BAYESIAN APPROACH IN NEURAL NETWORKS *

UDC 330.4 *DOI: 10.52063/25792652-2024.2.21-187*

NARE DAVTYAN
Yerevan State University,
Faculty of Economics and Management,
Chair of Mathematical Modeling in Economics, Ph.D. Student;
«Information Systems Agency of Armenia», Data Analyst,
Yerevan, the Republic of Armenia,
naredavtyan987@gmail.com
ORCID: 0000-0002-9757-4654

With the development of Deep learning, neural networks have become very popular. But nowadays neural networks which are being used are based on standard approach of Statistics. The main purpose of this work is to present Bayesian approach, highlight the main differences between Bayesian and frequentist approaches, their principles. The problem was set to show in which cases Bayesian neural networks can be more preferable.

The purpose was achieved through the following stages. The main steps of neural networks are presented. After it the fundaments of Bayesian approach are described. As Bayesian approach is a common concept, not just an inference used in neural networks, initially author speaks about this approach generally and then tells about its usage in neural networks. After that the main advantages and limitations of Bayesian networks in Deep learning are spoken about: which problems they can solve.

The main conclusion of this paper is that Bayesian approach could be used to avoid overfitting. However, it is essential to understand the specific conditions under which Bayesian neural networks are preferable.

Keywords: *Deep learning, Machine learning, Bayesian and standard (frequentist) approaches, neural network, loss function, activation function, overfitting, prior distribution, likelihood, posterior distribution.*

INTRODUCTION

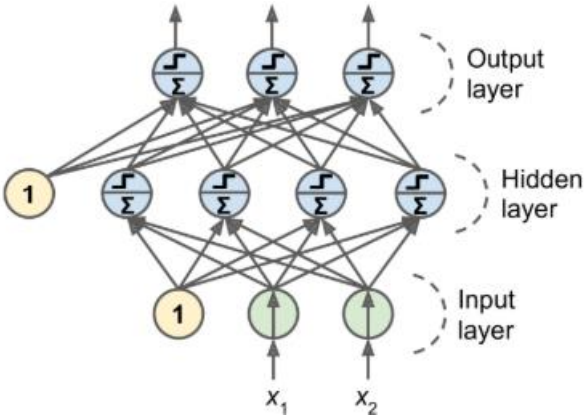
Nowadays a huge amount of data is being accumulated in different areas of the economy. And it is important for business managers to use the data about their companies effectively. This is one of the reasons why Deep learning algorithms have been developed. One of the most popular Deep learning algorithms are neural networks. In 1943 neurophysiologist Warren McCulloch and mathematician Walter Pitts introduced the first artificial neural networks. The logic behind these neurons was like the neurons of the human brain. Neurons get the information in the form of signals from the other neurons and then produce their own signals. (Geron 277-278) In the case of artificial neural networks (ANN) they get the data as input and then recycle these data and transfer it to the next layer of network. So, this simple logic has led to the development of neural networks with their many variations. Firstly, neural networks were introduced in the standard or frequentist approach of statistics. But in the late 1990s, with the development

* Հոդվածը ներկայացվել է 05.06.2024թ., գրախոսվել՝ 17.06.2024թ., տպագրության ընդունվել՝ 31.07.2024թ.:

of Bayesian inference, this approach also started to find its applications in neural networks. Standard and Bayesian approaches provide the main principles used in statistical analysis, Deep learning, and Machine Learning. Depending on the size of the dataset, our prior knowledge, the main purpose of our analysis and other factors, one of frequentist and Bayesian approaches will provide better and more accurate results. The main differences of those methods in Deep learning will be discussed afterwards in this paper.

NEURAL NETWORKS

Neural networks help us to model both linear and non-linear connections between variables. Although linear models are simple and comparatively easy to estimate, the relationships between real-world data are non-linear. That is why neural networks have more applications. As we have already mentioned earlier, neurons get input data and after processing it, return prediction results. Neural networks contain layers of neurons. The first layer is the input layer, where we input our data to be processed. The last layer is called output layer which returns model prediction results. Between the input layer and output layer we have hidden layers for performing computations. Below is the simple structure of neural network (circles represent data points and are called nodes).



The structure of neural network (Geron 286).

Neural networks are parametric models which help us to approximate the mapping: $X \to Y$ given a dataset $D = \{x_i, y_i\} \subset (X, Y)$ (Back, Keith, 16).

Now let us discuss the main steps of neural networks in a more detailed way.

- The input layer processes the given data and calculates the weighted sum of the data points: $z = x^t w + b = w_1 x_1 + w_2 x_2 + \cdots + w_n x_n$. In this equation x^t is the transposed matrix of data points, w is the vector of initially generated weights (those weights decide how each layer is connected to its next layer and usually are randomly generated for the first step), and b is bias (this is not mandatory). Using this formula in the first layer the activation or step function is being calculated. The most common step functions are:

$$heavside(z) = \begin{cases} 0; if\ z < 0 \\ 1; if\ z \geq 0 \end{cases}$$

$$sgn(z) = \begin{cases} -1; if\ z < 0 \\ 0; if\ z = 0 \\ 1; if\ z > 0 \end{cases}$$

- The result of activation function is being transferred to the next layer. Based on these results the nodes of the next layer are activated. And the process is repeated until the algorithm reaches the output layer.
- This process is called feed forward algorithm. After reaching the output layer, the loss function (output error) is calculated. For the loss function there also are many options: the most common one is sum of the squares of the differences between the actual (y) and predicted $(\hat{y})$ values.

$$L = \sum_{i=1}^{n} (\hat{y}_i - y_i)^2$$

- Then comes the backpropagation part of the algorithm. The algorithm tries to investigate how much each connection of layers contributes to the error calculated in the previous step. For that gradient of the loss function are being calculated across all the connections weights.
- Here weights are being updated based on the gradients. As our main goal is to minimize loss function, weights are updated in a way which minimizes the error (loss function) of the model (Geron 283-287).
- The process repeats until we reach initially defined threshold of loss functions or the number of epochs (one pass through the layers is called an epoch). Epochs' count and threshold of the loss function are model hyperparameters, which are being defined by the researcher initially. The desired value for these hyperparameters may be different based on the purpose of research.

In this part we have discussed the main principles of neural networks in the standard approach of statistics. Bayesian inference and its applications in neural networks will be discussed afterwards.

BAYESIAN INFERENCE AND NEURAL NETWORKS

Bayesian inference is one of two main approaches in statistical analysis. The main difference between the standard and Bayesian approaches is that in the case of the standard approach model parameters are unknown but fixed. In Bayesian approach model parameters are referred as random variables and instead of predicting one value for each parameter, Bayesian approach gives us an opportunity to get a distribution function for each of them. It helps us to use our prior knowledge about the model parameters in the Bayesian model which will lead to better results.

So, in case of Bayesian statistics, having some prior knowledge, we update this knowledge based on the observed data (Steorts 16-20).

Mathematically, Bayes' rule is the relation between the prior information and the posterior reallocation of credibility conditional on data.

Bayes' rule has many different ramifications in statistics: so, this concept is one of the most important fundaments in statistics. For Bayes' rule let us recall the main formula of conditional probability:

$$P(A|B) = \frac{P(A,B)}{P(B)}$$

The probability of the event A given B is equal to the probability that they happen together relative to the probability that B happens at all. After some algebraic changes we obtain:

$$P(A|B)P(B) = P(A,B)$$

Then we do the same with the probability of event B given event A. After the same changes we get:

$$P(B|A)P(A) = P(A,B)$$

The right-hand sides of the last two expressions are the same, so we can write:

$$P(A|B)P(B) = P(B|A)P(A)$$

Now divide both sides of last equation by $P(B)$:

$$P(A|B) = \frac{P(B|A)P(A)}{P(B)}$$

$$P(A|B) = \frac{P(B|A)P(A)}{\sum_a P(B|a)}$$

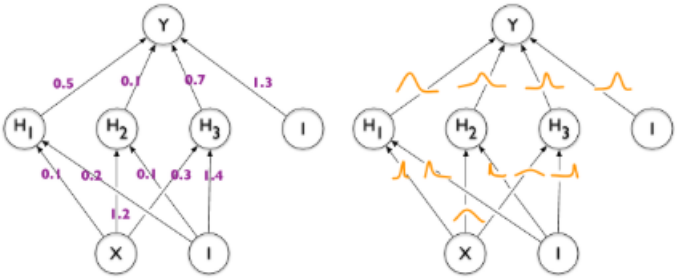
In the numerator of the last equation A is a fixed value, whereas in the denominator A takes all values. This simple equation is the basis of Bayes' theorem.

Changing events A and B in the Bayes' rule into the data (D) and parameters (θ), we get an equation which corresponds to the Bayesian inference better.

$$P(\theta|D) = \frac{P(D|\theta)P(\theta)}{\sum_{\theta^*} P(D|\theta^*)P(\theta^*)}$$

The factors in this equation have specific names.

- $P(\theta)$ - *prior*, expresses our initial belief about the parameter: the probability of the parameter value before observing the data.
- $P(D|\theta)$ - *likelihood*: the probability that the observed data could be generated with parameter value equal to θ.
- $P(\theta|D)$ - *posterior*: the probability of parameter value θ considered the observed data. Shows the probability of getting the parameter value θ on condition of the observed data.
- $P(D) = \sum_{\theta^*} P(D|\theta^*)\,P(\theta^*)$ - *marginal likelihood*: the overall probability of the data determined by averaging across all possible parameter values weighted by the strength of belief in those parameter values (Kruschke 274-277) (Brewer 15-16).
- In neural networks in Bayesian analysis, we express our prior beliefs about model parameters (in this case about network weights) by introducing prior distributions over the weights of the layers. In the picture below standard (left) and Bayesian (right) neural networks are shown (Back, Keith 29).



Difference of two main approaches in neural networks (Back, Keith 29).

THE MAIN ADVANTAGES AND LIMITATIONS OF BAYESIAN APPROACH IN DEEP LEARNING

Bayesian neural networks are being used in various areas from image recognition and game development to stock price prediction. In Bayesian inference we use our prior knowledge (based on experts' opinion, our initial assumptions, historical data, etc.) to define prior distribution of parameters.

Bayesian neural networks are valuable for addressing issues in fields with limited data, helping to avoid overfitting. In Deep learning and Machine learning algorithms the dataset is divided into 2 parts: training, on which model is trained, and test, on which we make predictions and test the obtained results. In the case of standard neural networks model is being trained on the same data for many epochs and "learns" all the main features of this data that there is a possibility that model will overfit and show low accuracy on the new data. Overfitting is the issue when the model performs well on training data set but fails on test data. Although in case of Bayesian neural networks model trains on the same data many times, here we add out prior knowledge to get posterior distribution. So, the prediction is not only based on the information coming from the training data and the risk of overfitting is reduced. If our main goal is making predictions (and not revealing relationships between variables), overfitting may cause serious problems.

There are some experiments using Bayesian neural networks. One of them is filtering junk email. As we know, junk emails are being determined via key words (e.g., *win, free money*). And for this problem the Bayesian approach is quite a workable solution because for the models where our input variables are words, we need to present them as numbers (detailed discussion of this method is out of the scope of this work). And usually, these numbers are sparse. And in that research authors have trained Bayesian neural networks which have shown 92% accuracy (the model has classified 92% of truly junk emails as junk) (Sahami, Dumais 58-60).

BNNs are effectively being used in stock price prediction. An experiment was held for comparing the performance of BNNs and Artificial neural networks. Stocks of 6 companies (Apple, Facebook, Microsoft, J.P. Morgan, Procter & Gamble and Walmart) were selected for this research. As performance metric, the loss functions were selected. After prediction, for all 6 companies, the loss function in case of Bayesian neural networks was less than in case of Standard neural networks (Wang, Qi 225-226).

The main advantages of Bayesian neural networks are:

- They are particularly useful in molecular biology and medical diagnosis, where data collection is typically expensive and challenging.
- BNNs enable you to automatically compute the error associated with your predictions when handling data with unknown targets.

• They allow you to estimate uncertainty in predictions, which is a valuable feature for fields such as medicine (Databricks 2024).

Despite the effectiveness of Bayesian neural networks in some cases, they also have limitations. Here are some of them:

• While they can achieve superior results for numerous tasks, they are exceedingly challenging to scale to larger problems.

• Training speed. As Bayesian neural networks predict distribution and not a single value, it may take longer to train BNNs.

• In some cases, our prior predictions about model parameters' distributions may not quietly correspond to reality (for example, due to lack of information). This may result in a less accurate estimation.

CONCLUSION

Bayesian neural networks help to overcome the problem of overfitting. They can be used both for regression and classification problems. If we have enough prior information about the model parameters, for example, their distribution, and our training dataset is not so big, then Bayesian neural networks are recommended, instead of the standard neural networks. However, as we have already mentioned, Bayesian neural networks are based on researcher's prior knowledge or information. Sometimes this information could be incomplete and describe our dataset not so accurately. Before applying Bayesian approach in neural networks, researchers need to be sure that the initial information which will be applied (in the form of likelihood), describes all the main peculiarities of model parameters. So, Bayesian neural networks could be particularly useful if applied correctly.

REFERENCES

1. Aurelien Geron. *Hands-on Machine learning with Scikit-learn, Keras and Tensorflow*, 2nd edition, USA, 2019.

2. Alexander Back and William Keith, *Bayesian Neural Networks for Financial Asset Forecasting,* 2019.

3. Brendon J. Brewer. *Introduction to Bayesian Statistics,* 2013.

4. Databricks. *Bayesian Neural Network, 2024* https://www.databricks.com/glossary/bayesian-neural-network. Accessed: 04.06.2024.

5. John K. Kruschke. *Doing Bayesian Data Analysis,* 2nd edition, 2015.

6. Mehran Sahami, Susan Dumais and others. *A Bayesian Approach to Filtering Junk E-Mail*, 1998.

7. Rebecca C. Steorts. *Some of Bayesian Statistics: The Essential Parts*, 2016.

8. Zhang Wang, ZiYi Qi. *Future Stock Price Prediction on Bayesian LSTM and CRSP,* Beijing, China, 2023.

ԲԱՅԵՍՅԱՆ ՄՈՏԵՑՈՒՄԸ ՆԵՅՐՈՆԱՅԻՆ ՑԱՆՑԵՐՈՒՄ

ՆԱՐԵ ԴԱՎԹՅԱՆ
Երևանի պետական համալսարանի
տնտեսագիտության և կառավարման ֆակուլտետի
տնտեսագիտության մեջ մաթեմատիկական մոդելավորման ամբիոնի ասպիրանտ,
Հայաստանի տեղեկատվական համակարգերի գործակալության
տվյալների վերլուծաբան,
ք. Երևան, Հայաստանի Հանրապետություն

Խոր ուսուցման զարգացման հետ մեկտեղ նեյրոնային ցանցերը ստացել են բավականին լայն կիրառություն: Սակայն մեր օրերում օգտագործվող նեյրոնային ցանցերը հիմնված են վիճակագրության ստանդարտ մոտեցման վրա: Ավելի շատ վերլուծություններ կան ստանդարտ, քան բայեսյան մոտեցման կիրառմամբ: Այս աշխատանքի հիմնական նպատակն է ներկայացնել բայեսյան մոտեցումը, առանձնացնել բայեսյան և ստանդարտ մոտեցումների հիմնական տարբերությունները, դրանցում կիրառվող սկզբունքները: Սահմանված խնդիրն էր ցույց տալ, թե որ դեպքերում կարող են ավելի նախընտրելի լինել բայեսյան նեյրոնային ցանցերը:

Նպատակին հասնելուն օգնել են հետևյալ քայլերը: Ներկայացվում են նեյրոնային ցանցերի հիմնական փուլերը: Այնուհետև նկարագրվում են բայեսյան մոտեցման հիմունքները: Քանի որ բայեսյան մոտեցումը ընդհանրական հասկացություն է, այլ ոչ միայն նեյրոնային ցանցերում օգտագործվող եզրակացություն, սկզբում հեղինակը ներկայացնում է ընդհանուր առմամբ այս մոտեցման և ապա միայն նեյրոնային ցանցերում դրա օգտագործումը: Այնուհետև նշվում է խոր ուսուցման մեջ բայեսյան ցանցերի հիմնական առավելությունների և սահմանափակումների և այն խնդիրների մասին, որոնք դրանք կարող են լուծել:

Այս աշխատանքի հիմնական եզրակացությունն այն է, որ բայեսյան մոտեցումը կարող է օգտագործվել գերհարմարեցումից խուսափելու համար, սակայն պետք է իմանալ, թե որ պայմանների բավարարման դեպքում են բայեսյան նեյրոնային ցանցերն ավելի նախընտրելի:

Հիմնաբառեր՝ *խոր ուսուցում, մեքենայական ուսուցում, բայեսյան և ստանդարտ (հաճախական) մոտեցումներ, նեյրոնային ցանց, կորստի ֆունկցիա, սխալ, ակտիվացման ֆունկցիա, գերհարմարեցում, նախնական բաշխում, ճշմարտանմանություն, վերջնական բաշխում:*

БАЙЕСОВСКИЙ ПОДХОД В НЕЙРОННЫХ СЕТЯХ

НАРЕ ДАВТЯН
аспирант кафедры математического моделирования в экономике
факультета экономики и менеджмента
Ереванского государственного университета,
аналитик данных Агентства информационных систем Армении,
г. Ереван, Республика Армения

С развитием углубленного обучения нейронные сети стали очень популярными. Однако используемые современные нейронные сети основаны на стандартном подходе статистики; анализов в стандартном подходе больше, чем в байесовском.

Основная цель данной работы – представить байесовский подход, рассказать о главных различиях между байесовским и частотным подходами, их принципах. Была поставлена задача показать, в каких случаях байесовские нейронные сети могут оказаться предпочтительнее.

В достижении поставленной цели мы опирались на следующие этапы работы над исследованием: были представлены основные этапы работы нейронных сетей, далее описываются основы байесовского подхода. Поскольку байесовский подход является общей концепцией, а не просто методом вывода, используемым в нейронных сетях, то возникла необходимость сначала рассказать об этом подходе в общем, а затем – о его применении в нейронных сетях. Далее обсуждаются основные преимущества и ограничения байесовских сетей в глубоком обучении и то, какие проблемы они способны решать.

Основной вывод данного исследования заключается в том, что байесовский подход может быть использован для предотвращения переобучения, но необходимо учесть, при каких условиях байесовские нейронные сети более предпочтительны.

Ключевые слова: *глубокое обучение, машинное обучение, байесовский и стандартный (частотный) подходы, нейронная сеть, функция потерь, функция активации, переобучение, априорное распределение, правдоподобие, апостериорное распределение.*