Bayesian Decision Rule Report

Pattern Recognition Software

Professor. Haralick; Fall 2010

Joel Kemp

Table of Contents

Introduction

Purpose:

The purpose of this software is to analyze the expected gains of a Bayesian decision rule under varying parameters. The system contained the following varying parameters:

Parameter	Description
ND	The number of measurements in measurement space.
NC	The number of classes.
fd	Dominant fraction
Nd	Number of dominant measurements

A number of experiments were run to analyze the expected gain for both a testing set and training set that contained prior and conditional probabilities generated through subroutines in the software. The probabilities were generated using pseudorandom number generators provided by the Python language, using an **initial seed** value of **100**. This seed number was chosen at random and fixed so that subsequent runs of the software would yield identical results, within sampling error.

Technical Details:

This software was developed using Python 2.7, Numpy Scientific Library, Pylab, and Matplotlib; under a Microsoft Windows 7 environment.

The tests/experiments discussed in the following sections were run on a laptop computer with an Intel dual-core 2.2 Ghz processor with 2GB of RAM. The specifications of the machine are given only as supporting information as to the environment under which these tests were run.

Plot 1 - Testing Set Expected Gain

Parameter	Values
ND	[10, 50, 100, 200, 500]
NC	[2, 5, 10]
fd	[0.7, 0.75, 0.8, 0.85, 0.9, 0.95, 0.99]
Nd	[0.8, 0.9, 0.95, 0.97, 0.99]

The first set of plots is wireframe surface plots of the testing set expected gain for each combination of ND and NC as a function of the 35 combinations of *fd* and *Nd*. It should be stated that the computation of the expected gain is achieved in a succession of steps:

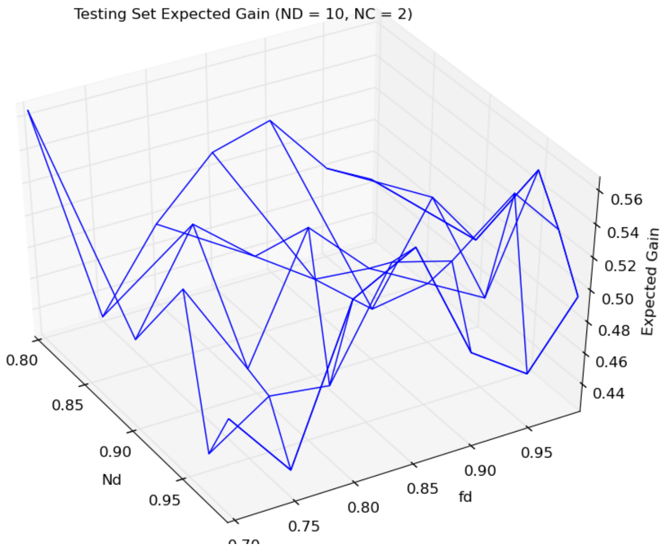
1. A joint probability table is generated of dimensions (NC x ND)
2. A training sample of some pre-determined size is generated by randomly picking a (class, measurement) pair from the table in step 1.
3. We then use the joint probabilities and the samples to generate a conditional probability table for the training set.
4. A decision rule is then constructed that will decide a class to which each measurement belongs. In other words, a class will be assigned to each measurement based on the conditional probabilities.
5. A contingency (confusion) matrix is then generated based on the decisions made by the decision rule for the training set. For each class, the contingency matrix contains the summation of the conditional probabilities for each measurement that has been assigned to that class.
6. The **training set expected gain** is then computed using the elements of the contingency matrix multiplied by elements in the Gain Matrix. For the purposes of simplifying our decision rule, the gain matrix is set to the identity matrix of dimensions (NC x NC).

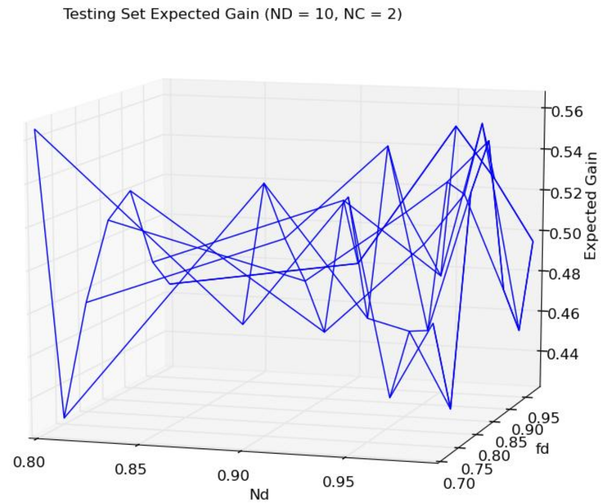
7. Similar to step 2, another sample is generated that will provide our *testing set* of data.
8. Similar to step 3, the conditional probabilities for the testing set are computed.
9. We then use the decisions made for the training set to evaluate our decision rule on the testing set. This evaluation gives yield to another contingency matrix.
10. We then compute the **testing set expected gain** using the elements of the contingency matrix generated in step 9 along with the identity gain matrix.

The effects of the dominant fraction (*fd*) and the number of dominant measurements (*Nd*), should also be noted, as they are heavily used in the following series of experiments.

The dominant fraction is used to constrain the probability of the measurements P(d) where *d* is a measurement in ND. This restriction, in turn, lessens the conditional probability of the class given the measurement P(c, d) where *c* is a class in NC and *d* is a measurement in ND. The conditional probabilities directly influence the decision rule in determining the class that is assigned to the measurement. Of course, it is difficult to predict whether or not an increase in *fd* results in an increase in expected gain since there are quite a few random variables being used in the conditional probability for a class given a measurement computation.

The number of dominant measurements (*Nd*) is used to specify how many measurements are affected by the dominant fraction. The conditional probability of a class given a dominant measurement uses the dominant fraction in its computation, whereas the conditional probabilities of the classes given non-dominant measurements are not affected by *fd*. Although it is equally difficult to determine how an increase in the number of dominant measurements could affect the expected gain, we could assume that as *Nd* approaches 1, all measurements are computed equally. In other words, all measurements are equally affected by the dominant fraction.

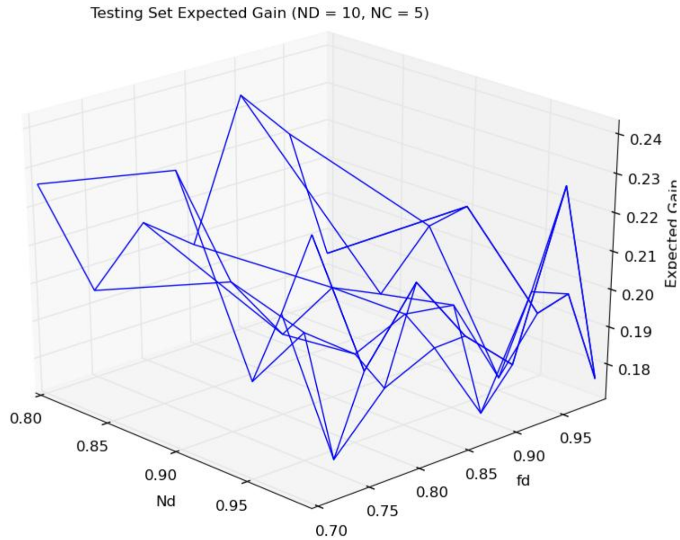


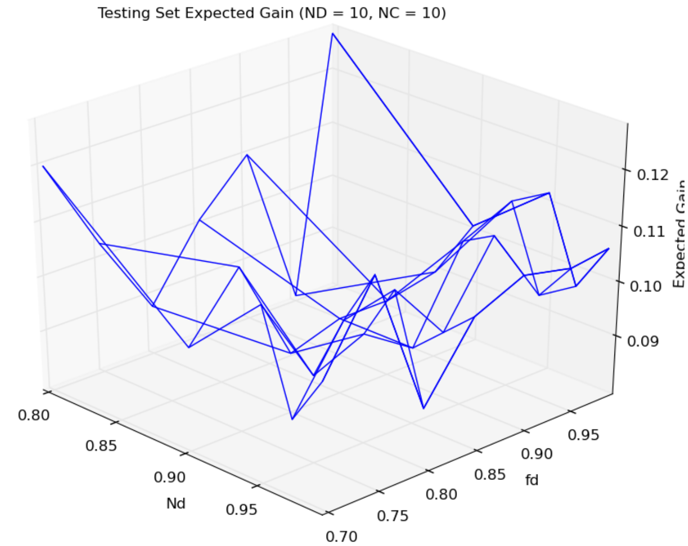


Our experiments begin with the combination ND = 10 and NC = 2. In turn, we refer to this as the **2-class decision problem**. We can see from the above figures that there is a gradual increase in the expected gain for the testing set as both of the variables *fd* and *Nd* increase. Both of the above figures are of the same experiment, but different views are provided to reiterate the increase in expected gain. Of course, there is a very high peak in the expected gain where *Nd* = 0.8 and *fd* = 0.7. This is an unexpected result since, in general, parameter values close to the previous values that generated the peak do not result in equally high expected gains.

We can observe a generally high (which we will conclude when looking at the other experiments) testing set expected gain for our decision rule with the different combinations of *fd* and *Nd*.

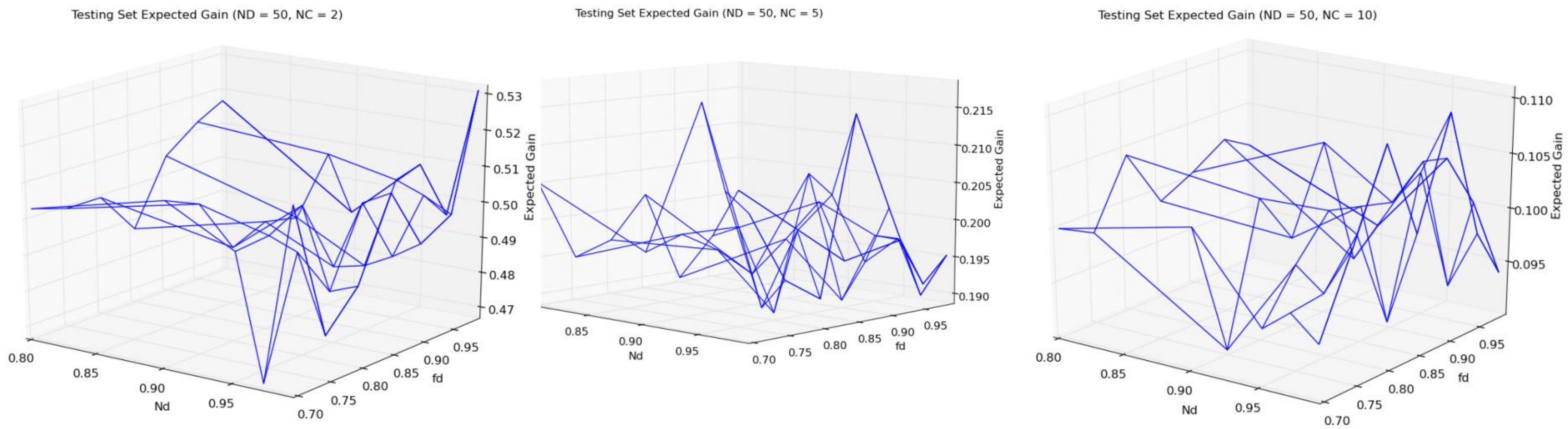
The next set of experiments will increase the number of classes (NC) for 10 measurements (ND), while varying the *fd* and *Nd* parameters. We can then observe trends/patterns that might exist in our tests.



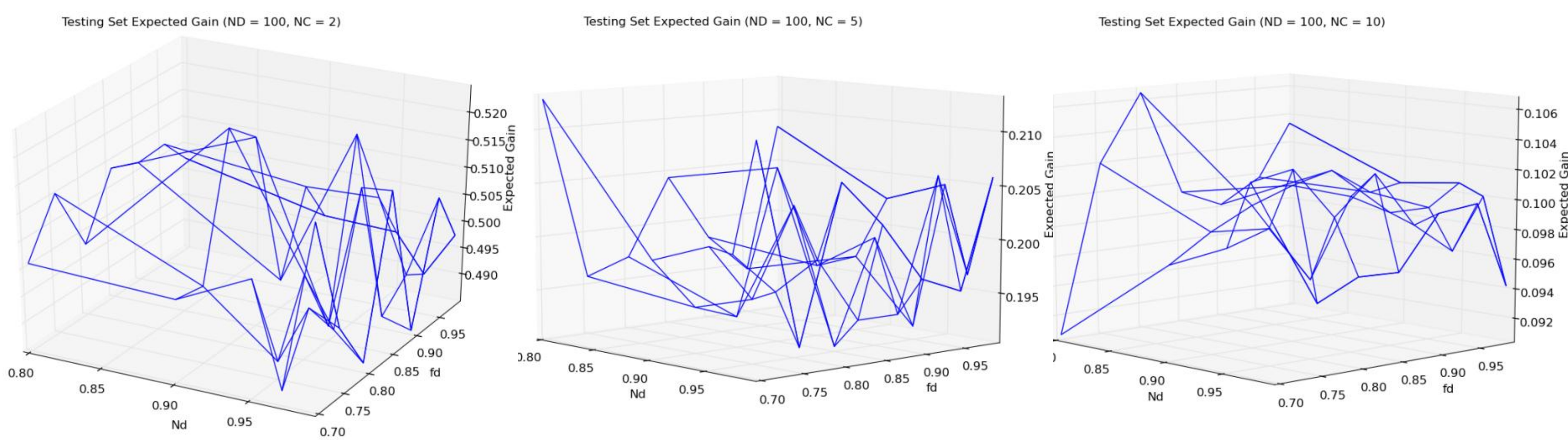


The above figures show the testing set expected gain for 10 measurements with an increasing number of classes and varying parameters *fd* and *Nd*. We must firstly note that the expected gain has significantly dropped for the 5-class and 10-class decision problems, when compared to the 2-class case. In fact, the expected gain gets worse as the number of classes increases for this 10-measurement case, decreasing from a peak of about 0.24 (in the 5-class case) to 0.12 expected gain.

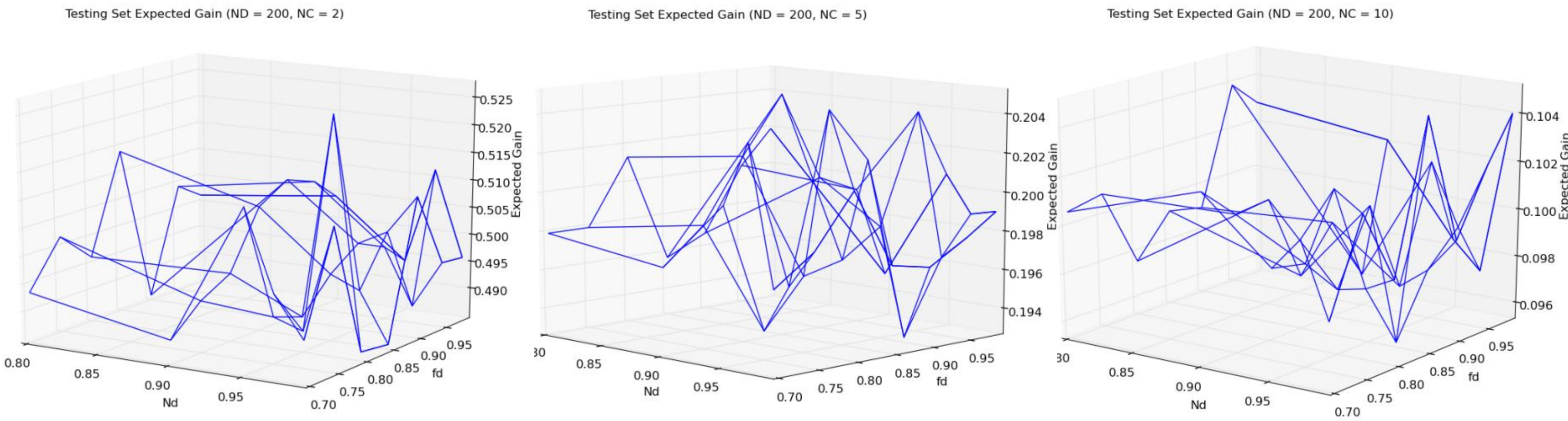
The next set of figures will attempt to show what happens when we increase the number of measurements from 10 to 50, and repeat the tests. Since there are numerous figures, we will move more rapidly through our observations, only noting important patterns/trends.



The above three figures show the behavior of the testing set expected gain for 50 measurements when increasing the number of classes under varying *fd* and *Nd*. We can observe that our decision rule performs well, yet again, for the 2-class decision problem, even with an increased number of measurements. Of course, the results are not as consistently strong as they were in the 10-measurement case (off by a decrease of 0.03), but we have still achieved stronger results than the 5-class and 10-class versions seen in the middle and right-most figures, respectively. The generated expected gain is significantly lower and decreases as the number of classes increases.



The above figures (for the ND = 100 case) show similar results to the previous ND = 50 experiment. Again, the system performs well in the 2-class case, and decreasingly so with an increased number of classes.



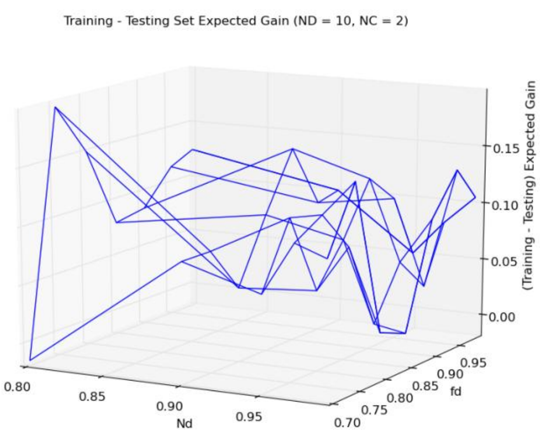
For the figures above, we observe similar results to the previous tests in the testing set expected gain for an increased number of measurements.

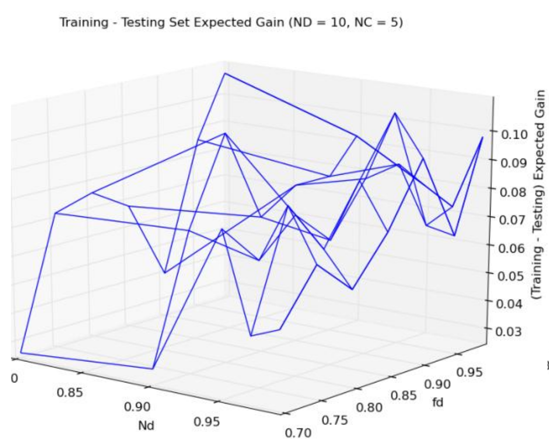
For the final set of plots analyzing the testing set expected gain for the maximal number of measurements, we note the consistency of the patterns in our experiments. Through our tests, the decision rule has the best results for the 2-class case, degrading the expected gain almost insignificantly about the increasing number of measurements. The same *cannot* be said about the expected gain for the experiments increasing the number of classes about fixed measurements. We have noticed significant degradations in the expected gain for those cases.

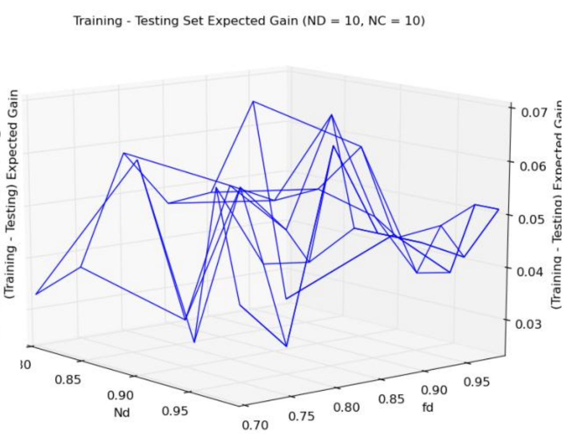
Plot 2 – Training Set Expected Gain - Testing Set Expected Gain

Parameter	Values
ND	[10, 50, 100, 200, 500]
NC	[2, 5, 10]
fd	[0.7, 0.75, 0.8, 0.85, 0.9, 0.95, 0.99]
Nd	[0.8, 0.9, 0.95, 0.97, 0.99]

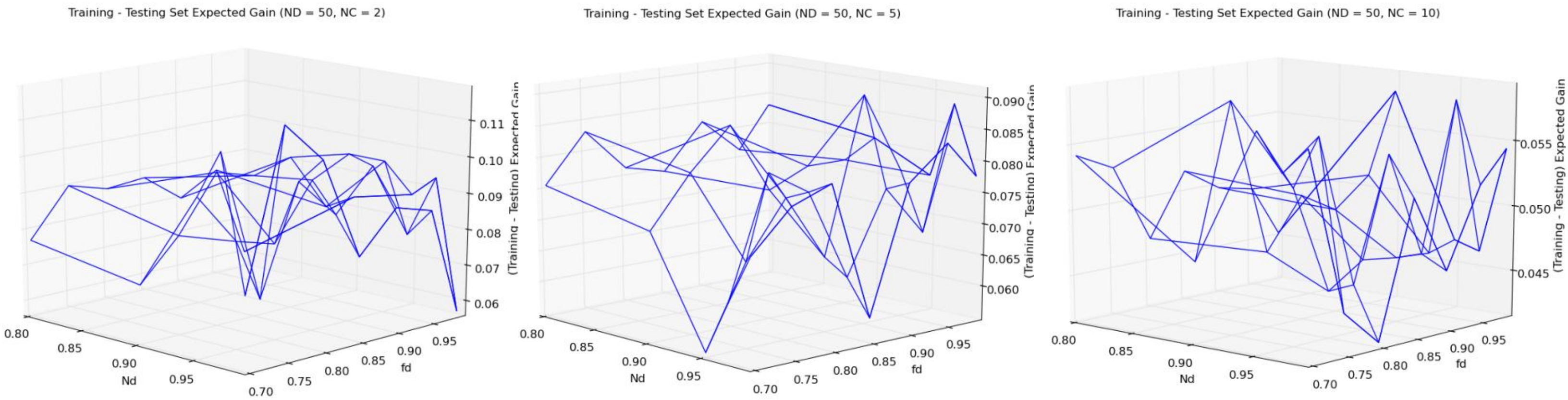
The next set of plots details the differences in expected gain for the training and testing sets generated by the system. Note that the parameter values are the same as in the previous experiment, and the table has been listed again for convenience.





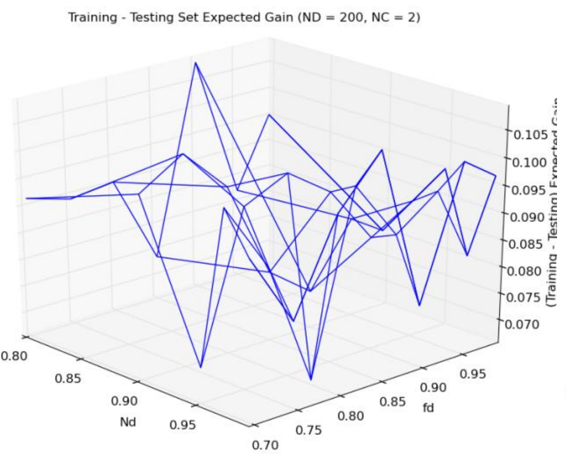


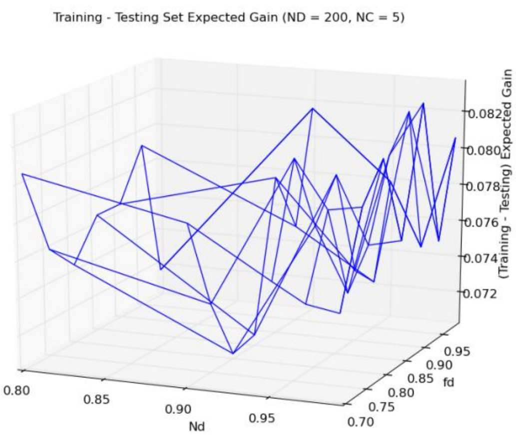
In order to recognize trends/patterns in the series of tests that yields the differences between the training and testing set expected gains, we will show a few sets of plots, grouped by the number of measurements with an increasing number of classes; all under varying values of dominant fractions and the numbers of dominant measurements.

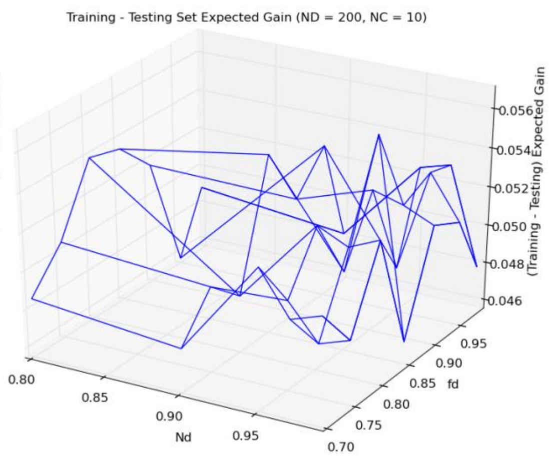


We can observe from the two sets of plots in this section (the above set showing the differences in expected gain for 50-measurements and the prior set showing the differences for 10-measurements), that the differences between the two expected gains are decreasing (towards zero) as the number of measurements increases. This means that as the number of measurements increases, the decisions made by the decision rule become similar for both the training and the testing sets of data.

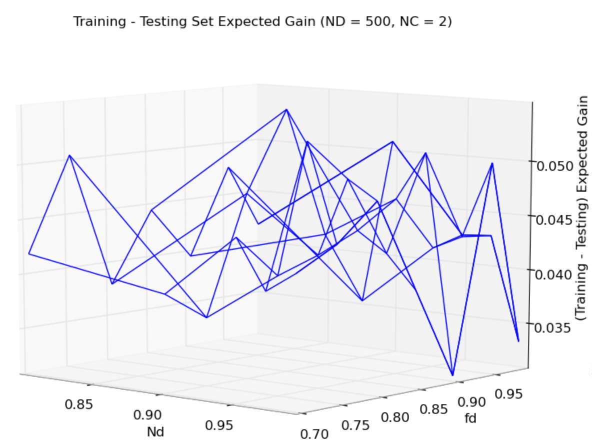
The above set of figures shows the differences in the expected gain for the training and testing sets given 100 measurements and an increasing number of classes. In comparison with the prior set of results, we can see that an increase in the number of classes makes the differences in expected gain approach zero. An increasing number of measurements, however, will make the differences approach zero *slightly* quicker.

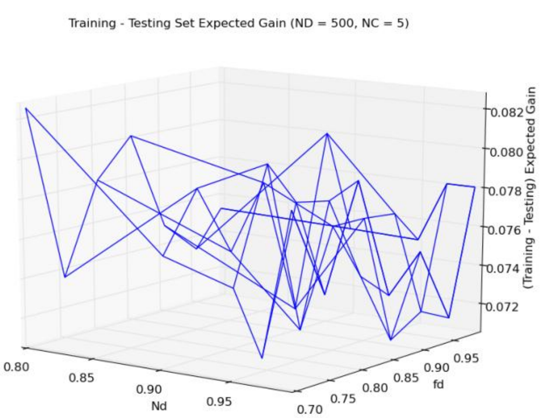


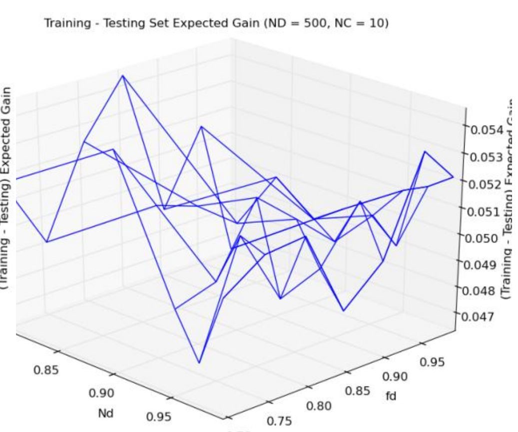




The aforementioned trend still holds true when we fix the number of measurements to 200 and vary the other parameters; as can be seen from the above wireframe surface plots.







This final set of plots, showcasing the differences in expected gain between the training and testing sets, further emphasizes the trend of the differences approaching zero as both the number of measurements and the number of classes increase.

Plot 3 – Testing Set Expected Gain for Special Cases

Parameter	Values
ND	[10, 50, 100, 200, 500]
NC	[2, 5, 10]
[fd, Nd]	[0.7, 0.8], [0.99, 0.99], [0.85, 0.9], [0.7, 0.99], [0.99, 0.8]

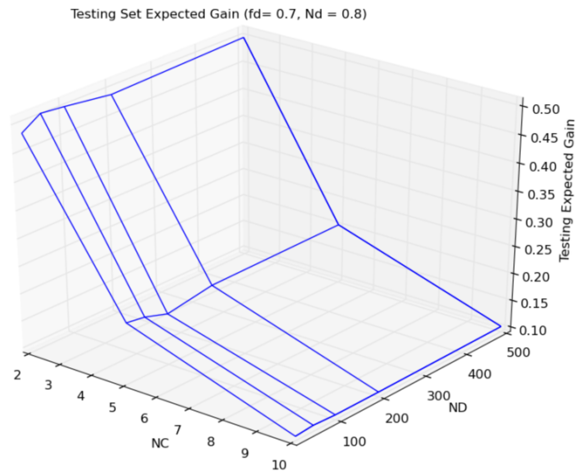
In this experiment, we found some special instances of *fd* and *Nd* combinations that had interesting behavior when analyzing the testing set expected gains. The term "interesting" refers to sharp spikes in expected gain, or combinations that would theoretically increase the expected gain.

For example, let us consider the pair [0.99, 0.99] for *fd* and *Nd*, respectively. This pair should yield interesting results because in the computation for the conditional probability of a class given a measurement, when *fd* and *Nd* are near one, all measurements are computed equally – negating the effects of the conditional probabilities for a class given non-dominant (i.e. measurements in ND but not in the mathematical floor of (*Nd**ND)) measurements. This pair [0.99, 0.99] could be considered the **upper bounds** of the parameter values.

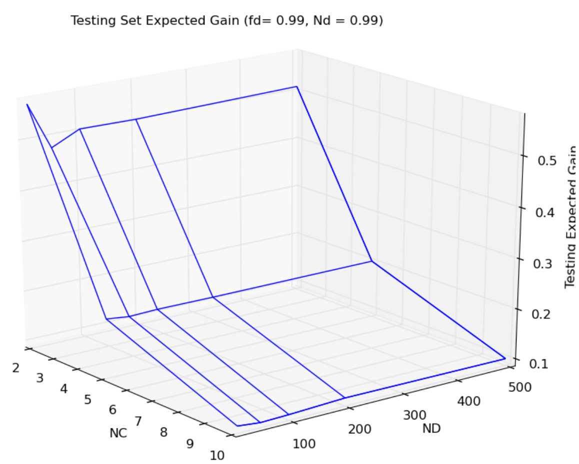
Another interesting pair [0.7, 0.8] is the **lower bounds** of the two parameters. Likewise, it should be interesting to see how the effects of both *fd* and *Nd* manipulate the expected gain of the testing set. Two other pairs [0.7, 0.99] and [0.99, 0.8] are the **mixed bounds**: [lower bound, upper bound] and [upper bound, lower bound] pairs for the parameters *fd* and *Nd*, respectively.

Lastly, the pair [0.85, 0.9] was chosen because it resulted in fluctuations in expected gain for a few of the simulations.

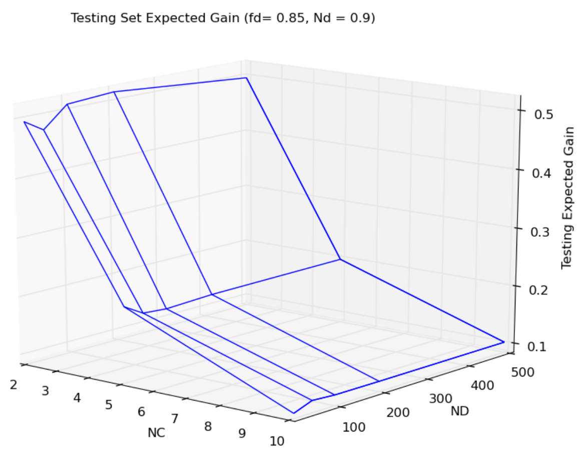
The following plots will showcase the testing set expected gain results of the fixed pairs of *fd* and *Nd*, being simulated over all possible combinations of ND and NC.



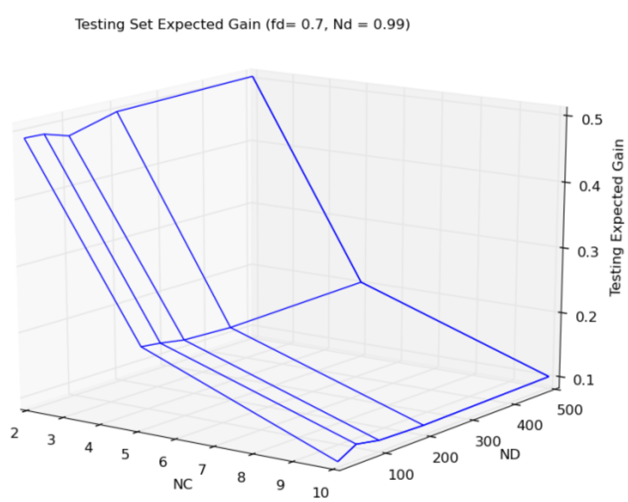
In this plot of the pair [0.7, 0.8], also known as the lower bounds of each parameter, we note the strong resulting expected gain for the 2-class decision problem. However, as the number of classes increased to the 5-class and 10-class problems, the testing set expected gain got increasingly worse. The significant drop in gain can be observed between the 2-class and 5-class cases: a decrease of about 0.3 expected gain.

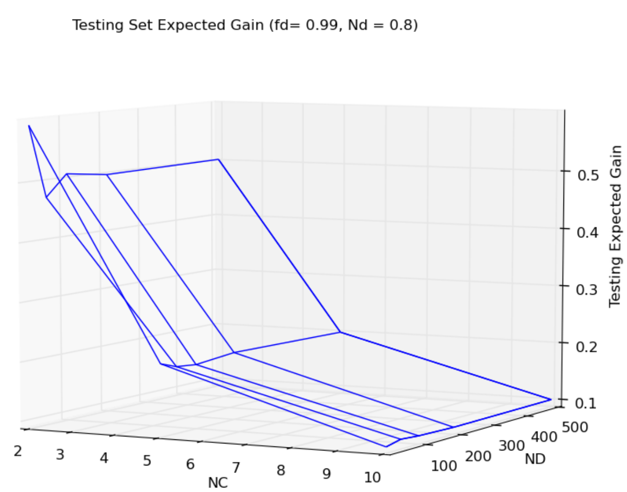


The above figure shows the simulation being run for the upper bound values of *fd* and *Nd*, namely 0.99 and 0.99, respectively. Similarly to the prior, lower bound simulation, we have a strong expected gain being noted for the 2-class case and a decreasing gain as the number of classes increases. This plot, however, shows that the maximal expected gain in our simulation is obtained from the NC = 2 and ND = 10 parameter values. There is an approximate 0.06 drop in expected gain from the 10 measurement case to the 50 measurement case.



This plot further strengthens the patterns observed in the last two plots of testing set expected gain for these special fixed cases of the dominant fraction and number of dominant measurements. Again, the 2-class decision problem shows the strongest results in expected gain.



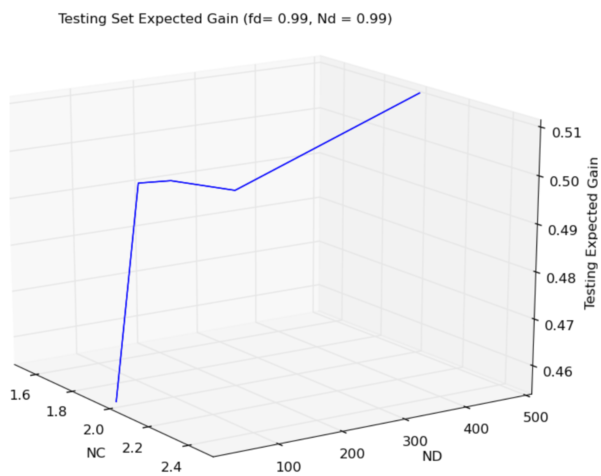


When analyzing the final special case plots for the mixed bounds, we notice almost no drastic change in the observations/results from the previous plots of the upper and lower bounds. It can be noted however, that the system performs best (in regards to resulting expected gain) when the dominant fraction *fd* is fixed at its upper bound. This yields the maximal expected gain.

Plot 5 – Interesting Plots

The next series of plots deals with parameter values that yielded special results or allowed us to reach a conclusion about the system's performance in various cases. The parameter list will be provided in tables, and the plots will be discussed for trends and patterns.

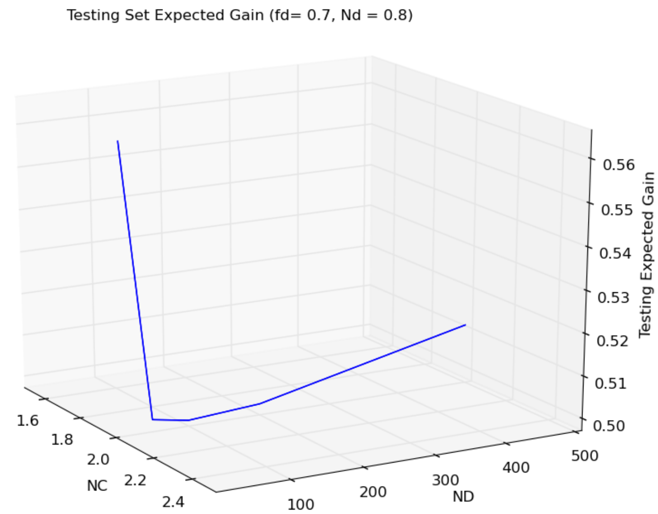
Parameter	Values
ND	[10, 50, 100, 200, 500]
NC	[2]
fd	0.99
Nd	0.99



The first plot focuses our analysis on the 2-class decision problem, where the decision rule performed adequately well with a small number of measurements. In the plots for expected gain for special cases, we noted that the system performed reasonably well when the dominant fraction (*fd*) was fixed to its upper bound: 0.99. The above plot showcases the testing set expected gain for the 2-class decision problem with an increasing number of measurements over a fixed set of parameter values, namely [0.99, 0.99] for *fd* and *Nd*, respectively.

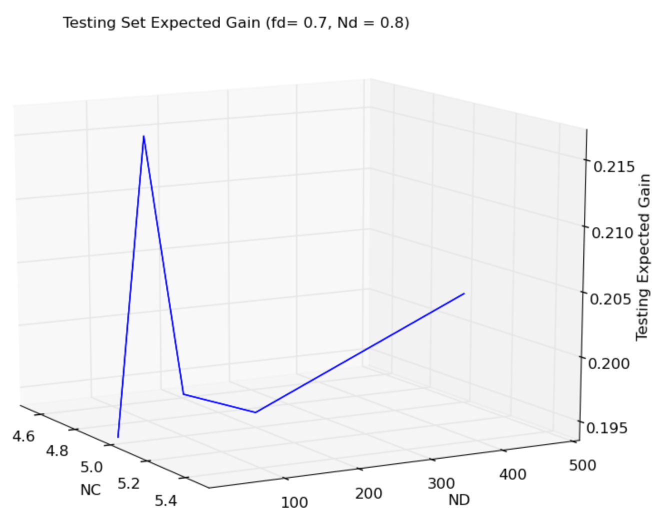
The above plot allows us to conclude that the system yields a consistently high expected gain in the 2-class scenario, especially when the number of measurements is increasing. Of course, this is not the set of parameter values that yields the highest expected gain.

Parameter	Values
ND	[10, 50, 100, 200, 500]
NC	[2]
fd	0.7
Nd	0.8



The above experiment isolates the optimal set of parameter values that yielded the maximal expected gain in our trials. As we can clearly observe from the figure above, the system performs at its best under the given parameter values and ND = 10. Increasing the number of measurements from 10 to 50 results in the 0.06 drop in expected gain; however, surprisingly, it seems that the system gradually recovers in its performance as the number of measurements increased.

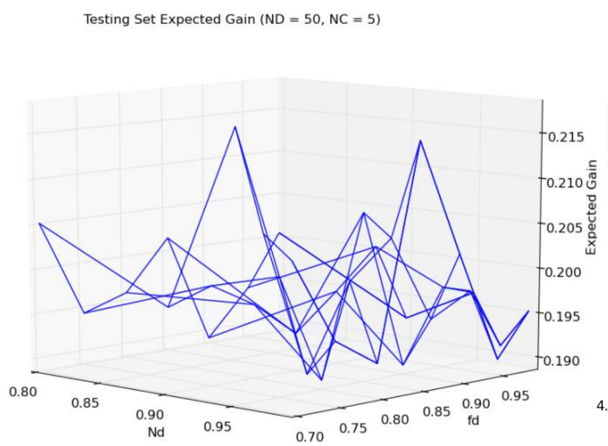
Parameter	Values
ND	[10, 50, 100, 200, 500]
NC	[5]
fd	0.7
Nd	0.8

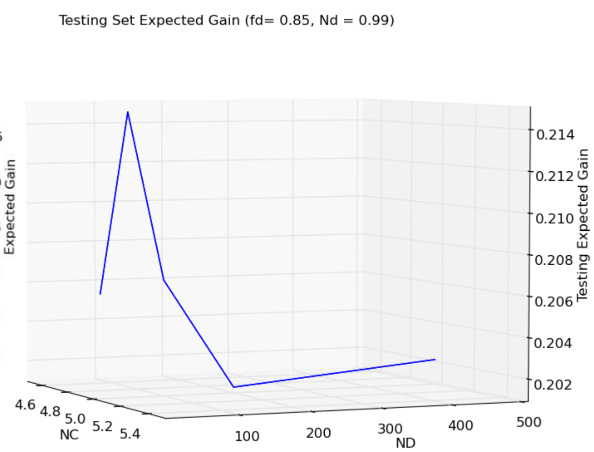


The 5-class decision problem typically yielded significantly lower testing set expected gains when compared to the 2-class case. The above plot attempts to observe the behavior of the expected gain when fixing the dominant fraction and number of dominant measurements over an increasing number of measurements. The values of *fd* and *Nd* were set to 0.75 and 0.8, respectively. These values were chosen because they yielded some of the highest expected gains (in relation to the expected gains across all 5-class scenarios). As we can see from the above figure, the 5-class problem performs best under the 50-measurement case with an expected gain of 0.215. Although there is a sharp decline in expected gain after 50 measurements, the expected gain surprisingly climbs back to around 0.205. This is quite an unexpected behavior that leads us to wonder where the gain cap would occur if the number of measurements was increased beyond 500.

Parameter	Values
ND	[10, 50, 100, 200, 500]
NC	5
fd	0.85
Nd	0.99

The prior plots showed that there was a spike in the testing set expected gain for the 5-class problem with 50 measurements when the parameters *fd* and *Nd* were set to 0.7 and 0.8, respectively. Of course, the expected gain in that plot gradually decreased, which inspired curiosity to see if there were any other notable spikes for the 50-measurement 5-class decision problem.





The left-most plot above shows the testing set expected gain for the 50-measurement 5-class decision problem under varying parameters *fd* and *Nd*. The notable part of the plot is the spike around the *fd* and *Nd* values 0.85 and 0.99, respectively. The spike showed that the resulting expected gain could be equal to that of the result for *fd* = 0.7 and *Nd* = 0.8 (as seen in the prior plots).

The right-most plot above shows the behavior of the expected gain for the 5-class decision problem under a varying number of measurements for fixed values of *fd* and *Nd* to 0.85 and 0.99, respectively. This plot shows that there is barely a gradual increase in expected gain when the number of measurements increases. This is a worse result than the prior parameter values 0.7 (*fd*) and 0.8 (*Nd*), because not only is the maximum gain for this plot 0.214 (a .001 difference than the previous test), there was actually an increase in expected gain as the number of measurements increases for the prior test.

This particular experiment further strengthens the idea that the parameter values defined for the previous experiment (fd = 0.7, Nd = 0.8, NC = 5, and ND = 50) yields the maximal expected gain for the 5-class case.

Conclusion:

Our experiments yielded very interesting results. As can be seen in the Plot 5 set, tests have solidified the idea that the system works best for the 2-class decision problem; with expected gains close to 0.6. Of course, a larger expected gain was anticipated – nearing 1.0; however, achieving a gain close to 0.6 is a very noteworthy decision rule. As an immediate "fix" to adjusting the resulting gain closer to our initial estimations, we believe that the use of values higher than one in the gain matrix (where we currently used 1's along the diagonal) or biasing the decision rule to be rewarded for a certain decision – could push the expected gain values closer to 1.0. Irrespective of the magnitude of the expected gain, the results are conclusively strong for the 2-class decision problem.

The 5-class decision problem did not yield results that were as promising as the 2-class case, but special situations (as seen in the prior section), still yielded consistently decent results. Although the 10-class decision problem was not explored in the previous section, we conclude that this analysis is not necessary, as the resulting expected gains for the 10-class cases were roughly 10 times worse than the resulting gains for the 2-class case. It is undeniable that the system does not perform well for the 10-class case.

Noticing the poor gains for the 5-class and 10-class decision problems further strengthened the decision rule's superiority in the 2-class case. It is our hope that with further testing and revision, we could adjust the decision rule to yield better results in the five and ten class decision problems.