

CS 486/686 Assignment 4

Due Date: 11:59pm on Monday, December 3, 2018

Instructions

- Submit the assignment in the Dropbox labeled Assignment 4 Submissions on Learn.
- No late assignment will be accepted.
- This assignment is to be done individually.
- For any programming question, you may use the language of your choice. We highly recommend using Python if you are familiar with it.
- Lead TAs: Aravind Balakrishnan (a4balakr) and Benjamin Armstrong (b8armstr). Their office hours will be posted on the course website.

1 Markov Decision Processes (35 marks + 5 bonus marks)

In class, we studied the 4 by 3 grid world and computed the optimal policy for this world when $R(s) = -0.04$ and the discount factor is 1. The transition model stays the same: the agent moves in the intended direction with probability 0.8, moves to the left of the intended direction with probability 0.1, and moves to the right of the intended direction with probability 0.1.

You will study this grid world for two other values of $R(s)$.

	1	2	3	4
1				
2		X		-1
3				+1

1. (15 marks) Consider $R(s) = -0.05$.

Execute the value iteration algorithm for 10 iterations. You may do this by hand, by using an excel spreadsheet, by running a program, or by using any other tool available to you.

(3 marks) Show your work for calculating $U_3(s_{13})$ based on the values of U_2 .

For iteration i , show the following

- (5 marks) A diagram showing the values of the true utility $U_i(s)$ for each state s .
- (5 marks) A diagram showing the optimal policy for each state.
If multiple actions achieve the best expected utility, show all of the actions as the optimal policy for a state.
- (2 marks) Highlight the states for which the optimal policy for the state changed from iteration $i - 1$ to iteration i .

2. (15 marks) Consider $R(s) = -0.1$.

Execute the value iteration algorithm for 10 iterations. You may do this by using any tool available to you.

(3 marks) Show your work for calculating $U_3(s_{13})$ based on the values of U_2 .

For iteration i , show the following

- (5 marks) A diagram showing the values of the true utility $U_i(s)$ for each state s .

- (5 marks) A diagram showing the optimal policy for each state.
If multiple actions achieve the best expected utility, show all of the actions as the optimal policy for a state.
 - (2 marks) Highlight the states for which the optimal policy for the state changed from iteration $i - 1$ to iteration i .
3. (5 marks) Compare the optimal policies in iteration 10 for $R(s) = -0.05$ and $R(s) = -0.1$.
- (2 marks) List all of the states for which the optimal policies in iteration 10 are different for the two values of $R(s)$.
- (3 marks) Why are the optimal policies in iteration 10 different for these states?
4. (5 bonus marks) Are the optimal policies for $R(s) = -0.04$ and $R(s) = -0.05$ different? If they are the same, state the optimal policy. If they are different, list the states for which the two optimal policies are different and describe the difference.

2 Learning Decision Trees (75 marks)

You will implement a decision tree learning algorithm to classify samples into one of three species of the Iris flower (Iris setosa, Iris versicolor, and Iris virginica) based on four features measured from each sample: the length and the width of the sepals and petals, in centimeters. The structure of the provided data set is identical to that of the Iris Dataset available on `scikit-learn.org` but their contents are different.

You are provided with data set A in `set_a.csv` on the course website. Data set A has 100 data points. Each data point has four features and a classification, all separated by commas. The four features in order are sepal length (cm), sepal width (cm), petal length (cm), and petal width (cm). The three classes in order are setosa, versicolor, and virginica. For example, the data point (4.9,2.6,6.0,2.4,2.0) means that the Iris flower has sepal length 4.9 cm, sepal width 2.6 cm, petal length 6.0 cm, petal width 2.4 cm, and belongs to the class Iris virginica.

We generated a separate data set B , which has 200 data points. Both data sets A and B are generated using the same decision tree. Unfortunately, data set B is not available to you.

Both data sets have real-valued features. When generating a decision tree, use only binary tests at each node. That is, you should choose a threshold value for each feature and each node in the decision tree should test a feature has a value greater or smaller than a threshold value. At each node, decide the feature to test and the threshold value using the information gain metric as follows.

- a. For each feature, order all of its values in the remaining examples.
- b. For each value that is half way between two consecutive feature values, compute the information gain assuming that we use this value as the threshold value.
- c. Choose the combination of feature and threshold value that results in the highest information gain.

Note that, along a path from the root node to a leaf node, a feature may be tested multiple times with different threshold values.

If your code is incomplete or it is not working well, use the following command to answer the questions to earn up to 48 marks. See the detailed mark breakdowns in each section.

```
sklearn.tree.DecisionTreeClassifier(criterion='entropy')
```

Please complete the following tasks.

1. Use the ID3 algorithm to generate a decision tree based on data set A . At each node, choose a feature and a threshold value for the feature using the information gain metric. Make sure that your decision tree achieves a classification accuracy of 100% on the training set.

What to submit: (25 marks)

You can earn up to 16 marks (3 + 13) if you complete this part using the `DecisionTreeClassifier` from `scikit-learn`.

- (5 marks) Well-documented code for generating the decision tree. Clearly describe how the TA should run your program.
 - (20 marks) Show the generated decision tree which classifies all training examples perfectly. Draw this tree by hand or show a print out of the tree from your program.
2. A decision tree that achieves perfect classification accuracy may not generalize well to unseen data. You will experiment with generating smaller trees with the hope of achieving better accuracy on the test set.

Implement a modified version of your algorithm which takes as input a maximum depth. Build decision trees with increasing maximum depth until a full tree is obtained.

To determine the best maximum depth for the decision tree, you will use ten-fold cross-validation. In ten-fold cross-validation, each example serves double duty – as training data and validation data. First, split the data into ten equal subsets. Then, perform ten rounds of learning. On each round, $\frac{1}{10}$ of the data is held out as a validation set and the remaining examples are used as training data. For each value of the maximum depth, generate a decision tree with the maximum depth using only the training data. Then, calculate the average prediction accuracy of the decision tree on the training data and on the validation data. Finally, choose the maximum depth with the highest average prediction accuracy.

What to submit: (35 marks)

You can earn up to 22 marks (3 + 13 + 2 + 3 + 1) if you complete this part using the `DecisionTreeClassifier` from `scikit-learn`.

- (5 marks) Well documented code for generating decision trees with different maximum depths and for performing cross-validation. Clearly describe how the TA should run your program.
- (20 marks) Plot the average prediction accuracy with respect to the maximum depth of the decision tree on the training set and on the validation set.
10 marks are for plotting the prediction accuracy on the training set and 10 marks are for plotting the prediction accuracy on the validation set.

- (3 marks) What is the maximum depth of your decision tree that maximizes the average prediction accuracy of the generated tree on the validation set?
 - (5 marks) Generate a decision tree with the best maximum depth using the entire data set A . Draw this tree by hand or show a print out of the tree from your program.
 - (2 marks) What is the prediction accuracy of the decision tree with the best maximum depth on data set A ?
3. We will evaluate the best decision tree you came up with in part 2 using data set B . To do this, you will need to write a program, which produces the best decision tree based on data set A , reads in data set B and outputs the prediction accuracy of the tree on data set B . Data set B has the exact same format as data set A . However, data set B has 200 data points where data set A has 100 data points. Thus, you can test your program on data set A to make sure that it runs correctly.

What to submit: (15 marks)

You can earn up to 10 marks (3 + 7) if you complete this part using the `DecisionTreeClassifier` from `scikit-learn`.

- (5 marks) Clear instructions for the TA to run your program to generate the prediction accuracy of your decision tree on data set B .
- (10 marks) You will get 10 marks if your decision tree achieves a prediction accuracy of at least 80% on the test set. You will get 8, 6, 4, 2, or 0 marks if your decision tree achieves a prediction accuracy of at least 70%, 60%, 50%, 40%, or 30% respectively.