**Abstract**

Data is in the eye of the beholder, or at least that’s the goal of unsupervised learning. Through feature selection and transformation data alternate perspectives of data may provide useful insight into solving problems. Whether it be grouping similar observations via clustering or the more traditional supervised learning problems, unsupervised learning offers a new perspective on the data, one that aims to shed new light on familiar data. This paper examines two datasets in a variety of these contexts: via clustering, as well as data transformations to reduce dimensions by implementing PCA, ICA, Random Projection and subsequently clustering based on these projections. The strengths and weaknesses as well as insights provided by these steps are further examined.

# Datasets

Interesting analysis requires interesting problems, and in order illuminate the strengths, weaknesses and quirks of the examined supervised learning algorithms two well known datasets from the UCI machine learning repository data are examined.

## Madelon

### Instances: 5000 | Attributes: 440 | Data Types: Continuous (440) | Classes: 0, 1

The MADELON dataset is an artificial dataset created in 2003 for the NIPs conference as part of a feature selection challenge. The target class comes from a group of 32 clusters on the vertices of a five dimensional hypercube. Those points were randomly assigned a class (either 1 or -1). Additionally the five dimensions were transformed by linear combinations to form fifteen more features. To complicate the problem, 480 features of random noise were added to the dataset.

Of interest here is that the Madelon dataset presents a highly non-linear problem where the signal-to noise ratio is very low. 1% of the features are truly useful (the 5 dimensions) while 15 (3%) are superfluous albeit still informative. This leaves 96% as completely useless to learn from.

## Cars

### Instances: 1728 | Attributes: 6 (19 when one hot encoded) | Data Types: Categorical (6) | Classes: 0, 1, 2 for Cars with ratings of Unacceptable (0), Acceptable (1), Good/Very Good(2)

This dataset measures the hierarchical characteristics of cars to predict the overall quality/acceptability rating of the car. There are two overarching characteristics measured: price (buying and maintenance) and technical (number of doors, rider capacity, trunk size and safety rating) . Within those six attributes measured, each has multiple ordinal levels. The dataset was originally designed to showcase a simple hierarchical model for decision making. It serves as a test for a learner’s ability to recognize structure within the six dimensions. After processing the categorical values to a single dimension for each (e.g. rider capacity processed from one dimension with three levels into three dimensions with one level, each designating a separate condition (capacity-2, capacity4, capacity5+), the dataset expands to 1728 instances with 19 dimensions. A learner will have to distinguish which of the dimensions are related (e.g. capacity-2, capacity-4, capacity5 are not independent) and model them accordingly. Additionally, the classification problem is imbalanced with the vast majority of cars classified as unacceptable (70%), while 22% are acceptable and the final 8% classified as good/very good. The distribution of each category is balanced between levels (e.g. capacity-2, capacity-4, capacity5 each have 1/3 of the total instances in each level).

***NN Baseline Results***

In previous analysis, an artificial neural network was trained on each dataset. The optimal results are shown below:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Dataset** | **5 Fold CV Score** | **Activation Function** | **Learning Rate** | **Hidden Layer Size** |
| Cars | .5159 | Relu | .01 | (18) |
| Madelon | .7628 | Relu | .0001 | (62, 62, 62) |

These show very different architectures of neural networks. Cars is a small (single hidden layer) NN with mediocre performance while Madelon is a deep NN (3 hidden layers, each with 62 neurons) and a very small learning rate. This is an indication of the types of datasets for each, small and high signal for Cars and larger with complex non-linear relationships for Madelon. Also of note here is that the relu activation may have performed better than the logistic activation due to its sparsity in subsequent layers, a form of feature selection.

# Algorithms & Methodology

All clustering and projection algorithms were implemented via the python machine learning package sci-kit learn. For each algorithm, the learner was five fold cross validation trained using balanced accuracy as the performance metric across a variety of hyperparameters. Balanced accuracy was chosen in light of the Cars dataset’s class imbalance, as without the cost balancing most learners would likely not focus on correctly classifying ‘good’ cars as they represent only 8% of the sample. The best parameters were stored, with the best performing classifier then trained on varying amounts of the data with its performance and wall clock time recorded to illustrate its learning curve and computation cost. The variance or expressiveness of the iterative/ensemble learners (ANN, Boosting, SVMs) was tested by measuring the train and test accuracy across an increasing number of iterations using hyper parameters with high expressiveness (i.e. regularization parameters set to very low values). All models are presented with their best selected hyperparameters and their respective learning curves and timing curves. If the graphs revealed something interesting, the ‘expressiveness’ curves were presented as well. All hyper parameter results and subsequent discussion can be cross checked with the result csv files under

‘reports/output/{learner}\_{dataset}\_reg.csv’. The change in hyper parameters across a dimension is discussed but rarely plotted due to space constraints. The files ‘reports/output/{learner}\_{dataset}\_reg\_pivot\_table.csv’ illustrate how scores varied across certain parameters, which is further discussed. All figures for the plots used (plus additional unused graphs) are under ‘reports/figures’.

## Clustering Algorithms:

## K-Means

## Expectation Maximization (Gaussian Mixture Model)

## Unsupervised Learning / Dimension Reduction / Alternate Data Representation Algorithms:

## Principal Component Analysis (PCA)

## Independent Component Analysis (ICA)

## Random Projection (RP)

## Random Forest Feature Selection (RF)

# Results

## For each, talk mostly about representation of the data. How well does it retain information?

* Across number of components: reconstruction error and other individual metric

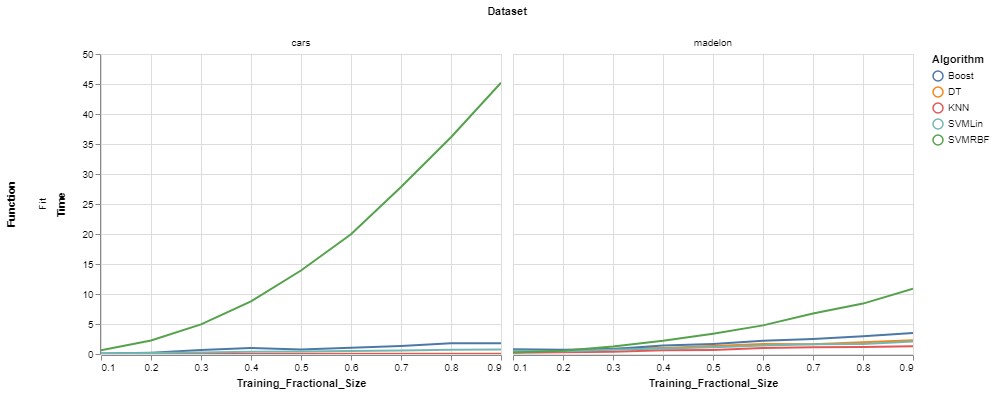
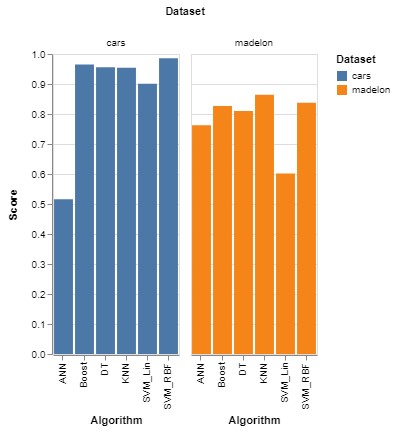
## Principal Component Analysis (PCA)

## Independent Component Analysis (ICA)

## Random Projection (RP)

## Random Forest Feature Selection (RF)

# Concluding Remarks



Left: 5 Fold CV Score

Bottom: Training Fit Times

(

ANN not included as it skewed the y axis as

to make comparison visually irrelevant)

The Cars and Madelon datasets provided contrasting problems for the supervised learners examined: one a small, high signal, hierarchical and sparse dataset (Cars) with the other a large (5000x500), high noise, non-linear dense dataset. The SVM with an RBF kernel performed the best overall with the highest score on Madelon and 2nd highest on Cars. Boosting did well as an ensemble method scoring 2nd best Cars and 3rd best on Madelon. The KNN proved to work best with the Madelon scoring a .86 overall (1st) but struggled on the Cars set (4th).

These problems and subsequent analysis further hammer home the necessity of aligning the problem space with the strengths of a learner and careful hyperparameter tuning to aid the learner with domain knowledge. Each of these learners possesses contrasting strengths and weaknesses- *this is not a bad thing* but rather an opportunity to craft a tailored solution to any problem. One size fits all solutions are never optimal, and such is the case here as evidenced by the datasets and the problems they pose. Before any model fitting takes place, it pays dividends to diagnose the unique aspects of the problem space and apply learners best suited to find a solution. Take the Cars and Madelon dataset as examples. It would make the most sense that a hierarchical problem would lend itself well to a decision tree. Lo and behold a boosted trees model scored very well on it, and with tuning performed very well with minimal computational expense. Similarly, for the Madelon data that was artificially created using points on a 5 dimensional hypercube was best approximated by KNN. KNN classified instances based on a weighted similarity score that came from a distance measure that lends itself well to high dimensions (manhatttan). The RBF SVM outperformed its linear counterpart due to the nonlinear nature of both datasets and its ability to project into higher dimensions.

Of course there are always more considerations when searching for the optimal solution, such as computational complexity. Although the RBF SVM performed best on the Cars dataset, it requires much more computation and time for convergence. If the use case involved massive amounts of data it would make far more sense to use a fast method such as a decision tree or boosted trees method which lends well to scale. Machines learning models are fast, accurate, and stupid. This of course is the case without careful analysis and thoughtful application, as this paper’s analysis illustrated the core importance of proper algorithm application and tuning involved in solving supervised learning problems.