

Bayesian Neural Network & Monte Carlo Dropout Regularization Technique

**Gubbala Durga Prasanth, Kendall Beaver,
V.S. Murali Krishna Chittlu**

INFO 410/510 Bayesian Modelling and
Inference, College of Information Science,
University of Arizona

Dr. Kunal Arekar
College of Information Science,
University of Arizona
Email: kunalarekar@arizona.edu

Office: Room 424, Harvill building

ABSTRACT

This project applies **Bayesian Neural Networks (BNN)** and **Monte Carlo Dropout (MC Dropout)** to predict loan approvals using the **Bank Loan Approval dataset** from Kaggle. The goal was to assess (1) the prediction effectiveness of the BNN and (2) the ability of MC Dropout to prevent overfitting. After fine-tuning the model, we found that applying MC Dropout (with rates from 30% to 60%) improved generalization, achieving 90% accuracy and a realistic loan approval rate of $\sim 8.5\%$. The results demonstrate the effectiveness of BNNs and MC Dropout in improving classification tasks.

1. INTRODUCTION

Initially, our group proposed using a Bayesian Neural Network (BNN) and Monte Carlo Dropout to predict employee stress levels using a dataset with global employee metadata. However, after finding no significant correlations in the data, we switched to the Bank Loan Approval dataset from Kaggle, which provided more relevant and correlated variables. This change greatly improved the performance of our BNN.

Our project questions were revised to focus on predicting loan approval and evaluating the effectiveness of MC Dropout in

preventing overfitting. While the original concepts and model parameters (e.g., weights and biases) remained unchanged, this shift to a more correlated dataset highlighted the power of BNNs in achieving better prediction results.

Modified Project Questions:

1. *How effectively can a neural network predict if a loan applicant will be approved or denied for a personal loan?*
2. *Can the Monte Carlo Dropout technique effectively regulate a neural network and keep it away from overfitting on training and testing data?*

Modified Dataset

The Bank Loan Approval dataset contains 5,000 observations and 12 variables related to financial and demographic information. The response variable, **Personal Loan**, is binary (1 for approved, 0 for denied).

Key features include:

1. Age (23 - 67)
2. Years of Work Experience (0 - 43)
3. Income (\$8,000 - \$224,000)
4. Family Size (1 - 4)
5. Credit Card Score (0 - 10)
6. Education Level (1 - 3)
7. Home Mortgage Value (\$0 - \$635,000)
8. Has Securities, CD, Online Banking, Credit Card Accounts (0 or 1)

1.1. Scope:

This project explores the use of Bayesian Neural Networks (BNN) and Monte Carlo Dropout for predicting loan approval decisions based on a variety of financial and personal attributes. The scope of the project includes:

- a) **Dataset:** We focus on the Bank Loan Approval dataset from Kaggle, which contains 5,000 observations and 12 features, such as income, credit score, and family size.
- b) **Modeling:** The primary objective is to train a BNN to predict whether a loan application is approved or denied, using Monte Carlo Dropout as a regularization technique to prevent overfitting.
- c) **Methodology:** The project involves training the model with different dropout rates across layers, fine-tuning hyperparameters (such as epochs, batch size, and learning rate), and evaluating model performance through accuracy and loss metrics.
- d) **Outcome:** The scope includes determining how well the BNN can predict loan approval and assessing whether the inclusion of MC Dropout improves model generalization by reducing overfitting.
- e) **Applicability:** While the model is focused on loan approval predictions, the techniques and methodologies developed can be applied to other classification problems with correlated and well-sampled datasets.

1.2 Objective:

The objectives of this project are to:

- a) Develop a Bayesian Neural Network (BNN) to predict loan approval using the Bank Loan Approval dataset.
- b) Evaluate the effectiveness of Monte Carlo Dropout (MC Dropout) in preventing overfitting and improving model generalization.
- c) Optimize model performance by fine-tuning parameters and dropout rates.
- d) Assess the applicability of BNNs and MC Dropout for other classification tasks.

1.3 Purpose:

The purpose of this project is to apply a Bayesian Neural Network (BNN) and Monte Carlo Dropout to improve the prediction accuracy and generalization for loan approval classification, demonstrating the effectiveness of these techniques in preventing overfitting and enhancing model performance.

2. OVERVIEW OF THE SYSTEM

A. Existing System:

Traditional models like **logistic regression** and **decision trees** are commonly used for loan approval predictions. However, these models can overfit the data and struggle to generalize, especially with complex datasets.

Disadvantages:

- Prone to overfitting with complex data.
- Limited ability to capture non-linear relationships.
- Less adaptable to new or unseen data.

B. Proposed System:

We propose using a Bayesian Neural Network (BNN) combined with Monte Carlo Dropout to prevent overfitting. The BNN captures complex relationships, while MC Dropout ensures better generalization and robustness in loan approval predictions.

Advantages:

- Better generalization and prevents overfitting.
- Captures complex relationships in data.
- Increased robustness to new data.

C. Modules: Implementation

- **Data Preprocessing:** Using readr, dplyr, and tidyr for cleaning and preparing the data.
- **Model Building:** Creating and training the BNN with Monte Carlo Dropout using keras.
- **Visualization:** Visualizing training/validation metrics with ggplot2.

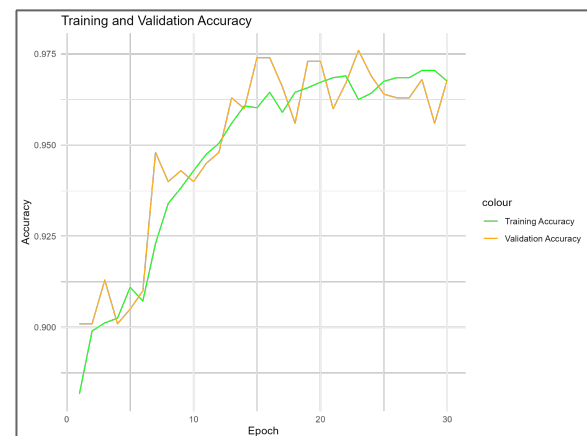
3. METHODS & SYSTEM DESIGN

- **Preprocessing:** Data cleaned and normalized.
- **Model:** 4 hidden layers with 128, 64, 32, and 16 units, respectively. Output with sigmoid activation.
- **Training:** 80% training, 20% testing. Epochs = 30, Batch Size = 5, Loss Function = Binary Cross-Entropy.
- **Optimization:** Adam Optimizer.

3.1 Data Analysis & Results for Q1

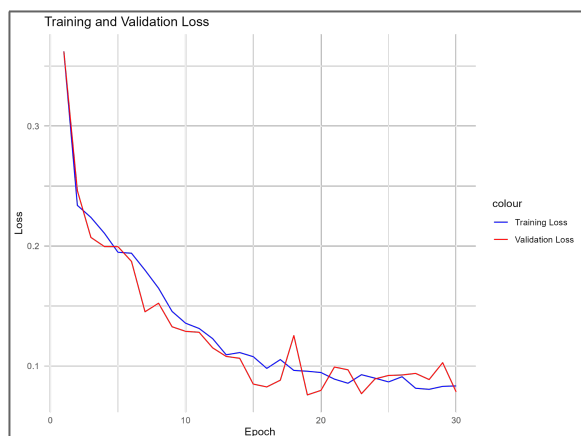
The neural network randomly initializes weights, so the results vary with each iteration. After multiple tests, the validation loss was around 10%, with the final iteration at 7.83%. This indicates overfitting, as the model closely memorizes the data. The Training and Validation Accuracy graphs showed over 95% accuracy, suggesting the model generalizes too well to new data, confirming overfitting.

```
32/32 [=====] - 0s 1ms/step - loss: 0.0783 -
accuracy: 0.9680
      loss accuracy
0.07829145 0.96799999
```



The following graph, Training and Validation Loss, shows the inverse performance of our model, where the goal is to minimize both losses. The more important metric is the Validation Loss, which is identical to the Validation Accuracy metric in the previous graph, and indicates how well the model generalizes to unseen data. The final, average loss for the Validation Loss was

7.83%, as reported earlier, and again, this indicates that the model is overfitting the data.



Additionally, after running the test data through the model, approximately 9.7% of loans were classified as "Approved."

```
prop_of_loan_approvals
0    1
903  97
```

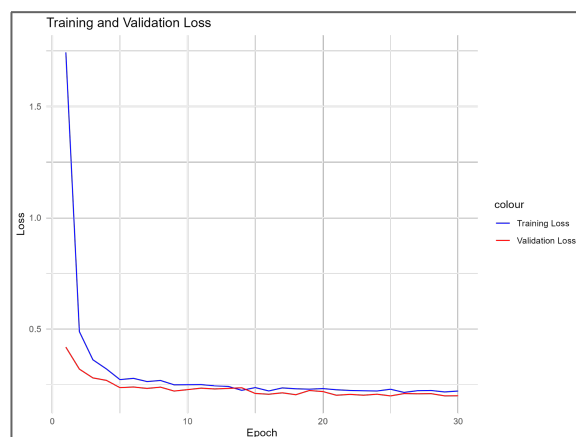
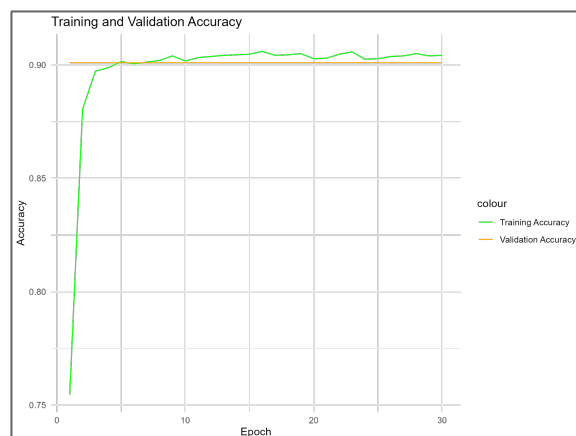
3.2 Analysis & Results of Question 2

We applied Monte Carlo Dropout (MC Dropout) to regulate the BNN and prevent overfitting. By adjusting dropout rates (30% for Layer 1, 40% for Layer 2, 50% for Layer 3, and 60% for Layer 4), the model's training loss increased to around 20%, while accuracy remained near 90%. This confirmed that MC Dropout effectively prevented overfitting.

```
32/32 [=====] - 0s 884us/step - loss: 0.1999 -
accuracy: 0.9010
loss accuracy
0.1999313 0.9010000
```

The prediction performance of the model with MC Dropout reduced the amount of loans being approved from 9.7% to 8.5% of loans, which seems to align more with real-world performance and is therefore a more reliable model.

```
prop_of_loan_approvals
0    1
915  85
```



4. CONCLUSION

We were able to successfully produce a BNN that accurately predicts classification labels for a multi-variable dataset and keep said model from overfitting the data by utilizing the MC Dropout technique. While our model was used specifically to determine if a loan applicant would be approved for a personal loan, it can also be used to determine if an applicant will be approved for any type of loan (car, mortgage, business etc.) or applied to any field that has a sufficiently-sampled, correlated dataset, and still provide a high degree of accuracy in predicting classification labels. Additionally, we have also proven that a BNN model is a good candidate to use with other standard, classification models: Naive Bayes, Gradient Boosting, Support Vector Machines, etc.

The code and supplemental analysis for our model is included in the attached R Markdown file, as well as on our rendered [Git-Page](#).

[GitHub Checkout Here](#)

4.1 Future Scope:

The Bayesian Neural Network (BNN) with MC Dropout can be expanded to other domains like fraud detection, customer churn prediction, or healthcare diagnostics, where large and complex datasets are involved. With improvements in computational power and techniques like distributed training, the model can be scaled to handle even larger datasets, making it applicable for real-time systems like loan approval or credit scoring.

Additionally, future research could explore combining the BNN with other machine learning models to further enhance accuracy. Techniques like batch normalization or regularization could also be applied to improve model generalization. Furthermore, enhancing model explainability will be crucial for applications where interpretability of decisions is required.

5. REFERENCES

- I. Bench, C. (2023). *Monte Carlo Dropout: A Practical Guide*. Retrieved from [Click here](#)
- II. Xu, W., Zhao, Z., & Zhao, T. (n.d.). *Tutorial: Dropout as Regularization and Bayesian Approximation*. Retrieved from [GitHub](#)
- III. Mae, Y., Kumagai, W., & Kanamori, T. (2021). Uncertainty propagation for dropout-based Bayesian neural networks. *Neural Networks*, 139, 279-290. [Click here](#)
- IV. Gal, Y., & Ghahramani, Z. (2016). Dropout as a Bayesian approximation: Representing model uncertainty in deep learning. *Proceedings of the 33rd International Conference on Machine Learning (ICML 2016)*, 48, 1050-1059. Retrieved from [Click here](#)
- V. Landbø, H. (2020). *Bayesian neural networks: A comparison of Monte Carlo variational inference and Monte Carlo dropout* (Master's thesis). Norwegian University of Science and Technology. Retrieved from [Click here](#)

6. About Authors:

Gubbala Durga Prasanth is a member of the team at *Dropout Dynamics*, School of Information, University of Arizona. His research interests include Machine Learning, Deep Learning, Statistical Modeling, and Artificial Intelligence.

Kendall Beaver is also part of *Dropout Dynamics*, School of Information, University of Arizona. His areas of expertise include Data Science, Machine Learning, and Big Data Analytics.

V.S. Murali Krishna Chittlu is a team member at *Dropout Dynamics*, School of Information, University of Arizona. His research interests focus on Machine Learning, Deep Learning, and Statistical Modeling.

Dr. Kunal Arekar is a faculty member at the *College of Information Science*, University of Arizona. His research focuses on Statistical and Bayesian Modeling, Deep Learning, and Computational Models. He can be reached via email at kunalarekar@arizona.edu, and his office is located at Room 424, Harvill Building, University of Arizona, Tucson - AZ.