Course

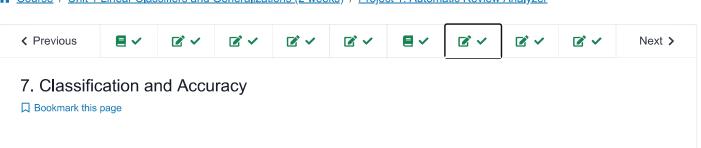
**Progress** 

<u>Dates</u>

**Discussion** 

Resources

☆ Course / Unit 1 Linear Classifiers and Generalizations (2 weeks) / Project 1: Automatic Review Analyzer



Project due Oct 7, 2020 05:29 IST Completed

Now we need a way to actually use our model to classify the data points. In this section, you will implement a way to classify the data points using your model parameters, and then measure the accuracy of your model.

#### Classification

1.0/1 point (graded)

Implement a classification function that uses  $\theta$  and  $\theta_0$  to classify a set of data points. You are given the feature matrix,  $\theta$ , and  $\theta_0$  as defined in previous sections. This function should return a numpy array of -1s and 1s. If a prediction is **greater than** zero, it should be considered a positive classification.

**Available Functions:** You have access to the NumPy python library as <code>np</code> .

**Tip::** As in previous exercises, when x is a float, "x=0" should be checked with  $|x|<\epsilon$ .

```
1 def classify(feature_matrix, theta, theta_0):
 2
 3
      A classification function that uses theta and theta_0 to classify a set of
4
      data points.
6
7
          feature_matrix - A numpy matrix describing the given data. Each row
8
              represents a single data point.
9
                 theta - A numpy array describing the linear classifier.
10
          theta - A numpy array describing the linear classifier.
11
          theta_0 - A real valued number representing the offset parameter.
12
13
      Returns: A numpy array of 1s and -1s where the kth element of the array is
14
      the predicted classification of the kth row of the feature matrix using the
      given theta and theta_0. If a prediction is GREATER THAN zero, it should
          considered a positive classification
```

Press ESC then TAB or click outside of the code editor to exit

Correct

#### Test results

```
See full output
CORRECT
See full output
```

Submit

You have used 2 of 25 attempts

## Accuracy

1.0/1 point (graded)

We have supplied you with an accuracy function:

```
def accuracy(preds, targets):
    """
    Given length-N vectors containing predicted and target labels,
    returns the percentage and number of correct predictions.
    """
    return (preds == targets).mean()
```

The <code>accuracy</code> function takes a numpy array of predicted labels and a numpy array of actual labels and returns the prediction accuracy. You should use this function along with the functions that you have implemented thus far in order to implement <code>classifier accuracy</code>.

The classifier\_accuracy function should take 6 arguments:

• a classifier function that, itself, takes arguments (feature\_matrix, labels, \*\*kwargs)

- the training feature matrix
- the validation feature matrix
- the training labels
- the valiation labels
- a \*\*kwargs argument to be passed to the classifier function

This function should train the given classifier using the training data and then compute compute the classification accuracy on both the train and validation data. The return values should be a tuple where the first value is the training accuracy and the second value is the validation accuracy.

Implement classifier accuracy in the coding box below:

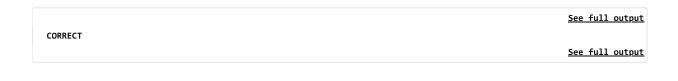
**Available Functions:** You have access to the NumPy python library as <code>np</code>, to <code>classify</code> which you have already implemented and to <code>accuracy</code> which we defined above.

```
1 def classifier_accuracy(
 2
          classifier,
          train_feature_matrix,
4
          val_feature_matrix,
 5
          train labels,
 6
          val_labels,
7
          **kwargs):
8
9
      Trains a linear classifier and computes accuracy.
10
      The classifier is trained on the train data. The classifier's
11
      accuracy on the train and validation data is then returned.
12
13
14
          classifier - A classifier function that takes arguments
15
              (feature matrix, labels, **kwargs) and returns (theta, theta_0)
           train feature matrix
                                - A numny matrix describing the training
```

Press ESC then TAB or click outside of the code editor to exit

Correct

### Test results



Submit

You have used 1 of 25 attempts

## **Baseline Accuracy**

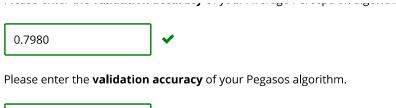
3/3 points (graded)

Now, uncomment the relevant lines in **main.py** and report the training and validation accuracies of each algorithm with T = 10 and  $\lambda$  = 0.01 (the  $\lambda$  value only applies to Pegasos).

Please enter the **validation accuracy** of your Perceptron algorithm.



Please enter the **validation accuracy** of your Average Perceptron algorithm.



0.7900

Submit

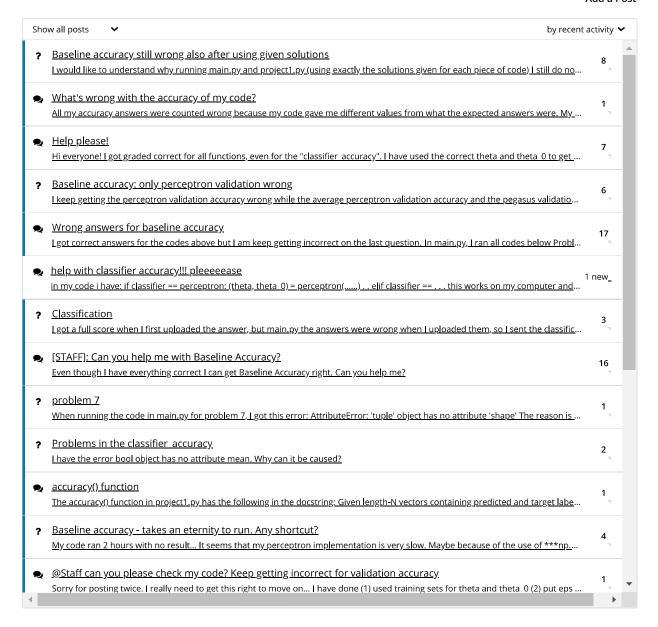
You have used 3 of 20 attempts

## Discussion

Hide Discussion

**Topic:** Unit 1 Linear Classifiers and Generalizations (2 weeks):Project 1: Automatic Review Analyzer / 7. Classification and Accuracy

Add a Post



⟨ Previous | Next >

© All Rights Reserved



# edX

<u>About</u>

<u>Affiliates</u>

edX for Business

Open edX

Careers

<u>News</u>

# Legal

Terms of Service & Honor Code

Privacy Policy

Accessibility Policy

Trademark Policy

<u>Sitemap</u>

## Connect

<u>Blog</u>

Contact Us

Help Center

Media Kit

**Donate** 















© 2020 edX Inc. All rights reserved. 深圳市恒宇博科技有限公司 <u>粤ICP备17044299号-2</u>

/