A Proof of Theorem 5

In this section we prove Theorem 5.

Theorem 5. *Let S and S^* be two sets of medians with $|S| = |S^*| = k$. Suppose $p, \rho \geq 0$, and $\ell \geq 1$ is an integer. If there are no ρ-efficient ℓ-swaps on S w.r.t the penalty cost p, then we have*

$$\mathsf{cost}_p(S) \leq \sum_{j \in C} \min\left\{\left(3+\frac{2}{\ell}\right) d_p(j, S^*), \left(1+\frac{1}{\ell}\right) p\right\} + k\rho.$$

Proof. By making copies of medians, we assume S and S^* are disjoint. For every $j \in C$, define $\sigma(j)$ and $\sigma^*(j)$ to be the closest median of j in S and S^* respectively. Let $O^* = \left\{j : d_p(j, S^*) \geq \frac{\ell+1}{3\ell+2} p\right\}$; these are the points j with $\min\left\{\left(3+\frac{2}{\ell}\right) d_p(j, S^*), \left(1+\frac{1}{\ell}\right) p\right\} = \left(1+\frac{1}{\ell}\right) p$. For every $i^* \in S^*$, define $\phi(i^*)$ to be the nearest median of i^* in S, according to the metric d_p, breaking ties arbitrarily. We partition S into three parts as follows:

- $S_0 := \{i \in S : \phi^{-1}(i) = \emptyset\}$.
- $S_1 := \{i \in S : 1 \leq |\phi^{-1}(i)| \leq \ell\}$.
- $S_+ := \{i \in S : |\phi^{-1}(i)| > \ell\}$.

Let $S_1^* := \phi^{-1}(S_1)$ (which is defined as $\bigcup_{i \in S_1} \phi^{-1}(i)$) and $S_+^* := \phi^{-1}(S_+)$; thus (S_1^*, S_+^*) is a partition of S^*. Moreover,$|S_1| \leq |S_1^*|$ and $|S_+| \leq |S_+^*|/(\ell+1)$. This implies

$$\begin{aligned} |S_0| &= k - |S_1| - |S_+| \\ &\geq (|S_1^*| - |S_1|) + (k - |S_1^*|) - |S_+^*|/(\ell+1) \\ &= (|S_1^*| - |S_1|) + |S_+^*| - |S_+^*|/(\ell+1) \\ &= |S_1^*| - |S_1| + \frac{\ell}{\ell+1}|S_+^*|. \end{aligned} \tag{1}$$

We define a random mapping $\beta : S^* \to S_0 \cup S_1$ in the following way. See Figure i for the illustration of the procedure. We first define β over S_1^*. For every $i \in S_1$, we take an arbitrary $i^* \in \phi^{-1}(i)$ and define $\beta(i^*) = i$; for all other facilities $i^{*\prime}$ in $\phi^{-1}(i)$, we define $\beta(i^{*\prime})$ to be an arbitrary median in S_0. So, $|S_1|$ medians in S_1^* are mapped to S_1 by β and the remaining $|S_1^*| - |S_1|$ facilities in S_1^* are mapped to S_0. By (1), we can make β restricted to S_1^* an injective function. Moreover, at least $\frac{\ell}{\ell+1}|S_+^*|$ facilities in S_0 do not have preimages so far; call the facilities free facilities. Then, we map S_+^* to these free facilities in a random way so that each free facility is mapped to at most twice and in expectation, each free facility in expectation has at most $\left(1+\frac{1}{\ell}\right)$ pre-images in the function β.

With the random β defined, we describe a set of *test swaps* that will be used in our analysis. For every $i \in S_1$, we have a test swap $(\phi^{-1}(i), \beta(\phi^{-1}(i)))$. For every $i^* \in S_+^*$, we have a test swap $(\{i^*\}, \{\beta(i^*)\})$. It is easy to see that each test swap (A^*, A) has $A^* \subseteq F^*, A \subseteq F$ and $|A^*| = |A| \leq \ell$. Moreover, we have the following properties:

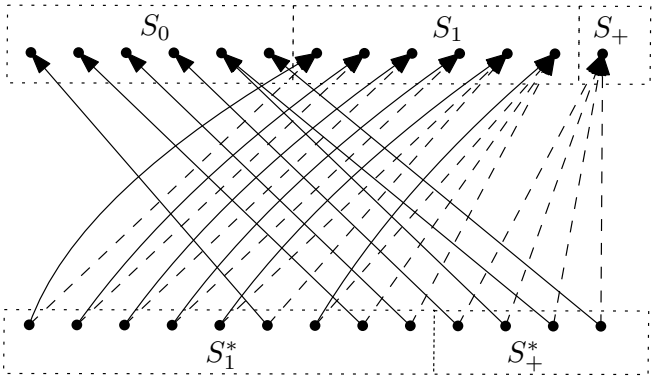


Figure i: The definition of the function β. The vertices at the top are S, the vertices at the bottom S^*, $\ell = 3$, and the dashed lines give the definition of ϕ. Then $S_0, S_1, S_+, S_1^*, S_+^*$ are depicted in the figure, and a possible function β is given by the solid lines and curves.

(P1) Every median in $i^* \in S^*$ is swapped in exactly once in all test swaps.

(P2) In expectation over all possible β's, every median in $i \in S$ is swapped out at most $1 + \frac{1}{\ell}$ times in the test swaps.

(P3) For any test swap (A^*, A), we have $\phi^{-1}(A) \subseteq A^*$.

(P1) and (P2) follow from the construction of β. To see (P3), consider the two types of test swaps. If the test swap is $(\{i^*\}, \{\beta(i^*)\})$ for some $i^* \in S_+^*$, then $\beta(i^*) \in S_0$ and thus $\phi^{-1}(\beta(i^*)) = \emptyset$. If the test swap is $(\phi^{-1}(i), \beta(\phi^{-1}(i)))$ for some $i \in S_1$, then $\beta(\phi^{-1}(i))$ contains i and all the other elements in the set are in S_0. Thus $\phi^{-1}(\beta(\phi^{-1}(i))) = \phi^{-1}(i)$.

Focus on a fixed test swap (A^*, A). After opening A^* and closing A, we can reconnect a subset of points in $\sigma^{-1}(A) \cup \sigma^{*-1}(A^*)$. We guarantee that all points in $\sigma^{-1}(j)$ will be reconnected. See Figure ii for how we reconnect the points.

- For a point $j \in \sigma^{*-1}(A^*) \setminus O^*$, we reconnect j from $\sigma(j)$ to $\sigma^*(j) \in A^*$. The decrease in the connection cost of j is $d_p(j, \sigma(j)) - d_p(j, \sigma^*(j)) = d_p(j, S) - d_p(j, S^*)$.
- For a point $j \in \sigma^{-1}(A) \setminus \sigma^{*-1}(A^*) \setminus O^*$, we reconnect j to $\phi(\sigma^*(j))$. Notice that $\sigma^*(j) \notin A^*$. By (P3), we have $\phi(\sigma^*(j)) \notin A$. Thus the connection is valid. By triangle inequalities and definition of ϕ, for every

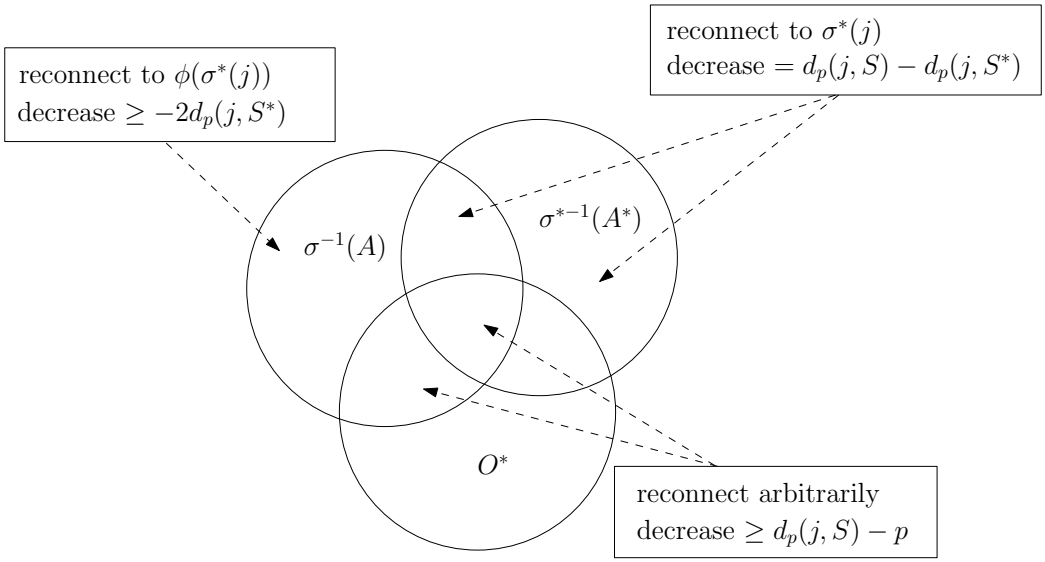


Figure ii: How to reconnect points and the lower bound for the decrement in the connection cost for each point j, using the Venn diagram for the three sets $\sigma^{-1}(A), \sigma^{*-1}(A^*)$ and O^*.

$j \in \sigma^{-1}(A) \setminus \sigma^{*-1}(A^*) \setminus O^*$, we have

$$\begin{aligned} & d_p(j, \phi(\sigma^*(j))) \\ \leq\ & d_p(j, \sigma^*(j)) + d_p(\sigma^*(j), \phi(\sigma^*(j))) \\ \leq\ & d_p(j, \sigma^*(j)) + d_p(\sigma^*(j), \sigma(j)) \\ \leq\ & d_p(j, \sigma^*(j)) + d_p(j, \sigma^*(j)) + d_p(j, \sigma(j)) \\ =\ & 2d_p(j, \sigma^*(j)) + d_p(\sigma(j), j). \end{aligned}$$

So the decrease in the connection cost of j is $d_p(j, \sigma(j)) - d_p(j, \phi(\sigma^*(j))) \geq -2d_p(j, \sigma^*(j)) = -2d_p(j, S^*)$.

- For a point $j \in \sigma^{-1}(A) \cap O^*$, we reconnect j arbitrarily, and the decrease in the connection cost of j is at least $d_p(j, \sigma(j)) - p = d(j, S) - p$ as p is the diameter of the metric d_p.

As the test swap operation is not ρ-efficient, we have

$$\begin{aligned} & \sum_{j \in \sigma^{*-1}(A^*) \setminus O^*} (d_p(j, S) - d_p(j, S^*)) - \\ & 2 \sum_{j \in \sigma^{-1}(A) \setminus O^*} d_p(j, S^*) + \sum_{j \in \sigma^{-1}(A) \cap O^*} (d_p(j, S) - p) \\ \leq\ & |A|\rho. \end{aligned} \tag{2}$$

Above, we used that that $\sigma^{-1}(A) \setminus \sigma^{*-1}(A^*) \setminus O^* \subseteq \sigma^{-1}(A) \setminus O^*$.

We now add up (2) over all test swap operations. We consider the expectation of the left side of the summation, over all random choices of β:

- The sum of the first term on the left side of (2) is always exactly $\sum_{j \in C \setminus O^*} \big(d_p(j, S) - d_p(j, S^*)\big)$, due to (P1).
- Consider the expectation of the sum of the second term on the left side of (2). Since each $i \in S$ is swapped out in at most $1 + \frac{1}{\ell}$ times in expectation by (P2), the expectation of the sum of the second term is at least $-\left(2 + \frac{2}{\ell}\right) \sum_{j \in C \setminus O^*} d_p(j, S^*)$.
- Consider the expectation of the sum of the third term on the left side of (2). Using that d_p has diameter at most p, and (P2), the expectation is at least $\left(1 + \frac{1}{\ell}\right) \sum_{j \in O^*} (d_p(j, S) - p) \geq \sum_{j \in O^*} d_p(j, S) - \left(1 + \frac{1}{\ell}\right) |O^*|p$. We changed the coefficient before a non-negative term from $\left(1 + \frac{1}{\ell}\right)$ to 1 in the inequality; this is sufficient.

Overall, the expectation of the sum of the left side of (2) over all test swap operations is at least

$$\begin{aligned} & \sum_{j \in C \setminus O^*} \big(d_p(j, S) - d_p(j, S^*)\big) \\ & \quad - \left(2 + \frac{2}{\ell}\right) \sum_{j \in C \setminus O^*} d_p(j, S^*) \\ & \quad + \sum_{j \in O^*} d_p(j, S) - \left(1 + \frac{1}{\ell}\right) |O^*|p \\ =\ & \sum_{j \in C} d_p(j, S) - \left(3 + \frac{2}{\ell}\right) \sum_{j \in C \setminus O^*} d_p(j, S^*) \\ & \quad - |O^*| \cdot \left(1 + \frac{1}{\ell}\right) p \\ =\ & \sum_{j \in C} d_p(j, S) - \sum_{j \in C} \min \left\{ \left(3 + \frac{2}{\ell}\right) d_p(j, S^*), \right. \\ & \quad \left. \left(1 + \frac{1}{\ell}\right) p \right\}, \end{aligned}$$

where the last equality used the definition of O^*.

The summation of the right side of (2) over all test swaps is always exactly $k\rho$. Therefore, we have

$$\sum_{j\in C} d_p(j,S) - \sum_{j\in C} \min\left\{\left(3+\frac{2}{\ell}\right)d_p(j,S^*), \left(1+\frac{1}{\ell}\right)p\right\} \leq k\rho.$$

Rearranging the terms and replacing $\sum_{j\in C} d_p(j,S)$ with $\mathsf{cost}_p(S)$ finish the proof of the theorem. □

B Experiments

In this section, we corroborate our theoretical findings by performing experiments on real world datasets[8]. Our goal is to empirically show that the local search algorithm is stable and does few reclusterings, while maintaining a good approximation factor.

Algorithm implementation: We modified our algorithm slightly to make it faster: when a new data point comes, instead of conducting local search directly, we assign the point to its nearest center; then we check whether the current cost is at least $(1+\alpha)$ times the cost *resulting from the last application of local search*, and if not we continue to the next data point without doing any local operations. It is easy to see that this will increase our approximation ratio by a $(1+\alpha)$ factor. Though this modification doesn't improve our worst-case recourse bound, it reduces the number of local operations needed when the incoming data are non-adversarial, which is often the case in practice. Throughout the experiment we set $\alpha = 0.2$.

Data set and parameter setting: We follow the experiment setting in Lattanzi and Vassilvitskii (2017). The algorithm is tested on three UCI data sets Lichman (2013): (i) Skin with 245,057 data points of dimension 4; (ii) Covertype with 581,012 data points of dimension 54; In the experiment we'll only use the first 10 features of Covertype because other features are categorical. (iii) Letter with 20,000 data points of dimension 16. To keep the duration of experiments short, we restrict the experiments to the first 10K data points in each data set. We set the algorithm parameters $\epsilon = 0.05$ and $\gamma = 1$; these were chosen to minimize the number of discarded outliers. We set the available center locations $F = C$, so when a new data point comes, it will be added to both F and C. Throughout the experiment, we set the number of outliers to be $z = 200$, and tried three different values of $k \in \{10, 50, 100\}$. We observe that in all the runs, our algorithm removes at most 840 outliers, hence achieving an approximation factor of 4.2 on the number of discarded outliers.

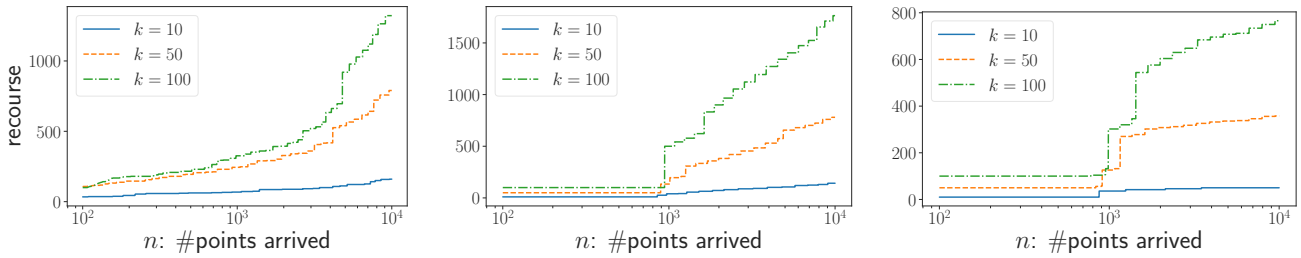
[8]The code can be found at `https://github.com/xyguo/OnlineKZMedian`

Results: We first show the how the recourse grows overtime in Figure iii. One can observe that the recourse dependence on k is roughly $O(k \log n)$ instead of the $O(k^2 \log n \log(nD))$ worst-case bound predicted by our theoretical result. We also observe that the growth rate of recourse is lower for Covertype and Letter data sets compared to Skin. This is because of the data ordering in Skin; if we randomly shuffle the Skin data set and re-run the algorithm then we get a graph similar to the other two data sets.

Now we turn to the quality of clustering maintained by our algorithm. Since the optimal solution is hard to compute, we follow the setting of Lattanzi and Vassilvitskii (2017) and use the clustering produced by offline k-means−− algorithmChawla and Gionis (2013) as an coarse estimation of OPT. Specifically, for every 50 newly-arrived data points, we compute 5 offline k-means−− solutions (with different initializations) for all already arrived data points, and choose the best one as the estimation for OPT at this time point. Then we linearly interpolate between these estimations to get an OPT curve for every time point. Figure iv shows the estimated approximation ratio over time. *We see that the ratio is bounded by* 1.5 *most of the times.* One might notice that the ratio sometimes even falls below 1. This is because of two reasons: 1) we only have an estimate of the real OPT; 2) the bi-criteria approximation means our algorithm might remove more than z outliers, while the OPT is calculated by removing at most z outliers.

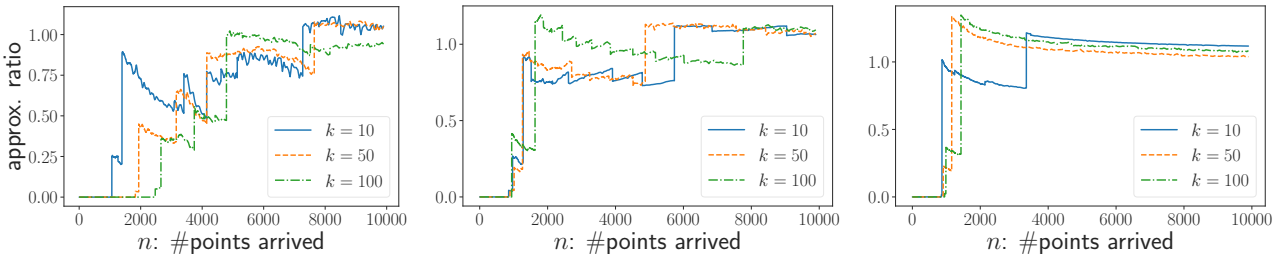
Results for incremental-z: In practice it might be more reasonalbe to allow the number of outliers grow with the amount of accumulated data. Here we include experiment results for this setting. We let z grow uniformly as follows: we still focus on the first 10K data points, and for each time point $t \in [1, 10000]$, we set the number of allowed outliers $z_t = \frac{t}{10000} \times 200$. So as more data points come, we allow to remove more outliers. All other parameters are the same as before: $\epsilon = 0.05, \gamma = 1, k \in \{10, 50, 100\}$, and available center locations $F = C$.

Figure v shows how the total recourse grows with time. One can see that it's largely the same as that in Figure iii, exhibiting an $O(k)$ dependence on k and $O(\log n)$ dependence on n. The major difference is that the recourse starts growing in very early time stages, while in Figure iii there's a longer warm-up phase. This is because in the setting of Figure iii the algorithm is allowed to remove roughly $4z = 800$ outliers from the beginning, which means it can simply ignore the first few hundred arrived data points and



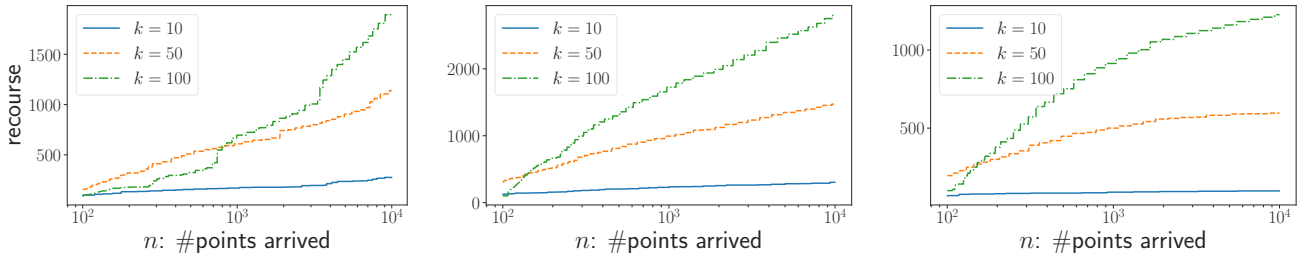
(a) Skin (b) Covertype (c) Letter

Figure iii: Recourse over time. The x-axis is plotted in the log-scale



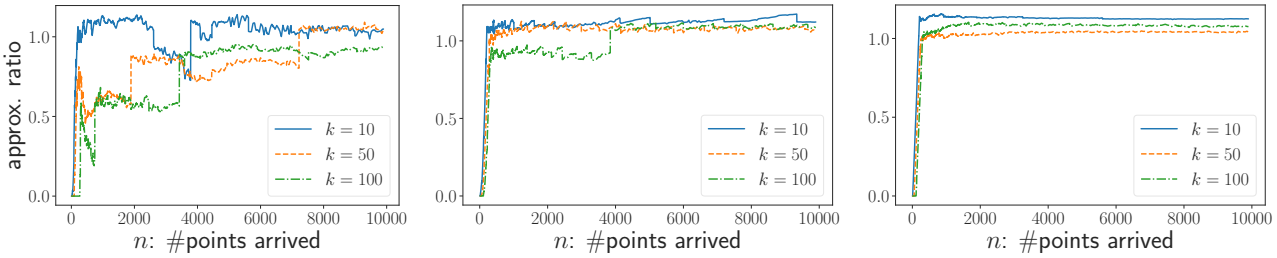
(a) Skin (b) Covertype (c) Letter

Figure iv: Estimated approximation ratio over time.



(a) Skin (b) Covertype (c) Letter

Figure v: Incremental-z setting: Recourse over time. The x-axis is plotted in the log-scale



(a) Skin (b) Covertype (c) Letter

Figure vi: Incremental-z setting: Estimated approximation ratio over time.

conduct no local operations, i.e., no recourse. Figure vi shows the clustering quality on the three data sets. One can see that our algorithm still achieves very good approximation ratio (nearly 1) on all three data sets.