(Un)certain asset price direction prediction using Bayesian Neural Networks

Robin Eriksson

September 2020

Project summary

This project will employ a Bayesian neural network classifier, using price features, i.e., technical indicators, as inputs to produce probability density estimates of a binary target, i.e., positive or negative returns.

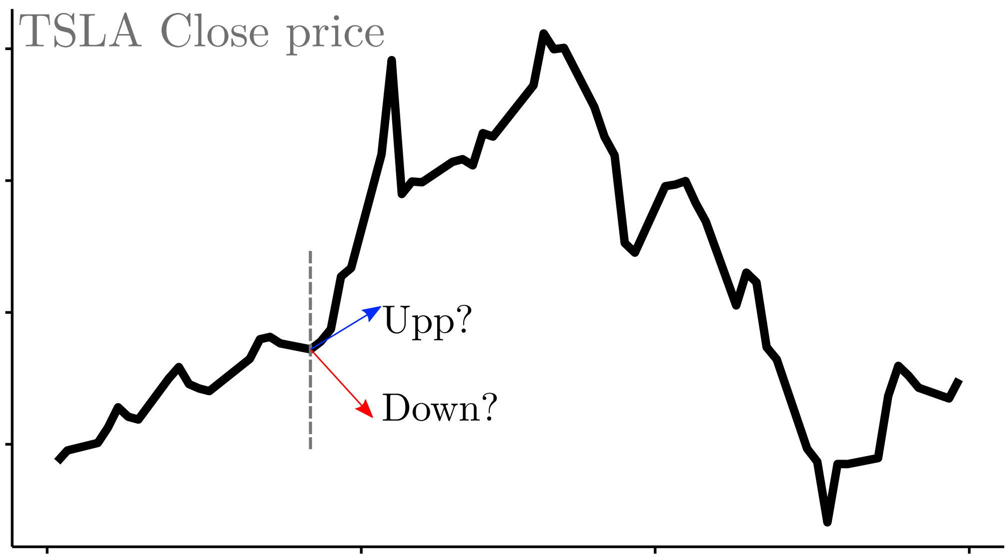


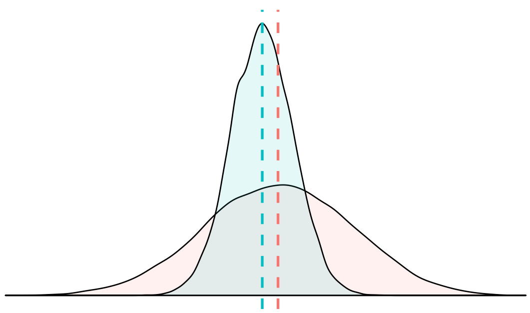
Figure 1: The problem of estimating the direction of the stock price given price features is well studied, however, in recent years, modern machine learning techniques have been proving themselves to be a potential solution. We aim to highlight the uncertainties is such estimates using methods that emphasize prediction densities.

Background

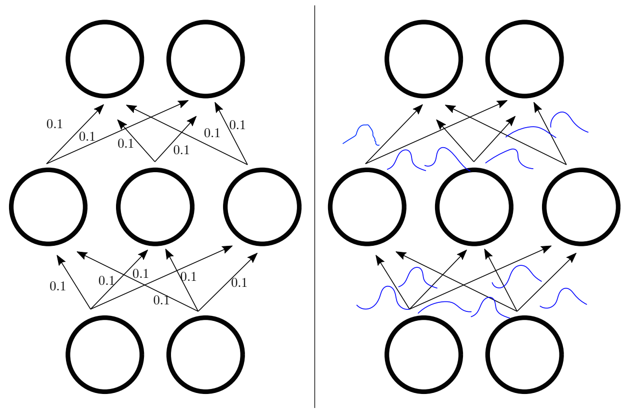
Estimating the direction of tomorrow's price, up or down, of a stock price can be seen as the flipping of a coin, i.e., it can be seen as a stochastic process. The coin might, however, be biased and we can try to estimate this bias. Price direction estimation with machine learning techniques has been tried using decision trees techniques [1], neural networks [4], gradient boosting [2].

What is missing from these methods is an estimate of the uncertainty in the predictions. To illustrate this problem, consider the potential pitfall of deciding to place a bet on $P(\text{up}) = 0.7$ without being told the error bars of this estimate. With them included it might be $P(\text{up}) = 0.7 \pm 0.3$, and we are therefore no longer certain about it being up any more. If forced to place a bet on the means of $P(\text{up}) = 0.8 \pm 0.35$ and $P(\text{up}) = 0.55 \pm 0.01$ one would choose the former, however, if the errors bars are included the opposite is true. To visualize the problem, see Figure 2a.

To estimate the error bars of the estimate, one can directly try to estimate the prediction density by Bayesian neural networks [3], see Figure 2b for the conceptional difference between the Bayesian and static neural network.



(a) Illustration of the problem where one does not include probability densities with computed estimates. Shaded red and blue areas are two densities, and the dashed lines are their respective mean. Red has a higher mean but a larger standard deviation than the blue one.



(b) Sketch of the conceptional difference between a (left) neural network and a (right) Bayesian neural network. In the regular network, the weights are fixed values, and in the Bayesian network, each weight is represented by a distribution of values. The addition of distribution of weights allows for uncertainty to propagate naturally through the network, while for the normal, static, network other workarounds have to be given.

Relevant courses

The following courses are relevant, but not required for the project: *Computational Finance: Pricing and Valuation, Statistical Machine Learning, Advanced Probabilistic Machine Learning*

References

[1] Suryoday Basak, Saibal Kar, Snehanshu Saha, Luckyson Khaidem, and Sudeepa Roy Dey. Predicting the direction of stock market prices using tree-based classifiers. *The North American Journal of Economics and Finance*, 47:552–567, 2019.

[2] Shubharthi Dey, Yash Kumar, Snehanshu Saha, and Suryoday Basak. Forecasting to classification: Predicting the direction of stock market price using xtreme gradient boosting. *PESIT, Bengaluru, India, Working Paper*, 2016.

[3] Laurent Valentin Jospin, Wray Buntine, Farid Boussaid, Hamid Laga, and Mohammed Bennamoun. Hands-on bayesian neural networks–a tutorial for deep learning users. *arXiv preprint arXiv:2007.06823*, 2020.

[4] Mehar Vijh, Deeksha Chandola, Vinay Anand Tikkiwal, and Arun Kumar. Stock closing price prediction using machine learning techniques. *Procedia Computer Science*, 167:599–606, 2020.