NATIONAL TECHNICAL UNIVERSITY OF ATHENS
SCHOOL OF ELECTRICAL AND COMPUTER ENGINEERING
DIVISION OF COMPUTER SCIENCE
KNOWLEDGE AND DATABASE SYSTEMS LABORATORY
http://www.dblab.ece.ntua.gr/

Advanced Topics in Database Systems

Academic Year 2023-24, 9th Semester
Instructor: Dimitrios Tsoumakos
Lab Assistant: Nikos Chalvantzis

3 November 2023

Term Project

Description

The current term project requires data analysis on (large) datasets, applying processing techniques used in data science projects. In the context of the project, you are asked to utilise Apache Hadoop (version>=3.0) and Apache Spark (version>=3.4). For the installation and configuration of your working environment, computing resources can be provisioned through the *~okeanos-knossos*[1] public cloud service. In short, the objectives of this project are:

- familiarization with and development of students' skills in the installation and management of distributed systems such as Apache Spark and Apache Hadoop.
- the use of modern techniques through the Spark APIs for the analysis large volumes of data.
- understanding the capabilities and limitations of these tools concerning the available resources and selected configurations.

[1] https://okeanos-knossos.grnet.gr/home/

Data-sets

Primary data-set: Los Angeles Crime Data

The main data-set used in the project is sourced from the public data repository of the United States government[2] . More specifically, it contains data describing recorded crimes in the city of Los Angeles from 2010 until this day. The data-set can be found in .csv file format at the following links:

- https://catalog.data.gov/dataset/crime-data-from-2010-to-2019
- https://catalog.data.gov/dataset/crime-data-from-2020-to-present

Alternatively, the same data-set is available through a repository of the municipality of Los Angeles, here:

- https://data.lacity.org/Public-Safety/Crime-Data-from-2010-to-2019/63jg-8b9z
- https://data.lacity.org/Public-Safety/Crime-Data-from-2020-to-Present/2nrs-mtv8

The latter links also include descriptions for each one of the 28 attributes of the data-set, *which can be useful in the context of the term project*, as well as some complimentary resources and data (section "**Attachments**").

Complimentary data-sets

Besides the primary data-set a few other, smaller sets will be used - also originating from public sources:

LA Police Stations: A small data-set that includes information about the location of the 21 police departments of the city of Los Angeles. This data-set can be found at a Los Angeles municipality repository and downloaded in .csv file format here:

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

Median Household Income by Zip Code (Los Angeles County): Another small data-set that contains data about the average income per household and Zip Code in the region of the Los Angeles County. The data have been collected through census and are offered by the Los Agneles Almanac webpage at the following links:

- http://www.laalmanac.com/employment/em12c_2015.php
- http://www.laalmanac.com/employment/em12c_2017.php
- http://www.laalmanac.com/employment/em12c_2019.php
- http://www.laalmanac.com/employment/em12c.php

To facilitate students, the data have been collected and are available in .csv file format at the link:

- http://www.dblab.ece.ntua.gr/files/classes/data.tar.gz

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

In the context of this project we will only use the data that refers to the 2015 census but students are encouraged to experiment with a larger subset.

Reverse Geocoding: The term "geocoding"(γεωκωδικοποίηση) αναφέρεται συνήθως στη μετάφραση μιας διεύθυνσης σε μια τοποθεσία σε σύστημα συντεταγμένων. usually refers to the translation of an address into a location in a coordinate system. The reverse process, i.e., mapping a pair of coordinates to an address, is known as "reverse geocoding". Within the scope of the project, there will be a need to map coordinates (latitude, longitude) to ZIP codes within the city of Los Angeles. This can be achieved programmatically using the assistance of web services known as geocoders and libraries such as geopy[3]. However, the process is slow due to the latency of web services. Therefore, a data-set with reverse geocoding information covering locations required for the queries of this project is provided to you. Of course, you are encouraged to experiment and, if you wish, to try implementing this specific functionality. The data-set is available in .csv file format here:

- http://www.dblab.ece.ntua.gr/files/classes/data.tar.gz

Queries

Query 1

Find, for **each** year, the top-3 months with highest number of recorded crimes committed. You are asked to print the month, year, number of criminal acts recorded, as well as the ranking of the month within the respective year. Results are expected to be sorted in ascending order with respect to the year and descending order with respect to the number of crimes (as in Table 1).

year	month	crime_total	#
2010	2	2145	1
2010	3	1492	2
2010	5	54	3
2011	12	4632	1
2011	6	2312	2
2011	4	312	3

Table 1: Example of expected results for Query 1

Query 2

Sort the different parts of the day taking into account crimes that were committed on the (STREET), in descending order. Consider the following pars of the day:

- Morning: 5.00am – 11.59am
- Afternoon: 12.00pm – 4.59pm
- Evening: 5.00pm – 8.59pm
- Night: 9.00pm – 3.59am

[3] https://geopy.readthedocs.io/en/stable/#module-geopy.geocoders

Query 3

Find the descent of the victims of recorded crimes in Los Angeles for the year 2015 in the 3 ZIP Code areas with the highest and the 3 ZIP Codes with the lowest income per household. Results are expected to be printed from highest to lowest number of victims per ethnic group (as in Table 2)

Victim Descent	#
White	413
Black	274
Unknown	132
Hispanic/Latin/Mexican	12

Table 2: Example of expected results for Query 3

Tips:

1. Victimless crimes exist: Filter out of your working data-set crimes for which no victims exist or their descent is not documented.
2. In cases where the **Reverse Geocoding** data-set contains more than one ZIP Code per coordinates tuple, you only need to use one of them (i.e., the first one).
3. Areas covered by the **Median Household Income by Zip Code** data-set refer to the extended region of Los Angeles County.
4. You can, optionally, use the translation of descent codes to full descriptions that is available in the information that accompanies the data-set, for a better presentation of your results.

Query 4

For the last query, you are asked to examine whether the police stations that respond to recorded crimes in Los Angeles are the closest ones to the crime scene. Towards that end, you are asked to implement and execute two pairs of similar queries and compare results:

- a) Calculate the number of crimes committed with the use of firearms of any kind and the average distance (in km) of the crime scene to the **police station that handled the case**. The results should appear ordered by year in ascending order. b) Additionally, calculate the same stats (number of crimes committed with the use of firearms of any kind and average distance) per police station. Results should appear ordered by number of incidents, in descending order (as in Table 3).
- a) Calculate the number of crimes committed with the use of firearms of any kind and the average distance (in km) of the crime scene to the **police station that is located closest to the crime scene**. The results should appear ordered by year in ascending order. b) Additionally, calculate the same stats (number of crimes committed with the use of firearms of any kind and average distance) per police station. Results should appear ordered by number of incidents, in descending order (as in Table 4).

year	average_distance	#
2010	2.352	7232
2011	2.312	6763
2012	2.276	8487
2013	2.392	9745

Table 3: Example of expected results for Query 4a)

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

Table 4: Example of expected results for Query 4b)

Tips:

1. Some records (mistakenly) refer to Null Island. They need to be filtered out and not taken into account in the calculation.

2. Incidents connected to the use of firearms of any kind correspond to codes of the form of “1xx” for column “Weapon Used Cd”.

3. Codes in column “AREA ” of **Los Angeles Crime Data** match those of column “PRECINCT” of **LA Police Stations** and refer to the police station that handled each recorded crime.

4. You are free to choose any implementation you prefer for the distance calculation between two points. For reference, you are given a Python implementation, using the geopy[4] library.

```python
import geopy.distance

# calculate the distance between two points [lat1, long1], [lat2, long2] in km
def get_distance(lat1, long1, lat2, long2):
    return geopy.distance.geodesic((lat1, long1), (lat2, long2)).km
```

Tasks

1. Install and configure correctly the infrastructure for storage and processing using Apache Spark and Hadoop. Spark should be executed on top of the Apache Hadoop resource manager, YARN. Format the Apache Hadoop Distributed File System and use it for input/output of all the processing tasks you will develop. The execution environment needs to be set up in a fully distributed configuration, with at least two working nodes. Finally, web interfaces for HDFS, YARN and Spark History Server need to be available and accessible. (10%)

2. Create a DataFrame that contains the main data-set. Keep the original column names but change the column types as instructed below:

[4] https://geopy.readthedocs.io/en/stable/

- Date Rptd: date
- DATE OCC: date
- Vict Age: integer
- LAT: double
- LON: double

Print the total number of rows for the entire data-set and the data type of every column. (5%)

3. Implement **Query 1** using both the DataFrame and SQL APIs. Execute both implementations with 4 Spark executors. Do you notice differences in the execution times? Justify your answer. (15%)
4. Implement **Query 2** using both the DataFrame/SQL and RDD APIs. Report and compare execution times for 4 Spark executors. (20%)
5. Implement **Query 3** using the DataFrame/SQL API. Report and compare execution times for 2, 3 and 4 Spark executors. (20%)
6. Implement **Query 4** using the DataFrame/SQL API. (20%)
7. For every join implemented in **Query 3** and **Query 4**, use the methods hint() and explain() of the DataFrame/SQL API to alter the physical execution plan (BROADCAST, MERGE, SHUFFLE_HASH, SHUFFLE_REPLICATE_NL). Notice the changes in the Spark History UI. Make a comment on (which ones) of the supported join strategies are a better fit and why. (10%)

Deliverables - Submission Conditions

- The project is available for individual students or groups of two.
  SUBMISSION DEADLINE: WEDNESDAY 10th JANUARY 2024, 23:59.
- The project deliverable will be submitted online through the helios webpage (the submission link will be available at a later time).
- The project contributes for 30% of the grading score you will receive for the course. For the project score to be taken into account, each participating individual or group need to submit a report and successfully pass an oral exam on the term project. The exam will take place after the submission of the reports. Exact dates will be announced in the future.
- The deliverable should be submitted as a pdf file. Its title should consist of the Student ID codes of the members of the group separated by an underscore, if necessary, e.g. 03100000.zip, ή 03100000_03100001.zip. The file should contain a report – addressing all questions presented in the **Tasks** section above, as well as a link to an accessible code repository (e.g. github, gitlab, bitbucket, etc.) where all source code of your implementations should be uploaded together with complimentary scripts or notes for the execution of the code. All submissions are strictly subject to the code of academic ethics of NTUA and the School of ECE. **The submitted code cannot be modified past the date and time of submission, otherwise your project will not be graded**.

- You can implement your solutions using Scala, Python or Java. Additionally, you can utilise your own resources (e.g. PCs, VMs) or sources provided by *~okeanos-knossos*. In any case, you will be asked to present a live demonstration of your implementation during the project examination.

- Questions/discussion on the project will be posted on the forum of the course's helios webpage. Please do not contact the tutors via mail with questions about the term project.