Bitcoins Option Price Forecasting using Multilayer Neural Network

Shatha Qamhieh Hashem[a] & Haya Sammaneh[b]
Emad Natsheh [b] & Manar Qamhieh[b]

Finance Department[a]
Computer Engineering Department[b]
An-Najah National University, Palestine

19 May, 2020

Table of Contents

Motivation

- Options are used in financial markets to control investment risks. An accurate estimation of the current or future option price is of great importance for investors in making effective investment decisions.

Motivation

- Options are used in financial markets to control investment risks. An accurate estimation of the current or future option price is of great importance for investors in making effective investment decisions.
- Cryptocurrency markets emerged after the successful release of Bitcoins in 2009 by Satoshi Nakamoto, and the Bitcoin option markets increased in importance since then despite its volatile trading history.

Motivation

- Options are used in financial markets to control investment risks. An accurate estimation of the current or future option price is of great importance for investors in making effective investment decisions.
- Cryptocurrency markets emerged after the successful release of Bitcoins in 2009 by Satoshi Nakamoto, and the Bitcoin option markets increased in importance since then despite its volatile trading history.
- Option pricing models used to obtain price forecasts include:

Motivation

- Options are used in financial markets to control investment risks. An accurate estimation of the current or future option price is of great importance for investors in making effective investment decisions.
- Cryptocurrency markets emerged after the successful release of Bitcoins in 2009 by Satoshi Nakamoto, and the Bitcoin option markets increased in importance since then despite its volatile trading history.
- Option pricing models used to obtain price forecasts include:
  - classical parametric models (e.g. Black-Scholes model, tree models, the finite difference method and the Monte Carlo simulation). Models are subject to stringent assumptions and non-objective parameter selection.

Motivation

- Options are used in financial markets to control investment risks. An accurate estimation of the current or future option price is of great importance for investors in making effective investment decisions.
- Cryptocurrency markets emerged after the successful release of Bitcoins in 2009 by Satoshi Nakamoto, and the Bitcoin option markets increased in importance since then despite its volatile trading history.
- Option pricing models used to obtain price forecasts include:
  - classical parametric models (e.g. Black-Scholes model, tree models, the finite difference method and the Monte Carlo simulation). Models are subject to stringent assumptions and non-objective parameter selection.
  - non-parametric models (e.g. multilayer neural network). Provide improved predictive performance without the stringent assumptions for parameter estimation.

Motivation

- Options are used in financial markets to control investment risks. An accurate estimation of the current or future option price is of great importance for investors in making effective investment decisions.
- Cryptocurrency markets emerged after the successful release of Bitcoins in 2009 by Satoshi Nakamoto, and the Bitcoin option markets increased in importance since then despite its volatile trading history.
- Option pricing models used to obtain price forecasts include:
  - classical parametric models (e.g. Black-Scholes model, tree models, the finite difference method and the Monte Carlo simulation). Models are subject to stringent assumptions and non-objective parameter selection.
  - non-parametric models (e.g. multilayer neural network). Provide improved predictive performance without the stringent assumptions for parameter estimation.
- In this work, we are interested in improving the accuracy of Bitcoin option price forecasts to support the investors investment choices in the volatile Bitcoin markets.

Table of Contents

- Neural networks are found to outperform classical models in terms of option price prediction within volatile markets (e.g. Yao et al. (2000), Amilon (2003), Binner et al. (2005), and Lin and Yeh (2005))

Literature: Neural Networks for Bitcoin Option Pricing

- Neural networks are found to outperform classical models in terms of option price prediction within volatile markets (e.g. Yao et al. (2000), Amilon (2003), Binner et al. (2005), and Lin and Yeh (2005))
- The two stage methodology introduced by Liang et al. (2009), was applied by Pagnottoni (2019) on cryptocurrency option pricing, and specifically on Bitcoins:

Literature: Neural Networks for Bitcoin Option Pricing

- Neural networks are found to outperform classical models in terms of option price prediction within volatile markets (e.g. Yao et al. (2000), Amilon (2003), Binner et al. (2005), and Lin and Yeh (2005))
- The two stage methodology introduced by Liang et al. (2009), was applied by Pagnottoni (2019) on cryptocurrency option pricing, and specifically on Bitcoins:
  - In the first stage, classical option pricing models are used to produce the prices forecast.

Literature: Neural Networks for Bitcoin Option Pricing

- Neural networks are found to outperform classical models in terms of option price prediction within volatile markets (e.g. Yao et al. (2000), Amilon (2003), Binner et al. (2005), and Lin and Yeh (2005))
- The two stage methodology introduced by Liang et al. (2009), was applied by Pagnottoni (2019) on cryptocurrency option pricing, and specifically on Bitcoins:
  - In the first stage, classical option pricing models are used to produce the prices forecast.
  - In the second stage, forecasts are used as inputs for the neural network.

Literature: Neural Networks for Bitcoin Option Pricing

- Neural networks are found to outperform classical models in terms of option price prediction within volatile markets (e.g. Yao et al. (2000), Amilon (2003), Binner et al. (2005), and Lin and Yeh (2005))
- The two stage methodology introduced by Liang et al. (2009), was applied by Pagnottoni (2019) on cryptocurrency option pricing, and specifically on Bitcoins:
  - In the first stage, classical option pricing models are used to produce the prices forecast.
  - In the second stage, forecasts are used as inputs for the neural network.
- The two steps classical-neural modelling for option pricing is found to improve price forecasts through reducing forecast errors.

Research Objectives

- In this work, we follow liang et al. (2009) and Pagnottoni (2019) in using the two step classical-neural modelling, and we improve the accuracy of Bitcoin option price forecasts by augmenting the classical models with a multilayer neural network.

Research Objectives

- In this work, we follow liang et al. (2009) and Pagnottoni (2019) in using the two step classical-neural modelling, and we improve the accuracy of Bitcoin option price forecasts by augmenting the classical models with a multilayer neural network.
- Our main aim is to utilize the multilayer neural networks to further reduce the forecasting errors, and thus achieving a more precise current and future Bitcoin option pricing forecast.

Research Objectives

- In this work, we follow liang et al. (2009) and Pagnottoni (2019) in using the two step classical-neural modelling, and we improve the accuracy of Bitcoin option price forecasts by augmenting the classical models with a multilayer neural network.
- Our main aim is to utilize the multilayer neural networks to further reduce the forecasting errors, and thus achieving a more precise current and future Bitcoin option pricing forecast.
- Nevertheless, our focus is not on the classical models performance themselves, nor on proving that neural networks are better in terms of option pricing prediction. Instead, our focus is on improving the prediction produced by the neural network through the improvement of its: Input features, structure, activation functions and training algorithm.

Parametric models for option pricing

- We use the classical models to price the European style, where $C_{t,i}$ denotes the option price, r is the risk free rate,$K_{t,i}$ is the option exercise price, $S_{t,i}$ is the underlying asset price, σ is the asset price volatility, T is the time to maturity, $\Delta t = T/n$ is the time steps in which n is the total number of time steps in a binomial tree

Parametric models for option pricing

- We use the classical models to price the European style, where $C_{t,i}$ denotes the option price, r is the risk free rate, $K_{t,i}$ is the option exercise price, $S_{t,i}$ is the underlying asset price, σ is the asset price volatility, T is the time to maturity, $\Delta t = T/n$ is the time steps in which n is the total number of time steps in a binomial tree
- Binomial tree: The upward rate $u = e^{\sigma\sqrt{\Delta t}}$, and the downward rate $d = 1/u$, the probability of upward movement, $p_d = \frac{e^{r\Delta t}-d}{u-d}$, the probability of downward movement $p_d = 1 - p$,and the option price is $C_{t-\Delta t,i} = e^{-r\Delta t}(p_u C_{t,i} + p_d C_{t,i+1})$

Parametric models for option pricing

- We use the classical models to price the European style, where $C_{t,i}$ denotes the option price, r is the risk free rate,$K_{t,i}$ is the option exercise price, $S_{t,i}$ is the underlying asset price, σ is the asset price volatility, T is the time to maturity, $\Delta t = T/n$ is the time steps in which n is the total number of time steps in a binomial tree
- Binomial tree: The upward rate $u = e^{\sigma\sqrt{\Delta t}}$, and the downward rate $d = 1/u$, the probability of upward movement, $p_d = \frac{e^{r\Delta t} - d}{u - d}$, the probability of downward movement $p_d = 1 - p$,and the option price is $C_{t-\Delta t,i} = e^{-r\Delta t}(p_u C_{t,i} + p_d C_{t,i+1})$
- Trinomial tree: The upward rate $u = e^{\sigma\sqrt{2\Delta t}}$, the downward rate $d = 1/u$, the middle rate $m = 1$ and the transition probabilities are:

Parametric models for option pricing

- We use the classical models to price the European style, where $C_{t,i}$ denotes the option price, r is the risk free rate,$K_{t,i}$ is the option exercise price, $S_{t,i}$ is the underlying asset price, σ is the asset price volatility, T is the time to maturity, $\Delta t = T/n$ is the time steps in which n is the total number of time steps in a binomial tree
- Binomial tree: The upward rate $u = e^{\sigma\sqrt{\Delta t}}$, and the downward rate $d = 1/u$, the probability of upward movement, $p_d = \frac{e^{r\Delta t} - d}{u-d}$, the probability of downward movement $p_d = 1 - p$,and the option price is $C_{t-\Delta t,i} = e^{-r\Delta t}(p_u C_{t,i} + p_d C_{t,i+1})$
- Trinomial tree: The upward rate $u = e^{\sigma\sqrt{2\Delta t}}$, the downward rate $d = 1/u$, the middle rate $m = 1$ and the transition probabilities are:
  - probability of upward movement $p_u = \left(\frac{e^{r\Delta t/2} - e^{-\sigma\sqrt{\Delta t/2}}}{e^{\sigma\sqrt{\Delta t/2}} - e^{-\sqrt{\Delta t/2}}}\right)^2$

Parametric models for option pricing

- We use the classical models to price the European style, where $C_{t,i}$ denotes the option price, r is the risk free rate,$K_{t,i}$ is the option exercise price, $S_{t,i}$ is the underlying asset price, σ is the asset price volatility, T is the time to maturity, $\Delta t = T/n$ is the time steps in which n is the total number of time steps in a binomial tree
- Binomial tree: The upward rate $u = e^{\sigma\sqrt{\Delta t}}$, and the downward rate $d = 1/u$, the probability of upward movement, $p_d = \frac{e^{r\Delta t} - d}{u - d}$, the probability of downward movement $p_d = 1 - p$,and the option price is $C_{t-\Delta t,i} = e^{-r\Delta t}(p_u C_{t,i} + p_d C_{t,i+1})$
- Trinomial tree: The upward rate $u = e^{\sigma\sqrt{2\Delta t}}$, the downward rate $d = 1/u$, the middle rate $m = 1$ and the transition probabilities are:
  - probability of upward movement $p_u = \left(\frac{e^{r\Delta t/2} - e^{-\sigma\sqrt{\Delta t/2}}}{e^{\sigma\sqrt{\Delta t/2}} - e^{-\sqrt{\Delta t/2}}} \right)^2$
  - probability of downward movement $p_d = \left(\frac{e^{r\Delta t/2} - e^{-r\sqrt{\Delta t/2}}}{e^{\sigma\sqrt{\Delta t/2}} - e^{-\sqrt{\Delta t/2}}} \right)^2$

Parametric models for option pricing

- We use the classical models to price the European style, where $C_{t,i}$ denotes the option price, r is the risk free rate,$K_{t,i}$ is the option exercise price, $S_{t,i}$ is the underlying asset price, σ is the asset price volatility, T is the time to maturity, $\Delta t = T/n$ is the time steps in which n is the total number of time steps in a binomial tree
- Binomial tree: The upward rate $u = e^{\sigma\sqrt{\Delta t}}$, and the downward rate $d = 1/u$, the probability of upward movement, $p_d = \frac{e^{r\Delta t} - d}{u - d}$, the probability of downward movement $p_d = 1 - p$,and the option price is $C_{t-\Delta t,i} = e^{-r\Delta t}(p_u C_{t,i} + p_d C_{t,i+1})$
- Trinomial tree: The upward rate $u = e^{\sigma\sqrt{2\Delta t}}$, the downward rate $d = 1/u$, the middle rate $m = 1$ and the transition probabilities are:
    - probability of upward movement $p_u = \left(\frac{e^{r\Delta t/2} - e^{-\sigma\sqrt{\Delta t/2}}}{e^{\sigma\sqrt{\Delta t/2}} - e^{-\sqrt{\Delta t/2}}}\right)^2$
    - probability of downward movement $p_d = \left(\frac{e^{r\Delta t/2} - e^{-r\sqrt{\Delta t/2}}}{e^{\sigma\sqrt{\Delta t/2}} - e^{-\sqrt{\Delta t/2}}}\right)^2$
    - probability of middle movement $p_m = 1 - p_u - p_d$

Parametric models for option pricing

- Finite difference method: The Crank-Nicholson is used as it is a combination of the explicit method (fast but unstable) and the implicit method (stable, but requires high computation). The Crank stability advantages are stability and fast convergence of accuracy. The Crank-Nicholson method performance is good for a relatively small price-time data

  Black-Schol partial differential quation:

  $rC = \frac{\partial C}{\partial t} + \frac{1}{2}\sigma^2 S^2 \frac{\partial^2 C}{\partial S^2} + rS\frac{\partial C}{\partial S}$

  The option price is $C_{i,j} = \frac{1}{1+r\Delta t}(p_u C_{i+1,j+1} + p_m C_{i+1,j} + p_d C_{i+1,j-1})$

  where the upward probability is $p_u = S_j r \frac{\Delta t}{2\Delta S} + \frac{1}{2}S_j^2\sigma^2\frac{\Delta t}{\Delta S^2}$

  where the upward probability is $p_m = 1 - S_j^2\sigma^2\frac{\Delta t}{\Delta S^2}$

  and the downward probability is $p_d = \frac{S_j r\Delta t}{2\Delta S} + \frac{1}{2}S_j^2\sigma^2\frac{\Delta t}{\Delta S^2}$

Parametric models for option pricing

- Finite difference method: The Crank-Nicholson is used as it is a combination of the explicit method (fast but unstable) and the implicit method (stable, but requires high computation). The Crank stability advantages are stability and fast convergence of accuracy. The Crank-Nicholson method performance is good for a relatively small price-time data

  Black-Schol partial differential quation:

  $rC = \frac{\partial C}{\partial t} + \frac{1}{2}\sigma^2 S^2 \frac{\partial^2 C}{\partial S^2} + rS\frac{\partial C}{\partial S}$

  The option price is $C_{i,j} = \frac{1}{1+r\Delta t}(p_u C_{i+1,j+1} + p_m C_{i+1,j} + p_d C_{i+1,j-1})$

  where the upward probability is $p_u = S_j r \frac{\Delta t}{2\Delta S} + \frac{1}{2} S_j^2 \sigma^2 \frac{\Delta t}{\Delta S^2}$

  where the upward probability is $p_m = 1 - S_j^2 \sigma^2 \frac{\Delta t}{\Delta S^2}$

  and the downward probability is $p_d = \frac{S_j r \Delta t}{2\Delta S} + \frac{1}{2} S_j^2 \sigma^2 \frac{\Delta t}{\Delta S^2}$

- Monte Carlo Simulation: the underlying asset price is assumed to follow a log-normal distribution L, and both of the call option price C_t, and the put option price P_t are averages of L simulated scenarios

  where the call option is $C_t = \frac{1}{L}\sum_{i=l}^{L} C_l$

  and the put option is $P_t = \frac{1}{L}\sum_{i=l}^{L} P_l$

Non Parametric models for option pricing

- Neural Networks: The nonlinear neural network is applied to the option price forecast derived from the classical models as a second stage to improve the prediction of option prices by reducing the error terms that are non linear in nature. We use a feed forward neural network, and we focus our improvement on:

Non Parametric models for option pricing

- Neural Networks: The nonlinear neural network is applied to the option price forecast derived from the classical models as a second stage to improve the prediction of option prices by reducing the error terms that are non linear in nature. We use a feed forward neural network, and we focus our improvement on:
  - The input features: we include the Binomial tree (Cox et al., 1979), the trimonial tree, the monte carlo and the finite difference method (Crank-Nicolson).

Non Parametric models for option pricing

- Neural Networks: The nonlinear neural network is applied to the option price forecast derived from the classical models as a second stage to improve the prediction of option prices by reducing the error terms that are non linear in nature. We use a feed forward neural network, and we focus our improvement on:
  - The input features: we include the Binomial tree (Cox et al., 1979), the trimonial tree, the monte carlo and the finite difference method (Crank-Nicolson).
  - The network structure: we use a multilayer neural network, we work with the number of network layers and neurons

Non Parametric models for option pricing

- Neural Networks: The nonlinear neural network is applied to the option price forecast derived from the classical models as a second stage to improve the prediction of option prices by reducing the error terms that are non linear in nature. We use a feed forward neural network, and we focus our improvement on:
  - The input features: we include the Binomial tree (Cox et al., 1979), the trimonial tree, the monte carlo and the finite difference method (Crank-Nicolson).
  - The network structure: we use a multilayer neural network, we work with the number of network layers and neurons
  - The hidden layer activation function: we use Tanh, ReLU, and Sigmod

Non Parametric models for option pricing

- Neural Networks: The nonlinear neural network is applied to the option price forecast derived from the classical models as a second stage to improve the prediction of option prices by reducing the error terms that are non linear in nature. We use a feed forward neural network, and we focus our improvement on:
  - The input features: we include the Binomial tree (Cox et al., 1979), the trimonial tree, the monte carlo and the finite difference method (Crank-Nicolson).
  - The network structure: we use a multilayer neural network, we work with the number of network layers and neurons
  - The hidden layer activation function: we use Tanh, ReLU, and Sigmod
  - The training algorithm: we use gradiant decent, scaled conjugate gradient, levenberg marquardt

Performance assessment

- we use the main three assessment measures, in which A denotes the actual option price, F denotes the fitted value from the specified option pricing model, including:

the mean absolute error $MAE = \frac{1}{N}\sum_{n=l}^{N} |A_{t,n} - F_{t,n}|$

the mean absolute percentage error $MAPE = \frac{1}{N}\sum_{n=l}^{N} |\frac{A_{t,n} - F_{t,n}}{A_{t,n}}|$

the mean squared error $MSE = \frac{1}{N}\sum_{n=l}^{N} (A_{t,n} - F_{t,n})^2$

Table of Contents

Data Collection

- Data are collected from 1 April 2020 to 1 May 2020 on a daily basis, every day at the same time (01:00 EEST).

Data Collection

- Data are collected from 1 April 2020 to 1 May 2020 on a daily basis, every day at the same time (01:00 EEST).
- Data include option prices for both European calls and puts.

Data Collection

- Data are collected from 1 April 2020 to 1 May 2020 on a daily basis, every day at the same time (01:00 EEST).
- Data include option prices for both European calls and puts.
- To avoid price fluctuations, the analysis is limited to the options having time to maturity ranging from 5 to 20 days, as well as to in-the-money options.

DataSet and Analysis

- The dataset contains 550 call and 600 put prices.

DataSet and Analysis

- The dataset contains 550 call and 600 put prices.
- Analysis Organization:

DataSet and Analysis

- The dataset contains 550 call and 600 put prices.
- Analysis Organization:
  - the first 4 weeks [1 April to 1 May] will be used for training the multilayers neural network model,

DataSet and Analysis

- The dataset contains 550 call and 600 put prices.
- Analysis Organization:
  - the first 4 weeks [1 April to 1 May] will be used for training the multilayers neural network model,
  - data collected from 1 May to 7 May are used for validating the performance of model.

DataSet and Analysis

- The dataset contains 550 call and 600 put prices.
- Analysis Organization:
  - the first 4 weeks [1 April to 1 May] will be used for training the multilayers neural network model,
  - data collected from 1 May to 7 May are used for validating the performance of model.
- Parameter specifications that are used in analytical models:

DataSet and Analysis

- The dataset contains 550 call and 600 put prices.
- Analysis Organization:
  - the first 4 weeks [1 April to 1 May] will be used for training the multilayers neural network model,
  - data collected from 1 May to 7 May are used for validating the performance of model.
- Parameter specifications that are used in analytical models:
  - Daily-based historical volatility for the deribit BTC index.

DataSet and Analysis

- The dataset contains 550 call and 600 put prices.
- Analysis Organization:
  - the first 4 weeks [1 April to 1 May] will be used for training the multilayers neural network model,
  - data collected from 1 May to 7 May are used for validating the performance of model.
- Parameter specifications that are used in analytical models:
  - Daily-based historical volatility for the deribit BTC index.
  - 2-months Libor interest rate as risk-free rate.

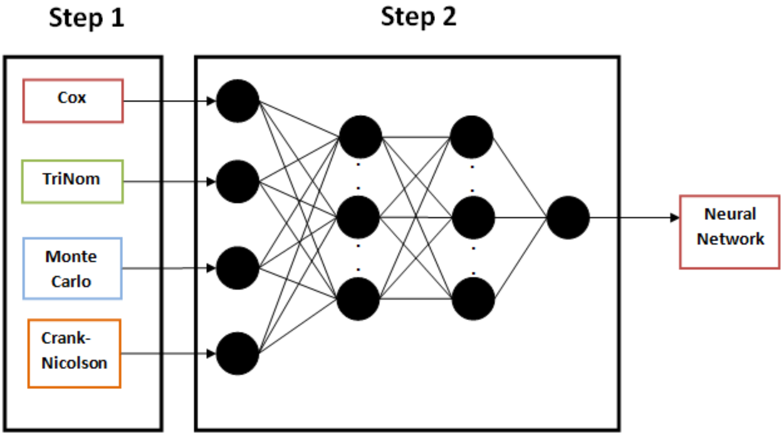
DataSet and Analysis

- The dataset contains 550 call and 600 put prices.
- Analysis Organization:
  - the first 4 weeks [1 April to 1 May] will be used for training the multilayers neural network model,
  - data collected from 1 May to 7 May are used for validating the performance of model.
- Parameter specifications that are used in analytical models:
  - Daily-based historical volatility for the deribit BTC index.
  - 2-months Libor interest rate as risk-free rate.
  - The finite difference method has a grid of maximum price Smax=10000 and grid size = 5 T.

DataSet and Analysis

- The dataset contains 550 call and 600 put prices.
- Analysis Organization:
  - the first 4 weeks [1 April to 1 May] will be used for training the multilayers neural network model,
  - data collected from 1 May to 7 May are used for validating the performance of model.
- Parameter specifications that are used in analytical models:
  - Daily-based historical volatility for the deribit BTC index.
  - 2-months Libor interest rate as risk-free rate.
  - The finite difference method has a grid of maximum price Smax=10000 and grid size = 5 T.
  - The Monte Carlo simulation involves 100 trials.

This work defines a two-step procedure in order to consistently evaluate option prices.



Call Options

Deep Neural Networks

1 Hidden Layer			2 Hidden Layer			3 Hidden Layer		
# of Neurons	MSE Train	MSE Test	# of Neurons	MSE Train	MSE Test	# of Neurons	MSE Train	MSE Test
10	545.59	533.88	15X15	756.62	1231.44	15X15X15	632.06	851.98
30	351.63	534.08	30X15	803.03	1022.1	30X15X15	848.27	912.63
50	540.33	638.67	30X30	1049.12	1487.11	30X30X30	677.48	891.66
70	606.48	651.18	[redacted]					
90	476.18	876.74						

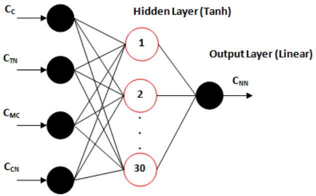
Neural Network Structure

Put Options

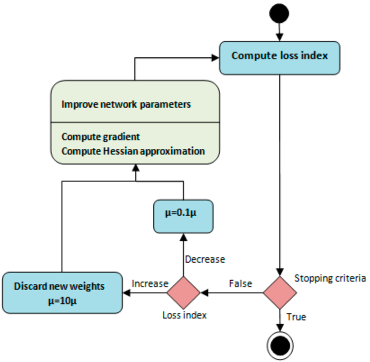
Deep Neural Networks

1 Hidden Layer			2 Hidden Layer			3 Hidden Layer		
# of Neurons	MSE Train	MSE Test	# of Neurons	MSE Train	MSE Test	# of Neurons	MSE Train	MSE Test
10	34297	90784	15X15	4996.84	5321.3	15X15X15	2729.4	4215.2
30	986.12	1369.58	30X15	2132.13	2331.2	30X15X15	3949.7	4012.5
50	1344.1	12003.1	30X30	7753.67	9762.27	30X30X30	2562.2	3321.8
70	1145.7	1707.58	[redacted]					
90	1245.5	2323.48						

Neural Network Structure



(a) Neural Network Structure

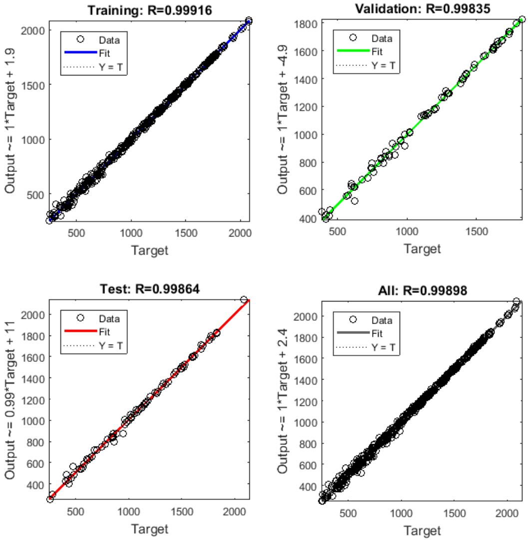


(b) Training algorithm (Levenberg Marquardt)

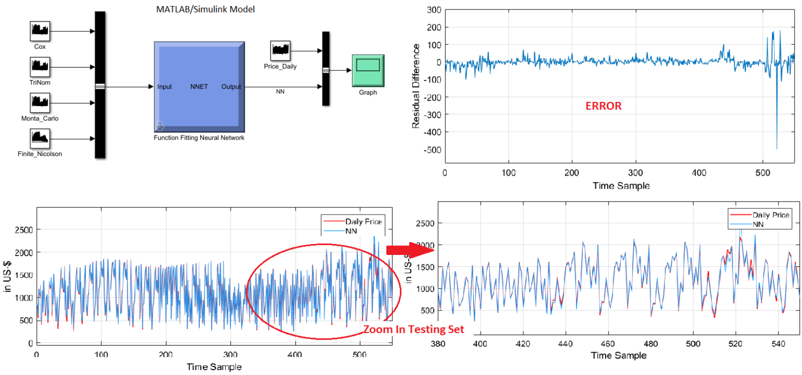
Figure:

Table of Contents

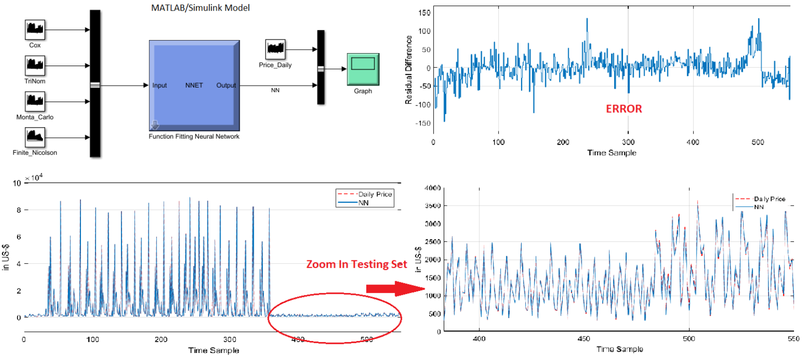
Coefficient of Determination (R^2) - Training, testing and validation sets.



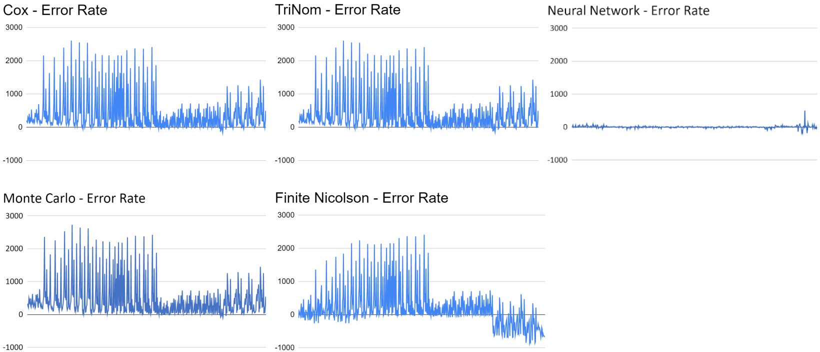
Performance Evaluation (Call - 550 sample))



Performance Evaluation (Put - 550 sample)



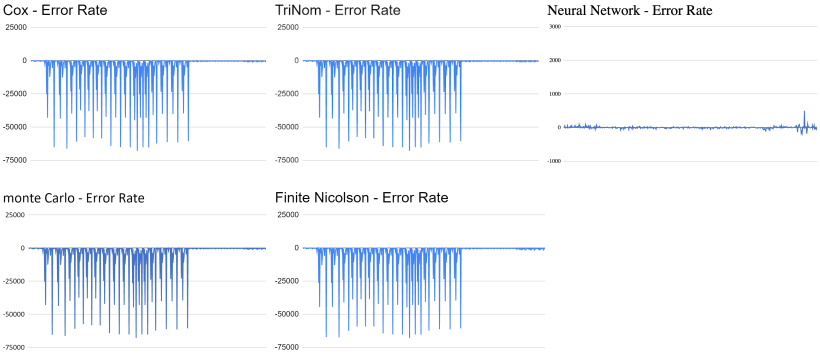
Error Rate Comparison - CALL



Performance Assessment - Call

	Cox	TriNom	Monte Carlo	Finite - Nicolson	Neural Network
MAE	319.9180328	319.7868852	352.9180328	414.7377049	31.47853449
MAPE	0.1990533858	0.1979574847	0.2277498435	0.3995503989	0.03268589453
MES	220391.2131	220724.3115	240785.0164	219942.4262	4088.136962

Error Rate Comparison - PUT



Performance Assessment - Put

	Cox	TriNom	Monte Carlo	Finite - Nicolson	Neural Network
MAE	276.7131148	276.4590164	306.9016393	354.6967213	29.94210565
MAPE	0.1553476095	0.1558498908	0.180309152	0.1908109449	0.026228912
MES	127468.9262	126606.0656	147293.0328	236427.0902	1508.775349

Table of Contents

Summary

- Neural Network has proven to be more effective compared to the conventional four models.

Summary

- Neural Network has proven to be more effective compared to the conventional four models.
- Single layer neural network gives better results compared to the multi-layer neural network.

Summary

- Neural Network has proven to be more effective compared to the conventional four models.
- Single layer neural network gives better results compared to the multi-layer neural network.
- **Future work:**
  Improve the learning process by gathering more data, and compare the proposed model with the NARX neural network (time series network)

References

- Amilon, H. (2003). A neural network versus blackâscholes: a comparison of pricing and hedging performances. J. Forecast. 22, 317â335. doi: 10.1002/for.867
- Binner, J. M., Bissoondeeal, R. K., Elger, T., Gazely, A. M., and Mullineux, A.W. (2005). A comparison of linear forecasting models and neural networks: an application to Euro inflation and Euro Divisia. Appl. Econ. 37, 665â680. doi: 10.1080/0003684052000343679
- Black, F., and Scholes, M. (1973). The pricing of options and corporate liabilities. J. Political Econ. 81, 637â654.
- Cox, J. C., Ross, S., and Rubinstein, M. (1979). Option pricing: a simplified approach. J. Financ. Econ. 7, 229â263.
- Liang, X., Zhang, H., Xiao, J., and Chen, Y. (2009). Improving option price forecasts with neural networks and support vector regressions.Neurocomputing 72, 3055â3065.
- Lin, C.-T., and Yeh, H.-Y. (2005). The valuation of taiwan stock index option price comparison of performances between black-scholes and neural network model. J. Stat. Manag. Syst. 8, 355â367. doi: 10.1080/09720510.2005.10701164
- Pagnottoni P (2019) Neural Network Models for Bitcoin Option Pricing. Front. Artif. Intell. 2:5. doi: 10.3389/frai.2019.00005
- Yao, J., Li, Y., and Tan, C. L. (2000).Option price forecasting using neural networks, Omega 28, 455â466. doi: 10.1016/S0305-0483(99)00066-3