Replication Package

Optimal Environmental Targeting

in the Amazon Rainforest

(The Review of Economic Studies)

Juliano Assunção, Robert McMillan,
Joshua Murphy, and Eduardo Souza-Rodrigues*

August 4, 2022

1 Overview

This replication package contains the data and the code to generate the figures and tables presented in the paper and online appendix. The tex files for the tables, and the files for the figures, are included as well.

2 Data Availability Statements

The paper uses the following sources of data:

1. Annual measures of land cover in the Brazilian Amazon come from the Brazilian government's satellite-based forest monitoring program, known as PRODES, developed by Brazil's National Institute for Space Research (INPE). The data are publicly available at both pixel and municipality levels at the website of INPE (2017a).

2. The official list of Priority municipalities (with precise dates for entry and exit) comes from the Brazilian's Ministry of the Environment (MMA, 2017).

*The code and this "read me" file were written together with Faisal Ibrahim, who provided outstanding research assistance. Contact information: Assunção – Departamento de Economia, PUC-Rio, Rua Marques de Sao Vicente, 225/F210, 22453-900 - Rio de Janeiro/RJ, Brazil (email: `juliano@econ.puc-rio.br`); McMillan – Department of Economics, University of Toronto, 150 St. George Street, Toronto, ON M5S 3G7, Canada, and NBER (email: `mcmillan@chass.utoronto.ca`); Murphy – Natural Resources Canada, 580 Booth Street, Ottawa, ON K1A 0E4, Canada (email: `joshua.murphy4@canada.ca`); Souza-Rodrigues – Department of Economics, University of Toronto, 150 St. George Street, Toronto, ON M5S 3G7, Canada (email: `e.souzarodrigues@utoronto.ca`).

3. The geographic boundaries of the Brazilian municipalities, the Amazon biome, and the Brazilian and South American territories come from the National Highway Plan from the Brazilian's Ministry of Transportation (PNLT, 2011).[1]

4. Information on protected areas are obtained from the National Register of Conservation Units, maintained by the Brazilian Ministry of the Environment (CNUC, 2016), and the National Indian Foundation, which is part of the Ministry of Justice and Public Security (FUNAI, 2016).

5. Longitudinal data on precipitation and temperature at the grid-level were calculated by Matsuura and Willmott (2018a,b).

6. Annual data on municipalities' gross domestic product come from the Brazilian Institute of Geography and Statistics' regional account system (IBGE, 2017c).

7. Annual data on the number of cattle and crop area per municipality come from the IBGE's surveys: the Municipal Livestock Survey (IBGE, 2017a) and the Municipal Crop Survey (IBGE, 2017b).

8. Data on prices of the main commodities – beef, soy, rice, corn, cassava, and cane sugar – are taken from the State Secretariat for Agriculture and Food Supply (SEAB-PR, 2019).

9. Data on agricultural suitability come from the Food and Agriculture Organization's Global Agro-Ecological Zones project (FAO, 2021).

10. Straight-line distances from the municipal seats to the nearest port were calculated by Souza-Rodrigues (2019), using data from the National Highway Plan, produced by the Brazilian Ministry of Transportation (PNLT, 2011). The calculated distances can be downloaded from the supplemental material in Souza-Rodrigues (2019).

11. The amount of carbon stock above ground is calculated by Baccini et al. (2012). Although the original data does not seem to be publicly available anymore, one can access the data product that expands on the methodology presented in Baccini et al. (2012) to generate a global map of aboveground live woody biomass density here: `https://data.globalforestwatch.org/datasets/gfw::aboveground-live-woody-biomass-density/about`.

12. Data on the number of fines issued for environmental offenses come from the Brazilian Protection Agency (IBAMA, 2016).

[1] Other publicly available sources exist; e.g., see `http://www.dpi.inpe.br/Ambdata/English/adm_units.php`.

13. Forest clearing alert data come from the Real-Time System for the Detection of Deforestation (DETER), developed and operated by the space agency INPE (INPE, 2017b).

14. Data on all rural loan contracts negotiated by farmers and banks are collected by the Brazilian Central Bank (BCB, 2020). The data are accessible via formal request to the Central Bank (`https://www.bcb.gov.br/en`) – noting that information on individual loans are confidential. (Total value of loans aggregated at the municipality-level, as used in this paper, however, are not subject to confidentiality issues.)

All data are publicly available, and we certify that the authors of the manuscript have legitimate access to and permission to use the data used in this manuscript.

3 Computational Requirements

1. `Stata 17`. The extra required packages are listed at the top of the script – the latest versions can be installed using the commands described there (to do so, one needs to uncomment the appropriate lines). We have also included the original versions of the programs used in the paper in this replication package (see more below). The required packages are the following:

   - `mvencode`. This ado file changes missing values to numeric values and vice versa. We use the version 4.0.5, 01oct2004. It can be installed using the command `ssc install mvencode`.
   - `cic`. This package was written by Melly and Santangelo (2015). It implements several hypothesis testings based on the changes-in-changes model. It can be installed using the command `net install cic, from("https://sites.google.com/site/mellyblaise/")`.
   - `qrprocess`. The `cic` program requires the dependency `qrprocess` code. We use the version 1.0.1, 10.07.2014, also written by Blaise Melly. It can be installed using the command `net install qrprocess, from("https://sites.google.com/site/mellyblaise/")`.
   - `moremata`. The `cic` program also requires the `moremata` package, which includes various Mata functions. It was written by Jann (2005b) (revised in Feburary 19, 2022), and it can be installed using the command `ssc install moremata`.
   - `mmat2tex`. This package exports a Mata matrix into a LaTeX table format and saves it. It was written by Ditzen (2015) – we use the version 2.0 - 11.01.2017. It can be installed using the command `ssc install mmat2tex`.

- `crtrees`. This package implements a classification tree algorithm. It was written by Mora (2018) – we use the version 2.0 February2019. It can be installed using the command `ssc install crtrees`.
- `estout`. This package provides tools for making regression tables in Stata. The package currently contains the following commands: `esttab` (a command for publication-style regression tables); `estout` (a generic program for making a table from one or more sets of estimation results); `eststo` (a utility command to store estimation results for later tabulation); `estadd` (a utility command to add additional results to an existing estimation set, so that they can be tabulated); and `estpost` (a utility command to post results from various non-eclass commands as estimation results, so that they can be tabulated). They were written by Jann (2005a, 2007) and Jann and Scott Long (2010). To install the `estout` package, run the command `ssc install estout`.
- `coefplot`. This package plots regression coefficients. It was written by Jann (2014) – we use the version version 1.8.4 17dec2020. It can be installed using the command `ssc install coefplot`.

2. `Matlab R2021a`. The set of main results are obtained using Matlab. The toolboxes used are: the Statistics Toolbox, the Optimization Toolbox, and the Global Optimization Toolbox (GADS). The required toolboxes are listed at the top of the script of the main Matlab code. The optimal lists are calculated using the `intlinprog` program (part of the Optimization Toolbox), which implements the mixed-integer linear program, and the `ga` program (part of the Global Optimization Toolbox), which runs the genetic algorithm.
3. `ArcGIS 10.8.1`. All maps were generated using this version of ArcGIS.
4. `Excel 2016`. Several results are stored in excel files, in properly formatted tables that are then exported manually (and adjusted when necessary) to LaTeX using the add-inn `Excel2LaTeX`, which can be downloaded at `https://ctan.org/tex-archive/support/excel2latex/`.

The code was last run on a desktop with Windows 10 Pro version 21H1 (16Gb).

4 Directory structure

All of the data and code used is organized into the following directory structure:[2]

`/Code_Data`

`/Build`

[2]We follow Gentzkow and Shapiro (2014)'s guidelines in organizing our code and data.

`/Analysis_Stata`

`/Analysis_Matlab`

`/Analysis_GIS`

The `/Build` folder contains the following three subfolders:

`/Input`

`/Code`

`/Output`

The basic data files are in the `/Build/Input` subfolder, the Stata do-file codes are in the `/Build/Code` subfolder, and the Stata ado-file codes are in the `/Build/Code/libraries` subfolder. The do-files create the datasets that are then used in `/Analysis_Stata` and `/Analysis_Matlab`.

The `/Analysis_Stata` folder contains only the `/Code` and `/Output` subfolders. The do-file code in `/Analysis_Stata/Code` takes the dataset created in `/Build/Output` and generates descriptive figures and statistics, and regression tables. It uses the ado-files that are in the `/Analysis_Stata/Code/libraries` subfolder. (The latest versions of these ado-files can be installed by uncommenting the appropriate lines at the top of the do-file script.) The results are saved in `/Analysis_Stata/Output`.

The `/Analysis_Matlab` folder contains the same subfolders as in the `/Build` folder and one additional subfolder (`/Log`), to store the log files. The Matlab datasets created in the `/Build/Output` folder are manually copied over to be used as the inputs in the `/Analysis_Matlab/Input` subfolder. The datasets are then used to implement Athey and Imbens (2006)'s 'changes-in-changes' methodology to estimate treatment effects. They are also used to run our ex-post optimal list analysis, and to generate relevant figures and statistics.

After the main results are obtained in Matlab, they are exported manually into the ArcGIS data in the `/Analysis_GIS` folder in order to construct (manually) the maps that are presented in the main paper.

Throughout this document, it is assumed that the directory structure we describe is preserved. Modifying this structure may result in errors when running the code. (Of note, all `/Output` folders – and their subfolders – are empty by default, but will be populated upon execution of scripts.)

5 Dataset List

The following data files are included in the replication package. All analysis data required to run the code are provided. The analysis data are

1. `Build/Input/MunNeighbors_AML.xlsx`. Source: the National Highway Plan from the Brazilian Ministry of Transportation. The data list the municipalities' neighbors for each municipality in the Brazilian Amazon, calculated previously in ArcGIS using the `Polygon Neighbors` command and the shapefile with the municipalities' boundaries in the Brazilian Legal Amazon.

2. `Build/Input/basicdata.dta`. Source: those described in Section 2. This is the main dataset. It is a municipality-year panel data that combine information on land cover, Priority status, and several determinants of land cover that are used throughout the analysis, including rain, temperature, share of protected areas, FAO-GAEZ soil quality measures, etc. The construction of the variables are explained in Section A of the Online Appendix of the paper.

In addition, the `Analysis_GIS` folder contains geographic data that are useful to generate the maps:

1. `Analysis_GIS/AmazonPriorityList.gdb`. Source: the National Highway Plan. The data contain several shapefiles with the geographic boundaries of South America, Brazil, the Amazon Biome, the Brazilian municipalities, and the protected areas.

6 Description of Programs

We now describe the programs, including the order in which they must be implemented.

6.1 Building the Datasets

The `/Build/Code` folder contains the following codes:

1. `AmazonCIC.do`. It generates the datasets for data analysis in Stata and Matlab.

    - Input: `Build/Input/basicdata.dta`.
    - Output: `Build/Output/analysisdataset.dta` and `Build/Output/deforest_cic.dat`. (It also generates the intermediary data `Build/Output/list_for_adjacency_matrix.dta`.)

2. `AdjacencyMatrix_toMatlab.do`. It generates the adjacency matrix to be used in Matlab. (It calls the ado-file `mvencode`, described previously in Section 3, and saved in the `/Build/Code/libraries` subfolder.)

    - Input: `Build/Input/MunNeighbors_AML.xlsx` and `Build/Output/list_for_adjacency_matrix.dta`.
    - Output: `Build/Output/WMatrix_Neighbors_AML.dat`.

To generate these outputs, follow the steps below:

1. Navigate to the `/Build/Code` directory and open both the `AmazonCIC.do` and the `AdjacencyMatrix_toMatlab.do` files.

2. Set the global `BasePath` variable at the top of each of these files to point to the `/Build` folder. (Please note that paths in scripts are set in relative terms. Future users can replace these relative paths for full paths pointing to the local replication directory, if necessary.)

3. Run the `AmazonCIC.do` and the `AdjacencyMatrix_toMatlab.do` files in that order.

4. Copy the `deforest_cic.dat` and `WMatrix_Neighbors_AML.dat` files into the `/Analysis_MATLAB/Input` folder.

Remark. The `/Build/Code` folder also contains the excel file `/Build/Code/Figure1.xlsx`, which produces the paper's Figure 1 – based on aggregated time-series information on annual deforestation increment and cattle and soybean price indices.

6.2 Stata analysis code

The `/Analysis_Stata/Code` directory contains the code to generate various tables and figures contained in the paper:

1. `Tables_Figures.do` generates summary statistic and regression tables, and figures illustrating deforestation trends, selection criteria scatterplots, among others. It also runs the classification tree algorithm to infer the selection equation determining the Priority List. (It calls several ado-files described previously in Section 3, and that are saved in the `/Analysis_Stata/Code/libraries` subfolder.)

   - Input: `/Build/Output/analysisdataset.dta`
   - Output: (a) Tables 1, 4 (main paper), H1, H2, H4, H5, and H6 (online appendix). (b) Figures 3a and 3b (main paper), H1, H4, H5, H6 (online appendix).
     All tables are saved in LaTeX format, and all figures are saved in EPS format, with two exceptions: (i) the output for Table H1 needs to be copied and pasted manually from Stata to LaTeX; and (ii) the outcome of the classification tree algorithm must be included manually in Figure H1's LaTeX file.
   - Run time: approximately 40 seconds.

6.3 Matlab analysis code

In order to run the Matlab code to calculate and generate all the treatment effects and the optimal list results, simply open up the `deforest_Main.m` file contained in the `/Analysis_Matlab/Code` directory and run it. Before running, or when prompted upon running, make sure that the current directory in Matlab is set to this directory. Note that the `/Analysis_MATLAB/Output` folder contains subdirectories that correspond to various spillover and outlier-trimming specifications:

```
/Analysis_MATLAB/Output
   /G_bp250
   /G_bp300
   /G_bp350
   /GNT_65_bp300
   /GNT_70_bp300
   /GNT_75_bp300
```

These subdirectories must be preserved in order to prevent runtime errors. The terminology used to define the alternative specifications is the following: "Group_bpX." We consider the following groups: G, GNT_65, GNT_70, and GNT_75. Group G refers to the "no spillovers" case; groups GNT_Y, for Y= $65, 70, 75$, correspond to the spillovers cases, in which the spillover group depends on having at least one neighbor treated and having criteria variables that are above Y% of the selection threshold.[3] The "bpX" term refers to the outlier-trimming specification, where X= $250, 300, 350$, correspond to the cases in which we trimm the observations below and above the percentiles $[2.5, 97.5]$, $[3.0, 97.0]$, and $[3.5, 96.5]$, respectively. The main specifications presented in the paper are "G_bp300" and "GNT_70_bp300."[4]

The following list describes what each Matlab code file does:

1. `deforest_Main.m` is the main code file. It calculates and generates the results for each specification of spillover group and trimming percentile. It first constructs the actual and counterfactual distributions for each group-treatment combination.[5] It then calculates treatment effect estimates and bootstrap confidence intervals. Next, it determines ex-post optimal lists and related quantities. Finally it generates the output files.

2. `deforest_Iter.m`. Given a dataset (either the original dataset, or a dataset generated as part of a bootstrap iteration), this file generates actual and counterfactual distributions of

[3] Note that G referes to group, N refers to neighbor, and T refers to the threhold rule, in the "GNT" nomenclature.

[4] We also consider three alternative baseline years: 2006, 2007, and the average of the 2006 and 2007 baseline years. The latter corresponds to the main specification in the paper, and is denoted by the year "3000" in the code.

[5] 'Group' here refers to a municipality's observed treatment assignment or spillover designation (if untreated). 'Treatment' refers to the possibly counterfactual treatment applied to that municipality. If 'treatment'='group', then the treatment refers to the actual treatment. Otherwise, treatment refers to a counterfactual treatment.

the (partial-out) residuals, and computes the treatment effects. It makes use of the following code files:

- `supp.m`. This function takes an unordered vector Y, and converts it to an ordered support vector of unique values. (Originally written by Athey and Imbens, 2006).
- `prob.m`. Given a vector Y and a vector of support points YS, this function calculates the sample proportions at each of the support points. (Originally written by Athey and Imbens, 2006).
- `makeFCO.m`. This function creates the counterfactual distribution of residuals for a municipality in group *group* with treatment *treat*. This distribution ignores the region where the support of *group*'s pre-treatment residuals lie outside the support of *treat*'s pre-treatment residuals. (It modifies and extends a code originally written by Athey and Imbens, 2006). It makes use of the following code files:
  - `cdf.m`. Given a cdf with discrete support points, this function calculates the cumulative distribution function at a specified value y. (Originally written by Athey and Imbens, 2006).
  - `cdfinv.m`. Given a cdf with discrete support points, this function calculates the inverse cumulative distribution function at a specified probability y. (Originally written by Athey and Imbens, 2006).
- `makeFCO_BD.m`. This function creates the lower-bound (LB) and upper-bound (UB) counterfactual distributions, which have the missing probability masses shifted leftward and rightward respectively. (It modifies and extends a code originally written by Athey and Imbens, 2006).
- `makeDensity.m` converts a given distribution function into a probability mass function.
- `makeSharesForActual.m`. Using its group's actual distribution of residuals, this code determines the expected share of deforestation (out of the total existing forested area), the expected level of deforestation, and the expected level of carbon emissions for each municipality.
- `makeShares.m`. Using a counterfactual distribution of residuals, it calculates the expected share of deforestation, the expected level of deforestation, and expected level of carbon emissions for each municipality.
- `makeBlendedBaseYr.m` calculates the average of the two baseline years, 2006 and 2007.

3. `deforest_Boot.m` generates bootstrap-based confidence intervals for all treatment and spillover effects by randomly selecting 490 municipalities (with replacement). We set `numIter` = 500 bootstrap replications. It makes use of the following code files:

- `initBootFld.m` initializes the bootstrap fields in the specified table.
- `addBootIter.m` adds the results of the current bootstrap iteration to the specified table.
- `addBootSd.m` calculates the standard deviation across the bootstrap values and includes it in the specified table.
- `addIMC.m` determines the Imbens and Manski (2004) constant and includes it in the specified table. It calls the function below:
  - `getIMC.m`. It calculates the appropriate Imbens and Manski (2004) constant.
- `addCI.m` computes the bootstrap-based confidence intervals and includes them in the specified table.

4. `makeResultSet.m`. It creates the set up for the table with the results.
5. `makeEP2.m`. It creates a table (EP2) to index the combinations of outcomes (deforestation, D, or carbon emissions, CE) for the alternative scenarios and baseline years, for various blacklists – including the optimal lists.[6] It makes use of many of the code files below (though not necessarily exclusively).
6. `makeRnOut.m`. Return the maximum expected deforestation across untreated and spillover (if applicable) for each municipality in a specified outcome year.
7. `makeRnOut2.m`. Similar to the previous code, except that it returns the maximum expected deforestation for each municipality summed over the outcome years, 2009 and 2010.
8. `getSC_OBS.m`. It computes the expected social cost for the actual treatment in terms of a specified outcome (deforestation or carbon emission) and outcome year (2009-2010).
9. `getSC_OBS2.m` is similar to the above but for the sum of the 2009 and 2010 outcomes.
10. `makeRndList.m` creates and returns a matrix indicating multiple lists of randomly chosen municipalities, with each list satisfying a specified constraint (e.g. a constraint on total municipality area or on the number of municipalities monitored).
11. `makeSrtList.m` creates a list of municipalities chosen in either increasing or decreasing order of municipality area up until a specified constraint.
12. `makeT1_mod.m`. It modifies the value taken by a specific covariate in order to calculate alternative optimal lists in which that covariate is fixed at a given value.

[6] The outcomes are based on the sum of the 2009 and 2010 outcomes. The commented-out file, `makeEP1.m`, creates a similar table (EP1) with the results that are not aggregated over the years as in EP2.

13. `makeObsVsOpt.m` creates a matrix indicating the extent to which the Priority List overlaps with the ex-post optimal lists of municipalities.

14. `makeNewExPostOpt.m` returns the optimal list and the social cost associated with the optimal list, allowing for spillover effects. It calls the Matlab `ga` code, which implements the genetic algorihm to perform the optimization. It also makes use of the following function:

    - `getNewSC.m`. This file calculates the social cost associated with a specified list, allowing for spillover effects. I.e., it provides the objective function that is used in the genetic algorithm.

15. `makeNewExPostOpt2.m` is the same as the previous code, except that the genetic algorithm options are modified to increase run times at the expense of precision. This is only used to generate graphs of the social cost versus the level of the constraint for the spillovers case. (This is commented out in the code to save on total computational time, as this is highly computational costly.)

16. `getNewSCforRND.m`. It calculates the average social cost associated with a matrix of randomly generated lists, satisfying a constraint and allowing for spillovers.

17. `deforest_genOutputs.m` generates all Matlab outputs associated with the spillovers and outlier-trimming specifications. Results are exported to Excel files using various templates previously prepared and saved in `/Analysis_Matlab/Code`. This code calls the following functions:

    - `getEstimates_TE.m`. It returns the average treatment effects estimates or confidence intervals for a given baseline and outcome years.
    - `outputEP2_mod.m` prepares the table comparing outcomes of different lists when a specific covariate has its value fixed at some level (for sensitivity analysis).

To summarize, we have:

- Input: The files `deforest_cic.dat` and `WMatrix_Neighbors_AML.dat`, generated in `/Build/Output` and that must be saved manually in the `/Analysis_Matlab/Input/` folder.

- Output: (i) Tables 2, 3, 5, 6 (main paper), H3, H7, H8, H9, H10, H11, and H12 (online appendix). (ii) Figures 5 (main paper), H2, and H3 (online appendix). (iii) Excel files for ArcGIS: `/Analysis_MATLAB/Output/G_bp300/AllLists.xlsx` and `/Analysis_MATLAB/Output/GNT_70_bp300/AllLists.xlsx` files.

  All figures are saved in EPS format and all tables are saved in Excel. The tables presented in the main paper and online appendix are exported manually from Excel to LaTeX, using the add-inn `Excel2LaTeX`, and adjusted when necessary. Below, we relate the tables presented in the paper and online appendix with the Excel files saved in `/Analysis_MATLAB/Output/`:

- Table 2 corresponds to the table in `G_bp300/TreatEstimates.xlsx`, sheet "*OutputLatex_Main.*"
- Table 3 corresponds to the table in `G_70_bp300/TreatEstimates.xlsx`, sheet "*OutputLatex_Main.*"
- Table 5 combines the tables in `G_bp300/obsVsOpt_3000.xlsx` and in `G_70_bp300/obsVsOpt_3000.xlsx`, sheets "*DeforestLatex.*"
- Table 6 combines the tables in `G_bp300/SocialCosts.xlsx` and in `G_70_bp300/SocialCosts.xlsx`, sheets "*OutputLatex.*"
- Table H3 corresponds to the table in file `G_bp300/Supp_table.xlsx`, sheet "*LatexSuppTable.*"
- Table H7 corresponds to the table in `G_bp300/TreatEstimates.xlsx`, sheet "*OutputLatex_Appendix.*"
- Table H8 corresponds to the table in `G_bp300/SocialCosts.xlsx`, sheet "*OutputLatex_Baselines.*"
- Table H9 combines the tables in `G_bp250/TreatEstimates.xlsx` and in `G_bp350/TreatEstimates.xlsx`, sheets "*OutputLatex_Main.*"
- Table H10 combines the tables in `G_bp250/SocialCosts.xlsx` and in `G_bp350/SocialCosts.xlsx`, sheets "*OutputLatex.*"
- Table H11 combines the tables in `GNT_65_bp300/TreatEstimates.xlsx` and in `GNT_75_bp300/TreatEstimates.xlsx`, sheets "*OutputLatex_Main.*"
- Table H12 combines the tables in `GNT_65_bp300/SocialCosts.xlsx` and in `GNT_75_bp300/SocialCosts.xlsx`, sheets "*OutputLatex.*"

Next, we relate the figures presented in the paper and online appendix with the figures saved in `/Analysis_MATLAB/Output/`:

- Figure 5 corresponds to `G_bp300/ExPostCEMUNAREACONSTRAINT_3000.jpg`.
- Figure H2a corresponds to `G_bp300/CDFs_AI_Actual_Untreated_2009_Baseline_2007.jpg`.
- Figure H2b corresponds to `G_bp300/CDFs_AI_Actual_Untreated_2010_Baseline_2007.jpg`.
- Figure H2c corresponds to `G_bp300/CDFs_AI_CF_Treated_2009_Baseline_2007.jpg`.
- Figure H2d corresponds to `G_bp300/CDFs_AI_CF_Treated_2010_Baseline_2007.jpg`.

– Figure H3a corresponds to `G_bp300/CDFs_AI_Actual_Treated_2009_Baseline_2007.jpg`.

– Figure H3b corresponds to `G_bp300/CDFs_AI_Actual_Treated_2010_Baseline_2007.jpg`.

– Figure H3c corresponds to `G_bp300/CDFs_AI_CF_Untreated_2009_Baseline_2007.jpg`.

– Figure H3d corresponds to `G_bp300/CDFs_AI_CF_Untreated_2010_Baseline_2007.jpg`.

- Run time: approximately 265 minutes.[7]

6.4 ArcGIS analysis code

To construct the maps presented in the main paper using ArcGIS 10.8.1, one must first create an ArcMap file (i.e., a `.mxd` file), using the shapefiles in `Analysis_GIS/AmazonPriorityList.gdb`, so that one can display and explore the GIS datasets.[8] Then, after running the Matlab code and obtaining the optimal lists, one can export them to the ArcMap file using the `Add Data` command. The files that must be exported are: `/Analysis_MATLAB/Output/G_bp300/AllLists.xlsx`, for the "no-spillovers" case; and `/Analysis_MATLAB/Output/GNT_70_bp300/AllLists.xlsx`, for the "spillovers" case.

After adding the data to the ArcMap file, join each of the Excel files into the `mun_bio_aml` shapefile using the command `Join`. The field in the `mun_bio_aml` layer that the join is based on is "COD_IBGE," and the field in the (imported Excel) table to base the join on is "Mun_ID." The figures are then prepared (i.e., setting the colors, the legend, title, etc.) and are saved in the `.jpg` format, using the `Export Map` command. To present the geographic distribution of the lists, note that, in the Excel files, the column for the Priority List status is denoted by "Observed;" the column for the optimal list status that minimizes total deforestation subject to the total area constraint is denoted by "MArea_D_OPT2;" and the column for the optimal list status that minimizes total deforestation subject to the number of municipalities constraint is denoted by "NumMun_D_OPT2." We use the same column names for the lists that ignore spillover effects (based on the `/Analysis_MATLAB/Output/G_bp300/AllLists.xlsx` file) and for those that incorporate spillover effects (based on the `/Analysis_MATLAB/Output/GNT_70_bp300/AllLists.xlsx` file).

To summarize, we have

- Input: (i) The GIS file `/Analysis_GIS/AmazonPriorityList.gdb`, and (ii) the Excel files generated in Matlab, `/Analysis_MATLAB/Output/G_bp300/AllLists.xlsx` and `/Analysis_MATLAB/Output/GNT_70_bp300/AllLists.xlsx`.

[7] It takes approximately 13 minutes to generate results for the no-spillovers case, and approximately 70-80 minutes, for each of the three spillover cases (GNT_65, GNT_70, and GNT_75).

[8] If one prefers to use ArcGIS Pro, the ArcMap document can be converted by following the instructions presented here: `https://pro.arcgis.com/en/pro-app/2.8/get-started/migrate-content-to-arcgis-pro.htm`.

- Output: Figures 4a, 4b, 4c, 4d, 6a, and 6b (main paper).

References

Athey, S. and G. W. Imbens (2006). Identification and inference in nonlinear difference-in-differences models. *Econometrica 74(2)*, 431–97.

Baccini, A., S. J. Goetz, W. S. Walker, N. T. Laporte, M. Sun, D. Sulla-Menashe, J. Hackler, P. S. A. Beck, R. Dubayah, M. A. Friedl, S. Samanta, and R. A. Houghton (2012). Estimated carbon dioxide emissions from tropical deforestation improved by carbon-density maps. *Nature Climate Change 2*, 182–185.

BCB (2020). Sistema de informações de crédito. Brazilian Central Bank. `https://www.bcb.gov.br/estabilidadefinanceira/scr` (the raw data used was obtained upon request and received on July 2017).

CNUC (2016). Unidades de conservação. Technical report, Cadastro Nacional de Unidade de Conservação, Ministério do Meio Ambiente. `http://mapas.mma.gov.br/ms_tmp/ucstodas.shp;http://mapas.mma.gov.br/ms_tmp/ucstodas.shx;http://mapas.mma.gov.br/ms_tmp/ucstodas.dbf` (accessed March 11, 2016).

Ditzen, J. (2015). MMAT2TEX: Stata module to output Mata matrix as LaTeX table. Technical report.

FAO (2021). Global Agro Ecological Zones version 4 (gaez v4). Food and Agriculture Organization's Global Agro-Ecological Zones. `https://gaez.fao.org/pages/data-access-download` (accessed January 8, 2021).

FUNAI (2016). Terras indígenas: shapefile. Technical report, Fundação Nacional do Índio, Ministério da Justica. `http://www.funai.gov.br/index.php/shape` (accessed June 22, 2016).

Gentzkow, M. and J. M. Shapiro (2014). Code and data for the social sciences: A practitioner‘s guide. *Unpublished manuscript*.

IBAMA (2016). Autuações ambientais: Todos, 2000-2015. Technical report, Instituto Brasileiro do Meio Ambiente e dos Recursos Naturais Renováveis, Ministério do Meio Ambiente. `https://servicos.ibama.gov.br/ctf/publico/areasembargadas/ConsultaPublicaAreasEmbargadas.php` (link to updated version available online, but the raw data used was obtained upon request via e-SIC and received by mail in physical media on May 2016).

IBGE (2017a). Pesquisa da pecuária municipal: Tabela 3939. Technical report, Instituto Brasileiro de Geografia e Estatística, Ministério da Economia. `https://sidra.ibge.gov.br/pesquisa/ppm/quadros/brasil` (accessed June 6, 2017).

IBGE (2017b). Produção agrícola municipal: Tabela 1612. Technical report, Instituto Brasileiro de Geografia e Estatística, Ministério da Economia. `https://sidra.ibge.gov.br/pesquisa/pam/tabelas` (accessed June 6, 2017).

IBGE (2017c). Produto interno bruto dos municípios 2017: Tabela 5938, 2002-2017. Technical report, Instituto Brasileiro de Geografia e Estatística, Ministério da Economia. `https://sidra.ibge.gov.br/pesquisa/pib-munic/tabelas` (acessed June 6, 2017).

Imbens, G. W. and C. F. Manski (2004). Confidence intervals for partially identified parameters. *Econometrica 72*, 1845–1857.

INPE (2017a). Monitoramento da floresta Amazônica Brasileira por satélite – Projeto PRODES. Technical report, Coordenação-Geral de Observação da Terra, Instituto Nacional de Pesquisas Espaciais, Ministério da Ciência, Tecnologia e Inovação. Available at `http://www.obt.inpe.br/prodes/` (accessed April 29, 2016).

INPE (2017b). Sistema de detecção de desmatamento em tempo real (DETER): Alertas. Technical report, Coordenação-Geral de Observação da Terra, Instituto Nacional de Pesquisas Espaciais, Ministério da Ciência, Tecnologia e Inovação. `http://www1.dpi.inpe.br/obt/deter/dados/` (accessed multiple times through 2016-2018).

Jann, B. (2005a). Making regression tables from stored estimates. *The Stata Journal 5*(3), 288–308.

Jann, B. (2005b). MOREMATA: Stata module (Mata) to provide various functions. Technical report, Statistical Software Components S455001, Boston College Department of Economics.

Jann, B. (2007). Making regression tables simplified. *The Stata Journal 7*(2), 227–244.

Jann, B. (2014). Plotting regression coefficients and other estimates. *The Stata Journal 14*(4), 708–737.

Jann, B. and J. Scott Long (2010). Tabulating spost results using estout and esttab. *The Stata Journal 10*(1), 46–60.

Matsuura, K. and C. Willmott (2018a). Terrestrial air temperature: 1900-2017 gridded monthly time series (v 5.01). Technical report, Department of Geography, University of Delaware, Newark, DE. `http://climate.geog.udel.edu/~climate/html_pages/download.html` (accessed July 20, 2018).

Matsuura, K. and C. Willmott (2018b). Terrestrial precipitation: 1900-2017 gridded monthly time series (version 5.01). Technical report, Department of Geography, University of Delaware, Newark, DE. `http://climate.geog.udel.edu/~climate/html_pages/download.html` (accessed July 20, 2018).

Melly, B. and G. Santangelo (2015). The changes-in-changes model with covariates. Working Paper, University of Bern.

MMA (2017). Lista de municípios prioritários da Amazônia. Technical report, Ministério do Meio Ambiente. `http://combateaodesmatamento.mma.gov.br/images/conteudo/lista_municipios_prioritarios_AML_2017.pdf` (accessed September 20, 2017).

Mora, R. (2018). CRTREES: Stata module to compute Classification and Regression Trees algorithms. Technical report.

PNLT (2011). Projeto de reavaliação de estimativas e metas do PNLT, relatório final. Technical report, Ministério dos Transportes. https://www.gov.br/infraestrutura/pt-br/assuntos/transporte-terrestre_antigo/pnlt-relatorio-executivo-2011 (accessed June 15, 2011).

SEAB-PR (2019). Preço médio - recebido pelo agricultor: boi gordo, arroz (em casca), cana-de-açúcar, milho, mandioca, 1990-2019. Technical report, Secretaria da Agricultura e do Abastecimento do Estado do Paraná, Departamento de Economia Rural [publisher], Instituto de Pesquisa Economica Aplicada, Ministério da Economia [distributor]. http://www.ipeadata.gov.br (accessed April 16, 2019).

Souza-Rodrigues, E. (2019). Deforestation in the Amazon: A unified framework for estimation and policy analysis. *Review of Economic Studies 86*(6), 2713–2744.