Comparative study of Classical Black-Scholes model and Machine Learning Models for Indian Call Option Pricing

Dr. A. D. Chauhan

Assistant Professor, J & J College of Science, Nadiad

Abstract—This paper compares the option prices of Indian stock's call options computed by classical Black-Scholes model with some machine learning models. Five machine learning models, Multiple Linear Regression model, Polynomial regression model, Support Vector regression model, Decision tree regression model and Random Forest regression model have used in this comparison. First, we have trained these machine learning models using call data downloaded from NSE website. Then use them to predict call option prices. After this we have compared these option prices with Black Scholes prices and actual market prices. Then we have used Root Mean Square Error (RMSE) to check which model computes the best option prices for Indian stocks.

Index Terms—Black–Scholes Option Pricing Model, Multiple Linear Regression, Polynomial Regression, Decision Tree Regression, Random Forest Regression, Support Vector Regression, Indian Stock Market.

I. INTRODUCTION

Indian stock market is fifth largest market in the world and continuously growing day by day. India's derivative market is also very large. National Stock Exchange (NSE) is the world's largest derivative exchange. Among derivatives, the options are most important in terms of risk management, speculation and hedging our position in the equity market. Because of this, the brokers, traders and financial institutions are using options as a hedging instrument. In this paper we have compared the option prices of different stock's call options for Six different models.

Black-Scholes (BS) model is widely used to compute theoretical price of European options. This model takes Stock price, Strike price, Risk-free interest rate, Time to maturity and volatility of underlying asset as input parameter and computes the option price as an output parameter. But Black-Scholes model takes some non-realistic assumptions like, volatility of underlying is constant, underlying asset's price cannot fall below zero, interest rate is constant and known throughout the period of contract, underlying asset does not pay dividend, etc. In real market like Indian stock market, these assumptions of BS model rarely hold. Due to these limitations, classical Black-Scholes model does not give accurate option prices.

To overcome the limitations of BS model many practitioners and researchers have tried to generalize the original BS model by removing some assumptions of model. There are many generalized/modified BS models available in literatures to increase the accuracy of the original BS model, but none of them are able to find accurate option prices of the underlying assets in different market situations. In recent years, because of the rise in data-driven and machine learning algorithms, researchers are exploring the new dimensions for finding the options prices by using the increasing power of computers. To overcome the limitations of original BS model some machine learning techniques have been adopted in past few years. Using machine learning models, we can find the complex relationship between option prices and its determinants. The main benefit of machine learning models is that they do not assume non-realistic market situations like original BS model.

Previous studies have shown that options pricing using machine learning models are more accurate than Black-Scholes models with real market conditions. Majority studies on machine learning models mainly focused on developed stock exchanges, where very few studies focused on Indian stock exchanges. In this paper we have focused on the comparative analysis of the option prices computed by original BS model and some machine learning models for Indian stock options. Specifically, we compared classical Black Scholes model with five machine learning models: Multiple Linear Regression, Polynomial Regression, Decision Tree Regression, Random Forest

Regression, and Support Vector Regression. We have first trained all these machine learning models by providing the actual stock options data traded on National Stock Exchange of India. Then we compared the prices computed using machine learning models and Black-Scholes model to check the effectiveness of the models on Indian stock market.

A. Black-Scholes Option pricing Model
This model was first proposed by Fischer Black and Myron Scholes in 1973. Then it was extended by Robert Merton. The Black-Scholes model is the most significant mathematical model in modern financial developments. After solving the partial differential equation, it gives the closed form option pricing formula to find the theoretical price of European-style options. It considers the underlying asset price as a stochastic process following a geometric Brownian motion.

The option price can be computed by BS model as given below. The formula for computing theoretical price of European style call options first given in Black F., Scholes M. (1973) [1] which is given below.
S_0 = Today's stock price
X = Strike price
T = expiration time
σ = volatility of underlying stock
r = risk-free interest rate
Then the value, V_c of call today is given by,

$$V_c = S_0 N(d_1) - X e^{-rT} N(d_2) \qquad (1)$$

In this formula, $N(x)$ denotes the standard normal distribution function.

And,

$$d_1 = \frac{ln\left(\frac{S_0}{X}\right) + \left(r + \frac{\sigma^2}{2}\right)T}{\sqrt{T}\sigma} \quad \text{and} \quad d_2 = d_1 - \sigma\sqrt{T} \quad (2)$$

Further, a European call price is related to a European put price. We can find European put price of any Stock option if we have European call price of that Stock option by using Put-Call parity formula. The formula for finding European put price is,

$$V_p = V_c - S_0 + e^{-rT} X$$

Where,
V_p = Value of Put option
V_c = Value of Call option

S_0 = Today's stock price
X = Strike price
T = expiration time
r = risk-free interest rate

B. Multiple Linear Regression model
Multiple linear Regression model is a generalization of simple linear regression model. Simple linear regression model gives the relations between one input (Independent) variable and one output (Dependent) variable, while Multiple linear Regression model used to find relationship between one output variable and more than one input variables. This model assumes that there is a linear relationship between dependent and independent variables. It uses the formula, $Y = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \cdots + \beta_n X_n + \epsilon$, where Y is dependent variable, each X_i is independent variable, β_0 is the intercept, β_i are the coefficients representing the effect of each predictor (X_i), and ϵ is the error term. This model is significantly used in the field of engineering, finance, economics and social sciences. By help of coefficients, we can get the insights about which variable has more impact on the output variable. As multiple liner regression model depends on some assumptions like linearity, independence, homoscedasticity, and normality of residuals, if our data meets these assumptions, we can get the powerful tool to find the complex association between input and output variables.

C. Polynomial Regression Model
Polynomial regression model is also one type of extension of simple linear regression model. In polynomial regression model we assume that the dependent and independent variables are related by the nth degree polynomial. In simple linear regression model since the relation between two variables are linear, we can fit a straight line. While polynomial regression model fit a curve to find the relation between complex, non-linear patterns in our dataset. If our data shows non-straight trend then this model helps us to find relationship. Unlike simple linear regression model polynomial regression model also uses the method of least squares to obtain the coefficients, but it uses the higher-order terms like x^2, x^3 etc to efficiently approximate the relationship of variables. Sometimes polynomial regression models overfit the data if we choose too high degrees

of independent variables. Therefore, choosing appropriate polynomial degrees are essential for balancing accuracy and generalization. This model is mainly used in fields such as economics, finance, engineering, and natural sciences where real-world relationships are mostly not linear.

C. Decision Tree Regression Model
A Decision Tree Regression model is one type of supervised learning model used to predict the continuous numerical values coming from input variables. Unlike linear regression model, which assumes that the input and output variable is linearly related, decision tree regression model doesn't assume such relations and instead divides the data into subsets based on variables values. The structure of this model is like tree, where each internal node represents a decision rule on a feature, each branch represents an outcome of that decision, and each leaf node provides a predicted value. This structure of the model allows the model to define complex, non-linear relationships between features and the output variable. Decision tree regression model is comparatively easy to handle. Due to this, it is used for exploratory data analysis and practical problem-solving.

D. Random Forest Regression Model
Random Forest Regression model is some sort of extension of decision tree model. It's accuracy and robustness are higher than decision tree model. The random forest is a collection of decision trees which trains random subsets of dataset separately so the chance of overfitting and high variance is very low. The final output of the random forest model is obtained by taking average of the outputs of these trees. This approach works on the principle "wisdom of the crowd" which means the multiple weak models (individual trees) combines to form a strong model. This model is useful in handling complex and nonlinear relations, it predicts the interactions between variables and is efficient than individual decision trees models. Random forest is used to predict both numerical and categorical data and requires fewer parameters to make it powerful and easy to use tool for regression analysis. It is widely used in the domains of engineering, finance and healthcare.

E. Support Vector Regression Model
Support Vector Regression (SVR) model is a supervised machine learning algorithm which works on the principles of Support Vector Machines (SVM). This model was primarily used for regression tasks. This model mainly focuses on fitting the data inside the predefined margin of tolerance, which is known as epsilon intensive zone. This allows the model to generalize better by ignoring small deviations while emphasizing significant errors. SVR works by mapping input data into a high-dimensional feature space using kernel functions such as linear, polynomial, or radial basis function (RBF), where it seeks to find the optimal regression line or hyperplane that captures the underlying relationship. One of its key strengths lies in its ability to handle non-linear relationships effectively and maintain robustness against overfitting, especially in high-dimensional spaces. Due to these advantages, SVR is widely applied in areas like financial forecasting, engineering, and scientific modelling where both accuracy and generalization are crucial.

II. LITERATURE REVIEW

Black & Scholes (1973) [1] proposed the Black–Scholes (B-S) model for pricing European options, which is still commonly used to value a wide range of derivative securities. Though this model has enormous uses world-wide, it has some well-known deficiencies. This deficiency always attracts Researchers and Academicians to revise/modify Black-Scholes model. For example, observed returns of the underlying asset from financial markets are not normally distributed and they are usually skewed shown by Peiro (1999) [8] and fat-tailed shown by Rachev et al. (2005) [9]. There are many modified B-S model in literature in which log- returns of underlying asset not taken as normally distributed. Generalized beta distribution of the second kind was used by McDonald & Bookstaber (1991) [7] while Burr-3 distribution was adopted by Sherrick et al. (1996) [11]. Other examples include Weibull distribution used by Savickas (2002) [10], g-and-h distribution studied by Dutta & Babbel (2005) [2] and generalized gamma distribution taken by Fabozzi et al. (2009) [3].

As we have shown earlier, the Black Scholes models contain some non-realistic assumptions of market. To overcome these limitations of Black Scholes and

Generalised Black-Scholes models, practitioners and researchers have tried some machine learning algorithms as a data driven alternative for finding the option prices. These ML algorithms do not depend on the theoretical assumptions and can be used to find the complex, nonlinear relations between option price and input variables like spot price, strike price, time to maturity, volatility, and interest rate.

In the early literatures like Hutchinson, Lo, and Poggio (1994) [12] proposed the neural networks to compute the option prices. They have demonstrated that the machine learning algorithms could outperform the classical option pricing models in the imperfect market scenario. After that García and Gençay (2000) [13] compared different Machine learning models and suggested that data driven algorithms mostly gives more accurate price of options.

In the recent years, many supervised machine learning models like, Decision tree regression model, Random forest regression model and Support vector regression model have been successfully applied to find option prices. Bennell and Sutcliffe (2004) [14] have compared support vector regression (SVR) and BS model and found that the SVR model and found that SVR model outperformed the BS model. Gradojevic, Gencay, and Kukolj (2009) [15] have found that the random forest model could also used to predict the option prices with lower prediction errors.

Despite these advancements, most prior research focuses on developed markets, particularly the U.S. and European options data. Studies on the application of ML models to Indian stock options remain limited. Given the growing liquidity and complexity of Indian derivative markets, there is a strong motivation to test and compare the performance of classical and machine learning models in this context.

This study aims to fill this gap by evaluating the predictive accuracy of several machine learning models—Multiple Linear Regression, Polynomial Regression, Support Vector Regression, Decision Tree Regression, and Random Forest Regression—against the classical Black–Scholes model using call option data from the Indian stock market.

III. DATA & METHODOLOGY

First, we collect all the features which are required to find the theoretical option prices using classical Black-Scholes model. Then make them i.e. stock price, strike price, time to maturity, interest rate, volatility as input features and market value of option as output variable of all the five machine learning models Multiple Linear Regression, Polynomial Regression, Support Vector Regression, Decision Tree Regression, and Random Forest Regression. Then set the actual market price of option as an output variable. Now train all five ML models by giving the call options data from NSE website. After training all the ML models test the model by giving new data from market. Give only input features without giving market value of options contracts. So, each ML model will give you predicted option prices of all call options. Now compare this predicted option prices with actual market price of option contracts. Further we find the error between predicted price of ML models and market price of options. In second stage we find the theoretical option prices of all the call options by Black-Scholes model and then compare this computed option prices with actual market prices. Finally, we can now compare both techniques machine learning data-driven method and theoretical option pricing method from the prices we have computed.

This study has been carried out on the call options on stocks listed out on National Stock Exchange of India. We have first downloaded daily data of top 20 most active stock calls with respect to the volume of contract from June 2025 to August 2025. So, we have collected 300 call options data with its market value. After this we have collected the stock price, strike price, time to maturity, interest rate, volatility and actual market price of all these call option contracts.

We have used the python programming language to write all five machine learning algorithms. We have provided the same training data to all the ML models and trained each model separately. After this we have tested these ML models on the different stock's call options listed on 12th August 2025. We have selected the stock option contracts listed in Table-I to test the ML models.

Then we get the predicted option prices from all five ML models. Which are listed in Table-II.

Then we have computed the theoretical option price of the above listed stock's call options which are listed in Table-III

IV. RESULTS & DISCUSSION

The predictive performance of all six models was assessed relative to actual market prices of call options, using Root Mean Square Error (RMSE) as the evaluation criterion. Table-IV reports the comparative results.

Table-IV

Model	RMSE
Black–Scholes Model	80.23
Multiple Linear Regression	37.87
Polynomial Regression	10.28
Decision Tree Regression	33.16
Support Vector Regression	29.62
Random Forest Regression	34.84

The results reveal clear differences in pricing accuracy across models. The Black–Scholes model exhibited the highest error (RMSE = 80.23), confirming its limited ability to replicate market prices under Indian conditions. This finding is consistent with earlier studies that documented the systematic mispricing of Black–Scholes in emerging markets, primarily due to its reliance on constant volatility and lognormal return assumptions. The model tended to overestimate call option prices, particularly for options with longer maturities and higher volatility.

In contrast, all machine learning models achieved substantial reductions in prediction error. Among them, Polynomial Regression delivered the best performance with an RMSE of 10.28, significantly outperforming both traditional and other machine learning approaches. This suggests that introducing nonlinearity in the regression framework captures the curvature in option pricing relationships more effectively than either purely linear or more complex tree-based methods.

The Support Vector Regression model (RMSE = 29.62) also demonstrated strong performance, benefiting from its ability to map input variables into higher-dimensional feature spaces through kernel functions. This allowed it to model complex relationships between option features and prices without overfitting.

Tree-based models yielded mixed results. The Decision Tree Regression model (RMSE = 33.16) improved accuracy relative to Black–Scholes and linear regression, but its tendency to overfit local patterns likely limited generalization. The Random Forest Regression model (RMSE = 34.84) provided more stability by averaging across multiple trees, yet its error remained slightly higher than SVR and polynomial regression. This may be attributed to the moderate dataset size, which can constrain ensemble methods from realizing their full potential.

The Multiple Linear Regression model (RMSE = 37.87) outperformed Black–Scholes but lagged behind all nonlinear approaches. This highlights the inadequacy of linear models in capturing the nonlinear payoff structures of options, where interactions among strike price, time to maturity, volatility, and interest rates play a decisive role.

Overall, the analysis demonstrates that machine learning models consistently outperform the Black–Scholes benchmark in pricing Indian equity call options. In particular, Polynomial Regression emerges as the most effective model, indicating that relatively simple nonlinear modelling can rival or surpass more advanced algorithms in predictive accuracy. These findings support the growing view that data-driven methods can provide more accurate and flexible tools for option pricing in emerging markets, where market inefficiencies and volatility dynamics often challenge classical models.

V. CONCLUSION

This study conducted a comparative analysis of the Black–Scholes model and five machine learning techniques—Multiple Linear Regression, Polynomial Regression, Decision Tree Regression, Random Forest Regression, and Support Vector Regression—for pricing call options in the Indian equity market. Using actual market prices as the benchmark, the performance of each model was evaluated in terms of Root Mean Square Error (RMSE). The results clearly demonstrate that machine learning approaches substantially outperform the traditional Black–Scholes framework in capturing the dynamics of option prices.

Among the tested models, Polynomial Regression achieved the lowest RMSE (10.28), indicating that even a relatively simple nonlinear extension of regression is highly effective in approximating option pricing patterns. Support Vector Regression (RMSE = 29.62) and tree-based models also delivered strong performance, while Multiple Linear Regression (RMSE = 37.87) showed modest

improvement over Black–Scholes (RMSE = 80.23). These findings highlight that nonlinearity plays a crucial role in option valuation and that data-driven approaches are better suited to address market imperfections and volatility structures prevalent in emerging markets like India.

The practical implication of these results is that traders, analysts, and risk managers may benefit from incorporating machine learning–based models into their pricing and hedging strategies, rather than relying solely on classical models. In particular, polynomial and kernel-based models provide flexible yet computationally efficient alternatives that can be readily implemented in real trading environments.

However, this study has certain limitations. The analysis was restricted to call options on a selected set of Indian equities and used RMSE as the sole evaluation metric. Incorporating a larger dataset, including put options and multiple maturities, may provide a more comprehensive picture. Furthermore, future research could extend this comparison to advanced deep learning architectures, hybrid models combining stochastic processes with machine learning, or volatility forecasting frameworks that feed into pricing models.

In conclusion, the evidence suggests that machine learning models, particularly nonlinear regressions, offer a promising complement to traditional financial models for option pricing in India. By bridging theoretical finance with data-driven analytics, such approaches have the potential to enhance accuracy, adaptability, and decision-making in rapidly evolving derivative markets.

REFERENCES

[1] Black, F., Scholes, M., 1973, 'The pricing of options and corporate liabilities', J. Polit. Econ. 637- 654.

[2] Dutta, K.K., Babbel, D.F., 2005, 'Extracting probabilistic information from the prices of interest rate options: Tests of distributional assumptions', J. Bus. 78 (3) 841–870.

[3] Fabozzi, F.J., Tunaru, R., Albota, G., 2009, 'Estimating risk-neutral density with parametric models in interest rate markets', Quant. Finance 9 (1) 55–70.

[4] Godman, V., Stampfli, J., 'The Mathematics of Finance: Modelling and Hedging', AMS.

[5] Khan, M.U., Gupta, A., Siraj, S., Ravichandran, N., 2012, 'The Overview of Financial Derivative and its Products', International Journal of Finance & Marketing, 2(3), 57-72.

[6] Khan, M. U., Gupta, A., Siraj, S., 2013, 'Empirical testing of modified Black-Scholes option pricing model formula on NSE derivative market in India', International Journal of Economics and Financial Issues, Vol. 3, No. 1, pp. 87-98.

[7] McDonald, J.B., Bookstaber, R.M., 1991, 'Option pricing for generalized distributions', Comm. Statist. Theory Methods, 20 (12) 4053–4068.

[8] Peiro, A., 1999, 'Skewness in financial returns', J. Banking Finance 23 (6) 847–862.

[9] Rachev, S.T., Menn, C., Fabozzi, F.J., 2005, 'Fat-tailed and Skewed Asset Return Distributions: Implications for Risk Management, Portfolio Selection, and Option Pricing', John Wiley & Sons, Vol. 139.

[10] Savickas, R. 2002, 'A simple option-pricing formula', Financ. Rev. 37 (2) 207–226.

[11] Sherrick, B.J., Garcia, P., Tirupattur, V., 1996, 'Recovering probabilistic information from option markets:Tests of distributional assumptions', J. Futures Mark. 16 (5) 545–560.

[12] Hutchinson, J. M., Lo, A. W., & Poggio, T. (1994). A non-parametric approach to pricing and hedging derivative securities via learning networks. Journal of Finance, 49(3),851-889. https://doi.org/10.1111/j.1540-6261.1994.tb00081.x

[13] Garcia, R., & Gençay, R. (2000). Pricing and hedging derivative securities with neural networks and a homogeneity hint. Journal of Econometrics, 94(1-2), 93-115. https://doi.org/10.1016/S0304-4076(99)00018-4

[14] Bennell, J., & Sutcliffe, C. (2004). Black–Scholes versus artificial neural networks in pricing FTSE 100 options. Intelligent Systems in Accounting, Finance and Management,12(4),243-260. https://doi.org/10.1002/isaf.254

[15] Gradojevic, N., Gençay, R., & Kukolj, D. (2009). Option pricing with modular neural networks. Working Paper Series No. 32_09, Rimini Centre for Economic Analysis. Available at https://ssrn.com/abstract=959544

Table-I

Symbol	Stock price	Strike price	Observed date	Expiry date	Market price
MARUTI	12,847.00	13,000.00	12-08-2025	28-08-2025	95
HAL	4,420.00	4,500.00	12-08-2025	28-08-2025	76
RELIANCE	1,381.00	1,400.00	12-08-2025	28-08-2025	14.2
TCS	3,039.80	3,100.00	12-08-2025	28-08-2025	27.25
HAL	4,420.00	4,600.00	12-08-2025	28-08-2025	46.8
HAL	4,420.00	4,400.00	12-08-2025	28-08-2025	117.35
HAL	4,420.00	4,700.00	12-08-2025	28-08-2025	28.5
MARUTI	12,847.00	12,900.00	12-08-2025	28-08-2025	135
MARUTI	12,847.00	13,500.00	12-08-2025	28-08-2025	19.45
TITAN	3,485.00	3,500.00	12-08-2025	28-08-2025	47.3
HAL	4,420.00	4,800.00	12-08-2025	28-08-2025	17.6
LT	3,679.80	3,700.00	12-08-2025	28-08-2025	38
SBIN	821.1	830	12-08-2025	28-08-2025	6.7
MARUTI	12,847.00	13,200.00	12-08-2025	28-08-2025	49.9
TCS	3,039.80	3,200.00	12-08-2025	28-08-2025	8.75
RELIANCE	1,381.00	1,390.00	12-08-2025	28-08-2025	17.9
SBIN	821.1	820	12-08-2025	28-08-2025	11.7
TITAGARH	801.45	800	12-08-2025	28-08-2025	26.5
SBIN	821.1	840	12-08-2025	28-08-2025	3.7
SIEMENS	3,129.20	3,200.00	12-08-2025	28-08-2025	50.8

Table-II

Symbol	Market	Multi Lin	Polynomial	Decision tree	Support vec	Random forest
MARUTI	95	143.57	106.49	76.2	67.41	88.28
HAL	76	46.68	85.9	59	76.43	92.175
RELIANCE	14.2	-31.85	15.97	15.9	16.66	15.5
TCS	27.25	-12.6	29.21	40.1	24.82	52.53
HAL	46.8	29.98	62.98	59	76.87	92.17
HAL	117.35	63.37	109.39	59	75.99	92.51
HAL	28.5	13.29	40.54	59	77.31	92.17
MARUTI	135	160.26	122.86	185	67.41	113.39
MARUTI	19.45	60.1	28.97	76.2	67.41	88.28
TITAN	47.3	7.98	50.75	96.65	30.85	58.27
HAL	17.6	-3.4	18.5	59	77.74	92.17
LT	38	16.86	59.05	96.65	38.18	56.13
SBIN	6.7	-35.32	10.22	8	15.38	10.39
MARUTI	49.9	110.18	74.9	76.2	67.42	88.28
TCS	8.75	-29.28	7.7	40.1	25.06	52.53
RELIANCE	17.9	-30.18	18.25	24.2	16.64	19.64
SBIN	11.7	-33.65	12.53	13.4	15.36	11.78
TITAGARH	26.5	32.34	21.79	2.8	37.23	9.4
SBIN	3.7	-36.99	7.93	4.6	15.41	4.94
SIEMENS	50.8	24.25	56.74	84.05	61.25	91.83

Table-III

Symbol	Market	BS price
MARUTI	95	266.8466
HAL	76	161.586
RELIANCE	14.2	29.39917

TCS	27.25	53.14235
HAL	46.8	123.1016
HAL	117.35	208.0681
HAL	28.5	91.99985
MARUTI	135	312.5494
MARUTI	19.45	107.1858
TITAN	47.3	93.06678
HAL	17.6	67.45875
LT	38	107.4726
SBIN	6.7	20.35126
MARUTI	49.9	189.8968
TCS	8.75	23.98471
RELIANCE	17.9	33.85011
SBIN	11.7	25.12259
TITAGARH	26.5	50.25632
SBIN	3.7	16.24937
SIEMENS	50.8	106.413