Financial Market Patterns: A Dimensionality Reduction Approach Using PCA

Group No. 27

Name	Matriculation No.
Sadia Afrin Mou	23214190
Pretom Das Hira	23419208
Md. Anisul Haque Sajeeb	23077892
Tanjil Islam Shuvo	23384206

Friedrich-Alexander-Universität
Erlangen-Nürnberg

Agenda

- Introduction
- Dimensionality Reduction
- Principal Component Analysis (PCA) – Overview
- Mathematical Explanation of PCA
- PCA in Stock Market Data
- Result and Analysis
- Conclusion and Future Directions

Introduction

- Exploring Dimensionality Reduction with **PCA**.
- To simplify complex financial datasets while retaining essential patterns and trends.
- **Stock market generates massive amounts of time-series data:** prices, volumes, and technical indicators.
- **Technical Indicators:** Moving Averages (MA), Relative Strength Index (RSI), Bollinger Bands.
- Often characterized by **high dimensionality**, **redundancy**, and noise, making analysis challenging.
- Noise and redundancy make machine learning **models** prone to **overfitting**.
- PCA helps handle the complexity of financial data effectively while retaining crucial information for better predictions.

Technical Indicators Overview

What is Technical Indicators?

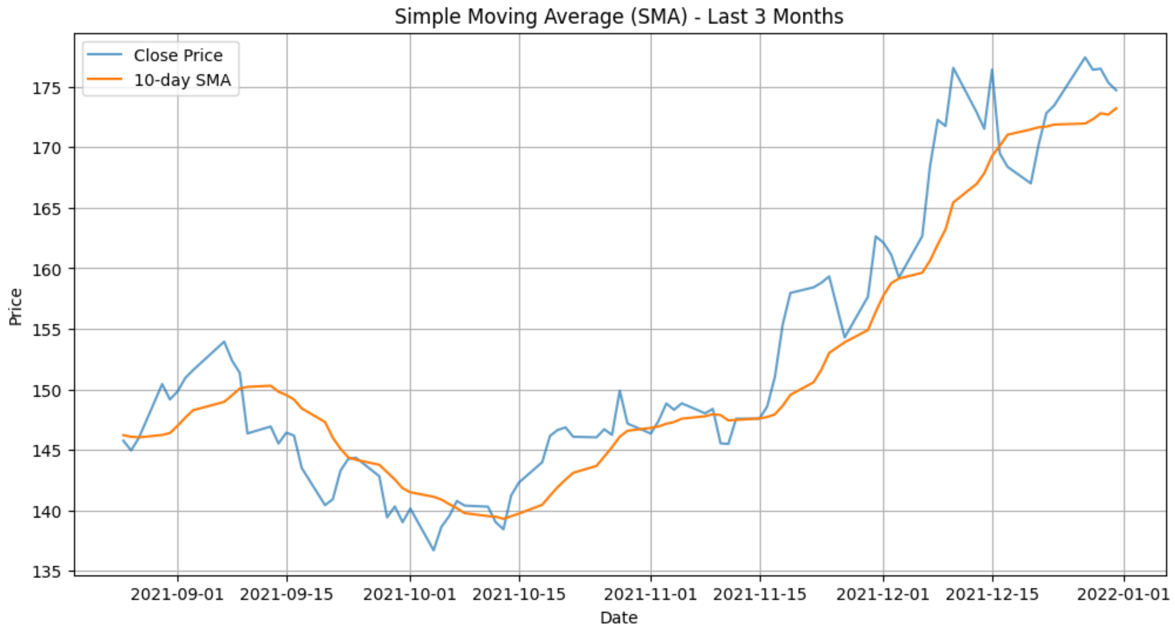
- Technical indicators are mathematical calculations based on stock prices, volume, or other market data.
- Used to identify trends, momentum, volatility, and price behavior.
- Aid in identifying buy/sell opportunities and understanding market sentiment.

Some Indicators:

Simple Moving Average (**SMA**)

$$\mathrm{SMA}_n = \frac{\sum_{i=1}^{n} \mathrm{Price}_i}{n}$$

- Average of closing prices over a specified period (e.g., 5 days).
- The SMA (**orange line**) smooths out short-term fluctuations, providing a clearer view of the overall trend.



Technical Indicators Overview

Relative Strength Index (**RSI**)

$$\mathrm{RSI} = 100 - \frac{100}{1 + \frac{\text{Average Gain}}{\text{Average Loss}}}$$

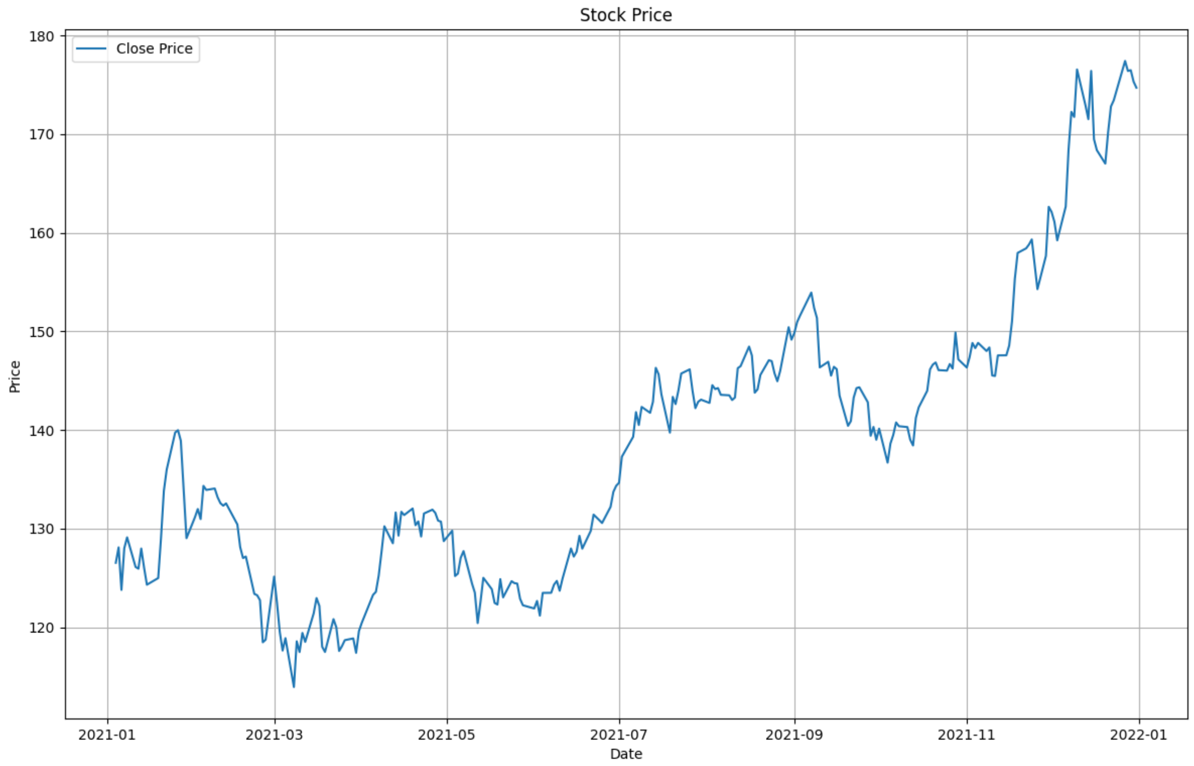
RSI measures momentum by comparing the magnitude of recent price gains to losses.

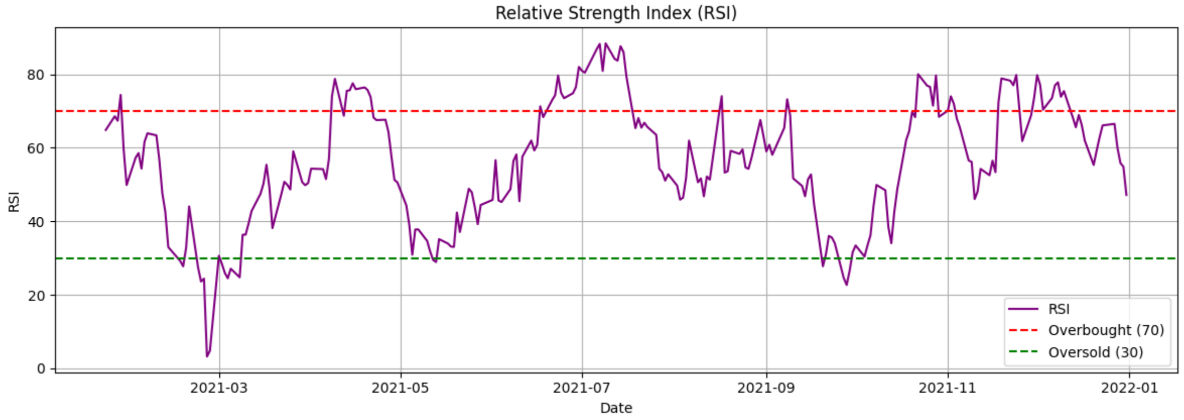
It quantifies overbought (high price momentum) or oversold (low price momentum) conditions.

- **RSI > 70**: **Overbought** (price may decline). Reflects excessive gains compared to losses.
- **RSI < 30**: **Oversold** (price may rise). Reflects excessive losses compared to gains.

Some other technical Indicators:

- MACD - identifies both momentum and trend direction by comparing two moving averages.
- ADX - quantifies the strength of a trend, distinguishing between trending and range-bound markets.
- Bollinger Bands, ATR, STD10, etc.





Dimensionality Reduction

Why Dimensionality Reduction?

- The "**curse of dimensionality**" increases computational cost and complexity.
- Redundant or highly correlated features reduce model efficiency.

Benefits:

- Simplifies feature space while retaining key patterns.
- Removes noise and reduces overfitting.
- Improves computational efficiency for predictive models.

Principal Component Analysis (PCA) – Overview

What is PCA?

- A statistical technique.
- Transforms high-dimensional data into lower dimensions.
- Preserves maximum variance.
- Creates uncorrelated features from correlated input variables.

Financial Market Point-of-view (POV):

- Original Features (Often **Highly Correlated**):
  - Moving averages (MA50 and SMA5) tend to move together.
  - Bollinger Bands (Upper and Lower) are derived from the same price data.
  - RSI and MACD both measure momentum.
- PCA Transformation:
  - Converts these correlated features into independent components.
  - Each component captures a unique aspect of market movement.
  - No redundant information between components.

Mathematical Explanation of PCA

Step 1: Standardize the data

- PCA works best when the data is standardized to have a mean of 0 and unit variance.

Mean and standard deviation: $$Z_{ij} = \frac{X_{ij} - \mu_j}{\sigma_j}$$

$$\mu_1 = \frac{2+0+1+4}{4} = 1.75, \quad \mu_2 = \frac{4+3+2+7}{4} = 4$$

$$\sigma_1 = \sqrt{\frac{(2-1.75)^2 + (0-1.75)^2 + (1-1.75)^2 + (4-1.75)^2}{4}} = 1.479$$

$$\sigma_2 = \sqrt{\frac{(4-4)^2 + (3-4)^2 + (2-4)^2 + (7-4)^2}{4}} = 1.825$$

Standardized Data:

$$Z_{11} = \frac{2-1.75}{1.479} = 0.169, \quad Z_{12} = \frac{4-4}{1.825} = 0$$

Sample Dataset	
Feature 1 (X1)	Feature 2 (X2)
2	4
0	3
1	2
4	7

Standardized Dataset	
Z1	Z2
0.169	0
-1.183	-0.548
-0.507	-1.096
1.521	1.644

Mathematical Explanation of PCA

Step 2: Covariance Matrix Computation

The covariance matrix captures the relationship between features.

$$C = \frac{1}{n-1} Z^\top Z$$

Using the standardized data *Z*,

$$C = \frac{1}{3} \begin{bmatrix} 0.169 & -1.183 & -0.507 & 1.521 \\ 0 & -0.548 & -1.096 & 1.644 \end{bmatrix} \begin{bmatrix} 0.169 & 0 \\ -1.183 & -0.548 \\ -0.507 & -1.096 \\ 1.521 & 1.644 \end{bmatrix}$$

Covariance matrix:

$$C = \begin{bmatrix} 1 & 0.989 \\ 0.989 & 1 \end{bmatrix}$$

Standardized Dataset	
Z1	Z2
0.169	0
-1.183	-0.548
-0.507	-1.096
1.521	1.644

Mathematical Explanation of PCA

Step 3: Eigenvalues and Eigenvectors

Eigenvalue equation to find the eigenvalues (λ) and eigenvectors (v):

$$C \cdot v = \lambda v$$

Eigenvalues,

$$\text{Solve: } \det(C - \lambda I) = 0$$

$$\det \begin{bmatrix} 1-\lambda & 0.989 \\ 0.989 & 1-\lambda \end{bmatrix} = 0$$

$$(1-\lambda)^2 - (0.989)^2 = 0$$

$$\lambda_1 = 1.989, \quad \lambda_2 = 0.011$$

Eigenvectors,

For $\lambda_1 = 1.989$:

$$\begin{bmatrix} 1-1.989 & 0.989 \\ 0.989 & 1-1.989 \end{bmatrix} \begin{bmatrix} v_{11} \\ v_{21} \end{bmatrix} = 0$$

Solve:

$$\begin{bmatrix} -0.989 & 0.989 \\ 0.989 & -0.989 \end{bmatrix} \begin{bmatrix} v_{11} \\ v_{21} \end{bmatrix} = 0$$

$$v_1 = \begin{bmatrix} 0.707 \\ 0.707 \end{bmatrix}$$

For $\lambda_2 = 0.011$:

$$v_2 = \begin{bmatrix} -0.707 \\ 0.707 \end{bmatrix}$$

Standardized Dataset	
Z1	Z2
0.169	0
-1.183	-0.548
-0.507	-1.096
1.521	1.644

Mathematical Explanation of PCA

Step 4: Project Data onto Principal Components

$$Z_{PCA} = Z \cdot V$$

Where $V = \begin{bmatrix} 0.707 & -0.707 \\ 0.707 & 0.707 \end{bmatrix}$.

$$Z_{PCA} = \begin{bmatrix} 0.169 & 0 \\ -1.183 & -0.548 \\ -0.507 & -1.096 \\ 1.521 & 1.644 \end{bmatrix} \cdot \begin{bmatrix} 0.707 & -0.707 \\ 0.707 & 0.707 \end{bmatrix}$$

Resulting PCA transformed data,

$$Z_{PCA} = \begin{bmatrix} 0.120 & -0.120 \\ -1.223 & 0.450 \\ -1.132 & -0.416 \\ 2.235 & 0.086 \end{bmatrix}$$

Step 5: Explained Variance

To understand how much information is retained by each principal component.

$$\frac{\lambda_i}{\sum \lambda}$$

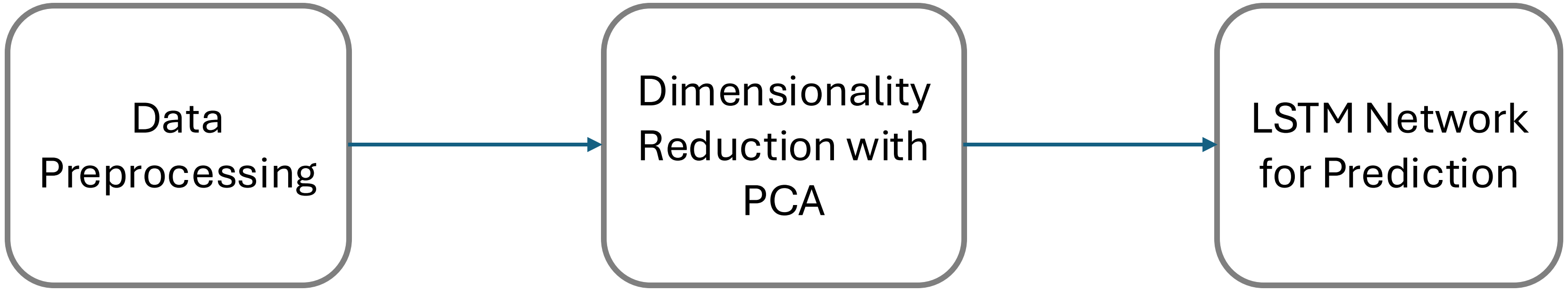
Variance ratio:

$$\lambda_1 = \frac{1.989}{1.989 + 0.011} = 0.9945$$

$$\lambda_2 = \frac{0.011}{1.989 + 0.011} = 0.0055$$

PCA in Stock Market Data

Workflow:



PCA in Stock Market Data

Data Preprocessing:

- Downloaded AAPL (Apple) stock data.
- Generated the technical indicators / features.
- Features: *Close, MA50, SMA5, RSI, MACD, Bollinger_Upper, Bollinger_Lower, ATR, ADX, STD10.*

```python
data = yf.download(stocks, start=start_date, end=end_date)
data = data[['Open', 'High', 'Low', 'Close', 'Volume']]
data.dropna(inplace=True)
data['Date'] = data.index

# technical indicators
data['Return'] = data['Close'].pct_change()
data['MA10'] = data['Close'].rolling(window=10).mean()
data['MA50'] = data['Close'].rolling(window=50).mean()
data['STD10'] = data['Close'].rolling(window=10).std()
data['MACD'] = data['Close'].ewm(span=12, adjust=False).mean()
- data['Close'].ewm(span=26, adjust=False).mean()
data['Bollinger_Upper'] = data['MA10'] + (data['STD10'] * 2)
data['Bollinger_Lower'] = data['MA10'] - (data['STD10'] * 2)
data['SMA5'] = data['Close'].rolling(window=5).mean()
```

```
Price              Close       MA50       SMA5        RSI       MACD  \
Ticker              AAPL
Date
2015-03-16  27.935472  26.707244  27.711452  32.710678  28.084503
2015-03-17  28.402739  26.788356  27.824579  45.533337  28.133462
2015-03-18  28.722445  26.889579  28.103150  44.971542  28.224075
2015-03-19  28.505587  26.986421  28.239530  47.391226  28.267384
2015-03-20  28.147873  27.069472  28.342823  41.765643  28.248998

Price      Bollinger_Upper Bollinger_Lower       ATR        ADX      STD10
Ticker
Date
2015-03-16       29.117901       27.123730  0.640696  25.134597  0.498543
2015-03-17       28.924903       27.212989  0.614347  23.646014  0.427979
2015-03-18       28.917936       27.216824  0.590872  23.031070  0.425278
2015-03-19       28.980195       27.203304  0.583206  22.495736  0.444223
2015-03-20       28.953343       27.198857  0.603328  20.902348  0.438621
```

PCA in Stock Market Data

PCA implementation:

- **Data Scaling:** Data is standardized using *StandardScaler* to normalize features for PCA.
- **PCA Application:** Reduced dimensionality to 7 components, retaining maximum variance.
- **Feature Transformation:** Original data transformed into principal components (PC1, PC2, ..., PC7).
- **Output:** Components are aligned with the original closing prices (Close) and dates.
- **Key Takeaway:**
  PCA simplifies high-dimensional financial data while retaining essential trends for analysis.

```python
scaler = StandardScaler()
data_scaled = scaler.fit_transform(data[features])

#  PCA to reduce dimensionality
pca = PCA(n_components=7)  # Retain more components
data_pca = pca.fit_transform(data_scaled)

data_pca_df = pd.DataFrame(data_pca, columns=[f'PC{i+1}' for i in range(7)])
data_pca_df['Close'] = data['Close'].values
data_pca_df['Date'] = data['Date'].values
```

```
        PC1       PC2       PC3       PC4       PC5       PC6       PC7  \
0 -2.175074 -1.009476  0.128279 -0.899189  0.313576  0.066143  0.018540
1 -2.208045 -0.593808 -0.201109 -0.400904  0.202419  0.078801  0.012462
2 -2.203090 -0.621304 -0.252047 -0.435824  0.218347  0.069745  0.018200
3 -2.201773 -0.551668 -0.332036 -0.336009  0.218184  0.062613  0.011423
4 -2.179367 -0.819916 -0.381346 -0.544315  0.288886  0.075811  0.012149

       Close        Date
0  27.935472  2015-03-16
1  28.402739  2015-03-17
2  28.722445  2015-03-18
3  28.505587  2015-03-19
4  28.147873  2015-03-20
```

PCA in Stock Market Data

- **LSTM Model Overview:**
- **Input Shape:** Sequential input shaped to PCA-transformed data dimensions.
- **LSTM Layers:**
  - Three layers with 128, 64, and 32 units, regularized using L2 for improved generalization.
  - return_sequences=True in the first two layers to maintain temporal information.
- **Normalization and Dropout:** Batch normalization and dropout (0.2) used to prevent overfitting.
- **Dense Layers:**
  - Final dense layers include one hidden layer with 16 units and ReLU activation.
  - Output layer predicts stock price as a single value.
- **Key Takeaway:**
  This LSTM architecture leverages PCA-transformed data for efficient stock price prediction.

```
model = Sequential([
    Input(shape=(X_train.shape[1], X_train.shape[2])),
    LSTM(128, return_sequences=True, kernel_regularizer=tf.keras.regularizers.l2(0.001)),
    BatchNormalization(),
    Dropout(0.2),
    LSTM(64, return_sequences=True, kernel_regularizer=tf.keras.regularizers.l2(0.001)),
    BatchNormalization(),
    Dropout(0.2),
    LSTM(32, return_sequences=False, kernel_regularizer=tf.keras.regularizers.l2(0.001)),
    BatchNormalization(),
    Dropout(0.2),
    Dense(16, activation='relu', kernel_regularizer=tf.keras.regularizers.l2(0.001)),
    Dense(1)
])
```

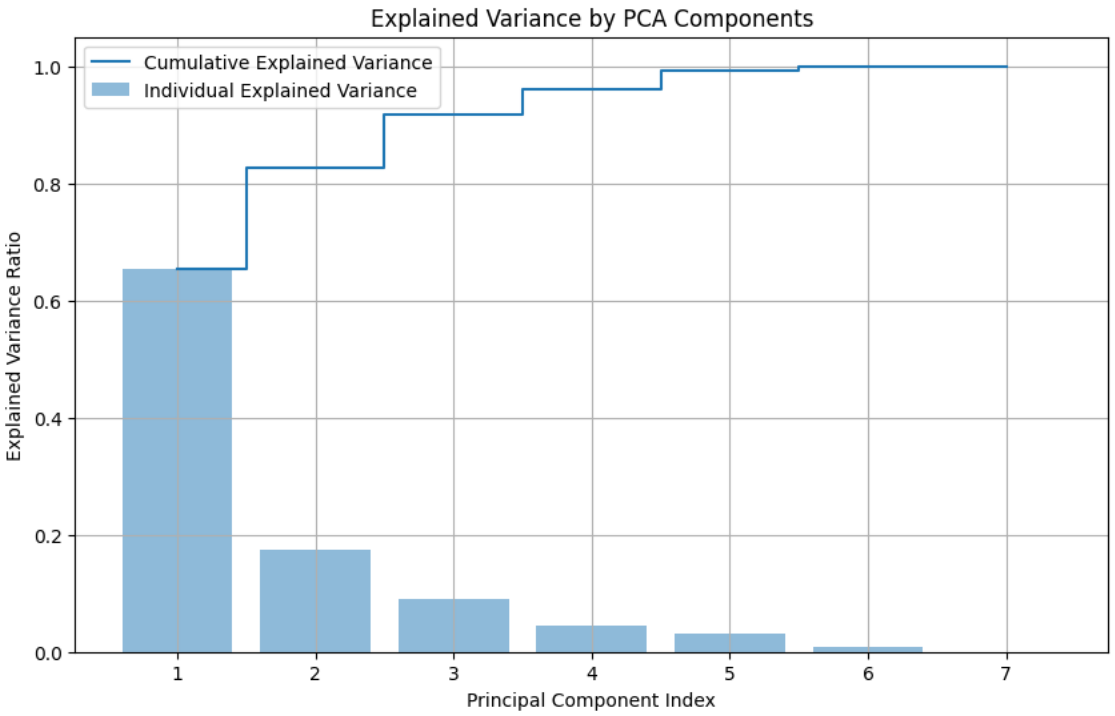
Result and Analysis

First Component (PC1):
- Accounts for about **65.39%** of the total variance.
- Most important and dominant price trend component.

Second Component (PC2):
- Adds approximately **17.38%** of explained variance.
- Together with PC1, explains about **82%** of total variance.

Cumulative Variance (Blue Line):
- Shows the running total of explained variance.
- Reaches nearly **100%** by the 7th component.
- About 90% of variance is explained by first 3 components.

Key Insights:
- Most of the important information is captured in the first 4 components.
- Can reduce dimensionality to 4 components while retaining **~96%** of the information.



Components	Explained Variance
PC1	0.6539 (65.39%)
PC2	0.1738 (17.38%)
PC3	0.0892 (8.92%)
PC4	0.0438 (4.38%)
PC5	0.0312 (3.12%)
PC6	0.0079 (0.79%)
PC7	0.0001 (0.01%)

Result and Analysis - Feature Contribution Analysis in PCA Components

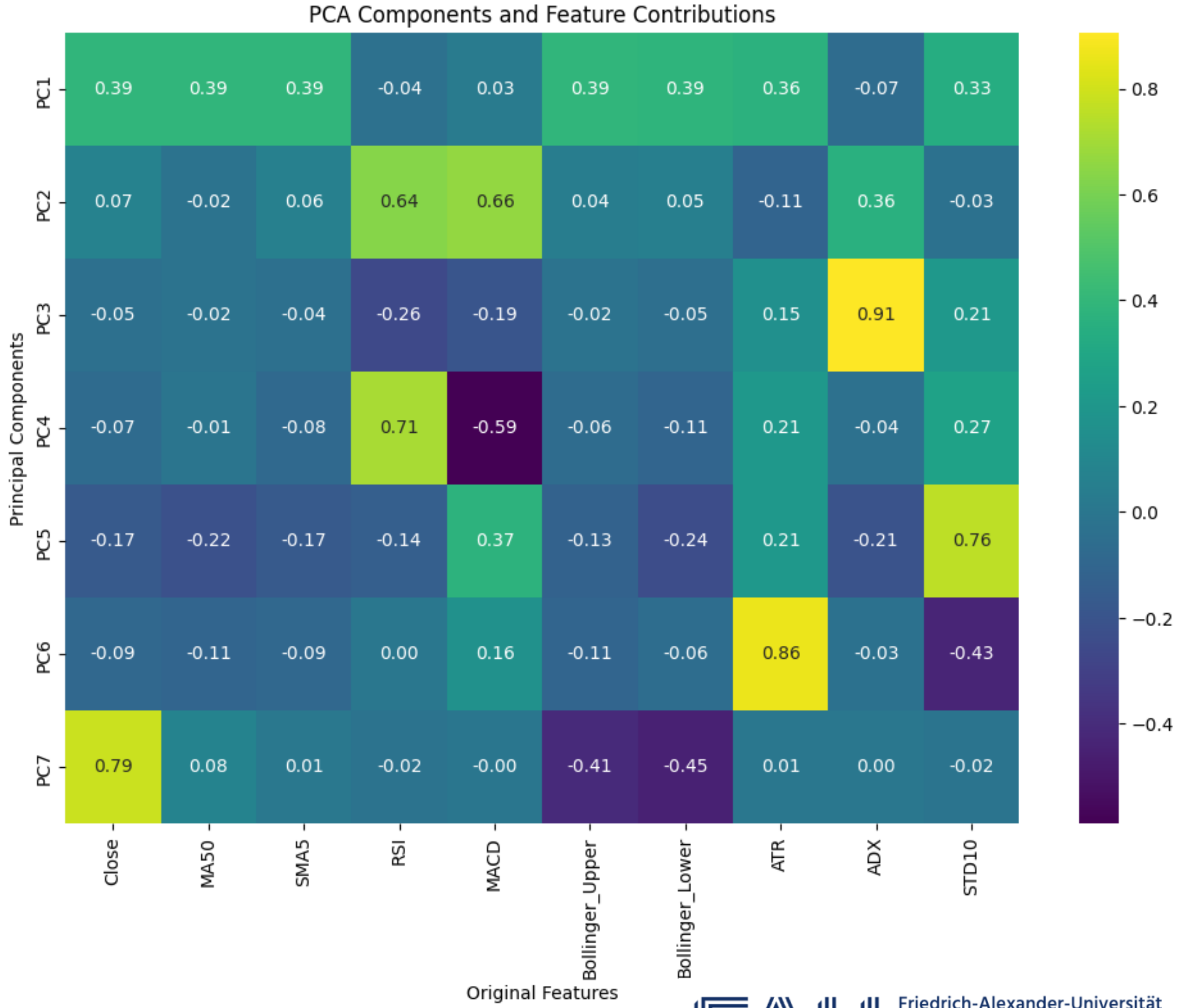
PC1 (+0.39 for price indicators)
- All price indicators (Close, MA50, SMA5, Bollinger Bands) move in harmony.
- Captures **primary** market trends, bull/bear market phases.

PC2 (RSI: +0.64, MACD: +0.66)
- **Momentum** indicators align with market direction.
- RSI confirms price momentum strength.
- MACD validates trend continuation.

PC3 (ADX +0.91)
- Strong ADX correlation indicates clear **trend strength**.
- Critical for trend quality assessment.

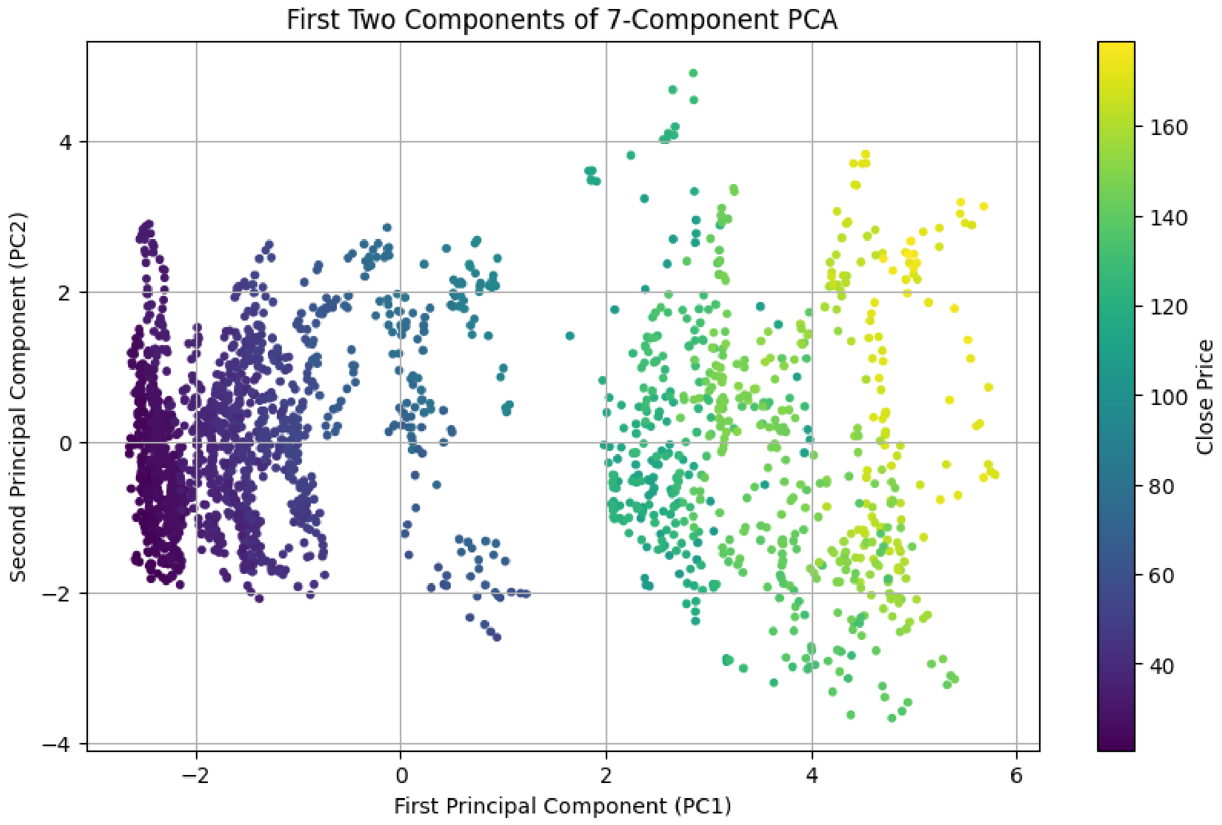
PC4 (RSI: +0.71 vs MACD: -0.59)
- **Divergence** between momentum indicators.
- Potential **trend reversal** signals.
- Early warning for trend changes.
- Market psychology shift indicator.

Positive Correlation **Negative Correlation**



Friedrich-Alexander-Universität Erlangen-Nürnberg

Result and Analysis - First Two Component Scatter Plot



- Each dot represents a **trading day** with:
  - **X**-axis: First Principal Component (PC1, 65.39% variance).
  - **Y**-axis: Second Principal Component (PC2, 17.38% variance).
  - Color: Closing price value (from ~$20 to $180).

- **Three** market regimes or **cluster.**

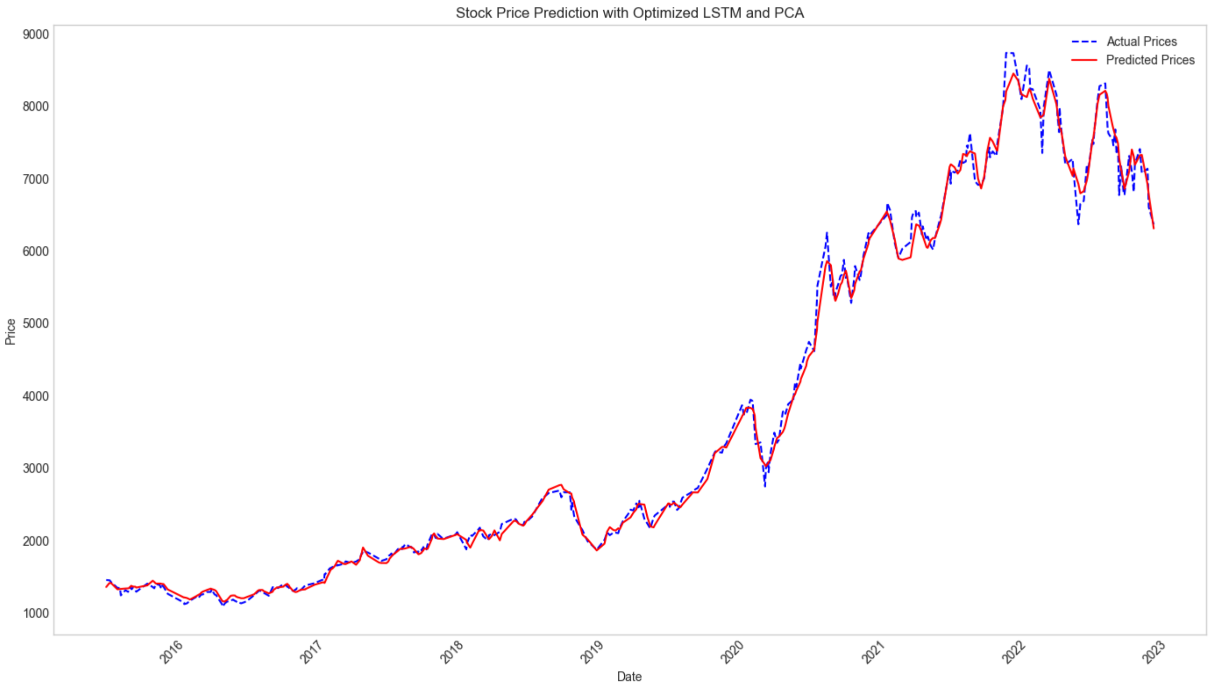
- **Left Cluster** (Purple):
  - **Bear** market.
  - Tight grouping indicates low volatility.

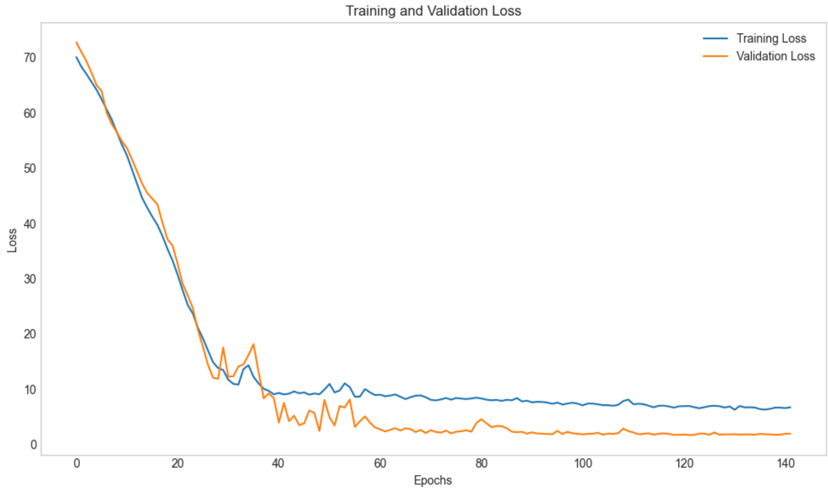
- **Middle Cluster** (Green):
  - **Normal** trading range.
  - Wider vertical spread indicates higher volatility.

- **Right Cluster** (Yellow):
  - **Bull** market conditions.
  - Wider vertical spread indicates higher volatility.

Friedrich-Alexander-Universität
Erlangen-Nürnberg

Result and Analysis – Price Prediction with PCA and LSTM model

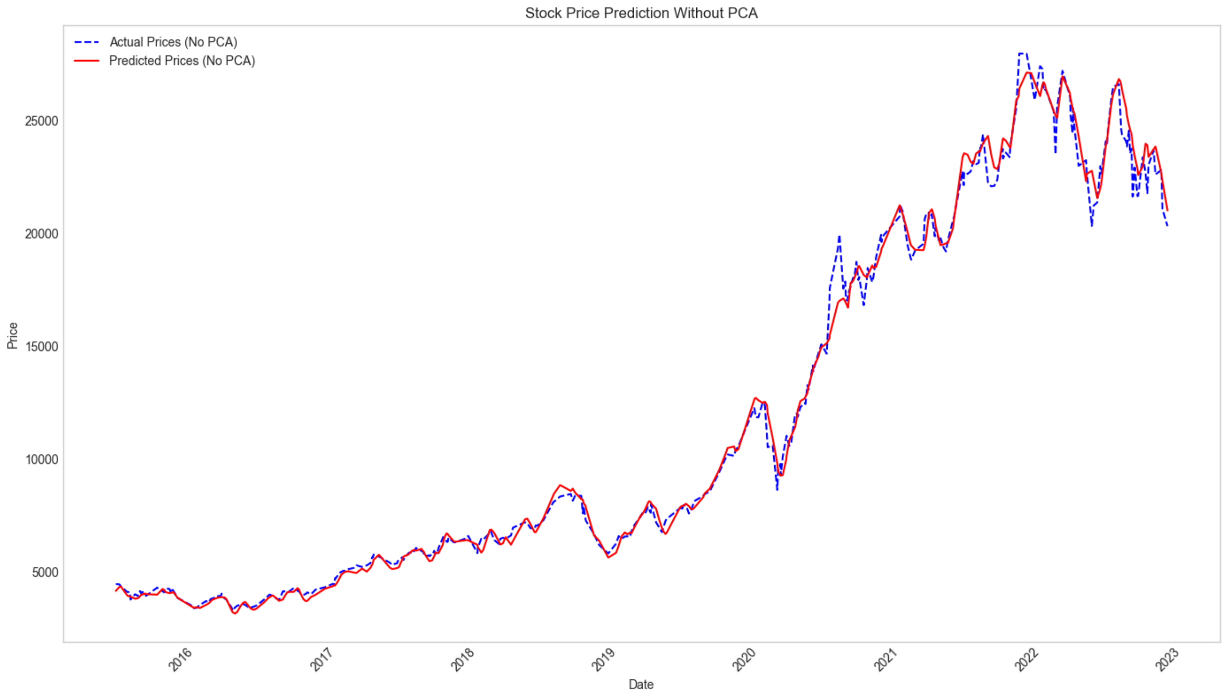


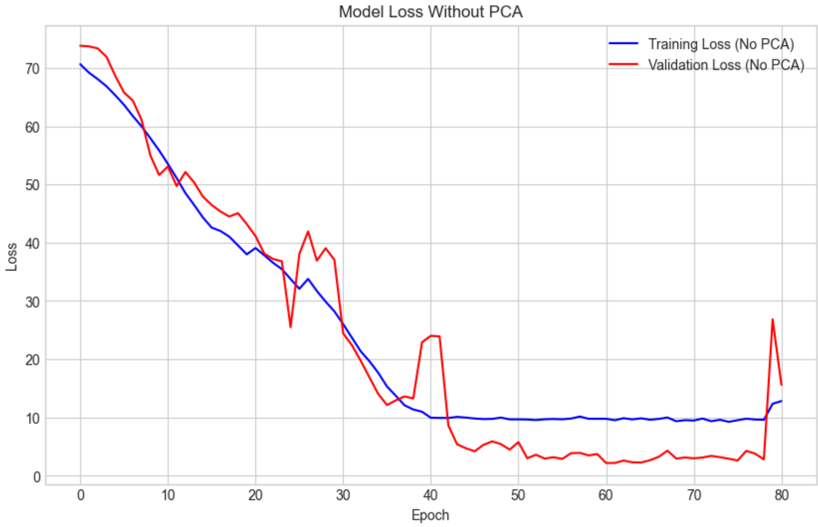


- **Training loss** reduces steadily over epochs and stabilizes with minimal fluctuation after ~40 epochs, indicating a well-fitted model.
- **Validation loss** is slightly lower than training loss in later epochs, reflecting good generalization without overfitting.
- The **predicted prices** closely align with actual prices across the time frame, reflecting PCA's efficiency in retaining essential features.

Friedrich-Alexander-Universität
Erlangen-Nürnberg

Result and Analysis – Price Prediction without PCA and LSTM model



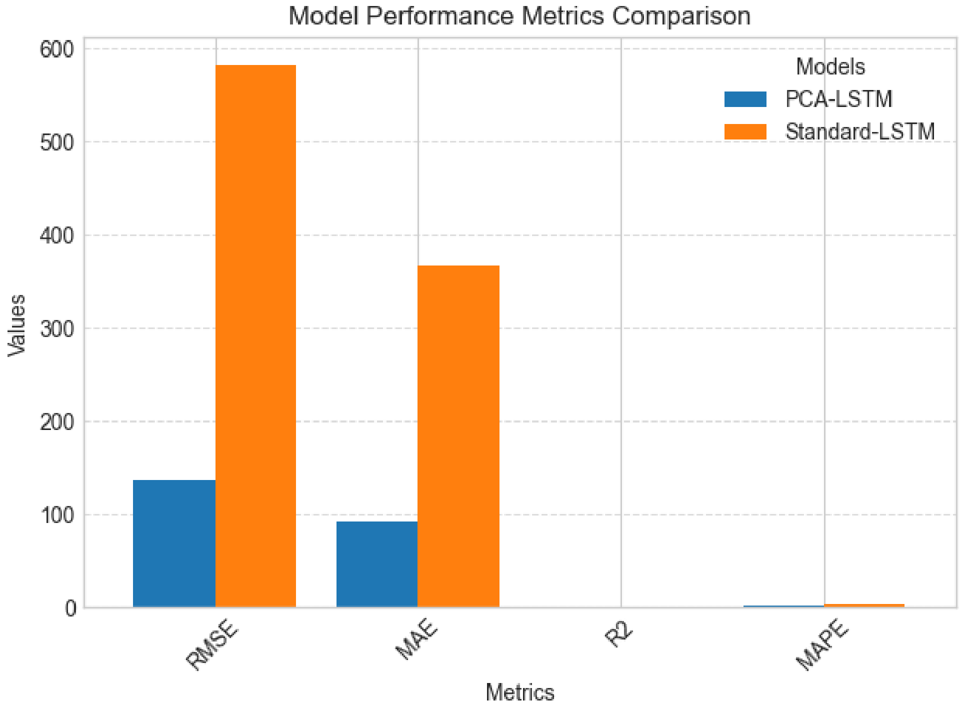


- The **validation loss** shows significant fluctuations, especially after ~40 epochs, indicating overfitting or noise influence in the model.
- The **training loss** stabilizes at a higher value compared to PCA-LSTM, showing less effective learning.
- **Predicted prices** deviate from actual values at several points, reflecting challenges in handling high-dimensional and redundant features.

Friedrich-Alexander-Universität
Erlangen-Nürnberg

Result and Analysis – Performance Analysis

Metric	PCA-LSTM	Standard-LSTM	Insights
RMSE	136.78	582.33	PCA significantly reduces prediction errors, achieving **76.5% lower** RMSE compared to Standard-LSTM.
MAE	91.37	366.33	PCA-LSTM is more precise in predicting stock prices with an **MAE reduction of 75%.**
R^2 Score	0.9969	0.9946	Both models demonstrate **high accuracy**, but PCA-LSTM slightly outperforms, indicating better fit to the data.
MAPE	2.60%	3.18%	PCA **reduces prediction variability**, making predictions more reliable.
Memory Usage	4.88 MB	6.98 MB	PCA reduces memory usage by approximately **30%**, making it more efficient for large datasets.



- RMSE: Root Mean Squared Error
- MAE: Mean Absolute Error
- R^2 Score: Coefficient of Determination
- MAPE: Mean Absolute Percentage Error

Result and Analysis – Trading Performance Analysis

Total Return vs. Risk:

• Standard LSTM achieves higher total returns but with greater risk.

• PCA-LSTM offers lower returns but significantly reduces risk, ideal for conservative strategies.

Loss Management:

• PCA-LSTM effectively manages losses:

  • Average loss (~ 74.91) **~ 4x smaller** than Standard LSTM (~285.72)

  • Max loss is ~4 times lower for PCA-LSTM (-596 vs. -2,404).

Win Rate and Loss Rate:

• Higher win rate (63.16%) and lower loss rate (36.32%) in PCA-LSTM indicate more stable and profitable trades.

Risk-Adjusted Performance:

• With a Sharpe Ratio of 5.31 vs. 3.59, PCA-LSTM better balances returns and risk.

Max Drawdown Comparison:

• PCA-LSTM's drawdown is ~7x smaller (-596 vs. -4,204), demonstrating greater resilience against adverse market conditions

Metric	PCA-LSTM	Standard-LSTM
Total Return	22,205.28	49,789.27
Sharpe Ratio	5.31	3.59
Max Drawdown	-596.19	-4,204.81
Win Rate	63.16%	58.68%
Average Loss	-74.91	-285.72
Max Loss	-596.19	-2,404.12
Loss Rate	36.32%	40.79%

Conclusion and Future Directions

- **Dimensionality Reduction Excellence:** PCA effectively reduces high-dimensional financial datasets to capture essential patterns while discarding noise and redundancy.
- **Improved Model Efficiency:** PCA reduces computational cost while maintaining or improving predictive accuracy.
- **Risk Management Advantage:** PCA-LSTM demonstrates superior loss management and reduced drawdowns, showcasing PCA's value in risk-aware trading strategies.

Future Directions:

- **Expanding Use Cases:** Apply PCA-driven models to different asset classes (commodities, cryptocurrencies) or market indices.
- **Integration of External Factors:** Incorporate macroeconomic indicators (interest rates, inflation), sentiment analysis into the PCA framework for broader insights.
- **Real-Time Analysis:** Optimize the workflow for minute-level data, leveraging PCA for faster computations in live trading scenarios.

THANK YOU