<u>Help</u> smitha_kannur -

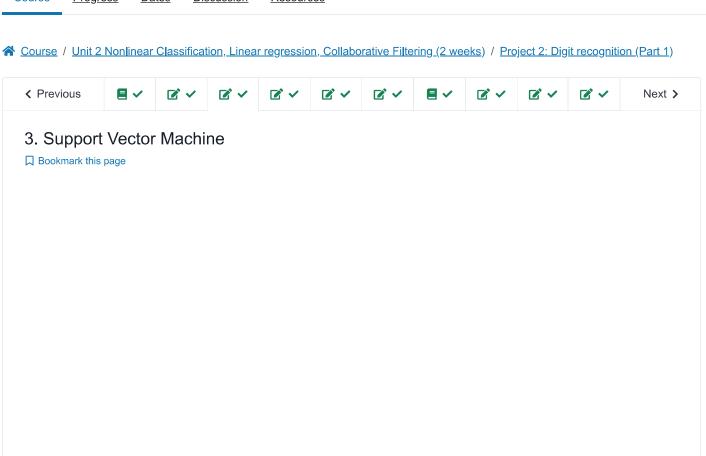
Course

<u>Progress</u>

<u>Dates</u>

Discussion

Resources



Project due Oct 21, 2020 05:29 IST Completed

Bob thinks it is clearly not a regression problem, but a classification problem. He thinks that we can change it into a binary classification and use the support vector machine we learned in Lecture 4 to solve the problem. In order to do so, he suggests that we can build an one vs. rest model for every digit. For example, classifying the digits into two classes: 0 and not 0.

Bob wrote a function run_svm_one_vs_rest_on_MNIST where he changed the labels of digits 1-9 to 1 and keeps the label 0 for digit 0. He also found that sklearn package contains an SVM model that you can use directly. He gave you the link to this model and hopes you can tell him how to use that.

You will be working in the file part1/svm.py in this problem

Important: For this problem, you will need to use the <u>scikit-learn</u> library. If you don't have it, install it using pip install sklearn

One vs. Rest SVM

5.0/5.0 points (graded)

Use the sklearn package and build the SVM model on your local machine. Use random_state = 0, C=0.1 and default values for other parameters.

Available Functions: You have access to the sklearn's implementation of the linear SVM as LinearSVC; No need to import anything.

```
1 def one_vs_rest_svm(train_x, train_y, test_x):
 2
3
      Trains a linear SVM for binary classification
4
5
 6
         train_x - (n, d) NumPy array (n datapoints each with d features)
7
          train_y - (n, ) NumPy array containing the labels (0 or 1) for each training data point
8
          test_x - (m, d) NumPy array (m datapoints each with d features)
9
      pred_test_y - (m,) NumPy array containing the labels (0 or 1) for each test data point
"""
10
11
12
      try:
13
          svm = LinearSVC(random_state=0, C=0.1)
14
          svm.fit(train_x, train_y)
15
          pred_test_y = svm.predict(test_x)
           return nred test
```

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

Correct

Test results

```
CORRECT

See full output

See full output

You have used 2 of 25 attempts
```

Binary classification error

5.0/5.0 points (graded)

Report the test error by running run_svm_one_vs_rest_on_MNIST.

Submit

You have used 1 of 20 attempts

1 Answers are displayed within the problem

Implement C-SVM

0.0/5.0 points (graded)

Play with the C parameter of SVM, what statement is true about the C parameter?

(Choose all that apply.)

✓ Larger C gives larger tolerance of violation.	

Larger C gives a larger-margin separating hyperplane.

Larger C gives smaller tolerance of violation. 🗸





Solution:

C represents the tolerance of error. A larger C means we are punishing more on the classification error, thus being less tolerant to misclassifications. Therefore, we will get a smaller margin hyperplane.

Submit

You have used 2 of 2 attempts

1 Answers are displayed within the problem

Multiclass SVM

5.0/5.0 points (graded)

In fact, sklearn already implements a multiclass SVM with a one-vs-rest strategy. Use LinearSVC to build a multiclass SVM model

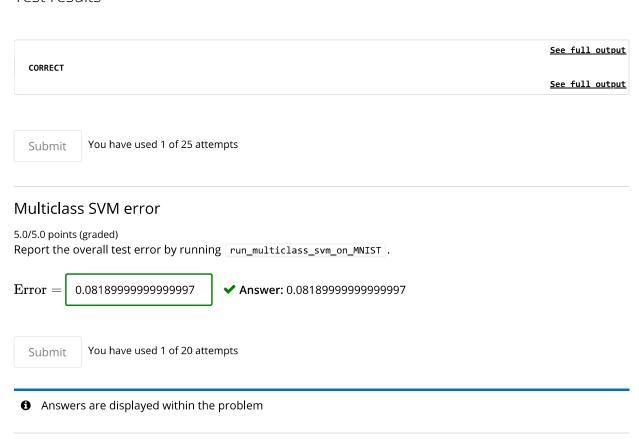
Available Functions: You have access to the sklearn's implementation of the linear SVM as LinearSVC; No need to import anything.

```
1 def multi_class_svm(train_x, train_y, test_x):
 3
      Trains a linear SVM for multiclass classification using a one-vs-rest strategy
 4
5
      Args:
 6
          train_x - (n, d) NumPy array (n datapoints each with d features)
7
          train_y - (n, ) NumPy array containing the labels (int) for each training data point
8
          test_x - (m, d) NumPy array (m datapoints each with d features)
9
      Returns:
10
          pred_test_y - (m,) NumPy array containing the labels (int) for each test data point
11
12
      try:
13
          svm = LinearSVC(random_state=0, C=0.1)
14
          svm.fit(train_x, train_y)
15
          pred_test_y = svm.predict(test_x)
```

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

Test results

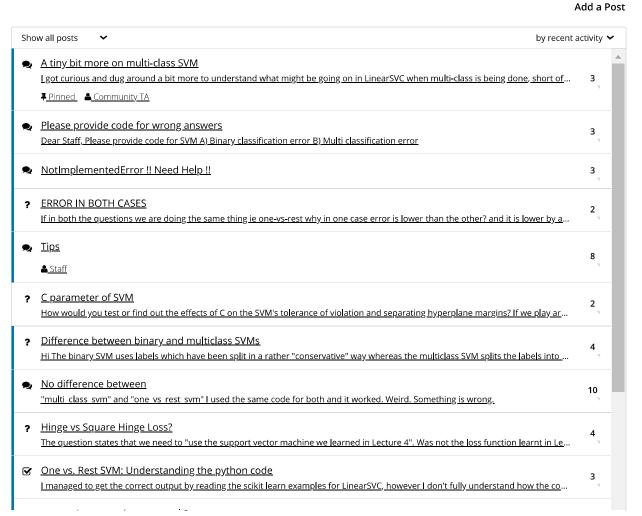
Discussion

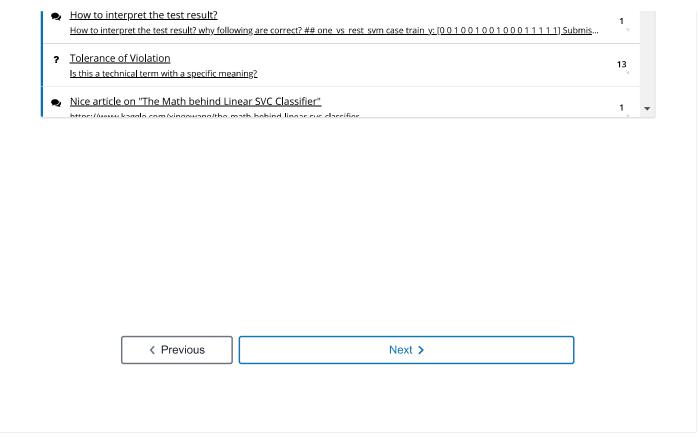


Topic: Unit 2 Nonlinear Classification, Linear regression, Collaborative Filtering (2

weeks):Project 2: Digit recognition (Part 1) / 3. Support Vector Machine

Hide Discussion





© All Rights Reserved



edX

<u>About</u>

<u>Affiliates</u>

edX for Business

Open edX

Careers

News

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>