Task 5: Machine Learning 2

Muhammad Ali, 103960437, COS30018, 22/09/2024, Tutor: Ru Jia.

Requirements / Deliverables –

- Implement a function that solves the multistep prediction problem to allow the prediction to be made for a sequence of closing prices of k days into the future.
- Implement a function that solve the simple multivariate prediction problem that considers the other features for the same company (including opening price, highest price, lowest price, closing price, adjusted closing price, trading volume) as the input for predicting the closing price of the company for a specified day in the future.
- Combine the above two functions to solve the multivariate, multistep prediction problem.

Result –

Multivariate, multistep prediction (combined) function –

```
def multivariate_prediction(k_days: int, company: str, train_start: str,
train_end: str, prediction_days: int,
                            n_features: int = 6, test_ratio: float = 0.15,
multivariate: bool = True):
```

- **K_days (int)** –
  - Specifies the number of future days to predict.
  - If k_days = 5, the model will predict the closing prices for the next 5 days.
- **Company (str)** –
  - Represents the ticker symbol of the stock to be analysed.
- **Train_start(str)** –
  - The start date of the training date in ‘YYYY-MM-DD’ format.
- **Train_end (str)** –
  - The end date of the training data in ‘YYYY-MM-DD’ format.
- **Prediction_days (int)** –
  - Defines the number of past days to use as input for the model when making predictions.

- **N_features (int) –**
    - Specifies the number of features to be used as input for the model. For instance '6' would include all features like opening price, highest price, lowest price, closing price, adjusted closing price, and trading volume.
- **Test_ratio (float) –**
    - Defines the ratio of the dataset reserved for testing. For e.g. If *'test_ratio = 0.15'* then 15% of the data us used as the test set.
- **Multivariate (bool) –**
    - Indicates whether to use multivariate inputs for prediction.
    - If set to *'true',* multiple features are considered in the prediction process; if '*false'*, only the closing price is used as input.

Multistep part of the function –

```
def multivariate_prediction(k_days: int, prediction_days:int)
```

k_days – Specifies the number of future days for which the model predicts the closing prices.

Prediction_days – defines the number of previous days used as input for the model.

Data Preparation –

The function uses '*prepare_multistep_data*' from data_processing.py to reshape the data to handle multistep predictions. This function creates input-output sequences where each sample contains data for the *past 'prediction_days'*, and the output includes predictions for '*k_days'* ahead.

```
def prepare_multistep_data(X, y, timesteps, future_steps):
    """
    Prepares data for time series models for multistep predictions.

    Parameters:
    - X (np.ndarray): Input feature matrix.
    - y (np.ndarray): Target vector.
    - timesteps (int): Number of timesteps for each sample.
```

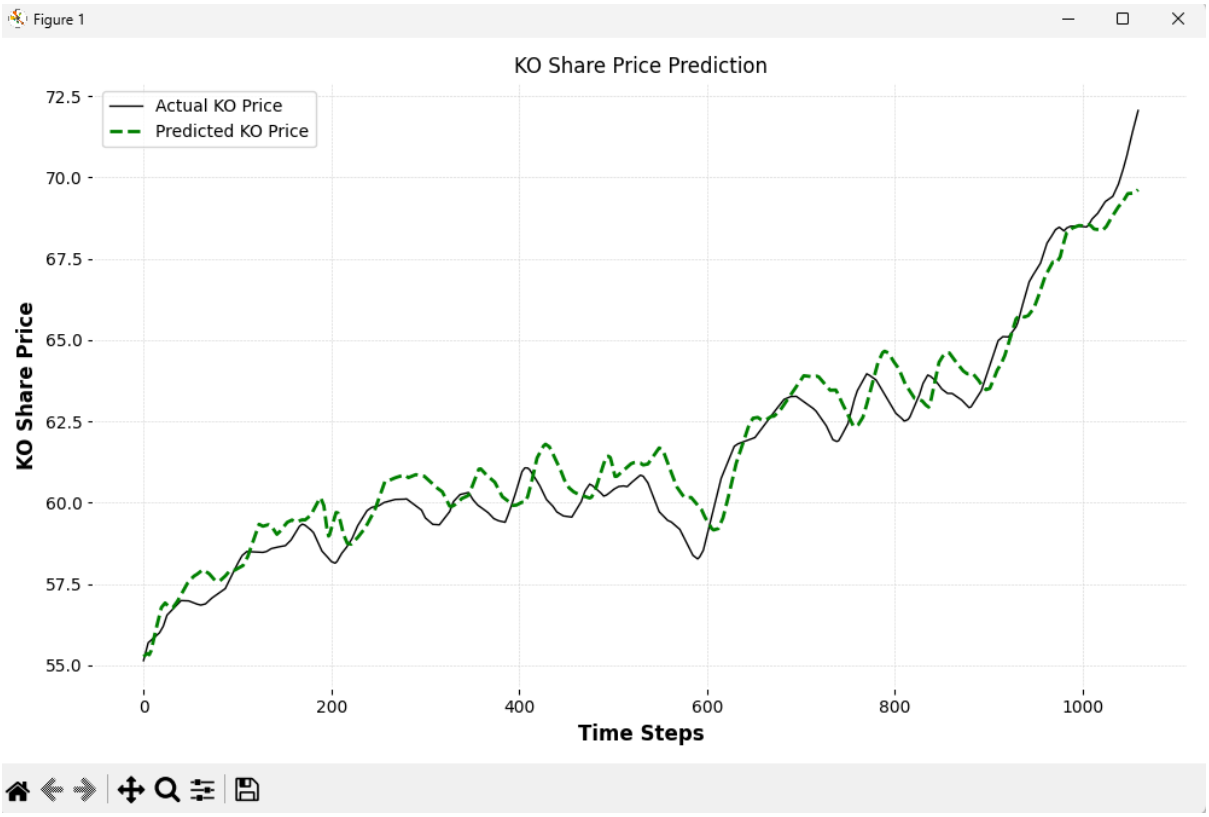
```
    - future_steps (int): Number of future steps to predict.

    Returns:
    - X_reshaped (np.ndarray): Reshaped input data for the model.
    - y_reshaped (np.ndarray): Corresponding reshaped target data
(multistep).
    """
    n_samples = X.shape[0] - timesteps - future_steps + 1

    X_reshaped = np.array([X[i:i + timesteps] for i in range(n_samples)])
    y_reshaped = np.array([y[i + timesteps:i + timesteps + future_steps]
for i in range(n_samples)])

    return X_reshaped, y_reshaped
```

Prediction –

After training, the model predicts on the test data, providing a sequence of predictions for '*k_days*' in the future.

```
# Predict on the test set
predicted_prices = model.predict(X_test_scaled)
predicted_prices_inv =
y_scaler.inverse_transform(predicted_prices.reshape(-1, 1)).reshape(-1,
k_days)
```

Visualisation –



Summary –

Multistep Prediction: The code solves the problem by creating sequences of inputs and outputs that allow predicting a sequence of k future days' closing prices.

Multivariate Prediction: By incorporating multiple features (open, high, low, volume), the function predicts the closing price of the company for a specified day in the future, leveraging broader information.