A EXPERIMENT ON NON-L_p METRICS

To demonstrate our extension to non-L_p metrics, we conduct an experiment on our synthetic datasets under two popular non-L_p metrics: chi-square histogram metric and Hellinger metric.

Experimental Setup. In our synthetic datasets, we use a feature vector $(x_1, x_2, \cdots, x_d)$ to denote an object u (*i.e.*, x_i is the coordinate of the object u), where d is the dimension of the dataset. Similarly, we use a feature vector $(y_1, y_2, \cdots, y_d)$ to represent another object v (*i.e.*, y_i is the coordinate of the objective v). Accordingly, the *chi-square histogram distance* [11, 49] between u and v is defined as the square root of the chi-square distance [53] in Eq. (20) and has been proved to be a distance metric in [49].

$$Dis_{\text{chi-square histogram}}(u, v) = \sqrt{\frac{1}{2}\sum_{i=1}^{d}\frac{(x_i - y_i)^2}{x_i + y_i}} \tag{20}$$

The Hellinger distance [53] between u and v is defined in Eq. (21).

$$Dis_{\text{Hellinger}}(u, v) = \frac{1}{\sqrt{2}}\sqrt{\sum_{i=1}^{d}(\sqrt{x_i} - \sqrt{y_i})^2} \tag{21}$$

To use the synthetic datasets in these metrics, we normalize each coordinate into the range of $[0, 1]$. Besides, we use the default setting of n (*i.e.*, $n = 5 \times 10^4$) and set the dimension d as 5. Notice, we only compare our proposed algorithm DCsam with the state-of-the-art baseline FRT, because FRT is the most efficient baseline and DCsam is the most efficient implementation of our proposed framework in Sec. 5. For the approximate nearest neighbor (ANN) search used in DCsam, we use a popular library called FLANN [47, 48], which supports several non-L_p metrics commonly seen in real applications such as image retrieval and texture classification. The other setup such as the experimental environment is the same as that in Sec. 5.

Experimental Result. Table 8 presents our experimental results under the *chi-square histogram metric*. First, we can observe that the distortion of our DCsam is always lower than that of the state-of-the-art baseline FRT. For example, on the *Exp* dataset, the distortion of DCsam is 2.2× lower than the distortion of FRT. Second, the time efficiency of DCsam is also always better than that of FRT. For instance, DCsam is 7.1× faster than FRT on the skewed dataset *Skew*. Third, in terms of the space cost, FRT takes 3.5MB space on average and DCsam takes 3.6MB space on average. Therefore, the space cost of DCsam is comparably efficient with that of FRT, since their gap is very marginal (*i.e.*, 0.1MB).

Table 9 lists our experimental results under the *Hellinger metric*. In terms of *distortion*, our proposed algorithm DCsam is always more effective than the baseline FRT. As for the *time cost*, DCsam is 8.4×-10.5× faster than FRT in these datasets. The results of their *space cost* are very similar to those under the chi-square histogram metric. For example, FRT consumes 3.5MB space on average and DCsam takes 3.6MB space on average. This is because the space cost of the HST constructed by either DCsam or FRT is $O(n)$, where n is the number of points and n is a constant (*i.e.*, 5×10^4) in these four datasets.

Summary. The main experimental findings are

- By using proper ANN algorithms, our solution can be easily extended to support non-L_p metrics.
- Under the chi-square histogram metric and Hellinger metric, which are popular non-L_p metrics, our proposed solution can still achieve better effectiveness (*i.e.*, lower distortion) and faster running time than the existing baseline (*e.g.*, the state-of-the-art FRT). For example, the distortion of DCsam is up to 2.2× lower than that of FRT. Moreover, DCsam is up to 10.5× faster than FRT.

B EXPERIMENT ON INSERTION AND DELETION

To verify our extension to the insertion/deletion scenario, we conduct an experiment on four real datasets, which are used to test this scenario in Ref. [64].

Experimental Setup. In this experiment, we also compare our proposed algorithm DCsam with the state-of-the-art baseline FRT. Specifically, as mentioned in Sec. 4.4, we only need to replace the construction procedure (*i.e.*, FRT) in [64] with DCsam. As we have shown DCsam can achieve better effectiveness and lower time cost than FRT for the static data, we are expecting that our extension has a lower distortion and better time efficiency than FRT under the insertion and deletion scenario. To process our real datasets in Table 5, we follow the method in [64] to generate the insertions and deletions. As suggested in [64], we generate 10 batches of updates and evaluate the average result of distortion and time cost over these 10 batches. The other setup such as the experimental environment is the same as that in Sec. 5.

Experimental Result. Table 10 presents our experimental result under the insertion and deletion scenario. First, we can observe that the distortion of our proposed algorithm DCsam is always lower than that of FRT. This pattern validates that our extension to this dynamic scenario can lead to better effectiveness. Second, the running time of DCsam is shorter than that of FRT, which proves our extension can also improve the time efficiency. For instance, DCsam is 16.2× faster than FRT on the *Haikou* dataset. Finally, in terms of space cost, DCsam takes 19MB-56MB space and FRT takes 19MB-55MB space. Thus, both methods are relatively efficient, since the difference of their space consumption is always below 1MB, which is marginal considering the RAM size of a modern server.

Summary. The main experimental findings are

- Based on the experiment, we can easily extend our proposed algorithm to the insertion and deletion scenario.
- Our extension still achieves better effectiveness and time efficiency. For example, DCsam is 16.2× faster than FRT on the *Haikou* dataset, where DCsam is our extension and FRT represents the state-of-the-art baseline proposed in [64].

Table 8: Comparison between our DCsam and the state-of-the-art FRT [26, 64] under the chi-square histogram metric

Dataset	*Uni*		*Nor*		*Exp*		*Skew*	
Metric	Distortion	Time cost (sec)	Distortion	Time cost (sec)	Distortion	Time cost (sec)	Distortion	Time cost (sec)
FRT	571.5	37.3	495.9	38.1	723.9	36.3	1493.5	41.3
DCsam	417.1	5.7	366.9	5.5	321.1	5.5	738.7	5.1

Table 9: Comparison between our DCsam and the state-of-the-art FRT [26, 64] under the Hellinger metric

Dataset	*Uni*		*Nor*		*Exp*		*Skew*	
Metric	Distortion	Time cost (sec)	Distortion	Time cost (sec)	Distortion	Time cost (sec)	Distortion	Time cost (sec)
FRT	601.8	61.8	495.9	61.7	677.4	62.1	1102.6	61.0
DCsam	330.9	6.1	404.9	6.5	356.0	6.6	809.1	5.3

Table 10: Comparison between our DCsam and the state-of-the-art FRT [26, 64] under the insertion/deletion scenario

Dataset	*NYC*		*Tokyo*		*Chengdu*		*Haikou*	
Metric	Distortion	Time cost (sec)	Distortion	Time cost (sec)	Distortion	Time cost (sec)	Distortion	Time cost (sec)
FRT	7237.27	1.30	4322.81	3.61	22812.95	73.08	8314.03	205.55
DCsam	2935.85	0.29	4079.75	0.67	11483.87	6.42	5629.57	11.95