University Leipzig

Faculty for Mathematics and Computer Science

Institute for Computer Science

Course Probabilistic Machine Learning

Seminar paper on the topic

**Stock Price Prediction**

Supervising lecturer: Dr. Alvaro Díaz Ruelas

Authors: Arne Michael Schulze (3722622)

Joshua Nerling (3772226)

Submitted on: 28.08.2025

# Motivation

The main goal of this project is to explore ways to predict the next closing price of Google stock using a lookback window of past data. Rather than aiming for long-term forecasts, our approach focuses on patterns in historical prices to estimate the most likely next price.

Stock price prediction remains a challenging task due to the inherent complexity and volatility of financial markets. Prices are shaped by a combination of stochastic behaviors, long-range dependencies, sudden jumps, and nonlinear influences from economic and external factors. To address these challenges, we investigate three distinct modeling strategies: a Bayesian Linear Regression model as a probabilistic and interpretable baseline, a Long Short-Term Memory network designed to capture nonlinear temporal dependencies, and a Bayesian variant built on the LSTM structure to incorporate uncertainty quantification. This comparative setup enables us to analyze the strengths and limitations of each approach while integrating both interpretability and advanced sequence modeling into the prediction task.

In addition, we compare the three models against each other, evaluate different lookback window lengths, and examine how the models perform in the presence of nonlinear events such as financial crises compared to periods where prices predominantly follow broader market trends.

It is important to note, however, that our approach is limited to predicting the next price based solely on historical patterns within the data. External influences such as macroeconomic indicators, unexpected market shocks, or rare black swan events are not incorporated into our models and therefore cannot be captured in our predictions.

Furthermore, this project also aims to highlight the inherent uncertainty in stock price forecasting and the limitations of relying solely on historical data.

# Dataset

"The foundation of any data-driven analysis is the availability of suitable and well-prepared data. This is particularly true in the financial domain, as stock prices represent highly dynamic time series. A clear presentation of the dataset used, as well as the processing steps applied, is therefore crucial for the traceability and validity of the results. The underlying data and its processing will be explained in the following chapters

## 2.1 Google Stock

For this project, we use historical stock data for Google obtained from Yahoo Finance. The dataset includes daily records from 2004 to 2025, containing the Open, High, Low, Close, and Adjusted Volume for each trading day. For simplicity and interpretability, our analysis focuses exclusively on the Close price, which serves as the primary input for our models. Our Figure shows a weekly candlestick chart of the last five years, providing a clear visualization of recent price trends. (Yahoo Finance, n.d.)

A graph of a stock market

AI-generated content may be incorrect.

Figure 1: Googl Stock Price - 5Year

## 2.2 Data Processing

The data used from Yahoo Finance was already very consistent and complete at the outset like Figure 2 demonstrates. To ensure that the closing prices could be processed in the best possible way for our application, they were transformed by a MinMax scaler into a value range between 0 and 1. This ensures numerical stability during training and also prevents outliers from influencing the optimization of the models. The training and test data were then divided into an 80/20 split in terms of time. This means that 80% of the historical data is used for training purposes and 20% for evaluation. Finally, the data was stored in sequences of fixed length to integrate a temporal dependency. The last ten closing prices were used as a lookback for this purpose. This transformation forms the basis for both the linear regression models and the LSTM networks and ensures that the models can learn patterns from historical movements to predict the next value. To ensure that the data remains interpretable, it is transformed back to the original price unit after the prediction.

A screenshot of a computer

AI-generated content may be incorrect.

Figure 2: Dataset

# Methods

The following chapter explains the used models which this project focuses on. The first approach covers the analysis of LSTM networks whereas the second method is a bayesian regression model.

## 3.1 Long Short-Term Memory

Long Short-Term Memory (LSTM) networks are a type of recurrent neural network specifically designed to capture long-term dependencies in sequential data, making them well-suited for time series prediction such as stock prices.

In our preprocessing step, we use the raw closing prices without applying normalization. While scaling is often recommended to stabilize training, in our case the model performed significantly better without it. Removing the MinMaxScaler led to a clear improvement in predictive accuracy, as shown by a reduction in MSE from 31.19 to 10.25 and MAE from 4.24 to 2.39. This suggests that the LSTM model was able to capture the underlying patterns in the original data more effectively when trained directly on unscaled values.

The dataset is split into 80% training data and 20% testing data to allow the model to learn patterns from the majority of the data while keeping a separate portion for unbiased evaluation. Additionally, we use a lookback window of 10 days, meaning that the model considers the previous 10 closing prices to predict the next value. We tested different window sizes (5, 10, and 15 days) and found that the model performed best with 10 days, achieving an MSE of 10.25 and MAE of 2.39, compared to 13.87 / 2.816 for 5 days and 51.19 / 5.4 for 15 days. This result suggests that a 10-day window provides the right balance between capturing relevant short-term trends and avoiding excessive noise from longer histories.

We use a TimeseriesGenerator to transform the stock price data into sequences with a lookback of 10 days, allowing the model to predict the next value from the previous 10 observations. The LSTM model consists of a single LSTM layer with 10 units and a Dense output layer with one neuron. It is compiled with the Adam optimizer and Mean Squared Error loss and trained for 25 epochs. To improve efficiency, the code first checks if a pre-trained model exists and loads it; otherwise, it trains a new model and saves it for later use.

While LSTMs are effective for sequence modeling, stock market data is highly volatile and noisy, making it difficult for the model to capture consistent patterns. (Hochreiter et. al., 1997), (Sunny et. al., 2020)

## 3.2 Bayesian Regression

In contrast to traditional machine learning approaches, the main approach pursued within this project is based on a probabilistic approach, the Bayesian model. The disadvantage of the LSTM approach lies in the significance of the deterministic predictions. Since no information is provided about the uncertainties associated with the forecast, this significantly limits the value of the model within the financial environment, as this is a decisive factor. The Bayesian approach offers a solution to this problem. It allows predictions to be viewed as complete probability distributions.

The probabilistic linear regression model “Bayesian Ridge Regression” is used within the project. With the help of this model, the closing price of a stock on the following day can be predicted. The historical price data in conjunction with the aforementioned “look-back” period are relevant for this. The target variable is limited to the immediately following value.

This procedure corresponds to an autoregressive approach: the next value is predicted from a sliding sequence of observations. To stabilize the modeling, the price data are first scaled before training. To ensure interpretability nonetheless, the data are transformed back into their original price units after the prediction. „

The model used is based on classic linear regression, but extends it with a Bayesian regularization mechanism. This assumes that the regression weights are not fixed, but rather random variables with normal distributions around zero. The strength of this regularization—i.e., how strongly the weights are pulled toward zero—is automatically estimated from the data. This results in a posterior distribution over the weights, from which both expected values and uncertainties can be derived. By default, the model returns the posterior mean for predictions, but it would also be possible to output an estimate of the prediction uncertainty.

After the model has been initialized and trained, the predictions are compared with the actual closing prices. The MSE, RSME, MAE, and R2 score are used to determine the performance of the Bayesian ridge regression model. These metrics explain the average accuracy and deviations of the predictions and their share of explained variance. (Klauenberg et al., 2015), (Minka et al., 2000)

## 3.3 Bayesian Neural Network

As a further model for comparison, the BNN is used as a third pillar for predicting stocks. The Bayesian Neural Network is an extension of the classic neural network to include the probabilistic approach. The weights are modeled as probability distributions instead of fixed parameters and can thus calculate a prediction distribution instead of a point prediction. This allows both the target values to be estimated and the uncertainty to be calculated. As a result, uncertainties can be recorded more robustly, which is particularly important in the field of finance.

The implemented BNN combines an LSTM encoder with a Bayesian approach. The LSTM layer is stabilized by a dense layer, followed by distribution through the Bayesian dense flipout layer. To verify the correct quantification of uncertainty, the loss function is used as NLL (negative log likelihood). Unlike deterministic networks, this structure allows explicit calibration of the prediction intervals and thus provides valuable metrics for risk assessment.

In this project, a BNN with Bayes-by-Backprop was implemented. This approach is based on variational inference, in which the unknown posterior distribution of the model parameters is approximated by an approximate distribution. Optimization is achieved by minimizing the Kullback-Leibler divergence between the approximation and the posterior. In practice, this is implemented in TensorFlow Probability using special Bayesian layers, such as DenseFlipout, which draw stochastic samples of the weights during each forward pass. This integrates the uncertainties directly into the prediction. (Jospin et. al., 2022)

# Results

The performance of the individual models will be examined in detail in the following chapter.

## 4.1 Connection with real events

In this study, our three models: LSTM (Long Short-Term Memory network), Bayesian Regression, and BNN (Bayesian Neural Network), were evaluated across five distinct periods that reflect different market conditions: the Global Financial Crisis from 2008-09-01 to 2009-06-30, the COVID-19 Crisis from 20202-02-01 to 2020-12-31, the Russia–Ukraine War from 2022-02-24 to 2022-06-30, a Stable Uptrend from 2017-01-01 to 2017-12-31 and the Last Four Years 2021-01 to 2025-01) representing the most recent developments.

For model assessment, three key performance metrics were applied. The Root Mean Squared Error (RMSE) measures the absolute prediction error in the same unit as the stock price, providing insight into the scale of deviations between predicted and actual values. The Mean Absolute Percentage Error (MAPE), in contrast, expresses the prediction error relative to the actual price, which allows results to be compared across different price levels and market phases. Finally, the Coefficient of Determination (R²) indicates how well the model explains the variance in the data, with values closer to 1 reflecting a stronger explanatory power and overall model fit

Table 1: Root Mean Squared Error

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| RMSE in USD | Financial Crises  2008/2009 | COVID-19 Crises 2020 | Rus-Ukr. War 2022 | Uptrend 2017 | Last 4 Years |
| LSTM | 0,11 | 3,63 | 8,32 | 0,24 | 6,55 |
| Bayesian | 0,33 | 1,93 | 2,98 | 0,21 | 2,46 |
| BNN | 2,5 | 2,8 | 4,73 | 0,90 | 5,014 |

Table 2: Mean Absolute Percentage Error

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| MAPE | Financial Crises  2008/2009 | COVID-19 Crises 2020 | Rus-Ukr. War 2022 | Uptrend 2017 | Last 4 Years |
| LSTM | 3,16% | 2,14% | 2% | 0,78% | 1,48% |
| Bayesian | 2,99% | 2,22% | 2,06% | 0,7% | 1,42% |
| BNN | 3,5% | 3,11% | 3,31% | 1,54% | 3,2% |

Table 3: R² coefficient

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| R² | Financial Crises  2008/2009 | COVID-19 Crises 2020 | Rus-Ukr. War 2022 | Uptrend 2017 | Last 4 Years |
| LSTM | 0,8995 | 0,9055 | 0,9055 | 0,9815 | 0,9902 |
| Bayesian | 0,99001 | 0,9024 | 0,9284 | 0,99843 | 0,9909 |
| BNN | 0,9 | 0,904 | 0,82 | 0,941 | 0,9622 |

During periods of crisis, such as the financial crisis of 2008/2009, the COVID-19 pandemic in 2020, and the Russia–Ukraine war in 2022, Bayesian Regression consistently demonstrates the most robust and stable performance. While LSTM achieves slightly lower RMSE values in some instances, Bayesian Regression provides a superior overall balance between accuracy and reliability, as reflected in its consistently lower MAPE and higher R² values. This indicates that the Bayesian model not only forecasts with precision but also explains market variance more effectively. BNN, by contrast, performs significantly weaker under crisis conditions, with high RMSE values and declining R², particularly evident during the Russia–Ukraine war.

In the case of the financial crisis and the COVID-19 pandemic, all models were able to adapt relatively quickly to the changing market conditions and returned to delivering accurate forecasts after a short adjustment period (Figures 3 and 4). During the Russia–Ukraine war, however, the models faced persistent difficulties in generating reliable predictions over the entire crisis window (Figure 6). During the stable uptrend in 2017, the Bayesian Regression model often predicted values that were too low, showing a cautious bias in strong bullish markets (Figure 7).

In phases of relative market stability, such as the 2017 uptrend and the aggregated results of the last four years, Bayesian Regression again proves to be the most reliable approach. Although LSTM shows competitive RMSE results and benefits from its ability to capture clear trends, the Bayesian model remains superior in terms of predictive consistency. Across both short-term trends and longer time horizons, it achieves excellent MAPE values and maintains nearly perfect R² scores, thereby offering stronger robustness against fluctuations in data quality or changing dynamics. BNN once again falls behind, with higher error levels and weaker explanatory power compared to both Bayesian Regression and LSTM.

Looking at the overall comparison, Bayesian Regression emerges as the most effective and reliable method across all scenarios. It provides the best trade-off between error minimization and model stability, outperforming LSTM in terms of consistency and robustness, while clearly surpassing BNN in every respect. LSTM remains a strong competitor, particularly in environments dominated by clear trends, but it lacks the stability and resilience that Bayesian Regression demonstrates across both crises and stable markets.

## 4.2 Final comparison of the models

The Bayesian regression model is transparent and computationally efficient, as no complex hyperparameter tests are required. At the same time, uncertainties in the parameters and predictions are systematically taken into account, which is a significant advantage over purely deterministic linear methods. This makes Bayesian ridge regression particularly well suited as a probabilistic base model and provides a robust basis for comparison with more complex methods such as recurrent neural networks.

However overall, the BNN provides higher point errors but significantly more meaningful uncertainty information than deterministic models, making it a robust tool for forecasts with risk assessment. In contrast, LSTM models often suffer from strong overfitting tendencies, especially in financial time series with limited training data, and they do not provide any intrinsic uncertainty estimates. This lack of probabilistic information limits their applicability in risk-sensitive forecasting tasks, even though they can achieve high accuracy in certain trend-following scenarios.

# Discussion

Overall, the dataset proved to be clean and easy to preprocess, which facilitated the implementation of different models. Among the tested approaches, the Bayesian was able to generate reasonably accurate forecasts and adapt well to underlying data patterns. However, our analysis had some limitations: we only used daily candlestick data, without incorporating technical indicators or other market signals, and we did not consider the broader macroeconomic context in our evaluations.

Two open questions arise from our results and limitations. First, how can probabilistic forecasts be better evaluated beyond point-wise error metrics such as Mean Squared Error or the Mean Absolute Percentage Error? While these measures are intuitive and widely used, they do not adequately capture the probabilistic nature of forecasts. Alternative time series evaluation methods that account for predictive distributions and uncertainty could therefore provide a more comprehensive assessment, but a detailed investigation of such approaches lies beyond the scope of our study.

Second, we considered whether a hybrid approach could combine the strengths of LSTM and Bayesian methods. While this idea is conceptually feasible and even explored in existing research through Bayesian neural networks on top of LSTM architectures, our own attempt to implement such a hybrid model did not yield superior results. In practice, the Bayesian regression model alone still delivered the most consistent and reliable performance across the evaluated scenarios.

# Sources

Hochreiter, S. ,Long Short-Term memory. (1997, 15. November). MIT Press Journals & Magazine | IEEE Xplore. <https://ieeexplore.ieee.org/abstract/document/6795963>

Jospin, L. V., Laga, H., Boussaid, F., Buntine, W., & Bennamoun, M. (2022). Hands-on Bayesian neural networks—A tutorial for deep learning users. IEEE Computational Intelligence Magazine, 17(2), 29-48.

Klauenberg, K., Wübbeler, G., Mickan, B., Harris, P. & Elster, C. (2015). A tutorial on Bayesian Normal linear regression. Metrologia, 52(6), 878–892.

Minka, T. (2000). Bayesian linear regression (pp. 1-12). Technical report, MIT.

Sunny, A. I, Deep Learning-Based Stock Price Prediction Using LSTM and Bi-Directional LSTM Model. (2020, 24. Oktober). IEEE Conference Publication | IEEE Xplore. https://ieeexplore.ieee.org/abstract/document/9257950

Yahoo Finance. (n.d.). Alphabet Inc. (GOOG) stock historical prices & data. Yahoo. Retrieved August 28, 2025, from [https://finance.yahoo.com/quote/GOOG/history/](https://finance.yahoo.com/quote/GOOG/history/?utm_source=chatgpt.com)

# Appendix

A graph of a pulse

AI-generated content may be incorrect.

Figure 3: Residuals - Corona Crises

A graph showing a number of data

AI-generated content may be incorrect.

Figure 4: Financial Crises

A graph of orange and blue lines

AI-generated content may be incorrect.

Figure 5: 80/20 Testsplit

A graph with orange lines

AI-generated content may be incorrect.

Figure 6: Russian-Ukrainine War

A graph showing a line of orange lines

AI-generated content may be incorrect.

Figure 7: Strong Uptrend