

Four Forest Restoration Initiative - Landscape Pattern Analysis

Luke J. Zachmann and Brett G. Dickson

April 28, 2017

A summary of data products created for the US Forest Service, including links to the data and web application used to render deliverables, metadata regarding the accuracy and precision of specific results, and guidance for appropriate uses of the data.

- Introduction
- Methods and results
 - Project area
 - Imagery
 - Model building
 - Training data development
 - Predictor variable development
 - Data aggregation
 - Feature selection and tuning for optimal model hyperparameters
 - Classification
 - Model evaluation
 - Spatial metrics
 - Corrections for the effects of shadows
 - Links to code and data
 - Usage
- Acknowledgements
- References

Introduction

The Four Forest Restoration Initiative (4FRI) is a landscape-scale project focused on restoring Ponderosa pine forests in Arizona. An important part of the successful implementation of this project is to assess the impacts of restoration treatments on forest structure. Northern Arizona University (NAU) and the United States Department of Agriculture (USDA) Forest Service collaborated to quantify and describe the amount, pattern, and distribution of canopy cover within and around Ponderosa pine forests of the South Kaibab and Coconino National Forests.

Methods and results

The architecture of this analysis contains many of the building blocks found in most predictive modeling applications: gathering and preparation of data, feature selection and model tuning, model evaluation/validation, and post hoc summaries. Each of these components are described in more detail below.

Project area

The project area encompasses 1,224,900 acres and intersects portions of both Coconino and Kaibab National Forests (Figure 1).

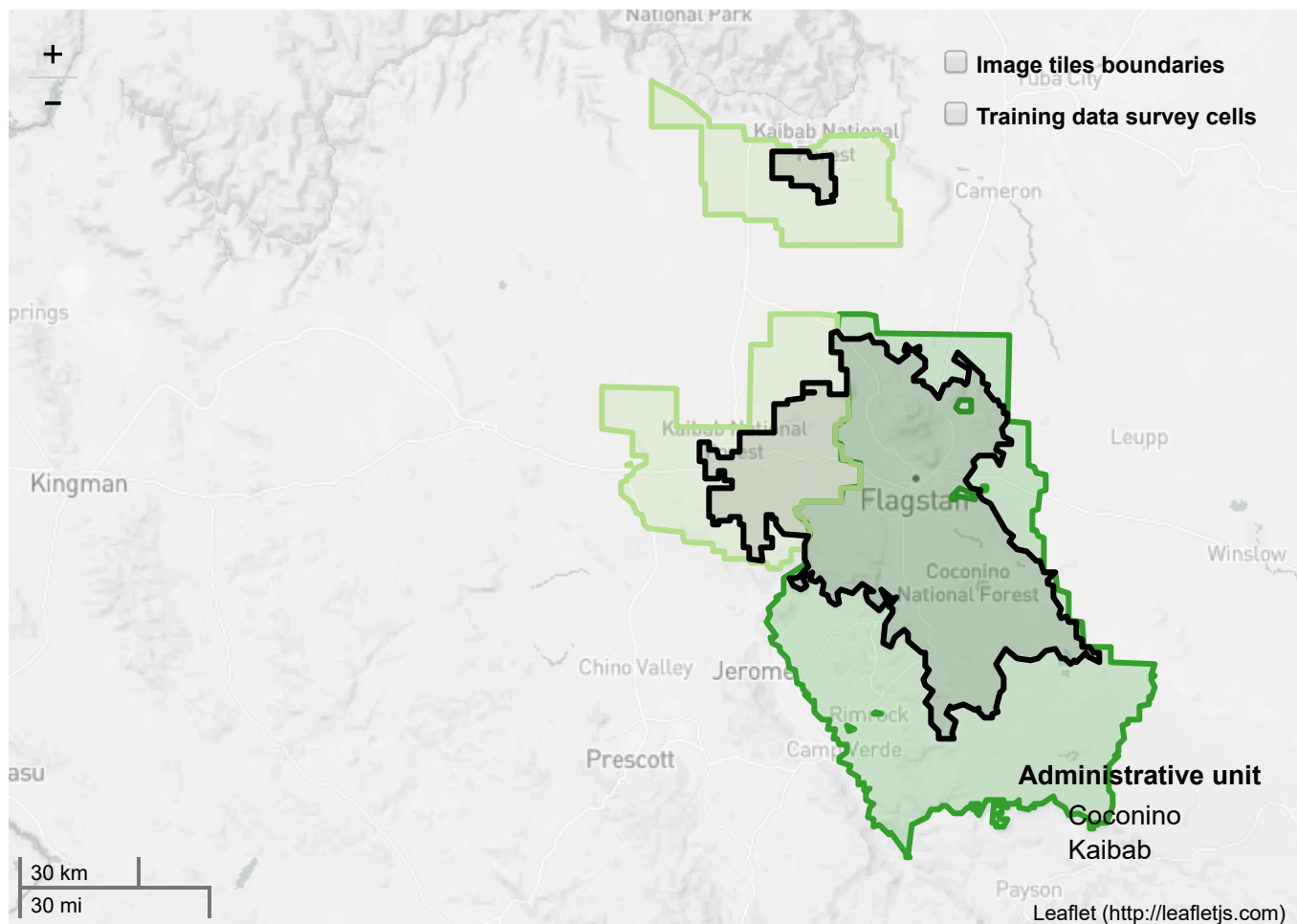


Figure 1: The project area / orthoimagery acquisition boundary in relation to the administrative boundaries of Coconino National Forest and the South Kaibab (Tusayan and Williams Ranger Districts). Selectable layers include image tile boundaries ($n = 619$) and the locations of the spatially balanced random survey cells used to develop training data for the canopy cover classification model ($n = 158$, ~300-acre cells). All told, 6,119 samples were used to train the model, 40% of which were collected completely at random (from cells in blue) and 50% of which were collected opportunistically (from cells in red). The remaining 10% of samples were collected completely opportunistically outside of the spatially balanced survey cells.

Imagery

High quality aerial imagery was acquired for the project area between June 6 and June 23, 2014 by the USDA Farm Service Agency Aerial Photography Field Office. The acquisition platform was a light aircraft flying ~5,570 m above ground level to achieve a nominal resolution of 0.3 m. The 4-band — red, green, blue, and near infrared (hereafter, 'NIR') — imagery was collected using a Microsoft UltraCam Eagle sensor with a 100.5 mm focal length. The images were orthorectified, mosaicked, and radiometrically adjusted to meet contract requirements. The primary difference between this imagery and the better-known NAIP image archive is quality: i.e., how far off nadir acquisitions are allowed to be, the increased overlap between photos, a requirement for no cloud cover, and restrictions on time of day (to reduce shadows). For the purposes of this analysis, the imagery was stored as an asset (i.e., a stack of images, referred to as an `ImageCollection`) in Google Earth Engine (<https://earthengine.google.com/>) (hereafter, 'Earth Engine'). Each `Image` object within the collection represents a single Digital Orthophoto Quarter Quarter Quadrangle (DOQQQ).

Model building

We developed a 3-class (tree, non-tree, and shadow) supervised classification model. Specifically, we used a random forest classifier (Breiman 2001). The model-building process entailed several steps, each of which are described in more detail below:

1. Training data development;
2. Predictor variable development;
3. Data aggregation;
4. Feature selection and tuning for optimal model hyperparameters; and
5. Classification.

Training data development

Spatially balanced random survey cells generated using the Reversed Randomized Quadrant-Recursive Raster algorithm (RRQRR; Theobald et al. 2007) were used to develop training data for the canopy cover classification model ($n = 158$, ~300-acre cells).¹ All told, 6,119 samples were used to train the model, 40% of which were collected completely at random and 50% of which were collected opportunistically (Figure 1). The remaining 10% of samples were collected completely opportunistically outside of the spatially balanced survey cells.

Predictor variable development

A large suite of predictors were developed using the imagery as well as digital elevation data. For example, we computed the Normalized Difference Vegetation Index (https://en.wikipedia.org/wiki/Normalized_Difference_Vegetation_Index) (NDVI) from the red and NIR bands in the imagery, and topographic layers (i.e., elevation, slope, aspect) using the USGS National Elevation Dataset (NED; Farr et al. 2007). Additionally, we applied edge detection — including a difference-of-Gaussians (DOG)² — as well as methods for estimating spatial texture, including entropy

([https://en.wikipedia.org/wiki/Entropy_\(information_theory\)](https://en.wikipedia.org/wiki/Entropy_(information_theory))), a gray-level co-occurrence matrix (GLCM)³, and a local measure of spatial association (Geary's C; Anselin 1995). All estimates of spatial texture were computed using the near infrared band.

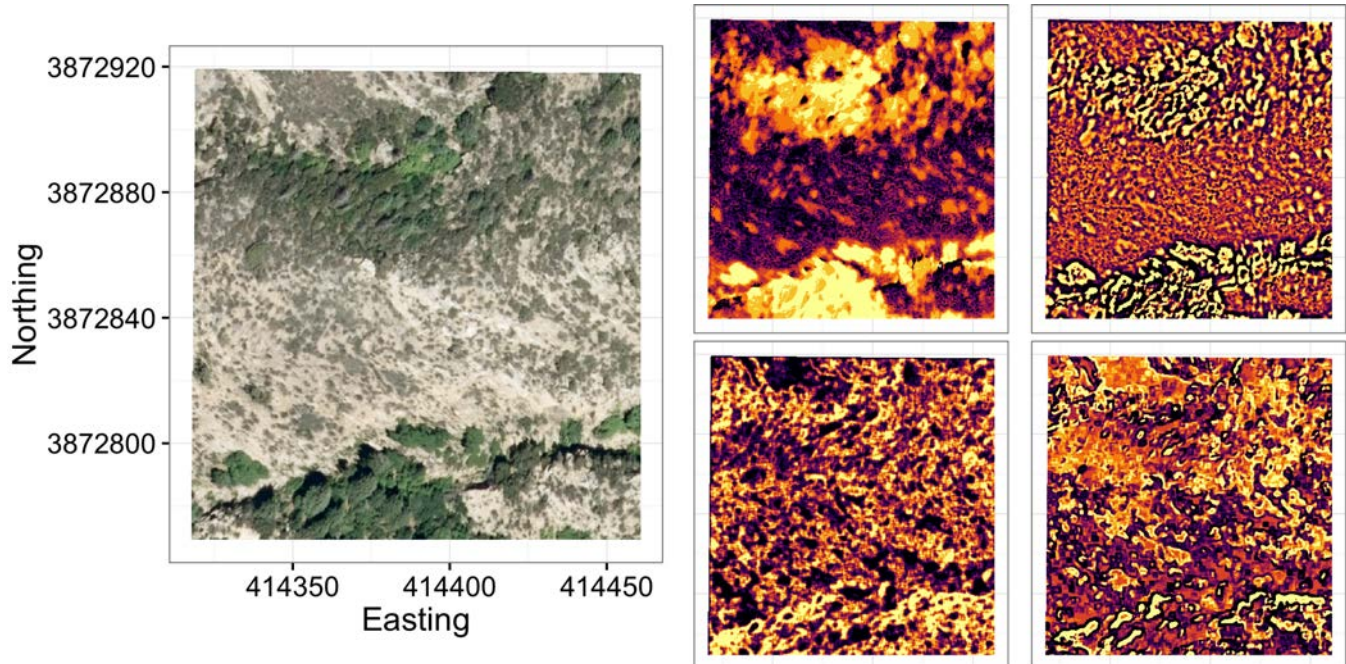


Figure 2: Examples of quantities derived using the imagery seen in true-color in the left-most panel. Predictors shown here (the singleband pseudocolor images in the right-most panel) include NDVI, DOG, entropy, and GLCM cluster shade.

Data aggregation

To extract predictor variable information to the locations of samples in the training data, we used reducers (i.e., the `ee.Reducer` class) in Earth Engine. Samples that fell in the overlapping area between DOQQQ tiles had two (or more) sets of covariate information. We did not allow these redundant copies enter the model-training step. Instead, we selected only one set of covariate information for these samples, specifically the set corresponding to the DOQQQ tile whose centroid was nearest the 'offending' sample.

Feature selection and tuning for optimal model hyperparameters

We used automatic feature selection methods to identify the attributes (predictors) that were required to build an accurate model. Approximately one-half of the variables were highly correlated (with absolute correlations of 0.90 or higher). In the interest of removing some of these highly-correlated variables, we used Recursive Feature Elimination (<https://topepo.github.io/caret/recursive-feature-elimination.html>) (De Martino et al. 2008), a backwards selection algorithm, with a target number of features of 5-30. The RFE identified 27 variables as being informative. The top 5 variables included (in order of importance) the red band, NDVI, another normalized difference based on the NIR band and NDVI, and green and blue bands.

The training samples and associated covariate information were brought into R, where we used the

`caret` (<https://github.com/topepo/caret>) package to find optimal parameters for the random forest algorithm. Specifically, we conducted a grid search of parameters, using repeated cross validation and accuracy as the performance metric, to select the optimal model.

Classification

The final, tuned model hyperparameters (i.e., the number of trees, variables per split, and minimum leaf population), were then used in the `ee.Classifier.randomForest` method in Earth Engine. We trained the model against a regionally-boosted sample for each image in the final prediction in order to better-calibrate the model against local conditions. The final canopy cover classification model can be viewed using the web application linked to in Figure 8.

Model evaluation

The performance of the classification model was evaluated against a ‘test’ partition, a set ($n = 621$) of samples selected at random from among the spatially-balanced training set and withheld from the model during tuning/training. Statistical measures of the performance of the model are provided in a confusion matrix (Table 1). Numbers along the diagonal (boldface font) represent correctly classified test samples. Overall accuracy (the sum of correctly classified samples divided by the total number of samples) was 98.4%.

Table 1: Confusion matrix for the final classification model.

	Predicted		
	Canopy	Shadow	Other
Actual			
Canopy	203	0	4
Shadow	6	201	0
Other	0	0	207

Off-diagonal elements represent different types of errors. For example, there were 4 samples that were misclassified as ‘other’ (non-tree/non-shadow) when the test data show they were actually canopy. Additional measures of the performance of the classifier for each class are reported in Table 2. For example, sensitivity measures the proportion of the actual samples in a given class that were correctly identified as such, while the positive predictive value (or precision) is the proportion of predictions in a given class that were correct. For more information regarding performance measures in Table 2, see Wikipedia (https://en.wikipedia.org/wiki/Sensitivity_and_specificity).

Table 2: Statistical measures of the performance of the model for each class. Class-wise statistics were computed using a ‘one against all’ approach.

		Positive predictive value	Negative predictive value
Sensitivity	Specificity		

	Sensitivity	Specificity	Positive predictive value	Negative predictive value
Canopy	0.981	0.986	0.971	0.990
Shadow	0.971	1.000	1.000	0.986
Other	1.000	0.990	0.981	1.000

Finally, we performed a ‘straight-face’ test (qualitative visual assessment) of the result. Though every iteration of the model we produced ultimately passed the statistical tests, we noticed that some regions within the project area were more problematic than others. For example, the craters in the NE quadrant of the study area were showing up with more area classified as tree canopy than expected, and a quick visual assessment confirmed that the model was likely confusing green grass in the understory as canopy. This is what led to the deployment of a larger opportunistic sample and the subsequent development of a geographically boosted training set.

Spatial metrics

Indicators of desired forest structural conditions (Science and Monitoring Working Group 2012) include patch size, density, and configuration. The US Forest Service selected eight composite metrics, a few of which consisted of several individual FRAGSTATS metrics, from Cushman et al. (2008) (Table 3). We used the R package `SDMTools` (<https://github.com/jjvanderwal/SDMTools>) to compute the majority of the LSMs. Some of the metrics (i.e., ENN- and GYRATE-based metrics) did not have direct analogs in `SDMTools` and, as such, were calculated ‘by hand’ following FRAGSTATS documentation (<http://www.umass.edu/landeco/research/fragstats/documents/fragstats.help.4.2.pdf>).

The landscape/classification was divided up into sublandscapes. Landscape structure metrics (LSMs) were computed for each sublandscape and subsequently mosaicked into a new raster. LSMs were developed at multiple scales (i.e., 1 and 5 acres) across the landscape.

Table 3: Horizontal forest structure (landscape pattern) metrics.

Landscape structure metric	Specific FRAGSTATS metric	FRAGSTATS acronym
Mean patch size	Mean patch area	AREA_MN
Edge contrast	Total edge contrast index ^a	TECI
Patch aggregation	Aggregation index	AI
Nearest neighbor distance	Mean nearest neighbor distance	ENN_MN
Patch shape complexity	Mean shape index	SHAPE_MN
Patch shape complexity	Mean fractal dimension index	FRAC_MN

Landscape structure metric	Specific FRAGSTATS metric	FRAGSTATS acronym
Patch shape complexity	Area-weighted mean fractal dimension index	FRAC_AM
Patch shape complexity	Fractal dimension index, coefficient of variation	FRAC_CV
Patch dispersion	Nearest neighbor distance, coefficient of variation	ENN_CV
Large patch dominance	Largest patch index	LPI
Large patch dominance	Area-weighted mean patch area	AREA_AM
Large patch dominance	Area-weighted mean core area index	CORE_AM
Large patch dominance	Area-weighted mean disjunct core area index ^b	DCORE_AM
Shape and correlation of large patches	Area-weighted mean radius of gyration	GYRATE_AM
Shape and correlation of large patches	Area-weighted mean shape index	SHAPE_AM

^a Computing TECI requires a matrix of contrast weights (the relative differences among patch types). However, these weights could not be defined, largely because the notion of contrast between canopy and the blanket class, 'other' – which includes a wide diversity of other patch types, from parking lots to water – and the contrast between canopy and shadow, is undefined. In other words, the magnitude of edge contrast for each pairwise combination of patch (class) types doesn't make sense in this application, but they may warrant consideration in future work, including future developments of this data.

^b The documentation for FRAGSTATS indicates that 'from an organism-centered perspective, a single patch may actually contain several disjunct patches of suitable interior habitat, and it may be more appropriate to consider disjunct core areas as separate patches.' Because the number and area of disjunct cores in each patch varies as a function of the specified edge depth (which would need to be defined according to the habitat requirements of a specific species), we left this particular metric out. Should applications for specific species arise, estimates of the (area-weighted) mean area per disjunct core could be generated using the canopy cover classification.

Corrections for the effects of shadows

To develop the data necessary to calibrate LSMs against the effects of shadows we:

1. simulated a large ($n = 5,000$) set of arbitrary (hypothetical) forested areas in which the number and crown widths of trees in a given area (e.g., a 1-acre window) were sampled from their respective distributions;
2. calculated the shadows that would have been cast by those canopies for a given sun angle and azimuth (both of which were drawn from the actual distributions of sun angles and azimuths present at the time the images were taken); and,
3. computed LSMs for the simulated forested area both *with* and *without* shadows.

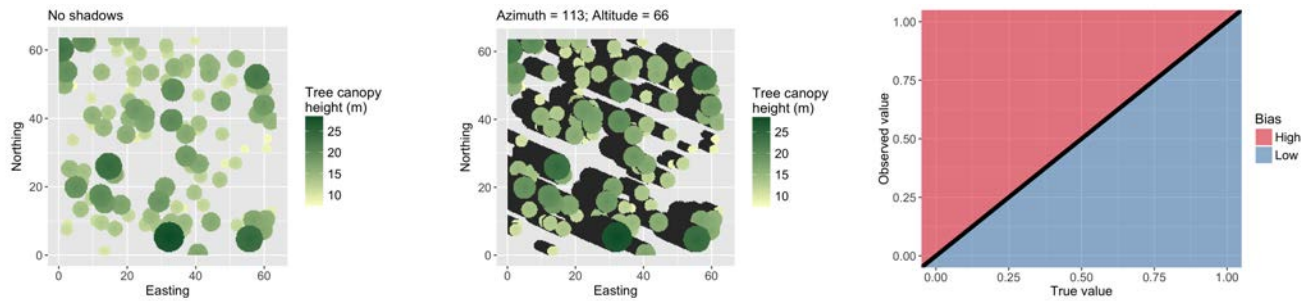


Figure 3. An example of a simulated forested area with and without (left and center panel, respectively) at a 1-acre scale of analysis. As a result of the presence of shadows (among other factors), estimates of LSMs can be biased high or low, relative to the true value (right panel).

Data from the simulations allowed us to fit a model to predict the 'true' value of each LSM, i.e., the value we would have obtained if shadows were not present. The data were modeled as

$$y_{ij} = \beta_0 + \beta_1 x_{1,ij} + \beta_2 x_{2,i} + \beta_3 x_{3,i} + \dots + \beta_m x_{m,i} + \epsilon_{ij}$$

where y_{ij} is the true value of metric i for sublandscape j . Predictors include the observed value of the metric $x_{1,ij}$, the proportion of canopy in the sublandscape, $x_{2,j}$, and the ratio of the proportion of shadow to the proportion of canopy in the sublandscape, $x_{2,j}$. Parameters $\beta_4 - m$ are tied to the two way interactions between $x_1 - 3$. We then leveraged the relationship between true and observed values to correct each metric for shadow effects. As expected, shadows have a stronger influence on some variables than others (e.g., see AI vs. FRAC_CV; Figure 4). An estimate of the degree of confidence in the calibrated LSM can be approximated by the variance of observations around the 1:1 line (Figure 5).

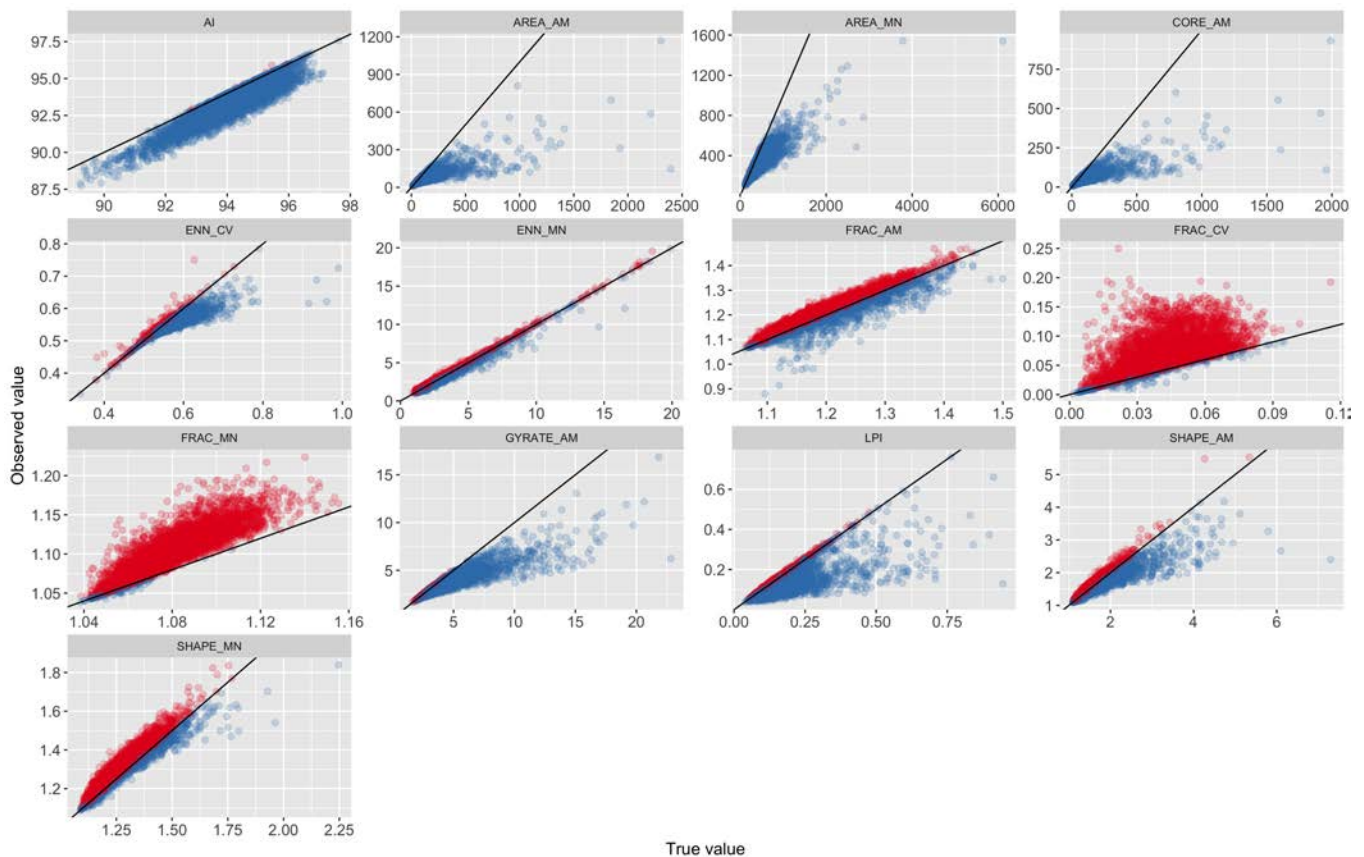


Figure 4: Scatterplots of the observed vs. true values of each LSM. Perfect estimates would fall along the 1:1 line. All others are either biased high (red) or low (blue; see the third panel in Figure 3).

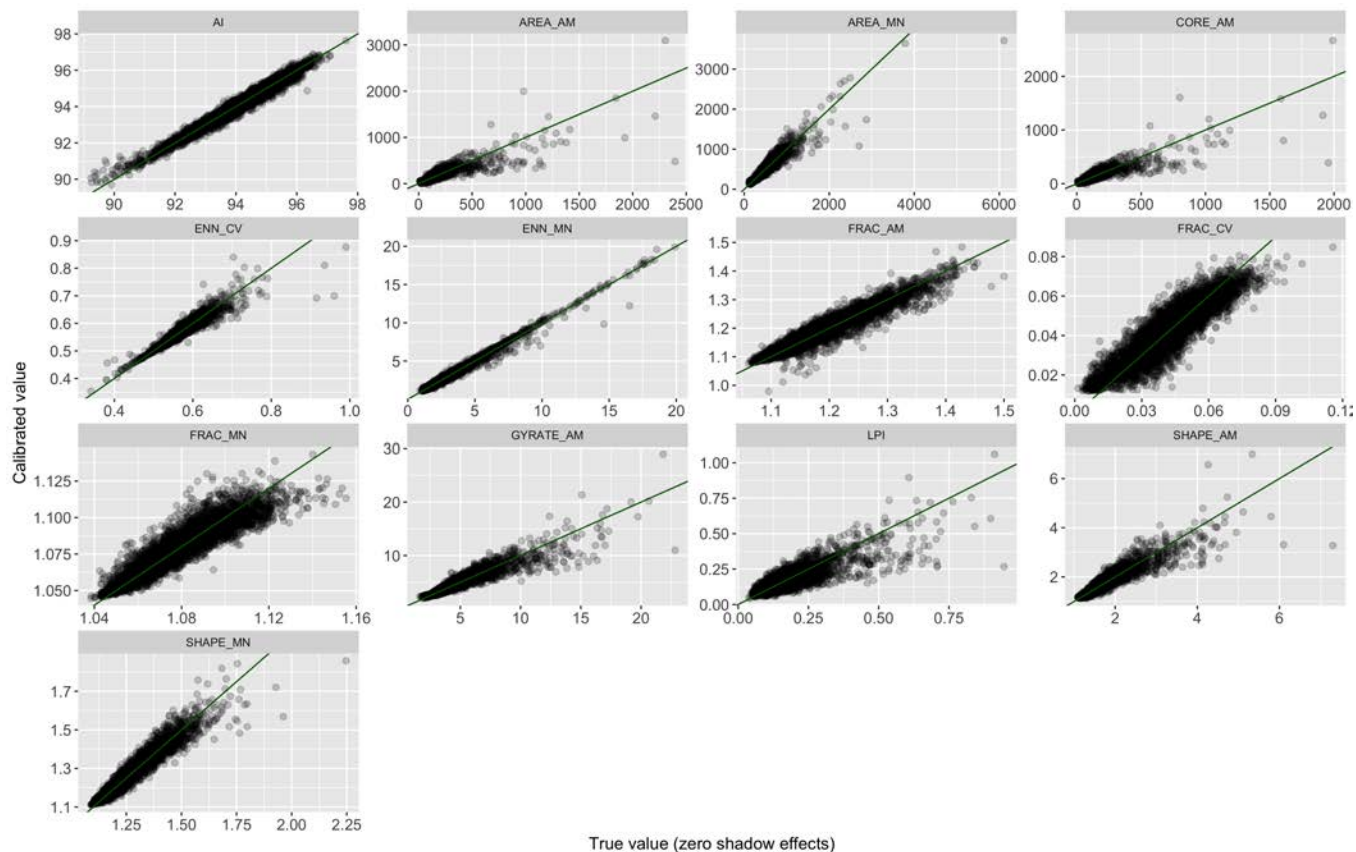


Figure 5: The predicted ('calibrated') values of each LSM against their true values.

Shadow calibration – additional methodological details

Generating reasonably 'realistic' and representative tree canopies required several steps. Specifically, we conditioned simulations on USFS Forest Inventory and Analysis (FIA) data, which we downloaded from the FIA DataMart (https://apps.fs.usda.gov/fiadb-downloads/CSV/datamart_csv.html) for Arizona. We filtered the plot table for plots within the project area⁴ and used these plot indices to select all relevant tree records from the tree table.⁵ There were 122 plots that met the criteria specified above. Taken together, these plots contained 2,150 trees.

Simulating trees and tree canopies in a large number of arbitrary 1-acre areas requires drawing numbers of stems either with replacement from the empirical distribution, or from a predictive distribution. We chose the latter to better sample across the gradient of potential tree densities in the project area. Specifically, we drew numbers of stems drawn from a negative binomial distribution characterized using a simple Bayesian model fit to the plot data. For the sake of brevity, we will refrain from describing these models here, but complete details regarding model parameterizations can be found on the repository (<https://bitbucket.org/lzachmann/4fri-lpa>).⁶ Posterior predictive checks for the model indicated that it approximated the data quite well. For example, Figure 6 shows the empirical and predicted distribution of the number of trees per acre (based on estimates generated at the plot level).

We assigned DBH values to stems using a complementary, but separate model – one built to characterize the distribution of stem sizes within plots (taking the number of trees into account). Stems

were then assigned crown widths following the equations in Bechtold (2003, 2004) that predict crown width from stem diameter. The height of each stem was predicted in a similar fashion. Stems were located completely at random within the 1-acre area.⁷

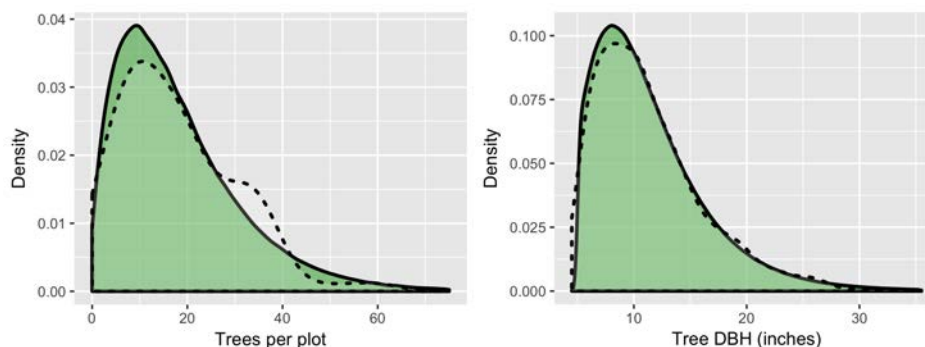


Figure 6. Probability density functions for the empirical (dotted, transparent white curve) and predictive (solid, green curves) distributions of the number of trees per plot and tree DBH used in simulations tied to the shadow calibration. Note that tree DBH varies as a function of trees per plot. As such, the model used to characterize the distribution of tree DBH includes trees per plot as a covariate.

Shadows were simulated using the `insol` (<https://github.com/cran/insol>) R package by assuming that tree crowns – from the altitude at which they were observed (5,770 m) – are effectively cylindrical. We pulled sun altitude and azimuth from the empirical distribution ($n = 1,975$) of sun positions recorded while images were being acquired.⁸

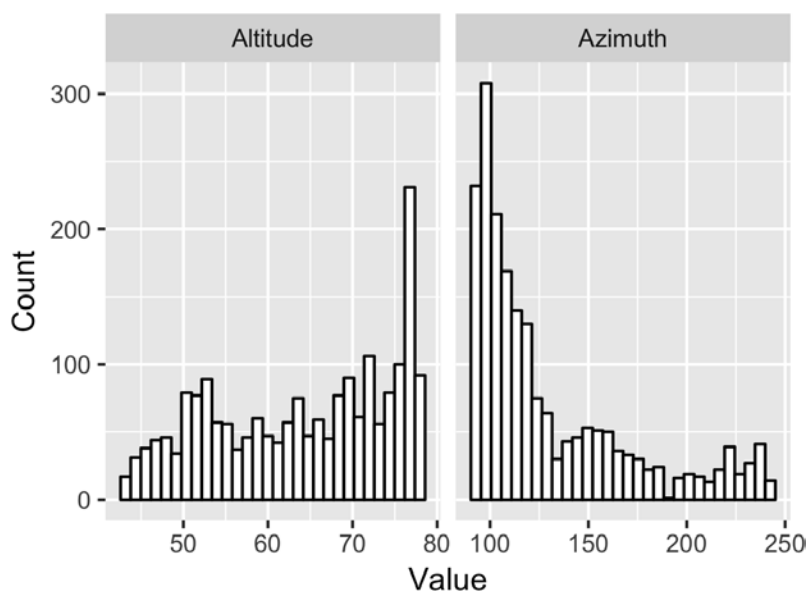
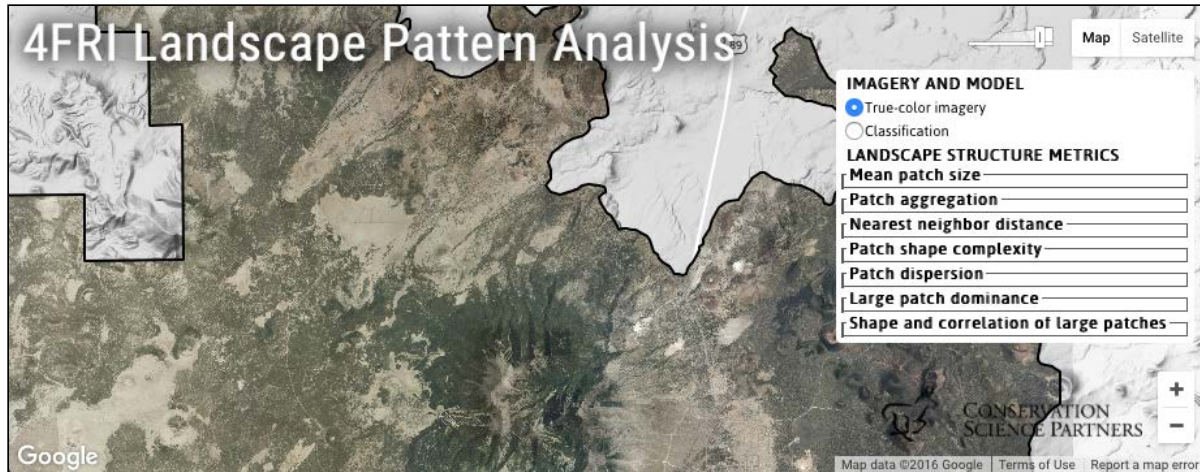


Figure 7. The empirical distributions of sun altitude and azimuth at the time the imagery was acquired.

Links to code and data

Perhaps the easiest way to access, interact with, and visualize the data is to connect to the web app we built for the project.

(<https://four-fri-lpa.appspot.com/>)



(<https://four-fri-lpa.appspot.com/>)

Figure 8: A screenshot of the web application used to visualize the data products. You can click anywhere on the figure to launch the application in a new browser window.

The datasets themselves are accessible via Google Cloud Platform Buckets. Accessing the data will require signing in with a Gmail account.

- Canopy cover models (<https://storage.cloud.google.com/four-fri-rf-ccc-v2>), one for each DOQQQ. Canopy cover models are at 0.3 m resolution and consist of 3 distinct values (0, 1, and 2 for 'canopy', 'shadow', and 'other', respectively).
- Canopy cover summaries (<https://storage.cloud.google.com/four-fri-cc-summaries-v2>) (proportion canopy cover) served as single large mosaicked images. Rasters show canopy cover at 1-, 10-, and 100-acres scales (i.e., within 1-, 10-, and 100-acre moving windows) at a 6 m resolution.
- Landscape structure metrics (<https://storage.cloud.google.com/four-fri-lpis>), which are also served as DOQQQs, though are nested in folders as either the raw, 'observed' LSMs or 'calibrated' LSMs. Each raster contains estimates of LSMs at a given scale and consists of 13 bands (1 for each LSM: AREA_MN, AI, ENN_MN, SHAPE_MN, FRAC_MN, FRAC_AM, FRAC_CV, ENN_CV, LPI, AREA_AM, CORE_AM, GYRATE_AM, SHAPE_AM, respectively). LSM rasters are at ~32 m resolution. The specific scale is conveyed in the suffix for each file: e.g., `lpis_at_1_acre_(calibrated_lpis).tif`.

Differentiating scales of analysis and image resolution⁹

LSMs (and other statistics involving image neighborhoods, such as canopy cover) can be assembled on a pixel-by-pixel basis. However, producing LSMs – in, for example, 1-acre neighborhoods – by sliding the window from one 0.3 m cell to the next would produce a lot of redundant information at an unnecessarily fine scale, not to mention at a prohibitively expensive time- and compute-cost. As such, we considered two alternative approaches to producing such statistics (albeit at somewhat lower final image resolutions): blocks and 'block subsampling'. In each case, blocks are sized according to the desired scale of analysis and the image is subdivided into discrete 'sublandscapes'. If simple, discrete

blocks are used the final resolution of the image would correspond directly to the scale of analysis. In other words, at a 1 acre scale of analysis, the landscape would be subdivided into many adjacent 1 acre cells (square regions ~64 m on a side). However, in the case of block subsampling, image neighborhoods (blocks) form an overlapping grid, which permits creating higher-resolution outputs in what is still an 'acceptable' amount of time (Figure 9). We took the latter approach to develop the 1-acre LSMs referred to above.

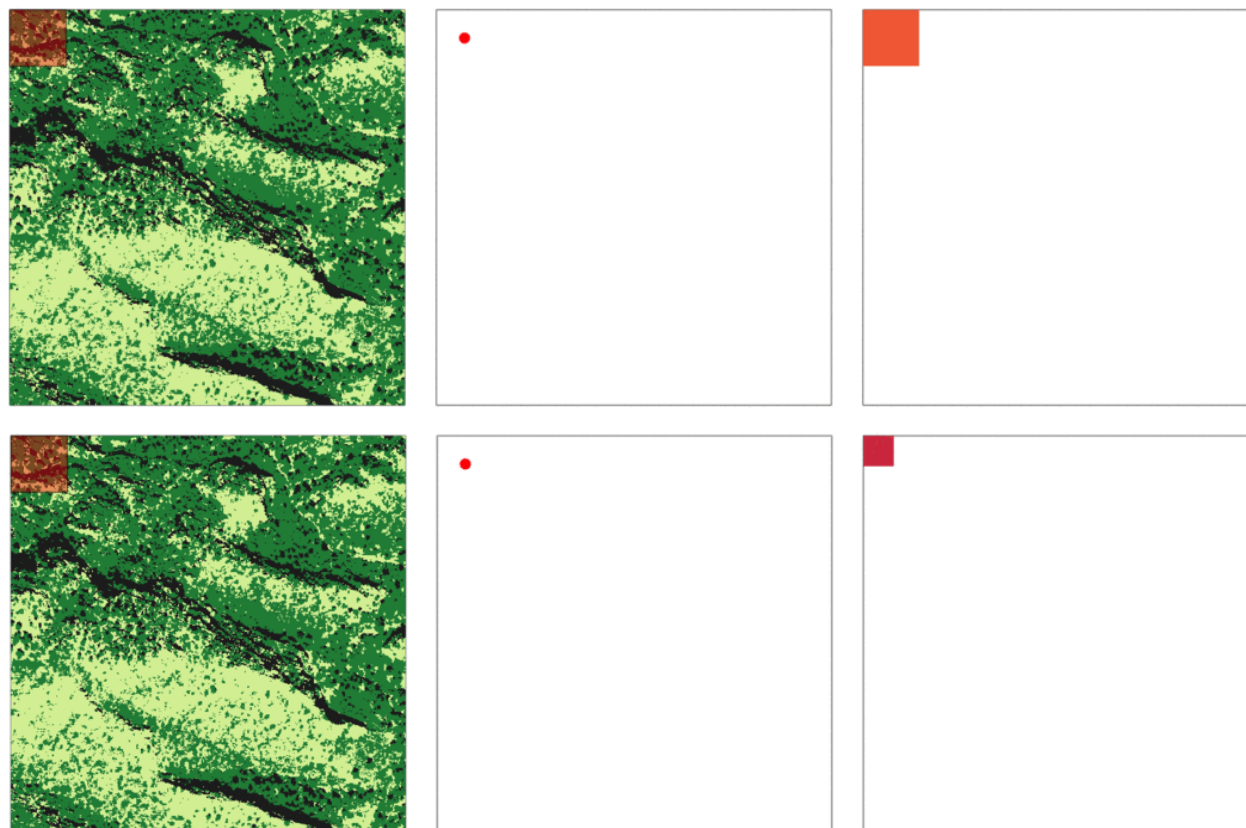


Figure 9. An example of the process used to create 1-acre landscape structure metrics (LSMs). A 1-acre window (in red) slides across the entire classification image incrementally. In the case of simple block sampling (the upper-most animation), the window moves in discrete 1-acre-sized steps or jumps. In the case of block subsampling (the animation on the bottom), the landscape is subdivided into overlapping 'sublandscapes'. The latter approach permits creating neighborhood-based metrics at somewhat higher resolutions (in this case 2X the resolution that would have been obtained by taking the discrete steps/jumps blocking approach). In each case, the LSM value within the moving window is mapped to the window's centroid (as illustrated by the red dot in the second panel of each animation), which in turn is subsequently used to populate the final LSM raster with data (shown in the right-most panel of each animation). We elected to use block subsampling to create the initial 1-acre LSM results.

Finally, several ancillary data files live in a Google Drive folder (<https://drive.google.com/drive/folders/0B75dDdBQu0NWcW5IbjMwbFRSMUU?usp=sharing>). Note that these files are provided largely for archival purposes. Many of the files in this directory were in-stream inputs or outputs to/of code and should not be considered deliverables in-and-of themselves. All of the code used in developing this analysis is available in a Bitbucket repository (<https://bitbucket.org/lzachmann/4fri-lpa>).

Usage

We recommend visual inspection of the classification (and any derived quantities) for class confusion in green meadow areas, or where *Ponderosa* gives way to scrub brush communities near the rim. Steep terrain poses an especially difficult challenge. Future work may include evaluating changes in canopy cover over time.

Acknowledgements

L. J. Zachmann and B. G. Dickson received support from the the USDA Forest Service, Coconino National Forest (challenge cost share supplemental project agreement 15-CS-11030420-033). Christopher Ray, Valerie Horncastle, and Michael Peters contributed to training (and other ancillary) data development.

References

- Anselin L. 1995. Local indicators of spatial association—LISA. *Geographical Analysis* **27**:93–115. Available from <http://onlinelibrary.wiley.com/doi/10.1111/j.1538-4632.1995.tb00338.x/abstract> (<http://onlinelibrary.wiley.com/doi/10.1111/j.1538-4632.1995.tb00338.x/abstract>) (accessed October 25, 2016).
- Breiman L. 2001. Random forests. *Machine Learning* **45**:5–32. Available from <http://link.springer.com/10.1023/A:1010933404324> (<http://link.springer.com/10.1023/A:1010933404324>) (accessed October 25, 2016).
- Connors RW, Trivedi MM, Harlow CA. 1984. Segmentation of a high-resolution urban scene using texture operators. *Computer Vision, Graphics, and Image Processing* **25**:273–310. Available from <http://www.sciencedirect.com/science/article/pii/0734189X8490197X> (<http://www.sciencedirect.com/science/article/pii/0734189X8490197X>) (accessed October 25, 2016).
- Cushman SA, McGarigal K, Neel MC. 2008. Parsimony in landscape metrics: Strength, universality, and consistency. *Ecological Indicators* **8**:691–703. Available from <http://www.sciencedirect.com/science/article/pii/S1470160X07001306> (<http://www.sciencedirect.com/science/article/pii/S1470160X07001306>) (accessed September 21, 2016).
- De Martino F, Valente G, Staeren N, Ashburner J, Goebel R, Formisano E. 2008. Combining multivariate voxel selection and support vector machines for mapping and classification of fMRI spatial patterns. *NeuroImage* **43**:44–58.
- Farr TG et al. 2007. The shuttle radar topography mission. *Reviews of Geophysics* **45**:RG2004. Available from <http://onlinelibrary.wiley.com/doi/10.1029/2005RG000183/abstract> (<http://onlinelibrary.wiley.com/doi/10.1029/2005RG000183/abstract>) (accessed October 25, 2016).
- Haralick RM, Shanmugam K, Dinstein I. 1973. Textural features for image classification. *IEEE Transactions on Systems, Man, and Cybernetics* **SMC-3**:610–621.

Theobald DM, Stevens DL, White D, Urquhart NS, Olsen AR, Norman JB. 2007. Using GIS to generate spatially balanced random survey designs for natural resource applications. *Environmental Management* 40:134–146. Available from <http://link.springer.com/article/10.1007/s00267-005-0199-x> (<http://link.springer.com/article/10.1007/s00267-005-0199-x>) (accessed September 21, 2016).

1. Level 14 of the nested hierarchical global grid.↵
2. DOG involving ‘fat’ and ‘skinny’ Gaussian kernels $\sigma = 7$ and 1, respectively, for a Gaussian kernel of radius 7 pixels.↵
3. For more information on GLCM outputs, see Haralick et al. (1973) and Connors et al. (1984).↵
4. Selected plots had to meet several additional criteria. Namely, we selected plots that were field visited (physically examined), and for which all four subplots were fully described. Additionally, we removed any variable-radius plots. If a given plot was sampled more than once, we selected data only from the most recent inventory year. Finally, only plots for which all tree records were ponderosa pine were selected.↵
5. As with the plot table, tree records used in this analysis had to meet several criteria. Specifically, they had to be live trees >5 inches DBH.↵
6. See `lpi_calib_n_trees_per_acre_(distribution).R` and `lpi_calib_stem_dbh_(distribution).R`.↵
7. We found no evidence of significant spatial correlation of DBH among stems within subplots (using both variograms and the Mantel test). Additionally, we evaluated whether stem locations followed a uniform Poisson point pattern, which they did.↵
8. Sun positions were calculated using the dates reported for each image and vetted using NOAA’s Solar Calculator (<http://www.esrl.noaa.gov/gmd/grad/solcalc/>).↵
9. Scales of analysis in the context of this work are typically reported as units of area, whereas resolution is generally reported as a linear unit (e.g., 1 acre and 0.3 m, respectively).↵

Copyright © 2016 Landscape Conservation Initiative. All rights reserved.