Noname manuscript No.
(will be inserted by the editor)

Mining Latent Patterns in *geoMobile* Data via *EPIC*

Arvind Narayanan* · Saurabh Verma* · Zhi-Li Zhang

Received: date / Accepted: date

Abstract We coin the term *geoMobile* data to emphasize datasets that exhibit geospatial features reflective of human behaviors. We propose and develop an *EPIC* framework to mine latent patterns from geoMobile data and provide meaningful interpretations: we first *'E'xtract* latent features from high dimensional geoMobile datasets via Laplacian Eigenmaps and perform clustering in this latent feature space; we then use a state-of-the-art visualization technique to *'P'roject* these latent features into 2D space; and finally we obtain meaningful *'I'nterpretations* by *'C'ulling* cluster-specific significant feature-set. We illustrate that the local space contraction property of our approach is most superior than other major dimension reduction techniques. Using diverse real-world geoMobile datasets, we show the efficacy of our framework via three case studies.

Keywords geoMobile · data mining · latent patterns · epic · regional patterns · feature distributions

1 Introduction

The wide proliferation of various kinds of (physical or virtual) sensors in the physical and/or cyber worlds has enabled us to collect a whole gamut of (spatial-temporal) data, e.g., voice calls between users at various locations in a cellular network, human

Arvind Narayanan
Department of Computer Science & Engineering
University of Minnesota, United States E-mail: arvind@cs.umn.edu

Saurabh Verma
Department of Computer Science & Engineering
University of Minnesota, United States E-mail: verma@cs.umn.edu

Zhi-Li Zhang
Department of Computer Science & Engineering
University of Minnesota, United States E-mail: zhzhang@cs.umn.edu

* These authors contributed equally to this work.

commuting behaviors across different locations in a transport network such as buses, subways, taxicabs or car sharing services, check-ins and social interactions among users at diverse locations in a location-based online social network such as Foursquare. We use the term *geoMobile* to refer to such datasets collected from these networks, as they are characterized with two salient features: they are associated with geo-locations (e.g., gathered at cell towers or tagged with location information), and more often they capture user actions on-the-move.

With abundance of diverse geoMobile datasets, mining them is an important activity that has wide applications, from cellular network traffic engineering to urban transportation management, smart city planning, social behavior analysis and cyber-physical world security. For example, one can ask questions such as: can geo-locations and user actions (e.g., making phone calls) at these locations capture and reflect certain underlying community structures? In other words, can one classify regions into various communities based on their associated human-actions at certain geo-locations? More broadly, how user mobility and behavior are associated with geo-locations? Unfortunately, gleaning meaningful and actionable knowledge from geoMobile data are non-trivial. We list several reasons why mining geoMobile datasets is a challenging task. First, there is huge heterogeneity in (user) activities associated with different geo-locations, which leads to very skewed data distributions. This is partly due to the fact that there are often very disparate factors driving user mobility and behavior at various geo-locations; geoMobile data is thus likely to more closely mirror user relations and interactions in the real "physical" world (than mere the "cyber" world). Second, depending upon the spatial and temporal resolutions, geoMobile datasets are often high-dimensional. Underlying patterns, if present, may either be a linear or a non-linear combination of a varying subset of features. Therefore, judicious feature engineering is paramount. However, without prior knowledge of the problem domain coupled with high pattern diversity in geoMobile datasets, feature selection or extraction from high-dimensional data becomes difficult. Third, even once we have appropriate representative (or latent) features and can obtain clusters, it is hard to make sense out of the clusters without the aid of a visualization technique. Lastly, the factors which cause the formation of (latent) clusters in the feature space are not always easily understood. It is important to relate and map a cluster back to its "raw" feature set (rather than the latent feature set) that are critical to its formation. Such information can help to naturally interpret the results.

We combine some of the popular algorithms with state-of-the-art machine learning tools to develop a framework to extract, visualize and interpret latent patterns arising from geoMobile datasets. We address the challenges discussed earlier and summarize our central contributions as follows:

1) Instead of directly working on observed features, we take into account the feature distribution of every data point and derive a new (symmetric) *similarity* matrix. This amounts to transforming the data points into a high-dimensional feature space. We apply the Laplacian Eigenmap (LE) method to *extract* latent features and "clusters" data points lying in certain lower-dimensional (non-linear) *sub-manifolds* (see § 3.1).

2) We show that a state of the art visualization technique t-SNE [van der Maaten and Hinton(2008)] is a *density preserving* algorithm. This provides a theoretical justification for its success in practice. To get insights about the structure of geoMobile data and visualize clusters in the feature space, we further *project* latent features into a 2-dimensional space using *Lt-SNE* – a proposed approach that uses t-SNE in conjunction with LE which is an improvement over standard t-SNE (see § 3.2). To show the effectiveness of Lt-SNE, we provide justification by studying and comparing its *local space contraction property* with other prominent dimension reduction techniques (see § 4.2).

3) Taking cue from *information theory*, we supplement our framework by designing an algorithm to further *cull* a set of raw (i.e., observable) features that are most significant in contributing to the cluster's formation (see § 3.3) so as to obtain meaningful *interpretations* of extracted latent patterns.

4) We evaluate our framework based on the performance of its individual components specifically clustering and visualization component (see § 4) and show its empirical superiority over other state-of-art baselines.

5) Finally to demonstrate the efficacy and generality of our proposed framework in real world, we share our experience of analyzing geoMobile datasets under multiple settings using several case studies (see § 5). We employ two real-world geoMobile datasets: i) a mobile call detail record (CDR) dataset consisting of more than 500 million voice calls and SMS messages between users collected at cell-tower levels spanning a couple of months from a nation-wide cellular service provider in Africa, and ii) a subway transit dataset collected over a week from Shenzhen, China with more than 2.7 million passengers. Despite very different nature of these two datasets, the results look promising.

1.1 Related Work

In literature, there exist multiple methods to extract latent patterns from geoMobile-based datasets. One of the classical approach is principal component analysis (PCA). PCA-based methods have been successfully applied to traffic matrix estimation, network tomography and anomaly detection [Wang et al(2012)Wang, Hu, Xu et al, Zhang et al(2005)Zhang, Ge, Greenberg, and Roughan, Lakhina et al(2004)Lakhina, Crovella, and Diot] using origin-destination (OD) matrices derived from Internet traffic. As discussed earlier, user actions and behavior are often driven by disparate factors leading to high diversity and skewed data distributions in geoMobile datasets rendering classical *linear* methods such as PCA and latent semantic indexing (LSI) ineffective. [Hristova et al(2016)Hristova, Williams, Musolesi et al] further provides a detailed analysis of measuring social diversities from mobility datasets and reveals the large diverse nature of such datasets. Another matrix factorization approach is *non-negative matrix factorization* NMF (e.g., [Zhang et al(2014)Zhang, Huang, Li et al]) developed to address the *interpretability* issue associated with the low-rank matrix approximations. A fundamental premise of NMF is that the entities lies in lower *linear subspaces* of the original higher-dimensional matrix which may not hold for geoMobile datasets. [Fan et al(2014)Fan, Song, and Shibasaki, Zhang et al(2013)Zhang, Wilkie, Zheng, and Xie] adopt tensor factorization, the generalization of NMF, to study city basic life pattern and analyze urban transportation.

Many other methods such as hidden Markov models (HMMs) and Gaussian mixtures models (GMMs) have also been developed to analyze and predict urban dynamics [Witayangkurn et al(2013)Witayangkurn, Horanont, Sekimoto et al, Ihler and Smyth(2006), Baratchi et al(2014)Baratchi, Meratnia, Havinga et al]. Unfortunately, inference in HMMs and GMMs suffer severe performance degradation in the high-dimensional setting due to overfitting and constraints (e.g. covariance matrices should have simple structure, say diagonal). These models have large number of free parameters that lead EM algorithm to converge to poor clustering results [Krishnamurthy(2011), Städler and Mukherjee(2013)].

Latent Dirichlet Allocation (LDA) models are also employed for extracting latent patterns for tasks such as identifying regions of different functions in urban areas and urban topic analysis [Yuan et al(2012)Yuan, Zheng, and Xie, Kling and Pozdnoukhov(2012)]. In general, LDA models have the capability to handle high dimensional data, however choice of hyper-parameters is not apparent [Wallach et al(2009)Wallach, Mimno, and McCallum] and relies upon approximate inference algorithms such as Gibbs sampling for efficiency.

Deep learning frameworks such as discussed in [Lv et al(2015)Lv, Duan, Kang, Li, and Wang, Alsheikh et al(2016)Alsheikh, Niyato, Lin, p. Tan, and Han] have also been developed to extract latent features and can be seen as a complement to our work. But we go beyond to include visualization and interpretation as an important step for aiding and justifying the data analysis part and provide theoretical and empirical evaluations with respect to other popular techniques.

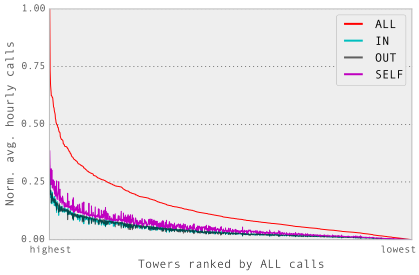
2 geoMobile Datasets & its Characteristics

In this paper, we primarily focus on two geoMobile datasets representing different application domains; 1) a mobile call detail record (CDR) dataset collected from a nationwide cellular network; and 2) a subway transit record dataset from a large city in China. We provide the description of the datasets and show the diversity in patterns inherent in them. Note, the nature and user population of both datasets are completely different from each other.

2.1 Dataset 1: Mobile Call Detail Records

Dataset 1 is a call detail record (CDR) dataset that comes from a national cellular service provider of a developing African nation. Every record of this dataset contains information such as `<timestamp, source base station, destination base station>` associated with a voice call or a SMS message (both of which we will refer to as *calls* in this paper). The dataset spans over a couple of months. This dataset consists of over 1,000 towers (or base stations) covering the entire country, with over 500 million call records.

We refer to a cellular base station as a *tower*. When Bob (*caller*), connected to tower A, makes a call to Alice (*callee*) who is connected to tower B, tower A is the *origin* tower, whereas tower B is the *destination* tower. In other words, this call will be considered as an *outgoing* call for tower A, and an *incoming* call for tower B. However, if both Alice and Bob are connected to the same tower C, i.e., both the origin and destination towers are the same, then we refer to such a call as a "local[1]" call. Geographic coordinates of all the towers are known *a priori*.



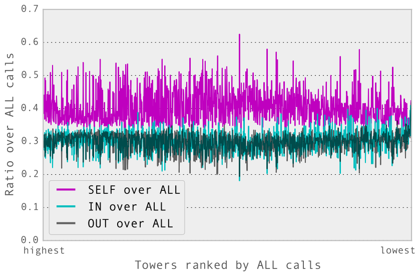


Fig. 1. Distribution of ALL, IN, OUT, and SELF calls, with a *fixed* order of towers (x-axis) ranked by the ALL calls (i.e. total number of calls).

Fig. 2. IN, OUT, and SELF call ratio distributions with a *fixed* order of towers (x-axis) ranked by the ALL calls (same as in Figure 1).

Terminologies: We use *call direction* to define four aggregated metrics associated with every tower *i*, 1) **SELF calls:** the total number of *local* calls for tower *i*, 2) **IN calls:** the total number of incoming calls received by tower *i* excluding SELF calls, 3) **OUT calls:** the total number of outgoing calls made by tower *i* excluding SELF calls, and, 4) **ALL calls:** the total number of calls seen at tower *i* (IN + OUT + SELF).

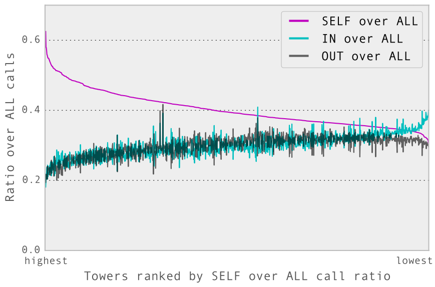


Fig. 3. Plot similar to Figure 2, only difference being, the x-axis is now ranked by SELF over ALL ratio.

Dataset Characteristics: In Figure 1, we fix the rank of the towers based on the average number of ALL calls seen per hour, and plot their distributions based on the total volume (ALL calls) as well as the call directions, IN, OUT, and SELF calls. We see that the distributions are highly skewed, with call volumes varying significantly among towers. Some cell towers experience significantlly more calls (either ALL, IN,

[1] Although calls involving two neighboring towers semantically qualify to be *local*, our reference of a call being *local* is solely from the tower's perspective where both the *caller* and *callee* of a call are associated with the same tower.

OUT or SELF calls) than others. Due to their larger population size, we would expect that *as a whole*, towers in urban cities would have higher call volume than the towers located in rural areas. While the capital city of this nation captures more than 25% of the entire call volume, we observe that the towers with the highest ALL calls are not just from the capital city but also from some of the tier-2 cities of the nation. Moreover, we also observe certain towers in the city do not have high ALL call volumes at all. For each individual cell tower (especially those with high call volumes), we also see that there are high variances in terms of calls to or from other cell towers; there are no discernible patterns across cell towers, suggesting there is high diversity among cell towers.

We now investigate the proportions of SELF, IN and OUT calls over ALL calls at the towers. In Figure 2, we fix the rank of towers the same as in Figure 1 and plot the distributions of call proportions – SELF over ALL (% of *local* calls), IN over ALL (% of incoming calls), and OUT over ALL (% of outgoing calls). We observe that in general SELF over ALL call ratios dominate compared to IN over ALL and OUT over ALL call ratios, implying people tend to make more SELF calls than IN or OUT calls. However, Figures 1 and 2 show no clear linear relationship between call volume distributions and call proportion distributions. To further investigate, we fix the rank of the towers based on SELF over ALL call ratio (decreasing order), and plot all the call ratio distributions (see Figure 3). We observe there is still high variance in the call proportions. For example, the SELF over ALL call proportions vary between 30% to 55%. This implies certain towers tend to make more SELF calls than others.

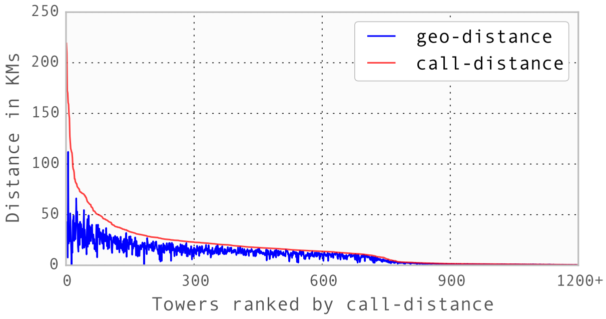


Fig. 4. Relation between (geo-distance) and (call-distance) of towers.

Diversity in Locality Effects: For every tower i in this dataset, we find its $\mathbb{N}$ geographically closest (or neighboring) set of towers G_i and compute tower i's geo-distance defined as: $gd_i = \sum_{j \in G_i} dist(i, j) \big/ |G_i|$, where $dist(i, j)$ is the geographic distance in kilometers (KM) between towers i and j. Similarly, we identify another $\mathbb{N}$ set of towers C_i with whom tower i communicates the most (or makes the most numbers of calls to). We then compute tower i's call-distance defined as: $cd_i = \sum_{j \in C_i} dist(i, j) \big/ |C_i|$, where $dist(i, j)$ has the same semantics as before. To compute geographic distance between towers using their geographic coordinates (which is known *a priori*), we use the *Haversine* formula [Robusto(1957)]. In this paper, value of k is set to be 5. We compute both *gd* and *cd* for all the towers, and show the results in Figure 4. We can clearly see that overall there is slight correlation between both *gd* and *cd* values of towers. In other words, majority of the localities (or towers) tend to make more calls to towers that are geographically closer to them, there by exhibiting certain *"locality"* effect. However, as seen in the plot, there is high *diversity* in such *locality effects*. While this maybe a side effect to choosing ℕ=5, our objective was to show the diversity in these relations. All in all, this gives us an intuition of the existence of certain communities of people (i.e. collection of towers) that tend to talk with each other more than others. However, as evident from the plot, geographic distances between such towers vary significantly. Later, we describe an approach (see § 3) to identify such communities and show the results obtained in the form of a case study (see § 5.1).

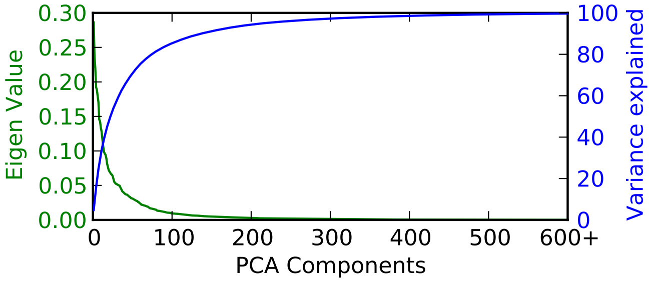


Fig. 5. Results of PCA from Dataset 1

All of these observations suggest that there are strong *dependency* relations between call volumes at tower levels, human activities of either "local" and "mobile" users around these towers. However, these relations are highly varied and *non-linear*, as evidenced by the eigenvalue plot of a call-volume based origin-destination (OD) matrix derived from the CDR dataset shown in Figure 5. We see that eigenvalues decrease slowly, requiring more than 100 eigenvalues to account for 90% of the variance in the OD matrix. This indicates that PCA is ill-suited for extracting patterns inherent in the OD matrix.

2.2 Dataset 2: Subway Transit Data

Our second dataset represents commute patterns in a subway transit system in Shenzhen, China. The dataset contains information such as – (`timestamp`, `smart card ID`, `direction` (entry/exit), `station ID`), collected for an entire week in March 2014. More than 2.7 million users traveled over this period. Shenzhen Metro has 5 subway lines (see Figure 15) comprising a total of 118 stations.
Data Preprocessing & Categorization: We first construct trips by using the *direction* field of the record – `ENTRY` indicates a user entering the station while `EXIT` indicates leaving. We match an `ENTRY` record with an `EXIT` record for the same user if both the records satisfy the following three conditions, 1) both records should have occurred

on the same day, 2) ENTRY *timestamp* is earlier than EXIT *timestamp*, 3) if there are multiple EXIT records, then we consider the one with the earliest timestamp. Matching user-specific ENTRY with EXIT records helped construct trip information. Next, we categorize users as *regular* or *adhoc*. A user is labeled as a *regular user* if they satisfy the following two conditions: 1) seen on all the working days of the week (Mon. to Fri.), 2) take at least 2 trips per day. We consider regular users to be of the working class population who use the subway system for their everyday commute between home and work. Finally, we consider users to be as *adhoc* if they are seen for not more than a day or if they just had a single trip for the entire week. We assume that the *adhoc* users are either *visitors* or users who take random trips. All remaining users were excluded from our study. More than 10% of the users were categorized as regular and ~80% as adhoc users. For gaining insights into temporal patterns, we create three intervals of 2 hours representing different periods of the day: 1) Morning: 7-9am, 2) Mid-day: 11am-12pm, 3) Evening: 3-5pm. We label records that fell in these intervals and excluded the rest from our analysis.

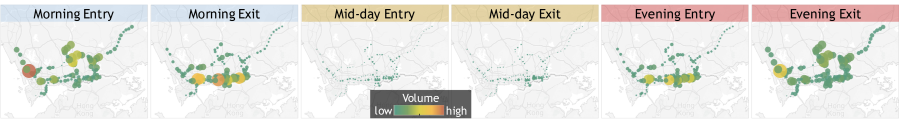


Fig. 6. Effect of *time-of-day* over travel patterns. Bubble size is proportional to the volume of users entering (or exiting) a station.

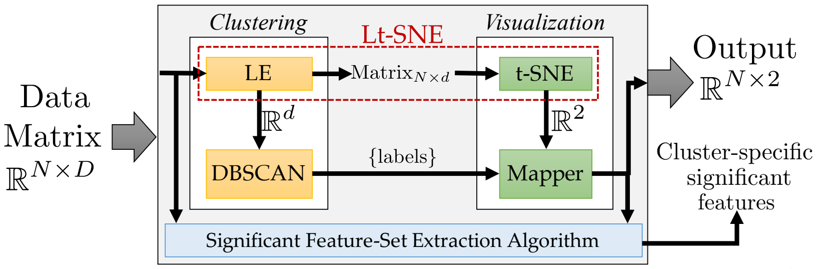
Dataset Characteristics: There is high diversity in the traveling patterns across stations; the distributions of number of passengers boarding and alighting from subway trains at each station are also skewed. We use the subway transit dataset to illustrate the temporal variations and latent patterns therein. Figure 6 is a bubble map showing volume of *regular* users entering and leaving the subway stations at different time intervals of the day. For example, during the morning rush hours, we see a large number of commuters entering at certain stations and exit in and around downtown area. This suggests, such stations with higher volume of entries correspond to residential areas, i.e. people board the train to go for work. An opposite pattern is seen during the evening rush hours. Volume of traffic during mid-day hours is drastically low compared to the rush hours. However, there are certain subway stations that have relatively higher volume of users entering and leaving. Thus, we observe diversity in the trip patterns seen over time between subway stations.

In summary, we observe both datasets contain highly diverse and skewed data distributions, rendering classical linear dimension reduction or clustering techniques ineffective. As characteristics differ depending upon what dataset is being analyzed, it is nontrivial to come up with a formal definition for diversity. Nonetheless, we assume there are latent factors driving human mobility and user behavior across various geo-locations and over time, as suggested by certain "locality" and "time-of-day" effects. For instance, peak usage hours of public transit systems depend on the general working hours associated to that region. In a similar vein, humans are more likely to interact with others (e.g. by making a voice call) who reside within their local community.

Extracting meaningful (latent) patterns from such geoMobile datasets requires us to go beyond classical linear methods to effectively account for the inherent high variability and diversity (thus strong *non-linearity*). In the next section we present such an approach.

3 *EPIC* Framework

geoMobile datasets (e.g. human mobility data) are rich in both spatial and temporal aspects. The underlying structures of such datasets are complex but can be understood better via deriving sets of latent features from observable features such as traffic volume density, peak traffic hours etc. However, it is quite likely that these sets of latent features give rise to low-dimensional sub-manifolds forming various kinds of clusters in the data. This results in having each cluster formed by few latent factors which are a (nonlinear) function of the observable features. Based on this intuition, we are interested in an approach that find clusters while accounting for possible latent features in the data. For this purpose, we consider Laplacian Eigenmaps (LE) [Belkin and Niyogi(2003)] – a theoretically sound non-linear dimension reduction technique and provide justification about its suitability in our case. As mentioned earlier, standard clustering techniques such as K-means and linear dimension reduction techniques such as PCA or NMF are not appropriate, due to the *curse of high dimensions* and strong *non-linearity*, respectively in geoMobile data. Although, other non-linear dimension reduction techniques like Deep Autoencoders, Hessian Locally Linear Embedding, Local Tangent Space Alignment or Kernel PCA can also be employed but we show (theoretically as well empirically) that LE in conjunction with t-SNE technique produces superior visualization maps over these dimension reduction techniques based on interesting local space contraction properties in latent feature space. Figure 7 depicts a schematic overview of our framework.



$D \gg d \gg 2$; $N \rightarrow$ num. of data points; $D \rightarrow$ num. of features (or dimensions)

Fig. 7. Overview of *EPIC* Framework

3.1 Extracting Latent Features from geoMobile Datasets

We propose a *simple but effective* enhancement of LE. This enhancement comes from carefully accounting for skewed data density distribution while computing the similarity matrix. LE extracts the latent features associated with each data point $\mathbf{x} \in \mathbb{R}^D$, where D is a feature dimension, by performing eigenvalue decomposition of graph Laplacian $\mathbf{L}$. We execute the following carefully devised algorithm, so that standard clustering algorithms can be applied on the newly obtained features which are *then* free from curse of dimensionality and data density skewness.

Handling the Skewness: The most *crucial* component for computing $\mathbf{L}$ is similarity matrix $\mathbf{W}$. Since the set of features are large, we take the exponential of euclidean distance in feature space to counteract the curse of dimensionality. More precisely, we adopt the following form of Gaussian kernel:

$$\mathbf{W}_{ij} = \exp\left(-\|\mathbf{x}_i - \mathbf{x}_j\|^2 / 2\sigma_i^2\right) \tag{1}$$

which is suitable under this condition, though more theoretical motivation can be found in [Belkin and Niyogi(2003)]. $\mathbf{W}_{ij}$ can also be seen as a conditional probability $p_{j|i}$ of picking data point x_j as the neighbor of x_i.

In particular, σ is kept same of each data point but we *stress* on computing specific σ_i at each data point x_i to handle the *skewness* in the data density. We choose σ_i based on our belief that the entropy of density distribution $p(\mathbf{x}_i, \sigma_i)$ given as,

$$p(\mathbf{x}_i, \sigma_i) = -\sum_j \Big(p_{j|i} \log p_{j|i} \tag{2}$$

remains constant at each data point and equal to $\log k$. Here k is a user defined parameter which physically represents a smooth measure of effective number of neighbors. We finally perform a binary search over the value of σ_i which gives $\log k$ entropy for each data point. Turns out that the similarity matrix is robust for different values of k and its typical value lies in the range of $5 - 50$.

Justification: We adopt LE for two main reasons. First, it can handle non-linearity in the data as shown in [Belkin and Niyogi(2003)] and ensures that the new reduced latent features obtained are similar if their respective feature distributions are also similar. This can be confirmed by looking at the LE objective function given as:

$$\min_{\mathbf{y}} \frac{1}{2} \sum_i \sum_j \Big(\mathbf{W}_{ij} \|\mathbf{y}_i - \mathbf{y}_j\|^2 \tag{3}$$

subjected to scaling constraints, where $\mathbf{y}_i \in \mathbb{R}^d (d << D)$ is a new latent feature vector for $\mathbf{x}_i$ data point. Secondly, it helps in producing superior visualization maps (see § 3.2, § 4.2). LE enforces two similar data points $\mathbf{x}_i$ and $\mathbf{x}_j$ to have similar latent features according to the weight $\mathbf{W}_{ij}$ which itself depends upon the original feature distribution. For computing $\mathbf{L}$, we adopt a symmetric normalized graph Laplacian proposed in [Ng et al(2002)Ng, Jordan, and Weiss]:

$$\mathbf{L} = \mathfrak{D}^{-1/2} W \mathfrak{D}^{-1/2} \tag{4}$$

as it is less susceptible to bad clustering when different clusters are connected with varied degree. where $\mathfrak{D}$ is the diagonal degree matrix whose elements are the sum of rows of the similarity matrix. From eigen decomposition of $\mathbf{L}$, d largest eigenvectors are stacked as columns in a $\mathbf{Y}$ matrix which is renormalized to yield latent features of points projected on a hypersphere in $\mathbb{R}^d$. Graph Laplacian *implicitly* provides a way to estimate d by *examining drop in eigenvalues* of $\mathbf{L}$ but more approaches can be also found in [Manor and Perona(2005)]. For our datasets LE approach was sufficient enough to yield faithful results. We observed there is an eigenvalue drop (see Figure 8) with 15 components pointing to the existence of 15 intrinsic dimensions in OD matrix which earlier PCA could not estimate correctly (see Figure 5). We choose DBSCAN clustering algorithm to be applied on obtained latent features due to its robustness against outliers. Next, we present our powerful Lt-SNE visualization algorithm.

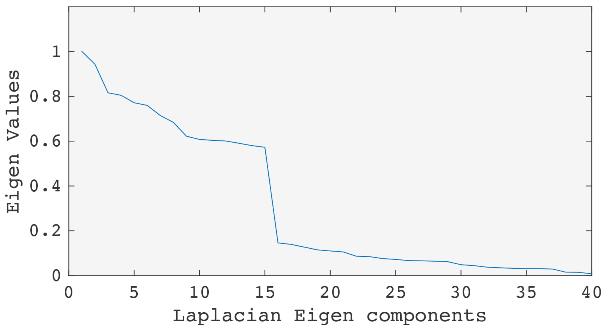


Fig. 8. Laplacian Eigenvalue Decomposition

3.2 Lt-SNE Visualization Algorithm

Density Preserving Maps: According to Gauss Theorema Egregium [Pressley(2010)], manifolds with *intrinsic curvature* cannot be mapped to the $\mathbb{R}^2$ plane (as it has zero Gaussian curvature) without distorting distances. However, no such obstruction exists for density preserving maps (see Moser Theorem [Ozakin et al(2011)Ozakin, II, and Gray]). Hence, we seek a method that *preserves* (probability) density maps rather than distances below intrinsic dimensions for visualization purpose.

Success of t-SNE: t-SNE [van der Maaten and Hinton(2008)] is a state-of-art technique for visualizing clusters inherent in the data by mapping latent features to $\mathbb{R}^2$ (or $\mathbb{R}^3$) space. But its theoretical justification remains somewhat a mystery. Here, we prove that the objective function of t-SNE upper bounds the loss function in kernel density estimation (KDE) (see Proposition 1). This makes t-SNE a *density preserving mapping* algorithm which provides a theoretical justification for its *success* as compared to other dimension reduction techniques which tend to preserve (geodesic) distances.

Proposition 1 *t-SNE is a density preserving algorithm which upper bounds the estimated kernel density loss function.*

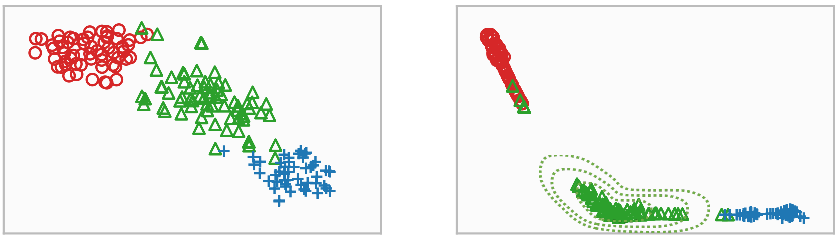
Proof: KDE is a non-parametric way to estimate probability density function; it leverages the chosen kernel in the input space for smooth estimation. Given sub-manifold density estimates $p(\mathbf{y}_i)$ for data points $\mathbf{y}_i \in \mathbb{R}^d\ \forall i$, we want to find a representation $\mathbf{z}_i \in \mathbb{R}^p\ \forall i$ such that the new density estimates $q(\mathbf{z}_i)$ agree with the original density estimates. Here K_H, and K_L denote the kernel in higher and lower dimension respectively, where h is the kernel bandwidth, $\mathbf{y} \in \mathbb{R}^d$, $\mathbf{z} \in \mathbb{R}^p$, $p < d$, and N is the number of data points. KDE in higher and lower dimensions (assuming bandwidth remains the same) are given by:

$$\begin{aligned} p(\mathbf{y}) &= \frac{1}{N}\sum_{j=1}^{N}\frac{1}{h^d}K_H\left(\frac{\|(\mathbf{y}-\mathbf{y}_j\|_d}{h}\right) \\ q(\mathbf{z}) &= \frac{1}{N}\sum_{j=1}^{N}\frac{1}{h^p}K_L\left(\frac{\|\mathbf{z}-\mathbf{z}_j\|_p}{h}\right), \text{ s.t. } \int K(u)du = 1 \end{aligned} \tag{5}$$

The KL divergence loss for KDE can be computed as follows:

$$\begin{aligned} \mathcal{L} &= \underset{\mathbf{z}}{min}\, KL(p\|q) = \underset{\mathbf{z}}{min} \sum_{i=1}^{N}\left(p(\mathbf{y}_i)\log\frac{p(\mathbf{y}_i)}{q(\mathbf{z}_i)}\right) \\ &= \underset{\mathbf{z}}{min}\,\frac{1}{Nh^d}\sum_{i=1}^{N}\sum\left(K_H(\mathbf{y}_i,\mathbf{y}_j)\log\frac{\sum_j K_H(\mathbf{y}_i,\mathbf{y}_j)}{\sum_j K_L(\mathbf{z}_i,\mathbf{z}_j)}\right) + c_1 \\ &\leq \frac{1}{Nh^d}\underbrace{\underset{\mathbf{z}}{min}\sum_{i=1}^{N}\sum_{j=1}^{N}K_H(\mathbf{y}_i,\mathbf{y}_j)\log\frac{K_H(\mathbf{y}_i,\mathbf{y}_j)}{K_L(\mathbf{z}_i,\mathbf{z}_j)}}_{\mathcal{J}} + c_1 \\ &\leq c_2 \times \mathcal{J} + c_1 \end{aligned} \tag{6}$$

$\mathcal{J}$ is the objective function of t-SNE (with specific kernels) which upper bounds (with a multiplicative scale and an additive constant) the estimated kernel density estimation loss function.



a. Using t-SNE

b. Lt-SNE (also shows contraction ratio contour lines)

Fig. 9. Lower dimension mapping results on WINE dataset.

Superiority of Lt-SNE: Instead of directly applying t-SNE on raw features, we feed latent features obtained via LE to t-SNE. This results in more superior maps (see Proposition 2, Figure 9 and Section § 4.2 for justification) and called as *“Lt-SNE"*. In Lt-SNE, we employ the same kernel functions as in t-SNE, i.e. normalized gaussian kernel in higher dimensions and heavy tailed kernel (a student t-distribution with one degree of freedom) in lower dimensions as follows.

$$\begin{aligned} K_H(\mathbf{y}_i, \mathbf{y}_j) = p_{ij} &= \frac{\exp\left(- \frac{\|\mathbf{y}_i - \mathbf{y}_j\|^2}{2\sigma_i^2}\right)}{\sum_{k \neq l} \exp\left(- \frac{\|\mathbf{y}_k - \mathbf{y}_l\|^2}{2\sigma_k^2}\right)} \text{ and} \\ K_L(\mathbf{z}_i, \mathbf{z}_j) = q_{ij} &= \frac{(1 + \|\mathbf{z}_i - \mathbf{z}_j\|^2)^{-1}}{\sum_{k \neq l}(1 + \|\mathbf{z}_k - \mathbf{z}_l\|^2)^{-1}} \end{aligned} \tag{7}$$

such that $K_H(\mathbf{y}_i, \mathbf{y}_j)$ and $K_L(\mathbf{z}_i, \mathbf{z}_j)$ sum to 1. Hence, Lt-SNE has the same final objective function as t-SNE which matches the expressions given in Eq. (6):

$$\mathcal{J} = \min_{\mathbf{z}} \sum_{i,j=1}^{N} p_{ij} \log \frac{p_{ij}}{q_{ij}} \tag{8}$$

The above optimization problem is non-convex, but the gradient descent method yields reasonable results. Although it is possible to directly apply t-SNE on the original data matrix, we demonstrate that Lt-SNE produces much better visualization maps in $\mathbb{R}^2$ than t-SNE, theoretically in case of finite mixture of nonparametric distributions (Proposition 2) and empirically in comparison with other major dimension reduction techniques (see § 4.2).

Proposition 2 *Assume that the data points are i.i.d. samples generated from a finite mixture of nonparametric distributions. Let* (i, j) *be any pair of data points belonging to different distribution. Then the mapping of Lt-SNE yields a larger separation distance as compare to t-SNE in the lower dimensions (with high probability) i.e.,*

$$\|\mathbf{y}_i^{'} - \mathbf{y}_j^{'}\|^2 \geq a_1 \|\mathbf{y}_i - \mathbf{y}_j\|^2 + a_2,$$

where $\mathbf{y}_i^{'}, \mathbf{y}_j^{'}$ *and* $\mathbf{y}_i, \mathbf{y}_j$ *are lower dimension feature vectors of Lt-SNE and t-SNE respectively.* a_1, a_2 *are positive constants.*

Proof: The proof relies on the "Finite-Sample Angular Structure" theorem [Schiebinger et al(2015)Schiebinger, Wainwright, Yu et al] shown for kernelized spectral clustering. The above proposition shows that if $a_1 \geq 1$, which is generally the case, different clusters in Lt-SNE are mapped farther to each other as compared to t-SNE with high probability.

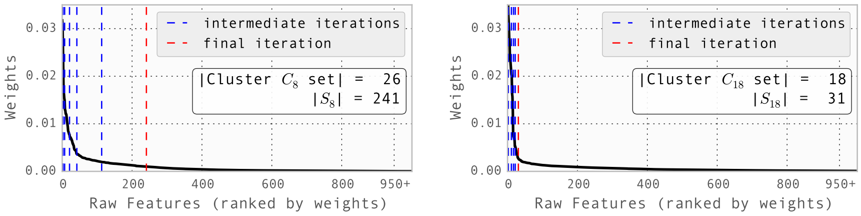
3.3 Culling Cluster-Specific Significant Feature-Set

To help characterize and interpret the relations among different clusters, we define the term *significant feature-set* of a cluster as a set of *observable features* that are most critical to the cluster's formation (as opposed to latent features based on which interpretation is difficult). We take cue from *information theory* and device an algorithm to fit in the framework of our analysis to cull cluster-specific significant feature-sets for meaningful interpretations.

After applying our extraction and projection steps, suppose we have obtained C clusters from some data matrix, say M, of size $N \times D$. Note that every cell in the data matrix M represents the relation between the data point and the observed feature. We slice the data matrix M horizontally into cluster-specific sub-matrices $\{m_c\}_{c \in C}$, where rows represent only those data points that are part of cluster c, and columns represent the feature set D. We therefore obtain $|C|$ sub-matrices, where every sub-matrix $\{m_c\}$ is of size $n_c \times D$, where n_c is the number of data points in cluster $c \in C$. Our goal is to inspect each of the sub-matrix m_c individually, and cull a subset of features S_c (or significant set) from the entire observable feature set D, such that the features selected are most critical to cluster c's existence. We now explain our approach to cull this cluster-specific significant feature set. For easier readability of equations, we refer sub-matrix m_c as m and its associated notations n_c, S_c as n, S, respectively. N remains unchanged. From some cluster-specific sub-matrix m, we first build a *weight* vector W defined by:

$$\mathsf{W} = \left\{ x_i = \left(\sum_j m_{ij} \Big/ \sum_k \sum_l \left(m_{kl} \right) \right) \forall j, k, l \right\}_{i \in D} \tag{9}$$

Every i^{th} element in W is an aggregated value that quantifies interactions between the i^{th} observed feature and all the data points belonging to cluster c. Using W vector, we check if any of its values *stand out* and are *significantly* different from others. An easy way to verify this would be to sort vector W, say in decreasing order, and observe the fall in the distribution.



a. Cluster 8

b. Cluster 18

Fig. 10. Step-by-step illustration of culling significant feature-set from Clusters 8 & 18 (also discussed in case study 1 § 5.1 - Dataset 1). |Cluster set x|: number of data points in cluster x; $|S_x|$: number of features in significant set of cluster x;

Black curves in Figures 10a and 10b show the values of weight vectors W for clusters 8 and 18, respectively. These clusters were obtained as part of a case study discussed later in § 5.1. We observe that cluster 8's slope drops relatively slower than that of cluster 18. In other words, data points of cluster 8 collectively state that more number of features are vital to its formation than cluster 18. In order to account and quantify these differences, we introduce a notion of *relative uncertainty* $\widetilde{\mathsf{RU}}(\mathsf{W})$ defined as $\tilde{\mathsf{H}}(\mathsf{W})/\log|\mathsf{W}|$, where $\tilde{\mathsf{H}}(\mathsf{W})$ is the "*entropy*-like" measure used to quantify the unpredictability of the values in vector W. $|\mathsf{W}|$ is the support (or size) of the vector W or the number of observed features. The degree of uniformity (or relative uncertainty) of W is given by:

$$\widetilde{\mathsf{RU}}\left(\mathsf{W}\right) = \frac{\tilde{\mathsf{H}}(\mathsf{W})}{\log|\mathsf{W}|} = \frac{-\sum_i \left(w_i \log w_i \right)}{\log|\mathsf{W}|} \tag{10}$$