# Individual Project

This is an individual assignment. **You may only communicate about this assignment with the instructor.** Any collaboration or other violation of the student code of academic integrity will be dealt with accordingly.

Your completed assignment must be submitted through the Individual Project assignment in compass before **11:59pm** on **Friday March 1, 2019**. As in homework, your submission must include your written report (Word, PDF, or knitted html document) and accompanying executable R (`.R` or `.rmd`) code file. Include your name in the file name for each file like we have done for homework. **R code must be in actual script files**, not code pasted into some other document. Efficiency of code matters.

**Any code based on reference material (e.g. code provided with the text or in course notes) must reference in code comments the source of the original code.**

**By accepting this assignment, you agree that you:**

- **will do the assignment by yourself**
- **will not discuss any portion of the assignment with anyone other than the instructor**
- **will abide by all aspects of the campus code of academic integrity (links available in the syllabus)**

To avoid having to re-compute the `msgWordsList` and `trainTable` results from Chapter 3, `.rda` files are provided. Place them in the directories shown in `IndividualProjectSetup.R`, and source `IndividualProjectSetup.R` to load those results, data frames saved to `.rda` files in Chapter 3, and a few functions from Chapter 3 you will likely need. You may also find others useful.

## Airline Delay Exercises

For the airline exercises, use data for 1990 through 1995.

You may need to consider whether you need to remove files to make space for the files for the object(s) you create, especially if you have created additional databases or `big.matrix` objects on your machine. Remember, you can use `df -h` to see free space, and `rm filename` for removing a file matching the `filename` pattern.

1) Create a `big.matrix` object for the airline data from 1990 to 1995, and add an additional variable to the matrix that indicates if the arrival was delayed. The variable should be 1 for arrival delays greater than 0 and 0 for arrival delays of 0 or less. Be sure to show the contents of any script files you ran at command line and any commands you ran at command line to process the data. These command line steps can be included as text in the report.
   Note: To demonstrate your `big.matrix` creation, you can first run the code for creation and then comment it out and use `attach.big.matrix` to define the matrix object. This way you can include the statement you used to create the object in your code submission without having to run it more than once. Also, the non-integer columns will not be used in these exercises, so you do not need to convert them to integers to complete the exercises. Without conversion, they will just become NAs.

   After constructing the object, obtain some basic descriptive statistics for arrival delays during this period of time. Specifically, compute the number of flights with and without known positive delayed arrival, percentage of flights with known positive delayed arrivals, and average known arrival time deviation from

expected (this will include all numeric arrival delays) in the data and comment on what those results tell us.

2) For this exercise, we explore quarterly statistics. Consider months 1 through 3 the first quarter, months 4 through 6 as the second quarter and so on.

   Obtain quarterly results for arrival delay percentiles (include all numeric values for arrival delays), average known arrival deviations (again, include all numeric values for arrival delays), and percentages of flights with known positive delay. For the percentiles, obtain enough percentiles to make a plot from 5% to 95%. Be efficient and avoid going through the data more often than necessary.

   Comment on differences in average known deviation across quarters and differences in percentages of flights with known positive delay across the quarters. Visually compare percentiles for the four quarters. Interpret what all of these results tell us about arrival delays across quarters from 1990 to 1995.

## Spam Detection Exercises

3) We used probabilities that a word did or did not appear in a message as the basis for Naïve Bayes classification in class. Now let's consider combining the words present and absent in the text with the hour of day the message was sent as the basis for a Naïve Bayes classification.
   Specifically, the basis will be

   $$P(\text{type} \mid \text{message body text AND hour in date field of header})$$

   where type is either spam or ham and hour is assumed independent of message text.

   Determine a good threshold for spam detection based on a Naïve Bayes model using the message text (words present and absent in the message body) and hour the message was sent. Comment on how the type I and type II errors compare to those for the model based only on message text content used in class, and to what degree the hour information improves or hurts spam detection.

   (Hint: Start with Bayes formula and the assumption that words and hour are independent to figure out what quantities are needed for the log likelihood ratio statistic. Then follow the examples from class to build an additional table storing the additional information needed to compute log likelihood ratios. To simplify coding, you can use the fact that the indices- the positions in the list- for email messages in `msgWordsList` and `emailDF` are the same, so your new log likelihood ratio can be a function of that index and use functions and results already obtained for the word lists and in the `emailDF`.)

4) Repeat the recursive partitioning fitting from the text using the data from the `easy_ham`, `hard_ham`, and `easy_spam` directories as training data and the `easy_ham_2` and `easy_spam_2` directories as test data. Use the default `rpart` control settings. Comment on similarities and differences between the most important classification features in this model and the model using random sampling and default control settings in the text. Compare error rates for the model using the new training and test data to the rates found using randomly sampled data. Also compare the error rates of the two models for `easy_ham_2` and `easy_spam_2` messages (the new test data). Provide thoughts on why any differences might exist.