Ton Duc Thang University
Faculty of Information Technology

FINAL PROJECT

Course: Mining Massive Datasets

Duration: 05 weeks

I. Formation

- The midterm project is conducted in groups of **04 – 05** students.
- The student group fulfills the requirements and submits the work according to the detailed instructions below.

II. Requirements

Given datasets in the **datasets** folder, students conduct tasks below.

File	Description
mnist_small_train.csv	Images and labels in the hand-written dataset MNIST[1]. 7000 samples/lines. Each line consists of 785 integers • The first integer: label (0, 1, 2, 3, …, 9) • 784 remaining integers are pixel of a grayscale image 28 x 28.
mnist_small_test.csv	Similar to **mnist_small_train.csv** but there are 3000 samples/lines.
ratings2k.csv	Product rating dataset. The first line is header. • **index**: row index • **user**: user ID • **item**: product ID • **rating**: rating (0.0-5.0) 2365 remaining lines are data samples.

[1] https://en.wikipedia.org/wiki/MNIST_database

stockHVN2022.csv	Stock prices of HVN code in HOSE in 2022 (until Nov 18th) The first line is header. • **Ngay**: date • **HVN**: price 219 remaining lines are data samples.

a) Task 1 (2.0 points): Clustering

Students use the **mnist_small_test.csv** file for this task.

YC1_1: Illustration

Use **DataFrame** of **pyspark.sql** and the **matplotlib.pyplot** library to draw a chart 3 x 5 to display the first 15 images.

- Convert image vectors of 784 dimensionality to matrices 28 x 28
- Display the image using **imshow**() of matplotlib.pyplot
- For each subplot display **title** as the label of the image.

YC1_2: Clustering

Use the **pyspark.ml.clustering.KMeans** library to cluster image vectors with three values of k, including 5, 10, 15.

For each experiment (one value of k):

- Initialize the model
- Cluster data samples, given k, using Euclidean distance
- Save the model to files
- Load the model from files
- Compute the summation of Euclidean distance from each data point to its centroid.

YC1_3: Result Visualization

Use the matplotlib.pyplot library to draw a bar chart visualizing the summations of Euclidean distance from each data point to its centroid corresponding to three values of k.

b) Task 2 (2.0 points): Dimensionality Reduction

YC2_1: Dimensionality reduction in the training set

Use the **pyspark** library and the **SVD** algorithm to reduce the dimensionality of image vectors in **mnist_small_train.csv** to 196 (14 x 14). After that, save the result to 01 csv file named **mnist_small_train_svd.csv**.

YC2_2: Dimensionality reduction in the test set

Perform the similar task as **YC2_1** in **mnist_small_test.csv** and save the result to 01 csv file named **mnist_small_test_svd.csv**.

c) Task 3 (1.0 points): Recommendation with Collaborative Filtering

Read the **ratings2k.csv** file to a **DataFrame** (**pyspark.sql**) in which

- Each row is of a **user** (74 users)
- Each column is of **item** (467 items)
- Rows are sorted in ascending order of users.

Students create a recommendation model applying the **Collaborative Filtering** method and **Pearson Correlation Coefficient** to infer all ratings of the remaining users.

Students compute the **Mean Squared Error** between **predicted rating values** and **actual ones** for user IDs from 71 and item IDs from 401 (excluding **missing rating values in data samples**).

d) Task 4 (2.0 points): Stock Price Regression

Students use the **stockHVN2022.csv** file for this task.

The problem is given prices of HVN stock code of 05 previous dates, then predicting the price of the following date.

Students use data samples from Jan to the end of Jun as the training set, and the remaining part, from Jul, as the test set.

For each set, students create a **DataFrame** with two columns

- **Prices of 05 previouse dates:** a vector of prices of 05 previous dates
- **Today price**: the price of the following date.

For example, given a series of prices a, b, c, d, e, f, g, h, students generate price tuples like

- *a, b, c, d, e* → *f*
- *b, c, d, e, f* → *g*
- *c, d, e, f, g* → *h*

- *etc.*

Students save the two **DataFrames** with meaningful and human-readable filenames, then

- Build up a **Linear Regression (pyspark)** model to predict stock prices above, in which learning data in the training set and evaluating in the test set.
- Save the model to files
- Load the model from files
- Compute the **Mean Square Error** in the training and test sets for the pre-trained model.
- Use the **matplotlib.pyplot** library to draw a bar chart visualizing **Mean Square Errors** in the training and test sets.

e) Task 5 (2.0 points): Multi-class Classification

Use files including **mnist_small_training/test.csv** and **mnist_small_training/test_svd.csv** for this task.

In total, there are two data sets including the original (training, test) and (training_svd, test_svd) from task b).

Students build up models for multi-class classification using **pyspark** to recognize hand-written image vector,

- *Input: image vector*
- *Output: label*
- *Loss function: Cross Entropy*
- *Metric: Accuracy.*

Students study and apply the three common models in **pyspark**, including

- Multi-layer Perceptron https://spark.apache.org/docs/latest/ml-classification-regression.html#multilayer-perceptron-classifier
- Random Forest https://spark.apache.org/docs/latest/ml-classification-regression.html#random-forest-classifier
- Linear Support Vector Machine:

https://spark.apache.org/docs/latest/ml-classification-regression.html#linear-support-vector-machine

Students draw a quartet-bar chart using the matplotlib.pyplot library to visualize the accuracies of the three model in the four data sets of MNIST, including the training, test, training_svd, and test_svd sets.

4 data sets x 3 models

f) Task 6 (1.0 point): Report

- Student groups compose a presentation to report your work.
- **THERE IS NO PRESENTATION TEMPLATES. STUDENTS ARANGE CONTENTS IN A LOGICAL LAYOUT BY YOURSELVES.**
- The presentation must include below contents
  - Student list: Student ID, Full name, Email, Assigned tasks, Complete percentage.
  - Briefly present approaches to solve tasks, should make use of pseudo code/diagrams.
  - AVOID EMBEDDING RAW SOURCE CODE IN THE PRESENTATION.
  - Study topics are introduced briefly with practical examples.
  - Advantages versus disadvantages
  - A table of complete percentages for each task.
  - References are presented in IEEE format.
- **Format requirements:** slide ratio of 4x3, avoid using dark background/colorful shapes because of projector quality, students ensure contents are clear enough when printing the presentation in grayscale.
- Presentation duration is **05 minutes**.
- Instructions for recording the presentation are in the appendix.

III. Submission Instructions

- Create a folder whose name is as

  <Student ID 1>_< Student ID 2>_< Student ID 3>_< Student ID 4>

- Content:

- **source.ipynb** → source code (remain all cell outputs)
- **source.pdf** → pdf of the notebook
- **presentation.pdf** → presentation.
- **video.txt** → URL to the presentation recording

- Compress the folder to a zip file and submit by the deadline.

IV. Policy

- **Student groups submitting late get 0.0 points for each member.**
- **Wrong student IDs in the submission filename cause 0.0 points for the corresponding students.**
- **Missing required materials in the submission loses at least 50% points of the presentation.**
- **Copying source code on the internet/other students, sharing your work with other groups, etc. cause 0.0 points for all related groups.**
- **If there exist any signs of illegal copying or sharing of the assignment, then extra interviews are conducted to verify student groups' work.**

-- THE END --

HƯỚNG DẪN
VIDEO THUYẾT TRÌNH
INSTRUCTIONS
FOR RECORDING PRESENTATION VIDEO

I. Mục tiêu/*Objectives*

- Nhóm sinh viên thực hiện quay video thuyết trình để báo cáo nội dung đồ án/đề tài.
  Student groups record a video to present your project/topic.
- Hình thức, công cụ, thời lượng được mô tả chi tiết ở mục tiếp theo.
  Formation, tools, and duration are described in the next section.

II. Yêu cầu/*Requirements*

- Công cụ: Zoom/Google Meet
  Tools: Zoom/Google Meet
- Thời lượng: theo mô tả trong đồ án
  Duration: designated in the project
- Hình thức:
  Formation
  - Nhóm sinh viên tạo một meeting để tham gia thuyết trình.
    Student groups create a meeting to present your work.
  - Đặt tên hiển thị theo dạng <MSSV>_<Họ tên>,
    Set the display name as <Student ID>_<Full Name>
    ví dụ 52200001_Nguyễn Văn A
    for example, 52200001_Nguyen Van A
  - Tất cả thành viên phải bật camera trong toàn bộ buổi thuyết trình.
    Every member must turn on your camera during the presentation.
  - Sinh viên trình chiếu bài thuyết trình nhưng phải đảm bảo hiển thị đầy đủ khuôn mặt của các thành viên còn lại.

Students show your presentation but ensuring to display all member facial thumbnails.

- Các sinh viên thay phiên trình bày các nội dung.
  Students share the role of presenting.
- Bài thuyết trình được quay lại, đảm bảo chất lượng hình ảnh và âm thanh.
  The presentation is recorded completely.

III. Hướng dẫn nộp bài/Submission Instructions

- Video tải lên Youtube và đặt ở chế độ "unlisted", tuyệt đối không để dạng "public".
  The recording is uploaded to Youtube with the "unlisted" sharing option. Do not share with the "public" option.
- Đặt tên theo cú pháp
  Rename the recording as below

  <Năm học>-<Học kỳ>-<Môn học>-<Tên nhóm>

  <Year>-<Semester>-<Course>-<Group Name>

  trong đó gồm:
  - <Năm học> theo dạng YYYY, ví dụ 2223, 2324, 2425
    <Year> in form of YYYY, for example, 2223, 2324, 2425
  - <Học kỳ> là "HK1" hoặc "HK2"
    <Semester> is "HK1" (Term 1) or "HK2" (Term 2)
  - <Môn học> là "AI" (NM Trí tuệ Nhân tạo) hoặc "MMDS" (Xử lý Dữ liệu lớn)
    <Course> is "AI" (Introduction to AI) or "MMDS" (Mining Massive Datasets)
  - <Tên nhóm> theo tên đã đăng ký
    <Group Name> as in registration
- Một thành viên nhóm đại diện nộp đường dẫn theo deadline được cho.
  Only one representative student submits the video URL by the deadline.

-- HẾT --

-- END --