Projeto 1 - FLS 6497

Pedro Schmalz 10389052

2022-10-22

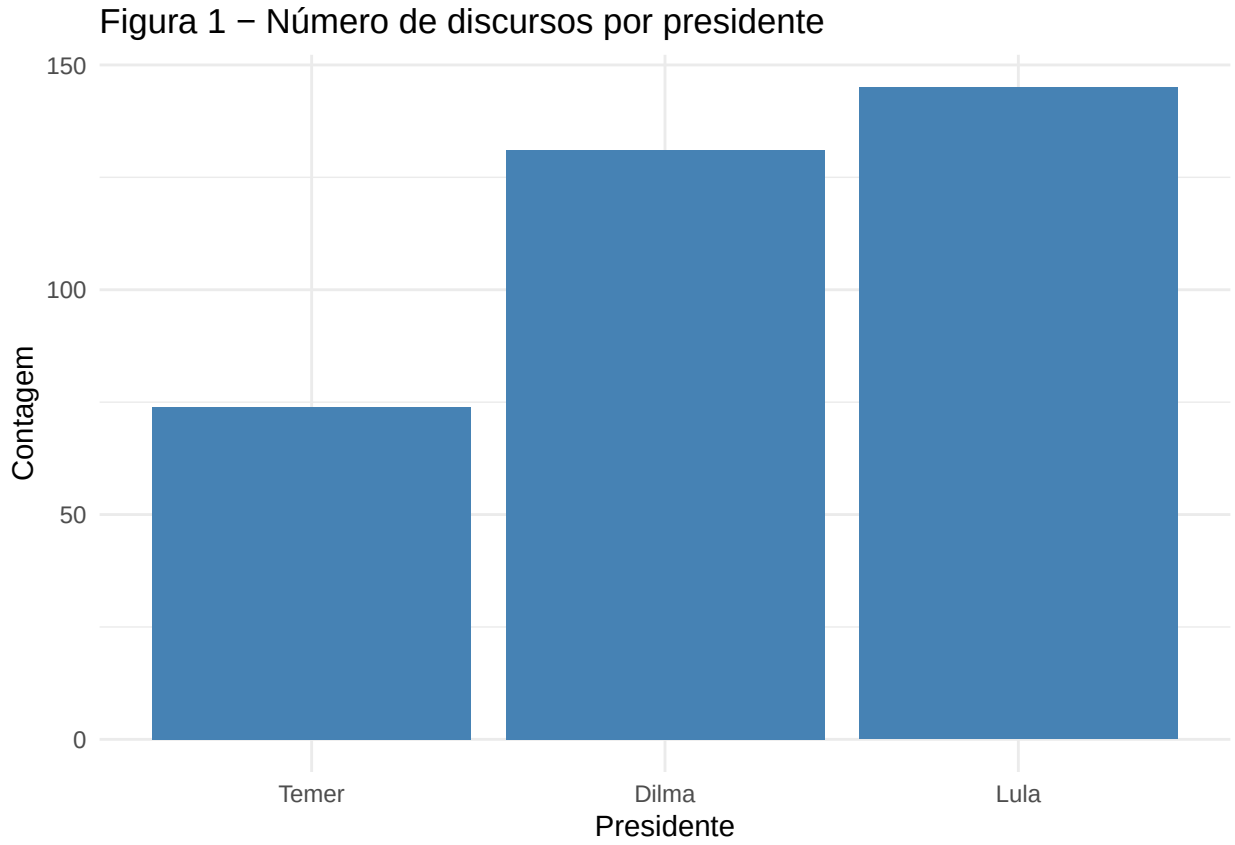
Prevendo a autoria de discursos presidenciais.

Neste trabalho, utilizaremos aprendizado de máquina (*machine learning*) para prever a autoria de discursos presidenciais.

Dados

Temos dois bancos distintos, um de treinamento e um de validação. Nestes bancos, há discursos de três presidentes: Lula, Dilma e Temer.

Nosso banco de treino se encontra dividido entre as classes da seguinte maneira:



Podemos ver que Temer é o presidente com menor número de discursos, mas a desproporcionalidade não parece ser grande o suficiente para gerar maiores problemas.

Table 2: Tabela 2 - Perfomance de cada pipeline em uma iteração

Pipeline	Accuracy	Bal. Acc.	Brier Score	Class. Error
baseline	0.5904762	0.5811094	0.8190439	0.4095238
baseline+stop	0.5809524	0.5682889	0.8380952	0.4190476
bigrams	0.5238095	0.5154369	0.9523810	0.4761905
bigrams+stopwords	0.5238095	0.5154369	0.9523810	0.4761905
trigrams	0.4190476	0.4285714	1.1619048	0.5809524
trigrams+stopwords	0.4190476	0.4285714	1.1619048	0.5809524

Testando diferentes pré-processamentos

Como forma de avaliar o impacto do pré-processamento na performance dos modelos, irei testar alguns pré-processamentos com um modelo (Naive-Bayes). As principais alterações no pré-processamento serão no número de ngrams (1, 2 e 3) e se há a opção do stopwords = "pt". Com isso, serão comparadas 3*2 = 6 pipelines diferentes de começo. A tabela 1 abaixo resume as diferentes pipelines:

Tabela 1 - Diferentes Pipelines de pré-processamento

Pipeline	ngram	stopwords
1 (Baseline)	1	não
2	1	sim
3	2	não
4	2	sim
5	3	não
6	3	sim

Benchmark (Pré-processamentos)

Nesta seção, faremos o benchmark dos pré-processamentos e compararemos os resultados do modelo utilizando o Naive Bayes como learner para todos os pré-processamentos. Devido à exigências do classificador, removi a coluna de data e transformei a variável de presidente em numérica, por ordem de mandato (Lula, 1; Dilma 2 e Temer, 3).

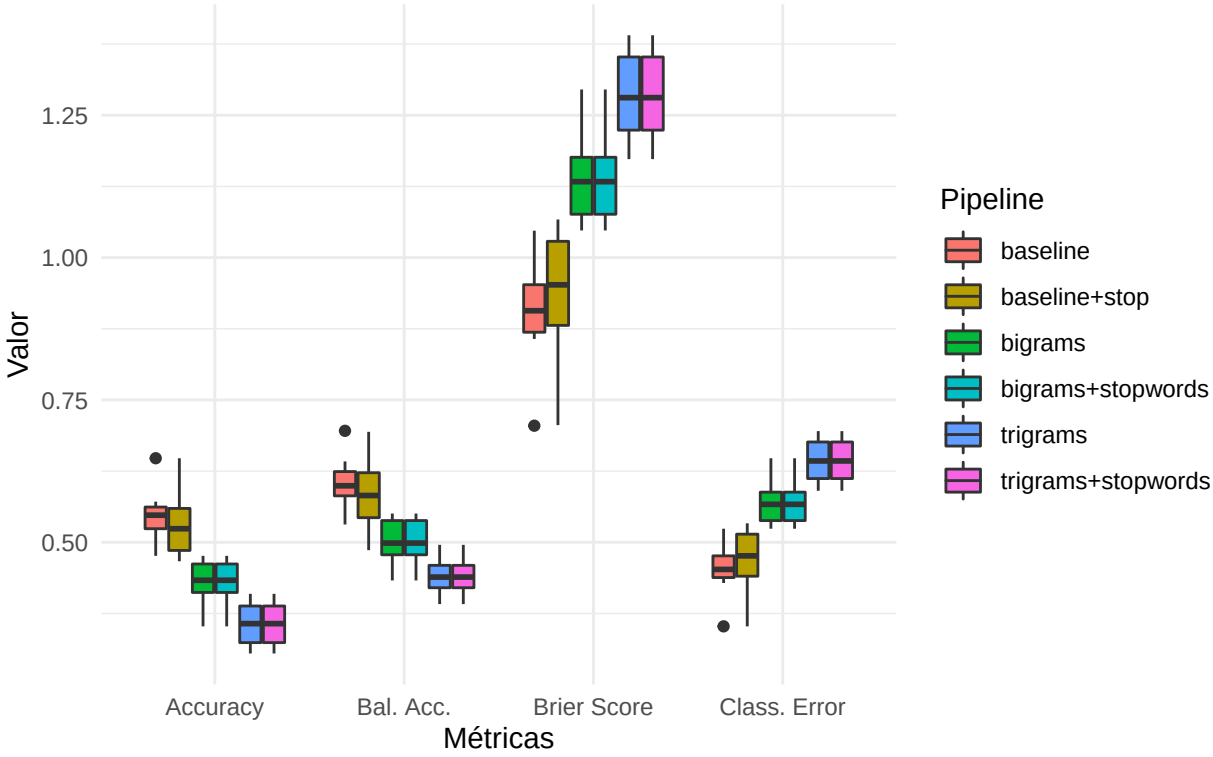
A tabela 2 mostra os resultados de uma iteração:

O pipeline que nos serve como baseline parece ter obtido o melhor resultado. De forma a confirmar isto, repetimos a operação 10x e computamos os resultados na figura 1:

Table 3: Tabela 3 - Perfomance de cada modelo em uma iteração

Modelo	Accuracy	Bal. Acc.	Brier Score	Class. Error
Naive Bayes (Baseline)	0.5714286	0.6283644	0.8571429	0.4285714
Tree	0.8761905	0.8744824	0.2076909	0.1238095
KNN	0.5047619	0.5282954	0.6152497	0.4952381
Forest	0.9523810	0.9349206	0.2360386	0.0476190

Figura 2 − Performance de cada pipeline

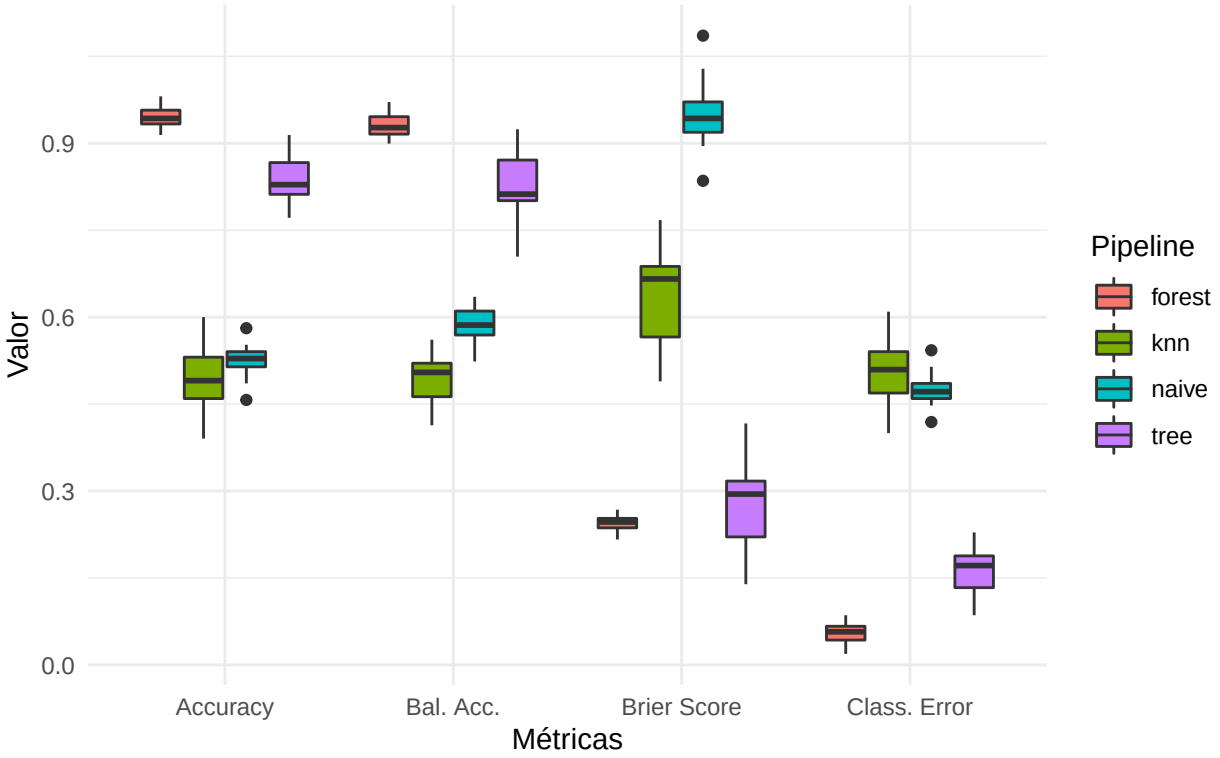


wer the Brier score is for a set of predictions, the better the predictions are calibrated.

o baseline com ngram = 1 e sem a correção de stop-words obteve melhores resultados. Portanto, ele será o utilizados para a comparação dos modelos. Na segunda parte, utilizaremos quatro modelos com o primeiro pipeline: o Naive Bayes, Tree, K-nearest Neighbors, e Random Forest. A tabela 3 mostra os resultados dos modelos em uma iteração.

Novamente, confirmamos o resultado em uma simulação de 10 iterações, representado na figura 2

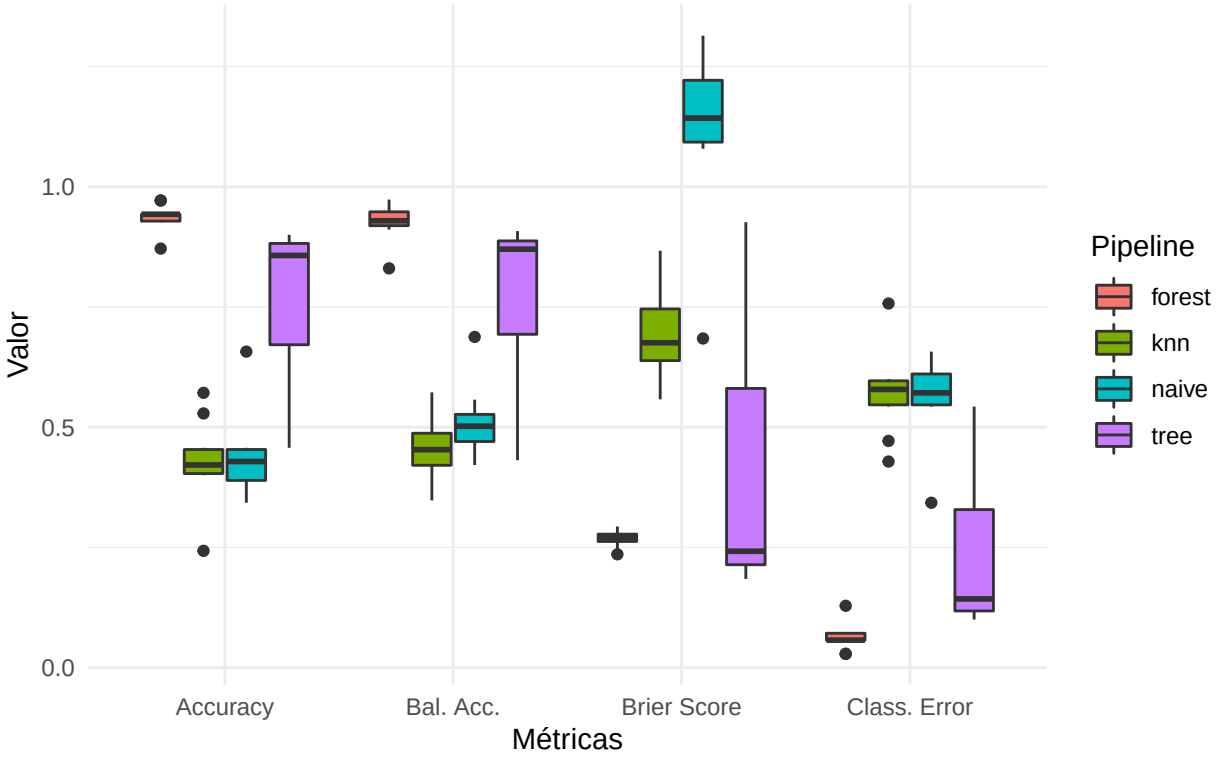
Figura 3 − Performance de cada modelo (ratio = 0.7)



Note: the lower the Brier score is for a set of predictions, the better the predictions are calibrated.

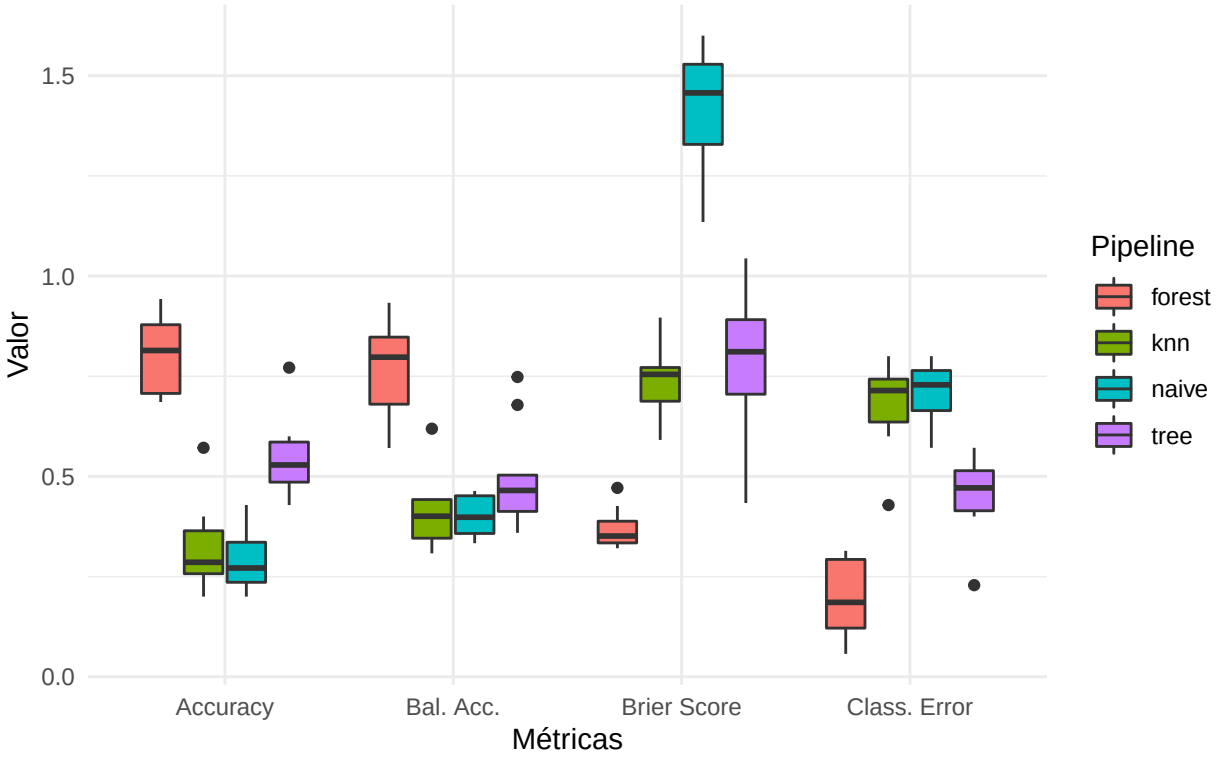
Como forma de melhorar a validação, iremos alterar o ratio do holdout para verificar a perfomance de cada modelo em diferentes tamanhos de bancos de treino. As figuras 3 e 4 mostram os resultados para todos os modelos em diferentes proporções de *holdout*.

Figura 4 − Performance de cada modelo (ratio = 0.8)



Note: the lower the Brier score is for a set of predictions, the better the predictions are calibrated.

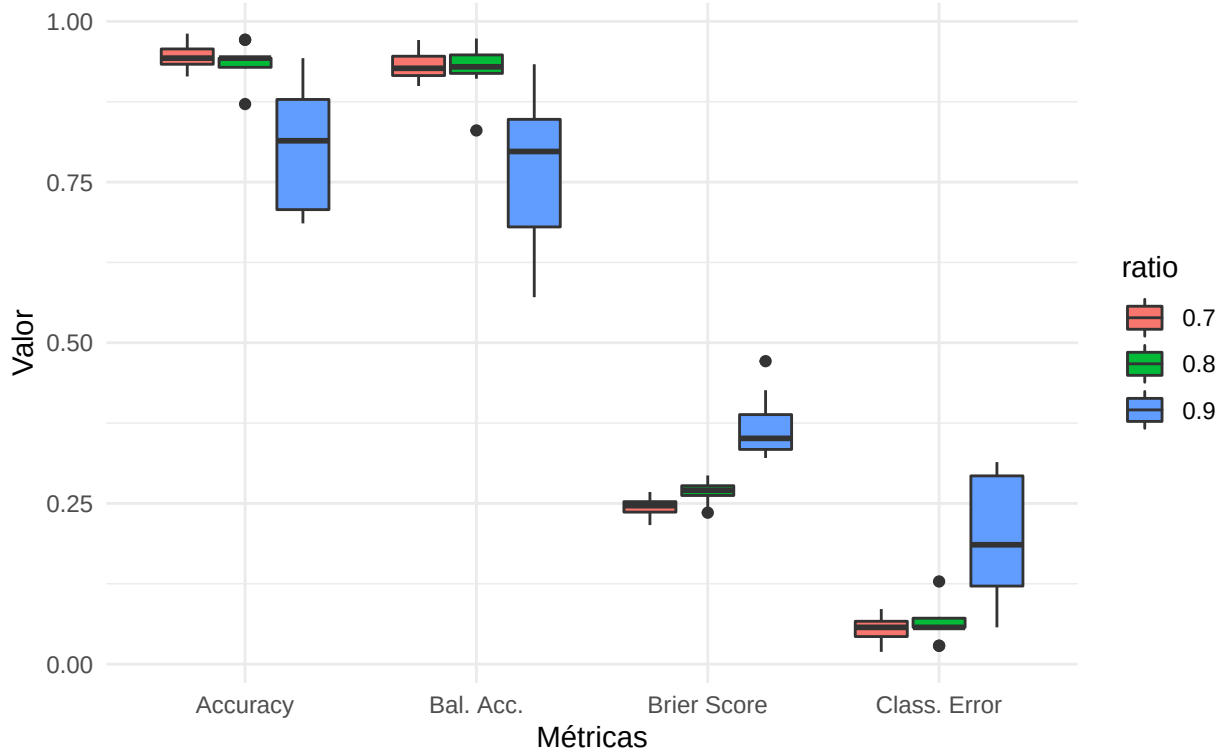
Figura 5 − Performance de cada modelo (ratio = 0.9)



Note: the lower the Brier score is for a set of predictions, the better the predictions are calibrated.

Olhando só para nosso melhor modelo (Figura 4) temos que:

Figura 6 − Performance do Forest por ratio



Note: the lower the Brier score is for a set of predictions, the better the predictions are calibrated.

Validação

Forest com o ratio de 0.7 parece ser nosso melhor modelo. Avaliaremos seus resultados no banco de validação:

```
## # A tibble: 25 x 4
##    discurso                                                    id pred  presi~1
##    <chr>                                                    <dbl> <fct> <chr>
##  1 "\nExcelentíssimo senhor Shinzo Abe, primeiro-ministro d~   137 2     Dilma
##  2 "\nFoto: Roberto Stuckert Filho/PR \n \nSenhor Laurent F~    90 2     Dilma
##  3 "\nExcelentíssimo senhor Paul Biya, presidente do Camero~   229 1     Lula
##  4 "\nEu quero cumprimentar, em primeiro lugar, a senhora M~     7 3     Temer
##  5 "\nQuero dirigir um cumprimento especial à Zoleka Mandel~    91 2     Dilma
##  6 "\nExcelentíssimo Senhor Álvaro Uribe, Presidente da Col~   364 1     Lula
##  7 "\nSão passados mais de cinco anos do início da crise fi~   153 2     Dilma
##  8 "\nÉ um grande prazer iniciar a primeira visita oficial ~   256 1     Lula
##  9 "\n\n\nQuero agradecer ao presidente Ahmadinejad pela ho~   254 1     Lula
## 10 "\nMeu caro amigo Raúl Castro, Presidente da República d~   374 1     Lula
## # ... with 15 more rows, and abbreviated variable name 1: presidente
```

Table 4: Tabela 3 - Predições do modelo grforest

id	pred	presidente
137	2	Dilma
90	2	Dilma
229	1	Lula
7	3	Temer
91	2	Dilma
364	1	Lula
153	2	Dilma
256	1	Lula
254	1	Lula
374	1	Lula
348	1	Lula
328	1	Lula
78	1	Lula
211	2	Dilma
118	2	Dilma
355	1	Lula
359	1	Lula
195	1	Lula
299	1	Lula
179	1	Lula
14	3	Temer
197	2	Dilma
306	1	Lula
26	1	Lula
244	1	Lula