Bike Sharing Dataset

Nurzhanat Zhussup

Contents

Introduction

Bike sharing systems are new generation of traditional bike rentals where whole process from membership, rental and return back has become automatic. Through these systems, user is able to easily rent a bike from a particular position and return back at another position. Currently, there are about over 500 bike-sharing programs around the world which is composed of over 500 thousands bicycles. Today, there exists great interest in these systems due to their important role in traffic, environmental and health issues.

Motivation for research

Leveraging User Behavior for Bike-Sharing Business Success Bike-sharing systems have transformed urban transportation. Understanding user behavior is pivotal for enhancing operational efficiency and crafting effective marketing strategies. This research seeks to unveil user patterns impacted by seasons, weather, weekdays, and holidays. The insights generated will empower bike-sharing businesses to optimize resources and attract and retain users, fostering system growth and economic sustainability.

Research questions

- How different are bike rental behaviors between casual and registered users?
- What are the bike rental patterns across seasons and months
- What is the impact of different weather conditions on bike rental
- Is there any significant differences in bike rental on holidays and workdays?
- Which variables are most important in predicting total number bike rentals?

Data description

```
- instant: record index
- dteday : date
- season : season (1:winter, 2:spring, 3:summer, 4:fall)
- yr : year (0: 2011, 1:2012)
- mnth : month ( 1 to 12)
- hr : hour (0 to 23)
- holiday : weather day is holiday or not (extracted from http://dchr.dc.gov/page/holiday-schedule)
- weekday : day of the week
- workingday : if day is neither weekend nor holiday is 1, otherwise is 0.
- weathersit :
    - 1: Clear, Few clouds, Partly cloudy, Partly cloudy
    - 2: Mist + Cloudy, Mist + Broken clouds, Mist + Few clouds, Mist
    - 3: Light Snow, Light Rain + Thunderstorm + Scattered clouds, Light Rain + Scattered clouds
    - 4: Heavy Rain + Ice Pallets + Thunderstorm + Mist, Snow + Fog
- temp : Normalized temperature in Celsius. The values are derived via (t-t_min)/(t_max-t_min), t_min=-
- atemp: Normalized feeling temperature in Celsius. The values are derived via (t-t_min)/(t_max-t_min),
- hum: Normalized humidity. The values are divided to 100 (max)
- windspeed: Normalized wind speed. The values are divided to 67 (max)
- casual: count of casual users
- registered: count of registered users
- cnt: count of total rental bikes including both casual and registered
```

Data

Data import

```r
bike <- read.csv(file = "../bike+sharing+dataset/hour.csv", header = TRUE)
```

Data structure

```r
str(bike)
```

```
## 'data.frame':    17379 obs. of  17 variables:
##  $ instant   : int  1 2 3 4 5 6 7 8 9 10 ...
##  $ dteday    : chr  "2011-01-01" "2011-01-01" "2011-01-01" "2011-01-01" ...
##  $ season    : int  1 1 1 1 1 1 1 1 1 1 ...
##  $ yr        : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ mnth      : int  1 1 1 1 1 1 1 1 1 1 ...
##  $ hr        : int  0 1 2 3 4 5 6 7 8 9 ...
##  $ holiday   : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ weekday   : int  6 6 6 6 6 6 6 6 6 6 ...
##  $ workingday: int  0 0 0 0 0 0 0 0 0 0 ...
##  $ weathersit: int  1 1 1 1 1 2 1 1 1 1 ...
##  $ temp      : num  0.24 0.22 0.22 0.24 0.24 0.24 0.22 0.2 0.24 0.32 ...
##  $ atemp     : num  0.288 0.273 0.273 0.288 0.288 ...
##  $ hum       : num  0.81 0.8 0.8 0.75 0.75 0.75 0.8 0.86 0.75 0.76 ...
##  $ windspeed : num  0 0 0 0 0 0.0896 0 0 0 0 ...
##  $ casual    : int  3 8 5 3 0 0 2 1 1 8 ...
##  $ registered: int  13 32 27 10 1 1 0 2 7 6 ...
##  $ cnt       : int  16 40 32 13 1 1 2 3 8 14 ...
```

Duplicated values

```r
bike[duplicated(bike),]
```

```
##  [1] instant    dteday     season     yr         mnth       hr
##  [7] holiday    weekday    workingday weathersit temp       atemp
## [13] hum        windspeed  casual     registered cnt
## <0 rows> (or 0-length row.names)
```

NO DUPLICATED DATA

Data formatting

```r
bike$dteday <- NULL
bike$instant <- NULL
```

Delete the dteday variable because the information as year, month, weekday, holiday, workingday, season and hour are already extracted.

Delete the instant variable because it's just the row count

Data summary

```r
summary(bike)
```

```
##      season            yr              mnth              hr
##  Min.   :1.000   Min.   :0.0000   Min.   : 1.000   Min.   : 0.00
##  1st Qu.:2.000   1st Qu.:0.0000   1st Qu.: 4.000   1st Qu.: 6.00
##  Median :3.000   Median :1.0000   Median : 7.000   Median :12.00
##  Mean   :2.502   Mean   :0.5026   Mean   : 6.538   Mean   :11.55
##  3rd Qu.:3.000   3rd Qu.:1.0000   3rd Qu.:10.000   3rd Qu.:18.00
##  Max.   :4.000   Max.   :1.0000   Max.   :12.000   Max.   :23.00
```

```
##     holiday           weekday        workingday       weathersit
##  Min.   :0.00000   Min.   :0.000   Min.   :0.0000   Min.   :1.000
##  1st Qu.:0.00000   1st Qu.:1.000   1st Qu.:0.0000   1st Qu.:1.000
##  Median :0.00000   Median :3.000   Median :1.0000   Median :1.000
##  Mean   :0.02877   Mean   :3.004   Mean   :0.6827   Mean   :1.425
##  3rd Qu.:0.00000   3rd Qu.:5.000   3rd Qu.:1.0000   3rd Qu.:2.000
##  Max.   :1.00000   Max.   :6.000   Max.   :1.0000   Max.   :4.000
##       temp           atemp              hum           windspeed
##  Min.   :0.020   Min.   :0.0000   Min.   :0.0000   Min.   :0.0000
##  1st Qu.:0.340   1st Qu.:0.3333   1st Qu.:0.4800   1st Qu.:0.1045
##  Median :0.500   Median :0.4848   Median :0.6300   Median :0.1940
##  Mean   :0.497   Mean   :0.4758   Mean   :0.6272   Mean   :0.1901
##  3rd Qu.:0.660   3rd Qu.:0.6212   3rd Qu.:0.7800   3rd Qu.:0.2537
##  Max.   :1.000   Max.   :1.0000   Max.   :1.0000   Max.   :0.8507
##      casual         registered         cnt
##  Min.   :  0.00   Min.   :  0.0   Min.   :  1.0
##  1st Qu.:  4.00   1st Qu.: 34.0   1st Qu.: 40.0
##  Median : 17.00   Median :115.0   Median :142.0
##  Mean   : 35.68   Mean   :153.8   Mean   :189.5
##  3rd Qu.: 48.00   3rd Qu.:220.0   3rd Qu.:281.0
##  Max.   :367.00   Max.   :886.0   Max.   :977.0
```

All variables are already numeric.

Missing values

```r
colSums(is.na(bike))
```

```
##     season         yr       mnth         hr    holiday    weekday workingday
##          0          0          0          0          0          0          0
## weathersit       temp      atemp        hum  windspeed     casual registered
##          0          0          0          0          0          0          0
##        cnt
##          0
```

NO MISSING VALUES

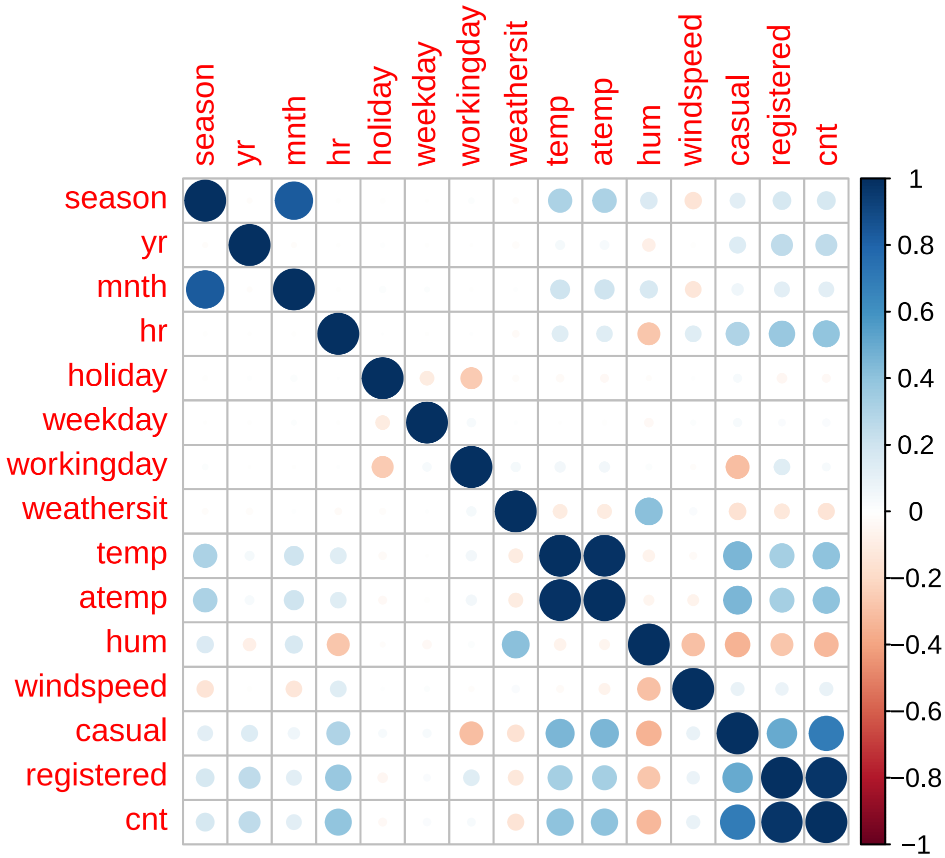
Exploratory data analysis

```r
library(corrplot)
```

```
## corrplot 0.92 loaded
```

Correlation

```r
corrplot(cor(bike[,sapply(bike, is.numeric)]))
```



There seem to be:

- Very strong correlation between cnt and registered, but it's only correlated because cnt = casual + registered. Next visual showcases this situation.
- Very strong correlation between atemp and temp, but it's explained by the fact that atemp is just a feeling temperature of the nominal temperature
- Strong correlation between season and month, but it's explained logically that the seasons include months. So these correlations aren't significant for our predictions.

In fact, cnt is good correlated with the following variables:

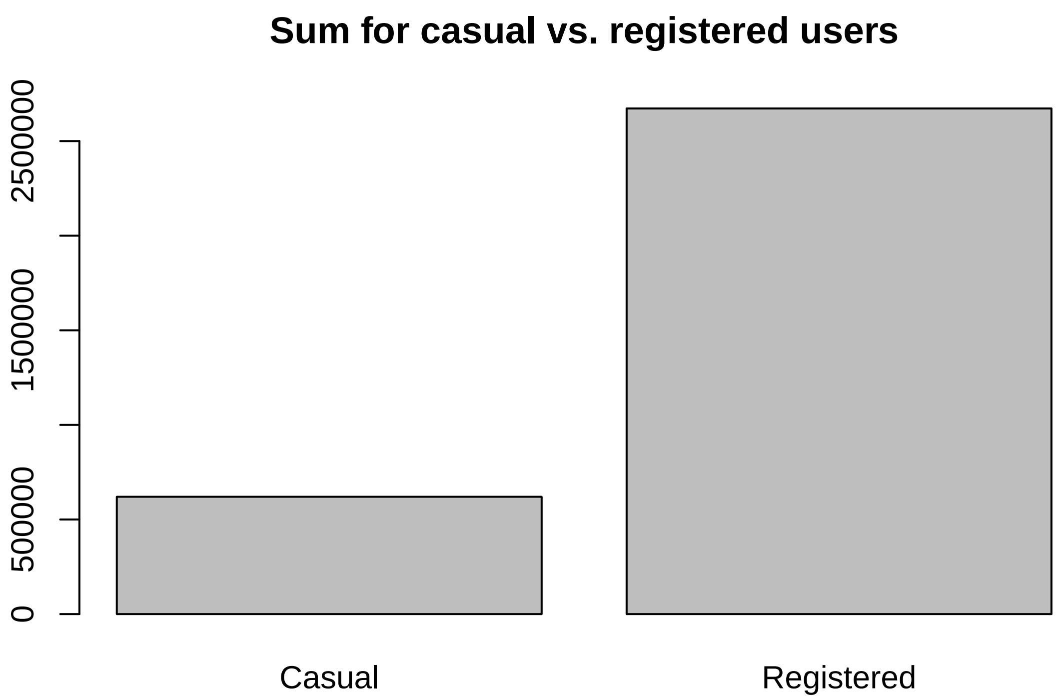
- hr (positive)
- temp & atemp (positive)
- hum (negative)

Next visuals will show in details the relationship between cnt and each independent variable

Casual vs. Registered

Differences in rental behaviors between casual and registered users

```
barplot(c(sum(bike$casual), sum(bike$registered)), names.arg = c("Casual", "Registered"), main = "Sum fo
```

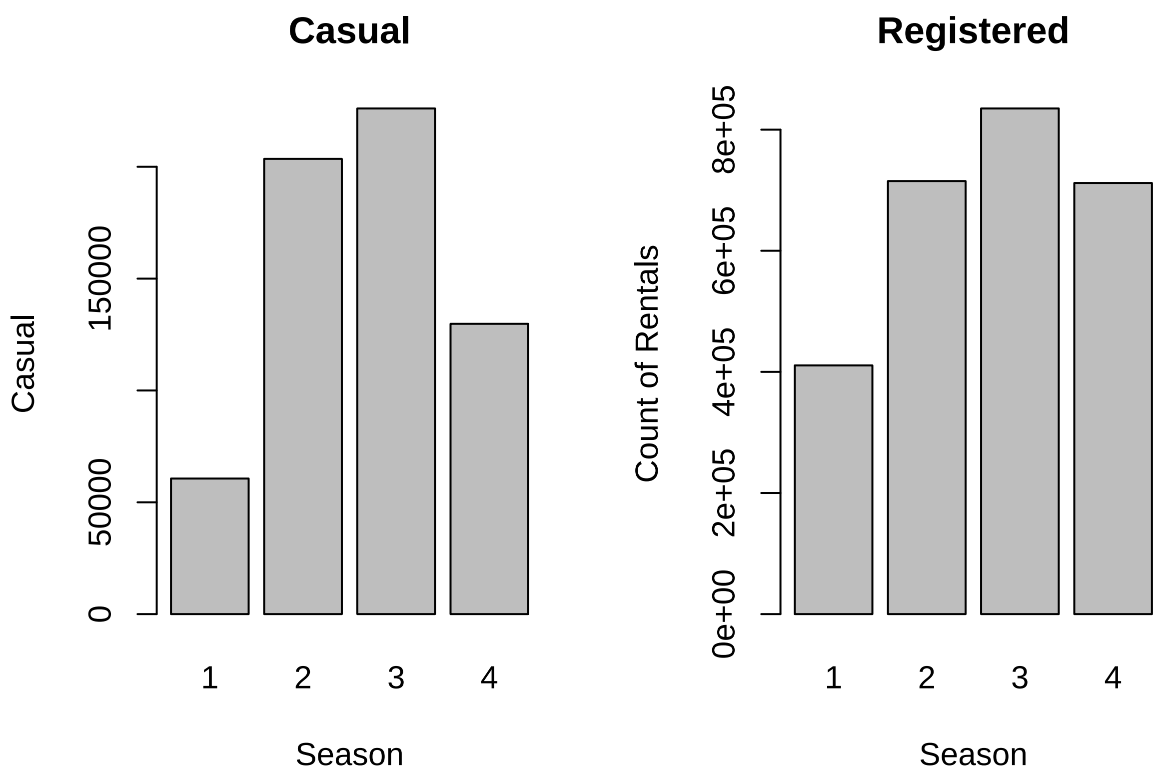


This basically explains why cnt is highly correlated with registered. There are a lot more registered customers than casual. cnt is just the sum of casual and registered.

by season

```
par(mfrow = c(1,2))

barplot(tapply(bike$casual, bike$season, sum), beside = TRUE,
        main = "Casual",
        xlab = "Season",
        ylab = "Casual")
barplot(tapply(bike$registered, bike$season, sum), beside = TRUE,
        main = "Registered",
        xlab = "Season",
        ylab = "Count of Rentals")
```

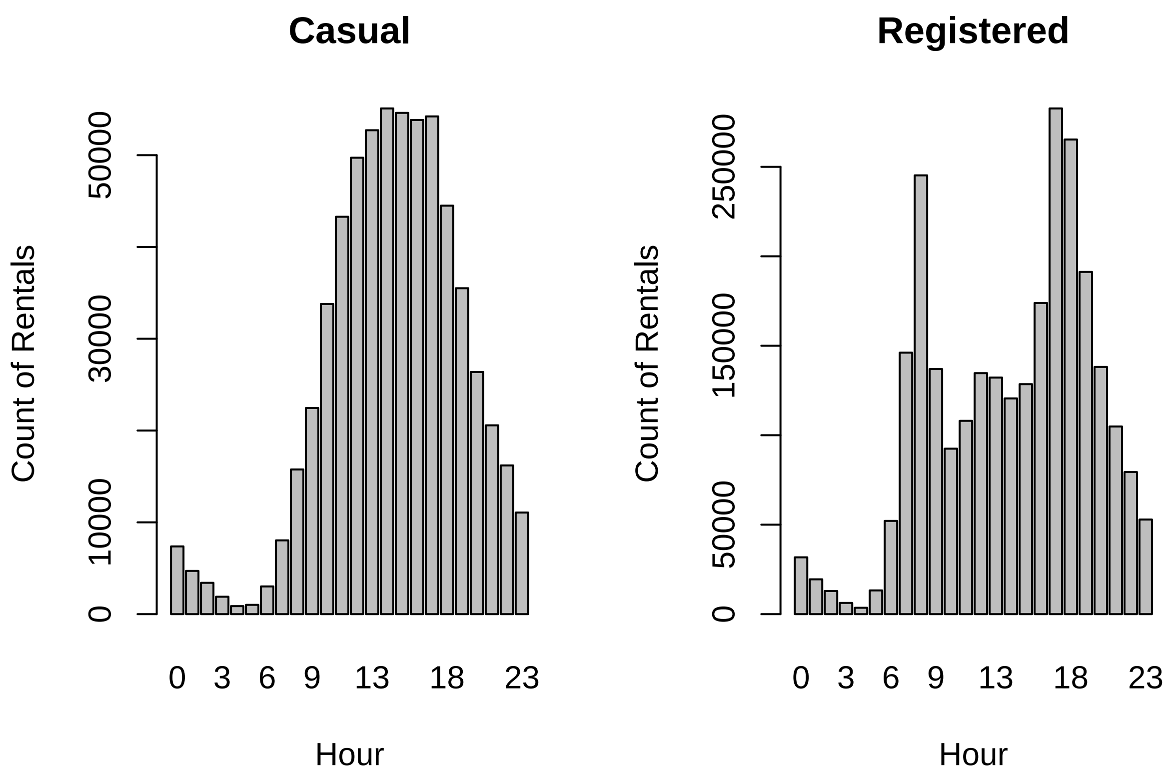


Registered customers have more bike rentals in winter and fall than casual customers.

by hour

```r
par(mfrow = c(1,2))

barplot(tapply(bike$casual, bike$hr, sum), beside = TRUE,
        main = "Casual",
        xlab = "Hour",
        ylab = "Count of Rentals")
barplot(tapply(bike$registered, bike$hr, sum), beside = TRUE,
        main = "Registered",
        xlab = "Hour",
        ylab = "Count of Rentals")
```

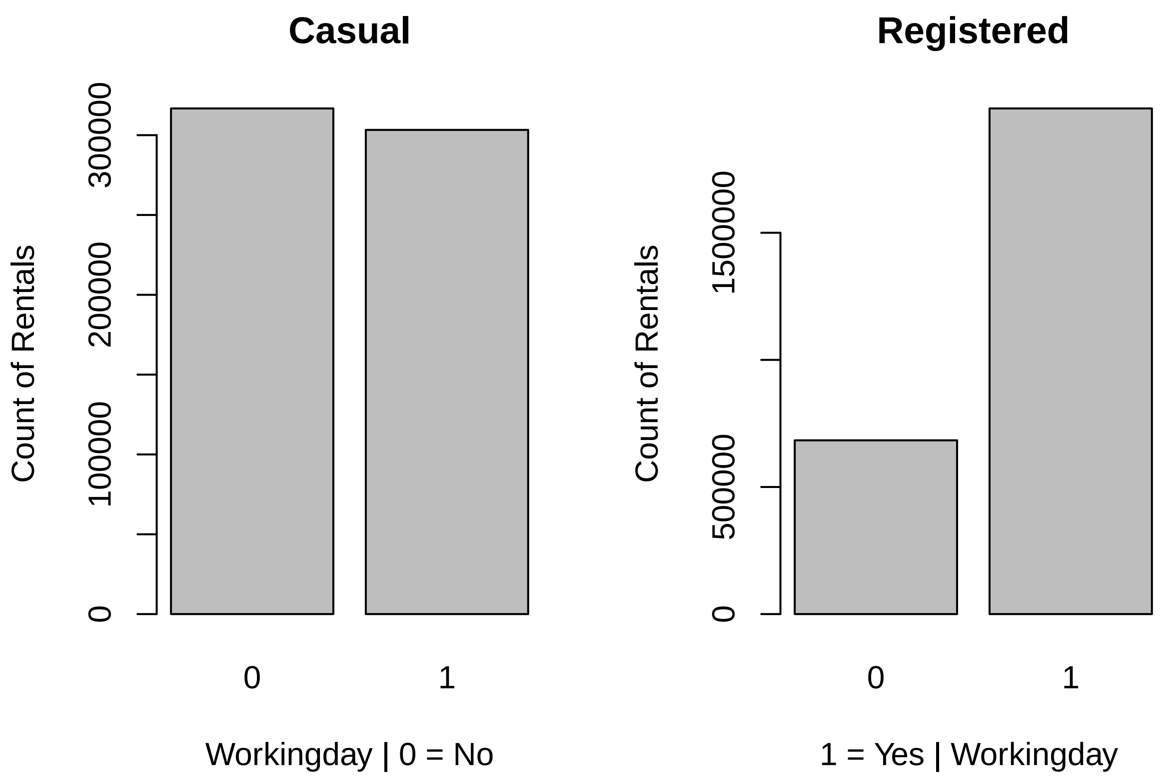


Casual customers mostly rent bikes in the mid-day time. For registered customers, the peak rental time is in the morning and evening, or, in other words, at the start and end of work.

by workingday

```r
par(mfrow = c(1,2))

barplot(tapply(bike$casual, bike$workingday, sum), beside = TRUE,
        main = "Casual",
        xlab = "Workingday | 0 = No",
        ylab = "Count of Rentals")
barplot(tapply(bike$registered, bike$workingday, sum), beside = TRUE,
        main = "Registered",
        xlab = "1 = Yes | Workingday ",
        ylab = "Count of Rentals")
```

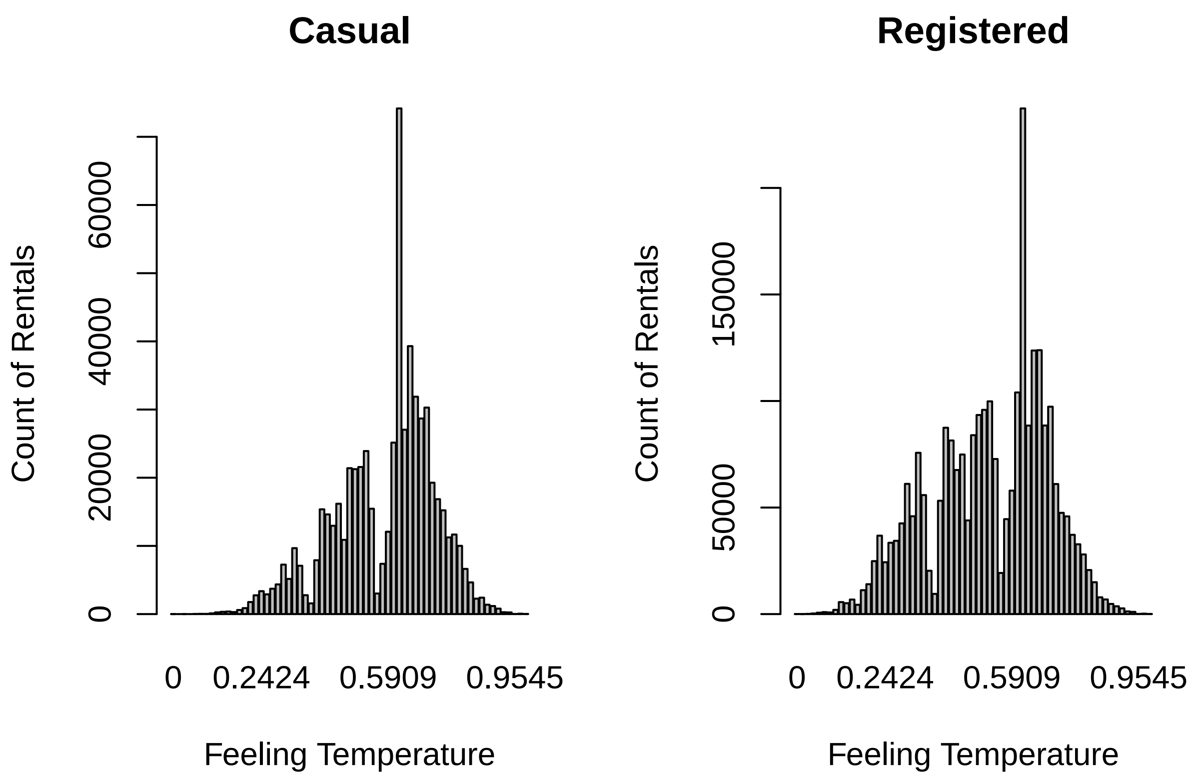


After this visual we can assume that registered customers are local working people who use bike rentals to commute to work. Casual customers can be tourists , as their rental behavior is indifferent between working days and holidays. But these are just assumptions.

by weathersit

```
par(mfrow = c(1,2))

barplot(tapply(bike$casual, bike$atemp, sum), beside = TRUE,
        main = "Casual",
        xlab = "Feeling Temperature",
        ylab = "Count of Rentals")
barplot(tapply(bike$registered, bike$atemp, sum), beside = TRUE,
        main = "Registered",
        xlab = "Feeling Temperature",
        ylab = "Count of Rentals")
```



Registered customers rent bikes on colder days also. Casulas not so often.

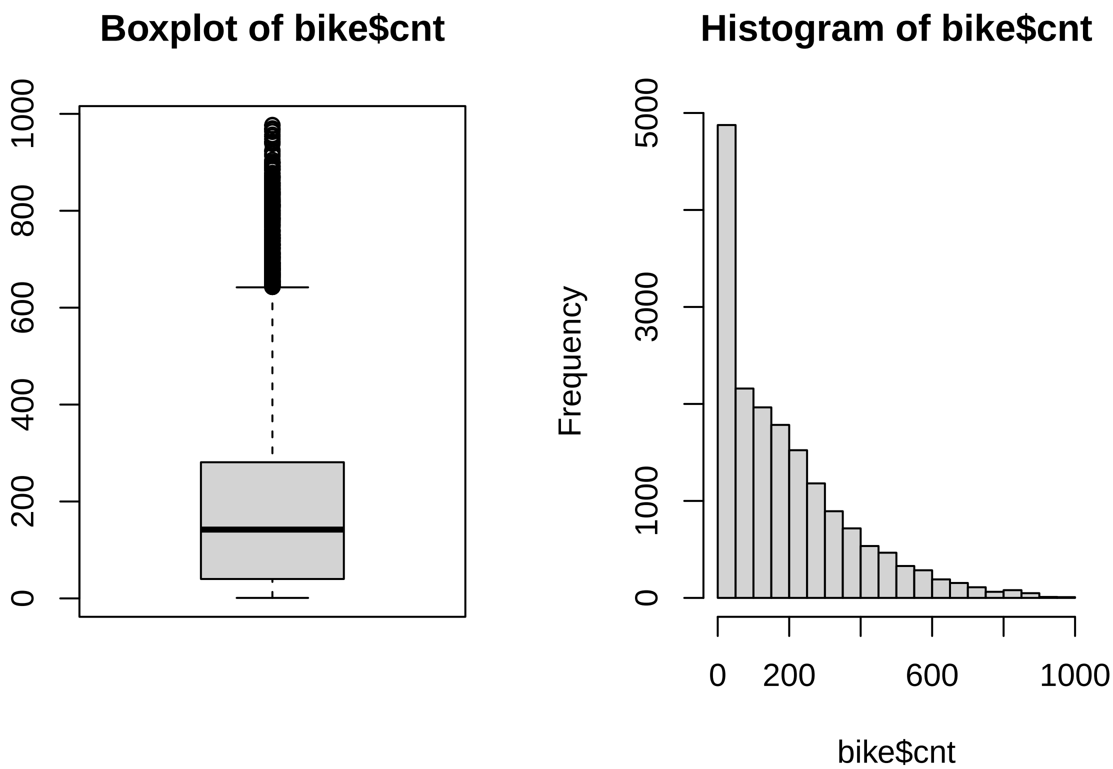
Total count (cnt)

cnt = registered + casual

Overall rental behavior across different weather situations and daytimes

Outliers cnt

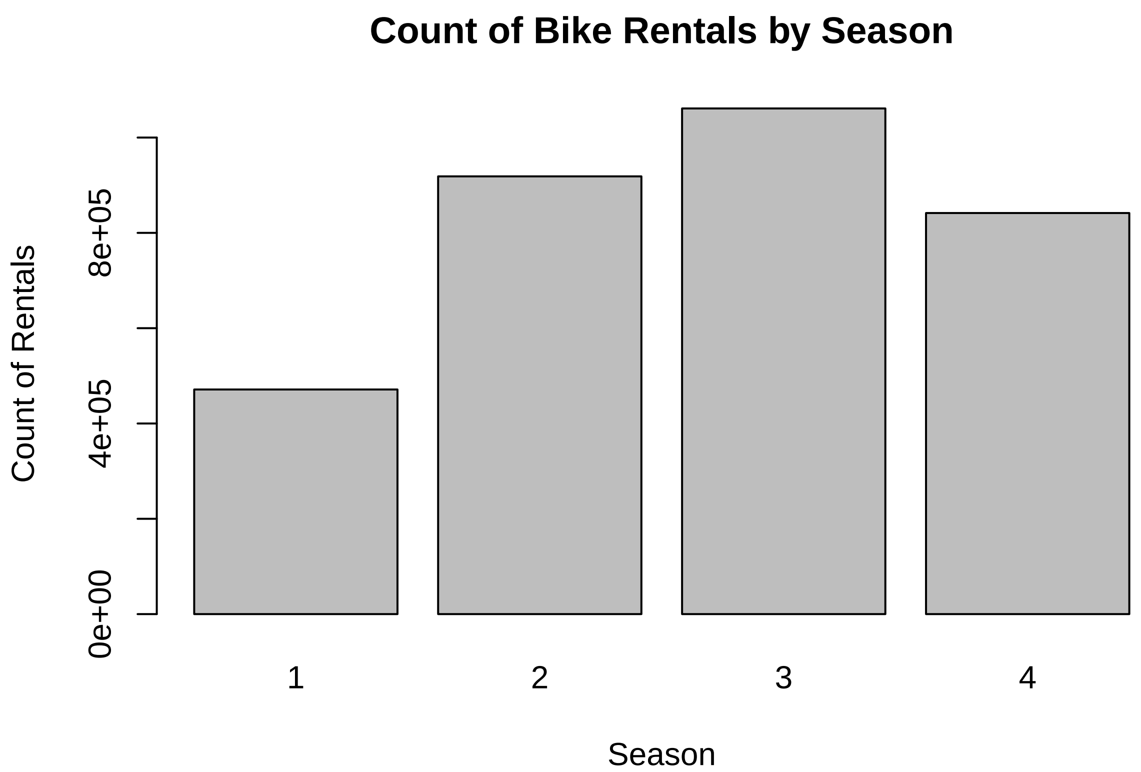
```
par(mfrow=c(1,2))
boxplot(bike$cnt, main='Boxplot of bike$cnt')
hist(bike$cnt,)
```



TOTAL COUNT OF RENTALS HAS STRONG RIGHT-SKEWED DISTRIBUTION, THEREFORE IT HAS SOME OUTLIERS BEGINNING FROM APPROX. 650 RENTALS. THIS MAY BE EXPLAINED BY NON LINEAR DISTRIBUTION OF THE INDEPENDENT VARIABLES. EG. A LOT MORE TOURISTS IN SUMMER THEREFORE, BIG INCREASE IN BIKE RENTALS IN SUMMER FOR SHORT PERIOD

by season

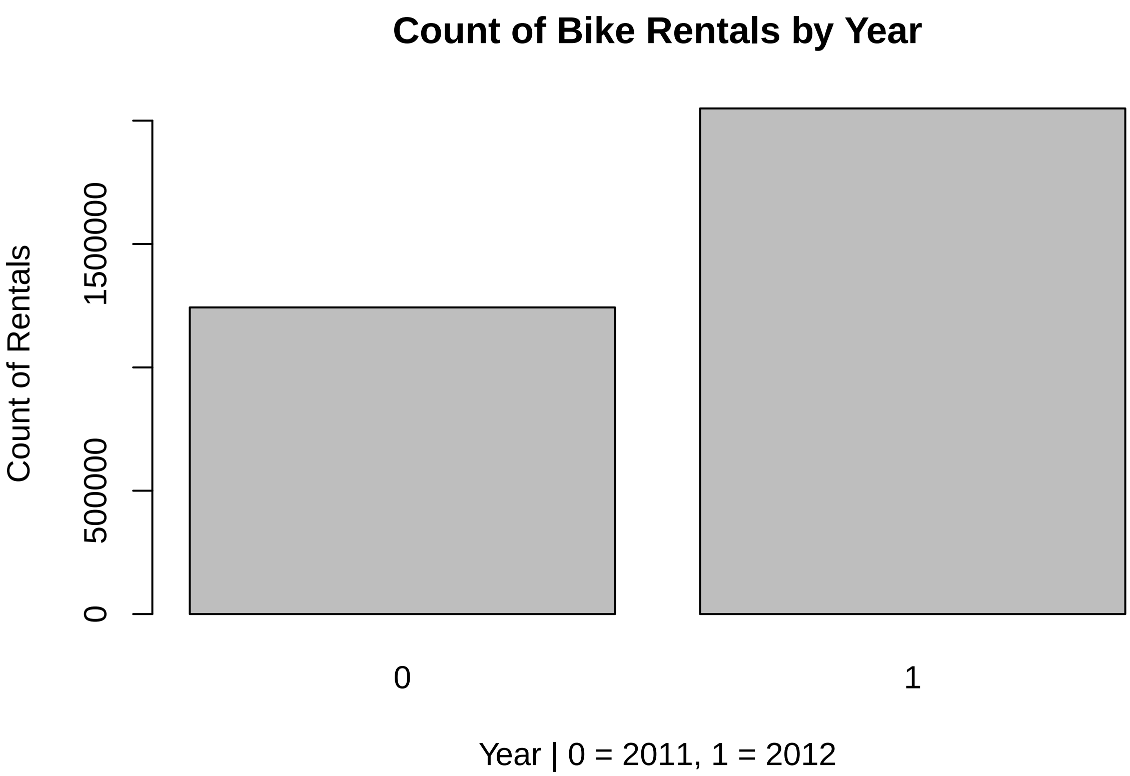
```
barplot(tapply(bike$cnt, bike$season, sum), beside = TRUE,
        main = "Count of Bike Rentals by Season",
        xlab = "Season",
        ylab = "Count of Rentals")
```



The are much more bike rentals in summer than in winter. And it's actually make sense, because most people would prefer to ride in warm summer times rather than in cold winter times.

by year

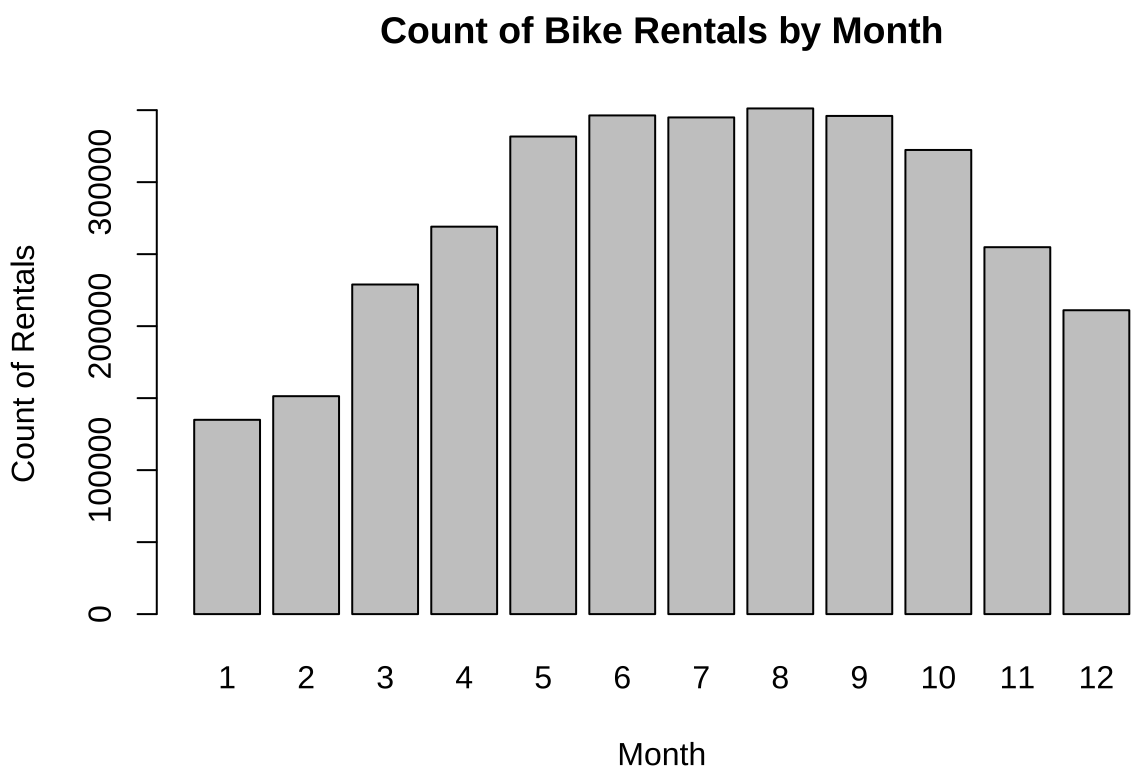
```
barplot(tapply(bike$cnt, bike$yr, sum), beside = TRUE,
        main = "Count of Bike Rentals by Year",
        xlab = "Year | 0 = 2011, 1 = 2012",
        ylab = "Count of Rentals")
```



This visual can show that our bike rental company has gained popularity between years 2011 and 2012. There are more bike rentals in 2012.

by month

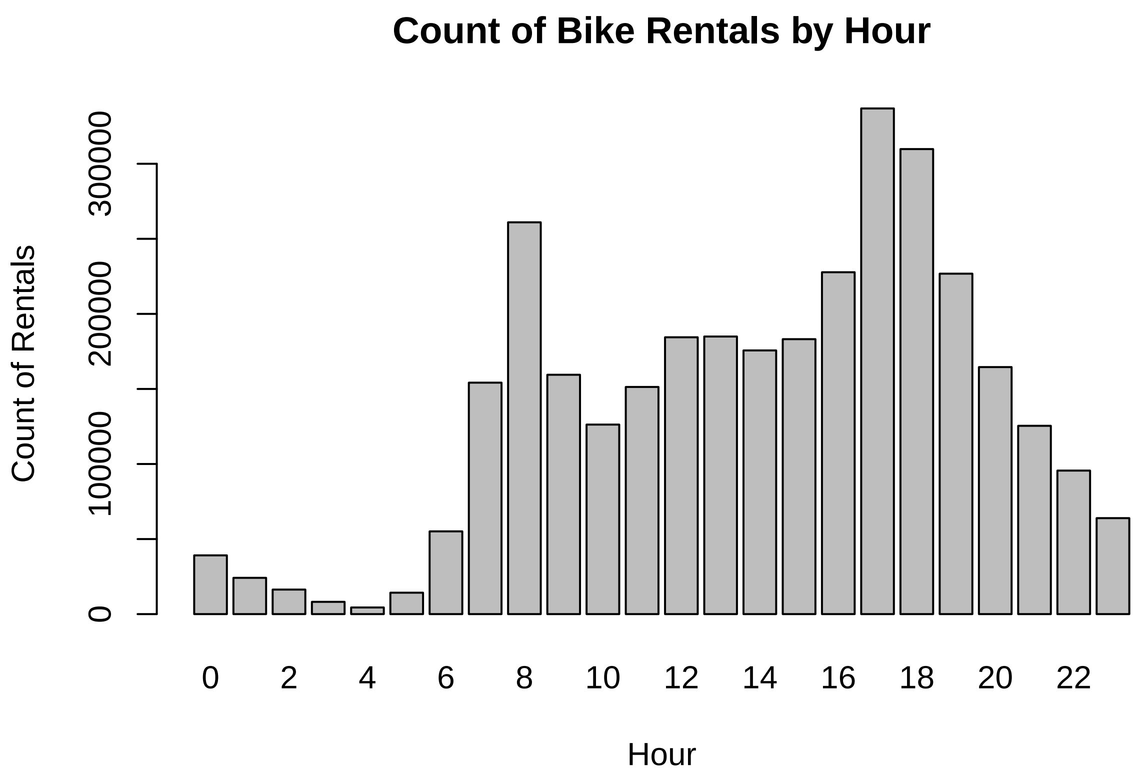
```
barplot(tapply(bike$cnt, bike$mnth, sum), beside = TRUE,
        main = "Count of Bike Rentals by Month",
        xlab = "Month",
        ylab = "Count of Rentals")
```



This visual explains the correlation between season and month. It also correlates with season barplot. There are much more bike rentals in summer times. (from 5th to 9-10th month)

by hour

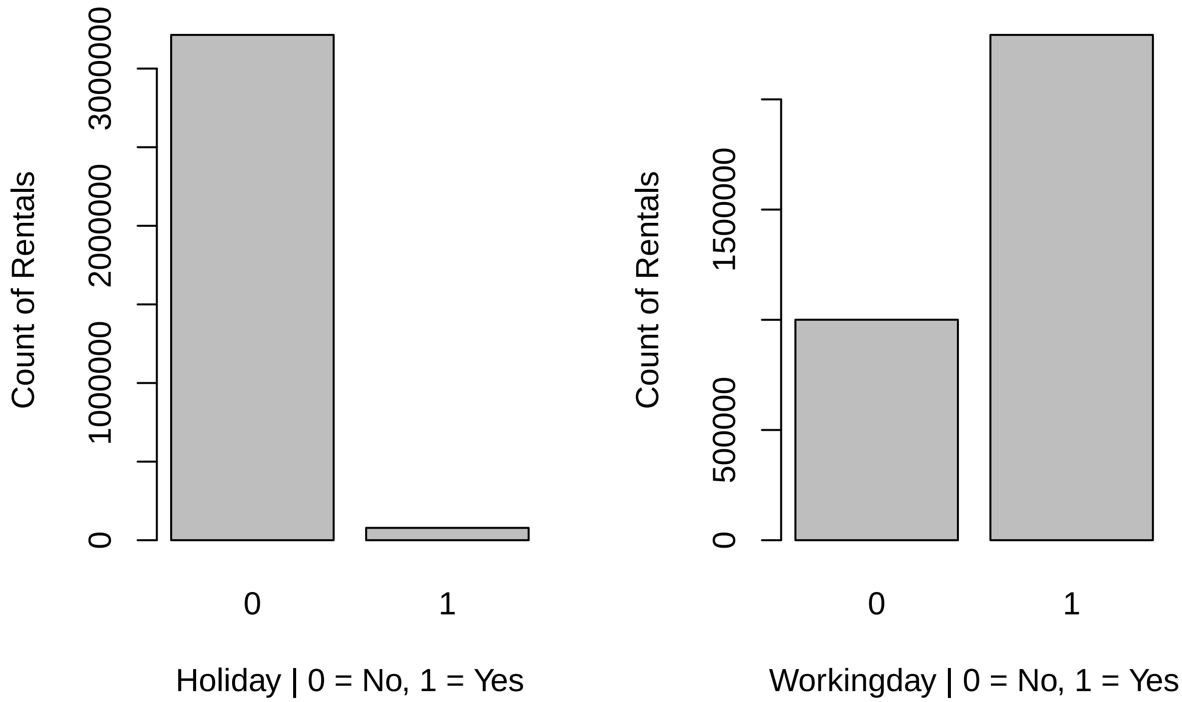
```
barplot(tapply(bike$cnt, bike$hr, sum), beside = TRUE,
        main = "Count of Bike Rentals by Hour",
        xlab = "Hour",
        ylab = "Count of Rentals")
```



The bike rental peaks are mainly during the morning (8 am.) and evening (5 pm. - 7pm.; 17:00-19:00) hours.

by holiday and workingday

```
par(mfrow = c(1,2))
barplot(tapply(bike$cnt, bike$holiday, sum), beside = TRUE,
        xlab = "Holiday | 0 = No, 1 = Yes",
        ylab = "Count of Rentals")
barplot(tapply(bike$cnt, bike$workingday, sum), beside = TRUE,
        xlab = "Workingday | 0 = No, 1 = Yes",
        ylab = "Count of Rentals")
```



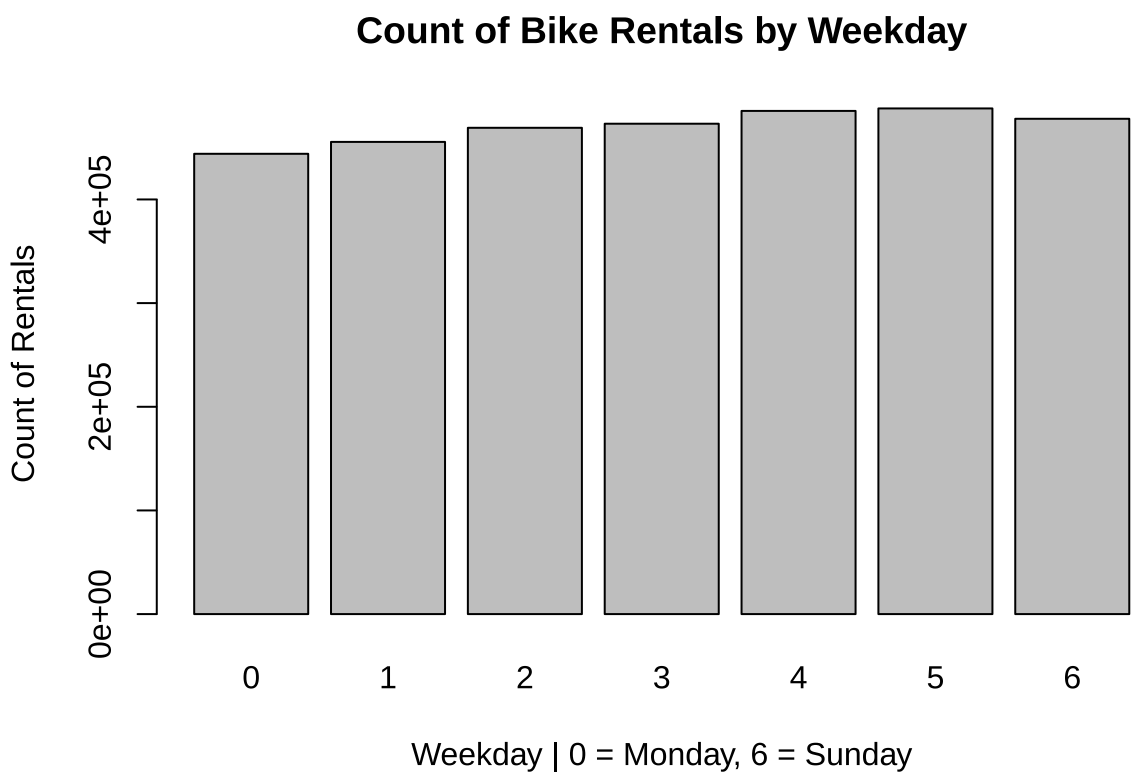
There are much less holidays than casual days. Therefore, not evenly distributed.

There are more bike rentals on working days than on holidays or weekends.

Combined with the hours and visual representation of the working day, it can be assumed that the majority of bike rental revenue is generated by working people who rent our bikes to commute to their place of work and return home during the evening hours

by weekday

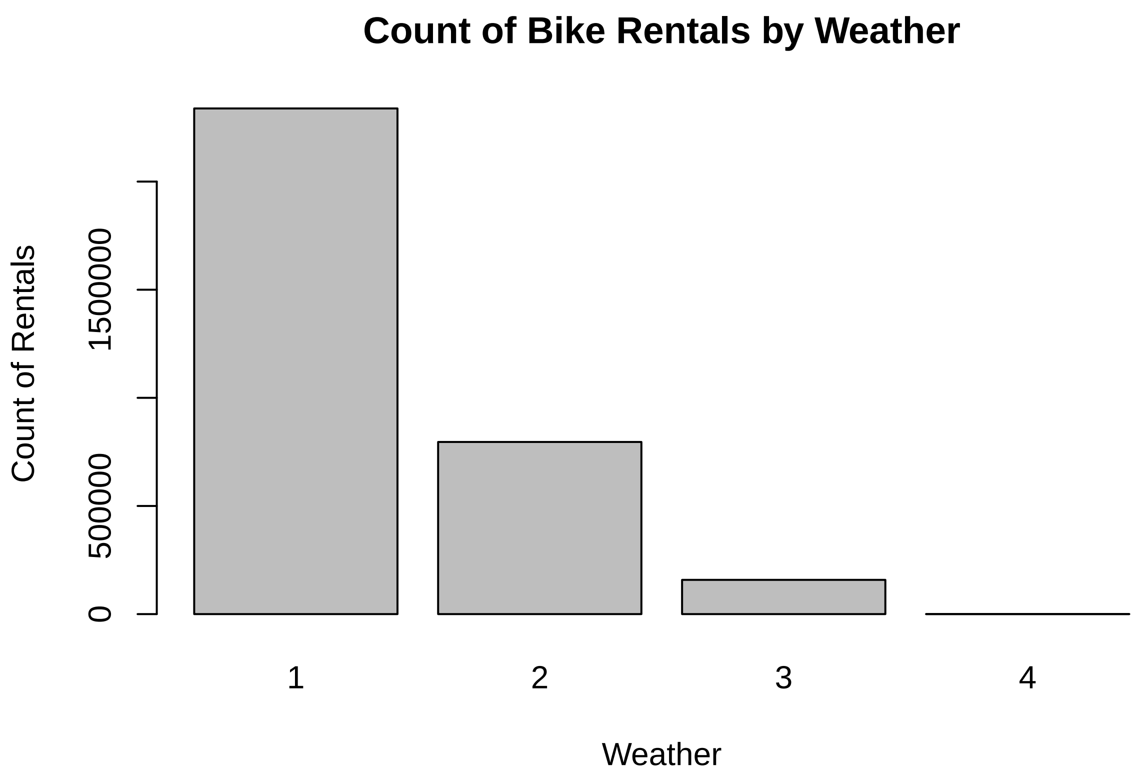
```r
barplot(tapply(bike$cnt, bike$weekday, sum), beside = TRUE,
        main = "Count of Bike Rentals by Weekday",
        xlab = "Weekday | 0 = Monday, 6 = Sunday",
        ylab = "Count of Rentals")
```



No clear information between weekdays

by weather

```
barplot(tapply(bike$cnt, bike$weathersit, sum), beside = TRUE,
        main = "Count of Bike Rentals by Weather",
        xlab = "Weather",
        ylab = "Count of Rentals")
```




```
- 1: Clear, Few clouds, Partly cloudy, Partly cloudy
- 2: Mist + Cloudy, Mist + Broken clouds, Mist + Few clouds, Mist
- 3: Light Snow, Light Rain + Thunderstorm + Scattered clouds, Light Rain + Scattered clouds
- 4: Heavy Rain + Ice Pallets + Thunderstorm + Mist, Snow + Fog
```

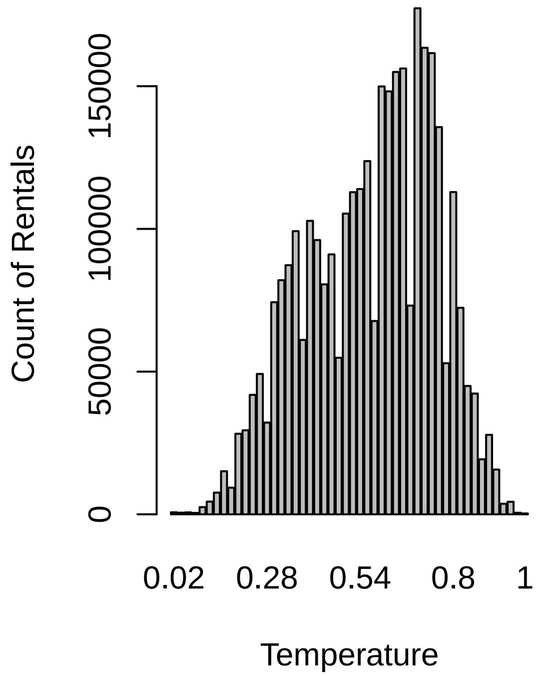
Most rentals on weather category 1, which is also typical for summer times.

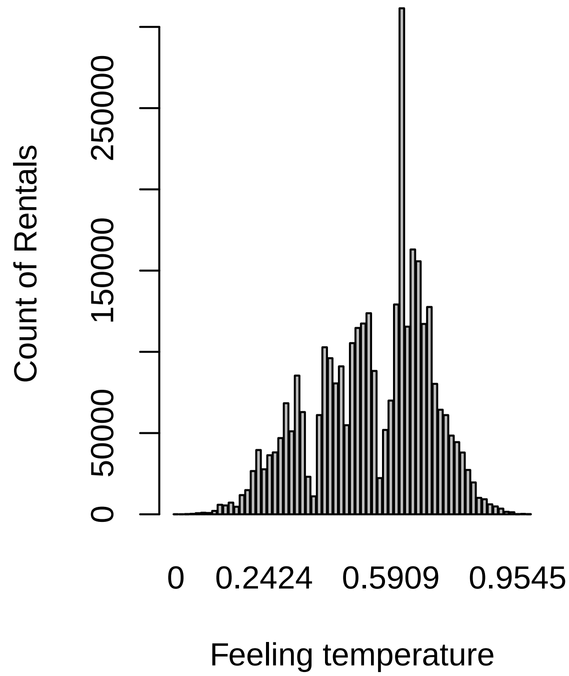
by temperature and feeling temperature (atemp)

```
par(mfrow = c(1,2))

barplot(tapply(bike$cnt, bike$temp, sum), beside = TRUE,
        xlab = "Temperature",
        ylab = "Count of Rentals")

barplot(tapply(bike$cnt, bike$atemp, sum), beside = TRUE,
        xlab = "Feeling temperature",
        ylab = "Count of Rentals")
```





Temperature in C

Normalized temperature by the following formula: (t-t_min)/(t_max-t_min) t_min = -8 t_max = +39

Feeling Temperature in C

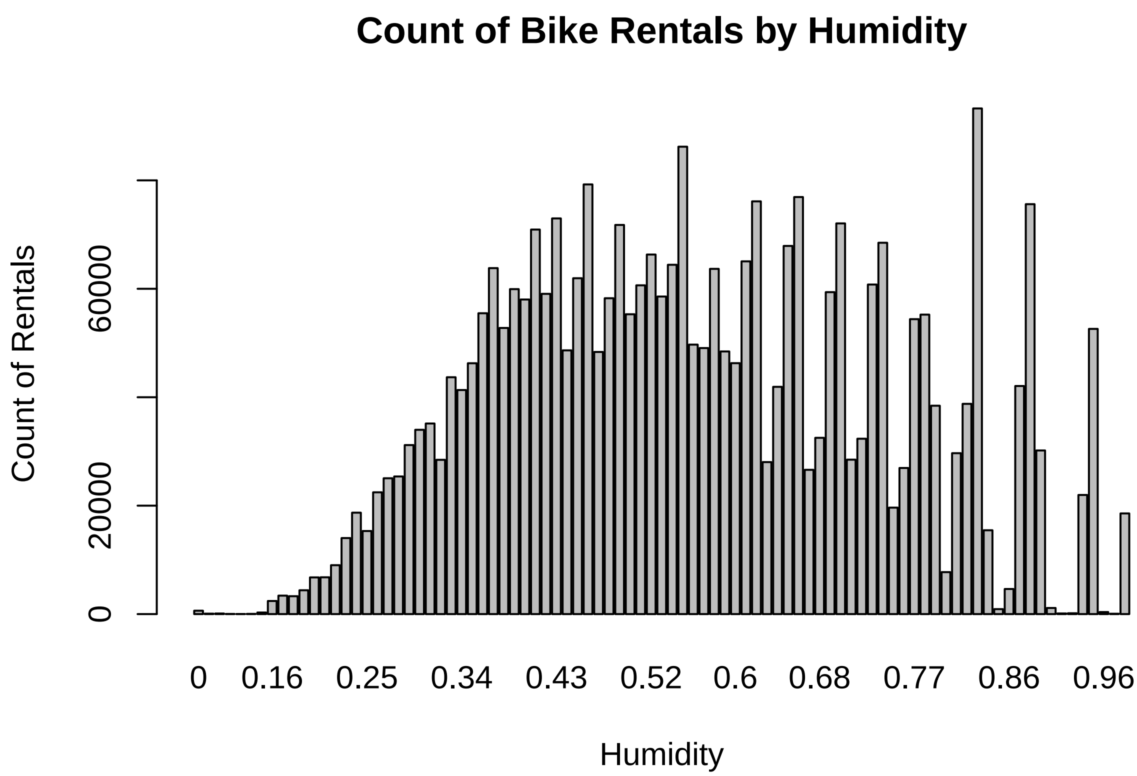
Normalized temperature by the following formula: (t-t_min)/(t_max-t_min) t_min = -16 t_max = +50

The distribution is left-skewed. That means that people tend to rent our bikes on more warm days, but not on burning hot days!

by humidity

Normalized humididty. The values are divided to 100 (max)

```
barplot(tapply(bike$cnt, bike$hum, sum), beside = TRUE,
        main = "Count of Bike Rentals by Humidity",
        xlab = "Humidity",
        ylab = "Count of Rentals")
```

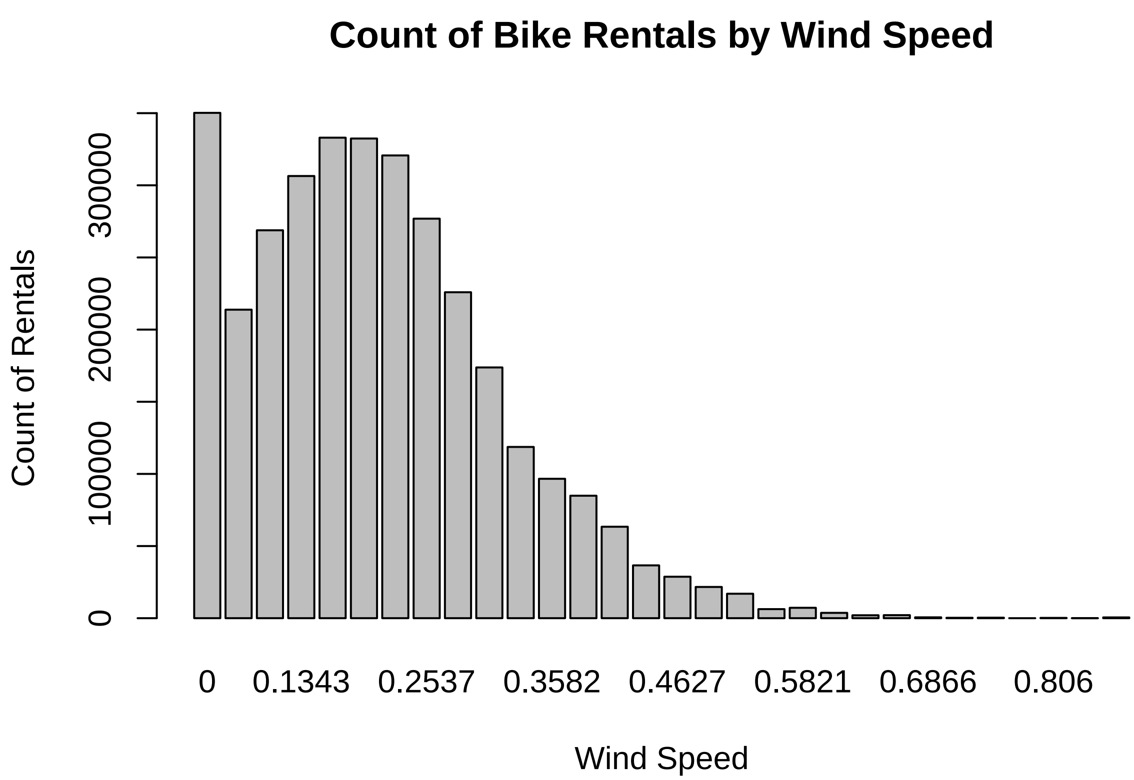


Most rentals in humidity range of 0.30 to 0.70

by wind speed

Normalized wind speed. The values are divided to 67 (max)

```
barplot(tapply(bike$cnt, bike$windspeed, sum), beside = TRUE,
        main = "Count of Bike Rentals by Wind Speed",
        xlab = "Wind Speed",
        ylab = "Count of Rentals")
```



This visual explains the negative correlation between wind speed and cnt. The lower wind speed the better for bike rental.

Data pre-processing before training

Fixing categorical variables

- season : season (1:winter, 2:spring, 3:summer, 4:fall)
  - mnth : month (1 to 12)
  - hr : hour (0 to 23)
  - holiday : weather day is holiday or not (extracted from http://dchr.dc.gov/page/holiday-schedule)
  - weekday : day of the week
  - workingday : if day is neither weekend nor holiday is 1, otherwise is 0.
  - weathersit 1,2,3,4.

```
# Convert 'season' to a factor
bike$season <- factor(bike$season, levels = c(1, 2, 3, 4), labels = c("winter", "spring", "summer", "fa

# Convert 'mnth' to a factor
bike$mnth <- factor(bike$mnth)

# Convert 'hr' to a factor
bike$hr <- factor(bike$hr)

# Convert 'holiday' to a factor
bike$holiday <- factor(bike$holiday, levels = c(0, 1), labels = c("not_holiday", "holiday"))

# Convert 'weekday' to a factor
bike$weekday <- factor(bike$weekday, levels = c(0, 1, 2, 3, 4, 5, 6), labels = c("Sunday", "Monday", "T
```

```r
# Convert 'workingday' to a factor
bike$workingday <- factor(bike$workingday, levels = c(0, 1), labels = c("weekend/holiday", "working_day

# Convert 'weathersit' to a factor
bike$weathersit <- factor(bike$weathersit)
```

```r
str(bike)
```

```
## 'data.frame':    17379 obs. of  15 variables:
##  $ season    : Factor w/ 4 levels "winter","spring",..: 1 1 1 1 1 1 1 1 1 1 ...
##  $ yr        : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ mnth      : Factor w/ 12 levels "1","2","3","4",..: 1 1 1 1 1 1 1 1 1 1 ...
##  $ hr        : Factor w/ 24 levels "0","1","2","3",..: 1 2 3 4 5 6 7 8 9 10 ...
##  $ holiday   : Factor w/ 2 levels "not_holiday",..: 1 1 1 1 1 1 1 1 1 1 ...
##  $ weekday   : Factor w/ 7 levels "Sunday","Monday",..: 7 7 7 7 7 7 7 7 7 7 ...
##  $ workingday: Factor w/ 2 levels "weekend/holiday",..: 1 1 1 1 1 1 1 1 1 1 ...
##  $ weathersit: Factor w/ 4 levels "1","2","3","4": 1 1 1 1 1 2 1 1 1 1 ...
##  $ temp      : num  0.24 0.22 0.22 0.24 0.24 0.24 0.22 0.2 0.24 0.32 ...
##  $ atemp     : num  0.288 0.273 0.273 0.288 0.288 ...
##  $ hum       : num  0.81 0.8 0.8 0.75 0.75 0.75 0.8 0.86 0.75 0.76 ...
##  $ windspeed : num  0 0 0 0 0 0.0896 0 0 0 0 ...
##  $ casual    : int  3 8 5 3 0 0 2 1 1 8 ...
##  $ registered: int  13 32 27 10 1 1 0 2 7 6 ...
##  $ cnt       : int  16 40 32 13 1 1 2 3 8 14 ...
```

Deleting casual and registered varaibles

```r
bike$casual <- NULL
bike$registered <- NULL
```

We are going to train our model on total count of rental. Therefore, we delete the registered and casual variables.

Data split

Split ratio:

- 80% train data
- 20% test data

```r
set.seed(123)
sample_data <- sample(x = c(1,2), size = nrow(bike),replace = T, prob = c(0.8,0.2))
train_data <- bike[sample_data == 1,]
test_data <- bike[sample_data == 2,]
```

Models

Importing libraries

```r
library(caret)
```

```
## Loading required package: ggplot2

## Loading required package: lattice
```

```r
library(randomForest)
```

```
## randomForest 4.7-1.1

## Type rfNews() to see new features/changes/bug fixes.

##
## Attaching package: 'randomForest'

## The following object is masked from 'package:ggplot2':
##
##     margin
```

```r
library(rpart)
library(rpart.plot)
```

```r
str(bike)
```

```
## 'data.frame':    17379 obs. of  13 variables:
##  $ season    : Factor w/ 4 levels "winter","spring",..: 1 1 1 1 1 1 1 1 1 1 ...
##  $ yr        : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ mnth      : Factor w/ 12 levels "1","2","3","4",..: 1 1 1 1 1 1 1 1 1 1 ...
##  $ hr        : Factor w/ 24 levels "0","1","2","3",..: 1 2 3 4 5 6 7 8 9 10 ...
##  $ holiday   : Factor w/ 2 levels "not_holiday",..: 1 1 1 1 1 1 1 1 1 1 ...
##  $ weekday   : Factor w/ 7 levels "Sunday","Monday",..: 7 7 7 7 7 7 7 7 7 7 ...
##  $ workingday: Factor w/ 2 levels "weekend/holiday",..: 1 1 1 1 1 1 1 1 1 1 ...
##  $ weathersit: Factor w/ 4 levels "1","2","3","4": 1 1 1 1 1 2 1 1 1 1 ...
##  $ temp      : num  0.24 0.22 0.22 0.24 0.24 0.24 0.22 0.2 0.24 0.32 ...
##  $ atemp     : num  0.288 0.273 0.273 0.288 0.288 ...
##  $ hum       : num  0.81 0.8 0.8 0.75 0.75 0.75 0.8 0.86 0.75 0.76 ...
##  $ windspeed : num  0 0 0 0 0 0.0896 0 0 0 0 ...
##  $ cnt       : int  16 40 32 13 1 1 2 3 8 14 ...
```

Linear Regression

```r
full_lm <- lm(cnt ~ ., data = train_data)
```

```r
stepwise_lm <- step(full_lm, direction = "backward")
```

```
## Start:  AIC=128866.9
## cnt ~ season + yr + mnth + hr + holiday + weekday + workingday +
##     weathersit + temp + atemp + hum + windspeed
##
##
## Step:  AIC=128866.9
## cnt ~ season + yr + mnth + hr + holiday + weekday + weathersit +
##     temp + atemp + hum + windspeed
##
##              Df Sum of Sq       RSS    AIC
## <none>                     144412490 128867
## - atemp       1    132955 144545446 128878
## - windspeed   1    161991 144574482 128881
## - temp        1    163337 144575827 128881
## - holiday     1    231363 144643853 128887
## - weekday     6    386447 144798938 128892
## - season      3   1827611 146240102 129036
```

```
## - hum          1   1857705 146270195 129043
## - mnth        11   2110182 146522673 129047
## - weathersit  3   3664127 148076618 129210
## - yr          1  25675478 170087968 131144
## - hr         23 159711571 304124061 139192
```

```r
summary(stepwise_lm)
```

```
##
## Call:
## lm(formula = cnt ~ season + yr + mnth + hr + holiday + weekday +
##     weathersit + temp + atemp + hum + windspeed, data = train_data)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -390.33  -60.79   -7.48   51.36  432.29
##
## Coefficients:
##                   Estimate Std. Error t value Pr(>|t|)
## (Intercept)        -84.620      7.434 -11.382  < 2e-16 ***
## seasonspring        38.544      5.405   7.131 1.05e-12 ***
## seasonsummer        33.776      6.409   5.270 1.38e-07 ***
## seasonfall          66.179      5.448  12.147  < 2e-16 ***
## yr                  86.988      1.752  49.662  < 2e-16 ***
## mnth2                5.104      4.376   1.166 0.243516
## mnth3               12.314      4.940   2.493 0.012692 *
## mnth4                5.989      7.325   0.818 0.413573
## mnth5               20.783      7.858   2.645 0.008179 **
## mnth6                2.433      8.087   0.301 0.763585
## mnth7              -15.145      9.046  -1.674 0.094123 .
## mnth8                3.883      8.830   0.440 0.660121
## mnth9               32.412      7.820   4.145 3.42e-05 ***
## mnth10              18.663      7.248   2.575 0.010034 *
## mnth11              -8.307      6.962  -1.193 0.232844
## mnth12              -6.609      5.516  -1.198 0.230863
## hr1                -16.408      6.038  -2.718 0.006585 **
## hr2                -26.771      6.004  -4.459 8.30e-06 ***
## hr3                -36.497      6.087  -5.995 2.08e-09 ***
## hr4                -38.165      6.058  -6.300 3.06e-10 ***
## hr5                -21.084      6.033  -3.495 0.000476 ***
## hr6                 36.014      5.999   6.003 1.98e-09 ***
## hr7                173.485      5.986  28.982  < 2e-16 ***
## hr8                314.962      5.977  52.695  < 2e-16 ***
## hr9                163.950      5.989  27.376  < 2e-16 ***
## hr10               111.421      6.012  18.532  < 2e-16 ***
## hr11               134.008      6.063  22.104  < 2e-16 ***
## hr12               174.785      6.093  28.685  < 2e-16 ***
## hr13               168.361      6.149  27.381  < 2e-16 ***
## hr14               154.362      6.156  25.074  < 2e-16 ***
## hr15               165.993      6.214  26.712  < 2e-16 ***
## hr16               223.673      6.204  36.053  < 2e-16 ***
## hr17               382.460      6.159  62.101  < 2e-16 ***
## hr18               350.081      6.093  57.452  < 2e-16 ***
## hr19               234.078      6.077  38.521  < 2e-16 ***
## hr20               155.906      6.021  25.893  < 2e-16 ***
```

```
## hr21              110.082      6.014  18.303  < 2e-16 ***
## hr22               72.494      5.972  12.138  < 2e-16 ***
## hr23               33.464      6.022   5.557 2.80e-08 ***
## holidayholiday    -25.569      5.424  -4.714 2.45e-06 ***
## weekdayMonday       8.173      3.332   2.453 0.014176 *
## weekdayTuesday      9.793      3.258   3.006 0.002655 **
## weekdayWednesday   13.972      3.250   4.300 1.72e-05 ***
## weekdayThursday    12.152      3.260   3.728 0.000194 ***
## weekdayFriday      16.100      3.258   4.941 7.87e-07 ***
## weekdaySaturday    16.415      3.229   5.084 3.75e-07 ***
## weathersit2       -10.217      2.153  -4.745 2.11e-06 ***
## weathersit3       -67.572      3.608 -18.729  < 2e-16 ***
## weathersit4       -63.766     59.101  -1.079 0.280635
## temp              126.901     32.037   3.961 7.50e-05 ***
## atemp             118.238     33.086   3.574 0.000353 ***
## hum               -83.049      6.217 -13.358  < 2e-16 ***
## windspeed         -31.058      7.873  -3.945 8.03e-05 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 102 on 13872 degrees of freedom
## Multiple R-squared:  0.6871, Adjusted R-squared:  0.6859
## F-statistic: 585.8 on 52 and 13872 DF,  p-value: < 2.2e-16
```

Direction = backward

Majority of variables are significant predictors.

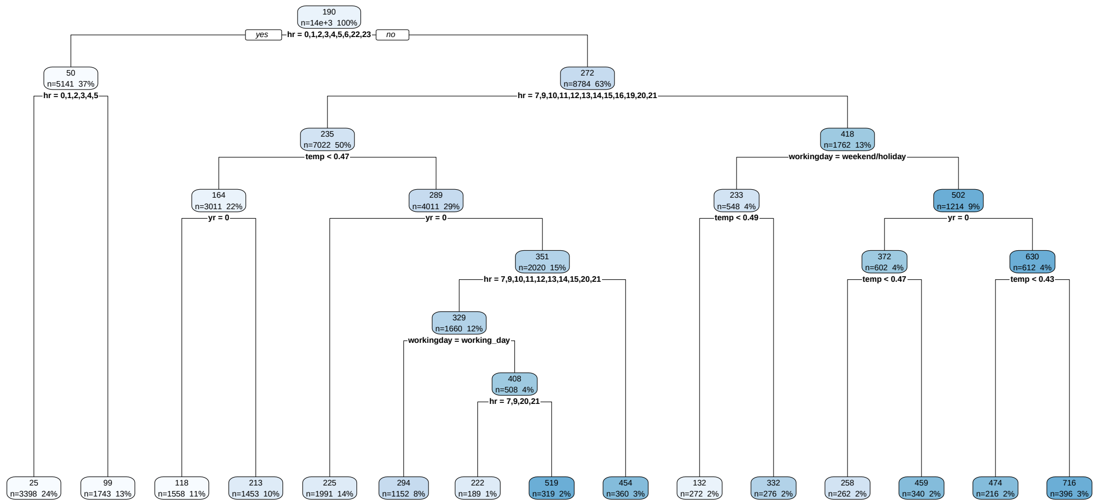
Interpretation of coefficients:

- **Season**:
  - seasonspring: The coefficient is 38.544, suggesting that, all else being equal, being in the spring season is associated with an increase of 38.544 units in the predicted count of bike rentals compared to the reference season.
  - seasonsummer: The coefficient is 33.776, indicating that, all else being equal, being in the summer season is associated with an increase of 33.776 units in the predicted count.
  - seasonfall: The coefficient is 66.179, implying that, all else being equal, being in the fall season is associated with an increase of 66.179 units in the predicted count.
- **Month**:
  - The coefficients for different months (mnth2 to mnth12) indicate the change in predicted bike rentals compared to the reference month (presumably mnth1). For example, mnth12 has a coefficient of -6.609, suggesting a decrease of 6.609 units in predicted bike rentals in December compared to the reference month.
- **Hour**:
  - The coefficients for different hours (hr1 to hr23) represent the change in predicted bike rentals for each hour compared to the reference hour. For example, hr7 has a coefficient of 173.485, indicating a substantial increase in predicted bike rentals at 7 a.m. compared to the reference hour.

Regression Tree

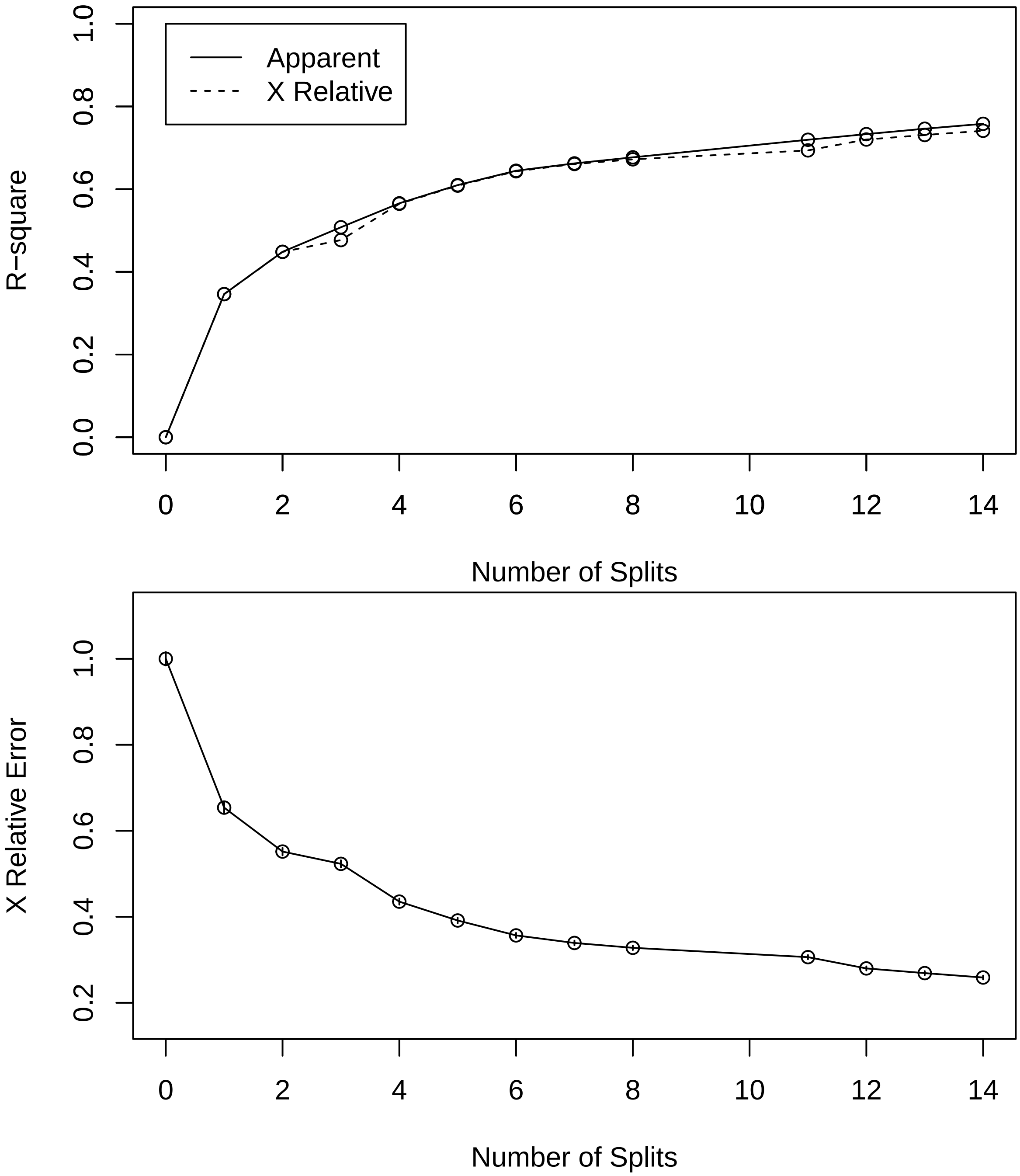
```r
full_rt <- rpart(cnt ~ ., data =train_data, control = list(c = 0))
```

```r
rpart.plot(full_rt, extra = 101,)
```



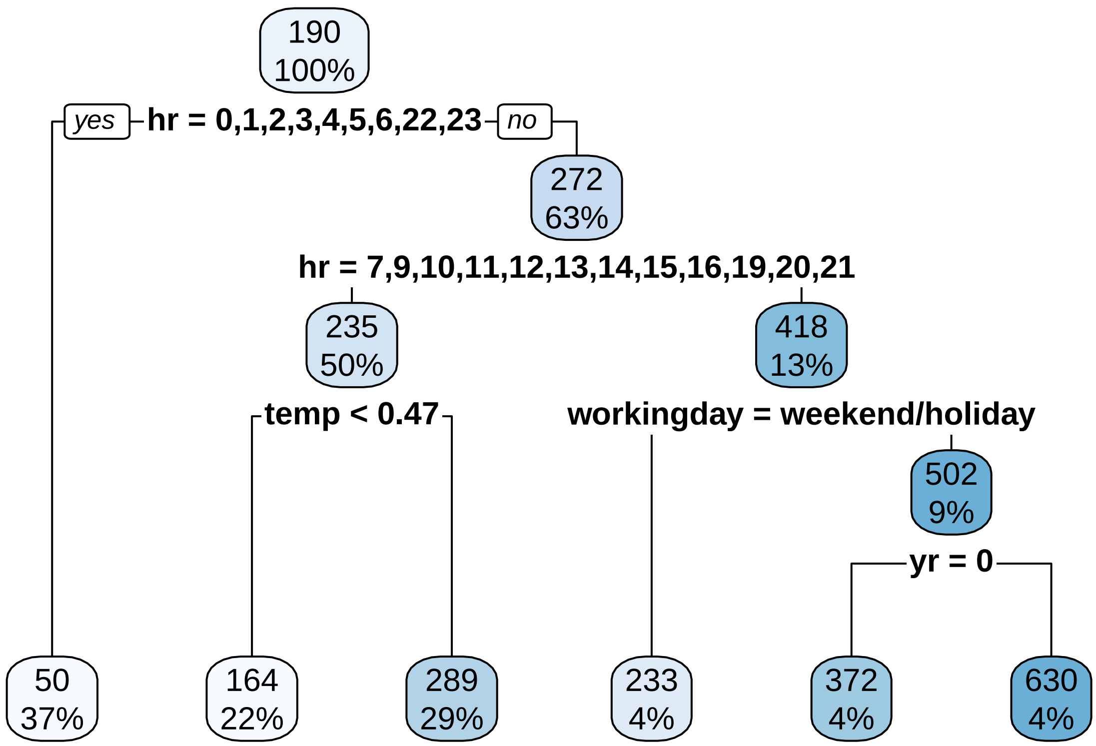

```r
rsq.rpart(full_rt)
```

```
##
## Regression tree:
## rpart(formula = cnt ~ ., data = train_data, control = list(c = 0))
##
## Variables actually used in tree construction:
## [1] hr         temp       workingday yr
##
## Root node error: 461538329/13925 = 33145
##
## n= 13925
##
##          CP nsplit rel error  xerror      xstd
## 1  0.346152      0   1.00000 1.00005 0.0157659
## 2  0.102318      1   0.65385 0.65392 0.0117532
## 3  0.059381      2   0.55153 0.55169 0.0088805
## 4  0.057892      3   0.49215 0.52315 0.0084669
## 5  0.044033      4   0.43426 0.43531 0.0071103
## 6  0.034837      5   0.39022 0.39143 0.0064319
## 7  0.017666      6   0.35539 0.35676 0.0058293
## 8  0.014681      7   0.33772 0.33916 0.0053596
## 9  0.014172      8   0.32304 0.32790 0.0052566
## 10 0.013649     11   0.28052 0.30623 0.0049753
## 11 0.013020     12   0.26688 0.27996 0.0047591
## 12 0.011923     13   0.25386 0.26905 0.0047101
## 13 0.010000     14   0.24193 0.25880 0.0045073
```




```
pruned_rt <- prune(full_rt, cp = 0.044033)
```

```
rpart.plot(pruned_rt)
```



The number of splits that is in our opinion optimal in order to minimize X Relative Error is **4 with cp = 0.044033**

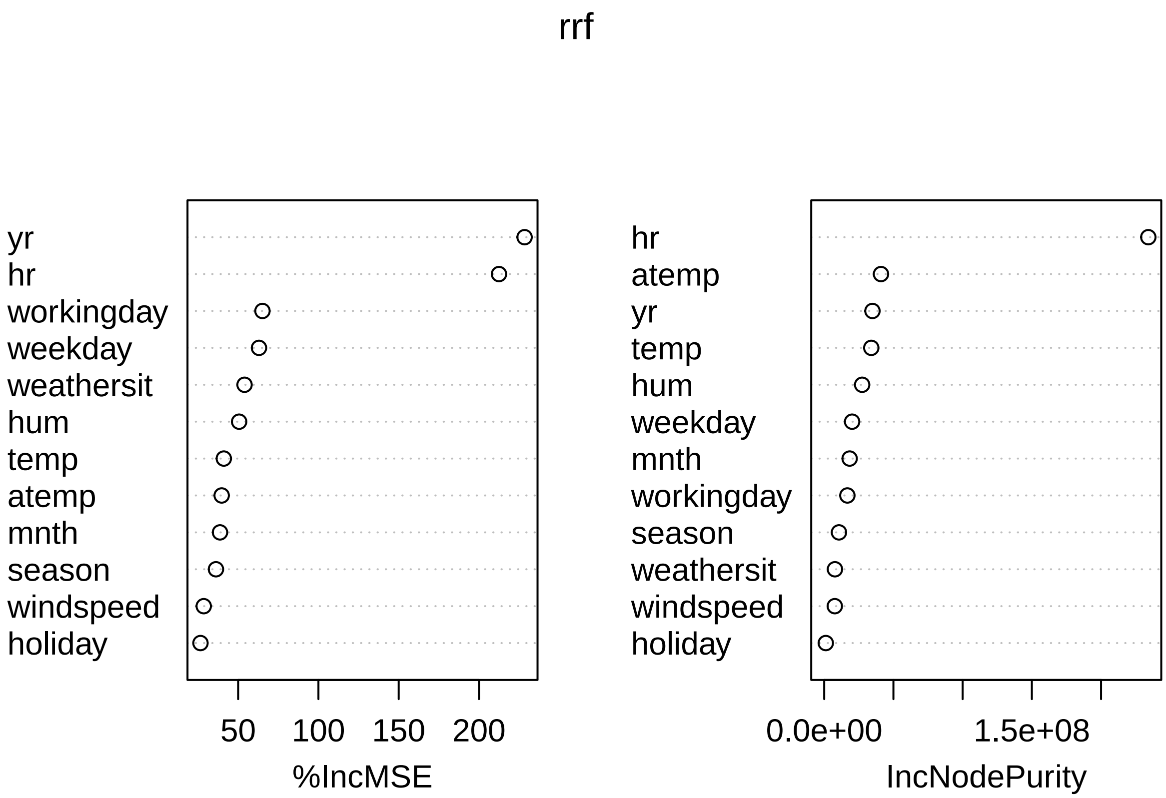
Random forest

```
rrf <- randomForest(cnt ~.,
                    data = train_data,
                    mtry = 4,
                    importance = TRUE
                    )
```

```
rrf
```

```
##
## Call:
##  randomForest(formula = cnt ~ ., data = train_data, mtry = 4,      importance = TRUE)
##                Type of random forest: regression
##                      Number of trees: 500
## No. of variables tried at each split: 4
##
##           Mean of squared residuals: 2336.046
##                     % Var explained: 92.95
```

```
varImpPlot(rrf)
```




```r
importance(rrf)
```

```
##              %IncMSE IncNodePurity
## season      36.17111      10660647
## yr         228.41064      34848546
## mnth        38.68979      18381643
## hr         212.47504     234124681
## holiday     26.53114       1139391
## weekday     62.97866      20153054
## workingday  65.13580      16736194
## weathersit  53.97868       7753768
## temp        41.06882      34006466
## atemp       39.73952      41017747
## hum         50.61156      27408835
## windspeed   28.56683       7647619
```

mtry = 4

In decision tree algorithms, "mtry" represents the number of randomly selected features considered at each node when building a tree. Setting mtry to 4 means that, at each node, the algorithm selects 4 features from the dataset and evaluates them for the best split. This randomness helps prevent overfitting and enhances the model's ability to generalize to new data.

- **Hour (hr)**:
  - %IncMSE: 212.47504
  - IncNodePurity: 234124681
  - The high %IncMSE value for the 'hour' variable suggests that the hour of the day is a crucial predictor in the model. An increase in the hour leads to a substantial increase in Mean Squared Error, indicating that this variable contributes significantly to the model's predictive power. The corresponding IncNodePurity value reinforces this, indicating that splits based on the hour in the

decision tree contribute to increased node purity.

- **Year (yr)**:
  - %IncMSE: 228.41064
  - IncNodePurity: 34848546
  - The 'year' variable also has a high %IncMSE value, indicating that it is an important predictor. An increase in the year contributes significantly to the model's predictive performance. The IncNodePurity value suggests that splits based on the year contribute to improved node purity in the decision tree.
- **Temperature (temp)**:
  - %IncMSE: 41.06882
  - IncNodePurity: 34006466
  - The 'temp' variable has a moderate %IncMSE value, suggesting that it is an important predictor for the model. An increase in temperature contributes to an increase in Mean Squared Error, indicating its relevance in predicting the target variable. The IncNodePurity value also supports the importance of 'temp' in decision tree splits.

Model Evaluations

Predictions

```
predlm <- predict(stepwise_lm, newdata = test_data)
predrt <- predict(pruned_rt, newdata = test_data)
predrrf <- predict(rrf, newdata = test_data)
```

Results

```
resultslm <- postResample(predlm, test_data$cnt)
resultsrt <- postResample(predrt, test_data$cnt)
resultsrrf <- postResample(predrrf, test_data$cnt)
```

```
resultslm
```

```
##        RMSE    Rsquared         MAE
## 100.7249812   0.6828328  74.4598580
```

```
resultsrt
```

```
##        RMSE    Rsquared         MAE
## 112.5128137   0.6033501  82.0715589
```

```
resultsrrf
```

```
##       RMSE   Rsquared        MAE
## 46.0671290  0.9395406 31.0706531
```

Factorizing the categorical variables helped to improve the linear model performance. The RMSE of LM went down by 27,40%

RANDOM FOREST HAS THE BEST PERFORMANCE

RANDOM FOREST

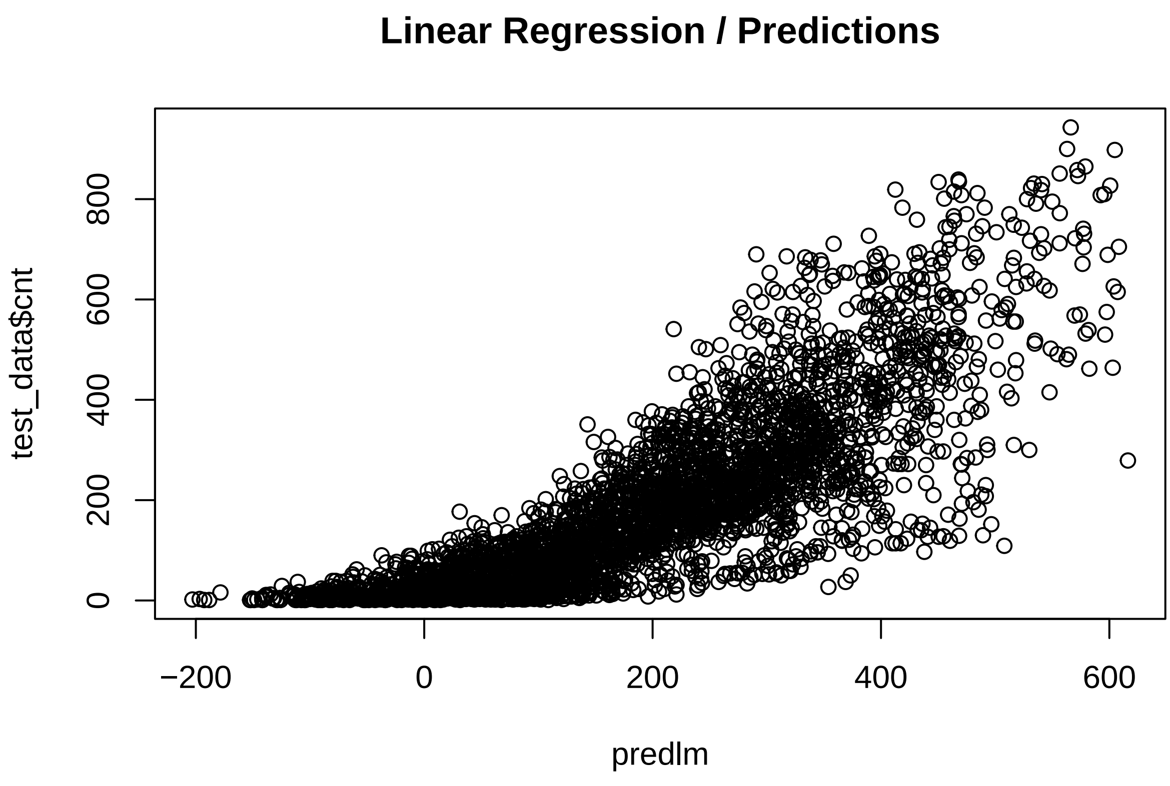
- **Root Mean Squared Error (RMSE)**:
  - Value: 46.0671290

- Interpretation: RMSE represents the square root of the average squared differences between the predicted values and the actual values. Lower RMSE values indicate better model performance, as they suggest that, on average, the model's predictions are closer to the actual values.

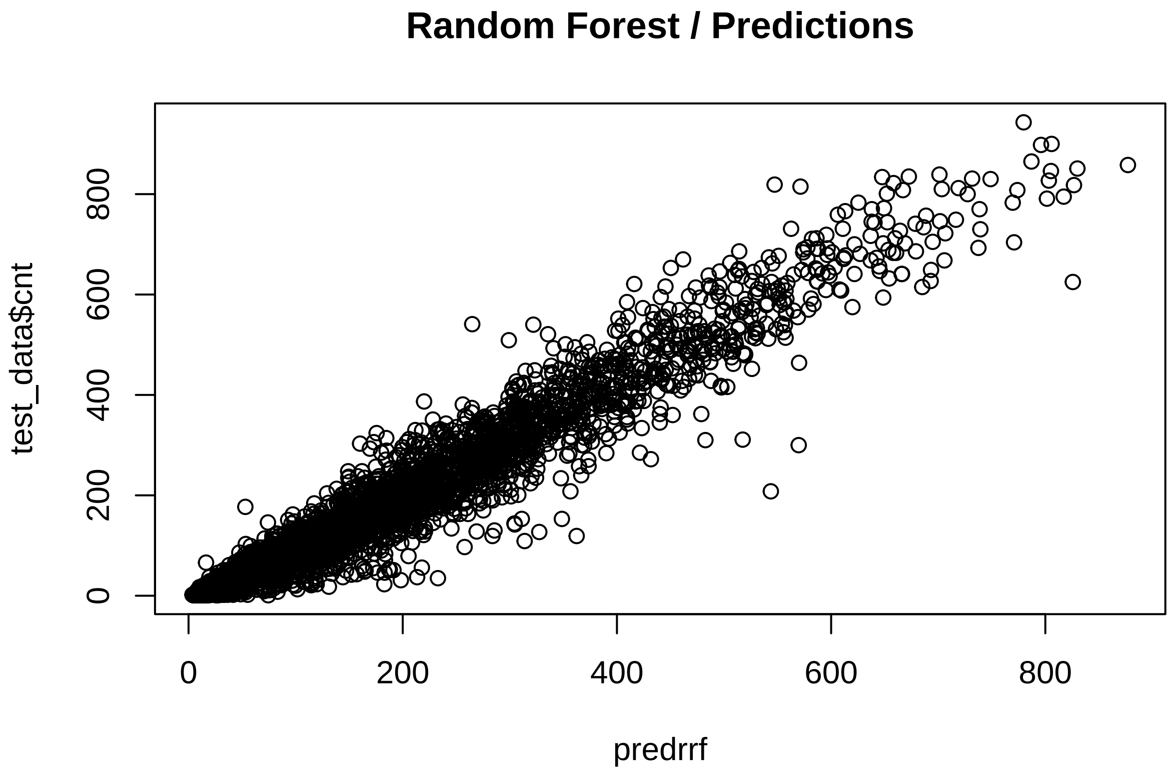
- **R-squared (Rsquared)**:
  - Value: 0.9395406
  - Interpretation: R-squared is a measure of the proportion of the variance in the dependent variable that is explained by the model. In this case, an R-squared of 0.9395406 suggests that approximately 94% of the variance in the bike rental count is explained by the Random Forest model. Higher R-squared values indicate better goodness of fit, meaning that the model captures a large portion of the variability in the target variable.
- **Mean Absolute Error (MAE)**:
  - Value: 31.0706531
  - Interpretation: MAE represents the average absolute differences between the predicted values and the actual values. In Random Forest model an MAE of 31.07 indicates the average magnitude of the errors in predicting bike rental counts. Like RMSE, lower MAE values are desirable, as they suggest that, on average, the model's predictions are closer to the actual values

Actual target value / Predicted value

```
plot(x=predlm, test_data$cnt, main = 'Linear Regression / Predictions')
```




```
plot(x=predrrf, y=test_data$cnt, main = 'Random Forest / Predictions')
```



These two plots showcase the better fit of Random Forest over the Linear Regression.