CP003 - Modelos No Lineales

Mayra Goicochea Neyra

23/10/2019

Caso Bike Sharing

Introducción

Se solicita realizar un modelo predictivo sobre la asignación de las bicicletas para préstamo. Los sistemas de préstamo de bicicletas son la nueva generación de alquiler de bicicletas donde el proceso desde la suscripción, alquiler y retorno se ha vuelto automático.

El proceso de alquiler de bicicletas está altamente correlacionado con el entorno ambiental y estacional. El dataset Bike Sharing considera el registro histórico de dos años (2011 y 2012) del sistema Capital Bikeshare de Washington D.C., EE. UU., que está disponible públicamente en http://capitalbikeshare.com/system-data. Se agregó la información meteorológica y estacional correspondiente (proveniente de http://www.freemeteo.com)

```
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
```

```
library(readr)
library(kknn)
library(rsample)
```

```
## Loading required package: tidyr
```

```
library(tidyverse)
```

```
## -- Attaching packages ---------------------------------------------------
----------------- tidyverse 1.2.1 --
```

```
## v ggplot2 3.2.1     v purrr   0.3.2
## v tibble  2.1.3     v stringr 1.4.0
## v ggplot2 3.2.1     v forcats 0.4.0
```

```
## -- Conflicts -------------------------------------------------------
------------ tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()

library(ggplot2)
library(skimr)

##
## Attaching package: 'skimr'

## The following object is masked from 'package:stats':
##
##     filter

library(gam)

## Loading required package: splines

## Loading required package: foreach

##
## Attaching package: 'foreach'

## The following objects are masked from 'package:purrr':
##
##     accumulate, when

## Loaded gam 1.16.1

library(splines)
hour_rawdata <- read.csv("../Data/hour.csv")
day_rawdata <- read.csv("../Data/day.csv")
dim(day_rawdata)

## [1] 731  16

dim(hour_rawdata)

## [1] 17379    17
```

Se tiene 731 casos diarios con 16 variables, adicionalmente se tiene la información por hora, que está representada en 17379 casos. Las variables son las siguientes:

- instant: Índice de registro
- dteday: Fecha
- season: Estación del año (1: primavera, 2: verano, 3: otoño, 4: invierno)
- yr: Año (0: 2011, 1: 2012)
- mnth: Mes (1 a 12) * hr: Hora (0 a 23)

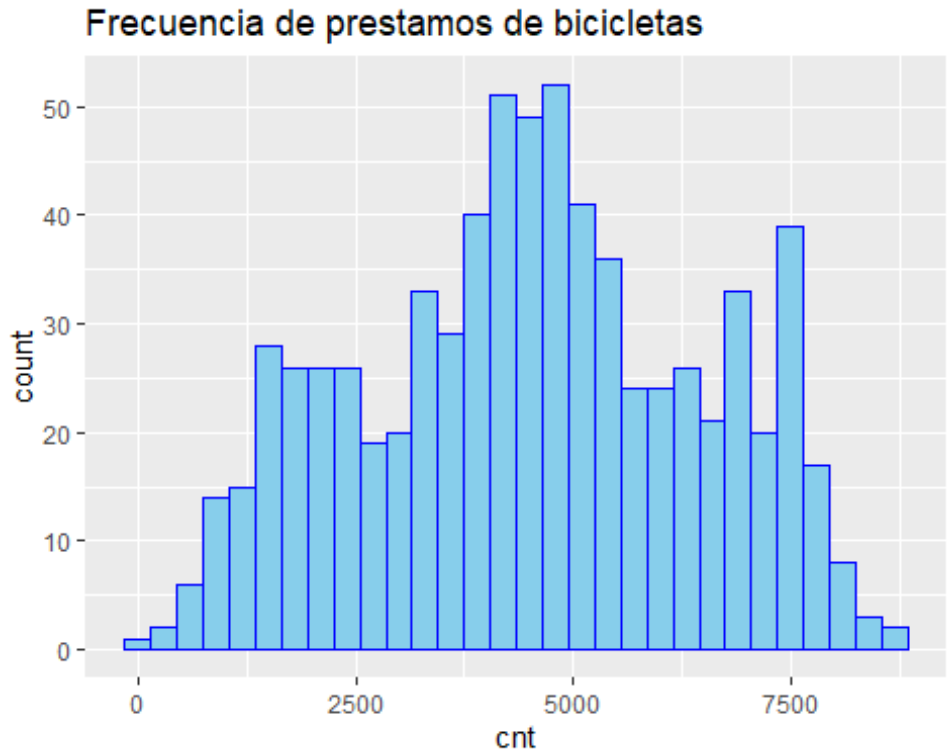
- holiday: si el día es feriado o no (extraído de http://dchr.dc.gov/page/holiday-schedule)
- weekday: Día de la semana
- workingday: Si el día no es fin de semana ni feriado es 1, de lo contrario es 0.
- weathersit: Clima expresado en 4 factores:
  - 1: despejado, pocas nubes, parcialmente nublado, parcialmente nublado
  - 2: Niebla + Nublado, Niebla + Nubes rotas, Niebla + Pocas nubes, Niebla
  - 3: nieve ligera, lluvia ligera + tormenta eléctrica + nubes dispersas, lluvia ligera + nubes dispersas
  - 4: lluvia intensa + paletas de hielo + tormenta eléctrica + niebla, nieve + niebla
- temp: temperatura normalizada en grados Celsius. Los valores se dividen en 41 (máx.)
- atemp: Sensación térmica normalizada en grados Celsius. Los valores se dividen en 50 (máx.)
- hum: Humedad normalizada. Los valores se dividen en 100 (máx.)
- windspeed: Velocidad del viento normalizada. Los valores se dividen en 67 (máx.)
- casual: Número de alquileres por usuarios ocasionales.
- registered: Número de alquileres por usuarios registrados.
- cnt: Número total de bicicletas alquiladas, incluidas las casuales y las registradas

EDA

La variable dependiente (o target) es CNT y las variables explicativas son: SEASON, HOLIDAY, WEEKDAY, WORKINGDAY, WEATHERSIT, TEMP, ATEMP. HUM y WINDSPEED.

```
#---------------------Data Wrangling-----------------------------
#Se quitan los campos instant y date
bday <- day_rawdata[,-c(1,2)]
bhour <- hour_rawdata[,-c(1,2)]
#Se asignan los factores correspondientes a las variables: yr, season, holida
y, weathersit
bday$yr <- factor(format(bday$yr, format = "%A"), levels = c("0", "1") , labe
ls = c("2011","2012"))
bhour$yr <- factor(format(bhour$yr, format = "%A"), levels = c("0", "1") , la
bels = c("2011","2012"))

bday$season <- factor(format(bday$season, format = "%A"),levels = c("1", "2",
"3","4") , labels = c("Spring","Summer","Fall","Winter"))
```

```r
bhour$season <- factor(format(bhour$season, format = "%A"),levels = c("1", "2
","3","4") , labels = c("Spring","Summer","Fall","Winter"))

bday$holiday <- factor(format(bday$holiday, format = "%A"),levels = c("0", "1
") , labels = c("Working Day","Holiday"))
bhour$holiday <- factor(format(bhour$holiday, format = "%A"),levels = c("0",
"1") , labels = c("Working Day","Holiday"))

bday$weathersit <- factor(format(bday$weathersit, format = "%A"),levels = c("
1", "2","3","4") ,labels = c("Good:Clear/Sunny","Moderate:Cloudy/Mist","Bad:
Rain/Snow/Fog","Worse: Heavy Rain/Snow/Fog"))
bhour$weathersit <- factor(format(bhour$weathersit, format = "%A"),levels = c
("1", "2","3","4") ,labels = c("Good:Clear/Sunny","Moderate:Cloudy/Mist","Bad
: Rain/Snow/Fog","Worse: Heavy Rain/Snow/Fog"))

ggplot(bday, aes(x = cnt)) +
  geom_histogram(colour = "blue", fill = 'skyblue', bins=30) +
  ggtitle("Frecuencia de prestamos de bicicletas")
```




```r
min(bday$cnt)
```

```
## [1] 22
```

```r
max(bday$cnt)
```

```
## [1] 8714
```

```r
mean(bday$cnt)
```

```
## [1] 4504.349
```

```r
median(bday$cnt)
```

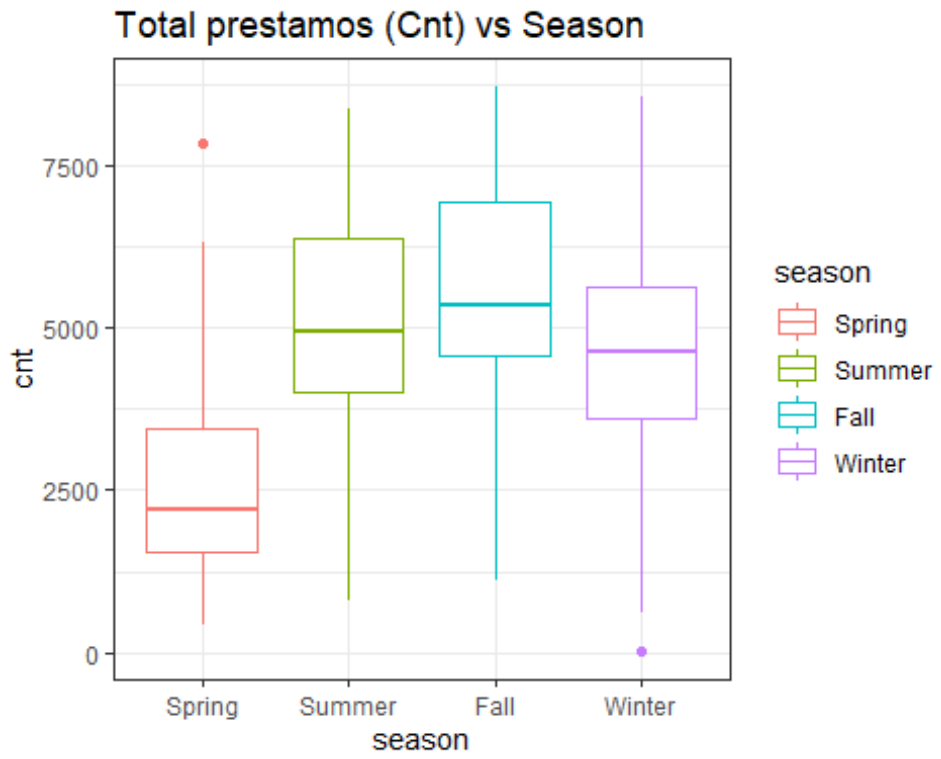
```
## [1] 4548
```

```r
sd(bday$cnt)
```

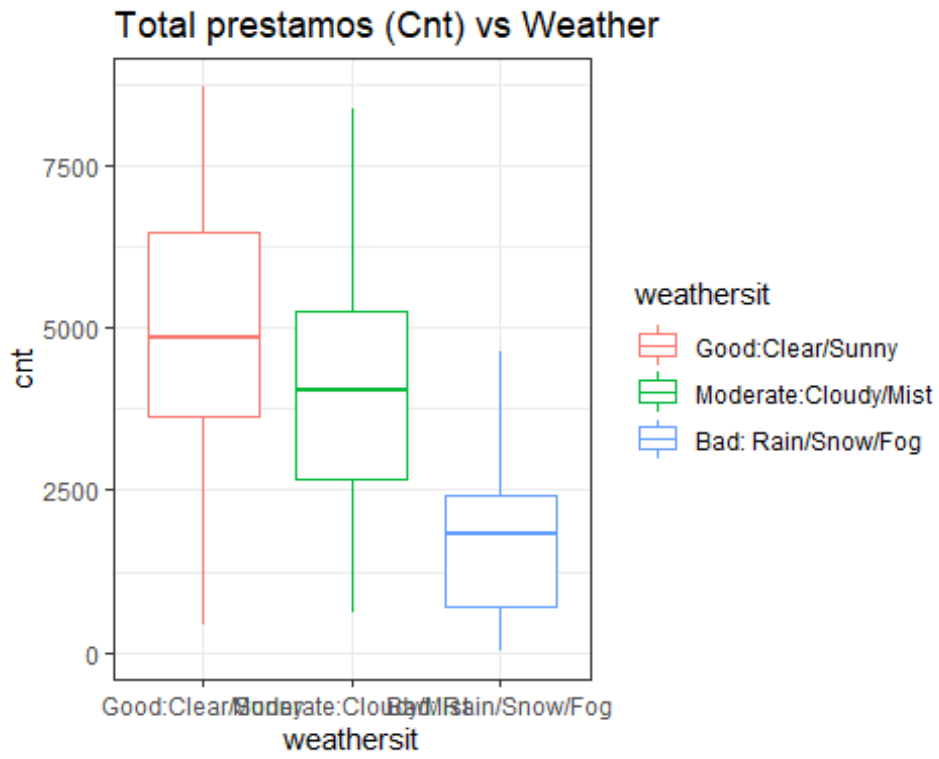
```
## [1] 1937.211
```

En cuanto a la frecuencia de la variable CNT, actúa en forma discreta (número enteros y positivos). Los valores mínimo y máximo son 22 y 8714 respectivamente, en cuanto a su valor medio es 4504.3 con una desviación típica de 1937.21. La mediana es de 1937.21.

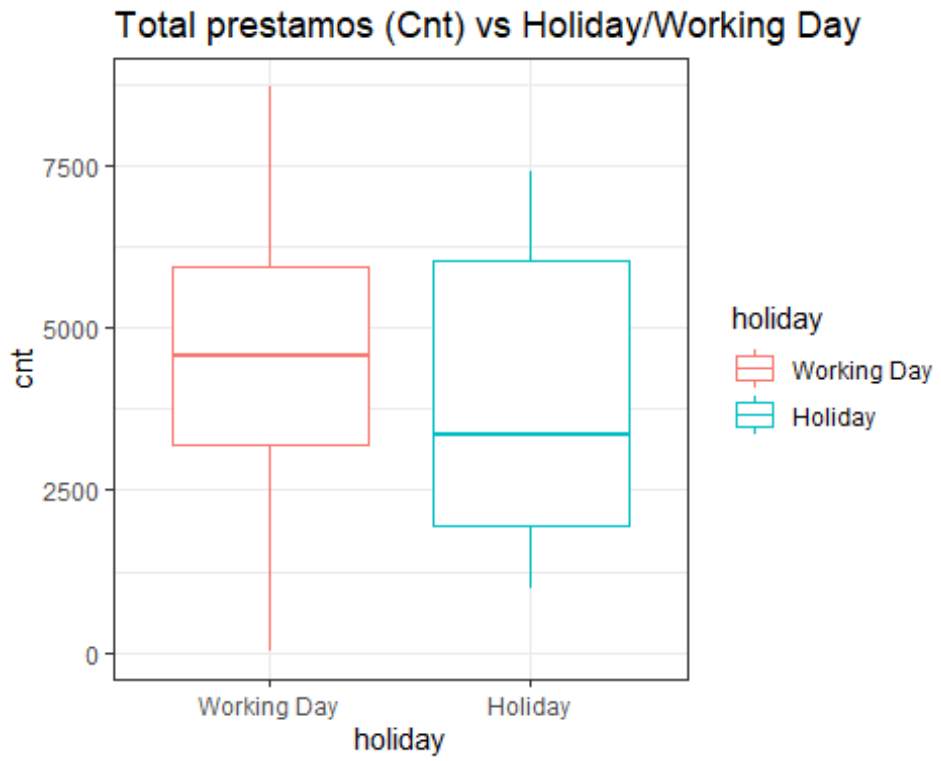
```r
ggplot(bday, aes(x = season, y = cnt)) +
  geom_boxplot( aes(colour=season)) +
  ggtitle("Total prestamos (Cnt) vs Season") +
  theme_bw()
```




```r
ggplot(bday, aes(x = weathersit, y = cnt)) +
  geom_boxplot( aes(colour=weathersit)) +
  ggtitle("Total prestamos (Cnt) vs Weather") +
  theme_bw()
```

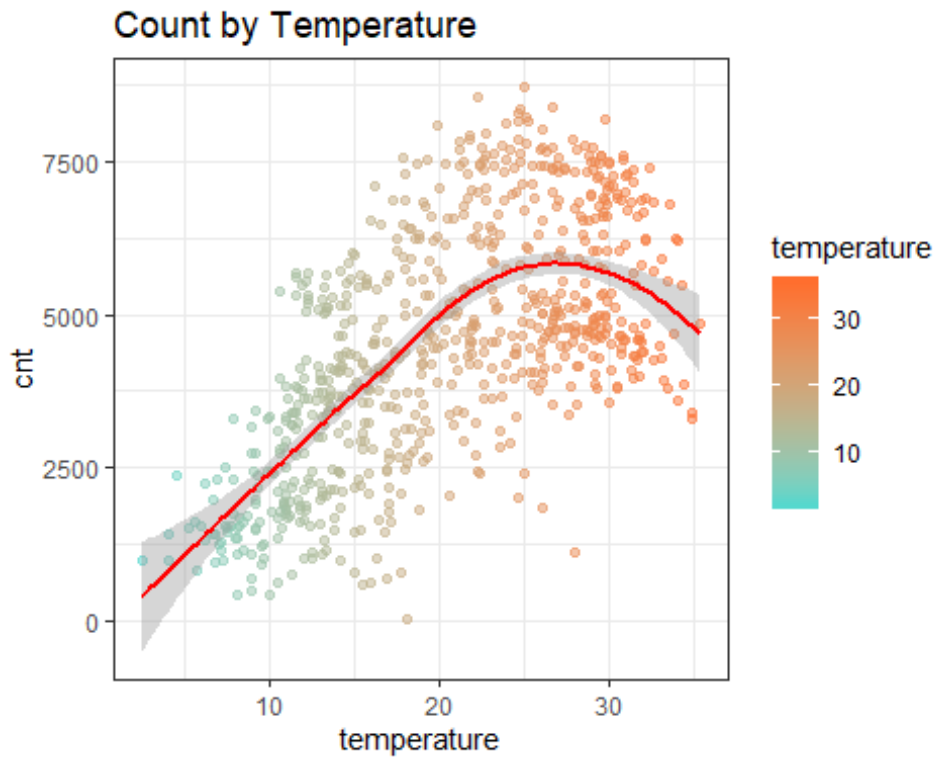



```
ggplot(bday, aes(x = holiday, y = cnt)) +
  geom_boxplot( aes(colour=holiday)) +
  ggtitle("Total prestamos (Cnt) vs Holiday/Working Day") +
  theme_bw()
```



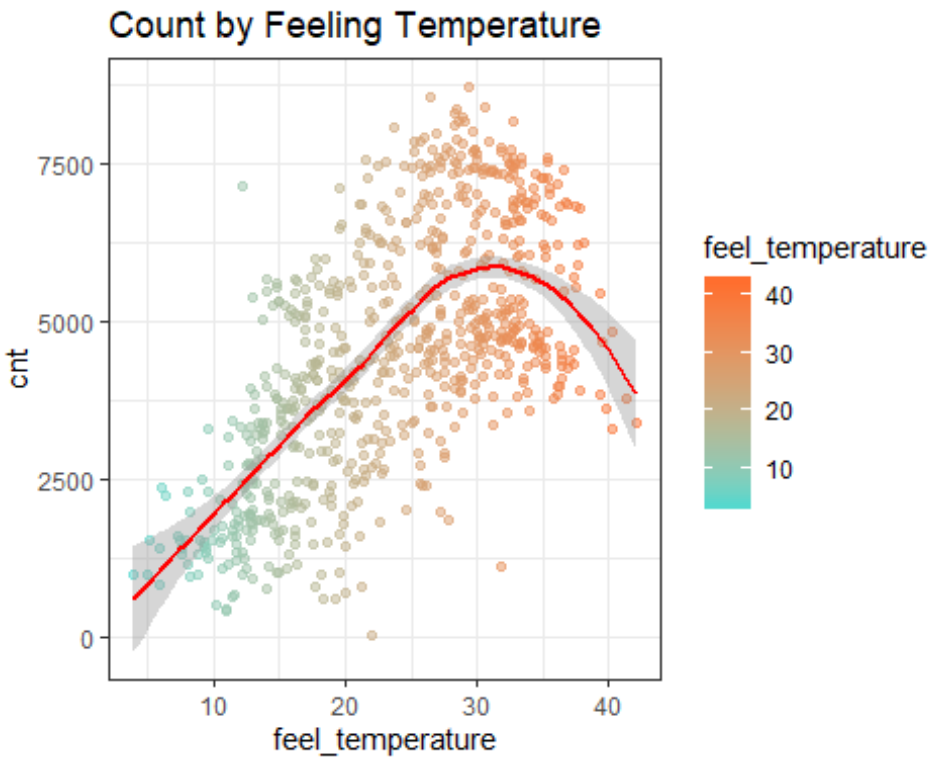
En cuanto a la relación con Season, el número promedio de alquiler de bicicletas es mayor en verano y otoño. En cuanto a primavera toma valores menores a las otras estaciones. Con respecto al clima, en un clima bueno y soleado se tiene el valor promedio mayor de alquiler. Sobre la variable Holiday, se muestra que el número medio de alquileres es ligeramente mayor en working day que en holiday; sin embargo, la muestra indica que hay más varianza en los días feriados y no se podría afirmar que tengan menores valores con respecto a los días laborales.

```
bday %>% mutate(temperature = temp*41) %>%
  ggplot( aes(temperature,cnt)) +
  geom_point(alpha = 0.5, aes(color = temperature)) +
  geom_smooth(colour = "red") +
  ggtitle("Count by Temperature") +
  scale_color_continuous(low = '#55D8CE',high = '#FF6E2E') +
  theme_bw()

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



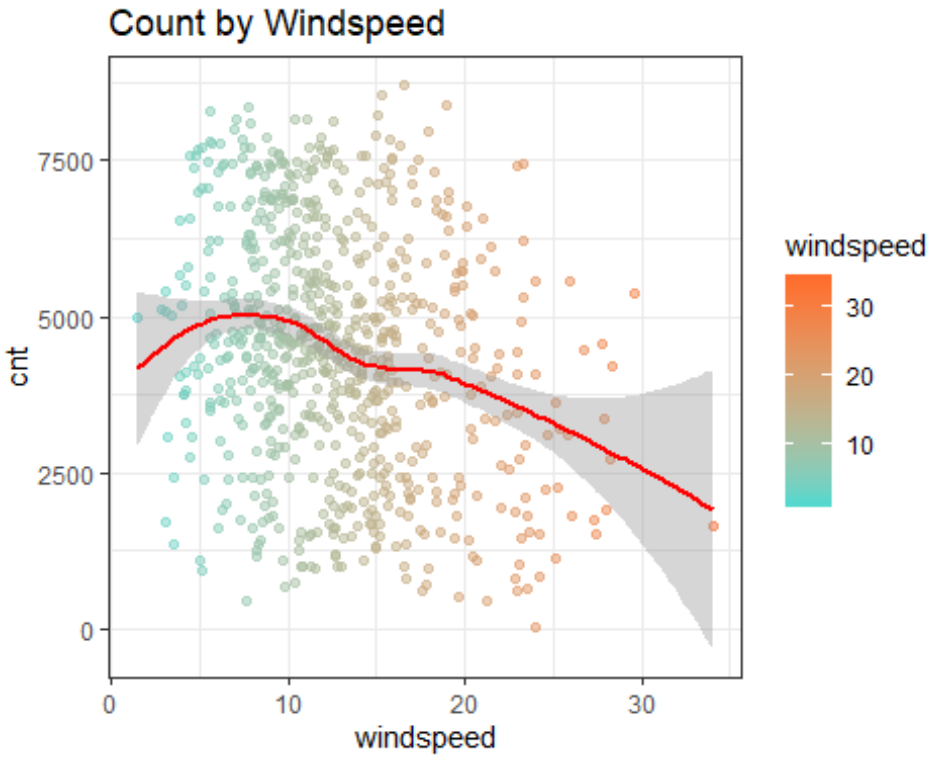

```
bday %>% mutate(feel_temperature = atemp*50) %>%
  ggplot( aes(feel_temperature,cnt)) +
  geom_point(alpha = 0.5, aes(color = feel_temperature)) +
  geom_smooth(colour ="red") +
  ggtitle("Count by Feeling Temperature") +
  scale_color_continuous(low = '#55D8CE',high = '#FF6E2E') +
  theme_bw()
```

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



Cuando se compara el número de alquiler con la temperatura, no se tiene una tendencia lineal, pero si se observa que los valores incrementan cuando la temperatura llega valores entre 20 y 25, de allí decrece a temperaturas altas. De similar manera actúa con respecto a la sensación térmica.

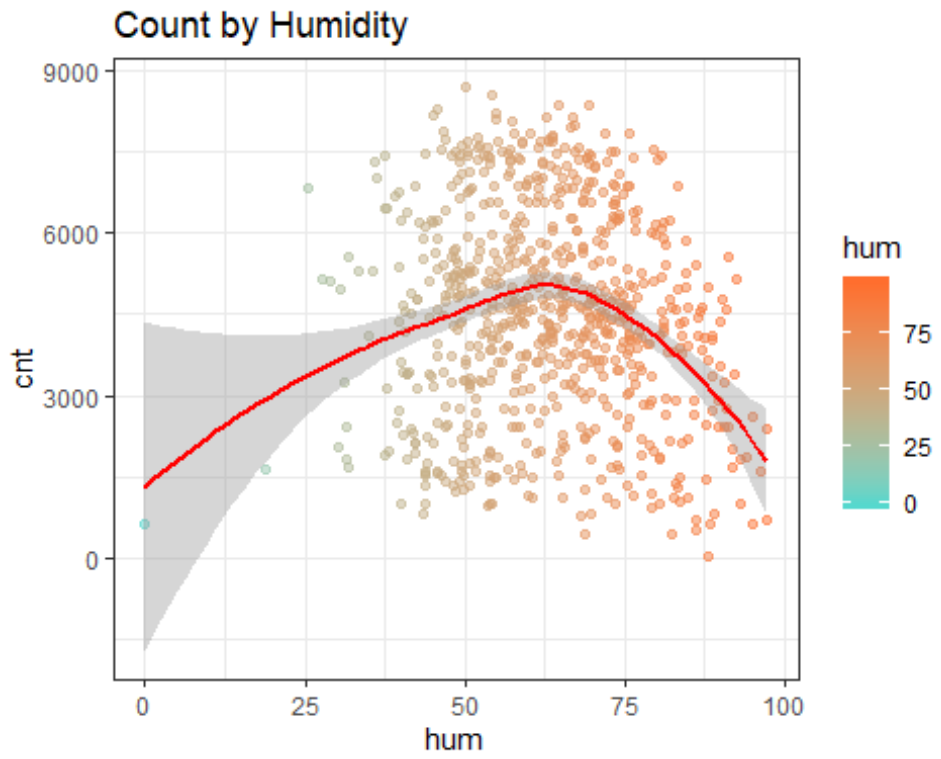
```
bday %>% mutate(windspeed = windspeed*67) %>%
  ggplot(aes(windspeed,cnt)) +
  geom_point(alpha = 0.5, aes(color = windspeed)) +
  geom_smooth(colour = "red") +
  ggtitle("Count by Windspeed") +
  scale_color_continuous(low = '#55D8CE',high = '#FF6E2E') +
  theme_bw()

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



Sobre la velocidad del viento, el número de alquiler crece ligeramente cuando se acerca a la velocidad 10 y luego decrece, esto sería debido a lo riesgoso que puede ser utilizar bicicleta cuando el viento sopla fuertemente.

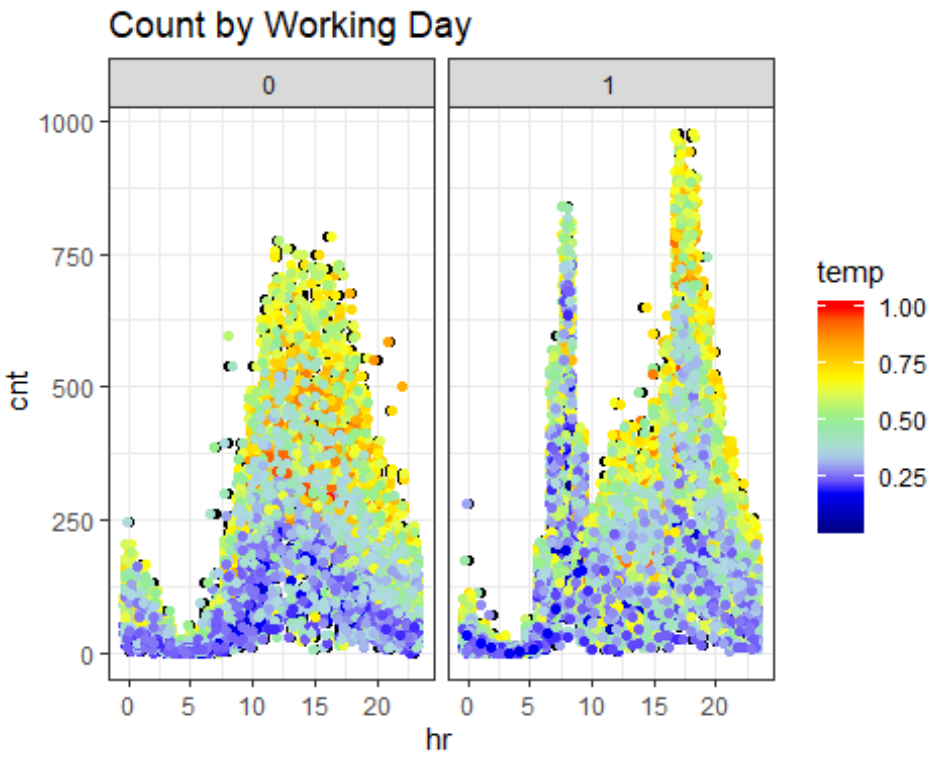
```
bday %>% mutate(hum = hum*100) %>%
  ggplot(aes(hum,cnt)) +
  geom_point(alpha = 0.5, aes(color = hum)) +
  geom_smooth(colour="red") +
  ggtitle("Count by Humidity") +
  scale_color_continuous(low = '#55D8CE',high = '#FF6E2E') +
  theme_bw()
```

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



Sobre la humedad, el valor más alto se tiene cuando se tiene humedad de 67.5% y de allí decrece.

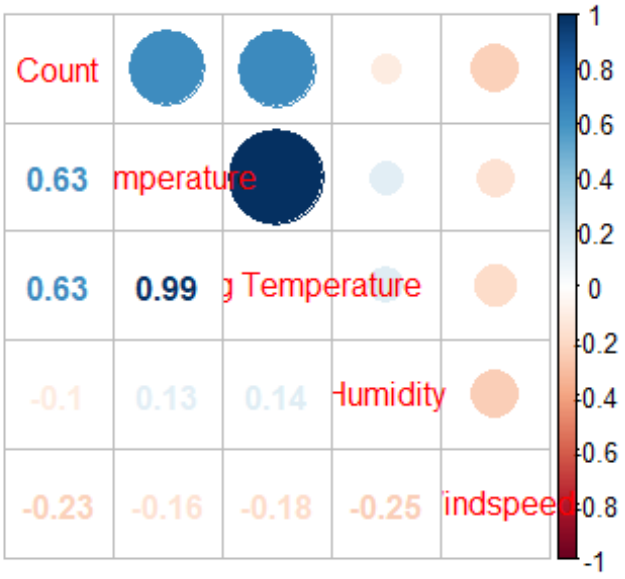
```
ggplot(bhour, aes(hr,cnt)) +
  geom_point() +
  ggtitle("Count by Working Day") +
  geom_point(aes(color = temp), position = "jitter") +
  scale_color_gradientn(colours = c('dark blue','blue','light blue','light gr
een','yellow','orange','red'))+
  theme_bw() +
  facet_grid(~workingday)
```



Se muestra en estos gráficos como en un día laborable, el alquiler de bicicletas es mayor en horas de 6 a 8 de la mañana y 17:30 a 20 horas. En cambio, en el caso de un día no laborable, el alquiler es mayor entre las 12 a 18 horas.

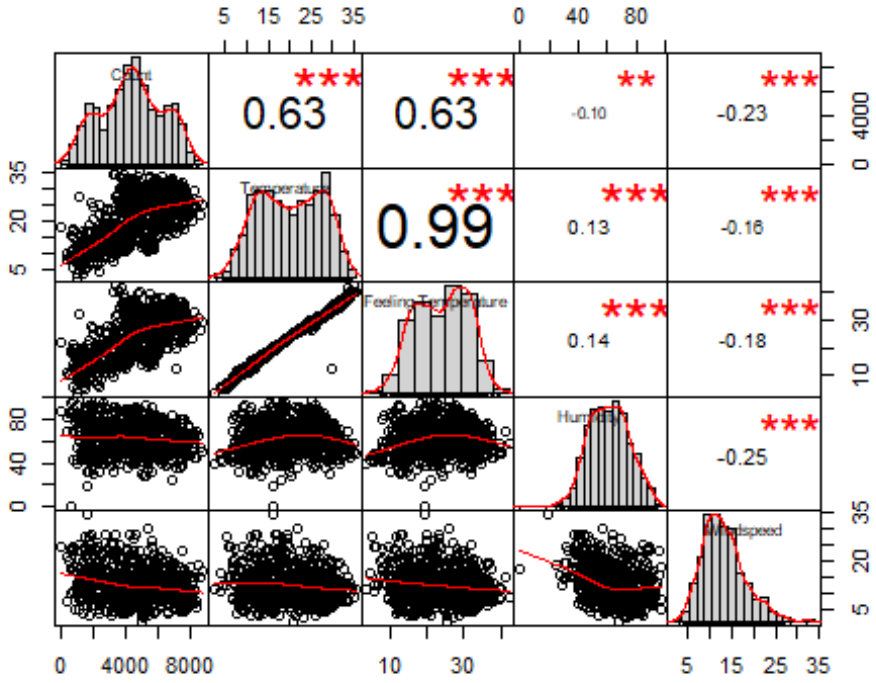
Matriz de Correlación

```
mday_cor <- bday %>% mutate(temp = 41*temp, atemp = atemp*50, hum = hum*100,
windspeed = windspeed*67 )
colnames(mday_cor) <- c("Season", "Year","Month", "Holiday","Weekday","Workin
g day","Weather","Temperature","Feeling Temperature","Humidity","Windspeed","
Casual","Registered","Count")
corrplot::corrplot.mixed(cor(mday_cor[,c("Count","Temperature","Feeling Tempe
rature","Humidity","Windspeed")]))
```




```
PerformanceAnalytics::chart.Correlation(mday_cor[,c("Count","Temperature","Feeling Temperature","Humidity","Windspeed")],histogram = TRUE, pch = 19)
```

```
## Registered S3 method overwritten by 'xts':
##   method     from
##   as.zoo.xts zoo
```



Mediante la matriz se observa que las variables Temperature y Feeling Temperature están altamente correlacionadas con el número de alquiler.

Modelos No lineales

Se formula el modelo inicial: cnt ~ s(temp,df=9.1)+s(atemp,df=8.8)+s(hum,df=4.55)+s(windspeed,df=6.01)+season+holiday+weekday+workingday+weathersit

```
gam1 <- gam(cnt ~ s(temp,df=9.1)+s(atemp,df=8.8)+s(hum,df=4.55)+s(windspeed,df=6.01)+season+holiday+weekday+workingday+weathersit, data=bday)

## Warning in model.matrix.default(mt, mf, contrasts): non-list contrasts
## argument ignored

summary(gam1)

##
## Call: gam(formula = cnt ~ s(temp, df = 9.1) + s(atemp, df = 8.8) +
##     s(hum, df = 4.55) + s(windspeed, df = 6.01) + season + holiday +
##     weekday + workingday + weathersit, data = bday)
## Deviance Residuals:
##     Min      1Q  Median      3Q     Max
## -3146.2  -918.2  -114.5  1000.8  3047.3
##
## (Dispersion Parameter for gaussian family taken to be 1340479)
##
```

```
##     Null Deviance: 2739535392 on 730 degrees of freedom
## Residual Deviance: 929675441 on 693.5395 degrees of freedom
## AIC: 12426.3
##
## Number of Local Scoring Iterations: 17
##
## Anova for Parametric Effects
##                             Df    Sum Sq   Mean Sq  F value    Pr(>F)
## s(temp, df = 9.1)         1.00 967006322 967006322 721.3884 < 2.2e-16 ***
## s(atemp, df = 8.8)        1.00   2144697   2144697   1.5999  0.206335
## s(hum, df = 4.55)         1.00 175162910 175162910 130.6718 < 2.2e-16 ***
## s(windspeed, df = 6.01)   1.00 132448657 132448657  98.8069 < 2.2e-16 ***
## season                    3.00  93088075  31029358  23.1480 2.736e-14 ***
## holiday                   1.00   5955530   5955530   4.4428  0.035406 *
## weekday                   1.00  11768299  11768299   8.7792  0.003151 **
## workingday                1.00     96336     96336   0.0719  0.788718
## weathersit                2.00  29726666  14863333  11.0881 1.819e-05 ***
## Residuals               693.54 929675441   1340479
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Anova for Nonparametric Effects
##                           Npar Df Npar F     Pr(F)
## (Intercept)
## s(temp, df = 9.1)             8.1 61.451 < 2.2e-16 ***
## s(atemp, df = 8.8)            7.8 22.461 < 2.2e-16 ***
## s(hum, df = 4.55)             3.5  5.881 0.0002389 ***
## s(windspeed, df = 6.01)       5.0  0.907 0.4760823
## season
## holiday
## weekday
## workingday
## weathersit
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

El p-value obtenido para la función del predictor windspeed (0.476) no muestra evidencias de que la relación entre cnt y windspeed no sea lineal, es así que deja la posibilidad de crear un modelo con menor complejidad con una relación lineal. El análisis ANOVA dará respuesta de que modelo es más conveniente. Para ello, se formuló 5 modelos de menor a mayor complejidad.

```
m_1 <- gam(cnt ~ s(temp,df=9.1)+s(atemp,df=8.8)+s(hum,df=4.55)+season+holiday
+weekday+workingday+weathersit, data=bday)

## Warning in model.matrix.default(mt, mf, contrasts): non-list contrasts
## argument ignored

m_2 <- gam(cnt ~ s(temp,df=9.1)+s(atemp,df=8.8)+s(hum,df=4.55)+windspeed+seas
on+holiday+weekday+workingday+weathersit, data=bday)
```

```
## Warning in model.matrix.default(mt, mf, contrasts): non-list contrasts
## argument ignored
```

```
m_3 <- gam(cnt ~ s(temp,df=9.1)+s(atemp,df=8.8)+s(hum,df=4.55)+s(windspeed,df
=6.01)+season+holiday+weekday+workingday+weathersit, data=bday)
```

```
## Warning in model.matrix.default(mt, mf, contrasts): non-list contrasts
## argument ignored
```

```
m_4 <- gam(cnt ~ s(temp,df=9.1)+s(atemp,df=8.8)+s(hum,df=4.55)+season+holiday
+weekday+workingday, data=bday)
```

```
## Warning in model.matrix.default(mt, mf, contrasts): non-list contrasts
## argument ignored
```

```
m_5 <- gam(cnt ~ s(atemp,df=8.8)+windspeed+season+holiday+weekday+weathersit,
data=bday)
```

```
## Warning in model.matrix.default(mt, mf, contrasts): non-list contrasts
## argument ignored
```

```
anova(m_1,m_2,m_3,m_4,m_5,test='F')
```

```
## Analysis of Deviance Table
##
## Model 1: cnt ~ s(temp, df = 9.1) + s(atemp, df = 8.8) + s(hum, df = 4.55)
+
##     season + holiday + weekday + workingday + weathersit
## Model 2: cnt ~ s(temp, df = 9.1) + s(atemp, df = 8.8) + s(hum, df = 4.55)
+
##     windspeed + season + holiday + weekday + workingday + weathersit
## Model 3: cnt ~ s(temp, df = 9.1) + s(atemp, df = 8.8) + s(hum, df = 4.55)
+
##     s(windspeed, df = 6.01) + season + holiday + weekday + workingday +
##     weathersit
## Model 4: cnt ~ s(temp, df = 9.1) + s(atemp, df = 8.8) + s(hum, df = 4.55)
+
##     season + holiday + weekday + workingday
## Model 5: cnt ~ s(atemp, df = 8.8) + windspeed + season + holiday + weekday
+
##     weathersit
##   Resid. Df Resid. Dev       Df  Deviance       F    Pr(>F)
## 1    699.55  996412038
## 2    698.55  935273676   1.0000  61138363 45.6093 3.056e-11 ***
## 3    693.54  929675441   5.0102   5598235  0.8336    0.5262
## 4    701.55 1024743302  -8.0102 -95067861  8.8538 1.352e-11 ***
## 5    713.20 1095802042 -11.6498 -71058740  4.5503 5.897e-07 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

El segundo modelo resulta ser el mejor, con las variables windspeed, season, holiday, weekday, workingday, weathersit con relación lineal y ajustando las variables temperature, feeling temperature y humidity a splines.