Replication Data for Economies of Density versus Natural Advantage: Crop Choice on the Back Forty

This document describes the programs and data files required to replicate empirical results in "Economies of Density versus Natural Advantage: Crop Choice on the Back Forty". We do not provide original raw data. (Most raw data are freely available from original sources described in our paper. We provide programs to process these data.) Contact Sanghoon Lee at Sanghoon.lee@sauder.ubc.ca for further questions.

Final Analyses

The following table shows programs and input data sets for each table in the paper. We do not have programs for tables 1 and 2 because they contain simple summary statistics. We used Stata, SAS, and ArcGIS 9.2 for this project. The next section will show the steps required to generate the intermediate input data sets in these analyses.

Table	Programs	Input Data
Table 3	Table_r2.do	Ndqqco.dta
Table 4	Basic Analysis.do Basic_Analysis_init_Search.do	Base.dta
Table 5	Basic Analysis By County.do Base_Data_on_Prime.do Baisc Analysis on Prime Land Only Basic_Analysis_Init_Search.do	Base.dta
Table 6	Fitted_Values.do Decomposition.do	Base.dta crop_name.dta
Table 7	cass_parcel_anal.sas cnty_adm_qq.sas cnty_parcel_qq.sas	cnty_parcel_qq.sas7bdat cnty_adm_qq.sas7bdat
Table 8	Fitted_Values.do Table_Var2.do Table_FakeQ_Var2.do	Base.dta Actualq_fitted_on_fakes.dta
Table 9	FakeQ Analysis.do Basic_Analysis_Init_Search.do County_Admin_Analysis.do Restat_Extended_Analysis_Init_Search.do Ownership_Analysis.do Restat_Extended_Analysis_Init_Search.do	FakeQ.dta Base.dta County_admin.dta Cnty_parcel_qq.dta

Steps to Construct the Intermediate Data

The following describes the steps to construct the intermediate data used in previous section. The names in parentheses indicate the programs used in the steps.

ndqqco.dta

1. (Extractor, ConvertOutputtoSAS.sas) ArcGIS program "Extractor" extracts crop planting, soil type, section id, and XY coordinate (latitude and longitude) at each point of 30 meter by 30 meter grid, from GIS map files. (See the next section for detailed instruction on how to run the ArcGIS program.) ConvertOuputtoSAS.sas cleans up the outputs and converts them into SAS datasets. We obtain about 40.8 million points.
2. (crop_qq.sas, section_ij_xxx.sas) We assign quarter and quarter-quarter section id to each point in the data set obtained in step 1. We aggregate the data up to quarter-quarter section level. The resulting data set has section id, section coordinates (i,j), quarter-quarter section id within a section, average planting for each crop, and the number of grid points for each quarter-quarter section. We throw out points near the boundaries.
3. (soil_qq.sas, macro_soil_qq.sas, mapunit_ndxxx.sas) We aggregate soil type data obtained in step 1, up to quarter-quarter section level using soil_qq.sas and macro_soil_qq.sas. The resulting data set has section_id, section coordinates i,j, quarter-quarter section id within a section, longitude and latitude, and soil type dummies. Each soil type dummy has a percentage of grids points belonging to the soil type. We throw out points near the boundaries.
4. (Soil_US_2002.mdb) We extract soil characteristics for each soil type from SSURGO tabular data. The SSURGO provides an MS-Access template. After reading soil characteristics data into MS-Access, we use StatTransfer to convert "component" table in the MS-Access file to a SAS file. The data has soil characteristics for each soil type, such as slope gradient (slope-r), soil loss by wind (wei), elevation, soil taxonomy code, etc. for each soil type.
5. We calculate average soil characteristics for each quarter-quarter section by merging data sets from steps 3 and 4. Note that the data set from step 3 has soil type distribution for each quarter-quarter section and the data set from step 4 has characteristics for each soil type.
6. We merge data sets from steps 3 and 5. We obtain a data set containing XY coordinates (latitude and longitude), average planting percentage for each crop, and a list of average soil characteristics for each quarter-quarter section. We convert this data set to a Stata data set and name it "ndqqco.dta".

Base.dta

7. We create base.dta by merging ndqqco.dta with township_section_link.dta. township_section_link.dta has townshipid for each section. We obtain this file by intersecting township map with section map in ArcGIS.

Actualq_fitted_on_fakeqs.dta

8. (fakeqq_data.do, fakeq_id.do) fakeqq_data.do generates fakeqq_data.dta using ndqqco.dta, county_admin.dta, cnty_parcel_qq.dta, and section_ij.dta. fakeq_id.do generates fakeq_id.dta using fakeqq_data.dta.

9. Actualq_fitted_on_fakesq.do generates actualq_fitted_on_fakeqs.dta.

"Extractor" ArcGIS program

This program extracts data on planted crop, soil, section, X-Y coordinate from GIS data files.

1. The whole Extractor folder has to be placed in c:\.
2. We used ArcGIS 9.2. (The program does not run on ArcGIS 10.)
3. You can run the code by double clicking the tool in ArcCatalog to process one county, or by creating a model for processing multiple counties at one time.
4. In order to run the code, you need to supply the county fips code, soil data, section data. The soil data are original "soilmu_a_xxxxx" shape files in the SoilSurvey data, and the section data are from http://134.129.78.3/geospatial/gisdata.htm for ND.
5. Before you run the code, delete all files in Scratch and Output folder.
6. If the code crashes, open Extractor.py in c:\Extractor\Tools and increase NPart in parameter section. (Its default value is 2.) This will partition a county into smaller pieces.
7. After ArcGIS finishes its job, run ConvertOutputtoSAS program in SAS. Change county fips code at the beginning and it will create output SAS file at root folder (C:\).
8. Use the following Coordinate System if you add the points to the arcview: WGS 1984 (Geographic Coordinate Systems -> World -> WGS 1984.prj)
9. The points are generated from the CDL data clipped to county level, so each point will be at the center of each cell of CDL data clipped at county level, but not of the original state level CDL data.
10. Refer to NDCountyName.EMF and NDCountyFIPS.EMF to look up county name and fips code for ND.