README

Under the (Neighbor)Hood: Understanding Interactions Among Zoning Regulations

Amrita Kulka, Aradhya Sood, and Nicholas Chiumenti

Contents

Introduction **4**
- Authors . . . 4
- Paper abstract . . . 4
- Acknowledgements . . . 4
- Repository overview . . . 4
  - Directory structure . . . 4
  - Software requirements . . . 5

Start Here **5**
- Overview . . . 5
- Navigating this Guide . . . 6

Walkthrough **7**
- 1. Boundary Selection . . . 7
  - Step 1.1: Run `new_mf_definitions.do` . . . 7
  - Step 1.2: Run `create_admissible_boundaries.py` . . . 7
- 2. Data Setup . . . 7
  - Step 2.1: Run the master data setup file. . . . 7
  - Step 2.2: Compile the warren group data . . . 8
  - Step 2.3: Run `zone_assignments.ipynb` . . . 8
  - Step 2.4: Run `closest_boundary_matches.ipynb` . . . 9
  - Step 2.5: Run `20_boundary_matches.do` . . . 9
  - Step 2.6: Calculate density measures . . . 9
  - Step 2.7: Compile the CoStar data . . . 9
  - Step 2.8: NHPD data, clean, compile, and match to Warren Group properties . . . 10
  - Step 2.9: repeat Step 8 but with the Chapter 40B files . . . 11
  - Step 2.10: Compile the final dataset . . . 11
  - Step 2.11, run `80_amentiy_datasets.do` . . . 12
  - Step 2.12, create the striaght line boundary files . . . 12
  - Step 2.13: Create the no roads files . . . 12
  - Step 2.14: Calculate trasit distances . . . 13
  - Step 2.15: Assign soil quality data . . . 13
  - Step 2.16: Create walkability data . . . 13
  - Step 2.17: Create `final_dataset_town_comparisons.dta` . . . 13
- 3. Analysis . . . 14
  - Step 3.1: Run the master analysis file . . . 14
  - Step 3.2: Run `analysis_within_town_setup.do` . . . 14
  - Step 3.3 (optional): Run the remaining analysis setup files . . . 14
  - Step 3.n: Run the remaining analysis files . . . 14
- File-to-Figure Map . . . 15

Tables-to-Files . . . 15
Figures-to-files . . . 15

Data Setup Files Guide **16**
Overview . . . 16
Order of files . . . 16
Main Setup Files . . . 17
00_data_setup_master_file.do . . . 17
10_warren_data_compile.do . . . 17
20_boundary_matches.do . . . 18
30_density_measures.do . . . 19
40_costar.do . . . 19
50_nhpd.do . . . 20
60_ch40b.do . . . 20
70_final_dataset.do . . . 20
80_amenity_datasets.do . . . 21
Python Setup Files . . . 22
zone_assignments.ipynb . . . 22
closest_boundary_matches.ipynb . . . 22
soil_quality_matching.ipynb . . . 23
all_stations.ipynb . . . 23
dist_prop_to_station.ipynb . . . 23
dist_to_south_station.ipynb . . . 24
station_boundary_dist.ipynb . . . 24
walking_distances_orsm.ipynb . . . 25
BatchAddressMatch_final.ipynb . . . 25
Miscellaneous Setup Files . . . 25
effective_dist_export.do . . . 25
shp2dta.do . . . 26
town_list_export.do . . . 26
warren_geocode_fixes.do . . . 26

Analysis Files Guide **27**
Overview . . . 27
Code Structure . . . 27
File Descriptions . . . 28
analysis_master_file.do . . . 28
analysis_within_town_setup.do . . . 28
analysis_noroads_setup.do . . . 29
analysis_town_comparison_setup.do . . . 29
amenities_mtlines.do . . . 29
amenities_muni_boundary.do . . . 30
bindingness.do . . . 30
chars_mtlines.do . . . 30
counterfactual_01_spatial_heterogeneity.do . . . 31
counterfactual_02_train_station_means.do . . . 31
counterfactual_03_means.do . . . 31
counterfactual_04_calculations_combined.do . . . 32
existence.do . . . 32
external_effects.do . . . 33
histogram.do . . . 33
main_mtlines.do . . . 33
main_noroads.do . . . 34
predicted_prices_mtlines.do . . . 34

residuals.do . . . 34
robustness_mtlines.do . . . 35
straight_v_walking_dist.do . . . 35
table_1_replication.do . . . 36
within_town_mtlines.do . . . 36
within_town_mtlines_robustse.do . . . 36
assessed_vs_sales.do . . . 37
figure_A6.do . . . 37
included_excluded_towns.do . . . 37
regulations_map.do . . . 38
straight_line_boundary_map.do . . . 38
warren_group_property_map.do . . . 38

Data Sources Guide **39**
Non-public data sources . . . 39
Warren Group . . . 39
Costar . . . 39
Public data sources . . . 39
American Community Survey (ACS) . . . 40
Chapter 40B . . . 40
City centers (centroids) . . . 40
Consumer Price Index . . . 41
Green Space . . . 41
Major and Minor Roadways . . . 41
Municipality Boundaries . . . 42
Metropolitan Area Planning Council (MAPC) Zoning Atlas . . . 42
National Housing Preservation Database (NHPD) . . . 42
School Attendance Boundaries . . . 43
School Districts . . . 43
Soil Quality Data . . . 43
Train Stations & Transit Distances . . . 44
Walk Score (Walkability) . . . 44
Water Features (hydrography) . . . 44

Introduction

This repository contains the code, data and output underlying the paper *Under the (Neighbor)Hood: Understanding Interactions Among Zoning Regulations*, published in the *Review of Economics and Statistics*, by Amrita Kulka, Aradhya Sood and Nicholas Chiumenti.

Paper SSRN Link

GitHub Repo Link

Authors

Amrita Kulka
amrita.kulka@warwick.ac.uk
https://sites.google.com/site/kulkaamrita/home

Aradhya Sood
aradhya.sood@rotman.utoronto.ca
https://sites.google.com/view/aradhyasood

Nicholas Chiumenti
nick.chiumenti@gmail.com

Paper abstract

We study how various zoning regulations combine to affect housing supply, prices, and rents of single- and multifamily homes using novel lot-level zoning data from Greater Boston and a cross-sectional boundary discontinuity design at regulation boundaries. Looser density restrictions, alone or with other less restrictive regula- tions, are most effective in increasing supply and reducing per-housing-unit rents and prices. We theoretically and empirically show that restrictive zoning regula- tions shift housing stock towards larger units, increasing prices per housing unit. Counterfactual simulations imply that a recent Massachusetts law increasing build- ing density near transit can reduce long-run rents and prices, particularly in sub- urbs.

Acknowledgements

For their helpful and insightful comments, we thank Treb Allen, Nate Baum-Snow, Kirill Borusyak, Leah Brooks, Ingrid Gould Ellen, Fernando Ferreira, Lucie Gadenne, Jeffrey Lin, Jenny Schuetz, Will Strange, Jeff Thompson, Matt Turner, Paul Willen, and Jeff Zabel as well as seminar participants at various institutions. We also thank Can Ay, Levi Berger, Hope Bodenschatz, Mike Corbett, and Eli Inkelas for providing invaluable research and coding assistance.

Repository overview

It is recommended you familiarize yourself with the documentation provided in this repository before attempting to run any code or replicate any results. Refer to the documentation folder for details, or click the link below to follow the walkthrough guide.

Start Here!

Directory structure

`/analysis` – Contains any *output* from programs in the `./code/analysis_files` directory.

`/code` – Contains all coding files used for data setup and analysis.

`/data` – Contains raw input data, as welll as intermediate data files, and final data files used in analysis that are created from programs in the `./code/data_setup_files` directory.

`/docs` – Contains repo associated .md files, guides, walkthroughs, codebooks, etc.

Software requirements

This project is primarily coded in Stata and Python. Jupyter Notebook was used to code up python files. It is recommended to have access to ArcPro or an equivalent geospatial analysis tool to view and work with with the source .shp files.

Suggested program versions are given below:

- Stata v16.0 or later.
- Python 3.9 or later
- Jupyter Notebooks v6.0 or later (recommended)
- ArcPro v3.6 (or equivalent)

Start Here

The purpose of this document is to aid future users who wish to replicate the analysis found in *Under the (Neighbor)Hood: Understanding Interactions Among Zoning Regulations*.

Overview

We ***strongly*** advise users to familiarize themselves with the documentation provided in this repository before attempting to run any code or replicate any results. Any and all relevant documentation is located in `/docs`.

The repository is structured as follows:

`/code` – Contains all coding files used for data setup and analysis.

`/code/analysis_files` – Contains programs associated with final output analysis.

`/code/data_setup_files` Contains programs associated with data cleaning and initial setup.

`/data` – Contains raw input data, intermediate data files, final data files.

`/docs` – Contains repo associated .md files, guides, walkthroughs, codebooks, etc.

> **!** Important
>
> **A note about data availability**
> To the extent possible we have provided raw input, intermediate, and final versions of data files in this repository. However, we are limited in what we are able to share in this repository in two (2) major ways.
> 1) **Data license limitations** - We are unable to data files that were made available through the Federal Reserve Bank of Boston. These include any source files of the Warren Group's property tax assessment records data, and the CoStar rental property history data. In both cases, the Federal Reserve Bank of Boston and/or the Federal Reserve System is the holder of these data license agreements. This limitation extends to any derived output, except for aggregated output, included matching and crosswalk files.
> 2) **Data file size limits** - GitHub limits file size uploads to to 100MB. We are not able to upload and share large files that exceed this limit but they are available upon request. When possible, files have been compressed into .zip files to facilitate upload. All files we are able to share, regardless of size, were uploaded the *Review of Economics and Statistics Dataverse*.

Placeholder files are located throughout this repository to illustrate where original data files should be located if they were unable to be shared via this repository. These placeholder files follow the format of `<original_filename>.txt` and include text information within them.

For example, the original file `MA_assessor_annual_expanded.dta` cannot be shared because it is under a license agreement. It would be located at the following file path:

`/data/warren/originals/MA_assessor_annual_expanded.dta`

In its place is the following:

`/data/warren/originals/MA_assessor_annual_expanded.dta.txt`

If opened in a text editor, it displays the following:

```
Source: MA_assessor_annual_expanded.dta

This is a placeholder file illustrating where the source file is located within
the directory structure.

Cannot be shared due to data license agreement.
```

Directory tree files called `_tree.txt` have been added throughout the `/data` directory and list all files and subfolder that should be located within them, even if they are not available through this repository.

For example if `/data/walkability/_tree.txt` is opened in a text editor it displays the following:

```
Folder PATH listing for volume OS
Volume serial number is F054-B23E
C:.
    .gitignore
    National Walkability Index_Methodology and User Guide_June2021.pdf
    Natl_WI.gdb.zip
    Natl_WI.gdb.zip.txt
    tree.txt
    warren_group_walkability.dta

No subfolders exist
```

Navigating this Guide

Proceed to one of the following for more information:

The Walkthrough guide details the steps that should be taken to go from raw data to final results.

The Data Setup Guide details the files involved in the data setup process.

The Analysis Guide details the files involved in the analysis. Users can refer here to find the process that creates a particular figure or table in the paper.

The Data Sources Guide provides information and links to the various data sources used in the paper.

Walkthrough

This walkthrough provides the steps involved with reproducing the data used in *Under the (Neighbor)Hood: Understanding Interactions Among Zoning Regulations*. It is assumed the users has access to the input data files, including those not shared in this repository.

> **Note**
>
> **Program Requirements** - Stata 16, or later version - Python 3.9, or later version - ArcPro v3.6, or later version, or comparable GIS software
> It is recommended the user have Jupyter Notebook installed for viewing and running the `.ipynb` files. Alternatively most modern IDEs have `.ipynb` file compatibility.

1. Boundary Selection

A core component of this paper are the zoning boundaries that get matched to every residential property in Greater Boston. The steps below detail how the initial set of boundaries are created.

Step 1.1: Run `new_mf_definitions.do`

In Stata, run `new_mf_definitions.do`. This file can be found under `code/data_setup/miscellaneous_setup_files`. Either the global `$DATAPATH` variable to point to the local path of `/data`.

Step 1.2: Run `create_admissible_boundaries.py`

Note that this file is coded as a Python `.py` file, but can be run in Jupyter Notebooks. Run all parts of `create_admissible_boundaries.py`.

This file is coded to run out of the directory `/data/boundary_selection`, so the `.py` file first copies over multiple input `.zip` files that contain the polygon data. However you can comment-out this code and simply direct the paths to their source files in the corresponding `/data` folder.

The result of this step is the creation of `amd3.shp`. This is the final admissible boundaries file that contains the individual zoning boundaries where Warren Group properties will be assigned to one of two sides (left or right) for comparison.

2. Data Setup

The steps below are in a kind of chronological order. That is, **steps are in the order that the code was written** for this project. The result is that some steps may have duplicative components, or be superseded by later steps. Ultimately we decided not to consolidate code and remove vestigial parts of the repository because we felt that would conflict with the goal of transparency.

Step 2.1: Run the master data setup file.

In Stata, run `00_data_setup_master_file.do`. This file sets all necessary global path variables that will be used throughout the setup process. Any changes to the global path variables to match a user's local directory structure should be made here. The most important global variable to set correctly is `$DATAPATH`, as this will point to (almost) all subsequent data files and their assumed sub-directories.

Note that this file does set the current working directory to `$DOPATH`, which is not strictly necessary.

Step 2.2: Compile the warren group data

In Stata, run `10_warren_data_compile.do`. This creates several base Warren Group property-level data files. It requires a access to `MA_assessor_annual_expanded.dta`, which is the raw Warren Group data extracted for the purposes of this paper.

This step calls three (3) sub-scripts, which handle geocoding tasks (`11_geocoding.do`), identify residential properties and remove non-residential property records (`12_res_types.do`), and clean and aggregate condominium records for use later on (`13_condo_collapse.do`). Note that in the end condominium properties were excluded form the analysis.

After completing this step the following files should have been created:

- `warren_MA_all_annual.dta` - All *residential* property records in Massachusetts, by year.
- `warren_MA_all_unique.dta` - A unique set of all *residential* property records in Massachusetts (across years).
- `warren_MAPC_all_annual.dta` - All *residential* property records in MAPC region, by year.1
- `warren_MAPC_all_unique.dta` - A unique set of all *residential* property records in MAPC region (across years).

Step 2.2.a. (optional)

If necessary, within `11_geocoding.do`, you may wish to uncomment and run `warren_geocode_fixes.do`. This sub-process handles output from `BatchAddressMatch_final.ipynb`, which in turn uploads structured addresses to the Census Geocoder API in order to identify and correct lat/lon coordinates for a handful of records. This step only needs to be done once and is very time consuming as you are rate limited in the use of the API.

1. stop the process at line 137 of `11_gecoding.do`
2. uncomment line 138,
3. run `./data/warren/geocode_fixes/warren_geocode_fixes.do` up to line 102.
4. navigate to `./data_setup/python_programs/census_geocoder_api`
5. edit `BatchAddressMatch_final.ipynb` to point to the `.txt` files created by step 2.2.a.3.
6. ensure the output is saved to `$DATAPATH/warren/geocode_fixes`
7. run `BatchAddressMatch_final.ipynb`, ensure `geocoder_export_`DateStamp'.csv' is created
8. run `warren_geocode_fixes.do` in full, ensure `address_corrections_output.dta` is created
9. comment-out line 138 in `11_gecoding.do`
10. re-start `11_geocoding.do` from the top, or, re-run `10_warren_data_compile.do` in full.

1 MAPC is the Metropolitan Area Planning Council, which created the Zoning Atlas data used in this report. The towns in MAPC's region are the basis for our definition of Greater Boston.

Step 2.3: Run `zone_assignments.ipynb`

In Jupyter Notebook, run all parts of `zone_assignments.ipynb`. This will create a dataset which matches Warren Group properties to the correct zoning area, zone-use type, and school district. It will allow for the correct assignment of properties to zoning boundaries within the same city/town, school district, zoning area, and zone-use type area.

The result of this step will create the `zone_assignments_export.csv`. Ensure that `zone_assignments_export.csv` has been generated ***in the same directory as*** ***`zone_assignments.ipynb`***. This file will contain all Warren Group property IDs, and the variables:

- `zo_usety` - the zone use type code
- `l_r_fid` - the unique zoning area ID
- `ncessch` - the unique school attendance area

Note that `zone_assignments.ipynb` requires correct paths to the following:

- warren_MAPC_all_unique.dta
- sabs_unique_latlong.shp - a version of SABS_1516_Primary.shp with unique, non-overlapping, school district boundaries.
- roads_mapc_union_sd_dissolved.shp - a version of zoning_atlas_latlong.shp with unique, non-overlapping, zoning areas.
- zoning_atlas_latlong.shp - contains the zone-use type areas.

Step 2.4: Run `closest_boundary_matches.ipynb`

In Jupyter Notebook, run all parts of `closest_boundary_matches.ipynb`. This will match all of the Warren Group properties to the five (5) closest zoning boundary pairs identified in `adm3_crs4269.shp`. It takes the output from Step 2.3 and iterates over all possible matches, holding school attendance area, zoning area, zone-use type, and municipality constant.

The result of this step will create `closest_boundary_matches.csv`. Ensure that `closest_boundary_matches.csv` has been generated ***in the same directory as*** ***`closest_boundary_matches.ipynb`***. It should include the following variables:

- `unique_id` - the unique identifier of the boundary pair
- `LEFT_FID` - the unique identifier of a zoning area for the 'left' side of the boundary pair
- `RIGHT_FID` - the unique identifier of a zoning area for the 'right' side of the boundary pair
- `boundary_side` - text entry identifying if the property matched with LEFT_FID, RIGHT_FID, or both
- `nearest_point_dist` - the distance of the nearest point on the boundary to the Warren property
- `nearest_point_lat` - the nearest point latitude coordinates
- `nearest_point_lon`- the nearest point longitude coordinates
- `match_num` - the number match indicator (1-5)

Step 2.5: Run `20_boundary_matches.do`

In Stata, run `20_boundary_matches.do`. This code file will identify the single best match of a warren group property to a zoning boundary, out of the five closest, defined as the closest boundary with comparable regulations on both sides of the boundary (left and right side). It will also match on the associated zoning regulations and assign them to the home zoning area of the property (`home_`) and the zoning area identified for comparison (`nn_`).

The result of this step will create `closest_boundary_matches_with_regs.dta`.

Step 2.6: Calculate density measures

Note that this files handles setup for analysis that is no longer relevant to the paper. However, future files reference output from these steps and so may throw runtime errors if not present.

In Stata, run `30_density_measures.do`. This file calculates the share of properties in a .1 mile radius around every Warren Group property that is a 2-3 unit (gentle density) and 4+ unit (high density) building.

The result will be the file `warren_density_measures.dta`, which stores the density data and is merged on later.

Step 2.7: Compile the CoStar data

In Stata, run `40_costar.do` to compile all of the raw data scraped from the costar website. `40_costar.do` also calls two subscripts:

- `41_costar_warren_xwalk.do` - creates a crosswalk that is used to identify matching warren group properties and their corresponding costar record, if present.
- `42_costar_rent_history.do` - compiles the historic rent data scraped from Costar and uses the crosswalk made in `41_costar_warren_xwalk.do` to identify corresponding warren group properties.

> **Note**
>
> Refer to the `_tree_data_dir.txt file` in order to view the structure and contents of the `/data/costar` directory used to complete this step. `costar_props.ipynb` is stored within this directory and is responsible for creating `costar_property_list.xlsx`. This script is run in recursively checks the `./property rent history` directory, loads the scraped data, and adds it to a dataframe that is then exports it to excel. We were unable to include `costar_props.ipynb` in this repository.

The result of this step are three .dta files:

- `costar_mf_all.dta` stores all of the costar information with matched warren group property ids.
- `costar_mf_destring.dta` is the same as the above, but with all variables converted to float.
- `costar_rent_hist.dta` stores the rental history data that was available in costar.

Step 2.8: NHPD data, clean, compile, and match to Warren Group properties

This is a multi step process that handles the raw NHPD data, cleans it, assigns it to zoning boundaries, and matches it to warren group properties. All sub-steps are required if you wish to reproduce this component of the data. Note however, it is not used in the final analysis. However, future files reference output from these steps and so may throw runtime errors if not present.

Step 2.8.a.: Run `50_nhpd.do` up until line 459.

In Stata, run `50_nhpd.do`. This will create `nhpd_mapc.dta`, which is a list of all subsidized housing properties in the Metropolitan Area Planning Council region within Massachusetts.

Ensure that all paths correctly point to the the following files: - `All Properties.xlsx`

Step 2.8.b.: Run `zone_assignments_nhpd.ipynb`

In Jupyter Notebook, run `zone_assignments.ipynb`. This is an identical process to Step 3, above, but instead this file is coded to handle `nhpd_mapc.dta`. It will result in the file `zone_assignments_nhpd_export.csv`. All other input files are the same as Step 2.3.

Ensure that all paths currectly point to the the following file(s):

- `nhpd_mapc.dta`

The result of this step are the following file(s):

- `zone_assignments_nhpd_export.csv`

Step 2.8.c.: Run `closest_boundary_matches_nhpd.ipynb`

In a Python IDE, run `closest_boundary_matches_nhpd.ipynb`. This is an identical process to Step 4, above, but instead this file is coded to handle `zone_assignments_nhpd_export.csv`. All other input files are the same as Step 4.

Ensure that all paths correctly point to the the following file(s):

- `zone_assignments_nhpd_export.csv`

The result of this step are the following file(s):

- `closest_boundary_matches_nhpd.csv`

Step 2.8.d.: In `50_nhpd.do`, run line 467

This will call `51_nhpd_boundary_matches.do`, which is similar process to Step 5, above. It will identify the closest zoning boundary and assign regulations using the same criteria at Step 5.

Ensure that all paths correctly point to the the following file(s):

- `closest_boundary_matches_nhpd.csv`
- `regulation_types.dta`

The result of this step are the following file(s):

- `closest_boundary_matches_nhpd_with_regs.dta`

Step 2.8.e.: In `50_nhpd.do`, run line 469

This will call `52_nhpd_warren_xwalk.do`, which will identify the corresponding warren group property and assign its ID to the NHPD record. It follows a similar process used in Step 7, above, (specifically in `41_costar_warren_xwalk.do`), but also uses the assigned zoning boundary as a criteria for matching.

Ensure that all paths correctly point to the the following file(s):

- `nhpd_mapc.dta`
- `warren_MA_all_unique.dta`
- `closest_boundary_matches_nhpd_with_regs.dta`

The result of this step are the following file(s):

- `nhpd_warren_xwalk.dta`

Step 2.9: repeat Step 8 but with the Chapter 40B files

The process for handling the Chapter 40B data is essentially the same as the NHPD data in Step 2.8. You should follow along in the same way but with the differently component files, usually denoted with the '_ch40b' suffix.

Step 2.9.a.: Run `60_ch40b.do` up until line 704.

Step 2.9.b.: Run `zone_assignments_ch40b.ipynb`

Step 2.9.c.: Run `closest_boundary_matches_ch40b.ipynb`

Step 2.9.d.: In `60_ch40b.do`, run line 710

Step 2.9.e.: In `60_ch40b.do`, run line 712

Step 2.10: Compile the final dataset

In Stata, run `70_final_dataset.do`. There isn't much to describe for this step as the file essentially just merges all of the disparate output from Steps 1-9 into the final (or nearly final) dataset.

> **Note**
>
> One important thing to note is that the output `final_dataset.dta` is a generic name. In many of the analysis setup files there is reference to `final_dataset_10-28-2021.dta`. This is the same file but date-stamped to preserve the output at the time. Another variation is `final_dataset_town_comparisons.dta`, which differs in that the municipality boundaries are not controlled for when zoning boundaries are assigned.

The required input files are:

- `warren_MAPC_all_annual.dta`

- closest_boundary_matches_with_regs.dta
- warren_density_measures.dta
- costar_warren_xwalk.dta
- nhpd_warren_xwalk.dta
- chapter40b_warren_xwalk.dta
- costar_rent_hist.dta
- costar_mf_all.dta
- nhpd_mapc.dta
- chapter40b_mapc.dta
- closest_boundary_matches_nhpd_with_regs.dta

The result of this step is the following: - final_dataset.dta

Step 2.11, run 80_amentiy_datasets.do

In Stata, run 80_amentiy_datasets.do. This file takes a number of .shp files, converts them for use in Stata (data and corresponding coordinate .dta files), and calcualtes the distance from a warren property to the 'closest' amenity (i.e., nearest road, green space, etc.).

Technically speaking this step can be completed at any point after Step 2.2, and only requires that warren_MAPC_all_unique.dta be present.

As a result of this step, warren_MAPC_all_unique_closest_stuff.dta will be created which has the ID for every Warren Group property in final_dataset.dta, along with the distance from that property to various amenties.

Step 2.12, create the striaght line boundary files

The original zoning boundaries used for analysis were amended to only include those which are straight line boundaries. The process is an off-shoot of the closest boundary matches and identifies if a Warren Group property falls along a straight segment of a zoning boundary, per Turner et al. (2014).1

Step 2.12.a.

In Jupyter Notebook, run all parts of closest_boundary_matches_mtlines.ipynb. It will it to create closest_boundary_matches_mtlines.csv in the same path.

Step 2.12.b.

In Stata, run mt_orthogonal_lines.do, located under /code/data_setup/miscellaneous_setup_files. Ensure that the correct path is specified to the ouput of Step 12.a. above.

The result of this step will be mt_orthogonal_dist_100m.dta file with the variable straight_line. This variable equals one (1) if the property lies on a straight-line segment of the assigned zoning boundary, else equals zero (0). As with Step 2.10 this is a generic file output name and will be referenced as mt_orthogonal_dist_100m_07-01-22_moreregs.dta

1 Turner, Matthew A, Andrew Haughwout, and Wilbert VanDer Klaauw, "Land Use Regulation and Welfare," Econometrica, 2014, 82 (4), 1341–1403.

Step 2.13: Create the no roads files

The 'no roads' boundaries are the orignal zoning boundaries we identified with all overlapping roadways removed (not just highways and major roads). Unlike the original zoning boundaries, the entire process of matching to the closest boundary and assigning regulations is down in the .ipynb file.

In Jupyter Notebook, run all parts of `closest_boundary_matches_noroads.ipynb`. Note that this file uses a number of one-off datasets.

- `adm3_no_roads_crs26986.shp` - is a version of the final zoning boundaris but with all overlapping roads removed. This was created in ArcGIS using `adm3.shp` and. The complete road network shapefile `EOTROADS.shp`.
- `warren_address_points_assigned.shp` - This is a simple export of the warren group properties with the latitude/longitude locations. It was made at the time to more easily code up the file but any file format can be used so long as the unique ID and coordinates variables are present.
- `regulation_types_moreregs.dta` - holds the boundary regulation data.

The result will be `closest_boundary_matches_noroads.csv`, which contains warren group properties matched to zoning boundaries that do not overlap any roads.

Step 2.14: Calculate trasit distances

There are two `.ipynb` files that handle public transit distance measures. This step covers both of them.

Step 2.14.a.

In Jupyter Notebook, run `dist_prop_to_station.ipynb`. This script calcualtes the distance of a warren property to it's closest public transit train station using both a manhattan (to approximate walking distance) and euclidean method.

The result will produce the file `transit_distance.csv`

Step 2.14.b.

In Jupyter Notebook, run `dist_to_south_station.ipynb`. This script calculates the distance of warren property to South Station in downtown Boston, used as a central reference point for the main business district in the region.

The result will produce the file `dist_south_station_2022_09_29.csv`

Step 2.15: Assign soil quality data

In a Python IDE, run `soil_quality_matching.ipynb`. This script assigns warren group properties to the corresponding parcel of soil quality data found in `Soil_Parcel_Data_Shape.shp`

The result will produce `soil_quality_matches.dta`.

Step 2.16: Create walkability data

This step does not have a defined file associated with handling the process. It was produced in ArcPro in early 2024 and involved spatially matching warren group properties to the polygons found in Natl_WI.gbd. The version used to complete this step is ArcPro v3.6.

Step 2.17: Create `final_dataset_town_comparisons.dta`

This `.dta` is used as input for `amenities_muni_boundary.do` only. You are essentially following steps 2.4, 2.5 and 2.10 but with the following modifications:

- Instead of `closest_boundary_matches.ipynb`, run `closest_town_boundary_matches.ipynb`.
- Instead of `20_boundary_matches.do` run `warren_town_boundary_matches.do`. This will save `warren_town_closest_matches_with_regs.dta`
- Instead of 70_final_dataset.do, run `final_dataset_town_boundary_comparisons.do`

3. Analysis

So long as Data Setup Steps 1–16 have been completed successfully the required input files for the analysis code files should be present in the `/data` directory.

Step 3.1: Run the master analysis file

In Stata, run `analysis_master_file.do`. This file sets all necessary global path variables that will be used throughout the analysis process. Any changes to the global path variables to match a user's local directory structure should be made here. The most important global variables to set correctly are `$DATAPATH`, which should refer to the main directory storing the data files, and `$EXPORTPATH`, which should point to the directory where any output will be stored.

Step 3.2: Run `analysis_within_town_setup.do`

In Stata, run `analysis_within_town_setup.do`. This file will handle a lot of intermediate setup that is common throughout most of the analysis files. It will also create `within_town_analysis_data.dta`, which is a kind of *final* `final_dataset.dta`.

Prior versions of the code called `analysis_within_town_setup.do` from within the overall analysis file. However, `analysis_within_town_setup.do` takes a long time to run and so it is recommend that the output is stored for repeated use.

Step 3.3 (optional): Run the remaining analysis setup files

In addition to `analysis_within_town_setup.do`, there are two (2) additional analysis setup files that handle distince procedures for their corresponding analysis files. These are:

- `analysis_noroads_setup.do`, which is used by `main_noroads.do`
- `analysis_town_comparisons_setup.do`, wich is used by `amenities_muni_boundary.do`

Both setup files take a while to run, however since they only relate to one analysis file they are not coded to save output at this time. However, you may want to save the ouput from these files if you plan to run their corresponding analysis files multiple times in order to save time.

Step 3.n: Run the remaining analysis files

For details on which tables/figures correspond to which analysis files, refer to The File-to-Figure Map section or the Analysis Files Guide.

The remaining analysis files can be run in any order, with two (2) exceptions:

1. If you plan to run either `main_noroads.do` and/or `amenities_muni_boundary.do`, but have not saves the output from Step 3, you must do so beforehand or confirm they are called within their correspond analysis files.
2. The `counterfactual_` files must be run in the order denote by their infix. Specifically the order should be follows:
    1. `counterfactual_01_spatial_heterogeneity.do`
    2. `counterfactual_02_train_station_means`
    3. `counterfactual_03_means.do`
    4. `counterfactual_04_calculations_combined.do`

File-to-Figure Map

Below are the figure/table paper references mapped to the corresponding file which generates some or all components.

Tables-to-Files

Table Ref	File
Table 1	table1_replication.do
Table 2	amenities_mtlines.do predicted_prices_mtlines.do
Table 3	within_town_mtlines.do within_town_mtlines_robustse.do
Table C.1	amenities_mtlines.do
Table C.2	amenities_muni_boundary.do
Table C.3	residuals.do
Table C.4	within_town_mtlines.do within_town_mtlines_robustse.do
Table C.5	within_town_mtlines.do within_town_mtlines_robustse.do
Table C.6	within_town_mtlines.do within_town_mtlines_robustse.do
Table C.7	within_town_mtlines.do within_town_mtlines_robustse.do
Table C.8	chars_mtlines.do robustness_mtlines.do
Table C.9	chars_mtlines.do robustness_mtlines.do
Table C.10	chars_mtlines.do robustness_mtlines.do
Table C.11	external_effects.do
Table C.12	robustness_mtlines.do
Table C.13	external_effects.do
Table D.1	counterfactual_01_spatial_heterogeneity.do
Table D.2	counterfactual_01_spatial_heterogeneity.do
Table E.1	1
Table E.2	2

1 Non-exhaustive list meant to illustrate recent reforms, based on various news sources.

2 Constructed based on data from Knauss, Norman L, Zoned Municipalities in the United States, Vol. 374, Division of Build- ing and Housing, Bureau of Standards, 1933.

Figures-to-files

Figure Ref	File
Figure 1	main_mtlines.do
Figure 2	chars_mtlines.do
Figure 3	main_mtlines.do
Figure 4	main_mtlines.do
Figure 5	counterfactual_04_calculations_combined.do
Figure A.1	3
Figure A.2	4
Figure A.3	existence.do
Figure A.4	histogram.do
Figure A.5	histogram.do
Figure A.6	figure_A6.do 5
Figure A.7	straight_line_boundary_map.do
Figure C.1	amenities_mtlines.do
Figure C.2	amenities_mtlines.do

Figure Ref	File
Figure C.3	main_mtlines.do
Figure C.4	main_mtlines.do
Figure C.5	chars_mtlines.do main_noroads.do
Figure C.6	main_mtlines.do
Figure C.7	main_mtlines.do
Figure C.8	robustness_mtlines.do
Figure C.9	robustness_mtlines.do
Figure C.10	robustness_mtlines.do
Figure C.11	bindingness.do chars_mtlines.do
Figure C.12	main_noroads.do robustness_mtlines.do
Figure D.1	straight_line_boundary_map.do 6
Figure E.1	regulations_map.do
Figure E.2	regulations_map.do
Figure E.3	regulations_map.do
Figure E.4	assessed_vs_sales.do
Figure E.5	included_excluded_towns.do
Figure E.6	straight_v_walking_dist.do
Figue E.7	warren_group_property_map.do
Figure E.8	7

3 Diagram made via MS Paint.

4 Screenshots of online mapping software overlaying historic zoning area raster images with final boundary file.

5 The main parts of the figure were made in ArcGIS, calculations for the boundary lengths are in figure_A6.do.

6 A color version of Figure A.7

7 Screenshot of MAPC service region cities and towns by community types.

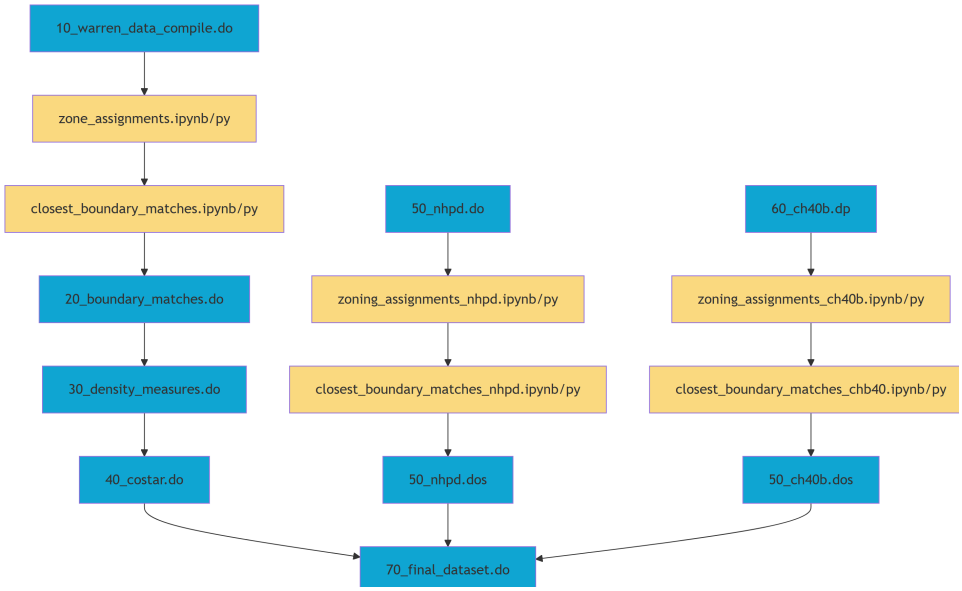
Data Setup Files Guide

Overview

The files under the `/code/data_setup` create the main datasets used in the working paper. The files under `/code/data_setup/python_programs` handle core setup functions that were done using Python and Jupyter Notebooks. The files under `/code/data_setup/miscellaneous_setup_files` handle smaller setup or one-off data processing jobs.

Order of files

Setup files should be run in a specific order. This will include python programs detailed elsewhere. The workload jumps between Stata and Python code files. Below is an outline the general order of file:



> **!** Important
>
> Directory structure is important in order to limit errors that come with running the code files. To the best of our ability we have tried to generalize the paths defined in code so that *in theory* a user can simple download this repository and change a few key components. Despite our best efforts, there are bound to be areas we missed, and some of the directory structure has likely changed over time. We have provided `_tree_data_dir.txt` in an effort to illustrate where some files should be located that were not able to be shared in this repository.

Main Setup Files

`00_data_setup_master_file.do`

Description

This .do file is a master .do file for the data setup files that calls all relevant sub files in the order they should be run. It is also sets global file paths used throughout the setup process.

Inputs

n/a

Outputs

n/a

Sub-scripts

n/a

`10_warren_data_compile.do`

Description

Uses the Warren Group MA time series property file to construct 4 data sets. These datasets contain total properties for each year between 2007-2019 and a unique set of every unique property record during those years.

`MA_assessor_annual_expanded.dta` is the source warren data held by the Federal Reserve Bank of Boston (see data sources for details).

Inputs

- `MA_assessor_annual_expanded.dta` - all residential properties in MA, unique by year and prop_id
- `cb_2018_25_cousub_500k_shp.dta` - coordinates file of converted county subdivision shapefile
- `cb_2018_25_cousub_500k.dta` - data file of converted county subdivision shapefile
- `MAPC_town_list.dta` - hand-coded list of cities and town in MAPC region

Outputs

- `warren_MA_all_annual.dta` - all residential properties in MA, unique by year and prop_id
- `warren_MA_all_unique.dta` - a unique list of all properties in MA, by prop_id
- `warren_MAPC_all_annual.dta` - all residential properties in the MAPC region, unique by year and prop_id
- `warren_MAPC_all_unique.dta` - unique list of all residential properties in the MAPC region,

Sub-scripts

- `11_geocoding.do` fixes some lat/lon geocoding issues present in initial raw data via to fix lat/lon geocoding issues
- `12_res_types.do` condenses property type classification variables and trims the dataset of non-residential properties.
- `13_condo_collapse.do` collapses condo buildings so the number of units is summed to the address (note in the end condos were excluded because they are not captured uniformly across municipalities.)

`20_boundary_matches.do`

Description

Requires running a series of python scripts beforehand to have the necessary input files (see Python Setup Files).

The file takes the output of closest_boundary_matches.ipynb and finds the best closest boundary match between warren group property and MAPC zoning boundary. Assigns home zoning regulations and regulations to the comparison zoning area on the other side of the boundary.

`regulation_types.dta` is a file constructed using ArcGIS directly from the MAPC zoning atlas, and contains the paired (home and neighbor) boundary-level zoning regulations data that is matched onto each boundary-property match. `warren_town_boundary_matches.do` is a miscellaneous setup file version of this program that handles the town boundary matching.

Inputs

- `closest_boundary_matches.csv` - output from `closest_boundary_matches.ipynb` that matches a warren group property to the five closest zoning boundaries.
- `regulation_types.dta` - zoning area level regulations data.

Outputs

- `closest_boundary_matches_with_regs.dta` - property level dataset with matched zoning boundary regulations.

Sub-scripts

n/a

`30_density_measures.do`

Description

Calculates the share of properties that are single-family and 2-3 units around .1 miles of every property record that is 1 mile or less from the zone boundary.

Inputs

- `warren_MAPC_all_annual.dta` - all residential properties in the MAPC region, unique by year and prop_id.
- `closest_boundary_matches_with_regs.dta` - property level dataset with matched zoning boundary regulations.

Outputs

- `warren_density_measures.dta` - property level dataset that has data on local density measures, within one mile of property.

Sub-scripts

n/a

`40_costar.do`

Description

Imports all data from excel file downloads and stores in one Stata .dta file. Uses the first row as variable headers. Data was downloaded from CoStar.com in
batches for all city and towns in the MAPC service region. Contains data only on multi-family properties in CoStar which usually excludes 1-4 unit properties.

This main file also calls 2 sub-files that can be run independently to export a warren → costar crosswalk and a costar rent history dataset.

CoStar data had to be hand scraped from their website due to limitations with the platform. Refer to the `_tree.txt` file under `/data/costar/` for the structure.

For the property rent history data there is a missing python file `costar_props.ipynb` that iterated through the property rent history directory to combine these all into one excel file.

Inputs

Multiple costar input .xlsx files from hand-scraping the CoStar data.

- `cb_2018_25_cousub_500k.dta/cb_2018_25_bg_500k_shp.dta` and related shapefile for geocoding.

Outputs

- `costar_mf_all.dta` - all costar property data within the MAPC region, result of hand-scraping.
- `costar_mf_destring.dta` - a de-stringed version of the above for easier use later on.

Sub-scripts

- `41_costar_warren_xwalk.do` - matches warren group properties to the corresponding costar data.
- `42_costar_rent_history.do` - compiles the rent history data for costar properties and matches to the warren group data.

`50_nhpd.do`

Description

Cleans the original 'All Properties.xlsx' download. Returns 2 datasets of (1) all properties in MA and all properties in MAPC region. The MAPC region file also has boundary IDs assigned to properties (for use in the Warren/NHPD crosswalk).

NHPD properties are matched independently to MAPC zoning boundaries via their own python file.

Inputs

- `All Properties.xlsx` - the raw data downloaded from www.preservationdatabase.org
- `cb_2018_25_cousub_500k.dta/cb_2018_25_bg_500k_shp.dta` and related shapefile for geocoding.

Outputs

- `nhpd_ma.dta` - subsidized property data for all of massachusetts
- `nhpd_ma.dta` - subsidized property data for MAPC region

Sub-scripts

- `51_nhpd_boundary_matches.do` - matches NHPD properties to their corresponding zoning boundary.
- `52_nhpd_warren_xwalk.do` - matches warren group properties to the corresponding costar data.

`60_ch40b.do`

Description

Cleans the original ch40b file. Returns two datasets of (1) all ch40b properties in raw but usable format and (2) a clean version with boundary IDs attached (for use in the warren/ch40b crosswalk).

Inputs

- `Data for Nick Chiumenti 12-23-20_match_criteria.xlsx` - is the original data with corresponding cleaned addresses. SHI refers to the subsidized housing inventory reference list. The `jordan_` and `roy_` files are additional small project mappings done for properties not identified previously.
- `chapter40b_geocoder_export_20211020.csv` is the lat/lon coordinates geocoded using the census geocoder batch api.

Outputs

- `ch40b_ma.dta` - ch40b data for all of massachusetts
- `ch40b_ma.dta` - ch40b property data for MAPC region

Sub-scripts

- `61_ch40b_boundary_matches.do` - matches ch40b properties to their corresponding zoning boundary.
- `62_ch40b_warren_xwalk.do` - matches warren group properties to the corresponding costar data.

`70_final_dataset.do`

Description

Creates the final dataset before analysis stage. This files combines the warren property data, and boundary matches, CoStar, NHPD, and ch40b data, and the density measures into one final dataset that is used as the basis for almost all analysis files. `final_dataset_town_boundary_comparisons.do` is a version of this file that creates the town boundary matches final dataset.

> **Note**
>
> The Chapter 40B and NHPD data are not used in the final submitted manuscript. Even though they are not used they still included here because they retain the ability to run these files without interruption as editing them to omit their inclusion would likely result in files that error-out.

Inputs

- `warren_MAPC_all_annual.dta` - all residential properties in the MAPC region, unique by year and prop_id.
- `closest_boundary_matches_with_regs.dta` - warren group properties with their assigned closest boundary segment and associated zoning regulations.
- `warren_density_measures.dta` - warren group properties with the local area density measures (gentle and high density).
- `costar_` files contain the costar to warren crosswalks, the rent history, and the property information.
- `nhpd_mapc.dta` - is the match nhpd property information for subsidized housing
- `chapter40b_mapc.dta` - is the matched chapter 40b property information.

Outputs

- `final_dataset.dta` - represents the final working version of the analysis data, used as the basis for all subsequeny files. In practice, this file was created multiple times.

> **Note**
>
> The version `final_dataset_10-28-2021.dta` represents the final version which is used and is referenced in essentially all analysis files. In practice, this final working file is altered susequently to respond to various comments, changes, and referee requests.

Sub-scripts

n/a

`80_amenity_datasets.do`

Description

This file combines a number of 'amenities' into one .dta file to merge on for analaysis later one.

Essentially, this file compiles the distance between warren group properties and the closest amenity type (schools, green space, roads, rivers, etc.)

This file is not part of the creation of the main working data but is integral to susequent analysis.

Inputs

All input datasets are conversions of shape files, with the distance to the closest feature measured using *geonear*.

Outputs

`warren_MAPC_all_unique_closest_stuff.dta` - contains warren group property ids and distances to the closest amentiy feature.

Sub-scripts

n/a

Python Setup Files

`zone_assignments.ipynb`

Description

This program takes the unique set of all warren group property tax records in the MAPC region and assigns them to (1) a schools attendance area `ncessch`, (2) a zone use type area `zo_usety`, and (3) a left/right boundary id and regulation type area `l_r_fid` and `reg_type`. The exported .csv file is used in the `closest_boundary_matches.ipynb` program.

> **Note**
>
> There are multiple iterations of this program, denoted by their suffix, each handling a different input data source. The base `zone_assignments.ipynb` handles the Warren Group properties.
> - zone_assignments.ipynb
> - zone_assignments_ch40b.ipynb
> - zone_assignments_nhpd.ipynb

Inputs

- `warren_MAPC_all_unique.dta`
- `./sabs_unique_latlong.shp`
- `./roads_mapc_union_sd_dissolved.shp`
- `./zoning_atlas_latlong.shp`

Outputs

- `./zone_assignments_export.csv`

Sub-scripts

n/a

`closest_boundary_matches.ipynb`

Description

This is a version of the original boundary matching file used to match address points to zoning boundaries. The output of this program is used as input for ./20_boundary_matches.do

> **Note**
>
> There are multiple iterations of this program, denoted by their suffix, each handling a different cut of data.
> - closest_boundary_matches.ipynb
> - closest_boundary_matches_ch40b.ipynb
> - closest_boundary_matches_nhpd.ipynb
> - closest_boundary_matches_mtlines.ipynb
> - closest_boundary_matches_noroads.ipynb
> - closest_town_boundary_matches.ipynb
>
> `closest_boundary_matches_mtlines.ipynb` is used in the final analysis but only identifies straight line boundaries. `closest_boundary_matches.ipynb` is what matches the actual boundaries to warren properties.

Inputs

- zone_assignments_export.csv
- adm3_latlong.shp
- regulation_types.dta

Outputs

- closest_boundary_matches.csv

Sub-scripts

n/a

---

soil_quality_matching.ipynb

Description

This file takes the soil quality data .shp file created by an RA and matches onto it our dataset of mapc warren group properties.

Inputs

- Soil_Parcel_Data_shape.shp
- warren_MAPC_all_unique.dta

Outputs

- soil_quality_matches.dta

Sub-scripts

n/a

---

all_stations.ipynb

Description

This file compiles the commuter rail stations and the mbta rapid transit stations into one file to be used in calculating the distance to downtown measure in amenities_mtines.do.

Inputs

- TRAINS_NODE.shp
- MBTA_NODE.shp

Outputs

- all_stations.csv

Sub-scripts

n/a

---

dist_prop_to_station.ipynb

Description

This file calculates the distance from a property to its closest train stop in manhattan and euclidean distance. The output is used in the amenities file.

Inputs

- all_stations.csv
- warren_MAPC_all_unqiue.dta

Outputs

- transit_distance.csv

Sub-scripts

n/a

---

dist_to_south_station.ipynb

Description

This program takes a dateset of MBTA and commuter rail stations and calculates the travel distance from that station to South Station in downtown Boston, MA. The output of this program is eventually combined with the manhattan and euclidean distance output and used in the amenities file.

Uses the HERE transit routing api to calculate the travel distance (in meters)

Note: The transit routes are date/time dependent and so will change depending on when the program is run.

Inputs

- all_stations.csv

Outputs

- dist_to_south_station.csv

Sub-scripts

n/a

---

station_boundary_dist.ipynb

Description

Takes the train stations file all_stations.csv and calcualtes the distance between that and the zoning boundaries in adm3.shp. Exports station_boundary_dist.csv.

Inputs

- adm3_latlong.shp
- all_stations.csv

Outputs

- station_boundary_dist.csv

Sub-scripts

n/a

`walking_distances_orsm.ipynb`

Description

Calculates the walking distances (in meters) between closest properties to a boundary on either side of a boundary. Requires running effective_dist_export.do first

Inputs

- `effective_distance_inputs.csv`

Outputs

- `effective_distance_outputs.csv`

Sub-scripts

n/a

`BatchAddressMatch_final.ipynb`

Description

Takes an exported set of warren group properties from geocode_fixes.do that have incorrect lat/lon geocoding and uploads them to the census's geocoder api website, downloads the correct lat/lon coordinates and saves it to a new file. Note, that this file also handles some random geocoding tasks throughout the setup process for data no longer used in the analysis (chap 40b and nhpd data).

Inputs

- `warren\geocode_fixes\<various>.txt`

Outputs

- `warren\geocode_fixes_<date>.csv`

Sub-scripts

n/a

Miscellaneous Setup Files

`effective_dist_export.do`

Description

Creates the effective distances input `.csv` file used in `walking_distances_orsm.ipynb`.

Inputs

- `final_dataset_10-28-2021.dta`

Outputs

- `effective_distance_inputs.csv`

Sub-scripts

- n/a

`shp2dta.do`

Description

Extracts '.shp' and saves them as '.dta' files for use in mapping.

Inputs

various .shp files

Outputs

various .dta files

Sub-scripts

n/a

`town_list_export.do`

Description

Creates two .dta files with a list of admissable cities and towns in Massachusetts and in the MAPC region.

Inputs

- `2019_gaz_cousubs_25.txt`
- `MAPC_town_list.txt`

Outputs

- `MA_cousub_list.dta`
- `MAPC_town_list.dta`

Sub-scripts

n/a

`warren_geocode_fixes.do`

Description

Identifies those properties in the Warren data that have miss-coded geo-ids, identified by those where the lat/long do not match the city/town they are in. It is assumed that the city/towns are correct because those come from the town tax assessors and have the least likelihood of being wrong.

Inputs

- `MA_assessor_annual.dta`
- `address_corrections_input.dta`

Outputs

- `geocoder_export_`DateStamp'.csv'
- `address_corrections_input.dta`
- `address_corrections_output.dta`

Sub-scripts n/a

Analysis Files Guide

Overview

`/code/analysis_files` contains any and all files that conduct analysis or output figures and tables used in Under the (Neighbor)Hood: Understanding Interactions Among Zoning Regulations.

Code Structure

> **Note**
>
> To easily find references to tables and/or figures in code files, search for text tag `[PAPER SOURCE]` using `ctrl/cmd+f`.

All `./analysis_files` files are Stata `.do` files and must be run with a working version of Stata.

Almost all .do files begin with the same preface. Below is an example from `amenities_mtlines.do`:

example file preface

```
* start here
clear all
log close _all
set linesize 255

local name ="amenities_mtlines"  // <--- change when necessry

* create an output directory if none exists
global EXPORTPATH "$WORKINGDIR/analysis/`name'_output"

capture confirm file "$EXPORTPATH"

if _rc != 0 {
    di "making directory $EXPORTPATH"
    shell mkdir $EXPORTPATH
}

* start log file
local date_stamp : di %tdCY-N-D date("$S_DATE","DMY")

log using "$EXPORTPATH/`name'_log_`date_stamp'.log", replace
```

All `.do` files will have a header. Again an example from `amenities_mtlines.do`:

example file header

```
*******************************************************************************
* File name:        amenities_mtlines.do
*
* Project title:    Under the (Neighbor)Hood: Understanding Interactions Among
*                   Zoning Regulations
*
* Description:      Primarily a robustness focused file that tests the model
*                   against various amenities indicators to check if there is
*                   any discontinuity across boundaries.
*
* Inputs:           mt_orthogonal_dist_100m_07-01-22_v2.dta
```

```
*                   dist_south_station_2022_09_29.csv
*                   transit_distance.csv
*                   soil_quality_matches.dta
*                   warren_group_walkability.dta
*                   within_town_analysis_data.dta
*
* Outputs:          Table 2, Figure C.1 (a-e), Table C.1 means
*
* Date Created:     06/23/2021
*
* Last Updated:     01/09/2026
********************************************************************************
```

File Descriptions

`analysis_master_file.do`

Description:

Sets global file paths and Stata run options for analysis files. Can be set to run through all analysis files automatically, if needed.

> **!** Important
>
> This file ***must*** be run first as it sets global paths for all analysis files. The most important globals to set are `$DATAPATH`, which needs to point to the `/data` directory, and `$EXPORTPATH`, which will store `.gph`, `.pdf` and `.tex` files.

Inputs:

n/a

Outputs:

n/a

`analysis_within_town_setup.do`

Description

This setup file takes the cleaned warren group, costar, etc. data that is the result of the `./data_setup .do` files and prepares it for use in the analysis files. It handles a number of extra setup steps that were added over the life of the project. There is still plenty of other setup done within each analysis file, this merely handles the bulk of the common setup steps.

It creates `within_town_analysis_data.dta`, which is the primary default input data for most analysis files.

Inputs

- `final_dataset_10-28-2021.dta`
- `warren_MAPC_all_unique_closest_stuff.dta`
- `warren_sales_data.dta`
- `CPI_2019.dta`
- `costar_mf_destring.dta`
- `costar_rent_hist.dta`

Outputs

- within_town_analysis_data.dta

---

analysis_noroads_setup.do

Description

This setup file is simialr in purpose to analysis_within_town_setup.do but specific for no roads analysis, of which “main_noroads.do” is the only real analysis file run currently that uses it.

Its chief difference is that it uses closest_boundary_matches_noroads.csv instead of closest_boundary_matches.csv

Inputs

- closest_boundary_matches_noroads.csv
- final_dataset_10-28-2021.dta
- warren_MAPC_all_unique_closest_stuff.dta

Outputs

None, is coded to run as a sub-script of main_noroads.do with option to save analysis_noroads_data.dta in that file to save on repeat runtimes.

---

analysis_town_comparison_setup.do

Description

This setup file is simialr in purpose to analysis_within_town_setup.do but specific for no roads analysis, of which amenities_muni_boundary.do is the only real analysis file run currently that uses it.

Its chief difference is that it is coded up based on final_dataset_town_comparisons.dta which uses different identifiers for lam_seg variables.

Inputs

- dist_south_station_2022_09_29.csv
- transit_distance.csv
- soil_quality_matches.dta
- warren_group_walkability.dta
- final_dataset_town_comparisons.dta

Outputs

None, is coded to run as a sub-script of amenities_muni_boundary.do

---

amenities_mtlines.do

Description:

Primarily a robustness check file that tests the model against various amenities indicators to check if there is any discontinuity across boundaries.

Inputs

- mt_orthogonal_dist_100m_07-01-22_v2.dta
- dist_south_station_2022_09_29.csv
- transit_distance.csv
- soil_quality_matches.dta

- warren_group_walkability.dta
- within_town_analysis_data.dta

Outputs

- Table 2;
- Figure C.1
- Figure C.2
- Table C.1 means

---

amenities_muni_boundary.do

Description

File is similar to amenities_mtlines.do except it compares across town boundaries instead of zoning boundaries. Calls analysis_town_comparisons_setup.do.

Inputs

- dist_south_station_2022_09_29.csv
- transit_distance.csv
- soil_quality_matches.dta
- warren_group_walkability.dta
- final_dataset_town_comparisons.dta

Outputs

- Table C.2

---

bindingness.do

Description

Analyzes bindingness of different regulations and boundaries with binding regs.

Inputs

- mt_orthogonal_dist_100m_07-01-22_moreregs.dta
- soil_quality_matches.dta
- warren_zoning_regulations_match.dta
- within_town_analysis_data.dta
- blocks_2010.dta
- acs_amenities.dta

Outputs

- Table C.11

---

chars_mtlines.do

Description

Handles all of the analysis on neighborhood characteristics acrsoss zoning boundaries.

Inputs

- mt_orthogonal_dist_100m_07-01-22_v2.dta

- within_town_analysis_data.dta

Outputs

- Table C.8
- Figure 2
- Figure C.11(a)
- Table C.9
- Table C.10
- Figure C.11
- Figure C.5

`counterfactual_01_spatial_heterogeneity.do`

Description

Runs the spatial heterogeneity file for boston zoning paper.

Inputs

- mt_orthogonal_dist_100m_07-01-22_v2.dta
- within_town_analysis_data.dta

Outputs

- Table D.1
- Table D.2
- inputs for subsequent counterfactual files

`counterfactual_02_train_station_means.do`

Description

This file is part 2 of the counterfactual analysis. It calculates means around train stations using the finalized set of warren group data.

Inputs

- mt_orthogonal_dist_100m_07-01-22_v2.dta
- within_town_analysis_data.dta
- station_boundary_dist.csv
- adm3_crs4269.dta
- cb_2018_25_cousub_500k_shp.dta
- cb_2018_25_cousub_500k.dta
- all_stations.csv

Outputs

- train_station_means.dta, which is an input for counterfactual #3.

`counterfactual_03_means.do`

Description

This is part 3 of the counterfactual analysis.This file calcs a bunch of means at different levels (property, boundary, etc.). It then creates a town level version that also has train station level means attached. If run in order, on the same day, counterfactual output should be in the same folder.

Inputs

- `mt_orthogonal_lines/mt_orthogonal_dist_100m_07-01-22_v2.dta`
- `within_town_analysis_data.dta`
- `means_town_lvl.dta`
- `means_town_lvl_tomerge.dta`

Outputs

- `means_lpm.dta`
- `means_property_lvl.dta`
- `means_boundary_lvl.dta`
- `means_town_lvl.dta`
- `means_town_lvl_tomerge.dta`
- `means_town_train_stations.dta`

`counterfactual_04_calculations_combined.do`

Description

This dofile imports the means calculated in prior counterfactual #3 prior and prepares the dataset to be merged with data on coefficients, calculates policy numbers and plots them on a map.

Inputs

- `means_lpm.dta`
- `means_property_lvl.dta`
- `means_boundary_lvl.dta`
- `means_town_lvl.dta`
- `means_town_lvl_tomerge.dta`
- `means_town_train_stations.dta`

Outputs

- Figure 5

`existence.do`

Description

Creates the A.3 scatter plot of number of boundaries per acre relative to various soc-econ measures.

Inputs

- `mt_orthogonal_dist_100m_07-01-22_v2.dta`
- `dist_south_station_2022_09_29.csv`
- `transit_distance.csv`
- `soil_quality_matches.dta`
- `warren_group_walkability.dta`
- `within_town_analysis_data.dta`
- `blocks_2010.dta`
- `acs_amenities.dta`
- `muni_tax_rates.dta`

- muni_land_area.dta

Outputs

- Figure A.3

---

external_effects.do

Description

Near-far external lot analysis following Turner et al. straight line boundaries (matt turner orthogonal lines) for house prices, rents. regression output is tables only. Printed w/o characteristics or exclusions (a) and w/ (b).

Inputs

- mt_orthogonal_dist_100m_07-01-22_moreregs.dta
- within_town_analysis_data.dta

Outputs

- Table C.13

---

histogram.do

Description

Makes a bunch of histograms and scatter plots for use in paper. Note the final alpha-num sequence may have changed.

Inputs

- within_town_analysis_data.dta

Outputs

- Figure A.4
- Figure A.5

---

main_mtlines.do

Description

Main regression specifications for the paper. Runs through multiple models and specifications. Produces almost all headline analysis in the paper.

Inputs

- mt_orthogonal_dist_100m_07-01-22_v2.dta
- dist_south_station_2022_09_29.csv
- transit_distance.csv
- soil_quality_matches.dta
- within_town_analysis_data.dta

Outputs

- Figure 1
- Figure C.3
- Figure C.4

- Figure 3
- Figure C.6
- Figure 4
- Figure C.7

`main_noroads.do`

Description

Runs a version of the main regression specifications for the paper that uses a version of the boundaries which removes all roads, not just the major ones, that overlap boundaries

Inputs

- `mt_orthogonal_dist_100m_07-01-22_v2.dta`
- `analysis_noroads_data.dta`
- `dist_south_station_2022_09_29.csv`
- `transit_distance.csv`
- `soil_quality_matches.dta`
- `within_town_analysis_data.dta`
- `blocks_2010.dta`
- `acs_amenities.dta`

Outputs

- Figure C.12
- Appendix C.5.1

`predicted_prices_mtlines.do`

Description

Calculates predicted prices in Table 2

Inputs

- `mt_orthogonal_dist_100m_07-01-22_v2.dta`
- `analysis_noroads_data.dta`
- `dist_south_station_2022_09_29.csv`
- `transit_distance.csv`
- `soil_quality_matches.dta`
- `walkability.dta`
- `within_town_analysis_data.dta`

Outputs

- Table 2 Panel (C)

`residuals.do`

Description

Calculates residuals for Table C.3

Inputs

- `mt_orthogonal_dist_100m_07-01-22_v2.dta`
- `analysis_noroads_data.dta`
- `dist_south_station_2022_09_29.csv`
- `transit_distance.csv`
- `soil_quality_matches.dta`
- `within_town_analysis_data.dta`

Outputs

- Table C.3

`robustness_mtlines.do`

Description

This file runs a myriad of robustness checks, not main line specifications. it has been trimmed down considerably. As such, the .do file 'part' numbers are not consecutive but they should align to prior versions of the file.

Inputs

- `mt_orthogonal_dist_100m_07-01-22_moreregs.dta`
- `dist_south_station_2022_09_29.csv`
- `transit_distance.csv`
- `soil_quality_matches.dta`
- `warren_group_walkability.dta`
- `within_town_analysis_data.dta`
- `final_addon_regs_intersect.dta`
- `blocks_2010.dta`
- `acs_amenities.dta`

Outputs

- Figure C.12
- Figure C.8
- Figure C.9
- Table C.12
- Figure C.10

`straight_v_walking_dist.do`

Description

Calculates the straight line (as the crow flies) distance from the closest property to a boundary to it's closest neighbor on the other side. Exports a .csv file to be used in walking_distances_orsm.ipynb to calculate the walking/effective distance between these two properties.

> **Note**
>
> This file depends on the Python program `walking_distances_orsm.ipynb`. See file for details.
> You will need to have a working Python environment to run this file fully, and will need to manually move over to running python if running all parts of this file.

Inputs

- mt_orthogonal_dist_100m_07-01-22_v2.dta
- within_town_analysis_data.dta
- walking_distance_inputs.csv
- walking_distance_outputs.csv

Outputs

- Figure E.6

`table_1_replication.do`

Description

Creates Table 1, thats about it.

Inputs

- mt_orthogonal_dist_100m_07-01-22_moreregs.dta
- dist_south_station_2022_09_29.csv
- transit_distance.csv
- within_town_analysis_data.dta
- blocks_2010.dta
- acs_amenities.dta

Outputs

- Table 1

`within_town_mtlines.do`

Description

Runs specifications for gentle and high density units as well as some endogeneity checks.

Inputs

- mt_orthogonal_dist_100m_07-01-22_moreregs.dta
- within_town_analysis_data.dta

Outputs

- Table 3
- Table C.4
- Table C.5
- Table C.7
- Table C.6

`within_town_mtlines_robustse.do`

Description

Runs specifications for gentle and high density units as well as some endogeneity checks. Similar to the other within town file but **with robust clustering.**

Inputs

- mt_orthogonal_dist_100m_07-01-22_moreregs.dta

- within_town_analysis_data.dta

Outputs

- Table 3
- Table C.4
- Table C.5
- Table C.7
- Table C.6

---

assessed_vs_sales.do

Description

Creates the bin scatter of accessed vs sales price values figure.

Inputs

- MA_assessor_hist.dta
- CPI_2019.dta
- final_dataset_10-28-2021.dta

Outputs

- Figure E.4

---

figure_A6.do

Description

Calculates the boundary length data found in the figure. The actual display part of the figure is made in ArcGIS.

Inputs

- polylines_feasible_new.shp
- mapc_minus_muni_minus_river_minus_roads_new.shp
- mapc_minus_muni_minus_river_minus_roads_minus_attendance_minus_sd_minus_zo_new.shp
- mt_orthogonal_lines.shp

Outputs

Select numbers found in Figure A.6.

---

included_excluded_towns.do

Description

Creates a figure showing which Metropolitan Area Planning Council cities and towns are included or excluded in the analysis.

Inputs

- cb_2018_25_cousub_500k_latlong.dta
- cb_2018_25_cousub_500k_latlong_shp.dta
- MAPC_town_list.dta

Outputs

- Figure E.5

`regulations_map.do`

Description

Shows the three main regulation types used in the paper (dwelling units per acre, building height, and multifamily permitted) by levels of 'strictness'.

Inputs

- `cb_2018_25_cousub_500k_latlong.dta`
- `cb_2018_25_cousub_500k_latlong_shp.dta`
- `zoning_atlas_latlong.dta`
- `zoning_atlas_latlong_shp.dta`

Outputs

- Figure E.1
- Figure E.2
- Figure E.3

`straight_line_boundary_map.do`

Description

Shows the actual zoning boundaries, drawn over MAPC city/town borders, by their final identified type (e.g. only multifamily, only dupac, etc.)

Inputs

- `cb_2018_25_cousub_500k_latlong.dta`
- `cb_2018_25_cousub_500k_latlong_shp.dta`
- `MAPC_town_list.dta`
- `mt_orthogonal_lines_4269.dta`
- `regulation_types.dta`

Outputs

- Figure A.7, Figure D.1 (color version)

`warren_group_property_map.do`

Description

Shows just the vacant (no development but buildable) parcels of land in the MAPC region

Inputs

- `cb_2018_25_cousub_500k_latlong.dta`
- `cb_2018_25_cousub_500k_latlong_shp.dta`
- `final_dataset_10-28-2021.dta`

Outputs

- Figure E.7

Data Sources Guide

We detail below the data sources used for the paper. To the best of our ability we have listed all sources of data used in the report. If gaps exist please reach out so we may update this repository with the missing information.

Any data that we are able to share publicly through this repository is located in the `data` directory.

> **!** Important
>
> We are unable to share sources of data that were made available through the author's prior professional associations with the Federal Reserve Bank of Boston. These specifically relate to the Warren Group's property tax assessment records data, and the CoStar rental property history data. In both cases, the Federal Reserve Bank of Boston and/or the Federal Reserve System is the holder of these data license agreements.

Non-public data sources

Warren Group

Source:

https://www.thewarrengroup.com/

Description:

The Warren Group data is a time series of property tax assessment records for all cities and towns in Massachusetts. The raw data is under a data license agreement and cannot be shared publicly.

Directory location:

`/data/warren`

Costar

Source:

https://www.costar.com/

Description:

The CoStar data is property-level real-estate records encompassing the rental market in Greater Boston. Data is primarily for 4+ unit structures and is aggregated from a variety of source by CoStar, as well as compiled through their own survey efforts. The raw data is under a data license agreement and cannot be shared publicly. For CoStar, property level data had to be hand scraped in batches of 100-1000 records, depending on if it pertained to the property data or the rental history data.

Directory location:

`/data/costar`

Public data sources

Below we have provide links to the source files were original raw input data was downloaded from and/or provided directory path links to the corresponding raw files in the repo.

American Community Survey (ACS)

Source:

ACS 5-year 2009–2018 downloaded from Social Explorer

ACS 5-year 2019 downloaded IPUMS

Description:

ACS data at the block group data was downloaded and cleaned, and then compiled into the final `acs_amenities.dta` file. The code for cleaning the raw data downloads are in `/data/acs/RAW DATA/` under their corresponding year sub-directories.

Directory location:

`/data/acs`

Chapter 40B

Source:

From a data request to the Massachusetts Department of Housing and Community Development in late 2020. Also see Chapter 40B Planning and Information for details.

Description:

The Chapter 40B data covers properties in the state's subsidized housing inventory and that are constructed or qualify under the 1969/1970 statue requiring minimum affordable housing inventory for municipalities in the state. While this data is no longer used in any analysis, it remains a part of the setup code files and so is documented here.

An initial partial list of CH40B properties was received back in December 2020. We received an updated full list of properties in May of 2021.

Directory location:

`/data/chapter40B`

City centers (centroids)

Source:

U.S. Census Cartographic Boundary Files

Description:

The cartographic boundary files are simplified representations of selected geographic areas from the Census Bureau's Master Address File/Topologically Integrated Geographic Encoding and Referencing (MAF/TIGER) System. These boundary files are specifically designed for small scale thematic mapping. Variables `_CX` and `_CY` denote polygon centers and were used to identify city/town centers.

Directory location:

`/data/city_centroid`

Consumer Price Index

Source:

St. Louis Fed Federal Reserve Economic Data, CPI All Urban Consumers

Description:

Basic CPI data covering the 2010–2018 period of the analysis used to adjust prices to 2019.

Directory location:

`/data/fred_cpi`

Green Space

Source:

Metropolitan Area Planning Council Zoning Atlas

Description:

The source file for this MAPC Zoning Atlas. Identifying green/open space is based on the variable `zo_usety`. Code for creating the output file `green_space.dta` file is included under 80_amenity_datasets.do. Additional hand-coded green space records are found in `green_space_save.dta`.

Directory location:

`/data/green_space`

> **Note**
>
> The source files use the cartesian coordinates system. The versions provided were converted to degrees (lat/lon) using Python and EPSG 4269.

Major and Minor Roadways

Source:

MassGIS Data: MassGIS-MassDOT Roads

Description:

This layer is the official state-maintained street transportation dataset available from MassGIS. The layer is a hybrid set of linework, most coming from MassDOT with some additional attributes and linework added by MassGIS. This layer represents all the public and many of the private roadways in Massachusetts, including designations for Interstate, U.S. and State routes.

Directory location:

`/data/roads`

> **Note**
>
> The source files use the cartesian coordinates system. The versions provided were converted to degrees (lat/lon) using Python and EPSG 4269.

Municipality Boundaries

Source:

MassGIS Data: Municipalities

Description:

This layer is the most accurate representation of Massachusetts' municipal (city and town) boundaries; this representation is based on the legislatively approved record of municipal boundaries. Authoritative determination of municipal boundary locations can only be provided by a licensed land surveyor. MassGIS regularly makes corrections or refinements to this data layer as information becomes available; the list of those changes is at the bottom of this web page.

Directory location:

`/data/municipalities`

Metropolitan Area Planning Council (MAPC) Zoning Atlas

Source:

Metropolitan Area Planning Council Zoning Atlas

Description:

The MAPC Zoning Atlas is a web-based mapping archive tool that documents the zoning regulations at the municipal level. It covers about 100 cities and towns in eastern Massachusetts and was developed between 2010 and 2018. The towns included in this atlas fall under MAPC's general service region, as one of the regional planning commissions in the State.

Directory location:

`/data/zoning_boundaries` (raw data) `/data/regulation_data` (regulations only)

> **Note**
>
> This is a core component of the analysis in the paper and many derivative outputs were created from it. The data files under `/data/regulation_data` are a prime example.

National Housing Preservation Database (NHPD)

Source:

National Housing Preservation Database (NHPD)

Description:

The NHPD is an address-level inventory of federally assisted rental housing in the US. The agencies and departments that fund these programs have data on the individual programs that they manage, but there is no central location where all of these data are integrated. This makes it difficult to get a clear picture of the current stock of public and affordable housing in a community. It also means those who wish to preserve public and affordable housing in their community, cannot easily get the information they need about particular properties. By creating the NHPD, the PAHRC and NLIHC hope to address these issues.

As with the Chapter 40B data, this data source is no longer used in any analysis. However, it is a core component of the initial setup files and so is included here.

Directory location:

`/data/nhpd`

School Attendance Boundaries

Source:

NCES School Attendance Boundary Survey (SABS)

Description:

The School Attendance Boundary Survey (SABS) was an experimental survey conducted by NCES' EDGE program with assistance from the U.S. Census Bureau to collect school attendance boundaries for the 2013-2014 and 2015-2016 school years. The shapefiles include feature geometry for elementary, middle, and high school boundaries as well the name, ID, grade span, and other attributes for each school.

Directory location:

`/data/schools`

School Districts

Source:

MassGIS Data: Public School Districts

Description:

These polygon data layers depict the boundaries of public school districts in the Commonwealth of Massachusetts.

Directory location:

`/data/schools`

Soil Quality Data

Source:

MassGIS: Soils SSURGO-Certified NRCS

Description:

NRCS SSURGO-certified soils data for Massachusetts. The raw shapefile data is highly detailed, containing many overlapping layers. The data on this repo was flatten to combine multiple measures of soil quality. The file `Soil_Parcel_data_Shape.shp` is this flattened version as the original raw download was too large to include.

Directory location:

`/data/soil`

> **Note**
>
> An RA cleaned up the initial input dataset in order to make it a flat shape file. Then soil_quality_quality_matching.ipynb matches the soil quality data to the property lots found in Warren Group.

Train Stations & Transit Distances

Source:

MassGIS: MBTA Rapid Transit; MassGIS: Trains

Description:

There are two components to this data. The MBTA rapid transit layers (`MBTA_NODE.shp` and `MBTA_ARC.shp`) represent the core subway system lines and stations within the Greater Boston area. The rail linework and station points (`TRAINS_NODE.shp` and `TRIANS_ARC.shp`) are for passenger, freight, and Amtrak and MBTA Commuter Rail trains.

Directory location:

`/data/train_stops`

Walk Score (Walkability)

Source:

EPA walkability Score

Description:

The National Walkability Index is a nationwide geographic data resource that ranks block groups according to their relative walkability. The national dataset includes walkability scores for all block groups as well as the underlying attributes that are used to rank the block groups.

Directory location:

`/data/walkability`

Water Features (hydrography)

Source:

MassGIS Data: Major Ponds and Major Streams

MassGIS Data: Hydrography (1:100,000)

Description:

The Major Ponds and Major Streams data layers represent a subset of hydrographic features from the Hydrography (1:100,000) layer. Large water bodies and rivers are included in these two layers, respectively, and are meant to be used for plotting small-scale maps. The layers are named MAJPOND_POLY and MAJSTRM_ARC.

Directory location:

`/data/rivers`

> **Note**
>
> The source files use the cartesian coordinates system. The versions provided were converted to degrees (lat/lon) using Python and EPSG 4269.