Final Assignment 22.1: Part 1: Parallel Computing with Pandas, NumPy, and DASK (120:00)

Start Assignment

Due Nov 13 by 11:29am **Points** 60 **Submitting** a file upload
Available after Nov 6 at 8am

- (https://classroom.emeritus.org/courses/9296/modules/items/1556599)
- (https://classroom.emeritus.org/courses/9296/modules/items/1556601)
- (https://classroom.emeritus.org/courses/9296/modules/items/1556602)
- (https://classroom.emeritus.org/courses/9296/modules/items/1556603)
- (https://classroom.emeritus.org/courses/9296/modules/items/1556604)
- (https://classroom.emeritus.org/courses/9296/modules/items/1556606)
- (https://classroom.emeritus.org/courses/9296/modules/items/1556607)
- (https://classroom.emeritus.org/courses/9296/modules/items/1556608)
- (https://classroom.emeritus.org/courses/9296/modules/items/1556609)
- (https://classroom.emeritus.org/courses/9296/modules/items/1556610)
- (https://classroom.emeritus.org/courses/9296/modules/items/1556611)
- (https://classroom.emeritus.org/courses/9296/modules/items/1556612)
- s://classroom.emeritus.org/courses/9296/modules/items/1556613)

(https://classroom.emeritus.org/courses/9296/modules/items/1556614)

(https://classroom.emeritus.org/courses/9296/modules/items/1556615)

(https://classroom.emeritus.org/courses/9296/modules/items/1556616)

(https://classroom.emeritus.org/courses/9296/modules/items/1556617)

(https://classroom.emeritus.org/courses/9296/modules/items/1556618)

(https://classroom.emeritus.org/courses/9296/modules/items/1556619)

(https://classroom.emeritus.org/courses/9296/modules/items/1556621)

(https://classroom.emeritus.org/courses/9296/modules/items/1556622)

(https://classroom.emeritus.org/courses/9296/modules/items/1556623)

(https://classroom.emeritus.org/courses/9296/modules/items/1556624)

(https://classroom.emeritus.org/courses/9296/modules/items/1556625)

(https://classroom.emeritus.org/courses/9296/modules/items/1556626)

(https://classroom.emeritus.org/courses/9296/modules/items/1556628)

(https://classroom.emeritus.org/courses/9296/modules/items/1556629)

(https://classroom.emeritus.org/courses/9296/modules/items/1556630)

(https://classroom.emeritus.org/courses/9296/modules/items/1556631)

(https://classroom.emeritus.org/courses/9296/modules/items/1556632)

(https://classroom.emeritus.org/courses/9296/modules/items/1556633)

(https://classroom.emeritus.org/courses/9296/modules/items/1556634)

(https://classroom.emeritus.org/courses/9296/modules/items/1556636)

Learning Outcome Addressed:

2. Run parallel operations in DASK.

The final assignment for this module is divided into two parts.

In Part 1 of the final assignment, you will compare the performance of the pandas, NumPy, and DASK *libraries* when performing calculations. In the first section, you will be working with NumPy and DASK

arrays to analyze which *library* is faster. In the next section, you will make the same comparison working with pandas and DASK *dataframes*.

Part 1 of the final assignment is worth 60 points.

To complete this assignment, follow these steps:

1. Download the **Assignment 22.1 (https://classroom.emeritus.org/courses/9296/files/2493747/download)** folder and open it using your local instance of Jupyter Notebook. There are 14 questions throughout the Jupyter Notebook for this activity. Some questions will require you to modify code, and others will require open-ended written responses.
2. Read the instructions and modify the code that is provided in the related cells for the following questions:
   a. Part 1: 1, 2, 4, 5, 6, 8, 9
   b. Part 2: 11, 13
3. Read the instructions and complete the open-ended questions for questions 3, 7, 10, 12, and 14. Below each cell that contains a question, you will see a Markdown cell in which you can answer that question. Responses should fully answer the question that is provided, and each response should be approximately two to three sentences.

Submission Instructions:

Your submission for this activity should be a Jupyter Notebook that includes your completed code and your open-ended responses:

Part 1: NumPy vs. DASK

1. Update the code cell by filling in the ellipsis to create a two-dimensional NumPy *array*, `arr`, with entries from 1 to 1,000 and dimensions 2,000 by 2,000.
2. Update the code cell by setting the value of the `chunks` *argument* to be equal to a *tuple* with elements equal to 250 and 250 to divide the NumPy *array* into smaller *chunks*, each with dimensions 250 by 250.
3. Describe your observations from the result that *prints* from running the code provided. State the size of each *chunk* and how many *chunks* the NumPy *array* is divided into.
4. Update the code cell by calling the `npartitions` *method* on the DASK *array* to *print* the number of partitions to the screen.
5. Update the code cell by setting the `axis` *argument* equal to 0 to sum over the rows.
6. Update the code cell by calling the correct DASK *function* to visualize how each row is summed.
7. Explain your observations of the graph produced by the code provided.
8. Update the code cell by calling the `numpy_arr_chk()` *function* and assigning the result to the `num_time` variable.
9. Update the code cell by calling the `dask_arr_chk()` *function* and assigning the result to the `dask_time` variable.

10. Describe which *library* performs better, NumPy or DASK, and explain your reasoning.

Part 2: Pandas vs. DASK

11. Update the code cell by completing the code to read the same dataset using DASK with the DASK `read_csv()` *function*.
12. Describe which *dataframe* takes longer, pandas or DASK, and explain your reasoning.
13. Update the code cell by setting the `npartition` *argument* inside of the `from_pandas` *function* equal to 2 and run the code cell to compare the `df_pandas_big` and `df_dask_big` *dataframes*.
14. Describe which *library* takes less time to run, pandas or DASK, and explain your reasoning.

Reference

Banerjee, Sourav. "Data Professionals Salary — 2022." *Kaggle*. 2022. https://www.kaggle.com/iamsouravbanerjee/analytics-industry-salaries-2022-india/version/9. (https://www.kaggle.com/iamsouravbanerjee/analytics-industry-salaries-2022-india/version/9)

Additional Details:

- **Estimated time:** 2 hours
- *This is a required assignment and counts toward course completion.*

Final Assignment 22.1

Criteria	Ratings		Pts
Part 1: 1. Update the code cell by filling in the ellipsis to create a two-dimensional NumPy array, arr, with entries from 1 to 1,000 and dimensions 2,000 by 2,000.	**3 pts** **Complete** The code has been updated correctly and runs without errors.	**0 pts** **Incomplete** The code has not been updated correctly and/or runs with errors.	3 pts
Part 1: 2. Update the code cell by setting the value of the chunks argument to be equal to a tuple with elements equal to 250 and 250 to divide the NumPy array into smaller chunks, each with dimensions 250 by 250.	**4 pts** **Complete** The code has been updated correctly and runs without errors.	**0 pts** **Incomplete** The code has not been updated correctly and/or runs with errors.	4 pts
Part 1: 3. Describe your observations from the result that prints from running the code provided. State the size of each chunk and how many chunks the NumPy array is divided into.	**5 pts** **Complete** The response is correct and includes all of the key details.	**0 pts** **Incomplete** The response is incorrect or lacks the required details for accuracy.	5 pts
Part 1: 4. Update the code cell by calling the npartitions method on the DASK array to print the number of partitions to the screen.	**4 pts** **Complete** The code has been updated correctly and runs without errors.	**0 pts** **Incomplete** The code has not been updated correctly and/or runs with errors.	4 pts
Part 1: 5. Update the code cell by setting the axis argument equal to 0 to sum over the rows.	**4 pts** **Complete** The code has been updated correctly and runs without errors.	**0 pts** **Incomplete** The code has not been updated correctly and/or runs with errors.	4 pts
Part 1: 6. Update the code cell by calling the correct DASK function to visualize how each row is summed.	**4 pts** **Complete**	**0 pts** **Incomplete**	4 pts

Criteria	Ratings		Pts
	The code has been updated correctly and runs without errors.	The code has not been updated correctly and/or runs with errors.	
Part 1: 7. Explain your observations of the graph produced by the code provided.	**5 pts** **Complete** The response is correct and includes all of the key details.	**0 pts** **Incomplete** The response is incorrect or lacks the required details for accuracy.	5 pts
Part 1: 8. Update the code cell by calling the numpy_arr_chk() function and assigning the result to the num_time variable.	**4 pts** **Complete** The code has been updated correctly and runs without errors.	**0 pts** **Incomplete** The code has not been updated correctly and/or runs with errors.	4 pts
Part 1: 9. Update the code cell by calling the dask_arr_chk() function and assigning the result to the dask_time variable.	**4 pts** **Complete** The code has been updated correctly and runs without errors.	**0 pts** **Incomplete** The code has not been updated correctly and/or runs with errors.	4 pts
Part 1: 10. Describe which library performs better, NumPy or DASK, and explain your reasoning.	**5 pts** **Complete** The response is correct and includes all of the key details.	**0 pts** **Incomplete** The response is incorrect or lacks the required details for accuracy.	5 pts
Part 2: 11. Update the code cell by completing the code to read the same dataset using DASK with the DASK read_csv() function.	**4 pts** **Complete** The code has been updated correctly and runs without errors.	**0 pts** **Incomplete** The code has not been updated correctly and/or runs with errors.	4 pts

Criteria	Ratings		Pts
Part 2: 12. Describe which dataframe takes longer, pandas or DASK, and explain your reasoning.	**5 pts** **Complete** The response is correct and includes all of the key details.	**0 pts** **Incomplete** The response is incorrect or lacks the required details for accuracy.	5 pts
Part 2: 13. Update the code cell by setting the npartition argument inside of the from_pandas function equal to 2 and run the code cell to compare the df_pandas_big and df_dask_big dataframes.	**4 pts** **Complete** The code has been updated correctly and runs without errors.	**0 pts** **Incomplete** The code has not been updated correctly and/or runs with errors.	4 pts
Part 2: 14. Describe which library takes less time to run, pandas or DASK, and explain your reasoning.	**5 pts** **Complete** The response is correct and includes all of the key details.	**0 pts** **Incomplete** The response is incorrect or lacks the required details for accuracy.	5 pts
			Total Points: 60

Top Questions

😔 😐 😔

It's all empty here!

If you have any questions ask one