Explorartory Data Excercse Project 1

Curtis ONeal

May 3, 2016

The following report outlines Explorartory Data Excercse (EDA) of the example Diamonds Dataset, specifically looking the relationship between the weight of the diamonds in carats to their price, without using a linear model to describe that relationship.

The report will follow this general outline: R code will be provided, with output as tables or charts, and a narrative passage of observations will follow, or will be interspersed with the output as comments.

# Dataset Description

The Diamonds data set has the following Variables: "carat" "cut" "color" "clarity" "depth" and "price."

These unused variables were removed:"table" "x" "y" "z"

#The Strucutre command tells us the number of obsevations, and details about what the variables are, their kinds, and some sample values.  
str(pf)

## Classes 'tbl\_df', 'tbl' and 'data.frame': 53940 obs. of 6 variables:  
## $ carat : num 0.23 0.21 0.23 0.29 0.31 0.24 0.24 0.26 0.22 0.23 ...  
## $ cut : Ord.factor w/ 5 levels "Fair"<"Good"<..: 5 4 2 4 2 3 3 3 1 3 ...  
## $ color : Ord.factor w/ 7 levels "D"<"E"<"F"<"G"<..: 2 2 2 6 7 7 6 5 2 5 ...  
## $ clarity: Ord.factor w/ 8 levels "I1"<"SI2"<"SI1"<..: 2 3 5 4 2 6 7 3 4 5 ...  
## $ depth : num 61.5 59.8 56.9 62.4 63.3 62.8 62.3 61.9 65.1 59.4 ...  
## $ price : int 326 326 327 334 335 336 336 337 337 338 ...

The Diamonds data set has 53,940 obs (rows). After removing 4 unused variables (colums), it has 6 remaining.

The Price variable is a numeral in Dollars. It is seen as dependent on the other numerical and ordered-categorical factors of the other variables, that describe the qualities of the diamond. The categorical values are seen as having a range of qualities that get better on a scale - so they are ordered.

The following are the ordered categorical (factor) variables, with the number of their categories (levels): Cut (5), Clarity(7), and Color(8).

Depth and Carat, are numerical variables.

Each observation is a diamond with each of these graded factors listed in columns. This exploration looks for a relationship in the set between weight as Carat, and Price, and a general distribution of the factors in the dataset.

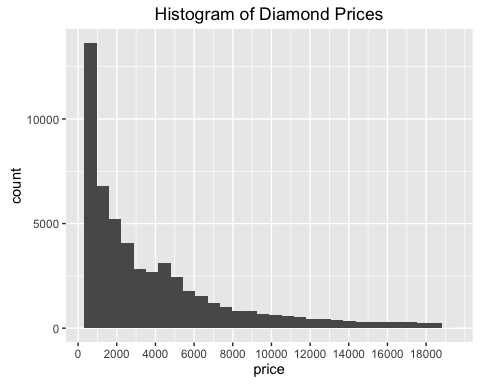
# Examining Price:

#This code produces a summary of diamond Prices  
summary(pf$price)

## Min. 1st Qu. Median Mean 3rd Qu. Max.   
## 326 950 2401 3933 5324 18820

#This code produces a historgram of diamond Prices  
qplot(x =price, data=pf, main = 'Histogram of Diamond Prices')+ scale\_x\_continuous(breaks = seq(0, 18820, 2000))

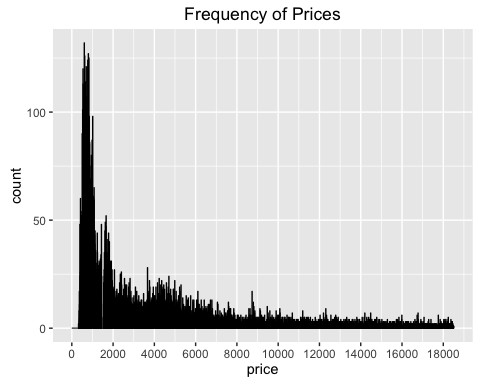
## `stat\_bin()` using `bins = 30`. Pick better value with `binwidth`.



#This code produces a historgram of diamond Prices  
qplot(x = price, data= pf, binwidth = .1, geom = 'freqpoly', main = 'Frequency of Prices')+  
 scale\_x\_continuous(lim = c(0,18500), breaks = seq(0, 18500, 2000)) +  
 scale\_y\_continuous(limits = c(0,NA))

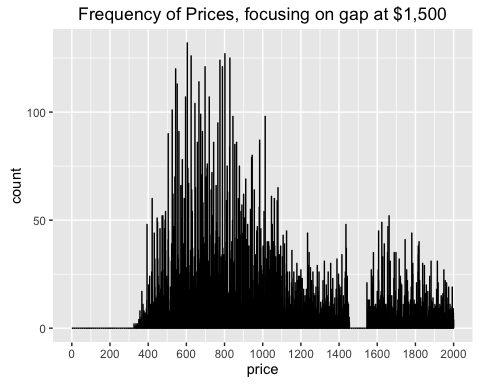
## Warning: Removed 102 rows containing non-finite values (stat\_bin).

## Warning: Removed 2 rows containing missing values (geom\_path).



#This code produces a historgram of diamond Prices  
qplot(x = price, data= pf, binwidth = .1, geom = 'freqpoly', main = 'Frequency of Prices, focusing on gap at $1,500')+  
 scale\_x\_continuous(lim = c(0,2000), breaks = seq(0, 2000, 200)) +  
 scale\_y\_continuous(limits = c(0,NA))

## Warning: Removed 29733 rows containing non-finite values (stat\_bin).  
  
## Warning: Removed 2 rows containing missing values (geom\_path).



The price histogram is right skewed with the largest number of samples in the "$1,000 and under," category. The average price is $3,933 dollars, and the Median (middle) price is $2,401. The prices in the set range from $326 to $18,820.

In Frequency distribution shows an odd gap of missing values between $1,450 and $1,550 that should cause us to examine the data further.

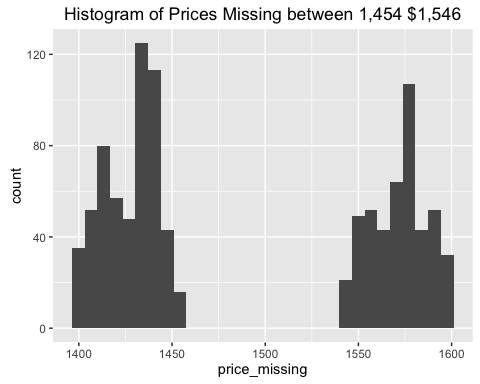
# Mising Price Frequencies

#Collect, and sort ascending the unique values of price between $1,400 and $1,600  
p<-sort(unique(subset(pf$price, (pf$price < 1600 & pf$price > 1400), drop = TRUE)))  
p

## [1] 1401 1402 1403 1404 1405 1406 1407 1408 1409 1410 1411 1412 1413 1414  
## [15] 1415 1416 1417 1418 1419 1420 1421 1422 1423 1424 1425 1426 1427 1428  
## [29] 1429 1430 1431 1432 1433 1434 1435 1436 1437 1438 1439 1440 1441 1442  
## [43] 1443 1444 1445 1446 1447 1448 1449 1450 1451 1452 1453 1454 1546 1547  
## [57] 1548 1549 1550 1551 1552 1553 1554 1555 1556 1557 1558 1559 1560 1561  
## [71] 1562 1563 1564 1565 1566 1567 1568 1569 1570 1571 1572 1573 1574 1575  
## [85] 1576 1577 1578 1579 1580 1581 1582 1583 1584 1585 1586 1587 1588 1589  
## [99] 1590 1591 1592 1593 1594 1595 1596 1597 1598 1599

#Put these into a vector  
price\_missing <- subset(pf$price, (pf$price < 1600 & pf$price > 1400), drop = TRUE)  
#Table was not useful   
  
#Create a histogram of these valuesof price between $1,400 and $1,600   
qplot(price\_missing, main="Histogram of Prices Missing between 1,454 $1,546")

## `stat\_bin()` using `bins = 30`. Pick better value with `binwidth`.



A gap exists for values of price between $1,454 and $1,546 inclusive.

# Examining Carat:

#"How many values does Carat take?"  
length(unique(pf$carat))

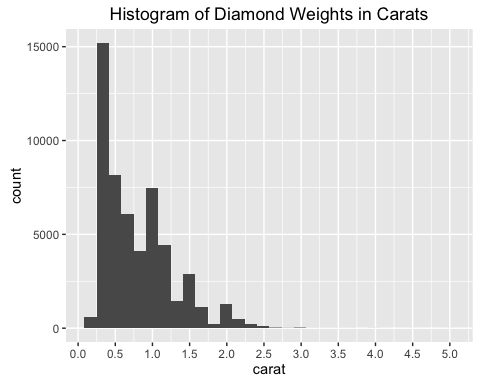
## [1] 273

#A descriptive statistical sumary of the values   
summary(pf$carat)

## Min. 1st Qu. Median Mean 3rd Qu. Max.   
## 0.2000 0.4000 0.7000 0.7979 1.0400 5.0100

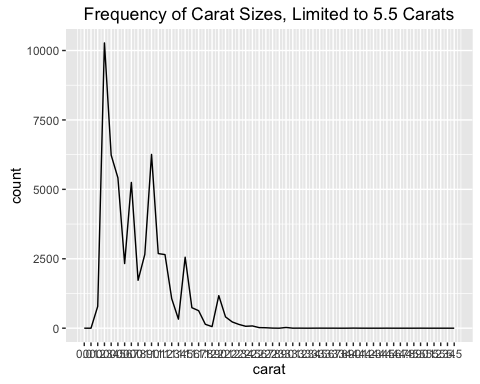
#table(pf$carat) #Produces a table of counts by the 273 values  
  
#What values does Carat take?"  
#sort(unique(pf$carat))  
#This commented out code produces a sorted list of the 273 unique values that are too many - to view or to make it a factor.  
  
#This code produces a historgram of Carat Values  
qplot(x =carat, data=pf, main = 'Histogram of Diamond Weights in Carats') + scale\_x\_continuous(breaks = seq(0, 5.5, .5))

## `stat\_bin()` using `bins = 30`. Pick better value with `binwidth`.



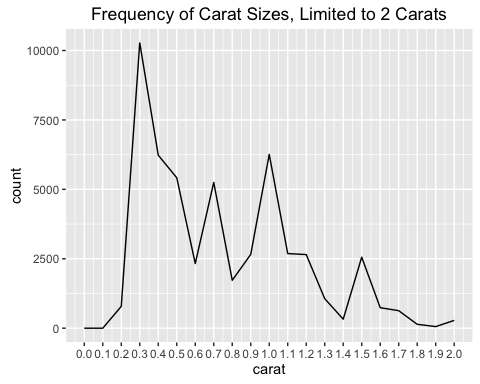
#A histogram of the carat values. Note, although there are a few in the 5.0 #range there are too few to see at this scale.  
  
#This code produces Frequency plots of Carat Values  
qplot(x = carat, data= pf, binwidth = .1, geom = 'freqpoly', main = 'Frequency of Carat Sizes, Limited to 5.5 Carats')+  
 scale\_x\_continuous(lim = c(0,5.5), breaks = seq(0, 5.5, .1))

## Warning: Removed 2 rows containing missing values (geom\_path).



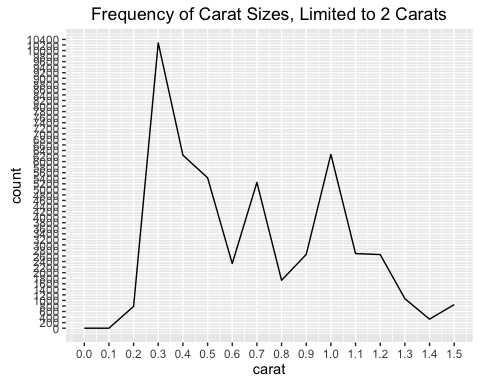
qplot(x = carat, data= pf, binwidth = .1, geom = 'freqpoly', main = 'Frequency of Carat Sizes, Limited to 2 Carats')+  
 scale\_x\_continuous(lim = c(0,2), breaks = seq(0, 2, .1))

## Warning: Removed 1889 rows containing non-finite values (stat\_bin).  
  
## Warning: Removed 2 rows containing missing values (geom\_path).



qplot(x = carat, data= pf, binwidth = .1, geom = 'freqpoly', main = 'Frequency of Carat Sizes, Limited to 2 Carats')+  
 scale\_x\_continuous(lim = c(0,1.5), breaks = seq(0, 5.5, .1)) +  
 scale\_y\_continuous(limits = c(0,NA), breaks = seq(0, 10500, 200))

## Warning: Removed 5442 rows containing non-finite values (stat\_bin).  
  
## Warning: Removed 2 rows containing missing values (geom\_path).



# May need to use zoom for .8

In the histogram we see that Carat weights are continuos number ranges and are not confined by convention to a set series of increments such as to the .2, or .5. There are too many in the data set to create a number of factors - for other tablulation summaries.

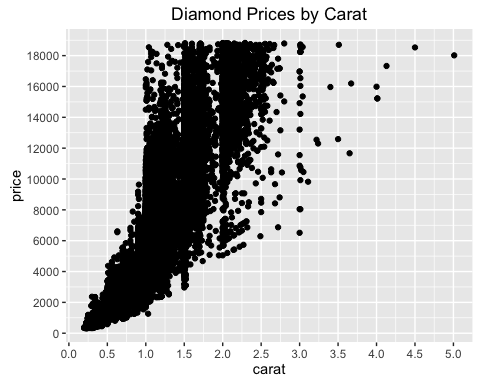
The summary statistics are given above in the above code. Carats range from .2 to 5.01, with a median of .7, and a mean of .7979.

The carat distribution is non-normal and skewed slightly right. The long tail in the r carat range is too small to see at this scale.

The most numerous range of the sample is in the .2 to .5 carat range. But 50% of the sample is between the range of 0.4 to 1.04 Carats.

# Examining Price and Carat Together:

#This code produces a scatter plot of Prices by Carat  
qplot(x =carat, y=price, data=pf, main = 'Diamond Prices by Carat') +  
 scale\_x\_continuous(breaks = seq(0, 5.5, .5)) +  
 scale\_y\_continuous(breaks = seq(0, 18820, 2000))



The above is a scatter plot of Prices by Carat, and a general linear relationship can be seen as carat increases typically price does also.

Many samples are outside of a linear fit line, producing a wide path, suggesting these other factors not inlcuded in this plot affect the price ranking.

There is a noticable pattern at the 1.0, 1.5, and 2.0 lines suggesting that diamonds are cut for these as minimum sizes.

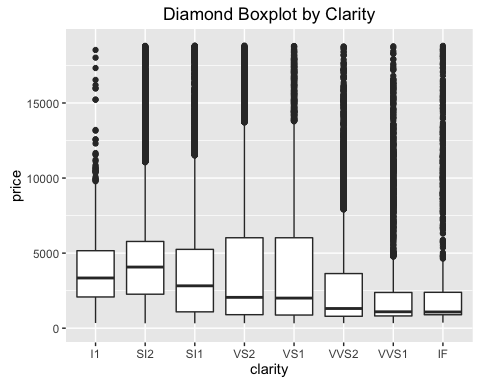
A linear regression would be run by the following code, but is not assigned in this report: CaratFirst <- lm(pfprice) PriceFirst <- lm(pfcarat)

# Examining Clarity

#This code gives the summary statistics for Clarity  
summary(pf$clarity)

## I1 SI2 SI1 VS2 VS1 VVS2 VVS1 IF   
## 741 9194 13065 12258 8171 5066 3655 1790

#This code gives a Boxplot of Clarity values  
qplot(x = clarity, y= price,   
 data= pf,   
 geom = "boxplot", main = 'Diamond Boxplot by Clarity') +  
 scale\_y\_continuous(limits = c(0, 19000))

 Clarity Category Values Ordered Worst to best --> I1 SI2 SI1 VS2 VS1 VVS2 VVS1 IF 741 9194 13065 12258 8171 5066 3655 1790

Each category has the above counts in the dataset.

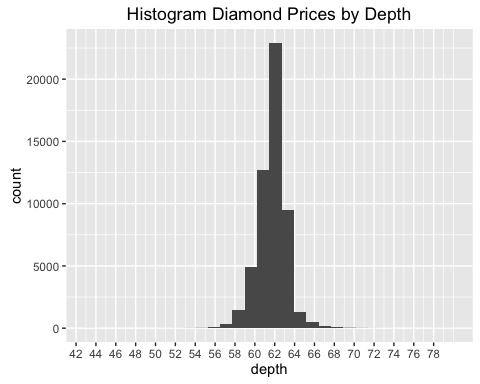
# Examining Depth

#This code gives the summary statistics for Depth  
summary(pf$depth)

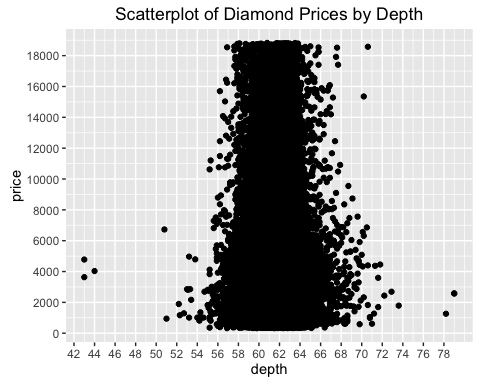
## Min. 1st Qu. Median Mean 3rd Qu. Max.   
## 43.00 61.00 61.80 61.75 62.50 79.00

#This code gives the Histogram of Prices for Depth  
qplot(x =depth, data=pf, main = 'Histogram Diamond Prices by Depth') + scale\_x\_continuous(breaks = seq(40, 79, 2))

## `stat\_bin()` using `bins = 30`. Pick better value with `binwidth`.



#This code produces a scatterplot for Prices by Depth  
qplot(x =depth, y=price, data=pf, main = 'Scatterplot of Diamond Prices by Depth') +  
 scale\_x\_continuous(breaks = seq(40, 79, 2)) +  
 scale\_y\_continuous(breaks = seq(0, 18820, 2000))



# Depth seems to have no real affect on price.

## Depth Values are Numeric Values

Depth is normally distributed centering on value 62, with 50% of the values between value 61 and 62.5.

From the plot we see that Depth seems to have almost no affect on price, since a diamond can have any depth value and be at the height or bottom of the price range.

It would be possible to creat a plot of all 273 incremental carat values in the sample, but this would not tell use very much that is useful.

# Examining Color

#This code gives the summary statistics for Color  
summary(pf$color)

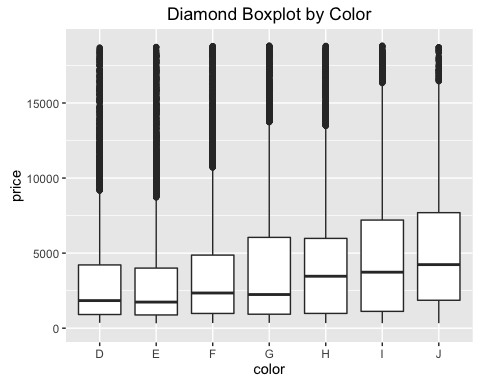
## D E F G H I J   
## 6775 9797 9542 11292 8304 5422 2808

Color Category Values Ordered Worst to Best --> D E F G H I J 6775 9797 9542 11292 8304 5422 2808

Each category has the above counts in the dataset.

# Examining Color and Price

#This code gives a boxplot of Color values  
qplot(x = color, y= price,   
 data= pf,   
 geom = "boxplot", main = 'Diamond Boxplot by Color') +  
 scale\_y\_continuous(limits = c(0, 19000))



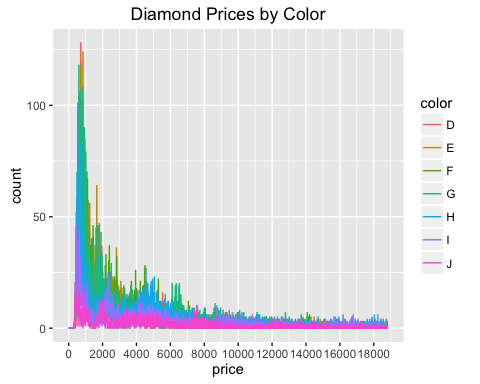
#Note the categorical value should be x, and continuous variable should be y #always true for boxplots

Using the Box plot of color it is easy to see that the median of the prices increase with the scale of the colors.

#This code gives a frequency plot of Prices, colored by Color values  
qplot(x = price, data= pf, binwidth = 10, geom = 'freqpoly', color = color, main = 'Diamond Prices by Color')+  
 scale\_x\_continuous(lim = c(0,18820), breaks = seq(0,18820, 2000))

## Warning: Removed 1 rows containing non-finite values (stat\_bin).

## Warning: Removed 14 rows containing missing values (geom\_path).

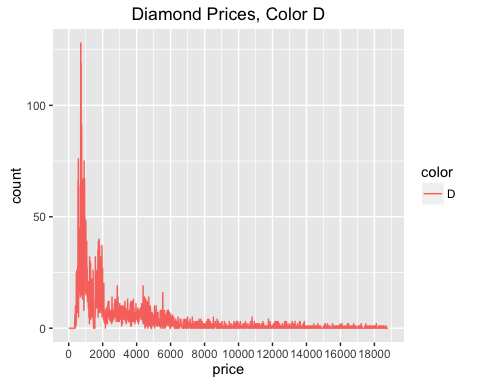


Since this frequency chart is so overplotted it may be useful to see separate freequency charts of the individual "color" categories to see if they show any patterns in prices or quantity.

# Prices by Individual Colors

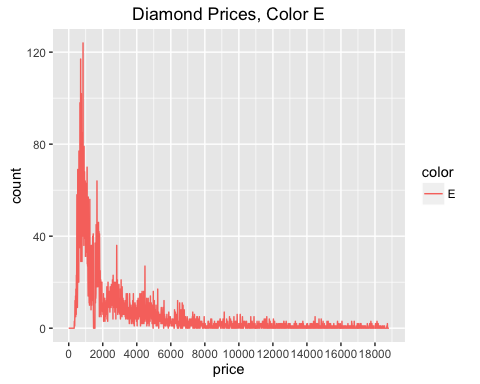
#The below code gives, A Frequency Plot for Each Color  
select\_d <- subset( pf, color == "D", select = c(color, price), drop = TRUE)  
only\_d<- qplot(x = price, data= select\_d, binwidth = 10, geom = 'freqpoly', color = color, main = 'Diamond Prices, Color D')+  
 scale\_x\_continuous(lim = c(0,18820), breaks = seq(0,18820, 2000))  
only\_d

## Warning: Removed 2 rows containing missing values (geom\_path).



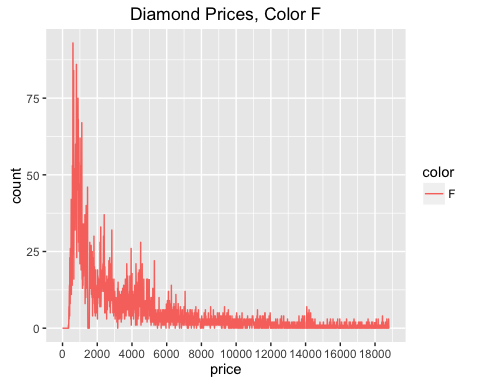
select\_e <- subset( pf, color == "E", select = c(color, price), drop = TRUE)  
only\_e<- qplot(x = price, data= select\_e, binwidth = 10, geom = 'freqpoly', color = color, main = 'Diamond Prices, Color E')+  
 scale\_x\_continuous(lim = c(0,18820), breaks = seq(0,18820, 2000))  
only\_e

## Warning: Removed 2 rows containing missing values (geom\_path).



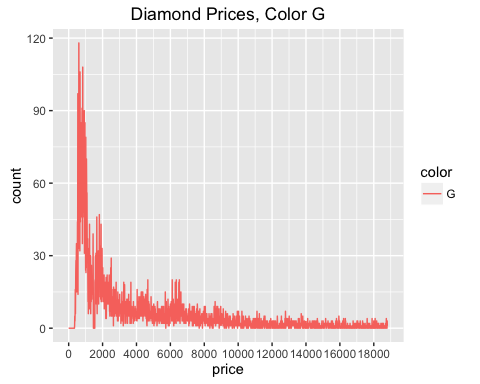
select\_f <- subset( pf, color == "F", select = c(color, price), drop = TRUE)  
only\_f<- qplot(x = price, data= select\_f, binwidth = 10, geom = 'freqpoly', color = color, main = 'Diamond Prices, Color F')+  
 scale\_x\_continuous(lim = c(0,18820), breaks = seq(0,18820, 2000))  
only\_f

## Warning: Removed 2 rows containing missing values (geom\_path).



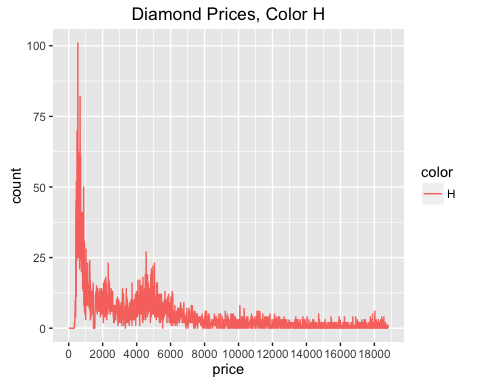
select\_g <- subset( pf, color == "G", select = c(color, price), drop = TRUE)  
only\_g<- qplot(x = price, data= select\_g, binwidth = 10, geom = 'freqpoly', color = color, main = 'Diamond Prices, Color G')+  
 scale\_x\_continuous(lim = c(0,18820), breaks = seq(0,18820, 2000))  
only\_g

## Warning: Removed 2 rows containing missing values (geom\_path).



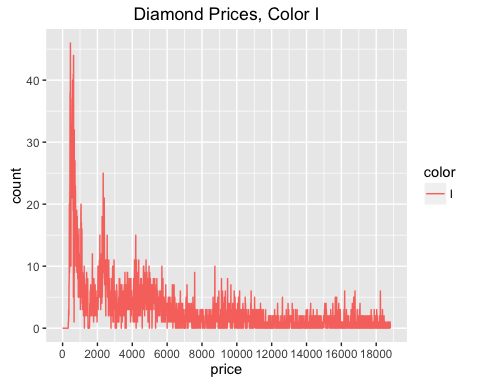
select\_h <- subset( pf, color == "H", select = c(color, price), drop = TRUE)  
only\_h<- qplot(x = price, data= select\_h, binwidth = 10, geom = 'freqpoly', color = color, main = 'Diamond Prices, Color H')+  
 scale\_x\_continuous(lim = c(0,18820), breaks = seq(0,18820, 2000))  
only\_h

## Warning: Removed 2 rows containing missing values (geom\_path).



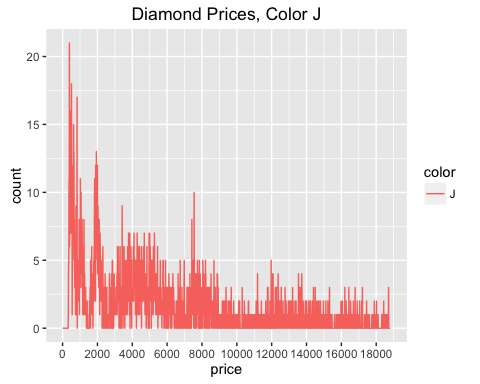
select\_i <- subset( pf, color == "I", select = c(color, price), drop = TRUE)  
only\_i<- qplot(x = price, data= select\_i, binwidth = 10, geom = 'freqpoly', color = color, main = 'Diamond Prices, Color I')+  
 scale\_x\_continuous(lim = c(0,18820), breaks = seq(0,18820, 2000))  
only\_i

## Warning: Removed 1 rows containing non-finite values (stat\_bin).  
  
## Warning: Removed 2 rows containing missing values (geom\_path).



select\_j <- subset( pf, color == "J", select = c(color, price), drop = TRUE)  
only\_j<- qplot(x = price, data= select\_j, binwidth = 10, geom = 'freqpoly', color = color, main = 'Diamond Prices, Color J')+  
 scale\_x\_continuous(lim = c(0,18820), breaks = seq(0,18820, 2000))  
only\_j

## Warning: Removed 2 rows containing missing values (geom\_path).



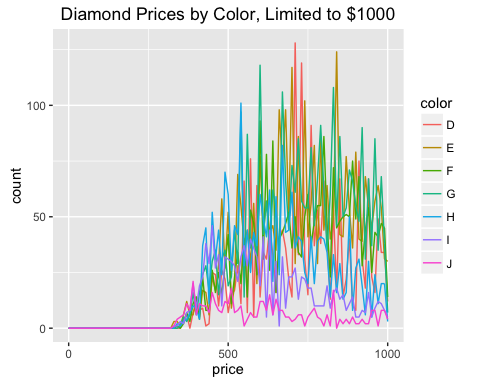
#grid.arrange(only\_d, only\_e, only\_f, only\_g, only\_h, only\_i, only\_j, ncol = 1, newpage = TRUE)   
#This code normally creates all the charts in a single column

The individual plots show that colors come in all prices, but that their scarcity roughly correseponds to the scale value, but also suggest that no specific color causes an automatic price increase.

#The below code gives, A Frequency Plot for All Colors, Zooming in to prices up to $1,000  
  
qplot(x = price, data= pf, binwidth = 10, geom = 'freqpoly', color = color, main = 'Diamond Prices by Color, Limited to $1000')+  
 scale\_x\_continuous(lim = c(0,1000), breaks = seq(0,1000, 500))

## Warning: Removed 39416 rows containing non-finite values (stat\_bin).

## Warning: Removed 14 rows containing missing values (geom\_path).



The full color graph of "Price Frequencies by Color, FLimited to $1,000" is the zoomed-in version of the full color graph.

From this chart we can infer three main ideas:

Although the Color feature ranked with J the best and D the lowest, these colornames/letters are found at all price ranges. For Example, a J color does not automatically mean that a Diamond is worth much more.

Looking at the chart that is cropped to a Price of $1,000, it is easier to see that the Color ranks do correspond roughly to how common each color class is. We see that J is the least common of the Colors, and the others are roughly ranked as more numerous or more commonly found the closer to the D end of the scale.

Finally, from the individiaul plots we can tell that the actual price range and distribution is very large for each color, althought certian colors are more common at certain price ranges the range of prices is broadly distributed in the color categories.

# More Price to Color Frequencies

#This code provides the table of Counts for Diamonds of this Color  
table(pf$color)

##   
## D E F G H I J   
## 6775 9797 9542 11292 8304 5422 2808

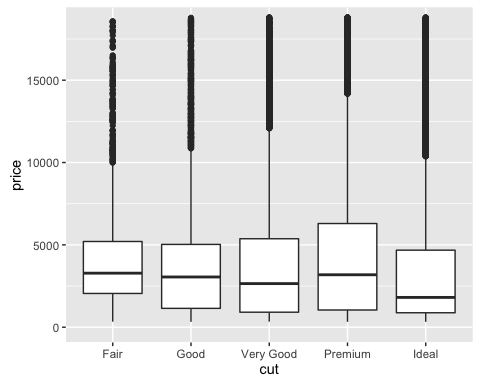
#This code provides the table of Sums for Diamond's Prices of this Color  
by(pf$price, pf$color, sum)

## pf$color: D  
## [1] 21476439  
## --------------------------------------------------------   
## pf$color: E  
## [1] 30142944  
## --------------------------------------------------------   
## pf$color: F  
## [1] 35542866  
## --------------------------------------------------------   
## pf$color: G  
## [1] 45158240  
## --------------------------------------------------------   
## pf$color: H  
## [1] 37257301  
## --------------------------------------------------------   
## pf$color: I  
## [1] 27608146  
## --------------------------------------------------------   
## pf$color: J  
## [1] 14949281

The above table lists the quantities of diamonds by color, and then sums the prices for all diamonds of that color.

## Examining the Cut Variable

qplot(x = pf$cut, y= price, xlab = "cut",  
 data= pf,   
 geom = "boxplot") +  
 coord\_cartesian(ylim = c(0, 18500))



Strangely the cuts, although graded on a scale from Fair to Ideal, do not seem to have a clearly positive relationship to increase in price.

In summary, several factors affect the price of a diamond in the dataset, with rare color, and size appearing as the two strongest predictors. But even the easiest relationship to see in size/carat is not necessarily fully predictive since there is considerable variation in the prices of the 2.5 - 5.0 carat price range. To determine a full understanding, a model or algorithm would need to be applied.