

**Western University**  
**Faculty of Engineering**  
**Electrical and Computer Engineering Department**  
**ECE 9309/9039 Machine Learning**  
**Final Exam**  
April 2, 2019

**Exam Instructions:**

- This exam is a *take-home* exam.
- The exam takes place from **Tuesday, April 2, 2019 at 5:00 pm to Friday, April 5, 2019 at 9:00 pm.**
- You are allowed to use any software like Matlab, Python, ...etc for solving the exam questions.
- You may consult with your classmates or friends for help on the exam, but you may **NOT** share or show your codes/solutions. Showing one another the solutions is **NOT** a consultation.
- No clarifications or corrections will be provided. If you believe that there is an error, inconsistency, or omission in the exam, please state your assumptions about the issue within your discussion of that issue.
- Any external source of code and ideas must be cited to give credit to the original source.
- Your final submission to this final exam should include:
  - All codes/scripts that are used for solving the exam questions.
  - Clear documentation that includes any explanations, comments or observations of the exam solutions along with the input commands and output (files/graphs) from these programs. The submitted document has to be in a pdf format.
- All files should be compressed into a zip file with the naming convention: `studentID.FirstName_LastName.zip` and submitted it on OWL in Final Exam Field under the Assignment section. An example of a submitted file name is: i.e., `250XXXXXX_John_Smith.zip`.

**HONOR STATEMENT:**

BY SUBMITTING THIS EXAM THROUGH OWL, I AFFIRM ON MY HONOR THAT I AM AWARE OF ALL EXAM INSTRUCTION, AND I HAVE NOT SHARED MY CODES/SOLUTIONS WITH OTHERS

## Dataset Description

Attached with the exam instructions, you will find the `datasets.zip` file. After unzipping the file, you will find several `.csv` files, where each file represents real-world measurement data of a heat experiment inside a steel furnace. Each file has a prefix number representing the experiment heat ID. File names in the given dataset have two formats, those end with `_ALARM_OUT.csv` which corresponds to experiments with no anomalies, and on the other hand, heat experiments containing anomalies have a suffix name `"_ALARM_OUT_tag.csv"`, where the anomaly tags are added in the last column of each file (1 = anomaly, 0 = normal). In the datasets, the features are the vibration measurements in columns A, B, ..., H which correspond to (X1, X2, ..., X8) measurement signals. Each feature represents a vibration signal inside the furnace at several frequency bands. Data should be considered only when it is in steady-state conditions. This information is in column I (`"Sds_Armed"`), where steady-state data is only when `"Sds_Armed=1"`. Column J represents the anomaly tags. Each example row is a measurement recorded at a time instance, which is considered a time-series data measurements.

## Data Preparation [20 points]

- **Question 1)** - Filter all *"Normal Experiments"* by taking into account only active examples `"SDS Armed = 1"`, and then, merge them in a new file named as `"merged_exp_normal.csv"`. Write a script that performs this task and indicate the number of examples of the merged dataset [10 points].
- **Question 2)** - Filter all *"Experiments with Anomalies"* by taking into account only active examples `"SDS Armed = 1"` similar to the requirements in Questions 1, and then, merge them in a new file named as `"merged_exp_contains_anomalies.csv"`. Write a script that performs this task and indicate the number of examples of the merged dataset [10 points].

## Building A Statistical-Based Anomaly Detection Algorithm [40 points]

- **Question 3)** - Since the `merged_exp_contains_anomalies.csv` contains anomalies, apply any significance test to rank the significance of each feature (X1, X2, ..., X8) as being a distinctive feature of anomalies [5 points].
- **Question 4)** - Model the normal process `"merged_exp_normal.csv"` using Gaussian distribution. Assume that the features are independent. Characterize your model using the following cases:
  - Consider all features (X1, X2, ..., X8) [5 points].
  - Mark the most important two features (obtained from the significance test in Question 3) [2 points].
  - The projection of the feature space into the first two components using Principle Component Analysis (PCA) (obtained from the significance test in Question 3) [5 points].
- **Question 5)** - Model the same normal process `"merged_exp_normal.csv"` using Gaussian distribution with all requirements in Question 4. However, assume that the features are dependent [10 points].  
*Hint: Think about the co-variance matrix!*

- **Question 6)** Develop an anomaly alarm by adjusting a threshold  $\epsilon$  to your Gaussian models obtained in Questions 3 and 4, and accordingly, generate an alarm accordingly. Use any experiment that contains anomaly as a test case [**8 points**].
- **Question 7)** Plot the generated alarm, true anomaly flags (given from the dataset), and the feature  $X_1$  [**5 points**].

## Alternative Ways For Anomaly Detection [40 points]

- **Question 8)** Apply one supervised learning approach for classifying the events to normal and anomalies [**10 points**].
- **Question 9)** Apply any clustering based algorithm you learn in the class, i.e., (hard and soft clustering with K-means, EM, ..., etc.) to decouple the anomaly data from the normal ones. Is there a direct mapping to the true anomaly tags? discuss your findings [**10 points**].
- **Question 10)** Compare the Gaussian-based anomaly detection algorithm, the supervised learning approach you picked, and the clustering approach in terms of [**20 points**]:
  - Detection capabilities (use the relevant metrics discussed in the class).
  - Time complexity and memory requirements during the training phase.
  - Time complexity and memory requirements during the execution phase.

## Bonus Questions [20 points]

- **Question 11)** Optimize the parameter  $\epsilon$  from **Question 6** with the objective of maximizing the detection rate and minimizing the false alarm rate. Compare the results before and after optimizing  $\epsilon$  [**15 points**]. Particularly, consider the following objectives “jointly”:
  - Reduce the number of generated false alarms.
  - Increase the number true anomalies discovered.
- **Question 12)** If the features in the Gaussian-based approach do not follow the Gaussian distribution, apply a suitable transformation to make better suit the Gaussian shape. Compare the results before and after the transformation [**5 points**].