

# LABORATORIO –JOSE RICARDO MERIDA

```
library(dplyr)
```

```
## Warning: package 'dplyr' was built under R version 4.2.3
```

```
##
```

```
## 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(ggplot2)
```

```
## Warning: package 'ggplot2' was built under R version 4.2.3
```

```
library(nycflights13)
```

```
## Warning: package 'nycflights13' was built under R version 4.2.3
```

```
mis_datos <- read.csv("dataset.csv")
```

```
data_frame_datos <- data.frame(mis_datos)
```

```
#print(data_frame_datos)
```

```
library(dplyr)
```

## ————— PREGUNTA 1 —————

Cree un conjunto de columnas nuevas: día, mes, año, hora y minutos a partir de la columna datetime, para esto investigue como puede “desarmar” la variable datetime utilizando lubridate y mutate.

## ————— RESPUESTA 1 —————

no existe la columna datetime

```
grupos14<-tidyr::separate(data_frame_datos, dteday, c("y", "m", "d"))
```

```
str(grupos14)
```

```
## 'data.frame':   17379 obs. of  19 variables:
```

```
## $ instant   : int  1 2 3 4 5 6 7 8 9 10 ...
```

```
## $ y         : chr  "2011" "2011" "2011" "2011" ...
```

```
## $ m         : chr  "01" "01" "01" "01" ...
```

```
## $ d      : chr "01" "01" "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 ...
```

## — PREGUNTA 2 —

¿Qué mes es el que tiene la mayor demanda? Muestre una tabla y una gráfica

## — RESPUESTA 2 —

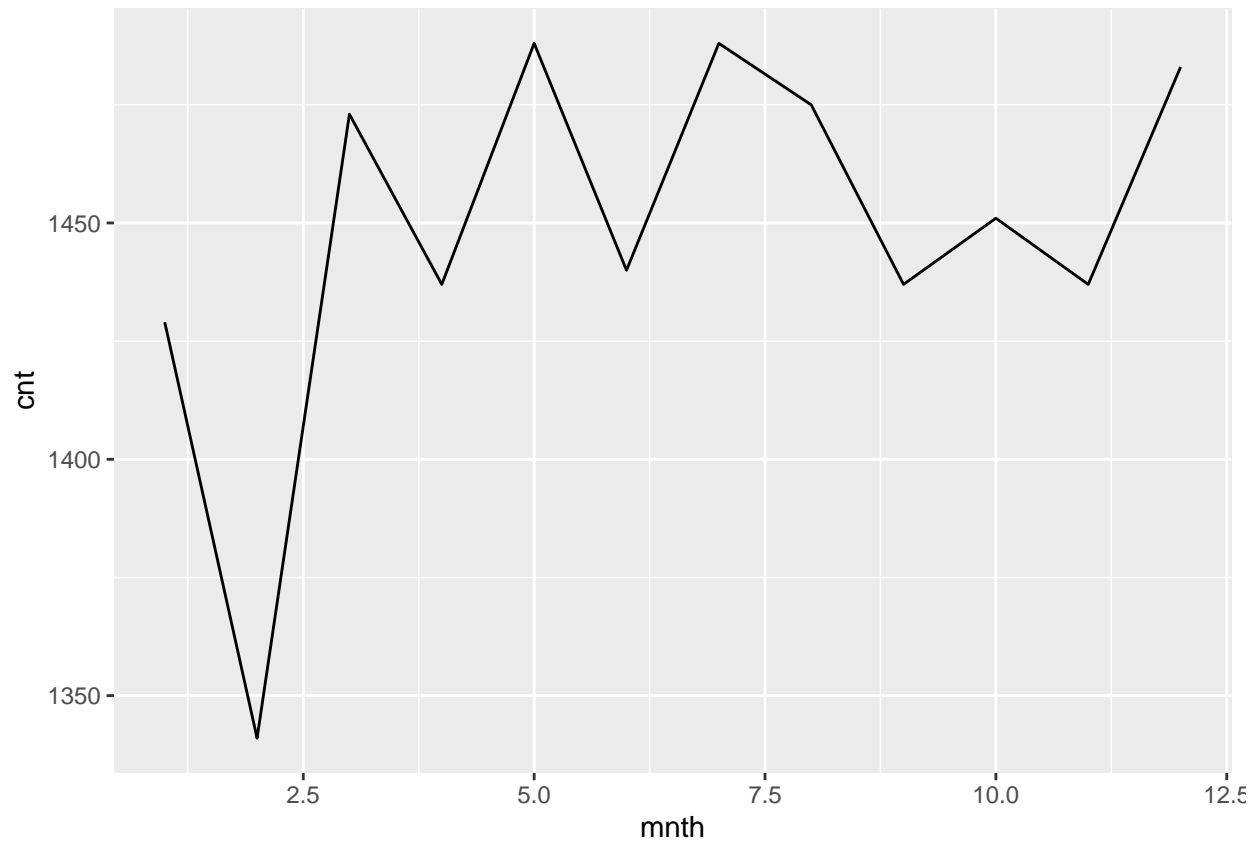
```
grupos <- data_frame_datos %>%
  group_by(mnth)%>%
  summarise(cnt = n())

print(grupos)
```

```
## # A tibble: 12 x 2
##   mnth    cnt
##   <int> <int>
## 1     1  1429
## 2     2  1341
## 3     3  1473
## 4     4  1437
## 5     5  1488
## 6     6  1440
## 7     7  1488
## 8     8  1475
## 9     9  1437
## 10    10  1451
## 11    11  1437
## 12    12  1483
```

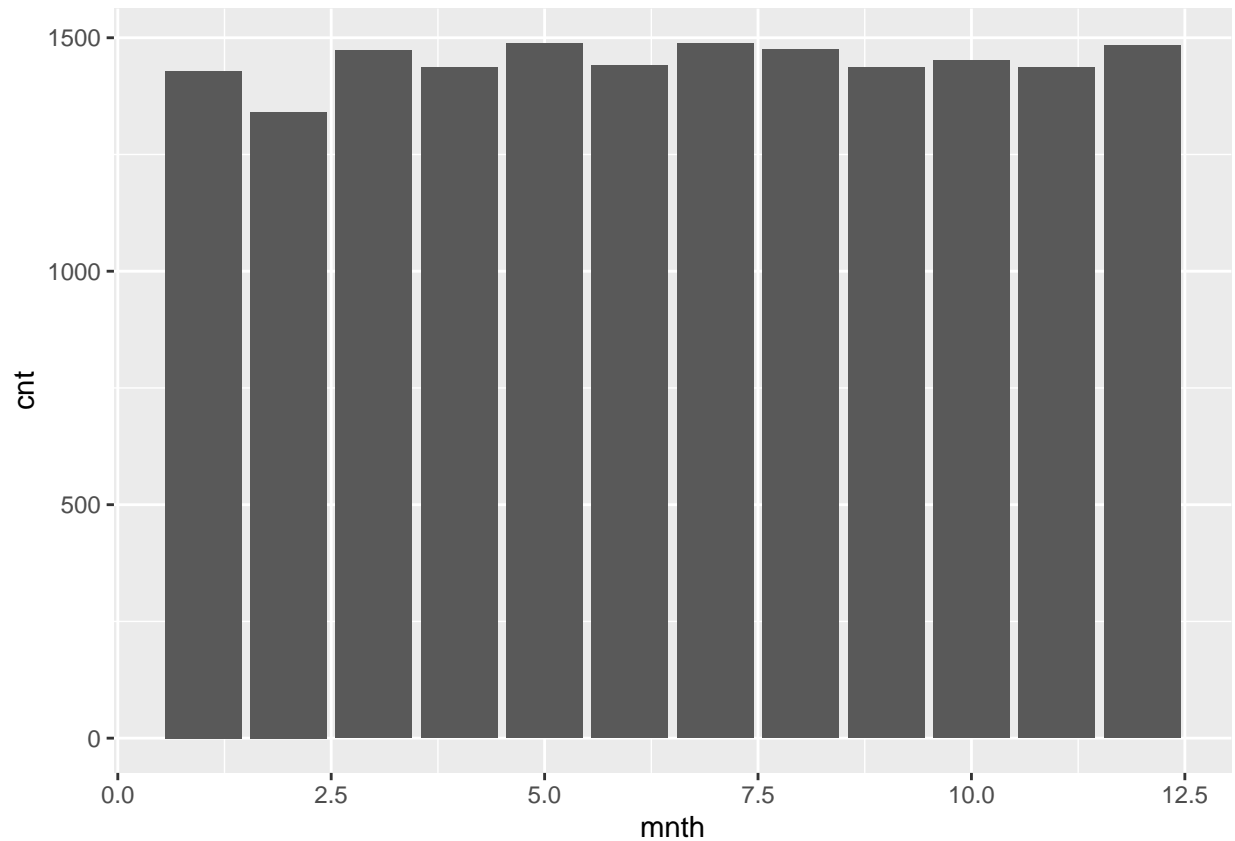
```
qplot(x=mnth,y=cnt,data=grupos,geom='line')
```

```
## Warning: `qplot()` was deprecated in ggplot2 3.4.0.
## This warning is displayed once every 8 hours.
## Call `lifecycle::last_lifecycle_warnings()` to see where this warning was
## generated.
```



```
p2<-ggplot(data=grupos, aes(x=mnt, y=cnt)) +  
  geom_bar(stat="identity")
```

p2



```
top1 <- grupos %>%
  top_n(1,cnt)

print("Los meses que tienen mayor demanda son")

## [1] "Los meses que tienen mayor demanda son"
print(top1)

## # A tibble: 2 x 2
##   mnth  cnt
##   <int> <int>
## 1     5 1488
## 2     7 1488
```

### ———— PREGUNTA 3 ————

¿Qué rango de hora es la de mayor demanda? Muestre una tabla y una gráfica

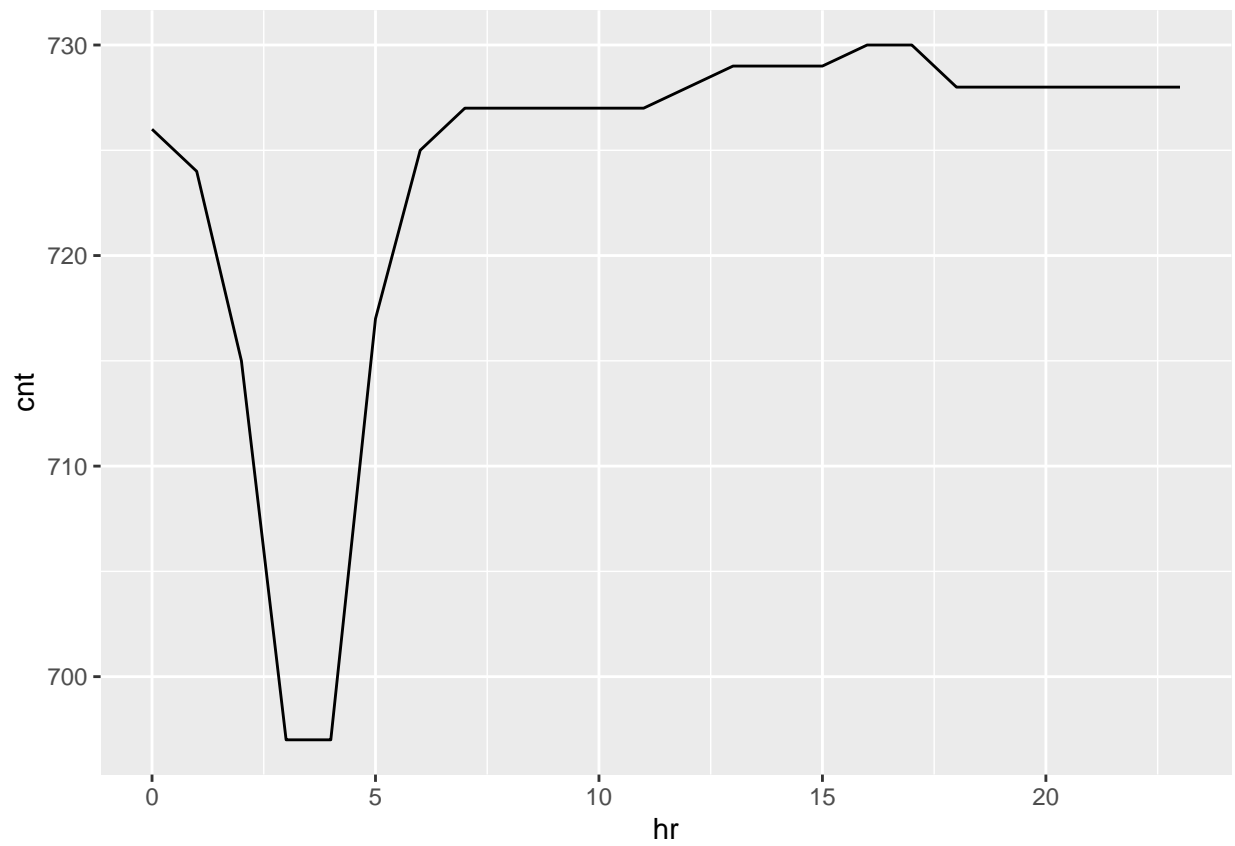
### ———— RESPUESTA 3 ————

```
grupos3 <- data_frame_datos %>%
  group_by(hr)%>%
  summarise(cnt = n())

print(grupos3)
```

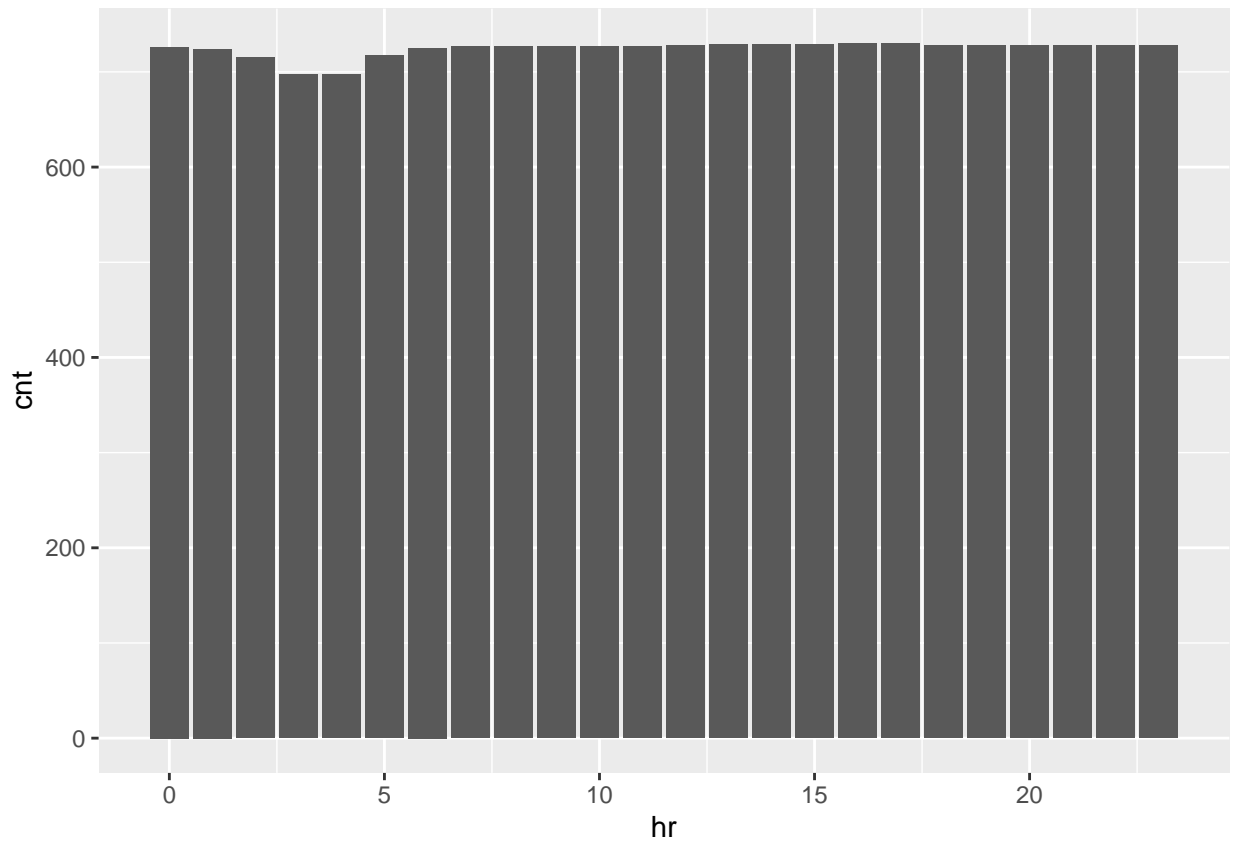
```
## # A tibble: 24 x 2
##       hr    cnt
##   <int> <int>
## 1     0   726
## 2     1   724
## 3     2   715
## 4     3   697
## 5     4   697
## 6     5   717
## 7     6   725
## 8     7   727
## 9     8   727
## 10    9   727
## # i 14 more rows
```

```
qplot(x=hr,y=cnt,data=grupos3,geom='line')
```



```
p3<-ggplot(data=grupos3, aes(x=hr, y=cnt)) +
  geom_bar(stat="identity")
```

```
p3
```



```
top3 <- grupos3 %>%
  top_n(1,cnt)

print(top3)
```

```
## # A tibble: 2 x 2
##   hr    cnt
##   <int> <int>
## 1    16  730
## 2    17  730
```

#### ———— PREGUNTA 4 ————

¿Qué temporada es la mas alta? Muestre una tabla.

#### ———— RESPUESTA 4 ————

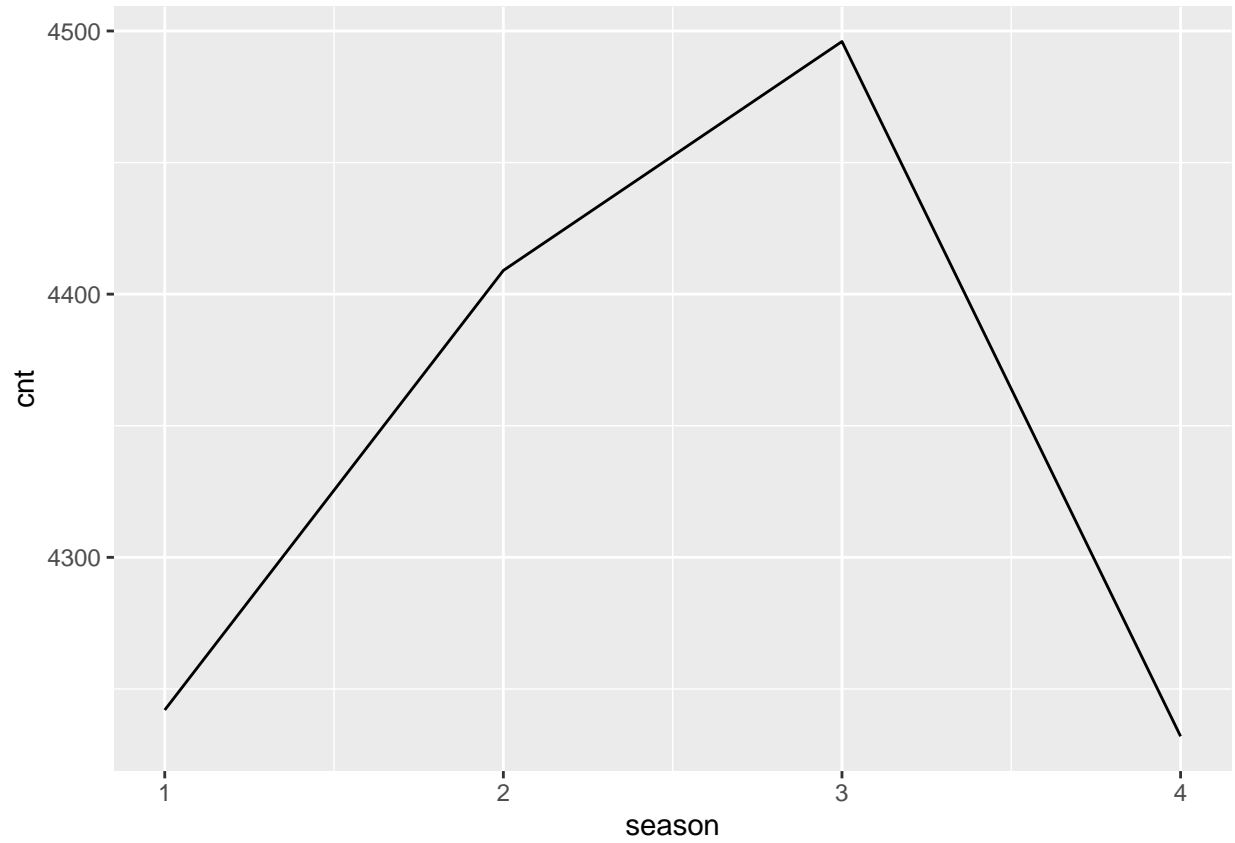
```
grupos4 <- data_frame_datos %>%
  group_by(season)%>%
  summarise(cnt = n())

print(grupos4)
```

```
## # A tibble: 4 x 2
##   season    cnt
##   <int> <int>
## 1      1  4242
```

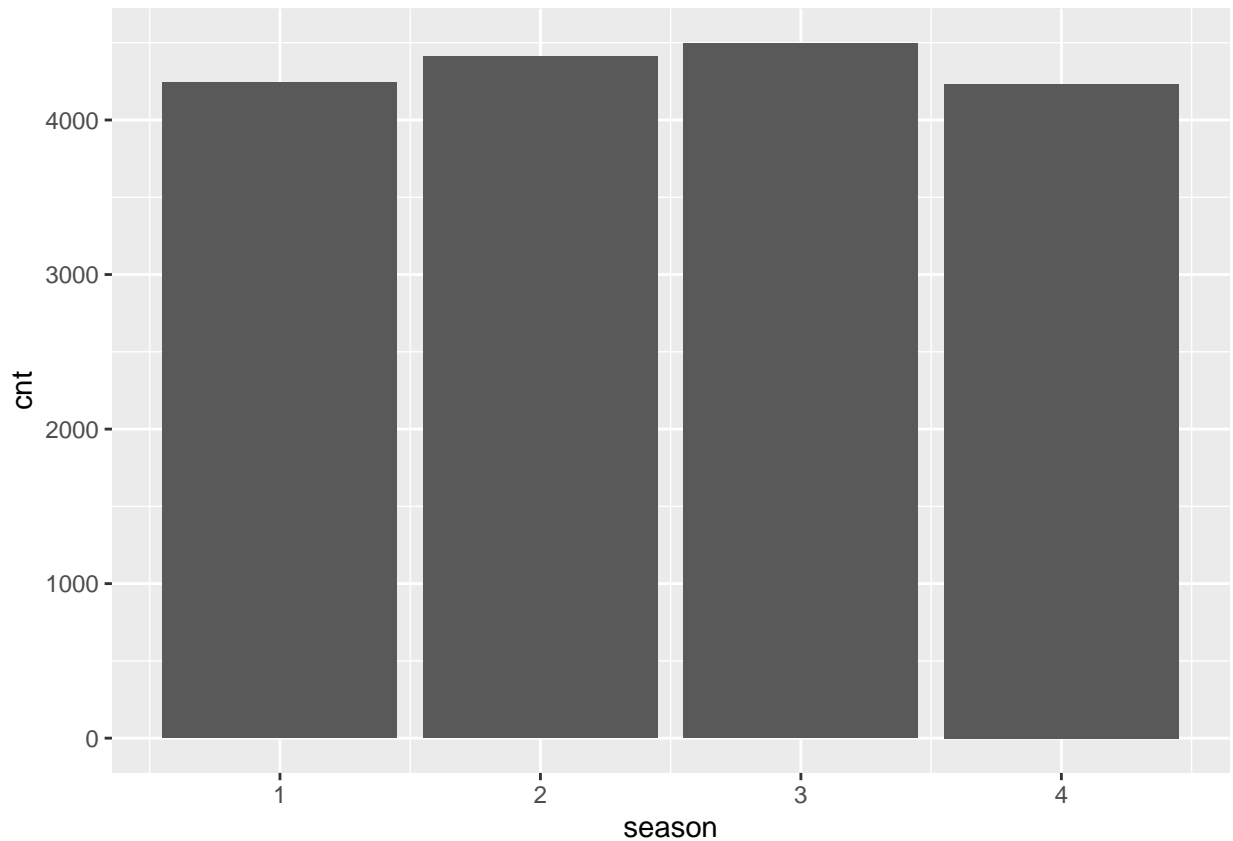
```
## 2      2  4409
## 3      3  4496
## 4      4  4232
```

```
qplot(x=season,y=cnt,data=grupos4,geom='line')
```



```
p4<-ggplot(data=grupos4, aes(x=season, y=cnt)) +  
  geom_bar(stat="identity")
```

```
p4
```



```
top4 <- grupos4 %>%
  top_n(1,cnt)

print(top4)
```

```
## # A tibble: 1 x 2
##   season  cnt
##   <int> <int>
## 1      3 4496
```

### ———— PREGUNTA 5 ————

¿A que temperatura disminuye la demanda? Muestre una gráfica para analizar y dar su respuesta.

### ———— RESPUESTA 5 ————

```
grupos5 <- data_frame_datos %>%
  group_by(temp)%>%
  summarise(cnt = n())

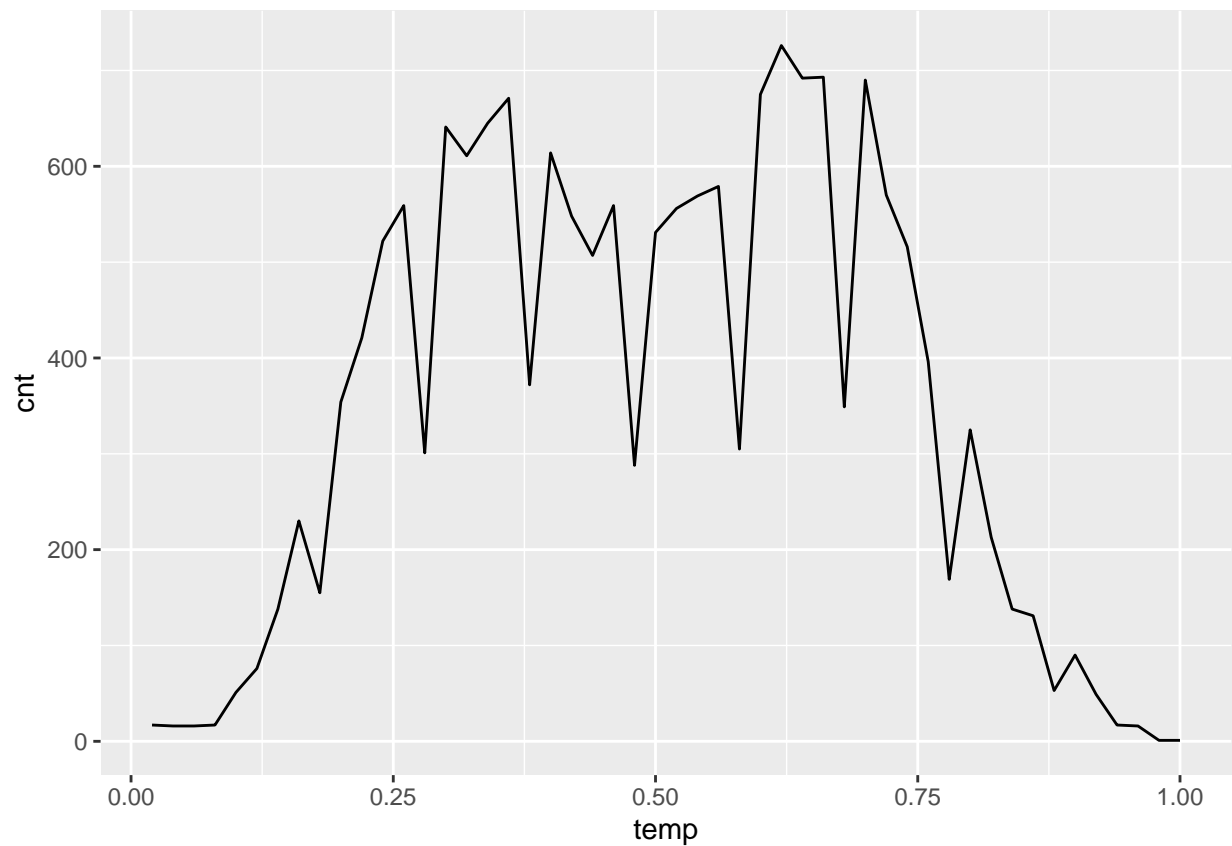
print(grupos5)
```

```
## # A tibble: 50 x 2
##   temp  cnt
##   <dbl> <int>
```



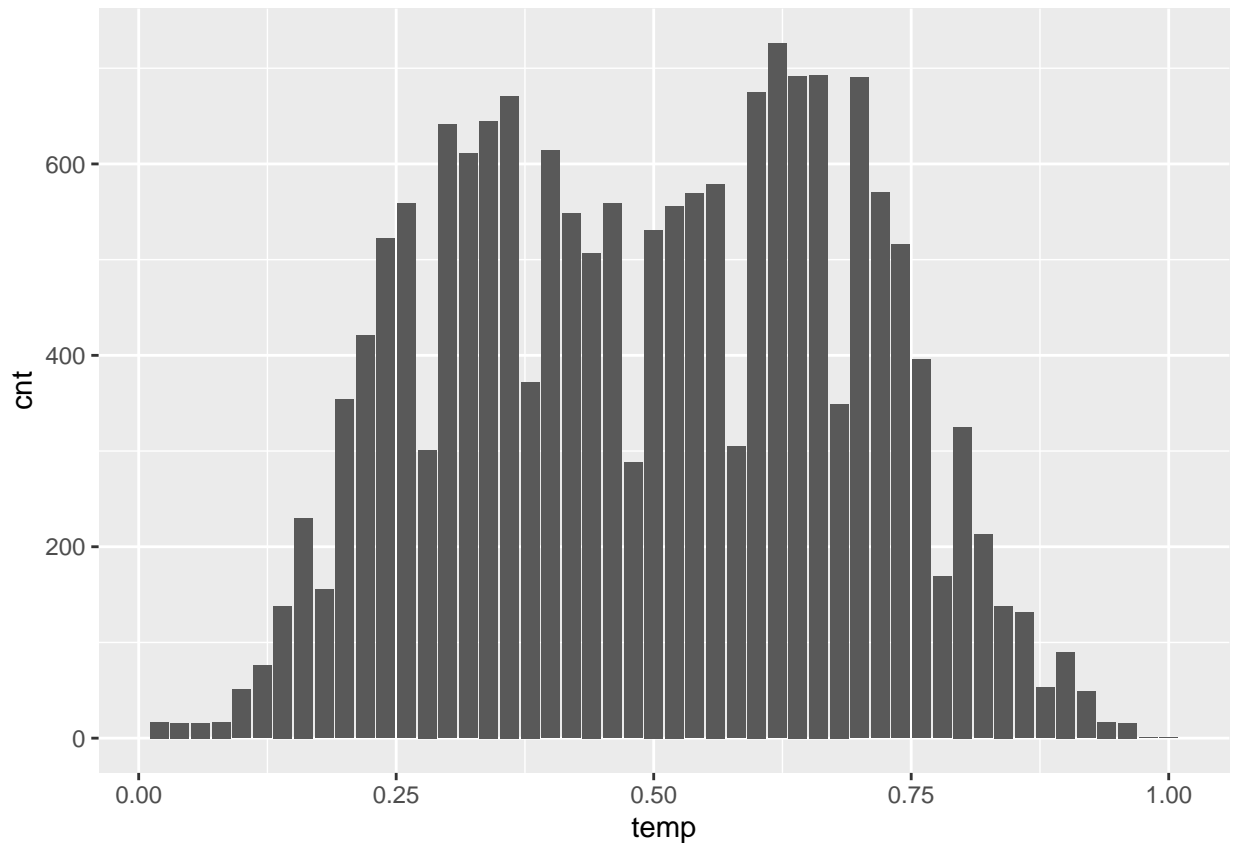
```
## 1 0.02 17
## 2 0.04 16
## 3 0.06 16
## 4 0.08 17
## 5 0.1 51
## 6 0.12 76
## 7 0.14 138
## 8 0.16 230
## 9 0.18 155
## 10 0.2 354
## # i 40 more rows
```

```
qplot(x=temp,y=cnt,data=grupos5,geom='line')
```



```
p5<-ggplot(data=grupos5, aes(x=temp, y=cnt)) +
  geom_bar(stat="identity")
```

```
p5
```



```
top5 <- grupos5 %>%
  top_n(1,-cnt)

print(top5)
```

```
## # A tibble: 2 x 2
##   temp cnt
##   <dbl> <int>
## 1  0.98     1
## 2    1     1
```

### PREGUNTA 6

¿A que humedad disminuye la demanda? Muestre una gráfica para analizar y dar su respuesta.

### RESPUESTA 6

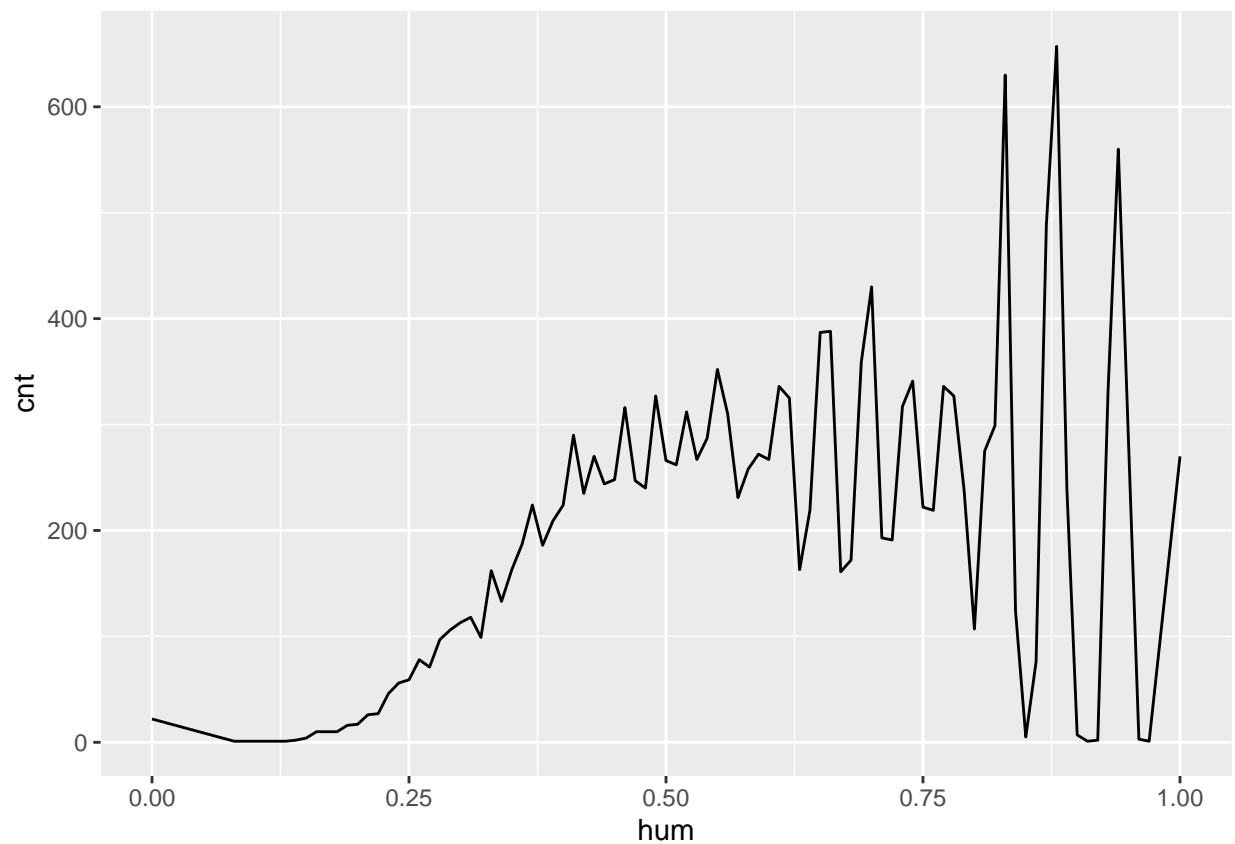
```
grupos6 <- data_frame_datos %>%
  group_by(hum)%>%
  summarise(cnt = n())

print(grupos6)
```

```
## # A tibble: 89 x 2
##   hum cnt
```

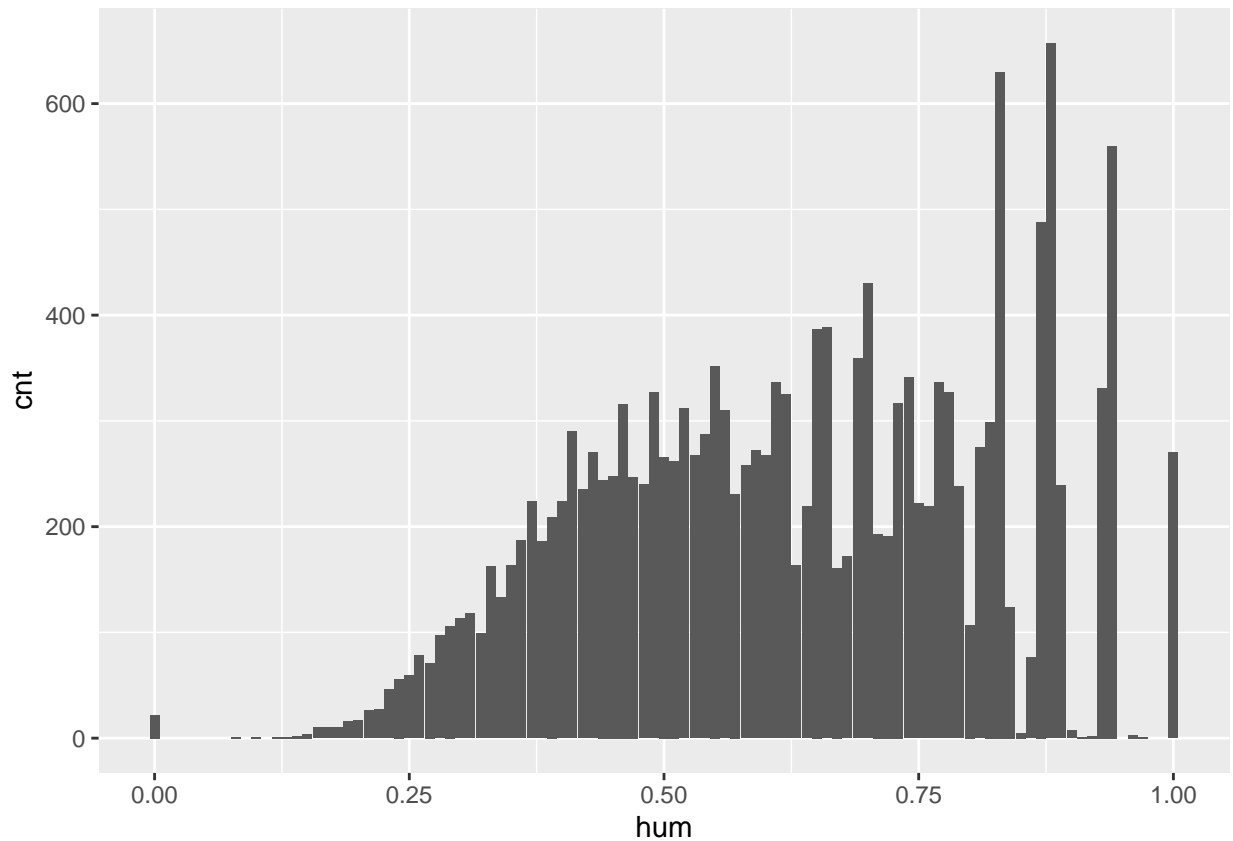
```
##      <dbl> <int>
## 1  0      22
## 2  0.08    1
## 3  0.1     1
## 4  0.12    1
## 5  0.13    1
## 6  0.14    2
## 7  0.15    4
## 8  0.16   10
## 9  0.17   10
## 10 0.18   10
## # i 79 more rows
```

```
qplot(x=hum,y=cnt,data=grupos6,geom='line')
```



```
p6<-ggplot(data=grupos6, aes(x=hum, y=cnt)) +
  geom_bar(stat="identity")
```

```
p6
```



```
top6 <- grupos6 %>%
  top_n(1,-cnt)

print(top6)
```

```
## # A tibble: 6 x 2
##   hum    cnt
##   <dbl> <int>
## 1  0.08     1
## 2  0.1      1
## 3  0.12     1
## 4  0.13     1
## 5  0.91     1
## 6  0.97     1
```

### ———— PREGUNTA 7 ————

¿Que condiciones climáticas serian ideales para nuestra demanda?  
(considere una función de densidad bivariable para la temperatura  
y la humedad)

### ———— RESPUESTA 7 ————

```
grupos7 <- data_frame_datos %>%
  group_by(hum,temp)%>%
```

```

summarise(cant_1 = n())

## `summarise()` has grouped output by 'hum'. You can override using the `.groups`
## argument.

print(grupos7)

## # A tibble: 1,073 x 3
## # Groups:   hum [89]
##   hum temp cant_1
##   <dbl> <dbl> <int>
## 1 0     0.34     5
## 2 0     0.36     4
## 3 0     0.38     2
## 4 0     0.4     3
## 5 0     0.42     3
## 6 0     0.44     5
## 7 0.08  0.52     1
## 8 0.1    0.54     1
## 9 0.12   0.48     1
## 10 0.13  0.44     1
## # i 1,063 more rows

top7 <- grupos7 %>%
  arrange(desc(cant_1))

print(top7)

## # A tibble: 1,073 x 3
## # Groups:   hum [89]
##   hum temp cant_1
##   <dbl> <dbl> <int>
## 1 0.83  0.66   117
## 2 0.89  0.64   114
## 3 0.83  0.62   105
## 4 0.88  0.6   103
## 5 0.79  0.7    94
## 6 0.74  0.7    91
## 7 0.84  0.7    90
## 8 0.88  0.56    88
## 9 0.73  0.64    86
## 10 0.89  0.66    86
## # i 1,063 more rows

```

---

### PREGUNTA 8

---

Muestre una gráfica de la densidad de rentas.

---

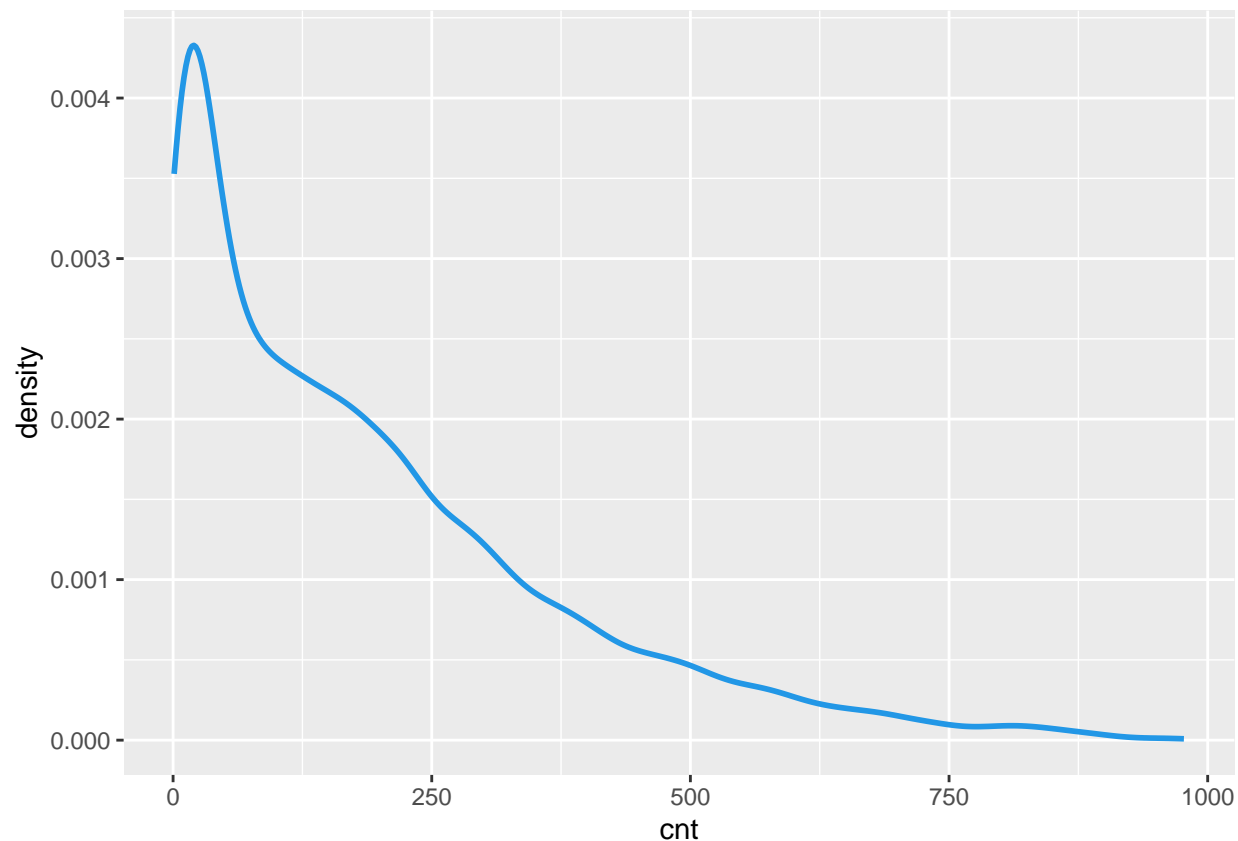
### RESPUESTA 8

---

```

ggplot(data_frame_datos, aes(x = cnt)) +
  geom_density(color = 4,      # Color
               lwd = 1,       # Ancho
               linetype = 1) # Tipo

```



#### PREGUNTA 9

¿En promedio de personas que rentan bicicletas y están registradas?

#### RESPUESTA 9

```
promedio_p9 <- mean(data_frame_datos$registered)
promedio_p9 <- round(promedio_p9,0)
promedio_p9
```

```
## [1] 154
```

#### PREGUNTA 10

Determine la mediana de personas que rentan bicicletas y no están registradas.

#### RESPUESTA 10

```
promedio_p10 <- mean(data_frame_datos$casual)
promedio_p10 <- round(promedio_p10,0)
promedio_p10
```

```
## [1] 36
```

### PREGUNTA 11

Deterimne la renta total, renta promedio por cada tipo de estación.

### RESPUESTA 11

```
grupos11 <- data_frame_datos %>%  
  group_by(season)%>%  
  summarise(cnt = n())  
  
print(grupos11)
```

```
## # A tibble: 4 x 2  
##   season    cnt  
##   <int> <int>  
## 1     1  4242  
## 2     2  4409  
## 3     3  4496  
## 4     4  4232
```

```
top11 <- grupos11 %>%  
  arrange(desc(cnt))  
print("Renta total por estación")
```

```
## [1] "Renta total por estación"
```

```
print(top11)
```

```
## # A tibble: 4 x 2  
##   season    cnt  
##   <int> <int>  
## 1     3  4496  
## 2     2  4409  
## 3     1  4242  
## 4     4  4232
```

```
grupos11_1 <- data_frame_datos %>%  
  group_by(season)%>%  
  summarise(promedio_season=mean(cnt))
```

```
grupos11_1
```

```
## # A tibble: 4 x 2  
##   season promedio_season  
##   <int>           <dbl>  
## 1     1           111.  
## 2     2           208.  
## 3     3           236.  
## 4     4           199.
```

---

### PREGUNTA 12

---

Determine y muestre una gráfica de barras la cantidad de rentas por tipo de temporada.

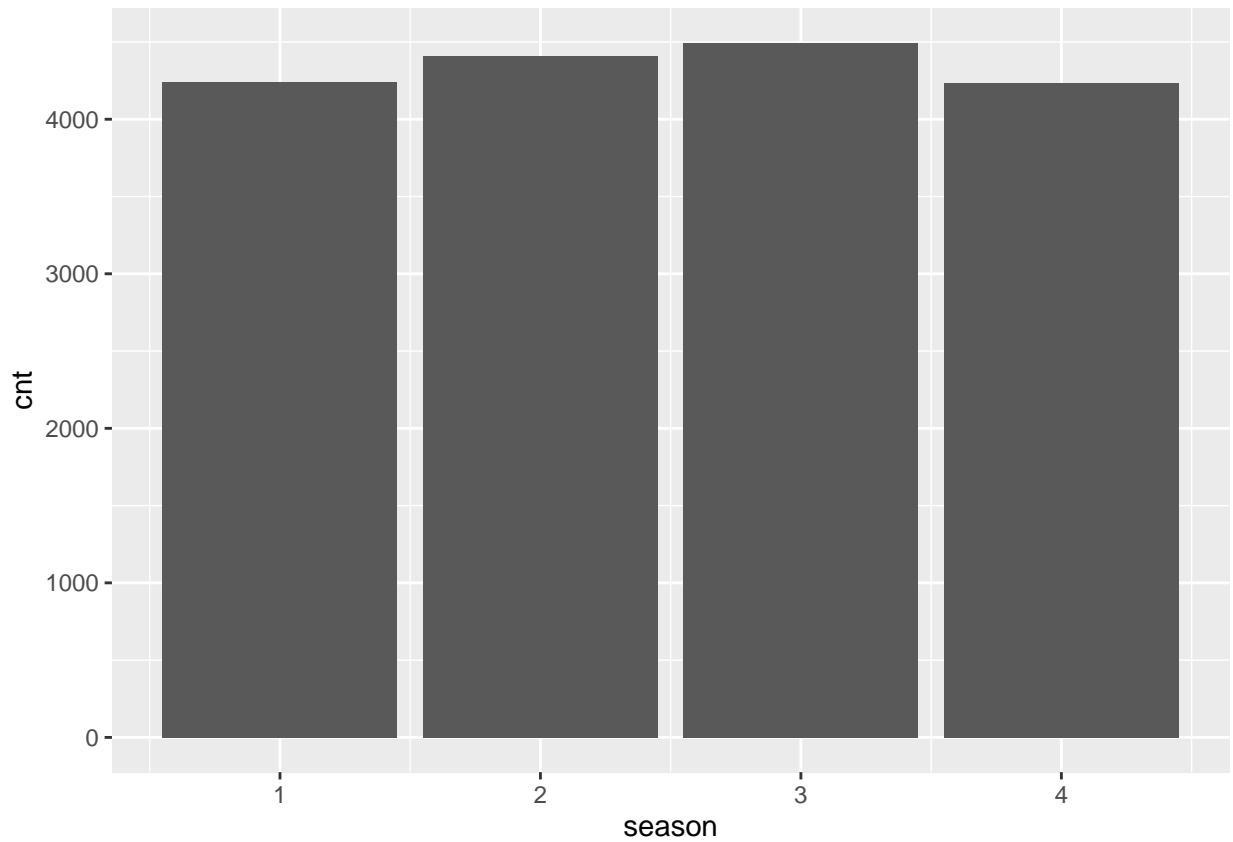
---

### RESPUESTA 12

---

```
p12<-ggplot(data=grupos11, aes(x=season, y=cnt)) +  
  geom_bar(stat="identity")
```

p12



---

### ## PREGUNTA 13

---

Muestre una gráfica de la densidad por hora.

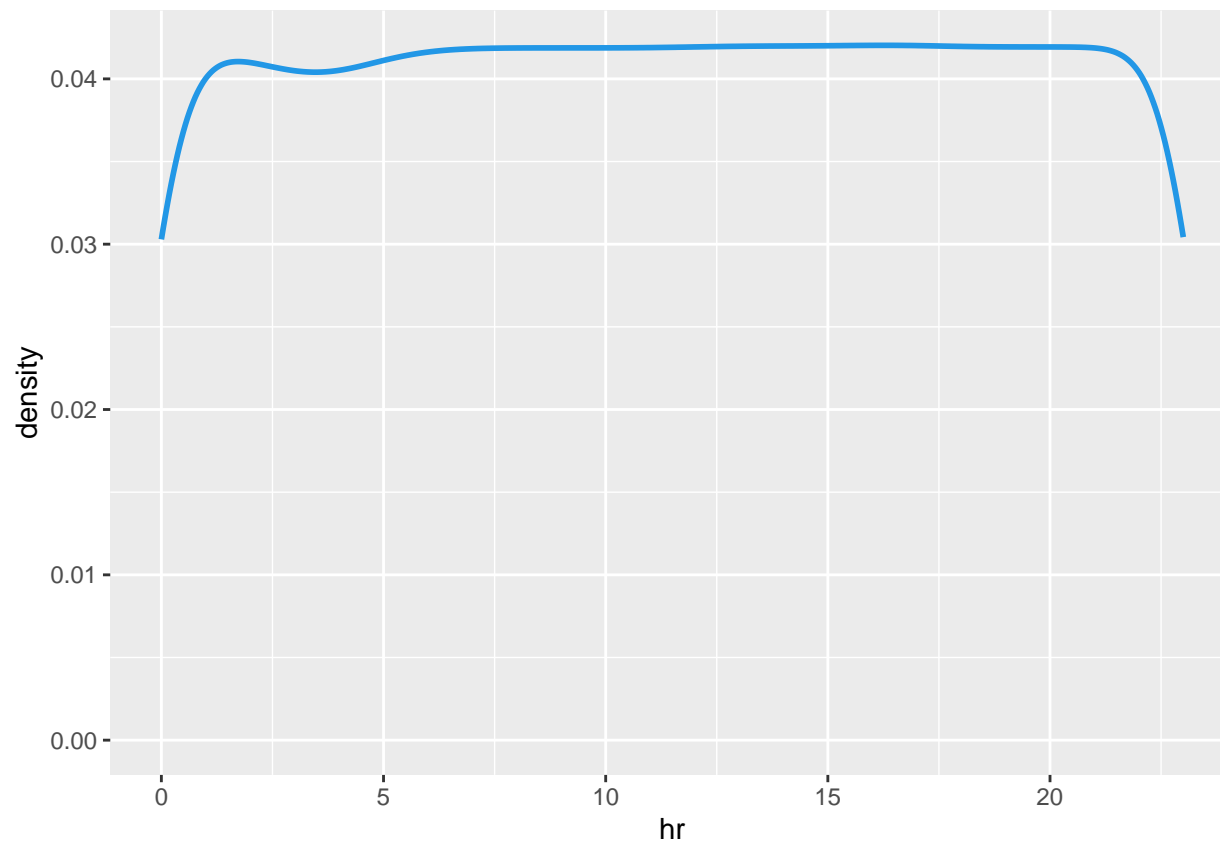
---

### RESPUESTA 13

---

```
ggplot(data_frame_datos, aes(x = hr)) +  
  geom_density(color = 4,      # Color  
              lwd = 1,        # Ancho  
              linetype = 1) # Tipo
```





## ————— PREGUNTA 14 —————

Muestre una gráfica de barras por día del mes como eje x y la cantidad total de alquileres como el eje Y.

————— RESPUESTA 14 —————

```
grupos14<-tidyr::separate(data_frame_datos, dteday, c("y", "m", "d"))
str(grupos14)
```

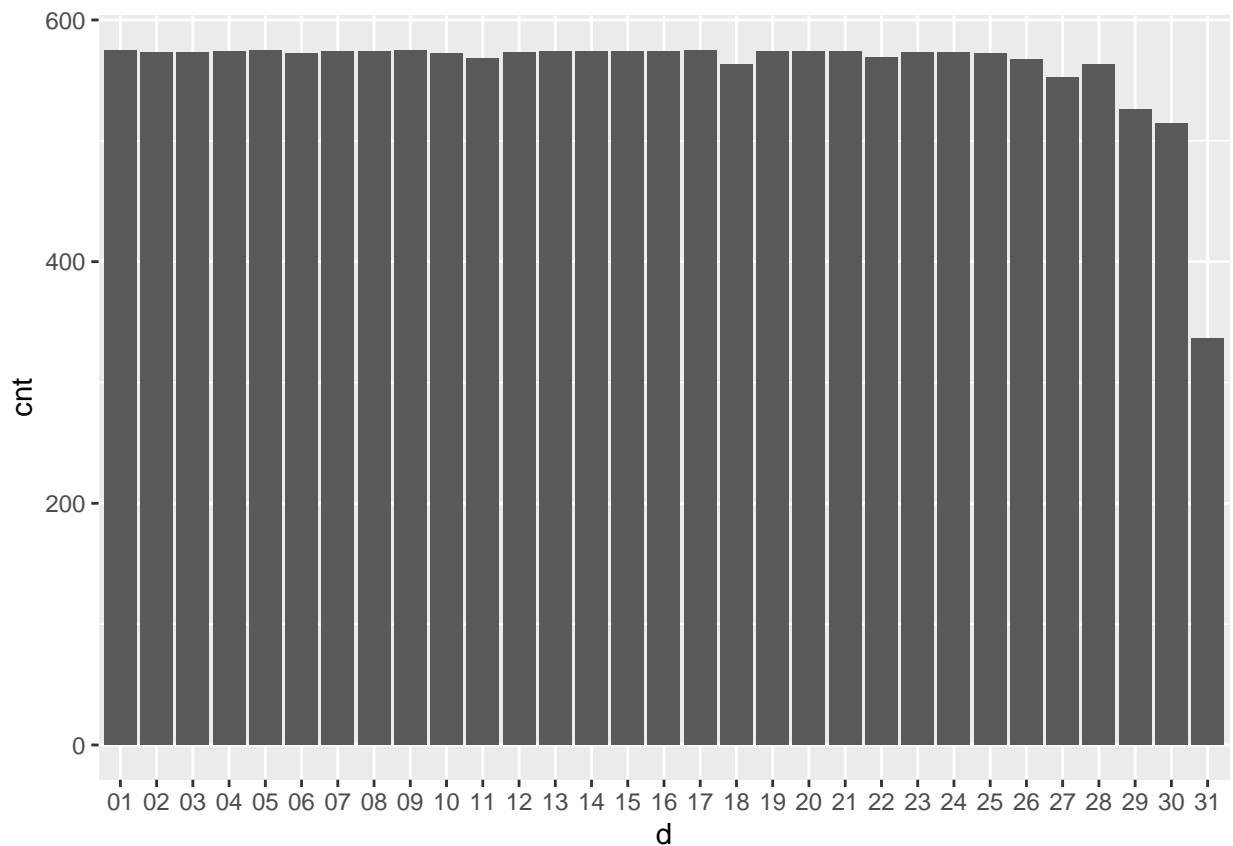
```
## 'data.frame':   17379 obs. of  19 variables:
## $ instant      : int   1 2 3 4 5 6 7 8 9 10 ...
## $ y            : chr   "2011" "2011" "2011" "2011" ...
## $ m            : chr   "01" "01" "01" "01" ...
## $ d            : chr   "01" "01" "01" "01" ...
## $ season       : int   1 1 1 1 1 1 1 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 ...
```

```
grupos14 <- grupos14 %>%
  group_by(d)%>%
  summarise(cnt = n())
```

```
p12<-ggplot(data=grupos14, aes(x=d, y=cnt)) +
  geom_bar(stat="identity")
```

p12



## ————— PREGUNTA 15 —————

Muestre una serie temporal (gráfica) con el tiempo (mes-día-año) como eje de tiempo y la cantidad de alquileres como eje Y.

————— RESPUESTA 15 —————

```
library(tidyverse)
```

```
## Warning: package 'tidyverse' was built under R version 4.2.3
```

```
## Warning: package 'tibble' was built under R version 4.2.3
```

```

## Warning: package 'tidyr' was built under R version 4.2.3
## Warning: package 'readr' was built under R version 4.2.3
## Warning: package 'purrr' was built under R version 4.2.3
## Warning: package 'stringr' was built under R version 4.2.3
## Warning: package 'forcats' was built under R version 4.2.3
## Warning: package 'lubridate' was built under R version 4.2.3
## -- Attaching core tidyverse packages ----- tidyverse 2.0.0 --
## v forcats   1.0.0     v stringr   1.5.0
## v lubridate 1.9.2     v tibble   3.2.1
## v purrr     1.0.1     v tidyr    1.3.0
## v readr     2.1.4
## -- Conflicts ----- tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()     masks stats::lag()
## i Use the conflicted package (<http://conflicted.r-lib.org/>) to force all conflicts to become errors

grupos15<-tidyr::separate(data_frame_datos, dteday, c("y", "m", "d"))

grupos15 <- grupos15 %>%
  group_by(y,m)%>%
  summarise(cnt = n())

## `summarise()` has grouped output by 'y'. You can override using the `.groups`
## argument.

grupos15 $ date <- paste (grupos15$y, grupos15$m, sep = "-")

grupos15 <- grupos15 %>%
  select(date,cnt)

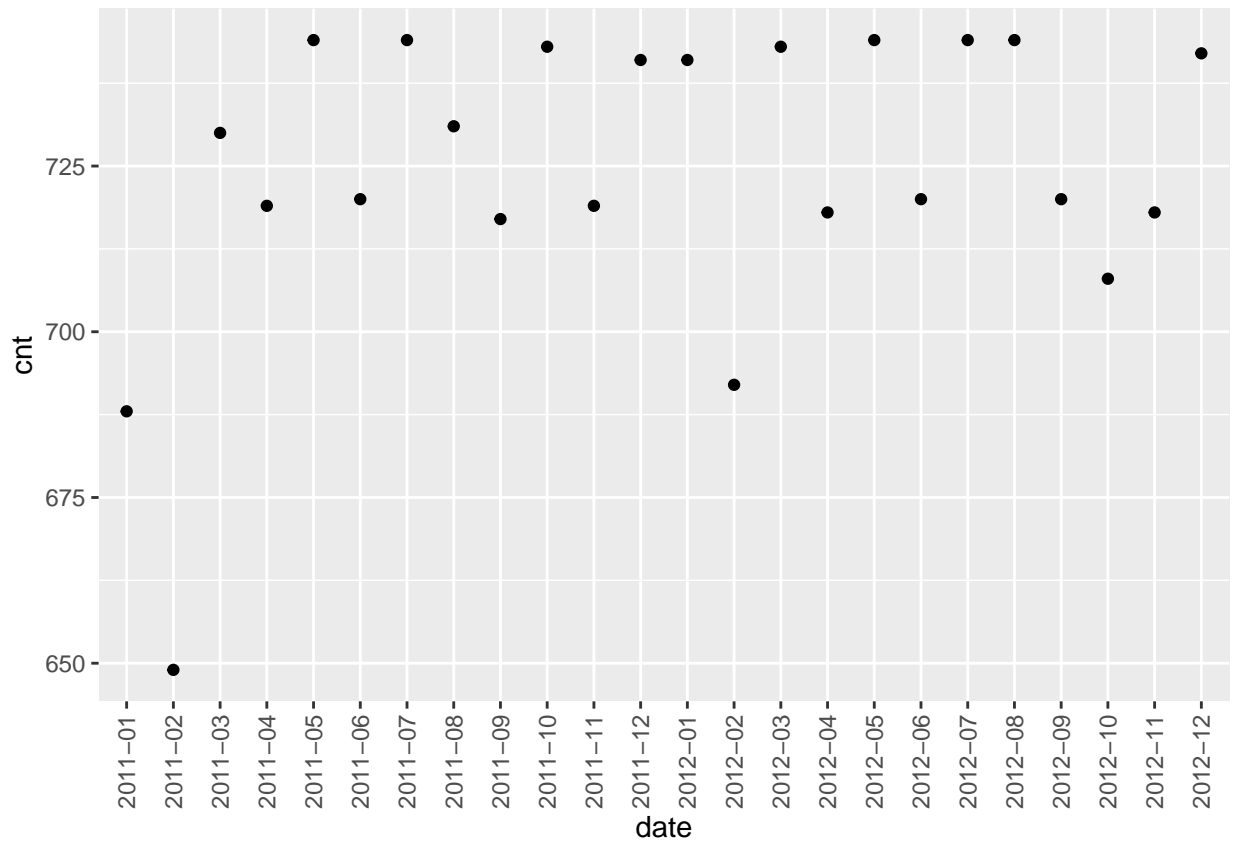
## Adding missing grouping variables: `y`

grupos15

## # A tibble: 24 x 3
## # Groups:   y [2]
##   y     date     cnt
##   <chr> <chr>   <int>
## 1 2011 2011-01    688
## 2 2011 2011-02    649
## 3 2011 2011-03    730
## 4 2011 2011-04    719
## 5 2011 2011-05    744
## 6 2011 2011-06    720
## 7 2011 2011-07    744
## 8 2011 2011-08    731
## 9 2011 2011-09    717
## 10 2011 2011-10   743
## # i 14 more rows

qplot(x=date,y=cnt,data=grupos15,geom='point')+
  theme(axis.text.x = element_text(angle = 90, vjust = 0.5, hjust=1))

```



## ————— PREGUNTA 16 —————

Muestre una gráfica de puntos que relaciones la temperatura como eje X y la humedad como eje Y.

————— RESPUESTA 16 —————

```
grupos16 <- data_frame_datos %>%
  select(hum,temp)

qplot(x=temp,y=hum,data=grupos16,geom='point')+
  theme(axis.text.x = element_text(angle = 90, vjust = 0.5, hjust=1))
```

