Prevendo fragmentação partidária em eleições municipais

Yuri Lucatelli Taba

1 Introdução e apresentação do problema de pesquisa

Ao menos desde Duverger (1951), estudos sobre fragmentação partidária destacam o comportamento estratégico de eleitores e elites por meio da coordenação. Eleitores evitam o desperdício de votos, culminando no voto estratégico, e elites partidárias respondem a este comportamento limitando a oferta de candidaturas, ação conhecida pelo termo "entrada estratégica".

Estas ações ocorrem em momentos separados no tempo. Primeiro, elites tomam a decisão sobre a entrada em determinada competição. As possibilidades colocadas neste momento envolvem a entrada solo, em uma coalizão, ou a retirada estratégica. Estas ações são tomadas levando em consideração o voto estratégico, que ocorre no momento do voto ser depositado pelo eleitor. Neste intervalo de tempo, que corresponde à campanha eleitoral, há diversas oportunidades para eleitores buscarem evitar o desperdício do voto, se concentrando em torno de candidaturas viáveis, e para elites se coordenarem, por exemplo, direcionando recursos partidários para as candidatas com chances reais de vitória, evitando o desperdício de tais recursos (Zhirnov, 2016; Ziegfeld, 2021).

Em ambos os casos há uma condição informacional a ser preenchida: eleitores e elites devem estar bem informados sobre quem são os candidatos viáveis e inviáveis. Nos termos de Rozenas e Sadanandan (2017), trata-se de uma hipótese informacional que pode ou não ser confirmada de acordo com variações no contexto sob o qual os atores estão inseridos. Como exercício inicial para verificar o efeito do contexto informacional sobre a coordenação de atores políticos em eleições municipais no Brasil, utilizo as prestações de conta das campanhas eleitorais das candidaturas a vereador entre 2004 e 2016 como *proxy* para este contexto informacional.

Dadas as limitações de uma abordagem convencional de modelos de regressão para verificar o efeito dos gastos de campanha sobre a fragmentação partidária, sigo a proposta de Streeter (2019) e busco verificar se a inclusão destes gastos como *feature* em diferentes técnicas de aprendizado de máquina permite a previsão dos níveis de fragmentação nos municípios brasileiros. De maneira preliminar, os testes realizados verificam a diferença da precisão da previsão realizada entre as técnicas que incluem ou não os gastos de campanha como componentes dos modelos de aprendizado de máquina. Resultados melhores nos modelos que incluem estas informações, podem constituir evidência favorável à hipótese levantada: gastos de campanha geram maior fluxo de informação durante as campanhas eleitorais, influenciando o grau de fragmentação partidária ao nível do distrito. [1]

2 Apresentação dos dados

Neste exercício inicial, utilizarei uma base de dados com o número efetivo de partidos em votos (NEP-V) para os pleitos municipais realizados entre 2004 e 2016. Nesta base, cada observação corresponde a um município em uma eleição dentro deste período. A tabela ainda conta com informações do total de votos depositados, a unidade da federação, o tamanho da população, a magnitude do distrito e o total de gastos declarados ao Tribunal Superior Eleitoral (TSE) por todas as campanhas eleitorais para vereador em cada município. Os resultados eleitorais foram extraídos pelo pacote `electionsBR` e as informações de gastos de campanha do repositório de dados do TSE.

Os gastos de campanha aparecem nos modelos treinados de duas maneiras distintas: o gasto total realizado por todas candidatas de todos partidos em cada município e, seguindo a proposta de Thomsen (2023), o número efetivo de partidos medido pelos gastos de campanha (NEP-G), calculado da mesma maneira que o NEP em votos, que permite verificar o grau de concentração dos gastos em poucos ou muitos partidos.

A Figura 1 apresenta a distribuição das principais variáveis de interesse.

Com a finalidade de focar apenas em eleições que são regidas exatamente sob as mesmas instituições eleitorais, neste exercício inicial uso apenas os pleitos com nove cadeiras em disputa, o mínimo aceito constitucionalmente, que representa 17.320 eleições no período (em um universo de 22.247 disputas - 77,85%). A média de despesas contabilizadas na prestação de contas feita ao Tribunal Superior Eleitoral (TSE) é de 146.136,00 reais, com desvio padrão de R$ 247.013,00.

[1] Aqui restaria entender se dá para saber se a relação é negativa, ou seja, se mais gastos de campanha geram maior coordenação e, portanto, menores índices de fragmentação.

Dentre os municípios com nove cadeiras em disputa, excluí da amostra a eleição de 2008 em Maracajá, Santa Catarina, tendo em vista um provável registro errado, que atribui a um gasto em combustíveis um valor superior a 23 milhões de reais. Dos distritos que permaneceram, nove não possuem registros de gastos de campanha. O valor mínimo, com exceção destes municípios sem gastos, é de oito reais em Nova Olinda do Maranhão (MA) em 2004, e o máximo de 5,38 milhões de reais em Lucas do Rio Verde (MT) em 2016. Nota-se também que são raros os municípios que registram gastos superiores a 500 mil reais.

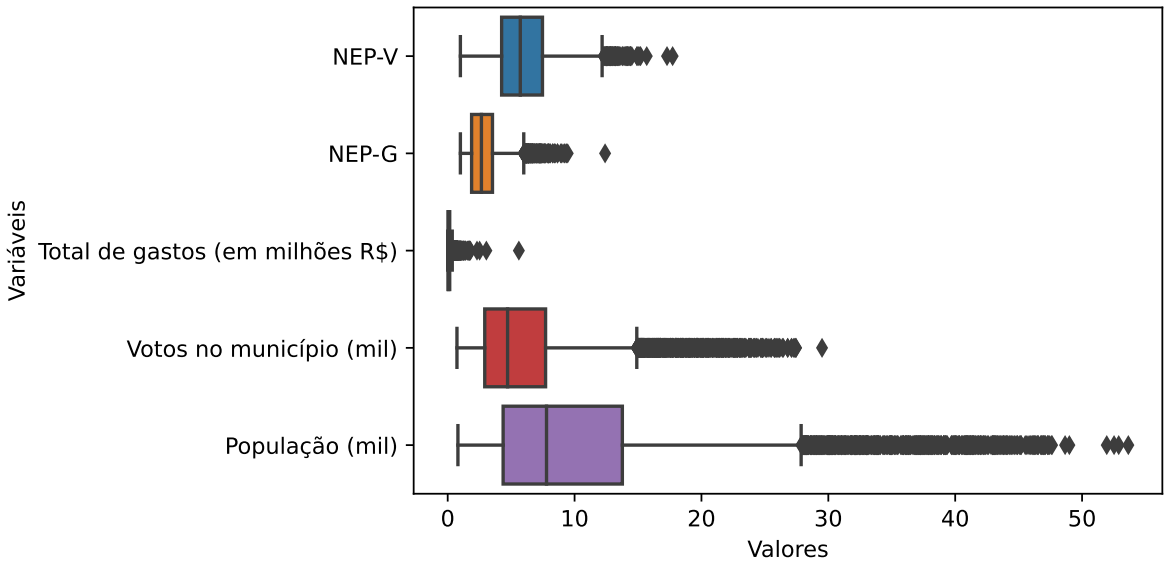


Figura 1: Boxplot variáveis de interesse

3 Modelos

Para estimar o nível de fragmentação partidária nos municípios, treino dois modelos de *ensemble* que possuem árvores de decisão como base para o algoritmo: Random Forest e XGBoost.

Os modelos foram treinados com os dados das eleições de 2004 a 2012 e validados em uma base de validação com 30% dos dados. Após o treino e o *tuning* dos parâmetros, escolhi o modelo com melhor desempenho para prever a fragmentação partidária nas eleições de 2016.

Como etapas de pré processamento, além da divisão das amostras de treino e validação, ainda utilizei *one hot encoding* para adaptar a informação da unidade da federação de

cada município e `SimpleImputer` para tratar valores faltantes nas *features* escolhidas, incluindo a mediana de cada variável. Por fim, levando em consideração as diferentes escalas de medida entre os componentes, usei `StandardScaler` para padronizar as observações.

```
<5561x32 sparse matrix of type '<class 'numpy.float64'>'
    with 33366 stored elements in Compressed Sparse Row format>
```

3.1 Random Forest

O primeiro modelo treinado é um regressor Random Forest, um *ensemble* sobre árvores de decisão. Este estimador treina um número determinado de árvores de decisão em diversas sub amostras, usando *averaging* para definir a predição final.

Como parâmetros definidos já na inicialização do modelo, `random_state` configura e controla a aleatoriedade do *bootstraping* e sub amostras realizadas. Também testo diferentes aplicações dos parâmetros `n_estimators` e `criterion`. O primeiro determina o número de árvores de decisão utilizados, o segundo controla a função que mede a qualidade da divisão entre as árvores. A Figura 2 apresenta o resultado do *root mean squared error* (RMSE) para cada parâmetro. Com relação ao número de árvores de decisão presentes no modelo, nota-se que não há ganho significativo a partir das 110 árvores, portanto configuraremos o parâmetro do modelo para este valor. Já sobre o critério que avalia a qualidade da divisão entre as árvores, o parâmetro `absolute_error` apresenta um RMSE menor quando comparado a suas alternativas.

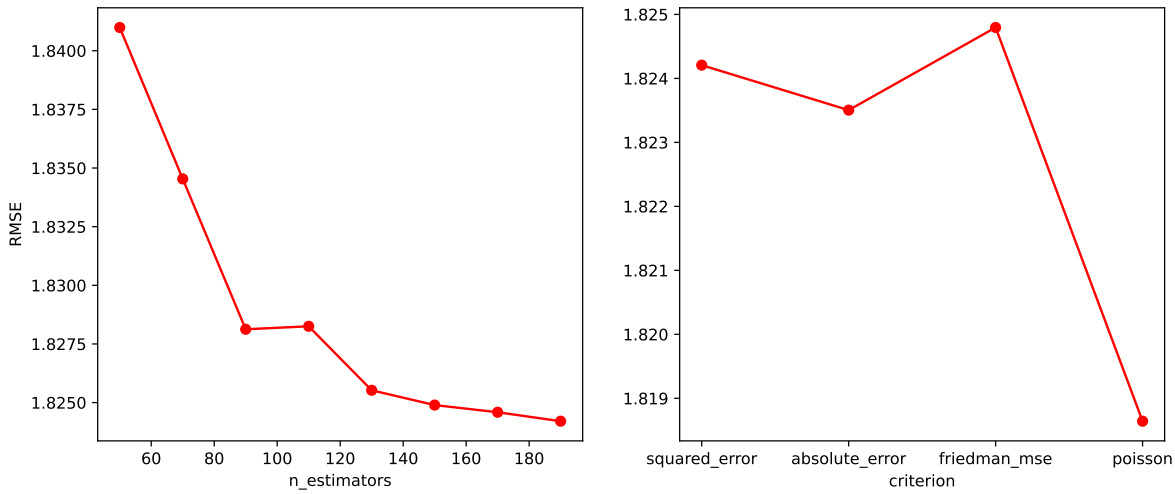


Figura 2: Parameter tuning do modelo Random Forest

Combinando os parâmetros testados acima em um modelo, atingimos um RMSE de 1,894. Este valor será comparado ao mesmo resultado do próximo modelo a ser treinado: `XGBoost`.

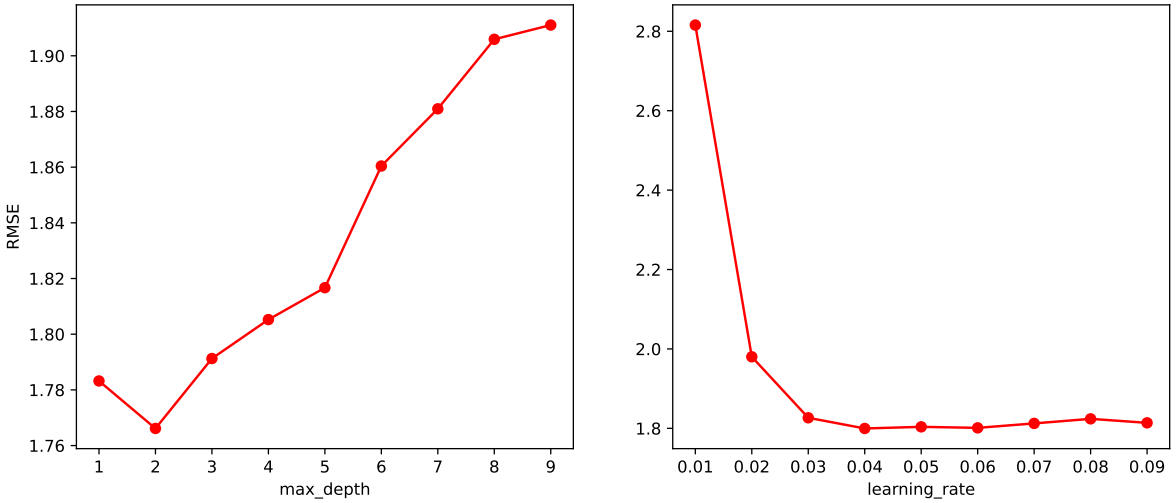
3.2 XGBoost

`XGBoost` significa Extreme Gradient Boosting e, de maneira geral, também é um *ensemble* sobre diversas árvores de decisão. O modelo é treinado utilizando um algoritmo de *gradient descent optimization*, que minimiza a *loss gradient* durante o próprio treino do modelo.

De maneira similar ao modelo anterior, este também foi treinado sob diferentes configurações de dois parâmetros centrais para o modelo: `max_depth`, que mede a profundidade máxima que cada árvore de decisão pode chegar; e `learning_rate`, que controla o tamanho da etapa em que o algoritmo atualiza os pesos utilizados no modelo. A Figura 3 apresenta os resultados em RMSE, os melhores parâmetros são `max_depth = 3` e `learning_rate = 0.4`.

```
Text(0.5, 0, 'learning_rate')
```

Parameter tuning do modelo XGBoost



A combinação destes parâmetros resulta em um RMSE de 1,881 na amostra de validação, ligeiramente melhor que a performance do modelo Random Forest.

3.3 Aplicação do melhor modelo na base de 2016

Por fim, utilizo o modelo XGBoost treinado na seção anterior para prever a fragmentação partidária (NEP-V) nas eleições municipais de 2016. O modelo foi empregado em dois conjuntos distintos de dados. As observações são exatamente as mesmas em ambos, o que os diferencia é a presença ou não da *feature* que apresenta o total de gastos de campanha naquele município.

A ideia central é verificar se a inclusão dos gastos de campanha neste modelo resulta em uma capacidade preditiva melhor do que no modelo sem esta variável.

Os modelos foram treinados utilizando o mesmo conjunto de variáveis das seções acima, respeitando o mesmo procedimento de pré processamento dos dados: preenchimento de *missing values* com a mediana da variável, padronização e *one hot encoding* para as unidades da federação.

A aplicação do modelo XGBoost repetindo os melhores parâmetros definidos anteriormente resultou em um RMSE de 2,319 para a versão da base que não inclui o total de gastos de campanha como *feature*.

Já na versão que inclui esta variável, obtive um RMSE de 2,028, 12,5% menor que o RMSE calculado pelo modelo sem as variáveis de gasto de campanha.

4 Considerações finais

Ainda que a performance do modelo que inclui os gastos de campanha tenha sido melhor, sugerindo alguma relação relevante entre estes gastos e a fragmentação partidária, os resultados encontrados pela aplicação do modelo sobre uma base de dados nova apresentam uma performance pior em termos de erro, com RMSE maiores, com relação ao encontrado nas amostras de treino e validação. Isso é um sintoma de que o modelo treinado está gerando *overfitting* dos dados. Este problema precisa ser melhor enderaçado em versões seguintes deste exercício, sobretudo testando novas configurações para os diversos parâmetros que o XGBoost possui.

Este será o caminho das próximas versões deste trabalho, também pensando em incluir novas variáveis que possam nos ajudar a prever melhor o *target*.

5 Referências

Duverger, Maurice. (1970 [1951]). Os partidos políticos. São Paulo: Zahar.

Rozenas, Arturas; Sadanandan, Anoop (2017). Literacy, Information, and Party System Fragmentation in India. Comparative Political Studies, 51, 5.

Streeter, Shea (2019). Lethal Force in Black and White: Assessing Racial Disparities in the Circumstances of Police Killings. The Journal of Politics, 81, 3.

Thomsen, Danielle (2023). Competition in Congressional Elections: Money versus Votes. American Political Science Review, 117, 2.

Zhirnov, Andrei (2016) Electoral coordination in India: The role of costly campaign communication. India Review, 15:4, 359-378

Ziegfeld, Adam (2021). What accounts for Duverger's law? The behavioral mechanisms underpinning two-party convergence in India. Electoral Studies. 73.