Clustering is a term used to describe a wide variety of unsupervised learning algorithms. One popular definition of clustering is that it attempts to partition a given dataset into subsets (or clusters) such that similar points share the same cluster and dissimilar points are separated into different clusters. On a closer look, we see that this definition is problematic. Consider a set of (say n) points on a straight line where each pair of adjacent points have a small distance (say α) between them. If we impose the requirement that each pair of similar points should share a cluster then all the points would end up in a single cluster. This would violate the second requirement as dissimilar points would also share the same cluster.

The basic definition does not have enough information to resolve this conflict. We say that the clustering problem is not well-defined or is *under-specified*. As a more concrete example, consider the problem of clustering the users of an online retail service into different groups. The output of this clustering can be used to recommend similar items to similar users. Another possible application could be to group the users based on their spending patterns etc. Both these requirements impose different restrictions on the desired solution. One of the main focus of this thesis is to address this problem.

We propose a principled approach to incorporate domain knowledge into the clustering problem by allowing the algorithm to interact with an oracle. The algorithm interacts with the oracle by asking whether two points should belong to the same or different clusters. The oracle replies either 'yes' or 'no' to the *same-cluster* query depending upon whether the two points belong to the same or different clusters. In this case, the goal of the clustering algorithm is to recover the clustering which the oracle has in its 'mind'.

We study the *computational* and *query* complexity of various clustering problems in this framework. Consider the following simple observation. Given any clustering instance, if the algorithm is allowed to make n^2 (where n is the size of the dataset) queries to the same-cluster oracle then recovering the true or target clustering is trivial. In this dessertation, one of the important questions that we examine is the following. Is it possible to efficiently solve (in polyniomial time) an otherwise intractable clustering problem while making a 'small' number of same-cluster queries to the oracle?

Now, we discuss the second problem; the issue of *noise-robustness* of clustering algorithms. The basic definition of clustering says that the goal is to partition a dataset into clusters such that similar points share a cluster while dissimlar points are separated into different clusters. This definition

makes sense when the given dataset has a cohesive structure. That is, the dataset can be partitioned into groups or clusters which have some inter-cluster separation. However, real-world datasets, on top of this structure, have a significant subset of points which are unstructured. The addition of these *noisy* points makes it difficult for the clustering algorithm to detect the structure of remaining points. The precise definition of 'unstructuredness' or noisy points varies depending upon the structure that the clustering algorithm is trying to detect. In this dissertation, we consider two definitions of noise and develop noise-robust clustering algorithms in each case.

We address both the issues, noise-robustness and under-specificity under a formal framework. The goal is to have a framework which can be mathematically analysed and is also relevant to practitioners. Obviously, the exact framework varies depending upon the clustering problem we are considering. To analyse these issues in a formal framework, we will be mostly relying on mathematical theorems and proofs. But in some cases (where applicable), we have also complimented our results with experiments and simulations.

1 Our Contributions

As we alluded to before, our contributions can be divided into two categories. One is dealing with under-specificity and the second related to noise robustness of clustering algorithms. In this next sub-sections, we go into more details of each of these and outline our objectives and contributions made.

1.1 Under-specificity

The first main contribution is to develop a formal framework to incorporate domain knowledge into the clustering problem. We introduce a semi-supervised clustering framework. The learner is allowed to interact with a domain expert (or an oracle) by asking whether two data instances belong to the same or different cluster (same-cluster query). The oracle has a target clustering in its mind and responds by answering either 'yes' or 'no' (depending upon whether the two points belong to the same or different clusters according to the target clustering). We assume that the oracle is perfect and has complete knowledge of the ground truth clustering. Hence, given any pair of points it always gives the correct response. We consider two clustering problems under this framework.

Clustering with advice (k-means)

We consider a setting where the oracle conforms to a center-based clustering with a notion of margin. That is, the target clustering has the following property. Every cluster-center is 'more' closer (γ-times closer) to points in its own cluster than to points in a different cluster. Larger values of γ imply greater separation between different clusters.

Under this framework, we study the query and computational complexity of recovering the target clustering. We provide an algorithm which runs in $O\big(kn \log n\big)$ time and makes $O\big(k \log n + k^2 \log k\big)$ same-cluster queries to the oracle and succeeds in recovering the oracle's clustering with high probability. Here n is the size of the dataset and k is the number of clusters.

We also consider the case when the oracle conforms to the optimal k-means clustering under γ-margin. Then, our query-based algorithm can find the optimal solution in polynomial time. Interestingly, we prove that even margin under conditions, without queries, finding the optimal k-means solution is NP-hard. Thus having access to relatively few oracle queries can allow efficient solutions to otherwise intractable problems.

Correlation clustering with advice

We consider the problem of correlation clustering under the semi-supervised clustering framework. Correlation clustering is very useful for modelling the record de-duplication problem; the task of detecting multiple records that correspond to the same real-world entity in a database. Here the goal is to put records corresponding to the same physical entity in the same cluster and putting records corresponding to different physical entities into different clusters.

Formally, given a complete graph G with the edges labelled 0 and 1, the goal of correlation clustering is to find a clustering that minimizes the number of 0 edges within a cluster plus the number of 1 edges across different clusters. In other words, the goal is to find a clustering which minimizes the *correlation loss* w.r.t the graph G.

Promise correlation clustering

The optimal clustering C^* can also be viewed as a complete graph G^* (unknown to the clustering algorithm) with edges corresponding to points in the same cluster being labelled 1 and other edges being labelled 0. If it is known

that the edge difference between G and G^* is zero, then finding C^* is easy (find the connected components of G). We consider a variant of this problem where it is promised that the edge difference between G and the unknown G^* is "small". The goal is to find the clustering which minimizes the correlation loss w.r.t G.

We now wish to analyse the computational and query complexity of the promise correlation clustering (PCC) problem. We prove that the promise version is still NP-Hard. Rather surprisingly, we further prove that even with access to a same-cluster oracle, the promise version is NP-Hard as long as the number queries to the oracle is $o(n)$ (the proof assumes the Exponential Time Hypothesis; n is the number of vertices).

Restricted correlation clustering
Given these negative results, we consider a restricted version of correlation clustering. First observe that in the standard version, the goal is to find a clustering over the class of all possible clusterings of the dataset. Here, we restrict ourselves to a given class $\mathcal{F}$ of clusterings. Another difference is that we want to minimize the correlation loss w.r.t the unknown target clustering C^* rather than a graph G.

We now wish to analyse the query and computational complexity of this problem. We offer an algorithmic approach (using same-cluster queries) and prove that the query complexity is upper bounded by a term that depends only on the VC-Dim($\mathcal{F}$) and is independent of the dataset size. We also provide computationally efficient algorithms for a few common classes of clusterings.

1.2 Noise-robustness

We now describe our framework to address the issue of noise in clustering algorithms. We are given a dataset X which is made up of two parts. The first is the structured or clusterable component S. Mathematically, this is captured by introducing notions of 'clusterability of data'. Intuitively, these notions say that the set S is composed of k different clusters and the clusters are 'well-separated' from each other. In this thesis, we consider two such notions, center-proximity and center-separation. Each of them formalize the idea of well-separatedness in a different way. The second component of the dataset is the noisy or unstructured part N. The clustering algorithm receives X as its input. It does not have any knowledge about S (or its substructure)

or N. The goal is to partition X into components so that the structure of S is *preserved*. Any algorithm which achieves this goal is said to be noise-robust.

Detecting cluster structure in the presence of outliers

We propose a generic regularization-based method that transforms any center-based clustering objective into a noise-robust one. We use our procedure to obtain regularized versions of two common clustering algorithms based on the k-means objective function. We prove that these regularized algorithms are robust to outliers (under clusterability assumptions and mildness properties of the noisy points).

Detecting cluster structure in the presence of sparse noise

We consider the problem of detecting the structure of S when the noisy part N is sparse. That is, the only restriction about the noisy part of the data is that it does not create significantly large clusters. We introduce efficient algorithms that discover and cluster every subset S of the data with the following property. S has a meaningful structure (as captured by a notion of clusterability) and its complement is structureless or sparse. We say that our algorithm is robust to sparse noise. Notably, the success of our algorithms do not depend on any upper bound on the fraction of noisy data. We complement our results by showing that when either the notions of structure or the noise requirements are relaxed, no such results are possible.

2 Reading the thesis

This dissertation is composed of two components. The first addresses the problem of under-specificity in clustering and is covered in Chapters **??** and **??**. The second part is about the issue of noise-robustness of clustering algorithms and is covered in Chapters **??** and **??**. The two parts are independent of one another and the reader can start with whichever one he/she is more interested in. The author feels that $60 - 65\%$ of the thesis is about the first part (dealing with under-specificity) and the remaining deals with noise-robustness.

We have ensured that each chapter is self-contained. Note that the thesis does not have a dedicated chapter on related work or on notation/preliminaries.

Rather these are included in the relevant chapters. Some of the missing proofs can be found in appendices at the end of the corresponding chapters.