AutoWeka

An Automated Data Mining Software Based on Weka

AutoWeka

User Guide

Version 1

Chanin Nantasenamat
Saksiri Jamsak
Likit Preeyanon
Chartchalerm Isarankura-Na-Ayudhya
Virapong Prachayasittikul

Center of Data Mining and Biomedical Informatics
Faculty of Medical Technology
Mahidol University

AutoWeka is available at http://www.mt.mahidol.ac.th/autoweka

Last updated on October 12, 2012

Contents

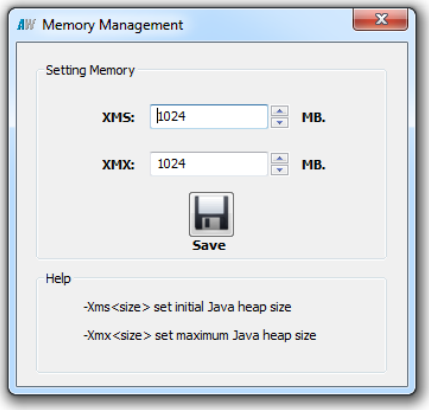
1. Installing and running AutoWeka

1. Download AutoWeka from http://www.mt.mahidol.ac.th/autoweka
2. Unzip the AutoWeka-1.0.zip file into the root folder of your hard drive such as at *C:* so that the folder resides at *C:\AutoWeka*.
3. Go into the AutoWeka folder and double click on AutoWeka.exe file.
4. You should now be able to see the AutoWeka program window as follows:

2. Adjusting the memory setting

1. On the main menu, click on Tools → Memory setting

2. A new window will appear and in here adjust the memory to the desired value. Here both the *XMS* and *XMX* are set as default at 1024 MB.



3. Building predictive models

Creating predictive models with AutoWeka is relatively simple, as it requires only a few steps that will be described in this user manual.

3.1 Creating the ARFF input file

Firstly, users must prepare input files in the *Attribute-Relation File Format* (ARFF). The ARFF file format is essentially a *comma-separated value* (CSV) file format that contains a description of the variables as the header portion of the file.

For a given CSV file that looks like the following:

```
x1, x2, x3, x4, y
0.29, 0.34, 0.00, 0.70, 0.59
0.65, 0.61, 0.38, 0.08, 0.25
0.95, 0.60, 0.21, 0.73, 0.66
0.45, 0.44, 0.91, 0.24, 0.32
```

The contents of the corresponding ARFF file will look as follows:

```
@relation

@attribute x1 numeric
@attribute x2 numeric
@attribute x3 numeric
@attribute x4 numeric
@attribute y numeric

@data
0.29, 0.34, 0.00, 0.70, 0.59
0.65, 0.61, 0.38, 0.08, 0.25
0.95, 0.60, 0.21, 0.73, 0.66
0.45, 0.44, 0.91, 0.24, 0.32
```

The ARFF file for a data set called *ABC* should thus be saved as *ABC.arff*. It can be seen that the first four variables are the independent variables while, as default, the last variable is the depedent variable.

It should be noted that in the above example, all variables are quantitative in nature and its attribute description is thus *numeric*. For a data set with qualitative variable, braces are used to encompass the unique values present in the given data set as shown in the example below.

```
x1, x2, x3, x4, y
low, 0.34, 0.00, 0.70, yes
medium, 0.61, 0.38, 0.08, no
high, 0.60, 0.21, 0.73, yes
medium, 0.44, 0.91, 0.24, no
```

The corresponding ARFF file will look as follows:

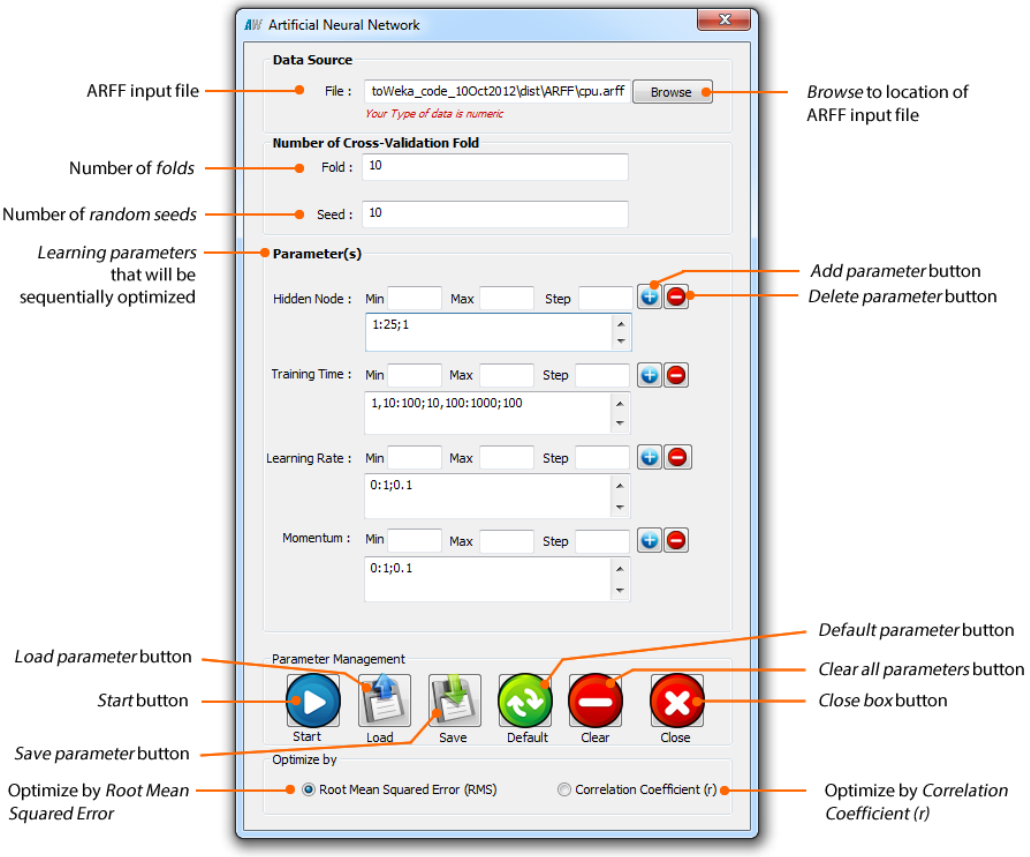
```
@relation

@attribute x1 {low, medium, high}
@attribute x2 numeric
@attribute x3 numeric
@attribute x4 numeric
@attribute y {yes, no}

@data
low, 0.34, 0.00, 0.70, yes
medium, 0.61, 0.38, 0.08, no
high, 0.60, 0.21, 0.73, yes
medium, 0.44, 0.91, 0.24, no
```

Note: Preparation of the ARFF input file can be performed in a text editor such as *Notepad*++ (available for free at http://notepad-plus-plus.org).

3.2 Build models with Artificial Neural Network

1. **Initiating Artificial Neural Network calculation.** To build artificial neural network (ANN) models, click on Run → Artificial Neural Network.

2. **Artificial Neural Network Parameters.** A new window will appear asking for the ARFF input file to use and other necessities such as the parameter values to use during model development.



2.1. **ARFF input file.** Users should start by clicking on the *Browse* button under the *Data Source* section to select the ARFF input file to use. After that has been done, notice the red text immediately below the path to the ARFF file, which will tell you the data type of the dependent *Y* variable whether it is quantitative or qualitative (this is important for support vector machine calculations and will be discussed later).

2.2. **Number of folds.** By default, the number of folds to use will be set to *leave-one-out cross-validation* meaning that the number is dependent on the sample size of the data set. Therefore, the *N* value of the data set will be used as the default number of folds. For example, a data set comprising of 100 samples will have 100 folds. For this example, we will set the fold number to 10. (Note: Cross-validation is data sampling approach that divides the data set into N sets of data and leaves out 1 set as the testing set while using the remaining *N*-1 as the training set by which a predictive model is constructed. The constructed predictive model is then tested on the set of data that was left out. This is iteratively performed for *N* times until all sets had a chance to be used as the testing set)

2.3. **Number of seeds.** By default, the number of random seed is set to 10. The seed number has something to do with the random initialization of the weight values that interconnects the nodes of the neural network. This value can be left as is because it should give reasonable performance.

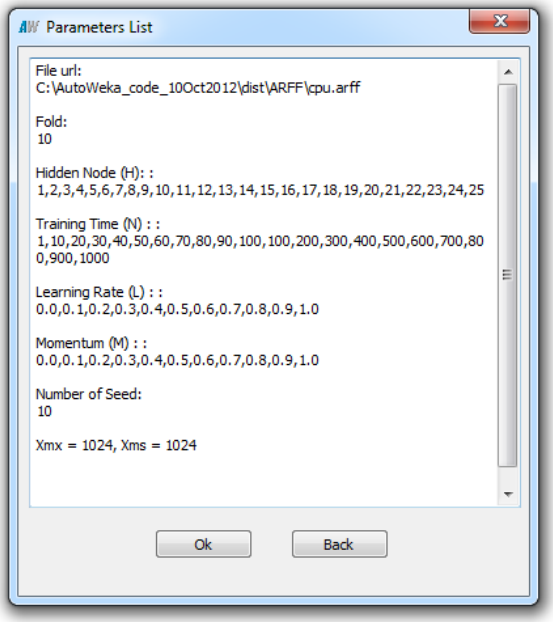
2.4. **Parameters.** Users are recommended to use the default parameters as it should yield satisfactory performance. Therefore, go ahead and click on the *Default* button. Just in case that you would like to make modifications to the parameters, feel free and enter the values that you desire by placing the values inside the *Min*, *Max* and *Step* boxes (make sure that the big bottom box is blank before doing this, if it is not blank then go ahead and clear the values in the box) and click on the "+" button.

2.5. **Parameter Management.** Users can save and load parameter settings by invoking the *Save* and *Load* buttons. Users can also start all over again by clicking on the *Clear* button to clear the values of all parameters.

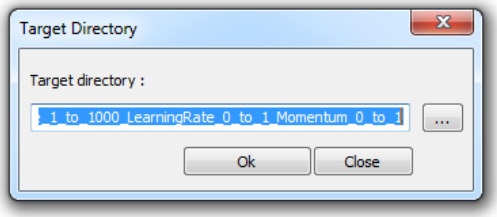
2.6. **Optimize by.** AutoWeka provides the option to optimize parameters based on one of two approaches by using: (1) root mean squared error or (2) correlation coefficient as performance metrics by which to judge the relative performance of the predictive models. For example, if the *root mean squared error* is selected then a set of learning parameters providing the lowest root mean squared error is deemed to provide good performance. However, if *correlation coefficient* is used as the performance metric then models affording the highest correlation coefficient can be deemed to provide good performance.

2.7. Finally, users can either choose to proceed with the calculation by clicking on the *Start* button or cancel the calculation by clicking on the *Close* button.

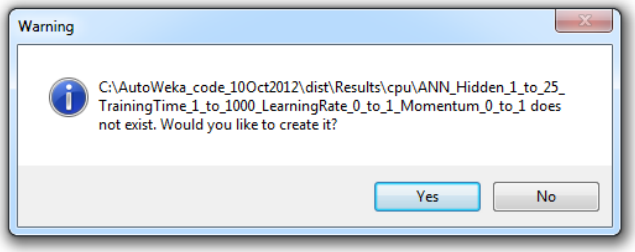
3. **Parameter List.** After all parameters have been entered the program will generate a summary of the parameters that will be used in the forthcoming calculation, to proceed click on the *OK* button:



4. **Target directory.** The program will by default generate the file path for which the constructed models will reside. Users only have to click on the *OK* button to proceed.

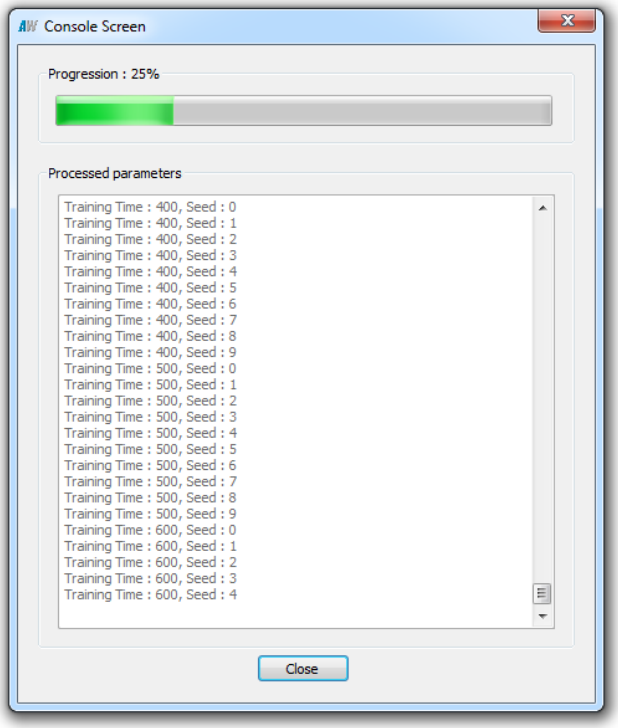


5. **Warning.** The program may issue a warning saying that the folder that it will create does not yet exist and is asking for our confirmation. Users only have to click on the *OK* button to proceed.

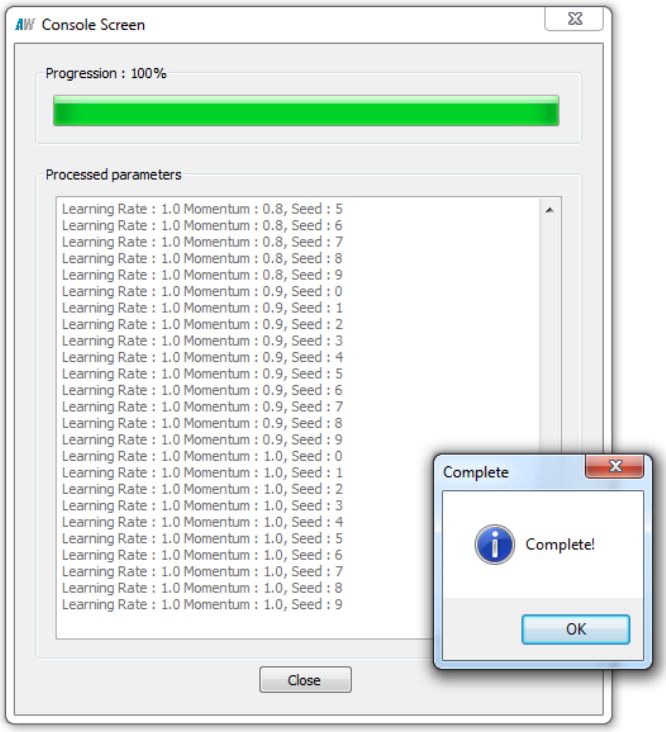


6. **Status of the Calculation.** The calculation will now begin and the progress toward completion will be illustrated by the green status bar located at the top with the

processed and currently calculating parameter shown in the large white box located immediately below the green status bar.



7. **Calculation completed.** When the calculation has completed it will produce a pop-up message as shown below:



8. **Retrieving the results.** Once the calculation has completed, the results can be obtained from the *Results* folder. If the data set is called ABC.arff then a sub-folder called *ABC* should reside in the Results folder; double clicking on this folder should produce the following folder named *ANN_Hidden_1_to_25_TrainingTime_1_to_1000_LearningRate_0_to_1_Momentum_0_to_1* and inside this folder there will be 3 additional folders comprising of *HiddenNode*, *LearningAndMomentum* and *TrainingTime*.

8.1. **Data of prediction results.** A text file with the full data of the prediction results is provided in the *AvgHiddenNode.txt*, *AvgLearningAndMomentum.txt* and *AvgTrainingTime.txt*. These files can be copied and directly pasted into Microsoft Excel or a graphical plotting software for further analysis and plot creation.

Here, we provide an example of the contents of *AvgHiddenNode.txt*:

Hidden_Node	Training_correlation	Training_RMS	Testing_correlation	Testing_RMS
1	0.99852	10.75714	0.99438	16.6819
2	0.9996	4.95634	0.99391	17.37294
3	0.99958	5.1518	0.99345	17.93404
4	0.99959	4.97472	0.99364	17.80626
5	0.99957	5.09627	0.99256	19.1421
6	0.99955	5.46855	0.99307	18.38153
7	0.99958	5.16741	0.9929	18.67296
8	0.99952	5.32637	0.99296	18.66021
9	0.99939	6.02274	0.99389	17.41026
10	0.99942	5.75727	0.99461	16.45766
11	0.99941	5.73905	0.99431	16.77958
12	0.99941	5.87036	0.99453	16.5145
13	0.99938	5.7581	0.99498	15.73311
14	0.9994	5.69449	0.9942	16.91954
15	0.99938	5.79578	0.99493	15.80735
16	0.99934	5.93469	0.99539	15.04039
17	0.99931	6.07683	0.99555	14.79371
18	0.99933	6.16435	0.99634	13.35235
19	0.99931	6.15614	0.99606	13.91939
20	0.99931	6.09891	0.9962	13.6734
21	0.99933	6.45891	0.99605	13.91758
22	0.99932	6.35406	0.99676	12.59459
23	0.99933	6.48	0.99666	12.70992
24	0.99934	6.64052	0.99684	12.42863
25	0.99936	6.5201	0.99722	11.62854
26	0.99935	6.27937	0.99758	10.86828
27	0.99937	6.67993	0.99749	11.07855
28	0.99935	6.64846	0.99736	11.36284
29	0.99935	7.08234	0.99738	11.22487
30	0.99934	6.9431	0.99667	12.56199
31	0.99933	6.65729	0.99649	12.7985
32	0.99934	6.88492	0.99665	12.48713
33	0.99931	7.13273	0.99674	12.4174
34	0.9993	6.53199	0.99689	12.24278
35	0.99931	6.57954	0.99691	12.08423
36	0.99926	7.30046	0.99692	12.18978
37	0.99927	7.48889	0.99629	13.44079
38	0.99922	8.41166	0.99514	15.0335
39	0.99921	7.97689	0.99589	14.05054
40	0.99911	9.69131	0.99469	15.76061

The contents shown above were parsed from several raw data text files as described in the following section 8.3.

8.2. Summary of prediction results. A summary text file of each step of the parameter optimization process is provided by *SummaryHiddenNode.txt*, *SummaryLearningAndMomentum.txt* and *AvgTrainingTime.txt*.
Here, we provide an example of the contents of *SummaryHiddenNode.txt*:

```
Method
---------------------------------------
weka.classifiers.functions.MultilayerPerceptron

Data set
---------------------------------------
C:\AutoWeka\tmpARFF\cpu.arff

Optimal Hidden Node : 26
Optimal RMS = 10.86828
Optimal Correlation = 0.99758

-----------------------------------
Min RMS = 10.86828
Max RMS = 49.84573

Min Correlation Coefficient = 0.9669
Max Correlation Coefficient = 0.99758
```

8.3. Raw and parsed data. The above files are results that have been parsed and post-processed to be in a format that is ready for further analysis. Additionally, all the raw data of the investigated parameters are also provided in each of the 3 parameters folders (i.e. *HiddenNode*, *LearningAndMomentum* and *TrainingTime*).

Contents of the raw data of each investigated parameters generate results as obtained from a typical Weka calculation as follows (here the contents of *0001H1S0.txt* are shown, *H1* denotes 1 hidden node and *S0* denotes the seed number of 0):

```
Options: -H 1 -S 0
Linear Node 0
    Inputs    Weights
    Threshold 2.116750666116899
    Node 1    -3.2466345721829906
Sigmoid Node 1
    Inputs    Weights
Threshold    0.9470785920512156
    Attrib X1    0.057669952658492896
    Attrib X2    -0.5796561833140699
    Attrib X3    -1.0111148589116203
    Attrib X4    -0.43028414725180286
Class
    Input
    Node 0

Time taken to build model: 0.08 seconds
Time taken to test model on training data: 0.02 seconds
```

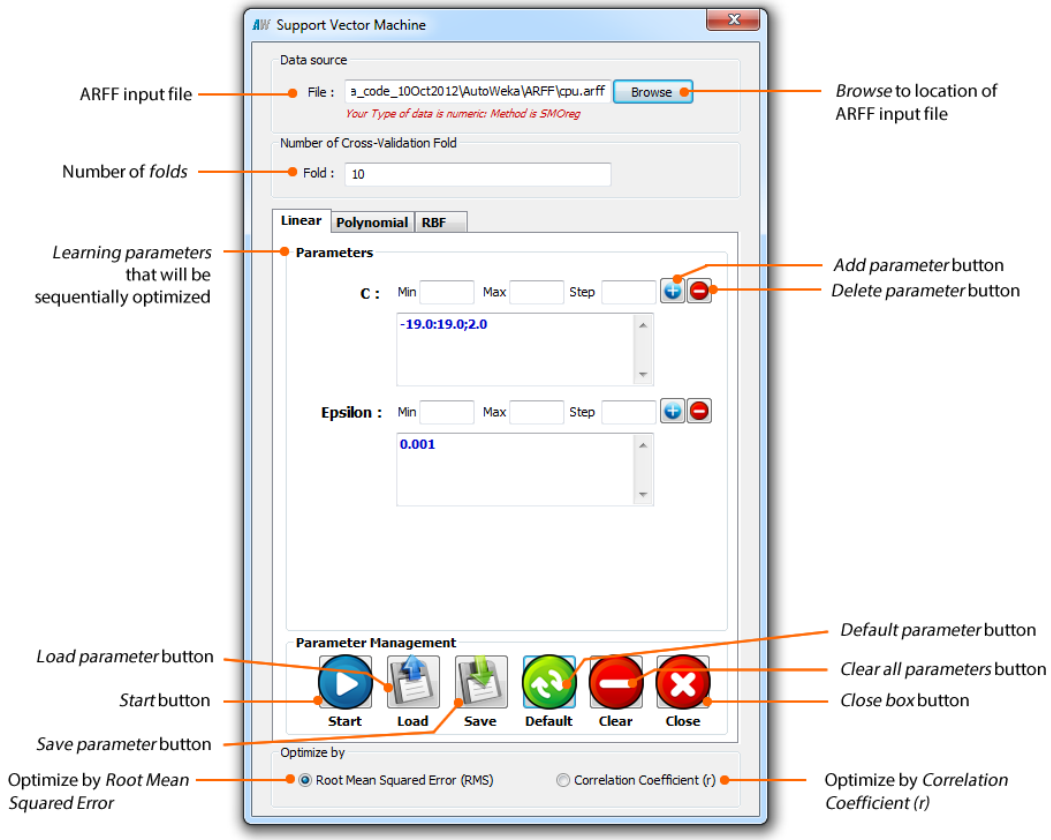
```
=== Error on training data ===
Correlation coefficient                  0.9986
Mean absolute error                      5.1198
Root mean squared error                  8.4464
Relative absolute error                  5.859
% Root relative squared error            5.4709 %
Total Number of Instances              209
=== Cross-validation ===
Correlation coefficient                  0.9937
Mean absolute error                      9.873
Root mean squared error                 17.4651
Relative absolute error                 11.2631
% Root relative squared error           11.285
% Total Number of Instances            209
```

9. **Plotting graphs of the results.** Now that we have obtained the necessary data, we can go ahead and make some plots, which will be useful in visually assisting us in identifying the best set of learning parameters. More information is provided in section 4.

3.3 Build models with Support Vector Machine

1. **Initiating Support Vector Machine calculation.** To build support vector machine (SVM) models, click on Run → Support Vector Machine.

2. **Support Vector Machine Parameters.** A new window will appear asking for the ARFF input file to use and other necessities such as the parameter values to use during model development.



In this window, we can see that there are tabs to three different learning kernels (i.e. linear, polynomial and RBF) and the default tab is currently set to *Linear* kernel. The linear kernel essentially requires the optimization of the C parameter and here we will explore the C values from -19 to 19 that is from 2^{-19} to 2^{19} in sequential steps of 2 or 2^2. The epsilon value may also be optimized but in this case we will use the default value.

We will go ahead and show the *polynomial kernel* tab as follows:

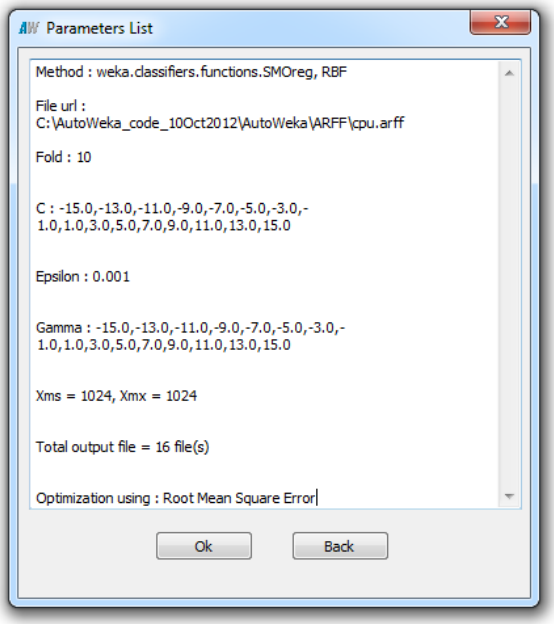
In addition to the C parameter, the polynomial kernel also explores the exponential value starting from 2 onwards (it should be noted that an exponential value of 1 is equivalent to that of a linear kernel).

The radial basis function kernel or the *RBF* tab is shown below. Here, an additional parameter called the gamma value is also a critical parameter to optimize.

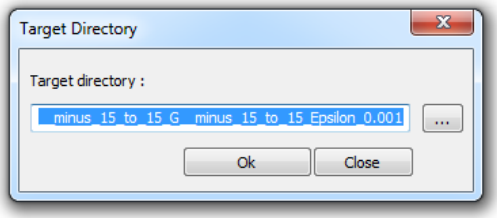
Here, the default C and Gamma values will explore values from -19 to 19 that is from 2^{-19} to 2^{19} using incremental steps of 2 or 2^2. But for the demonstrations in this user guide, we will explore a smaller search space from -15 to 15.

Now we must decide on which of the three learning kernels that we will use in model development. In this user manual, we will proceed with the RBF kernel and so we will continue by using the default values by clicking on the *Default* button and finally on the *Start* button to initiate the SVM calculation.

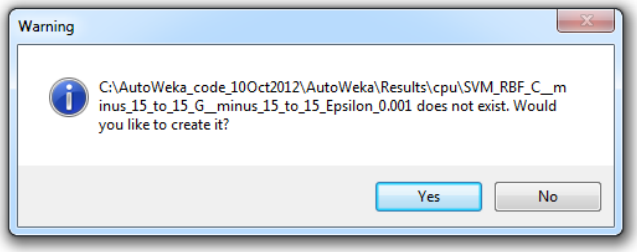
3. **Parameter List.** After all parameters have been entered the program will generate a summary of the parameters that will be used in the forthcoming calculation, to proceed click on the *OK* button:



4. **Target directory.** The program will by default generate the file path for which the constructed models will reside. Users only have to click on the *OK* button to proceed.

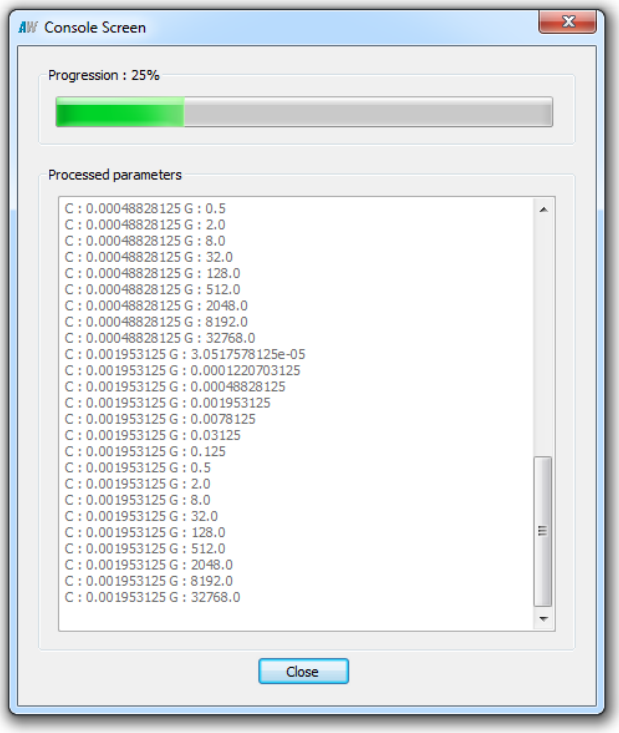


5. **Warning.** The program may issue a warning saying that the folder that it will create does not yet exist and is asking for our confirmation. Users only have to click on the *OK* button to proceed.

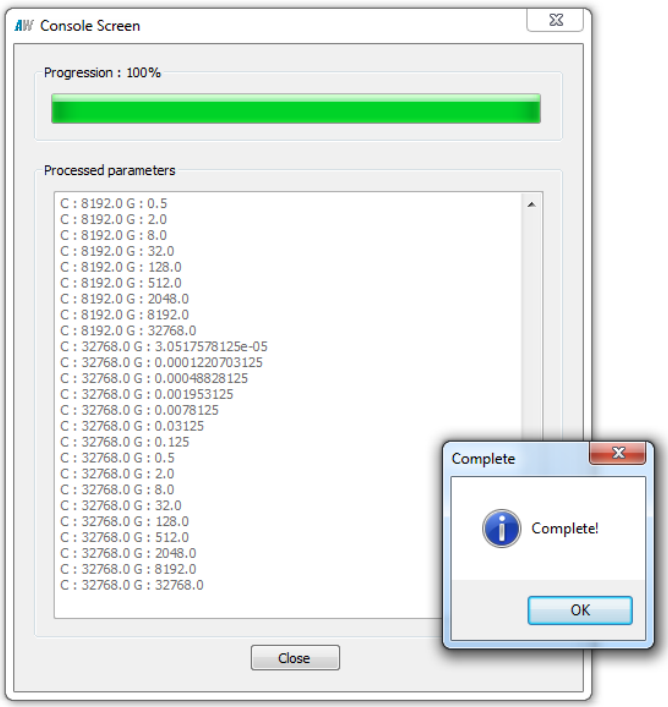


6. **Status of the Calculation.** The calculation will now begin and the progress toward completion will be illustrated by the green status bar located at the top with the

processed and currently calculating parameter shown in the large white box located immediately below the green status bar.



7. **Calculation completed.** When the calculation has completed it will produce a pop-up message as shown below:



8. **Retrieving the results.** Once the calculation has completed, the results can be obtained from the *Results* folder. If the data set is called ABC.arff then a sub-folder called *ABC* should reside in the Results folder; double clicking on this folder should produce the following folder named *SVM_RBF_C__minus_15_to_15_G__minus_15_to_15_Epsilon_0* and inside this folder there will be 1 sub-folder called *C_G*.

8.1. **Data of prediction results.** A text file with the full data of the prediction results is provided in the *parse.txt* file. The contents of this file can be copied and directly pasted into Microsoft Excel or a graphical plotting software for further analysis and plot creation.

Here, we provide an example of the contents of *parse.txt*:

C	G	Training_Correlation	Training_RMS	Testing_Correlation	Testing_RMS
-15.0	-15.0	0.87	163.7413	-0.0413	163.7
-15.0	-13.0	0.8707	163.7412	-0.0413	163.7
-15.0	-11.0	0.8707	163.7409	-0.0411	163.6997
-15.0	-9.0	0.8705	163.7398	-0.0405	163.6987
-15.0	-7.0	0.8695	163.7352	-0.0379	163.6947
-15.0	-5.0	0.8657	163.7173	-0.0279	163.6794
-15.0	-3.0	0.8497	163.6542	0.0055	163.6278
-15.0	-1.0	0.7819	163.4931	0.0824	163.5078
-15.0	1.0	0.6027	163.2011	0.1589	163.3892
-15.0	3.0	0.4842	163.2259	0.1948	163.2906
-15.0	5.0	0.4035	163.4137	0.1519	163.3733
-15.0	7.0	0.3629	163.6497	0.0351	163.5495
-15.0	9.0	0.3917	163.7196	-0.0349	163.6924
-15.0	11.0	0.3918	163.7251	-0.0479	163.737
-15.0	13.0	0.3915	163.726	-0.0482	163.7305
-15.0	15.0	0.3973	163.7266	-0.0549	163.727
-13.0	-15.0	0.8707	163.7412	-0.0413	163.7
-13.0	-13.0	0.8708	163.7409	-0.0411	163.6997
-13.0	-11.0	0.8707	163.7398	-0.0405	163.6987
-13.0	-9.0	0.8705	163.7351	-0.0378	163.6946
-13.0	-7.0	0.8695	163.7167	-0.0274	163.6786
-13.0	-5.0	0.8657	163.6454	0.0125	163.6173
-13.0	-3.0	0.8497	163.3931	0.1558	163.4054
-13.0	-1.0	0.7819	162.7615	0.4053	162.958
-13.0	1.0	0.6034	162.094	0.5186	162.399
-13.0	3.0	0.4827	161.8914	0.44	162.198
-13.0	5.0	0.3991	162.5389	0.3393	162.625
-13.0	7.0	0.36	163.3502	0.2038	163.2687
-13.0	9.0	0.3928	163.6632	0.0103	163.6152
-13.0	11.0	0.3918	163.6765	-0.0342	163.705
-13.0	13.0	0.3915	163.6802	-0.0388	163.7076
-13.0	15.0	0.3973	163.6827	-0.04	163.6819
-11.0	-15.0	0.8708	163.7409	-0.0411	163.6997
-11.0	-13.0	0.8708	163.7398	-0.0405	163.6987
-11.0	-11.0	0.8707	163.7351	-0.0378	163.6946
-11.0	-9.0	0.8705	163.7166	-0.0273	163.6784
-11.0	-7.0	0.8695	163.643	0.0146	163.6142
-11.0	-5.0	0.8657	163.3577	0.183	163.3634
-11.0	-3.0	0.8497	162.3668	0.595	162.5498
-11.0	-1.0	0.7815	160.0087	0.7371	160.6346
-11.0	1.0	0.6163	157.7601	0.6049	158.471
-11.0	3.0	0.4866	156.8006	0.4801	157.8133
-11.0	5.0	0.3994	158.7316	0.3809	159.5936
-11.0	7.0	0.3613	161.7505	0.3221	162.2186
-11.0	9.0	0.3924	163.1266	0.1668	163.39
-11.0	11.0	0.3919	163.1524	0.0475	163.5816
-11.0	13.0	0.3916	163.1664	0.0204	163.6138
-11.0	15.0	0.3978	163.176	0.0152	163.6232
-9.0	-15.0	0.8708	163.7398	-0.0405	163.6987
-9.0	-13.0	0.8708	163.7351	-0.0378	163.6946

-9.0	-11.0	0.8707	163.7165	-0.0273	163.6783
-9.0	-9.0	0.8705	163.6423	0.0152	163.6134
-9.0	-7.0	0.8695	163.3481	0.191	163.351
-9.0	-5.0	0.8657	162.2275	0.6588	162.3772
-9.0	-3.0	0.8497	158.581	0.8488	159.1052
-9.0	-1.0	0.8094	148.9463	0.7748	151.3861
-9.0	1.0	0.6566	140.9688	0.6389	143.6676
-9.0	3.0	0.5323	144.733	0.5151	146.5447
-9.0	5.0	0.4616	151.8362	0.4384	153.3646
-9.0	7.0	0.4071	157.5944	0.3542	158.8431
-9.0	9.0	0.3959	161.5538	0.29	162.6474
-9.0	11.0	0.4037	162.211	0.1667	163.2403
-9.0	13.0	0.4048	162.3603	0.1255	163.3096
-9.0	15.0	0.4155	162.3332	0.1159	163.3331
-7.0	-15.0	0.8708	163.7351	-0.0378	163.6946
-7.0	-13.0	0.8708	163.7165	-0.0272	163.6783
-7.0	-11.0	0.8707	163.6422	0.0153	163.6132
-7.0	-9.0	0.8705	163.3456	0.193	163.3478
-7.0	-7.0	0.8695	162.1892	0.6738	162.3277
-7.0	-5.0	0.8657	157.9663	0.8804	158.4387
-7.0	-3.0	0.8785	143.3052	0.8808	145.4801
-7.0	-1.0	0.8559	117.8939	0.8198	123.7382
-7.0	1.0	0.7432	119.129	0.7013	124.1637
-7.0	3.0	0.6321	130.0417	0.5975	133.8022
-7.0	5.0	0.576	139.0774	0.5276	142.9507
-7.0	7.0	0.5257	148.663	0.4447	152.5699
-7.0	9.0	0.4718	154.7397	0.3495	158.8242
-7.0	11.0	0.4526	157.5889	0.2251	161.5562
-7.0	13.0	0.4528	157.8133	0.1852	161.8727
-7.0	15.0	0.459	158.0521	0.1823	162.0127
-5.0	-15.0	0.8708	163.7165	-0.0272	163.6783
-5.0	-13.0	0.8708	163.6422	0.0153	163.6131
-5.0	-11.0	0.8707	163.345	0.1936	163.347
-5.0	-9.0	0.8705	162.1794	0.6775	162.3148
-5.0	-7.0	0.8693	157.8501	0.889	158.2428
-5.0	-5.0	0.8985	141.632	0.9109	143.1041
-5.0	-3.0	0.9142	106.0177	0.9135	109.0342
-5.0	-1.0	0.9172	85.6708	0.8853	94.4909
-5.0	1.0	0.8423	94.3579	0.7845	104.2951
-5.0	3.0	0.7497	109.9702	0.6885	118.8163
-5.0	5.0	0.6956	120.4272	0.6037	130.1355
-5.0	7.0	0.6335	129.7062	0.5188	140.8012
-5.0	9.0	0.595	138.4927	0.3977	150.9473
-5.0	11.0	0.5613	142.7255	0.2729	155.8876
-5.0	13.0	0.5587	143.7415	0.2321	157.0426
-5.0	15.0	0.5665	144.0329	0.2219	157.5143
-3.0	-15.0	0.8708	163.6421	0.0154	163.6131
-3.0	-13.0	0.8708	163.3448	0.1937	163.3468
-3.0	-11.0	0.8707	162.177	0.6785	162.3115
-3.0	-9.0	0.8702	157.8118	0.8909	158.1896
-3.0	-7.0	0.9028	141.1776	0.9172	142.5019
-3.0	-5.0	0.9259	103.3108	0.9372	103.6739
-3.0	-3.0	0.9407	78.9479	0.9378	82.0011
-3.0	-1.0	0.9674	52.3288	0.924	69.908
-3.0	1.0	0.9452	62.3209	0.8528	88.1302
-3.0	3.0	0.9013	77.6178	0.7821	102.6138
-3.0	5.0	0.862	88.7872	0.7	116.0177
-3.0	7.0	0.8103	99.8032	0.5794	129.7693
-3.0	9.0	0.7758	107.1097	0.4363	141.4615
-3.0	11.0	0.7611	110.778	0.3088	148.3111
-3.0	13.0	0.7565	111.5659	0.2726	149.7989
-3.0	15.0	0.7566	111.664	0.2643	150.1793
-1.0	-15.0	0.8708	163.3448	0.1937	163.3467
-1.0	-13.0	0.8707	162.1764	0.6787	162.3107
-1.0	-11.0	0.8704	157.8022	0.8916	158.1861
-1.0	-9.0	0.9007	140.7086	0.9195	142.3037
-1.0	-7.0	0.9283	102.592	0.9418	102.2288
-1.0	-5.0	0.9387	79.9302	0.941	81.1265
-1.0	-3.0	0.9763	48.2938	0.9714	52.4432
-1.0	-1.0	0.9936	23.3064	0.9532	54.8865
-1.0	1.0	0.9965	14.4254	0.9184	62.5821
-1.0	3.0	0.9929	20.3267	0.8719	77.9327
-1.0	5.0	0.9867	27.0708	0.81	95.4884

-1.0	7.0	0.983	32.5312	0.649	120.9647
-1.0	9.0	0.9764	40.7906	0.46	138.1405
-1.0	11.0	0.9737	43.4544	0.3282	145.9587
-1.0	13.0	0.973	44.039	0.2881	147.8708
-1.0	15.0	0.9729	44.1614	0.2782	148.3197
1.0	-15.0	0.8708	162.1762	0.6788	162.3105
1.0	-13.0	0.8705	157.7998	0.8914	158.1733
1.0	-11.0	0.9025	140.8088	0.9197	142.2989
1.0	-9.0	0.9294	102.8595	0.9422	102.1643
1.0	-7.0	0.9374	80.7051	0.9416	80.9261
1.0	-5.0	0.9564	59.7643	0.9551	60.6986
1.0	-3.0	0.9962	18.9156	0.9864	33.4441
1.0	-1.0	0.9997	3.6861	0.9752	35.8895
1.0	1.0	0.9999	2.027	0.9181	62.2966
1.0	3.0	1	1.4071	0.873	77.1532
1.0	5.0	1	1.4903	0.8175	92.6828
1.0	7.0	1	1.422	0.6657	118.63
1.0	9.0	1	1.3871	0.466	137.4667
1.0	11.0	1	1.4528	0.331	145.7576
1.0	13.0	1	1.3867	0.2888	147.8219
1.0	15.0	1	1.2614	0.2775	148.3357
3.0	-15.0	0.8705	157.7992	0.8917	158.182
3.0	-13.0	0.9024	140.8754	0.92	142.2821
3.0	-11.0	0.929	101.8437	0.9425	102.0057
3.0	-9.0	0.9372	79.9323	0.9415	81.0145
3.0	-7.0	0.9439	64.5514	0.9455	64.9433
3.0	-5.0	0.9833	38.2751	0.9794	39.4662
3.0	-3.0	0.9995	5.1489	0.995	17.6578
3.0	-1.0	0.9999	2.6372	0.9749	36.1494
3.0	1.0	0.9999	1.7391	0.9176	62.4329
3.0	3.0	1	1.3698	0.8727	77.2276
3.0	5.0	1	1.4903	0.8175	92.6828
3.0	7.0	1	1.422	0.6657	118.63
3.0	9.0	1	1.3871	0.466	137.4667
3.0	11.0	1	1.4528	0.331	145.7576
3.0	13.0	1	1.3867	0.2888	147.8219
3.0	15.0	1	1.2614	0.2775	148.3357
5.0	-15.0	0.9008	140.7098	0.9202	142.2694
5.0	-13.0	0.9286	102.3709	0.9433	101.9063
5.0	-11.0	0.9371	79.8722	0.9411	81.0863
5.0	-9.0	0.9396	65.7518	0.9427	66.072
5.0	-7.0	0.9585	54.7228	0.9562	55.3316
5.0	-5.0	0.9988	8.3999	0.9931	20.7315
5.0	-3.0	0.9997	3.5346	0.9965	13.7997
5.0	-1.0	0.9999	2.0837	0.9746	36.4224
5.0	1.0	0.9999	1.5606	0.9184	62.2583
5.0	3.0	1	1.3698	0.8727	77.2276
5.0	5.0	1	1.4903	0.8175	92.6828
5.0	7.0	1	1.422	0.6657	118.63
5.0	9.0	1	1.3871	0.466	137.4667
5.0	11.0	1	1.4528	0.331	145.7576
5.0	13.0	1	1.3867	0.2888	147.8219
5.0	15.0	1	1.2614	0.2775	148.3357
7.0	-15.0	0.9302	102.3546	0.9432	101.897
7.0	-13.0	0.9375	79.7675	0.9404	81.2234
7.0	-11.0	0.9391	66.8557	0.9403	67.1429
7.0	-9.0	0.9441	61.285	0.943	61.4659
7.0	-7.0	0.9851	36.1361	0.9805	37.0665
7.0	-5.0	0.9996	4.6553	0.9985	8.4274
7.0	-3.0	0.9998	2.7157	0.9961	14.7164
7.0	-1.0	0.9999	1.851	0.9768	34.7957
7.0	1.0	1	1.3799	0.9187	62.2295
7.0	3.0	1	1.3698	0.8727	77.2276
7.0	5.0	1	1.4903	0.8175	92.6828
7.0	7.0	1	1.422	0.6657	118.63
7.0	9.0	1	1.3871	0.466	137.4667
7.0	11.0	1	1.4528	0.331	145.7576
7.0	13.0	1	1.3867	0.2888	147.8219
7.0	15.0	1	1.2614	0.2775	148.3357
9.0	-15.0	0.937	79.9472	0.94	81.265
9.0	-13.0	0.9389	66.0918	0.9413	66.5439
9.0	-11.0	0.9412	62.1225	0.9386	63.4435
9.0	-9.0	0.9588	54.2352	0.9573	54.2771

9.0	-7.0	0.9991	6.8103	0.9936	18.4797
9.0	-5.0	0.9997	3.851	0.9989	7.5587
9.0	-3.0	0.9999	2.2919	0.9959	15.3045
9.0	-1.0	0.9999	1.6734	0.9828	30.565
9.0	1.0	1	1.3523	0.9188	62.1997
9.0	3.0	1	1.3698	0.8727	77.2276
9.0	5.0	1	1.4903	0.8175	92.6828
9.0	7.0	1	1.422	0.6657	118.63
9.0	9.0	1	1.3871	0.466	137.4667
9.0	11.0	1	1.4528	0.331	145.7576
9.0	13.0	1	1.3867	0.2888	147.8182
9.0	15.0	1	1.2614	0.2775	148.3357
11.0	-15.0	0.9383	66.5937	0.9416	66.6025
11.0	-13.0	0.9399	62.5926	0.938	63.7047
11.0	-11.0	0.9448	60.2431	0.9435	60.9174
11.0	-9.0	0.9857	35.559	0.9809	36.293
11.0	-7.0	0.9996	4.5184	0.9987	7.8953
11.0	-5.0	0.9998	3.0468	0.9987	8.2951
11.0	-3.0	0.9999	2.0849	0.9954	15.6186
11.0	-1.0	1	1.5462	0.9844	29.6898
11.0	1.0	1	1.3523	0.9188	62.1997
11.0	3.0	1	1.3698	0.8727	77.2276
11.0	5.0	1	1.4903	0.8175	92.6828
11.0	7.0	1	1.422	0.6657	118.63
11.0	9.0	1	1.3871	0.466	137.4667
11.0	11.0	1	1.4528	0.331	145.7576
11.0	13.0	1	1.3867	0.2888	147.8182
11.0	15.0	1	1.2614	0.2775	148.3357
13.0	-15.0	0.94	62.6686	0.9369	63.8563
13.0	-13.0	0.9402	61.9408	0.9382	63.1207
13.0	-11.0	0.9589	54.3367	0.9575	54.1302
13.0	-9.0	0.9991	6.8271	0.9935	18.2593
13.0	-7.0	0.9997	4.0463	0.999	6.9526
13.0	-5.0	0.9998	2.7418	0.9984	9.2845
13.0	-3.0	0.9999	1.8768	0.9953	16.1575
13.0	-1.0	1	1.4264	0.9926	20.6268
13.0	1.0	1	1.3523	0.9189	62.1869
13.0	3.0	1	1.3698	0.8727	77.2276
13.0	5.0	1	1.4903	0.8175	92.6828
13.0	7.0	1	1.422	0.6657	118.63
13.0	9.0	1	1.3871	0.466	137.4667
13.0	11.0	1	1.4528	0.331	145.7576
13.0	13.0	1	1.3867	0.2888	147.8182
13.0	15.0	1	1.2614	0.2775	148.3357
15.0	-15.0	0.9395	62.2252	0.9369	63.5102
15.0	-13.0	0.9446	60.5632	0.9434	60.8107
15.0	-11.0	0.9862	34.8241	0.9809	36.0963
15.0	-9.0	0.9996	4.5212	0.9986	8.4151
15.0	-7.0	0.9997	3.7079	0.999	6.983
15.0	-5.0	0.9999	2.3453	0.9987	8.3386
15.0	-3.0	0.9999	1.6946	0.9986	9.2143
15.0	-1.0	1	1.3907	0.9931	19.5405
15.0	1.0	1	1.3922	0.919	62.1702
15.0	3.0	1	1.3698	0.8728	77.22
15.0	5.0	1	1.4903	0.8175	92.6828
15.0	7.0	1	1.422	0.6657	118.6302
15.0	9.0	1	1.3871	0.466	137.4667
15.0	11.0	1	1.4528	0.3311	145.7562
15.0	13.0	1	1.3867	0.2888	147.8182
15.0	15.0	1	1.2614	0.2775	148.3357

8.2. **Summary of prediction results.** A summary text file is provided by the *Summary.txt* file.

8.3.

Here, we provide an example of the contents of *Summary.txt*:

```
Method
---------------------------------------
weka.classifiers.functions.SMOreg, RBF

Data set
---------------------------------------
C:\AutoWeka_code_10Oct2012\AutoWeka\tmpARFF\cpu.arff

Cross validation
=======================================


Opt G: -7.0 : 0.0078125

Epsilon : 0.001
Optimal C = 13.0 : 8192.0


RMS :6.9526
Correlation :0.999


<RMS>
Min : 6.9526
Max : 163.737

<Correlation>
Min : -0.0549
Max : 0.999
```

8.4. **Raw and parsed data.** The above files are results that have been parsed and post-processed from several raw data files to be in a format that is ready for further analysis.

Contents of the raw data of each investigated parameters generate results as obtained from a typical Weka calculation as follows (here the contents of *0001_C-15.0_G-15.0..txt* are shown, *C-15.0* denotes a C parameter with a value of -15 while *G-15.0* denotes the Gamma parameter with a value of -15):

```
Options: -S 0.001 -C 3.0517578125e-05 -R -G 3.0517578125e-05

SMOreg

Kernel used :
  RBF kernel : K(x,y) = e^-(3.0517578125E-5* <x-y,x-y>^2)

Support Vector Expansion :
(normalized) class =
 +       0.0243
```

```
Number of support vectors: 0

Number of kernel evaluations: 21945 (100    % cached)

Time taken to build model: 0.03 seconds
Time taken to test model on training data: 0.02 seconds

=== Error on training data ===

Correlation coefficient                  0.87
Mean absolute error                     70.9553
Root mean squared error                163.7413
Relative absolute error                 81.2006 %
Root relative squared error            106.0594 %
Total Number of Instances              209



=== Cross-validation ===

Correlation coefficient                 -0.0413
Mean absolute error                     71.4297
Root mean squared error                163.7
Relative absolute error                 81.4865 %
Root relative squared error            105.774  %
Total Number of Instances              209
```

9. **Plotting graphs of the results.** Now that we have obtained the necessary data, we can go ahead and make some plots, which will be useful in visually assisting us in identifying the best set of learning parameters. More information is provided in section 4.

4. Plotting graphs

It is often said that a picture is worth a thousand words, therefore plots and graphs are invaluable in allowing us to readily discern the inherent insights, patterns and trends that are contained within these data. This section will detail the procedures that are needed to create plots and graphs using the results from the ANN (AvgHidden.txt, AvgTrainingTime.txt and AvgLearningAndMomentum.txt) and SVM (parsed.txt) calculations. The text files of the results can be obtained from the folders previously mentioned in sections 3.2.8 and 3.3.8.

It should be noted that the artificial neural network calculations performed a 3 step parameter optimization comprising of: (1) number of nodes in the hidden layer, (2) number of learning epochs and (3) learning rate and momentum. Let us proceed with creating plots of these data. Furthermore, support vector machine calculation optimizes: (1) only the C parameter in a linear kernel, (2) C and exponential parameters for the polynomial kernel and the (3) C and Gamma parameters for the RBF kernel.

Users can use a third party software of their choice to create plots and graphs but here we will describe the Python scripts that we provide on our website that users can download and use in their plot making efforts. The latter approach is fairly straightforward and best of all, it is also free.

1. **Installing Python and Matplotlib module.** Before we can create plots and graphs we should first install Python (version 2.X as our scripts was written on this platform) by downloading it from http://www.python.org/download/. Then proceed to installing the Matplotlib module by downloading it from http://matplotlib.org/.

2. **Consolidating the results text files and script files into the same folder.** For files from the artificial neural network calculations, copy the following text files from the following respective sub-folders of the *Results* folder as shown below on the left and right, respectively:

```
AvgHidden.txt               C:\AutoWeka\Results\<data set name>\HiddenNode
AvgTrainingTime.txt         C:\AutoWeka \Results\<data set name>\TrainingTime
AvgLearningAndMomentum.txt  C:\AutoWeka \Results\<data set name>\LearningAndMomentum
```

   and paste them into the same working folder of your choice (this folder should already contain the Python scripts (*.py files) extracted from the *autoweka-plots-and-graph-scripts.zip* file provided from the AutoWeka website at that users can download from http://www.mt.mahidol.ac.th/autoweka/download.html.

   For files from the support vector machine calculations, copy the following text file from the following respective sub-folder of the *Results* folder as shown below on the left and right, respectively:

```
parsed.txt                  C:\AutoWeka\Results\<data set name>\C_G
```

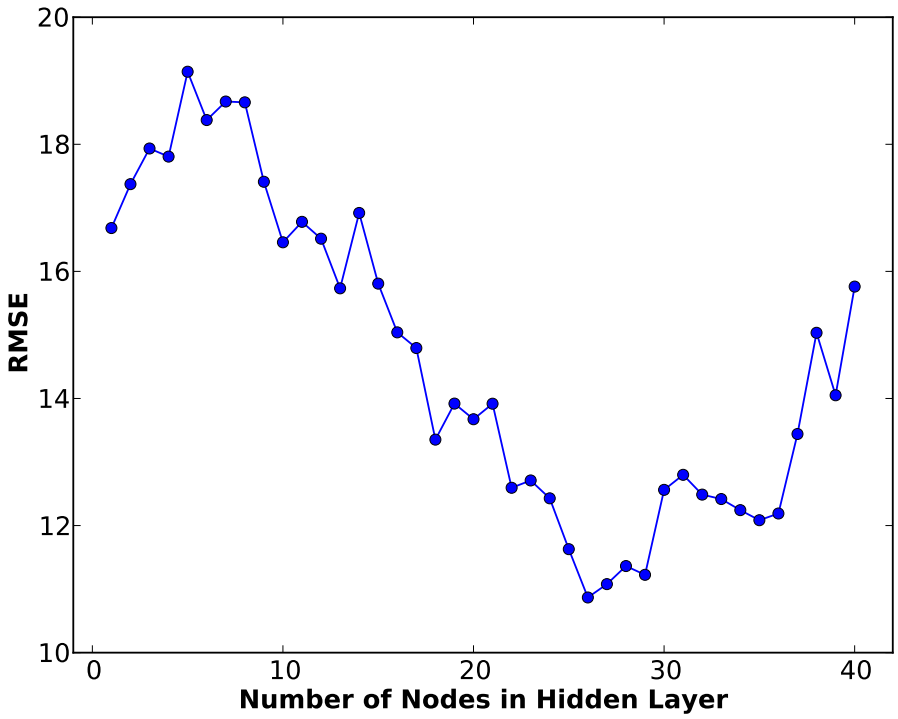
3. **Running the scripts to create the plots and graphs.** Now that the necessary input data and script files are consolidated into the same folder, we can now proceed with executing the Python scripts, which will generate the plots and graphs as PNG and PDF files.

   3.1. **Run the command prompt.** Initiate the MS Windows command prompt by clicking on Start → Run buttons, which will bring up a small pop-up window. Then type *cmd* into the text box and press the *Enter* button of the keyboard. (On MS Windows 7, users need only click on the *Start* button then click on the white text box and type *cmd* into it followed by pressing the *Enter* button of the keyboard.

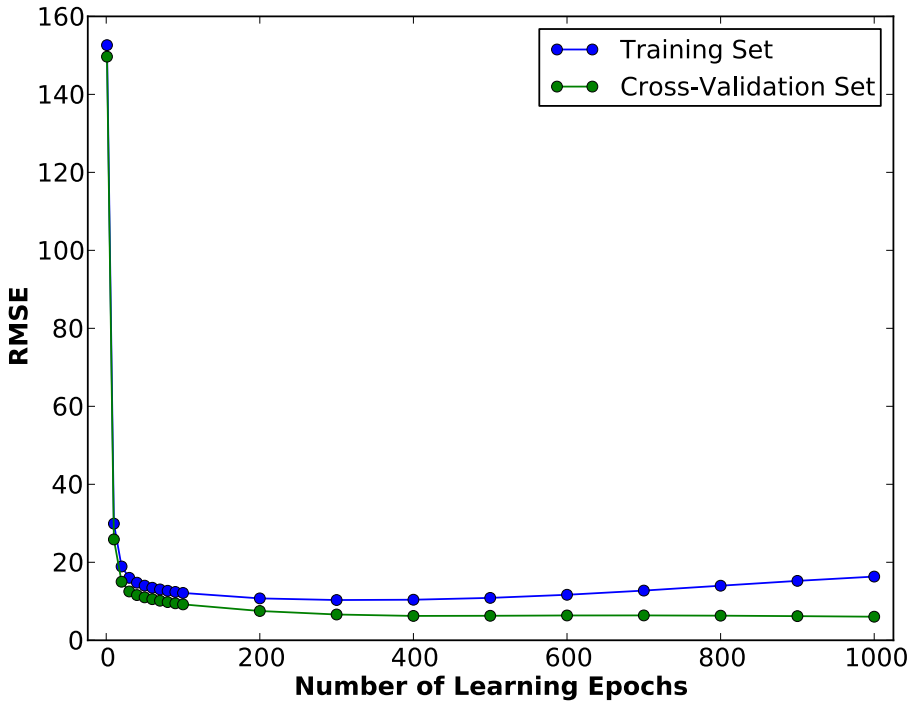
   3.2. **Change to working directory.** In this command prompt users will need to change the current directory (which may be at *C:\Users\<User Name>*) to the user's working directory (the folder where all input data and script files are located) by invoking the command cd *C:\path\to\working\directory*.

   3.3. **Running the scripts.** Users can type the following commands to create the respective plots and graphs (as shown below the commands) for results from ANN (the first 3 plots) and SVM (the last plot) calculations:

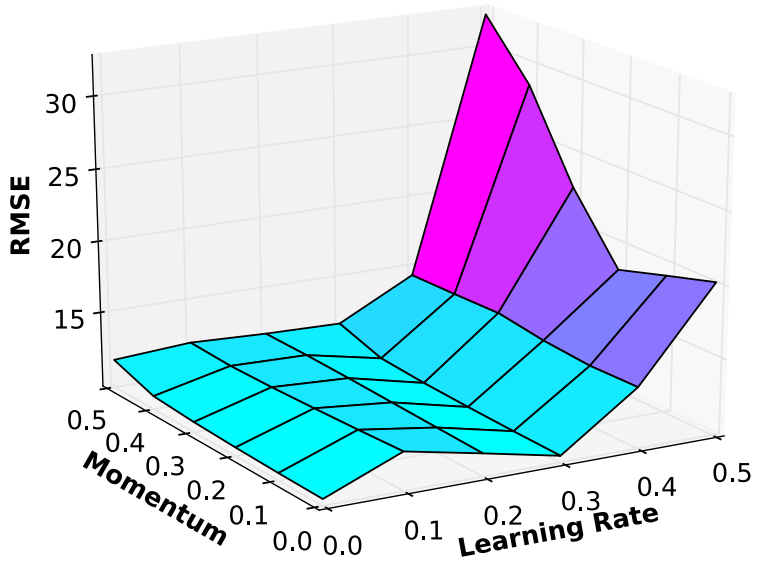
```
python AvgHidden.py AvgHidden.txt
```



```
python AvgTrainingTime.py AvgTrainingTime.txt
```




```
python AvgLearningAndMomentum.py AvgLearningAndMomentum.txt
```



```
python parse.py parse.txt
```