MAXIMUM LIKELIHOOD ESTIMATION OF ROBUST CONSTRAINED GAUSSIAN MIXTURE MODELS

A DISSERTATION SUBMITTED TO
THE DEPARTMENT OF ELECTRICAL AND ELECTRONICS ENGINEERING
AND THE GRADUATE SCHOOL OF ENGINEERING AND SCIENCE
OF BILKENT UNIVERSITY
IN PARTIAL FULFILLMENT OF THE REQUIREMENTS
FOR THE DEGREE OF
DOCTOR OF PHILOSOPHY

By
Çağlar Arı
January, 2013

I certify that I have read this thesis and that in my opinion it is fully adequate, in scope and in quality, as a dissertation for the degree of Doctor of Philosophy.

Prof. Dr. Orhan Arıkan(Advisor)

I certify that I have read this thesis and that in my opinion it is fully adequate, in scope and in quality, as a dissertation for the degree of Doctor of Philosophy.

Asst. Prof. Dr. Selim Aksoy(Co-Advisor)

I certify that I have read this thesis and that in my opinion it is fully adequate, in scope and in quality, as a dissertation for the degree of Doctor of Philosophy.

Prof. Dr. Ergin Atalar

I certify that I have read this thesis and that in my opinion it is fully adequate, in scope and in quality, as a dissertation for the degree of Doctor of Philosophy.

Asst. Prof. Dr. Pınar Duygulu Şahin

I certify that I have read this thesis and that in my opinion it is fully adequate, in scope and in quality, as a dissertation for the degree of Doctor of Philosophy.

Assoc. Prof. Dr. Sinan Gezici

I certify that I have read this thesis and that in my opinion it is fully adequate, in scope and in quality, as a dissertation for the degree of Doctor of Philosophy.

Prof. Dr. Aydın Alatan

Approved for the Graduate School of Engineering and Science:

Prof. Dr. Levent Onural
Director of the Graduate School

ABSTRACT

MAXIMUM LIKELIHOOD ESTIMATION OF ROBUST CONSTRAINED GAUSSIAN MIXTURE MODELS

Çağlar Arı
Ph.D. in Electrical and Electronics Engineering
Supervisors: Prof. Dr. Orhan Arıkan and Asst. Prof. Dr. Selim Aksoy
January, 2013

Density estimation using Gaussian mixture models presents a fundamental trade off between the flexibility of the model and its sensitivity to the unwanted/unmodeled data points in the data set. The expectation maximization (EM) algorithm used to estimate the parameters of Gaussian mixture models is prone to local optima due to nonconvexity of the problem and the improper selection of parameterization. We propose a novel modeling framework, three different parameterizations and novel algorithms for the constrained Gaussian mixture density estimation problem based on the expectation maximization algorithm, convex duality theory and the stochastic search algorithms. We propose a new modeling framework called Constrained Gaussian Mixture Models (CGMM) that incorporates prior information into the density estimation problem in the form of convex constraints on the model parameters. In this context, we consider two different parameterizations where the first set of parameters are referred to as the information parameters and the second set of parameters are referred to as the source parameters. To estimate the parameters, we use the EM algorithm where we solve two optimization problems alternatingly in the E-step and the M-step. We show that the M-step corresponds to a convex optimization problem in the information parameters. We form a dual problem for the M-step and show that the dual problem corresponds to a convex optimization problem in the source parameters. We apply the CGMM framework to two different problems: Robust density estimation and compound object detection problems. In the robust density estimation problem, we incorporate the inlier/outlier information available for small number of data points as convex constraints on the parameters using the information parameters. In the compound object detection problem, we incorporate the relative size, spectral distribution structure and relative location relations of primitive objects as convex constraints on the parameters using the source parameters. Even with the propoper selection of the parameterization,

density estimation problem for Gaussian mixture models is not jointly convex in both the E-step variables and the M-step variables. We propose a third parameterization based on eigenvalue decomposition of covariance matrices which is suitable for stochastic search algorithms in general and particle swarm optimization (PSO) algorithm in particular. We develop a new algorithm where global search skills of the PSO algorithm is incorporated into the EM algorithm to do global parameter estimation. In addition to the mathematical derivations, experimental results on synthetic and real-life data sets verifying the performance of the proposed algorithms are provided.

Keywords: Gaussian mixture models, expectation maximization, convex optimization, duality, particle swarm optimization.

ÖZET

GÜRBÜZ KISITLI GAUSS KARIŞIM MODELLERİNİN ENBÜYÜK OLABİLİRLİK KESTİRİMİ

Çağlar Arı
Elektrik ve Elektronik Mühendisliği, Doktora
Tez Yöneticileri: Prof. Dr. Orhan Arıkan ve Yrd. Doç. Dr. Selim Aksoy
Ocak, 2013

Gauss karışım modelleri ile dağılım kestirimi yaparken modelin esnekliği ile veri kümesindeki istenmeyen/modellenmeyen veri noktalarına olan hassaslığı arasında temel bir ikilem durumu ortaya çıkmaktadır. Uygun olmayan parametre seçimi ve problemin içbükey olmamasından dolayı Gauss karışım modellerinin parametrelerinin kestirimi için kullanılan beklenti enbüyükleme (EM) yöntemi en iyi parametreleri bulamayabilmektedir. Bu tezde, beklenti enbüyükleme yöntemi, içbükey eşleklik (duality) teorisi ve rasgele arama yöntemlerini temel alan kısıtlı Gauss karışım modelleri için yeni bir modelleme sistemi, üç farklı parametrizasyon ve özgün yöntemler önerilmektedir. Kısıtlı Gauss karışım modelleri (CGMM) olarak adlandırdığımız modelleme sisteminde dağılım kestirimi problemi hakkında sahip olunan bilgiler model parametreleri üzerine içbükey kısıtlar koyularak kullanılabilmetedir. Bu durum için bilgi parametreleri ve kaynak parametreleri olarak ifade ettiğimiz iki parametrizasyon düşünülmektedir. Parametrelerin kestirimi için kullandığımız EM yönteminin E-adımı ve M-adımında sıra ile iki eniyileme problemi çözülmektedir. M-adımındaki problemin bilgi parametreleri cinsinden içbükey eniyileme problemi olduğu gösterilmektedir. M-adımı için eşlek (dual) problem oluşturulup bu problemin ise kaynak parametreleri cinsinden içbükey eniyileme problemi olduğu gösterilmektedir. CGMM modelleme sistemi gürbüz dağılım kestirimi ve bileşik nesne bulma problemlerine uygulanmaktadır. Gürbüz dağılım kestirimi probleminde, az sayıda veri noktası için var olan istenilen/aykırı nokta bilgileri bilgi parametreleri üzerine içbükey kısıtlar koyarak modellenmektedir. Bileşik nesne bulma probleminde ise basit nesneler hakkında sahip olduğumuz göreceli boyut, spektral dağılım yapısı ve göreceli yer bilgileri kaynak parametreleri üzerine içbükey kısıtlar koyarak modellenmektedir. Uygun parametre seçimi yapılsa dahi Gauss karışım modelleri ile dağılım kestirimi problemi içbükey eniyileme problemine denk gelmemektedir. Genelde

rasgele arama, özelde parçaçık sürüsü eniyileme (PSO) yöntemlerinin etkili kullanılmasına olanak sağlamak için kovaryans matrislerinin özdeğer ayrıştırmasına dayalı üçüncü bir parametrizasyon önerilmektedir. Evrensel parametre kestirimi yapabilmek için PSO yönteminin evrensel arama becerilerini EM yöntemine eklediğimiz yeni bir yöntem sunulmaktadır. Matematiksel analiz ve gösterimlere ek olarak sentetik ve gerçek hayat veri kümeleri kullanılarak önerilen yöntemlerin başarılı olduğu gösterilmektedir.

Anahtar sözcükler: Gauss karışım modelleri, beklenti enbüyükleme, içbükey eniyileme, eşleklik, parçacık sürüsü eniyileme.

Acknowledgement

I would like to express my sincere gratitude to my supervisors, Asst. Prof. Dr. Selim Aksoy and Prof. Dr. Orhan Arıkan for their guidance and support throughout the development of this thesis.

I would also like to thank Prof. Dr. Ergin Atalar, Asst. Prof. Dr. Pınar Duygulu Şahin, Assoc. Prof. Dr. Sinan Gezici, Prof. Dr. Aydın Alatan, Assoc. Prof. Dr. Nail Akar and Prof. Dr. Gözde Bozdağı Akar for accepting to read the manuscript and commenting on the thesis.

This work was supported in part by TÜBİTAK (The Scientific and Technological Research Council of Turkey) Grants 104E074 and 109E193.

Finally, I want to express my gratitude to my family, Utku, Şentaç and Aydoğan Arı for their support and understanding. I dedicate this dissertation to them.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Objective and Contributions

Density estimation can be considered as the most general form of estimation problems. It provides a probabilistic framework that allows us to formulate the problem in a mathematically principled way where the principles such as the maximum likelihood [1],[2], [3] and the maximum entropy [4], [5], [6], [7], [8] can be used to estimate the problem parameters [9], [10], [11], [11], [12], [13], [14].

Gaussian mixture models [15], [16], [17], [18], [19] are very flexible density models and have been widely used in speech processing [20], [21], [22], [23], image processing [24], [25], [26], [27], [28], [29], [30], [31] computer vision [32], [33] and pattern recognition [19], [18], [17]. The maximum likelihood is the most popular and commonly used principle to estimate the parameters of Gaussian mixture models [19], [22], [17]. However, the negative log-likelihood function for Gaussian mixture models is not a convex function of the parameters. Thus, there is no algorithm that is guaranteed to find the globally optimal parameter estimates [34], [35], [36].

The expectation maximization (EM) algorithm and its variants [37], [38], [39] are the most commonly used algorithms to estimate the parameters of Gaussian

mixture models. The EM algorithm is a very general and popular algorithm used for doing maximum likelihood estimation of the parameters in models with hidden variables. The fundamental idea behind the EM algorithm is to use an upper bound function on the negative log-likelihoods of the observed variables by introducing distributions over the hidden variables. This bound is a function of the negative log-likelihoods of the joint distributions of both the hidden and the observed variables and the introduced distributions over the hidden variables. The EM algorithm consists of two steps called the E-step and the M-step. In the E-step, the bound function is minimized over the introduced distributions over the hidden variables while holding the parameters found in the previous iteration fixed. In the M-step, the bound function is minimized over the parameters while holding the introduced distributions found in the E-step fixed. This procedure is then repeated until a fixed point of the algorithm corresponding to a local optimum is reached. This method is guaranteed to monotonically decrease the negative log-likelihood and to converge to a local minimum [37], [38].

There are two major problems which prevent the effective use of the EM algorithm for Gaussian mixture models. First, the EM algorithm does not address the question of parameterization. There are two commonly used parameterizations for Gaussian mixture models. The most common way is to use the probabilities, the mean vectors and the covariance matrices of Gaussian components for parameterization [19], [22], [17] which we refer to as the source parameterization. An alternative way is to use the log probabilities, information vectors and information matrices (inverse covariance matrices) for parameterization [37], [40], [41], [42] which we refer to as the information parameters. Considering that the original objective function was not a convex function of the parameters, one expects to have a convex optimization problem with the proper selection of the parameterization for the M-step where the bound can easily be minimized over the parameters. The second problem is that the EM algorithm does not address the dual problems for the M-step which correspond to convex optimization problems [36] for alternative parameterizations.

Density estimation using Gaussian mixture models presents a fundamental

trade off between the flexibility of the model and its sensitivity to the unwanted/unmodeled data points in the data set. The most common approach to tackle these problems is to incorporate the prior knowledge about the problem in the form of prior distributions over the parameters [43], [44], [45], [46], [42] to encode preferences about different parameter settings. In practice, it is hard to come up with prior distributions that will encode the desired interrelationships between the parameters. Furthermore, this is a very indirect way of formulating the parameter relationships.

Another important problem with the density estimation using Gaussian mixture models is that the number of parameters required for the covariance matrices grows quadratically with the dimension of the data set. This is a common problem encountered in domains such as speech recognition, image processing, computer vision and pattern recognition where the dimensionality of the data is often high and the size of the data set is relatively small. To overcome this problem, researchers often constrain the Gaussian mixture parameters to decrease the number of independent parameters. For instance, in speech recognition researchers generally use diagonal covariance matrices with several Gaussian components rather than fewer Gaussian components with full covariance matrices [22], [21], [23]. In image processing, computer vision and pattern recognition, it is desirable to limit the number of independent parameters by taking advantage of the independences between the subsets of the variables using the domain knowledge. For example, zero entries in the information (inverse covariance) matrices correspond to conditional independence relations between the variables given the rest of the variables [40], [47], [3] and there are lots of algorithms trying to estimate the sparsity pattern of the information matrices automatically [40], [48], [49], [50], [51], [52]. Similarly, zero entries in the covariance matrices correspond to marginal independence relations between the variables [47], [3] and such restrictions are often used in speech recognition, computer vision and pattern recognition [53], [54], [33], [55], [56], [57] and there are lots of algorithms that try to estimate the sparsity pattern of the covariance matrices automatically [58], [59], [60], [61].

Similar problems due to the relatively small size of the available data sets arise in density adaptation problems with Gaussian mixture models. For instance, in

speech recognition previously learned Gaussian mixture models are needed to be adapted to different speakers or environmental conditions using relatively small amount of new data samples [62], [63], [64], [65], [66], [67], [68]. In this context, the new mean vectors of the Gaussian components are constrained to be an (unknown) affine transformation of the previously learned mean vectors [62], [69], [70], [66]. Moreover, this idea can also be extended to the diagonal and arbitrary covariance covariance matrices where the new covariance matrices are constrained to be an (unknown) affine transformation of the previously learned covariance matrices [71], [66], [53]. Algorithms based on linear regression are proposed to estimate the constrained mean vectors, constrained covariance matrices, and their corresponding affine transformations [62], [69], [70], [71], [66], [53].

In the first part of this thesis, we present a novel constrained Gaussian mixture model framework that incorporates the prior information about the problem directly as convex constraints on the model parameters. The proposed framework can handle convex constraints either on the information parameters or on the source parameters (but it cannot handle the both simultaneously). Putting constraints on the model parameters allows us a more direct way to encode the interrelationships between the model parameters. We show that the M-step for the EM algorithm corresponds to a convex optimization problem in the information parameters, and additional convex inequality and affine equality constraints on the information parameters can be handled by solving a constrained convex optimization problem. Furthermore, using the convex duality theory, we present an unconstrained dual problem for the M-step which corresponds to a convex optimization problem in the source parameters. Hence, if the constraints on the parameters of the Gaussian mixture models can be represented as convex inequality and affine equality constraints on the source parameters, we can solve the constrained convex dual optimization problem for the M-step to handle the convex constraints on the source parameters. The initial version of the proposed framework described in this part is also presented in [72].

In many problems, the data points of interest are observed as part of a larger set of observations where some of the points do not follow the assumed restricted parametric distribution. We refer to the data points being distributed according

to the assumed distribution as the inliers and the rest of the data points as the outliers. In practice, it is often hard to know the distribution of the outliers. In parametric density models, the common way to detect the outliers is to select a threshold level for the log-likelihood function and classify the data points below the selected threshold as the outliers and the data points above the threshold value as the inliers. However, instead of trying different threshold values, it is desirable to find the threshold value using inlier/outlier information available for few data points.

In the second part of this thesis, we present a probabilistic framework for the robust estimation of the Gaussian mixture models. We assume that the inliers are distributed according to a Gaussian mixture model. We present an EM algorithm so that when the posterior distributions of outliers given the data points are constrained to take only $0-1$ binary values and the likelihoods of the data points given they are outliers are assumed to be equal to a constant value, we can determine the inliers and the outliers without any additional information about the outliers in the E-step, and we can estimate the information parameters of Gaussian mixture density modeling the inliers in the M-step. Furthermore, we incorporate the inlier/outlier information available for small number of data points as affine inequality constraints on the information parameters and estimate both the consistent information parameters and the constant value for the likelihoods of the data points given they are outliers simultaneously by solving a constrained convex optimization problem for the M-step. The initial version of the model described in this part is also partly described in [72].

Even with the proper selection of the parameterization, density estimation problem for Gaussian mixture models is not jointly convex in both the E-step variables and the M-step variables. Hence the EM algorithm is prone to local optima. The common approach is to run the EM algorithm many times from different initial configurations and to use the result corresponding to the highest log-likelihood value. However, even with some heuristics that have been proposed to guide the initialization, this approach is usually far from providing an acceptable solution especially with increasing dimensions of the data space. Furthermore, using the results of other algorithms such as k-means [20], [73] for

initialization is also often not satisfactory because there is no mechanism that can measure how different these multiple initializations are from each other. In addition, this is a very indirect approach as multiple EM procedures that are initialized with seemingly different values might still converge to similar local optima. Consequently, this approach may not explore the solution space effectively using multiple independent runs.

Researchers dealing with similar problems have increasingly started to use population-based stochastic search algorithms [74], [75], [76] where different potential solutions are allowed to interact with each other. These approaches enable multiple candidate solutions to simultaneously converge to possibly different optima by making use of the interactions. Genetic algorithm (GA) [77], [78], [79], [80], differential evolution (DE) [81], [82], and particle swarm optimization (PSO) [83], [84] have been the most common population-based stochastic search algorithms used for the estimation of some form of GMMs. Although many different versions of these algorithms have been proposed for various optimization problems, their applications in GMM estimation share some common properties.

The general GA framework creates successive generations of candidate solutions having improved goodness values by applying reproduction operators and selection mechanisms. Contrary to the classical use of binary string representations, the GA variants for problems that involve continuous parameters like in GMM estimation represent the candidate solutions as sets of real numbers [85],[86], [87], [88],[89]. A GA procedure usually consists of four basic stages: initialization, fitness assignment, reproduction, and selection. A population of candidate solutions are randomly generated during initialization. Then, a fitness function such as the sum of squared error [89] or the likelihood function [85], [86],[88] is used to assign a goodness value to each candidate solution. Candidate solutions with high fitness values are selected for reproduction in a stochastic manner. New candidate solutions are created from the selected solutions called parents using crossover and mutation operators. Crossover determines which parts of the chosen parents will be copied into new candidate solutions. Alpha blended crossover operators [85], [88], [89] are used with real-coded parameters

where some convex combination of the parents are formed. Adding a small random vector to the parent is commonly used as the mutation operator [85], [89]. In the selection phase, a new population is formed by replacing existing solutions with poor fitness values with newly created ones.

DE is another population-based stochastic search algorithm that is very similar to real-coded GAs. After similar initialization and fitness assignment steps, the mutation operator involves the formation of a mutant vector by adding the weighted differences of two randomly selected candidate solutions to another randomly selected candidate solution, and the crossover stage takes some parts of the mutant vector and some parts of a candidate solution to form a new vector considered for selection [90].

PSO is a relatively newer optimization technique that has also been used for GMM estimation. In PSO, candidate solutions are called particles where each particle consists of a position vector that encodes the parameters and a velocity vector that determines the new position in the parameter space in the next iteration. The velocity vectors are updated using the particles' current velocity, the difference between its personal best position and its current position, and the difference between the global best position and its current position in a stochastic manner [91], [92], [93], [94]. The personal best and global best are selected according to the positions achieving the highest fitness values in the personal history of the candidate solution of interest and in the histories of all candidate solutions, respectively.

Even though these approaches have been shown to perform better than non-stochastic alternatives such as k-means and fuzzy c-means, the interaction mechanism that forms the basis of the power of the stochastic search algorithms has also limited the use of these methods due to some inherent assumptions in the candidate solution parameterization. For example, the crossover and mutation operators in GA and DE, and the update operations in PSO involve randomized addition, swapping, and perturbation of the individual parameters of the candidate solutions. However, randomized modification of individual elements of a covariance matrix independently as in the mutation and update operations

does not guarantee the result to be a valid (i.e., symmetric and positive definite) covariance matrix. Likewise, partial exchanges of parameters between two candidate solutions as in crossover operations lead to similar problems. Hence, these problems confined the related work to either use no covariance structure (i.e., implicitly use identity matrices centered around the respective means) [89], [90], [91], [92], [94] or constrain the covariances to be diagonal [85],[93]. Consequently, most of these approaches were limited to the use of only the mean vectors in the candidate solutions and to the minimization of the sum of squared errors as in the k-means setting instead of the maximization of a full likelihood function.

Exceptions where both mean vectors and full covariance matrices were used in candidate solutions include [86], [87] where EM was used for the actual local optimization by fitting Gaussians to data in each iteration and a GA was used only to guide the global search by selecting individual Gaussian components from existing candidate solutions in the reproduction steps. However, treating each Gaussian component as a whole in the search process and fitting it locally using the EM iterations may not explore the whole solution space effectively especially in higher dimensions. Another example is [88] where two GA alternatives for the estimation of multidimensional GMMs were proposed. The first alternative encoded the covariance matrices for d-dimensional data using $d + d^2$ elements where d values corresponded to the standard deviations and d^2 values represented a correlation matrix. The second alternative used d runs of a GA for estimating 1D GMMs followed by d runs of EM starting from the results of the GAs. Experiments using 3D synthetic data showed that the former alternative was not successful and the latter performed better. We can conclude that full exploitation of the power of GMMs involving arbitrary covariance matrices estimated using stochastic search algorithms necessitates new parameterizations where the individual parameters are independently modifiable so that the resulting matrices remain valid covariance matrices after the stochastic updates and have bounded ranges so that they can be searched within a finite solution space.

Another important problem that has been largely ignored in the application of stochastic search algorithms to GMM estimation problems in the pattern recognition literature is identifiability. In general, a parametric family of probability

density functions is identifiable if distinct values of the parameters determine distinct members of the family [15], [19]. For mixture models, the identifiability problem exists when there is no prior information that allows discrimination between its components. When the component densities belong to the same parametric family (e.g., Gaussian), the mixture density with K components is invariant under the $K!$ permutations of the component labels (indices). Consequently, the likelihood function becomes invariant under the same permutation, and this invariance leads to $K!$ equivalent modes, corresponding to equivalence classes on the set of mixture parameters. This lack of uniqueness is not a cause for concern for the iterative computation of the maximum likelihood estimates using the EM algorithm, but can become a serious problem when the estimates are iteratively computed using simulations when there is the possibility that the labels (order) of the components may be switched during different iterations [15], [19]. Considering the fact that most of the search algorithms depend on the designed interaction operations, performances of the operations that assume continuity or try to achieve diversity cannot work as intended, and the discontinuities in the search space will make it harder for the search algorithms to find directions of improvement. In an extreme case, the algorithms will fluctuate among different solutions in the same equivalence class, hence, among several equivalent modes of the likelihood function, and will have significant convergence issues. This problem is known as "label switching" in the statistics literature for the Bayesian estimation of mixture models using Markov chain Monte Carlo (MCMC) strategies. The label switching corresponds to the interchanging of the parameters of some of the mixture components and the invariance of the likelihood function as well as the posterior distribution for a prior that is symmetric in the components under such permutations [19]. The label switching and the associated identifiability problem have been well-investigated in several Bayesian estimation studies. Proposed solutions include artificial identifiability constraints that involve relabeling of the output of the MCMC sampler based on some component parameters (e.g., sorting of the components based on their means for 1D data) [19], deterministic relabeling algorithms that select a relabeling at each iteration that minimizes the posterior expectation of some loss function [95], [96], and probabilistic relabeling algorithms that take into consideration the uncertainty in the relabeling that

should be selected on each iteration of the MCMC output [97].

Even though the label switching problem also applies to the stochastic search procedures, only a few pattern recognition studies (e.g., only [88], [89] among the ones discussed above) mention its existence during GMM estimation. In particular, Tohka et al. [88] ensured that the components were ordered based on their means in each iteration. This ordering was possible because 1D data were used in the experiments but such artificial identifiability constraints are not easy to establish for multivariate data. Since they have an influence on the resulting estimates, these constraints are also known to lead to over- or under-estimation [19] and create a bias [95]. Chang et al. [89] proposed a greedy solution that sorted the components of a candidate solution based on the distances of the mean vectors of that solution to the mean vectors of a reference solution that achieved the highest fitness value. However, such heuristic orderings depend on the ordering of the components of the reference solution that is also arbitrary and ambiguous.

It is clear that a formulation that involves unique, independently modifiable, and bounded parameters is needed for effective utilization of stochastic search algorithms for the maximum likelihood estimation of unrestricted Gaussian mixture models.

In the third part of this thesis, we present a parameterization based on eigenvalue decomposition of covariance matrices which is suitable for stochastic search algorithms in general, and particle swarm optimization (PSO) algorithm in particular. We develop a new algorithm where global search skills of the PSO algorithm is incorporated into the EM algorithm to do global parameter estimation. In addition to the mathematical derivations, experimental results on synthetic and real-life data sets verifying the performance of the proposed algorithms are provided.

Our major contributions in this part are twofold: we present a novel parameterization for arbitrary covariance matrices where the individual parameters can be independently modified in a stochastic manner during the search process, and describe an optimization formulation for resolving the identifiability problem

for the mixtures. Our first contribution, the parameterization, uses eigenvalue decomposition, and models a covariance matrix in terms of its eigenvalues and Givens rotation angles extracted using QR factorization of the eigenvector matrices via a series of Givens rotations. We show that the resulting parameters are independently modifiable and are bounded so they can be naturally used in different kinds of stochastic global search algorithms. We also describe an algorithm for ordering the eigenvectors so that the parameters of individual Gaussian components are uniquely identifiable. Unlike the existing work that use only the means [89], [90], [91], [92], [94] or means and standard deviations alone [85], [93] in the candidate solutions, this parameterization allows the use of full covariance matrices in the GMM estimation.

As our second major contribution in this part, we propose an algorithm for ordering of the Gaussian components within a candidate solution for obtaining a unique correspondence between two candidate solutions during their interactions for parameter updates throughout the stochastic search. The correspondence identification problem is formulated as a minimum cost network flow optimization problem where the objective is to find the correspondence relation that minimizes the sum of Kullback-Leibler divergences between pairs of Gaussian components, one from each of the two candidate solutions. Our method can be considered as a deterministic relabeling algorithm according to the categorization of label switching solutions as discussed above. We illustrate the proposed parameterization and identifiability solutions using PSO for density estimation. Earlier versions of this part are also described in [98], [99], [100],

One of the most challenging problems of the remote sensing image analysis is the compound object detection problem. Recently available multispectral channels in very high spatial resolution (VHR) images contain a large number of intrinsically heterogeneous structures. We refer to these structures as compound objects. For instance, different kinds of residential areas, commercial areas, and industrial areas which are comprised of various spatial arrangements of primitive objects such as buildings and roads can be considered as compound objects.

There has been a great deal of research in computer vision on the issue of

object representation with a widespread agreement on the object models that are comprised of various spatial arrangements of primitive objects or parts. Representation of compound objects as a collection of spatially related primitive objects or parts has a long history in computer vision [101], [32], [102], [103], [104], [33], [105]. There are two commonly used approaches for object recognition which can simply be classified as probabilistic [104], [105], [55], and deterministic [106], [107] methods. In these methods, first, candidate primitive objects locations and scales are determined using methods for extracting distinctive invariant features from images that can be used to perform reliable matchings between the primitive objects [108], [109], [110]. Second, additional set of local features [111], [32] are extracted around the found candidate primitive object locations. Third, these local features, their locations and scales are put into a some cost function [107], [106] or log-likelihood ratio test function [104], [32] to determine if the compound object of interest is present.

These methods are reported to work well in commercial image databases for the detection of objects such as faces, pedestrians, bicycles, cars, etc [106], [32], [105]. These objects consist of distinctive primitive objects and the proposed algorithms heavily rely on methods for extracting distinctive invariant features to find the candidate primitive object locations. These assumptions do not hold for high-resolution remote sensing images that contain a large number of primitive objects which do not generate distinctive invariant features. Moreover proposed methods can only handle simple geometric relations like left to/right to or nearby [106], [105]. On the other hand, remote sensing images contain tens or hundreds of similar primitive objects as shown in Figure 1.1 and the main distinguishing factor between different compound objects are the different spatial arrangements of the primitive objects.

In the fourth part of this thesis, we present a compound object detection algorithm as an application to the robust constrained Gaussian mixture models. We incorporate the relative size, spectral distribution structure, relative location relations of primitive objects and the independence relations between the location and spectral parts as convex constraints on the source parameters. We formulated the detection problem as the identification of the required number (learned from

Figure 1.1: Compound structures in WorldView-2 images of Ankara and Kusadasi, Turkey.

the reference compound objects) of pixels which are relatively close and have similar spectral and spatial arrangement properties to the primitive objects. The initial version of the algorithm described in this part is also presented in [72]

1.1.1 Summary of Contributions

The main contributions of this thesis are as follows. As our first contribution, we propose a constrained Gaussian mixture model framework which allows us to incorporate prior information about the problem in the form of convex constraints on the parameters. We study the information and the source parameterizations of Gaussian mixture models, and show their relationship using the convex duality theory. Moreover, we provide convex primal and dual problems for the M-step suitable for adding convex constraints on the parameters. As our second contribution, we propose a probabilistic model for the robust estimation of Gaussian mixture models which incorporates the inlier/outlier information available for

small number of data points as convex constraints on the parameters. As our third contribution, we present a novel algorithm where global search skills of the particle swarm optimization algorithm is incorporated into the EM algorithm to do global parameter estimation. As our fourth contribution, we present a new detection algorithm for the compound object detection problem based on robust constrained Gaussian mixture models. The initial versions of the algorithms described in this thesis were also published in [100], [72], [98], [99].

1.2 Organization of the Thesis

The organization of the thesis is as follows.

In Chapter 2, we summarize the necessary mathematical background and introduce the notations used for subsequent developments in this thesis. We describe mathematical principles drawn primarily from two areas: parameter estimation in exponential family models, and convex optimization and duality theory.

In Chapter 3, we consider the constrained Gaussian mixture models which serves as the fundamental modeling framework for the robust density estimation and the compound object detection problems. In this Chapter we discuss the source parameterization and the information parameterization of the Gaussian mixture models and provide an expectation maximization algorithm where convex constraints on the parameters can be handled by solving convex optimization problems for the M-step.

In Chapter 4, we describe a probabilistic model for the robust estimation of Gaussian mixture models which incorporates the inlier/outlier information available for small number of data points as convex constraints on the parameters.

In Chapter 5, we present a stochastic search algorithm framework for the global optimization of the Gaussian mixture model parameters. We describe a new parameterization for the covariance matrices and present a novel algorithm

where global search skills of the particle swarm optimization algorithm is incorporated into the expectation maximization algorithm to do global parameter estimation.

In Chapter 6, we describe a novel algorithm for compound object detection based on robust constrained Gaussian mixture models.

In Chapter 7, we summarize our conclusions and plans for future work.

Chapter 2

Background

This Chapter summarizes the necessary mathematical background and introduces the notations used for subsequent developments in this thesis. We use mathematical principles drawn primarily from two areas: parameter estimation in exponential family models, and convex optimization and duality theory.

Most of the standard discrete and continuous distributions used in practice, such as the Bernoulli, multinomial, Gaussian, exponential, Poisson, etc., and more complicated probabilistic models including fully observed Gaussian mixture models, Bayesian and Markov Networks can be represented in exponential family form [41], [42], [3], [17]. Exponential families and their various properties have been extensively studied and used in statistics, pattern recognition and machine learning [112], [113], [114], [115], [41], [42], [3]. Exponential families provide a general framework for the selection of different parameterizations of distributions by defining different sufficient statistics [7]. Thus, different parameterizations, their geometric structure and various other properties have been extensively studied in the information geometry literature [113], [112], [114], [115]. The exponential family framework also addresses the maximum likelihood (ML) [1], [2], [3] parameter estimation problem for alternative parameterizations and shows which parameterizations lead to easy estimation problems [3], [41], [42] that correspond to convex optimization problems.

Furthermore, estimation of the parameters of additional distributions, such as Gaussian mixture models, that do not belong to the exponential family but can be modeled as marginalized form of an exponential family distribution such as fully observed Gaussian mixture models, can be performed using popular algorithms like the expectation maximization (EM) algorithm [37]. Moreover, using the formalism of exponential families provides us a general framework where various important results can be derived with ease.

Optimization formulations are integral to many disciplines of engineering and science [36], [35], [116], [117], [118], [119], [79], [81], [120], [121], [122], [123], [124], [74], [75], [76], [125], [80], [82], [126], [127]. Convex optimization [36], [35], [116], [128], [129], [34] uses the ideas from convex analysis [130], [131] which at a simplistic level is the study of properties of convex functions and convex sets. With the advancement of powerful algorithms [36], [132], [118], [133], [134], [134], [135], [136], [137], [138], [139] and software packages [140], [141], [142], [143] for specifying and solving convex optimization problems, this class of problems can be solved globally and efficiently. Furthermore, convex analysis and duality theory of which there are various closely related forms (Fenchel/Legendre and Lagrangian duality) [144], [145], plays a significant role in the analysis of optimization problems. What is more, duality theory not only introduces conceptual insights to the optimization problems but also provides important practical methods for developing optimization algorithms. There exists a huge literature on convex optimization and convex optimization based heuristics for solving nonconvex optimization problems [36], [35], [116], [128], [129], [34].

There are strong connections between exponential family distributions and convex optimization. For instance, maximum entropy distributions subject to linear constraints take exponential family form [4], [5], [6], [8], [7]. Moreover, there exist two different parameterizations called the natural and the moment parameterizations for a given exponential family distribution which are connected via the Fenchel duality relation [5] between the log partition and the entropy functions [5]. This duality relation (or more precisely the gradients of the log partition and the negative entropy functions) provides mappings between the two parameterizations. Furthermore, the maximum likelihood (ML) [1], [2], [3] and the

maximum entropy (ME) [4], [5], [6], [7], [8] parameter estimation problems are related through Lagrangian duality. In this case, the maximum likelihood problems correspond to a convex optimization problem in the natural parameters as optimization variables and the maximum entropy problems lead to a convex optimization problem in the moment parameters as optimization variables. Mappings between the natural and the moment parameters are provided by the gradients of the log partition and the entropy functions due to the Fenchel duality relation.

Convexity of the parameter estimation problems for exponential families breaks down in the existence of hidden (unobserved) variables. The expectation maximization algorithm [37], which can be interpreted as doing alternating optimization on a surrogate bound function [38], is widely used for parameter estimation problems with hidden variables. In such estimation problems, Fenchel duality provides a mathematically principled way to obtain the bound function on the log-likelihood of the marginal distribution of the observed variables as a function of the log-likelihood of joint distribution of both observed and hidden variables. What is more, convex optimization provides an explanation for why parameter estimation problems for exponential family distributions are easier using such bounds.

The following Sections briefly summarize the key ideas and notations used to present the main mathematical results in this thesis. The notation used to describe the optimization problems with very basic definitions and properties of convex sets and functions are introduced in Section 2.1. Section 2.2 presents the Fenchel/Legendre conjugate function and the Lagrangian duality. The maximum likelihood and the maximum entropy principles used for parameter estimation are given in Section 2.3. Exponential family distributions and their important properties used in this thesis are described in Section 2.4.

2.1 Optimization Problems

Optimization in general and convex optimization in particular is of central importance in this thesis. There are lots different notations and definitions used in the optimization literature [36], [130], [132], [131], [128], [129]. In this thesis we follow the notation used by [36]. Furthermore, properties of convex sets and functions are heavily used in this thesis. We will give the definitions and describe the properties that play significant roles. For the rest, we will cite the relevant sources. We use the optimization and convex optimization problem definitions given in [36].

Definition 1. *An optimization problem has the form*

$$\begin{aligned} &\textit{minimize } \ f_0(\mathbf{x}) \\ &\textit{subject to } f_i(\mathbf{x}) \leq b_i, \ i = 1, \ldots, m \\ &\textit{subject to } h_i(\mathbf{x}) = 0, \ i = 1, \ldots, p \end{aligned} \tag{2.1}$$

where the vector $\mathbf{x} \in \mathbb{R}^n$ *is called the optimization variable, and the function* $f_0 : \mathbb{R}^n \to \mathbb{R}$ *is called the objective function or cost function. The functions* $f_i : \mathbb{R}^n \to \mathbb{R}$, $i = 1, \ldots, m$ *are called the inequality constraint functions and the inequalities* $f_i(\mathbf{x}) \leq b_i$, $i = 1, \ldots, m$ *are called the inequality constraints. The constants* b_i, $i = 1, \ldots, m$ *are called limits or bounds on the inequality constraints. The functions* $h_i : \mathbb{R}^n \to \mathbb{R}$, $i = 1, \ldots, p$ *are called the equality constraint functions and the equalities* $h_i(\mathbf{x}) = 0$, $i = 1, \ldots, p$ *are called the equality constraints. A vector* $\mathbf{x}^*$ *is called optimal, or an optimal solution of the problem in* (2.1)*, if it has the smallest objective value among all vectors that satisfy the constraints.*

In practice, optimization problems sometimes arise as maximization of some objective function. Maximization problems can be put into the form in (2.1) as minimization of the negative of the objective function.

2.1.1 Convex Optimization Problems

Before we define the special class of optimization problems called convex optimization problems, first we will give simple definitions of convex sets and convex functions.

Definition 2. *Let $\mathcal{S}$ denote a set. If for any $\mathbf{x},\ \mathbf{y} \in \mathcal{S}$ and any λ where $0 \leq \lambda \leq 1$, we have*

$$\lambda\mathbf{x} + (1-\lambda)\mathbf{y} \in \mathcal{S}, \tag{2.2}$$

set $\mathcal{S}$ is convex.

Definition 3. *Let $f : \mathbb{R}^n \to \mathbb{R}$ be a function. If the domain of f ($\mathbf{dom}\, f$) is a convex set and if for all $\mathbf{x},\ \mathbf{y} \in \mathbf{dom}\, f$, and λ where $0 \leq \lambda \leq 1$, we have*

$$f(\lambda\mathbf{x} + (1-\lambda)\mathbf{y}) \leq \lambda f(\mathbf{x}) + (1-\lambda)f(\mathbf{y}), \tag{2.3}$$

function f is convex.

The inequality in (2.3) extends to integrals where λ is replaced by a continuous function of $\mathbf{x}$. This general form is widely known as the Jensen's inequality.

Definition 4. *Let $\mathbf{x}$ be a random vector with pdf $p(\mathbf{x}) \geq 0$ taking values in the sample space $\Omega_{\mathbf{x}}$ where $\int_{\Omega_{\mathbf{x}}} p(\mathbf{x})d\mathbf{x} = 1$. A function $f : \mathbb{R}^n \to \mathbb{R}$ is convex, if it satisfies the Jensen's inequality*

$$f\Big(\int_{\Omega_{\mathbf{x}}} p(\mathbf{x})\mathbf{x}d\mathbf{x}\Big) \leq \int_{\Omega_{\mathbf{x}}} p(\mathbf{x})f(\mathbf{x})d\mathbf{x}, \tag{2.4}$$

$$f(E[\mathbf{x}]) \leq E[f(\mathbf{x})]. \tag{2.5}$$

Another useful inequality is the Holder's inequality

Definition 5. *Holder's inequality*

$$\int f(\mathbf{x})g(\mathbf{x})d\mathbf{x} \leq \Big(\int |f(\mathbf{x})|^p d\mathbf{x}\Big)^{1/p}\Big(\int |f(\mathbf{x})|^q d\mathbf{x}\Big)^{1/q} \tag{2.6}$$

Finally, we define convex optimization problems using convex objective and constraint functions.

Definition 6. *Optimization problems where the convex objective function is minimized or concave objective function is maximized subject to constraints where the inequality constraint functions are convex and the equality constraint functions are affine are defined as convex optimization problems.*

2.2 Convex Duality

2.2.1 Fenchel Duality

Definition 7. *The Fenchel conjugate function of $f : \mathbb{R}^n \to \mathbb{R}$ is $f^* : \mathbb{R}^n \to \mathbb{R}$, where f^* is defined as*

$$f^*(\nu) = \sup_{\theta \in \mathbf{dom}\, f} \theta^T \nu - f(\theta). \tag{2.7}$$

The values $\nu \in \mathbb{R}^n$ where the supremum is finite determines the domain of the conjugate function. The conjugate function f^* of differentiable f is also known as the Legendre transform of f where

$$f^*(\nu) = [\theta^T \nu - f(\theta)]_{\nu = \nabla_\theta f(\theta)}. \tag{2.8}$$

Corollary 1. *By definition, since $f^*(\nu)$ is a supremum of affine functions of ν, it is always convex, even if $f(\theta)$ is not a convex function of θ [36].*

In addition, if f is also convex and its epigraph is a closed set, then the conjugate of the conjugate function is equal to the original function, i.e., $f^{**} = f$ [36], [130]. In this case, for any θ and ν, $f(\theta)$ and $f^*(\nu)$ are related through the Fenchel inequality.

Definition 8. *The Fenchel-Young inequality is given as*

$$f^*(\nu) + f(\theta) - \theta^T \nu \ \geq 0. \tag{2.9}$$

Furthermore, the Fenchel inequality holds with equality when we have $\nabla_\theta f(\theta) = \nu$ and $\nabla_\nu f^*(\nu) = \theta$. This follows from the definition of the conjugate function. Note that the gradients $\nabla_\theta f(\theta)$ and $\nabla_\nu f^*(\nu)$ also provide gradient mappings between the parameters θ and ν. In particular, $\nabla_\theta f(\theta)$ maps θ to ν, whereas $\nabla_\nu f^*(\nu)$ provides an inverse mapping from ν to θ.

2.2.2 Lagrangian Duality

Definition 9. *Consider an optimization problem of the form* (2.1)*. Assume that the domain of the problem* $\mathbf{dom}\, P = \left(\bigcap_{i=0}^{m} \mathbf{dom}\ f_i\right) \cap \left(\bigcap_{i=1}^{p} \mathbf{dom}\ h_i\right)$ *is nonempty and let* p^* *denote the optimal value. Lagrangian* $L : \mathbb{R}^n \times \mathbb{R}^m \times \mathbb{R}^p \to \mathbb{R}$ *of the problem* (2.1) *is defined as*

$$L(\mathbf{x}, \lambda, \nu) = f_0(\mathbf{x}) + \sum_i \lambda_i (f_i(\mathbf{x}) - b_i) + \sum_i \nu_i h_i(\mathbf{x}) \tag{2.10}$$

where $\mathbf{dom}\ L = \mathbf{dom}\ P \times \mathbb{R}^m \times \mathbb{R}^p$*. The variables* λ_i*,* $i = 1, \ldots, m$ *and* ν_i*,* $i = 1, \ldots, p$ *are called the Lagrange multipliers or dual variables.*

Definition 10. *The Lagrange dual function* $g : \mathbb{R}^m \times \mathbb{R}^p \to \mathbb{R}$ *is defined as the minimum value of the Lagrangian* (2.10) *over* $\mathbf{x}$ *for any* λ *and* ν*:*

$$g(\lambda, \nu) = \inf_{\mathbf{x} \in \mathbf{dom}\ P} L(\mathbf{x}, \lambda, \nu). \tag{2.11}$$

Corollary 2. *The dual function* $g(\lambda, \nu)$ *in* (2.11) *is a concave function of* λ *and* ν *since it is defined as the infimum of affine functions of* λ *and* ν*.*

Definition 11. *The Lagrange dual optimization problem of* (2.1) *is defined as*

$$\begin{aligned} &\textit{maximize}\ \ g(\lambda, \nu) \\ &\textit{subject to}\ \lambda \geq 0. \end{aligned} \tag{2.12}$$

Corollary 3. *The Lagrange dual optimization problem in* (2.12) *is a convex optimization problem since a concave objective function is to be maximized and the constraints are convex.*

2.3 Parameter Estimation

In this Section, we will introduce two popular principles called the maximum likelihood (ML) [1], [2], [3] and the maximum entropy (ME) [4], [5], [6], [7], [8] used for parameter estimation.

In estimation problems, we are given a data set of N random vectors $\mathcal{X} = \{\mathbf{x}_1, \ldots, \mathbf{x}_N\}$, $\mathcal{X} \in \Omega_{\mathcal{X}}$, corresponding to a random sample from an unknown distribution. In general, it is convenient to use the empirical distribution of the data set $\mathcal{X}$.

Definition 12. *Empirical density $\tilde{p}(\mathbf{x})$ of a set of N observations $\mathcal{X} = \{\mathbf{x}_1, \ldots, \mathbf{x}_N\}$ is defined as*

$$\tilde{p}(\mathbf{x}) = \sum_{j=1}^{N} \frac{1}{N} \delta(\mathbf{x} - \mathbf{x}_j) \tag{2.13}$$

where for $\mathbf{x}_j \in \Omega_{\mathbf{x}_j}$, $\delta(\mathbf{x} - \mathbf{x}_j)$ denotes the Dirac delta function for continuous sample space and Kronecker delta function for discrete sample space.

Similarly, we often use averages of some function of the data points $\mathcal{X}$ which are addressed as empirical moments.

Definition 13. *For a specified statistics function $\phi : \Omega_{\mathbf{x}} \to \mathbb{R}^d$, the expected statistics with respect to the empirical distribution $\tilde{p}(\mathbf{x})$, i.e., $E_{\tilde{p}(\mathbf{x})}[\phi(\mathbf{x})]$, is defined as the empirical moment.*

2.3.1 Maximum Likelihood Principle

Suppose we are given a data set of N random vectors $\mathcal{X} = \{\mathbf{x}_1, \ldots, \mathbf{x}_N\} \in \Omega_{\mathcal{X}}$ corresponding to a random sample from a parametric probability density function $p(\mathcal{X}|\theta) \in \mathcal{F}$ belonging to a family of probability distributions $\mathcal{F}$ parametrized by the parameter $\theta \in \mathcal{C}_\theta$ where $\mathcal{C}_\theta$ is called the parameter space and denotes the values the parameter θ can take so that $p(\mathcal{X}|\theta) \in \mathcal{F}$ is a valid distribution. When

$p(\mathcal{X}|\theta)$ is considered as a function of the parameter θ, it is called the likelihood function, and is denoted by $\mathcal{L}(\theta|\mathcal{X})$ as

$$\mathcal{L}(\theta|\mathcal{X}) = p(\mathcal{X}|\theta). \tag{2.14}$$

In general, it is more convenient to work with the log-likelihood function $\ell(\theta|\mathcal{X})$ as

$$\ell(\theta|\mathcal{X}) = \log p(\mathcal{X}|\theta). \tag{2.15}$$

Definition 14. *The maximum likelihood principle [1], [2], [3] states that the best estimate $\hat{\theta}$ of the parameter θ maximizes the log-likelihood $\ell(\theta|\mathcal{X})$ of the data $\mathcal{X}$ as*

$$\hat{\theta} = \underset{\theta}{arg\ max}\, \ell(\theta|\mathcal{X}) \tag{2.16}$$

where there is an implicit constraint $\theta \in \mathcal{C}_\theta$ incorporated into the domain of the objective function $\ell(\theta|\mathcal{X})$

$$\ell(\theta|\mathcal{X}) = \begin{cases} \log p(\mathcal{X}|\theta), & if\ \theta \in C_\theta, \\ -\infty, & otherwise. \end{cases} \tag{2.17}$$

If the data set of random vectors $\mathcal{X}$ are independent and distributed according to parametric probability density functions $p(\mathbf{x}_j|\theta_j)$, log-likelihood function $\ell(\Theta|\mathcal{X})$ can be simplified as follows

$$\begin{aligned} \ell(\theta|\mathcal{X}) &= \log p(\mathcal{X}|\theta) \\ &= \log p(\mathbf{x}_1, \ldots, \mathbf{x}_N|\theta_1, \ldots, \theta_N) \\ &= \log \prod_{j=1}^{N} p(\mathbf{x}_j|\theta_j) \\ &= \sum_{j=1}^{N} \log p(\mathbf{x}_j|\theta_j) \\ &= \sum_{j=1}^{N} \ell(\theta_j|\mathbf{x}_j) \end{aligned} \tag{2.18}$$

In practice, the data set of the random vectors $\mathcal{X}$ are independent and identically distributed (i.i.d.) according to the probability density function $p(\mathbf{x}|\theta)$. This further simplifies (2.18) as

$$\begin{aligned}\ell(\theta|\mathcal{X}) &= \sum_{j=1}^{N} \log p(\mathbf{x}_j|\theta_j) \\ &= \sum_{j=1}^{N} \log p(\mathbf{x}_j|\theta) \\ &= \sum_{j=1}^{N} \ell(\theta|\mathbf{x}_j). \end{aligned} \tag{2.19}$$

In this thesis, we will generally consider the optimization problem corresponding to the ML parameter estimation for an i.i.d. data set $\mathcal{X}$ of N random vectors in the following form

$$\text{minimize} \quad -\frac{1}{N}\sum_{j=1}^{N} \ell(\theta|\mathbf{x}_j) \tag{2.20}$$

where the distribution parameter θ is the optimization variable and the log-likelihood of the j'th data point $\mathbf{x}_j$ is denoted by $\ell(\theta|\mathbf{x}_j) = \log p(\mathbf{x}_j|\theta)$.

2.3.2 Maximum Entropy Principle

In this Section, we will introduce the maximum entropy principle used for parameter estimation. First we will give simple definitions of the entropy function and the Kullback-Leibler (KL) divergence. Entropy is used as a measure of randomness or uncertainty for probability distributions and KL divergence is a metric used to measure similarity between two probability distributions [146], [147], [8].

Definition 15. *Given a probability distribution $p(\mathbf{x})$ defined on some sample space $\Omega_{\mathbf{x}}$, the (differential) entropy [8] is defined as*

$$H(p(\mathbf{x})) = -\int_{\Omega_{\mathbf{x}}} p(\mathbf{x}) \log p(\mathbf{x}) d\mathbf{x}. \tag{2.21}$$

For discrete spaces $\Omega_{\mathbf{x}}$, $d\mathbf{x}$ is taken to be a counting measure so that the equation is written with a sum rather than an integral.

Definition 16. *The relative entropy or Kullback-Leibler divergence between two probability distributions $p(\mathbf{x})$ and $q(\mathbf{x})$ defined on some sample space $\Omega_{\mathbf{x}}$ is defined as*

$$D(p||q) = \int_{\Omega_{\mathbf{x}}} p(\mathbf{x}) \log \frac{p(\mathbf{x})}{q(\mathbf{x})} d\mathbf{x}. \tag{2.22}$$

Suppose we are given a data set of N random vectors $\mathcal{X} = \{\mathbf{x}_1, \ldots, \mathbf{x}_N\}$ where all random vectors take values in the same sample space $\Omega_{\mathbf{x}}$, i.e., $\mathbf{x}_j \in \Omega_{\mathbf{x}}$ for $j = 1, \ldots, N$. Let $\phi : \Omega_{\mathbf{x}} \to \mathbb{R}^d$ denote the statistics function and $E_{\tilde{p}(\mathbf{x})}[\phi(\mathbf{x})] = \nu_s$ denote the empirical moment where the expectation is taken with respect to the empirical distribution of the data set denoted by $\tilde{p}$. Consider a parametric probability density function $p(\mathbf{x}|\nu) \in \mathcal{F}$ belonging to a family of probability distributions $\mathcal{F}$ parametrized by the moment parameter $\nu = E_{p(\mathbf{x}|\nu)}[\phi(\mathbf{x})]$. Let $\mathcal{C}_\nu$ denote the realizable moment parameter space.

Definition 17. *The maximum entropy (ME) principle [4], [5], [6], [8], [7] states that the best estimate $\hat{\nu}$ of the parameter ν maximizes the entropy $H(\nu)$ of the distribution $p(\mathbf{x}|\nu)$ subject to the linear moment constraints $\nu = \nu_s$ as*

$$\hat{\nu} = \underset{\nu=\nu_s}{arg\ max}\, H(\nu) \tag{2.23}$$

where there is an implicit constraint $\nu \in \mathcal{C}_\nu$ incorporated into the domain of the objective function $H(\nu)$

$$H(\nu) = \begin{cases} H(\nu), & if\ \nu \in \mathcal{C}_\nu, \\ -\infty, & otherwise. \end{cases} \tag{2.24}$$

Here $\nu \in \mathbb{R}^d$ is the optimization variable and $\nu = \nu_s$ corresponds to the linear moment constraints on ν. In other words, the ME principle aims to find the moment parameter estimate $\hat{\nu}$ that leads to the least informative distribution $p(\mathbf{x}|\nu)$ among the family of distributions $\mathcal{F}$ that is consistent with the specified moment constraints $\nu = \nu_s$.

2.4 Exponential Family Models

Exponential family models are of central importance to this thesis. The distribution of the fully observed Gaussian mixture models [41], [42] can be represented in exponential family form. Hence, the marginal distribution of the observed variables can be viewed as a marginal distribution of an exponential family distribution. Using the exponential family framework with duality theory illuminates various connections between different parameterizations of Gaussian mixture models. In addition, it provides conceptual insights for the bound function used in the expectation maximization algorithm to do parameter estimation.

A broad class of probabilistic models can be represented in exponential family form [41], [42], [3], [148], [149], [150], [151], [152], [44], [47], [153]. Exponential family models have lots of nice features and are studied extensively in the statistics literature [112], [113], [114], [115]. In this thesis, we are mainly interested in parameter estimation problem for Gaussian mixture models. Hence we will describe a minimum set of properties of exponential family models that are important for the parameter estimation. In Section 2.4.1 we will define exponential family distributions and introduce the natural and the moment parameters. In Section 2.4.2 we will show that the log partition function is a convex function of the natural parameters and the gradient of the log partition function provides a mapping from the natural parameters to the moment parameters. In Section 2.4.3 we derive the Fenchel duality relation between the log partition and the entropy functions. In Section 2.4.4 we will introduce the maximum likelihood (ML) and the maximum entropy (ME) principles for parameter estimation and in Section 2.4.5 we will show that the ML and the ME problems are dual problems using Lagrangian duality. In Section 2.4.6 we will introduce multinomial and Gaussian distributions.

2.4.1 Exponential Family Distributions

Definition 18. *A set $\mathcal{F}$ of parametrized distributions over a random vector $\mathbf{x}$ taking values in some sample space $\Omega_{\mathbf{x}}$ of the form*

$$p(\mathbf{x}|\theta) = \exp(\theta^T \phi(\mathbf{x}) - \Phi(\theta)) \tag{2.25}$$

is called exponential family where $\theta \in \mathbb{R}^n$ are called the natural parameters, $\phi : \Omega_{\mathbf{x}} \to \mathbb{R}^n$ are called the sufficient statistics, $\theta^T \phi(\mathbf{x})$ denote the Euclidean inner product in $\mathbb{R}^n$ and $\Phi(\theta)$ is called the log partition function

$$\Phi(\theta) = \log \int_{\mathbf{x} \in \Omega_{\mathbf{x}}} \exp(\theta^T \phi(\mathbf{x})) d\mathbf{x} \tag{2.26}$$

which serves to normalize the distribution to 1. For discrete spaces, $d\mathbf{x}$ is taken to be a counting measure so that log partition $\Phi(\theta)$ is written with a sum rather than an integral. We denote the set of all parameters θ where $\Phi(\theta)$ is well-defined with $\mathcal{C}_\theta = \{\theta \in \mathbb{R}^n | \Phi(\theta) < \infty\}$. For regular exponential family $\mathcal{F} = \{p(\mathbf{x}|\theta) | \theta \in \mathcal{C}_\theta\}$, the set of parameters $\mathcal{C}_\theta$ is an open convex set in $\mathbb{R}^n$.

Expected value of sufficient statistic function $\phi : \Omega_{\mathbf{x}} \to \mathbb{R}^n$ is defined to be the moment of the probability distribution $p(\mathbf{x}|\theta) \in \mathcal{F}$. Associated with the sufficient statistic function $\phi(\mathbf{x}) \in \mathbb{R}^n$, there is a moment parameter $\nu \in \mathbb{R}^n$ which is defined by the expectation

$$\nu = E_{p(\mathbf{x}|\theta)}[\phi(\mathbf{x})]. \tag{2.27}$$

We denote the set of all realizable moment parameters with $\mathcal{C}_\nu = \{\nu \in \mathbb{R}^d | \nu = E_{p(\mathbf{x}|\theta)}[\phi(\mathbf{x})], p(\mathbf{x}|\theta) \in \mathcal{F}\}$.

2.4.2 Log Partition and Entropy Functions

We can see the relation between the moment parameters $\nu \in \mathcal{C}_\nu$ and the natural parameters $\theta \in \mathcal{C}_\theta$ using the moment generating property of the log partition function $\Phi(\theta)$.

Proposition 1. *Gradient of the log partition function $\Phi(\theta)$ with respect to natural parameters θ is equal to the moment parameters ν.*

Proof.

$$\begin{aligned}
\nabla_\theta \Phi(\theta) =& \frac{\int_{\mathbf{x}\in\Omega_\mathbf{x}} \phi(\mathbf{x}) \exp(\theta^T \phi(\mathbf{x})) d\mathbf{x}}{\int_{\hat{\mathbf{x}}\in\Omega_\mathbf{x}} \exp(\theta^T \phi(\hat{\mathbf{x}})) d\hat{\mathbf{x}}} \\
=& \frac{\int_{\mathbf{x}\in\Omega_\mathbf{x}} \phi(\mathbf{x}) \exp(\theta^T \phi(\mathbf{x})) d\mathbf{x}}{\exp\log \int_{\hat{\mathbf{x}}\in\Omega_\mathbf{x}} \exp(\theta^T \phi(\hat{\mathbf{x}})) d\hat{\mathbf{x}}} \\
=& \frac{\int_{\mathbf{x}\in\Omega_\mathbf{x}} \phi(\mathbf{x}) \exp(\theta^T \phi(\mathbf{x})) d\mathbf{x}}{\exp \Phi(\theta)} \\
=& \int_{\mathbf{x}\in\Omega_\mathbf{x}} \phi(\mathbf{x}) \exp(\theta^T \phi(\mathbf{x}) - \Phi(\theta)) d\mathbf{x} \\
=& E_{p(\mathbf{x}|\theta)}[\phi(\mathbf{x})] \\
=& \nu
\end{aligned} \tag{2.28}$$

□

Notice that the gradient of the log partition function $\nabla_\theta \Phi(\theta)$ provides a mapping $\nabla_\theta \Phi : \mathcal{C}_\theta \to \mathcal{C}_\nu$ from the natural parameters $\theta \in \mathcal{C}_\theta$ to the moment parameters $\nu \in \mathcal{C}_\nu$. This property implies that the moment parameters can also be used to characterize exponential family distributions.

As we will make it clear later, from the maximum likelihood estimation point of view, the most important property of the log partition function $\Phi(\theta)$ is its being a convex function of the natural parameters θ.

Proposition 2. *The log partition function $\Phi(\theta)$ is a convex function of the natural parameters θ.*

Proof. To prove that the log partition function $\Phi(\theta)$ is a convex function of the natural parameters θ, we show $\Phi(\alpha\theta_1 + (1-\alpha)\theta_2) \leq \alpha\Phi(\theta_1) + (1-\alpha)\Phi(\theta_2)$ using

the Holder's inequality (2.6).

$$\begin{aligned}
\Phi(\alpha\theta_1 + (1-\alpha)\theta_2) &= \log \int_{\mathbf{x}\in\Omega_\mathbf{x}} \exp\big((\alpha\theta_1 + (1-\alpha)\theta_2)^T \phi(\mathbf{x})\big) d\mathbf{x} \\
&= \log \int_{\mathbf{x}\in\Omega_\mathbf{x}} \exp\big(\alpha\theta_1^T \phi(\mathbf{x}) + (1-\alpha)\theta_2^T \phi(\mathbf{x})\big) d\mathbf{x} \\
&= \log \int_{\mathbf{x}\in\Omega_\mathbf{x}} \exp\big(\alpha\theta_1^T \phi(\mathbf{x})\big) \exp\big((1-\alpha)\theta_2^T \phi(\mathbf{x})\big) d\mathbf{x} \\
&\leq \log \Big[\Big(\int_{\mathbf{x}\in\Omega_\mathbf{x}} \exp\big(\frac{\alpha}{\alpha}\theta_1^T \phi(\mathbf{x})\big) d\mathbf{x} \Big)^{\alpha} \\
&\qquad \Big(\int_{\mathbf{x}\in\Omega_\mathbf{x}} \exp\big(\frac{1-\alpha}{1-\alpha}\theta_2^T \phi(\mathbf{x})\big) d\mathbf{x} \Big)^{1-\alpha} \Big] \\
&= \log \Big[\Big(\int_{\mathbf{x}\in\Omega_\mathbf{x}} \exp(\theta_1^T \phi(\mathbf{x})) d\mathbf{x} \Big)^{\alpha} \\
&\qquad \Big(\int_{\mathbf{x}\in\Omega_\mathbf{x}} \exp(\theta_2^T \phi(\mathbf{x})) d\mathbf{x} \Big)^{1-\alpha} \Big] \\
&= \alpha \log \Big(\int_{\mathbf{x}\in\Omega_\mathbf{x}} \exp(\theta_1^T \phi(\mathbf{x})) d\mathbf{x} \Big) \\
&\quad + (1-\alpha) \log \Big(\int_{\mathbf{x}\in\Omega_\mathbf{x}} \exp(\theta_2^T \phi(\mathbf{x})) d\mathbf{x} \Big) \\
&= \alpha\Phi(\theta_1) + (1-\alpha)\Phi(\theta_2) \qquad (2.29)
\end{aligned}$$

□

2.4.3 Fenchel Duality

There is a close relationship between the entropy function and the log partition function. In particular, the log partition function and the negative entropy function are Fenchel conjugate functions.

Proposition 3. *The log partition function $\Phi(\theta)$ and the negative entropy function $H(\nu)$ are Fenchel conjugate functions*

$$-H(\nu) = \sup_{\theta\in\mathbf{dom}\,\Phi} \theta^T\nu - \Phi(\theta) \qquad (2.30)$$

and

$$\Phi(\theta) = \sup_{\nu\in\mathbf{dom}\,H} \theta^T\nu + H(\nu). \qquad (2.31)$$

Proof. First we notice that the entropy of exponential family distributions as an affine function of the moment parameters ν can be written as

$$\begin{aligned} H(\nu) &= -E_{p(\mathbf{x}|\theta)}[\log p(\mathbf{x}|\theta)] \\ &= -E_{p(\mathbf{x}|\theta)}[\theta^T \phi(x) - \Phi(\theta)] \\ &= -\theta^T \nu + \Phi(\theta) \end{aligned} \tag{2.32}$$

We can rewrite the entropy equation (2.32) as follows:

$$\Phi(\theta) - H(\nu) - \theta^T \nu = 0 \tag{2.33}$$

Here, we notice that equation (2.33) actually corresponds to the Fenchel-Young inequality in (2.9) between the log partition function $\Phi(\theta)$ and the negative of the entropy function $H(\nu)$ holding with equality. Recall from the relation in (2.8) that equation (2.30) achieves the supremum when $\nu = \nabla_\theta \Phi(\theta)$ and we showed the moment generating property of the log partition function in Proposition 1, i.e., we have $\nu = \nabla_\theta \Phi(\theta)$. Hence, the Fenchel conjugacy relation (2.30) is true. For (2.31), recall from Proposition 2 that the log partition function $\Phi(\theta)$ is a convex function of θ, thus we conclude that the conjugate of the negative entropy function is the log partition function, i.e., $(-H(\nu))^* = \Phi(\theta)^{**} = \Phi(\theta)$. □

As a result we can write the Fenchel inequality for the log partition function $\Phi(\theta)$ and the negative entropy function $-H(\nu)$.

Corollary 4. *The log partition function $\Phi(\theta)$ and the entropy function $H(\nu)$ satisfy the following inequality*

$$\Phi(\theta) - H(\nu) - \theta^T \nu \geq 0 \tag{2.34}$$

for all θ, ν.

Proposition 4. *Fenchel duality relations between the log partition function $\Phi(\theta)$ and the entropy function $H(\nu)$ can be written as*

$$H(\nu) = \inf_{\theta \in \mathbf{dom}\, \Phi} \Phi(\theta) - \theta^T \nu \tag{2.35}$$

and

$$\Phi(\theta) = \sup_{\nu \in \mathbf{dom}\, H} H(\nu) + \nu^T \theta \tag{2.36}$$

Proof. For the relation (2.35) we have

$$\begin{aligned} -H(\nu) &= \sup_{\theta \in \mathbf{dom}\, \Phi} \theta^T \nu - \Phi(\theta) \\ H(\nu) &= - \sup_{\theta \in \mathbf{dom}\, \Phi} \theta^T \nu - \Phi(\theta) \\ &= \inf_{\theta \in \mathbf{dom}\, \Phi} -\theta^T \nu + \Phi(\theta) \\ &= \inf_{\theta \in \mathbf{dom}\, \Phi} \Phi(\theta) - \theta^T \nu \end{aligned} \tag{2.37}$$

For the relation (2.36), since (2.36) is same as (2.31), the statement is correct. □

As shown in Proposition 2 and Corollary 1, the log partition function $\Phi(\theta)$ is a convex function of θ and the negative entropy function, $-H(\nu)$, is a convex function of ν. Since the Fenchel-Young inequality holds with equality in (2.33) as discussed in Definition 7, the natural parameters θ and the moment parameters ν are related through gradient pairs $\nabla_\theta \Phi(\theta), -\nabla_\nu H(\nu)$. In particular the gradient of the log partition function $\nabla_\theta \Phi(\theta)$ provides a mapping $\nabla \Phi : \mathcal{C}_\theta \to \mathcal{C}_\nu$ from the natural parameters $\theta \in \mathcal{C}_\theta$ to the moment parameters $\nu \in \mathcal{C}_\nu$ and the gradient of the negative entropy function $-\nabla_\nu H(\nu)$ provides a mapping $-\nabla H : \mathcal{C}_\nu \to \mathcal{C}_\theta$ from the moment parameters $\nu \in \mathcal{C}_\nu$ to the natural parameters $\theta \in \mathcal{C}_\theta$.

2.4.4 Parameter Estimation for Exponential Family

2.4.4.1 ML Estimation

We consider the exponential family $\mathcal{F}$ of distributions $p(\mathbf{x}|\theta) \in \mathcal{F}$ over a random vector $\mathbf{x}$ taking values in the sample space $\Omega_\mathbf{x}$ parameterized by the natural parameters $\theta \in \mathbb{R}^n$ with the sufficient statistic function $\phi : \Omega_\mathbf{x} \to \mathbb{R}^n$. Given a data set $\mathcal{X} = \{\mathbf{x}_1, \ldots, \mathbf{x}_N\}$ of N independent and identically distributed (i.i.d.) random vectors corresponding to random samples from the distribution $p(\mathbf{x}|\theta) \in$

$\mathcal{F}$, our objective is to find the maximum likelihood (ML) estimate $\hat{\theta} \in \mathbb{R}^n$ of the natural parameters θ. The ML estimation problem can be written in minimization form (2.20) as

$$\hat{\theta} = \arg\min_{\theta} \quad -\frac{1}{N}\sum_{j=1}^{N} \ell(\theta|\mathbf{x}_j). \tag{2.38}$$

For exponential family distributions $p(\mathbf{x}|\theta) \in \mathcal{F}$, the individual log-likelihoods $\ell(\theta|\mathbf{x}_j)$ can be expressed as

$$\begin{aligned} \ell(\theta|\mathbf{x}_j) &= \log \exp\left(\theta^T \phi(\mathbf{x}_j) - \Phi(\theta)\right) \\ &= \theta^T \phi(\mathbf{x}_j) - \Phi(\theta). \end{aligned} \tag{2.39}$$

The overall objective function is then given by

$$\begin{aligned} -\frac{1}{N}\sum_{j=1}^{N} \ell(\theta|\mathbf{x}_j) &= -\frac{1}{N}\sum_{j=1}^{N} \Big(\theta^T \phi(\mathbf{x}_j) - \Phi(\theta)\Big) \\ &= \Phi(\theta) - \theta^T \Big(\frac{1}{N}\sum_{j=1}^{N} \phi(\mathbf{x}_j)\Big) \\ &= \Phi(\theta) - \theta^T \nu_s \end{aligned} \tag{2.40}$$

where $\nu_s = \frac{1}{N}\sum_{j=1}^{N} \phi(\mathbf{x}_j)$ denote the empirical moments.

Now, we can write the corresponding optimization problem as

$$\text{minimize} \;\; \Phi(\theta) - \theta^T \nu_s \tag{2.41}$$

where $\theta \in \mathbb{R}^n$ is the optimization variable. There is an implicit constraint $\theta \in \mathcal{C}_\theta$ denoting the convex set of parameter values where the log partition function $\Phi(\theta)$ is well-defined incorporated into the domain of $\Phi(\theta)$.

Proposition 5. *The maximum likelihood estimation problem* (2.41) *for exponential family distributions $p(\mathbf{x}|\theta) \in \mathcal{F}$ is a convex optimization problem in optimization variables θ.*

Proof. As shown in Proposition 2, the log partition function defined over the convex set $\mathcal{C}_\theta$ is a convex function of θ. Notice that $-\theta^T \nu_s$ is a linear function of θ, and since convex function plus a linear function is convex [36], we conclude that the maximum likelihood (ML) problem in (2.41) is a convex optimization problem in the variable θ. □

2.4.4.2 ME Estimation

We consider the family $\mathcal{F}$ of distributions $p(\mathbf{x}|\nu) \in \mathcal{F}$ over a random vector $\mathbf{x}$ taking values in the sample space $\Omega_{\mathbf{x}}$ parameterized by the moment parameters $\nu \in \mathbb{R}^n$ with the sufficient statistic function $\phi : \Omega_{\mathbf{x}} \to \mathbb{R}^n$ where $\nu = E_{p(\mathbf{x}|\nu)}[\phi(\mathbf{x})]$. Given a data set $\mathcal{X} = \{\mathbf{x}_1, \ldots, \mathbf{x}_N\}$ of N random vectors taking values in the same sample space $\Omega_{\mathbf{x}}$ with empirical moment $\nu_s = \frac{1}{N}\sum_{j=1}^{N} \phi(\mathbf{x}_j)$, our objective is to find the maximum entropy (ME) (2.23) estimate $\hat{\nu} \in \mathbb{R}^n$ of the moment parameters ν as

$$\hat{\nu} = \arg\max_{\nu=\nu_s} \; H(\nu). \tag{2.42}$$

We can write the corresponding optimization problem as

$$\begin{aligned} &\text{maximize} \quad H(\nu) \\ &\text{subject to } \nu = \nu_s \end{aligned} \tag{2.43}$$

where $\nu \in \mathbb{R}^n$ is the optimization variable.

2.4.5 Lagrangian Duality

Here we will show the Lagrangian duality relation between the maximum likelihood estimation and the maximum entropy estimation problems for exponential family distributions. The Lagrangian duality relation is as follows: Minimization of the negative log-likelihood in the natural parameters and the maximization of the entropy in the moment parameters subject to equality constraints on the moment parameters are Lagrange dual optimization problems.

Proposition 6. *The maximum likelihood estimation problem in* (2.41) *and the maximum entropy estimation problem in* (2.43) *are Lagrange dual optimization problems.*

Proof. The Lagrange dual function of the problem in (2.41) is the constant p^* where

$$p^* = \inf_{\theta \in \mathbf{dom}\,\Phi} \Phi(\theta) - \theta^T \nu_s. \tag{2.44}$$

Now let us reformulate the problem in (2.41) as

$$\begin{aligned} &\text{minimize} \quad \Phi(\theta) - \bar{\theta}^T \nu_s \\ &\text{subject to} \;\; \bar{\theta} = \theta \end{aligned} \tag{2.45}$$

Here we introduced new variables $\bar{\theta} \in \mathbb{R}^n$, as well as new equality constraints $\bar{\theta} = \theta$. The problems in (2.41) and (2.45) are clearly equivalent. The Lagrangian $L : \mathbb{R}^n \times \mathbb{R}^n \times \mathbb{R}^n \to \mathbb{R}$ of the reformulated problem in (2.45) is

$$L(\theta, \bar{\theta}, \nu) = \Phi(\theta) - \bar{\theta}^T \nu_s + \nu^T (\bar{\theta} - \theta) \tag{2.46}$$

where the variables $\nu \in \mathbb{R}^n$ are the Lagrange multipliers.

To find the Lagrange dual function we minimize L over θ and $\bar{\theta}$. The Lagrange dual function $g : \mathbb{R}^n \to \mathbb{R}$ is

$$\begin{aligned} g(\nu) &= \inf_{\theta \in \mathbf{dom}\,\Phi,\, \bar{\theta} \in \mathbb{R}^n} L(\theta, \bar{\theta}, \nu) \\ &= \inf_{\theta \in \mathbf{dom}\,\Phi,\, \bar{\theta} \in \mathbb{R}^n} \Big(\Phi(\theta) - \bar{\theta}^T \nu_s + \nu^T (\bar{\theta} - \theta) \Big). \end{aligned} \tag{2.47}$$

The Lagrangian L is seperable in θ and $\bar{\theta}$, therefore, it can be infimized separately over θ and $\bar{\theta}$.

$$\begin{aligned} g(\nu) &= \inf_{\theta \in \mathbf{dom}\,\Phi} \Big(\Phi(\theta) - \nu^T \theta \Big) + \inf_{\bar{\theta} \in \mathbb{R}^n} \Big(- \bar{\theta}^T \nu_s + \nu^T \bar{\theta} \Big) \\ &= \inf_{\theta \in \mathbf{dom}\,\Phi} \Big(\Phi(\theta) - \theta^T \nu \Big) + \inf_{\bar{\theta} \in \mathbb{R}^n} \Big(- \bar{\theta}^T \nu_s + \bar{\theta}^T \nu \Big) \\ &= \inf_{\theta \in \mathbf{dom}\,\Phi} \Big(\Phi(\theta) - \theta^T \nu \Big) + \inf_{\bar{\theta} \in \mathbb{R}^n} \Big(\bar{\theta}^T (\nu - \nu_s) \Big). \end{aligned} \tag{2.48}$$

Using the Fenchel duality relation (2.35) between the log partition function $\Phi(\theta)$ and the entropy function $H(\nu)$, i.e.,

$$H(\nu) = \inf_{\theta \in \mathbf{dom}\,\Phi} \Phi(\theta) - \theta^T \nu \tag{2.49}$$

we have

$$\begin{aligned} g(\nu) &= \inf_{\theta \in \mathbf{dom}\,\Phi} \Big(\Phi(\theta) - \theta^T \nu \Big) + \inf_{\bar{\theta} \in \mathbb{R}^n} \Big(\bar{\theta}^T (\nu - \nu_s) \Big) \\ &= H(\nu) + \inf_{\bar{\theta} \in \mathbb{R}^n} \bar{\theta}^T (\nu - \nu_s). \end{aligned} \tag{2.50}$$

Notice that Lagrangian is linear in $\bar{\theta}$ so $g(\nu) = -\infty$ unless $\nu - \nu_s = 0$. So the dual function $g(\nu)$ is

$$g(\nu) = \begin{cases} H(\nu), & \text{if } \nu = \nu_s \\ -\infty, & \text{otherwise} \end{cases} \tag{2.51}$$

Thus, the Lagrange dual of the reformulated problem can be expressed as

$$\begin{aligned} &\text{maximize} \quad H(\nu) \\ &\text{subject to } \nu = \nu_s \end{aligned} \tag{2.52}$$

which is same as the maximum entropy problem in (2.43).

Now we will show that the Lagrangian dual of the maximum entropy (ME) problem in (2.43) corresponds to the maximum likelihood (ML) problem in (2.41). Since both ME and ML problems are convex, we can take the dual of the ME problem and get the ML problem. The Lagrangian $L : \mathbb{R}^n \times \mathbb{R}^n \to \mathbb{R}$ of the problem in (2.52) is

$$L(\nu, \theta) = H(\nu) + \theta^T(\nu - \nu_s) \tag{2.53}$$

where the variables $\theta \in \mathbb{R}^n$ are the Lagrange multipliers.

To find the Lagrange dual function we maximize L over ν. The Lagrange dual function $g : \mathbb{R}^n \to \mathbb{R}$ is

$$\begin{aligned} g(\theta) &= \sup_{\nu \in \mathbf{dom}\, H} L(\nu, \theta) \\ g(\theta) &= \sup_{\nu \in \mathbf{dom}\, H} \Big(H(\nu) + \theta^T(\nu - \nu_s) \Big) \\ &= \sup_{\nu \in \mathbf{dom}\, H} \Big(H(\nu) + \nu^T\theta \Big) - \theta^T\nu_s \\ &= \Phi(\theta) - \theta^T\nu_s. \end{aligned} \tag{2.54}$$

So the dual of the maximum entropy problem is the maximum likelihood problem

$$\text{minimize } \Phi(\theta) - \theta^T\nu_s. \tag{2.55}$$

where $\theta \in \mathbb{R}^n$ is the optimization variable. □

Corollary 5. *We can find the ML estimates of the natural parameters and the ME estimates of the moment parameters via solving convex optimization problems.*

2.4.6 Multinomial and Gaussian Distributions

In this Section we will introduce the multinomial and Gaussian distributions (which are used as building blocks of Gaussian mixture models) within the exponential family formulation. First, we will introduce the commonly used parameterizations used for these distributions. Then, we will provide their representations in the exponential family form and describe the relations between the natural and the moment parameterizations induced by the exponential family representation and the commonly used parameterizations.

2.4.6.1 Multinomial Distribution

The multinomial distribution is one of the most widely used discrete multidimensional distributions in machine learning and statistics [42], [154], [151], [150], [3]. In this thesis we will only use one dimensional multinomial distributions; hence, to avoid clutter here we restrict our treatment to the one dimensional case. We consider a discrete (multinomial) random variable y taking values in the sample space $\Omega_y = \{1, \ldots, K\}$ with source parameters α corresponding to a set of probabilities $\{\alpha_1, \ldots, \alpha_K\}$. We can write the probability density function $p(y|\alpha)$ as

$$p(y|\alpha) = \prod_{k=1}^{K} \alpha_k^{\delta(y=k)} \tag{2.56}$$

where $\delta(y = k)$ denotes the Kronecker delta function which is equal to 1 when y takes the value k and 0 otherwise.

Notice that to be a valid probability density function, probabilities α should sum to 1, i.e., $\sum_{k=1}^{K} \alpha_k = 1$. However this leads to an over complete representation. To overcome this problem, we parametrize the probability density function

$p(y|\alpha)$ using the first $K-1$ components of α.

$$\begin{aligned} p(y|\alpha) &= \prod_{k=1}^{K} \alpha_k^{\delta(y=k)} \\ &= \prod_{k=1}^{K-1} \alpha_k^{\delta(y=k)} \alpha_K^{\delta(y=K)} \\ &= \prod_{k=1}^{K-1} \alpha_k^{\delta(y=k)} \big(1 - \sum_{i=1}^{K-1} \alpha_i\big)^{\left(1-\sum_{i=1}^{K-1} \delta(y=i)\right)} \\ &= p(y|\hat{\alpha}) \end{aligned} \tag{2.57}$$

Where $p(y|\hat{\alpha})$ uses only the first $K-1$ probabilities as parameters $\hat{\alpha}$ where $\hat{\alpha} = \{\alpha_1, \ldots, \alpha_{K-1}\}$.

We would like to represent the multinomial distribution $p(y|\hat{\alpha})$ in in the following exponential family form

$$p(y|\theta_y) = \exp\left(\theta_y^T \phi_y(y) - \Phi(\theta_y)\right) \tag{2.58}$$

where $\theta_y \in \mathbb{R}^{K-1}$ denotes the natural parameters, $\phi_y : \Omega_y \rightarrow \mathbb{R}^{K-1}$ denotes the sufficient statistic function and $\Phi(\theta_y)$ denotes the log partition function. To obtain exponential family form $p(y|\theta_y)$, we rewrite $p(y|\hat{\alpha})$ as

$$\begin{aligned} p(y|\hat{\alpha}) &= \exp\log \prod_{k=1}^{K-1} \alpha_k^{\delta(y=k)} \big(1 - \sum_{i=1}^{K-1} \alpha_i\big)^{\left((1-\sum_{i=1}^{K-1} \delta(y=i))\right)} \\ &= \exp\Big(\sum_{k=1}^{K-1} \delta(y=k) \log \alpha_k + \big(1 - \sum_{i=1}^{K-1} \delta(y=i)\big) \log(1 - \sum_{i=1}^{K-1} \alpha_i)\Big) \\ &= \exp\Big(\sum_{k=1}^{K-1} \delta(y=k) \log \alpha_k + \log(1 - \sum_{i=1}^{K-1} \alpha_i) \\ &\qquad - \sum_{k=1}^{K-1} \delta(y=k) \log(1 - \sum_{i=1}^{K-1} \alpha_i)\Big) \\ &= \exp\Big(\sum_{k=1}^{K-1} \delta(y=k) \log \frac{\alpha_k}{(1 - \sum_{i=1}^{K-1} \alpha_i)} + \log(1 - \sum_{i=1}^{K-1} \alpha_i)\Big) \\ &= \exp\Big(\sum_{k=1}^{K-1} \log \frac{\alpha_k}{(1 - \sum_{i=1}^{K-1} \alpha_i)} \delta(y=k) - \log(1 - \sum_{i=1}^{K-1} \alpha_i)^{-1}\Big) \end{aligned} \tag{2.59}$$

We select the sufficient statistic function $\phi_y : \Omega_y \to \mathbb{R}^{K-1}$ as

$$\phi_y(y) = (\delta(y = 1), \ldots, \delta(y = K - 1))^T$$

which leads to the natural parameters $\theta_y \in \mathbb{R}^{K-1}$ as

$$\theta_y = (\log \frac{\alpha_1}{(1 - \sum_{i=1}^{K-1} \alpha_i)}, \ldots, \log \frac{\alpha_{K-1}}{(1 - \sum_{i=1}^{K-1} \alpha_i)})^T$$

and the log partition function $\Phi(\theta_y)$

$$\Phi(\theta_y) = \log(1 + \sum_{k=1}^{K-1} \exp \theta_{y=k})$$

where we used the following relation

$$\begin{aligned}
\log(1 - \sum_{k=1}^{K-1} \alpha_k)^{-1} &= \log(\frac{1 - \sum_{j=1}^{K-1} \alpha_j + \sum_{k=1}^{K-1} \alpha_k}{1 - \sum_{k=1}^{K-1} \alpha_k}) \\
&= \log(1 + \sum_{k=1}^{K-1} \frac{\alpha_k}{1 - \sum_{j=1}^{K-1} \alpha_j}) \\
&= \log(1 + \sum_{k=1}^{K-1} \exp \log \frac{\alpha_k}{1 - \sum_{j=1}^{K-1} \alpha_j}) \\
&= \log(1 + \sum_{k=1}^{K-1} \exp \theta_{y=k}) \qquad (2.60)
\end{aligned}$$

Then we can rewrite Eq. (2.59) in terms of θ_y as

$$p(y|\theta_y) = \exp\Big(\sum_{k=1}^{K-1} \theta_{y=k} \delta(y = k) - \log(1 + \sum_{k=1}^{K-1} \exp \theta_{y=k}) \Big) \qquad (2.61)$$

To summarize, we can represent the multinomial distribution in exponential family form

$$p(y|\theta_y) = \exp\left(\theta_y^T \phi_y(y) - \Phi(\theta_y)\right)$$

with the natural parameters $\theta_y \in \mathbb{R}^{K-1}$, sufficient statistic function $\phi_y : \Omega_y \to \mathbb{R}^{K-1}$ and the log partition function $\Phi(\theta_y)$ where

$$\theta_y = (\theta_{y=1}, \ldots, \theta_{y=K-1})^T \qquad (2.62)$$

$$\phi_y(y) = (\delta(y = 1), \ldots, \delta(y = K - 1))^T \tag{2.63}$$

$$\Phi(\theta_y) = \log(1 + \sum_{k=1}^{K-1} \exp \theta_{y=k}) \tag{2.64}$$

Now, we will derive the moment parameters induced by the sufficient statistic function (2.63) and the entropy function induced by the log partition function (2.64).

As shown in Proposition 1, we have seen that it is possible to obtain the moment parameters ν_y as a function of the natural parameters θ_y, and the gradient of the log partition function (2.64) provides a mapping $\nabla_{\theta_y}\Phi : \theta_y \rightarrow \nu_y$. We can obtain the moment parameters ν_y by taking the gradient of the log partition function (2.64) with respect to θ_y. For the partial derivatives $\frac{\partial \Phi(\theta_y)}{\partial \theta_{y=k}}$ we have

$$\frac{\partial \Phi(\theta_y)}{\partial \theta_{y=k}} = \frac{\exp \theta_{y=k}}{1 + \sum_{i=1}^{K-1} \exp \theta_{y=k}} \tag{2.65}$$

$$= \nu_{y=k} \tag{2.66}$$

By noticing

$$\log \nu_{y=k} = \theta_{y=k} + \log \left(1 + \sum_{i=1}^{K-1} \exp \theta_{y=i}\right)^{-1}$$

$$\log \nu_{y=k} - \log \left(1 + \sum_{i=1}^{K-1} \exp \theta_{y=i}\right)^{-1} = \theta_{y=k} \tag{2.67}$$

and using the relation we found in (2.60) where

$$\log \left(1 + \sum_{i=1}^{K-1} \exp \theta_{y=i}\right) = \log(1 - \sum_{k=1}^{K-1} \nu_{y=k})^{-1}$$

we have

$$\theta_{y=k} = \log \frac{\nu_{y=k}}{1 - \sum_{i=1}^{K-1} \nu_{y=i}} \tag{2.68}$$

Now we will derive the corresponding entropy function $H(\nu_y)$ for the multinomial distribution. As shown in the Fenchel duality relations between the log partition

function and the entropy function in (2.32), (2.35), the entropy function can be written as

$$
\begin{aligned}
H(\nu_y) &= \inf_{\theta_y} \Phi(\theta_y) - \theta_y^T \nu_y \\
&= [\Phi(\theta_y) - \theta_y^T \nu_y]_{\nu_y = \nabla_{\theta_y}\Phi(\theta_y)} \\
&= [\log\big(1 + \sum_{i=1}^{K-1} \exp\theta_{y=i}\big) - \sum_{k=1}^{K-1} \theta_{y=k}\nu_{y=k}]_{\theta_{y=k} = \log\frac{\nu_{y=k}}{1-\sum_{i=1}^{K-1}\nu_{y=i}}} \\
&= \log\big(1 + \sum_{k=1}^{K-1} \exp\log\frac{\nu_{y=k}}{1-\sum_{i=1}^{K-1}\nu_{y=i}}\big) \\
&\quad - \sum_{k=1}^{K-1} \log(\frac{\nu_{y=k}}{1-\sum_{i=1}^{K-1}\nu_{y=i}})\nu_{y=k} \\
&= \log\big(1 + \sum_{i=1}^{K-1} \frac{\nu_{y=k}}{1-\sum_{i=1}^{K-1}\nu_{y=i}}\big) \\
&\quad - \sum_{k=1}^{K-1} \log(\frac{\nu_{y=k}}{1-\sum_{i=1}^{K-1}\nu_{y=i}})\nu_{y=k} \\
&= \log\big(\frac{1-\sum_{i=1}^{K-1}\nu_{y=i} + \sum_{i=1}^{K-1}\nu_{y=i}}{1-\sum_{i=1}^{K-1}\nu_{y=i}}\big) \\
&\quad - \sum_{k=1}^{K-1} \log(\frac{\nu_{y=k}}{1-\sum_{i=1}^{K-1}\nu_{y=i}})\nu_{y=k} \\
&= \log(1 - \sum_{i=1}^{K-1}\nu_{y=i})^{-1} - \sum_{k=1}^{K-1}\nu_{y=k}\log\nu_{y=k} \\
&\quad - \sum_{k=1}^{K-1}\nu_{y=k}\log(1-\sum_{i=1}^{K-1}\nu_{y=i})^{-1} \\
&= -\sum_{k=1}^{K-1}\nu_{y=k}\log\nu_{y=k} + (1-\sum_{k=1}^{K-1}\nu_{y=k})\log(1-\sum_{i=1}^{K-1}\nu_{y=i})^{-1} \\
&= -\sum_{k=1}^{K-1}\nu_{y=k}\log\nu_{y=k} - (1-\sum_{k=1}^{K-1}\nu_{y=k})\log(1-\sum_{k=1}^{K-1}\nu_{y=k}) \qquad (2.69)
\end{aligned}
$$

2.4.6.2 Gaussian Distribution

The most common way of parameterizing the Gaussian distribution is in terms of the mean vector $\mu = E[\mathbf{x}]$ and covariance matrix $\Sigma = E[(\mathbf{x} - \mu)(\mathbf{x} - \mu)^T]$. In terms of these parameters, the Gaussian distribution is defined as follows [42], [3]

Definition 19. *A random vector* $\mathbf{x}$ *with the sample space* $\Omega_{\mathbf{x}} = \mathbb{R}^d$ *has a Gaussian distribution with mean* $\mu \in \mathbb{R}^d$ *and covariance matrix* $\Sigma \in \mathcal{S}_{++}^d$ *if its pdf is given by*

$$p(\mathbf{x}|\mu, \Sigma) = N(\mathbf{x}|\mu, \Sigma) \tag{2.70}$$

$$= \frac{1}{(2\pi)^{d/2}|\Sigma|^{1/2}} \exp\big(-\frac{1}{2}(\mathbf{x} - \mu)^T\Sigma^{-1}(\mathbf{x} - \mu)\big) \tag{2.71}$$

The parameterization of Gaussian distribution in terms of the mean vector μ *and the covariance matrix* Σ *in* (2.71) *is referred to as the source form and the parameters* μ, Σ *are called the source parameters.*

An alternative popular parameterization of the Gaussian distribution is provided by the information form [42], [3]

Definition 20. *A random vector* $\mathbf{x}$ *with the sample space* $\Omega_{\mathbf{x}} = \mathbb{R}^d$ *has a Gaussian distribution with information vector* $m \in \mathbb{R}^d$ *and information matrix* $S \in \mathcal{S}_{++}^d$ *if its pdf is given by*

$$p(\mathbf{x}|m, S) = N(\mathbf{x}|m, S) \tag{2.72}$$

$$= \exp\big(m^T\mathbf{x} + tr(-\frac{1}{2}S\mathbf{x}\mathbf{x}^T) + \frac{1}{2}\log|S| - \frac{1}{2}m^TS^{-1}m - \frac{d}{2}\log 2\pi\big) \tag{2.73}$$

The parameterization of Gaussian distribution in terms of the information vector m *and the information matrix* S *in* (2.73) *is referred to as the information form and the parameters* m, S *are called the information parameters.*

Gaussian distribution belongs to exponential family and the source parameters are closely related to the moment parameters while information parameters are closely related to the natural parameters.

We can represent the Gaussian distribution in information form $N(\mathbf{x}|m, S)$ (2.73) in exponential family form as

$$\begin{aligned} p(\mathbf{x}|\theta_{\mathbf{x}}) &= N(\mathbf{x}|m, S) \\ &= \exp\left(m^T\mathbf{x} + tr(-\frac{1}{2}S\mathbf{x}\mathbf{x}^T) + \frac{1}{2}\log|S| - \frac{1}{2}m^TS^{-1}m - \frac{d}{2}\log 2\pi\right) \\ &= \exp\Big(m^T\mathbf{x} + tr(-\frac{1}{2}S\mathbf{x}\mathbf{x}^T) - \big(-\frac{1}{2}\log|S| + \frac{1}{2}m^TS^{-1}m + \frac{d}{2}\log 2\pi\big)\Big) \\ &= \exp(\theta_{\mathbf{x}}^T\phi_{\mathbf{x}}(\mathbf{x}) - \Phi(\theta_{\mathbf{x}})) \end{aligned} \tag{2.74}$$

where sufficient statistic function $\Phi_{\mathbf{x}} : \mathbb{R}^d \to \mathbb{R}^d \times \mathcal{K}_+^d$, $\mathcal{K}_+^d = \{vec(R) \in \mathbb{R}^{d(d+1)/2} \mid R \in \mathcal{S}_+^d\}$, is

$$\phi_{\mathbf{x}}(\mathbf{x}) = (\mathbf{x}^T, vec(\mathbf{x}\mathbf{x}^T)^T)^T \tag{2.75}$$

which induces the natural parameters $\theta_{\mathbf{x}} \in \mathbb{R}^d \times \mathcal{K}_-^d$, $\mathcal{K}_-^d = \{vec(-\frac{1}{2}S) \in \mathbb{R}^{d(d+1)/2} \mid S \in \mathcal{S}_+^d\}$, as

$$\theta_{\mathbf{x}} = (m^T, vec(-\frac{1}{2}S)^T)^T \tag{2.76}$$

and the log partition function $\Phi : \mathbb{R}^d \times \mathcal{K}_{--}^d \to \mathbb{R}$, $\mathcal{K}_{--}^d = \{vec(-\frac{1}{2}S) \mid S \in \mathcal{S}_{++}^d\}$, is

$$\Phi(\theta_{\mathbf{x}}) = -\frac{1}{2}\log|S| + \frac{1}{2}m^TS^{-1}m + \frac{d}{2}\log 2\pi \tag{2.77}$$

which leads to the moment parameters $\nu \in \mathbb{R}^d \times \mathcal{K}_+^d$, as

$$\nu_{\mathbf{x}} = (\mu^T, vec(\Sigma + \mu\mu^T)^T)^T \tag{2.78}$$

where the moment parameters correspond to the expected values of the sufficient statistic function, i.e., $\nu_{\mathbf{x}} = E_{p(\mathbf{x}|\theta)}[\phi_{\mathbf{x}}(\mathbf{x})]$.

We can see the relation between the information parameters m, S and the

source parameters μ, Σ by noticing

$$\begin{aligned} N(\mathbf{x}|\mu,\Sigma) &= \frac{1}{(2\pi)^{d/2}|\Sigma|^{1/2}} \exp\Big(-\frac{1}{2}(\mathbf{x}-\mu)^T\Sigma^{-1}(\mathbf{x}-\mu)\Big) \\ &= \exp\Big(-\frac{1}{2}\mathbf{x}^T\Sigma^{-1}\mathbf{x} + \mathbf{x}^T\Sigma^{-1}\mu - \frac{1}{2}\mu^T\Sigma^{-1}\mu + \frac{1}{2}\log|\Sigma^{-1}| - \frac{d}{2}\log 2\pi\Big) \\ &= \exp\Big(tr(-\frac{1}{2}\Sigma^{-1}\mathbf{x}\mathbf{x}^T) + (\Sigma^{-1}\mu)^T\mathbf{x} - \frac{1}{2}(\Sigma^{-1}\mu)^T\Sigma(\Sigma^{-1}\mu) \\ &\qquad + \frac{1}{2}\log|\Sigma^{-1}| - \frac{d}{2}\log 2\pi\Big) \\ &= \exp\Big((\Sigma^{-1}\mu)^T\mathbf{x} + tr(-\frac{1}{2}\Sigma^{-1}\mathbf{x}\mathbf{x}^T) + \frac{1}{2}\log|\Sigma^{-1}| \\ &\qquad - \frac{1}{2}(\Sigma^{-1}\mu)^T\Sigma(\Sigma^{-1}\mu) - \frac{d}{2}\log 2\pi\Big) \\ &= \exp\Big(m^T\mathbf{x} + tr(-\frac{1}{2}S\mathbf{x}\mathbf{x}^T) + \frac{1}{2}\log|S| - \frac{1}{2}m^TS^{-1}m - \frac{d}{2}\log 2\pi\Big) \end{aligned} \tag{2.79}$$

where we have $m = \Sigma^{-1}\mu$, $S = \Sigma^{-1}$ and $\mu = S^{-1}m$, $\Sigma = S^{-1}$.

We can show the same parameter relations using the moment generating property of the log partition function, i.e., $\nabla_{\theta_\mathbf{x}}\Phi(\theta_\mathbf{x}) = \nu_\mathbf{x}$. Notice that

$$\nabla_{\theta_\mathbf{x}}\Phi(\theta_\mathbf{x}) = \begin{bmatrix} \nabla_m\Phi(m,S) \\ \nabla_{vec(-\frac{1}{2}S)}\Phi(m,S) \end{bmatrix} \tag{2.80}$$

We have

$$\begin{aligned} \nabla_m\Phi(m,S) &= S^{-1}m \\ &= \mu \end{aligned} \tag{2.81}$$

and

$$\begin{aligned} \nabla_{-\frac{1}{2}S}\Phi(m,S) &= S^{-1} + S^{-1}mm^TS^{-1} \\ &= \Sigma + \mu\mu^T \end{aligned} \tag{2.82}$$

We can find the entropy function $H(\nu_\mathbf{x}) = \frac{1}{2}\log|\Sigma| + \frac{d}{2}\log(2\pi e)$ using the Fenchel

duality relation. To avoid cluttered derivation, we will use the information parameters m, S and the source parameters μ, Σ.

$$
\begin{aligned}
H(\nu_{\mathbf{x}}) &= \inf_{\theta_{\mathbf{x}} \in \mathbf{dom}\,\Phi} \Phi(\theta_{\mathbf{x}}) - \theta_{\mathbf{x}}^T \nu_{\mathbf{x}} \\
&= \Big[\Phi(\theta_{\mathbf{x}}) - \theta_{\mathbf{x}}^T \nu_{\mathbf{x}}\Big]_{\nabla_{\theta_{\mathbf{x}}} \Phi(\theta_{\mathbf{x}}) = \nu_{\mathbf{x}}} \\
&= \Big[-\frac{1}{2}\log|S| + \frac{1}{2} m^T S^{-1} m + \frac{d}{2}\log 2\pi - m^T \mu \\
&\qquad - tr(-\frac{1}{2} S(\Sigma + \mu\mu^T))\Big]_{m=\Sigma^{-1}\mu, S=\Sigma^{-1}} \\
&= -\frac{1}{2}\log|\Sigma^{-1}| + \frac{1}{2}(\Sigma^{-1}\mu)^T \Sigma (\Sigma^{-1}\mu) + \frac{d}{2}\log 2\pi - (\Sigma^{-1}\mu)^T \mu \\
&\quad - tr(-\frac{1}{2}\Sigma^{-1}(\Sigma + \mu\mu^T)) \\
&= \frac{1}{2}\log|\Sigma| + \frac{1}{2}\mu^T\Sigma^{-1}\mu + \frac{d}{2}\log 2\pi - \mu^T\Sigma^{-1}\mu + \frac{1}{2} tr(\Sigma^{-1}\Sigma) \\
&\quad + \frac{1}{2} tr(\Sigma^{-1}\mu\mu^T) \\
&= \frac{1}{2}\log|\Sigma| + \frac{1}{2}\mu^T\Sigma^{-1}\mu^T + \frac{1}{2}\mu^T\Sigma^{-1}\mu^T - \mu^T\Sigma^{-1}\mu^T + \frac{d}{2}\log 2\pi \\
&\quad + \frac{1}{2} tr(\Sigma^{-1}\Sigma) \\
&= \frac{1}{2}\log|\Sigma| + \frac{d}{2}\log 2\pi + \frac{d}{2} \\
&= \frac{1}{2}\log|\Sigma| + \frac{d}{2}\log 2\pi + \frac{d}{2}\log(e) \\
&= \frac{1}{2}\log|\Sigma| + \frac{d}{2}\log(2\pi e)
\end{aligned} \tag{2.83}
$$

Chapter 3

Constrained Gaussian Mixture Models

3.1 Introduction

In this Chapter we consider the constrained Gaussian mixture models which serves as the fundamental modeling framework for the robust density estimation and the compound object detection problems described in the following Chapters. We consider two different parameterizations which we refer to as the information parameterization and the source parameterization. In constrained Gaussian mixture models, our objective is to obtain the maximum likelihood estimates of Gaussian mixture model parameters satisfying convex inequality and affine equality constraints. To estimate the parameters we use the expectation maximization algorithm which consists of two steps called the E-step and the M-step. In the E-step, we compute the posterior distributions of the hidden variables given the observed variables while in the M-step we optimize the expected joint log-likelihood of the observed and the hidden variables over the model parameters. As our first contribution, we show that the M-step for the Gaussian mixture models correspond to a convex optimization problem in the information parameters and we can handle the convex constraints on the information parameters

by solving a constrained convex optimization problem. We refer to this problem as the primal problem for the M-step. As our second contribution, we form the Lagrangian dual problem of the primal problem for the M-step and show that it corresponds to an equality constrained convex optimization problem in the source parameters. As our third contribution, we provide an unconstrained version of the dual problem and show that the optimal parameter estimates are the same. Then we show that we can handle the convex constraints on the source parameters by solving the convex dual problem. The unifying idea in this Chapter is that we can handle convex constraints on the Gaussian mixture parameters by solving convex optimization problems for the M-step.

The organization of this Chapter is as follows. In Section 3.2 we derive a representation for the joint distribution of the Gaussian mixture models in exponential family form. In Section 3.3 we consider the maximum likelihood estimation problem for Gaussian mixture models. We introduce the expectation maximization (EM) algorithm, and show that the M-step corresponds to a convex optimization problem in the natural parameters. Then, we form the Lagrangian dual problem which corresponds to an equality constrained convex optimization problem in the moment parameters. Afterwards, we provide an unconstrained dual problem and show that the optimal parameter estimates are the same. Lastly, we express the primal convex optimization problem for the M-step in terms of the information parameters and the dual convex optimization problem for the M-step in terms of the source parameters. In Section 3.4 we summarize the constrained Gaussian mixture model framework and the EM algorithm used to estimate the parameters. Example constraints and the conclusions are given in Sections 3.5 and 3.6, respectively.

3.2 Gaussian Mixture Models

We consider the family $\mathcal{F}$ of distributions of Gaussian mixture models with K Gaussian components denoted by $p(\mathbf{x}, y|\theta) \in \mathcal{F}$ over d dimensional continuous

random vector $\mathbf{x} \in \mathbb{R}^d$ and a multinomial random variable $y \in \{1, \ldots, K\}$ parameterized with the natural parameters $\theta \in \mathcal{C}_\theta$, $\mathcal{C}_\theta = \mathbb{R}^{K-1} \times \otimes_{k=1}^{K} \mathbb{R}^d \times \mathcal{K}_-^d$ in exponential family form as

$$p(\mathbf{x}, y|\theta) = \exp(\theta^T \phi(\mathbf{x}, y) - \Phi(\theta, y)) \tag{3.1}$$

The marginal distribution of y denoted by $p(y|\theta_y)$ can be written in exponential family form as

$$p(y|\theta_y) = \exp\left(\theta_y^T \phi_y(y) - \Phi(\theta_y)\right) \tag{3.2}$$

where $\theta_y \in \mathbb{R}^{K-1}$ denotes the natural parameters, $\phi_y : \Omega_y \to \mathbb{R}^{K-1}$ denotes the sufficient statistic function and $\Phi(\theta_y)$ denotes the log partition function.

Conditioned on the value of the multinomial variable $y = k$, the conditional distribution $p(\mathbf{x}|y = k, \theta_{\mathbf{x}|y=k})$ of d dimensional random vector $\mathbf{x} \in R^d$ is a Gaussian with the natural parameters $\theta_{\mathbf{x}|y=k} \in \mathbb{R}^d \times \mathcal{K}_-^d$.

$$p(\mathbf{x}|y = k, \theta_{\mathbf{x}|y=k}) = p(\mathbf{x}|\theta_{\mathbf{x}|y=k}) \tag{3.3}$$

Gaussian distributions can be written in exponential family form as follows

$$p(\mathbf{x}|\theta_{\mathbf{x}|y=k}) = \exp\left(\theta_{\mathbf{x}|y=k}^T \phi_{\mathbf{x}}(\mathbf{x}) - \Phi(\theta_{\mathbf{x}|y=k})\right) \quad \text{for } k = 1, \ldots, K \tag{3.4}$$

where the natural parameters of the kth Gaussian is denoted with $\theta_{\mathbf{x}|y=k} \in \mathbb{R}^d \times \mathcal{K}_-^d$, sufficient statistic function is denoted with $\phi_{\mathbf{x}} : \mathbb{R}^d \to \mathbb{R}^d \times \mathcal{K}_+^d$ and the log partition function is denoted by $\Phi(\theta_{\mathbf{x}|y=k})$.

The joint distribution $p(\mathbf{x}, y|\theta)$ is given by

$$p(\mathbf{x}, y|\theta) = p(y|\theta_y) p(\mathbf{x}|y, \theta_{\mathbf{x}|y}) \tag{3.5}$$

To form the joint distribution, we write the conditional distribution $p(\mathbf{x}|y, \theta_{\mathbf{x}|y})$

of Gaussian $\mathbf{x}$ given multinomial y as follows

$$\begin{aligned} p(\mathbf{x}|y,\theta_{\mathbf{x}|y}) &= \prod_{k=1}^{K-1} p(\mathbf{x}|\theta_{\mathbf{x}|y=k})^{\delta(y=k)} p(\mathbf{x}|\theta_{\mathbf{x}|y=K})^{\left(1-\sum_{i=1}^{K-1}\delta(y=i)\right)} \\ &= \prod_{k=1}^{K} p(\mathbf{x}|\theta_{\mathbf{x}|y=k})^{\delta_{yk}} \\ &= \exp\log\prod_{k=1}^{K} p(\mathbf{x}|\theta_{\mathbf{x}|y=k})^{\delta_{yk}} \\ &= \exp\Big(\sum_{k=1}^{K}\delta_{yk}\log p(\mathbf{x}|\theta_{\mathbf{x}|y=k})\Big) \end{aligned} \tag{3.6}$$

where to avoid cluttered notation, we defined the constrained vector δ_y of delta functions as follows

$$\delta_y = (\delta(y=0),\ldots,\delta(y=K-1),(1-\sum_{i=1}^{K-1}\delta(y=i)))^T \tag{3.7}$$

Now to get a compact conditional exponential family representation we substitute exponential family represention of the k'th Gaussian in (3.4) for $p(\mathbf{x}|\theta_{\mathbf{x}|y=k})$. Then we have

$$\begin{aligned} p(\mathbf{x}|y,\theta_{\mathbf{x}|y}) &= \exp\Big(\sum_{k=1}^{K}\delta_{yk}\big(\theta_{\mathbf{x}|y=k}^T\phi_{\mathbf{x}}(\mathbf{x})-\Phi(\theta_{\mathbf{x}|y=k})\big)\Big) \\ &= \exp\Big(\sum_{k=1}^{K}\theta_{\mathbf{x}|y=k}^T\big(\delta_{yk}\phi_{\mathbf{x}}(\mathbf{x})\big)-\sum_{k=1}^{K}\delta_{yk}\Phi(\theta_{\mathbf{x}|y=k})\Big) \\ &= \exp\Big(\sum_{k=1}^{K}\theta_{\mathbf{x}|y=k}^T\phi_{\mathbf{x}|y=k}(\mathbf{x},y)-\sum_{k=1}^{K}\delta_{yk}\Phi(\theta_{\mathbf{x}|y=k})\Big) \\ &= \exp\big(\theta_{\mathbf{x}|y}^T\phi_{\mathbf{x}|y}(\mathbf{x},y)-\sum_{k=1}^{K}\delta_{yk}\Phi(\theta_{\mathbf{x}|y=k})\big) \\ &= \exp\big(\theta_{\mathbf{x}|y}^T\phi_{\mathbf{x}|y}(\mathbf{x},y)-\Phi(\theta_{\mathbf{x}|y},y)\big) \end{aligned} \tag{3.8}$$

Hence, we can write the conditional distribution $p(\mathbf{x}|y,\theta_{\mathbf{x}|y})$ of Gaussian $\mathbf{x}$ given multinomial y in exponential family form as

$$p(\mathbf{x}|y,\theta_{\mathbf{x}|y}) = \exp\big(\theta_{\mathbf{x}|y}^T\phi_{\mathbf{x}|y}(\mathbf{x},y)-\Phi(\theta_{\mathbf{x}|y},y)\big) \tag{3.9}$$

where the natural parameters $\theta_{\mathbf{x}|y}$ are

$$\theta_{\mathbf{x}|y} = (\theta_{\mathbf{x}|y=1}^T, \ldots, \theta_{\mathbf{x}|y=K}^T)^T \tag{3.10}$$

sufficient statistic function $\phi_{\mathbf{x}|y}(\mathbf{x}, y)$ is

$$\begin{aligned}\phi_{\mathbf{x}|y}(\mathbf{x}, y) &= \left((\delta_{y1}\phi_{\mathbf{x}}(\mathbf{x}))^T, \ldots, (\delta_{yK}\phi_{\mathbf{x}}(\mathbf{x}))^T\right)^T \\ &= \left(\phi_{\mathbf{x}|y=1}(\mathbf{x}, y), \ldots, \phi_{\mathbf{x}|y=K}(\mathbf{x}, y)\right)^T \end{aligned} \tag{3.11}$$

and the log partition function $\Phi(\theta_{\mathbf{x}|y}, y)$ is

$$\Phi(\theta_{\mathbf{x}|y}, y) = \sum_{k=1}^{K} \delta_{yk}\Phi(\theta_{\mathbf{x}|y=k}) \tag{3.12}$$

Given the multinomial and conditional Gaussian distributions in exponential family form, $p(y|\theta_y)$ and $p(\mathbf{x}|y, \theta_{\mathbf{x}|y})$, we can write the joint distribution $p(\mathbf{x}, y|\theta)$ in exponential family form as follows

$$\begin{aligned} p(\mathbf{x}, y|\theta) &= p(y|\theta_y)p(\mathbf{x}|y, \theta_{\mathbf{x}|y}) \\ &= \exp(\theta_y^T\phi_y(y) - \Phi(\theta_y)) \exp\left(\theta_{\mathbf{x}|y}^T\phi_{\mathbf{x}|y}(\mathbf{x}, y) - \Phi(\theta_{\mathbf{x}|y}, y)\right) \\ &= \exp\left(\theta_y^T\phi_y(y) + \theta_{\mathbf{x}|y}^T\phi_{\mathbf{x}|y}(\mathbf{x}, y) - \Phi(\theta_y) - \Phi(\theta_{\mathbf{x}|y}, y)\right) \\ &= \exp\left(\theta_y^T\phi_y(y) + \theta_{\mathbf{x}|y}^T\phi_{\mathbf{x}|y}(\mathbf{x}, y) - (\Phi(\theta_y) + \Phi(\theta_{\mathbf{x}|y}, y))\right) \\ &= \exp(\theta^T\phi(\mathbf{x}, y) - \Phi(\theta, y)) \end{aligned} \tag{3.13}$$

where for the natural parameters we have $\theta = (\theta_y^T, \theta_{\mathbf{x}|y}^T)^T$, for the sufficient statistic function we have $\phi(\mathbf{x}, y) = (\phi_y(y)^T, \phi_{\mathbf{x}|y}(\mathbf{x}, y)^T)^T$ and $\Phi(\theta, y) = \Phi(\theta_y) + \Phi(\theta_{\mathbf{x}|y}, y)$ is the log partition function.

3.3 Maximum Likelihood Estimation

In density estimation problems with Gaussian mixture models, we are given a data set $\mathcal{X} = \{\mathbf{x}_1, \ldots, \mathbf{x}_N\}$ of N data points and the corresponding multinomial variables $\mathcal{Y} = \{y_1, \ldots, y_N\}$ are treated as hidden variables. Given a data set $\mathcal{X} = \{\mathbf{x}_1, \ldots, \mathbf{x}_N\}$ of N independent and identically distributed (i.i.d.) random vectors corresponding to random samples from the marginal distribution

$p(\mathbf{x}|\theta) = \sum_{k=1}^{K} p(\mathbf{x}, y = k|\theta)$, our objective is to find the maximum likelihood (ML) estimate $\hat{\theta} \in \mathcal{C}_\theta$ of the natural parameters $\theta \in \mathcal{C}_\theta$. The ML estimation problem can be written in minimization form in 2.20 as

$$\hat{\theta} = \arg\min_{\theta} \; -\frac{1}{N}\sum_{j=1}^{N} \ell(\theta|\mathbf{x}_j) \tag{3.14}$$

3.3.1 Expectation Maximization Algorithm

The expectation maximization algorithm is a very general and popular algorithm used for doing maximum likelihood estimation of the parameters in models with hidden variables. The fundamental idea behind the expectation maximization algorithm is to use an upper bound function $F(\mathcal{Q}, \theta)$ on the negative log likelihoods, $-\ell(\theta|\mathbf{x}_j)$ for $j = 1, \ldots, N$, of the observed variables, $\mathcal{X} = \{\mathbf{x}_1, \ldots, \mathbf{x}_N\}$, by introducing distributions $\mathcal{Q} = \{q(y_1), \ldots, q(y_n)\}$ over the hidden variables $\mathcal{Y} = \{y_1, \ldots, y_N\}$. The bound function $F(\mathcal{Q}, \theta)$ is a function of the negative log likelihoods, $-\ell(\theta|\mathbf{x}_j, y_j)$ for $j = 1, \ldots, N$, of the joint distributions of both the hidden, $\mathcal{Y}$, and the observed variables, $\mathcal{X}$, and the introduced distributions $\mathcal{Q}$ over the hidden variables $\mathcal{Y}$.

The expectation maximization algorithm consists of two steps called the E-step and the M-step. In the E-step, the bound function $F(\mathcal{Q}, \theta^{t-1})$ is minimized over the introduced distributions $\mathcal{Q}$ over the hidden variables $\mathcal{Y}$ while holding the parameters ,θ^{t-1}, found in the previous iteration $t-1$ fixed.

$$\mathcal{Q}^t = \arg\min_{\mathcal{Q}} F(\mathcal{Q}, \theta^{t-1}) \tag{3.15}$$

In the M-step, the bound function $F(\mathcal{Q}^t, \theta)$ is minimized over the parameters θ while holding the distributions, $\mathcal{Q}^t$, found in the E-step fixed.

$$\theta^t = \arg\min_{\theta} F(\mathcal{Q}^t, \theta) \tag{3.16}$$

3.3.2 Bound on Log-likelihood

Fenchel conjugate duality relation between the logsum and the negative entropy function provides a mathematically principled way to bound the log likelihoods of the marginal distributions with a bound function which is a function of the log likelihoods of the joint distributions and distributions over hidden variables.

Definition 21. *The logsum function* $\Phi(\ell(y)) = \log \sum_{k=1}^{K} \exp \ell(y = k)$ *and the negative entropy function* $-H(q(y)) = \sum_{k=1}^{K} q(y = k) \log q(y = k)$ *are Fenchel conjugate dual functions and they satisfy the Fenchel inequality [36], [130]*

$$\Phi(\ell(y)) - H(q(y)) \geq \sum_{k=1}^{K} q(y = k)\ell(y = k) \quad (3.17)$$

$$\log \sum_{k=1}^{K} \exp \ell(y = k) + \sum_{k=1}^{K} q(y = k) \log q(y = k) \geq \sum_{k=1}^{K} q(y = k)\ell(y = k) \quad (3.18)$$

We can find the upper bound function by using the Fenchel inequality relation between the logsum and the negative entropy functions in (3.18) by substituting $\ell(y) = \log p(\mathbf{x}, y|\theta)$ as follows

$$\begin{aligned}
&\log \sum_{k=1}^{K} \exp \log p(\mathbf{x}, y = k|\theta) + \sum_{k=1}^{K} q(y = k) \log q(y = k) \\
&\qquad \geq \sum_{k=1}^{K} q(y = k) \log p(\mathbf{x}, y = k|\theta) \\
&\log \sum_{k=1}^{K} p(\mathbf{x}, y = k|\theta) + \sum_{k=1}^{K} q(y = k) \log q(y = k) \\
&\qquad \geq \sum_{k=1}^{K} q(y = k) \log p(\mathbf{x}, y = k|\theta) \\
&\log p(\mathbf{x}|\theta) + \sum_{k=1}^{K} q(y = k) \log q(y = k) \geq \sum_{k=1}^{K} q(y = k) \log p(\mathbf{x}, y = k|\theta) \\
&- \log p(\mathbf{x}|\theta) - \sum_{k=1}^{K} q(y = k) \log q(y = k) \leq - \sum_{k=1}^{K} q(y = k) \log p(\mathbf{x}, y = k|\theta)
\end{aligned} \quad (3.19)$$

Hence we have

$$-\log p(\mathbf{x}|\theta) \leq -\sum_{k=1}^{K} q(y=k)\log p(\mathbf{x}, y=k|\theta) + \sum_{k=1}^{K} q(y=k)\log q(y=k) \tag{3.20}$$

Using the inequality (3.20) we define the bound function $F(q(y_j), \theta)$ on the negative log-likelihood of individual observed variables $-\ell(\theta|\mathbf{x}_j)$ as follows

$$\begin{aligned} -\ell(\theta|\mathbf{x}_j) &= -\log p(\mathbf{x}_j|\theta) \\ &\leq -\sum_{k=1}^{K} q(y_j=k)\log p(\mathbf{x}_j, y_j=k|\theta) + \sum_{k=1}^{K} q(y_j=k)\log q(y_j=k) \\ &= F(q(y_j), \theta) \end{aligned} \tag{3.21}$$

Bound function $F(q(y_j), \theta)$ is function of the parameters θ and the unknown distribution $q(y_j)$. We will write the bound function in two different forms that provides us two different insights.

3.3.3 E-step

To get an insight for the E-step, we rewrite the bound function in 3.21 as function of the negative log-likelihood of the observed variables, $-\ell(\theta|\mathbf{x}_j) = -\log p(\mathbf{x}_j|\theta)$,

and the distributions over hidden variables $q(y_j)$. Notice that

$$\begin{aligned}
F(q(y_j), \theta) &= -\sum_{k=1}^{K} q(y_j = k) \log p(\mathbf{x}_j, y_j = k|\theta) + \sum_{k=1}^{K} q(y_j = k) \log q(y_j = k) \\
&= -\sum_{k=1}^{K} q(y_j = k) \log p(y_j = k|\mathbf{x}_j, \theta) p(y_j = k|\theta) \\
&\quad + \sum_{k=1}^{K} q(y_j = k) \log q(y_j = k) \\
&= -\sum_{k=1}^{K} q(y_j = k) \log \frac{p(y_j = k|\mathbf{x}_j, \theta) p(y_j = k|\theta)}{q(y_j = k)} \\
&= -\sum_{k=1}^{K} q(y_j = k) \log \frac{p(y_j = k|\mathbf{x}_j, \theta)}{q(y_j = k)} - \sum_{k=1}^{K} q(y_j = k) \log p(\mathbf{x}_j|\theta) \\
&= \sum_{k=1}^{K} \left(q(y_j = k) \log \frac{q(y_j = k)}{p(y_j = k|\mathbf{x}_j, \theta)} \right) - \log p(\mathbf{x}_j|\theta) \\
&= KL(q(y_j)||p(y_j|\mathbf{x}_j, \theta)) - \ell(\theta|\mathbf{x}_j) \qquad (3.22)
\end{aligned}$$

Then the overall bound function $F(\mathcal{Q}, \theta)$ is

$$\begin{aligned}
F(\mathcal{Q}, \theta) &= \frac{1}{N} \sum_{j=1}^{N} \left(KL(q(y_j)||p(y_j|\mathbf{x}_j, \theta)) - \ell(\theta|\mathbf{x}_j) \right) \\
&= \frac{1}{N} \sum_{j=1}^{N} KL(q(y_j)||p(y_j|\mathbf{x}_j, \theta)) - \frac{1}{N} \sum_{j=1}^{N} \ell(\theta|\mathbf{x}_j) \\
&= KL_N(\mathcal{Q}||p(\mathcal{Y}|\mathcal{X}, \theta)) + \ell_N(\theta|\mathcal{X}) \qquad (3.23)
\end{aligned}$$

where we defined

$$KL_N(\mathcal{Q}||p(\mathcal{Y}|\mathcal{X}, \theta)) = \frac{1}{N} \sum_{j=1}^{N} KL(q(y_j)||p(y_j|\mathbf{x}_j, \theta)) \qquad (3.24)$$

and

$$\ell_N(\theta|\mathcal{X}) = -\frac{1}{N} \sum_{j=1}^{N} \ell(\theta|\mathbf{x}_j) \qquad (3.25)$$

In the E-step, we minimize the bound function

$$F(\mathcal{Q}, \theta^{(t-1)}) = KL_N(\mathcal{Q}||p(\mathcal{Y}|\mathcal{X}, \theta^{(t-1)})) + \ell_N(\theta^{(t-1)}|\mathcal{X})$$

with respect to the distributions over the hidden variables $\mathcal{Q} = \{q(y_1), \ldots, q(y_N)\}$ for fixed parameters $\theta^{(t-1)}$. Notice that $\ell_N(\theta^{(t-1)}|\mathcal{X})$ does not depend on $\mathcal{Q}$, therefore, the E-step in the EM algorithm can be interpreted as minimizing the difference between the sum of the negative log-likelihoods of the observed variables, $\ell_N(\theta^{(t-1)}|\mathcal{X})$, and the bound function $F(\mathcal{Q}, \theta^{(t-1)})$ which can be seen by looking at the following

$$\begin{aligned}\ell_N(\theta^{(t-1)}|\mathcal{X}) &\leq F(\mathcal{Q}, \theta^{(t-1)}) \\ &= KL_N(\mathcal{Q}||p(\mathcal{Y}|\mathcal{X}, \theta^{(t-1)})) + \ell_N(\theta^{(t-1)}|\mathcal{X}) \end{aligned} \tag{3.26}$$

Since E-step simply corresponds to minimizing the KL divergence term $KL_N(\mathcal{Q}||p(\mathcal{Y}|\mathcal{X}, \theta^{(t-1)}))$, we can write the E-step as follows

$$\mathcal{Q}^t = \arg\min_{\mathcal{Q}} KL_N(\mathcal{Q}||p(\mathcal{Y}|\mathcal{X}, \theta^{(t-1)})) \tag{3.27}$$

Furthermore, setting the distributions over hidden variables $q(y_j)$ to the posterior distributions $p(y_j|\mathbf{x}_j, \theta^{(t-1)})$ not only minimizes the sum of KL divergence terms but also makes the sum zero. In other words we have

$$KL_N(\mathcal{Q}||p(\mathcal{Y}|\mathcal{X}, \theta^{(t-1)})) = 0 \text{ for } q(y_j) = p(y_j|\mathbf{x}_j, \theta^{(t-1)}),\ j = 1, \ldots, N \tag{3.28}$$

Thus after the E-step, the original objective function $\ell_N(\theta^{(t-1)}|\mathcal{X})$ and the bound function $F(p(\mathcal{Y}|\mathcal{X}, \theta^{(t-1)}), \theta^{(t-1)})$ becomes equal because we have

$$\begin{aligned}\ell_N(\theta^{(t-1)}|\mathcal{X}) &\leq F(\mathcal{Q}, \theta^{(t-1)}) \\ &= KL_N(p(\mathcal{Y}|\mathcal{X}, \theta^{(t-1)})||p(\mathcal{Y}|\mathcal{X}, \theta^{(t-1)})) + \ell_N(\theta^{(t-1)}|\mathcal{X}) \\ &= 0 + \ell_N(\theta^{(t-1)}|\mathcal{X}) \\ &= \ell_N(\theta^{(t-1)}|\mathcal{X}) \end{aligned} \tag{3.29}$$

3.3.4 Primal Problem for the M-step

To get an insight for the M-step, we rewrite the bound function in 3.21 as function of joint negative log-likelihoods, $-\log p(\mathbf{x}_j, y_j|\theta)$ as follows

$$\begin{aligned}F(q(y_j), \theta) &= -\sum_{k=1}^{K} q(y_j = k) \log p(\mathbf{x}_j, y_j = k|\theta) + \sum_{k=1}^{K} q(y_j = k) \log q(y_j = k) \\ &= E_{q(y_j)}[-\log p(\mathbf{x}_j, y_j|\theta)] - H(q(y_j)) \end{aligned} \tag{3.30}$$

Then the overall bound function $F(\mathcal{Q}, \theta)$ can be written as

$$\begin{aligned} F(\mathcal{Q}, \theta) &= \frac{1}{N} \sum_{j=1}^{N} \Big(E_{q(y_j)}[-\log p(\mathbf{x}_j, y_j|\theta)] - H(q(y_j)) \Big) \\ &= \frac{1}{N} \sum_{j=1}^{N} E_{q(y_j)}[-\log p(\mathbf{x}_j, y_j|\theta)] + \frac{1}{N} \sum_{j=1}^{N} -H(q(y_j)) \\ &= E_{\mathcal{Q}}[-\log p(\mathcal{X}, \mathcal{Y}|\theta)] + H_N(\mathcal{Q}) \end{aligned} \tag{3.31}$$

where we defined

$$E_{\mathcal{Q}}[-\log p(\mathcal{X}, \mathcal{Y}|\theta)] = \frac{1}{N} \sum_{j=1}^{N} E_{q(y_j)}[-\log p(\mathbf{x}_j, y_j|\theta)] \tag{3.32}$$

and

$$H_N(\mathcal{Q}) = \frac{1}{N} \sum_{j=1}^{N} -H(q(y_j)) \tag{3.33}$$

In the M-step, we minimize the bound function

$$F(\mathcal{Q}^{(t)}, \theta) = E_{\mathcal{Q}^{(t)}}[-\log p(\mathcal{X}, \mathcal{Y}|\theta)] + H_N(\mathcal{Q}^{(t)})$$

with respect to the parameters θ for fixed distribution over hidden variables $\mathcal{Q}^{(t)}$. Notice that $H_N(\mathcal{Q}^{(t)})$ does not depend on the parameters θ, therefore, the M-step in the EM algorithm can be interpreted as minimizing sum of the expected negative log-likelihoods of both the observed and the hidden variables, $E_{\mathcal{Q}^{(t)}}[-\log p(\mathcal{X}, \mathcal{Y}|\theta)]$. Thus we can write the M-step as follows

$$\theta^t = \arg\min_{\theta} E_{\mathcal{Q}^{(t)}}[-\log p(\mathcal{X}, \mathcal{Y}|\theta)] \tag{3.34}$$

Considering the Gaussian mixture distribution in exponential family form $p(\mathbf{x}, y|\theta) \in \mathcal{F}$, expected joint negative log-likelihoods $E_{q(y_j)}[-\log p(\mathbf{x}_j, y_j|\theta)]$ can be expressed as

$$
\begin{aligned}
E_{q(y_j)}[-\log p(\mathbf{x}_j, y_j|\theta)] &= E_{q(y_j)}[-\log \exp\left(\theta^T \phi(\mathbf{x}_j, y_j) - \Phi(\theta, y_j)\right)] \\
&= E_{q(y_j)}[-\theta^T \phi(\mathbf{x}_j, y_j) + \Phi(\theta, y_j)] \\
&= E_{q(y_j)}[-\theta^T \phi(\mathbf{x}_j, y_j)] + E_{q(y_j)}[\Phi(\theta, y_j)] \\
&= -\theta^T E_{q(y_j)}[\phi(\mathbf{x}_j, y_j)] + E_{q(y_j)}[\Phi(\theta, y_j)] \\
&= E_{q(y_j)}[\Phi(\theta, y_j)] - \theta^T \left(E_{q(y_j)}[\phi(\mathbf{x}_j, y_j)]\right) \\
&= E_{q(y_j)}[\Phi(\theta_y) + \Phi(\theta_{\mathbf{x}|y}, y_j)] \\
&\quad - \theta_y^T (E_{q(y_j)}[\phi_y(y_j)]) - \sum_{k=1}^{K} \theta_{\mathbf{x}|y=k}^T (E_{q(y_j)}[\phi_{\mathbf{x}|y=k}(\mathbf{x}_j, y_j)]) \\
&= E_{q(y_j)}[\Phi(\theta_y) + \sum_{k=1}^{K} \delta_{yjk} \Phi(\theta_{\mathbf{x}|y=k})] \\
&\quad - \sum_{k=1}^{K-1} \theta_{y=k} E_{q(y_j)}[\delta_{yjk}] - \sum_{k=1}^{K} \theta_{\mathbf{x}|y=k}^T (E_{q(y_j)}[\delta_{yjk} \phi_{\mathbf{x}}(\mathbf{x}_j)]) \\
&= \Phi(\theta_y) + \sum_{k=1}^{K} E_{q(y_j)}[\delta_{yjk}] \Phi(\theta_{\mathbf{x}|y=k}) \\
&\quad - \sum_{k=1}^{K-1} \theta_{y=k} E_{q(y_j)}[\delta_{yjk}] - \sum_{k=1}^{K} \theta_{\mathbf{x}|y=k}^T E_{q(y_j)}[\delta_{yjk}] \phi_{\mathbf{x}}(\mathbf{x}_j) \\
&= \Phi(\theta_y) + \sum_{k=1}^{K} q(y_j = k) \Phi(\theta_{\mathbf{x}|y=k}) \\
&\quad - \sum_{k=1}^{K-1} \theta_{y=k} q(y_j = k) - \sum_{k=1}^{K} \theta_{\mathbf{x}|y=k}^T q(y_j = k) \phi_{\mathbf{x}}(\mathbf{x}_j)
\end{aligned}
\tag{3.35}
$$

By substituting the individual terms in (3.35), we can write the

$E_{\mathcal{Q}}[-\log p(\mathcal{X}, \mathcal{Y}|\theta)]$ as follows

$$
\begin{aligned}
&= \frac{1}{N}\sum_{j=1}^{N}\Big(E_{q(y_j)}[-\log p(\mathbf{x}_j, y_j|\theta)]\Big) \\
&= \frac{1}{N}\sum_{j=1}^{N}\Big(\Phi(\theta_y) + \sum_{k=1}^{K} q(y_j = k)\Phi(\theta_{\mathbf{x}|y=k}) \\
&\quad - \sum_{k=1}^{K-1}\theta_{y=k}q(y_j = k) - \sum_{k=1}^{K}\theta_{\mathbf{x}|y=k}^{T}q(y_j = k)\phi_{\mathbf{x}}(\mathbf{x}_j)\Big) \\
&= \Phi(\theta_y) + \sum_{k=1}^{K}\Big(\frac{1}{N}\sum_{j=1}^{N} q(y_j = k)\Big)\Phi(\theta_{\mathbf{x}|y=k}) \\
&\quad - \sum_{k=1}^{K-1}\theta_{y=k}\Big(\frac{1}{N}\sum_{j=1}^{N} q(y_j = k)\Big) - \sum_{k=1}^{K}\theta_{\mathbf{x}|y=k}^{T}\Big(\frac{1}{N}\sum_{j=1}^{N} q(y_j = k)\phi_{\mathbf{x}}(\mathbf{x}_j)\Big) \\
&= \Phi(\theta_y) + \sum_{k=1}^{K}\alpha_{sk}\Phi(\theta_{\mathbf{x}|y=k}) - \sum_{k=1}^{K-1}\theta_{y=k}\nu_{sy=k} - \sum_{k=1}^{K}\alpha_{sk}\theta_{\mathbf{x}|y=k}^{T}\nu_{s\mathbf{x}|y=k}
\end{aligned}
\tag{3.36}
$$

where $\alpha_{sk} = \frac{1}{N}\sum_{j=1}^{N} q(y_j = k)$ for $k = 1, \ldots, K$ denote the expected empirical probabilities of Gaussian components, $\nu_{sy=k} = \alpha_{sk}$ for $k = 1, \ldots, K-1$ denote the expected empirical moments of y, $\nu_{s\mathbf{x}|y=k} = \frac{1}{\alpha_{sk}N}\sum_{j=1}^{N} q(y_j = k)\phi_{\mathbf{x}}(\mathbf{x}_j)$ for $k = 1, \ldots, K$ denote the expected empirical moments of $\mathbf{x}|y$.

Now, we can write the optimization problem for the M-step as

$$
\text{minimize} \;\; \Phi(\theta_y) + \sum_{k=1}^{K}\alpha_{sk}\Phi(\theta_{\mathbf{x}|y=k}) - \sum_{k=1}^{K-1}\theta_{y=k}\nu_{sy=k} - \sum_{k=1}^{K}\alpha_{sk}\theta_{\mathbf{x}|y=k}^{T}\nu_{s\mathbf{x}|y=k}
\tag{3.37}
$$

where $\theta \in \mathbb{R}^n$ is the optimization variable. There is an implicit constraint $\theta \in C_\theta$ denoting the convex set of parameter values where the log partition function $\Phi(\theta)$ is well-defined incorporated into the domain of the $\Phi(\theta)$.

Proposition 7. *The bound minimization problem in 3.37 corresponding to the M-step for Gaussian mixture models parameterized by the natural parameters θ, is a convex optimization problem in optimization variables θ.*

Proof. As shown in Proposition 2, log partition functions $\Phi(\theta_y)$, $\Phi(\theta_{\mathbf{x}|y=1})$, $\ldots$,$\Phi(\theta_{\mathbf{x}|y=K})$ are convex in θ and nonnegative combinations of convex functions,

$\Phi(\theta_y) + \sum_{k=1}^{K} \alpha_{sk}\Phi(\theta_{\mathbf{x}|y=k})$, defined over the convex set C_θ is convex function of θ [36]. Notice that $-\sum_{k=1}^{K-1} \theta_{y=k}\nu_{sy=k} - \sum_{k=1}^{K} \alpha_{sk}\theta_{\mathbf{x}|y=k}^T\nu_{s\mathbf{x}|y=k}$ is a linear function of θ and since convex function plus a linear function is convex [36], we conclude that the bound minimization problem in 3.37 is a convex optimization problem in variables θ. □

3.3.5 Dual Problem for the M-step

We have seen that the M-step corresponds to convex optimization problem in natural parameters θ. Now we will form the Lagrange dual optimization problem which will correspond to a convex optimization problem in moment parameters ν.

The Lagrange dual function of the problem in (3.37) is the constant p^* where

$$p^* = \inf_{\theta \in \mathbf{dom}\,\Phi} \Phi(\theta_y) + \sum_{k=1}^{K} \alpha_{sk}\Phi(\theta_{\mathbf{x}|y=k}) - \sum_{k=1}^{K-1} \theta_{y=k}\nu_{sy=k} - \sum_{k=1}^{K} \alpha_{sk}\theta_{\mathbf{x}|y=k}^T\nu_{s\mathbf{x}|y=k}. \tag{3.38}$$

Now let us reformulate the problem in (3.37) as

$$\begin{aligned} \text{minimize} \quad & \Phi(\theta_y) + \sum_{k=1}^{K} \alpha_{sk}\Phi(\theta_{\mathbf{x}|y=k}) - \bar{\theta_y}^T\nu_{sy} - \sum_{k=1}^{K} \alpha_{sk}\bar{\theta}_{\mathbf{x}|y=k}^T\nu_{s\mathbf{x}|y=k} \\ \text{subject to} \quad & \bar{\theta_y} = \theta_y \\ & \alpha_{sk}\bar{\theta}_{\mathbf{x}|y=k} = \alpha_{sk}\theta_{\mathbf{x}|y=k} \quad \text{for} \quad k = 1, \ldots, K \end{aligned} \tag{3.39}$$

Here we introduced new variables $\bar{\theta} \in \mathbb{R}^n$, as well as new equality constraints $\bar{\theta_y} = \theta_y$ and $\alpha_{sk}\bar{\theta}_{\mathbf{x}|y=k} = \alpha_{sk}\theta_{\mathbf{x}|y=k}$ for $k = 1, \ldots, K$. Here we assume that α_{sk}'s are positive real numbers. The only reason for using scaled equality constraints $\alpha_{sk}\bar{\theta}_{\mathbf{x}|y=k} = \alpha_{sk}\theta_{\mathbf{x}|y=k}$ instead of unscaled equality constraints $\bar{\theta}_{\mathbf{x}|y=k} = \theta_{\mathbf{x}|y=k}$ is to avoid the rescaling of the Lagrange multipliers (dual variables in the dual problem) which would lead to a cluttered derivation. The problems in (3.37) and (3.39) are clearly equivalent. The Lagrangian $L : \mathbb{R}^n \times \mathbb{R}^n \times \mathbb{R}^n \to \mathbb{R}$ of the

reformulated problem in (3.39) is

$$L(\theta,\bar{\theta},\nu) =\Phi(\theta_y) + \sum_{k=1}^{K} \alpha_{sk}\Phi(\theta_{\mathbf{x}|y=k}) - \bar{\theta_y}^T\nu_{sy} - \sum_{k=1}^{K} \alpha_{sk}\bar{\theta}^T_{\mathbf{x}|y=k}\nu_{s\mathbf{x}|y=k} + \nu_y^T(\bar{\theta}_y - \theta_y) + \sum_{k=1}^{K} \nu^T_{\mathbf{x}|y=k}\big(\alpha_{sk}\bar{\theta}_{\mathbf{x}|y=k} - \alpha_{sk}\theta_{\mathbf{x}|y=k}\big) \tag{3.40}$$

where the variables $\nu = (\nu_y^T, \nu^T_{\mathbf{x}|y=1}, \ldots, \nu^T_{\mathbf{x}|y=K})^T \in \mathbb{R}^n$ are the Lagrange multipliers.

To find the Lagrange dual function we minimize L over θ and $\bar{\theta}$. The Lagrange dual function $g : \mathbb{R}^n \to \mathbb{R}$ is

$$\begin{aligned} g(\nu) &= \inf_{\theta\in\mathbf{dom}\,\Phi,\bar{\theta}\in\mathbb{R}^n} L(\theta,\bar{\theta},\nu) \\ &= \inf_{\theta\in\mathbf{dom}\,\Phi,\bar{\theta}\in\mathbb{R}^n} \Phi(\theta_y) + \sum_{k=1}^{K} \alpha_{sk}\Phi(\theta_{\mathbf{x}|y=k}) - \bar{\theta_y}^T\nu_{sy} - \sum_{k=1}^{K} \alpha_{sk}\bar{\theta}^T_{\mathbf{x}|y=k}\nu_{s\mathbf{x}|y=k} \\ &\quad + \nu_y^T(\bar{\theta}_y - \theta_y) + \sum_{k=1}^{K} \alpha_{sk}\nu^T_{\mathbf{x}|y=k}\big(\bar{\theta}_{\mathbf{x}|y=k} - \theta_{\mathbf{x}|y=k}\big). \end{aligned} \tag{3.41}$$

The Lagrangian L is separable in θ and $\bar{\theta}$, therefore it can be infimized separately over θ and $\bar{\theta}$.

$$\begin{aligned} g(\nu) = &\inf_{\theta\in\mathbf{dom}\,\Phi} \Phi(\theta_y) - \theta_y^T\nu_y + \sum_{k=1}^{K} \alpha_{sk}\big(\Phi(\theta_{\mathbf{x}|y=k}) - \theta^T_{\mathbf{x}|y=k}\nu_{\mathbf{x}|y=k}\big) \\ &+ \inf_{\bar{\theta}\in\mathbb{R}^n} \bar{\theta}_y^T(\nu_y - \nu_{sy}) + \sum_{k=1}^{K} \alpha_{sk}\bar{\theta}^T_{\mathbf{x}|y=k}(\nu_{\mathbf{x}|y=k} - \nu_{s\mathbf{x}|y=k}). \end{aligned} \tag{3.42}$$

Using the Fenchel duality relations (2.35) between the log partition functions $\Phi(\theta_y)$,$\Phi(\theta_{\mathbf{x}|y=1}), \ldots, \Phi(\theta_{\mathbf{x}|y=K})$ and the entropy functions $H(\nu_y)$, $H(\nu_{\mathbf{x}|y=1})$, $\ldots, H(\nu_{\mathbf{x}|y=K})$, i.e.,

$$H(\nu) = \inf_{\theta\in\mathbf{dom}\,\Phi} \Phi(\theta) - \theta^T\nu \tag{3.43}$$

we have

$$\begin{aligned} g(\nu) =&H(\nu_y) + \sum_{k=1}^{K} \alpha_{sk}H(\nu_{\mathbf{x}|y=k}) \\ &+ \inf_{\bar{\theta}\in\mathbb{R}^n} \bar{\theta}_y^T(\nu_y - \nu_{sy}) + \sum_{k=1}^{K} \alpha_{sk}\bar{\theta}^T_{\mathbf{x}|y=k}(\nu_{\mathbf{x}|y=k} - \nu_{s\mathbf{x}|y=k}). \end{aligned} \tag{3.44}$$

Notice that Lagrangian is linear in $\bar{\theta}_y, \bar{\theta}_{\mathbf{x}|y=1}, \ldots, \bar{\theta}_{\mathbf{x}|y=K}$ so $g(\nu) = -\infty$ unless $\nu_y - \nu_{sy} = 0, \nu_{\mathbf{x}|y=k} - \nu_{s\mathbf{x}|y=k} = 0, \ldots, \nu_{\mathbf{x}|y=k} - \nu_{s\mathbf{x}|y=k} = 0$. So the dual function $g(\nu)$ is

$$g(\nu) = \begin{cases} H(\nu_y) + \sum_{k=1}^{K} \alpha_{sk} H(\nu_{\mathbf{x}|y=k}), & \text{if } \nu = \nu_s \\ -\infty, & \text{otherwise} \end{cases} \tag{3.45}$$

Thus, the Lagrange dual of the reformulated problem can be expressed as

$$\begin{aligned} &\text{maximize} \quad H(\nu_y) + \sum_{k=1}^{K} \alpha_{sk} H(\nu_{\mathbf{x}|y=k}) \\ &\text{subject to } \nu = \nu_s \end{aligned} \tag{3.46}$$

Because of the equality constraints $\nu = \nu_s$, the dual problem (3.46) is not suitable for adding new constraints on the moment parameters ν. Hence we will reformulate the dual as an unconstrained optimization problem like the primal expected maximum likelihood problem (3.37) which has the same optimum solution as

$$\text{maximize} \quad H(\nu_y) + \sum_{k=1}^{K} \alpha_{sk} H(\nu_{\mathbf{x}|y=k}) + \nu_y^T \theta_{sy} + \sum_{k=1}^{K} \alpha_{sk} \nu_{\mathbf{x}|y=k}^T \theta_{s\mathbf{x}|y=k} \tag{3.47}$$

where the moment parameters $\nu = (\nu_y^T, \nu_{\mathbf{x}|y=1}^T, \ldots, \nu_{\mathbf{x}|y=K}^T)^T \in \mathbb{R}^n$ are the optimization variables and the expected empirical natural parameters are denoted by $\theta_s = (\theta_{sy}^T, \theta_{s\mathbf{x}|y=1}^T, \ldots, \theta_{s\mathbf{x}|y=K}^T)^T$.

Proposition 8. *The optimum solutions of the expected maximum likelihood problem in* (3.37) *and the unconstrained dual problem in* (3.47) *leads to same optimal parameters.*

Proof. Notice that both problems are unconstrained optimization problems so we can solve both problems by setting the gradients of the corresponding objective functions w.r.t to the corresponding optimization variables to zero. We will use the gradient mapping properties of the log partition functions and the negative entropy functions given in (2.8). Recall that The gradient of the log

partition function w.r.t to the natural parameters equals to the moment parameters, i.e.,$\nabla_\theta \Phi(\theta) = \nu$, and the gradient of the negative entropy function w.r.t. the moment parameters equals to the natural parameters, i.e., $-\nabla_\nu H(\nu) = \theta$. We start by taking the gradient of the expected maximum likelihood problem in (3.37) and setting it to the zero where we have

$$\nabla_\theta \Big(\Phi(\theta_y) + \sum_{k=1}^{K} \alpha_{sk} \Phi(\theta_{\mathbf{x}|y=k}) - \theta_y^T \nu_{sy} - \sum_{k=1}^{K} \alpha_{sk} \theta_{\mathbf{x}|y=k}^T \nu_{s\mathbf{x}|y=k} \Big) = 0$$

$$\begin{bmatrix} \nabla_{\theta_y} \big(\Phi(\theta_y) - \theta_y^T \nu_{sy} \big) \\ \nabla_{\theta_{x|y=1}} \big(\alpha_{s1} \Phi(\theta_{\mathbf{x}|y=1}) - \alpha_{s1} \theta_{\mathbf{x}|y=1}^T \nu_{sx|y=1} \big) \\ \vdots \\ \nabla_{\theta_{x|y=K}} \big(\alpha_{sK} \Phi(\theta_{\mathbf{x}|y=K}) - \alpha_{sK} \theta_{\mathbf{x}|y=K}^T \nu_{sx|y=K} \big) \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ \vdots \\ 0 \end{bmatrix}$$

which can be written as

$$\begin{bmatrix} \nabla_{\theta_y} \Phi(\theta_y) \\ \nabla_{\theta_{x|y=1}} \Phi(\theta_{\mathbf{x}|y=1}) \\ \vdots \\ \nabla_{\theta_{x|y=K}} \Phi(\theta_{\mathbf{x}|y=K}) \end{bmatrix} = \begin{bmatrix} \nu_{sy} \\ \nu_{sx|y=1} \\ \vdots \\ \nu_{sx|y=K} \end{bmatrix} \tag{3.48}$$

Since $\nabla_{\theta_y} \Phi(\theta_y) = \nu_y$ and $\nabla_{\theta_{x|y=k}} \Phi(\theta_{\mathbf{x}|y=k}) = \nu_{x|y=k}$. we have

$$\begin{bmatrix} \nu_y \\ \nu_{x|y=1} \\ \vdots \\ \nu_{x|y=K} \end{bmatrix} = \begin{bmatrix} \nu_{sy} \\ \nu_{sx|y=1} \\ \vdots \\ \nu_{sx|y=K} \end{bmatrix} \tag{3.49}$$

Hence setting gradient equal to zero leads to $\nu_y = \nu_{sy}$ and $\nu_{x|y=k} = \nu_{sx|y=k}$ for $k = 1, \ldots, K$.

Now we take the gradient of the objective function of the unconstrained dual

problem in (3.47) and set it to zero

$$\nabla_{\nu}\Big(H(\nu_y)+\sum_{k=1}^{K}\alpha_{sk}H(\nu_{\mathbf{x}|y=k})+\nu_y^T\theta_{sy}+\sum_{k=1}^{K}\alpha_{sk}\nu_{\mathbf{x}|y=k}^T\theta_{s\mathbf{x}|y=k}\Big)=0$$

$$\begin{bmatrix}\nabla_{\nu_y}\big(H(\nu_y)+\nu_y^T\theta_{sy}\big)\\ \nabla_{\nu_{x|y=1}}\big(\alpha_{s1}H(\nu_{\mathbf{x}|y=1})+\alpha_{s1}\nu_{\mathbf{x}|y=1}^T\theta_{s\mathbf{x}|y=1}\big)\\ \vdots\\ \nabla_{\nu_{x|y=K}}\big(\alpha_{sK}H(\nu_{\mathbf{x}|y=K})+\alpha_{sK}\nu_{\mathbf{x}|y=K}^T\theta_{s\mathbf{x}|y=K}\big)\end{bmatrix}=\begin{bmatrix}0\\0\\\vdots\\0\end{bmatrix}$$

which can be written as

$$\begin{bmatrix}\nabla_{\nu_y}H(\nu_y)\\ \nabla_{\nu_{x|y=1}}H(\nu_{\mathbf{x}|y=1})\\ \vdots\\ \nabla_{\nu_{x|y=K}}H(\nu_{\mathbf{x}|y=K})\end{bmatrix}=\begin{bmatrix}-\theta_{sy}\\ -\theta_{s\mathbf{x}|y=1}\\ \vdots\\ -\theta_{s\mathbf{x}|y=K}\end{bmatrix}\tag{3.50}$$

Since $\nabla_{\nu_y}H(\nu_y)=-\theta_y$ and $\nabla_{\nu_{x|y=k}}H(\nu_{\mathbf{x}|y=k})=-\theta_{\mathbf{x}|y=k}$. we have

$$\begin{bmatrix}\theta_y\\ \theta_{\mathbf{x}|y=1}\\ \vdots\\ \theta_{\mathbf{x}|y=K}\end{bmatrix}=\begin{bmatrix}\theta_{sy}\\ \theta_{s\mathbf{x}|y=1}\\ \vdots\\ \theta_{s\mathbf{x}|y=K}\end{bmatrix}\tag{3.51}$$

Hence setting gradient equal to zero leads to $\theta_y=\theta_{sy}$ and $\theta_{\mathbf{x}|y=k}=\theta_{s\mathbf{x}|y=k}$ for $k=1,\ldots,K$. Since expected sufficient statistics $\nu_{sy},\nu_{sx|y=1},\ldots,\nu_{sx|y=K}$ and $\theta_{sy},\theta_{s\mathbf{x}|y=1},\ldots,\theta_{s\mathbf{x}|y=K}$ are related through the parameter relations, the found optimum parameters, $\theta_y,\theta_{\mathbf{x}|y=1},\ldots,\theta_{\mathbf{x}|y=K}$ and $\nu_y,\nu_{x|y=1},\ldots,\nu_{x|y=K}$ are related through the parameter relations hence we conclude that the optimum solutions of the expected maximum likelihood problem in (3.37) and the unconstrained dual problem in (3.47) leads to same optimal parameters. □

3.3.6 Parameterizations for the M-step

In this Section, we will express the primal problem for the M-step as a convex optimization problem in terms of the information parameters and the dual problem for the M-step as a convex optimization problem in terms of the source parameters.

3.3.6.1 Primal Problem for the M-step in Information Form

We can write the objective function of the problem in (3.37) in terms of the information parameters $\eta, m_1, S_1, \ldots, m_K, S_K$. As discussed in (2.77) and (2.83), we have the expressions for the log partition functions in terms of the information parameters. We write the natural parameters θ in terms of the information parameters $\eta, m_1, S_1, \ldots, m_K, S_K$ as

$$\theta_y = \eta \tag{3.52}$$

$$\theta_{\mathbf{x}|y=k} = (m_k^T, vec(-\frac{1}{2}S_k)^T)^T \quad \text{for } k = 1, \ldots, K \tag{3.53}$$

and the expected empirical moment parameters $\nu_{sy}, \nu_{s\mathbf{x}|y=1}, \ldots, \nu_{s\mathbf{x}|y=K}$ in terms of the source parameters $\alpha_s, \mu_{s1}, \Sigma_{s1}, \ldots, \mu_{sK}, \Sigma_{sK}$ as

$$\nu_{sy} = \alpha_s \tag{3.54}$$

$$\nu_{s\mathbf{x}|y=k} = (\mu_{sk}^T, vec(\Sigma_{sk} + \mu_{sk}\mu_{sk}^T)^T)^T \quad \text{for} \quad k = 1, \ldots, K \tag{3.55}$$

For the log partition functions we have

$$\Phi(\theta_y) = \log(1 + \sum_{k=1}^{K-1} \exp \eta_k) \tag{3.56}$$

$$\Phi(\theta_{\mathbf{x}|y=k}) = -\frac{1}{2}\log|S_k| + \frac{1}{2}m_k^T S_k^{-1} m_k + \frac{d}{2}\log 2\pi \tag{3.57}$$

and for the inner product terms we have

$$\begin{aligned} \theta_y^T \nu_{sy} &= \eta^T \alpha_s \\ &= \sum_{k=1}^{K-1} \eta_k \alpha_{sk} \end{aligned} \tag{3.58}$$

$$\begin{aligned} \theta_{\mathbf{x}|y=k}^T \nu_{s\mathbf{x}|y=1} &= (m_k^T, vec(-\frac{1}{2}S_k)^T)(\mu_{sk}^T, vec(\Sigma_{sk} + \mu_{sk}\mu_{sk}^T)^T)^T \\ &= m_k^T \mu_{sk} + tr\big((-\frac{1}{2}S_k)(\Sigma_{sk} + \mu_{sk}\mu_{sk}^T)\big) \\ &= m_k^T \mu_{sk} - \frac{1}{2}tr\big(S_k(\Sigma_{sk} + \mu_{sk}\mu_{sk}^T)\big) \end{aligned} \tag{3.59}$$

We can write the convex optimization problem for the M-step in terms of the

information parameters as

$$\begin{aligned}\text{minimize} \quad & \log(1+\sum_{k=1}^{K-1}\exp\eta_k)+\sum_{k=1}^{K}\alpha_{sk}\big(-\frac{1}{2}\log|S_k|+\frac{1}{2}m_k^TS_k^{-1}m_k+\frac{d}{2}\log 2\pi\big)\\ & -\sum_{k=1}^{K-1}\eta_k\alpha_{sk}-\sum_{k=1}^{K}\alpha_{sk}\big(m_k^T\mu_{sk}-\frac{1}{2}tr(S_k(\Sigma_{sk}+\mu_{sk}\mu_{sk}^T))\big)\end{aligned} \tag{3.60}$$

where $\eta \in \mathbb{R}^{K-1}$, $m_k \in \mathbb{R}^d$, $S_k \in \mathcal{S}_+^d$ for $k = 1, \ldots, K$ are the optimization variables. The expected empirical probabilities

$$\alpha_{sk} = \frac{1}{N}\sum_{j=1}^{N} q(y_j = k),\ k = 1, \ldots, K$$

the expected empirical means

$$\mu_{sk} = \frac{1}{\alpha_{sk}N}\sum_{j=1}^{N} q(y_j = k)\mathbf{x}_j,\ k = 1, \ldots, K$$

and the expected empirical covariance matrices

$$\Sigma_{sk} = \frac{1}{\alpha_{sk}N}\sum_{j=1}^{N} q(y_j = k)\mathbf{x}_j\mathbf{x}_j^T - \mu_{sk}\mu_{sk}^T,\ k = 1, \ldots, K$$

are the problem parameters which were calculated apriori after the E-step.

Notice that this is an unconstrained optimization problem and the values of the optimization variables depend on the values of the expected sufficient statistics α_{sk},μ_{sk} and Σ_{sk} for $k = 1, \ldots, K$.

3.3.6.2 Dual Problem for the M-step in Source Form

We can write the objective function of the problem in (3.47) in terms of the source parameters $\alpha, \mu_1, \Sigma_1, \ldots, \mu_K, \Sigma_K$. As discussed in (2.69) and (2.83), we have the expressions for the entropy functions in terms of the source parameters. We write the moment parameters ν in terms of the source parameters $\alpha, \mu_1, \Sigma_1, \ldots, \mu_K, \Sigma_K$

as

$$\nu_y = \alpha \tag{3.61}$$

$$\nu_{\mathbf{x}|y=k} = (\mu_k^T, vec(\Sigma_k + \mu_k\mu_k^T)^T)^T \quad \text{for } k = 1, \ldots, K \tag{3.62}$$

and the expected empirical natural parameters $\theta_{sy}, \theta_{s\mathbf{x}|y=1}, \ldots, \theta_{s\mathbf{x}|y=K}$ in terms of the information parameters $\eta_s, m_{s1}, S_{s1}, \ldots, m_{sK}, S_{sK}$ as

$$\theta_{sy} = \eta_s \tag{3.63}$$

$$\theta_{s\mathbf{x}|y=k} = (m_{sk}^T, vec(-\frac{1}{2}S_{sk})^T)^T \quad \text{for} \quad k = 1, \ldots, K \tag{3.64}$$

For the entropy functions we have

$$H(\nu_y) = -\sum_{k=1}^{K-1} \alpha_k \log \alpha_k - (1 - \sum_{k=1}^{K-1} \alpha_k) \log(1 - \sum_{k=1}^{K-1} \alpha_k) \tag{3.65}$$

$$H(\nu_{\mathbf{x}|y=k}) = \frac{1}{2} \log |\Sigma_k| + \frac{d}{2} \log(2\pi e) \tag{3.66}$$

and for the inner product terms we have

$$\begin{aligned} \nu_y^T \theta_{sy} &= \alpha^T \eta_s \\ &= \sum_{k=1}^{K-1} \alpha_k \eta_{sk} \end{aligned} \tag{3.67}$$

$$\begin{aligned} \nu_{\mathbf{x}|y=k}^T \theta_{s\mathbf{x}|y=1} &= (\mu_k^T, vec(\Sigma_k + \mu_k\mu_k^T)^T)(m_{sk}^T, vec(-\frac{1}{2}S_{sk})^T)^T \\ &= \mu_k^T m_{sk} + tr\big((\Sigma_k + \mu_k\mu_k^T)(-\frac{1}{2}S_{sk})\big) \\ &= \mu_k^T m_{sk} - \frac{1}{2}tr(\Sigma_k S_{sk}) - \frac{1}{2}tr(\mu_k\mu_k^T S_{sk}) \\ &= \mu_k^T m_{sk} - \frac{1}{2}tr(\Sigma_k S_{sk}) - \frac{1}{2}\mu_k^T S_{sk}\mu_k \end{aligned} \tag{3.68}$$

Hence we can write the convex optimization problem in (3.47) as

$$\begin{aligned} \text{maximize} \quad & -\sum_{k=1}^{K-1} \alpha_k \log \alpha_k - (1 - \sum_{k=1}^{K-1} \alpha_k) \log(1 - \sum_{k=1}^{K-1} \alpha_k) \\ & + \sum_{k=1}^{K} \alpha_{sk}\big(\frac{1}{2} \log |\Sigma_k| + \frac{d}{2} \log(2\pi e)\big) \\ & + \sum_{k=1}^{K-1} \alpha_k \eta_{sk} + \sum_{k=1}^{K} \alpha_{sk}\big(\mu_k^T m_{sk} - \frac{1}{2}tr(\Sigma_k S_{sk}) - \frac{1}{2}\mu_k^T S_{sk}\mu_k\big) \end{aligned} \tag{3.69}$$

where $\alpha \in \mathbb{R}^{K-1}$, $\mu_k \in \mathbb{R}^d$, $\Sigma_k \in \mathcal{S}_+^d$ for $k = 1, \ldots, K$ are the optimization variables and the expected empirical information parameters denoted by

$$\eta_{sk} = \log \frac{\alpha_{sk}}{1 - \sum_{i=1}^{K-1} \alpha_{si}},\ k = 1, \ldots, K-1$$
$$m_{sk} = \Sigma_{sk}^{-1} \mu_{sk},\ k = 1, \ldots, K$$
$$S_{sk} = \Sigma_{sk}^{-1},\ k = 1, \ldots, K$$

are the problem parameters which were calculated apriori after the E-step using the expected empirical probabilities

$$\alpha_{sk} = \frac{1}{N} \sum_{j=1}^{N} q(y_j = k),\ k = 1, \ldots, K$$

the expected empirical means

$$\mu_{sk} = \frac{1}{\alpha_{sk} N} \sum_{j=1}^{N} q(y_j = k) \mathbf{x}_j,\ k = 1, \ldots, K$$

and the expected empirical covariance matrices

$$\Sigma_{sk} = \frac{1}{\alpha_{sk} N} \sum_{j=1}^{N} q(y_j = k) \mathbf{x}_j \mathbf{x}_j^T - \mu_{sk} \mu_{sk}^T,\ k = 1, \ldots, K$$

.

Notice that this is an unconstrained optimization problem and the values of the optimization variables depend on the values of the expected empirical natural parameters η_{sk},m_{sk} and S_{sk} for $k = 1, \ldots, K$.

3.4 Constrained Gaussian Mixture Model Framework

3.4.1 Problem Definition

We consider the family $\mathcal{F}$ of distributions of Gaussian mixture models with K Gaussian components denoted by $p(\mathbf{x}, y|\theta) \in \mathcal{F}$ over d dimensional continous

random vector $\mathbf{x} \in \mathbb{R}^d$ and a multinomial random variable $y \in \{1, \ldots, K\}$ parametrized with the information parameters $\theta = \{\eta, m_1, S_1, \ldots, m_K, S_K\}$ where $\eta \in \mathbb{R}^{K-1}$, $m_k \in \mathbb{R}^d$, $S_k \in \mathcal{S}_+^d$ for $k = 1, \ldots, K$.

In density estimation problems with constrained Gaussian mixture models, we are given a data set $\mathcal{X} = \{\mathbf{x}_1, \ldots, \mathbf{x}_N\}$ of N independent and identically distributed (i.i.d.) random vectors corresponding to random samples from the marginal distribution $p(\mathbf{x}|\theta) = \sum_{k=1}^{K} p(\mathbf{x}, y = k|\theta)$. In addition, we are given a set of constraints denoted by $\mathcal{C}$ which either can be formulated as convex constraints in the information parameters $\theta = \{\eta, m_1, S_1, \ldots, m_K, S_K\}$ or can be formulated as convex constraints in the source parameters $\nu = \{\alpha, \mu_1, \Sigma_1, \ldots, \mu_K, \Sigma_K\}$ where $\alpha \in \mathbb{R}^{K-1}$, $\mu_k \in \mathbb{R}^d$, $\Sigma_k \in \mathcal{S}_+^d$ for $k = 1, \ldots, K$. In other words, we assume that the given set of constraints $\mathcal{C}$ can be expressed as convex constraints either in terms of the information parameters θ or the source parameters ν. Our objective is to find the maximum likelihood (ML) estimate $\hat{\theta}$ of the model parameters θ satisfying the constraints in $\mathcal{C}$. The ML estimation problem can be written in minimization form in 2.20 as

$$\hat{\theta} = \arg\min_{\theta \in \mathcal{C}} \; -\frac{1}{N} \sum_{j=1}^{N} \ell(\theta|\mathbf{x}_j) \tag{3.70}$$

3.4.2 Expectation Maximization Algorithm

We use the expectation maximization algorithm to solve the maximum likelihood estimation problem in (3.70). In the E-step, we calculate the distributions $\mathcal{Q}$ over hidden variables $\mathcal{Y}$ by solving the following optimization problem

$$\mathcal{Q}^t = \arg\min_{\mathcal{Q}} KL_N(\mathcal{Q}||p(\mathcal{Y}|\mathcal{X}, \theta^{(t-1)})) \tag{3.71}$$

For the M-step we either solve the primal problem where the optimzation variables are the information parameters $\theta = \{\eta, m_1, S_1, \ldots, m_K, S_K\}$ or solve the dual problem where the optimization variables are the source parameters $\nu = \{\alpha, \mu_1, \Sigma_1, \ldots, \mu_K, \Sigma_K\}$ and then calculate the information parameters θ from the source parameters ν. If the constraint set $\mathcal{C}$ can be formulated as convex constraints using the information parameters $\theta = \{\eta, m_1, S_1, \ldots, m_K, S_K\}$, we

solve the primal problem for the M-step. In the primal problem for the M-step we compute the information parameters $\theta = \{\eta, m_1, S_1, \ldots, m_K, S_K\}$ subject to constraints $\theta \in \mathcal{C}_\theta$ where the constraint set is denoted by $\mathcal{C}_\theta$ which consists of the constraints in $\mathcal{C}$ expressed in terms of the information parameters θ by solving the following optimization problem

$$\begin{aligned} \text{minimize} \quad & \log(1 + \sum_{k=1}^{K-1} \exp \eta_k) + \sum_{k=1}^{K} \alpha_{sk} \big(-\frac{1}{2} \log |S_k| + \frac{1}{2} m_k^T S_k^{-1} m_k + \frac{d}{2} \log 2\pi \big) \\ & - \sum_{k=1}^{K-1} \eta_k \alpha_{sk} - \sum_{k=1}^{K} \alpha_{sk} \big(m_k^T \mu_{sk} - \frac{1}{2} tr(S_k(\Sigma_{sk} + \mu_{sk}\mu_{sk}^T)) \big) \\ \text{subject to} \quad & (\eta, m_1, S_1, \ldots, m_K, S_K) \in \mathcal{C}_\theta \end{aligned} \tag{3.72}$$

where $\eta \in \mathbb{R}^{K-1}$, $m_k \in \mathbb{R}^d$, $S_k \in \mathcal{S}_+^d$ for $k = 1, \ldots, K$ are the optimization variables and $\mathcal{C}_\theta$ denotes the convex constraint set including convex inequality and affine equality constraints. The expected empirical probabilities

$$\alpha_{sk} = \frac{1}{N} \sum_{j=1}^{N} q^t(y_j = k), \ k = 1, \ldots, K$$

the expected empirical means

$$\mu_{sk} = \frac{1}{\alpha_{sk} N} \sum_{j=1}^{N} q^t(y_j = k) \mathbf{x}_j, \ k = 1, \ldots, K$$

and the expected empirical covariance matrices

$$\Sigma_{sk} = \frac{1}{\alpha_{sk} N} \sum_{j=1}^{N} q^t(y_j = k) \mathbf{x}_j \mathbf{x}_j^T - \mu_{sk} \mu_{sk}^T, \ k = 1, \ldots, K$$

are the problem parameters which were calculated apriori after the E-step.

On the other hand, if the constraint set $\mathcal{C}$ can be formulated as convex constraints using the source parameters $\nu = \{\alpha, \mu_1, \Sigma_1, \ldots, \mu_K, \Sigma_K\}$, we solve the dual problem for the M-step and then find the information parameters $\theta = \{\eta, m_1, S_1, \ldots, m_K, S_K\}$ using the parameter conversion formulas where $\eta_k = \log \frac{\alpha_k}{1 - \sum_{i=1}^{K-1} \alpha_i}$ for $k = 1, \ldots, K-1$, $m_k = \Sigma_k^{-1} \mu_k$, $S_k = \Sigma_k^{-1}$ for $k = 1, \ldots, K$. In the dual problem for the M-step we compute the source parameters $\nu = \{\alpha, \mu_1, \Sigma_1, \ldots, \mu_K, \Sigma_K\}$ subject to constraints $\nu \in \mathcal{C}_\nu$ where the

constraint set is denoted by $\mathcal{C}_\nu$ which consists of the constraints in $\mathcal{C}$ expressed in terms of the source parameters ν by solving the following optimization problem

$$\begin{aligned}
\text{maximize} \quad & -\sum_{k=1}^{K-1}\alpha_k\log\alpha_k - (1-\sum_{k=1}^{K-1}\alpha_k)\log(1-\sum_{k=1}^{K-1}\alpha_k) \\
& + \sum_{k=1}^{K}\alpha_{sk}\big(\frac{1}{2}\log|\Sigma_k| + \frac{d}{2}\log(2\pi e)\big) \\
& + \sum_{k=1}^{K-1}\alpha_k\eta_{sk} + \sum_{k=1}^{K}\alpha_{sk}\big(\mu_k^T m_{sk} - \frac{1}{2}tr(\Sigma_k S_{sk}) - \frac{1}{2}\mu_k^T S_{sk}\mu_k\big) \\
\text{subject to } & (\alpha,\mu_1,\Sigma_1,\ldots,\mu_K,\Sigma_K)\in\mathcal{C}_\nu
\end{aligned} \tag{3.73}$$

where $\alpha \in \mathbb{R}^{K-1}$, $\mu_k \in \mathbb{R}^d$, $\Sigma_k \in \mathcal{S}_+^d$ for $k = 1,\ldots,K$ are the optimization variables and $\mathcal{C}_\nu$ denotes the convex constraint set including convex inequality and affine equality constraints. The expected empirical information parameters are denoted by

$$\begin{aligned}
\eta_{sk} =& \log\frac{\alpha_{sk}}{1-\sum_{i=1}^{K-1}\alpha_{si}},\ k=1,\ldots,K-1 \\
m_{sk} =& \Sigma_{sk}^{-1}\mu_{sk},\ k=1,\ldots,K \\
S_{sk} =& \Sigma_{sk}^{-1},\ k=1,\ldots,K
\end{aligned}$$

which were calculated apriori after the E-step using the expected empirical probabilities

$$\alpha_{sk} = \frac{1}{N}\sum_{j=1}^{N} q^t(y_j = k),\ k=1,\ldots,K$$

the expected empirical means

$$\mu_{sk} = \frac{1}{\alpha_{sk}N}\sum_{j=1}^{N} q^t(y_j = k)\mathbf{x}_j,\ k=1,\ldots,K$$

and the expected empirical covariance matrices

$$\Sigma_{sk} = \frac{1}{\alpha_{sk}N}\sum_{j=1}^{N} q^t(y_j = k)\mathbf{x}_j\mathbf{x}_j^T - \mu_{sk}\mu_{sk}^T,\ k=1,\ldots,K$$

3.5 Example Constraints

In this Section, we will discuss various practical scenarios and show how different parameter dependency relations can be formulated as convex constraints either on the information or the source parameters of Gaussian mixture models.

First we will consider example cases which can be formulated as convex constraints using the information parameters.

- Known null entries $(i,j) \in \mathcal{I}$ in the information matrices $S_k \in \mathcal{S}_+^d$ $(S_k \succeq 0)$ for $k = 1, \ldots, K$ corresponding to the conditional independence relations [40], [47], [3] between the pair of random variables indexed by i, j can be formulated as linear equality and convex inequality constraints in the variables $S_1, \ldots, S_K$ as

$$\begin{aligned} S_k^{i,j} &= 0 \quad \text{for} \quad (i,j) \in \mathcal{I}, \quad k = 1, \ldots, K \\ S_k &\succeq 0 \quad \text{for} \quad k = 1, \ldots, K \end{aligned} \tag{3.74}$$

- We can constrain any information matrix $S_k \in \mathcal{S}_+^d$ to be diagonal and put nonnegative known upper bounds and lower bounds, $u^{i,i} \geq l^{i,i} \geq 0$, $(i,i) \in \mathcal{I}$ on the diagonal entries. These constraints have been considered as desired properties of the covariance matrices in speech recognition [22], [21], [23]. They can be handled in our framework using linear equality and convex inequality constraints in the variables $S_1, \ldots, S_K$ as

$$\begin{aligned} S_k^{i,j} &= 0 \quad \text{for} \quad i \neq j \\ S_k^{i,i} &\geq l^{i,i} \quad \text{for} \quad (i,i) \in \mathcal{I} \\ S_k^{i,i} &\leq u^{i,i} \quad \text{for} \quad (i,i) \in \mathcal{I} \\ S_k &\succeq 0 \quad \text{for} \quad k = 1, \ldots, K \end{aligned} \tag{3.75}$$

- We can constrain any information matrix $S_k \in \mathcal{S}_+^d$ to be diagonal and be related to a known diagonal information matrix $\tilde{S}_k \in \mathcal{S}_+^d$ via an unknown affine transformation modeled with nonnegative variables $a_1, \ldots, a_d$. These constraints have been considered as desired properties of the covariance

matrices in speech recognition [71], [66], [53]. They can be handled in our framework using linear equality and convex inequality constraints in the variables $S_k, a_1, \ldots, a_d$ as

$$\begin{aligned} S_k^{i,j} &= 0 \quad \text{for} \quad i \neq j \\ S_k^{i,i} &= a_i \tilde{S}_k^{i,i} \quad \text{for} \quad i = 1, \ldots, d \\ S_k &\succeq 0 \\ a_i &\geq 0 \quad \text{for} \quad i = 1, \ldots, d \end{aligned} \tag{3.76}$$

- The constraint $S_k = A\tilde{S}_k A^T$ describes a relation where an arbitrary information matrix $S_k \in \mathcal{S}_+^d$ is related to a known arbitrary information matrix $\tilde{S}_k \in \mathcal{S}_+^d$ via an unknown affine transformation $A \in \mathbb{R}^{d \times m}$. This constraint have been considered as a desired property of the covariance matrices in speech recognition [53]. This constraint does not correspond to an affine equality constraint in the variables S_k, A and since it is not affine it cannot be handled in our framework. However its semi-definite programming (SDP) relaxation $S_k \succeq A\tilde{S}_k A^T$ corresponds to a convex inequality constraint in the variables S_k, A [36].

Next we will consider example cases which can be formulated as convex constraints using the source parameters.

- We can constrain any mean vector $\mu_k \in \mathbb{R}^d$ to be related to a known vector $\tilde{\mu}_k \in \mathbb{R}^m$ via an unknown affine transformation $A \in \mathbb{R}^{d \times m}$, $b \in \mathbb{R}^m$. This constraint have been considered as a desired property of the mean vectors in speech recognition [62], [69], [70], [71], [66]. It can be handled in our framework using linear equality constraints in the variables μ_k, A, b as

$$\mu_k = A\tilde{\mu}_k + b \tag{3.77}$$

- We can constrain the difference of the mean vectors $\mu_i \in \mathbb{R}^d$, $\mu_j \in \mathbb{R}^d$ to be equal to the known displacement vectors $\tilde{d}_{ij} \in \mathbb{R}^d$ plus unknown deviation vectors $t_{ij} \in \mathbb{R}^d$ where the l1 norm of the deviation vectors are constrained to be less than a known positive number $u > 0$ for $i = 1, \ldots, K-1$,

$j = i+1, \ldots, K$ using affine equality and convex inequality constraints in the variables $\mu_1, \ldots, \mu_k, t_{1,2}, \ldots, t_{K,K-1}$ as

$$\begin{aligned} \mu_i + \tilde{d}_{ij} = \mu_j + t_{ij} \quad &\text{for} \quad i = 1, \ldots, K-1, j = i+1, \ldots, K \\ ||t_{ij}||_1 \leq u \quad &\text{for} \quad i = 1, \ldots, K-1, j = i+1, \ldots, K \end{aligned} \tag{3.78}$$

- Known null entries $(i,j) \in \mathcal{I}$ in the covariance matrices $\Sigma_k \in \mathcal{S}_+^d$ ($\Sigma_k \succeq 0$) for $k = 1, \ldots, K$ corresponding to the marginal independence relations [47], [3], between the pair of random variables indexed by i, j can be formulated as linear equality constraints and convex inequality constraints in the variables $\Sigma_1, \ldots, \Sigma_K$ as

$$\begin{aligned} \Sigma_k^{i,j} = 0 \quad &\text{for} \quad (i,j) \in \mathcal{I}, \quad k = 1, \ldots, K \\ \Sigma_k \succeq 0 \quad &\text{for} \quad k = 1, \ldots, K \end{aligned} \tag{3.79}$$

- We can constrain any covariance matrix $\Sigma_k \in \mathcal{S}_+^d$ to be block diagonal, such as

$$\Sigma_k = \begin{bmatrix} \Sigma_k^1 & 0 \\ 0 & \Sigma_k^2 \end{bmatrix}$$

where $\Sigma_k^1 \in \mathcal{S}_+^m$, $\Sigma_k^2 \in \mathcal{S}_+^{d-m}$ and put limits on their corresponding eigenvalues where the eigenvalue limits are known nonnegative numbers, $\tilde{\lambda}_{max,k}^1 \geq \tilde{\lambda}_{min,k}^1 \geq 0$ using linear equality and convex inequality constraints in the variables Σ_k as

$$\begin{aligned} &\Sigma_k^{ij} = 0 \quad \text{for} \quad i = 1, \ldots, m, \ j = m+1, \ldots, d \\ &\Sigma_k^{ij} = 0 \quad \text{for} \quad i = m+1, \ldots, d, \ j = 1, \ldots, m \\ &\Sigma_k^1 \preceq \tilde{\lambda}_{max,k}^1 I_m \\ &\Sigma_k^1 \succeq \tilde{\lambda}_{min,k}^1 I_m \\ &\Sigma_k^2 \preceq \tilde{\lambda}_{max,k}^2 I_{d-m} \\ &\Sigma_k^2 \succeq \tilde{\lambda}_{min,k}^2 I_{d-m} \\ &\Sigma_k \succeq 0 \end{aligned} \tag{3.80}$$

- We can constrain any covariance matrix $\Sigma_k \in \mathcal{S}_+^d$ to be diagonal and put nonnegative known upper and lower bounds, $u^{i,i} \geq l^{i,i} \geq 0$, $(i,i) \in \mathcal{I}$ on

the diagonal entries. These constraints have been considered as desired properties of the covariance matrices in speech recognition [22], [21], [23]. They can be handled in our framework using linear equality and convex inequality constraints in the variables $\Sigma_1, \ldots, \Sigma_K$ as

$$\begin{aligned} \Sigma_k^{i,j} &= 0 \quad \text{for} \quad i \neq j \\ \Sigma_k^{i,i} &\geq l^{i,i} \quad \text{for} \quad (i,i) \in \mathcal{I} \\ \Sigma_k^{i,i} &\leq u^{i,i} \quad \text{for} \quad (i,i) \in \mathcal{I} \\ \Sigma_k &\succeq 0 \quad \text{for} \quad k = 1, \ldots, K \end{aligned} \tag{3.81}$$

- We can constrain any covariance matrix $\Sigma_k \in \mathcal{S}_+^d$ to be diagonal and be related to a known diagonal covariance matrix $\tilde{\Sigma}_k \in \mathcal{S}_+^d$ via an unknown affine transformation modeled with nonnegative variables $a_1, \ldots, a_d$. These constraints have been considered as desired properties of the covariance matrices in speech recognition [71], [66], [53]. They can be handled in our framework using linear equality and convex inequality constraints in the variables $\Sigma_k, a_1, \ldots, a_d$ as

$$\begin{aligned} \Sigma_k^{i,j} &= 0 \quad \text{for} \quad i \neq j \\ \Sigma_k^{i,i} &= a_i \tilde{\Sigma}_k^{i,i} \quad \text{for} \quad i = 1, \ldots, d \\ \Sigma_k &\succeq 0 \\ a_i &\geq 0 \quad \text{for} \quad i = 1, \ldots, d \end{aligned} \tag{3.82}$$

- The constraint $\Sigma_k = A\tilde{\Sigma}_k A^T$ describes a relation where an arbitrary covariance matrix $\Sigma_k \in \mathcal{S}_+^d$ is related to a known arbitrary covariance matrix $\tilde{\Sigma}_k \in \mathcal{S}_+^d$ via an unknown affine transformation $A \in \mathbb{R}^{d \times m}$. This constraint have been considered as a desired property of the covariance matrices in speech recognition [53]. This constraint does not correspond to an affine equality constraint in the variables Σ_k, A and since it is not affine it cannot be handled in our framework. However, its SDP relaxation $\Sigma_k \succeq A\tilde{\Sigma}_k A^T$ corresponds to a convex inequality constraint in the variables Σ_k, A [36].

3.6 Conclusions

A novel constrained Gaussian mixture model framework (CGMM) is proposed to handle the affine equality and convex inequality constraints on either the information or the source parameters. The expectation maximization (EM) algorithm used to estimate the parameters are explained in detail. We have shown that the primal problem for the M-step corresponds to a convex optimization problem in the information parameters and we can handle convex constraints on the information parameters by solving a constrained convex optimization problem. Then, we have developed an unconstrained dual convex optimization problem for the M-step which is convex in the source parameters and suitable for adding new constraints on the source parameters. Thus, we can handle convex constraints on the source parameters by solving the dual convex optimization problem. The unifying idea in this Chapter is that we can handle affine equality and convex inequality constraints on either the information or the source parameters by solving a constrained convex optimization problem for the M-step. Moreover, we have shown that many parameter relations of practical importance can be formulated as convex constraints either using the information or the source parameters.

Chapter 4

Robust Gaussian Mixture Models

4.1 Introduction

In many problems, the data points of interest are observed as part of a larger set of observations where some of the points do not follow the assumed restricted parametric distribution. We refer to the data points being distributed according to the assumed distribution as inliers and the rest of the data points as outliers. In practice, it is hard to know the outlier distributions and hence it is important to have flexible models that make as few assumptions as possible. Furthermore, in outlier detection problems with Gaussian mixture models one needs to select a threshold level so that given new data points, he/she can determine which data points are the outliers. This is time consuming work and it is desirable to automatically determine the threshold level using inlier and outlier information available for few data points.

In this Chapter, we first study a general probabilistic mixture model where initially we assume that we know both the inlier and the outlier distributions. Then, we show that in the E-step of the expectation maximization (EM) algorithm, if we constrain the posteriors distributions to take binary values and assume that the likelihood of any data point being an outlier is a constant value, we do not need any other additional information to detect the outliers. Second,

as an example to the constrained Gaussian mixture model framework, we develop a robust Gaussian mixture model where inlier/outlier information available few data points are incorporated as convex constraints on the information parameters. Using this model we show that we can simultaneously learn both the model parameters that are consistent with this information and determine the threshold value needed to determine the outliers.

The organization of this Chapter is as follows. In Section 4.2 we study a general probabilistic mixture model. In Section 4.3, as an application to constrained Gaussian mixture model framework, we develop a robust Gaussian mixture model where inlier/outlier information available few data points are incorporated as convex constraints on the information parameters. We illustrate the capabilities of the proposed model on two-dimensional data set in Section 4.4. Conclusions are provided in Section 4.5.

4.2 General Robust Model

In many problems, the data points of interest are observed as part of a larger set of observations where some of the points do not follow the assumed restricted parametric distribution $p(\mathbf{x}|\theta)$ parameterized by the parameters θ. We refer to the data points being distributed according to the assumed distribution as inliers and the rest of the data points as outliers. We assume that a given set of N data points $\mathcal{X} = \{\mathbf{x}_1, \ldots, \mathbf{x}_N\}$ where $\mathbf{x}_j \in \mathbb{R}^d$ are independent and identically distributed according to a robust mixture probability density function on $\mathbb{R}^d$ indexed by the set of parameters $\Theta = \{\theta, \psi, \beta\}$. For the data points $\mathcal{X}$, we have a set of N hidden inlier Bernoulli variables $\mathcal{O} = \{o_1, \ldots, o_N\}$ where $o_j \in \{0, 1\}$ denotes whether the data point $\mathbf{x}_j$ is an inlier or not with probability $\beta \in [0, 1]$. The inliers are distributed according to the parametric distribution $p(\mathbf{x}|\theta)$ and the outliers are distributed according to the parametric distribution $p(\mathbf{x}|\psi)$. The

robust mixture probability density function $p(\mathbf{x}|\Theta)$ can be written as

$$\begin{aligned} p(\mathbf{x}|\Theta) &= \sum_{m=0}^{1} p(\mathbf{x}, o=m|\Theta) \\ &= \sum_{m=0}^{1} p(o=m|\Theta)p(\mathbf{x}|o=m,\Theta) \\ &= p(o=0|\Theta)p(\mathbf{x}|o=0,\Theta) + p(o=1|\Theta)p(\mathbf{x}|o=1,\Theta) \\ &= p(o=0|\beta)p(\mathbf{x}|o=0,\psi) + p(o=1|\beta)p(\mathbf{x}|o=1,\theta) \\ &= (1-\beta)p(\mathbf{x}|o=0,\psi) + (\beta)p(\mathbf{x}|o=1,\theta). \end{aligned} \tag{4.1}$$

4.2.1 Maximum Likelihood Estimation

Given a data set $\mathcal{X} = \{\mathbf{x}_1, \ldots, \mathbf{x}_N\}$ of N independent and identically distributed (i.i.d.) random vectors corresponding to random samples from the marginal distribution $p(\mathbf{x}|\Theta) = \sum_{m=0}^{1} p(\mathbf{x}, o=m|\Theta)$, our objective is to find the maximum likelihood (ML) estimate $\hat{\Theta}$ of the parameters Θ. The ML estimation problem can be written in minimization form in (2.20) as

$$\hat{\Theta} = \arg\min_{\Theta} \ \ell_N(\Theta|\mathcal{X}) \tag{4.2}$$

where we used $\ell_N(\Theta|\mathcal{X}) = -\frac{1}{N}\sum_{j=1}^{N} \ell(\Theta|\mathbf{x}_j)$.

4.2.2 Expectation Maximization Algorithm

To estimate the parameters, we use the expectation maximization (EM) algorithm which consists of two steps called the E-step and the M-step, and uses a bound function $F(\mathcal{R}, \Theta)$ which is a function of the model parameters Θ and the set of introduced distributions $\mathcal{R} = \{r(o_1), \ldots, r(o_N)\}$ over the hidden inlier indicator variables $\mathcal{O}$. The EM algorithm and the derivation of the bound functions are discussed in detail in Sections 3.3.1 and 3.3.2. In the E-step of iteration t, the bound function $F(\mathcal{R}, \Theta^{t-1})$ is minimized over the introduced set of distributions $\mathcal{R}$ while holding the parameters, Θ^{t-1}, found in the previous iteration $t-1$ fixed

as

$$\mathcal{R}^t = \arg\min_{\mathcal{R}} F(\mathcal{R}, \Theta^{t-1}). \tag{4.3}$$

In the M-step of iteration t, the bound function $F(\mathcal{R}^t, \Theta)$ is minimized over the parameters Θ while holding the distributions, $\mathcal{R}^t$, found in the E-step fixed as

$$\Theta^t = \arg\min_{\Theta} F(\mathcal{R}^t, \Theta). \tag{4.4}$$

4.2.3 E-step

Following the ideas discussed in Section 3.3.3, we express the bound function $F(\mathcal{R}, \Theta)$ as

$$F(\mathcal{R}, \Theta) = KL_N(\mathcal{R}||p(\mathcal{O}|\mathcal{X}, \Theta)) + \ell_N(\Theta|\mathcal{X}) \tag{4.5}$$

where we defined

$$KL_N(\mathcal{R}||p(\mathcal{O}|\mathcal{X}, \Theta)) = \frac{1}{N}\sum_{j=1}^{N} KL(r(o_j)||p(o_j|\mathbf{x}_j, \Theta)). \tag{4.6}$$

In the E-step, we minimize the bound function $F(\mathcal{R}, \Theta)$ over the introduced set of distributions $\mathcal{R}$ while holding the parameters Θ fixed. Thus, we can write the corresponding optimization problem as

$$\mathcal{R} = \arg\min_{\mathcal{R}} KL_N(\mathcal{R}||p(\mathcal{O}|\mathcal{X}, \Theta)) + \ell_N(\Theta|\mathcal{X}) \tag{4.7}$$

As discussed in detail in Section 3.3.3, for the optimum solution the introduced distribution should be equal to the posterior distribution, i. e., $r(o_j) = p(o_j|\mathbf{x}_j, \Theta)$. To calculate the posterior distributions $p(o_j|\mathbf{x}_j, \Theta)$, we need to know the outlier distributions $p(\mathbf{x}_j|o_j = 0, \psi)$ and the outlier probabilities $1 - \beta$ which is proportional to the number of outliers.

4.2.4 Constrained E-step

We have seen that for the general case we need to know the outlier distributions $p(\mathbf{x}_j|o_j = 0, \psi)$ and the outlier probabilities $1 - \beta$ to calculate the posterior

distributions $p(o_j|\mathbf{x}_j, \Theta)$ for the E-step. However in practice it is hard to know the outlier distributions. In the proof of the Propositions 9 and 10 we will show that if we constrain the values that the introduced distributions can take to be binary, and assume that the likelihoods of the data points given they are outliers, $p(\mathbf{x}_j|o_j = 0, \psi)$, are equal, then an optimum solution of the constrained E-step can be calculated without any additional information about the outlier distributions $p(\mathbf{x}_j|o_j = 0, \psi)$. To make it easier to see, we expand the bound $F(\mathcal{R}, \Theta)$ as

$$
\begin{aligned}
F(\mathcal{R}, \Theta) &= KL_N(\mathcal{R}||p(\mathcal{O}|\mathcal{X}, \Theta)) + \ell_N(\Theta|\mathcal{X}) \\
&= \frac{1}{N}\sum_{j=1}^{N}\sum_{m=0}^{1} r(o_j = m) \log \frac{r(o_j = m)}{p(o_j = m|\mathbf{x}_j, \Theta)p(\mathbf{x}_j = m|\Theta)} \\
&= \frac{1}{N}\sum_{j=1}^{N}\sum_{m=0}^{1} r(o_j = m) \log \frac{r(o_j = m)}{p(\mathbf{x}_j|o_j = m, \Theta)p(o_j = m|\Theta)} \\
&= \frac{1}{N}\sum_{j=1}^{N}\sum_{m=0}^{1} r(o_j = m) \log r(o_j = m) \qquad (4.8) \\
&\quad - \frac{1}{N}\sum_{j=1}^{N}\sum_{m=0}^{1} r(o_j = m) \log p(\mathbf{x}_j|o_j = m, \Theta) \qquad (4.9) \\
&\quad - \frac{1}{N}\sum_{j=1}^{N}\sum_{m=0}^{1} r(o_j = m) \log p(o_j = m|\beta). \qquad (4.10)
\end{aligned}
$$

Proposition 9. *If the introduced distribution $\mathcal{R}$ can take only binary values, i.e., $r(o_j) \in \{0, 1\}$ for $j = 1, \ldots, N$, the number of inliers is a known fixed number $\tilde{N}$, i.e., $\sum_{j=1}^{N} r(o_j = 1) = \tilde{N}$.Furthermore, if the likelihoods of the data points given they are outliers are equal to a constant $\tilde{p}$, i.e., $p(\mathbf{x}_j|o_j = 0, \psi) = \tilde{p}$ for $j = 1, \ldots, N$, then setting $r(o_j = 1) = 1$ for the $\tilde{N}$ biggest $\log p(\mathbf{x}_j|o_j = 1, \theta)$ values and $r(o_j = 1) = 0$ for the rest corresponds to an optimum solution of the optimization problem in* (4.7).

Proof. We use Term1,Term2 and Term3 to address (4.8), (4.9) and (4.10), respectively. We can rewrite the Term1 using the relations $r(o_j = 0) = 1 - r(o_j = 1)$

for $j = 1, \ldots, N$ as

$$\text{Term1} = \frac{1}{N}\sum_{j=1}^{N}\big(1 - r(o_j = 1)\big)\log\big(1 - r(o_j = 1)\big) + r(o_j = 1)\log r(o_j = 1) \tag{4.11}$$

Binary constraints $r(o_j) \in \{0, 1\}$ make Term1 zero because $0 \log 0 = 1 \log 1 = 0$. Similarly, we can rewrite the Term3 using the relations $r(o_j = 0) = 1 - r(o_j = 1)$ for $j = 1, \ldots, N$ as

$$\begin{aligned}\text{Term3} &= -\frac{1}{N}\sum_{j=1}^{N}\big(1 - r(o_j = 1)\big)\log\big(1 - \beta\big) + r(o_j = 1)\log\beta \\ &= -\log\big(1 - \beta\big) - \frac{1}{N}\sum_{j=1}^{N} r(o_j = 1)\log\frac{\beta}{1 - \beta}\end{aligned} \tag{4.12}$$

Substituting $\beta = \frac{\tilde{N}}{N}$, $1 - \beta = \frac{N - \tilde{N}}{N}$ and $\sum_{j=1}^{N} r(o_j = 1) = \tilde{N}$ we have

$$\begin{aligned}\text{Term3} &= -\log\big(\frac{N - \tilde{N}}{N}\big) - \frac{\tilde{N}}{N}\log\frac{\tilde{N}}{N - \tilde{N}} \\ &= -\frac{N - \tilde{N}}{N}\log(\frac{N - \tilde{N}}{N}) - \frac{\tilde{N}}{N}\log(\frac{\tilde{N}}{N})\end{aligned} \tag{4.13}$$

Hence Term3 is constant. Similarly, we can rewrite Term2 using the relations $r(o_j = 0) = 1 - r(o_j = 1)$ for $j = 1, \ldots, N$ as

$$\begin{aligned}\text{Term2} &= -\frac{1}{N}\sum_{j=1}^{N}\big(1 - r(o_j = 1)\big)\log p(\mathbf{x}_j|o_j = 0, \psi) + r(o_j = 1)\log p(\mathbf{x}_j|o_j = 1, \theta) \\ &= -\frac{1}{N}\sum_{j=1}^{N}\log p(\mathbf{x}_j|o_j = 0, \psi) - \frac{1}{N}\sum_{j=1}^{N} r(o_j = 1)\log\frac{p(\mathbf{x}_j|o_j = 1, \theta)}{p(\mathbf{x}_j|o_j = 0, \psi)}\end{aligned} \tag{4.14}$$

Substituting $p(\mathbf{x}_j|o_j = 0, \psi) = \tilde{p}$ for $j = 1, \ldots, N$, we have

$$\text{Term2} = -\frac{1}{N}\sum_{j=1}^{N}\log\tilde{p} - \frac{1}{N}\sum_{j=1}^{N} r(o_j = 1)\log\frac{p(\mathbf{x}_j|o_j = 1, \theta)}{\tilde{p}} \tag{4.15}$$

Sum of all terms (Term1,Term2 and Term3) is an affine function of $r(o_j = 1)$ for $j = 1, \ldots, N$. Ignoring the constant parts, we can find the solution to the

problem in 4.7 by solving

$$\begin{aligned} \text{minimize } & -\frac{1}{N}\sum_{j=1}^{N} r(o_j=1)\log p(\mathbf{x}_j|o_j=1,\theta) \\ \text{subject to } & \sum_{j=1}^{N} r(o_j=1)=\tilde{N} \\ & r(o_j=1)\in\{0,1\} \text{ for } j=1,\ldots,N \end{aligned} \tag{4.16}$$

The objective is linear in $r(o_j = 1)$'s and they are constrained to sum to $\tilde{N}$. Thus setting $r(o_j = 1) = 1$ for the $\tilde{N}$ biggest $\log p(\mathbf{x}_j|o_j = 1, \theta)$ values and $r(o_j = 1) = 0$ for the rest corresponds to an optimum solution of the optimization problem in 4.7. For a more detailed proof based on the linear programming relaxation of the optimization problem in (4.16), we refer to [36]. □

Proposition 10. *If the introduced distribution $\mathcal{R}$ can take only binary values, i.e., $r(o_j) \in \{0, 1\}$ for $j = 1, \ldots, N$, and the likelihoods of the data points given they are outliers are equal to a constant $\tilde{p}$, i.e., $p(\mathbf{x}_j|o_j = 0, \psi) = \tilde{p}$ for $j = 1, \ldots, N$, then setting $r(o_j = 1) = 1$ for the positive $\log \frac{(\beta)p(\mathbf{x}_j|o_j=1,\theta)}{(1-\beta)\tilde{p}}$ values and $r(o_j = 1) = 0$ for the rest corresponds to an optimum solution of the optimization problem in 4.7.*

Proof. We use Term1, Term2 and Term3 to address 4.8, 4.9 and 4.10, respectively. Binary constraints $r(o_j) \in \{0, 1\}$ make Term1 zero because $0 \log 0 = 1 \log 1 = 0$. For the Term3, we have

$$\text{Term3} = -\log\big(1-\beta\big) - \frac{1}{N}\sum_{j=1}^{N} r(o_j=1)\log\frac{\beta}{1-\beta} \tag{4.17}$$

For the Term2, we have

$$\text{Term2} = -\frac{1}{N}\sum_{j=1}^{N}\log\tilde{p} - \frac{1}{N}\sum_{j=1}^{N} r(o_j=1)\log\frac{p(\mathbf{x}_j|o_j=1,\theta)}{\tilde{p}} \tag{4.18}$$

Sum of all terms (Term1,Term2 and Term3) is an affine function of $r(o_j = 1)$ for $j = 1, \ldots, N$. Ignoring the constant parts, we can find the solution to the

problem in 4.7 by solving

$$\begin{aligned} \text{minimize } & -\frac{1}{N}\sum_{j=1}^{N} r(o_j=1)\log\frac{(\beta)p(\mathbf{x}_j|o_j=1,\theta)}{(1-\beta)\tilde{p}} \\ \text{subject to } & \sum_{j=1}^{N} r(o_j=1) \leq N \\ & r(o_j=1)\in\{0,1\} \text{ for } j=1,\ldots,N \end{aligned} \tag{4.19}$$

The objective is linear in $r(o_j = 1)$'s. Furthermore we are minimizing the sum of the negative log-likelihood ratios which is equivalent to the maximization of the sum of the log-likelihood ratios. Since only positive values increases the sum we only want to have positive log-likelihood ratios. Thus setting $r(o_j = 1) = 1$ for the positive $\log \frac{(\beta)p(\mathbf{x}_j|o_j=1,\theta)}{(1-\beta)\tilde{p}}$ values and $r(o_j = 1) = 0$ for the rest corresponds to an optimum solution of the optimization problem in 4.7. For a more detailed proof based on the linear programming relaxation of the optimization problem in (4.19), we refer to [36]. □

4.3 Robust Gaussian Mixture Models

4.3.1 Problem Definition

We are given a set of N data points $\mathcal{X} = \{\mathbf{x}_1, \ldots, \mathbf{x}_N\}$ where $\mathbf{x}_j \in \mathbb{R}^d$ are independent and distributed according to a robust Gaussian mixture probability density function on $\mathbb{R}^d$ indexed by the set of parameters $\Theta = \{\theta_{in}, \theta_{out}, \theta_r\}$. For the data points $\mathcal{X}$, we have a set of N hidden inlier Bernoulli variables $\mathcal{O} = \{o_1, \ldots, o_N\}$ where $o_j \in \{0, 1\}$ denotes whether the data point $\mathbf{x}_j$ is an inlier denoted by $o_j = 1$ or not with probability $\frac{\exp\theta_r}{1+\exp\theta_r} \in [0, 1]$. The outliers are assumed to be equally likely where $\log p(\mathbf{x}_j|o_j = 0, \theta_{out}) = \theta_{out}$. The inliers are distributed according to the Gaussian mixture distribution $p(\mathbf{x}_j|o_j = 1, \theta_{in})$ with K components parametrized by $\theta_{in} = \{\eta, m_1, S_1, \ldots, m_K, S_K\}$ where $\eta \in \mathbb{R}^{K-1}$, $m_k \in \mathbb{R}^d$, $S_k \in \mathcal{S}_+^d$ for $k = 1, \ldots, K$ are the information parameters. Hence, for the data points $\mathcal{X}$, we also have a set of N hidden multinomial variables $\mathcal{Y} =$

$\{y_1, \ldots, y_N\}$ where $y_j \in \{1, \ldots, K\}$ denotes the index of the Gaussian components for the data point $\mathbf{x}_j$. The robust mixture probability density function $p(\mathbf{x}|\Theta)$ can be written as

$$p(\mathbf{x}|\Theta) = (\frac{1}{1+\exp\theta_r})p(\mathbf{x}|o=0,\theta_{out}) + (\frac{\exp\theta_r}{1+\exp\theta_r})\sum_{k=1}^{K} p(\mathbf{x}, y=k|o=1,\theta_{in}) \tag{4.20}$$

Moreover, we have a data set $\mathcal{X}_{in} = \{\mathbf{x}_{in,1}, \ldots, \mathbf{x}_{in,N_{in}}\}$ of N_{in} data points known to be inliers and a data set $\mathcal{X}_{out} = \{\mathbf{x}_{out,1}, \ldots, \mathbf{x}_{out,N_{out}}\}$ of N_{out} data points known to be outliers. We form affine inequality constraints on the parameters to ensure that the inlier data points $\mathcal{X}_{in}$ have higher and the outlier data points $\mathcal{X}_{out}$ have lower log-likelihood values than the threshold value $\theta_{out} - \theta_r$. For any inlier data point $\mathbf{x}_{in,i}$, we would like to have

$$\log\big(p(\mathbf{x}_{in,i}|o_{in,i}=1,\theta_{in})p(o_{in,i}=1)\big) > \log\big(p(\mathbf{x}_{in,i}|o_{in,i}=0)p(o_{in,i}=0)\big)$$

$$\log\big(p(\mathbf{x}_{in,i}|o_{in,i}=1,\theta_{in})\big) > \log\big(p(\mathbf{x}_{in,i}|o_{in,i}=0)\big) - \log\big(\frac{p(o_{in,i}=1)}{p(o_{in,i}=0)}\big)$$

which is equal to

$$\log\big(\sum_{k=1}^{K} p(\mathbf{x}_{in,i}, y_{in,i}=k|o_{in,i}=1,\theta_{in})\big) > \theta_{out} - \theta_r \tag{4.21}$$

Similarly for any outlier data point $\mathbf{x}_{out,i}$, we would like to have

$$\log\big(\sum_{k=1}^{K} p(\mathbf{x}_{out,i}, y_{out,i}=k|o_{out,i}=1,\theta_{in})\big) \leq \theta_{out} - \theta_r \tag{4.22}$$

Notice that for any data point $\mathbf{x}_j$, we have

$$p(y_j=k|\mathbf{x}_j, o_j=1,\theta_{in}) = \frac{p(\mathbf{x}_j, y_j=k|o_j=1,\theta_{in})}{\sum_{m=1}^{K} p(\mathbf{x}_j, y_j=m|o_j=1,\theta_{in})},\ k=1,\ldots,K \tag{4.23}$$

Hence we have K equalities for $\log\big(\sum_{m=1}^{K} p(\mathbf{x}_j, y_j=m|o_j=1,\theta_{in})\big)$ as

$$\begin{aligned}&\log\big(\sum_{m=1}^{K} p(\mathbf{x}_j, y_j=m|o_j=1,\theta_{in})\big)\\ &= \log p(\mathbf{x}_j, y_j=k|o_j=1,\theta_{in}) - \log p(y_j=k|\mathbf{x}_j, o_j=1,\theta_{in}),\ k=1,\ldots,K\end{aligned} \tag{4.24}$$

When we expand $\log p(\mathbf{x}_j, y_j = k | o_j = 1, \theta_{in})$, we get $tr\big(S_k(-\frac{1}{2}\mathbf{x}_j\mathbf{x}_j^T)\big) + m_k^T\mathbf{x}_j$ plus some additional terms. To handle both inliers and outliers, we need to have both greater than and less than equal to constraints on the parameters. For these constraints to be convex, they have be affine in the parameters. To form affine constraints, we use K additional variables $c_1, \ldots, c_K$, $c_k \in \mathbb{R}$ for $k = 1, \ldots, K$ in place of those additional terms. The inequality constraints denoted by $\mathcal{C}$ are as

$$\begin{aligned} &tr\big(S_k(-\frac{1}{2}\mathbf{x}_{in,i}\mathbf{x}_{in,i}^T)\big) + m_k^T\mathbf{x}_{in,i} + c_k \\ &- \log p(y_{in,i} = k | \mathbf{x}_{in,i}, o_{in,i} = 1, \theta_{in}) > \theta_{out} - \theta_r,\ k = 1, \ldots, K,\ i = 1, \ldots, N_{in} \end{aligned} \tag{4.25}$$

$$\begin{aligned} &tr\big(S_k(-\frac{1}{2}\mathbf{x}_{out,i}\mathbf{x}_{out,i}^T)\big) + m_k^T\mathbf{x}_{out,i} + c_k \\ &- \log p(y_{out,i} = k | \mathbf{x}_{out,i}, o_{out,i} = 1, \theta_{in}) \leq \theta_{out} - \theta_r,\ k = 1, \ldots, K,\ i = 1, \ldots, N_{out} \end{aligned} \tag{4.26}$$

Our objective is to find the maximum likelihood (ML) estimate $\hat{\Theta}$ of the parameters Θ. The ML estimation problem can be written in minimization form in 2.20 as

$$\begin{aligned} \text{minimize} \quad & -\frac{1}{N}\sum_{j=1}^{N} \ell(\Theta | \mathbf{x}_j) \\ \text{subject to} \quad & (\Theta, c_1, \ldots, c_K) \in \mathcal{C} \end{aligned} \tag{4.27}$$

where $\Theta, c_1, \ldots, c_K$ are the optimization variables and $\mathcal{X}_{in}$ and $\mathcal{X}_{out}$ are the optimization parameters used in the constraint set $\mathcal{C}$ which are assumed be known.

4.3.2 Expectation Maximization Algorithm

We use the EM algorithm to solve the maximum likelihood estimation problem in (4.27). In the E-step, we compute the posterior distributions.

E-step

$$q^t(y_j) = \frac{p(\mathbf{x}_j, y_j | o_j = 1, \theta_{in}^{t-1})}{\sum_{i=1}^{K} p(\mathbf{x}_j, y_j = i | o_j = 1, \theta_{in}^{t-1})} \text{ for } j = 1, \ldots, N \tag{4.28}$$

$$r^t(o_j = 1) = \begin{cases} 1, \text{ if } E_{q^t(y_j)}[\log p(\mathbf{x}_j, y_j | o_j = 1, \theta_{in}^{t-1})] + H(q^t(y_j)) > \theta_{out}^{t-1} - \theta_r^{t-1} \\ 0, \text{ o.w.} \end{cases}$$
$$\text{for } j = 1, \ldots, N \tag{4.29}$$

In the M-step we compute the information parameters $\theta_{in} = \{\eta, m_1, S_1, \ldots, m_K, S_K\}$ and the constant outliers log-likelihood value denoted by θ_{out} and log-ratio of the outlier probabilities denoted by θ_r subject to the affine inequality constraints ensuring that the inlier data points have higher and the the outlier data points have lower log-likelihood values than the threshold value $\theta_{out} - \theta_r$ by solving the following optimization problem

M-step

$$\begin{aligned} \text{minimize} \quad & \log(1 + \sum_{k=1}^{K-1} \exp \eta_k) + \sum_{k=1}^{K} \alpha_{sk} \big(-\frac{1}{2} \log |S_k| + \frac{1}{2} m_k^T S_k^{-1} m_k + \frac{d}{2} \log 2\pi \big) \\ & - \sum_{k=1}^{K-1} \eta_k \alpha_{sk} - \sum_{k=1}^{K} \alpha_{sk} \big(m_k^T \mu_{sk} - \frac{1}{2} tr(S_k(\Sigma_{sk} + \mu_{sk}\mu_{sk}^T)) \big) \\ & + \log(1 + \exp \theta_r) - \theta_r \beta \end{aligned}$$

subject to

$$tr\big(S_k(-\frac{1}{2}\mathbf{x}_{in,i}\mathbf{x}_{in,i}^T)\big) + m_k^T \mathbf{x}_{in,i} + c_k - \log q^t(y_{in,i} = k) \geq \theta_{out} - \theta_r,$$
$$\text{for } k = 1, \ldots, K, \ i = 1, \ldots, N_{in} \tag{4.30}$$

$$tr\big(S_k(-\frac{1}{2}\mathbf{x}_{out,i}\mathbf{x}_{out,i}^T)\big) + m_k^T \mathbf{x}_{out,i} + c_k - \log q^t(y_{out,i} = k) \leq \theta_{out} - \theta_r,$$
$$\text{for } k = 1, \ldots, K, \ i = 1, \ldots, N_{out} \tag{4.31}$$

where $\eta, m_1, S_1, \ldots, m_K, S_K, c_1, \ldots, c_K, \theta_r, \theta_{out}$ are the optimization variables. For each k, the inequality constraints in 4.30 are affine in the optimization variables $m_k, S_k, c_k, \theta_{out}, \theta_r$ ensuring that the inlier data points $\mathbf{x}_{in,i}$ for $i = 1, \ldots, N_{in}$ have higher log-likelihood values than the threshold value $\theta_{out} - \theta_r$. Similarly, for each k, the inequality constraints in 4.31 are affine in the optimization variables

$m_k, S_k, c_k, \theta_{out}, \theta_r$ ensuring that the outlier data points $\mathbf{x}_{out,i}$ for $i = 1, \ldots, N_{out}$ have lower log-likelihood values than the threshold value $\theta_{out} - \theta_r$. The expected empirical probabilities

$$\alpha_{sk} = \frac{1}{N} \sum_{j=1}^{N} r^t(o_j = 1) q^t(y_j = k),\ k = 1, \ldots, K$$

the expected empirical means

$$\mu_{sk} = \frac{1}{\alpha_{sk} N} \sum_{j=1}^{N} r^t(o_j = 1) q^t(y_j = k) \mathbf{x}_j,\ k = 1, \ldots, K$$

the expected empirical covariance matrices

$$\Sigma_{sk} = \frac{1}{\alpha_{sk} N} \sum_{j=1}^{N} r^t(o_j = 1) q^t(y_j = k) \mathbf{x}_j \mathbf{x}_j^T - \mu_{sk} \mu_{sk}^T,\ k = 1, \ldots, K$$

the expected empirical inlier probabilities

$$\beta = \frac{1}{N} \sum_{j=1}^{N} r^t(o_j = 1)$$

the posterior probabilities for the inliers $\mathcal{X}_{in}$ and the outliers $\mathcal{X}_{out}$ denoted by

$$q^t(y_{in,i}) = \frac{p(\mathbf{x}_{in,i}, y_{in,i} | o_{in,i} = 1, \theta_{in}^{t-1})}{\sum_{k=1}^{K} p(\mathbf{x}_{in,i}, y_{in,i} = k | o_{in,i} = 1, \theta_{in}^{t-1})},$$

and

$$q^t(y_{out,i}) = \frac{p(\mathbf{x}_{out,i}, y_{out,i} | o_{out,i} = 1, \theta_{in}^{t-1})}{\sum_{k=1}^{K} p(\mathbf{x}_{out,i}, y_{out,i} = k | o_{out,i} = 1, \theta_{in}^{t-1})}$$

respectively, are the problem parameters which were calculated apriori after the E-step.

4.4 Experiments

To illustrate the capabilities of the proposed model, we consider a simple example for the robust constrained GMM estimation problem using a two dimensional synthetic data set. We generated a random GMM with $K = 3$ Gaussian components. Then, we sampled 300 data points from the generated GMM and 100 data points from a uniform distribution $[0, 100]^2$.

We have considered four cases. In the first case, we used the standard EM algorithm on the whole data set consisting of 400 data points. For the rest of the three cases, we used the proposed EM algorithm for robust constrained GMM. We have selected two inlier data points and four outlier data points. In all of the three cases, two inliers data points are kept the same. For the second case, we have selected four outlier data points at the corners which are assumed to be the least informative. For the third and the fourth cases, we have selected four outlier data points which are closer to the Gaussians in the reference GMM by eyeballing the data.

In all cases, EM algorithms were initialized the same way. Following the common practice in the literature, the initial mean vector for each component was set to a randomly selected data point. The initial covariance matrices and the initial mixture weights were calculated from the probabilistic assignment of the data points to the Gaussian components with the initial mean vectors and the identity covariance matrices. 50 different initializations were obtained this way, and the EM algorithms were run for each initial configuration until convergence for maximum 500 iterations. The final result of each EM run was selected as the parameters corresponding to the best out of 50 runs having the highest log-likelihood.

Fig. 4.1 shows 300 data points generated from the reference GMM. All 300 data points are marked in blue. The reference Gaussians used to generate the 300 data points are overlayed as red ellipses drawn at three standard deviations.

Fig. 4.2 shows 100 data points generated from a uniform distribution. All 100 data points are marked in blue.

Fig. 4.3 shows 400 data points used as the training data set. All 400 data points in the training data set are marked in blue. The reference Gaussians used to generate the 300 data points are overlayed as red ellipses drawn at three standard deviations.

Fig. 4.4 shows 400 data points used as the training data set. All 400 data points in the training data set are marked in blue. The Gaussians obtained using

the best out of 50 runs of the standard EM algorithm are overlayed as red ellipses drawn at three standard deviations.

Fig. 4.5 shows 300 data points generated from the reference GMM and 100 data points generated from a uniform distribution in $[0, 100]^2$. Two data points at coordinates $(24.8, 63.2)$ and $(44.1, 24.0)$ are selected as inliers and are marked in green. Four data points at coordinates $(2.9, 98.2)$, $(1.0, 7.5)$, $(95.7, 1.7)$ and $(92.4, 98.2)$ are selected as outliers and are marked in white. The reference Gaussians used to generate the 300 data points are overlayed as red ellipses drawn at three standard deviations.

Fig. 4.6 shows the detected inliers, outliers and the resulting Gaussians obtained using the proposed EM algorithm for the constrained robust GMMs. 323 data points detected as inliers are marked in green. 77 data points detected as outliers are marked in white. The resulting Gaussians obtained using the best out of 50 runs of the proposed EM algorithm for the constrained robust GMMs are overlayed as red ellipses drawn at three standard deviations.

Fig. 4.7 shows 300 data points generated from the reference GMM and 100 data points generated from a uniform distribution in $[0, 100]^2$. Two data points at coordinates $(24.8, 63.2)$ and $(44.1, 24.0)$ are selected as inliers and are marked in green. Four data points at coordinates $(93.1, 41.55)$, $(4.1, 39.7)$, $(68.2, 20.9)$ and $(20.7, 74.2)$ are selected as outliers and are marked in white. The reference Gaussians used to generate the 300 data points are overlayed as red ellipses drawn at three standard deviations.

Fig. 4.8 shows the detected inliers, outliers and the resulting Gaussians obtained using the proposed EM algorithm for the constrained robust GMMs. 318 data points detected as inliers are marked in green. 82 data points detected as outliers are marked in white. The resulting Gaussians obtained using the best out of 50 runs of the proposed EM algorithm for the constrained robust GMMs are overlayed as red ellipses drawn at three standard deviations.

Fig. 4.9 shows 300 data points generated from the reference GMM and 100 data points generated from a uniform distribution in $[0, 100]^2$. Two data points

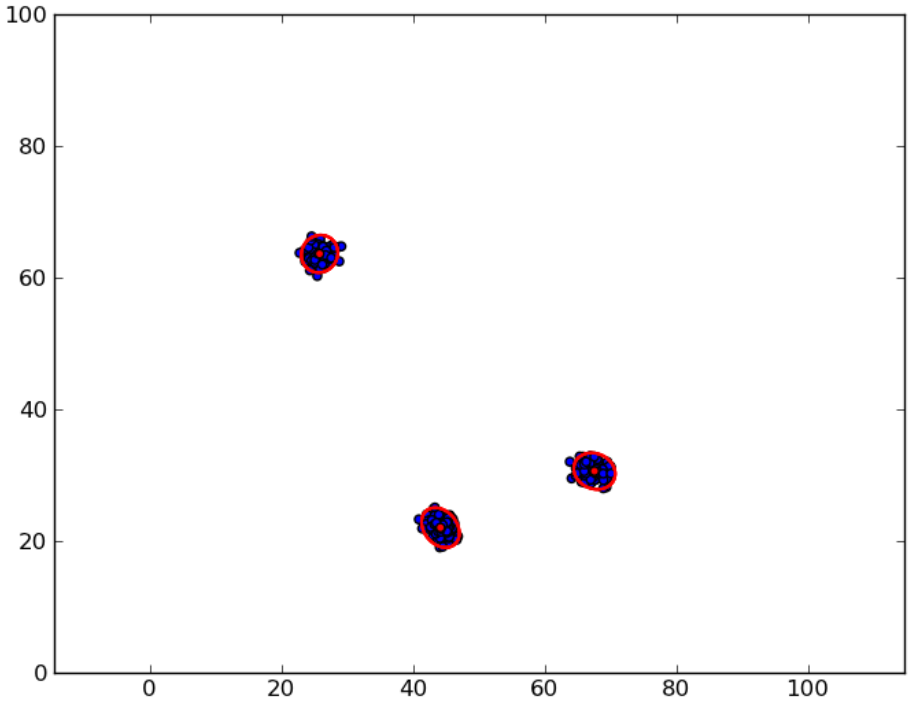


Figure 4.1: 300 data points sampled from a GMM are marked in blue. The reference Gaussians used to generate the data points are overlayed as red ellipses drawn at three standard deviations.

at coordinates $(24.8, 63.2)$ and $(44.1, 24.0)$ are selected as inliers and are marked in green. Four data points at coordinates $(88.3, 18.1)$, $(91.8, 7.3)$, $(45.7, 32.5)$ and $(31.6, 40.9)$ are selected as outliers and are marked in white. The reference Gaussians used to generate the 300 data points are overlayed as red ellipses drawn at three standard deviations.

Fig. 4.10 shows the detected inliers, outliers and the resulting Gaussians obtained using the proposed EM algorithm for the constrained robust GMMs. 310 data points detected as inliers are marked in green. 90 data points detected as outliers are marked in white. The resulting Gaussians obtained using the best out of 50 runs of the proposed EM algorithm for the constrained robust GMMs are overlayed as red ellipses drawn at three standard deviations.

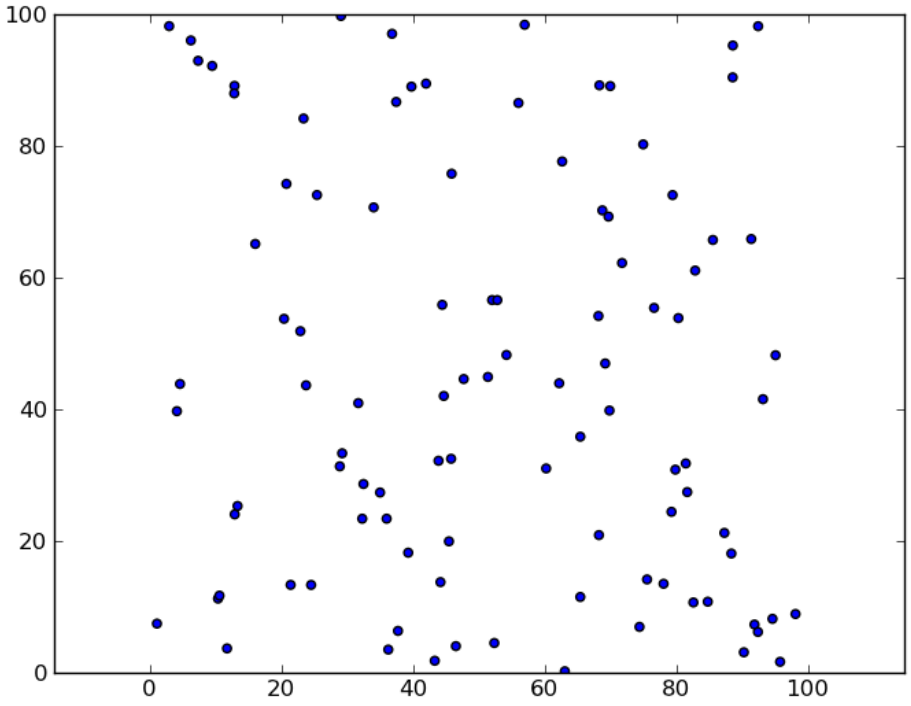


Figure 4.2: 100 data points corresponding to samples from a uniform distribution $[0, 100]^2$ are marked in blue.

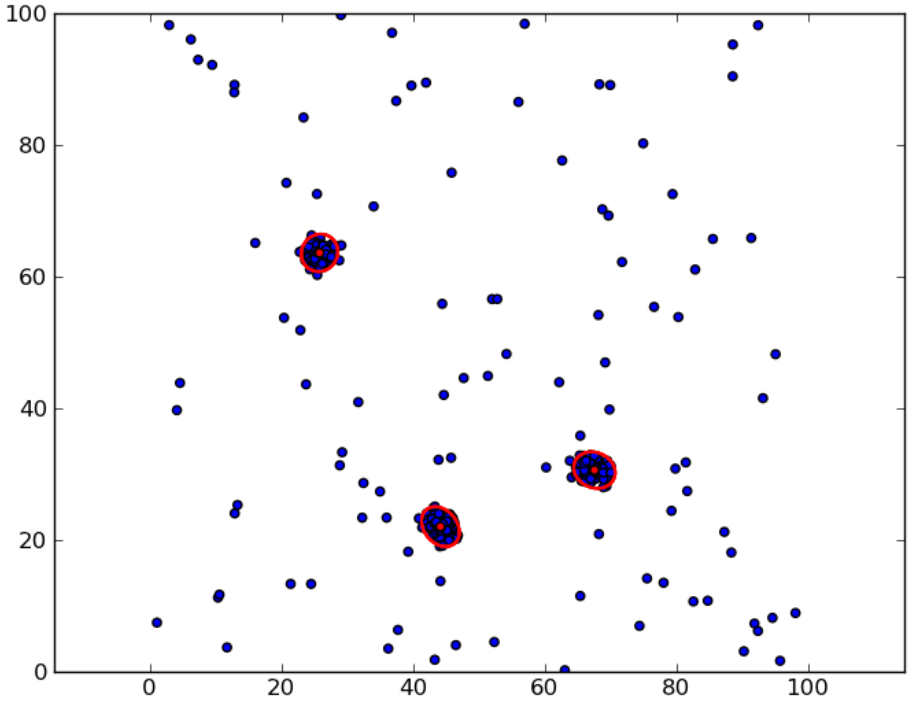


Figure 4.3: 400 data points in the training data set are marked in blue. The reference Gaussians are overlayed as red ellipses drawn at three standard deviations.

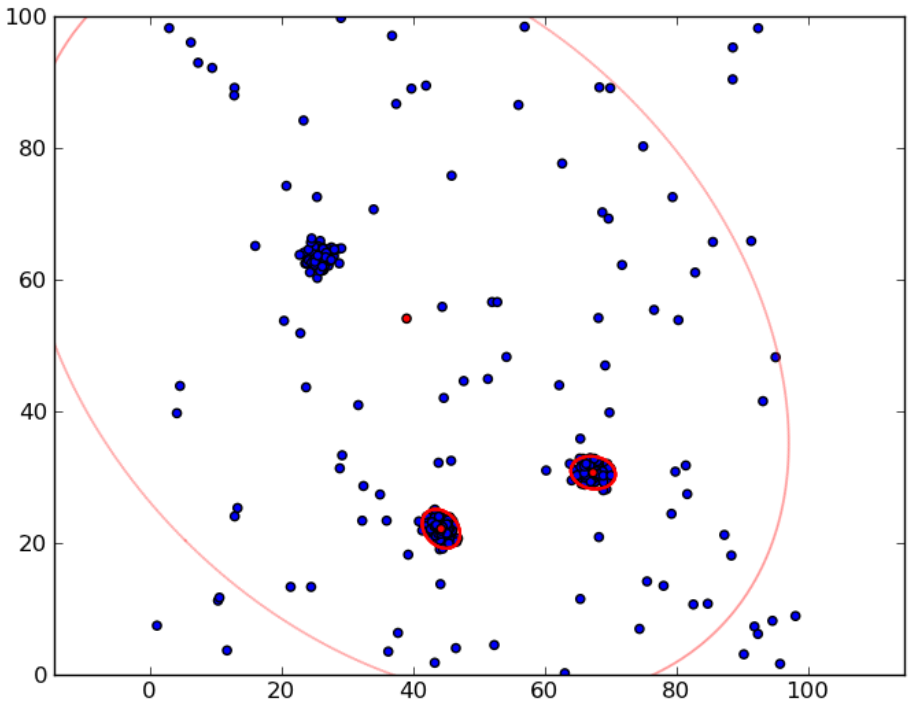


Figure 4.4: 400 data points in the training data set are marked in blue. The resulting Gaussians obtained using the best out of 50 runs of the standard EM algorithm are overlayed as red ellipses drawn at three standard deviations.

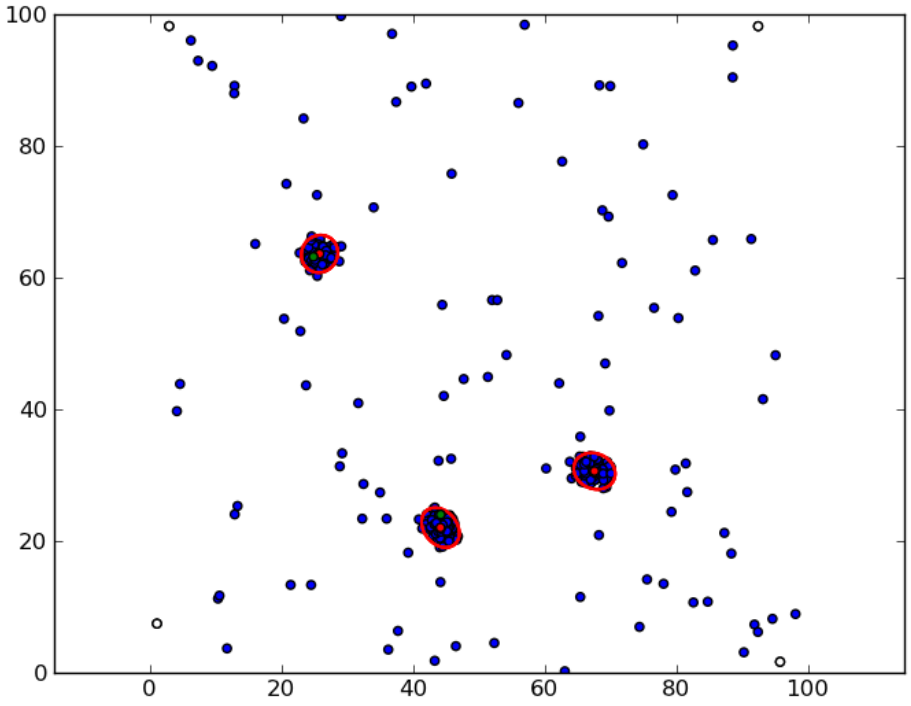


Figure 4.5: Two data points at coordinates $(24.8, 63.2)$ and $(44.1, 24.0)$ selected as inliers are marked in green. Four data points at coordinates $(2.9, 98.2)$, $(1.0, 7.5)$, $(95.7, 1.7)$ and $(92.4, 98.2)$ selected as outliers are marked in white. The rest of the data points in the data set is marked in blue. The reference Gaussians are overlayed as red ellipses drawn at three standard deviations.

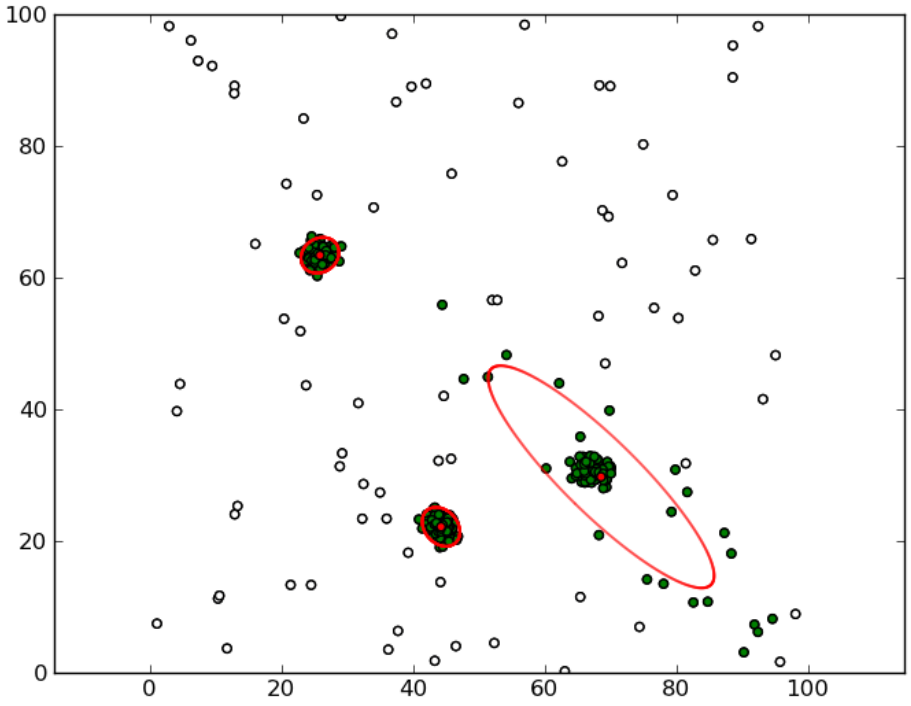


Figure 4.6: 323 data points detected as inliers are marked in green. 77 data points detected as outliers are marked in white. The resulting Gaussians obtained using the best out of 50 runs of the proposed EM algorithm for the constrained robust GMMs are overlayed as red ellipses drawn at three standard deviations.

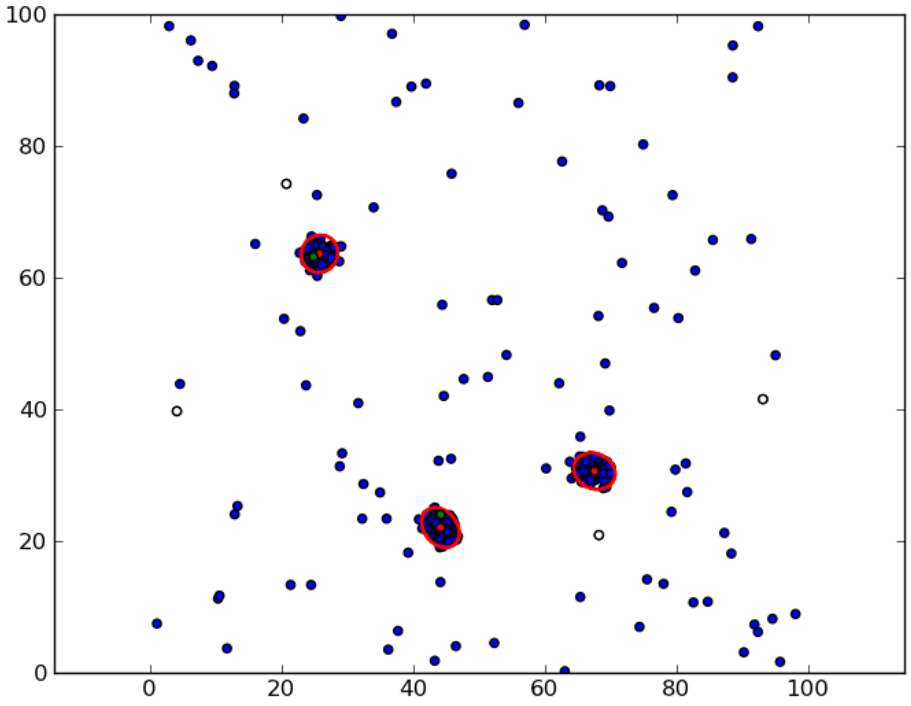


Figure 4.7: Two data points at coordinates $(24.8, 63.2)$ and $(44.1, 24.0)$ selected as inliers are marked in green. Four data points at coordinates $(93.1, 41.55)$, $(4.1, 39.7)$, $(68.2, 20.9)$ and $(20.7, 74.2)$ selected as outliers are marked in white. The rest of the data points in the data set is marked in blue. The reference Gaussians are overlayed as red ellipses drawn at three standard deviations.

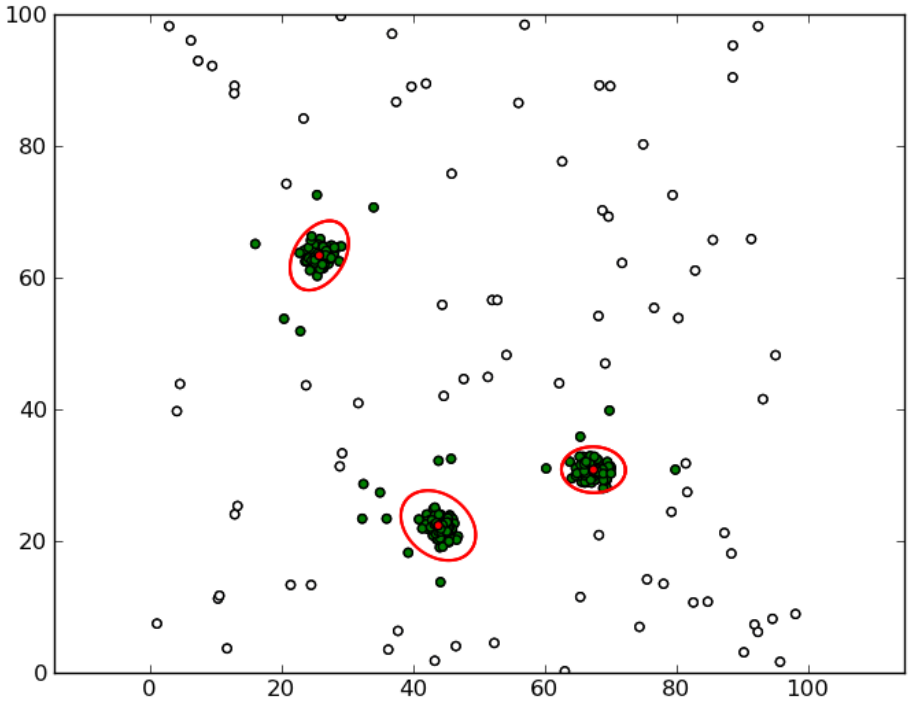


Figure 4.8: 318 data points detected as inliers are marked in green. 82 data points detected as outliers are marked in white. The resulting Gaussians obtained using the best out of 50 runs of the proposed EM algorithm for the constrained robust GMMs are overlayed as red ellipses drawn at three standard deviations.

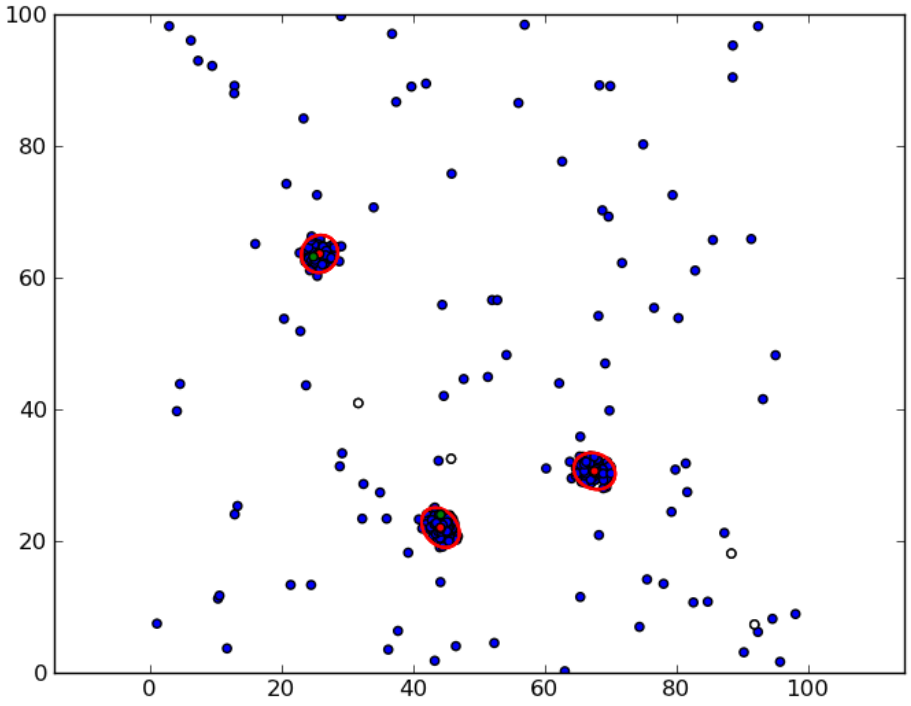


Figure 4.9: Two data points at coordinates $(24.8, 63.2)$ and $(44.1, 24.0)$ selected as inliers are marked in green. Four data points at coordinates $(88.3, 18.1)$, $(91.8, 7.3)$, $(45.7, 32.5)$ and $(31.6, 40.9)$ selected as outliers are marked in white. The rest of the data points in the data set is marked in blue. The reference Gaussians are overlayed as red ellipses drawn at three standard deviations.

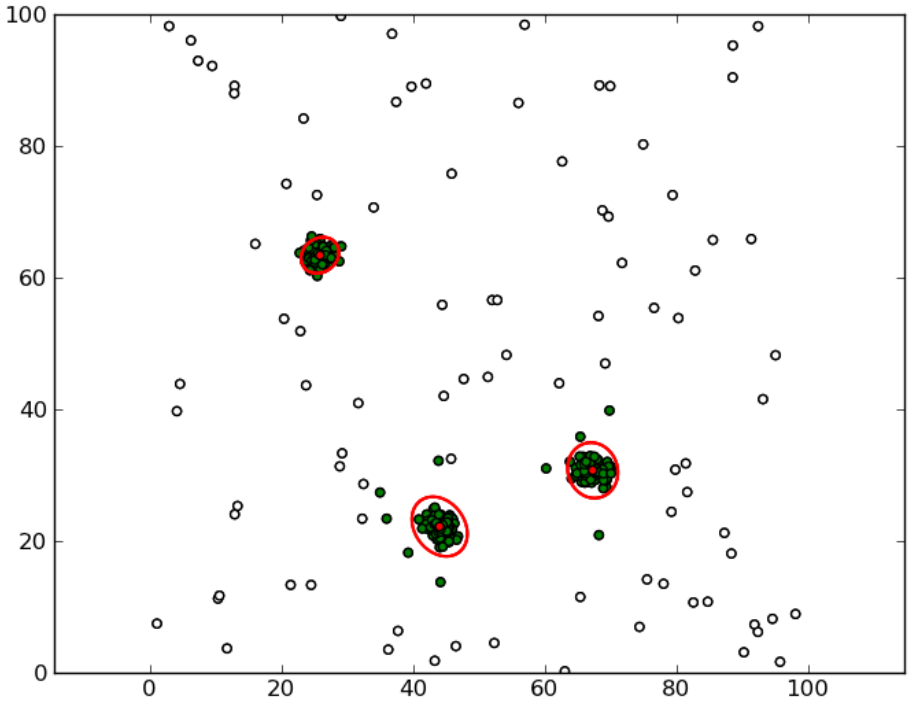


Figure 4.10: 310 data points detected as inliers are marked in green. 90 data points detected as outliers are marked in white. The resulting Gaussians obtained using the best out of 50 runs of the proposed EM algorithm for the constrained robust GMMs are overlayed as red ellipses drawn at three standard deviations.

4.5 Conclusions

In this Chapter, we studied the robust estimation of the Gaussian mixture models and provided a robust Gaussian mixture model as an application to the constrained Gaussian mixture model framework. We developed a robust Gaussian mixture model where inlier/outlier information available for few data points can be incorporated as convex constraints on the information parameters. We developed an EM algorithm to learn both the model parameters that are consistent with the available inlier/outlier information and the threshold value needed to determine the outliers. Furthermore, we have illustrated the capabilities of the proposed model on two dimensional synthetic data set.

Chapter 5

Maximum Likelihood Estimation of Gaussian Mixture Models Using Stochastic Search

5.1 Introduction

The conventional algorithm used to do the maximum likelihood estimation of Gaussian mixture model parameters is the expectation maximization (EM) algorithm. One of the main problems with the EM algorithm is that the algorithm converges to a local optimum. This is because the negative log-likehood function is not a convex function of the Gaussian mixture model parameters. Moreover, there is also the associated problem of initialization as it influences which local optima of the negative log-likelihood function is attained.

In this Chapter, a novel global search algorithm based on the expectation maximization and particle swarm optimization algorithms is presented to do the maximum likelihood estimation of the Gaussian mixture model parameters. Our major contributions in this Chapter are twofold. First, a novel parameterization for arbitrary covariance matrices that allow independent updating of individual

parameters while preserving the symmetry and the positive definiteness properties is presented. Second, an effective component matching technique to correct the problems due the existence of multiple candidate solutions which are equivalent under the permutations of the Gaussian mixture components is proposed. Experiments on synthetic and real-life data sets verifies the perfomance of the proposed algorithms.

The rest of the Chapter is organized as follows. Section 5.2 introduces the definition of the estimation problem. Section 5.3 gives the summary of the update equations for the expectation maximization algorithm in terms of the source parameters. Section 5.4 presents the details of the proposed covariance parameterization and the solution for the identifiability problem. Section 5.5 describes the proposed algorithm based on the expectation maximization and the particle swarm optimization algorithms. Section 5.6 presents the experiments and discussion using both synthetic and real data sets. Finally, Section 5.7 provides the conclusions of the Chapter.

5.2 Problem Definition

We consider a family of mixtures of K multivariate Gaussian distributions in $\mathbb{R}^d$ indexed by the source parameters $\Xi = \{\alpha_1, \mu_1, \Sigma_1, \ldots, \alpha_K, \mu_K, \Sigma_K\}$. Each $\{\mu_k, \Sigma_k\}$ represents the parameters of the k'th Gaussian distribution $p(\mathbf{x}|\mu_k, \Sigma_k)$ such that $\mu_k \in \mathbb{R}^d$ and $\Sigma_k \in \mathcal{S}_{++}^d$ are the means and the covariance matrices, respectively, for $k = 1, \ldots, K$. Mixing probabilities $\alpha_k \in [0, 1]$ are constrained to sum up to 1, i.e., $\sum_{k=1}^{K} \alpha_k = 1$. Given a set of N data points $\mathcal{X} = \{\mathbf{x}_1, \ldots, \mathbf{x}_N\}$ where $\mathbf{x}_j \in \mathbb{R}^d$ are independent and identically distributed (i.i.d.) according to the mixture probability density function $p(\mathbf{x}|\Xi) = \sum_{k=1}^{K} \alpha_k p(\mathbf{x}|\mu_k, \Sigma_k)$, the objective is to obtain the maximum likelihood estimate $\hat{\Xi}$ by finding the parameters that minimize the negative log-likelihood function

$$-\frac{1}{N} \sum_{j=1}^{N} \log \left(\sum_{k=1}^{K} \alpha_k p(\mathbf{x}_j|\mu_k, \Sigma_k) \right). \tag{5.1}$$

The negative log-likelihood function is not a convex function of the Gaussian mixture parameters. The common practice for reaching a local optimum of the negative log-likelihood function is to use the expectation-maximization (EM) algorithm.

5.3 Expectation Maximization Algorithm

For completeness we briefly present the update equations for the expectation maximization algorithm in terms of the source parameters in the this Section. Details of the expectation maximization algorithm can be found in Chapter 3.
E-step

$$q(y_j = k)^{(t)} = p(y_j = k|\mathbf{x}_j, \Xi^{(t)}) = \frac{\alpha_k^{(t)} p(\mathbf{x}_j|\mu_k^{(t)}, \Sigma_k^{(t)})}{\sum_{i=1}^{K} \alpha_i^{(t)} p(\mathbf{x}_j|\mu_i^{(t)}, \Sigma_i^{(t)})} \tag{5.2}$$

M-step

$$\alpha_k^{(t+1)} = \frac{1}{N} \sum_{j=1}^{N} q(y_j = k)^{(t)} \tag{5.3}$$

$$\mu_k^{(t+1)} = \frac{1}{\alpha_k^{(t+1)} N} \sum_{j=1}^{N} q(y_j = k)^{(t)} \mathbf{x}_j \tag{5.4}$$

$$\Sigma_k^{(t+1)} = \frac{1}{\alpha_k^{(t+1)} N} \sum_{j=1}^{N} \left(q(y_j = k)^{(t)} \mathbf{x}_j \mathbf{x}_j^T \right) - \mu_k^{(t+1)} (\mu_k^{(t+1)})^T \tag{5.5}$$

where t indicates the iteration number.

5.4 Stochastic Search

The EM algorithm coverges to a local optimum. To overcome this problem, the common practice is to use multiple random initializations to find different local optima, and to use the result corresponding to the highest log-likelihood value. This method can be viewed as a simple stochastic global search algorithm. However, even with some heuristics that have been proposed to guide the initialization,

this approach is usually far from providing an acceptable solution because there is no mechanism that can measure how different these multiple initializations are from each other. Furthermore, for relatively more complex data sets for which the likelihood function may contain a large number of local optima, the results for a large number of independent EM runs can still be unsatisfactory because these multiple initializations do not have a guarantee of a sufficient coverage of the solution space.

As discussed in Chapter 1, an alternative is to use population-based stochastic search algorithms where different candidate solutions are allowed to interact with each other. The interactions in the commonly used GA, DE, and PSO algorithms are typically implemented using operations such as randomized selection, swapping, addition, and perturbation of the individual parameters of the candidate solutions. For example, the crossover operation in GA and DE randomly selects some parts of two candidate solutions to create a new candidate solution during the reproduction of the population. Similarly, the mutation operation in GA and DE and the update operation in PSO perturb an existing candidate solution using a vector that is created using some combination of random numbers and other candidate solutions.

However, the continuation of the iterations that search for better candidate solutions assume that the parameters remain valid. The validity and boundedness of the mean vectors are relatively easy to implement but direct use of covariance matrices introduce problems. For example, one might consider to use $d(d+1)/2$ potentially different entries of a real symmetric $d \times d$ covariance matrix as a direct parameterization of the covariance matrix. Although this ensures the symmetry property, it cannot guarantee the positive definiteness where arbitrary modifications of these entries may produce non-positive definite matrices. This is illustrated in Table 5.1 where a new covariance matrix is constructed from three valid covariance matrices in a simple arithmetic operation. Even though the input matrices are positive definite, the output matrix is often not positive definite for increasing dimensions. Another possible parameterization is to use Cholesky factorization but the resulting parameters are unbounded (real numbers in the $(-\infty, \infty)$ range). Therefore, lack of a suitable parameterization for

Table 5.1: Simulation of the construction of a covariance matrix from three existing covariance matrices. Given the input matrices Σ_1, Σ_2, and Σ_3, a new matrix is constructed as $\Sigma_{\text{new}} = \Sigma_1 + (\Sigma_2 - \Sigma_3)$ in an arithmetic operation that is often found in many stochastic search algorithms. This operation is repeated for $100,000$ times for different input matrices at each dimensionality reported in the first row. As shown in the second row, the number of Σ_{new} that is positive definite, i.e., a valid covariance matrix, decreases significantly at increasing dimensions. This shows that the entries in the covariance matrix cannot be directly used as parameters in stochastic search algorithms.

Dimension	3	5	10	15	20	30
# valid	44,652	27,443	2,882	103	1	0

arbitrary covariance matrices has limited the flexibility of the existing approaches in modeling the covariance structure of the components in the mixture.

In this Section, first, we propose a novel parameterization where the parameters of an arbitrary covariance matrix are independently modifiable and can have upper and lower bounds. We also describe an algorithm for unique identification of these parameters from a valid covariance matrix. Then, we describe a new solution to the mixture identifiability problem where different orderings of the Gaussian components in different candidate solutions can significantly affect the convergence of the search procedure. The proposed approach solves this issue by using a two-stage interaction between the candidate solutions. In the first stage, the optimum correspondences among the components of two candidate solutions are identified. Once these correspondences are identified, in the second stage, desirable interactions such as selection, swapping, addition, and perturbation can be performed. Both the proposed parameterization and the solutions for the two identifiability problems allow effective use of population-based stochastic search algorithms for the estimation of GMMs.

5.4.1 Covariance Parameterization

The proposed covariance parameterization is based on eigenvalue decomposition of the covariance matrix. For a given d-dimensional covariance matrix $\mathbf{\Sigma} \in \mathbb{S}_{++}^{d}$,

let $\{\lambda_i, \boldsymbol{\nu}_i\}$ for $i = 1, \ldots, d$ denote the eigenvalue-eigenvector pairs in a particular order where $\lambda_i \in \mathbb{R}_{++}$ for $i = 1, \ldots, d$ correspond to the eigenvalues and $\boldsymbol{\nu}_i \in \mathbb{R}^d$ such that $\|\boldsymbol{\nu}_i\|_2 = 1$ and $\boldsymbol{\nu}_i^T \boldsymbol{\nu}_j = 0$ for $i \neq j$ represent the eigenvectors. A given d-dimensional covariance matrix $\boldsymbol{\Sigma}$ can be written in terms of its eigenvalue-eigenvector pairs as $\boldsymbol{\Sigma} = \sum_{i=1}^{d} \lambda_i \boldsymbol{\nu}_i \boldsymbol{\nu}_i^T$. Let the diagonal matrix $\boldsymbol{\Lambda} = \text{diag}(\lambda_1, \ldots, \lambda_d)$ denote the eigenvalue matrix, and the unitary matrix $\mathbf{V} = (\boldsymbol{\nu}_1, \ldots, \boldsymbol{\nu}_d)$ denote the corresponding eigenvector matrix where the normalized eigenvectors are placed into the columns of $\mathbf{V}$ in the order determined by the corresponding eigenvalues in $\boldsymbol{\Lambda}$. Then, the given covariance matrix can be written as $\boldsymbol{\Sigma} = \mathbf{V}\boldsymbol{\Lambda}\mathbf{V}^T$.

Due to its symmetric structure, an arbitrary d-dimensional covariance matrix has $d(d+1)/2$ degrees of freedom; thus, at most $d(d+1)/2$ parameters are needed to represent this matrix. The proposed parameterization is based on the following theorem.

Theorem 1. *An arbitrary covariance matrix with $d(d+1)/2$ degrees of freedom can be parametrized using d eigenvalues in a particular order and $d(d-1)/2$ Givens rotation matrix angles $\phi^{pq} \in [-\pi/4, 3\pi/4]$ for $1 \leq p < q \leq d$ computed from the eigenvector matrix whose columns store the eigenvectors in the same order as the corresponding eigenvalues.*

The proof is based on the following Definition, Proposition, and Lemma. An example parameterization for a 3×3 covariance matrix is given in Figure 5.1.

Definition 22. *A Givens rotation matrix $\mathbf{G}(p, q, \phi^{pq})$ with three input parameters corresponding to two indices p and q that satisfy $p < q$, and an angle ϕ^{pq} has the form*

$$\mathbf{G}(p, q, \phi^{pq}) = \begin{pmatrix} 1 & \cdots & 0 & \cdots & 0 & \cdots & 0 \\ \vdots & \ddots & \vdots & & \vdots & & \vdots \\ 0 & \cdots & cos(\phi^{pq}) & \cdots & sin(\phi^{pq}) & \cdots & 0 \\ \vdots & & \vdots & \ddots & \vdots & & \vdots \\ 0 & & -sin(\phi^{pq}) & \cdots & cos(\phi^{pq}) & \cdots & 0 \\ \vdots & & \vdots & & \vdots & \ddots & \vdots \\ 0 & \cdots & 0 & \cdots & 0 & \cdots & 1 \end{pmatrix}. \tag{5.6}$$

Premultiplication by $\mathbf{G}(p, q, \phi^{pq})^T$ corresponds to a counterclockwise rotation of ϕ radians in the plane spanned by two coordinate axes indexed by p and q [155].

Proposition 11. *A Givens rotation can be used to zero a particular entry in a vector. Given scalars a and b, the* $c = \cos(\phi)$ *and* $s = \sin(\phi)$ *values in* (5.6) *that can zero b can be computed as the solution of*

$$\begin{pmatrix} c & s \\ -s & c \end{pmatrix}^T \begin{pmatrix} a \\ b \end{pmatrix} = \begin{pmatrix} h \\ 0 \end{pmatrix} \tag{5.7}$$

using the following algorithm [155]

if $b = 0$ ***then***
  $c = 1;\ s = 0$
else
  if $|b| > |a|$ ***then***
    $\tau = -a/b;\ s = 1/\sqrt{1+\tau^2};\ c = s\tau$
  else
    $\tau = -b/a;\ c = 1/\sqrt{1+\tau^2};\ s = c\tau$
  end if
end if

where ϕ *can be computed as* $\phi = \arctan(s/c)$*. The resulting Givens rotation angle* ϕ *is in the range* $[-\pi/4, 3\pi/4]$ *by definition (because of the absolute values in the algorithm).*

Lemma 1. *An eigenvector matrix* $\mathbf{V}$ *of size* $d \times d$ *can be written as a product of* $d(d-1)/2$ *Givens rotation matrices whose angles lie in the interval* $[-\pi/4, 3\pi/4]$ *and a diagonal matrix whose entries are either* $+1$ *or* -1*.*

Proof of Lemma 1. Existence of such a decomposition can be shown by using QR factorization via a series of Givens rotations. QR factorization decomposes any real square matrix into a product of an orthogonal matrix $\mathbf{Q}$ and an upper triangular matrix $\mathbf{R}$, and can be computed by using Givens rotations where each rotation zeros an element below the diagonal of the input matrix. When the QR algorithm is applied to $\mathbf{V}$, the angle ϕ^{pq} for the given indices p and q is calculated using the values $\mathbf{V}(p,p)$ and $\mathbf{V}(q,p)$ as the scalars a and b, respectively, in Definition 11, and then, $\mathbf{V}$ is premultiplied with the transpose of the Givens rotation matrix as $\mathbf{G}(p,q,\phi^{pq})^T\mathbf{V}$ where $\mathbf{G}$ is defined in Definition 22. This

multiplication zeros the value $\mathbf{V}(q,p)$. This process is continued for $p = 1, \ldots, d-1$ and $q = p+1, \ldots, d$, resulting in the orthogonal matrix

$$\mathbf{Q} = \prod_{p=1}^{d-1} \prod_{q=p+1}^{d} \mathbf{G}(p, q, \phi^{pq}) \tag{5.8}$$

and the triangular matrix

$$\mathbf{R} = \mathbf{Q}^T \mathbf{V}. \tag{5.9}$$

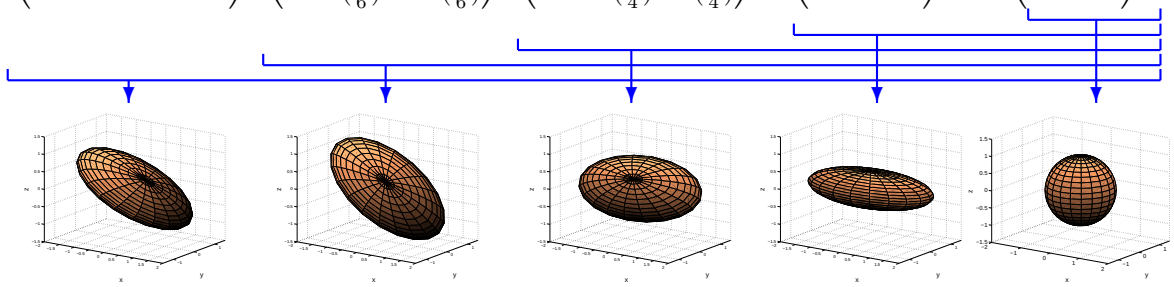
Since the eigenvector matrix $\mathbf{V}$ is orthogonal, i.e., $\mathbf{V}^T\mathbf{V} = \mathbf{I}$, $\mathbf{R}^T\mathbf{Q}^T\mathbf{Q}\mathbf{R} = \mathbf{I}$ leads to $\mathbf{R}^T\mathbf{R} = \mathbf{I}$ because $\mathbf{Q}$ is also orthogonal. Since $\mathbf{R}$ should be both orthogonal and upper triangular, we conclude that $\mathbf{R}$ is a diagonal matrix whose entries are either $+1$ or -1. □

Proof of Theorem 1. Following Lemma 1, an eigenvector matrix $\mathbf{V}$ in which the eigenvectors are stored in a particular order can be written using $d(d-1)/2$ angle parameters for the $\mathbf{Q}$ matrix and an additional d parameters for the $\mathbf{R}$ matrix. However, since both $\boldsymbol{\nu}_i$ and $-\boldsymbol{\nu}_i$ are valid eigenvectors ($\boldsymbol{\Sigma}\boldsymbol{\nu}_i = \lambda_i\boldsymbol{\nu}_i$ and $\boldsymbol{\Sigma}(-\boldsymbol{\nu}_i) = \lambda_i(-\boldsymbol{\nu}_i)$), we can show that those additional d parameters for the diagonal of $\mathbf{R}$ are not required for the parameterization of eigenvector matrices.

This follows from the invariance of the Givens rotation angles to the rotation of the eigenvectors with π radians such that when any column of the $\mathbf{V}$ matrix is multiplied by -1, only the $\mathbf{R}$ matrix changes, while the $\mathbf{Q}$ matrix, hence the Givens rotation angles, do not change. To prove this invariance, let $\mathcal{P} = \{\mathbf{P} | \mathbf{P} \in \mathbb{R}^{d \times d}, \mathbf{P}(i,j) = 0, \forall i \neq j, \text{and } \mathbf{P}(i,i) \in \{+1, -1\} \text{ for } i = 1, \ldots, d\}$ be a set of modification matrices. For a given $\mathbf{P} \in \mathcal{P}$, define $\hat{\mathbf{V}} = \mathbf{V}\mathbf{P}$. Since $\mathbf{V} = \mathbf{Q}\mathbf{R}$, we have $\hat{\mathbf{V}} = \mathbf{Q}\mathbf{R}\mathbf{P}$. Then, defining $\hat{\mathbf{R}} = \mathbf{R}\mathbf{P}$ gives $\hat{\mathbf{V}} = \mathbf{Q}\hat{\mathbf{R}}$. Since $\mathbf{Q}$ did not change and $\hat{\mathbf{R}} = \mathbf{R}\mathbf{P}$ is still a diagonal matrix whose entries are either $+1$ or -1, it is a valid QR factorization. Therefore, we can conclude that there exists a QR factorization $\hat{\mathbf{V}} = \mathbf{Q}\hat{\mathbf{R}}$ with the same $\mathbf{Q}$ matrix as the QR factorization $\mathbf{V} = \mathbf{Q}\mathbf{R}$.

The discussion above shows that the $d(d-1)/2$ Givens rotation angles are sufficient for the parameterization of the eigenvectors because the multiplication

$$
\begin{aligned}
&\mathbf{V}\sqrt{\mathbf{\Lambda}} \\
&= \quad \mathbf{G}(1,2,\pi/3) \qquad \mathbf{G}(1,3,\pi/6) \qquad \mathbf{G}(2,3,\pi/4) \qquad \sqrt{\mathbf{\Lambda}} \qquad \mathbf{I} \\
&= \begin{pmatrix} \cos(\frac{\pi}{3}) & \sin(\frac{\pi}{3}) & 0 \\ -\sin(\frac{\pi}{3}) & \cos(\frac{\pi}{3}) & 0 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} \cos(\frac{\pi}{6}) & 0 & \sin(\frac{\pi}{6}) \\ 0 & 1 & 0 \\ -\sin(\frac{\pi}{6}) & 0 & \cos(\frac{\pi}{6}) \end{pmatrix} \begin{pmatrix} 1 & 0 & 0 \\ 0 & \cos(\frac{\pi}{4}) & \sin(\frac{\pi}{4}) \\ 0 & -\sin(\frac{\pi}{4}) & \cos(\frac{\pi}{4}) \end{pmatrix} \begin{pmatrix} 2 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0.5 \end{pmatrix} \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}
\end{aligned}
$$



$$
= \begin{pmatrix} 0.87 & 0.44 & 0.39 \\ -1.50 & 0.66 & 0.02 \\ -1.00 & -0.61 & 0.31 \end{pmatrix}
$$

$$
\mathbf{\Sigma} = (\mathbf{V}\sqrt{\mathbf{\Lambda}})\ (\mathbf{V}\sqrt{\mathbf{\Lambda}})^T = \begin{pmatrix} 0.87 & 0.44 & 0.39 \\ -1.50 & 0.66 & 0.02 \\ -1.00 & -0.61 & 0.31 \end{pmatrix} \begin{pmatrix} 0.87 & 0.44 & 0.39 \\ -1.50 & 0.66 & 0.02 \\ -1.00 & -0.61 & 0.31 \end{pmatrix}^T =
$$

$$
\begin{pmatrix} 1.10 & -1.00 & -1.01 \\ -1.00 & 2.69 & 1.10 \\ -1.01 & 1.10 & 1.47 \end{pmatrix}
$$

Figure 5.1: Example parameterization for a 3×3 covariance matrix. The example matrix can be parametrized using $\{\lambda_1, \lambda_2, \lambda_3, \phi^{12}, \phi^{13}, \phi^{23}\} = \{4, 1, 0.25, \pi/3, \pi/6, \pi/4\}$. The ellipses from right to left show the covariance structure resulting from each step of premultiplication of the result of the previous step, starting from the identity matrix.

of any eigenvector by -1 leads to the same covariance matrix $\mathbf{\Sigma}$, i.e.,

$$
\begin{aligned}
\mathbf{\Sigma} &= \sum_{i=1,\, i\neq j}^{d} \lambda_i \boldsymbol{\nu}_i \boldsymbol{\nu}_i^T + \lambda_j (-\boldsymbol{\nu}_j)(-\boldsymbol{\nu}_j)^T \\
&= \sum_{i=1,\, i\neq j}^{d} \lambda_i \boldsymbol{\nu}_i \boldsymbol{\nu}_i^T + \lambda_j (\boldsymbol{\nu}_j)(\boldsymbol{\nu}_j)^T \qquad (5.10) \\
&= \sum_{i=1}^{d} \lambda_i \boldsymbol{\nu}_i \boldsymbol{\nu}_i^T .
\end{aligned}
$$

Finally, together with the d eigenvalues, the covariance matrix can be constructed as $\mathbf{\Sigma} = \mathbf{V}\mathbf{\Lambda}\mathbf{V}^T$. □

Table 5.2: To demonstrate its non-uniqueness, all equivalent parameterizations of the example covariance matrix given in Figure 5.1 for different orderings of the eigenvalue-eigenvector pairs. The angles are given in degrees.

λ_1	λ_2	λ_3	ϕ^{12}	ϕ^{13}	ϕ^{23}
4	1	0.25	60.00	30.00	45.00
4	0.25	1	60.00	30.00	-45.00
1	4	0.25	123.43	-37.76	39.23
1	0.25	4	123.43	-37.76	129.23
0.25	4	1	-3.43	-37.76	-39.23
0.25	1	4	-3.43	-37.76	50.77

5.4.2 Identifiability of Individual Gaussians

Theorem 1 assumes that the eigenvalue-eigenvector pairs are given in a particular order. However, since any d-dimensional covariance matrix $\mathbf{\Sigma} \in \mathbb{S}^d_{++}$ can be written as $\mathbf{\Sigma} = \sum_{i=1}^{d} \lambda_i \boldsymbol{\nu}_i \boldsymbol{\nu}_i^T$ and there is no inherent ordering of the eigenvalue-eigenvector pairs, it is possible to write this summation in terms of $d!$ different eigenvalue and eigenvector matrices as $\mathbf{\Sigma} = \mathbf{V}\mathbf{\Lambda}\mathbf{V}^T$ simply by changing the order of the eigenvalues and their corresponding eigenvectors in the eigendecomposition matrices $\mathbf{\Lambda}$ and $\mathbf{V}$. For example, all equivalent parameterizations of the example covariance matrix in Figure 5.1 are given in Table 5.2. Furthermore, multiplying any column of the eigenvector matrix by -1 still gives a valid eigenvector matrix, resulting in 2^d possibilities. Since we showed that there exists a unique $\mathbf{Q}$ matrix and a corresponding set of unique Givens rotation angles can be extracted via QR factorization in the proof of Theorem 1, the result is invariant to these 2^d possibilities. However, for an improved efficiency in the global search, it is one of our goals to pair the parameters between alternate solution candidates before performing any interactions among them. Therefore, the dependence of the results on the $d!$ orderings and the resulting equivalence classes still need to be eliminated.

In order to have unique eigenvalue decomposition matrices, we propose an ordering algorithm based on the eigenvectors so that from a given covariance matrix we can obtain uniquely ordered eigenvalue and eigenvector matrices, leading to a unique set of eigenvalues and Givens rotation angles as the parameters. The

ordering algorithm uses only the eigenvectors and not the eigenvalues because the eigenvectors correspond to the principal directions of the data whereas the eigenvalues indicate the amount of the extent of the data along these directions.

The proposed eigenvalue-eigenvector ordering algorithm uses an orthogonal basis matrix as a reference. In this greedy selection algorithm, the eigenvector among the unselected ones having the maximum absolute inner product with the i'th reference vector is put into the i'th column in the output matrix. Let $\mathcal{S}^{\text{in}} = \{\{\lambda_1^{\text{in}}, \boldsymbol{\nu}_1^{\text{in}}\}, \ldots, \{\lambda_d^{\text{in}}, \boldsymbol{\nu}_d^{\text{in}}\}\}$ denote the input eigenvalue-eigenvector pair set, $\mathbf{V}^{\text{ref}} = (\boldsymbol{\nu}_1^{\text{ref}}, \ldots, \boldsymbol{\nu}_d^{\text{ref}})$ denote the reference orthogonal basis matrix, $\boldsymbol{\Lambda}^{\text{out}} = \text{diag}(\lambda_1^{\text{out}}, \ldots, \lambda_d^{\text{out}})$ and $\mathbf{V}^{\text{out}} = (\boldsymbol{\nu}_1^{\text{out}}, \ldots, \boldsymbol{\nu}_d^{\text{out}})$ denote the final output eigenvalue and eigenvector matrices, and $\mathcal{I}$ be the set of indices of the remaining eigenvalue-eigenvector pairs that need to be ordered. The ordering algorithm is defined in Algorithm 1.

Algorithm 1 Eigenvector ordering algorithm.

Input: $\mathcal{S}^{\text{in}}$, $\mathbf{V}^{\text{ref}}$, $\mathcal{I} = \{1, \ldots, d\}$
Output: $\boldsymbol{\Lambda}^{\text{out}}$, $\mathbf{V}^{\text{out}}$
1: **for** $i = 1$ to d **do**
2: $\quad i^* = \arg\max_{j \in \mathcal{I}} |(\boldsymbol{\nu}_j^{\text{in}})^T (\boldsymbol{\nu}_i^{\text{ref}})|$
3: $\quad \lambda_i^{\text{out}} \leftarrow \lambda_{i^*}^{\text{in}}$
4: $\quad \boldsymbol{\nu}_i^{\text{out}} \leftarrow \boldsymbol{\nu}_{i^*}^{\text{in}}$
5: $\quad \mathcal{I} \leftarrow \mathcal{I} - \{i^*\}$
6: **end for**

Any reference basis matrix $\mathbf{V}^{\text{ref}}$ in Algorithm 1 will eliminate the dependency on the $d!$ orderings, and will result in a unique set of parameters. However, the choice of $\mathbf{V}^{\text{ref}}$ can affect the convergence of the likelihood during estimation. We performed simulations to determine the most effective reference matrix $\mathbf{V}^{\text{ref}}$ for eigenvector ordering. The maximum likelihood estimation problem to estimate the covariance matrix of a single Gaussian is given as follows. Given a set of N data points $\mathcal{X} = \{\mathbf{x}_1, \ldots, \mathbf{x}_N\}$ where each $\mathbf{x}_j \in \mathbb{R}^d$ is independent and identically distributed according to a Gaussian with zero mean and covariance matrix Σ, the negative log-likelihood function

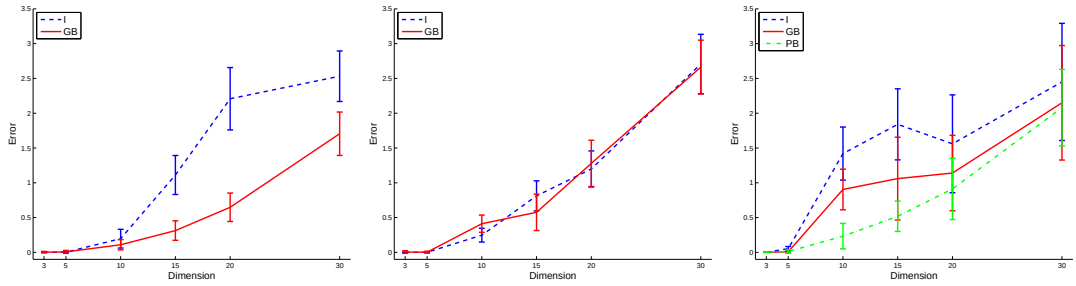
$$\frac{1}{2}\log(|\Sigma|) - \frac{1}{2N}\sum_{j=1}^{N} \mathbf{x}_j^T \Sigma^{-1} \mathbf{x}_j - \frac{d}{2}\log(2\pi) \tag{5.11}$$

can be rewritten as

$$-\frac{1}{2}\log(|\Sigma^{-1}|) - \frac{1}{2}\text{tr}(\Sigma^{-1}\Sigma_s) - \frac{d}{2}\log(2\pi) \tag{5.12}$$

where $\Sigma_s = \frac{1}{N}\sum_{i=1}^{N}\mathbf{x}_i\mathbf{x}_i^T$. Thus, the maximum likelihood estimate of Σ can be found as the one that minimizes $-\log(|\Sigma^{-1}|) - \text{tr}(\Sigma^{-1}\Sigma_s)$. We solved this minimization problem using GA, DE, and PSO implemented as described in Chapter 1. For GA and DE, candidate reference matrices were the identity matrix and the eigenvector matrix corresponding to the global best solution. For PSO, candidate reference matrices were the identity matrix, the eigenvector matrix corresponding to each particle's personal best, and the eigenvector matrix corresponding to the global best particle. For each case, 100 different target Gaussians (Σ_s in (5.12)) were randomly generated by sampling the eigenvalues from the uniform distribution Uniform$[0.1, 1.0]$ and the Givens rotation angles from the uniform distribution Uniform$[-\pi/4, 3\pi/4]$. This was repeated for dimensions $d \in \{3, 5, 10, 15, 20, 30\}$, and the respective optimization algorithm was used to find the corresponding covariance matrix (Σ in (5.12)) that minimizes the negative log-likelihood using 10 different initializations. Figure 5.2 shows the plots of estimation errors resulting from the $1,000$ trials. The error was computed as the difference between the target log-likelihood computed from the true Gaussian parameters ($\Sigma = \Sigma_s$) and the resulting log-likelihood computed from the estimated Gaussian parameters. Based on these results, we can conclude that the eigenvector matrix corresponding to the personal best solution for PSO, and the eigenvector matrix corresponding to the global best solution for GA and DE (no personal best is available in GA and DE) can be used as the reference matrix in the eigenvector ordering algorithm.

Summary: The discussion above demonstrated that a d-dimensional covariance matrix $\mathbf{\Sigma} \in \mathbb{S}_{++}^d$ can be parametrized using d eigenvalues $\lambda_i \in \mathbb{R}_{++}$ for $i = 1, \ldots, d$ and $d(d-1)/2$ angles $\phi^{pq} \in [-\pi/4, 3\pi/4]$ for $1 \leq p < q \leq d$. We showed that, for a given covariance matrix, these parameters can be uniquely extracted using eigenvalue decomposition, the proposed eigenvector ordering algorithm that aligns the principal axes of the covariance ellipsoids among alternate candidate solutions, and QR factorization using the Givens rotations method.



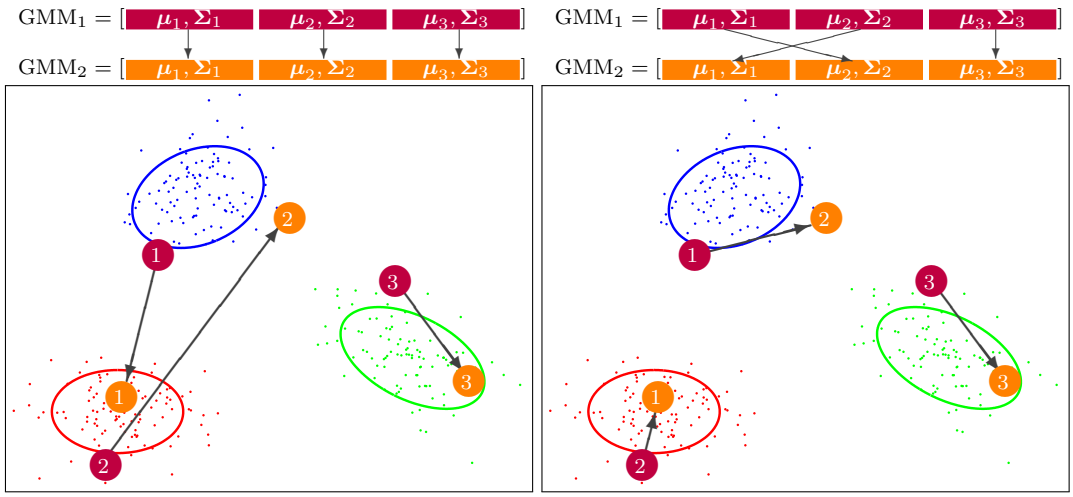
(a) GA (b) DE (c) PSO

Figure 5.2: Average error in log-likelihood and its standard deviation (shown as error bars at one standard deviation) in 1,000 trials for different choices of reference matrices in eigenvector ordering during the estimation of the covariance matrix of a single Gaussian using stochastic search. Choices for the reference matrix are I: identity matrix, GB: the eigenvector matrix corresponding to the global best solution, and PB: the eigenvector matrix corresponding to the personal best solution.

We also showed that, given these parameters, a covariance matrix can be generated from the eigenvalue matrix $\mathbf{\Lambda} = \text{diag}(\lambda_1, \ldots, \lambda_d)$ and the eigenvector matrix $\mathbf{V} = \prod_{p=1}^{d-1} \prod_{q=p+1}^{d} \mathbf{G}(p, q, \phi^{pq})$ as $\mathbf{\Sigma} = \mathbf{V}\mathbf{\Lambda}\mathbf{V}^T$.

5.4.3 Identifiability of Gaussian Mixtures

Similar to the problem of ordering of the parameters within individual Gaussian components to obtain a unique set of parameters as discussed in the previous section, ordering of the Gaussian components within a candidate solution is important for obtaining a unique correspondence between two candidate solutions during their interactions for parameter updates throughout the stochastic search. The correspondence identifiability problem that arises from the equivalency of $K!$ possible orderings of individual components in a candidate solution for a mixture of K Gaussians affects the convergence of the search procedure. First of all, when the likelihood function has a mode under a particular ordering of the components, there exists $K!$ symmetric modes corresponding to all parameter sets that are in the same equivalence class formed by the permutation of these components. When these equivalencies are not known, a search algorithm may



(a) Default correspondence relation (b) Desired correspondence relation

Figure 5.3: Example correspondence relations for two GMMs with three components. The ellipses represent the true components corresponding to the colored sample points. The numbered blobs represent the locations of the components in the candidate solutions. When the parameter updates are performed according to the component pairs in the default order, some of the components may be updated based on interactions with components in different parts of the data space. However, using the reference matching procedure, a more desirable correspondence relation can be found enabling faster convergence.

not cover the solution space effectively as equivalent configurations of components may be repeatedly explored. In a related problem, in the extreme case, a reproduction operation applied to two candidate solutions that are essentially equal may result in a new solution that is completely different from its parents. Secondly, the knowledge of the correspondences helps performing the update operations as intended. For example, even for two candidate solutions that are not in the same equivalence class, alignment of their components enables effective use of both direct interactions and cross interactions. For instance, cross interactions may be useful to increase diversity; on the other hand, direct interactions may be more helpful to find local minima. Without such alignment of the components, these interactions cannot be controlled as desired, and the iterations proceed with arbitrary exploration of the search space. Figure 5.3 shows examples for default and desired correspondence relations for two GMMs with three components.

We propose a matching algorithm for finding the correct correspondence relation between the components of two GMMs to enable interactions between the corresponding components in different solution candidates. In the following, the correspondence identification problem is formulated as a minimum cost network flow optimization problem. The objective is to find the correspondence relation that minimizes the sum of Kullback-Leibler (KL) divergences between pairs of Gaussian components. For two Gaussians $g_1(\mathbf{x}|\mu_1, \Sigma_1)$ and $g_2(\mathbf{x}|\mu_2, \Sigma_2)$, the KL divergence has the closed form expression

$$D(g_1 \| g_2) = \frac{1}{2}\left(\log\frac{|\Sigma_2|}{|\Sigma_1|} + \mathrm{tr}\Big(\Sigma_2^{-1}\Sigma_1\Big) - d + (\mu_1 - \mu_2)^T\Sigma_2^{-1}(\mu_1 - \mu_2)\right). \quad (5.13)$$

Consequently, given a target GMM with parameters $\{\{\mu_1^{\mathrm{tar}}, \Sigma_1^{\mathrm{tar}}\}, \ldots \{\mu_K^{\mathrm{tar}}, \Sigma_K^{\mathrm{tar}}\}\}$ and a reference GMM with parameters $\{\{\mu_1^{\mathrm{ref}}, \Sigma_1^{\mathrm{ref}}\}, \ldots \{\mu_K^{\mathrm{ref}}, \Sigma_K^{\mathrm{ref}}\}\}$, the cost of matching the i'th component of the first GMM to the j'th component of the second GMM is computed as

$$c_{ij} = \log\frac{|\Sigma_j^{\mathrm{ref}}|}{|\Sigma_i^{\mathrm{tar}}|} + \mathrm{tr}\Big((\Sigma_j^{\mathrm{ref}})^{-1}\Sigma_i^{\mathrm{tar}}\Big) + (\mu_i^{\mathrm{tar}} - \mu_j^{\mathrm{ref}})^T(\Sigma_j^{\mathrm{ref}})^{-1}(\mu_i^{\mathrm{tar}} - \mu_j^{\mathrm{ref}}), \quad (5.14)$$

and the correspondences are found by solving the following optimization problem:

$$\begin{aligned}
\underset{I_{11},\ldots,I_{KK}}{\text{minimize}} \quad & \textstyle\sum_{i=1}^{K}\sum_{j=1}^{K} c_{ij} I_{ij} \\
\text{subject to} \quad & \textstyle\sum_{i=1}^{K} I_{ij} = 1, \quad \forall j \in \{1,\ldots,K\} \\
& \textstyle\sum_{j=1}^{K} I_{ij} = 1, \quad \forall i \in \{1,\ldots,K\} \\
& I_{ij} = \begin{cases} 1, & \text{correspondence between} \\ & \quad i\text{'th and } j\text{'th components} \\ 0, & \text{otherwise.} \end{cases}
\end{aligned} \quad (5.15)$$

In this formulation, the first and third constraints force each component of the first GMM to be matched with only one component of the second GMM, and the second constraint makes sure that only one component of the first GMM is matched to each component of the second GMM. This optimization problem can be solved very efficiently using the Edmonds-Karp algorithm [156]. Note that the solution of the optimization problem in (5.15) does not change under any permutation of the component labels in the target and reference GMMs. Figure 5.4 illustrates the optimization formulation for the example in Figure 5.3. Once

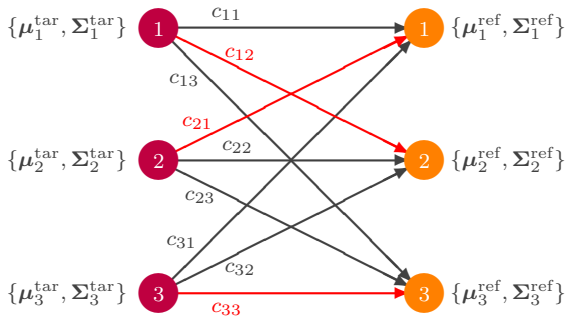


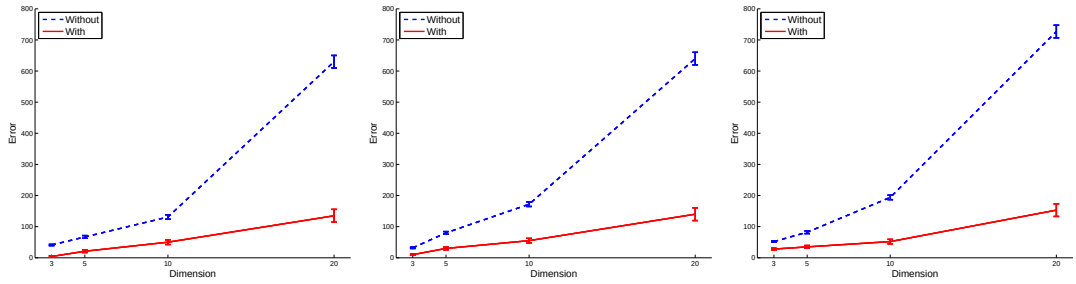
Figure 5.4: Optimization formulation for two GMMs with three components shown in Figure 5.3. The correspondences found are shown in red.

the correspondences are established, the parameter updates can be performed as described in the examples for different stochastic search algorithms in Section 5.1 in general and as presented for the particle swarm optimization in Section 5.5 in particular.

We performed simulations to evaluate the effectiveness of correspondence identification using the proposed matching algorithm. We ran the stochastic search algorithms GA, DE, and PSO for the maximum likelihood estimation of the Gaussian mixture model parameters that were synthetically generated as follows. The mixture weights were sampled from a uniform distribution such that the ratio of the largest weight to the smallest weight was at most 1.3 and all weights summed up to 1. The mean vectors were sampled from the uniform distribution $\text{Uniform}[0,1]^d$ where d was the number of dimensions. The covariance matrices were generated by sampling the eigenvalues from the uniform distribution $\text{Uniform}[1, 1.6]$ and the Givens rotation angles from the uniform distribution $\text{Uniform}[-\pi/4, 3\pi/4]$. The minimum separation between the components in the mixture was controlled with a parameter called c. Two Gaussians are defined to be c-separated if

$$\|\mu_1 - \mu_2\|_2 \leq c\sqrt{d \max\{\lambda_{\max}(\Sigma_1), \lambda_{\max}(\Sigma_2)\}} \tag{5.16}$$

where $\lambda_{\max}(\Sigma)$ is the largest eigenvalue of the given covariance matrix [157]. The randomly generated Gaussian components in a mixture were forced to satisfy the pairwise c-separation constraint. The mixtures in the simulations were generated



(a) GA (b) DE (c) PSO

Figure 5.5: Average error in log-likelihood and its standard deviation (shown as error bars at one standard deviation) in 1,000 trials without and with the correspondence identification step in the estimation of GMMs using stochastic search.

for $c = 4.0$, $K = 5$, and dimensions $d \in \{3, 5, 10, 20\}$. 100 such mixtures were generated, and 1,000 points were sampled from each mixture. The parameters in the candidate solutions in GA, DE, and PSO were randomly initialized as follows. The mean vectors were sampled from the uniform distribution Uniform$[0, 1]^d$, the eigenvalues of the covariance matrices were sampled from the uniform distribution Uniform$[0, 10]$, and the Givens rotation angles were sampled from the uniform distribution Uniform$[-\pi/4, 3\pi/4]$. 10 different initializations were used for each mixture, resulting in 1,000 trials. The true parameters were compared to the estimation results obtained without and with correspondence identification. Figure 5.5 shows the plots of estimation errors resulting from the 1,000 trials. The error was computed as the difference between the target log-likelihood computed from the true GMM parameters and the resulting log-likelihood computed from the estimated GMM parameters. Based on these results, we can conclude that using the proposed correspondence identification algorithm leads to significantly better results for all stochastic search algorithms used.

5.5 Particle Swarm Optimization

We illustrate the proposed solutions for the estimation of GMMs using stochastic search in a particle swarm optimization (PSO) framework. The following Sections

briefly describe the general PSO formulation by setting up the notation, and then present the details of the GMM estimation procedure using PSO.

5.5.1 General Formulation

PSO is a population-based stochastic search algorithm that is inspired by the social interactions of swarm animals. In PSO, each member of the population is called a particle. Each particle $\mathbf{Z}^{(m)}$ is composed of two vectors, a position vector $\mathbf{Z}_u^{(m)}$ and a velocity vector $\mathbf{Z}_v^{(m)}$ where $m = 1, \ldots, M$ indicates the particle index in a population of M particles. The position of each particle $\mathbf{Z}_u^{(m)} \in \mathbb{R}^n$ corresponds to a candidate solution for an n-dimensional optimization problem.

A fitness function defined for the optimization problem of interest is used to assign a goodness value to a particle based on its position. The particle having the best fitness value is called the *global best*, and this position is denoted as $\mathbf{Z}_u^{(\mathrm{GB})}$. Each particle also remembers its best position throughout the search history as its *personal best*, and this position is denoted as $\mathbf{Z}_u^{(m,\mathrm{PB})}$.

PSO begins by initializing the particles with random positions and small random velocities in the n-dimensional parameter space. In the subsequent iterations, each of the n velocity components in $\mathbf{Z}_v^{(m)}$ is computed independently using its previous value, the global best, and the particle's own personal best in a stochastic manner as

$$\begin{aligned}\mathbf{Z}_v^{(m)}(t+1) = \eta\, \mathbf{Z}_v^{(m)}(t) + c_1\, U_1(t) \left(\mathbf{Z}_v^{(m,\mathrm{PB})}(t) - \mathbf{Z}_v^{(m)}(t)\right) \\ + c_2\, U_2(t) \left(\mathbf{Z}_v^{(\mathrm{GB})}(t) - \mathbf{Z}_v^{(m)}(t)\right) \quad (5.17)\end{aligned}$$

where η is the inertia weight, U_1 and U_2 represent random numbers sampled from Uniform$[0, 1]$, c_1 and c_2 are acceleration weights, and t is the iteration number. Each particle moves from its old position to a new position using its new velocity vector as

$$\mathbf{Z}_u^{(m)}(t+1) = \mathbf{Z}_u^{(m)}(t) + \mathbf{Z}_v^{(m)}(t+1), \quad (5.18)$$

and its personal best is modified if necessary. Additionally, the global best of the population is updated based on the particles' new fitness values.

The main difference between PSO and other popular search algorithms like genetic algorithms and differential evolution is that PSO is not an evolutionary algorithm. In evolutionary algorithms, a newly created particle cannot be kept unless it has a better fitness value. However, in PSO, particles are allowed to move to worse locations and this mechanism allows the particles to escape from local optima gradually without the need of any long jump mechanism. In evolutionary algorithms, this can generally be achieved by mutation and crossover operations but these operations can be hard to design for different problems. In addition, PSO uses the global best to coordinate the movement of all particles and uses personal bests to keep track of all local optima found. These properties make it easier to incorporate problem specific ideas into PSO where the global best serves as the current state of the problem and the personal bests serve as the current states of the particles.

5.5.2 GMM Estimation Using PSO

The solutions proposed in this Chapter enable the formulation of a PSO framework for the estimation of GMMs with arbitrary covariance matrices. This formulation involves the definition of the particles, the initialization procedure, the fitness function, and the update procedure.

5.5.2.1 Particle Definition

Each particle that corresponds to a candidate solution stores the parameters of the means and covariance matrices of a GMM. Assuming that the number of components in the mixture is fixed as K, the position vector of the m'th particle is defined as

$$\mathbf{Z}_u^{(m)} = ((\mu_u^{(m,k)})^T, \lambda_{1,u}^{(m,k)}, \ldots, \lambda_{d,u}^{(m,k)}, \phi_u^{12,(m,k)}, \ldots \phi_u^{(d-1)(d),(m,k)}, \text{for } k = 1, \ldots, K) \tag{5.19}$$

where $\mu_u^{(m,k)} \in \mathbb{R}^d$ for $k = 1, \ldots, K$ denote the mean vectors parametrized using d real numbers, $\lambda_{i,u}^{(m,k)} \in \mathbb{R}_{++}$ for $i = 1, \ldots, d$ and $k = 1, \ldots, K$ denote the

eigenvalues of the covariance matrices, and $\phi_u^{pq,(m,k)} \in [-\pi/4, 3\pi/4]$ for $1 \leq p < q \leq d$ and $k = 1, \ldots, K$ denote the Givens rotation angles as defined in Section 5.4.1. The velocity vector $\mathbf{Z}_v^{(m)}$ is defined similarly. The K mixture weights $\alpha_1, \ldots, \alpha_K$ are calculated from the probabilistic assignments of the data points to the components, and are not part of the PSO particles.

5.5.2.2 Initialization

Initialization of each particle at the beginning of the first iteration can be done using random numbers within the ranges defined for each parameter. The proposed parameterization makes this possible because the angles are in a fixed range while lower and upper bounds for the mean values and upper bounds for the eigenvalues can easily be selected with the knowledge of the data. As an alternative, one can first randomly select K data points as the means, and form the initial components by assigning each data point to the closest mean. Then, the covariance matrices can be computed from the assigned points, and the parameters of these matrices can be extracted using eigenvalue decomposition and QR factorization using the Givens rotations method as described in Section 5.4.1. Another alternative for selecting the initial components is the k-means initialization procedure described in [73].

5.5.2.3 Fitness Function

The PSO iterations proceed to find the maximum likelihood estimates by minimizing the negative log-likelihood defined in (5.1).

5.5.2.4 Update Equations

Before updating each particle as in (5.17) and (5.18), the correspondences between its components and the components of the global best particle are found. This is done by using the particle's personal best as the reference GMM and the

global best particle as the target GMM in (5.14). The correspondence relation computed using (5.14) and (5.15) is denoted with a function $f(k)$ that maps the current particle's component index k to the global best particle's corresponding component index $f(k)$. Using this correspondence relation, the mean parameters are updated as

$$\mu_v^{(m,k)}(t+1) = \eta\,\mu_v^{(m,k)}(t) + c_1(t)\left(\mu_u^{(m,\mathrm{PB},k)}(t) - \mu_u^{(m,k)}(t)\right) + c_2(t)\left(\mu_u^{(\mathrm{GB},f(k))}(t) - \mu_u^{(m,k)}(t)\right), \tag{5.20}$$

$$\mu_u^{(m,k)}(t+1) = \mu_u^{(m,k)}(t) + \mu_v^{(m,k)}(t+1), \tag{5.21}$$

and the eigenvalues and angles as the covariance parameters are updated as

$$\lambda_{i,v}^{(m,k)}(t+1) = \eta\,\lambda_{i,v}^{(m,k)}(t) + c_1(t)\left(\lambda_{i,u}^{(m,\mathrm{PB},k)}(t) - \lambda_{i,u}^{(m,k)}(t)\right) + c_2(t)\left(\lambda_{i,u}^{(\mathrm{GB},f(k))}(t) - \lambda_{i,u}^{(m,k)}(t)\right), \tag{5.22}$$

$$\lambda_{i,u}^{(m,k)}(t+1) = \lambda_{i,u}^{(m,k)}(t) + \lambda_{i,v}^{(m,k)}(t+1) \tag{5.23}$$

$$\phi_v^{pq,(m,k)}(t+1) = \eta\,\phi_v^{pq,(m,k)}(t) + c_1(t)\left(\phi_u^{pq,(m,\mathrm{PB},k)}(t) - \phi_u^{pq,(m,k)}(t)\right) + c_2(t)\left(\phi_u^{pq,(\mathrm{GB},f(k))}(t) - \phi_u^{pq,(m,k)}(t)\right), \tag{5.24}$$

$$\phi_u^{pq,(m,k)}(t+1) = \phi_u^{pq,(m,k)}(t) + \phi_v^{pq,(m,k)}(t+1). \tag{5.25}$$

The uniform random numbers U_1 and U_2 are incorporated into c_1 and c_2. The rest of the notation is same as in Sections 5.4.1 and 5.5.1.

The convergence of the search procedure can also be improved by running a set of EM iterations for each particle at the end of each iteration. After the covariance parameters are updated as above, new covariance matrices are constructed from the parameters using $\Sigma = \mathbf{V}\mathbf{\Lambda}\mathbf{V}^T$, the EM procedure is allowed to converge to a local optimum, and new parameters are computed by performing another set of eigenvalue decomposition and QR factorization steps. These EM iterations help converging to local optima effectively, whereas the PSO iterations handle the search for the global maximum. The overall estimation procedure is summarized in Algorithm 2.

Algorithm 2 PSO algorithm for GMM estimation.

Input: d-dimensional data set with N samples, number of components (K), PSO parameters (η, c_1, and c_2)

Initialize population with M particles as in (5.19)
for $t = 1$ to T_1 **do** {T_1: number of PSO iterations}
    for $m = 1$ to M **do**
        Construct K eigenvalue matrices
        Construct K eigenvector matrices by multiplying Givens rotation angles
        Run EM for local convergence for T_2 iterations {T_2: number of EM iterations for each PSO iteration}
        Compute K eigenvalue and eigenvector matrices via singular value decomposition of new covariance matrices
        Reorder eigenvalues and eigenvectors of each covariance matrix according to personal best
        Extract Givens rotation angles using QR factorization
        Replace particle's means, eigenvalues, and angles
        Calculate log-likelihood
        Update personal best
    end for
    Update global best
    for $m = 1$ to M **do**
        Reorder components of global best according to personal best
        Update particle's means, eigenvalues, and angles as in (5.20)–(5.25)
    end for
end for

5.6 Experiments

We evaluated the framework for GMM estimation (Sections 5.4 and 5.5) using both synthetic and real data sets. Comparative experiments were also done using the EM algorithm (Section 5.3). The procedure used for synthetic data generation and the results for both synthetic and real data sets are given below.

5.6.1 Experiments on Synthetic Data

Data sets of various dimensions $d \in \{5, 10, 15, 20, 30, 40\}$ and number of components $K \in \{5, 10, 15, 20\}$ were generated. For dimensions $d \in \{5, 10, 15\}$, $d = 20$, and $d \in \{30, 40\}$, the sample size N was set to $1,000$, $2,000$, and $4,000$, respectively. The d and N values were chosen based on real data sets used for the experiments described in the next Section. For a particular d and K combination, a GMM was generated as follows. The mixture weights were sampled from a uniform distribution such that the ratio of the largest weight to the smallest weight was at most 2 and all weights summed up to 1. The mean vectors were sampled from the uniform distribution Uniform$[0, 100]^d$. The covariance matrices were generated using the eigenvalue/eigenvector parameterization described in Section 5.4.1. The eigenvalues were sampled from the uniform distribution Uniform$[1, 16]$, and the Givens rotation angles were sampled from the uniform distribution Uniform$[-\pi/4, 3\pi/4]$. Furthermore, the proximity of the components were controlled using c-separation defined in (5.16). Different values of $c \in \{2.0, 4.0, 8.0\}$ were used to control the difficulty of the estimation problem. The selection of c value was based on visual observations in 2-dimensional data. We observed that the minimum value of c where K individual Gaussian components were distinguishable by visual inspection was close to 2.0, and $c = 8.0$ corresponded to the case where the components were well separated. Consequently, we divided the relative difficulties of the data sets into three. The *easy* settings corresponded to $d \in \{5, 10\}$ and $c = 8.0$, the *medium* settings corresponded to $d \in \{10, 15, 20\}$ and $c = 4.0$, and the *hard* settings corresponded to $d \in \{20, 30, 40\}$ and $c = 2.0$. 10 different mixtures with N samples each were

generated for each setting.

The PSO and EM parameters were initialized similarly for a fair evaluation. We assumed that the number of components was known a priori for each data set. Following the common practice in the literature, the initial mean vector for each component was set to a randomly selected data point. The initial covariance matrices and the initial mixture weights were calculated from the probabilistic assignment of the data points to the components with the initial mean vectors and identity covariance matrices. The initial mixture weights were used only in the EM procedure as the proposed algorithm does not include the weights as parameters. After initialization, the search procedure constrained the components of the mean vectors in each particle defined in (5.19) to stay in the data region defined by the minimum and maximum values of each component in the data used for estimation. Similarly, the eigenvalues were constrained to stay in $[\lambda_{\min}, \lambda_{\max}]$ where $\lambda_{\min} = 10^{-5}$ and $\lambda_{\max}$ was the maximum eigenvalue of the covariance matrix of the whole data, and the Givens rotation angles were constrained to lie in $[-\pi/4, 3\pi/4]$. The PSO parameters η, c_1, and c_2 in (5.17) were fixed at $\eta = 0.728$, $c_1 = c_2 = 1.494$ following the common practice in the PSO literature [84]. Thus, no parameter tuning was done during both initialization and search stages in the experiments.

For each test mixture, each PSO run consisted of M particles that were updated for T_1 iterations where each iteration also consisted of at most T_2 EM iterations as described at the end of Section 5.5.2. Each primary EM run consisted of a group of M individual secondary runs where the initial parameters of each secondary run was the same as the parameters of one of the M particles in the corresponding PSO run. Each secondary run was allowed to iterate for at most $T_1 \times T_2$ iterations or until the relative change in the log-likelihood in two consecutive iterations was less than 10^{-6}. The number of iterations were adjusted such that each PSO run (M particles with T_1 PSO iterations and T_2 EM iterations for each PSO iteration) and the corresponding primary EM run (M secondary EM runs with $T_1 \times T_2$ iterations each) were compatible.

Table 5.3 shows the details of the synthetic data sets generated using these

Table 5.3: Details of the synthetic data sets used for performance evaluation. The three groups of rows correspond to the settings categorized as *easy*, *medium*, and *hard* with respect to their relative difficulties. The parameters are described in the text.

Setting #	d	K	c	N	M	T_1	T_2	$T_1 \times T_2$
1	5	5	8.0	1,000	20	30	20	600
2	5	10	8.0	1,000	20	30	20	600
3	10	5	8.0	1,000	20	30	20	600
4	10	5	4.0	1,000	20	30	20	600
5	10	10	4.0	1,000	20	30	20	600
6	10	15	4.0	1,000	20	30	20	600
7	15	5	4.0	1,000	30	30	20	600
8	15	10	4.0	1,000	30	30	20	600
9	15	15	4.0	1,000	30	30	20	600
10	20	5	4.0	2,000	30	50	20	1,000
11	20	10	2.0	2,000	30	50	20	1,000
12	20	15	2.0	2,000	30	50	20	1,000
13	20	20	2.0	2,000	30	50	20	1,000
14	30	10	2.0	4,000	40	100	20	2,000
15	30	15	2.0	4,000	40	100	20	2,000
16	30	20	2.0	4,000	40	100	20	2,000
17	40	15	2.0	4,000	40	100	20	2,000
18	40	20	2.0	4,000	40	100	20	2,000

settings. For each setting, 10 different mixtures with N samples each were generated as described above. For each mixture, the target log-likelihood was computed from the true GMM parameters. Then, for each mixture, 10 different initializations were obtained as described above, and both the PSO and the EM procedures were run for each initial configuration. The parameters of the global best particle were selected as the final result of each PSO run at the end of the iterations. The final result of each primary EM run was selected as the parameters corresponding to the best secondary run having the highest log-likelihood among the M secondary runs. The estimation error was computed as the difference between the target log-likelihood and the resulting log-likelihood computed from the estimated GMM parameters.

Table 5.4 and Figure 5.6 present the error statistics computed from the 100 runs (10 different mixtures and 10 different initializations for each mixture) for

each setting. When all settings were considered, it could be seen that the proposed PSO algorithm resulted in better estimates compared to those by the EM algorithm for all settings. In particular, the PSO algorithm converged to the true GMM parameters in more than half of the runs for 11 out of 18 settings (all of the 10 *easy* and *medium* settings and one *hard* setting) with a median error of zero, whereas the EM algorithm could do the same for only five settings. For all settings, the average error obtained by the PSO algorithm was significantly lower than the error by the EM algorithm. For the settings with a small number of components, both EM and PSO had no problem in finding the optimum solution. This was mainly due to good initial conditions where it was relatively easier to find a small number of good initial data points that behaved as good initial means. Note that a good initialization for only one of the M secondary runs for each primary EM run was sufficient to report a perfect performance because the best out of M was used.

The above argument could be extended for PSO to all settings relatively independent of the number of dimensions and the number of components. We could conclude that the proposed algorithm was less sensitive to initializations because in every iteration the particles took small number of steps toward one of the local optima using the local EM iterations, and then due to their interaction with the global best, they could move away from that local optimum. We could argue that the common characteristic of the small number of wrong convergences of PSO was the initialization of most of the particles including the global best near the same local optimum. In that case, both the local EM iterations and the global best particle attracted all particles toward the same region. This problem could be eliminated by a more sophisticated initialization procedure that increased the diversity of the particles. However, we used the same initialization procedure that used the same random points for both EM and PSO algorithms to do a fair comparison.

In this thesis, we only investigated the advantages of correspondence identification with regard to finding better global minima of the negative log-likelihood. We showed that stochastic search algorithms performed better in finding global optima. However, correspondence identification can also be useful in increasing

the population diversity. For instance, once we find the correspondence relations via the proposed matching algorithm, we can force the parameters to be updated with the distant (not matching) ones in the global best in some random way to increase the diversity. Another approach may be to temporarily modify the update equations so that the particles move away from the global best if the KL divergence between their personal best and the global best becomes too small in early iterations to overcome premature convergence to a local optimum.

We did not try to tune the parameters of PSO such as η, c_1, and c_2. For different settings, parameter tuning might be useful in terms of increased convergence speed and increased estimation accuracy. However, such tuning could have led to an unfair advantage of PSO over the EM algorithm. We also did not tune the number of particles and the number of iterations except increasing them linearly with increasing dimension. Increasing the number of iterations will not improve the performance of EM after its convergence but larger number of iterations will allow PSO to explore a larger portion of the parameter space. However, the number of iterations were fixed to the same number for EM and PSO to allow a fair comparison.

5.6.2 Experiments on Real Data

We also used four data sets from the UCI Machine Learning Repository [158] for real data experiments. These data sets are referred to as *Glass* (glass identification), *Wine*, *ImgSeg* (Statlog image segmentation), and *Landsat* (Statlog Landsat satellite). Table 5.5 summarizes the characteristics of these data sets and the corresponding experimental settings. For each data set and for each K value, both PSO and EM were run using 10 different initial configurations that were generated as described in the previous Section. The resulting log-likelihood values for each setting for each data set are shown in Figure 5.7. The results show that the proposed PSO algorithm performed better than the EM algorithm for all settings.

Table 5.4: Statistics of the estimation error for the synthetic data sets using the GMM parameters estimated via the EM and PSO procedures. The mean, standard deviation (std), median, and median absolute deviation (mad) are computed from 100 different runs for each setting.

	EM				PSO			
Setting #	mean	std	median	mad	mean	std	median	mad
1	6.18	61.46	0.00	0.00	0.00	0.00	0.00	0.00
2	304.99	183.36	362.71	71.94	41.30	112.55	0.00	0.00
3	66.59	335.93	0.00	0.00	17.42	122.22	0.00	0.00
4	20.32	115.54	0.00	0.00	0.00	0.00	0.00	0.00
5	283.29	135.85	331.03	37.41	27.15	81.98	0.00	0.00
6	500.68	110.17	480.89	78.46	69.80	83.05	0.00	0.00
7	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
8	300.83	174.13	367.08	68.42	11.28	55.66	0.00	0.00
9	654.48	145.67	654.23	163.56	51.39	100.70	0.00	0.00
10	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
11	490.14	307.53	615.89	126.93	112.75	227.90	0.00	0.00
12	842.94	242.63	880.06	192.40	224.89	231.03	97.21	75.14
13	975.60	152.44	912.21	113.53	261.34	98.73	120.66	45.12
14	1,171.30	592.29	1,105.42	205.61	236.63	315.23	102.31	102.70
15	1,651.47	518.35	1,576.24	124.21	309.21	232.49	272.18	58.23
16	2,098.39	460.39	1,971.43	384.08	523.84	183.92	375.28	114.02
17	2,328.13	676.15	2,093.80	403.16	609.92	281.59	412.54	93.84
18	2,946.89	760.48	2,882.77	425.04	697.02	292.17	468.27	100.57

Table 5.5: Details of the real data sets used for performance evaluation. K_{true} corresponds to the number of classes in each data set. K corresponds to the number of Gaussian components used in the experiments. The rest of the parameters are described in the text.

Data set	d	K_{true}	K	N	M	T_1	T_2	$T_1 \times T_2$
Glass	9	6	{ 6, 7, 8, 9, 10 }	214	20	30	20	600
Wine	13	3	{ 3, 4, 5, 6, 7 }	178	30	30	20	600
ImgSeg	19	7	{ 7, 8, 9, 10, 11 }	2,310	30	50	20	1,000
Landsat	36	7	{ 7, 8, 9, 10, 11 }	4,435	40	100	20	2,000

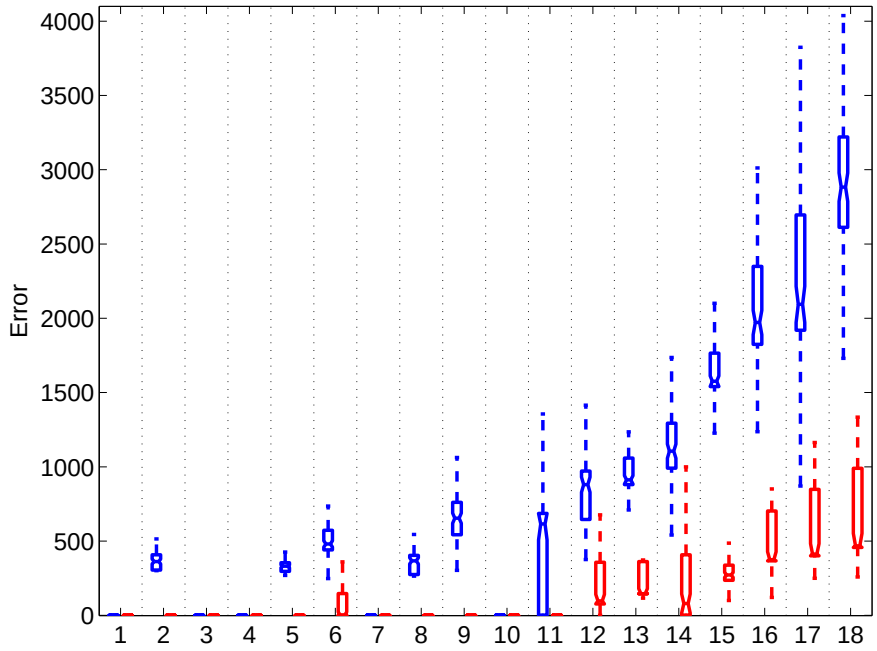
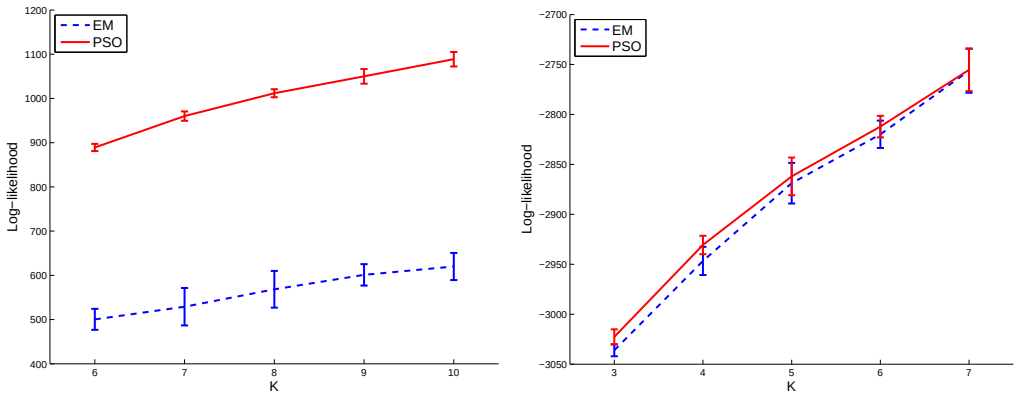


Figure 5.6: Statistics of the estimation error for the synthetic data sets using the GMM parameters estimated via the EM (blue) and PSO (red) procedures. The boxes show the lower quartile, median, and upper quartile of the error. The whiskers drawn as dashed lines extend out to the extreme values.

5.7 Conclusions

In this Chapter, we presented a framework for effective utilization of stochastic search algorithms for the maximum likelihood estimation of Gaussian mixture models. One of the contributions of this work was a covariance parameterization that enabled the use of arbitrary covariance matrices in the search process. The parameterization used eigenvalue decomposition, and modeled each covariance matrix in terms of its eigenvalues and Givens rotation angles extracted from the eigenvector matrices. This parameterization allowed the individual parameters to be independently modifiable so that the resulting matrices remained valid covariance matrices after the stochastic updates. Furthermore, the parameters had bounded ranges so that they could be searched within a finite solution space. We also described an algorithm for ordering the eigenvectors so that the parameters of individual Gaussian components were uniquely identifiable.



(a) *Glass* (b) *Wine*

(c) *ImgSeg* (d) *Landsat*

Figure 5.7: Average log-likelihood and its standard deviation (shown as error bars at one standard deviation) computed from 10 different runs of EM and PSO procedures for the real data sets.

Another contribution of this work was an optimization formulation for resolving the identifiability problem for the mixtures. The proposed solution allowed a unique correspondence between two candidate solutions so that desirable interactions became possible for parameter updates throughout the stochastic search.

We showed that the proposed methods can be used effectively with different stochastic search algorithms such as genetic algorithms, differential evolution, and particle swarm optimization. The final set of experiments using particle swarm optimization with synthetic and real data sets showed that the proposed algorithm could achieve significantly higher likelihood values compared to those obtained by the conventional EM algorithm under the same initial conditions.

Chapter 6

Compound Object Detection

6.1 Introduction

Recently available multispectral channels in very high spatial resolution (VHR) images acquired from new generation satellites have enabled new applications as the increased spectral resolution enhanced the capability to distinguish different physical materials. However, the increased amount of spatial detail in these images also necessitates new advanced algorithms for automatic analysis. For example, the commonly used classification algorithms that require an initial segmentation of the image into homogeneous regions cannot always cope with the increasing complexity because such homogeneous regions often correspond to very small details.

An alternative approach in the recent years has been to model the spatial arrangements of simple image regions to identify complex region groups. Gaetano et al. [159] performed hierarchical texture segmentation assuming that frequent neighboring regions are strongly related. They clustered the image regions to compute the frequencies of quantized region pairs with discrete labels, and used these frequencies to build a segmentation tree where some of the nodes correspond to complex structures. Zamalieva et al. [160] found the significant relations between neighboring regions as the modes of a probability distribution estimated

Figure 6.1: Compound structures in WorldView-2 images of Ankara and Kusadasi, Turkey.

using the continuous features of region co-occurrences. The resulting modes were used to construct the edges of a graph, and a graph mining algorithm was used to find subgraphs that may correspond to compound structures. Vanegas et al. [161] proposed a method based on fuzzy measures of relative direction between objects to detect aligned object groups. They first detected locally aligned groups of three objects, and then checked for global alignment using these local alignments. Akcay and Aksoy [162] described a procedure that combined statistical characteristics of primitive objects modeled using spectral, shape, and position information with structural characteristics encoded using spatial alignments of neighboring similar object groups. However, all of these approaches required an initial segmentation for the identification of the primitive regions. Furthermore, they were designed to detect only a particular type of arrangement such as co-occurrence or alignment.

In this Chapter we describe a new approach that combines statistical and structural characteristics of simple objects to discover compound structures in

VHR images. The compound structures of interest can include different types of residential, commercial, industrial, and agricultural areas that are comprised of spatial arrangements of primitive objects such as buildings, roads, and trees corresponding to locally homogeneous details. The proposed approach uses a probabilistic representation of the compound objects based on constrained Gaussian mixture models introduced in Chapter 3.

In this model, each Gaussian component in the mixture models a group of pixels corresponding to a particular primitive object. Each pixel is represented using a feature vector that encodes both spectral and spatial information consisting of the pixel's multispectral data and its coordinates, respectively. Gaussian components are partioned into two parts: spectral and spatial where the spectral mean corresponds to the color of the object, the spectral covariance corresponds to the homogeneity of the color content, the spatial mean corresponds to the position of the object, and the spatial covariance models its shape.

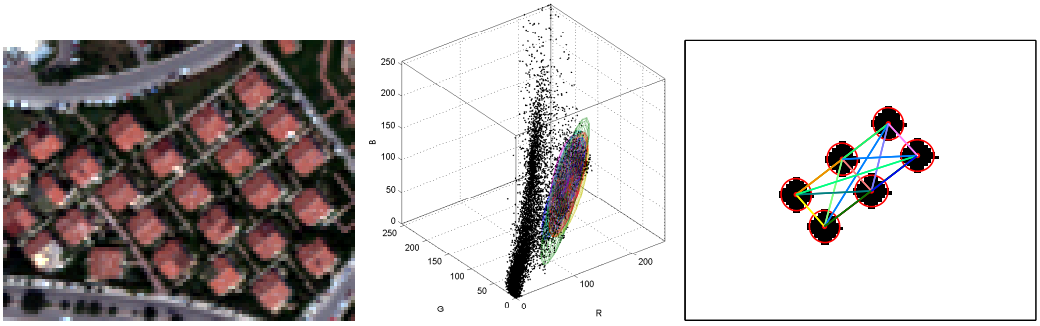
Given example compound structures of interest that are comprised of multiple primitive objects, first, a Gaussian mixture model is fit to the pixels corresponding to the selected structures. This Gaussian mixture model is then used as the reference model in the detection algorithm for identifying the occurrences of other similar compound structures. We describe a novel detection algorithm based on the expectation maximization algorithm for the robust extension of the constrained Gaussian mixture models with known number of inliers as described in Sections 3.4 and 4.2.4. Proposed detection algorithm tries to find the given number of pixels in the new image data that are most similar to the pixels in the reference compound object. Using the language of the Chapter 4, these most similar pixels we are trying to find in the new image are considered to be the inliers and the rest of the pixels are treated as the outliers. The inlier pixels are assumed to be distributed according to a Gaussian mixture model similar to the reference model. Proposed detection algorithm tries to determine both the inlier pixels and the parameters of the new Gaussian mixture model corresponding to the inlier pixels. In this Chapter, we use the source parameterization for the Gaussian mixture models. The new Gaussian mixture model has to satisfy various convex constraints on the source parameters. These constraints are formed using the

parameters of the reference Gaussian mixture model and are described in detail in Section 6.3. The main idea is that these constraints do not allow the spectral parts of the new Gaussian mixture model to be very different from the reference model. On the other hand, spatial parts modeling the locations and the shape of the new primitive objects to be found are allowed to change while preserving the relative location and the size relations given in the reference model. In the detection algorithm, we run the expectation maximization algorithm initialized from different locations on the image data corresponding to the target images. The result is a list of compound structures detected in target images by grouping pixels that have high likelihoods of belonging to the Gaussian object models while satisfying the spatial layout constraints. A very important feature of the proposed model is that it can perform object detection without any requirement of initial segmentation where the only assumption is that the spectral and spatial content of the primitive objects can be modeled in terms of Gaussians.

The rest of the Chapter is organized as follows. Section 6.2 defines the properties of compound structures of interest. Section 6.3 describes the proposed Gaussian mixture model. Section 6.4 presents the detection algorithm. Section 6.5 provides experimental results on an 8-band multispectral WorldView-2 image of Ankara, Turkey. Finally, Section 6.6 lists the conclusions.

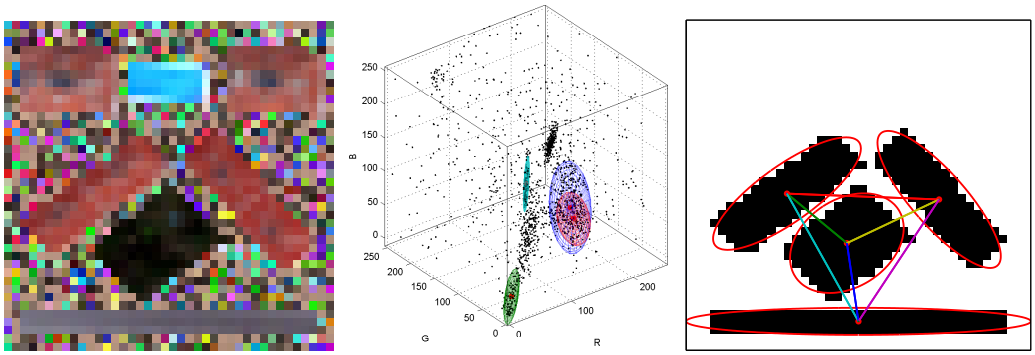
6.2 Definition of Compound Structures

In this thesis, compound structures are defined as high-level heterogeneous objects that are composed of spatial arrangements of multiple, relatively homogeneous, and compact primitive objects. To build the model for these structures, first, each pixel is represented using a d-dimensional feature vector $\mathbf{x} = (\mathbf{x}^{ms}; \mathbf{x}^{xy})$ where $\mathbf{x} \in \mathbb{R}^d$ is formed by concatenating a $d-2$ dimensional vector $\mathbf{x}^{ms}$ containing the multispectral values and a 2-dimensional vector $\mathbf{x}^{xy}$ containing the pixel's coordinates in the image. Since each primitive object is assumed to have a relatively homogeneous spectral content and a compact shape, we further assume that it can be modeled using a Gaussian that is defined in terms of the mean



(a) RGB image (b) Spectral model (c) Spatial model

Figure 6.2: An example model for six buildings in a grid formation.



(a) Synthetic image (b) Spectral model (c) Spatial model

Figure 6.3: An example model for four objects in a synthetic image.

$\mu = (\mu^{ms}; \mu^{xy})$ and the block diagonal covariance matrix $\Sigma = (\Sigma^{ms}, 0; 0, \Sigma^{xy})$ with an additional assumption that the multispectral values and the pixel coordinates are independent, i.e., $p(\mathbf{x}) = p(\mathbf{x}^{ms})p(\mathbf{x}^{xy})$. Given a group of pixels forming the primitive object, the spectral mean μ^{ms} corresponds to the average color of the object, the spectral covariance Σ^{ms} corresponds to the homogeneity of the color content, the spatial mean μ^{xy} corresponds to the position of the object, and the spatial covariance Σ^{xy} models its shape. Figure 6.2 illustrates both the spectral and the spatial parts of the models for example objects.

A compound structure consisting of K primitive objects can then be modeled using a Gaussian mixture model (GMM) expressed in terms of the source

parameters as

$$p(\mathbf{x}|\nu) = \sum_{k=1}^{K} \alpha_k p(\mathbf{x}|\nu_{\mathbf{x}|y=k}) \tag{6.1}$$

that is fully defined by the set of parameters $\nu = \{\alpha_1, \nu_{\mathbf{x}|y=1}, \ldots, \alpha_K, \nu_{\mathbf{x}|y=K}\}$ where each $\nu_{\mathbf{x}|y=k} = \{\mu_k, \Sigma_k\}$ represents the source parameters of the k'th Gaussian component $p(\mathbf{x}|\nu_{\mathbf{x}|y=k})$ that corresponds to the k'th primitive object. $\mu_k \in \mathbb{R}^d$ denotes the mean vector and $\Sigma_k \in \mathbb{S}^d_{++}$ denotes the covariance matrix of the k'th Gaussian component. $\alpha_k \in [0, 1]$ denotes the probability of a pixel belonging to the k'th Gaussian component, and is proportional to the number of pixels, i.e., size, of the corresponding primitive object. The sizes are normalized for the whole compound structure, i.e., $\alpha_1, \ldots, \alpha_K$ are constrained to sum up to 1 as $\sum_{k=1}^{K} \alpha_k = 1$. Since each pixel can belong to one of the K Gaussian components, we also define a corresponding label variable $y_j \in \{1, \ldots, K\}$ for each pixel $j = 1, \ldots, N$ where $y_j = k$ denotes the event of the j'th pixel belonging to the k'th Gaussian component.

The primitive objects can form different compound structures according to different spatial arrangements. In addition to its effectiveness of modeling both the homogeneity and the uncertainty in the spectral and shape content of the primitive objects, the power of the proposed compound structure model comes from its capability of modeling their arrangements. We use a fully connected layout model that is defined in terms of the displacement vectors between the centroids (spatial means) μ^{xy} of the primitive objects. Given K primitive objects, the spatial layout of the compound structure is modeled using a total of $K(K-1)/2$ displacement vectors $\mathbf{d}_{ij}, i = 1, \ldots, K-1, j = i+1, \ldots, K$, where each of these vectors is defined for a particular pair of primitive objects. Figure 6.2(c) shows the layout model of the proposed spatial GMM structure.

6.3 Constrained Gaussian Mixture Model

In the compound object detection problem, we assume that we are given an example compound structure of interest. This input, called the reference structure,

is expected to be in the form of individually delineated regions for the primitive objects. The regions corresponding to the primitive objects can be obtained using basic low-level operations such as morphological opening/closing or image segmentation, or can be obtained via manual selection.

The total of $\tilde{N}$ pixels, $\mathbf{x}_j, j = 1, \ldots, \tilde{N}$, belonging to the reference structure consisting of K primitive objects are used to fit a GMM with K components where each primitive object is modeled by one of the Gaussian components. Since the memberships of all reference pixels to the Gaussian components, $y_j, j = 1, \ldots, \tilde{N}$, are known, the source parameters of the reference GMM can be directly obtained using the maximum likelihood estimates

$$\tilde{\alpha}_k = \frac{1}{\tilde{N}} \sum_{j=1}^{\tilde{N}} \delta(y_j = k) \tag{6.2}$$

$$\tilde{\mu}_k = \frac{1}{\tilde{\alpha}_k \tilde{N}} \sum_{j=1}^{\tilde{N}} \delta(y_j = k) \mathbf{x}_j \tag{6.3}$$

$$\tilde{\Sigma}_k = \frac{1}{\tilde{\alpha}_k \tilde{N}} \sum_{j=1}^{\tilde{N}} \left(\delta(y_j = k) \mathbf{x}_j \mathbf{x}_j^T \right) - \tilde{\mu}_k \tilde{\mu}_k^T \tag{6.4}$$

where $\delta(y_j = k)$ is the Kronecker delta function that gives 1 if $y_j = k$, and 0 otherwise. The resulting reference GMM, $p(\mathbf{x}|\tilde{\nu})$, is defined by its source parameters $\tilde{\nu} = \{\tilde{\alpha}_1, \tilde{\nu}_{\mathbf{x}|y=1}, \ldots, \tilde{\alpha}_K, \tilde{\nu}_{\mathbf{x}|y=K}\}$ where $\tilde{\nu}_{\mathbf{x}|y=k} = \{\tilde{\mu}_k, \tilde{\Sigma}_k\}$, $k = 1, \ldots, K$.

In addition to the GMM source parameters, we also extract the spatial layout of the reference structure in terms of the displacement vectors $\tilde{\mathbf{d}}_{ij}, i = 1, \ldots, K-1, j = i+1, \ldots, K$, that are computed using

$$\tilde{\mu}_i^{xy} + \tilde{\mathbf{d}}_{ij} = \tilde{\mu}_j^{xy}. \tag{6.5}$$

Given a target image with N pixels $\mathbf{x}_j, j = 1, \ldots, N$, the goal is to identify the pixels in this image that are the most similar to those in the reference structure. This can be formulated as a detection problem for the localization of the subregions, i.e., the pixels of interest, that are most likely to correspond to the reference compound object. However, an inherent difficulty in this detection problem is that the pixels of interest, whose number is expected to be similar to

the number of pixels in the reference structure, are typically observed as part of a significantly larger set of observations ($N \gg \tilde{N}$) where the rest of the pixels have an unknown distribution. Using the language of Chapter 4, the pixels of interest can be considered to be the inliers and the rest of the pixels can be treated as the outliers. In this case, for the data points $\mathcal{X}$, we have a set of N hidden inlier Bernoulli variables $\mathcal{O} = \{o_1, \ldots, o_N\}$ where $o_j \in \{0, 1\}$ denotes whether the data point $\mathbf{x}_j$ is an inlier or not denoted by $o_j = 1$, $o_j = 0$, respectively. Furthermore, we assume that the inliers are distributed according to a Gaussian mixture model, i.e., $p(\mathbf{x}|0 = 1)$ is a Gaussian mixture density function. In Section 4.2.4, propostion 9, we have shown that for general robust mixture models if we assume that the posterior distributions of data points being outliers or inliers $\mathcal{R} = \{r(o_1), \ldots, r(o_N)\}$, $r(o_j) = p(o_j|\mathbf{x}_j)$, for $j = 1, \ldots, N$, can take only binary values, i.e., $r(o_j) \in \{0, 1\}$ for $j = 1, \ldots, N$, the number of inliers is a known fixed number $\tilde{N}$, i.e., $\sum_{j=1}^{N} r(o_j = 1) = \tilde{N}$, and the likelihoods of the data points given they are outliers are equal to a constant $\tilde{p}$, i.e., $p(\mathbf{x}_j|o_j = 0) = \tilde{p}$ for $j = 1, \ldots, N$, then we can determine the inliers where $r(o_j = 1) = 1$ by setting $r(o_j = 1) = 1$ for the $\tilde{N}$ biggest $\log p(\mathbf{x}_j|o_j = 1)$ values and $r(o_j = 1) = 0$ for the rest.

The detection process involves the identification of the pixels of interest of the target image modeled with a GMM with K components where K is the same as the number of components in the reference GMM and estimating the target GMM parameters modeling the pixels of interest. The estimation of the parameters of the target GMM, $p(\mathbf{x}|o = 1, \nu)$, that leads to the highest log-likelihood, also uses the reference GMM, $p(\mathbf{x}|\tilde{\nu})$, to form spectral and spatial constraints on the target GMM parameters. Once the target GMM is obtained, the pixels of interest correspond to the ones that are the most likely under the estimated model.

The proposed estimation algorithm is presented in Section 6.4. The algorithm uses the following constraints that are defined between pairs of Gaussian components, one from the reference GMM and the other one from the target GMM.

- We want to keep the relative sizes of the components of reference and target structures the same, i.e., $\alpha_k = \tilde{\alpha}_k$ for $k = 1, \ldots, K$.

- We want the average spectral content of the reference and target components to be similar. Thus, we constrain the multispectral part of each target mean to lie inside a confidence ellipsoid around the reference mean, i.e., $(\mu_k^{ms} - \tilde{\mu}_k^{ms})^T(\tilde{\Sigma}_k^{ms})^{-1}(\mu_k^{ms} - \tilde{\mu}_k^{ms}) \leq \beta$ for $k = 1, \ldots, K$ where β is a constant.

- We also want the homogeneity of the spectral content of the corresponding reference and target components to be the same, i.e., $\Sigma_k^{ms} = \tilde{\Sigma}_k^{ms}$ for $k = 1, \ldots, K$.

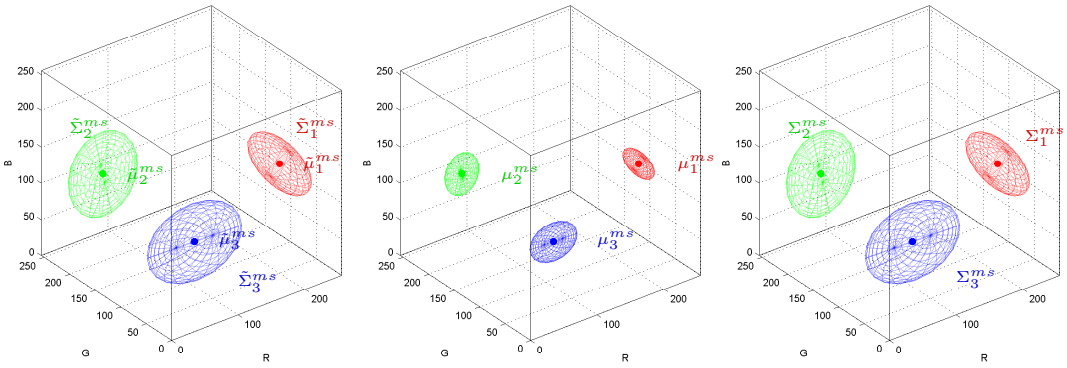
- We want to preserve the spatial layout of the reference structure in the target structure. Thus, given the $K(K-1)/2$ displacement vectors $\tilde{\mathbf{d}}_{ij}, i = 1, \ldots, K-1, j = i+1, \ldots, K$, that are computed between the spatial parts of the reference means as in (6.5), the spatial layout of the target structure is constrained as $\mu_i^{xy} + \tilde{\mathbf{d}}_{ij} - \mu_j^{xy} = \mathbf{t}_{ij}$ where $\|\mathbf{t}_{ij}\|_1 \leq u$ and the constant $u \in \mathbb{R}_+$ specify the allowed amount of deviation from the reference spatial relations.

- Finally, we want the aspect ratio of each reference primitive object to be preserved in the target. Thus, we constrain the minimum and maximum eigenvalues, λ_{min} and λ_{max}, respectively, of the spatial parts of the reference and target covariances to be the same, i.e., $\lambda_{min}(\Sigma_k^{xy}) = \lambda_{min}(\tilde{\Sigma}_k^{xy})$ and $\lambda_{max}(\Sigma_k^{xy}) = \lambda_{max}(\tilde{\Sigma}_k^{xy})$ for $k = 1, \ldots, K$. Note that this allows different rotations of the primitive objects.

The spectral and spatial constraints are illustrated in Figures 6.4 and 6.5, respectively.

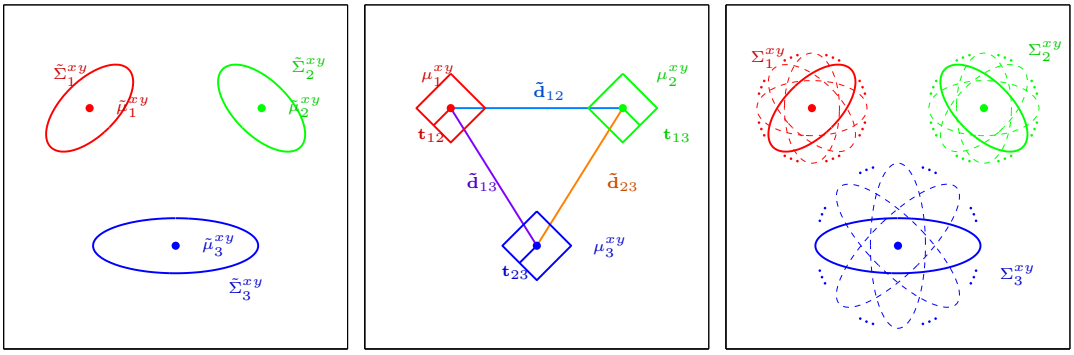
6.4 Detection Algorithm

The input to the detection problem is the reference GMM, $p(\mathbf{x}|\tilde{\nu})$, i.e. estimated from $\tilde{N}$ pixels in the reference compound structure, and a target image containing N pixels, $\mathbf{x}_1, \ldots, \mathbf{x}_N$, among which an unknown subset of size $\tilde{N}$ constitutes the



(a) (b) (c)

Figure 6.4: Spectral constraints. (a) Reference spectral model. (b) Mean constraints: $(\mu_k^{ms} - \tilde{\mu}_k^{ms})^T(\tilde{\Sigma}_k^{ms})^{-1}(\mu_k^{ms} - \tilde{\mu}_k^{ms}) \leq \beta$. (c) Covariance constraints: $\Sigma_k^{ms} = \tilde{\Sigma}_k^{ms}$.



(a) (b) (c)

Figure 6.5: Spatial constraints. (a) Reference spatial model. (b) Mean constraints: $\mu_i^{xy} + \tilde{\mathbf{d}}_{ij} - \mu_j^{xy} = \mathbf{t}_{ij}$, $\|\mathbf{t}_{ij}\|_1 \leq u$ where $\tilde{\mu}_i^{xy} + \tilde{\mathbf{d}}_{ij} = \tilde{\mu}_j^{xy}$. (c) Covariance constraints: $\lambda_{min}(\Sigma_k^{xy}) = \lambda_{min}(\tilde{\Sigma}_k^{xy})$ and $\lambda_{max}(\Sigma_k^{xy}) = \lambda_{max}(\tilde{\Sigma}_k^{xy})$.

pixels of interest. The goal of the detection algorithm is to identify the pixels of interest modeled by the set inlier indicator variables

$$R = \{r(o_1 = 1), \ldots, r(o_N = 1)\}, \;\; r(o_j = 1) \in \{0, 1\} \;\text{ for }\; j = 1, \ldots, N$$

and estimate the parameters of the target GMM, $p(\mathbf{x}|o = 1, \nu)$, that minimizes the negative weighted log-likelihood

$$\{\nu^*, \mathcal{R}^*\} = \arg\min_{\nu, \mathcal{R}} -\frac{1}{\tilde{N}} \sum_{j=1}^{N} r(o_j = 1) \log p(\mathbf{x}_j | o_j = 1, \nu). \tag{6.6}$$

The GMM parameters and the indicator variables can be obtained via the expectation maximization algorithm using the dual problem for the M-step described in Section 3.4.

Let $\mathcal{Q} = \{q(y_1), \ldots, q(y_N)\}$ denote distributions over the label variables The upper bound function $F(\mathcal{R}, \mathcal{Q}, \nu)$ for the negative log-likelihood function $l(\mathcal{R}, \nu)$ can be obtained as

$$\begin{aligned} l(\mathcal{R}, \nu) &\leq -\frac{1}{\tilde{N}} \sum_{j=1}^{N} r(o_j = 1) \left(\sum_{k=1}^{K} q(y_j = k) \log \left(\frac{p(\mathbf{x}_j, y_j = k | o_j = 1, \nu)}{q(y_j = k)} \right) \right) \\ &= F(\mathcal{R}, \mathcal{Q}, \nu). \end{aligned} \tag{6.7}$$

Based on the bound function $F(\mathcal{R}, \mathcal{Q}, \nu)$, we can write the E-step and the dual problem for the M-step of the expectation maximization algorithm as follows.

6.4.1 Expectation Maximization Algorithm

In the E-step, we compute the constrained posterior distributions as follows.

E-step

$$q^t(y_j) = \frac{p(\mathbf{x}_j, y_j | o_j = 1, \nu^{t-1})}{\sum_{i=1}^{K} p(\mathbf{x}_j, y_j = i | o_j = 1, \nu^{t-1})} \text{ for } j = 1, \ldots, N \tag{6.8}$$

$$r^t(o_j = 1) = \begin{cases} 1, & \text{for } \tilde{N} \text{ data points with the highest} \\ & E_{q^t(y_j)}[\log p(\mathbf{x}_j, y_j | o_j = 1, \nu^{t-1})] + H(q^t(y_j)) \\ 0, & \text{o.w.} \end{cases}$$

$$\text{for } j = 1, \ldots, N \tag{6.9}$$

where the index t corresponds to the iteration number.

In the M-step the source parameters $\nu = \{\alpha_1, \mu_1, \Sigma_1, \ldots, \alpha_K, \mu_K, \Sigma_K\}$ that satisfy the constraints defined in Section 6.3 is computed by solving the following dual convex optimization problem.

$$\begin{aligned} \text{maximize} \quad & -\sum_{k=1}^{K-1} \alpha_k \log \alpha_k - (1 - \sum_{k=1}^{K-1} \alpha_k) \log(1 - \sum_{k=1}^{K-1} \alpha_k) \\ & + \sum_{k=1}^{K} \alpha_{sk} \big(\frac{1}{2} \log |\Sigma_k| + \frac{d}{2} \log(2\pi e) \big) \\ & + \sum_{k=1}^{K-1} \alpha_k \eta_{sk} + \sum_{k=1}^{K} \alpha_{sk} \big(\mu_k^T m_{sk} - \frac{1}{2} tr(\Sigma_k S_{sk}) - \frac{1}{2} \mu_k^T S_{sk} \mu_k \big) \\ \text{subject to } & \alpha_k = \tilde{\alpha}_k, \ k = 1, \ldots, K-1 \qquad (6.10) \\ & (\mu_k^{ms} - \tilde{\mu}_k^{ms})^T (\tilde{\Sigma}_k^{ms})^{-1} (\mu_k^{ms} - \tilde{\mu}_k^{ms}) \leq \beta, \\ & \qquad k = 1, \ldots, K, \qquad (6.11) \\ & \mu_i^{xy} + \tilde{\mathbf{d}}_{ij} - \mu_j^{xy} = \mathbf{t}_{ij}, \ \|\mathbf{t}_{ij}\|_1 \leq u, \\ & \qquad i = 1, \ldots, K-1, j = i+1, \ldots, K \qquad (6.12) \\ & \Sigma_k^{ms} = \tilde{\Sigma}_k^{ms}, \ k = 1, \ldots, K, \qquad (6.13) \\ & \lambda_{min}(\tilde{\Sigma}_k^{xy}) \mathbf{I}_2 \preceq \Sigma_k^{xy} \preceq \lambda_{max}(\tilde{\Sigma}_k^{xy}) \mathbf{I}_2, \ k = 1, \ldots, K, \qquad (6.14) \\ & \Sigma_k^i = 0 \text{ for } i \neq ms, \ i \neq xy, \ k = 1, \ldots, K \qquad (6.15) \end{aligned}$$

where $\alpha, \mu_1, \Sigma_1, \ldots, \mu_K, \Sigma_K, \mathbf{t}$ are the optimization variables. Constraints in the

problem above are convex inequality and affine equality constraints in the optimization variables. The expected empirical information parameters are denoted by

$$\begin{aligned}
\eta_{sk} &= \log \frac{\alpha_{sk}}{1 - \sum_{i=1}^{K-1} \alpha_{si}},\ k = 1, \ldots, K-1 \\
m_{sk} &= \Sigma_{sk}^{-1} \mu_{sk},\ k = 1, \ldots, K \\
S_{sk} &= \Sigma_{sk}^{-1},\ k = 1, \ldots, K
\end{aligned}$$

which were calculated apriori after the E-step using the expected empirical probabilities

$$\alpha_{sk} = \frac{1}{\tilde{N}} \sum_{j=1}^{N} r^t(o_j = 1) q^t(y_j = k),\ k = 1, \ldots, K$$

the expected empirical means

$$\mu_{sk} = \frac{1}{\alpha_{sk} \tilde{N}} \sum_{j=1}^{N} r^t(o_j = 1) q^t(y_j = k) \mathbf{x}_j,\ k = 1, \ldots, K$$

and the expected empirical covariance matrices

$$\Sigma_{sk} = \frac{1}{\alpha_{sk} \tilde{N}} \sum_{j=1}^{N} \left(r^t(o_j = 1) q^t(y_j = k) \mathbf{x}_j \mathbf{x}_j^T \right) - \mu_{sk} \mu_{sk}^T,\ k = 1, \ldots, K$$

Here we use $\mathbf{I}_2$ to denote the 2-by-2 identity matrix.

The procedure is run by starting from different initializations of the target GMM on the target image. The algorithm alternates between the E and M steps until an allowed maximum number of iterations is attained or until the difference between the log-likelihood values at two successive iterations falls below some given threshold value. For each initialization, the algorithm gives the GMM parameters and the indicator variables. Each result corresponds to a grouping of the pixels that have high likelihoods of belonging to the reference Gaussian object models while satisfying the spatial layout constraints. The results can be sorted in descending order of the likelihood values, and a list of compound structures detected in the target image can be obtained by truncating this list at a particular likelihood value.

6.5 Experiments

Experiments were performed on an 8-band multispectral WorldView-2 image of Ankara, Turkey with 500×500 pixels and 2 m spatial resolution. The reference compound structures were obtained by manual delineation of the individual primitive objects. The parameters of the reference Gaussian components were obtained using maximum likelihood estimation. In particular, the component probabilities $(\tilde{\alpha}_k, k = 1, \ldots, K)$ were estimated using the ratio of the number of pixels in each primitive object to the total number of pixels in the compound structure, and the means and the covariance matrices were estimated using the pixels belonging to each primitive object. After this supervised step, the rest of the detection process was performed fully unsupervised using the EM algorithm described in section 6.4. Note that, the algorithm does not require any initial segmentation while performing object detection because it can group individual pixels that have high likelihoods of belonging to the Gaussian object models while satisfying the spatial layout constraints.

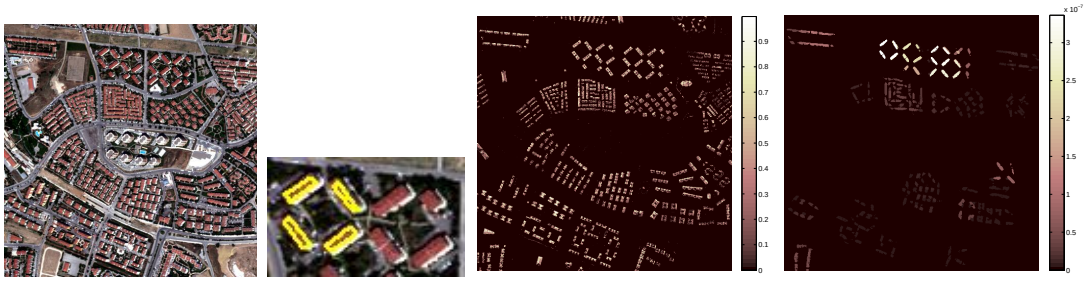
Since each different initialization of the EM algorithm converges to a local optimum of the likelihood function and there is no additional information about the expected locations of similar compound structures in the target image, we used a straightforward initialization procedure using uniform sampling of the image coordinates. First, the reference structure was placed at the top-left corner of the target image. Then, the x and y coordinates were incremented by 25 pixels to form a grid of points that were used as offsets to be added to the centroids of the reference objects for initialization while preserving the displacement relations of the centroids computed from the reference GMM. This resulted in $19 \times 19 = 361$ runs for the EM algorithm. For each run, after calculating the initial centroids using these offset values, the spatial covariances were initialized to the reference GMM's corresponding spatial covariances. Furthermore, the means and covariances corresponding to multispectral values were also initialized to the reference GMM's corresponding means and covariances. Similarly, the Gaussian component probabilities were initialized to reference Gaussian component probabilities. Finally, the number of inliers was set to the total number of pixels in the reference

structure. For all experiments, the number of mixture components was fixed to the number of primitive objects in the reference structure.

Fig. 6.6 shows an example structure composed of four buildings with red roofs placed in a diamond formation. The resulting target GMMs obtained after the convergence of the EM algorithm for each of the 361 runs were ranked according to the resulting likelihood values.Fig. 6.7 shows the top sixteen structures that corresponded to the highest likelihood values. The spatial layout model and the constraints defined in sections 6.2 and 6.3, respectively, allow the individual Gaussian components to rotate around their centroids while preserving the relative displacements computed from the reference GMM. Therefore, some of the detected structures corresponded to formations by rotated buildings (e.g., cross-like formation of four buildings, and parallel groups of two buildings) where the displacements between pairwise centroids were always very similar to those in the reference structure because of the constraints used.

Fig. 6.8 shows another example structure corresponding to an intersection of four road segments. Similar to the previous example, the resulting target GMMs obtained after the convergence of the EM algorithm for each of the 361 runs were ranked according to the resulting likelihood values. Fig. 6.9 shows the top eight structures that corresponded to the highest likelihood values. All results except the third one corresponded to intersections that were similar to the reference structure. The third result shows an interesting case where nearby road segments formed a different structure because of the allowed rotations around the centroids with almost identical displacement. Additional constraints can be used to restrict or relax both the appearances and the spatial layout of the primitive objects within the compound structures of interest.

Fig. 6.10 and Fig. 6.11 show another example structure composed of four buildings and a pool. Similar to the previous examples, the resulting target GMMs obtained after the convergence of the EM algorithm.



(a) RGB image (b) Reference structure (c) Unconstrained GMM result (d) Constrained GMM result

Figure 6.6: Detection of an example structure composed of four buildings with red roofs in a diamond formation in a multispectral WorldView-2 image of Ankara. (a) shows the RGB image formed by the visible bands. (b) shows a close up of the four patches, that were manually delineated as primitive objects, overlayed on the RGB image as yellow polygons. (c) shows the likelihood results obtained with unconstrained GMM. (d) shows the likelihood results obtained with the proposed constrained GMM model

6.6 Conclusions

We presented a new Gaussian mixture model that uses the individual Gaussian components to represent the spectral and shape contents of basic primitive objects, and proposed a new expectation-maximization algorithm that can incorporate spectral and spatial constraints for the detection of compound structures that are comprised of spatial arrangements of such objects. Given an example compound structure of interest, first, a reference GMM was estimated from the pixels belonging to the manually delineated primitive objects. Then, the EM algorithm was used to fit a robust GMM to the target image data so that the pixels that had high likelihoods of belonging to the Gaussian object models and satisfied the spatial layout constraints could be grouped to perform unsupervised object detection.

The initial experiments showed that the proposed method can detect high-level structures that cannot be modeled using traditional techniques. Furthermore, it has a very important advantage of not requiring any initial segmentation while performing object detection by grouping individual pixels. In the proof-of-concept experiments presented in this Chapter, all primitive objects corresponded

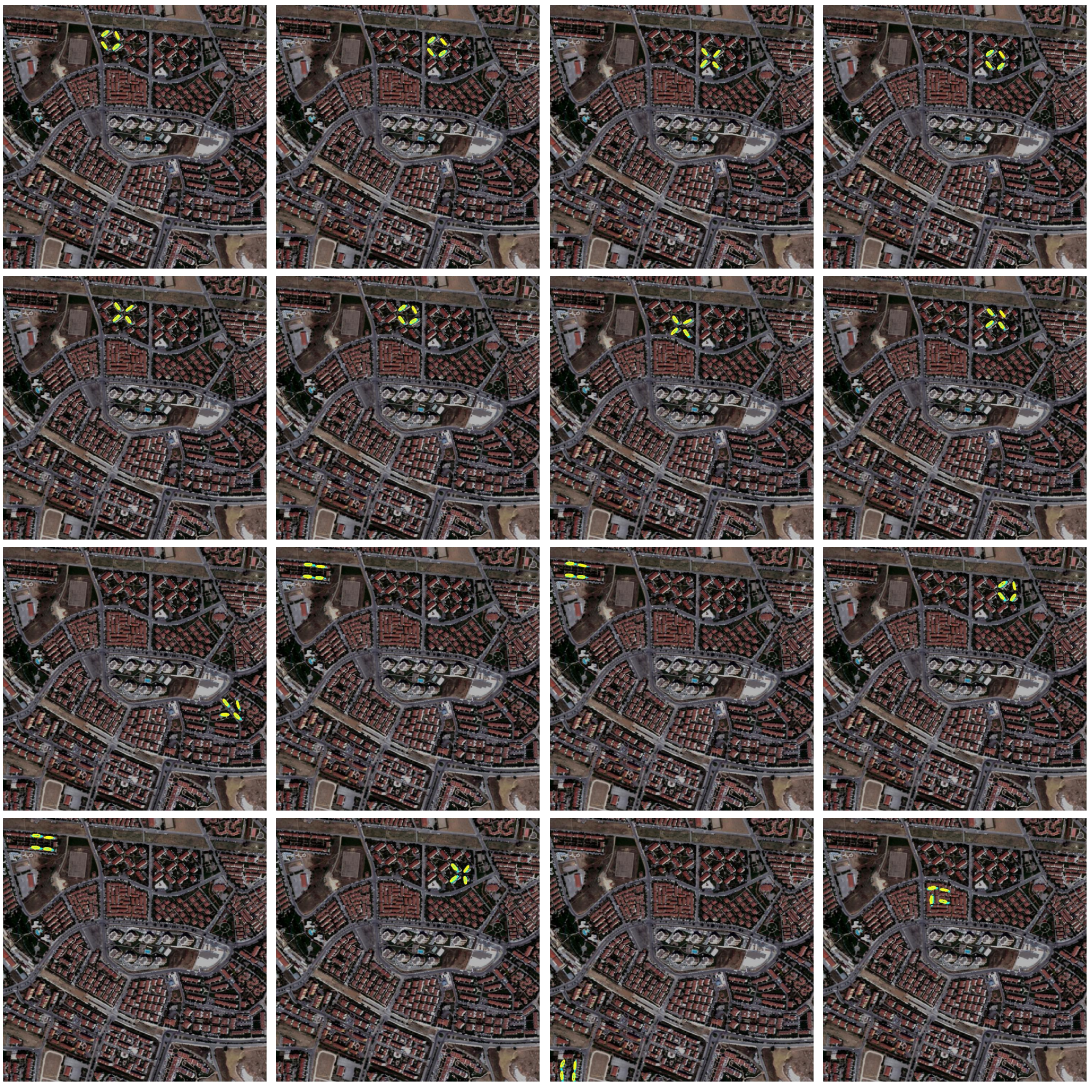
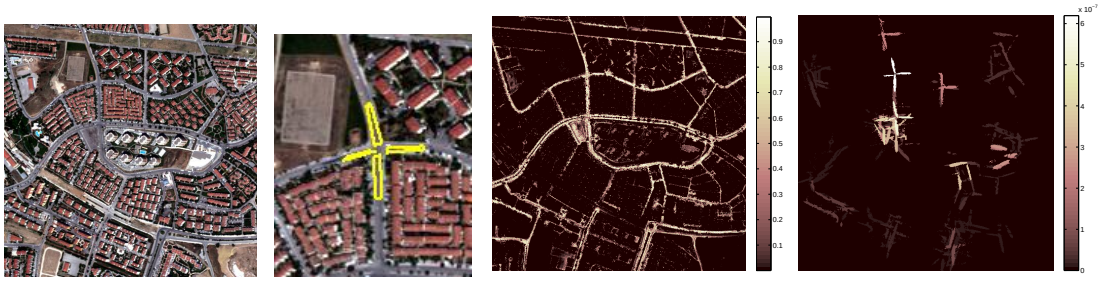


Figure 6.7: The top 16 structures that corresponded to the highest likelihood values at the end of all runs of the EM algorithm. For each result, the pixels selected as inliers are marked in cyan, and the resulting Gaussians are overlayed as yellow ellipses drawn at three standard deviations.



(a) RGB image (b) Reference structure (c) Unconstrained GMM result (d) Constrained GMM result

Figure 6.8: Detection of an example structure corresponding to an intersection of four road segments in a multispectral WorldView-2 image of Ankara. (a) shows the RGB image formed by the visible bands. (b) shows a close up of the four patches, that were manually delineated as primitive objects, overlayed on the RGB image as yellow polygons. (c) shows the likelihood results obtained with unconstrained GMM. (d) shows the likelihood results obtained with the proposed constrained GMM model.

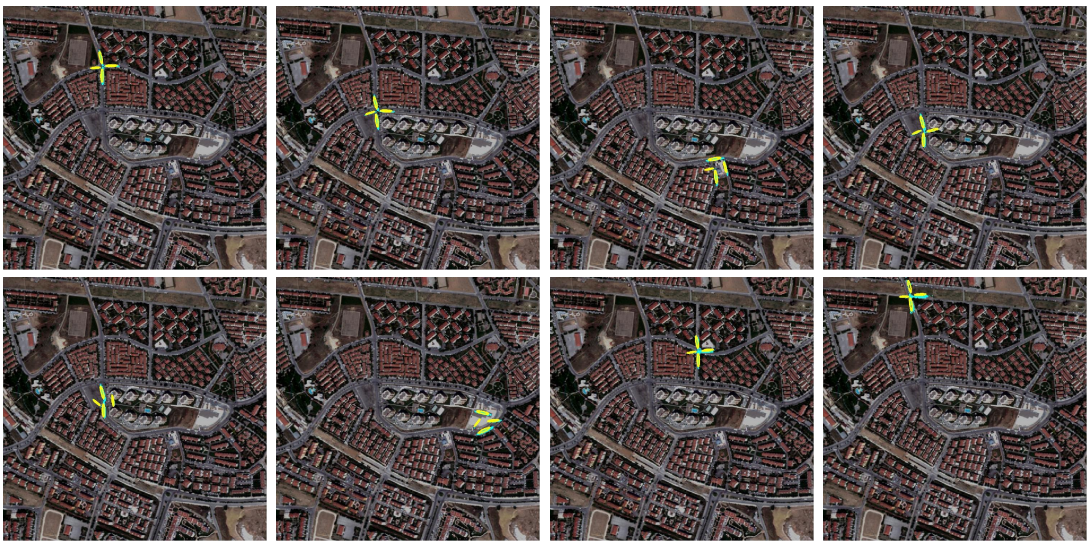
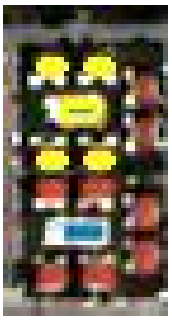
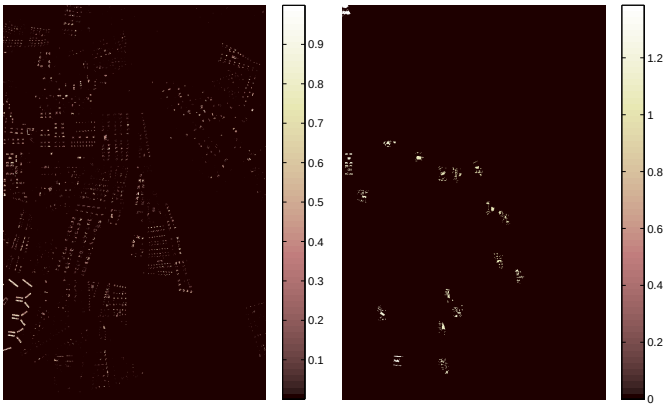


Figure 6.9: The top eight structures that corresponded to the highest likelihood values at the end of all runs of the EM algorithm. For each result, the pixels selected as inliers are marked in cyan, and the resulting Gaussians are overlayed as yellow ellipses drawn at three standard deviations.

(a) RGB image



(b) Reference structure

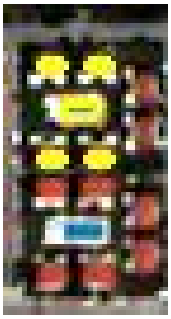


(c) Unconstrained GMM result

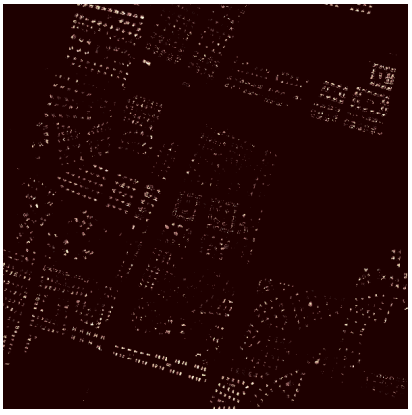
(d) Constrained GMM result

Figure 6.10: Detection of an example structure composed of four buildings and a pool in a multispectral WorldView-2 image of Kusadasi.

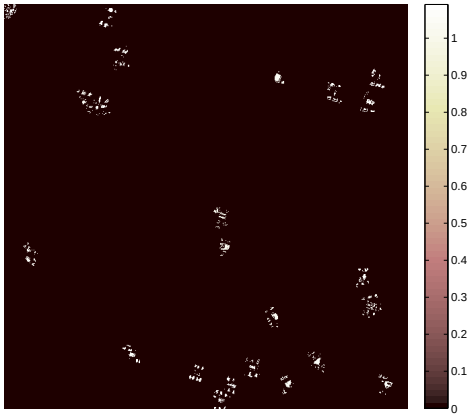
(a) RGB image



(b) Reference structure



(c) Unconstrained GMM result



(d) Constrained GMM result

Figure 6.11: Detection of an example structure composed of four buildings and a pool in another multispectral WorldView-2 image of Kusadasi.

to the same type, i.e., buildings in Fig. 6.6 and roads in Fig. 6.8, but the algorithm can use any type of primitive object. Therefore, future work includes experiments with other types of compound structures in larger data sets. We are also planning to extend the model with additional constraints to handle the scale and orientation changes.

Chapter 7

Conclusions and Future Work

In this thesis, a novel framework called constrained Gaussian mixture models where convex constraints either on the information or the source parameters can be handled by solving constrained convex optimization problems for the M-step of the expectation maximization algorithm was presented. This framework provides a mathematically principled way to handle convex constraints on both the information and the source parameters for Gaussian mixture models.

Second, a new probabilistic model for the robust estimation of the Gaussian mixture models was proposed. We showed that we can incorporate the inlier/outlier information available for small number of data points as convex constraints on the information parameters. This model allows us to estimate the information parameters consistent with available inlier/outlier information. Furthermore, using available inlier/outlier information we showed that we can also determine a threshold level for outlier detection.

Third, novel parameterization based on eigenvalue decomposition of covariance matrices suitable for stochastic search algorithms was developed. A new algorithm where global search skills of the PSO algorithm and the local search skills of the expectation maximization algorithm can be exploited to do global parameter estimation was presented.

Fourth, a compound object detection algorithm as an application to the robust constrained Gaussian mixture models was developed. We showed that various prior information about the objects can be effectively modeled using convex constraints on the source parameters.

The unifying idea in this thesis was that various prior information available for the problem can be incorporated in the form of convex constraints either on the source or the information parameters for the Gaussian mixture models and we can handle these constraints by solving constrained convex optimization problems for the M-step of the expectation maximization algorithm.

We showed that constrained robust Gaussian mixture models can be successfully used for data analysis and object detection. Improving the capabilities of the proposed models, searching for new applications and developing specialized convex optimization solvers for specific applications can be directions for future research.

Bibliography

[1] R. Fisher, *Statistical Methods and Scientific Inference*. Hafner Publishing Co., 1956.

[2] Y. Pawitan, *In All Likelihood: Statistical Modelling and Inference Using Likelihood*. OUP Oxford, 2001.

[3] D. Koller and N. Friedman, *Probabilistic Graphical Models: Principles and Techniques*. MIT Press, 2009.

[4] E. Jaynes, "Information theory and statistical mechanics," *Physical review*, vol. 106, no. 4, p. 620, 1957.

[5] N. Wu, *The Maximum Entropy Method*. Springer, 1997.

[6] I. Good, "Maximum entropy for hypothesis formulation, especially for multidimensional contingency tables," *The Annals of Mathematical Statistics*, vol. 34, no. 3, pp. 911–934, 1963.

[7] S. Della Pietra, V. Della Pietra, and J. Lafferty, "Inducing features of random fields," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 19, no. 4, pp. 380–393, 1997.

[8] T. Cover and J. Thomas, *Elements of Information Theory*. Wiley-interscience, 2006.

[9] S. Phillips, M. Dudík, and R. Schapire, "A maximum entropy approach to species distribution modeling," in *Proceedings of the Twenty-first International Conference on Machine Learning*, p. 83, ACM, 2004.

[10] M. Dudík and R. Schapire, "Maximum entropy distribution estimation with generalized regularization," *Learning Theory*, pp. 123–138, 2006.

[11] M. Dudık, S. Phillips, and R. Schapire, "Maximum entropy density estimation with generalized regularization and an application to species distribution modeling," *Journal of Machine Learning Research*, vol. 8, pp. 1217–1260, 2007.

[12] J. Johnson, V. Chandrasekaran, and A. Willsky, "Learning markov structure by maximum entropy relaxation," *Artificial Intelligence and Statistics*, 2007.

[13] J. Johnson, *Convex relaxation methods for graphical models: Lagrangian and maximum entropy approaches*. PhD thesis, Massachusetts Institute of Technology, 2008.

[14] A. Erkan, *Semi-supervised learning via generalized maximum entropy*. PhD thesis, New York University, 2010.

[15] R. Redner and H. Walker, "Mixture densities, maximum likelihood and the em algorithm," *SIAM review*, vol. 26, no. 2, pp. 195–239, 1984.

[16] D. Titterington, A. Smith, and U. Makov, *Statistical Analysis of Finite Mixture Distributions*. Wiley New York, 1985.

[17] C. Bishop, *Pattern Recognition and Machine Learning*. Springer-Verlag New York, Inc., Secaucus, NJ, 2006.

[18] R. Duda, P. Hart, and D. Stork, *Pattern Classification*. Wiley, 2001.

[19] G. McLachlan and D. Peel, *Finite Mixture Models*. Wiley-Interscience, 2000.

[20] S. Lloyd, "Least squares quantization in pcm," *IEEE Transactions on Information Theory*, vol. 28, no. 2, pp. 129–137, 1982.

[21] L. Rabiner, "A tutorial on hidden markov models and selected applications in speech recognition," *Proceedings of the IEEE*, vol. 77, no. 2, pp. 257–286, 1989.

[22] L. Rabiner and B. Juang, *Fundamentals of Speech Recognition*. Prentice Hall, 1993.

[23] F. Jelinek, *Statistical Methods for Speech Recognition*. MIT Press, 1998.

[24] T. Pappas, "An adaptive clustering algorithm for image segmentation," *IEEE Transactions on Signal Processing*, vol. 40, no. 4, pp. 901–914, 1992.

[25] N. Friedman and S. Russell, "Image segmentation in video sequences: A probabilistic approach," in *Proceedings of the Thirteenth Conference on Uncertainty in Artificial Intelligence*, pp. 175–181, Morgan Kaufmann Publishers Inc., 1997.

[26] C. Stauffer and W. Grimson, "Adaptive background mixture models for real-time tracking," *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, vol. 2, 1999.

[27] M. Harville, "A framework for high-level feedback to adaptive, per-pixel, mixture-of-gaussian background models," *European Conference on Computer Vision*, pp. 37–49, 2002.

[28] X. Yang and S. Krishnan, "Image segmentation using finite mixtures and spatial information," *Image and Vision Computing*, vol. 22, no. 9, pp. 735–745, 2004.

[29] J. Cheng, J. Yang, Y. Zhou, and Y. Cui, "Flexible background mixture models for foreground segmentation," *Image and Vision Computing*, vol. 24, no. 5, pp. 473–482, 2006.

[30] S. Sanjay-Gopal and T. Hebert, "Bayesian pixel classification using spatially variant finite mixtures and the generalized em algorithm," *IEEE Transactions on Image Processing*, vol. 7, no. 7, pp. 1014–1028, 1998.

[31] C. Nikou, N. Galatsanos, and A. Likas, "A class-adaptive spatially variant mixture model for image segmentation," *IEEE Transactions on Image Processing*, vol. 16, no. 4, pp. 1121–1130, 2007.

[32] B. Moghaddam and A. Pentland, "Probabilistic visual learning for object representation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 19, no. 7, pp. 696–710, 1997.

[33] D. Forsyth and J. Ponce, *Computer Vision: A Modern Approach*. Prentice Hall, 2002.

[34] A. Ben-Tal and A. Nemirovski, *Lectures on Modern Convex Optimization: Analysis, Algorithms, and Engineering Applications*. Society For Industrial Mathematics, 1987.

[35] A. Ben-Tal and A. Nemirovski, "Convex optimization in engineering: Modeling, analysis, algorithms." `http://ssor.twi.tudelft.nl/prodanov/frame3.htm`, 1998.

[36] S. Boyd and L. Vandenberghe, *Convex Optimization*. Cambridge University Press, 2004.

[37] A. Dempster, N. Laird, and D. Rubin, "Maximum likelihood from incomplete data via the em algorithm," *Journal of the Royal Statistical Society: Series B (Methodological)*, pp. 1–38, 1977.

[38] R. Neal and G. Hinton, "A view of the em algorithm that justifies incremental, sparse, and other variants," in *Learning in Graphical Models* (M. Jordan, ed.), pp. 355–370, Kluwer Academic Publishers, 1998.

[39] G. McLachlan and T. Krishnan, *The EM Algorithm and Extensions*. Wiley-Interscience, 2007.

[40] A. Dempster, "Covariance selection," *Biometrics*, pp. 157–175, 1972.

[41] M. Wainwright and M. Jordan, "A variational principle for graphical models," *New Directions in Statistical Signal Processing*, p. 155, 2005.

[42] M. Wainwright and M. Jordan, "Graphical models, exponential families, and variational inference," *Foundations and Trends® in Machine Learning*, vol. 1, no. 1-2, pp. 1–305, 2008.

[43] Z. Ghahramani, M. Beal, *et al.*, "Graphical models and variational methods," *Advanced Mean Field Method: Theory and Practice*, 2000.

[44] M. Jordan, Z. Ghahramani, T. Jaakkola, and L. Saul, "An introduction to variational methods for graphical models," *Machine Learning*, vol. 37, no. 2, pp. 183–233, 1999.

[45] S. Waterhouse, D. MacKay, T. Robinson, *et al.*, "Bayesian methods for mixtures of experts," *Advances in Neural Information Processing Systems*, pp. 351–357, 1996.

[46] H. Attias, "Inferring parameters and structure of latent variable models by variational bayes," in *Proceedings of the Fifteenth Conference on Uncertainty in Artificial Intelligence*, pp. 21–30, Morgan Kaufmann Publishers, 1999.

[47] D. Edwards, *Introduction to Graphical Modelling*. Springer, 2000.

[48] N. Meinshausen and P. Bühlmann, "High-dimensional graphs and variable selection with the lasso," *The Annals of Statistics*, vol. 34, no. 3, pp. 1436–1462, 2006.

[49] J. Dahl, L. Vandenberghe, and V. Roychowdhury, "Covariance selection for nonchordal graphs via chordal embedding," *Optimization Methods & Software*, vol. 23, no. 4, pp. 501–520, 2008.

[50] O. Banerjee, L. El Ghaoui, and A. d'Aspremont, "Model selection through sparse maximum likelihood estimation for multivariate gaussian or binary data," *Journal of Machine Learning Research*, vol. 9, pp. 485–516, 2008.

[51] J. Friedman, T. Hastie, and R. Tibshirani, "Sparse inverse covariance estimation with the graphical lasso," *Biostatistics*, vol. 9, no. 3, pp. 432–441, 2008.

[52] C. Hsieh, I. Dhillon, P. Ravikumar, and A. Banerjee, "A divide-and-conquer method for sparse inverse covariance estimation," *Advances in Neural Information Processing Systems*, pp. 2339–2347, 2012.

[53] M. Gales, "Semi-tied covariance matrices for hidden markov models," *IEEE Transactions on Speech and Audio Processing*, vol. 7, no. 3, pp. 272–281, 1999.

[54] A. Butte, P. Tamayo, D. Slonim, T. Golub, and I. Kohane, "Discovering functional relationships between rna expression and chemotherapeutic susceptibility using relevance networks," *Proceedings of the National Academy of Sciences*, vol. 97, no. 22, pp. 12182–12186, 2000.

[55] E. Sudderth, A. Torralba, W. Freeman, and A. Willsky, "Learning hierarchical models of scenes, objects, and parts," in *Proceedings of the IEEE International Conference on Computer Vision*, vol. 2, pp. 1331–1338, 2005.

[56] Y. Mao, F. Kschischang, and B. Frey, "Convolutional factor graphs as probabilistic models," in *Proceedings of the 20th Conference on Uncertainty in Artificial Intelligence*, pp. 374–381, AUAI Press, 2004.

[57] M. Grzebyk, P. Wild, and D. Chouanière, "On identification of multi-factor models with correlated residuals," *Biometrika*, vol. 91, no. 1, pp. 141–151, 2004.

[58] S. Chaudhuri, M. Drton, and T. Richardson, "Estimation of a covariance matrix with zeros," *Biometrika*, vol. 94, no. 1, pp. 199–216, 2007.

[59] T. Anderson, "Statistical inference for covariance matrices with linear structure," in *Multivariate Analysis, II (Proc. Second Internat. Sympos., Dayton, Ohio, 1968)*, pp. 55–66, 1969.

[60] T. Anderson, "Estimation of covariance matrices which are linear combinations or whose inverses are linear combinations of given matrices," *Essays in Probability and Statistics*, pp. 1–24, 1970.

[61] T. Anderson and I. Olkin, "Maximum-likelihood estimation of the parameters of a multivariate normal distribution," *Linear Algebra and Its Applications*, vol. 70, pp. 147–171, 1985.

[62] C. Leggetter and P. Woodland, "Maximum likelihood linear regression for speaker adaptation of continuous density hidden markov models," *Computer Speech and Language*, pp. 171–185, 1995.

[63] D. Reynolds, "Automatic speaker recognition using gaussian mixture speaker models," in *The Lincoln Laboratory Journal*, 1995.

[64] D. Reynolds and R. Rose, "Robust text-independent speaker identification using gaussian mixture speaker models," *IEEE Transactions on Speech and Audio Processing*, vol. 3, no. 1, pp. 72–83, 1995.

[65] V. Digalakis, D. Rtischev, and L. Neumeyer, "Speaker adaptation using constrained estimation of gaussian mixtures," *IEEE Transactions on Speech and Audio Processing*, vol. 3, no. 5, pp. 357–366, 1995.

[66] M. Gales, "Maximum likelihood linear transformations for hmm-based speech recognition," *Computer Speech and Language*, vol. 12, no. 2, 1998.

[67] D. Reynolds, T. Quatieri, and R. Dunn, "Speaker verification using adapted gaussian mixture models," *Digital Signal Processing*, vol. 10, no. 1, pp. 19–41, 2000.

[68] D. Povey, "A tutorial-style introduction to subspace gaussian mixture models for speech recognition," *Microsoft Research, Redmond, WA*, 2009.

[69] C. Leggetter and P. Woodland, "Flexible speaker adaptation for large vocabulary speech recognition," in *Eurospeech Proceedings: 4th European Conference on Speech Communication and Technology*, vol. 2, pp. 1155–1158, 1995.

[70] C. Leggetter and P. Woodland, "Maximum likelihood linear regression for speaker adaptation of continuous density hidden markov models," *Computer Speech and Language*, vol. 9, no. 2, p. 171, 1995.

[71] M. Gales and P. Woodland, "Mean and variance adaptation within the mllr framework," *Computer Speech and Language*, vol. 10, no. 4, pp. 249–264, 1996.

[72] Ç. Arı and S. Aksoy, "Detection of compound structures using a gaussian mixture model with spectral and spatial constraints," in *Proceedings of SPIE Defense, Security, and Sensing: Algorithms and Technologies for Multispectral, Hyperspectral, and Ultraspectral Imagery XVIII*, (Baltimore, Maryland), April 23–27, 2012.

[73] D. Arthur and S. Vassilvitskii, "k-means++: The advantages of careful seeding," in *Proceedings of the Eighteenth Annual ACM-SIAM Symposium on Discrete Algorithms*, pp. 1027–1035, Society for Industrial and Applied Mathematics, 2007.

[74] A. Engelbrecht, *Computational Intelligence: An Introduction*. Wiley, 2007.

[75] P. Wang, *Computational Intelligence in Economics and Finance*. Springer, 2010.

[76] Y. Tenne and C. Goh, *Computational Intelligence in Optimization: Applications and Implementations*. Springer, 2010.

[77] D. Goldberg, *Genetic Algorithms in Search, Optimization, and Machine Learning*. Addison-Wesley Professional, 1989.

[78] L. Davis, *Handbook of Genetic Algorithms*. Van Nostrand Reinhold New York, 1991.

[79] M. Gen and R. Cheng, *Genetic Algorithm and Engineering Optimization*. Wiley, 2000.

[80] H. Adeli and K. Sarma, *Cost Optimization of Structures: Fuzzy Logic, Genetic Algorithms, and Parallel Computing*. Wiley, 2006.

[81] D. GA and S. Okdem, "A simple and global optimization algorithm for engineering problems: differential evolution algorithm," *Turk J Elec Engin*, vol. 12, no. 1, 2004.

[82] R. Storn, K. Price, and J. Lampinen, *Differential Evolution: A Practical Approach to Global Optimization*. Springer, Berlin, 2005.

[83] J. Kennedy and R. Eberhart, "Particle swarm optimization," in *Proceedings of the IEEE International Conference on Neural Networks*, vol. 4, pp. 1942–1948, 1995.

[84] Y. Shi and R. Eberhart, "Particle swarm optimization: developments, applications and resources," in *Proceedings of the IEEE Congress on Evolutionary Computation*, vol. 1, pp. 81–86, 2001.

[85] P. Schroeter, J. Vesin, T. Langenberger, and R. Meuli, "Robust parameter estimation of intensity distributions for brain magnetic resonance images," *IEEE Transactions on Medical Imaging*, vol. 17, no. 2, pp. 172–186, 1998.

[86] A. Martınez and J. Vitria, "Learning mixture models using a genetic version of the em algorithm," *Pattern Recognition Letters*, vol. 21, no. 8, pp. 759–769, 2000.

[87] F. Pernkopf and D. Bouchaffra, "Genetic-based em algorithm for learning gaussian mixture models," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 27, no. 8, pp. 1344–1348, 2005.

[88] J. Tohka, E. Krestyannikov, I. Dinov, A. Graham, D. Shattuck, U. Ruotsalainen, and A. Toga, "Genetic algorithms for finite mixture model based voxel classification in neuroimaging," *IEEE Transactions on Medical Imaging*, vol. 26, no. 5, pp. 696–711, 2007.

[89] D. Chang, X. Zhang, and C. Zheng, "A genetic algorithm with gene rearrangement for k-means clustering," *Pattern Recognition*, vol. 42, no. 7, pp. 1210–1222, 2009.

[90] U. Maulik and I. Saha, "Modified differential evolution based fuzzy clustering for pixel classification in remote sensing imagery," *Pattern Recognition*, vol. 42, no. 9, pp. 2135–2149, 2009.

[91] M. Omran, A. Engelbrecht, and A. Salman, "Particle swarm optimization method for image clustering," *International Journal of Pattern Recognition and Artificial Intelligence*, vol. 19, no. 03, pp. 297–321, 2005.

[92] A. Abraham, S. Das, and S. Roy, "Swarm intelligence algorithms for data clustering," *Soft Computing for Knowledge Discovery and Data Mining*, pp. 279–313, 2008.

[93] A. Paoli, F. Melgani, and E. Pasolli, "Clustering of hyperspectral images based on multiobjective particle swarm optimization," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 47, no. 12, pp. 4175–4188, 2009.

[94] S. Kiranyaz, T. Ince, A. Yildirim, and M. Gabbouj, "Fractional particle swarm optimization in multidimensional search space," *IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics*, vol. 40, no. 2, pp. 298–319, 2010.

[95] G. Celeux, M. Hurn, and C. Robert, "Computational and inferential difficulties with mixture posterior distributions," *Journal of the American Statistical Association*, vol. 95, no. 451, pp. 957–970, 2000.

[96] M. Stephens, "Dealing with label switching in mixture models," *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, vol. 62, no. 4, pp. 795–809, 2000.

[97] M. Sperrin, T. Jaki, and E. Wit, "Probabilistic relabelling strategies for the label switching problem in bayesian mixture models," *Statistics and Computing*, vol. 20, no. 3, pp. 357–366, 2010.

[98] Ç. Arı and S. Aksoy, "Unsupervised classification of remotely sensed images using gaussian mixture models and particle swarm optimization," in *Proceedings of the IEEE International Geoscience and Remote Sensing Symposium*, (Honolulu, Hawaii), pp. 1859–1862, July 25–30, 2010.

[99] Ç. Arı and S. Aksoy, "Maximum likelihood estimation of gaussian mixture models using particle swarm optimization," in *Proceedings of 20th IAPR International Conference on Pattern Recognition*, (Istanbul, Turkey), pp. 746–749, August 23–26, 2010.

[100] Ç. Arı, S. Aksoy, and O. Arıkan, "Maximum likelihood estimation of gaussian mixture models using stochastic search," *Pattern Recognition*, vol. 45, pp. 2804–2816, July 2012.

[101] M. Fischler and R. Elschlager, "The representation and matching of pictorial structures," *IEEE Transactions on Computers*, vol. 100, no. 1, pp. 67–92, 1973.

[102] Y. Amit and D. Geman, "A computational model for visual selection," *Neural Computation*, vol. 11, no. 7, pp. 1691–1715, 1999.

[103] H. Schneiderman and T. Kanade, "A statistical method for 3d object detection applied to faces and cars," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, vol. 1, pp. 746–751, 2000.

[104] M. Burl, M. Weber, and P. Perona, "A probabilistic approach to object recognition using local photometry and global geometry," *European Conference on Computer Vision*, pp. 628–641, 1998.

[105] M. Weber, M. Welling, and P. Perona, "Unsupervised learning of models for recognition," *European Conference on Computer Vision*, pp. 18–32, 2000.

[106] P. Felzenszwalb and D. Huttenlocher, "Pictorial structures for object recognition," *International Journal of Computer Vision*, vol. 61, no. 1, pp. 55–79, 2005.

[107] P. Felzenszwalb and D. Huttenlocher, "Efficient matching of pictorial structures," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, vol. 2, pp. 66–73, 2000.

[108] D. Lowe, "Object recognition from local scale-invariant features," in *Proceedings of the IEEE International Conference on Computer Vision*, vol. 2, pp. 1150–1157, 1999.

[109] D. Lowe, "Distinctive image features from scale-invariant keypoints," *International Journal of Computer Vision*, vol. 60, no. 2, pp. 91–110, 2004.

[110] T. Kadir and M. Brady, "Saliency, scale and image description," *International Journal of Computer Vision*, vol. 45, no. 2, pp. 83–105, 2001.

[111] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, vol. 1, pp. 886–893, 2005.

[112] O. Barndorff-Nielsen, *Information and Exponential Families: In Statistical Theory*. Wiley New York, 1978.

[113] B. Efron, "The geometry of exponential families," *The Annals of Statistics*, vol. 6, no. 2, pp. 362–376, 1978.

[114] N. Chentsov, "A systematic theory of exponential families of probability distributions," *Theory of Probability and Its Applications*, vol. 11, no. 3, pp. 425–435, 1966.

[115] S. Amari and H. Nagaoka, *Methods of Information Geometry*. Amer Mathematical Society, 2007.

[116] Z. Luo and W. Yu, "An introduction to convex optimization for communications and signal processing," *IEEE Journal on Selected Areas in Communications*, vol. 24, no. 8, pp. 1426–1438, 2006.

[117] M. Bazaraa, J. Jarvis, H. Sherali, and M. Bazaraa, *Linear Programming and Network Flows*. Wiley, 1990.

[118] M. Bazaraa, H. Sherali, and C. Shetty, *Nonlinear Programming: Theory and Algorithms*. Wiley, 2006.

[119] S. Rao, *Engineering Optimization: Theory and Practice*. Wiley, 2009.

[120] D. Maringer, *Portfolio Management with Heuristic Optimization*. Springer, 2005.

[121] D. Corne, M. Oates, D. Smith, and J. Wiley, *Telecommunications Optimization: Heuristics and Adaptive Techniques*. Wiley, 2000.

[122] M. Resende and P. Pardalos, *Handbook of Optimization in Telecommunications*. Springer, 2006.

[123] D. Anderson, D. Sweeney, T. Williams, and R. Martin, *An Introduction to Management Science: Quantitative Approaches to Decision Making*. South-Western Pub, 2007.

[124] F. Hillier, G. Lieberman, and M. Hillier, *Introduction to Operations Research*. McGraw-Hill, 1990.

[125] T. Richardson and R. Urbanke, *Modern Coding Theory*. Cambridge University Press, 2008.

[126] A. Dixit, *Optimization in Economic Theory*. Oxford University Press, USA, 1990.

[127] W. Baumol and W. Baumol, *Economic Theory and Operations Analysis*. Prentice-Hall Englewood Cliffs, NJ, 1977.

[128] J. Borwein, A. Lewis, J. Borwein, and A. Lewis, *Convex Analysis and Nonlinear Optimization: Theory and Examples*. Springer New York, 2006.

[129] J. Hiriart-Urruty and C. Lemaréchal, *Convex Analysis and Minimization Algorithms: Part 1: Fundamentals*, vol. 1. Springer, 1996.

[130] R. Rockafellar, *Convex Analysis*. Princeton University Press, 1996.

[131] D. Bertsekas, A. Nedi, and A. Ozdaglar, *Convex Analysis and Optimization*. Athena Scientific, 2003.

[132] D. Bertsekas, *Nonlinear Programming*. Athena Scientific, 1999.

[133] M. Andersen, J. Dahl, M. ApS, Z. Liu, and L. Vandenberghe, "1 interior-point methods for large-scale cone programming," *Optimization for Machine Learning*, pp. 55–83, 2011.

[134] Y. Nesterov, A. Nemirovskii, and Y. Ye, *Interior-point Polynomial Algorithms in Convex Programming*. SIAM, 1994.

[135] Y. Ye, *Interior Point Algorithms: Theory and Analysis*. Wiley-Interscience, 1997.

[136] R. Byrd, M. Hribar, and J. Nocedal, "An interior point algorithm for large-scale nonlinear programming," *SIAM Journal on Optimization*, vol. 9, no. 4, pp. 877–900, 1999.

[137] G. Dantzig, *Linear Programming and Extensions*. Princeton University Press, 1998.

[138] P. Wolfe, "The simplex method for quadratic programming," *Econometrica: Journal of the Econometric Society*, pp. 382–398, 1959.

[139] M. Bergounioux, M. Haddou, M. Hintermüller, and K. Kunisch, "A comparison of a moreau–yosida-based active set strategy and interior point methods for constrained optimal control problems," *SIAM Journal on Optimization*, vol. 11, no. 2, pp. 495–521, 2000.

[140] M. S. Andersen and L. V. J. Dahl, "CVXOPT: A python package for convex optimization, version 1.1.5." `http://abel.ee.ucla.edu/cvxopt`, 2012.

[141] T. T. D. Rubira, "CVXPY: A python package for modeling convex optimization problems." `http://code.google.com/p/cvxpy/`, 2012.

[142] I. CVX Research, "CVX: Matlab software for disciplined convex programming, version 2.0." `http://cvxr.com/cvx`, Aug. 2012.

[143] M. Grant and S. Boyd, "Graph implementations for nonsmooth convex programs," in *Recent Advances in Learning and Control* (V. Blondel, S. Boyd, and H. Kimura, eds.), Lecture Notes in Control and Information Sciences, pp. 95–110, Springer-Verlag Limited, 2008. `http://stanford.edu/~boyd/graph_dcp.html`.

[144] W. Fenchel, "On conjugate convex functions," *Canad. J. Math*, vol. 1, pp. 73–77, 1949.

[145] W. Fenchel, *Convex Cones, Sets, and Functions*. Princeton University, 1953.

[146] C. Shannon, W. Weaver, R. Blahut, and B. Hajek, *The Mathematical Theory of Communication*. University of Illinois Press Urbana, 1949.

[147] R. Gallager, *Information Theory and Reliable Communication*. Wiley, 1968.

[148] D. Barber, *Bayesian Reasoning and Machine Learning*. Cambridge University Press, 2012.

[149] T. Koski and J. Noble, *Bayesian Networks: An Introduction*. Wiley, 2011.

[150] T. Nielsen and F. Jensen, *Bayesian Networks and Decision Graphs*. Springer, 2007.

[151] F. Jensen, *An Introduction to Bayesian Networks*. UCL Press London, 1996.

[152] R. Cowell, P. Dawid, S. Lauritzen, and D. Spiegelhalter, *Probabilistic Networks and Expert Systems: Exact Computational Methods for Bayesian Networks*. Springer, 2007.

[153] B. Frey, *Graphical Models for Machine Learning and Digital Communication*. MIT Press, 1998.

[154] S. Lauritzen, *Graphical Models*. Oxford University Press, USA, 1996.

[155] G. Golub and C. Van Loan, *Matrix Computations*. Johns Hopkins University Press, 1996.

[156] J. Edmonds and R. Karp, "Theoretical improvements in algorithmic efficiency for network flow problems," *Journal of the ACM (JACM)*, vol. 19, no. 2, pp. 248–264, 1972.

[157] S. Dasgupta, "Learning mixtures of gaussians," in *40th Annual Symposium on Foundations of Computer Science*, pp. 634–644, 1999.

[158] A. Asuncion and D. Newman, "Uci machine learning repository. irvine, ca: University of california, school of information and computer science." `http://www.ics.uci.edu/~mlearn/MLRepository.html`, 2007.

[159] R. Gaetano, G. Scarpa, and G. Poggi, "Hierarchical texture-based segmentation of multiresolution remote-sensing images," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 47, no. 7, pp. 2129–2141, 2009.

[160] D. Zamalieva, S. Aksoy, and J. Tilton, "Finding compound structures in images using image segmentation and graph-based knowledge discovery," in *Proceedings of the IEEE International Geoscience and Remote Sensing Symposium*, vol. 5, pp. V–252, 2009.

[161] M. Vanegas, I. Bloch, and J. Inglada, "Detection of aligned objects for high resolution image understanding," in *Proceedings of the IEEE International Geoscience and Remote Sensing Symposium*, pp. 464–467, 2010.

[162] H. Akçay and S. Aksoy, "Detection of compound structures using hierarchical clustering of statistical and structural features," in *Proceedings of the IEEE International Geoscience and Remote Sensing Symposium*, pp. 2385–2388, 2011.