Lista II: Análise de Dados da CAPES

Dan Nogueira da Silva

2025-03-05

Nesta lista, o objetivo será analisar a produção de teses e dissertações de programas de pós-graduação notas 4, 5, 6 e 7 na Capes, das áreas de Sociologia, entre os anos de 1987 e 2022. A análise explora a evolução da produção ao longo do tempo, a distribuição por subtemas e a concentração regional das defesas, seguindo um recorte de palavras-chave específico. Os dados foram processados e analisados utilizando o software R e pacotes específicos para análise de dados e visualização.

Tendências na produção de teses e dissertações em Sociologia

Metodologia

A análise foi realizada utilizando dados coletados de duas fontes:

- **Banco de dados de defesas de teses e dissertações da CAPES:** Contém informações sobre as defesas realizadas entre 1987 e 2022, incluindo dados sobre o programa, a instituição, o autor, o título, as palavras-chave e a área de avaliação.
- **Banco de dados de programas de pós-graduação da CAPES:** Contém informações sobre os programas de pós-graduação, incluindo o código do programa, o estado e o conceito CAPES.

1. Importando arquivos

Para importar a base de dados contendo as dissertações e teses de programas na CAPES, optei por utilizar a função `map_df`. Nesse caso, a map_df é mais prática que o loop por economizar linhas e espaço de memória.

```
banco_defesas <- map_df(c("csvs/capes_1987-1992.csv", "csvs/capes_1993-1998.csv","csvs/capes_1999
```

Usando `map_df` é possível combinar todas as planilhas contendo as teses e dissertações defendidas entre 1987 a 2022, em um único tibble. Como resultado, a planilha possui 13 variáveis: código do programa, ano, sigla, nome da instituição, nome do programa, grande área, área de conhecimento, área de avaliação, autor, titulo, nivel, palavas-chave e resumo. Cada observação diz respeito a uma defesa.

Para carregar a planilha com informações sobre os programas de pós-graduação, usei a função `import()` do pacote `rio`. A planilha resultante possui 3 variáveis: código do programa, estado e conceito CAPES. Cada observação diz respeito a um programa de pós-graduação

```
banco_programas <- import("programas.csv")
```

1. Importando

Para importar a base de dados contendo as dissertações e teses de programas na CAPES, optei por utilizar o `map_df` pois nesse caso achei mais prático do que usar um laço de repetição, por economizar linhas e espaço de memória.

Usando `map_df` é possível combinar todas as planilhas contendo as teses e dissertações defendidas entre 1987 a 2022, em um único tibble. Essa planilha possui 13 variáveis: código do programa, ano, sigla, nome da instituição, nome do programa, grande área, área de conhecimento, área de avaliação, autor, título, nível, palavras-chave e resumo. Cada observação diz respeito a uma defesa.

Para carregar a planilha com informações sobre os programas de pós-graduação, usei a função `import()` do pacote `rio`. Essa planilha possui 3 variáveis: código do programa, estado e conceito CAPES. Cada observação diz respeito a um programa de pós-graduação.

```
teses_e_programas <- banco_defesas |>
  left_join(banco_programas, by = c("codigo_programa" = "CD_PROGRAMA"))
```

Para concatenar as informações sobre os programas com as informações sobre as defesas, utilizei a função `left_join` para juntar as informações partindo de uma variável em comum: o código do programa.

Ao analisar a nova base, percebi alguns missings na variável UF. Tentei arrumar a lista me guiando a partir das colunas `siglas_ies` e `UF`. Além disso, incluí a variável `regiao`, que vai ser útil mais à frente.

```
banco_tidy <- teses_e_programas |>
  mutate(UF = case_when(
    str_detect(sigla_ies, "RJ|RIO|UENF|UFF|UCAM") ~ "RJ",
    str_detect(sigla_ies, "SP|UNICAMP") ~ "SP",
    str_detect(sigla_ies, "ES|UVV") ~ "ES",
    str_detect(sigla_ies, "AC") ~ "AC",
    str_detect(sigla_ies, "AL") ~ "AL",
    str_detect(sigla_ies, "AP") ~ "AP",
    str_detect(sigla_ies, "UFAM") ~ "AM",
    str_detect(sigla_ies, "BA") ~ "BA",
    str_detect(sigla_ies, "CE|FJN|UFC") ~ "CE",
    str_detect(sigla_ies, "DF|UNB") ~ "DF",
    str_detect(sigla_ies, "GO|UFG") ~ "GO",
    str_detect(sigla_ies, "MA") ~ "MA",
    str_detect(sigla_ies, "MT") ~ "MT",
    str_detect(sigla_ies, "MS|UFGD") ~ "MS",
    str_detect(sigla_ies, "MG") ~ "MG",
    str_detect(sigla_ies, "PA") ~ "PA",
    str_detect(sigla_ies, "PB|UFCG") ~ "PB",
    str_detect(sigla_ies, "PR|UEL") ~ "PR",
    str_detect(sigla_ies, "PE|UNIVASF") ~ "PE",
```

```
    str_detect(sigla_ies, "PI") ~ "PI",
    str_detect(sigla_ies, "RN") ~ "RN",
    str_detect(sigla_ies, "RS|UFRGS") ~ "RS",
    str_detect(sigla_ies, "RO") ~ "RO",
    str_detect(sigla_ies, "RR") ~ "RR",
    str_detect(sigla_ies, "SC") ~ "SC",
    str_detect(sigla_ies, "SE") ~ "SE",
    str_detect(sigla_ies, "TO") ~ "TO",
    TRUE ~ "Outros"
  )) |>
  mutate(regiao = case_when(
    UF %in% c("AM", "RR", "AP", "PA", "TO", "RO", "AC") ~ "Norte",
    UF %in% c("MA", "PI", "CE", "RN", "PE", "PB", "SE", "AL", "BA") ~ "Nordeste",
    UF %in% c("MT", "MS", "GO", "DF") ~ "Centro Oeste",
    UF %in% c("PR", "SC", "RS") ~ "Sul",
    UF %in% c("SP", "RJ", "ES", "MG") ~ "Sudeste",
    TRUE ~ "Não reportado"
  ))
```

Após limpar a lista, filtrei apenas observações que continham programas com notas maiores que 4 sobre minha área de interesse (sociologia). Por algum motivo os missings continuavam a ser considerados quando eu utilizava a lógica `CONCEITO == 4|5|6|7`, então achei melhor filtrar usando a negação das observações indesejadas.

```
teses_sociologia <- banco_tidy |>
  filter(CONCEITO != "NA|3|A") |>
  filter(str_detect(nome_programa, "SOCIOLOGIA"))
```

2. Seleção de palavras-chave

Para o meu desenho de pesquisa, as 3 palavras-chave mais interessantes são ensino superior, desigualdade e educação. Filtrei a base para mostrar apenas defesas que se enquadravam em pelo menos uma das 3 palavras-chave.

```
teses_relevantes <- teses_sociologia |>
 filter(str_detect(palavras_chave,"ensino superior|desigualdade|educação"))
```

3. Evolução ao longo do tempo

Para criar uma visualização que reporte de forma sucinta e informativa a produção de teses e dissertações no meu tema por ano, primeiro era preciso criar uma tabela de contagem das ocorrências de defesas por ano e palavra-chave. Criei a variável `subtema`, que classificava as defesas por palavra-chave correspondente. Depois contei quantas ocorrências cada variável `ano` tinha em relação a cada observação da variável `subtema`.

```r
teses_por_ano_subtema <- teses_relevantes |>
 mutate(subtema = case_when(
  str_detect(palavras_chave, "ensino superior") ~ "Ensino Superior",
  str_detect(palavras_chave, "desigualdade") ~ "Desigualdade",
  str_detect(palavras_chave, "educação") ~ "Educação",
  TRUE ~ "Outros"
 )) |>
 count(ano, subtema) |>
 rename(frequencia = n)
```

Para visualizar, escolhi o gráfico de barras empilhadas para ver quantas observações foram feitas para cada ano do eixo x, qualificando a frequência por palavras-chave. Assim consigo ver quantas teses foram defendidas em cada ano, ao mesmo tempo em que consigo ver a frequência de cada subtema que considerei relevante para minha pesquisa. O gráfico em barras empilhadas é perfeito para visualizar a relação entre uma variável numérica e uma variável categórica.

```r
ggplot(teses_por_ano_subtema, aes(x = ano, y = frequencia, fill = subtema)) +
    geom_bar(stat = "identity", position = "stack") +
    labs(title = "Produção de Teses e Dissertações em Sociologia, por Palavra-Chave (1987-2022)",
         x = "
            Ano de defesa", y = "Número de defesas
        ") +
    scale_x_continuous(breaks = unique(teses_por_ano_subtema$ano)) +
    theme_classic() +
    theme(panel.grid.minor = element_blank())+
    theme(axis.text.x = element_text(angle = 45, hjust = 1)) +
    scale_fill_viridis_d(labels = c("Ensino Superior", "Desigualdade", "Educação", "Outros"),
                         name = "Legenda:")
```

4. Diferenças regionais

Nesta seção, serão calculados o total de trabalhos defendidos ao longo de todo o período por estado e serão criadas duas visualizações: em uma, reporta a frequência de trabalhos por região; em outra, a frequência de trabalhos por unidade da federação.

Mapa da frequência de trabalhos por Unidade da Federação usando geobr

```r
coordenadas_estados <- read_state() # função geobr

# Agrupando teses por estado e juntando com coordenadas do pacote geobr
teses_por_estado <- teses_sociologia |>
  group_by(UF) |>
  summarise(total_trabalhos_uf = n()) |>
```

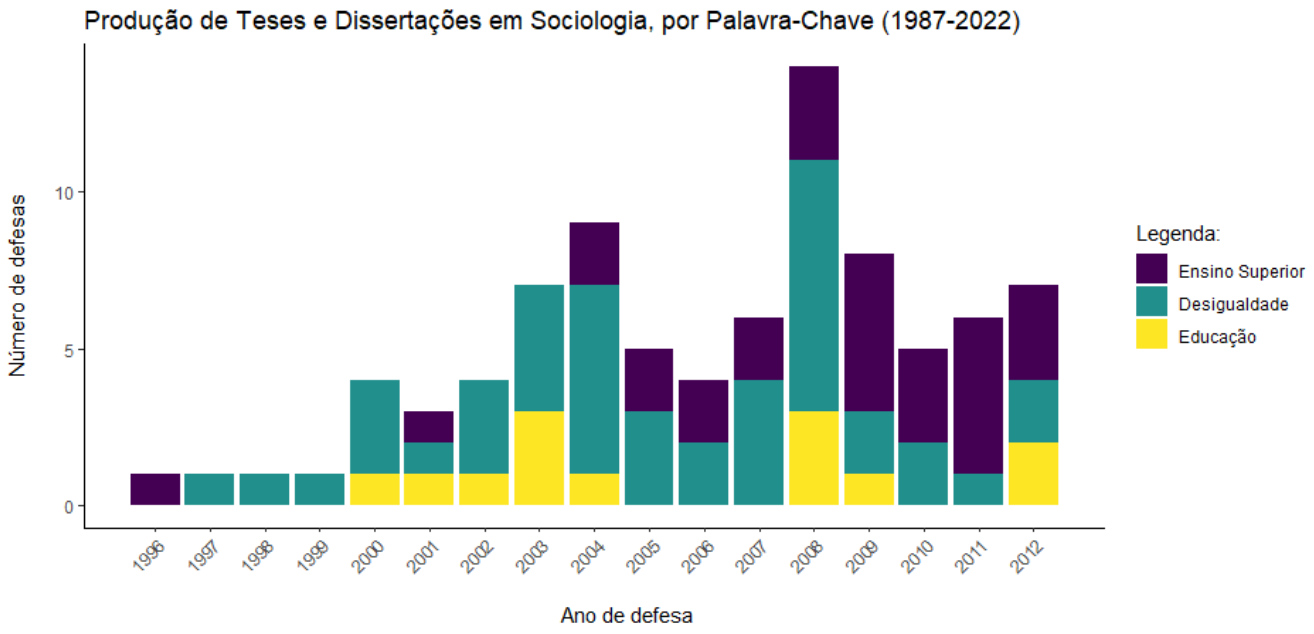

Figura 1: Gráfico de Produção de Teses e Dissertações em Sociologia, por Palavra-Chave (1987-2022)

```
  full_join(coordenadas_estados, by = c("UF" = "abbrev_state")) |>
  st_as_sf()

# Mapa da frequência de defesas por estado

  teses_por_estado |>
  ggplot(aes(fill = total_trabalhos_uf, label =  paste(UF, " (", total_trabalhos_uf, ")", sep = "
  geom_sf()+
  geom_sf_label(fill = "white", size = 1.9, nudge_x = 0.5)+
  scale_fill_viridis_c(name = "Número de Trabalhos")+
  ggtitle("Frequência de Trabalhos de Sociologia Defendidos por Unidade da Federação")+
  theme_void()+
  labs(subtitle = "1987 - 2022",
       caption = "FONTE: CAPES")
```

É possível observar que São Paulo é de longe o estado que mais produz na seleção que fiz.

Mapa da frequência de trabalhos por região

```
teses_por_regiao <- teses_sociologia |>
  group_by(regiao) |>
  summarise(total_trabalhos_regiao = n()) |>
  full_join(coordenadas_estados, by = c("regiao" = "name_region")) |>
  st_as_sf()
```

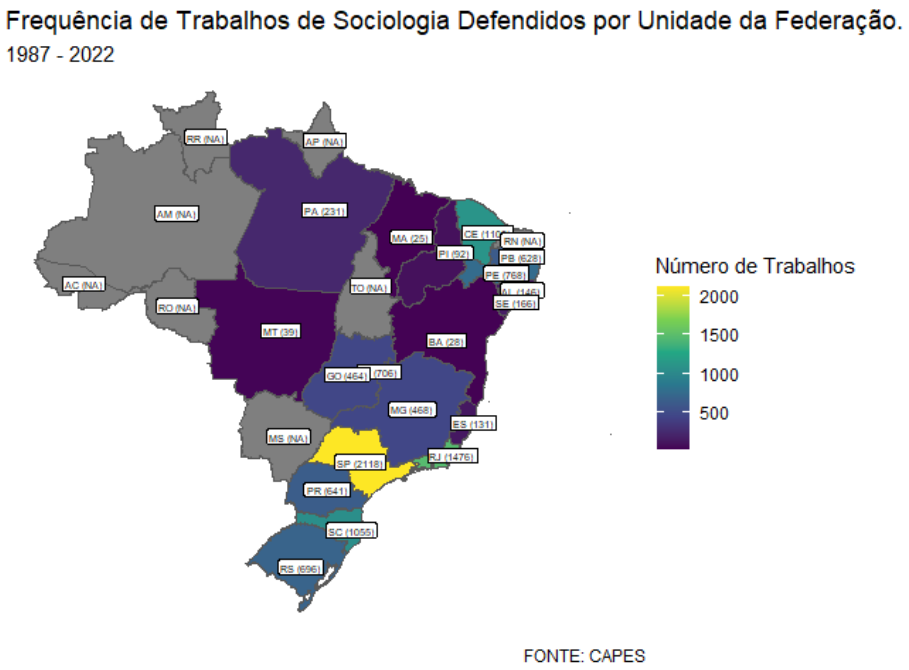

Figura 2: Frequência de Trabalhos de Sociologia Defendidos por Unidade da Federação

```
ggplot(teses_por_regiao) +
  geom_sf(aes(fill = total_trabalhos_regiao)) +
  scale_fill_viridis_c(name = "Número de Trabalhos") +
  theme_void() +
  labs(title = "Frequência de Trabalhos de Sociologia Defendidos por Região do País",
       subtitle = "1987 - 2022",
       caption = "FONTE: CAPES")
```

5. Produção por programa

Nesta seção, será calculado o total de teses e dissertações defendidas por programa de pós-graduação. Em seguida, será reportado em uma tabela o número de trabalhos defendidos pelos 10 programas com maior produção.

```
trabalhos_por_programa <- teses_sociologia |>
  mutate(tese_ou_defesa = case_when(
    str_detect(nivel, "Mestrado|MESTRADO|MESTRADO PROFISSIONAL") ~ "dissertacao",
    str_detect(nivel, "Doutorado|DOUTORADO") ~ "tese",
  )) |>
  count(sigla_ies, nome_programa, tese_ou_defesa, CONCEITO) |>
  rename(trabalhos = n)

trabalhos_por_programa <- pivot_wider(trabalhos_por_programa, names_from = tese_ou_defesa, values
```

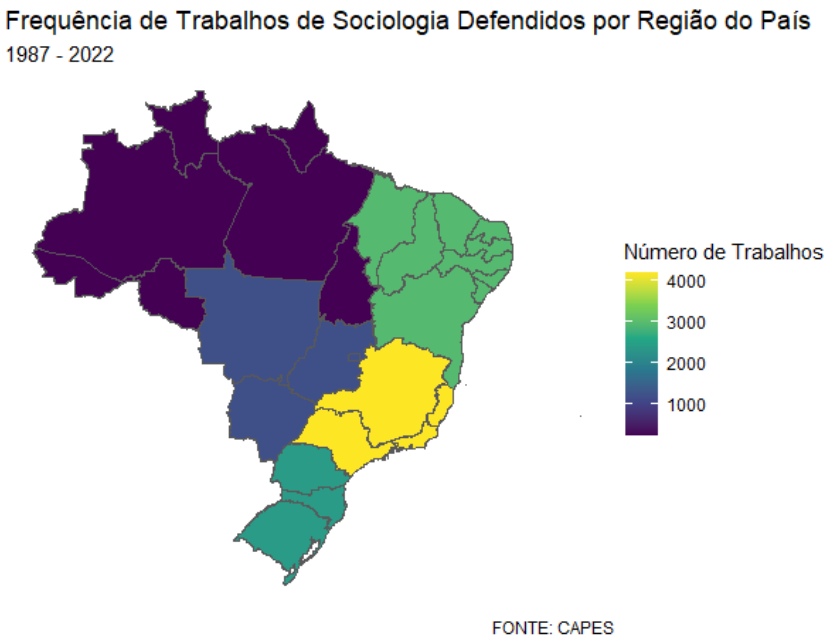

Figura 3: Frequência de Trabalhos de Sociologia Defendidos por Região do País

```
  mutate(tese = case_when(
    tese > 0  ~ tese,
    TRUE ~ 0
  )) |>
  mutate(total_defesas = dissertacao + tese) |>
  arrange(-total_defesas)|>
  slice(1:10)|>
  select(-total_defesas)


ggplot(trabalhos_por_programa, aes(x =reorder(sigla_ies, -(tese+dissertacao)), y=  tese + dissert
  geom_bar(stat = "identity", position = "stack") +
  labs(title = "Produção em Sociologia, por programa de pós-graduação (1987-2022)",
       x = "Sigla da instituição de ensino superior", y = "Número de defesas") +
  theme_classic() +
  theme(panel.grid.minor = element_blank())+
  scale_fill_viridis_d(name = "Nome do programa")
```

6. Exportação

Nesta última etapa, será criada uma base menor contendo apenas as seguintes variáveis: ano, estado, programa, título, resumo e autor(a). Essa base será exportada para uma planilha de Excel.

```
resumo <- teses_relevantes |>
  select(ano, UF, nome_programa, titulo, resumo, autor)
```

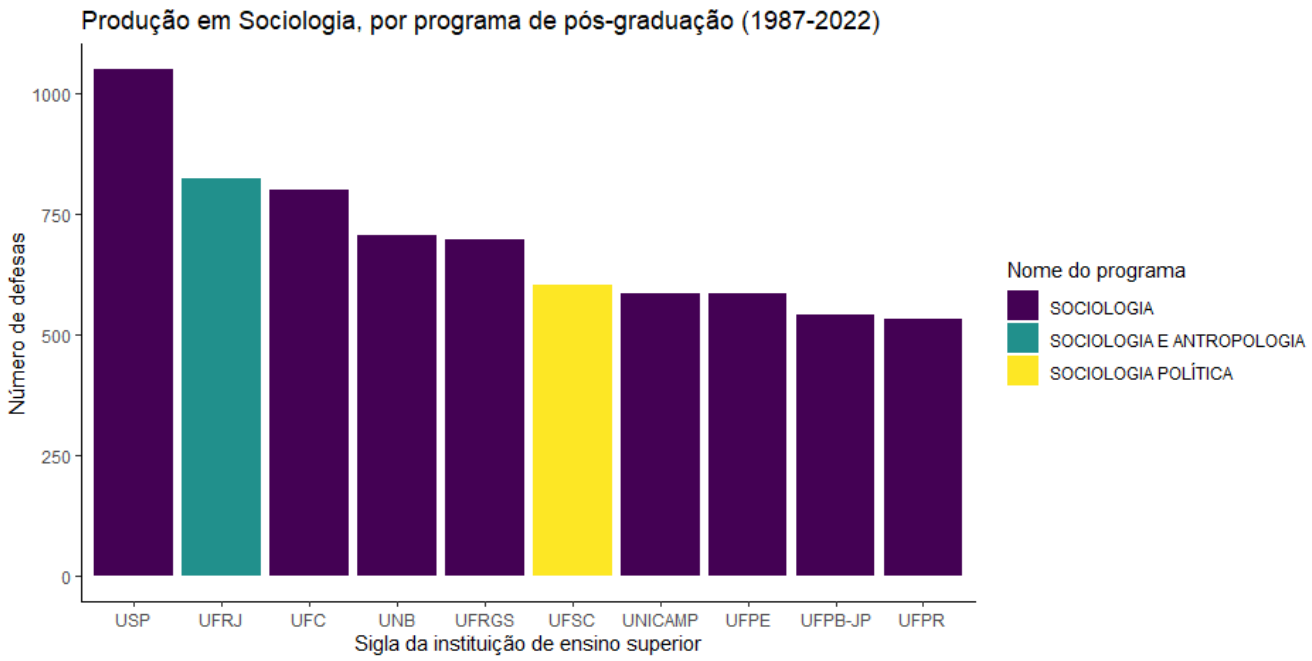

Figura 4: Produção em Sociologia, por programa de pós-graduação (1987-2022)

```
write_csv(resumo, "resumo.csv")
```