Carnegie Mellon University
15-826 – Multimedia Databases and Data Mining
Fall 2014, C. Faloutsos
Homework 1
Due Date: Oct 2nd, 3:00pm, in class

By: Yuning He (Q1, Q2) & Yan Zhang (Q3, Q4).

IMPORTANT REMINDERS

- All homeworks are to be done **INDIVIDUALLY**.
- All written answers should be **TYPED**.
- For code submission to blackboard, make three directories /Q1, /Q2, /Q4, and then put your code for question 1, 2, 4 to the corresponding directory (Q3 has no code to deliver). Then `tar` them, compress them into a file (`[andrew-id].tar.gz`) and submit it to blackboard. As always, make sure you **exclude** redundant/derived files, in your tar-file.

Other reminders, FYI

- Weight: 30% of total homeworks weight = 3% of course weight.
- Expected effort for this homework (order-of-magnitude):
  - Q1: $\approx$ 8-10 hours
  - Q2: $\approx$ 6-8 hours
  - Q3: $\approx$ 2-3 hours
  - Q4: $\approx$ 2-3 hours

Q1 – R-Tree [40 pts]

Print answers on separate page, with '[course-id] [hw#] [question#] [andrew-id] [your name]'

Problem Description: The goal is to become familiar with the R-Tree algorithm. Your task is to add new functionality to the provided R-Tree package[1] from [`http://www.cs.cmu.edu/~christos/courses/826.F14/HOMEWORKS/HW1/Q1/drtree.tar.gz`]

Setup: Please build the R-Tree Package

```
tar -xvf drtree.tar.gz; cd DRTree; make demo
```

This creates the `bin/DRmain` program. Run it on some small datasets and have some fun! It has been tested on the Unix/linux platform on the andrew machines - it most probably runs under mac-osx, and Cygwin on Windows. Running

```
make hw1
```

should load the appropriate dataset, and print ' *Algorithm not implemented*' message. Currently the R-tree package supports 's' for range search, 'i' for insertion etc.

Implementation Details: You are required to implement the so-called *Ring-Search*. Figure 1 shows a 2-d scenario: given an outer rectangle and an inner rectangle, return all the rectangles that intersect (or touch, even at a single, corner point) the shaded area between these two rectangles. Thus, for Figure 1, you code should return all rectangles, except F.

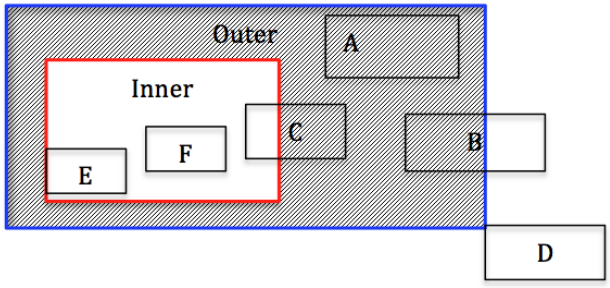

Figure 1: Example of "ring-search": It should return all the data rectangles that intersect, or touch, the shaded area, i.e., all rectangles, except F.

Implement the command 'g' for 'rin(g)-search' using the input dataset given in `hw1.input` (which is called by default with `make hw1`). Your code should work for *any* dimensionality

[1] FYI, the 'D' stands for 'deferred split' R-tree - but you don't need to worry about that.

d ($d = 1, 2, 3, \ldots$).

Hint:

- Do explore the range query functions in the code package. Modify or follow the code to implement your own functionality.
- For the final results, run `make spotless` before you run `make hw1` to clear the index file.

Input Format: Next, we give 2-d examples, but as we said, your code should work for *any* dimensionality, like the rest of the R-Tree code. All the input parameters are in the same line

```
g outer-x-low outer-x-high outer-y-low outer-y-high
  inner-x-low inner-x-high inner-y-low inner-y-high
```

For example, one valid input could be

```
g    100 600 100 600    200 300 300 400
```

Output Format: Your program should print

- the total number N of qualifying MBRs on the first line,
- and then, N lines, one for each qualifying data rectangle (record number, and their coordinates (x-low, x-high, y-low, y-high)) in tab-separated (`tsv`) format.
- We will also accept the total number N of qualifying MBRs, to be printed in the last line. (Because this will make your coding easier)

What to turn in:

- **Code:** [**30 pts**] Create a `tar` file named `hw1.q1.tar.gz` with your code and results under directory /Q1. We will grade it using the commands:

  ```
  tar -xvf hw1.q1.tar.gz; cd DRTree; make hw1;
  ```

  When typing in '`make hw1`' , we should see your answers.

- **Answers:** [**10 pts**] On hard copy, submit
  1. the function(s) you wrote (NOT the whole package), and
  2. the (`tsv`) output of running '`make hw1`'.

Q2 – Hilbert and Z-ordering [30 pts]

Print answers on separate page, with '[course-id] [hw#] [question#] [andrew-id] [your name]'

Problem Description: To goal is to become familiar with Hilbert and z-ordering algorithm. In the following tasks, we assume the ordering of both Hilbert and z-curve follows what you have seen in the lecture.

1. [**10 pt**] Write a program to compute the `znext` coordinates. `znext` means the next point on the z-curve given the coordinates. The command-line syntax should be: znext -n <order-of-curve> -d <dimension-of-curve> <x1> <x2> ... <xd>. Thus:
   - `znext -n 2 -d 2 0 0` *# should return 0 1 - going vertically!*
   - `znext -n 2 -d 3 0 0 0` *# should return 0 0 1 - going vertically, again*
2. [**10 pt**] Write a program to compute the `hnext` coordinates. `hnext` means the next point on the Hilbert-curve given the coordinates. To simplify the problem, we assume it is a 2-d Hilbert curve. The format should be:
   - `hnext -n 2 0 0` *# should return 1 0*
   - `hnext -n 2 0 1` *# should return 0 2*
   - `hnext -n 3 0 0` *# should return 0 1*
3. [**5 pt**] Give the results of your program on the input files [`http://www.cs.cmu.edu/~christos/courses/826.F14/HOMEWORKS/HW1/Q2/input.tar.gz`] Make sure you echo the input, so that it is clear which answer refers to which input.
4. [**5 pt**] Using your programs, write code to plot a z-curve and a Hilbert curve of order 6 (64 * 64 grid) and dimension 2. For plotting, we recommend `gnuplot`, but anything else that runs on the linux/andrew machines, is fine.

Hint:

- For Hilbert curve, we recommend the code/algorithm from the following papers (click below, to get their pdf)
  - Jagadish [SIGMOD 90]
  - Roseman+ [PODS 89]
- Make your code robust and guard against all the corner cases. E.g. negative coordinates, non-integer input, coordinates out of range, etc.

What to turn in:

- **Code:** Put your code for Q2.1, Q2.2, and Q2.4, along with a `makefile`, in a `tar` file `hw1.q2.tar.gz` under directory /Q2. Typing `make` should print your responses to the input files, and generate the plots for the z-/Hilbert-curves.
- **Answers:** Hard copy of your code for Q2.1 and Q2.2, your output for Q2.3, and your plots for Q2.4.

Q3 – Fractal Detectives [15 pts]

Print answers on separate page, with '[course-id] [hw#] [question#] [andrew-id] [your name]'

Problem Description: In this problem we will see how the fractal dimension can help us guess some properties of a cloud of points.
Suppose that a physicist colleague of yous (say, 'Mike'), has been doing experiments, measuring M=6 attributes, like pressure, temperature, etc., in N different settings. Thus, he has a file with N rows and M numbers per row, and he suspects that there are correlations among the M variables, obeying a yet-to-be-discovered physics law. *The goal is to help 'Mike' as much as you can.*
Example: if the measurements were about the gravitational force, which obeys Newton's law

$$F = \frac{C * m_1 * m_2}{r^2}$$

then, you would have M=4 attributes: force F, 2 masses m_1, m_2, distance r; and 3 degrees of freedom: m_1, m_2, r. Thus, the intrinsic (fractal) dimension of such a dataset, should be close to 3.

Implementation Details: Download 5 datasets `http://www.contrib.andrew.cmu.edu/~yanzhan2/5-mystery-data.tar.gz` and answer the following questions for each of the 5 datasets.

1. **[1 pt]** Plot the correlation integral for the dataset. We recommend the FDNQ package `http://www.cs.cmu.edu/~christos/SRC/fdnq_h.zip`.
2. **[1 pt]**
   (a) Are there any correlations among the M variables (yes/no)?
   (b) Are there clusters in Mike's cloud of points (yes/no)?
   (c) What can you say about the intrinsic dimensionality of the dataset? (give a number, or say '*undefined*', if there are clusters).
3. **[1 pt]** Based on the correlation integral alone, which of the choices are plausible, among (A)-(I), below? Report *all* that apply.
4. **[0 pt]** (Extra question - no points - just the admiration of the teaching staff :)) What else can you tell Mike about his dataset, to help him discover a new physics law? Eg., can you say that, in some appropriate projection, his cloud of points looks like a point? or line? or sinusoid? or spiral? Even harder question: Can you guess the equations we used, to generate Mike's 'mystery' dataset?

The choices of possible shapes of each 'mystery' dataset, are:
(A) **1-d: CIRCLE**: periphery of a circle, embedded in M-d space
(B) **2-d: DISK**: a 2d disk, embedded in M-d space

(C) **SIERPINSKI**: a Sierpinski triangle, embedded in M-d space
(D) **3-d: CUBE**: a cube, embedded in M-d space
(E) **3-d: FOOTBALL**: the cloud is 3-d ellipsoid (= elongated sphere, like an American football), embedded in M-d space.
(F) **3-d: PYRAMID**: the points form a 3-d pyramid (embedded in M-d space)
(G) **UNIFORM**: a cloud of points, uniformly distributed in M-d space
(H) **CLUSTERS**: the data points are clustered in C clusters
(I) **NONE**: none of the above

What to turn in:

- **Code:** No code to turn in.
- **Answers:** On hard copy, submit your plots for Q3.1, and your *typed* answers for Q3.2-Q3.4, for all the 5 'mystery' datasets.

Q4 – Correlation Integral [15 pts]

Print answers on separate page, with '[course-id] [hw#] [question#] [andrew-id] [your name]'

Problem Description: The goal is to gain a stronger intuition about the fractal dimension and the correlation integral. We want to generate a synthetic dataset, whose correlation integral will match the one of Figure 2. Let N be the number of points in this yet-to-be-created dataset. Notice the break-points at radius $r_1 = 10^{-6}$, $r_2 = 10^{-5}$, $r_3 = 10^{-3}$ and $r_4 = 10^{-1}$.

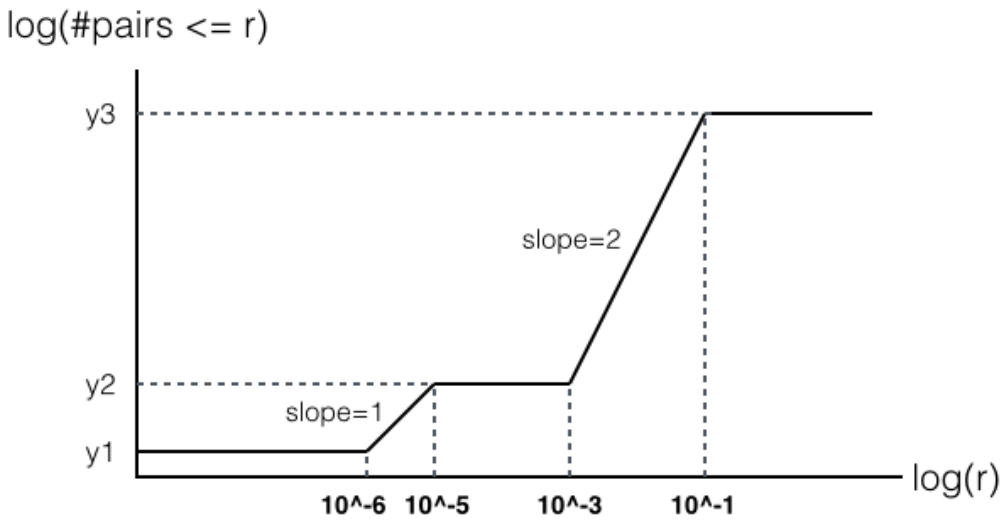

Figure 2: target correlation integral

1. **[1 pt]** Given that the count of points is N, what is be the value of $y1$? We are looking for an answer like N^3, or $\sqrt{N}$, or $N/10^5$, etc
2. **[1 pt]** What would be the value of $y3$ (again, as a function of the count of data points N)?
3. **[3 pt]** What is the only value of N, that could generate the correlation integral of Figure 2? *Hint:* : estimate the difference $y3 - y1$ as a function of N; derive an alternative estimation, as a plain number, using the information from the slopes and from $r_1, \ldots, r_4$.
4. **[3 pt]** As we learned in class, a plateau is an indication of clusters. What is your estimate for the count of clusters C in the dataset of Figure 2? We expect a number like, e.g., 2^{10}.

5. **[4 pt]** Write code, to generate your own 2-dimensional dataset that would have the correlation integral of Figure 2. (There are many correct answers - any one of them is fine).
6. **[3 pt]** Draw and print the correlation integral of your dataset. Please also plot *grid-lines*, and show coordinates in *log-base-10* to make grading easier[2]. Again, we recommend the FDNQ package[`http://www.cs.cmu.edu/~christos/SRC/fdnq_h.zip`]. *Hint:* It is OK if your plot has rounded, instead of sharp corners, at the break points.

What to turn in:

- **Code:** Submit your code to generate the dataset - we recommend `python`, but any language is acceptable, as long as it runs on andrew/linux, and it has a `makefile`, so that
  - `make dataset` would generate your dataset and
  - `make` would generate the drawing `integral.pdf` of its correlation integral. We recommend `pdf`, but jpg, png, etc are all fine.
- **Answers:** a hard copy with
  1. your answers to Q4.1-2-3-4,
  2. a copy of your code for Q4.5, and
  3. the plot for Q4.6

[2] in gnuplot: `set grid; set logscale xy 10; set format x "10^{%L}"`