Data visualization(Edureka)

SAGAR MEHTA

21/05/2020

1. Load the required libraries and the data.

```
library(ggplot2)
library(dplyr)
```

```
##
## Attaching package: 'dplyr'
```

```
## The following objects are masked from 'package:stats':
##
##     filter, lag
```

```
## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

#2. Understand the data structure and provide concise summary on the following – #• no of observations #• total number of variables #• number of continuous variables #• number of categorical variables #• number of variables which have missing values

```
hd <- read.csv('Housing_data.csv',stringsAsFactors = T)
head(hd)
```

```
##     Record Gender No_kids    Education HasCar Income PropertyValue Loan_Period
## 1  Record1 Female       0     Graduate     No    710         90400         456
## 2  Record8   Male       0     Graduate     No   6516        168800         336
## 3  Record9   Male       0     Graduate    Yes   7040        160000         336
## 4 Record10   Male       0 Not Graduate     No   4730        155200         336
## 5 Record11   Male       0     Graduate     No   9167        149600         336
## 6 Record12   Male       0     Graduate     No  10459        149600         336
##   Credit_Record Housing_type Property_Purchased
## 1             1   Affordable                 Y
## 2             1   Affordable                 Y
## 3             1   Affordable                 Y
## 4             1   Affordable                 Y
## 5             1   Affordable                 Y
## 6             1   Affordable                 Y
```

```
dim(hd)
```

```
## [1] 505  11
```

```
class(hd)
```

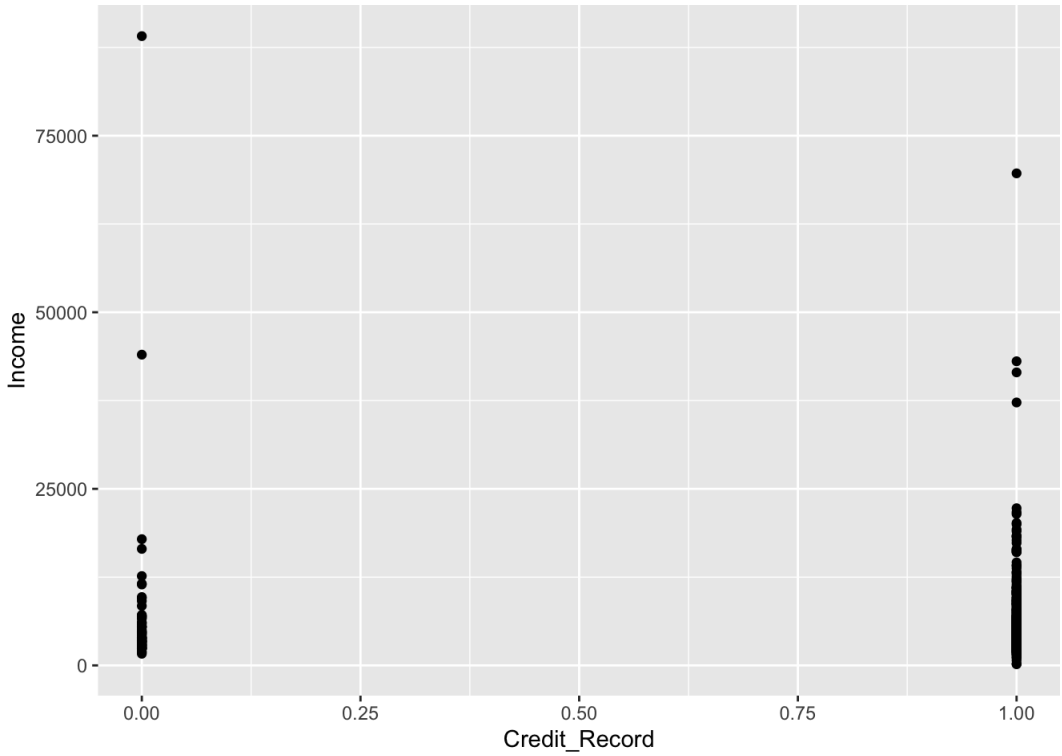
```
## [1] "data.frame"
```

```
str(hd)
```

```
## 'data.frame':    505 obs. of  11 variables:
##  $ Record           : Factor w/ 505 levels "Record1","Record10",..: 1 484 495 2 13 24 35 57 68 90 ...
##  $ Gender           : Factor w/ 2 levels "Female","Male": 1 2 2 2 2 2 2 2 2 2 ...
##  $ No_kids          : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ Education        : Factor w/ 2 levels "Graduate","Not Graduate": 1 1 1 2 1 1 1 1 1 1 ...
##  $ HasCar           : Factor w/ 3 levels "No","Not Answered",..: 1 1 3 1 1 1 1 3 1 1 ...
##  $ Income           : int  710 6516 7040 4730 9167 10459 2888 10960 8692 4044 ...
##  $ PropertyValue    : int  90400 168800 160000 155200 149600 149600 149600 144000 144000 137600 ...
##  $ Loan_Period      : int  456 336 336 336 336 336 336 336 336 336 ...
##  $ Credit_Record    : int  1 1 1 1 1 1 1 1 1 1 ...
##  $ Housing_type     : Factor w/ 3 levels "Affordable","Mid Range",..: 1 1 1 1 1 1 1 1 1 1 ...
##  $ Property_Purchased: Factor w/ 2 levels "N","Y": 2 2 2 2 2 2 2 2 2 2 ...
```

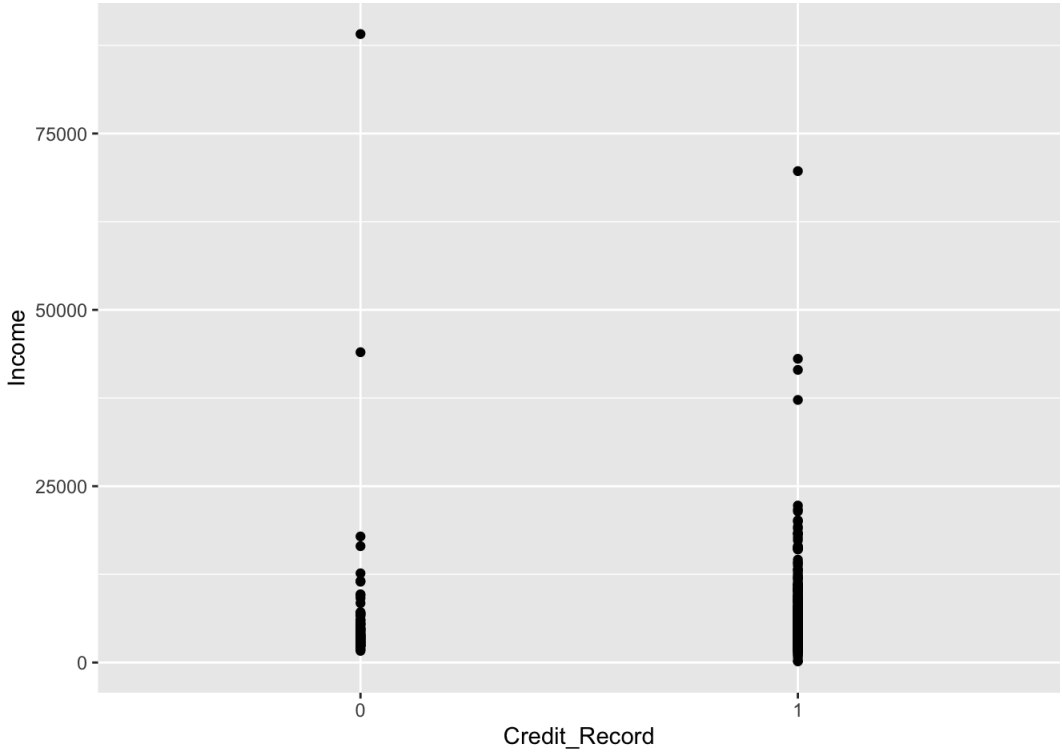
#3. Create a scatter plot between Credit_Record on x-axis and Income on y-axis. #• Is the plot satisfying, if not, what could be the reason? #• Change the command executed in the previous line so that Credit_Record is #treated as factor. #• what is the change in the above two plots?

```
ggplot(data = hd,aes(x = Credit_Record,y = Income))+
  geom_point()
```




```
# there are only two values so the plot is different
hd1 <- hd%>%select(Credit_Record,Income)
hd1$Credit_Record <- as.factor(hd1$Credit_Record)

ggplot(data = hd1,aes(x = Credit_Record,y = Income))+
  geom_point()
```

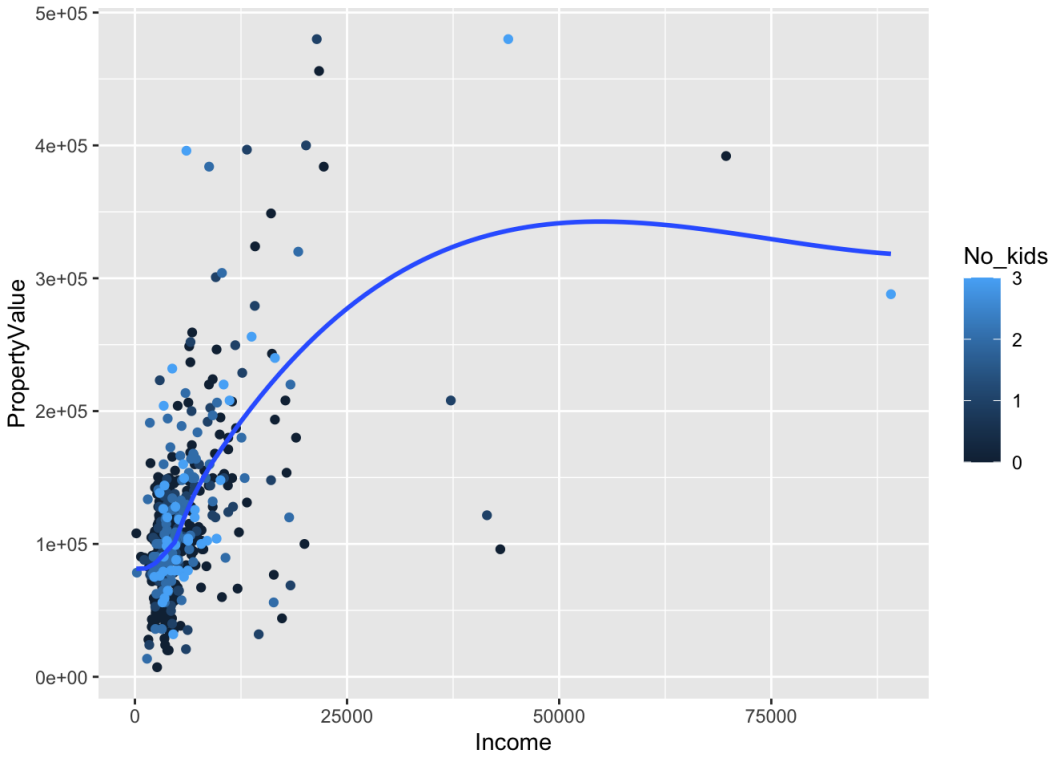



```
# there are only two outcomes on the x-axis
```

#4. Create a scatter plot between Income on x-axis and PropertyValue on y-axis. #• In the above plot, add the color argument which should be dependent on the #No_kids of the applicant #• In the above plot, now add the size argument which should be dependent on #the No_kids of the applicant. #• Now, in the above plot, please add the smooth line using the geom_smooth() #function.

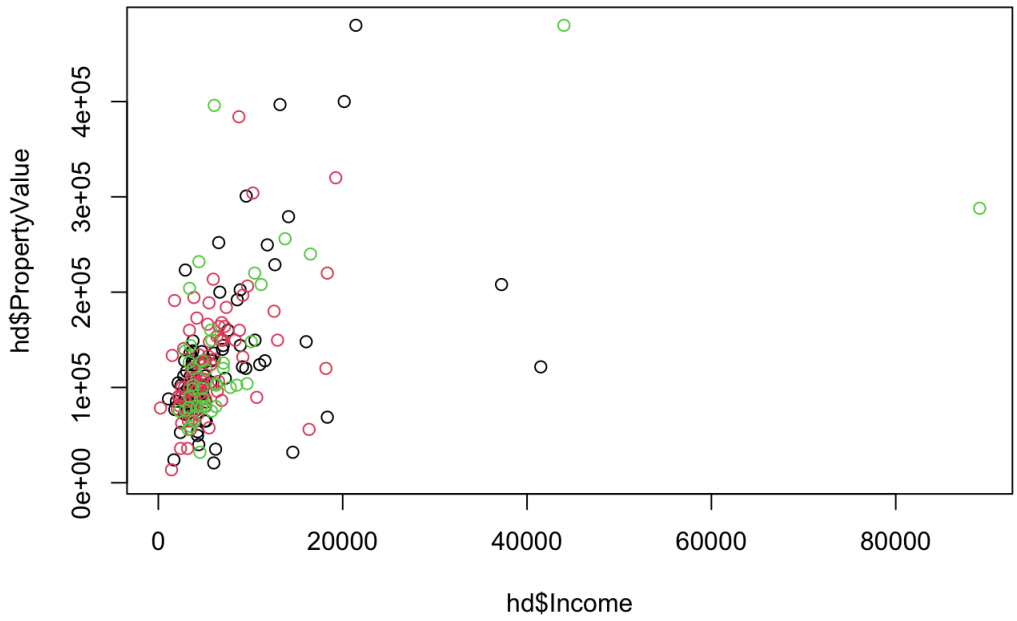
```
ggplot(data = hd,aes(x = Income, y = PropertyValue,col = No_kids))+
  geom_point()+
  geom_smooth(se = FALSE)
```

```
## `geom_smooth()` using method = 'loess' and formula 'y ~ x'
```

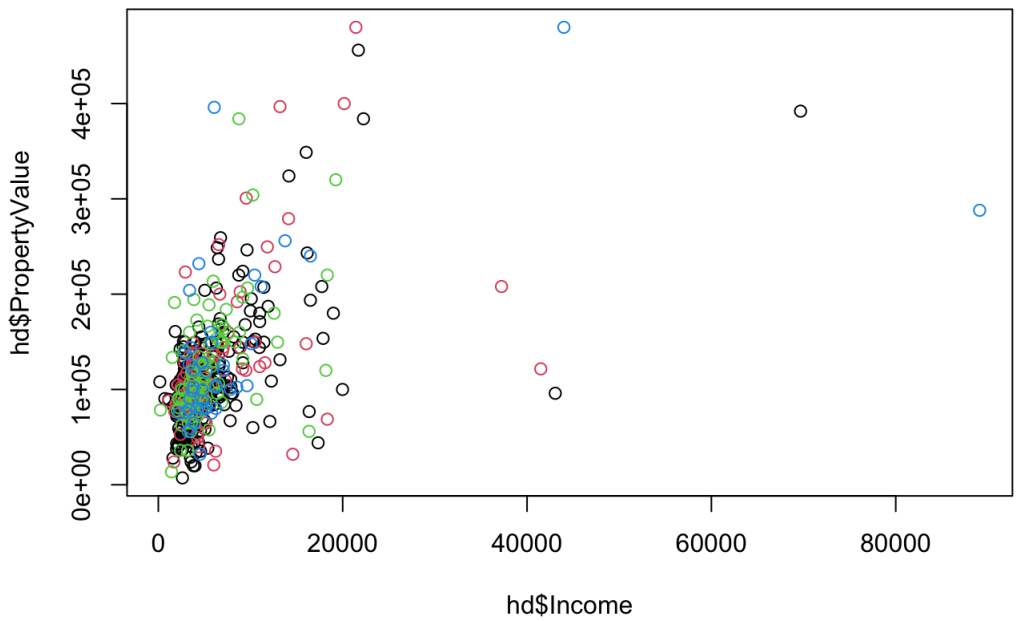


#5. ggplot comparson with Base plot : #• Using the base package plot(), make a scatter plot with Income on the x-axis #and PropertyValue on the y-axis, colored according to No of kids (use the col #argument). #• Now, Change No_kids in previous step to a factor #• Now, Make the same plot as in the first instruction - 5a #• Now, recreate the same plot as above using the ggplot functon.

```
#base R
plot(x = hd$Income, y = hd$PropertyValue,col = hd$No_kids)
```



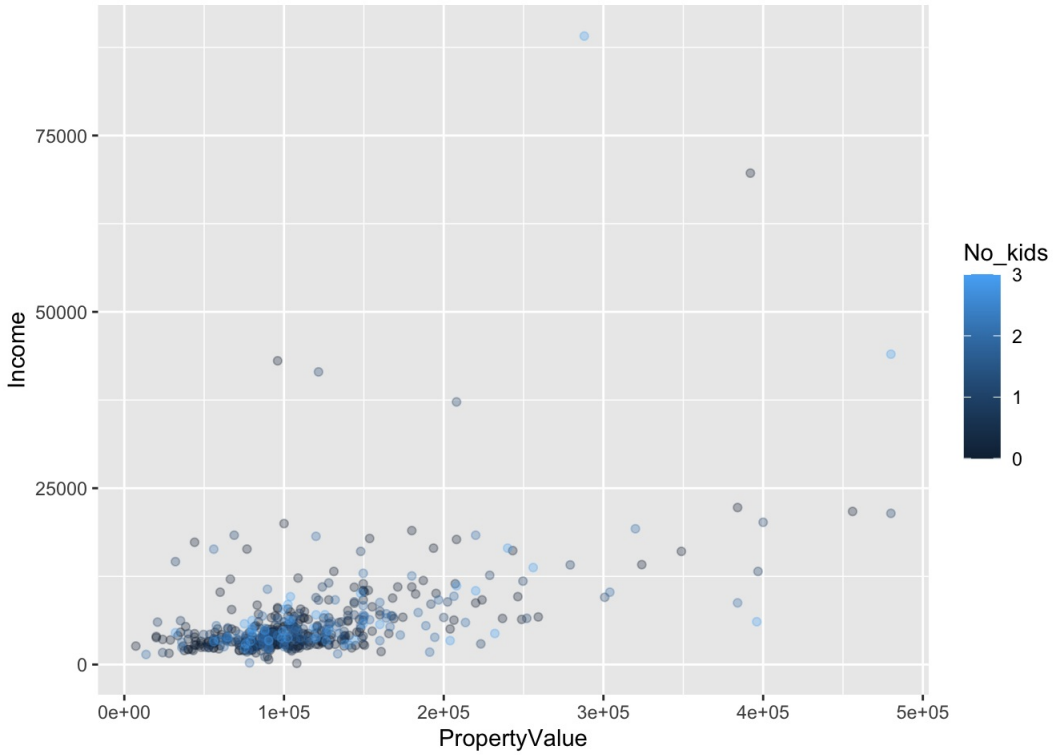

```
#no of kids as factors
plot(x = hd$Income, y = hd$PropertyValue,col = as.factor(hd$No_kids))
```



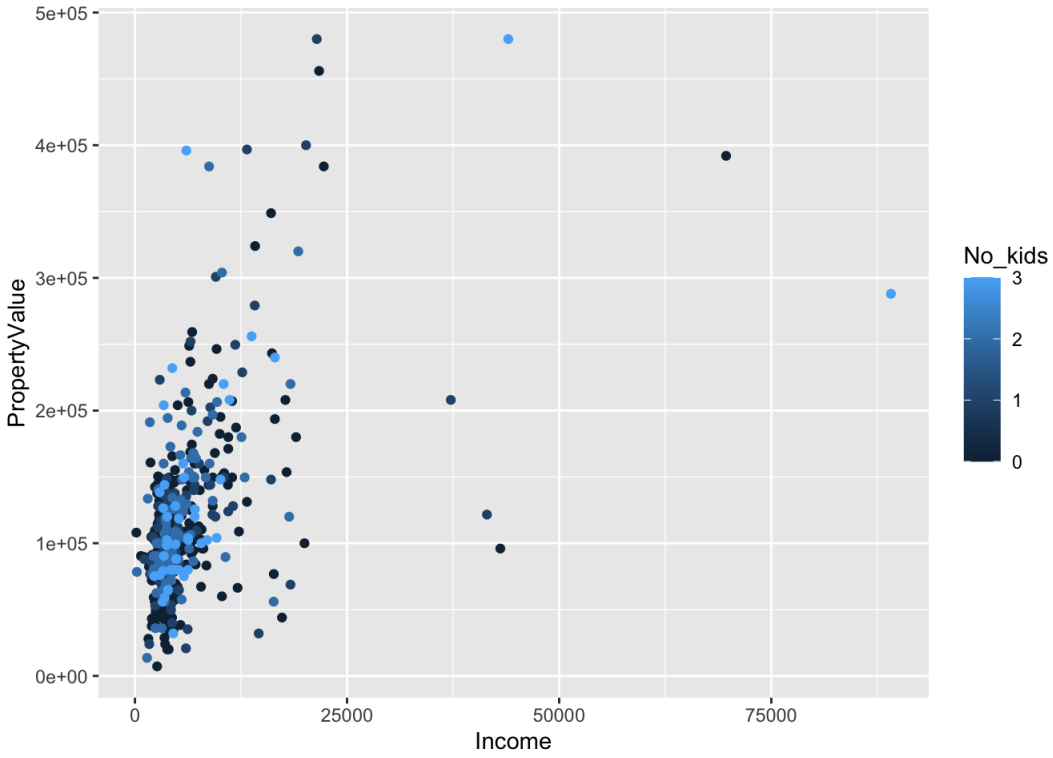

```
#ggplot
ggplot(data = hd,aes(x = Income, y = PropertyValue,col = as.factor(No_kids)))+
  geom_point()
```

#6. Aesthetics: #• Map Income to x and Property Value to y #• Reverse: Map Property Value to x and Income to y #• Map Income to x and Property Value to y and No of kids to col #• Change shape and size of the points in the above plot.

```
ggplot(data = hd,aes(x = PropertyValue, y = Income,col = No_kids))+
geom_point(alpha=0.3)
```

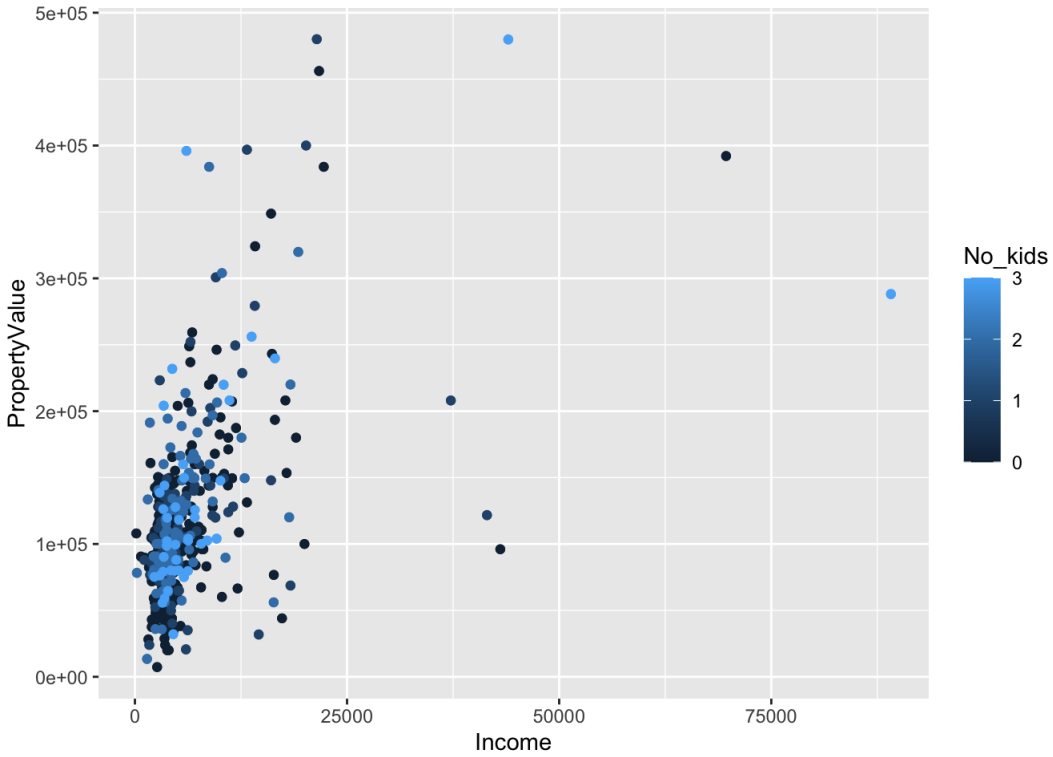



```
#after reversing the values
ggplot(data = hd,aes(x = Income, y = PropertyValue,col = No_kids))+
  geom_point()
```



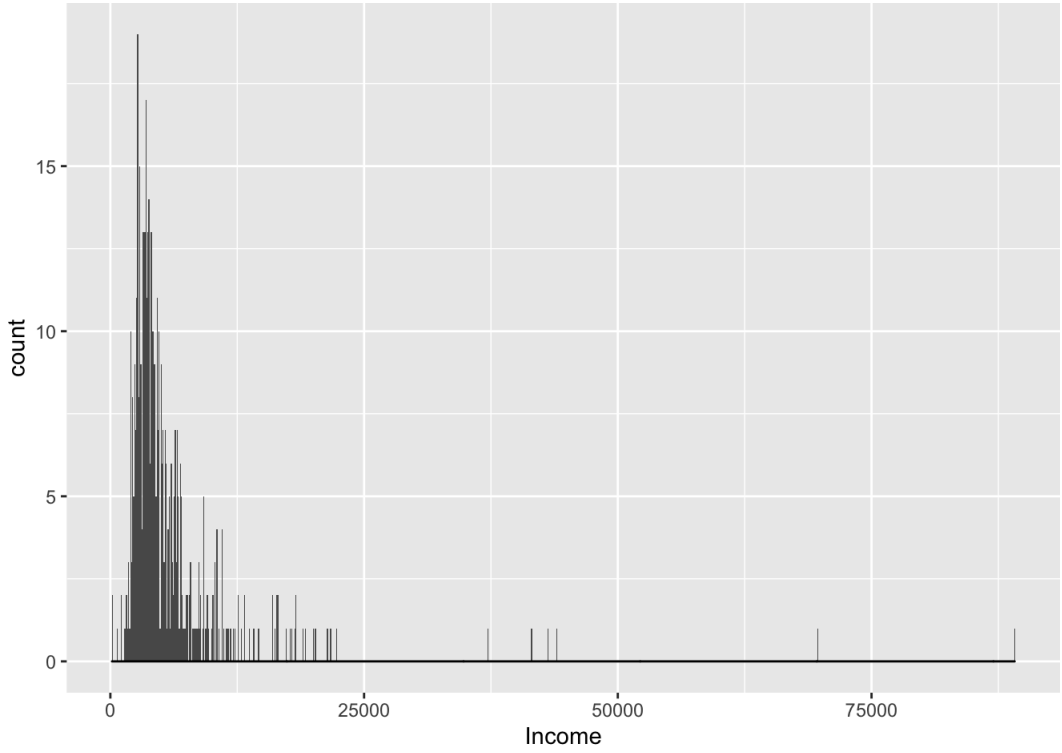
#7. Geometry: #• Start with creating a scatter plot mapping Income to x and Property Value to #y. #• Make a plot With geom_jitter() function #• Now, in the above plot, Set width in geom_jitter(). Take the width value as 0.1

```
ggplot(data = hd,aes(x = Income, y = PropertyValue,col = No_kids))+
  geom_point()+
  geom_jitter(width = 0.1)
```

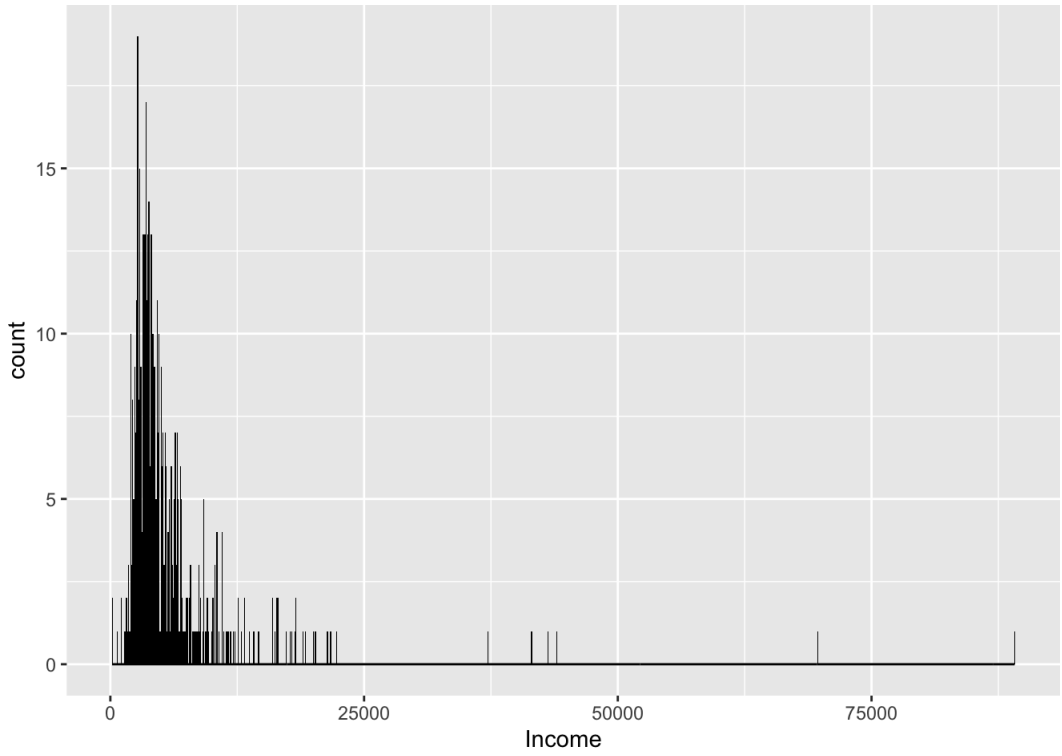


1. Histogram: #• Make a univariate histogram on Income #• In the above plot, add set binwidth to 100 in the geom layer #• In the above plot, MAP ..density.. to the y aesthetic (i.e. in a second aes() #function) #• Finally, in the above plot, plus SET the fill attribute to "#377EB8".

```
ggplot(data = hd,aes(x = Income))+
  geom_histogram(binwidth = 100)+
  geom_density()
```

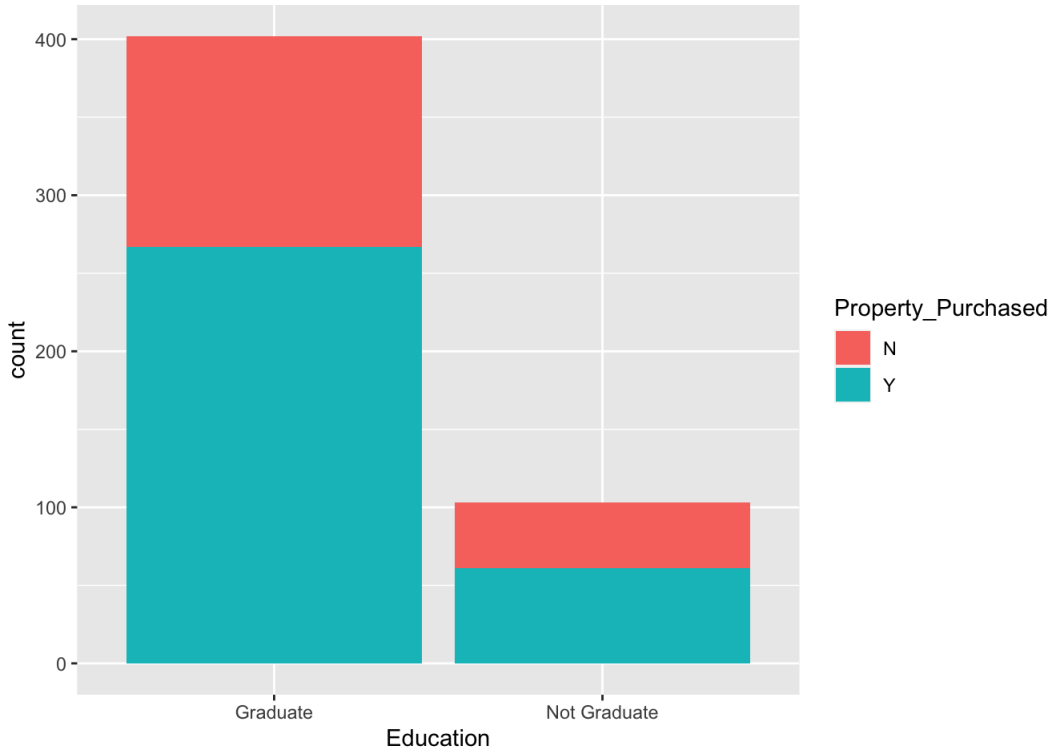



```
#density
ggplot(data = hd,aes(x = Income,))+
  geom_histogram(fill = '377EB8',binwidth = 100)+
  geom_line(stat = "density")
```

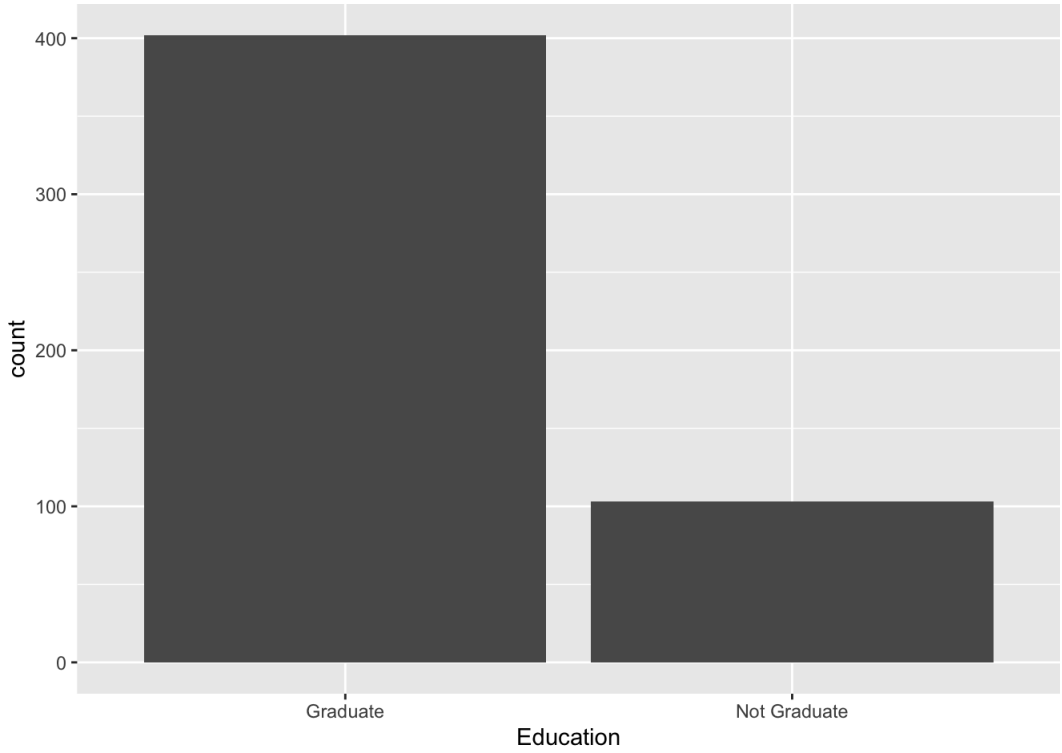


#9. Bar Plot: #• Draw a bar plot of Property_Purchased, filled according to Education #• In the above plot, Change the position argument to “stack” #• In the above plot, Change the position argument to “fill” #• In the above plot, Change the position argument to “dodge”

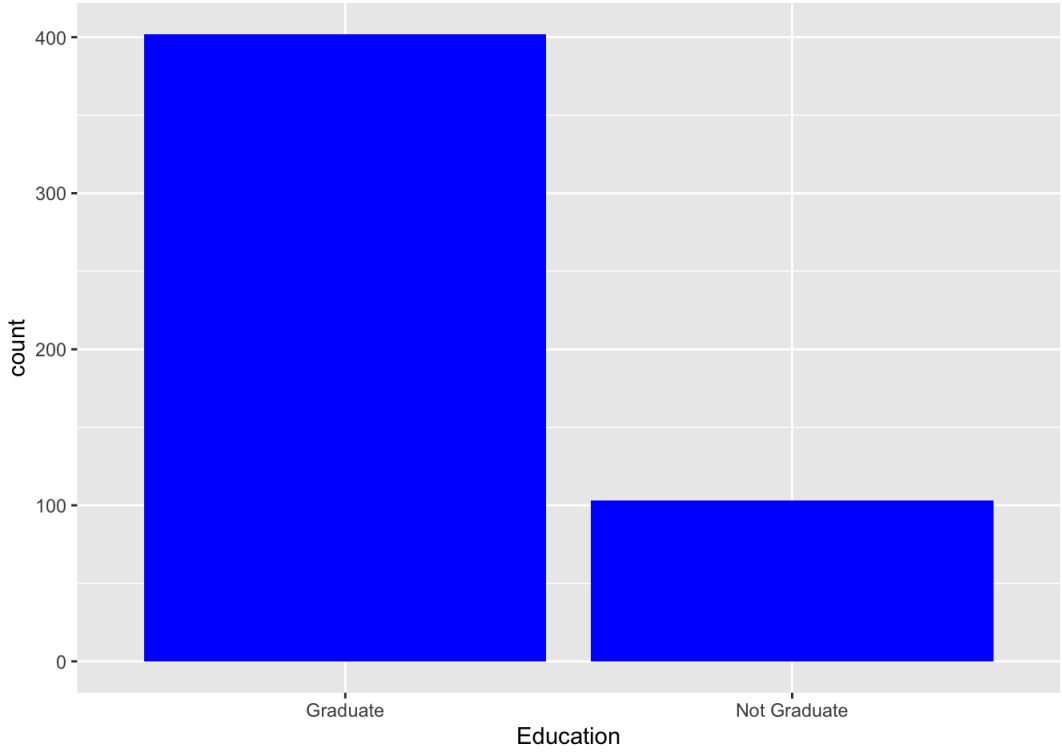
```
ggplot(data = hd,aes(x = Education, fill = Property_Purchased))+
  geom_bar()
```




```
ggplot(data = hd,aes(x = Education, stack = Property_Purchased))+
  geom_bar()
```

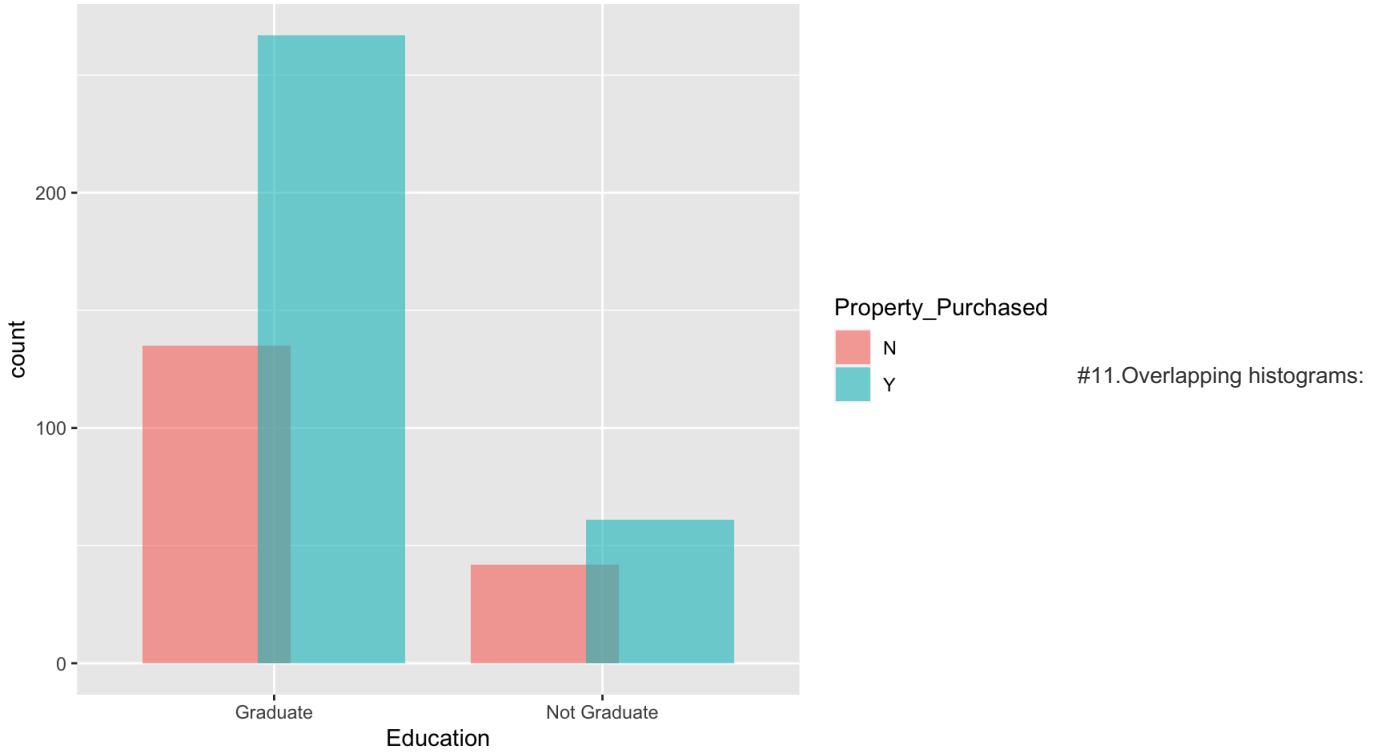



```
ggplot(data = hd,aes(x = Education, dodge = Property_Purchased))+
  geom_bar(fill = 'blue')
```



#10.Overlapping bar plots: #• Take the last plot form the previous exercise #• In the above plot, Define posn_d with position_dodge(). Take value as 0.7 #• Change the position argument to posn_d in the last plot made in Step 9(d) #• Use posn_d as position and adjust alpha to 0.6 - can you see the overlap in #bars. If not, change the value of alpha

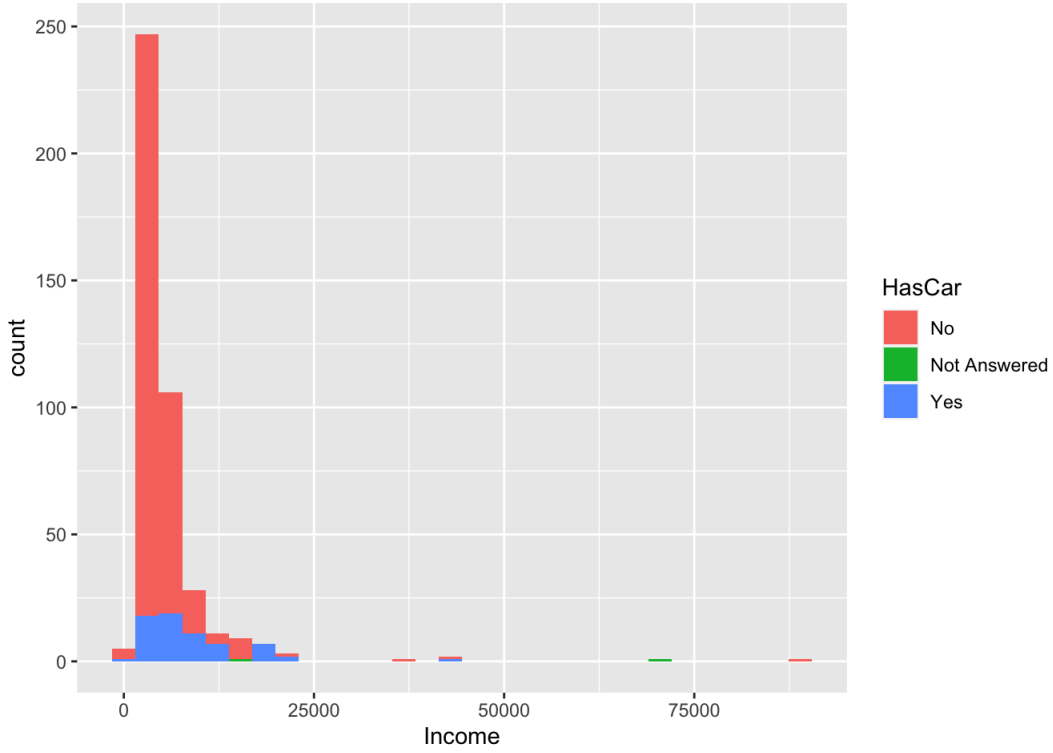
```
posn_d <- function(x){
  position_dodge(0.7)
}
ggplot(data = hd,aes(x = Education, fill = Property_Purchased))+
  geom_bar(position = posn_d(0.7),alpha = 0.6)
```



#11.Overlapping histograms:

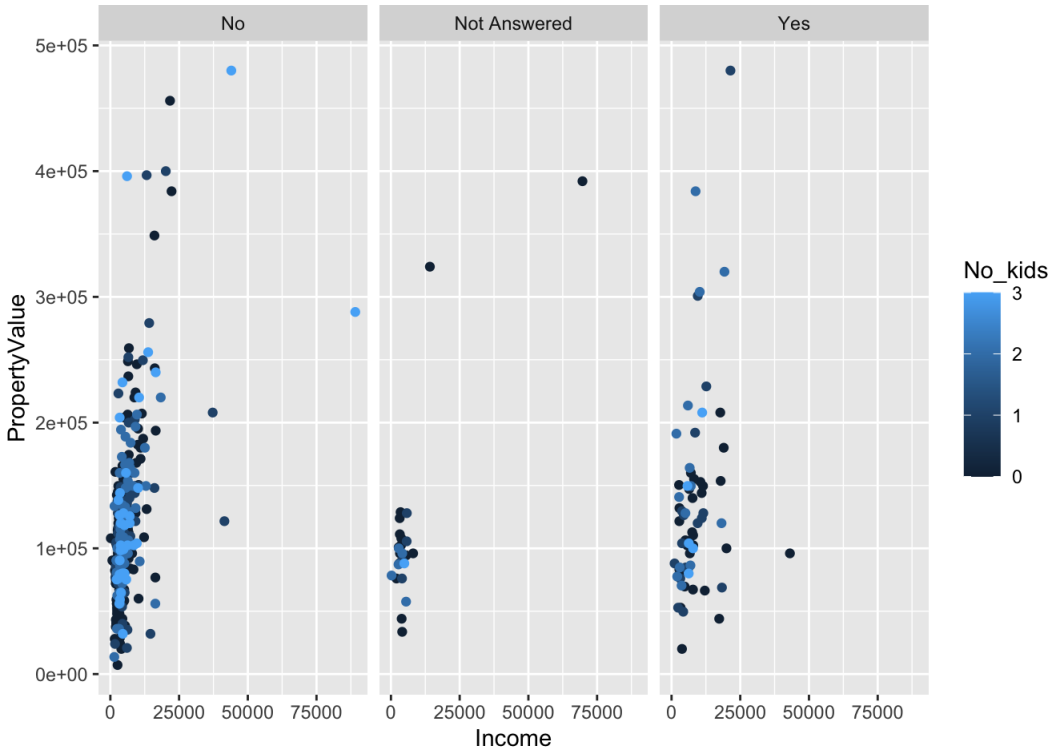
#• A basic histogram, add coloring defined by Income and filled by HasCar, select #a suitable binwidth #• In the above plot, In the above chart, Change position to identity

```
ggplot(data = hd,aes(x = Income,fill = HasCar,color = Income ))+
  geom_histogram(bins = 30,position = 'identity')
```

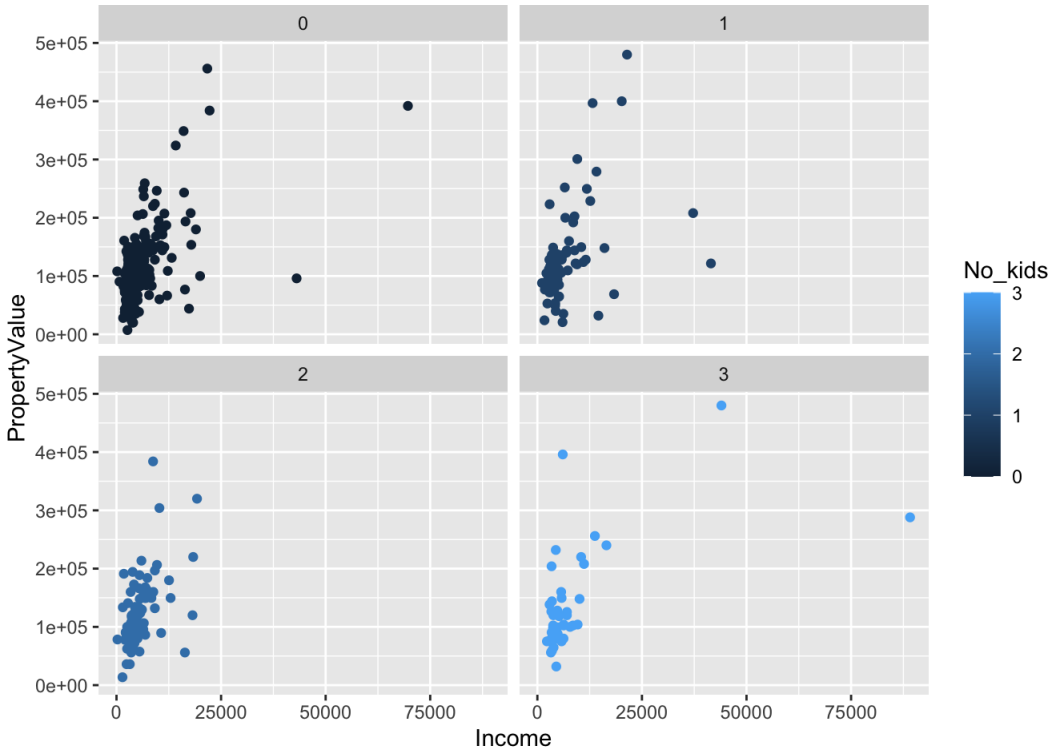


#12. Faceting: #• Now create a basic scatter plot between income and property value variables #• In the above plot, Separate rows according to HasCar #• In plot made in step 12b, Separate columns according to No of kids #• In plot made in step 12b, , Separate by both HasCar and No of kids

```
ggplot(data = hd,aes(x = Income, y = PropertyValue,col = No_kids))+
  geom_point()+
  facet_wrap(~HasCar)
```




```
ggplot(data = hd,aes(x = Income, y = PropertyValue,col = No_kids))+
  geom_point()+
  facet_wrap(~No_kids)
```




```
ggplot(data = hd,aes(x = Income, y = PropertyValue,col = No_kids))+
  geom_point()+
  facet_wrap(~No_kids +HasCar)
```