Assignment 4 - CS661A
Sampling and Reconstruction

Due date: April 17, 2023, 11:59pm

Grade: 100 points (10% of the course grade)

In this assignment, you will write a program to perform simple random sampling on a volume data and then reconstruct the volume data back from the sampled points. You will measure the reconstruction quality for two different reconstruction method.

Here are the tasks:

- Implement a simple random sampling (SRS) function. Your function should take a sampling percentage as an input parameter and return a set of sampled points. You will select points randomly following the simple random sampling (SRS) method. You should explicitly select the eight corner points of the data set in your sampled output. You should also preserve the data values for each sampled point so that later you can reconstruct the entire volume data. You can preserve the data values by attaching an array with your point locations containing data values for each selected point. Output the sampled points as a VTKPolyData (*.vtp) file and write it to disk. You can load the *.vtp file in ParaView and use point Gaussian representation to visualize the sampled points.
- Next, you are going to reconstruct the volume data from the sampled points using 'scipy.interpolate.griddata' method. The griddata() method provides a few ways to reconstruct data values in missing locations and you will use nearest neighbor, and linear interpolation methods.
    - First reconstruct the volume using 'nearest' as your interpolation method. Store the reconstructed volume data as a *.vti file into disk.
    - Next, reconstruct the volume data using 'linear' as your interpolation method. In this case, the method may result in some locations with 'nan' values as we may have to extrapolate some locations, and this method does not work for such cases. So, you will first perform reconstruction with the 'linear' interpolation method, and then the locations where you find 'nan' values, you will replace those locations with their nearest neighbor value. You can use 'np.isnan' method to check if a value is 'nan'. So, after doing this, you will produce a reconstructed data set and then store it as a *.vti file on disk. Your reconstructed volume files should be readable by ParaView.
- Finally, you will evaluate the reconstruction quality of your data set by comparing it with the original raw volume data. You will use Signal-to-Noise (SNR) ratio as a metric for quality comparison. You will evaluate the quality for reconstruction using the 'nearest' and 'linear' methods and further measure the reconstruction time. You will perform this study for the following sampling percentages: {1%, 3%, and 5%}. You will produce a table where you should present comparison results of quality and reconstruction time for both 'nearest' and 'linear' methods for the three different sampling percentages mentioned above. You will add this table in a pdf file and submit it along with your sampling code.
- You should submit a Python script that performs sampling for a single sampling percentage. Your Python script should take two input parameters: (1) sampling percentage, (2) reconstruction method: {nearest, linear}. Your script should output a *.vti file, which is the reconstructed data and a *.vtp file, which contains the sampled points as a VTKPolyData. Your output files should be readable by ParaView.

Use the following function to compute SNR. The first parameter is a numpy array containing ground truth data values and the second parameter is a numpy array that contains reconstructed data values in same order.

```
## compute SNR
def compute_SNR(arrgt,arr_recon):
    diff = arrgt - arr_recon
    sqd_max_diff = (np.max(arrgt)-np.min(arrgt))**2
    snr = 10*np.log10(sqd_max_diff/np.mean(diff**2))
    return snr
```

Dataset for this task:

The dataset that you will use in this assignment is a 3D scalar field volume. The VTI file is provided with the assignment.

How to submit?

The HelloIITK portal will be set up for submission. There will be a time limit set, and if you miss the deadline, you will start losing points as per the late submission policy. If you miss the deadline, please contact me, and I will deal with it on a case-by-case basis. Please start early and finish it by the deadline. **You should submit a README file where you should describe how to run your code. You should submit your code as a single Python script and a pdf file containing the table for the comparative study into a single compressed (*.zip) file.** Your code should be commented properly. Name your compressed zip file as "Lastname_rollnum_Assignment3.zip".

**** You must ensure you are submitting your correct submission files. If you upload wrong files, you will not be allowed to submit again. You have two late days. No submissions will be accepted after the allowed late days. ****