Ai Rand Forest and Decision Tree

August 3, 2025

[1]:
```
#ECGR-4105 Random Forest and Decision Tree Stock Predictor
#Steve Revens
#Harrison Hall

import yfinance as yf
import pandas as pd
import numpy as np
import os
import time
from datetime import datetime
from sklearn.ensemble import RandomForestRegressor
from sklearn.tree import DecisionTreeRegressor
from sklearn.metrics import mean_squared_error, r2_score
from sklearn.model_selection import train_test_split
from sklearn.preprocessing import LabelEncoder
import matplotlib.pyplot as plt
import seaborn as sns

# Debugging help
NUM_COMPANIES = 500                    # Set to 500 for full run
FORCE_REDOWNLOAD = True                # Set to False if the spreadsheets are␣
 ↪downloaded and current, stops the program from re-downloading the␣
 ↪spreadsheets
SAVE_DIR = "data"
FUNDAMENTAL_DIR = "fundamentals"

# Get S&P tickers
def get_sp500_tickers():
    url = "https://datahub.io/core/s-and-p-500-companies/r/constituents.csv"
    df = pd.read_csv(url)
    return df['Symbol'].str.replace('.', '-', regex=False).tolist()

# Download the data
def download_stock_data(tickers, start="2000-01-01", end=None,␣
 ↪save_dir=SAVE_DIR, force_redownload=False):
    if end is None:
        end = datetime.today().strftime("%Y-%m-%d")
```

```
    os.makedirs(save_dir, exist_ok=True)
    os.makedirs(FUNDAMENTAL_DIR, exist_ok=True)
    data = {}

    for i, ticker in enumerate(tickers):
        file_path = os.path.join(save_dir, f"{ticker}.csv")
        fundamentals_path = os.path.join(FUNDAMENTAL_DIR,␣
↪f"{ticker}_fundamentals.csv")

        try:
            stock = yf.Ticker(ticker)
            if os.path.exists(file_path) and not force_redownload:
                print(f"Reading cached data for {ticker}")
                df = pd.read_csv(file_path)
                if 'Date' not in df.columns or 'Close' not in df.columns:
                    continue
                df['Date'] = pd.to_datetime(df['Date'])
                df.set_index('Date', inplace=True)
            else:
                print(f"Downloading {ticker} ({i+1}/{len(tickers)})")
                df = stock.history(start=start, end=end, auto_adjust=False)
                if df.empty or len(df) < 200:
                    continue
                df.reset_index(inplace=True)
                df.to_csv(file_path, index=False)
                df.set_index('Date', inplace=True)

            info = stock.info
            new_row = {
                'Date': datetime.today().strftime("%Y-%m-%d"),
                'Ticker': ticker,
                'peRatio': info.get('trailingPE', np.nan),
                'pegRatio': info.get('pegRatio', np.nan),
                'priceToSales': info.get('priceToSalesTrailing12Months', np.
↪nan),
                'priceToBook': info.get('priceToBook', np.nan),
                'debtToEquity': info.get('debtToEquity', np.nan),
                'ebitda': info.get('ebitda', np.nan)
            }

            if os.path.exists(fundamentals_path):
                existing = pd.read_csv(fundamentals_path)
                if not ((existing['Date'] == new_row['Date']) &␣
↪(existing['Ticker'] == new_row['Ticker'])).any():
                    updated = pd.concat([existing, pd.DataFrame([new_row])],␣
↪ignore_index=True)
                    updated.to_csv(fundamentals_path, index=False)
```

```
            else:
                pd.DataFrame([new_row]).to_csv(fundamentals_path, index=False)

            df['Ticker'] = ticker
            data[ticker] = df
            time.sleep(0.2)

        except Exception as e:
            print(f"Error with {ticker}: {e}")

    return data

# Preprocess for training
def preprocess_data(data):
    frames = []
    for ticker, df in data.items():
        try:
            if 'Close' not in df.columns:
                raise ValueError(f"Missing Close column, shape: {df.shape}")

            df = df[['Open', 'High', 'Low', 'Volume', 'Close']].copy()
            df['Lag_Close'] = df['Close'].shift(1)
            df['Lag_1'] = df['Close'].shift(1)
            df['MA_5'] = df['Close'].rolling(5).mean()
            df['Return'] = df['Close'].pct_change()
            df['Target'] = df['Close'].shift(-5)
            df.dropna(inplace=True)

            if len(df) < 10:
                continue

            fund_file = os.path.join(FUNDAMENTAL_DIR, f"{ticker}_fundamentals.
 ↪csv")
            if not os.path.exists(fund_file):
                continue

            row = pd.read_csv(fund_file).sort_values('Date').iloc[-1]
            for col in ['peRatio', 'pegRatio', 'priceToSales', 'priceToBook',␣
 ↪'debtToEquity', 'ebitda']:
                df[col] = row.get(col, np.nan)

            df['Ticker'] = ticker
            frames.append(df)

        except Exception as e:
            print(f"Preprocessing error on {ticker}: {e}")
```

```
    if not frames:
        raise ValueError("All tickers failed during preprocessing.")

    result = pd.concat(frames)
    result.reset_index(inplace=True)
    return result

# Extract features and labels
def prepare_features(df):
    y = df['Target']
    X = df.drop(columns=['Target'])
    non_numeric_cols = X.select_dtypes(include=['object', 'datetime64']).
 ↪columns.tolist()
    for col in ['Date', 'Datetime']:
        if col in X.columns:
            non_numeric_cols.append(col)
    X.drop(columns=list(set(non_numeric_cols)), inplace=True)
    le = LabelEncoder()
    if 'Ticker' in df.columns:
        X['Ticker'] = le.fit_transform(df['Ticker'])
    return X, y

# Train and evaluate
def train_and_evaluate(X, y, model_name="RandomForest"):
    X_train, X_test, y_train, y_test = train_test_split(X, y, shuffle=False,␣
 ↪test_size=0.2)
    if model_name == "RandomForest":
        model = RandomForestRegressor(n_estimators=100, random_state=1)
    elif model_name == "DecisionTree":
        model = DecisionTreeRegressor(random_state=1)
    else:
        raise ValueError("Unknown model")
    model.fit(X_train, y_train)
    preds = model.predict(X_test)
    rmse = mean_squared_error(y_test, preds, squared=False)
    r2 = r2_score(y_test, preds)
    return model, X_test.index, y_test, preds, rmse, r2

# Plot predictions
def plot_predictions(index, y_true, rf_pred, dt_pred):
    plt.figure(figsize=(14, 6))
    plt.plot(index, y_true, label='Actual', linewidth=2)
    plt.plot(index, rf_pred, label='Random Forest', alpha=0.7)
    plt.title("Random Forest Predictions vs Actual")
    plt.xlabel("Index")
    plt.ylabel("Price")
    plt.legend()
```

```
    plt.grid(True)
    plt.tight_layout()
    plt.show()

    plt.figure(figsize=(14, 6))
    plt.plot(index, y_true, label='Actual', linewidth=2)
    plt.plot(index, dt_pred, label='Decision Tree', alpha=0.7)
    plt.title("Decision Tree Predictions vs Actual")
    plt.xlabel("Index")
    plt.ylabel("Price")
    plt.legend()
    plt.grid(True)
    plt.tight_layout()
    plt.show()

# Plot feature importances
def plot_feature_importances(model, X, title):
    if not hasattr(model, 'feature_importances_'):
        raise ValueError("Model does not support feature importances.")
    importances = model.feature_importances_
    indices = np.argsort(importances)[::-1]
    plt.figure(figsize=(12, 6))
    sns.barplot(x=importances[indices], y=np.array(X.columns)[indices])
    plt.title(title)
    plt.xlabel("Importance")
    plt.ylabel("Feature")
    plt.tight_layout()
    plt.show()
    return importances

# Main Prog
tickers = get_sp500_tickers()[:NUM_COMPANIES]
stock_data = download_stock_data(tickers, force_redownload=FORCE_REDOWNLOAD)
processed = preprocess_data(stock_data)
X, y = prepare_features(processed)

rf_model, rf_index, rf_y, rf_pred, rf_rmse, rf_r2 = train_and_evaluate(X, y,␣
 ↪model_name="RandomForest")
dt_model, dt_index, dt_y, dt_pred, dt_rmse, dt_r2 = train_and_evaluate(X, y,␣
 ↪model_name="DecisionTree")

plot_predictions(rf_index, rf_y, rf_pred, dt_pred)
rf_importances = plot_feature_importances(rf_model, X, "Random Forest Feature␣
 ↪Importances")
dt_importances = plot_feature_importances(dt_model, X, "Decision Tree Feature␣
 ↪Importances")
```

```
print(f"\nRandom Forest: RMSE = {rf_rmse:.2f}, R2 Score = {rf_r2:.4f}")
print(f"Decision Tree: RMSE = {dt_rmse:.2f}, R2 Score = {dt_r2:.4f}")
```

```
Downloading MMM (1/500)
Downloading AOS (2/500)
Downloading ABT (3/500)
Downloading ABBV (4/500)
Downloading ACN (5/500)
Downloading ADBE (6/500)
Downloading AMD (7/500)
Downloading AES (8/500)
Downloading AFL (9/500)
Downloading A (10/500)
Downloading APD (11/500)
Downloading ABNB (12/500)
Downloading AKAM (13/500)
Downloading ALB (14/500)
Downloading ARE (15/500)
Downloading ALGN (16/500)
Downloading ALLE (17/500)
Downloading LNT (18/500)
Downloading ALL (19/500)
Downloading GOOGL (20/500)
Downloading GOOG (21/500)
Downloading MO (22/500)
Downloading AMZN (23/500)
Downloading AMCR (24/500)
Downloading AEE (25/500)
Downloading AEP (26/500)
Downloading AXP (27/500)
Downloading AIG (28/500)
Downloading AMT (29/500)
Downloading AWK (30/500)
Downloading AMP (31/500)
Downloading AME (32/500)
Downloading AMGN (33/500)
Downloading APH (34/500)
Downloading ADI (35/500)
Downloading AON (36/500)
Downloading APA (37/500)
Downloading APO (38/500)
Downloading AAPL (39/500)
Downloading AMAT (40/500)
Downloading APTV (41/500)
Downloading ACGL (42/500)
Downloading ADM (43/500)
Downloading ANET (44/500)
Downloading AJG (45/500)
```

```
Downloading AIZ (46/500)
Downloading T (47/500)
Downloading ATO (48/500)
Downloading ADSK (49/500)
Downloading ADP (50/500)
Downloading AZO (51/500)
Downloading AVB (52/500)
Downloading AVY (53/500)
Downloading AXON (54/500)
Downloading BKR (55/500)
Downloading BALL (56/500)
Downloading BAC (57/500)
Downloading BAX (58/500)
Downloading BDX (59/500)
Downloading BRK-B (60/500)
Downloading BBY (61/500)
Downloading TECH (62/500)
Downloading BIIB (63/500)
Downloading BLK (64/500)
Downloading BX (65/500)
Downloading XYZ (66/500)
Downloading BK (67/500)
Downloading BA (68/500)
Downloading BKNG (69/500)
Downloading BSX (70/500)
Downloading BMY (71/500)
Downloading AVGO (72/500)
Downloading BR (73/500)
Downloading BRO (74/500)
Downloading BF-B (75/500)
Downloading BLDR (76/500)
Downloading BG (77/500)
Downloading BXP (78/500)
Downloading CHRW (79/500)
Downloading CDNS (80/500)
Downloading CZR (81/500)
Downloading CPT (82/500)
Downloading CPB (83/500)
Downloading COF (84/500)
Downloading CAH (85/500)
Downloading KMX (86/500)
Downloading CCL (87/500)
Downloading CARR (88/500)
Downloading CAT (89/500)
Downloading CBOE (90/500)
Downloading CBRE (91/500)
Downloading CDW (92/500)
Downloading COR (93/500)
```

```
Downloading CNC (94/500)
Downloading CNP (95/500)
Downloading CF (96/500)
Downloading CRL (97/500)
Downloading SCHW (98/500)
Downloading CHTR (99/500)
Downloading CVX (100/500)
Downloading CMG (101/500)
Downloading CB (102/500)
Downloading CHD (103/500)
Downloading CI (104/500)
Downloading CINF (105/500)
Downloading CTAS (106/500)
Downloading CSCO (107/500)
Downloading C (108/500)
Downloading CFG (109/500)
Downloading CLX (110/500)
Downloading CME (111/500)
Downloading CMS (112/500)
Downloading KO (113/500)
Downloading CTSH (114/500)
Downloading COIN (115/500)
Downloading CL (116/500)
Downloading CMCSA (117/500)
Downloading CAG (118/500)
Downloading COP (119/500)
Downloading ED (120/500)
Downloading STZ (121/500)
Downloading CEG (122/500)
Downloading COO (123/500)
Downloading CPRT (124/500)
Downloading GLW (125/500)
Downloading CPAY (126/500)
Downloading CTVA (127/500)
Downloading CSGP (128/500)
Downloading COST (129/500)
Downloading CTRA (130/500)
Downloading CRWD (131/500)
Downloading CCI (132/500)
Downloading CSX (133/500)
Downloading CMI (134/500)
Downloading CVS (135/500)
Downloading DHR (136/500)
Downloading DRI (137/500)
Downloading DDOG (138/500)
Downloading DVA (139/500)
Downloading DAY (140/500)
Downloading DECK (141/500)
```

```
Downloading DE (142/500)
Downloading DELL (143/500)
Downloading DAL (144/500)
Downloading DVN (145/500)
Downloading DXCM (146/500)
Downloading FANG (147/500)
Downloading DLR (148/500)
Downloading DG (149/500)
Downloading DLTR (150/500)
Downloading D (151/500)
Downloading DPZ (152/500)
Downloading DASH (153/500)
Downloading DOV (154/500)
Downloading DOW (155/500)
Downloading DHI (156/500)
Downloading DTE (157/500)
Downloading DUK (158/500)
Downloading DD (159/500)
Downloading EMN (160/500)
Downloading ETN (161/500)
Downloading EBAY (162/500)
Downloading ECL (163/500)
Downloading EIX (164/500)
Downloading EW (165/500)
Downloading EA (166/500)
Downloading ELV (167/500)
Downloading EMR (168/500)
Downloading ENPH (169/500)
Downloading ETR (170/500)
Downloading EOG (171/500)
Downloading EPAM (172/500)
Downloading EQT (173/500)
Downloading EFX (174/500)
Downloading EQIX (175/500)
Downloading EQR (176/500)
Downloading ERIE (177/500)
Downloading ESS (178/500)
Downloading EL (179/500)
Downloading EG (180/500)
Downloading EVRG (181/500)
Downloading ES (182/500)
Downloading EXC (183/500)
Downloading EXE (184/500)
Downloading EXPE (185/500)
Downloading EXPD (186/500)
Downloading EXR (187/500)
Downloading XOM (188/500)
Downloading FFIV (189/500)
```

```
Downloading FDS (190/500)
Downloading FICO (191/500)
Downloading FAST (192/500)
Downloading FRT (193/500)
Downloading FDX (194/500)
Downloading FIS (195/500)
Downloading FITB (196/500)
Downloading FSLR (197/500)
Downloading FE (198/500)
Downloading FI (199/500)
Downloading F (200/500)
Downloading FTNT (201/500)
Downloading FTV (202/500)
Downloading FOXA (203/500)
Downloading FOX (204/500)
Downloading BEN (205/500)
Downloading FCX (206/500)
Downloading GRMN (207/500)
Downloading IT (208/500)
Downloading GE (209/500)
Downloading GEHC (210/500)
Downloading GEV (211/500)
Downloading GEN (212/500)
Downloading GNRC (213/500)
Downloading GD (214/500)
Downloading GIS (215/500)
Downloading GM (216/500)
Downloading GPC (217/500)
Downloading GILD (218/500)
Downloading GPN (219/500)
Downloading GL (220/500)
Downloading GDDY (221/500)
Downloading GS (222/500)
Downloading HAL (223/500)
Downloading HIG (224/500)
Downloading HAS (225/500)
Downloading HCA (226/500)
Downloading DOC (227/500)
Downloading HSIC (228/500)
Downloading HSY (229/500)
Downloading HPE (230/500)
Downloading HLT (231/500)
Downloading HOLX (232/500)
Downloading HD (233/500)
Downloading HON (234/500)
Downloading HRL (235/500)
Downloading HST (236/500)
Downloading HWM (237/500)
```

```
Downloading HPQ (238/500)
Downloading HUBB (239/500)
Downloading HUM (240/500)
Downloading HBAN (241/500)
Downloading HII (242/500)
Downloading IBM (243/500)
Downloading IEX (244/500)
Downloading IDXX (245/500)
Downloading ITW (246/500)
Downloading INCY (247/500)
Downloading IR (248/500)
Downloading PODD (249/500)
Downloading INTC (250/500)
Downloading ICE (251/500)
Downloading IFF (252/500)
Downloading IP (253/500)
Downloading IPG (254/500)
Downloading INTU (255/500)
Downloading ISRG (256/500)
Downloading IVZ (257/500)
Downloading INVH (258/500)
Downloading IQV (259/500)
Downloading IRM (260/500)
Downloading JBHT (261/500)
Downloading JBL (262/500)
Downloading JKHY (263/500)
Downloading J (264/500)
Downloading JNJ (265/500)
Downloading JCI (266/500)
Downloading JPM (267/500)
Downloading K (268/500)
Downloading KVUE (269/500)
Downloading KDP (270/500)
Downloading KEY (271/500)
Downloading KEYS (272/500)
Downloading KMB (273/500)
Downloading KIM (274/500)
Downloading KMI (275/500)
Downloading KKR (276/500)
Downloading KLAC (277/500)
Downloading KHC (278/500)
Downloading KR (279/500)
Downloading LHX (280/500)
Downloading LH (281/500)
Downloading LRCX (282/500)
Downloading LW (283/500)
Downloading LVS (284/500)
Downloading LDOS (285/500)
```

```
Downloading LEN (286/500)
Downloading LII (287/500)
Downloading LLY (288/500)
Downloading LIN (289/500)
Downloading LYV (290/500)
Downloading LKQ (291/500)
Downloading LMT (292/500)
Downloading L (293/500)
Downloading LOW (294/500)
Downloading LULU (295/500)
Downloading LYB (296/500)
Downloading MTB (297/500)
Downloading MPC (298/500)
Downloading MKTX (299/500)
Downloading MAR (300/500)
Downloading MMC (301/500)
Downloading MLM (302/500)
Downloading MAS (303/500)
Downloading MA (304/500)
Downloading MTCH (305/500)
Downloading MKC (306/500)
Downloading MCD (307/500)
Downloading MCK (308/500)
Downloading MDT (309/500)
Downloading MRK (310/500)
Downloading META (311/500)
Downloading MET (312/500)
Downloading MTD (313/500)
Downloading MGM (314/500)
Downloading MCHP (315/500)
Downloading MU (316/500)
Downloading MSFT (317/500)
Downloading MAA (318/500)
Downloading MRNA (319/500)
Downloading MHK (320/500)
Downloading MOH (321/500)
Downloading TAP (322/500)
Downloading MDLZ (323/500)
Downloading MPWR (324/500)
Downloading MNST (325/500)
Downloading MCO (326/500)
Downloading MS (327/500)
Downloading MOS (328/500)
Downloading MSI (329/500)
Downloading MSCI (330/500)
Downloading NDAQ (331/500)
Downloading NTAP (332/500)
Downloading NFLX (333/500)
```

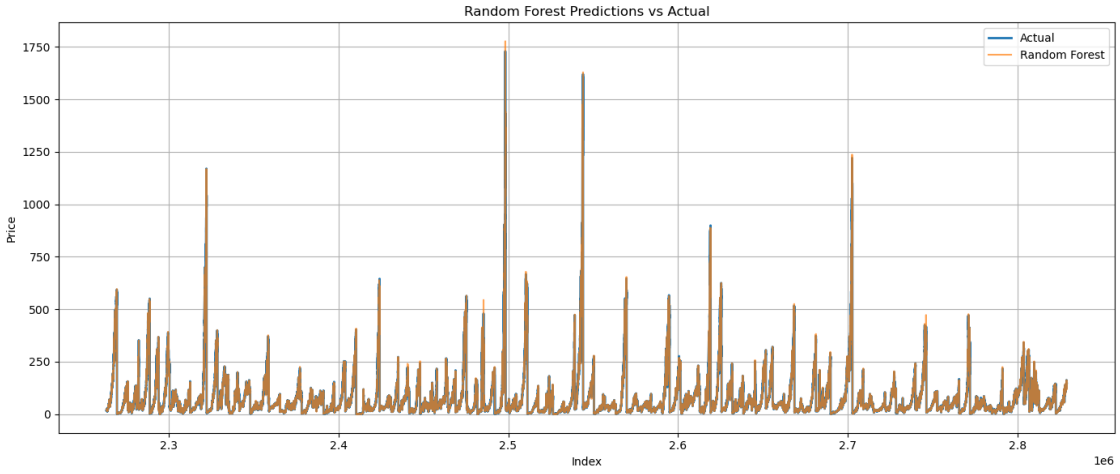
```
Downloading NEM (334/500)
Downloading NWSA (335/500)
Downloading NWS (336/500)
Downloading NEE (337/500)
Downloading NKE (338/500)
Downloading NI (339/500)
Downloading NDSN (340/500)
Downloading NSC (341/500)
Downloading NTRS (342/500)
Downloading NOC (343/500)
Downloading NCLH (344/500)
Downloading NRG (345/500)
Downloading NUE (346/500)
Downloading NVDA (347/500)
Downloading NVR (348/500)
Downloading NXPI (349/500)
Downloading ORLY (350/500)
Downloading OXY (351/500)
Downloading ODFL (352/500)
Downloading OMC (353/500)
Downloading ON (354/500)
Downloading OKE (355/500)
Downloading ORCL (356/500)
Downloading OTIS (357/500)
Downloading PCAR (358/500)
Downloading PKG (359/500)
Downloading PLTR (360/500)
Downloading PANW (361/500)
Downloading PARA (362/500)
Downloading PH (363/500)
Downloading PAYX (364/500)
Downloading PAYC (365/500)
Downloading PYPL (366/500)
Downloading PNR (367/500)
Downloading PEP (368/500)
Downloading PFE (369/500)
Downloading PCG (370/500)
Downloading PM (371/500)
Downloading PSX (372/500)
Downloading PNW (373/500)
Downloading PNC (374/500)
Downloading POOL (375/500)
Downloading PPG (376/500)
Downloading PPL (377/500)
Downloading PFG (378/500)
Downloading PG (379/500)
Downloading PGR (380/500)
Downloading PLD (381/500)
```

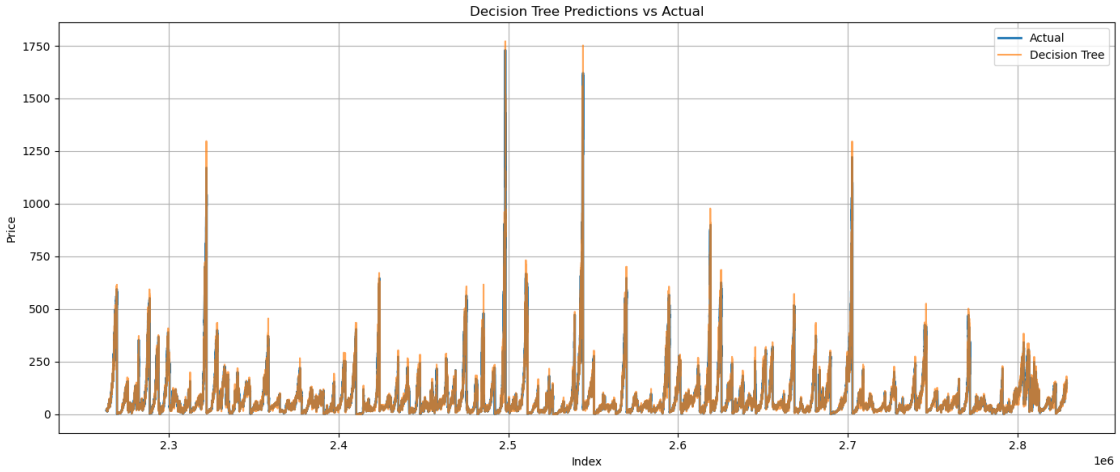
```
Downloading PRU (382/500)
Downloading PEG (383/500)
Downloading PTC (384/500)
Downloading PSA (385/500)
Downloading PHM (386/500)
Downloading PWR (387/500)
Downloading QCOM (388/500)
Downloading DGX (389/500)
Downloading RL (390/500)
Downloading RJF (391/500)
Downloading RTX (392/500)
Downloading O (393/500)
Downloading REG (394/500)
Downloading REGN (395/500)
Downloading RF (396/500)
Downloading RSG (397/500)
Downloading RMD (398/500)
Downloading RVTY (399/500)
Downloading ROK (400/500)
Downloading ROL (401/500)
Downloading ROP (402/500)
Downloading ROST (403/500)
Downloading RCL (404/500)
Downloading SPGI (405/500)
Downloading CRM (406/500)
Downloading SBAC (407/500)
Downloading SLB (408/500)
Downloading STX (409/500)
Downloading SRE (410/500)
Downloading NOW (411/500)
Downloading SHW (412/500)
Downloading SPG (413/500)
Downloading SWKS (414/500)
Downloading SJM (415/500)
Downloading SW (416/500)
Downloading SNA (417/500)
Downloading SOLV (418/500)
Downloading SO (419/500)
Downloading LUV (420/500)
Downloading SWK (421/500)
Downloading SBUX (422/500)
Downloading STT (423/500)
Downloading STLD (424/500)
Downloading STE (425/500)
Downloading SYK (426/500)
Downloading SMCI (427/500)
Downloading SYF (428/500)
Downloading SNPS (429/500)
```

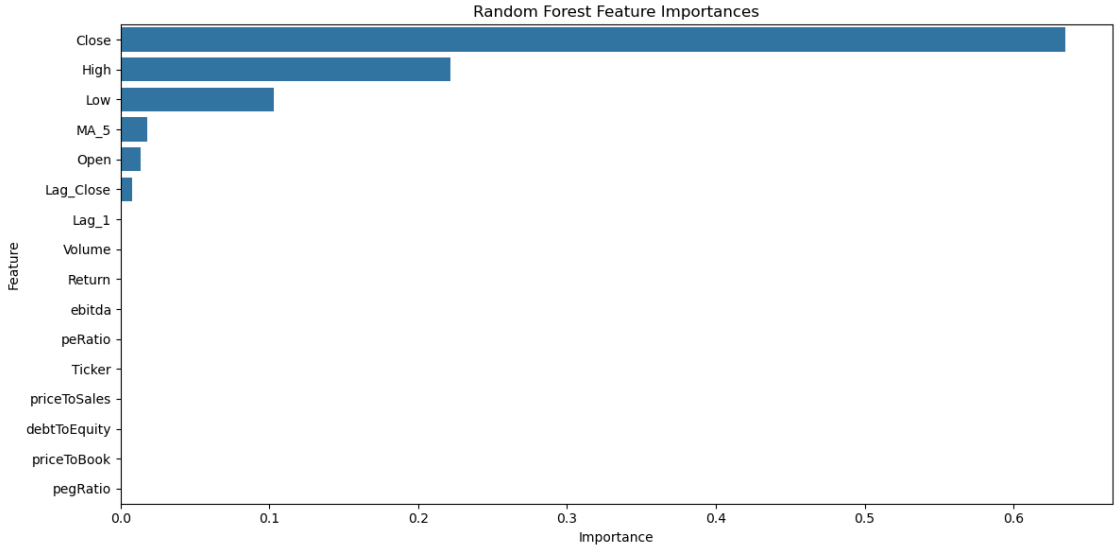
```
Downloading SYY (430/500)
Downloading TMUS (431/500)
Downloading TROW (432/500)
Downloading TTWO (433/500)
Downloading TPR (434/500)
Downloading TRGP (435/500)
Downloading TGT (436/500)
Downloading TEL (437/500)
Downloading TDY (438/500)
Downloading TER (439/500)
Downloading TSLA (440/500)
Downloading TXN (441/500)
Downloading TPL (442/500)
Downloading TXT (443/500)
Downloading TMO (444/500)
Downloading TJX (445/500)
Downloading TKO (446/500)
Downloading TTD (447/500)
Downloading TSCO (448/500)
Downloading TT (449/500)
Downloading TDG (450/500)
Downloading TRV (451/500)
Downloading TRMB (452/500)
Downloading TFC (453/500)
Downloading TYL (454/500)
Downloading TSN (455/500)
Downloading USB (456/500)
Downloading UBER (457/500)
Downloading UDR (458/500)
Downloading ULTA (459/500)
Downloading UNP (460/500)
Downloading UAL (461/500)
Downloading UPS (462/500)
Downloading URI (463/500)
Downloading UNH (464/500)
Downloading UHS (465/500)
Downloading VLO (466/500)
Downloading VTR (467/500)
Downloading VLTO (468/500)
Downloading VRSN (469/500)
Downloading VRSK (470/500)
Downloading VZ (471/500)
Downloading VRTX (472/500)
Downloading VTRS (473/500)
Downloading VICI (474/500)
Downloading V (475/500)
Downloading VST (476/500)
Downloading VMC (477/500)
```

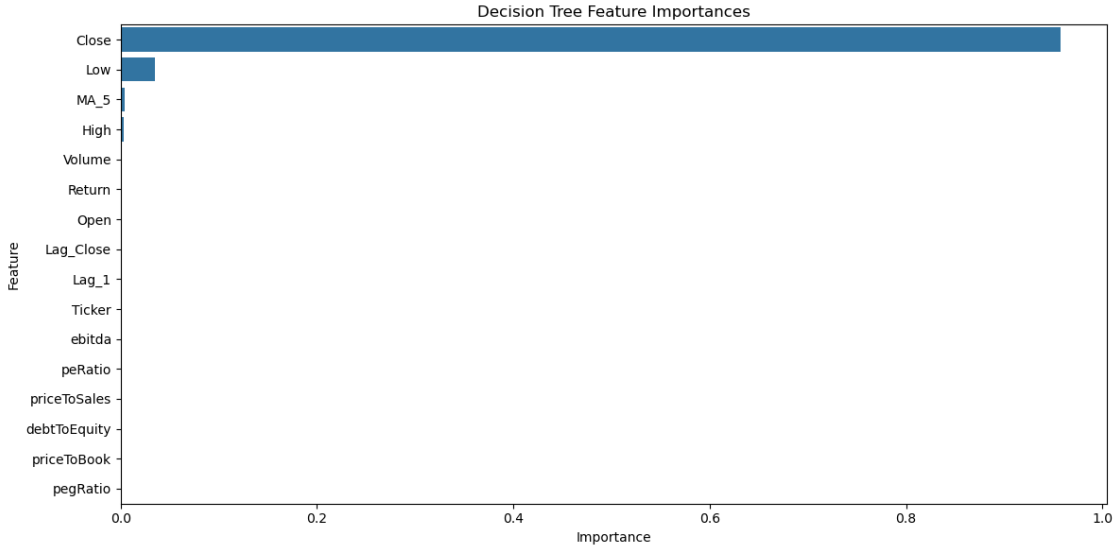
```
Downloading WRB (478/500)
Downloading GWW (479/500)
Downloading WAB (480/500)
Downloading WBA (481/500)
Downloading WMT (482/500)
Downloading DIS (483/500)
Downloading WBD (484/500)
Downloading WM (485/500)
Downloading WAT (486/500)
Downloading WEC (487/500)
Downloading WFC (488/500)
Downloading WELL (489/500)
Downloading WST (490/500)
Downloading WDC (491/500)
Downloading WY (492/500)
Downloading WSM (493/500)
Downloading WMB (494/500)
Downloading WTW (495/500)
Downloading WDAY (496/500)
Downloading WYNN (497/500)
Downloading XEL (498/500)
Downloading XYL (499/500)
Downloading YUM (500/500)
```

```
C:\Users\Steve\anaconda3\Lib\site-packages\sklearn\metrics\_regression.py:492:
FutureWarning: 'squared' is deprecated in version 1.4 and will be removed in
1.6. To calculate the root mean squared error, use the
function'root_mean_squared_error'.
  warnings.warn(
C:\Users\Steve\anaconda3\Lib\site-packages\sklearn\metrics\_regression.py:492:
FutureWarning: 'squared' is deprecated in version 1.4 and will be removed in
1.6. To calculate the root mean squared error, use the
function'root_mean_squared_error'.
  warnings.warn(
```




Decision Tree Predictions vs Actual
Actual
Decision Tree
Price
Index
1e6
0
250
500
750
1000
1250
1500
1750
2.3
2.4
2.5
2.6
2.7
2.8


Random Forest Feature Importances
Feature
Importance
Close
High
Low
MA_5
Open
Lag_Close
Lag_1
Volume
Return
ebitda
peRatio
Ticker
priceToSales
debtToEquity
priceToBook
pegRatio
0.0
0.1
0.2
0.3
0.4
0.5
0.6




```
Random Forest: RMSE = 6.09, R2 Score = 0.9968
Decision Tree: RMSE = 9.08, R2 Score = 0.9929
```