BLOWING IN THE WIND: THE EFFECT OF COVID ON POLITICAL SUPPORT IN BRAZIL

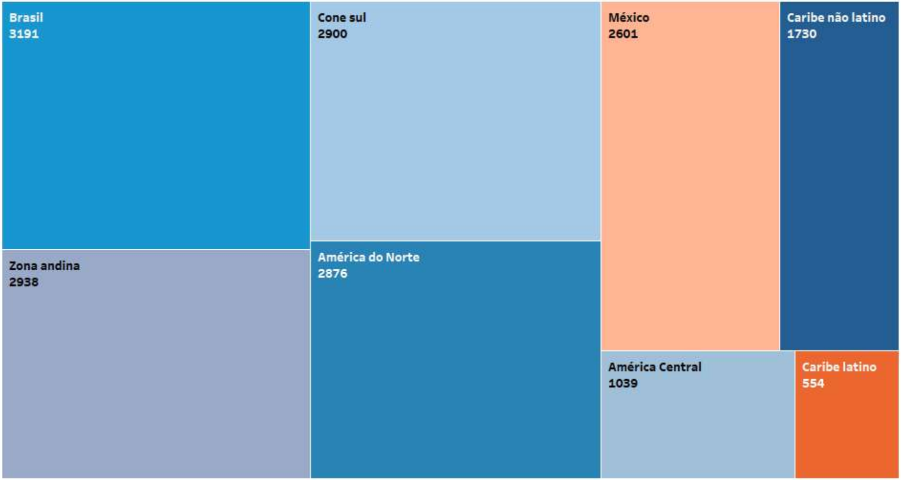
Rafael Pintro Schmitt

October 22, 2023

BACKGROUND

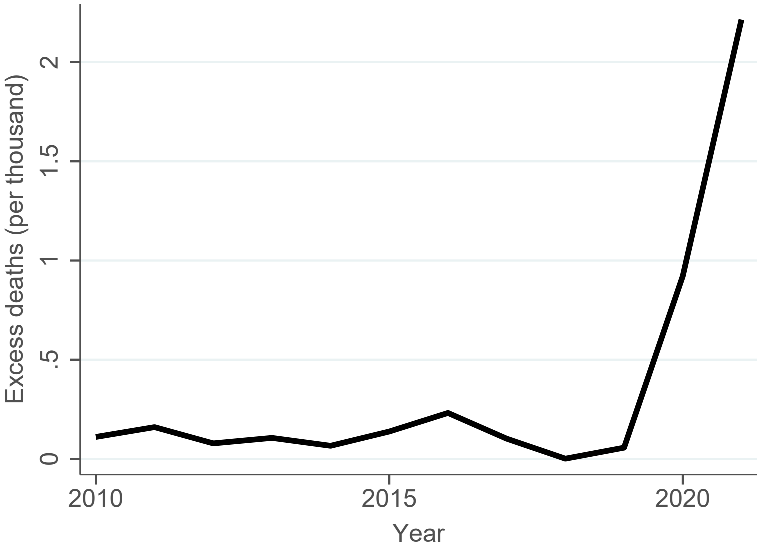
- 701,494 confirmed COVID-19 deaths in Brazil.
- Bolsonaro during the pandemic:
  - Several statements downplaying the extent of the pandemic.
  - Vaccine skepticism.
  - *CPI da COVID*: parliamentary comission of inquiry, highly publicized.
- More people died of COVID in regions with strong Bolsonaro support (Cabral et al. 2021). Event studies show that Bolsonaro's rhetoric also had a higher impact on reducing social distancing in such places (Ajzenman et al., 2020).

BACKGROUND - DEATHS PER 100,000 INHABITANTS



Extracted from: https://hia.paho.org/es/covid22-saude

EXCESS DEATHS OVER TIME



RESEARCH QUESTION

- What was the effect of COVID deaths at the city level on Bolsonaro's electoral performance?
  - Look at the difference in voting outcomes in 2022 versus 2018.
  - Focus on local death rates rather than national outcomes.
- Did people's beliefs about the ex-president shift due to his handling of the pandemic?
- Could it be just an incumbent effect?

ESTIMATION AND POTENTIAL ISSUES I

- Want to estimate β in:

$$\Delta_i = \alpha + \beta \text{excess\_mortality}_i + \delta X_i + \varepsilon_i \tag{1}$$

Where:

  - Δ_i is the difference in PT (workers' party) runoff vote-share from 2018 to 2022 (comparable to previous election cycles).
  - $\text{excess\_mortality}_i$ is the number of yearly deaths per thousand above a linear extrapolation of the trend of deaths in city i (in 2021).
  - X_i is a matrix collecting (potentially unobservable/omitted) confounders.

ESTIMATION AND POTENTIAL ISSUES II

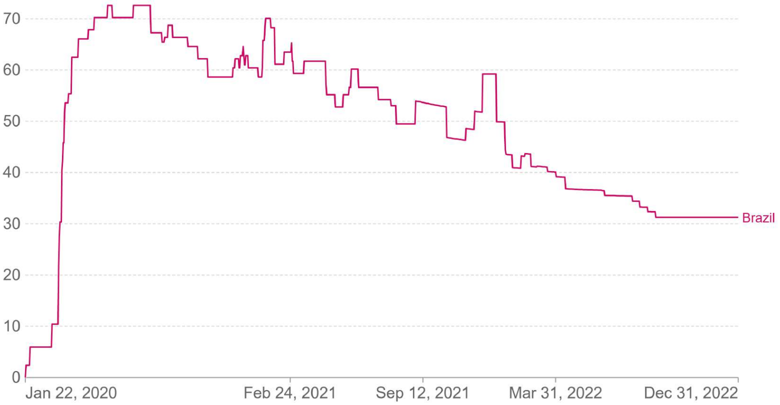
- Main concern for identification:
  - Cities in which people assign a higher "cost" to lock-downs or are vaccine hesitant may have higher death rates $\Rightarrow$
  - Approval of Bolsonaro's policies and attitude in such cities may yield a positive correlation between deaths and increased Bolsonaro support (Δ_i)
- Idea: wind has been found to reduce outdoors COVID contagion in some contexts (Rendana, 2020; Clouston et al, 2021).
- As social distancing and lock-down measures declined in 2021, and information on best practices became widespread (e.g. meeting in open areas), wind should start mattering more.

LOCK-DOWN STRINGENCY INDEX

COVID-19 Containment and Health Index

Our World in Data

This is a composite measure based on thirteen policy response indicators including school closures, workplace closures, travel bans, testing policy, contact tracing, face coverings, and vaccine policy rescaled to a value from 0 to 100 (100 = strictest). If policies vary at the subnational level, the index is shown as the response level of the strictest sub-region.



Source: Oxford COVID-19 Government Response Tracker, Blavatnik School of Government, University of Oxford – Last updated 2 May 2023
OurWorldInData.org/coronavirus •

DATA

DATA

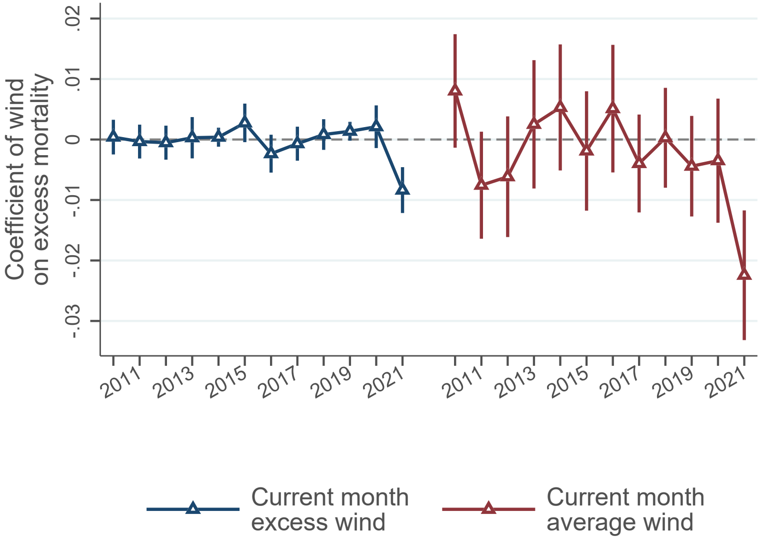
- *Sistema de Informações sobre Mortalidade (SIM)*: death-counts by municipality (1996-2022).
- *Instituto Brasileiro de Geografia e Estatistica (IBGE)*: Micro- and mesoregion definitions, population estimates per city, covariates from 2010 Census.
- *Tribunal Superior Eleitoral (TSE)*: National- and state-level election results.
- TerraClimate (Abatzoglou et al., 2018): high resolution global weather dataset, providing monthly wind data on a $.04° \times .04°$ spatial resolution grid ($.01° \approx 1.11$km).

*Resources used: Basedosdados, DataZoom, brclimr (R-package).

INSTRUCTION CONSTRUCTION

WIND AND EXCESS MORTALITY

Coefficient of (excess) wind on excess mortality. City and region-period FE's.



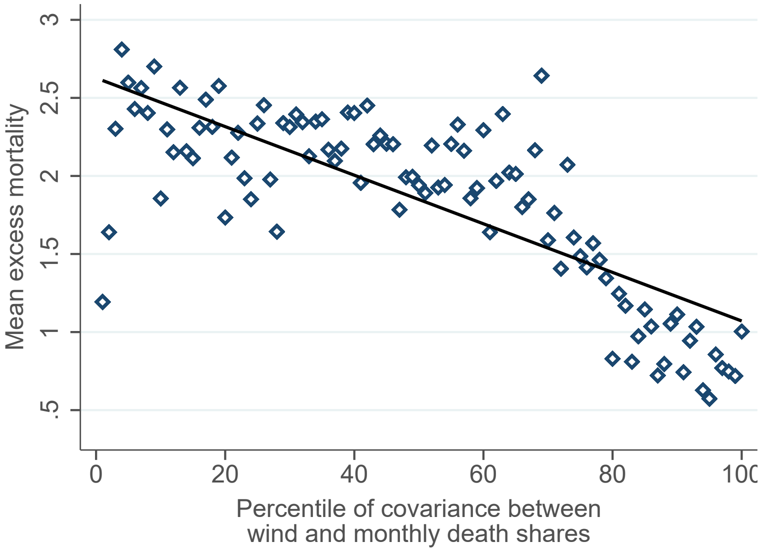
ISSUE

- Pandemic dynamics introduce month-to-month dependency of excess deaths. Together with wind auto-correlation, this can lead to coefficients flipping.
  - E.g. high death month 1 followed by low death month 2. High wind by high wind. $\Rightarrow$ Coefficient of month 2 may flip (even when causal effect is constant).
- The relationship is lost at the yearly level. To estimate the impact of 2021 excess mortality on voting outcomes in 2022, we need an instrument that affected yearly excess mortality.

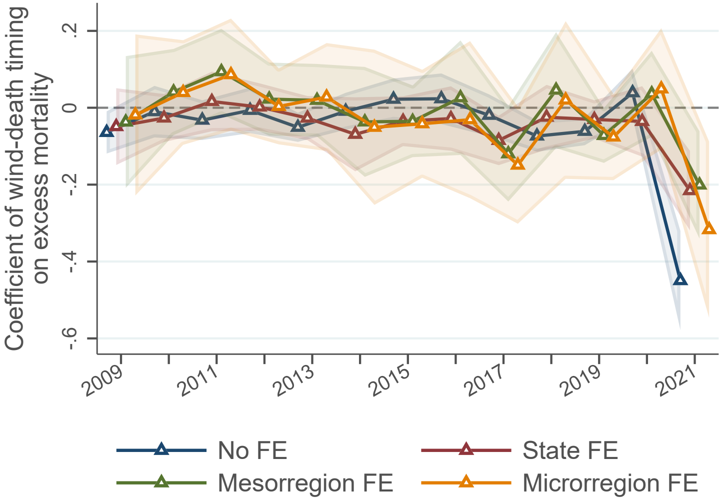
SOLUTION

- Use the *timing* of wind:
  - Cities that had higher wind speeds during COVID waves in 2021 should have lower overall excess mortality in 2021.
  - Instrument: the covariance between wind speeds in month m and the leave-one-out share of national yearly deaths that occurred in month m.
- National monthly death shares reasonably not affected by *local* wind dynamics - eliminates flipping coefficients issue.
- Wind-death covariance depends on the *city level* of wind only through (possibly) larger wind fluctuations (higher variance of the instrument).

FIRST-STAGE ASSOCIATION



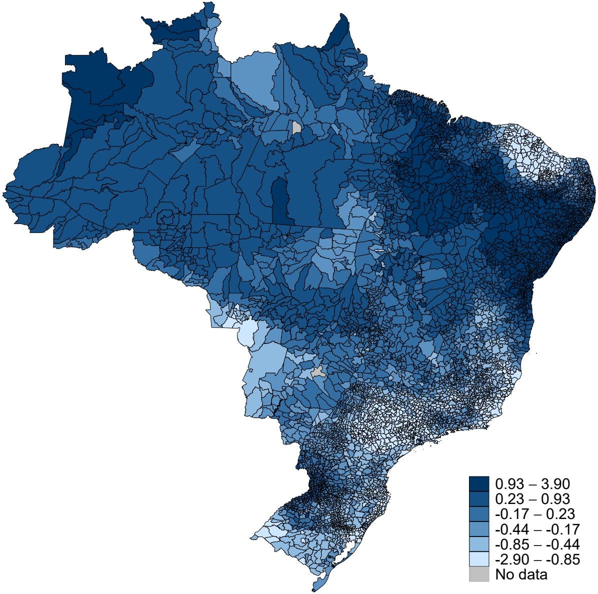
WIND-DEATH TIMING AND EXCESS MORTALITY



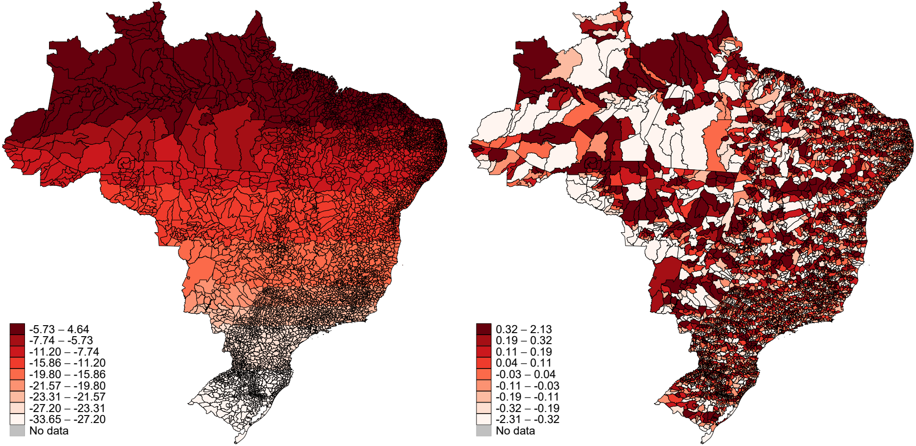
SEEMINGLY STRAIGHT FORWARD...

- If we want to evaluate the impact of COVID deaths on voting outcomes:
  - We know that high wind-death covariance $\Rightarrow$ less deaths.
  - If high wind-death covariance $\Rightarrow$ less votes for PT, then we have a good argument that more deaths $\Rightarrow$ more votes for PT.
- To argue for causality, we just need that (1) wind-death covariance itself doesn’t affect voting outcomes except through changes in death rates (2) relatedly, that wind-death covariance is "randomly assigned" - that is, that there is no *selection* of cities into treatment based on (un)observable characteristics.

HOWEVER... INSTRUMENT GEOGRAPHY



POTENTIAL CONFOUNDER



(a) Without FE

(b) With microregion FE

ISSUE

- Controlling for fine-grained fixed effects may not be enough if both the instrument and a potential confounder have a geographic gradient (e.g. north-south).
- The two groups, (1) high-wind during COVID waves (treated) and (2) low-wind during COVID waves (control), are not comparable!
- Indeed, even controlling for state fixed effects:
  - The two groups of cities are different across several covariates.
  - Have different voting paths (i.e. the way vote-shares vary over time is different across groups).

SOLUTION

- The two groups still carry information about each other.
- Traditionally, one would perform some kind of matching (weight units in the control group so that it resembles the treated, etc).
- Idea: using a graph neural network (GNN) to predict the changes in the vote-share for the treated group using only units in the control group.
  - Construction of a counterfactual as a prediction problem.
  - If we can estimate the treated group's outcomes with the controls reliably before treatment, but not after treatment, we have evidence of a causal effect!

OUTLINE

- Construct a graph/network for each year, by linking each city to its 10 closest neighbors (defined by geographical distance, which is itself used as an edge attribute).
- Collect data on Brazilian municipalities' presidential vote-shares from 2010 to 2022. Covariates from the 2010 census. Use them as node characteristics.
- Note: I include the outcome (changes in vote-share) as a covariate. This will become clearer when I explain the training of the model.
- Note II: I take the outcome as the percentile in the distribution of changes in vote-share (100 equally sized bins).

OUTLINE II

- For every year, split the sample into training (2500 cities), validation and test sets (remaining cities).
- Train the model only on 2010, 2014 and 2018 data (treatment doesn't affect the training).
- Objective of the model: given a sample of Brazilian cities' changes in PT vote-share, predict the outcome for all other cities (node-level classification task).
- How: at every step of training, drop the information for half of the municipalities, and define the loss function to reduce *overall* losses - the model must learn to predict the omitted cities' changes in vote-share (given also neighbors' outcomes)!

THE MODEL

```
class GAT(nn.Module):
    def __init__(self):
        super().__init__()
        torch.manual_seed(j)
        self.conv1 = GATConv(in_features, in_features, heads=2)
        self.conv2 = GATConv(in_features*2, in_features, heads=2)
        self.lin1 = Linear(in_features*2, out_features)

    def forward(self, x, edge_index, edge_attr):
        #make drops start at 0.9 and decrease by 0.05 every 10 epochs, until it reaches 0.1
        drops = 0.9 - (0.05 * (epoch // 10))
        if drops < 0.5:
            drops = 0.5
        x = F.dropout(x, p=drops, training=self.training)
        x = self.conv1(x, edge_index, edge_attr)
        x = F.relu(x)
        x = self.conv2(x, edge_index, edge_attr)
        x = self.lin1(x)
        return x
```

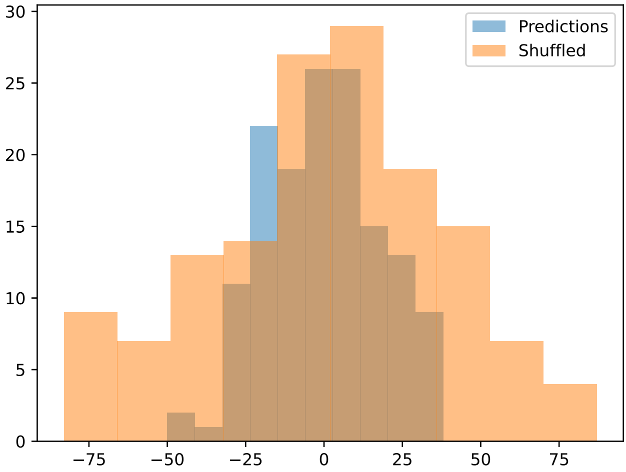
THE TRAINING

```
optimizer = torch.optim.Adam(model.parameters(), lr=0.01, weight_decay=5e-4)
criterion = torch.nn.CrossEntropyLoss()

def train():
    #loop over items of datalist
    for data in data_list:
        model.train()
        optimizer.zero_grad()  # Clear gradients.
        out = model(data.x, data.edge_index, data.edge_attr)  # Perform a single forward pass.
        # Compute the loss solely based on the training nodes.
        loss = criterion(out[data.train_mask], data.y[data.train_mask])
        loss.backward()  # Derive gradients.
        optimizer.step()  # Update parameters based on gradients.
    return loss

for epoch in range(1, 100):
    drops = 0.9
    loss = train()
    print(f'Epoch: {epoch:03d}, Loss: {loss:.4f}')
```

SUMMARY

- Very simple GNN:
    1. A dropout layer (with varying dropout rates).
    2. Two attention layers, with a RELU in between.
    3. A linear layer.
- Training:
    1. At every epoch, perform a forward pass and take an optimizer step for each year in the training data (2010-2018).
    2. 100 epochs - 300 forward passes and optimization steps.

PERFORMANCE



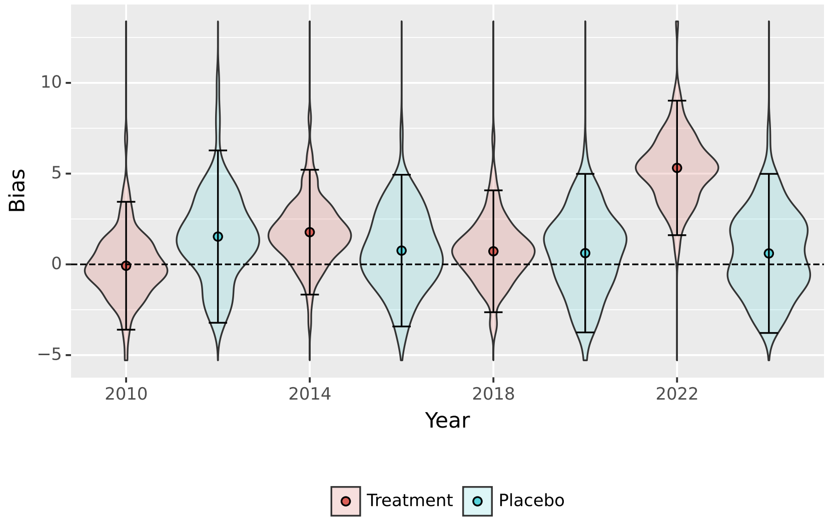
PERFORMANCE II

- The model has a "fake" R^2 around 0.8 (1 - MSE of the model/MSE of random guesses).
- It’s accuracy is usually 2 to 3 times higher than random guesses.
- The perfomance on the training and test sets is similar.
- The model is on average unbiased (more on this later).

TREATMENT EFFECT EVALUATION

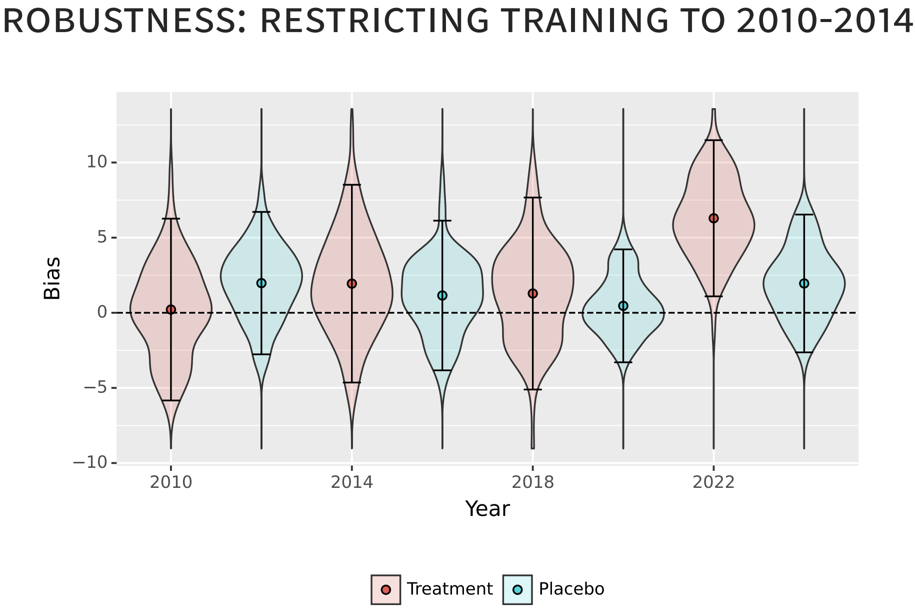
- We can finally turn to the evaluation of treatment effects. I define as "treated" those cities with wind-death covariance above the median in their state.
- For each year (2010, 2014, 2018, 2022) I omit the outcomes of the treated cities and feed the data into the network. I get predictions for each treated city which are based only on the control cities.
- For each year, I take the bias (the average error). I then repeat the training and evaluation 100 times.
- I perform the same exercise randomly selecting "treatment" cities at every training round and year, as a placebo test.

RESULTS: BIAS DISTRIBUTION BY YEAR



TAKEAWAY

- The model is on average unbiased before treatment for the treated, and unbiased before/after treatment for the placebo.
- The model is positively biased in 2022 (after treatment). That is, the model predicts a higher shift towards the PT than what actually happened.
- If we take the model's prediction as a counterfactual, under no COVID/no differential wind-death timing, treated cities would've had a higher shift in vote-share towards the PT.
- That is, higher wind-death covariance $\Rightarrow$ lower deaths $\Rightarrow$ lower delta PT vote-share.
- COVID deaths at the city-level caused less votes for Bolsonaro.



ROBUSTNESS: RESTRICTING TRAINING TO 2010-2014
Bias
10
5
0
−5
−10
2010
2014
2018
2022
Year
Treatment
Placebo

CONCLUSION

- Neural networks can provide credible counterfactuals even when treatment and control groups are different and follow different pre-trends in the outcome variable.
- I provide strong evidence that COVID deaths at the city level caused people to vote less for Bolsonaro - channels may be economic, beliefs, etc.
- Limitations:
  - Little interpretability of the estimates, in particular of the confidence intervals and quantification of uncertainty.
  - Relatedly: lack of asymptotic properties, though conformal prediction is useful for longer time series.