GenrELM: An ELM Approach to Automatic Music Genre Classification using Bass Lines

J. Gabrielle M. Macaraig

March 13, 2013

Outline

- Introduction
- Literature Review
- Statement of the Problem
- Objectives
- Proposed Approach
- Experiments and Results
- Conclusion and Future Works


Introduction

Music

- "the science or art of ordering tones or sounds in succession, in combination, and in temporal relationships to produce a composition having unity and continuity" *(Merriam-Webster)*

Elements of Music

- Rhythm
- Melody
- Harmony
- Texture
- Dynamics
- Tone Color

Elements of Music

- Rhythm
  - element of “time”
  - beat

Elements of Music

- Melody
  - horizontal representation of pitch
  - tune

Elements of Music

- Harmony
  – vertical representation of pitch

Elements of Music

- Texture
  - number of individual melodies and their relationships
    - monophonic
    - polyphonic
    - homophonic

Elements of Music

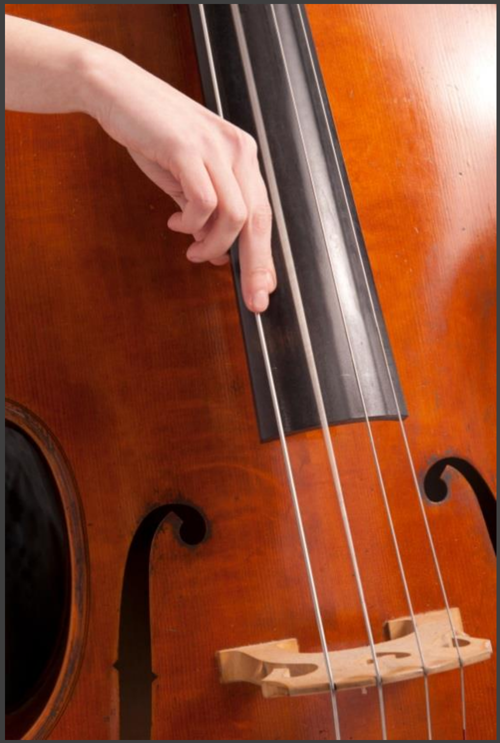
- Dynamics
  – relative loudness and softness of music

Elements of Music

- Tone Color
  - timbre/ quality of sound

Bass Lines

- a monophonic instrumental melody
- played by instruments having a low-pitched sound (*i.e. bass guitar, double bass*)



Bass Lines

- bridges together melodic and rhythmic sections in most musical styles
- "...the groundwork or foundation upon which all musical composition is to be erected." (Christopher Simpson)

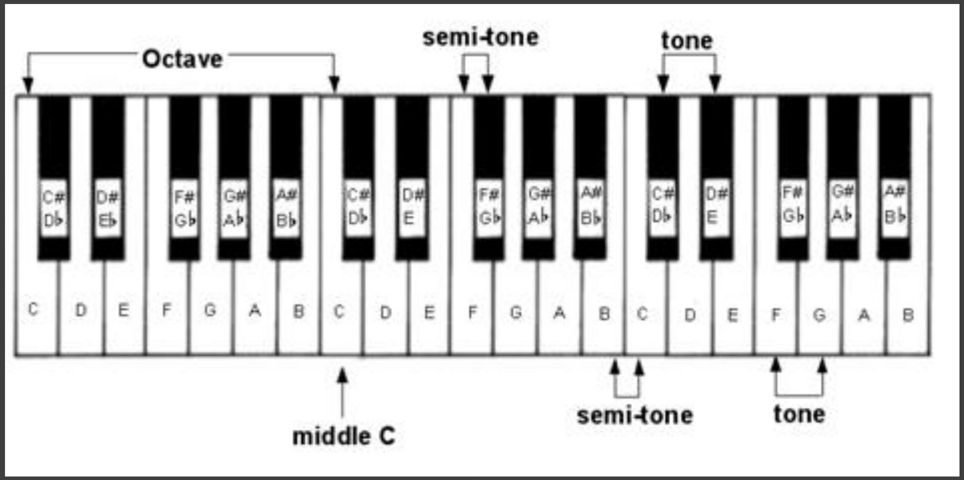
Bass Lines

- can establish rhythm
- has tune
- does not harmonize in itself alone
- monophonic
- low-pitched sound

Melodic Intervals

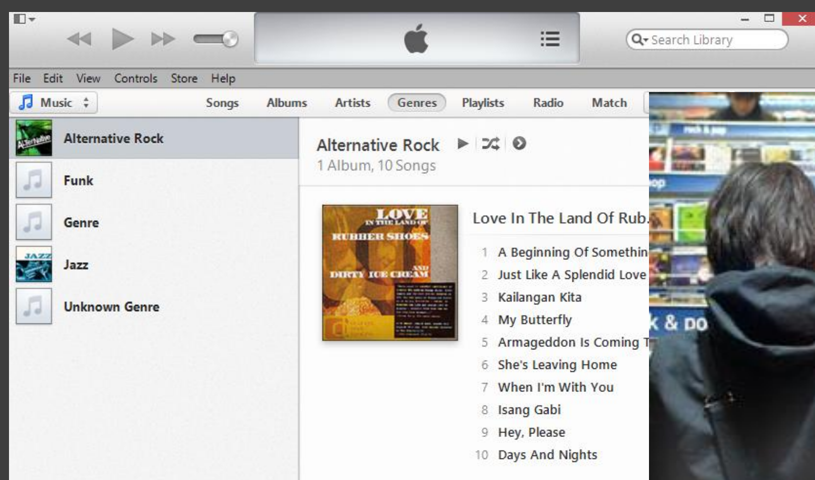
- represents the number of semitones in a given time interval

  semitone – half-step-distance between two adjacent notes



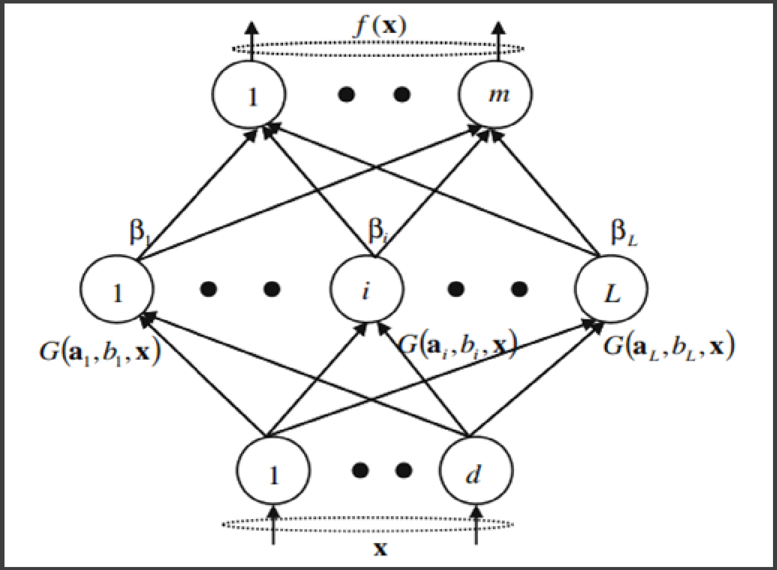
Genre

- a way of organizing, classifying, and grouping music
- a kind of music (Franco Fabbri)



Extreme Learning Machine (ELM)

- machine learning algorithm
- created for single-hidden layer feed forward networks



Extreme Learning Machine (ELM)

- tries to overcome the challenges encountered among traditional computing techniques such as SVMs and NNs
  - slow learning speed
  - over training
  - local optima entrapment

Extreme Learning Machine (ELM)

Given a training set , hidden node output function , and hidden node number *L*,

step 1 Randomly generate hidden parameters (a_i, b_i), *i* = 1,…,*L*.

step 2 Calculate the hidden layer output matrix **H**.

step 3 Calculate the output weight vector **β**:

$$\boldsymbol{\beta} = \mathbf{H}^{\dagger}\mathbf{T}, \text{ where}$$

$$\mathbf{T} = [\boldsymbol{t}_1, \dots, \boldsymbol{t}_N]^{\mathrm{T}}.$$

Literature Review

Literature Review

- Karpov, I.: *Hidden Markov Classification for Musical Genres* (2002)
- McKay, C. and Fujinaga, I.: *Automatic Genre Classification using Large High-Level Musical Feature Sets* (2004)

Literature Review

- Haggblade, M., Hong, Y., and Kao, K.: *Music Genre Classification* (2011)
  - K-Nearest Neighbour, K-means, multiclass SVM, and neural networks

Literature Review

- Meng, A., et al.: *Temporal Feature Integration for Music Genre Classification* (2007)
- Şimşekli, U.: *Automatic Music Genre Classification using Bass Lines* (2010)
  - K-nearest neighbor classifier

Statement of the Problem

Statement of the Problem

- to create a platform for automatic music genre classification using information on bass lines

Objectives

Objectives

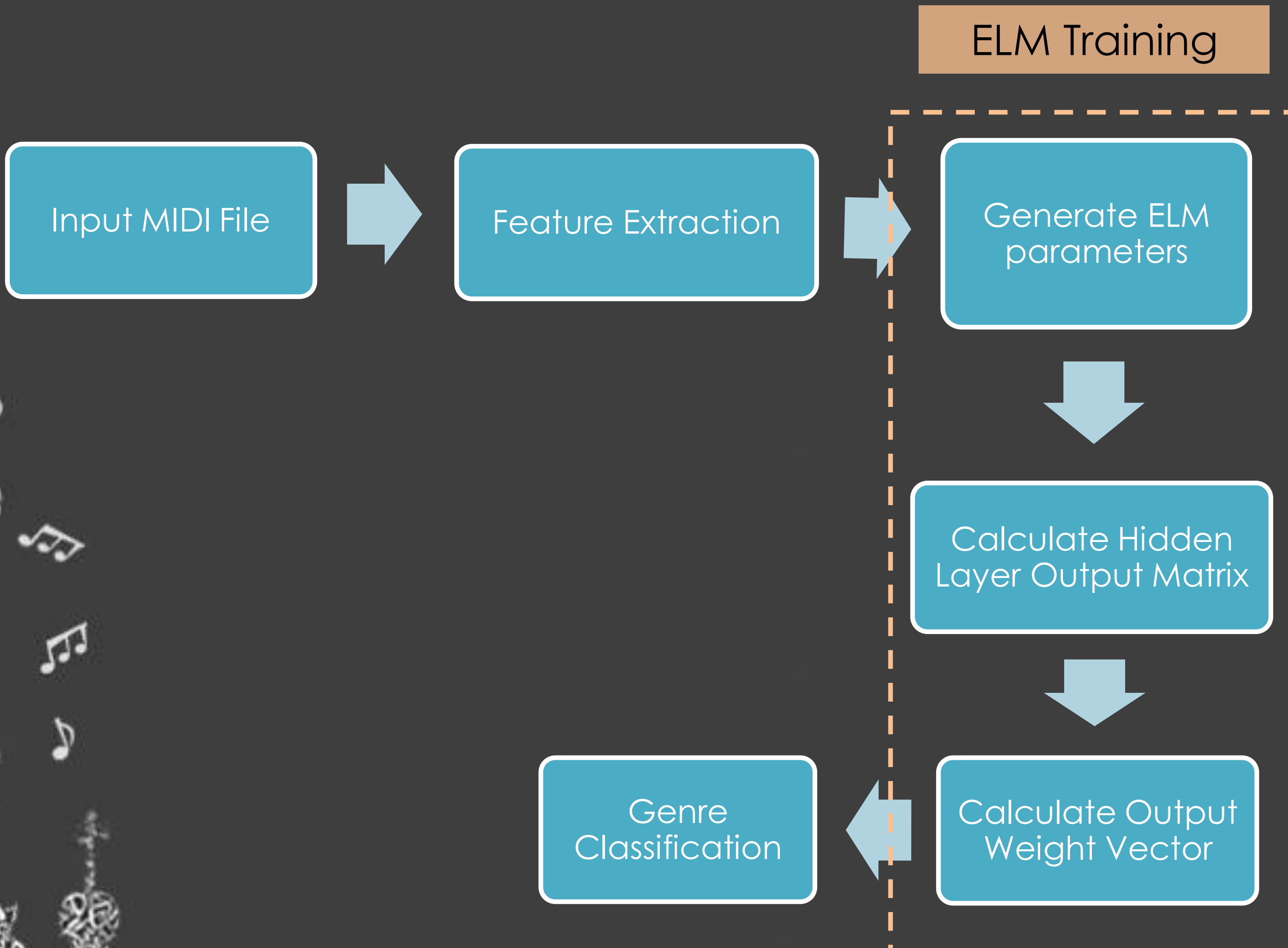
- to apply the ELM algorithm to music genre classification
- to evaluate the performance of ELM in genre classification by varying the number of its hidden nodes,

Objectives

- to evaluate the performance of ELM in genre classification by introducing a regularization coefficient, and
- to evaluate the performance of ELM in genre classification by altering the network structure.

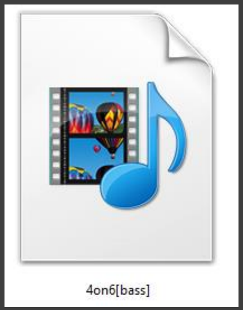
Proposed Approach

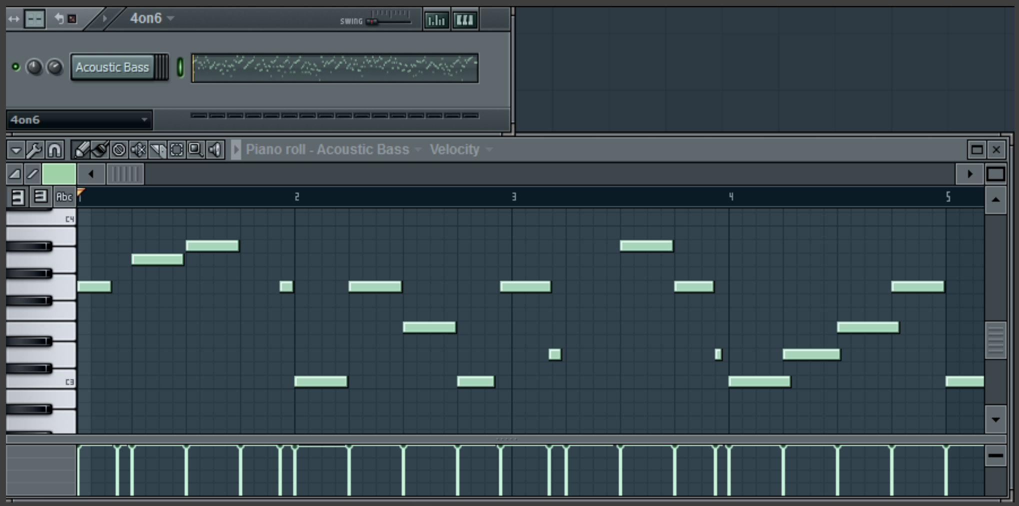
General Flow



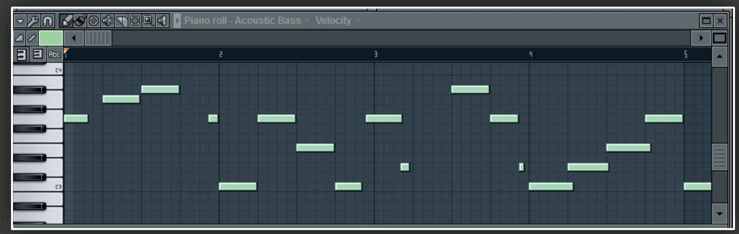
Input

- MIDI file containing bass lines only
- Any length

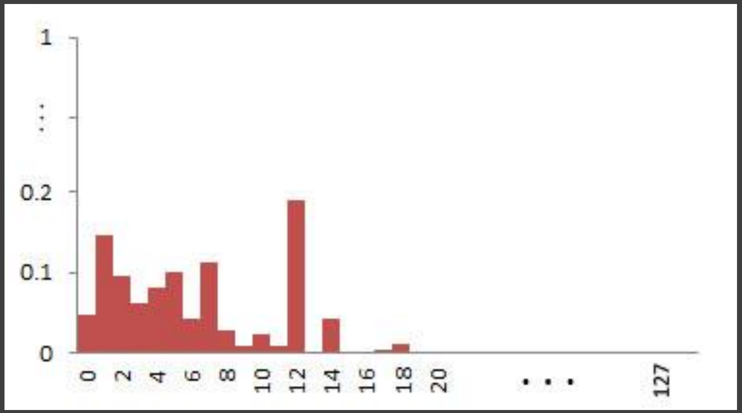




Feature Extraction



MIDI Bass Line



Melodic Interval Histogram

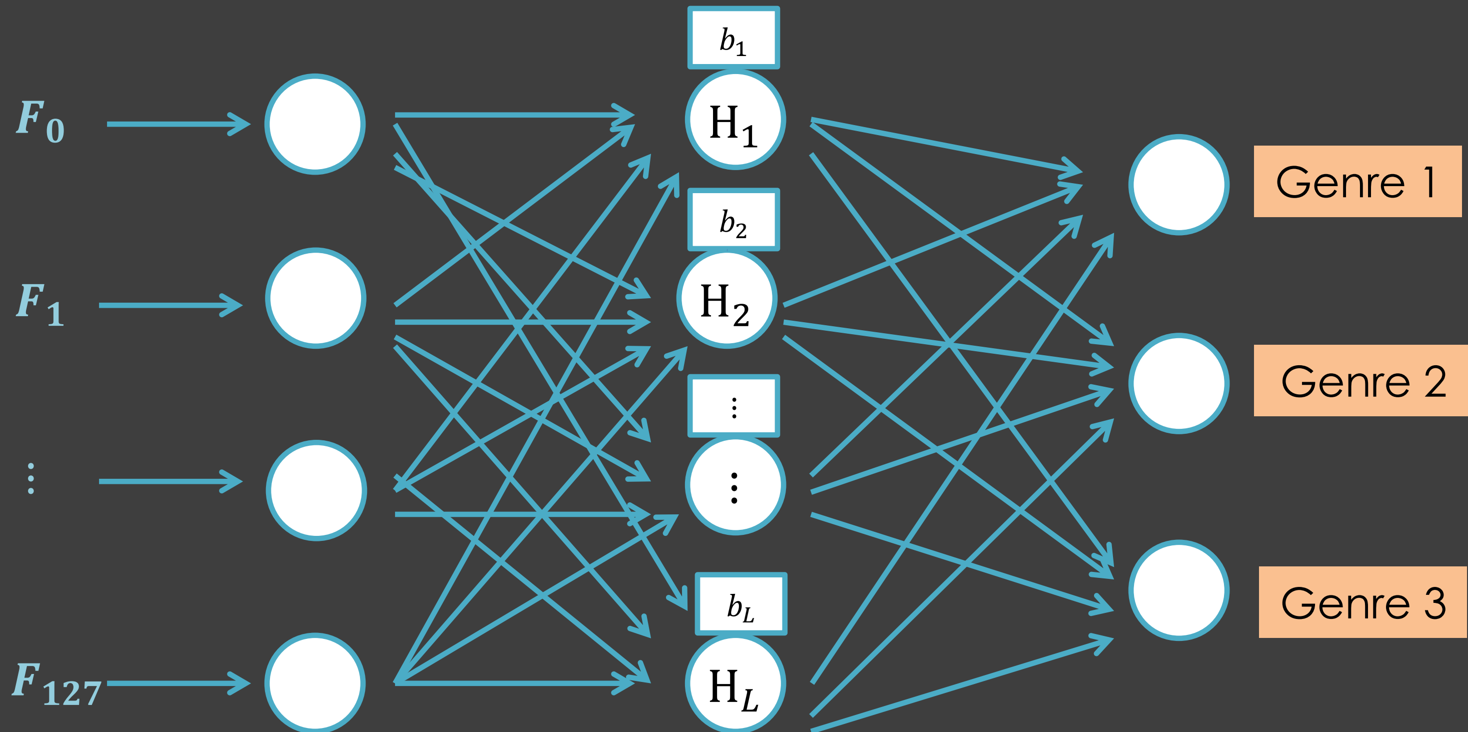
0.046	0.147	0.097	0.062	0.081	0.100	0.042	0.112	0.027	0.008	0.023	0.008	0.189	0	0.042	0	0	0.004	0.012	0	0	0
0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	...	127

feature vector (*F*)

Genre Taxonomy

Jazz	Rhythm & Blues	Rock	- root genre
Bebop	Blues Rock	Hard Rock	{ leaf genres
Swing	Funk	Metal	
Bossa Nova	Rock 'n Roll	Alternative Rock	

ELM Training



L - no. of hidden nodes

ELM Training

- Generate Hidden Node Parameters
  - random-generated integers from -1 to 1
  - input layer to hidden layer

$$a = \begin{bmatrix} -0.804 & \cdots & -0.855 \\ \vdots & \ddots & \vdots \\ -0.299 & \cdots & -0.181 \end{bmatrix}_{d\ x\ L}$$

input weights

$$b = \begin{bmatrix} 0.528 \\ \vdots \\ -0.269 \end{bmatrix}_{L}$$

bias

d – input vector length
L – no.of hidden nodes

ELM Training

- Calculate Hidden Layer Output Matrix

$$H = \begin{bmatrix} G(\boldsymbol{a}_i, b_i, \boldsymbol{x}_1) & \cdots & G(\boldsymbol{a}_L, b_L, \boldsymbol{x}_1) \\ \vdots & \cdots & \vdots \\ G(\boldsymbol{a}_i, b_i, \boldsymbol{x}_N) & \cdots & G(\boldsymbol{a}_L, b_L, \boldsymbol{x}_N) \end{bmatrix}_{N\ x\ L} \qquad \begin{matrix} j \in 1, 2, \dots, N \\ i \in 1, 2, \dots, L \end{matrix}$$

$$f(x) = \frac{1}{1 + \exp(-w)}$$

activation function

$$w = \left(\sum F_i * a \right) + \mathrm{b}$$

ELM Training

- Calculate Output Weight Vector

$$\boldsymbol{\beta} = \boldsymbol{H}^{\dagger}\,\boldsymbol{T}$$

$$\boldsymbol{T} = \begin{bmatrix} 1 & 0 & 0 \\ \vdots & \cdots & \vdots \\ 0 & 0 & 1 \end{bmatrix}_{N\,x\,m}$$

$$\boldsymbol{\beta} = \begin{bmatrix} \cdots & \cdots & \cdots \\ \vdots & \cdots & \vdots \\ \cdots & \cdots & \cdots \end{bmatrix}_{L\,x\,m}$$

L – no. of hidden nodes
N – training set size
m – no. of output nodes

ELM Training

- w/ regularization parameter
  - any positive number C

$$\boldsymbol{\beta} = \mathbf{H}^{\mathbf{T}}\left(\frac{I}{C} + \mathbf{H}\mathbf{H}^{\mathbf{T}}\right)^{-1}\mathbf{T}$$

ELM Training

- Target Output Mapping

Genre	Node 1	Node 2	Node 3
Jazz	1	0	0
Rhythm & Blues	0	1	0
Rock	0	0	1

Genre (Root: Jazz)	Node 1	Node 2	Node 3
Bebop	1	0	0
Swing	0	1	0
Bossa Nova	0	0	1

ELM Training

- Target Output Mapping (cont'd)

Genre (Root: RnB)	Node 1	Node 2	Node 3
Blues Rock	1	0	0
Funk	0	1	0
Rock 'n Roll	0	0	1

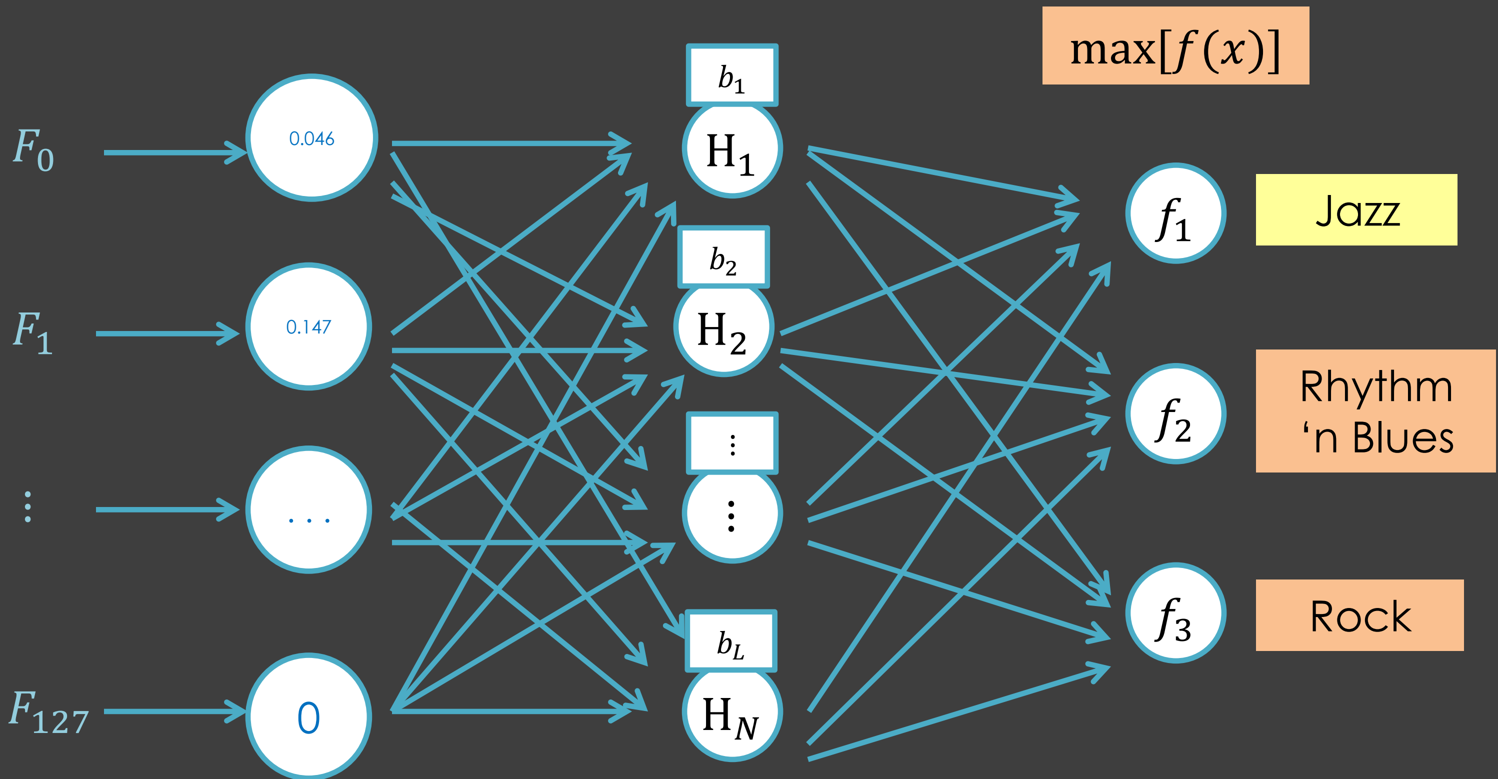
Genre (Root: Rock)	Node 1	Node 2	Node 3
Hard Rock	1	0	0
Metal	0	1	0
Alternative Rock	0	0	1

Output

- two-step manner
  1. Classify according to root genre
  2. Based on result in step 1, classify according to its corresponding leaf genres
- output function

$$f(x) = \mathrm{H}(x)\beta$$

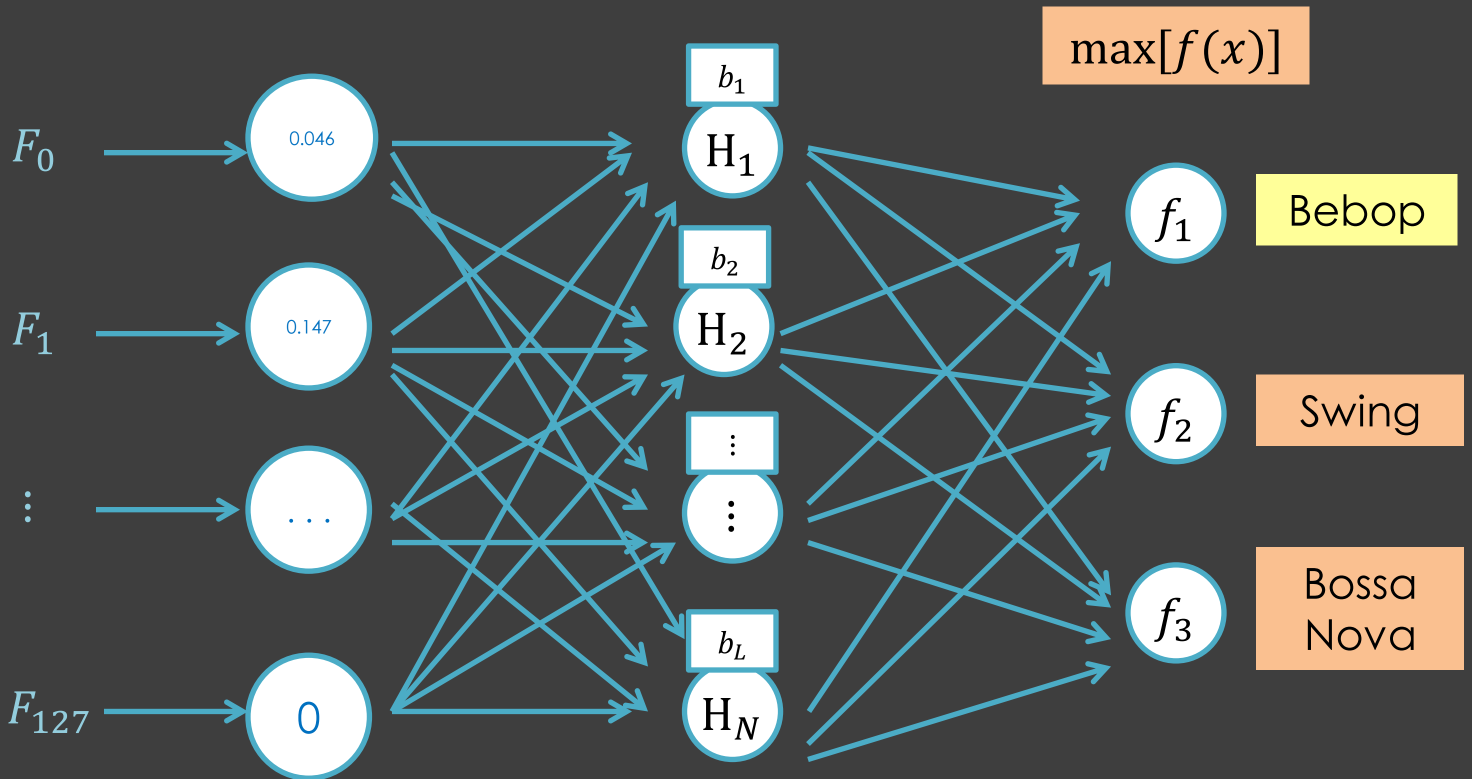
Output


max[f(x)]
F_0
F_1
⋮
F_{127}
0.046
0.147
. . .
0
b_1
b_2
⋮
b_L
H_1
H_2
⋮
H_N
f_1
f_2
f_3
Jazz
Rhythm 'n Blues
Rock

Genre Taxonomy

Jazz	Rhythm & Blues	Rock	- root genre
Bebop	Blues Rock	Hard Rock	{ leaf genres
Swing	Funk	Metal	
Bossa Nova	Rock 'n Roll	Alternative Rock	

Output



F_0
F_1
⋮
F_{127}
0.046
0.147
. . .
0
b_1
H_1
b_2
H_2
⋮
b_L
H_N
$\max[f(x)]$
f_1
f_2
f_3
Bebop
Swing
Bossa Nova

Output

- Max. Output Node-to-Genre Mapping

Output Node with Max. Value	Genre
1	Jazz
2	Rhythm & Blues
3	Rock

Output Node with Max. Value	Genre (Jazz)
1	Bebop
2	Swing
3	Bossa Nova

Output

- Max. Output Node-to-Genre Mapping (cont'd)

Output Node with Max. Value	Genre (R'nB)
1	Blues Rock
2	Funk
3	Rock 'n Roll

Output Node with Max. Value	Genre (Rock)
1	Hard Rock
2	Metal
3	Alternative Rock

Experiments and Results

Experiments and Results

- System Specifications
  - Hardware
    - Intel Core i5 CPU M 460 @ 2.53GHz
    - 2048MB RAM
  - Software:
    - Windows 8 Pro 32-bit (6.2, Build 9200)
    - Eclipse Juno IDE
    - Java Virtual Machine

Experiments and Results

- Experimental Setups
  - Variation in training and test sets used
  - Variation in number of hidden nodes
  - Introduction of a regularization parameter
    - Variation in reg. coeff values

Experiments and Results

- Experimental Setups
  - Variation in network structure *(i.e. no. of output nodes)*
  - Comparison with existing study

Experiments and Results

- Experimental Settings
  - 30 trials per setup
  - Dataset
    - 225 MIDI Files
      - Training: 80%
      - Testing: 20%
    - Random-stratified

Experiments and Results

- Experimental Settings
  - Dataset (cont'd.)

	Training	Testing
Jazz	60	15
R'nB	60	15
Rock	60	15
Root	**180**	**45**

Experiments and Results

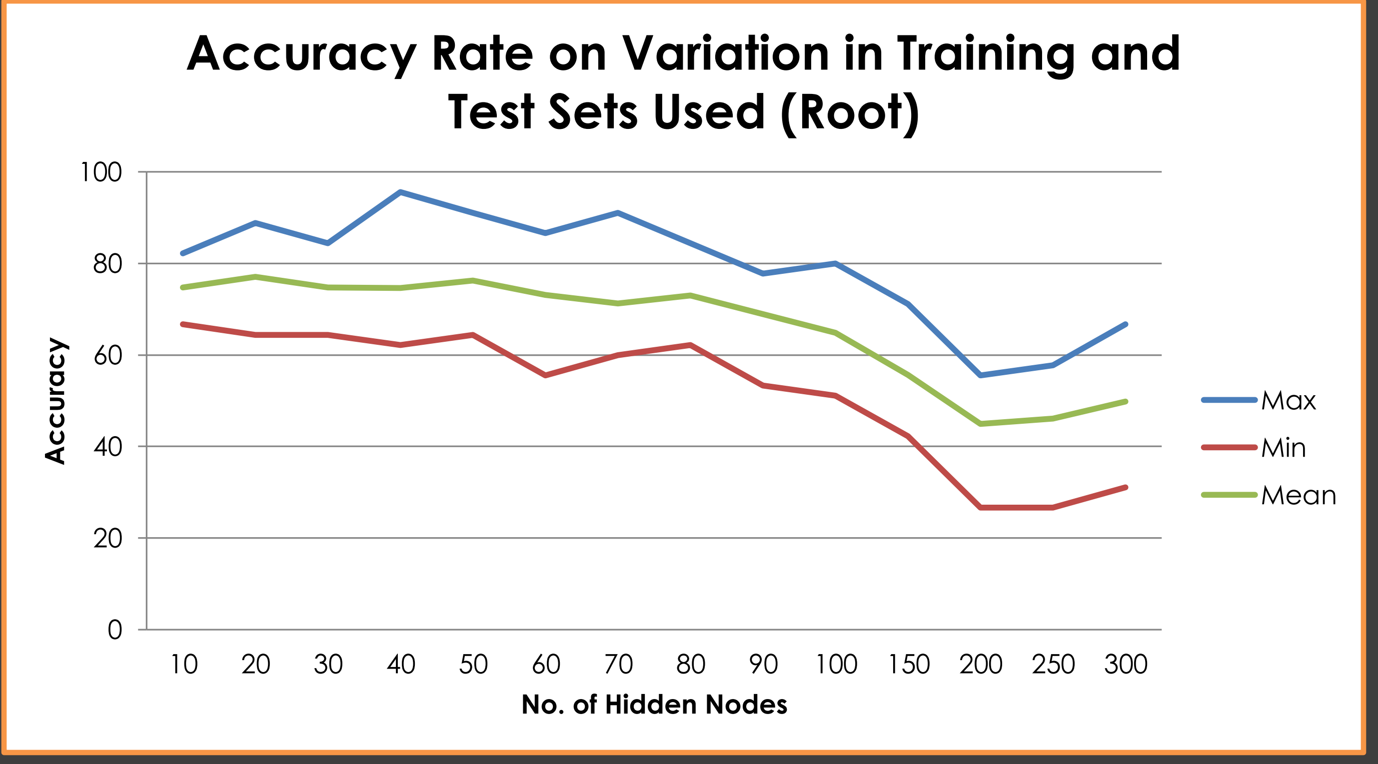
- Experimental Settings
  - Dataset (cont'd.)

		Training	Testing
Jazz	Bebop	20	5
	Swing	20	5
	Bossa Nova	20	5
R'nB	Blues Rock	20	5
	Funk	20	5
	Rock 'n Roll	20	5
Rock	Hard Rock	20	5
	Metal	20	5
	Alternative	20	5

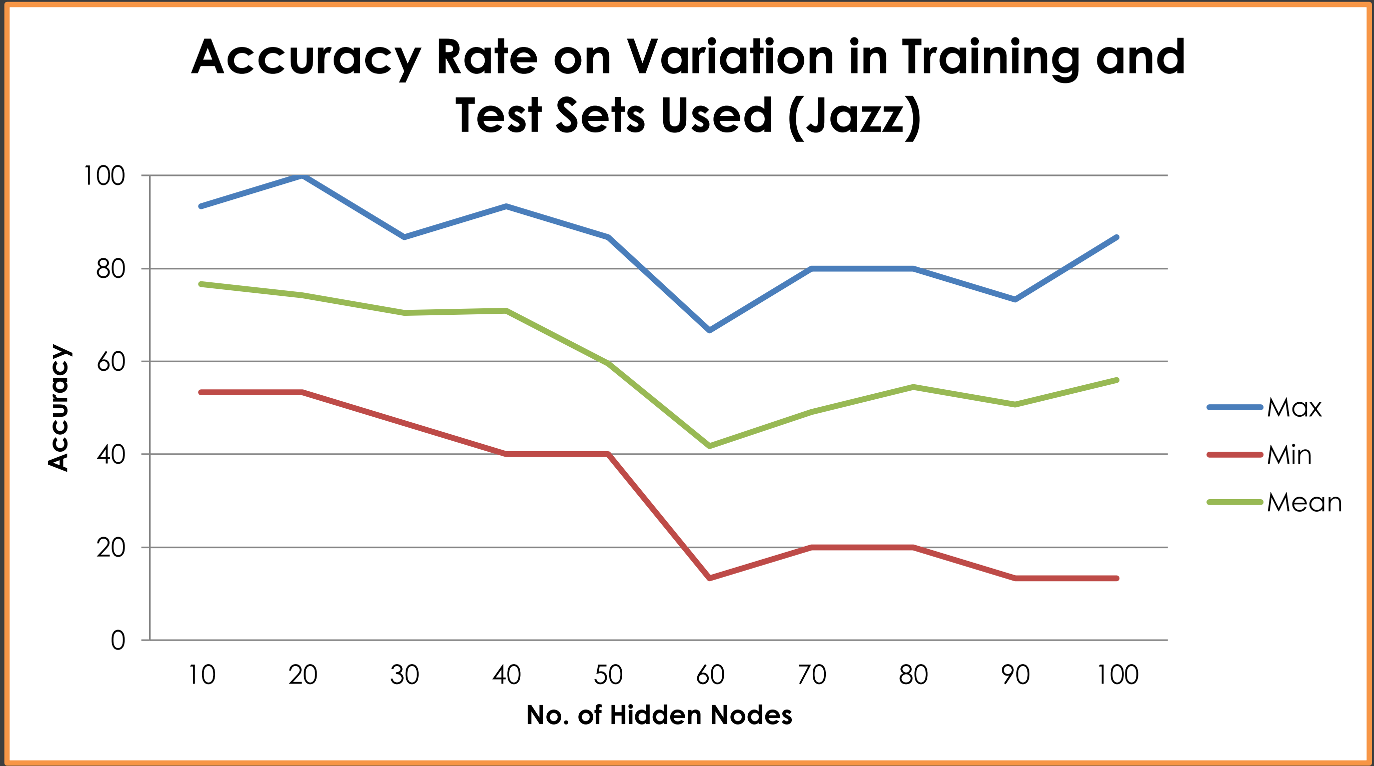
Experiments and Results

- Experimental Settings
  - initial parameters
    - Basic ELM
      - No. of hidden nodes 10
    - ELM w/ regularization parameter
      - Regularization coeff 0.2
  - dataset
    - random partitioning of dataset
    - fixed dataset

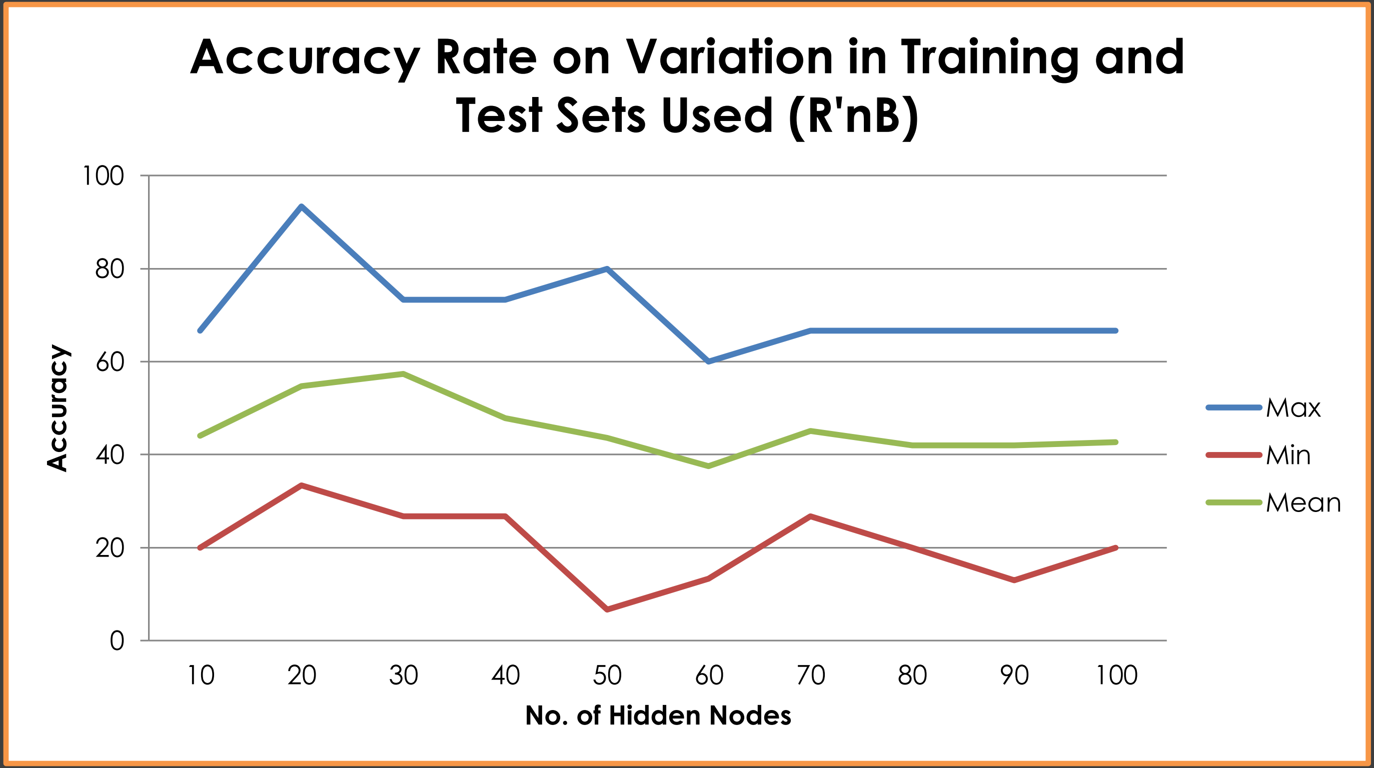
Experiments and Results



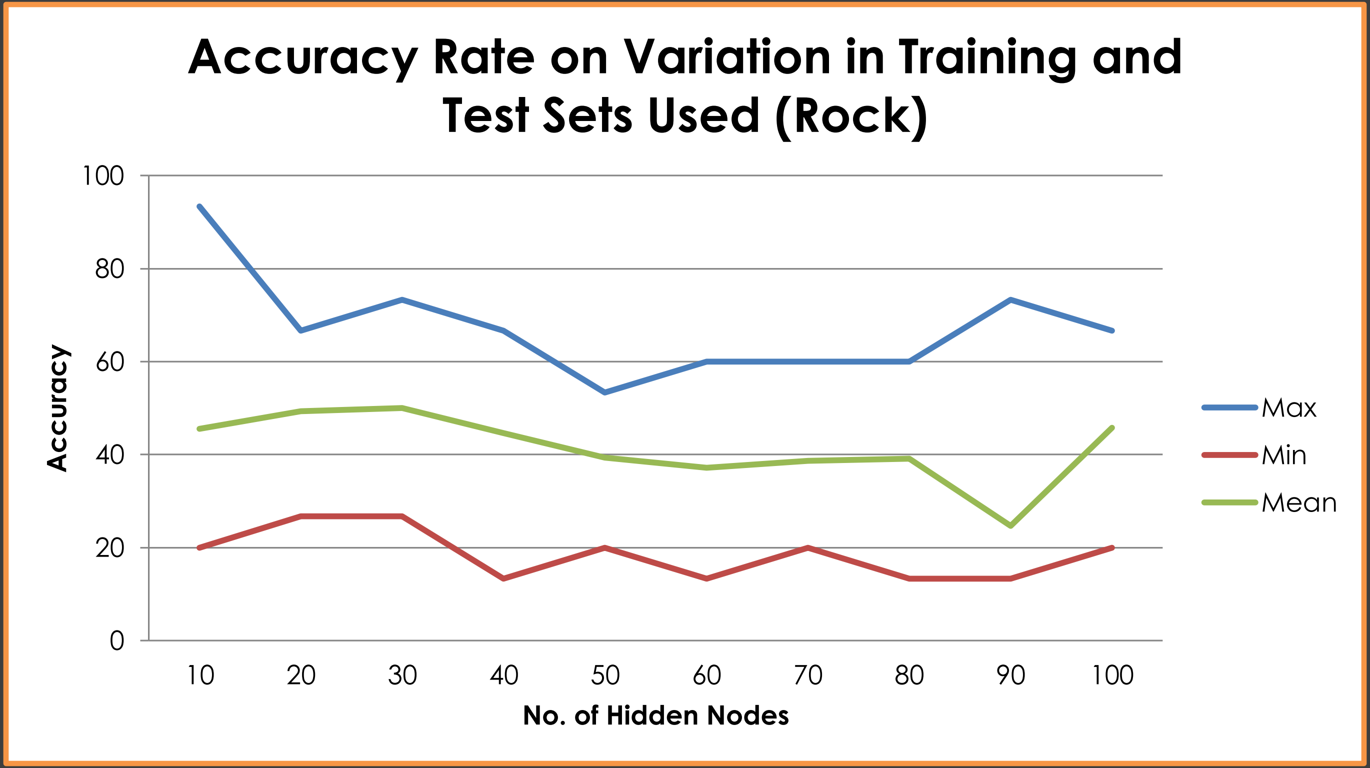
Experiments and Results



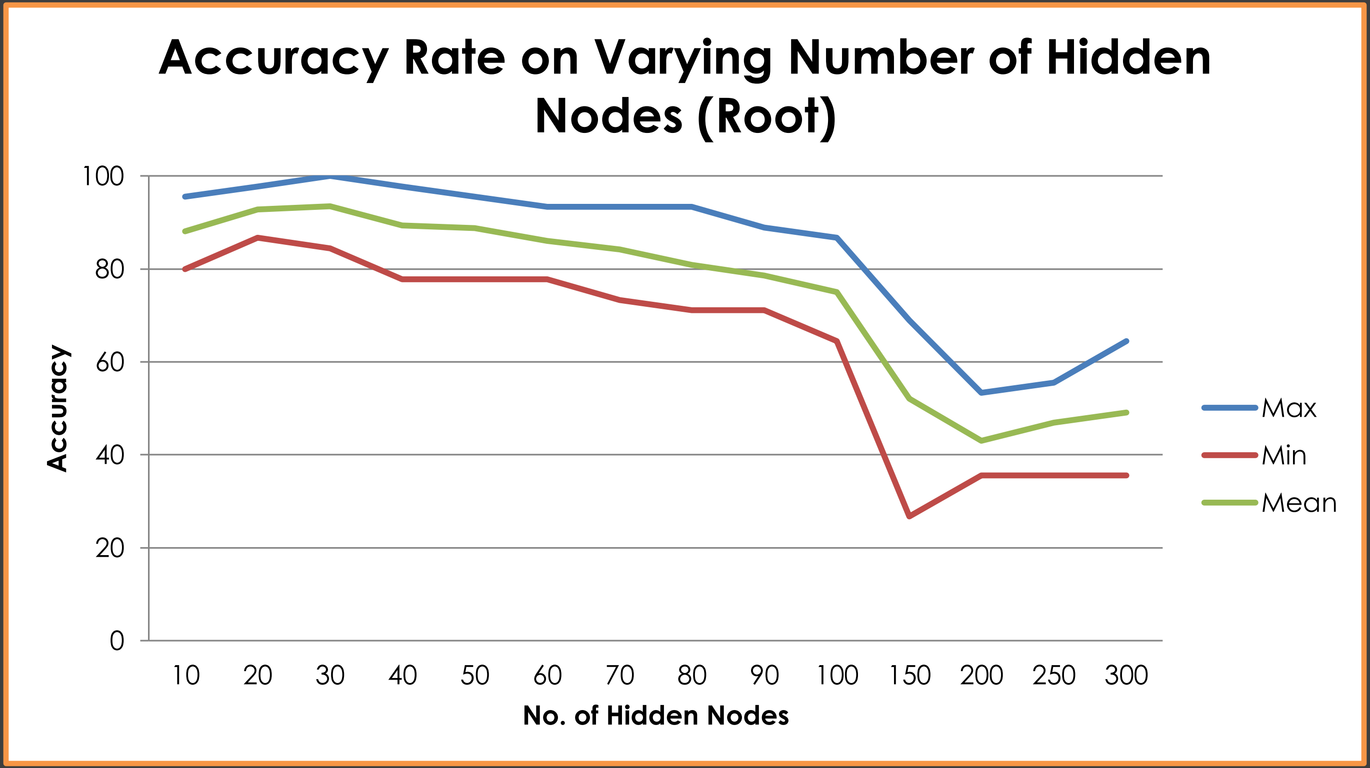
Experiments and Results



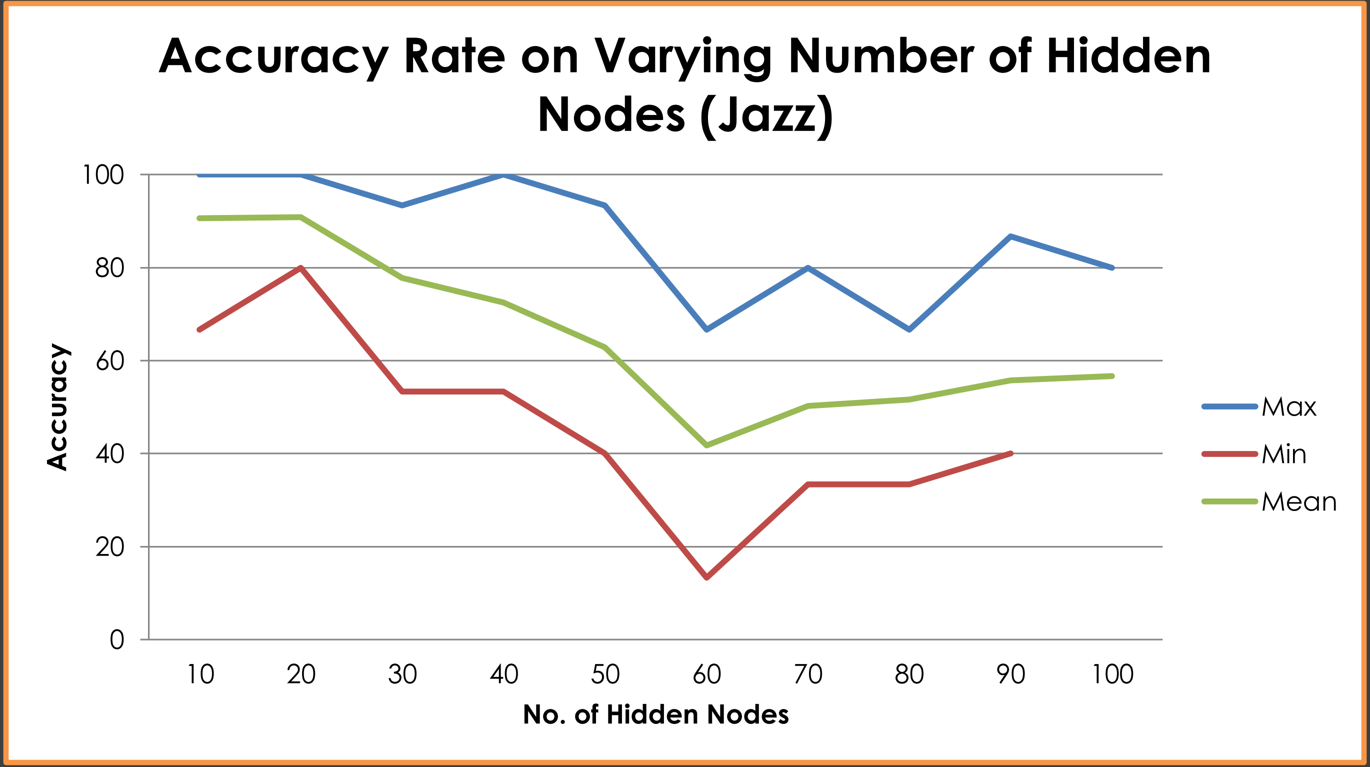
Experiments and Results



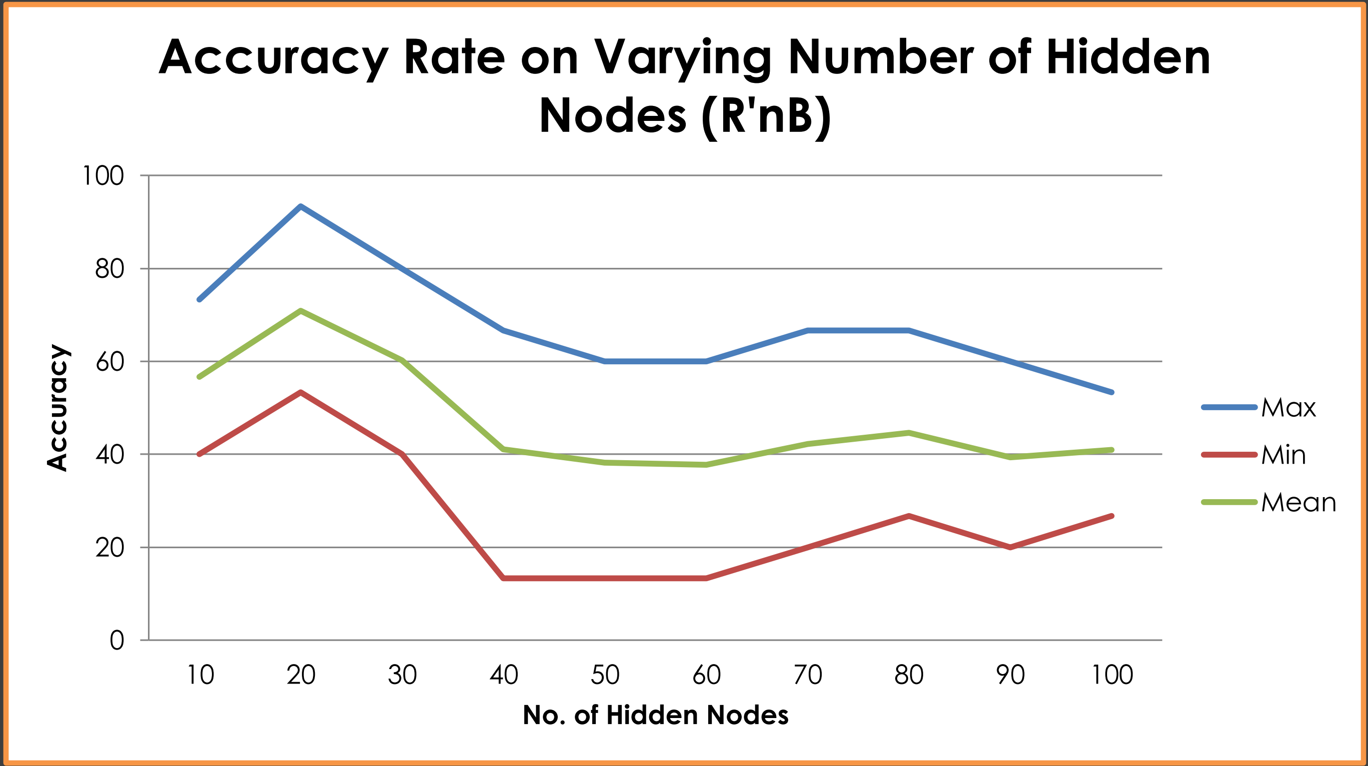
Experiments and Results



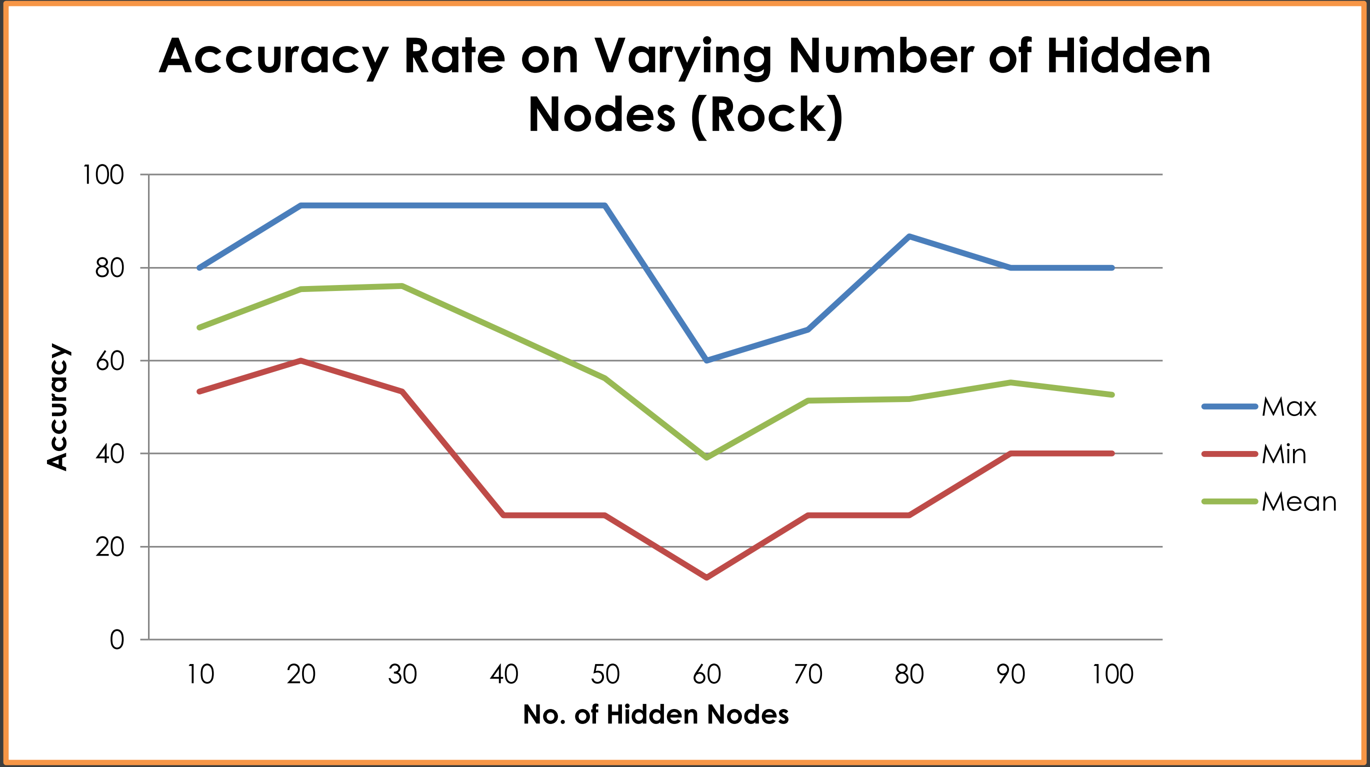
Experiments and Results



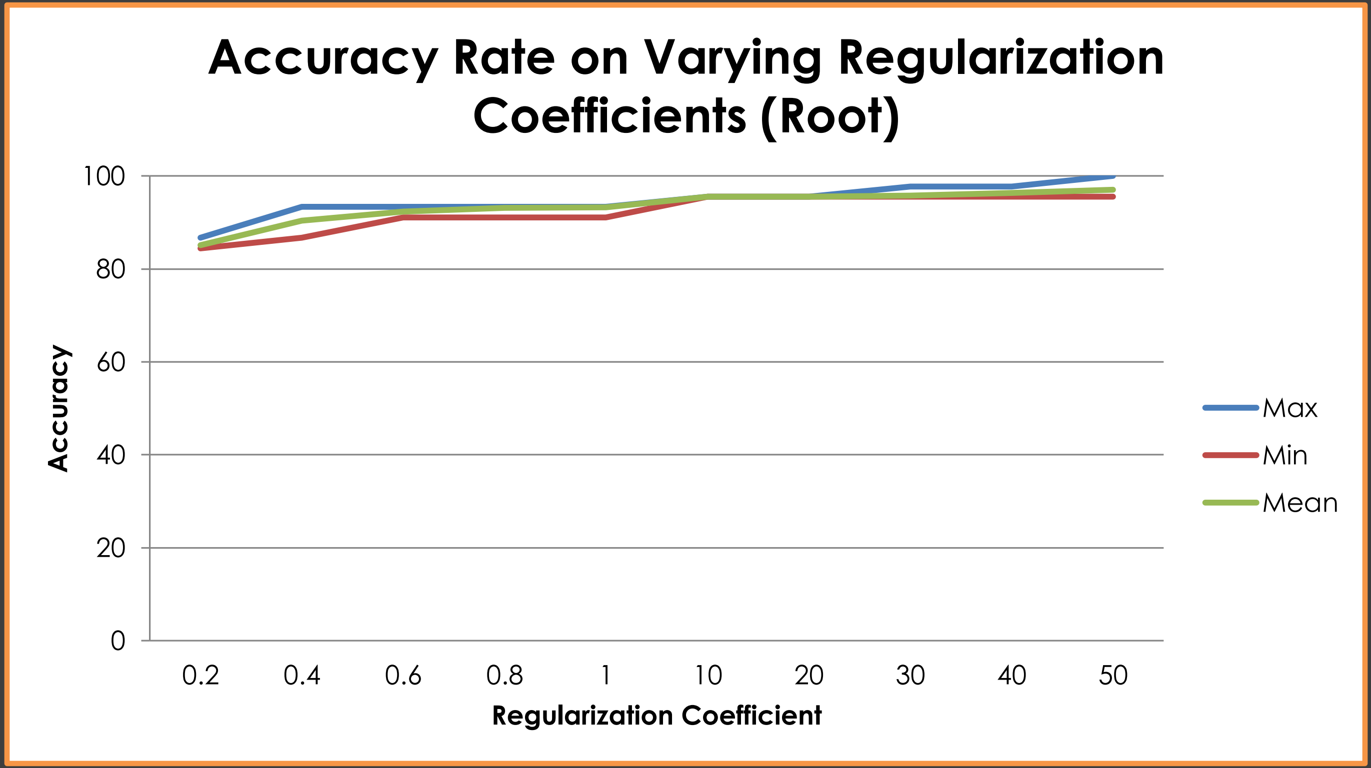
Experiments and Results



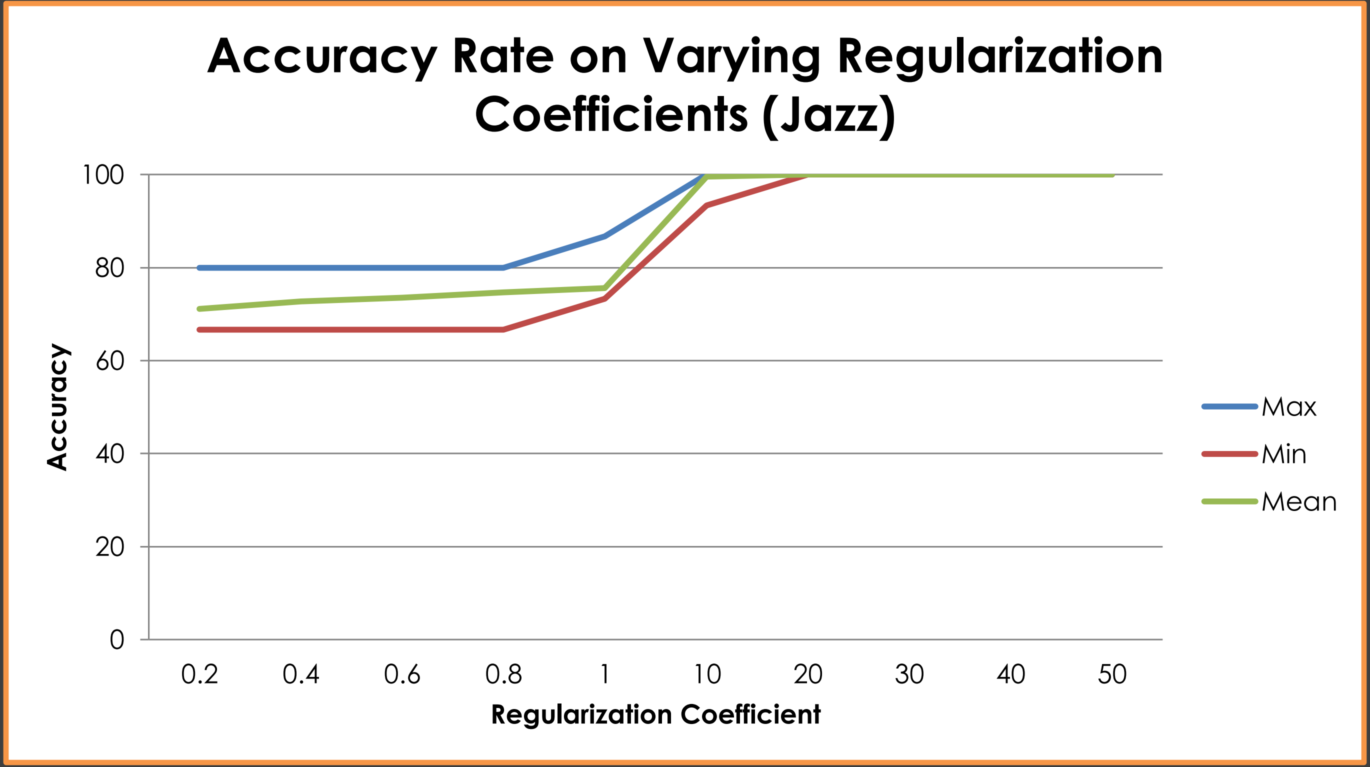
Experiments and Results



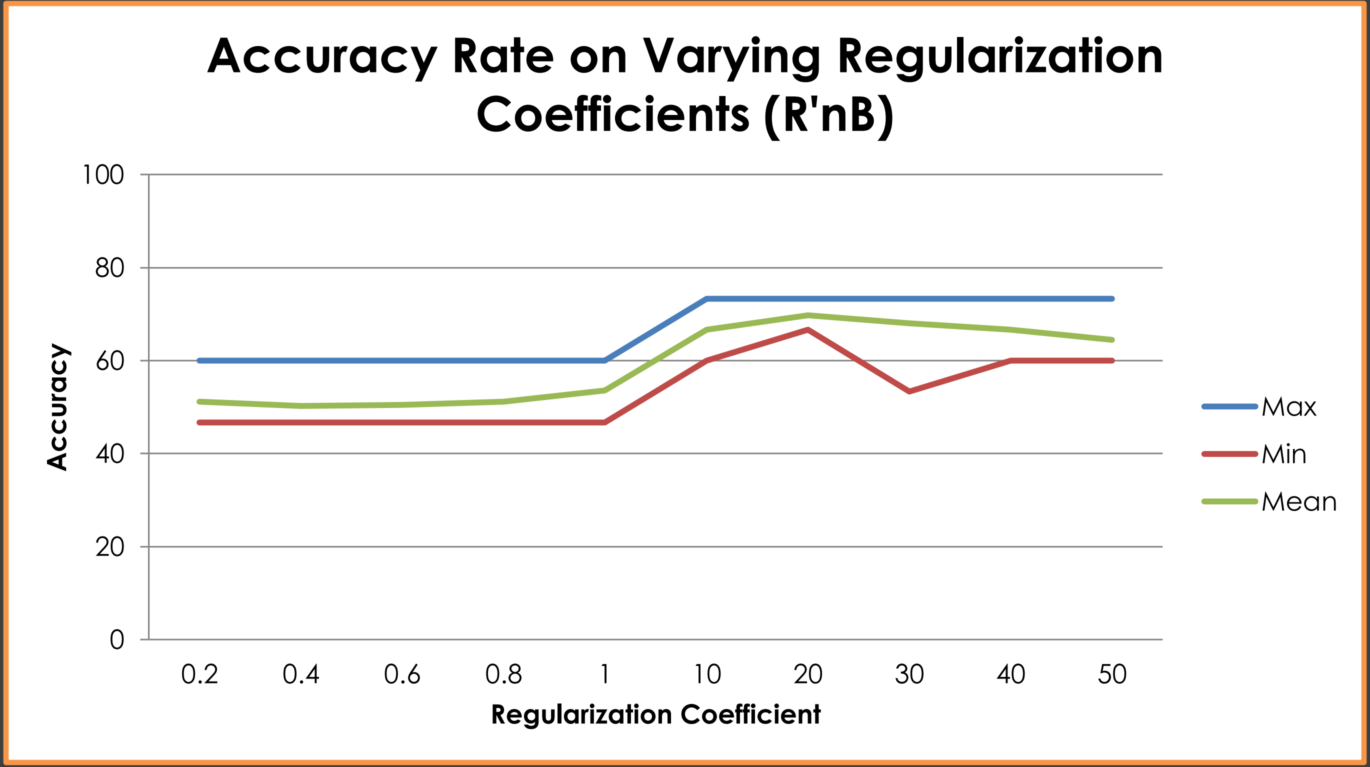
Experiments and Results



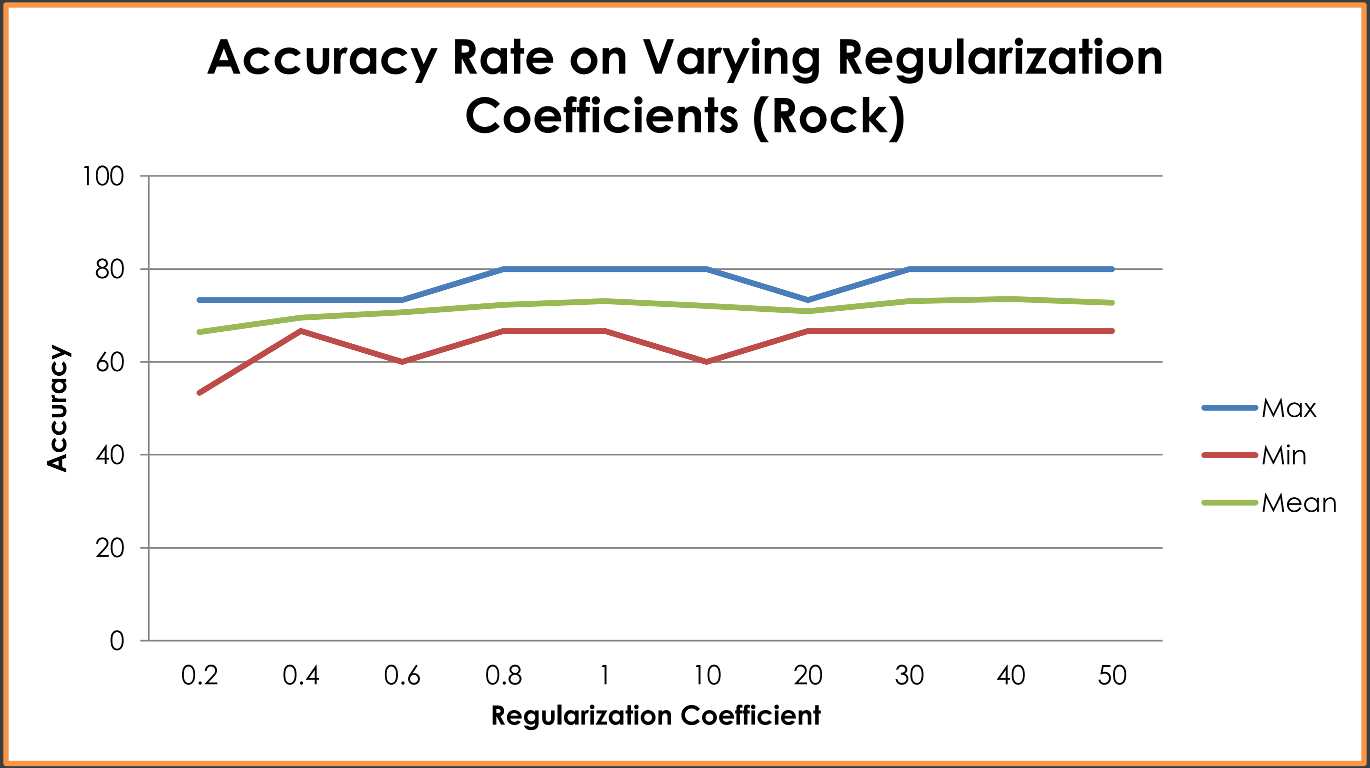
Experiments and Results



Experiments and Results



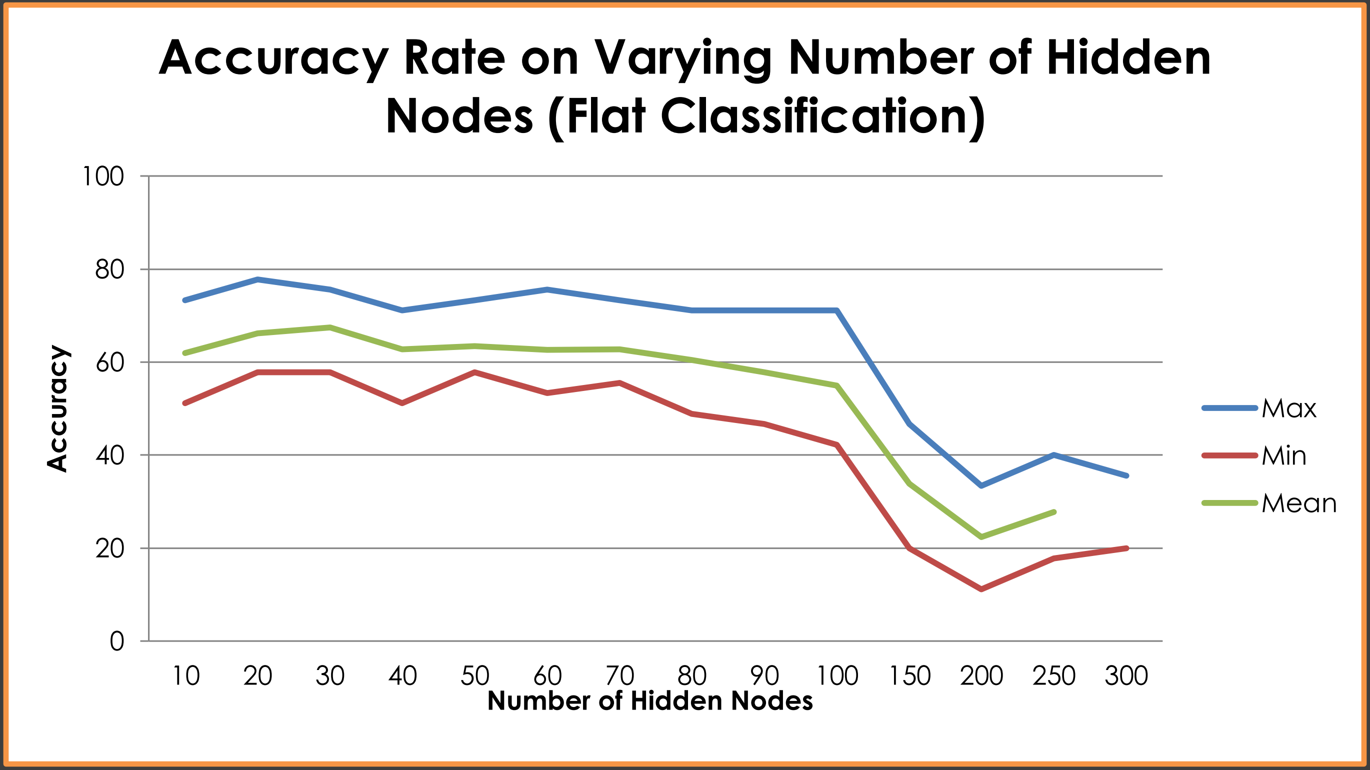
Experiments and Results



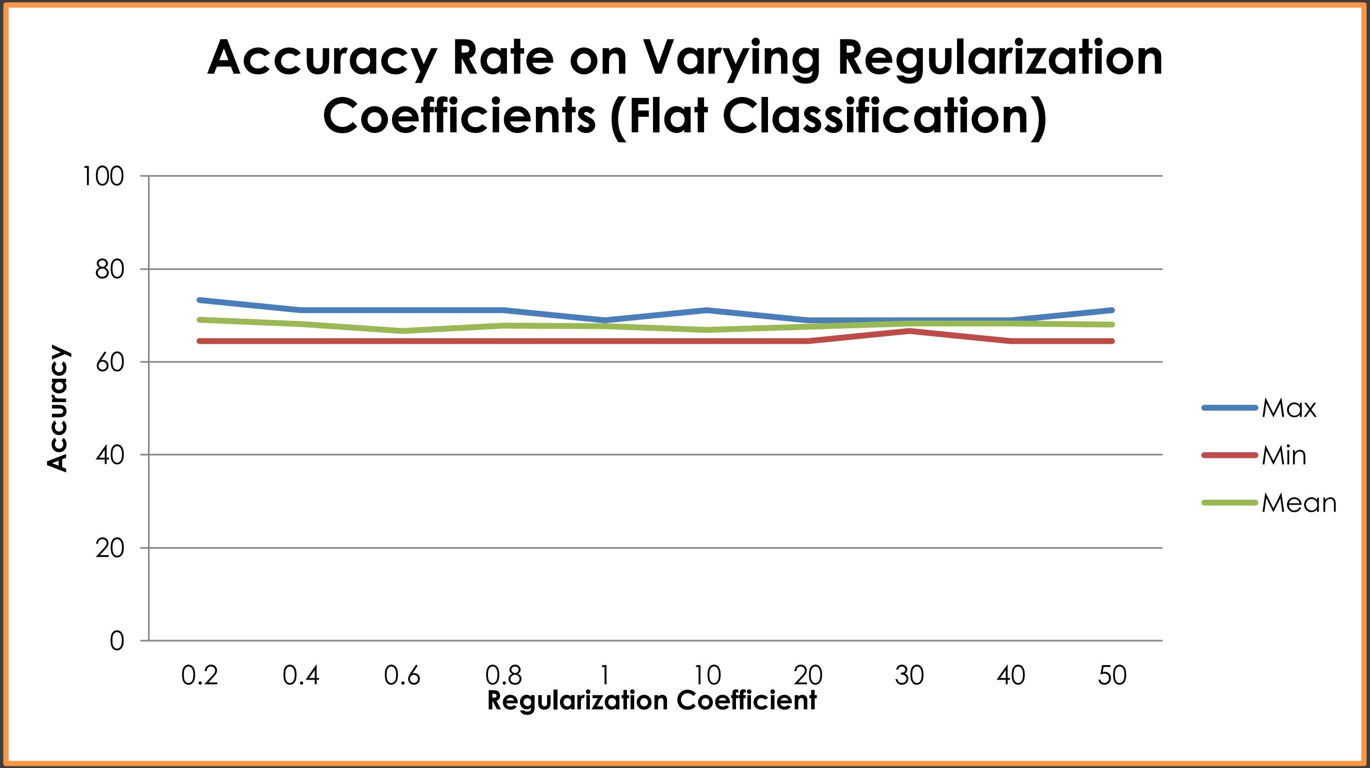
Experiments and Results

	Root (300 hidden nodes)	Leaf (100 hidden nodes)
Basic ELM	49.11%	50.08%
ELM w/ reg. coeff (C. @ 0.2)	**85.11%**	**62.96%**

Experiments and Results



Experiments and Results



Experiments and Results

Algorithm	Distance Metric / Type	Root Accuracy (%)	Leaf Accuracy (%)
KNN	ED	97.78	84.44
	EMD	93.33	80.00
	KL_2D	93.33	80.00
	NCD	75.56	66.67
	PWED	**100**	86.67
ELM	---	**100**	**95.56**

Conclusion and Future Works

Conclusion

- ELM is robust
  - good for music genre classification
- ELM is able to make good generalizations even with just a few hidden nodes, but declines when the number of hidden nodes is increased

Conclusion

- Adding a regularization parameter makes the ELM more stable.
- Hierarchical approach is better than flat classification

Future Works

- Online-Sequential ELM (OS-ELM)
  - No need for retraining when data/training set is updated
- Larger dataset
- Bassline Transcription