

Introdução ao R

Willber Nascimento

2022-06-23

Aula 5 - Personalização de gráficos com `ggplot2`

Rótulos dos eixos, títulos e notas

Uma parte central de um gráfico estatístico são seus rótulos. Dois deles são essenciais: título do gráfico e rótulo ou título dos eixos. Além disso, é comum a apresentação de notas embaixo do gráfico. Abaixo veremos como inserir essas informações.

Para os próximos exemplos, usaremos o banco de dados de eleições disponibilizado por Daniel Sampaio¹. O código abaixo carrega um banco de dados no formato .sav usando a função `read.spss` do pacote `foreign`.

```
library(foreign)

###  
eleicoes <- read.spss('..../dados/Banco de dados da dissertacao - vf.sav', to.data.frame = T, reencode = 'T')  
  
## re-encoding from UTF-8  
  
str(eleicoes)  
  
## 'data.frame': 14958 obs. of 57 variables:  
## $ SequencialCandidato: num 1.2e+11 1.2e+11 2.1e+11 2.1e+11 2.1e+11 ...  
## $ UF : chr "MS" "MS" "RS" "RS" ...  
## $ SigladaUE : num 55050 55050 89540 89540 89540 ...  
## $ Sigla_ibge : num 5006275 5006275 4314548 4314548 4314548 ...  
## $ NomedaUE : chr "PARAÍSO DAS ÁGUAS" "PARAÍSO DAS ÁGUAS" "PI"  
## $ Eleitores_na_ue : num 4082 4082 2297 2297 2297 ...  
## $ categ_ue : num 1 1 1 1 1 1 1 1 1 ...  
## $ SiglaPartido : chr "PMDB" "PTB" "PMDB" "PDT" "..."  
## $ Cargo : chr "Prefeito" "Prefeito" "Prefeito" "Prefeito" ...  
## $ Nomecandidato : chr "IVAN DA CRUZ PEREIRA" "JOÃO C"  
## $ CPFdocandidato : num 5.62e+10 5.63e+10 3.12e+10 6.63e+10 8.96e+10 ...  
## $ Modernos_cand : num 10173 46242 4400 7785 0 ...  
## $ Organização_cand : num 3170 1973 0 0 0 ...  
## $ Outros_cand : num 13770 3187.1 88.7 0 0 ...  
## $ Tradicional_cand : num 69457 43542 10015 6953 400 ...  
## $ Soma_gastos_cand : num 96570 94943 14504 14738 400 ...  
## $ Soma_Trad_ue : num 112999 112999 17368 17368 17368 ...  
## $ Soma_Mod_ue : num 56415 56415 12185 12185 12185 ...
```

¹<https://osf.io/c8dsm/>

```

## $ Soma_Trad_Mod      : num 169413 169413 29553 29553 29553 ...
## $ vi_trad            : num 0.615 0.385 0.577 0.4 0.023 ...
## $ vi_mod              : num 0.18 0.82 0.361 0.639 0 ...
## $ VotosValidos_na_ue : num 3189 3189 2123 2123 2123 ...
## $ Votos_cand1         : num 2124 1065 1409 602 112 ...
## $ vd_votos            : num 0.666 0.334 0.6637 0.2836 0.0528 ...
## $ sit_tse              : chr "APTO" "APTO" "APTO" "APTO" ...
## $ desc_sit             : chr "DEFERIDO" "DEFERIDO" "DEFERIDO" ...
## $ Sit_geral            : chr "ELEITO" "NÃO ELEITO" "ELEITO" "NÃO ELEITO" ...
## $ Sit_1t               : chr "ELEITO" "NÃO ELEITO" "ELEITO" "NÃO ELEITO" ...
## $ cod_sucesso          : num 1 0 1 0 0 1 0 0 0 1 ...
## $ Sexo                 : chr "MASCULINO" "MASCULINO" "MASCULINO" "MASCULINO" ...
## $ cod_sexo              : num 0 0 0 0 0 1 0 0 0 ...
## $ grau_instrucao       : chr "ENSINO MÉDIO COMPLETO" "ENSINO MÉDIO COMPLETO" ...
## $ cod_grau_inst         : num 0 0 1 1 0 0 0 1 0 0 ...
## $ cor                  : chr "BRANCA" "BRANCA" "BRANCA" "BRANCA" ...
## $ cod_cor               : num 1 1 1 1 1 1 1 0 1 ...
## $ Reeleição             : chr "S" "N" "N" "S" ...
## $ cod_reeleicao         : num 1 0 0 1 0 0 0 0 0 0 ...
## $ N_cand_mun           : num 2 2 3 3 3 4 4 4 4 2 ...
## $ IC                   : num 0.5 0.5 1 1 1 1.5 1.5 1.5 1.5 0.5 ...
## $ IDHM                 : num NA NA NA NA 0.854 0.854 0.854 0.854 0.806 ...
## $ ln_vd_votos           : num -0.406 -1.097 -0.41 -1.26 -2.942 ...
## $ ln_vi_trad            : num -0.487 -0.954 -0.551 -0.915 -3.771 ...
## $ ln_vi_mod              : num -1.713 -0.199 -1.019 -0.448 NA ...
## $ Zln_vi_mod            : num -0.715 0.9519 0.0495 0.6776 NA ...
## $ Zln_vi_trad           : num 0.6983 0.0851 0.6145 0.1352 -3.6145 ...
## $ ln_n_eleitores         : num 8.31 8.31 7.74 7.74 7.74 ...
## $ Zln_n_eleitores        : num 0.725 0.725 -1.144 -1.144 -1.144 ...
## $ ln_idhm               : num NA NA NA NA ...
## $ ZIC                   : num -0.632 -0.632 0.718 0.718 0.718 ...
## $ Zln_idhm              : num NA NA NA NA ...
## $ Zcod_cor               : num 0.631 0.631 0.631 0.631 0.631 ...
## $ Zcod_sucesso           : num 1.15 -0.87 1.15 -0.87 -0.87 ...
## $ Zcod_grau_inst          : num -0.93 -0.93 1.07 1.07 -0.93 ...
## $ Zcod_reeleicao          : num 1.878 -0.532 -0.532 1.878 -0.532 ...
## $ Zcod_sexo               : num -0.408 -0.408 -0.408 -0.408 -0.408 ...
## $ Interacao_trad         : num 0.5062 0.0617 -0.703 -0.1547 4.1355 ...
## $ Interacao_mod            : num -0.5183 0.69 -0.0566 -0.7753 NA ...
## - attr(*, "variable.labels")= Named chr [1:57] "Sequencial Candidato" "UF" "UE TSE" "UE IBGE" ...
## ..- attr(*, "names")= chr [1:57] "SequencialCandidato" "UF" "SigladaUE" "Sigla_ibge" ...
## - attr(*, "codepage")= int 65001

```

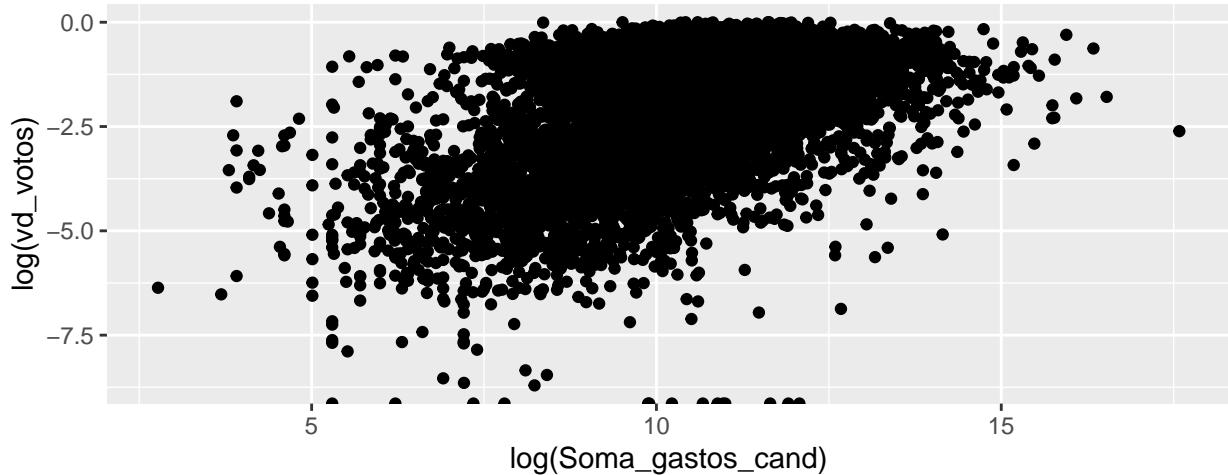
Primeiro, criamos um gráfico com ggplot. Esse gráfico, plt_gasto_votos, é um gráfico de dispersão entre gastos de campanha e votos de prefeitos. As duas variáveis foram transformadas em seu logarítmico natural para ajustar distribuições assimétricas.

```

plt_gasto_votos <- ggplot(eleicoes, aes(log(Soma_gastos_cand), log(vd_votos))) +
  geom_point()

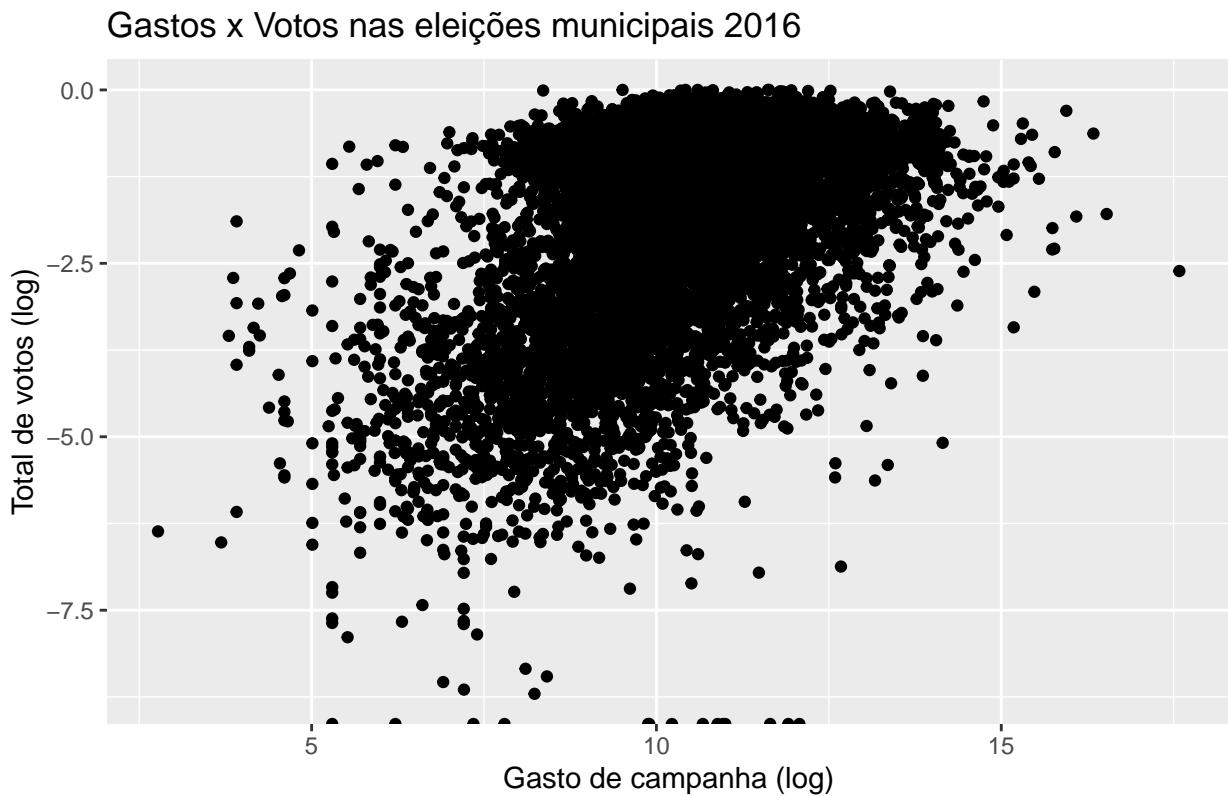
plt_gasto_votos

```



Como você ver, o gráfico não possui título e usa o nome das variáveis como estão no banco de dados. Em ambos os casos é uma prática ruim. As informações apresentadas em gráfico precisam ser bem identificadas com rótulos apropriados. com o ggplot há diversas funções que permitem alterar título e rótulos dos eixos, tais como `xlab`, `ylab`, `ggtitle` e outros. Contudo, a forma mais prática é usar a função `labs` para alterar essas informações em uma única chamada.

```
plt_gasto_votos +
  labs(x='Gasto de campanha (log)', y='Total de votos (log)', title = 'Gastos x Votos nas eleições municipais 2016')
```



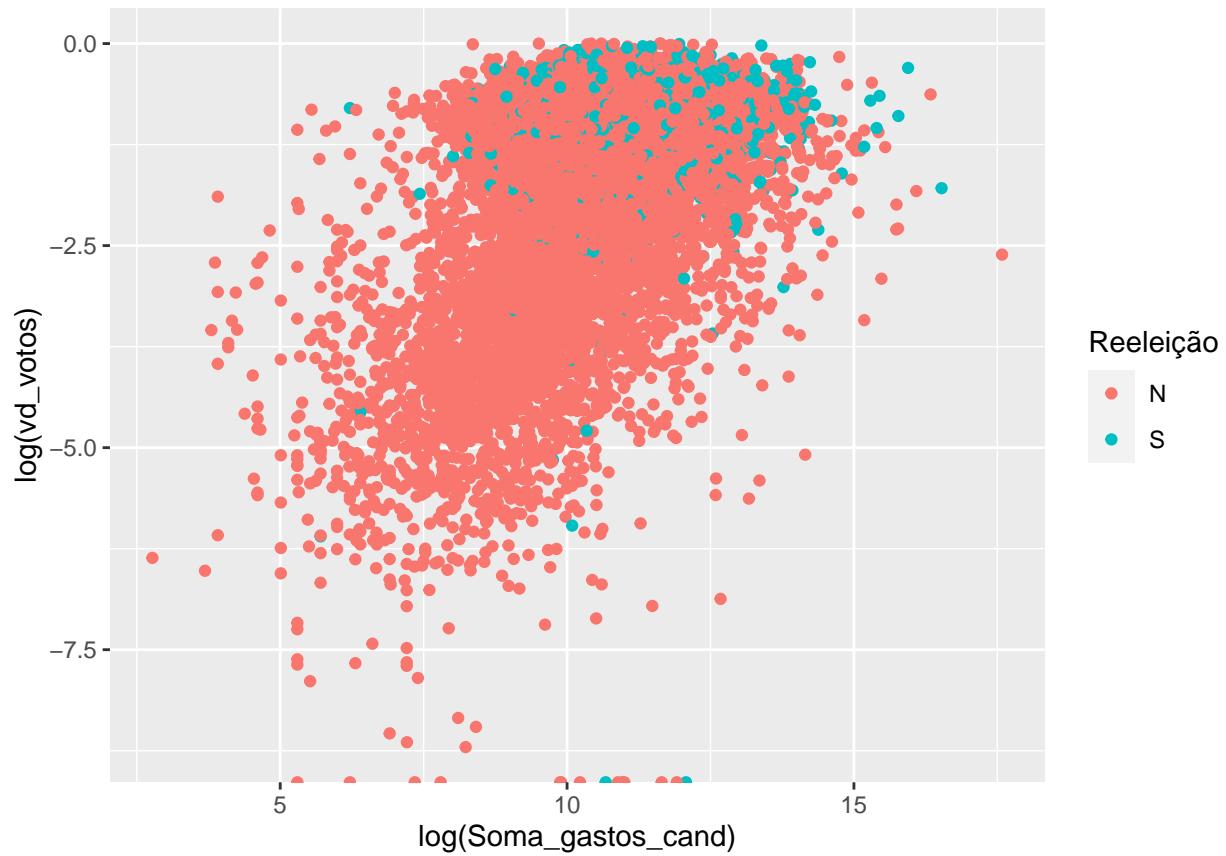
Elaboração: Willber Nascimento
Dados: Daniel Sampaio.

Com apenas uma chamada de `labs` inserimos um título informativo, bem como rótulos aos eixos e uma pequena nota sobre a elaboração e fonte dos dados. Mesmo sem um texto explicativo, esse gráfico poderia ser entendido de maneira muito fácil.

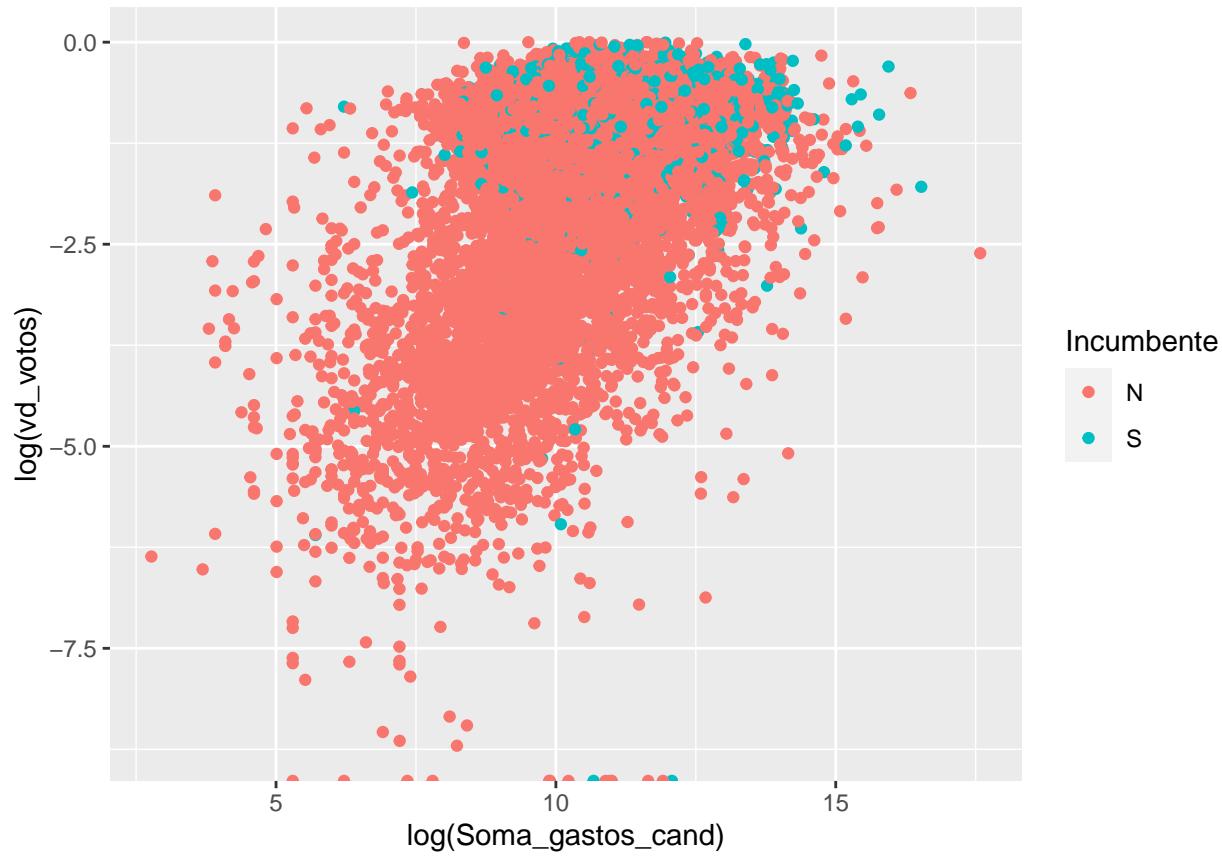
Rótulos de legendas Da mesma forma, é possível alterar o rótulo das legendas no ggplot usando a função `labs`. Vejamos o mesmo gráfico, mas agrupando por tipo de candidato.

```
gastos_votos_incumbentes <- ggplot(eleicoes, aes(log(Soma_gastos_cand), log(vd_votos), group=Reeleição))  
geom_point()
```

```
gastos_votos_incumbentes
```



```
gastos_votos_incumbentes +  
  labs(color = 'Incumbente')
```



Rótulo de dados

Para adicionarmos rótulos nos dados, podemos usar o argumento `label` diretamente na estética principal e indicar uma variável com os rótulos. Vamos carregar um novo banco de dados, `eleicoes-agregado.rds`, que contém informações agregadas dos partidos nas eleicoes municipais.

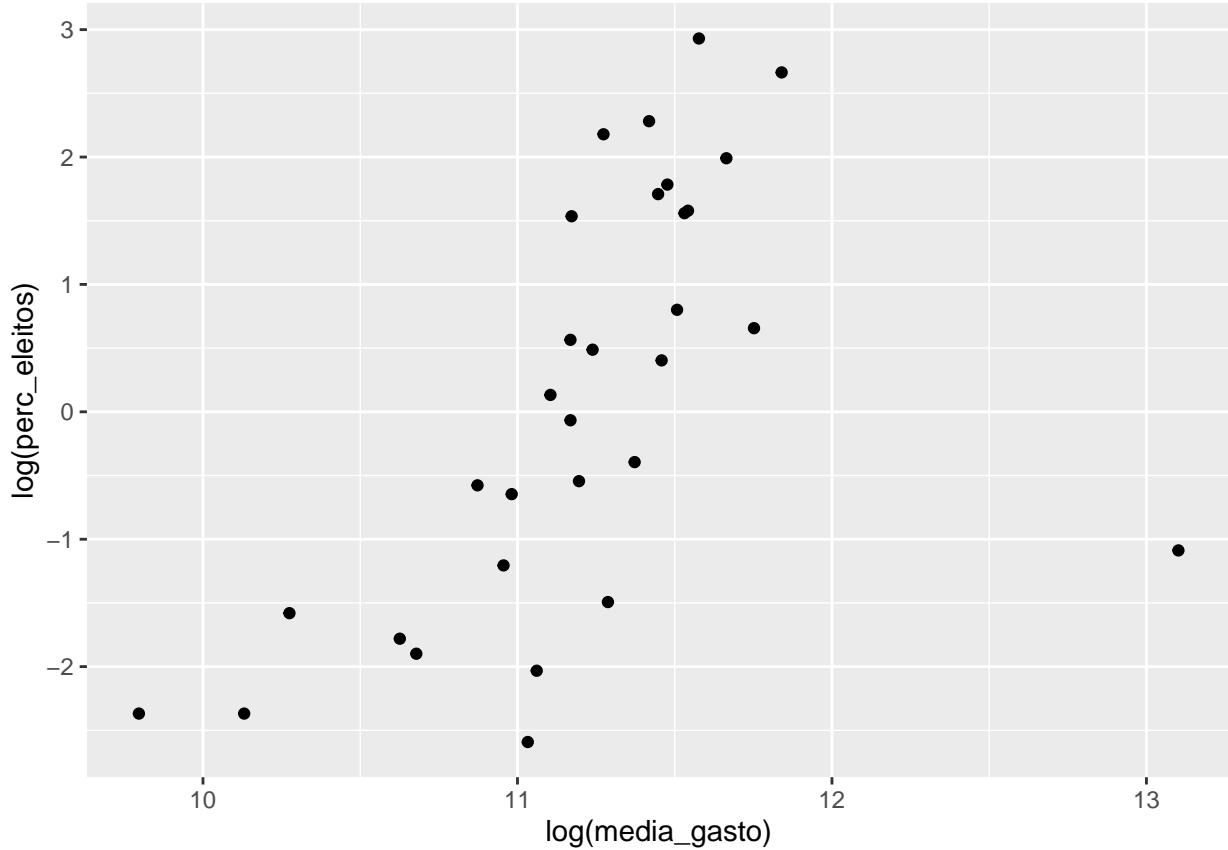
```
eleicoes_agg <- readRDS('..../dados/eleicoes-agregado.rds')
```

```
head(eleicoes_agg)
```

```
##   Sigla Partido cod_sucesso   n perc_eleitos media_gasto media_votos
## 1   DEM      1 259  4.84837140 102974.81  0.3939817
## 2  PC do B   1  80  1.49756645  94690.88  0.2969439
## 3    PDT      1 318  5.95282666  96439.80  0.3797812
## 4    PEN      1  11  0.20591539  28994.98  0.1285276
## 5    PHS      1  36  0.67390490  86907.87  0.2368827
## 6    PMB      1   4  0.07487832  61868.61  0.1566741
##   perc_eleitos_feminino sucesso_feminino
## 1          4.9441786  Acima da media
## 2          1.5948963  Abaixo da Media
## 3          5.9011164  Acima da media
## 4          0.1594896  Abaixo da Media
## 5          0.1594896  Abaixo da Media
## 6          0.3189793  Abaixo da Media
```

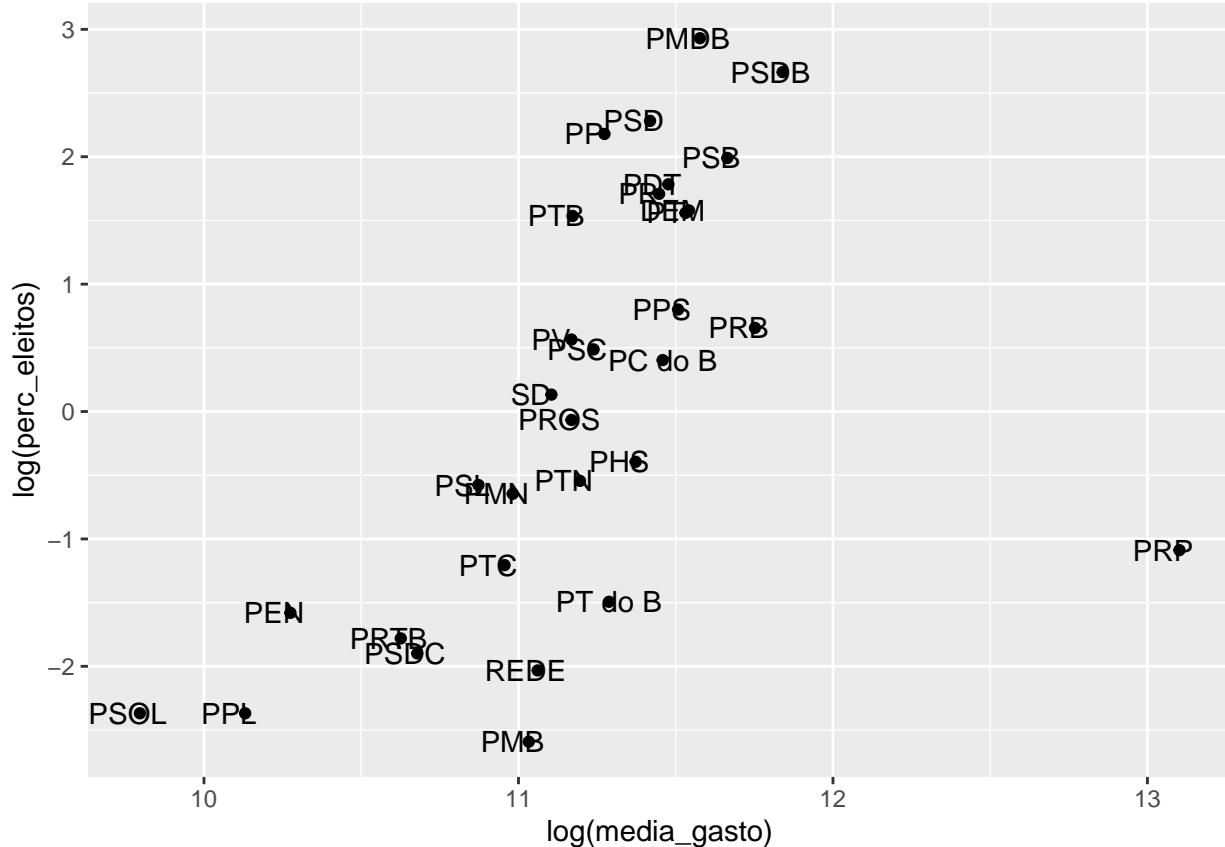
Com esses dados vamos plotar a relação entre gastos e votos e rotular os pontos com o nome dos partidos.

```
rot1 <-  
ggplot(eleicoes_agg, aes(log(media_gasto), log(perc_eleitos), label=SiglaPartido))+  
  geom_point()  
  
rot1
```



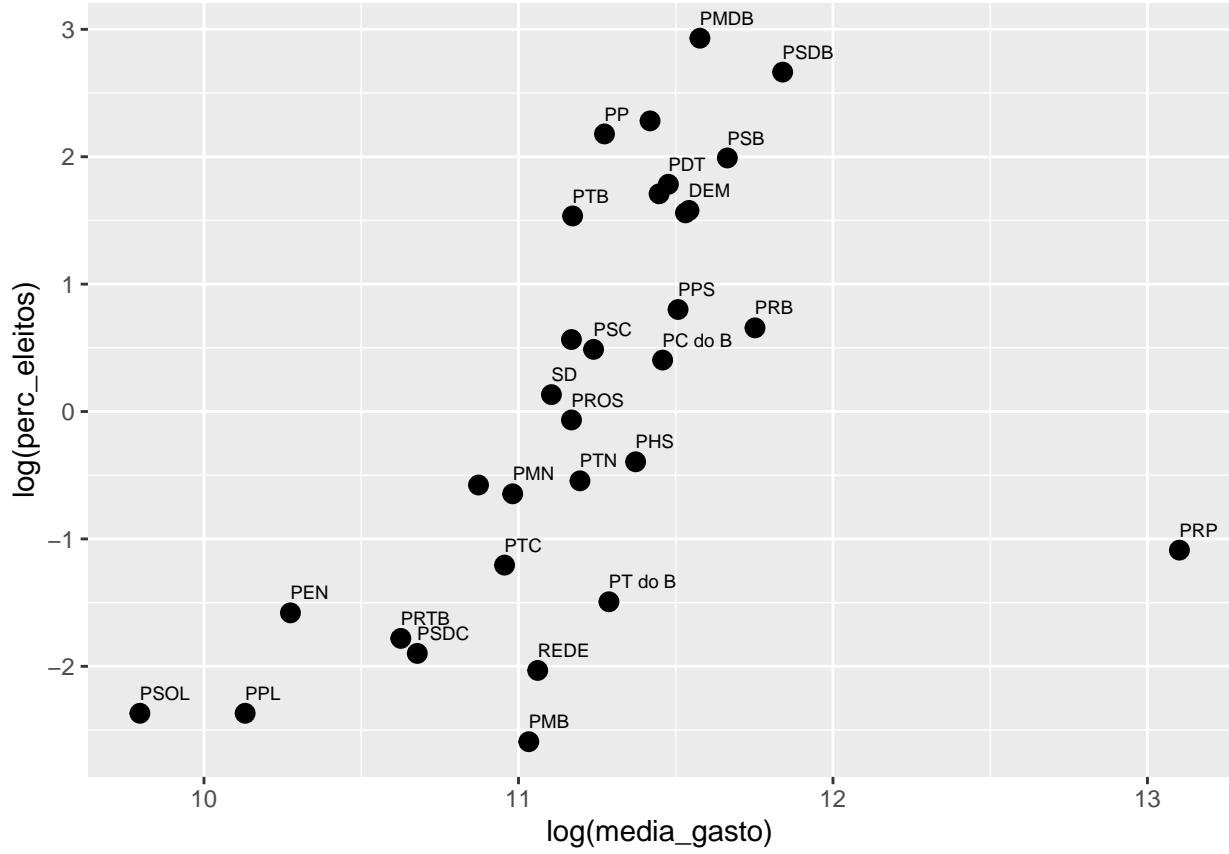
Contudo, como você ver, é necessário usar uma outra camada para que o rótulo indicado por `label` seja renderizado. Essa é a função do `geom_text`.

```
rot1 +  
  geom_text()
```



Como você ver, `label` indica a variável e `geom_text` o apresenta. Veja a documentação e perceba que você poderia ter passado `label` na estética de `geom_text` que é o local onde configuraremos tudo para que o texto seja apresentado da forma que desejamos. Veja que o texto é muito maior que os pontos e que eles o sobrepõem. Podemos editá-los.

```
rot1 + geom_point(size=3)+  
  geom_text(size=2.5, hjust=0, vjust=-1, check_overlap = TRUE)
```



Em primeiro lugar, aumentamos o tamanho dos pontos com o `size` dentro de `geom_point`, enquanto reduzimos o tamanho da fonte do label dentro de `geom_text` usando `size` também. Ajustamos a posição horizontalmente com `hjust` e verticalmente com `vjust`. Além disso, indicamos que deveria ser checado o overlap com `check_overlap` para que não haja sobreposição entre os textos, o que fez com que o partido à esquerda do PMN não tivesse seu rótulo apresentado.

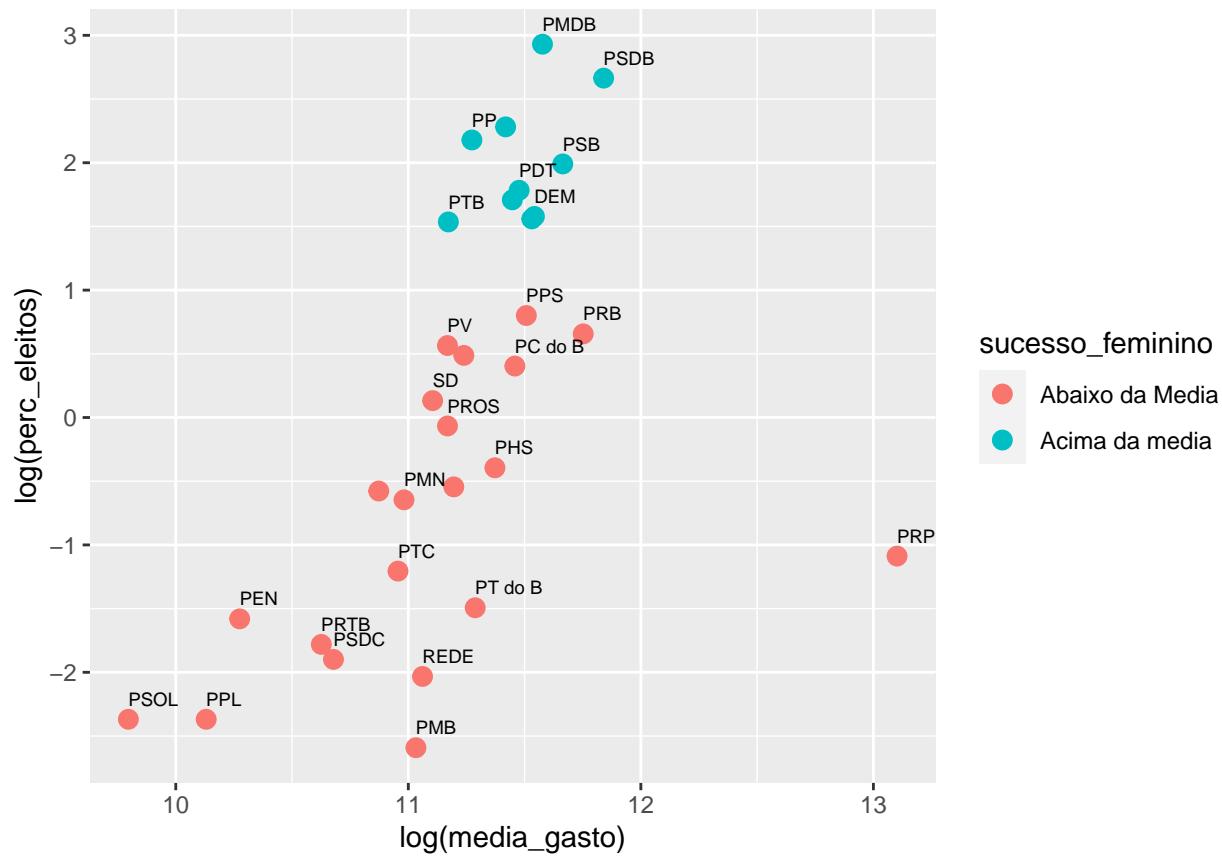
```

rot2 <-
rot1 +
  geom_point(size=3, aes(color=sucesso_feminino))+
  geom_text(size=2.5, hjust=0,vjust=-1, check_overlap = TRUE)

rot2

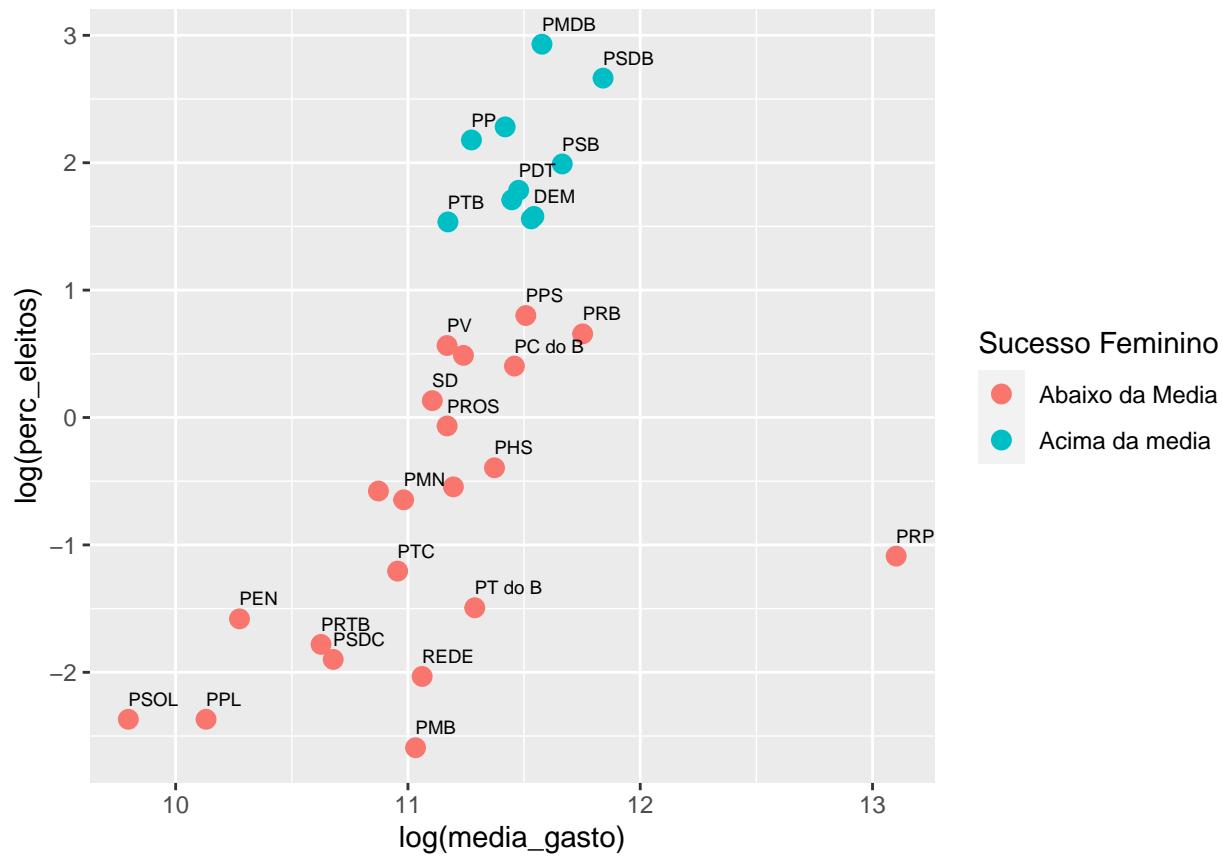
```

Usando categorias para rotular



usar cores para separar os grupos facilita consideravelmente a análise servindo como um tipo de rótulo. Para renomearmos a legenda podemos usar `labs` novamente e indicar o novo rótulo para o elemento.

```
rot2 +
  labs(color='Sucesso Feminino')
```

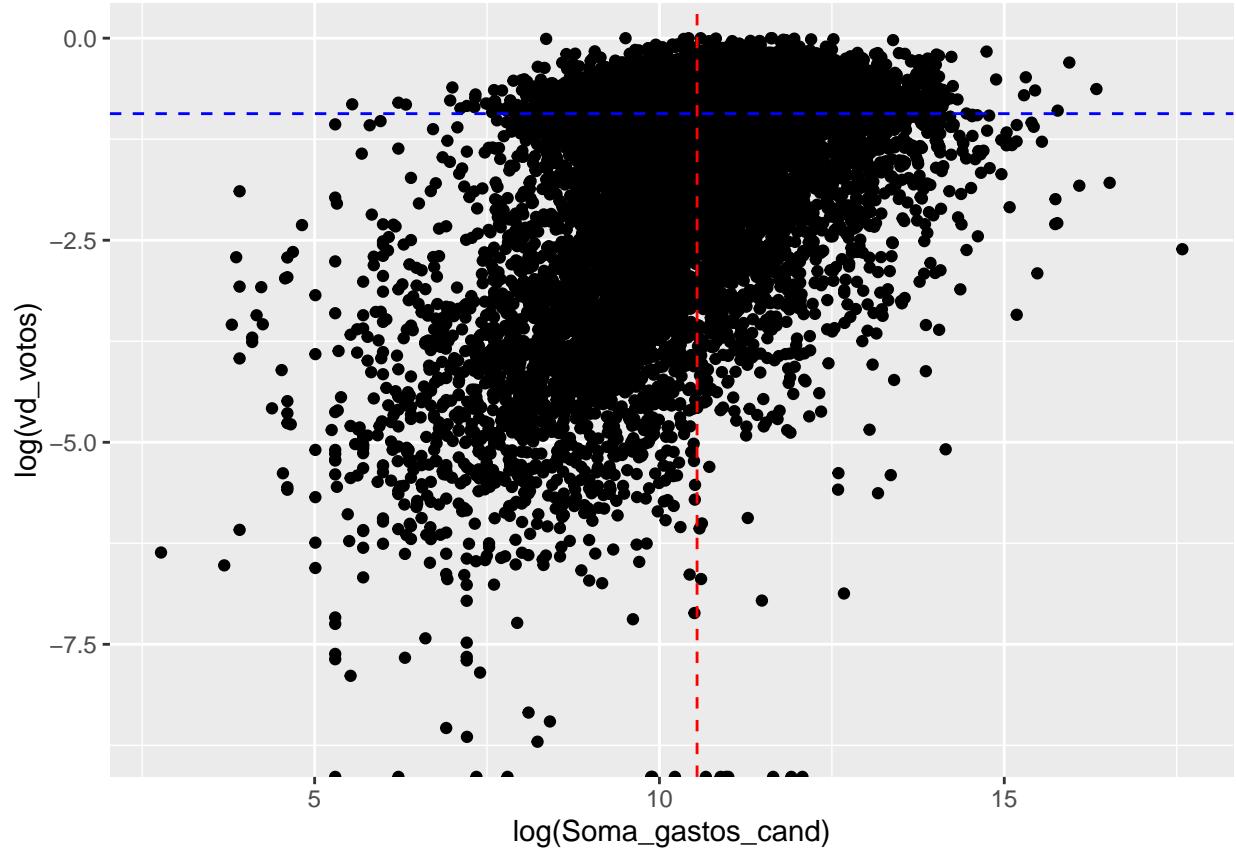


Linhas de referência vertical e horizontal, reta de ajuste

Frequentemente queremos adicionar informações comparativas nos gráficos ou mesmo um tipo de ajuste estatístico. Para isso, podemos usar `geom_hline`, `geom_vline` e `geom_smooth`.

```
novo_plot <-
plt_gasto_votos +
  geom_hline(yintercept = mean(log(eleicoes$vd_votos + 0.1)), linetype='dashed', color='blue') +
  geom_vline(xintercept = mean(log(eleicoes$Soma_gastos_cand + 0.1)), linetype='dashed', color='red')

novo_plot
```

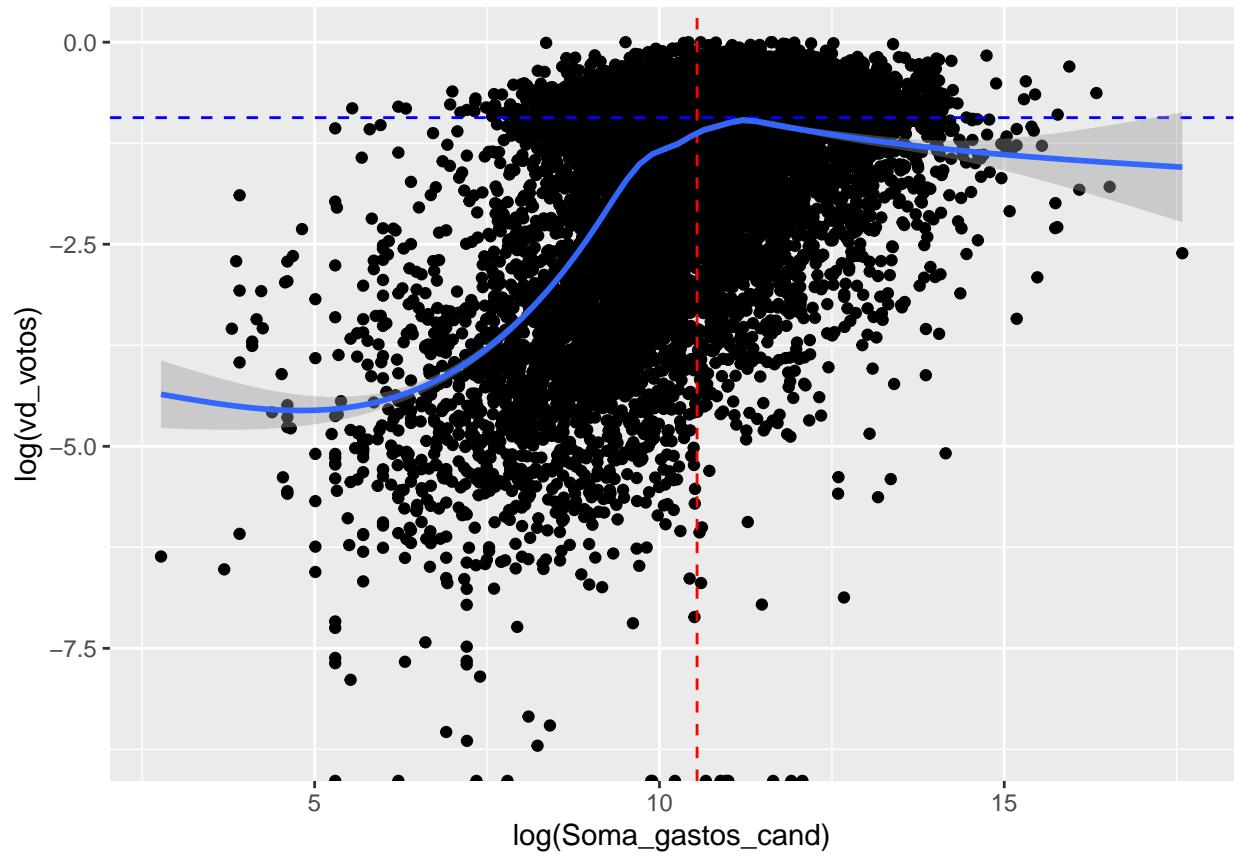


Como você pode imaginar, `vline` é uma abreviação para uma linha vertical e `hline` para uma horizontal. Além disso, linhas possuem atributos que podem ser alterados em uma linguagem consistente, por isso usamos `linetype` para alterar o tipo da linha e `color` para alterar sua cor. Seria possível também alterar o tamanho usando a opção `size`. Outra linha referência são os ajustes estatísticos. Neles podemos plotar os dados e compará-lo a um modelo específico, como no gráfico abaixo.

```
novo_plot +
  geom_smooth()

## `geom_smooth()` using method = 'gam' and formula 'y ~ s(x, bs = "cs")'

## Warning: Removed 14 rows containing non-finite values (stat_smooth).
```

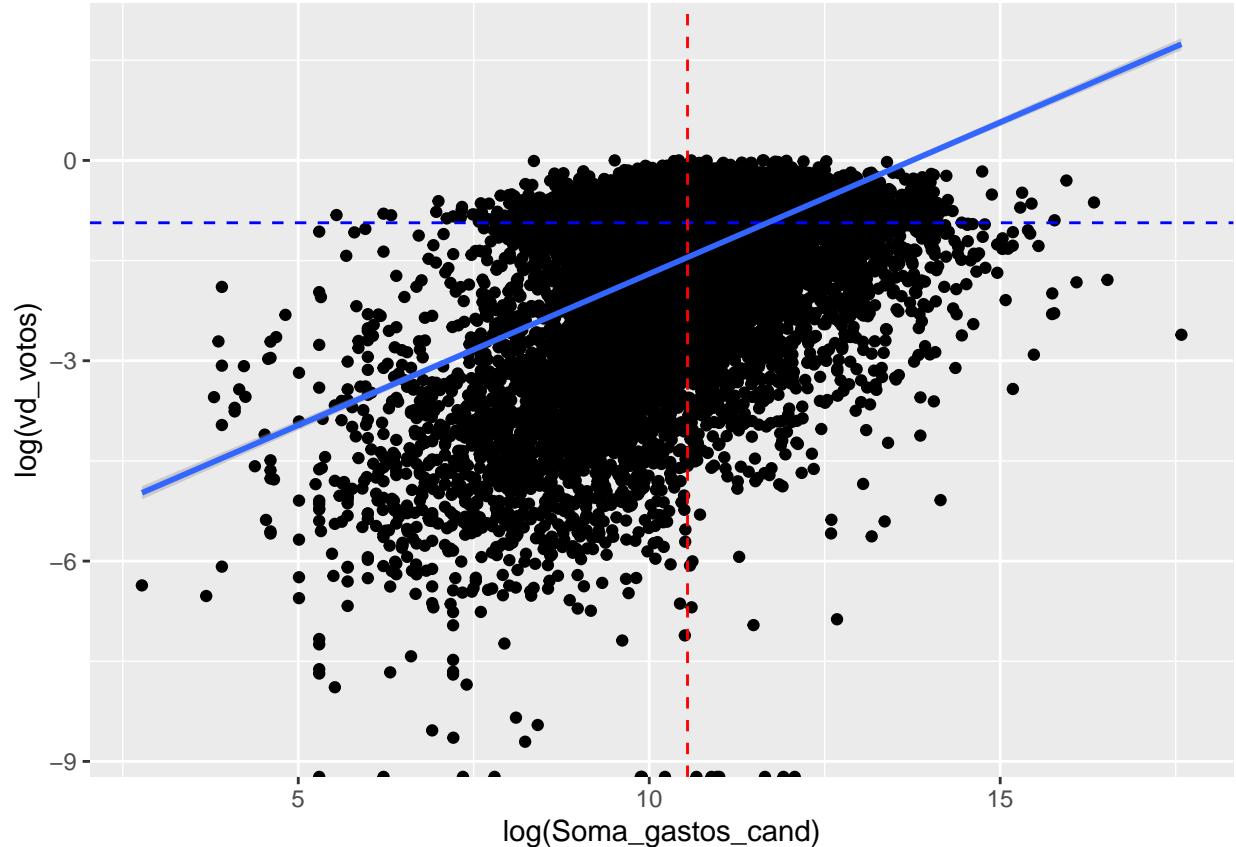


Podemos alterar o modelo pré-definido por um linear.

```
novo_plot +
  geom_smooth(method = 'lm')

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

## Warning: Removed 14 rows containing non-finite values (stat_smooth).
```

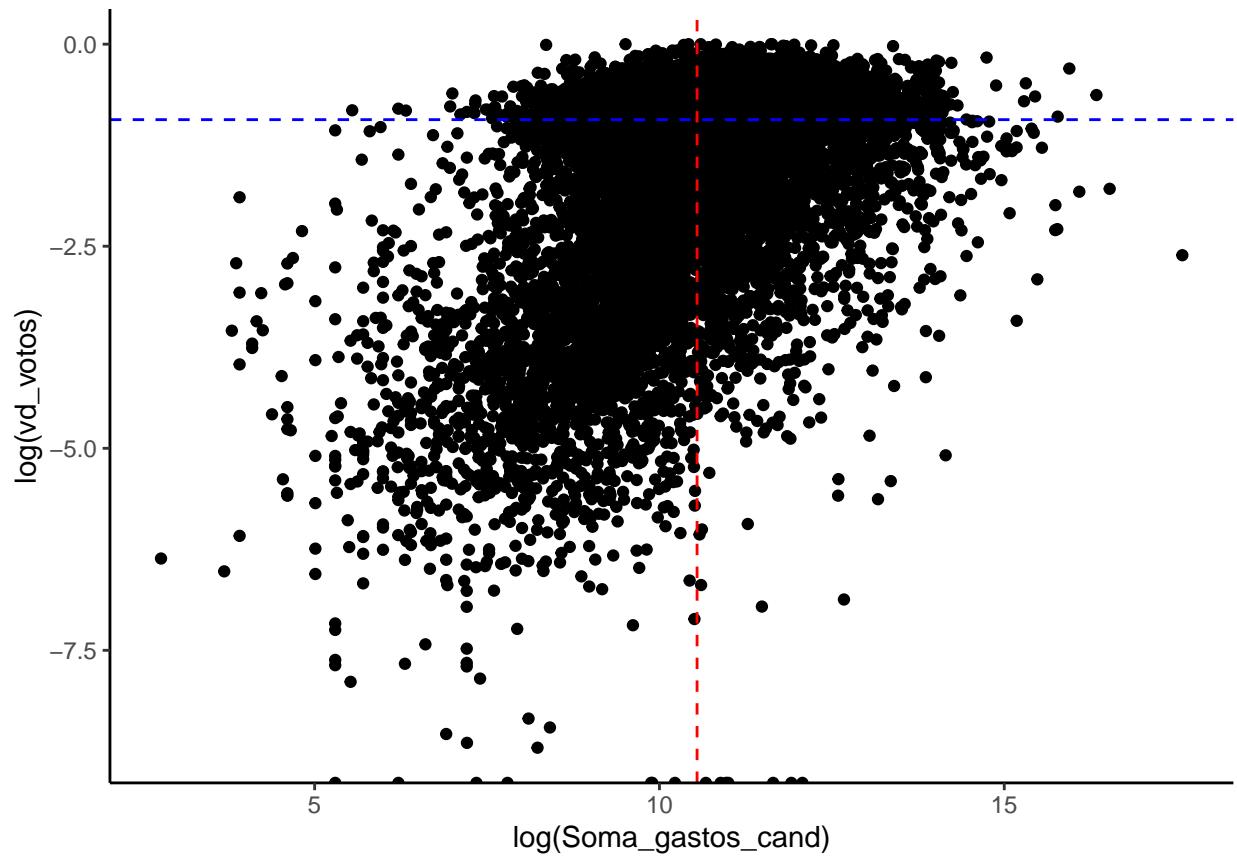


`geom_smooth` possui um argumento `method` que recebe uma string indicando o método a ser calculado com base nas variáveis passada na estética. Contudo, ele é flexível o suficiente para receber modelos personalizados usando `formula`.

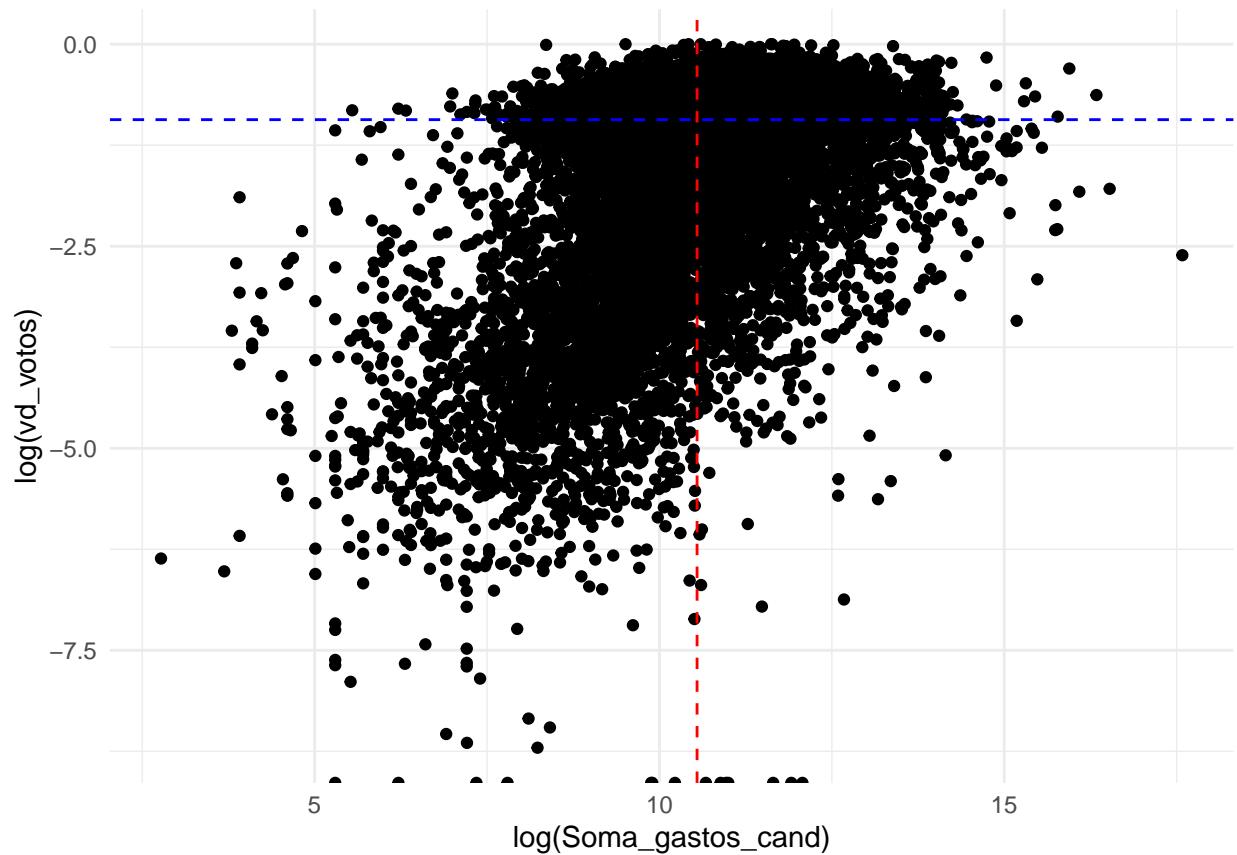
Temas

É possível alterar a aparência dos gráficos ggplots de uma maneira bastante simples usando temas. Normalmente você vai preferir usar um pronto e disponível gratuitamente na internet. Contudo, você também pode criar seus próprios temas. Primeiro, vamos reutilizar o gráfico anterior

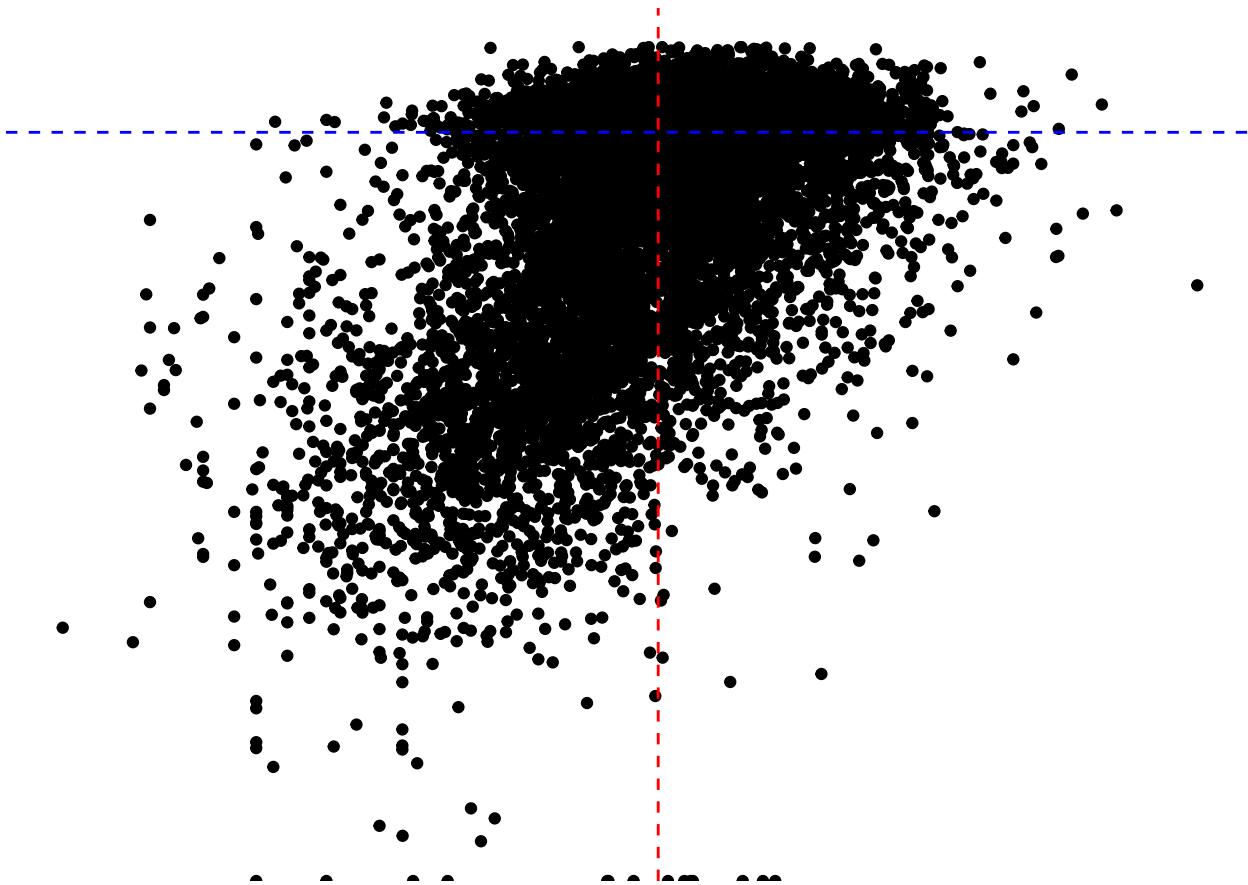
```
novo_plot +
  theme_classic()
```



```
novo_plot +  
  theme_minimal()
```

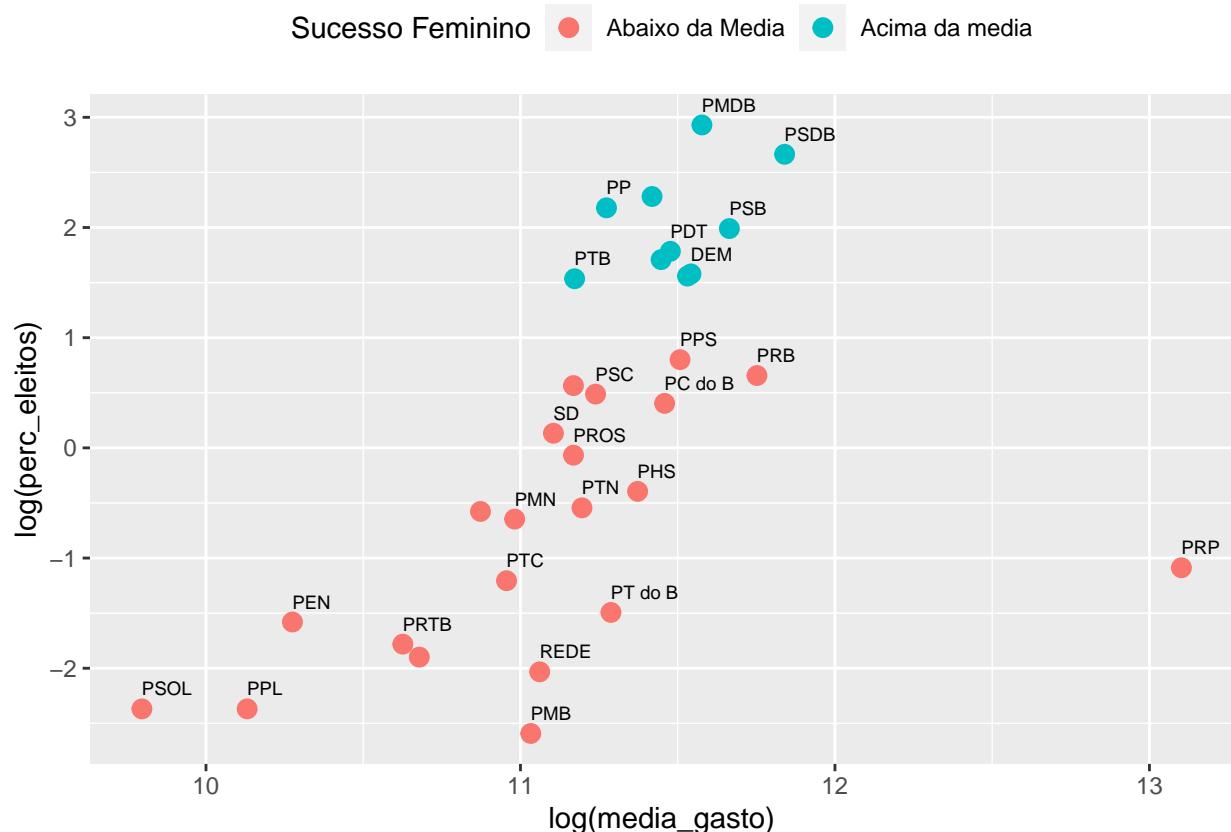


```
# sem tema
novo_plot +
  theme_void()
```



Contudo, é importante saber que você pode editar tudo que você precisa na aparência de um `ggplot` usando a função `theme`. Por questão de tempo, não poderemos ver todas as possibilidades aqui, mas vejamos como alterar a posição das legendas que é algo muito frequente.

```
rot2 +  
  labs(color='Sucesso Feminino') +  
  theme(  
    legend.position = 'top'  
  )
```



neste exemplo, indicamos que a posição da legenda deveria ser no topo. Contudo, poderia ser `bottom`, “`left`”, “`right`” ou “`none`” para não exibir a legenda. Ainda seria possível passar um vetor com dois elementos numéricos para indicar a posição.

É dentro de `theme` que personalizamos ao máximo nos gráficos, então é bom conhecer um pouco mais sobre ela.

Exportação

Para salvar os gráficos você pode usar a função `ggsave`. Com o auxílio dessa função é possível definir tamanho, formato e outras características da figura final.

```
fig1 <-  
novo_plot +  
  theme_minimal()  
  
ggsave('figuras/fig1.pdf', plot = fig1)
```