

Grammar of Graphics - ggplot2

Azarudheen

March 10, 2019

Introduction

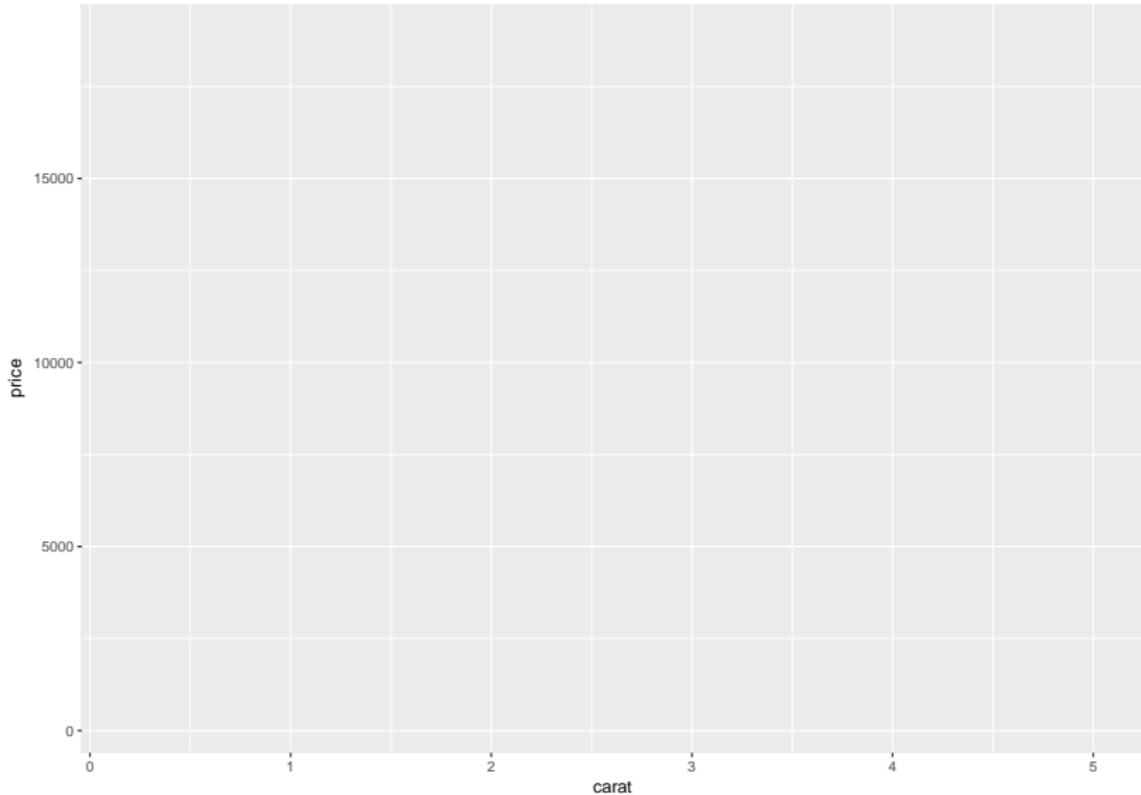
- ▶ Data visualization is an essential component in the field of Data science. In the journey of becoming Data Scientist, the very basic skill required is Data visualization.
- ▶ It is statistics and design combined in a meaningful way to interpret the data with graphs and plots.
- ▶ `ggplot2` is a graphic package which was developed and maintained by Hadley Wickham. It is easy to learn grammar for visualisation structure for R graphics code.

Understanding ggplot syntax

```
library(ggplot2)  
View(diamonds)
```

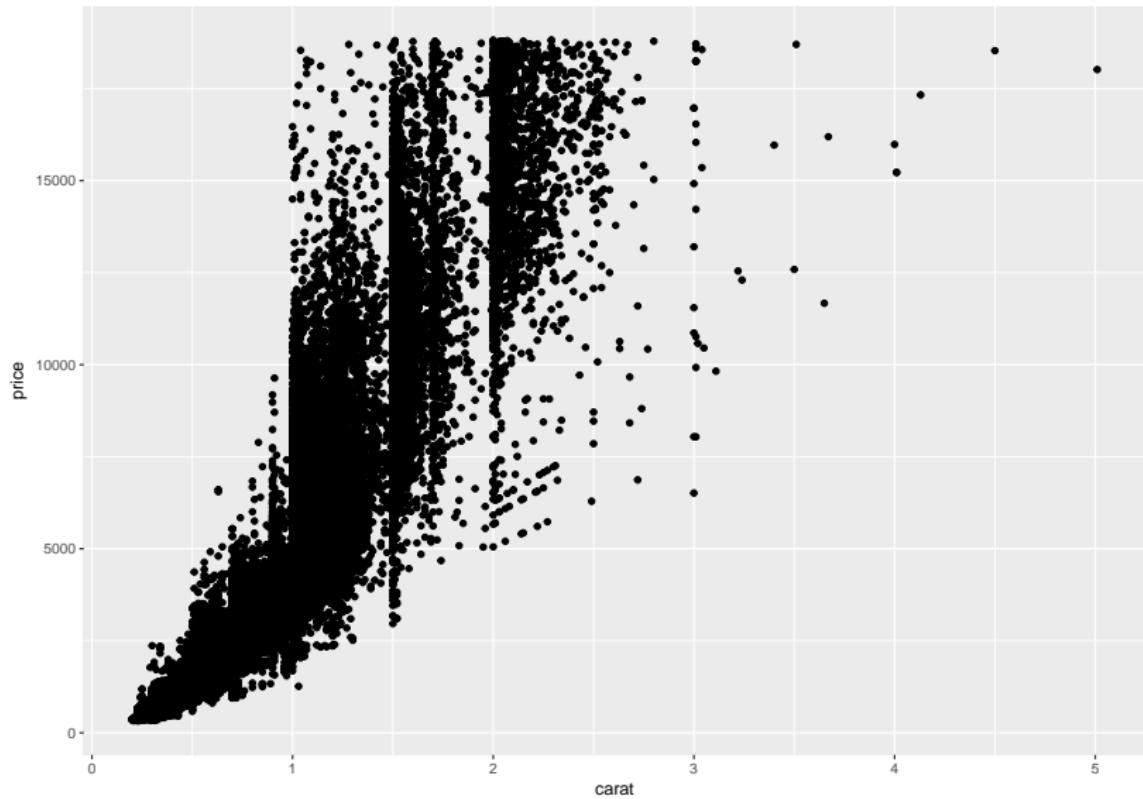
Visualize how price of the diamond varies with respect to its carat.

```
ggplot(diamonds, aes(x = carat, y = price))
```



Simple Scatter Plot

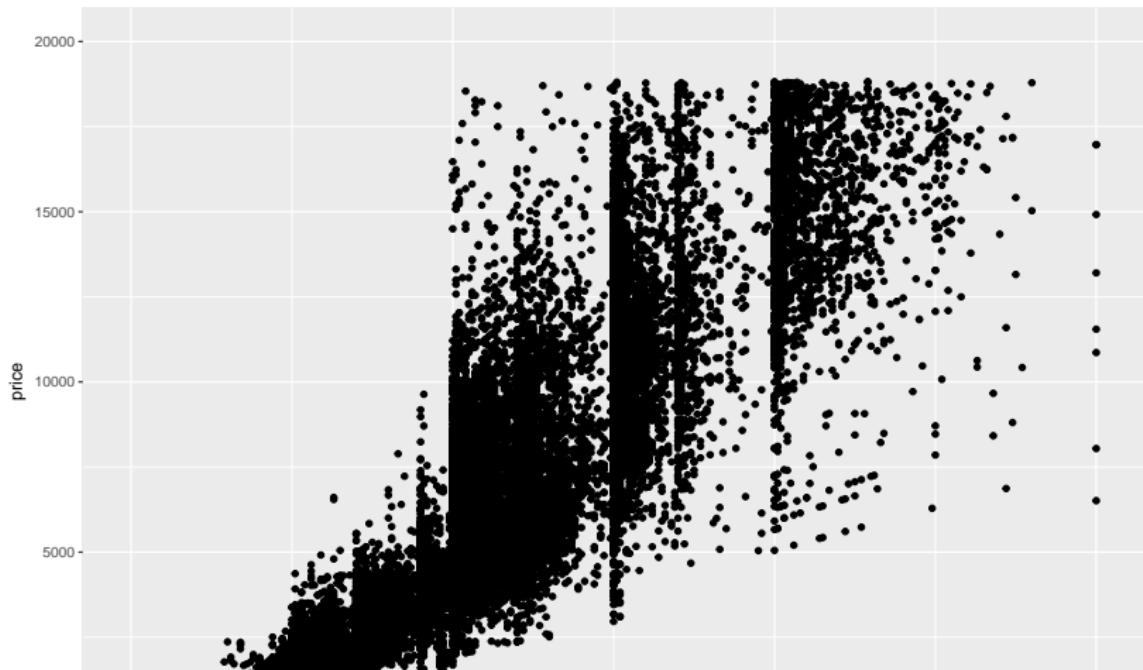
```
ggplot(diamonds, aes(x = carat, y = price)) + geom_point()
```



Adjusting X and Y limits

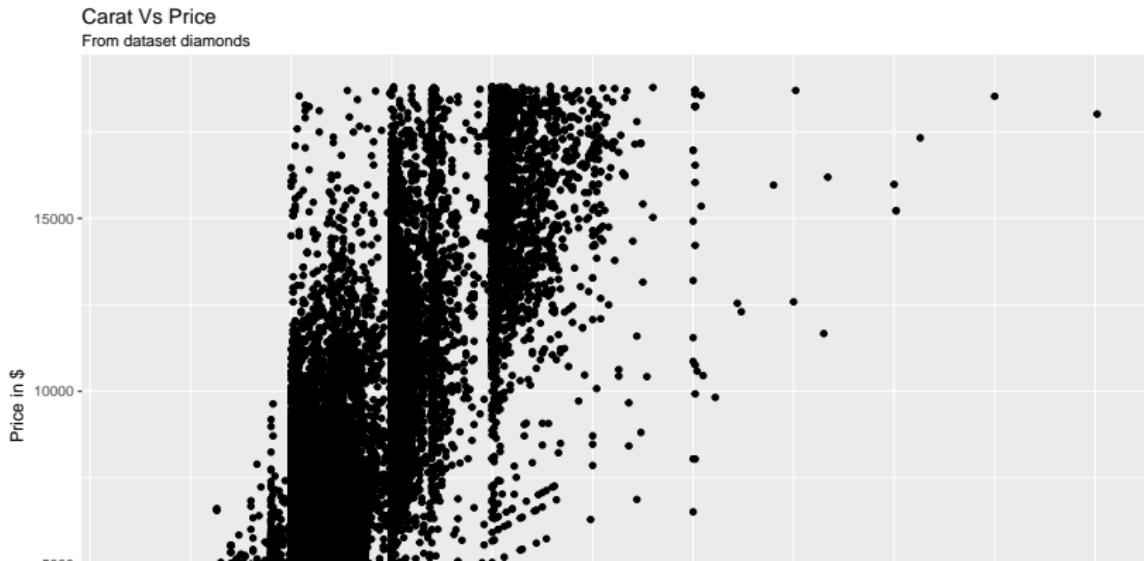
```
ggplot(diamonds, aes(x = carat, y = price)) +  
  geom_point() + xlim(c(0,3)) + ylim(c(0,20000))
```

```
## Warning: Removed 32 rows containing missing values (geom_point)
```



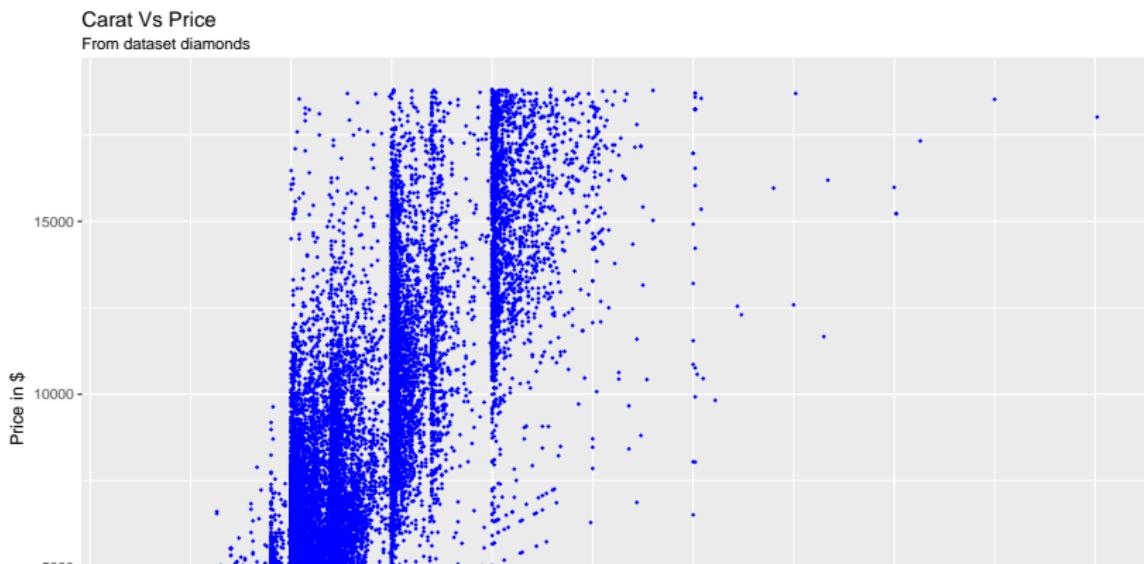
Adding Title and labels

```
ggplot(diamonds, aes(x = carat,  
                      y = price)) + geom_point() +  
  labs(title = "Carat Vs Price",  
       subtitle = "From dataset diamonds",  
       x = "Carat", y = "Price in $",  
       caption = "Plot shows how price varies with respect to carat")
```



Adding color and size of points

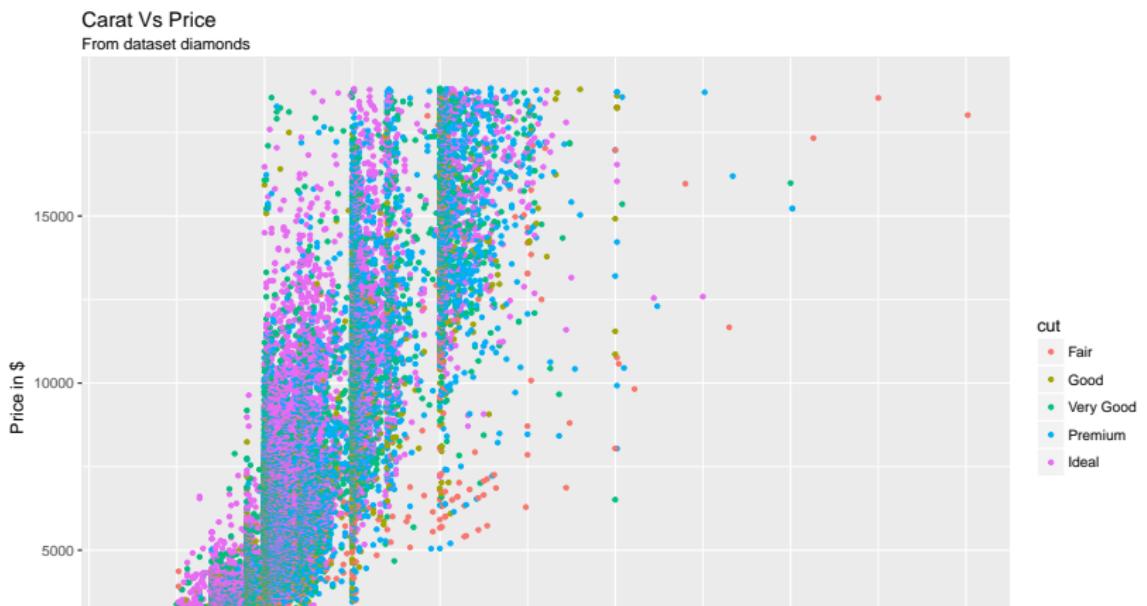
```
ggplot(diamonds, aes(x = carat, y = price)) +  
  geom_point(col = "blue", size = 1, shape = 18) +  
  labs(title = "Carat Vs Price",  
       subtitle = "From dataset diamonds",  
       x = "Carat", y = "Price in $",  
       caption = "Plot shows how price varies with respect to carat")
```



Show price of the diamond varies with respect to its carat for different cuts.

Changeing colors for each cluster

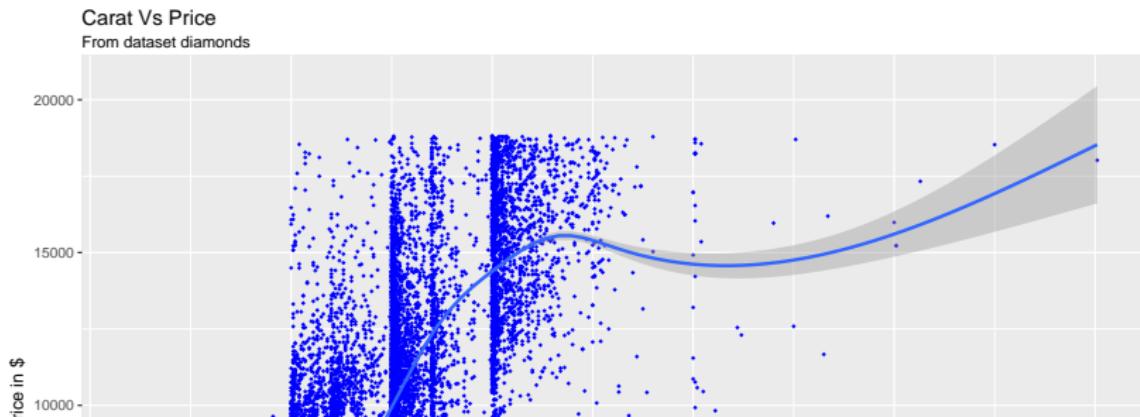
```
ggplot(diamonds, aes(x = carat, y = price)) +  
  geom_point(aes(col = cut)), size = 1) +  
  labs(title = "Carat Vs Price", subtitle = "From dataset dia  
  x = "Carat", y = "Price in $",  
  caption = "Plot shows how price varies with respect
```



Fitting a line

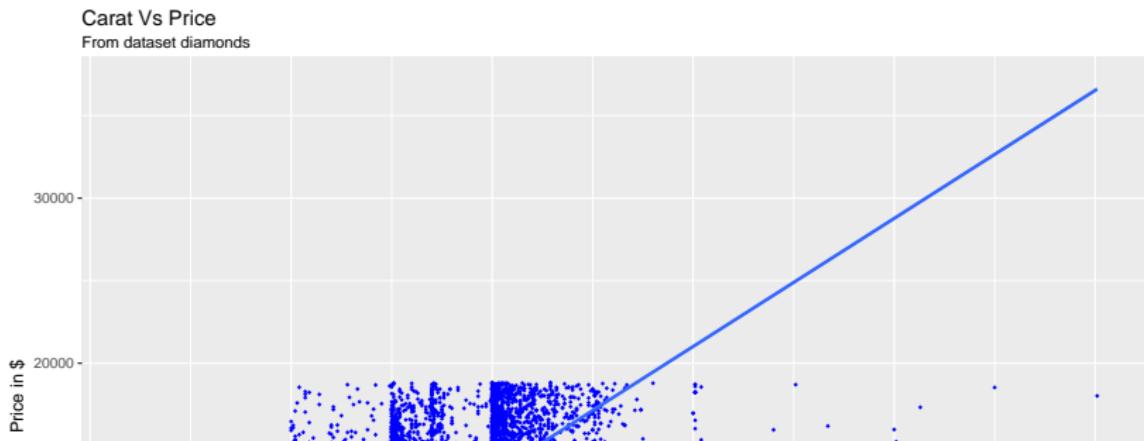
```
ggplot(diamonds, aes(x = carat,y = price))+  
  geom_point(col = "blue", size = 1,  
             shape = 18)+geom_smooth()  
  labs(title="Carat Vs Price",  
       subtitle = "From dataset diamonds",  
       x = "Carat", y = "Price in $",  
       caption = "Plot shows how price varies with respect
```

```
## `geom_smooth()` using method = 'gam'
```



Fitting a Linear Line

```
ggplot(diamonds, aes(x = carat, y = price))+
  geom_point(col = "blue", size = 1,
             shape = 18) +
  geom_smooth(method = "lm") +
  labs(title="Carat Vs Price",
       subtitle = "From dataset diamonds",
       x = "Carat", y = "Price in $",
       caption = "Plot shows how price varies with respect
```

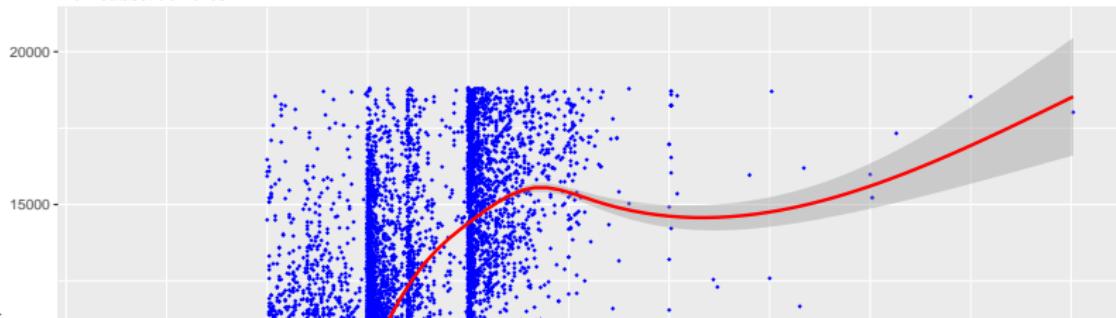


Changing the color of fitted line

```
ggplot(diamonds, aes(x = carat, y = price)) +  
  geom_point(col = "blue", size = 1,  
             shape = 18) +  
  geom_smooth(col = "Red") +  
  labs(title = "Carat Vs Price",  
       subtitle = "From dataset diamonds",  
       x = "Carat", y = "Price in $",  
       caption = "Plot shows how price varies with respect to carat")
```

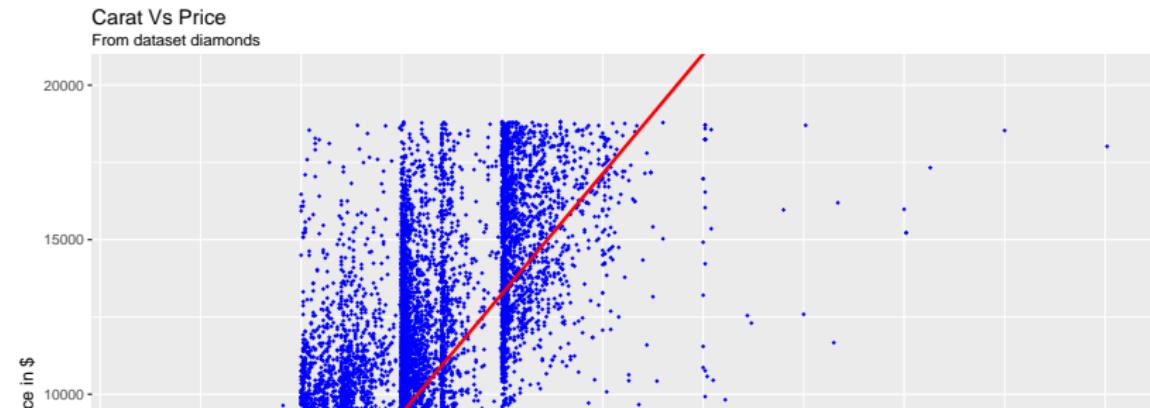
```
## `geom_smooth()` using method = 'gam'
```

Carat Vs Price
From dataset diamonds



Zooming the plot

```
ggplot(diamonds, aes(x = carat,y = price))+  
  geom_point(col = "blue", size = 1,  
             shape = 18)+  
  geom_smooth(method = "lm", col = "Red") +  
  labs(title="Carat Vs Price",  
       subtitle = "From dataset diamonds",  
       x = "Carat", y = "Price in $",  
       caption = "Plot shows how price varies with respect  
coord_cartesian(ylim=c(0,20000))
```



```
ggplot(diamonds, aes(x = carat, y = price)) +  
  geom_point(aes(color = cut), size = 1) +  
  geom_smooth() +  
  labs(title = "Carat Vs Price",  
       subtitle = "From dataset diamonds",  
       x = "Carat", y = "Price in $",  
       caption = "Plot shows how price varies with respect
```

```
## `geom_smooth()` using method = 'gam'
```



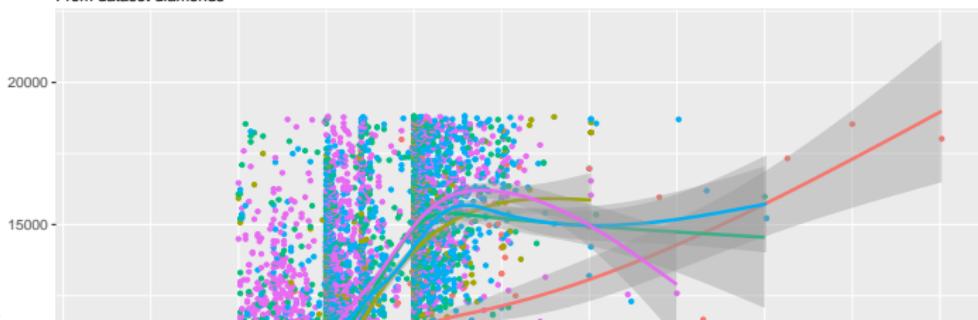
Fitting lines based on cut of diamond

```
ggplot(diamonds, aes(x = carat, y = price,  
                      color = cut)) +  
  geom_point(size = 1) +  
  geom_smooth() +  
  labs(title = "Carat Vs Price",  
       subtitle = "From dataset diamonds",  
       x = "Carat", y = "Price in $",  
       caption = "Plot shows how price varies with respect to carat")
```

```
## `geom_smooth()` using method = 'gam'
```

Carat Vs Price

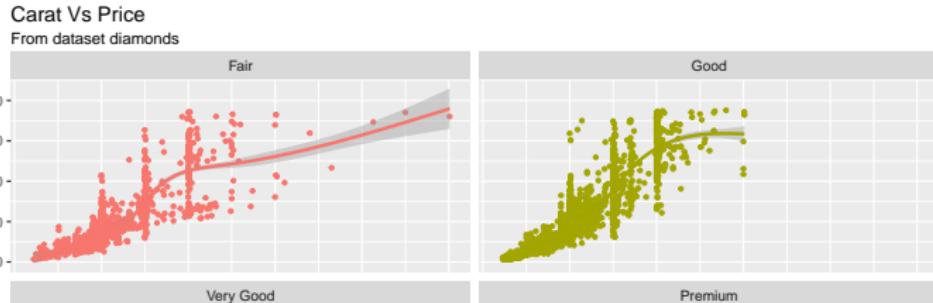
From dataset diamonds



Separate plot for each cut

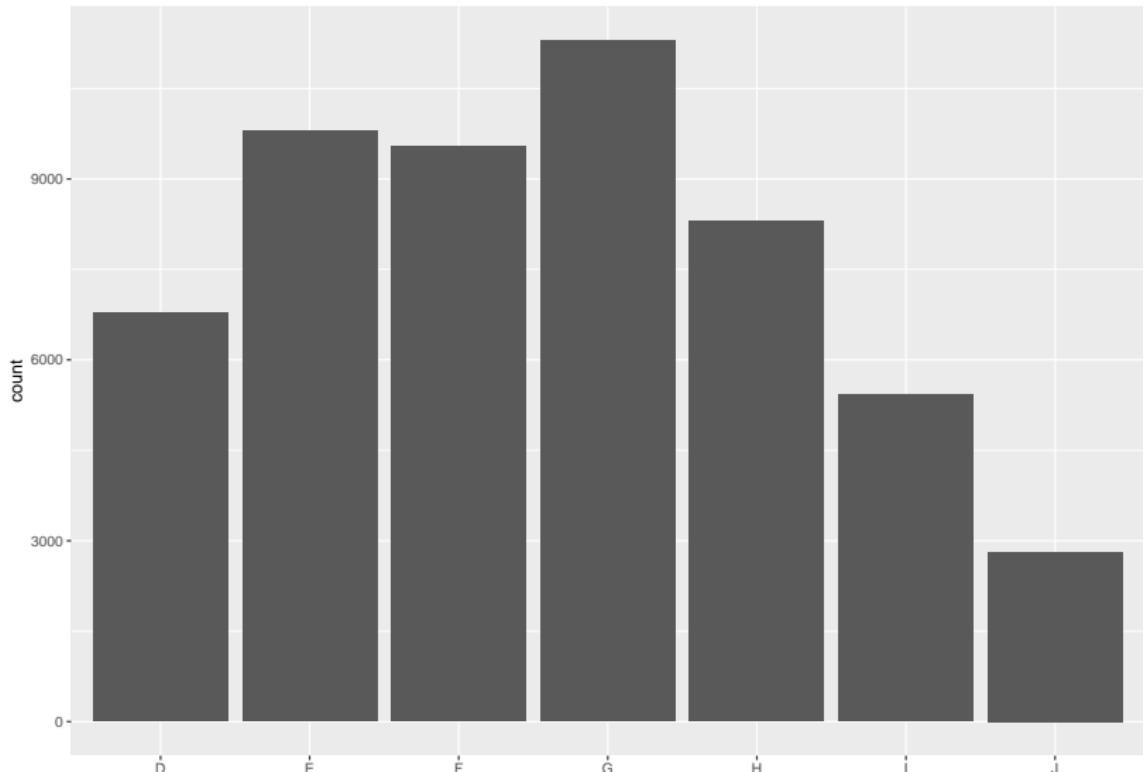
```
ggplot(diamonds, aes(x = carat, y = price,  
                      color = cut)) +  
  geom_point(size = 1) +  
  geom_smooth() +  
  labs(title = "Carat Vs Price",  
       subtitle = "From dataset diamonds",  
       x = "Carat", y = "Price in $",  
       caption = "Plot shows how price varies with respect  
facet_wrap(~cut, ncol = 2)
```

```
## `geom_smooth()` using method = 'gam'
```



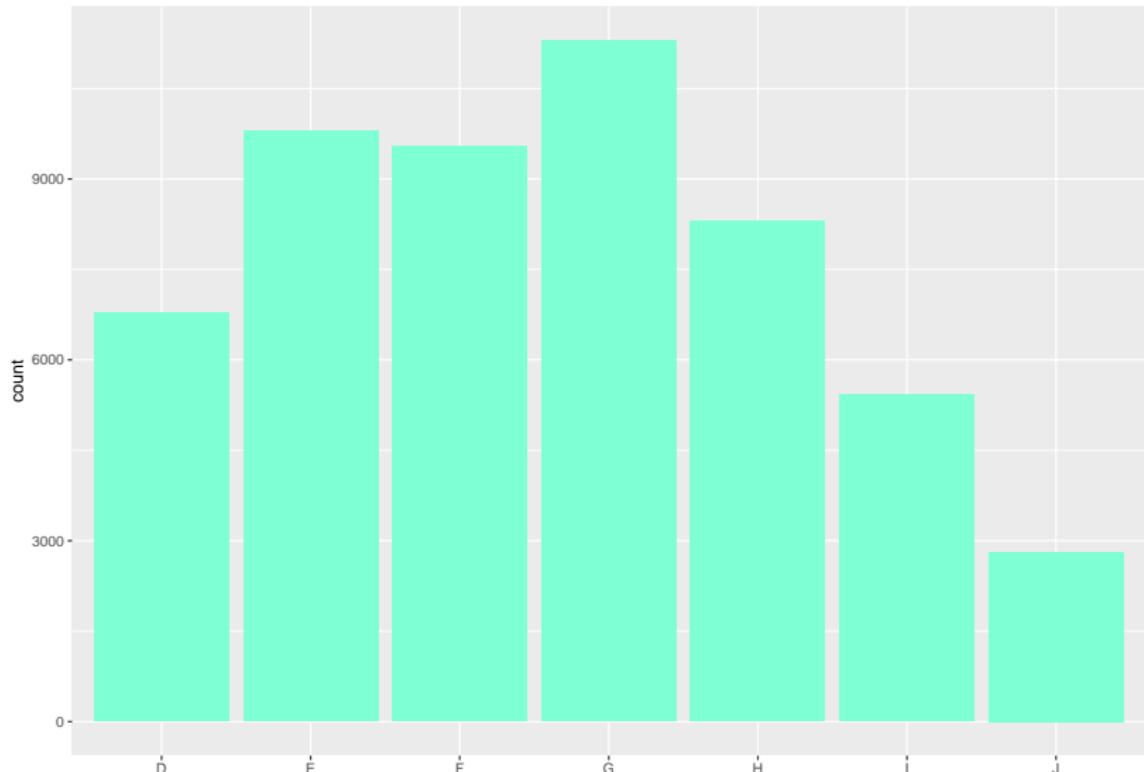
Bar Plot

```
A = ggplot(diamonds, aes(x = color)) + geom_bar()  
A
```

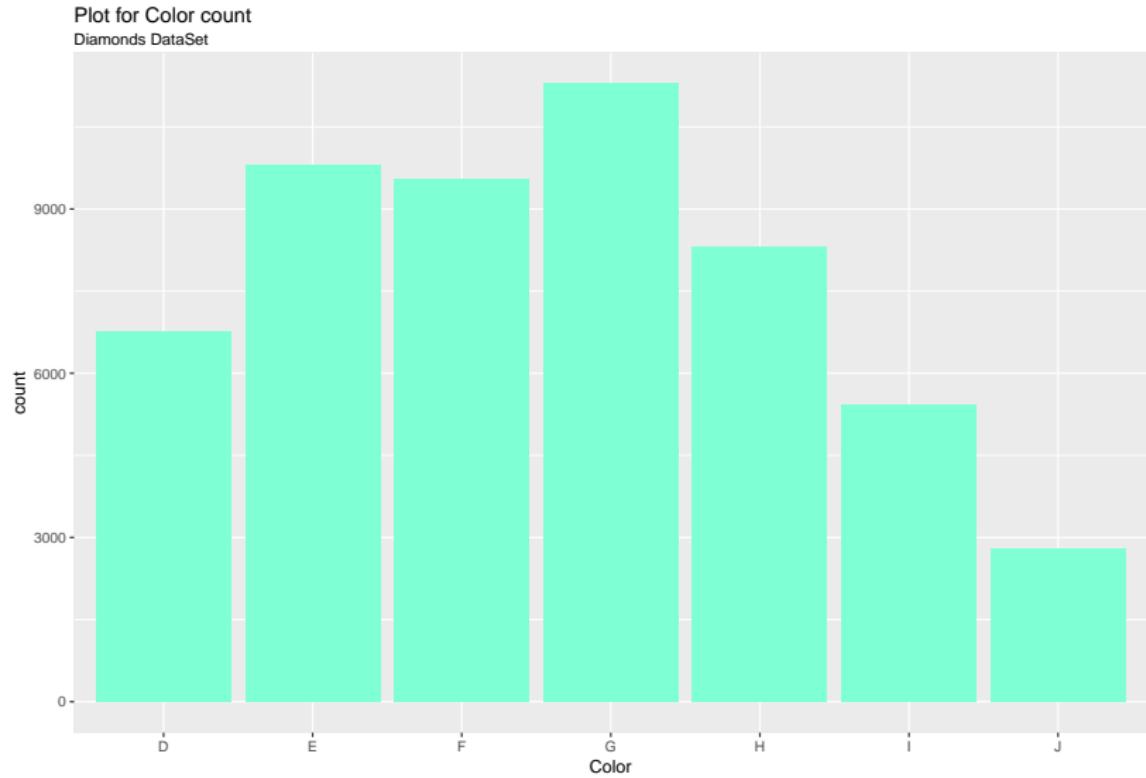


Adding colour

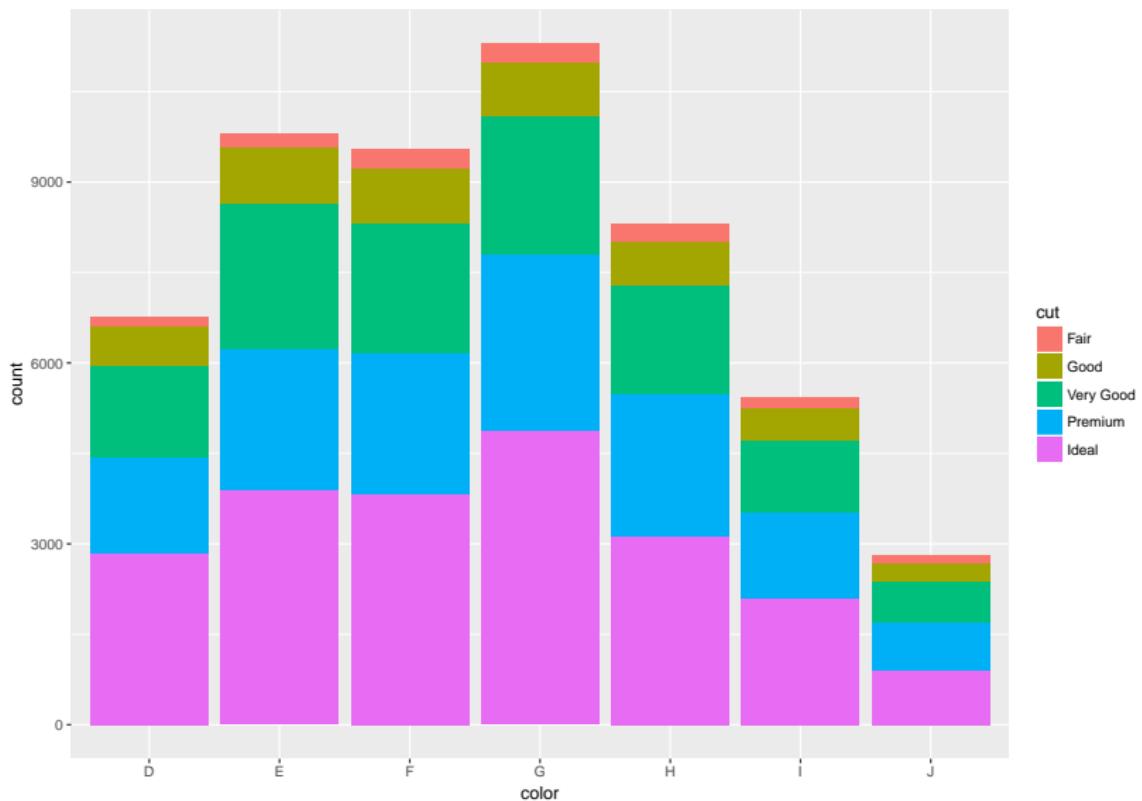
```
A = ggplot(diamonds, aes(x = color)) + geom_bar(fill = "aquamarine")  
A
```



```
A+labs(title = "Plot for Color count",  
       subtitle = "Diamonds DataSet", x = "Color",  
       y = "count", caption="bar plot")
```

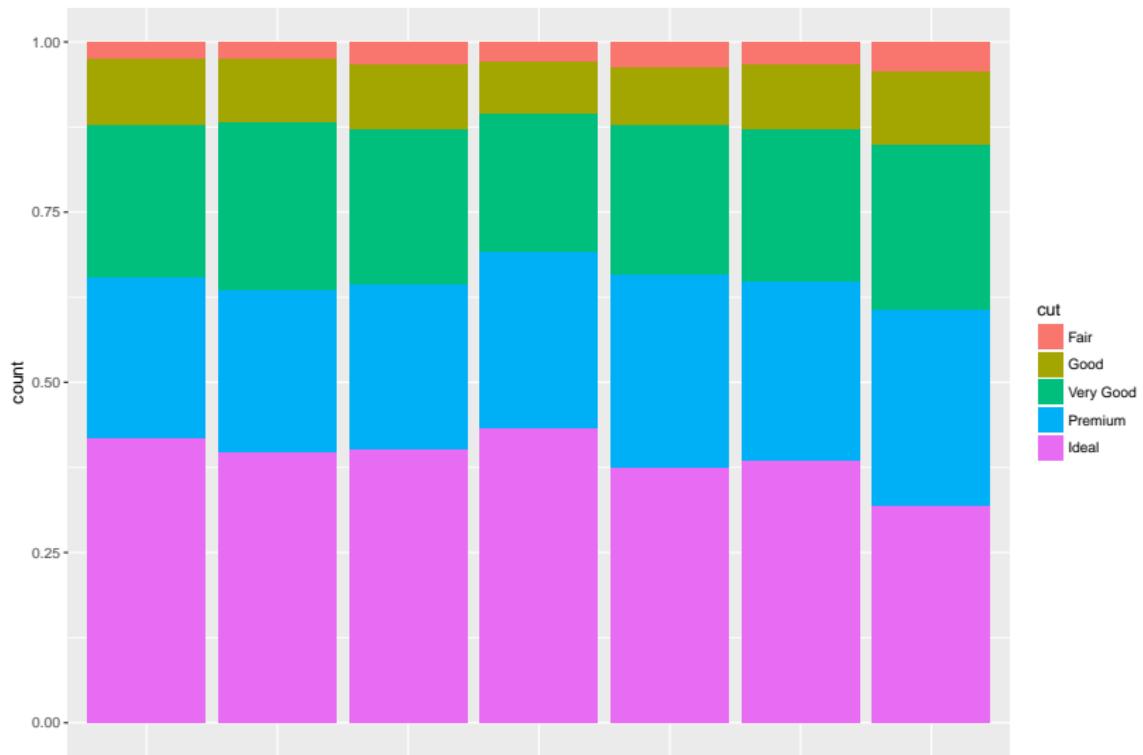


```
C = ggplot(diamonds, aes(x=color, fill=cut)) + geom_bar()  
C
```



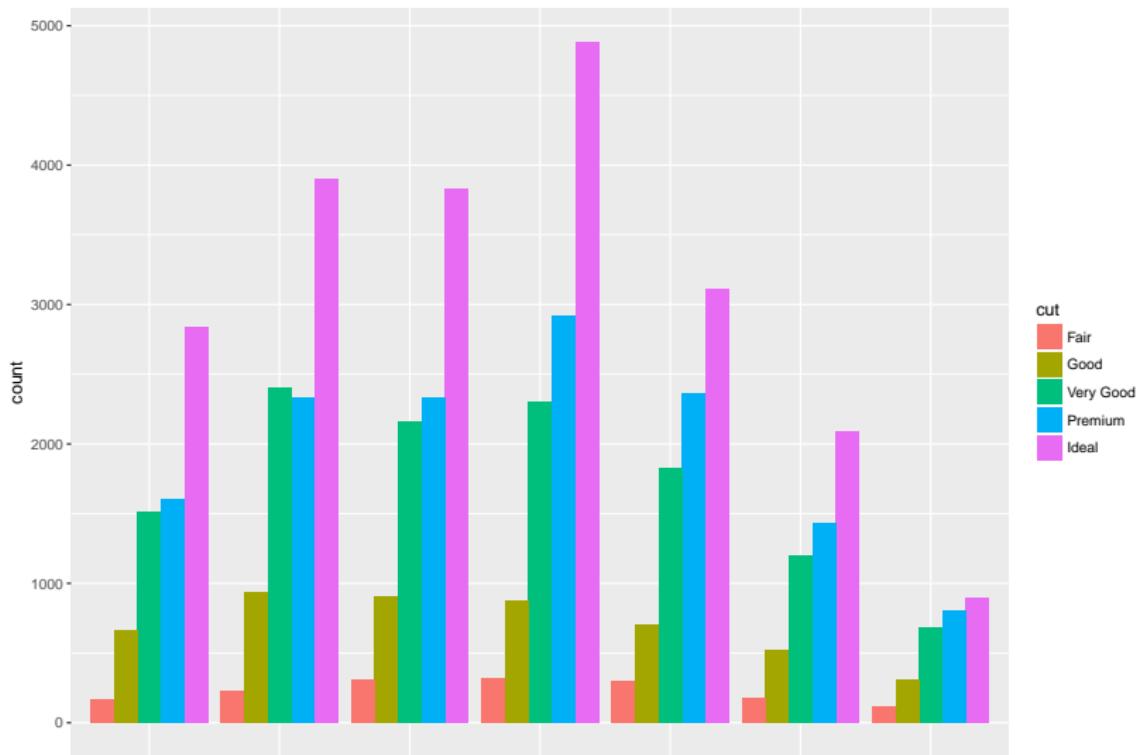
Bar Plot in terms of proportion

```
ggplot(diamonds, aes(x=color, fill=cut)) +  
  geom_bar(position = "fill")
```



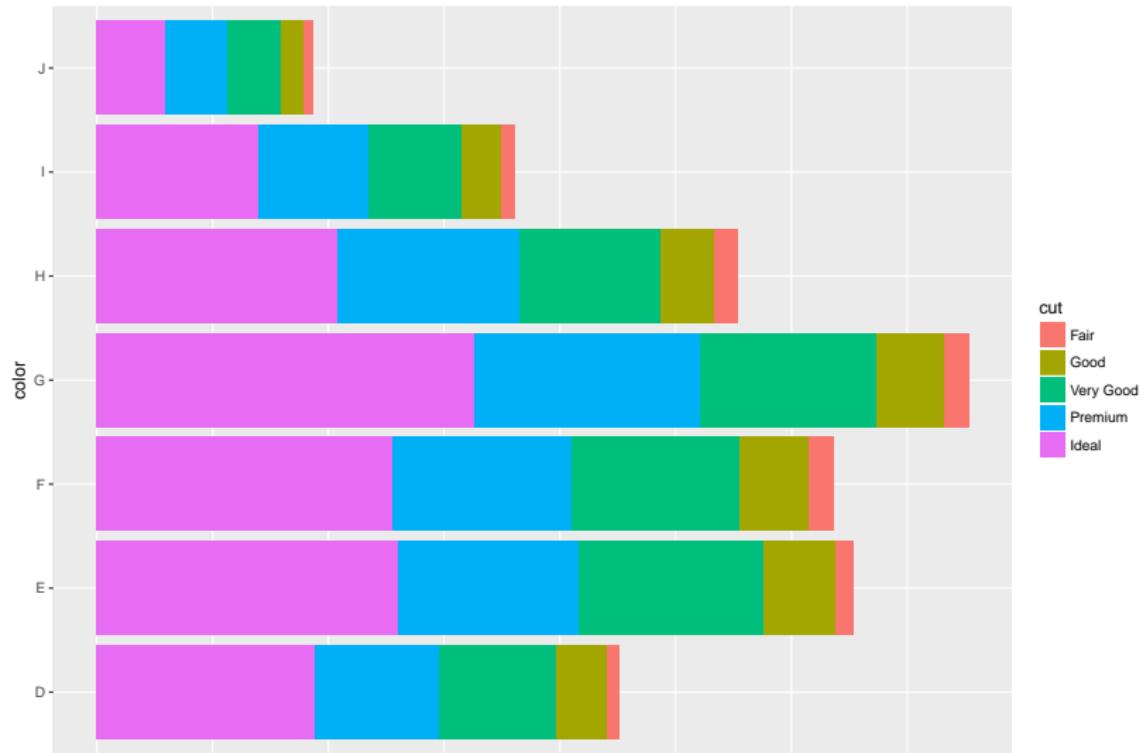
Multiple Bar graph

```
ggplot(diamonds, aes(x=color, fill=cut)) +  
  geom_bar(position = "dodge")
```



Inverted Bar

```
ggplot(diamonds, aes(x=color, fill=cut)) +  
  geom_bar() + coord_flip()
```



C+facet_grid(~cut)

