Audrey Ekuban

3 April 2016

```r
library(ggplot2)
data("diamonds")
summary(diamonds)
```

```
##      carat               cut        color        clarity     
##  Min.   :0.2000   Fair     : 1610   D: 6775   SI1    :13065  
##  1st Qu.:0.4000   Good     : 4906   E: 9797   VS2    :12258  
##  Median :0.7000   Very Good:12082   F: 9542   SI2    : 9194  
##  Mean   :0.7979   Premium  :13791   G:11292   VS1    : 8171  
##  3rd Qu.:1.0400   Ideal    :21551   H: 8304   VVS2   : 5066  
##  Max.   :5.0100                     I: 5422   VVS1   : 3655  
##                                     J: 2808   (Other): 2531  
##      depth           table           price             x         
##  Min.   :43.00   Min.   :43.00   Min.   :  326   Min.   : 0.000  
##  1st Qu.:61.00   1st Qu.:56.00   1st Qu.:  950   1st Qu.: 4.710  
##  Median :61.80   Median :57.00   Median : 2401   Median : 5.700  
##  Mean   :61.75   Mean   :57.46   Mean   : 3933   Mean   : 5.731  
##  3rd Qu.:62.50   3rd Qu.:59.00   3rd Qu.: 5324   3rd Qu.: 6.540  
##  Max.   :79.00   Max.   :95.00   Max.   :18823   Max.   :10.740  
##                                                                  
##        y                z         
##  Min.   : 0.000   Min.   : 0.000  
##  1st Qu.: 4.720   1st Qu.: 2.910  
##  Median : 5.710   Median : 3.530  
##  Mean   : 5.735   Mean   : 3.539  
##  3rd Qu.: 6.540   3rd Qu.: 4.040  
##  Max.   :58.900   Max.   :31.800  
## 
```

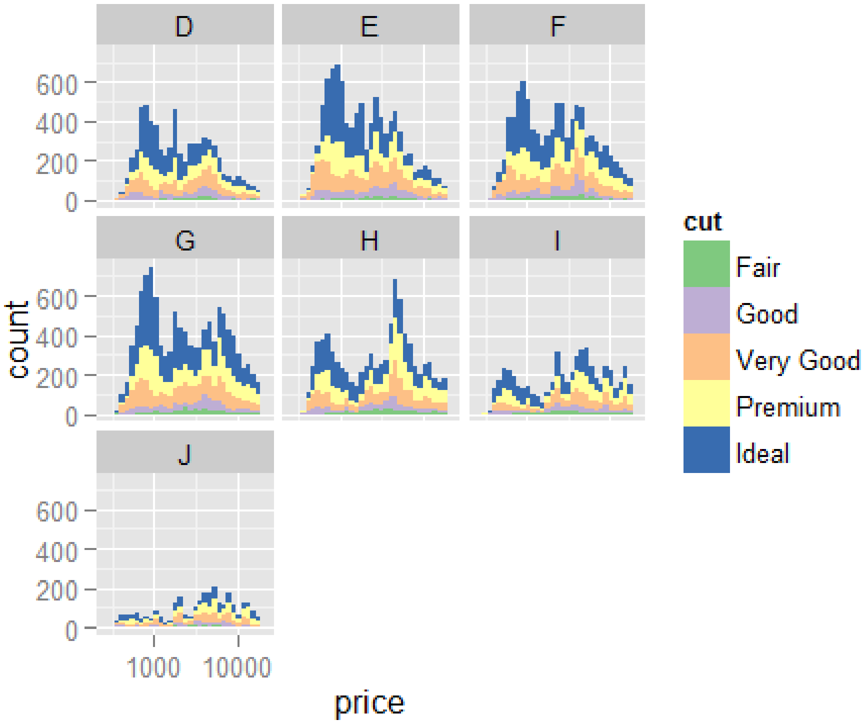
```r
str(diamonds)
```

```
## 'data.frame':    53940 obs. of  10 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 ...
##  $ table  : num  55 61 65 58 58 57 57 55 61 61 ...
##  $ price  : int  326 326 327 334 335 336 336 337 337 338 ...
##  $ x      : num  3.95 3.89 4.05 4.2 4.34 3.94 3.95 4.07 3.87 4 ...
##  $ y      : num  3.98 3.84 4.07 4.23 4.35 3.96 3.98 4.11 3.78 4.05 ...
##  $ z      : num  2.43 2.31 2.31 2.63 2.75 2.48 2.47 2.53 2.49 2.39 ...
```

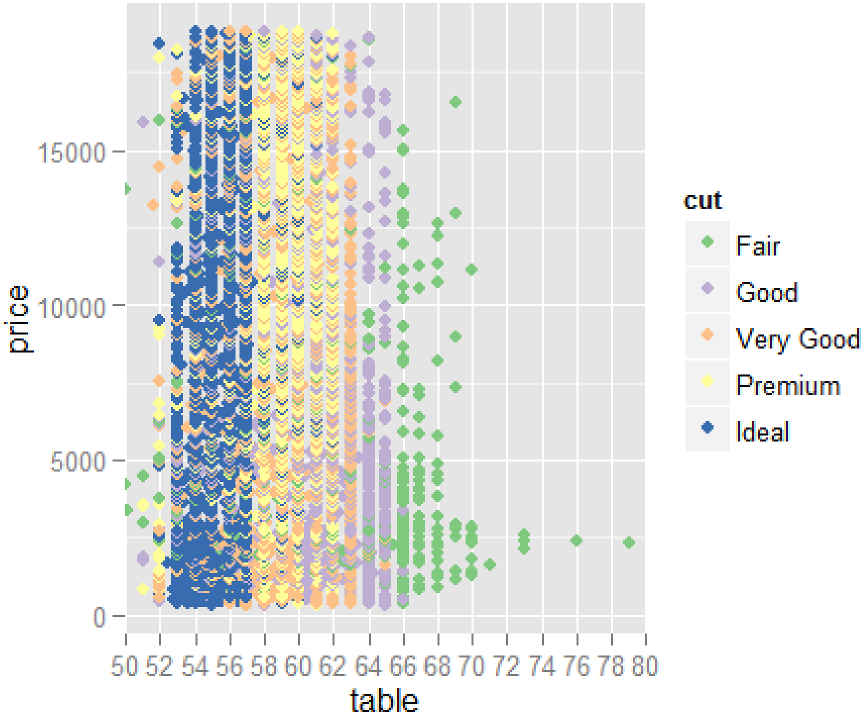
```r
#?diamonds
```

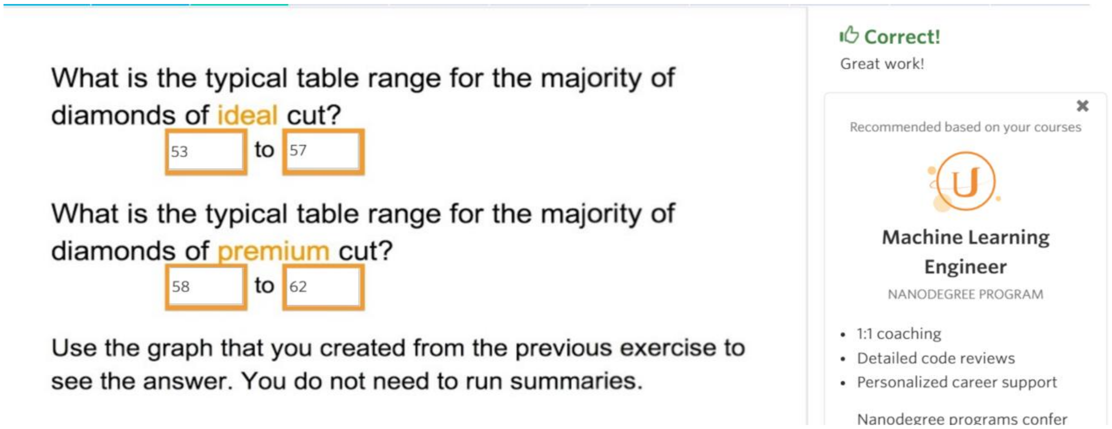
```
ggplot(aes(x=price, fill = cut), data=diamonds) +
  geom_histogram() +
  facet_wrap(~color) +
  scale_fill_brewer(type = 'qual') +
  scale_x_log10()
```

```
## stat_bin: binwidth defaulted to range/30. Use 'binwidth = x' to adjust
this.
## stat_bin: binwidth defaulted to range/30. Use 'binwidth = x' to adjust
this.
## stat_bin: binwidth defaulted to range/30. Use 'binwidth = x' to adjust
this.
## stat_bin: binwidth defaulted to range/30. Use 'binwidth = x' to adjust
this.
## stat_bin: binwidth defaulted to range/30. Use 'binwidth = x' to adjust
this.
## stat_bin: binwidth defaulted to range/30. Use 'binwidth = x' to adjust
this.
## stat_bin: binwidth defaulted to range/30. Use 'binwidth = x' to adjust
this.
```



```
ggplot(aes(x = table, y = price), data = diamonds) +
  geom_point() +
  geom_point(aes(color = cut)) +
  scale_color_brewer(type='qual') +
  coord_cartesian(xlim = c(50,80)) +
  scale_x_discrete(breaks = seq(50,80,2))
```





What is the typical table range for the majority of diamonds of ideal cut?

53 to 57

What is the typical table range for the majority of diamonds of premium cut?

58 to 62

Use the graph that you created from the previous exercise to see the answer. You do not need to run summaries.

Correct!

Great work!

Recommended based on your courses

Machine Learning Engineer

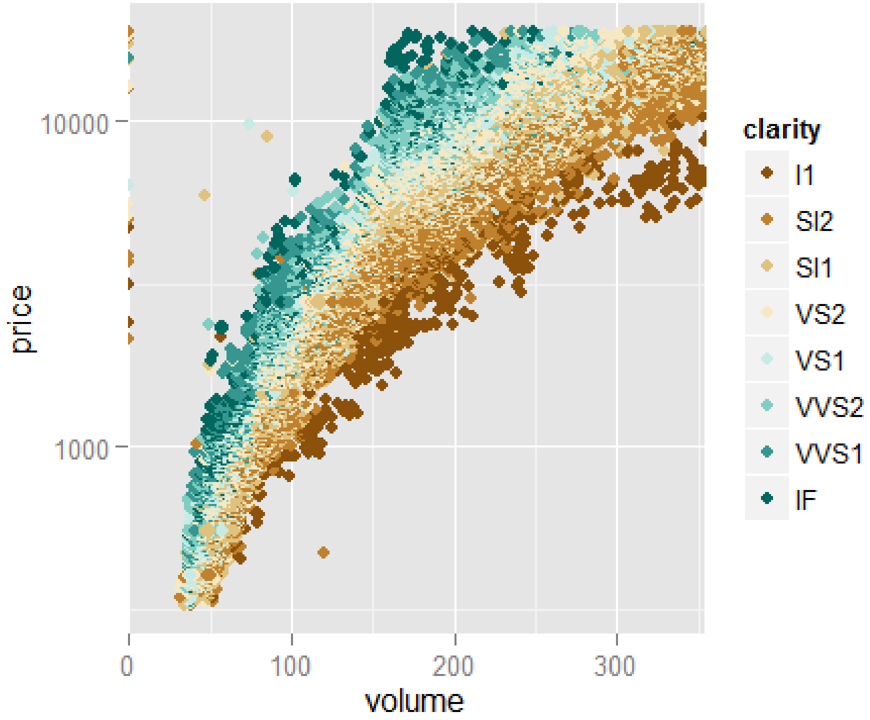
NANODEGREE PROGRAM

- 1:1 coaching
- Detailed code reviews
- Personalized career support

Nanodegree programs confer

```r
diamonds$volume = diamonds$x*diamonds$y*diamonds$z

#plot scatter of price vs volume colored by clarity
ggplot(aes(x = volume, y = price), data = diamonds) +
  geom_point(aes(color = clarity)) +
  scale_color_brewer(type = 'div') +
  coord_cartesian(xlim=c(0,quantile(diamonds$volume,0.99))) +
  scale_y_log10()
```


price
10000
1000
volume
0
100
200
300
clarity
I1
SI2
SI1
VS2
VS1
VVS2
VVS1
IF