Data Visualization for Twitter Sentiment Analysis by Denali

CSE-5559 Project

By

Zhenyu Wu

The Ohio State University

2016

Instructor: Yusu Wang

Abstract

Using twitter data, I introduce a novel machine learning approach to analyze the sentiment influence of public figures' behavior or speech on twitter users. As is expected, there are extremely high dimensional data in this project, such as the sparse feature vectors and the parameter space for the Deep Neural Network. These data are topology space with scalar function defined on it. Some tree structures can be extracted from it. With the help of Denali, the high dimensional data can be visualized. My main contributions are: (1) Embedding tweets from text samples to feature vectors (2) Visualizing sentiment coexisting by extracting contour tree from feature vectors topology space (3) Visualizing the impact of metrics by extracting hierarchical clustering tree from feature vectors topology space (4) Visualizing cost function by extracting contour tree from the DNN parameter space (5) Writing callback function to visualize DNN performance

Keywords: data visualization, Denali, contour tree, hierarchical clustering tree, scalar function, feature vector, Deep Neutral Network, sentiment coexistence.

Introduction

1. Background

Twitter is a very popular social network platform. It provides an important site for expressing political opinions throughout the world. In the year leading up to the June 2016 Republican Party (GOP) presidential primaries in the United States, a tool for real-time analysis of sentiment expressed through Twitter towards presidential candidates has been developed. Sentiment is classified into 3 polarities, positive, negative, neutral. The steps can be summarized as data collection, preprocessing, feature extraction, and computing sentiment score by a 3-parellel-DNN architecture for each sentiment polarity. After feature extraction, each tweet text sample is represented by a high dimensional feature vector. Moreover, the 3-parell-DNN architecture produce an extremely high dimensional parameter space. Luckily, tree structures like contour tree [1] or hierarchical clustering tree [2] can be extracted from the data set. With the help of Denali, these high dimensional data can be visualized, thus giving us some insight into the topological structure of the data.

2. Overview

In this project, I am doing data visualization for the high-dimensional data like sparse feature vectors and parameter space for deep neural network. Hierarchical clustering tree is extracted from the feature vectors by applying average-linkage algorithm. It is expected to see 3 big clusters since the data set is made up of tweets labeled with 3 sentiment polarities, which are positive, negative and neural. Also, contour tree is extracted from the feature vectors to validate the sentiment coexistence assumption. The 2 hidden layer deep neural network has a parameter space, which is extremely high dimensional. With cost function defined on the parameter space, it can be visualized by extracting contour tree. It is expected to see under-fitting and over-fitting phenomenon.

3. Outlines

The rest of the paper is organized as follows. In section 2, a brief introduction to Denali is given. In section 3, details are given about the two high-dimensional data set. In section 4, experiment setup and results are presented. I make conclusion and give future works of project in section 5.

Denali Introduction

Denali is a tool for visualizing trees as landscape metaphors. Denali allow data sets that is cumbersome to visualize as graphs in the plane to be visually parsed and manipulated in an interactive and intuitive manner. In order to be visualized, the data set has to meet two prerequisites. First, each data point must have an associated scalar value. Second, it must be able to extracted some tree-like structure, usually hierarchical. There are many ways to extract hierarchical tree structure from the data. Computing contour tree is one approach while hierarchical clustering tree is another approach [3].

Data Set Introduction

1. Raw Twitter Data

In this project, a labeled dataset with 1,288 "tweets" is provided by the Kaggle website [5]. All of these "tweets" are related with Donald Trump, which is the most controversial and leading the topic on twitter. An example of the "tweet" information is shown in Table1.

Table 1 Example for Labeled Data

Tweets	Sentiment	Sentiment Confidence
@JGreenDC @realDonaldTrump In all fairness #BillClinton owns that phrase.#GOPDebate	Negative	0.6332
@MsPackyetti: Donald Trump's campaign reveals 1 important thing: Twitter Trolls are real people.	Positive	0.6957

2. Representing Feature Vector in Feature Space

The features collection in the training set of tweets can induce a feature space. After removing noise, a tweet text sample can be embedded as a feature vector. The number of occurrence of a single word in the list is taken into account. Therefore, every single word occurred in the list is represented as a positive integer, which is its number of occurrence, at a specific position in the feature vector. Obviously, the feature vector is very sparse with the dimension as high as 3198.

Table 2 Example for Feature Vectors

Tweets	"RT@TrumpIssues #GOPDebate #2016Debate elect Trump, elect Trump! Only he and @SarahPalinUSA can save America!"										
Feature	…	america	…	elect	…	gopdebate	…	save	…	trump	…
#Occurrence		1	0	2	0	1	0	1	0	2	0

3. Parameter Space induced from a 3-Parallel-DNN Architecture

According to Williams, P.[4], human emotions, even the contradicting ones like sad and happy, may mutually co-exist at the same time. Thus, it is assumed that different sentiment polarities can coexist in one tweet. That is to say, each tweet has a 3-D sentiment polarity score tuple, which represents positive, negative and neutral score respectively. The sentiment polarity having dominant score is used as the label of the tweet.

As is shown in Figure 1-1, a 3-parallel and identically structured DNN architecture is used to deals with the 3 sentiment polarities. Using different number of epochs, different configurations of architecture are obtained, thus inducing a parameter space. A DNN with 2 hidden layers with dimension of (3198,100) and (100, 50) is shown in Figure 1-2. The configuration of this architecture contains $3 \times (3199 * 100 + 101 * 50 + 51) = 3{\times}325001 = 975003$ parameters, which is very high dimensional.

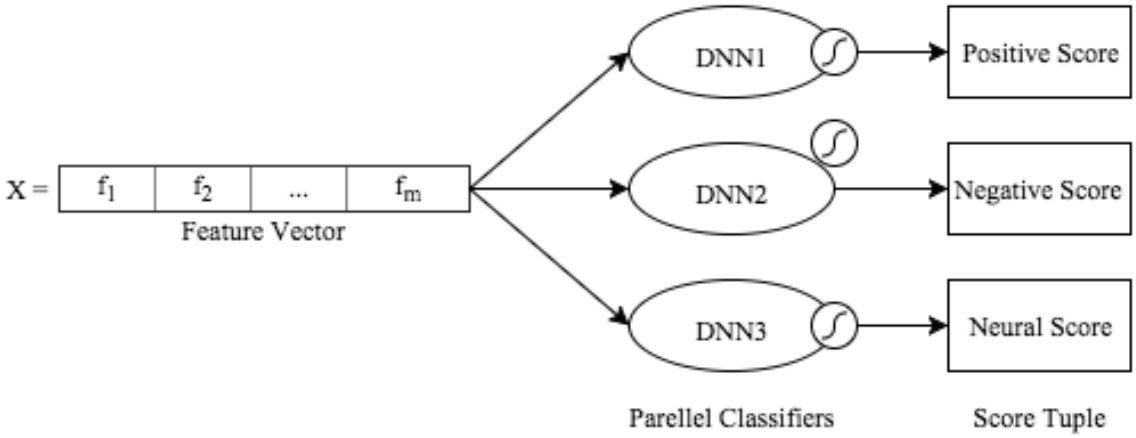

Figure 1-1 3-Parallel-DNN Architecture

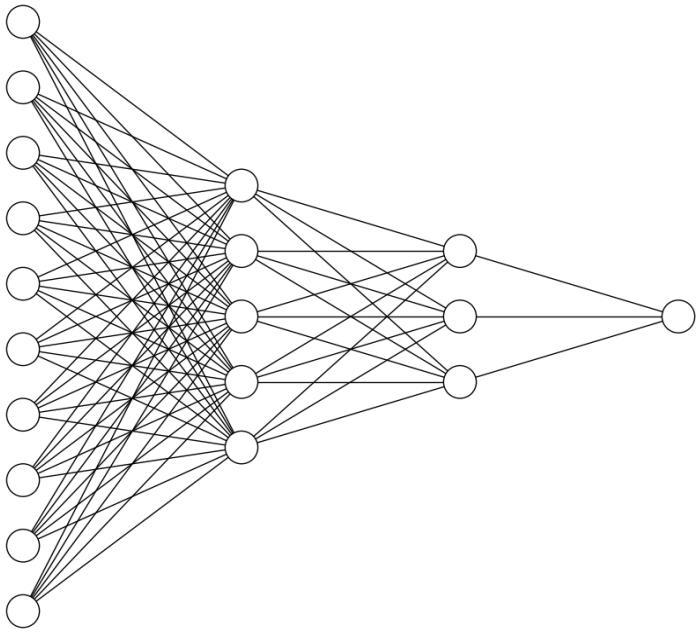

Figure 1-2 Deep Neural Network Illustration

Experiment Setups and Results

1. Visualizing Contour Tree Extracted from Feature Vectors

Given a topological space X and function $f: X \to R$, a contour tree can be extracted. The topological space is 1,288 feature vectors of dimension 3198. The scalar function is the variance for the score of 3 sentiment polarities of a tweet. The range of the score for each sentiment polarity is $[0, 1]$. The range of the scalar function is [0.0042, 0.2222]. It can be seen from the contour tree that the sentiment is coexisting for the majority of tweets.

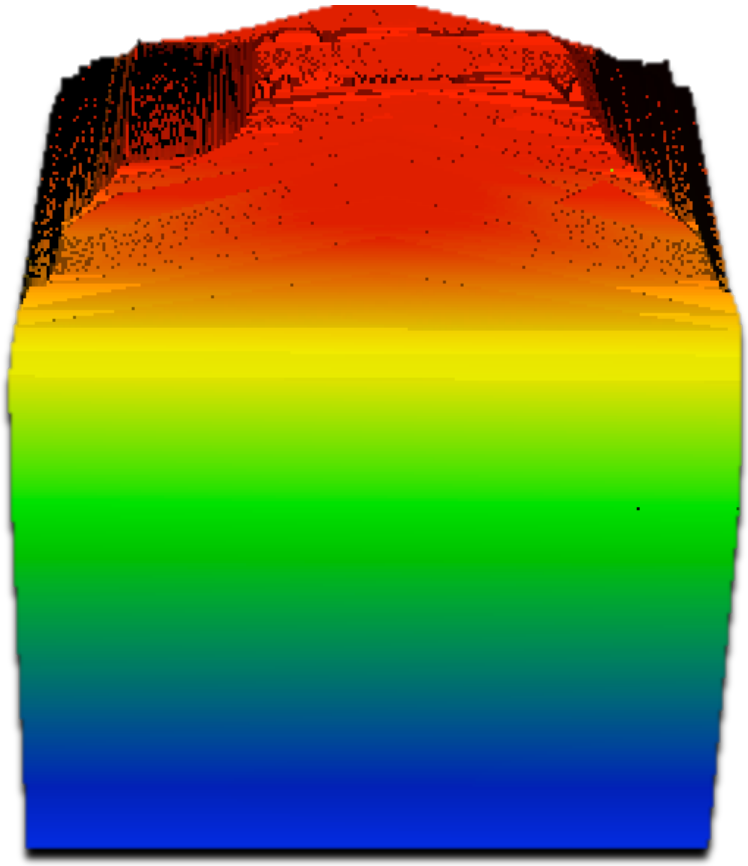

Figure 3 Contour Tree Extracted from Feature Vectors

2. Visualizing Hierarchical Clustering Tree Extracted from Feature Vectors

After applying average linkage clustering algorithm, a hierarchical clustering tree is extracted base on the dissimilarity between feature vectors. It is born with scalar function defined, which is the distance at which the clusters merge. It is expected to see 3 big clusters since the data set is made up of tweets labeled by 3 different polarities. The dissimilarity can be based on different metrics, like “Euclidean”, “Manhattan”, “Hamming” and “Cosine”. There is no much different among the HCT derived from Hamming distance and Euclidean. It reveals that for feature vector representation of the tweet sample, the number of occurrence for each feature doesn’t have big sentiment influence. Therefore, we can simplify the feature vector by making it binary.

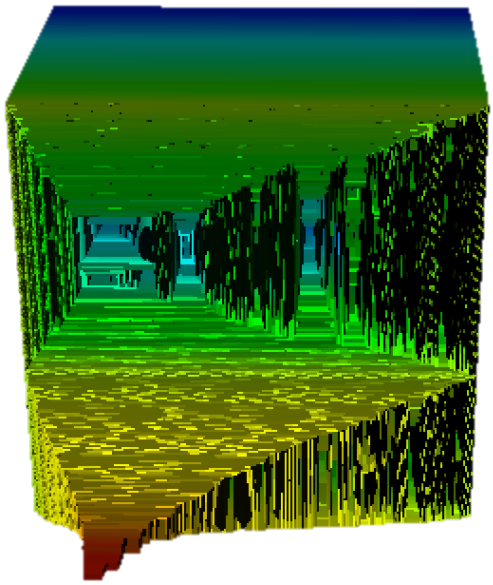

Figure 4 HCT (Euclidean)

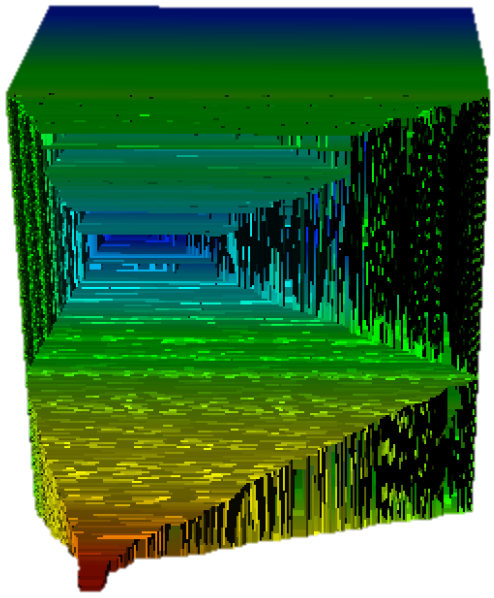

Figure 5 HCT (Manhattan)

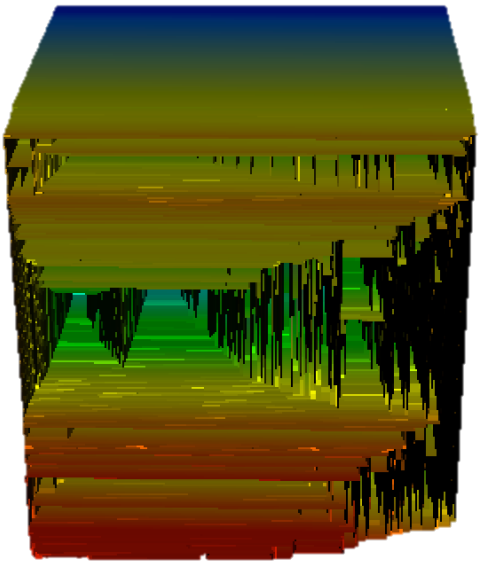

Figure 6 HCT (Cosine)

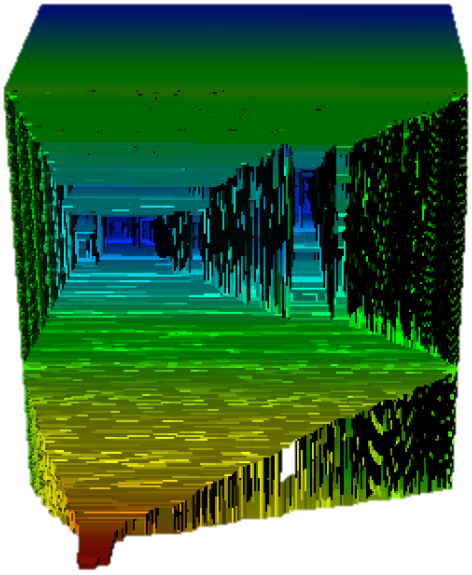

Figure 7 HCT (Hamming)

3. Visualizing the parameter space induced by 3-parallel-DNN Architecture

With different number of epochs, the 3-parallel DNN architecture induces parameter space. The cost function is defined on the parameter space. Thus a contour tree can be extracted from it. The contour tree is expected to show the phenomenon of overfitting and underfitting. Fig 7 shows the contour tree, with the scalar value of each node and member being the testing error. Fig 8 shows the same contour tree but with a training error as color map. There are two plateaus in the contour tree. Both of them indicated high testing error. But one is resulted from overfitting while the other one is resulted from underfitting, which can be inferred from the color map. As is expected, too many epochs for Deep Neural Network tend to overfit and too few epochs tend to underfit.

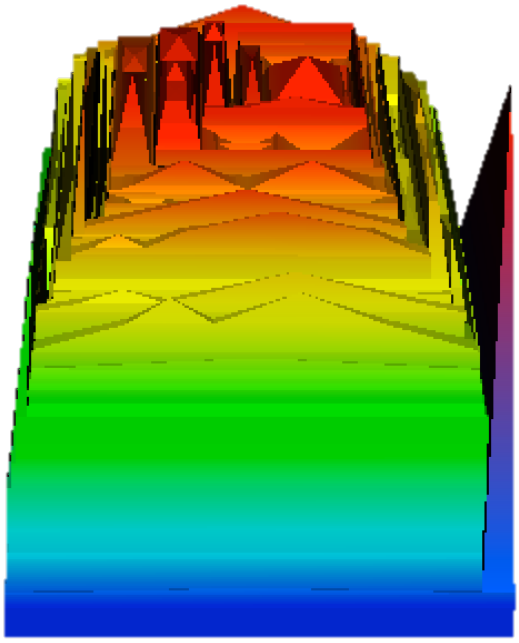

Fig 8 Contour Tree for Testing Error on Parameter Space

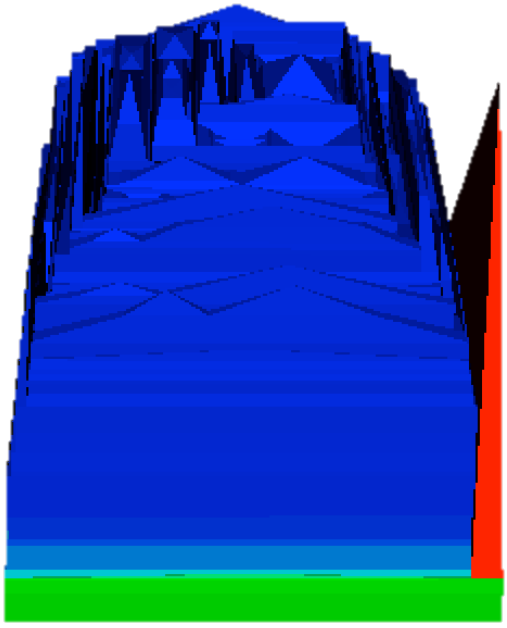

Fig 9 Contour Tree Configured with Color Map by Training Error

I write a call back function to visualize the performance for each configuration of the 3-parallel-DNN architecture. It is working asynchronously once a component is selected. It plots two figures to compare the training error and testing error by partitioning the data (1288 vectors) into training (1025 vectors) and testing (263). For the ease of visualization, I am doing principle component analysis here by reducing the 3198 dimensional data to 3 dimensional data. The misclassified points are drawn in green color while the correctly classified are drawn in red.

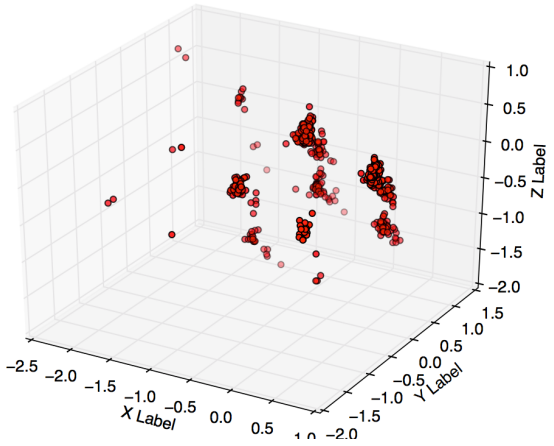

Fig 10 Illustration of Overfitting (Training Error)

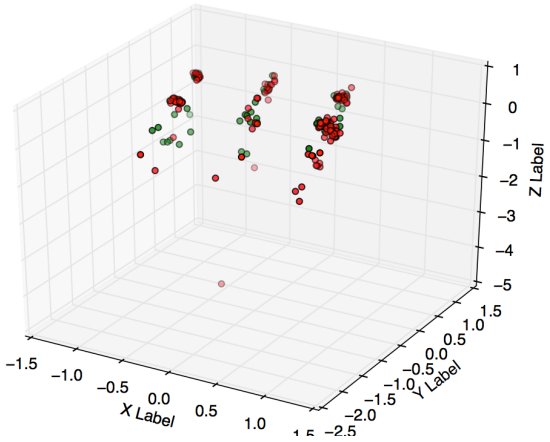

Fig 11 Illustration of Overfitting (Testing Error)

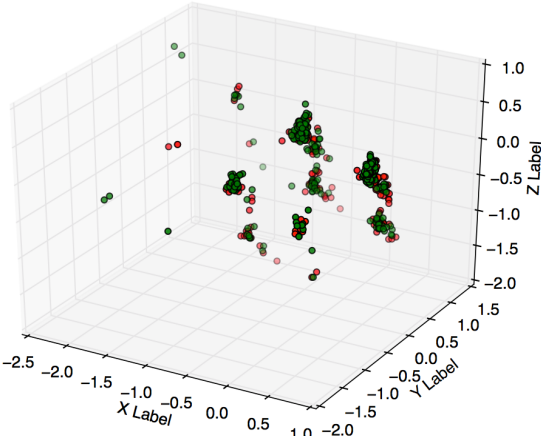

Fig 12 Illustration of Underfitting (Training Error)

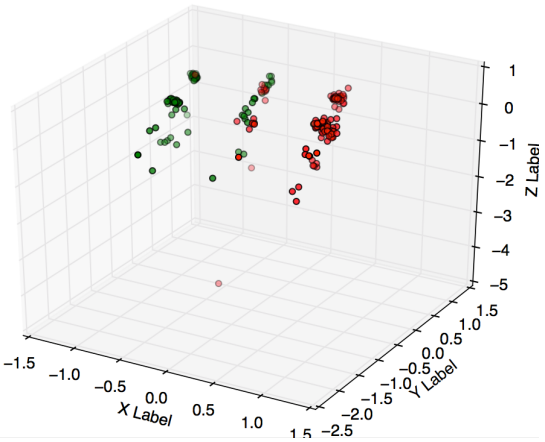

Fig 13 Illustration of Underfitting (Testing Error)

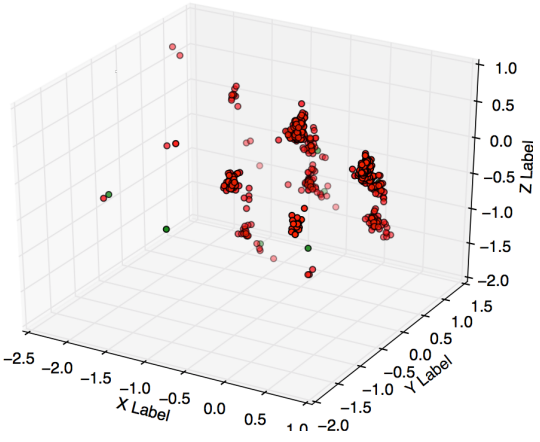

Fig 14 Illustration of Optimal Performance (Training Error)

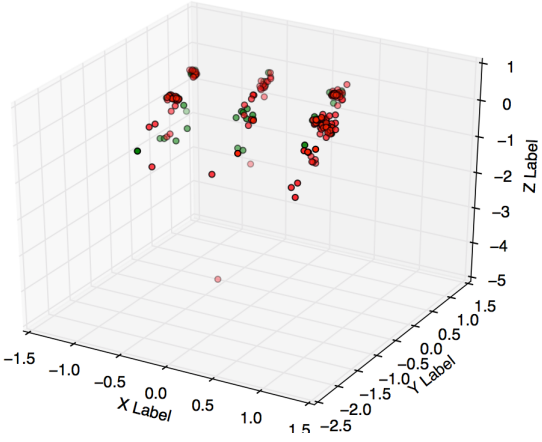

Fig 15 Illustration of Optimal Performance (Testing Error)

Conclusion and Future Works

1. Conclusion

In this project, I am mainly focusing on doing data visualization on the high dimensional data. For the feature vectors, by defining scalar function being the variance among sentiment polarity scores, I extracted contour tree from the data. After applying average linkage algorithm, a hierarchical clustering tree has been extracted. Using different dissimilarity metrics, different trees are obtained. For the parameter space induced by the 3-parallel-DNN architecture, it can be extracted contour tree by using the test error function as scalar function. What's more, configuring a color map using the training error gives some good visualization, which reveal the overfitting and underfitting very intuitively.

2. Future Works

2.1. Try More Hierarchical Clustering Algorithm and Do Simplification

I am using average linkage clustering to get the hierarchical tree. To my disappointment, the HCT doesn't give any interesting visualizations. Literature says that robust single linkage algorithm is more suitable for extremely sparse data set. Also, it is worth the try simplifying the tree structure to reduce the impact of noisy data.

2.2. Utilizing More Features in Denali

Denali is a very powerful tool for data visualization. There are still features not being utilized. Some features like weight mapping and resampling callback function should be helpful for emphasizing the important portion of data set.

Reference

1. Yusu, Wang. Topic 8: Reed Graphs and Contour Trees. Course slides. Apr 2016.
2. Yusu, Wang. Topic 9: Hierarchical Clustering Tree. Course slides, Apr 2016.
3. Justin E, Mikhail B, Yusu, W. Denali: A tool for visualizing scalar functions as landscape metaphors. Sep 21, 2014
4. Williams, P., and Aaker, J. L. 2002. Can Mixed Emotions Peacefully Coexist? Journal of Consumer Research, pp. 636–649, Apr 28, 2002.
5. GOP Debate Twitter Sentiment. Analyze tweets on the first 2016 GOP Presidential Debate: https://www.kaggle.com/crowdflower/first-gop-debate-twitter-sentiment.