# Projects ML-II (40%) – Fall 2022

## Project 1: An API to Generate Joint PDF given Training Data

In this project, you are supposed to generate the joint PDF of any given training data and comment on its performance through a comprehensive analysis

**Outputs Required:**

* Notebook with all comments, observations, analyses, and instructions (23%) – includes marks for the innovativeness of your approach
* Demo (5%)
* Online blog summarizing the work (8%) (I will post on my social media)
* Word document containing all the references you considered in designing your approach and other miscellaneous/rough information (2%)
* Excel sheet documenting your results (2%)

The following scenario applies:

* **Case 1:** The user decides to use only numerical data in the training data
* **Case 2:** The user decides to use both numerical and non-numerical data (categorical, binary, timestamp, ordinal etc. are present)

**Execution Path:**

* Download 10 representative datasets from UCI ML Repository, with different frequency of rows and columns, and 5 for classification and 5 for regression.
  + At the start of the notebook, briefly summarize the name and description of each dataset
* Develop Joint PDF generation technique for Case 1 and Case 2
  + All research done should be referenced.
  + For Case 2, you need to mix different densities together (this is a bit complicated)
  + For numerical data, you can safely assume Gaussian, but the chances of exponential, beta, gamma are also there
  + Testing can only be done by analyzing classification and regression performance.
* Train your approach for Case 1 on 5 of the datasets
* Test your approach on the remaining 5 (add more here to acquire confidence of your approach)
* Train your approach for Case 2 on 5 of the datasets
* Test your approach on the remaining 5 (add more here to acquire confidence of your approach)
* Classification and regression are good testing techniques. But there are other “statistical distances” as well, e.g., KL-divergence/JS-divergence. How can you use these to determine whether a row belongs to the training data distribution or not? Use them to prove your answer.
* If you plan to use Auto-ML, then use at least 2 API’s to be sure of your results
* In the demo, I will download a dataset of my own choice and test it with your API
* Document all your results above in an Excel sheet
* At the end of the notebook, summarize your overall results:
  + Do you think it is an effective idea to learn the joint of training data and use that in ML tasks?
  + Do you think this approach will be able to handle general outliers in the data which occur from time to time?

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

## Project 2: Analysis of Distribution-based Clustering Methods

In this project, you will do a comprehensive analysis to make some conclusions regarding GMMs.

**Outputs Required:**

* Notebooks with all comments, observations, and analyses (23%)
* Online blog summarizing the work (8%) (I will post on my social media)
* Demo (5%)
* Word document containing all the references you considered in designing your approach and other miscellaneous/rough information (2%)
* Excel Sheet (2%)

**Execution Path:**

* Download 10 datasets from [this link](https://archive.ics.uci.edu/ml/datasets.php?format=&task=clu&att=&area=&numAtt=&numIns=&type=&sort=nameUp&view=table).
  + These should be different from the ones used in the assignment.
  + They should include both classification and pure clustering ones.
  + At the start of the notebook, briefly summarize the name and description of each dataset
* Use both extrinsic and intrinsic measures (as outlined in the assignment)
* Experiment with different values of all possible hyperparameters of GMM (see [here](https://scikit-learn.org/stable/modules/generated/sklearn.mixture.GaussianMixture.html)):
  + What effect does each parameter have on the final clustering performance (determined through both extrinsic and intrinsic measures (for classification and non-classification datasets)
  + Analyze the result for each parameter change thoroughly
  + Document results in Excel sheet and generate graphs if needed to put in blog
  + The above analysis should present a thorough understanding of the GMM approach.
* For all the 10 datasets above:
  + Pick the best GMM performer and compare its performance with Bayesian Gaussian Mixture ([here](https://scikit-learn.org/stable/modules/generated/sklearn.mixture.BayesianGaussianMixture.html)), Mini-Batch K-means, OPTICS, DBSCAN, Agglomerative, Spectral Clustering and Affinity Propagation
  + Pick a reasonable configuration for the latter 6 algorithms (you can also do hyperparameter tuning if you want to obtain full marks)
  + Comment thoroughly on the results of each comparison (document in notebook only)
  + Document results in Excel sheet and generate graphs if needed to put in blog

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

## Project 3: Analysis of Dimensionality Reduction Techniques in Python

In this project, you will do a comprehensive analysis of the effect of different dimensionality reduction techniques on ML performance.

**Outputs Required:**

* Notebooks with all comments, observations, and analyses (23%)
* Online blog summarizing the work (8%) (I will post on my social media)
* Demo (5%)
* Word document containing all the references you considered in designing your approach and other miscellaneous/rough information (2%)
* Excel Sheet (2%)

**Execution Path:**

* Download 15 datasets from UCI ML Repository (8 for classification and 7 for regression – use the categories in the left-hand menu).
  + At the start of the notebook, briefly summarize the name and description of each dataset
* Scikit learn offers different DR techniques:
  + A good blog from MLM: <https://machinelearningmastery.com/dimensionality-reduction-algorithms-with-python/>
  + An excellent one from Medium, informing about PCA variants, and showing the type of comparison I want: <https://medium.com/@deepak.engg.phd/dimensionality-reduction-with-scikit-learn-ee5d2b69225b>
  + Another one: <https://scikit-learn.org/stable/modules/unsupervised_reduction.html>
* You are supposed to consider the above, and information from other websites to make a list of all possible algorithms being offered by scikit.
* In the notebook, write 3-4 sentences on your understanding of each algorithm. This should not be copied, and should be entirely in your own words
* Determine a basic hyperparameter configuration for each algorithm (e.g., by considering what the blogs are doing – you can make any assumptions you want)
* For each dataset:
  + Apply each DR algorithm and document the result (in an Excel sheet)
  + Perform a thorough comparison of all algorithms for each dataset (document in notebook only)
* Now summarize the overall result
  + What seem to be the pros and cons of each (explain from your results)

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

## Project 4: Analysis of Statistical Inference Methods vs Traditional ML (bonus available)

Inference is a method of predicting the label (in both classification and regression) through Markov Chain Monte Carlo (MCMC) methods and Variational Inference (VI) methods

**Outputs Required:**

* Notebooks with all comments, observations, and analyses (23%)
* Online blog summarizing the work (8%) (I will post on my social media)
* Demo (5%)
* Word document containing all the references you considered in designing your approach and other miscellaneous/rough information (2%)
* Excel Sheet (2%)
* There will be bonus percentage marks on an excellent overall project

**Execution Path:**

* Download 10 datasets from UCI ML Repository (5 for classification and 5 for regression – use the categories in the left-hand menu).
  + At the start of the notebook, briefly summarize the name and description of each dataset
* You need to acquaint yourself with PyMC3 API: <https://docs.pymc.io/en/v3/index.html> (will cover in class)
* PyMC3 offers different MCMC and VI methods (<https://docs.pymc.io/en/v3/api/inference.html#module-pymc3.sampling>):
  + NUTS
  + Metropolis
  + Slice
  + Hamiltonian MC
  + Sequential MC
  + MultiTrace
  + OPVI
  + VI Inference API etc
* For each dataset:
  + Apply MCMC – this will require you to do research on determining priors and likelihoods first; document this in the notebook or Word document
  + Apply LazyPredict and compare with MCMC – document your analysis in notebook
  + Apply Variational Inference - this will require you to do research on determining the likely components on the algorithms; document this in the notebook or Word document
  + Compare LazyPredict outputs with VI – document your analysis in notebook
  + Put all results in Excel Sheet to generate possible charts
* Now summarize the overall result
  + What seem to be the pros and cons of each (explain from your results)