Solution Design: Bike Sharing Demand

Anushka Tak Chandan Manjunath Group 22 Section 4

March 19, 2019

Contribution

Anushka 50% Chandan 50%

Signature

Anushka Chandan

PROGRESS

We have done Initial Exploratory Data Analysis which includes

1) Data Distribution,

2) Data Preprocessing,

3) Data Analysis and

4) Variable Selection.

We have plotted several plots of which we include only the ones of interest.

In conclusion, we summarise

1) Our data is fairly clean, doesn't contain any missing values. However, many windspeed values are 0, which can make the model biased. Therefore, we replace zeroes in the windspeed with the mean value.

2) There are a few variables which are highy correlated, example, temperature and "feels like" temperature(atemp); and temperature and humidity. This can induce errors in our model in the future(Multicollinearity problems). Therefore, we decide to drop atemp variable.

3) We find casual and registered predictor do not provide us with any relevant information. Therefore, we decide to drop those in our Modelling phase.

4) The data is split into train and test, andwill be split further into validation during the modelling phase. For now, wehave used training dataset for EDA. The split ratio is 60:40.

5) The dataset included datetime stamp which we made cleaner for understanding purposes by splitting into date, hour, minute, second.

6) For variable selection, we've used Stepwise Regression and chose AIC and ANOVA results to find significant variables. Results are shown in the document. But most important variables found out were,

Temperature

Humidity

Hour

Month

Season

Working day

All these variables seem intuitive from the domain knowledge too. Additionally, Anova and Svatterplot Matrices tempt us to include windspeed, workingday to our model.

7) We needed to transform count variable to avoid the bias led by its skewed distribution. We used log transformation.

8) We plot several graphs to study predictor behavior, trends and outliers to gain insights about the market behavior. Some insights are

a) Highest number of bike rentals are in the month of May, June, July, August, Sept, Oct. This could possibly be because of the pleasant weather conditions in those months(location North Hemisphere, WaSHINGTON D.c.)

b) Season graph supports this with higher bike rentals during Summer and Fall than Spring.

c) Pattern in hours of the day with respect to bike rentals is more during rush hours(7 am-8 pm).

d) We have included three predictor graphs to see an influence of each on the other. We observe as temperature drops, bike usage drops with hours the same from the above graph.

10) Outcome

Our response variable is a sum of bikes predicted to be rented in a particular hour. This anticipation will help the market profit of the commercial business, so that the supply always meets the demand. The problem we aim to address is to optimize the business approach of bike rentals.

11) Approach

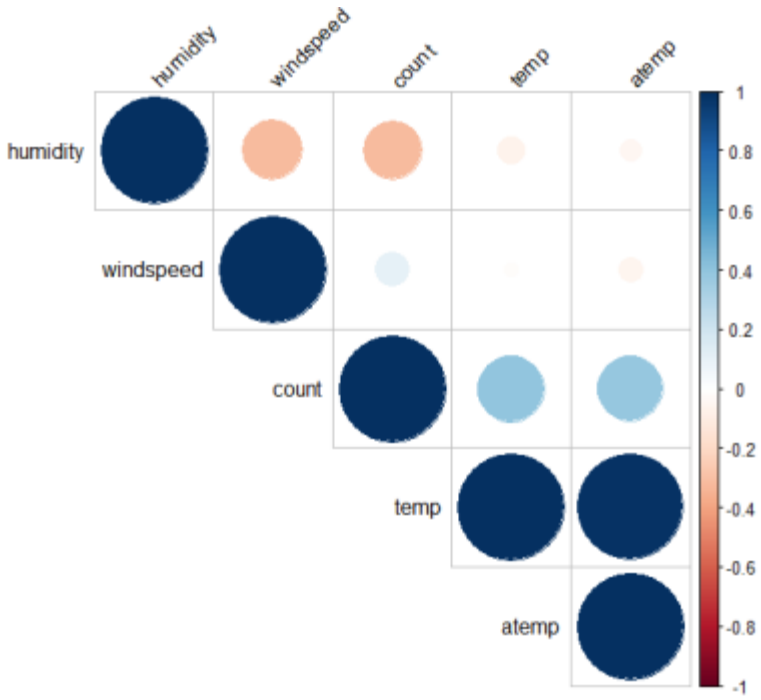
We intend to use various prediction algorithms to predict a count for a particular hour using our developed model. We are going to implement following algorithms in order to achieve our aim and will evaluate performance using RMSE criterion.

a) *Linear Regression*
b) *Random Forest*
c) *Gradient Boosting Regression*

12) Data Origin

We made an effort to reach out to Boston Bixi to collect Bike rental information of recent years in Boston itself, but couldn't manage to get it. Hence, we decided to go forward with publicly available data from a Washington D.C. based comapny 'Capital Bike Share' with data spanning two years from 2011.

13) Correlation Plot



Exploratory Data Analysis

Data Distribution

Libraries

```
library(MASS)
library(ggplot2)
library(car)
```

```
library(psych)
library(dplyr)
library(tidyverse)
library(lubridate)
library(dlookr)
```

Data Loading

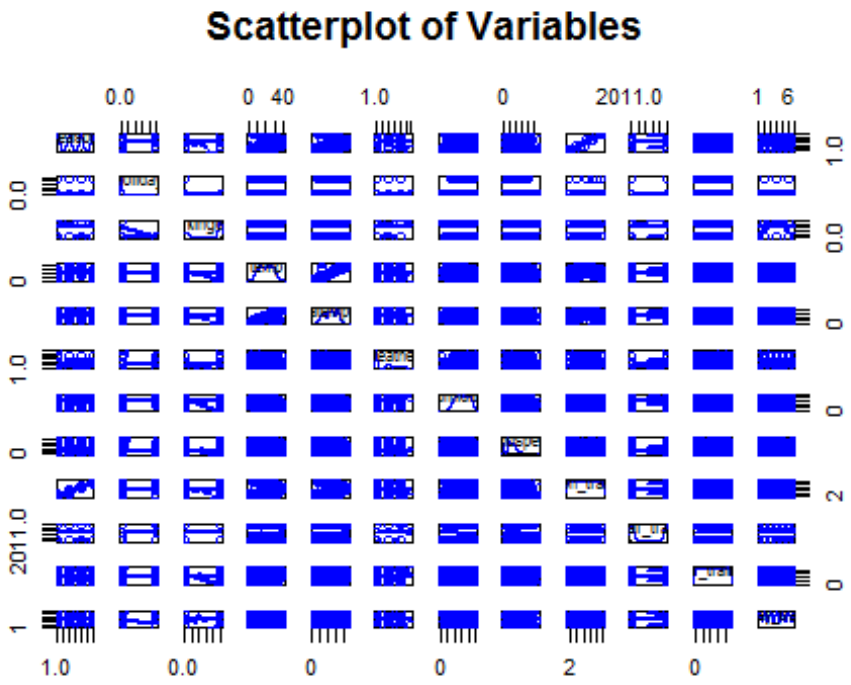
```
train<-read_csv("C:/Users/pc/Desktop/Spring2019/DM/dm_project/train.csv")
  test<-read_csv("C:/Users/pc/Desktop/Spring2019/DM/dm_project/test.csv")
  head(train,n=5)
```

```
## # A tibble: 5 x 12
##   datetime            season holiday workingday weather  temp atemp
##   <dttm>               <int>   <int>      <int>   <int> <dbl> <dbl>
## 1 2011-01-01 00:00:00      1       0          0       1  9.84  14.4
## 2 2011-01-01 01:00:00      1       0          0       1  9.02  13.6
## 3 2011-01-01 02:00:00      1       0          0       1  9.02  13.6
## 4 2011-01-01 03:00:00      1       0          0       1  9.84  14.4
## 5 2011-01-01 04:00:00      1       0          0       1  9.84  14.4
## # ... with 5 more variables: humidity <int>, windspeed <dbl>,
## #   casual <int>, registered <int>, count <int>
```

Making sense of variables 1) Dropping casual and registered variables as per the domain knowledge. 2) Separating datetime into hour min day date month

```
  time_train <- ymd_hms(train$datetime)
  year_train<-year(time_train)
  mon_train<-month(time_train)
  hr_train<-hour(time_train)
  minute_train<-minute(time_train)
  sec_train<-second(time_train)
  wday_train<-wday(time_train ,label=TRUE)
  date_train<-date(time_train)

  time_test <- ymd_hms(test$datetime)
  year_test<-year(time_test)
  mon_test<-month(time_test)
  hr_test<-hour(time_test)
  minute_test<-minute(time_test)
  sec_test<-second(time_test)
  wday_test<-wday(time_test ,label=TRUE)
 date_test<-date(time_test)

train_new<-
transmute(train,season,holiday,workingday,temp,atemp,weather,humidity,windspe
ed,date_train,year_train,mon_train,hr_train,minute_train,sec_train,wday_train
,count)
test_new<-
transmute(test,season,holiday,workingday,temp,atemp,weather,humidity,windspee
d,date_test,year_test,mon_test,hr_test,minute_test,sec_test,wday_test)
```

```
scatterplotMatrix(~season+holiday+workingday+temp+atemp+weather+humidity+wind
speed+mon_train+year_train+hr_train+wday_train, data=train_new,
   main="Scatterplot of Variables")
```




```
attach(train_new)
cor(train_new[c("temp","atemp","humidity","windspeed","count")])
```

```
##                  temp       atemp    humidity   windspeed       count
## temp       1.00000000  0.98494811 -0.06494877 -0.01785201  0.3944536
## atemp      0.98494811  1.00000000 -0.04353571 -0.05747300  0.3897844
## humidity  -0.06494877 -0.04353571  1.00000000 -0.31860699 -0.3173715
## windspeed -0.01785201 -0.05747300 -0.31860699  1.00000000  0.1013695
## count      0.39445364  0.38978444 -0.31737148  0.10136947  1.0000000
```

```
detach(train_new)
```

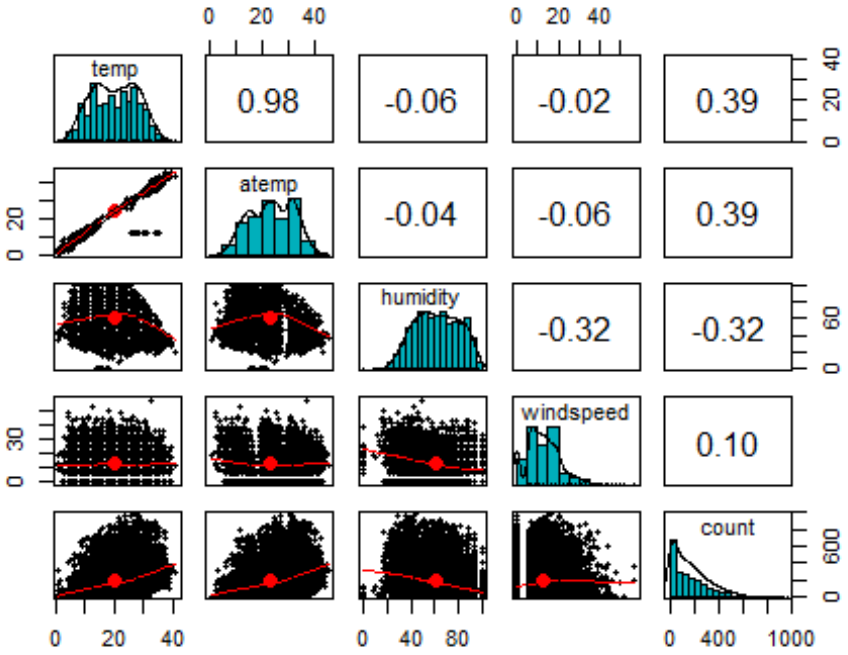
Data Preprocessing

Substituting 0 with mean values (Windspeed)

```
train_new[!complete.cases(train_new),]
```

```
## # A tibble: 0 x 16
## # ... with 16 variables: season <int>, holiday <int>, workingday <int>,
## #   temp <dbl>, atemp <dbl>, weather <int>, humidity <int>,
```

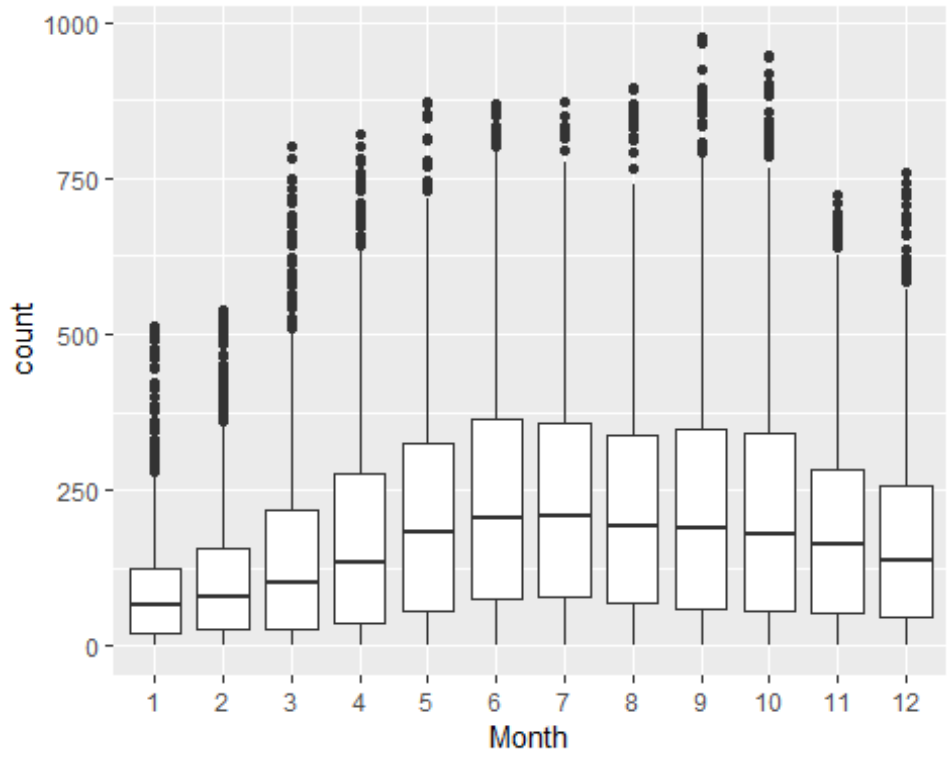
```
## #   windspeed <dbl>, date_train <date>, year_train <dbl>, mon_train <dbl>,
## #   hr_train <int>, minute_train <int>, sec_train <dbl>, wday_train <ord>,
## #   count <int>

diagnose(train_new)

## # A tibble: 16 x 6
##    variables  types missing_count missing_percent unique_count unique_rate
##    <chr>      <chr>         <int>           <dbl>        <int>       <dbl>
##  1 season     inte~             0               0            4   0.000367
##  2 holiday    inte~             0               0            2   0.000184
##  3 workingday inte~             0               0            2   0.000184
##  4 temp       nume~             0               0           49   0.00450
##  5 atemp      nume~             0               0           60   0.00551
##  6 weather    inte~             0               0            4   0.000367
##  7 humidity   inte~             0               0           89   0.00818
##  8 windspeed  nume~             0               0           28   0.00257
##  9 date_train Date              0               0          456   0.0419
## 10 year_train nume~             0               0            2   0.000184
## 11 mon_train  nume~             0               0           12   0.00110
## 12 hr_train   inte~             0               0           24   0.00220
## 13 minute_tr~ inte~             0               0            1   0.0000919
## 14 sec_train  nume~             0               0            1   0.0000919
## 15 wday_train orde~             0               0            7   0.000643
## 16 count      inte~             0               0          822   0.0755

pairs.panels(train_new[,c(-1,-2,-3,-6,-9,-10,-11,-12,-13,-14,-
15)],method="pearson",hist.col = "#00AFBB",density=TRUE)
```



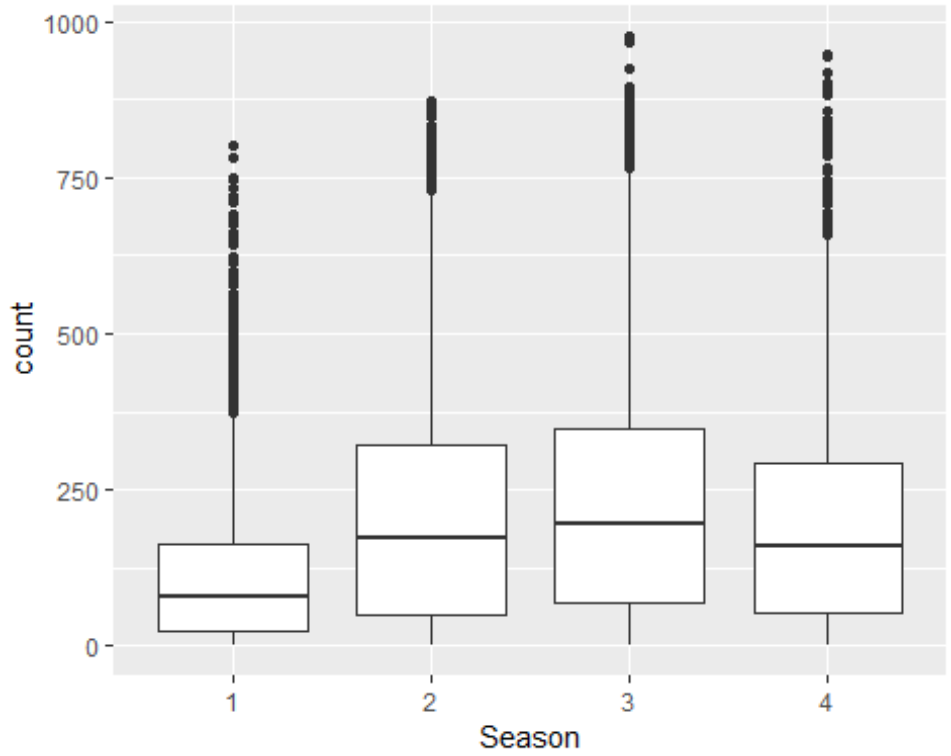

```
p<-mean(train_new$windspeed)
if (as.factor(train_new$windspeed)==0){train_new$windspeed<-p}
```

Data Analysis/ Trends/ Outliers

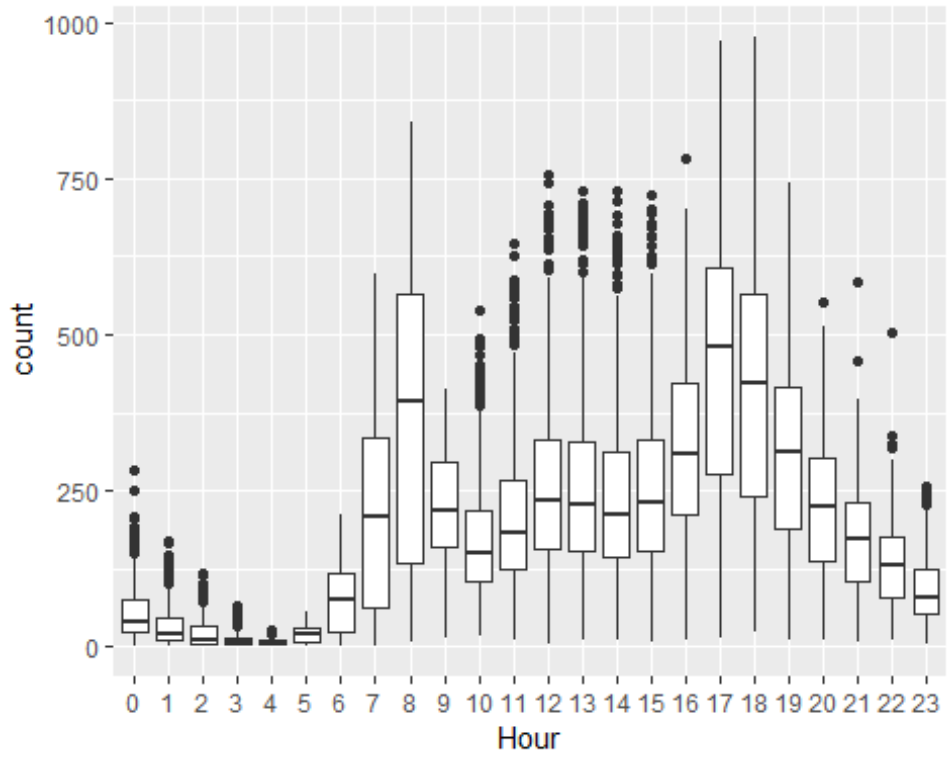
```
ggplot(train_new)+geom_boxplot(aes(x=factor(mon_train),y=count))+labs(x="Mont
h")
```



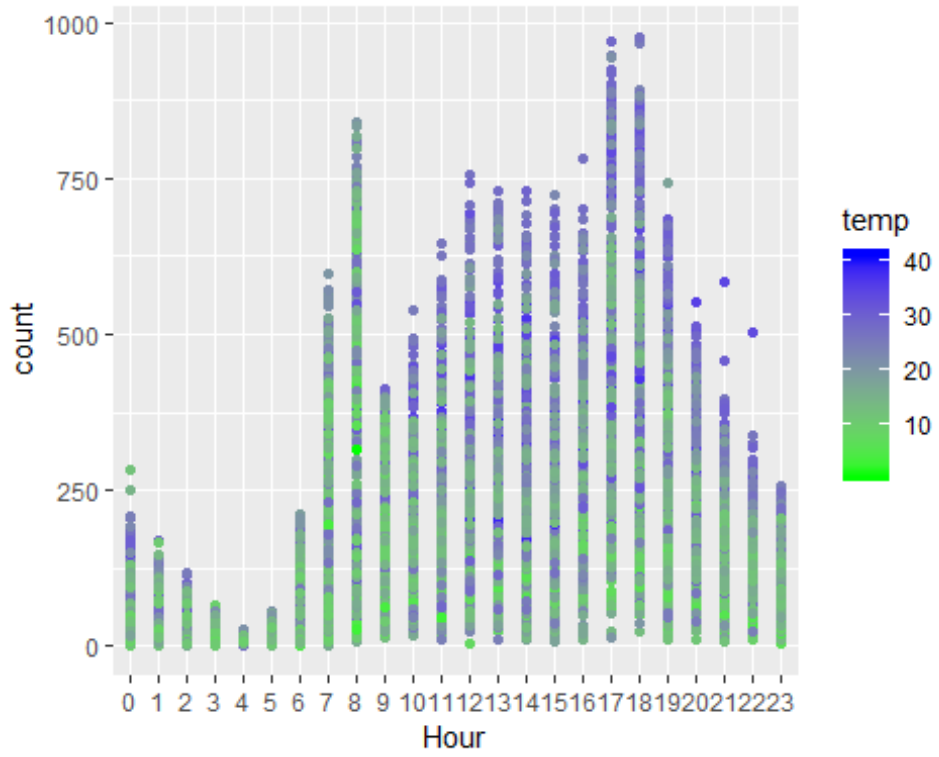

```
ggplot(train_new)+geom_boxplot(aes(x=factor(season),y=count))+labs(x="Season"
)
```



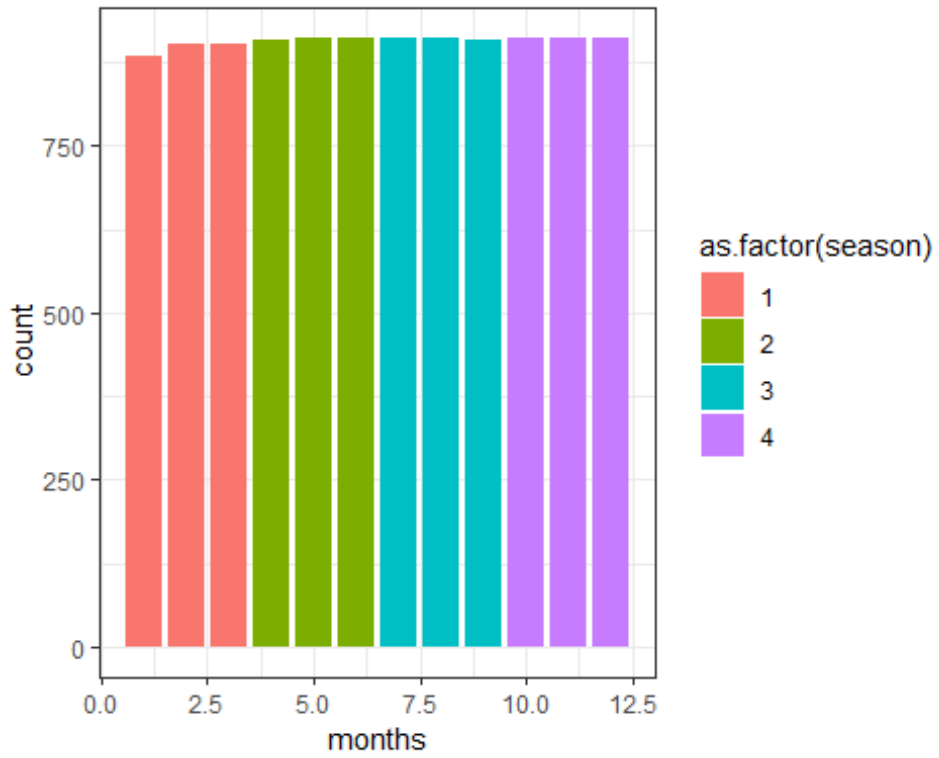
```
ggplot(train_new)+geom_boxplot(aes(x=factor(hr_train),y=count))+labs(x="Hour"
)
```



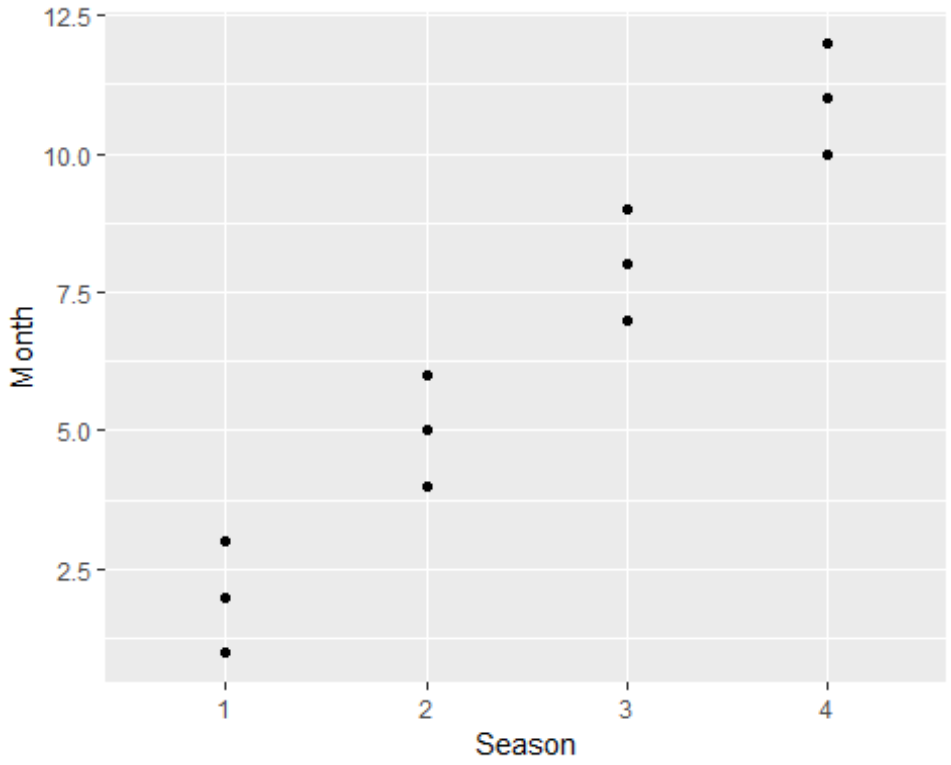

```
ggplot(train_new)+geom_point(aes(x=factor(hr_train),y=count,color=temp))+labs
(x="Hour")+
  scale_color_gradient(high="blue",low="green")
```



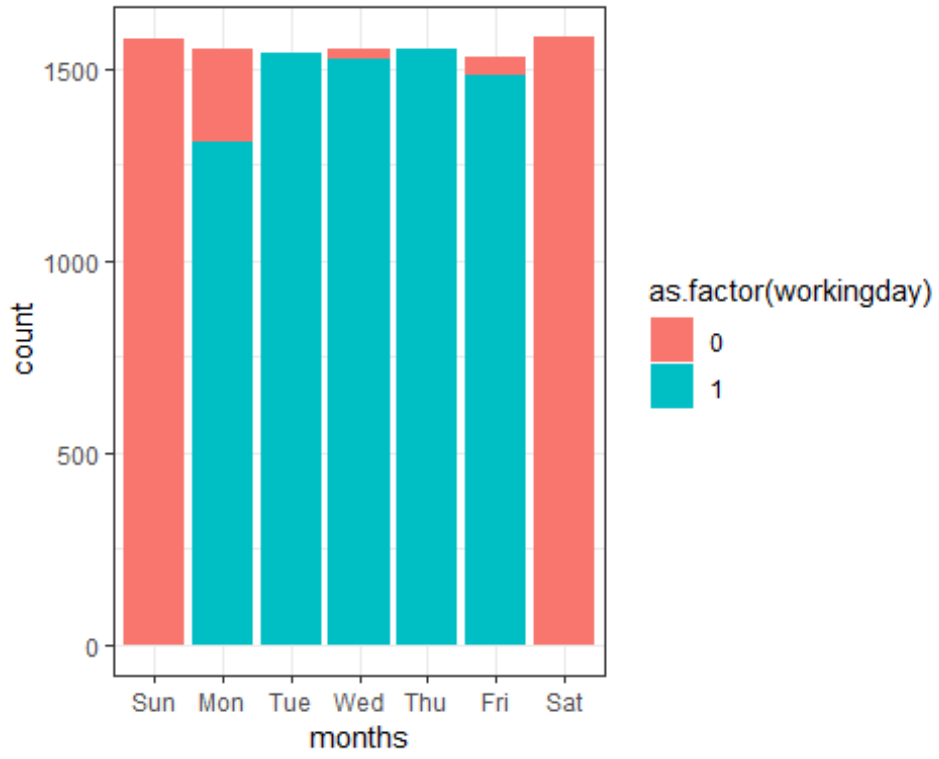

```
ggplot(train_new)+geom_bar(aes(x=mon_train,fill=as.factor(season)))+theme_bw(
)+labs(x="months")
```



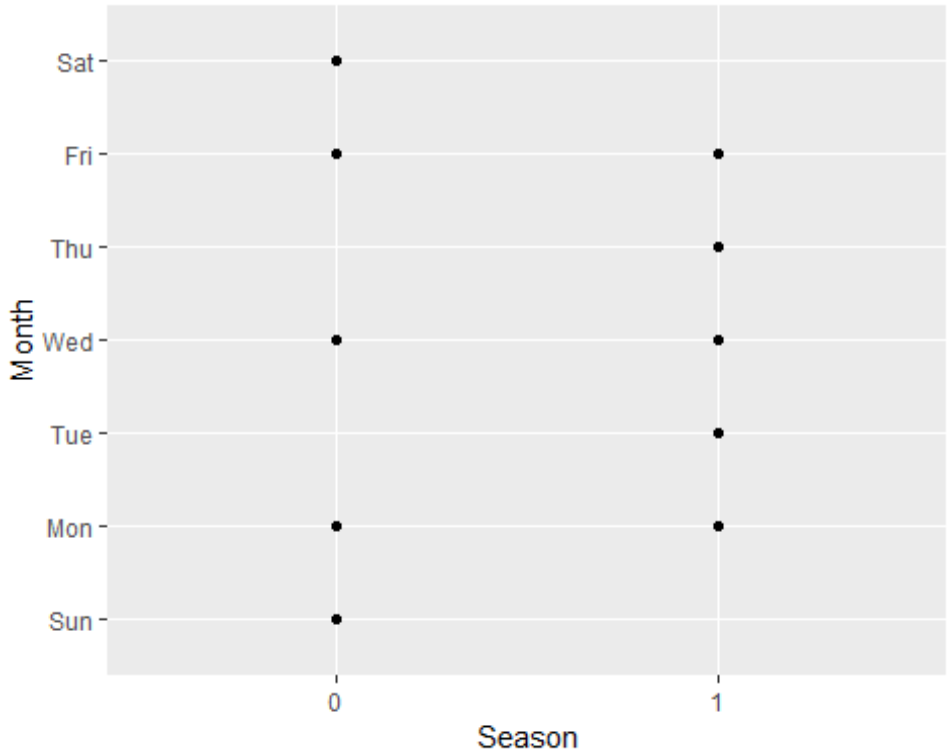
```
ggplot(train_new)+geom_point(aes(y=mon_train,x=as.factor(season)))+labs(x="Se
ason",y="Month")
```




```
ggplot(train_new)+geom_bar(aes(x=wday_train,fill=as.factor(workingday)))+them
e_bw()+labs(x="months")
```




```
ggplot(train_new)+geom_point(aes(y=wday_train,x=as.factor(workingday)))+labs(
x="Season",y="Month")
```



```
fit<-
lm(log(count)~season+holiday+workingday+temp+humidity+atemp+weather+hr_train+
humidity+windspeed+mon_train,data=train_new)
step<-stepAIC(fit,direction = "both")
```

```
## Start:  AIC=1989.82
## log(count) ~ season + holiday + workingday + temp + humidity +
##     atemp + weather + hr_train + humidity + windspeed + mon_train
##
##
## Step:  AIC=1989.82
## log(count) ~ season + holiday + workingday + temp + humidity +
##     atemp + weather + hr_train + mon_train
##
##              Df Sum of Sq   RSS    AIC
## - weather     1       0.7 13046 1988.4
## - holiday     1       0.7 13046 1988.4
## - season      1       2.3 13048 1989.8
## <none>                    13045 1989.8
## - temp        1       4.2 13050 1991.3
## - workingday  1      20.1 13065 2004.6
## - atemp       1      23.1 13068 2007.1
## - mon_train   1      45.0 13090 2025.3
## - humidity    1     796.5 13842 2633.0
## - hr_train    1    4840.8 17886 5423.4
##
## Step:  AIC=1988.37
## log(count) ~ season + holiday + workingday + temp + humidity +
##     atemp + hr_train + mon_train
##
##              Df Sum of Sq   RSS    AIC
## - holiday     1       0.7 13047 1986.9
## - season      1       2.3 13048 1988.3
## <none>                    13046 1988.4
## + weather     1       0.7 13045 1989.8
## - temp        1       4.4 13050 1990.1
## - workingday  1      19.8 13066 2002.9
## - atemp       1      22.7 13069 2005.3
## - mon_train   1      44.7 13091 2023.6
## - humidity    1     948.1 13994 2750.1
## - hr_train    1    4913.7 17960 5466.1
##
## Step:  AIC=1986.92
## log(count) ~ season + workingday + temp + humidity + atemp +
##     hr_train + mon_train
##
##              Df Sum of Sq   RSS    AIC
## <none>                    13047 1986.9
## - season      1       2.6 13049 1987.1
## + holiday     1       0.7 13046 1988.4
```

```
## + weather     1       0.7 13046 1988.4
## - temp        1       4.3 13051 1988.5
## - workingday  1      19.2 13066 2001.0
## - atemp       1      23.1 13070 2004.2
## - mon_train   1      46.5 13093 2023.7
## - humidity    1     948.6 13995 2749.0
## - hr_train    1    4913.6 17960 5464.4
```

```
step$anova
```

```
## Stepwise Model Path
## Analysis of Deviance Table
##
## Initial Model:
## log(count) ~ season + holiday + workingday + temp + humidity +
##     atemp + weather + hr_train + humidity + windspeed + mon_train
##
## Final Model:
## log(count) ~ season + workingday + temp + humidity + atemp +
##     hr_train + mon_train
##
##
##          Step Df  Deviance Resid. Df Resid. Dev      AIC
## 1                              10876   13045.30 1989.825
## 2 - windspeed  0 0.0000000     10876   13045.30 1989.825
## 3   - weather  1 0.6518987     10877   13045.95 1988.368
## 4   - holiday  1 0.6656661     10878   13046.62 1986.924
```

```
anova(fit)
```

```
## Analysis of Variance Table
##
## Response: log(count)
##               Df  Sum Sq Mean Sq   F value    Pr(>F)
## season         1   626.6   626.6  522.3830 < 2.2e-16 ***
## holiday        1     0.6     0.6    0.5375  0.463470
## workingday     1     8.4     8.4    7.0419  0.007974 **
## temp           1  2867.6  2867.6 2390.7103 < 2.2e-16 ***
## humidity       1  2574.0  2574.0 2145.9791 < 2.2e-16 ***
## atemp          1    34.7    34.7   28.9236 7.685e-08 ***
## weather        1    70.7    70.7   58.9236 1.780e-14 ***
## hr_train       1  4861.9  4861.9 4053.4044 < 2.2e-16 ***
## mon_train      1    45.0    45.0   37.4868 9.521e-10 ***
## Residuals  10876 13045.3     1.2
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

We observe temp and atemp are correlated to each other. Hence, we decide to drop one of these variables, in order to avoid multicollinearity. Moreover, neither gives additional information, hence it is safe to drop one of these variables.