Improved Music Feature Learning with Deep Neural Networks

Siddharth Sigtia and Simon Dixon

{sss31,simond}@qmul.ac.uk

Centre for Digital Music
Queen Mary University of London

Motivation

- Try to learn the most optimal features for a particular task and reduce dependency on hand-crafted features.
- How can we learn features for a particular task?: Neural nets with several hidden layers (deep neural networks).
- Can we learn features for MIR tasks with neural nets?: Lots of recent evidence suggests yes!

centre for digital music

Queen Mary University of London

Challenges with this approach?

- Optimising networks with several hidden layers is challenging.
- The error surface is highly non-linear w.r.t. parameters and the best we can do is hope to find a useful local minimum.
- The number of hyper-parameters can be quite large if we include momentum, learning rate schedules etc.
- For large networks, Stochastic Gradient Descent (SGD) can take prohibitively long to find useful minima even with unsupervised pre-training.
- In several domains (including music/audio), it is quite important to understand/interpret the learnt features. Something that is not clear with deep neural nets.

Can we do better?

- The use of neural networks for supervised learning has come full circle in some ways.
- Unsupervised pre-training is not considered to be necessary for finding good solutions.
- Gradient based optimisers starting with random parameter initialisation provide good results.
- Rectified Linear Units (ReLUs), Dropout, Hessian Free (HF) optimisation, Nesterov's Accelerated Gradient have all been applied to problems in various domains.
- The application of these new techniques to learning features for MIR tasks could provide improvements over existing methods.

Problem definition

- Learn features for a genre classification task using data from the GTZAN dataset.
- Train a classifier on the learned features and evaluate system performance.
- Inspect if features are general by using the same features on the ISMIR2004 genre dataset.

This area will contain a diagram of the general pipeline

Contributions Of The Paper

- Evaluate the use of ReLUs as hidden units.
- Use Dropout for regularisation.
- Use HF Optimisation for training sigmoid nets and compare.

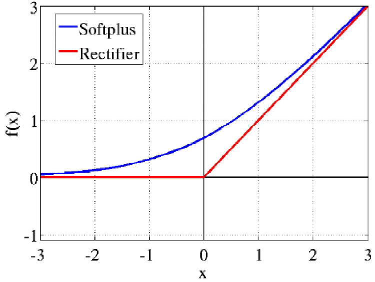
Hypothesis?

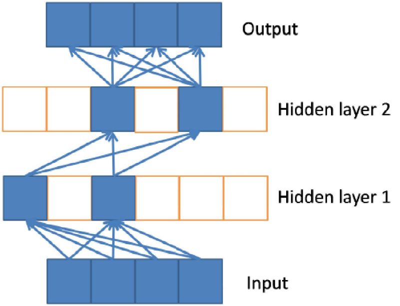
- ReLUs+Dropout eliminate the need for pre-training.
- Performance(ReLU+Dropout+SGD) $>=$ Performance(sigmoid nets+SGD)
- More efficient training of sigmoid nets with HF.

Feature Extraction

This slide is going to contain a pictorial representation of the feature extraction.

Rectified Linear Units



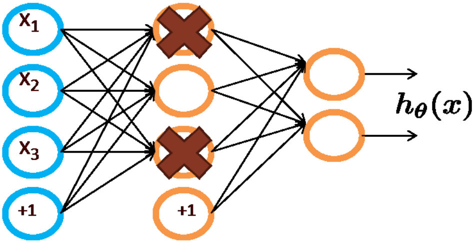


Activation function: $f(x) = max(0, x)$

Useful Properties of ReLUs

- No need for supervised pre-training.
- Hard sparsity in hidden layers.
- Gradients flow easily.
- Error surface is less convoluted w.r.t parameters because of the form of the activation function.

Dropout



Forward Propagation

$$y^l = \frac{1}{1-p} W^l (r^{l-1} * y^{l-1} + b^l)$$