Remote Sensing and Machine Learning to Study the Effect of Land Titling on Land Use in Mexico

David McLaughlin (joint work with Alain de Janvry, Marco Gonzalez-Navarro, Daley Kutzman, and Elisabeth Sadoulet)

March 30, 2019

Motivation and Background

- In many developing countries property rights over rural land are maintained through continuous personal use instead of by land titles.
- The establishment and enforcement of complete property rights over land arguably is one of the most effective efficiency enhancing policies in agriculture.
- Land ownership encourages allocative efficiency which can lead to higher agricultural yields, improve access to credit, and increase the efficiency of both land markets and natural resource management.
- **Procede** (Programa de Certificación de Derechos Ejidales y Titulación de Solares), Mexico's second reform of property rights, occurred between 1993-2007 and orchestrated the transition from "use-based" rights scheme to a formal land titling system for 3.5 million people.

Mexico's Eijidos

Figure 1: Eijidos (land owned by the state from 1917-1993)

Rollout of Procede

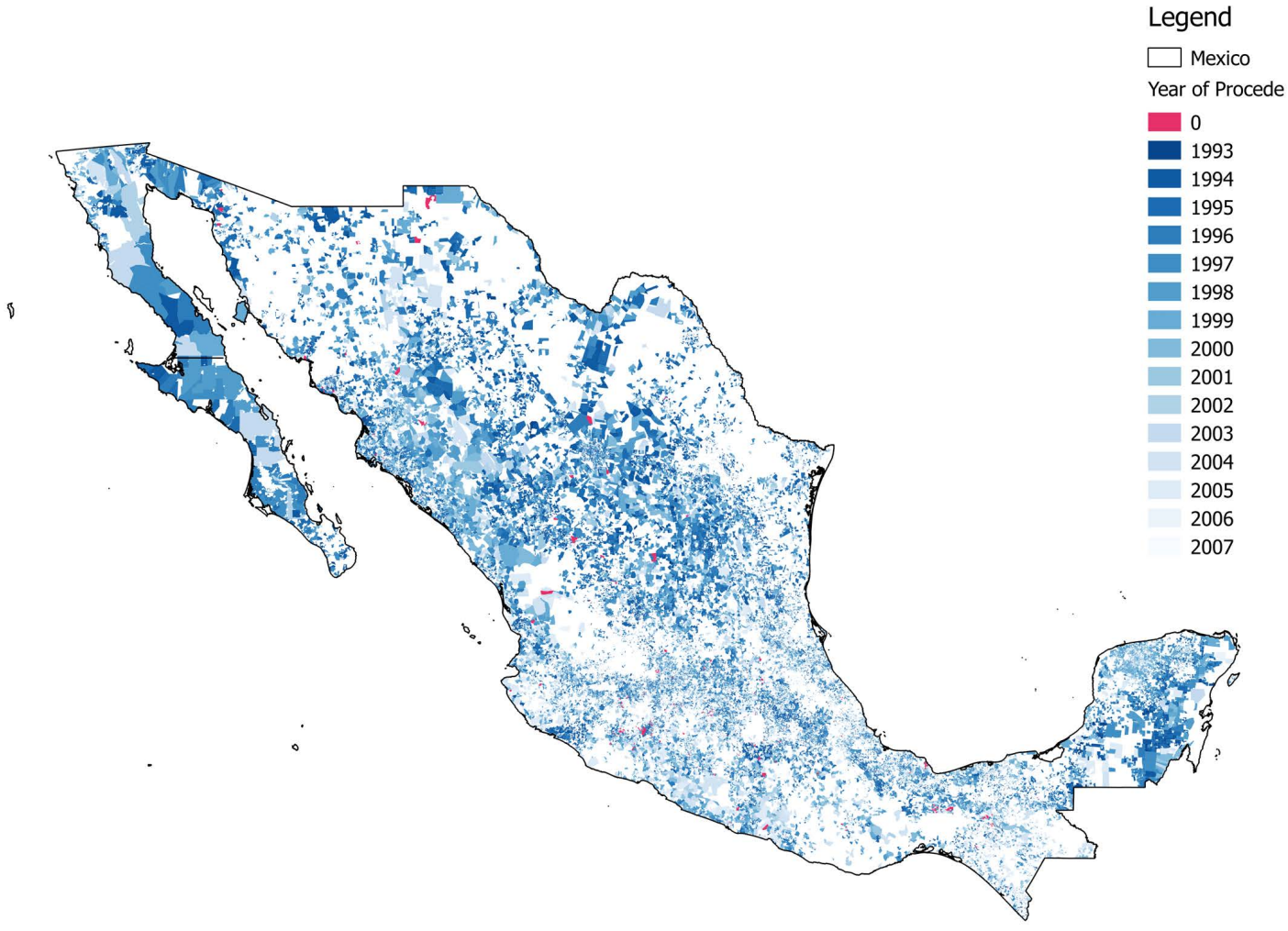


Figure 2: Rollout of Procede from 1993-2007

Research Questions

- How does the provision of property rights affect land use?
- Does deforestation increase, decrease, or remain the same following Procede?
- Do agriculture or pastureland uses contract?
- How do eijido land-use changes compare to that of private lands?
- Do we observe heterogeneity with respect to:
  - High or low levels of boundary disputes?
  - Paved roads?
  - Number of land users?

Why is this question interesting to empirical economists?

- Land is often the largest asset an household owns and its primary source of income.
- Changes in the establishment and enforcement of property rights for land are not often seen on this scale in a way which allows for quasi-experimental methods.
- Procede offers us a unique look as to the long-run economic and environmental changes brought by a reformation and strengthening of property rights.
- Household decision making on land use decisions outside of surveys and census data is difficult to measure.

What will be the main contributions from this paper?

- This paper will be the first to evaluate the effect of Procede on high-resolution land use in Mexico.
- This paper will be the first to use a novel panel of land use from 1985-2011 created using machine learning, satellite imagery, and remote sensing with Mexico's INEGI (Institute of Geography) Land Use Series II, III, and IV.
- This paper will provide environmental and development economists with a code template in Javascript and Python to use remotely sensed images to predict land use in Google Earth Engine.

Plan for today

- This is a work in progress, and we don't have final results to cover today.
- Today's plan is to cover the following:
  - Context
  - Empirical Approach
  - Identification Issue and why we need for ML, satellite imagery, in this setting
  - Land-use classification walk-through
  - Next steps
  - Conclude

Context

- Before Procede, eijidos were farmed by citizens under a "use-based" rights scheme and managed by the state.
- The state would seize land if left uncultivated for 2 years.
- In the 1990's, the eijidos and Mexico's agricultural sector were characterized by low investment and low productivity (de Janvry et. al., 1997; Deininger et. al., 2001).
- Procede provided land titles to 3.5 million people over a 10 year period, decoupling people from the land.
- Previous work by de Janvry et al. (2015) find that Procede led to increased migration.
- We will use a difference-in-differences design to exploit the timing of Procede and recover the causal effect of land titling on land use.

Empirical Approach (The regression we'd like to run)

- Difference-in-Differences

$$Y_{it} = \beta \times \mathbb{1}[\text{after Procede}] + \gamma_i + \delta_t + \epsilon_{it}$$

- Event-study specification

$$Y_{it} = \sum_{pre=\underline{t}}^{-2} \beta_{pre} \times \mathbb{1}[\text{year - procede year} = \text{t}] + \sum_{post=0}^{\bar{t}} \beta_{post} \times \mathbb{1}[\text{year - procede year} = \text{t}] + \gamma_i + \delta_t + \epsilon_{it}$$

- Y_{it} is land use in eijido i in time t, γ_i are eijido FE, δ_t are annual FE, and ϵ_{it} is a normally distributed error term. We will cluster SE's at the eijido-level (level of treatment) and municipality-level.

Lack of Pre-Procede Data, Issues with Identification and the Need for a Machine Learning Solution

- Mexico's National Institute of Geography (INEGI) has constructed a series of validated land use classifications periodically in the last 4 decades (1985, 1994, 2002, 2007, 2011, and 2014).
- The bulk of eijidos had reformed between 1993 and 2006.
- The first year of land-use data (1985) is approximate, uses a different methodology, and is more coarse in resolution.
- The difference-in-differences design would be ideal to take advantage of the timing of land-use reforms, however the sparsity of data in the pre-period makes analyzing pre-trends difficult.
- To address this, we turn to machine learning and remote sensing in Google Earth Engine (GEE) which allows us to classify land use for Mexico from 1985-2011 at 30m resolution.

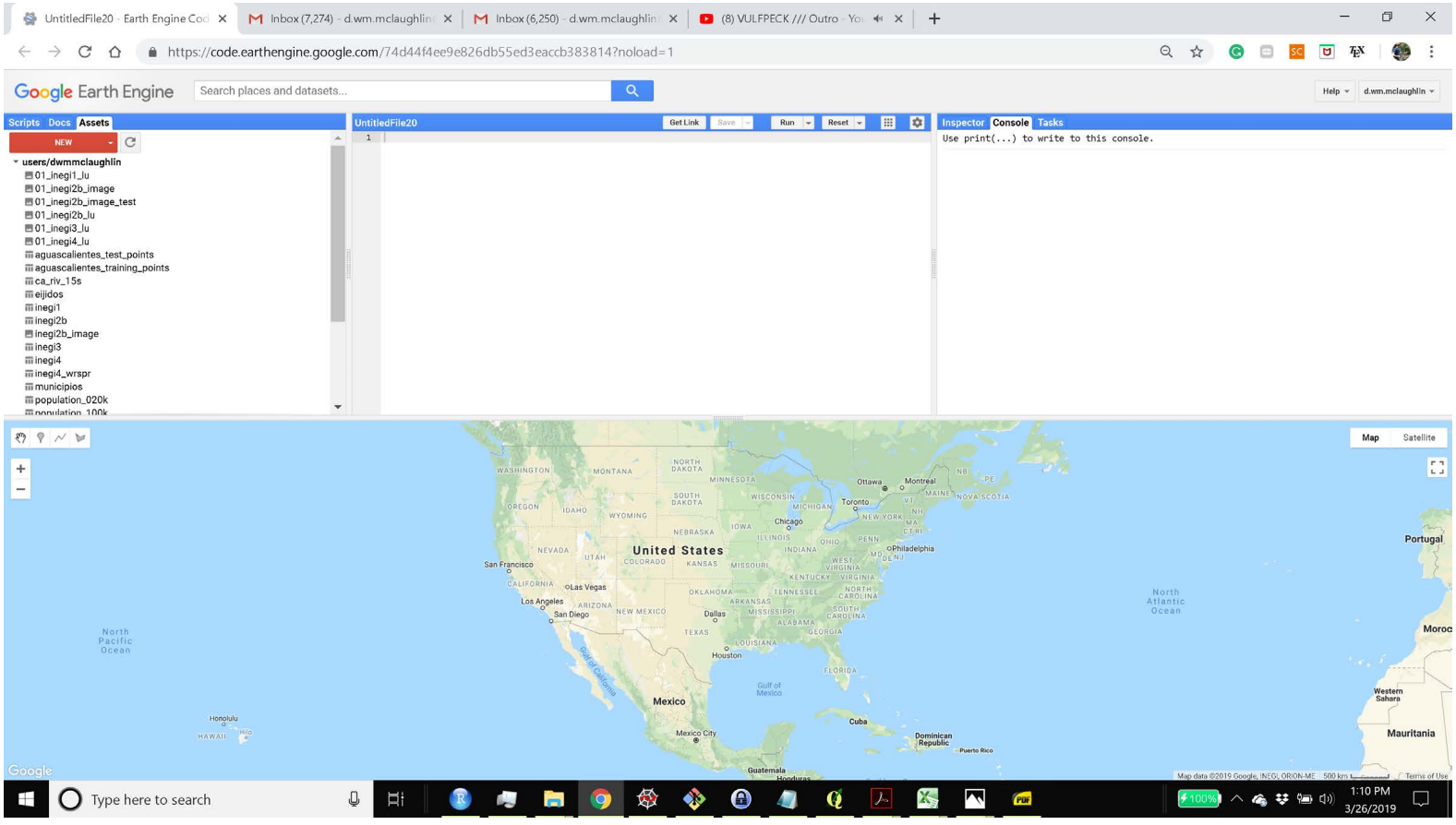
Land-use Classification Roadmap

1. Import land-use data (INEGI Series II (1994), III (2002), IV (2007))
2. Process imagery and construct training and validation datasets
3. Train and tune model (cross-validation with leave one-year-out)
4. Use model to predict land use in all years

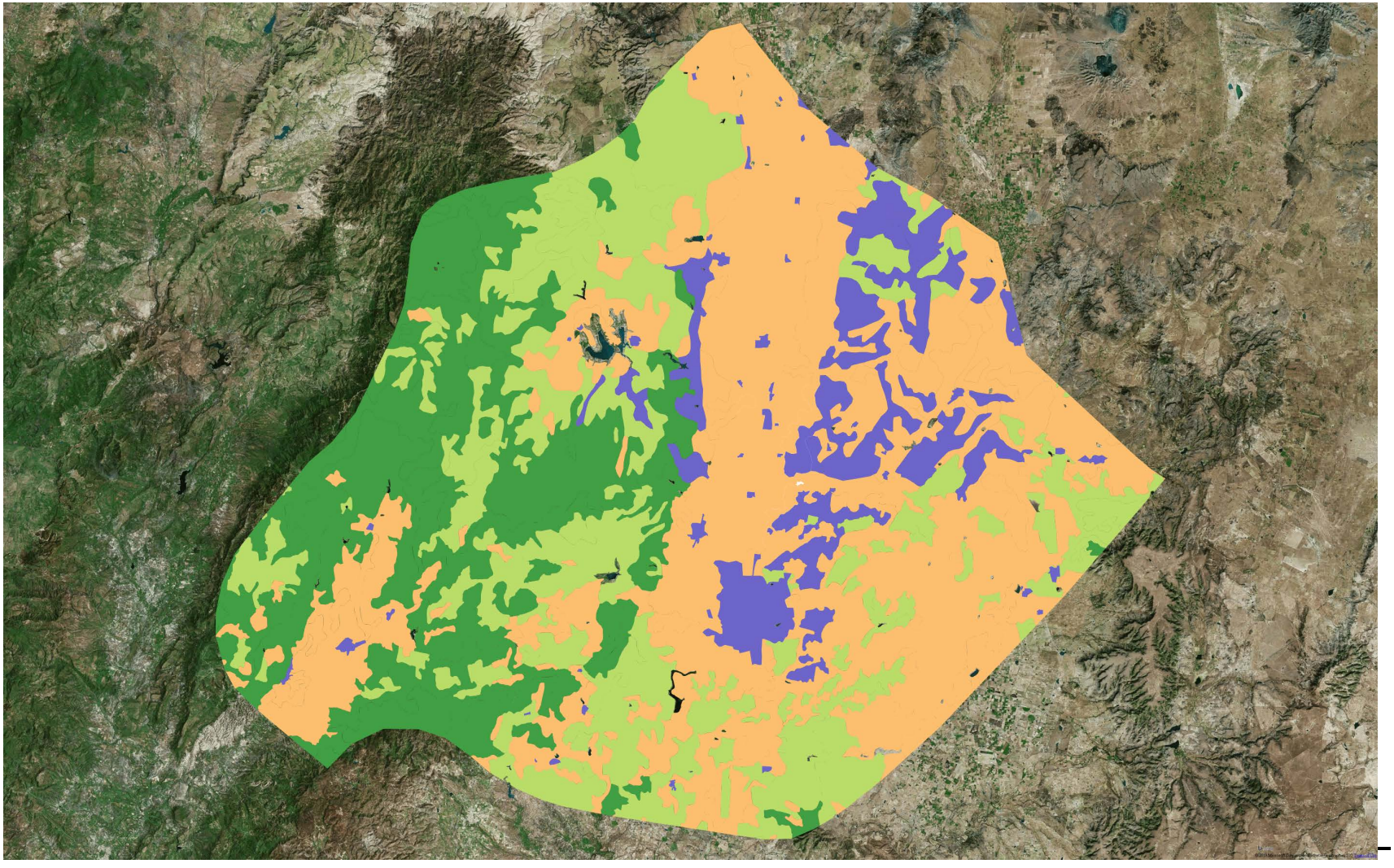
- I'll walk through these steps using the state of Aguascalientes as an example for today.

1. Import land-use data

- Ideal way is to import assets (shapefiles and raster) directly into GEE from Javascript API
- Link here `https://code.earthengine.google.com`

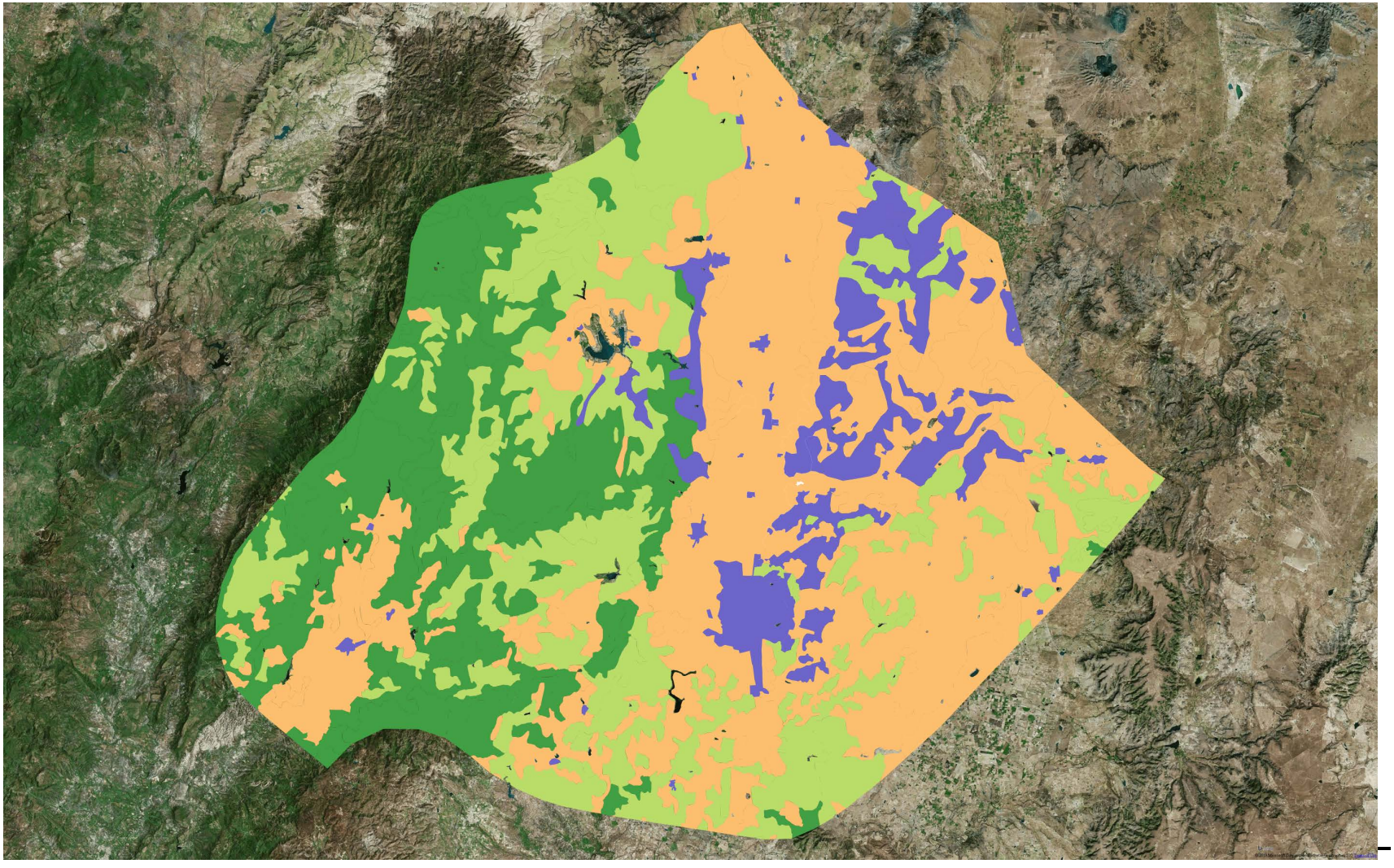


INEGI Series II for Aguascalientes (1994)



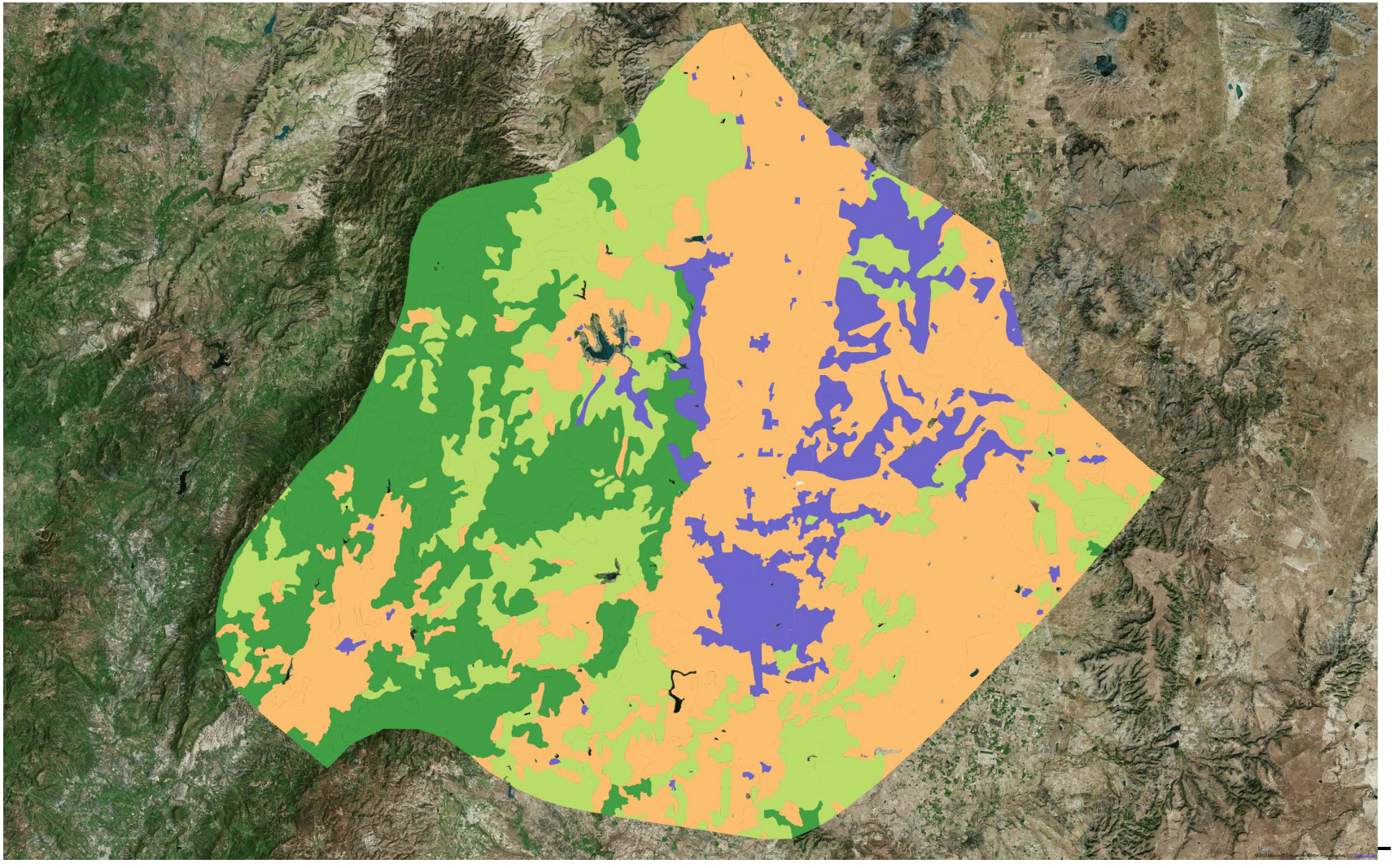
Orange is Agriculture, Dark Green is Forest, Purple is other, Light Green is Pasture

INEGI Series III Aguascalientes (2002)



Orange is Agriculture, Dark Green is Forest, Purple is other, Light Green is Pasture

INEGI Series IV Aguascalientes (2007)



Orange is Agriculture, Dark Green is Forest, Purple is other, Light Green is Pasture

2. Process INGEI landuse and imagery to construct training and vaidation datasets

- Rasterize INEGEI land use data
- Read in Landsat 5 images - orthorectified Tier 1 Surface Reflectance Product and remove clouds.
- Subset to images to years where we have data (1994, 2002, 2007)
- Obtain stratified random sample of 20 points per-image per-land-use class for training and validation and export

Rasterize INEGI landuse data

```
// Read in shapefile
var inegi = ee.FeatureCollection('users/dwmmclaughlin/inegi2b')
// Convert to image
var inegi = inegi.reduceToImage(['lu_id'], ee.Reducer.first())
// Rename band
var inegi = inegi.rename('lu_id')
var inegi = ee.Image(inegi)

// Convert to integer, set landuse id to 0,1,2,3 from 1,2,3,4
var inegi = inegi.int8()
var inegi = inegi.subtract(1)

// Export raster to drive
Export.image.toAsset({image: inegi,
  description: "01_inegi2b_lu",
  scale: 30,
  maxPixels: 1e10

});
```

Read in Landsat 5 images and remove clouds

```
var l5 = ee.ImageCollection('LANDSAT/LT05/C01/T1_SR');
// Import mexico shapefile
var mexico = ee.FeatureCollection('ft:1jgYOOLXCqEOJuAtwDIp7AA-5B9BztyNDjvm-G6cy', 'geometry');
// Select first state
var mexico_state = mexico.filter(ee.Filter.eq('state_id', 1));
// Keep only images that intersect with selected mexican state.
var l5 = l5.filterBounds(mexico_state);
// Subset images by date
var l5_date94 = l5.filterDate("1994-01-01", "1994-12-31");
// Function to mask clouds using qa band of Tier 1 SR Landsat 5 product
var maskClouds = function(image){
  // Select QA band
  var pixel_qa = image.select('pixel_qa');
  // Create shado mask
  var cloudShadowMask = pixel_qa.bitwiseAnd(8).eq(0).and(pixel_qa.bitwiseAnd(32).eq(0));
  // Mask image
  var out_image = image.mask(cloudShadowMask)
  return out_image;
};

// Map cloud mask function to image collection
var l5_date94 = l5_date94.map(maskClouds)
```

Obtain stratified random sample of 20 points per-image per-land-use class for training and validation

```
// Function to create table of training points from each image
var createTrainPoints = function(image){
  var inegi = ee.Image('users/dwmmclaughlin/01_inegi2b_lu')              // Read in land use image
  var image = image.select('B1', 'B2', 'B3', 'B4', 'B5', 'B6', 'B7')     // Subset image to B1-B7
  var inegi_mask = inegi.add(1).gte(1);                                  // Create image mask
  var image = image.updateMask(inegi_mask)                               // Mask image
  var monthBand = ee.Date(image.get('system:time_start')).get('month'); // Obtain month of image
  var image_temp = image_temp.addBands(inegi).addBands(monthBand);       // Add land use and month
  // Obtain stratified sample by land use class
  var points_out = image_temp.stratifiedSample({
    numPoints: 20,
    classBand: 'lu_id',
    dropNulls: true
  })
  return points_out;
};

// Map training poitns function to image collection
var train94 = l5_date94.map(createTrainPoints);

// Export training points to Google Drive
Export.table.toDrive({
  collection: train94,
  description: "landsat_data",
  fileFormat: 'CSV'
  });
```

3. Train and tune model

- It's easier to export the training and validation to R to train and tune hyper-parameters.
- We decided on the Random Forest model (Brieman and Cutler, 2001):
  - The Random Forest is a bootstrap aggregation algorithm which builds a large collection of de-correlated decision trees and averages them (majority vote) to create a prediction (classification).
  - Choose the number of trees, the number of variables each tree uses for splits, a method for choosing the best variable to split on, the splitting rule, and the stopping rule.
  - Random Forests are tuned by varying the number of variables used by each tree, and predicting to a validation set.
  - Our cross-validation scheme is where the model is trained on 90% of **images** and validated using 10% of **images** for K folds (in our case 10).
- The `caret` package authored by Max Kuhn is a great machine learning library in R (and python) which has built-in CV and leave-K-time or -spatial units out.

Random Forest Algorithm for Classification

1. Draw a bootstrap sample of size N from the training data.
   1.2 Grow a random-forest tree to the bootstrapped data, by recursively repeating the following steps for each node of the tree:
      1.2.1 Select m variables at random from the p variables.
      1.2.2 Pick the best variable split-point among the m (entropy or Gini).
      1.2.3 Split the node into two daughter nodes.
2. Output the ensemble of trees.
3. To classify, let each random forest tree vote. Then, the prediction is the majority vote of the ensemble.

- Paraphrased from Hastie, Tibishirani, and Freidman. *Elements of Statistical Learning* (2017), 12th ed.

Training model

- Train, tune, and select model in R using caret package

```r
libary(caret)
# Read in training data
landsat_data <- read.csv("./file_path/landsat_data.csv")
# Create tuning grid
tgrid <- expand.grid(
  .mtry = 1:9,
  .splitrule = "gini",
  .min.node.size = 1)
# Create k-folds for cross-validation by image
indices <- CreateSpacetimeFolds(landsat_data,spacevar="image", k=4)
# Train model
model_caret <- train(lu_type ~ .,
                     data = landsat_data[,list(lu_type, B1, B2,
                                               B3, B4, B5, B6, B7,
                                               month, elevation, slope, satellite)],
                     method = "ranger",
                     trControl = trainControl(method="cv", index=indices$index),
                     tuneGrid = tgrid,
                     sample.fraction = .50,
                     num.trees = t1000)
```

- Train selected model in GEE

```javascript
// Set up model in GEE (trees, mtry, min obs in leaf, left out fraction, , seed)
var rf = ee.Classifier.randomForest(1000, 9, 1, .5, false, 1982);
// Train model
var classifier_01 = rf.train(train_all, 'lu_id');
```

Results from Cross Validation

- Cross-validated accuracy is 62%.
- Test accuracy on left out images is 75%.
  - Agriculture accuracy is 71%.
  - Forest accuracy is 95%.
  - Other accuracy is 60%.
  - Pasture accuracy is 75%.
- This is definitely not perfect, and we may be able to improve upon the model.
- Let's look at the predictions for Aguascalientes.

INEGI Series II for Aguascalientes (1994)

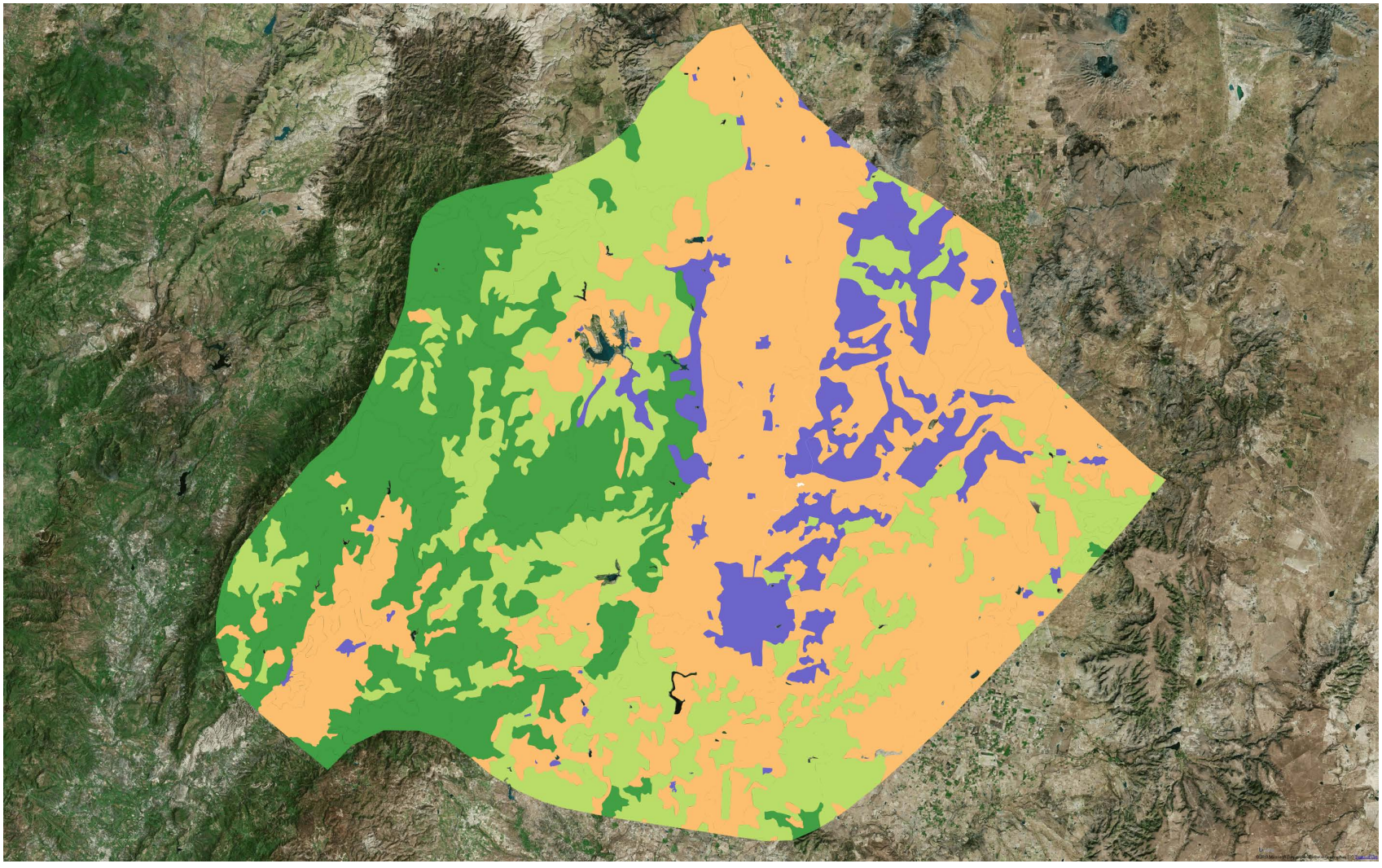


Figure 3: Aguascalientes 1992

4. Use model to predict land use in all years (1985)

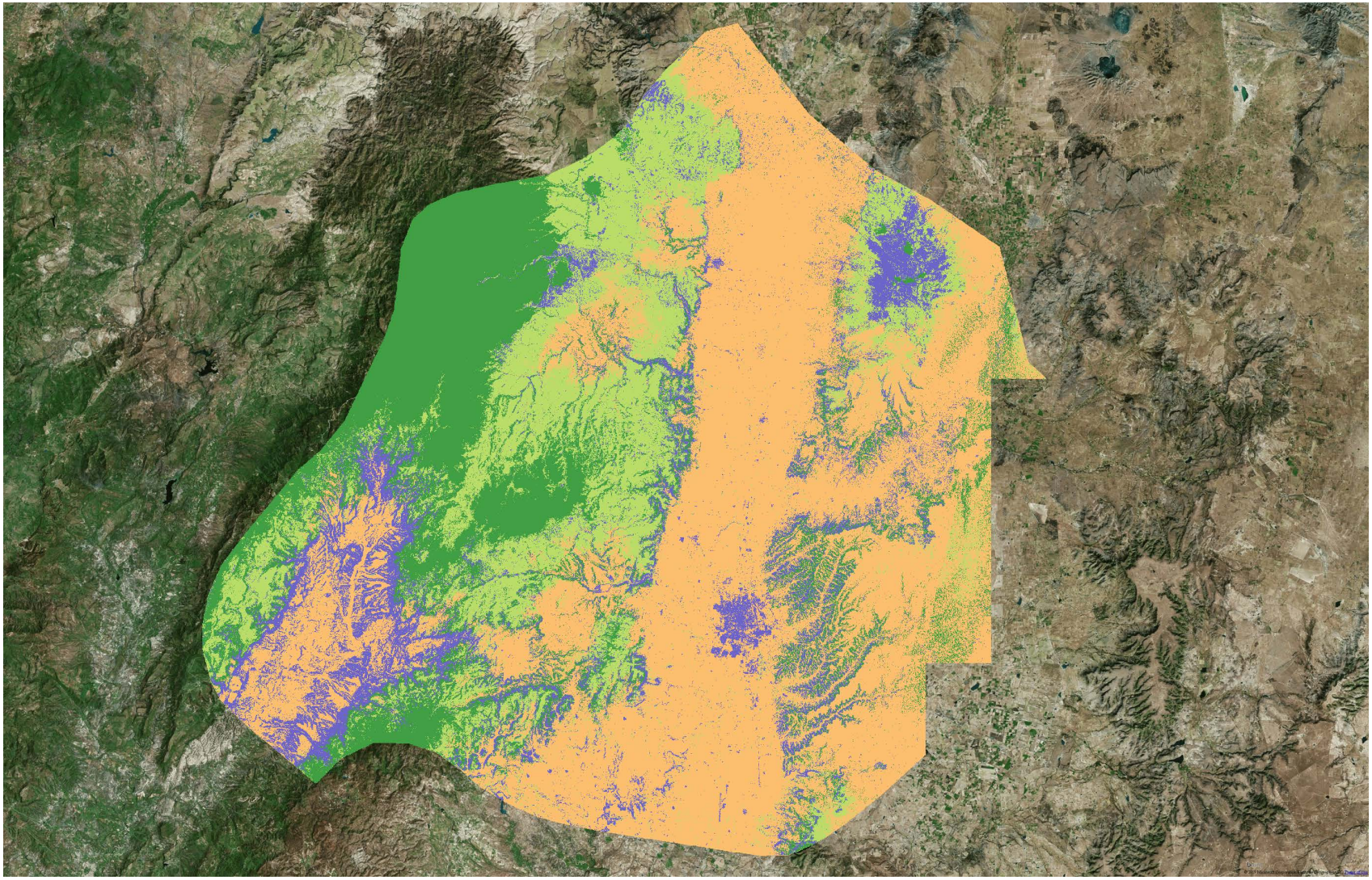


Figure 4: 1985 Prediction

4. Use model to predict land use in all years (1986)

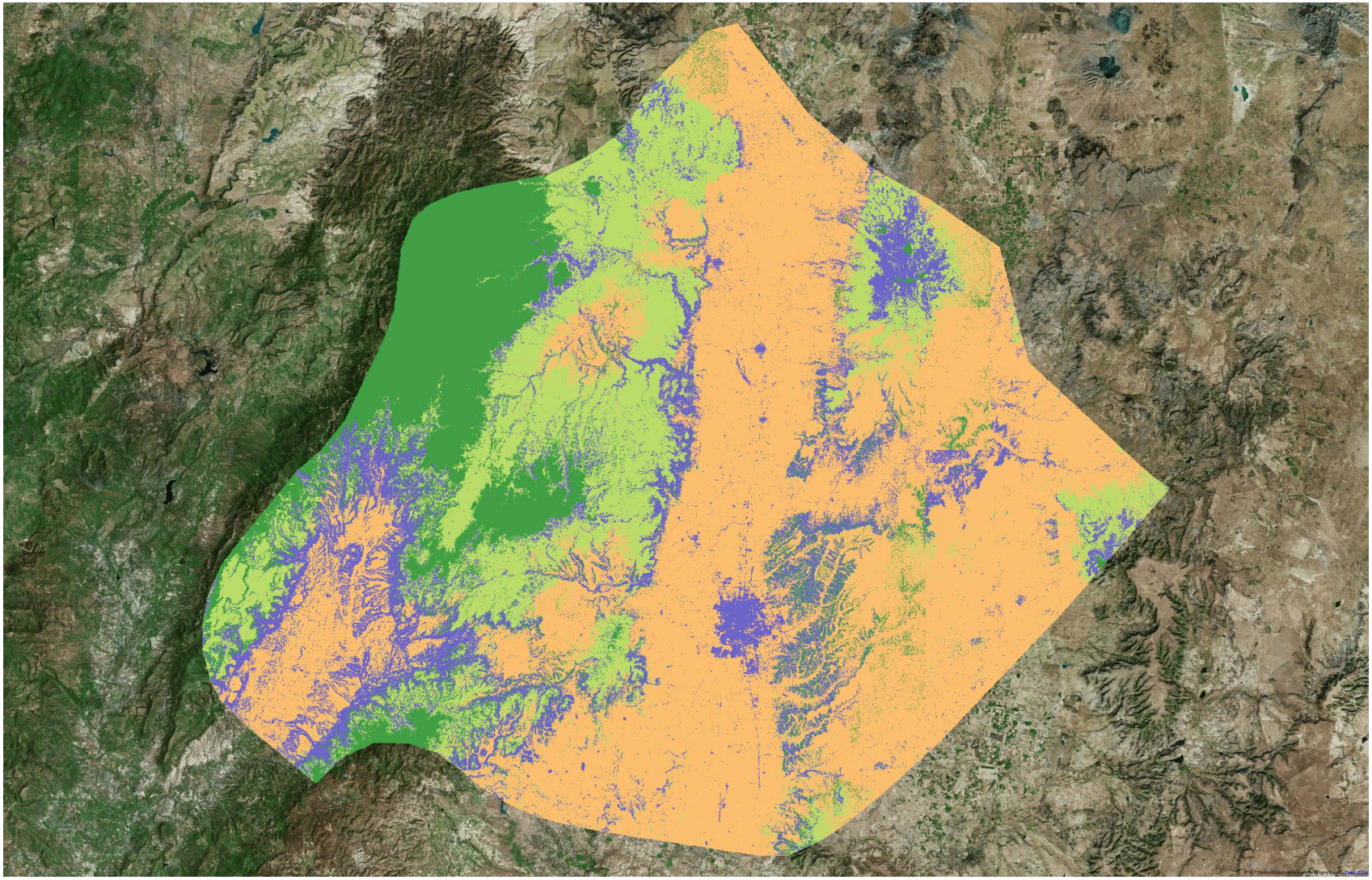


Figure 5: 1986 Prediction

4. Use model to predict land use in all years (1991)

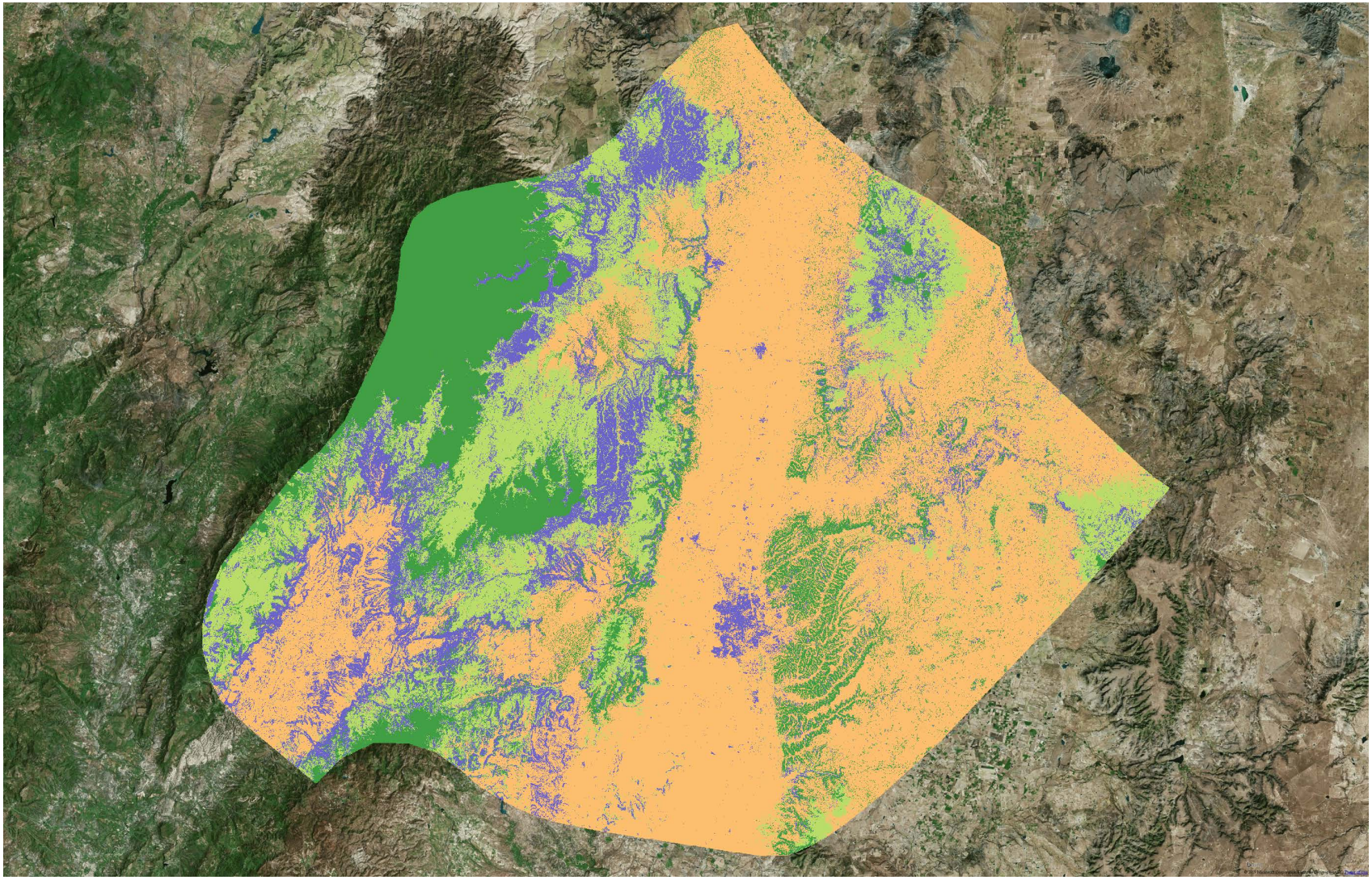


Figure 6: 1991 Prediction

4. Use model to predict land use in all years (1992)

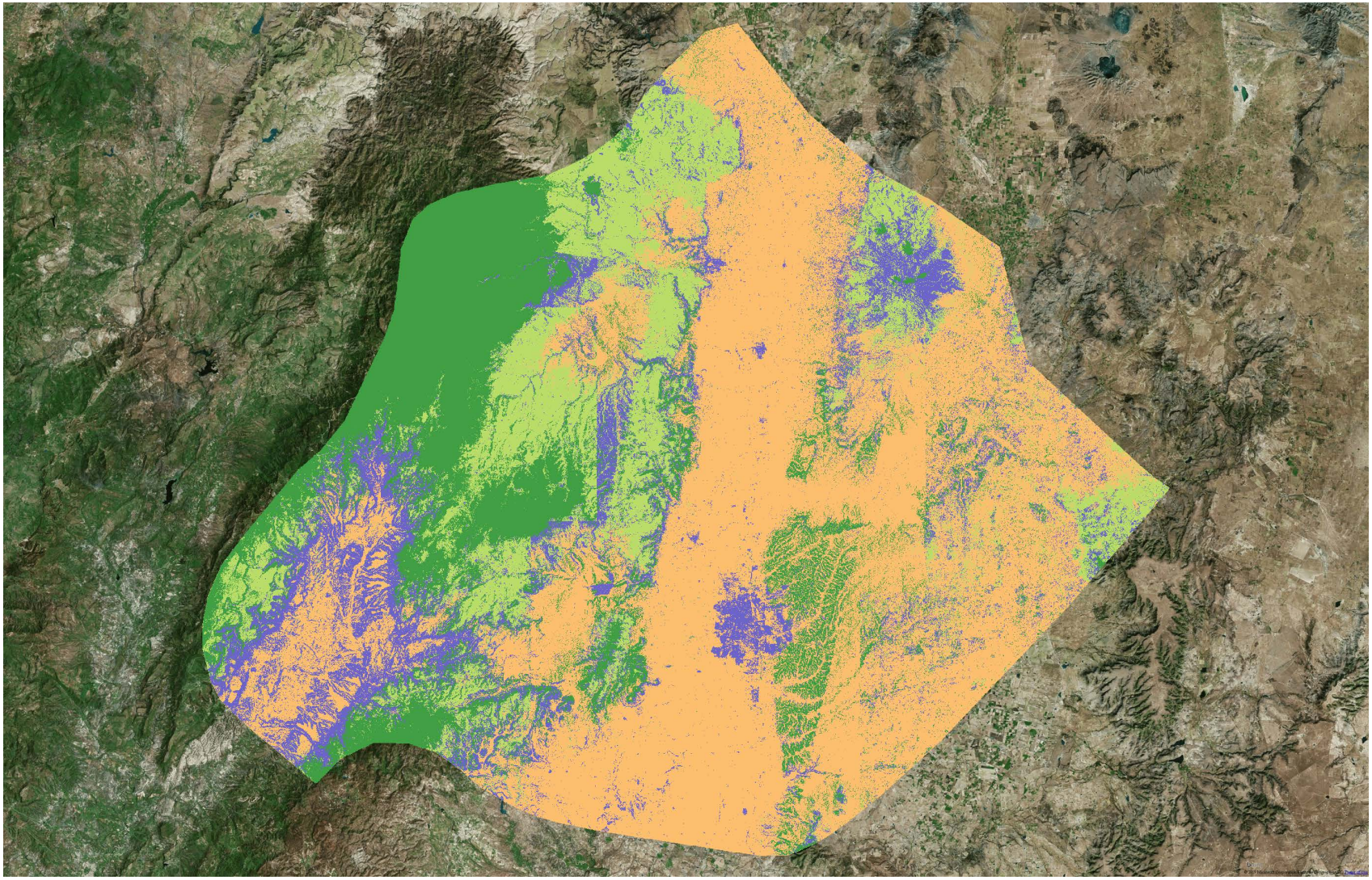


Figure 7: 1992 Prediction

4. Use model to predict land use in all years (1993)

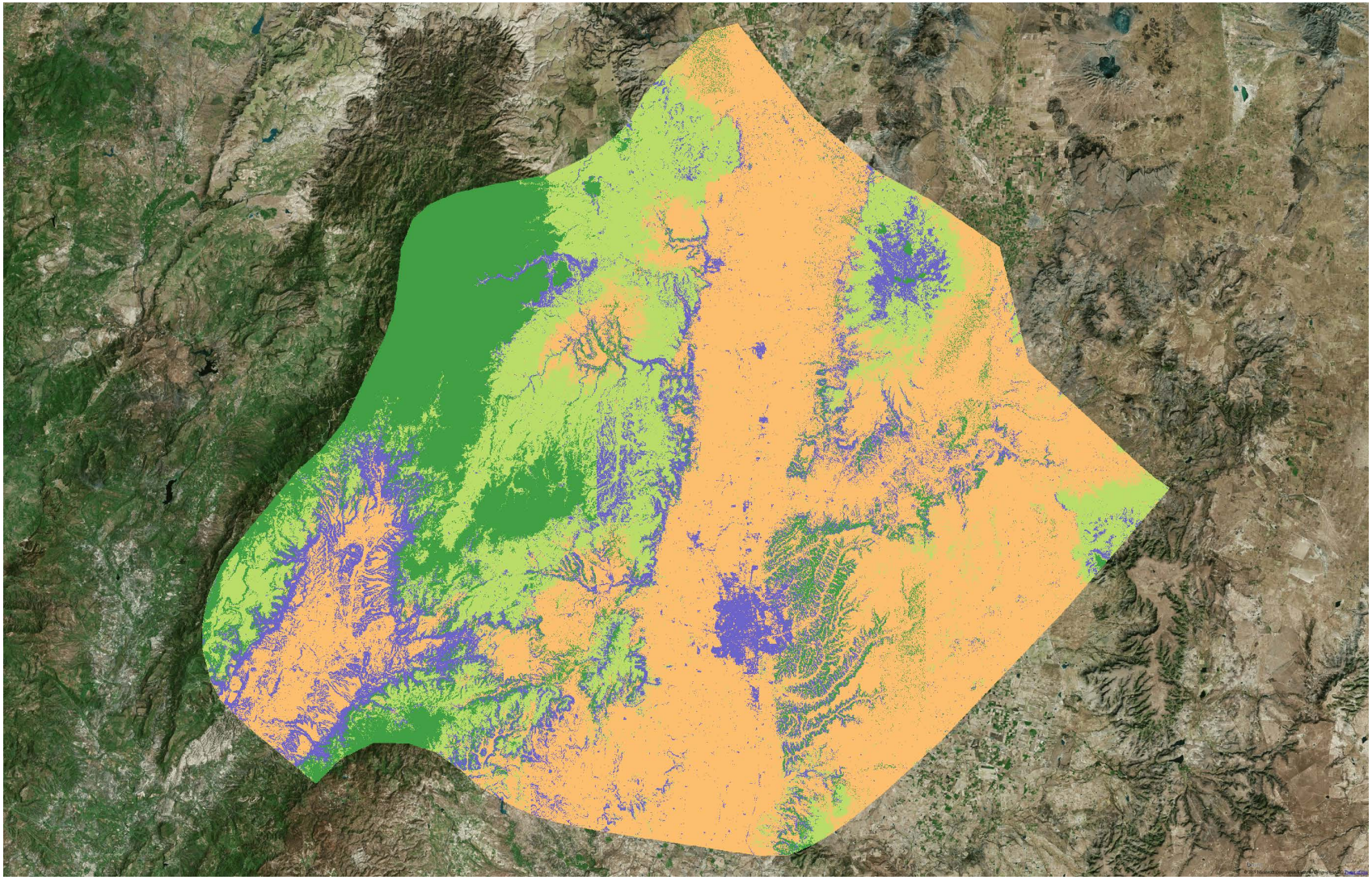
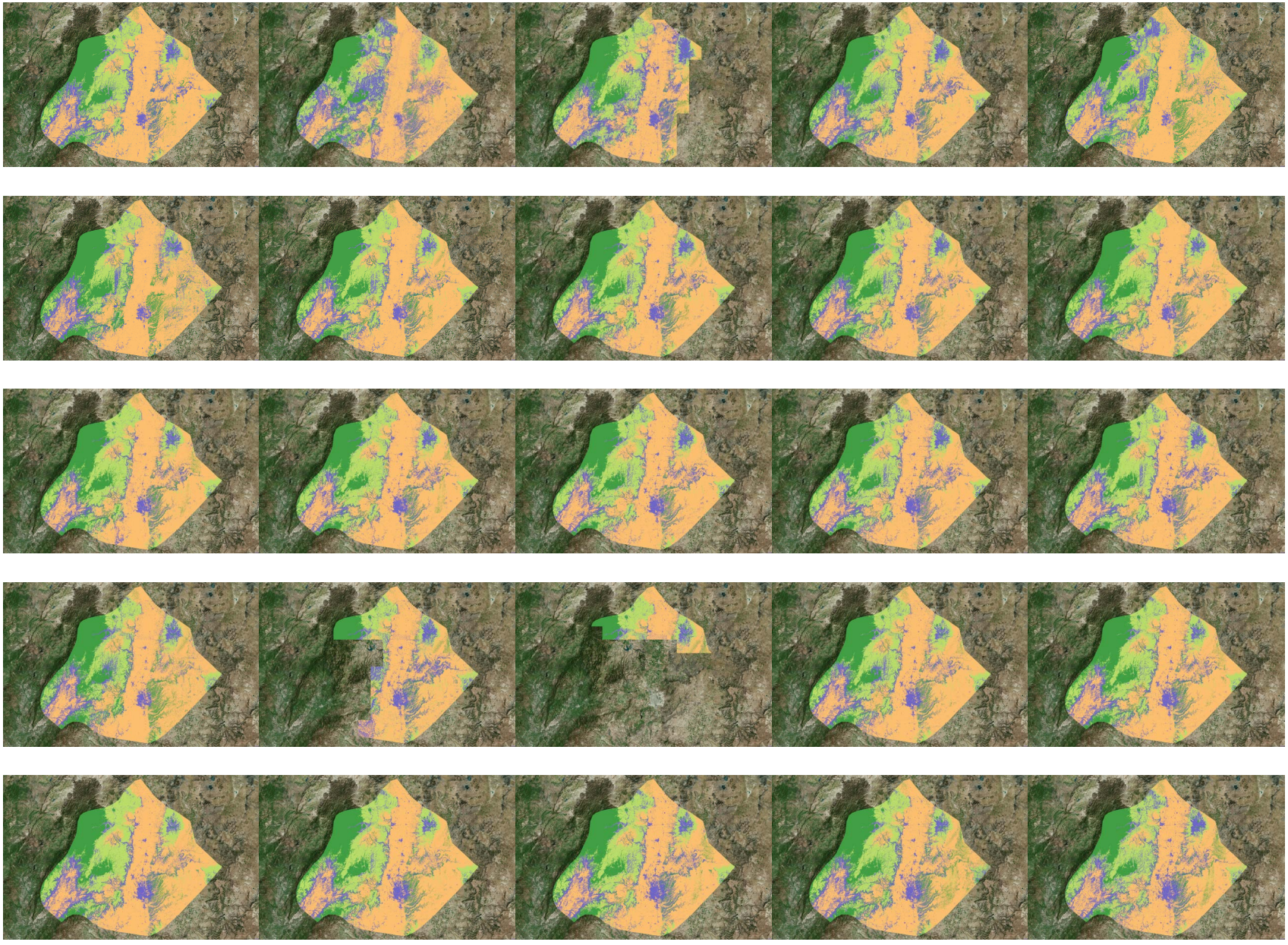


Figure 8: 1993 Prediction

4. Use model to predict land use in all years (1986-2011)

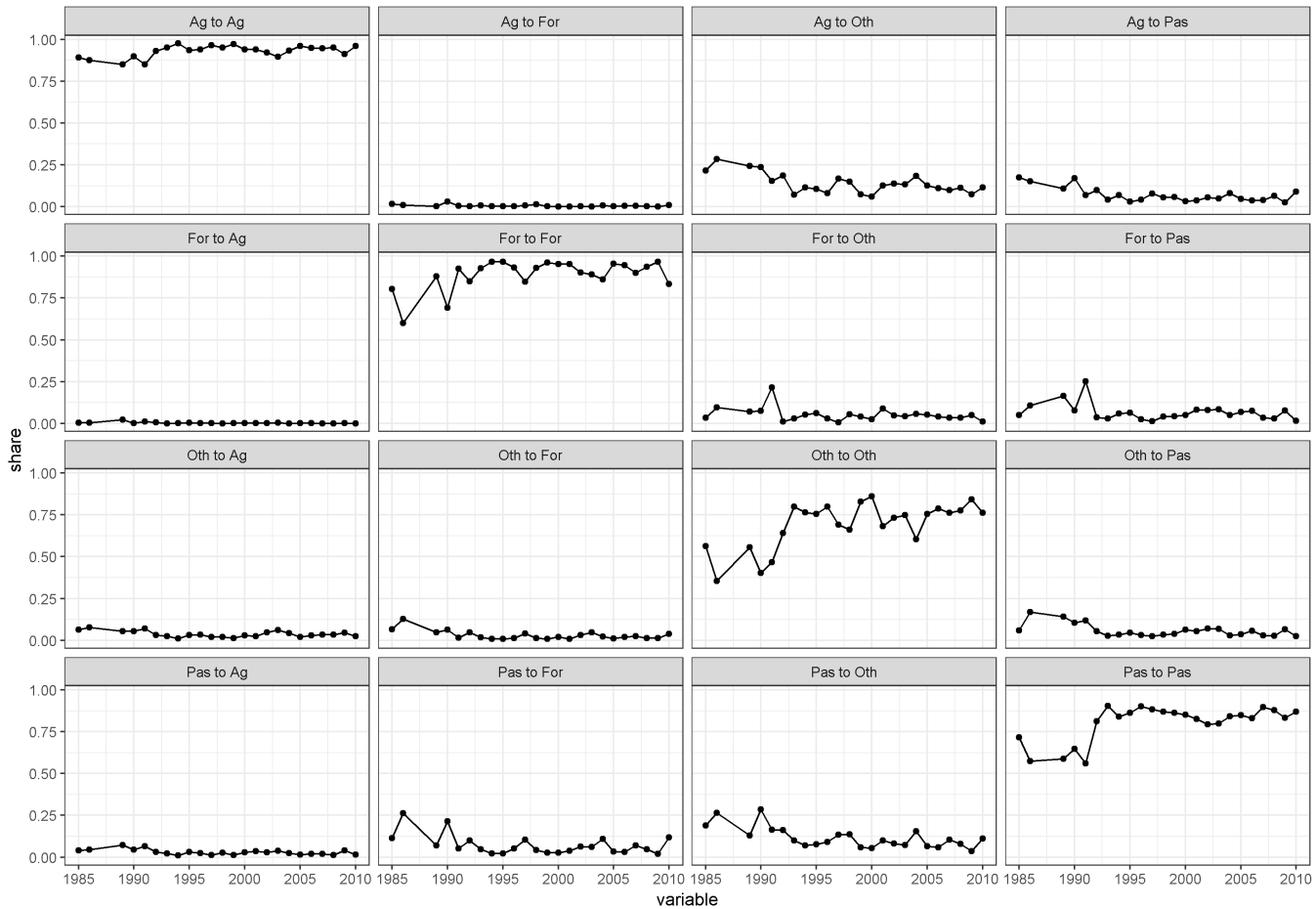


Potential issues and observations

- INEGI training data is not perfect, so there could be a ceiling on accuracy.
- Image availability is often limited.
- Image quality (e.g. incomplete cloud removal, data corruption) reduces prediction quality.
- The algorithm picks up:
  - Subtle changes in city extent and development;
  - Forests on either side of winding rivers;
  - Cliffs and mountain sides;
  - Trees and grasses in INEGI classified agricultural land.
- Examine transitions since prior is that land use is slow to change

Land-use transitions for Aguascalientes

- Tracks the 30m pixel transitions from one year to the next.
- Each row sums to 1.
- Diagonals should be smooth and high based on our prior that land use changes are gradual.



Next steps

- Complete final version of predictions.
- Add eijido characteristics.
- Run staggered difference-in-difference model.
- Heterogeneity and robustness.
- Write up results.

Should I start doing remote sensing using Google Earth Engine?

- Start with a good question, an economic model, and theory.
- Look for other sources of data before you sink time into learning GEE.
- If data are sparse or unsatisfying and satellite imagery could help you answer your question, and then GEE is the way to go.

How do I get started using Google Earth Engine?

- Sign up here: `https://signup.earthengine.google.com/#!/`
- Tutorials here: `https://developers.google.com/earth-engine/tutorials`
- Active developers community here: `https://groups.google.com/forum/#!forum/google-earth-engine-developers`
- Noel Gorelick is GEE's creator and an active contributor to the developer Google Group)

Thanks for your time

Any questions?

Email: d.wm.mclaughlin@berkeley.edu