Implementing Machine Learning Algorithms from scratch

1. LOGISTIC REGRESSION
2. NEURAL NETWORK

Jai Jain (2017A7PS1585H)
Anirudh Gupta (2017A7PS0149H)
Vidish Bharadwaj (2017A4PS1391H)
BITS F464

1. Logistic Regression

Feature Selection:

The Pearson correlation coefficient can be used to summarize the strength of the linear relationship between a feature and the output.

The Spearman's correlation coefficient can be used to summarize the strength of the nonlinear (non-Gaussian) relationship between a feature and the output.

Pearson's Correlation Coefficient

	Variance	Skewness	Curtosis	Entropy
Class	-0.72484	-0.44469	0.15588	-0.02342

Spearman's Correlation Coefficient

	Variance	Skewness	Curtosis	Entropy
Class	-0.73560	-0.42902	0.06160	-0.03275

The Spearman's Correlation Coefficient and the Pearson's Correlation Coefficient of the Entropy feature with the Output class is very close to Zero. Hence, we can drop this feature.

This is evident from the final weight of the 'Entropy' feature after gradient descent which is significantly less (close to zero) with and without regularization.

Feature Scaling:

The features are scaled using the equation - $z = \frac{x-\mu}{\sigma}$

The result of **standardization** (or **Z-score normalization**) is that the features will be rescaled so that they'll have the properties of a standard normal distribution with $\mu=0$ and $\sigma=1$.

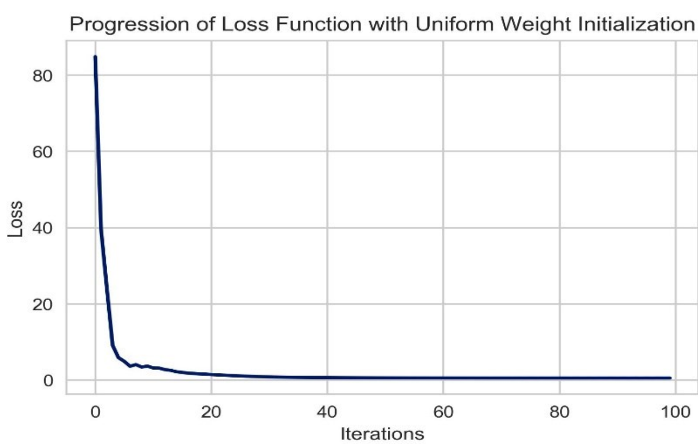
While using gradient descent optimization, the feature should be standardized otherwise some weights will update much faster than others.

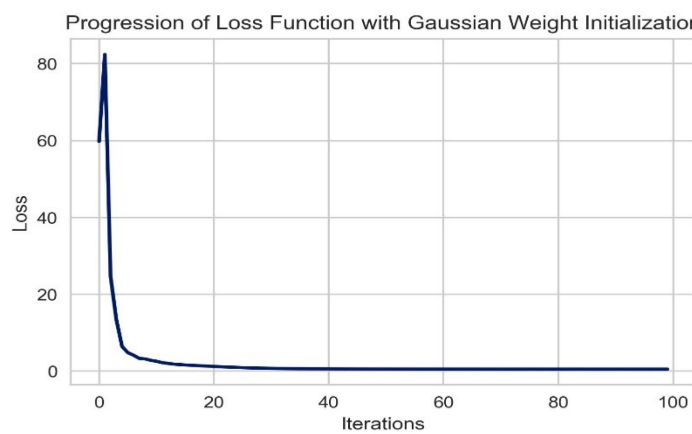
Without Regularization:

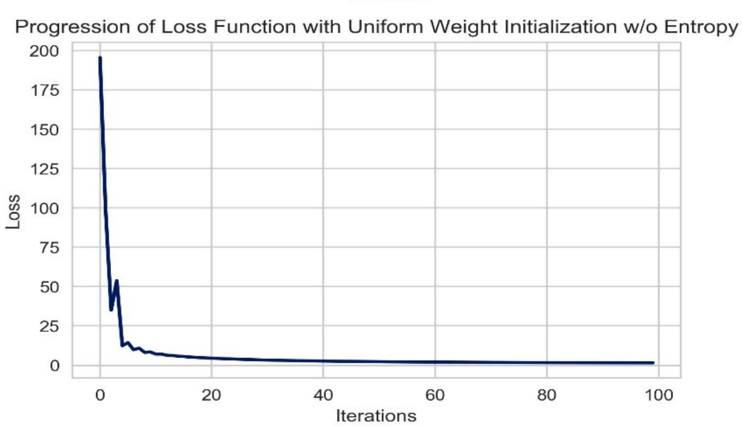
Optimum Value for Parameters:

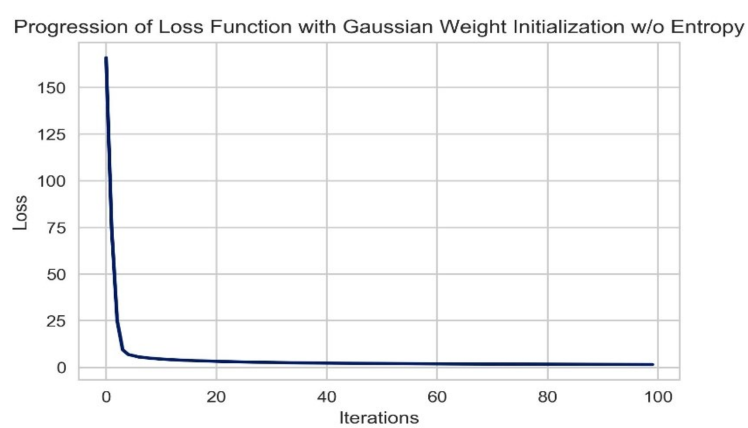
1. Weight Initialization – Gaussian/Uniform [0,1)
2. Iterations - 50
3. Learning Rate – 0.075
4. Final Weights: [-11.68573, -24.36882, -26.76758, -24.83826, -1.42284]

Sr No	Weight Initialization	Epochs	Learning Rate	Without Regularization	
				Accuracy	F-score
1	Uniform [0,1)	50	0.075	99.63	0.99535
2	Gaussian	50	0.075	99.63	0.99535
3	Uniform [0,1) Without Entropy	100	0.0005	99.63	0.99535
4	Gaussian Without Entropy	100	0.0005	99.63	0.99535









With Regularization:

1. L1 Regularization:

Optimum Value for Parameters:

1. Weight Initialization – Gaussian/Uniform [0,1)
2. Iterations - 50
3. Learning Rate – 0.075
4. Final Weights: [-6.27555, -13.45655, -14.98809, -13.72931, -0.42028]

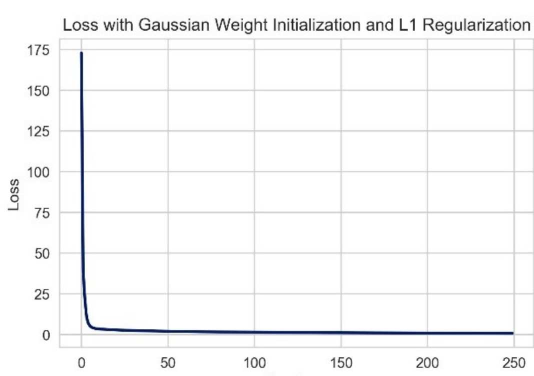
Sr No	Weight Initialization	Epochs	Learning Rate	Regularization Constant	L1 Regularization	
					Accuracy	F-score
1	Uniform [0,1)	50	0.05	0.00001	99.63	0.99535
2	Gaussian	50	0.05	0.00001	99.63	0.99535

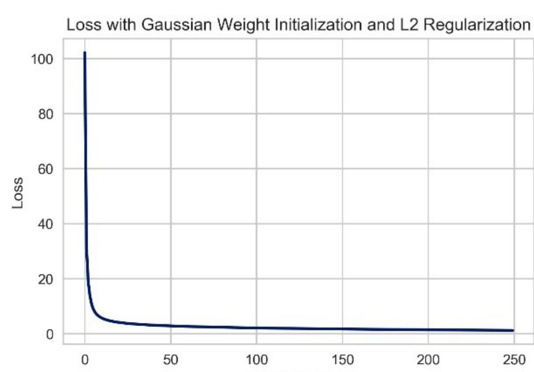
2. L2 Regularization:

Optimum Value for Parameters:

1. Weight Initialization – Gaussian/Uniform [0,1)
2. Iterations - 50
3. Learning Rate – 0.075
4. Final Weights: [-6.24352, -13.57593, -14.95926, -13.76691, -0.29476]

Sr No	Weight Initialization	Epochs	Learning Rate	Regularization Constant	L2 Regularization	
					Accuracy	F-score
1	Uniform [0,1)	50	0.05	0.00001	99.63	0.99535
2	Gaussian	50	0.05	0.00001	99.63	0.99535





2 Neural Network

Sr No	No of Neurons in Hidden Layers	Weight Initialization	Epochs	Learning Rate	The Output Accuracy and F- score using Activation Functions for the Hidden Layer		
					Tanh	Relu	Sigmoid
1	[5]	Gaussian	1000	0.00075	A = 88.014 F = 0.88054	A = 88.356 F = 0.88888	A = 89.726 F = 0.89864
2	[5]	Gaussian	1000	0.0005	A = 84.246 F = 0.86000	A = 86.644 F = 0.87043	A = 88.699 F = 0.88888
3	[5]	Gaussian	1000	0.00025	A = 88.699 F = 0.88813	A = 88.356 F = 0.88741	A = 88.699 F = 0.88737
4	[5]	Gaussian	1500	0.0001	A = 88.699 F = 0.89041	A = 86.644 F = 0.87128	A = 89.384 F = 0.89491
5	[5]	Gaussian	800	0.00025	A = 87.329 F = 0.87625	A = 88.014 F = 0.87804	A = 89.041 F = 0.89189
6	[5]	Gaussian	1000	0.001	A = 87.671 F = 0.87586	A = 88.356 F = 0.88356	A = 89.041 F = 0.89189
7	[5]	Uniform	800	0.00025	A = 89.041 F = 0.89261	A = 91.096 F = 0.91095	A = 88.356 F = 0.88590
8	[5]	Uniform	1000	0.0005	A = 88.699 F = 0.89036	A = 89.384 F = 0.89491	A = 88.699 F = 0.88963
9	[5]	Uniform	500	0.0005	A = 88.699 F = 0.88888	A = 88.699 F = 0.88659	A = 88.356 F = 0.88513
10	[5]	Uniform	500	0.00075	A = 89.384 F = 0.89562	A = 90.068 F = 0.90169	A = 88.014 F = 0.88215
11	[5]	Uniform	800	0.00075	A = 89.041 F = 0.89333	A = 89.041 F = 0.89041	A = 89.384 F = 0.89632
12	[5]	Uniform	1000	0.00075	A = 87.671 F = 0.88157	A = 90.068 F = 0.90235	A = 89.041 F = 0.89261

Sr No	No of Neurons in Hidden Layers	Weight Initialization	Epochs	Learning Rate	The Output Accuracy and F- score using Activation Functions for the Hidden Layer		
					Tanh	Relu	Sigmoid
13	[10]	Gaussian	1500	0.0001	A = 84.247 F = 0.88741	A = 88.014 F = 0.88135	A = 89.726 F = 0.89655
14	[10]	Gaussian	1500	0.00025	A = 85.959 F = 0.88000	A = 88.356 F = 0.88435	A = 88.699 F = 0.88963
15	[10]	Gaussian	1500	0.0005	A = 88.014 F = 0.88054	A = 88.699 F = 0.88888	A = 89.041 F = 0.89261
16	[10]	Uniform	1000	0.00025	A = 88.699 F = 0.88813	A = 90.068 F = 0.90102	A = 88.699 F = 0.88963
17	[10]	Uniform	1250	0.00025	A = 88.014 F = 0.88215	A = 90.411 F = 0.90540	A = 88.699 F = 0.88963
18	[10]	Uniform	800	0.00025	A = 88.699 F = 0.88813	A = 91.096 F = 0.91095	A = 88.699 F = 0.88963

Sr No	No of Neurons in Hidden Layers	Weight Initialization	Epochs	Learning Rate	The Output Accuracy and F- score using Activation Functions for the Hidden Layer		
					Layer 1 – Relu Layer 2 - Relu	Layer 1 – Relu Layer 2 - Tanh	Layer 1 - Tanh Layer 2 - Relu
19	[10], [5]	Gaussian	1500	0.0001	A = 90.06 8 F = 0.90169	A = 88.356 F = 0.88356	A = 88.014 F = 0.88524
20	[10], [5]	Uniform	1500	0.0001	A = 89.041 F = 0.89115	A = 89.384 F = 0.89562	A = 89.041 F = 0.88811

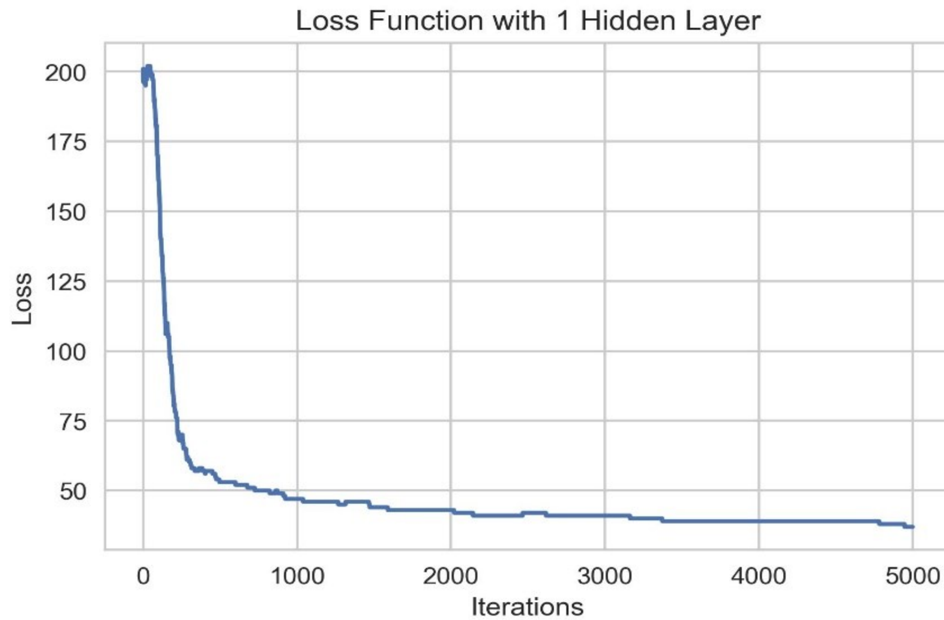
Thus, we get the best accuracy and F-score using 1 hidden layer with 5 hidden neurons.

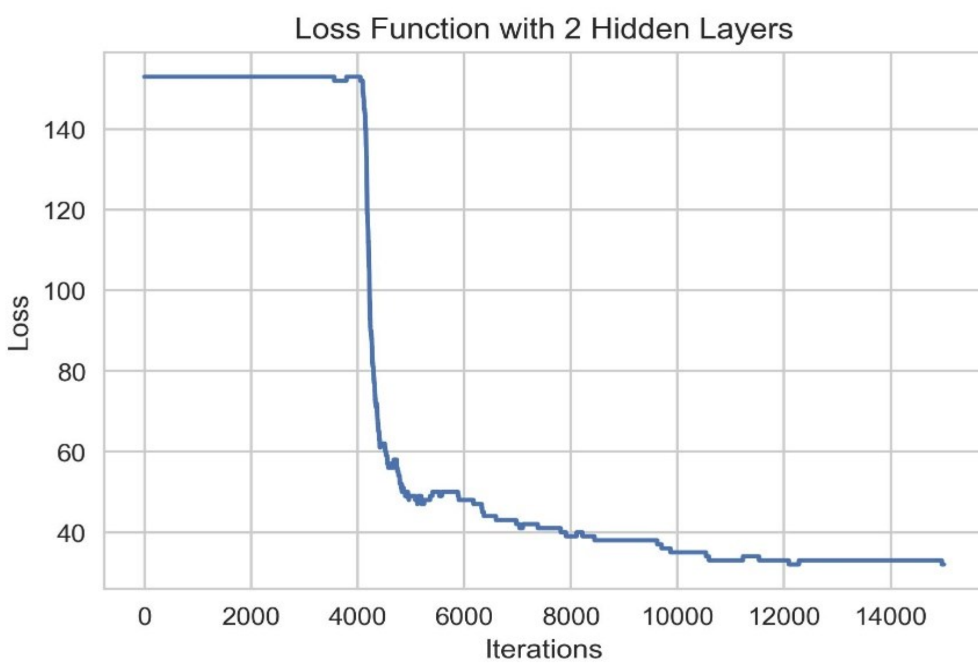
Activation Function of the hidden Layer – Relu

Weight Initialization – Uniform [0,1)

Learning Rate – 0.00025

Epochs – 800


Loss Function with 1 Hidden Layer
200
175
150
125
100
75
50
Loss
0
1000
2000
3000
4000
5000
Iterations


Loss Function with 2 Hidden Layers
140
120
100
80
60
40
Loss
0
2000
4000
6000
8000
10000
12000
14000
Iterations

3 References

https://machinelearningmastery.com/how-to-use-correlation-to-understand-the-relationship-between-variables/

https://sebastianraschka.com/Articles/2014_about_feature_scaling.html#about-standardization

https://kharshit.github.io/blog/2018/03/23/scaling-vs-normalization