Specific features of feed-forward neural networks application in classification and regression problems in algology

N Yu Grigoryeva¹, T R Zhangirov² and A A Liss²

¹Saint-Petersburg Federal Research Center of Russian Academy of Sciences, Scientific-Research Center for Ecological Safety, 18 ul. Korpusnaya, Saint-Petersburg 197110, Russia
²Saint-Petersburg Electrotechnical University “LETI”, 5 ul. Prof. Popov, Saint-Petersburg 197376, Russia

E-mail: renes3@mail.ru

Abstract. Abundance and biological diversity of phytoplankton communities, investigated in this work, are often used as a marker for the determination of environmental health and fresh water quality. Presently their routine analysis is very time consuming and expensive. A lot of articles are devoted to the development of a system for an in situ automated analysis of phytoplankton properties. However, the applied problems of biology, ecology and, in particular, algology usually are associated with some difficulties due to shortage and/or fuzziness of the experimental data. Hence, while using neural network modelling a set of specific problems can occur. In this article on the base of experimental data several such problems are presented and possible solutions are suggested. In particular, the illogical behavior of classifying neural network is revealed, while studying the biological diversity of cyanobacteria, and the original technique for results validation is presented. This problem is investigated on a set of spectroscopic data, recorded by means of confocal laser scanning microscopy. The generalization quality of the trained model is studied as the main learning parameter. Another problem of shortage dataset is examined in the frames of regression model for bioplankton abundance. This problem is solved by means of feed-forward back-propagation neural networks with two hidden layers. The modelling was carried out on a small experimental selection (only 39 observations were available), despite this, the relatively high determination coefficient was obtained for the training and test samples, while using dropout layout.

1. Introduction
Phytoplankton abundance and its biological diversity are the most common issues in ecological context, due to their importance for environmental health assessment in various water bodies. They are universally accepted indicators for an algae community response to environmental changes and for estimation of a system-level eutrophication. The quantification of temporal and spatial phytoplankton distribution within lakes and other reservoirs is important for estimating primary production and dominating phytoplankton species, since it is well established that excessive algal growth (algal booms) can be harmful to aquatic ecosystems. [1–3].

Nowadays the direct measurement of the biological processes is still not only difficult, but also expensive and time-consuming, thus the need for new methods is evident [4]. Moreover, the
complexity of current data acquisition and processing techniques makes effective on-line monitoring and control impossible. A lot of articles are devoted to the development of an in situ automated systems for analyzing the phytoplankton properties [5, 6]. However, most of the applied problems in biology, ecology and, particular, in algology usually are associated with some difficulties due to shortage and/or fuzziness of the experimental data. Recently, novel fluorescence methods were suggested to automate the data acquisition process for environmental monitoring of phytoplankton [7, 8]. Apart from this, several new data processing methods were elaborated to improve the automatic algae species discrimination and phytoplankton abundance prediction [9–13].

From mathematical point of view the estimation of phytoplankton abundance and biological diversity represent the problems of regression and classification, correspondingly. In general, both of these problems deals with a set of observations which are characterized by several features. The classification of observations is a vital application problem in most scientific areas. While solving the classification problem each observation should be mapped to some known target class [14, 15]. The solution of this problem is associated with the elaboration of a correct model with minimal classification errors, as well as with verification of the results of applying such model to the known and unknown observations. The regression problem is a some kind of prediction model, where the solution process is reduced to the determination of a certain function, which mapping a set of independent variables (features) to a set of dependent variables (output parameters). The solution of this problem deals with the minimization of the error between measured and predicted values. Thus the only difference between these two cases is that mapping in classification problem is conducted into discrete finite set of classes, while in regression problem the output values belong to a continuous set.

In the field of computer modeling both of these problems are well-known to be successfully solved by application of feed-forward artificial neural networks (ANN) with backpropagation [1, 2, 12, 13, 16–18]. And in both cases supervised learning is used to train ANN. However, some peculiarities make this analysis not so easy and straightforward. First of all, shortage and/or fuzziness of the experimental data may produce uncertain and ambiguous results. But in contrast to other mathematical models ANN does not need rigorous data preprocessing [15, 19]. Else, the weak predictability of the results quality for concrete ANN model is one of the central problems of the machine learning, which enforce the necessity to obtain a vast amount of trained models with following selection of the best ones. And the last but not least problem is that the neural network is a black-box model, and presently there are no effective methods to analyze and control its operation. All these problems should be solved in each special case separately.

In this paper we present some common hints of solving ANN problems listed above on the base of two examples the taxonomic classification of cyanobacteria and the phytoplankton abundance prediction. In particular, during the solution of classification problem the illogical behavior of classifying neural network was detected, while studying the biological diversity of cyanobacteria. The generalization quality of the trained model is studied as the main learning parameter. Here the illogical behavior means that the trained ANN gives a high classification accuracy and has a good ability for internal generalization, but at the same time fails in external generalization [20, 21]. This problem is investigated on a set of spectroscopic data, recorded by means of confocal laser scanning microscopy. In this work we suggest a special results validation technique to reveal such behavior among correctly trained models. During the solution of the regression problem for prediction of phytoplankton abundance some difficulties occur due to the shortage of experimental dataset (only 39 observations were available). This problem is solved by inserting of the dropout layer into ANN with two hidden layers [22]. Although the modelling was carried out on a small experimental dataset, a relatively high determination coefficient was obtained for the training and test samples.
2. Material and Methods

2.1. Data acquisition and treatment

For the classification problem 307 sets of seven single-cell fluorescentce spectra for cyanobacterial cells, belonging to 23 different strains (classes), were obtained by means of confocal laser scanning microscopy (CLSM). Cyanobacterial strains were provided by Core Facility Center for Culture Collection of Microorganisms of the Research Park of St. Petersburg State University. In the CALU collection the cyanobacterial strains are numbered and for the clarity of further narration the CALU numbers for corresponding strains were used below. To record self-fluorescence spectra of living cyanobacterial cells CLSM Leica TCS-SP5 was used. The obtained intrinsic single-cell fluorescence spectra are convenient for statistical processing and for subsequent computer analysis via various mathematical methods. For each cell 7 fluorescence emission spectra, corresponding to 7 excitation wavelengths (405, 458, 476, 488, 496, 514, 543 nm) were analyzed. Each initial spectrum is an array of 38-45 numbers. The main problem of this dataset is due to the high non-uniformity of the initial data and non-balanced class sizes (different number of observation in each class). The data preprocessing included the extraction from initial dataset the integral and statistical characteristics, which describe the shape of each spectrum [23]. A set of 63 features, describing the shape and statistical characteristics of single-cell seven-spectra set, has been determined for each observation. All 63 features were fitted to the range of [-1,1], which is convenient for ANN training.

For the regression problem the experimental observations were obtained at three biological scientific stations on southern and middle knee of the Kola bay littoral (The Barents sea). Data was collected every month from September, 2012 to September, 2013 – the total sample consists of 39 observations [24]. The input data include physicochemical parameters of the water (temperature, salinity, pH, dissolved oxygen, oxygen saturation of the water, biochemical oxygen consumption, ammonium nitrogen, nitrite nitrogen, nitrate nitrogen, phosphates), as well as the target phytoplankton abundance to be predicted. Obviously, to build a correct model the input data should be normalized, since some of the features initially differ by an order of magnitude. In this work, the linear normalization was carried out to bring the input and output values to the range of [0.1, 0.9].

2.2. Models development

Feed-forward ANN is a system of functional elements (so-called neurons), combined in layers and connected with each other in the adjacent layers. In such networks, signal propagates from input layer to output layer. The number of ANN layers and neurons on them can varied (figure 1).

![Common neural network architecture](image)

**Figure 1.** Common neural network architecture.

For each neuron of the network (except input layer neurons), the incoming signal is a weighted sum of signals from previous-layer neurons. While neuron proceeding input signal, it transforms it in accordance with activation function:
\[ z_i = F \left( \sum_{j=1}^{L} w_{ij} h_j \right), \]  
\[ (1) \]

where \( z_i \) – output signal of \( i \)-th neuron, \( h_j \) – value of \( j \)-th input signal, \( w_{ij} \) – weight between \( i \)-th and \( j \)-th neuron of two adjacent layers, \( L \) – number of neurons in previous layer, \( F \) – activation function on \( i \)-th neuron. Due to specific architecture of ANN, as a set of layers and a set of neurons on them, and due to the nonlinearity of activation functions the whole model have a high level of nonlinearity. The input layer size is given by the length of feature vector. And the size of the output layer is defined by a number of output parameters of the considered problem.

It should be mentioned here, that the weights initialization, i.e. the distribution law, by which the initial weights are generated, also plays an important role in an ANN design. The ANN training process as well as the obtaining result depends on the weights initialization. And the selection of weights initialization depends on ANN architecture, training parameters and input data characteristics.

2.2.1. ANN for classification problem. For multiclass classification problem (more than two classes), size of ANN's output layer equals to a number of target classes. In considered classification problem a feature vector size is 63, and the number of target classes is 16, so the neural network has 63 and 16 neurons on the input and output layers, respectively. During the investigation of various architectures, a model with one hidden layer with 31 neurons was selected, as the best fitting for considered problem. As an activation functions the hyperbolic tangent is used at hidden layer and Softmax function is used at output layer:

\[ F_j(s) = \exp(s_j) / \sum_{j=1}^{L} \exp(s_j), \]
\[ (2) \]

where \( s_j \) – state of neuron (weighted sum) on \( i \)-th neuron, \( s_j \) – state of neuron on \( j \)-th neuron , \( L \) – number of neurons in layer. Softmax is conventional function for multiclass classification problem, as the output vector of this function for one observation can be interpreted as a probabilities distribution of belonging to target classes.

During the preliminary study LeCun and He weight initializations were rejected [25]. Among three weight initializations the Xavier one was selected:

\[ W \sim U \left( -\sqrt{\frac{6}{L_k + L_{k+1}}} , \sqrt{\frac{6}{L_k + L_{k+1}}} \right) \],
\[ (3) \]

where \( U \) – uniform distribution, \( k \) – layer number, \( L_k \) – size of \( k \)-th layer.

For model optimization adaptive moment estimation method (Adam) with the Nesterov amendment was applied [26]. This is a first order method, that approximate the second order methods, i.e. it uses not only gradient of error, but also simulates an Hessian of error. It has been successfully used in computational biology [27]. This optimization method is tuned with following parameters: learning rate \( \eta = 0.05 \), momentum coefficient \( \nu = 0.001 \) and regularization coefficient \( \rho = 0.001 \). As a metric for error calculation the categorial cross-entropy is used:

\[ E = -\sum_{i=1}^{L} y_i \ln y_i , \]
\[ (4) \]

where \( y_i \) and \( y_j \) is obtained and expected value on \( i \)-th neuron of the output layer, respectively, \( L \) – size of output layer. It should be noted, that cross-entropy is better than mean square error for training of classifying neural networks, because it allows to measure distance between two distributions of probability. Whole dataset partition on training and test dataset is formed randomly in ratio 7:3. Final validation of the results was conducted on datasets belonging to new classes, that are not involved in training process. ANN model and traning algorithm were implemented on MATLAB software.
2.2.2. **ANN for regression problem.** To determine the best fitting ANN architecture for considered regression problem, the neural networks with one, two and three hidden layers and different number of neurons on them were investigated. As a result, ANN with two hidden layers with 9 and 5 neurons on them, respectively, was chosen. According to feature vector size, the number of input-layer neurons was set to 10. Since there is only one dependent variable in the considered problem, the output layer contains one neuron. Similarly to classification problem, the hyperbolic tangent was used here as the activation function on hidden layers. Usually a linear function is recommended for building regression models, but since the output values in the considered problem belongs to the interval [0, 1], the activation function Relu was initially selected: \( F(x) = \max(0, x) \). However, later it was revealed that such zeroing of negative values affects the learning process incorrectly, namely, in some cases, ANN stops learning and for all observations of the dataset gives a zero result. To fix this problem, the activation function at the output layer was replaced with ELU [28], which has a non-zero gradient for negative values. During this study the uniform and normal distribution laws with various parameters were considered for weights initialization. Finally, it was found, that different models with the same architecture are trained most stably, when the normal distribution law with a standard deviation of 0.65 and mean value 0 for initializing weights on the basic neurons was used, and the uniform distribution in the range [−1.5, −0.5] for a bias neurons at the hidden layers and the fixed value of 0.35 for a bias neuron at the output layer were applied.

Model was trained with NAdam (Nesterov-accelerated Adam) method [29] at learning rate \( \eta = 0.0003 \) (for this method there is no momentum coefficient) without regularization Also NAdam is the first order method like Adam that approximate second order methods. Whole dataset partition on training and test datasets was done by datasets belonging to different experimental stations. Training dataset consists of the data from two selected stations and for test dataset the data from one station was used. The considered neural network and selected traning algorithm were implemented on Python using Keras.

3. **Results and discussion**

Usually the analysis of the trained ANN model quality starts from estimation of error variation during training process. On this step overfitting (i.e. when error for test dataset starts to increase after reaching some minimal point, while for training dataset it still decreases) or underfitting (i.e. when error for training dataset stops changing and stay at high level) may be detected. In addition to the error function some other metrics can be used during training process, that better represents the model quality and easy to interpret. For classification problem such additional metric is a classification accuracy and for regression problem is a determination coefficient. They both better represents the model quality and helps to estimate the generalization quality of the trained ANN.

Next step is the analysis of model’s generalization quality. Generalization is very important characteristic of ANN, since it reflects the model’s ability to process correctly new data. Unfortunately, there is no unified method to check generalization quality in the case of small dataset (number of samples < 500) considered in this investigation, thus we will apply different ones in each special case.

3.1. **ANN illogical behavior in classification problem**

The ordinary shape of error function for models used in this work is presented on figure 2(a). Such slope shows that the model was trained correctly. At the same plot the variation of accuracy during learning process is shown, which reaches 97% for this model.

For classification problem it is a common practice in addition to error function build a confusion matrix, that shows how much observations are wrongly classified. This matrix can be built for any classifying methods and can be used for calculation of precision, recall and F1 metrics. This metrics describes quality of model better than accuracy, but only for enough big datasets. This leads to the fact that metrics calculated for entire dataset do not provide information about probability of false prediction of observation. In the case, when classifier is based on ANN with Softmax function at the
output layer it is possible to calculate averaged probability distribution within each class (or averaged class likelihood), i.e. the averaged probability of assigning the observations of a specific class to corresponding target classes. That averaged probability can be represented as a bar chart (see figure 2(b)).

Figure 2. ANN classification results for observations from 16 target classes, corresponding to different cyanobacterial strains. (a) – cross-entropy (left axis) and classification accuracy (right axis) versus number of epochs (Model 1); averaged class likelihood bar charts: (b) – for target classes (Model 1); (c) – for unknown classes (Model 1); (d) – for unknown classes, included in training process (Model 2). The numbers on the x-axis and in the legend indicate numbers of strains from CALU collection.

Another way to improve generalization quality control is to use observations that doesn’t belong to any target class, but information about similarity of these observations to known ones are available. In this work the similarity information was previously obtained by using linear discriminant analysis and calculating Mahalanobis distance [21]. This method of analysis can detect deep logical problems of trained ANN models. Classification results for additional seven unknown classes are presented on figure 2(c). This diagram reveals problems for classes with numbers 601 and 550. Comparing the results for previously selected model with results another model with less overall accuracy 95%, it can be noticed that the last model shows better generalization quality (figure 2(d)). Such behavior of the first ANN model can be called the illogical, which is due to the incorrect weights optimization during training process, and as a consequence, leads to low generalization quality.

3.2. Rising ANN generalization quality in regression problem
For regression problem the determination coefficient was used as an additional metric, like an accuracy in classification problem, and was tracked during the whole training process (figure 3(a)). According to its variation the beginning of overfitting can be clearly revealed around 1000-th epoch. In this area, the determination coefficient for the test sample reaches its maximum value (0.458), while the determination coefficient for training dataset is 0.579.
Figure 3. Training results for regression models, predicting phytoplankton abundance: (a), (b) – Model without dropout layer; (c), (d) – Model with dropout layer. (a), (c) – variation of the determination coefficient during training process, (b), (d) – plot of measured and predicted values at the optimal epoch for three experimental stations (Station 1, 2 – training sample; Station 3 – test sample).

A plot of measured and predicted values calculated at the optimal 1060-th epoch (figure 3 (b)) shows, that despite of rather low values of the determination coefficients, most of all experimental points are close to the test sample. The predicted values for the training dataset are close enough to the measured values. However, for several higher values of phytoplankton abundance the prediction error increases noticeably. The main problem of this model is that overfitting starts very fast (the shape of determination coefficient curve near the optimal epoch is very sharp). Usually, if the determination coefficient is much higher for training dataset than for test sample the generalization quality is low. Moreover, at the early epoch a high model error occur. In case of sharp shape of the determination coefficient curve near optimal epoch, more accurate procedure of early stopping and obtaining best model with good generalization is required. One of the existed methods to improve generalization quality of ANNs and make learning curve smoother is the use of dropout layer [22]. This layer does not perform any non-linear transformations, but placing between layers it randomly zeroing input signals at each neuron during training process. Thereby, this layer prevents neurons in a subsequent layer from learning only by one hidden feature. In this research, a dropout layer was added after the first hidden layer. The zeroing probability for each neuron was set at 10%. For the considered model, training results in the presence of a dropout layer are presented in figure 3(c). For this model a significant increase in the determination coefficient on the test sample was observed. At the optimal epoch (10000) the determination coefficient increases up to 0.9 and 0.639 for training and test dataset, respectively.

Comparison of the prediction results at the optimal epoch for the model without and with dropout layer (figure 3 (b) and figure 3 (d)), shows significant result improvement in the presence of dropout
layer, i.e. the predicted values lie closer to measured data for both datasets (training and test). There still are some problems with highest abundance values in the test dataset, but seems they are due to the restriction of training process. The effectiveness of dropout application for small samples is due to the fact, that during the training process an indirect augmentation of the input data occur. As a result of adding dropout layer, two problems were solved: the sharp slope of determination coefficient curve was smoothed and the generalization quality was significantly improved.

4. Conclusion
The use of automatic physical and chemical measuring systems with following ANN data processing has provided a substantial increase in the ability to monitor effectively the environmental parameters of coastal and fresh waters. However, some problems during ANN modelling can occur, such as illogical behavior and low generalization quality. In each special case these problems should be solved separately according to the required results. In this work we demonstrate two simple solutions of the common biological problem of shortage and fuzziness of the experimental data.

References
[1] Kang H Y, Rule R A and Noble P A 2012 Artificial neural network modeling of phytoplankton blooms and its application to sampling sites within the same estuary Treatise on Coastal and Estuarine Science (vol 9) ed M Wolanski (Amsterdam: Elsevier) chapter 9.09 pp 161–171
[2] Yabunaka K I, Hosomi M and Murakami A 1997. Novel application of a back-propagation artificial neural network model formulated to predict algal bloom Water Sci. Technol. 36(5) 89–97
[3] Kılıç H, Soyupak S, Gürbüz H and Kıvrak E 2006. Automata networks as preprocessing technique of artificial neural network in estimating primary production and dominating phytoplankton levels in a reservoir: An experimental work. Ecol. Inform. 1(4) 431–9
[4] Culverhouse P F, Williams R, Reguera B, Herry V and Gonzalez-Gil S 7-9 July 2003. Expert and machine discrimination of marine flora: a comparison of recognition accuracy of field-collected phytoplankton Proc. IEEE Int. Conf. Visual Information Engineering, Guildford, UK 177–181
[5] Wilkins M F, Morris C and Boddy L 1994. A comparison of radial basis function and backpropagation neural networks for identification of marine phytoplankton from multivariate flow cytometry data. Bioinformatics 10(3) 285–94
[6] Embleton K V, Gibson C E and Heaney S I 2003. Automated counting of phytoplankton by pattern recognition: a comparison with a manual counting method J. Plankton Res 25(6) 669–81
[7] Grigoryeva N Y 2019 Self-fluorescence of photosynthetic system – a powerful tool for investigation of microalgal biological diversity Microalgae - From Physiology to Application ed V Milada (London: IntechOpen) chapter 1 pp 3–26
[8] Grigoryeva N Y and Chistyakova L V 2018 Fluorescence Microscopic Spectroscopy for Investigation and Monitoring of Biological Diversity and Physiological State of Cyanobacterial Cultures Cyanobacteria ed A Tiwari, (London: IntechOpen) chapter 2 pp 11–43
[9] Millie D F, Schofield O M, Kirkpatrick G J, Johnsen G and Evens T J 2002 Using absorbance and fluorescence spectra to discriminate microalgae Eur. J. Phycol. 37(3) 313–22
[10] Balfoort H W, Snoek J, Smiths J R M, Breedveld L W, Hofstraat J W and Ringelberg J 1992 Automatic identification of algae: neural network analysis of flow cytometric data J. Plankton Res. 14(4) 575–89
[11] Culverhouse P F, Simpson R G, Ellis R, Lindley J A, Williams R, Parissini T, Reguera B, Bravo I, Zoppoli R, Earnshaw G and McCall H 1996. Automatic classification of field-collected dinoflagellates by artificial neural network. Mar. Ecol. Prog. Ser 139 281–7
[12] Huang J C, Gao J F, Mooij W M, Hörmann G and Fohrer N 2014. A comparison of three
approaches to predict phytoplankton biomass in Gonghu Bay of Lake Taihu. *J. Environ. Inform.* **24**(1) 39–51

[13] Mattei F, Franceschini S and Scardi M 2018. A depth-resolved artificial neural network model of marine phytoplankton primary production *Ecol. Modell.* **382** 51–62

[14] Langford J 2005 Tutorial on practical prediction theory for classification *J. Mach. Learn. Res.* **6** 273-306

[15] Dreiseitl S and Ohno-Machado L 2002 Logistic regression and artificial neural network classification models: a methodology review *J. Biomed. Inf.* **35.5-6** 352–9

[16] Jeong K S, Kim D K and Joo G J 2006. River phytoplankton prediction model by Artificial Neural Network: Model performance and selection of input variables to predict time-series phytoplankton proliferations in a regulated river system *Ecol. Inform. 1*(3) 235–45

[17] D'Alimonte D and Zibordi G 2003 Phytoplankton determination in an optically complex coastal region using a multilayer perceptron neural network *IEEE Trans. Geosci. Electron.* **41**(12) 2861–2868

[18] Millie D F, Weckman G R, Paerl H W, Pinckney J L, Pigg R J and Fahnstiel G L 2006 Neural net modeling of estuarine indicators: Hindcasting phytoplankton biomass and net ecosystem production in the Neuse (North Carolina) and Trout (Florida) Rivers, USA *Ecol. Indic.* **6**(3) 589–608

[19] Hajmeer M and Basheer I 2003 Comparison of logistic regression and neural network-based classifiers for bacterial growth *Food Microbiol* **20**(1) 43–55

[20] Altman E I, Marco Gand Varetto F 1994 Corporate distress diagnosis: Comparisons using linear discriminant analysis and neural networks (the Italian experience) *J. of banking & finance* **18**(3) 505–29

[21] Grigoryeva N Y, Zhangirov T R, Perkov A S, Ivanova S A and Liss A A 2019 Classifying neural networks and methods of their illogical behaviour revealing *J. Phys. Conf. Ser.* **1352**(1) 012024

[22] Srivastava N, Hinton G, Krizhevsky A, Sutskever I and Salakhutdinov R 2014 Dropout: a simple way to prevent neural networks from overfitting *J. Mach. Learn. Res.* **15**(1) 1929–58

[23] Zhangirov T R, Perkov A S, Ivanova S A, Liss A A, Grigoryeva N Y and Chistyakova L V 2019 Comparison of the efficiency of solution of classification task by methods of linear discriminant analysis and artificial neural networks *Izvestiya SPbGETU "LETI"* **5** 64–73 (in Russian)

[24] Lutsenko E S and Moskvina M I 2014 Annual dynamics of bacterioplankton and bacterioperiphyton populations of rock substrates on littoral of the Kola bay (The Barents sea) *Fundamental Research* **8** 79–83

[25] Glorot X and Bengio Y 2010 Understanding the difficulty of training deep feedforward neural networks *Proc. of the thirteenth international conference on artificial intelligence and statistics* **3** 249–256

[26] Kingma D and Ba J 2015 Adam: A method for stochastic optimization *Proc. Int. Conf. on Learning Representations* (San Diego:USA/arXiv) p 434

[27] Angermueller C, Parnamaa T, Parts L and Stegle O 2016 Deep learning for computational biology. *Molecular Systems Biology* **12**(7) 27474269

[28] Clevert D A, Unterthiner T and Hochreiter S 2015 Fast and accurate deep network learning by exponential linear units (ELUs) arXiv preprint arXiv:1511.07289. 2014 Available at: https://arxiv.org/abs/1511.07289 (accessed 14.05.20)

[29] Sutskever I, Martens J, Dahl G and Hinton G 17-19 June 2013 On the importance of initialization and momentum in deep learning *Proc. Int. Conf. on machine learning*, Atlanta, Georgia, USA 1139–1147