Constrained Clustering: Effective Constraint Propagation with Imperfect Oracles

Xiatian Zhu[1], Chen Change Loy[2], Shaogang Gong[1]
[1]Queen Mary, University of London, London E1 4NS, UK
[2]The Chinese University of Hong Kong, Shatin, N.T., Hong Kong
xiatian.zhu@eecs.qmul.ac.uk, ccloy@ie.cuhk.edu.hk, sgg@eecs.qmul.ac.uk

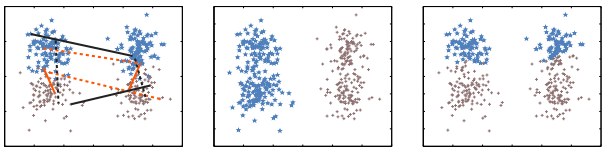
***Abstract*—While spectral clustering is usually an unsupervised operation, there are circumstances in which we have prior belief that pairs of samples should (or should not) be assigned with the same cluster. Constrained spectral clustering aims to exploit this prior belief as constraint (or weak supervision) to influence the cluster formation so as to obtain a structure more closely resembling human perception. Two important issues remain open: (1) how to propagate sparse constraints effectively, (2) how to handle ill-conditioned/noisy constraints generated by imperfect oracles. In this paper we present a unified framework to address the above issues. Specifically, in contrast to existing constrained spectral clustering approaches that blindly rely on all features for constructing the spectral, our approach searches for neighbours driven by discriminative feature selection for more effective constraint diffusion. Crucially, we formulate a novel data-driven filtering approach to handle the noisy constraint problem, which has been unrealistically ignored in constrained spectral clustering literature.**

***Keywords*-Constrained clustering, constraint propagation, feature selection, imperfect oracles, spectral clustering.**

I. Introduction

Constrained clustering has been studied extensively [11], [9]. The objective is to effectively exploit a small amount of supervision to help finding data partitions that capture consistent concepts as perceived by human. The supervision by oracles is typically expressed in the form of pairwise constraint, namely *must-link* - a pair of samples must be in the same cluster, and *cannot-link* - a pair of samples belong to different clusters. Though great strides have been made in this field, two important and non-trivial questions remain open.

(I) *Effective sparse constraint propagation*: Pairwise constraints are sparse in practice since exhaustive pairwise labelling are laborious and/or may not be available in data. Constraint propagation [9] is thus designed to propagate pairwise constraints from labelled samples to unlabelled samples for maximising the influence of constraints. Effective constraint propagation relies on robust identification of unlabelled nearest neighbours (NN) around the labelled samples in the feature space. Often, the NN search is susceptible to noisy or ambiguous features, especially so on image and video datasets. Trusting all the available features blindly for NN search (as what most existing constrained clustering approaches [11], [9] did) is likely to result in sub-optimal constraint diffusion. It is challenging to determine how to propagate their influence effectively to neighbouring unlabelled points. In particular, it is non-trivial to reliably identify the neighbouring unlabelled points for propagation.



(a) Ground Truth (b) Spectral Clustering (c) Our Approach

Figure 1. (a) Ground truth cluster formation, with invalid pairwise constraints highlighted in red colour; must- and cannot-links are represented by solid and dashed lines respectively; (b) the clustering result obtained using unsupervised clustering; (c) the result obtained using our method.

(II) *Noisy constraints from imperfect oracles*: Human annotators (oracles) may provide invalid/mistaken constraints. For instance, a portion of 'must-links' are actually 'cannot-links' and vice versa. For example, annotations or constraints obtained from online crowdsourcing services, e.g. Amazon Mechanical Turk, are very likely to contain errors or noises due to data ambiguity, unintentional human mistakes or even intentional errors by malicious workers. Learning such constraints blindly may result in sub-optimal cluster formation. Most existing methods make an unrealistic assumption that constraints are acquired from perfect oracles thus they are noise-free. It is non-trivial to quantify and determine which constraints are noisy prior to clustering.

To address the above issues, we formulate a novel COnstraint Propagation Random Forest (COP-RF), not only capable of effectively propagating sparse pairwise constraints, but also able to identify and thus filter noisy constraints produced by imperfect oracles. The COP-RF is flexible in that it generates an affinity matrix that encodes the constraint information for existing spectral clustering methods [10] for the subsequent constrained clustering.

More precisely, the proposed model allows for effective sparse constraint propagation through using the NN samples that are found in discriminative feature subspaces, rather than those that found considering the whole feature space, which can be suboptimal due to noisy and ambiguous features. This is made possible by introducing a new objective/split function into COP-RF, which searches for discriminative features that induce the best data subspaces while simultaneously considering the model parameters that best satisfy the pairwise constraints imposed. To identify

and filter noisy constraints generated from imperfect oracles, we introduce a filtering mechanism to discover consistent constraint subsets that incur *less internal conflict* with one another and *more coherent* with the underlying data distribution. This is achieved through quantifying the information gain induced by individual constraints during tree node splitting in COP-RF. Figure 1 shows an example to illustrate how a COP-RF is capable of discovering data partitions close to the ground truth clusters despite it is provided only with sparse and noisy constraints.

The sparse and noisy constraint issues are inextricably linked but no existing constrained clustering methods, to our knowledge, address them in a unified framework. This is the very first study that proposes a principled data-driven approach to address them jointly. In particular, our work makes the following contributions: (I) We formulate a novel discriminative-feature driven approach for effective sparse constraint propagation. Existing methods fundamentally ignore the role of feature selection in this problem. (II) We propose a data-driven method to filter potentially noisy constraints, a problem that is largely unaddressed by existing constrained clustering algorithms. All these capabilities are achieved using a single unified COF-RF model.

We evaluate the effectiveness of the proposed approach on UCI and video datasets. We demonstrate that the COP-RF is superior when compared to the state-of-the-art constrained clustering algorithms [11], [5], [9] in exploiting sparse constraints. In addition, we show that the proposed model, unlike existing methods, is capable of performing robust clustering even when noisy pairwise constraints are included in the learning process.

II. Effective Constraint Propagation

A. Problem Formulation

Given a set of samples denoted as $X = \{\mathbf{x}_i\}$, $i = 1, \ldots, N$, with N referring to the total number of samples, and $\mathbf{x}_i = (x_{i,1}, \ldots, x_{i,d}) \in \mathcal{F}$, d the feature dimensionality of the feature space $\mathcal{F} \subset \mathbb{R}^d$, the goal of unsupervised clustering is to assign each sample $\mathbf{x}_i$ with a cluster label c_i. In constrained clustering, additional pairwise constraints are available to influence the cluster formation. There are two typical types of pairwise constraints

$$\begin{aligned} \text{Must-link} \quad &: \quad \mathcal{M} = \{(\mathbf{x}_i, \mathbf{x}_j) \mid c_i = c_j\}, \\ \text{Cannot-link} \quad &: \quad \mathcal{C} \ = \{(\mathbf{x}_i, \mathbf{x}_j) \mid c_i \neq c_j\}. \end{aligned} \tag{1}$$

We denote the full constraint set as $\mathcal{P} = \mathcal{M} \cup \mathcal{C}$, and the cardinality of $\mathcal{P}$ as $|\mathcal{P}|$. The pairwise constraints may arise from pairwise similarity as perceived by a human annotator (oracle), temporal continuity, or prior knowledge on the sample class label. Acquiring pairwise constraints from a human annotator is expensive. In addition, owing to data ambiguity and human unintentional mistakes, the pairwise constraints are likely to be incorrect and inconsistent with the underlying data distribution.

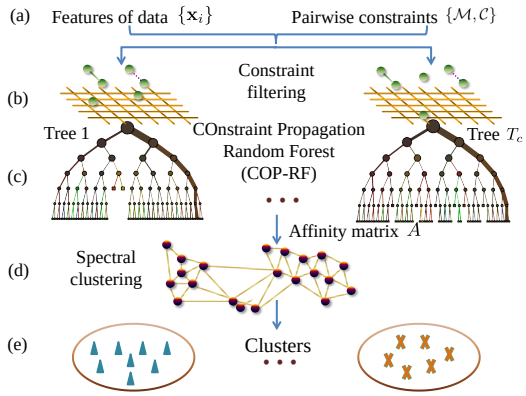


Figure 2. Overview of our approach.

We propose a model that can flexibly generate a constraint-aware affinity matrix, which is directly employed by existing spectral clustering methods as input for constrained clustering (Figure 2). Before detailing our model we briefly describe the conventional random forests.

B. Conventional Random Forests

Classification forests - A general form of random forests is the classification forests. A classification forest [2] is an ensemble of T_{class} binary decision trees $\mathcal{T}(\mathbf{x})$: $\mathcal{F} \rightarrow \mathbb{R}^K$, with $\mathbb{R}^K = [0, 1]^K$ denoting the space of class probability distribution over the label space $\mathcal{L} = \{1, \ldots, K\}$.

Tree training: Decision trees are learned independently from each other, each with a random training set $X^t \subset X$, i.e. bagging [2]. Growing a decision tree involves a recursive node splitting procedure until some stopping criterion is satisfied, e.g. the number of training samples arriving at a node is equal to or smaller than a threshold ϕ, and leaf nodes are then formed, and their class probability distributions are estimated with the labels of the arrival samples as well.

The training of each internal (or split) node is a process of optimising a binary split function defined as

$$\mathrm{h}(\mathbf{x}, \Theta) = \begin{cases} 0 & \text{if } \mathbf{x}_{\vartheta_1} < \vartheta_2, \\ 1 & \text{otherwise.} \end{cases} \tag{2}$$

This split function is parameterised by two parameters: (i) a feature dimension $\vartheta_1 \in \{1, \ldots, d\}$, and (ii) a feature threshold $\vartheta_2 \in \mathbb{R}$. All arrival samples of a split node will be channelled to either the left or right child node, according to the output of Equation (2). The optimal split parameter Θ^* is chosen via

$$\Theta^* = \underset{\{\Theta^i\}_{i=1}^{m_{\text{try}}}}{\operatorname{argmax}} \Delta\mathcal{I}_{\text{class}}, \tag{3}$$

where $\{\Theta^i\}$ represents the parameter space over m_{try} randomly selected features. That is, multiple candidate data splittings are performed on m_{try} random feature-dimensions during the above optimisation process. Typically, a greedy search strategy is exploited to identify Θ^*.

The information gain $\Delta\mathcal{I}_{\text{class}}$ is formulated as

$$\Delta\mathcal{I}_{\text{class}} = \mathcal{I}_s - \frac{|L|}{|S|}\mathcal{I}_l - \frac{|R|}{|S|}\mathcal{I}_r, \tag{4}$$

where s, l, r refer to a split node, the left and right child node, respectively. The sets of data routed into l and r are denoted as L and R, and $S = L \cup R$ as the sample set residing at s. The $\mathcal{I}$ can be computed as either the entropy or Gini impurity [2]. In this study we utilise the Gini impurity due to its simplicity and efficiency.

Clustering forests - In contrast to classification forests, clustering forests [8], [7] require no ground truth information during the training phase. A clustering forest consists of T_{clust} binary decision trees. The leaf nodes in each tree define a spatial partitioning of the training data. Interestingly, the training of a clustering forest can be performed using the classification forest optimisation approach by adopting the pseudo two-class algorithm [2], [7]. Specifically, we add N uniformly distributed pseudo samples $\bar{\mathbf{x}} = \{\bar{x}_1, \ldots, \bar{x}_d\}$, with $\bar{x}_i \sim \mathbf{U}\left(x_i | min\left(x_i\right), max\left(x_i\right)\right)$ into the original data space X. With this strategy, the clustering problem becomes a canonical classification problem that can be solved by the classification forest training method as discussed above.

C. Our Model: Constraint Propagation Random Forest

To address the issues of sparse and noisy constraints, we formulate a novel COnstraint Propagation Random Forest (COP-RF) (see Figure 2). We consider using a random forest, particularly a clustering forest [2], [7] as the basis to derive our new model for two main reasons: (I) It has been shown that random forest has a close connection with adaptive k-nearest neighbour methods, as a forest model adapts neighbourhood shape according to the local importance of different input variables [6]. This motivates us to exploit the adaptive neighbourhood shape[1] for effective constraint propagation. (II) The forest model also offers a way to evaluate information gain of the underlying data distribution. We can build upon it to quantify the consistency between constraints and the data distribution effectively, which could be useful in identifying noisy constraints.

The proposed COP-RF differs significantly from the conventional random forests in that the COP-RF is formulated with a new split function, which considers not only the bottom-up data information gain maximisation, but also the joint satisfaction of top-down pairwise constraints. Next, we discuss the mechanism to achieve effective sparse constraint propagation though discriminative feature subspaces.

Propagation via discriminative feature subspaces - We construct a COP-RF through learning a collection of T_c constraint-aware COP-trees. Similar to the training of an ordinary decision tree, to train a COP-tree we optimise the split function (Equation (2)) by finding Θ^* with both the best feature dimension and cut-point to partition its node training samples S. The difference is that the term 'best' or 'optimal' is no longer defined only as to maximising the bottom-up information gain, but also simultaneously satisfying the imposed top-down pairwise constraints. More precisely, at the t-th COP-tree, its training set X^t only encompasses a subset of the full constraint set $\mathcal{P}$, i.e. $\mathcal{P}^t = \{\mathcal{M}^t \cup \mathcal{C}^t\} \subset \mathcal{P}$. Instead of using the information gain in Equation (4), we optimise each internal node s in a COP-tree via Equation (3) with the information gain $\Delta\mathcal{I}$ defined as follow

$$\begin{aligned} \text{maximise} \quad \Delta\mathcal{I} &= \mathcal{I}_s - \frac{|L|}{|S|}\mathcal{I}_l - \frac{|R|}{|S|}\mathcal{I}_r, \\ \text{s.t.} \ \forall(\mathbf{x}_i, \mathbf{x}_j) &\in \mathcal{M}^t \Rightarrow c_i = c_j \in \{l, r\}, \\ \forall(\mathbf{x}_i, \mathbf{x}_j) &\in \mathcal{C}^t \Rightarrow c_i \neq c_j, \\ \text{where} \ \mathbf{x}_i, \mathbf{x}_j \in S, \ \mathcal{P}^t &= \mathcal{M}^t \cup \mathcal{C}^t. \end{aligned} \tag{5}$$

Equation (5) differs significantly from the conventional information gain function [2], [7] as the maximisation is bounded by the constraint set $\mathcal{P}^t$. More specifically, it automatically selects discriminative features and their optimal cut-point via information-based energy optimisation, whilst at the same time fulfilling the guiding conditions imposed by pairwise constraints. Algorithm 1 summarises the split function optimisation procedure in a COP-tree. Effective constraint propagation occurs when we construct a constraint-aware data affinity matrix for spectral clustering [10], taking into account the discriminative neighbourhoods induced by individual COP-trees.

Combining with spectral clustering - Conventionally, an affinity matrix is constructed by computing pairwise distance with some Euclidean-based measure. It is however observed in some studies that the Euclidean distance is often not an accurate representation of the underlying shape of data [3]. In addition, defining data neighbourhoods via the whole feature space can be susceptible to noisy features.

The learned COP-RF offers an effective way to derive a more meaningful affinity matrix, which not only defines data similarity through discriminative feature subspaces, but also encodes the pairwise constraint information. Note that the t-th COP-tree only considers a subset of constraints $\mathcal{P}^t$ but not the full constraint $\mathcal{P}$. Nevertheless, since a different tree considers a random set of $\mathcal{P}^t$ (due to the random set X^t), a good coverage of all constraints can be achieved by averaging many trees' statistics.

Formally, each individual tree within a COP-RF partitions the training samples at its leaves $\ell(\mathbf{x})$: $\mathbb{R}^d \rightarrow \mathbb{L} \subset \mathbb{N}$, where ℓ represents a leaf index and $\mathbb{L}$ refers to the set of all leaves in a given tree. For each COP-tree, we first compute a tree-level $N \times N$ affinity matrix A^t with elements defined as $A^t_{i,j} = \exp^{-\text{dist}^t(\mathbf{x}_i, \mathbf{x}_j)}$ with

$$\text{dist}^t(\mathbf{x}_i, \mathbf{x}_j) = \begin{cases} 0 & \text{if } \ell(\mathbf{x}_i) = \ell(\mathbf{x}_j), \\ +\infty & \text{otherwise.} \end{cases} \tag{6}$$

[1]The neighbours of a data $\mathbf{x}$ in forest interpretation are the points which fall into the same child node.

We assign the maximum affinity (affinity=1, distance=0) to points $\mathbf{x}_i$ and $\mathbf{x}_j$ if they fall into the same leaf, and the minimum affinity (affinity=0, distance=$+\infty$) otherwise. By averaging all the tree-level affinity matrices we obtain a smooth matrix as $A = \frac{1}{T_c}\sum_{t=1}^{T_c} A^t$. with $A_{i,i} = 0$. We then construct a sparse k-NN graph, whose edge weights are defined by A (Figure 2-c). Since the affinity matrix A is constraint-encoded, using the k-NN graph as input readily transforms the conventional spectral clustering methods [10] for constrained clustering.

Filtering noisy constraints from imperfect oracles - Constraint propagation should be reinforced by noisy constraint filtering to avoid error propagation to neighbouring unlabelled points. To this end, we formulate a novel method to identify noisy constraints through quantifying constraint inconsistency by the information gain criterion. Specifically, an inconsistent constraint is likely to

- Conflict with the majority of other constraints, assuming that most constraints are valid.
- Disagree with the underlying data distribution.

During the tree node splitting, we observe that satisfying constraints that disagree with the underlying data distribution would incur sub-optimal data partition, leading to low data information gain. Motivated by this observation, we exploit the information gain measure to filter possibly noisy constraints from the set $\mathcal{P}^t$. The filtering process does not physically remove the suspected noisy constraints. But it is a process that is conducted at the root node of each COP-tree t, so that only selected constraints $\mathcal{S}^t \subset \mathcal{P}^t$ will be used to perform the root data partition $\{L^0, R^0\}$. This partition would 'set a good starting point' for the subsequent data splittings in tree branches. As compared to physically removing suspected constraints, this scheme is more conservative but empirically gives better results.

Next, we describe the steps to estimate the consistency of a pairwise constraint and subsequently the way to determine $\mathcal{S}^t$. A conflict will only occur when we consider multiple constraints together. Hence, to better quantify the degree to which a constraint conflicts with other randomly selected constraints and the underlying data distribution, we repeat the following steps for f repetitions. For a repetition i

1) Randomly sample a temporary subset of constraints $\mathcal{Q}$ from $\mathcal{P}^t$, $\mathcal{Q} \subset \mathcal{P}^t$, where $|\mathcal{Q}| = \alpha|\mathcal{P}^t|$ and $0 < \alpha < 1$.
2) Compute the information gain $\delta\mathcal{I}$ following Algorithm 1 by using $\mathcal{Q}$ as the constraint set[2]. For any j-th constraint in the set $\mathcal{P}^t$, we assign its induced information gain $\delta\mathcal{I}_j^i$ as

$$\delta\mathcal{I}_j^i = \begin{cases} \delta\mathcal{I} & \text{the } j\text{-th constraint} \in \mathcal{Q}, \\ 0 & \text{otherwise.} \end{cases} \quad (7)$$

Each repetition employs a different random subset sampled from $\mathcal{P}^t$. Now the consistency of the j-th constraint in the set $\mathcal{P}^t$ can be quantified by the corresponding information gain averaged across f repetitions, i.e.

$$\widehat{\delta\mathcal{I}}_j = \frac{1}{r}\sum_{i=1}^{f} \delta\mathcal{I}_j^i, \quad (8)$$

where r is the number of times the j-th constraint is selected within the f repetitions. A noisy constraint would have a low value in $\widehat{\delta\mathcal{I}}$. Consequently, the optimal constraint subset $\mathcal{S}^t$ is selected as the top $\alpha \times |\mathcal{P}^t|$ constraints that achieve the highest values in $\widehat{\delta\mathcal{I}}$. In this study we set $\alpha = 0.5$ and $f = 500$ so that each individual constraint has a fair chance to be paired with other constraints.

[2] $\delta\mathcal{I}$ is computed in a similar way of $\Delta\mathcal{I}$ in Equation 5. We use a different symbol for clarity.

Algorithm 1: Split function optimisation in a COP-tree.

Input: At a split node s of a COP-tree t:
- Training samples available to s: S;
- Pairwise constraints: $\mathcal{P}^t = \mathcal{M}^t \cup \mathcal{C}^t$;

Output:
- The best feature cut-point Θ^* and;
- The associated child node partition $\{L, R\}$;

Optimisation:
Initialise $L = R = \emptyset$ and $\Delta\mathcal{I} = 0$;
for $var \leftarrow 1$ **to** m_{try} **do**
  Select a feature $f_{var} \in \{1, \ldots, d\}$;
  for *each possible cut-point of the feature* f_{var} **do**
    Split S into a candidate partition $\{\hat{L}, \hat{R}\}$;
    $dec = \textbf{respect\_all\_constraints}(\{\hat{L}, \hat{R}\}, \{\mathcal{M}^t, \mathcal{C}^t\})$;
    if dec *is true* **then**
      Compute information gain $\Delta\hat{\mathcal{I}}$ following Equation (5);
      if $\Delta\hat{\mathcal{I}} > \Delta\mathcal{I}$ **then**
        Update Θ^*;
        Update $\Delta\mathcal{I} = \Delta\hat{\mathcal{I}}$, $L = \hat{L}$, and $R = \hat{R}$.
      end
    end
    else
      Ignore the current splitting.
    end
  end
end
if *No valid splitting found* **then**
  A leaf is formed.
end
function $\textbf{respect\_all\_constraints}(\{L, R\}, \{\mathcal{M}, \mathcal{C}\})$
{
  $\forall(\mathbf{x}_i, \mathbf{x}_j) \in \mathcal{M}$,
    if ($\mathbf{x}_i \in L$ and $\mathbf{x}_j \in R$, or vice versa), return false.
  $\forall(\mathbf{x}_i, \mathbf{x}_j) \in \mathcal{C}$,
    if ($\mathbf{x}_i, \mathbf{x}_j \in L$, or $\mathbf{x}_i, \mathbf{x}_j \in R$), return false.
  Otherwise, return true.
}

III. Experimental Settings

Datasets - To evaluate the effectiveness of our method in coping with data of varying numbers of dimensions and clusters, we select five diverse UCI benchmark datasets [1]. We also collect an intrinsically noisy video dataset from a publicly available web-camera deployed in a university's Educational Resource Center (ERCe). This dataset consists of 600 video clips with six possible clusters of events (see Figure 3 for example images). The details of all datasets are summarised in Table I.

Features - For the UCI datasets, we use the original features provided. As for the ERCe video data, we segment a long video into non-overlapping clips, from which a number of

Table I
THE DETAILS OF DATASETS.

Dataset	# Clusters	# Features	# Instances
Ionosphere (Iono.)	2	34	351
Iris	3	4	150
Segmentation (Seg.)	7	19	210
Parkinsons (Park.)	2	22	195
Glass	6	10	214
ERCe	6	2672	600

Figure 3. Example images from the ERCe video dataset. It contains six events including (a) Student Orientation, (b) Cleaning, (c) Career Fair, (d) Group Study, (e) Gun Forum, and (f) Scholarship Competition.

visual features are then extracted, including colour, local texture, optical flow, holistic image features and object detections. We perform PCA on the resulting 2672-D feature vectors of video clips, and use the first 30 PCA components as the final representation. All raw features are scaled to the range of [-1,1].

Baselines - For comparison, we present the results of (1) *Spectral Clustering (SPClust)* [10], which does not exploit any pairwise constraint; (2) *COP-Kmeans* [11], a popular constrained clustering method based on k-means; (3) *Spectral Learning (SL)* [5], a constrained spectral clustering method without constraint propagation. It extends SPClust by trivially adjusting the elements in a data affinity matrix with 1 and 0 to satisfy must-link and cannot-link constraints, respectively; (4) a state-of-the-art constrained spectral clustering approach *E^2CP* [9], in which constraint propagation is achieved by manifold diffusion [12]; and (5) *Forest + E^2CP* – we modify E^{2}CP [9], i.e. instead of generating the data affinity matrix with Euclidean-based measure, we use a conventional clustering forest to generate the affinity matrix. This allows E^{2}CP to enjoy a limited capability of feature selection using a random forest model.

Evaluation metrics - We use the widely adopted adjusted Rand Index (ARI) [4] as the evaluation metric. Throughout all the experiments, we report the ARI values averaged over 10 trials. In each trial we generate a random pairwise constraint set from the ground truth cluster labels.

Implementation details - The number of trees, T_c, in a COP-RF is set to 1000. Each X^t is obtained by performing N times of random selection with replacement (see Section II-B). The depth of each COP-tree is governed by either constraint satisfaction, i.e. a node will stop growing if during any attempted data partition some constraints are violated (see Algorithm 1), or the size of a node equals to 1 (i.e. $\phi = 1$). We set m_{try} (see Equation (3)) to $\sqrt{d}$ with d the feature dimensionality of the input data and employ a linear data separation as the split function (see Equation (2)). We set $k \approx N/10$ for the k-nearest neighbour graph construction.

IV. EVALUATIONS

We conduct comparative experiments to (1) evaluate the effectiveness of different clustering methods in exploiting sparse but perfect pairwise constraints (Section IV-A), and (2) compare their clustering performances in the case of having imperfect oracles to provide ill-conditioned pairwise constraints (Section IV-B).

A. Evaluation of Sparse Constraint Propagation

In this experiment, we assume perfect oracles thus all the pairwise constraints agree with the ground truth cluster labels. Figure 4 reports the ARI curves of different methods along with varying numbers of pairwise constraints from 20 to 100. The overall performance of various methods can be quantified by the area under the ARI curve and the results are reported in Table II. It is evident from the results (Figure 4 and Table II) that on most datasets, the proposed COP-RF outperforms other baselines, by as much as $>$**300**% against COP-Kmeans[3] and $>$**30**% against the state-of-the-art E^{2}CP in averaged area under the ARI curve.

Table II
COMPARING DIFFERENT METHODS BY THE AREA UNDER THE ARI CURVE. PERFECT ORACLES ARE ASSUMED. HIGHER IS BETTER.

Dataset	SPClust	COP-Kmeans	SL	E^{2}CP	Forest + E^{2}CP	COP-RF
Iono.	0.43	0.65	0.23	0.37	**2.48**	2.15
Iris	3.47	0.55	3.53	**3.54**	3.49	3.51
Seg.	1.96	0.36	1.96	1.99	**2.20**	2.19
Park.	0.78	0.21	0.83	1.06	1.35	**1.45**
Glass	1.14	0.62	1.21	1.36	1.67	**2.22**
ERCe	2.76	0.84	2.74	2.40	3.01	**3.06**
Average	1.76	0.54	1.75	1.79	2.37	**2.43**

It is worth pointing out that although the state-of-the-art E^{2}CP performs generally better than other baselines, it is inferior to the proposed COP-RF, since its manifold construction still considers the full feature space, which may be corrupted by noisy features. We observe in some cases, such as the challenging ERCe dataset, the performance of E^{2}CP is worse than the naive SL method that comes without constraint propagation. This result suggests that propagation could be *harmful* when the feature space is noisy. The variant modified by us, i.e. Forest + E^{2}CP, employs a conventional clustering forest ([2], [7]) to generate the data affinity matrix. This allows E^{2}CP to take advantage of a limited capability of forest-based feature selection, and better results are obtained compared with the pure E^{2}CP. Nevertheless, Forest + E^{2}CP's performance is generally poorer than COP-RF's (see Table II). This is because the feature selection of

[3]COP-Kmeans fails to converge (early termination without a solution) on datasets Iris, Segmentation, and Glass.

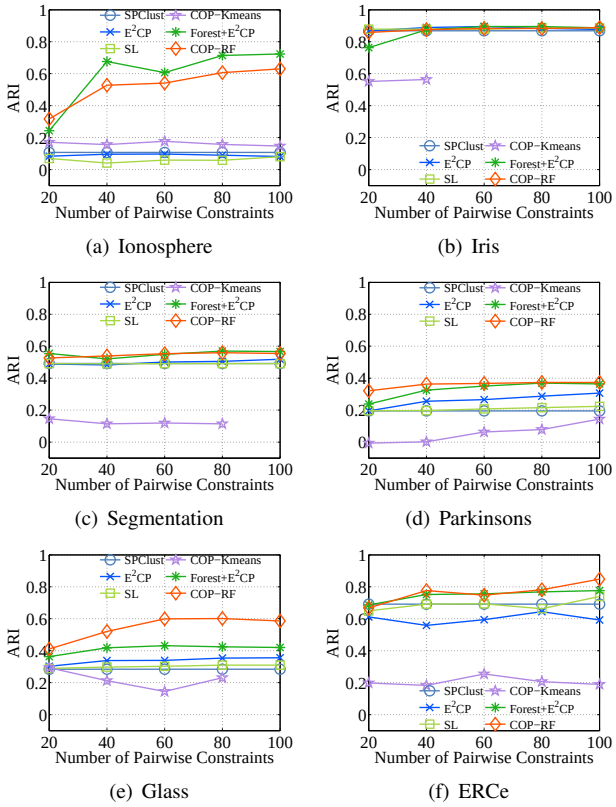


Figure 4. ARI comparison: comparison of clustering performance between different methods given a varying number of perfect pairwise constraints.

the ordinary forest model is less effective than that of COP-RF, which jointly considers information gain maximisation and constraint satisfaction.

B. Evaluation on Filtering Noisy Constraints

In this experiment, we assume imperfect oracles thus pairwise constraints are noisy. We deliberately prepare constraint sets that are mixed with a fixed ratio (15%) of random invalid constraints that disagree with the ground truth. This is to simulate the annotation behaviour of imperfect oracles for the comparison of our approach with existing models. We repeat the same experimental protocol as discussed in Section IV-A. It is observed from Table III that in spite of the imperfect oracle assumption, COP-RF again achieves better results than other constrained clustering models on most datasets as well as the best average clustering performance across datasets, e.g. $>$**300**% increase against COP-Kmeans and $>$**35**% increase against E^{2}CP. Furthermore, Table III also shows that COP-RF maintains encouraging performance given noisy constraints, in some cases such as the challenging ERCe video dataset even larger improvements are obtained over E^{2}CP and other models, than that when perfect constraints are provided. The results suggest the effectiveness of the proposed constraint filtering algorithm in coping with noisy constraints.

Table III
COMPARING DIFFERENT METHODS BY THE AREA UNDER THE ARI CURVE. IMPERFECT ORACLES ARE ASSUMED. HIGHER IS BETTER.

Datasets	SPClust	COP-Kmeans	SL	E^{2}CP	Forest + E^{2}CP	COP-RF
Iono.	0.43	0.59	0.22	0.23	**1.56**	1.38
Iris	3.47	0.52	**3.52**	3.51	3.13	3.39
Seg.	1.96	0.64	1.96	1.97	2.02	**2.11**
Park.	0.78	0.14	0.82	0.94	1.05	**1.11**
Glass	1.14	0.21	1.15	1.31	1.35	**1.68**
ERCe	2.76	0.79	1.21	1.19	2.31	**2.81**
Average	1.76	0.48	1.47	1.52	1.90	**2.08**

V. CONCLUSION

We have presented a unified constrained spectral clustering framework to (1) propagate sparse constraints effectively, and (2) handle noisy constraints generated by imperfect oracles. The proposed COP-RF model is novel in that it propagates constraints more effectively via discriminative feature subspaces. This is in contrast to existing methods that perform propagation considering the whole feature space, which may be corrupted by noisy features. Effective propagation regardless of the constraint quality could lead to poor clustering results. Our work addresses this crucial issue by formulating a way to quantify the inconsistency of constraints and effectively filter potentially noisy ones before propagation takes place. The model is flexible in that it generates a constraint-aware affinity matrix that can be used by the popular spectral clustering methods for constrained clustering. Experimental results on various datasets have demonstrated the advantages of the proposed approach over the state-of-the-art constrained clustering methods.

REFERENCES

[1] A. Asuncion and D. Newman. UCI machine learning repository, 2007.
[2] L. Breiman. Random forests. *ML*, 45(1):5–32, 2001.
[3] A. Frome, Y. Singer, and J. Malik. Image retrieval and classification using local distance functions. In *NIPS*, 2007.
[4] L. Hubert and P. Arabie. Comparing partitions. *Journal of Classification*, 2(1):193–218, 1985.
[5] S. D. Kamvar, D. Klein, and C. D. Manning. Spectral learning. In *IJCAI*, 2003.
[6] Y. Lin and Y. Jeon. Random forests and adaptive nearest neighbors. *Journal of the American Statistical Association*, 101(474):578–590, 2006.
[7] B. Liu, Y. Xia, and P. S. Yu. Clustering through decision tree construction. In *CIKM*, pages 20–29, 2000.
[8] C. Liu, S. Gong, C. C. Loy, and X. Lin. Person re-identification: what features are important? In *International Workshop on Re-Identification, ECCV*, pages 391–401, 2012.
[9] Z. Lu and H. H. Ip. Constrained spectral clustering via exhaustive and efficient constraint propagation. In *ECCV*, pages 1–14, 2010.
[10] A. Y. Ng, M. I. Jordan, Y. Weiss, et al. On spectral clustering: Analysis and an algorithm. *NIPS*, 2:849–856, 2002.
[11] K. Wagstaff, C. Cardie, S. Rogers, and S. Schrödl. Constrained k-means clustering with background knowledge. In *ICML*, pages 577–584, 2001.
[12] D. Zhou, J. Weston, A. Gretton, O. Bousquet, and B. Schölkopf. Ranking on data manifolds. *NIPS*, 16:169–176, 2004.