Pattern Recognition: Linear Classification

Kevin Mato – K5529

December 2019

1 Initial preparation with two classes

The perceptron function was first tested for its correctness with a ovo strategy. The perceptron was asked to divide digits, for example 3 and 4. The function was set up to refine the separation plane over 200 iterations and with a learning rate of *sqrt(1/i)*, where *i* is the number of iterations. Other learning rates were tested but with almost identical results. It's noticeable that not putting the learning rate causes spikes in the graph of misclassified samples per iterations, while with a learning rate the graph results smoother (Figure 1,2). If the learning process is stopped at the right iteration the results are similar with the two approaches.

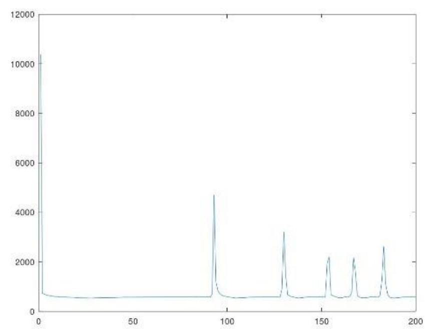

Figure 2: No learning rate

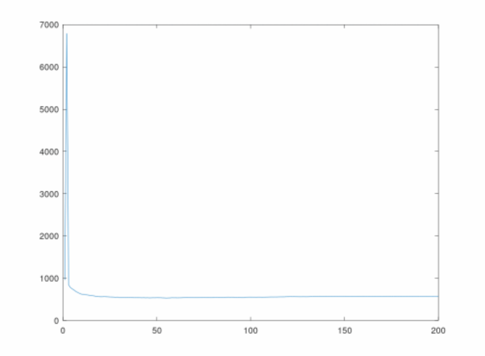

Figure 1: Learning rate sqrt(1/i)

Results from the Classification of digits 3 and 4 (Perceptron, 200 i, sqrt(1/i))				
Size of digit 4 set	**Size of digit 3 set**	**False Positive**	**False Negative**	**Error rate**
6131	5842	359	216	4.8%

Results from the classification of digits 3 and 4.

After seeing that the false positives were greater than the false positives, I checked if it was possible that the class of 4 digits was greater than class 3. Actually this is true and their difference in the train set is of 173 samples. The difference between false positive and false negative is of 143 samples. What if we balance class 4, could we drop the error rate?

Moreover from a little exploration, due to its smaller size compared to the train set, we could expect a kind of overfitting of the classifier on the test set, it means we can expect better accuracy in some cases on the test set. On the next image we see the distribution of the classes on train and test set, in blue the trainset, in orange test. The test set is smaller than the training set, and it seems to be very much like it but with less noise (for exceptions of class 3).

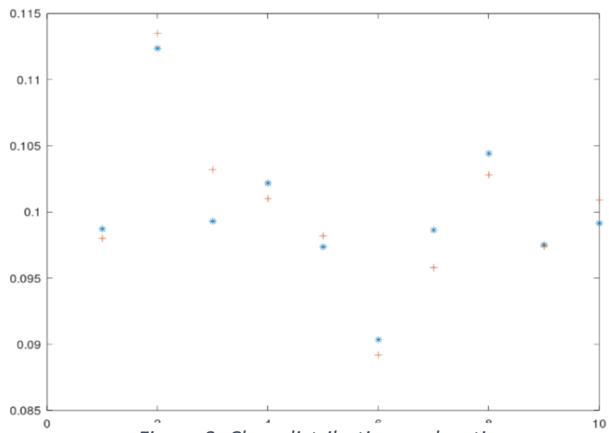

Figure 3: Class distribution exploration

2 One vs One ensemble with multidimensional digits data

After checking the validity of our functions we tried to classify the digits with a One vs One ensemble classifier, in our case with unanimity voting. The individual error rates of the one vs. one classifiers are shown below (Table c). The least error is found in the classifier 6-7 with 0.3%, and it's highlighted in green. The worst error rate is given by classifier 3-5 with almost 5.2%. In a second run, the pair 0-1 is performing better, and very similarly the couple 0-7. Even though it may be redundant keeping all these decimal numbers, I preferred keeping them to see minimal differences and to have a vision of how close were the error rates. The number of ovo classifiers in this case are 45 (*[classes*(classes-1)]/2*) and the errors achieved were computed using only 40 primary components. Although the number of false positives and false negatives are always a concern for our analysis, they are not shown here in this table. For curiosity sake the confusion matrixes are in the appendix of this report.

Couple		Error Rate	
0	1	0.003316225819	
0	2	0.01169935191	
0	3	0.01037000166	
0	4	0.004504887378	
0	5	0.01965796897	
0	6	0.01047208851	
0	7	0.004922874959	
0	8	0.01214540513	
0	9	0.00951819407	
1	2	0.01543307087	
1	3	0.01336129884	
1	4	0.004609027336	
1	5	0.008468305517	
1	6	0.004502369668	
1	7	0.007534404551	

Couple		Error Rate
1	8	0.02128166442
1	9	0.005909699787
2	3	0.03250889238
2	4	0.01915254237
2	5	0.0268916425
2	6	0.02745031997
2	7	0.02086230876
2	8	0.03497332543
2	9	0.01587301587
3	4	0.006431136724
3	5	0.05159279778
3	6	0.01070628268
3	7	0.01508551146
3	8	0.04139542647
3	9	0.02127483444

Couple		Error Rate
4	5	0.0113646453
4	6	0.01105442177
4	7	0.01478483522
4	8	0.01342683657
4	9	0.04155711984
5	6	0.02610459476
5	7	0.008813965429
5	8	0.04834989354
5	9	0.01961301671
6	7	0.003037018797
6	8	0.01520944855
6	9	0.004044830201
7	8	0.01502145923
7	9	0.05067954806
8	9	0.02338983051

Table c: Single ovo classifiers with related error rate

In table.a we can see the results of the ensemble on the train set, the coefficient for correct answers is slightly superior than 90%, while the error rate is 6.25% and the rejection rate is 2.7%. After assessing the quality of the classifier with the test set we even get more correct results (91.3%) at the price of less rejections (2.6%) and less misclassifications (6.01%), although the error coefficient is still higher than the rejection. In the confusion matrix based on the classification of the train set, there are highlighted the couples which cannot be misclassified, like couples 0-1, 3-4, 6-9. The confusion matrix for the test set presents even more great results of this type.

A second experiment is performed in order to achieve better error coefficients for the ovo ensemble. From the 40 features we had in the beginning we obtain 860 by combining them with each other. Increasing the feature space could be one way to complete our objective of having less errors and rejection rate higher than the error one. The runtime for the ovo classifiers training has increased. This factor is not negligible and it must be taken into account when looking for the best solution. The best error is the one of the couple 0-1 with 0,1% against the best (0.3%) with 40 features, and the worst is the couple 7-9 (0.1%) against 5%. The worst case has improved dramatically for the single OVOs.

With extended features of training set:

Couple		Error Rate
0	1	0.001026450849
0	2	0.005723423954
0	3	0.005475360876
0	4	0.002719932002
0	5	0.004055007052
0	6	0.007938518706
0	7	0.003446012471

Couple		Error Rate
1	8	0.01135551497
1	9	0.004806555827
2	3	0.00802382331
2	4	0.003220338983
2	5	0.002812197908
2	6	0.002526103065
2	7	0.007526793749

Couple		Error Rate
4	5	0.002308443576
4	6	0.004931972789
4	7	0.005533988602
4	8	0.003933977593
4	9	0.01060130608
5	6	0.007055295881
5	7	0.002652746877

0	8	0.005180907083	2	8	0.005588957575	5	8	0.005677785664
0	9	0.00496967655	2	9	0.004535147392	5	9	0.004925241865
1	2	0.009448818898	3	4	0.002589158941	6	7	0.001395387015
1	3	0.006292239571	3	5	0.01021468144	6	8	0.004758263234
1	4	0.005403687222	3	6	0.002904805378	6	9	0.002190949692
1	5	0.002877579545	3	7	0.00588899645	7	8	0.005199735886
1	6	0.004186413902	3	8	0.01001502253	7	9	0.01138038317
1	7	0.003613438918	3	9	0.008609271523	8	9	0.006949152542

Table.a	OVO		
	OK.	Error.	Rejection.
Training set	0.90978333333	0.06255000000	0.02766666667
Testing set	0.91380000000*	0.06010000000	0.02610000000
Training set extended	0.968016666667	0.024000000000	0.007983333333
Testing set extended	0.96020000000	0.02930000000	0.01050000000

*testing set performing better than train as said in the beginning

With extended feature set we can see that the correctness of the ovo ensemble is higher (96%), the error rate has decreased of 3/3.5% percent, and the rejection rate for the training set has decreased of 2%, for the test set extended it dropped of 2.5%. All this may seem possible but we haven't reached error rate < rejections rate.

In the next step we will check if with one vs rest classifiers we can solve this problem.

3 One vs Rest ensemble with multidimensional digits data

To perform the classification with OVR, new code was developed, to train the different classifiers and a new voting system was written.

With in mind code reuse the new functions are:

-trainOVRensemble

-OVRclassifier, instead of unamvoting, where the clab is assigned based on the results of voting

-OVRvoting, to perform the voting

-rds_ trainOVRensemble, this one does the training of the classifiers with more balanced negative class (rest).

The single error rates for the ovr classifiers seem good and in the same range of ovo, but if we take a look at table.b we can see a high rejection rate for both training and test set in ensemble, this is due to the voting method which is more sensible than unanimity, since it's rejecting every form of draw. Now we have more rejections than errors but we are not satisfied since the rejections go up to almost 20%. When printing the number of false positives and false negatives, the first ones are much greater in number due to the cardinality of the "rest" set of sample, this may have impacted on the single clf coefficient.

A second experiment with ovr was made. We tried to reduce the number of negative samples during the training, in order to move the separating planes further from the positive class towards the negative one (rest). After many trials where the error rate of the ensemble was increasing quickly and the correct classifications were drecreasing (in realation to the reduction coefficient of the negative samples), the best result is achieved by keeping 80% of the negative samples. The result of this experiment unfortunately is worse than expected.

Label	Error Rate	Error Rate with expanded features	Error Rate with balanced negative classes (80% of original)	Error Rate with balanced negative classes (80% of original , w. extended features)
0	0.0111	0.005466666667	0.01266569082	0.005773765959
1	0.01021666667	0.00595	0.01199546118	0.005916680175
2	0.02746666667	0.009883333333	0.03179184877	0.01099705255
3	0.03145	0.01173333333	0.03589274832	0.01350802356
4	0.0248	0.01398333333	0.02769869031	0.01142926869
5	0.04678333333	0.01126666667	0.05398468057	0.01197848761
6	0.01631666667	0.006716666667	0.01829751764	0.006953056702
7	0.01891666667	0.008716666667	0.02202732607	0.009907221309
8	0.04955	0.0129	0.0565746126	0.01189653069
9	0.04471666667	0.01741666667	0.05021039579	0.01677067876

The last experiment is trying to improve the OVR by expanding the features space during the training. As for ovo, we have a magnificent improvement, the error rate of the ensemble has decreased. The rejection rate is still high but these are samples that can be classified in a second attempt. After balancing the negatives classes we got the best result for OVR. Thanks to this we could say that we achieved our objective, but this result isn't as correct as the best ovo, there is still room for improvement.

Table.b	OVR		
	OK.	Error.	Rejection.
Training set	0.7668666667	0.03723333333	0.1959
Testing set	0.7761	0.0354	0.1885
Training set extended	0.9148666667	0.01681666667	0.06831666667
Testing set extended	0.9148	0.0192	0.066
Training set Balanced	0.7621333333	0.03936666667	0.1985
Testing set Balanced	0.771	0.0377	0.1913
Training set Balanced EXT	0.9205	0.01661666667	0.06288333333
Testing set Balanced EXT	0.9168	0.0181	0.0651

4 Improvement

The first idea was to check which classes had more misclassified samples with ovr, and trying to classifiy them with the ovo classifiers of that class. This solution is improving the correctness of classification but since we achieved good results problem, we should try to reduce the number of rejected samples.

The improvement is thought also in terms of code reuse. There are two phases, in the first one the samples are classified using an ovr ensemble and in the second the rejections of ovr are recycled and there is an attempt to classify them with them best ovo ensemble. It would be natural to think that were used ovr/ovo with expanded features but since this is computationally expensive there was an attempt with also the sets with 40 features. We didn't try the ovr/ovo with reduced negative samples since the benefit was minor.

The results are pretty delusional with not expanded features, as the error rate is bigger than the ovo with almost the same correctness. On the contrary with expanded features there's a surprise. The rejection drops from 6% to more or less 0.5%, 4% goes to correctness and 1% circa goes to error, probably due to the high error of the ovo ensemble.

	OK.	Error.	Rejection.
Training set	0.9061	0.07293333333	0.02096666667
Testing set	0.9076	0.0712	0.0212
Training set extended	0.9675833333	0.0274	0.005016666667
Testing set extended	0.9607	0.0319	0.0074

Conclusion

Comparison between standards and improvement			
	OK.	Error.	Rejection.
OVR OVO	0.9675833333	0.0274	0.005016666667
OVO	0.968016666667	0.024000000000	0.007983333333
OVR	0.9205	0.01661666667	0.06288333333

Here we have the best of each experiment. The improvement isn't as exceptional as we expected. OVO and the improvement have similar results, we just gained 0.2% from the rejected to add to the errors, OVO remains the best.

Results of OVO ensemble with unanimity voting on training set											
Result / Actual	0	1	2	3	4	5	6	7	8	9	Rejections
0	**5659**	0	24	13	8	63	34	4	16	7	95
1	0	**6474**	49	18	8	16	1	26	66	7	77
2	22	29	**5351**	59	50	22	67	54	77	15	212
3	11	22	77	**5429**	0	190	22	28	103	36	213
4	9	12	40	1	**5398**	5	20	24	30	183	120
5	30	12	24	179	20	**4672**	67	8	126	37	246
6	32	14	79	3	24	94	**5550**	1	25	0	96
7	11	16	65	27	40	8	1	**5789**	13	148	147
8	17	71	56	130	20	120	32	21	**5069**	45	270
9	19	19	26	59	178	24	2	200	42	**5196**	184

Results of OVR OVO ensemble on training set EXTENDED											
Result / Actual	0	1	2	3	4	5	6	7	8	9	Rejections
0	**5814**	0	10	5	4	8	23	4	16	7	32
1	0	**6609**	37	9	16	4	4	13	23	3	24
2	19	21	**5759**	30	11	2	6	42	27	11	30
3	2	6	53	**5866**	0	55	6	37	41	27	38
4	6	22	8	4	**5676**	2	22	12	8	60	22
5	15	3	7	46	9	**5230**	37	7	19	16	32
6	28	17	6	3	14	27	**5790**	0	12	1	20
7	5	31	32	15	16	2	0	**6074**	14	51	25
8	9	42	21	50	10	37	22	19	**5580**	22	39
9	12	19	8	40	58	15	3	75	23	**5657**	39

Appendix. Confusion Matrixes

In this first part more colour in the confusion matrix means more positive results.

Results of OVO ensemble with unanimity voting on testing set											
Result / Actual	**0**	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	Rejections
0	944	0	1	0	0	7	7	1	3	0	17
1	0	1102	3	3	0	3	0	1	11	0	12
2	3	1	942	8	5	3	9	9	17	1	34
3	3	0	9	918	0	24	0	10	12	4	30
4	1	0	5	1	911	0	6	5	4	24	25
5	8	1	3	37	4	764	9	3	24	6	33
6	10	2	10	1	8	14	895	0	3	0	15
7	0	5	20	10	5	0	0	932	6	22	28
8	4	3	2	28	4	24	6	5	854	5	39
9	8	5	3	8	32	5	1	33	10	876	28

Results of OVO ensemble with unanimity voting on training set EXTENDED											
Result / Actual	**0**	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	Rejections
0	5787	0	12	3	0	11	35	2	14	9	50
1	0	6607	34	10	9	1	4	9	25	5	38
2	20	41	5754	27	6	4	9	34	13	7	43
3	6	19	31	5881	0	45	6	28	32	22	61
4	4	14	8	1	5667	1	28	10	7	51	51
5	14	10	4	37	6	5239	27	11	17	18	38
6	31	17	3	3	10	29	5781	0	10	2	32
7	5	17	28	4	21	2	0	6093	5	44	46
8	9	55	15	37	11	29	17	16	5576	24	62
9	11	13	4	32	49	9	4	55	18	5696	58

Results of OVO ensemble with unanimity voting on testing set EXTENDED											
Result / Actual	**0**	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	Rejections
0	961	0	0	2	0	3	5	1	0	3	5
1	0	1115	3	1	2	0	3	1	6	0	4
2	4	1	991	5	4	1	3	8	3	2	10
3	0	0	4	969	0	9	0	9	5	5	9
4	2	1	3	1	943	0	8	1	1	9	13
5	2	2	0	11	0	844	4	2	5	5	17
6	7	3	1	0	6	5	925	0	3	1	7
7	1	2	7	3	2	0	0	980	3	13	17
8	5	2	4	12	4	5	1	7	920	2	12
9	3	4	3	7	13	2	1	9	2	954	11

==

Note: I kept colouring the ovr confusion matrix with the same patterns from ovo and from ovr train set just to see if there was a match between the different classifiers. For example when the train and test set are reduced we can notice which digits are becoming the get misclassified for others. I am aware that keeping the whole matrix is useless, since in one vs rest we are not interested in which class the positive samples is not misclassified, but for me it was still interesting, maybe adding some summary information on the false pos and neg would be useful, but it can be displayed in the matlab code.

Results of OVR ensemble on training set											
Result / Actual	**0**	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	Rejections
0	5215	0	5	4	4	18	7	1	3	0	666
1	1	6016	26	9	3	23	1	9	68	2	584

2	11	22	4519	41	40	12	25	31	68	5	1184
3	17	8	64	4566	0	133	11	20	49	44	1219
4	5	10	9	2	4454	5	10	4	41	78	1224
5	24	15	18	110	51	3320	43	7	55	38	1740
6	16	6	17	1	22	47	5026	0	15	3	765
7	18	11	47	10	20	15	2	5072	10	80	980
8	17	62	18	69	2	56	19	5	3813	21	1769
9	17	16	14	50	80	24	0	87	27	4011	1623

Results of OVR ensemble on testing set

Result / Actual	**0**	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	Rejections
0	865	0	0	0	0	4	2	1	0	0	108
1	0	1034	1	1	0	0	1	1	9	0	88
2	1	3	764	6	5	1	3	9	19	1	220
3	3	0	4	783	0	16	1	4	9	5	185
4	1	0	0	2	773	0	2	1	6	18	179
5	6	2	3	26	6	565	6	2	12	2	262
6	9	2	3	1	3	8	826	0	3	0	103
7	2	1	21	1	2	0	0	815	0	8	178
8	8	3	4	12	5	4	4	2	645	2	285
9	8	5	3	8	32	5	1	33	10	876	28

Results of OVR ensemble on training set EXTENDED

Result / Actual	**0**	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	Rejections
0	5653	0	8	1	3	5	9	1	11	3	229
1	0	6476	26	5	6	4	0	6	14	1	204
2	18	5	5465	14	5	0	5	26	17	3	400
3	0	4	34	5543	0	40	1	21	22	13	453
4	4	10	8	1	5234	0	14	7	11	40	513
5	13	0	6	18	7	4854	20	4	15	14	470
6	18	13	2	0	10	21	5566	0	5	0	283
7	5	15	19	4	11	1	0	5822	7	37	344
8	3	19	11	25	8	23	10	6	5168	21	557
9	11	8	5	32	41	10	4	57	24	5111	646

Results of OVR ensemble on testing set EXTENDED

Result / Actual	**0**	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	Rejections
0	941	0	0	0	0	0	1	1	1	0	36
1	0	1098	2	1	0	0	1	0	4	0	29
2	6	0	943	5	1	0	1	6	8	0	62
3	0	0	2	919	0	5	0	4	2	1	77
4	1	0	1	0	869	0	5	0	3	9	94
5	3	0	1	4	1	802	4	0	1	1	75
6	7	4	0	0	5	5	895	0	1	1	40
7	0	6	12	2	0	0	0	932	2	10	64
8	2	0	2	8	2	5	0	3	862	1	89
9	2	5	1	4	7	0	2	4	3	887	94

Results of OVR ensemble on training set reduced

Result / Actual	**0**	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	Rejections
0	5167	0	5	3	2	13	8	2	4	2	717
1	2	5958	29	10	4	24	1	7	67	4	636
2	12	16	4464	40	47	12	26	29	78	6	1228
3	16	10	58	4525	1	133	11	21	51	41	1264
4	4	11	14	3	4416	4	10	8	40	71	1261
5	26	13	21	125	56	3314	47	7	61	41	1710
6	13	9	21	1	23	44	4974	1	17	2	813
7	18	8	59	7	19	18	1	5020	10	79	1026
8	17	67	22	86	3	59	21	8	3882	24	1662

9	16	12	20	57	95	29	0	86	33	4008	1593

Results of OVR ensemble on testing set reduced											
Actual \ Result	0	1	2	3	4	5	6	7	8	9	Rejections
0	859	0	0	1	1	3	2	1	0	0	113
1	0	1018	2	1	0	0	1	1	11	0	101
2	2	2	759	8	5	1	4	8	21	0	222
3	2	0	4	779	0	14	2	4	10	6	189
4	1	0	1	1	762	1	3	1	9	15	188
5	6	2	3	27	5	567	7	2	12	2	259
6	7	1	2	1	3	8	816	1	5	0	114
7	4	2	24	0	1	1	0	812	3	8	173
8	7	1	3	13	6	6	6	1	651	2	278
9	3	3	4	2	18	6	0	8	2	687	276

Results of OVR ensemble on training set reduced extended											
Actual \ Result	0	1	2	3	4	5	6	7	8	9	Rejections
0	5666	0	4	2	1	6	19	1	8	4	212
1	0	6510	24	9	8	1	2	9	8	3	168
2	19	17	5458	16	7	1	2	18	23	5	392
3	1	1	36	5520	0	41	4	19	25	20	464
4	4	9	3	1	5337	1	18	8	7	35	419
5	10	2	8	21	6	4855	17	9	10	15	468
6	16	10	2	0	6	21	5615	0	6	2	240
7	6	21	19	4	14	1	0	5843	6	33	318
8	7	18	7	16	4	19	6	6	5184	15	569
9	10	5	6	41	41	10	3	50	18	5242	523

Results of OVR ensemble on testing set reduced extended											
Actual \ Result	0	1	2	3	4	5	6	7	8	9	Rejections
0	937	0	0	1	0	0	1	1	0	0	40
1	0	1096	2	1	2	0	3	0	2	0	29
2	6	0	948	3	2	0	1	7	5	0	60
3	0	1	2	913	0	4	0	6	2	1	81
4	1	0	2	1	889	0	4	1	2	9	73
5	2	1	1	4	1	807	4	0	1	1	70
6	6	3	0	0	4	2	897	0	0	0	46
7	1	7	10	2	1	0	0	935	0	9	63
8	4	0	3	7	0	4	0	3	852	0	101
9	3	4	1	3	7	0	2	6	1	894	88

Results of OVR OVO ensemble on training set											
Actual \ Result	0	1	2	3	4	5	6	7	8	9	Rejections
0	5672	1	23	15	5	76	32	2	14	3	80
1	1	6475	39	24	6	29	2	16	89	6	55
2	34	25	5330	69	78	23	67	59	100	22	151
3	22	15	94	5381	1	241	21	41	111	57	147
4	6	17	36	4	5390	10	29	24	58	177	91
5	59	24	29	201	55	4569	76	17	128	57	206
6	29	9	73	3	38	92	5563	0	30	2	79
7	25	21	82	29	44	21	2	5770	15	162	94
8	22	92	69	149	11	133	35	23	5042	57	218
9	17	18	35	80	186	37	3	204	58	5174	137

Results of OVR OVO ensemble on testing set											
Actual \ Result	0	1	2	3	4	5	6	7	8	9	Rejections
0	943	0	4	1	0	8	6	3	1	0	14
1	0	1097	4	2	0	4	2	1	18	0	7

2	9	3	923	11	8	2	8	12	29	1	26
3	1	0	10	914	0	33	2	8	14	7	21
4	1	0	4	1	904	1	6	4	8	33	20
5	9	2	4	48	7	759	7	5	25	4	22
6	12	3	8	1	8	17	891	1	5	0	12
7	2	4	27	7	6	0	0	937	5	23	17
8	7	6	6	26	5	26	7	6	845	5	35
9	5	5	4	8	35	9	1	30	11	863	38

Results of OVO OVR ensemble on testing set EXTENDED											
Result / **Actual**	**0**	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	Rejections
0	964	0	1	0	0	2	6	1	1	0	5
1	0	1115	3	1	1	0	3	0	6	0	6
2	6	0	989	6	5	0	3	11	8	1	3
3	0	0	5	962	0	11	1	10	4	4	13
4	2	2	5	0	947	0	4	2	2	12	6
5	5	1	1	11	2	847	6	1	5	4	9
6	9	3	0	0	8	7	926	0	2	1	2
7	1	5	11	4	2	0	0	981	3	14	7
8	5	1	4	13	3	8	1	6	919	2	12
9	3	6	2	7	12	1	1	7	2	957	11