A Multigraph Representation Algorithm and Extension

Shuzhi Su

2014.9.11

References

- [1] A Multigraph Representation for Improved Unsupervised/Semi-supervised Learning of Human Actions[J],CVPR,2014

- [2] Open Problems in Spectral Dimensionality Reduction[M],2014

- [3] feature extraction functions and applications[M],2006

Single graph method

- The disadvantage of this method:

1) the performance of single graph method is highly dependent on how well an affinity graph reflects the original data structure.

2) When an affinity matrix can be generate ,it really exists that significant information is lost from original samples.

For instance, the information loss is particularly severe for small datasets (each row of W is the low-dimensional),or when the dataset’s original feature space is a high dimensionality.

Single graph method

3) The bag-of-feature (BoF) histograms are the focus of this paper.

For this histogram data, there may be multiple statistically independent components. Therefore, a single graph would not be able to distinguish between these components.

Therefore, this paper uses multiple graphs are constructed, each corresponding to a different component, which can overcome the aforesaid issue.

Feature Grouped spectral multigraph

- For FGSM method, the original feature space of the dataset should be have two properties:

1) the feature space must be high dimensional .

2) the feature set must be divisible into several disjoint subsets with high independent between all the subsets and high dependence within the subsets.

The general steps of this algorithm

- First, mutually independent subsets of the original feature space is generated through feature clustering(Spectral clustering has been used in this literature).
- Secondly, construct a separate graph for each feature subset.
- Next, a spectral embedding (using spectral clustering) is calculated on each graph.
- Finally, the embeddings are aggregated into a single representation.

Details of the first step

- Calculate HSIC affinity matrix W between pairs of columns of data set X through the following formula:

$$W_{jk} = tr((L_j^T L_k)(L_k^T L_j))$$

- This formula is derived from the below prototype:

$$W_{jk} = \frac{1}{(1-n)^2} tr(HK_j Hk_k))$$

Where H=$\sigma_{jk} - n^{-1}$, K_j and $K_k$are the outer products of vectors X_j and $x_k$respectively, and n is the number of samples.

Details of the first step

- Using spectrally clustering algorithm, the m feature clusters ($C_1, C_2, \ldots\ldots, C_m$) can be generated.
- The spectrally clustering detailed steps are:

- 1) calculate diagonal degree matrix D with D_{ii} equal to the sum of the ith row of W, and obtain normalized affinity matrix L=$D^{-1/2}WD^{-1/2}$

Details of the first step

- 2) find the k highest eigenvectors of L ,and construct a matrix E columnwise as $[e_1, e_2, \ldots, e_k]$

- 3) Normalize E so each row sums to 1.

- 4) obtain m clusters through preforming k-means algorithm for rows of normalized E.

The second step: construct subgraphs

- Construct a separate graph from every feature subset.
- The key aspect of this step is generating an affinity matrix of each subgraph using the histogram intersection method:
- $W_{jk}^{i} = sum(\min(x_j^i, x_k^i))$

The key step: the third step

- Spectral embedding is performed on every subgraph as steps 1)-3) of the first step.
- Advantages of this spectral embedding:

1) spectral embedding methods is possible to uncover the underlying structure of a high-dimensional data.

2) Intuitively, through this spectral embedding, the original data can be transformed into the surface of the k-dimensional hypersphere, and form more “obvious” clusters.

The final step

- Fuse these embeddings into a single representation: concatenate all embedding (E_1, E_2,…., E_m) columnwise through scaling each embedding by λ_i,the standard deviation of all Euclidean distance between the subset C_i.

Extension of spectral embedding

- There are two important kinds of machine learning algorithms, induction learning algorithms and transduction learning algorithms.

1. The end result of most inductive machine algorithms is a function that minimizes the empirical average of a loss criterion (possibly plus regularization), and the ideal solution is a function that minimizes the expected value of that loss criterion under new samples, unknown true distribution, where the expected loss is known as the generalization error.

It is clear that these algorithms can be applied on new samples.

Extension of spectral embedding

2. Such a good characterization is missing for some transduction learning algorithms, such as spectral embedding. These algorithms share the same flaw:

any test data for which an embedding is desired must be included in the (unlabeled) “training set”.

The Nyström method

A data set X=$\{x_1, x_2, \ldots, x_n\}$, $x_i \in R^p$

A kernel matrix K corresponding to the data set X

$v_{r,i}$ represents the i-th coordinate the r-th eigenvectors of K (sorted in order of decreasing eigenvalues), associated with the eigenvalue l_r.

- With these notations, the Nyström formula is:

$$f_r(x) = \frac{\sqrt{n}}{l_r} \sum_{i=1}^{n} k(x, x_i) v_{r,i} = \frac{\sqrt{n}}{l_r} k(x, X) v_r$$

where f_r is the r-th Nyström estimator for the sample x

Explain the method

- The Nyström method can be explained from various views, but I choose kernel PCA to explain it, which can be easily understood.

Extension formula

- Through the above analysis, we can obtain a natural consequence of extending new samples to its corresponding embeddings.
- The unifying extension framework for spectral embedding algorithms is as following.

$$f_r(x_{new}) = \frac{1}{l_r}\sum_{i=1}^{n} k(x_{new}, x_i)v_{r,i} = \frac{1}{l_r} k(x_{new}, X)v_r$$

$P(x_{new}) = (f_1(x_{new}), f_2(x_{new}), \ldots\ldots, f_d(x_{new}))$

where $P(x_{new})$ is a embedding for a new sample x_{new}.

Thank you !