Neurotrophy: Improving Classification of Skin Diseases through Evolutionary Selection of Training Data

David Norrish[1]

[1] Research School of Computer Science,
Australian National University
Canberra, Australia
david.norrish@anu.edu.au / u4815128@anu.edu.au

Abstract. Perhaps surprisingly, reducing the size of a training set can, in certain circumstances, improve the generalisability, performance, and training time of feed-forward neural nets. Simple neural nets were trained to classify a family of dermatological diseases on the basis of clinical and histopathological features. An average accuracy of 97.7% was achieved with minimal optimisation of hyperparameters. Two methods were then applied to reduce the training set: heuristic pattern reduction and a novel evolutionary pattern reduction approach termed "neurotrophy". It is seen that heuristic pattern reduction allows reduction of the training set to 50% with no impact to accuracy, and to as low as 25% with minimal impact. Neurotrophy, when run for 15 generations, evolved to use 74% of the dataset for training, with no impact on performance compared to using the full training set.

Keywords: pattern reduction, evolution, dermatology, PyTorch, neural nets

1 Introduction

A number of painful erythemato-squamous diseases, such as psoriasis, seboreic dermatitis, and lichen planus, present with very similar clinical manifestations. As these dermatological diseases can require quite different treatment regimes, an accurate diagnosis is nonetheless crucial to achieving a positive clinical outcome. The conventional procedure for disease identification is time-consuming and invasive, requiring a biopsy and subsequent microscopic examination. There is therefore clear imperative to develop improved diagnosis procedures.

A number of machine learning approaches have been adapted to this domain in the literature. These include K-means clustering [1], boosted Decision Trees [2], voting feature intervals coupled to k-nearest neighbours [3], fuzzy extreme learning machines [4], and genetic algorithms [5]. These approaches have been high successful for a popular open data set of dermatological diseases [6], with accuries ranging from the low to mid 90%s in early papers which employed simple clustering methods, to upward of 99% accuracy for the latest fuzzy and evolutionary approaches.

There has been a lot of research recently (e.g. [7]) into hyper-parameter optimisation methods to maximise the performance of neural networks. These approaches have led to refinement of performance on a wide range of tasks, and in some cases (such as grid search), can deliver globally optimal performance — albeit at enormous computational cost. Another potentially useful approach, which has been relatively more neglected, is selection of the optimal *training data* to include. As well as reducing training time, this approach may improve model generalisability (e.g. by discarding unhelpful outlier samples) and provide insights into working with small dataset.

As a baseline for training data selection, I implement a technique known as heuristic pattern reduction, first described 25 years ago [8], which uses loss contribution to select samples for retention. To build from there, I explore the use of an evolutionary approach for discovering an improved set of training samples to use. At a high level, this approach treats each sample in the training dataset as a unit in a boolean "DNA", whose purpose is to determine which samples to train on. By allowing several agents to compete for the best performance and mutate their DNAs over generations, the process is expected to trend towards an improved subset of training samples. I term the approach "neurotrophy", combining the technique of neural nets with the Greek "troph", to nourish. The idea is that neural networks can evolve to learn the most valuable data samples to "nourish" themselves with by training on.

2 Method

2.1 Data preparation

The UCI Dermatology dataset comprises 35 columns, corresponding to disease classification and 34 assorted clinical and histopathological variables. There are 366 rows, corresponding to independent patient cases. Almost all variables come pre-banded as integer values in the range 0-3, indicating a lack of the symptom (0) or the symptom strongly present (3). These numbers roughly denote degree of severity, meaning they can be considered as numerical. Family history is a binary variable encoded as 0/1 for absent/present, and age ranges fall from 0 (a newborn) to 70.

Eight samples were identified with missing age data. As this represented only a small fraction of the data, these samples were discarded. The neural network architecture required equal size inputs, and problems may arise if age ended up being an important variable, i.e. weights from that neuron were high.

All variables were visualised to assess their distribution and scope for normalisation (see **fig. 1** for several examples). Age is normally distributed, which would suggest z-scores as being most appropriate to capture the significance of outliers. Of the banded variables, several seem roughly normally distributed, though this is hard to tell with only 4 possible values. Because they are already defined in a hard limit of 0-3, meaning there is no risk of outliers skewing results, they were uniformly normalised to floating point numbers in the range of 0-1. This is unlikely to materially affect results, and done more in keeping with convention. Age was converted to z-scores. This results in a slightly wider range of possible values for age than for the other variables, as well as negative input values. The difference in magnitude is not substantial though, and as the sigmoid activation function in the hidden layer can readily handle negative values, no further adjustment was deemed necessary.

The six disease classes were encoded as arbitrary integers in the range of 0-5. This format is suitable for PyTorch to compute cross-entropy loss during training.

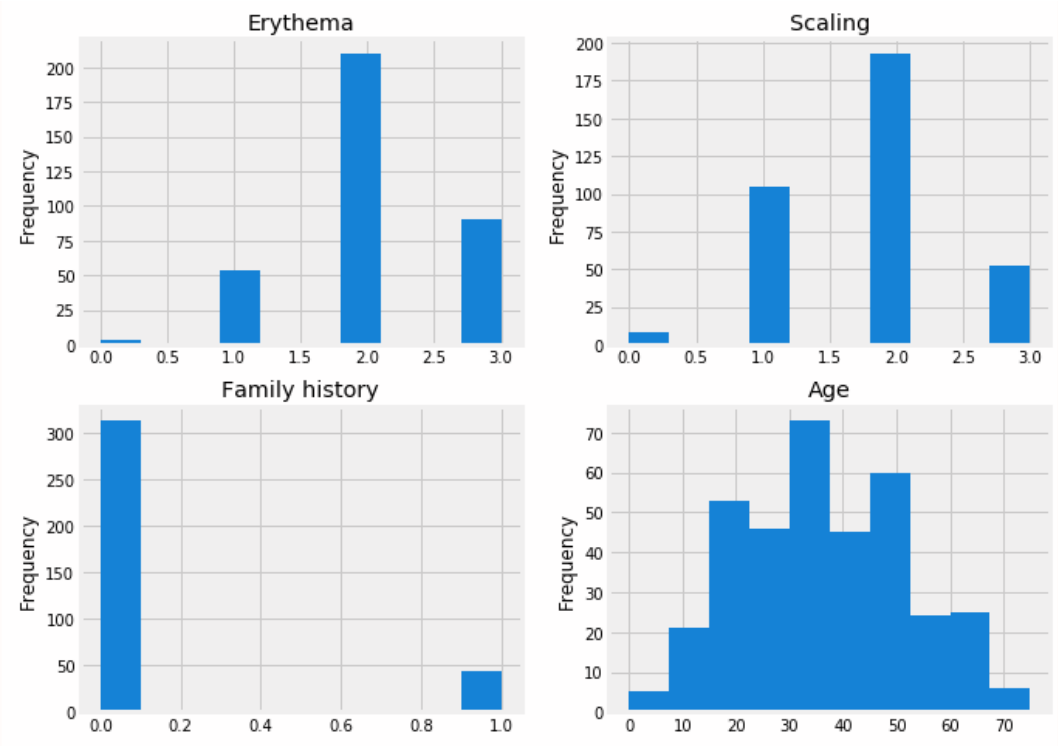

Fig. 1. Distributions of several characteristic variables, along with the two exceptions, family history and age. Most variables were banded and seemingly roughly normally distributed. as is the case for erthyema and scaling, two clinally obtainable variables.

Highly unbalanced ratios between examples of different training classes can skew a network toward learning just the most common class or classes. In such cases, interventions such as oversampling of minority classes or weighing the loss function based on class frequency may be important. We therefore visualised the count of each disease class (**fig. 2**). Four of the classes are roughly equally represented, with one class having significantly more examples and one class significantly fewer. This degree of unbalanced was deemed not serious enough to merit pre-emptive intervention, but a resolution was made to investigate classification efforts of the first trained networks with confusion matrices, and take action if there was failure seen for the minority classes.

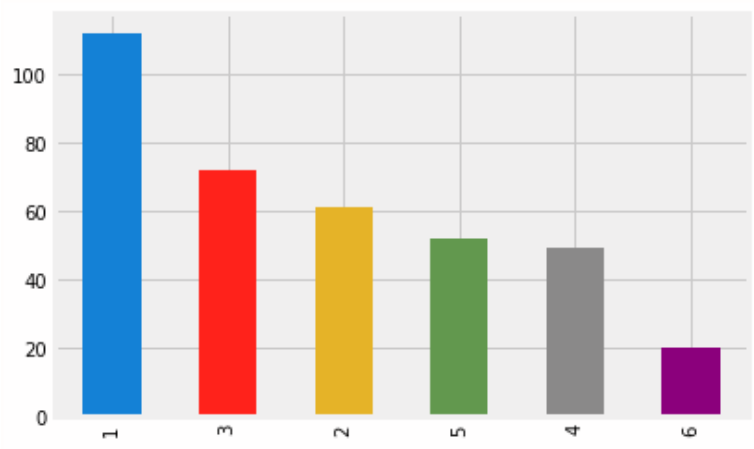

Fig. 2. Counts of the 6 disease classes in the original dataset. Class 1 (psoriasis) is notably more common than the other diseases, and class 6 (pityriasis rubra pilaris) is notably rarer.

Finally, the dataset was randomly permutated to dissolve any possible structure in the original listed order, and saved to disk as a CSV file.

2.2 Model validation

Two approaches were taken to validate the model. The first was using a holdout dataset, in keeping with the original heuristic pattern reduction paper [6], and the second was using 10-fold cross validation. In the holdout case, because this data set is slightly larger than the original Gedeon paper, only 70% of samples were used for training and the remaining 30% set aside as a validation set. Classification accuracy was uses as the foremost measure of model performance, as this is the ultimate desirable clinical application of applying machine learning to this dataset.

2.3 Neural network topology and hyperparameters

A simple feed-forward fully connected neural network model was defined using PyTorch (0.4.0) and used for all trials. This involved 34 inputs, corresponding to all variables of the dermatology dataset. These were fully connected to only 5 hidden neurons, which were then fed through a sigmoid activation function. The hidden layer was fully connected to 6 output neurons, representing the different disease classes.

Initial investigations were made into topologies with a greater number of hidden neurons, however this did not significantly improve performance, so the number was kept at 5 to i) reduce the risk of overfitting, ii) minimise training time, iii) more faithfully replicate the topology of the original heuristic reduction network [8].

For all training runs, 1000 epochs were used, with mini-batches of size 25. Models were validated using the full holdout set, generally comprising either 30% of the original dataset or 10% when 10-fold validation was used. As the task called for classification, cross-entropy was selected as a suitable loss function, with output neuron activations first passed through a softmax function to convert them to probability-like values in the appropriate range of [0, 1].

Learning rate was explored manually, with an optimal range of about 0.05 – 0.1 being found to minimise loss without triggering significant over-training within the 1000 epochs.

Finally, guided stochastic gradient descent was used to update network weights, and interrupt training if necessary. Every 50 epochs, the validation set was used to test loss on the partially-trained network, without updating weights. If loss was found to increase twice in a row, the network was considered to be overfitting to the training set. Training was then halted, and the weights were reverted to the epoch exhibiting the smaller validation set loss.

2.4 Heuristic pattern reduction

After training and obtaining results for our feed-forward network, we investigated the possibility of improving performance by replicating a technique described for heuristically reducing the size of the training set [8].

This involved determining contribution to total sum of squares for each training example, then ranking samples. This set could then be reduced in size by discarding e.g. every 2nd sample, and a de novo network trained. While squared error is generally deemed more appropriate for numerical regression-style tasks than classification, it was implemented here to capture the additional information that cross-entropy loses (i.e. activation for non-target classes), and more faithfully seek to reproduce the original paper. This involved one-hot encoding all disease labels into 6 vectors and using the different mean square error function on a sample by sample basis.

Loss and accuracy was used to compare the performance of networks trained on a range of heuristically reduced samples, including an 80% set, 67%, 50%, 33% and 25%.

2.5 Evolutionary process

In order to evolve toward a training set comprising a valuable subset of the available samples, a *neurotroph* class was defined. This encapsulates a boolean “DNA” list indicating which data samples to include when training, a method to spawn offspring entities, and a mutation rate which determines the likelihood of each DNA boolean (“base pair”) flipping state when creating a child. The mutation rate was constrained in the range of 0 (no mutation possible) to 0.5 (50% chance of any given base changing). The mutation rate could in turn mutate itself. A beta distribution (**fig. 3**) was used to determine child mutation rates based on parent mutation rate, with the alpha and beta parameters set so as to create an expected new value equal to the parents value (see script file #4 for further detail).

A pool of 10 neurotroph instances was generated, and 15 generations were run. In each generation, standard neural nets are instantiated for each neurotroph; 30% of data samples are randomly assigned to the validation set and the remaining 70% assigned as available. Each neurotroph network is then partially trained, using only the training samples prescribed by its DNA, for 2000 presentations of batch size 6. The small batch size allows the potential to evolve very sparse DNAs involving few training samples.

Accuracy rates on the validation set are then determined for each neurotroph network. “Hall of fame” and “hall of shame” approaches are used to retain and discard the best and worst performing neurotrophs respectively. Fitness proportionate selection is employed (**equation 1**) to probabilistically select which additional neurotrophs to retain for the following generation, summing to 50% of the original size (i.e. 5 in this case). Each surviving neurotroph then spawns a single child to rejuvenate the population, with a child DNA mutating from its parent’s DNA probabilistically, according to the mutation rate.

Finally, a concensus DNA is taken across the DNAs of all surviving neurotrophs. This was used to train a final network, and contrast its performance with a similar network trained using the full training set, again using k-fold cross-validation.

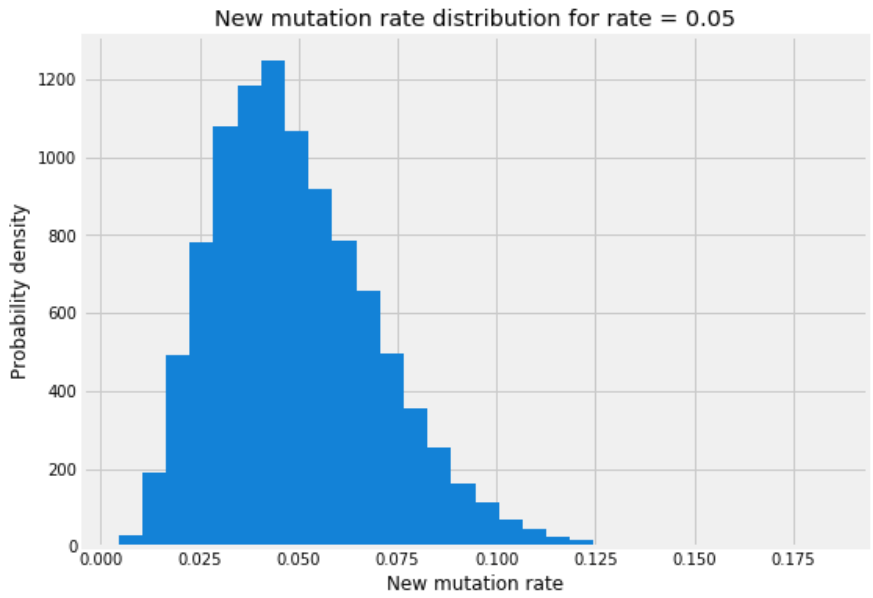

Fig. 3. Probability distribution of child neurotroph mutation rate for a parent mutation rate of 0.05. The mean (expected) value is still 0.05, however the beta distribution function compresses spread of possible mutation rates that would exceed 0 or 0.5

$$p_i = \frac{f_i}{\Sigma_{j=1}^{N} f_j},$$

Equation. 1. Fitness proportionate selection. Accuracies of all neurotrophs are summed, then individual neurotrophs selected probabilistically, with likelihood proportionate to their contribution to the total accuracy.

3 Results and Discussion

3.1 Feed-forward network performance

Loss and accuracy of the training sets were recorded every 50 epochs during network training, using either a 30% holdout validation set or 10-fold cross validation. Guided stochastic gradient descent was employed, whereby validation loss was regularly tested (without training the network), and if loss was seen to be increasing, the network was considered to be overfitting on the training data and training halted.

Using a learning rate in the general range of 0.05 – 0.1, and 5 hidden neurons with sigmoid activation functions, networks typically trained to a loss < 0.25, and achieved near perfect accuracy on the training set within 1000 epochs (**fig. 4**).

Using 10-fold cross validation yielded an overall accuracy of **97.7%**. On this dataset, this is a comparable or even superior result to those achieved by early papers which used approaches such as K-means clustering [1] or boosted decision tree [2], generally achieving accuracies only as high as 96.72%. While this performance is impressive and perhaps surprising given the simple topology and very limited degree of hyperparameter optimisation, better results have been obtained by more recent papers that employed techniques such as voting feature intervals with kNN [3], fuzzy extreme learning machines [4] or AdaBoost couple with the “Apriori” affinity rules algorithm [5]. These modern approaches are able to achieve accuracies as high as 99.57%, or only a single misclassified sample.

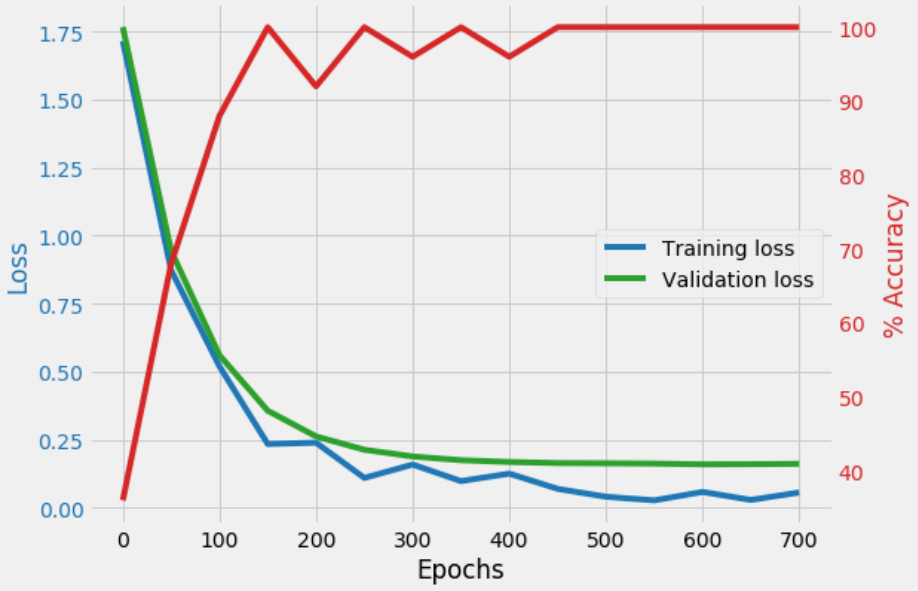

Fig. 4. Representative loss/accuracy curves observed for training of feed-forward neural network. Results shown for training and validation datasets. Loss was measured every 50 epochs, along with accuracy for the training set. Guided stochastic gradient descent was used to safeguard against overfitting, with training interrupted if two consecutive increases took place for validation loss.

3.2 Quantifying Unbalanced Class Effects

There was a concern during the data pre-processing step that due to the unbalanced nature of the disease classes, the model may predominately learn the majority class, or at least fail to adequately learn the minority class, which may have accounted for the few percentage points of errors in prediction. To investigate this possibility, and potentially take corrective action, confusion matrices were generated to plot predicted disease classes against the true labels for the validation sets (**fig. 5**).

It transpired this concern was unsubstantiated. For every iteration that was run, every instance of the minority class was predicted correctly. In fact, that only errors ever seen were misclassifications between two of the intermediate frequency diseases, seboreic dermatitis and pityriasis rosea. Given the very similar symptomology of these two conditions [7], this likely represents a genuine challenge in differentiation rather than a perverse failure of the network.

It is worth noting that half of the 10-fold validation runs achieved 100% accuracy, and other than these two occasionally conflated diseases, all other diseases were identified with 100% accuracy in every run.

3.3 Heuristic pattern reduction

Due to the high degree of accuracy achieved, and the essentially perfected performance on this dataset in the literature, it was of interest to explore a technique which could offer improvements in ways other than our primary measure of interest, accuracy. For this reason, a technique called heuristic pattern reduction was implemented [6]. This involves determining contribution of each training sample to the total sum of squares on a trained network, sorting by contribution, and removing samples at regular intervals, e.g. every 2^{nd} or 3^{rd} sample.

The idea is that samples giving rise to similar square error may contain similar information, so the network could theoretically see proportionally fewer of each type of sample and still train to a high degree of performance. In fact, an improvement to performance may even be seen, due to “simplification of the error surface in pattern space traversed by the network” [10].

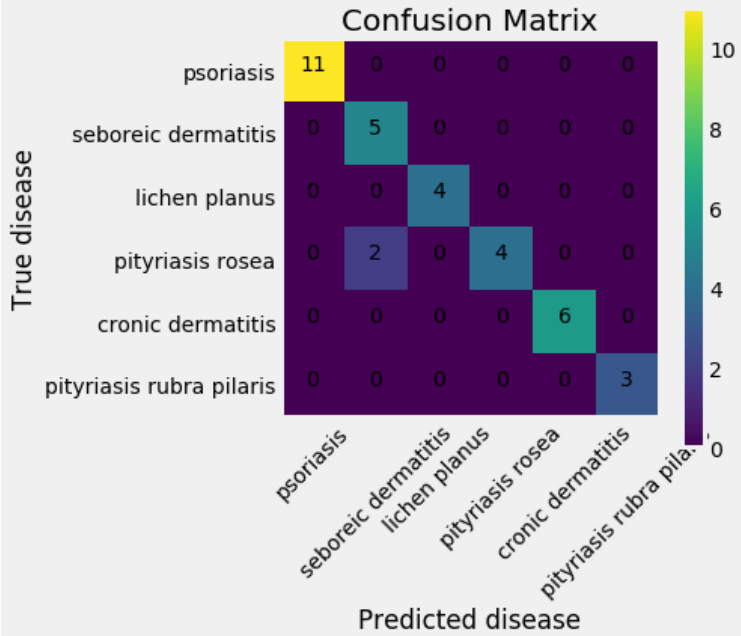

Fig. 5. Confusion matrix showing typical performance of a trained network at predicting disease class of the validation dataset after 1000 epochs of training (or until halted by validation loss increasing). Similar results were seen whether using a 30% holdout set or 10-fold cross validation. Misclassification errors were only ever seen between seboreic dermatitis and pityriasis rosea.

As per the original paper, a holdout set was created (in this case 107 samples, 30% of the total) and the remaining 251 (70%) used to train a new network for 1000 epochs. The same result was repeated using 10-fold cross-validation, yielding a 322/36 (90%/10%) split. Following training, the disease label vector was converted to a one-hot encoded matrix for the 6 disease classes, and the same samples were passed through again. This time their contribution to total sum of squares was determined, in accordance with the original paper [6]. Samples were then ranked in order of increasing error contribution and visualised (**fig. 6**). It was seen that the great majority of samples were learned extremely well by the network and contribute close to 0 squared error, while a small fraction (approximately 8%) contributing almost the entirety of the error.

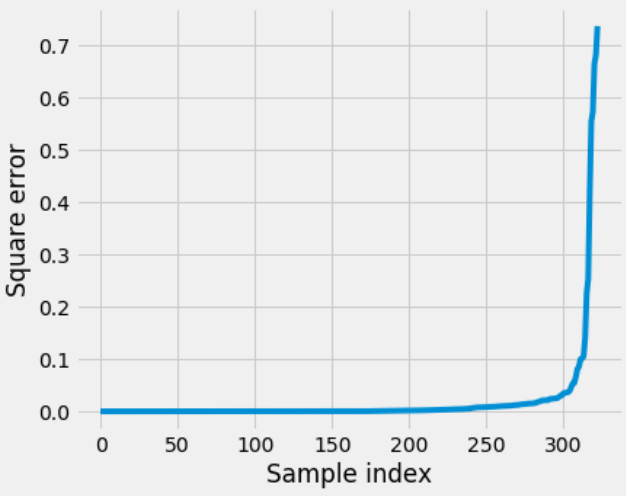

Fig. 6. All training samples were used to train a network for 1000 epochs. Then their activations were were passed through a softmax function and used to calculate sum of squares . Samples were then sorted by their square error across all 6 disease classes and ranked.

Using this adjacency vector of squared error, clusters of arbitrary sizes were assumed and used to create subsampled trainined datasets of various sizes. There were: full set, 80%, 67%, 50%, 33% and 25%. These reduced training sets

were then used to train 10 brand new networks each (being sure to use the same seeds for randomisation once per iteration for all datasets to avoid introducing error there).

As with the original network, the training and validation loss (**fig. 7**) and accuracy (**fig. 8**) were tracked during training. Performances on the 10 runs were averaged and visualised. Findings were broadly consistent with the original paper. As expected, there is no improvement to accuracy, though this may have been limited by the very high performance. Loss on the validation set and prediction accuracy during training show no discernible difference among any of the reduced training sets and the full set. Prediction accuracy of the fully trained nets showed no statistical differences from each other according to ANOVA, though there was a trend of sightly lower accuracy on the 25% and 33% networks, often around 96%, versus 98% for the larger training sets.

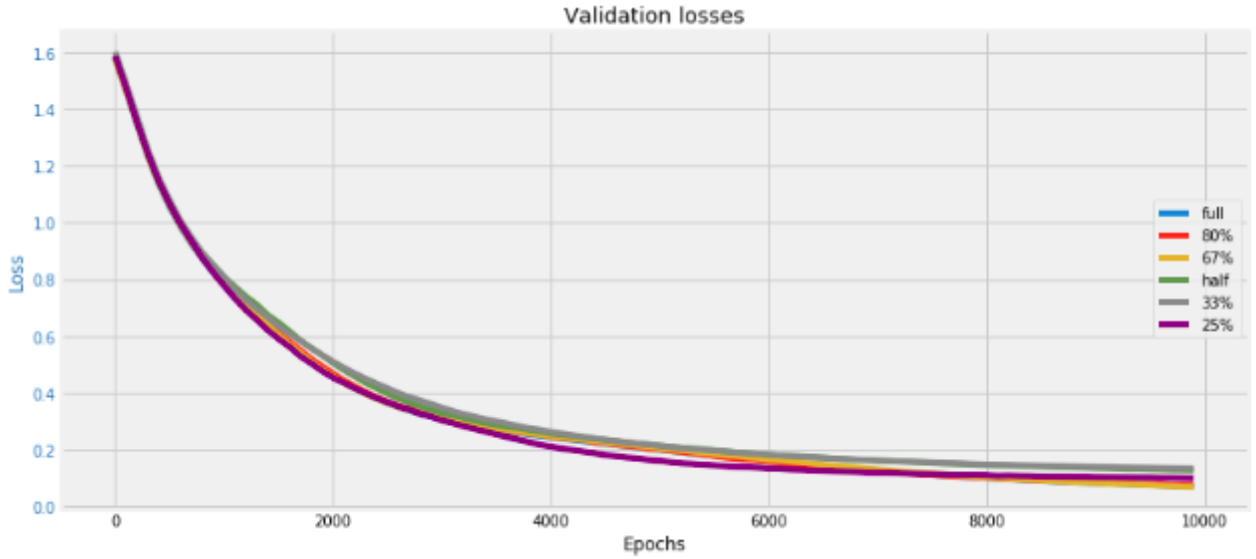

Fig. 7. Validation loss during training of networks using a full-sized training dataset (70% of total rows), or subsets with varying proportions of data discarded. Rows were discarded at regular intervals after all samples were sorted based on their contributing sum of squares. This is a representative sample of one of10 such runs which were made with different initialisation weight seeds

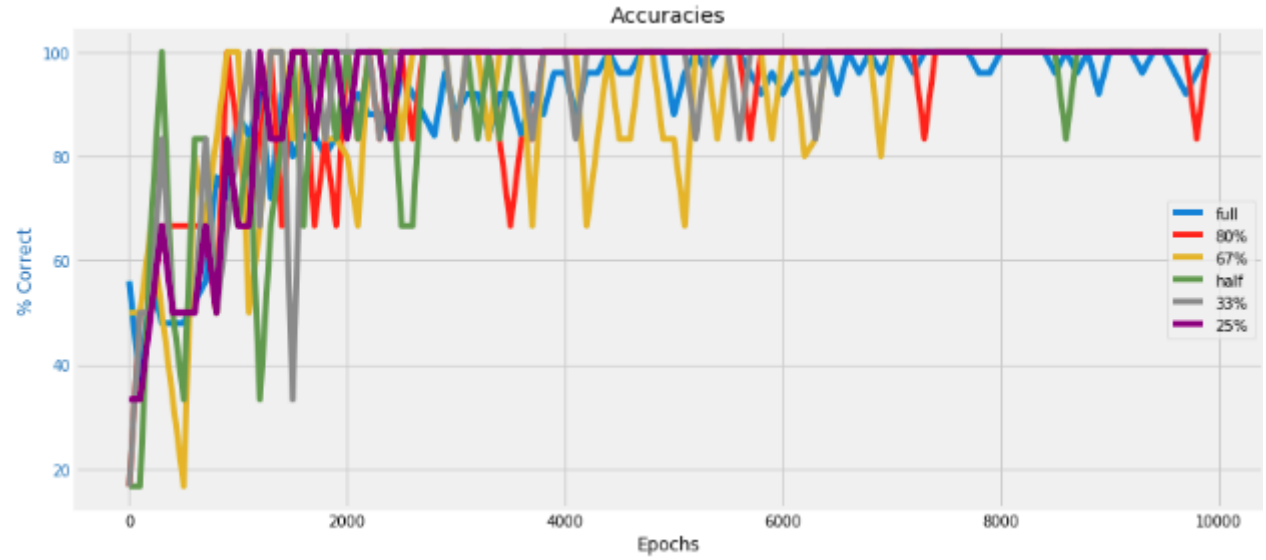

Fig. 8. Prediction accuracy during training of a network using a full-sized training dataset (70% of total rows), or subsets with varying proportions of data discarded, with batch size of 6. This is a representative sample of 10 runs.

3.4 Neurotrophy

A limitation of heuristic pattern reduction is that the virtual clusters that training samples are placed into are equally sized and determined merely on the basis of adjacency. It is conceivable however that relaxing this requirement to allow for more complex patterns of sample inclusion or exclusion may further improve performance, allowing either faster training or maintaining performance on even smaller training sets.

To this end, 10 “neurotroph” agents were initialised with uniform random DNAs determining which training samples to include, and random mutation rates (which could in turn mutate up or down). They each had a neural net partially trained (for 2000 batches only, rather that 10,000 epochs), then were assessed for performance on a 30% holdout set. The best performer was retained, the worst was discarded, and an addition 4 were selected for retention using fitness proporionate selection. Each of these surviving agents was used to spawn a child with mutated DNA to restore the population. This process was repeated for 15 generations, and the results plotted (**fig. 9**).

Several predictions were made: i) that average population accuracy would increase over generations, ii) that heterozygosity would decline and iii) that mutation rates would tend to decline as “fitter” DNAs were discovered. Heterozygosity is the degree of variety in the state of a given DNA base pair. If all neurotrophs in the population have the same state (either positive or negative), heterozygosity at that base pair is 0. Conversely, an equal split in the population at a given base pair results in a heterozygosity value of 1, with intermediate results scaled linearly in this range.

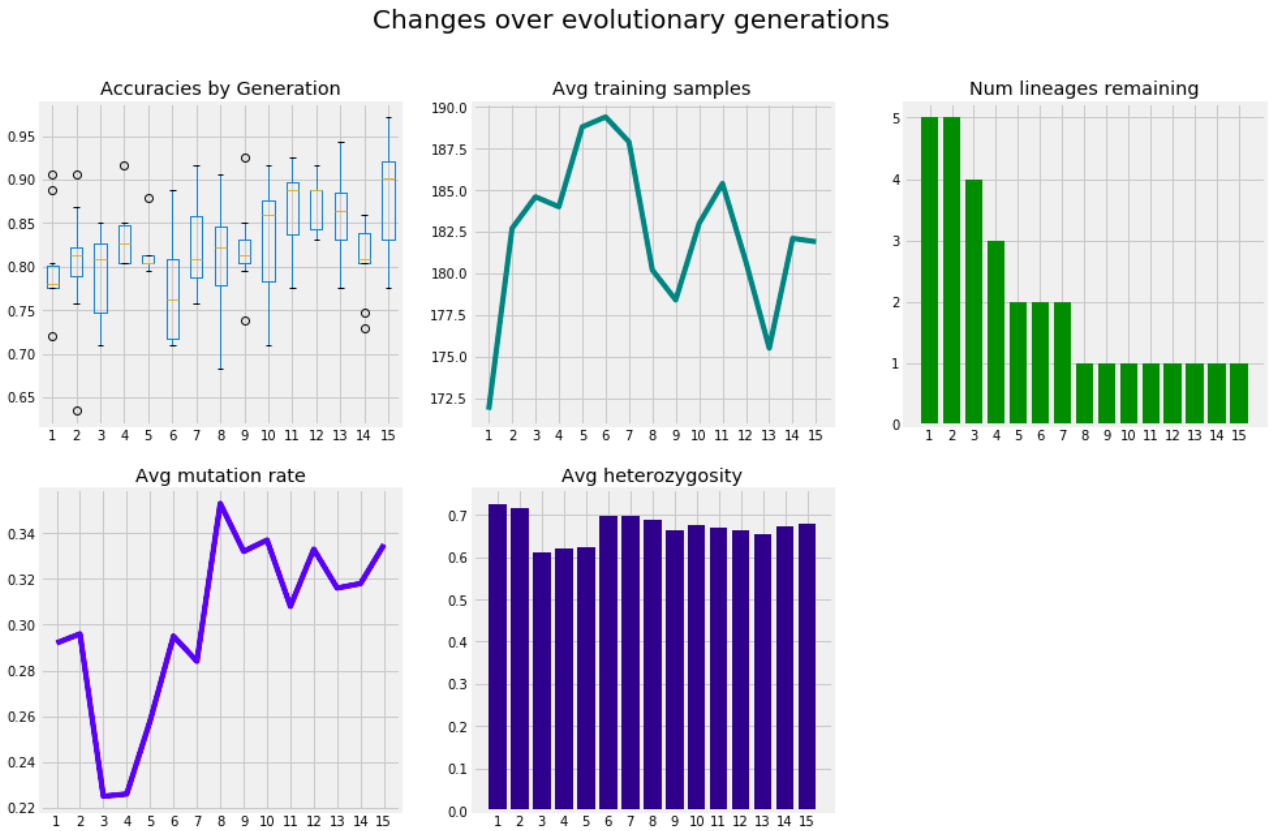

Fig. 9. Example of evolutionary performance over one run of 15 generations. In order: A) boxplot of population accuracies, B) average number of “positive” DNA bases per genome, determining the number of training samples to include, C) the number of lineages surviving from the original 10 neurotrophs, D) average mutation rate, average heterozygosity

Accuracy did show a noisy upward trend over time, however predictions ii) and iii) were not born out by the evidence. Heterozygosity barely changed on average, in one run starting a little above 0.7 (i.e. high variability about the state of each DNA base pair) and finished at 0.68. Mutation rate appeared to fluctuate freely, being able to greatly rise and fall throughout generations. The number of training samples also rose and fell seemingly stochastically. As expected, the number of surviving lineages quickly declines, generally with only a single lineage surviving by 10 generations.

Extensive further research would be required to understand the basis of all these observations. However, several possible explanations present themselves. Firstly, there may be very little difference in the information value of most training samples—in other words, low selective pressure. This would allow maintenance of a high degree of heterozygosity, and allow relative freedom in the size of positive DNA base pairs. Another consideration is mere training time. If mutation rates are initialised too high, the system may be quite chaotic, and it would take a certain number of generations for highly performing agents to evolve children with low mutation rates to maintain their DNA into future generations. An additional consideration is the population size and selective mechanism. Because fitness proportionate selection is not deterministic, there is a risk every generation of discarding highly-performing agents and retaining weaker ones. Due to the relatively minor differences in accuracies (generally around 10-20% between the best and worst agent in a population), poor selective decisions will be relatively common. The small population size (5 after each culling) means that large swings can happen each generation.

After 15 generations, a consensus DNA sequence was constucted by comparing the genomes of all surviving neurotrophs. Across several runs, the consensus genome often included 70-75% of the full training set, though with a fair bit of variation in this.

This consensus sequence was used with k-fold cross validation to train networks, and compare them to networks trained using the full dataset. As was the case for heuristic pattern reduction, no difference is seen in loss or accuracy during training (**fig. 10**). Average accuracies of the full dataset and neurotroph fully trained networks were statistically indistinguishable, 98.9% and 98.5% respectively.

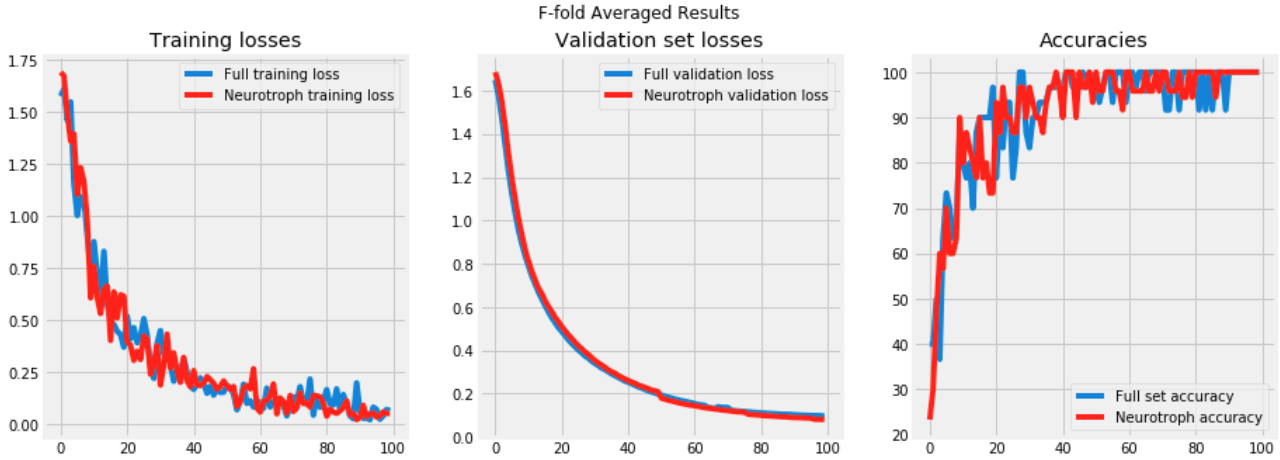

Fig. 9. Training loss, validation set loss and accuracy during training, comparing networks training on the full dataset versus on a neurotrophically-selected subset of the data, often comprising ~75% of the full set.

4 Conclusions and Future Work

The dermatology dataset presents a fertile, if possibly overly redundant opportunity to explore the use of various neural networks and pattern reduction approaches. The standard feed-forward network performed commendably with minimal hyperparameter tweaking, achieving a 97.7% average correct classification rate in one 10-fold run. Pleasingly, this exceeds accuracy of early approaches used in the literature such as k-Means; though unsurprisingly, is inferior to several highly refined models that have achieved upward of 99% accuracy on this dataset.

The heuristic pattern reduction implementation mirror general findings from the literature, demonstrating the large portions of the trainin data can be discarded (possibly up to 75%) without adversely affecting accuracy of the trained network. I did not however find any evident of improved training time on this dataset, possibly pointing to a lack of strong variation in the distinctiveness or unique information value of different training samples.

Surprisingly noisy dynamics were obtained from the experimental “neurotroph” investigation. The maintenance of a high level of heterozygosity is surpring as, even in the absence of any selection, high population turnover and regular founder effect should result in a rapid loss of variability in a population. While the approach developed here ultimately yielded a 25% reduced dataset with comparable performance as training on the full set, there is little reason to think the reduced set was superior to a random subset of the same size.

Several interesting possibilities present themselves as avenues for future work. Firstly, it would be valuable to extend this appoach to more complex datasets, where the effect of pattern reduction may be far more pronounced. For example, even monochrome 6x6 pixel image data would be far more complex and varied than the samples used in this dataset. Therefore for image or other feature-rich data, the effect of removing certain patterns is expected to be much more impactful.

A high degree of refinement is clearly possible in the evolutionary approaches and hyperparameters used. As an example, it may be advantageous to begin with a relatively larger population of agents, and reduce this number over generations as with simulated annealing. There is a question mark around the value of having a larger population versus running more generations: is greater refinement of favourable DNAs or exploring whole new regions of the search space more valuable? There is also a trade-off between discarding valuable variation too soon versus selecting too weakly, which risks diluting the evolutionary process. For this dataset, it seems possible that stricter selection would’ve been valuable, e.g. deterministically retaining the highest accuracy neurotrophs each generation. Additionally, it is likely that initial mutation rates and DNA distributions could be optimised, perhaps having lower initial mutation rates and exploring smaller positive DNA starting sizes, to extract more predictive work out of individual DNA bases. Optimising these parameters and approaches while maintaining a viable time complexity, presents a significant challenge.

As a final consideration, in this study all 34 variables of the dermatology dataset were used for training network models and making inferential predictions. It is however worth noting that only 12 of the 34 variables are obtainable from straightforward clinical evaluations. The remaining 22 derive from microscopic analysis of a histopathological biopsy, a resource and time-intensive activity. It would obviously be highly advantageous to be able to make robust classifications of erythemato-squamous diseased based on just the clinically obtainable variables. Perplexingly, despite the high degree of attention this dataset has received from machine learning publications, few (if any) studies have attempted classification using only these clinical variables. This oversight speaks to the importantance of interdisciplinary teams: in this case, data scientists and medical professionals . A first priority for further investigations would therefore be to attempt classification using this restricted subset of variables, which stands to greatly bolster real-world applicability of any approaches developed.

References

1. Ubeyli E., Doğdu E.: Automatic detection of erythemato-squamous diseases using k-means clustering. J Med Syst. 34(2), 179--184 (2010)
2. Menai M., Altayash N.: Differential Diagnosis of Erythemato-Squamous Diseases Using Ensemble of Decision Trees. In: Part II of the 27th International Conference on Modern Advances in Applied Intelligence IEA/AIE, vol. 8482, pp. 369–377. Springer-Verlag New York, Inc. (2014)
3. Badrinath, N., Gopinath, G., Ravichandran, K.S.: Design of Automatic Detection of Erythemato-squamous Diseases Through Threshold-based ABC-FELM Algorithm. J Artificial Intelligence, 6: 245--256 (2013)
4. Badrinath, N., Gopinath, G., Ravichandran, K.S., Soundhar, R.: Estimation of automatic detection of erythemato-squamous diseases through AdaBoost and its hybrid classifiers. Artif Intell Rev. 45--471 (2016)
5. Fidelis, M.V., Lopes, H.S., Freitas, A.A.: Discovering comprehensible classification rules with a genetic algorithm. Evolutionary Computation,. Proceedings of the Congress (2000)
6. Dermatology Data Set. http://archive.ics.uci.edu/ml/datasets/Dermatology
7. Bergstra J., Bengio Y.: Random Search for Hyper-Parameter Optimization. J Machine Learning Research. 13, 281-305 (2012)
8. Gedeon, T.D., Bowden, T.G.: Heuristic Pattern Reduction. International Joint Conf. on Neural Networks, Beijing, vol. 2: 449-453 (1992)
9. The Mayo Clinic. https://www.mayoclinic.org/