Probabilistic Enhanced Mapping with the Generative Tabular Model

Rodolphe Priam
LMA UMR 6086 CNRS/Université de Poitiers
Site Noron, 8 Rue Archimède, 79000 Niort

Mohamed Nadif
CRIP5, Université Paris Descartes,
45 rue des Saints-Pères, 75006 Paris

Abstract

Visualization of the massive datasets needs new methods which are able to quickly and easily reveal their contents. The projection of the data cloud is an interesting paradigm in spite of its difficulty to be explored when data plots are too numerous. So we study a new way to show a bidimensional projection from a multidimensional data cloud: our generative model constructs a tabular view of the projected cloud. We are able to show the high densities areas by their non equidistributed discretization. This approach is an alternative to the self-organizing map when a projection does already exist. The resulting pixel views of a dataset are illustrated by projecting a data sample of real images: it becomes possible to observe how are laid out the class labels or the frequencies of a group of modalities without being lost because of a zoom enlarging change for instance. The conclusion gives perspectives to this original promising point of view to get a readable projection for a statistical data analysis of large data samples.

1 Introduction

Increasing number of nonlinear methods \( \mathcal{M} \) as [15, 3, 14, 7, 8, 6] are proposed in the literature to project a data sample \( \{x_i\}_{i=1}^I \), demonstrating that the data cloud projection \( \{y_i = \mathcal{M}(x_i)\}_{i=1}^I \) paradigm is a good way to see the structure of the distribution of a dataset. An alternative to this point of view is the self organizing maps or SOM [10] which construct a tabular projection of the data cloud, with neighbour cells on the table enough similar in the data space to preserve the topology. This way to see data is certainly more readable than a cloud of numerous points on the plane, where points are often hard to see while the shape of the projected distribution appears. For this reason, some zoom tools are generally needed, but can be a way to be lost on the data projection when one has to change the scale of the enlarging. As these new methods are certainly projecting data in a complementary way, and that the tabular view is easily understood, we aim in this paper to add a tabular view for any 2D data cloud by an alternative approach to the self-organizing map.

The SOM solution for getting the tabular view would be to construct a self organizing map over the bidimensional projection. To construct a tabular view of a data set, the self-organizing map algorithms are clustering methods with their center vectors constrained with an imaginary regular mesh. Each node of this mesh corresponds to one class center, and the aim of the algorithm is to train the center vectors such as nearer they are on the mesh, nearer they are in the data space. This strong hypothesis is only true for parts of the mesh because the data distribution can be complex in practice. So a SOM algorithm constructs well organized areas separated by a frontier which reveals that these areas are not connected in the data space. To show a tabular view of the projected values \( y_i \), self-organizing maps are an appealing solution: in 2D, they give a discretized view of the distribution shape with zoom capabilities over high density areas. Several algorithms already exist like the GTM or Generative Topological Mapping [2] for instance. This model is a Gaussian mixture [13] with centers constrained to lay on a discretized surface, and constituting the knot of this discrete surface.

In our case, we are interesting in showing the tabbled data as they are laid out in the plane, as proposed in the tabular model: it is a new constrained Gaussian mixture where means are lying on linear rows and columns which are free to move to show area with high density. Our study permits to construct readable tables from multimedia corpuses. When cells or classes are numerous like in the case of the massive datasets, a Pixel-Based [9] approach is necessary, carrying interesting and synthetic information. In this paper, we present the model, we give the obtained pixel maps for an image dataset, then we explain how visual datamining is beneficial, and finally we conclude with discussion and some perspectives.

2 The Generative Tabular Model

In this section, we suppose that a mapping \( \mathcal{M} \) was used to obtain the bidimensional coordinates \( y_i = (y_{is})_{s=1,2} \)
which are processed to get a tabular view of the projected data cloud. The basis of the algorithm is the Gaussian mixture model which extends the K-means [12] method to a probabilistic expression with a classifying hidden variable. A Gaussian mixture density is written \( P(y_i) = \sum_{1 \leq k \leq K} \pi_k G(y_i; m_k, \sigma) \) with \( K \) factors, components, or clusters, where the \( k \)-th \( G \) density is a normal density with the mean \( m_k \) and the spherical variance parameter \( \sigma \). The parameter \( \pi_k \) is the probability that an observation belongs to the \( k \)-th component, it therefore corresponds to the proportion of the \( k \)-th cluster. The log-likelihood of the observed data \( D \) assumed to be an i.i.d sample \( \{y_i\}_{i=1}^{l} \) from the probability distribution with density \( P(y_i) \) is given by:

\[
\mathcal{L}(\theta | D) = \sum_{1 \leq i \leq l} \log \left\{ \sum_{1 \leq k \leq K} \pi_k G(y_i; m_k, \sigma) \right\},
\]

where \( \theta = (m_1, m_2, \ldots, m_K, \pi_1, \pi_2, \ldots, \pi_{K-1}, \sigma) \). Inference of this model is done by maximizing the loglikelihood which is intractable in an exact closed form solution. The EM algorithm [4] is applied to this problem by using the likelihood completed by the knowledge of the partition \( Z = \{Z_1, Z_2, \ldots, Z_K\} \),

\[
\mathcal{L}(\theta, Z | D) = \prod_{1 \leq i \leq l} \pi_{z_i} G(y_i; m_{z_i}, \sigma)
\]

where \( z_i \) the latent variable of which the unknown value is in \( K = \{1, 2 \ldots, K\} \).

It proceeds iteratively in two steps, E (for expectation) and M (for maximization), in maximizing the conditional expectation of \( \mathcal{L}(\theta, Z | D) \), given a previous current estimate \( \theta^{(t)} \):

\[
Q(\theta | \theta^{(t)}) = \sum_{1 \leq i \leq l} \sum_{1 \leq k \leq K} q_k^{(t)}(y_i) \log \{ \pi_k G(y_i; m_k, \sigma) \}
\]

where

\[
q_k^{(t)}(y_i) = \frac{\pi_k^{(t)} G(y_i; m_k^{(t)}, \sigma^{(t)})}{\sum_{1 \leq \ell \leq K} \pi_{\ell}^{(t)} G(y_i; m_{\ell}^{(t)}, \sigma^{(t)})}
\]

denotes the conditional probability, given \( y \) and \( \theta^{(t)} \), that \( y_i \) arises from the mixture component with density \( G(y_i; m_k^{(t)}, \sigma^{(t)}) \). Each iteration of EM uses the following steps:

- **E-step:** compute the conditional expectation of \( \mathcal{L}(\theta, Z | D) \); in the mixture case this step reduces to the computation of the conditional density \( q_k^{(t)}(y_i) \).

- **M-step:** compute \( \theta^{(t+1)} \) maximizing \( Q(\theta | \theta^{(t)}) \) which leads to:

\[
\pi_k^{(t+1)} = \frac{1}{l} \sum_{1 \leq i \leq l} q_k^{(t)}(y_i),
\]

and

\[
m_k^{(t+1)} = \frac{\sum_{1 \leq i \leq l} q_k^{(t)}(y_i) y_i}{\sum_{1 \leq i \leq l} q_k^{(t)}(y_i)},
\]

\[
\sigma^{(t+1)} = \sqrt{\frac{1}{l} \sum_{1 \leq i \leq l} \sum_{1 \leq k \leq K} q_k^{(t)}(y_i) \| y_i - m_k^{(t+1)} \|^2}.
\]

In the following paragraph, we constrain the \( m_k \) vectors to get a matrix representation. We construct a regular mesh with \( K_1 \) columns and \( K_2 \) rows, such as \( k \) is indexed by \( k = K_1 \times (k_2 - 1) + k_1, (k_1 = 1, \ldots, K_1; k_2 = 1, \ldots, K_2) \), and \( K = K_1 K_2 \). Let’s have:

\[
m_k = \begin{bmatrix} m_{k,1} \\ m_{k,2} \end{bmatrix}
\]

with \( m_{k,s} = \frac{\sum_{1 \leq i \leq l} \exp(u_{i,s})}{\sum_{1 \leq i \leq K_s} \exp(u_{i,s}) + 1} \),

where \( s \in \{1, 2\} \), and \( u_{k,s} \) is a real unknown parameter which can be obtained in maximizing \( Q(\theta | \theta^{(t)}) \). Our parametrization is such as \( 0 \leq m_{k,s} \leq 1 \) to get simplex expressions. Note that, the additive constant 1 in \( m_{k,s} \) can be replaced by any positive real, and the sum over the numerator induces a fixed topological direction for the corresponding probabilities: a topology-driven variant of the classical soft-max [1] parameterization. Finally, we normalize the range of the vector components by calculating \( (y_{is} - \min(y_{is})) / (\max(y_{is}) - \min(y_{is})) \) as a new \( y_{is} \). In this case, with \( q_k^{(t)}(y_i) = q_{(k_1,k_2)}^{(t)}(y_i) \) and \( \pi_k = 1/K \) - roughly speaking, to put more centers in high density areas, we propose the new marginal-sensitive criterion:

\[
Q_m(\theta | \theta^{(t)}) = \sum_{1 \leq s \leq 2} \sum_{2 \leq i \leq l} \sum_{1 \leq k \leq K_s} q_{k,s}^{(t)}(y_i)(y_{is} - m_{k,s})^2,
\]

where \( q_{k,s}^{(t)}(y_i) = q_{k,s}^{(t)}(y_i) \) is the marginal of \( q_{(k_1,k_2)}^{(t)}(y_i) \) over \( k_1 \) if \( s = 2 \), and \( k_2 \) if \( s = 1 \). A closed form for maximizing this quantity does not exist yet, so we use a gradient ascent to calculate:

\[
m^{(t+1)} = \arg\max_m Q_m(\theta | \theta^{(t)}).
\]

By deriving the criterion, we get the gradient vector \( \nabla m^{(t)} \) with the \( u_{\ell,s} \) derivative components \( \nabla m^{(t)}_{\ell,s} \):

\[
\sum_{1 \leq \ell \leq l} \sum_{1 \leq k \leq K_s} q_{k,s}^{(t)}(y_i)(m_{k,s}^{(t)} - y_{is})(\delta_{\ell,k} - m_{k,s}^{(t)}) \Delta m_{\ell,s}^{(t)},
\]

with \( \delta_{\ell,k} \) is one if \( \ell \leq k \) and zero else, and with \( \Delta m_{\ell,s}^{(t)} = (m_{k,s}^{(t)} - m_{k,s}^{(t-1,s)}) \) where \( m_{0,s} = 0 \). Finally, we end to the gradient step:

\[
u_{\ell,s}^{(t+1)} = u_{\ell,s}^{(t)} - \rho^{(t)} \nabla m_{\ell,s}^{(t)},
\]

for a well chosen \( \rho^{(t)} \) decreasing function to get a converging value for our \( u_{\ell,s} \) parameters. A Newton-Raphson step by calculating the Hessian is an alternative; its expression
is not given here. Finally, iterating calculus of the \( u_{k,s} \) and \( \sigma \) converges towards a maximum (local). We note with a hat the final parameters. The resulting centers, in the data space, are easily found from the \( m_k \) vectors by the inverse preceding translation, homothety, and a possible rotation when needed, as explained at the end of the paper.

3 Application to visualization

We illustrate the tabular view on 500 images from the dataset\(^1\) of 2000 images from handwritten digits digitalized in binary images. The projecting method is here the Locally Linear Embedding or LLE mapping [14]. This method is very efficient when dealing with images database which contains classes of objects with similar shapes because locally linear relations exist. We get the resulting projection or \( y_i \) sample in Figure 1. Our model is a complementary approach to the existing solutions in visualization, with the strong perspective to ameliorate their own definitions thanks to the help of the local density of the data cloud. One is able to explore the data cloud on the plane without taking care of the exact relative plot positions or local density change which should have induced a possible annoying variation of the mental map. When the number of rows and columns grows up, we need a new way to see how classes are organized among the cells because the table is too large. A pixel view is then very natural and brings us to a synthetic and possibly evolutive or interactive visualization. We give the location of the ten classes over the discretized plane in Figure 2: each cluster has a rectangular shape and its size is sensitive to the local density. Here a high density induces a smaller shape than for a low density which permits more summarization on the following global picture of the classes. Each class is showed separately by a map of bits on the well-defined visualization in Figure 3: every rectangular clusters from Figure 2 are reduced to the same size near few pixels only, and for each class a bitmap is drawn. For a given class, its bitmap shows the new reduced cluster with colour filling if any datum which have the class label is inside the corresponding original cluster. We illustrate the pixel-view paradigm: the bitmaps show in a synthetic fashion the relations, for LLE, between the 10 classes of digits. A potentially interactive application is to select an area from the original 2D projection and then discover from the pixel maps the corresponding classes intensities with gray level.

We are able too to zoom high densities areas by the following proposed formula, with \( \Phi \) the Gaussian cumulative distribution function and \( P_\omega \) the marginal of \( P \) on \( y_{3-s} \):

\[
y_{is} = \frac{\int_0^{y_{is}} P_s(y_s) dy_s}{\int_0^1 P_s(y_s) dy_s} = \frac{\sum_{1 \leq k \leq K} \Phi\left(\frac{y_{is} - m_k}{\sigma_k}\right) - \Phi\left(\frac{-m_k}{\sigma_k}\right)}{\sum_{1 \leq k \leq K} \Phi\left(\frac{y_{is} - m_k}{\sigma_k}\right) - \Phi\left(\frac{-m_k}{\sigma_k}\right)}
\]

This formula is based on increasing entropy of the coordinates of the projection in Figure 4 by adding space between points which are the most numerous, i.e. when density of

4 Discussion and perspectives

We have proposed a new way to explore a data projection by a new generative model which constructs a table from a cloud which is generally difficult to read when data are too numerous. Our method has several outstanding properties which permit it to carry out several interesting operations over the projection and to get a final interpretable representation.

When the data sample is large, a good initialization and supra-linear estimation by a Newton algorithm are necessary to get a quick and help convergence to avoid local minima of the likelihood. The initialization is easily driven from a classical percentiles discretization along rows and columns. We suppose that we have the $2 \times K$ percentile values $h_{ks}$. Then initial values for $u_{i\alpha}$ are written:

$$u_{i\alpha}^{(0)} = \log \{(G_s^T G_s)^{-1} G_s^T h_{ks}\},$$

where $\log$ is the componentwise log function, $u_{i\alpha}^{(0)}$ is the column-vector with $u_{i\alpha}$ for components, $h_{ks}$ is the column vector with $h_{ks}$ as components, and the matrix $G_s$ has for $k$-st row the vector $[1 - h_{ks}, 1 - h_{ks}, \ldots, 1 - h_{ks}, -h_{ks}, \ldots, -h_{ks}]$ with $k_s$ components $(1 - h_{ks})$ followed by $K - k_s$ components $h_{ks}$.

When the variance data is not very well oriented compared to the axes of the reference mark, it makes sens to introduce a rotation to get most of the inertia for the tabular view. So we propose here an ad hoc rotation over the data sample: we add a matrix transformation such as we replace $y_{i\alpha}$ by $\bar{y}_{i\alpha} + W(y_{i\alpha} - \bar{y})$ where $W$ is a $2\times2$ matrix and $\bar{y}$ is the empirical (sample) mean of the vectors $y_{i\alpha}$. The gradient is updated by replacing $y_{i\alpha}$ by $\bar{y}_{i\alpha} + \sum s_w^{(t)} y_{is}$ where $\bar{y}_{is} = (y_{is} - \bar{y}_{is})$. A more elegant and general transformation is the true rotation matrix $W^{(t)}$, written, for the rotation angle $\alpha^{(t)}$, and an implicit proportional scaling $\beta$ for $\bar{y}_{is}$:

$$W^{(t)} = \begin{bmatrix} \cos \alpha^{(t)} & -\sin \alpha^{(t)} \\ \sin \alpha^{(t)} & \cos \alpha^{(t)} \end{bmatrix}. $$

So, we get the 1-st derivative $D\alpha^{(t)}$ and 2-nd derivative $H\alpha^{(t)}$ formulas which provide the Newton-Raphson update step:

$$\alpha^{(t+1)} = \alpha^{(t)} - D\alpha^{(t)}/H\alpha^{(t)},$$

ending to the sought rotation. The complete enhanced model shares interesting properties with several approaches from data analysis.

which gives almost the same rotation: when the centers are enough numerous, and with an unsmoothing affection of the data vectors to classes. The corresponding solution is similar to the PCA projection because:

$$
\sum_i ||y_i - (\bar{y} + \hat{W}^{-1}(\hat{m}_z) - \bar{y})||^2 = \sum_i ||\tilde{y}_i - \hat{W}^{-1}\tilde{m}_z||^2
$$

is then minimal. The method is connected also to a discretization approach and a density histogram construction for bivariate continuous data: we obtain two new discrete variables, along rows and columns, with interval choice sensitive to the marginal density of the bivariate projection. We are able to see the data cloud in an easier and quicker readable way. Roughly speaking, our method is similar to a PCA rotation preceding by a bivariate discretization near a percentile-based one which can be used to initialize the parameters. This is an alternative to an equally-spaced cut of the plane to provide us most of the visual information to show. The pixel-based visualization is introduced to show the final maps and to allow their comparison all together. The method is independent of the projection choice to obtain the 2D plots. It is an alternative to the self-organizing map over the original projection, when a biformal projection yet exists by any nonlinear method.

Moreover, a self-organizing map could have been used to analyse the 2D projection instead of the tabular model. The difference is the risk to loose the exact plot locations over the original projection. This alternative approach is complementary to our work and demands to be studied in the future. Some additive constraints can be used to enhance the tabular contents. It seems that the use of the density of a projected data cloud is under-used in the current tools for visualizing massive datasets. We believe that such information can be further used instead of the classical 3D continuous bivariate density in particular. Several extensions of our approach are possible too, like new ways to see the local density of the data cloud, new ways to zoom a part of the plane, and to take advantage of the pixel paradigm. Finally, the Generative Tabular Model extended to a 3D clustering with a data projection embedded inside the Gaussian distributions is an interesting perspective for visual datamining.

References