[http://machinelearning.wustl.edu/mlpapers/paper\_files/AISTATS2012\_RaikoVL12.pdf](http://machinelearning.wustl.edu/mlpapers/paper_files/AISTATS2012_RaikoVL12.pdf%20) : Original Document

SUMMARIZATION DONE BY KUNAL DARGAN:

**‘DEEP LEARNING MADE EASIER BY LINEAR TRANSFORMATIONS IN**

**PERCEPTRONS’**

**By: Tapani Raiko Harri Valpola Yann LeCun**

**Aalto University Aalto University New York University**

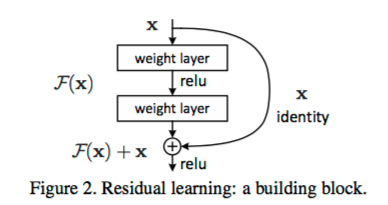
{1} Summarizer’s comment in Bibliography

Introduction and Aim of the Paper:

This paper talks about the *Transformations* for implementing Deep Neural Networks using short-connections architecture of two to five hidden layers deep to model and which is implemented to solve the problem of Image recognition in MNIST Handwritten digits and CFIAR-10 classification (using unsupervised autoencoder).Writer have first analyzed theoretically ,that transformations by noting , ‘they make Fisher information matrix close to a diagonal matrix and hence standard gradient closer to the natural gradient(Quasi-Newton’s Method or BFGS)[5]’ ,which is much faster algorithm for optimization of the model but due to harder implementation in higher dimensions it cannot be used.

Further, Proposed paper experimentally proves given the input as normalized (scaled) and centered (close to zero) values to address the problem of vanishing/exploding gradients, output is transformed according to the transformations which results in basic gradient descent can be applied as efficiently as other state-of-the-art algorithms in terms speed and even provides better generalization and be used on both supervised and unsupervised way of learning.

Short-Connections



<https://qph.ec.quoracdn.net/main-qimg-b1fcbef975924b2ec4ad3a851e9f3934> In the context of artificial neural networks, the relu is an activation function fx

Deep neural networks (up to 50 layers or 150 layers) suffers from the problem of accuracy saturation [2], i.e. after reaching a certain max accuracy (plateau or U curve) model doesn’t gets better or accuracy starts decreasing, and this not due to overfitting, here authors have proved experimentally that if a shallow neural network achieves a certain accuracy then any deeper NN can be at least as well as the shallower NN. Extra layers instead of providing better estimates starts contributing to bigger error and learning becomes inefficient, it follows that

**{Zero Mapping} F(x) = H(x)-x 🡺** F(x) is the error value, H(x) is the predicted value of x

Here the role of short cut connections comes into play, short cut connections are those which skips one or more layers (fig2.) and contribute (x) directly to f(x) hence changing zero mapping (trying to get to identity mapping, [2] Microsoft paper proved an intuitive hypothesis that it is easier to optimize new identity mapping than to previous zero mapping (estimation of a small value error).

**Zero Mapping} F(x) +x= H(x) 🡺** short connection, easier to optimize and doesn’t added any extra parameter.

For further readings into short cut connection refer to [2]: Microsoft Research

Fisher Information Matrix

Before going further, it is essential to get an idea of what fisher Information matrix and what is log-likelihood which forms the basis of entire theory in statistics called point estimation, (referred from statistics course Nptel)

Likelihood: In statistics *MLE (Maximum likelihood Estimation)* is a method of estimation of underlying parameter given the observations of a statistical distribution and likelihood is a function l

It is not simply the summation of natural log of probability density function f(x/Ɵ), or observed values which are set into pdf, but the trick is to maximize L wrt. Ɵ and estimate it using MAP (maximum a posteriori).

**Fisher Information**: ***“Fisher information****(sometimes simply called****information*** [*[1]*](https://en.wikipedia.org/wiki/Fisher_information#cite_note-1)*) is a way of measuring the amount of information that an observable random variable X carries about an unknown parameter θ of a distribution that models X.”{Wiki}* Formally it is the variance of the score or the expected value of the observation. *“The Fisher-information matrix is used to calculate the*[*covariance matrices*](https://en.wikipedia.org/wiki/Covariance_matrices)*associated with*[*maximum-likelihood*](https://en.wikipedia.org/wiki/Maximum_likelihood)[*estimates*](https://en.wikipedia.org/wiki/Estimates)*” {Wiki}*

All of this gets importance when we look to find out Fisher’s Information Matrix, which is defined as : Let us consider a family of density functions F that maps a parameter θ ∈ RP to a probability density function P(z), P : R N → [0, ∞), where z∈ R N . Any choice of θ ∈ R P defines a particular density function Pθ(z) = Fθ(z) and by considering all possible θ values, we explore the set F,For further derivation refer {Wiki: <https://en.wikipedia.org/wiki/Fisher_information>}

Transformations

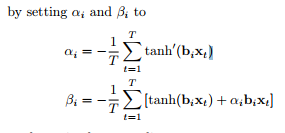
*“In this paper, we transform the nonlinearities in the hidden neurons.* *We explain the usefulness of these transformations by studying the Fisher information matrix”*

Transformations

For implementation of the proposed transformations writer has made use of **yt = Af (Bxt) + Cxt + Ɛt ,** Which represents a direct connection with bias over which activation has been applied ***”*** *A, B, and C are the weight matrices, and* **Ɛt** *is the noise which is assumed to be zero mean and Gaussian”,* weights contains bias ; αi and βi parameters for each activation function f:

**fi (bixt) = tanh (bixt) + αibixt + βi (1)**

**bi for B weight matrix i row**

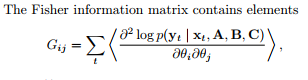


Here , writer have converted f’(xi)=0 and f’’(xi)=0 so that overall function , which gives freedom to introduce shortcut mapping at this point , Deep –residual NN is easier to optimize .

Basic GD to Natural GD: TRANSFORMATION

As second order optimization techniques perform better but cannot be used for high dimensional and large models due to heavy computations, writer has put these transformations in front

{2nd derivative of Log-likelihood function is fisher information matrix : Layman’s Explanation}



Transformations are result in matrix of higherdemisions ie, matrix of FIM (containing eigen values )is mapped like a covariance matrix ie, “When the units are not completely uncorrelated put 0 in Information matrix ,eventually this technique should move FIM(Fisher’s Information matrix to a diagonal matrix)

Techniques used:

Positive Side Effect: By the result of Normalization of problem of saturation and plateau solver

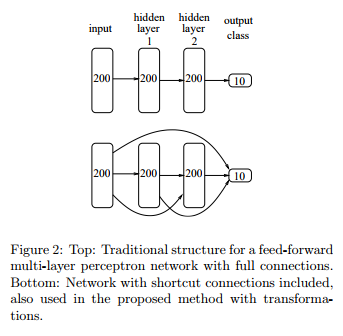
Issues:

Learning rate is determined as estimation error of MLE estimation of Ɵ,

**Ɵ <- Ɵi+ygi**

Online learning: 2 step process of training (weights randomization) and validation separately(learning rate adjustment)

Multiple Hidden layers: shortcut mappings have to be included in first transformation, it increases weights quadratically as in figure 2



**MNIST DATASET**

As MNIST data consists of huge set of 60000 images as samples and 10000 test samples, Dimensionality reduction has been performed by subtracting mean activation values from the data, it reduces 28\*28=784 input parameters to 200 ,thereafter random rotations has been applied in order to make model generic and insensitive from in-class changes,

Model: 200-200-200-10:3 models Original, shortcuts (including shortcut mapping) and transformations and 200-10 linear network for baseline comparison, parameter was λ = 0.0001 and the regularization noise was used with standard deviation 0.4 for each input

Result: 1.1% best using transformations

Implementation:*” The experiments were run with an 8-core desktop computer and Matlab implementation. The learning time was measured with Matlab function cputime, which counts time spent by each core”*

Transformation have performed satisfactorily and have decreased the norm of FIM and hence small learning rate and small gradient.

**CIFAR-10 CLASSIFICATION**

Data: 32\*32 pictures, 50000 samples and 10000 test samples

Model: PCA converted 32\*32\*3=3072 to 500 ; 500-500-500-10 structure of NN ; n. Learning time was restricted to 10000 seconds, the base learning rate was set by hand to γ = 0.3, weight decay parameter was λ = 0.001 and the regularization noise was used with standard deviation 0.4 for each input

Result: 43.70% error which is better from 48.47% and 52.92% previous results

**MNIST AUTOENCODER (UNSUPERVISED LEARNING)**

Model: The network topology is 784–500–250–30–250–500–784 with tanh, output scaling -1 to 1 to match tanh, learning rate was γ = 0.05, weight decay parameter was λ = 0.001

Result: error is 2.44% better compared to other well-known algorithms

**Conclusion:**

Benefits of Transformations and future scope of discussions:

* A basic stochastic gradient optimization became faster than state-of-the-art learning algorithms
* Pushes Standard GD to Natural GD by making non diagonal terms in FIM close to 0
* Method generalizes well and can applied or further discussed for future applications
* Another future direction is to introduce a third transformation. While currently we aim at making the Fisher information matrix diagonal, we could also make it closer to the unit matrix.

# Bibliography

“Summarizer’s final comment: This paper is really one of high class, readers are requested to try to understand the concepts in brief and think more about the concepts, how they affect overall fulfilment of the objective and brief introduction about every concept is included in this review, for formal proofs and deeper understanding it’s highly recommended to go through Resources all. Thankyou“

Similar Paper with basis for explanation about On- Line Learning Back Propagation Algorithm:

* Garcia, S. D. (n.d.). An Online Backpropagation Algorithm with Validation Error-Based Adaptive Learning Rate. *Orange Labs, 4, Rue du Clos Courtel, 35512 Cesson-S´evign´e, France*. http://liris.cnrs.fr/Documents/Liris-6097.pdf
* Deep residual learning for Image Recognition, Shortcut mapping in Neural networks [https://arxiv.org/pdf/1512.03385.pdf](https://arxiv.org/pdf/1512.03385.pdf%20) - **MICROSOFT RESEARCH**
* <https://blog.waya.ai/deep-residual-learning-9610bb62c355,basic> , Back propagation calculus: welch labs: [https://www.youtube.com/watch?v=bxe2T-V8XRs&list=PLiaHhY2iBX9hdHaRr6b7XevZtgZRa1PoU](https://www.youtube.com/watch?v=bxe2T-V8XRs&list=PLiaHhY2iBX9hdHaRr6b7XevZtgZRa1PoU%20)
* Fisher Information matrix , Log-likelihood, MATLAB implementation of Deep Neural Networks, Normalization, Regularization : <http://cs231n.github.io/neural-networks-2/>
* <https://in.mathworks.com/help/nnet/ref/trainbfg.html> : MATLAB Newton’s method

***Feedback Appreciated***

***Contact: kdkunal.94@gmail.com EFFORTS BY: KUNAL DARGAN***