Information Theoretic Prototype Selection for Unattributed Graphs

Lin Han[1], Luca Rossi[2], Andrea Torsello[2],
Richard C. Wilson[1], and Edwin R. Hancock[1]

[1] Department of Computer science,University of York,UK
[2] Department of Environmental Science, Informatics and Statistics,
Ca' Foscari Univerisity of Venice, Italy

Abstract. In this paper we propose a prototype size selection method for a set of sample graphs. Our first contribution is to show how approximate set coding can be extended from the vector to graph domain. With this framework to hand we show how prototype selection can be posed as optimizing the mutual information between two partitioned sets of sample graphs. We show how the resulting method can be used for prototype graph size selection. In our experiments, we apply our method to a real-world dataset and investigate its performance on prototype size selection tasks.

Keywords: Prototype Selection, Mutual information, Importance Sampling, Partition function.

1 Introduction

Relational graphs provide a convenient means of representing structural patterns. Examples include the arrangement of shape primitives or feature points in images, molecules and social networks. Recently, there has been considerable interest in learning prototype graphs which can capture the structural variations given a set of sample graphs [1,12]. These approaches are frequently sample-based, having several candidate prototype graphs in hand, we are confronted with the problem of selecting the best one. This problem falls into the category of model selection, which is one of the fundamental tasks in pattern analysis. A good model should be able to summarize the observed data well. Moreover, it should have good predictive capabilities. There are a wealth of principles in the literature for selecting the best model [9,11,10]. Generally speaking, although the principles are motivated from different viewpoints, most of them employ penalizing the parameters (or complexity) of the model in order to generalize well on a new dataset. For example, the Akaike information criterion(AIC) penalizes the model using the value of twice the number of free parameters of the model [13], while the minimum description length criterion uses a universal coding [14].

The main drawback of these approaches is that they cannot be easily extended from the vector domain to the graph domain. On the other hand, other frameworks such as the approximate set coding [3] can be transformed to the graph domain with the help of sampling techniques such as Importance Sampling.

G.L. Gimel' farb et al. (Eds.): SSPR & SPR 2012, LNCS 7626, pp. 33–41, 2012.

In this paper we present an approach to selecting the optimal prototype size for a set of sample graphs. Our method is an extension of the theory of the approximate set coding to graph data. The prototype of optimal size is that which maximizes the mutual information between the two partitioned sets of the sample graphs. To measure the mutual information, we need to compute the partition functions of the two partitioned sets and their joint partition function. The computation of the partition function involves exploring the complete hypothesis space and this is a NP hard problem for graphs. We locate an approximate solution to this problem by using the importance sampling approach.

The remainder of the paper is organized as follows. We first briefly introduce the theory of the approximate set coding [3]. Then we describe how we extend the theory on model selection in vector domain to the graph domain. This includes how to characterize the sample sets using the partition function and how to approximate the value of the partition function using the importance sampling approach. In the last part we provide some preliminary experimental results.

2 Approximation Set Coding

In this section we briefly review the theory of the approximate set coding proposed in [3,4]. In this context, a *hypothesis* is a solution to our pattern recognition problem. In this specific case, a hypothesis c is a mapping (matching) of all of our sample graphs to a prototype graph. We also have a *cost function* $R(c)$ which evaluates the quality of a particular matching. Naturally $R(c)$ depends on the prototype graph proposed for the data samples.

Given a prototype graph drawn from set of possible prototypes (usually of different sizes or complexity), we can find the best matching and prototype configuration by optimizing $R(c)$. We denote the best hypothesis as $c^{\perp}$. As usual, we cannot use $R(c)$ to select the best prototype from the set, as the more complex prototypes have lower costs (they fit the samples better) but do not generalize well.

Approximation set coding uses the observation that there are a set of transformations which alter the sample data without essentially changing the prototype in any way. For example, if we consider the sample graphs in a different order, or their nodes are permuted in some way, then the structure of the recovered prototype should be the same (although the prototype graph nodes may also be in a different order). We can use this fact to measure how good our prototype is at recovering these transformations when they are coded using the prototype graph and sent through a noisy channel. To do this, we split the sample data into two partitions. The first partition is used to code the transformation, and the second partition provides a prototype graph to decode the transformation. We then attempt to maximize the amount of information transmitted. The analysis in [3] shows that the mutual information between sender and receiver is

$$I_\gamma = \frac{1}{n} \log \left(\frac{|T||\varDelta C_{\gamma,12}|}{|C_{\gamma,1}|C_{\gamma,2}} \right) \tag{1}$$

where $|C_{\gamma,1}|$ is the number of hypotheses which are within a cost γ of the best cost in set 1 (and likewise for $|C_{\gamma,2}|$). The quantity $|\varDelta C_{\gamma,12}|$ is the number of hypotheses *on set 2* which are within a cost γ of the best cost *in set 1.* To calculate this, we need a way of transferring hypotheses from set 2 to set 1. For more details of these techniques, the reader is referred to [3,4].

3 Prototype Selection for Graphs

In this section, we extend the methodology of the approximate set coding from the vector domain to the graph domain. Our main contribution here is that we redefine three important ingredients in the approximate set coding (i.e. hypothesis, cost function and partition function), and generalize them from the vector domain to the graph domain. In the following, we commence by introducing our problem and then give formal definitions of the ingredients.

Given a set of sample graphs, our aim is to select the optimal size of the prototype graph for the sample graphs. To ensure that the optimal prototype graph generalizes well on a new dataset, we adopt a two-sample set scenario and partition the sample graphs into two sets of the same size $\mathcal{G}_1 = \{G_1^{(1)}, G_2^{(1)}, ..., G_n^{(1)}\}$, $\mathcal{G}_2 = \{G_1^{(2)}, G_2^{(2)}, ..., G_n^{(2)}\}$. Here the superscripts indicate different sample-set and the subscripts indicate the graph indices. The best prototype graph is determined according to its generalization capability on the two sets.

3.1 Hypothesis

The hypotheses originally proposed in the clustering problem (where approximate set coding was first used) are the assignments of data points to clusters [4]. Here in our problem the hypotheses consist of a set of mappings of each of the sample graphs onto its corresponding prototype graph. By direct analogy with the clustering problem, each mapping is equivalent to an assignment of a point to a cluster; the prototype graph here is equivalent to the cluster centroid. For each dataset $\mathcal{G}_q (q \in \{1, 2\})$ a hypothesis is $c_q = \{S_1^{(q)}, S_2^{(q)}, ..., S_n^{(q)}\}$ where $S_i^{(q)}$ $(i \in \{1, ..., n\})$ is the assignment matrix between graph i from set q and its corresponding prototype graph $G_M^{(q)}$. The set of all possible hypotheses is $\mathcal{C}$, which consists of all the possible mappings between all samples and the prototype graph.

3.2 Cost Function

To proceed, we require a cost function $R_q(c_q)$ to quantify the effectiveness of a particular hypothesis c_q. The cost function measures how consistent the given mappings are with the prototype graph. Here the cost function of a hypothesis is the negative logarithm of the matching probability between the sample graph and the prototype graph under the hypothesis modelled using the technique described in [15].

$$\begin{aligned} R_q(c_q) &= -\log P(\mathcal{G}_q | G_M^{(q)}, c_q) \\ &= \sum_{G_i^{(q)}} \sum_{a \in V_i^{(q)}} -\log \sum_{a \in V_M^{(q)}} K_a^i \exp\left[\mu \sum_{b \in V_i^{(q)}} \sum_{\beta \in V_M^{(q)}} D_{iab}^{(q)} M_{\alpha\beta}^{(q)} S_{ib\beta}^{(q)}\right] . \end{aligned} \quad (2)$$

In the above, $D_i^{(q)}$ is the adjacency matrix for the sample graph G_i from set q and $M^{(q)}$ is the adjacency matrix for the prototype graph $G_M^{(q)}$. The matrix $S_i^{(q)}$ is the assignment matrix between the two graphs. If nodes a and b of the sample graph $G_i^{(q)}$ are connected, their corresponding element $D_{iab}^{(q)}$ in $D_i^{(q)}$ has a unit value otherwise it is zero. This is same for the nodes α, β of the prototype graph $G_M^{(q)}$. The elements of the assignment matrix $S_{ia\alpha}^{(q)}$ are unit if node a in graph $G_i^{(q)}$ is matched to node α in graph G_M. The cost function above is a natural choice in our problem because it is also involved in measuring the similarity between the sample graphs and the prototype graph during the learning procedure of the prototype graph.

In order to normalize the minimum cost of the hypotheses to zero, we define the relative cost of hypothesis. Suppose the optimal hypothesis (i.e., the hypothesis yielding the lowest costs between the sample graphs and their prototype graph) is $c_q^{\perp}$, the relative cost of hypothesis c_q is $\mathit{\Delta} R_q(c_q) = R_q(c_q) - R_q(c_q^{\perp})$.

3.3 Partition Function

The measurement of the mutual information of the two sample-sets requires counting the number of hypotheses which are within a certain cost of the optimal solution. However, this is hard to do since it involves exploring all the hypotheses. Fortunately, this value can be estimated using concepts from statistical physics. Considering the hypotheses as microcanonical ensembles in statistical mechanics, their number can be estimated by calculating the partition function [4]

$$\mathcal{Z}_q = \sum_{c_q \in \mathcal{C}_q} \exp[-\beta \mathit{\Delta} R_q(c_q)] \tag{3}$$

where β is a positive scaling parameter known as the inverse computational temperature. Essentially, β coarsens the precision of the partition function approximating the number of hypotheses that fit the sample set [3]. When β is zero, the partition function is equal to the number of all the possible hypotheses. When β is very large, the partition function only counts the number of optimal hypotheses. Because β controls the number of hypotheses fitting the sample set, we will call these β-optimal hypotheses. In our case, the hypothesis space is the set of all the possible mappings between the sample graphs and their prototype graph. The hypothesis space is very large and the computation of the partition function will be expensive. Later we show how we use the importance sampling approach to sample the mapping between the sample graphs and their prototype graph and approximate the partition function.

To measure how well the hypotheses generalize for the two sample sets, we count the number of β-optimal hypotheses in the first set which also exist in the second set, when transferred to the first set. We therefore need a way of transferring hypotheses from the second dataset to the first. We denote the cost of the hypothesis c_2 between the transferred graphs and prototype graph $G_M^{(2)}$ as $R_t(c_2)$. This is the cost of making hypothesis c_2 for the graphs $\mathcal{G}_2$ when evaluated against the data in $\mathcal{G}_1$. The following procedure may be used to find the transfer.

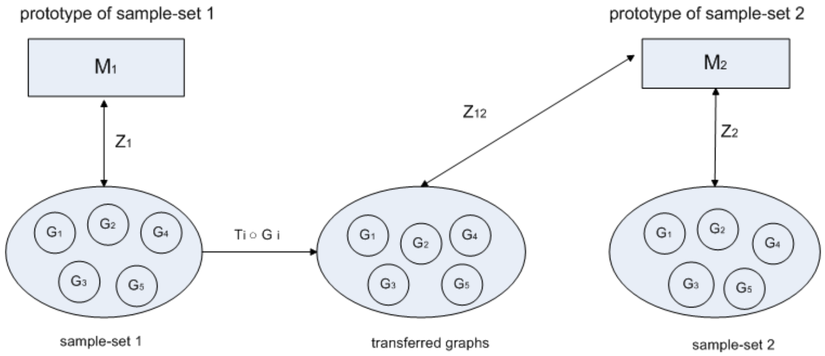


Fig. 1. A diagram illustrates the procedure of computing the three partition functions. When we compute the partition function Z_{12}, we need to count how many of our hypotheses are β-optimal when we use the prototype from set 2 and the data graphs from set 1. We therefore need a way of transferring hypotheses from the second set to the first.

For each $G_i^{(1)}$ graph in $\mathcal{G}_1$, we find the most similar graph in $\mathcal{G}_2$ and the mapping between T_i between the two. $T_i \circ G_i^{(i)}$ is then the image of this graph in the second set. From these images, we compute the cost of c_2 by comparing the images to the prototype graph $\mathcal{G}_M^{(2)}$ under the mappings in c_2. Finally, the joint partition function is formulated as

$$\mathcal{Z}_{12} = \sum_{c_2 \in \mathcal{C}_2} \exp[-\beta(\Delta R_t(c_2) + \Delta R_2(c_2))] \quad . \tag{4}$$

The quantity $\Delta R_t(c_2)$ is the relative cost of hypothesis c_2 between the image graphs of $\mathcal{G}_1$ in the second set and the prototype graph $G_M^{(2)}$. It is equivalent to the cost of hypothesis c_2 between the image graphs and $G_M^{(2)}$ minus their minimum cost. Figure 1 illustrates the procedure of computing partition functions Z_1, Z_2 and the joint partition function Z_{12}.

Prototype graphs with different sizes are ranked according to their mutual information between the two sets

$$I_\beta = \frac{1}{n} \log\left(\frac{kZ_{12}}{Z_1 Z_2}\right) \quad . \tag{5}$$

In the above equation, Z_1 and Z_2 are the respective partition functions of two sample sets, and Z_{12} is their joint partition function. The constant k is a normalization factor which keeps the value of the mutual information equal to zero when β is zero. The value of the mutual information can be interpreted as the maximum generalization capacity of prototype graphs. Hence our problem is posed as that of finding the prototype graph that maximizes this mutual information.

3.4 Importance Sampling

In order to deal with the super-exponential growth of the set of hypotheses, we resort to an Importance Sampling [7] approach in a manner similar to that reported by Torsello [2].

Importance Sampling is a variance reduction sampling technique used to compute Monte Carlo estimations of averages of the type $E[h(x)] = \frac{1}{||A||} \int_A h(x)dx$, where $h(x)$ is a real function taking values in A. This requires sampling the domain from a non necessarily uniform distribution f, thus yielding

$$E_f[h(x)] \approx \frac{1}{k} \sum_{i=1}^{k} h(x_i) \frac{\frac{1}{||A||}}{f(x_i)} \tag{6}$$

where $\frac{\frac{1}{||A||}}{f(x_i)}$ is the *importance factor* used to correct the bias introduced when sampling from the distribution f. Note that in the limit if $f = \frac{h(x)}{\int_A h(x)dx}$ then the variance of the estimator is zero. In practice then, we would like choose f as close as possible to $\frac{h(x)}{\int_A h(x)dx}$.

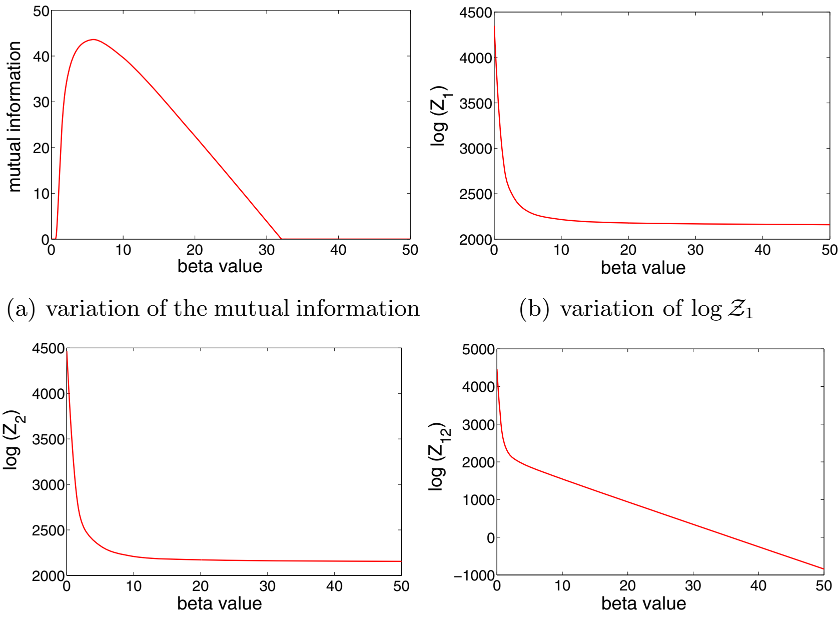
In this paper, we need to approximate the value of the partition functions $\mathcal{Z}_1$, $\mathcal{Z}_2$ and $\mathcal{Z}_{12}$. Since the approximation procedure is going to be the same in all the three cases, we simply review the equations for $\mathcal{Z}_1$. In this case, $||A|| = n!$ and $h(x) = \exp[-\beta \Delta R_1(c_1)]$, and thus

$$\mathcal{Z}_1 = E_{c_1}\Big[\exp[-\beta \Delta R_1(c_1)]\Big] n! \approx \frac{1}{|\mathcal{C}_1|} \sum_{c_1 \in \mathcal{C}_1}^{k} \frac{\exp[-\beta \Delta R_1(c_1)]}{P(c_1)} \tag{7}$$

To implement the importance sampler along the lines suggested in [2], recall that $\Delta R_q = R_q(c_q) - R_q(c_q^{\perp})$ and $R_q(c_q) = -\log P(\mathcal{G}_q | G_M^{(q)}, c_q)$, where G_q is the observed graph and $G_M^{(q)}$ is the prototype graph. We aim to sample a mapping $c_q \in \mathcal{C}_q$ with probability close to $\frac{P(\mathcal{G}_q | G_M^{(q)}, c_q)}{\sum_{c_q \in \mathcal{C}_q} P(\mathcal{G}_q | G_M^{(q)}, c_q)}$. The procedure is as follows. Assume that we know the node-correspondence matrix $\bar{M} = (m_{\alpha a})$ giving the probability that graph node a was generated by prototype node α. Then we can first sample a correspondence for the prototype node 1 with probability m_{1a_1}. The next step is to condition the matrix to the current match by taking into account the structural information between the sampled nodes and all the remainder. Finally we project the conditioned matrix onto a double-stochastic matrix by using the Sinkhorn process [16], yielding the matrix $\bar{M}_1^{a_1}$. We repeat this procedure for each node of the prototype graph, until we have sampled a mapping c_q with probability $P(c_q) = (\bar{M})_{1,a_1} \cdot (\bar{M}_1^{a_1})_{2,a_2} \cdot \ldots \cdot (\bar{M}_{1,\ldots,n-1}^{a_1,\ldots,a_{n-1}})_{n,a_n}$.

4 Experiments

In this section, we report some experimental results of the application of our prototype size selection method on real-world dataset. The dataset used is the



(a) variation of the mutual information

(b) variation of $\log \mathcal{Z}_1$

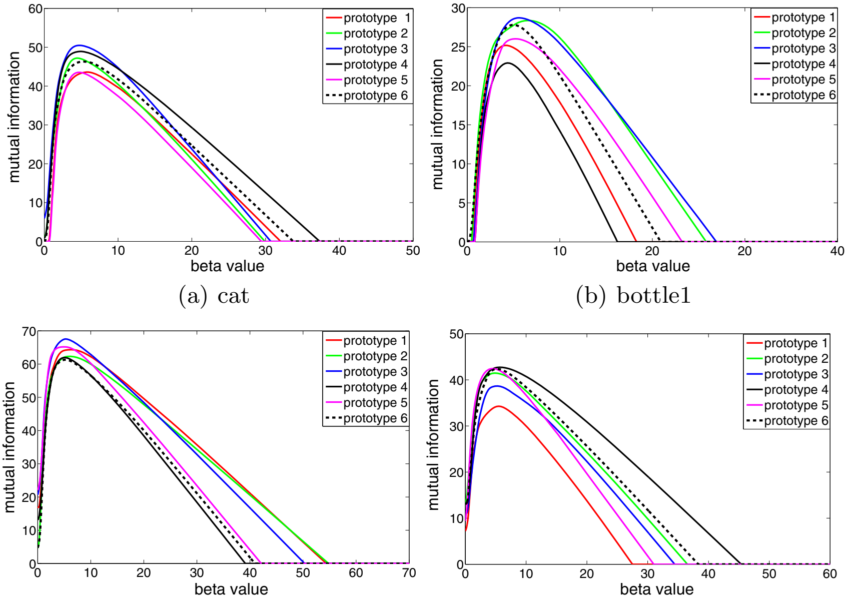
(c) variation of $\log \mathcal{Z}_2$

(d) variation of $\log \mathcal{Z}_{12}$

Fig. 2. How the mutual information and logarithm of partition functions change as β increases from 0 to 50

COIL [5] which consists of images of different objects, with 72 views of each object obtained from equally spaced viewing directions over 360°. We extract corner features from each image and use the detected feature points as nodes to construct sample graphs by Delaunay triangulation.

We first investigate how the value of the mutual information and the three partition functions vary as the value of β increases. To do this, we randomly partition the graphs from a given object, e.g. the cat images, into a training set and a test set which are of the same size. The bijective mapping of the graphs between the two sets is located by minimizing the sum of the edit distances between the mapped graphs. We learn two prototype graphs of the same size for the two sets using the method in [1]. Given this setting, we compute the value of the mutual information and the logarithms of the three partition functions $\log \mathcal{Z}_1$, $\log \mathcal{Z}_2$ and $\log \mathcal{Z}_{12}$. Figure 2 shows how these quantities vary as we increase the value of β from 0 to 50. From the plot in Figure 2(a), we observe that the mutual information initially increases and achieves the highest value around $\beta = 8$, and afterwards it begins to decrease. To maintain the non negativity of the mutual information, we set its value to zero when it falls below zero. Figure 2(b) and 2(c) respectively show the value of the logarithms of partition functions $\log \mathcal{Z}_1$ and $\log \mathcal{Z}_2$. From the plots it is clear that these two quantities converge to a horizontal asymptote. The reason for this is that while the relative cost of the optimal hypothesis is zero and thus its contribution to the partition function is a constant positive value, the



(a) cat

(b) bottle1

(c) pig

(d) bottle2

Fig. 3. Variation of the mutual information of 6 prototype graphs of the four objects

exponential of the relative costs given by the non optimal hypotheses converges to zero as β increases, thus yielding the observed horizontal asymptote. On the other hand, the logarithm of the joint partition function $\log \mathcal{Z}_{12}$ in Figure 2(d) continues to decrease as β increases. This indicates that the optimal hypotheses of the graphs in the test set do not necessarily generalize to the optimal hypotheses of their mapped graphs in the training set. For this reason the relative costs of all the hypotheses in the joint partition function are positive values. As a result their exponentials converge to zero as β increases. Consequently, the joint partition function converges to zero and its corresponding logarithm becomes both large and negative.

Our second experimental goal is to select the optimal size of the prototype graph for several objects from the COIL dataset. Here the objects we used are the cat, pig and two bottles. To perform these tasks, for each object we learn 6 prototype graphs of different size using the method in [1] and then compute the mutual information of these prototype graphs. The optimal size of the prototype graph is that which gives the highest mutual information as β is varied. Figure 3 shows plots of the mutual information versus β for the four objects. From the plots it is clear that for each objet there is a prototypes size that gives optimized performance. Finally, note that unlike what is expected using other standard model complexity selection methods, which may choose the model with the smallest size, in our experiments we observe that in 3 out of 4 objects the proposed method favours some value between the largest and the smallest size.

5 Conclusion

In this paper we have developed a method for selecting the optimal size of a prototype graph used to represent a set of sample graphs. The optimal size of the prototype graph is selected so as to maximize the mutual information of the two partitioned sets of the sample graphs. To compute the mutual information, we extend the theory of approximate set coding from the vector domain to the graph domain. Experimental results show that our method works well for prototype graph selection in object recognition. Future work will concentrate on validating our prototype graph size selection method. Moreover, while the prototype selection step is currently a separate post processing step which takes place after the learning procedure, we intend to investigate how to integrate the two together, so as to reduce the overall complexity.

Acknowledgement. Edwin Hancock was supported by a Royal Society Wolfson Research Merit Award.

References

1. Han, L., Wilson, R.C., Hancock, E.R.: A Supergraph-based Generative Model. In: ICPR, pp. 1566–1569 (2010)
2. Torsello, A.: An Importance Sampling Approach to Learning Structural Representations of Shape. In: CVPR, pp. 1–7 (2008)
3. Buhmann, J.M., Chehreghani, M.H., Frank, M., Streich, A.P.: Information Theoretic Model Selection for Pattern Analysis. JMLR: Workshop and Conference Proceedings 7, 1–15 (2011)
4. Buhmann, J.M.: Information Thereotic Model Validation for Clustering. In: International Symposium on Information Theory, pp. 1398–1402 (2010)
5. Nene, S.A., Nayar, S.K., Murase, H.: Columbia Object Image Library(COIL100). Columbia University (1996)
6. National Center for Biotechnology Information, `http://www.ncbi.nlm.nih.gov`
7. Hammersley, J.M., Handscomb, D.C.: Monte Carlo Methods. Wiley, New York (1964)
8. Han, L., Hancock, E.R., Wilson, R.C.: Learning Generative Graph Prototypes Using Simplified von Neumann Entropy. In: GbRPR, p. 4251 (2011)
9. Rissanen, J.: Modelling by Shortest Data Description. Automatica 14, 465–471 (1978)
10. Schwarz, G.E.: Estimating the dimension of a model. Annals of Statistics 6, 461–464 (1978)
11. Foster, D.P., George, E.I.: The Risk Inflation Criterion for Multiple Regression. Annals of Statistics 22, 1947–1975 (1994)
12. White, D., Wilson, R.C.: Parts based generative models for graphs. In: ICPR, pp. 1–4 (2008)
13. Akaike, H.: A new look at the statistical model identification. IEEE Transactions on Automatic Control 19, 716–723 (1974)
14. Grnwald, P.D., Myung, I.J., Pitt, M.A.: Advances in Minimum Description Length: Theory and Applications. The MIT Press (2005)
15. Luo, B., Hancock, E.R.: Structural graph matching using the EM alogrithm and singular value decomposition. IEEE Transactions on PAMI 23, 1120–1136 (2001)
16. Sinkhorn, R.: A Relationship Between Arbitrary Positive Matrices and Doubly Stochastic Matrices. The Annals of Mathematical Statistics 35, 876–879 (1964)