**Final Project for MGSC 695: Optimization for Data Science**

For the final project for this course, you are required to implement a machine learning

method of your choice from scratch and test its performance on an appropriate dataset.

That is, you cannot use a black box package or module of the method. However, you are

also expected to compare the performance of your implementation with any available blackbox implementation of the same method. Your project implementation will be judged on its

difficulty and the extent of execution.

**The due date for the final project is March 5th, 6 pm.**

Deliverables: Code & Report

The report will summarize the steps involved in the project. Ideally, this report will contain

1. description and details on the method thereby exhibiting your understanding of the method you have chosen,
2. details on the underlying optimization problem and the approach to solve it (the formulation and identify whether it’s a linear or a quadratic or a convex optimization problem etc., the optimization approach and so on).
3. You can also use the report to explain the challenges you faced and whether you were successful in overcoming them.
4. Please always cite any resources you have relied on.
5. The report should be no longer than 10 pages (at least single line spacing and 12 pt font).

Your project report will be evaluated by the depth of your understanding, clarity, presentation, and completeness. A list of potential topics is provided below:

You are also free to demonstrate your implementation on any appropriate dataset.

1. Mixture Models (GMM or other mixture models)

2. K-means clustering (related methods and extensions)

3. Regularized regression methods (Lasso, ridge regression, etc.)

4. Separating hyperplanes (Reference: Section 4.5 in Elements of Statistical Learning by

Gareth James et al.)

5. Tree-based methods (e.g., regression trees, decision trees)

6. Association learning methods (e.g., Apriori, ECLAT)

7. Hierarchical clustering methods

8. Dimensionality reduction (e.g., Principal Component Analysis)

9. Support Vector Machines