Scalable Algorithms for Unsupervised Classification and Anomaly Detection in Large Geospatiotemporal Datasets

Richard Tran Mills (Intel Corporation, Hillsboro, OR), Jitendra Kumar (Oak Ridge National Laboratory, Oak Ridge, TN), Jon Weiner (University of California, Berkeley, CA), and Forrest M. Hoffman (Oak Ridge National Laboratory, Oak Ridge, TN)

1. Introduction

- The increasing availability of high-resolution geospatiotemporal data sets from sources such as observatory networks, remote sensing platforms, and computational Earth system models has opened new possibilities for knowledge discovery and mining of ecological data sets fused from disparate sources.
- Traditional algorithms and computing platforms are impractical for the analysis and synthesis of data sets of this size; however, new algorithmic approaches that can effectively utilize the complex memory hierarchies and the extremely high levels of available parallelism in state-of-the-art high-performance computing platforms can enable such analysis.
- We examine some of these approaches and a few practical applications to the analysis of climatic, remotely-sensed vegetation phenology, and LiDAR data sets and speculate on some of the other applications that such scalable analysis methods may enable.

2. Accelerated k-means Clustering

- We have two implementations of accelerated k-means clustering, following two parallel programming models
  - A master-worker (MW) model: Central master assigns "aliquots" of work to workers. Facilitates dynamic load balancing but has memory and performance scalability limits due to single, central process.
  - Fully distributed (FD): All processes use static distribution of work. Very scalable, but no dynamic load balancing.
- We "accelerate" the k-means process using two techniques described by Phillips (doi:10.1109/IGARSS.2002.1026202):
  - Use the triangle inequality to eliminate unnecessary point-to-centroid distance computations based on the previous cluster assignments and the new inter-centroid distances.
  - Reduce evaluation overhead by sorting inter-centroid distances so that new candidate centroids c_j are evaluated in order of their distance from the former centroid c_i. Once the critical distance $2d(p, c_i)$ is surpassed, no additional evaluations are needed, as the nearest centroid is known from a previous evaluation.

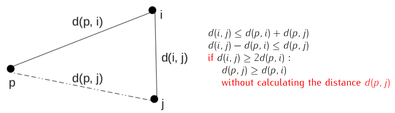


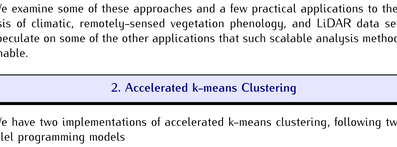
Figure 1: *The triangle inequality is used to eliminate unnecessary distance calculations.*

- We also improve cluster quality by moving or "warping" clusters that become empty to locations in data space where points that are farthest from their current cluster centroids reside.

2.1 Parallel Performance

2.1.1 Accelerated k-means code

- In 2011, we would use ∼1024 AMD Opteron cores on a machine like Jaguar, the Cray XT5 at ORNL, for our analyses.
- In 2015, we can do larger analyses on a single compute node of Intel's Endeavor cluster with Intel® Xeon® E7-8890 v3 ("Haswell-EX") processors.
  - AVX2 instruction set: 256-bit (8 single precision `float`s) vector registers with dual-issue fused multiply-add
  - Four 18 core (36 thread) CPUs; over 500 GB DRAM



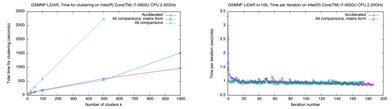
(a) Jaguar (b) Endeavor

Figure 2: *Times to cluster different versions of the 2000–2009 ForWarn phenology data set on (a) 1024 cores of the Jaguar Cray XT5, ca. 2011 at ORNL and (b) a single 72-core "Haswell-EX" node on Intel's Endeavor cluster. The data set used on Jaguar is the 16 day product, while the one on Endeavor is the 8 day product and is therefore twice as large (251 GB in single precision).*

- With acceleration, an equal distribution of observation vectors among processes does not guarantee load balance. Figure 2b illustrates the benefit of using smaller aliquots to enable dynamic load balancing in the MW clustering code.

2.1.2 Improving computational intensity

- We have recently realized that it is possible to achieve greater computational intensity of the observation–centroid distance calculations by expressing the calculation in matrix form:
  - For observation vector x_i and centroid vector z_j, the squared distance between them is $D_{ij} = \|x_i - z_j\|^2$.
  - Via binomial expansion, $D_{ij} = \|x_i\|^2 + \|z_j\|^2 - 2x_i \cdot z_j$
  - The matrix of squared distances can thus be expressed as $D = \bar{x}\mathbf{1}^\mathsf{T} + \mathbf{1}\bar{z}^\mathsf{T} - 2X^\mathsf{T}Z$, where X and Z are matrices of observations and centroids, respectively, stored in columns, $\bar{x}$ and $\bar{z}$ are vectors of the sum of squares of the columns of X and Z, and $\mathbf{1}$ is a vector of all 1s.
- The above expression for D can be calculated in terms of a level-3 BLAS operation (xGEMM), followed by two rank-one updates (xGER, a level-2 operation).
- Level 2 and 3 BLAS operations admit very computationally efficient implementations, and libraries such as Intel® MKL provide highly optimized versions.
- We have experimented with using the above, matrix formulation for the distance calculations and have found that it is dramatically faster than the straightforward loop over vector distance calculations when many distance comparisons must be made.
- Using the matrix formulation for distance comparisons in early k-means iterations is straightforward; a more complicated approach we will explore is using the matrix formulation in combination with the acceleration techniques described above, in which only a subset of observation–centroid distances are calculated.



(a) (b)

Figure 3: *Timings for clustering the GSMNP LiDAR dataset from section 2.3 using a single worker process on an Intel® Core™ i7-5650U CPU operating at 2.20GHz. (a) Total timings for k-means clustering using the acceleration techniques; doing all distance comparisons but forming the distance matrix using BLAS operations provided by Intel® MKL; and doing all distance comparisons without the benefit of the matrix formulation and BLAS. (b) Timings per iteration for k=100 when using the acceleration technique compared to the matrix formulation for the distance calculations. In early iterations, where many distance comparisons are required, the matrix formulation offers better performance.*

2.2 Applications: Quantitative Ecoregionalization and Change Detection

(a) (b)

Figure 4: *Geospatiotemporal clustering of a combination of observational data and downscaled general circulation model results projects dramatic shifts in location of Alaska ecoregions using downscaled 4 km GCM results. Arctic tundra projected to be at 0.78% of current extent by 2099. DOI: 10.1007/s10980-013-9902-0.*

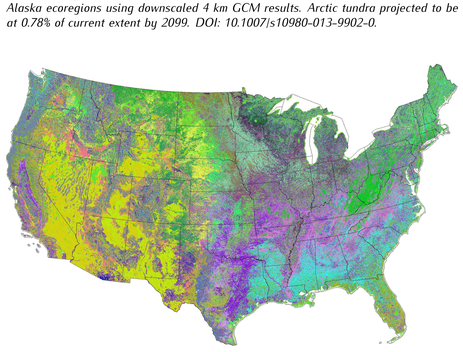


Figure 5: *A map of "phenoregion" assignments for the year 2012, based on k-means analysis with $k = 50$ of the entire MODIS-derived ForWarn NDVI product for years 2000–2012. The body of observation vectors being clustered consists of the year-long NDVI time series for every map pixel, for each year. The map indicates cluster membership (in random colors) for the phenology observed in 2012 at each map pixel.*

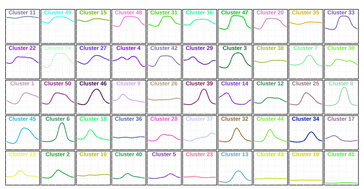
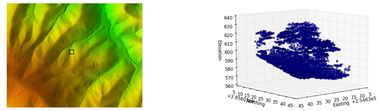


Figure 6: *The fifty centroids (corresponding to "phenoregion" prototypes) used for the membership assignments in the map in Figure 5. The colors the centroid plot correspond to the map colors.*

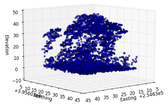
2.3 Applications: Classification of Vegetation Canopy Structure using LiDAR

- Airborne Light Detection and Ranging (LiDAR) enables large scale remote sensing of topography, built infrastructure, and vegetation structure.
- Multiple laser "returns" produce "point clouds" used to map the ground surface, buildings, roads, and utility infrastructure, and to reconstruct the structure of vegetation canopies.
- Large data volumes pose significant computational challenges to employing LiDAR to monitor and manage forests and animal habitats.

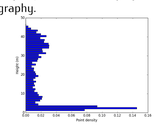


a) 3-D LiDAR point cloud extent at 30×30 m (black square) shown in a typical GSMNP cove forest.

b) Raw LiDAR point cloud (3,985 points), showing imprints of underlying topography.



c) LiDAR point cloud after topographic detrending and filtering (3,936 points).



d) Vertical distribution of LiDAR point density in a cove forest dominated by tall trees and a dense understory.

Figure 7: *Shown here are the steps involved in converting a LiDAR point cloud into a vertical vegetation canopy distribution for subsequent cluster analysis.*

Figure 8: *This map shows the 30 most-different classes of vegetation canopy structure, randomly colored, as identified by k-means clustering for the Tennessee portion of the Great Smoky Mountains National Park.*

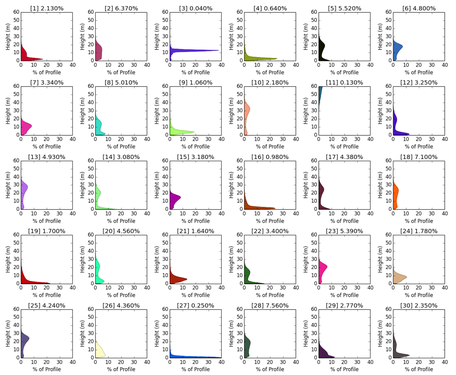
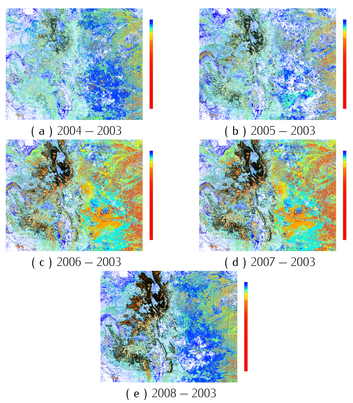


Figure 9: *The 30 centroids represent vegetation canopy structure prototypes. Cluster 3, covering a small 0.04% of the area, likely represents bad data. Cluster 11, covering 0.13% of the area, represents objects above the tallest trees in the Park (e.g., birds, bugs, particles, aerosols).*



(a) 2004 − 2003 (b) 2005 − 2003

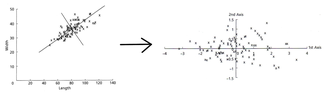
(c) 2006 − 2003 (d) 2007 − 2003

(e) 2008 − 2003

Figure 10: *Maps showing the relative state space transition distances (how different phenoregion assignments are for given years) between years in Colorado and southern Wyoming. Pine beetle mortality correlates strongly with high transition distances. Black-outlined polygons are disturbed areas indicated on aerial sketch maps.*

3. Principal Components Analysis

Principal Components Analysis (PCA) determines, for a p-dimensional data set, an orthogonal set of p new axes (linear combinations of the original p variables) such that the first axis explains the greatest variance, the second explains the next most variance, and so on:



- Commonly used to determine dominant patterns in data
- But can also be used to determine the anomalous patterns: Observations that score strongly on low order components do not follow the correlation structure of the data.

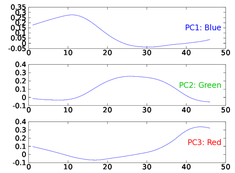


Figure 11: *The loadings (coefficients in the linear combination of the 46 original variables) along the three varimax-rotated principal axes. The x-axis corresponds to the eight-day NDVI-acquisition windows and loadings are shown on the y-axis.*

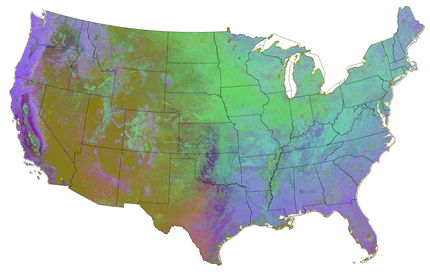


Figure 12: *Phenoregion assignment map for year 2000 with $k = 1000$. Similarity colors are used to indicate cluster membership.*

3.1 Parallel Principal Components Analysis Tool

- We have developed a prototype parallel tool to perform PCA.
- Rather than explicitly forming the covariance matrix, computes thin SVD of the adjusted data matrix.
- Uses the Lawson-Hanson-Chan factorization to exploit the "tall and skinny" ($m >> n$) nature of our matrices: ($m >> n$)
  - Form reduced factorization A = QR (via parallel PLAPACK routine)
  - Gather the matrix R to process 0.
  - Process 0 calls LAPACK DGESVD to compute the SVD R = USV^T.
  - Optionally, back transform Q to get Q ← QU.
  - Final SVD is: A = QSV^T
- A serial bottleneck exists where the SVD of R is computed, but this matrix is so small (only 46×46 for our NDVI data set) that this serial portion is essentially negligible.

3.2 Detecting anomalous observations with PCA

- Can identify anomalies two complementary ways:
- Look at sum of scores onto r lowest-order components: $\sum_{i=p-r+1}^{p} \frac{y_i^2}{\lambda_i}$ greater than some outlier threshold
- Look at squared prediction error: How well an observation can be represented in subspace of q highest order components?
  - Idea: decompose into modeled and residual parts: $x = \hat{x} + \tilde{x}$
  - $P = [v_1\ v_2\ \ldots\ v_q]$
  - $\hat{x} = PP^T x = Cx$ and $\tilde{x} = (I - PP^T)x = \tilde{C}x$
  - Abnormal if SPE $= \|\tilde{x}\|^2 = \|\tilde{C}x\|^2$ exceeds threshold
- Can also do cross-comparison: Construct subspace from one data set, then see how well observations from another can be represented in that space.

3.3 Detecting anomalies within a single year, single NEON domain

- These approaches will flag any observations that are somehow *"unusual" for the collection of data from which the principal components have been calculated.*
- Some judgement required: choice of NDVI data subset used in the PCA calculation will affect what constitutes a "normal" or "abnormal" observation.
- E.g., Extremely low NDVI may appear normal when using PCA based on national dataset due to presence of areas like the Mohave; appears anomalous when using PCA based only on humid Southeast.
- Here we use PCAs computed over single years and within a spatial domain conforming to the eco-climatic domains established by the National Ecological Observatory Network.

- In all examples, PC vectors 10–46 are used as the basis for the "abnormal" space, which explains 5–10% of the variance.
- In all of examples, certain features that are not disturbances but possess very anomalous NDVI traces (e.g., bodies of water) show up very strongly.

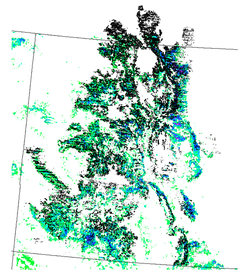


Figure 14: *Portion of the Southern Rockies–Colorado Plateau NEON Domain for year 2008, showing map cells scoring in the 85th percentile. Black paloggons show damaged areas noted in aerial detection surveys; extensive damage due to mountain pine beetle and sudden aspen decline are evident.*

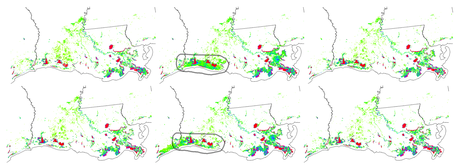


Figure 15: *Portions of the PCA-based anomaly maps (map cells scoring in the 90th percentile are shown) for the Southeast NEON Domain for years 2004–2009, showing the area in the vicinity of the Louisiana coast. From left to right, the top row shows years 2004, 2005, and 2006, respectively, and the bottom row years 2007, 2008, and 2009. The affected regions are circled in the 2005 and 2008 maps. The prominent red features are water bodies.*

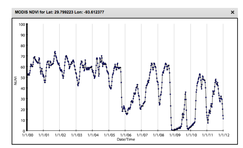


Figure 16: *NDVI trajectory as viewed via the Forest Change Assessment Viewer for a location (close to the center of the circled region in Figure 15) near the coast in southwestern Louisiana showing apparent hurricane-induced mortality from events in 2005 and 2008.*

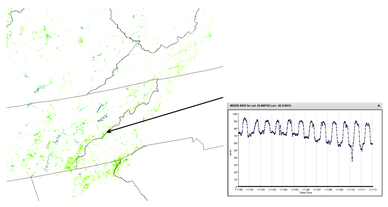


Figure 17: *At left, a portion of the PCA-based anomaly map (map cells scoring in the 90th percentile are shown) for the Southern Appalachians/Cumberland Plateau NEON Domain for year 2010. The arrow indicates a location thought to be affected by hemlock woolly adelgid, and the corresponding NDVI trajectory is shown at right.*

4. Notices and Disclaimers