Experiments:

1. pam_uni(n=3,850,505, dim=4):

   -All the experiments below have the same parameters about "main.cpp"

   data_file_name = "data/pam_uni.txt";

   new_data_file_name = "data/new_pam_uni_20W.txt";

   leaf_percent = 0.001;

   // Sample-based semi-convex hull tree.app main()

   ① Test the performance of "Sample-based semi-convex hull tree" KNN algorithm for K between 10 and 150

   FILE * log_file = fopen("exp_log_kNN(gpu_pam_20W).txt", "a+");

   test_gpu_KNN(…, K = 10, leaf_percent, log_file); 0.66s
   test_gpu_KNN(…, K = 20, leaf_percent, log_file); 0.97s
   test_gpu_KNN(…, K = 30, leaf_percent, log_file); 1.73s
   test_gpu_KNN(…, K = 40, leaf_percent, log_file); 2.66s
   test_gpu_KNN(…, K = 50, leaf_percent, log_file); 4.03s
   test_gpu_KNN(…, K = 70, leaf_percent, log_file); 7.29s
   test_gpu_KNN(…, K = 100, leaf_percent, log_file); 13.55s
   test_gpu_KNN(…, K = 120, leaf_percent, log_file); 18.24s
   test_gpu_KNN(…, K = 150, leaf_percent, log_file); 27.01s

   sample_percent = 0.1

   FILE* log_file = fopen("exp_log_kNN(gpu_improved_pam_20W_10%).txt", "a+");

   test_gpu_KNN_imp_sample(…, K = 10, leaf_percent, …); 0.72s
   test_gpu_KNN_imp_sample(…, K = 20, leaf_percent, …); 0.95s
   test_gpu_KNN_imp_sample(…, K = 30, leaf_percent, …); 1.31s
   test_gpu_KNN_imp_sample(…, K = 40, leaf_percent, …); 1.79s
   test_gpu_KNN_imp_sample(…, K = 50, leaf_percent, …); 2.49s
   test_gpu_KNN_imp_sample(…, K = 70, leaf_percent, …); 3.81s
   test_gpu_KNN_imp_sample(…, K = 100, leaf_percent, …); 6.43s
   test_gpu_KNN_imp_sample(…, K = 120, leaf_percent, …); 8.20s
   test_gpu_KNN_imp_sample(…, K = 150, leaf_percent, …); 12.23s

   sample_percent = 0.15

   FILE* log_file = fopen("exp_log_kNN(gpu_improved_pam_20W_15%).txt", "a+");

   test_gpu_KNN_imp_sample(…, K = 10, leaf_percent, …); 0.80s
   test_gpu_KNN_imp_sample(…, K = 20, leaf_percent, …); 1.12s
   test_gpu_KNN_imp_sample(…, K = 30, leaf_percent, …); 1.59s
   test_gpu_KNN_imp_sample(…, K = 40, leaf_percent, …); 2.21s
   test_gpu_KNN_imp_sample(…, K = 50, leaf_percent, …); 3.11s
   test_gpu_KNN_imp_sample(…, K = 70, leaf_percent, …); 5.42s
   test_gpu_KNN_imp_sample(…, K = 100, leaf_percent, …); 9.18s

test_gpu_KNN_imp_sample(…, K = 120, leaf_percent, …); 11.65s
test_gpu_KNN_imp_sample(…, K = 150, leaf_percent, …); 17.01s

sample_percent = 0.2
FILE* log_file = fopen("exp_log_kNN(gpu_improved_pam_20W_20%).txt", "a+");
test_gpu_KNN_imp_sample(…, K = 10, leaf_percent, …); 0.79s
test_gpu_KNN_imp_sample(…, K = 20, leaf_percent, …); 0.91s
test_gpu_KNN_imp_sample(…, K = 30, leaf_percent, …); 1.38s
test_gpu_KNN_imp_sample(…, K = 40, leaf_percent, …); 2.01s
test_gpu_KNN_imp_sample(…, K = 50, leaf_percent, …); 2.87s
test_gpu_KNN_imp_sample(…, K = 70, leaf_percent, …); 4.65s
test_gpu_KNN_imp_sample(…, K = 100, leaf_percent, …); 8.19s
test_gpu_KNN_imp_sample(…, K = 120, leaf_percent, …); 10.29s
test_gpu_KNN_imp_sample(…, K = 150, leaf_percent, …); 15.55s

② Test the performance of "Sample-based semi-convex hull tree" KNN algorithm for sample_percent between 0.01 and 0.15 when K is equal to 150.

FILE* log_file = fopen("exp_log_kNN(gpu_improved_pam_20W_1%).txt", "a+");
test_gpu_KNN_imp_sample(…, sample_percent = 0.01, …); 56.92s
FILE* log_file = fopen("exp_log_kNN(gpu_improved_pam_20W_2%).txt", "a+");
test_gpu_KNN_imp_sample(…, sample_percent = 0.02, …); 70.95s
FILE* log_file = fopen("exp_log_kNN(gpu_improved_pam_20W_3%).txt", "a+");
test_gpu_KNN_imp_sample(…, sample_percent = 0.03, …); 32.73s
FILE* log_file = fopen("exp_log_kNN(gpu_improved_pam_20W_4%).txt", "a+");
test_gpu_KNN_imp_sample(…, sample_percent = 0.04, …); 17.26s
FILE* log_file = fopen("exp_log_kNN(gpu_improved_pam_20W_5%).txt", "a+");
test_gpu_KNN_imp_sample(…, sample_percent = 0.05, …); 14.40s
FILE* log_file = fopen("exp_log_kNN(gpu_improved_pam_20W_7%).txt", "a+");
test_gpu_KNN_imp_sample(…, sample_percent = 0.07, …); 13.01s
FILE* log_file = fopen("exp_log_kNN(gpu_improved_pam_20W_10%).txt", "a+");
test_gpu_KNN_imp_sample(…, sample_percent = 0.1, …); 13.82s
FILE* log_file = fopen("exp_log_kNN(gpu_improved_pam_20W_12%).txt", "a+");
test_gpu_KNN_imp_sample(…, sample_percent = 0.12, …); 15.41s
FILE* log_file = fopen("exp_log_kNN(gpu_improved_pam_20W_15%).txt", "a+");
test_gpu_KNN_imp_sample(…, sample_percent = 0.15, …); 17.01s

2. house_uni(n=1,906,698, dim=7):

-All the experiments below have the same parameters about "main.cpp"
l data_file_name = "data/house_uni.txt";
new_data_file_name = "data/new_house_uni_20W.txt";
leaf_percent = 0.001;

// Sample-based semi-convex hull tree.app main()

① Test the performance of "Sample-based semi-convex hull tree" KNN algorithm for K between 10 and 150

```
FILE * log_file = fopen("exp_log_kNN(gpu_house_20W).txt", "a+");
test_gpu_KNN(…, K = 10, leaf_percent, log_file);                    0.69s
test_gpu_KNN(…, K = 20, leaf_percent, log_file);                    0.82s
test_gpu_KNN(…, K = 30, leaf_percent, log_file);                    1.17s
test_gpu_KNN(…, K = 40, leaf_percent, log_file);                    1.52s
test_gpu_KNN(…, K = 50, leaf_percent, log_file);                    2.36s
test_gpu_KNN(…, K = 70, leaf_percent, log_file);                    4.40s
test_gpu_KNN(…, K = 100, leaf_percent, log_file);                   8.51s
test_gpu_KNN(…, K = 120, leaf_percent, log_file);                   11.80s
test_gpu_KNN(…, K = 150, leaf_percent, log_file);                   17.61s

sample_percent = 0.1
FILE* log_file = fopen("exp_log_kNN(gpu_improved_house_20W_10%).txt", "a+");
test_gpu_KNN_imp_sample(…, K = 10, leaf_percent, …);                0.82s
test_gpu_KNN_imp_sample(…, K = 20, leaf_percent, …);                0.98s
test_gpu_KNN_imp_sample(…, K = 30, leaf_percent, …);                1.27s
test_gpu_KNN_imp_sample(…, K = 40, leaf_percent, …);                1.66s
test_gpu_KNN_imp_sample(…, K = 50, leaf_percent, …);                2.24s
test_gpu_KNN_imp_sample(…, K = 70, leaf_percent, …);                3.44s
test_gpu_KNN_imp_sample(…, K = 100, leaf_percent, …);               6.16s
test_gpu_KNN_imp_sample(…, K = 120, leaf_percent, …);               8.07s
test_gpu_KNN_imp_sample(…, K = 150, leaf_percent, …);               12.86s

sample_percent = 0.15
FILE* log_file = fopen("exp_log_kNN(gpu_improved_house_20W_15%).txt", "a+");
test_gpu_KNN_imp_sample(…, K = 10, leaf_percent, …);                0.81s
test_gpu_KNN_imp_sample(…, K = 20, leaf_percent, …);                1.00s
test_gpu_KNN_imp_sample(…, K = 30, leaf_percent, …);                1.38s
test_gpu_KNN_imp_sample(…, K = 40, leaf_percent, …);                1.85s
test_gpu_KNN_imp_sample(…, K = 50, leaf_percent, …);                2.46s
test_gpu_KNN_imp_sample(…, K = 70, leaf_percent, …);                3.89s
test_gpu_KNN_imp_sample(…, K = 100, leaf_percent, …);               6.79s
test_gpu_KNN_imp_sample(…, K = 120, leaf_percent, …);               8.89s
test_gpu_KNN_imp_sample(…, K = 150, leaf_percent, …);               13.69s

sample_percent = 0.2
FILE* log_file = fopen("exp_log_kNN(gpu_improved_house_20W_20%).txt", "a+");
test_gpu_KNN_imp_sample(…, K = 10, leaf_percent, …);                0.81s
test_gpu_KNN_imp_sample(…, K = 20, leaf_percent, …);                1.01s
test_gpu_KNN_imp_sample(…, K = 30, leaf_percent, …);                1.33s
```

test_gpu_KNN_imp_sample(…, K = 40, leaf_percent, …); 1.79s
test_gpu_KNN_imp_sample(…, K = 50, leaf_percent, …); 2.42s
test_gpu_KNN_imp_sample(…, K = 70, leaf_percent, …); 3.89s
test_gpu_KNN_imp_sample(…, K = 100, leaf_percent, …); 6.78s
test_gpu_KNN_imp_sample(…, K = 120, leaf_percent, …); 8.71s
test_gpu_KNN_imp_sample(…, K = 150, leaf_percent, …); 13.01s

② Test the performance of "Sample-based semi-convex hull tree" KNN algorithm for sample_percent between 0.01 and 0.15 when K is equal to 150.

FILE* log_file = fopen("exp_log_kNN(gpu_improved_house_20W_1%).txt", "a+");
test_gpu_KNN_imp_sample(…, sample_percent = 0.01, …); 62.36s
FILE* log_file = fopen("exp_log_kNN(gpu_improved_house_20W_2%).txt", "a+");
test_gpu_KNN_imp_sample(…, sample_percent = 0.02, …); 77.52s
FILE* log_file = fopen("exp_log_kNN(gpu_improved_house_20W_3%).txt", "a+");
test_gpu_KNN_imp_sample(…, sample_percent = 0.03, …); 63.12s
FILE* log_file = fopen("exp_log_kNN(gpu_improved_house_20W_4%).txt", "a+");
test_gpu_KNN_imp_sample(…, sample_percent = 0.04, …); 47.72s
FILE* log_file = fopen("exp_log_kNN(gpu_improved_house_20W_5%).txt", "a+");
test_gpu_KNN_imp_sample(…, sample_percent = 0.05, …); 35.04s
FILE* log_file = fopen("exp_log_kNN(gpu_improved_house_20W_7%).txt", "a+");
test_gpu_KNN_imp_sample(…, sample_percent = 0.07, …); 18.67s
FILE* log_file = fopen("exp_log_kNN(gpu_improved_house_20W_10%).txt", "a+");
test_gpu_KNN_imp_sample(…, sample_percent = 0.1, …); 12.94s
FILE* log_file = fopen("exp_log_kNN(gpu_improved_house_20W_12%).txt", "a+");
test_gpu_KNN_imp_sample(…, sample_percent = 0.12, …); 13.05s
FILE* log_file = fopen("exp_log_kNN(gpu_improved_house_20W_15%).txt", "a+");
test_gpu_KNN_imp_sample(…, sample_percent = 0.15, …); 13.57s

3. USCensus1990_uni(n=365,100, dim=8):

-All the experiments below have the same parameters about "main.cpp"
l data_file_name = "data/USCensus1990_uni.txt";
new_data_file_name = "data/new_ USCensus1990_uni_20W.txt";
leaf_percent = 0.008;

// Sample-based semi-convex hull tree.app main()

① Test the performance of "Sample-based semi-convex hull tree" KNN algorithm for K between 20 and 220

FILE * log_file = fopen("exp_log_kNN(gpu_uscensus_20W).txt", "a+");
test_gpu_KNN(…, K = 20, leaf_percent, log_file); 1.03s

test_gpu_KNN(…, K = 50, leaf_percent, log_file); 1.98s
test_gpu_KNN(…, K = 70, leaf_percent, log_file); 3.46s
test_gpu_KNN(…, K = 100, leaf_percent, log_file); 6.87s
test_gpu_KNN(…, K = 120, leaf_percent, log_file); 9.22s
test_gpu_KNN(…, K = 150, leaf_percent, log_file); 14.98s
test_gpu_KNN(…, K = 170, leaf_percent, log_file); 18.95s
test_gpu_KNN(…, K = 200, leaf_percent, log_file); 26.05s
test_gpu_KNN(…, K = 220, leaf_percent, log_file); 31.50s

sample_percent = 0.1
FILE* log_file = fopen("exp_log_kNN(gpu_improved_usensus_20W_10%).txt", "a+");
test_gpu_KNN_imp_sample(…, K = 20, leaf_percent, …); 1.17s
test_gpu_KNN_imp_sample(…, K = 50, leaf_percent, …); 2.41s
test_gpu_KNN_imp_sample(…, K = 70, leaf_percent, …); 3.83s
test_gpu_KNN_imp_sample(…, K = 100, leaf_percent, …); 6.55s
test_gpu_KNN_imp_sample(…, K = 120, leaf_percent, …); 8.40s
test_gpu_KNN_imp_sample(…, K = 150, leaf_percent, …); 12.42s
test_gpu_KNN_imp_sample(…, K = 170, leaf_percent, …); 15.66s
test_gpu_KNN_imp_sample(…, K = 200, leaf_percent, …); 21.77s
test_gpu_KNN_imp_sample(…, K = 220, leaf_percent, …); 26.93s

sample_percent = 0.15
FILE* log_file = fopen("exp_log_kNN(gpu_improved_uscensus_20W_15%).txt", "a+");
test_gpu_KNN_imp_sample(…, K = 20, leaf_percent, …); 1.14s
test_gpu_KNN_imp_sample(…, K = 50, leaf_percent, …); 2.30s
test_gpu_KNN_imp_sample(…, K = 70, leaf_percent, …); 3.90s
test_gpu_KNN_imp_sample(…, K = 100, leaf_percent, …); 6.83s
test_gpu_KNN_imp_sample(…, K = 120, leaf_percent, …); 8.77s
test_gpu_KNN_imp_sample(…, K = 150, leaf_percent, …); 13.17s
test_gpu_KNN_imp_sample(…, K = 170, leaf_percent, …); 16.36s
test_gpu_KNN_imp_sample(…, K = 200, leaf_percent, …); 21.77s
test_gpu_KNN_imp_sample(…, K = 220, leaf_percent, …); 25.76s

sample_percent = 0.2
FILE* log_file = fopen("exp_log_kNN(gpu_improved_uscensus_20W_20%).txt", "a+");
test_gpu_KNN_imp_sample(…, K = 20, leaf_percent, …); 1.16s
test_gpu_KNN_imp_sample(…, K = 50, leaf_percent, …); 2.35s
test_gpu_KNN_imp_sample(…, K = 70, leaf_percent, …); 3.73s
test_gpu_KNN_imp_sample(…, K = 100, leaf_percent, …); 6.80s
test_gpu_KNN_imp_sample(…, K = 120, leaf_percent, …); 8.91s
test_gpu_KNN_imp_sample(…, K = 150, leaf_percent, …); 13.34s
test_gpu_KNN_imp_sample(…, K = 170, leaf_percent, …); 16.84s
test_gpu_KNN_imp_sample(…, K = 200, leaf_percent, …); 22.23s
test_gpu_KNN_imp_sample(…, K = 220, leaf_percent, …); 26.80s