In this assignment, we will use the mathematical notations from the following reference books:

- **Reference books:**

  1. *An Introduction to Statistical Learning with Applications in R* **(ISLR)**. Authors: James, G., Witten, D., Hastie, T., Tibshirani, R.

  2. *The Elements of Statistical Learning Data Mining, Inference, and Prediction* **(ESL)**. Authors: Hastie, T., Tibshirani, R., Friedman, J.

# 1 Programming

## 1.1 Color Compression using K-means

In this problem, you will use K-means clustering algorithm for color compression.

(a) **(5 points)** Read (load) the *'Picture.jpg'* image and show it in the report. What is the size of the image?

*Hint: In Python, you can use OpenCV to read and write images*

(b) **(15 points)** We can view the image pixels as a cloud of points. To do this, reshape the image to a 2D matrix of size [number of pixels x number of channels]. Rescale the colors so that they lie between 0 and 1. Now, visualize these pixels in this color space, using a subset of 10000 pixels. Plot the input color space as 2 plots: 1st plot Green vs Red, 2nd plot Blue vs Red

(c) **(20 points)** *Color compression:* Now reduce all the millions of colors (in original pixel space) to just K colors, where K = 8 and K = 16. For this, you will use K-means clustering algorithm. Show the two color compressed images in your report.

*Hint: K-means algorithm can be slow if the dataset is large. In Python, you may use mini batch K-means function to address this issue. Mini batch k-means operates on subsets of the data and computes the result faster than the standard k-means algorithm.*

(d) **(5 points)** Plot the new compressed color space as 2 plots: 1st plot Green vs Red, 2nd plot Blue vs Red.

(e) **(5 points)** Compare the color compressed images to the original image and discuss any observed differences.

## 1.2 Predictive Analytics of ICU data

**(Extra Credit: 50 points)** In this question, you will experiment with multiple classification algorithms for predicting mortality on a real-world ICU dataset.

**Dataset** We will use the *ICU dataset* from Physionet 2012 challenge. The training/validation/test sets are provided along with the assignment in Blackboard as `train.data`, `valid.data`, and `test.data`. For description of the ICU datasets, please refer to https://physionet.org/challenge/2012/.

**Classifiers** We will use Random Forests, and Deep learning models for predicting mortality of ICU patients.

**Performance Comparison** Compare the performance of two approaches (Random Forests, and Deep learning models) on the provided dataset.

**Random Forests: (15 points)** Train random forests using sklearn. Report the training, validation and test accuracy and AUROC for different hyperparameter settings. Set the number of trees in the forest to 10, 50, 100; maximum depth of the tree to 3 and 5. You may also change the class weights (optional).

**Deep Learning models: (35 points)** Implement a feed forward neural network for predicting ICU mortality. Try different number of layers (for example: 5, 10), different activation functions (For example: sigmoid, relu), and different optimizers (for example: Adam, RMSprop) and show all results (i.e. accuracy and AUROC for training, validation and test datasets). Discuss your findings.

**Submission Instruction:** You need to provide the followings:

- Provide your answers/plots/results to the programming problems in a PDF file, named as `IS777_hw#_fa19_YourLastName.pdf`. You need to submit the homework in electronic version as pdf file on Blackboard.
- Submit ALL the code and report via Blackboard. Recommend to use Python programming language. For your program, you MUST include the main function called `IS777_hw#_fa19.py` in the root of your folder. After running this main file, your program should be able to generate all of the results needed for this programming assignment, either as plots or console outputs. You can have multiple files (i.e your sub-functions), however, the only requirement is that once we unzip your folder and execute your main file, your program should execute correctly. Please double-check your program before submitting. You should only submit one `.zip` file. No other formats are allowed except `.zip` file. Also, please name it as `[lastname]_[firstname]_hw1_fa19.zip`.

**Collaboration:** You may collaborate. However, collaboration has to be limited to discussion only and you need to write your own solution and submit separately. You also need to list with whom you have discussed.