Machine Learning Project

PhD. Iara Del Pilar Solar Diaz

(exemplo prático)

O principal objetivo deste projeto é o de explorar os dados, realizar um aprendizado sobre os dados de forma a entender padrões e comportamentos, e sobre isso, encontrar o melhor modelo para ajuste, predição e inferência.

Instruções

Complete as tarefas abaixo.

Pegue os dados, na pasta o nome é "dados.txt"

Os dados são fictícios, mas fornecem uma base para algum entendimento e prática. Tentaremos ajustar um modelo de regressão linear.

DADOS: dados.csv

#datetime - data de mensuração

#season - estações - 1 = primavera, 2 = verão, 3 = outono, 4 = inverno

#holiday - onde os dias foram considerados como feriados

#workingday - dias aos quais foram mensurados os animais

#weather - previsão do tempo

1: céu limpo, poucas nuvens, parcialmente nublado, parcialmente nublado

2: Névoa + Nublado, Névoa + Nuvens quebradas, Névoa + Poucas Nuvens, Névoa

3: neve fraca, chuva fraca + trovoada + nuvens dispersas, chuva fraca + # nuvens dispersas

#4: Chuva forte + Paletes de gelo + Trovoada + Névoa, Neve + Nevoeiro

#temp - temperatura em Celsius

#atemp - "aproximada" temperatura em Celsius

#humidity - humidade relativa

#windspeed - velocidade do vento

#GC - grupos de contemporaneos

#CSM - consumo de matéria seca

Importe os dados e transforme os em data.frame

```
In [7]: df <- read.table('dados.txt', sep=',', header = TRUE)
```

Observe as primeiras linhas dos dados importados e a quantidade de observações totais

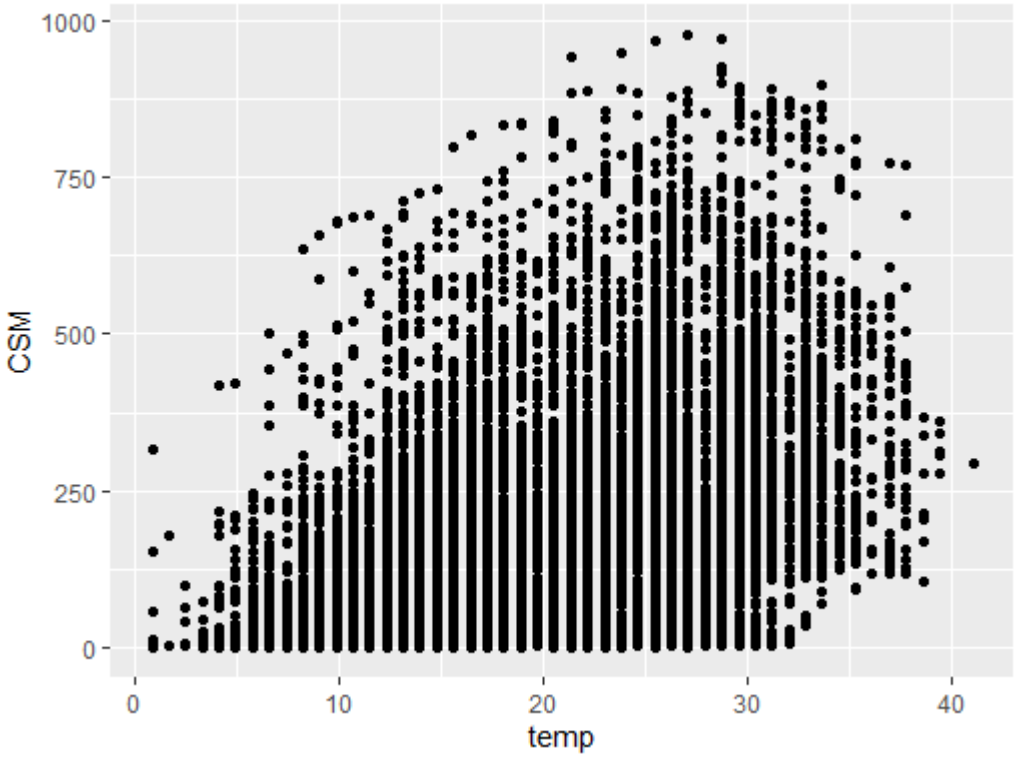
```
In [8]: head(df)
```

Out[8]:

	datetime	season	holiday	workingday	weather	temp	atemp	humidity	windspeed	casual	GC	CSM	hour
1	2011-01-01 00:00:00	1	0	0	1	9.84	14.395	81	0.0000	3	13	16	0
2	2011-01-01 01:00:00	1	0	0	1	9.02	13.635	80	0.0000	8	32	40	1
3	2011-01-01 02:00:00	1	0	0	1	9.02	13.635	80	0.0000	5	27	32	2
4	2011-01-01 03:00:00	1	0	0	1	9.84	14.395	75	0.0000	3	10	13	3
5	2011-01-01 04:00:00	1	0	0	1	9.84	14.395	75	0.0000	0	1	1	4
6	2011-01-01 05:00:00	1	0	0	2	9.84	12.880	75	6.0032	0	1	1	5

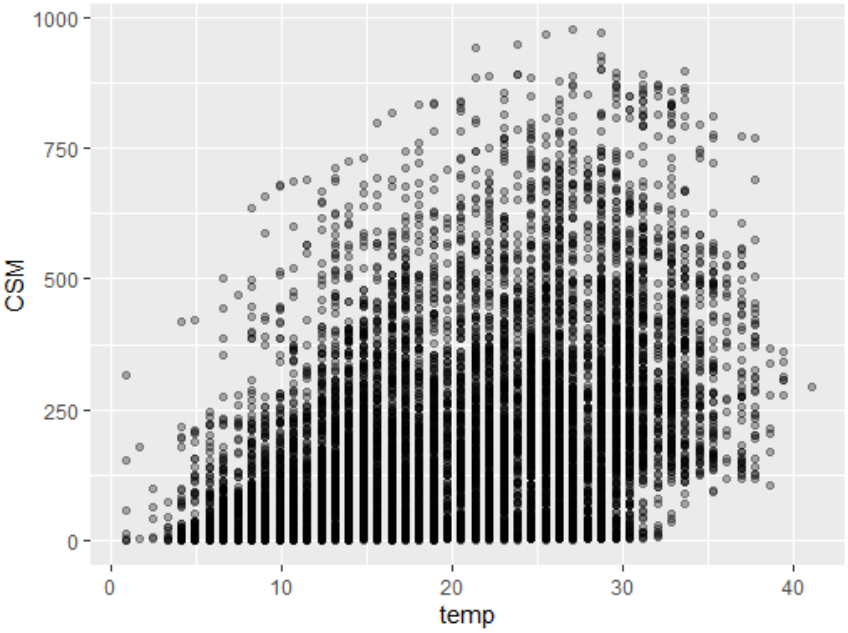
Queremos predizer o consumo dos animais, no total, durante cada hora, utilizando apenas os dados disponíveis. Seria possível?
Para isso, é necessário que comecemos a explorar os dados e observar se a variável atende algum padrão de comportamento.

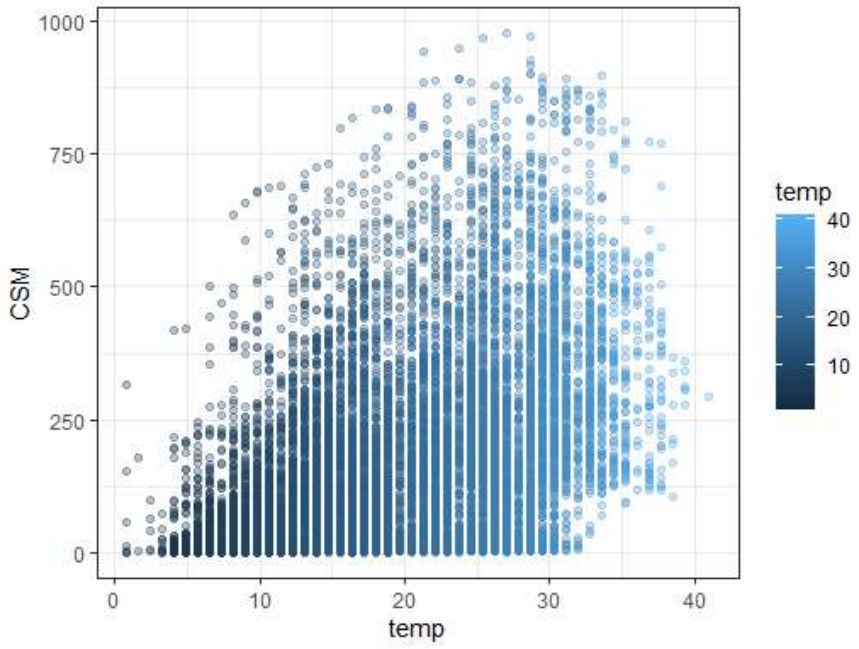
```
In [9]: library(ggplot2)
ggplot(df,aes(temp,CSM)) + geom_point()
```



A partir daqui começamos a explorar, de forma mais detalhada, as variáveis.

In [27]: ggplot(df,aes(temp,CSM)) + geom_point(alpha=0.3)
In [28]: ggplot(df,aes(temp,CSM)) + geom_point(alpha=0.3,aes(color=temp)) + theme_bw()

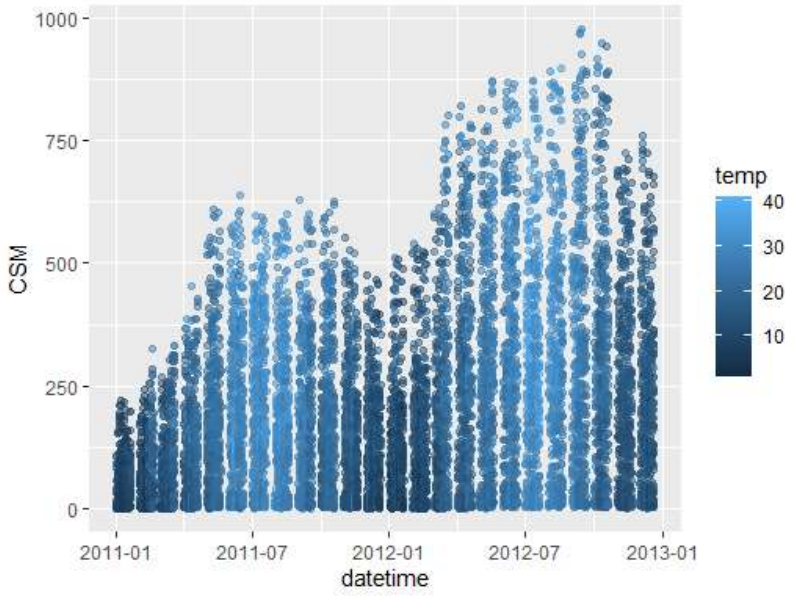




Para uma melhor visualização, faça um gráfico de consumo x data de mensuração (datetime) com cores em gradiente baseado na temperatura. O formato da data precisará ser convertido em POSIXct antes de ser plotado.

In [12]: df$datetime <- as.POSIXct(df$datetime,format="%Y-%m-%d %H:%M")
In [26]: head(df)

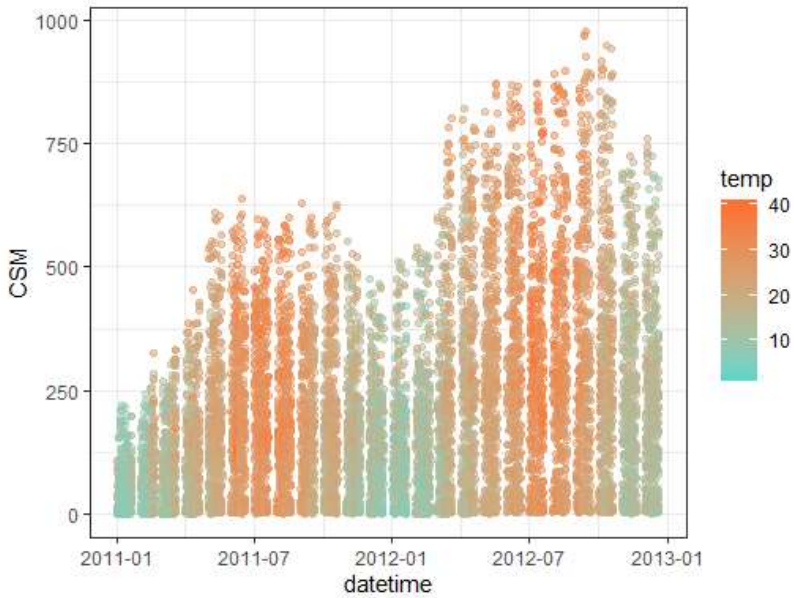
```
pl <- ggplot(df,aes(datetime,CSM)) + geom_point(aes(color=temp), alpha=0.5)

print(pl)
```



Espero que você tenha notado duas coisas: uma sazonalidade dos dados, para inverno e verão. Além disso, o consumo dos animais está aumentando, de forma geral. Isso pode representar um problema se formos tentar modelar a função de ajuste dos dados através de um modelo regressão linear, uma vez que os dados aparentam comportamento não linear.

Adicionamos mais cores para observar melhor o padrão de comportamento

In [26]: pl + scale_color_continuous(low='#55D8CE', high='#FF6E2E') + theme_bw()



Vamos ter uma visão geral rápida dos prós e contras da regressão linear agora:

Pros:

- Simples de explicar;
  - Altamente interpretável;
  - O treinamento dos dados e a predição na validação, são rápidos;
  - Nenhum ajuste é necessário (excluindo regularização);
  - As variáveis não precisam ser transformadas;
  - Pode ter um bom desempenho com um pequeno número de observações;

Contras:

- Assume uma relação linear entre as observações e a variável resposta;

• O desempenho (geralmente) não é competitivo comparado com os outros métodos de aprendizagem supervisionada devido ao alto viés;

• Não consegue "aprender" automaticamente as interações entre as variáveis

Importante também é observarmos a correlação entre as duas variáveis:

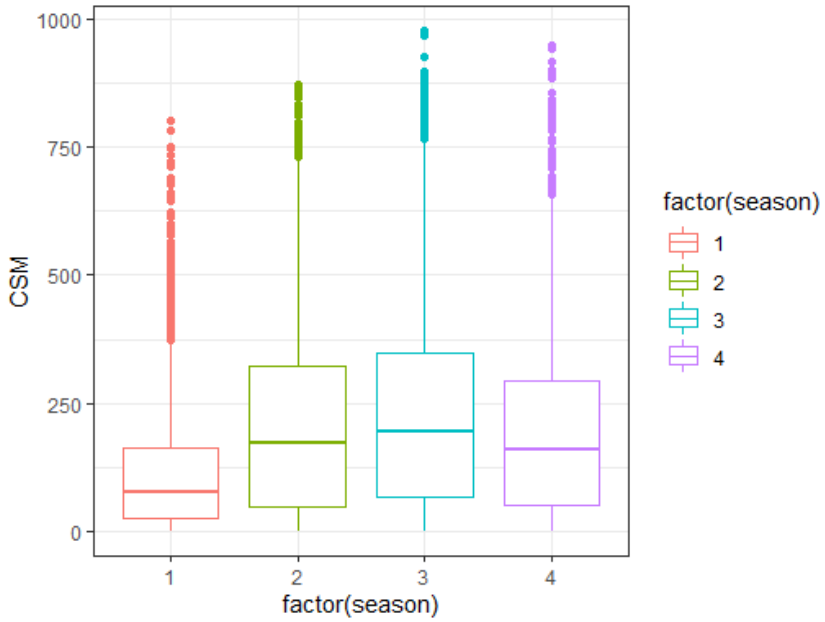
```
In [32]: cor(df[,c('temp','CSM')])
```

Out[32]:

	temp	**count**
temp	1.0000000	0.3944536
count	0.3944536	1.0000000

Continuamos explorando os dados de estação
criaremos um gráfico box plot com o Y indicando CSM (consumo dos animais) e cada box para cada estação (eixo X)

```
In [115]: ggplot(df,aes(factor(season),CSM)) + geom_boxplot(aes(color=factor(season))) +
theme_bw()
```



Observe o que este gráfico diz:

- Uma linha não sera suficiente para explicar a relação não linear entre as variáveis.
- O consumo dos animais aumenta no inverno que no outono.

É importante notar que retiramos essa conclusão apenas do número relacionado ao aumento do consumo, entretanto sabemos que existem outras variáveis que causam tal resultado.

Crie uma nova variável (hora) que será retirada das datas (datetime)

```
time.stamp <- bike$datetime[4]

format(time.stamp, "%H")
```

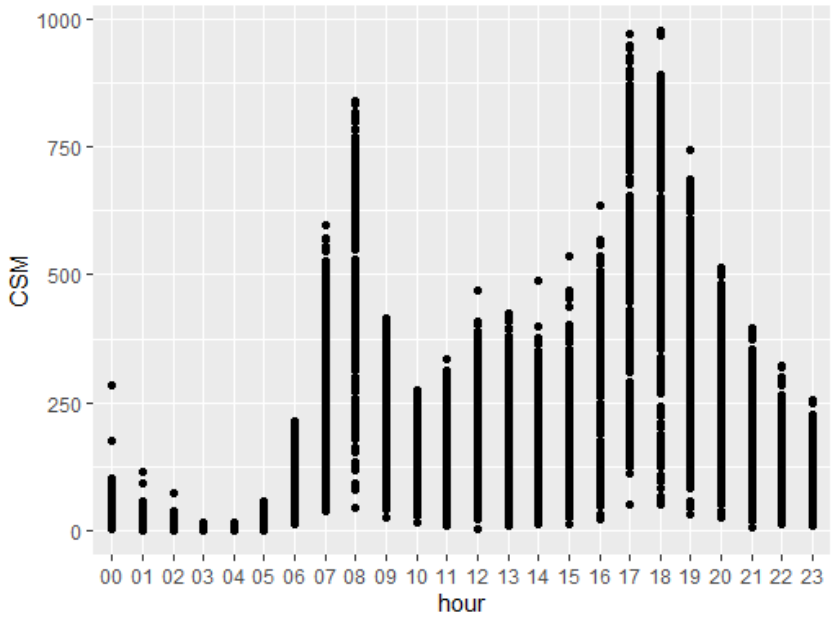
```
In [60]: df$hour <- sapply(df$datetime,function(x){format(x,"%H")})

head(df)
Out[61]:
```

	datetime	season	holiday	workingday	weather	temp	atemp	humidity	windspeed
1	2011-01-01 00:00:00	1	0	0	1	9.84	14.395	81	0.0000
2	2011-01-01 01:00:00	1	0	0	1	9.02	13.635	80	0.0000
3	2011-01-01 02:00:00	1	0	0	1	9.02	13.635	80	0.0000
4	2011-01-01 03:00:00	1	0	0	1	9.84	14.395	75	0.0000

```
5 2011-01-01 04:00:00      1      0          0          1 9.84 14.395         75    0.0000
6 2011-01-01 05:00:00      1      0          0          2 9.84 12.880         75    6.0032
  casual GC CSM hour
1      3 13  16   00
2      8 32  40   01
3      5 27  32   02
4      3 10  13   03
5      0  1   1   04
6      0  1   1   05
```

Com a nova variável, criaremos um gráfico com hora e consumo.

```
In [78]: library(dplyr)

pl <- ggplot(filter(df,workingday==1),aes(hour,CSM))
pl <- pl + geom_point()

print(pl)
```

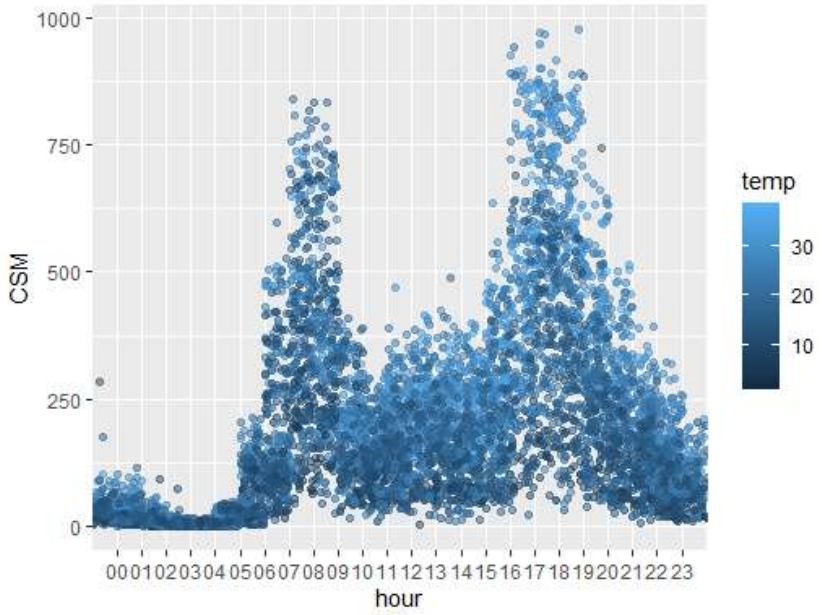


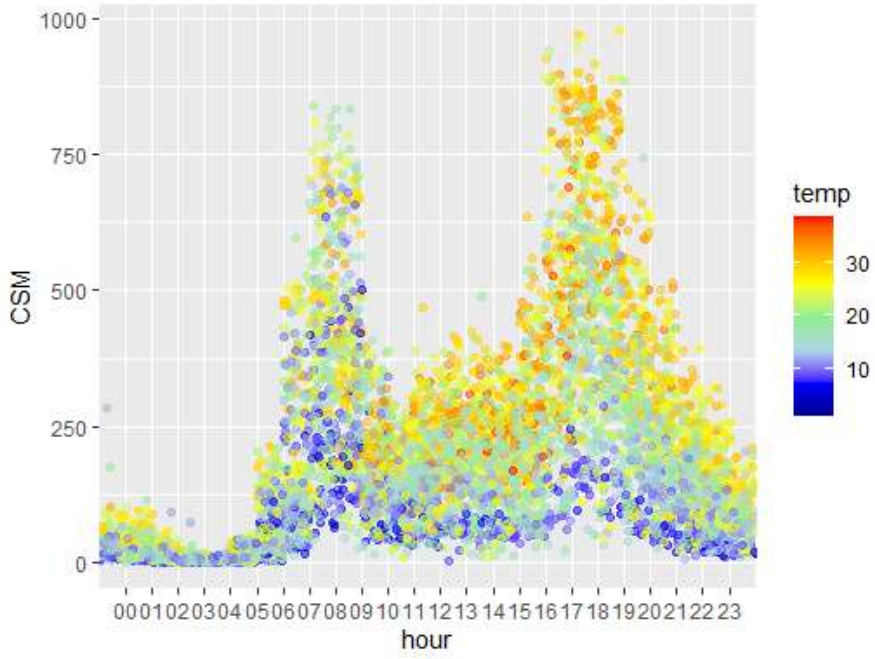
pl <- ggplot(filter(df,workingday==1),aes(hour,CSM))
pl <- pl + geom_point(position=position_jitter(w=1,h=0),aes(color=temp),alpha=0.5)

print(pl)

pl <- pl + scale_color_gradientn(colours = c('dark blue', 'blue', 'light blue', 'light green','yellow', 'orange', 'red'))

print(pl)





Por estes gráficos, podemos notar que existe um "pico" de atividade no consumo durante a manhã (~8am) e novamente em torno das 17 horas.

Após a observação deste padrão, iremos construir e "treinar" o modelo.

Construindo o modelo.

Vemos primeiro o comportamento e ajuste para depois ver se é eficiente fazer um training data e um test data

#sazonalidade dos dados

Use lm() para construir o modelo que prediz o consumo baseado apenas na variável temperatura. Daremos o nome de temp.model

In [105]: temp.model <- lm(CSM ~ temp, df)

Obtenha a estatística do modelo - temp.model

In [107]: summary(temp.model)

Out[107]:

```
Call:
lm(formula = CSM ~ temp, data = df)

Residuals:
   Min     1Q  Median     3Q    Max
-293.32 -112.36  -33.36   78.98  741.44

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)   6.0462     4.4394   1.362    0.173
temp          9.1705     0.2048  44.783   <2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 166.5 on 10884 degrees of freedom
Multiple R-squared:  0.1556,  Adjusted R-squared:  0.1555
F-statistic:  2006 on 1 and 10884 DF,  p-value: < 2.2e-16
```

O que o intercepto e o coeficiente linear te dizem sobre os dados?

#QUEREMOS PREDIZER QUANTO DE CONSUMO EU TEREI SE EU TIVER UMA TEMPERATURA DE 25 GRAUS??

```
predicao <- 6.0462 + 9.17*25
```

```
cat(paste("predição de CS dado que o nosso modelo foi apenas treinado com a temperatura : ") , predicao)
```

```
temp.test <- data.frame(temp=c(25))
predict(temp.model, temp.test)


df$hour <-sapply(df$hour, as.numeric)
head(df)

model <- lm(CSM ~ . -casual - datetime - atemp, df)
summary(model)
```

Out[128]:

```
Call:
lm(formula = CSM ~ . - casual - datetime - atemp, data = df)

Residuals:
    Min      1Q  Median      3Q     Max 
-84.497 -19.824  -3.182  13.776 260.267 

Coefficients:
             Estimate Std. Error t value Pr(>|t|)    
(Intercept)  33.319077   1.913141  17.416  < 2e-16 ***
season       -0.436602   0.310416  -1.407   0.1596    
holiday     -13.402453   1.989385  -6.737 1.70e-11 ***
workingday  -41.465854   0.717472 -57.794  < 2e-16 ***
weather       2.510886   0.565287   4.442 9.01e-06 ***
temp          2.220932   0.044603  49.793  < 2e-16 ***
humidity     -0.611899   0.020865 -29.326  < 2e-16 ***
windspeed    -0.080265   0.042185  -1.903   0.0571 .  
GC            1.117463   0.002489 448.902  < 2e-16 ***
hour          0.394025   0.051654   7.628 2.58e-14 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 33.46 on 10875 degrees of freedom
  (1 observation deleted due to missingness)
Multiple R-squared:  0.9659,  Adjusted R-squared:  0.9659 
F-statistic: 3.424e+04 on 9 and 10875 DF,  p-value: < 2.2e-16
```

O modelo teve um bom desempenho nos dados de treinamento? O que você acha de usar um modelo de regressão linear nesses dados?

Você deve ter notado que esse tipo de modelo não funciona bem, uma vez que esses dados mostram sazonalidade e de série temporal. Precisamos de um modelo que possa levar em conta esse tipo de tendência, como por exemplo: Regression Forests. Uma

outra opção, é separar os dados em treinamento e validação. Mas em vez de uma divisão aleatória, sua divisão deve ser dados "futuros" para teste, dados "anteriores" de acordo com a sazonalidade.