Model Space Visualization for Multivariate Linear Trend Discovery

Zhenyu Guo Matthew O. Ward Elke A. Rundensteiner

Computer Science Department
Worcester Polytechnic Institute
{zyguo,matt,rundenst}@cs.wpi.edu *

ABSTRACT

Discovering and extracting linear trends and correlations in datasets is very important for analysts to understand multivariate phenomena. However, current widely used multivariate visualization techniques, such as parallel coordinates and scatterplot matrices, fail to reveal and illustrate such linear relationships intuitively, especially when more than 3 variables are involved or multiple trends coexist in the dataset. We present a novel multivariate model parameter space visualization system that helps analysts discover single and multiple linear patterns and extract subsets of data that fit a model well. Using this system, analysts are able to explore and navigate in model parameter space, interactively select and tune patterns, and refine the model for accuracy using computational techniques. We build connections between model space and data space visually, allowing analysts to employ their domain knowledge during exploration to better interpret the patterns they discover and their validity. Case studies with real datasets are used to investigate the effectiveness of the visualizations.

Keywords: Knowledge Discovery, visual analysis, multivariate linear model construction, model space visualization.

Index Terms: H.5.2 [Information Interfaces and Presentation]: User Interfaces—Graphical user interfaces

1 INTRODUCTION

Discovering and extracting useful insights in a dataset are basic tasks in data analysis. The insights may include clusters, classifications, trends, outliers and so on. Among these, linear trends are one of the most common features of interest. For example, when users attempt to build a model to represent how horsepower x_0 and engine size x_1 influence the retail price y for predicting the price for a given car, a simple estimated linear trend model ($y = k_0x_0 + k_1x_1 + b$) could be helpful and revealing. Many computational approaches for constructing linear models have been developed, such as linear regression [6] and response surface analysis [3]. However, the procedure and results are not always useful for the following reasons:

- *Lack of efficiency*: When discovering trends in a large dataset, users are often only concerned with a subset of the data that matches a given pattern, so only these data should be used for the computation procedure rather than the whole dataset. Furthermore, locating a good estimation of the trend as an initial input for the regression analysis could expedite the convergence, especially for high dimensional datasets.
- *Lack of accuracy*: Computational results are often not as accurate as the user expects because users are unable to apply their own domain knowledge and perceptual ability during and after discovering models. User-driven modeling and tuning may be required.

*This work is supported under NSF grant IIS-0812027.

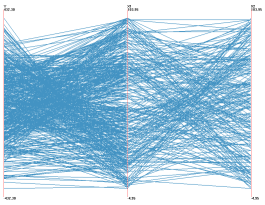

Figure 1: A dataset with a simple linear trend: $y = 3x_1 - 4x_2$ is displayed with parallel coordinates. The axes from left to right are y, x_1 and x_2 respectively.

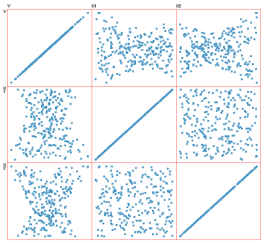

Figure 2: A dataset with two linear trends: $y = 3x_1 - 4x_2$ and $y = 4x_2 - 3x_1$ is displayed with a scatterplot matrix.

- *Parameter setting problem*: Most model estimation techniques require users to specify parameters, such as the minimum percentage of data points the model includes, maximum error tolerance and iteration count. These are often particular to a concrete dataset, application, and task, but users often don't know conceptually how to set them.
- *Multiple model problem*: If multiple phenomena coexist in the same dataset, many analytic techniques will extract poor models.

Locating patterns in a multivariate dataset via visualization techniques is very challenging. Parallel coordinates [10] is a widely used approach for revealing high-dimensional geometry and analyzing multivariate datasets. However, parallel coordinates often performs poorly when used to discover linear trends. In Figure 1, a simple three dimensional linear trend is visualized in parallel coordinates. The trend is hardly visible even though no outliers are involved. Scatterplot matrices, on the other hand, can intuitively reveal linear correlations between two variables. However, if the linear trend involves more than two dimensions, it is very difficult to directly recognize the trend. When two or more models coexist in the data (Figure 2), scatterplot matrices tend to fail to differentiate them.

Given a multivariate dataset, one question is how to visualize the model space for users to discern whether there are clear linear trends or not. If there are, is there a single trend or multiple trends? Are the variables strongly linearly correlated or they just spread loosely in a large space between two linear hyperplane boundaries? How can we visually locate the trend efficiently and measure the trend accurately? How can we adjust arbitrarily the computational model estimation result based on user knowledge? Can users identify outliers and exclude them to extract the subset of data that fits the trend with a user indicated tolerance? How can we partition the dataset into different subsets fitting different linear trends?

We seek to develop a system focusing on these questions. Specifically, we have designed a visual interface allowing users to navigate in the model space to discover multiple coexisting linear trends, extract subsets of data fitting a trend, and adjust the computational result visually. The user is able to select and tune arbitrary

high-dimensional linear patterns in a direct and intuitive manner. We provide a sampled model space measurement map that helps users quickly locate interesting exploration areas. While navigating in the model space, the related views that provide metrics for the current selected trend, along with the status of data space, are dynamically displayed and changed, which gives users an accurate estimation to evaluate how well the subset of data fits the trend.

The primary contributions of this paper include:

- *A novel linear model space environment*: It supports users in selecting and tuning any linear trend pattern in model space. Linear patterns of interest can be discovered via interactions that tune the pattern hyperplane position and orientation.
- *A novel visualization approach for examining the selected trend*: We project color-coded data points onto a perpendicular hyperplane for users to decide whether this model is a good fit, as well as clearly differentiating outliers. Color conveys the degree to which the data fits the model. A corresponding histogram is also provided, displaying the distribution relative to the trend center.
- *A sampled measurement map to visualize the distribution in model space*: This sampled map helps users narrow down their exploration area in the model space. Multiple hot-spots indicate that multiple linear trends coexist in the datasets. Two modes with unambiguous color-coding scheme help users conveniently conduct their navigation tasks. Two color-space interactions are provided to highlight areas of interest.
- *Linear trend dataset extraction and management*: We present a line graph trend tolerance selection for users to decide the tolerance (maximum distance error tolerance from a point to the regression line) for the current model. Users can refine the model using a computational modeling technique after finding a subset of linearly correlated data points. We also allow the user to extract and save data subsets to facilitate further adjustment and examination of their discovery.

The remainder of this paper is organized as follows: In Section 2 existing techniques for model space visualization, user exploration, and visual linear trend discovery are reviewed. In Section 3 we introduce our model space visualization method and how to navigate in the model space via our proposed pattern selection panel. The related views that provide metrics for the chosen pattern are then presented. Section 4 describes the sample model space map that guides user in exploring model space efficiently, and also discusses the power of this map for revealing multiple trends. Section 5 is dedicated to the discussion of a case study involving the analysis of a real dataset. We conclude this paper in Section 6 with a summary and possible future research directions.

2 Related work

Visual data mining techniques [20, 5] suggest that a more efficient and powerful data mining strategy should involve users in the visual analytical processes rather than being carried out completely by machines. Recently, numerous visual analytics based systems have been presented to solve knowledge discovery tasks. Schreck et al. [18] propose a user-supervised SOM clustering algorithm that enables users to control and monitor the computation process visually to leverage their domain knowledge. ClusterSculptor [15] describes a framework to assist users in extracting clusters directly in N-D data space, allowing them to tune the parameters interactively based on visual presentation of data characteristics. Savikhin et al. [17] created a system for helping subjects improve decision making through the use of an interactive visual analytics program. The Nugget Management System (NMS) [21] provides a framework for analysis-guided visual exploration of multivariate data for users to manage their discoveries from range queries.

Model space visualization and model-driven analytics have been studied for many years. The Hough Transform, widely used in image processing and computer vision, builds a connection between data space patterns and model space representations for detecting arbitrary shapes, such as lines, circles and ellipses [7]. The GGobi package [19] provides a Grand Tour [1] method for viewing multivariate data via orthogonal projections onto a sequence of two-dimensional subspaces (scatter plots). Garg et al. [8] present a visual high-dimensional data navigation and pattern painting system that enables users to construct models and study relationships present in the dataset. These techniques, however, are not designed to reveal linear models within multi-dimensions effectively and do not provide metrics for the patterns users discovered and measurement of the model space they explored.

Model estimation and selection is a very important task for understanding multivariate characteristics, attributes and correlations. Numerous computational approaches and algorithms have been proposed and discussed. Cherkassky and Ma [4] introduced a constructive support vector machine (SVM)-based approach for multiple regression estimation. Li et al. [11] present a new class of variable-structure (VS) algorithms for multiple-model estimation. D2MS [9] is a research system for knowledge discovery with support for model selection and visualization, providing the user with the ability to try various alternatives of algorithm combinations and their settings. MVE [2] is a multivariate visual system that helps user interactively construct models and analyze multi-dimensional relationships. These techniques are primarily directed toward automatic or unsupervised model discovery. We will not only propose a novel approach for model space visualization for the purpose of linear trend discovery, but also present a system that assists users in applying their domain knowledge for customized linear pattern detection.

3 Model Space Visualization

3.1 Linear Trend Nugget Definition

We define a *nugget* as a pattern within a dataset that can be used for reasoning and decision making [21]. A linear trend in n-dimensional space can be represented as $(w,X)-b=0$, where $X_i \in \mathbb{R}^n$ denotes a combination of independent variable vector x_i ($x_i \in \mathbb{R}^{n-1}$) and a dependent target value y ($y \in \mathbb{R}$). Here w and b are respectively a coefficient vector and a constant value ($w \in \mathbb{R}^n$, $b \in \mathbb{R}$). The data points located on this hyperplane construct the center of the trend. A data point x that fits the trend should satisfy the constraint

$$|(w,x)-b| < \varepsilon$$

Considering that noise could exist in all variables (not just the dependent variable), it may be appropriate to use the Euclidean distance from the regression hyperplane in place of the vertical distance error used above [12]. We define a *linear trend nugget* (LTN) as a subset of the data near the trend center, whose distance from the model hyperplane is less than a certain threshold E:

$$LTN(X) = \{x | \frac{|(w,x)-b|}{\|w\|} < E\}$$

Here E is the maximum distance error, which we call *tolerance*, for a point to be classified as within the trend. If the distance from a data point to the linear trend hyperplane is less than E, it is covered and thus should be included in this nugget. Otherwise it is considered as an outlier or a point that does not fit this trend very well. The two hyperplanes whose offsets from the trend equal E and $-E$ construct the boundaries of this trend. The goal of our approach is to help users conveniently discover a good linear model, denoted by

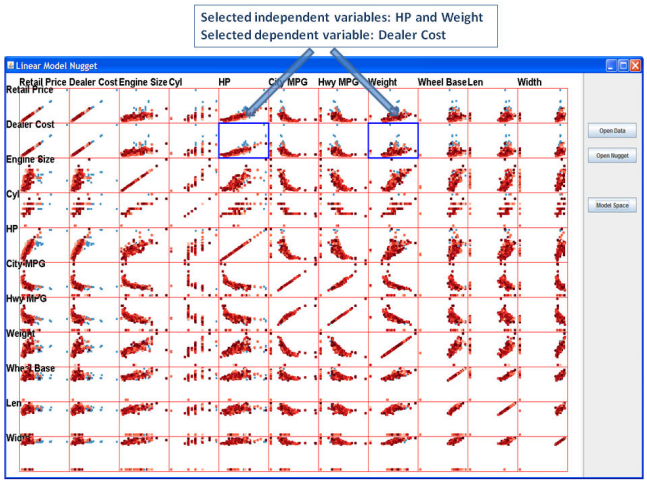

Figure 3: The Data Space interface overview.

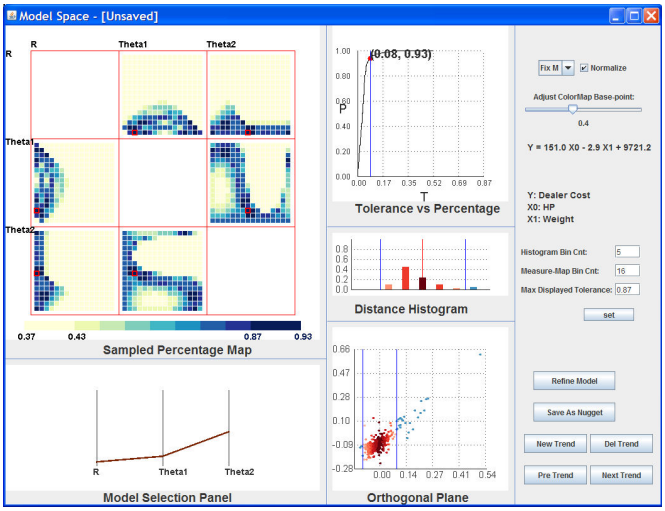

Figure 4: The Model Space interface overview.

a small tolerance and, at the same time, covering a high percentage of the data points.

As the range of the values in the coefficient vector could be very large and even infinite, we transform this linear equation into a normal form to make $\|w\| = 1$ and then represent this vector as S^n, a unit vector in hypersphere coordinates [14] as described in [7]:

$$w_0 = \cos(\theta_1)$$
$$w_1 = \sin(\theta_1)\cos(\theta_2)$$
$$\cdots$$
$$w_{n-2} = \sin(\theta_1)\cdots\sin(\theta_{n-2})\cos(\theta_{n-1})$$
$$w_{n-1} = \sin(\theta_1)\cdots\sin(\theta_{n-2})\sin(\theta_{n-1})$$

Now our multivariate linear expression can be expressed as:

$$y\cos(\theta_1) + x_1\sin(\theta_1)\cos(\theta_2) + \cdots +$$
$$x_{n-2}\sin(\theta_1)\sin(\theta_2)\cdots\sin(\theta_{n-2})\cos(\theta_{n-1}) +$$
$$x_{n-1}\sin(\theta_1)\sin(\theta_2)\cdots\sin(\theta_{n-2})\sin(\theta_{n-1}) = r$$

The last angle θ_{n-1} has a range of 2π and others have a range of π. The range of r, the constant value denoting the distance from the origin to the trend hyperplane, is $(0, \sqrt{n})$ after normalizing all dimensions.

An arbitrary linear trend can now be represented by a single data point $(\theta_1, \theta_2, \cdots, \theta_{n-1}, r)$ in the model parameter space. Users can select and adjust any linear pattern in data space by clicking and tuning a point in the model space.

3.2 System Overview

We now briefly introduce the system components and views. The overall interface is depicted in Figures 3 and 4. The user starts from a data space view displayed with a scatterplot matrix. To explore in the linear model space, the user first indicates the dependent variable and independent variables via clicking several plots in one row. The clicked plots are marked by blue margins; clicking the selected plot again undoes the selection. The selected row is the dependent variable and the columns clicked indicate the independent variables. After the user finishes selecting the dependent and independent variables, he/she clicks the "model space" button to show and navigate in the model space. The points in the data space scatterplot matrix are now colored based on their distance to the currently selected linear trend and dynamically change when the user tunes the trend in the model space. As shown in Figure 3, the selected dependent variable is "Dealer Cost" and the two independent variables are "Hp" and "Weight". The points are color-coded based on the currently selected trend: dark red means near the center and lighter red means further from the center; blue means the points do not fit the trend. Figure 4 is the screen shot of the model space view. Each view in the model space is labeled indicating the components, as described in the following sections.

3.3 Linear Trend Selection Panel

We employ Parallel Coordinates (PC), a common visualization method for displaying multivariate datasets [10], for users to select and adjust any linear trend pattern. Each poly-line, representing a single point, describes a linear trend in data space. PC was chosen for its ability to display multiple trends at the same time, along with the metrics for each trend. For example, average residual and outlier percentage are easily mapped to poly-line attributes, such as line color and line width. Users can add new trends, delete trends and select trends via buttons in the model space interaction control panel. Users can drag up and down in each dimension axis to adjust parameter values. During dragging, the poly-line attributes (color and width) dynamically change, providing users easy comprehension of pattern metrics. The parameter value of the current axis is highlighted beside the cursor. This direct selection and exploration allows users to intuitively tune linear patterns in model space, sensing the distance from hyperplane to origin as well as the orientations rotated from the axes. Because the parameters in hypersphere coordinates can be difficult to interpret, the familiar formula in the form of $y = k_0x_0 + k_1x_1 + \cdots + k_{n-1}x_{n-1} + b$ is calculated and displayed in the interface. In Figure 5, three linear trends for a 3-D dataset are displayed. The percentage of data each trend covers (with the same model tolerance) is mapped to the line width and the average residual is mapped to color (dark brown means a large value and light yellow means small).

3.4 Views for Linear Trend Measurement

When the user tunes a trend in model space, it is necessary to provide detailed information in data space related to the currently selected trend. Based on this the user can differentiate datasets having linear trends from non-linear trends or without any clear trends, as well as discover a good model during tuning. We provide users three related views for discovering trends and deciding the proper

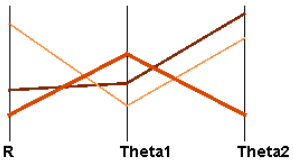

Figure 5: The Model Space Pattern Selection Panel.

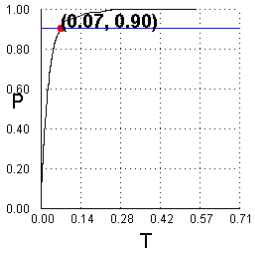

Figure 6: The Line Graph of Model Tolerance vs. Percent Coverage.

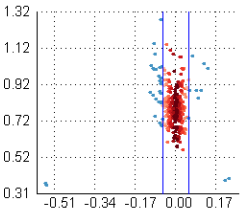

Figure 7: The Orthogonal Projection Plane.

model parameters.

Line Graph: Model Tolerance vs. Percent Coverage

For any multi-dimensional linear trend, there is a positive correlation between the tolerance of the model (the distance between the trend hyperplane and the furthest point considered belonging to the trend) and the percentage of data points this model covers: the larger the model tolerance is, the higher the percentage it covers. There is a trade-off between these two values, because users generally search for models with small tolerance that cover a high percentage of the data. The users expect to find the answer to the following two questions when deciding the model tolerance and percentage it covers: (a) If the model tolerance is decreased, will it lose a large amount of the data? (b) If this trend is expected to cover a greater percentage of the data, will it significantly increase the model tolerance?

To answer these questions, we introduce an interactive line graph for the currently selected model. Model Tolerance vs. Percent Coverage is provided for users to evaluate this model and choose the best model tolerance. It is clear that the line graph curve always goes from $(0,0)$ to $(1,1)$, after normalizing. This line graph also indicates whether this model is a good fit or not. If this curve passes the region near the $(0,1)$ point, there is a strong linear trend existing in the dataset, with a small tolerance and covering a high percentage of the data. This interactive graph also provides a selection function for the model tolerance. The user can drag the point position (marked as a red filled circle in Figure 6) along the curve to enlarge or decrease the tolerance to include more or fewer points.

Figure 6 shows an example of how to use this view to discover a good model. The line graph for a linear trend with about 9 percent outliers is shown. The red point on the curve indicates the current status of model tolerance and percentage. From the curve of the line graph, it is easy to confirm that when dragging the point starting from $(0,0)$ and moving towards $(1,1)$, the model tolerance increases slowly as the percentage increases, meaning that a strong linear trend exists. After moving across 0.90 percent, the model tolerance increases dramatically while the included point percentage hardly increases, indicating that the enlarged model tolerance is mostly picking up outliers. So for this dataset, the user could claim that a strong trend is discovered covering 90 percent of the data points because the model tolerance is very small (0.07). The corresponding Orthogonal Projection Plane view and Histogram view showing the distribution of data points are displayed in Figure 7 and Figure 8 (described next).

Projection on the Orthogonal Plane

Given an n-dimensional dataset and an n-dimensional linear trend hyperplane, if the user wants to know whether the dataset fits the plane (the distance from points to the hyperplane is nearly 0), a direct visual approach is to project each data point onto an orthogonal hyperplane and observe whether the result is nearly a straight line.

In particular, we project each high-dimensional data point to a 2-dimensional space and display it in the form of a scatterplot, similar to the Grand Tour [1]. Two projection vectors are required: the first vector v_0 is the normal vector of the trend plane, i.e. the unit vector w described before; the second vector v_1, which is orthogonal to v_0, can be formed similar to v_0, simply by setting $\theta_1 = \theta_1 + \pi/2$. The positions of data points in the scatterplot are generated by the dot products between the data points and the two projection vectors, denoting the distance from the points to the trend hyperplane and another orthogonal plane, respectively. This view presents the position of each point based on their distance to the current trend, which provides users not only a detailed distribution view based on the current trend, but also the capability of discovering the relative positions of ouliers. Figure 7 shows the projection plane. The two blue vertical lines denote the two model boundaries. Data points are color-coded based on their distance to the trend center (not displayed). The red points are data points covered by this trend; darker red means near the center and lighter red means further from the center. The blue points are data that are outliers or ones that do not fit this trend very well.

Linear Distribution Histogram

The histogram view displays the distribution of data points based on their distance to the current model. As shown in Figure 8, the middle red line represents the trend center and the right half represents the points above the trend hyperplane; and the left half are those below the trend hyperplane. Users can set the number of bins; the data points included in the trend are partitioned into that number of bins based on their distance to the trend center. The two blue lines represent the boundary hyperplanes. The trend covered bars are red and color-coded according to their distance. The color-mapping scheme is the same as the projection plane view so the user can easily compare these two views. The two blue bars represent the data outside the trend; the right bar is for the data whose position is beyond the upper boundary and the left bar is for the data whose position is beyond the lower boundary.

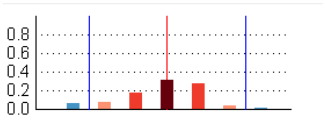

Figure 8: The Histogram View.

3.5 Nugget Refinement and Management

After finding a good model covering a larger number of data points, the analyst can use a refinement function to tune the model using a computational technique. We employ Least Median Squares [16], a robust regression technique, to compute the regression line based only on the points covered in the current trend, so it is more efficient than basing it on the whole dataset and more accurate because

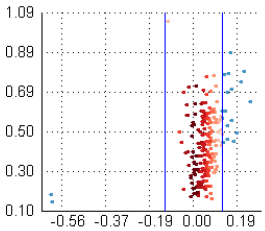

Figure 9: The Projection Plane view before refinement.

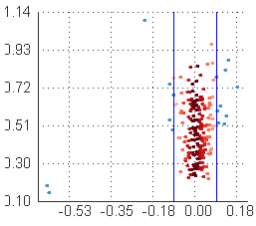

Figure 10: The Projection Plane view after refinement.

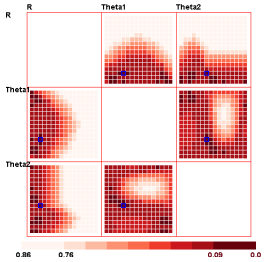

Figure 11: The Measurement Map: mode is "fix coverage".

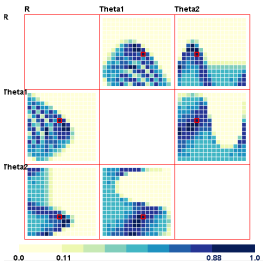

Figure 12: The Measurement Map: mode is "fix model tolerance".

the outliers are not considered. Figure 9 shows the user-discovered trend before refinement and Figure 10 shows the refinement results.

A linear trend nugget is a subset of data points that lie within trend boundaries. Assuming the user has discovered a trend within several dimensions, it is useful to save it to a file and reload it to examine, adjust and distribute it to other users. After the users find a strong trend, they can extract the points in the trend by saving it as a nugget file. This model selection method is similar to brushing techniques and provides a convenient way for users to identify and exclude outliers that deviate from the trend. This data selection technique is also useful if multiple phenomena are present in the dataset, since the user can save and manage them separately.

4 Navigation in Model Space and Linear Trend Model Discovery

4.1 Sampled Measurement Map Construction

Even with the metrics of a linear pattern mapped to the poly-line attributes and with the related views for single model measurement mentioned in Section 3, the user may still feel challenged when searching for good linear trends by tuning the parameter space values, due to the large search area associated with multiple data dimensions. We introduce a sampled model space measurement map for users to view the high dimensional model measurement distribution and navigate in the model space directly and efficiently. The basic idea is that we sample some points in the model space and calculate the measurements for each point (linear pattern), so the user can tune the patterns starting from good parameter sets.

This map is constructed via the following three steps:

(a) We first partition each parameter space variable into several bins. The points in model space located in the center of each combination of bins are selected as sample patterns and the metrics are calculated for model measuring.

(b) Then we eliminate the patterns with low measurement values and project a high dimensional sampled pattern set to a series of two dimensional pairs. Specifically, for each paired bin position in two dimensions, only the largest measurement (assume larger measurement values denote better models) with the same bin position of these two dimensions is kept as the map value. For example, the bottom left bin in one plot corresponds to the two first bin position in that dimensional pair, say, bin position 1 for dimension i and bin position 1 for dimension j (the bin number starts from 1). The map value for this position of this dimension pair is selected as the largest measurement in all the sampled patterns whose bin position in the ith dimension and the jth dimension are both 1.

(c) The map values are color-coded based on the metrics. All the pairwise measurement maps are displayed in a matrix view. The initial parameter values are set at the center of the bin with the best measurement, i.e. the minimum tolerance or the maximum percent coverage when fixing the other, which generally provides a good linear pattern for users to start tuning.

The complexity of construction is $PrP_1P_2\cdots P_{n-1}N$, where N is the size of dataset;Pr is the number of patitions for r and P_i is the number of patitions for θ_i.

Two alternative modes are associated with this view, fixed percent coverage and fixed model tolerance, corresponding to the two measurements for the trends. As mentioned before, the user could change the model tolerance and coverage together in the line graph view. For the first mode, with model tolerance as the measurement, each bin on the map represents a model tolerance with a user-indicated fixed coverage. When the user changes the percentage, this map is dynamically re-calculated and changed (Figure 11). For each pairwise bin position in the two dimensional pair, the minimum model tolerance is selected as map value and mapped to color. In this mode, the percentage of points the user wants to include in the trend is designated and users can search for the smallest model tolerances.

The second mode is similar to the first one (Figure 12). The difference is we change the measurement to coverage, with a user-indicated fixed model tolerance. This mode is designed for users to specify the maximum model tolerance and search for models that cover a high percentage of points.

For the two modes of measurement map, we use two unambiguous color-coding schemes: (a) Model tolerance is mapped from dark red to light pink, with dark red meaning small model tolerance. (b) The coverage is mapped to color from yellow to blue, with blue meaning large coverage.

When the user moves the cursor over each bin, the map value is shown. The bin in which the current model resides is highlighted by a colored boundary. The parameter values are dynamically changed to the bin center, with the largest measurement value as mentioned before, when the user clicks or drags to a certain bin position. This map indicates roughly where good models can be found before tuning the model in the parallel coordinates view. Figure 12 shows the coverage distribution map in a 3 dimensional linear trend display. Users can easily find interesting hot spots and drag or click the current selected bin into a dark blue area.

4.2 Color Space Interactions

It is common that several bins with similar values of interest are shown at the same time in the sampled map near the local maximum, making it hard to locate the best settings. To solve this problem, we provide two interactions in color space:

(a) Scale the map value to employ the whole color range. Because the values are normalized to $(0,1)$ and then mapped to color, it is possible that all map values are in a small range; for example, all the coverage values in the map might be located in $(0.7,1)$ for a very large tolerance in the second mode. In other words, the color map range is not fully used. We allow the user to scale the value range to $(0,1)$ to use the whole color map.

(b) Color map base-point adjustment. For the sampled measurement map, the user is only concerned with the high metric values, so

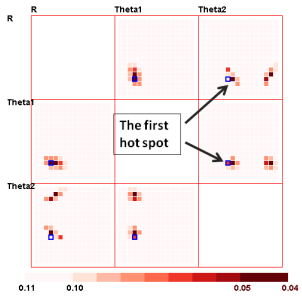

Figure 13: The first hot spot is selected representing the first linear trend.

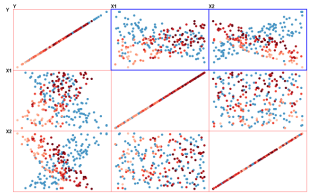

Figure 14: The data points that fit the first trend are highlighted in red color.

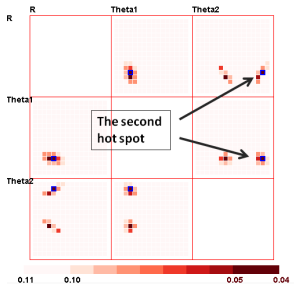

Figure 15: The second hot spot is selected representing another linear trend.

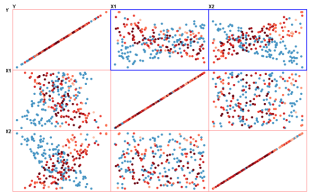

Figure 16: The data points that fit the second trend are highlighted in red color.

a "filter" function to map values less than a threshold to 0 is useful for users to locate the local maximum. In particular, we provide a function for users to change the color map base-point as the threshold. After filtering out the uninteresting areas with low metrics, users can more easily find the positions of good models.

The color space interactions are illustrated from Figures 18 to 21 and described in Section 5.

4.3 Multiple Coexisting Trends Discovery

This map is also designed to reveal when multiple linear trends coexist in the dataset, which is very hard to find without visualization. Figure 2 shows an example where two linear trends, $y = 3x_1 - 4x_2$ and $y = 3x_2 - 4x_1$ coexist in the three dimension dataset mentioned earlier. Each trend has 50 percent of the data points. When the user fixes the percentage at 0.50, there are clearly two separate hot spot regions indicating two linear trends coexist. Figure 13 shows two different hot spots in the sampled map with one of them selected (colored bin). The corresponding subset of data that fit this trend are colored as shown in Figure 14. Red means the point fits the model and blue means it doesn't. The other trend and fitting data are shown in Figure 15 and 16.

5 Case Study

In this section, we discuss case studies showing how to discover single or multiple linear trends and construct models for real datasets. The dataset was obtained from the Mn/DOT Traveler Information [13], that collects traffic data on the freeway system throughout the Twin Cities Metro area. Each data point is the traffic information collected by detectors every 30 seconds. The information includes the following variables:

(a) *Volume*: the number of vehicles passing the detector during the 30 second sample period. (b) *Occupancy*: the percentage of time during the 30 second sample period that the detector sensed

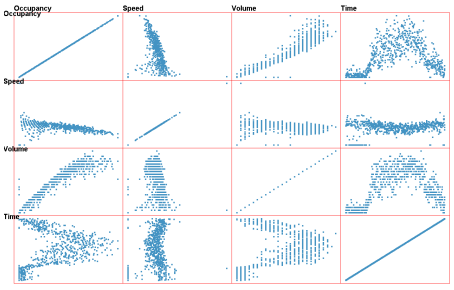

Figure 17: Traffic dataset data space view (scatterplot matrix).

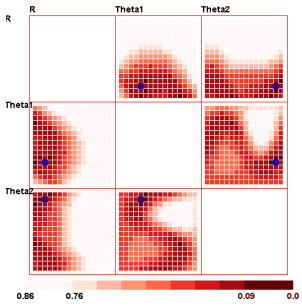

Figure 18: The measurement map with the original color range.

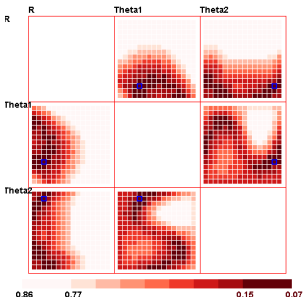

Figure 19: After full use of the color map.

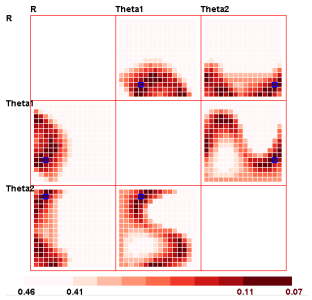

Figure 20: Adjust the color map base point to 0.46.

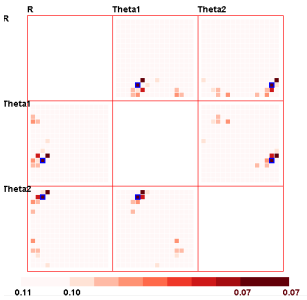

Figure 21: Adjust the color map base point to 0.11.

a vehicle. (c) *Speed*: the average speed of all vehicles passing the detector during the 30 second sample period.

We collected the traffic information for a whole day and added another variable based on the index order to represent the time stamp. Figure 17 shows the dataset displayed in a scatterplot matrix. Assume a user wants to analyze the correlations between dependent variable occupancy and independent variables speed and volume and construct linear models for these three variables. The aim of this study is to analyze how the average speed and vehicle numbers interactively influence the occupancy. The result is helpful for detecting anomalies, dealing with missing data points and predicting traffic conditions, which can be used for traffic surveillance and control.

If the user wants to build a single linear model to explain the correlations, the first step is to select the view mode and adjust the point on the line graph view to indicate the model tolerance or coverage. Here we use the first mode to discover a model and indicate 85 percent of the data to be covered by the trend, and then search for models with small tolerances.

For further analysis, users can navigate in sampled measurement

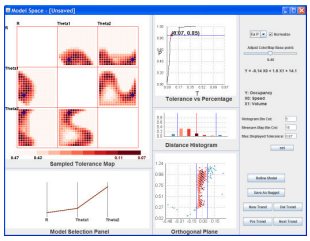

Figure 22: The model space view: a discovered linear trend in a bin center.

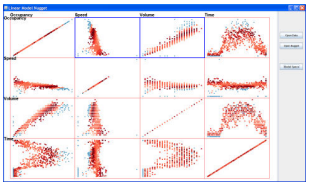

Figure 23: The corresponding data space view.

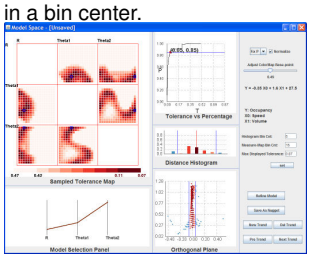

Figure 24: The model space view: a better linear trend after user adjustment and computational refinement.

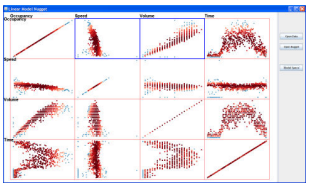

Figure 25: The corresponding data space view.

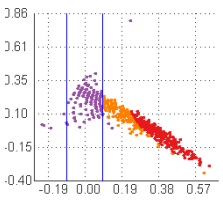

Figure 26: Trend fit the data points with low volume.

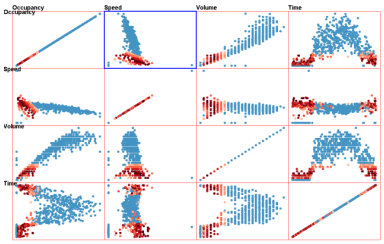

Figure 27: Data Space view. The two dimensional trend is $y = -0.11x + 13.8$ (y: Occupancy, x: speed).

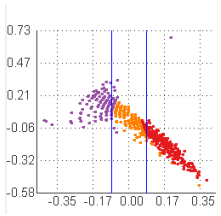

Figure 28: Trend fit the data points with medium volume.

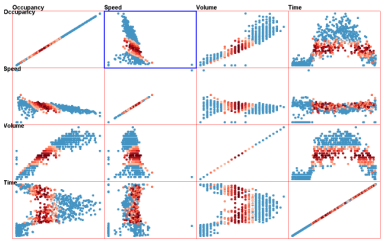

Figure 29: Data Space view. The two dimensional trend is $y = -0.17x + 29.7$ (y: Occupancy, x: speed).

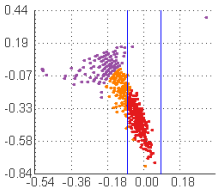

Figure 30: Trend fit the data points with high volume.

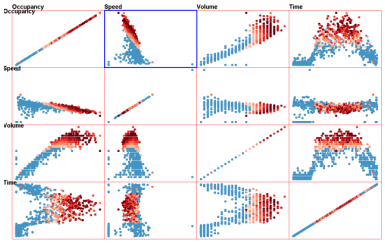

Figure 31: Data Space view. The two dimensional trend is $y = -0.38x + 60.2$ (y: Occupancy, x: speed).

map and model selection panel alternately to observe the orthogonal projection plane and histogram to decide whether the current model is a good estimation. To narrow down the search area, the user explores first in the sampled measurement map to drag or click a bin with a good estimation of the model parameters. Notice that the user is only concerned with dark red bins indicating a small tolerance; the user could interact with the color space to fully use the color map and then adjust the color map base-point until the uninteresting areas are eliminated and only red areas remain.

Figures 18 to 21 show the manipulation details for locating the local maximum value in the sampled measurement map. Figure 18 shows the map with the original color range and Figure 19 shows the map after fuller use of the color range. Figures 20 and 21 show the process of adjusting the base point from 0.86 to 0.46 and then 0.11 (shown in the color bar legend). If the map value (tolerance) is larger than this base point, then it will be set to 1 and then mapped to color. From Figure 22, the user can easily locate the approximate position of good models and then tune them in the model selection panel.

Figure 22 shows the model metric views for the trend in the bin center (model tolerance is 0.07); its corresponding data space view is shown in Figure 23. Figure 24 shows the adjusted model that fits the data better (model tolerance is 0.05) via tuning the parameter values in the parallel coordinate view; Figure 25 displays the data space view.

After refining the model, a good linear estimation for the three variables is constructed: a trend with small tolerance (0.05) covering more than 85 percent of the data points ($y = -0.29x_0 + 1.4x_1 + 25.3$, y: Occupancy, x_0: Speed, x_1: Volume). From the linear equation, we notice that occupancy is negatively correlated with car speed and positively correlated with volume. This three dimensional linear trend plane could also be observed after projection to a two dimensional plane in the data space view displayed by scatterplot matrices. From this we conclude that the more vehicles and the lower speed of the vehicles, the higher percentage of time the detector sensed vehicles, which is fairly intuitive.

Can we use this model to estimate occupancy when we know the speed and vehicle numbers? When we look at the data space view in which the data points are colored according to their distance to the trend, we found this model estimates occupancy well for most of the data points, except the data collected at noon and night. Therefore, a single linear trend could not fit all the data points well, except by increasing the model tolerance to a larger value.

If users want to explain the phenomenon by a single linear trend, the slope of the trend line of occupancy vs. speed does not change for different volume numbers (only the intercept changes). If users want to construct a more complex model with several trends to estimate the occupancy more accurately, multiple linear trends considering different levels of volume can be discovered.

For multiple trend modeling, each trend is not required to cover a large percentage of data points. Conversely, each trend needs to be a strong linear trend represented by a very small tolerance. Therefore, we chose the second mode, i.e. fixed tolerance, and adjust the tolerance to a very small value and then explore in model space as mentioned before. Notice that the value of volume is a discrete number, so it is easy to observe from the Orthogonal Projection Plane view that each subset of data with the same volume value is nearly a straight line in three-dimensional space and the lines are nearly parallel. Thus we adjust the parameter values until each subset of data with a similar volume value aligns to the trend center (Figure 32). Adjust the first parameter value (the distance from the hyperplane to the origin) from zero to maximum to extract the data points with different volume values (3 different levels: low volume,

median volume and high volume, colored by purple, yellow and red respectively). We can observe from the data space view that different subsets of data reveal different linear trends in the plot of speed vs. occupancy.

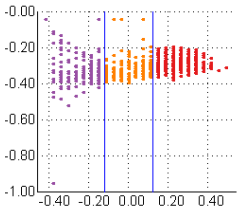

Figure 32: The Orthogonal Projection Plane view after adjusting so that data points with similar volume align to the linear trend center. Color coding: purple points are low volume; yellow points are median volume; red points are high volume.

We then select two dimensional correlation with occupancy as the dependent variable and speed as the independent variable. We color-code the third dependent variable *volume* with three levels in the orthogonal projection plane view and adjust the parameters to fit different subsets of data with different levels of volume. Figure 26 to 31 show the three subsets of data fit to different discovered linear trends after refinement in the orthogonal projection plane view and data space view. We can observe from the data space view that as the number of vehicles passing the detector changes, the trend for speed and occupancy alters: the more vehicles passing through, the higher the trend line is and, also the steeper the slope of the trend line. If the volume and speed are known for estimating the occupancy, the user can classify the volume into three bins: low, medium and high, and use different trend lines of speed vs. occupancy to estimate the occupancy value.

How can one explain this model with multiple linear trends for different volumes? If it is ensured that when the detector senses a vehicle, there is only a single car (without any overlapping) passing the detector, then the occupancy is mainly influenced by volume (also influenced a little by speed, but not significantly when volume number changes); it is also clear that low volume indicates low occupancy, which is demonstrated by the lower and less steep trend for speed vs. occupancy when volume is low. But sometimes, especially when volume is large, several vehicles pass the detector together: consider that when two overlapping vehicles pass the detector together, the volume increases but occupancy doesn't. As the volume increases, the occupancy increases, and meanwhile, the degree of vehicle overlapping increases. When the volume is large, meaning that several vehicles pass the detector together with overlapping, the occupancy is not as predictable just based on volume as it is when volume is small. This suggests the average speed will be more helpful for estimating occupancy. A steeper and higher trend for speed vs. occupancy when volume is large means that occupancy depends more on speed than on volume.

6 Conclusion

In this paper, we describe a novel model space visualization technique to support users in discovering linear trends among multiple variables. Using this system, analysts can discover linear patterns and extract subsets of the data that fit the trend well by navigating in the model space and building connections between model space and data space visually. The case studies show how our system can be used effectively to reveal single and multiple linear trends and to build explanation models for multivariate datasets. In the future, we plan to expand our system to support constructing more complex and generalized models other than linear ones, such as quadratic and logarithmic correlations among variables. In addition, we believe many of the same techniques can be applied to the design of linear classifiers, which we are actively pursuing.

References

[1] D. Asimov. The grand tour: a tool for viewing multidimensional data. *SIAM J. Sci. Stat. Comput.*, 6(1):128–143, 1985.

[2] S. Barlowe, T. Zhang, Y. Liu, J. Yang, and D. Jacobs. Multivariate visual explanation for high dimensional datasets. *IEEE Symposium on Proc. VAST '08.*, pages 147–154, Oct. 2008.

[3] G. E. P. Box and N. R. Draper. *Empirical model-building and response surface*. John Wiley & Sons, Inc., New York, NY, USA, 1986.

[4] V. Cherkassky and Y. Ma. Multiple model regression estimation. *Neural Networks, IEEE Transactions on*, 16(4):785–798, July 2005.

[5] M. C. F. de Oliveira and H. Levkowitz. From visual data exploration to visual data mining: A survey. *Journal of Computational and Graphical Statistics*, 4(6):113–122, 1996.

[6] B. A. Dobson, A.J. *An Introduction to Generalized Linear Models*. Chapman and Hall, London, 2008.

[7] R. O. Duda and P. E. Hart. Use of the hough transformation to detect lines and curves in pictures. *Commun. ACM*, 15(1):11–15, 1972.

[8] S. Garg, J. Nam, I. Ramakrishnan, and K. Mueller. Model-driven visual analytics. *IEEE Symposium on Proc. VAST '08.*, pages 19–26, Oct. 2008.

[9] T. B. Ho and T. D. Nguyen. Visualization support for user-centered model selection in knowledge discovery in databases. *Proceedings of the 13th International Conference on Tools with Artificial Intelligence*, pages 228–235, Nov 2001.

[10] A. Inselberg and B. Dimsdale. Parallel coordinates: a tool for visualizing multi-dimensional geometry. In *VIS '90: Proceedings of the 1st Conference on Visualization '90*, pages 361–378.

[11] X. R. Li, V. P. Jilkov, and J. Ru. Multiple-model estimation with variable structure. *IEEE Trans. Automatic Control*, 41:478–493, 1996.

[12] A. Madansky. The fitting of straight lines when both variables are subject to error. *Journal of the American Statistical Association*, 54(285):173–205, 1959.

[13] Minnesota Department of Transportation. Mn/DOT traveler information. http://www.dot.state.mn.us/tmc/trafficinfo/index.html/, accessed on Feb. 16, 2009.

[14] E. Moura and D. G. Henderson. *Experiencing geometry: on plane and sphere*. Prentice Hall, 1996.

[15] E. J. Nam, Y. Han, K. Mueller, A. Zelenyuk, and D. Imre. Clustersculptor: A visual analytics tool for high-dimensional data. *IEEE Symposium on Proc. VAST 2007.*, pages 75–82, 30 2007-Nov. 1 2007.

[16] P. J. Rousseeuw. Least median of squares regression. *Journal of the American Statistical Association*, 79(388):871–880, 1984.

[17] A. Savikhin, R. Maciejewski, and D. Ebert. Applied visual analytics for economic decision-making. *IEEE Symposium on Proc. VAST '08.*, pages 107–114, Oct. 2008.

[18] T. Schreck, J. Bernard, T. Tekusova, and J. Kohlhammer. Visual cluster analysis of trajectory data with interactive kohonen maps. In *IEEE Symposium on VAST '08.*, pages 3–10, 2008.

[19] D. F. Swayne, D. T. Lang, A. Buja, and D. Cook. Ggobi: evolving from xgobi into an extensible framework for interactive data visualization. *Computational statistics and data analysis*, 43(4):423–444, 2003.

[20] P. Wong. Visual data mining. *IEEE Computer Graphics and Applications*, 19(5):20–21, 1999.

[21] D. Yang, E. A. Rundensteiner, and M. O. Ward. Analysis guided visual exploration of multivariate data. In *IEEE Symposium on Proc. VAST 2007.*, pages 83–90, 2007.