Joint Optimization of Fidelity and Commensurability for Manifold Alignment and Graph Matching

by

Sancar Adali

A dissertation submitted to The Johns Hopkins University in conformity with the requirements for the degree of Doctor of Philosophy.

Baltimore, Maryland

March, 2014

© Sancar Adali 2014

All rights reserved

Abstract

In this thesis, we investigate how to perform inference in settings in which the data consist of different modalities or views. For effective learning utilizing the information available, data fusion that considers all views of these *multiview* data settings is needed. We also require dimensionality reduction to address the problems associated with high dimensionality, or "the curse of dimensionality." We are interested in the type of information that is available in the multiview data that is essential for the inference task. We also seek to determine the principles to be used throughout the dimensionality reduction and data fusion steps to provide acceptable task performance. Our research focuses on exploring how these queries and their solutions are relevant to particular data problems of interest.

Primary Reader: Carey E Priebe

Secondary Reader: Donniell E Fishkind

Dedication

This thesis is dedicated to myself because I did all the hard work and to my family who supported me in every way, especially my mother, from whom I inherit my love of science.

Contents

CONTENTS

CONTENTS

List of Tables

List of Figures

List of Notations

Chapter 1

Introduction

1.1 Data Settings

It is a challenge to perform a tractable analysis on data obtained from disparate sources (such as multiple sensors). The increasing variety of sensor technologies and the large number of sensors introduce challenges but also hold promise for effective inference. One of our contributions is the development of well-defined simple settings that provide intuition about the right approaches to data fusion and lead to the development of inference methods that are useful in practice.

Our world view of data fusion from multiple sensors is depicted in Figure 1.1. We refer to the entities of interest for pattern recognition as *objects*. These might be real objects or abstract concepts. The data consist of measurements for a collection of these objects.

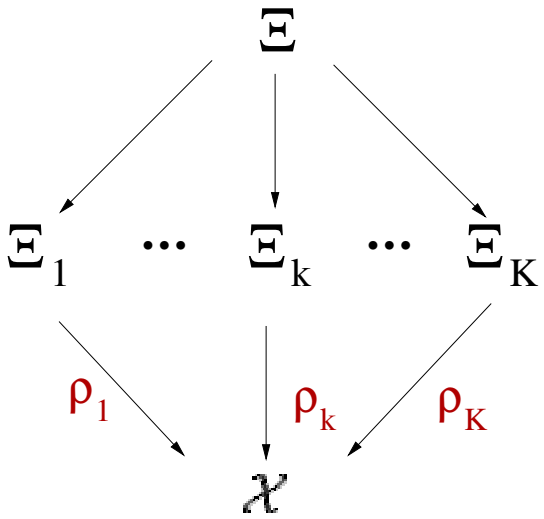


Figure 1.1: Multiple Sensor setting

We assume that these objects lie in some "object" space Ξ and that each sensor has another "view" of the objects. The measurements recorded by the i^{th} sensor lie in some "measurement space" Ξ_i. The usual approach in pattern recognition is to use feature extractors on the spaces for a feature representation in Euclidean space and to use classical pattern recognition tools for the exploitation task. The alternative approach is to acquire the dissimilarities between the group of objects and use them either to find an embedding in a low-dimensional Euclidean space for which classic statistical tools are available for inference or to use dissimilarity-based versions of pattern recognition tools [1]. We use the embedding approach so that we avoid the "curse of dimensionality" with the low embedding dimension, allowing us to still use classic statistical tools. Additionally, the embeddings of dissimilarities from different conditions need to be "commensurate" so that sensor measurements can be compared in a meaningful way (i.e., the degree of

(dis)similarity can be inferred) or jointly used in inference. This is accomplished by maps $\rho_k, k = 1, \ldots, K$ from measurement spaces Ξ_k to a low-dimensional commensurate space $\mathcal{X}$, visualized in Figure 1.1. Learning these maps from data is an important part of our novel approach.

1.1.1 Exploitation Task

Data fusion is a very general concept, and here, we will clarify the specific meaning of data fusion and the setting that we have in mind. The exploitation task in which we are interested might involve (perhaps notional) complex objects or abstract entities that are not practically representable. The objects are members of a (perhaps notional) space called "object" space, Ξ in Figure 1.1. We will extract different "views", "measurements", or "data modalities" from these objects (which we will refer to as "conditions"), and these observations will be elements of the measurement space for those conditions (Ξ_k for k^{th} condition). Each of the objects will have an observation in each of the different conditions, and the corresponding observations across different conditions will be "matched". Given new observations from these different conditions, is it possible to determine whether they are "matched"? If a group of observations from each condition are "matched" to each other but the specific correspondences are unknown, is it possible to find the true correspondences? Different approaches are proposed in this dissertation to address these questions.

1.2 Dissimilarity representation

Significant progress been made in the theory and applications of pattern recognition, particularly in problem settings in which the data are available or assumed to be available as vectors in metric spaces. There are still many problems for which, due to the nature of the setting, one only has access to dissimilarities, proximities, or distances between measurements or a subjective assessment of the similarities of objects. While our approach depends naturally on the representation, the inference task is agnostic about this representation issue. The gap between the two kinds of representation of data can be bridged using various techniques, such as different kinds of embedding methods, and by computing dissimilarities between entities.

[1] is an excellent resource that compiles the research on learning from dissimilarity-based representation. In the introduction, the authors clarify the distinction between statistical and structural (syntactic) pattern recognition, which was discussed previously in [2]. Statistical pattern recognition addresses the analysis of features, which are measured values for object attributes. Syntactic pattern recognition uses a relational view of objects for representation. In both cases, the task of discrimination can rely on distances (however they are defined). Pękalska and Duin suggest that dissimilarity measures are a natural bridge between these types of information, and their applicability to multiple settings motivates our use of dissimilarity representation in information fusion. For feature-based representation, the features are either raw or processed measurements from

sensors that observe the objects, and the representation of each object is a single point in the representation space, each dimension corresponding to a feature. Dissimilarity-based representation relies on a dissimilarity measure, a way of quantifying the dissimilarity, proximity, or similarity between any two objects. Preferably, the dissimilarity is *designed for* the inference task at hand. There are multiple ways of comparing entities (some more natural than others), which is the basis for one of the arguments behind our approach to information fusion from disparate data sources, including separate sources of the same modality. When the data come from separate sensors that are of the same type, the same measurements might have different dissimilarity representations according to subjective judgments or different dissimilarity measures.

1.3 Match Detection

We will now provide a formal description of the problem that was mentioned in subsection 1.1.1, which was the initial motivation for our investigations, along with a few general remarks. We will describe this problem in more detail in chapter 4.

Consider n distinct objects, which are described with a finite number of measurements. Each of the measurements x_{ik} lies in the corresponding space Ξ_k, and the mea-

surements x_{ik} are matched for the same k index.

$$\begin{array}{cccc} & \Xi_1 & \cdots & \Xi_K \\ Object\ 1 & \boldsymbol{x}_{11} & \sim \cdots \sim & \boldsymbol{x}_{1K} \\ \vdots & \vdots & \vdots & \vdots \\ Object\ n & \boldsymbol{x}_{n1} & \sim \cdots \sim & \boldsymbol{x}_{nK} \end{array}$$

To each pair of measurements x_{ik}, x_{jk} in the same space, we can assign a dissimilarity value $\delta_{ijk} = \delta\{x_{ik}, x_{jk}\}$, which is dependent on the space Ξ_k. We assume the dissimilarities are symmetric, are always non-negative and that they are positive and 0 according as the two arguments x_{ik}, x_{jk} are different or the same. We exploit this training set of dissimilarities to perform inference on the following exploitation task:

Given the dissimilarities between K new measurements/observations ($\boldsymbol{y}_k;\ k \in [K]$) and the previous n objects under K conditions, we test the null hypothesis that "these measurements are from the same object" against the alternative hypothesis that "they are not from the same object" [3]:

$$H_0 : \boldsymbol{y}_1 \sim \boldsymbol{y}_2 \sim \cdots \sim \boldsymbol{y}_K \text{ versus } H_A : \exists i, j, 1 \leq i < j \leq K : \boldsymbol{y}_i \nsim \boldsymbol{y}_j$$

The null hypothesis can be restated as the case in which the dissimilarities are *matched*, and the alternative can be restated as the case in which they are *not matched*.

We represent the dissimilarities between n objects in the form of $n \times n$ dissimilarity matrices $\{\Delta_k; k = 1, \ldots, K\}$ where the entries for the k^{th} dissimilarity matrix are $\{\delta_{ij}^{(k)}; i = 1, \ldots, n; \quad j = 1, \ldots, n\}$. For the matching task, we are given K vectors of new dissimilarities $\{\mathcal{D}_k, k = 1, \ldots, K\}$ each of which has the entries

$\{\delta_{i,new}^{(k)}; i = 1, \ldots, n; \quad k = 1, \ldots, K\}$, where $\delta_{i,new}^{(k)}$ is the dissimilarity between x_{ik} and y_k.

For the hypothesis testing problem, we are to compute the test statistics for the objects represented by the given dissimilarities. In order to compute the test statistics, it is necessary to obtain a collection of mappings (one from each condition) to a lower-dimensional space such that new observations from each condition are made commensurate when they are mapped to this space. These mappings do not need to be explicitly defined; they can be the results of the embedding operation for a particular dataset. If the embedding of the in-sample dissimilarities ($\{\Delta_k; k = 1, \ldots, K\}$) results in a unique mapping, out-of-sample (OOS) embeddings could be adjoined to the embedding of in-sample dissimilarities.

A few points should be mentioned to distinguish our approach from related approaches and emphasize the specific challenges of the inference task.

Remark Because the data sources are "disparate", it is not immediately obvious how a dissimilarity between an object in one condition and another object in another condition can be computed or even meaningfully defined. In general, these between-condition, between-object similarities are not available.

Remark Whether the data are collected in dissimilarity representations for each condition or whether dissimilarities are computed for the observations that are feature observations at each condition is not relevant to our exploitation task. We assume that dissimilarities for each condition are made available for inference purposes (perhaps by

experts in the problem domain).

Remark The exploitation task under consideration is *not* an accurate reconstruction of these feature observations, even if it does exist. If the embeddings are considered good enough to be useful for the inference task, the quality of the embeddings are considered acceptable. Therefore, the quality of our representation will be dependent on the bias-variance tradeoff, where, by choosing a low-dimensional representation, we might be introducing more model bias, but the representation will be more robust with respect to noise, which might result in smaller errors in the inference task.

We will use this inference problem to elucidate two concepts that we introduce in chapter 5. Our novel solution to this matching problem will use those concepts as two error criteria to be minimized. We seek the mappings from each condition to the common low-dimensional space that minimize these error criteria and are most appropriate for the inference task.

Chapter 2

Related Work

2.1 Multiple View Learning

When data are collected using a multitude of sensors or under significantly different environmental conditions, we refer to the data setting as a multiple view setting, in which each "view" provides possibly complementary information about the observed objects[1]. Multiple view learning seeks to exploit these views simultaneously to be more successful in the learning task.

In data settings for multiview learning, the data are observations from $K \geq 2$ views, where both the relationship between the features from different views and the relationship between the features and the quantity to be predicted are unknown. The objective

[1]We use the term "object" loosely because the observed objects could be topics or concepts and the collected data could be text documents about those topics or images that are related to a concept, for example.

is to train the best predictor. It is possible to use all of the features in different views (i.e., concatenate the observation vector from each view) and perform feature selection without considering from which view a feature is obtained. However, this ignores the fact that the modalities can be quite diverse and that combining features from different modalities is not always meaningful. Consider features extracted from an image and an audio segment as features from different modalities. A classifier that treats these features in the same way without considering their modalities is unlikely to perform well. It is more reasonable to use the prior information that the features in the same modality are much more likely to be correlated or commensurate with each other than features in different views and use predictors more suited to each modality if the different modalities are diverse.

Multiple View Learning is a burgeoning field, and there are many cases where one has to leverage many different related datasets for an inference task. For example, for learning tasks related to webpages (such as webpage categorization and ranking of relevant webpages), both the content of the webpages and the hyperlink structure between the webpages can be used.

For social networks, people have different relationships with other people in their networks; networks may be based on similarity of interests, geographical proximity and job relationships, among factors. Combining information from different social networks would provide a more complete perspective of the underlying social life of the people in the network, and one would expect a better performance for all kinds of inference based

on the complete social network data compared with a social network based on a single type of relationship (assuming one does not fall into the trap of overfitting due to having more features in the complete social network data).

In addition, when it is necessary to collect more data, it is often easier to collect data in different modalities than it is to collect more samples in a single modality. For example, in medical studies, it is much easier to collect medical data from already recruited patients compared with recruiting new patients. Data from different modalities might provide complementary information and could result in much more effective predictors, as opposed to data from a single modality that provides diminishing returns with increasing sample size.

Some of the well-studied subfields of machine learning, such as dimensionality reduction, are also relevant to our multiple view setting. As more data are collected, a low-dimensional representation of the data is necessary to be learned to avoid the curse of dimensionality. An interesting question is how dimensionality reduction can be performed in a multiple view setting: is it better to perform dimensionality reduction separately for each modality and concatenate the resulting low-dimensional representations or to find a joint low-dimensional representation for all of the modalities simultaneously? This is a question that we attempt to answer for the data settings we discuss in this thesis.

In the case of missing data, observations of features in the same view could be missing altogether. In the case of such structurally missing data, it makes sense to train an ensemble of predictors that use features from different views independently, so that accurate

predictions can be made even if observations from some of the views are missing.

In [4], the authors discuss an example of multiview learning problems: classification of a multi-lingual document corpus. They co-train classifiers for single-language data that jointly minimize the loss in each single language along with the disagreement between classifiers on training examples. Their findings support the intuition that classifiers based on multiview learning perform better than classifiers trained with data from only a single view.

In [5], the inference task is classification. Features from multiple modalities are fused via canonical correlational analysis, a classical statistical method which computes maximally correlated projections of data. This fusion leads to better classification performance compared with the original set of features in a typical classification problem.

A popular approach to multiview learning is multiple kernel learning, which is the task of learning a kernel matrix for each modality and combining these kernels in an optimal way (with respect to the inference task). For K views, let the i^{th} datum for the k^{th} view be represented as $X_{ik}, \quad i \in \{1, \dots n\}, k \in \{1, \dots, K\}$. For the data in the same k^{th} view, let $\mathcal{K}_k$ be kernel matrix defined for that view, whose $(i,j)^{th}$ entry is $\kappa_k(X_{ik}, X_{jk})$ where $i, j \in 1, \dots, n$. Because any convex combination of the kernels, [2] is also a kernel, it is possible to compute a joint kernel κ that uses all of the multiview data by a convex combination of the kernels in each view. Assuming that a kernel can be defined for each view, the learning problem is to find the optimal (for the inference

[2] A convex combination of the kernels is $\sum_{k'=1}^{K} \alpha_{k'} \kappa_{k'}$ such that $\sum_{k'=1}^{K} \alpha_{k'} = 1$ and $\alpha_{k'} \geq 0, k' \in \{1, \dots, K\}$.

task) set of coefficients $\{\alpha_k\}$. These parameters are usually estimated using training data. Denoting the optimal $\{\alpha_k\}$ by $\{\hat{\alpha}_k\}$, $\hat{\mathcal{K}} = \sum_k \hat{\alpha}_k \mathcal{K}_k$ is the optimal kernel whose $(i,j)^{th}$ entry is $\sum_k \hat{\alpha}_k \kappa_k(X_{ik}, X_{jk})$. Given a new datum $x = [x_1 \dots x_k]$ which consist of K views, the kernel function for each view, κ_k, along with $\{\hat{\alpha}_k\}$ is used to compute the inner product for the joint kernel:

$$\kappa(x, .) = \sum_i \sum_k \hat{\alpha}_k \kappa_k(x_k, X_{ik}).$$

There are many papers on "Multiple Kernel Learning" in the literature [6–8], which are reviewed in a comprehensive survey [9]. Choi et al. [10] use the Markov random walk interpretation of multiple kernel matrices to find a single kernel matrix that depends on the joint probability of the random walks in different views. [11] is another work that uses the random walk interpretation to deal with multiview data. The learning task in [11] is spectral clustering with multiple graphs.

2.2 Transfer Learning

Methods that utilize training data in one domain as auxiliary information for learning in another domain are categorized as "transfer learning" [12]. Sometimes, the source domain and the target domain are actually the same, but the distribution of the data is different, due to the inherent differences between the way in which the training and test data were collected. We call this phenomenon sample selection bias or covariance shift (SSB/CS) [12, 13].

According to [14], this SSB/CS phenomenon is commonly seen in real-life data analysis problems and is usually understated by practitioners. To evaluate novel classifiers, the classifiers are trained on a portion of the available data and tested on the held-out data. Therefore, in the evaluations of classifiers, the assumption that training and test data come from the same distribution is usually valid. However, any performance improvements that a new classifier model has over the baseline would be overwhelmed by the sample selection bias. Thus, one should be skeptical about improving accuracy scores for benchmark datasets in machine learning and treating them as evidence of progress.

We now clarify the differences between transfer learning and SSB/CS problems. Let y denote the random variable for the class label for classification or the dependent variable for regression and X denote the random variates that we use for the learning task. We use the common assumption that the data are *iid*. Suppose we have two domains $\mathcal{D}_s$ and $\mathcal{D}_t$ from which the training data and test data, respectively, are collected. These are called the source and target domains, respectively. The training data $(X_i, y_i) \in \mathcal{D}_s$ and are drawn from the joint distribution $\mathbb{P}(X, y)$. The test data $(X'_i, y'_i) \in \mathcal{D}_t$ and are drawn from the joint distribution $\mathbb{P}'(X, y)$. The most common objective is to infer $\mathbb{P}'(X, y)$ given an *iid* sample of $(X_i, y_i) \in \mathcal{D}_s$. The learning task is usually to minimize $E[\ell(y, \arg\max_y \hat{\mathbb{P}}'(y|X))]$, with respect to $\hat{\mathbb{P}}'(y|X)$ where $\ell(\cdot, \cdot)$ is the loss function chosen for the task, $\hat{\mathbb{P}}'(y|X)$ is an approximation to $\mathbb{P}'(y|X)$ based on the training and the test data. Basically, we require an inference method for the data distribution of the target domain $\hat{\mathbb{P}}'(y|X)$ that minimizes the expected loss for prediction in the target domain.

In the classical supervised learning setting, the source and target domains are the same, and $\mathbb{P}(X, y)$ is assumed to be the same as $\mathbb{P}'(X, y)$. In the *covariate shift* problem setting, the target domain is the same as the source domain, $\mathcal{D}_s = \mathcal{D}_t = \mathcal{D}$, and $\mathbb{P}(y|X) \approx \mathbb{P}'(y|X)$, whereas $\mathbb{P}(X) \neq \mathbb{P}'(X)$. When we cannot make either of the assumptions $\mathbb{P}(X) = \mathbb{P}'(X)$ or $\mathbb{P}(y|X) = \mathbb{P}'(y|X)$, we have the *sample selection bias* problem [13].

In some learning problems, the source and target domains are different $\mathcal{D}_s \neq \mathcal{D}_t$, and all or a considerable portion of the labels $\{y'_i\}$ in $(X'_i, y'_i) \in \mathcal{D}_t$ are missing. In this case, domain adaptation methods allow for the exploitation of both the data in the source domain $\{(X_i, y_i)\}$ and the data in the target domain $\{(X'_i, y'_i)\}$ (where some y'_i might be missing) to construct a good predictor for the target domain [15–18].

Various "domain adaptation" approaches [18, 19] assume the existence of mappings to a common latent space $\mathcal{D}_{com}$, $\Psi_s : \mathcal{D}_s \to \mathcal{D}_{com}$ and $\Psi_t : \mathcal{D}_t \to \mathcal{D}_{com}$ such that the class conditional distributions $\mathbb{P}(\Psi_s(X)|y) \approx \mathbb{P}(\Psi_t(X')|y' = y)$. If these mappings to the commensurate space can be inferred, then they can be used to predict y' given $\Psi_t(X')$, even if no $(X'_i, y'_i) \in \mathcal{D}_t$ pairs exist. In [18], for example, the distance between the conditional distributions $\mathbb{P}(\Psi_s(X)|y)$ $\mathbb{P}(\Psi_t(X')|y' = y)$ is computed using the Maximum Mean Discrepancy measure, and the mappings Ψ_s and Ψ_t are inferred using the minimization of the MMD measure.

2.3 Manifold Alignment

Many efforts have been made toward solving "manifold alignment", which is a problem related to both our data fusion problem and the transfer learning problem (section 2.2). "Manifold alignment" seeks to find correspondences between observations from different "conditions". The setting that is most similar to ours is the semi-supervised setting, in which a set of correspondences are given and the task is to find correspondences between a new set of points in each condition. In contrast, our hypothesis testing task is to determine whether any given pair of points is "matched". The proposed solutions follow a common approach in that they look for a common commensurate space such that the representations (possibly projections or embeddings) of the observations in the commensurate space match.

Note the similarity of the description of "manifold alignment" to the latent space approach for domain adaptation. For both domain adaptation and manifold alignment, the objective is to find mappings to a common space so that the data in one domain can be used for inference in the other domain.

Wang and Mahedavan [20] suggest an approach that uses embedding followed by Procrustes Analysis to find a map to a commensurate space. Given a paired set of points, Procrustes Analysis [21] finds a transformation from one set of points to another in the same space that minimizes the sum of squared distances, subject to some constraints on the transformation (see chapter 7). In the problem mentioned in [20], the paired

set of points correspond to low-dimensional embeddings of kernel matrices. For the embedding step, Laplacian Eigenmaps were used, though their algorithm allows for any appropriate embedding method.

Zhai et al. [22] find two projection matrices to minimize three terms in an energy function similar to our Joint Optimization of Fidelity and Commensurability (JOFC) approach (see chapter 5). One of the terms is the *correspondence-preserving term*, which is the sum of the squared distances between corresponding points and is analogous to our commensurability error term. The other two terms are *manifold regularization terms* and consist of the reconstruction error for a Locally Linear Embedding of the projected points. These terms, which are analogous to fidelity terms, ensure that the projections in the lower dimension retain the structure of the original points by preserving the local neighborhood of points. For fidelity error terms in our setting, the preservation of the structure is accomplished by preserving the dissimilarities. Ham and Lee [23] solve the problem in a semi-supervised setting using a similar approach: minimizing a cost function of three terms, with two terms for fidelity of embedding and one term for commensurability.

In a paper by Baumgartner et al. [24], the joint embedding of kernel matrices is formulated as the optimization of a single objective function that combines Fidelity and Commensurability terms. They use Local Linear Embedding Method for the joint embedding and introduce a tradeoff parameter between *inter-dataset* and *intra-dataset error* (corresponding to commensurability and fidelity, respectively) into the objective func-

tion. This approach could be used as another tool for the investigation of the tradeoff between Fidelity and Commensurability .

Three-way Multidimensional Scaling [25, 26] assumes that the different "conditions" of the data are linear transformations of a single configuration and aims to find this single configuration and the linear transformation. In this approach, the mappings $\{\rho_k\}$ that we define in Figure 1.1 and Figure 1.1 are assumed to be embeddings followed by linear transformations (see also subsection 3.2.7).

Chapter 3

Variants of Multidimensional Scaling and Principal Components Analysis

3.1 Multidimensional Scaling

Multidimensional Scaling (MDS) [1, 26, 27] is the general term that is used to describe methods to embed dissimilarities as points in a Euclidean space. The embeddings are a configuration of points in the Euclidean space with a chosen dimension d such that the distances between the embeddings are as close as possible (in various senses) to the respective original dissimilarities. Different criterion functions can be used to measure how close the distances are to the given dissimilarities, thereby leading to different embedded configurations. These different variants of MDS can be described using a single formulation, which are introduced in section 3.2.

Consider a set of n objects. Let us denote the set $\{1, \ldots, n\}$ by $[n]$. For each pair of objects with indices $i, j \in [n]$, the dissimilarity value, denoted by δ_{ij}, is a nonnegative real number that quantifies how dissimilar those two objects are. The collection of these values form the matrix Δ, which is an nxn dissimilarity matrix.

The dissimilarities have to satisfy $\delta_{ij} \geq 0$, $\delta_{ij} = 0$ if and only if $i = j$, and $\delta_{ij} = \delta_{ji}, \quad \forall i, j \in [n]$. Therefore, Δ is nonnegative, hollow, symmetric and its only zero entries appear on the diagonal. If, in addition, each triplet of dissimilarities δ_{ij}, δ_{ik} and δ_{ik}, $i, j, k \in [n]$ satisfies the triangle inequality, then Δ is called a distance matrix.

3.2 Different criteria for MDS

Multidimensional Scaling methods find a configuration of points $\{\boldsymbol{x_i}; i \in [n]\}$ in a finite-dimensional Euclidean space, whose interpoint distances approximate the given dissimilarities $\{\delta_{ij}; i, j \in [n]\}$. There are various variants of MDS that use different measures of error for this approximation. In general, the criteria minimize the discrepancy between $f(\delta_{ij})$ and $d(\boldsymbol{x_i}, \boldsymbol{x_j})$ $\forall i, j \in [n]$, with respect to $\{\boldsymbol{x_i}\}$ where $d(\cdot, \cdot)$ is the Euclidean distance function and $f(\cdot)$ is a monotonically increasing function that depends on the MDS variant. Depending on whether the MDS variant is "metric" or "non-metric", $f(\cdot)$ is either a linear or nonlinear transformation. Specific variants of MDS are defined by $f(\cdot)$ and the measure of discrepancy between $f(\cdot)$ and $d(\cdot, \cdot)$. We call the latter *the criterion function*, which is optimized with respect to the embedding coordinates.

We represent these coordinates in a $ntimesd$ configuration matrix $\mathbb{X} \in \mathbb{M}_{ntimesd}$, whose i^{th} row is $\boldsymbol{x_i}$. We also represent the $n \times n$ distance matrix whose entries are interpoint distances between the rows of $\mathbb{X}$ with the matrix-valued function, $\mathcal{D}(\mathbb{X})$ and the $(i,j)^{th}$ entry of the distance matrix (the distance between i^{th} and j^{th} rows of $\mathbb{X}$) with $\mathcal{D}_{ij}(\mathbb{X})$.

3.2.1 Metric MDS

For metric MDS, transformations of the form $f(z) = az + b$ are allowed where $a > 0$ and b are scalars.

3.2.1.1 Stress Criterion

Setting $f(z) = z$ and choosing the discrepancy measure between the dissimilarities and distances of embedded points to be ℓ_2, the criterion of the resulting MDS variant is called the raw stress criterion. Additionally, weights ($\{w_{st}, \quad s,t \in [n]\}$) can be introduced for each discrepancy term. Denoting the matrix composed of the weights by W, and the configuration matrix $\mathbb{X}$ that represent the embedded points, we write

$$\sigma_W(\mathbb{X}) = \sum_{s,t \in [n]} w_{st}(\mathcal{D}_{st}(\mathbb{X}) - \delta_{st})^2 \tag{3.1}$$

Subtypes of the Stress criterion are identified by different choices for $\{w_{st}, s,t \in [n]\}$ that depend on the original dissimilarities δ_{st}. For example, choosing all w_{st} to be $\left[\sum_{k,l \in [n]} \delta_{kl}^2\right]^{-1}$, $\forall s,t \in [n]$ normalizes the stress so that the stress value is always between 0 and 1. One can compare different configurations by this standardized stress

value and determine whether a configuration is a good fit based on this value.

Another related criterion is the S-Stress criterion, which involves squares of dissimilarities and distances:

$$\sigma_{SSTRESS}(\mathbb{X}) = \sum_{s,t\in[n]} \left(\mathcal{D}_{st}^2(\mathbb{X}) - \delta_{st}^2\right)^2.$$

3.2.1.2 Sammon Mapping Criterion

This is a specific case of the Stress criterion in which the weights $\{w_{st},\ s,t \in [n]\}$ are set to be $\delta_{st}^{-1}\left[\sum_{k<l}\delta_{kl}\right]^{-1}$. These weights normalize the squared discrepancies in the stress criterion by the magnitude of the original dissimilarities so that the discrepancy terms for the larger dissimilarities do not dominate the optimization of the criterion function. As a result, small δ_{st} are preserved just as well as large δ_{st}.

3.2.2 Ordinal (Nonmetric) MDS

For Nonmetric MDS, $f(\cdot)$ is allowed to be any monotonic transformation. Specifically, in psychometric applications of MDS, the assumption that the dissimilarities are a scaled-shifted version of the "true" dissimilarity is an unwarranted assumption. Even, the existence of a "true" dissimilarity is questionable. Even if the dissimilarities are physical distances, humans tend to have biased estimates of those distances [28]. (i.e. long distances are usually underestimated.) The Nonmetric variant of MDS is also called "ordinal", because what is preserved is the rank of dissimilarities, not their magnitude.

3.2.3 Classical MDS and the Strain Criterion

An $n \times n$ matrix $\Delta = [\delta_{st}]$ is a distance matrix iff

- $\delta_{st} = \delta_{ts}$, $\forall s, t \in [n]$,
- $\delta_{ss} = 0$, $\forall s \in [n]$,
- $\delta_{st} > 0$, $\forall s, t \in [n], s \neq t$ and
- if it obeys the triangle inequality $\delta_{sr} + \delta_{rt} \geq \delta_{st}$ for any triple $s, r, t \in [n]$.

Δ is Euclidean if there exists a configuration of points $\boldsymbol{x}_i \in \mathbb{R}^d$ such that for any pair $s, t \in [n]$, $\delta_{st} = d(\boldsymbol{x}_s, \boldsymbol{x}_t)$.

Consider the case in which Δ is Euclidean. Note that if $\{\boldsymbol{x}_i\}$, $i \in [n]$ satisfy $\delta_{st} = d(\boldsymbol{x}_s, \boldsymbol{x}_t)$ for any pair (s, t), then, for any constant vector $\boldsymbol{u}$ and any rotation/reflection matrix R, the same group of points transformed using R and $\boldsymbol{u}$, i.e. $\{R\boldsymbol{x}_i + \boldsymbol{u}\}$, also satisfy the same distance constraints. To remove the translational ambiguity, we set $\sum_{i=1}^{n} \boldsymbol{x}_i$ to $\mathbf{0}$. How can we recover the original configuration $\{\boldsymbol{x}_i, i \in [n]\}$ from Δ (perhaps up to rotation/reflection)?

The relation between the entries of Δ and $\{\boldsymbol{x}_i\}$ can be written as

$$\delta_{st}^2 = d(\boldsymbol{x}_s, \boldsymbol{x}_t)^2 = \|\boldsymbol{x}_s\|^2 + \|\boldsymbol{x}_t\|^2 - 2\boldsymbol{x}_s \cdot \boldsymbol{x}_t \tag{3.2}$$

for $s, t \in [n]$. Summing (3.2) over s, over t, and then over s and t, we obtain the following identities

$$\sum_{s'} \delta^2_{s't} = \sum_{s'} \|\boldsymbol{x}_{s'}\|^2 + n\|\boldsymbol{x}_t\|^2 - 2\sum_{s'} \boldsymbol{x}_{s'} \cdot \boldsymbol{x}_t \quad t \in [n] \tag{3.3}$$

$$\sum_{t'} \delta^2_{st'} = \sum_{t'} \|\boldsymbol{x}_{t'}\|^2 + n\|\boldsymbol{x}_s\|^2 - 2\sum_{t'} \boldsymbol{x}_s \cdot \boldsymbol{x}_{t'} \quad s \in [n] \tag{3.4}$$

$$\sum_{s',t'} \delta^2_{s't'} = 2n\sum_{t'} \|\boldsymbol{x}_{t'}\|^2 - 2\sum_{s'} \boldsymbol{x}_{s'} \cdot \sum_{t'} \boldsymbol{x}_{t'} \tag{3.5}$$

.

Dividing each equality by $\frac{1}{n}$, $\frac{1}{n}$ and $\frac{1}{n^2}$, respectively, and using the fact that $\sum_{i=1}^n \boldsymbol{x}_i = \mathbf{0}$,

$$\frac{1}{n}\sum_{s'} \delta^2_{s't} = \frac{1}{n}\sum_{s'} \|\boldsymbol{x}_{s'}\|^2 + \|\boldsymbol{x}_t\|^2 \quad t \in [n] \tag{3.6}$$

$$\frac{1}{n}\sum_{t'} \delta^2_{st'} = \frac{1}{n}\sum_{t'} \|\boldsymbol{x}_{t'}\|^2 + \|\boldsymbol{x}_s\|^2 \quad s \in [n] \tag{3.7}$$

$$\frac{1}{n^2}\sum_{s',t'} \delta^2_{s't'} = \frac{2}{n}\sum_{t'} \|\boldsymbol{x}_{t'}\|^2. \tag{3.8}$$

Reorganizing terms, we obtain

$$\|\boldsymbol{x}_t\|^2 = \frac{1}{n}\sum_{s'} \delta^2_{s't} + \frac{1}{n}\sum_{s'} \|\boldsymbol{x}_{s'}\|^2 \quad t \in [n] \tag{3.9}$$

$$\|\boldsymbol{x}_s\|^2 = \frac{1}{n}\sum_{t'} \delta^2_{st'} + \frac{1}{n}\sum_{t'} \|\boldsymbol{x}_{t'}\|^2 \quad s \in [n] \tag{3.10}$$

$$0 = -\frac{2}{n}\sum_{t'} \|\boldsymbol{x}_{t'}\|^2 + \frac{1}{n^2}\sum_{s',t'} \delta^2_{s't'}. \tag{3.11}$$

Summing the three equations, (3.9), (3.10), and (3.11) and replacing $\|\boldsymbol{x}_s\|^2 + \|\boldsymbol{x}_t\|^2$ in the original equation (3.2) with this sum, we obtain

$$\delta_{st}^2 = \frac{1}{n}\sum_{s'}\delta_{s't}^2 + \frac{1}{n}\sum_{s'}\|\boldsymbol{x}_s\|^2 + \frac{1}{n}\sum_{t'}\delta_{st'}^2 + \frac{1}{n}\sum_{t'}\|\boldsymbol{x}_{t'}\|^2 - \frac{2}{n}\sum_{t'}\|\boldsymbol{x}_{t'}\|^2 + \frac{1}{n^2}\sum_{s',t'}\delta_{s't'}^2 - 2\boldsymbol{x}_s\boldsymbol{x}_t$$

for all $s, t \in [n]$.

This expression is simplified to

$$\delta_{st}^2 = \frac{1}{n}\sum_{s'}\delta_{s't}^2 + \frac{1}{n}\sum_{t'}\delta_{st'}^2 + \frac{1}{n^2}\sum_{s',t'}\delta_{s't'}^2 - 2\boldsymbol{x}_s\boldsymbol{x}_t.$$

for all $s, t \in [n]$.

Rearranging terms, we obtain the dot product of $\boldsymbol{x}_s$ and $\boldsymbol{x}_t$:

$$\boldsymbol{x}_s\boldsymbol{x}_t = \frac{-1}{2}\{\delta_{st}^2 - \frac{1}{n}\sum_{s'}\delta_{s't}^2 - \frac{1}{n}\sum_{t'}\delta_{st'}^2 + \frac{1}{n^2}\sum_{s',t'}\delta_{s't'}^2\}.$$

Some of the sums in the above expression can be written in matrix notation as follows:

$$\frac{1}{n}\mathbf{1}^T\Delta^2 = \frac{1}{n}\sum_{s'}\delta_{s't}^2$$

$$\frac{1}{n}\Delta^2\mathbf{1} = \frac{1}{n}\sum_{t'}\delta_{st'}^2$$

where Δ^2 is the $n \times n$ matrix whose entries are the squares of the respective entries of Δ.

Using the above expressions and placing $\{\boldsymbol{x_i}\}$ row-wise into an $n \times d$ matrix $\mathbb{X}$, we can write all of the terms in matrix notation:

$$\mathbb{X}\mathbb{X}^T = \frac{-1}{2}\{\Delta^2 - \frac{1}{n}\mathbf{1}\mathbf{1}^T\Delta^2 - \frac{1}{n}\Delta^2\mathbf{1}\mathbf{1}^T + \frac{1}{n^2}\mathbf{1}\mathbf{1}^T\Delta^2\mathbf{1}\mathbf{1}^T\}.$$

The final expression is

$$\mathbb{X}\mathbb{X}^T = \frac{-1}{2}\{(I_n - \frac{1}{n}\mathbf{1}\mathbf{1}^T)\Delta^2(I_n - \frac{1}{n}\mathbf{1}\mathbf{1}^T\}.$$

Therefore, the configuration matrix $\mathbb{X}$ can be recovered using a eigenvalue decomposition of $Z = \frac{-1}{2}(H\Delta^2 H)$, where $H = (I_n - \frac{1}{n}\mathbf{1}\mathbf{1}^T)$. If the eigenvalue decomposition of Z is $Z = UDU^T$, the solution for $\mathbb{X}$ is $\hat{\mathbb{X}} = UD^{\frac{1}{2}}$ where $D^{\frac{1}{2}}$ is the entrywise square-root of D. Also, $\hat{\mathbb{X}} = \mathbb{X}R$ for some rotation matrix R, i.e. the solution has a rotation ambiguity. Note that because Δ is Euclidean, all diagonal elements of D are nonnegative, and the entries of $D^{\frac{1}{2}}$ are real numbers.

Here it is useful to make the following definition:

Definition 1. *A $n \times n$ matrix Δ is d-Euclidean Distance Matrix (d-EDM) iff it is Euclidean for embedding dimension d, but not $d-1$.*

For dimensionality reduction, we require a lower-dimensional configuration in $\mathbb{R}^{d'}$, where $d' < d$ whose interpoint distances approximate Δ (a d-EDM) . For classical MDS, we seek the configuration $\mathbb{X}_{d'}$ that minimizes $\|\mathbb{X}\mathbb{X}^T - \mathbb{X}_{d'}\mathbb{X}_{d'}^T\|_F^2$. This criterion function is called the "strain" criterion. The minimizer of the strain is found by using the d' largest diagonal elements of D (which are the eigenvalues of Z) as the diagonal elements of $D_{d'}$ and the corresponding eigenvectors as the columns of $U_{d'}$. These matrices yield an $n \times d'$ configuration matrix, $\hat{\mathbb{X}}_{d'} = U_{d'}D_{d'}^{\frac{1}{2}}$.

If Δ is not Euclidean, Z is not positive semidefinite and has negative eigenvalues. In this case, these eigenvalues would be replaced by zeros. We would then proceed with choosing d' largest eigenvalues of Z.

Note also that the classical MDS solution is nested, i.e., if the $n \times d'$ matrix, $\mathbb{X}_{d'}$, is the cMDS solution of the d' dimensional configurations, the first $d'-1$ columns of $\mathbb{X}_{d'}$

comprise the solution for $d' - 1$ dimensional configurations (assuming that the diagonal entries of $D_{d'}$ are sorted in descending order.)

3.2.4 Relationship with other embedding methods

Note that Tang et al. [29] note another connection between embedding methods by showing that the spectral embedding for an unnormalized Laplacian matrix, L (subject to an appropriate scaling of dimensions), is equivalent to the classical MDS solution with the inner product matrix $Z = L^{\dagger}$, where $L^{\dagger}$ is the pseudo-inverse of L [29]. Therefore, for any d-dimensional spectral embedding of the Laplacian L with Laplacian Eigenmaps, there exists an omnibus dissimilarity matrix M, the (d-dimensional) cMDS embedding of which would give the same configuration.

3.2.5 Effect of Perturbations

To determine how robust the embeddings are to error in dissimilarity measurements, perturbation analysis is necessary. Two papers by Sibson [30] investigate how small changes in the dissimilarity matrix change the configuration matrix obtained by classical MDS embedding. The main result in [30] says the following:

"Let $\mathcal{E} = \Delta^2$ for a Euclidean distance matrix Δ and $B = -\frac{1}{2}H\mathcal{E}H$. Let λ be a simple eigenvalue of B with unit-length eigenvector e. Let F be a symmetric matrix whose diagonal entries are zeros. Let $\widetilde{\mathcal{E}}(\epsilon) = \mathcal{E} + \epsilon F + \mathcal{O}(\epsilon^2)$ be the perturbed version

of $\mathcal{E}$. Then, the eigenvalue and eigenvector of $\widetilde{\mathcal{E}}(\epsilon)$ (the perturbed versions of λ and e) are $\widetilde{\lambda}(\epsilon) = \lambda + \epsilon\upsilon + \mathcal{O}(\epsilon^2)$, where $\upsilon = (-\frac{1}{2}e^T Fe)$ and $\widetilde{e}(\epsilon) = e + \epsilon(\frac{1}{2}(B - \lambda I)^\dagger Fe + \frac{1}{2}(\lambda n)^{-1}(\mathbf{1}_N^T Fe)\mathbf{1}_N) + \mathcal{O}(\epsilon^2)$." Because υ, the first-order perturbation of λ, is linear with respect to F, we can conclude

$$E[\upsilon] = -\frac{1}{2}e^T E[F]e.$$

This result provides us with intuition about how much the eigenvalue λ would change according to a perturbation of ϵ. Specifically, the magnitude of change denoted by υ in the eigenvalue λ is upperbounded by the maximum eigenvalue of F. This change in the eigenvalue leads to the scaling of the particular dimension of the cMDS embedding λ corresponds to.

3.2.6 Maximum Likelihood MDS and MULTISCALE

Various probabilistic MDS methods with specific error assumptions have been proposed. In [31], Mackay assumes that the "original" coordinates have normally distributed errors that are independent in each dimension in the embedded space (the correlated error case can be simplified to the independent error case). As a result, the individual dissimilarities have the same distribution as a weighted sum of independent chi-square-distributed random variables. The embedding coordinates can be estimated using the maximum likelihood method. This method is implemented in an MDS program named MULTISCALE [26, 32].

3.2.7 Three-way MDS

Three-way MDS refers to a variant of MDS that is used to analyze many different dissimilarity matrices on the same collection of n objects. The different dissimilarity matrices can consist of dissimilarities judged by different people or different dissimilarity measures applied to the same group of observations. We say that these different dissimilarity matrices are from different "conditions", as mentioned in chapter 1, and we denote them by $\{\Delta_k, k \in \{1, \ldots, K\}\}$, where k indexes the conditions. The three-way array in which the third "way" is indexed by k can be interpreted as a tensor and comprises the stack of the two-dimensional dissimilarity matrices.

There are two ways of dealing with such three-way data. One can compute a separate MDS solution for each condition and match the configuration matrices by transformations. The second step in this two-step approach is similar to Generalized Procrustes Analysis section 7.3 of chapter 7. The general approach assumes that there is a common configuration $\mathbf{G}$ (n points in $\mathbb{R}^d$) and K $d \times d$ transformation matrices $\mathbf{T}_k$ such that each dissimilarity matrix Δ_k is obtained from the transformed configuration $\mathbf{G}_k = \mathbf{G}\mathbf{T}_k$. The inference problem then involves computing $\mathbf{T}_k, \quad k \in \{1, \ldots, K\}$.

Another approach mentioned in [26] involves mapping the dissimilarities (or proximities) into one distance matrix (which is the idea behind Multiple Kernel Learning 2.1).

3.3 Principal Components Analysis

Let X be a random vector of d dimensions and μ and Σ be its mean vector and covariance matrix, respectively. Then, for a given dimension $d' \leq d$, consider the successive (as i goes from 1 to d') maximization of

$$Var[u_i^T(X-\mu)] = E[u_i^T(X-\mu)(X-\mu)^T u_i] = u_i^T \Sigma u_i$$

with respect to u_i, where u_i is a d-dimensional unit vector ($u_i^T u_i = 1$) and $u_i \perp u_j, \quad 1 \leq j < i$[1]. The maximizers $\{u_i, 1 \leq i \leq d'\}$ are the principal directions and the projections of X via $\{u_i\}$ yield the principal *components* of X. These principal components capture the maximum variance possible from X, subject to the orthogonality constraints of all pairs of directions.

Another way of understanding PCA is considering the principal directions jointly. Consider $d' \times d$-matrix U whose rows are $u_i,\ i \in \{1, \ldots, d'\}$ which forms an orthonormal basis. $U^T U$ is the projection matrix that captures the maximum variance from X. That is, the elements of the random vector UX are uncorrelated and have the highest amount of "total variance" for any orthogonal projection of X. The term "total variance" should be interpreted as the sum of the variances of the variates in the principal directions which is equal to the trace of the covariance matrix of UX.

PCA also yields the best linear approximation of X in a least-squares sense for a particular projection dimension, d'. The matrix U, composed of the principal directions

[1]Due to the orthogonality constraints, the projections of X to the different dimensions are uncorrelated. *i.e.*, $E[u_i^T(X-\mu)(X-\mu)^T u_j] = 0, 1 \leq j < i$.

of X, minimizes $Var[X - (\Gamma^T \Gamma X)]$ with respect to Γ, when Γ is constrained to be a $d' \times d$ matrix.

For a sample of size n drawn from the same distribution as X, consider the sample estimates $\hat{\mu}$, $\hat{\Sigma}$. The sample principal components are computed by replacing the distribution parameters with the sample estimates

$$\hat{u_i} = \arg \max_{u_i^T u_i = 1, u_i \perp u_j, j<i} u_i^T \hat{\Sigma} u_i.$$

for the i^{th} principal component.

Suppose $\mathbb{X}$ is an $n \times d$ configuration matrix, representing the sample of X (n *i.i.d.* realizations of X). For simplicity of notation, suppose that the configuration is zero-centered, i.e., $\mathbf{1}^T \mathbb{X} = \mathbf{0}$. Additionally, suppose that $\mathbb{X}$ has a singular value decomposition $\mathbb{X} = V \Lambda U^T$, where the singular values on the diagonal of Λ are sorted in descending order and V and U are orthogonal $n \times n$ and $d \times d$ matrices, respectively. The PCA solution is given by the eigenvalue decomposition of $\hat{\Sigma}$ estimate, $\frac{1}{n}\mathbb{X}^T\mathbb{X} = U \left(\frac{1}{n}\Lambda^2\right) U^T$. The columns of U, $\{u_i\}$ are the principal directions. Note that we have U in the SVD of $\mathbb{X}$, so we do not need to compute $\mathbb{X}^T\mathbb{X}$.

The principal coordinates are the projections of the samples of X along the principal directions. For example, the first principal coordinates of the n samples would be given by $u_1^T \mathbb{X}^T$. The first d' principal coordinates can be represented with the $n \times d'$ configuration matrix $\mathbb{X}_{d'} = \mathbb{X} U_{d'}$. Therefore, using the SVD decomposition of $\mathbb{X}$, the principal

coordinates are found to be

$$\mathbb{X}_{d'} = V\Lambda U^T U_{d'} \tag{3.12}$$

$$= V\Lambda \begin{bmatrix} I_{d'} \\ \mathbf{0} \end{bmatrix} \tag{3.13}$$

$$= V\Lambda_{d'} \tag{3.14}$$

where $\Lambda_{d'}$ consist of the first d' columns of Λ.

3.3.1 Principal Components Analysis and Classical Multidimensional Scaling

The Principal Components Analysis method results in the same solution as classical Multidimensional Scaling (cMDS) when the dissimilarity matrix is $\Delta = \mathcal{D}(\mathbb{X})$. For cMDS, the eigenvalue decomposition of the $n \times n$ matrix $\mathbb{X}\mathbb{X}^T$ is used for embedding, while for PCA, the eigenvalue decomposition of the $d \times d$ matrix $\mathbb{X}^T\mathbb{X}$ is used to compute the principal directions (assuming $\mathbb{X}$ is a zero-centered configuration).

In the case of cMDS,

$$\Delta^2 = \mathbf{1}_n \boldsymbol{y} + \boldsymbol{y}\mathbf{1}_n - 2\mathbb{X}\mathbb{X}^T \tag{3.15}$$

where $\boldsymbol{y} = \left(\mathbb{X}\mathbb{X}^T\right)_{(d)}$ is an n-dimensional vector that consists of the diagonal of $\mathbb{X}\mathbb{X}^T$. For the classical MDS procedure, given the entrywise-squared distance matrix Δ^2, we

compute

$$Z = -\frac{1}{2} H \Delta^2 H. \tag{3.16}$$

Substituting Δ^2 with (3.15) in (3.16) gives $Z = \mathbb{X}\mathbb{X}^T$, which has the same non-zero eigenvalues as the PCA solution. The MDS solution, which is computed by the eigenvalue decomposition of Z, is given by the $n \times d'$ configuration matrix $V\Lambda_{d'}$. For the same embedding dimension, d', the two methods would yield the same configuration of n points in d'-dimensional space.

Chapter 4

An expository problem for Multiview Learning : Match detection

We are interested in problems in which the data sources are disparate and the inference task requires that observations from different data sources can be judged to be similar or dissimilar.

Consider a collection of English Wikipedia articles and French articles on the same topics. A pair of documents in different languages on the same topic are said to be "matched". The "matched" wiki documents are not necessarily direct translations of each other, and therefore, we do not restrict "matchedness" to be a well-defined bijection between documents in different languages. However the matched "documents" provide examples of "similar" observations coming from disparate sources, and we assume that the training data consist of a collection of "matched" documents.

The inference task that we consider is match detection, i.e., determining whether a new English article and a new French article have the same topic. Whereas a document in one language, say English, can be compared with other documents in English, a French document cannot be represented using the same features and therefore cannot be directly compared with English documents. It is necessary to derive a data representation in which the documents from different languages can be compared (are commensurate). We will use a finite-dimensional Euclidean space for this commensurate representation in which standard statistical inference tools can be used.

The label "disparate data" means that the observations are from different "conditions"; for example, the data might come from different types of sensors. Formally, the original data reside in a heterogeneous collection of spaces. In addition, the data might be structured and/or might reside in infinite-dimensional spaces. Therefore, it is possible that a feature representation of the data is not available or that inference using such a representation is fraught with complications (e.g., feature selection, non-i.i.d. data, infinite-dimensional spaces). This motivates our dissimilarity-centric approach.

Because we proceed to inference starting from a dissimilarity representation of the data, our methodology may be applicable to any scenario in which multiple dissimilarity measures are available. Some illustrative examples include pairs of images and their descriptive captions, the textual content and hyperlink graph structure of Wikipedia articles, and photographs taken under different illumination conditions. In each case, we have an intuitive notion of "matchedness": for photographs taken under different illu-

mination conditions, "matched" means that they are photographs of the same person. For a collection of linked Wikipedia articles, the different "conditions" are the textual content and hyperlink graph structure, "matched" means a text document, and a vertex corresponds to the same Wikipedia article.

The problem can be formally described as follows:

Let $(\Xi, \mathcal{F}, \mathcal{P})$ be a probability space, i.e., Ξ is a sample space, $\mathcal{F}$ is a sigma-field, and $\mathcal{P}$ is a probability measure. Consider K measurable spaces $\Xi_1, \cdots, \Xi_K$ and measurable maps $\pi_k : \Xi \to \Xi_k$. Each π_k induces a probability measure $\mathcal{P}_k$ on Ξ_k. We wish to identify a measurable metric space $\mathcal{X}$ (with distance function d) and measurable maps $\rho_k : \Xi_k \to \mathcal{X}$, inducing probability measures $\widetilde{\mathcal{P}}_k$ on $\mathcal{X}$, so that for $[x_1, \cdots, x_K]' \in \Xi_1 \times \cdots \times \Xi_K$, we may evaluate distances $d(\rho_{k_1}(x_{k_1}), \rho_{k_2}(x_{k_2}))$ in $\mathcal{X}$.

Given $\xi_1, \xi_2 \overset{iid}{\sim} \mathcal{P}$ in Ξ, we may reasonably hope that the random variable $d(\rho_{k_1} \circ \pi_{k_1}(\xi_1), \rho_{k_2} \circ \pi_{k_2}(\xi_1))$ is stochastically smaller than the random variable $d(\rho_{k_1} \circ \pi_{k_1}(\xi_1), \rho_{k_2} \circ \pi_{k_2}(\xi_2))$. That is, matched measurements $\pi_{k_1}(\xi_1), \pi_{k_2}(\xi_1)$ representing a single point ξ_1 in Ξ are mapped closer to each other than unmatched measurements $\pi_{k_1}(\xi_1), \pi_{k_2}(\xi_2)$ are in Ξ. This property allows for inference to proceed in the common representation space $\mathcal{X}$.

As the inference proceeds from dissimilarities, we cannot directly observe the object $\xi \in \Xi$, and the measurements $x_k = \pi_k(\xi) \in \Xi_k$ cannot be represented directly. Furthermore, we do not have knowledge of the maps π_k. We have well-defined dissimilarity measures $\delta_k : \Xi_k \times \Xi_k \to \mathbb{R}_+ = [0, \infty)$ such that $\delta_k(\pi_k(\xi_1), \pi_k(\xi_2))$ represents the "dis-

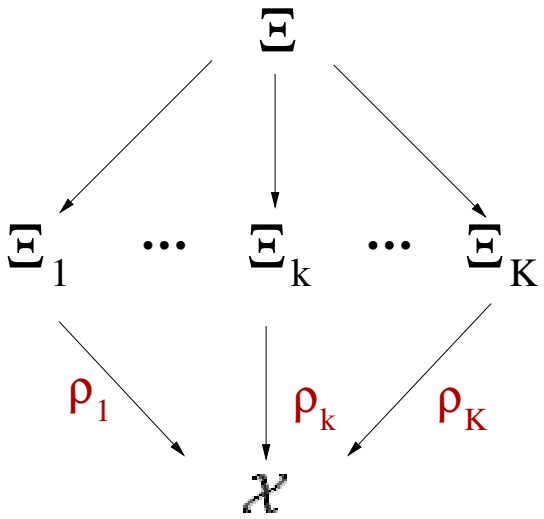


Figure 4.1: Maps π_k induce disparate data spaces Ξ_k from "object space" Ξ. Manifold matching involves using matched data $\{\boldsymbol{x}_{ik}\}$ to simultaneously learn maps $\rho_1, \ldots, \rho_K$ from disparate spaces $\Xi_1, \ldots, \Xi_K$ to a common "representation space" $\mathcal{X}$, for subsequent inference.

similarity" of the mappings of ξ_1 and ξ_2 under map π_k. The data we have consist of dissimilarities between a sample of n objects using $\{\delta_k\}_{k=1,\ldots,K}$. We propose to use sample dissimilarities for matched data in the disparate spaces Ξ_k to simultaneously learn maps ρ_k that allow for a powerful test of matchedness in the common representation space $\mathcal{X}$. This setting is visualized in Figure 4.1.

4.1 Problem Description

In the problem setting considered here, n different objects are measured under K different conditions (corresponding to, for example, K different sensors). We begin with dissimilarity measures. These will be represented in matrix form as K $n \times n$ matrices $\{\Delta_k, k = 1, \ldots, K\}$. In addition, for each condition, dissimilarities between a new object and the previous n objects $\{\mathcal{D}_k, k = 1, \ldots, K\}$ are available in the form of n-length vectors. Under the null hypothesis, "these new dissimilarities represent a single *new* object compared with the previous n objects", measured under K different conditions (the dissimilarities are matched). Under the alternative hypothesis, "the dissimilarities $\{\mathcal{D}_k\}$ represent separate *new* objects compared with the the previous n objects", measured under K different conditions (the dissimilarities are unmatched) [3].

In the case of the English-French Wikipedia example mentioned in the beginning of the chapter, dissimilarities between the new English article and n other English articles $(\mathcal{D}_1)$ are available, as they are for the new French article and other n French articles $(\mathcal{D}_2)$ [1]. The null hypothesis is that the new English and French articles are on the same topic, whereas the alternative hypothesis is that they are on different topics.

To derive a data representation in which dissimilarities from disparate sources $(\{\mathcal{D}_k\})$ can be compared, the dissimilarities must be embedded in a commensurate metric space in which the metric can be used to distinguish between "matched" and "unmatched" observations.

[1] In addition to the dissimilarities between articles in the same language $(\{\Delta_k\})$

To embed multiple dissimilarities $\{\Delta_k\}$ into a commensurate space, an omnibus dissimilarity matrix M [2] is constructed. Consider, for $K = 2$,

$$M = \begin{bmatrix} \Delta_1 & L \\ L^T & \Delta_2 \end{bmatrix} \tag{4.1}$$

where L is a matrix of imputed entries.

Remark For the purposes of exposition, we will consider $K = 2$; the generalization to $K > 2$ is straightforward.

Remark The imputation of the entries of L is an important detail. The entries correspond to dissimilarities between different conditions. We have clarified that we do not assume that these dissimilarities are available in our approach 1.3. Assuming that the dissimilarities are not strongly disparate, which means that the dissimilarities between a pair of objects in two different conditions are strongly correlated, the dissimilarities in L can be imputed as the average of the corresponding dissimilarities in Δ_1 and Δ_2:

$$L = \frac{\Delta_1 + \Delta_2}{2}.$$

For example, the dissimilarity between the i^{th} and j^{th} objects under the first and second conditions, respectively, ($[L]_{ij}$), can be imputed as the average of the dissimilarities between the i^{th} and j^{th} objects under the first condition ($[\Delta_1]_{ij}$) and between the same objects under the second condition ($[\Delta_2]_{ij}$), i.e., $[L]_{ij} = ([\Delta_1]_{ij} + [\Delta_2]_{ij})/2$. Note that this imputation would also make $[L]_{ii} = 0$ (the dissimilarities between the same object

[2] an $nk \times nk$ partitioned matrix whose diagonal blocks are given by $\{\Delta_k\}$

under the first and second conditions). This is consistent with our idea of matchedness, because we want the representations of "matched" observations to be highly similar. Therefore, 0 is a reasonable value for these dissimilarities between measurements of the same object. A more detailed reasoning for this choice is provided in 5.1.

Another imputation strategy is to treat nondiagonal elements of L as missing data (NA) and to set the diagonal entries ($[L]_{ii}$) to 0. We are then required to use an MDS embedding method that can deal with dissimilarity matrices that have NA entries. This is one of the justifications for our use of weighted raw stress 3.1 as the MDS criterion function. We mention this point along with other justifications in section 5.2.

We define the commensurate space to be $\mathbb{R}^d$, where the embedding dimension d is prespecified. The selection of d – a model selection problem – is a task that requires a great deal of attention. We will consider the effect of d on the performance; however, the general question of model selection requires detailed analysis, and we do not claim to have settled this question for our multiview data setting.

We use multidimensional scaling (MDS) [26] to embed the omnibus matrix in this space and obtain a configuration of $2n$ embedded points $\{\hat{x}_{ik}; i = 1, \ldots, n; k = 1, 2\}$ (which can be represented as $\hat{X}$, a $2n \times d$ matrix). The discrepancy between the inter-point distances of $\{\hat{x}_{ik}\}$ and the given dissimilarities in M is made as small as possible (as measured by an objective function $\sigma(\widetilde{X})$ [3]). In matrix form,

$$\hat{X} = \arg\min_{\tilde{X}} \sigma(\tilde{X}).$$

[3] $\sigma(\widetilde{X})$ that implicitly depends on the omnibus dissimilarity matrix M

Remark We will use x_{ik} to denote the – possibly notional – observation for the i^{th} object in the k^{th} condition, $\tilde{x}_{ik}$ to denote an argument of the objective function, and $\hat{x}_{ik}$ to denote the $\arg\min$ of the objective function, which are the coordinates of the embedded point. The notation for matrices $(X, \tilde{X}, \hat{X})$ follows the same convention.

Given the omnibus matrix M and the $2n \times d$ embedding configuration matrix $\hat{X}$ in the commensurate space, the OOS extension [33] for MDS will be used to embed the test dissimilarities $\mathcal{D}_1$ and $\mathcal{D}_2$. Once the test similarities are embedded as two points $(\hat{y}_1, \hat{y}_2)$ in the commensurate space, it is possible to compute the test statistic

$$\tau = d\left(\hat{y}_1, \hat{y}_2\right)$$

for the two "objects" represented by $\mathcal{D}_1$ and $\mathcal{D}_2$. For large values of τ, the null hypothesis will be rejected. If dissimilarities between matched objects are smaller than dissimilarities between unmatched objects with a large probability, and the embeddings preserve this stochastic ordering, we could reasonably expect the use of the test statistic to provide high statistical power.

4.2 Definition of an optimal embedding weight parameter: w^*

We have noted that we use the the weighted raw stress criterion function, $\sigma_W(\widetilde{X}; M)$ 3.1, for the joint embedding the omnibus matrix M. Rather than consider how each

entry of W separately effects the embedding, we will assume there is a single parameter $w \in (0, 1)$ which determines all of the entries of W, and consider its effect on the embedding, and its effect indirectly on the inference task. We will refer to this choice of the embedding weights as the simple weighting scheme. We do not provide details on how W is determined by w until $chapter\ 5$, because we first need to introduce fidelity and commensurability concepts.

Remark In our notation for this section, $(.)$ in superscript represents either one of the two hypotheses, either (m) or (u). In the former case, the expression refers to values under a "matched" hypothesis; in the latter, the expression refers to values under an "unmatched" hypothesis.

Let us denote the test dissimilarities ($\mathcal{D}_1$, $\mathcal{D}_2$) by ($\mathcal{D}_1^{(m)}$, $\mathcal{D}_2^{(m)}$) under the "matched" hypothesis and by ($\mathcal{D}_1^{(u)}$, $\mathcal{D}_2^{(u)}$) under the alternative. The OOS embedding of ($\mathcal{D}_1^{(m)}$, $\mathcal{D}_2^{(m)}$) involves the augmentation of the omnibus matrix M, which consists of n matched pairs of dissimilarities, with ($\mathcal{D}_1^{(m)}$, $\mathcal{D}_2^{(m)}$). The resulting augmented $(2n + 2) \times (2n + 2)$ matrix has the following form:

$$\Delta^{(m)} = \begin{bmatrix} & & \mathcal{D}_1^{(m)} & \vec{\mathcal{D}}_{NA} \\ & M & \vec{\mathcal{D}}_{NA} & \mathcal{D}_2^{(m)} \\ \mathcal{D}_1^{(m)T} & \vec{\mathcal{D}}_{NA}^T & 0 & \mathcal{D}_{NA} \\ \vec{\mathcal{D}}_{NA}^T & \mathcal{D}_2^{(m)T} & \mathcal{D}_{NA} & 0 \end{bmatrix} \tag{4.2}$$

where the scalar $\mathcal{D}_{NA}$ and $\vec{\mathcal{D}}_{NA}$ (an n-length vector of NAs) represent dissimilarities that are not available. In our JOFC procedure, these unavailable entries in $\Delta^{(m)}$ are either imputed using other dissimilarities that are available, in the way described in equation (3.1), or ignored in the embedding optimization. The former imputation method will result in a simpler notation, and thus, from now on, it will be assumed that the missing dissimilarities are imputed. Additionally, note that $\Delta^{(u)}$ has the same form as $\Delta^{(m)}$, where $\mathcal{D}_k^{(m)}$ is replaced by $\mathcal{D}_k^{(u)}$. Therefore, we will use $(.)$ in place of (m) and (u) to represent the two expressions under the two hypotheses with one expression.

We define the dissimilarity matrices $\{\Delta^{(m)}, \Delta^{(u)}\}$ which are of size $(2n+2)\times(2n+2)$ to be matrix-valued random variables.

Remark Suppose the objects in the k^{th} condition can be represented as points in a measurable space Ξ_k, and the dissimilarities in the k^{th} condition are given by a dissimilarity measure δ_k acting on pairs of points in Ξ_k. Assume that $\mathcal{P}_{(m)}$ is the joint probability distribution over matched objects, whereas the joint distribution of unmatched objects $\{k = 1, \ldots, K\}$ is $\mathcal{P}_{(u)}$. Assuming that the data are i.i.d., under the two hypotheses ("matched" and "unmatched", respectively), the $n+1$ pairs of objects are governed by the product distributions $\{\mathcal{P}_{(m)}\}^n \times \mathcal{P}_{(m)}$ and $\{\mathcal{P}_{(m)}\}^n \times \mathcal{P}_{(u)}$. The distributions of $\Delta^{(m)}$ and $\Delta^{(u)}$ are the induced probability distributions of these product distributions (induced by the dissimilarity measure δ_k applied to objects in k^{th} condition $\{k = 1, \ldots, K\}$).

We now consider the embedding of $\Delta^{(m)}$ and $\Delta^{(u)}$ with the weighted raw stress criterion function $\sigma_W(\widetilde{X}; \Delta^{(\cdot)})$. The arguments of the function are

$$\widetilde{X} = \begin{bmatrix} \widetilde{\mathcal{T}} \\ \widetilde{y}_1^{(\cdot)} \\ \widetilde{y}_2^{(\cdot)} \end{bmatrix}$$

where $\widetilde{\mathcal{T}}$ is the argument for the in-sample embedding of the first n pairs of matched points, $\widetilde{y}_1^{(\cdot)}$ and $\widetilde{y}_2^{(\cdot)}$ are the arguments for the embedding coordinates of the matched or unmatched pair, and the omnibus dissimilarity matrix $\Delta^{(\cdot)}$ is equal to $\Delta^{(m)}$ (or $\Delta^{(u)}$) for the embedding of the matched (unmatched) pair. Note that we use the simple weighting scheme; with a slight abuse of notation, we rewrite the criterion function as $\sigma_w(\widetilde{X}; \Delta^{(\cdot)})$, where $w \in (0, 1)$ is a scalar parameter. The embedding coordinates for the matched or unmatched pair $\hat{y}_1^{(\cdot)}, \hat{y}_2^{(\cdot)}$ are given by

$$\hat{y}_1^{(\cdot)}, \hat{y}_2^{(\cdot)} = \arg \min_{\widetilde{y}_1^{(\cdot)}, \widetilde{y}_2^{(\cdot)}} \left[\min_{\widetilde{\mathcal{T}}} \sigma_w \left(\begin{bmatrix} \widetilde{\mathcal{T}} \\ \widetilde{y}_1^{(\cdot)} \\ \widetilde{y}_2^{(\cdot)} \end{bmatrix}, \Delta^{(\cdot)} \right) \right].$$

Remark Note that the in-sample embedding of $\widetilde{\mathcal{T}}$ is necessary but irrelevant for the inference task; hence, the minimization with respect to $\widetilde{\mathcal{T}}$ is denoted by $\min$ instead $\arg\min$. It can be interpreted as a nuisance parameter for our hypothesis testing task.

Remark Note also that all of the random variables following the embedding, such as $\{\hat{y}_k^{(\cdot)}\}$, are dependent on w; for the sake of simplicity, this will be suppressed in the notation.

Under reasonable assumptions, the embeddings $\Delta^{(m)} \rightarrow \{\hat{y}_1^{(m)}, \hat{y}_2^{(m)}\}$ and $\Delta^{(u)} \rightarrow \{\hat{y}_1^{(u)}, \hat{y}_2^{(u)}\}$ are measurable maps for all $w \in (0, 1)$ [34]. Then, the distances between the embedded points are random variables and the test statistic τ is defined as the distance between the embedded points for a dissimilarity matrix sample of $\Delta^{(m)}$ or $\Delta^{(u)}$ (depending on whether the null or alternative hypothesis is true). Under the null hypothesis, the distribution of the statistic is governed by the distribution of $\hat{y}_1^{(m)}$ and $\hat{y}_2^{(m)}$; under the alternative, it is governed by the distribution of $\hat{y}_1^{(u)}$ and $\hat{y}_2^{(u)}$.

Then, the statistical power as a function of w is given by

$$\beta\left(w, \alpha\right) = 1 - F_{d\left(\hat{y}_1^{(u)}, \hat{y}_2^{(u)}\right)}\left(F^{-1}_{d\left(\hat{y}_1^{(m)}, \hat{y}_2^{(m)}\right)}(1-\alpha)\right),$$

where F_Y denotes the cumulative distribution function of Y. The area under the curve (AUC) as a function of w is defined as

$$AUC(w) = \int_0^1 \beta\left(w, \alpha\right) \, \mathrm{d}\alpha \, . \tag{4.3}$$

Although we might care about the optimal w with respect to $\beta\left(w, \alpha\right)$ (with a fixed Type I error rate α), it will be more convenient to define w^* in terms of the AUC function.

Finally, we define

$$w^* = \arg\max_w AUC\left(w\right).$$

Some important questions about w^* pertain to the nature of the AUC function. Although finding an analytical expression for the value of w^* is intractable, an estimate $\hat{w}^*$ based on estimates of $AUC(w)$ can be computed. For the Gaussian setting described

in section 6.1.1, a Monte Carlo simulation is used to find the estimate of $AUC(w)$ for different values of w.

4.2.1 Continuity of $AUC(\cdot)$

Let $T_0(w) = d(\hat{y}_1^{(m)}, \hat{y}_2^{(m)})$, and $T_a(w) = d(\hat{y}_1^{(u)}, \hat{y}_2^{(u)})$ denote the value of the test statistic under the distributions for the null and alternative hypotheses for the embedding with the simple weighting w. The AUC function can be written as

$$AUC(w) = \mathbb{P}\left[T_a(w) > T_0(w)\right],$$

where $T_a(\cdot)$ and $T_0(\cdot)$ can be considered stochastic processes whose sample paths are functions of w. We will prove that $AUC(w)$ is continuous with respect to w. We start with this lemma from [35].

Lemma 1. *Let z be a random variable. The functional $g(z;\gamma) = \mathbb{P}[z \geq \gamma]$ is upper semi-continuous in probability with respect to z. Furthermore, if $\mathbb{P}[z = \gamma] = 0$, $g(z;\gamma)$ is continuous in probability with respect to z.*

Proof. Suppose z_n converges to z in probability. Then, by definition, for any $\delta > 0$ and $\epsilon > 0$, $\exists N \in \mathbb{Z}^+$ such that for all $n \geq N$

$$\mathbb{P}[|z_n - z| \geq \delta] \leq \epsilon.$$

The functional $g(z;\gamma)$ is non-increasing with respect to γ. Therefore, for $\delta > 0$, $g(z_n;\gamma) - g(z;\gamma) \geq g(z_n;\gamma) - g(z;\gamma - \delta)$. Furthermore, $g(z;\gamma)$ is left-continuous with

respect to γ, and therefore, the difference between the two sides of the inequality can be made as small as desired.

$$\begin{aligned} g(z_n;\gamma) - g(z;\gamma-\delta) &= \mathbb{P}\left[z_n \geq \gamma\right] - \mathbb{P}\left[z \geq \gamma - \delta\right] && (4.4)\\ &\leq \mathbb{P}\left[\{z_n \geq \gamma\} \backslash \{z \geq \gamma - \delta\}\right] && (4.5)\\ &\leq \mathbb{P}\left[\{\{z_n \geq \gamma\} \backslash \{z \geq \gamma - \delta\}\} \cap \{z_n \geq z\}\right] && (4.6)\\ &= \mathbb{P}\left[\{z_n - z \geq \delta\}\right] \leq \epsilon. && (4.7) \end{aligned}$$

Because ϵ and δ are arbitrary, $\limsup_{n\to\infty}(g(z_n;\gamma) - g(z;\gamma)) = 0$ for any $\delta > 0$, i.e., $g(z;\gamma)$ is upper semi-continuous.

By arguments that are symmetric to (4.4)-(4.7), we can show that

$$g_(z;\gamma+\delta) - g(z_n;\gamma) \leq \epsilon. \tag{4.8}$$

In addition, assume that $\mathbb{P}\left[z = \gamma\right] = 0$. Then, $g(z;\gamma)$ is also right-continuous with respect to γ. Therefore, $g(z_n;\gamma) - g(z;\gamma) \leq g(z_n;\gamma) - g(z;\gamma+\delta)$, and the difference between the two sides of the inequality can be made as small as possible. Along with (4.8), this means that

$$\liminf_{n\to\infty}(g(z_n;\gamma) - g(z;\gamma)) = 0.$$

Therefore, $\lim_{n\to\infty} g(z_n;\gamma) = g(z;\gamma)$, i.e., $g(z;\gamma)$ is continuous in probability with respect to z. □

Theorem 1. *Let $T(w)$ be a stochastic process indexed by w in the interval (0,1). Assume that the process is continuous in probability (stochastic continuity) at $w = w_0$, i.e.,*

$$\forall a > 0 \quad \lim_{s \to w_0} \mathbb{P}\left[|T(s) - T(w_0)| \geq a\right] = 0 \tag{4.9}$$

for $w_0 \in (0, 1)$. Furthermore, assume that $\mathbb{P}\left[T(w_0) = 0\right] = 0$.

Then, $\mathbb{P}\left[T(w) \geq 0\right]$ is continuous at w_0.

Proof. Consider any sequence $w_n \to w_0$. Let $z_n = T(w_n)$ and $z = T(w_0)$ and choose $\gamma = 0$. Because $T(w)$ is continuous in probability at w_0 and $\mathbb{P}\left[T(w_0) = 0\right] = 0$, conditions for Lemma 1 hold, i.e.,as $w_n \to w_0$, z_n converges in probability to $z = T(w_0)$. By Lemma 1, we conclude that $g(T(w_n); 0) = \mathbb{P}\left[T(w_n) \geq 0\right]$ converges to $g(T(w_0); 0)$. Therefore, $g(T(w); 0)$ is continuous with respect to w. □

Corollary 1. *If $\mathbb{P}[T_a(w) - T_0(w) = 0] = 0$ and $T_a(w)$, $T_0(w)$ are continuous in probability for all $w \in (0, 1)$, then $AUC(w) = \mathbb{P}\left[T_a(w) - T_0(w) > 0\right]$ is continuous with respect to w in the interval $(0, 1)$.*

Proof. Let $T(w) = T_a(w) - T_0(w)$. Then, Theorem 1 applies everywhere in the interval (0,1). □

In any closed interval that is a subset of $(0, 1)$, the AUC function is continuous and therefore attains its global maximum in that closed interval.

We do not have closed-form expressions for the distributions under the null and alternative hypotheses of the test statistic τ (as a function of w), and therefore, we cannot

provide a rigorous proof of the uniqueness of w^*. However, for various data settings, the simulations described in chapter 10 always resulted in *unimodal* estimates for the AUC function, which indicates a unique w^* value.

We should also mention that the stochastic continuity of the test statistics $T(w)$ as a function of w is a reasonable assumption. Discontinuity in the test statistic can arise as a result of discontinuity of the embedded configurations with respect to the w parameter. The embedded configurations, which are the global minimizers of the criterion function, can have discontinuities if there exist multiple local minima, and infinitesimal changes in w will change the ordering of the "distinct" local minima. Although we present an example in which multiple local minima of the criterion function lead to a discontinuity of the embedded configuration with respect to w in chapter 9, other than such carefully constructed examples, we do not expect such discontinuities in the embedded configurations to occur for data generated from continuous case probability distributions. One can conclude that the stochastic continuity of the test statistic with respect to w is a valid assumption.

Chapter 5

Fidelity and Commensurability

5.1 The concepts of Fidelity and Commensurability

For the sake of argument, assume that the source of dissimilarities are actually observations that are vectors in Euclidean space. In general, MDS with raw stress will not result in a perfect reconstruction of the original observations. Note that this point is not relevant to our work, as the objective of the (joint) embedding is not *perfect* reconstruction, but rather the best embedding for the inference task. What is considered a "good" representation will be dependent on how well the original dissimilarities that are relevant to the inference task are preserved. "Fidelity" and "Commensurability" quantify this preservation of information.

Regardless of the inference task, to expect reasonable performance from the embedded data in the commensurate space for the inference task at hand, it is necessary to pay heed to these two error criteria:

- Fidelity describes how well the mapping to commensurate space preserves the original dissimilarities. The *loss of fidelity* can be measured using the within-condition *fidelity error*, given by

$$\epsilon_{f_{(k)}} = \frac{1}{\binom{n}{2}} \sum_{1 \leq i < j \leq n} (d(\widetilde{\boldsymbol{x}}_{ik}, \widetilde{\boldsymbol{x}}_{jk}) - \delta_{ijkk})^2.$$

  Here, δ_{ijkk} is the dissimilarity between the i^{th} object and the j^{th} object when both objects are in the k^{th} condition, and $\widetilde{\boldsymbol{x}}_{ik}$ is the embedded representation of the i^{th} object for the k^{th} condition; $d(\cdot, \cdot)$ is the Euclidean distance function.

- Commensurability describes how well the mapping to commensurate space preserves the matchedness of matched observations. The *loss of commensurability* can be measured by the between-condition *commensurability error*, which is given by

$$\epsilon_{c_{(k_1,k_2)}} = \frac{1}{n} \sum_{1 \leq i \leq n; k_1 < k_2} (d(\widetilde{\boldsymbol{x}}_{ik_1}, \widetilde{\boldsymbol{x}}_{ik_2}) - \delta_{iik_1k_2})^2$$

  for conditions k_1 and k_2; $\delta_{iik_1k_2}$ is the dissimilarity between the i^{th} object under conditions k_1 and k_2. Although the between-condition dissimilarities of the same object, $\delta_{iik_1k_2}$, are not available, it is reasonable to set these dissimilarities to 0 for all i, k_1, k_2. These dissimilarities correspond to diagonal entries of the submatrix L in the omnibus matrix M in equation (4.1). Setting these diagonal entries to 0

forces matched observations to be embedded close to each other. It is possible that this choice for between-condition dissimilarities is not optimal. However, seeking optimal values for these unknown dissimilarities would only serve to distract us from the problem of interest, namely, how much fidelity and commensurability are to be preserved for the inference task.

When the between-condition dissimilarities of the same object are imputed with zeros, the commensurability error term becomes

$$\epsilon_{c_{k_1k_2}} = \frac{1}{n} \sum_{1 \le i \le n; k_1 < k_2} (d(\widetilde{\boldsymbol{x}}_{ik_1}, \widetilde{\boldsymbol{x}}_{ik_2})))^2$$

The between-condition *separability error* is given by

$$\epsilon_{s_{k_1k_2}} = \frac{1}{\binom{n}{2}} \sum_{1 \le i < j \le n; k_1 < k_2} (d(\widetilde{\boldsymbol{x}}_{ik_1}, \widetilde{\boldsymbol{x}}_{jk_2}) - \delta_{k_1k_2}(\boldsymbol{x}_{ik_1}, \boldsymbol{x}_{jk_2}))^2.$$

The between-condition dissimilarities of different objects, $\delta_{ijk_1k_2}, i \neq j$, in the "separability" criterion are also not available. Ignoring them in the embedding by setting the associated weights in the raw stress function to be 0 is a reasonable choice. [1] We prefer these choices for between-condition dissimilarities to restrict our attention to the fidelity-commensurability tradeoff. An alternative solution would be to impute these dissimilarities using other available dissimilarities. This imputation approach is discussed in 4.1.

While the expressions for *fidelity* and *commensurability* errors are specific to the joint embedding of disparate dissimilarities, the concepts of fidelity and commensurability are

[1] These dissimilarities correspond to off-diagonal entries of the submatrix L in the omnibus matrix M in equation (4.1).

general enough to be applicable to other dimensionality reduction methods for multi-view data. For example, if the dissimilarities between different conditions were available, or imputed, a joint embedding could be performed using classical MDS. This joint embedding would also jointly optimize fidelity and commensurability, but we would have no control over which dissimilarities are prioritized for preservation in the embedding. We could thus not control the fidelity and commensurability tradeoff. This tradeoff is important for the inference task: we use simulations to show that there are significant improvements in statistical power when commensurability is prioritized compared with the baseline *uniform-weighting* case.

In general, we note that the omnibus embedding approach using any variant of MDS attempts to jointly optimize fidelity and commensurability by minimization of some measure of discrepancy between the given dissimilarities (which are either between-condition or within-condition dissimilarities) and the distances of the embedded configuration. This is most obvious in the raw stress version of MDS, because the individual terms can be separated according to whether they are contributing to the fidelity or commensurability error.

Consider the weighted raw stress criterion $\sigma_W(\cdot)$ with a weighting matrix W, given in equation (3.1). The omnibus matrix M (4.1) is a partitioned matrix consisting of matrices from two different conditions ($k = 1, 2$). The entries of the matrix will be indexed by a 4-tuple, i, j, k_1, k_2, which refers to the entry in the i^{th} row and j^{th} column of the block matrix in the k_1^{th} row partition and the k_2^{th} column partition. For example, the

entry $M_{2n,n}$ will have the indices $\{i, j, k_1, k_2\} = \{n, n, 2, 1\}$ in the new 4-tuple indexing scheme. The matrix-valued function $\mathcal{D}(\cdot)$ and the weight matrix W, which are of the same size as M, follow the same 4-tuple indexing. Then, the weighted raw stress for the joint embedding with the weight matrix W is

$$\begin{aligned}\sigma_W(\cdot) = & \sum_{i,j,k_1,k_2} w_{ijk_1k_2}(\mathcal{D}_{ijk_1k_2}(\cdot) - M_{ijk_1k_2})^2 \\ = & \underbrace{\sum_{i=j,k_1<k_2} w_{ijk_1k_2}(\mathcal{D}_{ijk_1k_2}(\cdot) - M_{ijk_1k_2})^2}_{Commensurability} + \underbrace{\sum_{i<j,k_1=k_2} w_{ijk_1k_2}(\mathcal{D}_{ijk_1k_2}(\cdot) - M_{ijk_1k_2})^2}_{Fidelity} \\ + & \underbrace{\sum_{i<j,k_1<k_2} w_{ijk_1k_2}(\mathcal{D}_{ijk_1k_2}(\cdot) - M_{ijk_1k_2})^2}_{Separability}. \end{aligned} \tag{5.1}$$

Because $\delta_{k_1k_2}(\boldsymbol{x}_{ik_1}, \boldsymbol{x}_{ik_2})$ are set to 0, the corresponding entries of the matrix M which appear in the commensurability terms of the sum will be 0.

Because the separability error is ignored, the weights for separability terms are chosen to be 0. Thus, the off-diagonal elements of L in equation (4.1) can also be ignored. When the separability terms are removed from equation (5.1), the resulting equation is the sum of the fidelity and commensurability error terms:

$$\sigma_W(\cdot) = \underbrace{\sum_{i=j,k_1<k_2} w_{ijk_1k_2}(\mathcal{D}_{ijk_1k_2}(\cdot))^2}_{Commensurability} + \underbrace{\sum_{i<j,k_1=k_2} w_{ijk_1k_2}(\mathcal{D}_{ijk_1k_2}(\cdot) - M_{ijk_1k_2})^2}_{Fidelity}.$$

This motivates our reference to the omnibus embedding approach as JOFC.

5.2 Fidelity and Commensurability Tradeoff

The weights in the raw stress function allow us to address the question of the optimal tradeoff of fidelity and commensurability. Let $w \in (0, 1)$. Setting the weights ($\{w_{ijk_1k_2}\}$) for the commensurability and fidelity terms to w and $1-w$, respectively, will allow us to control the relative importance of fidelity and commensurability terms in the objective function. Let us denote the raw stress function with these simple weights by $\sigma_w(\widetilde{X}; M)$. With simple weighting, when $w = 0.5$, all terms in the objective function have the same weights. We will refer to this weighting scheme in the rest of this dissertation as *uniform weighting*. The alternative scheme, $w \neq 0.5$, is called *nonuniform weighting*.

The initial expectation in the investigation of fidelity and commensurability was that there is a w^* that is optimal for the specific match detection task 4 (the w value, which yields the best statistical power for hypothesis testing) . In fact, the exploratory simulations presented in 10.1 confirm that the power of the tests varies with w and indicate the range in which the optimal w^* lies, assuming it exists. We show that w exists under certain conditions for the match detection task. Although we cannot provide a rigorous proof of the uniqueness of w^*, for various data settings, simulations in section 10.1 always resulted in *unimodal* estimates for the AUC function, which indicates a unique w^* value. Specifically, for the match detection task, we provide evidence in section 10.1 that uniform weighting does not necessarily yield the best fidelity-commensurability tradeoff in terms of subsequent inference and that one should consider nonuniform weighting

for better performance in the inference task [36].

Chapter 6

Data Models for the Match Detection Task

6.1 Two data settings for Match Detection

In this chapter, we present two generative data models that illustrate the idea of matchedness. We will use the Multivariate Normal and Dirichlet probability distributions, with the parameters p,r,q,c to generate matched dissimilarity data in $K = 2$ conditions.

6.1.1 Gaussian setting

Let $\Xi_1 = \mathbb{R}^p$ and $\Xi_2 = \mathbb{R}^p$. Let $\boldsymbol{\alpha}_i \sim^{iid} MultivariateNormal(\mathbf{0}, I_p)$ represent n "objects". Let $X_{ik} \sim^{iid} MultivariateNormal(\boldsymbol{\alpha_i}, \Sigma), \quad i \in \{1, \ldots, n\}, k \in \{1, 2\}$ rep-

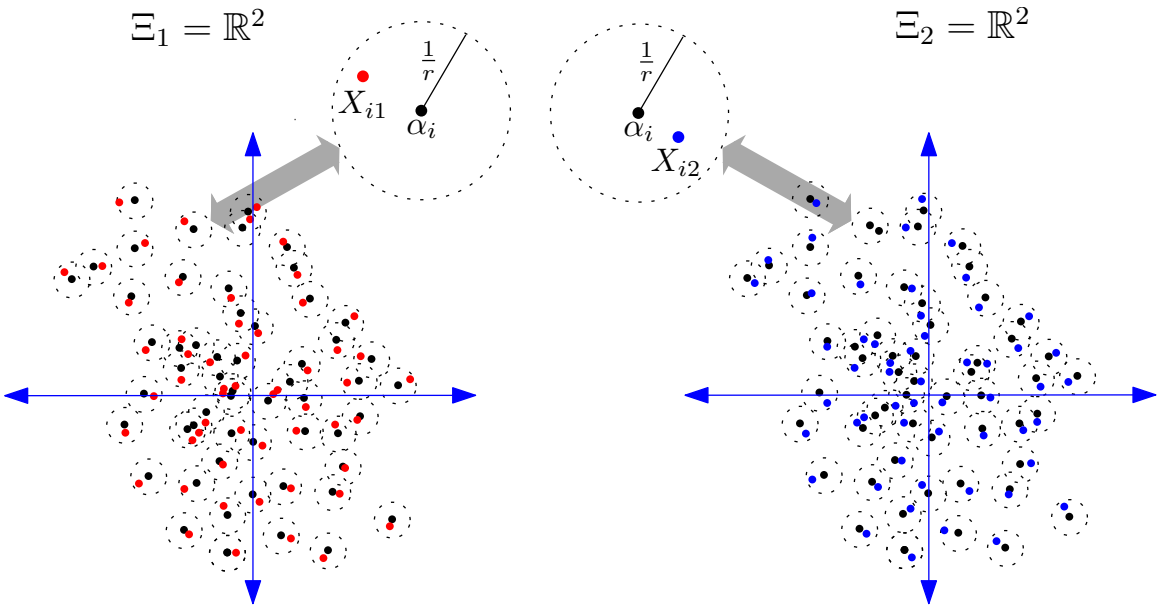


Figure 6.1: For the Gaussian setting (Section 6.1.1), the α_i are denoted by black points, and the X_{ik} are denoted by red and blue points.

resent $K = 2$ matched measurements (each under a different condition). Σ is a positive-definite $p \times p$ matrix such that the maximum eigenvalue of Σ $\frac{1}{r}$ and other eigenvalues are drawn from the uniform distribution between 0 and $\frac{1}{r}$ (see Figure 6.1).

The parameter r controls the variability between "matched" measurements. If r is large, it is expected that the distance between matched measurements X_{i1} and X_{i2} is stochastically smaller than X_{i1} and $X_{i'2}$ for $i \neq i'$; $i, i' \in \{1, \ldots, n\}$; if r is small, then "matched" is not informative in terms of the similarity of measurements. Smaller r values will make the decision problem harder and will lead to higher rates of errors or tests with smaller power for a fixed type I error rate α.

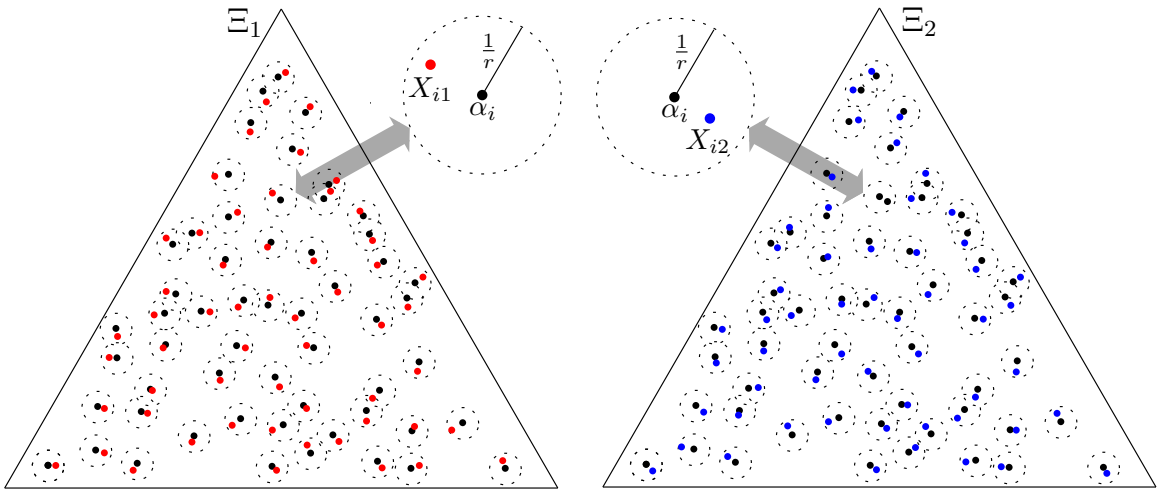


Figure 6.2: For the Dirichlet setting (Section 6.1.2), the α_i are denoted by black points, and the X_{ik} are denoted by red and blue points .

6.1.2 Dirichlet setting

Let $S^p = \{\boldsymbol{x} : \boldsymbol{x} \in \mathbb{R}^{(p+1)}, \sum_{l=1}^{p} x_l = 1\}$ be the standard p-simplex in $\mathbb{R}^{p+1}$. Let $\Xi_1 = S^p$ and $\Xi_2 = S^p$. Denote a $p+1$-length vector of ones by $\mathbf{1}_{p+1} \in \mathbb{R}^{(p+1)}$. Let $\boldsymbol{\alpha}_i \sim^{iid} Dirichlet(\mathbf{1}_{p+1})$ represent n "objects", and let $X_{ik} \sim^{iid} Dirichlet(r\boldsymbol{\alpha}_i + \mathbf{1}_{p+1})$ represent K measurements (Figure 6.2).

The parameter r controls the variability between "matched" measurements.

6.1.3 Noise

Measurements $\{X_{ik}\}$ carry the signal that is relevant to the exploitation task. Noise dimensions can be introduced to the measurements by concatenating a q-dimensional error vector whose magnitude is controlled by the parameter c. The noisy measurements

will be represented by the random vectors

$$\mathfrak{X}_{ik} = [(1-c)X_{ik} \;\; cE_{ik}] \tag{6.1}$$

where

$$E_{ik} \sim^{iid} Dirichlet(\mathbf{1}_{(q+1)}) \tag{6.2}$$

for the Dirichlet setting and

$$E_{ik} \sim^{iid} MultivariateNormal(\mathbf{0}, (1+\frac{1}{r})I_{q+1}) \tag{6.3}$$

for the Gaussian setting. $\mathfrak{X}_{ik}$ will be used instead of X_{ik} to compute dissimilarities in the "noisy" version of the problem. These noisy measurements allow for the comparison of different methods applied to the problem with respect to their robustness.

Chapter 7

Procrustes Analysis for Data Fusion

7.1 Procrustes Analysis

Given two configurations of n points in d-dimensional Euclidean space, Procrustean methods fit one configuration to the other so that the points align as well as possible in the ℓ_2-sense. Let us denote the configurations by two $n \times d$ matrices: $\mathbb{X}_1$, $\mathbb{X}_2$. The most general version of this method seeks an affine transformation with ρ *only* rotation, reflection, scaling and translation components that transforms the points in the configuration $\mathbb{X}_2$ to align with the target configuration $\mathbb{X}_1$. The transformation ρ is chosen such that the sum of squares of the distances from each ρ-transformed point of $\mathbb{X}_2$ to its corresponding point is minimized. For notational convenience, let $\varrho_\rho : \mathbb{M}_{n\times d} \to \mathbb{M}_{n\times d}$ be the mapping applied to a configuration matrix such as $\mathbb{X}_1$ whose rows correspond to the point coordinates, when each point is mapped by ρ. That is, ϱ_ρ applies the ρ trans-

formation to every point in the configuration. For example, if ρ is the identity map, $\varrho_\rho(\mathbb{X}_1) = \mathbb{X}_1$.

It is also possible to introduce extra constraints on the affine transformation, such as requiring the translation component to be a zero vector (if both of the point configurations are zero-centered) or setting the scaling component to 1 (if only rigid transformations are allowed). First, let us consider the general case where $\rho(z) = sz\mathbf{Q} + \mathbf{t}$, where $\mathbf{Q} \in \mathbb{M}_{d\times d}$, $s \in (0, \infty)$, $\mathbf{t} \in \mathbb{R}^d$. For configuration matrices, the mapping is $\varrho_\rho(\mathbb{X}_2) = s\mathbb{X}_2\mathbf{Q} + \mathbf{1}\mathbf{t}^T$. We will derive the components of the Procrustean transformation, s, $\mathbf{Q}$ and $\mathbf{t}$, following [26].

We seek to minimize

$$\begin{aligned}\mathcal{L}(s, \mathbf{Q}, \mathbf{t}) &= \|\mathbb{X}_1 - (s\mathbb{X}_2\mathbf{Q} + \mathbf{1}\mathbf{t}^T)\|_F^2 \\ &= \operatorname{trace}\left(\mathbb{X}_1 - (s\mathbb{X}_2\mathbf{Q} + \mathbf{1}\mathbf{t}^T)\right)^T \left(\mathbb{X}_1 - (s\mathbb{X}_2\mathbf{Q} + \mathbf{1}\mathbf{t}^T)\right).\end{aligned}$$

Setting the gradient $\frac{\partial \mathcal{L}}{\partial \boldsymbol{t}} = 2\left(\mathbb{X}_1^T\mathbf{1} - (s\mathbf{Q}^T\mathbb{X}_2^T\mathbf{1} + n\mathbf{t})\right)$ to $\mathbf{0}$, we solve for $\hat{\mathbf{t}}$:

$$\hat{\mathbf{t}} = n^{-1}\left(\mathbb{X}_1^T\mathbf{1} - s\mathbf{Q}^T\mathbb{X}_2^T\mathbf{1}\right).$$

Putting $\hat{\mathbf{t}}$ into $\mathcal{L}(s, \mathbf{Q}, \mathbf{t})$, we obtain

$$\mathcal{L}(s, \mathbf{Q}, \hat{\mathbf{t}}) = \operatorname{trace}\left((\boldsymbol{I}_n - \frac{\mathbf{1}\mathbf{1}^T}{n})\mathbb{X}_1 - (s\mathbb{X}_2\mathbf{Q})\right)^T \left((\boldsymbol{I}_n - \frac{\mathbf{1}\mathbf{1}^T}{n})\mathbb{X}_1 - (s\mathbb{X}_2\mathbf{Q})\right).$$

Let us denote the centering matrix $\left(\boldsymbol{I}_n - \frac{\mathbf{1}\mathbf{1}^T}{n}\right)$ with $\boldsymbol{H}$. Setting $\frac{\partial \mathcal{L}}{\partial s} = 2\operatorname{trace} s\mathbb{X}_2{}^T\boldsymbol{H}\mathbb{X}_2 - 2\operatorname{trace}\mathbb{X}_1{}^T\boldsymbol{H}\mathbb{X}_2\mathbf{Q} = \mathbf{0}$, we obtain $\hat{s} = \frac{\operatorname{trace}\mathbb{X}_1{}^T\boldsymbol{H}\mathbb{X}_2\mathbf{Q}}{\operatorname{trace}\mathbb{X}_2{}^T\boldsymbol{H}\mathbb{X}_2}$. Putting $\hat{s}$ into $\mathcal{L}(s, \mathbf{Q}, \hat{\mathbf{t}})$, we obtain $\mathcal{L}(\hat{s}, \mathbf{Q}, \hat{\mathbf{t}}) = \operatorname{trace}\mathbb{X}_1^T\boldsymbol{H}\mathbb{X}_1 - \frac{\left(\operatorname{trace}\mathbb{X}_1^T\boldsymbol{H}\mathbb{X}_2\mathbf{Q}\right)^2}{\operatorname{trace}\mathbb{X}_2^T\boldsymbol{H}\mathbb{X}_2}$.

The final step is computing $\mathbf{Q}$. Note that the only term in $\mathcal{L}(\hat{s}, \mathbf{Q}, \hat{\mathbf{t}})$ that depends on $\mathbf{Q}$ is $\left[\text{trace}\left(\mathbb{X}_1^T \boldsymbol{H} \mathbb{X}_2 \mathbf{Q}\right)\right]^2$. Subject to $\mathbf{Q}\mathbf{Q}^T = \mathbf{Q}^T\mathbf{Q} = \boldsymbol{I}_n$, minimizing $-\left[\text{trace}\left(\mathbb{X}_1^T \boldsymbol{H} \mathbb{X}_2 \mathbf{Q}\right)\right]^2$ is equivalent to minimizing $-\text{trace}\left(\mathbb{X}_1^T \boldsymbol{H} \mathbb{X}_2 \mathbf{Q}\right)$ (Because $\hat{s} > 0$, minimizing $-x$ is the same as minimizing $-x^2$ given a constraint on x). Thus,

$$\hat{\mathbf{Q}} = \arg \min_{Q^T Q = \boldsymbol{I}_n} -\text{trace}\left(\mathbb{X}_1^T \boldsymbol{H} \mathbb{X}_2 \mathbf{Q}\right). \tag{7.1}$$

Therefore, the solution for $\mathbf{Q}$ in the general Procrustes problem is equivalent to the solution of the orthogonal Procrustes problem.

For the orthogonal Procrustes problem, we seek an orthonormal matrix $\mathbf{Q}^*$ that minimizes the sum of squared distances between the target configuration $\mathbb{X}_1$ and the configuration $\mathbb{X}_2$ transformed by $\mathbf{Q}^*$, i.e., $\mathbf{Q}^* = \arg\min_{\mathbf{Q}^T\mathbf{Q} = \boldsymbol{I}_n} \|\mathbb{X}_1 - \mathbb{X}_2\mathbf{Q}\|_F^2$, where $\|\cdot\|_F$ is the Frobenius norm on matrices. Simplifying the norm expression, we obtain $\|\mathbb{X}_1 - \mathbb{X}_2\mathbf{Q}\|_F^2 = \text{trace}\,(\mathbb{X}_1 - \mathbb{X}_2\mathbf{Q})^T(\mathbb{X}_1 - \mathbb{X}_2\mathbf{Q}) = \text{trace}\,(\mathbb{X}_1^T\mathbb{X}_1 + \mathbb{X}_2^T\mathbb{X}_2) - 2\,\text{trace}\,(\mathbb{X}_1^T\mathbb{X}_2\mathbf{Q})$. Because the first term is independent of $\mathbf{Q}$, we can ignore that term. The second term is equivalent to (7.1) when $\mathbb{X}_1^T\mathbf{1} = \mathbb{X}_2^T\mathbf{1} = \mathbf{0}$. Then, the solution for $\mathbf{Q}^*$ is the $d \times d$ orthogonal matrix that maximizes $\text{trace}\,(\mathbb{X}_1^T\mathbb{X}_2\mathbf{Q})$.

Consider the singular value decomposition $\mathbb{X}_\mathbf{1}^\mathbf{T}\mathbb{X}_\mathbf{2} = \boldsymbol{U}\boldsymbol{\Sigma}\boldsymbol{V}^T$. The expression to be minimized can be written as $\text{trace}\,\boldsymbol{U}\boldsymbol{\Sigma}\boldsymbol{V}^T\mathbf{Q}$, which is equal to $\text{trace}\,\boldsymbol{V}^T\mathbf{Q}\boldsymbol{U}\boldsymbol{\Sigma}$ due to the circular invariancy of the trace operation.

Note that for an orthogonal matrix T and a diagonal matrix Λ with non-negative entries ($\Lambda_{ii} \geq 0$),

$$\text{trace}\,T\Lambda \leq \text{trace}\,\Lambda$$

with equality if $T = \boldsymbol{I}$.

Note that $\boldsymbol{\Sigma}$ is diagonal with nonnegative entries and that $\boldsymbol{Z} = \boldsymbol{V}^T\mathbf{Q}\boldsymbol{U}$ is also orthogonal. To see why $\boldsymbol{Z}$ is orthogonal, consider

$$\begin{aligned}\boldsymbol{Z}\boldsymbol{Z}^T &= \boldsymbol{V}^T\mathbf{Q}\boldsymbol{U}\boldsymbol{U}^T\mathbf{Q}^T\boldsymbol{V} \\ &= \boldsymbol{V}^T\mathbf{Q}\boldsymbol{I}_n\mathbf{Q}^T\boldsymbol{V} \\ &= \boldsymbol{V}^T\boldsymbol{I}_n\boldsymbol{V} \\ &= \boldsymbol{I}_n\end{aligned}$$

Each step is justified by the fact that the SVD of $\mathbb{X}_1^{\mathbf{T}}\mathbb{X}_{\mathbf{2}}$ results in matrices U and V with orthogonal columns, and R is already known to be orthogonal. Therefore,

$$\operatorname{trace} \boldsymbol{V}^T\mathbf{Q}\boldsymbol{U}\boldsymbol{\Sigma} \leq \operatorname{trace} \boldsymbol{\Sigma}$$

with equality if $\boldsymbol{V}^T\mathbf{Q}\boldsymbol{U} = \boldsymbol{I}_n$. The solution that achieves the bound is $\hat{\mathbf{Q}} = \boldsymbol{V}\boldsymbol{U}^T$.

7.2 Procrustes Analysis for Manifold Matching

Because separate condition dissimilarities are available, a straightforward approach is to embed each conditional dissimilarity matrix, Δ_1 and Δ_2, separately in d-dimensional Euclidean space (we denote these embedded configurations by the configuration matrices $\mathbb{X}_1$ and $\mathbb{X}_2$, respectively) and then find a mapping function $\rho^* : \mathbb{R}^d \to \mathbb{R}^d$, that maps each point in $\mathbb{X}_2$ approximately to its corresponding point in $\mathbb{X}_1$. This approach can be considered a specific example of the general setting in Figure 6.1 in which the commensurate

space is d-dimensional Euclidean space, ρ_1 is the identity map, and $\rho_2 = \rho^*$.

The mapping ρ^* is estimated by using Procrustes Analysis on the training data. This estimate, ρ, makes the separate MDS embeddings as commensurate as possible. Once such a mapping is computed, one can OOS embed new dissimilarities for each condition (separately) and use ρ to make the embeddings commensurate. One can then compute the test statistic τ (the distance between commensurate embeddings) for the hypothesis testing problem in chapter 4. This approach will be referred to as P∘M – Procrustes∘MDS.

Note that the Procrustes transformation ρ is limited to an affine transformation consisting of rotation and reflection and possibly also scaling components. The optimal mapping might very well be nonlinear. If a larger class of mappings is considered, this would result in a smaller model bias but also in a larger variability for the mapping function. By only considering the class of linear transformations, it is possible to learn ρ with the limited sample size.

7.2.1 Relation of P∘M and JOFC

In this section, we explain where Procrustes∘MDS stands in relation to the Fidelity--Commensurability tradeoff view of multiview dissimilarities.

Suppose, in equation (5.1), that the weights are chosen to be $w_{ijk_1k_2} = w$ for commensurability terms and $w_{ijk_1k_2} = 1 - w$ for fidelity terms. For the resulting weight

matrix W, define

$$f_w(\mathcal{D}(\cdot), M) = \sigma_W(\cdot) \tag{7.2}$$

where M is the omnibus matrix obtained from a given pair of dissimilarity matrices, Δ_1 and Δ_2, as in equation (4.1). As w goes to 0, the configuration embedded by JOFC converges to a configuration equivalent to (up to rotation and reflection) the configuration embedded by P∘M.

Theorem 2. *Define $\sigma(\cdot) = \sigma_{W=\mathbf{1}}(\cdot)$ (unweighted raw stress), where $\mathbf{1}$ is a matrix of 1's. Let $\mathbf{X}_1$ and $\mathbf{X}_2$ be the corresponding $n \times p$ configuration matrices with column means of $\mathbf{0}$ (obtained from separately embedding Δ_1 and Δ_2 by minimizing the raw stress $\sigma(\cdot)$). Let $\mathbf{Q} = \arg\min_{\mathbf{P^T P = P P^T = I}} \|\mathbf{X}_1 - \mathbf{X}_2\mathbf{P}\|_F^2$, $\tilde{\mathbf{X}}_2 = \mathbf{X}_2\mathbf{Q}$, and let $\mathbf{X} = \begin{bmatrix} \mathbf{X}_1 \\ \tilde{\mathbf{X}}_2 \end{bmatrix}$.*

For $w > 0$, let $\mathbf{Y}_w = \begin{bmatrix} \mathbf{Y}_1 \\ \mathbf{Y}_2 \end{bmatrix}$ be a $2n \times p$ configuration matrix obtained by the minimization of $f(\mathcal{Y}, M) = (1-w)\left(\sigma(\mathcal{Y}_1) + \sigma(\mathcal{Y}_2)\right) + w\|\mathcal{Y}_1 - \mathcal{Y}_2\|_F^2$ with respect to $\mathcal{Y} = \begin{bmatrix} \mathcal{Y}_1 \\ \mathcal{Y}_2 \end{bmatrix}$ with the constraint that $\mathcal{Y}_1$ and $\mathcal{Y}_2$ are two $n \times p$ configuration matrices with column means $\mathbf{0}$. Then,

$$lim_{w \to 0} \mathbf{Y}_w = \mathbf{XR}$$

for a $p \times p$ orthogonal matrix $\mathbf{R}$. ($\mathbf{R}$ is a transformation matrix with a rotation and possibly a reflection component.)

7.3 Generalized Procrustes Analysis ($K > 2$)

The Generalized Procrustes analysis is the extension of Procrustes analysis to more than two configurations of points. This extension has been studied in [37]. Suppose we have K configurations: $\mathbb{X}_1, \mathbb{X}_2, \ldots \mathbb{X}_K$. We wish to find K Procrustean transformations $\tau_k(\mathbb{X}_k) = s_k \mathbb{X}_k \boldsymbol{Q}_k + \boldsymbol{t}_k$ such that

$$\sum_{kl} \|\tau_k(\mathbb{X}_k) - \tau_k(\mathbb{X}_k)\|_F^2$$

is minimized. This problem does not have a single-step analytical solution for all of the components, similar to the original Procrustes analysis problem. The translation components, $\boldsymbol{t}_k$, of the transformations can be solved by subtracting the column sums of $\mathbb{X}_k$ ($\mathbf{1}\mathbb{X}_k$). The rotation/reflection components, $\boldsymbol{Q}_k$, can be solved iteratively by minimizing the error function with respect to $\mathbb{X}_k$ and keeping all other $\mathbb{X}_l, l \neq k$ constant for each k in turn. After the convergence of iterative solutions for $\boldsymbol{Q}_k$, the scaling components, s_k, can be solved for analytically.

Using Generalized Procrustes Analysis (GPA), we can obtain estimates for K mapping functions ρ_k depicted in 4.1 when $K > 2$. Given K dissimilarity matrices $\Delta_k, k = 1, \ldots, K$, we would compute separate MDS embeddings of $\{\Delta_k\}$ followed by GPA of all the embeddings. The separate embeddings mapped via $\{\tau_k\}$ would give us a single commensurate representation in which the disparate dissimilarities can be compared. New dissimilarities $\{\boldsymbol{D}_k\}$ can be OOS embedded and mapped by the same Procrustean transformations $\{\tau_k\}$ to the commensurate space. We will use this approach for the match

testing test presented in chapter 4, when $K > 2$. Simulation results are presented in 10.3.

Chapter 8

Canonical Correlation Analysis for Data Fusion

8.1 Canonical Correlational Analysis on Multidimensional Scaling embeddings

Canonical Correlational Analysis is another method for addressing the incommensurability of dissimilarities from different conditions. We will refer to the match detection task in chapter 4 and the data settings in section 6.1 to explain this alternative approach.

For the CCA approach, MDS is used to compute embedding configurations, $\mathbb{X}_1$ and $\mathbb{X}_2$ from the disparate dissimilarity matrices Δ_1 and Δ_2. For the data settings in section 6.1, it is desirable to perform the embedding into the highest-possible dimensional

space ($\mathbb{R}^s$, where $s = p + q$ for the Gaussian and Dirichlet settings) to preserve as many of the signal dimensions as possible (at the risk of possibly including some noise dimensions). CCA [38], then, yields two mappings $\mathcal{U}_1$ and $\mathcal{U}_2$ that map these embeddings in $\mathbb{R}^s$ to the low-dimensional commensurate space ($\mathbb{R}^d$).

While embedding the dissimilarities in the highest dimension possible is a good idea for preserving the signal dimensions, in the presence of noise dimensions (6.1.3), the noise will be incorporated into the embeddings. Even if the dissimilarities are errorless representations of measurements of a particular dimension d^*, for the sake of inference, it is preferable to embed at a lower dimension $d < d^*$ because of the bias-variance tradeoff. We call this variant of the CCA approach regularized CCA. Regularized CCA, for which the embedding dimension choice is s such that $d < s < (p + q)$, is expected to yield a better performance than the CCA approach by introducing more bias for the sake of removing variance. We expect to see a difference between CCA and regularized CCA in our data settings because we introduce noise into the dissimilarities 6.

8.2 Canonical Correlational Analysis

Let X and Y be two s-dimensional random vectors. If we want to find the pair of linear projection operators $U_1 : \mathbb{R}^s \to \mathbb{R}$, $U_2 : \mathbb{R}^s \to \mathbb{R}$ that maximize the correlation between the projections of X and Y, CCA provides the solution to this problem by

optimizing the objective function

$$(\hat{(u)}_1, \hat{(u)}_2) = \arg \max_{u_1 \in \mathbb{R}^s, u_2 \in \mathbb{R}^s} \frac{E[u_1^T X Y^T u_2]}{E[u_1^T X X^T u_1] E[u_2^T Y Y^T u_2]}$$

with the constraints $E[u_1^T X X^T u_1] = 1, E[u_2^T Y Y^T u_2] = 1$ for removing ambiguities. The constraints simplify the objective function to

$$\max_{u_1 \in \mathbb{R}^s, u_2 \in \mathbb{R}^s} E[u_1^T X Y^T u_2].$$

Then, the projection operators are $U_1(x) = (\hat{u}_1)^T x$ and $U_1(y) = (\hat{u}_2)^T y$.

Remark X and Y can be of different dimensions s_1 and s_2, but for the sake of simplicity, we will assume they have the same dimension s.

In general, the projections map to a pair of d-dimensional linear subspaces, that is, $U_1 : \mathbb{R}^s \to \mathbb{R}^d$, $U_2 : \mathbb{R}^s \to \mathbb{R}^d$. The projection matrices that represent the mappings are $\mathcal{U}_1$ and $\mathcal{U}_2$, and their rows are the direction vectors $u_{1(i)}, u_{2(i)}, i = 1, \ldots, d$. These additional pairs of projection vectors can be computed sequentially, with the constraints that the projections along the new directions are uncorrelated with the projections along previous directions. That is, the i^{th} pair of directions that maximize correlation is computed by

$$\hat{(u)}_{1(i)}, \hat{(u)}_{2(i)} = \arg \max_{u_{1(i)}, u_{2(i)} \in \mathbb{R}^s} E[u_{1(i)}^T X Y^T u_{2(i)}]$$

subject to constraints $E[u_{1(i)}^T X X^T u_{1(i)}] = 1$, $E[u_{2(i)}^T Y Y^T u_{2(i)}] = 1$, $E[u_{1(i)}^T X X^T u_{1(j)}] = 0$, $E[u_{2(i)}^T Y Y^T u_{2(j)}] = 0 \; \forall \quad j = 1, \ldots, i-1$. These directions are called "canonical" directions. The projections $\mathbb{X}\hat{(u)}_{1(i)}, i = 1, \ldots, d$ are called "canonical" variates.

For sample CCA, $E[XX^T]$,$E[YY^T]$ and $E[XY^T]$ are replaced with their sample estimates.

Note that s, the dimension of X and Y, is the embedding dimension s we use in the CCA approach. So we use MDS to separately embed the dissimilarities in Δ_1 and Δ_2 in $\mathbb{R}^s$, and then use CCA to project the embeddings to the d-dimensional Euclidean space. CCA$\circ$MDS provides the complete mapping from dissimilarities to the commensurate space.

As in P$\circ$M, new dissimilarities are OOS embedded and mapped to a commensurate space by maps provided by CCA. The test statistic τ, which is the distance between the points in the commensurate space that correspond to the OOS dissimilarities, can now be computed, and the null hypothesis is rejected for "large" values of the test statistic τ, as in Section 7.

8.3 Geometric Interpretation of Canonical Correlational Analysis

To complement CCA, we should also consider Canonical Variate Analysis (CVA). In CVA, the projections are also maximally correlated; however, one is concerned with the variates $a_i = u_{1(i)}^T X$ and $b_i = u_{2(i)}^T Y$, in contrast to the canonical directions $u_{1(i)}$ and $u_{2(i)}$. CVA is to CCA what Principal Coordinate Analysis is to Principal Component Analysis.

We should also define canonical angles as follows:

Definition 2. *For two subspaces $\mathcal{V}$ and $\mathcal{W}$ of $\mathbb{R}^d$, the first* canonical *(or* principal*) angle between them is* $\arccos \max_{v\in\mathcal{V},w\in\mathcal{W}} \frac{<v,w>}{\|v\|\|w\|}$. *Other ($i^{th}$) canonical angles are defined as* $\arccos \max_{v_i\in\mathcal{V},w_i\in\mathcal{W},v_i\perp v_j,w_i\perp w_j,\forall j<i} \frac{<v_i,w_i>}{\|v_i\|\|w_i\|}$. *The vectors v_i and w_i that maximize* $\frac{<v_i,w_i>}{\|v_i\|\|w_i\|}$ *are called* canonical *vectors.*

For two $n \times s$ configuration matrices $\mathbb{X}$ and $\mathbb{Y}$, consider the column spaces of the two matrices $\mathcal{L}\mathbb{X} = \{\mathbb{X}u : u \in \mathbb{R}^s\}$ and $\mathcal{L}\mathbb{Y}$. Note that these spaces are subspaces of $\mathbb{R}^n$, not of $\mathbb{R}^s$ (the n points of the configuration lie in $\mathbb{R}^s$). We already know that CCA/CVA maximizes the correlation of the variates. Let us we borrow terminology from pattern recognition and call any one-dimensional subspace of $\mathcal{L}\mathbb{X}$ (and $\mathcal{L}\mathbb{Y}$) a "feature". Each $u \in \mathbb{R}^s$ define a feature. Therefore, any linear combination of the the original feature vectors (rows of $\mathbb{X}$ and $\mathbb{Y}$) is also a feature. The (sample) correlation of the variates (defined by the canonical directions $u_{1(i)}$ and $u_{2(i)}$) in CCA is also the cosine of the angle between the features defined by the same directions. The uncorrelatedness condition of two canonical variates of $\mathbb{X}$ correspond to the perpendicularity of the corresponding feature vectors of $\mathbb{X}$. Thus, CCA/CVA for d variates solves the problem of finding the first d canonical angles and the corresponding canonical directions of $\mathcal{L}\mathbb{X}$ and $\mathcal{L}\mathbb{Y}$.

8.4 Relationship between Canonical Correlational Analysis and Commensurability

Theorem 3. *Let $\mathbb{X}_1$ and $\mathbb{X}_2$ be two $n \times s$ (configuration) matrices that represent pairs of points that are perfectly "matched" (there exists a matrix $\mathbf{Q}$ such that $\|\mathbb{X}_1\mathbf{Q} - \mathbb{X}_2\| = 0$). Suppose, for the joint embedding procedure, that the embedded configurations are constrained to be of the form $\widetilde{\mathbb{X}_1} = \mathbb{X}_1 U_1$ and $\widetilde{\mathbb{X}_2} = \mathbb{X}_2 U_2$ for some $U_1 \in \mathcal{U}$ and $U_2 \in \mathcal{U}$, where $\mathcal{U}$ be the set of all orthogonal d-frames (ordered set of d linearly independent vectors) of $\mathbb{R}^s$. Elements of $\mathcal{U}$ correspond to the unique projection operators to d-dimensional linear subspaces of $\mathbb{R}^s$. The commensurability error is defined as it is in equation* (5.1).

Canonical Correlational Analysis on the i.i.d. *sample of points represented by $\mathbb{X}_1$ and $\mathbb{X}_2$ gives $\mathbf{U}_1 \in \mathcal{U}$ and $\mathbf{U}_2 \in \mathcal{U}$, the two elements of $\mathcal{U}$ that maximize commensurability, subject to $U_1^T \mathbb{X}_1^T \mathbb{X}_1 U_1 = I_d$, and $U_2^T \mathbb{X}_2^T \mathbb{X}_2 U_2 = I_d$ (I_d is the $d \times d$ identity matrix).*

Proof. Consider 5.1 and its simplified form when $\delta_{ij} = 0$, as we assumed that there exists a perfect matching.

$$\epsilon_{c_{(k_1=1,k_2=2)}} = \frac{1}{n} \sum_{1 \le i \le n; k_1=1, k_2=2} (d(U_1 \boldsymbol{x}_{ik_1}, U_2 \boldsymbol{x}_{ik_2}))^2. \tag{8.1}$$

Equation (8.1) can be written as

$$\begin{aligned}\epsilon_{c12} &= \frac{1}{n}\sum_{j=1}^{d}\sum_{i=1}^{n}\left[((u_{j1}\boldsymbol{x}_{i1} - u_{j2}\boldsymbol{x}_{i2}))\right]^2 \\ &= \frac{1}{n}\sum_{j=1}^{d}\sum_{i=1}^{n}(u_{j1}\,\boldsymbol{x}_{i1})^2 + (u_{j2}\,\boldsymbol{x}_{i2})^2 - 2(u_{j1}\boldsymbol{x}_{i1}u_{j2}\boldsymbol{x}_{i2}),\end{aligned}$$

where u_{j1} and u_{j2} are the rows of U_1 and U_2.

Because $U_1^T\mathbb{X}_1^T\mathbb{X}_1U_1 = I_d$ (and $U_2^T\mathbb{X}_2^T\mathbb{X}_2U_2 = I_d$), for any j, $u_{j1}\mathbb{X}_1^T\mathbb{X}_1u_{j1}^T = 1$ ($u_{j2}\mathbb{X}_2^T\mathbb{X}_2u_{j2}^T = 1$).

Consider the sum of the first terms, $S_1 = \frac{1}{n}\sum_{i=1}^{n}\sum_{j=1}^{d}(u_{j1}\boldsymbol{x}_{i1})^2$. It is easier to show this sum is constant, if we use the the probabilistic definition of CCA in section 8.2 . Assume $\mathbb{X}_1$ and $\mathbb{X}_2$ represent n-sized sample of X and Y, respectively. As $n \to \infty$, the sum $S_1 = \sum_{j=1}^{d}\frac{1}{n}\sum_{i=1}^{n}(u_{j1}\boldsymbol{x}_{i1})^2$ converges to $\sum_{j=1}^{d}E[u_{j1}X]$. Each term of this limit sum is constrained to be 1 in the definition of CCA 8.2. Therefore the sum S_1 converges to d. By the same line of reasoning, we can conclude the sum of the second terms is also constant. The sum of the third terms can be written in the form of $(-2 \times \xi)$, where ξ is the sum of the products of dot products $u_{j1}\boldsymbol{x}_{i1}$ and $u_{j2}\boldsymbol{x}_{i2}$. Thus, maximizing ξ under the linearity constraints is maximizing the commensurability.

Note that

$$\xi = \frac{1}{n}\sum_{i=1}^{n}\sum_{j=1}^{d}(u_{j1}\boldsymbol{x}_{i1}u_{j2}\boldsymbol{x}_{i2}) = \frac{1}{n}\left[\text{trace}U_1^T\mathbb{X}_1^T\mathbb{X}_2U_2 - \sum_{1\le j\le d}\sum_{1\le i<l\le d}u_{j1}\boldsymbol{x}_{i1}\boldsymbol{x}_{l2}^Tu_{j2}^T\right]$$

where the dot products $u_{j1}^T\boldsymbol{x}_{i1}$ and $\boldsymbol{x}_{l2}^Tu_{j2}$ are uncorrelated because $\boldsymbol{x}_{i1}, \boldsymbol{x}_{l2}, i \neq l$ are independent samples. Therefore, as $n \to \infty$, the second sum vanishes, and only the trace term remains. Thus, $\xi = \frac{1}{n}\text{trace}U_1^T\mathbb{X}_1^T\mathbb{X}_2U_2$, which is the objective function optimized

with respect to U_1 and U_2 in the canonical correlational analysis. Subject to constraints, CCA maximizes commensurability with respect to U_1 and U_2.

□

8.5 Spectral Embedding Generalization of CCA

Another way to view the connection between CCA and JOFC embedding (using classical MDS) is via connections to spectral embedding. Jagarlamudi et al. [39] show that CCA is a special case of Spectral Embedding with the restriction that the joint embedding coordinates are linear projections of the original multiview data, $\mathbb{X}_1$ and $\mathbb{X}_2$. First, we define "Spectral Embedding" as follows: Given a $k \times k$ weight matrix W, Spectral Embedding embeds k points in d-dimensional Euclidean space by minimizing the cost function $\sum_{i,j\in\{1,\ldots,k\}} W_{ij} \left(u_i - u_j\right)^2$, where $u_i, u_j \in \mathbb{R}^d$ are the embedded coordinates.

Assume that CCA is applied to $\mathbb{X}_1$ and $\mathbb{X}_2$, which yields two $n \times d$ matrices, $\widetilde{\mathbb{X}_1}$ and $\widetilde{\mathbb{X}_2}$, the embedded configuration matrices.

For the same multiview data, $\mathbb{X}_1$ and $\mathbb{X}_2$, let

$$Z = \begin{bmatrix} \mathbb{X}_1 & \mathbf{0} \\ \mathbf{0} & \mathbb{X}_2 \end{bmatrix}.$$

Let $W = \begin{bmatrix} \mathbf{0} & I_n \\ I_n & \mathbf{0} \end{bmatrix}$ be a $2n \times 2n$ weight matrix. We can assume that W is an adjacency

matrix that represents a graph that is bipartite, and the only edges lie between the i^{th} and $(n+i)^{th}$ vertices (which correspond to matched pairs in earlier chapters) for $i \in \{1, \ldots, n\}$. The degree matrix for this graph is then $D = I_{2n}$. The graph laplacian is $L = D - W$. Assume the constraint that the embedding coordinates of the i^{th} point $\widetilde{Z}_i = p^T Z_i$ are introduced for some $p \in \mathbb{R}^d$, i.e., p is a projection vector. We call this constraint the linearity condition. Then, the embedding of the i^{th} point of the $2n$ points via Spectral Embedding for the weighted adjacency matrix L is $\widetilde{Z}_i$, where $\widetilde{Z}_i = \left[\widetilde{X_{1i}}\ \mathbf{0}\right]$ or $\widetilde{Z}_i = \left[\mathbf{0}\ \widetilde{X_{2i}}\right]$ and $\widetilde{X_{1i}}$ and $\widetilde{X_{2i}}$ are the i^{th} rows of $\widetilde{\mathbb{X}_1}$ and $\widetilde{\mathbb{X}_2}$ yielded by CCA. As the authors note in [39], from W, we can take intra-view similarities into account (which means preserving more fidelity) and choose the diagonal block matrices in W to be nonzero. This would be akin to a JOFC-type embedding because the commensurability criterion is accounted for by using an identity matrix as the off-diagonal block matrix in W, and the fidelity criterion is accounted for by the nonzero diagonal block matrices in W.

As mentioned in 3.2.4, the joint embedding of a dissimilarity matrix via cMDS is equivalent to spectral embedding under certain conditions. Consider to the spectral embedding generalization of CCA we have just presented, using the multiview data Z and weight matrix W obeying the linearity condition. There is an equivalent classical MDS embedding with an omnibus dissimilarity matrix M for which $\tau(M) = -\frac{1}{2}JMJ^T$ corresponds to the pseudo-inverse of $L = D - W$.

8.6 Generalized CCA: $K > 2$

Whereas CCA is defined for $K = 2$ conditions, multiple generalizations are available because the correlation between more than two configurations can be defined in multiple ways [40]. Let $X_1, \ldots, X_K$ be random vectors for which a generalized CCA representation will be computed. Consider the first set of canonical variates to be computed, $Z_1^{(1)}, \ldots, Z_K^{(1)}$. Denote the correlation matrix of $Z_1^{(1)}, \ldots, Z_K^{(1)}$ by $\Phi^{(1)}$. The following three criteria are proposed in [40]:

- SUMCOR. Maximize the sum of the elements of $\Phi^{(1)}$: $\mathbf{1}^T(\Phi^{(1)})\mathbf{1}$,
- MAXVAR. Maximize the largest eigenvalue of $\Phi^{(1)}$: $\lambda_1^{(1)}$,
- MINVAR. Minimize the smallest eigenvalue of $\Phi^{(1)}$: $\lambda_1^{(m)}$.

One can interpret all of these criteria as different norms on the correlation matrix. An interesting question that will not be addressed here is whether one of these generalizations is more appropriate for H_{A1} or H_{A2}. We chose to use H_{A1} as the alternative hypothesis and the SUMCOR criterion as the generalization of CCA.

Chapter 9

Multiple Minima in Multidimensional Scaling

We previously considered configurations embedded via optimization of the MDS criterion functions, but we have not mentioned the difficulties that might rise in optimization, such as the lack of convergence or the existence of multiple local minima. Because raw stress minimization is solved using the iterative majorization algorithm, the MDS embedding method is prone to these global optimization problems. Their severity depends on the value of the original dissimilarities. We are particularly interested in the multiple local minima problem, as the configuration yielded by the optimization of MDS criterion might be a local minimum instead of the global minimum. In fact, a unique global minimum of the MDS criterion might not exist. In [41], the multiple local minima problem is discussed for raw stress. A simple example is constructed, which is

shown to have multiple local minima, one of which is the global minimum.

We focus our attention not on providing evidence of the existence of multiple local minima in our matched data settings but on investigating how multiple local minima might be related to w, when they do exist. We know the weighted raw stress function is continuous with respect to w. As w changes, the weighted raw stress function will change continuously, and the value of w might have an effect the local minimum in which the iterative algorithm[1] might terminate. As a result of a change in w, another local minimum can become the global minimum. In the latter case, the $\arg\min$ of the MDS criterion function jumps from one local minimum (a particular configuration) to another minimum/configuration with an infinitesimal change in w. In this case, the embedded configuration, $\mathbb{X}$, viewed as a matrix-valued function of w, has a point discontinuity. Because $AUC(w) = auc(\mathbb{X}(w))$ is a function of the embedded configuration, the discontinuity of the configuration $\mathbb{X}(w)$ at $w = w_d$ might also cause $AUC(w)$ to be discontinuous at that $w = w_d$. To investigate these issues, following the approach of [41], we design a simple example that is as informative as it is instructive.

[1]Because we are using raw-stress embedding, we are using iterative majorization to find the MDS solution.

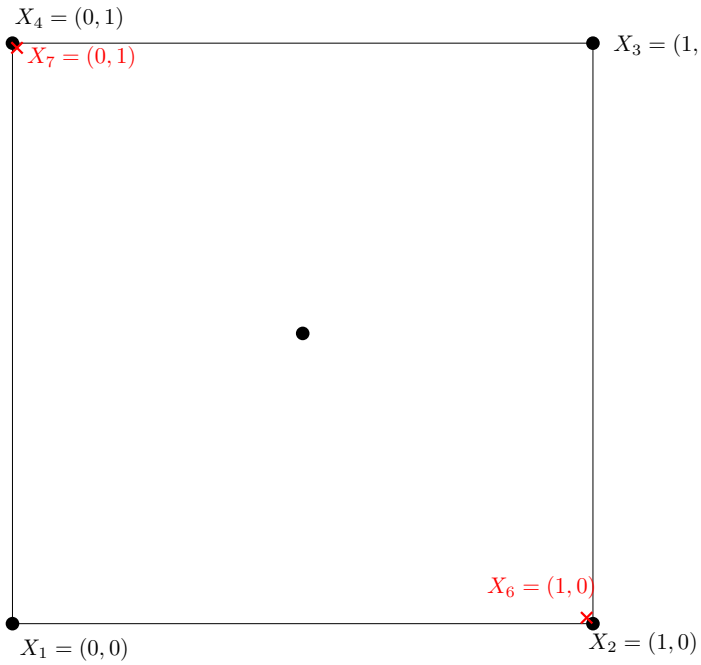


Figure 9.1: True configuration of X_i, $i \in 1, \ldots, 7$

9.1 Discontinuity in weighted raw stress OOS configurations

It is possible to construct an example in which the weight parameter w controls which of the local minima is the global minimum among the configurations of $\hat{X}_\cdot$.

Consider five in-sample points in $\mathbb{R}^2$ with locations $X_1 = (0, 0)$, $X_2 = (1, 0)$, $X_3 = (1, 1)$, $X_4 = (1, 0)$, and $X_5 = (.5, .5)$ and two OOS points with coordinates $X_6 = (1, 0)$ and $X_7 = (0, 1)$. We assume that X_6 is matched with X_2 and that X_7 is matched with X_4. Therefore, the weights for the dissimilarities between X_6 and X_2 (also X_7 and X_4) are w. The weights of other dissimilarities are $1 - w$.

Denote the Euclidean distance matrix by D. Suppose, due to noise, or due to dissimilarities corresponding to a non-Euclidean distance, the dissimilarity matrix is

$$D'_{ij} = \begin{cases} D_{ij} - 1.4 & \text{if } (i,j) \in \{(4,6),(6,4),(2,7),(7,2)\} \\ D_{ij} & \text{otherwise} \end{cases} .$$

The approximate values of the dissimilarity matrix is shown in Table 9.1.

	1	2	3	4	5	6	7
1	0.00	1.00	1.41	1.00	0.71	1.00	1.00
2	1.00	0.00	1.00	1.41	0.71	0.00	0.01
3	1.41	1.00	0.00	1.00	0.71	1.00	1.00
4	1.00	1.41	1.00	0.00	0.71	0.01	0.00
5	0.71	0.71	0.71	0.71	0.00	0.71	0.71
6	1.00	0.00	1.00	0.01	0.71	0.00	1.41
7	1.00	0.01	1.00	0.00	0.71	1.41	0.00

Table 9.1: The entries of the dissimilarity matrix (rounded to two decimal digits)

Remark This data setting does not exactly fit the data setting that we use for the match detection task because one of each matched pair is an in-sample point and there are not any multiple conditions. As we have noted, our aim is to just set up a simple setting with the weighted raw stress criterion function that still demonstrates interesting behavior.

The MDS criterion function is optimized with the iterative majorization algorithm

starting with different initial configurations. Depending on the initial configuration, the final embedding coordinates of $\hat{X}_6$ might be closer to X_4 than to X_2 because the iterative majorization algorithm terminates in that local minimum in the configuration space. Because $D_{46} \approx 0$ and $D_{27} \approx 0$, the configurations that place $\hat{X}_6$ and X_4 together (and $\hat{X}_7$ and X_2 together) would have raw stress close to the original configuration that has $\hat{X}_6$ and X_2 together (and $\hat{X}_7$ and X_4 together). We therefore have at least two local minima for this data setting.

We distinguish between these two types of local minima, one in which the embedded OOS points $\hat{X}_6$ and $\hat{X}_7$ end up on the same side as their respective matched points X_2 and X_4 (named "true" or real config.) and the other in which they end up on the sides opposite to their matched points (named "alternative" local min.).

For initial configurations in which the initial coordinates of $\hat{X}_6$ lie on the X_4 side of the $y = x$ line in $\mathbb{R}^2$ and $\hat{X}_7$ is on the X_2 side, the iterative majorization might terminate in an "alternative" local minimum. Assume that we start from such an initial configuration, interpreting the steps of iterative majorization as points moving toward low-stress configurations, $\hat{X}_6$ has to cross paths with the embeddings of X_1, X_3, X_5. However, embedding $\hat{X}_6$ close to these points would result in a high-stress configuration because it has nonzero dissimilarities with those points. The same argument can be made for $\hat{X}_7$. To qualitatively describe the situation, the three points X_1, X_5, and X_3 form a barrier that the OOS points need to cross to reach their matched counterparts.

The "alternative" local minima correspond to the case in which the OOS points are

unable to cross the "barrier". Other configurations such as those in which $\hat{X}_6$ and $\hat{X}_7$ are on the same side of $y = x$ line are not local minima because the original dissimilarity between X_6 and X_7) is large ($\sqrt{2}$) compared with dissimilarities between other pairs of points (the dissimilarity values are 0, 1, and $\frac{\sqrt{2}}{2}$) and because embedding them close would increase the raw stress significantly.

Whether it is easier or harder to get out of the "alternative" local minimum is based on the value of w. In addition, depending on w, these "alternative" configurations can have a lower stress than the "true" (real) configuration and result in a global minimum. That is, if w is small enough, the configuration in which $\hat{X}_6$ stays on the side of X_4 instead of that of X_2 might have a lower stress than the configuration in which $\hat{X}_6$ is near its matched point X_2 because the contribution of $D_{ij} - d(X_i, X_j)$ to the raw stress where $(i, j) = (4, 6)$ is multiplied by $1 - w$, whereas every other dissimilarity is multiplied by w.

For our simulation, we chose a grid of starting points for X_6 with the x-coordinates of $X_{6x} \in \{-0.5, 0.4, \ldots, 1.5\}$ and the y-coordinates $X_{6y} \in \{-0.2, -0.1, 0, \ldots, 1.6\}$. For X_7, the corresponding starting points were $(1 - X_{6x}, 1 - X_{6y})$. We embedded the pair of OOS points by minimizing raw stress with the IM algorithm starting from a pair of corresponding points from the grid.

Starting from a small enough value for w and increasing it until w is arbitrarily close to 1, there are two w values for which important changes in embedding configurations occur and final stress values are obtained.

The plots in Figures 9.3, 9.4, 9.5, 9.6, and 9.7 show the embedding configurations of X_6 (in red circles) and X_7 (in blue pluses). Each point in the plots is a point at which the IM algorithm terminates after starting from different initial configurations (one red and one blue point for each initial configuration). The point pairs plotted in the left box for each figure pair are those configurations in which X_6 and X_7 end up on the side of their matched points ("true" final configuration). The configurations on the right are those in which the points end up on the opposite side of their matched points. For the last four w values, $(0.82, 0.83, 0.84, 0.85)$, there are no initial configurations that end up in "alternative"-type configurations. Both types of configurations ("true" and "alternative") for selected w values are plotted in Figure 9.2.

We are also interested in which type of local minima has lower stress and contains the global minimum solution. We compute the minimum stress value among each type of local minima and compare these final stress values. The final stress values of the final configurations listed in Table 9.2 and plotted in figure 9.8 indicate that around $w = 0.5$, the "true" local minimum, begins to have a lower stress value compared with the "alternative" local minimum. This is the first w value that corresponds to an important change. This transition provides evidence that different local minima might become global minima, depending on the value of w.

It is also noteworthy that starting around $w = 0.8$ in Figure 9.5, all of the $\hat{X}_6$ and $\hat{X}_7$ pairs are on the verge of passing through the barrier and ending up on the side of their matched points because the barrier starts to become negligible and there are no separate

local minima. When $w > 0.8$, all of the point pairs end up in the "real" configuration 9.6. This is the other w value at which important changes in configurations and stress values occur. Further increasing w changes the final stress value, and the final embedding configuration moves closer to the original locations of X_i in 9.1 9.7.

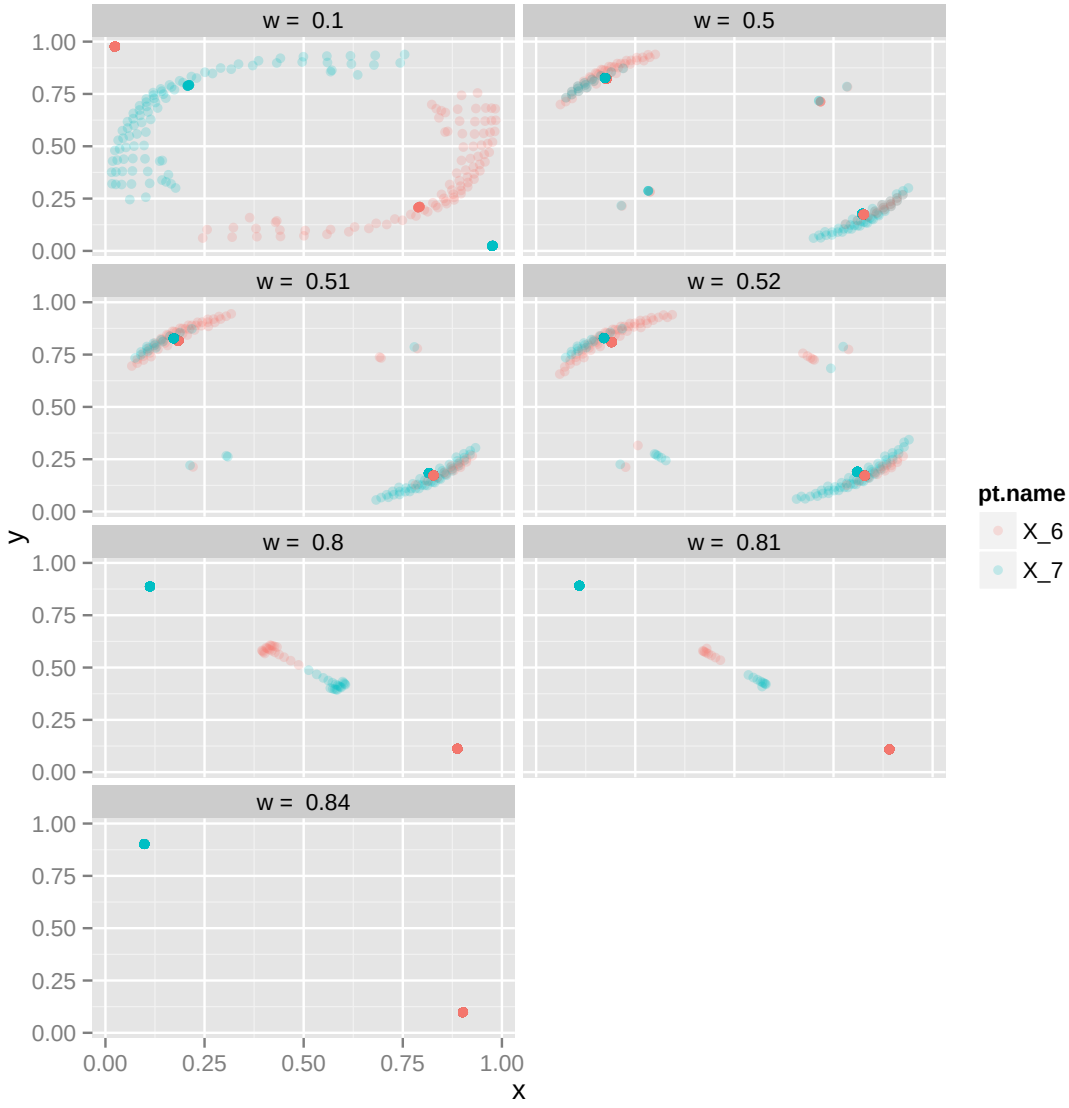


Figure 9.2: Embedded Point Pairs ($\hat{X}_6$ and $\hat{X}_7$) for all initial configurations for different w values

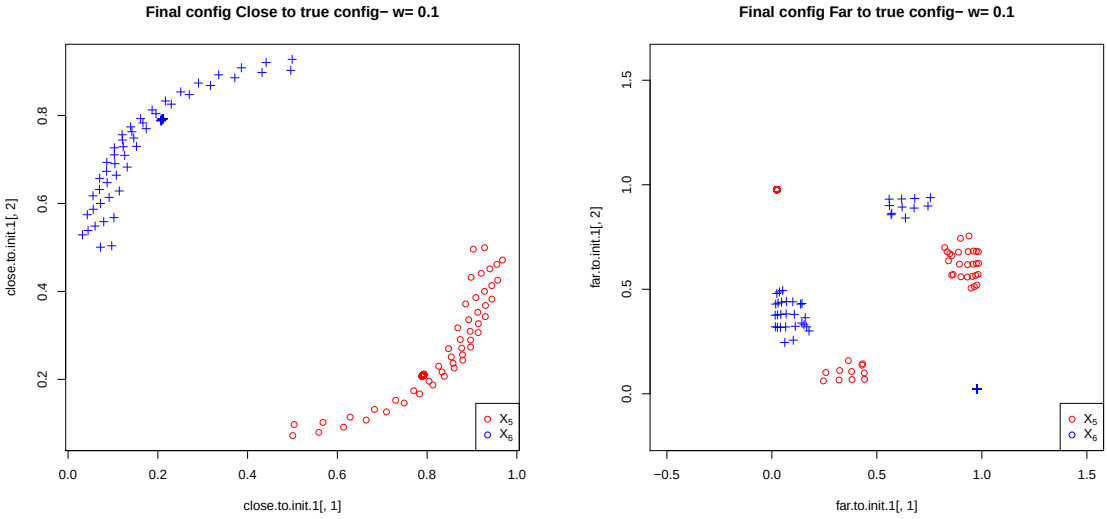


Figure 9.3: Final configurations for different initial configurations, $w = 0.1$

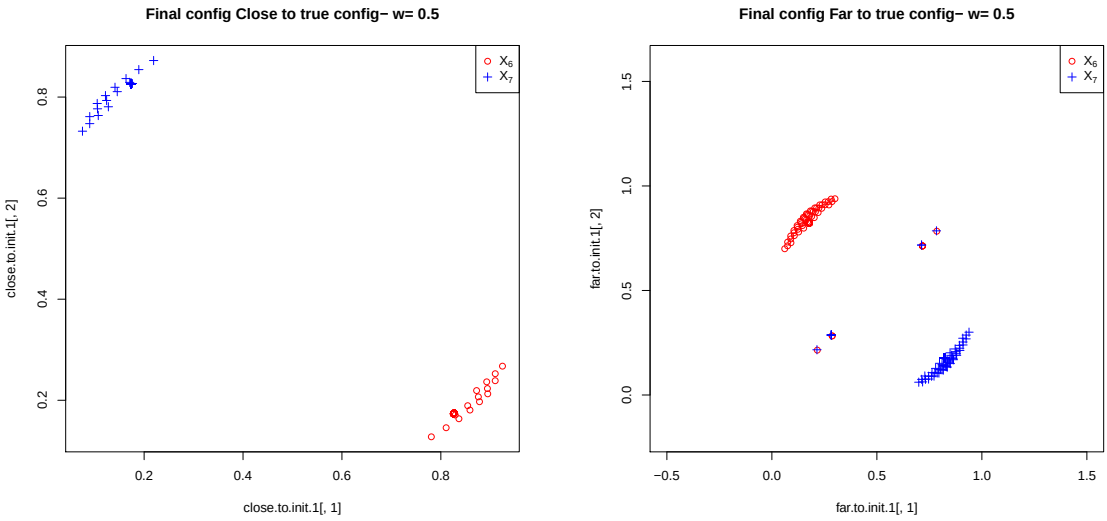


Figure 9.4: Final configurations for different initial configurations, $w = 0.5$

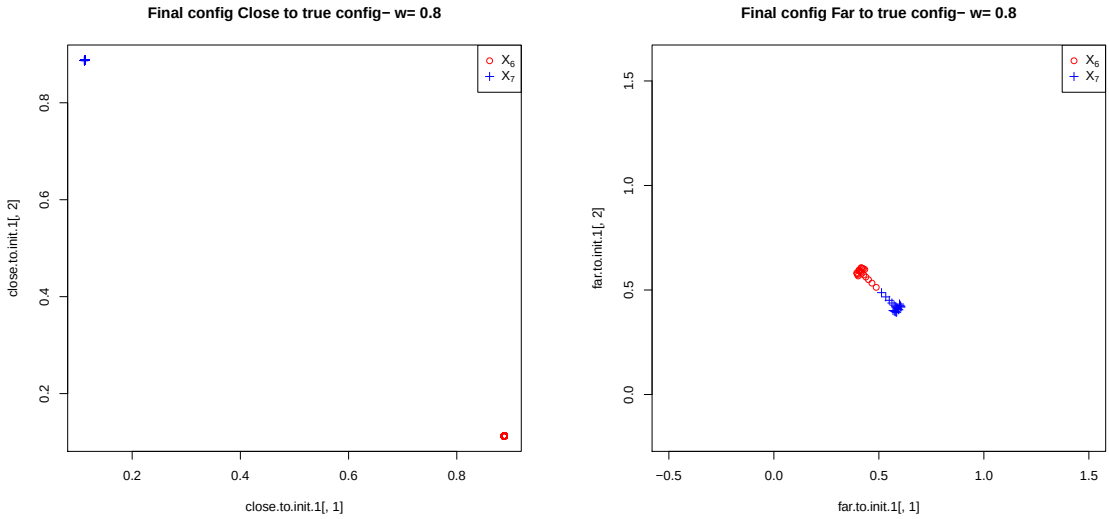


Figure 9.5: Final configurations for different initial configurations, $w = 0.8$

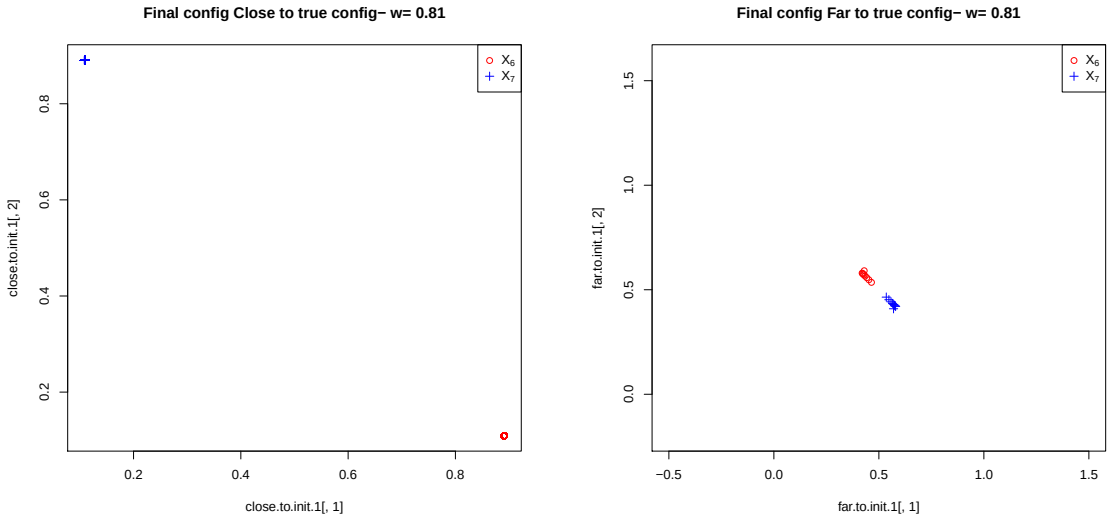


Figure 9.6: Final configurations for different initial configurations, $w = 0.81$

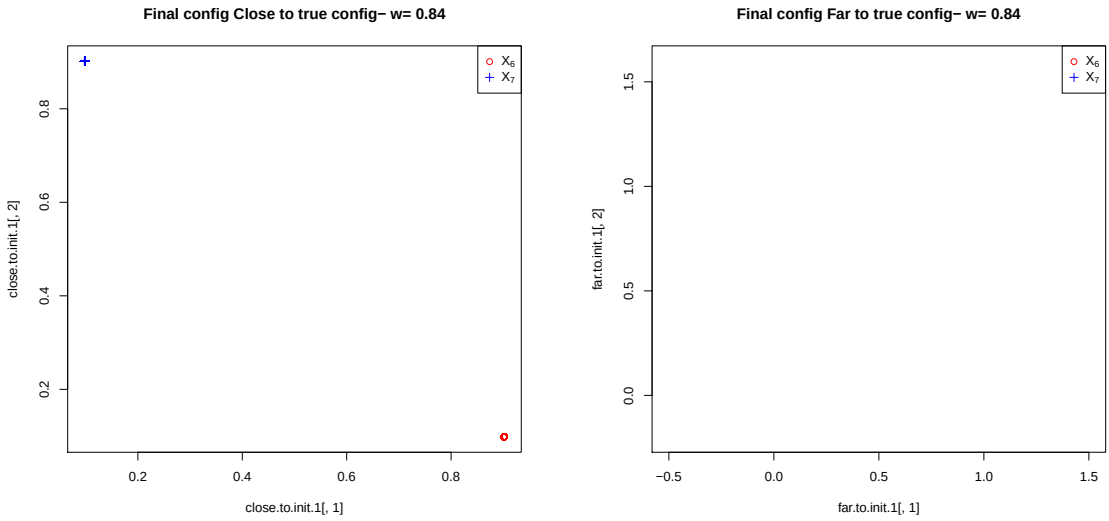


Figure 9.7: Final configurations for different initial configurations, $w = 0.84$

w	0.1	0.2	0.3	0.4	0.41	0.42	0.43	0.44	0.45	0.46	0.47
Local min for real config.	2.80	2.51	2.22	1.92	1.89	1.86	1.83	1.80	1.77	1.74	1.71
Alternative local min	0.39	0.76	1.10	1.40	1.43	1.46	1.48	1.51	1.53	1.56	1.58
w	0.48	0.49	0.5	0.51	0.52	0.53	0.54	0.55	0.6	0.65	0.7
Local min for real config.	1.68	1.65	1.62	1.59	1.56	1.53	1.50	1.47	1.32	1.17	1.01
Alternative local min	1.60	1.63	1.65	1.67	1.69	1.71	1.73	1.74	1.81	1.82	1.81
w	0.75	0.76	0.77	0.78	0.79	0.8	0.81	0.82	0.83	0.84	0.85
Local min for real config.	0.86	0.82	0.79	0.76	0.73	0.70	0.66	0.63	0.60	0.57	0.53
Alternative local min	1.79	1.77	1.75	1.72	1.69	1.66	1.64	NA	NA	NA	NA

Table 9.2: Final stress values for the two local minima configurations

This example was constructed carefully using a symmetric configuration of points. Under reasonable probability distributions for point configurations, it is unexpected that such a symmetry will appear with nonzero probability. Thus, we conjecture that such discontinuities with respect to w in the embedded configuration have a zero measure. Because the test statistic is a continuous function of the embedded configuration, the events for which the test statistic has discontinuities with respect to w also have a zero measure. This result suggests that the assumption of stochastic continuity of the test statistic that is used to show the continuity of the AUC function in subsection 4.2.1 is a reasonable assumption.

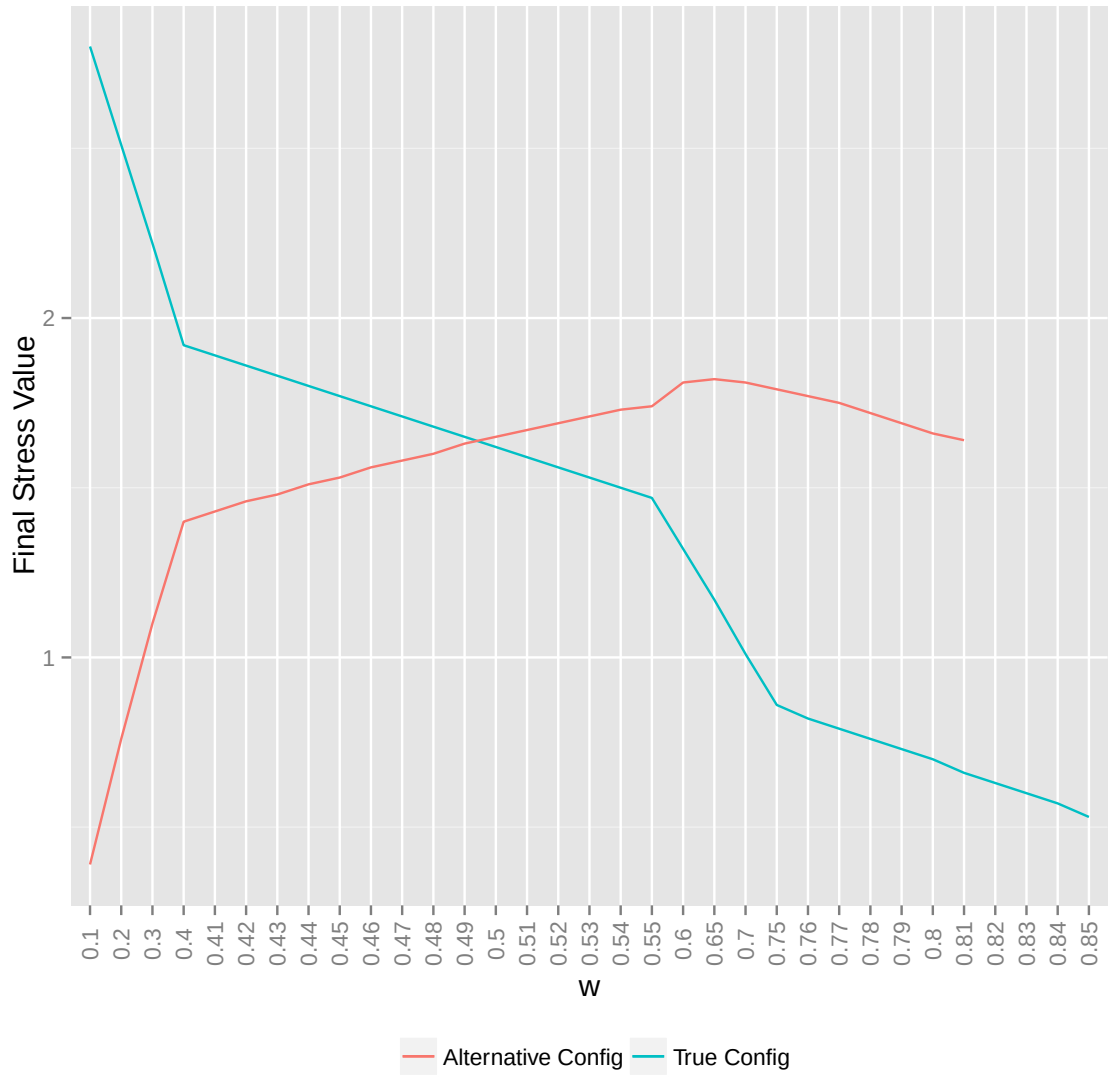


Figure 9.8: Final stress values vs w for the two true and alternative local minima configurations

Chapter 10

Simulations and Experiments

10.1 Simulation Results

To compare the different approaches, training data of matched pairs of measurements were generated according to the Dirichlet and Gaussian settings with parameters p, q, r and c 6. Dissimilarity representations (1.2) were computed from pairwise Euclidean distances of these measurements. A set of matched pairs and unmatched pairs of measurements were also generated for testing using the same distributions. Following the OOS embedding of the test pairs (computed via the P∘M 7, CCA 8, regularized CCA 8.2, or JOFC 4 approach), test statistics for matched and unmatched pairs (corresponding to null and alternative hypothesis, respectively) were used to compute power values at a set of fixed type I error rate α values. By using the same generated data for all of the approaches, we can compare the performance of different approaches using either the

area under the curve (AUC) measure or the statistical power at a desired α value.

Additionally, to consider the relative robustness of the methods, "noisy" measurements were created from the original measurements by concatenating randomly generated independent noise vectors (subsection 6.1.3). This setting will be referred to as the "noisy case". The magnitude of noise is controlled by the parameter c in equation (6.1)). The original setting, with $c = 0$, will be referred to as the "noiseless case". If the magnitude of noise is small enough and the embedding dimension is not larger than the signal dimension, the embeddings provided by PCA and MDS should not be affected significantly by the noise. However, if the number of noise dimensions (controlled by the parameter q in the distribution of E_{ik} as defined in equation (6.1)) is large enough, it is expected that embeddings via CCA will be affected due to spurious correlations between noisy dimensions.

We will now describe the steps of our Monte Carlo simulation in detail. Given the setting ("Gaussian","Dirichlet"), the steps for each Monte Carlo replicate are as follows:

- A training set ($\mathbf{T}_{mc}$), which consists of n pairs of matched measurements, is generated. If $c = 0$, the "noiseless" data setting is being simulated, and the measurements are p-dimensional vectors; otherwise, the "noisy" setting is being used to generate data and measurement vectors that are $(p + q)$-dimensional. $\mathbf{T}_{mc}$ = $\begin{matrix} X_{11} & \dots & X_{1K} \\ \dots & \dots & \dots \\ X_{n1} & \dots & X_{nK} \end{matrix}$ where each X_{ik} is a random vector of dimension $(p+q\times\mathbb{I}(c > 0))$

and the conditional distribution $X_{i.}|\boldsymbol{\alpha}_i$ is specified as an appropriate Multivariate Normal or Dirichlet distribution. The data generation is also described in detail in chapter 6.

- Dissimilarities are computed from X_{ik}, $[\Delta_k]_{ij} = d(X_{ik}, X_{jk})$ for each condition k. We use the Euclidean Distance for both Gaussian and Dirichlet settings.

- Dissimilarities are embedded in Euclidean space via MDS. For the P∘M approach, the embedding falls onto $\mathbb{R}^d$, followed by a transformation from $\mathbb{R}^d$ to $\mathbb{R}^d$. For CCA, the embedding falls onto $\mathbb{R}^{p+q}$, followed by projection onto $\mathbb{R}^d$. For regularized CCA, the embedding falls onto $\mathbb{R}^s$, where $s = (p+q)/2$[1], followed by projection onto $\mathbb{R}^d$. The final embeddings fall onto $\mathbb{R}^d$. We denote this in-sample embedding configuration as $\hat{\mathbf{T}}$. For the JOFC approach, the embedding is performed using the weighted raw stress function $\sigma_W(\cdot) = f_w(D(\cdot), M)$ in equation (7.2) with a common weight w for commensurability terms and another common weight $1-w$ for fidelity terms. We try different values of w in our simulations. For P∘M, CCA and regularized CCA, an unweighted raw stress function ($\sigma(\cdot)$) is used as a criterion function for embedding the dissimilarities.

- m pairs of matched measurements are generated that are treated as OOS, and

  - we compute the dissimilarities between these OOS points and the points in $\mathbf{T}_{mc}$,

[1] s could be chosen as any integer between d and $p+q$. This particular choice was a sensible one for the values of $p, q, andd$ in our simulations.

- we embed the OOS dissimilarities as pairs of embedded points via the OOS extension:

  $(\tilde{y}_1^{(1)}, \tilde{y}_1^{(2)}), \ldots, (\tilde{y}_m^{(1)}, \tilde{y}_m^{(2)})$, and

- we compute the test statistic τ for each pair, $\tau_i = d(\tilde{y}_i^{(1)}, \tilde{y}_i^{(2)});\ i = 1, \ldots, m$

The values of the statistic $\tau = tau_i, i = 1, \ldots, m$ are used to compute the empirical cumulative distribution function under the null hypothesis.

- Identical steps for m pairs of unmatched measurements result in the empirical cumulative distribution function of τ under the alternative hypothesis.

- For any fixed α value, a critical value for the test statistic and the corresponding power is computed.

For $p = 5$, $q = 10$, $d = 2$, and $c \in \{0, 0.01\}$ and for $n = 150$ and $m = 150$, the average of the power values for $nmc = 150$ Monte Carlo replicates are computed at different αs and are plotted in Figure 10.3 against α for the Gaussian setting. The plot in Figure 10.3 shows that for different values of w, the β-α curves vary significantly. The conclusion is that the match detection tests with JOFC embedding using specific w values perform better than other w values in terms of power. In Figure 10.3, $\beta(w)$ is plotted against w for fixed values of α. It is interesting that the optimal value of w seems to be in the range of $(0.85, 1)$ for all settings, which suggests that a significant emphasis on commensurability might be critical for the match detection task.

The value of w that results in the highest AUC measure is different for each Monte

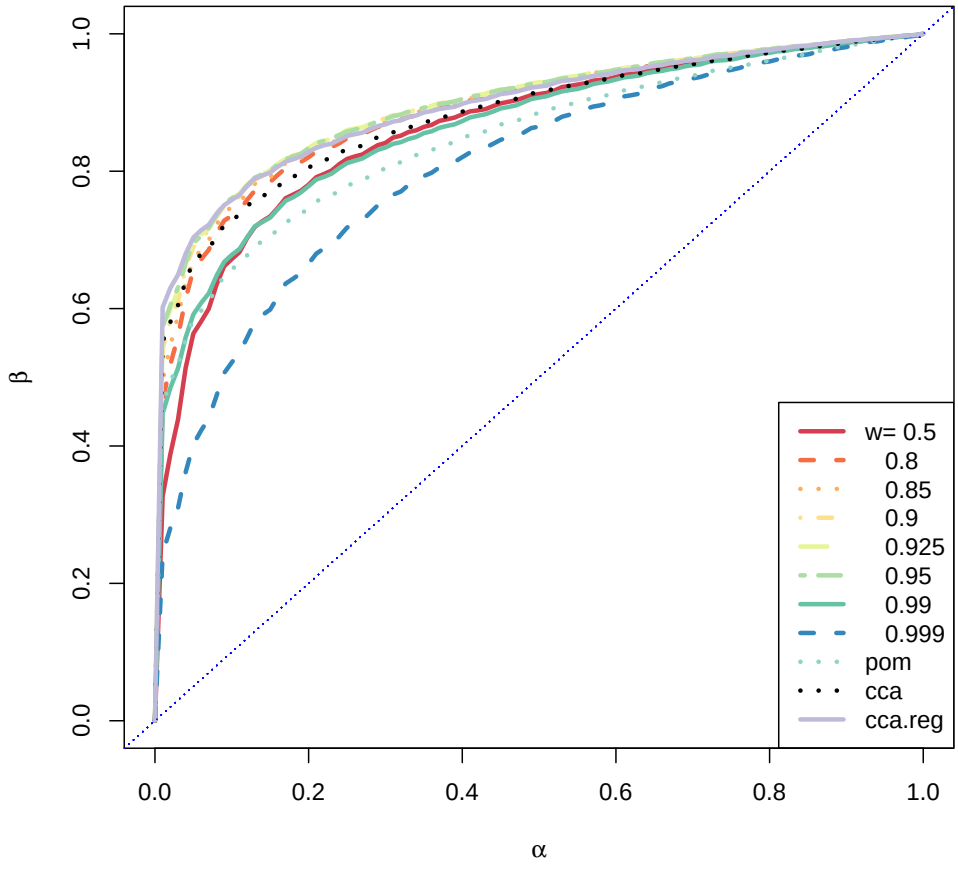


Figure 10.1: Power (β) vs Type I error (α) plot for different w values for the Gaussian setting (noisy case)

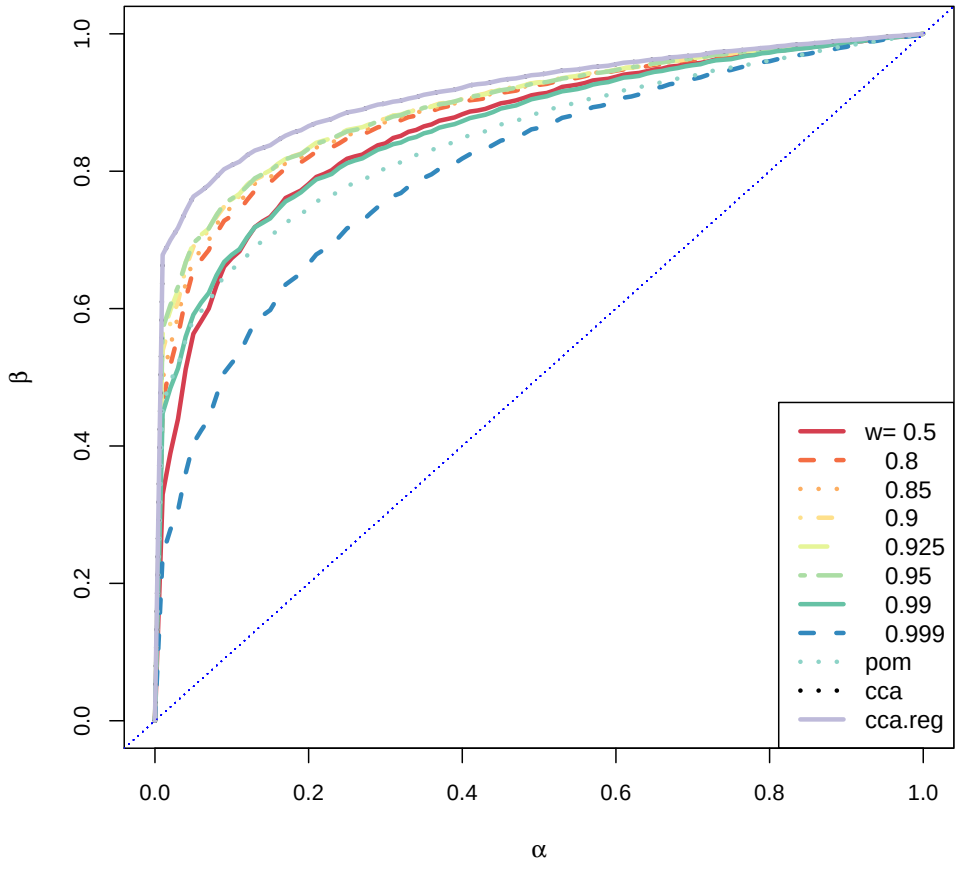


Figure 10.2: Power (β) vs Type I error (α) plot for different w values for the Gaussian setting (noiseless case)

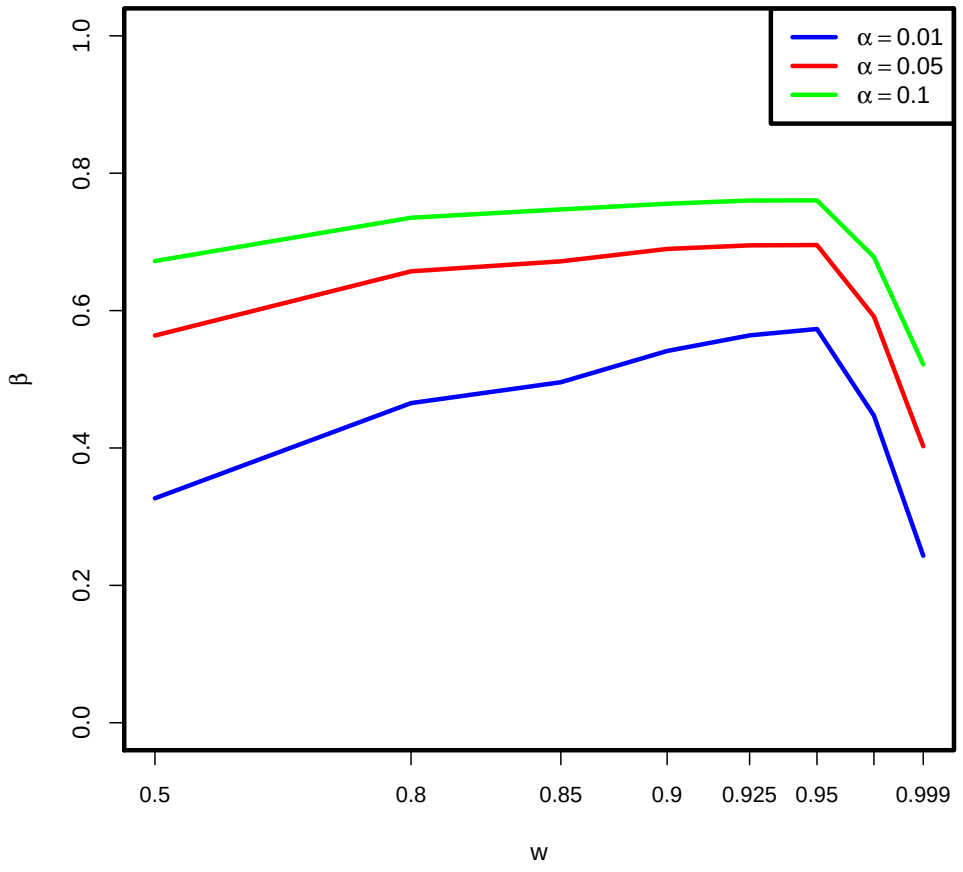


Figure 10.3: Power (β) vs w plot for different Type I error (α) values for the Gaussian setting (noisy case)

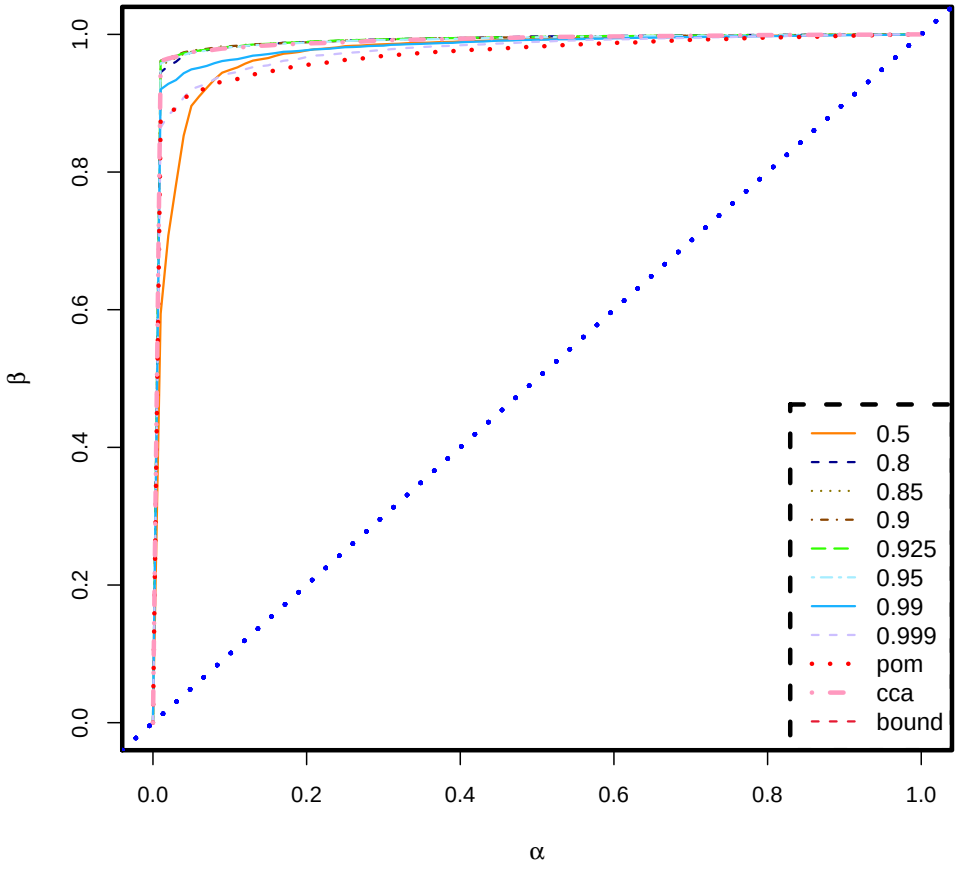


Figure 10.4: Power (β) vs Type I error (α) plot for different w values for the Dirichlet setting (noisy case)

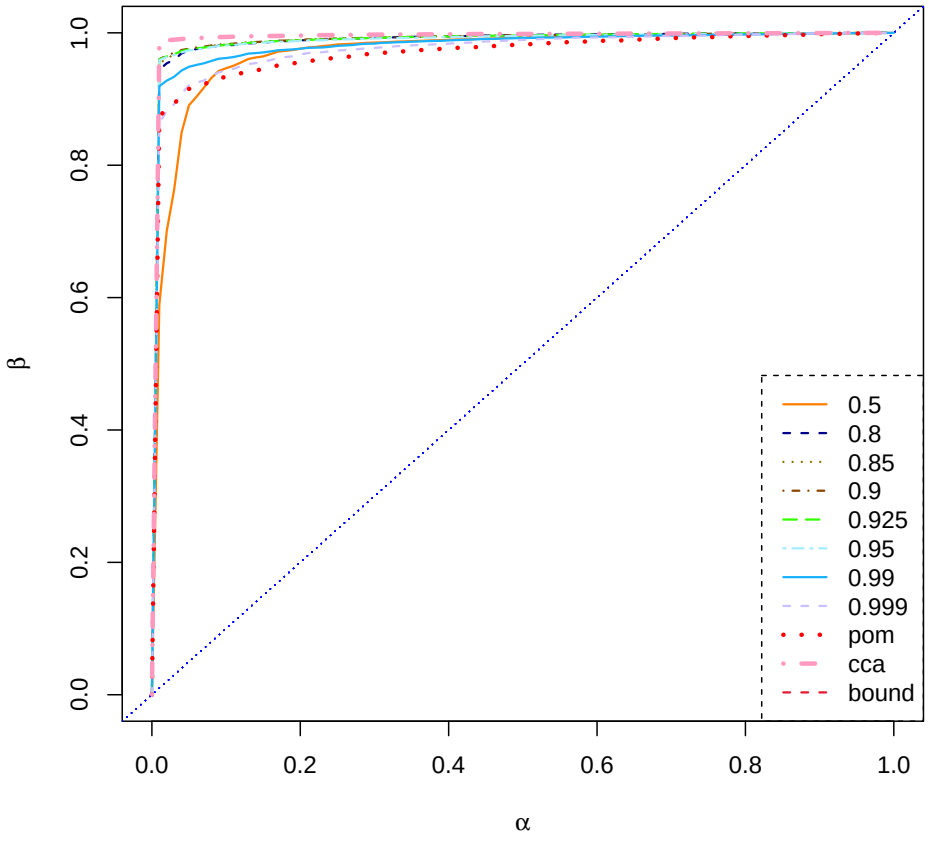


Figure 10.5: Power (β) vs Type I error (α) plot for different w values for the Dirichlet setting (noiseless case)

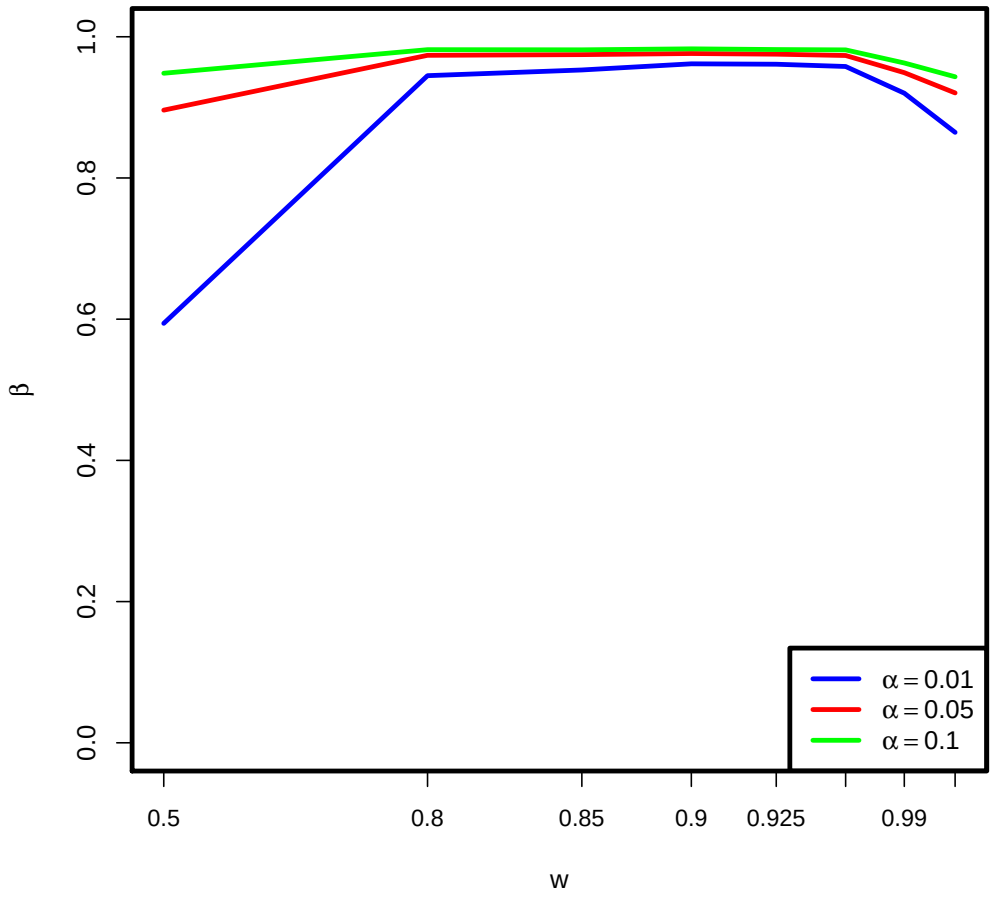


Figure 10.6: Power (β) vs w plot for different Type I error (α) values for the Gaussian setting (noisy case)

Carlo replicate. The number of replicates for which a particular w value resulted in the highest AUC measure is shown in the bar chart in Figure 10.7. Only the non-zero counts are shown in the plot. The estimate $\hat{w}^*$ can be chosen to be 0.925 because it is the mode of the w^* estimates from all of the replicates.

For each MC replicate, the estimate of w^* (the value of w that results in the highest AUC measure) might have a different value. This is hinted at by the fact that the $\beta(w)$ vs w plots exhibit a plateau near the maximum. The number of replicates for which a particular w value resulted in the highest AUC measure is shown in the bar chart in Figure 10.7 for 400 MC replicates. Figure 10.7 clearly shows that w^* should be estimated based on multiple MC replicates. The mode of the w^* values from each MC replicate is an appropriate estimator. For the results plotted in Figure 10.7, the estimate $\hat{w}^*$ can be chosen as 0.925.

Note that in Figure 10.3 for $\alpha = 0.05$, $\beta_{\alpha=0.05}(w = 0.99) \geq \beta_{\alpha=0.05}(w = 0.5)$. However, for $\alpha = 0.3$, $\beta_{\alpha=0.3}(w = 0.99) \leq \beta_{\alpha=0.3}(w = 0.5)$. This justifies our comment that w^* must be defined with respect to the AUC measure or a specific α value.

Note that for all of the settings, the estimate of the optimal w^* has higher power than $w=0.5$ (the unweighted case). To test the statistical significance of this observation, we consider the following hypothesis test: the null hypothesis that $\mathcal{H}_0 : \beta_\alpha(\hat{w}^*) \leq \beta_\alpha(w = 0.5)$ is tested against the alternative hypothesis $\mathcal{H}_A = \beta_\alpha(\hat{w}^*) > \beta_\alpha(w = 0.5)$. The least favorable null hypothesis is that $\mathcal{H}_0 : \beta_\alpha(\hat{w}^*) = \beta_\alpha(w = 0.5)$.

McNemar's test will be used to compare the two predictors (referred to as C_1 and C_2

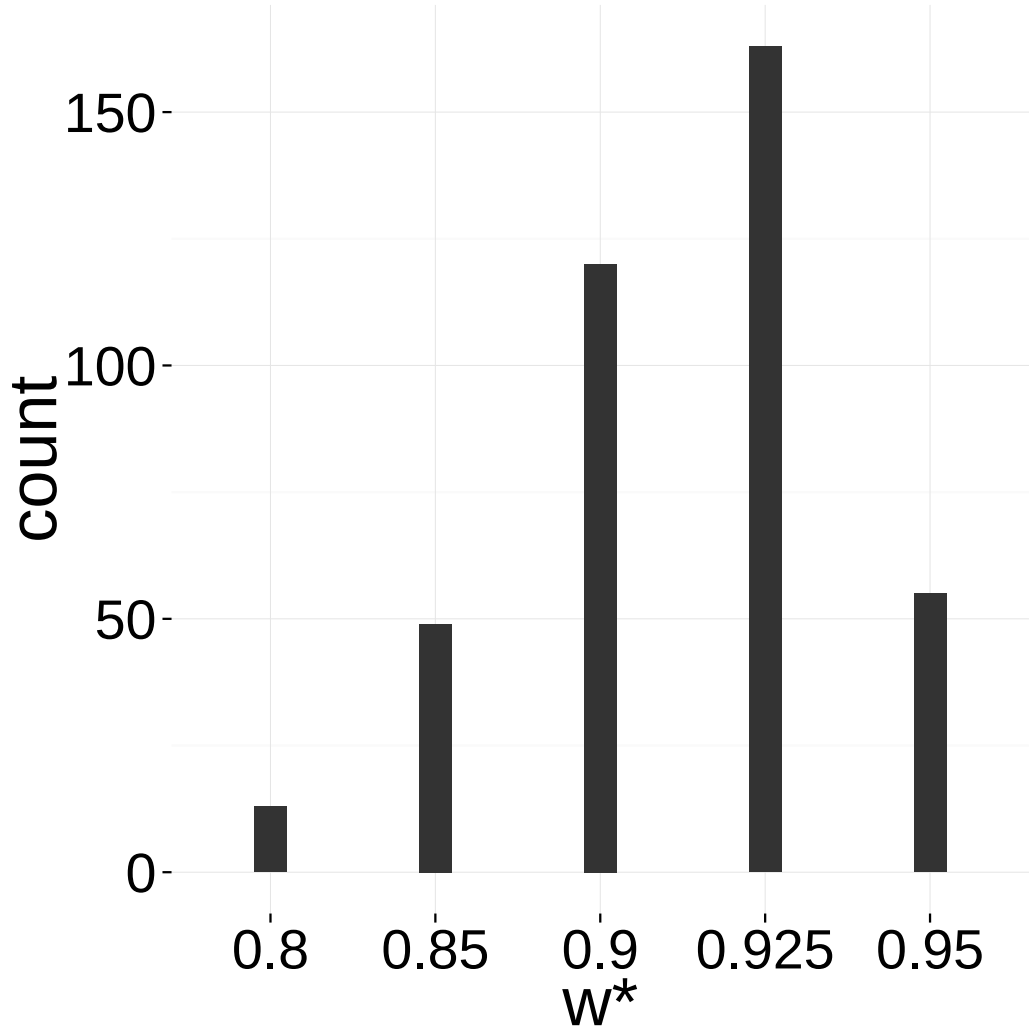


Figure 10.7: Histogram of w^* values for the Gaussian setting

with $w = 0.5$ and $w = w^*$ at a fixed α value.

10.1.1 McNemar's Test

Using the previous notation, the test statistic will be denoted by $T_a(w)$ under the alternative hypothesis and by $T_0(w)$ under the null hypothesis. For a fixed α value, one can compute two critical values:

$$c_{0.5} = max_l\{\mathbb{P}\left[T_0(0.5) > l\right] < \alpha\}, c_{w^*} = max_l\{\mathbb{P}\left[T_0(w^*) > l\right] < \alpha\}$$

. These critical values determine two binary classifiers if we interpret the hypothesis testing as deciding whether a new pair is "matched" or not and the test statistic as a score. Hypothesis testing is more nuanced than a binary decision problem, but for the sake of comparing the two tests, we can treat it as such. To compare the two statistical tests with $w = 0.5$ and $w = w^*$, simulation results are used to compute 2×2 contingency tables of correct decisions and incorrect decisions made by each statistical test (or, equivalently, true and false classifications made by two classifiers). Let $\mathscr{D}^{(i)}$ denote the test dissimilarities for the i^{th} new test pair, let $\tau(\mathscr{D}^{(i)})$ denote the test statistic for the oos-embedding of that pair, and let $m_{\mathscr{D}^{(i)}}$ denote a binary variable whose value is 1 if the pair is matched and 0 otherwise. We denote the decision outcome (whether the true or false decision is made) for the i^{th} test pair by two binary variables g_1^i and g_2^i,respectively. If $g_1^i = 1$ and $g_2^i = 0$ for the l^{th} MC replicate, the first test made the correct decision and the second

test made the incorrect decision with regard to the null and alternative hypotheses.

$$g_1^i = \mathbb{I}(\mathbb{I}(\tau(\mathscr{D}^{(i)}) > c_{0.5}) = m_{\mathscr{D}^{(i)}}) \text{ for the first statistical test}$$

$$g_2^i = \mathbb{I}(\mathbb{I}(\tau(\mathscr{D}^{(i)}) > c_{w^*}) = m_{\mathscr{D}^{(i)}}) \text{ for the second statistical test}$$

Consider the contingency table for any Monte Carlo replicate given by

$$G^{(l)} = \begin{array}{|c|c|} \hline e_{00}^{(l)} & e_{10}^{(l)} \\ \hline e_{01}^{(l)} & e_{11}^{(l)} \\ \hline \end{array}$$

where $e_{uv}^{(l)} = \sum_i \mathbb{I}(\{g_1^i = u\}\&\&\{g_2^i = v\})$ is equal to the number of instances at which the true hypothesis was identified correctly ($g_1^i = 1$) or incorrectly ($g_1^i = 0$) by the first test and correctly ($g_2^i = 1$) or incorrectly ($g_2^i = 0$) by the second test *in the l^{th} MC replicate.*

Under the null hypothesis that the two predictors have the same power at α,

$$\mathbb{P}[\big(\{g_1^i = 1\}\&\&\{g_2^i = 0\}\big)] = \mathbb{P}[\big(\{g_1^i = 0\}\&\&\{g_2^i = 1\}\big)].$$

Thus, a one-sided sign test is appropriate, in which the test statistic $e_{01}^{(l)}$ is distributed according to the binomial distribution, $\mathcal{B}(e_{10}^{(l)} + e_{01}^{(l)}, 0.5)$.

We consider simulated data with the noisy version of the Gaussian setting for this McNemar's test. The critical values $c_{0.5}$ and c_{w^*} were computed with type I error $\alpha = 0.05$ for the two tests. When comparing the null hypothesis that $\mathcal{H}_0 : \beta_\alpha(\hat{w}^*) = \beta_\alpha(w = 0.5)$ against the alternative $\mathcal{H}_A = \beta_\alpha(\hat{w}^*) > \beta_\alpha(w = 0.5)$, the p-value is $p < 1.09E - 24$, which indicates that the power obtained using the estimate of the optimal w^* is significantly greater than the power obtained when using $w = 0.5$.

Under the null distribution, we expect the p-values for each MC replicate to be uniformly distributed. We find that the distribution of p-values from McNemar's tests is skewed, and we reject $\mathcal{H}_0$ for 55% of the Monte Carlo replicates.

10.2 Effects of the parameters of the data model

Another topic to be investigated is how the parameters of the distribution of data, such as p, q, r, c, and d, affect the results. We speculated that as q, the number of noise dimensions, increases, the performance of the CCA approach would suffer due to spurious correlations. We tested our speculation using simulated data in the Gaussian Setting with $q = 90$. The results are visualized in the bundle of ROC curves in Figure 10.8. Both CCA and regularized CCA are not competitive with the JOFC approach with the appropriate w values. In fact, the ROC curve for CCA is not very distinct from a random guess. We conclude that the CCA approach is not robust with respect to a large number of noise dimensions, no matter what the magnitude of the noise is (which is controlled by the parameter c).

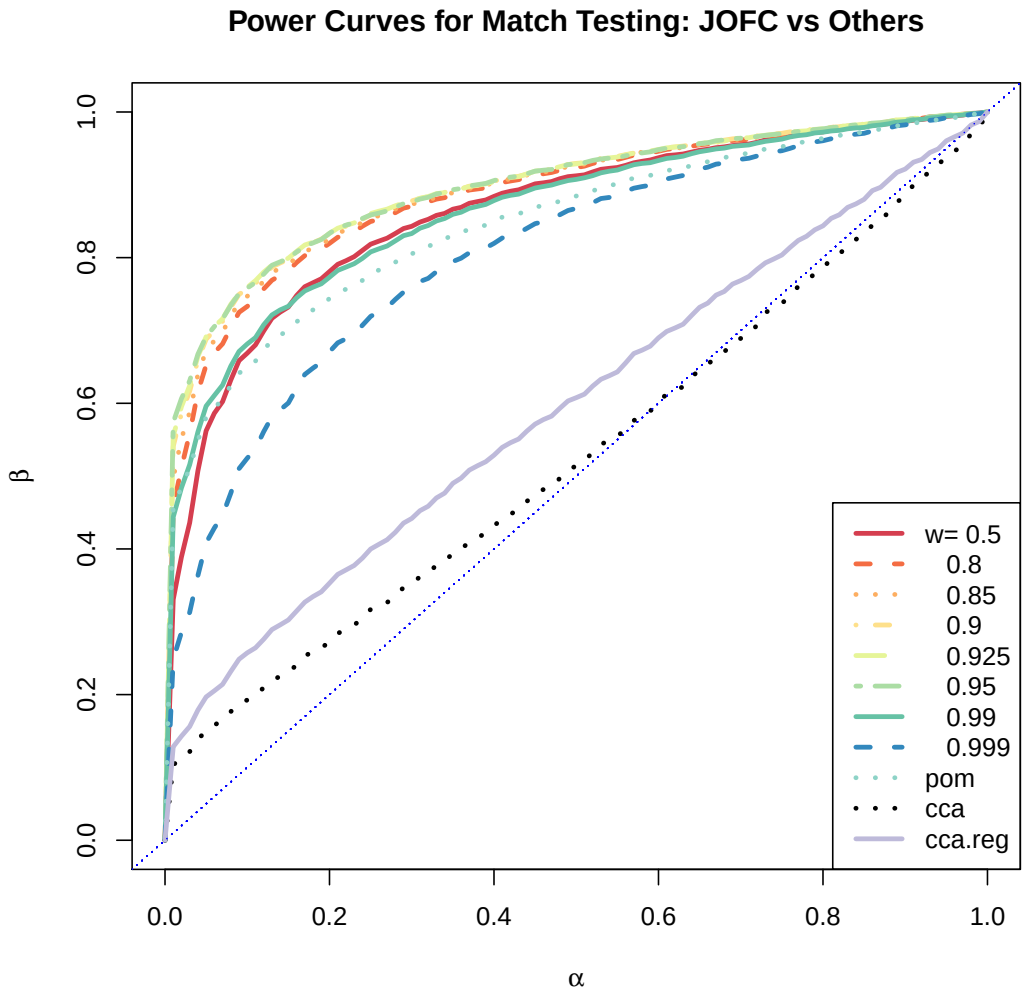


Figure 10.8: Large Noise Dimension Behavior of JOFC, P○ M and CCA approaches

10.3 Match Testing when the number of conditions, K is larger than 2

We noted previously that all of the approaches are generalizable to $K > 2$ conditions, although an ambiguity needs to be resolved. The alternative hypothesis could be defined as the case in which at least one of the K new dissimilarities are pairwise unmatched ($H_{A1} : \exists i, j, 1 \leq i < j \leq K : \boldsymbol{y}_i \nsim \boldsymbol{y}_j$) or could be defined as the case in which absolutely none of the K dissimilarities are pairwise matched ($H_{A2} : \forall i, j, 1 \leq i < j \leq K : \boldsymbol{y}_i \nsim \boldsymbol{y}_j$). We chose the alternative H_{A1} for our simulations.

To adapt the P∘M approach to this setting, one can use Procrustes Analysis generalized to more than two configurations. Generalized Procrustes Analysis [37] is described in section 7.3.

We have also described generalized CCA in section 8.6. Of the different choices for the generalization of CCA, the SUMCOR criterion was chosen.

To test whether the P∘M, JOFC, and generalized CCA approaches are appropriate for this setting, the simulations in section 10.1 were repeated with K-condition data that were generated by a multivariate normal model with $K = 3$ conditions.

We investigate the "noisy" case for this setting, i.e., q-dimensional noise vectors of magnitude c were added to the matched measurements, and the "signal" vectors were multiplied by $1 - c$.

The ROC curves for these simulations are shown in 10.9.

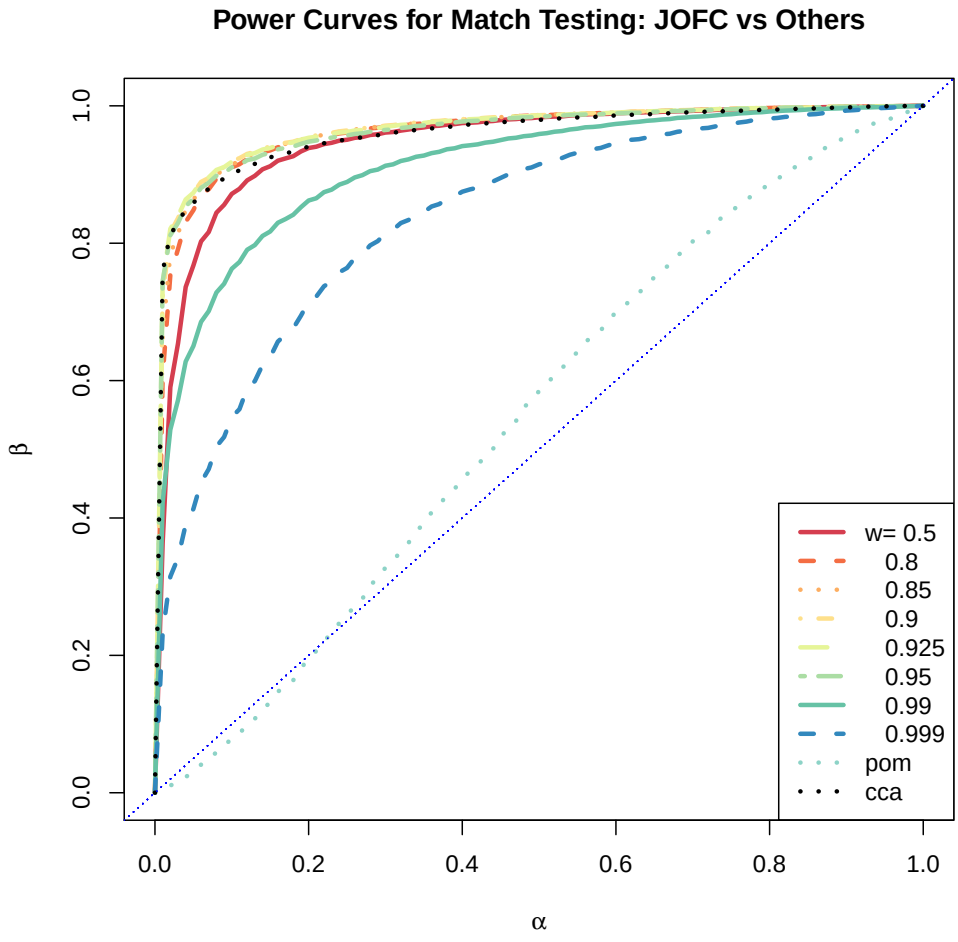


Figure 10.9: Power (β) vs Type I error (α) plot for different w values for the Gaussian setting with $K = 3$ conditions (noisy case)

The results indicate behavior that is similar to that of the $K = 2$ case. Different values of w have significantly different ROC curves. JOFC is thus a suitable approach for match detection when either two ($K = 2$) conditions or more than two ($K > 2$) conditions are used.

10.4 Experiments on Wiki Data

To test the JOFC approach with real data with different conditions, we used a collection of online Wikipedia articles. Based on the hyperlinks between Wikipedia articles, the directed 2-neighborhood of the document "Algebraic Geometry" were collected from the English Wikipedia site. This collection of 1382 articles and the correspondence of each article to the French Wikipedia site is our real-life dataset. It is possible to utilize both the textual content of the documents and the hyperlink graph structure. The textual content of the documents is summarized by the bag-of-words model. Dissimilarities between documents in the same language are computed using the Lin-Pantel discounted mutual information [42, 43] and cosine dissimilarity $k(x_{ik}; x_{jk}) = 1 - (x_{ik}x_{jk})/(\|x_{ik}\|_2\|x_{ik}\|_2)$. The dissimilarities based on the hyperlink graph of the collection of the articles are, for each pair of vertices i and j, the number of vertices one must travel to go from i to j. Further details about this dataset are available in [44]. Only dissimilarities based on the textual content will be considered for our experiments presented here.

The exploitation task is still testing for the matchedness of vertices between different conditions, which, in this case, are wiki articles on the same topic in different languages. For hypothesis testing, four randomly held-out documents – one matched pair and one unmatched pair – are used to compute the empirical type I error α and estimate of power based on the critical value computed from the distribution of the test statistic for the remaining 1380 matched pairs. The test statistic is computed using one of the three mentioned approaches: CCA, P∘M, or JOFC. The two sets of held-out matched pairs are embedded as $\tilde{y}_1$ and $\tilde{y}_2$, via OOS embedding, to estimate the null distribution of the test statistic $T_0 = d(\tilde{y}_1; \tilde{y}_2)$. This allows us to estimate critical values for any specified Type I error level. Then, the two sets of held-out unmatched pairs are embedded as $\tilde{y}_1^{(u)}$ and $\tilde{y}_2^{(u)}$ via OOS embedding. $T_a = d(\tilde{y}_1^{(u)}; \tilde{y}_1^{(u)})$ will give us an empirical distribution of the test statistic under the alternative hypothesis. The distribution under the null hypothesis and under the alternative hypothesis can be used to estimate the power. The target dimensionality d is determined by the Zhu and Ghodsi automatic dimensionality selection method [45], resulting in d = 6 for this data set.

The results show that fidelity is prioritized more compared with the Gaussian and Dirichlet simulations presented in section 10.1. Our conclusion is that there is no universal w^* because its value depends on the data distribution and the inference task.

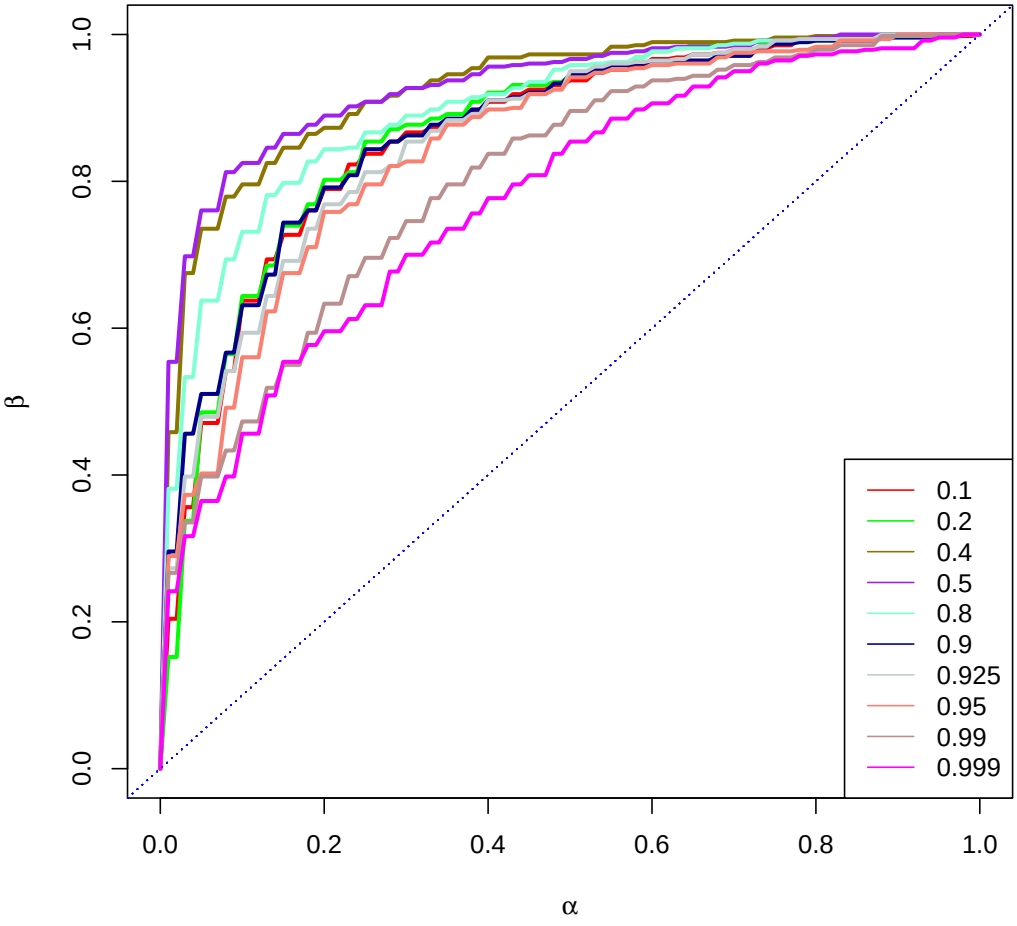


Figure 10.10: Match detection using the Wikipedia dataset. Different w values listed in the legend correspond to different ROC curves.

10.5 Model Selection

For the simulations presented until now, the embedding dimension d was set to 2. This was a convenient choice that allowed us to investigate various aspects of the JOFC and competing approaches. However, more care is required in the selection of this parameter because it plays such a large role in performance in general learning settings. To investigate the performance of the JOFC approach as d changes, we ran the usual Gaussian setting simulations. The signal dimension was set to $p = 10$, and different $d = 2, 5, 7, 10, 15$ values were used to test the JOFC approach.

The ROC curve plots in 10.11 and 10.12 show the effect of the d parameter on the performance of different methods for the Gaussian setting for the noisy case.

The results show the difference in sensitivity of the different approaches to the embedding dimension. For larger d, CCA and regularized CCA exhibit a serious degradation in performance. We expect that this degradation is again due to a spurious correlation phenomenon, where more noise dimensions appear in the embedding as the embedding dimension increases. At the same time, the performances of P$\circ$M and and JOFC with $w = 0.5$ improve with increasing embedding dimension, and they are the approaches that have the best performing test statistic. JOFC with the highest w values $0.95, 0.99, 0.999$ perform slightly worse, whereas the ROC curves for the other w values are more or less the same. Increasing the embedding dimension seems to push w^* toward the fi-

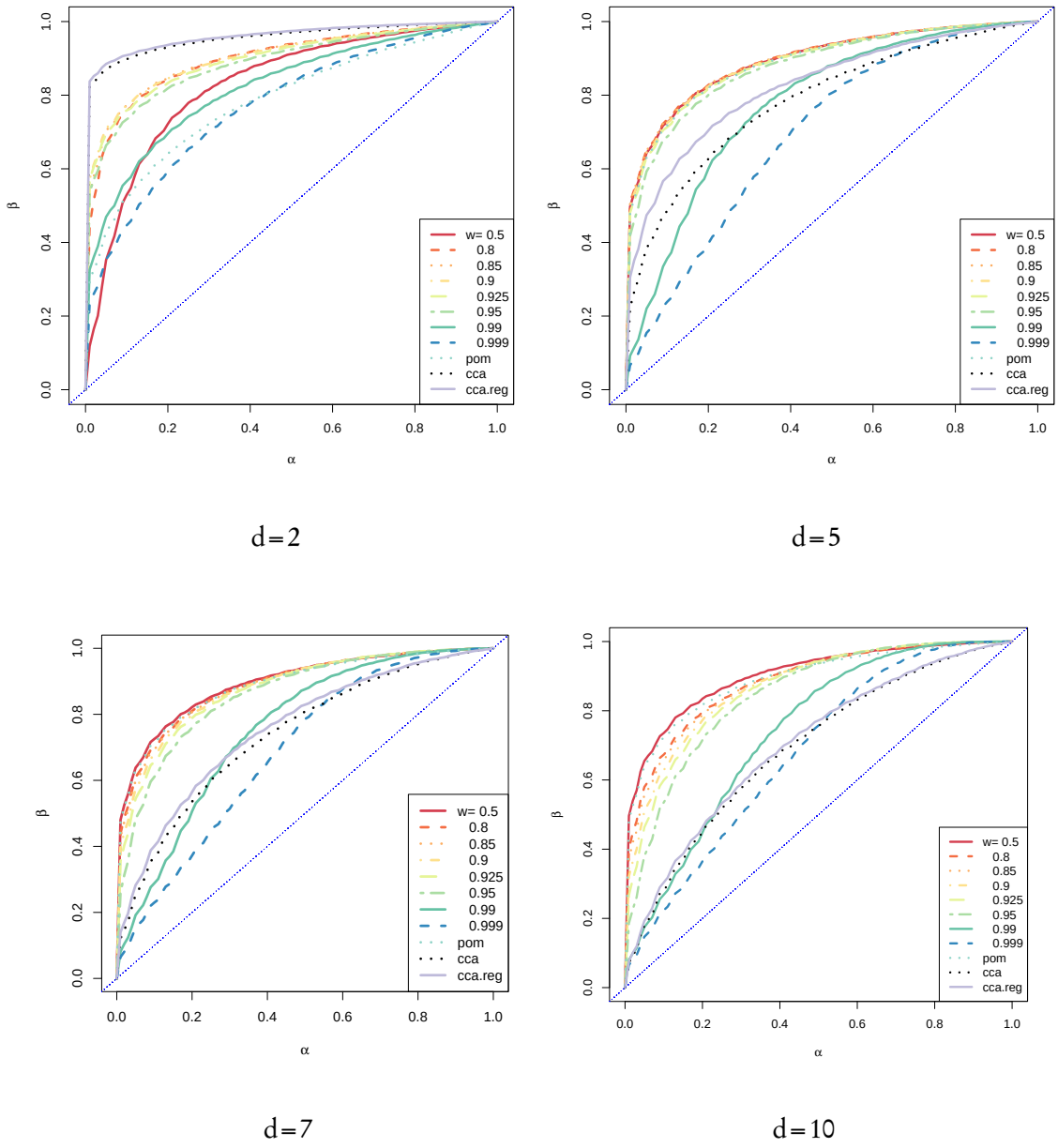


Figure 10.11: Effect of the d parameter on the ROC curves

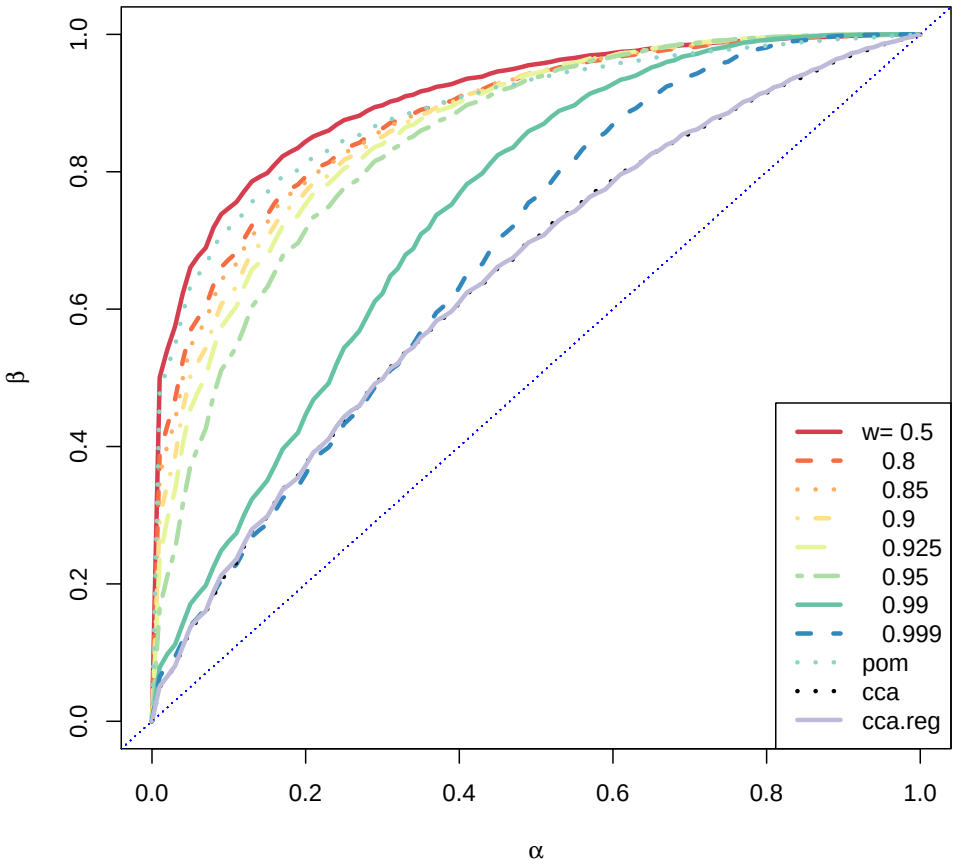


Figure 10.12: Effect of the d parameter on the ROC curves,d = 15

delity end ($w = 0$) of the fidelity-commensurability tradeoff. These results require more investigation before we can provide a rigorous explanation as to how the embedding dimension affects the different approaches (JOFC and P$\circ$M). Specifically, how the null and alternative distributions of the test statistic for the different approaches change with the embedding dimension d should be investigated.

Chapter 11

Seeded Graph Matching and Fast Approximate Quadratic Programming

11.1 Introduction to Graph Matching

Another application of the JOFC approach is a variant of the graph matching problem. First, we define the general graph matching problem.

11.1.1 Graph Matching

Consider two graphs $G_1 = (V_1, E_1)$ and $G_2 = (V_2, E_2)$ such that $|V_1| = |V_2|$. Let (u, v) denote the edge between vertices u and v. Suppose there exists a bijection $\mathfrak{f}$ between V_2 and V_1 such that

$$(\mathfrak{f}(u_2), \mathfrak{f}(v_2)) \in E_1 \text{ iff } (u_2, v_2) \in E_2 \quad \forall u_2, v_2 \in V_2.$$

Then, the *exact* graph matching problem is to determine $\mathfrak{f}$. No efficient algorithms are known to exist to solve this problem for general graphs [46]. Determining the existence of such an isomorphism between the two graphs is an easier decision problem referred to as *graph isomorphism*. *Graph isomorphism* is not only of unknown complexity, it is also a strong candidate for representing an intermediate complexity class between $\mathcal{P}$ and $\mathcal{NP}$, assuming $\mathcal{P} \neq \mathcal{NP}$.

Regardless of the existence of an isomorphism between the two graphs, we are interested in the bijections $\{f : V_2 \to V_1\}$ such that the graph (G_1') consisting of V_1 and the edges $(f(u_2), f(v_2)) \, \forall u_2, v_2 \in V_2$ is a good approximation of G_1. The *approximate* graph matching problem[1] is defined as the task of finding such a bijection f that minimizes "the degree of mismatch" between G_1' and G_1 . We measure this degree of mismatch with a function denoted by $\tau(f; G_1, G_2)$. In unweighted graphs, this degree of mismatch is the

[1] We will refer to this problem as *the* graph matching problem in this document.

number of edge disagreements.

$$\begin{aligned}\tau(f; G_1, G_2) = & \quad |(u_2, v_2) \in E_2 : (f(u_2), f(v_2)) \notin E_1| \\ + & \quad |(u_1, v_1) \in E_1 : (f^{-1}(u_2), f^{-1}(v_2)) \notin E_2| \end{aligned} \tag{11.1}$$

In weighted graphs, the degree of mismatch would be a function of the difference of the weights of the corresponding edges, such as:

$$\tau(f; G_1, G_2) = \sum_{u_2, v_2 \in V_2} |w(u_2, v_2) - w(f(u_2), f(v_2))| \tag{11.2}$$

where $w(a, b)$ is the weight of the edge between vertices a and b . If the existence and nonexistence of an edge between two vertices in an unweighted graph correspond to an edge weight of 1 and 0, respectively, for a weighted graph, then the degree of mismatch defined for the unweighted graph case is a special case of that defined for the weighted graph. [2]

The *approximate* graph matching problem is an important research topic, and has many practical applications [46, 48–51]. We will propose two approaches to solve this problem. In this chapter, we present the optimization based approach; in chapter 12, we will present the approach based on the JOFC embedding.

Assume that G_1 and G_2 are unweighted graphs. [3] We consider a specific version of the approximate graph matching problem in which part of a bijection between V_1 and V_2 are given, and the task is to complete the bijection minimizing number of disagreements.

[2] The τ function could depend on another function of the weights [47]. We choose to use the absolute difference between the weights to maintain the connection between the weighted and unweighted cases.

[3] We should note that most of the notation and methods carry over to the weighted case.

For $1 \leq r < n = |V_1|$, let S be a set containing r tuples with each tuple containing a unique element each from V_1 and V_2. Then, given the two graphs G_1, G_2 and the tuple set S, the objective function for the seeded graph matching problem is

$$\tau_{sgm}(f; G_1, G_2, S) := \tau(f; G_1, G_2) \tag{11.3}$$

subject to the constraint $\forall\, (v_1, v_2) \in S, \quad f(v_2) = v_1$.

The tuples of vertices are referred to as "seeds" and we will refer to this variant of the graph matching problem as the "seeded graph matching" (SGM) problem.

Remark Although in the definition of the problem, we have not assumed any relation between the two graphs or between the seed tuples, there is an implicit understanding that there is some correlation between the connectivity of the two graphs and the seed tuples provide some of the true correspondences. It is possible the correlation between the two graphs is weak and the seed tuples contain false correspondences. However, if there is an underlying correspondence between the vertices of the two graphs and the seed tuples contain a portion of the corresponding vertices, we could hope to recover a considerable part of the true correspondences and judge our performance with respect to the ground truth of true correspondences.

It will be convenient to formulate the SGM problem with the adjacency matrices as follows:

Suppose $A, B \in \mathbb{M}_{(m+l)\times(m+l)}$ are adjacency matrices for graphs G_1 and G_2 parti-

tioned as (m rows and then l rows, m columns and then l columns):

$$A = \begin{bmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{bmatrix} \qquad B = \begin{bmatrix} B_{11} & B_{12} \\ B_{21} & B_{22} \end{bmatrix}.$$

Without loss of generality, suppose that $V_1 = [m+l]$, $V_2 = [m+l]$ and $S = \{(i,i) : i \in [m]\}$, i.e., the first m vertices of G_1, correspond to the first m vertices of G_2, respectively in the given part of the bijection. We wish to complete the bijection by matching the remaining l pairs of vertices. That is, we seek a permutation matrix $P \in \mathbb{M}_{l\times l}$ such that the permutation represented by $(I_{m\times m} \oplus P)$ is the bijection f that minimizes $\tau_{sgm}(f; G_1, G_2, S)$.

It is obvious that P and the seed tuples in S determine f: $V_2 \to V_1$, the bijection between the two graphs. So given the seeding S, we define $\mathcal{F}_S(\cdot)$, a one-to-one mapping from the set of $l \times l$ permutation matrices denoted by Π_l to the set of bijections from V_2 to V_1 denoted by $\mathcal{B}_{V_2\to V_1} = \{g : V_2 \to V_1,\ g \text{ is one-to-one }\}$:

$$\mathcal{F}_S(P) : \Pi_l \to \mathcal{B}_{V_2\to V_1}.$$

Solving for a $l \times l$ permutation matrix that minimizes $\tau_{sgm}(\mathcal{F}_S(P); G_1, G_2, S)$ is equivalent to solving for a bijection that minimizes $\tau_{sgm}(f; G_1, G_2, S)$.

So we formulate the seeded graph matching problem as an optimization problem: we seek P that minimizes $h(P)$ over all permutation matrices of size $l \times l$, where the function $h(P) = \tau_{sgm}(\mathcal{F}_S(P); G_1, G_2, S)$ measures the mismatch between G_1 and the resulting graph when the permutation represented by P is applied to the vertices of G_2. For unweighted graphs, the degree of mismatch is characterized by the number

of adjacency disagreements, which is conveniently represented in terms of P and the adjacency matrices of G_1 and G_2:

$$h(P) = \|A - (I_{m\times m} \oplus P)^T B(I_{m\times m} \oplus P)\|_1$$

subject to P being a permutation matrix. $h(P)$ is written in terms of the partition block matrices as

$$\left\| \begin{bmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{bmatrix} - \begin{bmatrix} I_{m\times m} & 0_{m\times l} \\ 0_{l\times m} & P \end{bmatrix} \begin{bmatrix} B_{11} & B_{12} \\ B_{21} & B_{22} \end{bmatrix} \begin{bmatrix} I_{m\times m} & 0_{m\times l} \\ 0_{l\times m} & P^T \end{bmatrix} \right\|_1 .$$

It is possible to state the seeded graph matching problem as the minimization of various different functions over all permutation matrices P. Note that $\mathscr{P} = I_{m\times m} \oplus P$ is a permutation matrix, and both the columns and rows of B are permuted when it is left-multiplied and right-multiplied with $\mathscr{P}$ (which yields $\mathscr{P}B\mathscr{P}^T$). Instead of permuting both rows and columns of B, we can permute the columns of A (right-multiply by $\mathscr{P}$) and the rows of B (left-multiply by $\mathscr{P}$). Because the norm of the matrix difference is independent of the ordering of the rows and columns, $\|A\mathscr{P} - \mathscr{P}B\|_1$ would yield the same value as the original objective function, $\forall P \in \Pi_l$.

For the set of permutation matrices, the objective function with the ℓ_2-norm, $\|A - \mathscr{P}B\mathscr{P}^T\|_2$, is equivalent to the original objective function $\|A - \mathscr{P}B\mathscr{P}^T\|_1$, because the entries of the matrix difference between A and $\mathscr{P}B\mathscr{P}^T$ are either 0, 1 or -1 . Another ℓ_2 objective function $\|A\mathscr{P} - \mathscr{P}B\|_2$ can be shown to be equivalent to $\|A - \mathscr{P}B\mathscr{P}^T\|_2$ using the permutation argument we used in the previous paragraph for the ℓ_1 norm.

The minimization of $h(P)$ over the set of all permutation matrices can also be shown to be equivalent to the maximization of

$$\text{trace}\left(A^T(I_{m\times m} \oplus P)B(I_{m\times m} \oplus P^T)\right) \tag{11.4}$$

by expanding out $\|A - \mathscr{P}B\mathscr{P}^T\|_2^2 = \|A\|_2^2 + \|B\|_2^2 - 2 \cdot \text{trace}(A^T\mathscr{P}B\mathscr{P}^T)$ and ignoring the constant terms. Note that this simplified ℓ_2 formulation is a special case of the quadratic assignment problem (QAP). The quadratic assignment problem minimizes $\sum_{i,j\in[p]} \theta_{ij}\omega_{\pi(i)\pi(j)}$ with respect to a permutation π of p elements, given a collection of weights $\{\theta_{ij}; \quad i,j \in [p]\}$ and distances $\{\omega_{ij}; \quad i,j \in [p]\}$. The objective function can be written in matrix form as $\text{trace}(\Theta P \Omega P^T)$, where P is a permutation matrix and Θ and Ω are the matrices of weights $\{\theta_{ij}\}$ and distances $\{\omega_{ij}\}$, respectively. Thus, the ℓ_2 formulation in (11.4) is equivalent to the the special case of QAP, when weights and distances are constrained to be 0 or 1.

The different formulations are equivalent for the set of permutation matrices, i.e., their global extrema are the same. We will consider relaxations of each formulation, where we remove the integrality constraint of P and optimize the objective function over the set of $l \times l$ doubly stochastic matrices, $\mathcal{DS}_l$. For example, for the ℓ_1 formulation,

Given m seeds, minimize $h(P) = \|A - (I_{m\times m} \oplus P)^T B(I_{m\times m} \oplus P)\|_1$ with respect to a permutation matrix $P \in \Pi_l$, i.e.

$\mathbf{1}^T P = \mathbf{1}$, $P\mathbf{1} = \mathbf{1}$, and

$[P]_{ij} \in \{0, 1\}, \quad \forall i, j \in [l]$

is relaxed to

Given m seeds, minimize $h(P) = \|A - (I_{m\times m} \oplus P)^T B(I_{m\times m} \oplus P)\|_1$ with respect to a doubly stochastic matrix $P \in \mathcal{DS}_l$, i.e.

$\mathbf{1}^T P = \mathbf{1}$, $P\mathbf{1} = \mathbf{1}$ and

$[P]_{ij} \geq 0, \quad \forall i, j \in [l]$

by relaxing the integrality constraint of the entries of P to non-negativity. After this relaxation, the feasible region is expanded to the set of doubly stochastic matrices, which is the convex hull of permutation matrices. This means we have a feasible region that is a polyhedral set.

We can apply the same relaxation to different formulations, as the original feasible regions for all of the formulations are $P \in \Pi_l$. To show equivalencies between different objective functions we used the fact that $\mathscr{P}$ is a permutation matrix. Since this fact does not hold after the feasible region is expanded to $\mathcal{DS}_l$, different relaxations are not necessarily equivalent (see subsection 11.2.3). In fact, the different relaxation formulations might have different solutions, or might have different convergence behaviour as we find out during our investigations presented in subsection 11.2.4.

11.2 Fast Approximate Quadratic Programming for the Seeded Graph Matching problem

Consider the formulation (11.4) and its relaxation by expanding the feasible region to the set of doubly stochastic matrices. Solutions to this relaxation can be found via an iterative nonlinear optimization algorithm, the Frank-Wolfe Method. The idea of this method is to successively solve local linearizations of the objective function, using the solution of previous iteration as the location of the linearization in the current iteration. For the graph matching problem, the subproblem of solving the local linearization is equivalent to solving the linear assignment problem for which polynomial time algorithms exist. Most notable of these is the Hungarian algorithm [52].

Once the algorithm terminates due to a stopping criteria, the output of the algorithm is an approximation to the solution of the relaxed problem. The algorithm might terminate in a local minimum, and a common solution for this issue is multiple randomized initializations of the algorithm. Also the solution found by the Frank-Wolfe algorithm is possibly a non-integer solution. Since we must provide a permutation matrix as the solution to the seeded graph matching problem, we must choose a permutation matrix that is close to the solution of the Frank-Wolfe algorithm. This final step of projecting to the set of permutation matrices is also equivalent to the linear assignment problem

which can be solved via the Hungarian algorithm [52].

To summarize, the three steps of the FAQ algorithms are:

- Initialize the doubly stochastic $l \times l$ matrix P.
- Find a tentative solution $\hat{P}$ to the formulation in (11.4) (which is a QAP problem)
- Find the permutation matrix that is closest to $\hat{P}$.

We provide details of the FAQ algorithm for SGM in the following sections.

11.2.1 Frank-Wolfe algorithm

A brief review of the Frank-Wolfe algorithm is necessary before we further describe the FAQ method for Seeded Graph Matching. The F-W algorithm provides a solution for the minimization of a differentiable function, denoted by $h(x)$, over a bounded and convex domain $\mathbf{S}$.

In each iteration of the F-W algorithm, in the first step, a linear approximation of the function $h(x) \approx h(x^i) + (\nabla h(x^i))^T (x - x^i)$ is minimized. In the second step, the original function is minimized with the domain restricted to the line segment between $\hat{y}$ and x^i. When $h(x)$ is quadratic, a $\hat{\alpha}$ can be found analytically. The two steps are repeated until termination conditions are met.

Result: x^*

$i = 1$;

$\alpha = 1$;

$x^{(1)}$ = Random element of **S** or initial estimate of x^* ;

while $i < i_{max}$ *and (*$\hat{\alpha} > \epsilon$ *or* $\|\nabla h(x^{(i)})\| > \omega$*)* **do**

Solve $\hat{y} = \arg\min_y \nabla h(x^{(i)})^T y$ with respect to y ;

Solve $\hat{\alpha} = \arg\min_\alpha h(x^{(i)} + \alpha * (\hat{y} - x^{(i)}))$ over $\alpha \in [0, 1]$;

$x^{(i+1)} = x^{(i)} + \hat{\alpha} * (y - x^{(i)})$;

$i = i + 1$;

end

$x^* = x^{(i+1)}$;

Algorithm 1: Frank-Wolfe algorithm

11.2.2 rQAP$_1$ formulation of the Seeded Graph Matching problem and the FAQ Algorithm

Let us now present the derivation of the steps of the FAQ algorithm. The objective function that we use for FAQ is trace$A^T(I_{m\times m} \oplus P)B(I_{m\times m} \oplus P^T)$ in (11.4). This is a reformulation of the ℓ_2 norm of the matrix difference between A and $(I_{m\times m} \oplus P)B(I_{m\times m} \oplus P)^T$ when P is a permutation matrix. The feasible region for the optimization is the set of permutation matrices, Π_l. We relax this combinatorial optimization

problem by expanding the feasible region to the set of double stochastic matrices (the convex hull of the set of permutation matrices) of the same size. This yields our first formulation for the seeded graph matching problem which we call the relaxed quadratic assignment problem (rQAP$_1$).

Remark When the graphs that are matched are undirected graphs, the adjacency matrices A and B are symmetric matrices. Even though the symmetricity of A and B would allow us to further simplify the expressions, we do not make that assumption in the following derivation in order to make the results more general.

The objective function is

$$\begin{aligned} h(P) &= \text{trace}\left(\begin{bmatrix} A_{11}^T & A_{21}^T \\ A_{12}^T & A_{22}^T \end{bmatrix}\begin{bmatrix} I_{m\times m} & 0_{m\times l} \\ 0_{l\times m} & P \end{bmatrix}\begin{bmatrix} B_{11} & B_{12} \\ B_{21} & B_{22} \end{bmatrix}\begin{bmatrix} I_{m\times m} & 0_{m\times l} \\ 0_{l\times m} & P^T \end{bmatrix}\right) \\ &= \text{trace}\left(\begin{bmatrix} A_{11}^T & A_{21}^T \\ A_{12}^T & A_{22}^T \end{bmatrix}\begin{bmatrix} B_{11} & B_{12}P^T \\ PB_{21} & PB_{22}P^T \end{bmatrix}\right) \\ &= \text{trace}A_{11}^TB_{11} + \text{trace}A_{21}^TPB_{21} + \text{trace}A_{12}^TB_{12}P^T + \text{trace}A_{22}^TPB_{22}P^T \\ &= \text{trace}A_{11}^TB_{11} + \text{trace}P^TA_{21}B_{21}^T + \text{trace}P^TA_{12}^TB_{12} + \text{trace}A_{22}^TPB_{22}P^T \end{aligned}$$

which has the gradient $\nabla_P(h)$, presented as a matrix-valued function of P as

$$\boldsymbol{\nabla}(P) := A_{21}B_{21}^T + A_{12}^TB_{12} + A_{22}PB_{22}^T + A_{22}^TPB_{22}.$$

Note that $h(P)$ has a quadratic form with respect to P, which will help us with the one-dimensional optimization subproblem in the second step of each F-W iteration.

In our experiments, the Frank-Wolfe Algorithm was initialized with $\tilde{P} = \frac{1}{l}\mathbf{1}_l\mathbf{1}_l^T$. (This initialization is arbitrary. A random $\tilde{P}$ can be chosen for initializations. Different random initializations would alleviate the local minima problem.)

We now adapt the F-W algorithm to the minimization of $h(P)$. Consider iteration i of the algorithm. In the first step, let $\tilde{P}^{(i)}$ be the current estimate of P. We are supposed to compute $\hat{Q}$, which is the minimizer of $-\text{trace}Q^T\boldsymbol{\nabla}(\tilde{P}^{(i)})$ over all $l \times l$ doubly stochastic matrices $Q \in \mathcal{DS}_l$. Equivalently, trace $\left(Q^T\boldsymbol{\nabla}(\tilde{P}^{(i)})\right)$ is maximized with respect to Q.

$\hat{Q}$ can be assumed to be a permutation matrix. The Birkhoff-von Neumann theorem [53] states that the set of doubly stochastic matrices is the convex hull of the permutation matrices. Because trace$\left(Q^T\boldsymbol{\nabla}(\tilde{P}^{(i)})\right)$ is a linear function of Q, one of the extremum points of the convex hull (which are permutation matrices) will be a maximizer.[4] Thus, $\hat{Q}$ is a permutation matrix, and we can limit the feasible region to the set of permutation matrices. Therefore, the Hungarian Algorithm is used to minimize $-\text{trace}\left(Q^T\boldsymbol{\nabla}(\tilde{P}^{(i)})\right)$ subject to the constraint Q is a permutation matrix and will yield the optimal Q, which we denote by $\hat{Q}$.

The next step in the Frank-Wolfe algorithm is maximizing the objective function over the line segment from $\tilde{P}^{(i)}$ to $\hat{Q}$; i.e., maximizing the scalar-valued univariate function $z(\alpha) := h(\alpha\tilde{P} + (1-\alpha)\hat{Q})$ over $\alpha \in [0,1]$. This one-dimensional optimization can be

[4] Although the maximizer is possibly non-unique, the Hungarian algorithm that solves this linear problem will return one of the extremum points as its output.

solved with the quadratic formula once the coefficients have been computed. Denote

$$c := \text{trace}\left(A_{22}^T \tilde{P} B_{22}\right) \tilde{P}^T,\ d := \text{trace}\left(A_{22}^T \tilde{P} B_{22} \tilde{Q}^T + A_{22}^T \tilde{Q} B_{22} \tilde{P}^T\right)$$

, $e := \text{trace}\left(A_{22}^T \tilde{Q} B_{22} \tilde{Q}^T\right)$ and

$$u := \text{trace}\left(\tilde{P}^T A_{21} B_{21}^T + \tilde{P}^T A_{12}^T B_{12}\right), \quad v := \text{trace}\left(\tilde{Q}^T A_{21} B_{21}^T + \tilde{Q}^T A_{12}^T B_{12}\right).$$

Then (ignoring the additive constant trace$(A_{11}^T B_{11})$ without loss of generality), we have

$$z(\alpha) = c\alpha^2 + d\alpha(1-\alpha) + e(1-\alpha)^2 + u\alpha + v(1-\alpha),$$

which simplifies to $z(\alpha) = (c-d+e)\alpha^2 + (d-2e+u-v)\alpha + (e+v)$. Setting the derivative of z to zero yields the potential critical point $\hat{\alpha} := \frac{-(d-2e+u-v)}{2(c-d+e)}$. Since $\tilde{P}^{(i+1)}$ should be inside the feasible region (the convex hull of permutation matrices), we want the maximizer within the line segment from $\tilde{P}^{(i)}$ to $\hat{Q}$. So, we set $\hat{\alpha} := \min(1, \frac{-(d-2e+u-v)}{2(c-d+e)})$. If $\hat{\alpha} < \epsilon$, the algorithm terminates at that iteration because $\hat{P} = \tilde{P}^{(i)}$ has reached a local minimum. Otherwise, we set $\tilde{P}^{(i+1)} = \hat{\alpha}\tilde{P}^{(i)} + (1-\hat{\alpha})\hat{Q}$ if $\hat{\alpha} > \epsilon$ and repeat the steps for the next iteration.

At the termination of the Frank-Wolfe algorithm, it is quite possible that $\hat{P}$ is not a permutation matrix. One way to obtain a permutation matrix solution is to find $\tilde{P}^*$ that is as close as possible to $\hat{P}$ (in some sense). Assume that we require the closest permutation matrix in the ℓ_2 sense (the minimizer of $\|\hat{P} - \tilde{P}^*\|_2^2$ where $\tilde{P}^*$ is a permutation matrix). Note that in our discussion of the different formulations of the original optimization problem 11.4, we showed that maximizing $2 * \text{trace}\,(ST)$ with respect to

S was equivalent to minimizing $\|S - T\|_2^2$ when S was a permutation matrix. So, we compute the maximizer of trace $\left(\hat{P}\tilde{P}^*\right)$ instead of the ℓ_2 norm of the matrix difference. We have also shown, in the discussion of the FAQ algorithm, that the minimization of $2 *$ trace (ST) is solved by the Hungarian algorithm when S is constrained to be a permutation matrix. Therefore, we can use the Hungarian algorithm once more to obtain the closest permutation matrix to $\hat{P}$ in the ℓ_2 sense by minimizing trace $\left(\tilde{P}^*\hat{P}\right)$ with respect to $\tilde{P}^*$. This permutation matrix minimizer is the output of the FAQ algorithm for the SGM problem.

11.2.2.1 Demonstration of the FAQ algorithm on simulated data

We will demonstrate that the FAQ algorithm for SGM works by using graph data generated via the following data model:

Let $\mathbb{G}_1 = (V_1, E_1)$ be an Erdős-Renyi graph that consists of n vertices and let A be its adjacency matrix. The probability of an edge between any two vertices in V_1 is an independent Bernoulli trial. Thus, $[A]_{ij} \sim Bernoulli(0.5), \quad \forall i < j, \quad i, j \in [n]$, where $[A]_{ij}$ is the $(i, j)^{th}$ entry of the adjacency matrix A. Because A is symmetric, $[A]_{ji} = [A]_{ij}$. Another adjacency matrix, B, is a entry-wise bit-flipped version of the adjacency matrix of A, that is, $\{[B]_{ij} \,|\, [A]_{ij} = 0\} \sim Bernoulli(p_{10}), \quad \forall i < j, i, j \in [n]$ $\{[B]_{ij} \,|\, [A]_{ij} = 1\} \sim Bernoulli(p_{11}), \quad \forall i < j, i, j \in [n]$. We will introduce a perturbation parameter p_{pert} that determines the probability of bit-flipping an edge, $p_{pert} = p_{10} = 1 - p_{11}$. B is also symmetric, $[B]_{ji} = [B]_{ij}$.

We define the second graph $\mathbb{G}_2 = (V_2, E_2)$ with the adjacency matrix B. Note then, there is a true correspondence between $\mathbb{G}_1$ and $\mathbb{G}_2$, given by the identity permutation.

From n pairs of vertices, m ($0 \leq m < n$) seeds are randomly selected, yielding the subsets of vertices which are the seeds in each graph: $\sigma_m \subset V_1 = V_2$ (because the correspondence is given by the identity permutation, the seed pairs have the same vertex labels) . The assignment problem for the remaining $n - m$ pairs of vertices is solved using FAQ. The quality of the solution , $\hat{f}_m : V_2 \to V_1$, to the assignment problem, is evaluated by $\delta^{(m)} = \frac{|\{i \in V_1 - \sigma_m : \hat{f}_m(i) = i\}|}{n-m}$.

The results of our simulations are plotted in the figures 11.1,11.2,11.3. For the graph size, we chose $n = 600$. We generated pairs of random Erdős-Renyi graphs for different number of seeds, m, and we solved the FAQ problem for the remaining $n - m$ vertex pairs. The probability of flipping an entry of the adjacency matrix is the perturbation parameter p_{pert}, which is the variable on the x-axis. The performance measure, the proportion of true matches to the number of matches, is the variable on the y-axis. Note that under chance, the expected number of true matches is 1. This means that for completely random assignments of vertices, the performance measure of the assignments would be $\frac{1}{n-m}$, as shown with the dashed line. p_{pert} varies from 0 to 1 in increments of 0.1.

11.2.3 Relaxations of alternate formulations of the approximate seeded graph matching problem

Another quadratic assignment problem formulation of the approximate seeded graph matching problem, in which the objective function is minimized, is presented here, which we call rQAP$_2$. The objective function for rQAP$_2$ is $h(P) = \|A\mathscr{P} - \mathscr{P}B\|_2^2$, where $\mathscr{P} = (I_m \oplus P)$. Note that this function is *convex*.

For the unrelaxed problem, the feasible set for P is the set of permutation matrices. P is a orthogonal matrix, i.e., $\|PX\|_2 = \|X\|_2, \forall X \in \mathbb{M}_{l\times l}$. Using this norm-preserving property of P, the rQAP$_2$ objective function simplifies to -2 times the objective function of rQAP$_1$.

The objective function for rQAP$_2$ can be simplified as follows:

$$\begin{aligned} h(P) = & \quad \|A(I_m \oplus P) - (I_m \oplus P)B\|_F^2 \\ = & \quad \underbrace{\|A_{21} - PB_{21}\|_F^2}_{\text{Term (1)}} + \underbrace{\|A_{12}P - B_{12}\|_F^2}_{\text{Term (2)}} + \underbrace{\|A_{22}P - PB_{22}\|_F^2}_{\text{Term(3)}} + \underbrace{\|A_{11} - B_{11}\|_F^2}_{\text{Constant}} \end{aligned}$$

We now consider each term in turn. Consider term (1)

$$\begin{aligned}
\|A_{21} - PB_{21}\|_F^2 = & \quad \text{trace}\left[(A_{21} - PB_{21})^T (A_{21} - PB_{21})\right] \\
= & \quad \text{trace}\left[A_{21}^T A_{21} - B_{21}^T P^T A_{21} - A_{21}^T P B_{21} + B_{21}^T P^T P B_{21}\right] \\
= & \quad \text{trace}\left[A_{21}^T A_{21} - B_{21}^T P^T A_{21} - A_{21}^T P B_{21} + P^T P B_{21} B_{21}^T\right] \\
= & \quad \text{trace}\left[A_{21}^T A_{21} - 2 * B_{21}^T P^T A_{21} + P^T P B_{21} B_{21}^T\right] \\
= & \quad \underbrace{\text{trace}\left[A_{21}^T A_{21}\right]}_{(1.1)} - \underbrace{2\text{trace}\left[B_{21}^T P^T A_{21}\right]}_{(1.2)} + \underbrace{\text{trace}\left[P^T P B_{21} B_{21}^T\right]}_{(1.3)},
\end{aligned}$$

where the simplification in the fourth line occurs because $B_{21}^T P^T A_{21}$ and $A_{21}^T P B_{21}$ are transposes of each other. The three terms in the last line are referred to as (1.1), (1.2), and (1.3).

We make a similar simplification for term (2):

$$\begin{aligned}
\|A_{12}P - B_{12}\|_F^2 = & \quad \text{trace}\left[(A_{12}P - B_{12})^T (A_{12}P - B_{12})\right] \\
= & \quad \text{trace}\left[P^T A_{12}^T A_{12} P - B_{12}^T A_{12} P - P^T A_{12}^T B_{12} + B_{12}^T B_{12}\right] \\
= & \quad \text{trace}\left[P P^T A_{12}^T A_{12} - B_{12}^T A_{12} P - P^T A_{12}^T B_{12} + B_{12}^T B_{12}\right] \\
= & \quad \text{trace}\left[P P^T A_{12}^T A_{12} - 2 P^T A_{12}^T B_{12} + B_{12}^T B_{12}\right] \\
= & \quad \underbrace{\text{trace}\left[P P^T A_{12}^T A_{12}\right]}_{(2.1)} - \underbrace{2\text{trace}\left[P^T A_{12}^T B_{12}\right]}_{(2.2)} + \underbrace{\text{trace}\left[B_{12}^T B_{12}\right]}_{(2.3)}
\end{aligned}$$

The three trace terms are referred to as (2.1), (2.2), and (2.3).

Finally, for term (3),

$$\begin{aligned}
\|A_{22}P - PB_{22}\|_F^2 = & \quad \text{trace}\left[(A_{22}P - PB_{22})^T (A_{22}P - PB_{22})\right] \\
= & \quad \text{trace}\left[P^T A_{22}^T A_{22} P - B_{22}^T P^T A_{22} P - P^T A_{22}^T P B_{22} + B_{22}^T P^T P B_{22}\right] \\
= & \quad \text{trace}\left[P P^T A_{22}^T A_{22} - B_{22}^T P^T A_{22} P - P^T A_{22}^T P B_{22} + P B_{22} B_{22}^T P^T\right] \\
= & \quad \text{trace}\left[P P^T A_{22}^T A_{22}\right] - \text{trace}\left[B_{22}^T P^T A_{22} P\right] \\
- & \quad \text{trace}\left[P^T A_{22}^T P B_{22}\right] + \text{trace}\left[P B_{22} B_{22}^T P^T\right] \\
= & \quad \underbrace{\text{trace}\left[P P^T A_{22}^T A_{22}\right]}_{(3.1)} - \underbrace{2\text{trace}\left[P^T A_{22}^T P B_{22}\right]}_{(3.2)} + \underbrace{\text{trace}\left[P B_{22} B_{22}^T P^T\right]}_{(3.3)}
\end{aligned}$$

The three terms inside the brackets are referred to as (3.1), (3.2), and (3.3).

The gradient for rQAP$_2$ with hard seeds (minimization problem) is $\nabla_P f(P) = \underbrace{-2A_{21}B_{21}^T}_{(1.2)} + \underbrace{2PB_{21}B_{21}^T}_{(1.3)} - \underbrace{2A_{12}^T B_{12}}_{(2.2)} + \underbrace{2A_{12}^T A_{12}P}_{(2.1)} + \underbrace{2A_{22}^T A_{22}P}_{(3.1)} + \underbrace{2PB_{22}B_{22}^T}_{(3.3)} - \underbrace{4A_{22}^T PB_{22}}_{(3.2)}$. The numbers below the underbraces indicate which term of $h(P)$ each gradient term comes from.

For the second step of the F-W algorithm, we set $P = (1-\alpha)\hat{P} + \alpha\hat{Q}$ and maximize $h(P)$ with respect to α for $\hat{Q}$ found in the first step. We will now derive a simplification of this one-dimensional optimization problem.

The function in terms of α is

$$
\begin{aligned}
g(\alpha) = \quad & \alpha^2 \text{trace}\Big[\hat{P}^T\hat{P}\left(B_{21}B_{21}^T + B_{22}B_{22}^T\right) && (1.3+3.3)\\
& \quad + \left(A_{12}^TA_{12} + A_{22}^TA_{22}\right)\hat{P}\hat{P}^T && (2.1+3.1)\\
& \quad - 2\hat{P}^TA_{22}^T\hat{P}B_{22}\Big] && (3.2)\\
+ \quad & (1-\alpha)^2\,\text{trace}\Big[\hat{Q}^T\hat{Q}\left(B_{21}B_{21}^T + B_{22}B_{22}^T\right) && (1.3+3.3)\\
& \quad + \left(A_{12}^TA_{12} + A_{22}^TA_{22}\right)\hat{Q}\hat{Q}^T && (2.1+3.1)\\
& \quad - \hat{Q}^TA_{22}^T\hat{Q}B_{22}\Big] && (3.2)\\
+ \quad & \alpha(1-\alpha)\,\text{trace}\Big[\left(\hat{Q}^T\hat{P} + \hat{P}^T\hat{Q}\right)\left(B_{21}B_{21}^T + B_{22}B_{22}^T\right) && (1.3)+(3.3)\\
& \quad + \left(A_{12}^TA_{12} + A_{22}^TA_{22}\right)\left(\hat{Q}\hat{P}^T + \hat{P}\hat{Q}^T\right) && (2.1)+(3.1)\\
& \quad - 2\hat{P}^T\left[A_{22}^T\hat{Q}B_{22}\right] - 2\hat{Q}^T\left[A_{22}^T\hat{P}B_{22}\right]\Big] && (3.2)\\
+ \quad & \alpha\,\text{trace}\left[-2\hat{P}B_{12}^TA_{12} - 2\hat{P}^TA_{21}B_{21}^T\right] && [-(2.2)-(1.2)]\\
+ \quad & (1-\alpha)\,\text{trace}\left[-2\hat{Q}B_{12}^TA_{12} - 2\hat{Q}^TA_{21}B_{21}^T\right] && [-(2.2)-(1.2)]
\end{aligned}
$$

where the numbers at the right end of each line refer to the terms corresponding to $\|A_{21} - PB_{21}\|_F$,$\|A_{12}P - B_{12}\|_F^2$ and $\|A_{22}P - PB_{22}\|_F^2$ in the objective function. Writing $g(\alpha)$ in terms of α and (1-α),

$$g(\alpha) = c\alpha^2 + e(1-\alpha)^2 + d\alpha(1-\alpha) + u\alpha + v(1-\alpha)$$

$$
\begin{aligned}
c = &\ \text{trace}\left[\hat{P}^T\hat{P}\left(B_{21}B_{21}^T + B_{22}B_{22}^T\right) + \left(A_{12}^TA_{12} + A_{22}^TA_{22}\right)\hat{P}\hat{P}^T - 2\hat{P}^TA_{22}^T\hat{P}B_{22}\right] \\
d = &\ \text{trace}\Bigg[\left(\hat{Q}^T\hat{P} + \hat{P}^T\hat{Q}\right)\left(B_{21}B_{21}^T + B_{22}B_{22}^T\right) + \left(A_{12}^TA_{12} + A_{22}^TA_{22}\right)\left(\hat{Q}\hat{P}^T + \hat{P}\hat{Q}^T\right) \\
&\quad - \hat{P}^T\left[2A_{22}^T\hat{Q}B_{22}\right] - \hat{Q}^T\left[2A_{22}^T\hat{P}B_{22}\right]\Bigg] \\
e = &\ \text{trace}\left[\hat{Q}^T\hat{Q}\left(B_{21}B_{21}^T + B_{22}B_{22}^T\right) + \left(A_{12}^TA_{12} + A_{22}^TA_{22}\right)\hat{Q}\hat{Q}^T - 2\hat{Q}^TA_{22}^T\hat{Q}B_{22}\right] \\
u = &\ \text{trace}\left[-2\hat{P}B_{12}^TA_{12} - 2\hat{P}^TA_{21}B_{21}^T\right] \\
v = &\ \text{trace}\left[-2\hat{Q}B_{12}^TA_{12} - 2\hat{Q}^TA_{21}B_{21}^T\right]
\end{aligned}
$$

The coefficients of this polynomial in α in standard form are $g(\alpha) = a\alpha^2 + b\alpha + c$ equal $a = c + e - d$, $b = d - 2e + u - v$ and $c = e + v$.

Note that if this rQAP$_2$ formulation is further simplified by the unitary/orthogonality property of the permutation matrix, we obtain the first rQAP$_1$ formulation. When we use the constraints $P^TP = PP^T = I_l$, terms (1.3), (2.1), (3.1), and (3.3) become constant terms. The corresponding terms in $\boldsymbol{\nabla}_P f(P)$ vanish, and $\boldsymbol{\nabla}_P f(P) = -2A_{21}B_{21}^T + 2PB_{21}B_{21}^T - 2A_{12}^TB_{12} + 2A_{12}^TA_{12}P + 2(A_{22}^TA_{22}P + PB_{22}B_{22}^T - 2A_{22}^TPB_{22})$ becomes $-2*(A_{21}B_{21}^T + A_{12}^TB_{12} + A_{22}PB_{22}^T + A_{22}^TPB_{22})$, which is the -2 times gradient for the rQAP$_1$ formulation. It is interesting how this extra constraint affects the convergence properties of the Frank-Wolfe algorithm. This question is investigated in the comparison of the rQAP$_1$ and rQAP$_2$ formulations.

11.2.4 The comparison of the rQAP$_1$ against the alternative formulation rQAP$_2$

Although the two formulations are equivalent in the domain of permutation matrices and the global extrema of the two functions are the same, we expect different convergence properties. In particular, the extra terms in the gradient of rQAP$_2$, which vanish only for orthogonal matrices, provide a constant source of perturbation. The conclusion obtained from the literature on stochastic optimization [54] is that, under some conditions, injecting noise to the gradient would help convergence by overcoming local extrema. However, for the iterative algorithm to converge, the noise has to vanish to negligible levels. We have no evidence that these extra terms are small, whenever $P^{(i)}$ is in the neighborhood of the solution. Therefore, rQAP$_2$ might have convergence problems due to the constant source of perturbation provided by the extra terms. rQAP$_1$, on the other hand, will converge to a local solution, that is not necessarily a permutation matrix.

We make a performance comparison between rQAP$_1$ and rQAP$_2$, by matching the same pairs of bitflipped graphs subsubsection 11.2.2.1. We consider both the true matching ratios and number of iterations until convergence. The experiment in the subsubsection 11.2.2.1 was repeated with both rQAP$_1$ and rQAP$_2$. For the same pairs of graphs, the fraction of non-seed vertices correctly matched were computed for both methods. The results are plotted in Figure 11.4

The distinction between the two formulations are most prominently visible for $p_{pert} = 0.35$. Note that for a small number of hard seeds, the rQAP$_2$ is slightly better, whereas for a large number of hard seeds, the rQAP$_1$ formulation is clearly better. This observations are valid for other p_{pert} values, also, albeit to a smaller degree.

The average number of iterations of the Frank-Wolfe algorithm until termination for the two formulation are shown in Figure 11.5.

Our conclusion is that our expectations for the two formulations are warranted: the rQAP$_2$ converges slower (or does not converge but stays within the neighborhood of the extrema), whereas the rQAP$_1$ converges in very few steps. When the number of hard seeds is small (which corresponds to a lower number of constraints for P and a higher incidence of local minima near the true solution), the rQAP$_2$ formulation is slightly better than the rQAP$_1$ formulation.

A natural follow-up to the previous inquiry is whether one can have the best of both worlds by forming a hybrid of the two formulations: first, we start by minimizing the rQAP$_2$ function until the current iterate of the solution is relatively close to the true solution, and we continue by maximizing the rQAP$_1$ function.

11.2.5 A hybrid formulation: FAQ programming with a smooth transition from rQAP$_2$ to rQAP$_1$

For this hybrid form of the FAQ algorithm, we weight the terms that differ between the gradients of rQAP$_2$ and rQAP$_1$ by a decreasing weight r. $\nabla_P h(P) = r*\{2PB_{21}B_{21}^T + 2A_{12}^TA_{12}P + A_{22}^TA_{22}P + PB_{22}B_{22}^T\} - 2A_{21}B_{21}^T - 2A_{12}^TB_{12} - 4A_{22}^TPB_{22}$. As $r \to 0$, the gradient expression at each step of the F-W algorithm approaches $-2 * (A_{21}B_{21}^T + A_{12}^TB_{12}) + A_{22}PB_{22}^T + A_{22}^TPB_{22})$, which is -2 times the gradient in rQAP$_1$. We let $r = 0.5 - \frac{\tan((i-(i_{end}/2)))}{\pi}$, and thus, as the iteration counter, i, goes from 1 to i_{end}, r goes from 1 to 0. This hybrid formulation will behave like rQAP$_2$ for the initial iterations of F-W algorithm and will start to behave like rQAP$_1$ as i approaches i_{end}.

We find that the hybrid approach is always better than rQAP$_2$ formulation and mostly better than rQAP$_1$. For large values of m(the number of seeds), and for large values of p_{pert}, rQAP$_1$ may be slightly better than hybrid. This suggests some tuning may be necessary for the hybrid approach and further investigations into why rQAP$_2$ is worse than rQAP$_1$ for large number of seeds.

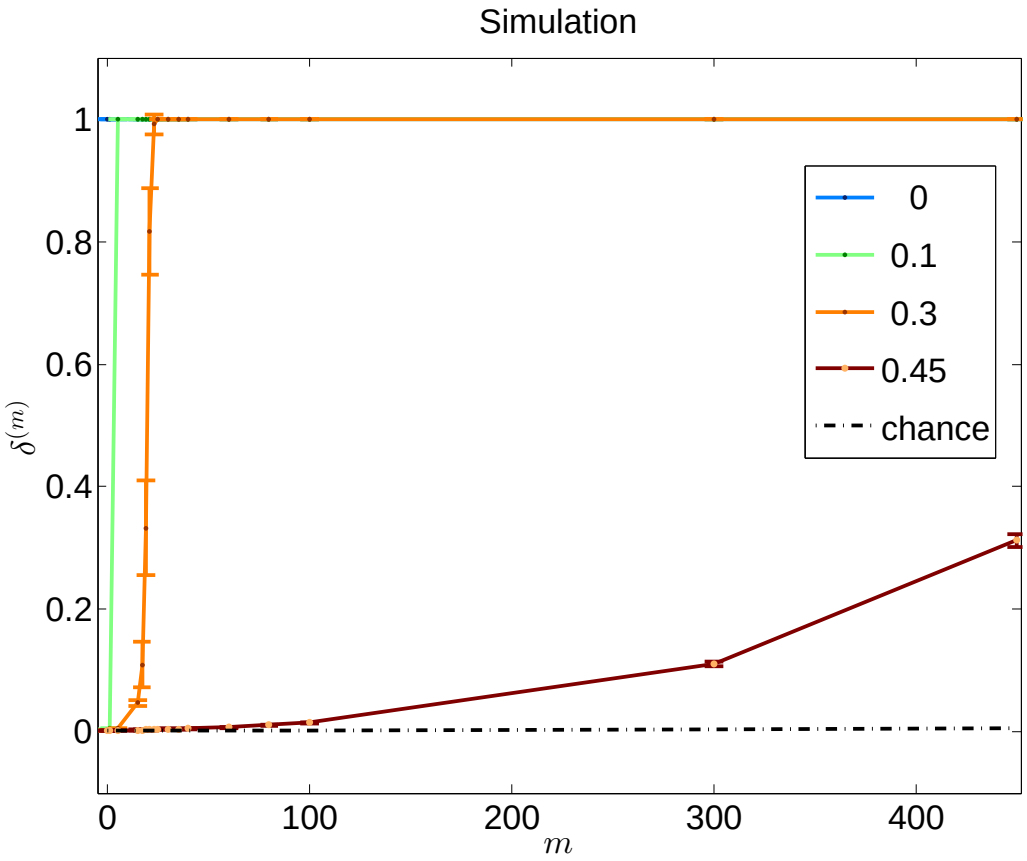


Figure 11.1: $\delta^{(m)}$ vs m for $n = 600$ vertices. The error bars represent two times the standard error of the mean of the true match ratio. Different colors listed in the legend correspond to different p_{pert} values.

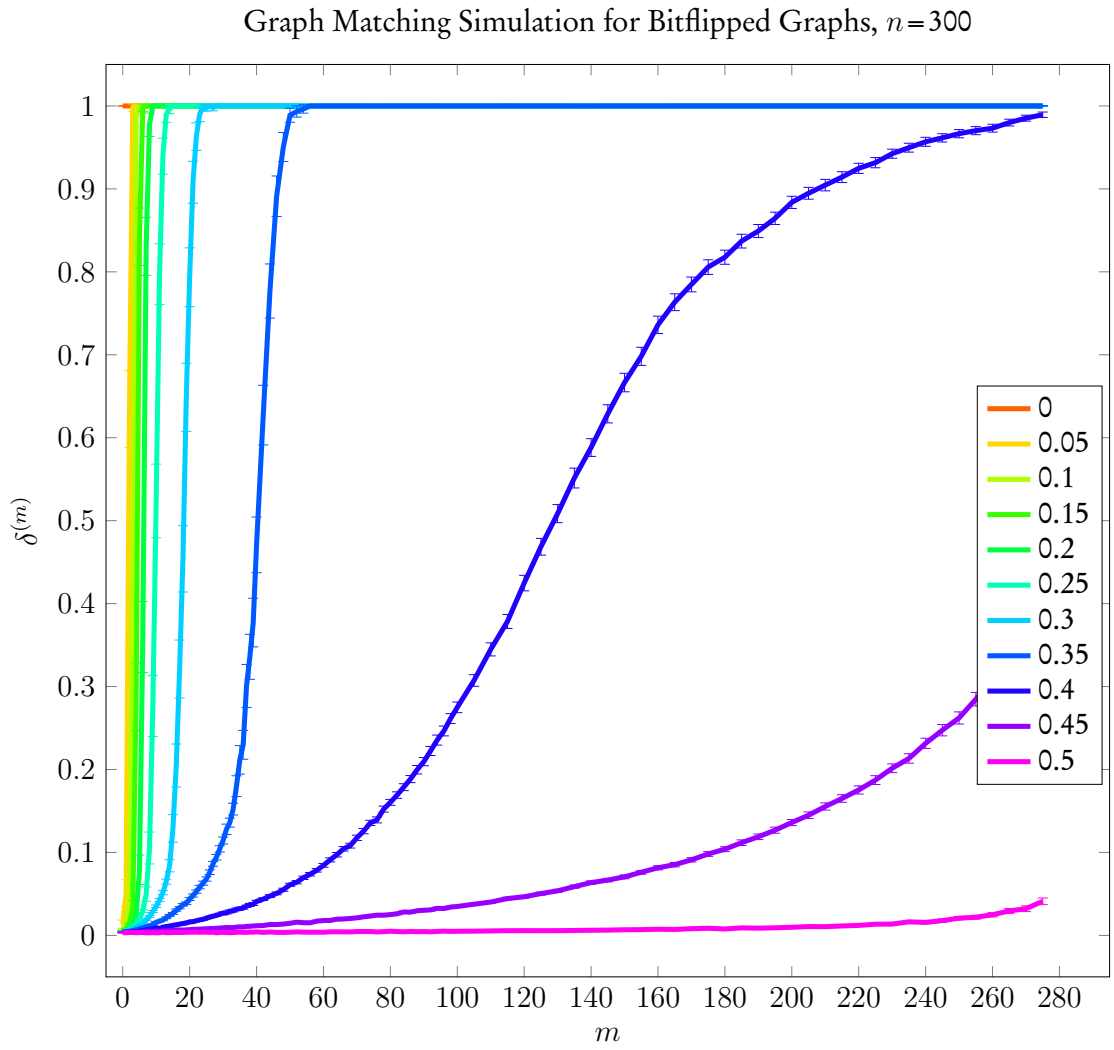


Figure 11.2: $\delta^{(m)}$ vs m for $n = 300$ vertices. The error bars represent two times the standard error of the mean of the true match ratio. Different colored lines correspond to different p_{pert} values.

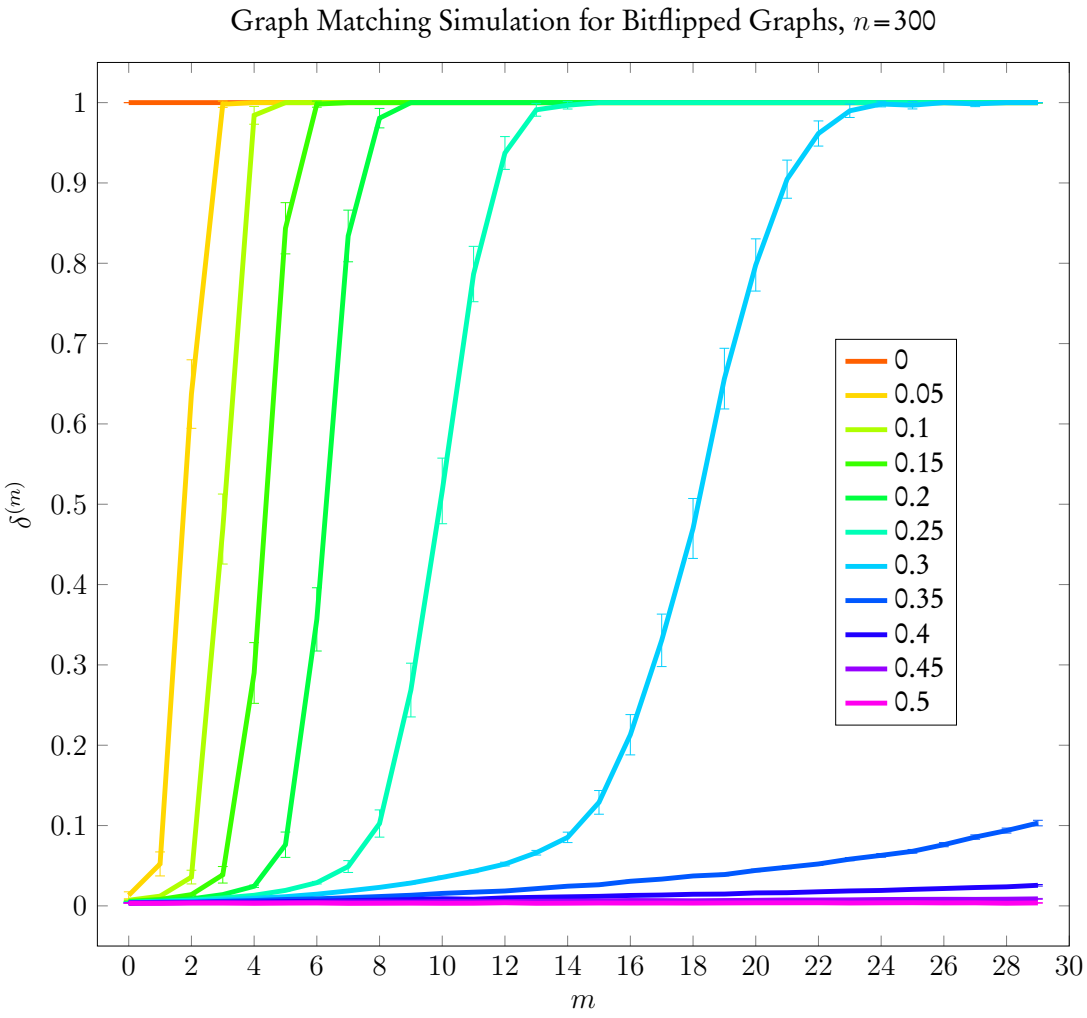


Figure 11.3: $\delta^{(m)}$ vs m for $n = 300$ vertices. This plot includes a portion of Figure 11.2, which includes the x-axis from $m = 0$ to $m = 29$. The error bars represent two times the standard error of the mean of the true match ratio. Different colored lines correspond to different p_{pert} values.

Figure 11.4: Fraction of correctly matched non-seed vertices for m seeds (x-axis). Different colors correspond to different p_{pert}. Solid and dashed lines correspond to rQAP$_1$ and rQAP$_2$ solutions, respectively, for the matching problem.

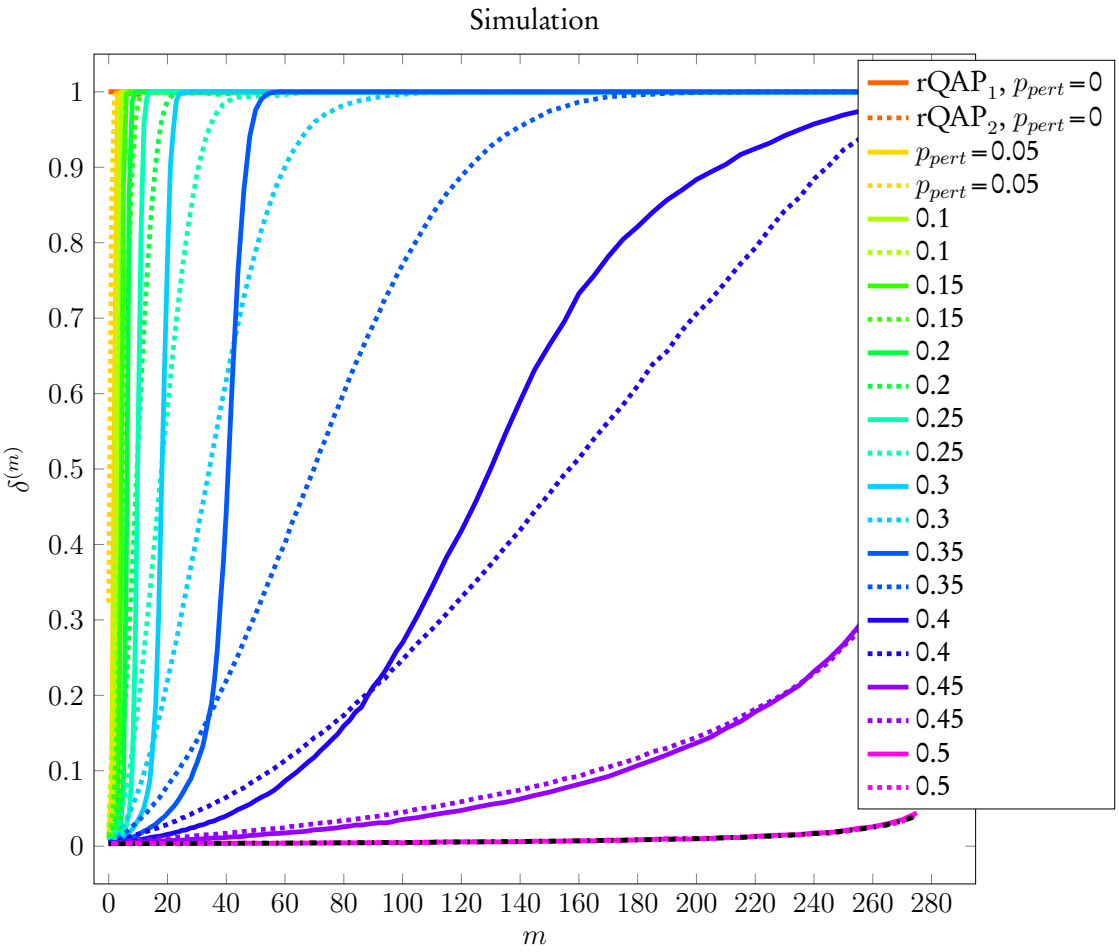


Figure 11.5: Number of Iterations for the $rQAP_1$ and $rQAP_2$ formulations to converge

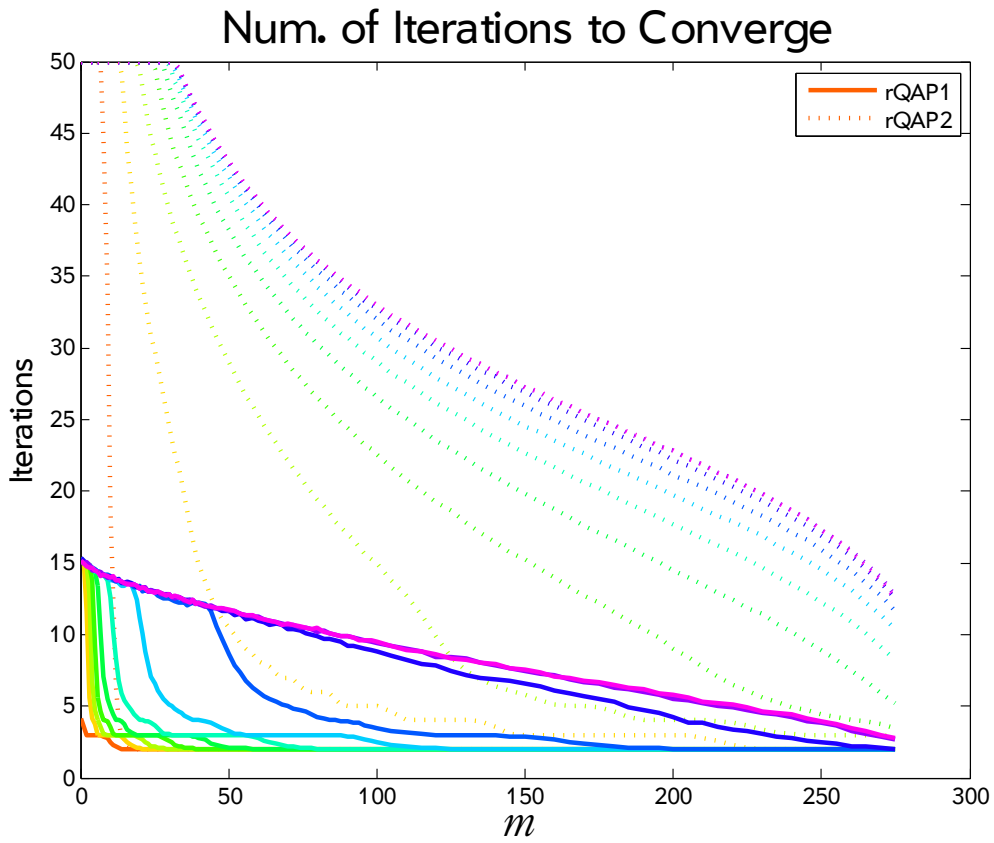
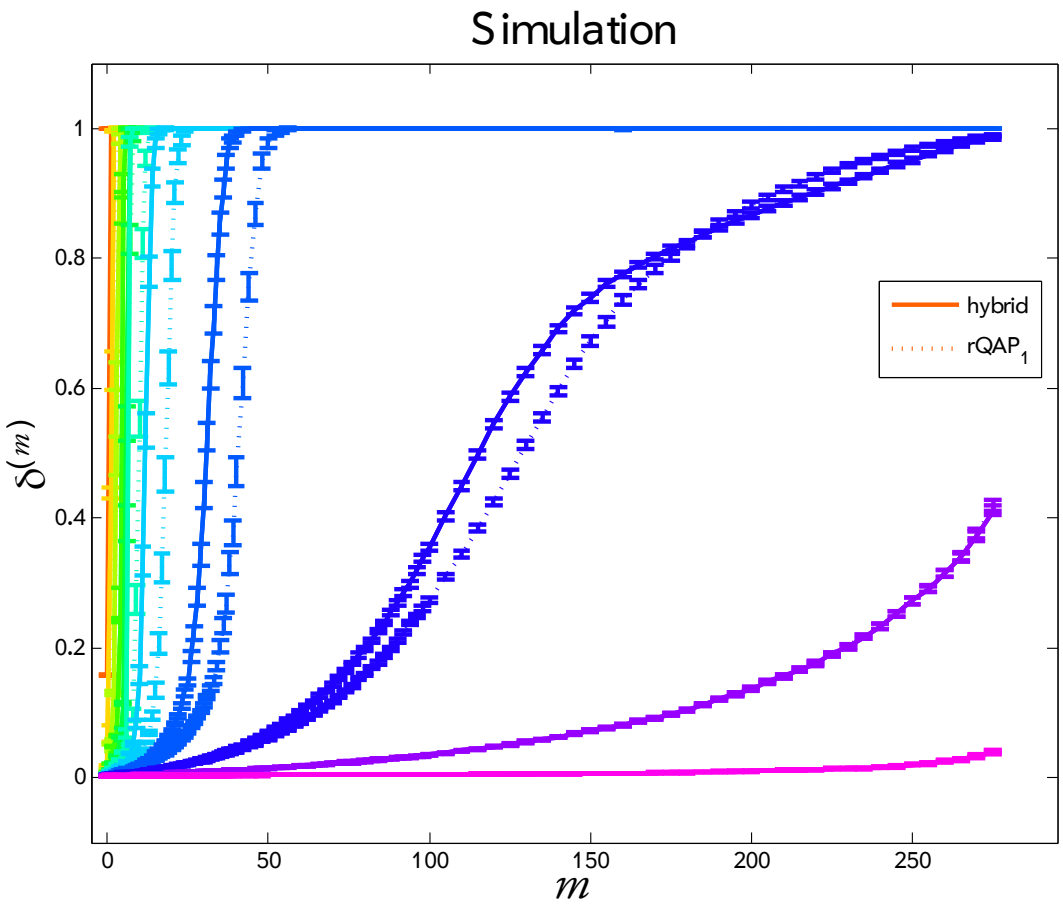


Figure 11.6: Fraction of correctly matched non-seed vertices for m seeds (x-axis). Different colors correspond to different p_{pert}. Dashed and solid lines correspond to rQAP$_1$ (FAQ) and the hybrid of the rQAP$_2$ and rQAP$_1$ (hybrid) solutions, respectively, for the matching problem.



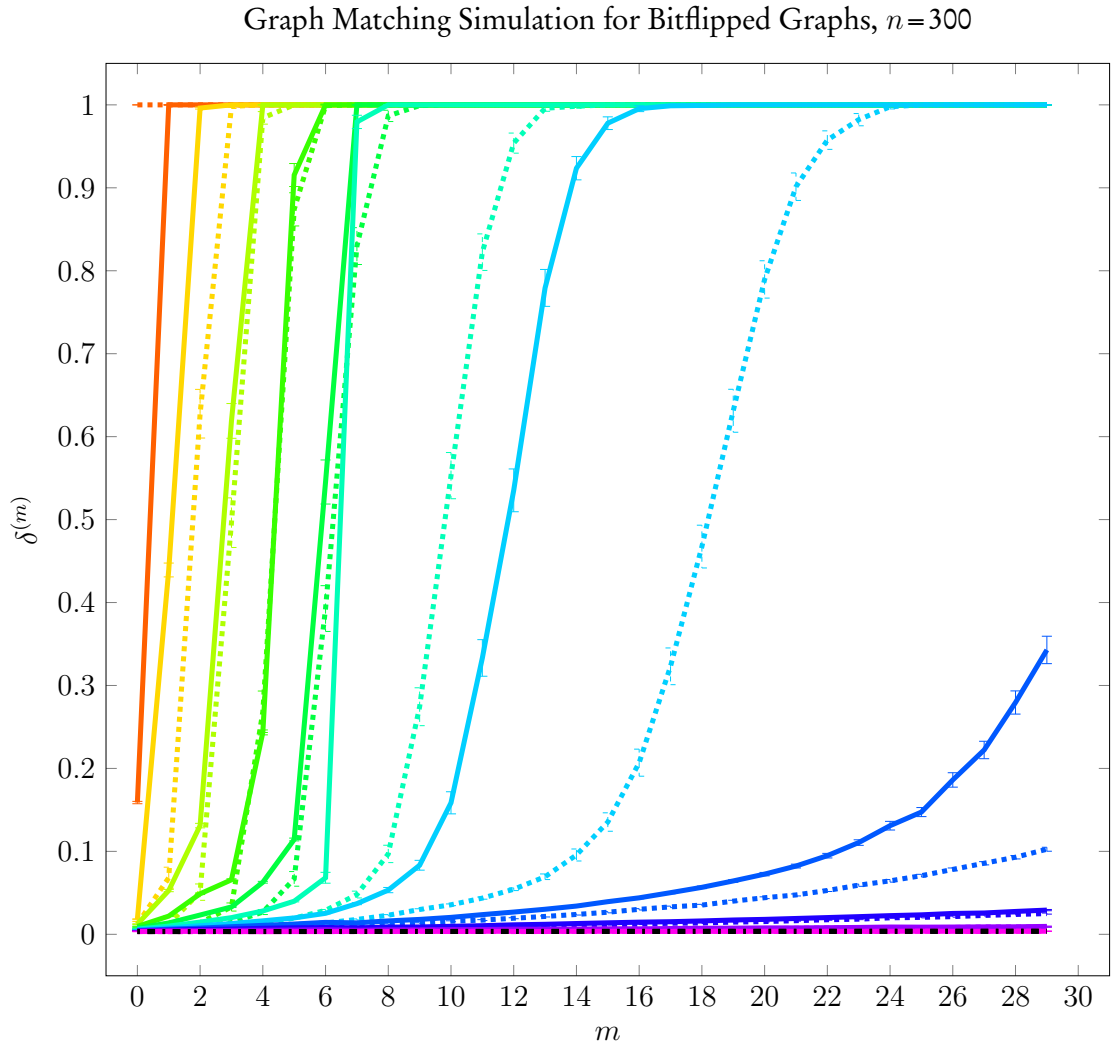


Figure 11.7: Same plot as Figure 11.6 restricted to $m < 30$ seeds. Fraction of correctly matched non-seed vertices for m seeds (x-axis) where $m < 30$. Different colors correspond to different p_{pert}. Dashed and solid lines correspond to rQAP$_1$ (FAQ) and the hybrid of the rQAP$_2$ and rQAP$_1$ (hybrid) solutions, respectively, for the matching problem.

Chapter 12

The Joint Optimization of Fidelity and Commensurability solution to Seeded Graph Matching

12.1 Overview

We first explain the relevance of the JOFC approach to the graph matching problem. The task of finding vertex correspondences is similar to detecting matched pairs4 in that both of the tasks require the quantification of a distance between vertex pairs in different graphs. A joint commensurate representation of the vertices of the two graphs can be used to compute these distances between vertex pairs.

Following our dissimilarity-centric approach, a dissimilarity matrix can be computed

for each graph using a dissimilarity measure for graph vertices. The choice of the dissimilarity measure is an important issue and we will consider this issue during our investigations. If we treat the known corresponding vertices in the seeded graph matching problem as matched, we can form an omnibus matrix by using dissimilarity matrices from the two graphs. We impute the off-diagonal matrix L in the omnibus matrix the usual way (the matched dissimilarities are zeros, other between-condition dissimilarities are missing). The joint embedding of the omnibus matrix yields the vertices of two graphs in a commensurate space. Therefore, the JOFC approach can be used to determine the pairwise distances in the commensurate space between the vertices of two graphs.

The next step is to use the pairwise distances as costs to find the optimal one-to-one assignment using the Hungarian algorithm [52]. The Hungarian algorithm finds an optimal matching between two sets of vertices such that the total cost, which is the sum of the pairwise distances of matched nodes, is minimized. This matching provides the bijection solution for the seeded graph matching problem.

12.2 Joint Embedding of Graphs via JOFC for Seeded Graph Matching

Now, we describe in detail how we use JOFC embedding for seeded graph matching. We begin by jointly embedding our two graphs, $\mathbb{G}_1$ and $\mathbb{G}_2$, into a common Euclidean space. Let $\Delta_1 \in \mathbb{M}_{n\times n}$ and $\Delta_2 \in \mathbb{M}_{n\times n}$ be two dissimilarity matrices computed by the

application of the dissimilarity measure to the vertices of the two graphs. Because we have two separate graphs, we have two conditions, and the default assumption is that we do not have between-graph (between-condition) dissimilarities. We will assume that prior to embedding, the dissimilarities have been normalized to have the same scale of magnitude.

Without loss of generality, we can assume that the vertices in both graphs are labeled as integers from 1 to n and that the labeling is consistent with the true correspondence of vertices from different graphs. Suppose that we have m, $0 \leq m < n$ seeds. Again, without loss of generality, let the seeded vertices be labeled as the first m vertices in both graphs, $S_1 = \{1, 2, \ldots, m\}$ and $S_2 = \{1, 2, \ldots, m\}$, so that

$$\Delta^{(1)} = \begin{array}{c} \\ S_1 \\ U_1 \end{array} \begin{array}{c} \begin{array}{cc} S_1 & U_{in} \end{array} \\ \begin{pmatrix} \Delta^{(1)}_{in,in} & \Delta^{(1)}_{in,oos} \\ \Delta^{(1)}_{oos,in} & \Delta^{(1)}_{oos,oos} \end{pmatrix} \end{array}, \; \Delta^{(2)} = \begin{array}{c} \\ S_2 \\ U_2 \end{array} \begin{array}{c} \begin{array}{cc} S_2 & U_2 \end{array} \\ \begin{pmatrix} \Delta^{(2)}_{in,in} & \Delta^{(2)}_{in,oos} \\ \Delta^{(2)}_{oos,in} & \Delta^{(2)}_{oos,oos} \end{pmatrix} \end{array}.$$

Note that the dissimilarities between seed vertices correspond to the matched in-sample dissimilarities that we considered in the match detection task 4. Because these seeds provide the known correspondences (matched observations in the match detection task), we embed them using the in-sample JOFC embedding methodology we introduced in 4.1. That is, we find a $2m \times d$ configuration matrix

$$\mathbb{X} = \begin{bmatrix} \mathbb{X}_1 \\ \mathbb{X}_2 \end{bmatrix}$$

such that the $2m \times 2m$ distance matrix $\mathcal{D}(\mathbb{X})$ is as close as possible to the omnibus

dissimilarity matrix,

$$M = \begin{bmatrix} \Delta^{(1)}_{in,in} & L \\ L^T & \Delta^{(2)}_{in,in} \end{bmatrix}.$$

The remaining non-seed vertices are embedded using OOS embedding with respect to the embedded seeds, i.e., we seek the configuration $\hat{\mathbf{Y}}$ consisting of $2(n-m)$ points in $\mathbb{R}^{d'}$ that correspond to the non-seed vertices of G_1 and G_2. The $n-m$ non-seed vertices of G_1, with the embedded coordinates $\{\hat{y}^{(1)}_{(m+1)}, \ldots, \hat{y}^{(1)}_{(n)}\}$, and the $n-m$ non-seed vertices of G_2, with the embedded coordinates $\{\hat{y}^{(2)}_{1}, \ldots, \hat{y}^{(2)}_{(n-m)}\}$, where $\hat{y}^{(k)}_{i}, 1 \leq i \leq (n-m), k \in 1, 2$ minimize the stress function:

$$\sigma(\mathbf{Y}) = \sum_{s=1}^{m}\sum_{t=m+1}^{n} W^{(1)}_{in,oos}(s,t-m)\left(d(X^{(1)}_s,\hat{y}^{(1)}_t) - \Delta^{(1)}_{in,oos}(s,t-m)\right)^2 \tag{12.1}$$

$$+ \sum_{s=m+1}^{n}\sum_{t=1}^{m} W^{(2)}_{oos,in}(s-m,t)\left(d(X^{(2)}_s,\hat{y}^{(2)}_t) - \Delta^{(2)}_{oos,in}(s-m,t)\right)^2 \tag{12.2}$$

$$+ \sum_{s=m+1}^{n}\sum_{t=1}^{m} W^{(2)}_{oos,oos}(s-m,t-m)\left(d(\hat{y}^{(1)}_s,\hat{y}^{(1)}_t) - \Delta^{(1)}_{oos,oos}(s-m,t-m)\right)^2 \tag{12.3}$$

$$+ \sum_{s=m+1}^{n}\sum_{t=1}^{m} W^{(2)}_{oos,oos}(s-m,t-m)\left(d(\hat{y}^{(2)}_s,\hat{y}^{(2)}_t) - \Delta^{(2)}_{oos,oos}(s-m,t-m)\right)^2 \tag{12.4}$$

$$+ \sum_{s=m+1}^{n}\sum_{t=1}^{m} W^{(1,2)}_{oos,oos}(s-m,t-m)\left(d(\hat{y}^{(1)}_s,\hat{y}^{(2)}_t) - \delta(\hat{y}^{(1)}_s,\hat{y}^{(2)}_t)\right)^2 \tag{12.5}$$

$$+ \sum_{s=m+1}^{n}\sum_{t=1}^{m} W^{(1,2)}_{oos,in}(s-m,t)\left(d(\hat{y}^{(1)}_s,X^{(2)}_t) - \delta(\hat{y}^{(1)}_s,X^{(2)}_t)\right)^2 \tag{12.6}$$

$$+ \sum_{s=1}^{m}\sum_{t=m+1}^{n} W^{(1,2)}_{in,oos}(s,t-m)\left(d(X^{(1)}_s,\hat{y}^{(2)}_t) - \delta(X^{(1)}_s,\hat{y}^{(2)}_t)\right)^2 \tag{12.7}$$

where $W^{(k)}_{a,b}$ and $W^{(k_1,k_2)}_{a,b}$ (for $a, b \in \{`in', `oos'\}$) are the weights for dissimilarities between in-/out-of-sample observations in k^{th} (or k_1^{th} and k_2^{th}) conditions.

Note that 12.5, 12.6 and 12.7 involve dissimilarities $\delta(\cdot^{(1)}, \cdot^{(2)})$ between different conditions, which are generally not available. Whereas the dissimilarities in 12.6 and 12.7 can be imputed via known dissimilarities, the dissimilarities in 12.5 cannot be imputed in any way. In fact, if these dissimilarities in 12.5 between OOS observations in different conditions were known, we could provide a solution to the assignment task because we would have the assignment cost of any two OOS observations in two different condi-

tions. The solution of the linear assignment problem with these costs would give us the matching of vertices from different graphs.

12.3 and 12.4 contain dissimilarities between OOS observations in the same condition. They can be used or ignored in the joint embedding, depending on whether one wishes to embed OOS observations all together or one at a time. If between-condition OOS dissimilarities are ignored, the OOS embedding function $\sigma(\mathbf{Y})$ is separable with respect to different $\hat{y}_s^{(k)}, \quad s \in \{m+1, \dots n\}, k \in \{1, 2\}$. Then, we would obtain the same embedding configuration if the OOS observations $\hat{y}_s^{(k)}$ are embedded one at a time or all at once.

Once we find a minimum configuration $\hat{\mathbf{Y}}$, we compute the pairwise distances between the points $c_{(s-m)(t-m)} = d(\hat{y}_s^{(1)}, \hat{y}_t^{(2)}), s \in \{(m+1), \dots n\}, t \in \{(m+1), \dots n\}$, which correspond to the entries of the off-diagonal block matrix of the distance matrix, $\mathcal{D}(\hat{\mathbf{Y}})$. This block matrix provides the assignment cost matrix C (whose $(i,j)^{th}$ entry is c_{ij}) for the linear assignment problem, which is to minimize trace$(A^T C) = \sum_{i,j \in \{(1), \dots n-m\}} a_{ij} * c_{ij}$ with respect to the permutation matrix A (whose $(i,j)^{th}$ entry is a_{ij}).

In summary, we provide a solution for the seeded graph matching problem by jointly embedding the two graphs, followed by a solution of the linear assignment problem where the assignment costs are the distances between embedded points.

While it is plausible that the JOFC approach can also be used to solve the seeded graph matching (SGM) problem, it is not obvious that it can compete with the FAQ

algorithm, which is specifically formulated to solve the seeded approximate graph matching problem. Why, then, should one choose to use the JOFC approach for SGM? One of the many problems with the analysis of real data is that the graph representation of real data is not always well-defined and that the correspondence of vertices may be ambiguous, one-to-many, or many-to-many. In such situations, we would prefer a robust algorithm that would still match seeded graphs with satisfactory performance. The FAQ algorithm, in the form that we have presented, cannot handle such pathologies, and significant changes must be made to the FAQ algorithm before it can handle them.

Our simulations using the correlated Erdős-Renyi graphs and experiments with real graph data are tailored for a comparison of the two approaches. Some of these results are presented in section 12.3. If the true match ratios of the assignments given by the JOFC approach is at least as high as the ones given by the FAQ algorithm, we can conclude that JOFC is reasonably competitive with the FAQ algorithm for seeded graph matching. We compared the algorithms whenever both of the approaches were feasible for the problem size (the running times are acceptable).

This application of the JOFC approach for seeded graph matching is also presented in [55] along with our experimental results.

12.2.1 Dissimilarity Measures for Vertices

One useful property of dissimilarity representation is that the structure of the data is irrelevant once an appropriate dissimilarity function for the data is available. That is

why the JOFC approach is directly applicable once an appropriate dissimilarity measure has been chosen.

There are many dissimilarities that can be defined between vertices in graphs. We assume that an appropriate dissimilarity measure between the vertices is available to us. In our experiments, we will use five different dissimilarities/distances between vertices in a graph:

- the shortest path on the unweighted graph whose adjacency matrix is available,
- the shortest path on a weighted version of the graph whose weight matrix is available,
- the diffusion distance between vertices on the (unweighted) graph,
- the weighted extension of the Czekanowski-Dice dissimilarity [56, 57], which simplifies to the original Czekanowski-Dice dissimilarity in the case of unweighted graphs (the C-D dissimilarity quantifies the local similarity of two vertices in a graph),
- the expected commute time for random walks on the graph.

Remark Note that these dissimilarities are defined between vertices of the same graph. Because the dissimilarities between vertices of different graphs are not available, we must resort to the same imputation workarounds as we did for the JOFC embedding in 4.1. We would again choose 0 for the dissimilarities between matched vertices and then either impute the remaining unknown dissimilarities or ignore them in the embedding.

12.3 Demonstrations

We perform SGM simulations with graphs generated according to a paired Erdős-Renyi graph model described in subsubsection 11.2.2.1 and experiments on real-life graphs for both the FAQ algorithm and the JOFC approach. The performance measure that we consider is the true match ratio: the number of true matchings of vertices divided by the number of pairs of vertices.

12.3.1 Simulations

We first present our exploratory simulations to test the JOFC approach and determine the reasonable choices for the dissimilarity measure and w (the parameter that controls the Fidelity-Commensurability tradeoff). We will also see how sensitive the results are for different choices of the dissimilarity measure and different w values.

We consider the same Erdős-Renyi correlated graphs as in the FAQ simulations introduced in subsubsection 11.2.2.1.

The probability of flipping an entry of the adjacency matrix is the perturbation parameter p_{pert}, which is the variable on the x-axis. The performance measure is the proportion of true matches to the number of matches. Note that under chance, the expected number of true matches is 1, as shown with the dashed line. In this particular simulation, we consider the JOFC approach with classical and raw stress variants and compare the performance of each in small graphs. For JOFC with the weighted raw stress function,

we set $w = 0.8$. The joint embedding with cMDS is compared with the JOFC approach to figure out how the performance measure is sensitive to the Fidelity-Commensurability tradeoff. p_{pert} varies from 0 to 1 in increments of 0.1.

Our first observation from the plot is that for low amounts of perturbation, JOFC performs satisfactorily, that is, most of the test vertices are matched correctly. Various dissimilarity measures can be chosen for the dissimilarity matrix. The appropriate dissimilarity measure might depend on the degree of the distribution and the size of the graph. Figure 12.1 shows that some dissimilarity measures result in significantly different behavior as p_{pert} changes.

As the perturbation parameter becomes larger, (for all dissimilarity measures) the performance of JOFC degrades until it is indistinguishable from random chance at $p_{pert} = 0.5$. For this p_{pert} value, there is no edge correlation between the two graphs because the mutual information between A_{ij} and B_{ij} is `0`. At that p_{pert} value, we expect, on average, the same number of true matches as that obtained by random chance. Further, this p_{pert} value means that the dissimilarity between truly matched vertices, say with the shortest path distance as the dissimilarity measure, is even larger than you would expect by chance, which means an even smaller number of true matches.

As p_{pert} approaches 1, $\mathbb{G}_2$ approaches the complement of $\mathbb{G}_1$. An interesting feature of the plot is the U-shape of the curve for some of the dissimilarities. This invariancy with respect to the complement of the graph should be investigated further.

Other than the dissimilarity measure, the embedding methodology could also have

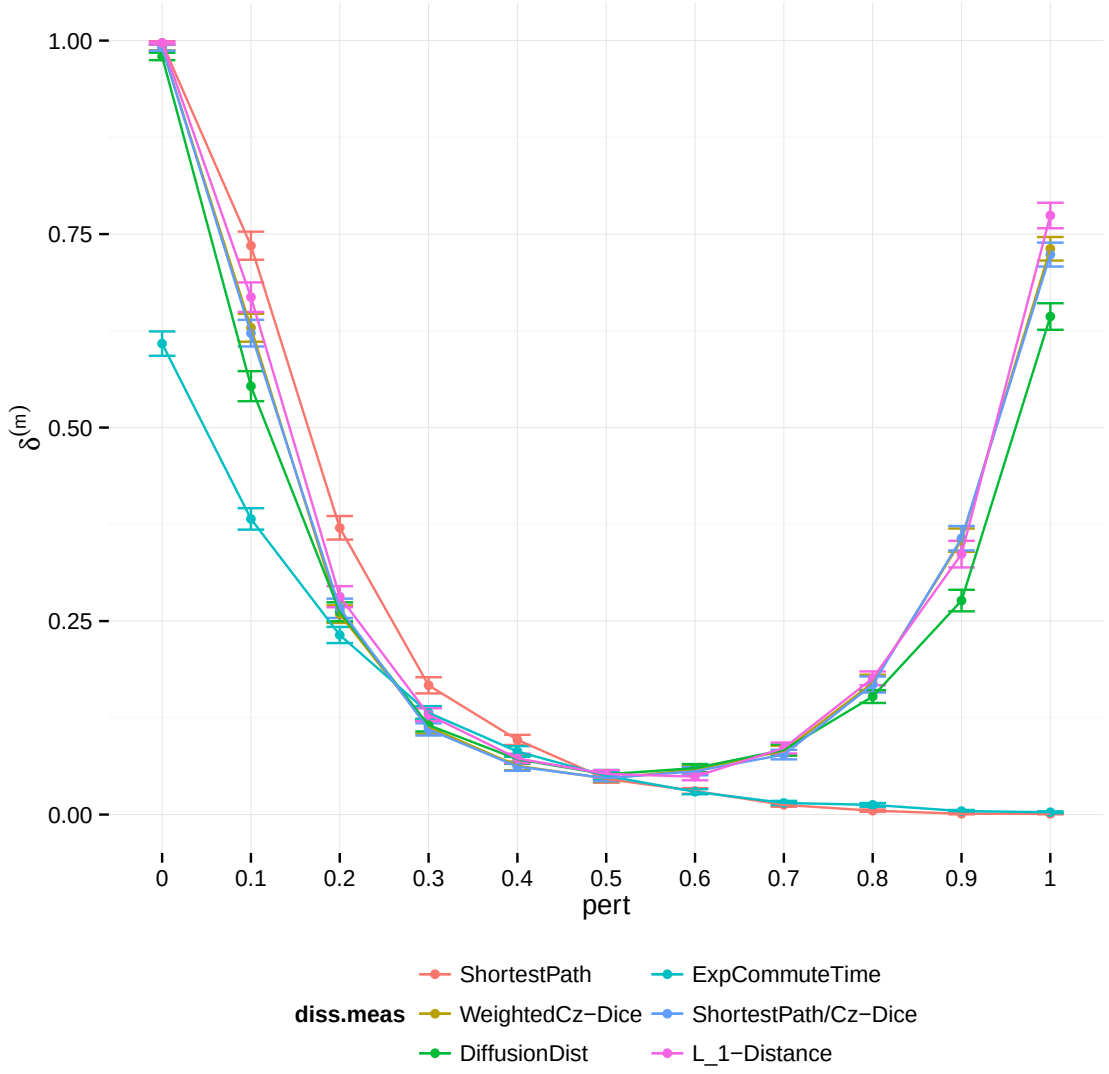


Figure 12.1: The matching ratio for seeded graph matching via JOFC is different for different dissimilarity measures. For $n = 50$ vertices and $m = 30$ seeds, the true matching ratio is plotted against the perturbation parameter p_{pert}. Note the U-shape of some of the dissimilarities.

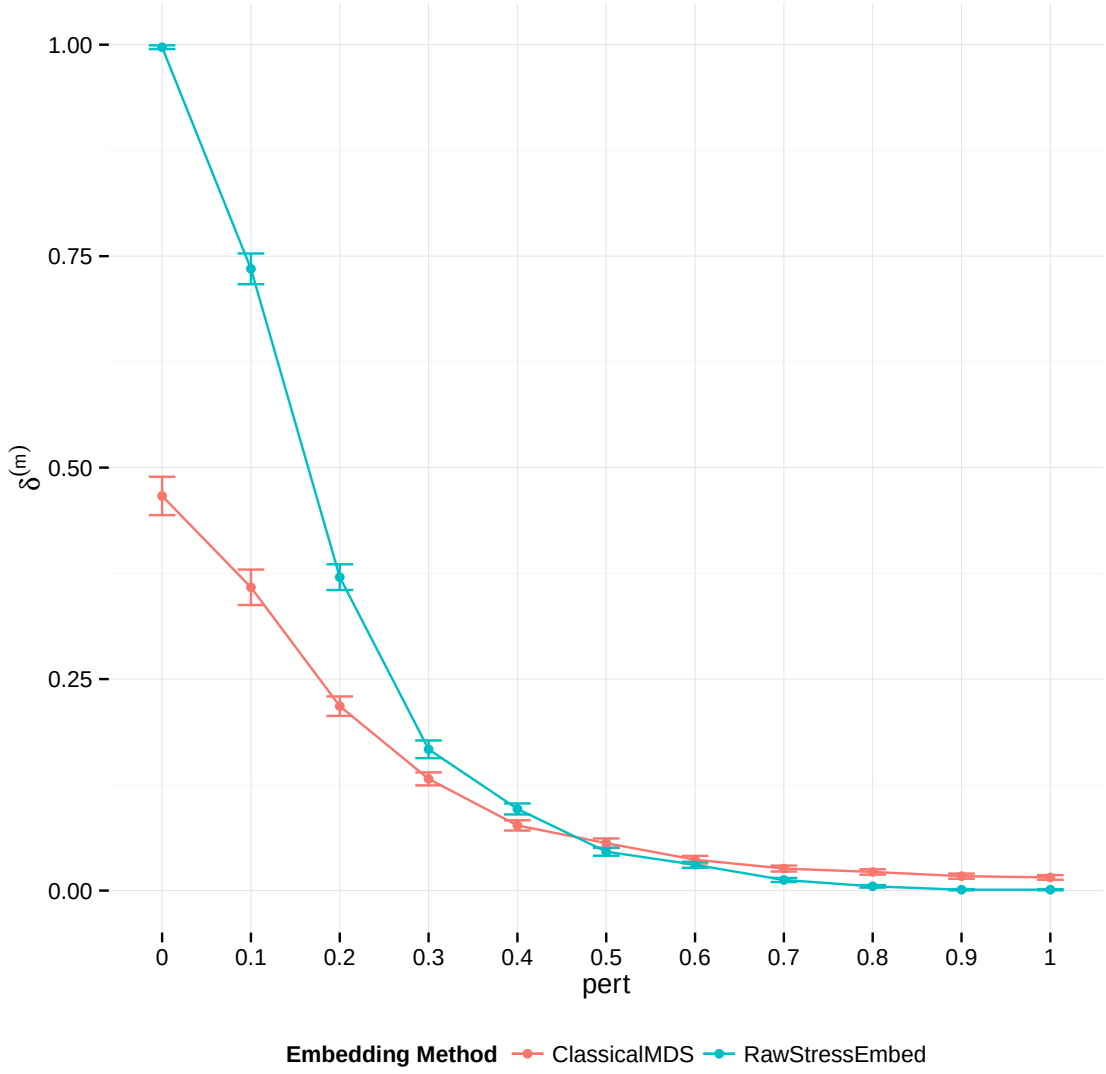


Figure 12.2: The matching ratio for seeded graph matching via JOFC is compared with classical MDS embedding with OOS extension. For $n = 50$ vertices and $m = 30$ seeds, the true matching ratio is plotted against the perturbation parameter p_{pert}.

a large impact on performance. Our simulations indicate that the performance of the JOFC embedding is significantly better than that of cMDS, as shown in Figure 12.2.

The graph in Figure 12.3 shows the effect of the weight parameter of stress w on the probability of true matches. We note that in this graph matching, for the setting using the shortest path distance as the dissimilarity measure and for the w values we tested, there is no significant difference between the true matching ratios. The choice of the parameter w is thus not necessarily critical for performance in all data settings.

There are a lot of interesting questions to ponder about the number of known correspondences, such as the following:

- How many known correspondences are necessary for satisfactory performance for graphs of a given size?
- Are there any "elbows" in the curve for the "match ratio" vs the number of known correspondences, after which the cost of more correspondences is not justified by the accompanying increase in "match ratio"?

. We attempt to answer these questions using the Erdős-Renyi graph pair model that we introduced along with some real-world graphs.

Figure 12.4 shows the "match ratio" plotted against the number of "seeds" for the bitflip simulations (the data are generated using the Erdős-Renyi graph pair model) using the Czekanowski-Dice dissimilarity measure. These results, along with the previous simulations, suggest that even with the perturbation, when a portion of the correspon-

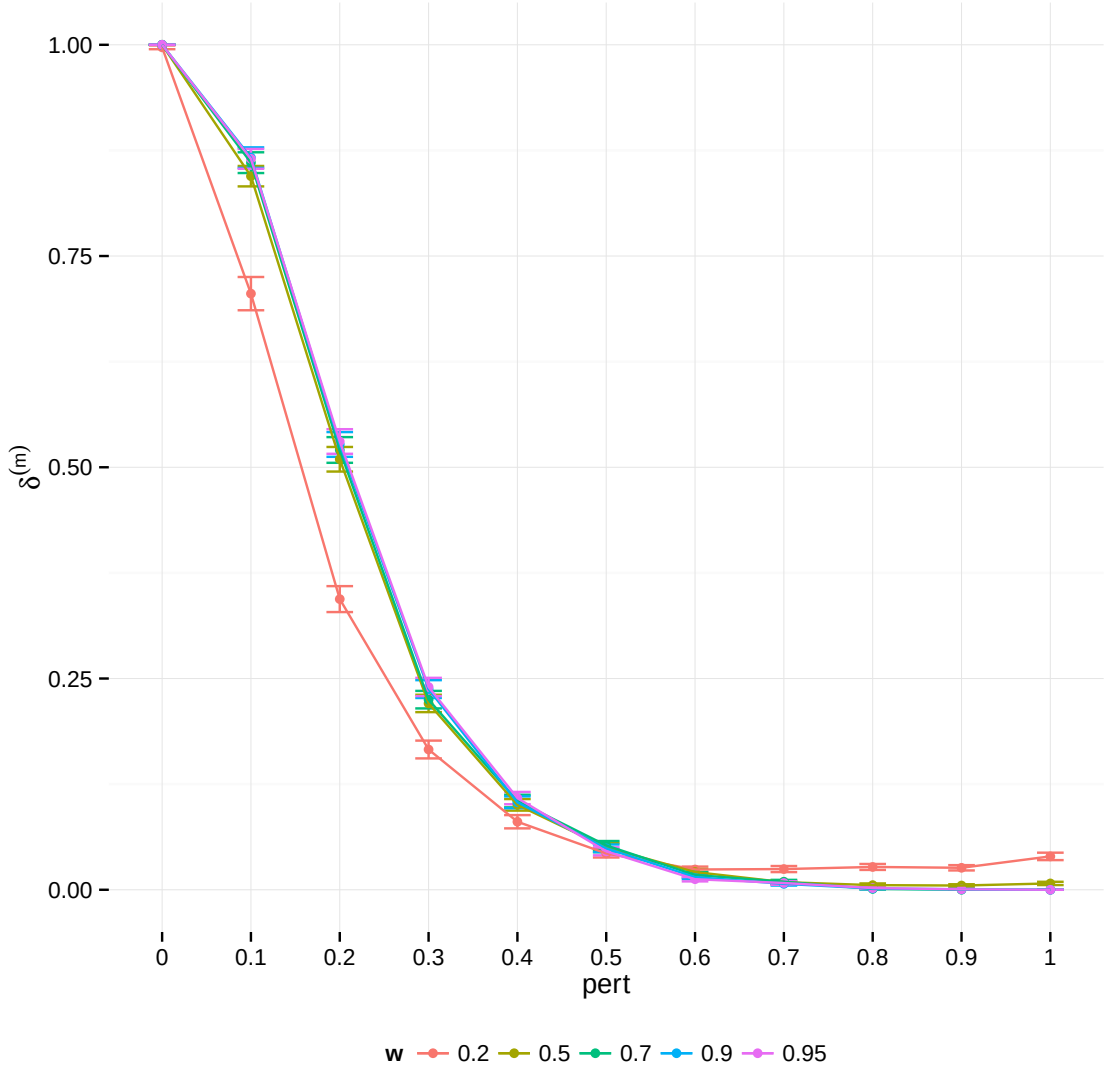


Figure 12.3: Seeded Graph matching performance (The true matching ratio) via JOFC for different w values (Fidelity-Commensurability tradeoff parameter). For $n = 50$ vertices and $m = 30$ seeds, the true matching ratio is plotted against the perturbation parameter p_{pert}.

dences are known, it is possible to recover most of the remaining correspondences using JOFC embedding of the pair of graphs. We note that the number of correspondences that must be known before the proportion of true matches are satisfactory depends on various factors, such as the size and average connectivity of the graphs, the connectivity of the seed vertices, the dissimilarity measure, and the amount of perturbation between the two graphs. Further investigations into these factors could be fruitful.

12.3.2 Experiments on real data

12.3.2.1 C. elegans connectome

We consider two connectivity graphs of 279 neurons of the nematode *Caenorhabditis elegans* as an example of real-world graph data. The two conditions correspond to the two ways of measuring connectivity among neurons $\mathbb{G}_c$ and $\mathbb{G}_g$. The connectivity in the first connectome is defined by chemical synapses, a directed connection between two cells. This connectome is represented by a weighted graph, where the weights correspond to the number of synapses identified in images of C. elegans specimens. The weight matrix for the first connectivity type is A_c, which is not symmetric, has values between 0 and 37, and is relatively sparse (has 2194 nonzero entries). The second connectivity type forms an unweighted graph $\mathbb{G}_g$ with the adjacency matrix A_g and is defined by the existence of gap junctions between neurons. $\mathbb{G}_g$ is even sparser (1031 nonzero entries) than $\mathbb{G}_c$.

We remove isolated vertices from the two graphs and keep the vertices that are con-

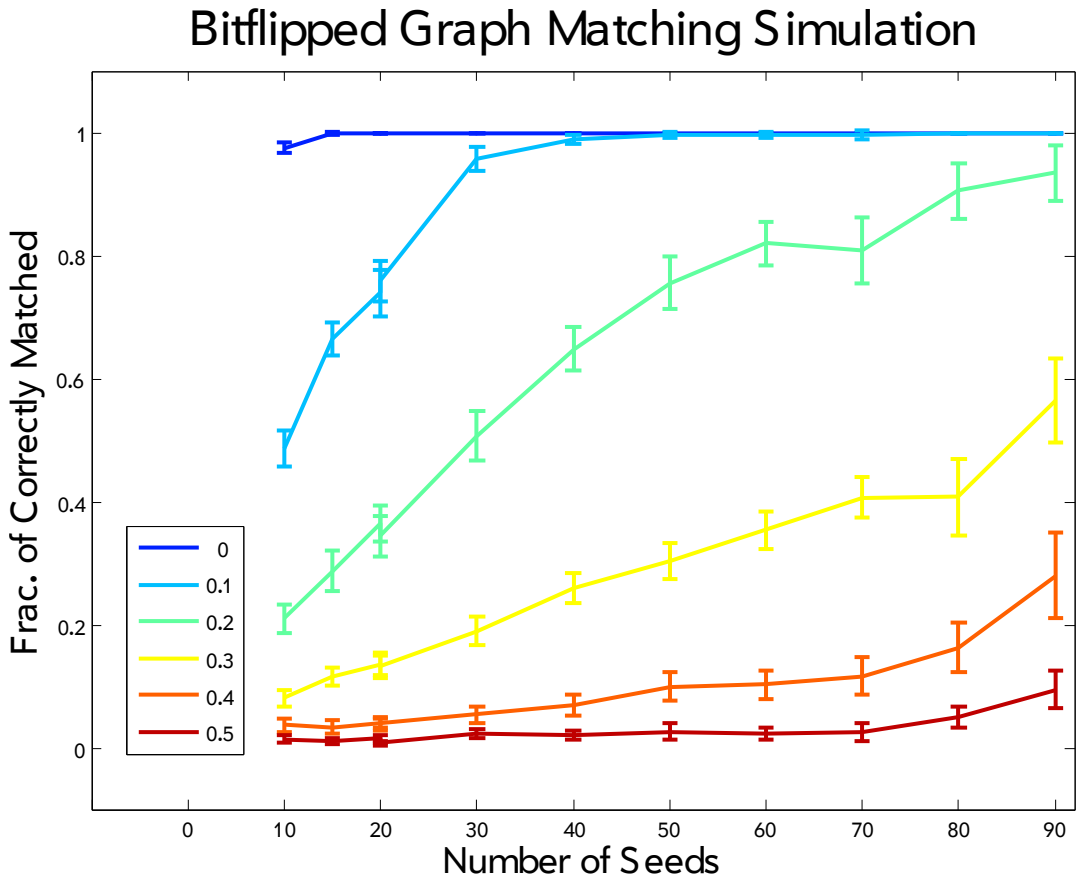


Figure 12.4: Seeded Graph Matching performance of the JOFC approach for bitflipped graph pairs of size $n = 100$. Fraction of correctly matched vertices among non-seeded vertices are plotted against number of seeds. Different colors correspond to different p_{pert} values.

nected in both graphs. This leaves $n = 253$ vertices. We consider both the original weighted graph for the first connectome and a binarized version of the same graph. We also consider symmetrized versions of each graph (which leads to directed and undirected graphs, respectively). In the case of weighted graphs, we normalize the two graphs so that they have approximately the same scale. For the JOFC approach, we use the weighted DICE dissimilarity (which simplifies to the generic DICE dissimilarity in the case of unweighted graphs) to compute Δ_c and Δ_g.

We compare the JOFC approach and the FAQ algorithm using the two connectomes. Whereas the true matching ratio ($\delta^{(m)}$) of both approaches is enhanced by the number of seeds, $\delta^{(m)}$ are relatively low compared with the maximum possible value (1). The correlation between the two connectomes is thus small, and we expect that there are biological explanations for this conclusion. The first conclusion we can reach from the comparison is that the two approaches are competitive. In fact, the JOFC approach provides significant improvement over the FAQ algorithm. The FAQ algorithm is not suitable for the situation when one graph is weighted and the other graph is unweighted (as in the weighted case) or the number of nonzero entries are significantly different, as in this case (as in both the weighted or unweighted cases). The JOFC approach works much better in both the weighted and unweighted cases.

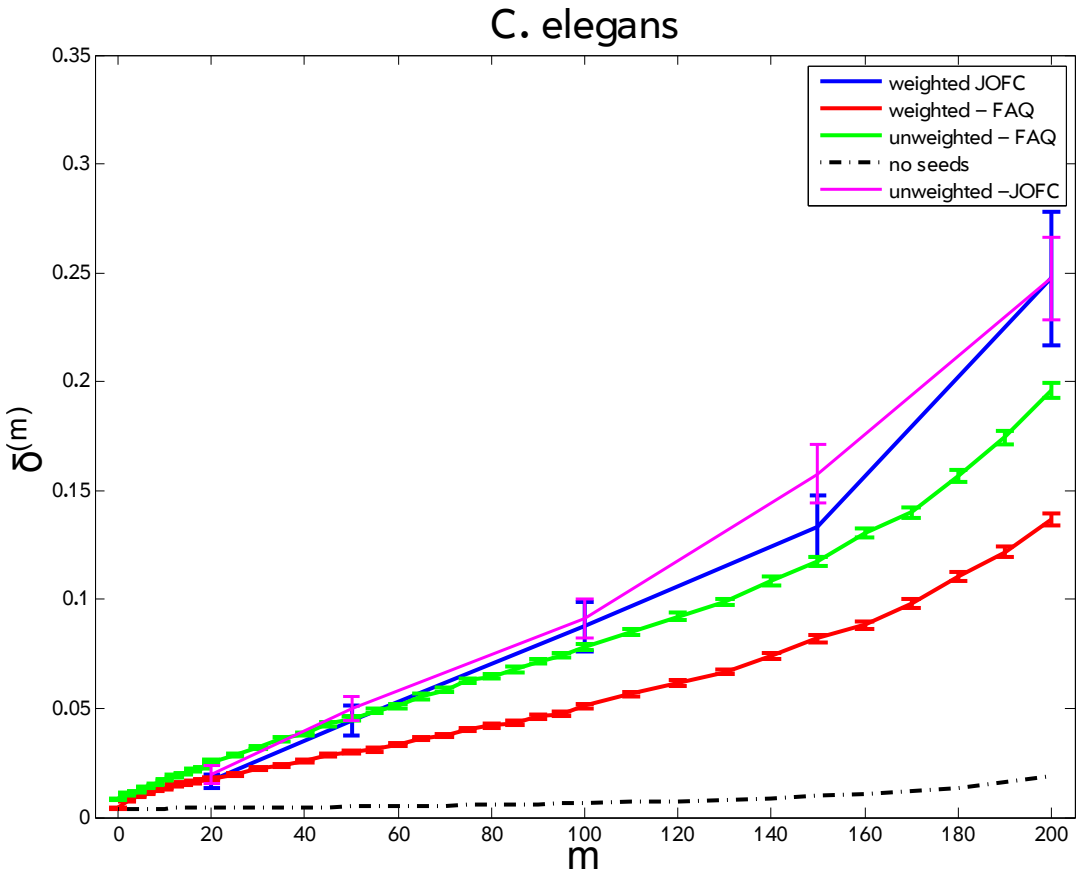


Figure 12.5: Seeded Graph Matching performance for the C. elegans connectomes using JOFC and FAQ algorithms. The true matching ratio is plotted against the number of seeds.

12.3.2.2 Enron communication graph

The Enron communication graph is extracted from the Enron email corpus, which was made public during criminal investigations by the Federal Energy Regulatory Commission. Though the number of actual users is approximately 150, each email alias is considered a vertex in the communication graph. The original number of email aliases is 184. The whole time interval is divided into 187 subintervals (each corresponding to a week). The emails are grouped according to the time interval of their timestamps. We then construct a time series of graphs $\mathfrak{G} = \{G^{(t)} = (V, E^{(t)})\}$, where $E^{(t)}$ correspond to emails that were sent at the t^{th} interval. We are interested in the intervals $t = 130$ and $t = 131$ (and $t = 132$ for some experiments), as previous investigations of the corpus found chatter anomalies at these time intervals [58]. When isolated vertices (and their corresponding vertices in the other graph) are removed in these two graphs, the number of vertices is reduced to 146. It is these pruned graphs that we match. The first two results are from matching $G(130)$ and $G(131)$. We consider both the undirected and directed versions of the two graphs.

We compare the performance of the modified-FAQ algorithm with the JOFC algorithm 12.6. Here, the modified-FAQ algorithm is significantly better than JOFC. This observation is valid for both directed and undirected versions of the graphs. With a large number of seeds, the difference between the two approaches gets smaller. We also note that the performance with the directed graphs is higher than that for the undirected graph for the FAQ algorithm, while for the JOFC approach, the results are better with

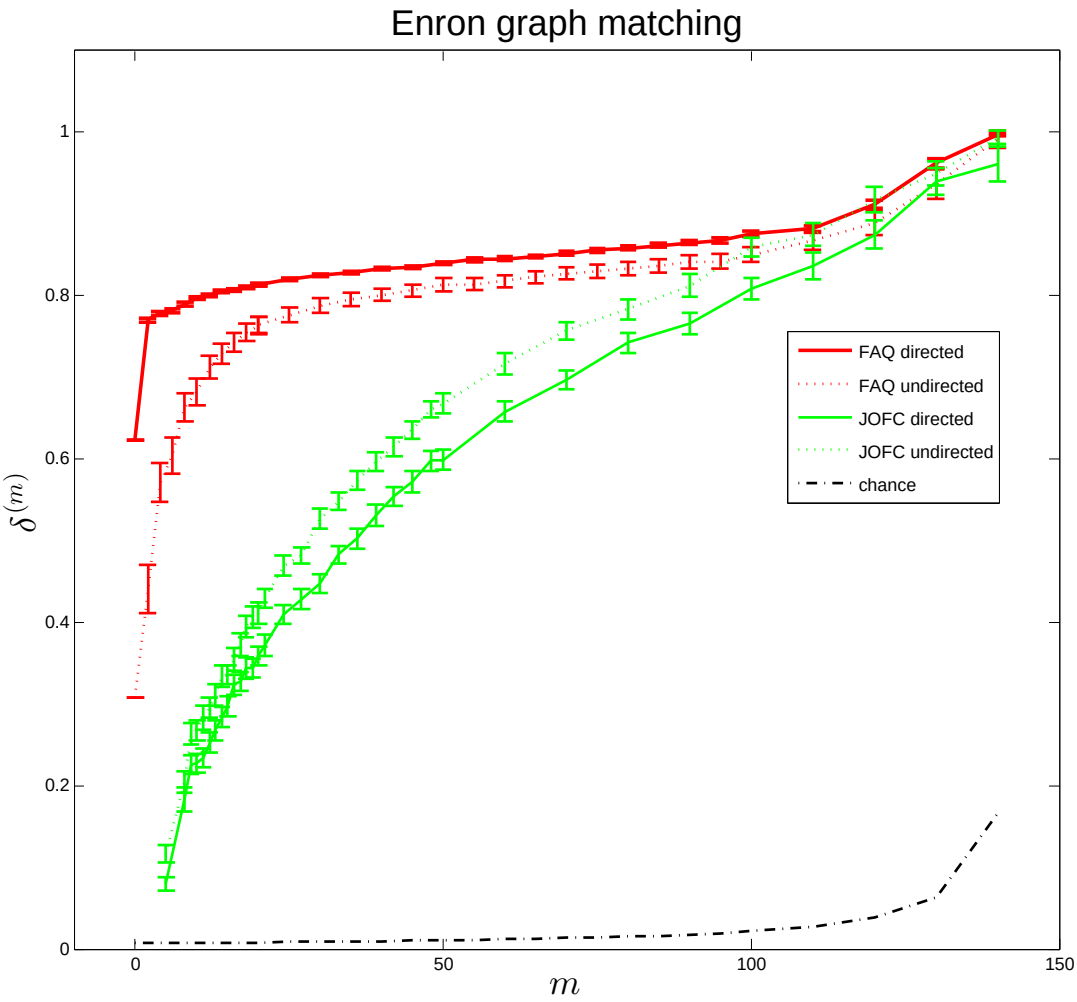


Figure 12.6: Seeded Graph Matching experiments on the Enron communication graphs for FAQ and JOFC and for undirected and directed versions of the two graphs.

the undirected graph.

For the plot in 12.7, we chose the embedding dimension d to be 20. These results are better compared with those in Figure 12.6, which leads us to conclude that d is another parameter that must be chosen with care.

The plot in Figure 12.8 is another result obtained from matching the Enron graphs using the FAQ algorithm, showing a comparison of the three pairs from $t = 130, 131,$ and 132. As one would expect, the graph matching between $G(130)$ and $G(131)$ is much more successful than is the graph matching between $G(130)$ and $G(132)$. We also note that if enough seeds are available, even the matching between $G(130)$ and $G(132)$ can be improved significantly. In fact, the improvement in the graph matching between $G(130)$ and $G(132)$ is larger than that obtained for the graph matching of the other two pairs.

We are also interested in the chatter anomaly detected in [58]. This anomaly is detected at $t = 132$. We also see from the graph matching results that the matching between $t = 130$ and 131 is better than the matching between $G(131)$ and $G(132)$. If there is excessive change in the connectivity, we expect the graph matching performance to suffer. This makes us wonder whether the true match ratio can be used to detect anomalies in a time series of graphs. Graph matching can be performed for the graph instances at consecutive time steps, and significant outliers would be labeled as outliers. The true matching ratio for a fixed number of seeds would be a statistic for anomaly detection.

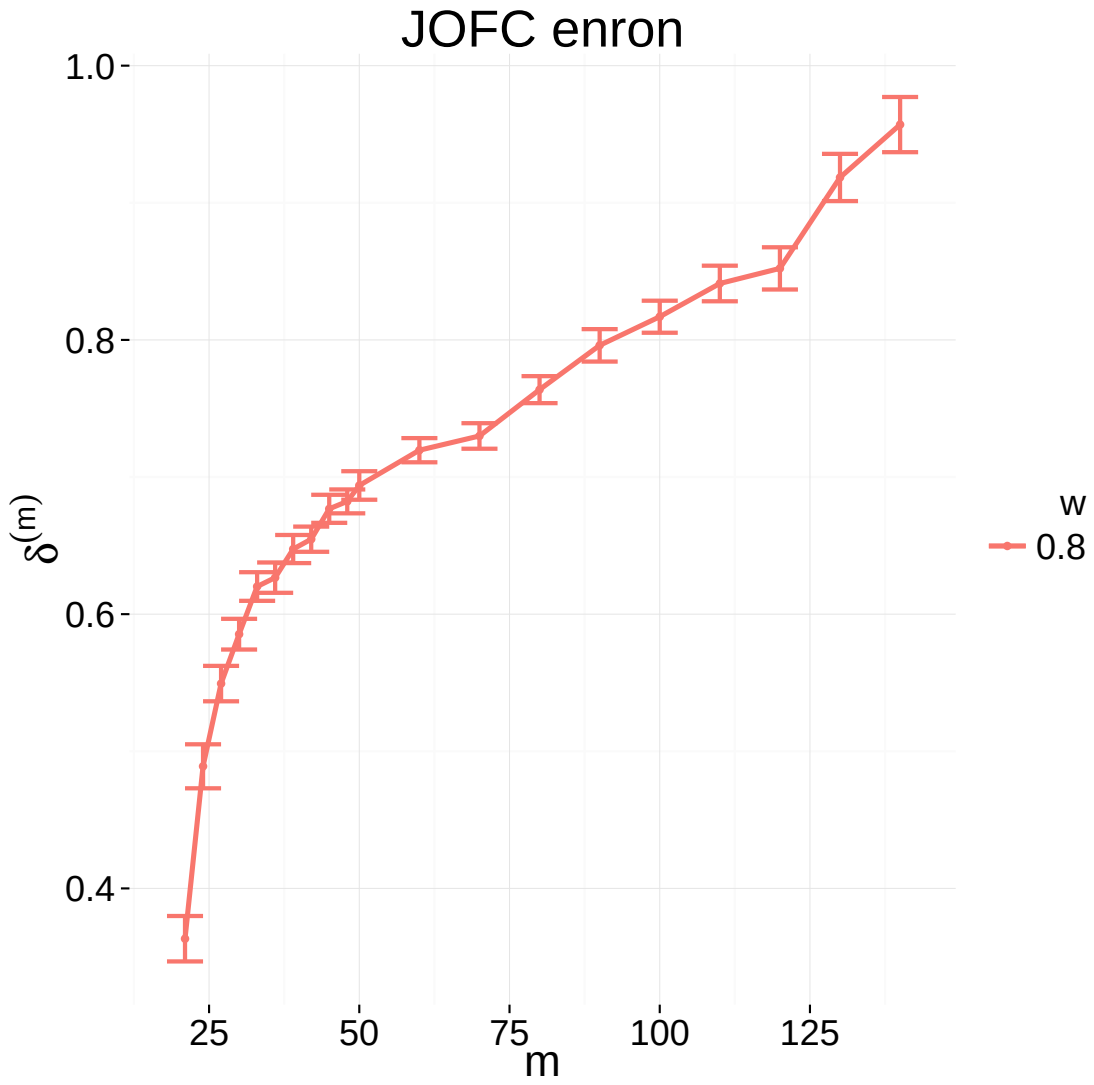


Figure 12.7: Seeded Graph Matching experiments on the Enron communication graphs for JOFC when the embedding dimension $d = 20$.

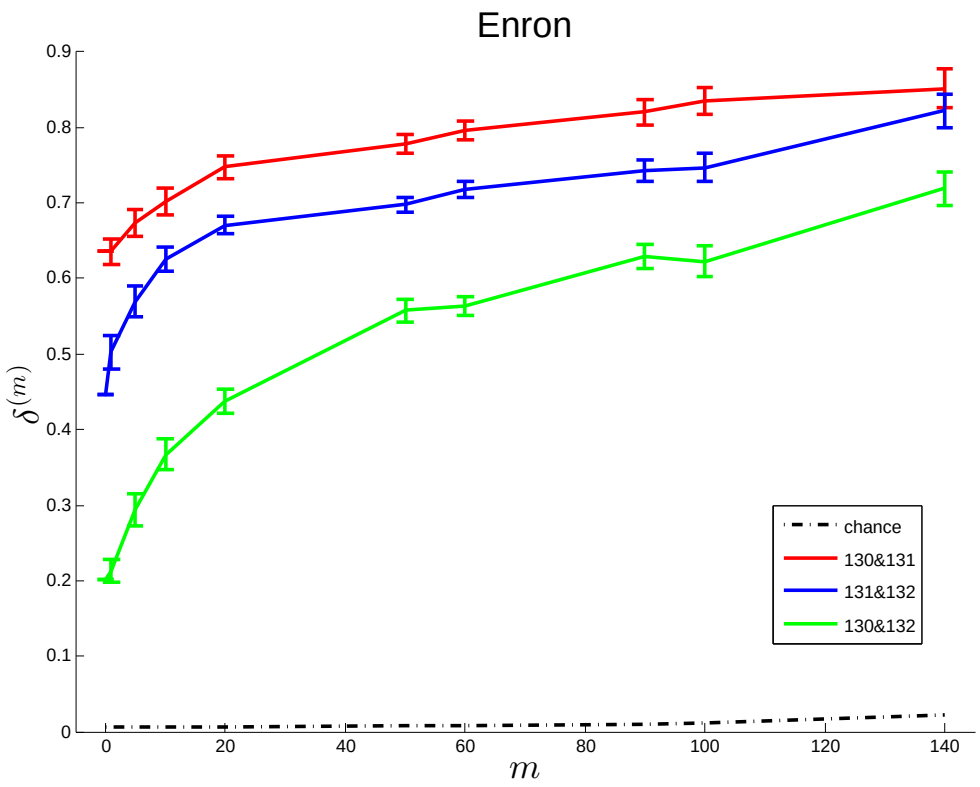


Figure 12.8: Seeded Graph Matching experiments on the Enron communication graphs for FAQ for t = 130, 131, and 132

12.3.2.3 Wikipedia hyperlink subgraph

Wikipedia is a free online encyclopedia created by volunteers around the world, consisting of millions of article in hundreds of languages (30 million articles in 287 languages, including over 4.3 million on the English Wikipedia site as of November 2013 [59]). Articles contain text, links to other articles, and multimedia content. We interpret the links as directed edges in a hyperlink graph, where vertices correspond to articles. A collection of articles was obtained from the English Wikipedia site that consisted of the directed 2-neighborhood of the document "Algebraic Geometry" [60]. This collection of 1382 articles and the correspondence of each article on the French Wikipedia site is our real-life dataset. For inference tasks, it is possible to utilize both the textual content of the documents and the hyperlink graph structure. The textual content of the documents is summarized by the bag-of-words model. Dissimilarities between documents in the same language are computed by the Lin-Pantel discounted mutual information [42, 43] and cosine dissimilarity $k(x_{ik}; x_{jk}) = 1 - (x_{ik}x_{jk})/(\|x_{ik}\|_2\|x_{ik}\|_2)$. The dissimilarities based on the hyperlink graph are the shortest-path distances in the graph, or for each pair of vertices i and j, the number of vertices one must travel to go from i to j. We consider the connected neighborhood of the English "Algebraic Geometry" topic; the induced graph for the French Wikipedia site of the 2-neighborhood from the English Wikipedia site might be a disconnected graph. Therefore, the shortest path dissimilarities from the French Wikipedia site are cut off at 6 (maximum shortest-path distance in the English Wikipedia graph). Further details about this dataset are available in [44].

Because this graph is relatively large compared with other real-life graphs that we considered, we only tested the FAQ algorithm. Note that testing JOFC on this graph is still possible, but we had no reason to believe that another JOFC experiment would provide any new insight.

The results show that there is a strong correlation between the two Wikipedia graphs and that as many as 50 seeds are enough to improve graph matching dramatically. More seeds improve the true matching ratio, but the improvements are much more modest. With graph matching using no seeds, we obtain a very small number of true matches.

12.3.2.4 Charitynet graph

The charitynet dataset consists of timestamped donation relationships between 8052 donors and 756 charities. The donations are divided into two time intervals according to whether they fall before the midpoint of the earliest and latest timestamps. Each group of donations are represented as edges in the bipartite graphs, where vertices correspond to donor or charity entities.

Let $tmid = \frac{(tmax - tmin)}{2}$. We build two bipartite graphs represented by B_1 and B_2 for $[tmin, tmid)$ and $(tmid, tmax]$, respectively – each $B^{(t)}$ is $n \times m$, where n is the total number of donors in all of charitynet and m is the total number of charities in all of charitynet. Therefore, $B^{(t)}_{ij}$ is a 1 if donor i gives to charity j during time interval t.

For charities i and j, let $A^{(t)}_{ij} = \sum_k B^{(t)}_{ki} B^{(t)}_{kj}$, i.e., the number of donors that give funds to both i and j during the time interval t. We consider the two graphs $\mathbb{G}_1$ and $\mathbb{G}_2$

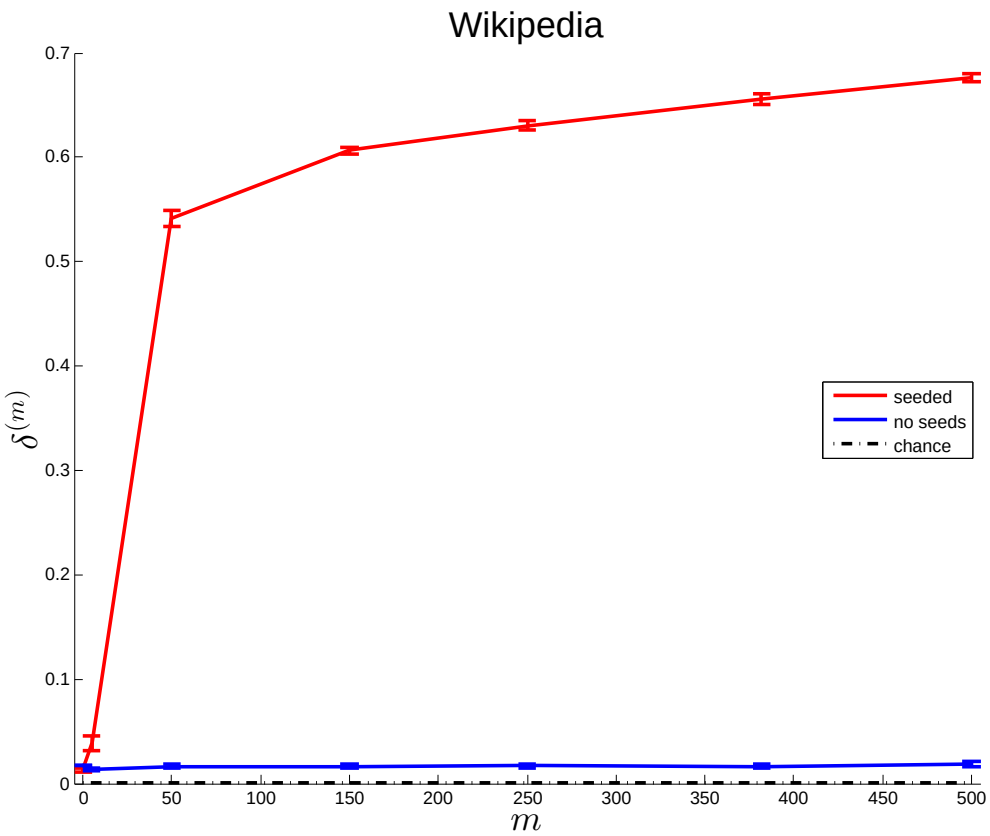


Figure 12.9: Seeded Graph Matching experiments on the English and French Wikipedia subgraph for FAQ

represented by $A^{[tmin,tmid)}$, $A^{[tmid,tmax]}$. These graphs consist of 8052 vertices representing all donors. Because the graphs are too large for matching by any algorithm, we sample pairs of subgraphs from these two graphs, where the pairs consist of the corresponding vertices. Consider the vertices from the largest connected components of $\mathbb{G}_1$ and $\mathbb{G}_2$, V_1^* and V_2^*, respectively. We will slightly abuse the notation by considering the corresponding vertices to be the same vertex, i.e., $\mathbb{G}_1 = (V, E_1)$, $\mathbb{G}_2 = (V, E_2)$. We randomly sample j vertices from $V_1^* \cap V_2^* \subset V$. For each sampled vertex pair $v \in V_1^* \cap V_2^*$, we consider $\mathcal{N}_1(v, k)$ and $\mathcal{N}_2(v, k)$: the k-neighborhood of v in each graph. The neighborhood size k is increased in increments of 1, starting from 1 until $\|\mathcal{N}_1(v, k) \cap \mathcal{N}_2(v', k)\| > n$, where n is the maximum allowable graph size, which depends on the computation time allotted for the graph matching. We match the subgraphs of $\mathbb{G}_1$ and $\mathbb{G}_2$ induced by $\mathcal{N}_1(v, k) \cap \mathcal{N}_2(v, k)$. There is no guarantee that either of these subgraphs are connected; however, for strongly correlated $\mathbb{G}_1$ and $\mathbb{G}_2$, they should be mostly connected, i.e., the largest connected component size is close to the graph size. This connection occurs because the local neighborhood of vertices must be the same for two strongly correlated graphs.

The comparison of the JOFC and FAQ approaches for CharityNet data12.10 show that the two approaches have comparable performances. Both of the approaches have small true matching ratios compared with 1, which is either due to the weak correlation between the two graphs, $\mathbb{G}_1$ and $\mathbb{G}_2$, or due to the related fact that the sampled subgraphs are not mostly connected.

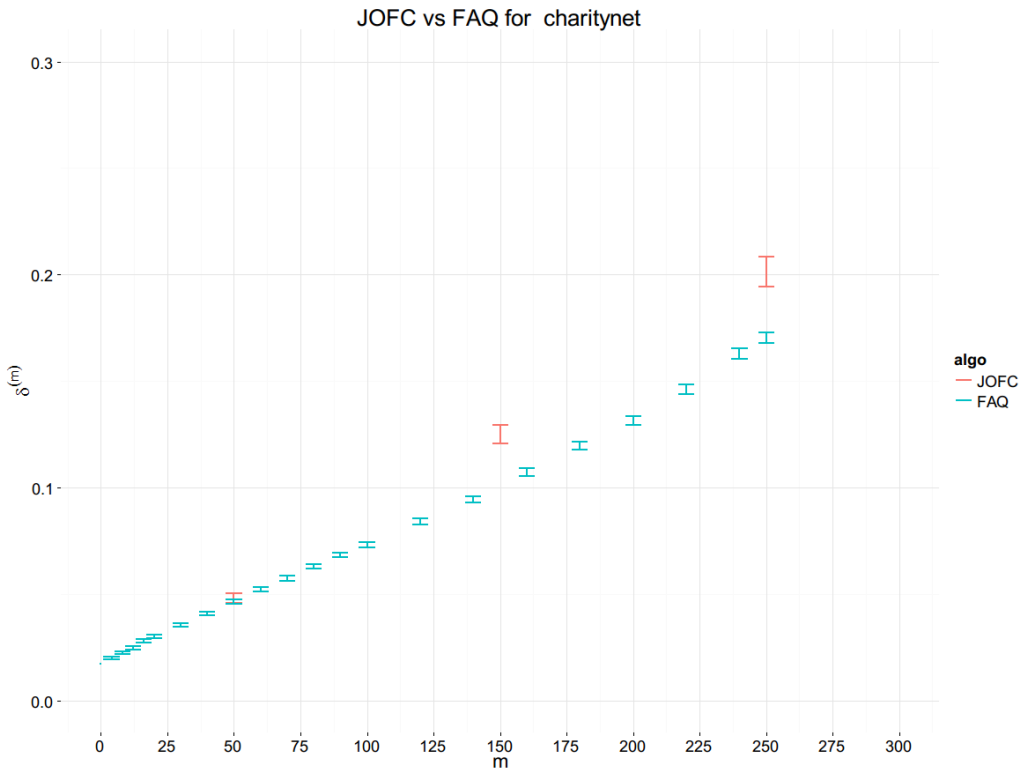


Figure 12.10: Graph Matching experiments on the two Charitynet graphs for JOFC

12.3.3 One-to-k matching of vertices

Consider the case in which the correspondences between vertices of $\mathbb{G}_1 = (V_1, E_1)$ and $\mathbb{G}_2 = (V_2, E_2)$ (represented by adjacency matrices, A and B, respectively) are not one-to-one. To describe the problem in its most general form, we could define group assignment functions $g_1 v_1 : V_1 \to \mathcal{G} = l_1, l_2, \ldots, l_v$, $g_2 v_2 : V_2 \to \mathcal{G}$, where the inverse images of $g_1 l_i$, $g_2 l_i$ are always nonempty. A vertex $v_1 \in V_1$ in one graph corresponds to a vertex $v_2 \in V_2$ in another graph, if $g_1(v_1) = g_2(v_2)$. We want to make the simple restriction that g_2 is a one-to-one mapping, that is, each vertex in $\mathbb{G}_2$ corresponds to at least one vertex in $\mathbb{G}_1$. For simulations, we consider a very simple case in which the i^{th} vertex in B corresponds to k_i vertices in A, where $1 \leq k_i \leq k_{max}$ and where k_{max} is the maximum number of corresponding vertices a B vertex can have. Denote by $g(\cdot) = g_2^{-1} \circ g_1(\cdot) : V_1 \to V_2$ the correspondence function from vertices in $\mathbb{G}_1$ to vertices in $\mathbb{G}_2$. Given r vertices in B and the corresponding vertices in V_1 for each of the r vertices ($u \in V_1$ such that $g(u) = v_2$), the task is to find at most k_max closest matches to each vertex of $\mathbb{G}_2$.

The following three information retrieval performance measures are used: Precision, Recall, and F-measure.

$$\text{Precision} := \frac{\text{Number of correct matches found}}{\text{Number of found matches}}$$

$$\text{Recall} := \frac{\text{Number of correct matches found}}{\text{Number of true matches}}$$

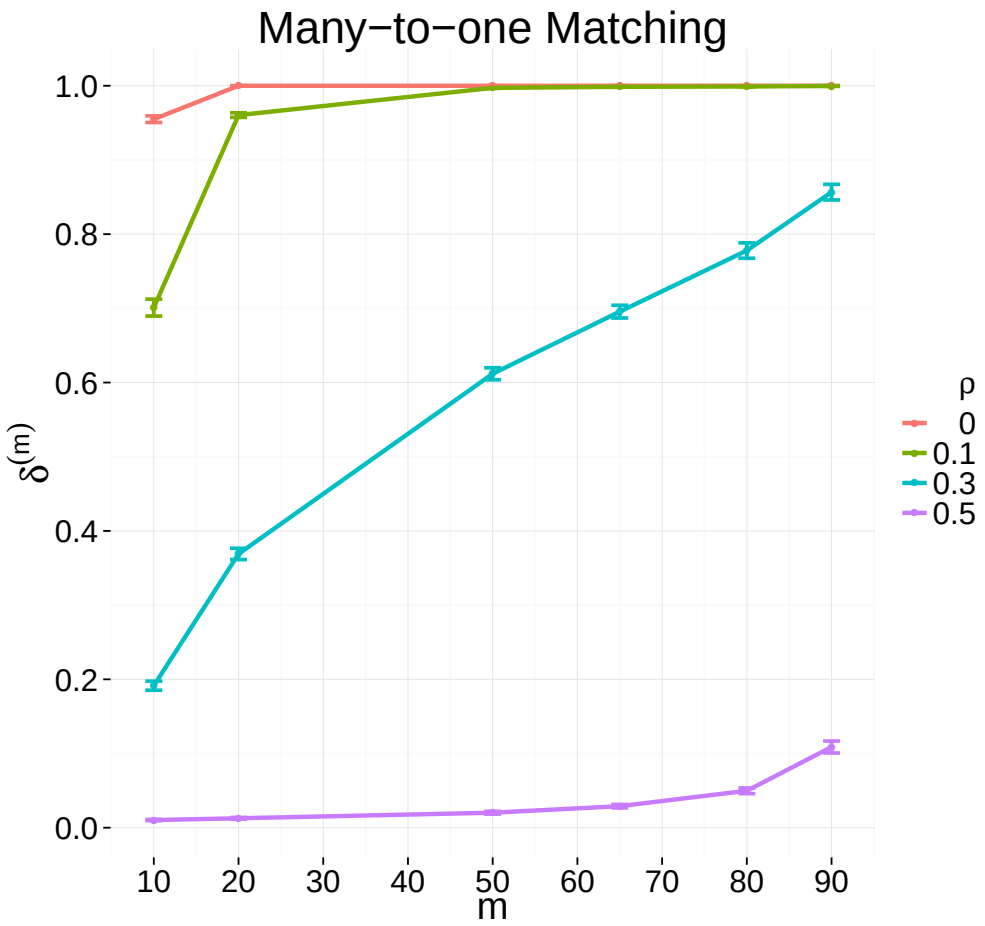


Figure 12.11: Graph Matching experiments on simulated graphs for JOFC

$$\text{F-measure} := \frac{2 \times \text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}}$$

For each vertex i of $\mathbb{G}_2$, the number of true matches is k_i. The three performance measures are calculated for each vertex of $\mathbb{G}_2$, and then, the three averages over all vertices B constitute the performance measures computed for the matching.

The results provide evidence that the JOFC approach is a suitable method for solv-

ing this variant of the graph matching problem. While both the JOFC approach and the FAQ algorithm were acceptable algorithms for the approximate graph matching (AGM) problem with one-to-one correspondence between vertices, with comparable matching performances, the FAQ algorithm cannot directly solve the GM problem when the vertex correspondences are one-to-k. In contrast, JOFC is quite adequate for solving AGM with one-to-k correspondence. We only need to use a matching algorithm that allows for multiple assignments for the vertices of one of the graphs. We use the full matching algorithm implemented in the R package [61], which finds "a matching between units that minimizes the average within-group distances, given a matrix describing the distances between two groups" [62]. That is, the *full* matching algorithm finds an assignment for all of the vertices in both graphs. Because the assignment problem is independent of the embedding, other matching algorithms can be used with the same embedding, if one is concerned about efficiency or other aspects of the matching algorithm.

Chapter 13

Conclusion

13.1 Conclusion

Our investigations began with a match detection problem where the data from disparate sources were available in dissimilarity representation. We formulated a joint embedding method to render the disparate data commensurate and applied the method to inference tasks such as match detection and graph matching. We introduced two criteria, fidelity and commensurability, that are essential for any inference task that uses such data. We investigated the tradeoff between fidelity and commensurability and its relation to the weighted raw stress criterion for MDS.

For hypothesis testing such as the exploitation task, we compared different values of the tradeoff parameter w in terms of testing power. The results indicate that for a joint optimization, one should consider an optimal compromise point between fidelity

and commensurability, which corresponds to an optimal weight w^* of the weighted raw stress criterion, in contrast to uniform weighting.

We then proved under which conditions w^* might exist. The uniqueness of w^* is not settled upon, and further investigations are necessary. In our search for the necessary conditions for the existence of w^*, we attempted to construct a counter-example in which w^* does not exist. This led to an interesting setting in which the embedding configuration has a discontinuity with respect to w in the raw stress function 9. While this is an interesting finding, this unique phenomenon appears in a carefully constructed point configuration, and we can reasonably assume that the probability measure of such a configuration is zero for most data settings.

We also introduced two alternative approaches for solving the same multiple view data problems, which are based on Procrustes Analysis and Canonical Correlational Analysis. These methods can also be viewed in terms of the fidelity-commensurability tradeoff. Procrustes$\circ$MDS optimizes fidelity with commensurability as the secondary priority. CCA optimizes commensurability subject to the linearity of projections to the commensurate space. Other studies in the scientific literature introduce fidelity and commensurability- like terms for solving multiple view problems. Our investigations of fidelity and commensurability are therefore relevant to understanding alternative approaches for solving similar multiview problems.

The different views in the multiview data can include two different graphs with vertices that share the same set of labels. Here, we assume that the corresponding vertices

with the same label represent the same object and that the edges incident to corresponding vertices are strongly correlated, if not identical. Given only the two simple graphs with no vertex label information, the graph matching problem is then the task of finding vertex correspondences. These correspondences are found by minimizing the edge disagreements between corresponding vertices. In some cases, a portion of the correspondences are known a priori, or can be discovered at no cost. If we know even a small portion of the correspondences, we can solve the seeded graph matching problem by inferring the rest of the correspondences by exploiting the vertices with the known correspondences known as seeds.

We proposed two solutions for seeded graph matching. The first one is based on the JOFC approach, where we use a dissimilarity measure to compute dissimilarities between vertices separately in each graph and jointly embed the dissimilarities. Using the pairwise distances between the embedded points as costs in an assignment problem, we find a match between the non-seeded vertices. The second solution we proposed is based on a relaxation of the combinatorial optimization problem. Given the adjacency matrices of the two graphs to be matched, A and B, we minimize $\|A - \mathscr{P}B\mathscr{P}^T\|_2$ with respect to the permutation matrix P. This problem is equivalent to a specific case of the quadratic assignment problem. The continuous relaxation we consider minimizes the same function over the set of doubly stochastic matrices. The Frank-Wolfe algorithm provides an iterative solution to this relaxed optimization problem over the convex domain of doubly stochastic matrices. The relaxed solution can then be projected to the set

of permutation matrices. We adapted this algorithm, called Fast Approximate Quadratic graph matching algorithm [63], to the seeded graph matching problem, where part of $\mathscr{P}$ is known due to some number of seeds. Without loss of generality, we let $\mathscr{P} = I \oplus P$, where P is the permutation matrix for the non-seeded vertex pairs. We are able to show through simulations and experiments on real graphs that even if a small portion of correspondences are known, our proposed modified FAQ algorithm is able to match the remaining vertex pairs much more successfully than unseeded graph matching.

While the modified FAQ algorithm is more suitable for matching pairs of simple graphs, there are many cases in which the correspondences are not well defined. A vertex in one graph may match to many or to none of the vertices in the other graph. The modified FAQ algorithm as it is currently described cannot handle such pathologies. For different variations of the simple seeded graph matching problem, the JOFC approach is much more suitable.

We also tested our approaches using real-world graph data. The matching ratio for a given number of seeds depends on various factors, such as how correlated the graph pairs are and how connected the vertices are, among others. It is possible that even if most of the correspondences are known, we would not obtain satisfactory performance on matching the rest of the vertices. This was the case for SGM with the Charitynet and C. elegans connectome graph pairs. However, even in these cases, seeds improve the graph matching performance significantly. In addition, our JOFC algorithm is competitive with and, in some cases, improves upon the modified FAQ algorithm, a modification of

the state-of-the art graph matching algorithm presented in [63].

13.2 Future Directions

Our findings warrant further investigations. For example, under what conditions is the fidelity-commensurability tradeoff parameter w^* unique? Our results indicate that for some settings, the value of w does not have a significant effect on performance in the neighborhood of the optimal value. For simulations using the data models in chapter 6, the $AUC(w)$ function exhibited a plateau near the optimal value. For graph matching, most w values yielded the same matching performance. However, for match testing on the Wikipedia data, varying w values had a very significant effect on matching performance. The ability to predict whether the tradeoff parameter w will or will not have a significant effect on the performance for a particular dataset is important, and thus, further investigation of this sensitivity issue is needed.

While we chose MDS with the weighted raw stress function as the embedding method, there are other possible embedding methods that can be used, such as local linear embedding and spectral embedding, for which fidelity and commensurability error-like terms can be defined. Investigations with alternative embedding approaches may result in performance improvements in the inference tasks and prove the generality of fidelity and commensurability criteria.

As in many data problems, model selection is an important issue that has significant

impact on our inference tasks. What should be the embedding dimension of dissimilarities? While we considered different d values, we have not really addressed this issue. A useful heuristic for our graph matching experiments was the selection of the dimension based on the matching ratio of the in-sample dissimilarities. That is, we embed and match the in-sample dissimilarities (for which we know the true matching). We then choose the minimum embedding dimension which fully or mostly recovers the true matching of the in-sample dissimilarities.

The seeded graph matching algorithm derived from the FAQ algorithm provides avenues for further research. Our efforts to improve the matching results by using a convex function as the objective function in the rQAP_2 formulation yielded mixed results. For a small number of seeds, the average true matching ratio for the rQAP_2 formulation was larger compared to that of the rQAP_1 formulation, whereas for a large number of seeds, it was smaller. These findings are most likely due to convergence issues, as shown in Figure 11.5. While the hybrid approach holds promise, it is still not good enough to provide the best of both worlds and calls for further tuning.

Various similar approaches in the literature for multiple view data can be investigated in the light of the fidelity and commensurability tradeoff. Most of these approaches have a tradeoff parameter that corresponds to our w. Some of these tradeoff parameters could be more amenable to analysis, which could lead to rigorous results related to the uniqueness and the existence of the parameter w.

Bibliography

[1] E. Pękalska and R. Duin, *The dissimilarity representation for pattern recognition: foundations and applications*, ser. Series in machine perception and artificial intelligence. World Scientific, 2005. [Online]. Available: http://books.google.co.uk/books?id=YPPr6eypHFwC

[2] M. Nadler and E. P. Smith, *Pattern recognition engineering*, 1st ed. Wiley, 1993. [Online]. Available: http://books.google.com/books/about/Pattern_recognition_engineering.html?id=gmdTAAAAMAAJ&pgis=1

[3] C. Priebe, D. Marchette, Z. Ma, and S. Adali, "Manifold Matching: Joint Optimization of Fidelity and Commensurability," *Brazilian Journal of Probability and Statistics*, submitted for publication.

[4] M.-R. Amini and C. Goutte, "A co-classification approach to learning from multilingual corpora," *Machine Learning*, vol. 79, no. 1-2, pp. 105–121, May 2009. [Online]. Available: http://www.springerlink.com/content/j8l1445j04x20703

[5] Q.-s. Sun, S.-g. Zeng, Y. Liu, P.-a. Heng, and D.-s. Xia, "A new method of feature

fusion and its application in image recognition," *Pattern Recognition*, vol. 38, pp. 2437 – 2448, 2005.

[6] B. McFee and G. Lanckriet, "Learning Multi-modal Similarity," *The Journal of Machine Learning Research*, vol. 12, pp. 491–523, February 2011. [Online]. Available: http://portal.acm.org/citation.cfm?id=1953048.1953063

[7] Y. Lin, T. Liu, and C. Fuh, "Dimensionality reduction for data in multiple feature representations," *Advances in Neural Information Processing Systems*, vol. 21, pp. 961–968, 2009. [Online]. Available: http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.144.4222&rep=rep1&type=pdf

[8] G. Lanckriet, N. Cristianini, P. Bartlett, L. Ghaoui, and M. Jordan, "Learning the kernel matrix with semidefinite programming," *The Journal of Machine Learning Research*, vol. 5, pp. 27–72, 2004. [Online]. Available: http://portal.acm.org/citation.cfm?id=1005334

[9] M. Gönen and E. Alpaydın, "Multiple Kernel Learning Algorithms," *Journal of Machine Learning Research*, vol. 12, no. July, p. 2211?2268, 2011. [Online]. Available: http://jmlr.org/papers/v12/gonen11a.html

[10] H. Choi, S. Choi, and Y. Choe, "Manifold integration with Markov random walks," in *Proceedings of the 23rd national conference on Artificial intelligence - Volume 1*. AAAI Press, 2008, pp. 424–429. [Online]. Available: http://portal.acm.org/citation.cfm?id=1619995.1620064

[11] D. Zhou and C. J. C. Burges, "Spectral clustering and transductive learning with multiple views," in *Proceedings of the 24th international conference on Machine learning - ICML '07*. New York, New York, USA: ACM Press, Jun. 2007, pp. 1159–1166. [Online]. Available: http://dl.acm.org/citation.cfm?id=1273496.1273642

[12] S. J. Pan and Q. Yang, "A Survey on Transfer Learning," *Knowledge and Data Engineering, IEEE Transactions on*, vol. 22, no. 10, pp. 1345–1359, 2010.

[13] B. Zadrozny, "Learning and evaluating classifiers under sample selection bias," *ACM International Conference Proceeding Series; Vol. 69*, 2004. [Online]. Available: http://portal.acm.org/citation.cfm?id=1015425

[14] D. J. Hand, "Classifier Technology and the Illusion of Progress," *Statistical Science*, vol. 21, no. 1, pp. 1–14, Feb. 2006. [Online]. Available: http://arxiv.org/abs/math/0606441

[15] H. Daumé III and D. Marcu, "Domain adaptation for statistical classifiers," *Journal of Artificial Intelligence Research*, vol. 26, no. 1, pp. 101–126, 2006. [Online]. Available: http://www.aaai.org/Papers/JAIR/Vol26/JAIR-2603.pdf

[16] S. Ben-David, J. Blitzer, K. Crammer, and F. Pereira, "Analysis of representations for domain adaptation," *Advances in Neural Information Processing Systems*, vol. 19, p. 137, 2007. [Online]. Available: http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.71.7478&rep=rep1&type=pdf

[17] X. Ling, W. Dai, G.-R. Xue, Q. Yang, and Y. Yu, "Spectral domain-transfer learning," *International Conference on Knowledge Discovery and Data Mining*, pp. 488–496, 2008. [Online]. Available: http://portal.acm.org/citation.cfm?id=1401951

[18] S. J. Pan, J. T. Kwok, and Q. Yang, "Transfer learning via dimensionality reduction," *Aaai Conference On Artificial Intelligence*, 2008. [Online]. Available: http://portal.acm.org/citation.cfm?id=1620163.1620177

[19] B. Chen, W. Lam, I. W. Tsang, and T.-L. Wong, "Discovering Low-Rank Shared Concept Space for Adapting Text Mining Models." *IEEE transactions on pattern analysis and machine intelligence*, vol. 6, no. 1, Oct. 2012. [Online]. Available: http://www.ncbi.nlm.nih.gov/pubmed/23165006

[20] C. Wang and S. Mahadevan, "Manifold alignment using Procrustes analysis," in *Proceedings of the 25th international conference on Machine learning - ICML '08*. New York, New York, USA: ACM Press, 2008, pp. 1120–1127. [Online]. Available: http://portal.acm.org/citation.cfm?doid=1390156.1390297

[21] R. Sibson, "Studies in the Robustness of Multidimensional Scaling: Procrustes Statistics," *Journal of the Royal Statistical Society. Series B (Methodological)*, vol. 40, no. 2, pp. 234–238, 1978.

[22] D. Zhai, B. Li, H. Chang, S. Shan, X. Chen, and W. Gao, "Manifold Alignment via

Corresponding Projections," in *Proceedings of the British Machine Vision Conference*. BMVA Press, 2010, pp. 3.1–3.11, doi:10.5244/C.24.3.

[23] J. Ham, D. Lee, and L. Saul, "Semisupervised alignment of manifolds," in *Proceedings of the Annual Conference on Uncertainty in Artificial Intelligence, Z. Ghahramani and R. Cowell, Eds*, vol. 10. Citeseer, 2005, pp. 120–127. [Online]. Available: http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.59.8098&rep=rep1&type=pdf

[24] C. F. Baumgartner, C. Kolbitsch, J. R. McClelland, D. Rueckert, and A. P. King, "Groupwise Simultaneous Manifold Alignment for High-Resolution Dynamic MR Imaging of Respiratory Motion," in *Information Processing in Medical Imaging*, ser. Lecture Notes in Computer Science, J. C. Gee, S. Joshi, K. M. Pohl, W. M. Wells, and L. Zöllei, Eds. Springer Berlin Heidelberg, 2013, vol. 7917, pp. 232–243. [Online]. Available: http://dx.doi.org/10.1007/978-3-642-38868-2_20

[25] B. Castle, M. W. Trosset, and C. E. Priebe, "A Nonmetric Embedding Approach to Testing for Matched Pairs," no. TR-11-04, October 2011. [Online]. Available: http://www.stat.indiana.edu/files/TR/TR-11-04.pdf

[26] I. Borg and P. Groenen, *Modern Multidimensional Scaling. Theory and Applications*. Springer, 1997.

[27] W. Torgerson, "Multidimensional scaling: I. theory and method," *Psychometrika*, vol. 17, pp. 401–419, 1952.

[28] M. W. Trosset, "Applications of multidimensional scaling to molecular conformation," *Computing Science and Statistics*, vol. 29, pp. 148–152„ 1998.

[29] M. W. Trosset and M. Tang, "On Combinatorial Laplacian Eigenmaps," Department of Statistics, Indiana University, Tech. Rep., 2010.

[30] R. Sibson, "Studies in the Robustness of Multidimensional Scaling: Perturbational Analysis of Classical Scaling," *Journal of the Royal Statistical Society. Series B (Methodological)*, vol. 41, no. 2, pp. pp. 217–229, 1979. [Online]. Available: http://www.jstor.org/stable/2985036

[31] D. MacKay, "Probabilistic multidimensional scaling: An anisotropic model for distance judgments," *Journal of Mathematical Psychology*, vol. 33, no. 2, pp. 187–205, 1989. [Online]. Available: http://www.sciencedirect.com/science/article/pii/0022249689900308

[32] J. Ramsay, "Maximum likelihood estimation in multidimensional scaling," *Psychometrika*, vol. 42, no. 2, pp. 241–266, 1977. [Online]. Available: http://dx.doi.org/10.1007/BF02294052

[33] M. W. Trosset and C. E. Priebe, "The out-of-sample problem for classical multidimensional scaling," *Comput. Stat. Data Anal.*, vol. 52, pp. 4635–4642, June 2008. [Online]. Available: http://portal.acm.org/citation.cfm?id=1377056.1377408

[34] W. Niemiro, "Asymptotics for $ M $-estimators defined by convex minimization," *The Annals of Statistics*, vol. 20, no. 3, pp. 1514–1533, 1992. [Online]. Available: http://projecteuclid.org/euclid.aos/1176348782

[35] E. Raik, "On the Stochastic Programming Problem with the Probability and Quantile Functionals," *Izvestia Akademii Nauk Estonskoy SSR. Phys and Math.*, vol. 21, no. 2, pp. 142–148, 1972.

[36] S. Adali and C. E. Priebe, "Fidelity-Commensurability Tradeoff in Joint Embedding of Disparate Dissimilarities," submitted to Journal of Classification.

[37] J. Gower, "Generalized procrustes analysis," *Psychometrika*, vol. 40, pp. 33–51, 1975, 10.1007/BF02291478. [Online]. Available: http://dx.doi.org/10.1007/BF02291478

[38] D. R. Hardoon, S. R. Szedmak, and J. R. Shawe-taylor, "Canonical Correlation Analysis: An Overview with Application to Learning Methods," *Neural Computation*, vol. 16, pp. 2639–2664, December 2004. [Online]. Available: http://portal.acm.org/citation.cfm?id=1119696.1119703

[39] J. Jagarlamudi, R. Udupa, and H. D. III, "Generalization of CCA via Spectral Embedding."

[40] J. R. Kettenring, "Canonical Analysis of Several Sets of Variables," *Biometrika*,

vol. 58, no. 3, pp. pp. 433–451, 1971. [Online]. Available: http://www.jstor.org/stable/2334380

[41] M. W. Trosset and R. Mathar, "On the Existence of Nonglobal Minimizers of the Stress Criterion for Metric Multidimensional Scaling," *American Statistical Association: Proceedings Statistical Computing Section*, 1997. [Online]. Available: www.caam.rice.edu/ ~ trosset/asa97.ps

[42] D. Lin and P. Pantel, "Concept discovery from text," in *Proceedings of the 19th international conference on Computational linguistics - Volume 1*, ser. COLING '02. Stroudsburg, PA, USA: Association for Computational Linguistics, 2002, pp. 1–7. [Online]. Available: http://dx.doi.org/10.3115/1072228.1072372

[43] P. Pantel and D. Lin, "Discovering word senses from text," in *Proceedings of the eighth ACM SIGKDD international conference on Knowledge discovery and data mining*, ser. KDD '02. New York, NY, USA: ACM, 2002, pp. 613–619. [Online]. Available: http://doi.acm.org/10.1145/775047.775138

[44] Z. Ma, D. J. Marchette, and C. E. Priebe, "Fusion and inference from multiple data sources in a commensurate space," *Statistical Analysis and Data Mining*, vol. 5, no. 3, pp. 187–193, 2012. [Online]. Available: http://dx.doi.org/10.1002/sam.11142

[45] M. Zhu and A. Ghodsi, "Automatic dimensionality selection from the scree plot via the use of profile likelihood," *Computational Statistics & Data Analysis*, vol. 51, no. 2, pp. 918–930, 2006.

[46] D. Conte and P. Foggia, "Thirty years of graph matching in pattern recognition," *International Journal of Pattern Recognition and Artificial Intelligence*, vol. 18, no. 3, pp. 265–298, 2004. [Online]. Available: http://www.worldscientific.com/doi/abs/10.1142/S0218001404003228

[47] S. Gold and A. Rangarajan, "A Graduated Assignment Algorithm for Graph Matching," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 18, no. 4, pp. 377–388, 1996.

[48] E. Bengoetxea, "Inexact graph matching using estimation of distribution algorithms," *These de Doctorat Specialite Signal et Images, Ecole*, 2002. [Online]. Available: http://www.sc.ehu.es/acwbecae/ikerkuntza/these/thesis.pdf

[49] H. Bunke, "Recent developments in graph matching," in *Proceedings 15th International Conference on Pattern Recognition. ICPR-2000*, vol. 2. IEEE Comput. Soc, pp. 117–124. [Online]. Available: http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=906030

[50] J. T. Vogelstein, J. M. Conroy, L. J. Podrazik, S. G. Kratzer, D. E. Fishkind, R. J. Vogelstein, and C. E. Priebe, "Fast Inexact Graph Matching with Applications in Statistical Connectomics," *submitted to Computational Statistics & Data Analysis.*

[51] M. Zaslavskiy, F. Bach, and J.-P. Vert, "Global alignment of protein-protein interaction networks by graph matching methods." *Bioinformatics (Oxford, England)*, vol. 25, no. 12, pp. i259–67, Jun. 2009. [Online]. Available: http://bioinformatics.oxfordjournals.org/cgi/content/abstract/25/12/i259

[52] H. W. Kuhn, "The Hungarian method for the assignment problem," *Naval Research Logistics (NRL)*, vol. 52, no. 1, pp. 7–21, 2005. [Online]. Available: http://dx.doi.org/10.1002/nav.20053

[53] R. A. Horn and C. R. Johnson, *Matrix Analysis*. Cambridge University Press, 1990.

[54] J. L. Maryak and D. C. Chin, "Global Random Optimization by Simultaneous Perturbation Stochastic Approximation," *IEEE Trans. Automat. Contr.*, vol. 53, no. 3, pp. 780–783, 2008.

[55] V. Lyzinski, S. Adali, J. T. Vogelstein, Y. Park, and C. Priebe, "Seeded Graph Matching via Joint Optimization of Fidelity and Commensurability," 2012, in preparation.

[56] B. Fichet and G. Le Calvé, "Structure géométrique des principaux indices de dissimilarité sur signes de présence-absence," *Statistique et analyse des données*, vol. 9, no. 3, pp. 11–44, 1984.

[57] J.-B. Angelelli, A. Baudot, C. Brun, and A. Guénoche, "Two local dissimilarity measures for weighted graphs with application to protein interaction networks," *Advances in Data Analysis and Classification*, vol. 2, pp. 3–16, 2008. [Online]. Available: http://dx.doi.org/10.1007/s11634-008-0018-3

[58] C. E. Priebe, J. M. Conroy, D. J. Marchette, and Y. Park, "Scan Statistics on Enron

Graphs," *Computational & Mathematical Organization Theory*, vol. 11, no. 3, pp. 229–247, 2005. [Online]. Available: http://dx.doi.org/10.1007/s10588-005-5378-z

[59] [Online]. Available: www.wikipedia.org

[60] [Online]. Available: http://en.wikipedia.org/wiki/Algebraic_Geometry

[61] B. B. Hansen and S. O. Klopfer, "Optimal full matching and related designs via network flows," *Journal of Computational and Graphical Statistics*, vol. 15, no. 3, pp. 609–627, 2006.

[62] M. M. Fredrickson, *Package 'optmatch'*. [Online]. Available: http://cran.r-project.org/web/packages/optmatch/optmatch.pdf

[63] J. T. Vogelstein, J. M. Conroy, L. J. Podrazik, S. G. Kratzer, E. T. Harley, D. E. Fishkind, R. J. Vogelstein, and C. E. Priebe, "Fast Approximate Quadratic Programming for Large (Brain) Graph Matching," *ArXiv e-prints*, Dec. 2011.

Vita

Sancar Adali received the B. Sc. degree in Electrical and Electronics Engineering from Bogazici University in Istanbul, Turkey in 2003, and enrolled in the Electrical Sciences and Computer Engineering graduate program in the School of Engineering at Brown University. He received the M. Sc. in Engineering degree from Brown University in 2005. He enrolled in the Applied Mathematics and Statistics Ph.D. program at Johns Hopkins University in 2006 and he received the M. Sc. degree from the same department in 2008. In 2010, he received Acheson J. Duncan award for the Advancement of Research in Statistics. His research focuses on statistical machine learning, applications of dissimilarity representation in pattern recognition in addition to analysis of graph data and graph inference problems.