Due data: 10/10/2018, end of the day. **Please submit the following 2 files via Canvas:**

1) For question 1 – 6, please submit in a word file or a PDF file;
2) For question 7, please submit a .ipynb file (Python jupiter notebook file).

I. Please answer the following questions related to Machine Learning concepts:

1. [*2 points*] How would you define Machine Learning?
2. [*6 points*] Explain the following concepts: 1) supervised learning, 2) unsupervised learning, 3) online learning, 4) batch learning, 5) model-based learning, 6) instance-based learning.
3. [*2 points*] What is *overfitting* of training data? What is *regularization*?
4. [*2 points*] What is *cross-validation*? Please describe a strategy of selecting validation data set.
5. [*2 points*] Please name several main challenges in Machine Learning?

II. Please answer the following questions related to Concept Learning:

6. [*10 points*] Consider the *EnjoySport* learning task and the hypothesis space H that we described in lecture 3. Let us define a new hypothesis space H' that consist of all pairwise disjunctions of the hypothesis in H. For example, a typical hypothesis in H' is

   <?, Cold, High, ?, ?, ?> ∨ *<Sunny, ?, High, ?, ?, Same>*

   Trace the Candidate-Elimination algorithm for the hypothesis space H' given the sequence of training examples of lecture 3 (i.e., show the sequence of S and G boundary sets):

Sky	Temp	Humid	Wind	Water	Forecst	EnjoySpt
Sunny	Warm	Normal	Strong	Warm	Same	Yes
Sunny	Warm	High	Strong	Warm	Same	Yes
Rainy	Cold	High	Strong	Warm	Change	No
Sunny	Warm	High	Strong	Cool	Change	Yes

III. Programming problem:

7. [*16 points*] Implement the **Find-S algorithm** in **Python**. Please use **Jupiter notebook** tool to include your code and your results and discussion, and submit **a single .ipynb file**.

   1) First verify that it successfully produces the trace that we produced in lecture 3 (slide #32) for the *EnjoySport* example.
   2) Then use this program to study the number of random training examples required to exactly learn the target concept. Implement a training example generator that generates random instances, then classifies them according to the target concept:

      <Sunny, Warm, ?, ?, ?, ?>

   Consider training your Find-S program on randomly generated examples and measuring the number of examples required before the program's hypothesis is identical to the target concept. Run the experiment at least 20 times and report the mean number of examples required.