Bikes Rentals

Introduction

This study refers to a rental bikes system which contains hourly and daily counts informations between the years 2011 and 2012. The datasets was obtained from the UCI repository (https://archive.ics.uci.edu/ml/datasets/bike+sharing+dataset) and the subject of this study is to predict the number of bikes rentals (cnt response variable) regardless the climatic conditions or period of the year.

For that, we are going to make the use of three machine learning algorithms to predict the output variable: the Quasipoisson Regression, Linear Regression and Random Forest, and then, choose the best model that better fits the predictions against the actual values.

Attribute information

- instant: record index
- dteday : date
- season : season (1:springer, 2:summer, 3:fall, 4:winter)
- 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 [Web Link])
- 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=-8, t_max=+39 (only in hourly scale)
- atemp: Normalized feeling temperature in Celsius. The values are derived via (t-t_min)/(t_max-t_min), t_min=-16, t_max=+50 (only in hourly scale)
- 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

Part I - Data analysis and transformations

1. Importing the data

```
bikes <- read.csv('hour.csv')
```

2. Variables transformations

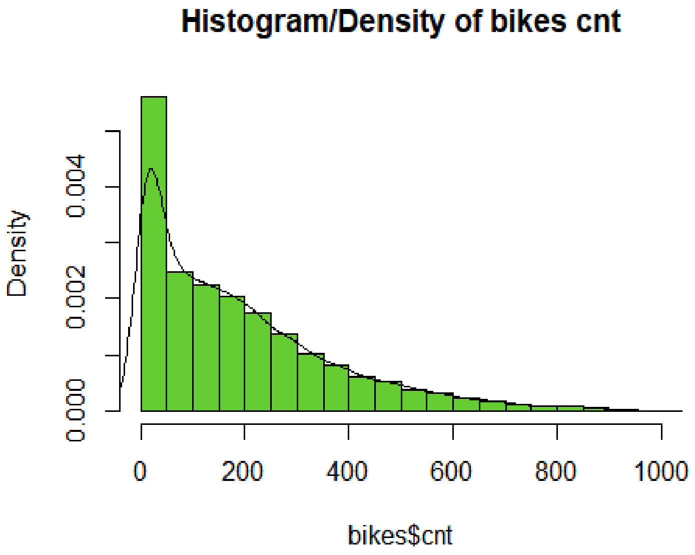
For better modeling, some variables will be transformed into categorical values and others into numeric.

```
for (i in 3:10) {
    bikes[, i] <- factor(bikes[, i])
}

for (i in 11:17) {
    bikes[, i] <- as.numeric(bikes[, i])
}

library(lubridate)

bikes$dteday <- ymd(bikes$dteday)

str(bikes)

## 'data.frame':    17379 obs. of  17 variables:
##  $ instant   : int  1 2 3 4 5 6 7 8 9 10 ...
##  $ dteday    : Date, format: "2011-01-01" "2011-01-01" ...
##  $ season    : Factor w/ 4 levels "1","2","3","4": 1 1 1 1 1 1 1 1 1 1 ...
##  $ yr        : Factor w/ 2 levels "0","1": 1 1 1 1 1 1 1 1 1 1 ...
##  $ 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 "0","1": 1 1 1 1 1 1 1 1 1 1 ...
##  $ weekday   : Factor w/ 7 levels "0","1","2","3",..: 7 7 7 7 7 7 7 7 7 7 ...
##  $ workingday: Factor w/ 2 levels "0","1": 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    : num  3 8 5 3 0 0 2 1 1 8 ...
##  $ registered: num  13 32 27 10 1 1 0 2 7 6 ...
##  $ cnt       : num  16 40 32 13 1 1 2 3 8 14 ...
```

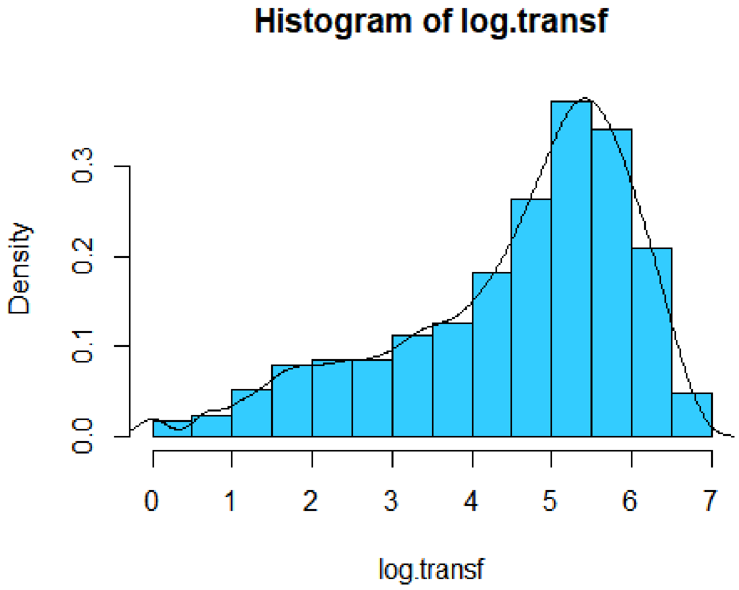
3. Log transformation

To minimize the right skew curve behavior, the log transformation was applied to the response variable, once to approximate to the normal curve and to be able to define the standard deviation interval.

```
hist(bikes$cnt, main = 'Histogram/Density of bikes cnt', col = '#66cc33', prob = T
RUE)
lines(density(bikes$cnt))
```



```
log.transf <- log(bikes$cnt)
hist(log.transf, col = '#33ccff', prob = TRUE)
lines(density(log.transf))
```



4. Scaling output variable

Assuming that after the log transformation applied previously, it will be defined on the transformed normal curve an interval of +/- 3 standard deviations around the mean, with 99% of confidence interval to identify and exclude the outliers values outside of this interval.

```
library(dplyr)

bikes$scl.cnt <- as.numeric(scale(bikes$cnt, center = T, scale = T))
range(bikes$scl.cnt)

[1] -1.039008  4.341735

bikes <- bikes %>%
  filter(scl.cnt >= -3.0 & scl.cnt <= 3.0)

range(bikes$scl.cnt)

[1] -1.039008  2.996549
```

5. Data analysis

In this section it will be displayed some graphs comparing the behaviors of the casual and registered users in the way that they rent the bikes according to climatic conditions, weekday, month or hour of the day. It is observable that the casual users are basically composed by the tourists, eventual users or the people that don´t utilize regularly the bikes for the working destination. Otherwise the proposal of use of the registered cyclists is essentially for the work locomotion.

a. temperature vs count

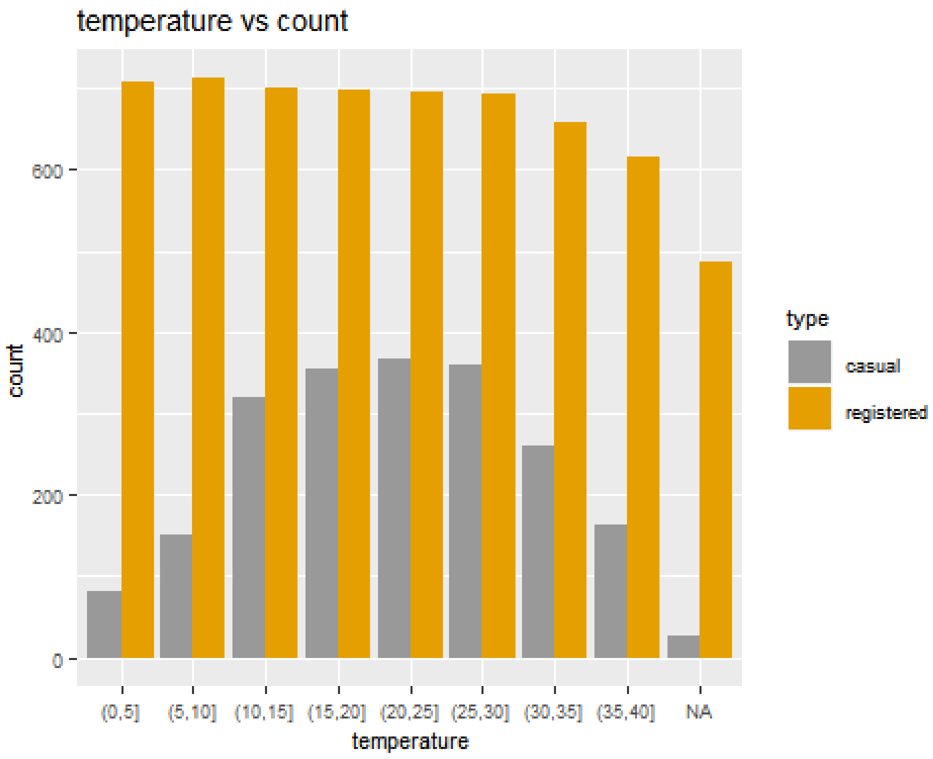
As observed, it is very clear that the casual users rent the bikes when the temperature conditions are nice, instead of the registered users, which the rental amounts practically remain unchanged, except in the extreme temperature conditions.

```
library(dplyr)
library(ggplot2)

library(tidyr)

bikes$temp.celsius <- bikes$temp*(max(bikes$temp)-min(bikes$temp))+min(bikes$temp)
bikes$temp.celsius <- bikes$temp*(39-(-8))+(-8)


casual <- bikes[, 15]
registered <- bikes[, 16]
temperature <- bikes[, 19]
weekday <- bikes[, 8]
weathersit <- bikes[, 10]
season <- bikes[, 3]
hour <- bikes[, 6]
month <- bikes[, 5]
```

```r
customer <- data.frame(casual, registered, temperature, weekday, weathersit, seaso
n, hour, month)
head(customer)
```

```
##   casual registered temperature weekday weathersit season hour month
## 1      3         13        3.28       6          1      1    0     1
## 2      8         32        2.34       6          1      1    1     1
## 3      5         27        2.34       6          1      1    2     1
## 4      3         10        3.28       6          1      1    3     1
## 5      0          1        3.28       6          1      1    4     1
## 6      0          1        3.28       6          2      1    5     1
```

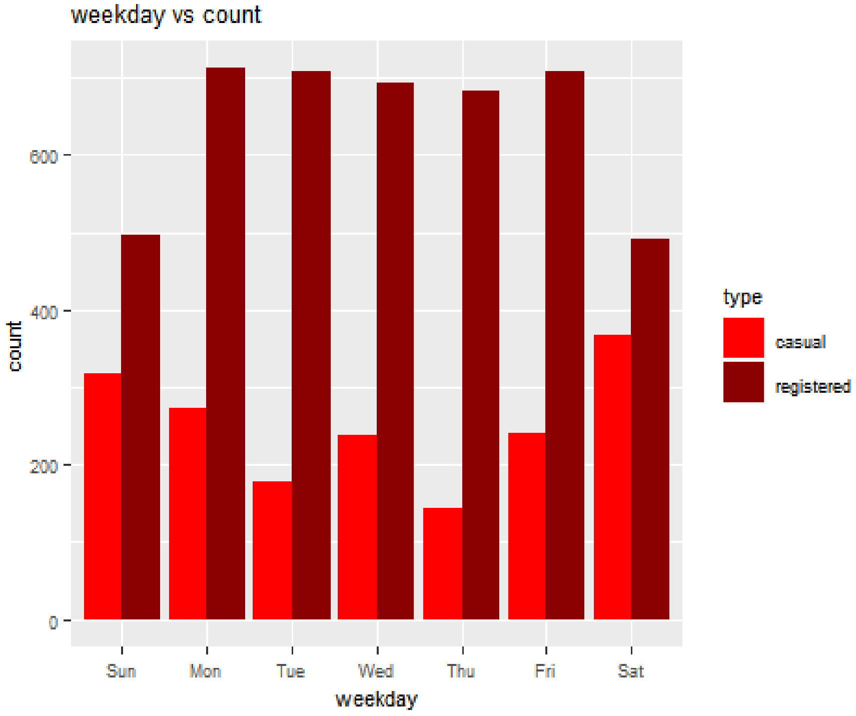
```r
cstm <- customer %>% gather(type, count, -c(temperature, weekday, weathersit, seas
on, hour, month))

cstm1 <- cstm %>%
  mutate(temperature = cut(temperature, breaks= c(0, 5, 10, 15, 20, 25, 30, 35, 40
)))

ggplot(data = cstm1, aes(x = temperature, y = count, fill = type)) +
  geom_bar(stat = 'identity', position = position_dodge())+
  scale_fill_manual(values = c("#999999", "#E69F00")) +
  ggtitle("temperature vs count") +
  theme(text = element_text(size=8.5),
        axis.text.x = element_text(angle=0))
```



b. weekday vs count

As predicted, the main utilization of the bikes for the registered users are for working, seen during the rentals of the week. In the opposite, on the weekend the rental proportion of casual users increases compared to weekday, denoting the recreational use adopted by the most of the casual users.

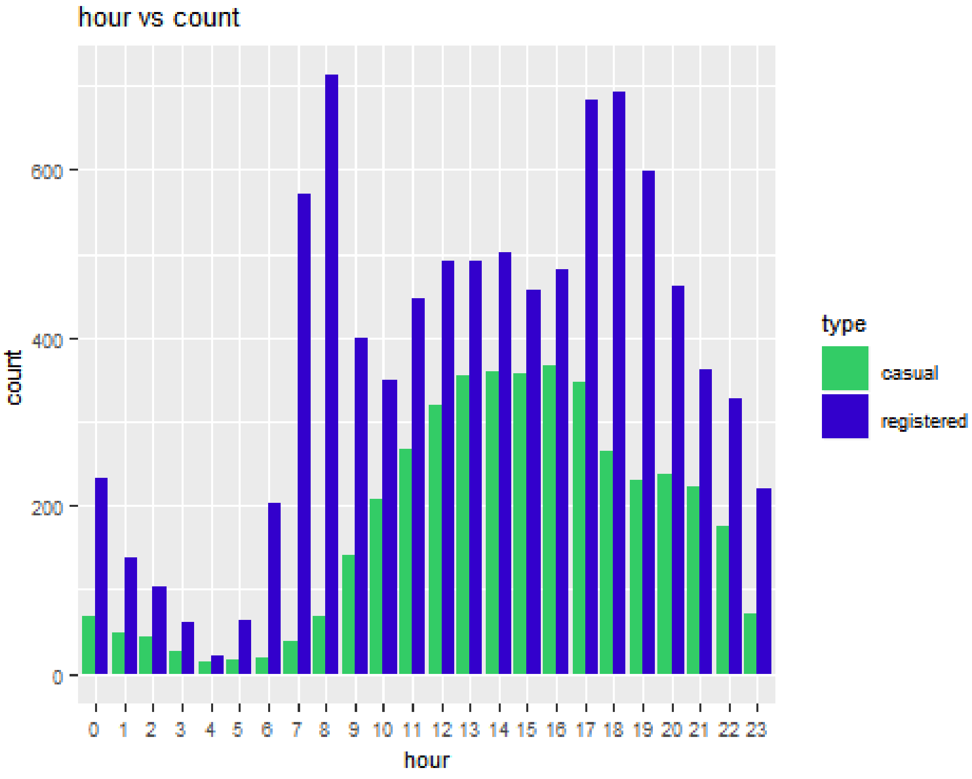
```r
library(plyr)

cstm$weekday <- mapvalues(cstm$weekday, from = c(0,1,2,3,4,5,6), to = c("Sun", "Mo
n", "Tue", "Wed", "Thu", "Fri", "Sat"))

ggplot(data = cstm, aes(x = weekday, y = count, fill = type)) +
  geom_bar(stat = 'identity', position = position_dodge()) +
  scale_fill_manual(values = c("red", "darkred")) +
  ggtitle("weekday vs count") +
  theme(text = element_text(size=8),
        axis.text.x = element_text(angle=0))
```



c. hour vs count

To corroborate the working utilization of the bikes by the registered users, the demand peaks can be clearly observed on the entrance and the exit regular working times, not noted for the casual cyclists.

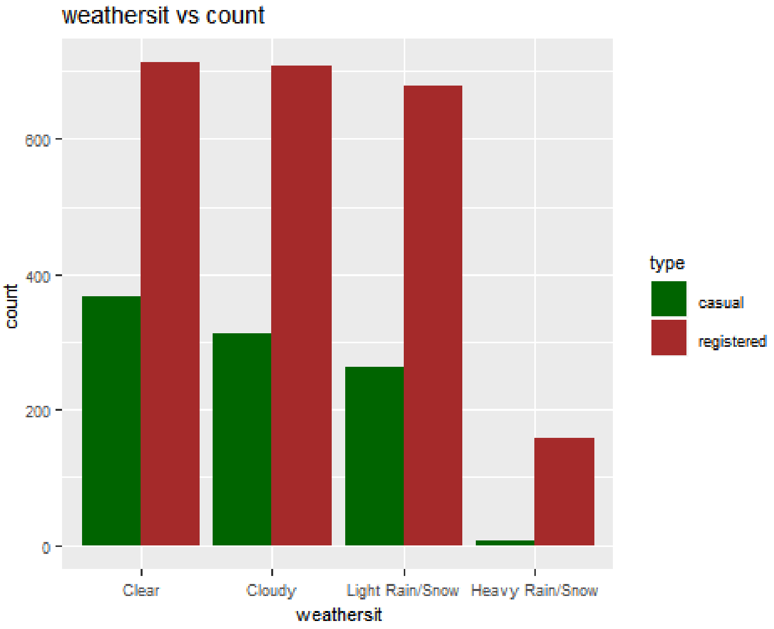
```r
ggplot(data = cstm1, aes(x = hour, y = count, fill = type)) +
  geom_bar(stat = 'identity', position = position_dodge())+
  scale_fill_manual(values = c("#33cc66", "#3300cc")) +
  ggtitle("hour vs count") +
  theme(text = element_text(size=8),
        axis.text.x = element_text(angle=0))
```



d. weathersit vs count

The weather situation as the temperature have strong correlation for the casual users to utilize or not the bikes due to climatic conditions. How worse it is, less they will rent it. But not in the case of registered users, on account the dependency for the use of the work.

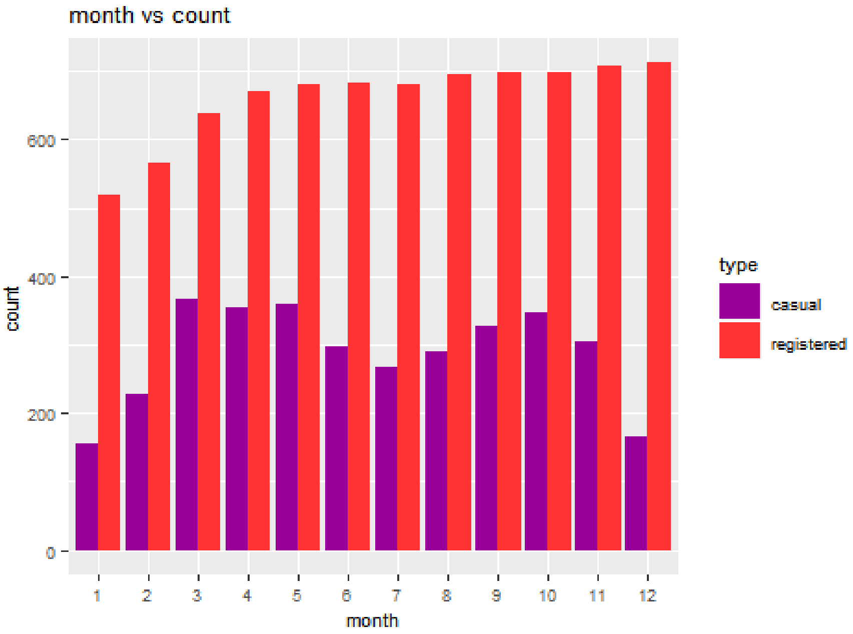
```
cstm1$weathersit <- mapvalues(cstm$weathersit, from = c(1,2,3,4), to = c("Clear",
"Cloudy", "Light Rain/Snow", "Heavy Rain/Snow"))

ggplot(data = cstm1, aes(x = weathersit, y = count, fill = type)) +
  geom_bar(stat = 'identity', position = position_dodge())+
  scale_fill_manual(values = c("darkgreen", "brown")) +
  ggtitle("weathersit vs count") +
  theme(text = element_text(size=8.5),
        axis.text.x = element_text(angle=0))
```



e. month vs count

Similarly to weather and temperature, casual cyclists are highly influenced by the climatic conditions.

```
ggplot(data = cstm1, aes(x = month, y = count, fill = type)) +
  geom_bar(stat = 'identity', position = position_dodge())+
  scale_fill_manual(values = c("#990099", "#FF3333")) +
  ggtitle("month vs count") +
  theme(text = element_text(size=8),
        axis.text.x = element_text(angle=0))
```



Part II - Predictions models

For the prediction of the number of rentals, the Quasipoisson Regression, Linear Regression and Random Forest models will be applied on the dataset, as well the error rate to measure the accuracy of each model.

1. Standardization of numeric values.

```
scl.num <- scale(bikes[, 11:16], center = T, scale = T)
scl.num <- as.data.frame(scl.num)
str(scl.num)

## 'data.frame':    17135 obs. of  6 variables:
##  $ temp      : num  -1.32 -1.43 -1.43 -1.32 -1.32 ...
##  $ atemp     : num  -1.08 -1.17 -1.17 -1.08 -1.08 ...
##  $ hum       : num  0.939 0.887 0.887 0.628 0.628 ...
##  $ windspeed : num  -1.55 -1.55 -1.55 -1.55 -1.55 ...
##  $ casual    : num  -0.653 -0.55 -0.612 -0.653 -0.715 ...
##  $ registered: num  -0.976 -0.836 -0.873 -0.998 -1.065 ...

bikes1 <- cbind(bikes[, 1:10], scl.num)

bikes <- cbind(bikes1, bikes[, 17])
names(bikes)[17] <- paste('cnt')
bikes$cnt <- as.integer(bikes$cnt)
str(bikes)

## 'data.frame':    17135 obs. of  17 variables:
##  $ instant   : int  1 2 3 4 5 6 7 8 9 10 ...
##  $ dteday    : Date, format: "2011-01-01" "2011-01-01" ...
##  $ season    : Factor w/ 4 levels "1","2","3","4": 1 1 1 1 1 1 1 1 1 1 ...
##  $ yr        : Factor w/ 2 levels "0","1": 1 1 1 1 1 1 1 1 1 1 ...
##  $ 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 "0","1": 1 1 1 1 1 1 1 1 1 1 ...
##  $ weekday   : Factor w/ 7 levels "0","1","2","3",..: 7 7 7 7 7 7 7 7 7 7 ...
##  $ workingday: Factor w/ 2 levels "0","1": 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  -1.32 -1.43 -1.43 -1.32 -1.32 ...
##  $ atemp     : num  -1.08 -1.17 -1.17 -1.08 -1.08 ...
##  $ hum       : num  0.939 0.887 0.887 0.628 0.628 ...
##  $ windspeed : num  -1.55 -1.55 -1.55 -1.55 -1.55 ...
##  $ casual    : num  -0.653 -0.55 -0.612 -0.653 -0.715 ...
##  $ registered: num  -0.976 -0.836 -0.873 -0.998 -1.065 ...
##  $ cnt       : int  16 40 32 13 1 1 2 3 8 14 ...
```

2. Applying one hot encoding to categorical variables

```
library(caret)

library(lubridate)
dummy <- dummyVars("~ season + yr + mnth + hr + holiday + weekday + workingday + w
eathersit", data = bikes)
bikes1 <- data.frame(predict(dummy, newdata = bikes))
bikes <- cbind(bikes[, 1:2], bikes1, bikes[, 11:17])
str(bikes)
```

```
## 'data.frame':    17135 obs. of  66 variables:
##  $ instant     : int  1 2 3 4 5 6 7 8 9 10 ...
##  $ dteday      : Date, format: "2011-01-01" "2011-01-01" ...
##  $ season.1    : num  1 1 1 1 1 1 1 1 1 1 ...
##  $ season.2    : num  0 0 0 0 0 0 0 0 0 0 ...
##  $ season.3    : num  0 0 0 0 0 0 0 0 0 0 ...
##  $ season.4    : num  0 0 0 0 0 0 0 0 0 0 ...
##  $ yr.0        : num  1 1 1 1 1 1 1 1 1 1 ...
##  $ yr.1        : num  0 0 0 0 0 0 0 0 0 0 ...
##  $ mnth.1      : num  1 1 1 1 1 1 1 1 1 1 ...
##  $ mnth.2      : num  0 0 0 0 0 0 0 0 0 0 ...
##  $ mnth.3      : num  0 0 0 0 0 0 0 0 0 0 ...
##  $ mnth.4      : num  0 0 0 0 0 0 0 0 0 0 ...
##  $ mnth.5      : num  0 0 0 0 0 0 0 0 0 0 ...
##  $ mnth.6      : num  0 0 0 0 0 0 0 0 0 0 ...
##  $ mnth.7      : num  0 0 0 0 0 0 0 0 0 0 ...
##  $ mnth.8      : num  0 0 0 0 0 0 0 0 0 0 ...
##  $ mnth.9      : num  0 0 0 0 0 0 0 0 0 0 ...
##  $ mnth.10     : num  0 0 0 0 0 0 0 0 0 0 ...
##  $ mnth.11     : num  0 0 0 0 0 0 0 0 0 0 ...
##  $ mnth.12     : num  0 0 0 0 0 0 0 0 0 0 ...
##  $ hr.0        : num  1 0 0 0 0 0 0 0 0 0 ...
##  $ hr.1        : num  0 1 0 0 0 0 0 0 0 0 ...
##  $ hr.2        : num  0 0 1 0 0 0 0 0 0 0 ...
##  $ hr.3        : num  0 0 0 1 0 0 0 0 0 0 ...
##  $ hr.4        : num  0 0 0 0 1 0 0 0 0 0 ...
##  $ hr.5        : num  0 0 0 0 0 1 0 0 0 0 ...
##  $ hr.6        : num  0 0 0 0 0 0 1 0 0 0 ...
##  $ hr.7        : num  0 0 0 0 0 0 0 1 0 0 ...
##  $ hr.8        : num  0 0 0 0 0 0 0 0 1 0 ...
##  $ hr.9        : num  0 0 0 0 0 0 0 0 0 1 ...
##  $ hr.10       : num  0 0 0 0 0 0 0 0 0 0 ...
##  $ hr.11       : num  0 0 0 0 0 0 0 0 0 0 ...
##  $ hr.12       : num  0 0 0 0 0 0 0 0 0 0 ...
##  $ hr.13       : num  0 0 0 0 0 0 0 0 0 0 ...
##  $ hr.14       : num  0 0 0 0 0 0 0 0 0 0 ...
##  $ hr.15       : num  0 0 0 0 0 0 0 0 0 0 ...
##  $ hr.16       : num  0 0 0 0 0 0 0 0 0 0 ...
##  $ hr.17       : num  0 0 0 0 0 0 0 0 0 0 ...
##  $ hr.18       : num  0 0 0 0 0 0 0 0 0 0 ...
##  $ hr.19       : num  0 0 0 0 0 0 0 0 0 0 ...
##  $ hr.20        : num  0 0 0 0 0 0 0 0 0 0 ...
##  $ hr.21        : num  0 0 0 0 0 0 0 0 0 0 ...
##  $ hr.22        : num  0 0 0 0 0 0 0 0 0 0 ...
##  $ hr.23        : num  0 0 0 0 0 0 0 0 0 0 ...
##  $ holiday.0    : num  1 1 1 1 1 1 1 1 1 1 ...
##  $ holiday.1    : num  0 0 0 0 0 0 0 0 0 0 ...
##  $ weekday.0    : num  0 0 0 0 0 0 0 0 0 0 ...
##  $ weekday.1    : num  0 0 0 0 0 0 0 0 0 0 ...
##  $ weekday.2    : num  0 0 0 0 0 0 0 0 0 0 ...
##  $ weekday.3    : num  0 0 0 0 0 0 0 0 0 0 ...
##  $ weekday.4    : num  0 0 0 0 0 0 0 0 0 0 ...
##  $ weekday.5    : num  0 0 0 0 0 0 0 0 0 0 ...
##  $ weekday.6    : num  1 1 1 1 1 1 1 1 1 1 ...
##  $ workingday.0: num  1 1 1 1 1 1 1 1 1 1 ...
##  $ workingday.1: num  0 0 0 0 0 0 0 0 0 0 ...
##  $ weathersit.1: num  1 1 1 1 1 0 1 1 1 1 ...
##  $ weathersit.2: num  0 0 0 0 0 1 0 0 0 0 ...
##  $ weathersit.3: num  0 0 0 0 0 0 0 0 0 0 ...
##  $ weathersit.4: num  0 0 0 0 0 0 0 0 0 0 ...
##  $ temp         : num  -1.32 -1.43 -1.43 -1.32 -1.32 ...
##  $ atemp        : num  -1.08 -1.17 -1.17 -1.08 -1.08 ...
##  $ hum          : num  0.939 0.887 0.887 0.628 0.628 ...
##  $ windspeed    : num  -1.55 -1.55 -1.55 -1.55 -1.55 ...
##  $ casual       : num  -0.653 -0.55 -0.612 -0.653 -0.715 ...
##  $ registered   : num  -0.976 -0.836 -0.873 -0.998 -1.065 ...
##  $ cnt          : int  16 40 32 13 1 1 2 3 8 14 ...
```

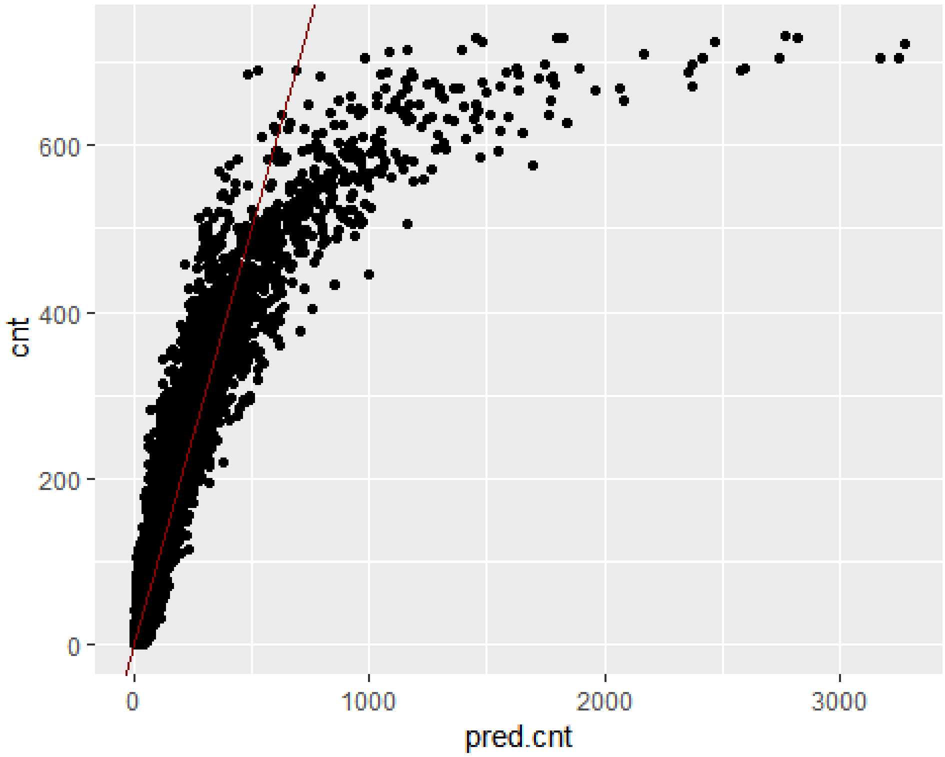
3. Machine learning models applications

a. Quasipoisson model

```
set.seed(123)

n <- nrow(bikes)
shuffled <- bikes[sample(n),]
train_indices <- 1:round(0.7 * n)
bikes.train <- shuffled[train_indices, ]
test_indices <- (round(0.7 * n) + 1):n
bikes.test <- shuffled[test_indices, ]

mean_bikes <- mean(bikes$cnt)
mean_bikes

## [1] 180.4905

var_bikes <- var(bikes$cnt)
var_bikes

## [1] 27589.57
```

```
model.glm <- glm(log(cnt) ~., data = bikes.train, family = quasipoisson)
bikes.test$pred.cnt <- predict(model.glm, bikes.test, type = 'response')

bikes.test$pred.cnt <- exp(bikes.test$pred.cnt)

bikes.test %>%
  mutate(residual = cnt - pred.cnt) %>%
  summarize(rmse  = sqrt(mean(residual^2)))

##        rmse
##       163.23
```

```
library(ggplot2)

ggplot(bikes.test, aes(x = pred.cnt, y = cnt)) +
  geom_point() +
  geom_abline(color = "darkred")
```

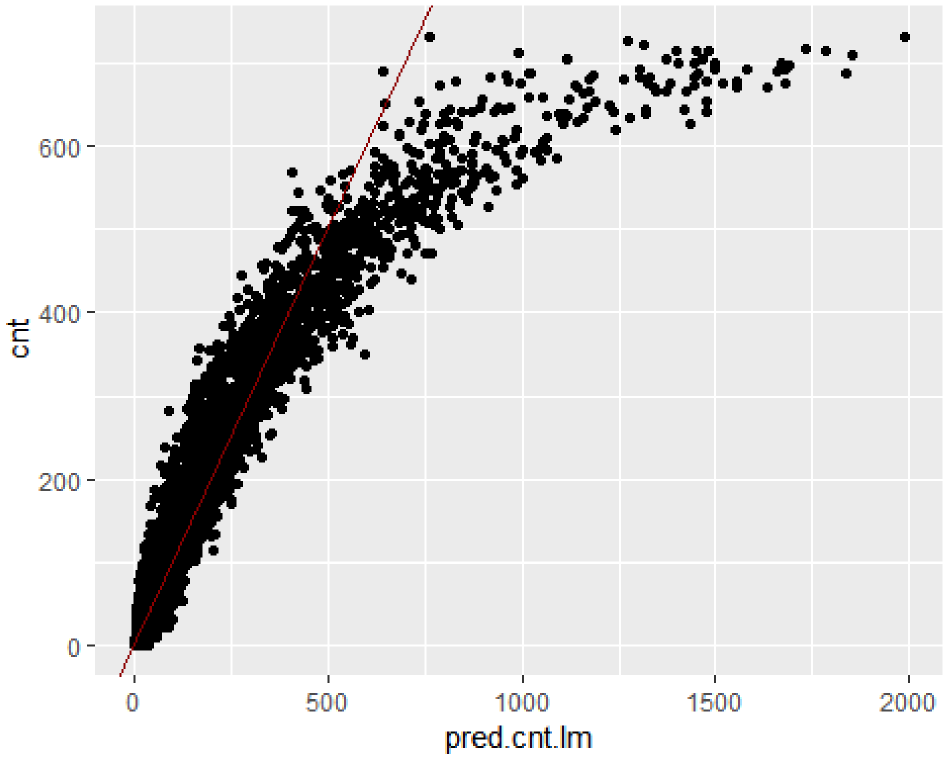


```
bikes.test %>%
  select(cnt, pred.cnt) %>%
  summarise(error = mean((abs(cnt - pred.cnt)/cnt)*100))

##       error
##       52.59
```

b. Linear regression

```
model.lm <- lm(log(cnt)~., data = bikes.train)
bikes.test$pred.cnt.lm <- predict(model.lm, bikes.test)

bikes.test$pred.cnt.lm <- exp(bikes.test$pred.cnt)

bikes.test %>%
  mutate(residual = cnt - pred.cnt.lm) %>%
  summarize(rmse  = sqrt(mean(residual^2)))

##       rmse
##      106.87
```

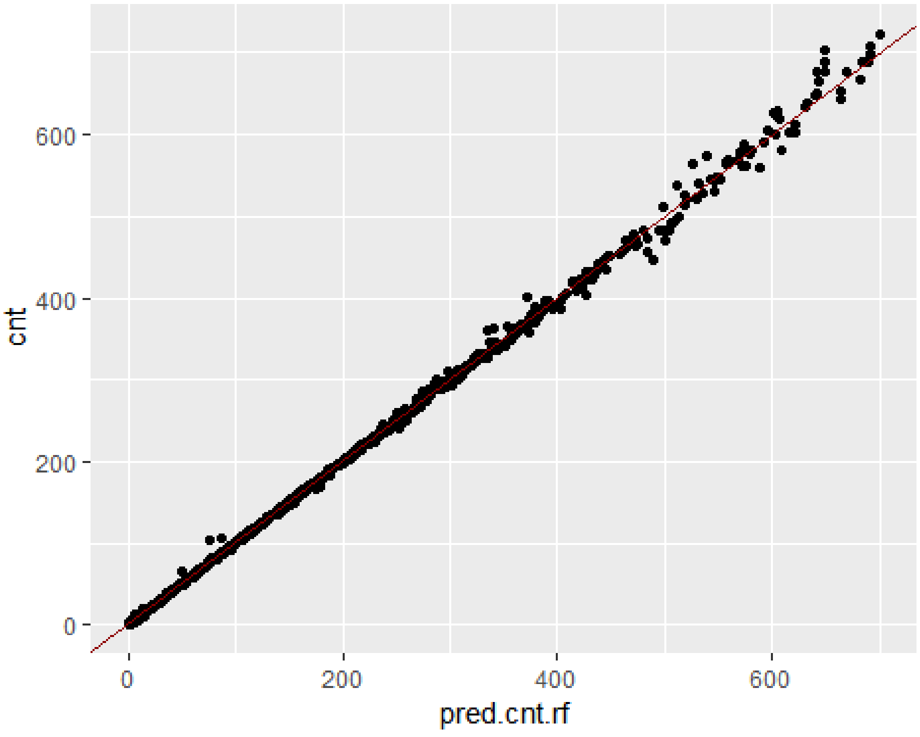
```
ggplot(bikes.test, aes(x = pred.cnt.lm, y = cnt)) +
  geom_point() +
  geom_abline(color = "darkred")
```



```
bikes.test %>%
  select(cnt, pred.cnt.lm) %>%
  summarise(error = mean((abs(cnt - pred.cnt.lm)/cnt)*100))

##      error
##      40.70
```

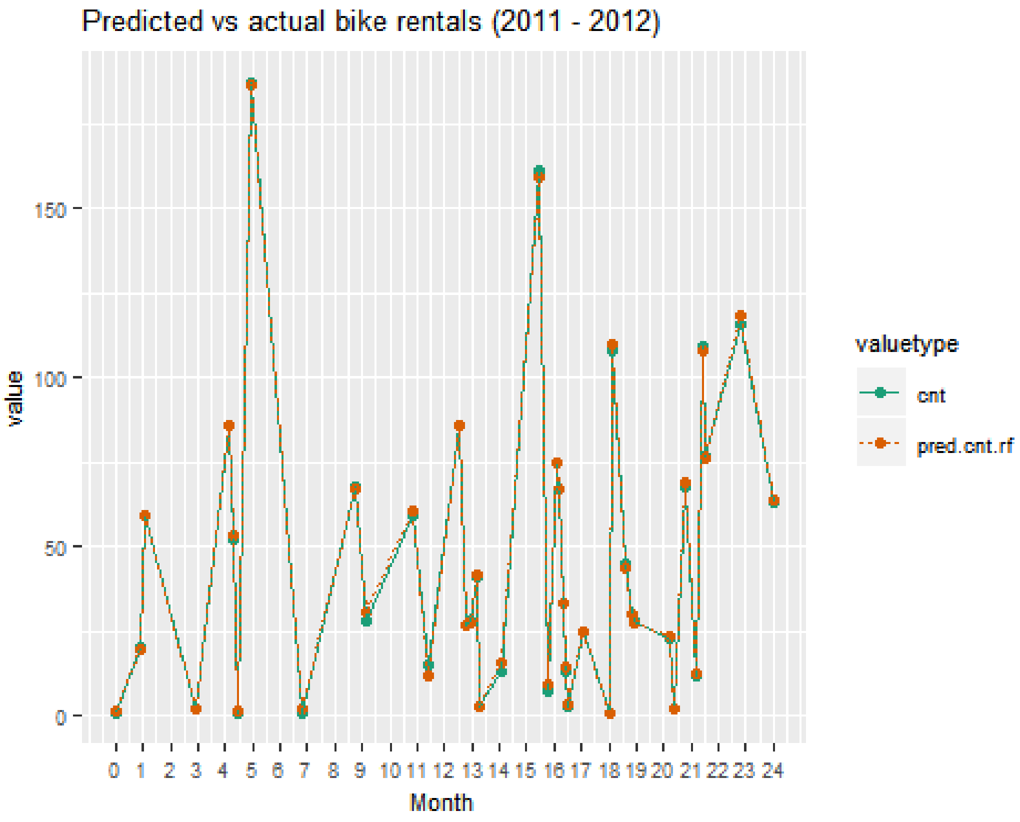
c. Random Forest

```
library(caret)
n <- nrow(bikes)
bikes <- bikes[sample(n),]
bikes <- subset(bikes[1:3000,])

partition <- createDataPartition(bikes$cnt, p = .7,
                                 list = FALSE,
                                 times = 1)
train.rf <- bikes[partition, ]
test.rf <- bikes[-partition, ]

control <- trainControl(method = 'cv', number = 10)
model.rf <- train(cnt ~., data = train.rf, method = 'rf')
test.rf$pred.cnt.rf <- predict(model.rf, test.rf)


test.rf %>%
  mutate(residual = cnt - pred.cnt.rf) %>%
  summarize(rmse  = sqrt(mean(residual^2)))

##        rmse
##        6.16

ggplot(test.rf, aes(x = pred.cnt.rf, y = cnt)) +
  geom_point() +
  geom_abline(color = "darkred")
```



```
test.rf %>%
  select(cnt, pred.cnt.rf) %>%
  summarise(error = mean((abs(cnt - pred.cnt.rf)/cnt)*100))

##       error
##       3.55
```

```
library(tidyr)

test.rf %>%
  # set start to 0, convert unit to days
  mutate(instant = (instant - min(instant))/30.3) %>%
  # gather cnt and pred into a value column
  gather(key = valuetype, value = value, cnt, pred.cnt.rf) %>%
  filter(instant < 24) %>% # restric to first 14 days
  # plot value by instant
  ggplot(aes(x = instant, y = value, color = valuetype, linetype = valuetype)) +
  geom_point() +
  geom_line() +
  scale_x_continuous("Month", breaks = 0:24, labels = 0:24) +
  scale_color_brewer(palette = "Dark2") +
  ggtitle("Predicted vs actual bike rentals (2011 - 2012)") +
  theme(text = element_text(size=8),
        axis.text.x = element_text(angle=0))
```



Conclusion

By the data analysis it was observed the main differences of the bikes employment between the casual and the registered users, the first one using occasionally and on the better climatic conditions, and the second group focusing as a vehicle to working destination.

Regarding the predictions algorithms, despite the intense computational needs, the Random Forest model presented a much better accuracy results compared to others, justifying its use for the bike rentals predictions.