Prediction of beer consumption in Sao Paulo Brazil

Loading dependent libraries:

```
library(dplyr)
library(ggplot2)
library(readr)
library(caret)
library(stringr)
library(purrr)
library(lubridate)
library(corrplot)
library(caretEnsemble)
```

Here we import the data using the readr library.

In our raw data commas are used as decimal points. readr reads the temperature data as integers and ignores the commas so we import the numeric values as characters and will convert them later on after replacing the commas with a decimal point

```
beerdata<-read_csv('Consumo_cerveja.csv',col_types = cols(

  `Temperatura Media (C)` = col_character(),
  `Temperatura Minima (C)` = col_character(),
  `Temperatura Maxima (C)` = col_character(),
  `Precipitacao (mm)` = col_character(),
  `Final de Semana` = col_character()
))
```

```
str(beerdata)
```

```
## Classes 'spec_tbl_df', 'tbl_df', 'tbl' and 'data.frame': 941 obs. of  7 variables:
##  $ Data                       : Date, format: "2015-01-01" "2015-01-02" ...
##  $ Temperatura Media (C)      : chr  "27,3" "27,02" "24,82" "23,98" ...
##  $ Temperatura Minima (C)     : chr  "23,9" "24,5" "22,4" "21,5" ...
##  $ Temperatura Maxima (C)     : chr  "32,5" "33,5" "29,9" "28,6" ...
##  $ Precipitacao (mm)          : chr  "0" "0" "0" "1,2" ...
##  $ Final de Semana            : chr  "0" "0" "1" "1" ...
##  $ Consumo de cerveja (litros): num  25.5 29 30.8 29.8 28.9 ...
##  - attr(*, "spec")=
##   .. cols(
##   ..   Data = col_date(format = ""),
##   ..   `Temperatura Media (C)` = col_character(),
##   ..   `Temperatura Minima (C)` = col_character(),
##   ..   `Temperatura Maxima (C)` = col_character(),
##   ..   `Precipitacao (mm)` = col_character(),
##   ..   `Final de Semana` = col_character(),
##   ..   `Consumo de cerveja (litros)` = col_double()
##   .. )
```

Converting the column names from Spanish to English

```
colnames(beerdata)<- c('date','mean_temp','min_temp','max_temp','rainfall','endofweek','beerconsumption')
```

```
head(beerdata)
```

```
## # A tibble: 6 x 7
##   date       mean_temp min_temp max_temp rainfall endofweek beerconsumption
##   <date>     <chr>     <chr>    <chr>    <chr>    <chr>               <dbl>
## 1 2015-01-01 27,3      23,9     32,5     0        0                    25.5
## 2 2015-01-02 27,02     24,5     33,5     0        0                    29.0
## 3 2015-01-03 24,82     22,4     29,9     0        1                    30.8
## 4 2015-01-04 23,98     21,5     28,6     1,2      1                    29.8
## 5 2015-01-05 23,82     21       28,3     0        0                    28.9
## 6 2015-01-06 23,78     20,1     30,5     12,2     0                    28.2
```

Trimming the dataset to include only the first 365 rows as the data only pertains to one year and the remainig values are NULL.

```
beerdata<-beerdata[ 1:365,]
tail(beerdata)
```

```
## # A tibble: 6 x 7
##   date       mean_temp min_temp max_temp rainfall endofweek beerconsumption
##   <date>     <chr>     <chr>    <chr>    <chr>    <chr>               <dbl>
## 1 2015-12-26 23,34     17,8     29,8     94,8     1                    22.0
## 2 2015-12-27 24        21,1     28,2     13,6     1                    32.3
## 3 2015-12-28 22,64     21,1     26,7     0        0                    26.1
## 4 2015-12-29 21,68     20,3     24,1     10,3     0                    22.3
## 5 2015-12-30 21,38     19,3     22,4     6,3      0                    20.5
## 6 2015-12-31 24,76     20,2     29       0        0                    22.4
```

Replacing commas with decimal points

```
removecomma <- function(x){
x<-str_replace(x,',','.')
return(x)
}
```

```
beerdata$min_temp<-beerdata$min_temp%>%map_chr(removecomma)
beerdata$max_temp<-beerdata$max_temp%>%map_chr(removecomma)
beerdata$mean_temp<-beerdata$mean_temp%>%map_chr(removecomma)
beerdata$rainfall<-beerdata$rainfall%>%map_chr(removecomma)
```

```
for(i in 2:5){
  beerdata[[i]]<-as.double(beerdata[[i]])
}
```

```
head(beerdata)
```

```
## # A tibble: 6 x 7
##   date       mean_temp min_temp max_temp rainfall endofweek beerconsumption
##   <date>         <dbl>    <dbl>    <dbl>    <dbl> <chr>               <dbl>
## 1 2015-01-01      27.3     23.9     32.5      0   0                    25.5
## 2 2015-01-02      27.0     24.5     33.5      0   0                    29.0
## 3 2015-01-03      24.8     22.4     29.9      0   1                    30.8
## 4 2015-01-04      24.0     21.5     28.6      1.2 1                    29.8
## 5 2015-01-05      23.8     21       28.3      0   0                    28.9
## 6 2015-01-06      23.8     20.1     30.5     12.2 0                    28.2
```

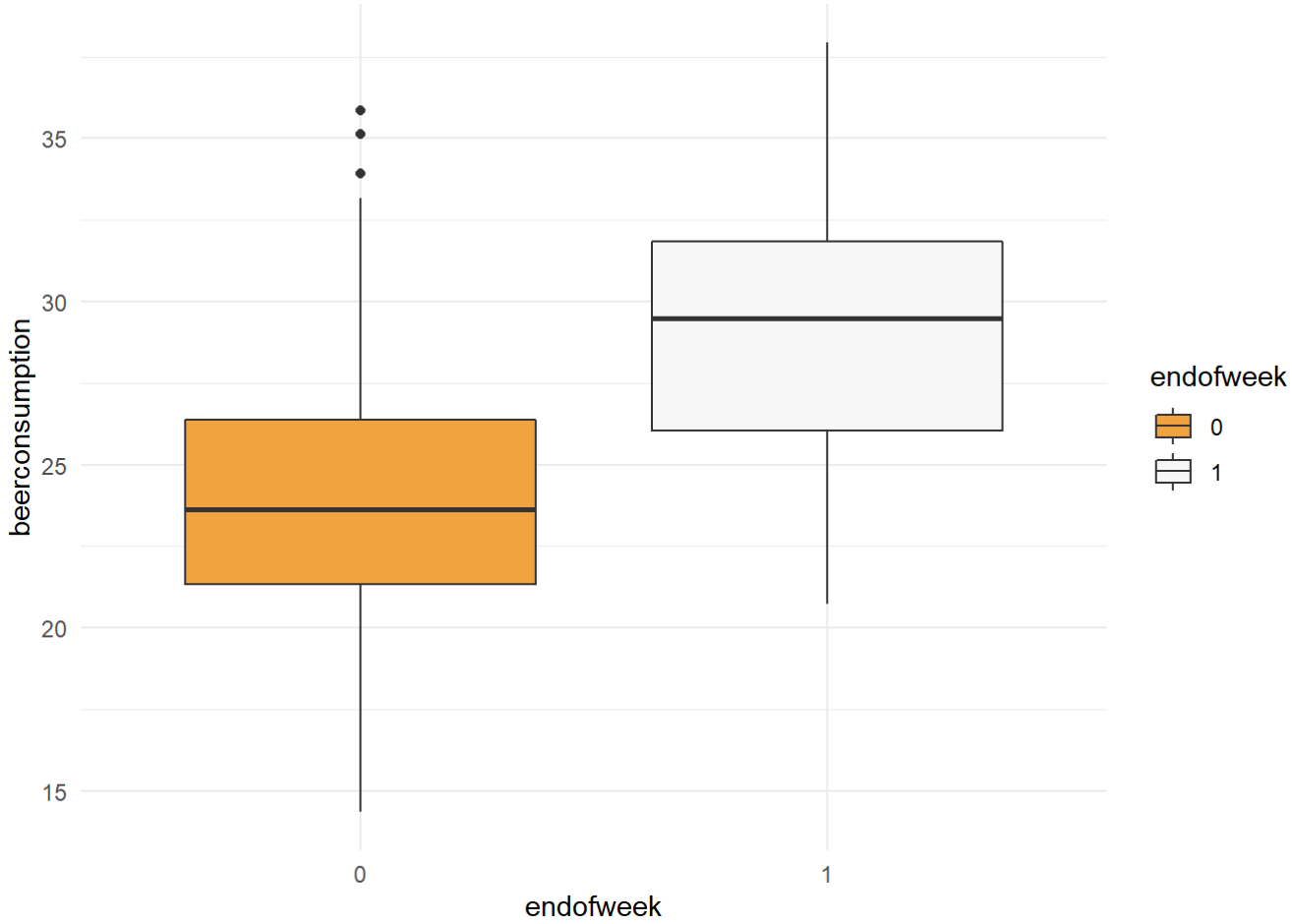
Checking for NULL values

```
sum(is.na(beerdata))
```

```
## [1] 0
```

It seems there is higher beer consumtion on weekends

```
ggplot(data=beerdata,aes(x=endofweek,y=beerconsumption))+geom_boxplot(aes(fill=endofweek))+

scale_fill_brewer(type='div',palette = 4)+theme_minimal()
```

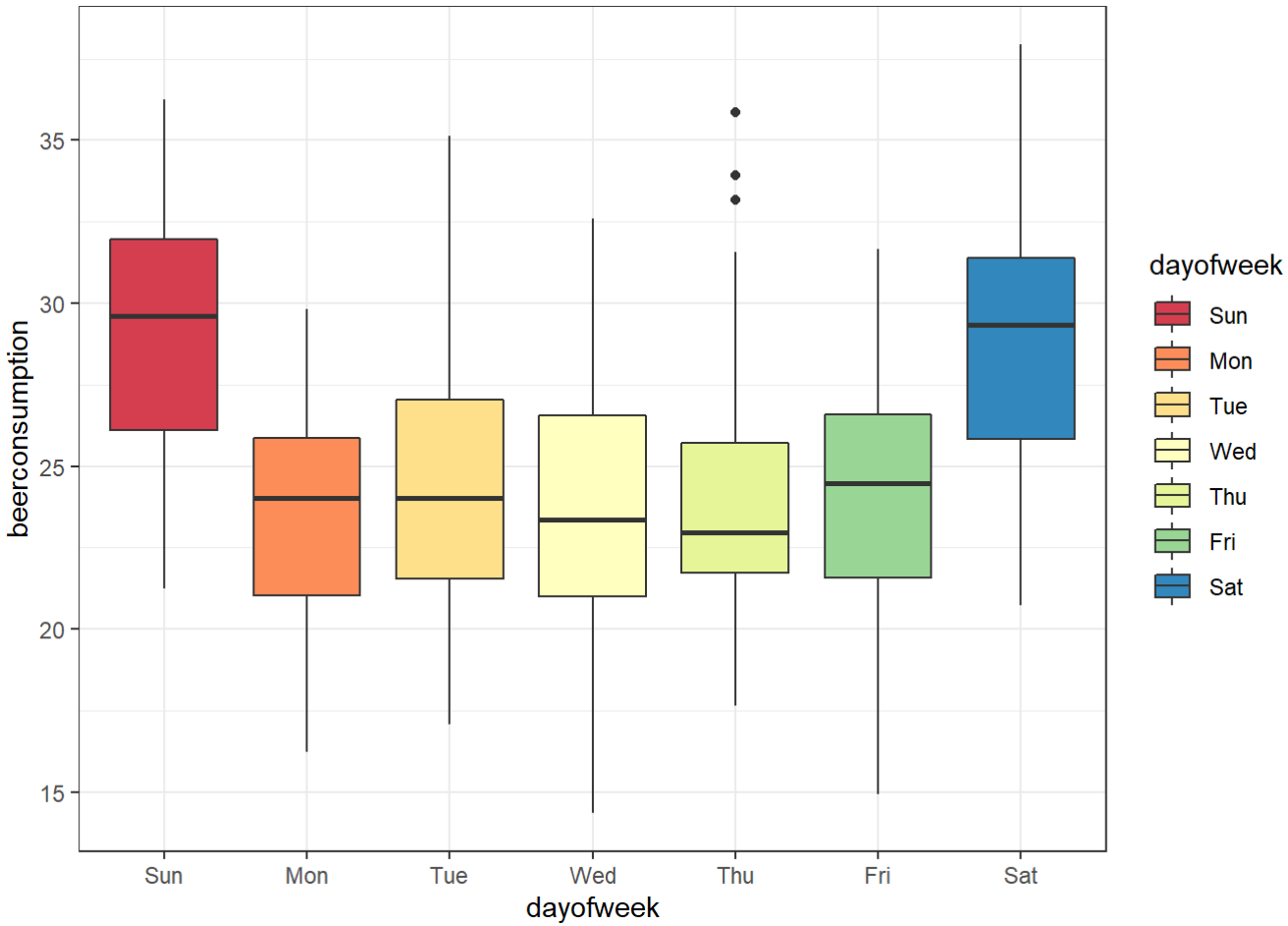


Adding day of week as a new column

```
beerdata<-beerdata%>%mutate(dayofweek=wday(beerdata$date,label = TRUE))
```

There is a marked increase in beer consumtion on SATUDRAY and SUNDAY

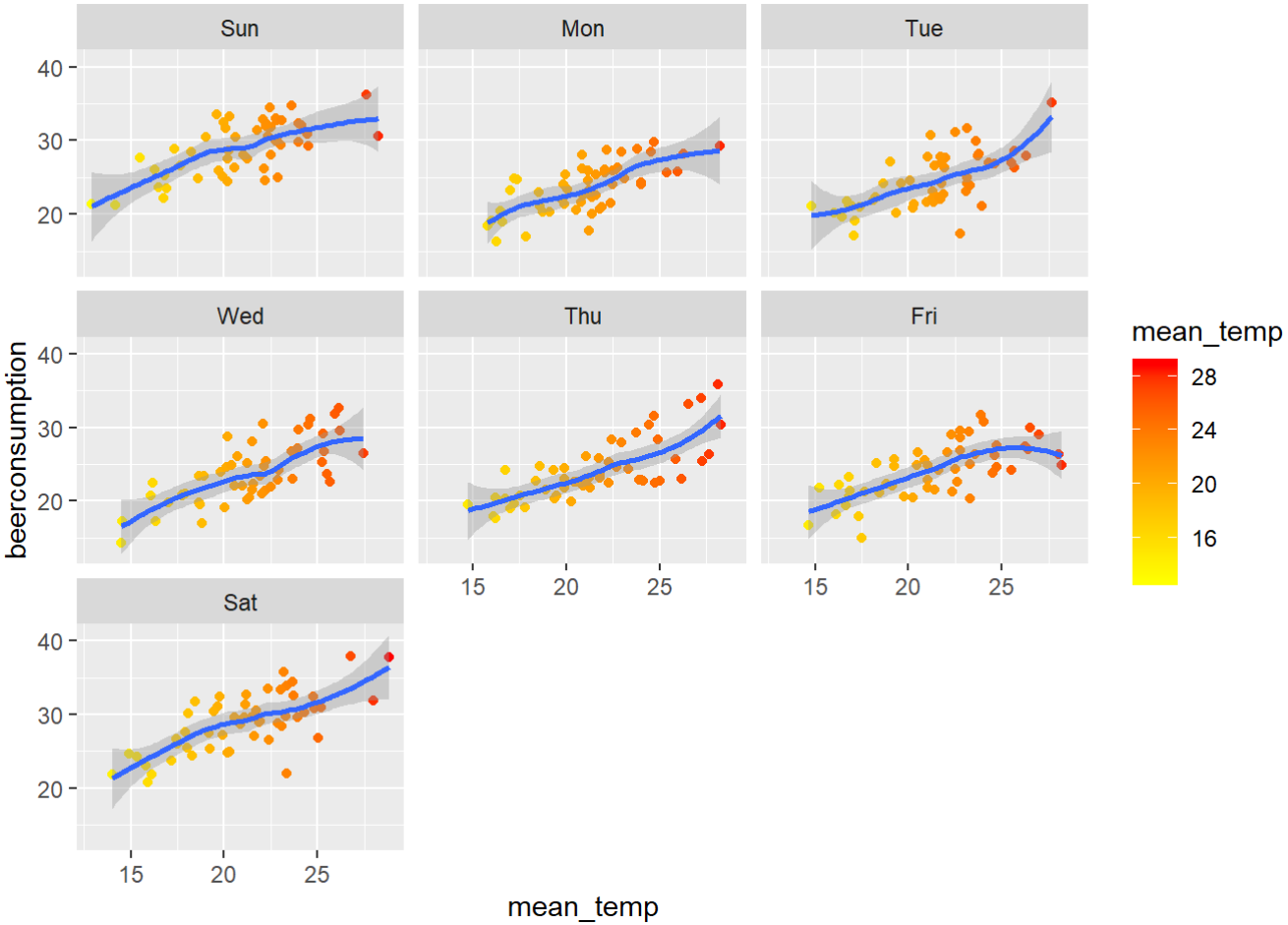
```
ggplot(data=beerdata,aes(x=dayofweek,y=beerconsumption))+geom_boxplot(aes(fill=dayofweek))+

scale_fill_brewer(palette = 'Spectral')+theme_bw()
```



There seems to be a positive relationship between temperature and beer consumption fo evey day of the week.

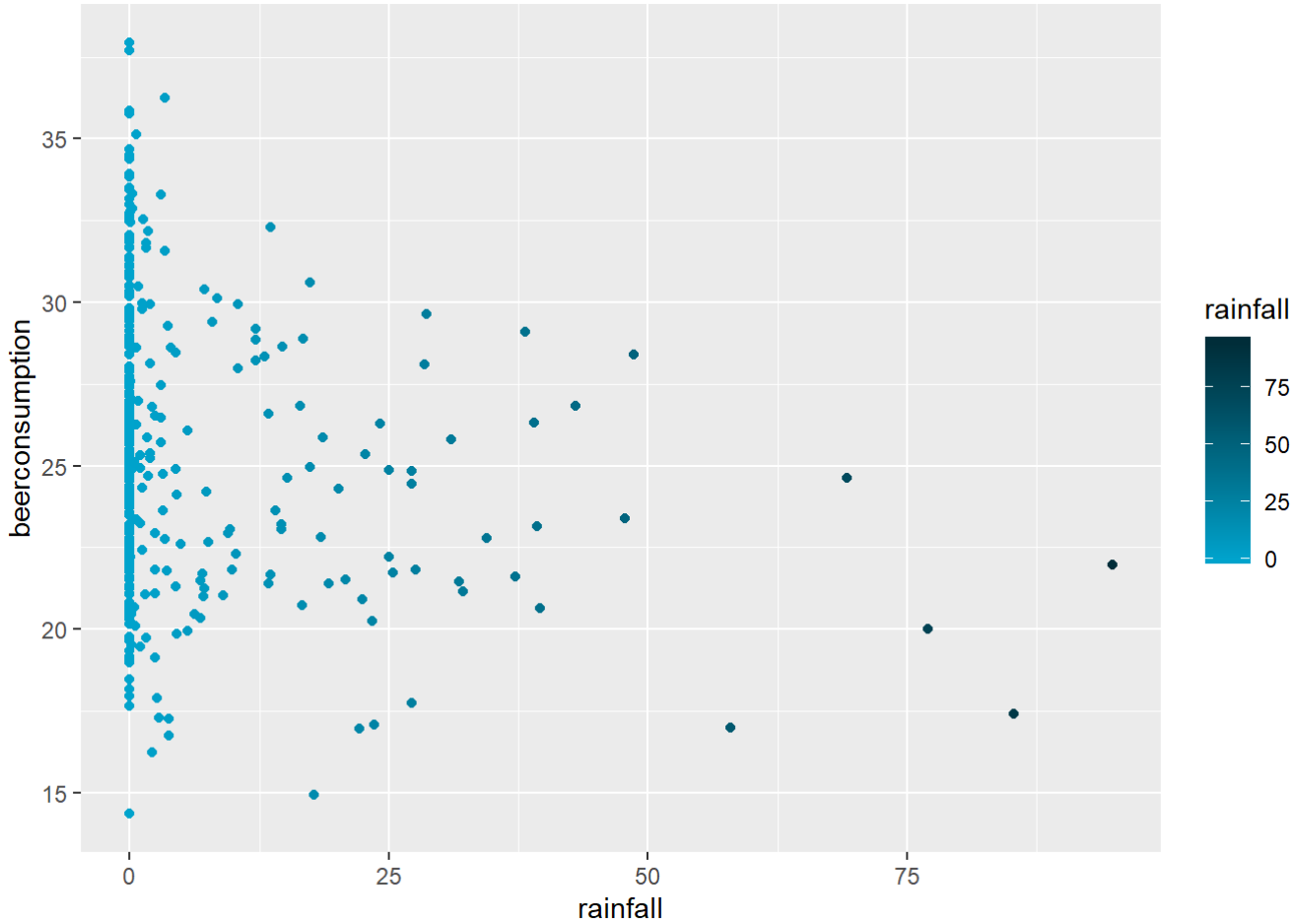
```
ggplot(data=beerdata,aes(x=mean_temp,y=beerconsumption))+geom_point(aes(color=mean_temp))+
  geom_smooth()+facet_wrap(~dayofweek)+scale_color_gradient(low="yellow", high="red")
```

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



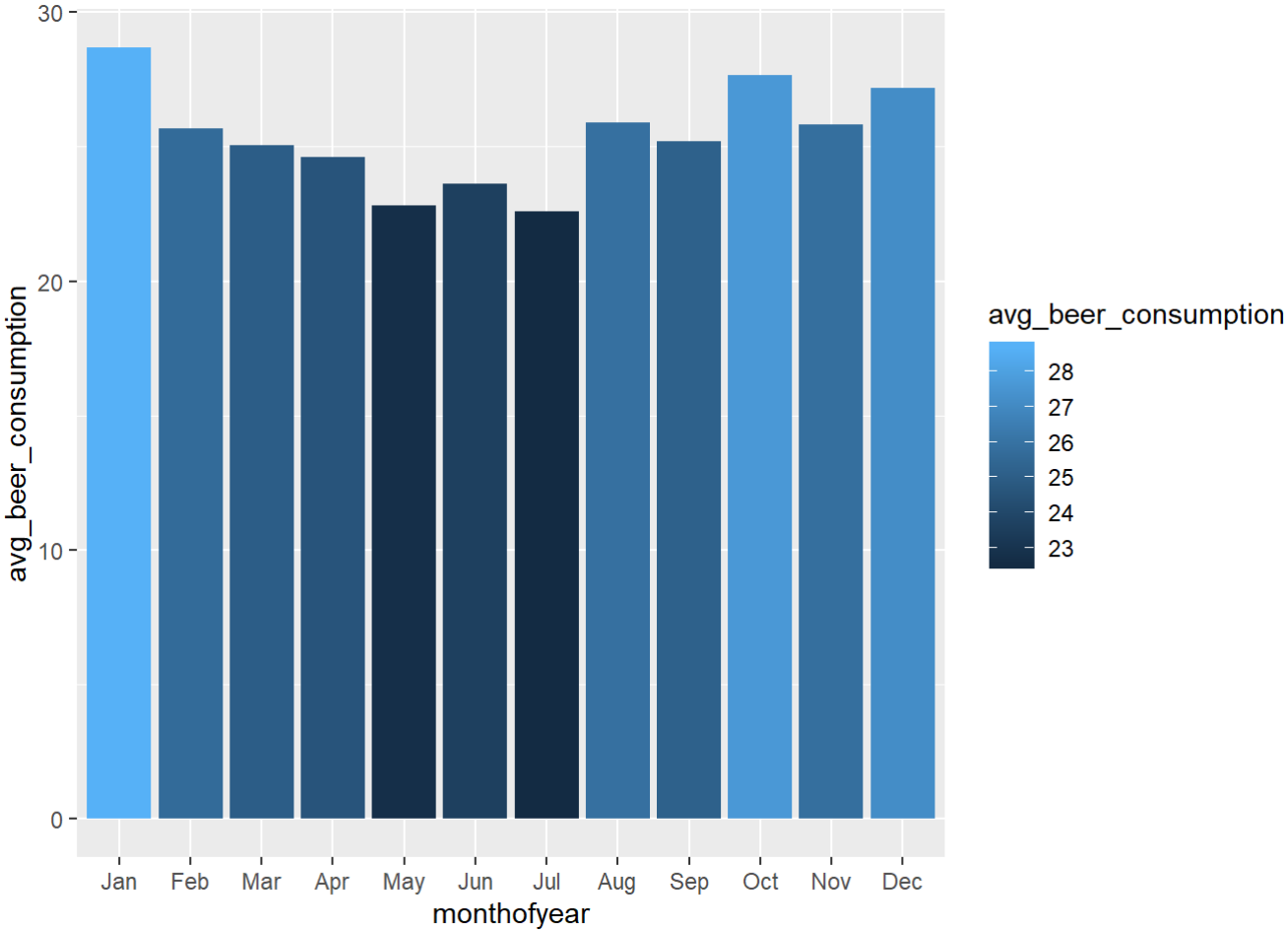
Since most days of the year in So Paulo have little o no rainfall,it is hard to infer any relationship between rainfall and beer consumption

```
ggplot(data=beerdata,aes(x=rainfall,y=beerconsumption))+geom_point(aes(color=rainfall))+
scale_color_gradient(low="#00a3cc", high="#002d39")
```



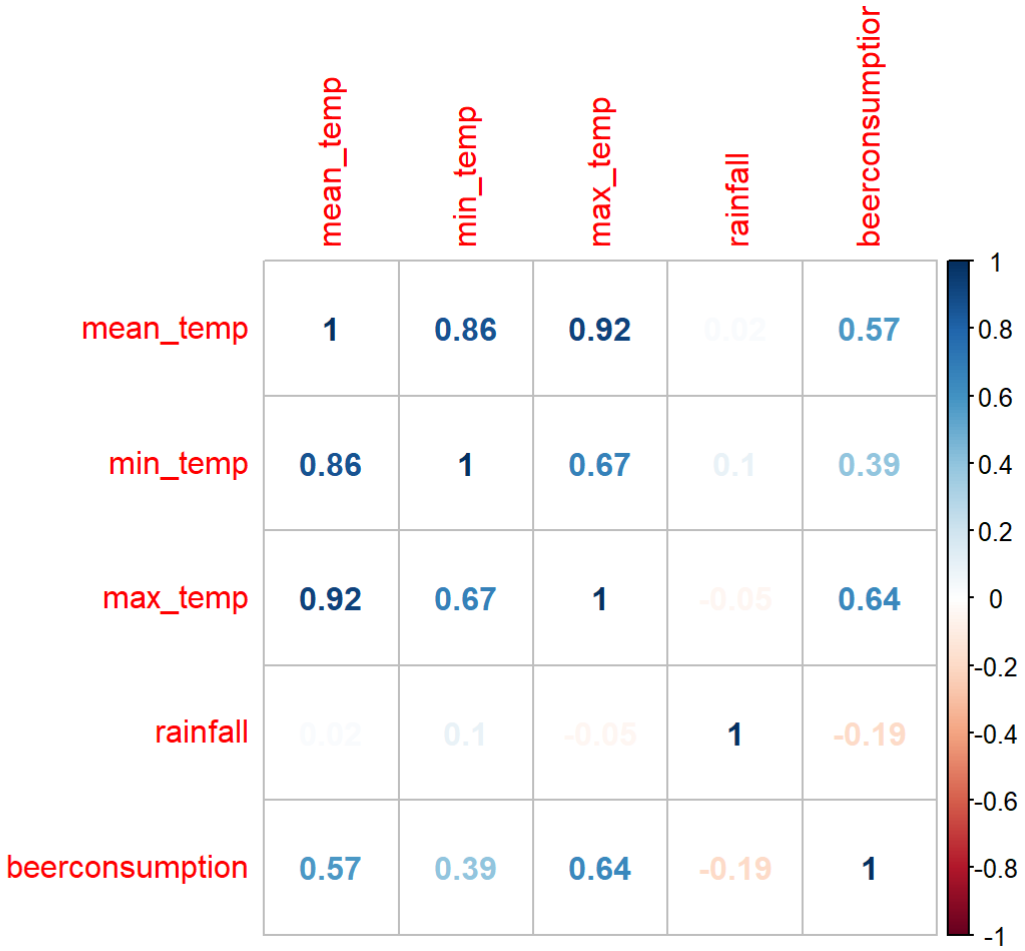
Adding the month to the data we notice that the consumption of beer dips towards the middle of the year but not by much

```
beerdata%>%mutate(monthofyear=month(date,label = TRUE))%>%group_by(monthofyear)%>%summarise(avg_beer_consumption=mean(beerco
nsumption))%>%ggplot(aes(x=monthofyear,y=avg_beer_consumption,fill=avg_beer_consumption))+geom_bar(stat = 'identity')
```



None of the features are too correlated with beer consumtion

```
correlation<-cor(beerdata%>%select(-c(date,endofweek,dayofweek)))
corrplot(correlation, method="number")
```

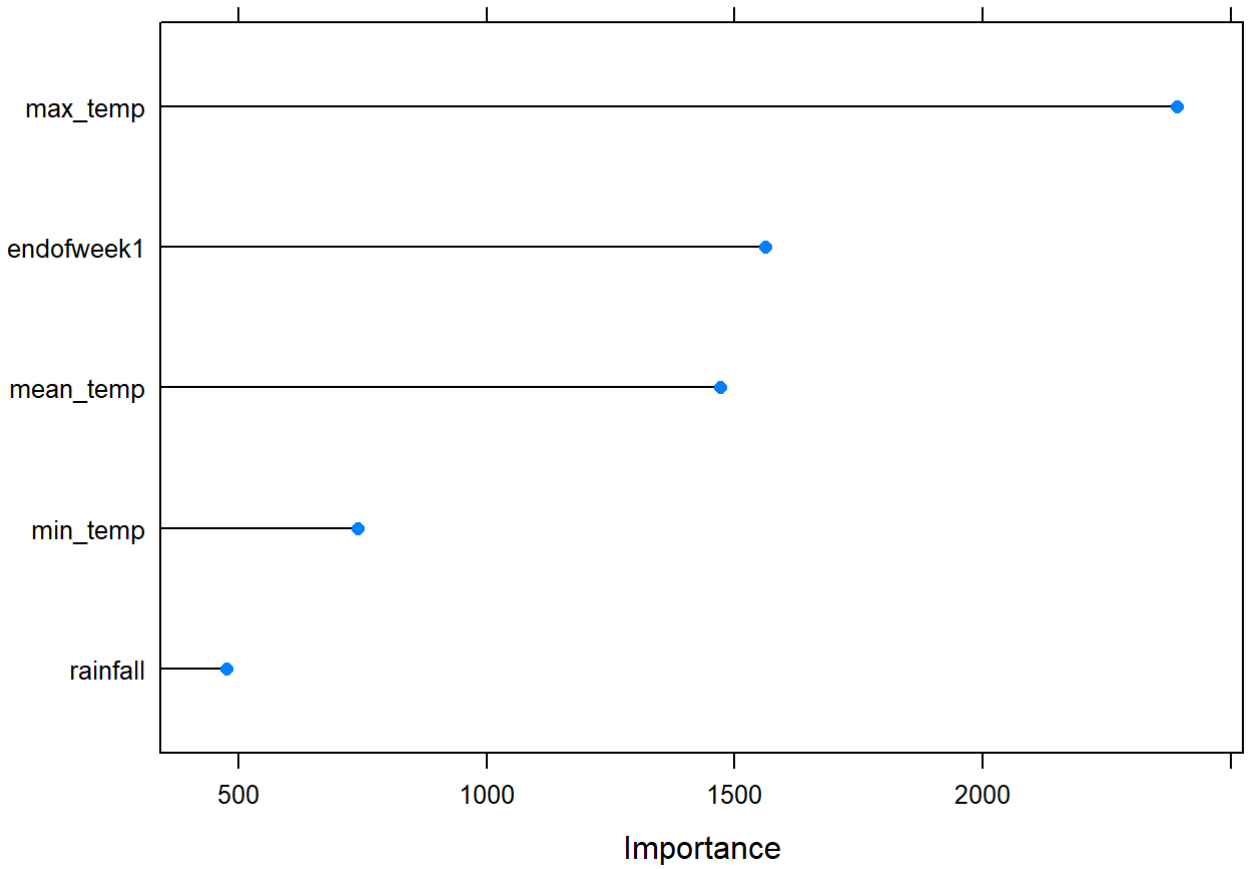


Finding important features using a random forest model

```
control <- trainControl(method="repeatedcv", number=5, repeats=3)

model <- train(beerconsumption~., data=beerdata%>%select(-c(date,dayofweek)), method="rf", preProcess="scale", trControl=con
trol)
importance <- varImp(model, scale=FALSE)

print(importance)
```

```
## rf variable importance
##
##             Overall
## max_temp    2391.1
## endofweek1  1560.8
## mean_temp   1470.4
## min_temp     740.6
## rainfall     476.0
```

```
plot(importance)
```



Choosing the temperature variables and end-of-week as the final features and split the data into test and train datasets

```
beerdata<-beerdata%>%select(max_temp,endofweek,min_temp,mean_temp,beerconsumption)

trainIndex <- createDataPartition(beerdata$beerconsumption, p = .8,
                                  list = FALSE,
                                  times = 1)

Train <- beerdata[ trainIndex,]
Test  <- beerdata[-trainIndex,]

head(Test)
```

```
## # A tibble: 6 x 5
##   max_temp endofweek min_temp mean_temp beerconsumption
##      <dbl> <chr>        <dbl>     <dbl>           <dbl>
## 1     28.6 1             21.5      24.0            29.8
## 2     33.7 0             19.5      24              29.7
## 3     35.4 0             21.4      26.0            25.7
## 4     34   0             21.3      26.0            31.8
## 5     26.1 0             19.2      21.7            25.8
## 6     30   1             18.1      24.4            31.1
```

Running 5 fold cross validation on five different models

- linear regression
- LASSO
- RIDGE
- Random Forest
- K neares neighbours

```
fitControl <- trainControl(method = "cv",   number = 5,
                           savePredictions = 'final',allowParallel = TRUE)

models<-caretList(beerconsumption ~., data = Train,methodList =  c("lm","rf","lasso","ridge","knn"),
             preProcess=c('center','scale'),
             trControl = fitControl)
```

```
## Warning in trControlCheck(x = trControl, y = target): indexes not defined in
## trControl. Attempting to set them ourselves, so each model in the ensemble will
## have the same resampling indexes.
```

```
models$knn
```

```
## k-Nearest Neighbors
##
## 293 samples
##   4 predictor
##
## Pre-processing: centered (4), scaled (4)
## Resampling: Cross-Validated (5 fold)
## Summary of sample sizes: 234, 235, 233, 234, 236
## Resampling results across tuning parameters:
##
##   k  RMSE      Rsquared   MAE
##   5  2.811196  0.6128913  2.317143
##   7  2.707750  0.6389631  2.236279
##   9  2.716018  0.6361594  2.239441
##
## RMSE was used to select the optimal model using the smallest value.
## The final value used for the model was k = 7.
```

```
models$ridge
```

```
## Ridge Regression
##
## 293 samples
##   4 predictor
##
## Pre-processing: centered (4), scaled (4)
## Resampling: Cross-Validated (5 fold)
## Summary of sample sizes: 234, 235, 233, 234, 236
## Resampling results across tuning parameters:
##
##   lambda  RMSE      Rsquared   MAE
##   0e+00   2.520458  0.6857104  2.096004
##   1e-04   2.520392  0.6857425  2.095922
##   1e-01   2.538009  0.6853815  2.104495
##
## RMSE was used to select the optimal model using the smallest value.
## The final value used for the model was lambda = 1e-04.
```

Linear regression and Ridge models provided the lowest root mean squared error overall

```
model_results <- data.frame(
 LM = mean(models$lm$results$RMSE),
 KNN = mean(models$knn$results$RMSE),
 RF = mean(models$rf$results$RMSE),
 LASSO = mean(models$lasso$results$RMSE),
 RIDGE = mean(models$ridge$results$RMSE)
 )
print(model_results)
```

```
##         LM      KNN       RF    LASSO    RIDGE
## 1 2.520458 2.744988 2.824832 3.139631 2.526286
```