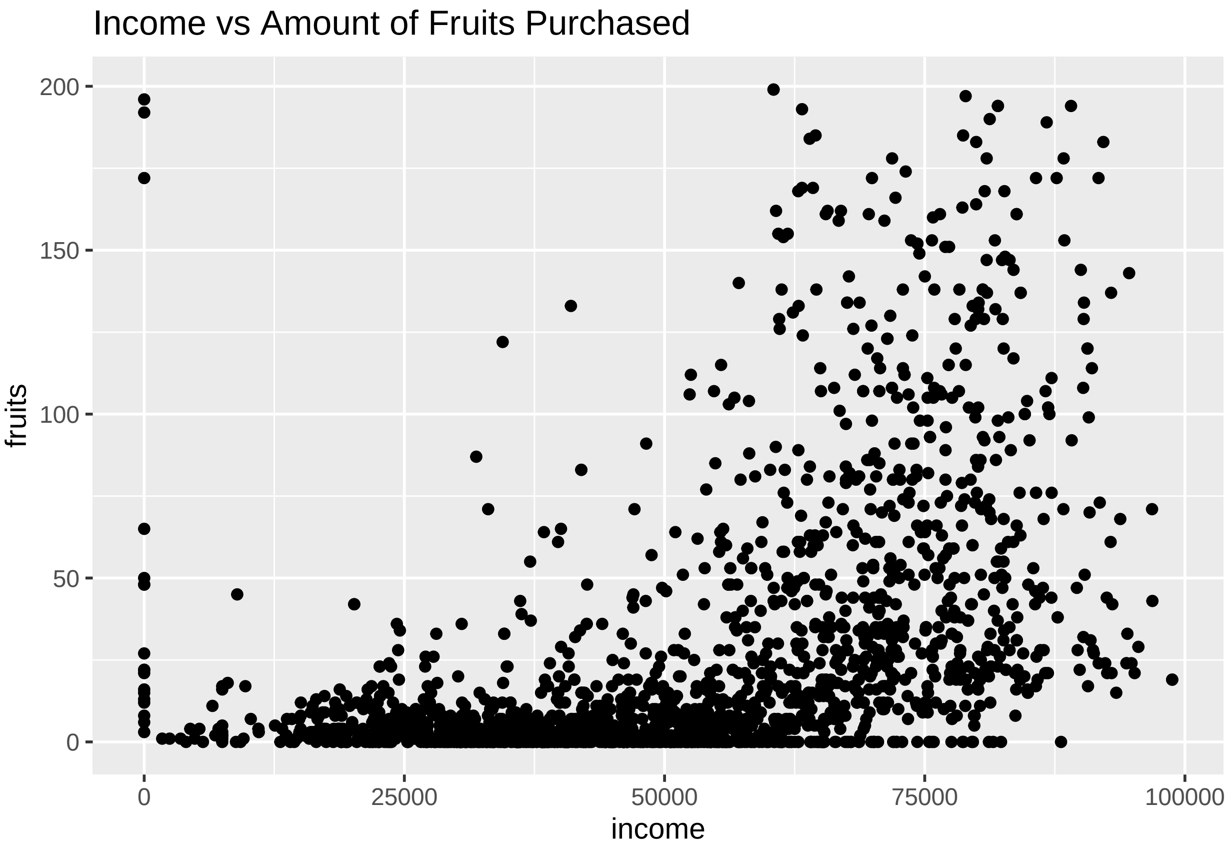
```
income_fish<-fish_filter%>%
  ggplot(aes(income,fish))+
  geom_point()+
  ggtitle("Income vs Amount of Fish Purchased")+
  geom_abline(slope=0.001361,intercept=-33.948985,color="red")
income_fish
```



The summary of the linear model indicates a strong p value, but a very weak multiple R-square value, which may improve with a nonlinear model applied to the data.

The last variable I looked at which follows the same steps as the previous five item are fruits.

```
fruit_boxplot<-marketing%>%
  ggplot(aes(income,fruits))+
  geom_boxplot()

fruits_filter<-marketing%>%
  filter(fruits<300,income<1e+05)

income_fruit_plain<-fruits_filter%>%
  ggplot(aes(income,fruits))+
  geom_point()+
ggtitle("Income vs Amount of Fruits Purchased")
income_fruit_plain
```



```
cor(fruits_filter$income,fruits_filter$fruits)
```

```
## [1] 0.508888
```

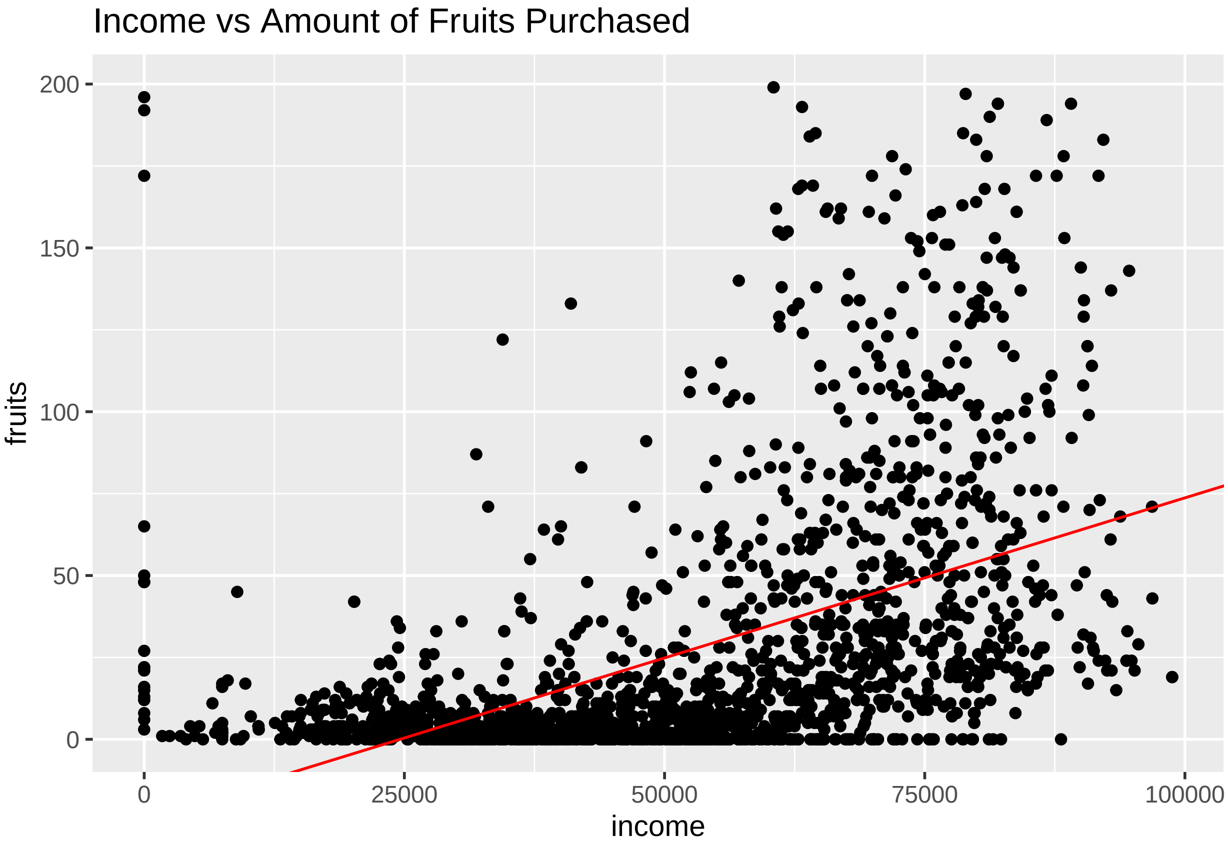
The correlation coefficient of these variables is also moderate (0.508).

Applying a linear regression model below, will return a multiple R-square value of 0.26 which is very low, but like the last variable, the p value is significant so the R value may improve with a nonlinear model.

```
lm_fruit<-lm(fruits~income,fruits_filter)
summary(lm_fruit)
```

```
## 
## Call:
## lm(formula = fruits ~ income, data = fruits_filter)
## 
## Residuals:
##     Min      1Q  Median      3Q     Max 
## -62.069 -20.769  -7.976   8.708 220.014 
## 
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)    
## (Intercept) -2.401e+01  2.042e+00  -11.76   <2e-16 ***
## income       9.771e-04  3.679e-05   26.56   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
## 
## Residual standard error: 34.5 on 2019 degrees of freedom
## Multiple R-squared:  0.259,  Adjusted R-squared:  0.2586
## F-statistic: 705.6 on 1 and 2019 DF,  p-value: < 2.2e-16
```

```
income_fruit<-fruits_filter%>%
  ggplot(aes(income,fruits))+
  geom_point()+
  ggtitle("Income vs Amount of Fruits Purchased")+
  geom_abline(slope=9.771e-04,intercept=-2.401e+01,color="red")
income_fruit
```



After conducting these linear regression models and applying them on my scatterplots, I realized adding another demographic factor to my models would increase the accuracy of my models, so for each variable, I filtered the level of education each participant has received. I chose 'Graduation' as the level of education.

The first step I took for each model was filtering out only the education variables classified as "Graduation". Below is an example of the filter:

```
marketing_education_filter1<-wine_filter1%>%
  filter(Education=="Graduation")
```

Next, each of the plots and new linear models will be included and explained.

```
cor(marketing_education_filter1$income,marketing_education_filter1$wine)
```

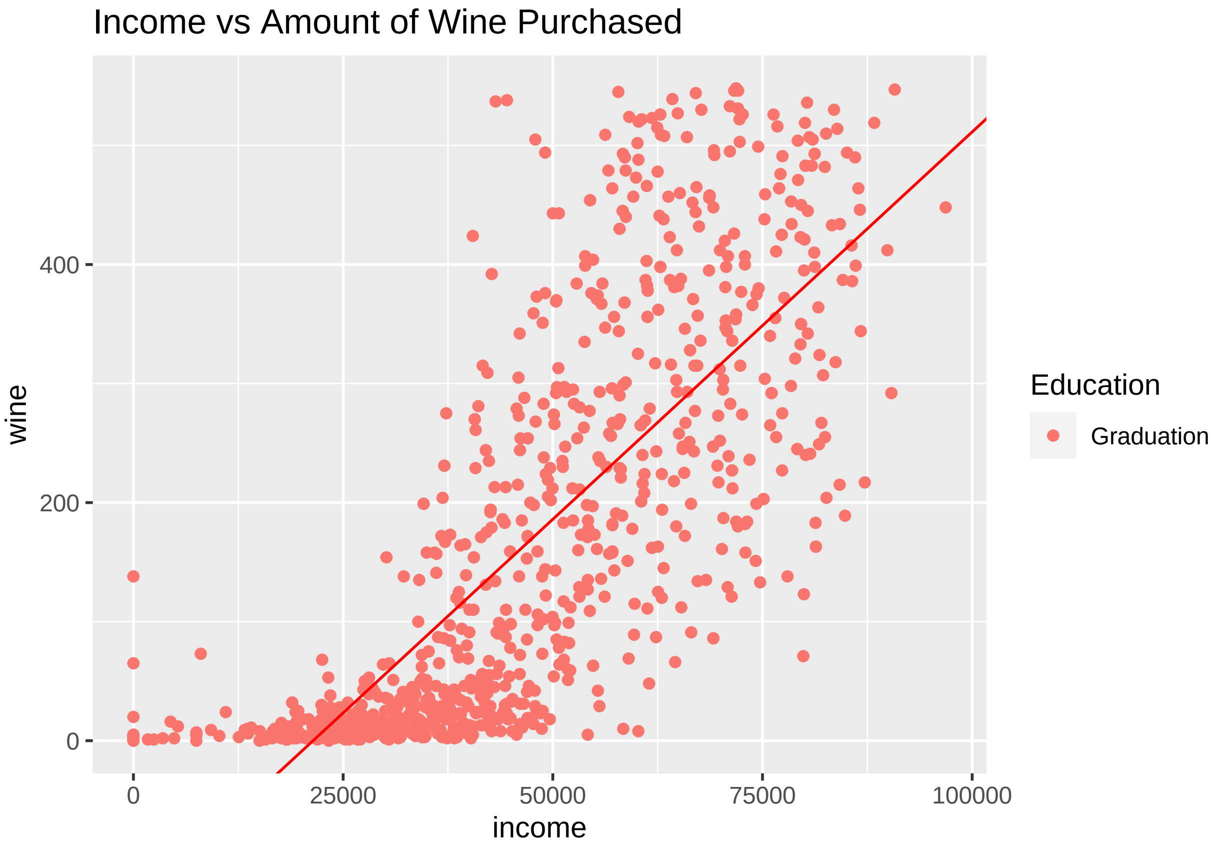
```
## [1] 0.7753248
```

```
 summary(lm(wine~income,marketing_education_filter1))
```

```
##
## Call:
## lm(formula = wine ~ income, data = marketing_education_filter1)
##
```

```
## Residuals:
##     Min      1Q  Median      3Q     Max 
## -309.34  -71.92  -15.09   58.57  395.37 
## 
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)    
## (Intercept) -1.394e+02  8.866e+00  -15.72   <2e-16 ***
## income       6.508e-03  1.755e-04   37.07   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
## 
## Residual standard error: 105.9 on 912 degrees of freedom
## Multiple R-squared:  0.6011, Adjusted R-squared:  0.6007 
## F-statistic:  1374 on 1 and 912 DF,  p-value: < 2.2e-16
```

```
income_wine2<-marketing_education_filter1%>%
  ggplot(aes(income,wine,color=Education))+
  geom_point()+
  ggtitle("Income vs Amount of Wine Purchased")+
  geom_abline(slope=6.508e-03,intercept=-1.394e+02,color="red")
income_wine2
```



There is a moderately high positive correlation between the data (0.77), a 0.60 multiple R-square value, and the income's p value is very significant.

Next, I will analyze the relationship between gold, wine, and education filtered.

```
gold_filter_education<-gold_filter%>%
  filter(Education=="Graduation")
```

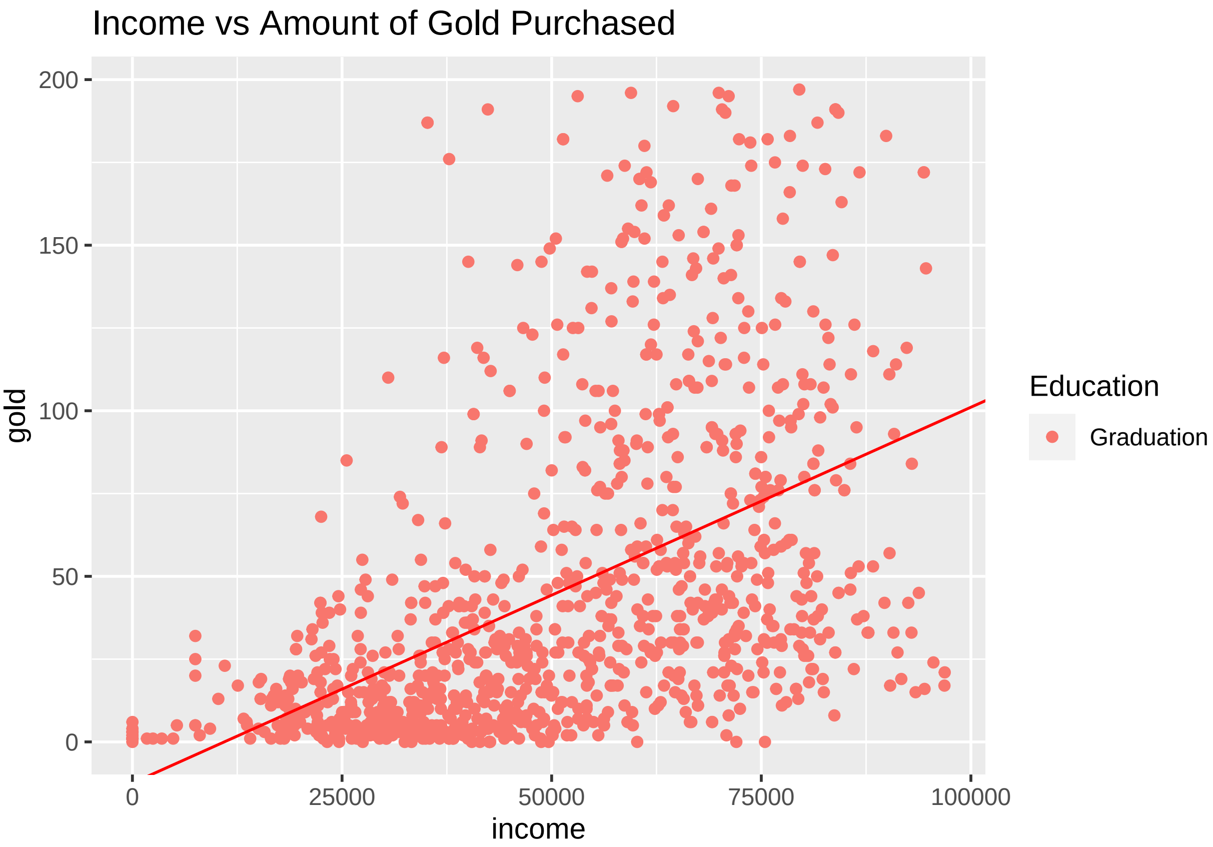
```
lm_gold_education<-lm(gold~income,gold_filter_education)
summary(lm_gold_education)
```

```
##
## Call:
## lm(formula = gold ~ income, data = gold_filter_education)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -80.467 -26.542  -9.176  14.511 159.491
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept) -1.243e+01  3.245e+00  -3.831 0.000135 ***
## income       1.135e-03  5.899e-05  19.237  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 41.22 on 1090 degrees of freedom
## Multiple R-squared:  0.2535, Adjusted R-squared:  0.2528
## F-statistic: 370.1 on 1 and 1090 DF,  p-value: < 2.2e-16
```

```
cor(gold_filter_education$gold,gold_filter_education$income)
```

```
## [1] 0.5034495
```

```
income_gold_education<-gold_filter_education%>%
  ggplot(aes(income,gold,color=Education))+
  geom_point()+
  ggtitle("Income vs Amount of Gold Purchased")+
  geom_abline(slope=0.001135,intercept=-12.431358 ,color="red")
income_gold_education
```



The filtered linear model still gave a relatively weak R squared value, and the correlation is moderate between gold and income still, but it improved from the non-filtered sample of the data.

This next section looks at the relationship between sweets and income for those with a graduate level of education.

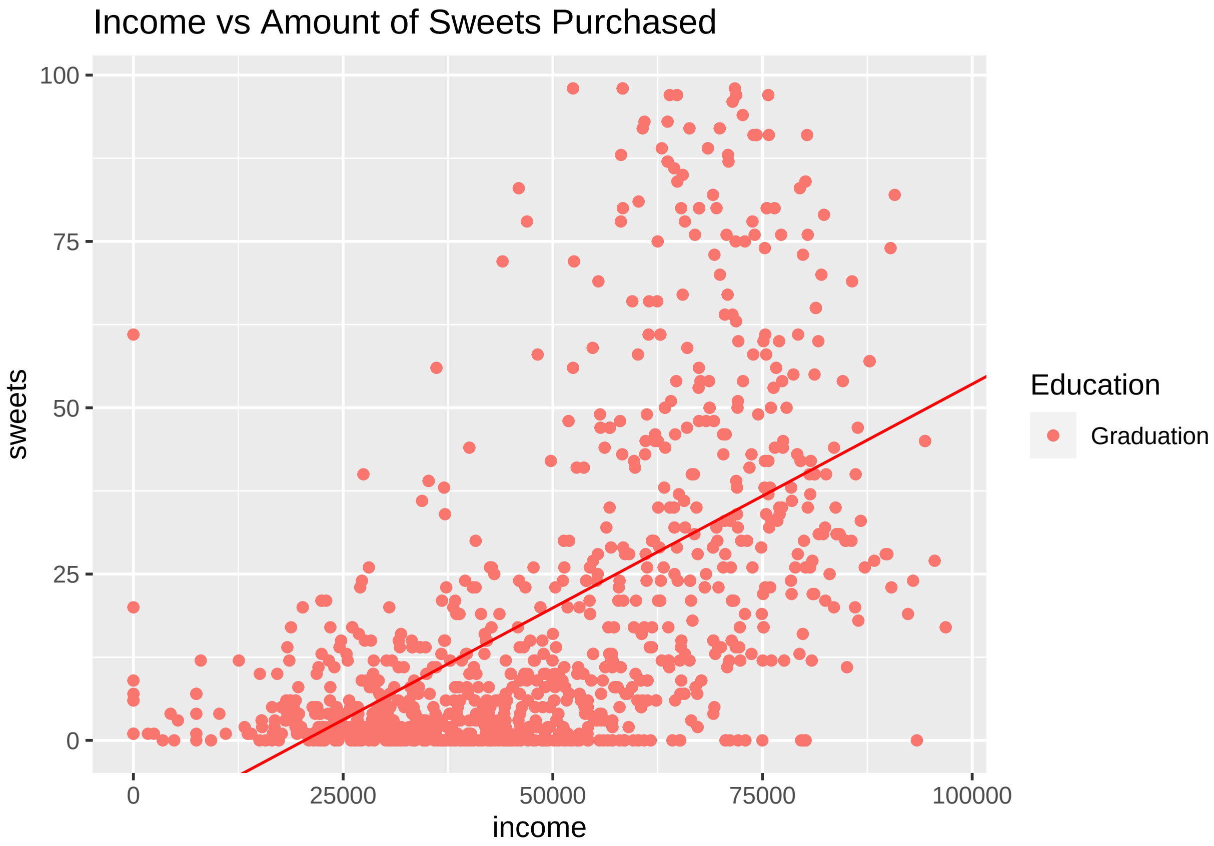
```
sweets_filter_education<-sweets_filter%>%
  filter(Education=="Graduation")

lm_sweets_education<-lm(sweets~income,sweets_filter_education)
summary(lm_sweets_education)
```

```
##
## Call:
## lm(formula = sweets ~ income, data = sweets_filter_education)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -49.120 -12.723  -3.570   6.416  76.464
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept) -1.373e+01  1.626e+00  -8.447   <2e-16 ***
## income       6.729e-04  3.074e-05  21.888   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 19.82 on 1006 degrees of freedom
## Multiple R-squared:  0.3226, Adjusted R-squared:  0.3219
## F-statistic: 479.1 on 1 and 1006 DF,  p-value: < 2.2e-16
```

```
cor(sweets_filter_education$income,sweets_filter_education$sweets)
```

```
## [1] 0.567975
```

```
income_sweets_education<-sweets_filter_education%>%
  ggplot(aes(income,sweets,color=Education))+
  geom_point()+
  ggtitle("Income vs Amount of Sweets Purchased")+
  geom_abline(slope=6.729e-04 ,intercept=-1.373e+01,color="red")
income_sweets_education
```



The R squared value returns a weak value for the strength of the linear model, but again it showed an improvement from before, as well as the correlation coefficient for the values.

This next section looks at the relationship between meat and income for those with a graduate level of education.

```
meat_filter_education<-meat_filter%>%
  filter(Education=="Graduation")

lm_meat_education<-lm(meat~income,meat_filter_education)
summary(lm_meat_education)
```

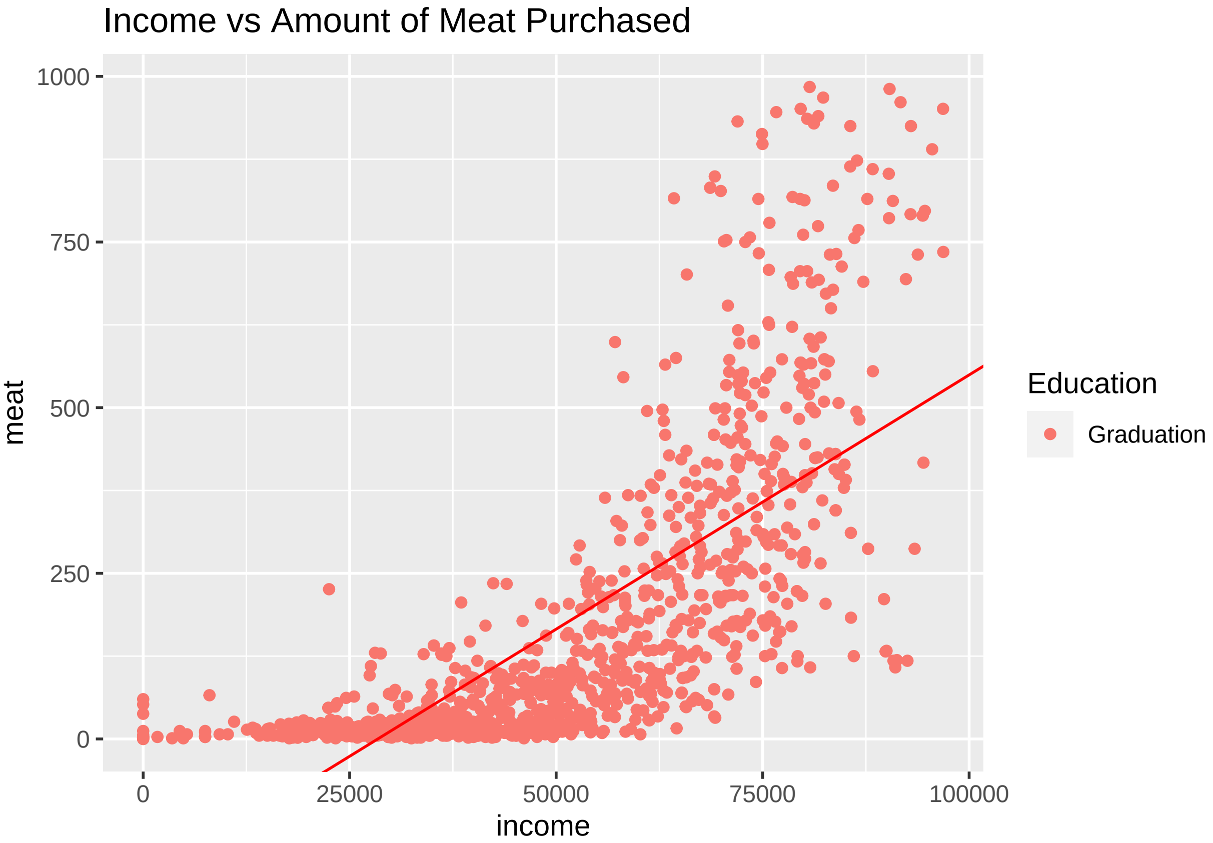
```
##
## Call:
## lm(formula = meat ~ income, data = meat_filter_education)
##
## Residuals:
```

```
##     Min      1Q  Median      3Q     Max
## -374.03  -96.25  -19.38   65.73  597.81
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept) -2.180e+02  1.168e+01  -18.66   <2e-16 ***
## income       7.674e-03  2.101e-04   36.52   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 149.4 on 1118 degrees of freedom
## Multiple R-squared:  0.5439, Adjusted R-squared:  0.5435
## F-statistic:  1333 on 1 and 1118 DF,  p-value: < 2.2e-16
```

```
cor(meat_filter_education$income,meat_filter_education$meat)
```

```
## [1] 0.7375233
```

```
income_meat_education<-meat_filter_education%>%
  ggplot(aes(income,meat,color=Education))+
  geom_point()+
  ggtitle("Income vs Amount of Meat Purchased")+
  geom_abline(slope= 7.674e-03,intercept= -2.180e+02,color="red")
income_meat_education
```



There has been a slight improvement in the R squared value for meat, and the correlation coefficient is on the higher end of a moderately strong relationship.

This next section looks at the relationship between fish and income for those with a graduate level of education.

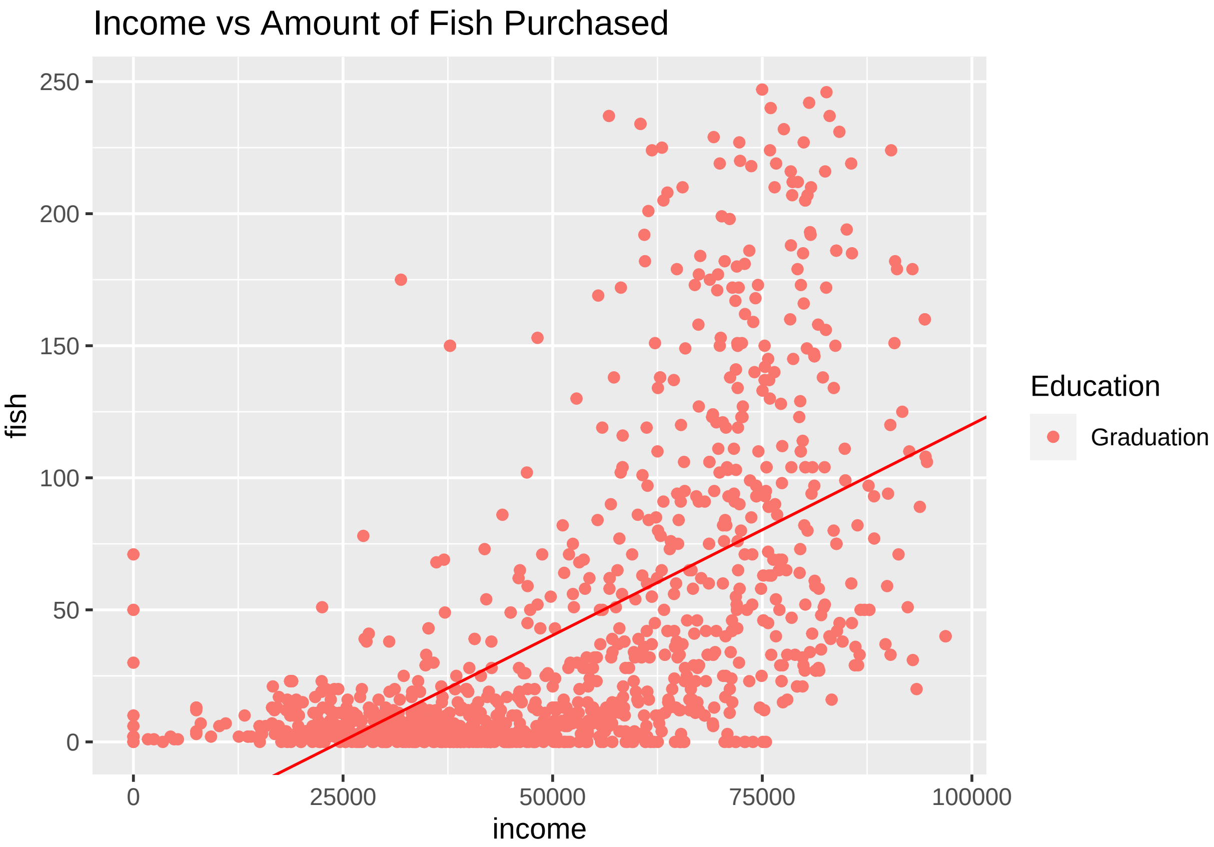
```
fish_filter_education<-fish_filter%>%
  filter(Education=="Graduation")

lm_fish_education<-lm(fish~income,fish_filter_education)
summary(lm_fish_education)
```

```
##
## Call:
## lm(formula = fish ~ income, data = fish_filter_education)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -89.746 -29.876  -8.125  16.491 185.894
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept) -3.954e+01  3.638e+00  -10.87   <2e-16 ***
## income       1.598e-03  6.557e-05   24.37   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 46.43 on 1114 degrees of freedom
## Multiple R-squared:  0.3478, Adjusted R-squared:  0.3472
## F-statistic: 594.1 on 1 and 1114 DF,  p-value: < 2.2e-16
```

```
cor(fish_filter_education$income,fish_filter_education$fish)
```

```
## [1] 0.5897461
```

```
income_fish_education<-fish_filter_education%>%
  ggplot(aes(income,fish,color=Education))+
  geom_point()+
  ggtitle("Income vs Amount of Fish Purchased")+
  geom_abline(slope=0.001598,intercept=-39.540585,color="red")
income_fish_education
```



The multiple R square value is relatively weak, but the correlation coefficient is moderately strong which means the variables are still good predictors of one another and have a significant relationship because like all the models filtered for education, income has three asterisks next to its p values.

This final section looks at the relationship between fruits and income for those with a graduate level of education.

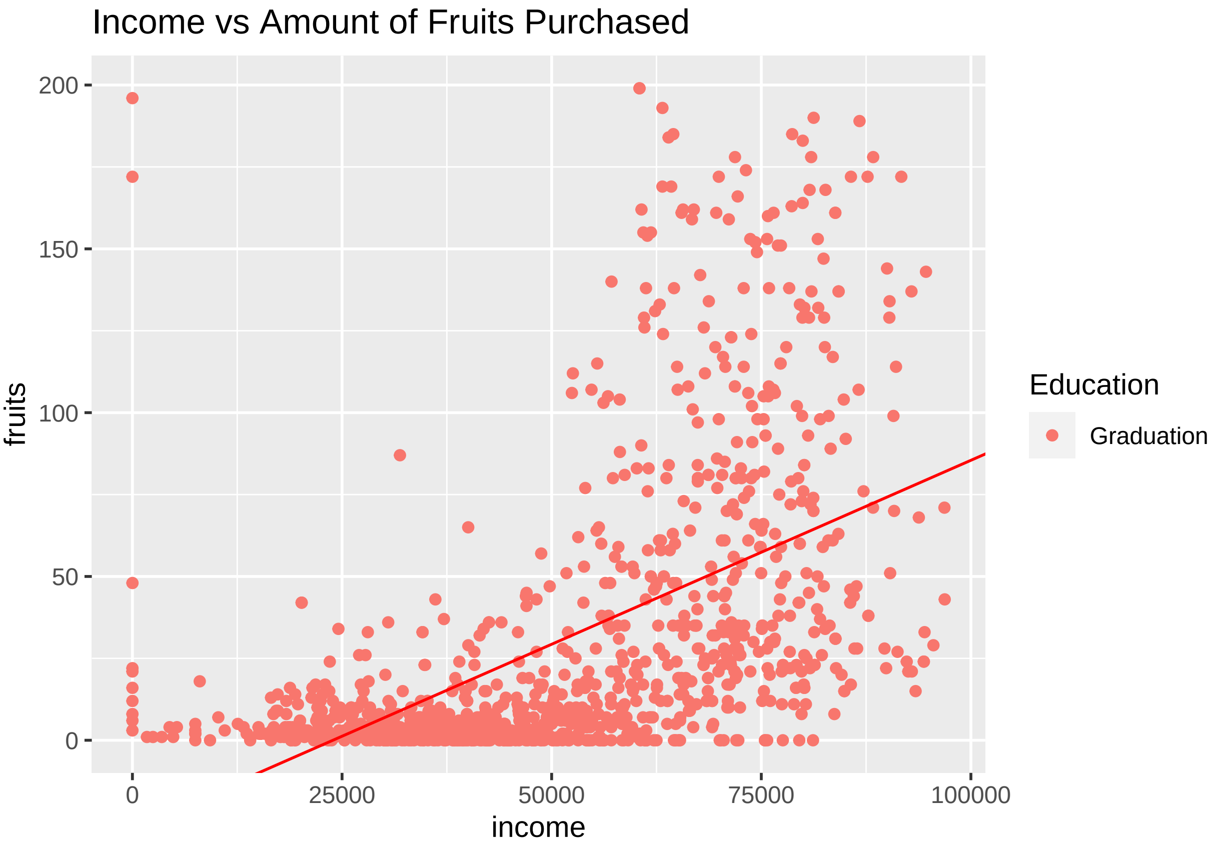
```
fruits_filter_education<-fruits_filter%>%
  filter(Education=="Graduation")

lm_fruit_education<-lm(fruits~income,fruits_filter_education)
summary(lm_fruit_education)
```

```
##
## Call:
## lm(formula = fruits ~ income, data = fruits_filter_education)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -64.326 -23.244  -7.791  10.682 222.844
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept) -2.684e+01  2.908e+00   -9.23   <2e-16 ***
## income       1.123e-03  5.231e-05   21.47   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 37.18 on 1118 degrees of freedom
## Multiple R-squared:  0.292,  Adjusted R-squared:  0.2914
## F-statistic: 461.1 on 1 and 1118 DF,  p-value: < 2.2e-16
```

```
cor(fruits_filter_education$income,fruits_filter_education$fruits)
```

```
## [1] 0.5403708
```

```
income_fruit_education<-fruits_filter_education%>%
  ggplot(aes(income,fruits,color=Education))+
  geom_point()+
  ggtitle("Income vs Amount of Fruits Purchased")+
  geom_abline(slope=0.001123,intercept=-26.843835,color="red")
income_fruit_education
```



Finally, this model as well as all others have pretty low R squared values, but what is important to note is their improvement as more demographic variables are added to the linear models.

I saw a consistent trend suggesting that linear models are not the best indicators of the relationships between my data, and decided to test nonlinear models on the variables best suited for this experiment.

Wine:

```
lm_wine2<-lm(wine~income+I(income^2),marketing_education_filter1)
summary(lm_wine2)
```

```
##
## Call:
## lm(formula = wine ~ income + I(income^2), data = marketing_education_filter1)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -331.75  -64.41  -15.85   54.66  408.50
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept) -7.563e+01  1.665e+01  -4.542 6.32e-06 ***
## income       3.240e-03  7.456e-04   4.346 1.55e-05 ***
## I(income^2)  3.443e-08  7.639e-09   4.507 7.42e-06 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 104.8 on 911 degrees of freedom
## Multiple R-squared:  0.6098, Adjusted R-squared:  0.609
## F-statistic: 711.9 on 2 and 911 DF,  p-value: < 2.2e-16
```

```r
lm_wine3<-lm(wine~income+I(income^2)+I(income^3),marketing_education_filter1)
summary(lm_wine3)
```

```
##
## Call:
## lm(formula = wine ~ income + I(income^2) + I(income^3), data = marketing_education_filter1)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -310.70  -59.90   -2.27   42.31  410.74
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept)  6.217e+01  2.185e+01   2.845  0.00454 **
## income      -1.061e-02  1.664e-03  -6.375  2.9e-10 ***
## I(income^2)  3.943e-07  3.974e-08   9.921  < 2e-16 ***
## I(income^3) -2.647e-12  2.873e-13  -9.212  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 100.3 on 910 degrees of freedom
## Multiple R-squared:  0.6431, Adjusted R-squared:  0.6419
## F-statistic: 546.6 on 3 and 910 DF,  p-value: < 2.2e-16
```

The best nonlinear model for wine as a function of income was 'lm_wine3' because it had the highest multiple R-square value and all p values were significant, so I can use the formula:

wine = 62.17 -1.061e-02*income + 3.943e-07*(income^2) + -2.647e-12*(income^3)

to make a prediction regarding how much wine a person with an income of $100,000 would buy.

```r
62.17 -1.061e-02*100000 + 3.943e-07*(100000^2) + -2.647e-12*(100000^3)
```

```
## [1] 297.17
```

The equation tells us that this consumer is likely to spend a predicted value of $297.17 on wine over the course of two years.

Gold:

```r
lm_gold2<-lm(gold~income+I(income^2),gold_filter_education)
summary(lm_gold2)
```

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

```
## lm(formula = gold ~ income + I(income^2), data = gold_filter_education)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -77.145 -26.886  -9.404  15.655 159.061
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept) -1.639e+01  6.249e+00  -2.623  0.00884 **
## income       1.327e-03  2.658e-04   4.993 6.93e-07 ***
## I(income^2) -1.916e-09  2.584e-09  -0.741  0.45864
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 41.23 on 1089 degrees of freedom
## Multiple R-squared:  0.2538, Adjusted R-squared:  0.2525
## F-statistic: 185.2 on 2 and 1089 DF,  p-value: < 2.2e-16
```

```
lm_gold3<-lm(gold~income+I(income^2)+I(income^3),gold_filter_education)
summary(lm_gold3)
```

```
##
## Call:
## lm(formula = gold ~ income + I(income^2) + I(income^3), data = gold_filter_education)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -76.143 -26.848  -8.023  14.100 163.996
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept)  1.533e+01  8.703e+00   1.762   0.0784 .
## income      -1.573e-03  6.190e-04  -2.541   0.0112 *
## I(income^2)  6.779e-08  1.372e-08   4.943 8.90e-07 ***
## I(income^3) -4.807e-13  9.292e-14  -5.173 2.74e-07 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 40.75 on 1088 degrees of freedom
## Multiple R-squared:  0.2718, Adjusted R-squared:  0.2697
## F-statistic: 135.3 on 3 and 1088 DF,  p-value: < 2.2e-16
```

The best model for these two variables ended up being the original linear model filtered for education because it had the highest multiple R-square value and all p values were significant, so I can use the formula:

gold = -12.431358 + 0.001135*income

to make a prediction regarding how much wine a person with an income of $100,000 would buy.

```
-12.431358 + 0.001135*100000
```

```
## [1] 101.0686
```

The equation tells us that this consumer is likely to spend a predicted value of $101.07 on gold over the course of two years.

Meat:

```
lm_meat2<-lm(meat~income+I(income^2),meat_filter_education)
summary(lm_meat2)
```

```
##
## Call:
## lm(formula = meat ~ income + I(income^2), data = meat_filter_education)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -560.24  -58.46   -0.09   31.04  595.91
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept)  7.516e+01  2.017e+01   3.726 0.000204 ***
## income      -6.491e-03  8.552e-04  -7.590 6.73e-14 ***
## I(income^2)  1.406e-07  8.281e-09  16.976  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 133.3 on 1117 degrees of freedom
## Multiple R-squared:  0.6375, Adjusted R-squared:  0.6368
## F-statistic: 982.1 on 2 and 1117 DF,  p-value: < 2.2e-16
```

```
lm_meat3<-lm(meat~income+I(income^2)+I(income^3),meat_filter_education)
summary(lm_meat3)
```

```
##
## Call:
## lm(formula = meat ~ income + I(income^2) + I(income^3), data = meat_filter_education)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -565.86  -57.31   -1.11   31.33  597.26
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept)  6.630e+01  2.856e+01   2.321  0.02045 *
## income      -5.685e-03  2.027e-03  -2.805  0.00512 **
## I(income^2)  1.213e-07  4.477e-08   2.709  0.00685 **
## I(income^3)  1.327e-13  3.026e-13   0.438  0.66113
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 133.3 on 1116 degrees of freedom
## Multiple R-squared:  0.6375, Adjusted R-squared:  0.6366
## F-statistic: 654.3 on 3 and 1116 DF,  p-value: < 2.2e-16
```

The best nonlinear model for wine as a function of income was 'lm_meat2' because although both models had the same multiple R-square values, the p values were more significant in the first nonlinear model, so I can use the following formula:

meat = 75.16 -6.491e-03*income+1.406e-07(income^2)

to make a prediction regarding how much wine a person with an income of $100,000 would buy.

```
75.16 -6.491e-03*100000+1.406e-07*(100000^2)
```

```
## [1] 832.06
```

The equation tells us that this consumer is likely to spend a predicted value of $832.06 on meat over the course of two years.

Fish:

```
lm_fish2<-lm(fish~income+I(income^2),fish_filter_education)
summary(lm_fish2)
```

```
##
## Call:
## lm(formula = fish ~ income + I(income^2), data = fish_filter_education)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -122.020  -23.842   -5.504    9.891  195.278
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept)  8.175e+00  6.860e+00   1.192   0.2336
## income      -7.092e-04  2.913e-04  -2.435   0.0151 *
## I(income^2)  2.293e-08  2.825e-09   8.118 1.25e-15 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 45.14 on 1113 degrees of freedom
## Multiple R-squared:  0.3843, Adjusted R-squared:  0.3832
## F-statistic: 347.3 on 2 and 1113 DF,  p-value: < 2.2e-16
```

```
lm_fish3<-lm(fish~income+I(income^3),fish_filter_education)
summary(lm_fish3)
```

```
##
## Call:
## lm(formula = fish ~ income + I(income^3), data = fish_filter_education)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -124.383  -24.096   -6.713   10.807  195.138
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept) -7.127e+00  5.761e+00  -1.237   0.2164
## income       4.209e-04  1.766e-04   2.383   0.0173 *
## I(income^3)  1.377e-13  1.924e-14   7.154 1.52e-12 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 45.42 on 1113 degrees of freedom
## Multiple R-squared:  0.3765, Adjusted R-squared:  0.3754
## F-statistic:   336 on 2 and 1113 DF,  p-value: < 2.2e-16
```

The best model for wine as a function of income was the original linear model filtered for education level, because it had the highest multiple R-square value and all p values were significant, so I can use the formula:

fish = -39.540585 + 0.001598*income

to make a prediction regarding how much wine a person with an income of $100,000 would buy.

```
-39.540585 + 0.001598*100000
```

```
## [1] 120.2594
```

The equation tells us that this consumer is likely to spend a predicted value of $120.23 on fish over the course of two years.

Fruits:

```
lm_fruit2<-lm(fruits~income+I(income^2),fruits_filter_education)
summary(lm_fruit2)
```

```
##
## Call:
## lm(formula = fruits ~ income + I(income^2), data = fruits_filter_education)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -88.546 -17.105  -5.364   4.881 184.790
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept)  1.121e+01  5.483e+00   2.044  0.04116 *
## income      -7.140e-04  2.325e-04  -3.072  0.00218 **
## I(income^2)  1.823e-08  2.250e-09   8.100 1.43e-15 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 36.15 on 1117 degrees of freedom
## Multiple R-squared:  0.3313, Adjusted R-squared:  0.3301
## F-statistic: 276.7 on 2 and 1117 DF,  p-value: < 2.2e-16
```

```
lm_fruit3<-lm(fruits~income+I(income^3),fruits_filter_education)
summary(lm_fruit3)
```

```
##
## Call:
## lm(formula = fruits ~ income + I(income^3), data = fruits_filter_education)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -88.958 -17.676  -6.419   5.787 198.332
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept) -2.332e+00  4.614e+00  -0.505   0.6133
## income       2.343e-04  1.412e-04   1.660   0.0972 .
## I(income^3)  1.036e-13  1.533e-14   6.759 2.24e-11 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 36.46 on 1117 degrees of freedom
## Multiple R-squared:  0.3198, Adjusted R-squared:  0.3186
## F-statistic: 262.6 on 2 and 1117 DF,  p-value: < 2.2e-16
```

The best model for wine as a function of income was 'lm_fruit2', because it had the highest multiple R-square value and all p values were more significant than 'lm_fruit3', so I can use the formula:

fruit=11.21-7.140e-04*income+1.823e-08*(income^2)

to make a prediction regarding how much wine a person with an income of $100,000 would buy.

```r
11.21-7.140e-04*100000+1.823e-08*(100000^2)
```

```
## [1] 122.11
```

The equation tells us that this consumer is likely to spend a predicted value of $122.11 on fruits over the course of two years.

Some conclusions I can make based on my analyses are that relationships between variables can improve when adding more information. This relates to my overall question: Would adding demographic information increase the accuracy of predicting what consumers are likely to buy? This is such valuable information for marketers to use to understand where their efforts are going to be the most impactful. They could potentially use the formulas I created to decide if someone is a potential consumer based on their level of income, and level of education. I also learned that creating nonlinear models improve the predictability of my data. Some things that could improve include the level of depth and amount of demographic information I could add to improve the accuracy of my models. One variable that I left out which could be influential is the age of each consumer. Had I segmented each age group and included that information, there could have been a stronger correlation coefficient for my variables which would lead to stronger predictions. However, I still am happy with the conclusions I was able to make, considering my data was not the most suitable for linear regression, I was still able to create models that could act as guildelines for marketers to use.