Econ 403A: Project 2

David Contento, John Macke, Pujan Thakrar, Mark Vandre

December 5, 2018

First we check the structure of our data and determine whether or not there are any NAs in the data.

```
data <- read.csv("BlackFriday.csv", header = TRUE, stringsAsFactors = TRUE)
str(data)
```

```
## 'data.frame':    537577 obs. of  12 variables:
##  $ User_ID                   : int  1000001 1000001 1000001 1000001 1000002 1000003 1000004 1000004 1
##  $ Product_ID                : Factor w/ 3623 levels "P00000142","P00000242",..: 671 2375 851 827 273
##  $ Gender                    : Factor w/ 2 levels "F","M": 1 1 1 1 2 2 2 2 2 2 ...
##  $ Age                       : Factor w/ 7 levels "0-17","18-25",..: 1 1 1 1 7 3 5 5 5 3 ...
##  $ Occupation                : int  10 10 10 10 16 15 7 7 7 20 ...
##  $ City_Category             : Factor w/ 3 levels "A","B","C": 1 1 1 1 3 1 2 2 2 1 ...
##  $ Stay_In_Current_City_Years: Factor w/ 5 levels "0","1","2","3",..: 3 3 3 3 5 4 3 3 3 2 ...
##  $ Marital_Status            : int  0 0 0 0 0 0 1 1 1 1 ...
##  $ Product_Category_1        : int  3 1 12 12 8 1 1 1 1 8 ...
##  $ Product_Category_2        : int  NA 6 NA 14 NA 2 8 15 16 NA ...
##  $ Product_Category_3        : int  NA 14 NA NA NA NA 17 NA NA NA ...
##  $ Purchase                  : int  8370 15200 1422 1057 7969 15227 19215 15854 15686 7871 ...
```

```
#checking which variables have NA's
colSums(is.na(data))
```

```
##                    User_ID                 Product_ID
##                          0                          0
##                     Gender                        Age
##                          0                          0
##                 Occupation              City_Category
##                          0                          0
## Stay_In_Current_City_Years             Marital_Status
##                          0                          0
##         Product_Category_1         Product_Category_2
##                          0                     166986
##         Product_Category_3                   Purchase
##                     373299                          0
```

Since we have NAs in Product_Category_2 and Product_Category_3 we combine those into a dummy variable called "multi". Multi takes on a value of 1 if the product belongs to multiple categories and a value of 0 if the product belongs to only one category.

```
#combining product category 2 and 3 variable into dummy:
#with 1 if in more than one product category and 0 if not
data$multi = 1
data[is.na(data$Product_Category_2), "multi"] = 0

#remove product category 2 and 3 variables
data <- data[, -c(10:11)]
str(data)
```

```
## 'data.frame':    537577 obs. of  11 variables:
##  $ User_ID                   : int  1000001 1000001 1000001 1000001 1000002 1000003 1000004 1000004 1
```

```
##  $ Product_ID                : Factor w/ 3623 levels "P00000142","P00000242",..: 671 2375 851 827 273
##  $ Gender                    : Factor w/ 2 levels "F","M": 1 1 1 1 2 2 2 2 2 2 ...
##  $ Age                       : Factor w/ 7 levels "0-17","18-25",..: 1 1 1 1 7 3 5 5 5 3 ...
##  $ Occupation                : int  10 10 10 10 16 15 7 7 7 20 ...
##  $ City_Category             : Factor w/ 3 levels "A","B","C": 1 1 1 1 3 1 2 2 2 1 ...
##  $ Stay_In_Current_City_Years: Factor w/ 5 levels "0","1","2","3",..: 3 3 3 3 5 4 3 3 3 2 ...
##  $ Marital_Status            : int  0 0 0 0 0 0 1 1 1 1 ...
##  $ Product_Category_1        : int  3 1 12 12 8 1 1 1 1 8 ...
##  $ Purchase                  : int  8370 15200 1422 1057 7969 15227 19215 15854 15686 7871 ...
##  $ multi                     : num  0 1 0 1 0 1 1 1 1 0 ...
```

```
colSums(is.na(data))
```

```
##                   User_ID                Product_ID
##                         0                         0
##                    Gender                       Age
##                         0                         0
##                Occupation             City_Category
##                         0                         0
## Stay_In_Current_City_Years            Marital_Status
##                         0                         0
##         Product_Category_1                  Purchase
##                         0                         0
##                      multi
##                         0
```

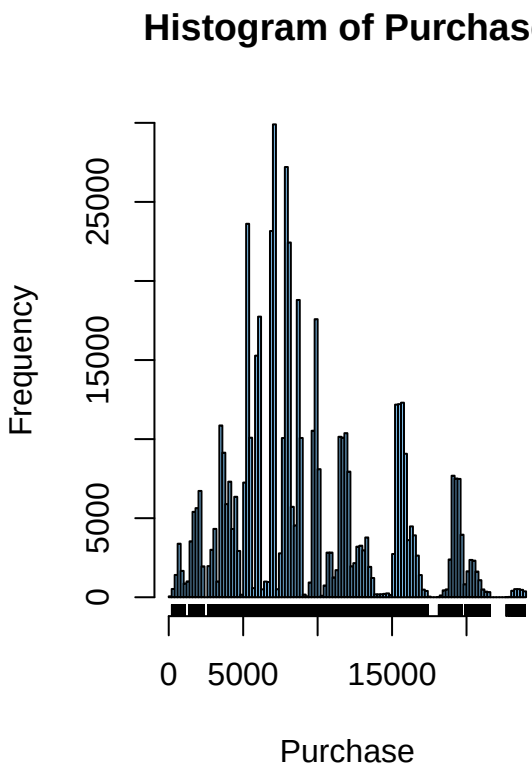
Now we will move on to the descriptive analysis. Notice that most of our variables are categorical and therefore we only make histograms for Purchase and Occupation.

We also look at the Quantile-Quantile plots for Purchase in order to determine its normality.

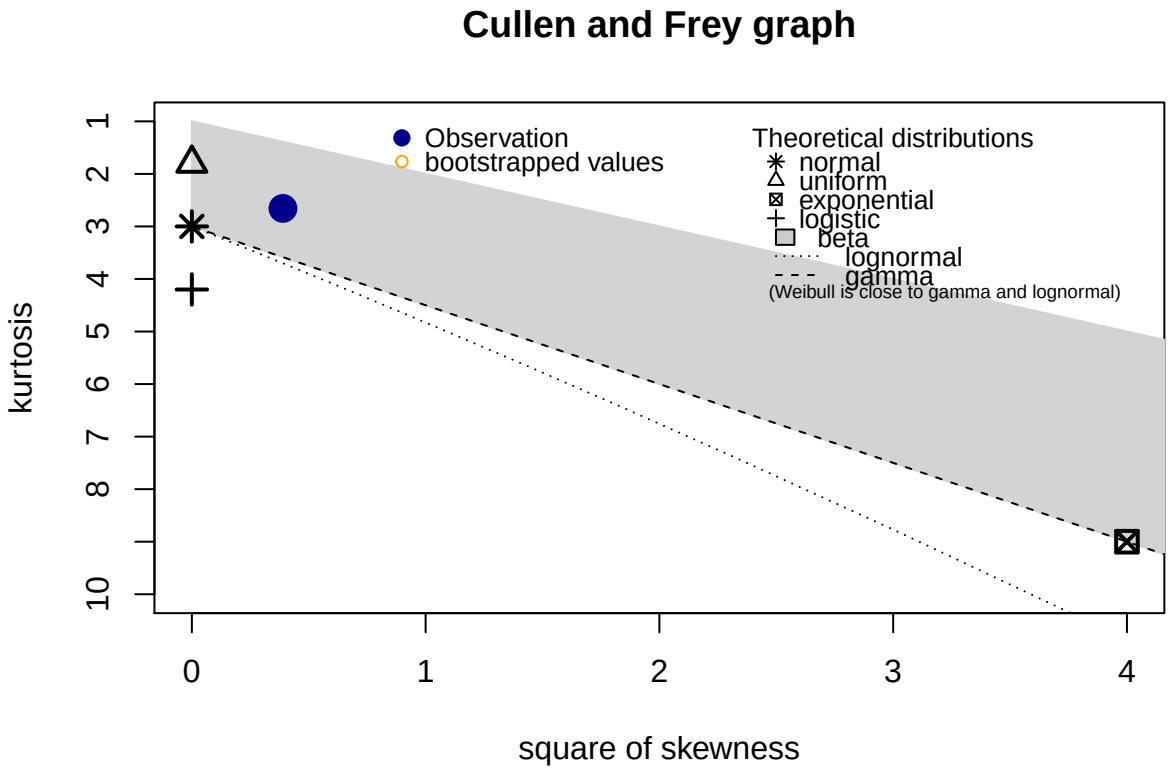
```
attach(data)

par(mfrow=c(1,2))

hist(Purchase, breaks = "FD", col = "skyblue2")
rug(Purchase)
S(Purchase)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##     185    5866    8062    9334   12073   23961
```

```
#inc

par(mfrow=c(1,1))
```

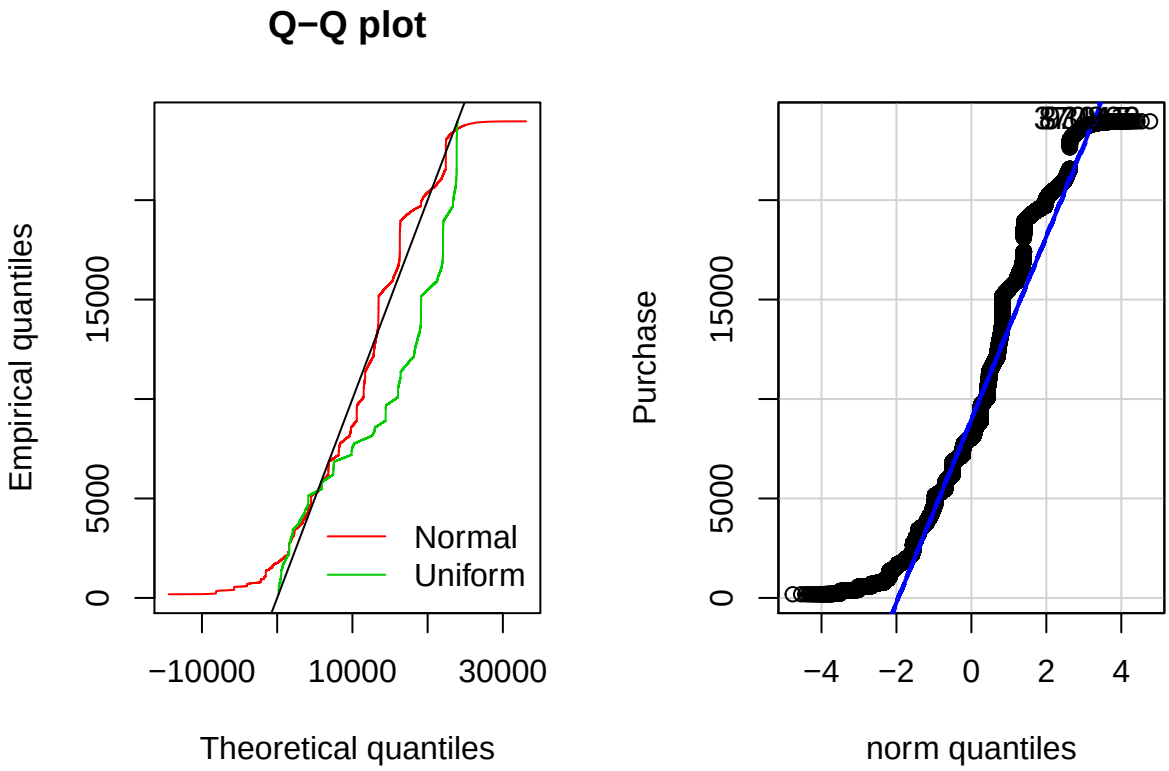



```
descdist(Purchase, boot = 1000)
```




```
## summary statistics
## ------
## min:  185   max:  23961
## median:  8062
## mean:  9333.86
## estimated sd:  4981.022
## estimated skewness:  0.6242797
## estimated kurtosis:  2.656879
```

```
fit.norm = fitdist(as.numeric(Purchase), "norm")
fit.unif = fitdist(as.numeric(Purchase),"unif")
plot.legend = c("Normal", "Uniform")
par(mfrow=c(1,2))
qqcomp(list(fit.norm, fit.unif), legendtext = plot.legend)
qqPlot(~ Purchase, data = data, id = list(n=3))
```

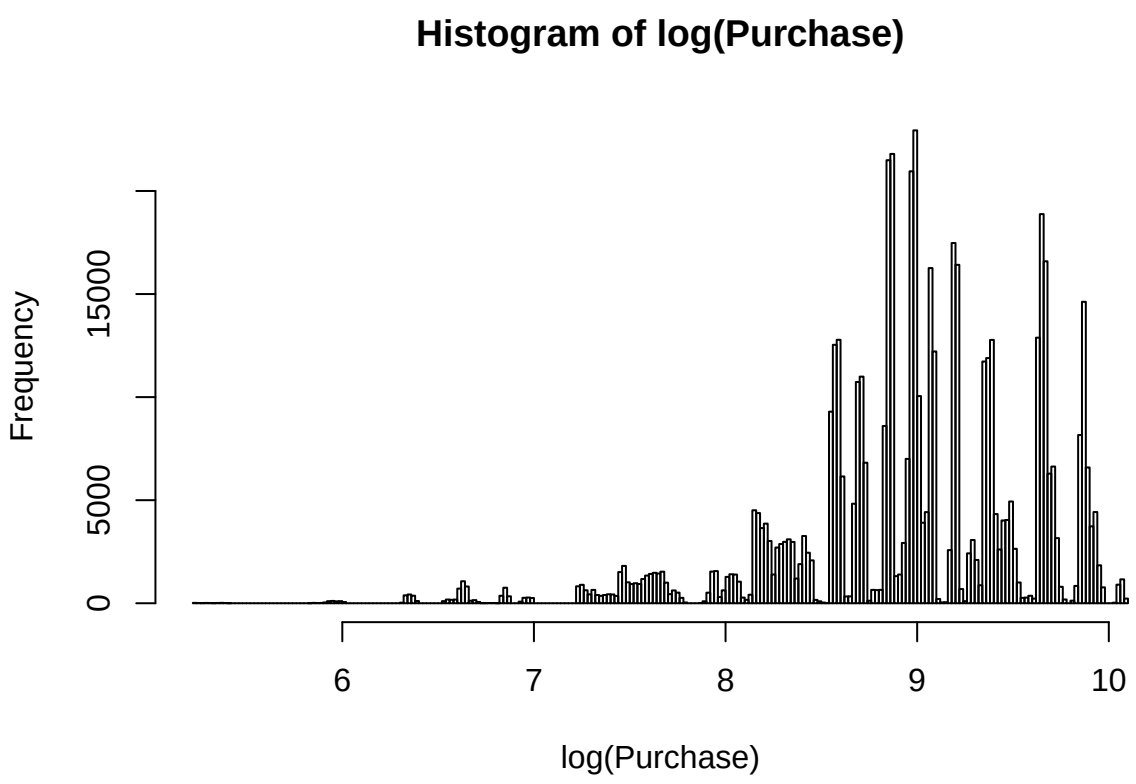



```
## [1]  87441  93017 370892
```

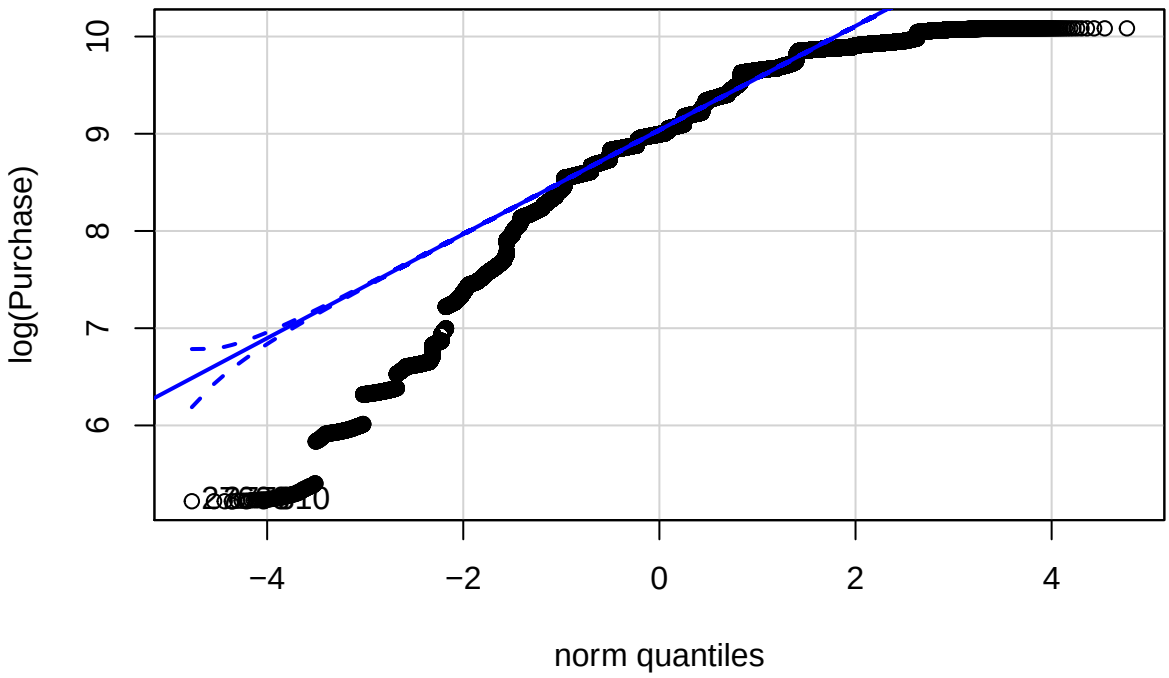
```
#i fit a gamma distribution to the inc data
#par(mfrow=c(1,1))
#denscomp(list(fit.gamma))
```

Purchase looks fairly normal from the QQ-Plots but we log the data just to make sure there isn't a better transformation.

```
hist(log(Purchase), breaks = "FD")
```



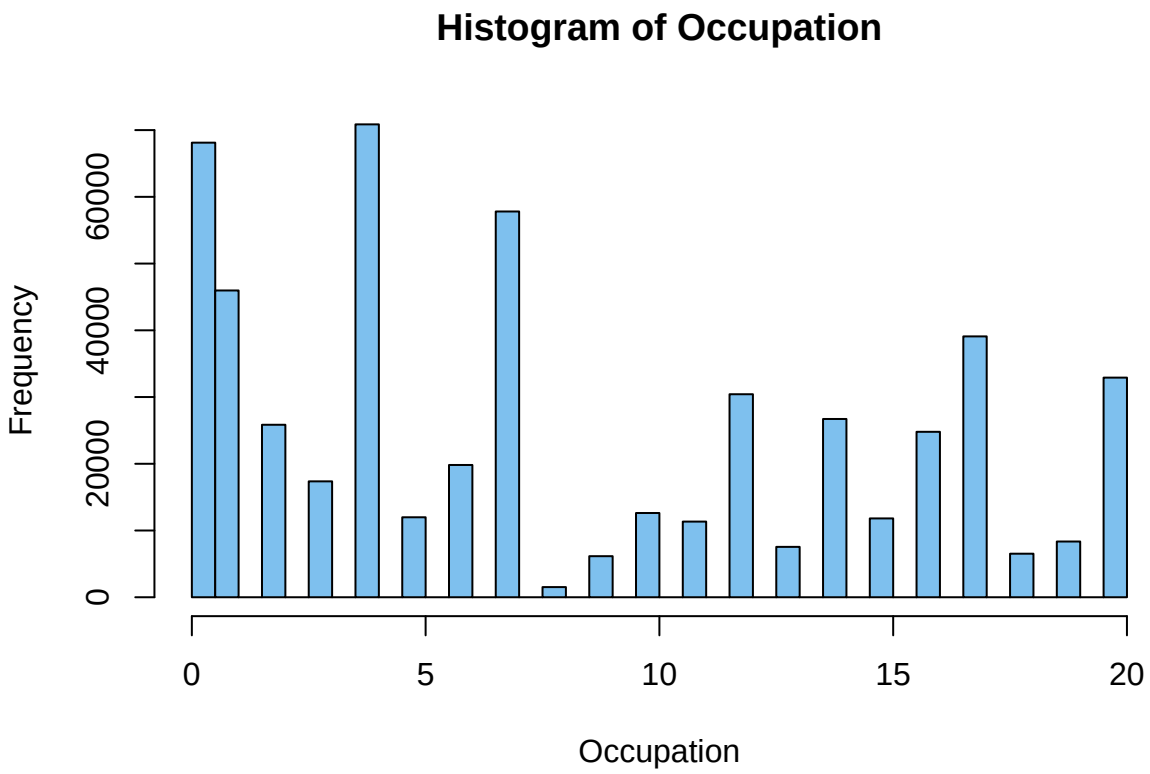

```
qqPlot(log(Purchase))
```




```
## [1]  27603 377310
```

We can see from this that Purchase should not be transformed for now. Next, we look at the histogram for Occupation.

```
par(mfrow=c(1,1))
hist(Occupation, breaks = "FD", col = "skyblue2")
```



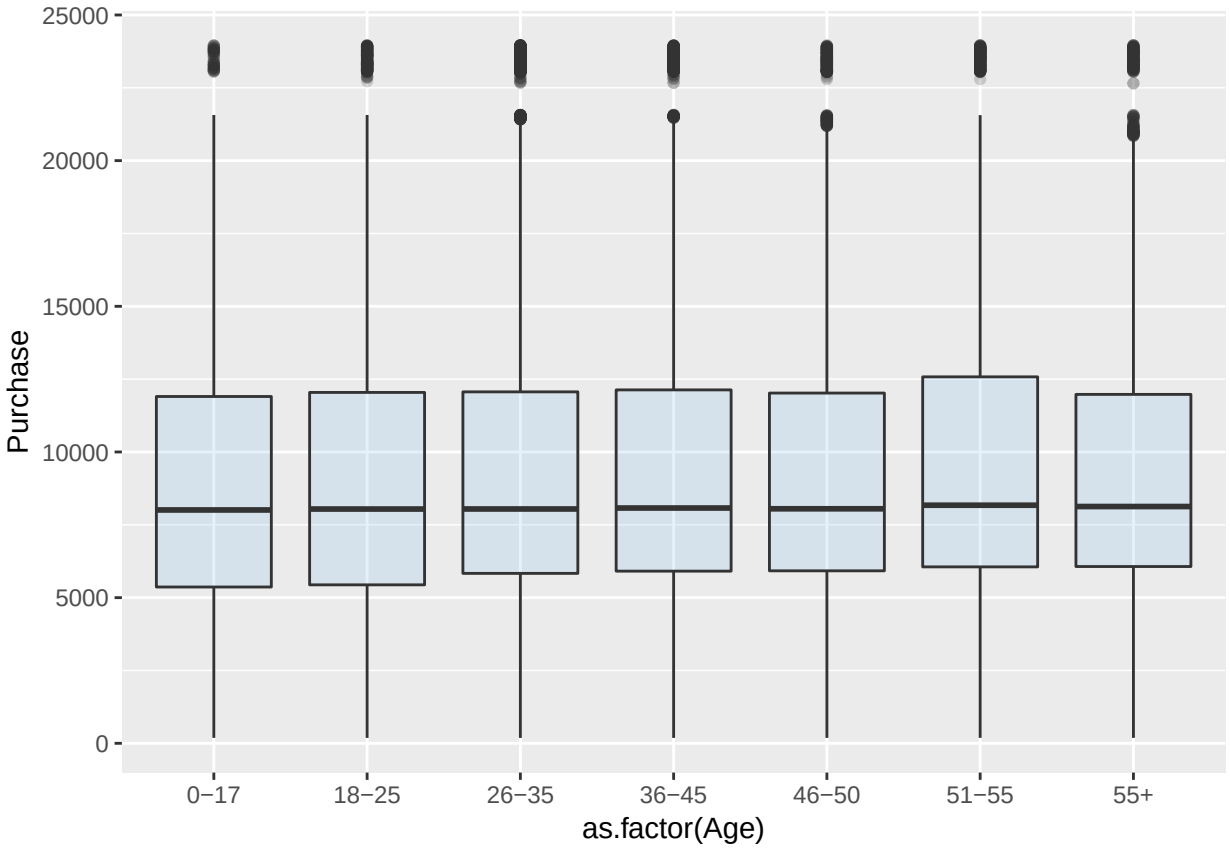

```
S(Occupation)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   0.000   2.000   7.000   8.083  14.000  20.000
```

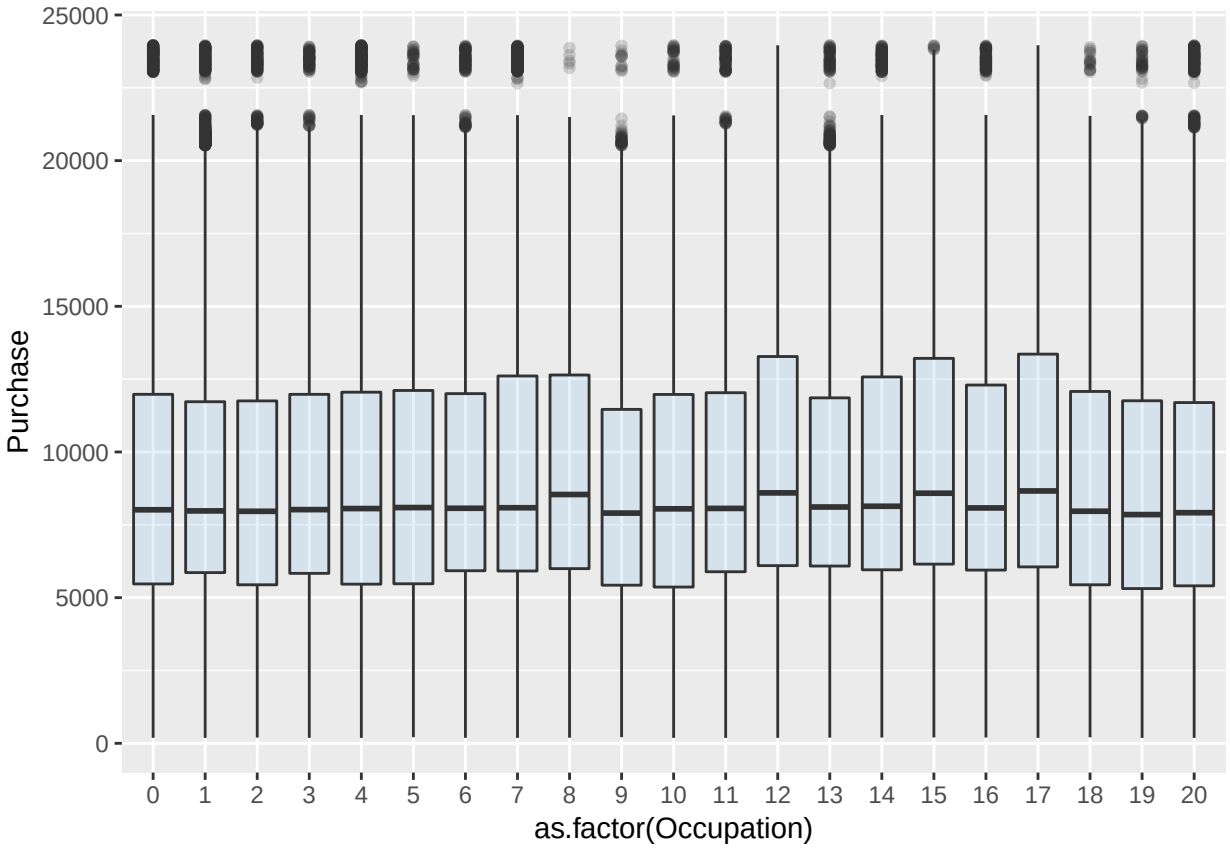
This plot illustrates how many observed individuals fall in specific Occupation categories.

Next, we look at the boxplots to see what the spread of purchase amount looks like for each of the variables.

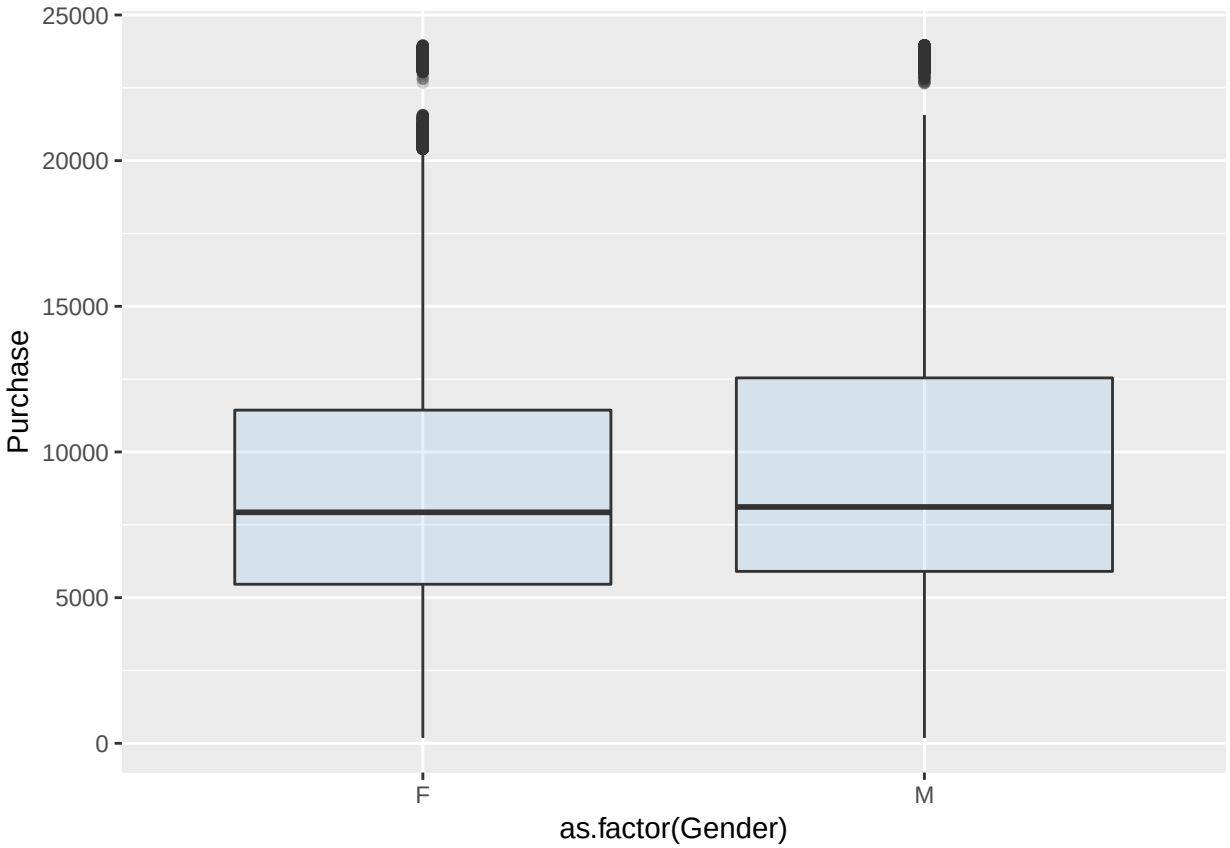
```
ggplot(data, aes(x=as.factor(Age), y=Purchase)) +
  geom_boxplot(fill="skyblue2", alpha=0.2)
```



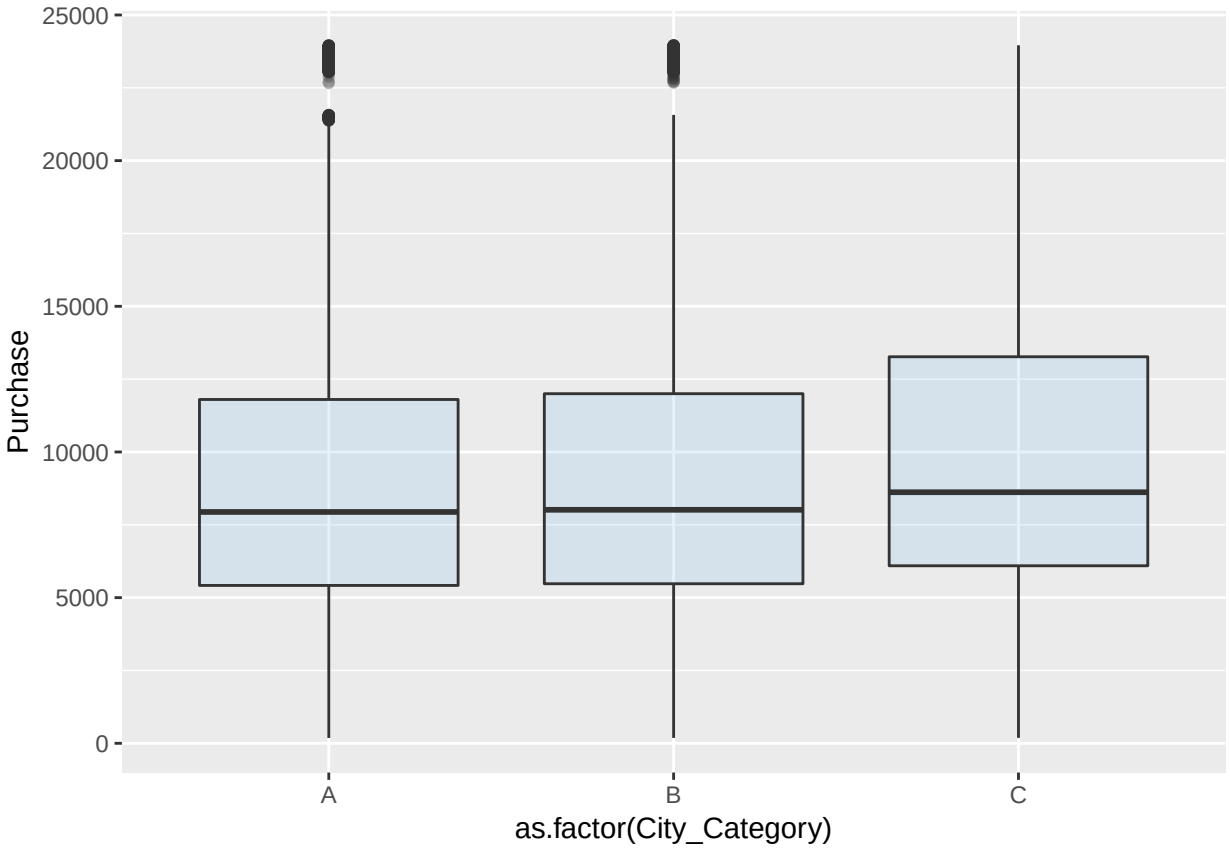

```
ggplot(data, aes(x=as.factor(Occupation), y=Purchase)) +
  geom_boxplot(fill="skyblue2", alpha=0.2)
```



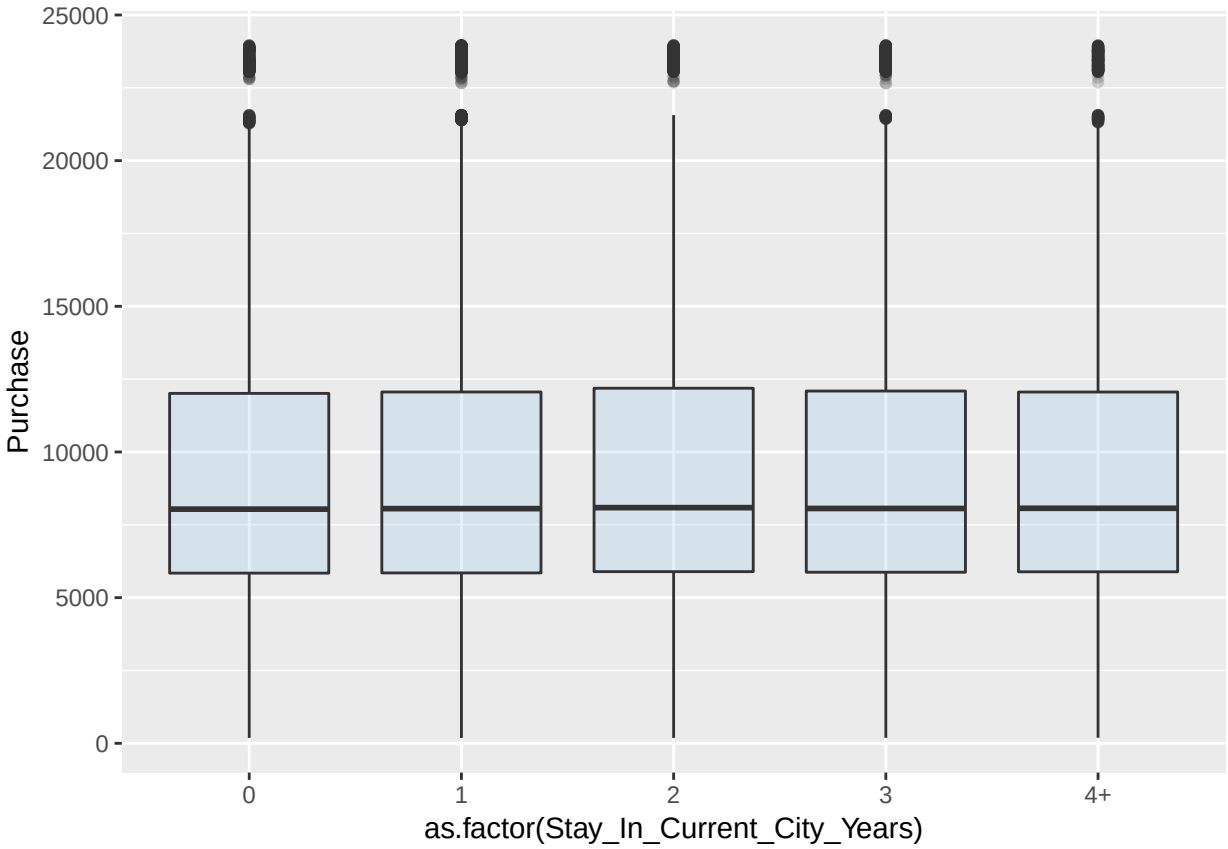

```
ggplot(data, aes(x=as.factor(Gender), y=Purchase)) +
  geom_boxplot(fill="skyblue2", alpha=0.2)
```



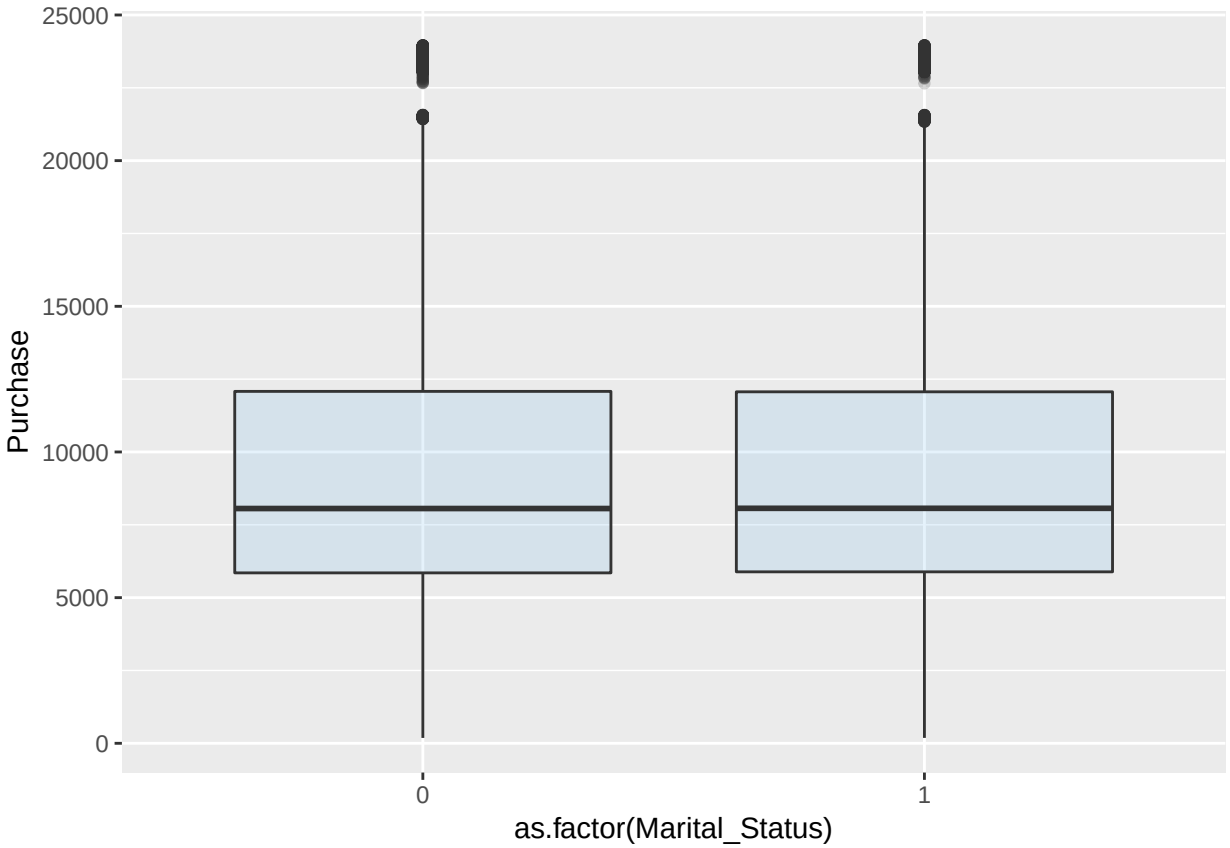

```
ggplot(data, aes(x=as.factor(City_Category), y=Purchase)) +
  geom_boxplot(fill="skyblue2", alpha=0.2)
```



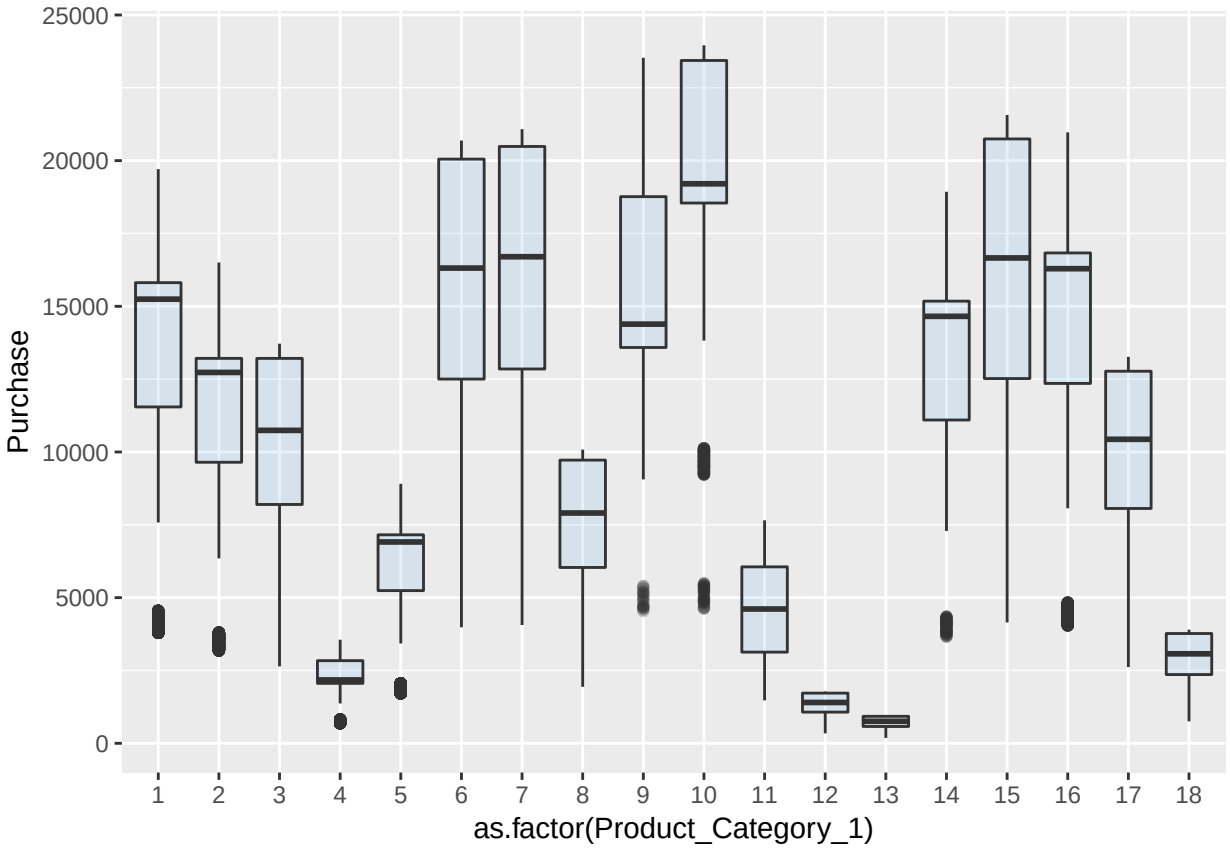

```
ggplot(data, aes(x=as.factor(Stay_In_Current_City_Years), y=Purchase)) +
  geom_boxplot(fill="skyblue2", alpha=0.2)
```



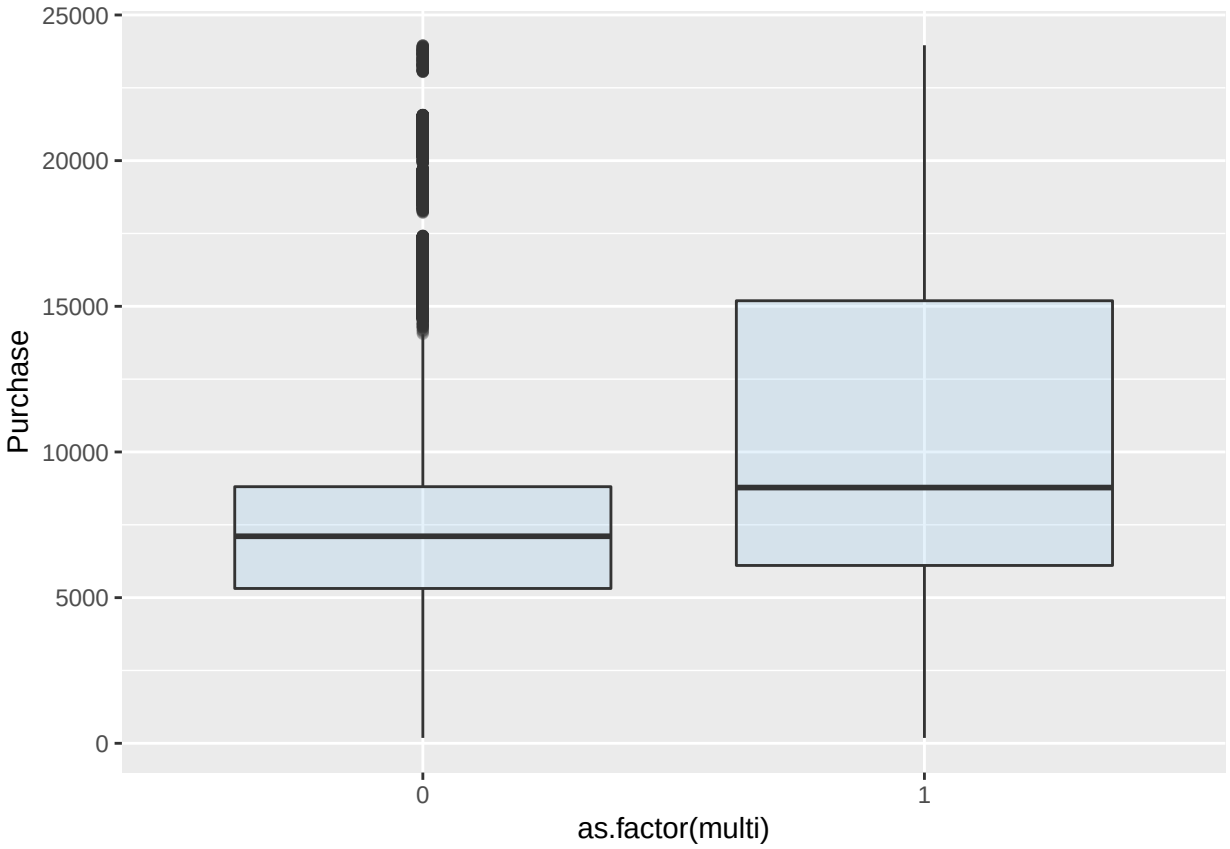

```
ggplot(data, aes(x=as.factor(Marital_Status), y=Purchase)) +
  geom_boxplot(fill="skyblue2", alpha=0.2)
```




```
ggplot(data, aes(x=as.factor(Product_Category_1), y=Purchase)) +
  geom_boxplot(fill="skyblue2", alpha=0.2)
```




```
ggplot(data, aes(x=as.factor(multi), y=Purchase)) +
  geom_boxplot(fill="skyblue2", alpha=0.2)
```



We make the following ooservations: 1. The mean purchase amount is fairly evenly distributed across the specific category levels within each variable except for Product Category and multi.

2. We notice that there is a larger spread in purchase amount when the product belongs to only one category.
3. Different product categories have different purchase amount means with different spreads.

Now that we have looked at some of the univariate and bivariate characteristics, we create a regression model with all of the additive terms.

```
mod.1 <- lm(Purchase ~ Gender + Age + Occupation +  City_Category+Stay_In_Current_City_Years +
              Marital_Status+Product_Category_1 + multi, data)
S(mod.1)
```

```
## Call: lm(formula = Purchase ~ Gender + Age + Occupation + City_Category +
##          Stay_In_Current_City_Years + Marital_Status + Product_Category_1 + multi,
##          data = data)
##
## Coefficients:
##                               Estimate Std. Error  t value Pr(>|t|)
## (Intercept)                 9132.1573    48.2767  189.163  < 2e-16 ***
## GenderM                      523.1621    14.9960   34.887  < 2e-16 ***
## Age18-25                     349.5520    41.7249    8.378  < 2e-16 ***
## Age26-35                     543.4284    40.5208   13.411  < 2e-16 ***
## Age36-45                     633.9828    41.6525   15.221  < 2e-16 ***
## Age46-50                     606.6089    45.7437   13.261  < 2e-16 ***
## Age51-55                     934.5792    46.7410   19.995  < 2e-16 ***
```

```
## Age55+                        739.0741    51.2880   14.410  < 2e-16 ***
## Occupation                      6.1314     0.9947    6.164 7.10e-10 ***
## City_CategoryB                162.8656    15.8793   10.256  < 2e-16 ***
## City_CategoryC                717.7012    17.1714   41.796  < 2e-16 ***
## Stay_In_Current_City_Years1    28.5154    20.5123    1.390  0.16448
## Stay_In_Current_City_Years2    62.9861    22.8902    2.752  0.00593 **
## Stay_In_Current_City_Years3    27.0766    23.2618    1.164  0.24443
## Stay_In_Current_City_Years4+   44.4407    23.8489    1.863  0.06240 .
## Marital_Status                -53.9095    13.8472   -3.893 9.89e-05 ***
## Product_Category_1           -355.3684     1.8840 -188.628  < 2e-16 ***
## multi                       1139.8707    15.2280   74.854  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard deviation: 4685 on 537559 degrees of freedom
## Multiple R-squared: 0.1153
## F-statistic:  4120 on 17 and 537559 DF,  p-value: < 2.2e-16
##      AIC      BIC
## 10612982 10613194
```

Since Stay_In_Current_City was not significant (except for Stay_In_Current_City_2) we use a Chow-Test to determine if the estimated coefficient of any of the "Stay in Current City" variables are equal to zero.

```
hyp <- c("Stay_In_Current_City_Years1 = 0", "Stay_In_Current_City_Years2 = 0",
         "Stay_In_Current_City_Years3 = 0", "Stay_In_Current_City_Years4+ = 0")
linearHypothesis(mod.1, hyp)
```

```
## Linear hypothesis test
##
## Hypothesis:
## Stay_In_Current_City_Years1 = 0
## Stay_In_Current_City_Years2 = 0
## Stay_In_Current_City_Years3 = 0
## Stay_In_Current_City_Years4  +  = 0
##
## Model 1: restricted model
## Model 2: Purchase ~ Gender + Age + Occupation + City_Category + Stay_In_Current_City_Years +
##     Marital_Status + Product_Category_1 + multi
##
##   Res.Df      RSS Df Sum of Sq      F  Pr(>F)
## 1 537563 1.18e+13
## 2 537559 1.18e+13  4 186223076 2.1209 0.07539 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Because the p-value is greater than 0.05, there is insufficient evidence to justify keeping "Stay in Current City" in the model. We remove this in the following model.

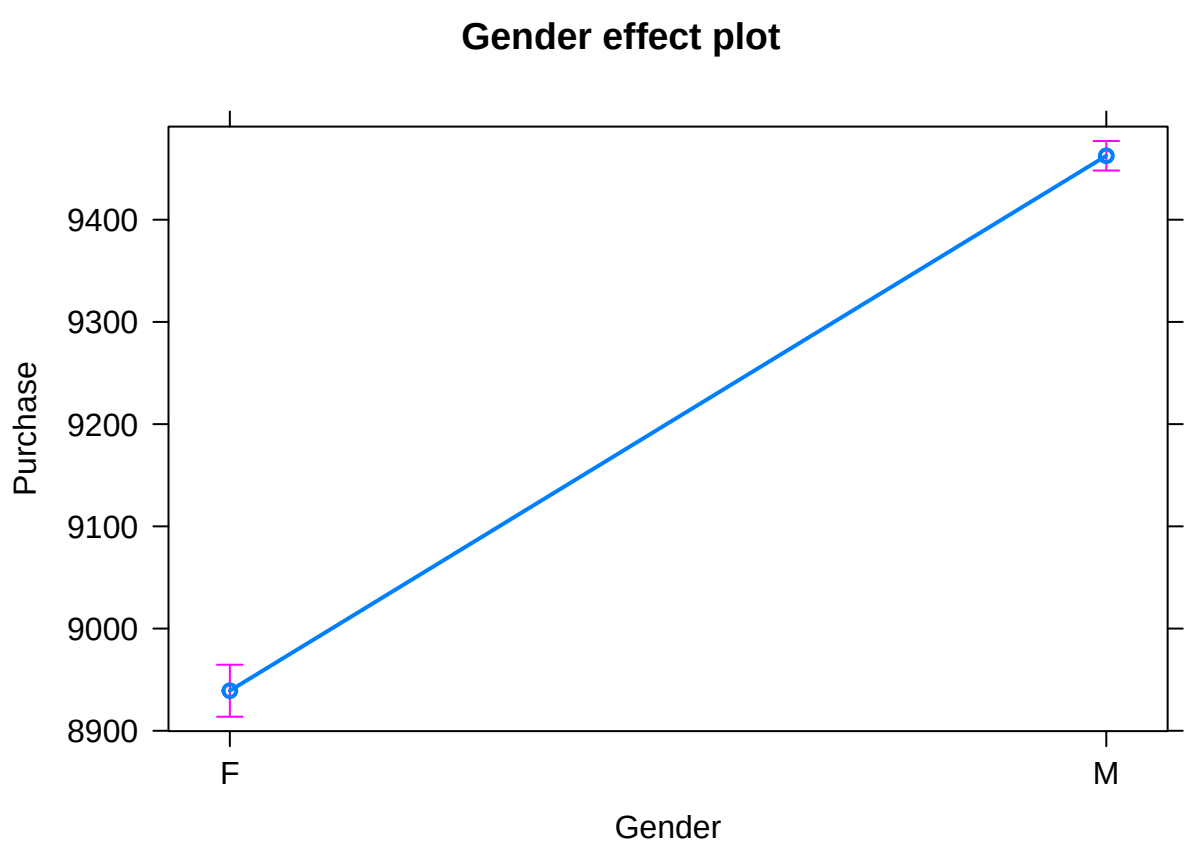
```
mod.2 <- lm(Purchase ~ Gender + Age + Occupation + City_Category +
              Marital_Status+Product_Category_1 + multi, data)
S(mod.2)
```

```
## Call: lm(formula = Purchase ~ Gender + Age + Occupation + City_Category +
##          Marital_Status + Product_Category_1 + multi, data = data)
##
## Coefficients:
```

```
##                   Estimate Std. Error  t value Pr(>|t|)
## (Intercept)       9164.473     45.747  200.330  < 2e-16 ***
## GenderM            523.371     14.983   34.932  < 2e-16 ***
## Age18-25           348.786     41.709    8.362  < 2e-16 ***
## Age26-35           543.423     40.500   13.418  < 2e-16 ***
## Age36-45           633.682     41.641   15.218  < 2e-16 ***
## Age46-50           605.493     45.716   13.245  < 2e-16 ***
## Age51-55           933.942     46.696   20.001  < 2e-16 ***
## Age55+             738.935     51.273   14.412  < 2e-16 ***
## Occupation           6.151      0.994    6.188 6.11e-10 ***
## City_CategoryB     163.903     15.859   10.335  < 2e-16 ***
## City_CategoryC     719.382     17.154   41.937  < 2e-16 ***
## Marital_Status     -53.926     13.846   -3.895 9.83e-05 ***
## Product_Category_1 -355.386      1.884 -188.655  < 2e-16 ***
## multi              1140.011     15.228   74.863  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard deviation: 4685 on 537563 degrees of freedom
## Multiple R-squared: 0.1153
## F-statistic:  5387 on 13 and 537563 DF,  p-value: < 2.2e-16
##      AIC      BIC
## 10612982 10613150
```

Now that all the variables are statistically significant, we look at the effects plots. Our R^2 remains the same.

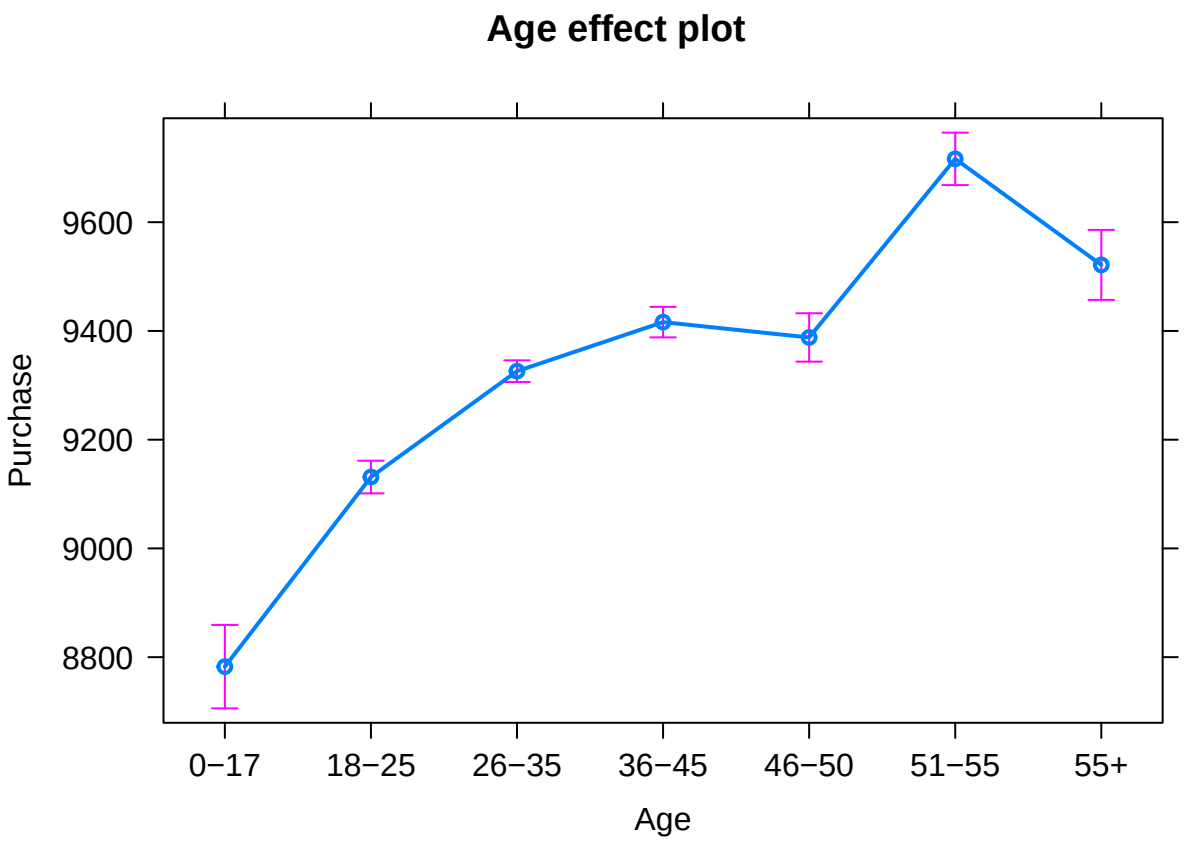
```
plot(effect(mod = mod.2, "Gender"))
```



From the Gender effect plot, we have that Males spent more on Black Friday than Females. The spread on Purchases for Males is also smaller than the spread on Purchases for Females.

Intuitively, this may be a result of Males buying more expensive items on Black Friday than females.

```
plot(effect(mod = mod.2, "Age"))
```

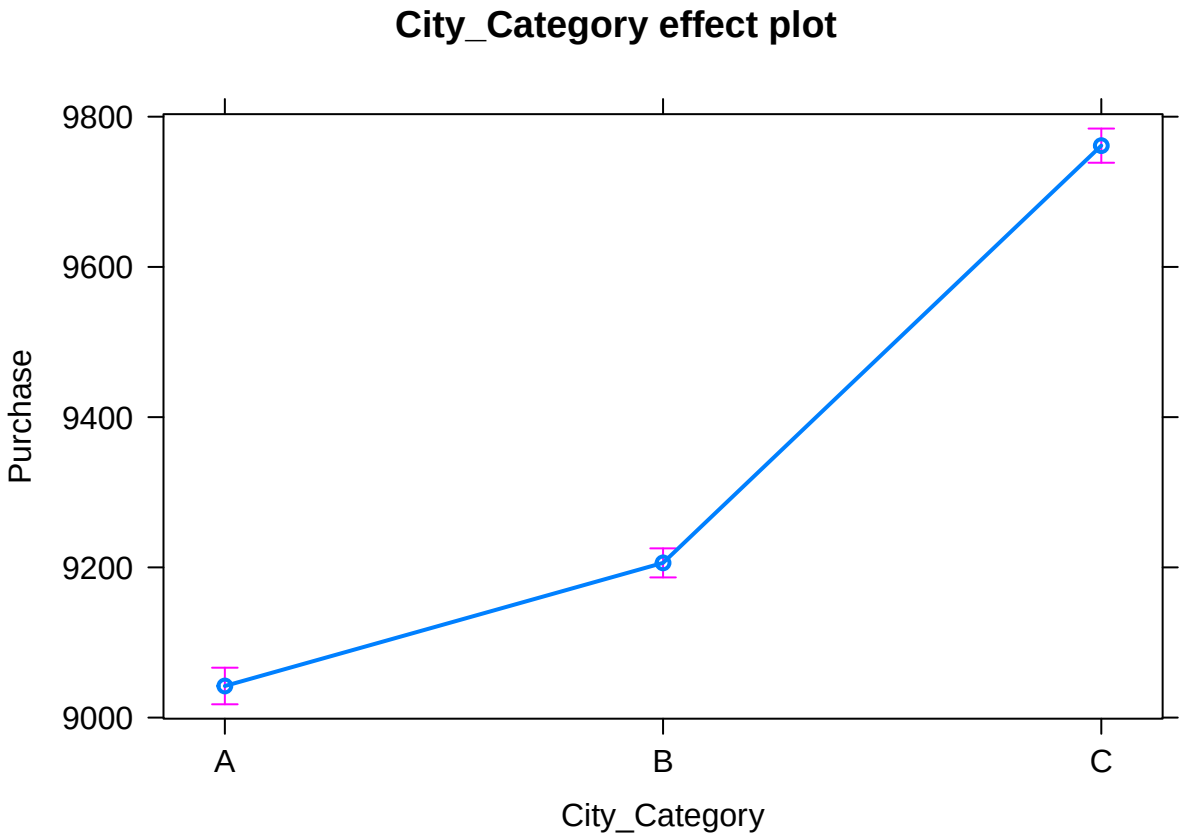


Here we see that in general, purchases go up as age increases, with more variability amongst the lower and higher age ranges.

However, we do see that purchases plateay between the age groups of 36-45 and 46-50.

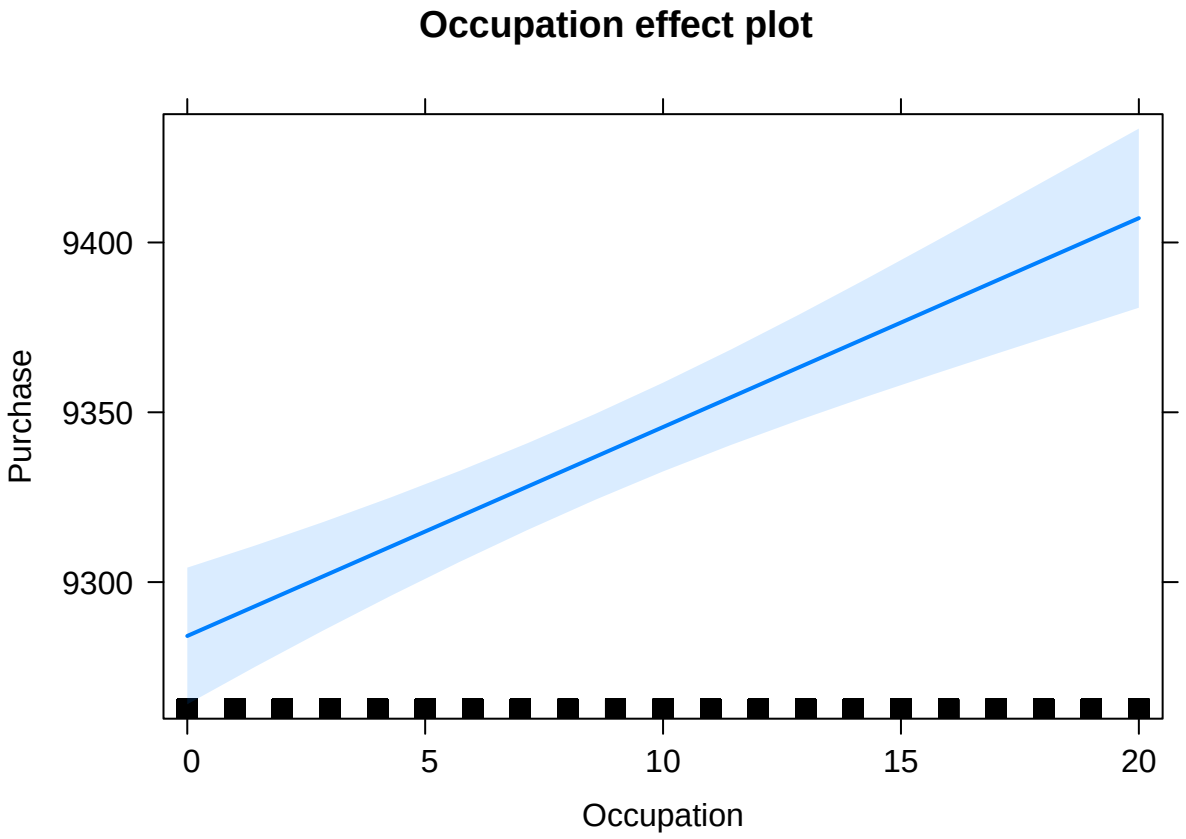
Intuitively, we believe the variability is due to younger age groups because they could be spending either their money or their parents' money. For the older age groups the variability could be due to retirees in this group with lower disposable income.

```
plot(effect(mod = mod.2, "City_Category"))
```



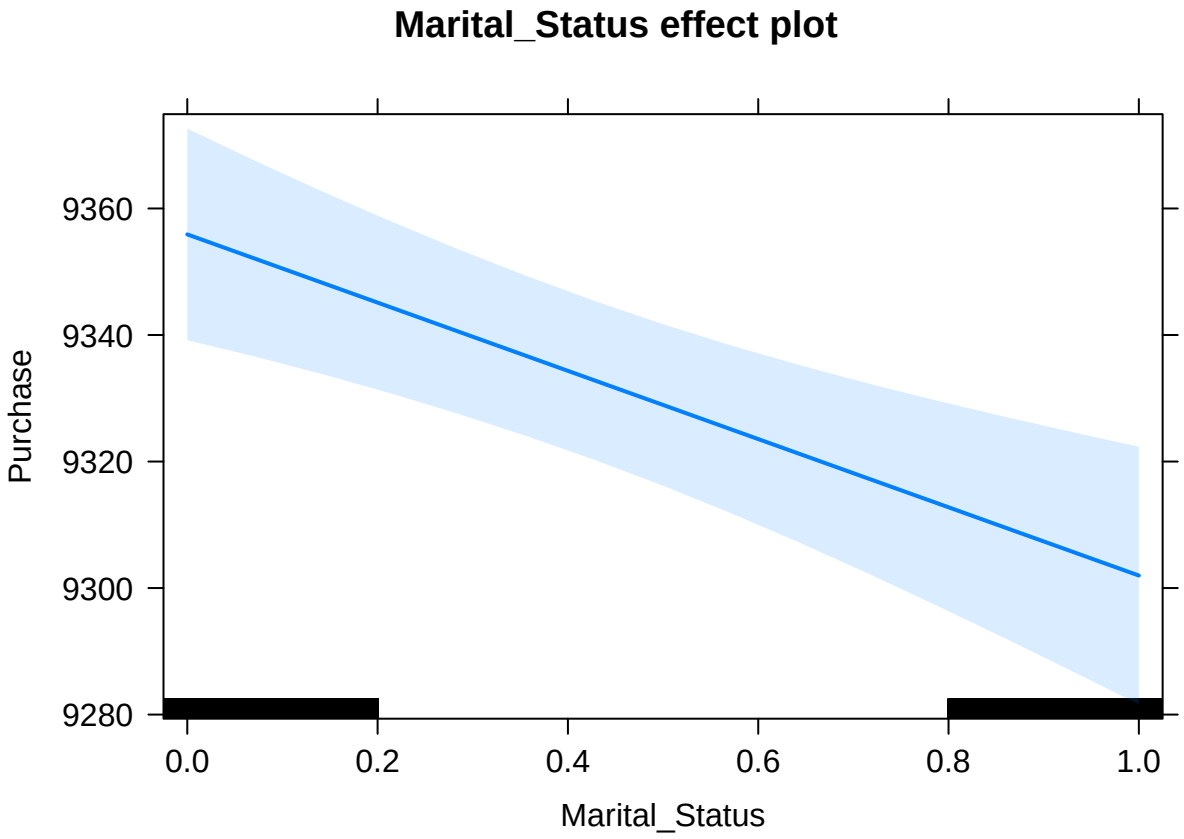
In City Category, we can see that the overall dollar value of purchases made in City C were much higher than B and C. This could mean that items are more expensive in city C or there is more variety (people shop more).

```
plot(effect(mod = mod.2, "Occupation"))
```



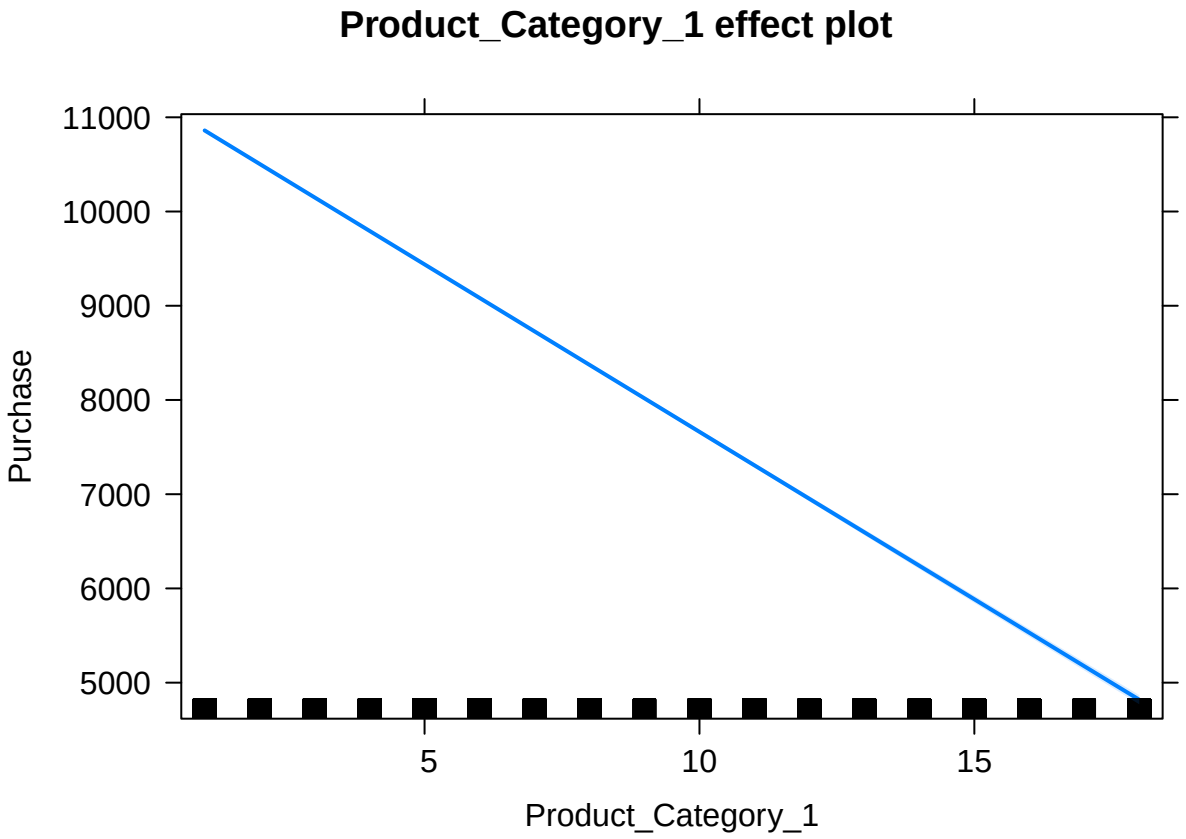
The Occupation effect plot shows us that as the Occupation category increases, then purchases increase. This could indicate that the larger the occupation category, the higher the income. If this were the case, intutively, it makes sense that there is a little more variation at the lower and higher occupation category values. This is because at lower levels of income, you are likely to have a different spending behaviour than other people in the same income bracket.

```
plot(effect(mod = mod.2, "Marital_Status"))
```



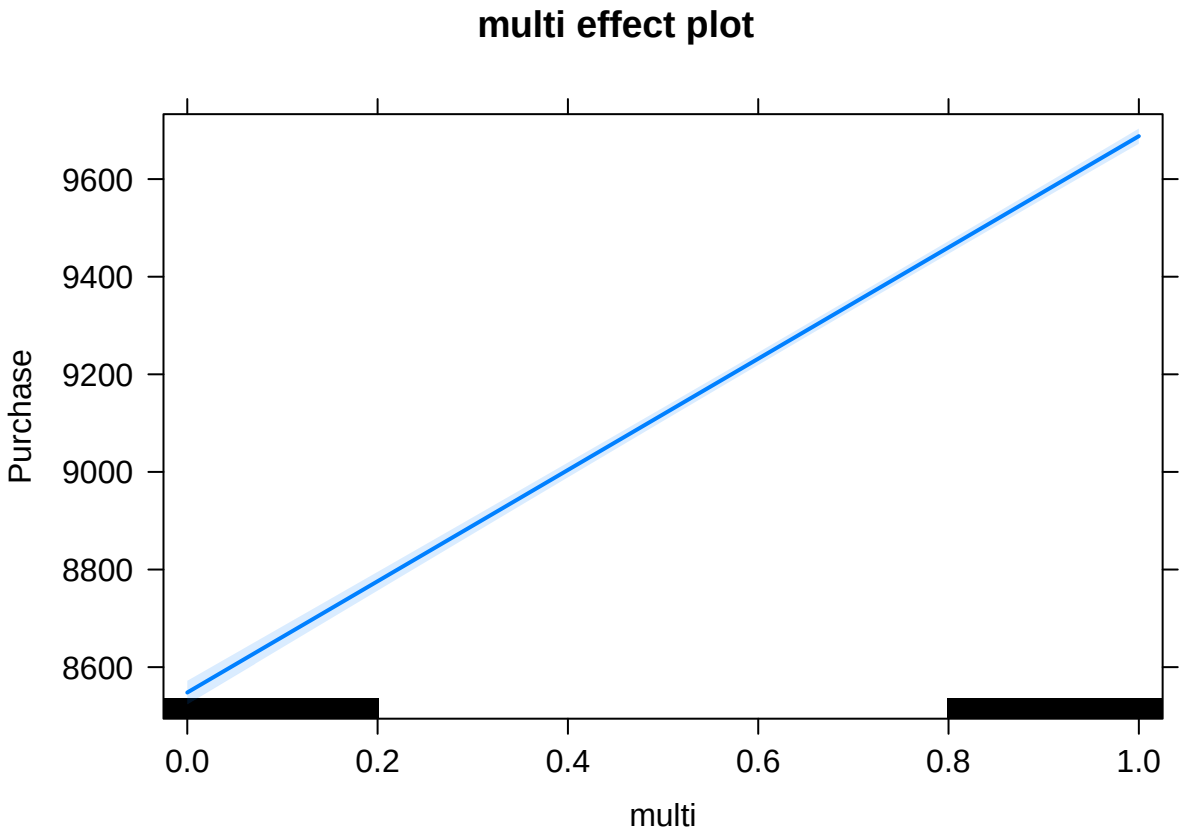
From this effects plot, it seems that purchases are lower for married individuals versus single individuals.

```
plot(effect(mod = mod.2, "Product_Category_1"))
```



For Product Category_1, we see that the higher the number of the category, the lower the purchase dollar amount. This could mean that higher category numbers are items that are cheaper or that less people buy them.

```
plot(effect(mod = mod.2, "multi"))
```



This plot shows that if a product belongs to more than one category then, the dollar value of purchases increase. This could indicated that the items hold more value if they belong to multiple categories or they are items that are purchased more.

From the effects plots, we noticed that Gender and multi looked almost identical and wanted to test if there was any degree of collinearity between the two variables. In order to do this, we look at their correlation.

```
cor(multi, (as.numeric(Gender))-1)
```

```
## [1] 0.01197696
```

Because the result is so low we can safely assume that our multi and Gender variables are significantly different from each other.

Now that we have the effects plots, we use the Ramsey RESET test in order to determine whether or not our current model is misspecified.

```
resettest(mod.2, power=2, type="regressor")
```

```
##
##  RESET test
##
## data:  mod.2
## RESET = 24835, df1 = 4, df2 = 537560, p-value < 2.2e-16
```