ISTE 780 DATA DRIVEN KNOWLEDGE DISCOVERY
HOMEWORK INSTRUCTIONS
MICK MCQUAID

INTRO

Part of your homework grade will depend on following these instructions carefully. These instructions serve two purposes. First, they make it easier for the instructor to concentrate on what you did right or wrong by removing variables like formatting. Second, they give you practice using R Markdown, an increasingly popular format for reporting machine learning results.

INSTRUCTIONS

R Markdown Files. For each homework assignment, you will turn in two files. You will turn in an R markdown file and a pdf file that results from rendering the R markdown file.

0. Do not zip any files you turn in. There should be a single `.Rmd` file named with the number of the homework as a lowercase m followed by an arabic numeral as well as a similarly named pdf file. For example, milestone 2 should be in two files called `m2.Rmd` and `m2.pdf`. MyCourses will add your name (plus a great many numbers) to the file name automatically.

1. Use a full-featured text editor that saves your files as UTF-8 or use R Studio.

2. Save the files with the extension `.Rmd` and make the file name match the letter m and the arabic numeral of the

assignment but in lower case. For example, the second assignment will be called `m2.Rmd` and so on.

3. The beginning of the file contains the following info: the homework number, title, your name(s), the date, and the output format. For example, for the second homework, my file would begin as follows (note that each line begins flush left).

```
---
title: 'm2: The Highway Safety Project'
author: 'Mick McQuaid'
date: '2018-09-14'
output:
    pdf_document:
        latex_engine: xelatex
---
```

4. Whenever your answer includes R code, write the code flush left surrounded by code fences. A code fence consists of a line with three backticks flush left and no other text except the letter r in curly braces (you can put certain other code inside the curly braces as well and you will learn that later) for the opening code fence and none at all for the closing code fence. Here is an example.

````
    ```{r label}
    library(ISLR)
    pairs(Auto)
    with(Auto,(plot(mpg,cylinders)))
````
````

```
sapply(Auto[,3:7],mean)
```
```

5. Notice the word `label` right after the letter `r` at the beginning of the preceding code chunk. Every code chunk needs a unique label or the R Markdown file will not render (knit) correctly. So you don't actually put the word `label` there, you put something unique, even if it is just `part1`, `part2`, etc.

6. If you want to include mathematical expressions, you may write them in LaTeX and surround them with dollar signs. This will allow you to say things like $\widehat{\mu}_0$. If you cannot understand LaTeX (it will be demonstrated for you and examples will be given), you can write the expressions out phonetically. For example, the above expression is pronounced *mu hat nought* and is written in LaTeX as `$\widehat\mu_0$`.

7. R markdown is documented in Chapters 21, 23, and 24 of the book *R For Data Science* as well as in other publications we will discuss. You will need to familiarize yourself with this format to some extent.

## PROJECT

**Project Overview.** For the project, you will work in teams of not more than four students on a problem of your choosing that is interesting, significant, and relevant to developing data analytics algorithms for knowledge extraction from a non-trivial dataset with a reasonable size. You will have great latitude in what you choose to work on, so take
```

advantage of this opportunity to make a big impact! This may form the basis for your capstone, so plan accordingly. Please choose your own partners for this project.

The primary requirements of the project are:

- Your project must use some non-trivial data that your team collects. Here are some widely known data repositories:
  - Open Government Data: https://www.data.gov, https://www.data.gov.uk, https://www.data.gouv.fr
  - Kaggle: https://www.kaggle.com
  - KDD Nuggets: https://www.kdnuggets.com/datasets/
  - UCI Machine Learning Repository: https://archive.ics.uci.edu/ml
  - StatDat: https://lib.stat.cmu.edu/StatDat
  - TwitteR: https://cran.r-project.org/web/packages/twitteR/index.html
  - rfigshare: https://figshare.com, https://cran.r-project.org/web/packages/rfigshare/index.html
- Your project must perform statistical learning algorithms to analyze the data.
- Choose one algorithm as your core algorithm and include four other algorithms for performance comparison. The core algorithm must be studied more closely and fine-tuned for better performance.
- Include at least one algorithm that has NOT been covered in the class. The new algorithm can be used as either the core algorithms or the comparing algorithms.

- For milestones 1 through 5, you *must* use R markdown to generate your .pdf files and you must include your .Rmd file along with the .pdf file in the Assignments box. This ensures that your report is reproducible and that your code is completely present and understandable.
- For milestones 1 through 5, you must include your data file or a URL that permits automatic retrieval of the data. In other words, if your data is supplied at a URL that can be understood by the `read.csv` function or a similar data-reading function in R, then you need not include the actual file in your submission because the instructor can just run your code to get the data. Otherwise, include your data file and use a `read.csv` or similar function (e.g., `fread()`) to read the file from the current working directory so that the instructor can easily run your code. Do not hard code your own directory structure into the reading function.
- For milestones 1 through 5, gradually add material so that the final product is a polished paper. In other words, the first milestone contains an introduction and motivation, and those sections will persist in your later submissions. The second milestone will add a data summary section to your paper. The third milestone will add a testing section, while the fourth milestone will add a fine tuning, final testing, and conclusion to the paper. The final milestone will add related work.
- The suggested page limits for the milestones reflect how much each milestone will *add* to the finished product. Each submission will grow by the size indicated. In other words, the first milestone will be two to three pages, while the second milestone will *add* three to five pages to that initial amount.
- Your code should be well documented. This can amount

to comments in the code chunks or narrative outside the code chunks. Part of the purpose of R markdown is to give you more flexibility in commenting your code than would be achievable in a .r file. Given that you can interleave code chunks and narrative, you can use formatted text, mathematical expressions, and images in addition to plain text.

Grading Criteria. The course project counts for seventy-five percent of your final grade. Your project will be graded based on five key milestones, each weighted as fifteen percent of your final course grade.

Milestone 1: Project proposal. Each group should post a 2-3 page project proposal as `m1.Rmd` and `m1.pdf` to Assignments on myCourses. Provide a brief, descriptive name of your project. Your name should be something memorable! In the proposal, you should address the following issues:

- Description of the problem.
- What is exactly the function of your system? That is, what will it do? (Supervised or Unsupervised? Regression or classification?)
- Why would we need such a tool and who would you expect to use it and benefit from it?
- You should mention the data and the candidate core algorithms that you will use.

Milestone 2: Data summary/visualization. For the second project milestone, which will add 3 to 5 pages to your report, you must have collected your dataset that your project will ultimately use. You should summarize the major characteristics of the data and use different ways to visualize the data. Data summary and visualizations should be presented in captioned tables and figures. Post your milestone 2 report as `m2.Rmd` and `m2.pdf` to Assignments on myCourses.

Milestone 3: Algorithm Testing. You should test at least three candidate algorithms and report the result along with your detailed analysis, which will include a comparison of the algorithms. Post your milestone 3 report as `m3.Rmd` and `m3.pdf` to Assignments on myCourses.

Milestone 4: Core Algorithm Fine Tuning. For the fourth project milestone, you must have chosen your core algorithm and included all four other algorithms for comparison. The core algorithm should be fine tuned to improve its performance. It is important to provide a detailed analysis about the performance of the core algorithm, especially what fine-tuning steps are conducted and whether/how they improve the core algorithm. Detailed comparison with other algorithms should also be provided. If you are fine tuning a random forest algorithm, you may want to look at https://machinelearningmastery.com/tune-machine-learning-algorithms-in-r/. For svm, look at https://blog.revolutionanalytics.com/2015/10/the-5th-tribe-support-vector-machines-and-caret.html. For comparisons, you may want to look at https://machinelearningmastery.com/compare-models-and-select-the-best-using-the-caret-r-package/. Post your milestone 4 report as `m4.Rmd` and `m4.pdf` to Assignments on myCourses.

Milestone 5: Final Project Report. Project final report revised based on any feedback from previous milestones.

- Final reports, submitted as `m5.Rmd` and `m5.pdf` to the Assignments dropbox, should include motivation, problem definition, key issues and alternative ways of resolving those, related work and their limitations, your approach, validation, conclusions (key contributions), and future work (assumptions and potential extensions).
- You are expected to visit office hours of the instructor

and meet with your peers to insure timely completion of each step.