Forecasting Stock Price Movements: A Comparative Analysis of Machine Learning Techniques

AISHIK PAL [1] HIMMAT PATEL [2] ANOOP BEHERA[3]
HARDIK SINGH[4] JAGDISH SUTHAR[5]

[1]Indian Institute of Technology Jodhpur
{b22ee085,b22ee033,b22cs011,b22ee092,b22ai064}@iitj.ac.in

Abstract

Stock exchange is a significant factor in the capital market and often is indicative of the state of a country's economy. This paper focuses on the involuted landscape of stock market analysis, utilising a diverse set of features and indicators extracted from historical stock data and machine learning techniques. We have utilised traditional statistical methods, namely, Moving Average Convergence Divergence (MACD), Moving Average (MA), Relative Strength Index (RSI), Stochastic Oscillator (%K and %D), Volume Moving Average (Volume MA), Volume Rate of Change (Volume ROC), Average True Range (ATR), volatility indicators, Williams %R (%R), and candlestick patterns. Machine Learning techniques are proving to be much more accurate and faster compared to traditional prediction techniques. The models used include K-Nearest Neighbors (KNN), Support Vector Machine (SVM), Decision Tree, Artificial Neural Network (ANN), Bayesian Learning, Perceptron, and Regression models. The objective is to predict market trends as either upward or downward based on preceding-day stock parameters. Through simulation, it is demonstrated that Random Forest emerges as the most effective model. Thus, we propose it for real-time implementation as a headway in market trend prediction methodologies.

Keywords: Machine Learning, Stock Price Prediction, yfinance, Moving Average Convergence Divergence (MACD), Moving Average (MA), Relative Strength Index (RSI), Stochastic Oscillator (%K and %D), Volume Moving Average (Volume MA), Volume Rate of Change (Volume ROC), Average True Range (ATR), Volatility Indicators, Williams %R (%R), Candlestick Patterns, K-Nearest Neighbors (KNN), Support Vector Machine (SVM), Decision Tree, Artificial Neural Network (ANN), Bayesian Learning, Perceptron, Regression.

Contents

1 Introduction

A stock market or share market is the aggregation of buyers and sellers of stocks (also called shares). These stocks are used to represent ownership claims on businesses. The fluctuations in stock market values have significant implications for stakeholders' profitability. When market prices rise, and stocks are readily available, stakeholders reap profits from their investments. Conversely, downturns in market prices lead to losses for stakeholders. Investors aim to purchase stocks at lower prices and sell them at higher prices to maximise profits (Tae Kyun Lee et al., 2019). Predicting future market movements has long captivated individuals with its blend of adventure, allure, and financial risk. Researchers from various disciplines, including business and computer science, have delved into stock market prediction, employing diverse methodologies and algorithms to analyse market dynamics. The choice of attributes in prediction models often depends upon the aspects of market performance we wish to capture. Investors typically engage in two types of stock analyses before making investment decisions: fundamental analysis and technical analysis. Fundamental analysis entails evaluating the intrinsic value of stocks and assessing industry performance, economic conditions, and political climate. On the other hand, technical analysis involves scrutinising market statistics derived from past prices and trading volumes to anticipate future market trends. Machine learning algorithms present promising avenues for forecasting market movements by utilising historical data and discerning underlying patterns. In this study, we wish to investigate the efficacy of various machine learning techniques in predicting stock prices using a comprehensive array of features extracted from historical stock data. Through empirical analysis and evaluation, we aim to examine the effectiveness of these techniques in navigating the complexities of stock market prediction and possibly gain insights that may empower investors and stakeholders in their decision-making processes.

1.1 Citing Paper

Previous studies in stock market prediction have primarily centred on conventional statistical methods and econometric models. However, the emergence of machine learning has catalysed a paradigm shift towards data-driven approaches capable of capturing obscure nonlinear relationships within financial data. Some articles have proposed prediction systems for stock market prices based on historical exchange scenarios using machine learning techniques [6]. Others have employed deep learning models like Recurrent Neural Networks (RNNs), particularly the Long Short-Term Memory (LSTM) model, to forecast future stock market values [3, 8]. Additionally, some researchers have integrated additional attributes such as oil rates, gold and silver prices, interest rates, foreign exchange rates, and sentiment analysis derived from news and social media feeds [4, 7, 9]. Despite these advancements, there remains a gap in the literature regarding the prediction of future market trends and the comparative analysis of multiple algorithms using standardised feature sets.

1.2 Figures

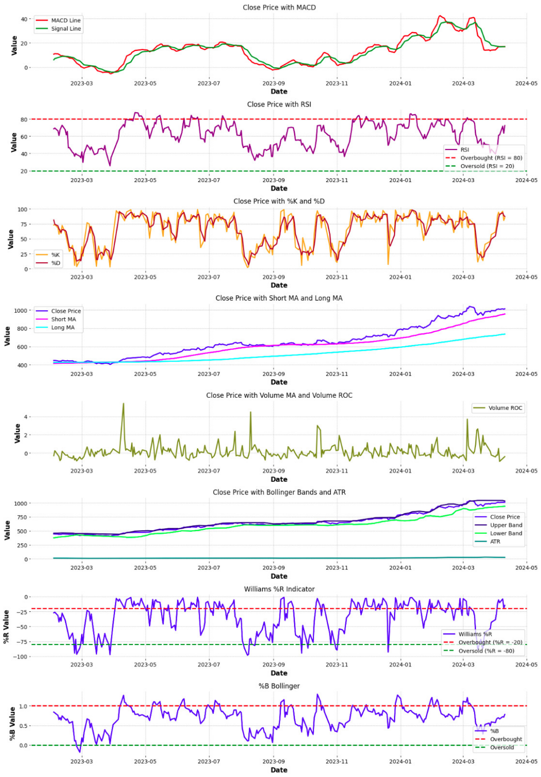

Figure 1: Various Indicators applied on the data

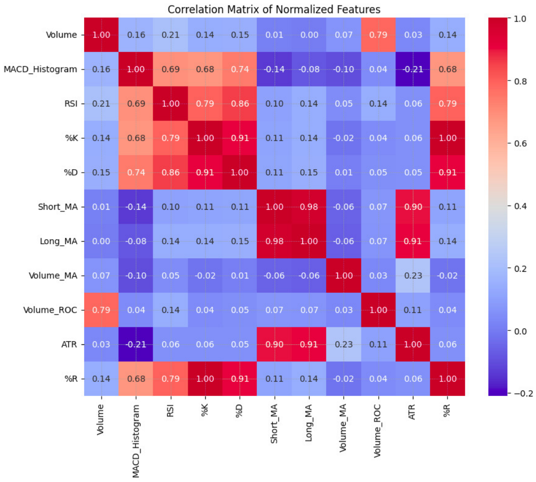

Figure 2: cool-warm Correlation Matrix

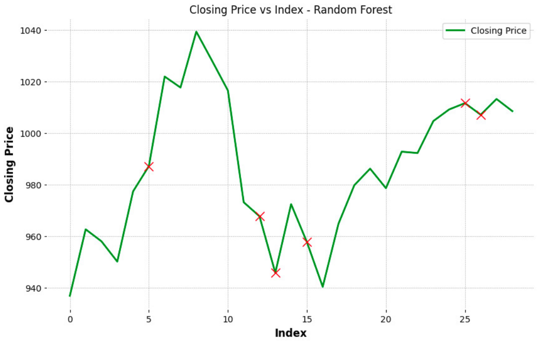

Figure 3: Misclassification in the test data on closing price

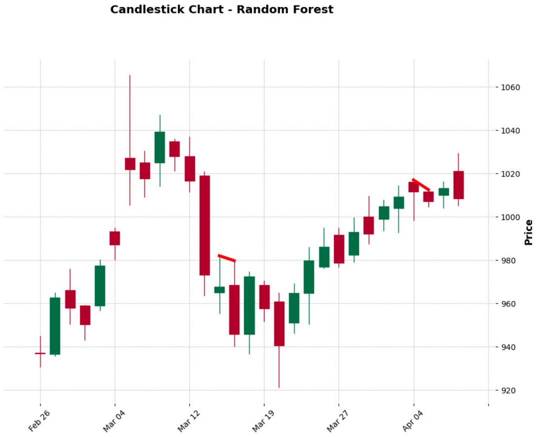

Figure 4: Misclassification in the test data on candlestick

2 Approaches Tried

This study explores several machine learning approaches to predict stock market trends and movements based on the extracted technical indicators. These approaches encompass diverse algorithms and methodologies, each offering unique strengths and capabilities in analysing and interpreting historical stock data.

1. **K-Nearest Neighbors (KNN)**: KNN is a non-parametric algorithm used for classification and regression tasks. In our study, we implemented KNN using historical stock data to get training data points with indicators like MACD, MA, RSI, and volume-related indicators as features. KNN calculates the similarity between data points in the feature space and predicts the class or value based on the majority vote or averaging of its k-nearest neighbours.

2. **Support Vector Machine (SVM)**: SVM is a supervised learning algorithm used for classification and regression tasks. SVM constructs a hyperplane that best separates different classes of data points in a high-dimensional feature space. By maximising the margin between classes, SVM aims to improve generalisation. We utilised different kernel functions- linear, polynomial, and radial basis function (RBF) to try to capture any nonlinear relationships in the data [1].

3. **Decision Tree**: Decision trees are versatile algorithms capable of handling both classification and regression tasks. Decision trees recursively partition the feature space into hierarchical decision nodes. Each node represents a test/decision on an attribute/feature, and each branch is an outcome of that decision. Decision trees are easily interpretable, making them suitable for analysing stock market trends and identifying key features that may influence price movements.

4. **Artificial Neural Network (ANN)**: ANN is a deep learning algorithm inspired by the structure and function of biological neural networks in animal brains. An ANN is a multi-layer perceptron with multiple hidden layers. ANN learns complex nonlinear mappings between input features and output predictions through forward propagation and backpropagation and performing gradient descent. ANN aims to minimise prediction errors and optimise predictive performance by adjusting the weights and biases of neurons. We used the stock price indicators in the ANN's input layer and labels for price movement in the output layer.

5. **Naive Bayes classifier**: Naive Bayes classifiers are a family of linear "probabilistic classifiers" which assume that the features are conditionally independent, given the target class. In our study, we applied the Gaussian Naive Bayes classifier, which also assumes that the continuous values associated with each class are distributed according to a normal (or Gaussian) distribution.

6. **Perceptron** : Perceptron is a simple linear classifier used for binary classification tasks. We explored the application of perceptron for stock market prediction by training a linear predictor function with a binary output: price moves up or down. Perceptron iteratively adjusts weights based on misclassified samples to minimise the classification error. While perceptron may not capture complex nonlinear relationships in the data, it provides a computationally efficient baseline for evaluating predictive performance.

7. **Regression**: Regression analysis is a statistical technique used for predicting continuous variables based on one or more independent variables. In our study, we employed the linear regression model. We tried to estimate the relationship between input features and price movement by fitting the regression function to the historical stock data.

8. **Random Forest**:Random Forest is a machine learning algorithm that operates by constructing multiple decision trees during training and outputting the mode of the classes (classification) or the mean prediction (regression) of the individual trees. Each tree in the ensemble is built from a random subset of the training data and a random subset of features, introducing diversity and reducing overfitting.During prediction, each tree independently provides a prediction, and the final output is determined by aggregating these predictions.

9. **XGBoost**: It works by sequentially adding weak learners, typically decision trees, to an ensemble model, with each tree learning and correcting the errors of the previous ones. XGBoost employs a

gradient boosting framework, optimizing a differentiable loss function to minimize errors. It incorporates regularization techniques to prevent overfitting and parallel processing to enhance computational efficiency.

3 Experiments and Results

3.1 Dataset

We made use of the "yfinance" library to obtain the stock price data. It offers a seamless interface for accessing an extensive data repository hosted by Yahoo Finance. This dataset encompasses a broad spectrum of financial information, spanning historical stock prices, trading volumes, key performance indicators, and company statistics.

With its provision of real-time stock prices, Yfinance equips users with timely insights into market movements, enabling traders to capitalise on fleeting opportunities and adapt their strategies in response to dynamic developments. This real-time data feed is a crucial asset for informed decision-making in the fast-paced world of finance.

3.2 Experimental Setting

A dataset was extracted from yfinance for a specific stock over the defined period. This dataset encompassed a multitude of historical data points, including open, high, low, close prices, and trading volumes. From this dataset, we calculated a range of technical indicators - MACD, MA, RSI, %K, %D, %B, Volume MA, Volume ROC, ATR, volatility indicators, %R, and candlestick patterns. Each indicator was computed using appropriate formulas tailored to capture crucial trends and patterns in the data.

To enhance our analysis, we utilised an open-source candlestick library from GitHub to label each data point with its corresponding candlestick pattern. This labelling process augmented our dataset with additional information about market sentiment and potential trend reversals.

Subsequently, the dataset has two parts: features and labels. The features comprise the indicators mentioned above. Meanwhile, the labels consisted of binary values representing the trend direction (up or down) corresponding to the mean of the next three days.

The dataset was further partitioned into training and testing subsets to facilitate model training and evaluation. The training data were used to train various machine learning models, including K-Nearest Neighbors (KNN), Support Vector Machine (SVM), Decision Tree, Artificial Neural Network (ANN), and Naive Bayes classifier (Gaussian). Each model was trained on the feature-rich training dataset to learn the underlying patterns and relationships between features and labels.

Once trained, these models were then deployed to predict the future or real-time trend of the stock market. By leveraging the learned insights from the training data, the models aimed to forecast the direction of future price movements, thereby assisting investors and stakeholders in making informed decisions in the dynamic stock market environment.

3.3 Comparison of Results

To assess the performance of the proposed models, we utilise evaluation metrics viz, Accuracy, Precision, Recall, and F1-Score. These metrics provide insights into the models' ability to classify stock market trends and movements accurately. Precision and Recall, crucial components in evaluating these metrics, are calculated based on True Positive (TP), False Positive (FP), True Negative (TN), and False Negative (FN) rates, as defined in Eqs. (1)–(4).

Precision is computed as the weighted average of precision for positive and negative classes, while Recall is the weighted average of recall for positive and negative classes. Accuracy and F1-Score are then derived using Eqs. (5) and (6) respectively.

Table 1 presents the performance of various classifiers on datasets pertaining to "ADANI ENTERPRISES" stock prices of 1 day over a period of two years. While the accuracies across classifiers hover around 60%, the Decision Tree exhibits a slightly higher accuracy, approaching 66%.

The Random Forest's superior accuracy may stem from its capability to handle obscure relationships between features and target variables. Unlike classifiers such as KNN and linear SVM, Random Forest is less sensitive to data scaling and data normalisation.

Market fluctuations introduce noise into the data, potentially affecting models like ANN and Naive Bayes. ANN's performance may be impacted by its sensitivity to noisy data. At the same time, Naive

Bayes's assumption of feature independence may not hold true in the presence of complex relationships between technical indicators. We see from Figure .4 that some points are misclassified on certain days when the market shift abruptly for a short period of time.

In summary, while all classifiers demonstrate reasonable accuracy, the Random Forest emerged as the preferred choice due to its ability to handle complex relationships and its robustness to noisy data, characteristics essential for effective stock market prediction.

4 Summary

This project aims to predict stock market trends and movements using machine learning techniques and technical indicators extracted from historical stock data. By leveraging a diverse set of technical indicators, including Moving Average Convergence Divergence (MACD), Relative Strength Index (RSI), Stochastic Oscillator, and Volume Rate of Change (Volume ROC), among others, we sought to capture key trends and patterns in the data. The dataset was sourced from yfinance, technical indicators were calculated for features and labels were extracted to train the ML models.

Various machine learning algorithms, including K-Nearest Neighbors (KNN), Support Vector Machine (SVM), Decision Tree, Artificial Neural Network (ANN), and Naive Bayes classifier, were trained and evaluated on the dataset to predict future stock market trends. Performance metrics such as Accuracy, Precision, Recall, and F1-Score were utilised to assess the effectiveness of each model in capturing market dynamics and predicting price movements.

4.1 Conclusion and Future Work

In conclusion, our study demonstrates the feasibility of using machine learning techniques and technical indicators to reasonably predict stock market trends. The Random Forest emerged as the preferred model, exhibiting superior performance in handling complex relationships between features and target variables and robustness to noisy data. The utilisation of diverse technical indicators provided valuable insights into market trends and potential buy or sell signals, enhancing the predictive capabilities of the models.

Despite the promising results, it is important to acknowledge the inherent challenges and limitations in stock market prediction, including market volatility, data noise, and uncertainty. Future research endeavours may focus on refining existing models, exploring ensemble learning and deep learning techniques, and integrating alternative data sources, such as sentiment analysis from news and social media feeds, to improve prediction accuracy and reliability.

5 Formulas

The formulae used for evaluation:

1. **Precision (Positive Class):**
$$\text{Precision}_{\text{positive}} = \frac{\text{TP}}{\text{TP} + \text{FP}} \tag{1}$$

2. **Precision (Negative Class):**
$$\text{Precision}_{\text{negative}} = \frac{\text{TN}}{\text{TN} + \text{FN}} \tag{2}$$

3. **Recall (Positive Class):**
$$\text{Recall}_{\text{positive}} = \frac{\text{TP}}{\text{TP} + \text{FN}} \tag{3}$$

4. **Recall (Negative Class):**
$$\text{Recall}_{\text{negative}} = \frac{\text{TN}}{\text{TN} + \text{FP}} \tag{4}$$

5. **Accuracy:**
$$\text{Accuracy} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{FP} + \text{TN} + \text{FN}} \tag{5}$$

6. **F1-Score:**
$$\text{F1-score} = \frac{2 \times \text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}} \tag{6}$$

Formulae used for calculating Technical indicators [2, 5]:

1. **Moving Average (MA):**
   - *Formula:* $\text{MA} = \frac{\text{Sum of Closing Prices over 'n' periods}}{n}$
   - *Description:* MA is a widely used technical analysis tool that smooths out price data by creating a constantly updated average price.
2. **Moving Average Convergence Divergence (MACD):**
   - *Formula:* $\text{MACD Line} = \text{12-day EMA} - \text{26-day EMA}$
   - *Description:* MACD is a trend-following momentum indicator that shows the relationship between two moving averages of a security's price.
3. **Relative Strength Index (RSI):**
   - *Formula:* $\text{RSI} = 100 - (100 \div (1 + \text{RS}))$, where $\text{RS} = \frac{\text{Average of upward price changes over 'n' periods}}{\text{Average of downward price changes over 'n' periods}}$
   - *Description:* RSI is a momentum oscillator that measures the speed and change of price movements. It oscillates between 0 and 100.
4. **Stochastic Oscillator (%K and %D):**
   - *Formula (for %K):* $\%K = \left(\frac{\text{Current Close} - \text{Lowest Low}}{\text{Highest High} - \text{Lowest Low}}\right) \times 100$
   - *Formula (for %D):* $\%D = \text{3-day SMA of \%K}$
   - *Description:* %K and %D are momentum indicators that compare a closing price to its price range over a given period. They oscillate between 0 and 100.
5. **Volume Moving Average (Volume MA):**
   - *Formula:* $\text{Volume MA} = \frac{\text{Sum of Trading Volume over 'n' periods}}{n}$
   - *Description:* Volume MA is a moving average of trading volume over a specified period, used to identify changes in trading activity.
6. **Volume Rate of Change (VolumeROC):**
   - *Formula:* $\text{VolumeROC} = \left(\frac{\text{Current Volume} - \text{Volume 'n' periods ago}}{\text{Volume 'n' periods ago}}\right) \times 100$
   - *Description:* VolumeROC measures the percentage change in trading volume over a specified period, indicating shifts in market interest.
7. **Average True Range (ATR):**
   - *Formula:* $\text{ATR} = \text{Average of (True Range over 'n' periods)}$
   - *Description:* ATR measures market volatility by calculating the average range between the high and low prices over a specified period.
8. **Volatility Indicator:**
   - *Formula:* Volatility Indicator (e.g., Bollinger Bands, Average True Range) varies depending on the specific indicator chosen.
   - *Description:* Volatility indicators measure the degree of variation in a security's price over time, providing insights into market volatility and potential price movements.
9. **Williams %R (%R):**
   - *Formula:* $\%R = \left(\frac{\text{Highest High} - \text{Current Close}}{\text{Highest High} - \text{Lowest Low}}\right) \times -100$

- *Description:* %R is a momentum oscillator that measures overbought or oversold conditions in a security. It oscillates between -100 and 0.

10. **Candlestick Patterns:**
    - *Description:* Candlestick patterns are graphical representations of price movements over a specified time period, typically displayed as red (bearish) or green (bullish) candles. They provide visual cues about market sentiment and potential trend reversals.

11. **%B (Percent B) Indicator:**
    - *Formula:* $\%B = \frac{\text{Close}-\text{Lower Band}}{\text{Upper Band}-\text{Lower Band}}$
    - *Description:* %B is a technical analysis indicator that measures a security's relative position compared to its upper and lower Bollinger Bands. Developed by John Bollinger, %B helps traders identify potential buying or selling opportunities based on price volatility.

References

[1] Malti Bansal, Apoorva Goyal, and Apoorva Choudhary. Stock market prediction with high accuracy using machine learning techniques. *Procedia Computer Science*, 215:247–265, 2022. ISSN 1877-0509. `doi:https://doi.org/10.1016/j.procs.2022.12.028`. URL `https://www.sciencedirect.com/science/article/pii/S1877050922020993`. 4th International Conference on Innovative Data Communication Technology and Application.

[2] P Kavya, S Saagarika, R Subavarsshini, C Nivetheni, Marimuthu Muthuvel, and Palaniswamy Velvadivu. Stock market analysis. page 7, 07 2021.

[3] Adil Moghar and Mhamed Hamiche. Stock market prediction using lstm recurrent neural network. *Procedia Computer Science*, 170:1168–1173, 2020. ISSN 1877-0509. `doi:https://doi.org/10.1016/j.procs.2020.03.049`. URL `https://www.sciencedirect.com/science/article/pii/S1877050920304865`. The 11th International Conference on Ambient Systems, Networks and Technologies (ANT) / The 3rd International Conference on Emerging Data and Industry 4.0 (EDI40) / Affiliated Workshops.

[4] Ishita Parmar, Navanshu Agarwal, Sheirsh Saxena, Ridam Arora, Shikhin Gupta, Himanshu Dhiman, and Lokesh Chouhan. Stock market prediction using machine learning. In *2018 First International Conference on Secure Cyber Computing and Communication (ICSCCC)*, pages 574–576, 2018. `doi:10.1109/ICSCCC.2018.8703332`.

[5] Jigar Patel, Sahil Shah, Priyank Thakkar, and K Kotecha. Predicting stock and stock price index movement using trend deterministic data preparation and machine learning techniques. *Expert Systems with Applications*, 42(1):259–268, 2015. ISSN 0957-4174. `doi:https://doi.org/10.1016/j.eswa.2014.07.040`. URL `https://www.sciencedirect.com/science/article/pii/S0957417414004473`.

[6] Reshma R, Usha S, Sathiyavathi V, and Sairamesh Lakshmanan. *Stock Market Prediction Using Machine Learning Techniques*, pages 331–340. 11 2021. `isbn:978-1-64368-219-8`. `doi:10.3233/APC210156`.

[7] Nusrat Rouf, Majid Bashir Malik, Tasleem Arif, Sparsh Sharma, Saurabh Singh, Satyabrata Aich, and Hee-Cheol Kim. Stock market prediction using machine learning techniques: A decade survey on methodologies, recent developments, and future directions. *Electronics*, 10(21), 2021. ISSN 2079-9292. `doi:10.3390/electronics10212717`. URL `https://www.mdpi.com/2079-9292/10/21/2717`.

[8] Aryendra Singh, Priyanshi Gupta, and Narina Thakur. An empirical research and comprehensive analysis of stock market prediction using machine learning and deep learning techniques. *IOP Conference Series: Materials Science and Engineering*, 1022(1):012098, jan 2021. `doi:10.1088/1757-899X/1022/1/012098`. URL `https://dx.doi.org/10.1088/1757-899X/1022/1/012098`.

[9] Mehak Usmani, Syed Hasan Adil, Kamran Raza, and Syed Saad Azhar Ali. Stock market prediction using machine learning techniques. In *2016 3rd International Conference on Computer and Information Sciences (ICCOINS)*, pages 322–327, 2016. `doi:10.1109/ICCOINS.2016.7783235`.

A Contribution of each member

1. **AISHIK PAL** contributed to writing code and preparing video recording.
2. **HIMMAT PATEL** contributed to writing code, technical analysis, cleaned the data.
3. **ANOOP BEHERA** contributed to write code and report.
4. **HARDIK SINGH** contributed to writing code and preparing the project page.
5. **JAGDISH SUTHAR** contributed to writing code and making slides.