2017 IEEE International Conference on Data Mining Workshops

Quantifying seasonal patterns in disparate environmental variables using the PolarMetrics R package

Bjorn-Gustaf J. Brooks
Danny C. Lee
Lars Y. Pomara
and William W. Hargrove
Eastern Forest Environmental Threat Assessment Center
Southern Research Station
USDA Forest Service
Asheville, NC, USA
Email: bjorn@geobabble.org

Ankur R. Desai
University of Wisconsin-Madison
Department of Atmospheric and Oceanic Sciences
Madison, WI, USA

***Abstract*—Certain environmental processes, while influential, are inherently difficult to quantify and detect using traditional time series analyses, particularly among variables with different seasonal progressions. Disturbances that only manifest in part of a season (e.g., spring defoliation) or subtle climate shifts can pose detection challenges when they occur in the presence of other variability. Increasing sampling rates or even adding new sensors may not reveal the anticipated patterns. Eddy covariance tower data are a useful example for which various environmental drivers influence the overall signal, contributing noise and seemingly discordant variation. While eddy flux data are a rich representation of information, distinguishing expected seasonal responses within a signal can be challenging, especially where drivers may have either fast or lagged responses. A conventional solution might be to analyze and effectively smooth the data over daily to monthly intervals. However, such smoothed data will not exhibit the same variance, and subsequent regressions may not isolate relationships and anomalies to specific seasons. This paper introduces and demonstrates the use of a newly developed R software package, PolarMetrics, which is used to analyze 20 years of data from one AmeriFlux tower using a polar (circular) approach that reduces data volume to a smaller set of derived seasonal timing and magnitude metrics. Polar metrics quantify the annual cycle of input variables, and permit direct comparison of the strength and timing of seasonality. While performing the analysis over all years produces a synoptic result, analyzing year-by-year characterizes interannual variability.**

I. Introduction

We introduce a new R software package, PolarMetrics, for analyzing seasonal patterns of change in environmental time series. A polar metrics data reduction approach to noisy environmental data can aid in detection of seasonal and multi-annual signals and relationships among variables.The examples in this paper are focused on relationships between multiple environmental variables to carbon exchange, but the polar approach is generic, and can be used to characterize the annual cycles present in many environmental variables. Seasonal lags in relationships between canopy ecosystem carbon exchange and related environmental variables has been studied [1]. Here we offer a complementary approach that further examines inter-annual changes in timing, seasonality and the degree to which these variables vary over inter-annual timescales. Linear and regression analysis have already revealed insight about the interplay of factors affecting ecosystem carbon exchange processes, but a polar approach can provide additional insight by (1) Isolating the analysis to just a portion of the year of greatest activity/interest (e.g. summer extremes), (2) Reducing data volume by expressing data through relatable seasonal timing and magnitude measures that are comparable across variables, and (3) Comparing environmental variables based on the same calendar year or based on their own empirically derived year, which is often shifted from the human calendar year.

In this work we do not intend to fully explore and explain the history and usage of polar transformation. While polar transformation has a long history of use in atmospheric sciences, engineering, and astronomy, more recently, a number of noteworthy papers in ecological and environmental fields have found new applications [2], [3], [4], [5]. While this study focuses on the potential utility of polar metrics extracted from multiple environmental data streams collected at a single site, our intent in developing this R package and technique concerns its application to land surface phenology data as part of a landscape dynamics analysis system. That application, however, will be described in future papers.

The process of translating time series into polar measures begins with radial time coordinates and projecting the data onto a polar plane, just as with wind direction data. The polar coordinate data are transformed into vector components to facilitate the calculation of several derived polar measures, each of which quantify an aspect of the seasonal evolution of the data. Table I gives some example output of three polar metrics (early season day of year, late season day

2375-9259/17 $31.00 © 2017 IEEE
DOI 10.1109/ICDMW.2017.45

IEEE computer society

TABLE I
ANNUAL POLAR METRICS (ES: EARLY SEASON DOY, LS: LATE SEASON DOY, SMA: SEASONAL MAGNITUDE) CALCULATED FOR AIR TEMPERATURE, PRECIPITATION, AND NET ECOSYSTEM EXCHANGE. TIMING METRICS, ES & LS, SHOW YEAR TO YEAR CHANGE IN THE RATE OF SEASONAL PROGRESSION. MAGNITUDE METRIC (SMA) MEASURES THE ASYMMETRY OR SEASONAL INTENSITY, DEFINED AS THE LENGTH OF THE AVERAGE VECTOR BETWEEN ES AND LS.

	Air Temperature			Precipitation			Net Ecosys. Exch.		
Year	ES	LS	SMa	ES	LS	SMa	ES	LS	SMa
1995	112	308	0.76	139	279	0.18	65	310	0.60
1996	111	300	0.76	96	236	0.11	64	309	0.61
1997	109	305	0.76	101	290	0.14	62	300	0.62
1998	101	304	0.79	93	296	0.10	61	306	0.60

of year and seasonal magnitude) calculated from hundreds of measurements per environmental variable. Polar metrics concatenate and summarize each year of measurements and permit direct comparison on a year-to-year basis. An added advantage of polar transformation is that timing metrics can be calculated for the calendar year (e.g., days starting from Jan 1), or can be instead calculated on a variable year that is rotated to begin at the time of least activity of the variable, which may be important for analyses that seek to compare data across latitudinal, elevational, or other gradients.

II. METHODOLOGY

The PolarMetrics R package including documentation, sample NDVI data, and example analyses can be downloaded or installed directly from https://github.com/bjornbrooks/PolarMetrics. The focus here is to use the PolarMetrics package to evaluate seasonal features in NEE and other eddy flux data. We also compare the signal of NEE productivity against MODIS NDVI extracted and averaged over the tower footprint. This paper includes supplemental R code that was used to produce the figures and results. These scripts are available separately from https://github.com/bjornbrooks/PolarMetricsMSscripts. After installing the PolarMetrics R package, the main program in PolarMetricsMSscripts, `mk_figs.R`, reproduces the figures and results contained in this paper. `mk_figs.R` handles the downloading, loading, parsing, gap-filling, and plotting that is summarized in Procedures 2–5 below.

A. Park Falls site and data description

The Park Falls eddy covariance tall tower (WLEF, http://ameriflux.lbl.gov/sites/siteinfo/US-PFa) [6] is located in northern Wisconsin and is among the tallest flux towers in operation. Measurements from the 396 m level have a daytime fetch on the order of 5 km, integrating over a landscape of upland forests and wetlands. Winters can be cold and long, and the forests are mainly deciduous but also include a substantial coniferous fraction. Differences in wetland to upland land cover and patchiness of the landscape due to management contribute substantially to landscape heterogeneity [1]. Among over thirty available variables that are routinely collected, this paper focuses on the following four:

```
Procedure 1
# The following packages are suggested
#   additions for PolarMetrics
install.packages(c("MODISTools",
   "plotrix", "xts"))
# If not already present,
#   install devtools
install.packages("devtools")
# Now install PolarMetrics
devtools::install_github(
  "bjornbrooks/PolarMetrics")
download.file( # Download supp. material
  paste("https://github.com/bjornbrooks/",
        "PolarMetricsMSscripts/",
        "archive/master.zip",sep=""),
        destfile="master.zip")
unzip("master.zip")   # Extract files
setwd("PolarMetricsMSscripts-master")
library(PolarMetrics) # Load library
source("mk_figs.R")   # Reproduce figs
```

30 meter air temperature (TA, °C), precipitation (P, mm), net ecosystem exchange from the 396 m level (NEE_PI, μmol CO_2 $m^{-2}s^{-1}$), and incoming photosynthetic photon flux density (PPFD_IN, μmol Photon $m^{-2}s^{-1}$). WLEF tower data were downloaded from the AmeriFlux data distribution portal (http://dx.doi.org/10.17190/AMF/1246090). In addition to WLEF tower data we added the Normalized Difference Vegetation Index (NDVI) as a remotely sensed vegetation index for comparison to carbon balance (Net Ecosystem Exchange). NDVI data from MODIS (http://modis.gsfc.nasa.gov/) data subsets were downloaded using the MODISTools R package [7] for retrieving MODIS subsets through the Oak Ridge National Laboratory (ORNL) DAAC web service (SOAP). The NDVI data used here are a spatial average of 289 MODIS pixels spanning a $\sim$4 km^2 area surrounding the WLEF tower (see `get_modis.R` script).

B. Step 1, Load data

First manually download and extract AmeriFlux BASE data file (e.g., AMF_US-PFa_BASE_HR_9-1.csv). These data represent multiple variables collected over 21 years 1995–2015). The supplemental code repository above contains a master script, `mk_figs.R`, that will processes the data by first loading the WLEF time series, as in Procedure 2. `mk_figs.R` also automatically downloads MODIS NDVI data (2000–2015), through a wrapper script `get_modis.R`, which requires the MODISTools R package. WLEF data are smoothed from hourly to 7-day averages (Fig. 1), while the NDVI data are spline interpolated from 16-day to 7-day values. Note that Procedure 2 outlines only the basic steps of data development. For example a step not shown in Procedure 2 is gap-filling. Gap-filling was achieved by replacing missing 7-day values with corresponding averages for that point in the annual cycle. That is, a series of 52 values (averages) was developed from

```
Procedure 2
# Load hourly WLEF tower data
lef <- read.csv(
  file = "AMF_US-PFa_BASE_HR_9-1.csv")
# Make time stamp variable into a
#   date-time object
date.time <- as.POSIXct(
  gsub("(.{8})", "\\1 ",
       lef$TIMESTAMP_START),
  format="%Y%m%d %H%M",
  tz="America/Chicago")
# Make lef an xts object
lef.xts <- xts(lef, order.by=date.time,
  tz = "America/Chicago")
lef.dy <- apply.daily(lef.xts, FUN=mean,
  na.rm=T) # Calculate daily averages
ep <- seq(from=1, to=365,
  by=7) # Define 7-day sampling points
lef.7dy <- rep(NA, 52*21) # Initialize
                          # output array
for (I in 1:21) { # Loop over all yrs
  yr <- I + 1994
  bidx <- 1 + (I-1) * 52
  eidx <- bidx + 51
  dy <- lef.dy[paste(yr)] # Use xts
    # features to create annual subset
  dy7 <- period.apply(dy, INDEX=ep,
    FUN=mean, na.rm=T) # 7-day averages
  lef.7dy[bidx:eidx] <- dy7 # Insert
                  # into output array
}
```

```
Procedure 3
# Load PolarMetrics
library(PolarMetrics)
# Extract days of yr from xts time index
t <- as.numeric(strftime(index(lef.7dy),
                         format = "%j"))
# Transform DOY to radians
r <- t2rad(t, dpc=365)
```

all the 7-day values by calculating the average value for each 7-day period in the year, across all years of data. Any given 7-day period without a value was imputed by its corresponding average.

C. Step 2, Calculate angles from calendar dates

Convert time series dates into radians in the range 0–2 pi (Procedure 3). Now that time coordinates are transformed into angles (radians) the environmental variable values (e.g., °C, %, μmol $m^{-2}s^{-1}$) can be plotted on a polar plane (Fig. 2). This reveals a series of overlapping orbits. The irregularities in the orbits reflect variation in seasonality between years.

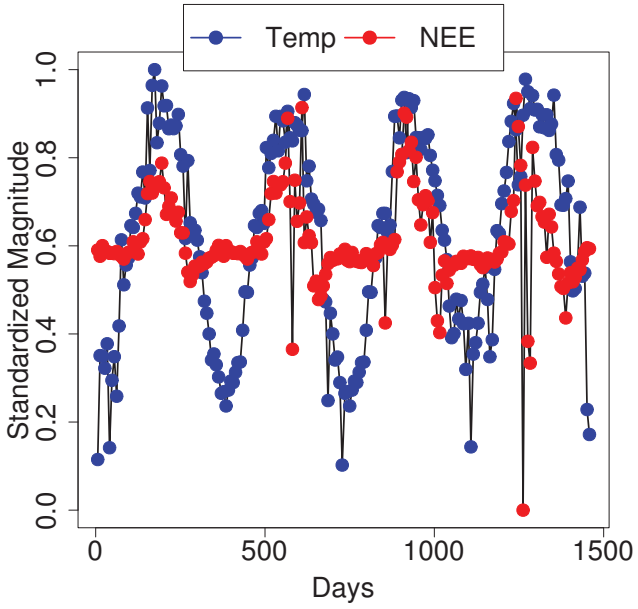

Fig. 1. Example weekly time series of air temperature and NEE over four example years (1995–1998)

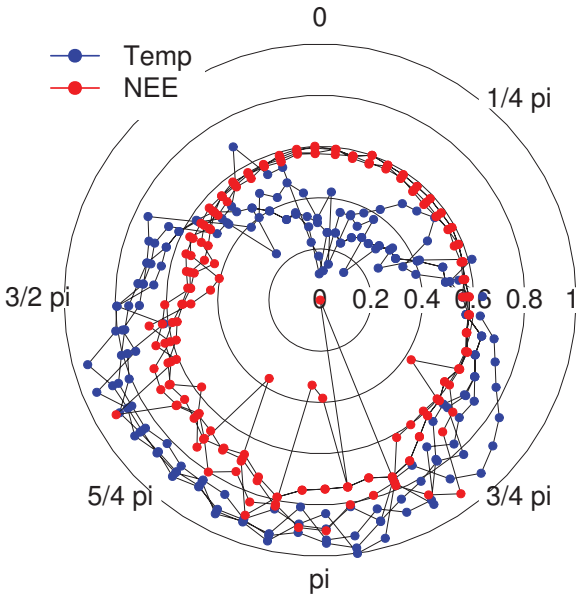

Fig. 2. Example polar plot of the same data as in Fig. 1. Dates are transformed to radial coordinates, listed around the perimeter of the polar plot. January 1 occurs just after 0 pi. The calendar year progresses clockwise toward July 1, which occurs at about 1 pi and continues through December 31 at 0 pi. Measurement values are expressed as the distance from the central axis, which is standardized from 0 to 1 for each variable.

```
Procedure 4
# Transform first variable into its
#   component vectors
vecs <- calc_metrics(lef.7dy$TA,
  yr_type="cal_yr", spc=52,
  lcut=0.15, hcut=0.85,
  return.vecs=TRUE)$vectors
# Repeat calculation and return overall
# average vectors (RV and AV)
avg_vecs <- calc_metrics(lef.7dy$TA,
  yr_type="cal_yr", spc=52,
  lcut=0.15, hcut=0.85,
  return.vecs=TRUE)$avg.vectors
```

D. Step 3, Calculate polar vectors [optional]

The calculation of the component vectors is embedded within the `calc_metrics` function, but the component vectors can be output for specific analysis. `calc_metrics` is a wrapper script that automatically calls several PolarMetrics base functions to derive the final annual polar metrics. Setting the argument `return.vecs = TRUE` will return, as a list object, the horizontal and vertical component vectors in addition to the resultant and anti-vectors.

The resultant vector `RV` has a direction (angle) and force (magnitude) proportional to the resultant force of all vectors, and its anti-vector (`AV`), can be thought of as the point in the year of least activity. Both of these vectors form a straight line bisecting the polar graph into two areas of equal mass (Fig. 3). That is, the sum of values to one side of the line is equivalent to the sum on the other side. The segment of the bisecting line pointing in the direction of the larger seasonal values is the resultant vector, (`RV`). Its magnitude (length), reflects the degree to which values are massed in the direction of the average vector, and quantifies the strength of seasonality. Highly seasonal data will have large average vector magnitudes. In the opposite direction, the anti-vector `AV`, is the least active period in terms of vector displacement, indicating a logical average start of the "vector centered year" (or the offset from the calendar year). Different environmental variables (temperature, precipitation) will have seasonalities with different values of `AV`. Furthermore, measurement of the same environmental variable from different geographic locations can results in differing values of `AV` due to differences in latitude, elevation, and other physiographic factors.

E. Step 4, Calculate polar metrics

The resultant vector and its anti-vector vector represent aggregations over the entire input series. However, one novel approach made possible by the vectorization is to divide the time series into regular cycles (as in Table I) that begin and end at the point of least seasonal activity (`AV`). One can then examine differences in seasonality, which correspond to each variable's own empirically defined season. The empirical seasonality of a series is frequently not synchronized to

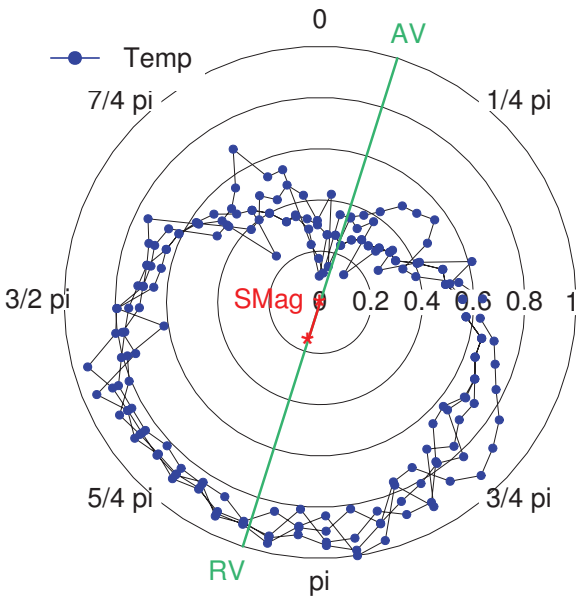

Fig. 3. Polar plot of one year of weekly averaged 30 m air temperature data at WLEF showing resultant vector `RV`, its magnitude (or seasonality `SMag`), and anti-vector `AV`, which defines the start of the vector centered year and marks the offset of air temperature at WLEF from the human calendar.

calendar year, and `AV` can be extracted for use as a descriptive metric of that degree of offset.

For many ecosystem processes, changes that are expressed during a primary season, e.g., an interval of high plant productivity or biogeochemical cycling, are often of greater interest than changes that occur during periods of reduced activity, which may contribute unwanted complexity to the analysis. One way to emphasize a primary season of interest is to exclude some portion of the year using defined fixed thresholds for the season dates of interest. But this ignores inter-annual variation. An alternative presented here is to use thresholds for the cumulative annual totals themselves as they accumulate throughout the year. How much data to exclude from the beginning and end of each cycle is arbitrary (see `lcut` and `hcut` arguments in `calc_metrics`), and can be adjusted from 0–99% of the accumulated annual total, using the `calc_metrics` function. Here we have excluded data corresponding to the initial and final 15% of the cumulative annual total for each year (see Fig. 4) resulting in the creation of a primary-season subset containing 70% of the accumulated total.

`calc_metrics` calls several base functions in the PolarMetrics package, principally `sum_cycle`, which as discussed above, accumulates values as a proportion of the total within each year. `window_idx` subsequently identifies and filters values that do not correspond to the primary season. Ultimately, the metrics of interest are calculated from the remaining subset for each year (in this case the 70% of accumulated total surrounding the annual peak). The exact

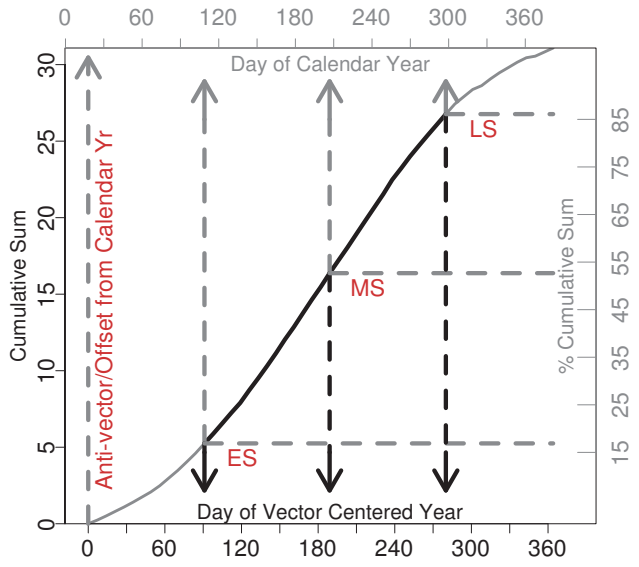

Fig. 4. Cumulative values and thresholds. Beginning from the seasonal minimum values are cumulatively summed within each year. Early season threshold (ES) is calculated as the day of year corresponding to 15% of the cumulative total. The day of year generated by `calc_metrics` is given either according to the calendar year (days since Jan. 1) or the variable year (days since the average variable minimum).

timing of values that are included in the primary season subset can differ from year to year based on the rate of accumulation of values throughout the year. Also of key importance to note is that annualized polar metrics (for the same given output year from `calc_metrics`) can capture primary seasons that occur in different times of the calendar year. Consider for example the lag in temperature seasonality from seasonality in insolation, or the seasonality of precipitation in the Eastern US which is nearly 180 days out of phase with the peak of precipitation in American Southwest. One should keep in mind during any analysis that uses PolarMetrics that each input variable has its own empirically defined start of year, `AV`. Whenever `AV` is offset from the calendar year, both the initial values prior to `AV` and final values after the last complete cycle are discarded. Thus, any time series of n complete calendar years will have $n-1$ cycles in the resulting output.

The `calc_metrics` function is included in PolarMetrics simply for convenience as a faster way to run the data through all the steps of polar transformation. Any variety of custom measures can be explored and calculated using the base functions in the PolarMetrics package. See the source code in the PolarMetrics/R/ directory.

Fig. 5 illustrates several timing and magnitude similarities and differences among environmental variables at the WLEF tower. Although NDVI, temperature, and precipitation all have very different variances they can be compared in terms of their peak in mid-season timing (MS). This contrasts from

```
Procedure 5
# Calc. polar measures for first variable
metrics <- calc_metrics(lef.wk$TA,
                        yr_type="CalYr", spc=52,
                        lcut=0.15, hcut=0.85,
                        return.vecs=FALSE)$metrics
```

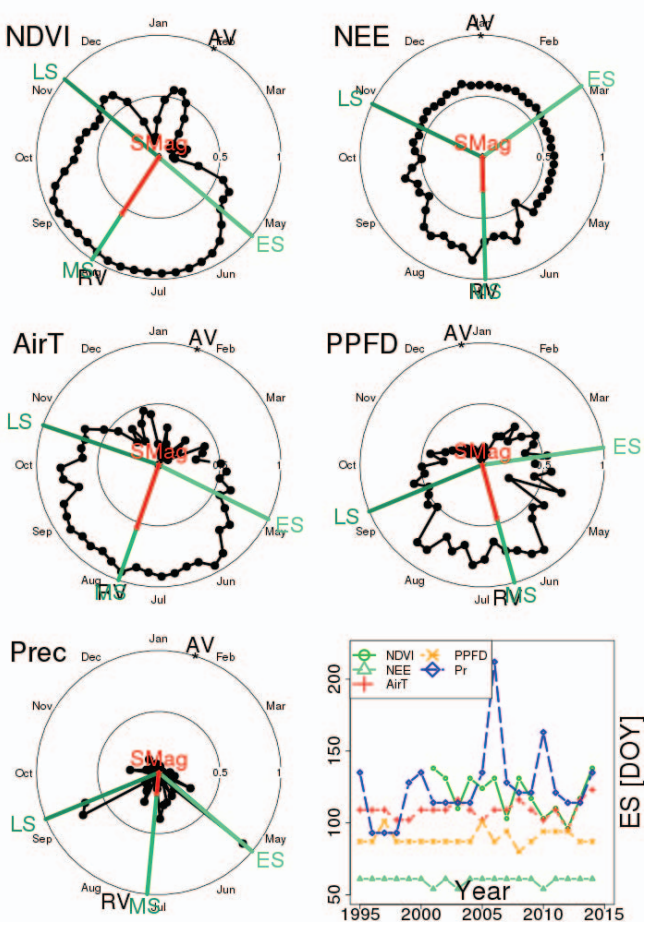

Fig. 5. Polar plots showing timing and magnitude metrics for NDVI, NEE, temperature, photon flux, and precipitation for the 2014 year as well as a time series showing early season timing across all environmental variables for all years. Timing metrics ES, MS, LS are shown by green vectors. Seasonality (`Smag`) is given by the length of the red line. Note that `RV` marks the timing of the average vector and `AV` marks start of the vector-centered year.

NEE and PPFD variables, which tend to peak about one month earlier. Although these polar metrics are derivatives, each measure represents a facet of the seasonal progression, which is dependent on the entire season of values. If tracked through time the dynamics of inter-annual variable change can be explored, as shown in the sub-plot in the lower right of Fig. 5.

III. Results and Discussion

This paper introduces the PolarMetrics R package and describes how to use it to derive seasonal timing and mag-

nitude measures from environmental time series. Polar plots have visual advantages in that they highlight departures from seasonal patterns. Polar metrics have quantitative advantages in that they enumerate differences between years (or cycles). Polar metrics compactly characterize a variety of seasonal features whose timing can be relevant to ecosystem function; however, the same metrics (e.g., length of season) could be similarly calculated by non polar approaches. This begs the question: What advantage is gained from polar transformation? Furthermore, 'Do polar metrics help us recognize patterns that are not otherwise apparent through time series analysis?', and 'Do polar metrics offer any computational savings?'

Above all, polar metrics are simpler to interpret than many time series. Changes in seasonal timing, symmetry and amplitude are directly revealed in polar derived measures (see Table I). Polar metrics also are a starting point for further exploration of relationships between infrequently examined and lagged metrics, which for example, could be explored through a correlation matrix of the annual polar metrics values.

While a correlation matrix of seasonal measures could be put together using non-polar approaches, it is the development of vectors that gives the polar transformation approach certain analytical advantages that are not possible through conventional time series methods. The PolarMetrics package facilitates new variations on analyses through its component vector output (i.e., `vecs` in Procedure 4). Polar vectors have a dual quality because they provide information about force (intensity) and direction (timing) that can be instructional. For example, in Fig. 6 we applied a 1-year sliding window function to weekly component vectors generated by `calc_metrics` in order to search for nonstationarity in WLEF NEE and precipitation data and to investigate its source. For each input time series a new series of resultant vector values was calculated, each reflecting the timing (day of year) and magnitude (seasonality from 0–1) for that 1-year interval. Changes in the resultant vector reflect nonstationary behavior in the underlying input data. Further, any changes can be differentiated as being due to timing or seasonality through the angle of the resultant vector (cf. `RV`, Fig. 3) or its length (cf. `SMag`, Fig. 3).

Fig. 6 shows four time traces revealing that in terms of timing both NEE and precipitation at WLEF appear to be trend stationary, showing only what appears to be white noise variance. NEE and precipitation seasonality on the other hand have a particularly non-random nonstationarity that recurs on fairly regular 4-year cycles, which could be due to a regional interannual driver (e.g. El Niño). While this analysis did not reveal a trend stationary result, which would indicate some deterministic processes, it is nonetheless useful and suggests a higher degree of autocorrelation in the data with respect to seasonality than with regard to timing.

While the data reduction and computational savings potential of the polar metrics technique is particularly attractive, so far we have not discussed the differences in representativeness between polar metrics and their original time series. While we cannot comprehensively deal with this topic here, we

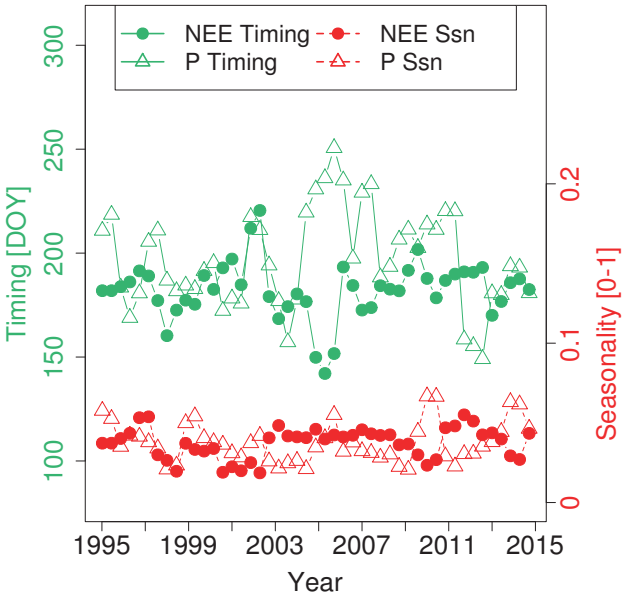

Fig. 6. Sliding window analysis of the magnitude (seasonality) and timing of the resultant vector (`RVec`) for NEE and precipitation. Notice that there does not appear to be a clear trend in the magnitude of NEE or precipitation at WLEF, but there do appear to be fairly regular 3–4 year excursions. Excursions in seasonality are only sometimes coincident with excursions in timing of the average vector.

have included results from a simulated representativeness experiment. As one test of how well polar metrics capture variation in data as compared to the original time series, we used an unsupervised stratified sampling algorithm (k-means) to differentiate the degree to which time series or polar metrics capture dominant axes of variability. An optimized measurement system should allow for partitioning of its data into groups that are highly dissimilar, but at the same time containing groups within which similarity is high. One way to measure this is the ratio of the total within-cluster sum of squares (WSS) divided by the total sum of squares (TSS). Low within-cluster dissimilarity indicates ideal representativeness, where the average variance within the cluster groups (WSS) is very small relative to the global variance. Contrasting time series cluster results against polar metric results across a range of cluster divisions should reveal insight into how robustly these two data sets express change. However, the results are only significant insofar as the input data are considered relevant.

The MODIS NDVI data from Fig. 5 were re-used in this simulated experiment to generate time series and polar metric inputs for clustering. Rather than averaging the 289 MODIS pixels as before, each pixel and year was clustered based on its series of 52 weekly NDVI values. The time series input was reformed into a 4335 x 52 matrix representing 4335 pixel-years (289 pixels x 15 years). This generated an input matrix for clustering where each column of the matrix represented a

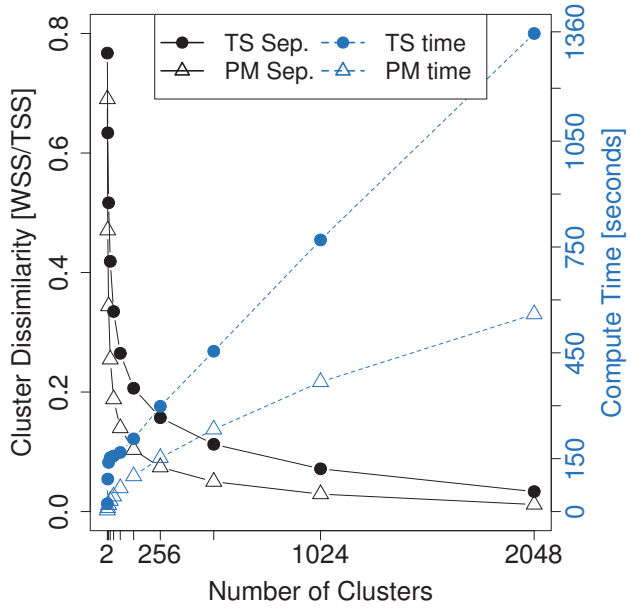

Fig. 7. Representativeness test. Ideal representativeness here is measured as low within-cluster dissimilarity (within-cluster sum of squares / total sum of squares). The k-means clustering algorithm, which minimizes variances, is used to evaluate time series and polar metric representation of the same NDVI data (289 MODIS pixels, 2000-2015). NDVI data represented through polar metrics form more distinct and coherent groups than time series at any given number of divisions, and the polar metric clustering takes less than half the time to complete.

weekly sampling point and each row a pixel-year of values. The polar metric input was generated by first passing the same time series values through the `calc_metrics` function, then reforming it to yield a matrix of 4335 x 11 values, also 4335 pixel-years, where each of the 11 columns was a polar metric (e.g., length of season) describing an aspect of the annual pattern. Thus the essential difference between the input data were that one described the annual pattern through weekly NDVI values, whereas the other described the annual pattern through polar metrics.

Fig. 7 shows that across separate trials that varied the number of cluster divisions from 2–2048, clustering polar metric values resulted in lower within-cluster dissimilarity relative to the total variance of the input. This indicates that cluster centroids from polar data were computationally easier to group into representative clusters. While polar metrics in this case were grouped with greater relative-accuracy than time series this could be due to the reduced dimensionality (11 vs. 52), which is not the same as the capacity of the 11 variables to measure changes that are of interest. Through this test alone we cannot determine if the better performance of polar metric clustering is due to transformation of the data into ideal measures or reduced dimensionality, but this could be explored in future work.

An additional caveat for the PolarMetrics technique is that missing values must be gap-filled. Seasonal metrics are calculated based on the accumulation of annual values, therefore any missing values will bias the seasonal statistics. In this paper missing weekly averages were imputed from an average annual series. Gap-filling weeks (rather than the original hourly values) is a simple approach for reducing error probabilities of imputed values, although more rigorous Bayesian approaches are also possible.

In this paper we introduced the PolarMetrics R package, which we developed as a customizable tool for accelerating the analysis of environmental time series. PolarMetrics is generalizable and simple enough to be used to analyze data over long, or unusual multi-annual cycles, as well as shorter diel cycles. For interannual cycles, naturally one would typically first deseasonalize the time series, after which all the same steps can be applied (the argument `spc` in `calc_metrics` should be set to the correct number of samples in each cycle of interest e.g., for a two year cycle of weekly data, `spc = 104`).

Acknowledgment

We thank the authors, developers, and maintainers of the R (R Development Core Team, 2011) open source statistical programming language and environment, as well as Jim Lemon for maintaining the plotrix package. For distributing data we thank the AmeriFlux Network and for funding of AmeriFlux data resources we thank the U.S. Department of Energy's Office of Science. We thank NASA LP-DAAC for making MODIS data freely available as well as ORNL DAAC for providing the MODIS Land Product Subsets web service.

References

[1] A. R. Desai, "Influence and predictive capacity of climate anomalies on daily to decadal extremes in canopy photosynthesis," *Photosynthesis Research*, vol. 119, no. 1, pp. 31–47, Feb 2014.

[2] R. Burgan, C. Hardy, D. Ohlen, G. Fosnight, and R. Treder, "Ground sample data for the conterminous u.s. land cover characteristics database," USDA Forest Service, Rocky Mountain Research Station, MA, Tech. Rep. General Technical Report RMRS-GTR-41, Nov. 1999.

[3] L. P. C. Morellato, L. Alberti, and I. L. Hudson, *Applications of Circular Statistics in Plant Phenology: a Case Studies Approach*. Dordrecht: Springer Netherlands, 2010, pp. 339–359. [Online]. Available: https://doi.org/10.1007/978-90-481-3335-2_16

[4] I. Melendez-Pastor, J. Navarro-Pedreo, M. Koch, I. Gmez, and E. I. Hernndez, "Land-cover phenologies and their relation to climatic variables in an anthropogenically impacted mediterranean coastal area," *Remote Sensing*, vol. 2, no. 3, pp. 697–716, 2010. [Online]. Available: http://www.mdpi.com/2072-4292/2/3/697

[5] J. A. Long and P. C. Stoy, "Peak tornado activity is occurring earlier in the heart of tornado alley," *Geophysical Research Letters*, vol. 41, no. 17, pp. 6259–6264, 2014, 2014GL061385. [Online]. Available: http://dx.doi.org/10.1002/2014GL061385

[6] A. R. Desai, K. Xu, H. Tian, P. Weishampel, J. Thom, D. Baumann, A. E. Andrews, B. D. Cook, J. Y. King, and R. Kolka, "Landscape-level terrestrial methane flux observed from a very tall tower," *Agricultural and Forest Meteorology*, vol. 201, pp. 61 – 75, 2015. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0168192314002688

[7] S. L. Tuck, H. R. Phillips, R. E. Hintzen, J. P. Scharlemann, A. Purvis, and L. N. Hudson, "Modistools – downloading and processing modis remotely sensed data in r," *Ecology and Evolution*, vol. 4, no. 24, pp. 4658–4668, 2014. [Online]. Available: http://dx.doi.org/10.1002/ece3.1273