Project 3

This is a solo project. You'll be tasked with the following:

- Finding a data set you can fit supervised learning models with (you cannot use the datasets we've been using in class - there are many places with free data out there such as the UCI machine learning repository and kaggle)
- Using a numeric or binary response, fitting five different classes of models and choosing a best model
- Writing a narrative (via a notebook) with explanations and discussions as you go through the above. Your audience is someone that is considering taking this course. That is, they understand basic statistics and basic programming but are interested in understanding what the course is about. You should output a final .html or .pdf file and turn in both the notebook and final document. **Please read your data in via a URL of include the data as a file in your submission.**

Report Components

Introduction (10 pts)

Start by introducing the idea of supervised learning and describe what the point of using these models is. You should then discuss the dataset you'll be using (including where you found it) and the goals/question you want to answer with your supervised learning models.

Splitting the Data, Metrics, and Models (35 pts)

This section should start with a discussion of model metrics and describing two metrics you'll be using to judge your models. You should discuss why you are using the metrics and what their advantages/disadvantages may be. Explain **in detail** why we want to split our data into training and test sets and split your data.

You'll be fitting five different classes of models. One class of model should be one that we didn't cover in class. You should have subsections that describe the five methods that you'll be using (no code or anything here, just concepts and ideas about what the models are doing). These discussions should be clear to someone that knows statistics but doesn't know the modeling type/algorithm!

Model Fitting (45 pts)

Next, you should use Spark MLlib to fit your five different classes models to the training data. This should be done using pipelines and cross validation to choose your best model for each model type. You should compare your models using two different metrics. If the metrics differ in terms of which model to choose, think about what makes the most sense to do and explain why!

Notes:

- You should set up a pipeline in `pyspark` for each of your models
- You should do your transformations using the functions from MLlib to easily put them into the pipeline. At least one of the pipelines should use four or more transformations prior to the model fit (`estimator`)
  - VectorAssembler counts as a transformation
  - Doing something like a log transform counts as well
  - Adding polynomial terms or interaction terms counts
  - etc.
- You can use the same set of transformations for multiple models (if appropriate)

Model Testing (10 pts)

Lastly, you should evaluate the best models from each class on the test set and state which overall model was deemed the best.

Notes on grading:

- **If you are unable to get the modeling to work in pyspark, you can do everything via sklearn. However, you would then earn a maximum of 80 points on the project.**
- For each section, your grade will be lowered by 3-5 points for each each error (syntax, logical, or other) in the code or for each lacking description of a model etc.
- **You should use Good Programming Practices when coding (see wolfware). If you do not follow GPP you can lose up to 25 points on the project.**
- The reports should include a narrative throughout, section headings, etc. To be clear **be sure to include markdown text describing what you are doing, even when not explicitly asked for!** Points will be deducted from appropriate sections as appropriate.