Department of
Information Engineering
and Computer Science
DISI

UNIVERSITY
OF TRENTO - Italy

DISI - Via Sommarive, 5 - 38123 POVO, Trento - Italy
http://disi.unitn.it

CLASSTERING: A SEMI-SUPERVISED ALGORITHM BASED ON MIXTURE OF FACTOR ANALYSERS

E. Sansone, F. G. B. De Natale

April 2015

Technical Report # DISI-15-009

Abstract

In this work we propose a novel parametric Bayesian model to solve the problem of semi-supervised learning, including classification and clustering. Standard approaches of semi-supervised classification can recognize classes but cannot find groups of data. On the other hand, semi-supervised clustering techniques are able to discover groups of data but cannot find the associations between clusters and classes. The proposed model can classify and cluster samples simultaneously, leading to the possibility of solving the problem of annotation in the presence of an unknown number of classes and/or arbitrary number of clusters per class. Preliminary results performed on synthetic datasets show the effectiveness of the framework.

1 Introduction

Semi-supervised learning (SSL) is a well-known area of machine learning. The main idea is to exploit both labeled and unlabeled data to increase the performance of classification and clustering. This is motivated by the fact that unlabeled data are largely available and can be collected at low cost. Furthermore, Zhu et al. [1] have demonstrated that human beings learn in a semi-supervised way. In their experiments, participants were asked to classify visual stimuli coming from two classes of pollen particles. The authors noticed that a change in the decision strategy happens when unlabeled data are used.
For semi-supervised learning, many approaches [2] have been proposed, including

- **Self-training algorithms.** Basically, a classifier is trained first by exploiting the labeled samples, and secondly by using the most confident samples together with their predicted labels. These methods are straightforward but are sensitive to the error propagation of mislabeled data.
- **Generative models.** They are based on the assumption that observed data are generated by an underlying probability density function. The estimation of the distribution can be performed in two different ways, namely through *parametric* and *non-parametric* approaches. In the first case, a family of parametric distributions is chosen, and the parameters are estimated in order to fit properly the observations. The main concern of these approaches is the model selection. In fact, if the model is not adeguate, then the estimator may become biased as the number of unlabeled samples grows [3].
  In contrast, non-parametric approaches make as few assumptions as possible about the shape of the real distribution, and therefore they may be applied to a large variety of problems. For example, Anand et al. [4] proposed a non-parametric framework for semi-supervised clustering, namely the semi-supervised kernel mean shift clustering (SKMS). The algortihm first mappes input data into a high-dimensional kernel space, fulfilling the constraints imposed by the user, and secondly clusters data with the mean shift clustering algorithm.
- **Transductive SVM.** Also called semi-supervised SVM (S3VM). This is the natural extension of the standard SVM to semi-supervised learning. Basically, the algorithm starts by enumerating all possible labelings for unlabeled data. Secondly, it trains a standard SVM for each possible combination of labels, and finally retains the classifier with the largest margin [5]. Although the method provides low misclassification rate, the search of the optimal solution is NP-hard. For this purpose, many solutions have been proposed. For example, Joachims [6] proposed an optimization algorithm for S3VMs, called S3VM^{light}, where the space of possible solutions is reduced thanks to a local combinatorial search. Chapelle et al. [7] applied branch and bound techniques to S3VM in order to find global optimal solutions.
- **Graph-based methods.** They represent data by a graph, where nodes are labeled and unlabeled samples, and links are weighted according to a predefined metric of similarity between nodes. There is a function associated to the graph, which defines the mapping between the sample and the label domain. The function has to fulfill two requirements, namely (i) it has to approximate well the given labels of the labeled nodes, and (ii) it has

to be smooth enough in order to avoid the oversegmentation of the graph. The problem can be therefore cast into an optimization task. Blum and Chawla [8] proposed to solve the problem through the mincut algorithm. In the binary case, positive samples are considered as sources, while negative samples act as sinks. The algorithm propagates labels to unlabeled samples after computing the cuts in the graph. Zhu and Ghahramani [9] formulated the problem as a fully connected Boltzmann machine. After learning the parameters of the model, they were able to find an approximate solution through MCMC methods. A more recent work [10] relaxed the assumption of discrete labels to cope with the continuous space by proposing a model based on a Gaussian random field. It has been proven that in this case the obtained solution is a harmonic function. Despite the clear mathematical formulation of graph-based methods, their performance strongly depends on the graph structure and the definition of the similarity metric.

- **Multiview algorithms.** These methods assume that data can be represented by multiple views. For example, if an item has a textual and a visual representation, then a classifier for text and one for images are trained separately with the labeled data. After that, the classifiers predict the labels of all unlabeled data. Finally, the classifiers are retrained by exchanging the predicted labels [11]. With respect to self-training approaches, these methods are less sensitive to mistakes, but rely on the assumptions that there exist multiple representations of the same item.

Beyond that, it is necessary to emphasize the difference between semi-supervised clustering algorithms and semi-supervised classification methods. In the former case, labeled information is often cast in the form of pairwise constraints (e.g. must- and cannot-links) and is used to guide the optimization procedure to find the groups of data [4, 12, 13, 14, 15]. In the latter case, labels indicate the membership of samples to the available classes and the goal is to exploit that information to learn the mapping function between the sample and the class space [5, 16, 17, 18, 19]. To the best of our knowledge, none of the existing algorithms are able to perform clustering and classification, simultanelously. In that regard, our method can satisfy the requirement, and leads to the possibility of solving the problem of annotation with an unknown number of classes.

2 The model

Given a set of N observations $Y = \{\mathbf{y}_n\}_{n=1}^N$, where $y \in \mathbb{R}^d$, and the respective set of labels $C = \{c_n\}_{n=1}^N$, where c_n specifies that $\mathbf{y}_n$ belongs to one among K predefined classes, the goal is to recover the underlying distribution generating the observations.
In particular, we assume without loss of generality that the latent structure is a Gaussian mixture.[1] As a consequence, we need to automatically determine the number of components of the mixture and their local dimensionality, in order to be able to model the widest range of probability distributions.
Ghahramani and Beal [21] proposed a fully-Bayesian model based on mixture of factor analysers to select automatically the number of components. A stochastic procedure is used in the variational optimisation to avoid local maxima.
Since their model is well-suited for unsupervised problems, our purpose consists of extending it to the semi-supervised setting.

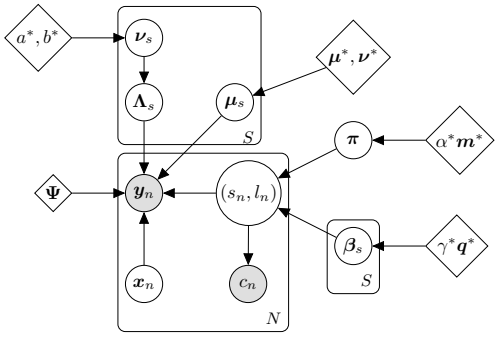


Figure 1: Generative model.

[1] Since almost any kind of distribution can be arbitrarily approximated by an appropriate Gaussian mixture [20].

We assume that each data vector $\mathbf{y}_n$ is generated independently from the same mixture of factor analysers (MFA) and the density can be modeled according to

$$p(\mathbf{y}_n|\boldsymbol{\mu}, \boldsymbol{\Lambda}, \boldsymbol{\Psi}, \boldsymbol{\pi}, \mathbf{x}_n) = \sum_{s_n=1}^{S} p(s_n|\pi) p(\mathbf{y}_n|s_n, \boldsymbol{\mu}_{s_n}, \boldsymbol{\Lambda}_{s_n}, \boldsymbol{\Psi}, \mathbf{x}_n) \tag{1}$$

where $\boldsymbol{\pi}$ is the vector of the mixing proportions, s_n is an indicator variable activating one of the S factor analysers, $\boldsymbol{\Lambda}_{s_n}$ is the factor loading matrix for factor analyser s_n, $\mathbf{x}_n$ is the latent factor of $\mathbf{y}_n$ and $\boldsymbol{\Psi}$ corresponds to the covariance matrix associated with the additive Gaussian noise in the MFA model. Based on that, it is possible to show that $p(\mathbf{y}_n|s_n, \boldsymbol{\mu}_{s_n}, \boldsymbol{\Lambda}_{s_n}, \boldsymbol{\Psi})$ follows a Gaussian distribution with mean value $\boldsymbol{\mu}_{s_n}$ and covariance matrix equal to $\boldsymbol{\Lambda}_{s_n}\boldsymbol{\Lambda}_{s_n}^T + \boldsymbol{\Psi}$ [21]. In particular, $\boldsymbol{\Lambda}_{s_n}$ takes into account the dimensionality of component s_n, while $\boldsymbol{\Psi}$ models the variability of data inside that component, namely the noise variance. The parameters can be estimated for each analyser in order to model properly the observed set Y.
We introduce a latent variable called I_n, defined as a pair of variables (s_n, l_n), where l_n represents an indicator of the class which $\mathbf{y}_n$ belongs to. I_n is modeled by a distribution which takes into account all possible combinations between Gaussians and classes. In order to guarantee an univocal association between Gaussians and classes,[2] we introduce a set of random vectors, called $\{\boldsymbol{\beta}_s\}_{s=1}^S$, governed by Dirichlet priors, where the i-th component of the random vector $\boldsymbol{\beta}_s$ represents the probability that Gaussian s is associated with class i.
Figure 1 shows the complete probabilistic graphical model. The complete set of conditional distributions and priors is defined as follows:

$$\begin{aligned}
&p(I_n|\boldsymbol{\pi}, \{\boldsymbol{\beta}_s\}_{s=1}^S) \doteq \boldsymbol{\pi}(s_n)\boldsymbol{\beta}_{s_n}(l_n)\\
&p(c_n|I_n) \doteq \delta(c_n - l_n)\\
&p(\boldsymbol{\Lambda}_s|\boldsymbol{\nu}_s) \doteq \prod_{j=1}^{d} \mathcal{N}(0, I/\boldsymbol{\nu}_s(j))\\
&p(\boldsymbol{\nu}_s|a^*, b^*) \doteq \prod_{j=1}^{d} Ga(\boldsymbol{\nu}_s(j)|a^*, b^*)\\
&p(\boldsymbol{\mu}_s|\boldsymbol{\mu}^*, \boldsymbol{\nu}^*) \doteq \mathcal{N}(\boldsymbol{\mu}^*, diag(\boldsymbol{\nu}^*)^{-1})\\
&p(\boldsymbol{\pi}|\alpha^*\boldsymbol{m}^*) \doteq Dir(\alpha^*\boldsymbol{m}^*)\\
&p(\boldsymbol{\beta}_s|\gamma^*\boldsymbol{q}^*) \doteq Dir(\gamma^*\boldsymbol{q}^*)\\
&p(\boldsymbol{x}_n) \doteq \mathcal{N}(0, I)
\end{aligned}$$

where I is the identity matrix and $\boldsymbol{\nu}_s$ is a d-dimensional vector whose elements govern the columns of $\boldsymbol{\Lambda}_s$. This mechanism is known with the name of automatic relevance determination (ARD) which is used for example to improve the task of dimensionality reduction [22].
If we define $\mathcal{H} = \{\boldsymbol{x}_n, I_n\}$ as the set of hidden variables and $\Theta = \{\boldsymbol{\pi}, \{\boldsymbol{\beta}_s, \boldsymbol{\Lambda}_s, \boldsymbol{\mu}_s, \boldsymbol{\nu}_s\}_{s=1}^S\}$ as the set of parameters, then we can express the log-likelihood function of Y and C as

$$\ln p(Y, C) = \ln \int d\Theta p(\Theta) \int d\mathcal{H} p(Y, C, \mathcal{H}|\Theta)$$

and by exploiting the conditional dependencies defined by the graphical model we obtain that

$$\ln p(Y, C) = \ln \int d\Theta p(\Theta) \prod_{n=1}^{N} \sum_{s_n=1}^{S} \sum_{l_n=1}^{K} p(I_n|\Theta) p(c_n|I_n) \int d\boldsymbol{x}_n p(\boldsymbol{x}_n) p(\boldsymbol{y}_n|\Theta, \boldsymbol{x}_n, I_n, \boldsymbol{\Psi})$$

which can be lower bounded by using the variational approximation [23], namely

[2] Necessary to predict labels of new samples.

$$
\begin{aligned}
\ln p(Y,C) \geq & \int d\boldsymbol{\pi} q(\boldsymbol{\pi}) \ln \frac{p(\boldsymbol{\pi}|\alpha^* \boldsymbol{m}^*)}{q(\boldsymbol{\pi})} + \sum_{s=1}^{S} \int d\boldsymbol{\beta}_s q(\boldsymbol{\beta}_s) \ln \frac{p(\boldsymbol{\beta}_s|\gamma^* \boldsymbol{q}^*)}{q(\boldsymbol{\beta}_s)} \\
& + \sum_{s=1}^{S} \int d\boldsymbol{\nu}_s q(\boldsymbol{\nu}_s) \Big[\ln \frac{p(\boldsymbol{\nu}_s|a^*, b^*)}{q(\boldsymbol{\nu}_s)} + \int d\tilde{\boldsymbol{\Lambda}}_s q(\tilde{\boldsymbol{\Lambda}}_s) \ln \frac{p(\tilde{\boldsymbol{\Lambda}}_s|\boldsymbol{\nu}_s, \boldsymbol{\mu}^*, \boldsymbol{\nu}^*)}{q(\tilde{\boldsymbol{\Lambda}}_s)} \Big] \\
& + \sum_{n=1}^{N} \sum_{s_n=1}^{S} \sum_{l_n=1}^{K} q(I_n) \Big[\int d\boldsymbol{\pi} q(\boldsymbol{\pi}) \int d\boldsymbol{\beta}_{s_n} q(\boldsymbol{\beta}_{s_n}) \ln \frac{p(I_n|\boldsymbol{\pi}, \boldsymbol{\beta}_{s_n})}{q(I_n)} \\
& + \int d\boldsymbol{x}_n q(\boldsymbol{x}_n|I_n) \ln \frac{p(\boldsymbol{x}_n)}{q(\boldsymbol{x}_n|I_n)} + \ln p(c_n|I_n) \\
& + \int d\tilde{\boldsymbol{\Lambda}}_s q(\tilde{\boldsymbol{\Lambda}}_s) \int d\boldsymbol{x}_n q(\boldsymbol{x}_n|I_n) \ln p(\boldsymbol{y}_n|\tilde{\boldsymbol{\Lambda}}_s, \boldsymbol{x}_n, I_n, \boldsymbol{\Psi}) \Big] \\
\doteq & \mathcal{F}\big(q(\boldsymbol{\pi}), \{q(\boldsymbol{\beta}_s), q(\boldsymbol{\nu}_s), q(\tilde{\boldsymbol{\Lambda}}_s)\}_{s=1}^{S}, \{q(I_n), \boldsymbol{x}_n|I_n)\}_{n=1}^{N}\big)
\end{aligned} \tag{2}
$$

where $q(\cdot)$ represent the variational posteriors over hidden variables or parameters and $\tilde{\boldsymbol{\Lambda}}_s$ represents the concatenation between $\boldsymbol{\Lambda}_s$ and $\boldsymbol{\mu}_s$.
By maximizing the functional $\mathcal{F}$, the lower bound becomes tight to the true log-likelihood function $\ln p(Y, C)$ and can be used as an approximation of it. In this framework $\mathcal{F}$ is then used to perform model comparison in order to choose automatically the best values for S.
The model can be easily extended to perform semi-supervised learning by introducing the set of unlabeled observations, namely $Y' = \{\mathbf{y'}_m\}_{m=1}^{N'}$ and averaging over all possible labels. The extended log-likelihood function can be lower bounded in the following way:

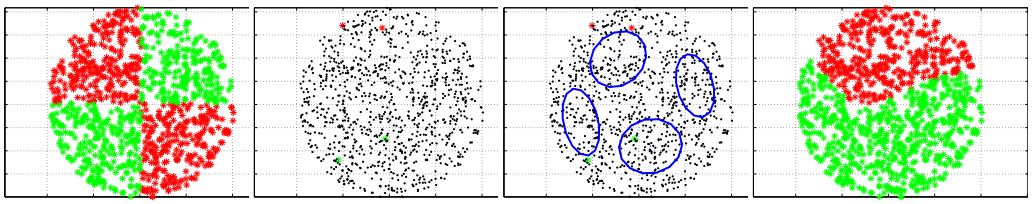
$$
\begin{aligned}
\ln p(Y, Y', C) \geq & \mathcal{F}\big(q(\boldsymbol{\pi}), \{q(\boldsymbol{\beta}_s), q(\boldsymbol{\nu}_s), q(\tilde{\boldsymbol{\Lambda}}_s)\}_{s=1}^{S}, \{q(I_n), \boldsymbol{x}_n|I_n)\}_{n=1}^{N}\big) \\
& + \sum_{m=1}^{N'} \sum_{s_m=1}^{S} \sum_{l_m=1}^{K} q(I_m) \Big[\int d\boldsymbol{\pi} q(\boldsymbol{\pi}) \int d\boldsymbol{\beta}_{s_m} q(\boldsymbol{\beta}_{s_m}) \ln \frac{p(I_m|\boldsymbol{\pi}, \boldsymbol{\beta}_{s_m})}{q(I_m)} \\
& + \int d\boldsymbol{x}_m q(\boldsymbol{x}_m|I_m) \ln \frac{p(\boldsymbol{x}_m)}{q(\boldsymbol{x}_m|I_m)} \\
& + \int d\tilde{\boldsymbol{\Lambda}}_s q(\tilde{\boldsymbol{\Lambda}}_s) \int d\boldsymbol{x}_m q(\boldsymbol{x}_m|I_m) \ln p(\boldsymbol{y'}_m|\tilde{\boldsymbol{\Lambda}}_s, \boldsymbol{x}_m, I_m, \boldsymbol{\Psi}) \Big]
\end{aligned} \tag{3}
$$

where the last three terms are related to unlabeled data.

3 Preliminary results

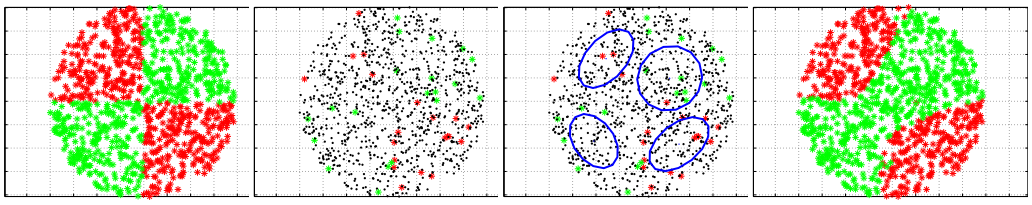
We present some qualitative results on several synthetic datasets. Figures 2-4 show the case where data are uniformly distributed in a round shape and the classes are not linearly separable.[3] For this kind of distribution, traditional clustering techniques can only identify one single cluster and supervised techniques can not exploit unlabeled samples. Our framework adopts both labeled and unlabeled data to estimate the underlying distribution. With the number of labeled samples increase, the performance of our framework improves significantly.
Figures 5-7 show the results for linearly separable data where each class consists of multiple clusters. In particular, it is worth to note that labels are inferred correctly even when only two labeled samples per class are provided.[4] In Figure 5(b), there are some clusters without labels, namely the topmost and the rightmost cluster. In this case, more labels are required in order to determine if these groups belong to some new class or not. Existing semi-supervised algorithms can not perform clustering and classification simultaneously, therefore they can not deal with this case.
Figure 8-9 shows the results of other synthetic datasets[5]. In both cases, data are distributed in two

[3] In the pictures, colors are used to highlight different classes. Black points are unlabeled samples.
[4] Except for the boundary region.
[5] `http://www.uni-marburg.de/fb12/datenbionik/data?language_sync=1`



(a) True labels (b) Some labeled samples (c) Estimated model (d) Inferred labels

Figure 2: Cake - Synthetic dataset with $N = 1000$ and 2 labeled samples per class.



(a) True labels (b) Some labeled samples (c) Estimated model (d) Inferred labels

Figure 3: Cake - Synthetic dataset with $N = 1000$ and 20 labeled samples per class.

different manifolds with few labeled samples. This implies the possibility of applying the framework to high-dimensional space, where data lie on a lower-dimensional manifold.

References

[1] Xiaojin Zhu, Timothy Rogers, Ruichen Qian, and Chuck Kalish. Humans perform semi-supervised classification too. In *Proceedings of the National Conference on Artificial Intelligence*, volume 22, page 864. Menlo Park, CA; Cambridge, MA; London; AAAI Press; MIT Press; 1999, 2007.

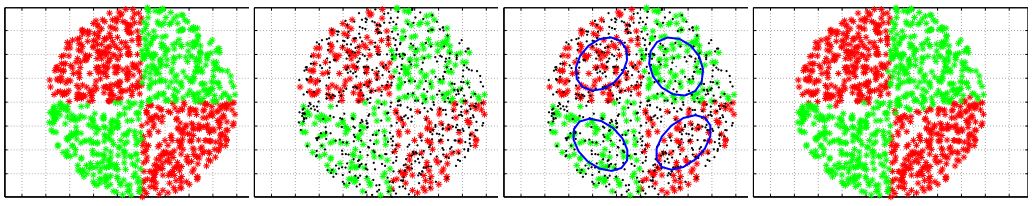
[2] Xiaojin Zhu. Semi-supervised learning literature survey. 2005.

[3] Fabio Gagliardi Cozman, Ira Cohen, Marcelo Cesar Cirelo, et al. Semi-supervised learning of mixture models. In *ICML*, pages 99–106, 2003.

[4] Saket Anand, Sushil Mittal, Oncel Tuzel, and Peter Meer. Semi-supervised kernel mean shift clustering. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 36(6):1201–1215, 2014.

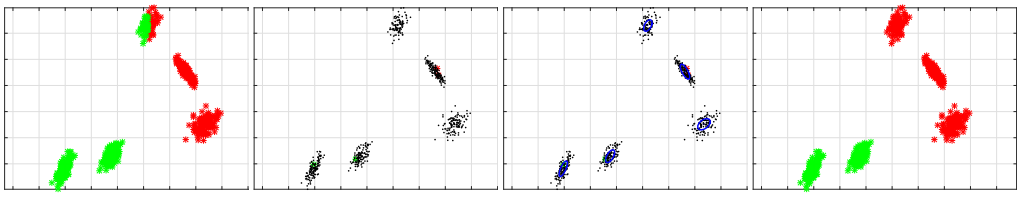
[5] Kristin Bennett, Ayhan Demiriz, et al. Semi-supervised support vector machines. *Advances in Neural Information processing systems*, pages 368–374, 1999.

[6] Thorsten Joachims. Transductive inference for text classification using support vector machines. In *ICML*, volume 99, pages 200–209, 1999.



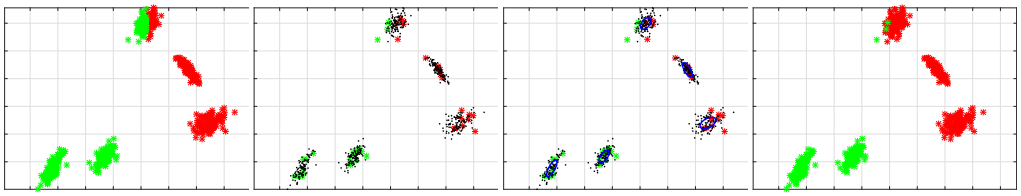
(a) True labels (b) Some labeled samples (c) Estimated model (d) Inferred labels

Figure 4: Cake - Synthetic dataset with $N = 1000$ and 200 labeled samples per class.



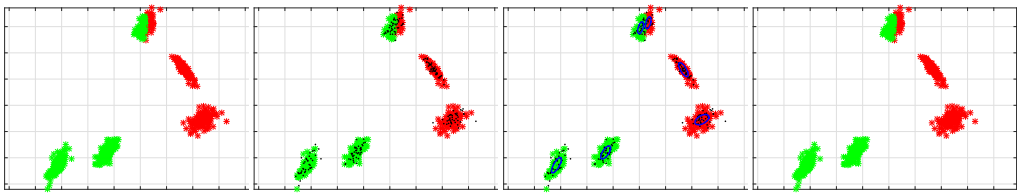
(a) True labels (b) Some labeled samples (c) Estimated model (d) Inferred labels

Figure 5: Gauss - Synthetic dataset with $N = 600$ and 2 labeled samples per class.



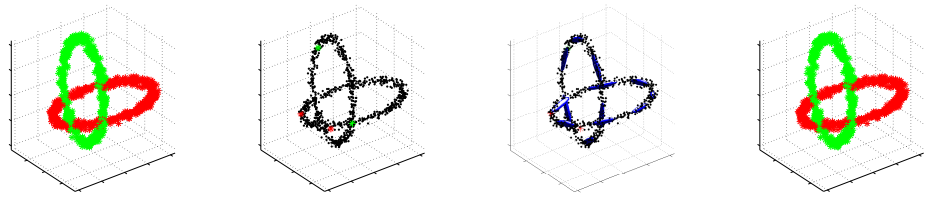
(a) True labels (b) Some labeled samples (c) Estimated model (d) Inferred labels

Figure 6: Gauss - Synthetic dataset with $N = 600$ and 20 labeled samples per class.



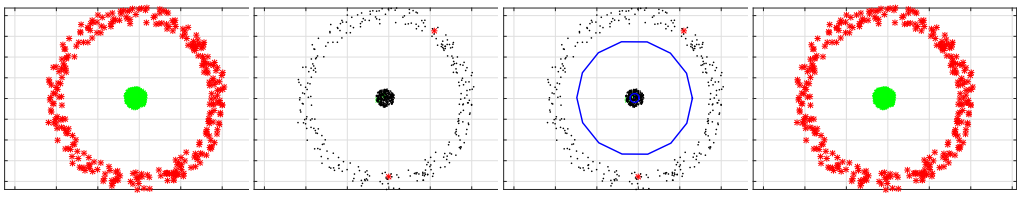
(a) True labels (b) Some labeled samples (c) Estimated model (d) Inferred labels

Figure 7: Gauss - Synthetic dataset with $N = 600$ and 200 labeled samples per class.



(a) True labels (b) Some labeled samples (c) Estimated model (d) Inferred labels

Figure 8: Chain - Synthetic dataset with $N = 1000$ and 2 labeled samples per class.



(a) True labels (b) Some labeled samples (c) Estimated model (d) Inferred labels

Figure 9: Ring - Synthetic dataset with $N = 1000$ and 2 labeled samples per class.

[7] Olivier Chapelle, Vikas Sindhwani, and SS Keerthi. Branch and bound for semi-supervised support vector machines. In *Advances in Neural Information Processing Systems*, pages 217–224, 2007.

[8] Avrim Blum and Shuchi Chawla. Learning from labeled and unlabeled data using graph mincuts. In *Proceedings of the Eighteenth International Conference on Machine Learning*, pages 19–26. Morgan Kaufmann Publishers Inc., 2001.

[9] Xiaojin Zhu and Zoubin Ghahramani. Towards semi-supervised classification with markov random fields. 2002.

[10] Xiaojin Zhu, Zoubin Ghahramani, John Lafferty, et al. Semi-supervised learning using gaussian fields and harmonic functions. In *ICML*, volume 3, pages 912–919, 2003.

[11] Avrim Blum and Tom Mitchell. Combining labeled and unlabeled data with co-training. In *Proceedings of the eleventh annual conference on Computational learning theory*, pages 92–100. ACM, 1998.

[12] Brian Kulis, Sugato Basu, Inderjit Dhillon, and Raymond Mooney. Semi-supervised graph clustering: a kernel approach. *Machine learning*, 74(1):1–22, 2009.

[13] Noam Shental, Aharon Bar-hillel, Tomer Hertz, and Daphna Weinshall. Computing gaussian mixture models with em using equivalence constraints. In *Advances in Neural Information Processing Systems*, pages 465–472, 2004.

[14] Mikhail Bilenko, Sugato Basu, and Raymond J Mooney. Integrating constraints and metric learning in semi-supervised clustering. In *Proceedings of the twenty-first international conference on Machine learning*, page 11. ACM, 2004.

[15] Jinfeng Yi, Lijun Zhang, Rong Jin, Qi Qian, and Anil Jain. Semi-supervised clustering by input pattern assisted pairwise similarity matrix completion. In *Proceedings of the 30th International Conference on Machine Learning (ICML-13)*, pages 1400–1408, 2013.

[16] Diederik P Kingma, Shakir Mohamed, Danilo Jimenez Rezende, and Max Welling. Semi-supervised learning with deep generative models. In *Advances in Neural Information Processing Systems*, pages 3581–3589, 2014.

[17] Mikhail Belkin, Partha Niyogi, and Vikas Sindhwani. Manifold regularization: A geometric framework for learning from labeled and unlabeled examples. *The Journal of Machine Learning Research*, 7:2399–2434, 2006.

[18] Olivier Chapelle and Alexander Zien. Semi-supervised classification by low density separation. *AISTATS 2005*, page 57, 2005.

[19] Neil D Lawrence and Michael I Jordan. Semi-supervised learning via gaussian processes. In *Advances in neural information processing systems*, pages 753–760, 2004.

[20] Walter Rudin. Functional analysis. international series in pure and applied mathematics, 1991.

[21] Zoubin Ghahramani, Matthew J Beal, et al. Variational inference for bayesian mixtures of factor analysers. In *NIPS*, pages 449–455, 1999.

[22] CM Bishop. Variational principal components. In *Artificial Neural Networks, 1999. ICANN 99. Ninth International Conference on (Conf. Publ. No. 470)*, volume 1, pages 509–514. IET, 1999.

[23] Hagai Attias. Inferring parameters and structure of latent variable models by variational bayes. In *Proceedings of the Fifteenth conference on Uncertainty in artificial intelligence*, pages 21–30. Morgan Kaufmann Publishers Inc., 1999.