- **10%** of your final grade
- Due **16 June** (23:59pm; before midnight)
- Work independently
- Cite any and all resources used

Written answers should be included in **document** (PDF, RTF, ODT, DOC; in that order of preference) separate from your code. With your code, include clear instructions on how it should be run. This may be comments or mark-up in your code / notebook.

Submit your completed assignments as a single file to gerow@dal.ca with your name in the filename (eg. a1-[your_name].zip).

Q1: Regression

Regressions are a common statistical technique used to analyse the dependence between various parameters in a dataset. They provide both predictions, as well as estimates of various effects. There are, however, many kinds of regressions. The Python statsmodels package implements many types of regression, only some of which we covered in class.

The following dataset contains various information about housing in the Boston area:

https://archive.ics.uci.edu/ml/datasets/Housing

Move through the modeling work-flow (Lectures 6 & 7) to …

a) Understand the data.
   i) Compute summary / descriptive statistics.
   ii) Code any categorical data as numeric indicator columns.
   iii) Check for missing values, NaNs and major outliers (beyond +/- 4 s.d.)
b) Develop a hypothesis / conclusion:
   i) What dependent variable(s) do you think are related to other parameters? Many are related, but pick at least **two** conclusions you want to test. These should be interesting (conceptually) and statistically sound (you have a reason to believe the data could be used to support or refute them).
c) Chose a type of **regression** (from statsmodels):
   i) Linear regression, logit models, etc..

   For guidance, see https://stats.idre.ucla.edu/other/mult-pkg/whatstat/
d) Understand the important diagnostics
   i) This will take some outside research for some types of models. In addition to the statsmodels documentation UCLA has a great set of tutorials (http://stats.idre.ucla.edu/)
   ii) Make sure you understand how your models will be assessed.
e) Specify and fit your models:
   i) Report model-wide diagnostics (GoF and model-selection values like AIC)
   ii) Report estimates of the effects
   iii) Report (and plot; see below) coefficient estimates and confidence intervals from your models.

Your answer to this question should include…

a) Code for the major items above (lettered points; in a jupyter notebook).
b) A write-up, of two pages or less, about your results. This should include two to three sentences for every point above, and any plots and / or tables. You should write for someone who has not seen the data (and never will). Remember to highlight **findings**, not just your code.
c) Extra credit (possible 7 / 5): There are many ways to visualize the results of a regression. Have a look through the statsmodels and / or seaborn packages, and develop convincing visualizations of your results.

Q2: Classification

The idea of this task is to determine whether it is possible to predict whether an individual's annual income exceeds $50,000 based on demographic information. You will use this dataset extracted from the 1994 U.S. Census:

https://archive.ics.uci.edu/ml/datasets/Adult

Your task is to compare at least **three machine learning methods** to see which methods work best for this task when evaluated in the **testing set** (adult.test). On the other file (adult.data) you should use **the first 80%** of these data as **training** while **the remaining 20%** will be used for **validation**. Scikit-learn implements a number of suitable classification algorithms:

http://scikit-learn.org/stable/auto_examples/classification/plot_classifier_comparison.html

And be sure to properly preprocess your data:

http://scikit-learn.org/stable/modules/preprocessing.html

(Particularly Sections 4.3.1 and 4.3.4)

In addition to code, your answer to this question should be a one page write-up that explains your preprocessing choices, the classification models you tested and their results. Focus primarily on a comparison of their performance.