# Análise textual: Machado de Assis

Mariana de Castro Pasqualini

13/12/2020

## Motivação

Eu sempre gostei bastante de ler livros literários e quantificar características de alguma obra sempre me pareceu bastante interessante! Atualmente, isso tem se tornado possível dado os contínuos trabalhos nos últimos anos com a língua portuguesa e a computação. Escolhi três obras de Machado de Assis, **Quincas Borba (1891)**, **Memórias Póstumas de Brás Cubas (1881)** e **Dom Casmurro (1899)**, por serem de extrema relevância para a literatura brasileira e terem um estilo bastante característico.

### Versão do R e pacotes:

```
R.version.string
```

```
## [1] "R version 4.0.3 (2020-10-10)"
```

```
library(tidyverse)
library(data.table)
library(tidytext)
library(lexiconPT)
library(gutenbergr)
library(udpipe)
library(ggthemes)
library(viridis)
library(topicmodels)
```

## Análise

### Obtendo os dados

Ao invés de ler arquivos de texto para obter os livros, aproveitei o pacote do R `gutenbergr` que permite fazer o download de obras disponibilizadas pelo Projeto Gutenberg. É possível obter mais de um livro com apenas uma chamada da função, a partir do identificador da obra: `55682` refere-se ao livro **Quincas Borba (1891)**, `54829` ao **Memórias Póstumas de Brás Cubas (1881)**, `55752` é a obra **Dom Casmurro (1899)**.

```
livros_machado <- gutenberg_download(c(55752, 55682, 54829))
head(livros_machado, n = 10)
```

```
## # A tibble: 10 x 2
##    gutenberg_id text
##           <int> <chr>
##  1        54829 "MEM\xd3RIAS P\xd3STHUMAS"
##  2        54829 ""
##  3        54829 "DE"
##  4        54829 ""
##  5        54829 "BRAZ CUBAS"
##  6        54829 ""
##  7        54829 "POR"
##  8        54829 ""
##  9        54829 "MACHADO DE ASSIS"
## 10        54829 ""
```

Logo de início, conseguimos identificar que os caracteres especiais não foram lidos corretamente e que o formato do texto não parece o mais adequado. Além disso, é legal ter em mente que as primeiras linhas são o header do texto, ou seja, informações sobre o autor, publicação e editora, que não vão agregar na análise. Voltarei nisso mais adiante.

Também fiz o download do modelo de tags de POS (*part of speech*) do `udpipe`, que usei para o processo de lemmatização dos tokens e obtenção das tags de *adjetivo*, *substantivo* etc.

```
udp_model <- udpipe_download_model(language = "portuguese")
udp_model <- udpipe_load_model(udp_model)
```

### Preparando

**Identificando capítulos e caracteres especiais** A primeira coisa que fiz foi identificar os capítulos de cada livro. Dividi em duas partes.

Para **Quincas Borba** e **Memórias Póstumas de Brás Cubas**, todos os capítulos começam com "CAPITULO", ou seja, bem fácil de identificar! Só precisei trocar "primeiro" por "I" para seguir no padrão de números romanos. Os capítulos foram identificados por uma *expressão regular* que identifica "CAPITULO" seguido de números romanos, conforme a estrutura do texto. Se eu identificasse apenas números romanos em qualquer posição da string, algumas frases do texto seriam excluídas, pois o autor comenta alguns capítulos durante o texto.

```
livros_cap <-
  livros_machado %>%
  group_by(gutenberg_id) %>%
  mutate(text = replace(text, text == "CAPITULO PRIMEIRO", "CAPITULO I")) %>%
  filter(gutenberg_id != 55752) %>%
  mutate(chapter = cumsum(str_detect(text, regex("CAPITULO [MDCLXVI\\.]+$",
                                                 ignore_case = TRUE)))) %>% ungroup()
```

E como fica **Dom Casmurro**?

```
livros <- livros_machado %>%
  filter(gutenberg_id == 55752) %>%
  mutate(chapter = cumsum(str_detect(text, regex("^[MDCLXVI\\.]+$",
                                                 ignore_case = TRUE)))) %>% rbind(livros_cap) %>%
  mutate(text = map(text, function(x) iconv(x, from = "ISO-8859-1", to = "UTF-8"))) %>%
  mutate(text = stringi::stri_trans_general(text, "Latin-ASCII")) %>%
```

```
  mutate(text = gsub("[^-0-9A-Za-z ]","",text)) %>%
  mutate(text = str_squish(text))
```

Identifiquei os capítulos de Dom Casmurro separadamente porque eles não têm "CAPITULO" na frente, apenas os números romanos no **início** da string.

Além disso, no código acima eu também removi caracteres especiais como acentos e cedilha, além de retirar espaços em branco extras.

**Removendo header e footer dos livros** Os livros do Projeto Gutenberg têm um *header*, que contêm o nome do autor/editora/ano, e o *footer*, que em geral contempla o índice do livro. Esse formato pode mudar de acordo com cada livro, mas aqui todas as obras têm essas informações e elas não são úteis para a análise. Removi da seguinte forma:

```
livros <- as.data.table(livros) %>% filter(chapter != 0)
livros <- livros[, remove := .I %in% which(text == "FIM"):.N, by = gutenberg_id]
livros <- livros[remove == FALSE]
livros <- livros[-(16703:16876)] %>% select(-remove)
```

Com os capítulos identificados anteriormente, foi possível identificar que os capítulos **0** contemplavam o cabeçalho dos livros. Pensando no rodapé, todos os textos terminam com "FIM" então as linhas abaixo dessa string poderiam ser removidas. Deu certo para duas obras, mas uma delas tive que remover pelos números das linhas.

```
stop_words <- stopwords::stopwords(language = "pt") %>% as.data.frame()
colnames(stop_words) <- "word"
stop_words <-
  stop_words %>% mutate(word = stringi::stri_trans_general(word, "Latin-ASCII")) %>%
  add_row(word = c("elle", "ella", "tao", "capitulo", "d"))
```

**Identificando stopwords** As *stopwords* são outros elementos que não contribuem para a análise: são palavras como "de", "o/a", "meu" etc. Também adicionei 5 palavras que identifiquei nos textos e que não trazem nenhuma informação valiosa É interessante notar que adicionei "elle/ella" devido à escrita da época dos livros de Machado de Assis. Essas palavras serão removidas no próximo passo.

**Formato *tidy*** Uma das filosofias adotadas pelos usuários do R é o formato *tidy* dos dados, no qual cada variável é uma coluna e cada observação é uma linha. Podemos aplicar esse mesmo princípio para dados textuais com o pacote `tidytext`. Escolhi como *token* palavras porque são a unidade mais interessante para análise nesse contexto.

```
tidy_machado <-
  livros %>% unnest_tokens(word, text) %>%
  anti_join(stop_words)

head(tidy_machado)
```

```
##    gutenberg_id chapter   word
## 1:        55752       1      i
```

```
## 2:        55752       1 titulo
## 3:        55752       1  noite
## 4:        55752       1 destas
## 5:        55752       1  vindo
## 6:        55752       1 cidade
```

**Lemmatização** Esse processo consiste em retirar a flexão do verbo, como por exemplo: caminhando → caminhar. Fazer isso permite agregar essas diferentes flexões em uma palavra só. Felizmente, o pacote udpipe faz isso com bastante facilidade e rapidez para as obras em questão.

```
books_udpipe <- udpipe_annotate(udp_model, pull(tidy_machado, word), tokenizer = "vertical") %>% as.data
head(books_udpipe)
```

```
##   doc_id paragraph_id sentence_id sentence token_id  token  lemma upos xpos
## 1   doc1            1           1     <NA>        1      i     ir VERB <NA>
## 2   doc2            1           1     <NA>        1 titulo titulo NOUN <NA>
## 3   doc3            1           1     <NA>        1  noite  noite NOUN <NA>
## 4   doc4            1           1     <NA>        1 destas  desta NOUN <NA>
## 5   doc5            1           1     <NA>        1  vindo    vir VERB <NA>
## 6   doc6            1           1     <NA>        1 cidade cidade NOUN <NA>
##                                                 feats head_token_id dep_rel
## 1 Mood=Ind|Number=Sing|Person=1|Tense=Past|VerbForm=Fin             0    root
## 2                              Gender=Masc|Number=Sing             0    root
## 3                               Gender=Fem|Number=Sing             0    root
## 4                               Gender=Fem|Number=Plur             0    root
## 5                                         VerbForm=Ger             0    root
## 6                               Gender=Fem|Number=Sing             0    root
##   deps misc
## 1 <NA> <NA>
## 2 <NA> <NA>
## 3 <NA> <NA>
## 4 <NA> <NA>
## 5 <NA> <NA>
## 6 <NA> <NA>
```

**Etapa final da preparação** Aqui, mantive apenas variáveis interessantes e acrescentei o nome dos livros.

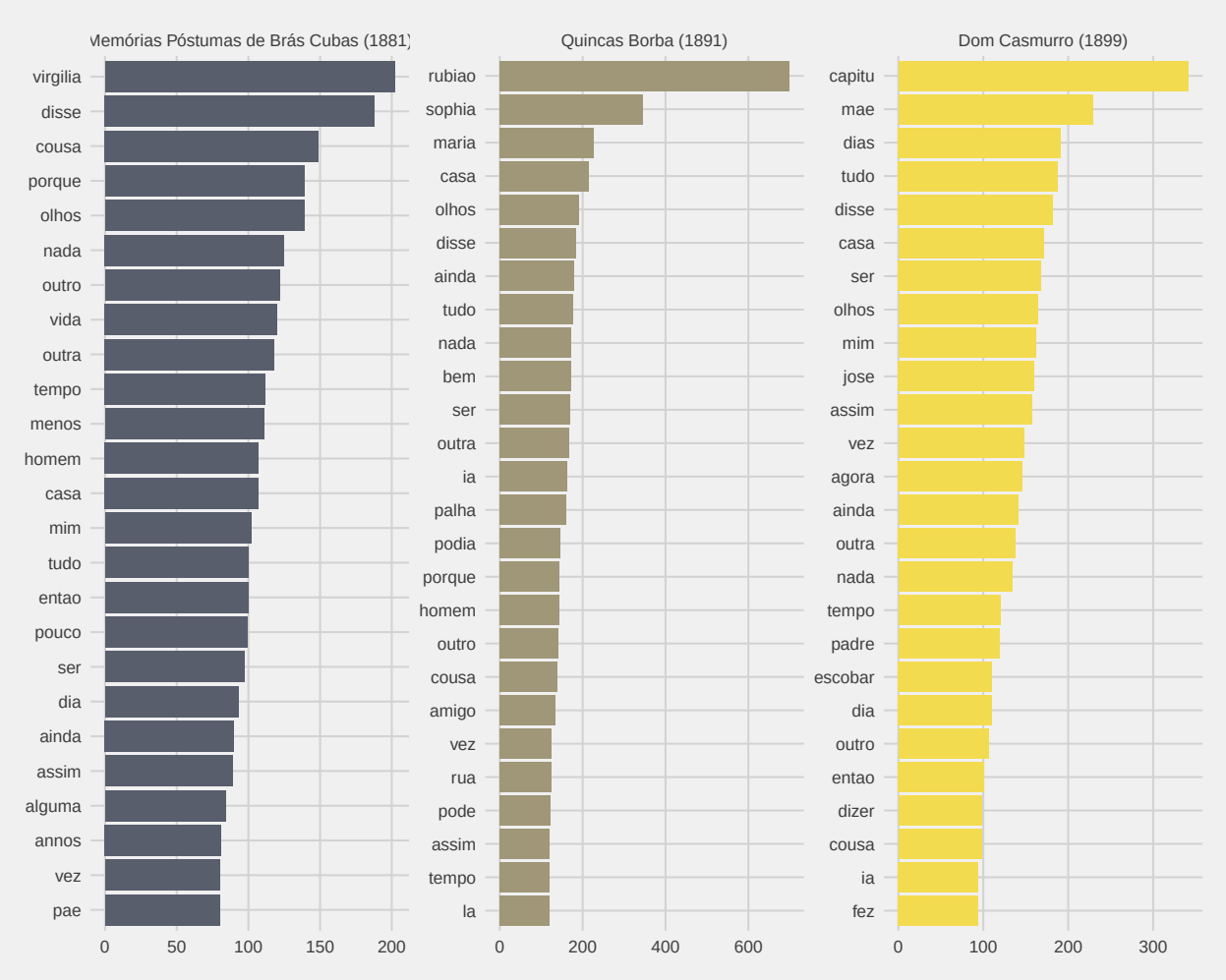
```
processed_machado <-
  tidy_machado %>%
  bind_cols(books_udpipe) %>%
  select(gutenberg_id, chapter, word, lemma, upos) %>%
  mutate(book = factor(case_when(gutenberg_id == 55682 ~ "Quincas Borba (1891)",
                                 gutenberg_id == 54829 ~ "Memórias Póstumas de Brás Cubas (1881)",
                                 gutenberg_id == 55752 ~ "Dom Casmurro (1899)"),
                       levels = c("Memórias Póstumas de Brás Cubas (1881)", "Quincas Borba (1891)", "Dom

head(processed_machado)
```

```
##    gutenberg_id chapter   word  lemma upos               book
## 1:        55752       1      i     ir VERB Dom Casmurro (1899)
## 2:        55752       1 titulo titulo NOUN Dom Casmurro (1899)
## 3:        55752       1  noite  noite NOUN Dom Casmurro (1899)
```

```
## 4:        55752       1 destas  desta NOUN Dom Casmurro (1899)
## 5:        55752       1  vindo    vir VERB Dom Casmurro (1899)
## 6:        55752       1 cidade cidade NOUN Dom Casmurro (1899)
```

### Visualizando frequências

```
processed_machado %>%
  group_by(book) %>%
  count(word, sort = TRUE) %>%
  mutate(word = reorder_within(word, n, book)) %>%
  top_n(n = 25) %>%
  ggplot(aes(x = n, y = word, fill = book)) +
  geom_col(show.legend = FALSE) +
  scale_y_reordered() +
  facet_wrap(~book, scales = "free") +
  scale_fill_viridis_d(option = "cividis", begin = 0.34, end = 0.94) +
  theme_fivethirtyeight()
```




A palavra mais frequente de cada livro são os personagens principais.

- Em Brás Cubas, **Virgilia** é a palavra que aparece mais vezes. Ela é o principal relacionamento de Brás Cubas.
- Em Quincas Borba, **Rubião** é o mais frequente. É interessante notar que essa é a única obra analisada que é escrita em 3ª pessoa. No caso das outras obras, o personagem principal não aparece nas mais frequentes, mas em Quincas Borba sim, pelo tipo de narrador.
- Em Dom Casmurro, **Capitu**, amor de Bentinho, é a palavra mais frequente.

Também notamos que existem palavras muito comuns nas três obras, como **olhos**, **casa** e **tempo**. Isso é bem esperado, dado que as três obras são consideradas pelos pesquisadores uma trilogia, a chamada *trilogia realista*, que explora alguns temas em comum e possuem o mesmo estilo de escrita.

### Análise de sentimento

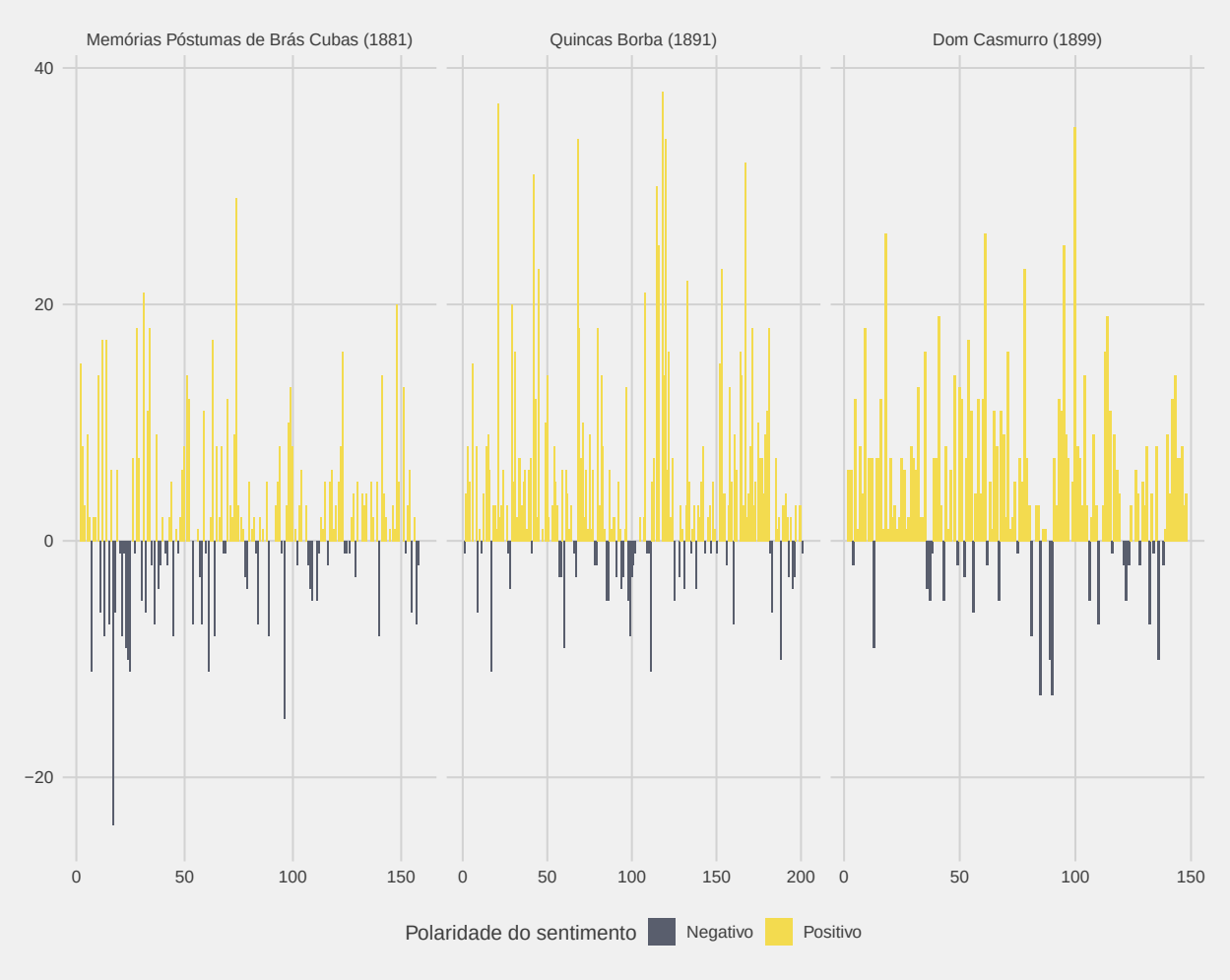
Será que as obras de Machado de Assis são "felizes"?

Usei o léxico de sentimento **OpLexicon** para obter a polaridade dos sentimentos, sendo -1 negativo, 0 neutro e 1 positivo. A ideia é ver o sentimento predominante de cada capítulo, então somei essas polaridades agrupadas por capítulo e livro.

```
sent <-
  processed_machado %>% inner_join(oplexicon_v3.0, by = c("word" = "term")) %>%
  group_by(book, chapter) %>%
  mutate(sentimento = sum(polarity)) %>%
  ungroup() %>%
  filter(sentimento != 0) %>%
  mutate(sentimento_cat = ifelse(sentimento < 0, "Negativo", "Positivo"))
```

**Visualizando sentimentos** Podemos visualizar os sentimento predominantes da seguinte forma:

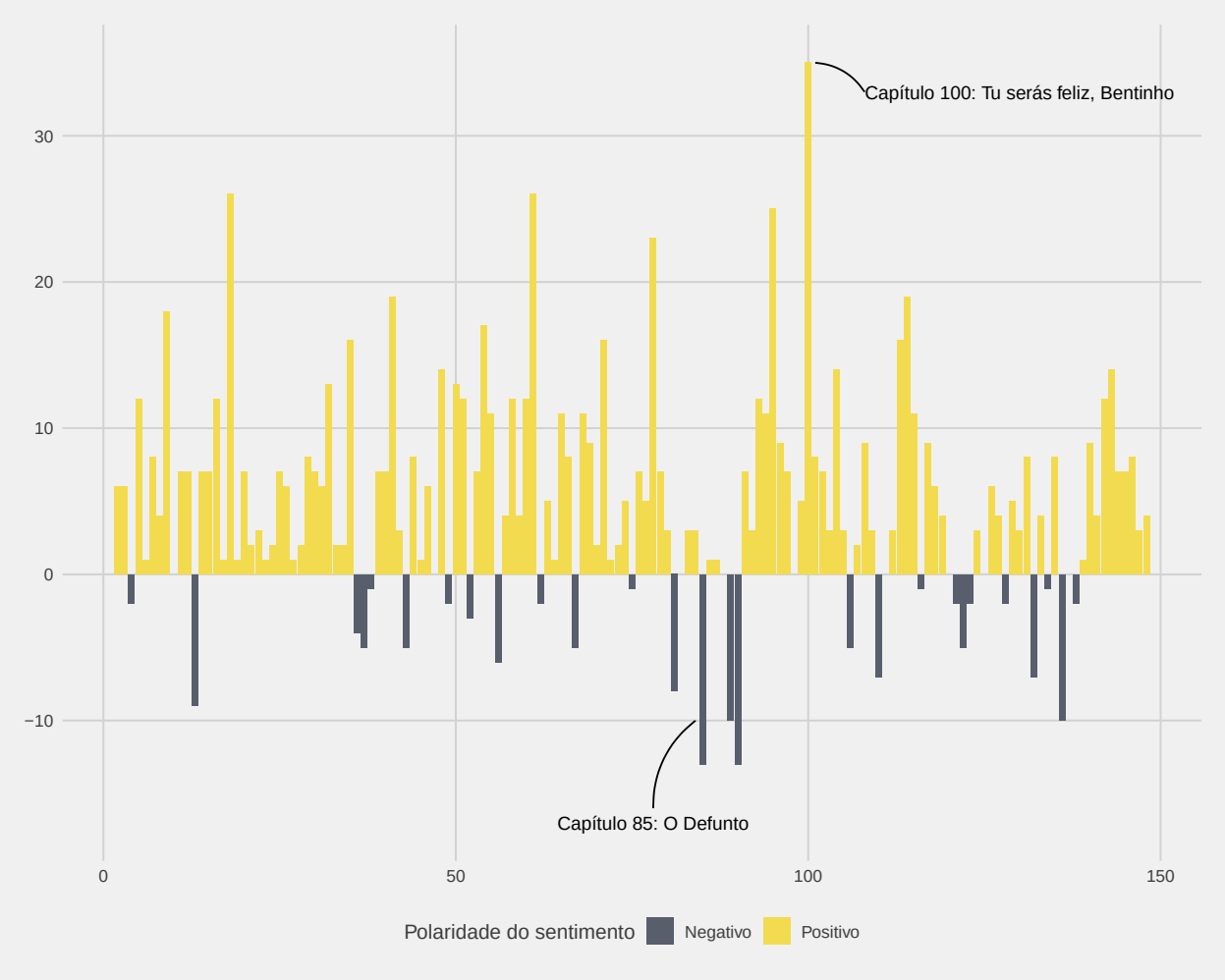
```
  sent %>%
  ggplot(aes(x = chapter, y = sentimento, fill = sentimento_cat)) +
  geom_bar(stat = "identity", position = "identity") +
  facet_grid(~book, scales = "free") +
  scale_fill_viridis_d(option = "cividis", begin = 0.34, end = 0.94) +
  labs(fill = "Polaridade do sentimento") +
  theme_fivethirtyeight()
```




- Brás Cubas parece ter a maior concentração de sentimentos **negativos**
- Os capítulos mais **positivos** vem da obra Quincas Borba

Olhando apenas para Dom Casmurro:

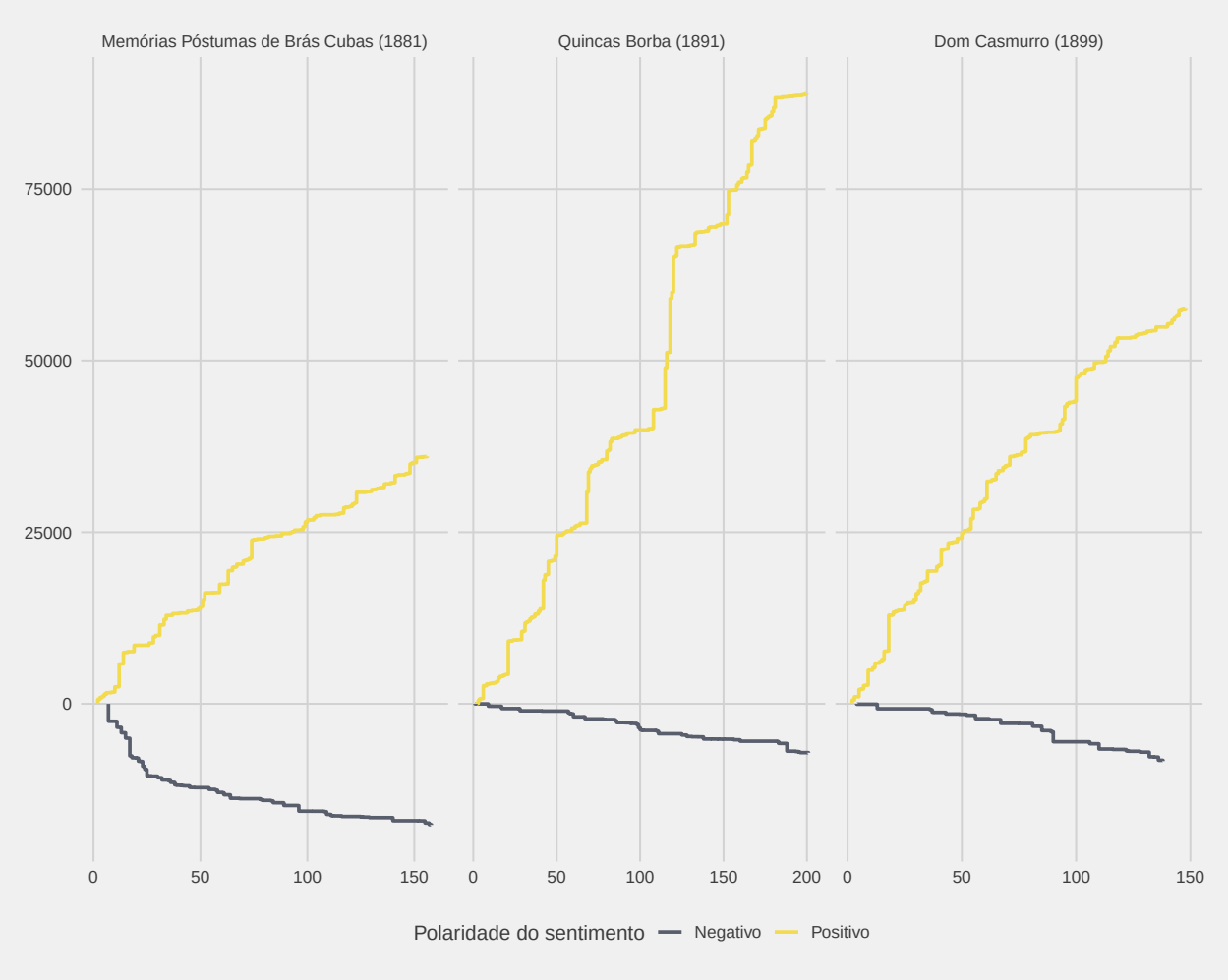
```
sent_plot_dc <-
sent %>%
  filter(gutenberg_id == 55752) %>%
  ggplot(aes(x = chapter, y = sentimento, fill = sentimento_cat)) +
  geom_bar(stat = "identity", position = "identity") +
  scale_fill_viridis_d(option = "cividis", begin = 0.34, end = 0.94) +
  labs(fill = "Polaridade do sentimento") +
  theme_fivethirtyeight()

sent_plot_dc +
  annotate(geom = "curve", x = 78, xend = 84, y = -16, yend = -10, curvature = -0.25) +
  annotate(geom = "text", x = 78, y = -17, label = "Capítulo 85: O Defunto") +
  annotate(geom = "curve", x = 101, xend = 108, y = 35, yend = 33, curvature = -0.25) +
  annotate(geom = "text", x = 130, y = 33, label = "Capítulo 100: Tu serás feliz, Bentinho")
```




Observamos que os títulos dos capítulos mais positivos e negativos já entregam bem o que é contemplado no texto.

E para concluir, podemos ver os sentimentos acumulados ao longo de cada obra:

```
sent %>%
group_by(book, sentimento_cat) %>%
mutate(cumsum = cumsum(sentimento)) %>%
ggplot(aes(x = chapter, y = cumsum, color = sentimento_cat)) +
geom_line(size = 1) +
facet_grid(~book, scales = "free") +
scale_color_viridis_d(option = "cividis", begin = 0.34, end = 0.94) +
labs(color = "Polaridade do sentimento") +
theme_fivethirtyeight()
```




Quincas Borba é, de longe, o livro mais positivo analisado, seguido de Dom Casmurro. Brás Cubas é o mais negativo, como esperado, já que o personagem narra sua história de uma forma bastante pessimista.

### Preparando... de novo!

Quando queremos aplicar modelos em dados textuais, o formato mais comum é o **document-term matrix** (DTM), no qual uma linha é um documento (no caso, três linhas para três livros), cada coluna é um termo (aqui, são várias palavras) e o valor dentro dessas células é a contagem de vezes que o termo aparece no documento. Vou passar os dados do formato *tidy* comentado acima para o DTM:

```
tidy_count <- tidy_machado %>%
  group_by(gutenberg_id) %>%
  count(word, sort = TRUE) %>% ungroup() %>% rename("document" = "gutenberg_id")

dtm <- tidy_count %>% cast_dtm(term = word, document = document,  value = n)
```

Agora podemos aplicar modelos de topificação.

### Tópicos latentes

Vou usar o modelo de alocação latente de Dirichlet (LDA), que permite entender semelhanças entre as observações através de variáveis latentes (que não podemos medir diretamente). Especificamente para textos, entende-se que um tópico é uma mistura de palavras e um documento é uma mistura de tópicos.

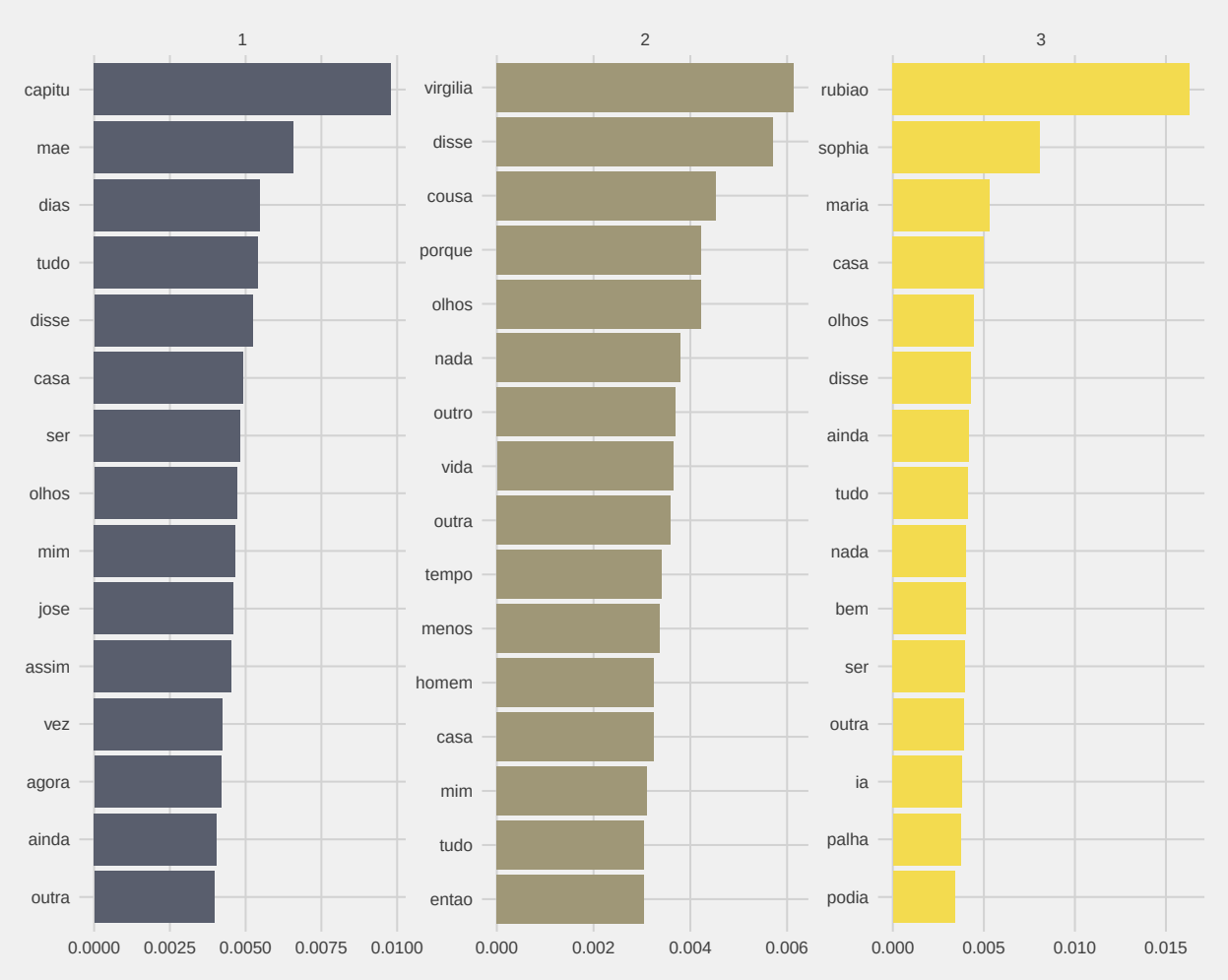
```
fit_lda <- LDA(dtm, k = 3, control = list(seed = 758))
fit_lda
```

### Corpus completo

```
## A LDA_VEM topic model with 3 topics.
```

Apliquei o modelo de LDA com **3 tópicos**, pois estou analisando três livros diferentes e quero agrupá-los em três grupos diferentes. Vamos ver as probabilidades por termo por tópico, ou seja, o "peso" de cada palavra em um tópico. Aqui selecionei as palavras mais relevantes.

```
machado_topics <- tidy(fit_lda, matrix = "beta")

machado_top_terms <- machado_topics %>%
  group_by(topic) %>%
  top_n(15, beta) %>%
  ungroup() %>%
  arrange(topic, -beta)

machado_top_terms %>%
  mutate(term = reorder_within(term, beta, topic)) %>%
  ggplot(aes(beta, term, fill = factor(topic))) +
  geom_col(show.legend = FALSE) +
  facet_wrap(~ topic, scales = "free") +
  scale_y_reordered() +
  scale_fill_viridis_d(option = "cividis", begin = 0.34, end = 0.94) +
  theme_fivethirtyeight()
```




Claramente os personagens principais dos três livros aparecem com grande peso em cada tópico e existe algumas palavras com pesos bem próximos nos três tópicos. No entanto, algumas palavras que apareceram não contribuem tanto para a caracterização do tópico, então podemos ajustar um modelo que contempla apenas **substantivos**.

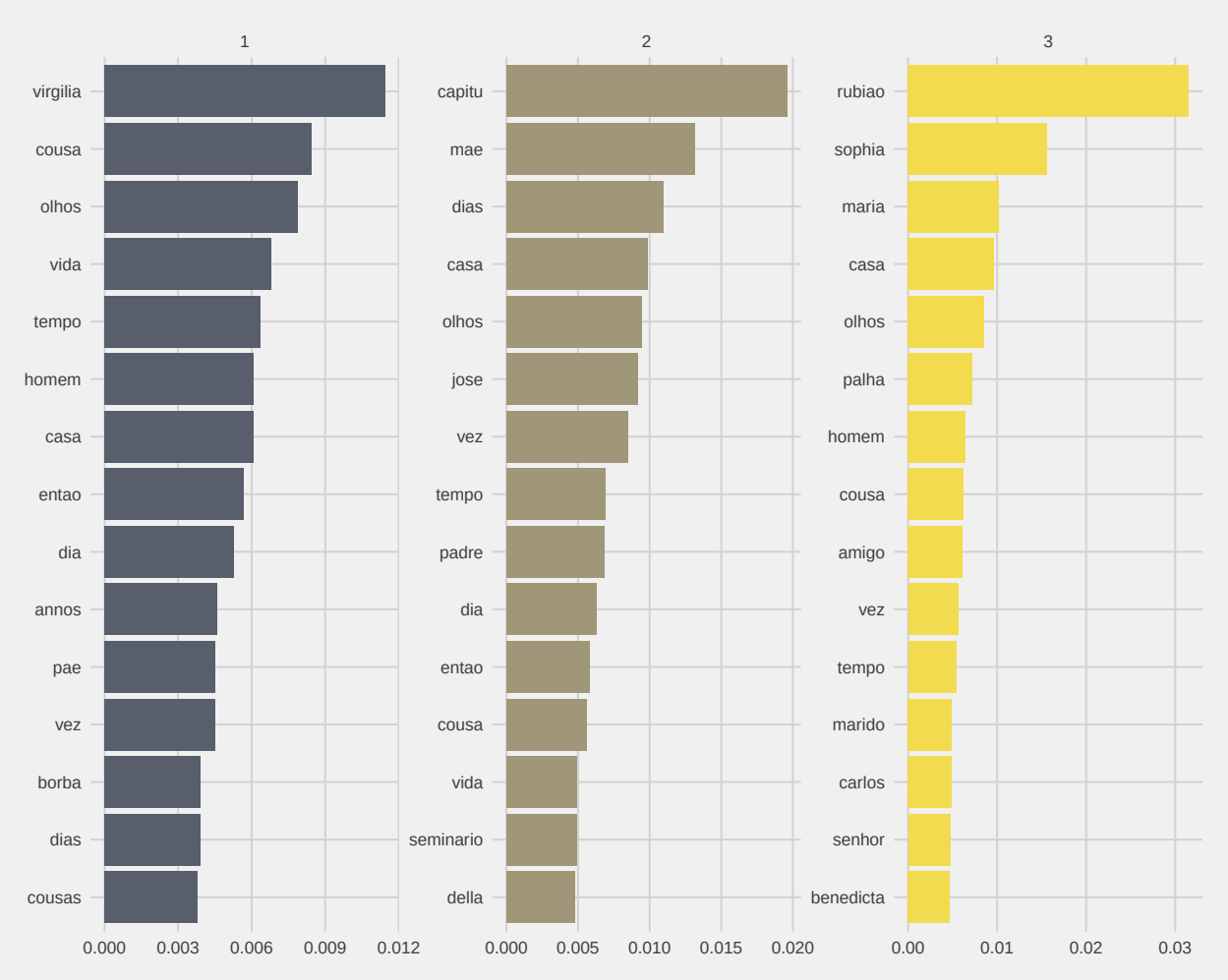
```
tidy_count_nouns <-
  processed_machado %>%
  filter(upos == "NOUN" | word == "capitu") %>%
  group_by(gutenberg_id) %>%
  count(word, sort = TRUE) %>%
  ungroup() %>%
  rename("document" = "gutenberg_id")

dtm_noun <- tidy_count_nouns %>% cast_dtm(term = word, document = document,  value = n)

fit_lda_noun <- LDA(dtm_noun, k = 3, control = list(seed = 758))
fit_lda_noun
```

**Apenas substantivos**

```
## A LDA_VEM topic model with 3 topics.
```

Os passos são bem parecidos com os realizados anteriormente, apenas filtrando a tag **noun** e incluindo **Capitu** como substantivo, que infelizmente foi tagueado errado como verbo. Vamos olhar novamente para os pesos das palavras para cada tópico:

```
machado_topics_nouns <- tidy(fit_lda_noun, matrix = "beta")

machado_top_nouns <- machado_topics_nouns %>%
  group_by(topic) %>%
  top_n(15, beta) %>%
  ungroup() %>%
  arrange(topic, -beta)

# more interesting than the raw corpus!
machado_top_nouns %>%
  mutate(term = reorder_within(term, beta, topic)) %>%
  ggplot(aes(beta, term, fill = factor(topic))) +
  geom_col(show.legend = FALSE) +
  facet_wrap(~ topic, scales = "free") +
  scale_y_reordered() +
  scale_fill_viridis_d(option = "cividis", begin = 0.34, end = 0.94) +
  theme_fivethirtyeight()
```

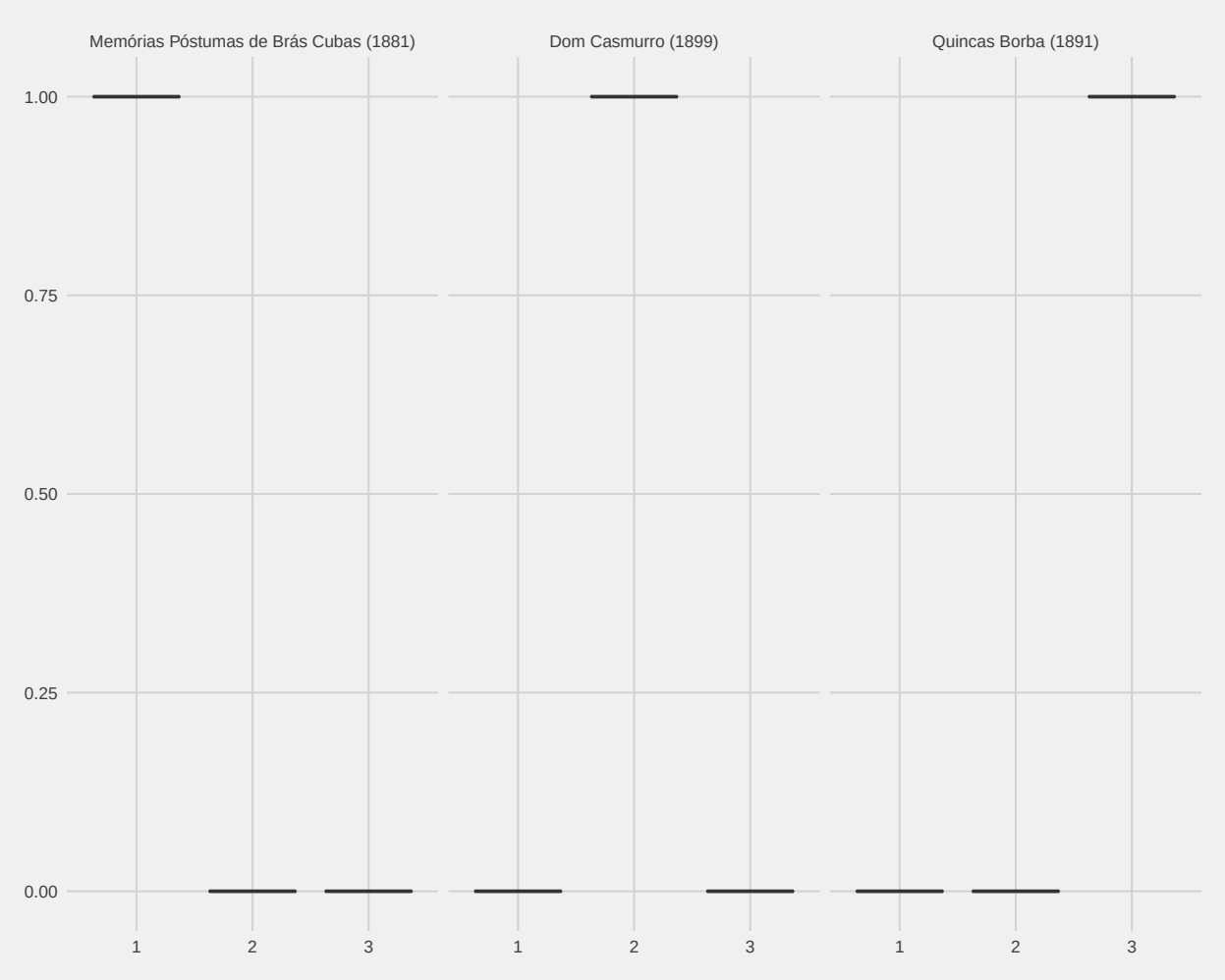



Parece um pouco mais interessante do que as palavras do modelo anterior! Aparecem as palavras **seminário** e **padre** no tópico 2, bem importante na história de Dom Casmurro, na qual a mãe de Bentinho deseja esse futuro para o filho. É interessante notar que, dado que as palavras estão separadas individualmente, vemos "dia" e "dias", sendo o primeiro em relação ao tempo e o segundo ao personagem **José Dias**.

Também vemos **Borba** no tópico 1, referente ao Quincas Borba, amigo de infância de Brás Cubas. No tópico 3, surge a palavra **marido**, importante para as relações entre Rubião, Sophia e Cristiano Palha da obra Quincas Borba.

Tendo a ideia de um documento ser uma mistura de tópicos, podemos ver as probabilidades de cada documento ter sido retirado de um determinado tópico.

```
book_names <- c("55682" = "Quincas Borba (1891)", "54829" = "Memórias Póstumas de Brás Cubas (1881)", "5

# gamma probabilities
tidy(fit_lda_noun, matrix = "gamma") %>%
  mutate(document = reorder(document, gamma * topic)) %>%
  ggplot(aes(factor(topic), gamma)) +
  geom_boxplot() +
  facet_wrap(~ document, labeller = labeller(document = book_names)) +
  scale_fill_viridis_d(option = "cividis", begin = 0.34, end = 0.94) +
  theme_fivethirtyeight()
```




Nas obras analisadas, cada documento ficou exclusivamente com cada tópico: ou seja, a probabilidade de Brás Cubas ter sido retirado do tópico 1 é praticamente 1 e 0 para os outros dois tópicos. O mesmo se segue para os outros dois textos de Machado de Assis.

## Conclusão

Dessa análise, podemos pensar em trabalhos futuros bem interessantes, como:

- Será que os mesmos tópicos da trilogia realista de Machado aparecem em suas outras obras?
- Como o ano ou o gênero da obra influenciam no texto do autor?
- O quão similar a outros escritores contemporâneos é a sua escrita?
- Qual a influência do estilo de Machado de Assis em outros autores?

Essas perguntas podem ser exploradas com técnicas mais sofisticadas de análise de texto e trazer uma perspectiva quantitativa da literatura brasileira.

## Referências


- Text mining with R: https://www.tidytextmining.com/index.html
- An introduction to Text processing and Analysis with R: https://m-clark.github.io/text-analysis-with-R/intro.html
- UDPipe Natural Language Processing - Topic Modelling Use Cases: https://cran.r-project.org/web/packages/udpipe/vignettes/udpipe-usecase-topicmodelling.html
- gutenbergr: Search and download public domain texts from Project Gutenberg: https://cran.r-project.org/web/packages/gutenbergr/vignettes/intro.html
- Introdução a NLTK com Dom Casmurro: https://medium.com/turing-talks/uma-an%C3%A1lise-de-dom-casmurro-com-nltk-343d72dd47a7