8. Parameter Tuning

You finally have your algorithms up and running, and a way to measure performance! But, it's still unclear what values the hyperparameters like T and λ should have. In this section, you'll tune these hyperparameters to maximize the performance of each model.

One way to tune your hyperparameters for any given Machine Learning algorithm is to perform a grid search over all the possible combinations of values. If your hyperparameters can be any real number, you will need to limit the search to some finite set of possible values for each hyperparameter. For efficiency reasons, often you might want to tune one individual parameter, keeping all others constant, and then move onto the next one; Compared to a full grid search there are many fewer possible combinations to check, and this is what you'll be doing for the questions below.

In **main.py** uncomment Problem 8 to run the staff-provided tuning algorithm from **utils.py**. For the purposes of this assignment, please try the following values for T: [1, 5, 10, 15, 25, 50] and the following values for λ [0.001, 0.01, 0.1, 1, 10]. For pegasos algorithm, first fix $\lambda = 0.01$ to tune T, and then use the best T to tune λ

Performance After Tuning

7/7 points (graded)
After tuning, please enter the best T value for each of the perceptron and average percepton algorithms, and both the best T and λ for the Pegasos algorithm.

Note: Just enter the values printed in your main.py. Note that for the Pegasos algorithm, the result does not reflect the best combination of T and λ.

For the **perceptron** algorithm:

$T =$ 25.0000 ✔

With validation accuracy =

0.7940 ✔

For the **average perceptron** algorithm:

$T =$ 25.0000 ✔

With validation accuracy =

0.8000 ✔

For the **pegasos** algorithm:

$T =$ 25.0000 ✔

$\lambda =$ 0.0100 ✔

With validation accuracy =

0.8060 ✔

Submit You have used 1 of 20 attempts

✔ Correct (7/7 points)

Accuracy on the test set

1/1 point (graded)

After you have chosen your best method (perceptron, average perceptron or Pegasos) and parameters, use this classifier to compute testing accuracy on the test set.

We have supplied the feature matrix and labels in `main.py` as `test_bow_features` and `test_labels`.

Note: In practice the validation set is used for tuning hyperparameters while a heldout test set is the final benchmark used to compare disparate models that have already been tuned. You may notice that your results using a validation set don't always align with those of the test set, and this is to be expected.

Accuracy on the test set :

0.8020 ✔

Submit You have used 1 of 20 attempts

✔ Correct (1/1 point)

The most explanatory unigrams

10/10 points (graded)

According to the largest weights (i.e. individual i values in your vector), you can find out which unigrams were the most impactful ones in predicting **positive** labels. Uncomment the relevant part in `main.py` to call `utils.most_explanatory_word`.

Report the top ten most explanatory word features for positive classification below:

Top 1 : delicious ✔

Top 2: great ✔

Top 3: ! ✔

Top 4: best ✔

Top 5: perfect ✔

Top 6: loves ✔

Top 7: wonderful ✔

Top 8: glad ✔

Top 9: love ✔

Top 10: quickly ✔

Also experiment with finding unigrams that were the most impactful in predicting negative labels.

Submit You have used 1 of 20 attempts

✔ Correct (10/10 points)

Discussion

Hide Discussion

Topic: Unit 1 Linear Classifiers and Generalizations (2 weeks):Project 1: Automatic Review Analyzer / 8. Parameter Tuning

Add a Post

Show all posts ▼ by recent activity ▼

- For the most explanatory unigrams, are we supposed to use the most performant algorithm used in the previous question? — 1
  Dear all, In the last question "The most explanatory unigrams", which of the 3 algorithms are we supposed to use, the most performant one used for the previous questio...
- best_theta ? — 11
  Why don't we need bias for `best_theta` ? I read the codes in utils.py but can't understand what the following 2 lines are doing wordlist = [word for (idx, word) in sorted(zi...
- Run time for Performance after Tuning exercise — 1
  Hi everyone, I wanted to ask if anyone else is also having to wait a long time for the script to complete its run for the first exercise. I have tried multiple times in the last fe...
- The idea of regularization in these alogrithoms — 1
  I think i miss (forgot) some important theory points after days of coding, need a clarification on the following: In the lectures, we introduced the idea of the trade off betw...
- ? Accuracy on the test set — 2
  Hi everyone, I completed all the other exercises up to this section (including "The most explanatory unigrams"), but I get an error in "accuracy on the test set", supposedly_...
- Data to use for "most explanatory unigrams" — 7
  In the question it should be clarified which data set we should use to obtain the list of top ten positive words. There are three possibilities and they each result in a differe...
- ? Accuracy on test set — 6
  Hi, Do I misunderstand the question? I pick the best T (and L if needed) from the previous question, where I have green ticks all the way....and use those parameters on th...
- why avg.perceptron need more computation resource? — 3
  My personal computer(i7 -7500U, 8GB RAM) arise "MemoryError" while tuning avg.perceptron, no error occur when tune perceptron and pegasos. Why avg.perceptron ne...
- Why are explanatory words mostly positive? — 6
  Almost all of the top 20 explanatory words are positive-sounding words. Is there general lesson? Or is it just an artefact of this particular data set?
- Performance After Tuning — 4
  I got all prior points passed with the grader but the results here are not all accepted. Someone else having the same issue?. For Perceptron it accepted T but not Best, for ...
- [Staff] Typo in last bullet point of answer to "Performance After Tuning": "betterperformance"

Title says it all. 1

The most explanatory unigrams 1

Love this question, a creative idea.

Grid parameters search vs random one 2

It's probably worth to mention that grid search is not the better method for hyperparameters tuning. See [Random Search for Hyper-Parameter Optimization][1]. [1]: http:...

Learn About Verified Certificates