Abstract  
Neurodegenerative Diseases and the complex biological forces which play into their inception beg study to tease apart a realm still overly mysterious. As populations age and their concern about neural degradation increases, the eyes of the scientific community focus with ever-growing intensity on empirical studies of the brain.   
Bioinformatics and Machine Learning methods may allow us to use large neurological datasets to approach the issue of Neurodegenerative diseases. In this paper, I perform a method which applies several machine-learning techniques to several large neurological datasets involving patients of Alzheimers Disease (AD). I gauge the success of several classification methods implemented in the RWeka data-mining package for R 3.0.1, alongside several methods for feature-selection taken from the FSelector package for R 3.0.1.   
  
Aim  
1. To implement classifiers on AD microarray datasets, gauging their success (accuracy, precision, recall) after feature-selection.  
2. To use feature-selection to identify significant genes uniquely expressed in individual Neurological structures. [This was not implemented due to time constraints]  
  
Hypotheses  
1. High-runtime classifiers will perform exceptionally poorly on the larger datasets, though they may succeed on the smaller sizes. The Tree and related classifiers should perform better on the larger dataset, provided there's a significant difference in gene expression between AD and Non-AD patients in the structure.

2. If Neurofibulary Tangles do have a unique and significant expressive-role in AD, then there will be a significant number of genes uniquely expressed in the Tangles. [This was not implemented due to time constraints.]

METHOD (FULL- skipped steps are high-lighted and the reason for their being skipped is outlined   
  
All preparation, coding, and analysis were done on R 3.0.1 for Windows 64-bit.   
This code has not been tested for other versions of R or other Operating Systems.   
This code has not been tested for compatability with any libraries other than those involved in the code.   
The following libraries are used in the code and must be installed prior to running it:

* knitr
* GEOquery
* stringr
* FSelector
* preprocessCore
* audio
* RWeka

I have included a README file with other compatibility information.

Data Gathering and Manual Preparation

1. Gather datasets from the GEO repository. The attached file "PrimaFacieDataAnalysis.pdf" contains a table with a superficial gaze at the fifteen datasets initially encountered for this experiment.
2. Remove any Datasets which are obviously incompatible. For this experiment, the datasets were Microarray (MA) samples with a flat value (i.e. no RGB channel). For this experiment, incompatible datasets included non-MA datasets and datasets with RGB channels. Although the analysis of this experiment focuses mainly on AD datasets, its methods need not entirely be restricted to AD-related datasets.
3. Open a directory exclusively for program running and output.
4. Load all desired Datasets from the GEO repository. These files will be individually loaded and resaved to the directory as preprocessed datasets.
5. Prepare a set of compatibility-evaluation files for the datasets. These files group the downloaded datasets by a certain type of compatibility. For example, disparate dataset metadata, such as differences between genes evaluated for a certain experiment, may prove cause for difficulty in evaluation and analysis, and must be kept into account. Examples of such files included with the zip-file are "EXPTABLE.csv" which groups datasets by experiment, and "REGIONTABLE.csv", which groups datasets by neurological region.

Data Preprocesssing

1. Using the original GEO repository formats, retrieve the diagnosis from the dataset and save it as a vector data-structure to eventually be added to a reformatted dataset. This allows for easily Class accessibility by Feature-Selection and Classification functions.
2. Convert each dataset into a data-structure more accessible in R. For this experiment, the datasets were all coerced into data-frames.
3. Change column (probe-site) names to remove lengthy GEO repository strings. For example, GEO-series datasets have columns initially named "Series-ID]\_series\_matrix.txt.gz.[probe-ID]", which is shortened to just the probe ID.
4. Add the diagnoses in a new column to the data-frame. For this experiment, all diagnoses were binary (i.e. the patient either did or did-not have AD).
5. Write the preprocessed dataset to the preprocessed folder in the working directory. In this experiment, the preprocessed directory is simply named "PP".
6. If the user wishes to test the performance of the analysis on data subsets, there is a convenience function provided which downloads and writes subsets of genes (the dimension of samples is preserved due to the length of the sets).

Data Normalization

1. Each dataset is reloaded into memory, but only one at a time.
2. Each sample of the dataset is log2 transformed. This step is intended to allow the program to deal only with positive data, and with signal-intensities appearing as 'fractions-of-expression' rather than negative or positive. [THIS STEP WAS NOT IMPLMENTED FOR THIS ANALYSIS to reduce runtime. It should not affect the results.]
3. The data is cast into a matrix and a target-sample for quantile normalization is determined. The target is then used to normalize the data between Samples (i.e. between Microarrays). This is intended to prevent strange differences due to unnatural experimental error or meta-data-rooted disparities.
4. The normalized files are written into the Normalized directory. For this experiment, this directory is called "NORM".

Data Enrichment and Feature Selection

1. Three steps of enrichment were intended in the original experiment. The different types were meant to group datasets by neurological region, in order to allow for larger datasets for classifiers to perform on, and also to allow the analysis to easily determine probes indicating higher Genetic expression unique to certain regions of the brain in AD patients.
2. For all three kinds of Enrichment, the following Feature-Selection methods were used: chi-squared, infogain, oner, forst, symmU. Thresholds for these measures were selected via RWeka's "biggest difference" function.
3. The following measures were used for Feature Selection, implemented by the FSelector package (all implemented by Piotr Romanksi):
   1. chi-squared
   2. Infogain
   3. OneR-Algorithm
   4. Random-Forest Importance
   5. Symmetrical Uncertainty
4. **A-Enrichment -** performed only on individual datasets, A-Enrichment applies each of the above-mentioned measures onto a single dataset, selecting probe-sites which may prove more useful for classification later on. Due to the exhaustive size of the datasets, the measure functions were applied to 1000 genes at a time, and probe-site candidates were thus chosen from groups of 1000 probe-sites. After an entire dataset is A-Enriched, it is recompiled and written to a file, which is linked to the dataset name and the Feature Selection method.
5. **B-Enrichment -** first, datasets are compiled into larger datasets using the pre-fabricated compatibility files mentioned earlier. If there are any disparities in data, such as genes which may have been left out for some reason or another from the original dataset, the empty slots are inserted as signals of 0 intensity. This step is intended to allow for higher-sample classification of compatible datasets. [This step was not implemented in the final analysis due to time-constraints and the higher-dimensionality of the method].
6. **C-Enrichment -** All A-enriched datasets are compiled into a large single dataset. Non-Ad (control) samples are removed, leaving only Feature-selected genes in samples of AD-patients. [This step was not implemented in the final project due to time constraints and bugs in the coding.]

Classification and Evaluation of Classification Success

1. For this experiment, 6 types of classification were performed on A-Enriched datasets. Each one was performed on an A-Enriched dataset corresponding to a mode of F-Selection. The implementation of the classification methods was left to the RWeka library. It's possible that other package-implementations may yield different results.

The classification types were

* ADABoostM1, a method establisehd by Freund and Schapire (1996).
* Bagging, a method established by Breiman (1996).
* J48, an unpruned C4.5 decision tree, introduced by Wang & Witten (1997), based on the original M5 tree by Quinlan(1992).
* DecisionStump, a tree classifier which only makes single splits.
* IBk, a lazy learning k-nearest classifier, established by Aha & Kibler (1991).
* LMT, or 'Logistic Model Trees', see Landwehr (2003) and Landwehr et al. (2005).
* *several other methods are available in the source-code but not implemented in the final analysis:*
  + *LBR, M5P, logiBoost, mBoost, stack (see the RWeka documentation for sources-cited).*

1. The predictions of the classifiers then generate unique confusion matrices, which are placed into a unique folder.
2. The graphs plotting success of the classifiers use these confusion matrices for their graphing.
3. Graphs were generated using the mean success of classifiers (mean taken over the different Feature Selection) across the datasets, the success of classifiers (mean taken over datasets) across the different modes of feature selection, the success of modes of feature selection (mean taken over the classifier) across the different datasets.
4. Graphics were also compiled showing the three different measures of success (accuracy, precision and recall) over the above-mentioned dimensions.
5. If a measure, dataset, or classifier is not included in a graphic, it is due to constraints on memory-space and runtime. Future analyses given more time might fill these gaps in data. Examples include dataset 13214 and the random-forest importance measure, the complexity of which demanded runtime not available in some places for this first analysis.
6. The graphics will not include results which were incomplete.

**Results**

1. All results are compiled into their respective folders in the working directory. Preprocessed sets appear in "PP", normalized sets in "NORM", A-Enriched in "A", and confusion matrices in "CM". I've included all Graphics in the "PLOTS" folder.
2. Since the nature of the hypothesis requires us to look most carefully at the success of classifiers over datasets, we take note of the "Success Collection" graphics, which graph all three measures of success (*red is accuracy, green is precision, and blue is recall)* in compressed boxplots against the datasets.
3. Looking chiefly at the file "Success Collection for Classifiers over Datasets", we note that all classifiers performed abnormally well (near 100%) for all scores over the first four A-enriched datasets. The Bag classifier shows the lowest scores, followed by Stump, and the J48 decision-tree.
4. Around the fourth and after datasets, all classifiers begin showing some drops in their scores. IBK drops it recall to near 0 for some nboth the 44678 and 44770 datasets, while its accuracy drops to around 60%, and its precision remains high.
5. J48 suffers in both its precision and accuracy for 44768 and 44770, while its recall remains high for these sets. It remains an almost impossibly successful predictor for all others.
6. LMT encounters almost the exact same performance at J48.
7. The Stump classifier remains powerful until it reaches the 44768 and 44700 datasets, where its accuracy drops to around 50%, while its recall fluctuates from 0 to around 50%, for 44768 and 44670 respectively.
8. ADA's results here are almost identical to Stump's.
9. The Bagging classifier has by far the most variation in its success. It performs the best for dataset 4226, while suffering disparities in its scores for all other datasets, with its lowest scores appearing in its recall for 44770, despite a high precision for the exact same dataset.
10. Other Graphics provide more intimate relations between Classifiers, Datasets, and Methods.
11. We turn to the file "Success Collection for Measures over Classifiers" to look at the relationship between the success scores of different Feature-Selection types in relation to the different methods used for feature-selection
12. What we see here (with red for accuracy, yellow for precision, and blue for recall) is interesting. All measures show similar distributions in their recall and accuracy between classifiers, with only one exception. All measures display widely distributed recall for the IBK, stump, and ADA classifiers. All methods also share similar accuracy distributions for these clasifiers.
13. All feature-selection methods show a similar precision distribution between the j48 and LMT classifiers, while also showing extremely high accuracy and recall.
14. The forst(random forest) method of feature selection shows uniquely high scores for the IBK classifier.
15. The Bag classifier shows widely distributed scores for all methods.
16. More results may be gleaned from other graphics in the PLOTS folder.

**Conclusions**

1. There's something about datasets 44768 and 44770. These two sets, originating from the same experiment, correlate with a dip in the mean performance success of all classifiers. These sets, taken from the Prefrontal Cortex and Visual cortex respectively, demand further review. It's possible the strange behavior results from a gap in the analysis performed here. A further experiment with a similar analysis of other datasets from similar regions of the brain may confirm a correlation between the region of the brain and general performance of the above-tested classification functions, or else it may reveal some issues either in the collection of the data or in the analysis provided here.
2. Classifiers may be badly in this analysis. The exceptional success of the classifiers (excluding Bag) for all other datasets demands some wariness. The results of Classifier Success versus the Feature-Selection method indicates some possibilities. The range of recall and precision seems to abate worries of overfitting somewhat, as we note a distribution more reasonably ranged than one entirely successful; however, we recall the concern with the strange results brought on by datasets 44768 and 44770, and realize that until a similar analysis is performed without these datasets, the danger of overfitting still seems present. A an analysis of classifier performance on non-enriched versions of the datasets might also provide some steps towards gauging the problem of fitting.
3. It is also possible the enriched datasets are not badly fit, and that, excepting the Bag classifier, all other classifiers prove to perform fairly well on most (but not all) datasets. If this is the case, then the structure of difficult to classify datasets, as well as the structure of the classifiers and the methods of feature selection, may bring explanations for the trends observed here.
4. Further analysis using B enrichment while also filling those gaps left by too-large A-Enrichmed datasets would allow us to possibly augment the patterns show mentioned above, and possibly unearth new ones.
5. The hypothesis regarding classifiers did not entirely hold. All tested classifiers seemed to perform fairly well after the A-Enrichment step, the only exception being the Bagging classifier. The datasets 44768, 44770, and other later-tested datasets proved a challenge for all-classifiers. None of the tested classifiers performed outstandingly better than all others, and only Bagging proved notably worse.
6. Research into the dynamics of the Feature-Selection methods, Classifiers and the datasets themselves may help illuminate the results further.