

EE6222 Assignment 1 Report

LI RIXUAN G1801134G

1. Used data set:

Datasets	Patterns	Features	Classes
ctg-3classes	2126	21	3
ctg-10classes	2126	21	10
contrac	1473	9	3
molec-biol-splice	3190	60	3
st-image	2310	18	7
titanic	2201	3	2

2. Experiment Result about Question 1,2,4,5:

In question 1&2, we only need to run the program with different selected data sets because there already exist four options of test accuracy represent whether we have bias and link or not. Bias=1 means bias exist, Bias=0 means bias doesn't exist, same to the link.

In question 4, I choose 2 activation functions: radbas&hardlim. In order to compare their performance, the 'option.ActivationFunction' in program RVFL_train_val should manually change between radbas and hardlim. In question 5, two solution algorithm: Regularized Least Square & Moore-Penrose pseudoinverse are manually changed by changing 'option.mode' in program RVFL_train_val to 1 or 2 respectively.

For every data set, we should set different variables and run the program 4 times to get 4 accuracy each time. So for 6 data sets, I run the program 24 times to get following 6 tables.

TABLE 1.1 Ctg_3classes

Solution & Activation Function		Bias = 0 Link = 0	Bias = 1 Link = 0	Bias = 0 Link = 1	Bias = 1 Link = 1
Regularized Least Square	Radbas	0.9063	0.9063	0.9077	0.9077 ¹
	Hardlim	0.8569	0.8569	0.8837	0.8837
Moore-Penrose pseudoinverse	Radbas	0.9068	0.9068	0.9110	0.9091
	Hardlim	0.8545	0.8545	0.8856	0.8856

TABLE 1.2 Ctg_10classes

Solution &Activation Function		Bias = 0 Link = 0	Bias = 1 Link = 0	Bias = 0 Link = 1	Bias = 1 Link = 1
Regularized Least Square	Radbas	0.7947	0.7966	0.7966 ²	0.7985
	Hardlim	0.5753	0.5753	0.6822	0.6822
Moore-Penrose pseudoinverse	Radbas	0.7863	0.7863	0.7952	0.7947
	Hardlim	0.5702	0.5702	0.6883	0.6883

TABLE 1.3 Contrac

Solution &Activation Function		Bias = 0 Link = 0	Bias = 1 Link = 0	Bias = 0 Link = 1	Bias = 1 Link = 1
Regularized Least Square	Radbas	0.4980	0.4980 ³	0.5292	0.5292
	Hardlim	0.4561	0.4561	0.4786	0.4786
Moore-Penrose pseudoinverse	Radbas	0.5231	0.5211	0.5462	0.5394
	Hardlim	0.5023	0.5023	0.5254	0.5254

TABLE 1.4 Molec-biol-splice

Solution &Activation Function		Bias = 0 Link = 0	Bias = 1 Link = 0	Bias = 0 Link = 1	Bias = 1 Link = 1
Regularized Least Square	Radbas	0.8152 ⁴	0.8168	0.8187	0.8184
	Hardlim	0.5950	0.5950	0.8011	0.8011
Moore-Penrose pseudoinverse	Radbas	0.8071	0.8118	0.8134	0.8156
	Hardlim	0.5891	0.5891	0.8033	0.8033

TABLE 1.5 St-image

Solution &Activation Function		Bias = 0 Link = 0	Bias = 1 Link = 0	Bias = 0 Link = 1	Bias = 1 Link = 1
Regularized Least Square	Radbas	0.9419	0.9419	0.9419	0.9424 ⁵
	Hardlim	0.7621	0.7621	0.8934	0.8934
Moore-Penrose pseudoinverse	Radbas	0.9450	0.9445	0.9471	0.9458
	Hardlim	0.7335	0.7335	0.8938	0.8938

TABLE 1.6 Titanic

Solution &Activation Function		Bias = 0 Link = 0	Bias = 1 Link = 0	Bias = 0 Link = 1	Bias = 1 Link = 1
Regularized Least Square	Radbas	0.7882	0.7882	0.7895 ⁶	0.7895
	Hardlim	0.7118	0.7118	0.7136	0.7136
Moore-Penrose pseudoinverse	Radbas	0.7892	0.7892	0.7951	0.7951
	Hardlim	0.7021	0.7021	0.7056	0.7056

3. Question 1 Conclusion:

Compare columns with bias=1&link=1 to columns with bias=1&link=0, and also compare columns with bias=0&link=1 to columns with bias=0&link=0 in all above 6 tables. It's very clear that all accuracy from link=1 groups are higher than link=0 groups. Except in St-image'case, we got two 0.9419 with and without link.

So it is obvious that direct link has big improvement to accuracy.

4. Question 2 Conclusion:

Compare columns with bias=1&link=0 to columns with bias=0&link=0, and also compare columns with bias=1&link=1 to columns with bias=0&link=1 in all above 6 tables. In almost all cases the accuracy from bias=1 groups are exactly the same to bias=0 groups, or slightly bigger.

Conclusion is that bias term having almost no effect in accuracy.

5. Question 4 Conclusion:

Compare rows 2 to 3 and 4 to 5 in each table above. We could easily observe that all accuracy in radbas groups are significantly higher than hardlim groups without exception.

That means activation function "radbas" enhance the performance a lot compare to "hardlim".

6. Question 5 Conclusion:

Compare rows 2 to 4 and 3 to 5 in each table above. In more than half cases the accuracies from regularized least square group are higher than Moore-Penrose pseudoinverse group. In other cases, Moore-Penrose pseudoinverse group performs a litter bit well than regularized least square group.

So the regularized least square method performs a certain extent better than Moore-Penrose pseudoinverse method for the computation of the output weights.

7. Question 3 Experiment Result and Conclusion:

In table1.1-1.6, I choose one case in each table with corner mark 1-6. They are all in regularized least square method and radbas group. I manually set their program's scaling factor from 2^{-5} to 2^5 with step size 2^1 . Then run the program 11*6 times to get corresponding accuracy for different cases and different scaling factors shown in table 6.1 below.

TABLE 6.1 Changing Scaling Factors

Method	$S=2^{-5}$	$S=2^{-4}$	$S=2^{-3}$	$S=2^{-2}$	$S=2^{-1}$	$S=2^0$	$S=2^1$	$S=2^2$	$S=2^3$	$S=2^4$	$S=2^5$
Case1	0.8801	0.8924	0.9081	0.9085	0.9108	0.9110	0.9102	0.9098	0.9098	0.9013	0.9008
Case2	0.7824	0.7951	0.7953	0.8044	0.8051	0.8062	0.8085	0.8078	0.8042	0.8014	0.7969
Case3	0.4823	0.4989	0.5001	0.5032	0.5056	0.5054	0.5032	0.5014	0.4980	0.4913	0.4895
Case4	0.8095	0.8124	0.8156	0.8195	0.8220	0.8221	0.8242	0.8253	0.8256	0.8251	0.8224
Case5	0.9152	0.9324	0.9523	0.9592	0.9615	0.9622	0.9623	0.9621	0.9613	0.9442	0.9402
Case6	0.7562	0.7912	0.7993	0.8045	0.8069	0.8073	0.8101	0.8102	0.8095	0.8091	0.8056

As we can observe that different scaling values may lead to relatively large change in accuracy. The number in red is the highest accuracy in each row. It's obvious that the when scaling value is more smaller or more bigger than the best scaling value, the corresponding accuracy is more smaller in almost all case.