

ggplot util

Gregorio ULSILVA

16/11/2021

```
library(ggplot2)
```

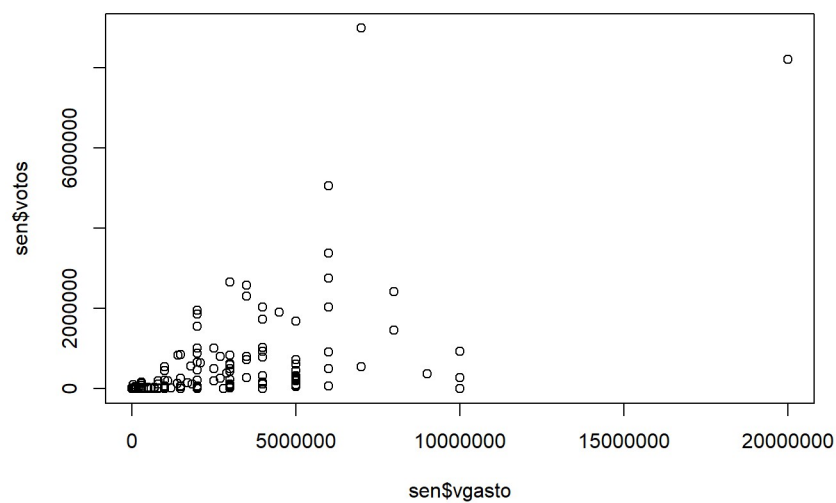
```
## Warning: package 'ggplot2' was built under R version 4.0.5
```

```
library(RColorBrewer)
options(scipen = 999)
```

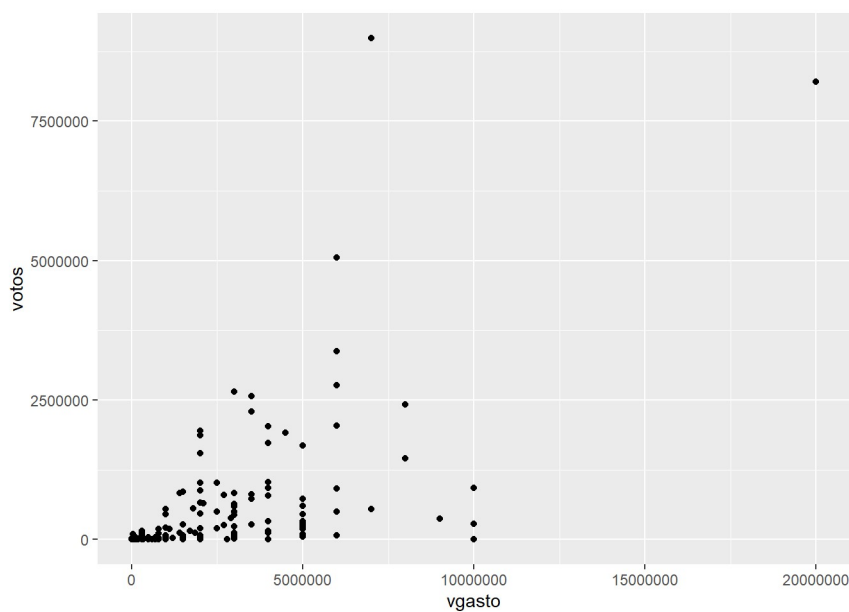
observação usei uma base do sen 2006, pq eu assisti a aula gravada e não foi disponibilizada a base da aula (sen2018)

```
load("C:/Users/grego/OneDrive/aulas R GGLOT_L00PS_PCA/senado2006.RData")
```

```
plot(sen$vgasto, sen$votos)
```

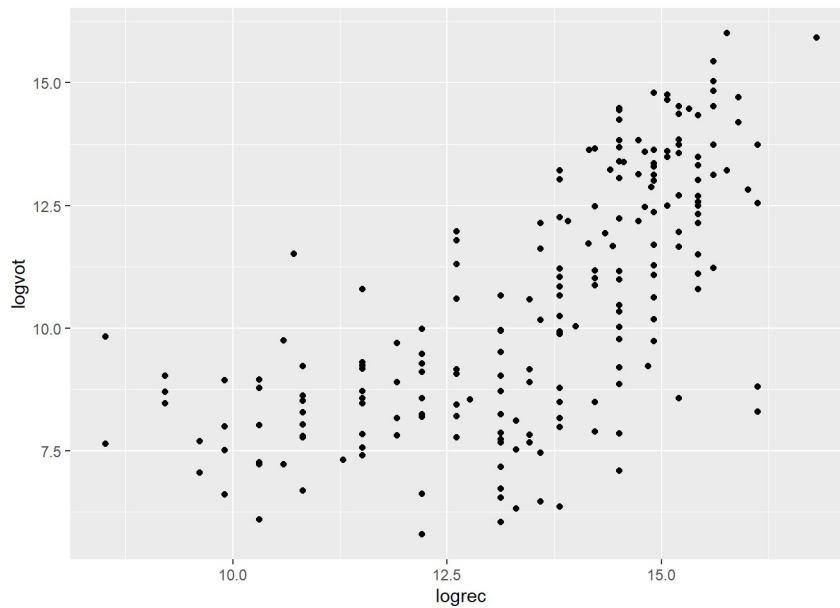


```
dis <- ggplot(sen, aes(vgasto, votos))
dis + geom_point()
```



```
sen <- subset(sen, votos > 0) # seleciona apenas casos com votos > 0
sen$logrec <- log(sen$vgasto) # Logaritimiza
sen$logvot <- log(sen$votos) # idem
```

```
dis <- ggplot(sen, aes(logrec, logvot))
dis + geom_point()
```

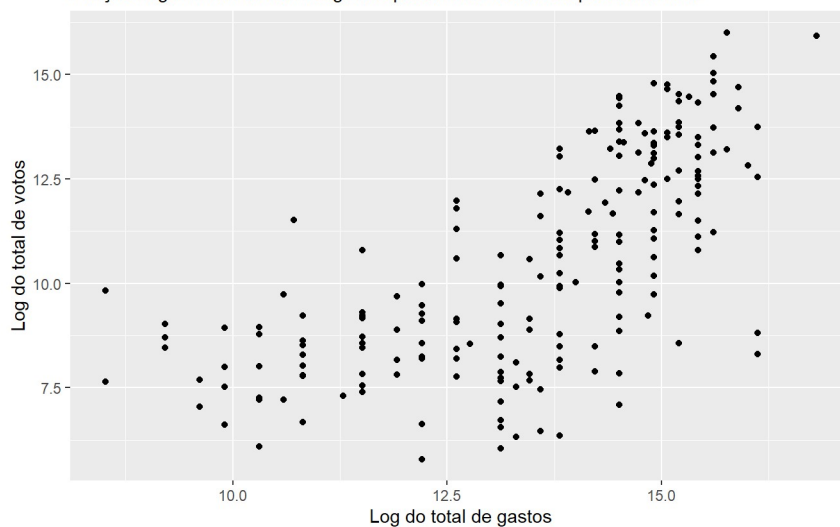


```
dis <- dis+geom_point() # salva no objeto 'dis' o mapa estético e o objeto geométrico
```

```
dis + labs(title = "Gastos e votos por candidato", subtitle = "Função Logarítmica do total de gastos pelos votos recebidos p  
elo candidato", x = "Log do total de gastos", y = "Log do total de votos", caption = "Fonte: TSE")
```

Gastos e votos por candidato

Função Logarítmica do total de gastos pelos votos recebidos pelo candidato

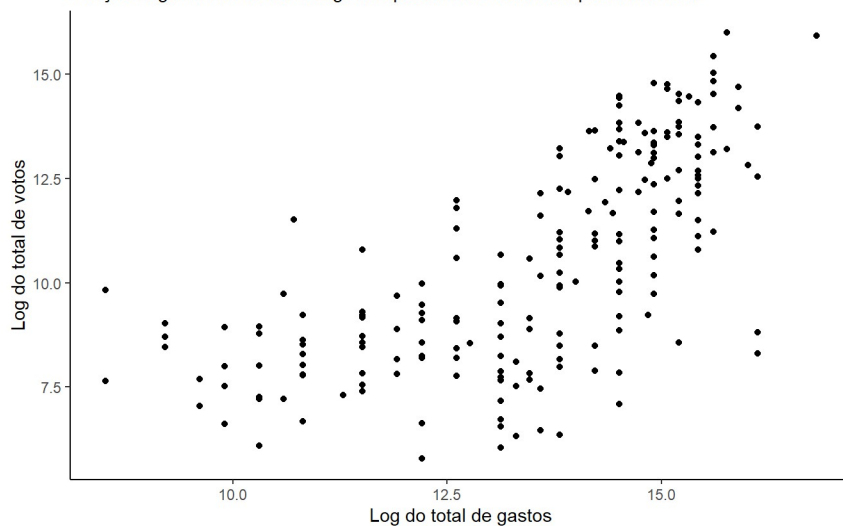


Fonte: TSE

```
dis + labs(title = "Gastos e votos por candidato", subtitle = "Função Logarítmica do total de gastos pelos votos recebidos p  
elo candidato", x = "Log do total de gastos", y = "Log do total de votos", caption = "Fonte: TSE") + theme_classic()
```

Gastos e votos por candidato

Função Logarítmica do total de gastos pelos votos recebidos pelo candidato



Fonte: TSE

Outros temas:

`theme_dark()`

`theme_grey()`

`theme_light()`

`theme_minimal()`

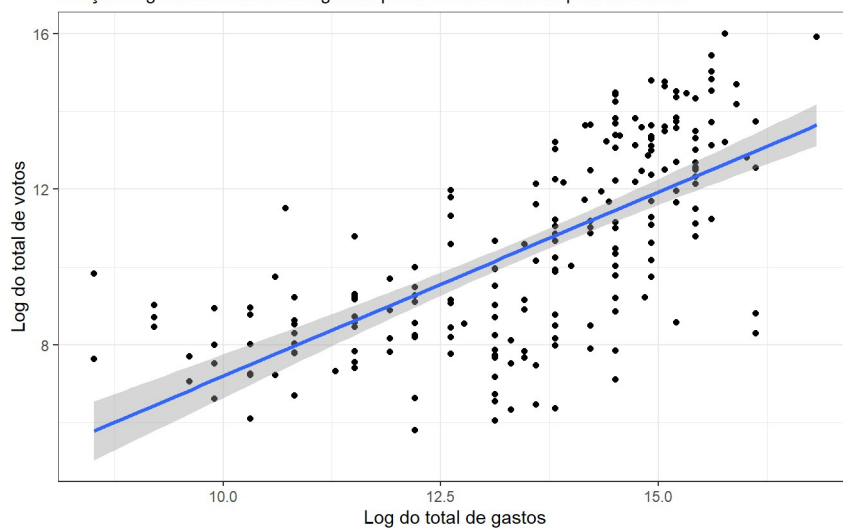
`theme_void()`

```
# adicionando linha do gráfico de dispersão
dis + geom_smooth(method = lm)+labs(title = "Gastos e votos por candidato", subtitle = "Função Logarítmica do total de gastos pelos votos recebidos pelo candidato", x = "Log do total de gastos", y = "Log do total de votos", caption = "Fonte: TSE") + theme_bw()
```

```
## `geom_smooth()` using formula 'y ~ x'
```

Gastos e votos por candidato

Função Logarítmica do total de gastos pelos votos recebidos pelo candidato



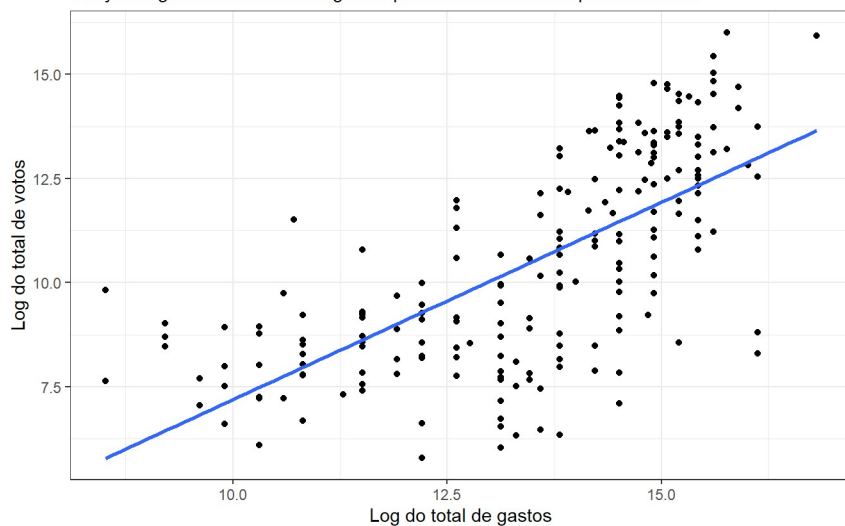
Fonte: TSE

```
dis + geom_smooth(method = lm, se = F)+labs(title = "Gastos e votos por candidato", subtitle = "Função Logarítmica do total de gastos pelos votos recebidos pelo candidato", x = "Log do total de gastos", y = "Log do total de votos", caption = "Fonte: TSE") + theme_bw()
```

```
## `geom_smooth()` using formula 'y ~ x'
```

Gastos e votos por candidato

Função Logarítmica do total de gastos pelos votos recebidos pelo candidato



Fonte: TSE

olha que legal somente a linha

adicionando um texto (R2) dentro do gráfico

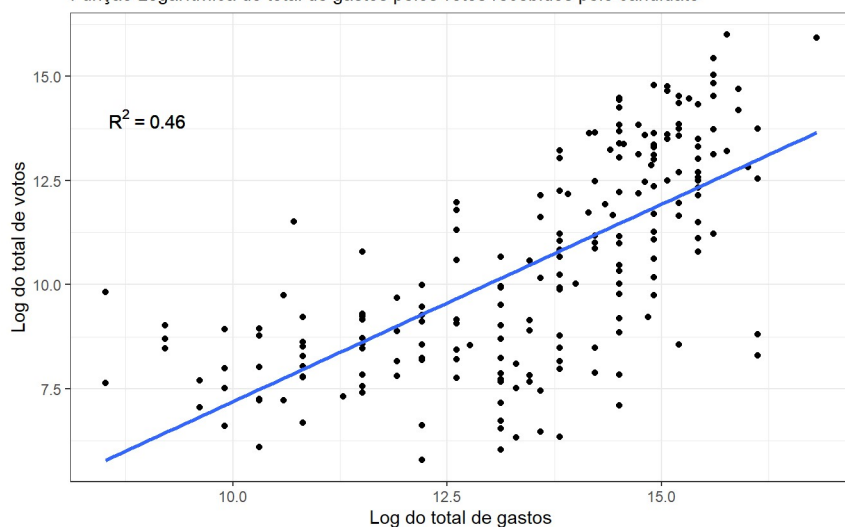
```
reg <- lm(logvot~logrec, data= sen)
r2lab <- bquote(R^2~"="~.(round(summary(reg)$r.squared,2))) # transformando o r2 em expressão
dis + geom_smooth(method = lm, se = F)+labs(title = "Gastos e votos por candidato", subtitle = "Função Logarítmica do total
de gastos pelos votos recebidos pelo candidato", x = "Log do total de gastos", y = "Log do total de votos", caption = "Font
e: TSE") + annotate(geom = 'text', x = 9, y = 14, label = r2lab) + theme_bw()
```

```
## `geom_smooth()` using formula 'y ~ x'
```

```
## Warning in is.na(x): is.na() aplicado a um objeto diferente de lista ou vetor de
## tipo 'language'
```

Gastos e votos por candidato

Função Logarítmica do total de gastos pelos votos recebidos pelo candidato



Fonte: TSE

para mudar tamanho de fonte

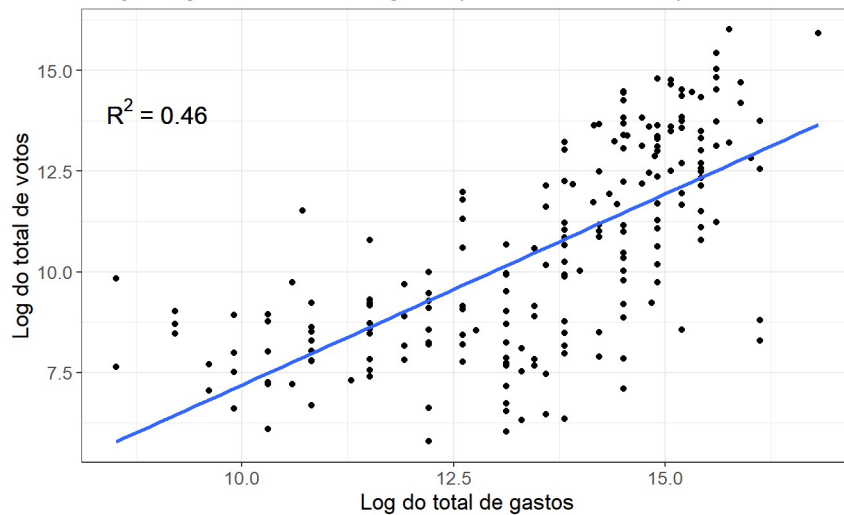
```
dis + geom_smooth(method = lm, se = F)+labs(title = "gastos e votos por candidato", subtitle = "Função Logarítmica do total
de gastos pelos votos recebidos pelo candidato", x = "Log do total de gastos", y = "Log do total de votos", caption = "Font
e: TSE") + annotate(geom = 'text', x = 9, y = 14, label = r2lab, size = 5) + theme_bw() + theme(text = element_text(size = 1
3))
```

```
## `geom_smooth()` using formula 'y ~ x'
```

```
## Warning in is.na(x): is.na() aplicado a um objeto diferente de lista ou vetor de
## tipo 'language'
```

gastos e votos por candidato

Função Logarítmica do total de gastos pelos votos recebidos pelo candidato



Fonte: TSE

só mudar alguns:

`axis.text.x`

`axis.text.y`

`axis.title`

etc

comando `help(theme)` para mais infos

Gráfico de Barras

primeiro vamos criar uma variável para servir de marcador - usa para fazer contagem de variáveis

```
sen$cont <- 1
```

vamos plotar o numero de candidatos de 4 dos partidos da lista PMDB, PT, PP e PSDB

Podemos fazer isso de duas formas.

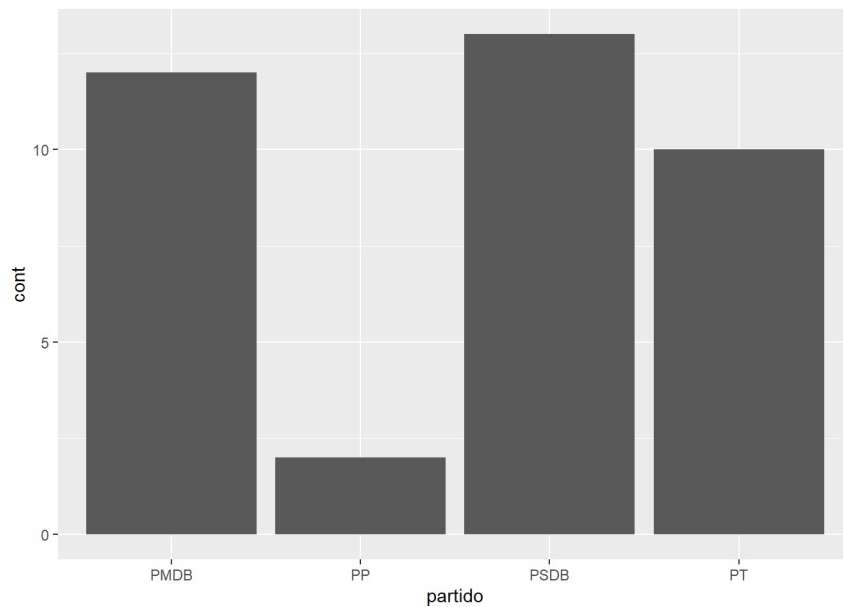
a primeira é criar um subset:

```
sen_4partidos <- subset(sen, partido == "PMDB" | partido == "PT" | partido == "PP" | partido == "PSDB")
table(sen_4partidos$partido)
```

```
##
##  PAN  PCB PC do B  PCO  PDT  PFL  PHS  PL  PMDB  PMN
##    0    0    0    0    0    0    0    0    12    0
##  PP  PPS  PRB  PRONA  PRP  PRTB  PSB  PSC  PSDB  PSDC
##    2    0    0    0    0    0    0    0    13    0
##  PSL  PSOL  PSTU  PT  PTB  PTC  PT do B  PTN  PV
##    0    0    0    10  0    0    0    0    0
```

outra forma dentro do `ggplot`(sem criar um objeto no ambiente global)

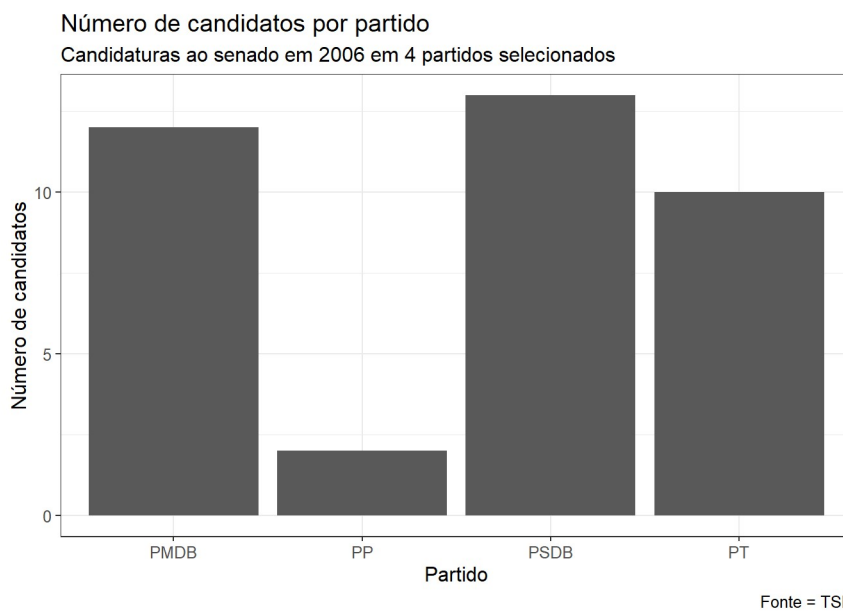
```
bar <- ggplot(subset(sen, partido == "PMDB" | partido == "PT" | partido == "PP" | partido == "PSDB"), aes(partido, cont))
bar + geom_bar(stat = "identity")
```



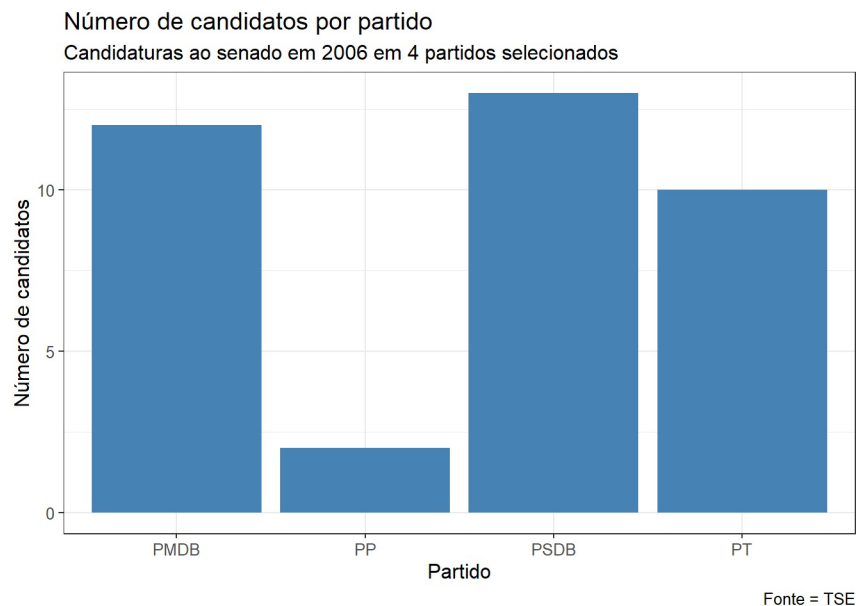
olha que maravilhaaaaaaaa

Alterando elementos textuais:

```
bar + geom_bar(stat = "identity") + labs(title = "Número de candidatos por partido", subtitle = "Candidaturas ao senado em 2006 em 4 partidos selecionados", x = "Partido", y = "Número de candidatos", caption = "Fonte = TSE") + theme_bw() + theme(text = element_text(size = 12))
```

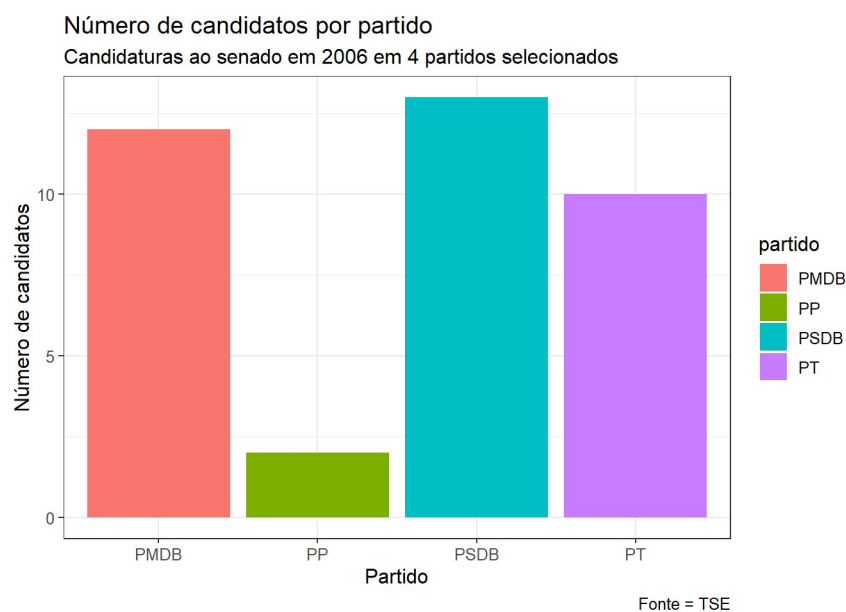


```
bar + geom_bar(stat = "identity", fill = "steelblue") + labs(title = "Número de candidatos por partido", subtitle = "Candidaturas ao senado em 2006 em 4 partidos selecionados", x = "Partido", y = "Número de candidatos", caption = "Fonte = TSE") + theme_bw() + theme(text = element_text(size = 12))
```



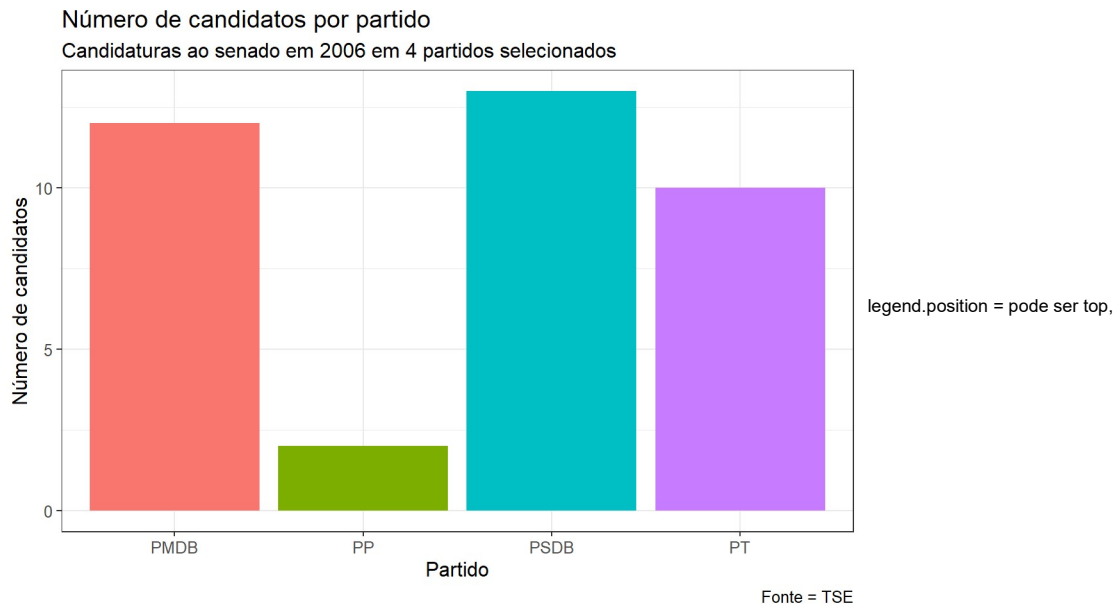
uma cor para cada partido?

```
bar <- ggplot(subset(sen, partido == "PMDB" | partido == "PT" | partido == "PP" | partido == "PSDB"), aes(partido, cont, fill = partido))
bar + geom_bar(stat = "identity") + labs(title = "Número de candidatos por partido", subtitle = "Candidaturas ao senado em 2006 em 4 partidos selecionados", x = "Partido", y = "Número de candidatos", caption = "Fonte = TSE") + theme_bw() + theme(text = element_text(size = 12))
```



removendo a legenda

```
bar <- ggplot(subset(sen, partido == "PMDB" | partido == "PT" | partido == "PP" | partido == "PSDB"), aes(partido, cont, fill = partido))
bar + geom_bar(stat = "identity") + labs(title = "Número de candidatos por partido", subtitle = "Candidaturas ao senado em 2006 em 4 partidos selecionados", x = "Partido", y = "Número de candidatos", caption = "Fonte = TSE") + theme_bw() + theme(text = element_text(size = 12), legend.position = "none")
```



bottom ou left

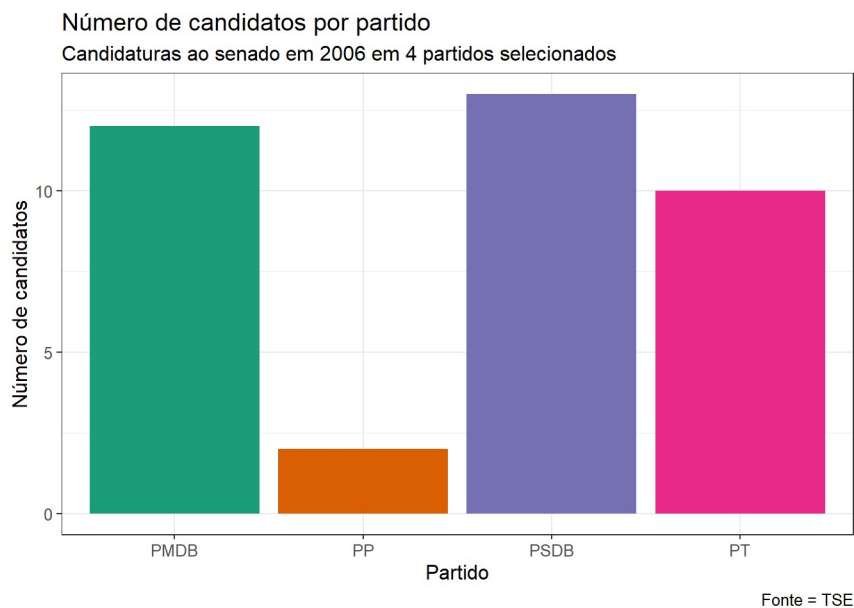
Usando paleta de cores

A função 'scale_fill_brewer()' emprega a paleta de cores do pacote 'RColorBrewer'

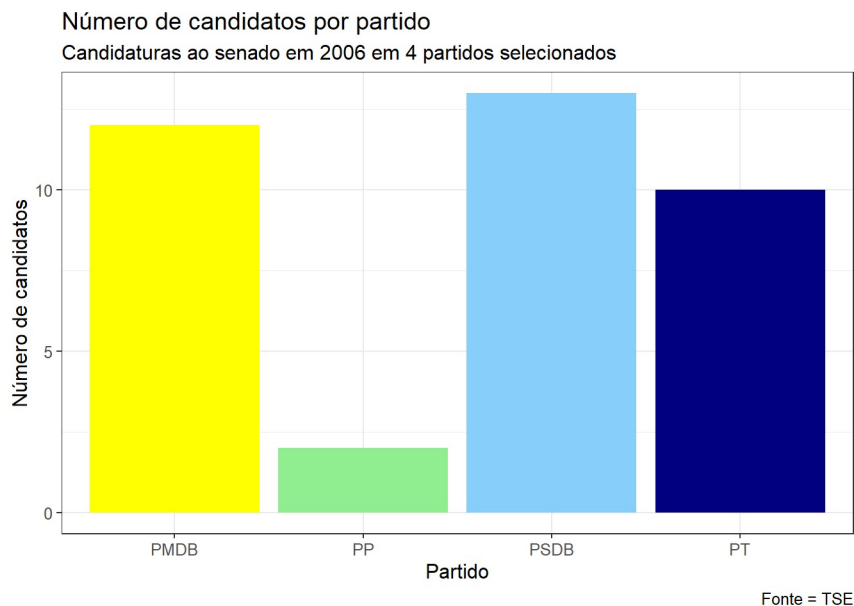
já 'scale_fill_manual()' atribui manualmente as cores. A atribuição é feita com a função c(), colocando o código HTML de cada cor , usando '#' antes do código.

já 'scale_fill_grey()' emprega tons de cinza

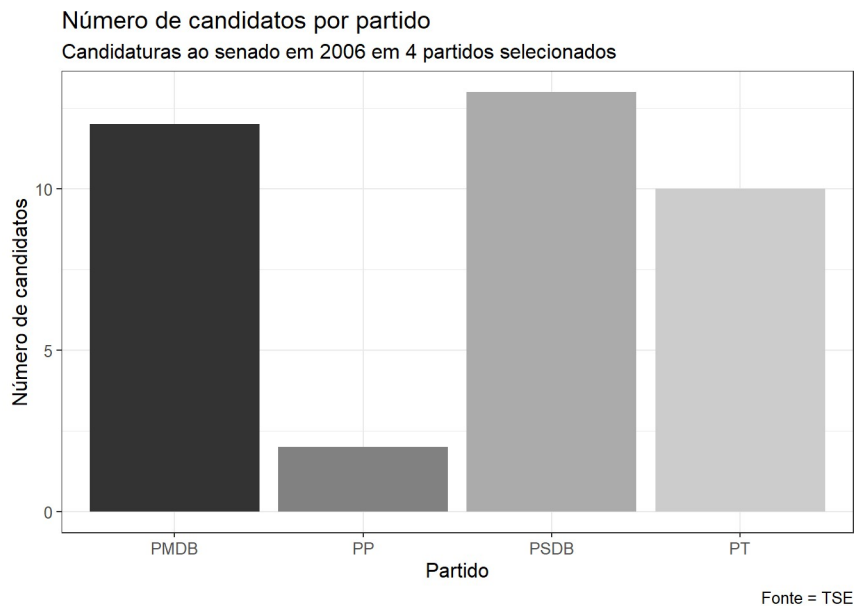
```
bar <- ggplot(subset(sen, partido == "PMDB" | partido == "PT" | partido == "PP" | partido == "PSDB"), aes(partido, cont, fill = partido))
bar + geom_bar(stat = "identity") + labs(title = "Número de candidatos por partido", subtitle = "Candidaturas ao senado em 2006 em 4 partidos selecionados", x = "Partido", y = "Número de candidatos", caption = "Fonte = TSE") + theme_bw() + theme(text = element_text(size = 12), legend.position = "none") + scale_fill_brewer(palette = "Dark2")
```



```
bar <- ggplot(subset(sen, partido == "PMDB" | partido == "PT" | partido == "PP" | partido == "PSDB"), aes(partido, cont, fill = partido))
bar + geom_bar(stat = "identity") + labs(title = "Número de candidatos por partido", subtitle = "Candidaturas ao senado em 2006 em 4 partidos selecionados", x = "Partido", y = "Número de candidatos", caption = "Fonte = TSE") + theme_bw() + theme(text = element_text(size = 12), legend.position = "none") + scale_fill_manual(values = c("#FFFF00", "#90EE90", "#87CEFA", "#000080"))
```

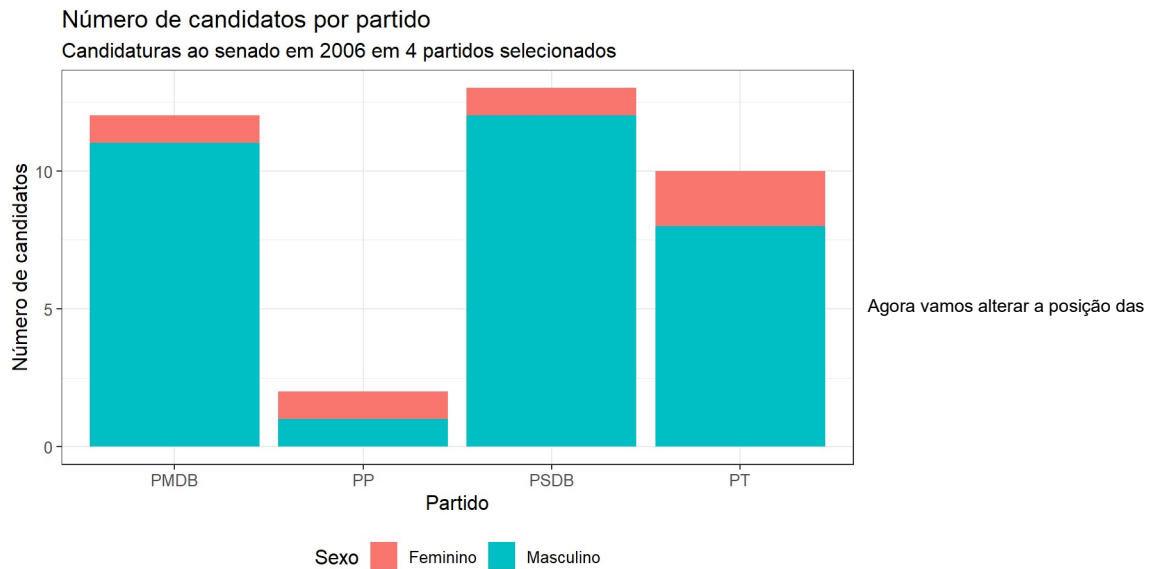
```
bar <- ggplot(subset(sen, partido == "PMDB" | partido == "PT" | partido == "PP" | partido == "PSDB"), aes(partido, cont, fill= partido))
bar + geom_bar(stat = "identity") + labs(title = "Número de candidatos por partido", subtitle = "Candidaturas ao senado em 2006 em 4 partidos selecionados", x = "Partido", y = "Número de candidatos", caption = "Fonte = TSE") + theme_bw() + theme(text = element_text(size = 12), legend.position = "none") + scale_fill_grey()
```



adicionando nova variável ao gráfico de barras

é necessário alterar o mapa estético, veja abaixo:

```
bar <- ggplot(subset(sen, partido == "PMDB" | partido == "PT" | partido == "PP" | partido == "PSDB"), aes(partido, cont, fill= sexo))
bar + geom_bar(stat = "identity") + labs(title = "Número de candidatos por partido", subtitle = "Candidaturas ao senado em 2006 em 4 partidos selecionados", x = "Partido", y = "Número de candidatos", caption = "Fonte = TSE") + theme_bw() + theme(text = element_text(size = 12), legend.position = "bottom") + scale_fill_discrete(name = "Sexo")
```



Fonte = TSE

barra

ggplot não faz automático precisamos de rearranjos como os de baixo

```
bar1 <- aggregate(cont~partido+sexo, data= subset(sen, partido == "PMDB" | partido == "PT" | partido == "PP" | partido == "PSDB"), FUN = "sum")
```

alí faz o gráfico:

```
bar <- ggplot(bar1, aes(partido, cont, fill = sexo))
```

```
bar <- ggplot(bar1, aes(partido, cont, fill = sexo))
bar+ geom_bar(stat = "identity", position=position_dodge(preserve = "single")) + labs(title = "Número de candidatos por partido", subtitle = "Candidaturas ao senado em 2006 em 4 partidos seleccionados", x = "Partido", y= "Número de candidatos", caption = "Fonte = TSE") + theme_bw() + theme(text = element_text(size = 12), legend.position = "bottom") + scale_fill_discrete(name = "Sexo")
```



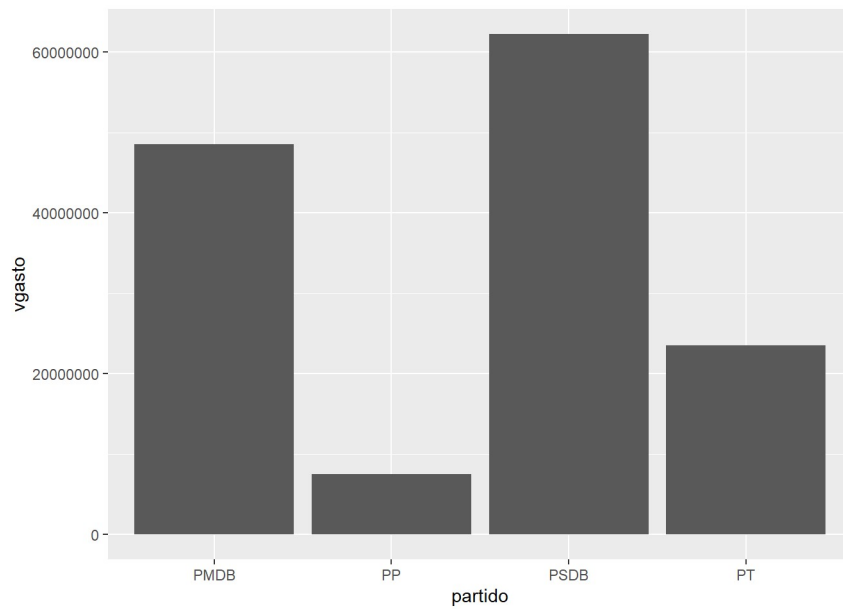
Fonte = TSE

esse single de cima resolveria o problema caso algum dos partidos tivesse zero em algum dos gêneros

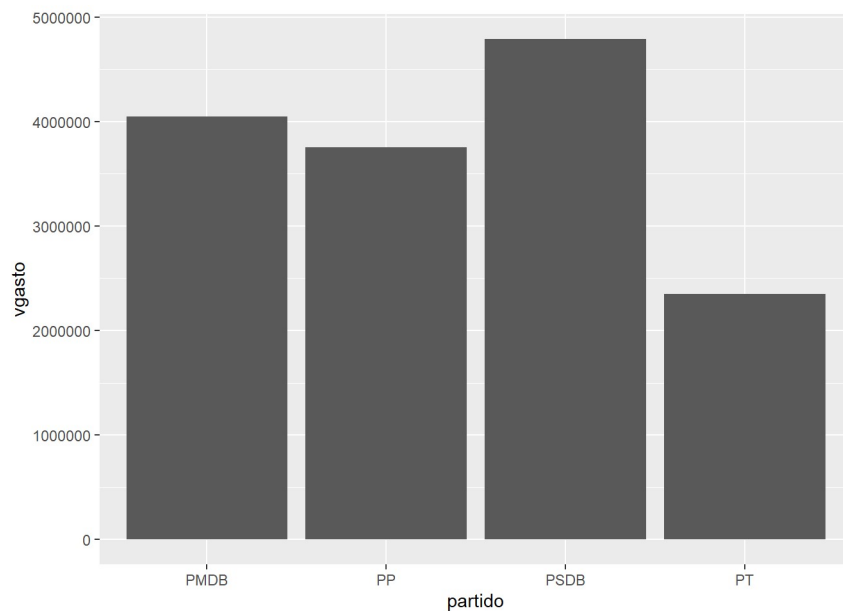
Agora plotar as médias

médias de gastos dos mesmos quatro partidos?

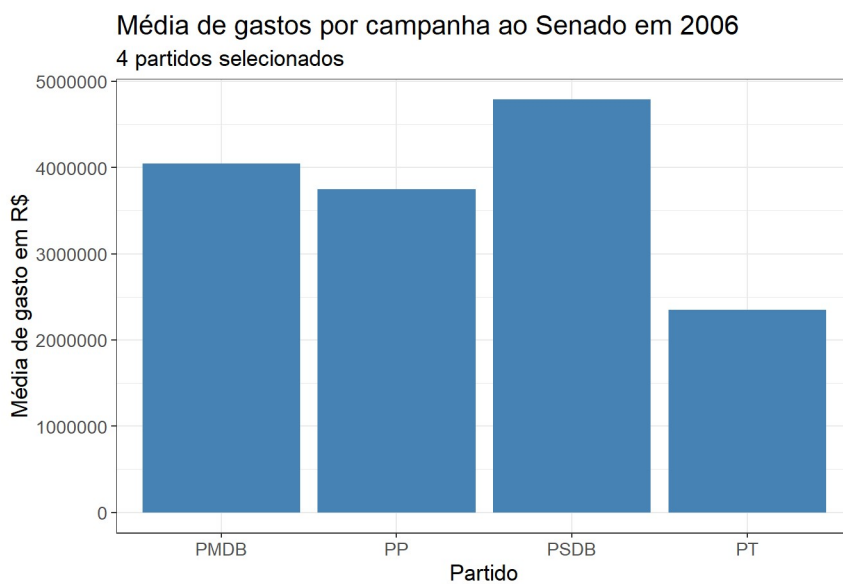
```
bar.m <- ggplot(subset(sen, partido == "PMDB" | partido == "PT" | partido == "PP" | partido == "PSDB"), aes(partido, vgasto))
bar.m + geom_bar(stat = "identity") # plota a soma de gastos por categoria
```



```
bar.m + geom_bar(stat = "summary", fun = "mean") # plota a média por categoria
```



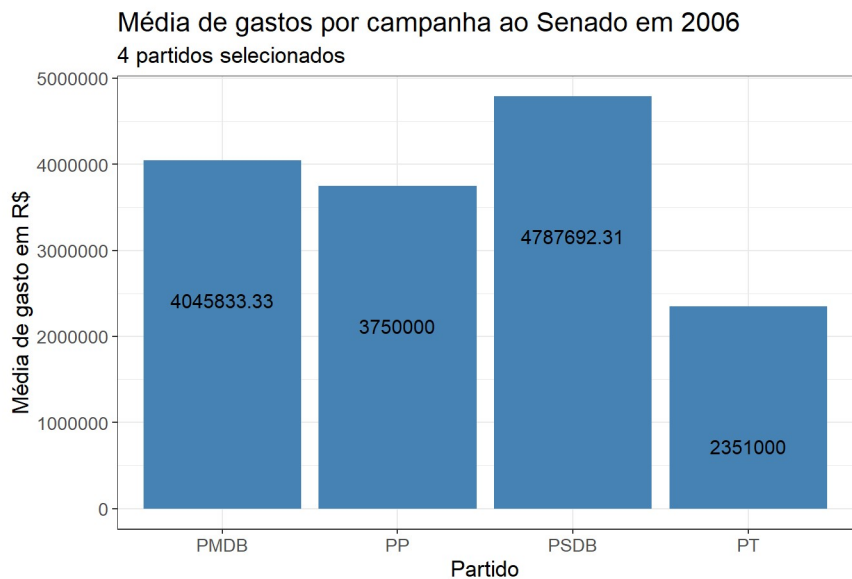
```
bar.m + geom_bar(stat = "summary", fun = "mean", fill = "steelblue") + labs(title = "Média de gastos por campanha ao Senado e  
m 2006", subtitle = "4 partidos selecionados", x = "Partido", y = "Média de gasto em R$", caption = "Fonte: TSE") + theme_bw  
( ) + theme(text = element_text(size = 13))
```



Fonte: TSE

Adicionando valor a cada barra.....

```
bar.m + geom_bar(stat = "summary", fun = "mean", fill = "steelblue") + stat_summary(aes(label=round(..y.., 2)), fun = mean,
geom = "text", size=4, vjust=10, color = "black") + labs(title = "Média de gastos por campanha ao Senado em 2006", subtitle =
"4 partidos selecionados", x = "Partido", y = "Média de gasto em R$", caption = "Fonte: TSE") + theme_bw() + theme(text = el
ement_text(size = 13))
```

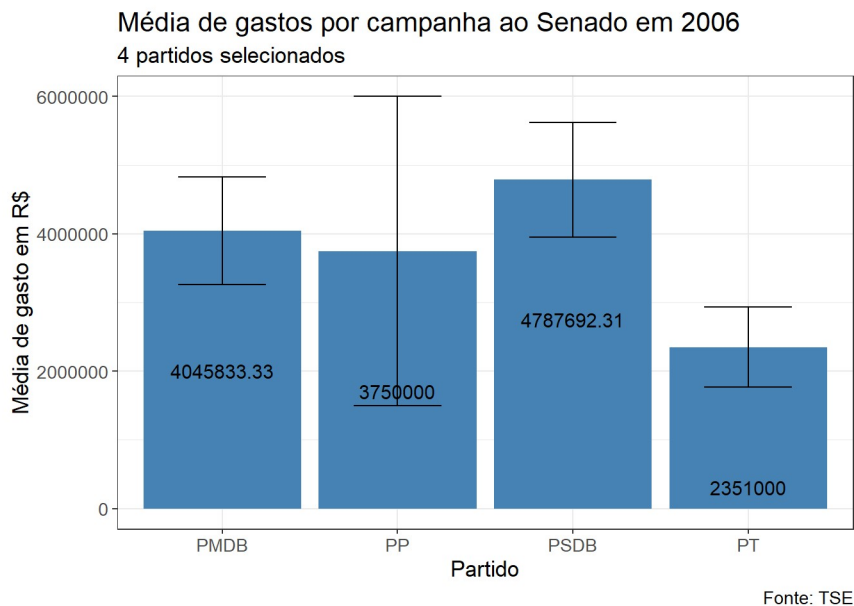


obs = y.., 2 valor de y com duas casa decimais size é tamanho vjust é a justificado

acima vê-se a média de cada partido

Adicionando barra de erro

```
bar.m + geom_bar(stat = "summary", fun = "mean", fill = "steelblue") + stat_summary(aes(label=round(..y.., 2)), fun = mean,
geom = "text", size=4, vjust=10, color = "black") + stat_summary(fun.data = mean_se, geom="errorbar", width = 0.5) + labs(titl
e = "Média de gastos por campanha ao Senado em 2006", subtitle = "4 partidos selecionados", x = "Partido", y = "Média de gas
to em R$", caption = "Fonte: TSE") + theme_bw() + theme(text = element_text(size = 13))
```



```
bar.m + geom_bar(stat = "summary", fun = "mean", fill = "steelblue") + stat_summary(fun.data = mean_se, geom="errorbar", wid
th = 0.5) + labs(title = "Média de gastos por campanha ao Senado em 2006", subtitle = "4 partidos selecionados", x = "Partid
o", y = "Média de gasto em R$", caption = "Fonte: TSE") + theme_bw() + theme(text = element_text(size = 13))
```

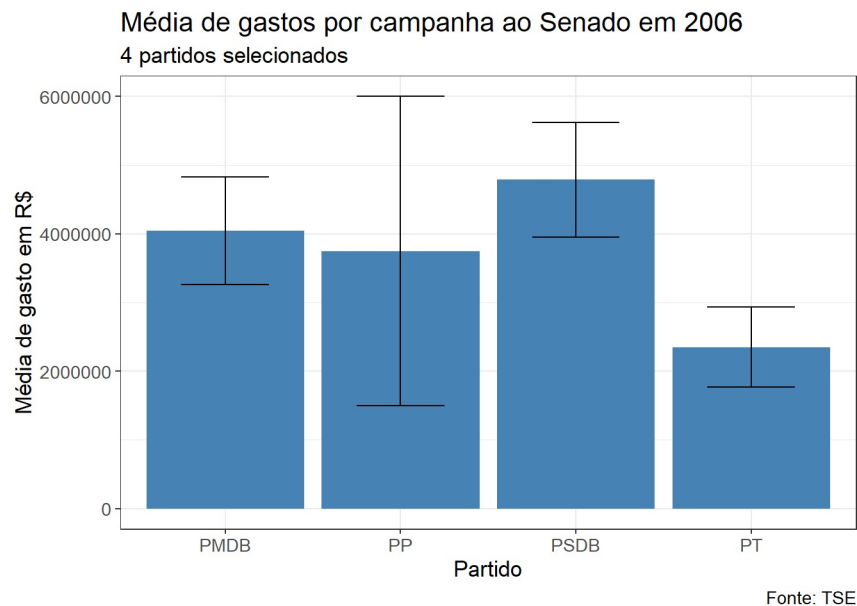


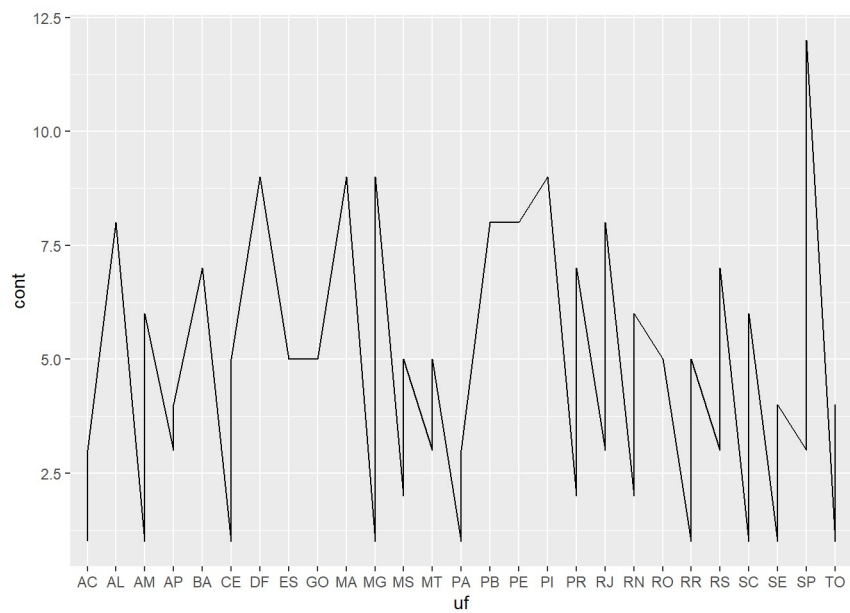
gráfico de linha

séries temporais + recomendado

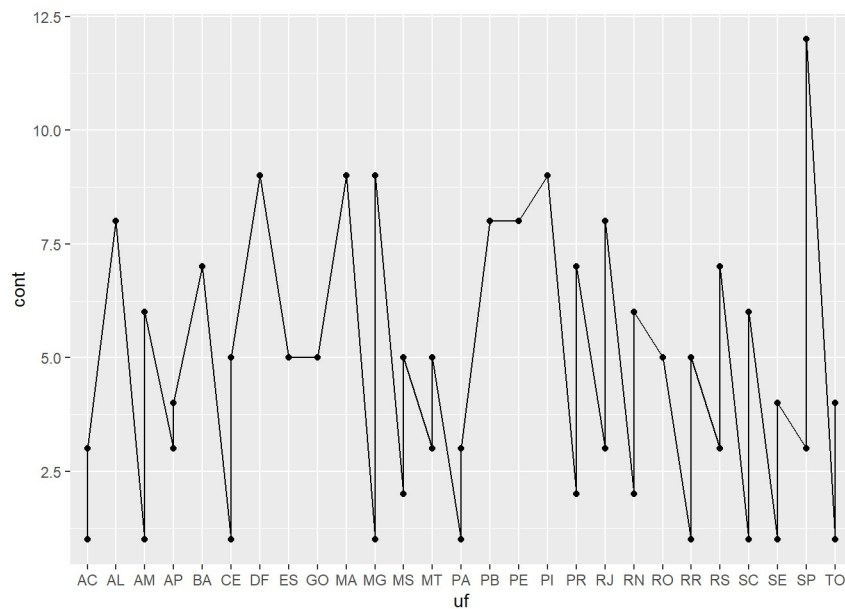
Aqui ele vai usar por estado

```
lin1 <- aggregate (cont~sexo+uf, data = sen, FUN="sum") # preparação dos dados
```

```
lin <- ggplot(lin1, aes(uf, cont, group = 1))
lin+geom_line()
```

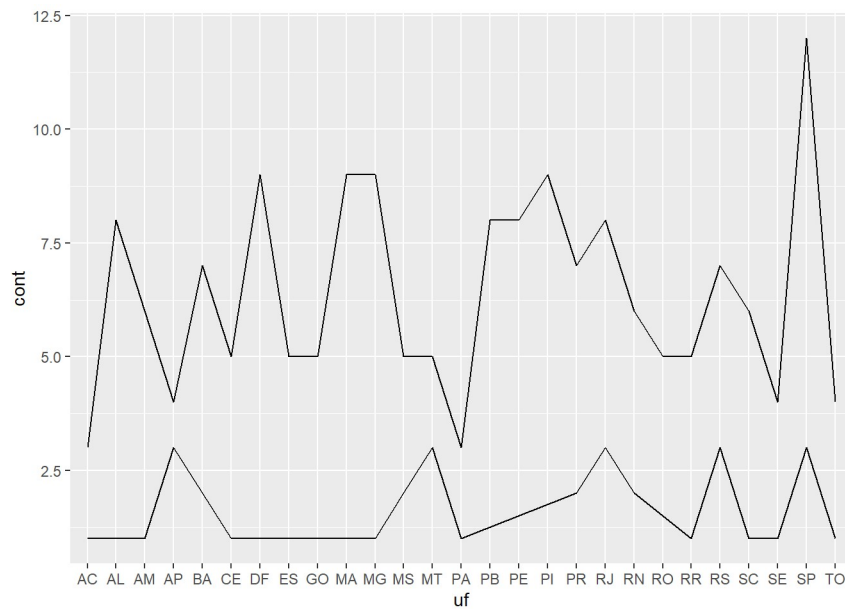


```
lin + geom_line() + geom_point()
```



adicionando duas linhas ao gráfico

```
lin <- ggplot(lin1, aes(uf, cont, group = sexo))
lin + geom_line()
```

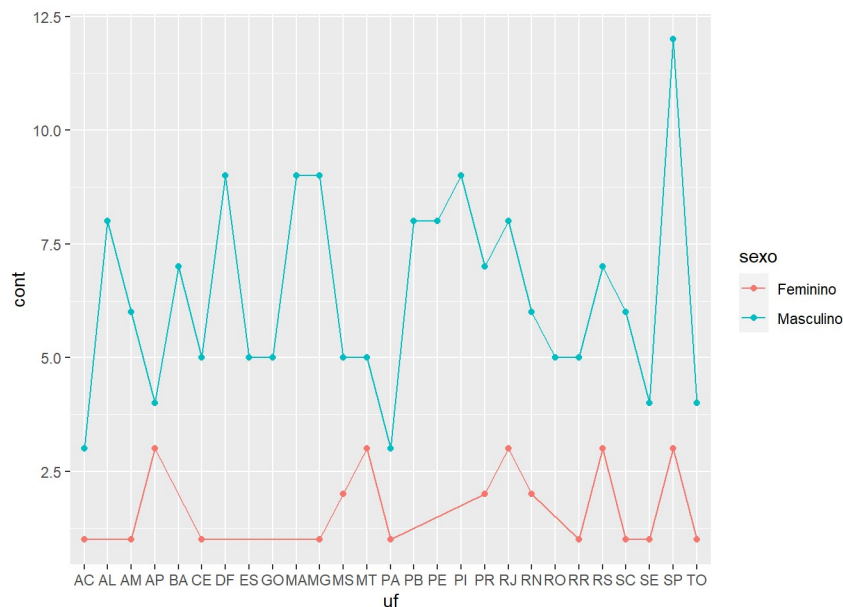


show tem q melhorar, mas tá no

caminho..

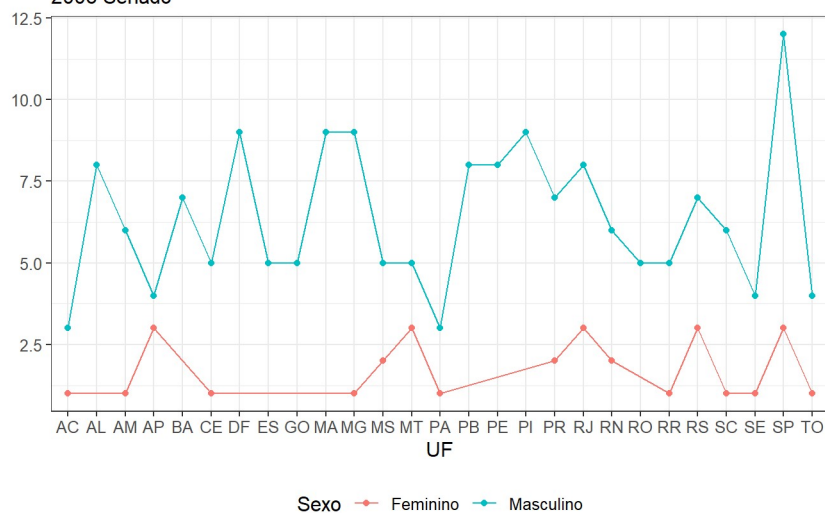
identificando as linhas do gráfico

```
lin <- ggplot(lin1, aes(uf, cont, group = sexo))
lin + geom_line(aes(colour = sexo)) + geom_point(aes(colour= sexo))
```



```
lin <- ggplot(lin1, aes(uf, cont, group = sexo))
lin + geom_line(aes(colour = sexo)) + geom_point(aes(colour= sexo)) + labs(title="Total de candidaturas por sexo e estado brasileiro", subtitle="2006 Senado", x = "UF", y = "", caption = "Fonte: TSE", color = "Sexo") + theme_bw() + theme(text = element_text(size = 12), legend.position = "bottom")
```

Total de candidaturas por sexo e estado brasileiro
2006 Senado



Fonte: TSE

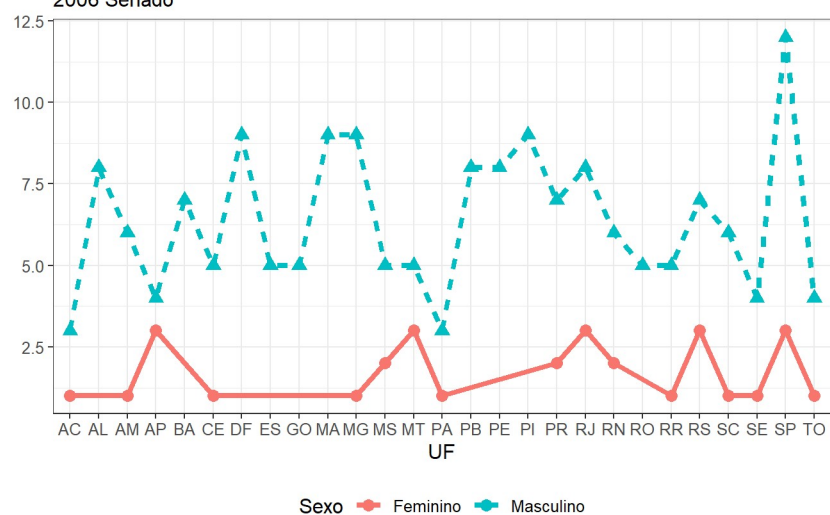
Abaixo Alternando a espessura da linha e ponto

é esse size, linetype e shape depois do argumento em 'geom_line' e em 'geom_point'. Também se relaciona com o último argumento 'guides'.
(importante)

```
lin <- ggplot(lin1, aes(uf, cont, group = sexo))
lin + geom_line(aes(colour = sexo, linetype = sexo), size = 1.5) + geom_point(aes(colour= sexo, shape = sexo), size = 3) + labs(title="Total de candidaturas por sexo e estado brasileiro", subtitle="2006 Senado", x = "UF", y = "", caption = "Fonte: TSE", color = "Sexo") + theme_bw() + theme(text = element_text(size = 12), legend.position = "bottom") + guides(linetype = F, shape = F)
```

```
## Warning: `guides(<scale> = FALSE)` is deprecated. Please use `guides(<scale> =`  
## "none")` instead.
```

Total de candidaturas por sexo e estado brasileiro
2006 Senado



Fonte: TSE