Assignment 8

Learning Objectives:

1. Practice optimization and error analysis skills

Description:
The data set you will be working with for this assignment is again the same data set you used for the midterm, but now it has been divided into 5 train/test folds using the stratified remove folds filer. The original dataset, which you used for Assignment 7, is also included. You might remember we discussed it in an earlier lecture.

Step-by-Step Guide:

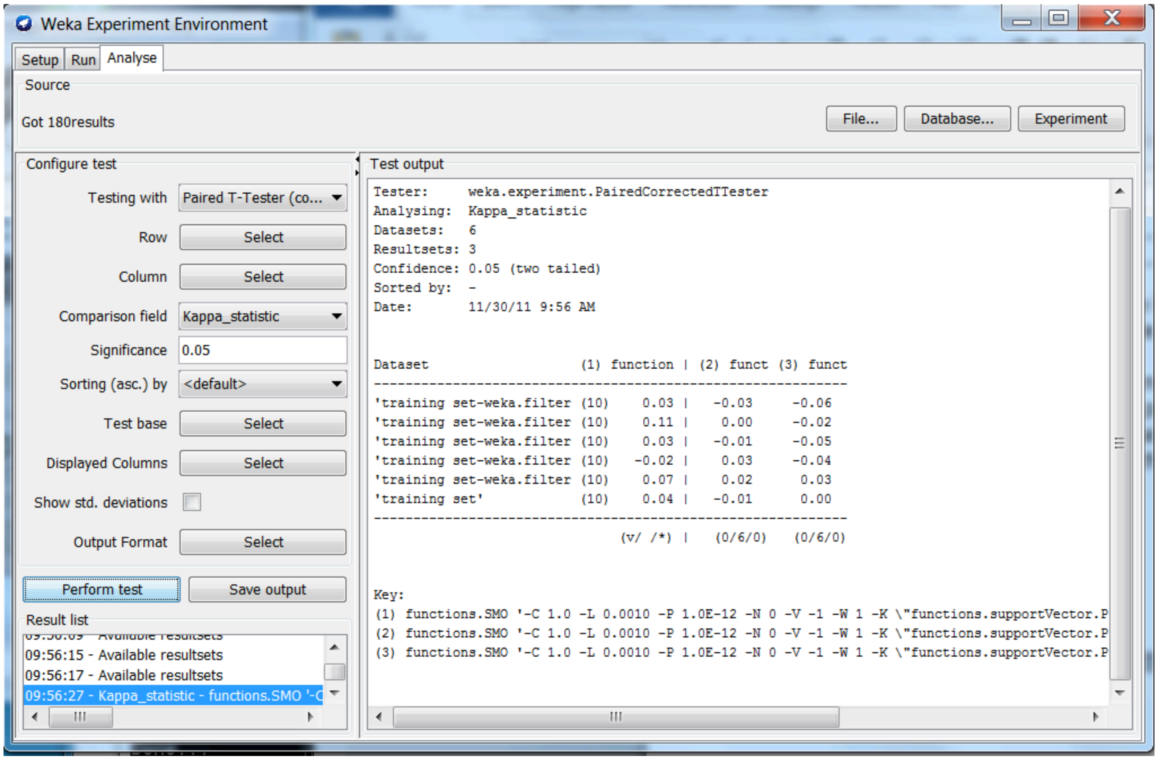
1. Complete the week 10 and 11 assigned readings about linear and statistical models, and review the lecture slides related to doing optimization in the Experimenter.

2. Use the experimenter to find the optimal setting for the Exponent parameter of the polynomial kernel on the SMO algorithm. Try 1.0, 2.0, and 3.0 as potential settings for that parameter. Evaluate the results in terms of Kappa. Fill in the following table (except for the Test set performance) from the results you see in the Experimenter:

Fold	Train Set Performance For Exponent = 1.0	Train Set Performance For Exponent = 2.0	Train Set Performance For Exponent = 3.0	Optimal Setting	Test Set Performance
1	.03	-.03	-.06	1	.027
2	.11	.00	-.02	1	-.015
3	.03	-.01	-.05	1	.06
4	-.02	.03	-.04	2	-.028
5	.07	-.01	0	1	-.069

Average = -.005

Also take a screen shot of the Experimenter when you do the analysis so that we can verify that you are able to interpret the output correctly.



Got 180results

Testing with: Paired T-Tester (co...

Comparison field: Kappa_statistic

Significance: 0.05

Sorting (asc.) by: <default>

Show std. deviations: □

Result list:

```
09:56:15 - Available resultsets
09:56:17 - Available resultsets
09:56:27 - Kappa_statistic - functions.SMO '-C
```

```
Tester:     weka.experiment.PairedCorrectedTTester
Analysing:  Kappa_statistic
Datasets:   6
Resultsets: 3
Confidence: 0.05 (two tailed)
Sorted by:  -
Date:       11/30/11 9:56 AM


Dataset                   (1) function | (2) funct (3) funct
------------------------------------------------------------
'training set-weka.filter (10)    0.03 |   -0.03     -0.06
'training set-weka.filter (10)    0.11 |    0.00     -0.02
'training set-weka.filter (10)    0.03 |   -0.01     -0.05
'training set-weka.filter (10)   -0.02 |    0.03     -0.04
'training set-weka.filter (10)    0.07 |    0.02      0.03
'training set'            (10)    0.04 |   -0.01      0.00
------------------------------------------------------------
                               (v/ /*) |   (0/6/0)   (0/6/0)


Key:
(1) functions.SMO '-C 1.0 -L 0.0010 -P 1.0E-12 -N 0 -V -1 -W 1 -K \"functions.supportVector.P
(2) functions.SMO '-C 1.0 -L 0.0010 -P 1.0E-12 -N 0 -V -1 -W 1 -K \"functions.supportVector.P
(3) functions.SMO '-C 1.0 -L 0.0010 -P 1.0E-12 -N 0 -V -1 -W 1 -K \"functions.supportVector.P
```

3. Now, using the Explorer and the optimal setting you observed on each fold, compute the test set performance by training a model using the optimal setting over the training data on that fold and then applying that model to the test data for that fold. Fill in these values under Test Set Performance.

4. Now using the Explorer and the whole set of data rather than the separate folds, determine what setting you would use to build a model over the whole set. What is that setting, and how would you estimate what performance you would get for that model over a new set of data?

The best setting was 1.0, which is also the default setting. The performance we would expect that model to get on a new dataset is the average of the test set performances we got in 3.

1. Looking at the comparison across settings and thinking about what the different settings mean in terms of what the algorithm is doing with the data when it builds its model, what conclusions can you draw?

When doing the kind of analysis where you're trying to understand the preference for one algorithm over another or one configuration of an algorithm over another, you should be taking a combined top-down/bottom-up approach where you start with what we discussed in lecture were the main differences and similarities between families of algorithms and different configurations within those families. Those ideas can help you form hypotheses about what circumstances would lead to a difference or not. Then you can proceed in a hypothesis driven fashion to look for confirming or disconfirming evidence related to the hypothesis. One way of doing that for this assignment is illustrated below.

The fact that nonlinear models typically did not improve performance over a linear model suggests that there are not important interactions between attributes in the feature space. The question is whether in Fold 4 where a different choice was made over the training set, there appeared to be such an interaction, and that is why a suboptimal choice was made.

I can test this hypothesis by first making a comparison over the whole set between the performance of linear and nonlinear in a different family of algorithms. So I will try that using naïve bayes and bayes nets.

Over the whole set Naïve Bayes gets .69 Kappa. Bayes nets gets .16 Kappa. This confirms the hypothesis that the reasons for the overall pattern of results we got was because of a lack of need for representing interactions between attributes.

Now I'll do the same thing on Fold 4 where a different choice was made. On that fold Naïve Bayes gets .84 kappa and Bayes Net gets .29 kappa. That doesn't give us any evidence that the reason for a different preference was because the subset of data inadvertently appeared to have such interactions. Considering that on no fold was there ever a statistically significant difference between options, this suggests that the small difference between linear and nonlinear on that one fold was just random.

Deliverables: Turn in the table you filled in, the screen shot, and the write up for 3 – 5.

Submission Mode:

Submit the assignment to blackboard as usual.