Available online at www.sciencedirect.com

SciVerse ScienceDirect

Procedia Computer Science 18 (2013) 2347 – 2356

Procedia Computer Science

International Conference on Computational Science, ICCS 2013

Contrasting Climate Ensembles: A Model-Based Visualization Approach for Analyzing Extreme Events

Robert Sisneros[a], Jian Huang[b], George Ostrouchov[c], Sean Ahern[c], B. David Semeraro[a]

[a]*National Center for Supercomputing Applications, Urbana, IL 61801, USA*
[b]*University of Tennessee, Knoxville, TN 37996, USA*
[c]*Oak Ridge National Laboratory, Oak Ridge, TN 37831, USA*

Abstract

The use of increasingly sophisticated means to simulate and observe natural phenomena has led to the production of larger and more complex data. As the size and complexity of this data increases, the task of data analysis becomes more challenging. Determining complex relationships among variables requires new algorithm development. Addressing the challenge of handling large data necessitates that algorithm implementations target high performance computing platforms. In this work we present a technique that allows a user to study the interactions among multiple variables in the same spatial extents as the underlying data. The technique is implemented in an existing parallel analysis and visualization framework in order that it be applicable to the largest datasets. The foundation of our approach is to classify data points via inclusion in, or distance to, multivariate representations of relationships among a subset of the variables of a dataset. We abstract the space in which inclusion is calculated and through various space transformations we alleviate the necessity to consider variables' scales and distributions when making comparisons. We apply this approach to the problem of highlighting variations in climate model ensembles.

Keywords: visualization; climate ensembles; multivariate classification

1. Introduction

Analysis of scientific data often involves the study of relationships among multiple variables. One approach to multivariate analysis is to form a hypothesis about a relationship, define it mathematically, and measure the distance of the data from that mathematical model. In this context, a "relationship" is any provable association, combination, or co-occurrence pattern between variable values. In relational databases, such an interaction is formally identifiable as a set where collections of predicates are true. The onerous task of building hypothetical, predictive systems of co-occurrence patterns is iterative by nature; refinement comes through stages of discovery and verification. To this end, we have developed an interactive tool for accelerating multiple steps of this process that is applicable to very large scientific datasets.

We propose a flexible technique in Section 3 to classify data into attribute-based sets according to user-defined *models*, or representations of a relationships among some *n* variables as a set of points in that *n*-dimensional variable space (not to be confused with a computational model, which we refer to as a simulation). The crux of

[*]Corresponding author. Tel.: +1-217-244-2741.
E-mail address: sisneros@illinois.edu.

doi:10.1016/j.procs.2013.05.406

our technique comes from a recognition that major differentiating factors among statistical methods for analyzing multivariate data are space transformations and distance metrics. The flexibility then arises from the fact that in our system several such transformations and metrics are implemented, and in any one pass of classification they may be customized and switched among on demand. In practice, we find the ability to switch among several transformations to be impactful when attempting to understand a complex specification.

In a single pass, each spatial location is classified by one of the user-defined models. This classification is accomplished by selecting the model that minimizes a distance function at a location. We will show how this execution model alone is quite useful for feature extraction, validation and analysis. Through changes in space and distance we are able to maximize exploratory power, to properly handle special cases, and to lower dependence on significant a priori knowledge about the data. However, a method must leverage available domain knowledge to realize full potential as a discovery and validation tool and we accomplish this with our two pass approach.

This technique has been integrated into a scalable analysis and visualization tool, VisIt [34]. The VisIt framework is an implementation of an analysis pipeline in which data passes through several filters that transform and combine data in different ways. Imbedding the new technique in this framework allows users to utilize other components to handle tasks such as thresholding missing data. Our implementation leverages VisIt's efficient parallel processing methods and is therefore applicable to very large data.

As a defining example of our intended use case: climate scientists are interested in studying the CO_2 exchange between hemispheres. Recent analysis in this domain was accomplished by tracking flow across hemispheres and looking for anomalies. In our approach, the first pass (region specification) classifies the data into two regions that correspond to the northern and southern hemispheres. In the model specification pass, models are specified to classify only within specific regions. For instance, how near is CO_2 in the Northern hemisphere to the average value. This is a simplified example of our general use case: domain knowledge is wrapped into regions, and verification/discovery classifications via models take place within these regions.

However, classifications and statistics may take place within *any combination* of regions allowing for the direct study of interactions among regions. We will outline a use case and replicate a previous result as a validation of our approach (Section 4.1). In Section 4.2 we follow with an ensemble study of this climate data. Pinpointing areas for simulation refinement through contrasting ensembles is a necessary and important concentration for many domain scientists. We will demonstrate the utility of our approach by building analyses that are increasingly complicated to reproduce with existing tools that also increasingly highlight ensemble variability. Furthermore, our calculated regions can be correlated with naturally occurring regions, e.g. ecoregions. In this case, doing so provides an intuitive context to aid in tuning a simulation.

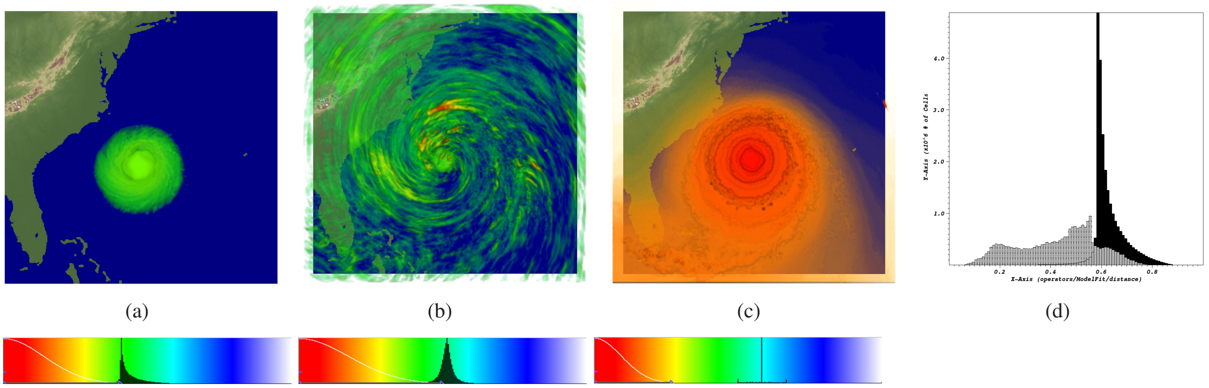
2. Background

2.1. Target Application

The coupling of modeling and simulation is indispensable in the scientific process of future climate projection, but offers significant challenges [1, 2]. For instance, the creation of an efficient and precise simulation is a project spanning multiple domains; the single task of verifying a completed code is daunting. Typically, ensemble runs are used in the necessary analysis it takes to evaluate, refine, verify, or even use a simulation. The utility of ensembles is immediate in that a simple average across ensembles tends to show improvements in precision [3]. There are also efforts focused on comparing and contrasting models both among each other and to observed events [4, 5]. Analysis of model variability yields more precise projections [6, 7] or probabilistic expected outcomes through uncertainty quantification [8, 9]. We believe a tool capable of pinpointing differences among models would be a useful addition to any of these analysis schemes.

2.2. Multivariate Visualization

Information Visualization: as surveyed by Wong and Bergeron [10], many successful techniques for multivariate information visualization start by informative overviewing and enable subsequent visual drill-down. Some of the more prominent approaches include parallel coordinates [11], creative design and use of shape, color and textures [12], dimension stacking [13], and hierarchical axes [14]. Recent work is directed toward innovative layouts of visual data abstractions, summarizing data based derived features, as well as data fusion [10, 15]. However,



(a) (b) (c) (d)

Fig. 1. Features from hurricane Isabel. Distance from the model of the eye in variable space [0, 1] (a). Distance from the movement model variable space (b) and probability space (c), both in range [0, 1]. The color bars show the distribution of each rendered variable along with the selection for rendering. Illustration of probability space (d): a variable before (black) and after (gray) transformation into probability space.

for domain users the most useful multivariate relationship visualization is often still the scatterplot. Improvements to these include brushing and linking [16, 17]. Scatterplots were applied to simulation data and extended to complex feature specification by Doleisch et al. [18]. Combining parallel-coordinate plots with scatterplots is also possible [19].

Scientific Visualization: as originally set up by Drebin et al. [20], multi-field volume visualization is used to display volumes classisfied to several different materials. In our case, we classify to several different models. The pivotal component is the transfer function. Scout [21] is likely the most versatile system with a programming language interface. Kniss et al. [22] proposed an approach where users can specify Gaussian functions for multi-field volume visualization. Finding the consistent existence of relationships is very beneficial, as exemplified by gradient based [23], curvature based [24], and importance-based [25] methods. There is also a recent trend to achieve interactive runtime data reduction by doing compound queries [26]. In a way, this is actually about relationships, just one at a time. Our approach provides sophisticated treatment of the problem space to enable many specified features.

Feature Extraction: in this work we propose a generalization of space and distance that can be generally applied to feature extraction. In [27] Maciejewski et al. used kernel density estimation to discover and cluster models in histogram space for the purpose of classifying features. Their approach can directly leverage all the generalization available under our framework and adopt the use of new spaces, transformations such as copula and new distance metrics for clustering. One can also use mathematical metrics based on information theory [28], wavelets [29], bio-diversity inspired data diversity metrics [30] and even morse smale topology [31] to characterize data importance and hence group features by similar importance.

2.3. *Clustering*

The classification step of our approach is similar to that in model-based clustering [32, 33], which is sometimes known as expectation-maximization (EM) clustering. The EM algorithm alternates between the E-step (conditional expectation) and the M-step (likelihood maximization). In the E step, cluster memberships are computed based on fixed models and then assigned to the model where they have the highest probability. In the M step, model parameters that maximize the expected likelihood given the cluster memberships are computed. There is no iterative optimization in our approach in that we focus on verifying the existence of relationships, i.e. a specified model may not be a good fit anywhere. Also, we represent models as sets of points in $\mathbb{V}$ space (variable space); in model-based clustering models are typically multivariate Gaussian.

3. The ModelFit System

In this section, we will first outline the foundation of our method (ModelFit), discuss and demonstrate its use, and provide examples of simple models from the IEEE Visualization 2004 contest dataset: the 500 × 500 × 100 hurricane Isabel simulation. We specify models from the following variables: pressure (P), temperature (TC), and precipitation (PRECIP) and use timesteps 24 and 25. It should be noted that parallelism in VisIt is based on the spatial, and not temporal, decomposition of data; the assumption is that analysis is of a single time step. While there is a mechanism to bring in variables from other time steps, we avoid this overhead by merging both used time steps into one. From VisIt's perspective this simply doubles the number of variables we are analyzing.

3.1. Models, Formally

Let $R_1, R_2, \ldots, R_k$ be a set of k models, each model being a set of points in $\mathbb{V}$. We begin with the simplest case: each model is only one point ($\mathbb{V}$ represents a single variable, and the value at a location is a single scalar value). Let this one point be $\mathbf{r}_j$ for model R_j. A location $\mathbf{x}$ is classified to model i if $|\mathbf{v_x} - \mathbf{r}_i|$ is minimized, that is, it is classified to the closest model. Next, if models contain more than one point, we need to introduce a link function L that reduces a model to one point that can be used in the distance calculation. Given an attribute vector $\mathbf{v} \in \mathbb{V}$ and a model R, the link function $L(\mathbf{v_x}, R)$ returns the point $\mathbf{r} \in R$ that is closest to $\mathbf{v_x}$. Generalizing to multipoint models by single linkage, a location $\mathbf{x}$ is classified to model i if $\left\|\mathbf{v_x} - L_1(\mathbf{v_x}, R_i)\right\|_2$ is minimized, that is, it is classified to the most closely linked model.

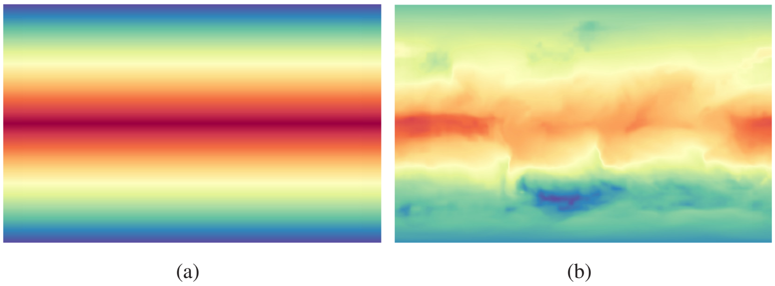
We finish by adding transformations of the $\mathbb{V}$ attribute space. Let f be a function (possibly nonlinear) that maps $\mathbb{V}$ into $\mathbb{R}^n$. That is, $\mathbf{v}' = f(\mathbf{v})$, where $\mathbf{v} \in \mathbb{V}$, $\mathbf{v}' \in \mathbb{R}^n$. The application of f to a set of points comprising a model is simply pointwise. By adding an attribute space transformation we have a final classification: a location $\mathbf{x}$ is classified to model i if $\left\|f(\mathbf{v_x}) - L_1(f(\mathbf{v_x}), f(R_i))\right\|_2$ is minimized. The point is classified to the model that is most closely linked in the transformed attribute space. This expression shows that attribute space transformations are applied first, then model linkage is considered, and finally a distance metric is applied.

3.2. Using ModelFit

Our system is implemented as a data-parallel operator called ModelFit in VisIt, with an early version available as of release 2.5. VisIt supports a number of visualization algorithms including pseudocolor and contour plots as well as volume renderings. Before visualization, *operators* may be applied to filter data or derive values. In VisIt, one can apply any of several operators to a data visualization. For example, application of the slice operator to a pseudocolor plot confines the pseudocolor visualization to the slice plane defined in the slice operator. VisIt presents the user with an interface for each operator that is used to specify the parameters of the operation, such as position or orientation for the slice plane for example. The ModelFit operator presents the user with an interface for the specification of models. This is accomplished by adding variables to define the variable space, $\mathbb{V}$. Then, the points representing some relationship are added to the model. One might wish to specify a model that displays where temperature and pressure are both near max or min values. The "and" in this example takes the model from one- to two-dimensional. The "or" makes this a two point model. The strength of the system comes from the fact that a user is expected to specify several such models. The result is then a summary of all models in that at any spatial location of the data, best fitting model classifies that point.

So far, the description for specifying a model is identical for specifying a region. The difference between models and regions is that for regions, classification and statistics calculations are always over all data. Although this may hold true for a particular model, models may be configured to classify only a subset of data. This subset for a model is defined as a union of regions. The approach takes place in two passes, all data is classified into regions, and then separate classifications are performed inside regions. We envision regions as areas corresponding to known features, i.e. places to impart domain knowledge. Furthermore, regions for statistics calculations need not match those for classification. For example, a model may classify a point in *Region A* based on its distance from the average of *Region B*. Our operator creates three accessible variables that may be used directly for plotting in VisIt or as input to other operators, the index of the classifying region, the index of the model, and the actual distance between the nearest model which is normalized to the range of $[0, 1]$.

Figure 1(a) is the result of classifying the data with a one point, two-dimensional model to extract the eye of a hurricane. The core of a hurricane contains both extremely low pressure as well as little precipitation relative to the rest of the storm; hence our specification is of minimum values for both variables. In the case that only a single model is specified, it will vacuously be the best fit at every location of the data. This image is therefore a volume rendering of the distance variable, where the best fits of the model have been selected with VisIt's transfer function widget.



(a) (b)

Fig. 2. Example of space changes with 2D models. Requesting average values from both latitude and ocean CO_2 emissions before (a) and after (b) transformation into 0-1 space.

3.3. Spaces

Potentially, each attribute has a different physical meaning and different units. In some cases, domain experts may prefer to specify models, such as known relationships or thresholds, in the original attribute units. In this case no transformation is needed. However, users are often interested in models that relate two main types of events: typical events and extreme events. To facilitate specifying such models, we provide methods to transform the attribute data values to a canonical form where such navigation is easier without strong domain knowledge. In this section, we detail transformations implemented in ModelFit.

3.3.1. Commodity Transformations

To produce a unit hypercube from the variable space, we use the following linear transformation:

$$f(\mathbb{V}) = \left(\frac{v_1 - \min(\mathbf{v}_1)}{\max(\mathbf{v}_1) - \min(\mathbf{v}_1)}, \ldots, \frac{v_n - \min(\mathbf{v}_n)}{\max(\mathbf{v}_n) - \min(\mathbf{v}_n)} \right) \tag{1}$$

Such a transformation allows makes it possible to navigate a variable space composed of variables with different scales. Figure 2 is an example with the climate dataset described in Section 4. In this example we have created a simple two variable model where the variables have different scales: CO_2 and latitude. Figure 2(a) shows that the latitude variable dominates the model's distance calculation. With the "0-1" space transformation applied (Figure 2(b)), each variable now has the same scale, and each contributes to the final distance calculation. We have also implemented a simple transform of taking the log of the data.

3.3.2. Copula (Probability) Space

Intuitive linear transformations do not provide a direct way to distinguish anomaly data from the rest of the dataset and hence may obscure true multivariate patterns of data. In this important use case, we consider probability space transformation – a nonlinear transformation that changes each attribute's units to probability within its empirical distribution function. This transformation was originally described by [35] as a canonical way to describe dependence. In fact, if the attributes were fully independent, they would now jointly have a multivariate uniform distribution. This is compelling for visualization because it makes a more uniform transfer function and, for our models, provides a way to access multivariate relationships in attribute extremes.

Figure 1(d) shows the distribution of the distance variable after classifying data by the same model in variable space (black) and probability space (gray). This illustrates how probability space is useful for analysis: values are more evenly spread throughout space facilitating selections of extreme events. More formally, if we consider the attributes to have a distribution function $H(\mathbf{v})$ over the attribute space $\mathbb{V}$, the copula C is defined by

$$H(\mathbf{v}) = H(v_1, v_2, \ldots, v_n) = C(H_1(v_1), H_2(v_2), \ldots, H_n(v_n)) \tag{2}$$

where $H_i(v_i)$ is the ith marginal distribution of H [36]. A simple empirical estimate of a distribution function for an attribute $\mathbf{v}_i$ from the ith column of $\mathbf{V}$, $(v_{i1}, v_{i2}, \ldots, v_{im})$, is

$$\hat{H}_i(v) = \frac{1}{m} \sum_{j=1}^{m} \delta(v_{ij} \leq v) \tag{3}$$

where $\delta(\cdot)$ is the Dirac delta function. This corresponds to simply dividing the sorted rank by the total number of values. Other estimators are possible [36] but we have found that values from simulation models do not contain prohibitive noise, and large data sizes work well with simple nonparametric estimators such as this one. The transformed attributes take values on the unit hypercube as they represent probability.

3.3.3. A Probability Space Example

As stated, a hurricane has very low pressure at its eye. Also, hurricanes require the water below their formations to be of a sufficiently high temperature, and as a hurricane passes over an area this temperature is reduced. Each of these phenomena are related to the way in which a hurricane moves across time. In an attempt to capture this movement, we have specified the following four variable model (or two variable, two time step model): at time step 24 pressure is near its minimum and temperature is near its maximum, and at time step 25 pressure is still near its minimum while temperature is now near its minimum. This model best fits a voxel where temperature switches from one extreme to another, and is therefore very unlikely to be a good fit anywhere in the data.

Figures 1(b) and (c) are the resulting images after classification by this model in both variable and probability space. Indeed, the model is exclusively a poor fitting one, but Figure 1(b) shows the areas in which the distance is relatively small. Interestingly, after transformation into probability space (Figure 1(c)), the distances are spread out such that a selection of less of the range, which can be seen by comparing transfer functions, results in not only an overall larger representation of the data, but one that indicates movement of/within the storm. Furthermore, the eye is especially highlighted.

3.3.4. Input vs. Classification Space

In our previous examples, mentions of transformations refer exclusively to those that change the space in which distances are calculated. However, we can also transform the input points of the model. For example, a value of “0” with the input space changed to the unit hypercube transformation refers to the minimum value for a variable. This is especially useful for analyses using a single model. Also, when a range around an extreme is desired, it is easy to specify in that we allow direct input of a range of values, rather than just a point. In addition, without any knowledge of the data, probability space input brings forth interesting use cases in that quartiles, or the like, may be requested in a specification.

4. Results

In this section we validate and demonstrate the utility of ModelFit on the NASA Goddard GEOS-5 general climate simulation. The spatial resolution of this data is $1^0 \times 1.25^0$ with 72 vertical levels. There are 5,840 time steps over two years (2000 and 2001); we use the January average for 2001 in Section 4.1 and the yearly average of 2000 in Section 4.2. We deal with five CO_2 variables: CO_2 biofuel burning emissions, total CO_2 from the CASAh simulation, total CO_2 from the CASAm simulation, CO_2 fossil fuel emissions, and CO_2 ocean emissions.

4.1. Validation

Kendall et al. [37] used a domain traversal particle advection approach to find areas of inter-hemisphere exchange of CO_2 in the GEOS-5 data. The resulting area of exchange is shown in Figure 3(a). Finding such an exchange between hemispheres was our inspiration in creating the *Regions* pass of ModelFit. When models are specified to classify different subsets of data in probability space, each subset is very likely to have different a probability distribution. Therefore visualizations will tend to show discontinuity across region boundaries. In Figures 3(b) and 3(c), the models have been specified in just this way – there are two models, the first classifies region 1 (the Southern hemisphere) and the second classifies region 2 (the Northern hemisphere). The point for each model is the median of the region it is *not* classifying (medians are specified as 0.5 when input in probability space). The idea is that by attempting to model similarities between regions, we are representing a more general interaction between them, a superset of direct exchange. For the following results, since only one model is classifying each half of the data, we are viewing ModelFit’s distance variable. As mentioned, this variable is in the range [0, 1]; the colormap is a typical hot/cold map with red mapped to value “0” (best fits) and blue to “1” (worst fits).

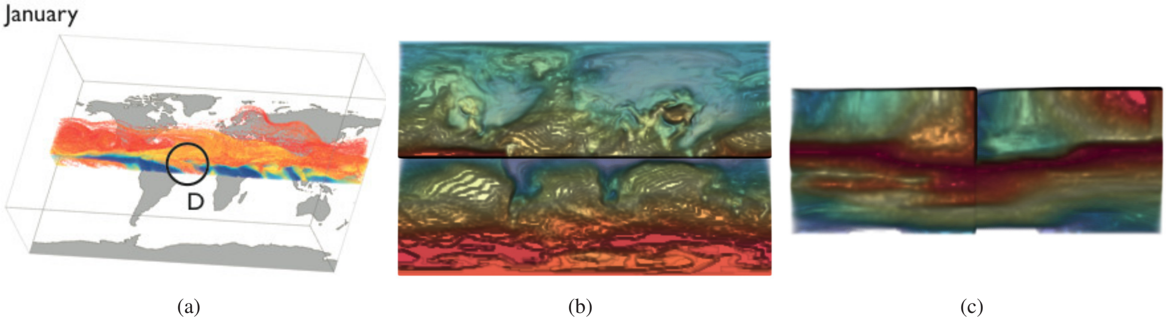


Fig. 3. A result from *DStep* [37] on the GEOS-5 data (a). The circled region corresponds to an area of CO_2 inter-hemisphere exchange. Initial distance rendering attempting to find areas of inter-hemispheric CO_2 exchange with ModelFit – front (b) and side (c) view.

We hypothesize that when comparing neighboring regions, areas where a transition between them is smooth represents a relationship *across* the boundary. In Figure 3(b), however, there appear to be no such regions. When viewed from the side (Figure 3(c)), it becomes clear that there is a smooth transition across the boundary through the vertical center of the data. Figure 4 is the result of thresholding all locations but the five percent with smallest distances (the color scale has been adjusted to span this new range). Upon close inspection, there is only one small location in which a value spans the boundary. In Figure 4(b) we have circled this region and have added an overlay of Africa and South America for reference. Interestingly, this region corresponds exactly to that in Figure 3(a). The exchange across boundaries is an excellent use case for this approach and illustrates the power of regions in an analysis setting; formulations of what corresponds to interesting areas outside of those across borders are still needed.

4.2. Utility

In this section we highlight differences among ensemble runs of the GEOS-5 climate data. To begin, we stitch all eight ensembles together into a single dataset (two rows of four ensembles). The typical use for ModelFit would be to treat variables from different ensembles uniquely and view a summary of ensembles in one setting. However, with eight ensembles, this arrangement allows for easy comparison and provides a natural and intuitive example use of regions. Also, each ensemble is averaged annually to facilitate detection of inner-annual variabilities. We will create four increasingly complex sets of models and regions to illustrate the different ways in which our approach may be leveraged for analysis. In all of the following results, the volume rendering transfer function widget is used to select only five percent of total distance values that correspond to best/worst fits.

It is typical when differentiating ensembles to compare extreme values. Figure 5 is a simple use of our system meant to represent a typical approach for exposing extreme values. In this image, there is a single, one variable and one point model in which the point's target value is the ocean's CO_2 emission average over all data, which

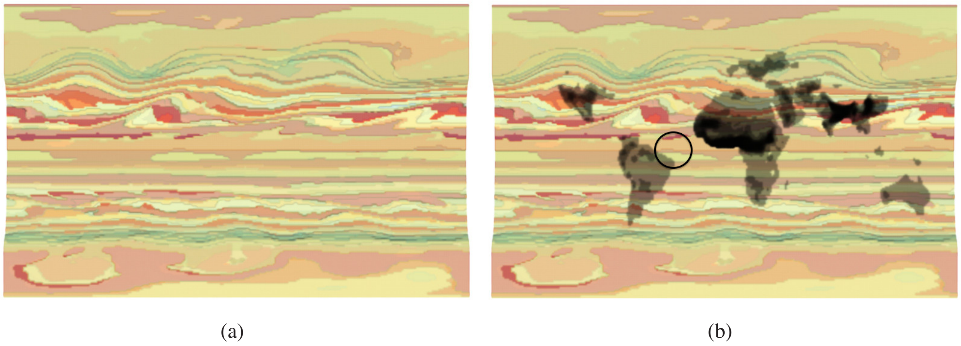


Fig. 4. Focus on best fits of inter-hemeshperic CO_2 exchange (a), with some context (b).

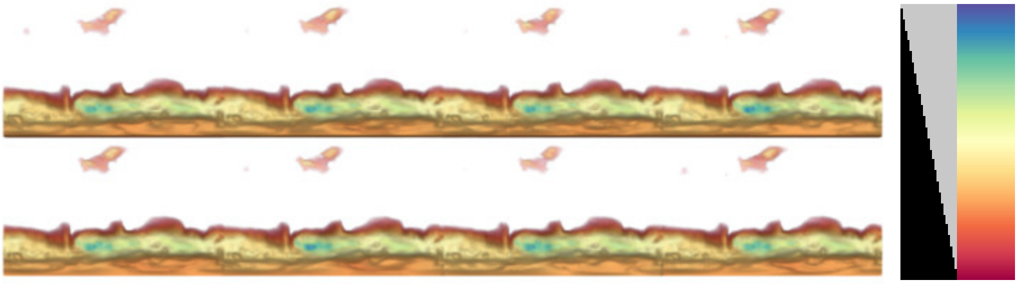


Fig. 5. Illustration of difficulties in highlighting differences among ensembles. We are highlighting extreme points, i.e. those farthest from the average CO_2 emissions. The rendering is of ModelFit's distance, [.95, 1]

in this case is all ensembles. Even when targeting extreme values, nearly all data is near the average; we have set the color and opacity variable as ModelFit's distance with the range representing the worst fitting locations as the farthest from the average are the extremes. There are noticeable differences among ensembles. However, the general similarity signifies the difficulty in finding such differences.

In Figure 6, we show how probability space can be used to further highlight ensemble differences. In the first row of Figure 6 we still only use a single, one variable model and the variable is still the ocean's CO_2 emissions. This time, we have a two point model and specify extremes directly, one point is the minimum value and the other is the maximum. Since we specify extremes directly, we are now focusing on only the smallest distances. There are now many visible differences among all ensembles. Although many areas are highlighted, there are still no easily recognizable features (such specific geographic areas). Therefore, for the second row of Figure 6 we have now specified five, single variable and two point models. While the threshold is identical to the previous result, the color variable is now ModelFit's model, with the five colors corresponding to the five specified models. The five models and variables are as follows: CO_2 biofuel burning emissions (gray), total CO_2 from the CASAh simulation (light orange), total CO_2 from the CASAm simulation (dark orange), CO_2 fossil fuel emissions (brown), and CO_2 ocean emissions (blue). In this image, there are still areas of ensemble differentiation, but there are also clear geographic features, i.e., there is a better context.

Finally, in the last row of Figure 6 we display a complicated series of models and regions for differentiating the ensemble runs telegraphing ModelFit's future uses. For this image, we use three of the CO_2 variables: biofuel burning (BB), fossil fuel (FF), and ocean emissions (OCN). We have created six regions: BB and FF are both near minimum values, BB and OCN are both near minimum values, FF and OCN are both near minimum values, and a similar three regions for maximum values. We then specify two single point models per region that target average values within the region, but of variables separate from those used to create the region: CASAm and CASAh. In addition, each of the 12 models classifies the entire dataset. So, for region 1 (BB and FF are both near minimum), we have model 1 and model 2. For model 1, we target the average value of CASAh and for model 2 we target the average value for the CASAm. As mentioned, each model is in contention to classify the entire dataset, but the "average" that is targeted is the average value of the variable only within a specific region. The 12 models alternate starting with CASAh (gray). With this being a direct comparison of two simulations, any contiguous regions (notably the blue and orange) correspond to target areas for the analysis of simulation precision.

5. Conclusions and Future Work

We have presented a two pass classification approach for facilitated and direct analysis of features that are otherwise difficult to express. There are several avenues we intend to explore after a user base is comfortable with this tool; each adds a level of complexity and generality to the approach. First, we exclusively use single linkage to calculate distances from points to models. Determining whether our approach could benefit from higher linkage, or whether linkage should be abstracted in the same way as space and distance could prove to be useful.

Similarly, there are many other possibilities for distance metrics. We believe it would be interesting to calculate metrics among regions, in addition to points. One such metric would be the Hellinger distance which quantifies

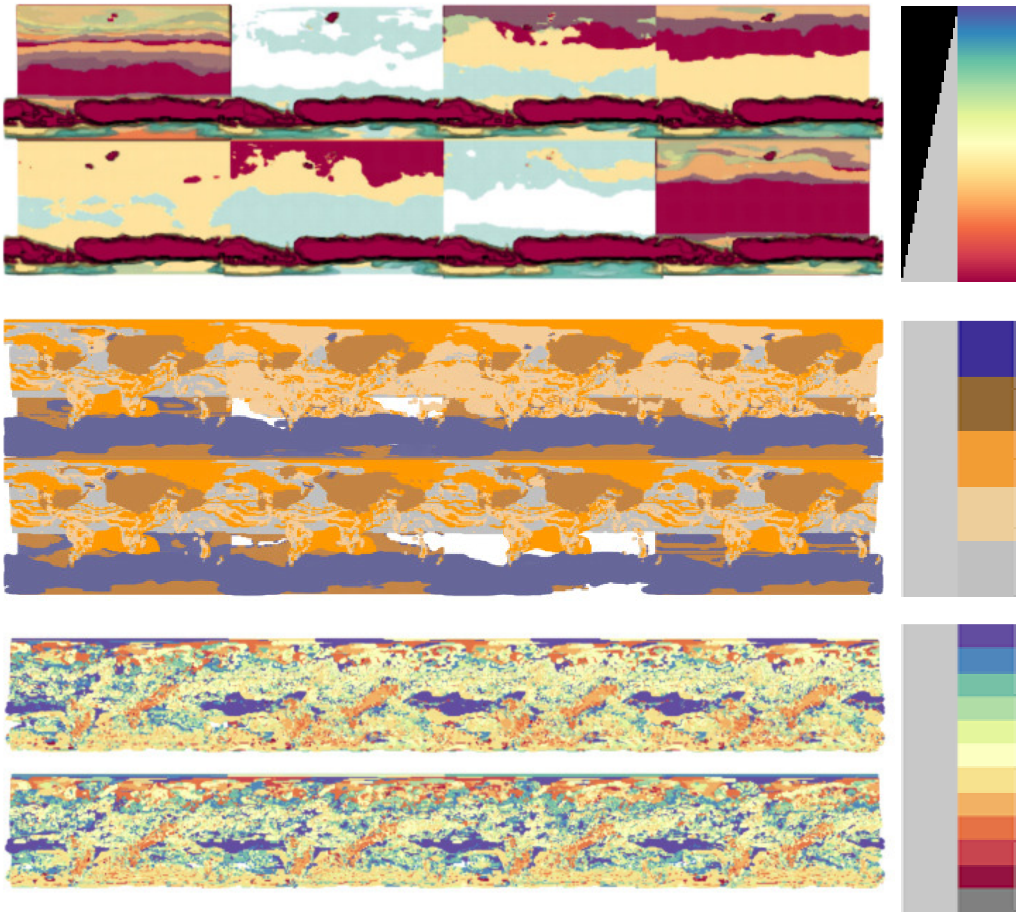


Fig. 6. ModelFit used to highlight ensemble differences. Row 1: similar to Figure 5, but classification is in probability space. Rendering is still of ModelFit's distance, but now in the range [0, 0.05] (extremes are specified directly, so we want best fits). Row 2: rendering of ModelFit's model variable. Extremes are specified for each of the five CO_2 variables in five separate models, hence the five color colormap. Row 3: Six regions are specified as extreme values for sets of two variables (from a total of three). There are 12 models (12 color colormap), each classifying the full dataset, but based on statistics limited to a single region.

differences in probability distributions. We are also interested in implementing the Mahalanobis distance. This is a generalization of Euclidean distance that brings in a local covariance matrix for each model. This would be useful when each model is a multivariate normal distribution with a different covariance matrix. It is a multivariate extension of measuring distances in local standard deviations instead of global units. In addition, we would like to implement more transformations, including those not assuming equal weights or independence among variables.

Finally, this work was developed to summarize complex interactions as single variables. However, to accomplish this there are certainly cases of (possibly substantial) information hiding. We would like to explore the possibilities of analyzing all data from the classification process, e.g. by creating a vector as output rather than a scalar.

References

[1] J. Katzav, H. A. Dijkstra, A. J. de Laat, Assessing climate model projections: State of the art and philosophical reflections, Studies in History and Philosophy of Science Part B: Studies in History and Philosophy of Modern Physics 43 (4) (2012) 258 – 276.

[2] R. Knutti, R. Furrer, C. Tebaldi, J. Cermak, G. Meehl, Challenges in combining projections from multiple climate models, Journal of Climate 23 (10) (2010) 2739–2758.

[3] P. J. Gleckler, K. E. Taylor, C. Doutriaux, Performance metrics for climate models, Journal of Geophysical Research: Atmospheres 113 (D6).

[4] C. D. Hewitt, D. J. Griggs, Ensembles-based predictions of climate changes and their impacts (ENSEMBLES), EOS 85 (2004) 566.

[5] F. M. Hoffman, C. C. Covey, I. Y. Fung, P. E. Thornton, Y. h Lee, N. A. Rosenbloom, R. C. Stckli, S. W. Running, D. E. Bernholdt, D. N. Williams, Results from the carbon-land model intercomparison project (c-lamp) and availability of the data on the earth system grid (esg).

[6] T. Bracegirdle, D. Stephenson, Higher precision estimates of regional polar warming by ensemble regression of climate model projections, Climate Dynamics 39 (12) (2012) 2805–2821.

[7] R. Furrer, S. Geinitz, S. R. Sain, Assessing variance components of general circulation model output fields, Environmetrics 23 (5) (2012) 440–450.

[8] C. Tebaldi, R. Smith, D. Nychka, L. Mearns, Quantifying uncertainty in projections of regional climate change: A bayesian approach to the analysis of multimodel ensembles, Journal of Climate 18 (10) (2005) 1524–1540.

[9] D. A. Stainforth, T. E. Downing, R. Washington, A. Lopez, M. New, Issues in the interpretation of climate model ensembles to inform decisions, Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences 365 (1857) (2007) 2163–2177.

[10] P. C. Wong, R. D. Bergeron, 30 years of multidimensional multivariate visualization, in: Scientific Visualization, Overviews, Methodologies, and Techniques, 1997, pp. 3–33.

[11] A. Inselberg, B. Dimsdale, Parallel coordinates: A tool for visualizing multi- dimensional geometry, in: Proc. of IEEE Visualization, 1990, pp. 361–375.

[12] H. Levkowitz, Color icons: Merging color and texture perception for integrated visualization of multiple parameters, in: Proc. of IEEE Visualization, 1991, pp. 164–170.

[13] J. LeBlanc, M. O. Ward, N. Wittels, Exploring n-dimensional databases, in: Proc. of IEEE Visualization, 1990, pp. 230–237.

[14] T. Mihalisin, J. Timlin, J. Schwegler, Visualization and analysis of multi-variate data: A technique for all fields, in: Proc. of IEEE Visualization, 1991, pp. 171–178.

[15] P. C. Wong, H. Foote, D. L. Kao, R. Leung, J. Thomas, Multivariate visualization with data fusion, Information Visualization 1 (3/4) (2002) 182–193.

[16] R. Becker, W. Cleveland, Brushing scatterplots, Technometrics 29 (2) (1987) 127–142.

[17] W. Stuetzle, Plot windows, J. of American Statistical Association 82 (398) (1987) pp. 466–475.

[18] H. Doleisch, M. Gasser, H. Hauser, Interactive feature specification for focus+context visualization of complex simulation data, in: Proceedings of the symposium on Data visualisation, VISSYM '03, Eurographics Association, Aire-la-Ville, Switzerland, pp. 239–248.

[19] X. Yuan, P. Guo, H. Xiao, H. Zhou, H. Qu, Scattering points in parallel coordinates, IEEE Trans on Visualization and Computer Graphics 15 (3) (2009) 1001–1008.

[20] R. A. Drebin, L. Carpenter, P. Hanrahan, Volume rendering, in: Proc. of SIGGRAPH'88, 1988, pp. 65–74.

[21] P. McCormick, J. Inman, J. Ahrens, C. Hansen, G. Roth, Scout: A hardware-accelerated system for quantitatively driven visualization and analysis, in: Proc. of IEEE Visualization, 2004, pp. 171–178.

[22] J. Kniss, S. Premoze, M. Ikits, A. Lefohn, C. Hansen, E. Praun, Gaussian transfer functions for multi-field volume visualization, in: Proc. of IEEE Visualization, 2003, pp. 497–504.

[23] G. Kindlmann, J. W. Durkin, Semi-automatic generation of transfer functions for direct volume rendering, in: IEEE Symp. on Volume Visualization, 1998, pp. 79–86.

[24] G. Kindlmann, R. Whitaker, T. Tasdizen, T. Möller, Curvature-based transfer functions for direct volume rendering: Methods and applications, in: Proc. of IEEE Visualization, 2003, pp. 513–520.

[25] I. Viola, A. Kanitsar, M. E. Gröller, Importance-driven volume rendering, in: Proc. of IEEE Visualization, 2004, pp. 139–145.

[26] K. Stockinger, J. Shalf, W. Bethel, K. Wu, Query driven visualization of large data sets, in: Proc. of IEEE Visualization, 2005, pp. 167–174.

[27] R. Maciejewski, I. Woo, W. Chen, D. Ebert, Structuring feature space: A non-parametric method for volumetric transfer function generation, IEEE Trans on Visualization and Computer Graphics 15 (6) (2009) 1473 –1480.

[28] H. Jänicke, A. Wiebel, G. Scheuermann, W. Kollmann, Multifield visualization using local statistical complexity, IEEE Trans on Visualization and Computer Graphics 13 (6) (2007) 1384–1391.

[29] C. Wang, H. Yu, K.-L. Ma, Importance-driven time-varying data visualization, IEEE Trans on Visualization and Computer Graphics 14 (6) (2008) 1547–1554.

[30] T. Pham, R. Hess, C. Ju, E. Zhang, R. Metoyer, Visualization of diversity in large multivariate data sets, IEEE Trans on Visualization and Computer Graphics 16 (6) (2010) 1053 –1062.

[31] S. Gerber, P. Bremer, V. Pascucci, R. Whitaker, Visual exploration of high dimensional scalar functions, IEEE Trans on Visualization and Computer Graphics 16 (6) (2010) 1271 –1280.

[32] J. D. Banfield, A. E. Raftery, Model-based gaussian and non-gaussian clustering, Biometrics 49 (1993) 803–821.

[33] C. Fraley, A. E. Raftery, Model-based clustering, discriminant analysis, and density estimation, J. of American Statistical Association 97 (2000) 611–631.

[34] H. Childs, E. S. Brugger, K. S. Bonnell, J. S. Meredith, M. Miller, B. J. Whitlock, N. Max, A contract-based system for large data visualization, in: Proceedings of IEEE Visualization 2005, 2005, pp. 190–198.

[35] A. Sklar, Fonctions de xrépartition à n dimensions et leurs marges, Publ. Inst. Statist. Univ. Paris 8 (1959) 229–231.

[36] R. I. Barbara Choroś, E. Permiakova, Copula estimation, in: P. Jaworski, F. Durante, W. Härdle, T. Rychlik (Eds.), Copula Theory and Its Applications, Springer, 2010, pp. 77–92, proc. of the Workshop held in Warsaw, 25-26 September 2009.

[37] W. Kendall, J. Wang, M. Allen, T. Peterka, J. Huang, D. Erickson, Simplified parallel domain traversal, in: Proceedings of 2011 International Conference for High Performance Computing, Networking, Storage and Analysis, ACM, pp. 10:1–10:11.