Clustering aims to group similar data instances together while separating dissimilar ones. However, often many datasets have, on top of cohesive groups, a subset of "unstructured" points as well. In such cases, the goal is to detect the structure while simultaneously separating the unstructured data points. Clustering algorithms that achieve this goal are said to be *robust* to noise.

The most common approach to clustering views it as an optimization problem. The idea is to associate a cost (or objective) with each possible partition of the input dataset (into k subsets) and then try to find a partition which has minimum cost. Examples of common objective functions include k-means and k-median cost functions. However, these common objective functions are not robust to the presence of noise.

In this chapter, we propose a generic method of regularization that can transforms any clustering objective which outputs k clusters to one that outputs $k+1$ clusters. The algorithm is now allowed to 'discard' points into an extra 'garbage' or noise cluster by paying a constant regularization penalty. The intuition is that allowing the clustering algorithm to discard a few points should make it easier to detect the structure in the remaining non-noisy points. However, we prove that finding the optimal solution to the regularized objective is NP-Hard [1]. The goal is to develop an efficient clustering algorithm which is provably noise-robust.

From a theoretical perspective, one common approach to dealing with such hardness results is the following. (i) Consider a convex relaxation of the original objective function (which can be efficiently solved using standard techniques). (ii) Prove that under certain data niceness conditions the solution obtained by solving the relaxed objective function coincides with the optimal solution of the original objective function. For example, [**?**] used this strategy to design an algorithm based on the sdp-relaxation of the k-means objective function. [**?**] proved that under the 'stochastic ball assumption', the solution of the sdp-based approach is indeed the optimal k-means clustering. In this chapter, we will use the same approach (sdp relaxation of the regularized k-means objective) to design an efficient and noise-robust clustering algorithm.

Our framework is the following. We are given an input dataset X made of two components. The first is the clusterable subset I which satisfies a niceness property. Namely, I is the union of k unit balls B_i each separated by a distance of atleast δ. The second is the unstructured or noise component N.

[1] Throughout this chapter, we consider the standard and regularized versions of the k-means objective

Note that the clustering algorithm only sees X and is not aware of I or N. The goal is to design an efficient clustering algorithm $\mathcal{A}$ such that the output of the algorithm $\mathcal{A}(X)$ when restricted to I, is able to detect and recover the structure of I (namely, the balls B_i). In this chapter, we consider two choices for $\mathcal{A}$. The first based on SDP relaxation of the regularized k-means objective and the second based on the LP relaxation.

For the noiseless case, [?] showed that for $\delta > 2\sqrt{2}(1 + 1/\sqrt{d})$ (where d is the dimension of the euclidean space) the algorithm based on SDP-relaxation of the standard k-means objective recovers (with high probability) the structure of I if the balls B_i are generated by an isotropic distribution (stochastic ball model). In the context of the sdp-based clustering algorithm, the stochastic ball assumption has been used in numerous other previous works like [?], [?], [?] and [?].

In this chapter, we improve over previous results and give success guarantees in the regime $\delta > 2(1 + \sqrt{k/d})$ which is near-optimal for large d. In the presence of noise, we prove that the algorithm based on the SDP-relaxation of the regularized objective recovers the clustering of I for $\delta > 2(1 + \sqrt{\zeta + k/d})$. Here ζ is a term which depends on the ratio of number of noisy points and the number of points in the smallest cluster. We obtain similar results for the LP based algorithm as well. However, in that the case both separation requirement and the restrictions on noise are stronger.

We also conduct simulation studies where we examine the effect of λ, the number of noisy points m, the separation δ and other parameters on the performance of our regularised SDP-based algorithm. We also perform experiments on the MNIST dataset. We observed that the regularised version performed better than k-means++ when the dataset had noisy points. In the absence of noise, the performance of both these algorithms were similar.

1 Related Work

The problem of robustifying a clustering objective function has been studied before. [?] give a generic procedure that can transform any clustering objective with metric d to an objective function which uses d' which is a truncated version of the original metric. More specifically, $d'(x, y) = \min(\lambda, d(x, y))$. Then they show that if there exists an algorithm which can solve the optimization problem then such an algorithm is robust to noise. [?] also studies breakdown properties of the truncated objective function, that is the number of outliers which can cause significant change in the estimates of any one of the centers. However, it is unclear as to how we would solve the optimization problem. In this work, we propose an efficient algorithm based on convex relaxation of our regularized objective function and give robustness guarantees for the same.

The problem of recovering the underlying cluster structure in the presence and absence of noise has been studied before both in distribution-free and distribution-based settings. In the distribution-free setting, the goal is to prove that if the data has some structure (is *clusterable*) then the (proposed or existing) clustering algorithm recovers that structure in the presence or absence of noise. Such works, make no assumption on the distribution that generated the data. Different works define different notions of 'clusterable' data. In the current work, the separation requirement on the clusters was global. That is, the each cluster was separated by atleast δ times the maximum radius amongst all the clusters. Another popular notion of clusterability is α-center proximity [**?**] which requires that two clusters be separated relative to their radii. [**?**] consider a dataset which has α-center proximity except for an ϵ-fraction of points. They propose an efficient algorithm which provides an $1 + O(\epsilon)$ approximation to the optimal k-median solution for $\alpha > 2 + \sqrt{7}$.

Different works on clustering have also made different assumptions on the type of noisy points. The most common is to assume that the noise is adversarial but the number of adversaries is not too large. For example, and [**?**] provide bounds for clustering in the presence of noise as long as the number of adversaries is constant-factor smaller compared to the size of the smallest cluster. [**?**] considered noise which is structureless, that is the noisy points do not form dense large subsets. Another field of work is to address noisy part of the data as being generated by some uniform random noise or gaussian perturbations [**?**], [**?**] and [**?**].

In the distribution-based setting, the goal is to estimate the parameters of the distribution (say the mean and variance of gaussian etc).

Another line of work which is related to ours is clustering a mixture of k gaussians with few adversaries. The best known result is by [**?**] which requires that the mean of the gaussians be separated by $\tilde{O}(\sigma\sqrt{k})$ where σ^2 is an upper bound on the variance of the k gaussians. Recently, [**?**] matched this result using different techniques. Although the distribution free setting considered in this chapter is different from the above works, the separation required for the SDP-based algorithm to succeed also has a similar dependence on k, namely $\delta > 2(1 + \sqrt{k})$.

Some works examine the robustness of different algorithms when the number of clusters is the same for the original data and the data with added noise. They show that in this setting the traditional algorithms are provably not noise-robust [?] and [?]. Another line of work which is related to ours is based on the convex relaxation of center-based clustering objectives. [?] was the first to formulate the k-means cost function as a 0-1 SDP and then subsequently relaxed it to a standard SDP. In this work, we use a similar technique to first obtain a 0-1 SDP and subsequently a relaxed SDP for the regularised k-means objective.

2 Preliminaries and definition

Let $(\mathbf{M}, d)$ be a metric space. Given a finite set $\mathcal{X} \subset \mathbf{M}$, a k-clustering $\mathcal{C}$ of $\mathcal{X}$ partitions the set into k disjoint subsets $\mathcal{C} = \{C_1, \ldots, C_k\}$. An objective-based clustering algorithm associates a cost with each possible partition of $\mathcal{X}$ and then tries to find the clustering with minimum cost. Throughout this section, f denotes a function on the nonnegative reals.

Definition 1 ((k, f)-objective algorithm)**.** *Given $\mathcal{X} \subset \mathbf{M}$ and a distance function d, a (k, f)-objective based algorithm $\mathcal{A}$ tries to find centers $\mu_1, \ldots, \mu_k \in \mathbf{M}$ so as to minimize the following function*

$$Cost(\mu_1, \ldots, \mu_k) = \sum_{x \in \mathcal{X}} f(d(x, \mu(x))), \quad \mu(x) = \operatorname*{arg\,min}_{\mu \in \{\mu_1, \ldots, \mu_k\}} d(x, \mu). \tag{1}$$

Note that algorithm $\mathcal{A}$ may not often find the optimal solution because for many common functions f, solving the optimization is NP-Hard. Thus, heuristics are used that can get stuck at a local minima. For example, when $f(x) = x^2$, the above definition corresponds to the k-means objective, and the Lloyd's algorithm that is used to solve this objective can get stuck at a local minima.

Definition 2 ((k, f)-λ-regularised objective algorithm)**.** *Given $\mathcal{X} \subset \mathbf{M}$ and a distance function d, a (k, f)-λ-regularised objective based algorithm $\mathcal{A}'$ tries to find centers $\mu_1, \ldots, \mu_k \in \mathbf{M}$ and set $\mathcal{I} \subseteq \mathcal{X}$ so as to minimize the following function*

$$Cost(\mu_1, \ldots, \mu_k, \mathcal{I}) = \sum_{x \in \mathcal{I}} f(d(x, \mu(x))) + \lambda |\mathcal{X} \setminus \mathcal{I}|, \tag{2}$$

where $\mu(x) = \arg\min_{\mu_i} d(x, \mu_i)$.

The regularised objective allows discarding certain points into a "garbage" cluster at the expense of paying a constant penalty. The intuition is that this will help the algorithm better detect the structure of the remaining points. We will see in §3 that minimizing this objective function is NP-Hard for all $k \geq 1$.

2.1 Robustification paradigms

Definition 3 (λ-Regularised Paradigm). *The λ-regularised paradigm is a robustification paradigm which takes as input a (k, f)-objective algorithm $\mathcal{A}$ and returns a (k, f)-λ-regularised objective algorithm $\mathcal{A}'$.*

In this work, we focus on robustification of the k-means objective. Hence, it is useful to define the regularised k-means objective as we will refer to it many times in the remainder of the chapter.

2.2 Regularised k-means objective

Given a finite set $\mathcal{X} \subset \mathbf{R}^d$ and an integer k, the regularised k-means objective aims to partition the data into $k+1$ clusters $\mathcal{C} = \{C_1, \ldots, C_k, C_{k+1}\}$ so as to solve

$$\min_{\substack{C_1,\ldots,C_{k+1} \\ c_1,\ldots,c_k}} \sum_{i=1}^{k} \sum_{x \in C_i} \|x - c_i\|_2^2 + \lambda |C_{k+1}|. \tag{3}$$

Note that the first term of the objective depends on the l_2 norm, while the second term depends only on the cardinality of the "noise" cluster. In order to make our objective function invariant to scaling, the regularization constant λ is added to the cost function.

Let $m(\mathcal{X}) := \min_{x \neq y \in \mathcal{X}} \|x - y\|_2^2$ and $N = |\mathcal{X}|$. Then, it is easy to see when $\lambda \leq \frac{m(\mathcal{X})}{2}$, then (3) admits a trivial solution: each cluster C_i for $i \leq k$ has exactly one point and all the remaining points are in C_{k+1}, leading to an objective $\lambda(N-k)$. Indeed, for any other clustering with $|C_i| = n_i$ its objective is at least $\sum_{i=1}^{k}(n_i - 1)m(\mathcal{X})/2 + \lambda n_{k+1} = (n - n_{k+1} - k)m(\mathcal{X})/2 + \lambda n_{k+1}$, where we have used the simple fact:

$$\forall C \subset \mathcal{X}, \quad \min_c \sum_{x \in C} \|x - c\|_2^2 \; = \; \frac{1}{2|C|} \sum_{x,y \in C} \|x - y\|_2^2. \tag{4}$$

Comparing the objectives we see the solution is indeed trivial when $\lambda \leq m(\mathcal{X})/2$. Surprisingly, for the interesting case when $\lambda > m(\mathcal{X})/2$, the problem suddenly becomes NP-Hard, as we prove below.

2.3 Robustness measure

Given two clusterings $\mathcal{C}$ and $\mathcal{C}'$ of the same set $\mathcal{X}$, we define the distance between them, $\Delta(\mathcal{C}, \mathcal{C}')$, as the fraction of pairs of points which are clustered differently in $\mathcal{C}$ than in $\mathcal{C}'$. Given $\mathcal{I} \subseteq \mathcal{X}$, $\mathcal{C}|\mathcal{I}$ denotes the restriction of the clustering $\mathcal{C}$ to the set $\mathcal{I}$.

Definition 4 (γ-robust [?]). *Given $\mathcal{X} \subset \mathbf{M}$ and clustering algorithm $\mathcal{A}$, let $\mathcal{A}'$ be its robustified version obtained using any robustification paradigm. Given $\mathcal{I} \subseteq \mathcal{X}$, we say that $\mathcal{I}$ is γ-robust w.r.t $\mathcal{X} \setminus \mathcal{I}$ and $\mathcal{A}'$ if*

$$\Delta(\mathcal{A}'(\mathcal{X})|\mathcal{I}, \mathcal{A}(\mathcal{I})) \leq \gamma \tag{5}$$

This measure tries to quantify the difference in the clustering of the set $\mathcal{I}$ after the addition of 'noisy' points $\mathcal{X} \setminus \mathcal{I}$. If $\mathcal{A}'$ is indeed robust to noisy points, then the clusterings should be similar.

3 Hardness of regularised k-means

In this section, we present hardness results for the regularised k-means objective. The proof for $k \geq 2$ is fairly straightforward and follows from known hardness results for the standard (non-regularised) k-means. The more interesting case is when $k = 1$. It is well-known that 1-means can be solved in linear time [?] hence the same reduction does not work any more. We reduce an instance of the MAX-CLIQUE problem to the regularised 1-means problem. We give a proof sketch for the reader's intuition. The technical details can be found in the supplementary section.

Theorem 5. *Given a clustering instance $\mathcal{X} \subset \mathbf{R}^d$. Finding the optimal solution to the regularised 1-means objective is NP-Hard for all $\lambda > m(\mathcal{X})/2$, where recall that $m(\mathcal{X}) := \min_{x \neq y \in \mathcal{X}} \|x - y\|_2^2$.*

Proof sketch. The proof has two parts. We first show that for fixed λ the problem is NP-Hard. The proof works by reducing an instance of MAX-CLIQUE to the regularised 1-mean instance. The idea is to define the

Algorithm 1 SDP-based regularised k-means algorithm

```
Input: $\mathcal{X} \subset R^d$, $k$, and hyperparameter $\lambda$.
Output: $\mathcal{C}' := \{C_1, \dots, C_k, C_{k+1}\}$.
Compute the matrix $D_{ij} = \|x_i - x_j\|_2^2$.
Solve the SDP (Eqn. 6) using any standard SDP solver and obtain matrix $Z$ and vector $y$.
Use the rounding procedure (Alg. 2) to obtain the partition $\mathcal{C}'$.
```

distance between any pair of vertices as 1 if there exists an edge between them. If not, then define the distance as $1 + \Delta$ for a suitably chosen Δ. This construction guarentees that the problem is NP-Hard for atleast one $\lambda > \frac{m(\mathcal{X})}{2}$. Next, using a scaling argument we show that if the problem is NP-Hard for one particular λ, then it is NP-Hard for all $\lambda > \frac{m(\mathcal{X})}{2}$. □

The above theorem infact shows that regularised k-means is hard for all $k \geq 1$. This is becuase we can reduce an instance of regularised 1-means to regularised k-means by placing $k - 1$ points very far away.

4 The regularised k-means SDP-based algorithm

In the previous section, we showed that the regularised k-means objective is NP-Hard to optimize. Hence, we cannot hope to solve the problem exactly unless $P = NP$. In this section, we develop an algorithm based on semi-definite programming relaxation of the regularised objective.

[**?**] developed an algorithm $\mathcal{A}$ (Alg. 1 with $\lambda = \infty$) which tries to minimize the k-means objective. They obtained a convex relaxation of the k-means objective and solved it polynomially using standard solvers. In this section, we use the same technique to obtain and efficiently solve the convex relaxation of the regularised k-means objective. Our algorithm $\mathcal{A}'$ (Alg. 1) is the robustified version of $\mathcal{A}$ using the λ-regularised paradigm. In §4.1, we give the details of how we transform the regularised objective into an SDP. §4.2 has our main results where we give robustness guarentees for $\mathcal{A}'$.

4.1 The SDP-based algorithm

The SDP relaxation of the regularised k-means objective is obtained in two steps. Using similar technique to that of [**?**], we translate Eqn. 3 into a 0-1

SDP (Eqn. 6). We then prove that solving the 0-1 SDP exactly is equivalent to solving the regularised k-means problem exactly. Then, we relax some of the constraints of the 0-1 SDP to obtain a tractable SDP which we then solve using standard solvers. We then describe the rounding procedure which uses the solution of the SDP to construct a clustering of the original dataset.

$$\begin{array}{c}\textbf{0-1}\\ \textbf{SDP}\end{array}\begin{cases}\min_{Z,y} & \mathrm{Tr}(DZ)+\lambda\langle\mathbf{1},y\rangle\\ \text{s.t.} & \mathrm{Tr}(Z)=k\\ & Z\cdot\mathbf{1}+y=\mathbf{1}\\ & Z\geq 0, Z^2=Z, Z^T=Z\\ & y\in\{0,1\}^n\end{cases}\xrightarrow{\text{relaxed}}\textbf{SDP}\begin{cases}\min_{Z,y} & \mathrm{Tr}(DZ)+\lambda\langle\mathbf{1},y\rangle\\ \text{s.t.} & \mathrm{Tr}(Z)=k\\ & \left(\frac{Z+Z^T}{2}\right)\cdot\mathbf{1}+y=\mathbf{1}\\ & Z\geq 0, y\geq 0, Z\succeq 0\end{cases} \tag{6}$$

Theorem 6. *Finding a solution to the 0-1 SDP (6) is equivalent to finding a solution to the regularised k-means objective (3).*

Equation 6 shows our 0-1 SDP formulation. The optimization is NP-Hard as it is equivalent to the regularised k-means objective. Hence, we consider a convex relaxation of the same. First, we replace $Z^2 = Z$ with $Z \succeq 0$. In addition, we relax $y \in \{0,1\}^n$ to $y \geq 0$, as the constraint $y \leq 1$ is redundant. Using these relaxations, we obtain the SDP formulation for our objective function.

We solve the SDP using standard solvers [**?**] thereby obtaining Z, y. The proof of Thm. 6 showed that the optimal solution of 0-1 SDP is of the following form. Z is a $n \times n$ block diagonal matrix of the form $\mathrm{diag}(Z_{I_1}, \dots, Z_{I_k}, 0)$, where $n = |\mathcal{X}|$ and $Z_{I_i} = \frac{1}{|C_i|}11^T$. Thus, given Z, we can extract the set of cluster centers $C = ZX$ which is an $n \times d$ matrix. Each row C_i contains the cluster center to which data point x_i belongs. For the points x_j assigned to the noise cluster, the corresponding row C_j is zero and $y_j = 1$. The SDP solver does not always return the optimal solution, as the relaxation is not exact. However, we expect that it returns a near-optimal solution. Hence, given Z and y returned by the solver, we use Alg. 2 to extract a clustering of our original dataset. The *threshold* parameter indicates our confidence that a given point is noise. In our experiments, we have used a threshold of 0.5. We did not tune the threshold as in our experiments the results are not very sensitive to it.

Algorithm 2 Regularised k-means rounding procedure

```
Input: $Z \subset \mathbb{R}^{n\times n}$, $y \subset \mathbb{R}^n$, $\mathcal{X}$, and $threshold \in [0,1]$.
Output: $\mathcal{C}'$.
If $y_i > threshold$ then
    Delete $z_i$ and $z_i^T$ from $Z$. Put $x_i$ in $C_{k+1}$.
    Delete $x_i$ from $X$.
$k$-cluster the columns of $X^T Z$ to obtain clusters $C_1, \dots, C_k$.
Output $\mathcal{C}' = \{C_1, \dots, C_k, C_{k+1}\}$.
```

4.2 Robustness guarantees

Assume that we are given a set $\mathcal{I}$ of k well-separated balls in $\mathbf{R}^d$. That is, $\mathcal{I} := \cup_{i=1}^k B_i$ where each B_i is a ball of radius at most one and centered at μ_i such that $\|\mu_i - \mu_j\| \geq \delta$. On top of this structure, points are added from the set $\mathcal{N}$. Let $\mathcal{A}$ and $\mathcal{A}'$ be the SDP based standard and regularised k-means algorithm respectively (as defined in the begining of §4). We will show that $\mathcal{I}$ is 0-robust w.r.t $\mathcal{A}'$ and $\mathcal{N}$ under certain conditions on δ and mildness properties of the set $\mathcal{N}$. To show this, we need to compare the clusterings $\mathcal{A}(\mathcal{I})$ and $\mathcal{A}'(\mathcal{X})|_{\mathcal{I}}$. We first prove recovery guarentees for $\mathcal{A}$ in the absense of noisy points.

Theorem 7. *Let $\mathcal{P}$ denote the isotropic distribution on the unit ball centered at origin in $\mathbf{R}^d$. Given points $\mu_1, \dots, \mu_k$ such that $\|\mu_i - \mu_j\| > \delta > 2$. Let $\mathcal{P}_i$ be the measure $\mathcal{P}$ translated with respect to the center μ_i. Let each B_i is drawn i.i.d w.r.t the distribution $\mathcal{P}_i$.*

Given a clustering instance $\mathcal{I} \subset \mathbf{R}^{N\times d}$ and k where $\mathcal{I} := \cup_{i=1}^k B_i$. Define $n := \min_{i\in[k]} |B_i|$ and $\rho = \frac{N}{nk}$. If

$$\delta > 1 + \sqrt{1 + \frac{2\theta\rho k}{d}\Big(1 + \frac{1}{\log N}\Big)^2}$$

where $\theta = \mathbf{E}[\|x_{pi} - c_p\|^2] < 1$, then there exists a constant $c > 0$ such that with probability at least $1 - 2d\exp(\frac{-cN\theta}{d\log^2 N})$ the k-means SDP finds the intended cluster solution $\mathcal{C}^* = \{B_1, \dots, B_k\}$.

Thm. 7 improves the result of Thm. 11 in [?]. Under the stochastic ball assumption, they showed that the k-means SDP finds the intended solution for $\delta > 2\sqrt{2}(1 + \frac{1}{\sqrt{d}})$. For $k \ll d$, which is the case in many situations our bounds are optimal in terms of the separation reqirement of the clusters. [**?**] obtained similar results but for $\delta > 2 + \frac{k^2}{d} cond(\mathcal{I})$. Asymptotically, as d goes to ∞ their bound matches our result. However, the condition number (ratio of maximum distance between any two centers to the minimum distance between any two centers) can be arbitrarily large. We also have a better dependence on k and d.

Next, we analyse the recovery guarentees for $\mathcal{A}'$ in the presence of noisy points $\mathcal{N}$. We decompose the noisy points into two disjoint sets $\mathcal{N}_1$ and $\mathcal{N}_2$. The set $\mathcal{N}_2$ consists of all the points which are far from any of the points in $\mathcal{I}$. The set $\mathcal{N}_1$ consists of points which are close to atleast one of the clusters. We also require that any point in $\mathcal{N}_1$ has an α-margin w.r.t to the centers of the balls $B_1, \ldots, B_k$. That is the difference of the distance between any point in $\mathcal{N}_1$ to a cluster center is atleast α. Now, we will show that if $\mathcal{N}$ has the aforementioned properties then $\mathcal{I}$ is robust w.r.t the regularised SDP algorithm $\mathcal{A}'$.

Theorem 8. *Let $\mathcal{P}$ denote the isotropic distribution on the unit ball centered at origin in $\mathbf{R}^d$. Given centers $\mu_1, \ldots, \mu_k$ such that $\|\mu_i - \mu_j\| > \delta > 2$. Let $\mathcal{P}_i$ be the measure $\mathcal{P}$ translated with respect to the center μ_i. Let B_i is drawn i.i.d w.r.t the distribution $\mathcal{P}_i$.*

Given a clustering instance $\mathcal{X} \subset \mathbf{R}^{N \times d}$ and k. Let $\mathcal{X} := \mathcal{I} \cup \mathcal{N}$ where $\mathcal{I} := \cup_{i=1}^{k} B_i$. Let $\mathcal{N} = \mathcal{N}_1 \cup N_2$ have the following properties. For all $n \in N_1$ and for all i, j, we have that $|||(n - \mu_i\|^2 - \|n - \mu_j\|^2| \geq \alpha$. For all $n \in \mathcal{N}_2$ and for all $x \in \mathcal{I}$, $\|n - x\| \geq \nu \geq \sqrt{(\delta - 1)^2 + 1}$. Note that $\mathcal{N}_1 \cap \mathcal{N}_2 = \phi$. Let $n = \min_i |B_i|$ and $\epsilon = \frac{|\mathcal{N}_1|}{n}$ and $\rho = \frac{|I|}{nk}$. If

- $\delta > 2 + \sqrt{O(\epsilon) + \frac{2\rho k \theta (1 + 1/\log(|I|))^2}{d}}$
- $\alpha \geq O(\epsilon) + \frac{2\rho k \theta (1 + 1/\log|I|)^2}{d}$
- $\frac{|\mathcal{N}_2|}{n} \leq \frac{\delta^2 - 2\delta - O(\epsilon)}{\lambda}$

then there exists a constant $c_2 > 0$ such that with probability at least $1 - 2d \exp(\frac{-c_2 |I| \theta}{d \log^2 |I|})$ the regularised k-means SDP finds the intended cluster solution $\mathcal{C}^* = \{C_1, \ldots, C_k, \mathcal{N}_2\}$ where $B_i \subseteq C_i$ when given $\mathcal{X}$ and $\delta^2 + 2\delta \geq \lambda \geq (\delta - 1)^2 + 1$ as input.

The proof of both the Thms. 7 and 8 use the following ideas. We construct a dual for the SDP. We then show that when the conditions of our theorems are satisfied then there exists a feasible solution for the dual program. Moreover, the objective function value of primal and dual sdp program are the same. Hence, the solution found is indeed optimal. The same idea was also used in the proof of Thm. 11 in [?]. However, our analysis is tighter which helps us to obtain better bounds. The details are in the supplementary section.

We have also developed a regularised version of the k-means LP based algorithm. The details and the robustness guarantees for the LP-based algorithm are in the appendix.

5 Experiments

We ran several experiments to analyse the performance of our regularised k-means algorithm. The first set of experiments were simulations done on synthetic data. The second set of experiments were done on real world datasets like MNIST where we compared the performance of our algorithm against other popular clustering algorithms like k-means++. All our experiments were run on Matlab. We solved the SDP formulation using the Matlab SDPNAL+ package [?]. To run k-means++ we used the standard implementation of the algorithm available on Matlab.

5.1 Simulation studies

The goal of these sets of experiments was to understand the effect of different parameters on the performance of the regularised SDP algorithm. Given the number of clusters k, the separation between the clusters δ, the dimension of the space d, the number of points in each cluster n and the number of noisy points m. We generate a clustering instance $\mathcal{X}$ in $\mathbf{R}^d$ as follows. We first pick k seed points $\mu_1, \ldots, \mu_k$ such that each of these points are separated by atleast δ. Next we generate n points in the unit ball centered at each of the $\mu_i' s$. Finally we add m points uniformly at random.

We analyse the performance of the regularised SDP algorithm as the parameters change. The most crucial amongst them is the separation between the clusters δ, the regularization constant λ and the dimension d. Fig. 5.1 shows the heatmap under different parameter values. For each setting of the parameters, we generated 50 random clustering instances. We then calculated

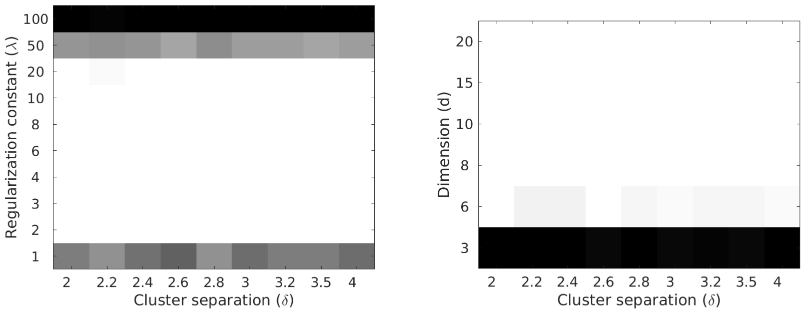

Figure 1: Heatmap showing the probability of success of the k-means regularised sdp algorithm. Lighter color indicates probability closer to one while darker indicates probability closer to zero.

the fraction of times the regularised sdp was able to recover the true clustering of the data. If the fraction is close to one, then its color on the plot is light. Darker colors represent values close to zero.

We see an interesting transition for λ. When λ is 'too small' then the probability of recovering the true clustering is also low. As λ increases the probability of success goes up which are represented by the light colors. However, if we increase λ to a very high value then the success probability again goes down. This shows that there is a 'right' range of λ as was also predicted by our theoretical analysis.

Another parameter of interest is the dimension of the space d. Note that from our theoretical analysis, we know that both the probability of success and the separation depend on d. Fig. 5.1 shows that for very low dimension, the regularised sdp fails to perfectly recover the underlying clustering. However, as the dimension grows so does the probability of success. For these two simulations, we fixed the number of points per cluster $n = 30$, $k = 8$ and the number of noisy points $m = 30$. We have similar plots for (δ, n) and (δ, k) and (δ, m) and (n, m). These plots very mostly light colored as long as the number of noisy points was not too large ($\frac{m}{n} \leq 5$). Hence, due to space constraints, we have included them only in the supplementary section.

5.2 Results on MNIST dataset

We compare our regularised SDP algorithm against k-means++ on the MNIST dataset. MNIST is a dataset of images of handwritten digits from zero to nine.

It contains 60,000 training images and 10,000 test images. We choose $k = 4$ different classes and randomly sample a total of $N = 1,000$ images from these classes. We then run both our regularised SDP algorithm and the k-means++ algorithm on this dataset. We repeat this process for 10 different random samples of MNIST. We measure the performance of the two algorithms in terms of the precision and recall over the pairs of points in the same cluster. Given a clustering $\mathcal{C}$ and some target clustering C^*. Define the precision p of $\mathcal{C}$ as the fraction of pairs that were in the same cluster according to $\mathcal{C}^*$ given that they were in the same cluster according to $\mathcal{C}$. The recall r of $\mathcal{C}$ is the fraction of pairs that were in the same clustering according to $\mathcal{C}$ given that they were in the same cluster according to $\mathcal{C}^*$. We finally measure the f_1 score of the clustering $\mathcal{C}$ as the harmonic mean of its precision and recall. $f_1 = \frac{2pr}{p+r}$.

Note that the regularised algorithm outputs $k+1$ clusters. Hence, to make a fair comparison, we finally assign each point in the noisy cluster (C_{k+1}) to one of the clusters $C_1, \ldots, C_k$ depending upon the distance of the point to the clusters. Another point is that the f_1 measures are sensitive to the choice of the k digits or classes. For some choice of k classes, the f_1 measures for both the algorithms are higher than compared to other classes. This shows that some classes are more difficult to cluster than other classes. Hence, we only report the difference in performance of the two algorithms.

We report the performance on datasets with and without noisy points. The first is when there are no outliers or noisy points. In this case, the difference in the f_1 values was about 4.34% in favor of k-means++. We then added noisy points to the dataset. In the first case, we added images from different datasets like EMNIST (images of handwritten letters). In this case, the difference was 2.54% in favor of the regularised algorithm. In the second case, besides images from different datasets, we also added a few random noisy points to the MNIST dataset. In this case, the difference increased further to about 6.9% in favor of the regularised algorithm.

6 Conclusion

We introduced a regularisation paradigm which can transform any center-based clustering objective to one that is more robust to the addition of noisy points. We proved that regularised objective is NP-Hard for common cost functions like k-means. We then obtained regularised versions of an existing clustering algorithm based on convex (sdp) relaxation of the k-means cost. We then proved noise robustness guarentees for the regularised algorithm. The proof improved existing bounds (in terms of cluster separation) for sdp-based standard (non-regularised) k-means algorithm. Our experiments showed that regularised sdp-based k-means performed better than existing algorithms like k-means++ on MNIST especially in the presence of noisy points.