Project Description

Due: ~~April 16, 2023 (tentatively)~~

April 23, 2023 (finalized)

Design a graphical user interface (GUI) that implements the following tasks, using a software tool that you feel comfortable with, such as Java or Matlab.

1. The GUI should have the capability to import txt input data set. An **input data set** is assumed to have the format of (id, [time], x, y, measurement), which records a set of measured values at location (x, y) and a time instance.

    - The domains of attributes *id*, *x*, *y* and *measurement* are scalar values: with *id* as integers, and *x*, *y* & *measurement* as floating-point numbers.
    - The *time* attribute is of an abstract data type. It can be of variety of scales: hour, day, month, quarter or year. Assume there are three possible time domains: (year), (year, month), and (year, month, day), where year, month, and day are all integers with month$\in$[1, 12] and day$\in$[1, 31]. At the beginning of the importing process, the user should be prompted to ask whether the time domain is day, month, or year.

    A sample data set of $PM_{2.5}$ measurements, pm25_2009_measured.txt, is available to download[1] for testing. It contains 146,125 spatiotemporal $PM_{2.5}$ measurements, with the time domain as *day*.

    Note: **You should design and implement a general solution to all the tasks in this project.** The program should not be implemented in a way that it only works for the pm25_2009_measured.txt data set. The program should be able to accept any input data set as long as the input data set fits the required format of (id, [time], x, y, measurement) using one of the three time domains.

2. [Optional] Visualize all the measurement locations in a U.S. state-level map. You can use a symbol such as a circle or a star to illustrate a location in the map, similar as the visualization result given in Figure 2 from *Paper to Review 2* published at *http://www.mdpi.com/1660-4601/13/8/749*. Use the input data set of pm25_2009_measured.txt to test your visualization. Since measurements are from 365 days, the set of measurement locations to visualize is the union of all the locations from 365 days. The boundary

---

[1] Particle pollution (also known as "particulate matter") in the air includes a mixture of solids and liquid droplets. Some particles are emitted directly; others are formed in the atmosphere when other pollutants react. Particles come in a wide range of sizes. (http://www.airnow.gov/index.cfm?action=aqibasics.particle).

EPA is concerned about particles that are 10 micrometers in diameter or smaller because those are the particles that generally pass through the throat and nose and enter the lungs. Ten micrometers is smaller than the width of a single human hair. Once inhaled, these particles can affect the heart and lungs and cause serious health effects. EPA groups particle pollution into two categories (http://www.epa.gov/air/particlepollution):

- Inhalable coarse particles (PM_{10}), such as those found near roadways and dusty industries, are larger than 2.5 micrometers and smaller than 10 micrometers in diameter.
- Fine particles ($PM_{2.5}$), such as those found in smoke and haze, are 2.5 micrometers in diameter and smaller. These particles can be directly emitted from sources such as forest fires, or they can form when gases emitted from power plants, industries and automobiles react in the air.

This project's test application data are daily $PM_{2.5}$ collected in 2009 in the conterminous U.S.

information of each state is stored in files st99_d00.dat and st99_d00a.dat, which are available to download.

3. Implement the **shape function** or **IDW** based spatiotemporal **reduction** or **extension** interpolation methods. Read *Paper to Review 1.pdf* for the detailed description of the methods:
    - Section 3.1.1 for the **shape function-based reduction** method
    - Section 3.1.2 for the **shape function-based extension** method
    - Section 4.2 for the **IDW-based reduction** method.
    - Section 4.2 for the **IDW-based extension** method.

    If you choose to implement IDW, the user should be prompted to input the values for the number of the nearest neighbors (integer type) and the exponent (floating-point type).

    At the beginning of the interpolation process, the user should be asked to do the following:

    - Import the data set with the locations to be interpolated. The data set is assumed to have the format of (id, x, y) with *id* as integers and *x* & *y* as floating-point numbers. These are usually the locations that do not have measurements. There are two choices for the location data set. You can try either one of them. But bonus points will be given if you use the location data set (2).
        (1) The **location data set**, county_xy.txt, is available to download for testing. It contains 3,109 locations. They are the centroids of 3,109 counties in the contiguous United States.
        (2) The **location data set**, blkgrp_xy.txt, is available to download for testing. It contains 207,630 locations. They are the centroids of 207,630 census blocks in the contiguous United States.
    - Specify the path and text file name for an **output data set** to store the interpolation results. One of the following interpolation result at each location (from the location data set) and time instance (with scale consistent with the input data set from Task 1) should be computed.
        (1) For the location data set of county_xy.txt, there should be 3,109 × 365 = 1,134,785 interpolation results in the output data set. This is because the time scale from the input data set pm25_2009_measured.txt is *day*, so each location should have 365 days to interpolate. Please name your output data set as county_id_t_w.txt for testing. The county_id_t_w.txt file should have the format of (county_id, year, month, day, pm25).
        (2) For the location data set of blkgrp_xy.txt, there should be 207,630 × 365 = 75,784,950 interpolation results in the output data set. This is because the time scale from the input data set pm25_2009_measured.txt is *day*, so each location should have 365 days to interpolate. Please name your output data set as blkgrp_id_t_w.txt for testing. The blkgrp_id_t_w.txt file should have the format of (block_id, year, month, day, pm25).

    [Open research question to address] When you implement spatiotemporal interpolation, for the time attribute, a "convenient" way is to use an integer to represent a time instance. For example, use integer 1 to represent 01/01/2009, 2 for 01/02/2009, … , and 365 for 12/31/2009, as shown in the table below. You can use this time encoding scheme to implement your chosen interpolation method for the $PM_{2.5}$ data.

A convenient encoding of time

The encoding of time	time
1	01/01/2009
2	01/02/2009
3	01/03/2009
…	…

365	12/31/2009

Using the above encoding scheme is convenient. But is it the best choice for the give $PM_{2.5}$ data set that gives the best interpolation results? For example, what if we use the double number 0.1 to represent 01/01/2009, 0.2 for 01/02/2009, …, and 36.5 for 12/31/2009, as shown in the table below?

An alternative encoding of time

The encoding of time	time
0.1	01/01/2009
0.2	01/02/2009
0.3	01/03/2009
…	…
36.5	12/31/2009

The general question would be whether the scale of time matters for spatiotemporal interpolation using the extension method? Some preliminary research shows that the choice of time scale does matter (see Section III.B of *Paper to Review 4.pdf*). This is a research topic in the field of temporal GIS that has been rarely studied. You are welcome to try to address this research challenge for this project. **Explain which time scale you choose for this project and why.** Extra credits will be given if you try to find a good choice of time scale. If you are interested in doing further related research after you complete the course, please let me know and we can work on this together, with the potential that leads to journal/conference publications.

4. [Optional] Visualize the daily $PM_{2.5}$ values in the extent of the contiguous U.S. in the year of 2009, using animation. You can assume the interpolated $PM_{2.5}$ value at the centroid of each county is the value for the whole county, or the interpolated $PM_{2.5}$ value at the centroid of each census block is the value for the whole census block. Design and implement a color rendering scheme to render different $PM_{2.5}$ values by different colors, similar as the visualization results given in Figure 5 from *Paper to Review 2* published at *http://www.mdpi.com/1660-4601/13/8/749*. Please note that Figure 5 is a seasonal visualization. As a bonus opportunity for this project, you need to do a simple daily visualization and add an animation feature so that each frame in your video is a snapshot of a daily visualization result.

5. Evaluate the interpolation methods using *cross validation*. You have two choices of cross validation:

   (1) You may use leave-one-out cross validation (LOOCV) that removes one of the n observation points and uses the remaining n − 1 points to estimate its value; and this process is repeated at each observation point. Read *Paper to Review 5.pdf* and Section 2.5.2 of *Paper to Review 3* published at *http://www.mdpi.com/1660-4601/11/9/9101* for the description of the *leave-one-out-cross validation*.

   (2) Alternatively, you may use *10-fold cross validation* that randomly splits the measurement data set into ten nearly equally sized folds. Read *Paper to Review 5.pdf* and Read *Paper to Review 5.pdf* and Section 2.3.1 of *Paper to Review 2* published at *http://www.mdpi.com/1660-4601/13/8/749* for the description of the 10-fold cross validation. Download 10FoldCrossValidation.zip. After unzipping the file, there are 10 fold directories with each directory having 4 files:

      a. st_sample.txt
      b. st_test.txt
      c. value_sample.txt
      d. value_test.txt

For each fold, you should use the sample data set to interpolate the points in the test data set. The original $PM_{2.5}$ measurements of the sample data are saved in value_sample.txt; the original $PM_{2.5}$ measurements of the test data are saved in value_test.txt. After the interpolation, each spatiotemporal point in a test data set will also have an interpolated $PM_{2.5}$ measurement.

Note: The time scale used in st_sample.txt and st_check.txt is 1 for 01/01/2009, 2 for 01/02/2009, … , and 365 for 12/31/2009, as shown in Table "A convenient encoding of time". If you implemented a different time encoding, please adjust the last time column in these files using your own time scale.

If you implemented IDW in Task 3, please evaluate the following IDW methods:

- IDW with 3 neighbors and exponents 1, 1.5, 2, 2.5, 3, 3.5, 4, 4.5 and 5.
- IDW with 4 neighbors and exponents 1, 1.5, 2, 2.5, 3, 3.5, 4, 4.5 and 5.
- IDW with 5 neighbors and exponents 1, 1.5, 2, 2.5, 3, 3.5, 4, 4.5 and 5.
- IDW with 6 neighbors and exponents 1, 1.5, 2, 2.5, 3, 3.5, 4, 4.5 and 5.
- IDW with 7 neighbors and exponents 1, 1.5, 2, 2.5, 3, 3.5, 4, 4.5 and 5.

For each of the above IDW methods:

(1) If you used LOOCV, save the interpolation results in a new file. Use $PM_{2.5}$ data for testing. The new text file should have the format of (original, n3e1, n3e1.5, n3e2, …, n7e4, n7e4.5, n7e5). The original column is the last column from pm25_2009_measured.txt. The rest of the columns store interpolated values for all the 146,125 spatiotemporal points from the input data set of pm25_2009_measured.txt. Since there are 5 possible numbers of neighbors and 9 possible exponent values, there should be 5×9=45 columns after the original column. Use the following naming convention for the file:

loocv_idw.txt

(2) If you used 10-fold cross validation, you will need to generate 10 new similar files:

10foldcv_idw_fold1.txt
10foldcv_idw_fold2.txt
…….
10foldcv_idw_fold10.txt

If you implemented shape functions (SF) in Task 3:

(1) If you used LOOCV, there should be just two columns in the new file with 146,125 rows using the format of (original, SF). Use the following naming convention for the file:

loocv_sf.txt

(2) If you used 10-fold cross validation, you will need to generate 10 new similar files:

10foldcv_sf_fold1.txt
10foldcv_sf_fold2.txt
…….
10foldcv_sf_fold10.txt

6. For the shape function-based method, or each of the above 45 IDW-based methods, if you used LOOCV, compute the error measurements MAE, MSE, RMSE, and MARE, based on the original and interpolated columns in the text files resulted from Task 5. Save the error measurement results in a new text file using the $PM_{2.5}$ data. Read Section 2.3.2 of of *Paper to Review 2* published at *http://www.mdpi.com/1660-4601/13/8/749* for the details of the error measurements MAE, MSE, RMSE, and MARE. If you used 10 fold cross validation, please calculate the average results of $\overline{MAE}$, $\overline{MSE}$, $\overline{RMSE}$, and $\overline{MARE}$.

   If implementing IDW, name your error statistics files as
   error_statistics_idw.txt

   This file should contain the error statistics (or average error statistics) of MAE, MSE, RMSE, and MARE for each of the 45 IDW methods.

   If implementing shape functions, name your error statistics file as

   error_statistics_sf.txt

7. Design and implement at least 1 spatiotemporal query based on the interpolation results of the $PM_{2.5}$ data.

8. Feel free to add any additional features to your graphic user interface.

9. Summarize all your results in a document named as Final_Project_lastName1_lastName2_lastName3_lastName4.doc, or Final_Project_ lastName1_lastName2_lastName3_lastName4.tex, if you would like to use LaTex. Generate a final PDF file Final_Project_lastName1_lastName2_lastName3_lastName4.pdf. Use the following structure:

   Paper title
   Authors (all the team members)
   Abstract
   Introduction
   Main text (method, data, visualization, error analysis results, queries, and discussion such as on time scale, etc.)
   Conclusions
   References

   A sample word document "sample report.doc" has been provided and attached. If you would like to have a template of a LaTeX file, please email me. Please explicitly describe in the beginning of your project report which one of the four interpolation methods you implemented in Task 3.

   Queries should be described in a SQL ~~similar~~ style ~~as the examples given in Section 6 of *Paper to Review 6.pdf*~~. If you implemented Tasks 2 and 4, include screenshots of the visualizations in the main text of the document. If you implemented IDW, you should discuss which one of the 45 IDW methods seems to be the best according to the LOOCV error analysis results. If you investigated the choice of time scale, you should discuss which time scale is your best choice. Feel free to refer to *Paper to Review 2* published at *http://www.mdpi.com/1660-4601/13/8/749* for ideas on organizing the document.

10. Submit the following:

    - Programming source codes,

- the output data set *county_id_t_w.txt* or *blkgrp_id_t_w.txt* from Task 3,
- cross validation text files from Task 5,
- error statistics text files from Task 6,
- *Final_Project_lastName1_lastName2.pdf, Final_Project_lastName1_lastName2.doc* or *Final_Project_lastName1_lastName2.tex,* as well as any other files such as figures and screenshots.

Grading

The final project is worth 100 points:

- 10 points for *county_id_t_w.txt* or *blkgrp_id_t_w.txt*,
- 10 points for cross validation text files,
- 10 points for error statistics text files,
- 70 points for your final report, as well as any supporting files such as figure files.

How to hand in the project

When you have finished your assignment, you will upload (submit) it to the same assignment area where you found it. You may only submit this assignment one time. I remind you that if you do not submit the assignment before the due date, you will not be able to submit it.

1. Complete the project.
2. Return to the assignment area for this assignment.
3. Look for the "Add Attachments" button in the assignment area.
4. Browse on your computer for the all the documents that you need to submit, and select them so that them upload to the assignment area.
5. Click "Submit".
6. You may or may not see a message that you have successfully submitted your document. If uncertain, click out of the assignment area, and then return to the assignment area. You should see information about your submission.