Parallel and Distributed Computing

Artificial Neural Networks

H20 Neural Network Project Raport

Filip Mazur - 226018

Wrocław University
of Science and Technology

Applied Computer Science
Department of Computer Science and Management
Wroclaw University of Science and Technology

2020-06-14

Contents

1. Introduction

This project concerns the preparation of a Python program which will use H2O server to train and test deep feedforward neural network with three hidden layers, each with 5, 20, 100 neurons, to solve a classification problem from UCI ML repository - Votes. To be more specific - based on how a given person votes on a given set of acts posing as attributes this Neural Network tries to extrapolate whether this person is a democrat or a republican.

1.1 Project description

This project is written in Python 3 under Linux OS using PyCharm IDE and H2O's H2ODeepLearningEstimator. To gather the results a graphical dashboard/interface H2O Flow was used. In order to run this project:

1. Download the latest release of H2O. [1]
2. Using your terminal unzip and run H2O by using following commands:
   cd /Downloads
   unzip h2o-3.30.0.2.zip
   cd h2o-3.30.0.2
   java -jar h2o.jar
3. Open your browser and go to http://localhost:54321 in order to open up the H2O Flow.
4. Run the script.
5. Open up jobs menu in H2O Flow.
6. If you wish to see the detailed results on a particular job click on it's title and then "View".

1.2 Data-set description

Chosen data-set for the project is called 'Congressional Voting Records Data Set' and was downloaded from UCI ML repository [2]. It consists of 435 entries and 17 attributes, 16 of which are enum attributes corresponding to the type of vote (for: 'y', against: 'n') a given individual gave for a given act and the remaining attribute is of enum type as well but informs whether a given individual is a 'republican' or a 'democrat'. Obviously the ML task for this data-set is of classification type. This data-set contains missing values. Before this data-set was loaded into the script some preprocessing was done on it. Firstly the place of an attribute containing the political connotation information was moved from first to last place in the table as the author is used for target attribute to be always the rightmost one. Secondly all the 'y' values of remaining attributes were changed to '1' and 'n' values to '-1' in order to hopefully make it easier for the model to learn. All of the missing values were left as-is.

[1] `http://h2o-release.s3.amazonaws.com/h2o/rel-zahradnik/4/index.html`
[2] `https://archive.ics.uci.edu/ml/datasets/Congressional+Voting+Records`

2. Results

2.1 Measurement method description

In this project two different types of neuron activation methods were used:

- tangent
- ReLU

As well as three different validation methods:

- no cross-validation (for this option binomial metrics for the most desired threshold was used in order to determine accuracy)
- 5-fold cross-validation
- 10-fold cross-validation

The measurements being performed in this project are:

1. The measurement of running time of each job (performed automatically by H2O and taken straight from Flow).
2. The measurement of accuracy as we change both the validation and neuron activation settings.

2.2 Measurement results

2.2.1 Tangent neurons and no cross-validation

Run Time 00:00:06.187

layer	units	type	dropout	l1	l2	mean_rate	rate_rms	momentum	mean_weight	weight_rms	mean_bias	bias_rms
1	16	Input	0									
2	5	Tanh	0	0	0	0.0015	0.0005	0	0.0123	0.2994	0.0285	0.0271
3	20	Tanh	0	0	0	0.0041	0.0035	0	-0.0055	0.2579	-0.0050	0.0380
4	100	Tanh	0	0	0	0.0296	0.0785	0	-0.0017	0.1271	0.0012	0.0214
5	2	Softmax		0	0	0.0119	0.0084	0	0.0085	0.5494	-0.0021	0.0315

Figure 2.1: Status of the neuron layers

threshold	f1	f2	f0point5	accuracy	precision	recall	specificity
1.0	0.1053	0.0685	0.2273	0.6277	1.0	0.0556	1.0
1.0	0.1379	0.0909	0.2857	0.6350	1.0	0.0741	1.0
0.9999	0.2000	0.1351	0.3846	0.6496	1.0	0.1111	1.0

Figure 2.2: Validation summary

2.2.2 Tangent neurons and 5-fold cross-validation

Run Time 00:00:20.406

layer	units	type	dropout	l1	l2	mean_rate	rate_rms	momentum	mean_weight	weight_rms	mean_bias	bias_rms
1	16	Input	0									
2	5	Tanh	0	0	0	0.3460	0.3733	0	-0.0368	0.3440	0.0382	0.0690
3	20	Tanh	0	0	0	0.6139	0.2819	0	0.0458	0.3528	-0.0119	0.1000
4	100	Tanh	0	0	0	0.9991	0.0076	0	0.0024	0.1849	-0.0255	0.1680
5	2	Softmax		0	0	0.7162	0.2741	0	0.0142	0.7048	0.0093	0.3032

Figure 2.3: Status of the neuron layers

	mean	sd	cv_1_valid	cv_2_valid	cv_3_valid	cv_4_valid	cv_5_valid
accuracy	0.95559037	0.037300546	0.97619045	0.9655172	0.92957747	0.9066667	1.0
auc	0.98631555	0.0117016435	0.9909091	0.99254656	0.974359	0.9737631	1.0
aucpr	0.9803665	0.018235294	0.99131566	0.9886055	0.95966864	0.96224296	1.0
err	0.04440963	0.037300546	0.023809524	0.03448276	0.07042254	0.093333334	0.0
err_count	3.0	2.9154758	1.0	2.0	5.0	7.0	0.0
f0point5	0.94091165	0.047066912	0.9649123	0.93495935	0.9322034	0.8724832	1.0
f1	0.94308525	0.051288974	0.9777778	0.9583333	0.8979592	0.88135594	1.0
f2	0.9460899	0.0627887	0.990991	0.982906	0.86614174	0.89041096	1.0
lift_top_group	2.6924183	0.6518623	1.9090909	2.5217392	2.7307692	2.586207	3.7142856
logloss	0.1961403	0.14572915	0.18455705	0.14806244	0.3992576	0.2488225	1.9709644E-6
max_per_class_error	0.07288746	0.058250718	0.05	0.057142857	0.15384616	0.10344828	0.0
mcc	0.90751994	0.07950775	0.9532553	0.93135846	0.8482025	0.8047833	1.0
mean_per_class_accuracy	0.9526384	0.041950073	0.975	0.9714286	0.9119658	0.9047976	1.0
mean_per_class_error	0.047361605	0.041950073	0.025	0.028571429	0.08803419	0.0952024	0.0
mse	0.050042223	0.034377083	0.042629458	0.042597182	0.08429869	0.08068577	1.1917457E-10
pr_auc	0.9803665	0.018235294	0.99131566	0.9886055	0.95966864	0.96224296	1.0
precision	0.939942	0.049808875	0.95652175	0.92	0.95652175	0.8666667	1.0
r2	0.78953165	0.14754704	0.82909465	0.8219914	0.6367951	0.659777	1.0
recall	0.9485411	0.07268098	1.0	1.0	0.84615386	0.8965517	1.0
rmse	0.19745311	0.11755046	0.20646903	0.20639084	0.2903424	0.2840524	1.0916711E-5
specificity	0.9567357	0.033397466	0.95	0.94285715	0.9777778	0.9130435	1.0

Figure 2.4: Cross-validation summary

2.2.3 Tangent neurons and 10-fold cross-validation

Run Time 00:00:39.704

layer	units	type	dropout	l1	l2	mean_rate	rate_rms	momentum	mean_weight	weight_rms	mean_bias	bias_rms
1	16	Input	0									
2	5	Tanh	0	0	0	0.6362	0.2268	0	0.0088	0.3348	-0.0135	0.0333
3	20	Tanh	0	0	0	0.8121	0.1988	0	0.0146	0.3169	0.0048	0.0615
4	100	Tanh	0	0	0	1.0036	0.0012	0	0.0039	0.1844	0.0206	0.1367
5	2	Softmax		0	0	0.8863	0.1850	0	0.0304	0.7069	0.0188	0.3669

Figure 2.5: Status of the neuron layers

	mean	sd	cv_1_valid	cv_2_valid	cv_3_valid	cv_4_valid	cv_5_valid	cv_6_valid	cv_7_valid	cv_8_valid	cv_9_valid	cv_10_valid
accuracy	0.96310186	0.034769177	0.9285714	1.0	0.93877554	0.96428573	0.92105263	0.9583333	1.0	0.92	1.0	1.0
auc	0.9799669	0.026454518	0.96428573	1.0	0.9862069	0.99444443	0.93538463	0.986014	1.0	0.93333334	1.0	1.0
aucpr	0.9617394	0.07442819	0.96297526	1.0	0.98046595	0.990469	0.75723374	0.98544675	1.0	0.94080347	1.0	1.0
err	0.03689814	0.034769177	0.071428575	0.0	0.06122449	0.035714287	0.078947365	0.041666668	0.0	0.08	0.0	0.0
err_count	1.2	1.2292726	2.0	0.0	3.0	1.0	3.0	1.0	0.0	2.0	0.0	0.0
f0point5	0.9497312	0.05185103	0.9285714	1.0	0.89285713	0.9259259	0.8695652	0.98039216	1.0	0.9	1.0	1.0
f1	0.9552455	0.043231536	0.9285714	1.0	0.9302326	0.95238096	0.8888889	0.95238096	1.0	0.9	1.0	1.0
f2	0.96148545	0.04116707	0.9285714	1.0	0.9708738	0.98039216	0.90909094	0.9259259	1.0	0.9	1.0	1.0
lift_top_group	2.4098485	1.0077612	2.0	2.6666667	2.45	2.8	0.0	2.1818182	4.0	2.5	2.5	3.0
logloss	0.21160321	0.16549031	0.2985492	3.128298E-7	0.20574659	0.21183428	0.39612627	0.19413805	0.044597913	0.5104426	0.24635617	0.008240607
max_per_class_error	0.05013415	0.04524027	0.071428575	0.0	0.10344828	0.055555556	0.08	0.09090909	0.0	0.1	0.0	0.0
mcc	0.9247972	0.07188025	0.85714287	1.0	0.8829554	0.9265991	0.8291617	0.91877955	1.0	0.8333333	1.0	1.0
mean_per_class_accuracy	0.964182	0.034779496	0.9285714	1.0	0.94827586	0.9722222	0.9215385	0.95454544	1.0	0.9166667	1.0	1.0
mean_per_class_error	0.035817992	0.034779496	0.071428575	0.0	0.05172414	0.027777778	0.078461535	0.045454547	0.0	0.083333336	0.0	0.0
mse	0.05136771	0.03512064	0.07720383	2.310869E-12	0.05544093	0.063791275	0.0881796	0.057176255	0.007587121	0.082611956	0.080344446	0.0013416893
pr_auc	0.9617394	0.07442819	0.96297526	1.0	0.98046595	0.990469	0.75723374	0.98544675	1.0	0.94080347	1.0	1.0
precision	0.94643706	0.05971971	0.9285714	1.0	0.8695652	0.90909094	0.85714287	1.0	1.0	0.9	1.0	1.0
r2	0.7836251	0.14750716	0.69118464	1.0	0.7704937	0.72215354	0.6082112	0.76969564	0.95953536	0.65578353	0.66523147	0.9939624
recall	0.96607393	0.04443957	0.9285714	1.0	1.0	1.0	0.9230769	0.90909094	1.0	0.9	1.0	1.0
rmse	0.19965588	0.11306459	0.27785578	1.5201542E-6	0.23545897	0.25256935	0.29695052	0.23911557	0.08710408	0.28742296	0.28345096	0.036629077
specificity	0.9622901	0.04151851	0.9285714	1.0	0.8965517	0.9444444	0.92	1.0	1.0	0.93333334	1.0	1.0

Figure 2.6: Cross-validation summary

2.2.4 ReLU neurons and no cross-validation

Run Time 00:00:05.500

layer	units	type	dropout	l1	l2	mean_rate	rate_rms	momentum	mean_weight	weight_rms	mean_bias	bias_rms
1	16	Input	0									
2	5	Rectifier	0	0	0	0.0019	0.0007	0	0.0431	0.3062	0.5060	0.0255
3	20	Rectifier	0	0	0	0.0030	0.0030	0	-0.0496	0.2831	0.9960	0.0232
4	100	Rectifier	0	0	0	0.0137	0.0583	0	-0.0009	0.1249	0.9992	0.0144
5	2	Softmax		0	0	0.0031	0.0044	0	0.0303	0.5450	0.0	0.0126

Figure 2.7: Status of the neuron layers

threshold	f1	f2	f0point5	accuracy	precision	recall	specificity	absolute_mcc
1.0	0.0364	0.0230	0.0862	0.6131	1.0	0.0185	1.0	0.1063
1.0	0.0714	0.0459	0.1613	0.6204	1.0	0.0370	1.0	0.1509
1.0	0.1053	0.0685	0.2273	0.6277	1.0	0.0556	1.0	0.1855

Figure 2.8: Validation summary

2.2.5 ReLU neurons and 5-fold cross-validation

Run Time 00:00:14.491

layer	units	type	dropout	l1	l2	mean_rate	rate_rms	momentum	mean_weight	weight_rms	mean_bias	bias_rms
1	16	Input	0									
2	5	Rectifier	0	0	0	0.5202	0.4310	0	-0.0031	0.3503	0.4808	0.0794
3	20	Rectifier	0	0	0	0.6240	0.3897	0	-0.0252	0.3325	0.9644	0.0538
4	100	Rectifier	0	0	0	0.8082	0.2276	0	-0.0010	0.1618	0.9983	0.0131
5	2	Softmax		0	0	0.6490	0.2082	0	0.0548	0.6068	0.0061	0.0060

Figure 2.9: Status of the neuron layers

	mean	sd	cv_1_valid	cv_2_valid	cv_3_valid	cv_4_valid	cv_5_valid
accuracy	0.95729226	0.03478513	0.9848485	0.93333334	1.0	0.9516129	0.9166667
auc	0.98491395	0.011625168	0.99039614	0.9762931	1.0	0.98722416	0.9706564
aucpr	0.97570634	0.01625057	0.98078966	0.9560053	1.0	0.97250503	0.9692317
err	0.042707723	0.03478513	0.015151516	0.06666667	0.0	0.048387095	0.083333336
err_count	2.6	2.302173	1.0	3.0	0.0	3.0	6.0
f0point5	0.931259	0.058594383	0.9876543	0.8695652	1.0	0.8974359	0.90163934
f1	0.94679654	0.03708456	0.969697	0.9142857	1.0	0.93333334	0.9166667
f2	0.96413237	0.02504778	0.95238096	0.96385545	1.0	0.9722222	0.9322034
lift_top_group	2.7648754	0.7421139	3.8823528	2.8125	2.12	2.952381	2.057143
logloss	0.19311838	0.108964965	0.098926544	0.29671347	0.07632407	0.18169063	0.31193718
max_per_class_error	0.068710126	0.043582287	0.05882353	0.10344828	0.0	0.07317073	0.10810811
mcc	0.912917	0.067152835	0.96039206	0.86890215	1.0	0.90054184	0.83474904
mean_per_class_accuracy	0.95993066	0.030323809	0.9705882	0.94827586	1.0	0.9634146	0.9173745
mean_per_class_error	0.04006935	0.030323809	0.029411765	0.05172414	0.0	0.036585364	0.082625486
mse	0.05350192	0.03291129	0.02783404	0.081843935	0.018207345	0.04643748	0.093186796
pr_auc	0.97570634	0.01625057	0.98078966	0.9560053	1.0	0.97250503	0.9692317
precision	0.9217994	0.07359767	1.0	0.84210527	1.0	0.875	0.8918919
r2	0.7687681	0.13124616	0.8544477	0.64281476	0.9269365	0.7926763	0.626965
recall	0.9768067	0.03176426	0.9411765	1.0	1.0	1.0	0.94285715
rmse	0.22172248	0.07366356	0.16683537	0.2860838	0.13493459	0.21549357	0.30526513
specificity	0.94305456	0.053686496	1.0	0.8965517	1.0	0.9268293	0.8918919

Figure 2.10: Cross-validation summary

2.2.6 ReLU neurons and 10-fold cross-validation

Run Time 00:00:32.116

layer	units	type	dropout	l1	l2	mean_rate	rate_rms	momentum	mean_weight	weight_rms	mean_bias	bias_rms
1	16	Input	0									
2	5	Rectifier	0	0	0	0.0315	0.0218	0	0.0116	0.3683	0.5610	0.1052
3	20	Rectifier	0	0	0	0.2472	0.2977	0	0.0284	0.3207	0.9844	0.0601
4	100	Rectifier	0	0	0	0.5976	0.3332	0	-0.0063	0.1892	0.9983	0.0295
5	2	Softmax		0	0	0.4732	0.2915	0	0.0073	0.6876	0.0033	0.0325

Figure 2.11: Status of the neuron layers

	mean	sd	cv_1_valid	cv_2_valid	cv_3_valid	cv_4_valid	cv_5_valid	cv_6_valid	cv_7_valid	cv_8_valid	cv_9_valid	cv_10_valid
accuracy	0.97318864	0.02825514	0.96875	1.0	0.9142857	0.95	0.975	0.9655172	0.9583333	1.0	1.0	1.0
auc	0.98453724	0.021246236	0.9916667	1.0	0.95424837	0.94	0.9946667	0.987013	0.9777778	1.0	1.0	1.0
aucpr	0.97659814	0.028457876	0.9860788	1.0	0.94011444	0.92115426	0.9919167	0.9663477	0.96036947	1.0	1.0	1.0
err	0.02681137	0.02825514	0.03125	0.0	0.08571429	0.05	0.025	0.03448276	0.041666668	0.0	0.0	0.0
err_count	0.8	0.9189366	1.0	0.0	3.0	1.0	1.0	1.0	1.0	0.0	0.0	0.0
f0point5	0.96117395	0.044128507	0.9375	1.0	0.87628865	0.9259259	0.9859155	0.9677419	0.9183673	1.0	1.0	1.0
f1	0.96672624	0.03205307	0.96	1.0	0.9189189	0.95238096	0.9655172	0.9230769	0.94736844	1.0	1.0	1.0
f2	0.97364676	0.036675345	0.9836066	1.0	0.96590906	0.98039216	0.9459459	0.88235295	0.9782609	1.0	1.0	1.0
lift_top_group	2.7008	0.5879584	2.6666667	2.642857	2.0588236	2.0	2.6666667	4.142857	2.6666667	2.5384614	2.625	3.0
logloss	0.17702185	0.19154903	0.16908742	4.048696E-4	0.55866784	0.34372997	0.1448648	0.12023307	0.38390198	0.005270419	6.401464E-4	0.04341797
max_per_class_error	0.059285715	0.061848264	0.05	0.0	0.16666667	0.1	0.06666667	0.14285715	0.06666667	0.0	0.0	0.0
mcc	0.94519174	0.054646466	0.9364417	1.0	0.8416254	0.90453404	0.94733095	0.9054699	0.9165151	1.0	1.0	1.0
mean_per_class_accuracy	0.9703571	0.030924132	0.975	1.0	0.9166667	0.95	0.96666664	0.9285714	0.96666664	1.0	1.0	1.0
mean_per_class_error	0.029642858	0.030924132	0.025	0.0	0.083333336	0.05	0.033333335	0.071428575	0.033333335	0.0	0.0	0.0
mse	0.037504368	0.037686437	0.03159984	5.7653606E-6	0.103313714	0.09899042	0.04141022	0.040418975	0.04165841	5.3224876E-4	8.489487E-6	0.017105587
pr_auc	0.97659814	0.028457876	0.9860788	1.0	0.94011444	0.92115426	0.9919167	0.9663477	0.96036947	1.0	1.0	1.0
precision	0.9582168	0.056982696	0.9230769	1.0	0.85	0.90909094	1.0	1.0	0.9	1.0	1.0	1.0
r2	0.8401199	0.15197486	0.86517406	0.9999755	0.58640754	0.6040383	0.8233164	0.7792704	0.82225746	0.99777067	0.999964	0.92302483
recall	0.9790476	0.047682498	1.0	1.0	1.0	1.0	0.93333334	0.85714287	1.0	1.0	1.0	1.0
rmse	0.15816326	0.117798075	0.17776343	0.0024011165	0.3214245	0.31462744	0.20349501	0.20104471	0.20410392	0.023070518	0.0029136725	0.13078833
specificity	0.96166664	0.05776175	0.95	1.0	0.8333333	0.9	1.0	1.0	0.93333334	1.0	1.0	1.0

Figure 2.12: Cross-validation summary

3. Conclusions

From the results we can clearly see that models using 10-fold cross-validation perform better than those using 5-fold cross-validation which in turn perform than models with no cross-validation. If we think about our data-set we'll notice that it has very little entries. One of the most basic notions behind the cross-validation method is that it's supposed to make our situation better and training more efficient if we have insufficient number of entries in our data in order to make an accurate prediction. Therefore we observed during this experiment that cross-validation performs exactly as expected.

As for the comparison of our two neuron activation methods we can observe that ReLU performs by a tiny margin better if cross-validation is used (especially if it's 10-fold) and tangent performs a tiny margin better if we don't use cross-validation (but still pretty badly). As for execution time, ReLU appears to be on average 27.23 % faster than tangent according to this experiment.

If we consider all those results we can come to a conclusion that the choice of our validation method and neuron activation method when designing a ML model for our Neural Network is not a trivial task and is dependant mostly on our data-set and on what exactly we want to do with it.