Leabra Networks

Applications to Machine Learning

Daniel Schultz

Innovative Computing Laboratory
Electrical Engineering & Computer Science
University of Tennessee, Knoxville

Computational Cognitive Neuroscience, 2019

Outline

- Introduction
  - Leabra Model
  - Multi-layer Perceptron (MLP)
  - Motivation
  - Handwritten digits
- Methodologies
  - Data preprocessing
- Performance
  - Methods

Leabra Model

Review

- Designed to be biologically realistic model of the neocortex.
- Combination learning mechanism
  Self-organized learning
  Error-driven learning
- Described as best combination for studying the brain

Multi-layer Perceptron (MLP)

background

- Type of artificial neural network used in machine learning
- Error-driven learning using back-propagation
- 3 layers: Input, Hidden, Output
  - Hidden layers usually implemented as a single hidden layer
- Easy to implement and used for performance comparison
- Common in the 1980's and the baseline for our problem.

Motivation

- Human brain can always recognize digits
- Orientation, size, obstruction, etc are irrelevant
- No need for millions of test cases to ”learn”
- Why cant machine learning do this?

Questions to Answer

1. Why use Leabra as a machine learning algorithm?
2. How is Leabra different then other artificial neural networks?
3. Is Leabra better at learning some tasks?

MNIST

- MNIST dataset of digits sourced from NIST dataset

MNIST

- MNIST dataset of digits sourced from NIST dataset
- Training Set: 60,000 images.
  Testing Set: 10,000 images.
  Normalized & centered on a fixed image size
  28x28 pixels

Digit Manipulations

- Artificial neural networks are very good at learning a known set.
- Classification accuracy for a known problem is high.
- Not as good for test cases not conforming to the training set.
- Generally can't solve problems not trained on. Can be tricked even:

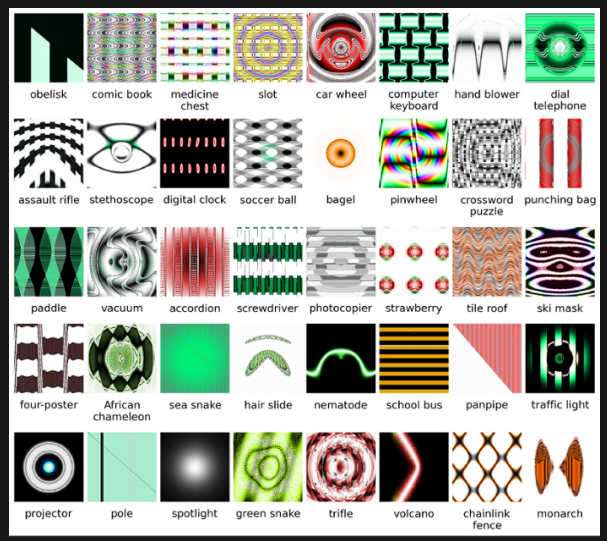


Figure: DNN classifications with greater than 99% confidence. (Nguyen A)

Digit manipulations

- 3 distinct data sets are generated from MNIST

Digit manipulations

- 3 distinct data sets are generated from MNIST
- Unmodified Set

Digit manipulations

- 3 distinct data sets are generated from MNIST
- Unmodified Set
- Skewed Set
  - Random Rotation
  - Affine Transformation
  - Horizontal/Vertical Flip
  - Scaling

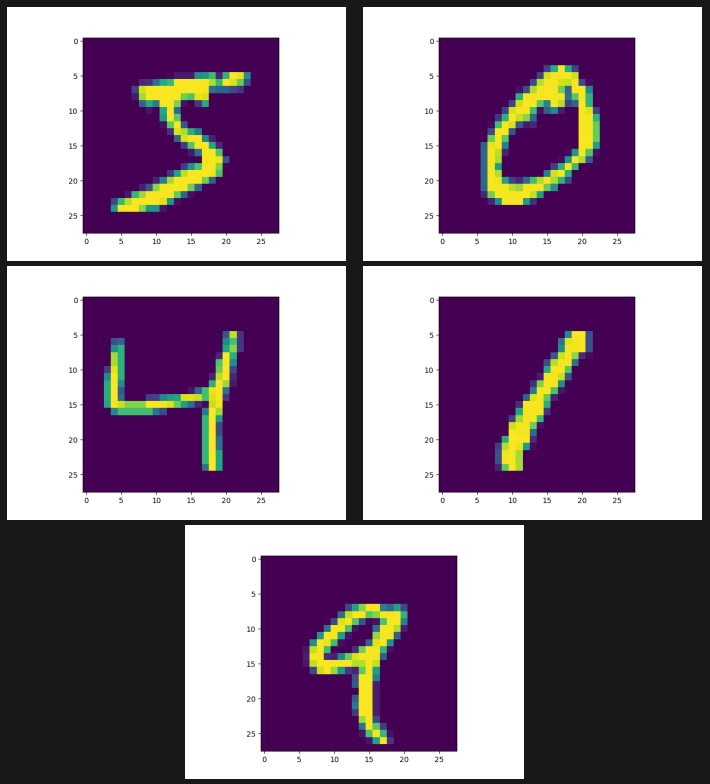
Digit manipulations

- 3 distinct data sets are generated from MNIST
- Unmodified Set
- Skewed Set
  - Random Rotation
  - Affine Transformation
  - Horizontal/Vertical Flip
  - Scaling
- Combination Set (roughly 50%)

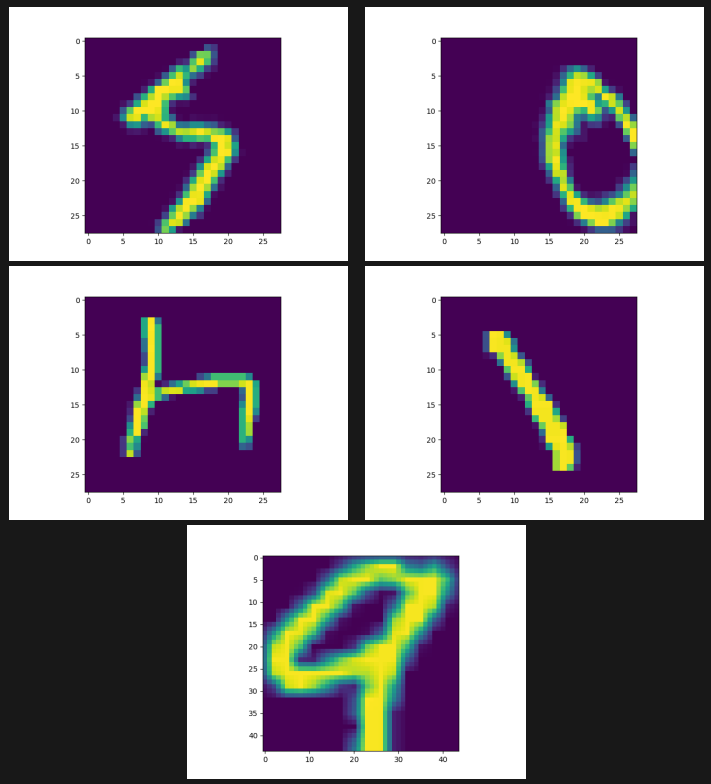
Digit manipulations

- 3 distinct data sets are generated from MNIST
- Unmodified Set
- Skewed Set
  - Random Rotation
  - Affine Transformation
  - Horizontal/Vertical Flip
  - Scaling
- Combination Set (roughly 50%)
- Training, Validation, Test subsets of each

Digit Manipulations



(a) Unmodified Digits



(b) Modified Digits

Figure: Example of Digit Manipulation

Methods

- Compare performance of Leabra learning against the multi-layer perceptron (MLP).
  3 data sets
  Normal (Unmodified)
  Skewed
  Combination

Methods

- Compare performance of Leabra learning against the multi-layer perceptron (MLP).
  3 data sets
  Normal (Unmodified)
  Skewed
  Combination
- 3 separate data sets
  3 trained networks
  3 data sets to gather performance data for each network

Methods

- Compare performance of Leabra learning against the multi-layer perceptron (MLP).
  3 data sets
  Normal (Unmodified)
  Skewed
  Combination
- 3 separate data sets
  3 trained networks
  3 data sets to gather performance data for each network
- 3 Leabra networks, 3 MLP networks
  3 training phases each
  3 data sets, 3 modes (Training/Validation/Testing), 3 networks
  27 performance metrics per network type, 54 total.

SSE Results

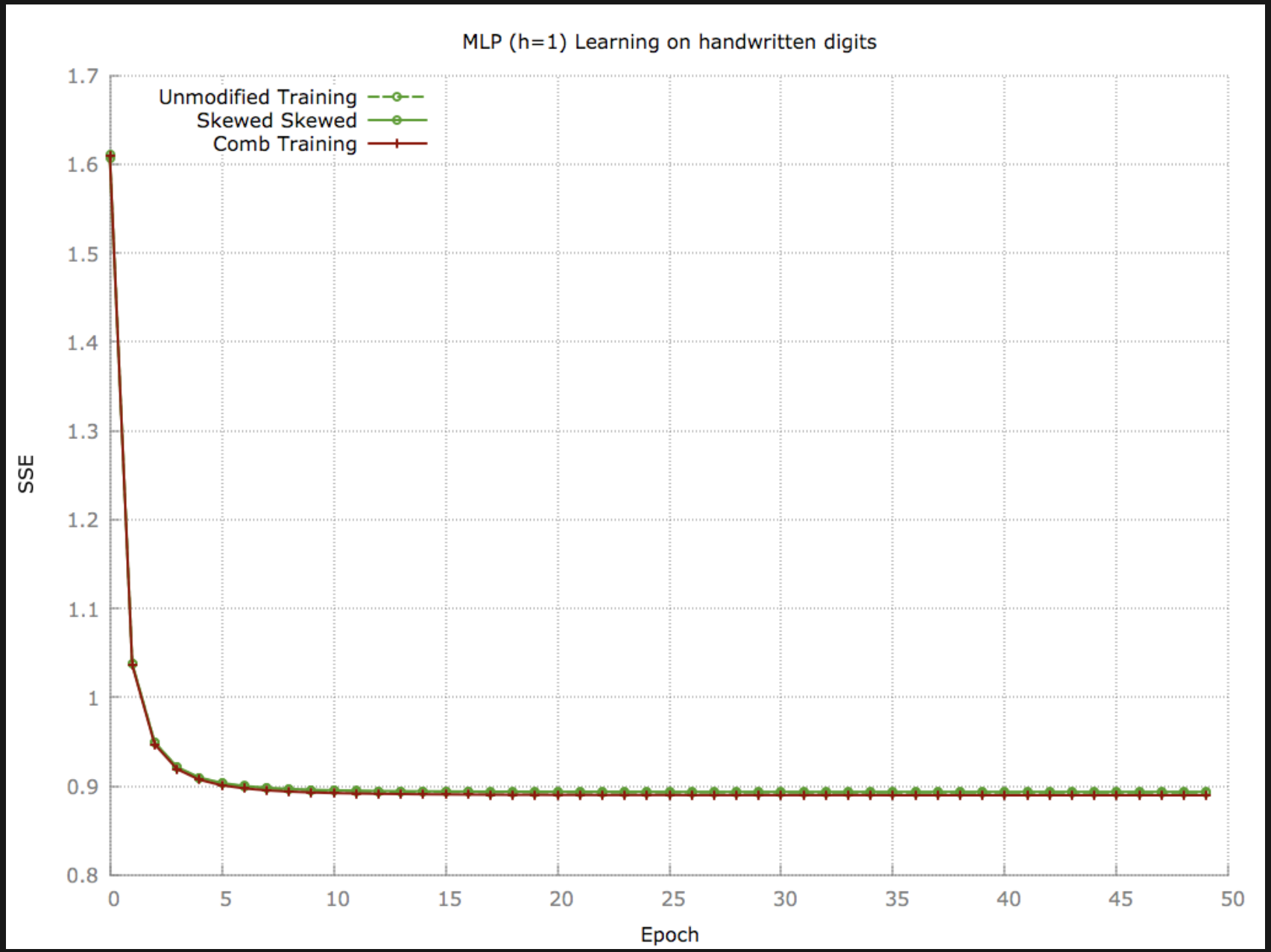
MLP(1 hidden layer)									
Group	NT	NV	NTsT	ST	SV	STsT	CT	CV	CTsT
Norm	0.89	0.9	0.91	0.92	0.9	0.91	0.91	0.9	0.9
Skew	0.92	0.91	0.91	0.89	0.91	0.92	0.92	0.91	0.91
Comb	0.91	0.91	0.92	0.92	0.92	0.91	0.89	0.91	0.91

(a) MLP ($h = 1$) SSE Performance

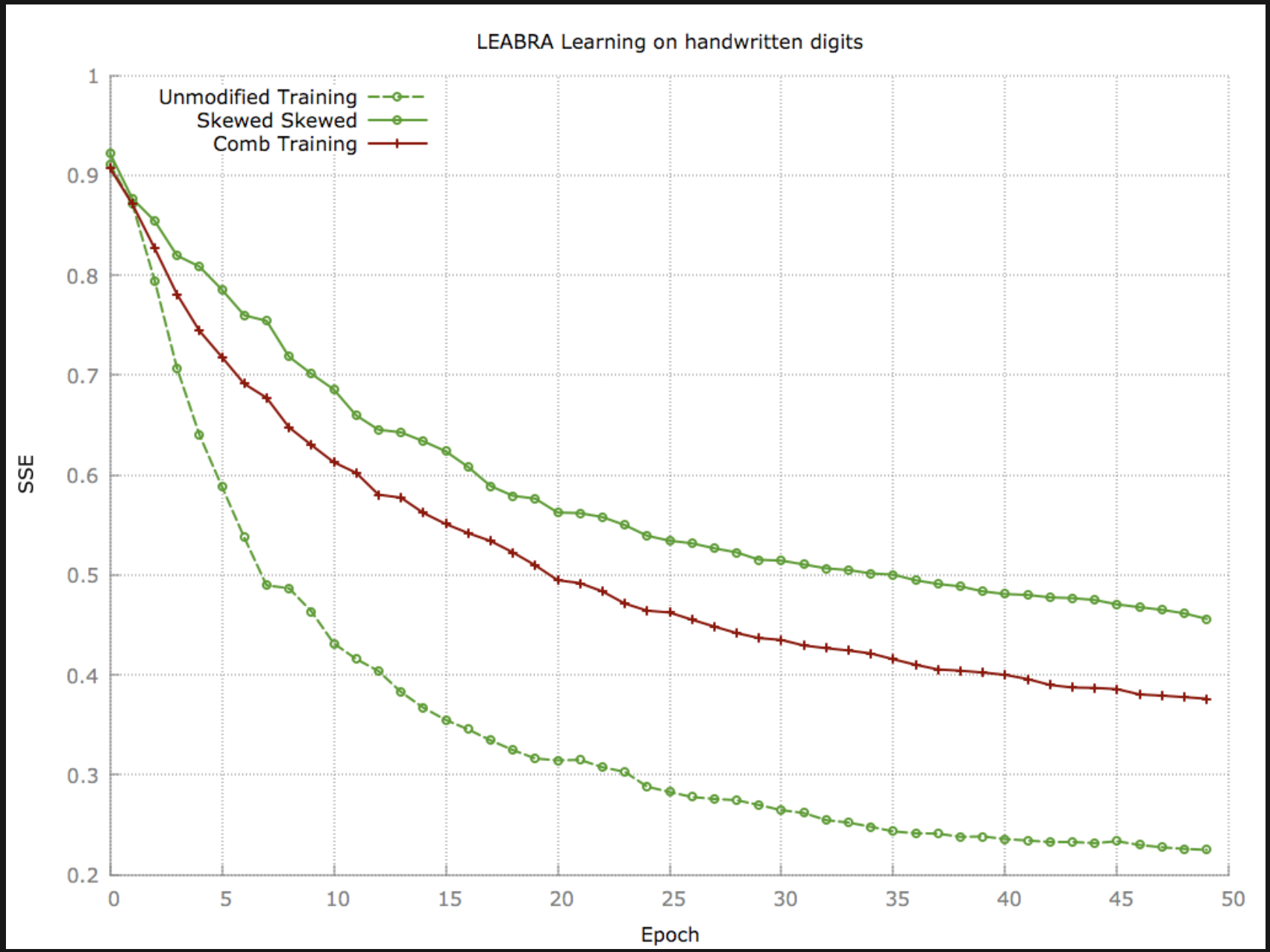
LEABRA (1 Hidden Layer)									
Group	NT	NV	NTsT	ST	SV	STsT	CT	CV	CTsT
Norm	0.40	0.36	0.54	0.87	0.84	0.85	0.68	0.68	0.74
Skew	0.83	0.75	0.72	0.52	0.78	0.81	0.74	0.75	0.77
Comb	0.66	0.65	0.65	0.80	0.79	0.83	0.49	0.73	0.76

(b) Leabra SSE Performance

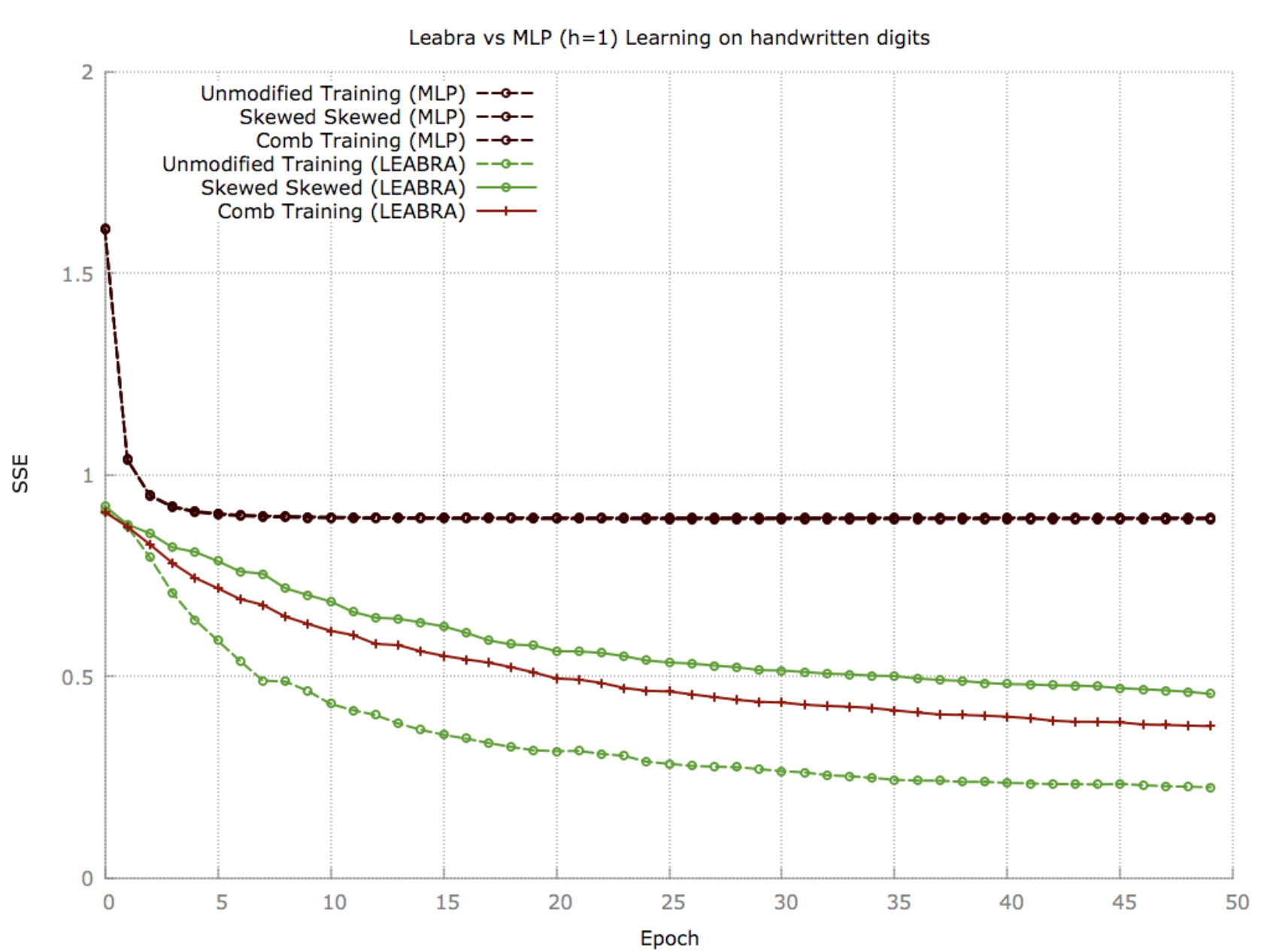
MLP Training



LEABRA Training



Comparison


Leabra vs MLP (h=1) Learning on handwritten digits
Unmodified Training (MLP)
Skewed Skewed (MLP)
Comb Training (MLP)
Unmodified Training (LEABRA)
Skewed Skewed (LEABRA)
Comb Training (LEABRA)
SSE
Epoch
0
0.5
1
1.5
2
0
5
10
15
20
25
30
35
40
45
50

Summary

1. Why use Leabra as a machine learning algorithm?

2. How is Leabra different then other artificial neural networks?

3. Is Leabra better at learning some tasks?

Summary

1. Why use Leabra as a machine learning algorithm?
   - Can learn a problem just as well or better than artificial neural networks used in machine learning.
2. How is Leabra different then other artificial neural networks?

3. Is Leabra better at learning some tasks?

Summary

1. Why use Leabra as a machine learning algorithm?
   - Can learn a problem just as well or better than artificial neural networks used in machine learning.
2. How is Leabra different then other artificial neural networks?
   - Doesn't rely entirely on error-driven learning.
   - Integrates a form of principal component analysis.
3. Is Leabra better at learning some tasks?

Summary

1. Why use Leabra as a machine learning algorithm?
   - Can learn a problem just as well or better than artificial neural networks used in machine learning.
2. How is Leabra different then other artificial neural networks?
   - Doesn't rely entirely on error-driven learning.
   - Integrates a form of principal component analysis.
3. Is Leabra better at learning some tasks?
   - Preliminary results show Leabra having roughly $\frac{1}{2}$ SSE compared to MLP over the same number of epochs & layers.

Conclusions

Leabra Networks

+ Better Inital SSE Performance
+ Better Overall SSE Performance
- Larger Memory Footprint
- High Computational Workload

MLP Networks

+Less Memory Footprint
+ Less Computationally Intensive
- Worse Initial SSE Performance
- Worse Overall SSE Performance
- Not good on non-trained problems

Questions?

References

1. Leabra PhD Thesis
   - https://www.researchgate.net/profile/Randall_OReilly/publication/270960245_The_Leabra_Model_of_Neural_Interactions_and_Learning_in_the_Neocortex/links/0fcfd5142b925b81de000000.pdf
2. Emergent Neural Network Simulator
   - https://grey.colorado.edu/emergent/index.php/Main_Page
3. Leabra Python Implementation
   - https://github.com/benureau/leabra
4. MNIST Dataset
   - http://yann.lecun.com/exdb/mnist/