Fourier Series Applications in Multitemporal Remote Sensing Analysis using Landsat Data

Evan Beren Brooks

Dissertation submitted to the faculty of Virginia Polytechnic Institute and State University in partial fulfillment of the requirements for the degree of

Doctor of Philosophy
in
Forestry

Randolph H. Wynne, Co-Chair
Valerie A. Thomas, Co-Chair
John W. Coulston
Philip J. Radtke
Curtis E. Woodcock

May 8, 2013
Blacksburg, Virginia

Keywords: harmonic analysis, phenology, interpolation, data fusion, trajectory, thinning, statistical process control, productivity, site index

Fourier Series Applications in Multitemporal Remote Sensing Analysis using Landsat Data

Evan Beren Brooks

ABSTRACT

Researchers now have unprecedented access to free Landsat data, enabling detailed monitoring of the Earth's land surface and vegetation. There are gaps in the data, due in part to cloud cover. The gaps are aperiodic and localized, forcing any detailed multitemporal analysis based on Landsat data to compensate.

Harmonic regression approximates Landsat data for any point in time with minimal training images and reduced storage requirements. In two study areas in North Carolina, USA, harmonic regression approaches were least as good at simulating missing data as STAR-FM for images from 2001. Harmonic regression had an $R^2 \geq 0.9$ over three quarters of all pixels. It gave the highest $R^2_{Predicted}$ values on two thirds of the pixels. Applying harmonic regression with the same number of harmonics to consecutive years yielded an improved fit, $R^2 \geq 0.99$ for most pixels.

We next demonstrate a change detection method based on exponentially weighted moving average (EWMA) charts of harmonic residuals. In the process, a data-driven cloud filter is created, enabling use of partially clouded data. The approach is shown capable of detecting thins and subtle forest degradations in Alabama, USA, considerably finer than the Landsat spatial resolution in an on-the-fly fashion, with new images easily incorporated into the algorithm. EWMA detection accurately showed the location, timing, and magnitude of 85% of known harvests in the study area, verified by aerial imagery.

We use harmonic regression to improve the precision of dynamic forest parameter estimates, generating a robust time series of vegetation index values. These values are classified into strata maps in Alabama, USA, depicting regions of similar growth potential. These maps are applied to Forest Service Forest Inventory and Analysis (FIA) plots, generating post-stratified estimates of static and dynamic forest parameters. Improvements to efficiency for all parameters were such that a comparable random

sample would require at least 20% more sampling units, with the improvement for the growth parameter requiring a 50% increase.

These applications demonstrate the utility of harmonic regression for Landsat data. They suggest further applications in environmental monitoring and improved estimation of landscape parameters, critical to improving large-scale models of ecosystems and climate effects.

DEDICATION

I dedicate this work to my daughters, Elanor and Bridget, in the hopes that it can help provide a better future for them.

ACKNOWLEDGMENTS

I want to thank my advisors, Val Thomas and Randy Wynne, for initiating me into the field of remote sensing and providing me with an environmentally-focused forum in which to apply my statistical background. I thank them deeply for all of their continued support and understanding, both in my studies and in life at large. They have always been ready with potent feedback and were willing to give me a wonderfully free rein in exploring the topics in this work. They have shown me, both in their classes and in their capacity as advisors, how rewarding and enjoyable the academic profession can be. I eagerly anticipate a very productive and exciting future with them both.

I also want to thank John Coulston of the Southern FIA program for his support. Without the funding provided by the FIA, this work would never have taken place. Without his ready willingness to provide an outside perspective, the work would have had a much narrower focus. And without his involvement in providing and helping me analyze the FIA plot data, the third segment of this work would not have been possible.

I also want to thank the other members of my committee, Phil Radtke and Curtis Woodcock. I greatly appreciate their willingness to listen in on my progress and their flexibility as the slings and arrows of research changed the scope of this work.

I also want to thank the Department of Forest Resources and Environmental Conservation for both the continued support and freedom they have given me to accomplish my duties as a student. In particular, I want to thank the wonderful Stacey Kuhar and Sue Snow for always being there to answer any administrative questions I have and take care of the behind-the-scenes work I don't even know about, and I want to thank the Department head, Janaki Alavalapati, for his enthusiasm and open door. I have never known a department head more responsive to the needs of his students.

I also want to thank everyone who has had a part in the Landsat program. The ability to view the Earth in such detail over decades is central to large-scale environmental monitoring. Without Landsat and its brethren sensors in orbit, we would be blind to so much. In particular, I want to thank the US Geological Survey for making Landsat data freely available to the public. That kind of forward thinking made both my work and the larger transition to a continuous monitoring paradigm possible.

Most of all, I want to thank my wife Kristina and my daughters Elanor and Bridget. Their constant love and support makes all of my effort worthwhile.

Table of Contents

List of Figures

List of Tables

Organization of Dissertation and Attributions

This dissertation is composed of five chapters, with an introduction, three manuscripts, and conclusion. Two of the three manuscripts have been accepted by, or are in review with, a peer-reviewed journal. The third manuscript is under preparation for submission to a peer-reviewed journal as well. The manuscripts were all collaborative efforts, chiefly involving the author and his advisors, Dr. Randolph Wynne and Dr. Valerie Thomas, as well as Dr. John Coulston of the USDA Forest Service Forest Inventory and Analysis program. Dr. Christine Blinn, also of the Department of Forest Resources and Environmental Conservation, contributed as well in the one of the manuscripts. The algorithm development and analysis were the work of the author, while the co-authors contributed in an advisory role and aided the author in reviewing. The manuscripts are organized as follows, with chapter numbers corresponding to the chapters in this work:

- Chapter 2, "Fitting the Multitemporal Curve: A Fourier Series Approach to the Missing Data Problem in Remote Sensing Analysis," was published in September 2012 in *IEEE Transactions in Geosciences and Remote Sensing*. The paper outlines a comparison of methods for interpolating missing data in Landsat time series. The author developed the algorithms and performed all analysis. Valerie Thomas, Randolph Wynne, and John Coulston all contributed to the development of the manuscript.
- Chapter 3, "Detecting Forest Disturbances with Statistical Quality Control Charts from Landsat Data: An On-the-Fly Massively Multitemporal Change Detection Method," is currently in review for *IEEE Transactions in Geosciences and Remote Sensing*. The paper demonstrates an effective method of detecting subtle forest disturbances using Landsat data. The author developed the algorithms and performed all analysis. Christine Blinn provided validation data in the form of harvest polygons and aerial image mosaics, and she, along with Randolph Wynne, Valerie Thomas, and John Coulston, contributed to the manuscript development.
- Chapter 4, "Improving the Precision of Dynamic Forest Parameter Estimates Using Landsat," offers preliminary results in using post-disturbance vegetation index time series to post-stratify Forest Inventory and Analysis plot information in order to achieve greater precision in estimating forest parameters from the plots. The author developed the

algorithms and performed the analysis, with the exception that John Coulston provided the FIA data and the base post-stratification code. John Coulston, Randolph Wynne, and Valerie Thomas all contributed to the development of the manuscript.

Chapter 1: General Introduction

If the proverbial picture is worth 1,000 words, then surely a video is worth 1,000 pictures. Being able to observe the variation of an object or landscape through time literally adds another dimension for interpretation. *Multitemporal analysis* of remotely sensed images of the Earth's surface can be defined as the use of multiple bands in time as well as space from the same sensor or sensor class in order to achieve the usual goals of remote sensing analysis. In effect, the researcher adds a fourth dimension to the analysis, date/time, to the existing dimensions of latitude, longitude, and wavelength. Now that Landsat satellite images for most of the globe are freely available (as of 2009), there is a great potential for the use of this sort of analysis. To access this potential, one must contend with the issues that are unique to multitemporal analysis. Such issues include gaps in the time series due to inclement weather or satellite malfunction, misalignment of images from one date to the next, changes in landscape due to changes in season, interannual shifts in the landscape as forests and other ecosystems go through succession, and sudden shifts due to catastrophic occurrences such as forest fires, clear cutting, or urban expansion [1-10]. Also significant is the sheer amount of processing power it takes to handle many images in an analysis and condense the results of the analysis down to something understandable and manageable.

1.1. Advantages and Disadvantages of Multitemporal Analysis in Remote Sensing

Because multiple images are used, there is a need to ensure that the images are aligned properly so that a given coordinate for one date can rightly be compared with the same location from other dates. Images are ideally corrected to top of atmosphere reflectance through a process such as LEDAPS [11-12] to reduce the effect of variations in atmospheric content on given dates. Additional corrections such as *dark object subtraction*, in which the brightness of the darkest *pixel* (picture element, the atomic unit of an image) is assumed to be 0 and the entire image is adjusted by subtracting the measured brightness of that pixel, may be relevant, especially if the timeframe of the images spans seasonal changes or multiple weather types.

Provided that the images in question are properly preprocessed as above, having multiple dates allows for comparisons of the images in ways that are impossible for single images. No

longer must similar pixels be found to compare to a given pixel, for that same pixel at another date may be observed! The applications in change detection are self-evident, but examples include observing urban development, tracking the succession of ecosystems across landscapes, or measuring the extent of a forest fire [1-10].

The use of multiple images over time allows for alternative ways of interpreting the data. When the pixel locations are allowed to vary over a fixed date and band/wavelength, then we have the usual image familiar to anyone using remote sensing software. On the other hand, if the time element is allowed to vary, then any given pixel can be viewed as having a time series profiling the changes in that pixel over the course of time. Each pixel has a unique temporal signature for each band. These signatures may be used to augment traditional classification approaches, adding a new axis along which to group similar pixels. When entire images are flipped through in chronological order, the scene becomes animated in literally the same manner as drawings in a movie, with the individual scenes serving as frames.

With the addition of a time dimension, there are other issues to consider that do not come up in single scene analyses. Perhaps the most notorious of these is the missing-data issue which arises when scenes on some dates are inaccessible, usually due to weather and cloud cover [13-15]. We also have the issue of how frequently a satellite flies over a given location. For example, the Landsat series of satellites, including the currently operational Landsats 7 and 8, have a flyover period of 16 days. If there is a substantial delay between flyovers, critical features of the scene can be lost. If we think about remotely sensed images as being frames in a movie film, then these issues may be thought of as smudged frames and a low frames-per-second, respectively.

Thus, for multitemporal analysis to be widely useful to researchers, methods are needed to cope with these issues. The methods' utility is ultimately determined by the type of desired analysis: for only two scenes, a simple comparison of two carefully selected scenes may suffice. For analyses spanning years of imagery, something that can distill the essence of the data is needed. This study has a focus on long-term analysis, and so the methods presented herein will relate to that aspect.

1.2. Fourier Series and Harmonic Regression

The term *Fourier series* has two usages in remote sensing, and while they are related, they should not be confused. In one sense, Fourier transformations of data can isolate features in the spectral space, decomposing the recorded waveforms of the image's pixels. This sort of analysis is used in high and low pass filters, among other things. While extremely useful and powerful, that is not the type of Fourier analysis referred to in this study. Rather, when the term *Fourier* is used, the idea here is to decompose a *temporal sequence* into its component parts. This allows a sequence of data points to be characterized by a (typically) much smaller sequence of Fourier coefficients.

The formal definition of a Fourier series in this context follows, available in many textbooks such as [16]. Here, we use the convention that capital letters denote sets of objects for which the corresponding lowercase letters are individual elements. Suppose we have d images spanning dates $T = \{t_1, t_2, \dots, t_d\}, t_1 < t_2 < \cdots < t_d$ such that these dates are written as days of the year, which without loss of generality are rescaled to the interval $[0,2\pi]$ by multiplying by $\frac{2\pi}{365}$. Note that in the event of multiple years as inputs, the modified day of the year may be further simplified by taking the remainder modulo 2π. Each image contains sets $X = \{x_1, x_2, \dots x_l\}$ of l lines and $Y = \{y_1, y_2, \dots y_s\}$ of s samples, conceivable as the columns and rows of the images, respectively. Each image also contains w wavelength bands in a set $B = \{b_1, b_2, \dots b_w\}$. Thus a given pixel may be considered as an ordered quadruple of vectors, namely the pixel p may be identified by the knowledge of (x, y, b, t). In this context, the somewhat abused notation (X, Y, b, t) identifies an image at a particular band and date, whereas (x, y, b, T) is a time series for a particular location and band and (x, y, B, t) is the approximate waveform for a particular location and time.

In this case, for a time series (x, y, b, T), with T being a subset of the interval $[0,2\pi]$, the Fourier series expansion of this time series is given by the functional form

$$f(x, y, b, t) \doteq f_{xyb}(t) = a_{xyb0} + \sum_{j=1}^{\infty} \left(a_{xybj} \sin(jt) + b_{xybj} \cos(jt) \right) \quad (1.1)$$

Note the simplification of the notation to reinforce that each pixel has its own unique time series. Provided that the time series remains finite over the interval $[t_1, t_d]$, this series exists and the equation is exact. The j indexes over the *harmonics* of the Fourier series, hence, Fourier analysis is often called *harmonic analysis*. Note that for a given pixel and band, the Fourier series is completely characterized by the coefficients. Thus, one may reconstruct the series, for any point in the period, from the vector of harmonic coefficients.

Practically speaking, we approximate the true series by truncating it after a chosen n^{th} harmonic so that

$$f_{xyb}(t) \approx \varphi_{xyb}(t) = a_{xyb0} + \sum_{j=1}^{n} \left(a_{xybj} \sin(jt) + b_{xybj} \cos(jt) \right) \quad (1.2)$$

For our purposes, we will model the time series by

$$f_{xyb}(t) = \varphi_{xyb}(t) + \varepsilon_{xyb}(t) \quad (1.3)$$

where the second term, $\varepsilon_{xyb}(t)$, is an error term which has some assumed probability distribution. For example, if this distribution is normal with mean 0, then estimating the coefficients for φ is a matter of multivariate regression, as is shown in part 2.3.2 of this work. In this context, such regression as in (1.3) is denoted as *harmonic regression* to emphasize the particular model being used. Again, note that the specific values of the terms are implicitly dependent on the pixel's geographic and spectral location.

Note that harmonic regression techniques have been used successfully for coarser resolution sensor platforms, such as the Moderate resolution Imaging Scanner (MODIS) series. [39] These platforms have a daily image turnover and coarser spatial resolution than the Landsat platforms, allowing for easy fitting from the large number of images over a given year. The challenge with Landsat data in particular is that its image turnaround time is 16 days, resulting in a much shallower pool of images on which to draw.

Regardless of the sensor platform, with a raster of harmonic coefficient estimates in hand, the researcher may then apply any technique from time series analysis or calculus on the smooth curves that the coefficients characterize. Harmonic regression will be the basis for managing the

issues presented in part 1.1. Its application paves the way for a multitude of multitemporal analyses of Landsat data, of which we focus on two for this work as launching platforms.

1.3. Quality Control Charts and Subtle Change Detection

A quality control chart is a statistical tool used to monitor measurements taken from a process, such as a manufacturing plant's output of a product. This chart relies on measurements taken at successive times and will *signal* the operator when the process moves outside the predetermined acceptable bounds, or *control limits*. There are many control charts in existence, tunable to different control limits (small deviations vs. large deviations, consistent shift vs. oscillations, etc.) [20-27]. While this method is applied routinely in industrial fields, there seems to be no literature available in using it for ecosystem monitoring.

The potential application to multitemporal analysis in remote sensing is clear: given a sequence of images, one can theoretically use a control chart to detect disturbances *as they happen*, up to the frequency of the incoming images. The signals could be placed on an ordered scale to indicate the severity of the disturbance, ranging from subtle thins well below the spatial resolution of the sensor to extreme changes like a clear-cut harvest or the aftermath of a fire. Two control charts which specialize in detecting small gradual changes include exponentially weighted moving average (EMWA) charts and cumulative sum (CUSUM) charts. [20-21]

Additionally, a simple Shewhart chart [20-21] would suffice for detecting major disturbances such as a clearcut, but perhaps it is even more useful in signaling for small-scale cloud interference that slips through the cloud detection masks. If so, one could employ more images from the archive to extract more information from the partially clouded images, broadening the ability of harmonic regression to fit curves to the data.

1.4. Dynamic Forest Parameters and Forest Inventory and Analysis Plots

The US Forest Service employs a system of Forest Inventory and Analysis (FIA) plots to take annual measurements of various forest parameters, ranging from tree species and size to crown conditions, undergrowth, and soil quality [17, 19, 28]. The plots are located on a variety of land classes and are subject to different classes of owners. Their locations are classified in accordance with the 2000 amendment to the Department of the Interior and Related Agencies Appropriations Act.

Each FIA plot is constructed according to the following pattern, shown in Figure 1.1. [28] From the centerpoint of the plot, a circle of radius 120' (feet) is drawn out. At equidistant points along the circumference of this circle, three points are marked out. From each of these points as well as the centerpoint of the plot, a circle of radius 24' is drawn, yielding four smaller circles arranged around and within the larger circle. For comparison, the overall FIA plot size (about 1/6 acre total for all four circles) is comparable to a 3x3 collection of Landsat 30-m pixels if the center of the neighborhood matches the center of the plot. Within each of the circles, forest measurements are taken, including but not limited to volume, heights, diameters at breast height, and number of trees. [17-18] Because diameter and height can be used to reasonably estimate biomass via allometric equations, the FIA plots are also used to estimate the amount of carbon stored in forests.

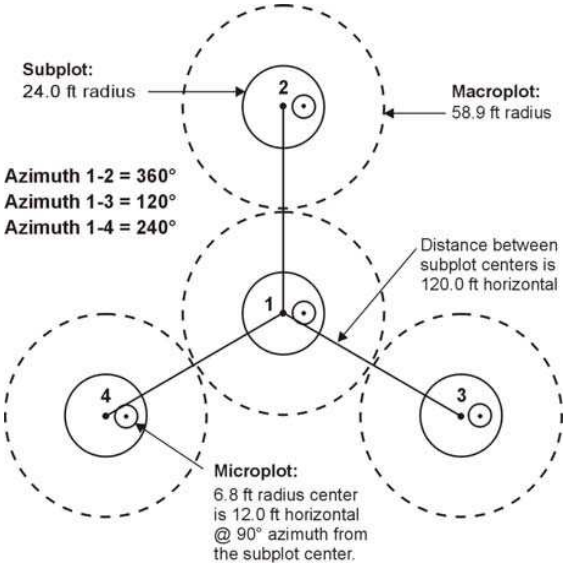


Figure 1.1. FIA plot layout. [28]

The FIA plots' initial positions were chosen according to random placement within a hexagonal grid. [17, 28] Historical measurement intervals vary by plot and by state, but typically each plot is measured at least once every decade.

Due to the infrequent sampling of any given plot, the inventory process is routinely augmented by using Landsat scenes which cover the area more regularly and with broader coverage. [28-38] In particular, Landsat-based thematic maps have been used successfully in increasing the precision of post-stratified estimates for many static forest parameters, such as carbon and forest coverage. [29]

With decades' worth of free Landsat data now available, it becomes possible to employ multitemporal approaches to improve the precision of estimates for dynamic forest parameters, such as growth, removal, and mortality. Improving these estimates would be of import not only for the FIA program but also for carbon and climate models that rely on precise estimates of such dynamic parameters.

1.5. Objectives

Based on parts 1.1 through 1.4, the objectives of this work become threefold, answering the following questions.

1) Is harmonic regression appropriate and desirable for imputing Landsat data that are temporally distributed in a scattered manner? Specifically, how do the fitting ability and robustness of harmonic regression compare to the alternative Landsat temporal imputation method of STAR-FM/ESTAR-FM?
2) What is the utility of applying quality control charts to the residual time series from the harmonic regression? Can we use them to detect landscape disturbances on a wide range of severities, in an on-the-fly manner?
3) Can we improve the precision in post-stratified estimates of dynamic forest parameters from FIA plots by employing harmonic regression-based vegetation time series?

In completing these objectives, it is my hope to illustrate not only the utility of harmonic regression of Landsat data, but also to introduce several new paradigms about disturbance detection, environmental monitoring, and dynamic classification of land surface features. These paradigms would then greatly expand the scope of multitemporal remote sensing at the Landsat resolutions, allowing for policymakers to incorporate more detailed and timely information into their decision-making processes.

1.6. References

[1] Miller, J. D., and Yool, S. R. (2002) "Mapping forest post-fire canopy consumption in several overstory types using multi-temporal Landsat TM and ETM data." *Remote Sensing of Environment*, 82(2), 481-496.

[2] Miller, J. D., Safford, H. D., Crimmins, M., and Thode, A. E. (2008). "Quantitative evidence for increasing forest fire severity in the Sierra Nevada and Southern Cascade mountains, California and Nevada, USA." *Ecosystems*, 12(1), 16-32.

[3] Matricardi, E. A. T., Skole, D. L., Pedlowski, M. A., Chomentowski, W., and Fernandes, L. C. (2010) "Assessment of tropical forest degradation by selective logging and fire using Landsat imagery." *Remote Sensing of Environment*, 114(5), 1117-1129.

[4] Santoro, M., Franson, J. E. S., Eriksson, L. E. B., and Ulander, L. M. H. (2010) “Clear-cut detection in Swedish boreal forest using multi-temporal ALOS PALSAR backscatter data.” *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 3(4), 618-631

[5] Hais, M., Jonasova, M., Langhammer, J., and Kucera, T. (2009) “Comparison of two types of forest disturbance using multitemporal Landsat TM/ETM+ imagery and field vegetation data.” *Remote Sensing of Environment*, 113(4), 835-845.

[6] Schroeder, T., A., Wulder, M. A., Healey, S. P., and Moisen, G. G. (2011) “Mapping wildfire and clearcut harvest disturbances in boreal forests with Landsat time series data.” *Remote Sensing of Environment*, 115(6), 1421-1433.

[7] Ayres, M. P., and Lombardero, M. J. (2000) “Assessing the consequences of global change for forest disturbance from herbivores and pathogens.” *Science of the Total Environment*, 262(3), 263-286.

[8] Munyati, C., and Kabanda, T. A. (2008) “Using multitemporal Landsat TM imagery to establish land use pressure induced trends in forest and woodland cover in sections of the Soutpansberg Mountains of Venda region, Limpopo Province, South Africa.” *Regional Environmental Change*, 9(1), 41-56.

[9] Liu, X., Li, X., Chen, Y., Tan, Z., Li, S., and Ai, B. (2010) “A new landscape index for quantifying urban expansion using multi-temporal remotely sensed data.” *Landscape Ecology*, 25(5), 671-682.

[10] Lu, D., Moran, E., and Hetrick, S. (2011) “Detection of impervious surface change with multitemporal Landsat images in an urban–rural frontier .” *ISPRS Journal of Photogrammetry and Remote Sensing*, 66(3), 298-306.

[11] Masek, J. G., Vermote, E. F., Saleous, N. E., Wolfe, R., Hall, F. G., Huemmrich, K. F., Gao, F., Kutler, J., and Lim, T.-K. (2006) "A Landsat surface reflectance dataset for North America, 1990–2000." *IEEE Geoscience and Remote Sensing Letters*, 3(1), 68-72.

[12] LEDAPS Tools website. http://ledaps.nascom.nasa.gov/tools/tools.html

[13] Asner, G. P. (2001) “Cloud cover in Landsat observations of the Brazilian Amazon.” *International Journal of Remote Sensing*, 22(18), 3855−3862.

[14] Jorgensen, P. V. (2000) “Determination of cloud coverage over Denmark using Landsat MSS/TM and NOAA-AVHRR.” *International Journal of Remote Sensing*, 21(17), 3363−3368.

[15] Ju, J. C. and Roy, D. P. (2008) “The availability of cloud-free Landsat ETM+ data over the conterminous United States and globally.” *Remote Sensing of Environment*, 112(3), 1196−1211.

[16] Bloomfield, P. (2004) *Fourier Analysis of Time Series: An Introduction*. Wiley-Interscience, 2nd ed. ISBN-10: 0471889482.

[17] USFS FIA Fact Sheet, “Data collection and analysis.” 2005.

[18] USFS FIA Fact Sheet, “Sampling and plot design.” 2005.

[19] USFS FIA Fact Sheet, “Phase 2 and Phase 3: ground measurements.” 2005.

[20] Gupta, B. C. and Walker, F. W. Bloomfield, P. (2007) *Statistical Quality Control for the Six Sigma Green Belt*. American Society for Quality, Quality Press. ISBN: 978-0-87389-686-3.

[21] Montgomery, D. C. (2008) *Introduction to Statistical Quality Control*. 6th ed. Wiley. ISBN: 978-0470169926.

[22] Ning, X., Shang, Y., Tsung, F. (2009) "Statistical process control techniques for service processes: a review." 6th International Conference on Service Systems and Service Management. pp.927-931.

[23] Shah, S., Shridhar, P., Gohil, D. (2010) “Control chart : a statistical process control tool in pharmacy.” *Asian Journal of Pharmaceutics*, 4(3), 184-92.

[24] Steiner, S. H., Jones, M. (2009) “Risk-adjusted survival time monitoring with an updating exponentially weighted moving average (EWMA) control chart.” *Statistics in Medicine*, 29(4), 444-454.

[25] Yu, J. and Liu, J. (2011) “LRProb control chart based on logistic regression for monitoring mean shifts of auto-correlated manufacturing processes.” *International Journal of Production Research*, 49(8), 2301-2326.

[26] Shewhart, W. A. (1980) *Economic Control of Quality of Manufactured Product*. American Society for Quality Control, Quality Press. ISBN: 978-087389-076-2.

[27] Reynolds, M.R.J. and Cho, G.Y. (2011) “Multivariate control charts for monitoring the mean vector and covariance matrix with variable sampling intervals.” *Sequential Analysis*, 30(1), 1-40.

[28] Scott, C. T., Bechtold, W. A., Reams, G. A., Smith, W. D., Westfall, J. A., Hansen, M. H., and Moisen, G. G. (2005) "Sample-based estimators used by the Forest Inventory and Analysis national information management system." *The Enhanced Forest Inventory and Analysis Program — National Sampling Design and Estimation Procedures.* U.S. Forest Service General Technical Report, SRS-80. pp. 43–67.

[29] McRoberts, R. E., Holden, G. R., Nelson, M. D., Liknes, G. C., and Gormanson, D. D. (2005) "Using satellite imagery as ancillary data for increasing the precision of estimates for the Forest Inventory and Analysis program of the USDA Forest Service." *Canadian Journal of Forest Research,* 35(12), 2968-2980.

[30] Wynne, R.H., Oderwald, R. G., Reams, G.A., and Scrivani, J.A. (2000) "Optical remote sensing for forest area estimation." *Journal of Forestry* 98(5):31-36.

[31] Hansen, M.H., and Wendt, D.G. (2000). "Using classified Landsat Thematic Mapper data for stratification in a statewide forest inventory". *Proceedings of the First Annual Forest Inventory and Analysis Symposium* (November 1999), 20-27.

[32] McRoberts, R. E., Wendt, D. G., Nelson, M. D., and Hansen, M. H. (2002a) "Using a land cover classification based on satellite imagery to improve the precision of forest inventory area estimates." *Remote Sensing of Environment,* 81(1), 36–44.

[33] Hoppus, M. L., and Lister, A. J. (2003) "A statistically valid method for using FIA plots to guide spectral class rejection in producing stratification maps." *Proceedings of the Third Annual Forest Inventory and Analysis Symposium* (October 2001), 17–19.

[34] McRoberts, R. E., Nelson, M. D., and Wendt, D. G. (2002b) "Stratified estimation of forest area using satellite imagery, inventory data, and the k-Nearest Neighbors technique." *Remote Sensing of Environment,* 8(2-3), 457-468.

[35] McRoberts, R. E., and Hansen, M. H. (1999) "Annual forest inventories for the North Central region of the United States." *Journal of Agricultural, Biological, and Environmental Statistics,* 4(4), 361-371.

[36] McRoberts, R. E., Gobakken, T., and Naesset, E. (2012) "Post-stratified estimation of forest area and growing stock volume using lidar-based stratifications." *Remote Sensing of Environment,* 125, 157-166.

[37] Hansen, M. C., Stehman, S. V., Potapov, P. V., Arunarwati, B., Stolle, F., and Pittman, K. (2009) “Quantifying changes in the rates of forest clearing in Indonesia from 1990 to 2005 using remotely sensed data sets.” *Environmental Research Letters*, 4(3).

[38] Westfall, J. A., Patterson, P. L., and Coulston, J. W. (2011) “Post-stratified estimation: within-strata and total sample size recommendations.” *Canadian Journal of Forest Research*, 41(5), 1130-1139.

[39] Hermance, J. F. (2007) “Stabilizing high-order, non-classical harmonic analysis of NDVI data for average annual models by damping model roughness.” *International Journal of Remote Sensing*, 28(12), 2801-2819.

Chapter 2: Fitting the Multitemporal Curve: A Fourier Series Approach to the Missing Data Problem in Remote Sensing Analysis

Evan B. Brooks[a], Valerie A. Thomas[a], Randolph H. Wynne[a], and John W. Coulston[b]

[a]Department of Forest Resources and Environmental Conservation, Virginia Tech, Blacksburg, VA, USA
[b]USDA Forest Service Southern Research Station, Forest Inventory and Analysis Unit, Knoxville, TN, USA

This chapter was published in September 2012 in IEEE Transactions in Geosciences and Remote Sensing, Volume 50, Issue 9, pp.3340-3353.

Abstract

With the advent of free Landsat data stretching back decades, there has been a surge of interest in utilizing remotely sensed data in multitemporal analysis for estimation of biophysical parameters. Such analysis is confounded by cloud cover and other image-specific problems, which result in missing data at various aperiodic times of the year. While there is a wealth of information contained in remotely-sensed time series, the analysis of such time series is severely limited due to the missing data. This paper illustrates a technique which can greatly expand the possibilities of such analyses; a Fourier regression algorithm, here on time series of Normalized Difference Vegetation Indices (NDVI) for Landsat pixels with 30 m resolution. It compares the results with those using the Spatial and Temporal Reflectance Fusion Model (STAR-FM), a popular approach that depends on having MODIS pixels with resolutions of 250 m or coarser. STAR-FM uses changes in the MODIS pixels as a template for predicting changes in the Landsat pixels. Fourier regression had an R^2 of at least 90% over three quarters of all pixels, and it had the highest $R^2_{Predicted}$ values (compared to STAR-FM) on two thirds of the pixels. The typical root mean square error for Fourier regression fitting was about 0.05 for NDVI ranging from 0 to 1. This indicates that Fourier regression may be used to interpolate missing data for multitemporal analysis at the Landsat scale, especially for annual or longer studies.

2.1. Introduction

The collection of Landsat scenes dating from 1972 is one of the largest continuous freely available satellite records of the Earth's surface. At 30 meter pixel resolution, Landsat imagery is used in a variety of moderate and broad-scale applications, including change detection in land use/land cover (LU/LC) classes [1] and ecosystem monitoring [2]-[4]. The continuous record, freely available to the public since 2009, has paved the way for a new level of time series analysis that can capitalize on high spatial and temporal sampling. However, there is a nominal 16 day gap in scenes for each satellite, and many scenes are at least partially obscured by cloud cover [5]-[7]. Thus, methods are needed to facilitate multitemporal analysis of Landsat data. These methods would ideally be robust, easily implemented, and minimize sources of error in their implementation.

Harmonic analysis using Fourier series appears to be an ideal way to facilitate multitemporal analyses using Landsat data, with demonstrated prior efficacy using coarser resolution data such as MODIS and AVHRR, particularly for phenological studies [8] - [14]." Fourier regression analysis has also been applied in areas of health [15] and land development [16].

Fourier series are superimposed sequences, over an interval of time, of a constant with sines and cosines of increasing integer multiples of the original frequency based on the time interval. The constant is called the *mean* of the series, and the pairings of sine and cosine at the specified frequencies are called the *harmonics* of the series. Fourier series can be tailored to any period length, baseline, and amplitude. As the number of harmonics used increases, the Fourier series can converge to any smooth periodic function. Figure 2.1 illustrates the concept of using Fourier series to estimate the underlying curve in a periodic time series.

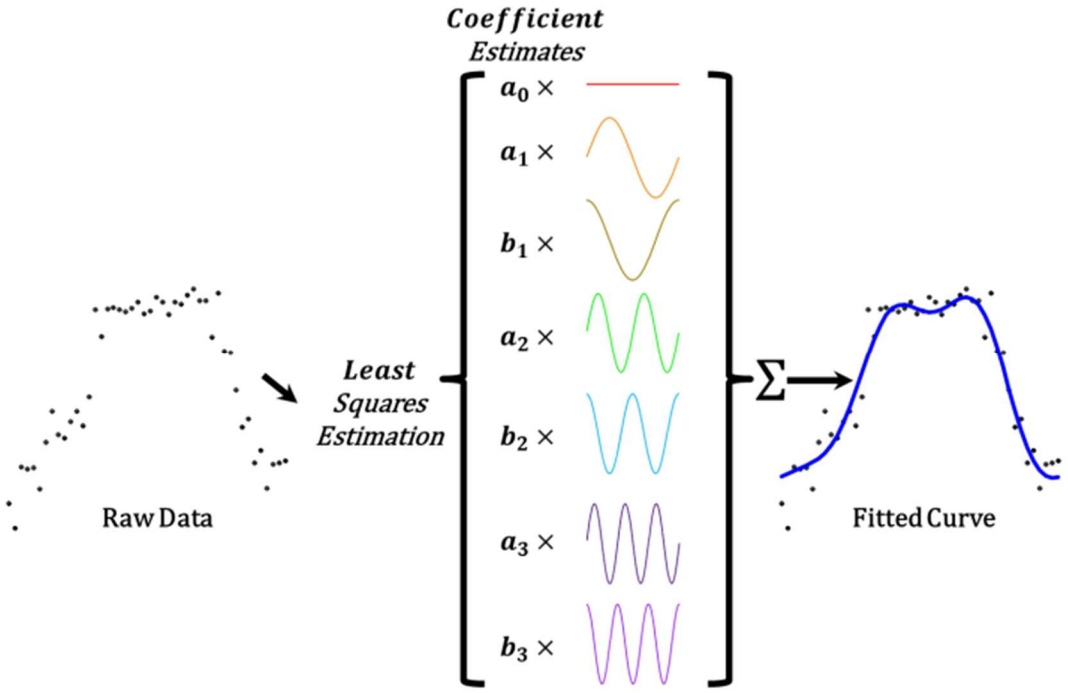


Figure 2.1. Concept of Fourier regression.

Fourier series have been shown to be useful in classification of vegetation types [12] and in the estimation of phenological markers such as start of season, peak of season, end of season, and photosynthetic activity over the growing season [8]-[11], [14]. These methods are largely focused on time series of the normalized difference vegetation index [17], or NDVI, and other similar indices, but they have not been applied to Landsat-scale data and general spectral bands. A thorough review of applications of Fourier series can be found in [14] and [10]. Some of the key results and concerns from the literature can be summed up as follows:

- The mean and first two harmonics cover most of the variation in the data [9], [10]
- Higher-order harmonics are often needed for classifying vegetation types on a more subtle level [11], [12]
- Fourier series of too high an order can swing wildly during times where data are missing, overfitting the remaining points (the “spurious oscillations” of [14])
- Fourier series coupled with polynomials (non-classical harmonic methods) can employ higher order harmonics by reducing the “roughness” in the fit [14]
- Multiple years may be employed in non-classical harmonic algorithms to improve the accuracy and long term trend detection [14].

The above findings all point to Fourier series as having great potential for use in multi-year Landsat analysis. In particular, they imply that if the object is merely to generate "fill" images in the Landsat time series, then a basic fit using only the first two harmonics and mean may suffice. If the object is the estimation of vegetation time series or other finer-scale applications, more harmonics may be required, though care must be taken to avoid overfitting the time series if it is sparse.

When fitting a functional curve to yearly data, there are several inherent advantages in using Fourier series. These include, but are not limited to:

- The fitted curve is periodic, provided that no polynomial terms are incorporated into the fit;
- No ancillary data are required, reducing possible error sources;
- Fourier terms are orthogonal, so there is a reduced chance of multicollinearity (defined as statistical linear association between assumed orthogonal terms in a regression model) provided the data include dates from throughout the year;
- Fourier terms are smooth, facilitating differential calculus approaches to time series analysis; and
- One can store the Fourier regression coefficients in raster form instead of generating images for each day of the year

Another approach to "filling in the gaps" for Landsat coverage is the Spatial and Temporal Adaptive Reflectance Fusion Model, or STAR-FM [18]. Instead of a periodic approach, STAR-FM relies on the inclusion of MODIS imagery to supplement the Landsat scenes. As MODIS has a daily temporal resolution, this can provide sequences of Landsat-scaled scenes. However, MODIS spatial resolution is at best 250 meters. This raises issues of accuracy in heterogeneous regions where MODIS pixels are frequently mixed [18], [19], although the enhanced ESTAR-FM [19] is designed to address this concern. Additionally, MODIS scenes are just as susceptible to cloud issues as those of Landsat. Nevertheless, STAR-FM has been shown to perform well, particularly for short (intra-annual) periods of time in homogeneous areas [18], [19].

The primary aim of this paper is to demonstrate the use of a Fourier regression algorithm in comparison with STAR-FM. In particular, the objective is to check that Fourier regression may be used in lieu of STAR-FM, particularly for annual or interannual analysis of Landsat-scale scenes. One might expect Fourier series, using only 30 m Landsat pixels, to be less impeded by

use in heterogeneous regions than STAR-FM, which depends on both 30 m Landsat pixels and 250 m or larger MODIS pixels. Furthermore, since no additional data are required and since the Fourier series is characterized by its coefficients, the computational and storage costs should be far less than those for using a fusion algorithm like STAR-FM. The primary objectives of this paper are to (1) compare Fourier regression accuracy, in terms of fit and prediction, to STAR-FM in homogeneous regions and (2) quantify the accuracy in heterogeneous regions. With a favorable comparison, remote sensing researchers using Landsat data will have access to another method for enabling multitemporal analysis, one that is particularly well-suited to multi-year research. To demonstrate this multi-year aspect of the method, the secondary objective of the paper is to perform Fourier regression on the study area using scenes from several years. The results of this secondary analysis are compared to those of the single-year version.

2.2. Data

For this paper, the objective is to compare the algorithm shown in Part III with STAR-FM. To further compare the accuracies of the algorithms in different landscape types, two study areas were chosen, shown in Figure 2.2. The areas are both in central North Carolina, USA. One of them, Greensboro, is primarily urban and suburban and is sharply heterogeneous. This area is 9.4 mi by 8.8 mi (15.1 km by 14.2 km), with an area of 82.3 mi^2 (213.3 km^2). The other, the eastern area of Chatham County including and east of Pittsboro, consists primarily of forested and agricultural land with a couple of lakes. This region is smaller than the Greensboro one, with dimensions of 6.9 mi by 6.6 mi (11.1 km by 10.6 km) with an area of 45.4 mi^2 (117.6 km^2). The areas of the land cover classes in this Pittsboro area are large and continuous enough to make this area fairly homogeneous. Between them, the areas include a variety of land cover classes and basic vegetation types typically found in the eastern United States, with varying degrees of heterogeneity. This makes them suitable for testing the Fourier series, especially for forestry and classification applications.

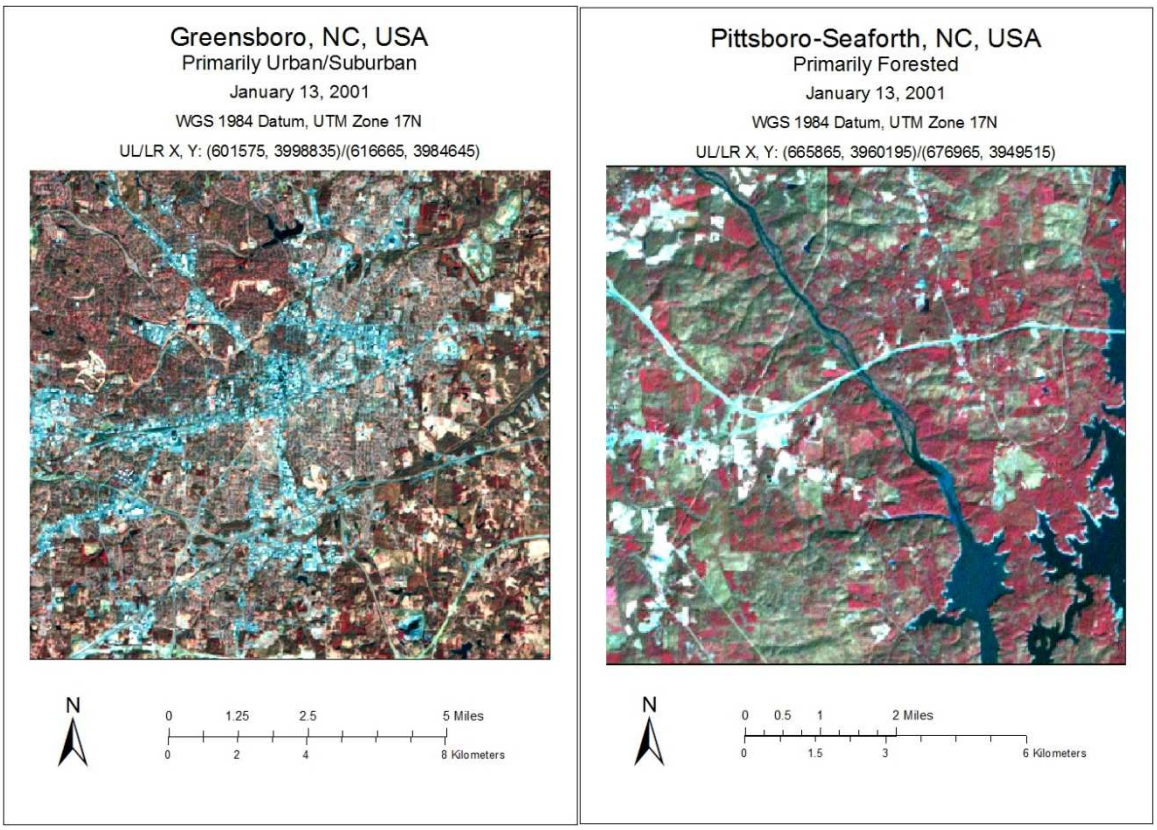


Figure 2.2. Study areas. Colors for both images are Landsat bands 4/3/2 in R/G/B.

Both study areas are from path/row 16/35 in Landsat and H/V 11/5 in MODIS. The areas are relatively proximate to control for the effects of weather and external conditions across the two areas. In all, 17 Landsat scenes were downloaded. Both MODIS Terra daily data (MOD09GQ, 346 scenes) and MODIS Terra 8-day composite images (MOD09Q1, 43 scenes) were downloaded. These scenes had a spatial resolution of 250 m. The study time for the STAR-FM comparison is the year 2001. At this time, both Landsat 5 and Landsat 7 were in operation, jointly providing images at a nominal 8-day interval. Also, at this time, the scan line corrector for Landsat 7 was still functioning. For the multi-year application, additional Landsat scenes were acquired from years 1998 through 2002 over the same study areas. For purposes of profiling and controlling results by LU/LC classification, the National Land Cover Dataset from 2006 was used to assign each pixel to a class. The goal of this paper is not to demonstrate an alternative classification method, although it would certainly be feasible to use Fourier regression-derived curves to aid in classification. The purpose of adding NLCD data was to allow for the possibility of controlling results by LU/LC class, in the event that one class fared particularly well or

poorly. Because the MODIS scenes include bands in the red and near infrared and are geared towards vegetation indices, it was appropriate to use the NDVI in this study. However, the algorithms are designed and intended for use in any spectral band or index. Accordingly, the Fourier regression algorithm was also run on six spectral bands from Landsat. All LU/LC classes were used in the subsequent analysis.

Preprocessing for Landsat scenes included atmospheric correction to surface reflectance using LEDAPS [20] as well as dark object subtraction in an effort to calibrate the images from the two Landsat satellites. Dark object subtraction using band minima was chosen because the year-long nature of the data rendered histogram matching ineffective due to seasonal vegetation changes. Preprocessing for MODIS included resampling, subsetting, and reprojecting the images via the MODIS Reprojection Tool [21] into Landsat scale and projection, converting the 250 m pixels to 30 m pixels.

2.3. Method

2.3.1. Review of STAR-FM and ESTAR-FM

Since the primary purpose of this paper is to compare the results of the Fourier regression approach with the results of STAR-FM, a review of the latter algorithm may be helpful. STAR-FM employs a sequence of linear transformations and regressions across a moving window centered on the pixel in question in order to predict the pixel's brightness value (in whatever band). Specifically, following the lead of [18] and [19], consider a situation in which we have a "fine" image at the Landsat scale of 30 m for time t_0, denoted F_0, and assume we have "coarse" images from the MODIS scale of 250 m for times t_0 and $t_1 > t_0$, designated C_0 and C_1. The inequality may be reversed without losing generality; i.e., one can make a prediction using an input pairing after the predicted date as well. When reprojected to Landsat resolution and coordinates, we can pick out a specific pixel for each image at the corresponding location x, y from the fine image. Then STAR-FM makes the fundamental assumption that for any given band B, the only real difference between the brightnesses of the pixels at time t_0 can be modeled by a linear bias, i.e.,

$$F_0(x, y, t_0, B) = a \times C_0(x, y, t_0, B) + b \quad (2.1)$$

Using linear regression to estimate the coefficients, the algorithm then applies those estimates (designated $\hat{a}$ and $\hat{b}$) to reverse-engineer estimated values for the would-be Landsat scene at time t_1 via

$$F_{predicted}(x, y, t_1, B) = \hat{a} \times C_1(x, y, t_1, B) + \hat{b} \quad (2.2)$$

In actuality, STAR-FM applies a weighted average of such estimates based on nearby pixels deemed similar to the target pixel x, y. The weights are based on three measures of the nearby pixels with respect to the target: the spectral difference between the brightness values in all the bands, the temporal distance between the pixel dates, and the spatial distance between the pixels. The estimation process can be further improved by adding a second "input pair" of fine and coarse images at another time, $t_2 > t_1 > t_0$.

It is easy to see why STAR-FM has reduced accuracy when the coarse pixels are mixed, as the assumption of uniformly linear bias is called into question. ESTAR-FM [19] addresses this concern by delving into the two-input-pair model, where each coarse pixel is a weighted average of the finer pixels comprising it. ESTAR-FM then estimates pixel-specific ratios suggested by the linear changes in the two fine pixels around the prediction date. In the final model, this ratio is added into the formula for the prediction from STAR-FM, further tailoring the prediction to specific observed changes in the fine pixels.

One weakness of both STAR-FM and ESTAR-FM that emerges is that when modeling year-round changes, particularly when large blocks of data are missing over seasonal changes, the assumption of linear change from one date to the next may not be justified. It is precisely in situations like this, where a straight line segment seems insufficient to model the change that we would prefer to see a smoothly undulating curve filling the gaps.

2.3.2. Basic Algorithm

In the primary objective of this paper, we explore a method using only harmonic terms for a single year, although variations and other Fourier regression-based methods in the literature could ostensibly be used. The motivation for choosing one year is to facilitate comparison with STAR-FM. The goal here is to compare accuracy of prediction with STAR-FM, but it is also

desirable to use enough harmonics to demonstrate the utility in LU/LC classification as well. To address the secondary objective of the paper, we use five years' worth of Landsat data in a standard Fourier regression context. This is done to compare the results with those of the single-year analysis.

An explanation of the Fourier regression algorithm follows, akin to that found in [22]. For a pixel p measured at times $\boldsymbol{t} = (t_1, t_2, \dots, t_d)$ to have brightness values across a spectral band, b, given by the time series $\boldsymbol{b} = (b_1, b_2, \dots, b_d)$, we generate linearly interpolated fill points for gaps larger than a specified threshold g, producing combined vectors $\boldsymbol{t}^* = (\boldsymbol{t}, \boldsymbol{t_{fill}})$ and $\boldsymbol{b}^* = (\boldsymbol{b}, \boldsymbol{b_{fill}})$. The time vector may be assumed to be rescaled to the interval $[0,2\pi]$ without loss of generality. The linear interpolation prevents the models from producing nonsensical values in fitting the relatively sparse sections of the time series. Note that the resulting dates are not necessarily evenly spaced, since the object here is Fourier regression and not a transform. Further note that the linear interpolation is intended to be used for gaps considerably larger than the typical interval between Landsat data points, as the goal is to suppress wild oscillations that result from large gaps in the data. If these gaps cross seasonal changes or cover major likely features of the time series, then the Fourier regression algorithm will likely have reduced accuracy, just as STAR-FM may. The major distinction to be drawn here is that the linearly interpolated points are used as supplemental training data for the regression algorithm, as opposed to being the predicted values in and of themselves.

Once the gaps are filled, we use least squares estimation to estimate harmonic coefficients, that estimate denoted here by $\boldsymbol{a}$. In particular, we generate a model matrix using n harmonics as

$$T = (\mathbf{1} \quad \sin(\boldsymbol{t}^*) \quad \cos(\boldsymbol{t}^*) \quad \cdots \quad \sin(n\boldsymbol{t}^*) \quad \cos(n\boldsymbol{t}^*)) \quad (2.3)$$

Then the coefficient estimates can be obtained using the usual least squares method of

$$\boldsymbol{a} = (T'T)^{-1}T'\boldsymbol{b}^* \quad (2.4)$$

where T' is the transpose of T. Each pixel and each spectral band for that pixel has its unique set of coefficients, so the output of the algorithm is a raster of coefficients. The more harmonics the user desires, the more layers in the output raster. Note that there is a relationship between the

choice of g and the values of n. Smaller values of g result in more fill points being generated, so small g values support a higher number of harmonics.

In order to run this paper's algorithm for Fourier regression, the user must

1. Input Landsat image files (preferably cloud-free) that are rectified and subsetted to the same size,
2. Specify the number of harmonics desired (more harmonics = more detailed fit → greater chance of overfitting sparse data and getting "spurious oscillations"), and
3. Specify the gap threshold (the smaller the threshold, the more the Fourier regression fit will resemble a linear interpolation).

The algorithm works over each discrete time series of the entire area at once, outputting multiple rasters of coefficients.

2.3.3. Specific Application

Preprocessing of the Landsat scenes was performed in ERDAS Imagine 2010 and R version 2.11.1 [23]. Imagine was used to subset and combine the images, and R was used, applying both the caTools [24] and abind [25] libraries to perform dark object subtraction and generate binary files. Preprocessing for the MODIS images included using the MODIS Reprojection Tool [21] to subset, resample, and reproject the images using nearest neighbor resampling. R was then used to generate binary files. R was used, using custom code, to perform the Fourier regression algorithm. STAR-FM was implemented using Linux source code obtained from NASA [26]. For STAR-FM, the default values in the input files were used.

The specific dates in 2001 (as days of the year) for each of the image time series are given in Figure 2.3, as are the dates of the 1998-2002 data, shown as days of the year for compactness's sake. These dates apply to both scenes, as they are subsets of the same base image. The most obvious feature of the data is that Landsat is missing values from May through mid-September. This has some interesting ramifications for the analysis, since the quality of the Fourier regression depends in some sense on whether enough representative points are available. In the multi-year analysis, this problem is resolved by using multiple years and achieving a more even spread of points over the course of the days of the year. In the interest of comparing with STAR-FM, only one year is used, due to the considerable processing requirements of running STAR-FM on daily MODIS images over the course of multiple years. Because the primary object of comparison is NDVI time series, for the type of land cover considered here the missing summer

months can ideally be represented as long as data are available for the peak of the greening in late spring and the start of senescence in autumn for the classes of land cover considered in this paper.

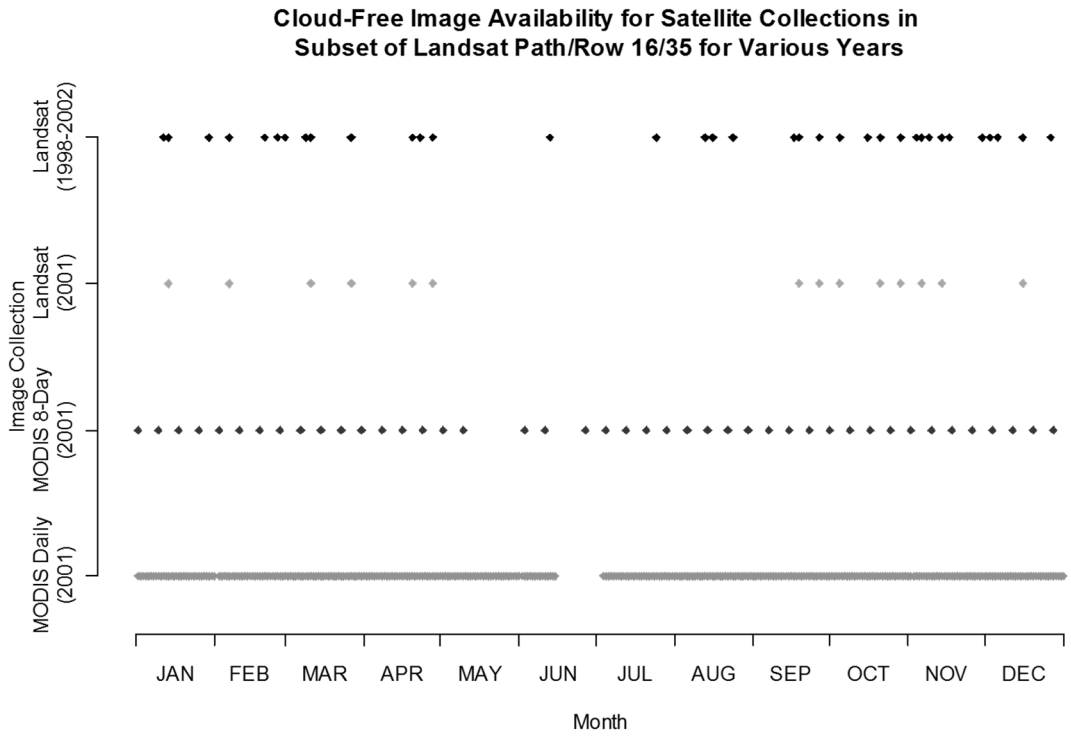


Figure 2.3. Distribution of image dates for Landsat and MODIS.

Due in part to the large gap in the Landsat time series, the gap threshold for the single year analysis was chosen to be $g = 32$ days. As a result, six fill points were generated for the summer values on a line interpolating the endpoints of the gap. If the gap was smaller, a smaller value of g would have been chosen, but to do so in this case would invite excessive fitting to interpolated data. A larger value would invite nonsensical undulations in the fit. It would be possible to develop an automated rule based on the distribution of dates across the year and a desired number of harmonics to determine a value for g, but this was beyond the scope of this paper in applying Fourier regression to the particular study areas. For the multi-year analysis, g was set at 365 days, guaranteeing that there would be no linear interpolation at all in the multi-year analysis.

The number of harmonics n was chosen to reflect the nature of NDVI (vegetative index) and the object of the paper (comparison of fit and predictive robustness with STAR-FM). With 12 months in the year, the first four harmonics were chosen to allow for variation on a month scale. This concept is illustrated in Figure 2.4, where the basic harmonics are shown as a constant mean and increasing pairs of sines and cosines with unit amplitude. The months are delineated by dashed lines, and by observing the harmonics' behavior between each pair of lines, one can see that each month has its own unique "harmonic address". There are biannual trends represented in the first harmonics, triannual trends in the second harmonics, etc. The fourth harmonics allow changes on a monthly scale, although using the fourth harmonics alone would force such changes to recur every other month. By using all of the harmonics together, a detailed curve for the year can thus be obtained. In more sparse datasets, there is a need to choose fewer harmonics to avoid generating misleading undulations, as use of four harmonics will compel the curve to detail monthly changes, even if there are not enough data to support them.

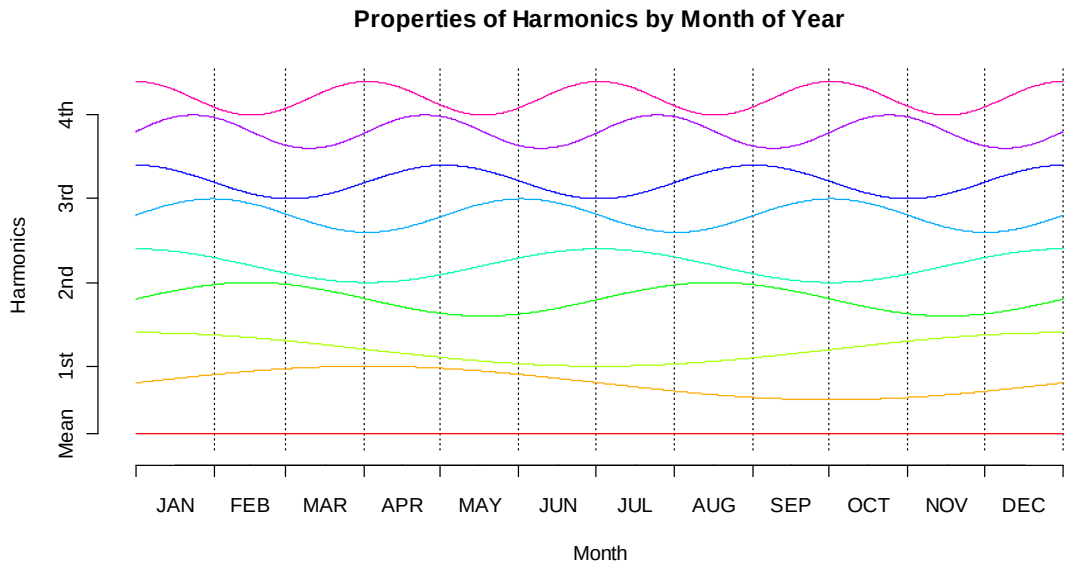


Figure 2.4. Intra-annual trends that may be captured by different harmonics in Fourier regression.

As a more concrete example, Figure 2.5 shows the effects of increasing harmonics on a sample time series. With only the first harmonic, the rough shape of the time series is outlined, but little else is fitted well. Increasing the harmonics shifts the peak of the curve towards the last known point before the gap in April. By the time 6 or 7 harmonics are used, minor details in the

raw time series are fitted more closely. The effect of the linear constraints in the gap can be seen in the higher harmonics, as well. For comparison, the typical deciduous forest NDVI time series can be thought of as "mesa-like" [27] with rapid rises and drops during the spring greening and fall senescence, respectively. The curve in Figure 2.5 certainly fits that description.

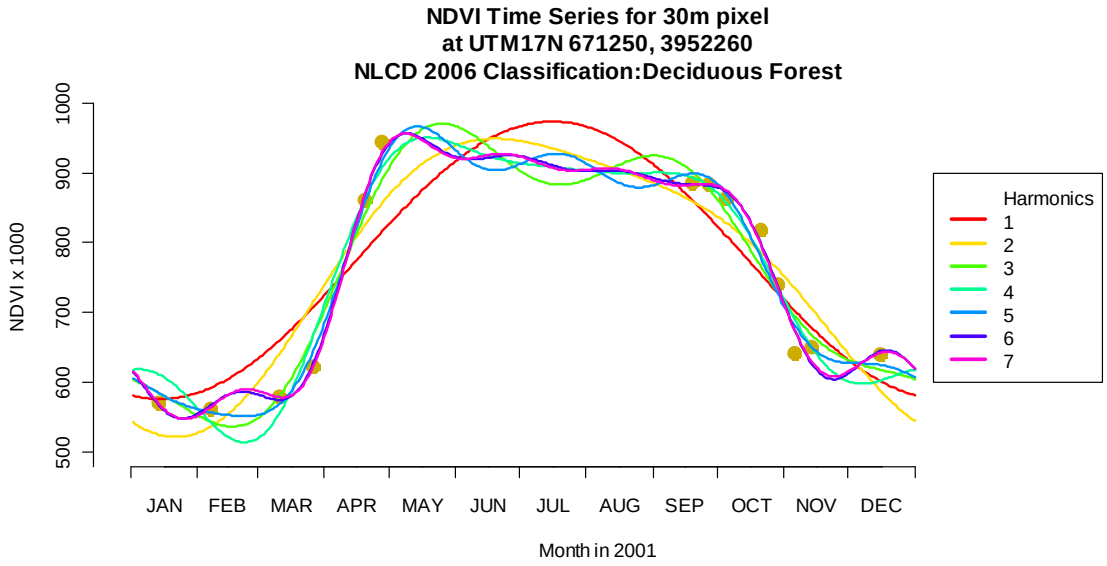


Figure 2.5. Effects of increasing the number of harmonics at a sample pixel.

2.4. Analysis

2.4.1. Comparison to STAR-FM

Since the primary objective is to compare the accuracy of Fourier regression with STAR-FM, some sort of validation data are needed. In this case, only cloud-free Landsat scenes were selected, and all algorithms were run with the assumption that every scene truly reflected conditions on the ground. The Fourier regression was performed on the NDVI values derived from the Landsat scenes (augmented with the six interpolated missing values for the summer), as well as on each of the six spectral bands independently. STAR-FM was run on both the daily and 8-day MODIS images, using two input pairs where possible. STAR-FM accepts inputs on a band-by-band basis, so it was run for both the red and NIR bands (MODIS bands 1 and 2, respectively) separately before combining the outputs to compute the NDVI.

Fourier regression was run only on the Landsat data, and STAR-FM was run using both the Landsat and MODIS reprojected data. Because the objective was to produce a good fit on vegetation index data over the course of 12 months, the Fourier regression was run, using the six interpolated fill values over the summer months, with four harmonics (potentially allowing

month-by-month variations in the basic curve). Fourier regression was applied to the Landsat NDVI and to each of the six Landsat bands independently. Fourier regression was not used on any of the MODIS data. STAR-FM was applied using both the daily MODIS imagery and the 8-day composites in conjunction with the Landsat scenes, effectively allowing for a 3-way comparison between Fourier regression and the two shades of STAR-FM.

In all cases, only algorithm outputs corresponding to the known Landsat dates were considered for the checking of fitting and predictive accuracy. The interpolated values were used in stabilizing the curve throughout the summer months, but those values were not checked for accuracy as there were no Landsat data available to check them against. In order to check the predictive accuracy, deleted residuals were calculated. In the general regression context where **y** is the response vector, for any point i, let $\hat{y}_{(i)}$ be the predicted value for the i^{th} point from the model generated by all points except the i^{th} point. Then the deleted residual is defined as $\hat{e}_{(i)} = y_i - \hat{y}_{(i)}$. Deleted residuals are desirable here because the interest is in the algorithms' abilities to predict values for missing dates, in addition to accuracy of overall fit. For the Fourier regression, this was achieved by removing each point one at a time and then implementing the fill interpolation each time before estimating coefficients and calculating the deleted residual. For STAR-FM, input pairs were used around the target date, using only the MODIS scene at the missing date to make the deleted prediction.

For accuracy of fit, the standard measures of root mean square error (RMSE) and R^2 may be used. For prediction, summing the squares of the deleted residuals gives the prediction sum of squares (PRESS) statistic, $\sum_{i=1}^{d} \hat{e}_{(i)}{}^2$. These PRESS statistics can then be compared (lower values imply greater overall accuracy), or alternately their corresponding predicted R^2, denoted by $R^2_{predicted}$, can be compared instead via the formula

$$R^2_{predicted} = 1 - \frac{PRESS}{Sum\ of\ Squares\ Total} \quad (2.5)$$

Due to the large number of pixels analyzed, violin plots [28] are a useful way to describe the results without resorting to single-number summary statistics. A violin plot may be thought of as the hybrid child of a boxplot and a continuous histogram. While the median, quartiles, and trimmed extremes are preserved as in a boxplot, a density estimation method is applied to the

data to generate a continuous curve. This curve is rotated and given symmetry, producing a "double" effect. The thickest parts of the plot correspond to the parts of the distribution which are most densely populated. These plots were used in summarizing the resulting statistics for the scenes.

2.4.2. Multi-Year Analysis

In the case of the second objective, for simplicity only the Landsat NDVI time series were considered. As in the single-year analysis, only cloud-free dates were chosen. The extra years did fill in the summer values missing from the 2001 data, as was shown in Figure 2.3. In order to perform Fourier regression, the dates for the scenes were converted into days of the year for the combined meta-year made from superimposing all the days of the year from the data. Additionally, the dates were also recorded as days counted from the beginning of 1998, allowing for the possibility of including polynomial terms in addition to the Fourier regression terms in the event a researcher wishes to use nonclassical harmonic analysis. Such a method would be well-suited to looking for interannual growth or decline trends over time.

As mentioned earlier, the multi-year analysis was performed using no linear interpolation whatsoever, as the gaps between points in the meta-year were much smaller. This analysis is pure harmonic regression. As before, R^2 values were calculated from the resulting fits. These values were compared to the R^2 values obtained from applying Fourier regression to the single-year data.

2.5. Results

A point worth noting is that the results which follow represent all of the LU/LC classes grouped together. Stratification along LU/LC lines was performed, but generally the results were so similar that the additional separation was not represented here. This is true of both the fitted and predictive results, as well as of the interannual analysis. The major exception to this rule was water pixels, which had wildly diverging NDVI patterns depending on the dates.

2.5.1. Fitting

The main results of fitting accuracy, the fitted R^2 and the RMSE, are shown in Figure 2.6. The units for the RMSE are in NDVIx1000, so an RMSE of 50, for example, implies that the standard deviation of observed values about the fitted values is about 0.05. It must be stressed that in the calculation of the resulting statistics, only points corresponding to the known Landsat dates were considered. The interpolated fill points generated by the algorithm served to stabilize the curve but were not used directly to calculate R^2 or RMSE.

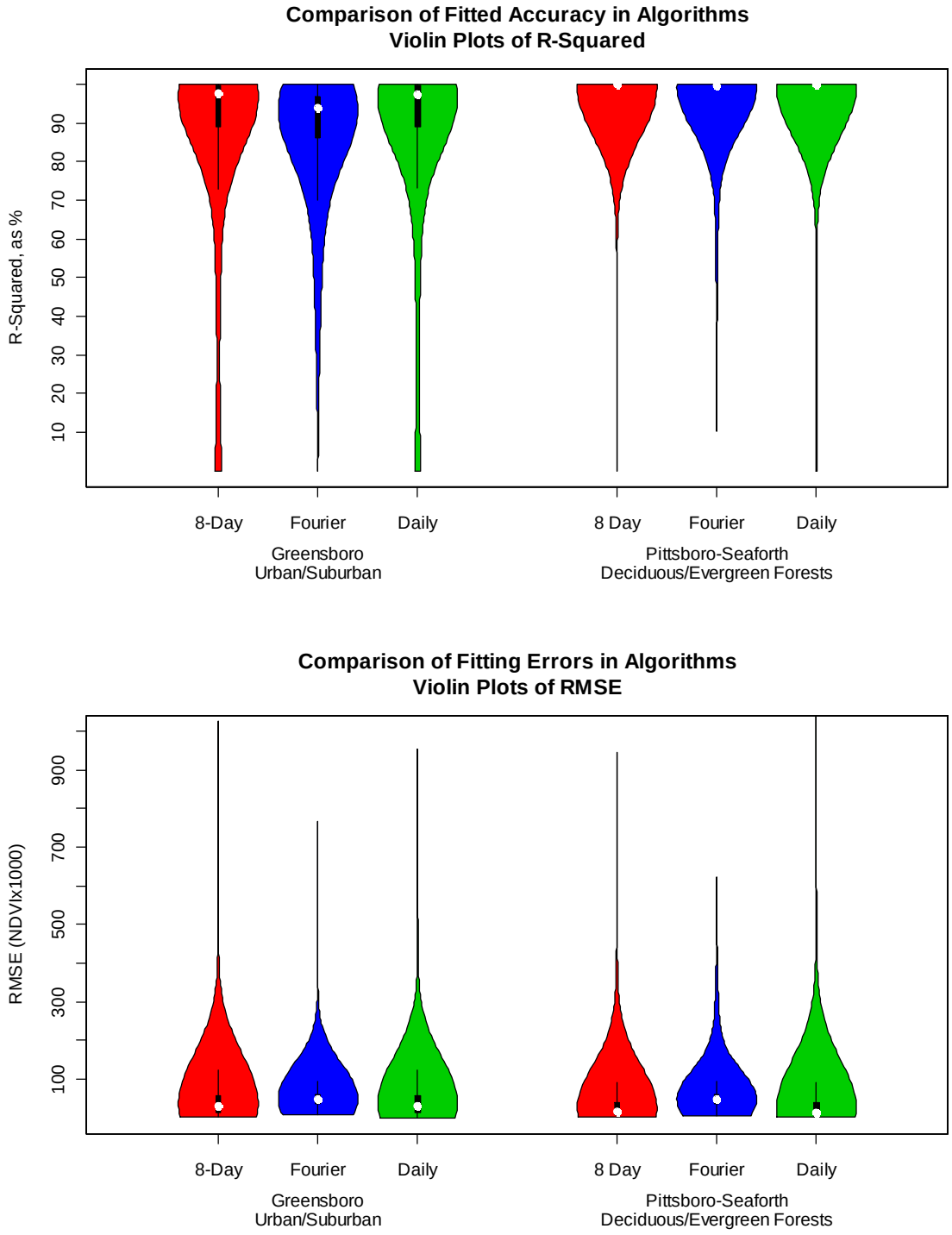


Figure 2.6. Distributions for fitting statistics.

Figure 2.7a shows a sample pixel's time series and the results of all three algorithms fitted to it. The cloud of STAR-FM points from the daily MODIS images illustrates the issue of cloud cover on MODIS pixels. A good number of pixels (about 46% of them) fall well outside the trend depicted by the raw data and the other algorithms. These dates are unsuitable for use in any sort of interpolation. Even the 8-day composite images are susceptible to this issue, as shown by the outlier point in late June. This is not a failing of STAR-FM, and these points were not used in calculating the fit and predictive statistics due to the fact that the Landsat scenes were chosen to be cloud-free, but it does illustrate the issue of cloud cover and how fusion methods must contend with it. On the other hand, the STAR-FM fits rise beyond the Fourier regression fits over the summer months, indicating that the last known Landsat point was before the greening for that year had been completed. The Fourier regression curve faithfully follows the linearly interpolated Landsat data, owing largely to the constraints placed on it by the choice of the gap threshold. In terms of comparison to the Landsat points, Fourier regression has the highest R^2 value, but it could have benefitted from at least one date in the summer.

Figure 2.7b shows a deciduous forest pixel, one in which the NDVI appears to remain fairly linear over the course of summer. The time series details a vegetative curve over the course of a year without disturbance. The trend appears smooth and undulating with a yearly period, precisely the sort of situation for which Fourier regression is suited. As in the previous figure, Fourier regression fits the known Landsat data tightly without any outlier issues, and it does so despite the fact that four months' worth of data are missing. The accuracy of the summer months—in which the fit is based primarily on the linear interpolation—cannot be determined, but the agreement of the curve with STAR-FM's output for the 8-day composites, particularly the early summer, is heartening.

Figure 2.7c shows another pixel in which the raw Landsat series contains some problematic points. While both images were restricted to dates that were as cloud-free as possible, there were some pixels that suffered from haze or shading. This one, near a water body, also mixed land reflectance with that of the water's surface, which had a deleterious effect on the flow of the series (i.e., one of the assumptions in the Fourier regression algorithm was violated). As a result, all three algorithms suffer in accuracy, but the STAR-FM points are better able to cope with the rapid swings of the time series.

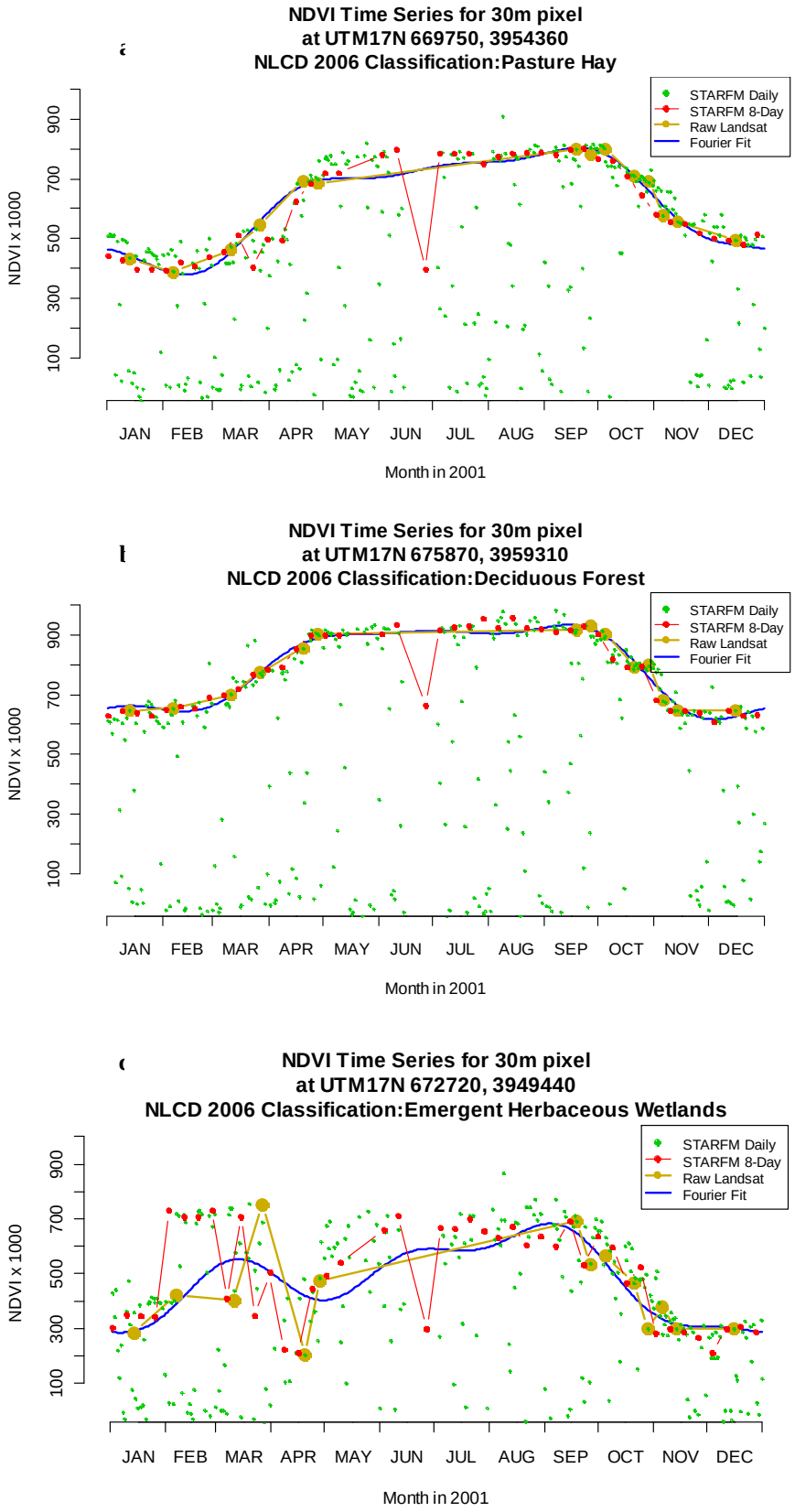


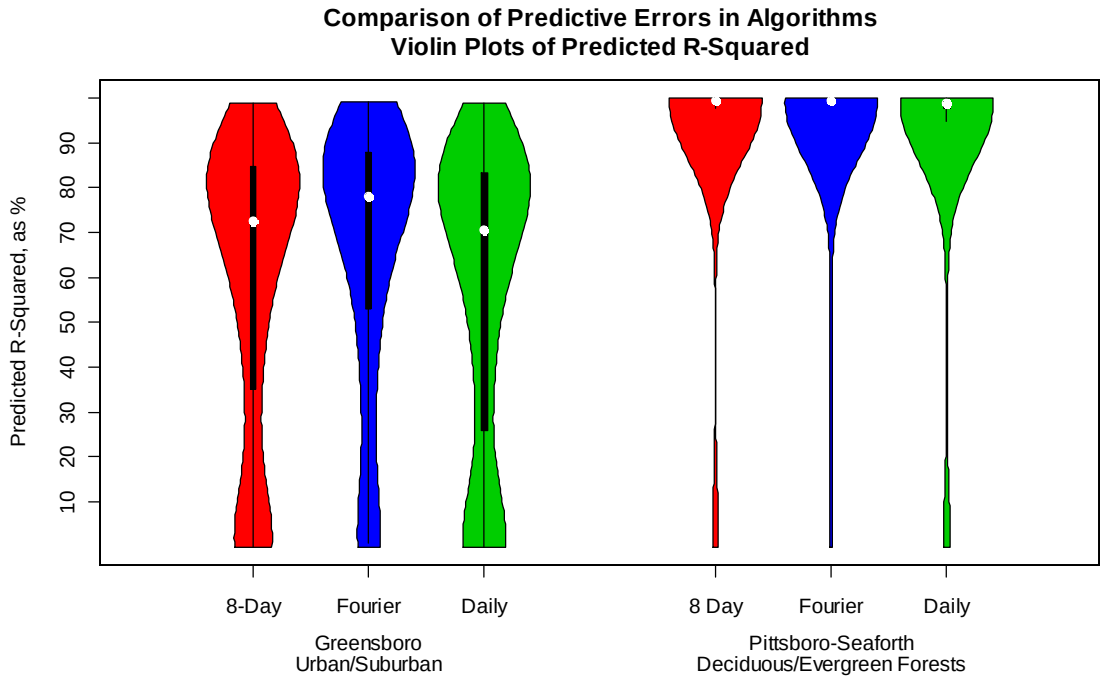
Figure 2.7. Example time series and algorithm fits.

From the violin plots of Figure 2.6, all three algorithms fit the Landsat data quite nicely most of the time. Note that in the Greensboro area the Fourier regression algorithm, while having a slightly lower median R^2 at 93.9%, also has thinner tails than the STAR-FM values and thus has a higher mean R^2 at 87.6%. The median values in the Pittsboro area are all above 99.8%, but note that Fourier regression again has the smallest "poor fit" tail of the three algorithms. The conclusion to be drawn from Figure 2.6 is that all three algorithms actually fit the known data quite well.

2.5.2. Prediction

The results of the predictive comparison are shown in Figure 2.8. Again, only the known Landsat dates were used in calculating both $R^2_{Predicted}$ and the predictive RMSE. Since there are hundreds of thousands of pixels in each area, any statistical comparison of means and medians will produce uninformatively significant results, but a visual inspection of the plots indicates the important features of the comparison. Clearly all three algorithms can be perturbed by missing data, but Fourier regression seems the least perturbed of the three. In particular, in the Greensboro area the STAR-FM algorithms suffer from higher predictive RMSEs than Fourier regression, including some truly extreme values. Generally, the algorithms did much better in the Pittsboro area, though upon checking, this was not due to a difference in land cover class distribution according to the land cover assignments made by the NLCD 2006 dataset.

The chief conclusion to be drawn from Figure 2.8 is that Fourier regression is more robust to missing data than STAR-FM, particularly in the relatively heterogeneous Greensboro area. This is somewhat surprising since STAR-FM had the benefit of ancillary data to compensate for missing values, but it speaks well for the assumption that the NDVI follows a curve that a Fourier series can appropriately model.



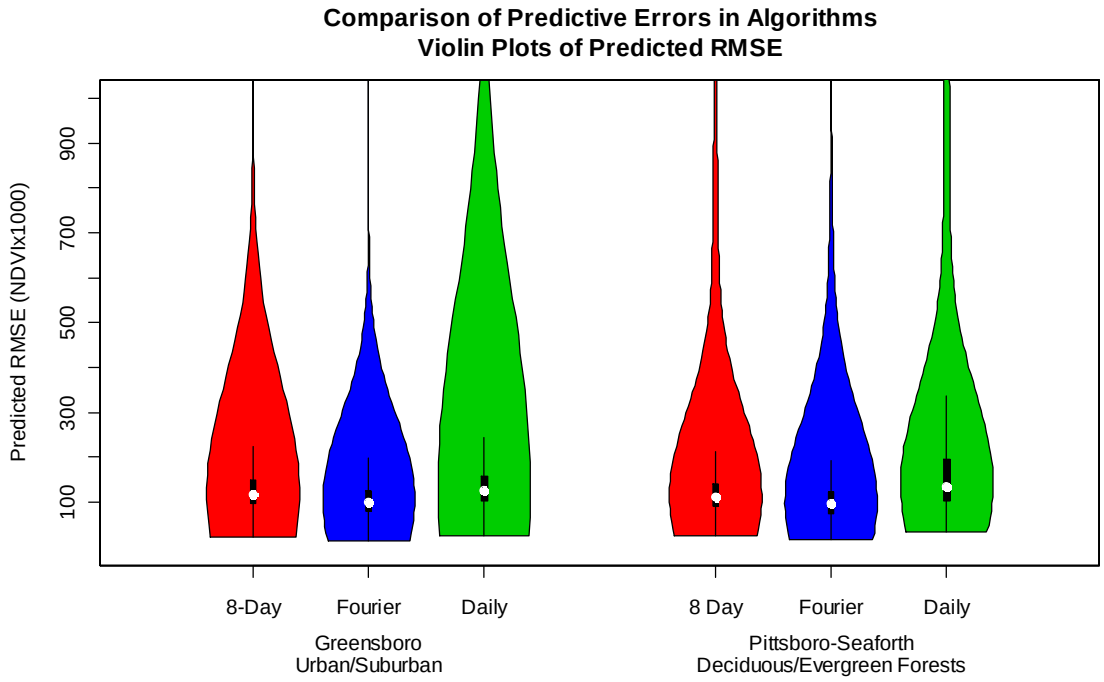


Figure 2.8. Distributions for predictive statistics.

2.5.3. Basic Landsat Bands

In addition to comparing the Fourier regression method to STAR-FM, Fourier regression was run on six Landsat bands: blue, green, red, near-infrared (NIR), and the two mid-infrared (MIR)

bands. The objective of this was to check the fitting accuracy and predictive robustness of Fourier regression when dealing with a non-index dataset, primarily for purposes of image generation or missing value imputation. The results of the analysis are shown in Figure 2.9.

It is immediately clear from Figure 2.9 that Fourier regression is more accurate in the infrared bands, both in terms of fit and in terms of prediction. In particular, the blue and green bands suffer dramatically in predictive R^2 to the point that the median values are below 70%. This is in part an artifact of the inherent band variability, as the variation in the visible bands tends to be much lower than that of the infrared bands, resulting in smaller total variation and thus lower R^2 values.

Despite the difficulties with the visible bands, it does appear that Fourier regression produces reasonably accurate facsimiles of the time series of basic Landsat bands, though those predictions are somewhat less robust than originally anticipated.

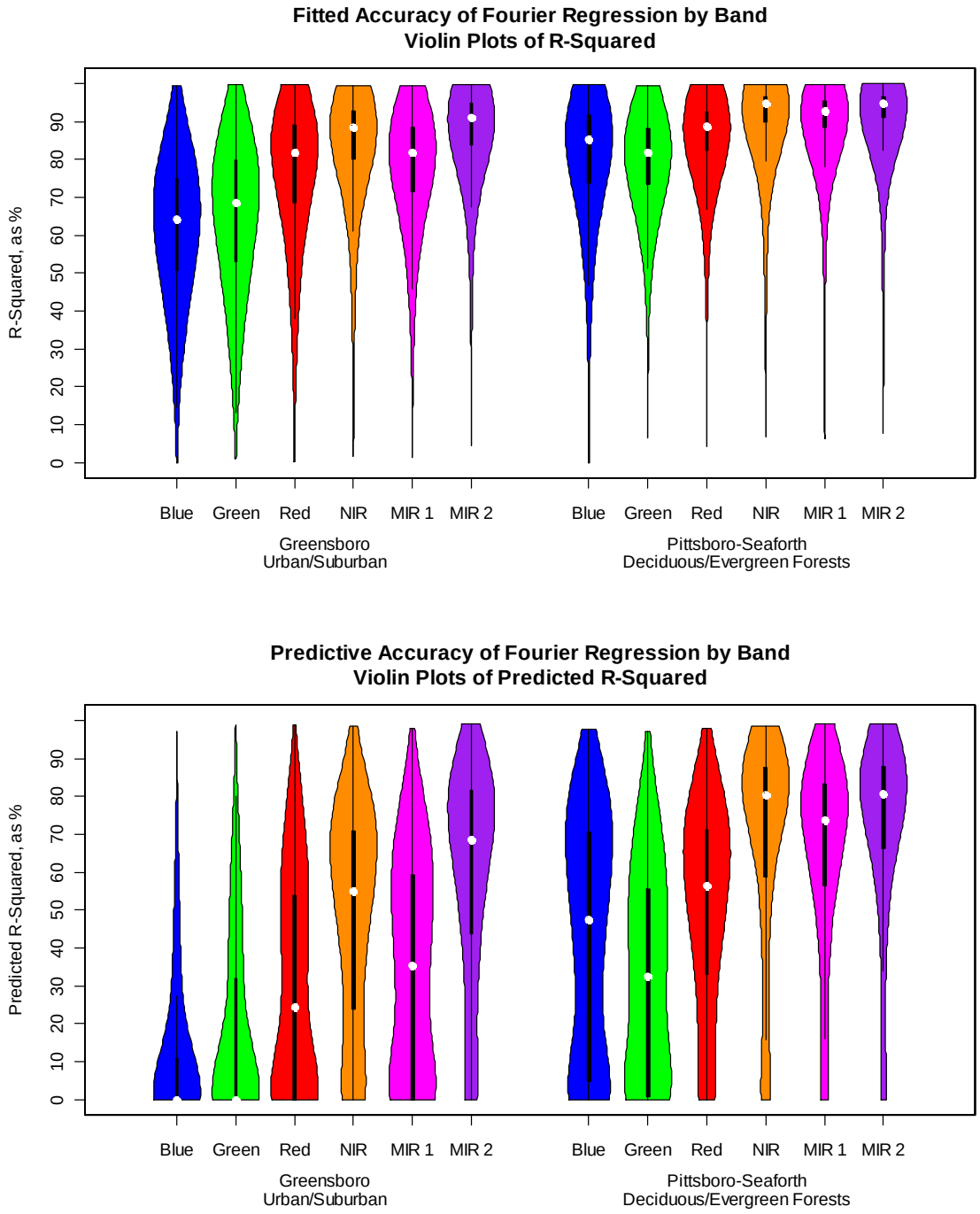


Figure 2.9. Distributions of fit and predictive statistics across Landsat bands.

2.5.4. Multi-Year Analysis and Comparison

The main results of the comparison between the 2001-only data and the data from 1998-2002 are shown in Figure 2.10. The chief interpretation is that using extra years greatly improved the

fitting accuracy of the method for pixels in the urban/suburban Greensboro area, while the extra years actually reduced the lower end of the fitting accuracy in the forested Pittsboro-Seaforth area. This reduction is not severe, as the 20th percentile of the multi-year data is still at 99.2%, compared to the 20th percentile value of 99.7% for the 2001-only data. The vast majority of the data are well above an R^2 of 99%.

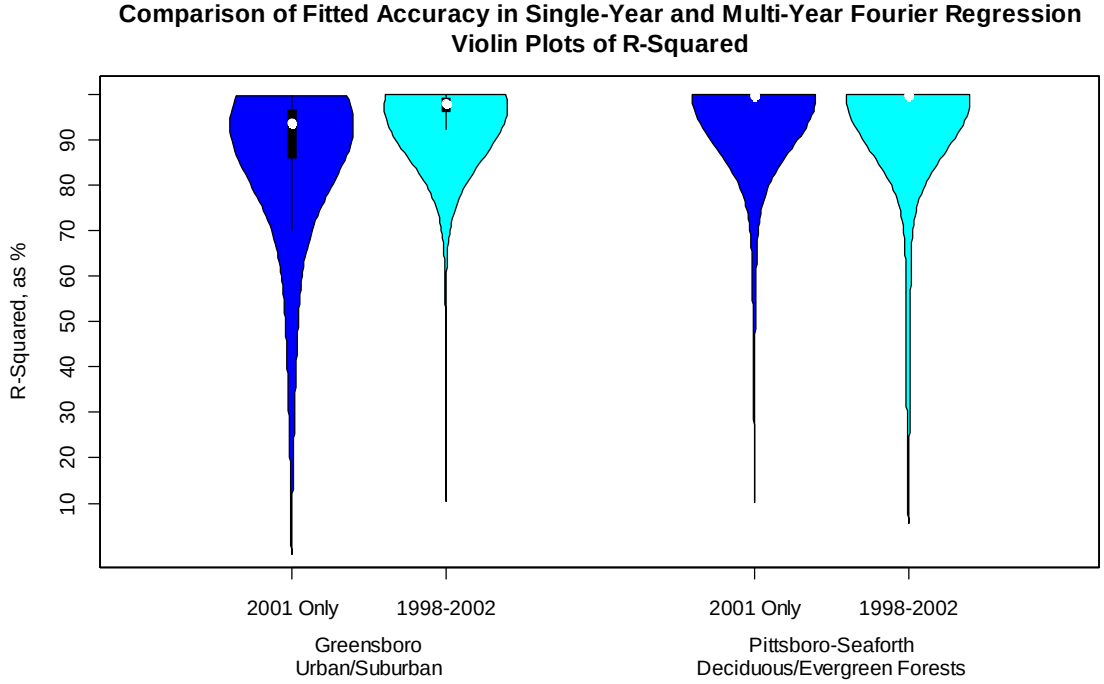


Figure 2.10. Distributions for fitted $\boldsymbol{R^2}$comparing single-year and multi-year approaches.

To further demonstrate the potential of multi-year analysis, information regarding a nicely-fitted pixel is shown in Figure 2.11. The pixel in question comes from the forested Pittsboro-Seaforth area. It is classified by the 2006 NLCD dataset as deciduous forest. Figure 2.11a shows the meta-year generated by superimposing days of the year from 1998 through 2002. It is easy to see that gaps from the 2001-only analysis are filled by values in the other years, eliminating the need to use a linearly interpolated fill algorithm before performing Fourier regression. The regression curve balances out the years' data well, resulting in a fitted R^2 of 99.8% on the known data points. In Figure 2.11b, the curve is redrawn over the course of the years, showing the way in which the yearly periodicity of the NDVI is captured by the curve. In particular, when comparing Figure 2.11b and Figure 2.11c, there is a slight increasing trend in

the NDVI series off the curve suggested over the years 2000 through 2002. This could be tested by a simple linear regression model to determine whether the coefficient is statistically significant. The residuals in Figure 2.11c are the key to applying Fourier regression in hopes of detecting disturbances and trends over time using multi-year Landsat data.

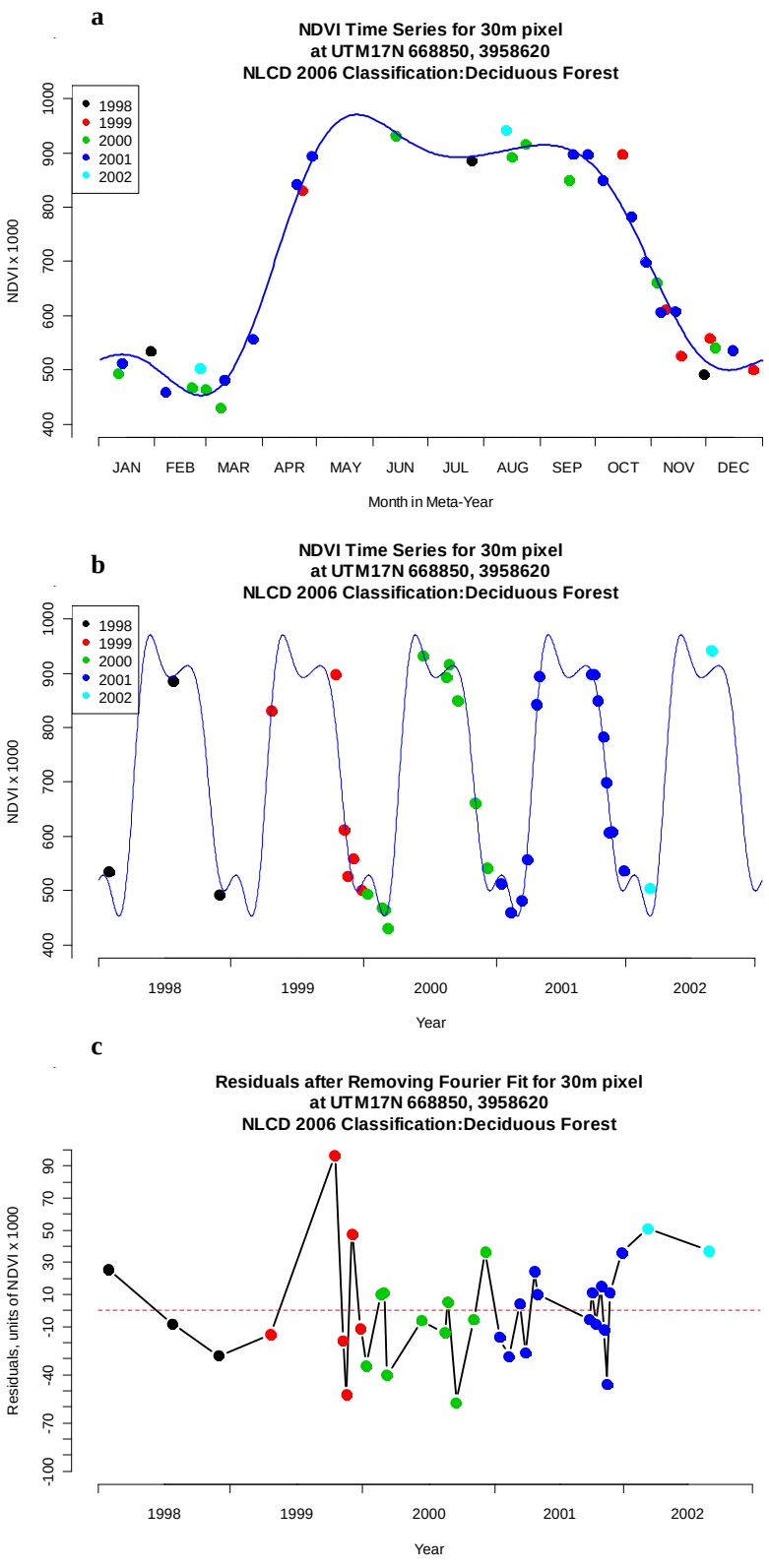


Figure 2.11. Multi-year analysis of a specific pixel, with residual time series.

With dates available throughout the year, there was an opportunity to see whether particular parts of the year, such as summer or winter, were better predicted by Fourier regression than other parts of the year. This was checked by calculating the residual values left over from subtracting the Fourier regression's fitted values from the known Landsat values and then summarizing the residuals across the forested Pittsboro-Seaforth study area by day of the year. It was assumed that the urban/suburban Greensboro area would have been of less interest from a phenological point of view. Figure 2.12 shows the resulting interquartile ranges by day of the year.

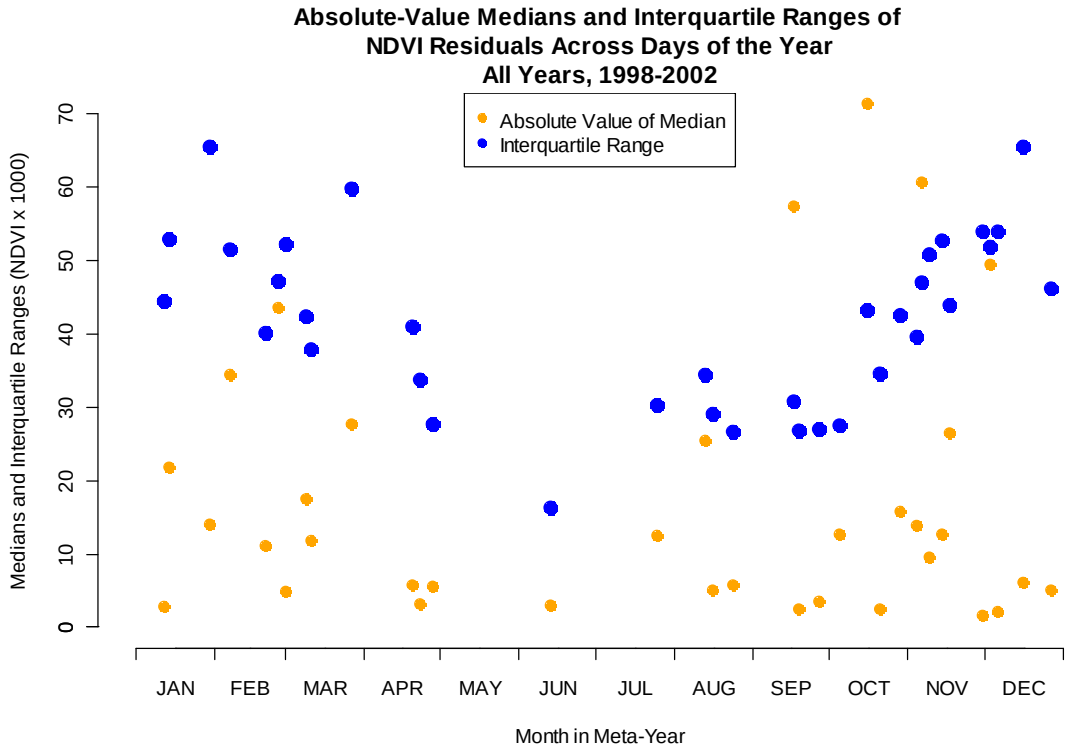


Figure 2.12. Summary statistics of fitted residuals, by day of year, for the forested Pittsboro-Seaforth area.

It is very interesting to note that the one season in which both the accuracy and range around the median were minimized was summer. This could be due to a number of factors. Firstly, the presence of snow in the winter months could have effects similar to cloud cover in calculating NDVI values, causing more error and wider variability around the expected values. Although the data had been screened to remove cloudy dates, they had not been screened to exclude all dates with snow on the ground. Secondly, the NDVI time series tend to peak in early summer

and remain high until the fall. Any curves exhibiting this basic trend, especially where the NDVI values are near to 1 anyway, would be likely to be close to the real values by virtue of the curves' construction. That being noted, it is informative to note that the curves had higher error in the transition seasons of spring and fall than in summer. It is also worth noting that this higher error is usually not in itself great, on the order of 0.05 to 0.10 in NDVI.

2.6. Conclusion

The results pertaining to the primary objective of the paper show that, for the types of land cover studied here, Fourier regression and STAR-FM are indeed comparable in the "middle ground" or a single-year analysis. In some sense, Fourier regression was put at a disadvantage in this analysis by using only a single year to train the data. In the results pertaining to the second objective, Fourier regression was performed using five years' worth of Landsat data, having the effect of improving Fourier regression's accuracy overall. The residual values left over after the interannual analysis are of considerable interest in their own right, opening avenues for change detection methods and trend observations over time for each pixel. There is potential for an "on-the-fly" disturbance detection, using previous years' data to check whether an incoming scene matches the expectation via a statistical method.

A key advantage of Fourier regression that emerged in the study was its ability to reduce storage and processing requirements. Suppose, for example, that one wished to generate data to cover a full 365-day year at daily temporal resolution for a scene comprising 1GB of data. Instead of generating 365 individual images (and requiring 365GB to save them), one can instead save the Fourier harmonic coefficients and use them to generate the data as needed (e.g., pixel by pixel). Even with six harmonics plus a constant, this would result in only storing 13 rasters instead of 365. If the coefficients are converted to an integer format through multiplying and truncating, then the entire interpolation can be saved in only 13GB of space. If a multi-year study was desired, one could save a couple of polynomial coefficients to account for interannual trends in land cover at the cost of only 1 or 2GB, as opposed to another full 365GB, further compounding the savings. Figure 2.13 illustrates this idea. In the multi-year analysis done in this paper, no polynomial terms were added, so the storage costs for the five-year model were equal to the costs for the single-year model. From this example and the results of that analysis, one can

intuit that adding more dates into Fourier regression can only improve the quality of the estimated coefficients for the model at the same storage cost.

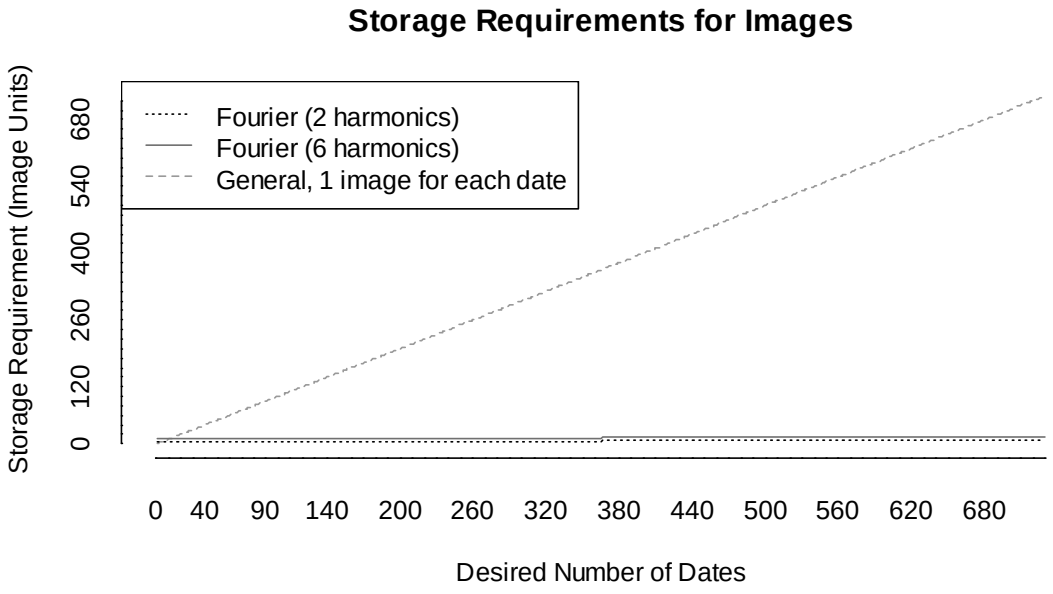


Figure 2.13. Storage requirements for interpolating Landsat data throughout a desired number of dates.

STAR-FM is not as well-suited to interannual analysis, owing to the need to generate images for each desired point in each year using multiple images to make each predicted image. Although the input images may be reused for different prediction dates, STAR-FM still requires nontrivial processing and space. Considering that Fourier regression has been shown to be comparably accurate, it does not seem efficient to use STAR-FM for this sort of analysis. On the other hand, if the time of interest is well within a single year, STAR-FM is not bound by the constraints of periodicity and does not require a full year of Landsat images to run. In such cases, STAR-FM is clearly a better choice than Fourier regression. Table 2.1 details some of the advantages and disadvantages of both methods.

Table 2.1. Comparison between Fourier regression and STAR-FM.

	Fourier Regression	STAR-FM
Advantages	• Robust, accurate prediction and fit • Reduced storage space (can save ~9 harmonic coefficients instead of ~350 prediction images) • No ancillary data • Suited for interannual studies • More harmonics = finer fit	• Robust, accurate prediction on cloud-free days • Availability of composite imagery • Able to handle sudden changes on a daily basis • Suited for intra-annual studies, especially for short duration
Disadvantages	• Must have input data at key points of curve • Harmonics limited by quantity of data • Requires at least one year of data • Produces undesirable "wiggles" • Fits poorly when pixel undergoes disturbance	• Nontrivial processing/computing requirements • Must generate images for each prediction date • Susceptible to cloud cover issues • Reduced accuracy in heterogeneous areas • MODIS has no blue band, only a blue-green

The ability to use smooth curves to represent yearly Landsat data makes many different forms of analysis possible. As examples of possible applications, one may use various curve features such as integral area, maximum/minimum, and the Fourier regression coefficient values as explanatory variables in regression models. These variables may then be tied to ground observations of biophysical parameters such as biomass, and from the resulting model one may estimate biomass for a given scene from the Landsat data. If a regression model requires the minimum or maximum value of a pixel over the year, the Fourier regression curves may be used to gain estimates.

The possibilities for application of a smooth periodic curve to represent changes in brightness values over time are legion. Only a few have been touched on in this paper, as the goal was to demonstrate a method for making such a curve and in comparing it to STAR-FM. However, any context in which at least one year's worth of Landsat data is available may make use of Fourier regression to fill in the missing values. The ultimate conclusion of this paper is that for the

purposes of annual and interannual time series analysis of Landsat scenes, particularly in regions similar to the eastern US, Fourier regression is a good choice for fitting the multitemporal curve.

2.7. Acknowledgement

We would like to thank Christine Blinn of the Virginia Tech Department of Forest Resources and Environmental Conservation for her help in implementing STAR-FM.

2.8. References

[1] Woodcock, C. E., and Ozdogan, M. (2004) "Trends in land cover mapping and monitoring." In Gutman (Ed.), *Land Change Science* (pp. 367−377). New York: Springer.

[2] Healey, S. P., Cohen, W. B., Yang, Z. Q., and Krankina, O. N. (2005) "Comparison of Tasseled Cap-based Landsat data structures for use in forest disturbance detection." *Remote Sensing of Environment*, 97(3), 301−310.

[3] Masek, J. G., and Collatz, G. J. (2006) "Estimating forest carbon fluxes in a disturbed southeastern landscape: Integration of remote sensing, forest inventory, and biogeochemical modeling." *Journal of Geophysical Research-Biogeosciences*, 111, G01006, doi:10.1029/2005JG000062.

[4] Masek, J. G., Huang, C. Q., Wolfe, R., Cohen, W., Hall, F., and Kutler, J. (2008) "North American forest disturbance mapped from a decadal Landsat record." *Remote Sensing of Environment*, 112(6), 2914−2926.

[5] Asner, G. P. (2001) "Cloud cover in Landsat observations of the Brazilian Amazon." *International Journal of Remote Sensing*, 22(18), 3855−3862.

[6] Jorgensen, P. V. (2000) "Determination of cloud coverage over Denmark using Landsat MSS/TM and NOAA-AVHRR." *International Journal of Remote Sensing*, 21(17), 3363−3368.

[7] Ju, J. C. and Roy, D. P. (2008) "The availability of cloud-free Landsat ETM+ data over the conterminous United States and globally." *Remote Sensing of Environment*, 112(3), 1196−1211.

[8] Carrao, H., Gonalves, P., and Caetano, M. (2010) "A nonlinear harmonic model for fitting satellite image time series: analysis and prediction of land cover dynamics." *IEEE Transactions on Geoscience and Remote Sensing*, 48(4), 1919-1930.

[9] Roerink, G. J., Menenti, M., and Verhoef, W. (2000) “Reconstructing cloud-free NDVI composites using Fourier analysis of time series.” *International Journal of Remote Sensing*, 21(9), 1911–1917.

[10] Moody, A. and Johnson, D. M. (2001) “Land-surface phenologies from AVHRR using the discrete Fourier transform.” *Remote Sensing of Environment*, 75(3), 305–323.

[11] Jakubauskas, M. E., Legates, D. R., and Kastens, J. H. (2001) “Harmonic analysis of time-series AVHRR NDVI data.” *Photogrammetric Engineering & Remote Sensing*, 67(4), 461–470.

[12] Geerken, R., Zaitchik, B., and Evans, J. P. (2005) “Classifying rangeland vegetation type and coverage from NDVI time series using Fourier filtered cycle similarity.” *International Journal of Remote Sensing*, 26(24), 5535-5554.

[13] Hermance, J. F., Jacob, R. W., Bradley, B. A., and Mustard, J. F. (2007) “Extracting phenological signals from multiyear AVHRR NDVI time series: framework for applying high-order annual splines with roughness damping.” *IEEE Transactions on Geoscience and Remote Sensing*, 45(10), 3264-3276.

[14] Hermance, J. F. (2007) “Stabilizing high-order, non-classical harmonic analysis of NDVI data for average annual models by damping model roughness.” *International Journal of Remote Sensing*, 28(12), 2801-2819.

[15] Hay, S. I., Snow, R. W., and Rogers, D. J. (1998) “Predicting malaria seasons in Kenya using multitemporal meteorological satellite sensor data.” *Transactions of the Royal Society of Tropical Medicine and Hygiene*, 92, 12-20.

[16] Immerzeel, W. W., Quiroz, R. A., and de Jong, S. M. (2005) “Understanding precipitation patterns and land use interaction in Tibet using harmonic analysis of SPOT VGT-S10 NDVI time series.” *International Journal of Remote Sensing*, 26(11), 2281-2296.

[17] Tucker, C. J. (1979) “Red and photographic infrared linear combinations for monitoring vegetation.” *Remote Sensing of Environment*, 8(2), 127-150.

[18] Gao, F., Masek, J., Schwaller, M., and Hall, F. (2006) “On the blending of the Landsat and MODIS surface reflectance: predicting daily Landsat surface reflectance *IEEE Transactions on Geoscience and Remote Sensing*, 44(8), 2207-2218.

[19] Zhu, X., Chen, J., Gao, F., Chen, X., and Masek, J. (2010) “An enhanced spatial and temporal adaptive reflectance fusion model for complex heterogeneous regions.” *Remote Sensing of Environment*, 114(11) 2610-2623.

[20] Masek, J. G., Vermote, E. F., Saleous, N. E., Wolfe, R., Hall, F. G., Huemmrich, K. F., Gao, F., Kutler, J., and Lim, T.-K. (2006) "A Landsat surface reflectance dataset for North America, 1990–2000." *IEEE Geoscience and Remote Sensing Letters*, 3(1), 68-72.

[21] Dwyer, J., Weiss, J., Schmidt, G., Logar, T., Burrel, R., Stubbendieck, G., Rishea, J., Misterek, B., Jia, S., and Heuser, K. (2001) “The MODIS reprojection tool.” *American Geophysical Union*, Spring Meeting, abstract #U21A-24.

[22] Bloomfield, P. (2004) *Fourier Analysis of Time Series: An Introduction*. Wiley-Interscience, 2nd ed. ISBN-10: 0471889482.

[23] R Development Core Team. 2010. “R: A language and environment for statistical computing.” R Foundation for Statistical Computing. Vienna, Austria. ISBN 3-900051-07-0, URL http://www.R-project.org.

[24] Tuszynski, J. (2010). “caTools: Tools: moving window statistics, GIF, Base64, ROC AUC, etc.” R package version 1.11. http://CRAN.R-project.org/package=caTools.

[25] Plate, T. and Heiberger, R. (2011) “abind: Combine multi-dimensional arrays.” R package version 1.3-0. http://CRAN.R-project.org/package=abind.

[26] LEDAPS Tools website. http://ledaps.nascom.nasa.gov/tools/tools.html

[27] Hargrove, W. W., Spruce, J. P., Gasser, G. E., and Hoffman, F. M. (2009) “Toward a national early warning system for forest disturbances using remotely sensed canopy phenology.” *Photogrammetric Engineering & Remote Sensing*, 75(10), 1150-1156.

[28] Hintze, J. L. and R. D. Nelson (1998) “Violin plots: a box plot-density trace synergism.” *The American Statistician*, 52(2), 181-4.

Chapter 3: Detecting Forest Disturbances with Statistical Quality Control Charts from Landsat Data

An On-the-Fly Massively Multitemporal Change Detection Method

Evan B. Brooks[a], Randolph H. Wynne[a], Valerie A. Thomas[a], Christine E. Blinn[a], and John W. Coulston[b]

[a]Department of Forest Resources and Environmental Conservation, Virginia Tech, Blacksburg, VA, USA
[b]USDA Forest Service Southern Research Station, Forest Inventory and Analysis Unit, Knoxville, TN, USA

This chapter was submitted to IEEE Transactions in Geosciences and Remote Sensing in January 2013. Reviewers' comments were received April 18, 2013. This version is updated according to those comments and ready for resubmission.

Abstract

One challenge to implementing spectral change detection algorithms using multitemporal Landsat data is that key dates and periods are often missing from the record due to weather disturbances and lapses in continuous coverage. This paper presents a method that utilizes the residuals from harmonic regression over years of Landsat data, in conjunction with statistical quality control charts, to signal subtle disturbances in vegetative cover. These charts are able to detect changes from both deforestation and subtler forest degradation and thinning. First, harmonic regression residuals are computed after fitting models to interannual training data. These residuals are then analyzed using methods of statistical process control; namely, the time series are subjected to Shewhart X-bar control charts and exponentially weighted moving average (EWMA) charts. The Shewhart X-bar charts are also utilized in the algorithm to generate a data-driven cloud filter, effectively removing clouds and cloud shadows on a location-specific basis. Disturbed pixels are indicated when the charts signal a deviation from data-driven control limits. The methods are applied to a collection of loblolly pine (*Pinus taeda*) stands in Alabama, USA. The results of the analysis are compared with stands for which known thinning has occurred at known times. In the case of thinning, the method yielded an overall accuracy of 85%, with the particular result that it provided afforestation/deforestation maps on a per-image basis, producing new maps with each successive incorporated image. These maps matched very

well with observed changes in aerial photography over the test period. As a result, the method is highly recommended for on-the-fly change detection, for changes ranging from clearcuts to plantation growth.

3.1. Introduction

3.1.1. Background

Detection of disturbances and changes in forest cover is a major application of remote sensing of the Earth. The simplest method of detecting changes from one time to another is to literally compare two images from the same area and measure the changes, often signaling changes outside some threshold. Depending on the features in the scene and the timing of the images, this method may be used to identify major changes such as clearcuts, but it is difficult to identify minor, yet still important, changes such as forest degradation or thinning. [1-4] While small changes may appear negligible on a pixel basis, on a cumulative basis they can be the most important feature of change across a scene. By and large, such subtle changes have gone unreported [4] while results of fires and wholesale logging are reported. This accounts for some of the uncertainty in forest mass estimates, for example. [1] The difficulty in detecting subtle changes is mostly due to the spatial resolution of the sensor typically being coarser than the resolution of such minor changes. Subtle changes in forest cover can also affect the spectral characteristics of the area in question, although these changes may be very small and difficult to detect on an image-to-image basis.

In addition to standard bi-temporal change detection methods, there are a variety of other methods which make use of multiple images, often over many years' worth of data, to improve accuracy, resolution, or to provide an ongoing record. These methods are best characterized as "massively multitemporal" in their nature, and they rely in large part on the free availability of Landsat data as of October 2008. Trajectory-based image analysis would be one subset of this class of methods. One corporate example of this paradigm is the persistent change detection used by the MDA Federal Company. [5] In this method, changes are only registered if a certain number of consecutive images before the candidate change are significantly different from a consecutive number after the change.

Some more examples of massively multitemporal change detection methods utilizing large Landsat time series applied to forestry follow. The Vegetation Change Tracker (VCT) [6-7] uses

the concept of normalized values for pixels based on the mean and standard deviation of known forest pixels from the same scene, using multiple images to improve the forest/non-forest classification. This concept has been used to detect and track changes in forest cover over several regions in the US [6], such as Mississippi [8]. In another vein, LandTrendr [9-10] is an automated trajectory-based image analysis algorithm. This algorithm allows users to automatically segment and identify different trends in land cover change for pixels over time. While not specifically designed to signal disturbances as they occur, the vertices in the segments allow for an easy classification of trends in time series, representing shifts from one stage of growth to another. The CLASlite software [2-3] applies its algorithm to a variety of satellite sources in a user-friendly format to produce forest change maps in tropical regions. This free software is distributed by the Carnegie Institution for Science to governments around the South American continent for monitoring of tropical deforestation and degradation, including selective logging.

The interest in developing massively multitemporal change detection methods continues to increase. As a prime example, during the time of drafting this paper, another method was published by Zhu et al. [11] This publication merits special interest here because of the similarity in initial approaches, to be discussed in later parts of this paper. Works such as [5-11] and this paper are evidence of a change in paradigm for change detection, transitioning from comparisons of temporally distant images to looking for deviations from model predictions.

While these other methods exist and are effective, there is still room for a simple algorithm which can produce change maps for a scene, updated from previous maps, with each new image of that scene. This sort of “on the fly” methodology would allow for a continuous monitoring paradigm to be implemented, rather than one of yearly summaries. Our goal in this paper is to introduce such an algorithm, based on a well-established statistical methodology used in other disciplines. We note here that [11] has also produced an algorithm that can incorporate incoming data, using different methods for determining when a pixel has undergone significant change. One clear distinction between the methods is that in this paper, images are considered to be a quasi-systematic sample, the derivatives of which are then used as inputs in statistical process control tools.

In previous work [11-12], it was shown that Fourier regression may be used to generate smooth curves fitted to interannual Landsat data. In particular, by collating multiple years’

worth of Landsat data, one can take advantage of the fact that the Landsat satellites do not have precise yearly return times to "fill in" even more days of the year. This allows one to simulate weather-free conditions on any given day of the year, with good accuracy, even if cloud cover prevents direct use of some images.

It was further shown that the residuals left from subtracting the curve from the data produce a record of shifts from the expectation in the time series development. Two things thus become apparent from this record. If the pixel remains stable over the course of the years, then an extremely good fitted curve will result in terms of accuracy and detail. If, on the other hand, the pixel undergoes changes or disturbances, either in the context of catastrophic disturbance or subtle changes due to forest growth or climate, then the residuals left over from subtracting the fitted "averaging" curve will produce a profile of the changes. Using this profile, it becomes possible to use incoming scene data to detect disturbances to the pixels, in a form of real-time environmental monitoring.

In the areas of industry and manufacturing, there has naturally been great interest in the idea of real-time monitoring of processes. Accordingly, there is a field of statistics, quality control, which addresses this interest through the creation of statistical tools for the purpose of actively monitoring processes. Key among these tools are *quality control charts*, which take systematically measured data (though not necessarily regular data) and *signal* the operator in the event that the monitored process goes out of control. For reference, a brief explanation of the quality control charts used in this paper follows.

3.1.2. Shewhart Charts

Shewhart charts, originally developed by Walter A. Shewhart in 1924 [13], are the foundation of the idea of statistical control charts. They have historically been used as the standard to test newer control charts against [14], and while they have drawbacks, their simplicity and versatility makes them reasonable to use even today. Consider a time sequence $t_1, t_2, \ldots$ with $t_n < t_{n+1}$ and associated measurements from a process, $x_1, x_2, \ldots$, where the measurements are assumed to be independently normally distributed with mean μ and standard deviation σ. If the mean of the process remains at μ, then one should expect the standardized value $z = \frac{x-\mu}{\sigma}$ to be within 3σ about 99.7% of the time. If the value is beyond this, it may be

evidence that the process mean has shifted. In general, a Shewhart X-bar (or $\bar{X}$) chart at time t_n is given by

$$z_n = \frac{x_n - \mu}{\sigma} \quad (3.1)$$

This chart *signals* (note the use as a verb) if the value moves outside the range $\pm L\sigma$, where $L\sigma$ is considered the generic *control limit* of the chart. An example of a Shewhart X-bar chart and its lower control limit is shown in Figure 3.1. The pixel in the figure is in a forested area that is stable, suggested by the lack of any clear trend in the residual values in the plot. Due to the statistical properties of the sample, the chart is bound to signal eventually, even if the process remains in control, but the control limits may be set so that this happens only rarely.

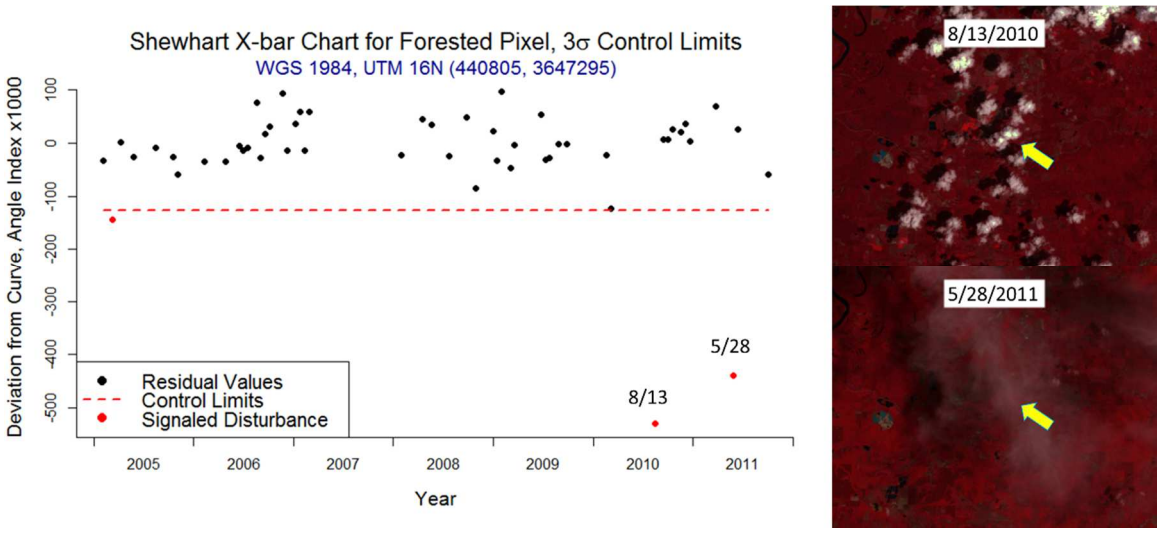


Figure 3.1. Shewhart X-bar chart for residual values after removing seasonality. The angle index is a vegetation index, with higher values corresponding to denser vegetation. Signaled dates, outside the control limits, are in red. They correspond to dates for which this pixel (yellow arrow) was shaded by clouds, also evidenced by the subsequent rapid return to the trend.

Shewhart charts are good for detecting large process mean shifts, but due to the fact that they only use the present point they may miss smaller consistent shifts. [13, 15-16] They are ideally suited for signaling anomalies in the history, such as a passing cloud. Figure 3.1 offers an example of this property. The red values in Figure 3.1 correspond to dates for which this pixel was shaded by clouds. They do not represent a sustained disturbance because the subsequent values demonstrate a rapid return to the original trend. X-bar charts could also be useful for

quickly detecting that a forest has been clearcut, but they are not as useful for detecting a subtle thinning in forest cover.

3.1.3. EWMA Charts

Again, consider a time sequence $t_1, t_2, ...$ with $t_n < t_{n+1}$, starting at $n = 0$,and associated measurements from a process, $x_1, x_2, ...$, where the measurements are assumed to be independently normally distributed with mean μ and standard deviation σ. Then for a *tuning parameter* $0 < \lambda < 1$, the EWMA chart [15-16] for the process at time t_n is given by

$$z^*_n = (1 - \lambda)z^*_{n-1} + \lambda x_n \quad (3.2)$$

Thus, the chart value for a given time is a function of the entire history of the chart. The extent to which this history is utilized is characterized by the value of λ, which determines how retrospective the chart is. Values of λ close to 1 will result in a chart that assigns little weight to previous values (the extreme case, $\lambda = 1$, yields a Shewhart X-bar chart after standardization by μ and σ), whereas a value of λ close to zero relies primarily on historical data. Smaller values for λ may be useful when the variation in process data is very great relative to the shift being detected (low signal to noise ratio). In such a case, incoming values should be tempered against the previous trend to avoid frequent false signals. An example of this chart is shown in Figure 3.2.

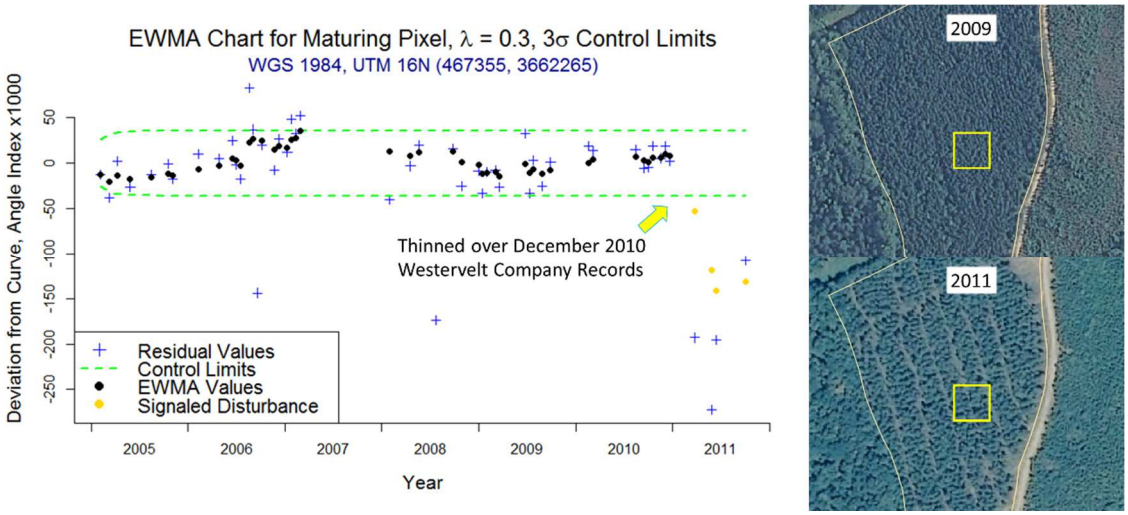


Figure 3.2. Exponentially Weighted Moving Average (EWMA) chart for residual values after removing seasonality. The pixel in question (yellow boxes at right) underwent a thin in December 2010, according to company records.

The chart signals when it exceeds the asymptotic control limits (*CL*) given by

$$CL = \mu \pm L_{EWMA} s \sqrt{\left(\frac{\lambda}{2-\lambda}\right)[1-(1-\lambda)^{2n}]} \quad (3.3)$$

Here, L_{EWMA} is the desired number of standard deviations for marking "out of control", and s is the estimated historical standard deviation of the data. Note how the control limit initializes at a lower value and approaches an asymptotic limit of $\mu \pm L_{EWMA} s \sqrt{\left(\frac{\lambda}{2-\lambda}\right)}$ as the chart continues to operate. As a consequence, it is fairly common for EWMA charts to experience "warm-up noise" in the form of false signals during the early training period. For practical purposes, this asymptote is very quickly achieved, typically in the first 15 to 20 samples taken. Thus, the control limits for EWMA charts, while appearing to grow continuously, may for all practical purposes be treated as constant limits.

(3.2) allows all past measurements to be used in the calculation, with greater weight on the recent ones. Of particular importance is the fact that the entire chart's history is encapsulated in a single term, the EWMA chart's previous value, z_{n-1}. Once that has been calculated, one may discard all the previous data and still compute the next value upon receipt of the next observation. This ease of future processing is the reason that EWMA charts are well-suited for

on-the-fly monitoring. Additionally, the running average aspect allows the EWMA chart to be fairly robust to the normality assumption, making it particularly appealing. Depending on how λ is set, the corresponding chart is fast with detecting small changes [15-16]. We anticipate that appropriately tuned EWMA charts will allow us to detect subtle changes caused by thinning of a forest or possibly changes in leaf area as a result of drought.

Based on the above reasons, EWMA charts represent an extremely useful tool for environmental monitoring via Landsat time series. There are different types of charts which are useful for detecting different sorts of disturbance. In the case of catastrophic change, a simple Shewhart chart would suffice. But for processes that show gradual change over time, other more time-weighted charts such as exponentially weighted moving average (EWMA) charts would be more appropriate. There are other types of control charts designed to signal for small shifts, such as the cumulative sum (CUSUM) chart. These may also have application in an environmental monitoring context, but including them here is beyond the scope of this paper.

Control charts are regularly used in areas of manufacturing [17], computing [18], pharmacology [19], and medicine [20], to name a few examples. The general field of quality control charts is included under the aegis of *statistical process control*. [16]

3.2. Data

Our study area included portions of Mississippi and Alabama, USA, for which the location and timing of private logging activities were known for the 2009-2011 timeframe. To have both training and testing data, we acquired a collection of cloud-free Landsat images (10% nominal cloud cover or less) from January 2005 through December 2011 from the USGS GLOVIS website [21], using only Landsat 5 imagery. We avoided using Landsat 7 imagery so that we would not need to contend with the SLC off problem in the study timeframe. Nevertheless, the process shown here could be used on Landsat 7 images, provided that care is taken to screen away no-data values at the outset, similar to a cloud mask.

For validation data, we used two datasets. Firstly, we used aerial photo mosaics from 2009 and 2011 as a high-resolution check. The images were obtained from the USDA NRC Geospatial Data Gateway [22]. We were able to use the mosaics to provide an effective “before and after” check for disturbances within Landsat-resolution regions. We used the entire spatial

extent of path/row 21/37, shown in Figure 3.3, focusing on the Westervelt polygons for accuracy assessment purposes.

Secondly, we used the records of timber management of loblolly pine (*Pinus taeda*) from the Westervelt Company, focusing specifically on harvests from 2009 through 2011 in order to match the aerial mosaics and the Landsat data. This harvesting data consisted of polygons for which the date of harvesting and the general type of harvesting were noted. We used only thinning polygons for this study, as any general method capable of detecting thinning should by default also detect clearcuts.

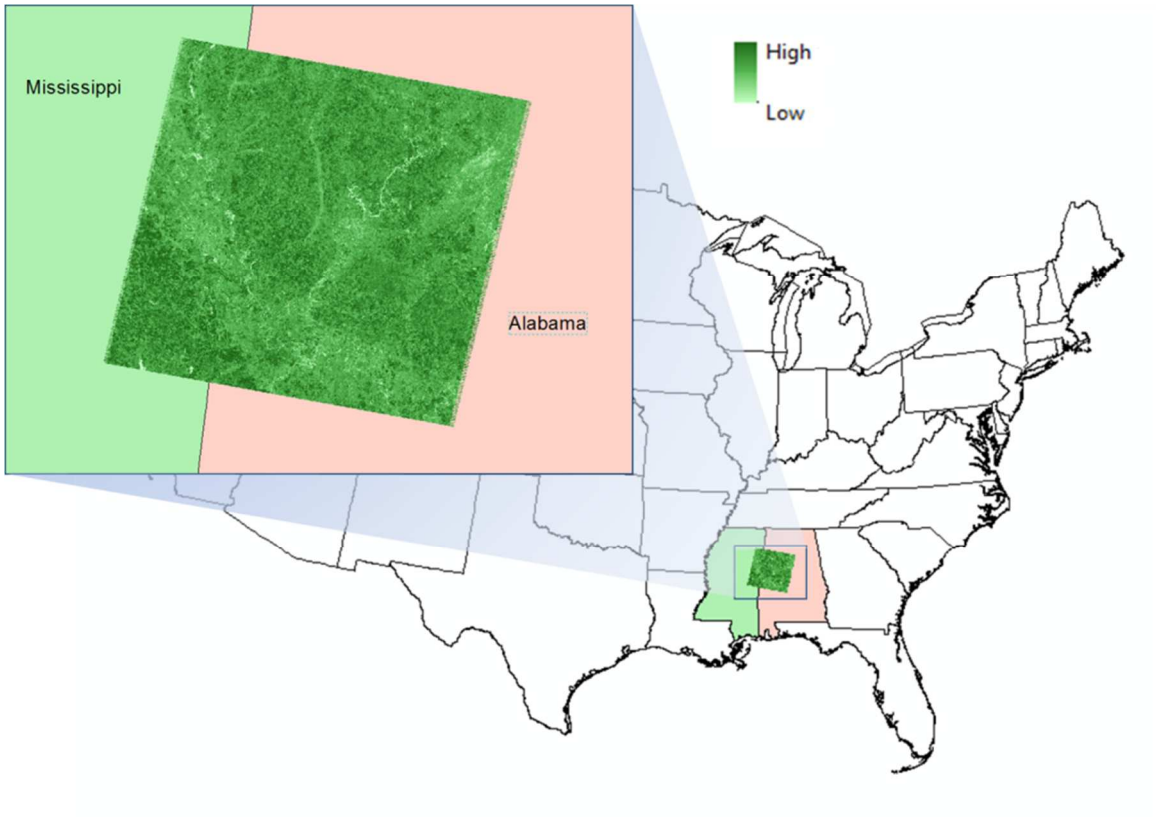


Figure 3.3. Study area, Landsat path/row 21/37. The detail is Tasseled Cap angle index from 10/3/2011, where higher values (here in white) indicate denser vegetation.

All Landsat scenes were corrected to surface reflectance by using LEDAPS [23], followed by dark object subtraction using band minima. The dark object subtraction had a significant effect in reducing time series noise, beyond that of LEDAPS alone, based on our empirical observations of the data. We converted the resulting base Landsat bands into Tasseled Cap values [24], and from these, we calculated the Tasseled Cap angle index (*AI*) [25] by the formula

$$AI = \tan^{-1} \frac{TC_{Greenness}}{TC_{Brightness}} \quad (3.4)$$

Note that while all Landsat scenes were nominally cloud-free, there were still anomalies present in the images, such as "popcorn clouds" scattered across the scenes at times. We chose the angle index because it has been shown to be sensitive to more subtle vegetation changes [25].

In order to have a solid baseline for disturbance detection, we used the first four years' worth of data (2005 through 2009) for the study area to train the harmonic regression. Figure 3.4 gives the temporal distribution of the Landsat images for the study area. Such a graph clearly presents the richness or paucity of the Landsat stack being used. Note that we selected a generous amount of training data to provide a full-year range of values in terms of day-of-the-year. Based on the distribution of image dates, it most likely would have been feasible to use any combination of 2005, 2006, or 2008 as training data. However, our aerial imagery was from 2009 and 2011, and so we chose the cutoff for training and testing based on that in order to get the closest possible matching between EWMA signals and the validation imagery. It is worth noting the lack of sparsely clouded images in 2007. These temporal features of the data also factored into our decision to use the four-year period as training.

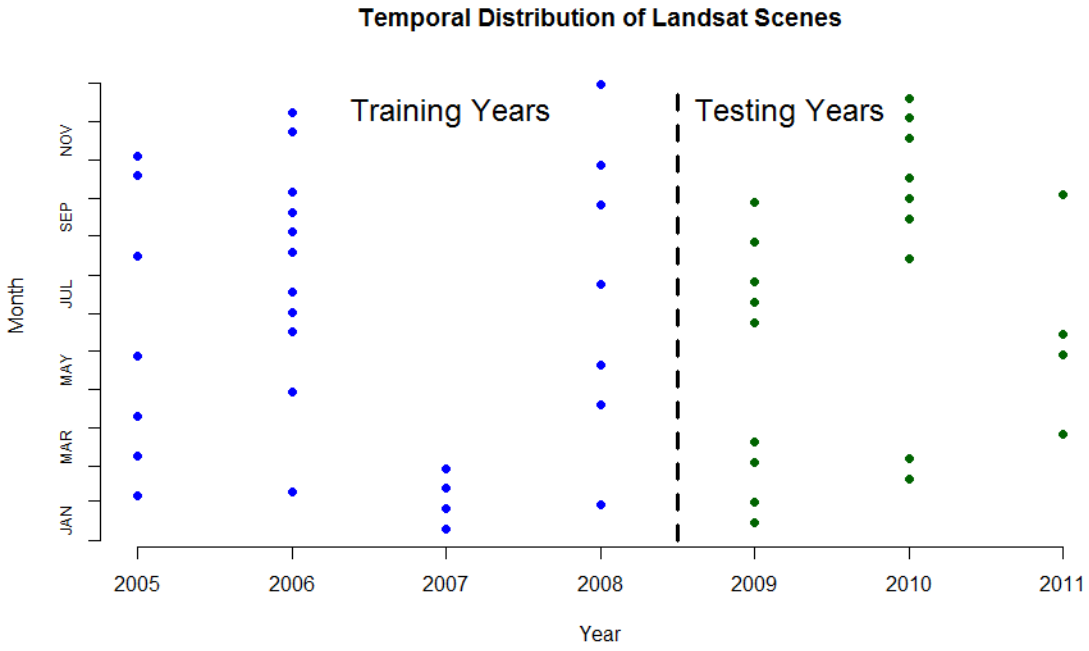


Figure 3.4. Temporal distribution of training and testing data used in this study.

3.3. Methods

In order to apply statistical process control to Landsat data, the data must first be processed and rendered into roughly independent normally distributed variables. This is accomplished by use of harmonic regression, a modified version of that used in [12] and similar to the initial steps of that used in [11]. Deciduous trees have an easily recognizable phenological curve, and loblolly pine trees lose approximately half of their needles in fall and winter [26], rendering their phenological curves amenable to modeling by harmonic regression by virtue of the clear seasonal pattern. By subtracting the fitted temporal curve from the existing data, in this case the angle index, we remove seasonality and the bulk of temporal autocorrelation from the data, obtaining a set of residual values that may be treated as being normally distributed and statistically independent. This process works over the entire image but provides results unique to each pixel. A full description of the algorithm follows, divided into elements of harmonic regression, adjustment, and EWMA chart processing, with a summary of the method given in Figure 3.5.

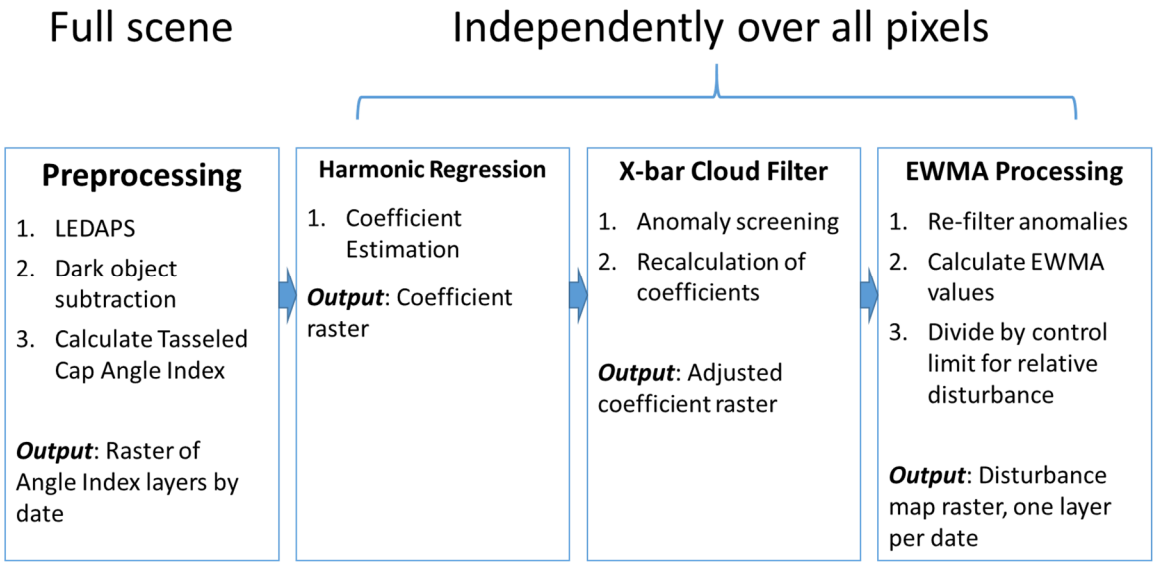


Figure 3.5. Flowchart for EWMA detection algorithm.

3.3.1. Harmonic Regression Algorithm

From a collection of d images over days of the year (from 1 to 365 or 366, depending on the year, possibly spanning multiple years), denote the dates, $\underline{T}$, and the pixel-specific values for those dates, $\underline{V_p}$, respectively as the $d \times 1$ column vectors

$$\underline{T}_{d\times 1} = [t_i]_{\,i\in\{1,2,\ldots,d\}} \text{ and } \underline{V_p}_{d\times 1} = [v_{pi}]_{\,i\in\{1,2,\ldots,d\}}, \quad (3.5)$$

noting that the *p* subscript emphasizes the vector's dependence on the pixel in question. For simplicity, scale T by converting days of the year to values on $[0,2\pi]$ by multiplying by $\frac{2\pi}{365}$ (or $\frac{2\pi}{366}$ when appropriate), yielding

$$\underline{\mathrm{T}}_{d\times 1} = [\tau_i]_{\,i\in\{1,2,\ldots,d\}} = \underline{T}\frac{2\pi}{365} \quad (3.6)$$

Let us assume that a correct linear model specification [27] for the time series by day of the year is given by a harmonic series with *m* harmonics, with m sufficiently smaller than d, and independent identically distributed normal errors, such that

$$\underline{V_p}_{d\times 1} = M_{d\times(1+2m)}\underline{\beta_p}_{(1+2m)\times 1} + \underline{\varepsilon_p}_{d\times 1} \quad (3.7),$$

where in this case, the input matrix is given, as a function of m and $\underline{\mathrm{T}}_{d\times 1}$, by

$$M_{d\times(1+2m)} = \begin{bmatrix} 1 & \sin(1\tau_1) & \cos(1\tau_1) & \sin(2\tau_1) & \cos(2\tau_1) & \cdots & \sin(m\tau_1) & \cos(m\tau_1) \\ 1 & \sin(1\tau_1) & \cos(1\tau_1) & \sin(2\tau_1) & \ddots & \ddots & \cdots & \vdots \\ \vdots & \vdots & \vdots & \vdots & \ddots & \ddots & \cdots & \vdots \\ 1 & \sin(1\tau_d) & \cos(1\tau_d) & \sin(2\tau_d) & \cos(2\tau_d) & \cdots & \sin(m\tau_d) & \cos(m\tau_d) \end{bmatrix} \quad (3.8),$$

the harmonic coefficients are given by

$$\underline{\beta_p}_{(1+2m)\times 1} = [a_{p0} \quad a_{p1} \quad b_{p1} \quad \cdots \quad a_{pm} \quad b_{pm}]' \quad (3.9),$$

with transposition denoted by $'$, and the errors are given by

$$\underline{\varepsilon_p}_{d\times 1} = [\varepsilon_{pi}]_{\,i\in\{1,2,\ldots,d\}}, \qquad \varepsilon_{pi} \sim i.i.d.N\left(0,{\sigma_p}^2\right), \sigma_p \in \mathbb{R}^+ \quad (3.10)$$

Then, estimate the pixel-specific harmonic coefficients

$$\underline{\hat{\beta}_p}_{(1+2m)\times 1} = \left[\hat{a}_{p0} \quad \hat{a}_{p1} \quad \hat{b}_{p1} \quad \cdots \quad \hat{a}_{pm} \quad \hat{b}_{pm}\right]' \quad (3.11)$$

by the usual least squares method [27],

$$\underline{\hat{\beta}_p}_{(1+2m)\times 1} = \left({M'}_{(1+2m)\times d}M_{d\times(1+2m)}\right)^{-1}{M'}_{(1+2m)\times d}\,\underline{V_p}_{d\times 1} \quad (3.12)$$

3.3.2. X-bar Cloud Filtering

In practice, a pixel may display anomalous values corresponding to small-scale cloud cover, shading, or other short-lived events that should not be modeled. As an additional precaution, the pixel-specific time series is scrubbed for anomalous values by checking the residuals from the above model against a low-threshold Shewhart X-bar chart [15-16], as follows.

If we denote the fitted values from the above model as

$$\underline{\hat{V}_p}_{d\times 1} = M_{d\times(1+2m)}\underline{\hat{\beta}_p}_{(1+2m)\times 1} \qquad (3.13)$$

then we calculate the residuals the usual way,

$$\underline{R_p}_{d\times 1} = [r_{pi}]_{i\in\{1,2,\ldots,d\}} = \underline{V_p}_{d\times 1} - \underline{\hat{V}_p}_{d\times 1} \qquad (3.14)$$

We then compute an estimated value for the error variance,

$$\hat{\sigma}_p{}^2 = \left(\frac{1}{d-1}\right)\underline{R_p}'_{1\times d}\underline{R_p}_{d\times 1} \qquad (3.15)$$

We take this value to determine which residuals are beyond a user-defined *control limit*, denoted here as L, and identify these dates as anomalous. Note that this is equivalent to processing the residuals in a Shewhart X-bar chart with control limits $\pm L\sigma$. [15-16] We thus obtain a vector of remaining dates,

$$\underline{\mathrm{T}^*}_{d^*\times 1} = [\tau_{pi}]_{i\ni(|r_{pi}|>L\hat{\sigma}_p)} = [\tau^*{}_{pj}]_{j=\{1,2,\ldots,d^*\}} \qquad (3.16)$$

where $d^* \leq d$ and the new index j reinforces the notion that the elements of $\underline{\mathrm{T}^*}_{d^*\times 1}$ do not necessarily correspond to the original elements of $\underline{\mathrm{T}}_{d\times 1}$.

From $\underline{\mathrm{T}^*}_{d^*\times 1}$, we recompute an estimate of the harmonic coefficients, denoted $\underline{\hat{\beta}_p{}^*}_{(1+2m)\times 1}$, using (3.7-3.12) and re-indexing the vectors and matrices in a manner similar to (3.16). This process is illustrated on a sample pixel in Figure 3.6. Note how the adjusted curve (green) is not deflected by the unusual points indicated in red.

This approach to filtering out clouds is of interest in its own right, because it relies only on the input Landsat data, a concept also explored in [11]. By extracting as much useful information as possible from each Landsat image, it may be possible to meaningfully utilize scenes with considerable cloud cover. In this study, we used relatively clear data, as our focus was on the detection of subtle changes to land cover, but the notion of using X-bar charts on harmonic residuals as a data-driven cloud filter is worth revisiting.

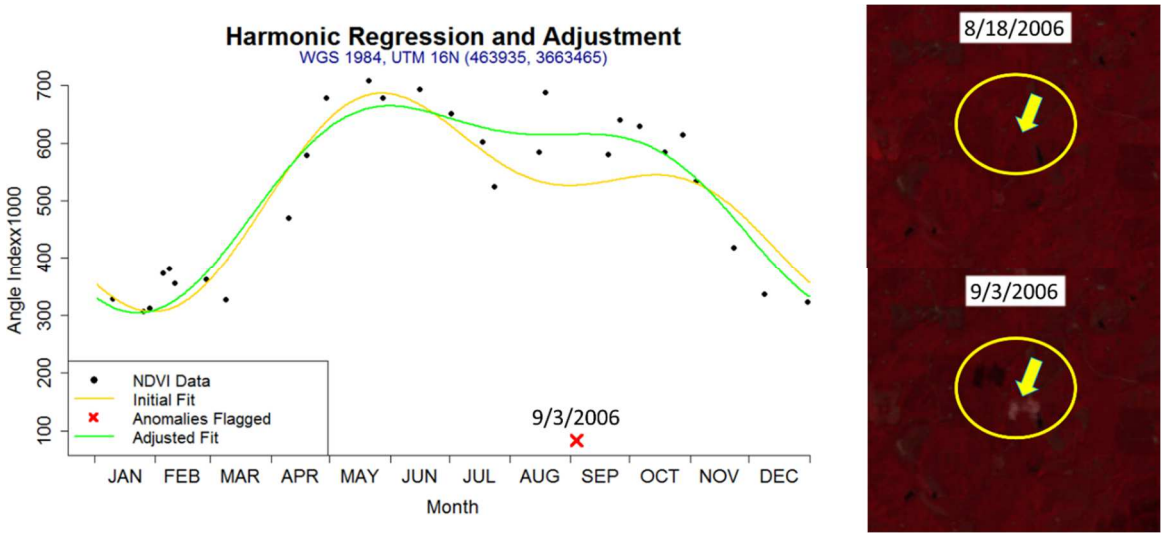


Figure 3.6. Illustration of X-bar adjustment for more robust harmonic coefficients. The anomalous value (red x) was excluded when calculating the adjusted fit (green). The images at right, with the pixel in question marked by the yellow arrow, show that the anomaly was caused by a small cloud.

3.3.3. EWMA Chart Algorithm

To illustrate how we process the time series in an EWMA chart, we make some slight modifications to our notation. In the context of the training and testing periods, we treat the training period as the first d dates in a larger timeframe of $D = d + d_1$ dates, and we accordingly extend the scaled date vector for all dates to $\underline{T}_{D\times 1}$, its input matrix $M_{D\times(1+2m)}$, and the pixel-specific value vector to $\underline{V_p}_{D\times 1}$. We first compute the fitted vector for all dates, using the previously computed adjusted coefficients, $\underline{\hat{\beta}_p^*}_{(1+2m)\times 1}$, to obtain

$$\underline{\hat{V}_p^*}_{D\times 1} = M_{D\times(1+2m)}\underline{\hat{\beta}_p^*}_{(1+2m)\times 1} \qquad (3.17)$$

We then compute the residual values,

$$\underline{R_p}^*_{D\times 1} = \underline{V_p}_{D\times 1} - \underline{\hat{V}_p}^*_{D\times 1} \quad (3.18)$$

As these residuals result from least squares estimation, we treat them as if they had a mean of 0. In practice, this is not the case because of the adjustment for anomalous values. Thus, it is necessary to once again account for these anomalous values, as incorporating them into an EWMA chart could extend their influence well beyond the date of anomaly, resulting in extended false signals. Thus, all of the residuals are reprocessed in a low-threshold X-bar chart with (possibly different) control limits $\pm L^*\sigma$ by reapplication of (3.15-3.16), resulting in a reduction of the residual time series to one of length D^*, denoted

$$\underline{R^{**}}_{D^*\times 1} = [r_{pi}]_{\,i\ni(|r_{pi}|>L^*\hat{\sigma}^*_p)} = [r^{**}{}_{pj}]_{\,j=\{1,2,\dots,D^*\}} \quad (3.19)$$

This reduced set of residuals is the object which we process directly in an EWMA chart. For a tuning parameter, $\lambda \in (0,1]$ ($\lambda = 0$ being undersirable), we generate a transformation matrix, Λ: [15-16]

$$\Lambda_{D^*\times D^*} = \begin{bmatrix} 1 & 0 & 0 & \cdots & 0 \\ (1-\lambda) & \lambda & 0 & \cdots & 0 \\ (1-\lambda)^2 & (1-\lambda)\lambda & \lambda & \ddots & \vdots \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ (1-\lambda)^{D^*} & (1-\lambda)^{D^*-1}\lambda & (1-\lambda)^{D^*-2}\lambda^2 & \cdots & \lambda \end{bmatrix} \quad (3.20)$$

Then the EWMA vector is

$$\underline{Z^*}_{D^*\times 1} = \Lambda_{D^*\times D^*}\underline{R^{**}}_{D^*\times 1} \quad (3.21)$$

It is this vector which we plot as a time series against its corresponding dates in the EWMA chart. Here, a great benefit of EWMA charts in general is realized, because for subsequent images on dates $D^*+1, D^*+2, \dots$, we simply compute the pixels' fitted values based on the adjusted harmonic coefficients, take residuals, screen them against the X-bar control limits of $\pm L^*\sigma$ in case of short-lived anomalies, and utilize the standard EWMA definition in (3.2):

$$z^*{}_{jp} = (1-\lambda)z^*{}_{(j-1)p} + \lambda r^{**}{}_{jp}, \quad j = \{D^*+1, D^*+2, \dots\} \quad (3.22)$$

This allows new images to be incorporated easily into the existing history, making the detection method on-the-fly.

Recall that the control limits for an EWMA chart with tuning parameter λ are given by

$$CL = \mu \pm L_{EWMA} s \sqrt{\left(\frac{\lambda}{2-\lambda}\right)[1-(1-\lambda)^{2j}]} \quad (3.23)$$

In our case, with the residuals having an assumed stable mean of 0, this becomes

$$\underline{CL}_{D^* \times 1} = \left[0 \pm L_{EWMA} \hat{\sigma}^*{}_p \sqrt{\left(\frac{\lambda}{2-\lambda}\right)[1-(1-\lambda)^{2j}]}\right]_{j=\{1,2,\ldots D^*\}} \quad (3.24)$$

Note that the control limits depend on an estimated value of the standard deviation. We compute this estimate using the training (or historical) data in the absence of prior knowledge about the pixel. [15-16] For pixels that underwent a disturbance in the training period, both the harmonic coefficients and the estimated standard deviation reflect this in the poor model fit. Since such pixels are not likely to remain a partly-disturbed, partly-stable state immediately after the disturbance, the practical effect is that pixels disturbed during the training period tend to signal almost immediately in the testing period.

Recalling that the control limits rapidly move towards the asymptote of $L_{EWMA} \hat{\sigma}^*{}_p \sqrt{\left(\frac{\lambda}{2-\lambda}\right)}$, and recalling that a variety of land cover classes exist in any given scene, we divide the EWMA chart by the control limits to get the disturbance record, or flag history, for that pixel,

$$\underline{F}_{D^* \times 1} = \underline{Z^*}_{D^* \times 1} \div \underline{CL}_{D^* \times 1}, \quad (3.25)$$

where ÷ represents element-wise division. We insert 0's artificially into the parts of the history that were screened out by (3.19), although any other code value could be inserted in the interest of easily tracking anomalies. By iterating this process over all pixels *p* in the scene, we obtain a raster with an equal number of layers as the input raster, corresponding to the initial temporal distribution, and giving relative disturbances from the conditions predicted by the training period.

3.3.4. Specific Application

For this study, given our stack of Landsat images converted to tasseled cap angle index, we took the dates associated with each image and used the images corresponding to the first four years (2005-2008, inclusive) of data as inputs into the EWMA detection algorithm, first computing harmonic coefficient estimates. In our case, we used $m = 2$ harmonics, resulting in five coefficient estimates for each pixel: a constant term with two sine and cosine terms of increasing frequency. We chose this value based on observations in [12] that two harmonics were usually sufficient to capture the vast majority of the periodic variation in the time series. It is worth noting that one effect of harmonic regression is the removal of seasonal temporal autocorrelation in stable pixels, as the regression captures such periodic behavior. It is also worth noting that by taking multiple years as training data, the coefficient estimates for pixels undergoing consistent vegetative growth "centered" the curve in the middle years of the training period.

Once the residuals were computed, we then calculated the historical standard deviation of each time series from the pixel-specific residuals and flagged any residual values farther than $L = 2$ standard units away from 0, in effect passing the residual time series through a low-threshold X-bar chart. We used 0 as the mean since the residuals under a stable pixel are assumed to be 0. We treated the flagged values as anomalous and temporarily discarded them, recalculating the least squares estimates of the harmonic coefficients for the remaining time series values. This simple second iteration had the effect of screening away most cloud and shadow interference without the use of cloud masks. The result of the process at this point was a five-layer raster of adjusted harmonic coefficient values, providing baseline information for phenological processes in the study area. This proved much easier to store and manipulate than a raster containing fitted values for all of the dates in the study timeframe.

From this baseline data, on a per-pixel basis, we recomputed fitted values for the entire history of the time series, both in the training and testing periods, 2005-2011, in all. We calculated residual values again by subtracting the fits from the observed angle index values, and once more we screened, from the training period only, anomalous values with a low-threshold (2σ) X-bar chart. This had the effect of reducing the estimate of the historical standard deviation, thus improving the sensitivity of the EWMA chart used next. We then used this standard deviation estimate for calculating the EWMA control limits, setting the mean part of the control limit to 0 for simplicity in light of the empirical observation that the calculated mean

estimates were very close to 0 in the overwhelming majority of pixels. In order to filter out short-term anomalies in the testing period and avoid biasing the subsequent EWMA chart values in the event of such an anomaly, we subjected the testing period residuals to a very-high-threshold ($L = 12$) X-bar chart. We chose this value based on empirical observation of the charts, balancing the need to screen short-term anomalies with the need to signal for persistent disturbances.

To test which weight parameter would be best for our purposes, we ran the algorithm on a subset of the scene, letting parameters for the charts range from $\lambda = 0.1$ to $\lambda = 1$ by increments of 0.1. These parameters drive the retrospective nature of the chart, with more retrospective charts being less likely to read false positives due to anomalous data. Upon initial testing, even light disturbances signaled across the range of weights. Accordingly, we chose a weight that reduced the chance of false signals due to singularly anomalous data values. That is, we chose a weight which gave charts that were stable in the presence of large singular deviations in the residual time series yet were responsive enough to signal disturbances within one to three dates, depending on the severity of the disturbance in question. Based on our observation of the charts, we determined that an EWMA weight value of $\lambda = 0.3$ worked well enough in our study area, being sensitive to disturbances while not providing excessive false signals. It is uncertain whether this value would generally perform well across other land cover types, but we do note that other weights in the range $0.1 \leq \lambda \leq 0.5$ gave comparable results in our study area.

In order to derive an estimate of the relative severity of the disturbances flagged, we took signaling values (those outside the control limits of the chart) and divided these values by that particular chart's control limits, rounding down to the nearest integer in absolute value. Since the control limits were stable by the time the testing period was reached, the division provided a way to compare relative disturbances. Thus, for each pixel, we obtained a time series of integer values, indicating both when disturbances were signaled, the relative severity of the disturbance, and the nature of the disturbance (growth or reduction in vegetative cover). We compiled these time series per pixel and generated an output raster with one layer for each date in the test timeframe.

3.4. Results

The algorithm's outputs took the form of a stacked raster, each layer corresponding to the signaled disturbances for a particular date in the original Landsat stack. In effect, our results could be checked along three dimensions: space, time, and severity. In the following sections, we attempt to give a sense of the results along these lines, using the aerial images and Westervelt polygons to validate our observations.

The challenge in analyzing the results comes in showing that the disturbances are accurately signaled. In order to do this, we used the Westervelt polygons in conjunction with the aerial imagery. Three questions were of interest in working with the harvest polygon data. Firstly, did the control chart algorithm accurately signal (or not signal) according to the actual changes on the landscape? This is a question of whether the EWMA charts properly identified disturbances in space, although answering this question depends on the disturbance severity and time of occurrence. The second question is one of accurately identifying the severity of the disturbance, distinguishing between subtle and gross disturbances. Was there a relationship between the severity of the disturbance signaled and the severity indicated by the polygon data? The third question was about the timing of the disturbances being signaled. Did the EWMA charts identify the correct times of disturbance?

It is worth noting some considerations in using the polygon information. Firstly, the polygons were all classified as types of thinning. Despite this, cursory observation of the aerial images showed that the polygons' treatments were not uniform in either time or space. That is, some polygons were thinned unevenly (a transport road cut through one section, for example), and commonly the polygons were large enough that it would have taken weeks to give them a full thinning treatment. Secondly, some of the treatments observed in the aerial imagery did not match the descriptions from the polygon data. For example, one polygon was in actuality completely clearcut based on the aerial photos. Another challenge came from the aerial photos themselves. Being mosaics, the images were taken at varying points throughout their respective years, with the majority being taken during the summer months. Thus, it was possible, for polygons harvested on the edge of the testing timeframe, that the photos missed the contrast in the treatment, either before or after. We illustrate this in Figure 3.7, where the documented thinning occurs both after the final Landsat image is taken as well as after the 2011 aerial photo.

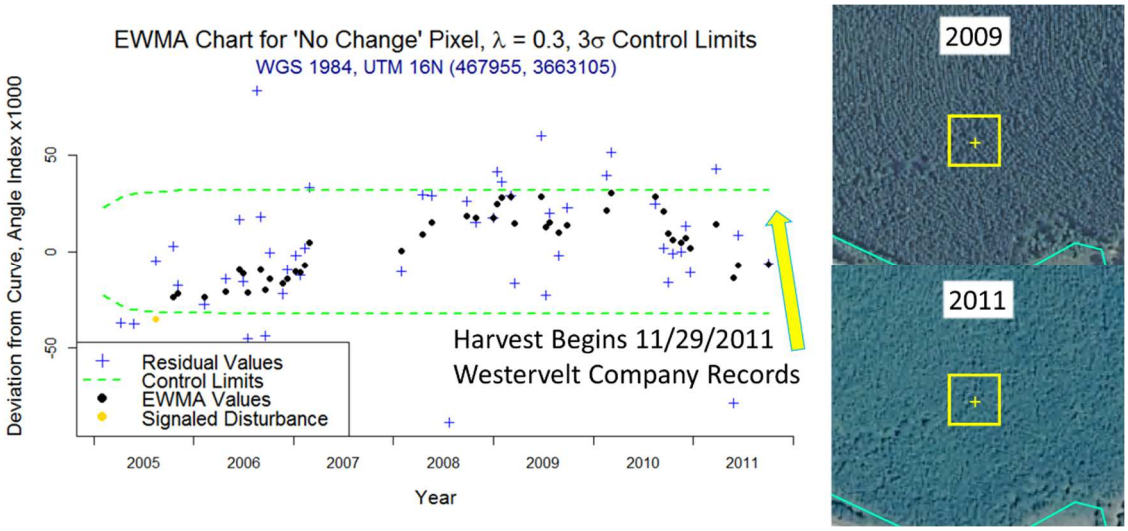


Figure 3.7. EWMA chart for a pixel that had a harvest after the timeframe. The images to the right depict the pixel in question (yellow box) according to the aerial images.

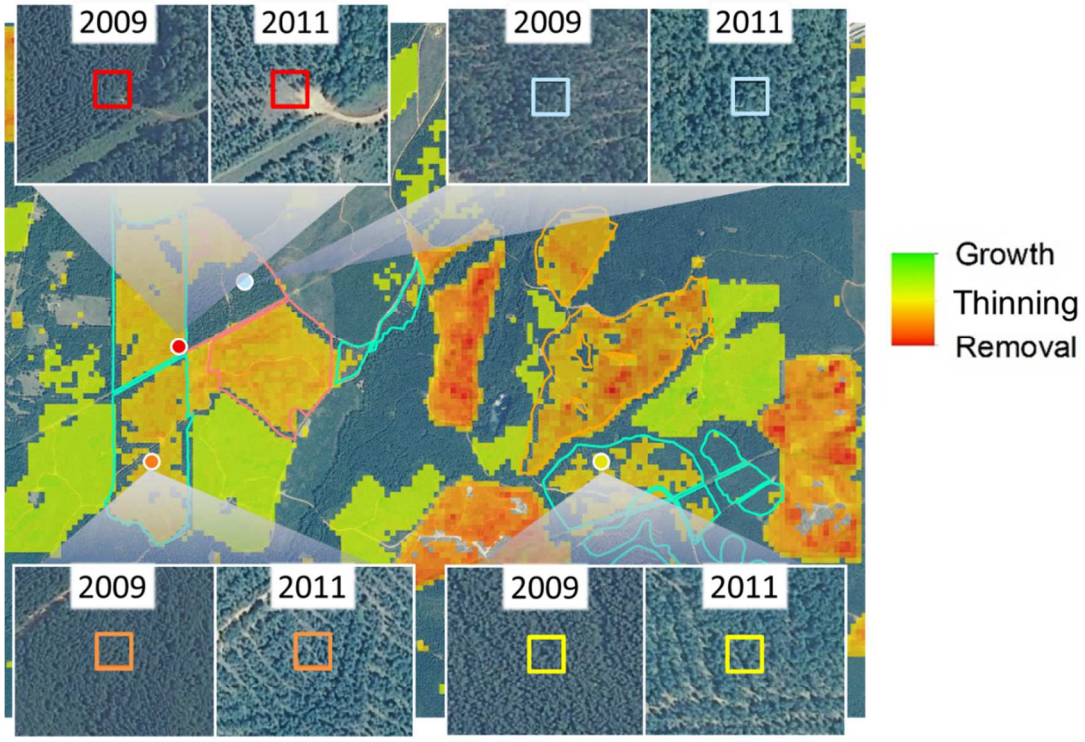
In light of the above challenges, we chose to conduct the accuracy assessment for the space and severity dimensions by selecting one point at random from within each Westervelt polygon (where we had records of harvesting), associating each point with the pixel containing it. In all, we selected 141 pixels in this manner. For each of these pixels, we selected the last element in the disturbance time series. We chose the last element in order to have the best possible chance at matching the disturbance record with the 2011 aerial imagery, because our validation involved observing the difference between 2009 and 2011 in the aerial mosaics. In such a situation, attempting to compare the disturbance record during earlier dates in the time series with the 2011 images would have resulted in mismatches due to post-disturbance recovery. An example of this situation is depicted in Figure 3.8, in the case of the blue example pixel. The stand at this pixel was thinned early on in 2009 (but after the aerial image for 2009). By 2011, the forest had recovered sufficiently for the aerial photographs to show little change in vegetative cover. If we had classified this pixel as thinned, it would have disagreed with the later aerial assessment.

Thus, the accuracy assessment was carried out as follows. For each test pixel, we recorded the final disturbance signal associated with that pixel. Then, independently observing the region at that pixel, we made a visual comparison between the 2009 image and the 2011 image. For consistency in estimation, we used the aerial classification shown in Table 1 and illustrated in Figure 3.8. Note that the criteria in the table are for relative changes, not absolute changes. Note further the effect of calculating pixel disturbances independently of their surroundings, as the red

example pixel in Figure 3.8 reads as a clearcut by relative cover change within the pixel, despite its neighborhood being obviously thinned.

Table 3.1. Accuracy assessment criteria.

Aerial Classification	Description
No Change	No visible difference, or possibly slight increase in vegetation
Light Thin	1-25% relative decrease in canopy or tree cover
Heavy Thin	26-75% relative decrease in canopy or tree cover
Clearcut	76-100% relative decrease in canopy or tree cover



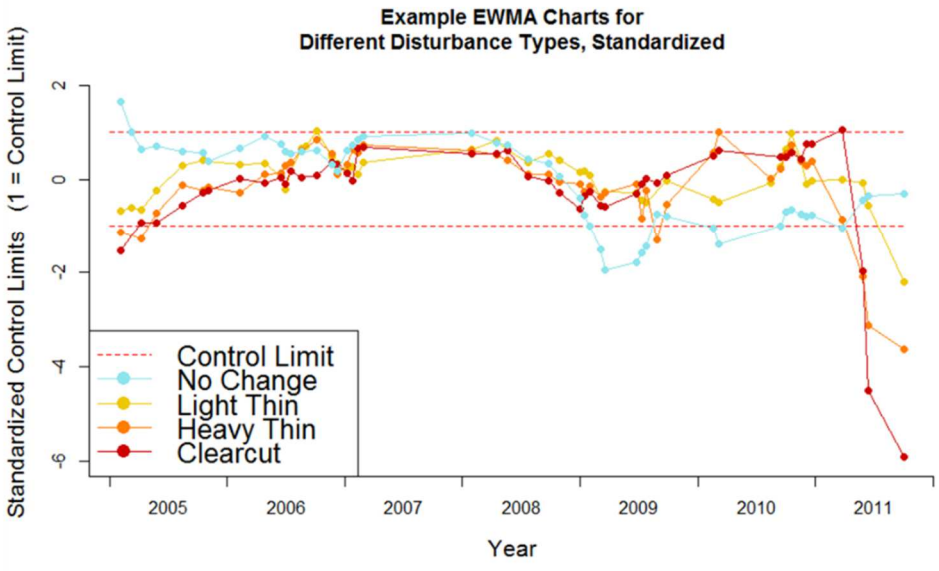


Figure 3.8. a) Example pixels of each type of disturbance, from a variety of Westervelt polygons. The squares represent Landsat pixels in scale and location, and the following descriptions apply only to the space within the squares. Clockwise from upper right: No Change (blue boxes), Light Thin (yellow), Heavy Thin (orange), Clearcut (red). The clearcut in this case was at the head of a logging road within a thinned stand. Green regions are where the algorithm signaled growth, orange and red regions are where the algorithm signaled thinning and removal.

b) EWMA charts for the example pixels in a). Note that the No Change pixel was actually thinned in 2009, in agreement with that polygon's information.

3.4.1. Accuracy Assessment (Space)

To assess the spatial accuracy, we simply observed the agreement between the EWMA charts for the relevant pixels and the aerial images on a change/no-change basis. We treated values for which the EWMA charts showed net growth as if they were no change, so that "change" in this context was equivalent to vegetative removal. The results of this dichotomous accuracy assessment are given in Table 2.

Table 3.2. Dichotomous accuracy assessment results.

EWMA / **Aerial**	No Disturbance	Disturbance
No Disturbance	27	15
Disturbance	6	93

The overall accuracy was 85% (120/141), and the Cohen's kappa for the dichotomous assessment was 0.621. In general, we observed good agreement between the EWMA charts and the aerial interpretations, both in the case of disturbance and in the case of no disturbance. The number of no-disturbance observations may seem surprising, given that each pixel in the assessment was contained in a polygon documented as being harvested between 2009 and 2011, but we have already shown typical examples of the no-disturbance occurrences in Figures 3.7 and 3.8 (blue pixel). Similarly, we have shown typical examples of disturbances in Figures 3.2 and 3.8 (red, orange, and yellow pixels).

The commission error (false signal rate) was 36% (15/42), although this was impacted by the timing of the aerial photographs and the ability of the algorithm to signal changes at the edge of the timeframe. One example of this is shown in Figure 3.9, in which the aerial photograph was taken after the thinning took place. In this case, the aerial interpretation was that of "no change", since the interpretations were based on *relative* forest cover. However, the thinning is documented and is signaled on the EWMA chart, leading to an error of commission.

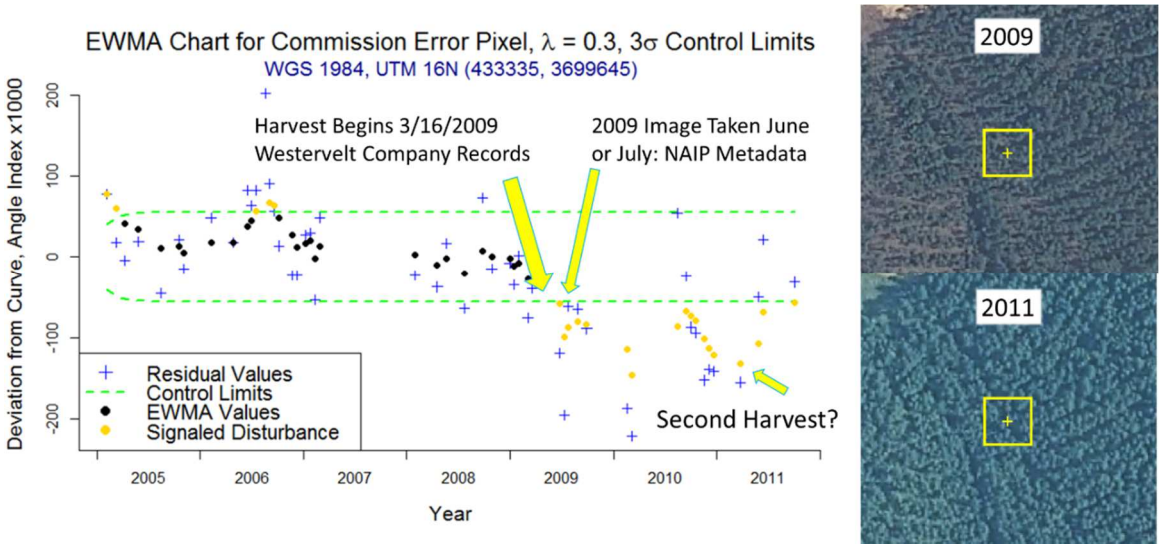


Figure 3.9. EWMA chart for a pixel with a commission error (false alarm). The images to the right depict the pixel in question (yellow box) according to the aerial images.

The omission error rate was 6% (6/99). We show a typical example of this type of error in Figure 3.10. In this pixel, the underlying forest was steadily maturing over time, as evidenced by the clear increasing trend in the EWMA chart values. By the time the pixel was thinned, the EWMA values were high enough that the thinning had the effect of returning the pixel to its original condition. Noting that the raw residual (blue cross) for the final value is quite low, relatively, we are confident that the EWMA chart would have signaled given one or maybe two additional images to confirm the new residual mean.

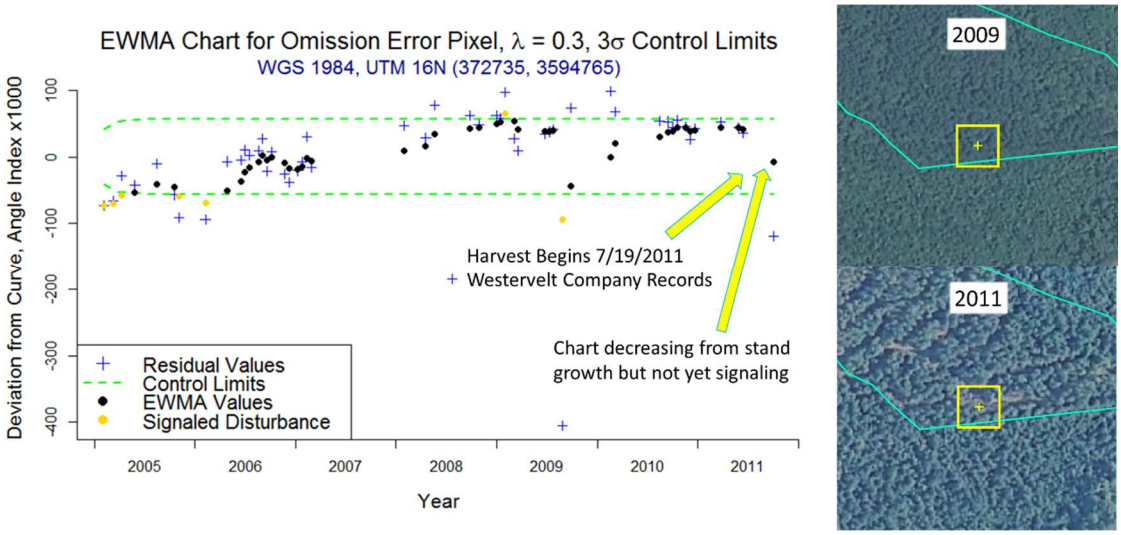


Figure 3.10. EWMA chart for a pixel with an omission error (failure to signal). The images to the right depict the pixel in question (yellow box) according to the aerial images.

These above examples suggest that the bulk of these errors, especially the commission errors, are "startup" and "cutoff" errors, artifacts of the timeframe's finite nature. This is because the EWMA chart necessarily trades a little in response time for the ability to detect a smaller change, as a result of its weighted average. From the examples, it is clear that the response time of the charts is still usually very fast, typically signaling in the first image after the disturbance. In a continuously running algorithm with no last date of observation, these errors probably would not have occurred, or at the least, would have been mitigated in the next iteration or two. We will discuss the temporal responsiveness of the EWMA charts further when answering our question about the time dimension.

3.4.2. Accuracy Assessment (Severity)

The basic question about the severity dimension was whether the EWMA charts produced signals in agreement with the aerial image comparison. In this case, we wanted to look for associations between the two methods. Because we were interested here in forest degradation only, we again treated the EWMA chart values signaling for growth in the pixels as "no change". The results are displayed in Figure 3.11. We used violin plots [28] to illustrate the distributions within each class.

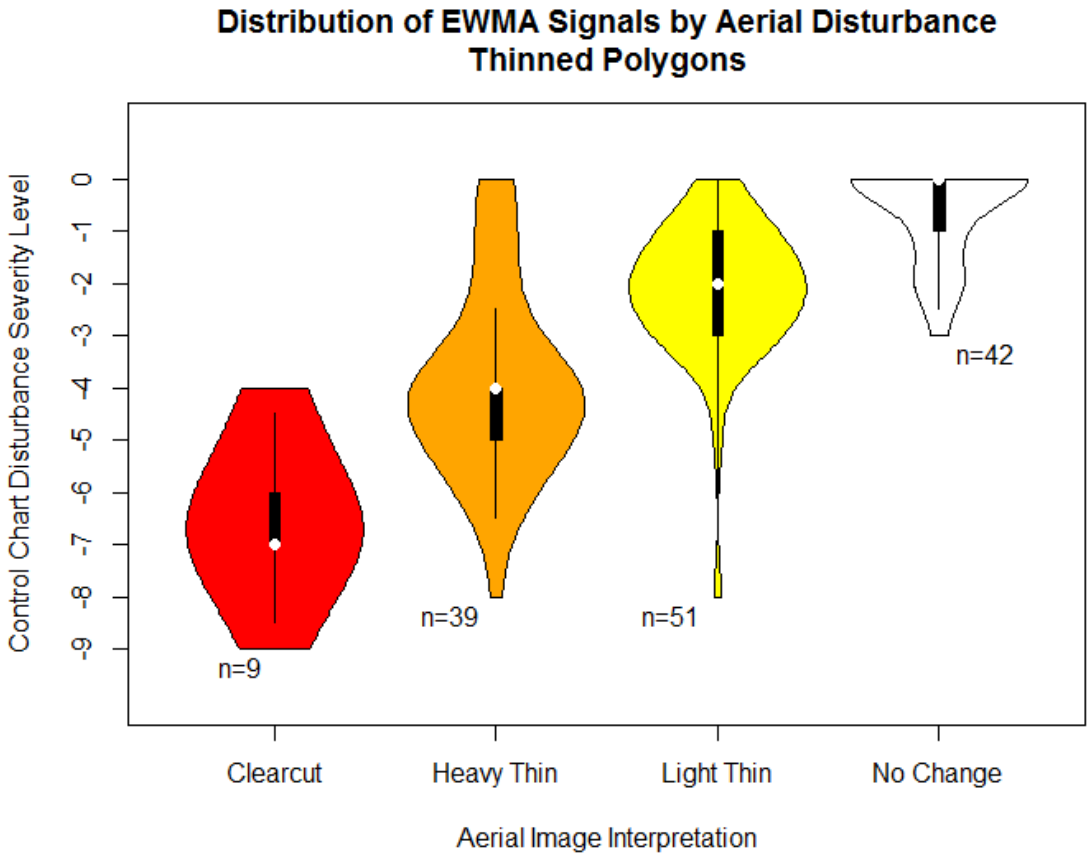


Figure 3.11. Comparison of algorithm disturbance level with that observed in aerial imagery. In violin plots [28], the black box and whiskers correspond to the usual 1st and 3rd quartiles, with the white dot representing the median. The thickness of the violin corresponds to the prevalence of data at the vertical value, similar to a histogram. Note the clear positive association between the EWMA outputs and the aerial categories.

It is clear from Figure 3.11 that there is a simple relationship between the disturbance signal and the manner of disturbance. The Spearman correlation ([29], used here due to the ordinal nature of the data) was 0.753, highly significantly different from 0. Succinctly put, the greater the signal's deviation from zero, the more severe the disturbance tends to be. Based on this, in our case we might use a rule of thumb that disturbance signals between -1 and -3 can be considered light thins, signals between -3 and -6 may be considered heavy thins, and signals beyond -6 tend to be clearcuts or wholesale removals. As the classifications in the assessment were in some sense arbitrary and based on regions that were thinned from mature stands, they may not reflect cutoff values across the remaining land use/land cover types in the scene and history. It is worth repeating that the values given by the algorithm are *relative disturbances*

against that pixel's baseline history. Thus, perhaps the best way to treat the disturbance signals is as fodder for a "disturbance heat map", illustrated for the final date in the timeframe in Figure 3.12. The long diagonal path of disturbance is that of a tornado that moved through the area on April 27, 2011. [30]

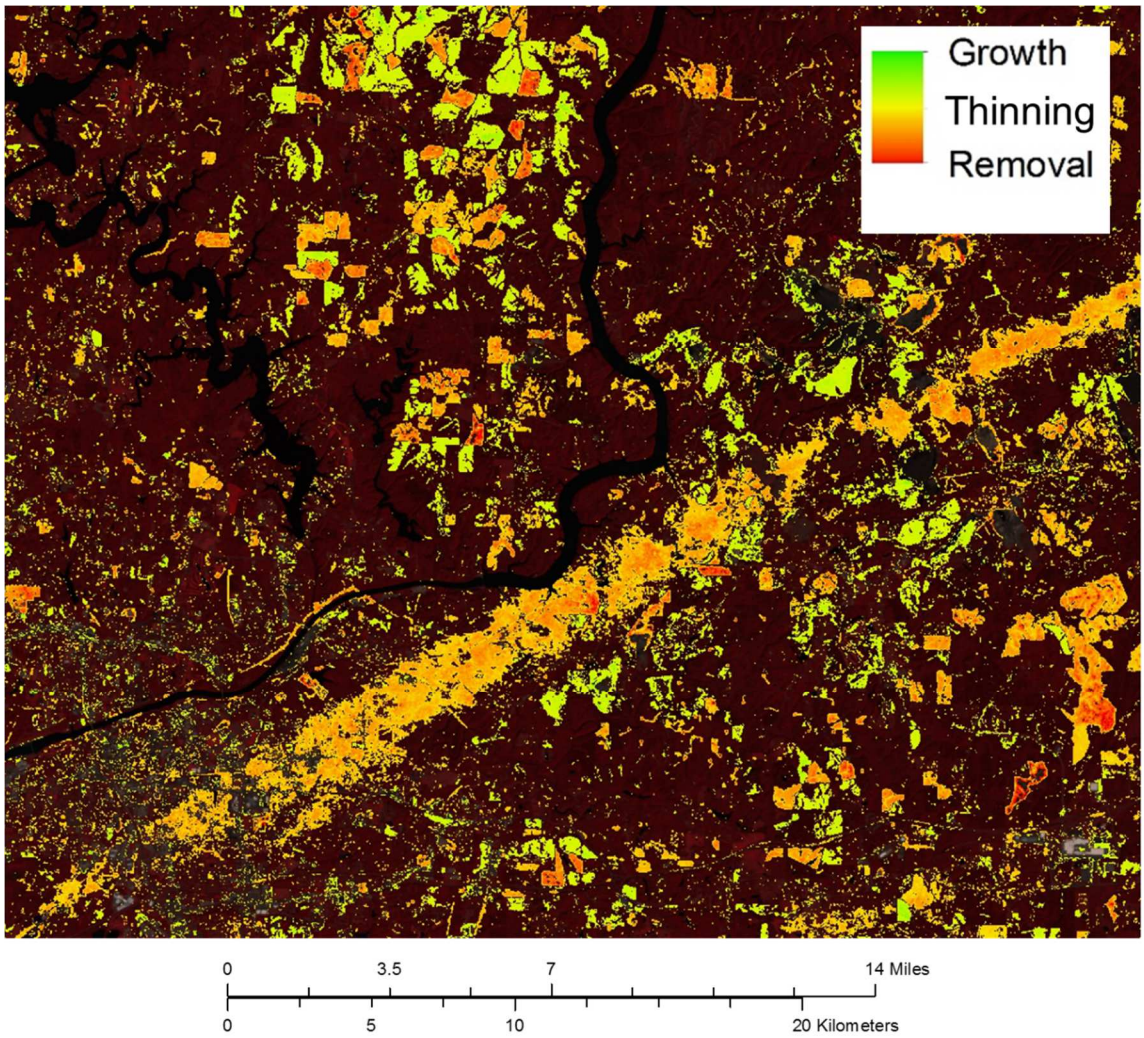


Figure 3.12. Disturbance magnitudes for 10/3/2011. The town of Tuscaloosa, Alabama, USA, is in the lower left. The diagonal linear feature is a tornado path. [30]

Indeed, by stacking the disturbance maps temporally, one may generate a "disturbance heat movie", allowing for additional dimension to tracking the changes on the land. Note as well that while the Westervelt polygons used here were only thins or removals of vegetation (shown as yellows, oranges, and reds), the image in Figure 3.12 also indicates regions where the vegetation has increased beyond the training baseline (shown in greens). We offer a more specific example

of this in Figure 3.13, in which we observe development in a young pine stand. While we did not have truth data by which to statistically check for stand growth, the example strongly suggests that one may track afforestation as easily as deforestation with EWMA charts.

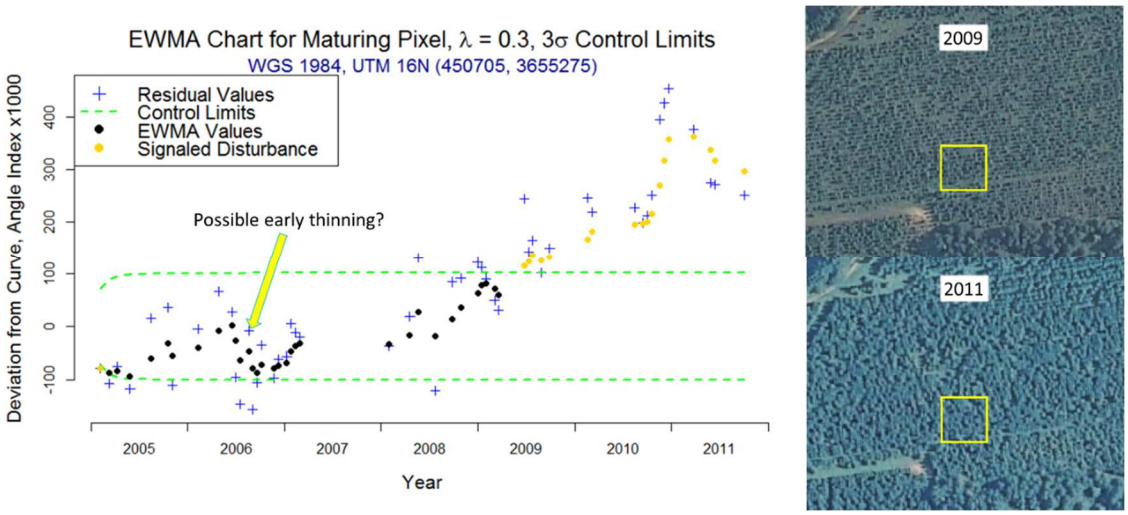


Figure 3.13. EWMA chart for a maturing pine stand pixel. The images to the right depict the pixel in question (yellow box) according to the aerial images.

We reinforce the notion of signaling for forest growth with the example shown in Fig. 14. In the process, we also illustrate the manner in which harmonic regression captures intra-annual variations to prevent false signals due to annually recurring disturbances such as plowing or planting of agricultural fields. In Fig. 14, we see a building with a young tree plantation to the north and an agricultural field to the south. Within a possible subpixel misregistration a few meters south and east, the EWMA algorithm precisely signals both the growth of the stand as well as the landscaping around the building, while offering no signal for the agricultural fields. It is worth recalling that the EWMA charts were performed on each pixel independently, with no masks to filter results. The algorithm does not signal the fields because the variations are intra-annual and not inter-annual. While we display the signals only for one date here, the same pattern was evident throughout almost all dates of the testing period, with slight signal variation around the edges of the field and stand. Recall that this is only an example, because our validation data for severity was only for harvest pine stands. However, the example offers compelling evidence of the algorithm's general utility.

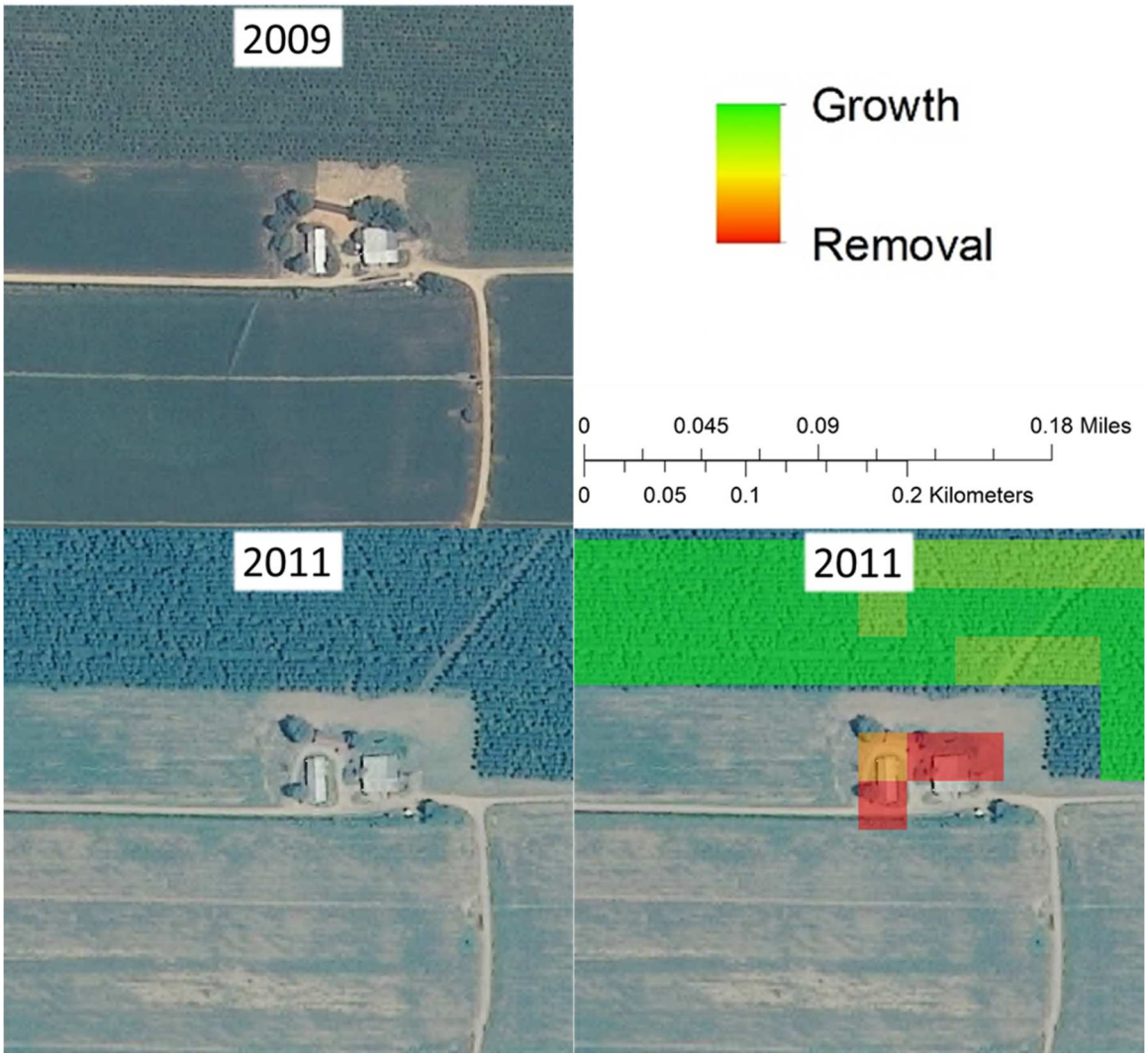


Figure 3.14. A region with both agricultural and silvicultural activities. The color code of the EWMA signals is exaggerated slightly for easier interpretation. Note the precision of the signals and the lack of signals for the agricultural fields.

As one final example of the sensitivity of the method, we present an area in which two major changes take place from 2009 to 2011. (Fig. 15) In this example, the western part of the area is a maturing young pine plantation, whereas the eastern side undergoes removal and conversion to a field. While no thinning occurs in this region (the region does not include any Westervelt polygons), it is clear that the algorithm is signaling according to the severity of the disturbances. In particular, note the way that colors in the eastern side correspond to the degree of the removal, with the harshest removals taking place where a dirt road was created (right-center of the region, red signals). It is also interesting to observe how the algorithm ignores areas of no change, such as a clearing in the maturing pine stand (lower left of the region, empty pixel). From this example

and the preceding ones, it is clear that EWMA detection is appropriately signaling the severity of the disturbances.

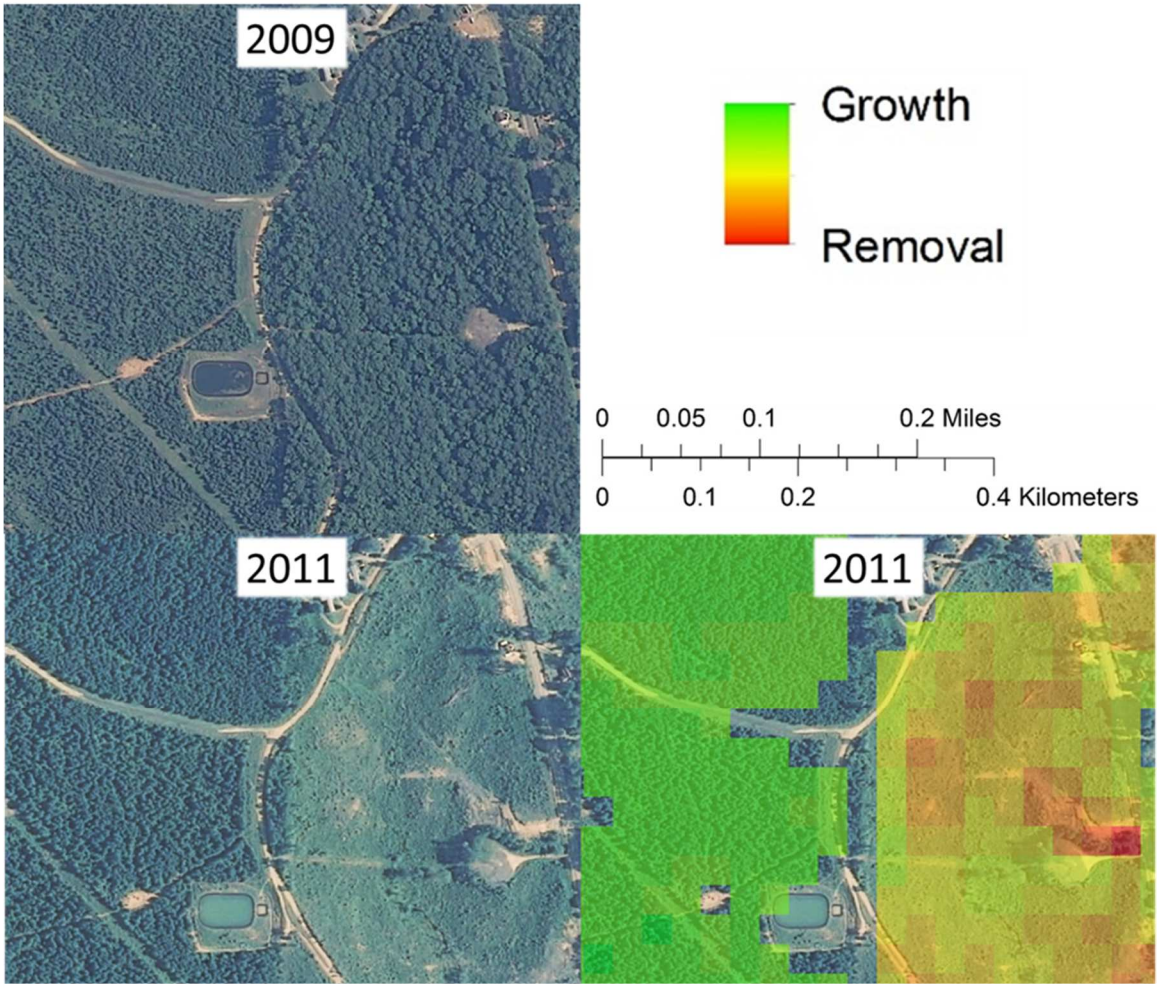


Figure 3.15. A region undergoing both stand maturation and stand removal, illustrating the manner in which the EWMA charts indicate severity of disturbances. The pixel colors are exaggerated slightly for easier interpretation.

3.4.3. Accuracy Assessment (Time)

We were able to use the aerial imagery from 2009 and 2011 to assess the effectiveness of EWMA charts in accurately signaling both the location and severity of disturbances as subtle as light thinning, also showing in the process that the charts can signal for forest growth as well. However, these results were in some sense limited by focusing only on the final image in the timeframe. We did this in order to most effectively utilize the 2011 aerial photos, but the

assessment thus far does not speak for the responsiveness of the EWMA charts, outside of the graphical examples.

Figure 3.14 depicts a subset of our study area, in which we move our focus from the *magnitude* of the disturbances to the *timing*. For this figure, we show the EWMA disturbances by the year in which they began signaling for at least four images. This consistency constraint is a simple way to avoid displaying anomalous false signals, such as those caused by the fringes of clouds and shadows, that may have slipped through the cloud filter in the algorithm. This approach of seeking persistent change has been used by other methods. [2-3,5-7,11]

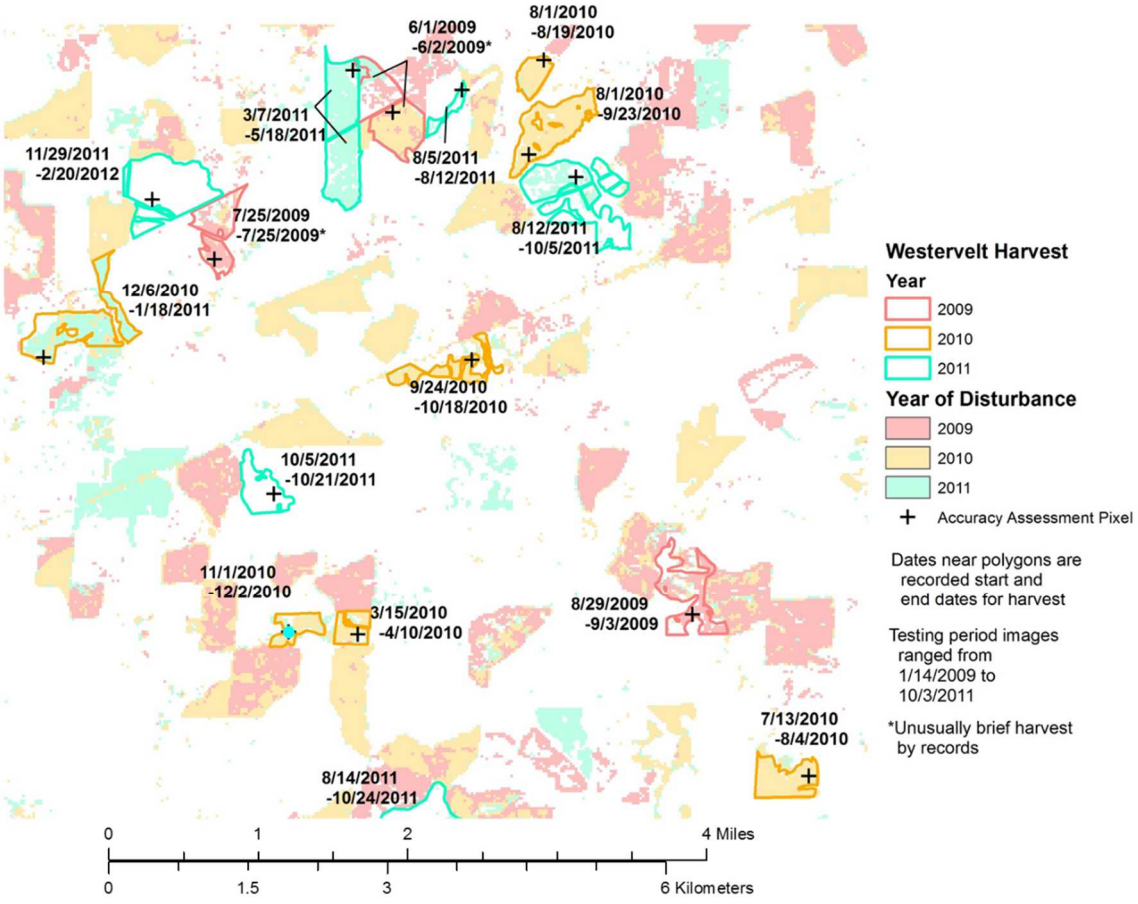


Figure 3.16. Disturbance map based on the year of measured disturbance. The polygon outlines are pine stands harvested by the Westervelt Company.

It is apparent from Figure 3.14 that the EWMA algorithm does quite well in identifying the disturbances in the Westervelt polygons in the year they happen. The exceptions in this figure derive from stands that were harvested after the final image in the timeframe. In the case of the

polygon in the northwest, the start date for the harvest was in late 2010, but the actual harvest extended into January of 2011, according to the records.

To get a sense of the finer responsiveness of EWMA charts, we consider a Westervelt polygon recorded as being thinned in September and October of 2010, shown in Figure 3.15. What is impressive about this polygon is that we can observe not only when the polygon was thinned, we can see the order in which the harvesters performed the thin. Through the use of an existing road (circled in the figure), it is clear that the harvesters opened up the center ahead of schedule before moving first east and then west of this point. This represents near real-time responsiveness, and we stress that the harvest was a thinning, subtle enough to be difficult to detect in its own right. Note that the stand to the northwest of the image was not in the Westervelt records, but it was harvested in early 2010.

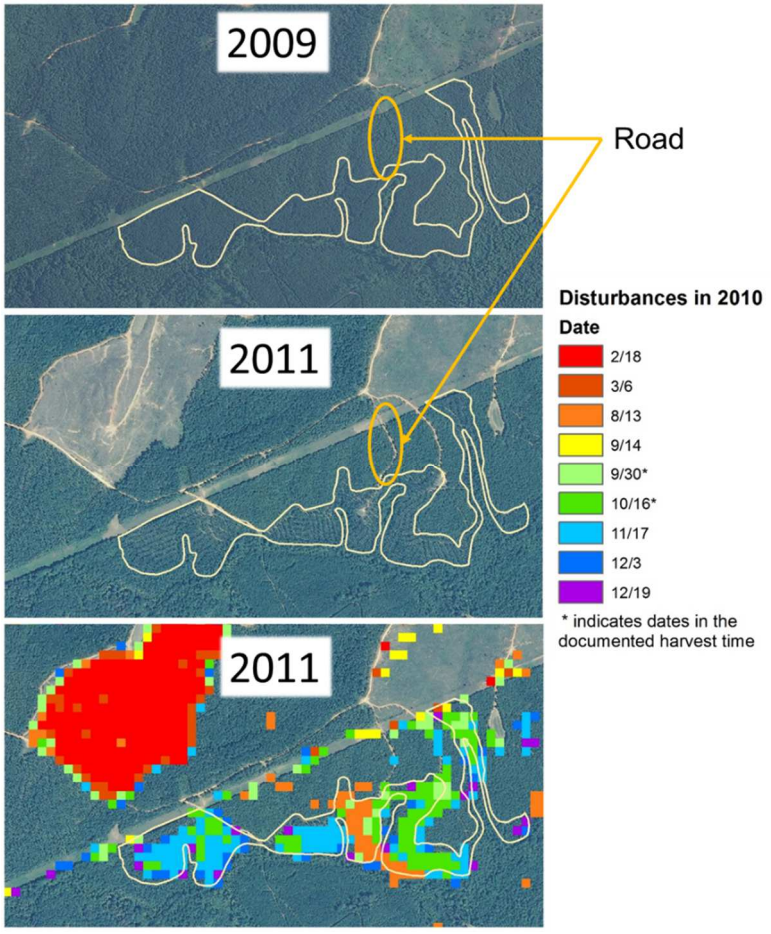


Figure 3.17. Disturbances based on date in 2010 for a pine stand undergoing thinning in September and October. There is a clear pattern of harvesting in the east side of the stand before the west side.

Because the thinnings typically took place over a range of dates within each polygon, it was difficult to concisely summarize the results of a temporal accuracy assessment. We will simply state that for the accuracy assessment pixels for which the stands underwent basic harvesting according to the Westervelt records, over half of the EWMA charts for those pixels signaled within the first two images after the recorded start date. This, coupled with the above examples, leads us to conclude that the EWMA chart method is very responsive in the time dimension, allowing near-immediate signaling for a wide range of disturbances. Given that the charts also accurately signal the magnitude of the disturbance (or growth) and easily incorporate new images as they arrive, we conclude from the above assessments that EWMA charts are particularly well-suited for on-the-fly environmental monitoring.

3.5. Discussion

It is well worth reviewing a few key points or challenges to the change detection method presented here. Perhaps the first of these may be the nature of our sampling and the validation data we used. In particular, each pixel we sampled was from a polygon designated to have been thinned in the testing period. Theoretically, this allowed us to test the algorithm's sensitivity to known changes, but it allowed no inference regarding the algorithm's specificity. That is, we had no control group of undisturbed pixels with which to look for false alarms.

Interestingly, of the 141 pixels thus sampled, 42 of them showed no visual sign of thinning or other vegetative removal. As we discussed in the results section, there are likely a variety of reasons for this high proportion of unthinned pixels in a collection that was supposed to be all thinned. These reasons appeared mostly to do with thinnings taking place at the fringes of the testing period, beyond the span of the images. As a result, we have a nice heuristic view of the algorithm's ability to detect false positives. While we cannot make any rigorous statements regarding the specificity of the algorithm here, the results appear promising.

Another issue that arises is that of independent sampling. Clearly there is temporal dependence from image to image. This is the reason that we used harmonic regression to remove all seasonal or periodic influences from the time series, resulting in a set of residuals that is temporally uncorrelated over the training period. In the event of a disturbance (possibly including stand growth), these residuals lose their uncorrelated nature, tracking the change as it

occurs. This is true if a pixel undergoes disturbance during the training period as well, with the result that the harmonic regression centers the residuals as best it can on the historical mean value. In either case, the time series appears to violate the original assumptions behind the method.

This, however, is desirable. Within a control chart framework, when everything is in process, errors behave as assumed, in this case independently and identically normally distributed. In effect, the control chart is testing the validity of this assumption at each successive measurement. When the assumptions fail to convincingly explain the observed measurements, the chart signals that the process is out of control. Thus, the method outlined here really extracts meaningful information (forest disturbance and severity) from those places where the harmonic regression did not adequately model the pixel.

Another question that may arise is that of how to discriminate between changes in forest land cover as opposed to changes in (or two and from) other land cover classes such as agricultural and urban land covers. Because the EWMA detection algorithm operates on each pixel independently of the others, and because the magnitude of signaled disturbances is relative to the conditions of the pixel during the training period, we did not make any attempt to discriminate between the land cover classes in this paper. If one was making estimates of change to forest cover over a specific area, then one would need a forest-nonforest mask during the training period in order to measure the appropriate areas.

One other question that may arise from this particular study is one of how much training data the EWMA detection algorithm needed. It was shown in [12] that harmonic regression requires at least one image from "key dates" of the year, typically corresponding in a forestry context to phenologically important times such as green-up in the spring and senescence in the fall. In that sense, our use of four years of training data was very generous, being largely a function of having the data on-hand. We are confident that the EWMA detection algorithm could run well off of a single year of training data, and we intend to demonstrate this in future studies.

There are several other areas for future work on and improvement for this method. For example, one of the underlying assumptions behind the EWMA and Shewhart charts is that the sampling is done systematically, with equal temporal intervals between measurements. Owing to the nature of the Landsat stacks after removing pixels or scenes that are unsuitable for analysis, this assumption is generally not met. In the preceding research, we simply allowed the EWMA

charts to act as if the measurements were still systematic, with the useful results presented here. However, it should be possible to modify the chart's construction to allow relative weighting to incoming data, relative to their temporal context. This modification would allow the control chart to work with less lag time, which is important, particularly when data are sparse. For example, after a period of extended silence, a new data point may be given particularly heavy weight; conversely, if measurements become particularly dense (for example if one uses data from multiple platforms), then it may do to reduce the impact of each successive point. These are issues of appropriateness of temporal scale and were not addressed in this paper, but they are certainly areas worth exploring.

The algorithm as presented here did not use ancillary cloud masks, but it could easily incorporate them. Doing so would improve the quality of the results further and simplify the preprocessing by screening out troublesome data quickly. However, the algorithm incorporates a pixel-specific cloud filter in the form of the low-threshold X-bar charts. Provided that the training data are sufficiently clean to initialize the algorithm with a curve that models the phenology more than the noise, the built-in cloud mask appears to be sufficient.

Another question that arises is what to do with pixels which have signaled. At some point, the pixel needs to be reevaluated and reset to a no-change status, presumably with a new baseline. For this, sufficient re-training data need to be collected, ideally long enough after the disturbance that the pixel has stabilized somewhat. This is a constraint on any continuous monitoring approach to change detection, such as [11] in addition to the EWMA detection method. In other words, the method may be refined by allowing a "penalty period" after a change has been signaled, during which time incoming images may be used to recalibrate the baseline harmonic coefficients. This would enable the algorithm to be run continuously and to be adaptive, so that old changes do not linger until, or unless, the original baseline is restored.

One last area for future work presented here (there are undoubtedly many other areas not mentioned) would be expanding the application of the algorithm to other areas of the world. Our study area was restricted to pine stands in the southeastern United States. The algorithm itself is general in its operation, but it depends on having sufficiently dense image stacks on which to train the harmonic regression. In areas where the stacks are more sparse, it is uncertain how long the training period would need to be extended to get a workable baseline. With regards to other land surface features, the algorithm is not confined to forest-specific indices, and so one might

expect it to perform well across different features, provided that a relevant band or index is used. All of these challenges and questions provide fertile ground for further work.

3.6. Conclusion

Since the Landsat archive became available at no cost to the user, there has been a surge of interest in leveraging the temporal richness of the archive to aid in change detection. [2-3,5-7,9-11] These massively multitemporal approaches represent a fundamental shift in the paradigm of change detection. Rather than treating images as sparse and isolated snapshots, researchers are able to treat them as statistical observations of underlying land surface processes. Accordingly, it is natural to utilize well-developed methods like quality control charts in this area.

When working in the context of change detection using large stacks of images, there are a number of algorithm attributes one may consider desirable. Firstly, accuracy in the signaled changes is requisite. Additionally, the ability to detect more subtle changes, in our case forest thinning in addition to clearcuts, is very desirable. One might also wish to be able to screen for changes from recently acquired Landsat images, providing an up-to-date change map. It would be even more efficient to be able to do this using older maps as the basis for this change detection, yielding a dynamic and iterative process which refreshes itself with minimal additional processing. The EWMA detection algorithm, a method of using EWMA charts on the residuals of harmonic regression fits, possesses all of these features, at least, in the study area presented here. We have shown it to be accurate in this paper in terms of space, severity, and time. It not only detects thinning, it can also discriminate between differing degrees of disturbance, both in afforestation and deforestation. It typically signals these disturbances in the images immediately following the disturbances. By detecting subtler changes in forest cover, it allows for improved estimates in forest parameter changes across regional or broader levels. Given its sensitivity in the temporal and severity dimensions, it is possible that the algorithm may benefit monitoring and response to migrating insect species, although this remains to be tested. It may also permit more responsive tracking of vegetation stress due to drought, perhaps aiding in determining the greatest threats due to wildfire in a region, although again this has not been tested yet. The method operates in a data-driven manner, requiring only a global specification of the number of harmonics to be used in baseline estimation and the tuning parameters for the charts across the entire scene. By virtue of the design of EWMA charts, the process can easily incorporate new

images without reprocessing all of the history. This is possibly the most powerful feature of the method. It allows a smaller archive of data to be stored for tracking recent changes in the landscape, as all the historical information is preserved in the most recent disturbance map. All of these properties, taken together, indicate the great potential of this massively multitemporal on-the-fly method.

3.7. Acknowledgement

We would like to thank the researchers behind the NASA Earth Exchange (NEX) at Ames Research Center for their contribution of computing resources and programming feedback. Specifically, we would like to thank James McCreight of NEX for his help in parallelizing R efficiently. We would like to thank Karl Sorensen for his work in preprocessing the Landsat data through LEDAPS. We would also like to thank Bruce De Haan of The Westervelt Company for providing us with the polygons used to check the algorithm. We also want to thank the USGS for making the Landsat archive freely available; without that forward-looking decision, any sort of massively multitemporal method would be much costlier. This work was supported by the USDA Forest Service Cooperative Agreement with Virginia Tech (Grant No. 10-CA-11330145-158). It was also supported by the Landsat Science Team (USGS contract number G12PC00073), the Pine Integrated Network: Education, Mitigation, and Adaptation Project (PINEMAP, Coordinated Agricultural Project funded in 2011 by the USDA National Institute of Food and Agriculture), the McIntire-Stennis Cooperative Forestry Research program (USDA CSREES, Project No. VA-136614), and the Department of Forest Resources and Environmental Conservation at Virginia Tech.

3.8. References

[1] Law, B. E., and Harmon, M. E. (2011) “Forest sector carbon management, measurement and verification, and discussion of policy related to climate change.” *Carbon Management*, 2(1), 73-84.

[2] Asner, G. P., Knapp, D. E., Broadbent, E. N., Oliveira, P. J. C., Keller, M. and Silva, J. N. (2005) “Selective logging in the Brazilian Amazon.” *Science*, 310(5747), 480-482.

[3] Asner, G. P., Knapp, D. E., Balaji, A., and Páez-Acosta, G. (2009) "Automated mapping of tropical deforestation and forest degradation: CLASlite." *Journal of Applied Remote Sensing*, 3(1), 24p.

[4] Penman, J. (2003) "Definitions and methodological options to inventory emissions from direct human-induced degradation of forests and devegetation of other vegetation types." *Institute for Global Environmental Strategies (IGES) for the IPCC.*

[5] MDA Federal Company, persistent change detection website. http://mdaus.com/Geospatial/Global-Change-Monitoring.aspx

[6] Huang, C., Goward, S., Schleewis, K., Thomas, N., Masek, J., and Zhu, Z. (2009) "Dynamics of national forests assessed using the Landsat record: case studies in eastern United States." *Remote Sensing of Environment*, 113(7), 1430-1442.

[7] Huang, C., Goward, S., Masek, J., Thomas, N., Zhu, Z., and Vogelmann, J. (2010) "An automated approach for reconstructing recent forest disturbance history using dense Landsat time series stacks." *Remote Sensing of Environment*, 114(1), 183-198.

[8] Li, M., Huang, C., Zhu, Z., Wen, W., Xu, D., and Liu, A. (2009) "Use of remote sensing coupled with a vegetation change tracker model to assess rates of forest change and fragmentation in Mississippi, USA." *International Journal of Remote Sensing*, 30(24), 6559-6574.

[9] Kennedy, R., Yang, Z., and Cohen, W. (2010) "Detecting trends in forest disturbance and recovery using yearly Landsat time series:1. LandTrendr — temporal segmentation algorithms." *Remote Sensing of Environment,* 114(12), 2897-2910.

[10] Cohen, W., Yang, Z., and Kennedy, R. (2010) "Detecting trends in forest disturbance and recovery using yearly Landsat time series: 2. TimeSync — tools for calibration and validation." *Remote Sensing of Environment*, 114(12), 2911-2924.

[11] Zhu, Z., Woodcock, C. E., and Olofsson, P. (2012) "Continuous monitoring of forest disturbance using all available Landsat imagery." *Remote Sensing of Environment*, Landsat Legacy Special Issue, 122, 75-91.

[12] Brooks, E., Thomas, V., Wynne, R., and Coulston, J. (2012) “Fitting the multitemporal curve: a Fourier series approach to the missing data problem in remote sensing analysis.” *IEEE Transactions in Geosciences and Remote Sensing*, 50(9), 3340-3353.

[13] Shewhart, W. A. (1980) *Economic Control of Quality of Manufactured Product*. American Society for Quality Control, Quality Press. ISBN: 978-087389-076-2.

[14] Reynolds, M.R.J. and Cho, G.Y. (2011) “Multivariate control charts for monitoring the mean vector and covariance matrix with variable sampling intervals.” *Sequential Analysis*, 30(1), 1-40.

[15] Gupta, B. C. and Walker, F. W. Bloomfield, P. (2007) *Statistical Quality Control for the Six Sigma Green Belt*. American Society for Quality, Quality Press. ISBN: 978-0-87389-686-3.

[16] Montgomery, D. C. (2008) *Introduction to Statistical Quality Control*. 6th ed. Wiley. ISBN: 978-0470169926.

[17] Yu, J. and Liu, J. (2011) “LRProb control chart based on logistic regression for monitoring mean shifts of auto-correlated manufacturing processes.” *International Journal of Production Research*, 49(8), 2301-2326.

[18] Ning, X., Shang, Y., Tsung, F. (2009) "Statistical process control techniques for service processes: a review." 6th International Conference on Service Systems and Service Management. pp.927-931.

[19] Shah, S., Shridhar, P., Gohil, D. (2010) “Control chart : A statistical process control tool in pharmacy.” *Asian Journal of Pharmaceutics*, 4(3), 184-92.

[20] Steiner, S. H., Jones, M. (2009) “Risk-adjusted survival time monitoring with an updating exponentially weighted moving average (EWMA) control chart.” *Statistics in Medicine*, 29(4), 444-454.

[21] USGS Global Visualization Viewer (GLOVIS) website. http://glovis.usgs.gov/

[22] USDA NRC Geospatial Data Gateway website. http://datagateway.nrcs.usda.gov/

[23] Masek, J. G., Vermote, E. F., Saleous, N. E., Wolfe, R., Hall, F. G., Huemmrich, K. F., Gao, F., Kutler, J., and Lim, T.-K. (2006) "A Landsat surface reflectance dataset for North America, 1990–2000." *IEEE Geoscience and Remote Sensing Letters*, 3(1), 68-72.

[24] Crist, E. P. (1985) "A TM tasseled cap equivalent transformation for reflectance factor data." *Remote Sensing of Environment*, 17(3), 301−306.

[25] Powell, S. L., Cohen, W. B., Healey, S. P., Kennedy, R. E., Moisen, G. C., Pierce, K. B., and Ohmann, J. L. (2010) "Quantification of live aboveground forest biomass dynamics with Landsat time-series and field inventory data: A comparison of empirical modeling approaches." *Remote Sensing of Environment*, 114(5), 1053-1068.

[26] Blinn, C. E., Albaugh, T. J., Fox, T. R., Wynne, R. H., Stape, J., Rubilar, R. A., and Allen, H. L. (2012) "A method for estimating deciduous competition in pine stands using Landsat." *Southern Journal of Applied Forestry*, 36(2), 71-78.

[27] Rencher, A. C, and Schaalje, G. B. (2008) *Linear Models in Statistics*. John Wiley & Sons, Inc., 2nd edition. ISBN: 978-0-471-75498-5.

[28] Hintze, J. L. and R. D. Nelson (1998) "Violin plots: a box plot-density trace synergism." *The American Statistician*, 52(2), 181-4.

[29] Hollander, M., and Wolfe, D. A. (1999) *Nonparametric Statistical Methods*. Wiley Interscience, 2nd edition. ISBN: 978-0-471-19045-5.

[30] National Oceanic and Atmospheric Administration, national weather service weather forecast office website. http://www.srh.noaa.gov/bmx/?n=event_04272011tuscbirm

Chapter 4: Improving the Precision of Dynamic Forest Parameter Estimates Using Landsat

Evan B. Brooks[a], John W. Coulston[b], Randolph H. Wynne[a], and Valerie A. Thomas[a],

[a]Department of Forest Resources and Environmental Conservation, Virginia Tech, Blacksburg, VA, USA
[b]USDA Forest Service Southern Research Station, Forest Inventory and Analysis Unit, Knoxville, TN, USA

This work has not yet been submitted

Abstract

The use of satellite-derived classification maps to improve the precision of post-stratified forest parameter estimates is well-established, with such maps largely being based on data relevant to the parameter in question. When reducing the post-stratification variability in estimates for forest productivity variables such as forest growth, it is thus logical to use a productivity-related stratum map. At the stand level, time series of Landsat images are ideally suited for producing such a map. While other Landsat-based techniques such as the Vegetation Change Tracker (VCT) have been employed to this purpose by indicating the time of forest disturbance, they have generally shown only slightly improved relative efficiencies (REs) over taking a simple random sample (SRS). In this study, we generate a stratum map based on the trajectory of forest recovery, as measured in the Normalized Difference Vegetation Index (NDVI) derived from Landsat TM data from 1985 through 2011, after a disturbance over a period of six years. These trajectories are classified according to a hierarchical clustering algorithm from a training sample, resulting in classes that resemble site index curves. The resulting stratum map is then used to calculate the relative efficiencies of the method for forest parameter estimation in an Alabama, USA study area. In particular, REs above 1.2 were observed for each of the seven parameters being estimated. In the growth parameter, an RE of 1.5 was observed. However, a number of potential confounding factors are recognized in the analysis. These factors ranged from issues as simple as insufficiently large study area to challenges in taking nonlinear NDVI values and scaling them effectively. The recognition of such factors offers several directions for further refinement of the approach. This technique of

using forest recovery strata has promising implications in the area of monitoring and modeling forest productivity.

4.1. Introduction

The use of satellite and other remote sensing data to supplement *in situ* measurements of forest parameters has been established for decades. [1-13] There are many advantages to the use of satellite data, among them being increased coverage, lower cost, and more frequent (and regular) measurement, allowing them to be used to scale up field measurements to estimate regional forest biophysical parameters.

These parameters may be categorized into two groups. *Static* parameters reflect the condition of the forest at a single point in time. Examples of these include percent carbon present, canopy cover, tree height, and diameter at breast height. In contrast, *dynamic* parameters reflect the manner in which a forest changes over time. Examples of these include net primary production, forest growth, removal, mortality, and carbon sequestered over time.

The Forest Inventory and Analysis (FIA) program of the USDA Forest Service is one organization that uses satellite data to augment its field measurements. [1,3-9,17,18] By stratifying observations from field plots based on satellite-derived maps, the FIA program is able to obtain more precise estimates of forest parameters than would have otherwise been possible or financially feasible. Because of this, one area of interest to researchers has been developing methods of stratification that best improve such precision. A general background on stratification follows.

4.1.1. Background on Post-Stratification

The standard method for improving the precision involves the statistical technique of *post-stratification*. [12, 16, 17] In *stratified sampling*, a population is divided into mutually exclusive *strata*, and within each stratum a random sample is taken. It is assumed that the strata represent distinct subsets of the population which are sufficiently different as to warrant the separate treatment. An example from remote sensing of natural resources would be estimating some parameter based on the land cover type, in which case one would stratify along the classes of land cover. These classes would be assigned from remote sensing data. In general, the number of elements sampled from each stratum is proportional to the relative size of that stratum with respect to the full population. In this manner, we are able to obtain an overall sample that better

represents the characteristics of the population. From this, we expect the estimates we obtain from the sample to be more precise than those computed from a simple random sample.

However, in many cases, the stratification can only take place *after* the sample is taken. By assigning weights to the newly generated strata relative to the respective proportions of the strata in the sample, we can obtain a framework that yields results nearly as precise as proportional stratified sampling. [16]

Post-stratification is a common technique in remote sensing. [1-15, 18] Parameter estimations that involve the use of established plot networks, such as FIA, are in effect post-stratified estimations. Because the characteristics of the plots (volume, average height, percent canopy cover, growth, mortality, etc.) are unknown until after the plots have been established, we must use a post-stratification framework to make the sample more precise. Even validating an image classification with an accuracy assessment is a form of post-stratified sampling, since the randomly chosen points in the image are classified based on the image-derived strata. In this last example, the congruence of the terms *strata* and *classes* is obvious.

The case that is relevant to us, and the object of this paper, is that of stratifying the FIA plots in such a way that we improve the precision of *dynamic* parameter estimates. Since the FIA plots are distributed over the land surface of the USA, the information containing the strata is best summarized in a thematic map or raster. In such a stratum map, each pixel is assigned a code value representing one stratum of the post-stratification solution.

4.1.2. Post-Stratification and the FIA Program

The FIA program is a monitoring organization. Because the FIA plots are fixed in location, and because the strata along which the plots are most effectively grouped change over time, the practice of using post-stratification techniques to improve precision in FIA plot-based estimation is extensive. The Rocky Mountain Research Station has used post-stratification techniques in conjunction with remote sensing, starting with aerial imagery, since 1965. [17] The practice of using aerial imagery in a two-phased post-stratification approach continued for decades. [3-4] However, the amount of cost, time and effort required to interpret the aerial images, coupled with their relative temporal sparseness, led the FIA programs to look into using digital and satellite imagery in lieu of aerial photos around the turn of the millennium. [3-4] In the decade that followed, researchers enjoyed success in cheaply improving the precision of many static forest parameter estimates using Landsat and its products (such as the National Land Cover Dataset).

[1-9] In particular, methods now exist which employ Landsat-based data to reduce the variability of static estimates such as forest area and stand volume. This is in part due to the spatial resolution of the Landsat Thematic Mapper (TM) sensors, 30 meters to a pixel side. At this resolution, trees are not resolved, but stands typically are. A method developed by the Northeast region of the FIA program stratifies pixels according to the number of forested pixels in a 5x5 neighborhood, resulting in a variety of strata based on the density of forested pixels around the target pixel. [1] The North Central region developed another approach that defines four strata of forest, forested edge, nonforested edge, and nonforest. [1] Both methods are capable of improving the precision of static parameter estimates.

Despite the advances in using Landsat data and products as a source of stratum information for static parameter estimates, there some difficulty in applying these techniques to dynamic parameter estimates. This is in part due to previous limitations in data archiving and processing power required to work with decades' worth of Landsat images, in part due to the fact that until 2009, the high cost associated with ordering so many images, and also due to the dynamic nature of the parameters. In order to derive strata maps suited to dynamic parameters, multiple images are needed to capture the essence of the change. Strata may be based on such variables as the time and severity of the change, or possibly on the pattern of regeneration after such a disturbance. Methods such as the Vegetation Change Tracker (VCT) [14], LandTrendr [15], and control chart disturbance (or EWMA) detection [25] can produce stratification criteria for time and severity of disturbance, but they may not be sufficient to provide regrowth information.

Accordingly, the question to be addressed here is two-pronged. Is it possible to use Landsat-derived post-disturbance time series to improve the precision of post-stratified FIA parameter estimates, particularly for dynamic parameters? If so, what methods of analyzing such time series are most effective?

In this paper, we will present variations on a single method that specifically stratifies FIA plots based on behavior after a disturbance. The variations use Landsat data over a multi-year timeframe to classify pixels, resulting in stratification maps that resemble a site index map. By employing these maps with FIA field data, we hope to produce more precise estimates of dynamic forest parameters than before, and in so doing, determine which variations are most effective at improving this precision.

In general, the method, by showing the form that forest regeneration takes, may give us the ability to map forest growth for large regional areas and to spatially profile the dynamics of above-ground biomass, which for practical purposes may be interchanged with volume or carbon. This information could potentially be used in climate models, regional and local policymaking, and plantation management, to name but a few examples. It is also our hope that by acting as a proxy for the site index, the stratum maps produced by the method might also be employed anywhere that site index is, in a format that is broad in coverage and digital in nature.

4.2. Data

For this study, we were interested in testing the ability of a regrowth-based method to discriminate between plots along site-related lines. For this reason, we chose a study area in west-central Alabama, USA, shown in Figure 4.1. The area is a subset of the coverage of Landsat path/row 21/37, allowing us to work with one scene only for simplicity. The area is largely forested (about 70% based on visual observation of images from 1985), and the dominant forest type is loblolly pine (*Pinus taeda*).

We also wanted to have a sufficient timeframe to allow for many instances of stand disturbance and ample recovery time after such disturbance. Since the coverage timeframe for Landsat 5 spanned the years from 1985 to 2011, we chose to use that timeframe in order to have a consistent sensor platform to work with, although we could have added information from Landsats 4 and 7 as needed.

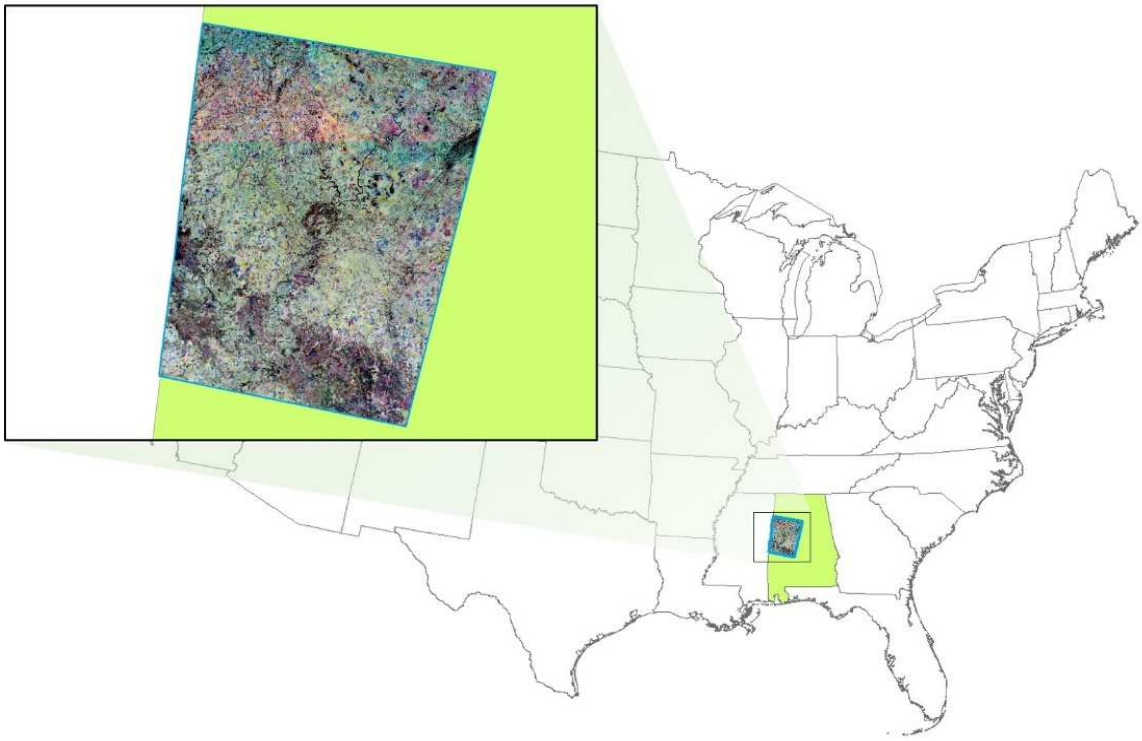


Figure 4.1. Study area. Inset RGB are the mean normalized difference vegetation index (NDVI) values for 3-year groups centered on 1985, 1988, and 1991, respectively.

4.2.1. FIA Plot Data

For purposes of making dynamic parameter estimates, we used FIA plot data corresponding to the spatial extent of the path/row 21/37 scene described above. The plots, which are on a variety of land classes and ownerships, are classified in accordance with the 2000 amendment to the US Department of the Interior and Related Agencies Appropriations Act. In our case, we had a total of 977 FIA plots in our study area.

Each FIA plot is constructed according to the following pattern, detailed in [12]. From the centerpoint of the plot, a circle of radius 35.58 meters (120 feet) is drawn out. At equidistant points along the circumference of this circle, three points are marked out. From each of these points as well as the centerpoint of the plot, a circle of radius 7.31 meters (24 feet) is drawn, yielding four smaller circles arranged around and within the larger circle. For comparison, the overall FIA plot size (about 675 square meters, or 1/6 acre, total for all four circles) is comparable to a 3x3 collection of Landsat 30-m pixels if the center of the neighborhood matches the center of the plot. Correspondingly, this is about $1/9^{th}$ of a single 250-m MODIS pixel.

Within each of the circles, forest measurements are taken, including but not limited to volume, heights, diameters at breast height, and number of trees. [19-20]

The FIA plots' initial positions were chosen according to random placement within a hexagonal grid. [12, 19] Historical measurement intervals vary by plot and by state, but in general the plots in our study area are measured according to a rotating panel design, with the panels being systematic samples of the area with an ideal rotation interval of 7 years, frequently enough to provide some measurements of growth and other dynamic parameters.

4.2.2. Satellite Data

As stated previously, we used Landsat 5 TM data covering our study area. (Figure 4.1) We chose a timeframe over the service life of Landsat 5, from 1985 through 2011. The temporal distribution of the images is shown in Figure 4.2. Note how for any consecutive 3-year period, there is a fair representation of dates across the course of a meta-year. For simplicity, we used only Landsat 5 as a data source, although Landsat 4 and 7 could have been used as well. We chose only scenes with 10% or less nominal cloud cover to reduce incoming noise in the time series. This subset comprised 174 images in all, downloaded from the USGS GLOVIS website [21].

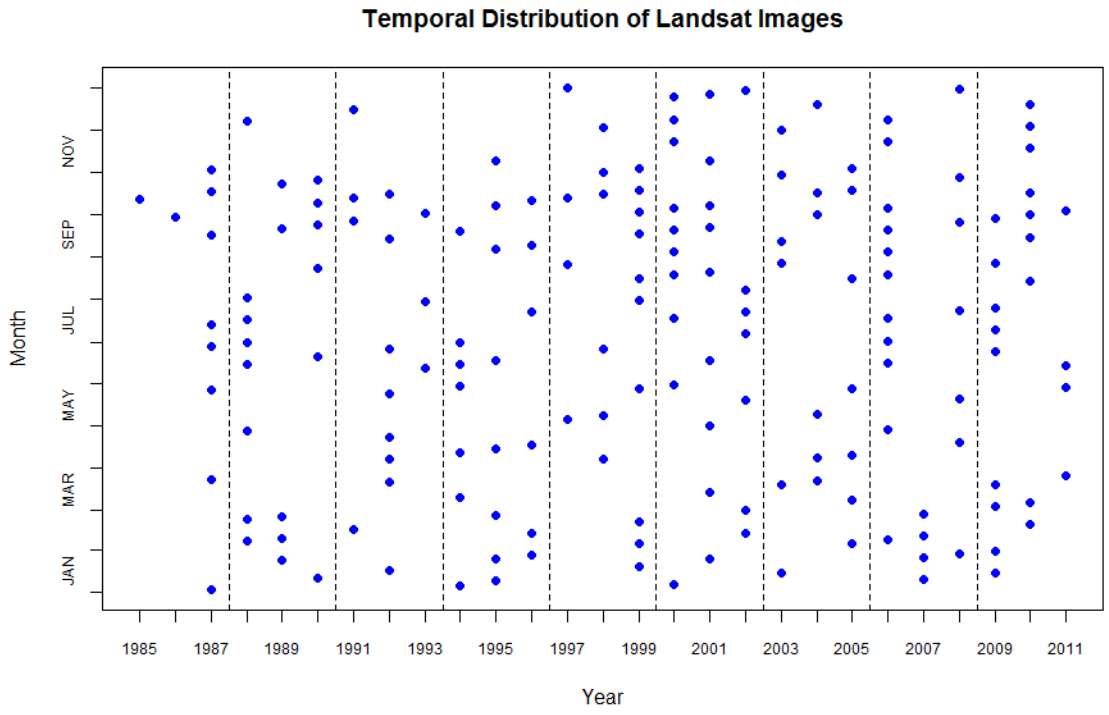


Figure 4.2. Temporal distribution of Landsat 5 images.

Upon downloading, we preprocessed each image. First, we passed the scenes through the LEDAPS algorithm [23] to adjust the images for radiometric interference due to aerosols by converting them to surface reflectance. To further calibrate the images, we also used a band-minimum based dark object subtraction, using the water bodies in the scene as the calibration objects.

Once the images had been preprocessed, we computed Normalized Difference Vegetation Index values (NDVI) [27] for each image. For simpler archiving and processing, we multiplied all the values by 1000 and rounded to the nearest integer. The resulting image stack was our basic data for stratum map generation.

In the context of this research, observations in the form of FIA plots are grouped by a Landsat-derived stratum map to provide post-stratified estimates of the forest parameters. The specific details and approaches to the process in this paper are given in the next section.

4.3. Methods

As with the data, there are two components to the method presented here. One component is the post-stratified estimation of parameters, and the other is the generation of a stratum map based on growth after disturbance. Both components follow.

4.3.1. Post-stratified sampling and estimation

The use of post-stratified sampling in an FIA context has been standard for decades. [12,17] Our basic method, outlined by [16] and refined for FIA use by [12], is summarized as follows, keeping with the general formulation of [12]. All of the equations that follow in this section are based on those of [12] as well. As a general rule, the population estimates are made by estimating means per land area from the observations (FIA plots) and multiplying these means by the total area.

Consider a study area with total area A. From this area, take a sample of n observations of a variable y divided among H pre-existing strata (derived in our case from Landsat data) so that the h^{th} stratum contains n_h observations, denoting the observations in the h^{th} stratum as $y_{1h}, y_{2h}, \ldots, y_{n_h h}$. Define w_h, the weight of the h^{th} stratum, as the proportion of the study area assigned to that stratum. We treat the weights as known quantities due to the large number of classified Landsat pixels in the study area. Then we have the total stratified estimate as the weighted sum of the within-stratum total estimates:

$$\hat{y}_{st} = A \sum_{h=1}^{H} w_h \bar{y}_h \quad (4.1)$$

where

$$\bar{y}_h = \frac{1}{n_h} \sum_{k=1}^{n_h} y_{kh} \quad (4.2)$$

We approximate the variance estimate of this stratified total as the sum of weighted squares of the within-stratum variances estimates:

$$Var(\hat{y}_{st}) = \frac{A^2}{n} \left[\sum_{h=1}^{H} w_h n_h Var(\bar{y}_h) + \sum_{h=1}^{H} (1 - w_h) \frac{n_h}{n} Var(\bar{y}_h) \right] \quad (4.3)$$

where

$$Var(\bar{y}_h) = \frac{1}{n_h - 1}\sum_{k=1}^{n_h}(y_{kh} - y_h)^2 \quad (4.4)$$

The precision of the estimate from (4.1) is the inverse of (4.3). In this case, the stratum weights are considered known. This is consistent with the infinite population framework used by the FIA. In order to compare this with the precision obtained from an estimate based on a simple random sample (SRS), we divide one by the other to arrive at the relative efficiency of the post-stratification (RE), given by

$$RE = \frac{Var(\hat{y}_{SRS})}{Var(\hat{y}_{st})} \quad (4.5)$$

We calculate the numerator by setting $H = 1$ and $w_h = 1$, assigning the entire sample to a single stratum, and reapply Equations (4.1-4.4). [12]

In general, the higher the value of (4.5), the more effective the stratification method is at increasing the precision of the estimate of the parameter. One may intuitively think of RE as a measure of how much larger a simple random sample would need to be in order to achieve the same precision that the post-stratified sample produced.

Note that as a rule, the FIA program requires at least five plots assigned to a stratum [1], with a minimum of 10 plots per stratum recommended for stability in the parameter estimates [18]. This effectively limits the number of strata available. Ideally, we want the strata to be maximally homogenous within strata and maximally heterogeneous among strata.

4.3.2. Stratum Map Generation

The criteria that strata should be as self-contained and separate from each other as possible is the same criterion underpinning cluster analysis. By identifying strata with clusters, it thus follows that a cluster analysis will produce useful stratum maps, but the question becomes one of what objects to cluster. Because we are estimating a dynamic variable in this study, the object to be clustered, the *clustering object*, should attempt to capture this dynamism.

In general, a forest stand grown from planting will grow most quickly in the first few years after planting, as evidenced by the rapid increase in leaf area index (LAI) in these first years. [31] Because of this, our objectives are to:

1) Identify the point at which the forest pixel is minimally vegetated, and

2) Construct the clustering object based on the subsequent observations from that point.

We used a very simple method of identification for disturbances. For a time series of mean annual NDVI values, we took the year for which the sharpest decrease in NDVI is measured to be the year of disturbance. From there, we took this year as well as 6 subsequent years as our basic data for clustering.

4.3.3. Mean generation

We used two basic methods of defining the mean NDVI each year in the timeframe, with the intent to compare the effectiveness of each. Both methods were based on using a modified harmonic regression algorithm to maximize the robustness of the resulting mean against seasonality and temporally scattered data. [24-25]. A brief summary of the method follows here; for full details, we suggest referring to [25].

For a given year, we collected all images with minimal cloud cover from that year and both the year preceding it and succeeding it. For example, in 1989, we took all minimally-clouded Landsat images from 1988-1990, as shown in the second column of Figure 4.2. From such a collection of *d* images over days of the year (from 1 to 365 or 366, depending on the year) with date vector

$$\underline{T}_{d\times 1} = [t_i]_{\,i\in\{1,2,\ldots,d\}} \quad (4.6),$$

for each pixel *p* denote the corresponding time series of NDVI values for that pixel as

$$\underline{V_p}_{d\times 1} = [v_{pi}]_{\,i\in\{1,2,\ldots,d\}} \quad (4.7)$$

We will use the vector notation in the rightmost term where convenient for compactness, noting that the *p* subscript emphasizes the vector's dependence on the pixel in question. For simplicity, scale T by converting days of the year to values on $[0,2\pi]$ by multiplying by $\frac{2\pi}{365}$, using this scaled vector to generate an input matrix, $M_{d\times(1+2m)}$, with columns corresponding to the constant and the sines and cosines of increasing frequency to m, the number of desired harmonics.

Let us assume that a correct linear model specification for the time series by day of the year is given by a harmonic series with *m* harmonics and independent identically distributed normal errors, such that

$$\underset{d\times 1}{V_p} = M_{d\times(1+2m)}\underset{(1+2m)\times 1}{\beta_p} + \underset{d\times 1}{\varepsilon_p} \qquad (4.8),$$

We then estimate the pixel-specific harmonic coefficients

$$\underset{(1+2m)\times 1}{\hat{\beta}_p} = \left[\hat{a}_{p0} \quad \hat{a}_{p1} \quad \hat{b}_{p1} \quad \cdots \quad \hat{a}_{pm} \quad \hat{b}_{pm}\right]' \qquad (4.9)$$

where transposition is denoted by $'$, by the usual least squares method.

Typically, a pixel may display anomalous values corresponding to small-scale cloud cover, shading, or other short-lived events that are not desirable in the model. Therefore we process the residuals from the above model in a low-threshold Shewhart X-bar chart [25, 29-30], using the remaining values to recompute the adjusted regression coefficients. This process is illustrated on a sample pixel in Figure 4.3.

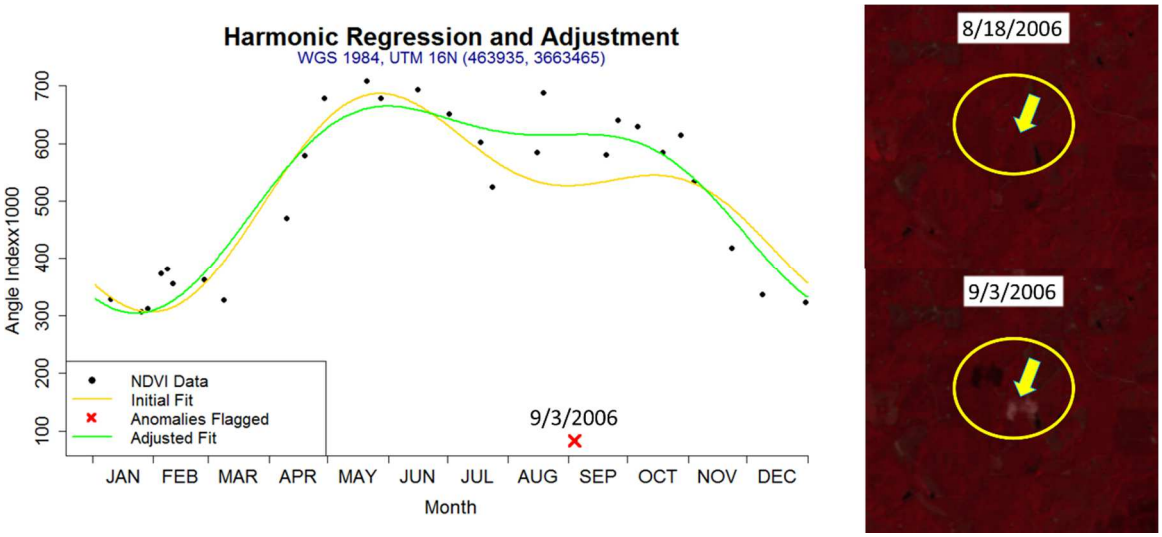


Figure 4.3. Modified harmonic regression algorithm on an NDVI time series for an example pixel.

At the end of this process, for this paper, we extract only the constant coefficient, $\hat{a}_{p0}$, from the second iteration of this process. We denote this value as the "mean" value for that pixel for that year.

In our specific case, we used 2 harmonics and a Shewhart X-bar threshold of $L = 2$. We performed this process for each pixel in the study area independently. We employed the algorithm for each target year.

Recall that we used two different methods in generating time series of mean NDVI for this paper, both of which depend on the harmonic regression algorithm. In reality, the distinction between the methods lies only in whether to allow overlapping of the input years. In one of the mean-generation methods, we grouped our images from 1985-2011 into 3-year segments, so that the first segment was 1985-1987, the second was 1988-1990, and so on. This yielded a 9-point time series of means (3-year groups over 27 years) for each pixel in the scene. From this, we took the point in each series which represented the sharpest decrease in mean NDVI. That is, if the time series of means is denoted $\{\tau_{p1}, \dots, \tau_{p9}\}$, then the point we took was simply the value $\tau_{pi}, i = \{2,3, \dots ,7\}$ for which $\tau_{pi} - \tau_{pi-1}$ was minimal. Note, by the way, that $\tau_{pi} = \hat{a}_{p0}$ for the corresponding year of calculation. We did not consider disturbances in the last two points of the time series because we desired two points in the series subsequent to the disturbance to get 6 years' worth of data after the disturbance. The output from this method was a three-layer raster of values. The first layer was the NDVI value at disturbance, and the second and third layers were NDVI values three and six years after disturbance, respectively.

In the other mean-generation method, we simply performed the method for each available year: 1986, 1987, etc., through 2010. This resulted in a time series of 25 points instead of 9, allowing potentially more detail in the time series. As before, we identified the disturbance as the year for which the decrease in mean NDVI was the sharpest. To be consistent with the temporal scope of the first method, we took the disturbance and the subsequent six years, resulting here in a seven-layer raster of NDVI values. Thus, we did not consider disturbances indicated in the last 6 years of the time series. Figure 4.4 illustrates both methods on three example pixels. All three pixels are in loblolly pine (*Pinus taeda*) stands, with one being sparser than the others. All three show some level of harvesting over their histories, although the degree and timing of the harvest vary.

From Figure 4.4, note that the two methods agree on the years for which the 3-year group means were calculated. Further note that the running means often show the disturbance as being at a slightly different time than the 3-year group means, so the two methods are typically producing different data for later stratification. One last point we wish to make here is that while

the approach of steepest descent seems to identify disturbances well for the 3-year means, the extra detail provided by the running means suggests that another method is perhaps more appropriate if the minimum value of the disturbance is targeted. A possible alternative would be a disturbance signal based on the concavity or local minima of the time series, so that the resulting subset of the series only deals with increasing NDVI values at the outset. For the purposes of this paper, we used the steepest descent approach for both means sets, under the assumption that the classification would still be able to discriminate regrowth patterns in the subsequent years within the running means groups.

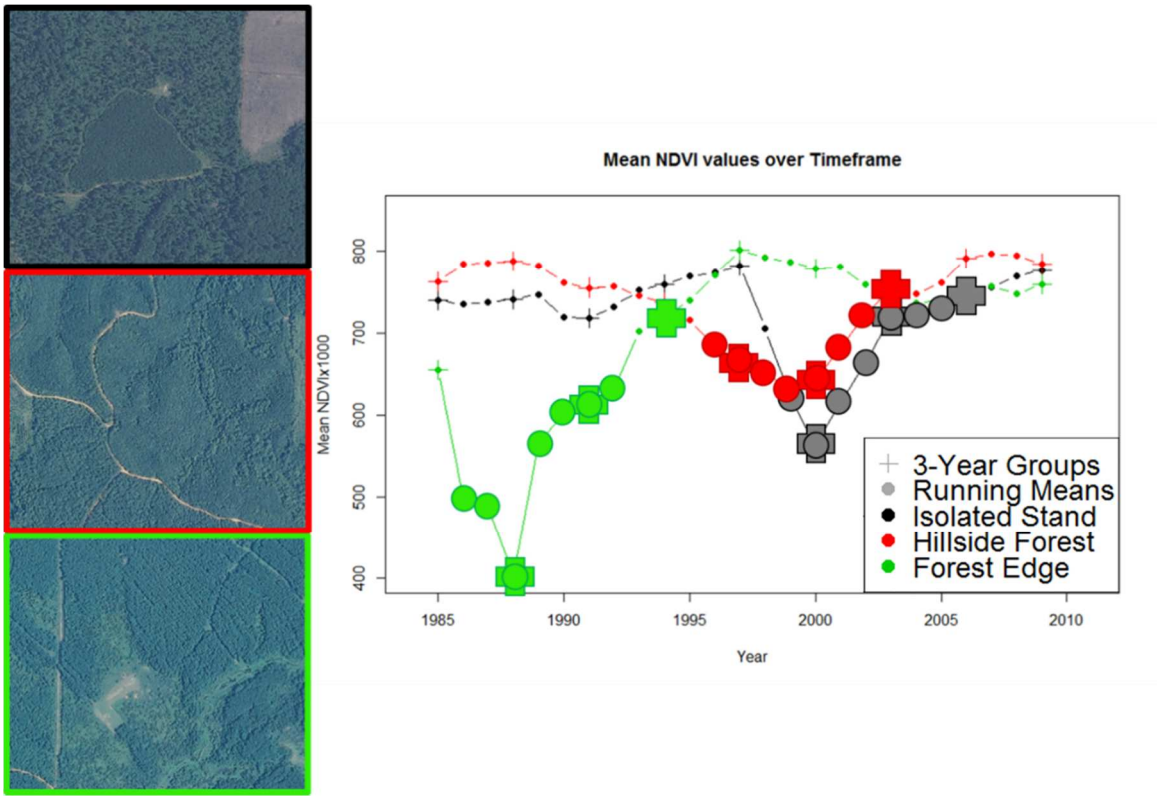


Figure 4.4. Running mean and 3-year mean generation methods, for three example pixels from different pine stands. It is evident from the time series that each stand underwent a harvest of some degree, although the degree and timing of each harvest varied. Emphasized plot characters show which values were extracted for further classification. Note that the two methods of generating means do not necessarily agree on which years to use.

4.3.4. Cluster object generation

Regardless of the method used in generating the post-disturbance mean raster, there are a multitude of variations by which the second objective, generating the actual clustering object,

can be achieved. We chose to focus on three transformations, illustrated in Figure 4.5, of the time series segments shown in Figure 4.4. In the first transformation, we simply took the means raster as it was, shown in Figure 4.5a. In the second transformation, we subtracted the value of the first layer (that at disturbance) from all of the layers. This had the effect of showing relative changes from the disturbance, allowing similar magnitudes of regrowth to be closer, as shown in Figure 4.5b. The third transformation involved dividing the differenced time series from the second transformation by the most extreme value in each series. This normalization forced the chief distinction between time series to be the shape of the time series, rather than the magnitude. This transformation is shown in Figure 4.5c. The key point in using different transformations was that under each transformation, the notion of which time series are similar can vary.

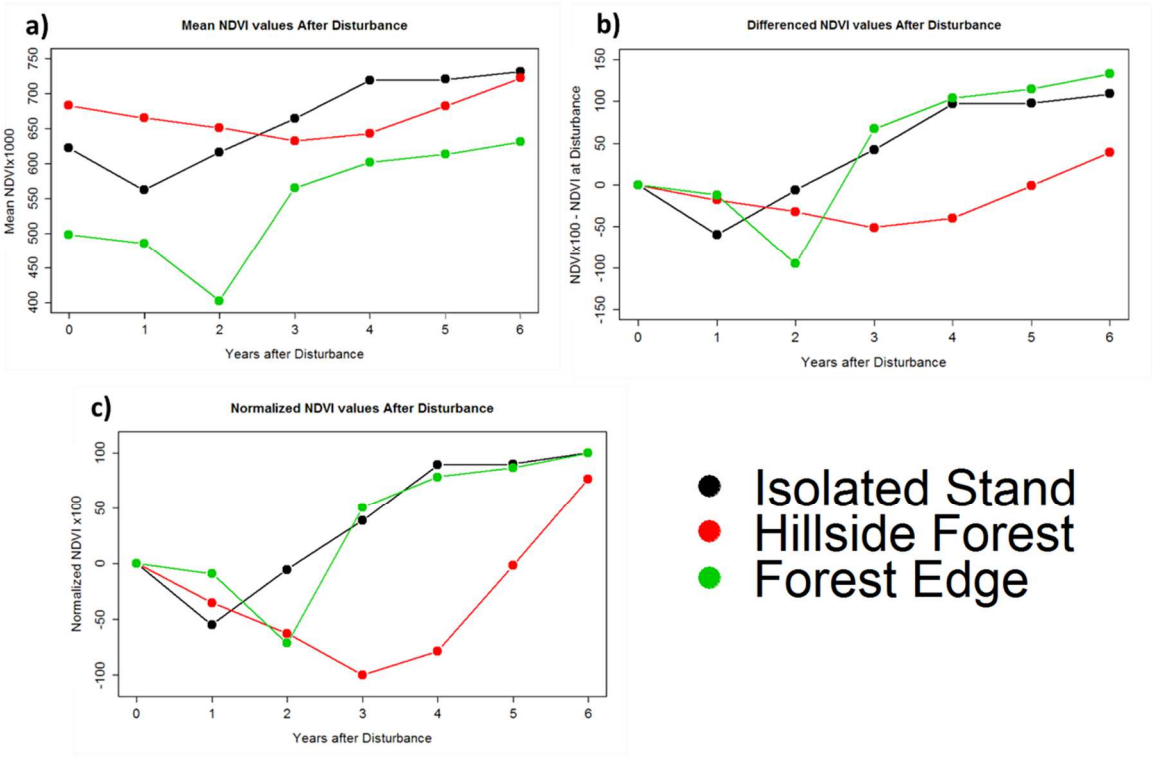


Figure 4.5. Post-disturbance time series for the three pine pixels shown in Fig. 4.4. Each time series represents the year of estimated disturbance, as well as the subsequent six years. The graphs depict three methods of transforming the data: a) NDVI means, b) differenced on the NDVI mean at disturbance, and c) normalized by the most extreme deviation from the NDVI at disturbance. Note how different time series appear closer depending on which transformation is used.

4.3.5. Cluster analysis

Whatever the object type we used for the regrowth raster, we used it as the input in an algorithm based on hierarchical clustering analysis (HCA). The specific details of the HCA algorithm are given below, with the concept shown in Figure 4.6.

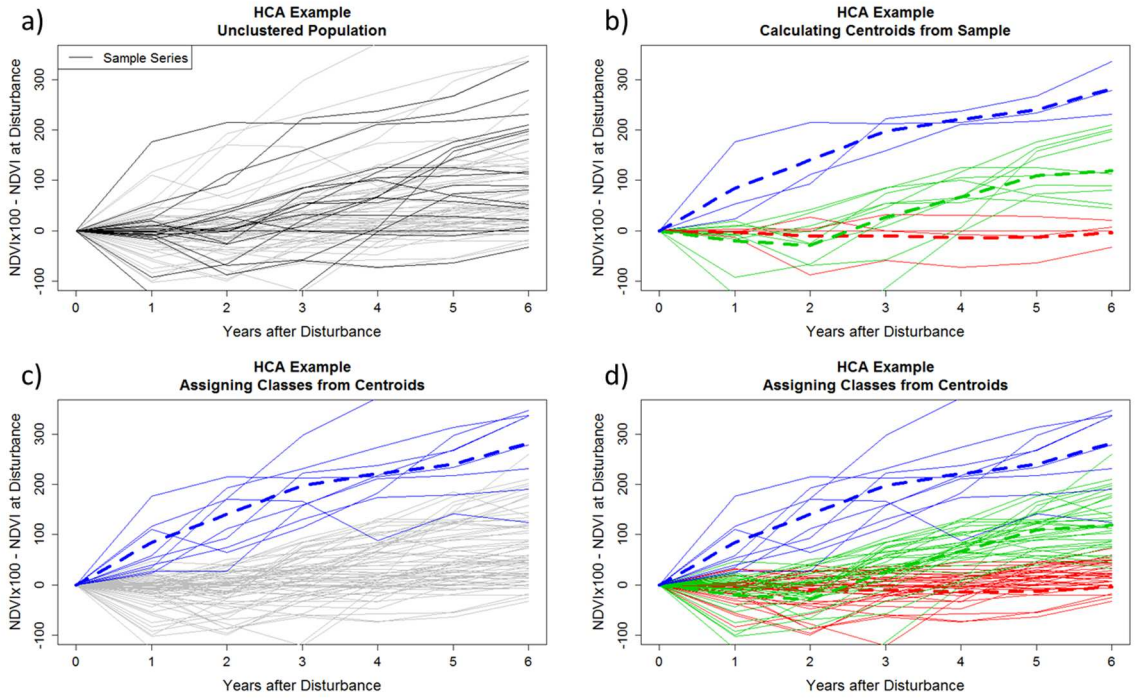


Figure 4.6. HCA algorithm, demonstrated on a sample of differenced running means time series for three clusters (colors). a) From the full population, take a random sample (sample in black). b) Classify this sample according to self-similarity, and compute cluster centroids (dashed lines). c,d) In the original population, classify each element based on the nearest centroid.

Consider an image with P pixels and B information elements per pixel. These can be brightness values corresponding to bands on a Landsat scene, multiple vegetation index values over time, or any other combination of variables. In our case, these elements are the regrowth values derived from the NDVI analysis. Denote the *information* on the k^{th}pixel as the column vector

$$\underline{p_k} = (x_{k1}, x_{k2}, \dots, x_{Bk})^T, \qquad k \in \{1, \dots, P\} \qquad (4.10)$$

Take a subset of pixels. (Figure 4.6a) This subset may be randomly generated or deliberately chosen to capture expected variation in each of the desired informational classes. Without loss of generality, suppose that $\{\underline{p}_1, \underline{p}_2, \ldots, \underline{p}_S\}$ constitute the sample.

For this sample, create a distance matrix. In this case, we used the Manhattan, or taxi-cab, distance (L1 norm), given by

$$\|j,k\|_1 = \left(\left|\underline{p}_j - \underline{p}_k\right|\right)^T \underline{1} \qquad (4.11)$$

The Manhattan distance has two key advantages over the Euclidean distance (L2 norm) in this context. Firstly, Manhattan has fewer multiplications, making it computationally more efficient. Because deviations are summed and not squared, it tends not to penalize single-value aberrations as heavily as the Euclidean distance.

Once the distance matrix has been created, perform HCA on the sample to generate a solution set of classes (or strata). (Figure 4.6b) HCA agglomerates individual elements into clusters by determining the distances from cluster to cluster and linking clusters based on some function of the distances. There are a variety of linkage methods available for HCA, including single linkage, complete linkage, average linkage, etc., but we used Ward's method [26]. Ward's method links clusters based on minimizing information loss in the form of sum-of-squares errors (not to be confused with the sum-of-squares distance between individual pixels). Practically, the method tends to be more computationally efficient than other methods, and it tends to produce clusters that are well-defined and "convex" in the information space.

One of the biggest advantages to HCA is the ability to save cluster solution sets for any specified number of clusters or range of "cluster sets". In a single analysis, the user may actually generate sample maps for as many solution sets (i.e. a map with 4 clusters, one with 5 clusters, etc.) as he or she desires.

Once the sample classification has been completed, we fill out the remaining pixels in the map by a nearest-neighbor approach to the cluster centroids. (Figure 4.6c, d) If we have K clusters in the map, denoted $C_1, \ldots, C_K$, then the centroid of the i^{th} cluster is

$$\underline{c}_i = \text{average}_{k \in C_i}\left[\underline{p}_k\right] \qquad (4.12)$$

Then for each pixel $\underline{p}_k, k \in \{1, \ldots, P\}$, we assign the cluster (stratum) to which the pixel is closest, again by the L1 norm. That is,

$$\mathrm{str}\left(\underline{p}_k\right) = \mathrm{index}\left(\min_{i\in\{1,\ldots,K\}} \left\|\underline{c}_i, \underline{p}_k\right\|_1\right) \qquad (4.13)$$

This process is repeated for each solution set specified by the user. This yields a raster wherein each layer depicts a different stratum map, one for each solution set.

Note that the method of assigning clusters to the map allows for shifts to occur in the class assignments of the sample points. Measuring the number of sample points thus shifted provides a diagnostic for the appropriateness of the number of spectral classes, as more shifts indicate that the spectral classes are not particularly separable. Additionally, we can use the minimum distance to the cluster centroids as a diagnostic. If a pixel has a large minimum distance to its assigned cluster centroid, then this suggests the need for additional clusters. This diagnostic is appropriate because Ward's method tends to produce convex clusters, making distance to the centroid a useful measure of proximity.

To review, the algorithm took a randomly chosen training sample from across the scene. For consistency, we used the same training sample for each of the six object generation methods. For each method, we computed the distance matrix for the objects in the training sample and used this to perform an HCA. We then calculated the centroids for the solution sets (4 clusters, 5 clusters, etc.) from the 4-cluster solution to the 12-cluster solution. For each solution set, we then computed the Manhattan distance from each cluster centroid in the solution set to each pixel in the scene. Cluster assignments were made based on minimum Manhattan distance to the pixel's regrowth values. Because the actual hierarchical clustering takes place on a relatively small subset of the scene, this approach allows for many of the benefits of HCA at a much faster computation time.

4.3.6. Specific application

In our case, for each map of cluster objects (six in all), we used a sample size of 10,000 pixels (less than .01% of the scene) to generate solution sets for the clusters ranging from 4 clusters to 12 clusters. We chose the training sample at random across the scene, using the same sample to produce cluster centroids for each solution set in order to control the results for training variation. Recall that in this context, *cluster* is interchangeable with *stratum*. The result of the nearest-centroid assignments for each cluster object and solution set was a single-layer stratum map. With the six different cluster object types and nine solution sets for each type, we obtained 54 different stratum maps over the same area, all using the same training sample.

The chief response measured in this study is the relative efficiency, or RE, of the stratification, calculated above in Equation (4.5) as a ratio of the variances of total estimates for a given parameter, such that the larger the RE, the more effective the stratification is at increasing the precision of the estimated total, compared to a simple random sample (SRS). Intuitively, the RE can be thought of as the percent increase in size for an SRS to achieve a variance on the total estimate similar to the stratified result. For example, an RE of 2 implies that the SRS sample size would need to be doubled to achieve the precision shown in the stratification, while an RE of 4 indicates that a quadrupling would be necessary. Recall that the RE is a ratio of variances, not standard deviations: an RE of 4 would correspond to the usual intuition that quadrupling the sample size of an SRS results in half the expected standard deviation. In the context of FIA plot-based parameter estimation, especially for dynamic parameters, there is no way to simply take more plots over an area. Those plots are fixed, and so post-stratification is a valuable tool for getting the most precise estimates from such constrained sampling.

With the different methods outlined above, our experimental setup could be considered in a 3-way ANOVA context, allowing for some comparisons between each cluster object method. Here, the factors were the mean type (running or 3-year), the form of normalization (non-normalized, differencing, full normalization), and the number of strata in the map (4 through 12). Table 4.1 summarizes this design. In this case, we had only one replication of each of the 54 treatment combinations. We could have increased the number of replications by taking multiple cluster training sets and recomputing the subsequent REs, but for the purposes of simple comparison between methods, 54 strata maps were sufficient.

Table 4.1. Experimental factors in the study.

Factor	**Levels**
Mean Type	*3-year groups*
	Running means
Normalization	*Non-normalized*
	Differenced
	Normalized
Number of Strata	*4, 5, 6, 7, 8, 9, 10, 11, 12*

In all, there were seven forest parameters of interest that we computed REs for. These parameters are shown in Table 4.2. Four of them are static parameters, and three of them are dynamic in nature.

Table 4.2. Forest parameters of interest in the study.

Parameter	Type
Carbon (tons)	Static
Forest Area (acres)	Static
Cut Area (acres)	Static
Planted Area (acres)	Static
Removal ($feet^3$/year)	Dynamic
Mortality ($feet^3$/year)	Dynamic
Growth ($feet^3$/year)	Dynamic

The population in this context is the land in the study area. We took our main sample to be the 977 FIA plots in the study area, each plot representing one observation from the population. The dynamic parameters are based on individual tree measurements, where trees are defined as having a diameter at breast height of at least 5 inches. For each of the 54 method treatments, we assigned to every plot the stratum value for the pixel containing the center of the plot. For each of the parameters of interest, we computed the total estimates and associated variances of those estimates by applying (4.1-4.5) to the observations. We did this both by stratum, letting the stratum weights be the proportion of the study area classified into each stratum, and for all of the observations in a single "stratum" to represent a simple random sample. By dividing the variance estimate from the simple random sample by the variance estimate from the post-stratified approach, we obtained a single RE for each parameter for each treatment combination.

4.4. Results

4.4.1. Main results

With 54 REs to present for each of seven parameters, concisely displaying all of the results can be challenging. Figure 4.7 attempts to do so. The REs are arranged horizontally within the parameter columns to show the effect of the stratum coarseness on RE. The larger characters to the left of each column represent the coarser stratum maps, with the left most values representing

the four-stratum solutions. The vertical arrangement of the plot elements represents the calculated RE values.

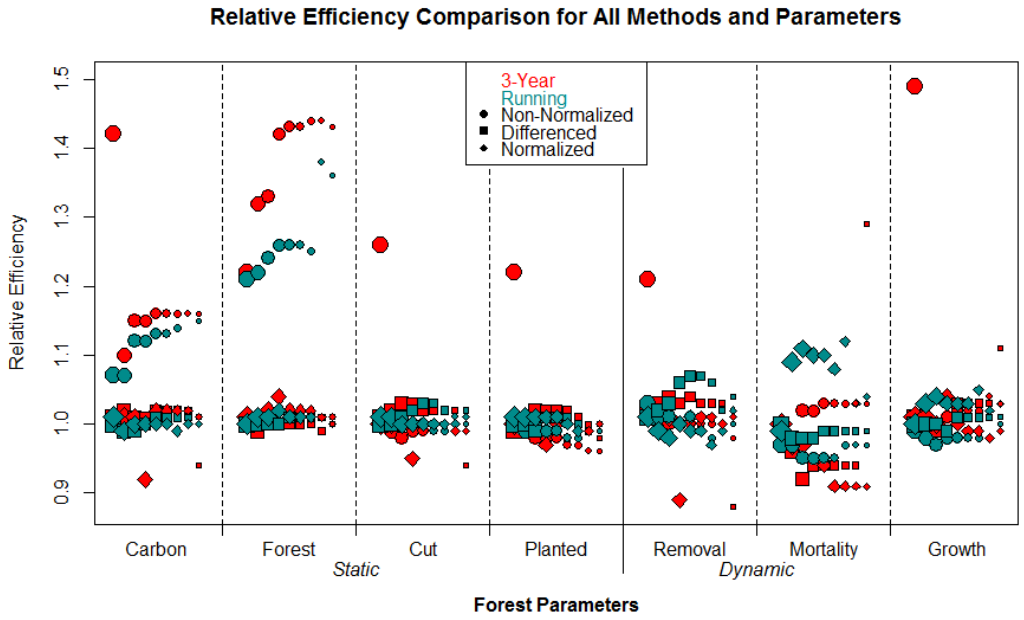


Figure 4.7. RE values for each of the forest parameters. The relative sizes of the plot elements describe the coarseness of the strata, with larger elements to the left of each column corresponding to treatments with fewer strata.

Some conclusions are immediately obvious from Figure 4.7. Firstly, the majority of the REs were quite low, with the vast majority of the REs being within 0.1 of 1. However, there are a number of cases where the RE is higher, on the order of 1.3 to 1.5. These high values do not compare favorably with REs for static variables from [1], which had REs for these parameters (from other regions of the USA) on the order of 2 or higher. Nevertheless, they do offer some sense of which approaches worked best here.

In the case of the Carbon and Forest Area parameters, we had consistently higher REs from the non-normalized transformations. This follows the intuition that for static variables related to the total amount of vegetation, the method which preserves the overall vegetation level should fare best. This is particularly true for the Forest Area parameter, since the method for determining Forest Area is calculating areal coverage, similar to the NDVI value of a Landsat pixel. In addition, the nonlinear nature of NDVI (as a ratio of a difference and a sum) may have caused the difference and normalization transformations to lead to inappropriate clustering. In

effect, this would introduce extra within-stratum variation and drive the corresponding RE down towards 1.

Furthermore, for these two static parameters, the 3-year group mean methods seem to outperform the running mean methods, within the context of the normalization approach. The most likely explanation for this was the possible misidentification of disturbance points, which the running mean dataset was more susceptible to.

Interestingly, for the dynamic parameters, the running means groups tend to dominate their 3-year counterparts, with the exception of the non-normalized data cases. This would conform to the intuition that the running means provide a more detailed regrowth profile for clustering. It is interesting that the 3-year group means are again paired with the non-normalized transformation for improved results.

It is telling that the REs for the dynamic variables generally fall very close to 1. There are exceptions to this rule, however. For Removal and Growth, there are instances of REs above 1.2, both in the cases of non-normalized 3-year groups again. Only Mortality seems to offer much to say in favor for the differenced and normalized transformations, a number of normalized REs (with running means) at or above 1.1 and a fine-strata differenced RE near 1.3. It is unclear why the REs for the Mortality parameter followed such a different pattern than the other variables. Nevertheless, the differences in the REs for Mortality (and for the other dynamic parameters) are generally very slight, less than 0.1 away from 1.

All told, there is a surprising lack of differentiation between the different methods in the dynamic parameters. This is suggestive that considerable variation was either not removed or actually reintroduced in the process of generating the strata maps.

4.4.2. Identifying method trends

In an effort to simplify the preprocessing for future exploration of regrowth-based stratifications, we compared the REs by the factors of mean type, normalization style, and the number of clusters. We first performed an exploratory graphical analysis, of which Figure 4.8 is one result. This figure shows the REs for the first parameter, Carbon. For reference, this is a revisualization of the first column from Figure 4.7, with more information about the number of clusters shown explicitly. The numeric values at each point are the sizes of the minimum strata for that solution, giving a sense of how reliable the estimate of the parameter, and thus the RE, is. Recall that a minimum sample size of 10 is recommended. [18]

As before, we may draw some simple conclusions immediately from the figure. As noted previously, the REs were close to 1 for differenced and normalized approaches, with the non-normalized approaches showing a slight improvement near 1.1. As the number of strata increases beyond 7, there is little additional gain in terms of the REs, suggesting that for our study area, 6 or 7 strata give us as much information as is useful.

The general effect of using finer strata appears to be a slight increase in the overall precision at the expense of confidence in the computed variances, due to fewer plots per stratum in the finer maps. There was one unusually high value (truncated here for easier discrimination of the other values), an RE of 1.42, for the non-normalized transformation on the 3-year group means with four strata. It is curious that the RE for the associated five-stratum map is so much lower at 1.10. It may suggest that the HCA division of one of the four strata into two was along some feature unrelated to the Carbon parameter.

In general, the number of sample units in the strata tended to fall below recommended levels at around 9 strata. This was particularly evident in the non-normalized methods, suggesting that a relatively small land cover feature may have been distinguished there which was differenced out in the other methods. Since the stratum sizes are mostly a function of the sample area, this issue may be easily overcome by choosing a larger area.

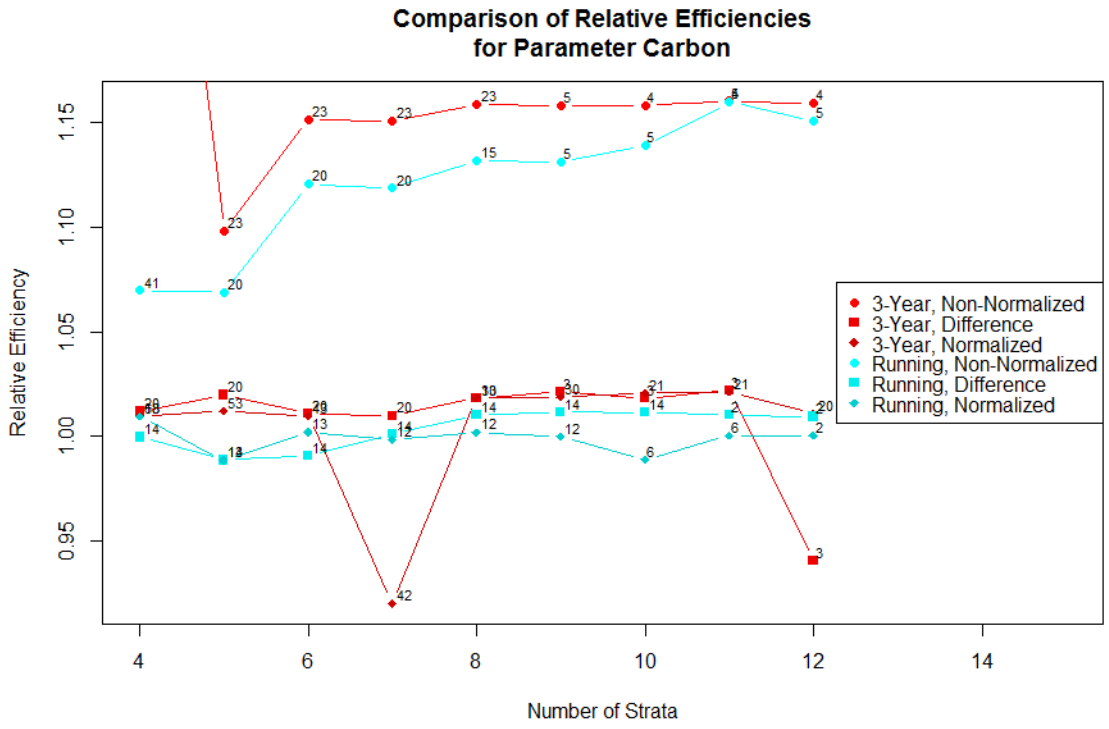


Figure 4.8. RE-by-stratum-number plot for Carbon, detailing possible effects of the different factors. Numeric values next to the points are the minimum number of plots in the smallest stratum for that solution. Note how the 3-year group outperforms the running means group in this case.

As noted before, it is interesting that the 3-year-group mean method for generating harmonic coefficients appears to be more effective than the running-means method. This is at first surprising because one would expect the running means to produce finer detail in the time series being used. However, the only part of that series actually used here was the 6-year segment after the sharpest measured drop in the time series. In effect, we measured the continued drop in vegetation for the first few years with the running means, whereas with the 3-year groups the sharpest descent was more accurate at defining the local minimum NDVI. This is the most likely cause for the relatively low running means scores for the static parameter value here.

We can loosely formalize the effects of each factor on the RE by means of a 3-way ANOVA. The residuals after fitting a no-interactions model roughly conformed to normality, which for the purposes of our heuristic analysis are good enough to take significant differences as meaningful. The ANOVA numerically verified the intuitions gleaned from Figure 4.7 and Figure 4.8; namely, that in the Carbon and Forest Area parameters the highest REs resulted from applying the 3-year group mean method and not normalizing the data. We got no other conclusive results from the ANOVA, also in keeping with the graphical conclusions.

4.5. Discussion

Given the myriad ways in which one can generate a stratified map, it is unsurprising that some approaches were fruitful and most were not. In particular, we did observe REs above 1.2 for each of the forest parameters, an improvement in precision that would have required a comparably precise simple random sample to be at least 20% larger. Generally, these came from strata maps created from non-normalized NDVI values from the 3-year group set of means. In particular, the highest RE observed in the study was from the 4-strata map of this type, in the dynamic forest parameter Growth. This is encouraging, suggesting that our approach of using post-disturbance information is appropriate.

It is even more informative to try and ascertain why the other approaches did not fare so well. Most of these unsatisfactory approaches resulted in REs that showed very little difference in effect from a simple random sample, and thus they showed little difference between each other, despite the major differences in the approaches on the front end. This suggests that our application did not allow for a fair comparison of the approaches. There are a number of possible contributing factors to this: 1) we may have identified the disturbances incorrectly, 2) we might have performed the cluster analysis inappropriately, 3) we might have chosen inappropriate cluster objects, or 4) we almost certainly chose a study area of insufficient size and variety in forest types. Any one of these four sources of error could have undermined our analysis and caused our strata maps to be an elaborately generated simple random sample from the perspective of the FIA plot data.

It is surprising for methods that specifically seek out vegetative regrowth profiles to have had a weak showing at improving the precision of dynamic parameter estimates. One potential cause for this result would be if we did not accurately mark the disturbances. In terms of the mean time series, the harmonic regression algorithm did filter out anomalous values in computing the means, but those anomalies should have been on the order of a few weeks and not the yearly timeframe we measured disturbances on. Thus, it is unlikely that harmonic regression would be a source of misidentification here. There is evidence that the approach of using the steepest descent to identify the disturbance may have provoked some misidentification. In the next attempt, we will try using the trough of the time series to mark the disturbance instead. If that offers little improvement in the resulting REs, we could try using other change detection approaches [14, 25, 32-33] to identify the time of disturbance instead.

In this paper, we used a training sample for the clustering process which did not account for whether the sample was in forested regions. This may have been problematic, since land cover features with vastly different regrowth patterns (e.g., agricultural fields on crop rotation) will force the HCA algorithm to accommodate them first. This would have the net effect of reducing the number of relevant clusters in any solution set, from a forest perspective. The most likely solution for this issue is to generate a forest mask based on various points throughout the timeframe, using (for training) only those regions which had forests on them at some point in the timeframe. This may affect the areal calculations for stratum weighting, but the same mask can be used to define a smaller area.

In the context of cluster objects, we ignored all measured disturbances in the last 6 years of the timeframe (from 2005-2011), since there was no additional data from which to construct a regrowth curve. If plot measurements were taken in those last 6 years, there would be considerable noise in the estimation process as a result. Because the FIA plots in the study area were measured on rotating schedules with 7-year intervals, it is quite possible that the changes measured in the FIA data did not appear in the clustering data. This could most easily be remedied by coordinating the FIA plot data to ensure that the measurements are temporally consistent.

In this light, it is also possible that there is a simpler alternative to identifying disturbances. In particular, if the years of measurement for the FIA plots were known, then we could simply look at the mean NDVI over that timeframe and either take those means as the clustering object directly or compute a simple summary of their behavior, such as linear slope from the first measurement period to the second. Doing so would not produce a proxy for site index, but it could be a better fit for improving FIA parameter estimate precision.

It is very likely that we used too small and homogenous a study area, based on the low plots-per-stratum values we observed in the finer stratum maps. The primary forest type in the study area was loblolly pine (*Pinus taeda*), and thus we may have been searching for variations which were too small to be easily distinguished. This would particularly be the case in light of using other land cover classes in the training data, at which point the useful signals from the FIA perspective are overcome by the sharper difference between forest-based and nonforest-based strata. While a finer strata map could overcome the forest-nonforest problem, it requires

considerably more FIA plots to ensure that each stratum contains the recommended 10 plots per stratum.

4.6. Conclusion

The results of this analysis showed some slight improvement in precision for dynamic parameter estimation by using post-disturbance based stratum assignments. We took a wide range of approaches to explore their potential, with no clear expectations of which approaches would be best. While the results of our approach in this paper were most likely hampered by possible missteps in the buildup to the analysis, we nevertheless did draw some useful conclusions to build on.

Firstly, we learned a couple of things about which methods were more effective. We saw that the running means approaches generally underperformed compared to the 3-year group approaches, despite the fact that the two approaches agreed on the common years. This suggests strongly that we misidentified the disturbances in the running means data and that the HCA algorithm was not able to adequately compensate for it. This in turn implies that a concavity-based approach will be more effective than a steepest-descent approach; this is easily accomplished. We also learned that the non-normalized transformations were more effective (or at least not *less* effective) than the others for static parameter estimates. This may be due to the nonlinear nature of NDVI calculations. It may also be due to the misidentification of disturbance points, an idea reinforced by the improved performance of the 3-year groups over the running means versions of the non-normalized transformations.

For the dynamic parameter estimates, the running means approaches showed a slight advantage over the 3-year groups when the transformations were normalized or differenced, despite the apparent misidentification of disturbance points mentioned previously. This suggests that the running means dataset is providing more detailed useful information to the HCA, from the standpoint of regrowth. It is possible that improved identification of disturbance points would help this class of approaches considerably.

More prospectively, we were able to clearly see some of the challenges to using this approach in action. This will guide us in the next effort to improve forest parameter estimate precision. Most critically, the next attempt should ensure that FIA plot timescales match the Landsat timescales, and it should incorporate a concavity-based method of identifying disturbances

instead of a slope-based one. It should also include special attention paid to training the clusters appropriately, namely, that the cluster solution is really discriminating between different forest conditions and not between broader land cover conditions. It may be worth applying these two corrections to the existing data as a test of concept before moving on. From there, the next attempt must include considerably larger areas, on the order of four to eight scenes of coverage, to ensure that a sufficient sample of FIA plots can be used for finer stratifications. This last requirement also reinforces the necessity of using easily calculated methods of cluster object generation.

In the end, we were able to demonstrate some improvement in the precision of dynamic forest parameter estimates. However, we think that we can improve considerably upon these results, and thus we view this work as a stepping stone for that improvement. We were able to rule out some methods of cluster object generation, and we have better, or perhaps more relevant, questions to ask of the data. We are confident that, upon answering these new questions, we can use Landsat to improve the precision of these dynamic parameter estimates yet more.

4.7. Acknowledgement

We also want to thank the USGS for making the Landsat archive freely available, enabling this study. We would also like to thank Karl Sorensen for his work in preprocessing the Landsat data through LEDAPS. This work was supported by the USDA Forest Service Cooperative Agreement with Virginia Tech (Grant No. 10-CA-11330145-158). It was also supported by the Landsat Science Team (USGS contract number G12PC00073), the Pine Integrated Network: Education, Mitigation, and Adaptation Project (PINEMAP, Coordinated Agricultural Project funded in 2011 by the USDA National Institute of Food and Agriculture), the McIntire-Stennis Cooperative Forestry Research program (USDA CSREES, Project No. VA-136614), and the Department of Forest Resources and Environmental Conservation at Virginia Tech.

4.8. References

[1] McRoberts, R. E., Holden, G. R., Nelson, M. D., Liknes, G. C., and Gormanson, D. D. (2005) "Using satellite imagery as ancillary data for increasing the precision of estimates for the Forest Inventory and Analysis program of the USDA Forest Service." *Canadian Journal of Forest Research,* 35(12), 2968-2980.

[2] Wynne, R.H., Oderwald, R. G., Reams, G.A., and Scrivani, J.A. (2000) “Optical remote sensing for forest area estimation.” *Journal of Forestry* 98(5):31-36.

[3] Hansen, M.H., and Wendt, D.G. (2000). “Using classified Landsat Thematic Mapper data for stratification in a statewide forest inventory”. *Proceedings of the First Annual Forest Inventory and Analysis Symposium* (November 1999), 20-27.

[4] McRoberts, R. E., Wendt, D. G., Nelson, M. D., and Hansen, M. H. (2002a) “Using a land cover classification based on satellite imagery to improve the precision of forest inventory area estimates.” *Remote Sensing of Environment,* 81(1), 36–44.

[5] Hoppus, M. L., and Lister, A. J. (2003) “A statistically valid method for using FIA plots to guide spectral class rejection in producing stratification maps.” *Proceedings of the Third Annual Forest Inventory and Analysis Symposium* (October 2001), 17–19.

[6] McRoberts, R. E., Nelson, M. D., and Wendt, D. G. (2002b) “Stratified estimation of forest area using satellite imagery, inventory data, and the k-Nearest Neighbors technique.” *Remote Sensing of Environment*, 8(2-3), 457-468.

[7] McRoberts, R. E., and Hansen, M. H. (1999) “Annual forest inventories for the North Central region of the United States.” *Journal of Agricultural, Biological, and Environmental Statistics*, 4(4), 361-371.

[8] McRoberts, R. E., Gobakken, T., and Naesset, E. (2012) “Post-stratified estimation of forest area and growing stock volume using lidar-based stratifications.” *Remote Sensing of Environment*, 125, 157-166.

[9] Hansen, M. C., Stehman, S. V., Potapov, P. V., Arunarwati, B., Stolle, F., and Pittman, K. (2009) “Quantifying changes in the rates of forest clearing in Indonesia from 1990 to 2005 using remotely sensed data sets.” *Environmental Research Letters*, 4(3).

[10] Nilsson, M., Holm, S., Reese, H., Wallerman, J., & Engberg, J. (2005). Improved forest statistics from the Swedish National Forest Inventory by combining field data and optical satellite data using post-stratification. *Proceedings of ForestSAT*, 31, 22-26.

[11] Fransson, J. E. S. (2000) “Estimation of forest parameters using CARABAS-II VHF SAR data.” *IEEE Transactions in Geosciences and Remote Sensing*, 38(2), 720-727.

[12] Scott, C. T., Bechtold, W. A., Reams, G. A., Smith, W. D., Westfall, J. A., Hansen, M. H., and Moisen, G. G. (2005) “Sample-based estimators used by the Forest Inventory and Analysis national information management system.” *The Enhanced Forest Inventory and*

Analysis Program — National Sampling Design and Estimation Procedures. U.S. Forest Service General Technical Report, SRS-80. pp. 43–67.

[13] Katila, M., and Tomppo, E. (2002) "Stratification by ancillary data in multisource forest inventories employing k-nearest-neighbour estimation." *Canadian Journal of Forest Research*, 32(9), 1548-1561.

[14] Huang, C., Goward, S., Masek, J., Thomas, N., Zhu, Z., and Vogelmann, J. (2010) "An automated approach for reconstructing recent forest disturbance history using dense Landsat time series stacks." *Remote Sensing of Environment*, 114(1), 183-198.

[15] Kennedy, R., Yang, Z., and Cohen, W. (2010) "Detecting trends in forest disturbance and recovery using yearly Landsat time series:1. LandTrendr — temporal segmentation algorithms." *Remote Sensing of Environment,* 114(12), 2897-2910.

[16] Cochran, W. G. (1977) *Sampling Techniques.* 3rd edition. Wiley, New York.

[17] Chojnacky, D. C. (1998) *Double sampling for stratification: a forest inventory application in the Interior West.* US Department of Agriculture, Forest Service, Rocky Mountain Research Station, 1998.

[18] Westfall, J. A., Patterson, P. L., and Coulston, J. W. (2011) "Post-stratified estimation: within-strata and total sample size recommendations." *Canadian Journal of Forest Research*, 41(5), 1130-1139.

[19] Burkman, B. (2005a) "Forest Inventory and Analysis sampling and plot design." *FIA Fact Sheet Series*. http://www.fia.fs.fed.us

[20] Burkman, B. (2005b) "Forest Inventory and Analysis data collection and analysis." *FIA Fact Sheet Series*. http://www.fia.fs.fed.us B

[21] USGS Global Visualization Viewer (GLOVIS) website. http://glovis.usgs.gov/

[22] USDA NRC Geospatial Data Gateway website. http://datagateway.nrcs.usda.gov/

[23] Masek, J. G., Vermote, E. F., Saleous, N. E., Wolfe, R., Hall, F. G., Huemmrich, K. F., Gao, F., Kutler, J., and Lim, T.-K. (2006) "A Landsat surface reflectance dataset for North America, 1990–2000." *IEEE Geoscience and Remote Sensing Letters*, 3(1), 68-72.

[24] Brooks, E. B., Thomas, V. A., Wynne, R. H., and Coulston, J. W. (2012) "Fitting the multitemporal curve: a Fourier series approach to the missing data problem in remote sensing analysis." *IEEE Transactions in Geosciences and Remote Sensing*, 50(9), 3340-3353.

[25] Brooks, E. B., Wynne, R. H., Thomas, V. A., Blinn, C. E., and Coulston, J. W. (2013) "Detecting forest disturbances with statistical quality control charts from Landsat data: An on-the-fly massively multitemporal change detection method." Submitted to *IEEE Transactions in Geosciences and Remote Sensing*.

[26] Ward, J. H. Jr., 1963. "Hierarchical grouping to optimize an objective function." *Journal of the American Statistical Association*, 58(301), 236-244.

[27] Tucker, C. J. (1979) "Red and photographic infrared linear combinations for monitoring vegetation." *Remote Sensing of Environment*, 8(2), 127-150.

[28] Bloomfield, P. (2004) *Fourier Analysis of Time Series: An Introduction*. Wiley-Interscience, 2nd ed. ISBN-10: 0471889482.

[29] Gupta, B. C. and Walker, F. W. Bloomfield, P. (2007) *Statistical Quality Control for the Six Sigma Green Belt*. American Society for Quality, Quality Press. ISBN: 978-0-87389-686-3.

[30] Montgomery, D. C. (2008) *Introduction to Statistical Quality Control*. 6th ed. Wiley. ISBN: 978-0470169926.

[31] Sampson, D. A., Wynne, R. H., and Seiler, J. (2008) "Edaphic and climatic effects on forest stand development, net primary production, and net ecosystem productivity simulated for Coastal Plain loblolly pine in Virginia." *Journal of Geophysical Research*, 113, G01003. 14p.

[32] Zhu, Z., Woodcock, C. E., and Olofsson, P. (2012) "Continuous monitoring of forest disturbance using all available Landsat imagery." *Remote Sensing of Environment*, Landsat Legacy Special Issue, 122, 75-91.

[33] Kennedy, R., Yang, Z., and Cohen, W. (2010) "Detecting trends in forest disturbance and recovery using yearly Landsat time series:1. LandTrendr — temporal segmentation algorithms." *Remote Sensing of Environment,* 114(12), 2897-2910.

Chapter 5: Conclusions

5.1. Summary

This work has shown both the power and promise of harmonic regression as applied to Landsat-based time series. It has established a working framework for using multiple years at a time to overcome the relatively long turnaround in Landsat images. It has also shown two very different and important applications of harmonic regression in the field of environmental monitoring, one in detecting subtle yet important changes to the landscape and one in improving the estimation of dynamic forest parameters. In both cases, the availability of rich data in the temporal dimension was leveraged to overcome limitations in the spatial dimensions.

More specifically, the work answers the three main questions outlined in Section 1.5.

5.1.1. Question 1

Is harmonic regression appropriate and desirable for imputing Landsat data that are temporally distributed in a scattered manner? Specifically, how do the fitting ability and robustness of harmonic regression compare to the alternative Landsat temporal imputation method of STAR-FM/ESTAR-FM? Chapter 2 of this work accomplished this, demonstrating that while there is a temporal region where the two approaches are comparable, harmonic regression is particularly well-suited to multi-year analyses due to both its flexibility in incorporating multiple years into the coefficient estimation and through the characterization of Fourier series through the storage of the coefficients. This latter property enables much simpler and cheaper storage of interpolated data than any fusion method can achieve.

5.1.2. Question 2

What is the utility of applying quality control charts to the residual time series from the harmonic regression? Can we use them to detect landscape disturbances on a wide range of severities, in an on-the-fly manner? Chapter 3 of this work accomplished this, demonstrating the generation of color-indexed disturbance maps. These maps did not only show how severe the disturbances were (including thinning as light as removing only 10% of the pixel area in tree crowns). They also showed when the disturbances occurred, within a few images of the event, and they showed a simple direct relationship between the severity of the signaled

disturbance with the interpretation on aerial images. Additionally, the maps also showed evidence of stand growth, both in terms of afforestation of harvested fields and in terms of maturation of existing stands. All of this was done using only Landsat image data, processed through harmonic regression and EWMA charts.

A particularly useful feature that came from this effort was the use of Shewhart X-bar charts to filter out pixel-specific clouds, shadows, and other short-lived anomalies. By filtering these anomalies out, the method generated more robust harmonic curves. By employing the Shewhart filtering, one can incorporate more of the original Landsat data, including partially clouded images, to extract as much useful information as possible from the data without concerns of cloud contamination.

5.1.3. Question 3

Can we improve the precision in post-stratified estimates of dynamic forest parameters from FIA plots by employing harmonic regression-based vegetation time series? An initial attempt at this was made in Chapter 4 of this work, using hierarchical clustering of post-disturbance time series to classify the plots according to a proxy for site index. While the application was hampered by a number of confounding factors, improvements in the precision of dynamic parameter estimates were indeed observed, in some cases reducing the variance of the estimates by a factor that would have required half again as many new FIA plots in a simple random sample. The chief confounding factors have been identified as a result of the analysis, and already another attempt is being executed with the goal of improving the precision by a factor of at least 2.

5.1.4. Overall Impact

Taken together, this work represents a significant step towards shifting the paradigm in multitemporal remote sensing at the Landsat level. Instead of taking only a few images for analysis, the work here shows a way to take large image stacks and treat them as statistical snapshots of a continuously changing landscape process. With such a model and framework in place, research questions begin to move from ex post facto analyses of singular events and instead move towards continuous monitoring and observing events in their entirety. In essence, the still frames of the past transition via a harmonic regression framework into a video of the future.

5.2. Future Work

In many ways, the applications shown in Chapters 3 and 4 of this work are only the beginning. Even from Chapter 2, there are a number of directions that further improvements to the harmonic regression method can take. A short and non-exhaustive list of future directions follows.

1) Develop a Harmonic Suitability Index (HSI) based solely on the temporal distribution of Landsat images to be used. Find the relationship between such an HSI and standard measures of model fit and robustness so that one may determine the suitability of harmonic regression at a glance.
2) Refine the X-bar chart-based cloud filter, developing an iterative algorithm that bootstraps working harmonic models from difficult data.
3) Expand on the idea of EWMA disturbance detection to incorporate retraining of pixels that have been disturbed. With such a protocol in place, begin real-time monitoring of multiple test regions to demonstrate and help fulfill the Landsat mission of monitoring of the Earth's resources. Use the results of the monitoring to provide early warning of regional changes due to climate or invasive species.
4) Improve the precision of dynamic forest parameter estimates yet more through refinements to the approach used in Chapter 4. With these improvements established, ramp the scale of estimation up to generate inputs for climate and ecosystem models, or use the approach as a model in its own right.
5) Develop a Landsat-based proxy for site index. Use this as a model input for regional productivity models. Additionally, use the site index proxy map to help determine the abundance of understory plants in a region (as a possible reverse of the indicator-plant approach) and thus improve Landsat-based Leaf Area Index (LAI) estimates, or use the map to forecast regions of potential plant migration under climate change.

These are only direct extensions of the work presented here. There are undoubtedly many new directions that one could take harmonic regression, control charts, or post-disturbance profiling. All of these directions will have the effect of utilizing the decadal and global coverage of Landsat data more fully, allowing for improved monitoring and modeling of the Earth's surface to meet the challenges facing us in the 21st century.

APPENDIX A. Data and Code for Chapter 2

List of scenes used:

Year	Month	Day	DOY
2001	1	13	13
2001	2	6	37
2001	3	10	69
2001	3	26	85
2001	4	19	109
2001	4	27	117
2001	5	5	125
2001	7	16	197
2001	8	25	237
2001	9	18	261
2001	9	26	269
2001	10	4	277
2001	10	20	293
2001	10	28	301
2001	11	5	309
2001	11	13	317
2001	12	15	349

Landsat images from additional years were chosen so that nominal cloudcover was <10%.

All available MODIS images were chosen.

R code excerpt (run on R 2.15.1 and 2.15.2)

```
##############################
# Generating STAR-FM ENVI rasters from STAR-FM outputs

####For the Daily Outputs

path="c:/Current Research/STARFM Leave-one-out Validation files/Area 1/Outputs 8-Day/"
list.files(path)
outpath="c:/Users/Evan/Desktop/R Outputs/"
```

```
list.files(outpath)

#Starts on 3685, ends on 9216
(9216-3684)/4/4

#Area code at spot 21
indices=as.integer(substring(list.files(path),21,21))
table(indices)

doys=as.integer(substring(list.files(path),25,27))
table(doys)

#Getting Area 1
library(caTools)
List=list.files(path)

Area=1
OutFile=file(paste(outpath,"STARFMPredictionArea1ValidationDays8Day","NDVI",sep=""), "wb")
Layers=17

if(Area==1) {Rows=472
        Cols=502}
if(Area==2) {Rows=354
        Cols=369}


for(File in 1:(1*17)){
  print(File)
  NIR=read.ENVI(paste(path,List[(File-1)*6+1],sep=""),headerfile=paste(path,List[(File-1)*6+2],sep=""))
  RED=read.ENVI(paste(path,List[(File-1)*6+4],sep=""),headerfile=paste(path,List[(File-1)*6+5],sep=""))
  NIR[NIR==(-9999)]=0
  RED[RED==(-9999)]=0
  #NIR=NIR-min(NIR)
```

```
  #RED=RED-min(RED)
  NDVI=floor((NIR-RED)/(NIR+RED+.00000001)*1000)
  NDVI[NDVI<0]=0

  writeBin(as.integer(c(t(NDVI))),OutFile, size=2)

}
close(OutFile)

intleave="bsq"
#Study Area 2
sink(paste(outpath,"STARFMPredictionArea1ValidationDays8Day","NDVI.hdr",sep=""))
cat("ENVI
description = { R-language data }
   samples = ",Cols,"
   lines = ",Rows,"
   bands = ",Layers,"
   data type = 2
   header offset = 0
   interleave =",intleave,"
   byte order = 0
   map info = {UTM, 1.000, 1.000, 601575, 3998835, 3.0000000000e+001, 3.0000000000e+001, 17,
North, WGS-84, units=Meters}
   ")
sink()


#Area 1
map info = {UTM, 1.000, 1.000, 601575, 3998835, 3.0000000000e+001, 3.0000000000e+001, 17, North,
WGS-84, units=Meters}
")
#Area 2
map info = {UTM, 1.000, 1.000, 665865, 3960165, 3.0000000000e+001, 3.0000000000e+001, 17, North,
WGS-84, units=Meters}
```

```
')


##########################################################

A=read.ENVI(paste(outpath,"STARFMPredictionArea2ValidationDays","NDVI",sep=""),headerfile=paste(o
utpath,"STARFMPredictionArea2ValidationDays","NDVI.hdr",sep=""))
Rows=dim(A)[1]
Cols=dim(A)[2]

STARdate=rep(0,346)
for(File in 1:(1*346)){
  STARdate[File]=substring(List[(File-1)*6+1],25,27)
}
STARdate=as.numeric(STARdate)

xsam=sample(1:Rows,1)
ysam=sample(1:Cols,1)
Y=A[xsam,ysam,1:17]
plot(LSDates[Y>0],Y[Y>0] ,"b",ylim=c(0,1000))



##############################################################################################
##############################################################################################
####For the 8-day composites

path="C:/Current Research/Everything Needed for STAR-FM 8-Day/STARFM Outputs for MODIS
Composite Images/"
list.files(path)
outpath="C:/Current Research/STARFM 8-Day NDVI/"
list.files(outpath)

#Starts on 3685, ends on 9216
(9216-3684)/4/4
```

```
#Area code at spot 21
indices=as.integer(substring(list.files(path),21,21))
table(indices)

doys=as.integer(substring(list.files(path),25,27))
table(doys)

#Getting Area 1
library(caTools)
List=list.files(path)

for(File in 1:(2*43)){
  print(File)
  if(File<44){Area=1}
  if(File>43){Area=2}
  NIR=read.ENVI(paste(path,List[(File-1)*6+1],sep=""),headerfile=paste(path,List[(File-1)*6+2],sep=""))
  RED=read.ENVI(paste(path,List[(File-1)*6+4],sep=""),headerfile=paste(path,List[(File-1)*6+5],sep=""))
  NIR[NIR==(-9999)]=0
  RED[RED==(-9999)]=0
  #NIR=NIR-min(NIR)
  #RED=RED-min(RED)
  NDVI=floor((NIR-RED)/(NIR+RED+.00000001)*1000)
  NDVI[NDVI<0]=0

  Rows=dim(NDVI)[1]
  Cols=dim(NDVI)[2]
  Layers=1
  NDVI=t(NDVI)
  image(NDVI)

  OutFile=file(paste(outpath,substring(List[(File-1)*6+1],1,27),"NDVI",sep=""), "wb")
  writeBin(as.integer(c(NDVI)),OutFile, size=2)
```

```
  close(OutFile)
  intleave="bsq"

  #Different headers for the different study areas
  if(Area==1){

    #Study Area 1
    sink(paste(outpath,substring(List[(File-1)*6+1],1,27),"NDVI.hdr",sep=""))
    cat("ENVI
description = { R-language data }
        samples = ",Cols,"
        lines = ",Rows,"
        bands = ",Layers,"
        data type = 2
        header offset = 0
        interleave =",intleave,"
        byte order = 0
        map info = {UTM, 1.000, 1.000, 601575, 3998835, 3.0000000000e+001, 3.0000000000e+001, 17,
North, WGS-84, units=Meters}
        ")
sink()
  }
  if(Area==2){
    #Study Area 2
    sink(paste(outpath,substring(List[(File-1)*6+1],1,27),"NDVI.hdr",sep=""))
    cat("ENVI
description = { R-language data }
        samples = ",Cols,"
        lines = ",Rows,"
        bands = ",Layers,"
        data type = 2
        header offset = 0
        interleave =",intleave,"
        byte order = 0
```

```
      map info = {UTM, 1.000, 1.000, 665865, 3960165, 3.0000000000e+001, 3.0000000000e+001, 17,
North, WGS-84, units=Meters}
      ")
sink()
 }
}

##############################################################################3
##############################################################################
OutFile=file(paste(outpath,"STARFMPredictionArea1AllDays","NDVI",sep=""), "wb")
Layers=43
Rows=502
Cols=472

for(File in 1:(1*43)){
 if(File<44){Area=1}
 NIR=read.ENVI(paste(path,List[(File-1)*6+1],sep=""),headerfile=paste(path,List[(File-1)*6+2],sep=""))
 RED=read.ENVI(paste(path,List[(File-1)*6+4],sep=""),headerfile=paste(path,List[(File-1)*6+5],sep=""))
 NIR[NIR==(-9999)]=0
 RED[RED==(-9999)]=0
 #NIR=NIR-min(NIR)
 #RED=RED-min(RED)
 NDVI=floor((NIR-RED)/(NIR+RED+.00000001)*1000)
 NDVI[NDVI<0]=0

 writeBin(as.integer(c(NDVI)),OutFile, size=2)

}
close(OutFile)

#Study Area 1
sink(paste(outpath,"STARFMPredictionArea1AllDays","NDVI.hdr",sep=""))
cat("ENVI
description = { R-language data }
```

```
    samples = ",Cols,"
    lines = ",Rows,"
    bands = ",Layers,"
    data type = 2
    header offset = 0
    interleave =",intleave,"
    byte order = 0
    map info = {UTM, 1.000, 1.000, 601575, 3998835, 3.0000000000e+001, 3.0000000000e+001, 17,
North, WGS-84, units=Meters}
    ")
sink()



##################################################
OutFile=file(paste(outpath,"STARFMPredictionArea2AllDays","NDVI",sep=""), "wb")
Area=2
Layers=43
Rows=369
Cols=354



for(File in 44:(2*43)){
  print(File)
  NIR=read.ENVI(paste(path,List[(File-1)*6+1],sep=""),headerfile=paste(path,List[(File-1)*6+2],sep=""))
  RED=read.ENVI(paste(path,List[(File-1)*6+4],sep=""),headerfile=paste(path,List[(File-1)*6+5],sep=""))
  NIR[NIR==(-9999)]=0
  RED[RED==(-9999)]=0
  #NIR=NIR-min(NIR)
  #RED=RED-min(RED)
  NDVI=floor((NIR-RED)/(NIR+RED+.00000001)*1000)
  NDVI[NDVI<0]=0



  writeBin(as.integer(c(NDVI)),OutFile, size=2)
```

```
}
close(OutFile)



#Study Area 2
sink(paste(outpath,"STARFMPredictionArea2AllDays","NDVI.hdr",sep=""))
cat("ENVI
description = { R-language data }
   samples = ",Cols,"
   lines = ",Rows,"
   bands = ",Layers,"
   data type = 5
   header offset = 0
   interleave =",intleave,"
   byte order = 0
   map info = {UTM, 1.000, 1.000, 665865, 3960165, 3.0000000000e+001, 3.0000000000e+001, 17,
North, WGS-84, units=Meters}
   ")
sink()

############################################################


A=read.ENVI(paste(path,"STARFMPredictionArea1AllDays","NDVI",sep=""),headerfile=paste(path,"STAR
FMPredictionArea1AllDays","NDVI.hdr",sep=""))
Rows=dim(A)[1]
Cols=dim(A)[2]


path="c:/Current Research/STARFM Daily NDVI/"
#path="c:/Documents and Settings/Evan/Desktop/"
list.files(path)
List=list.files(path)
STARdate=rep(0,346)
for(File in 1:(1*346)){
  STARdate[File]=substring(List[(File-1)*2+5],25,27)
```

```
}
STARdate=as.numeric(STARdate)
STARdate


xsam=sample(1:Rows,1)
ysam=sample(1:Cols,1)
Y=A[xsam,ysam,1:346]
plot(STARdate[A>0],A[A>0] ,"b")



##############################################
# Harmonic regression initial algorithm


#This program inputs an image and a table of dates, and it outputs images based on Fourier regression fitting of the data
library(caTools)


#Path setup
path="c:/Current Research/"
#path="c:/Documents and Settings/Evan/Desktop/"
list.files(path)
outpath="c:/Documents and Settings/Evan/Desktop/R Outputs/"
list.files(outpath)
#Input images
#Images in row-column format with cols 1,2 being x,y and subsequent columns being band values for the given dates


Area=1
d=read.csv(paste(path, "landsatstudyarea",Area,"nir.csv",sep=""), header=T)
d2=read.csv(paste(path, "landsatstudyarea",Area,"red.csv",sep=""), header=T)
head(d)
dim(d)
```

```
S=sample(1:dim(d)[1],100)
write.csv(d[S,1:2],paste(outpath,"Area 1 Sample Points.csv",sep=""),row.names=F)


#Raster information

Rows=dim(table(d[,2]))[1]
Cols=dim(table(d[,1]))[1]
Layers=dim(d)[2]-2
if(Area==1){
  xstart=601575
  ystart=3998835}
if(Area==2){
  xstart=665865
  ystart=3960165
}


#Converting to raster format
#For Export to ENVI, etc.  Automatically adjusts for X and y in cols 1 and 2
Raster=array(rep(0, Rows*Cols*Layers), dim=c(Rows, Cols, Layers))
#Raster=matrix(rep(0, Rows*Cols), nrow=Rows)
for(layer in 1:Layers){
  for(line in 1:Rows){
    Raster[line,,layer]=d[((line-1)*Cols+1):((line-1)*Cols+Cols),(layer+2)]
  }
}
NIR=Raster
#image(t(Raster[,,1]), col=rainbow(1000))

#For Export to ENVI, etc.  Automatically adjusts for X and y in cols 1 and 2
Raster=array(rep(0, Rows*Cols*Layers), dim=c(Rows, Cols, Layers))
#Raster=matrix(rep(0, Rows*Cols), nrow=Rows)
```

```
for(layer in 1:Layers){
  for(line in 1:Rows){
    Raster[line,,layer]=d2[((line-1)*Cols+1):((line-1)*Cols+Cols),(layer+2)]
  }
}
RED=Raster

#Dark object subtraction and NDVI computation
NDVI=array(rep(0, Rows*Cols*Layers), dim=c(Rows, Cols, Layers))
for(band in 1:Layers){
  NIR[,,band]=NIR[,,band]-min(NIR[,,band])
  RED[,,band]=RED[,,band]-min(RED[,,band])
  NDVI[,,band]=floor((NIR[,,band]-RED[,,band])/(NIR[,,band]+RED[,,band]
                                    +.00000001)*1000)
}
NDVI[NDVI<0]=0
image(NDVI[,,1])



#Making Blank Coefficient "Rasters"
C0=matrix(rep(0, Rows*Cols), nrow=Rows)
CS1=matrix(rep(0, Rows*Cols), nrow=Rows)
CC1=matrix(rep(0, Rows*Cols), nrow=Rows)
CS2=matrix(rep(0, Rows*Cols), nrow=Rows)
CC2=matrix(rep(0, Rows*Cols), nrow=Rows)
CS3=matrix(rep(0, Rows*Cols), nrow=Rows)
CC3=matrix(rep(0, Rows*Cols), nrow=Rows)
CS4=matrix(rep(0, Rows*Cols), nrow=Rows)
CC4=matrix(rep(0, Rows*Cols), nrow=Rows)
OLS=matrix(rep(0, Rows*Cols), nrow=Rows)
MSE=matrix(rep(0, Rows*Cols), nrow=Rows)
```

```
bintots=function(int){
  bits=0
  ts=c()
  while(int>0){
    ts=c(ts,int%%2)
    int=(int-int%%2)/2
    bits=bits+1}
  (ts*c(1:bits))[ts>0]}


#Input dates
x=c(13, 37, 69, 85, 109, 117,125,197,237,261,269,277,293,301,309,317,349)
x=x*2*pi/365

Raster=NDVI
for(i in 1:Rows){
  for(j in 1:Cols){
    print(i)
    print(j)
    Dth=100
    Sth=2500
    Upth=3000
    A=OLremover(x,Raster[i,j,], Dth,Sth,Upth)
    #Th=225
    #A=OLremover(x,Raster[i,j,], Th)
    OLS[i,j]=sum(2^(A[,3]-1))
    gap=32
    #A[(A[-1,1]-A[-dim(A)[1],1])>(2*pi*32/365),1]
    bigD=c(2:dim(A)[1])[(A[-1,1]-A[-dim(A)[1],1])>(2*pi*(gap+1)/365)]
    xm=(A[(bigD-1),1]+A[bigD,1])/2
    ydiff=(A[(bigD),2]-A[(bigD-1),2])
    xdiff=(A[(bigD),1]-A[(bigD-1),1])
```

```
    dervs=(ydiff)/xdiff
    ym=A[(bigD-1),2]+dervs*(xdiff)/2
    B=cbind(c(A[,1],xm),c(A[,2],ym))
    B=B[order(B[,1]),]
    bigD=c(2:dim(B)[1])[(B[-1,1]-B[-dim(B)[1],1])>(2*pi*(gap+1)/365)]
    xm=(B[(bigD-1),1]+B[bigD,1])/2
    ydiff=(B[(bigD),2]-B[(bigD-1),2])
    xdiff=(B[(bigD),1]-B[(bigD-1),1])
    dervs=(ydiff)/xdiff
    ym=B[(bigD-1),2]+dervs*(xdiff)/2
    B=cbind(c(B[,1],xm),c(B[,2],ym))
    B=B[order(B[,1]),]

X0=cbind(rep(1,length(B[,1])),sin(B[,1]),cos(B[,1]),sin(2*B[,1]),cos(2*B[,1]),sin(3*B[,1]),cos(3*B[,1]),sin(4*B
[,1]),cos(4*B[,1]))
    if(abs(det(t(X0)%*%X0))>.00001){
      D=(c(solve(t(X0)%*%X0)%*%t(X0)%*%B[,2]))
    }
    if(abs(det(t(X0)%*%X0))<=.00001){
      D=(c(solve(t(X0[,1])%*%X0[,1])%*%t(X0[,1])%*%B[,2],rep(0,8)))
    }

    plot(x,Raster[i,j,],"b")
    points(A[,1],A[,2],pch=20,col="blue","b")
    points(B[,1],B[,2],pch=20,col="red",cex=.75)


    MSE[i,j]=round(sqrt((t(B[,2]-X0%*%D)%*%(B[,2]-X0%*%D))/(length(B[,2])-length(D))),2)*100


    C0[i,j]=D[1]
    CS1[i,j]=D[2]
    CC1[i,j]=D[3]
    CS2[i,j]=D[4]
```

```
    CC2[i,j]=D[5]
    CS3[i,j]=D[6]
    CC3[i,j]=D[7]
    CS4[i,j]=D[8]
    CC4[i,j]=D[9]
  }
}

intleave="bsq"
OutFile=file(paste(outpath,"FourierFitPredictionsArea",Area,sep=""), "wb")
for(doy in 1:365){
  Test=floor(C0+CS1*sin(doy*2*pi/365)+CS2*sin(2*doy*2*pi/365)+CS3*sin(3*doy*2*pi/365)

+CC1*cos(doy*2*pi/365)+CC2*cos(2*doy*2*pi/365)+CC3*cos(3*doy*2*pi/365)+CS4*sin(4*doy*2*pi/365)+
CC4*cos(4*doy*2*pi/365))
  #image(Test)
  print(doy)
  writeBin(as.integer(c(t(Test))),OutFile, size=2)
}
close(OutFile)

sink(paste(outpath,"FourierFitPredictionsArea",Area,".hdr",sep=""))
cat("ENVI
description = { R-language data }
samples = ",Cols,"
lines = ",Rows,"
bands = ",365,"
data type = 2
header offset = 0
interleave =",intleave,"
byte order = 0
map info = {UTM, 1.000, 1.000,",xstart,",",ystart,", 3.0000000000e+001, 3.0000000000e+001, 17, North,
WGS-84, units=Meters}
")
```

```
sink()

##########################################
# Comparing R^2 and predicted R^2 across study areas

#Full Scene PRESS

path="c:/Current Research/"
list.files(path)
library(caTools)

LSDates=c(13, 37, 69, 85, 109, 117,125,197,237,261,269,277,293,301,309,317,349)

ED1=read.ENVI(paste(path,"STARFMPredictionArea1ValidationDays8DayNDVI",sep=""),headerfile=paste(path,"STARFMPredictionArea1ValidationDays8DayNDVI.hdr",sep=""))
ED2=read.ENVI(paste(path,"STARFMPredictionArea2ValidationDays8DayNDVI",sep=""),headerfile=paste(path,"STARFMPredictionArea2ValidationDays8DayNDVI.hdr",sep=""))
D1=read.ENVI(paste(path,"STARFMPredictionArea1ValidationDaysDailyNDVI",sep=""),headerfile=paste(path,"STARFMPredictionArea1ValidationDaysDailyNDVI.hdr",sep=""))
D2=read.ENVI(paste(path,"STARFMPredictionArea2ValidationDaysDailyNDVI",sep=""),headerfile=paste(path,"STARFMPredictionArea2ValidationDaysDailyNDVI.hdr",sep=""))
LS1=read.ENVI(paste(path,"LSKnownNDVIArea1",sep=""))
LS2=read.ENVI(paste(path,"LSKnownNDVIArea2",sep=""))
FP1=read.ENVI(paste(path,"Fourier Deleted Predictions Area1",sep=""))
FP2=read.ENVI(paste(path,"Fourier Deleted Predictions Area2",sep=""))
Key1=read.ENVI(paste(path,"NLCDArea1",sep=""))
Key2=read.ENVI(paste(path,"NLCDArea2",sep=""))

table(c(Key1,Key2))/sum(table(c(Key1,Key2)))*100
pie(table(c(Key1,Key2)),labels=names(Key),col=rainbow(length(Key)))

Key=as.numeric(names(table(c(Key1,Key2))))
#as.numeric(names(table(data[,6])))
```

```
names(Key)=c("Open Water","Developed, Open Space","Developed, Low Intensity","Developed, Medium Intensity","Developed, High Intensity","Barren Land","Deciduous Forest","Evergreen Forest","Mixed Forest", "Shrub/Scrub","Grassland/Herbaceous","Pasture Hay","Cultivated Crops","Woody Wetlands","Emergent Herbaceous Wetlands")

Key



EDR1=ED1-LS1

EDR2=ED2-LS2

DR1=D1-LS1

DR2=D2-LS2

FR1=FP1-LS1

FR2=FP2-LS2


library(abind)

LSMeans1=apply(LS1,1:2,mean)

LSDevs1=LS1

for(i in 1:17){

  LSDevs1[,,i]=LS1[,,i]-LSMeans1

}

SOS=function(x){sum(x^2)}

LSSST1=apply(LSDevs1,1:2,SOS)

FPRS1=round((1-(FPRESS1/LSSST1))*100,1)

EDPRS1=round((1-(EDPRESS1/LSSST1))*100,1)

DPRS1=round((1-(DPRESS1/LSSST1))*100,1)

FPRS1[FPRS1<0]=0

EDPRS1[EDPRS1<0]=0

DPRS1[DPRS1<0]=0



LSMeans2=apply(LS2,1:2,mean)

LSDevs2=LS2

for(i in 1:17){

  LSDevs2[,,i]=LS2[,,i]-LSMeans2
```

```
}
SOS=function(x){sum(x^2)}
LSSST2=apply(LS2,1:2,SOS)
FPRS2=round((1-(FPRESS2/LSSST2))*100,1)
EDPRS2=round((1-(EDPRESS2/LSSST2))*100,1)
DPRS2=round((1-(DPRESS2/LSSST2))*100,1)
FPRS2[FPRS2<0]=0
EDPRS2[EDPRS2<0]=0
DPRS2[DPRS2<0]=0


boxplot(c(FPRS1),c(EDPRS1),c(DPRS1))
boxplot(c(FPRS2),c(EDPRS2),c(DPRS2))
cor(cbind(c(FPRS1),c(EDPRS1),c(DPRS1)))
plot(c(FPRS1),c(EDPRS1),pch=19)
lines(0:100,0:100,col="red",lw=3,lty=2)
plot(c(FPRS1),c(DPRS1),pch=19)
lines(0:100,0:100,col="red",lw=3,lty=2)

round(table(c(FPRS1))/sum(table(ranks))*100,1)
T=t(table(c(FPRS1),c(Key1)))
rownames(T)=names(Key1)
apply(T,1,median)

FPRESS1=FPRS1
EDPRESS1=EDPRS1
DPRESS1=DPRS1

plot(0,0,xlim=c(0,8),col="white",ylim=c(0, 100),axes=F,ann=F)
axis(2,c(1:10)*100)
#axis(2,(c(0:floor(max(RMSES)/100))+1)*100)
#axis(1,at=c(1,2,4,5,7,8),c("8-Day 1","8 Day 2","Daily 1","Daily 2","Fourier 1","Fourier
2"))#colnames(PRESSES))
axis(1,at=c(1,2,3,5,6,7),c("8-Day","Fourier","Daily","8 Day","Fourier","Daily"))#colnames(PRESSES))
```

```
axis(1,at=c(2,6),c("Greensboro","Pittsboro-Seaforth"),pos=-150,lty=0)
title(main=paste("Comparison of Predictive Errors in Algorithms
Violin Plots of Predictive RMSE",sep=""),ylab="Predictive RMSE")
box()
vioplot(c(na.exclude(EDPRS1)),add=T,col=c(2),at=1)
vioplot(c(na.exclude(FPRS1)),add=T,col=c(4),at=2)
vioplot(c(na.exclude(DPRS1)),add=T,col=c(3),at=3)
vioplot(c(na.exclude(EDPRS2)),add=T,col=c(2),at=5)
vioplot(c(na.exclude(FPRS2)),add=T,col=c(4),at=6)
vioplot(c(na.exclude(DPRS2)),add=T,col=c(3),at=7)


summary(FPRS2[Key2==41])
summary(EDPRS2[Key2==41])
summary(DPRS2[Key2==41])
plot(FPRS2[Key2==41],EDPRS2[Key2==41],pch=19)
lines(0:100,0:100,col="red",lw=3,lty=2)
summary(c(FPRS2[Key2==42]-EDPRS2[Key2==42]))


plot(c(FPRS1),c(EDPRS1),c(DPRS1))




EDR1x=EDR1[,,2:16]
EDR2x=EDR2[,,2:16]
DR1x=DR1[,,2:16]
DR2x=DR2[,,2:16]
FR1x=FR1[,,2:16]
FR2x=FR2[,,2:16]
```

```
EDR1x=EDR1[,,-c(2)]
EDR2x=EDR2[,,-c(2)]
DR1x=DR1[,,-c(2)]
DR2x=DR2[,,-c(2)]
FR1x=FR1[,,-c(2)]
FR2x=FR2[,,-c(2)]



Rows=dim(ED1)[1]
Cols=dim(ED1)[2]
EDPRESS1=matrix(rep(0,Rows*Cols),ncol=Cols)
for(i in 1:Rows){
  for(j in 1:Cols){
    EDPRESS1[i,j]=sum(EDR1[i,j,]^2)
  }}

Rows=dim(ED2)[1]
Cols=dim(ED2)[2]
EDPRESS2=matrix(rep(0,Rows*Cols),ncol=Cols)
for(i in 1:Rows){
  for(j in 1:Cols){
    EDPRESS2[i,j]=sum(EDR2[i,j,]^2)
  }}

Rows=dim(D1)[1]
Cols=dim(D1)[2]
DPRESS1=matrix(rep(0,Rows*Cols),ncol=Cols)
for(i in 1:Rows){
  for(j in 1:Cols){
    DPRESS1[i,j]=sum(DR1[i,j,]^2)
```

```
  }}

Rows=dim(D2)[1]
Cols=dim(D2)[2]
DPRESS2=matrix(rep(0,Rows*Cols),ncol=Cols)
for(i in 1:Rows){
  for(j in 1:Cols){
    DPRESS2[i,j]=sum(DR2[i,j,]^2)
  }}

Rows=dim(FP1)[1]
Cols=dim(FP1)[2]
FPRESS1=matrix(rep(0,Rows*Cols),ncol=Cols)
for(i in 1:Rows){
  for(j in 1:Cols){
    FPRESS1[i,j]=sum(FR1[i,j,]^2)
  }}

Rows=dim(FP2)[1]
Cols=dim(FP2)[2]
FPRESS2=matrix(rep(0,Rows*Cols),ncol=Cols)
for(i in 1:Rows){
  for(j in 1:Cols){
    FPRESS2[i,j]=sum(FR2[i,j,]^2)
  }}



plot(FPRESS2,EDPRESS2,pch=20,cex=.3)
lines(c(0,10000000),c(0,10000000),col="black",lw=2)
for(i in 1:length(Key)){
  classval=Key[i]

points(FPRESS2[Key2==classval],EDPRESS2[Key2==classval],cex=.3,col=rainbow(length(Key))[i],pch=2
0)
```

```
}


sum(FPRESS2<EDPRESS2)/sum(FPRESS2>EDPRESS2|FPRESS2<EDPRESS2)

library(vioplot)
vioplot(c(sqrt(EDPRESS1/(17-9-1))))

plot(0,0,xlim=c(0,6),col="white",ylim=c(0, 1000),axes=F,ann=F)
axis(2,c(1:10)*100)
#axis(2,(c(0:floor(max(RMSES)/100))+1)*100)
axis(1,at=c(1,3,5),c("8-Day","Daily","Fourier"))#colnames(PRESSES))
title(main=paste("Comparison of Predictive Errors in Algorithms
Endpoints Included",sep=""),ylab="Predictive RMSE",xlab="Algorithm")
box()
vioplot(c(sqrt(EDPRESS1/(17-9-1))),add=T,col=c(2),at=1)
vioplot(c(sqrt(DPRESS1/(17-9-1))),add=T,col=c(3),at=3)
vioplot(c(sqrt(FPRESS1/(17-9-1))),add=T,col=c(4),at=5)


compraster=LS2[,,1]
compraster[FPRESS2<EDPRESS2]=1
compraster[FPRESS2>=EDPRESS2]=0
compraster=sqrt(FPRESS2/7)-sqrt(EDPRESS2/7)
boxplot(compraster~Key2)
lines(c(-1,100),c(0,0),col="red",lw=2,lty=2)


ranks=rep(0,dim(LS1)[1]*dim(LS1)[2])
PRESSES=cbind(c(FPRESS1),c(DPRESS1),c(EDPRESS1))
colnames(PRESSES)=c("Fourier Regression","Daily STARFM","8-Day STARFM")
for(runs in 1:(dim(LS1)[1]*dim(LS1)[2])){
  #print(runs)
```

```
  if(min(rank(PRESSES[runs,]))>1){
    ranks[runs]="Tie"
  }
  if(min(rank(PRESSES[runs,]))==1){
    ranks[runs]=colnames(PRESSES)[rank(PRESSES[runs,])==1]
  }
}
table(ranks)
colSums(PRESSES)/min(colSums(PRESSES))
round(table(ranks)/sum(table(ranks))*100,1)
T=t(table(ranks,c(Key1)))
rownames(T)=names(Key)
T


PRESSES2=PRESSES

PRESSES=PRESSES[,-2]

image(t(compraster))
write.ENVI(t(compraster),paste(path,"Comparing Press Area 2",sep=""))

compraster=LS1[,,1]
compraster[FPRESS1<EDPRESS1]=1
compraster[FPRESS1>=EDPRESS1]=0
compraster=sqrt(FPRESS2/7)#-sqrt(EDPRESS1/7)
boxplot(compraster~Key2)
lines(c(-1,100),c(0,0),col="red",lw=2,lty=2)
image(t(compraster),col=rainbow(10))
hist(c(compraster))
write.ENVI(t(compraster),paste(path,"Comparing Press Area 1 Y or N",sep=""))

MinRaster1=FPRESS1
```

```
MinRaster1[EDPRESS1<DPRESS1&EDPRESS1<FPRESS1]=1
MinRaster1[DPRESS1<FPRESS1&DPRESS1<EDPRESS1]=2
MinRaster1[FPRESS1<DPRESS1&FPRESS1<EDPRESS1]=3
table(c(MinRaster1))
image(MinRaster1)
write.ENVI(t(MinRaster1),paste(path,"Minimum PRESS Raster Area 1",sep=""))

MinRaster2=FPRESS2
MinRaster2[EDPRESS2<DPRESS2&EDPRESS2<FPRESS2]=1
MinRaster2[DPRESS2<FPRESS2&DPRESS2<EDPRESS2]=2
MinRaster2[FPRESS2<DPRESS2&FPRESS2<EDPRESS2]=3
table(c(MinRaster2))
image(MinRaster2)
write.ENVI(t(MinRaster2),paste(path,"Minimum PRESS Raster Area 2",sep=""))




ranks=rep(0,dim(LS1)[1]*dim(LS1)[2])
PRESSES=cbind(c(FPRESS1),c(DPRESS1),c(EDPRESS1))
colnames(PRESSES)=c("Fourier Regression","Daily STARFM","8-Day STARFM")
for(runs in 1:(dim(LS1)[1]*dim(LS1)[2])){
  print(runs/(dim(LS1)[1]*dim(LS1)[2])*100)
  if(min(rank(PRESSES[runs,]))>1){
    ranks[runs]="Tie"
  }
  if(min(rank(PRESSES[runs,]))==1){
    ranks[runs]=colnames(PRESSES)[rank(PRESSES[runs,])==1]
  }
}
table(ranks)
colSums(PRESSES)/min(colSums(PRESSES))
round(table(ranks)/sum(table(ranks))*100,1)
```

```
T=t(table(ranks,c(Key1)))
rownames(T)=names(Key1)
T


PRESSES=PRESSES[,-3]




pdf(paste(path,"PRESS Stats by NLCD Class and Area.pdf",sep=""),width=9)
for(i in 1:length(Key)){
  classval=Key[i]
  if(length(EDPRESS1[Key1==classval])>1){
    plot(0,0,xlim=c(0,8),col="white",ylim=c(0, 1000),axes=F,ann=F)
    axis(2,c(1:10)*100)
    #axis(2,(c(0:floor(max(RMSES)/100))+1)*100)
    axis(1,at=c(1,2,3,5,6,7),c("8-Day","Fourier","Daily","8 Day","Fourier","Daily"))#colnames(PRESSES))
    axis(1,at=c(2,6),c("Area 1","Area 2"),pos=-150,lty=0)
    #axis(1,at=c(1,2,3,5,6,7),c("8-Day 1","8 Day 2","Daily 1","Daily 2","Fourier 1","Fourier
2"))#colnames(PRESSES))
    title(main=paste("Comparison of Predictive Errors in Algorithms
NLCD 2006 Classification: ",names(Key[Key==classval]),"
                Endpoints Excluded",sep=""),ylab="Predictive RMSE")#,xlab="Algorithm")
box()
    #classval=Key[1]
    vioplot(c(sqrt(EDPRESS1[Key1==classval]/(17-9-1))),add=T,col=c(2),at=1)
    vioplot(c(sqrt(FPRESS1[Key1==classval]/(17-9-1))),add=T,col=c(4),at=2)
    vioplot(c(sqrt(DPRESS1[Key1==classval]/(17-9-1))),add=T,col=c(3),at=3)
    vioplot(c(sqrt(EDPRESS2[Key2==classval]/(17-9-1))),add=T,col=c(2),at=5)
    vioplot(c(sqrt(FPRESS2[Key2==classval]/(17-9-1))),add=T,col=c(4),at=6)
    vioplot(c(sqrt(DPRESS2[Key2==classval]/(17-9-1))),add=T,col=c(3),at=7)
  }
}
dev.off()
```

```
Rows=dim(ED1)[1]
Cols=dim(ED1)[2]
EDPRESS1=matrix(rep(0,Rows*Cols),ncol=Cols)
for(i in 1:Rows){
  for(j in 1:Cols){
    EDPRESS1[i,j]=sum(EDR1x[i,j,]^2)
  }}

Rows=dim(ED2)[1]
Cols=dim(ED2)[2]
EDPRESS2=matrix(rep(0,Rows*Cols),ncol=Cols)
for(i in 1:Rows){
  for(j in 1:Cols){
    EDPRESS2[i,j]=sum(EDR2x[i,j,]^2)
  }}

Rows=dim(D1)[1]
Cols=dim(D1)[2]
DPRESS1=matrix(rep(0,Rows*Cols),ncol=Cols)
for(i in 1:Rows){
  for(j in 1:Cols){
    DPRESS1[i,j]=sum(DR1x[i,j,]^2)
  }}

Rows=dim(D2)[1]
Cols=dim(D2)[2]
DPRESS2=matrix(rep(0,Rows*Cols),ncol=Cols)
for(i in 1:Rows){
  for(j in 1:Cols){
```

```
    DPRESS2[i,j]=sum(DR2x[i,j,]^2)
  }}

Rows=dim(FP1)[1]
Cols=dim(FP1)[2]
FPRESS1=matrix(rep(0,Rows*Cols),ncol=Cols)
for(i in 1:Rows){
  for(j in 1:Cols){
    FPRESS1[i,j]=sum(FR1x[i,j,]^2)
  }}

Rows=dim(FP2)[1]
Cols=dim(FP2)[2]
FPRESS2=matrix(rep(0,Rows*Cols),ncol=Cols)
for(i in 1:Rows){
  for(j in 1:Cols){
    FPRESS2[i,j]=sum(FR2x[i,j,]^2)
  }}



plot(FPRESS1,EDPRESS1,pch=20,cex=.3)
lines(c(0,10000000),c(0,10000000),col="red")
sum(FPRESS1<EDPRESS1)/sum(FPRESS1>EDPRESS1|FPRESS1<EDPRESS1)

library(vioplot)
vioplot(c(sqrt(EDPRESS1/(17-9-1))))




plot(0,0,xlim=c(0,8),col="white",ylim=c(0, 1000),axes=F,ann=F)
axis(2,c(1:10)*100)
#axis(2,(c(0:floor(max(RMSES)/100))+1)*100)
#axis(1,at=c(1,2,4,5,7,8),c("8-Day 1","8 Day 2","Daily 1","Daily 2","Fourier 1","Fourier
2"))#colnames(PRESSES))
```

```
axis(1,at=c(1,2,3,5,6,7),c("8-Day","Fourier","Daily","8 Day","Fourier","Daily"))#colnames(PRESSES))
axis(1,at=c(2,6),c("Greensboro","Pittsboro-Seaforth"),pos=-150,lty=0)
title(main=paste("Comparison of Predictive Errors in Algorithms
Violin Plots of Predictive RMSE",sep=""),ylab="Predictive RMSE")
box()
vioplot(c(sqrt(EDPRESS1/(17-9-1))),add=T,col=c(2),at=1)
vioplot(c(sqrt(FPRESS1/(17-9-1))),add=T,col=c(4),at=2)
vioplot(c(sqrt(DPRESS1/(17-9-1))),add=T,col=c(3),at=3)
vioplot(c(sqrt(EDPRESS2/(17-9-1))),add=T,col=c(2),at=5)
vioplot(c(sqrt(FPRESS2/(17-9-1))),add=T,col=c(4),at=6)
vioplot(c(sqrt(DPRESS2/(17-9-1))),add=T,col=c(3),at=7)
#hist(c(sqrt(FPRESS2/(17-9-1)))-c(sqrt(EDPRESS2/(17-9-1))))
#summary(c(sqrt(FPRESS1/(17-9-1)))-c(sqrt(EDPRESS1/(17-9-1))))
```

```
########################Making the Fourier Deleted Predictions
library(abind)
test=abind(NDVIvals,ym,along=3)
test[1,1,]
testsort=test[,,order(c(Dates,xm))]
plot(c(Dates),NDVIvals[1,1,],"b")
```

```
points(c(Dates,xm),test[1,1,],"b",pch=20)
points(AllDates,Fills[1,1,],"b",col="red",pch=19)



#Suspect points are 2,5, 7, 8, 9, 11
LSDates=LSDates[-c(2, 5,7,8,9,11)]
LS1=LS1[,,-c(2,5,7,8,9,11)]


Delpreds=array(rep(0,dim(LS1)[1]*dim(LS1)[2]*length(LSDates)),dim=c(dim(LS1)[1],dim(LS1)[2],length(L
SDates)))


for(k in 1:length(LSDates)){
  print(c(k/length(LSDates))*100)


  gap=32
  Dates=LSDates[-k]*2*pi/365
  NDVIvals=LS2[,,(-k)]


  bigD=c(2:length(Dates))[(Dates[-1]-Dates[-length(Dates)])>(2*pi*(gap+1)/365)]
  xm=(Dates[(bigD-1)]+Dates[bigD])/2
  ydiff=NDVIvals[,,bigD]-NDVIvals[,,bigD-1]
  xdiff=Dates[bigD]-Dates[bigD-1]
  dervs=(ydiff)/xdiff
  ym=NDVIvals[,,bigD-1]+dervs*xdiff/2
  Fills=abind(NDVIvals,ym,along=3)
  AllDates=c(Dates,xm)
  Fills=Fills[,,order(AllDates)]
  AllDates=AllDates[order(AllDates)]
  bigD=c(2:length(AllDates))[(AllDates[-1]-AllDates[-length(AllDates)])>(2*pi*(gap+1)/365)]
  while(length(bigD)>0){
    xm=(AllDates[(bigD-1)]+AllDates[bigD])/2
    ydiff=(Fills[,,(bigD)]-Fills[,,(bigD-1)])
    xdiff=(AllDates[(bigD)]-AllDates[(bigD-1)])
    dervs=(ydiff)/xdiff
```

```
  ym=Fills[,,(bigD-1)]+dervs*(xdiff)/2
  Fills=abind(Fills,ym,along=3)
  AllDates=c(AllDates,xm)
  Fills=Fills[,,order(AllDates)]
  AllDates=AllDates[order(AllDates)]
  bigD=c(2:length(AllDates))[(AllDates[-1]-AllDates[-length(AllDates)])>(2*pi*(gap+1)/365)]
 }

 ns=4
 nc=4

X=cbind(rep(1,length(AllDates)),sin(t(matrix(rep(c(1:ns),length(AllDates)),ncol=length(AllDates)))*AllDates
),cos(t(matrix(rep(c(1:nc),length(AllDates)),ncol=length(AllDates)))*AllDates))
 Prefix=solve(t(X)%*%X)%*%t(X)

Xnew=cbind(rep(1,length(1)),sin(t(matrix(rep(c(1:ns),length(1)),ncol=length(1)))*LSDates[k]*2*pi/365),cos(
t(matrix(rep(c(1:nc),length(1)),ncol=length(1)))*LSDates[k]*2*pi/365))



Xnew=cbind(rep(1,length(1:365)),sin(t(matrix(rep(c(1:ns),length(1:365)),ncol=length(1:365)))*c(1:365)*2*p
i/365),cos(t(matrix(rep(c(1:nc),length(1:365)),ncol=length(1:365)))*c(1:365)*2*pi/365))
 Xnew%*%Prefix%*%Fills[150,210,]



 plot(0,0,xlim=c(0,365),ylim=c(0,1000),col="white",axes=FALSE,ann=F)
 title(xlab="Month in 2001", ylab="NDVI x 1000",
      main=paste("NDVI Time Series for 30m pixel",sep=""))
 #Classification:",names(Key)[Key==as.numeric(data[i,6])],sep="" ))
 Months=c(0,31,28,31,30,31,30,31,31,30,31,30,31)
 monthmids=(Months[-1]+Months[-13])/2
 axis(2,c(1:10)*100)
 axis(1,at=cumsum(Months),lab=F)
```

```
axis(1,at=cumsum(monthmids),c("JAN","FEB","MAR","APR","MAY","JUN","JUL","AUG","SEP","OCT","NOV","DEC"),tck=0)
  #box()
  for(mon in 1:11){
    lines(c(cumsum(Months)[mon+1],cumsum(Months)[mon+1]),c(0,1000),lty=3,lw=.5)
  }
  #lines(c(cumsum(Months)[11],cumsum(Months)[11]),c(0,725),lty=3,lw=.5)
  #lines(c(cumsum(Months)[12],cumsum(Months)[12]),c(0,725),lty=3,lw=.5)

  points(LSDates,LS2[150,210,],pch=19,col="gold3","b",cex=1.5,lw=2)
  lines(c(1:365),(Xnew%*%Prefix%*%Fills[150,210,]),lty=2,col="blue")
  points(LSDates[k],Xnew%*%Prefix%*%Fills[150,210,],pch=17,col="blue")

  #Coefs=array(rep(0,dim(Fills)[1]*dim(Fills)[2]*(1+ns+nc)),dim=c(dim(Fills)[1],dim(Fills)[2],(1+ns+nc)))
  #for(i in 1:dim(Fills)[1]){
  #for(j in 1:dim(Fills)[2]){
  #print(c(i/dim(Fills)[1],j/dim(Fills)[2])*100)
  #Coefs[i,j,]=
  #Prefix%*%Fills[i,j,]
  #}
  #}



  for(i in 1:dim(Fills)[1]){
    for(j in 1:dim(Fills)[2]){
      #print(c(i,j))
      #print(c(i/dim(Fills)[1],j/dim(Fills)[2])*100)
      Delpreds[i,j,k]=Xnew%*%Prefix%*%Fills[i,j,]
    }
  }

}
```

```
OutFile=file(paste(path,"Fourier Deleted Predictions Area1",sep=""), "wb")
writeBin(as.integer(round(c(aperm(Delpreds,c(2,1,3))),0)),OutFile, size=2)
close(OutFile)
intleave="bsq"


LSDates=c(13, 37, 69, 85, 109, 117,125,197,237,261,269,277,293,301,309,317,349)
LS1=read.ENVI(paste(path,"LSKnownNDVIArea1",sep=""))
plot(LSDates,LS1[234,152,],"b",pch=19)
points(LSDates,FP1[234,152,],"b",col="blue",pch=19)
points(LSDates[-c(2,5,7,8,9,11)],Delpreds[234,152,],"b",col="red",pch=19)


xsam=sample(1:dim(LS1)[1],1)
ysam=sample(1:dim(LS1)[2],1)
plot(LSDates,LS1[xsam,ysam,],"b",ylim=c(0,1000))
points(LSDates,ED1[xsam,ysam,],"b",col="red")
points(LSDates,D1[xsam,ysam,],"b",col="green")
points(LSDates,FP1[xsam,ysam,],"b",col="blue")
c(EDPRESS1[xsam,ysam],DPRESS1[xsam,ysam],FPRESS1[xsam,ysam])/min(c(EDPRESS1[xsam,ysa
m],DPRESS1[xsam,ysam],FPRESS1[xsam,ysam]))
```

APPENDIX B. Data and Code for Chapter 3

List of scenes used:

Year	Month	Day	DOY
2005	2	4	35
2005	3	8	67
2005	4	9	99
2005	5	27	147
2005	8	15	227
2005	10	18	291
2005	11	3	307
2006	2	7	38
2006	4	28	118
2006	6	15	166
2006	7	1	182
2006	7	17	198
2006	8	18	230
2006	9	3	246
2006	9	19	262
2006	10	5	278
2006	11	22	326
2006	12	8	342
2007	1	9	9
2007	1	25	25
2007	2	10	41
2007	2	26	57
2008	1	28	28
2008	4	17	108
2008	5	19	140
2008	7	22	204
2008	9	24	268
2008	10	26	300
2008	12	29	364
2009	1	14	14
2009	1	30	30
2009	3	3	62
2009	3	19	78
2009	6	23	174
2009	7	9	190
2009	7	25	206
2009	8	26	238

2009	9	27	270
2010	2	18	49
2010	3	6	65
2010	8	13	225
2010	9	14	257
2010	9	30	273
2010	10	16	289
2010	11	17	321
2010	12	3	337
2010	12	19	353
2011	3	25	84
2011	5	28	148
2011	6	13	164
2011	10	3	276

R code excerpt (run on R 2.15.1 and 2.15.2)

```
require(caTools)
require(abind)
require(plyr)
require(calibrate)

setwd(<WORKING DIRECTORY>)
path=<INPUT PATH>
path2=<OUTPUT PATH>

############################################################
#Creating Date Information File
setwd(<WORKING DIRECTORY>)
path=<INPUT PATH>
path2=<OUTPUT PATH>

L=list.files(pattern=glob2rx(<WILDCARD FILENAMES>))
```

```
DateInfo=c()
for(theFile in L){

DateInfo=rbind(DateInfo,c(as.numeric(substr(theFile,8,11)),as.numeric(substr(theFile,12,13)),as.numeric(
substr(theFile,14,15))))
}

DOYconv=function(v1){
  DiM=c(0,31,(28+(v1[1]%%4==0)),31,30,31,30,31,31,30,31,30,31)
  v1[3]+cumsum(DiM)[v1[2]]
}
DOYs=apply(DateInfo,1,DOYconv)

DateInfo=data.frame(cbind(DateInfo,DOYs))
names(DateInfo)=c("Year","Month","Day","DOY")

write.csv(DateInfo,paste(path2,"DateInfo1.csv",sep=""))



############################################################
# Conversion to post-DOS Tasseled Cap Angle Index

setwd(<WORKING DIRECTORY>)
path=<INPUT PATH>
path2=<OUTPUT PATH>

L=list.files(pattern=glob2rx(<WILDCARD FILENAMES>))

DateInfo=read.csv("DateInfo.csv",header=T)

## DOS, Tasseled Cap and Angle Index Transformations
## Integer output
DOSTCAI=function(spectrum){
```

```
  ## Transformation matrices are based on the book Remote Sensing: Models And Methods for Image
Processing By Robert A. Schowengerdt
  ## both TM and ETM+ matrices assume band combinations of 1,2,3,4,5,7. 6 is not included!
  C5=rbind(
    c(.2909, .2493, .4806,.5568, .4438, .1706),
    c(-.2728,-.2174,-.5508,.7221,.0733,-.1648),
    c(.1446,.1761,.3322,.3396,-.6210,-.4186),
    c(.8461,-.0731,-.4640,-.0032,-.0492,.0119),
    c(.0549,-.0232,.0339,-.1937,.4162,-.7823),
    c(.1186,-.8069,.4094,.0571,-.0228,.0220)    )
  B5=cbind(c(10.3695,-.7310,-3.3828,.7879,-2.4750,-.0336))
  TC=(C5%*%cbind(spectrum[1:6]-band.minima)+B5)[1:2]
  #     AIpix=
  as.integer(round(atan(TC[2]/TC[1])*1000))
  #     output=list(AIpix)
  #     names(output)=c("AIpix")
  #     output
  #   C7=rbind(c(.3561,.3972,.3904,.6966,.2286,.1596),
  #            c(-.3344,-.3544,-.4556,.6966,-.0242,-.2630),
  #            c(.2626,.2141,.0926,.0656,-.7629,-.5388),
  #            c(.0805,-.0498,-.1950,-.1327,.5752,-.7775),
  #            c(-.7252,-.0202,.6683,.0631,-.1494,-.0274),
  #            c(.4000,-.8172,.3832,.0602,-.1095,.0985)  )
  #   if(platform==5){result=C5%*%cbind(spectrum[1:6])+B5}
  #   if(platform==7){result=C7%*%cbind(spectrum[1:6])}
  #   result
}


AI.stack=array(0,dim=c(2000,2000,51))


for(index in 1:51){
theFile=L[index]
print(theFile)
raster=read.ENVI(paste(path,theFile,sep=""))
```

```
# raster=raster[2500:6500,4000:6500,]
raster[is.na(raster)]=0
sum(is.na(raster))
Rows=dim(raster)[1]
Cols=dim(raster)[2]
Bands=dim(raster)[3]

cat("Performing Dark Object Subtraction for ",theFile,"...\n",sep="")
epsiloner=function(layer){min(layer[layer>0])}
band.minima=aaply(raster,3,epsiloner)
raster.DOS=raster*0
  for(band in 1:Bands){raster.DOS[,,band]=raster[,,band]-band.minima[band]}


raster.AI=raster.DOS[,,1]*0

cat("Calculating Angle Index for ",theFile,"...\n",sep="")
for(i in 1:Rows){
  if(i%%200==0){cat(i/Rows*100,"%...\n",sep="")}
  for(j in 1:Cols){
raster.AI[i,j]=DOSTCAI(raster.DOS[i,j,])
  }
}

cat("Adding Angle Index to stack for ",theFile,"...\n",sep="")
AI.stack[,,index]=raster.AI

}


con = file(paste(path,theFile,".hdr",sep=""), "r")
A=readLines(con, 12 )
Map.Info=A[10]
```

```
close(con)


mode(AI.stack)="integer"
out.file.name="AIx1000 Stack"
write.ENVI(AI.stack, paste(path2,out.file.name,sep=""))
sink(paste(path2,out.file.name,".hdr",sep=""))
cat("ENVI
description = { R-language data }
samples = ",dim(AI.stack)[2],"
lines   = ",dim(AI.stack)[1],"
bands   = ",dim(AI.stack)[3],"
header offset = 0
data type = 3
interleave = bsq
byte order = 0
",Map.Info,"
band names = {
By Date,
}",sep="")
sink()


#################################
# Harmonic Regression Algorithm


setwd(<WORKING DIRECTORY>)
path=<INPUT PATH>
path2=<OUTPUT PATH>

L=list.files(pattern=glob2rx(<WILDCARD FILENAMES>))
```

```
DateInfo=read.csv("DateInfo.csv",header=T)


DOYs=DateInfo$DOY[(DateInfo$Year<2009)] # Keeping with the EWMA paper, I set this to look only at the dates from 2005-2008


raster=read.ENVI("AIx1000 Stack")[,,(DateInfo$Year<2009)] # Again, only dates from 2005-2008 were used for training in the paper

raster[is.na(raster)]=0 # ensuring there are no missing data

sum(is.na(raster))


con = file("AIx1000 Stack.hdr", "r") # reading the input header for information to be used in the output header

A=readLines(con, 12 )

Rows=as.numeric(substr(A[4],11,14)) # the number of lines in the ENVI file

Cols=as.numeric(substr(A[3],11,14)) # the number of samples in the ENVI file

Map.Info=A[10] # spatial registration information for use in writing the header file for the output

close(con)




timedat=DOYs # Preserving the DOYs object while using it for the analysis

screenpass=2 # number of standard deviations used as a threshold for filtering out anomalous values below.


timedat=timedat*2*pi/365 # Converting to [0,2pi]


ns=2 # number of sine terms in the model

nc=2 # number of cosine terms in the model

X=cbind(rep(1,length(timedat)),sin(t(matrix(rep(c(1:ns),length(timedat)),ncol=length(timedat)))*timedat),cos(t(matrix(rep(c(1:nc),length(timedat)),ncol=length(timedat)))*timedat))

# Design matrix X has a column for the constant term and columns for sines and cosines determined by ns and nc

Hat=X%*%solve(t(X)%*%X)%*%t(X) #Getting the full-history hat matrix for the initial pass


# Pixel-by-pixel coefficient calculation

pixelstuff=function(tseries){
```

```
  Preds1=as.numeric(Hat%*%tseries) # Regression values

  Resids1=tseries-Preds1 # Residuals


  std=sd(Resids1)


  screen1=(abs(Resids1)>(screenpass*std))+0 # Determining which dates had unusually large residuals.
The +0 forces the output to be numeric


  keeps=which(screen1==0) # Dummy index for values used in recalculating the coefficients


  solve(t(X[keeps,])%*%X[keeps,])%*%t(X[keeps,])%*%tseries[keeps] # Refined coefficient estimates
(Beta).  The matrix multiplcations are the big processing sink here.


}


AC=raster[,,c(1:(1+ns+nc))]*0 # It's faster to take an already initialized array and multiply by 0 than to
initialize a new one, at least for large arrays


# Doing it serially here, but this could be easily parallelized

for(i in 1:Rows){

  if(i%%50==0){cat(i/Rows*100,"%...\n",sep="")} # progress meter

  for(j in 1:Cols){

    if(sum(raster[i,j,])>0){ # Ignoring "empty" pixels, as the coefficients for these would be 0 anyway

      AC[i,j,]=pixelstuff(raster[i,j,])  # At each pixel, compute the refined coefficient estimates

    }}}




# My biggest complaint about the write.ENVI file from caTools is that it tends to bungle the output header
file

#   information, ignoring the spatial information that the input header has.

# As a result, I cobble together a refined output header as below to preserve the information.
```

```
fileName="Training_Harmonic_Coefficients"
write.ENVI(AC,fileName) # outputting raster
sink(paste(fileName,".hdr",sep="")) # overwriting the sparse output header from the previous line with a more useful one
cat("ENVI
description = { R-language data }
samples = ",Cols,"
lines   = ",Rows,"
bands   = ",nc+ns+1,"
header offset = 0
data type = 5
interleave = bsq
byte order = 0
",Map.Info,"
band names = {
Constant,
Sin 1,
Sin 2,
Cos 1,
Cos 2,
}",sep="")
sink()



################################################
# EWMA Charts


setwd(<WORKING DIRECTORY>)
path=<INPUT PATH>
path2=<OUTPUT PATH>


L=list.files(pattern=glob2rx(<WILDCARD FILENAMES>))
```

```
DateInfo=read.csv("DateInfo.csv",header=T)

Obs=read.ENVI("AIx1000 Stack")
Coefs=read.ENVI("Training_Harmonic_Coefficients")

con = file("AIx1000 Stack.hdr", "r")
A=readLines(con, 12 )
Rows=as.numeric(substr(A[4],11,14)) # the number of lines in the ENVI file
Cols=as.numeric(substr(A[3],11,14)) # the number of samples in the ENVI file
Bands=as.numeric(substr(A[5],11,12)) # the number of layers in the ENVI file
Map.Info=A[10] # spatial registration information for use in writing the header file for the output
close(con)

flags=Obs*0 # Preparing the detection array
nc=2 # number of cosine coefficients
ns=2 # number of sine coefficients

x=DateInfo$DOY # Same initial date information across whole scene for now
timedat=x*2*pi/365 # converting to [0,2pi]
XAll=cbind(rep(1,length(timedat)),sin(t(matrix(rep(c(1:ns),length(timedat)),ncol=length(timedat)))*timedat),
cos(t(matrix(rep(c(1:nc),length(timedat)),ncol=length(timedat)))*timedat))
historybound=max(which(DateInfo$Year<2009)) # identifying which date demarcates the training and
testing periods


pixelstuff=function(k,j,lambda,lsigs,SCREENSIG1,SCREENSIG2){ # Performing EWMA detection on a
pixel-by-pixel basis

  y0=as.numeric(Obs[k,j,]-(XAll%*%Coefs[k,j,])) # Initial residuals
  y01=y0[1:historybound] # residuals for the training period
  y02=y0[(historybound+1):length(y0)] # residuals for the tesing period
  muest=mean(y0[1:historybound]) # initial mean estimates, not really used since they so closely
resemble 0
  sdest=sd(y0[1:historybound]) # initial sd estimates for training period (historical SD in the paper)
  ind0=c(1:length(y0)) # index for date values
```

```
ind01=ind0[1:historybound] # index for training period

ind02=ind0[(historybound+1):length(y0)] # index for testing period

mu=muest #

histsd=sdest


# Converting the date values to days after the start of the first year of the training period

#   Alternately could use first date in the training period

eaYear=c(0,365,365,365,366,365,365)

cuYear=cumsum(eaYear)

x0=cuYear[DateInfo$Year-2004]+DateInfo$DOY # full date history, both periods


UCL0=c(rep(SCREENSIG1,length(ind01)),rep(SCREENSIG2,length(ind02)))*histsd # first-pass upper control limit, low-threshold for the training period and high for the test period


x=x0[Obs[k,j,]>100 & abs(y0)<UCL0] # the first conditional screens away values with sufficently low vegetation index values.  This could be tuned or removed depending on the context

y=y0[Obs[k,j,]>100 & abs(y0)<UCL0] # residuals meeting the requirements

ind=ind0[Obs[k,j,]>100 & abs(y0)<UCL0] # index for the values that met requirements

histsd=sd(y01[which(abs(y01)<UCL0[1:historybound])]) # refined historical SD, recomputed only from the training period dates which met the threshold requirements.

# This refined SD is key to the EWMA success, as the resulting SD is generally quite low and allows for easy signaling of disturbances in the test period



tmp=rep(0,length(y0)) # Dummy vector for the disturbance values


ewma=y[1] # first control chart value is just the first residual


for(i in (2):length(y)){ # note we are only using values which met the threshold requirements

  ewma=c(ewma,(ewma[(i-1)]*(1-lambda)+lambda*y[i])) # computing EWMA values

}


UCL=histsd*lsigs*sqrt(lambda/(2-lambda)*(1-(1-lambda)^(2*c(1:length(y))))) # computing EWMA control limits
```

```
  tmp[ind]=ewma/UCL*(abs(ewma)>UCL) # EWMA values divided by control limits for relative disturbance severity
  # values for which the thresholds were exceeded are coded as 0 here because we overwrote them later, but
  #   they could be coded as another value for extra cloud/shadow detection capability

  # In order to preserve some continuity of disturbance under clouds, we set the disturbance flags for the badly-
  #   behaved dates to be equal to the last logged disturbance level
  if(min(ind)>1){ # If date 1 was "cloudy", force it into the detection vector
    tmp[1]=0  # This avoids an error in the next set
    ind=c(1,ind)
  }

  if(min(ind)==1){
    tmp[-ind]=tmp[c(1:length(y0))[-ind]-1] # Preserve last known disturbance level in the face of badly behaved dates
  }

  tmp  # output the disturbance history for the pixel

}

# I used a simple for loop here, but this could easily be parallelized
for(k in 1:Rows){
  if(k%%50==0){cat(k/Rows*100,"%...\n",sep="")} # progress meter
  for(j in 1:Cols){
    if(sum(Obs[k,j,])>0){ # No need to calculate disturbances for histories of 0's
      flags[k,j,]=pixelstuff(k,j,0.3,3,2,12) #EWMA disturbance record for each pixel added to raster
    }}}

# Refining the ENVI headers to improve on what write.ENVI does
#   Since the record contains information for each known date, the information of the output file matches that of the input file.
# Outputting the EWMA records to an ENVI file
```

```
#  Note the integer format for easier storage and thematic map generation
#  Each layer corresponds to a date from the input file

flags=round(flags) # Conversion to integer values
mode(flags)="integer"
fileName="EWMA Disturbances"
write.ENVI(flags,fileName)
sink(paste(fileName,".hdr",sep=""))
cat("ENVI
description = { R-language data }
samples = ",Cols,"
lines =   ",Rows,"
bands =   ",Bands,"
header offset = 0
data type = 3
interleave = bsq
byte order = 0
",Map.Info,"
band names = {
",paste(DateInfo$Year,DateInfo$Month,DateInfo$Day,rep(",",51)),"
}",sep="")
sink()
```

APPENDIX C. Data and Code for Chapter 4

List of scenes used:

Year	DOY	Month	Day
1985	284	10	11
1986	271	9	28
1987	2	1	2
1987	82	3	23
1987	146	5	26
1987	178	6	27
1987	194	7	13
1987	258	9	15
1987	290	10	17
1987	306	11	2
1988	37	2	6
1988	53	2	22
1988	117	4	26
1988	165	6	13
1988	181	6	29
1988	197	7	15
1988	213	7	31
1988	341	12	6
1989	23	1	23
1989	39	2	8
1989	55	2	24
1989	263	9	20
1989	295	10	22
1990	10	1	10
1990	170	6	19
1990	234	8	22
1990	266	9	23
1990	282	10	9
1990	298	10	25
1991	45	2	14
1991	269	9	26
1991	285	10	12
1991	349	12	15
1992	16	1	16
1992	80	3	20
1992	96	4	5
1992	112	4	21

1992	144	5	23
1992	176	6	24
1992	256	9	12
1992	288	10	14
1993	162	6	11
1993	210	7	29
1993	274	10	1
1994	5	1	5
1994	69	3	10
1994	101	4	11
1994	149	5	29
1994	165	6	14
1994	181	6	30
1994	261	9	18
1995	8	1	8
1995	24	1	24
1995	56	2	25
1995	104	4	14
1995	168	6	17
1995	248	9	5
1995	280	10	7
1995	312	11	8
1996	27	1	27
1996	43	2	12
1996	107	4	16
1996	203	7	21
1996	251	9	7
1996	283	10	9
1997	125	5	5
1997	237	8	25
1997	285	10	12
1997	365	12	31
1998	96	4	6
1998	128	5	8
1998	176	6	25
1998	288	10	15
1998	304	10	31
1998	336	12	2
1999	19	1	19
1999	35	2	4
1999	51	2	20

1999	147	5	27
1999	211	7	30
1999	227	8	15
1999	259	9	16
1999	275	10	2
1999	291	10	18
1999	307	11	3
2000	6	1	6
2000	150	5	29
2000	198	7	16
2000	230	8	17
2000	246	9	2
2000	262	9	18
2000	278	10	4
2000	326	11	21
2000	342	12	7
2000	358	12	23
2001	24	1	24
2001	72	3	13
2001	120	4	30
2001	168	6	17
2001	232	8	20
2001	264	9	21
2001	280	10	7
2001	312	11	8
2001	360	12	26
2002	43	2	12
2002	59	2	28
2002	139	5	19
2002	187	7	6
2002	203	7	22
2002	219	8	7
2002	363	12	29
2003	14	1	14
2003	78	3	19
2003	238	8	26
2003	254	9	11
2003	302	10	29
2003	334	11	30
2004	81	3	21
2004	97	4	6

2004	129	5	8
2004	273	9	29
2004	289	10	15
2004	353	12	18
2005	35	2	4
2005	67	3	8
2005	99	4	9
2005	147	5	27
2005	227	8	15
2005	291	10	18
2005	307	11	3
2006	38	2	7
2006	118	4	28
2006	166	6	15
2006	182	7	1
2006	198	7	17
2006	230	8	18
2006	246	9	3
2006	262	9	19
2006	278	10	5
2006	326	11	22
2006	342	12	8
2007	9	1	9
2007	25	1	25
2007	41	2	10
2007	57	2	26
2008	28	1	28
2008	108	4	17
2008	140	5	19
2008	204	7	22
2008	268	9	24
2008	300	10	26
2008	364	12	29
2009	14	1	14
2009	30	1	30
2009	62	3	3
2009	78	3	19
2009	174	6	23
2009	190	7	9
2009	206	7	25
2009	238	8	26

2009	270	9	27
2010	49	2	18
2010	65	3	6
2010	225	8	13
2010	257	9	14
2010	273	9	30
2010	289	10	16
2010	321	11	17
2010	337	12	3
2010	353	12	19
2011	84	3	25
2011	148	5	28
2011	164	6	13
2011	276	10	3

R code excerpt (run on R 2.15.1 and 2.15.2)

```
require(calibrate)

require(caTools)

require(plyr)

require(foreach)

# For Linux parallelization

require(doMC)

registerDoMC()


###################################

# Use harmonic regression algorithm to obtain 3-year group and running means coefficients in the manner of training coefficients in the EWMA paper, using only constant coefficeint as output



###################################

# Identifying disturbances

setwd(<WORKING DIRECTORY>)

path=<INPUT PATH>

path2=<OUTPUT PATH>
```

```
L=list.files(pattern=glob2rx(<3-YEAR OR RUNNING MEANS FILES, ALL YEARS>))
## Only 3-year processing shown here, but the running means processing is similar with a disturbance and six subsequent years

theFile=L[chunk] #Scene is broken into "chunks" or "parts" here for parallel processing
Part=substr(theFile,27,28)

con = file(paste(path,"/",theFile,".hdr",sep=""), "r")
A=readLines(con, 12 )
Rows=as.numeric(substr(A[4],9,12))
Cols=as.numeric(substr(A[3],11,14))
close(con)



dat=read.ENVI(theFile)
Rows=dim(dat)[1]
Cols=dim(dat)[2]

growdat=dat[,,1:3]*0

for(k in 1:Rows){
  for(j in 1:Cols){
    if(k%%100==0 & j==Cols){cat(k/Rows*100,"% for part ",Part,"...\n",sep="")}
    PoD=(min(which(dat[k,j,-1]-dat[k,j,-9]==min(dat[k,j,-1]-dat[k,j,-9]))+1))
    if(PoD<8){
      growdat[k,j,]=as.integer(as.numeric(dat[k,j,(PoD+0:2)]))
    }
  }
}

fileName=paste(path2,"3YeargroupRecoveryTracksWithPoDPart",Part,sep="")
storage.mode(growdat)="integer"
write.ENVI(growdat,fileName)
sink(paste(fileName,".hdr",sep=""))
```

```
cat("ENVI
description = { R-language data }
samples = ",Cols,"
lines = ",Rows,"
bands = 3
header offset = 0
data type = 3
interleave = bsq
byte order = 0
",A[10],"
band names = {
PoD,
AD1,
AD2,
}",sep="")
sink()



###################################
# Generating cluster objects from means

setwd(<WORKING DIRECTORY>)
path=<INPUT PATH>
path2=<OUTPUT PATH>

L=list.files(pattern=glob2rx(<3-YEAR OR RUNNING MEANS FILES, POST DISTURBANCE>))

dat=read.ENVI(L[1])
Rows=dim(dat)[1]
Cols=dim(dat)[2]
dim(dat)
```

```
diffdat=dat-abind(dat[,,1],dat[,,1],dat[,,1],along=3)


storage.mode(diffdat)="integer"
write.ENVI(diffdat,"FullScene3YearGroupRecoveryTracksSubtractedPoD")
# Employ header modifications in the same way as the EWMA paper for proper registration


normalizer=function(tseries){tseries/max(abs(tseries)+.01)}


normdat=diffdat*0
for(i in 1:Rows){
  if(i%%100==0){cat(i/Rows*100,"% done...\n",sep="")}
  for(j in 1:Cols){
    normdat[i,j,]=normalizer(diffdat[i,j,])
  }
}


normdat=round(normdat*100)
storage.mode(normdat)="integer"
write.ENVI(normdat,"FullScene3YearGroupRecoveryTracksNormalized")



########################################
# HCA Algorithm


## Generating Training Clusters
#
Files=c("AllPartsRegrowthTrackswithPoD","AllPartsRegrowthTracksSubtractedPoD","AllPartsRegrowthTr
acksNormalized")
Files=c("FullScene3YearGroupRecoveryTracksWithPoD","FullScene3YearGroupRecoveryTracksSubtract
edPoD","FullScene3YearGroupRecoveryTracksNormalized")
indexer=c("withPoD","SubtractedPoD","Normalized")


for(chunk in 1:3) { # chunk corresponds to the three NDVI transforms from above in indexer
  theFile=Files[chunk]
```

```
cat("Initializing ",theFile,"...\n",sep="")

con = file(paste(theFile,".hdr",sep=""), "r")
A=readLines(con, 12 )
close(con)

dat=read.ENVI(theFile)
dim(dat)
Rows=dim(dat)[1]
Cols=dim(dat)[2]

dat[sample(1:Rows,1),sample(1:Cols,1),]

################################################
#Setting parameters
lo=4
hi=12
vmin=1
Layers=dim(dat)[3]
vmax=dim(dat)[3]
sampsize=10000

################################################
## Calculated parameters
# For generating a new sample and storing it
sampixels=sample(1:(Rows*Cols),sampsize)
Xsam=sampixels%%Rows+1
Ysam=sampixels%%Cols+1

write.csv(cbind(Xsam,Ysam),"Training Sample Points.csv",row.names=F)

# For reading stored sample data
cat("Reading training data...\n",sep="")
```

```
traindat=read.csv("Training Sample Points.csv",header=T)
Xsam=traindat[,1]
Ysam=traindat[,2]

  td=matrix(rep(0,sampsize*Layers),ncol=Layers)
for(i in 1:sampsize){
  td[i,]=dat[Xsam[i],Ysam[i],]
}

cat("Clustering...\n",sep="")

nsamp=dim(td)[1]
nobs=Rows*Cols
crange=hi-lo+1
vrange=vmax-vmin+1
d=dist(td[,vmin:vmax], method="manhattan")
hc=hclust(d, method="ward")
Centroids=matrix(rep(0,vrange*sum(c(lo:hi))),nrow=sum(c(lo:hi)))
memb=matrix(rep(0,nsamp*crange),ncol=crange)
MEMB=matrix(rep(0,nobs[1]*crange),ncol=crange)
MEMBdisttocentroids=matrix(rep(0,nobs[1]*crange),ncol=crange)
date()


## Calculating Cluster Centroids
N=0
Ns=rep(0, crange-1)
for(n in lo:hi){
  #Clustering on the Sample
  memb[,(n-lo+1)]=cutree(hc, n)
  for(i in 1:n){
    #print(N+i)
    Centroids[N+i,]=colMeans(td[memb[,(n-lo+1)]==i,vmin:vmax])
```

```
  }
  N=sum(c(lo:n))
  Ns[n-lo+1]=sum(c(lo:n))
  print(N)
}
#Ns

cat("Generating Centroids...\n",sep="")
#making group names for centroids matrix
centnames=rep(0, max(Ns))
indexer=rep(0, max(Ns))
MEMBnames=rep(0, crange)
for(n in lo:hi){
  indexer[(Ns[(n-lo+1)]-n+1):(Ns[(n+1-lo)])]=c(1:n)
  centnames[(Ns[(n-lo+1)]-n+1):(Ns[(n+1-lo)])]=paste(n,"ClustersClass",indexer[(Ns[(n-lo+1)]-
n+1):(Ns[(n+1-lo)])],sep="")
  MEMBnames[n-lo+1]=paste(n,"Clusters",sep="")
}

colnames(Centroids)=colnames(dat)[vmin:vmax]
colnames(MEMB)=MEMBnames

## Filling out the map

cat("Beginning nearest neighbor fill...\n",sep="")
# date()
foreach(N=lo:hi) %dopar% {
  MEMB=matrix(rep(0,Rows*Cols),nrow=Rows)
  cthresh=6000


  cat("Assigning Clusters for group ",N,"...\n",sep="")
  Cent=Centroids[(Ns[(N-lo+1)]-N+1):(Ns[(N+1-lo)]),]
```

```
    for(i in 1:Rows){
      if(i%%300==0){cat(i/Rows*100,"% done...\n",sep="")}
      for(j in 1:Cols){
        # print(round(c(((i-1)*Cols+j)/(Rows*Cols)*100),2))
        cdist=c()
        for(n in 1:N){
          #print(n)
          cdist=rbind(cdist,(sum(abs(dat[i,j,]-Cent[n,]))))
        }
        MEMB[i,j]=c(1:N)[rank(cdist)==1]*(min(cdist)<cthresh)

      }
    }
    cat("Writing Cluster Solution for group ",N,"...\n",sep="")
    storage.mode(MEMB)="integer"

fileName=paste(path2,"HCA_Classifications_3_year_groups_Full_Scene_",N,"_Cluster_Solution_",index
er[chunk],sep="")
    write.ENVI(MEMB,fileName)


    sink(paste(fileName,".hdr",sep=""))
    cat("ENVI
description = { R-language data }
samples = ",Cols,"
lines = ",Rows,"
bands = 1
header offset = 0
data type = 3
interleave = bsq
byte order = 0
",A[10],"
",sep="")
sink()
```

```
  }
}

###################################
# Post stratified estimation

## Greenbook functions (Scott et al. 2005), provided by John Coulston

#function calculates strata means and variances
#colnum is the number of the column for the variable you want to estimate
#use columns 2 through 8
#cntcol is used to determine n  use either col 9 or 10
#the remeasured plots are <= current status plots because of hazardous/denied access
#to estimate carbon, forest area, area cut, area planted use col 10 'status'
#for growth removals or mortality use col 9 'grm'

eq4.11_4.12.fun<-function(yhid.in,colnum,cntcol){

  n.h<-sum(yhid.in[,cntcol])
  y.hid<-yhid.in[,colnum]
  ybar.hd<-mean(y.hid,na.rm=T)
  var.ybar.hd<-((sum(y.hid^2,na.rm=T))-(n.h*(ybar.hd^2)))/(n.h*(n.h-1))
  out<-data.frame(yhid.in$Rastervalu[1])
  names(out)<-"stratumcd"
  out$ybar.hd<-ybar.hd
  out$var.ybar.hd<-var.ybar.hd
  out$nh<-n.h
  return(out)
}


#this function works off of output from eq4.11_4.12.fun
#must also have strata weight W.h and total area A.t
```

```
eq4.13_4.14.fun<-function(stratmeans,description){
  n<-sum(stratmeans$nh)
  stratmeans$nhdn<-stratmeans$nh/n
  A.t<-stratmeans$A.t[1]
  out<-data.frame(description)
  names(out)<-"description"
  out$ysum.d<-sum(stratmeans$ybar.hd*stratmeans$W.h*A.t)
  out$var.ysum.d<-((A.t^2)/n)*(sum(stratmeans$W.h*stratmeans$nh*stratmeans$var.ybar.hd)+sum((1-
stratmeans$W.h)*stratmeans$nhdn*stratmeans$var.ybar.hd))
  return(out)
}



## Estimation, baseline code provided by John Coulston


setwd("")
require(calibrate)
list.files()
source("greenbook.functions.r")
plt.val<-read.csv("")  #plot value  see readme.txt for definitions
can.ppsa<-read.csv("")  #plot strata from overlay with  map


head(can.ppsa)



nameindex=c(  "3y10c1"  ,  "3y04c1"      ,  "3y04c2"      ,  "3y04c3"      ,  "3y05c1"      ,  "3y05c2"
          ,  "3y05c3"      ,  "3y06c1"      ,  "3y06c2"      ,  "3y06c3"      ,  "3y07c1"      ,  "3y10c2"
          ,  "3y07c2"      ,  "3y07c3"      ,  "3y08c1"      ,
           "3y08c2"        ,  "3y08c3"      ,  "3y09c1"      ,  "3y09c2"      ,  "3y09c3"      ,  "rm10c1"
          ,  "rm10c2"      ,  "3y10c3"      ,  "rm10c3"      ,  "rm11c1"      ,  "rm11c2"      ,  "rm11c3"
          ,  "rm12c1"      ,  "rm12c2"      ,  "rm12c3"      ,  "rm04c1"      ,  "rm04c2"      ,  "rm04c3"
          ,
           "3y11c1"        ,  "rm05c1"      ,  "rm05c2"      ,  "rm05c3"      ,  "rm06c1"      ,  "rm06c2"
          ,  "rm06c3"      ,  "rm07c1"      ,  "rm07c2"      ,  "rm07c3"      ,  "rm08c1"      ,  "3y11c2"
          ,  "rm08c2"      ,  "rm08c3"      ,  "rm09c1"      ,  "rm09c2"      ,  "rm09c3"      ,  "3y11c3"
          ,  "3y12c1"      ,  "3y12c2"      ,  "3y12c3"      )
```

```
round(apply(can.ppsa,2,max))


EUpath=""
EUdir=list.files(EUpath,pattern=glob2rx("??????.csv"))
EUdir


AllVars=array(dim=c(7,4,54))
dimnames(AllVars)=list(c("Carbon","% Forest","% Cut","%
Planted","Removal","Mortality","Growth"),c("ysum.d","var.ysum.d","RE","MinNh"),nameindex)


dimnames(AllVars)
for(theFile in 1:54){



  print(nameindex[theFile])
  can.eus=read.csv(paste(EUpath,nameindex[theFile],".csv",sep=""),header=T)


  can.eus$W.h<-can.eus$Count_/sum(can.eus$Count)


  A.t<-247.1*(sum(can.eus$Count))*30*30/1000/1000  #this is the total area of our population in acres
  # can.ppsa<-can.ppsa[,-5]
  names(can.ppsa)[theFile+2]<-"Rastervalu"
  can.ppsa2<-subset(can.ppsa,can.ppsa$Rastervalu!=-9999)


  ###canopy cover stratification estimates
  plots<-merge(plt.val,can.ppsa2[,c(2,theFile+2)],by="PLT_CN",all.y=T) #merge the plot stratification file
(can.ppsa2)
  #with the plot file (plt.val).


  names(plots)
  varindex=c(1:7)+1
  statindex=c(10,10,10,10,9,9,9)
```

```
  vars=c()
  res=c()

  for(i in 1:7){

    strat.means<-by(plots,plots$Rastervalu,FUN=eq4.11_4.12.fun,varindex[i],statindex[i])    #estimate
stratum means and variances
    #this calls a function in greenbook.functions.r
    #you must specify the column for the variable you
    #want to estimate.  Column 3 in this case because
    #we want to make a forest area estimate.  You must
    #also specify the column to determine n.  Column 10 in
    #this case because it's a 'status' estimate.  Open
    #greenbook.function.r for some more details.

    strat.means.df<-data.frame(do.call("rbind",strat.means))

    strat2<-merge(strat.means.df,can.eus,by.x="stratumcd",by.y="Value_") #merge the strata weights
(can.eus) with the
    #strata estimates
    strat2$A.t<-A.t                                    #this is just the area (acres) of the population

    vars=rbind(vars,eq4.13_4.14.fun(strat2,names(plots)[i+1]))

    plots$SRS=round(plots$Rastervalu/plots$Rastervalu)
    srs.means<-by(plots,plots$SRS,FUN=eq4.11_4.12.fun,varindex[i],statindex[i])
    srs.means.df<-data.frame(do.call("rbind",srs.means))
       srs2=srs.means.df
    srs2$OID=NA
    srs2$Value_=1
    srs2$Count_=as.numeric(colSums(can.eus)[3])
    srs2$W.h=1
    srs2$A.t<-A.t
```

```
    SRSVar=eq4.13_4.14.fun(srs2,names(plots)[i+1])[3]
    StratVar=eq4.13_4.14.fun(strat2,names(plots)[i+1])[3]
    res=rbind(res,as.numeric(SRSVar/StratVar))
    boxplot(plots[,varindex[i]]~plots$Rastervalu)
  }


  vars$RE=round(res,2)
  vars$MinNh=min(strat.means.df$nh)

  AllVars[,,theFile]=as.matrix(vars[2:5])

}


REplot=function(var){
  Varnames=c()
  Mean.Type=substr(nameindex,1,2)
  Number.Clusters=as.numeric(substr(nameindex,3,4))
  Normalization.Style=substr(nameindex,6,6)

  var1=Mean.Type
  var2=Normalization.Style
  resp=as.numeric(AllVars[var,3,])
  MinNs=as.numeric(AllVars[var,4,])

  legnames=c("3-Year, Non-Normalized", "3-Year, Difference", "3-Year, Normalized",
             "Running, Non-Normalized", "Running, Difference", "Running, Normalized")

  Method=paste(Mean.Type,Normalization.Style,sep="")
  Mean.Type=substr(nameindex,1,2)
  MT=(1*(Mean.Type=="rm"))
```

```
  Number.Clusters=as.numeric(substr(nameindex,3,4))
  Normalization.Style=substr(nameindex,6,6)


  names(table(Method))
  colrs=c("red1","red2","red3","cyan1","cyan2","cyan3")
  pchs=c(19,15,18,19,15,18)
  require(calibrate)


  plot(0,0,xlim=c(4,15),ylim=c(min(resp),quantile(resp,.95)),col="white",ann=F,cex.axis=1.4)
  title(xlab="Number of Strata",ylab="Relative Efficiency",main=paste("Comparison of Relative Efficiencies
for Parameter ",dimnames(AllVars)[[1]][var],sep=""),cex.main=1.4,cex.lab=1.4)
for(method in 1:6){

xvals=Number.Clusters[Method==names(table(Method))[method]][order(Number.Clusters[Method==nam
es(table(Method))[method]])]

yvals=jitter(resp[Method==names(table(Method))[method]][order(Number.Clusters[Method==names(table
(Method))[method]])])
  points(xvals,yvals,pch=pchs[method],col=colrs[method],"b")

textxy(jitter(xvals),jitter(yvals),MinNs[Method==names(table(Method))[method]][order(Number.Clusters[M
ethod==names(table(Method))[method]])],cx=1)
}
  legend("right",c(legnames),col=c(colrs),pch=pchs,cex=1.3)



}


REplot(1)


#### Making an Overall Results Plot
Mean.Type=substr(nameindex,1,2)
MT=(1*(Mean.Type=="rm"))
Number.Clusters=as.numeric(substr(nameindex,3,4))
Normalization.Style=substr(nameindex,6,6)
```

```
plot(0,0,xlim=c(.8,7.2),ylim=c(min(AllVars[,3,]),1.5),axes=F,ann=F)

box()

title(main="Relative Efficiency Comparison for All Methods and Parameters",

    xlab="Forest Parameters",font.lab=2)

title(ylab="Relative Efficiency")

axis(2)

lines(c(4.5,4.5),c(.8,1.37))

axis(1,at=c(2.5,5.5),pos=.84,tick=F,labels=c("Static","Dynamic"),font.axis=3)

axis(1,at=c(4.5),pos=.86,tick=T,labels=F)

axis(1,at=c(4.5),pos=.85,tick=T,labels=F)

axis(1,at=c(4.5),pos=.84,tick=T,labels=F)

axis(1,at=c(4.5),pos=.83,tick=T,labels=F)

axis(1,at=c(4.5),pos=.82,tick=T,labels=F)

axis(1,at=c(4.5),pos=.81,tick=T,labels=F)

axis(1,at=c(4.5),pos=.80,tick=T,labels=F)


axis(1,at=c(1:7),pos=.87,tick=F,labels=c("Carbon","Forest","Cut","Planted","Removal","Mortality","Growth"
))

axis(1,at=c(1:6)+.5,tick=T,labels=F)


for(var in 1:7){

  #   oints(jitter(rep(var,54),
amount=.3),(AllVars[var,3,]),bg=c("red","cyan4")[MT+1],col="black",pch=as.numeric(Normalization.Style)+
20,cex=(16-Number.Clusters)/6)

  points((var+.4*(Number.Clusters-
8)/5),(AllVars[var,3,]),bg=c("red","cyan4")[MT+1],col="black",pch=as.numeric(Normalization.Style)+20,cex
=(16-Number.Clusters)/6)

  lines(c(var+.5,var+.5),c(.8,1.6-.23*sum(var==3 | var==4)),lty=2,lwd=.5)

}

legend("top",c("3-Year","Running","Non-
Normalized","Differenced","Normalized"),col=c(NA,NA,"black","black","black"),bg=c(NA,NA,"black","black"
,"black"),pch=c(NA,NA,16,15,18),text.col=c("red","cyan4","black","black","black") ,cex=1)
```