July 21

I. Vignette - Save and Load Models
   A. See save and load in tensorflow https://www.tensorflow.org/tutorials/keras/save_and_load
   B. Go to colab Lab6 Cats & Dogs https://colab.research.google.com/github/lmoroney/mlday-tokyo/blob/master/Lab6-Cats-v-Dogs.ipynb#scrollTo=5qE1G6JB4fMn
   C. Commands to **save model** in google colab
      1. !mkdir -p saved_model
      2. #model.save('saved_model/my_model')
      3. model.save('saved_model/my_model.h5') <- this will create your .h5 file, but be sure you have os and keras imported
   D. Download .h5 file, move to your project folder in PyCharm Projects, use either of the following
      1. model = tf.keras.models.load_model('my_model_gcl.h5')
      2. model = tf.keras.models.load_model('my_model')
   E. To save accuracy and loss from saved model use pickle
      1. See here https://stackoverflow.com/questions/51004071/how-can-we-plot-accuracy-and-loss-graphs-from-a-keras-model-saved-earlier
      2. To see list of tensorflow libraries and commands, go to the API link — here https://www.tensorflow.org/versions/r2.3/api_docs/python/tf
II. Responses
   A. Premade estimators
      1. How did you split the labels from the training set?  What was the name of the labels dataset?
      2. List 5 different estimators from tf.estimator and include the base command as you would write it in a script (for example this script used the tf.estimator.DNNClassifier() function from the API).
      3. What are the purposes input functions and defining feature columns?
      4. Describe the command classifier.train() in detail.  What is the classifier and how did you define it?  Which nested function (and how have you defined it) are you applying to the training and test detests?
      5. Redefine your classifier using the **DNNLinearCombinedClassifier()** as well as the **LinearClassifier()**.  Retrain your model and compare the results using the three different estimators you instantiated.  Rank the three estimators in terms of their performance.
   B. Build a Linear Model
      1. Using the dftrain dataset, upload an image where you used the seaborn library to produce a sns.pairplot().  Also include a histogram of age using the training set and compare it to the seaborn plot for that same feature (variable).  What interpretation can you provide of the data based on this plot?
      2. What is the difference between a categorial column and a dense feature?
      3. Describe the feature columns that have been input to your LinearClassifier().  How would you assess the result from your initial output?  What is the purpose of adding a cross featured column?  Did your attempt to capture the interaction between age and gender and incorporate it into your model improve performance?  Include and interpret your predicted probabilities and ROC curve plots.
III. For tomorrow
   A. Boosted trees and boosted trees with model understanding
   B. Skip image augmentation for now (we will pair it with the tensorflow exercise for next Monday i.e. See the data augmentation & image segmentation exercises) — Start **Tokenization for NLP**
   C. Final Project: finish draft **Problem Statement** >>> start identifying sources of **Data**