Homework 1: Applications of Machine Learning in Hydrology (GEOL 701 – T)

Intro to Python and Scikit Learn

Due Feb 14

Install Anaconda and create a Jupyter notebook with the format of first initial, last name, underscore, homework number (this is the naming convention we will use for all python homeworks). For example, my submission to this assignment would be "mberghouse_hw1". You are allowed to use any resources including each other. Please email me your code and answers to any written/discussion questions at marc.berghouse@dri.edu

1. In this notebook complete the following basic programming tasks:
    a. Print "Hello World!"
    b. Use numpy to create an array of zeros with a shape of (400,400)
    c. Use a for loop to assign random float values (following gaussian distribution with mean of 128 and standard deviation of 20) to the array you created in the previous section
    d. Find a faster way to do c, using the numpy.random library
    e. Use %%timeit to determine how much faster method d is than method c
    f. Display a figure of size (10,10) of this random array with 300 DPI using plt.imshow()
    g. Use boolean indexing to set all values in the array greater than 128 to 0 and less than 128 to 255. The goal here is to turn a continuous random image into a binary mask (0 and 255), with values less than the mean equal to 0 and values greater than the mean equal to 255. If your array is all 0's or 255's, that is a problem.
    h. Display another figure of size (10,10) of this new array with 300 DPI using plt.imshow()
    i. Define a function to import and parse the USGS data ("USGS_data_sample.csv") and return a dataframe with appropriate column names
        i. Hint: Try this

```
count=0
df_gw=[]
!pip install csv
import csv
with open('/Users/marcberghouse/USGS_data_sample.csv', newline = '') as
data:
    data_reader = csv.reader(data, delimiter='\t')
    for row in data_reader:
        count=count+1
        if count>1953:
```

2. Import the following sklearn toy datasets: Iris and Diabetes (sklearn.datasets.**load_iris** and sklearn.datasets.**load_diabetes**). Hint: **from sklearn.datasets import** load_iris

a. Use pandas profiler to summarize the data (make a new pandas dataframe with the data.data and data.target arrays)
    i. !pip install pandas-profiling
    ii. from pandas_profiling import ProfileReport
    iii. profile = ProfileReport(df, title="Pandas Profiling Report")
b. Import logistic regression, linear regression, KNeighborsClassifier and KNeighborsRegressor from sci-kit learn
c. Which models should be used for the Iris Dataset? For the Diabetes Dataset?
d. Split both data sets into training and testing sets using the sklearn train_test_split
e. Use the appropriate models to generate predictions for the target variable of both datasets
    i. Train your model on the training sets
    ii. Test your model on the testing sets
f. For regression, plot your predictions vs actual values on a scatter plot
    i. Calculate and print the RMSE and R2 (you can find these in the sklearn metrics docs)
g. For classification, create a confusion matrix (look up in sklearn docs)
h. Which models work best for classification and regression? Why do you think that is?

3. Complete the Pandas Kaggle Tutorial listed in helpful links. Write down 5 things you learned from these tutorials (new libraries/functions, a different way to do something, a new concept, etc.). Take a screenshot of the completed tutorials and exercises.