ΕΘΝΙΚΟ ΜΕΤΣΟΒΙΟ ΠΟΛΥΤΕΧΝΕΙΟ

ΣΧΟΛΗ ΗΛΕΚΤΡΟΛΟΓΩΝ ΜΗΧΑΝΙΚΩΝ ΚΑΙ ΜΗΧΑΝΙΚΩΝ ΥΠΟΛΟΓΙΣΤΩΝ
ΤΟΜΕΑΣ ΤΕΧΝΟΛΟΓΙΑΣ ΠΛΗΡΟΦΟΡΙΚΗΣ ΚΑΙ ΥΠΟΛΟΓΙΣΤΩΝ
ΕΡΓΑΣΤΗΡΙΟ ΒΑΣΕΩΝ ΓΝΩΣΕΩΝ ΚΑΙ ΔΕΔΟΜΕΝΩΝ

Advanced Topics in Databases

2025-26, 9th Semester
Instructor: Dimitrios Tsoumakos
Lab Assistant: Nikolaos Chalvantzis

November 10, 2025

Semester Project

Description

This semester project requires analysis on (large) datasets, applying processing with techniques used in data science projects. The tools that will be used within the project framework are Apache Hadoop (version>=3.0) and Apache Spark (version>=3.5). You are asked to use the resources in the specially configured environment that has been granted to you on the AWS cloud. In summary, the purpose of the project is:

- Familiarization and development of students' skills in installing and managing the distributed systems Apache Spark and Apache Hadoop.
- Use of modern techniques through Spark APIs for large-scale data analysis.
- Understanding the capabilities and limitations of these tools in relation to available resources and chosen settings.

Data

This section will present the data you will be asked to use within the semester project framework. These are publicly available and free datasets collected from different sources.

For your convenience, all necessary datasets are accessible in the following AWS cloud S3 bucket: `s3://initial-notebook-data-bucket-dblab-905418150721/`.

Dataset	S3 URI
Los Angeles Crime Data (2010-2019)	s3://initial-notebook-data-bucket-dblab-905418150721/project_data/LA_Crime_Data/LA_Crime_Data_2010_2019.csv
Los Angeles Crime Data (2020-)	s3://initial-notebook-data-bucket-dblab-905418150721/project_data/LA_Crime_Data/LA_Crime_Data_2020_2025.csv
Census Blocks	s3://initial-notebook-data-bucket-dblab-905418150721/project_data/LA_Census_Blocks_2020.geojson
Census Blocks Fields	s3://initial-notebook-data-bucket-dblab-905418150721/project_data/LA_Census_Blocks_2020_fields.csv
Median Household Income by Zip Code	s3://initial-notebook-data-bucket-dblab-905418150721/project_data/LA_income_2021.csv
LA Police Stations	s3://initial-notebook-data-bucket-dblab-905418150721/project_data/LA_Police_Stations.csv
Race and Ethnicity Codes	s3://initial-notebook-data-bucket-dblab-905418150721/project_data/RE_codes.csv
MO Codes	s3://initial-notebook-data-bucket-dblab-905418150721/project_data/MO_codes.txt

Table 1: Datasets and their locations in S3 cloud.

Primary dataset: Los Angeles Crime Data[1] [2]

The primary dataset that will be used in the project comes from the public data repository of the United States government[3] . It includes crime recording data for the city of Los Angeles from 2010 to present, divided into two parts (2010-2019, 2020-). At the relevant links you can find data descriptions for each field (column).

Secondary datasets

In addition to the above data, a series of smaller-volume datasets will be used, which are also available in public repositories or sources:

Census Blocks (Los Angeles County)[4] : Dataset containing census data for Los Angeles County for the year 2020 in geojson format. Accompanied by a file with field descriptions (Census Blocks Fields).

Median Household Income by Zip Code (Los Angeles County)[5] : Dataset containing data regarding median household income for each ZIP Code in Los Angeles County. For convenience, the data has been collected and stored in csv format – as delimiter, the character ";". Refers to statistics from the year 2021.

LA Police Stations[6] : Small dataset containing data regarding the location of the 21 police stations located in the city of Los Angeles.

Race and Ethnicity codes: Small dataset containing the full descriptions corresponding to the racial profile encoding used in the primary dataset.

MO Codes[7] : Dataset with descriptions corresponding to the codes in the "Mocodes" column of **Los Angeles Crime Data**. Provided in txt format, where a code is at the beginning of each line and separated from the description by a space.

Queries

Query 1

Rank, in descending order, the age groups of victims in incidents involving any form of "aggravated assault". Consider the following age groups:

[1] https://data.lacity.org/Public-Safety/Crime-Data-from-2010-to-2019/63jg-8b9z

[2] https://data.lacity.org/Public-Safety/Crime-Data-from-2020-to-Present/2nrs-mtv8

[3] https://catalog.data.gov/dataset

[4] https://data.lacounty.gov/maps/lacounty::2020-census-blocks

[5] http://www.laalmanac.com/employment/em12c_2021.php

[6] https://geohub.lacity.org/datasets/lahub::lapd-police-stations/explore

[7] https://data.lacity.org/api/views/63jg-8b9z/files/e14442b9-a6b8-4531-83f3-f7ba980b1377

- Children: < 18
- Young adults: 18 – 24
- Adults: 25 – 64
- Elderly: >64

Query 2

Per year, find the 3 racial groups with the most victims of recorded crimes (Vict Descent) in Los Angeles. Results should be displayed in descending order of number of victims per racial group – also calculate and display the percentage of the total number of victims per case (see example result in Table 2).

year	Victim Descent	#	%
2024	White	413	32.5
2024	Black	274	25.4
2024	Unknown	132	22.3
2023	Hispanic/Latin/Mexican	512	30.2
⋮			

Table 2: Example result for Query 1

Query 3

Rank and display, in descending order of frequency, the crime commission methods and their corresponding codes (Mocodes). Use the **MO Codes** dataset to match codes with their descriptions.

Query 4

Calculate, per police station, the number of crimes that occurred closest to it than to any other, as well as its average distance from the locations where these specific incidents occurred. Results should be displayed sorted by number of incidents, in descending order (see example in Table 3).

division	average_distance	#
77TH STREET	2.208	7045
RAMPART	2.009	4595
FOOTHILL	3.597	3047
PACIFIC	2.739	2132

Table 3: Example result for Query 4.

Query 5

Using as reference the 2020 census data for population and the 2021 financial data for household income, calculate, for the two years 2020 and 2021, the correlation of average annual per capita income with the annual average crime rate per person for each area of Los Angeles. Repeat the calculation examining only the 10 areas with the highest and the 10 with the lowest annual per capita income.

Tips:

1. Crimes involving any form of ”aggravated assault” are considered all those incidents containing the term ”aggravated assault” in the relevant description.
2. For implementing queries involving geospatial analytics you must use the Apache Sedona library (version 1.6.1), which is installed in your working environment. As an example, you are given a usage guide in a relevant notebook that you can find in the corresponding section of your account. More information can be found in the documentation and website: https://sedona.apache.org/1.6.1/.
3. Consider that the various areas of Los Angeles are defined by the ”COMM” column of **Census Blocks**.
4. Some records in the primary dataset incorrectly refer to Null Island (0,0). These should be filtered and not taken into account in distance calculations.

Deliverables

1. Implement **Query 1** using DataFrame APIs (with and without UDF) and RDD API. Execute both implementations with 4 executors (1 core, 2GB memory). Is there a difference in performance among the three implementations? Comment on your findings. (20%)
2. Implement **Query 2** using DataFrame and SQL APIs. Execute both implementations with 4 executors (1 core, 2GB memory). Compare and comment on execution times between the two implementations. (20%)
3. Implement **Query 3** using DataFrame and RDD APIs. Execute both implementations with 4 executors (1 cores, 2GB memory). Compare and comment on execution times for each implementation. For the DataFrame API case, use the hint and explain methods to find which join strategies the catalyst optimizer uses. Experiment using different strategies (among BROADCAST, MERGE, SHUFFLE_HASH, SHUFFLE_REPLICATE_NL) and comment on the impact on performance. Which of Spark’s available join strategies is (are) most appropriate and why? (20%)
4. Implement **Query 4** using DataFrame or SQL API. For the joins executed, report which strategies Spark’s optimizer selects and comment on its choices. Execute your implementation applying scaling to the computational resources you will use: Specifically, you are asked to execute your implementation on 2 executors with the following configurations:
   - 1 core, 2 GB memory
   - 2 cores, 4GB memory
   - 4 cores, 8GB memory

   Comment on the results. (20%)
5. Implement **Query 5** using DataFrame or SQL API. For the joins executed, report which strategies Spark’s optimizer selects and comment on its choices. Execute your implementation using total resources of 8 cores and 16GB memory with the following configurations:
   - 2 executors × 4 cores, 8GB memory
   - 4 executors × 2 cores, 4GB memory
   - 8 executors × 1 core, 2GB memory

   Comment on the results. (20%)

Submission Terms

- The project should be completed in groups of at most 2 people.
- The submission deadline will be specified on helios in a link that will open soon. Late submissions with a delay of up to one (1) day will have a penalty of 50% of the grade. Beyond this delay, no submission will be graded. Submissions by means other than helios are not accepted.
- The project constitutes 30% of the total course grade. To receive a grade, each group must submit a report **and** successfully pass the mandatory oral examination on the project subject. The examination will take place after project submission (a relevant schedule will be posted on helios).
- As a deliverable, a .pdf file will be submitted with the name of the student IDs of group members separated by underscore (or the student ID in case of a single-member group), e.g., 03100000.zip, or 03100000_03100001.zip (depending on the number of people in the group). The file will contain a report (strictly with what is requested in the assignment) which will contain answers to the requirements as well as necessarily a link to a repository (github, gitlab, bitbucket, etc.) with the code you have implemented, as well as possible scripts/howtos for executing your code. All submissions are strictly subject to the academic ethics code of NTUA and the School of ECE.

  Your code must not change from the report submission date until course grading. If this happens, your grade will be ZERO (0).
- Each group can implement their code in Scala, Java, or Python. Additionally, you are given the option to use your own resources (e.g., personal computers, VMs in another cloud provider), as long as the project requirements are met. In any case, the examination will require live demonstration of your code.
- Questions/clarifications about the project will be done through the forum on the course page on helios. Do not send questions to instructors'/assistants' emails.