Carnegie Mellon University
Department of Electrical and Computer Engineering

18-752 Estimation, Detection and Learning Spring 2023

Course Project Issued: March 28, 2023

Due dates	
Project group and Project choice	Tuesday April 4 using Piazza.
Feature extraction	Tuesday April 18 uploaded to Canvas.
Final project video presentation	Tuesday May 2 uploaded to Canvas.

This handout describes the details of the course project.

Objective: The objective of the project is to apply the estimation, detection and learning concepts learned in this class to a real-world dataset using Matlab or Python or a combination of these. Thus, it is *important for you to relate* the methods you use in the project to concepts, results and mathematical notations learned in class. Demonstrating understanding of the ideas discussed in the course is the main objective of the project - not obtaining the best solution for your stated problem. Don't use heuristics in situations where principled approaches were taught in class.

Logistics: Students should do the class project in groups of two. A common grade will be assigned to both students, unless the students in a group convey to us that they deserve different grades (due to unequal effort put into the project.) If the students disagree with each other on whether different grades should be assigned, a more thorough investigation will be conducted to resolve the dispute.

Project choice: A project that involves applying estimation, detection and learning concepts to one real-world dataset must be selected. *You must obtain the dataset yourself.* This can be raw data obtained from various data collecting agencies. It can also be data you collect using personal devices, such as a smartphone, microphone, camera, or using laboratory equipment. *Do not use data that has been pre-processed by others for the purpose of machine learning.* This is because thinking about the type of dataset you need for the type of inference problem you want to solve, and pre-processing and feature extraction from the dataset is part of the project. If you need an exception to this policy, you should discuss this with us before you propose the project.

A separate handout will point out some possible data collection ideas and projects.

Project timeline and grading: The project is worth 20 points, as specified in the Course Logistics handout. All deadlines are by 11:59 pm on the stated date. Formally, the grading is split into the following parts.

1. *Begin work on project (0 points)*: Starting March 31.
   Find a Group partner and agree with him/her on a project topic. You can use the `project_teammate_search` folder of Piazza to search for a group partner. For example

you could post a short note in that folder describing your topic of interest. Overly broad or overly narrow topics may not receive much interest. Make sure that any post you make here is public, so that it can be seen by other students.

2. *Group partner and Project choice (2 points)*: Notify the Teaching team by April 4.

   Each student must post a *separate note* in the `project_group_title_choice` folder of Piazza with the following four pieces of information:

   (a) Team member names
   (b) Project title
   (c) Three or four sentence description of project
   (d) A declaration stating that "I have read and understood the Software coding and plagiarism policy in the project handout".

   If you are unable to find a partner by the due date, you can post the above pieces of information, with only your name in 'Team member names' section. In that case, you will receive full points, but a project partner and project topic will be assigned to you randomly later.

   You can change the project on a later date if you wish, but you must inform us about the change by posting another note in the same Piazza folder that details the change, and also a convincing reason as to why this change is required. For example, a valid reason may be that the dataset you planned to use does not seem suitable for the task you had proposed, after you did data visualization and feature extraction. An invalid reason would be that you chose the project topic in a hurry, and are just beginning to think about what that project involves.

3. *Feature extraction (5 points)*: Uploaded to Canvas by April 18.

   Each group only needs to make one submission (this part is a group activity.) For the team member who is not the one submitting, they should make a TEXT ENTRY on Canvas for this assignment mentioning who is submitting the files for the group. The team member submitting must upload three distinct files:

   (a) A small sample of your dataset. This can be a zip file or a text, csv or excel file.
   (b) Any software code you have developed for *Feature extraction*. This can be a zip file or a Matlab file or Python notebook.
   (c) A picture showing visualization of features extracted from your dataset, as specified below. This must be a single image, pdf or word file. *Do not upload a zip file for this part.*

   To do this part, collect a dataset and and then extract features from this dataset using at least 3 *substantially different* feature extraction methods discussed in class. For example, LDA and Kernel LDA are considered similar to each other, PCA, Kernel PCA and LSA are considered similar, k-Means and Gaussian clustering are considered similar. Of the 3 methods, at least one of them should be used to show us a visualization of the dataset, and at least one of them should be used for subsequent regression or classification. For image/video/audio/text applications, at least one method must be a Manual feature extraction method. You can use a feature extraction method you have read about in a publication, if that is more suited to your application.

   If you spend significant effort in collecting data (such as making measurements yourself, say using a cell phone, instead of simply downloading data), you only need show us one feature extraction method.

4. *Detection/estimation/learning (8 points)*: Due by May 2.

   Apply at least 4 methods to solve a an interesting regression or classification problem involving your dataset, and compare the results. Choose these 4 methods from the options listed below - but each option can be selected only once.

   At least one method must give regression normalized MSE $< 10\%$ or classification accuracy $> 85\%$, where Normalized MSE $\doteq \frac{E[(y-\hat{y})^2]}{\sigma_y^2}$. For $M-$ary classification with large M (such as $M \geq 5$), you must obtain classification accuracy $> 50\%$. If these metrics of performance are not suitable for your application, you can discuss with us another, more appropriate, metric and guarantee you would prefer to use.

   - Use a Generalized Linear Model (which includes Linear Statistical Regression) or Logistic Regression or MSELE for regression, classification, and time series inference, respectively.
   - Use Kalman filter, Extended Kalman filter, Hidden Markov model or another Graphical model for inference.
   - Use robust inference methods (support vector regression or classification).
   - Use neural network or deep learning.
   - Use a Bayesian learning method.
   - Use Boosting or Bagging.
   - Use sparsity aware learning method.
   - Use any other inference method that we have discussed in class (e.g., Naive Bayes classification, Classification/Regression tree, or by proposing your own statistical model), or that you have read about in the text book or in research papers.

   It is recommended to use the Matlab or Python package (such as `sklearn`) that implements these methods, instead of coding the algorithms yourself.

5. *Final project presentation (5 points)*: Upload to Canvas by May 2.

   Each group only needs to make one submission (this part is a group activity.) For the team member who is not the one submitting, they should make a TEXT ENTRY on Canvas for this assignment mentioning who is submitting the files for the group. The team member submitting must upload three distinct files:

   (a) A video presentation. This must be in a standard video format, such as mp4.
   (b) Presentation slides. This can be a single Powerpoint file. Alternatively, this can be a single pdf file of slides that was developed using some other presentation software.
   (c) Software code you have developed for the project, as well as a small sample of the dataset. This can be a zip file.

   Each group will give one unified Powerpoint presentation, sharing presentation responsibility, record the video on your computer using Zoom, with presentation slides and the presenter's face visible, and then upload the video. The video must be between 15 to 20 minutes. Use the project presentation Powerpoint template on Canvas we provided you to plan this presentation. Points in this section will be awarded based on the clarity of your video presentation.

   In the presentation, you will present the problem, show the results of your investigations, and discuss the significance of the results. This will involve visualizing the dataset, showing the results of feature extraction, showing various performance graphs or numbers and discussing the choice of models and statistical methods that you used.

Software coding and plagiarism policy: You can write software code in either Matlab or Python or both, and you are allowed to use any packages (such as machine learning packages such as `sklearn`) for this purpose. While simple grammar can be copied from online sources, especially from package or function documentation, if you copy a large part of code from another source, you must properly cite the code by providing the original author's name and a link to the source. *Failure to do so will be considered plagiarism, which is a violation of Carnegie Mellon's Academic Policy, and penalties may occur even if the plagiarism is detected well after the semester has ended.* Citation must also be provided if you copy your own code, such as code you submitted for another course.

While properly cited copied code is allowed, if most of your project consists of stringing together pieces of code written by others, we may feel that you have not contributed sufficient original content to the project. You should be able to convince us that at least half the *intellectual content* of your project is your own. (This means half the intellectual content - not number of lines of code.) If you find you are unable to contribute sufficient intellectual content, and are instead relying heavily on copying code written by others, you should downsize the scope of your project, so that you can complete it based on your own proficiency with the chosen language. Remember that the project is primarily meant to show your understanding of the course material - the objective is not to use the latest/greatest methods, especially those that you are unable to code and explain yourself.

The examples below illustrate when copied code is considered significant or not.

Simple grammar example: Suppose you find an online source that shows you how to use sklearn's Logistic Regression function as follows:

```
clf = LogisticRegression(random_state=0)
clf.fit(X, y)
```

You can use this piece of code without citation, since it simply shows how to use that specific grammar of Python.

Significant code example: Suppose you find a piece of code that shows you how to extract HOG vectors from an image and then cluster them together. This is a significant piece of code that goes beyond simple grammar. If you use this code, you must cite it to credit the author, even if you change the code in some way (such as changing the names of the variables).