Prophet modeling

Here we'll try to mimic the same process of finding the best train-test split, but for Facebook's Prophet library. Let's import everything.

In [1]:

```
from fbprophet import Prophet
from prophet.diagnostics import cross_validation
from prophet.plot import plot_cross_validation_metric
from prophet.diagnostics import performance_metrics
from sklearn.linear_model import LinearRegression
from iexfinance.stocks import Stock
import random
#from trafalgar import*
import pandas as pd
import pandas.tseries
import numpy as np
import seaborn as sb
import matplotlib.pyplot as plt
import statsmodels.api as sm
import matplotlib
import pmdarima as pm
import datetime as dt
from datetime import date
from datetime import timedelta
import yfinance as yf
import requests
from sklearn.model_selection import TimeSeriesSplit
from sktime.forecasting.model_selection import temporal_train_test_split
from pandas.plotting import lag_plot
from pandas import datetime
import re
from tiingo import TiingoClient
import json
from pandas_datareader import data as pdr
```

In [2]:

```
yf.__version__
```

Out[2]:

```
'0.1.63'
```

The test case: C

Just like the previous notebook, we'll try to run through a simple test case and expand it to other stocks. We'll use Citicgroup again.

Quick note: I encountered a bug with my earlier library, yfinance, so I've had to switch to tiingo, so a lot of code will be hashed out from earlier. And in case I can get the bugs worked out.

In [3]:

```
c = yf.Ticker("C")
```

In [4]:

```
df=c.history(period="2y")
```

In [5]:

```
df.head()
```

Out[5]:

Date	Open	High	Low	Close	Volume	Dividends	Stock Splits
2019-07-29	67.075884	67.476971	66.907984	66.935966	8561200	0.00	0
2019-07-30	66.255033	66.889320	65.835287	66.889320	9208300	0.00	0
2019-07-31	66.600161	66.842685	66.049820	66.376297	13617300	0.00	0
2019-08-01	65.919222	66.310986	63.055603	63.755184	22571000	0.00	0
2019-08-02	63.539043	63.961945	62.618048	63.539043	14771400	0.51	0

In [6]:

```
df.head()
```

Out[6]:

Date	Open	High	Low	Close	Volume	Dividends	Stock Splits
2019-07-29	67.075884	67.476971	66.907984	66.935966	8561200	0.00	0
2019-07-30	66.255033	66.889320	65.835287	66.889320	9208300	0.00	0
2019-07-31	66.600161	66.842685	66.049820	66.376297	13617300	0.00	0
2019-08-01	65.919222	66.310986	63.055603	63.755184	22571000	0.00	0
2019-08-02	63.539043	63.961945	62.618048	63.539043	14771400	0.51	0

In [7]:

```
deltas=['d','m','y']
att='200d'
```

In [8]:

```
att[-1]
```

Out[8]:

```
'd'
```

In [9]:

```
spl=len(att)-1
att[:spl]
```

Out[9]:

```
'200'
```

In [10]:

```
end=att[:spl] + ' days'
end
```

Out[10]:

```
'200 days'
```

In [11]:

```
df1=df['Close']
```

In [12]:

```
df1.head()
```

Out[12]:

```
Date
2019-07-29    66.935966
2019-07-30    66.889320
2019-07-31    66.376297
2019-08-01    63.755184
2019-08-02    63.539043
Name: Close, dtype: float64
```

In [13]:

```
df1=df1.to_frame()
```

In [14]:

```
# instantiate Prophet
prof_1 = Prophet()
```

In [15]:

```
df1.head()
```

Out[15]:

Date	Close
2019-07-29	66.935966
2019-07-30	66.889320
2019-07-31	66.376297
2019-08-01	63.755184
2019-08-02	63.539043

In [16]:

```
df1.index.names = ['ds']
df1.columns=['y']
```

In [17]:

```
df1.reset_index(level=0, inplace=True)
```

In [18]:

```
df1.head()
```

Out[18]:

	ds	y
0	2019-07-29	66.935966
1	2019-07-30	66.889320
2	2019-07-31	66.376297
3	2019-08-01	63.755184
4	2019-08-02	63.539043

In [19]:

```
df1.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 503 entries, 0 to 502
Data columns (total 2 columns):
 #   Column  Non-Null Count  Dtype
```

```
 0   ds      503 non-null    datetime64[ns]
 1   y       503 non-null    float64
dtypes: datetime64[ns](1), float64(1)
memory usage: 8.0 KB
```

In [20]:

```
prof_1.fit(df1)
```

```
INFO:fbprophet:Disabling yearly seasonality. Run prophet with yearly_seasonality=True to override this.
INFO:fbprophet:Disabling daily seasonality. Run prophet with daily_seasonality=True to ov
erride this.
```

Out[20]:

```
<fbprophet.forecaster.Prophet at 0x22efe5a5b00>
```

In [21]:

```
#Create a new dataframe for the predictions, 3 weeks out
future = prof_1.make_future_dataframe(periods=21)
```

In [22]:

```
forecast = prof_1.predict(future)
```

In [23]:

```
forecast.tail()
```

Out[23]:

	ds	trend	yhat_lower	yhat_upper	trend_lower	trend_upper	additive_terms	additive_terms_lower	additive_terms_
519	**2021-08-12**	**76.903740**	**69.819677**	**81.958577**	**76.646745**	**77.112426**	**-0.972526**	**-0.972526**	**-0.9**
520	**2021-08-13**	**76.951879**	**69.572250**	**82.305244**	**76.669904**	**77.212222**	**-0.981443**	**-0.981443**	**-0.9**
521	**2021-08-14**	**77.000018**	**73.411625**	**85.927658**	**76.691431**	**77.307015**	**2.491078**	**2.491078**	**2.4**
522	**2021-08-15**	**77.048156**	**73.271870**	**85.761768**	**76.701266**	**77.394474**	**2.491078**	**2.491078**	**2.4**
523	**2021-08-16**	**77.096295**	**69.244465**	**82.237461**	**76.709998**	**77.470911**	**-1.062413**	**-1.062413**	**-1.0**

In [24]:

```
forecast.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 524 entries, 0 to 523
Data columns (total 16 columns):
 #   Column                  Non-Null Count  Dtype
---  ------                  --------------  -----
 0   ds                      524 non-null    datetime64[ns]
 1   trend                   524 non-null    float64
 2   yhat_lower              524 non-null    float64
 3   yhat_upper              524 non-null    float64
 4   trend_lower             524 non-null    float64
 5   trend_upper             524 non-null    float64
 6   additive_terms          524 non-null    float64
 7   additive_terms_lower    524 non-null    float64
 8   additive_terms_upper    524 non-null    float64
 9   weekly                  524 non-null    float64
 10  weekly_lower            524 non-null    float64
 11  weekly_upper            524 non-null    float64
```

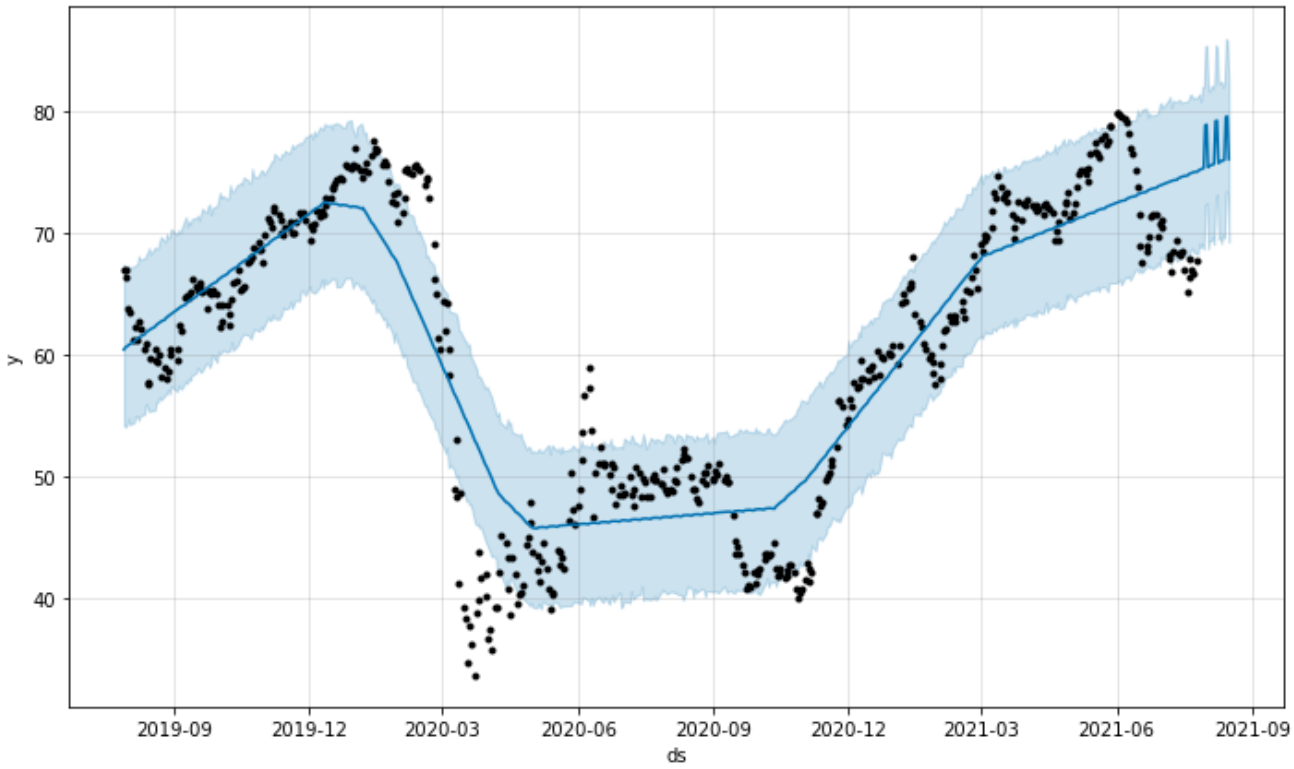
```
 12  multiplicative_terms        524 non-null    float64
 13  multiplicative_terms_lower  524 non-null    float64
 14  multiplicative_terms_upper  524 non-null    float64
 15  yhat                        524 non-null    float64
dtypes: datetime64[ns](1), float64(15)
memory usage: 65.6 KB
```

In [25]:

```
proph_pred=forecast['yhat']
```

In [26]:

```
prof_1.plot(forecast)
plt.show()
```



In [27]:

```
df1_cv = cross_validation(prof_1, initial='30 days', period='7 days', horizon = '14 days
')
```

```
INFO:prophet:Making 98 forecasts with cutoffs between 2019-09-02 00:00:00 and 2021-07-12
00:00:00
```

```
INFO:fbprophet:n_changepoints greater than number of observations. Using 19.
INFO:fbprophet:n_changepoints greater than number of observations. Using 23.
```

In [28]:

```
df1_pm = performance_metrics(df1_cv)
```

In [29]:

```
trains=['30 days','60 days','180 days']
tests=['7 days','14 days','21 days']
```

In [30]:

```
df1_pm.tail()
```

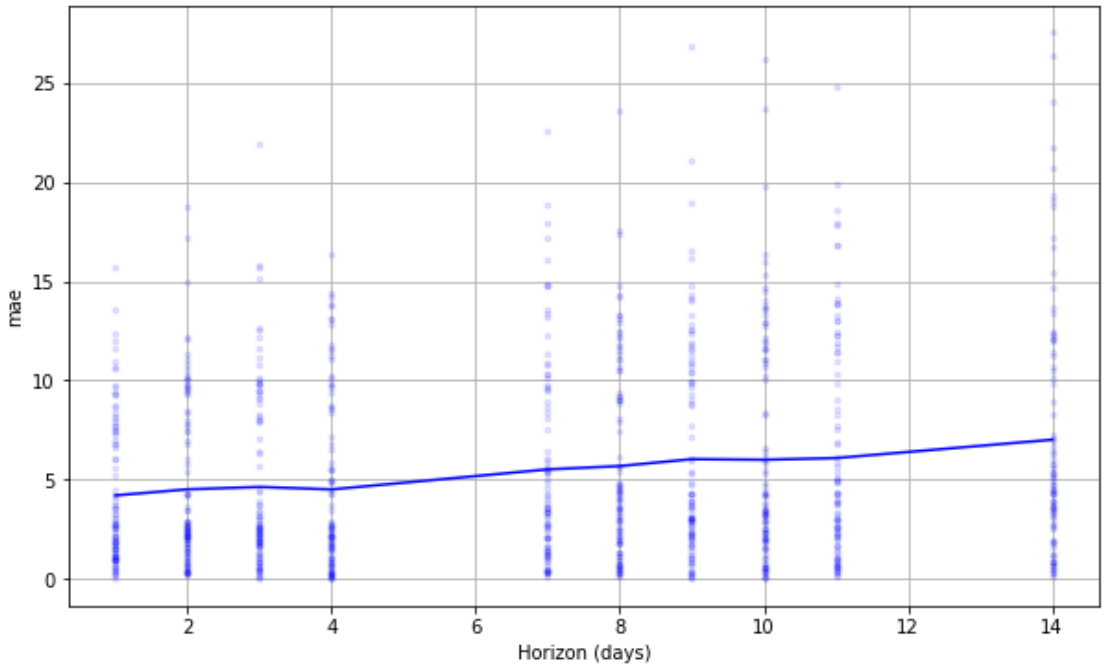
Out[30]:

	horizon	mse	rmse	mae	mape	mdape	smape	coverage
5	8 days	56.798665	7.536489	5.675017	0.107800	0.066421	0.110591	0.479592

	horizon	mse	rmse	mae	mape	mdape	smape	coverage
6	9 days	65.597044	8.099200	6.023453	0.115201	0.065023	0.118581	0.510417
7	10 days	68.186069	8.257486	5.992518	0.114085	0.062379	0.116996	0.510417
8	11 days	68.642071	8.285051	6.078169	0.114816	0.066957	0.118065	0.473515
9	14 days	89.278276	9.448718	7.004400	0.135089	0.078113	0.139062	0.392473

In [31]:

```
fig = plot_cross_validation_metric(df1_cv, metric='mae')
```



STONKS (Again)!!!

Just like in the ARIMA notebook, I want to test a number of train-test splits with Prophet. Since it works a little different from ARIMA, the code will be a little different, but still borrowing a lot. Here, we can import our stock symbols, and clean them up.

In [32]:

```
sp_500=pd.read_csv('Data/constituents_csv.csv')
nsdq=pd.read_csv('Data/nasdaq1.csv')
dow_30=pd.read_excel('Data/dow-jones-industrial-average-components.xls')
```

In [33]:

```
new_cols=['Name','Symbol','Weight%']
dow_30.columns=new_cols
```

In [34]:

```
nsdq.head()
```

Out[34]:

	Unnamed: 0	Symbol	Company Name
0	0	AAIT	iShares MSCI All Country Asia Information Tech...
1	1	AAL	American Airlines Group, Inc.
2	2	AAME	Atlantic American Corporation
3	3	AAOI	Applied Optoelectronics, Inc.
4	4	AAON	AAON, Inc.

In [35]:

```
nsdq.drop(columns='Unnamed: 0',inplace=True)
```

In [36]:

```
len(nsdq)
```

Out[36]:

1734

The following cells were necessary for cleaning the nasdaq stock list. But I saved it, and that's what I'm using.

In [37]:

```
#no_data=[]
#for each in nsdq['Symbol']:
#    x=yf.Ticker(each)
#    df=x.history(period='1d')
#    if len(df)==0:
#        no_data.append(each)
```

In [38]:

```
#len(no_data)
```

In [39]:

```
#nd_index=[]
#for each in no_data:
#    y=nsdq.loc[nsdq['Symbol']==each].index
#    nd_index.append(y[0])
```

In [40]:

```
#nsdq = nsdq.drop(labels=nd_index,drop=True, axis=0)
#nsdq.reset_index()
```

In [41]:

```
#nsdq.to_csv("/Users/Daniel/Documents/Flatiron/Capstone/Project/nasdaq1.csv")
```

In [42]:

```
def tt_test_p (asset,train_val,test_val):
    """This function will take in a financial asset (stock, etf) as well as 2 lists of in
tegers (training and testing days).
    Then the asset will be looked up through yahoo finance and gather the price history.
It will then run through the values
    of the training and testing lists and run prophet models on all of them. It will reco
rd the metrics and return a
    dataframe with all the results."""

    stock = yf.Ticker(asset)
    df1=stock.history(period='2y')
    print("Processing: ",stock)
    prof_1 = Prophet()
    df1=df1['Close']
    df1=df1.to_frame()
    df1.index.names = ['ds']
    df1.columns=['y']
    df1.reset_index(level=0, inplace=True)
    prof_1.fit(df1)
    future = prof_1.make_future_dataframe(periods=21)
    forecast = prof_1.predict(future)
    for train_val in trains:
        for test_val in tests:
            df1_cv = cross_validation(prof_1, initial=train_val, period=test_val, horizo
```

```
n = '21 days')
            df1_pm = performance_metrics(df1_cv)
            print('Training: ', train_val)
            print('Testing: ', test_val)
            print (df1_pm.tail())

    return forecast
```

In [43]:

```
cols2=['Symbol','Train_Len','Test_Len','MAE','RMSE']
reslts = pd.DataFrame(columns=cols2)
reslts.reset_index()
```

Out[43]:

index	Symbol	Train_Len	Test_Len	MAE	RMSE

In [44]:

```
results_CAT=tt_test_p('CAT',trains,tests)
```

```
INFO:fbprophet:Disabling yearly seasonality. Run prophet with yearly_seasonality=True to override this.
INFO:fbprophet:Disabling daily seasonality. Run prophet with daily_seasonality=True to ov
erride this.
Processing:  yfinance.Ticker object <CAT>
INFO:prophet:Making 97 forecasts with cutoffs between 2019-09-02 00:00:00 and 2021-07-05 00:00:00
INFO:fbprophet:n_changepoints greater than number of observations. Using 19.
INFO:fbprophet:n_changepoints greater than number of observations. Using 23.
INFO:prophet:Making 49 forecasts with cutoffs between 2019-09-02 00:00:00 and 2021-07-05 00:00:00
Training:  30 days
Testing:  7 days
   horizon         mse       rmse        mae      mape     mdape     smape  \
9  15 days  218.289636  14.774628  11.044398  0.075452  0.048079  0.075291
10 16 days  230.346149  15.177159  11.344151  0.076742  0.048230  0.076755
11 17 days  240.167915  15.497352  11.425325  0.077534  0.045453  0.077295
12 18 days  259.039415  16.094701  11.899763  0.080329  0.045840  0.079834
13 21 days  291.574721  17.075559  12.714797  0.087312  0.051852  0.086787

    coverage
9   0.470064
10  0.473859
11  0.445113
12  0.412256
13  0.425078
INFO:fbprophet:n_changepoints greater than number of observations. Using 19.
INFO:prophet:Making 33 forecasts with cutoffs between 2019-09-02 00:00:00 and 2021-07-05 00:00:00
Training:  30 days
Testing:  14 days
   horizon         mse       rmse        mae      mape     mdape     smape  \
9  15 days  229.943309  15.163882  11.141900  0.075656  0.048181  0.075435
10 16 days  232.299538  15.241376  11.187911  0.074944  0.045665  0.074729
11 17 days  228.868767  15.128409  11.066080  0.074211  0.043531  0.073893
12 18 days  251.263481  15.851293  11.688075  0.079194  0.040366  0.078547
13 21 days  300.353143  17.330699  12.946871  0.089745  0.053892  0.088598

    coverage
9   0.489355
10  0.524863
11  0.507042
12  0.461193
```

```
13  0.432724
```

```
INFO:fbprophet:n_changepoints greater than number of observations. Using 19.
INFO:prophet:Making 93 forecasts with cutoffs between 2019-09-30 00:00:00 and 2021-07-05
00:00:00
```

```
Training:  30 days
Testing:  21 days
   horizon         mse       rmse        mae      mape     mdape     smape  \
9  15 days  229.774068  15.158300  11.819291  0.078904  0.058921  0.079131
10 16 days  222.134427  14.904175  11.832079  0.077670  0.061964  0.078457
11 17 days  223.517096  14.950488  11.894005  0.076575  0.049084  0.077341
12 18 days  244.642206  15.641042  12.322506  0.079644  0.065716  0.080669
13 21 days  248.814387  15.773851  12.113769  0.081166  0.058998  0.082237

    coverage
9   0.405674
10  0.381044
11  0.345745
12  0.319149
13  0.389628
```

```
INFO:prophet:Making 47 forecasts with cutoffs between 2019-09-30 00:00:00 and 2021-07-05
00:00:00
```

```
Training:  60 days
Testing:  7 days
   horizon         mse       rmse        mae      mape     mdape     smape  \
9  15 days  211.664004  14.548677  10.761600  0.072264  0.046812  0.072447
10 16 days  221.646943  14.887812  11.033988  0.073189  0.047426  0.073670
11 17 days  231.123604  15.202750  11.117745  0.074003  0.043171  0.074265
12 18 days  251.999099  15.874479  11.669091  0.077415  0.049861  0.077386
13 21 days  283.424225  16.835208  12.479690  0.084327  0.050299  0.084359

    coverage
9   0.490869
10  0.494543
11  0.457192
12  0.427013
13  0.452256
```

```
INFO:prophet:Making 31 forecasts with cutoffs between 2019-10-14 00:00:00 and 2021-07-05
00:00:00
```

```
Training:  60 days
Testing:  14 days
   horizon         mse       rmse        mae      mape     mdape     smape  \
9  15 days  214.681340  14.652008  10.620241  0.070548  0.047165  0.070760
10 16 days  212.405573  14.574141  10.612031  0.069161  0.042421  0.069602
11 17 days  206.023032  14.353502  10.451250  0.068013  0.042686  0.068485
12 18 days  229.723433  15.156630  11.117851  0.073356  0.045743  0.073504
13 21 days  276.678207  16.633647  12.379136  0.083885  0.045296  0.083652

    coverage
9   0.518651
10  0.562265
11  0.551471
12  0.489216
13  0.451961
```

```
INFO:prophet:Making 76 forecasts with cutoffs between 2020-01-27 00:00:00 and 2021-07-05
00:00:00
```

```
Training:  60 days
Testing:  21 days
   horizon         mse       rmse        mae      mape     mdape     smape  \
9  15 days  227.855180  15.094873  11.537939  0.075162  0.060808  0.075432
10 16 days  220.892801  14.862463  11.589932  0.074145  0.055190  0.075038
11 17 days  225.576488  15.019204  11.768395  0.074020  0.051142  0.074834
12 18 days  252.568788  15.892413  12.422774  0.079008  0.054479  0.080002
13 21 days  258.474705  16.077149  12.248187  0.080988  0.054479  0.082002
```

```
    coverage
9   0.443182
10  0.406158
11  0.390909
12  0.350440
13  0.416667
```

```
INFO:prophet:Making 38 forecasts with cutoffs between 2020-02-03 00:00:00 and 2021-07-05
00:00:00
```

```
Training:  180 days
Testing:  7 days
   horizon         mse       rmse        mae      mape     mdape     smape  \
9  15 days  218.241198  14.772989  10.618077  0.069419  0.039057  0.069499
10 16 days  225.919300  15.030612  10.809982  0.069674  0.039337  0.070064
11 17 days  241.338877  15.535085  11.060267  0.071726  0.045961  0.071783
12 18 days  258.211061  16.068947  11.496599  0.074058  0.036824  0.073733
13 21 days  282.302513  16.801860  12.085926  0.079521  0.041846  0.079322

    coverage
9   0.538472
10  0.548565
11  0.503110
12  0.461091
13  0.495262
```

```
INFO:prophet:Making 26 forecasts with cutoffs between 2020-01-27 00:00:00 and 2021-07-05
00:00:00
```

```
Training:  180 days
Testing:  14 days
   horizon         mse       rmse        mae      mape     mdape     smape  \
9  15 days  225.311498  15.010380  10.593740  0.068275  0.038209  0.068433
10 16 days  222.086938  14.902582  10.578986  0.066774  0.036225  0.067122
11 17 days  221.701941  14.889659  10.685332  0.067582  0.034540  0.067837
12 18 days  240.657146  15.513128  11.140105  0.071309  0.042560  0.071243
13 21 days  282.381820  16.804220  12.246034  0.080928  0.038121  0.080536

    coverage
9   0.550413
10  0.607177
11  0.588517
12  0.523342
13  0.481818


Training:  180 days
Testing:  21 days
   horizon         mse       rmse        mae      mape     mdape     smape  \
9  15 days  230.272989  15.174748  11.436223  0.072651  0.042030  0.072677
10 16 days  215.961202  14.695618  11.213672  0.069488  0.043115  0.070278
11 17 days  229.240461  15.140689  11.600104  0.070510  0.061964  0.071259
12 18 days  254.036155  15.938512  12.240935  0.075429  0.065716  0.076275
13 21 days  247.329089  15.726700  11.654565  0.075012  0.043040  0.075841

    coverage
9   0.421171
10  0.423077
11  0.396050
12  0.367983
13  0.460541
```

In [45]:

```
results_MMM=tt_test_p('MMM',trains,tests)
```

```
INFO:fbprophet:Disabling yearly seasonality. Run prophet with yearly_seasonality=True to
override this.
INFO:fbprophet:Disabling daily seasonality. Run prophet with daily_seasonality=True to ov
erride this.
```

```
Processing:  yfinance.Ticker object <MMM>
```

```
INFO:prophet:Making 97 forecasts with cutoffs between 2019-09-02 00:00:00 and 2021-07-05
00:00:00
```

```
INFO:fbprophet:n_changepoints greater than number of observations. Using 19.
INFO:fbprophet:n_changepoints greater than number of observations. Using 23.
INFO:prophet:Making 49 forecasts with cutoffs between 2019-09-02 00:00:00 and 2021-07-05
00:00:00
```

```
Training:  30 days
Testing:  7 days
   horizon         mse       rmse       mae      mape     mdape     smape  \
9  15 days  118.471230  10.884449  8.439592  0.053141  0.044468  0.053503
10 16 days  121.333133  11.015132  8.463251  0.053016  0.044468  0.053505
11 17 days  122.762516  11.079825  8.681469  0.054342  0.038575  0.054731
12 18 days  130.268654  11.413529  8.947026  0.055980  0.044697  0.056208
13 21 days  155.063041  12.452431  9.664070  0.060906  0.050669  0.061321

    coverage
9   0.456019
10  0.456259
11  0.466917
12  0.455263
13  0.426630
```

```
INFO:fbprophet:n_changepoints greater than number of observations. Using 19.
INFO:prophet:Making 33 forecasts with cutoffs between 2019-09-02 00:00:00 and 2021-07-05
00:00:00
```

```
Training:  30 days
Testing:  14 days
   horizon         mse       rmse       mae      mape     mdape     smape  \
9  15 days  132.638100  11.516862  8.704179  0.054475  0.045053  0.054958
10 16 days  127.519805  11.292467  8.396987  0.052438  0.045041  0.052875
11 17 days  122.731639  11.078431  8.458148  0.052721  0.037709  0.053017
12 18 days  129.875465  11.396292  8.846456  0.055717  0.045160  0.055730
13 21 days  157.401815  12.545988  9.833561  0.062595  0.048021  0.062484

    coverage
9   0.439895
10  0.448692
11  0.492958
12  0.468385
13  0.404555
```

```
INFO:fbprophet:n_changepoints greater than number of observations. Using 19.
INFO:prophet:Making 93 forecasts with cutoffs between 2019-09-30 00:00:00 and 2021-07-05
00:00:00
```

```
Training:  30 days
Testing:  21 days
   horizon         mse       rmse       mae      mape     mdape     smape  \
9  15 days  123.175267  11.098435  8.568936  0.054324  0.045053  0.054753
10 16 days  113.822977  10.668785  8.157719  0.051553  0.045126  0.052157
11 17 days  117.486252  10.839108  8.552730  0.053749  0.034853  0.054232
12 18 days  126.315405  11.239013  8.900328  0.055698  0.047370  0.056362
13 21 days  125.105359  11.185051  8.569562  0.053658  0.037996  0.054838

    coverage
9   0.448227
10  0.442940
11  0.449468
12  0.442940
13  0.456117
```

```
INFO:prophet:Making 47 forecasts with cutoffs between 2019-09-30 00:00:00 and 2021-07-05
00:00:00
```

```
Training:  60 days
Testing:  7 days
```

```
   horizon         mse       rmse       mae      mape     mdape     smape  \
9  15 days  112.976577  10.629044  8.181924  0.051292  0.040446  0.051831
10 16 days  113.873891  10.671171  8.155846  0.050771  0.045084  0.051495
11 17 days  113.927046  10.673661  8.351350  0.051927  0.038575  0.052583
12 18 days  122.286195  11.058309  8.656108  0.053855  0.040816  0.054332
13 21 days  146.826553  12.117201  9.392643  0.058913  0.047223  0.059611

    coverage
9   0.472344
10  0.465254
11  0.483927
12  0.483188
13  0.435719
```

```
INFO:prophet:Making 31 forecasts with cutoffs between 2019-10-14 00:00:00 and 2021-07-05
00:00:00
```

```
Training:  60 days
Testing:  14 days
   horizon         mse       rmse       mae      mape     mdape     smape  \
9  15 days  123.866573  11.129536  8.323433  0.051889  0.041766  0.052558
10 16 days  114.903406  10.719301  7.935488  0.049219  0.041582  0.049936
11 17 days  107.036584  10.345849  7.948966  0.049132  0.040027  0.049774
12 18 days  113.613978  10.658986  8.341753  0.052173  0.044152  0.052543
13 21 days  138.687332  11.776559  9.304766  0.058855  0.046149  0.059163

    coverage
9   0.459469
10  0.482791
11  0.507353
12  0.466993
13  0.400327
```

```
INFO:prophet:Making 76 forecasts with cutoffs between 2020-01-27 00:00:00 and 2021-07-05
00:00:00
```

```
Training:  60 days
Testing:  21 days
   horizon         mse       rmse       mae      mape     mdape     smape  \
9  15 days  117.399119  10.835087  8.187891  0.051556  0.045824  0.052121
10 16 days  107.677308  10.376768  7.761311  0.048644  0.038990  0.049408
11 17 days  112.533556  10.608183  8.219773  0.051255  0.046193  0.051881
12 18 days  125.547430  11.204795  8.723264  0.054319  0.040077  0.055064
13 21 days  126.361746  11.241074  8.453962  0.052765  0.036434  0.053991

    coverage
9   0.467532
10  0.472141
11  0.456061
12  0.461877
13  0.465152
```

```
INFO:prophet:Making 38 forecasts with cutoffs between 2020-02-03 00:00:00 and 2021-07-05
00:00:00
```

```
Training:  180 days
Testing:  7 days
   horizon         mse       rmse       mae      mape     mdape     smape  \
9  15 days   97.061294   9.851969  7.470586  0.046793  0.036561  0.047422
10 16 days   95.964075   9.796126  7.384814  0.045801  0.035601  0.046594
11 17 days   98.829012   9.941278  7.637651  0.047297  0.035192  0.048025
12 18 days  104.474413  10.221273  7.908115  0.049009  0.034305  0.049573
13 21 days  120.706763  10.986663  8.463242  0.053049  0.042175  0.053868

    coverage
9   0.529381
10  0.499043
11  0.536722
12  0.534909
13  0.512420
```

```
INFO:prophet:Making 26 forecasts with cutoffs between 2020-01-27 00:00:00 and 2021-07-05
00:00:00

Training:  180 days
Testing:  14 days
   horizon         mse       rmse       mae      mape     mdape     smape  \
9  15 days  106.235293  10.307051  7.621592  0.047221  0.038675  0.048214
10 16 days   99.925011   9.996250  7.287977  0.044940  0.041162  0.045837
11 17 days   94.911785   9.742268  7.385587  0.045408  0.037709  0.046131
12 18 days   95.668845   9.781045  7.615250  0.047578  0.045270  0.048043
13 21 days  111.272198  10.548564  8.345758  0.053056  0.040783  0.053447

    coverage
9   0.495868
10  0.510048
11  0.535885
12  0.514005
13  0.463636


Training:  180 days
Testing:  21 days
   horizon         mse       rmse       mae      mape     mdape     smape  \
9  15 days  106.948632  10.341597  7.699360  0.048621  0.038675  0.048912
10 16 days   92.048836   9.594208  7.187669  0.045025  0.031477  0.045554
11 17 days   98.530864   9.926271  7.623483  0.047451  0.031477  0.047887
12 18 days  110.230724  10.499082  8.091080  0.050412  0.033079  0.050916
13 21 days   99.104652   9.955132  7.477914  0.047015  0.034872  0.047863

    coverage
9   0.539414
10  0.511435
11  0.500000
12  0.513514
13  0.548108
```

In [46]:

```
results_AXP=tt_test_p('AXP',trains,tests)
```

```
INFO:fbprophet:Disabling yearly seasonality. Run prophet with yearly_seasonality=True to
override this.
INFO:fbprophet:Disabling daily seasonality. Run prophet with daily_seasonality=True to ov
erride this.

Processing:  yfinance.Ticker object <AXP>

INFO:prophet:Making 97 forecasts with cutoffs between 2019-09-02 00:00:00 and 2021-07-05
00:00:00

INFO:fbprophet:n_changepoints greater than number of observations. Using 19.
INFO:fbprophet:n_changepoints greater than number of observations. Using 23.
INFO:prophet:Making 49 forecasts with cutoffs between 2019-09-02 00:00:00 and 2021-07-05
00:00:00

Training:  30 days
Testing:  7 days
   horizon         mse       rmse       mae      mape     mdape     smape  \
9  15 days  144.369362  12.015380  7.937559  0.079267  0.032190  0.080859
10 16 days  161.329044  12.701537  8.255916  0.083080  0.034841  0.084797
11 17 days  177.940652  13.339440  8.402717  0.085702  0.034841  0.086759
12 18 days  181.562300  13.474506  8.633099  0.087426  0.039523  0.088499
13 21 days  212.133464  14.564802  9.379787  0.095073  0.042663  0.096332

    coverage
9   0.523034
10  0.547570
11  0.550752
12  0.537669
13  0.522593

INFO:fbprophet:n_changepoints greater than number of observations. Using 19.
INFO:prophet:Making 33 forecasts with cutoffs between 2019-09-02 00:00:00 and 2021-07-05
```

INFO:prophet:Making 33 forecasts with cutoffs between 2019-09-02 00:00:00 and 2021-07-05 00:00:00

```
Training:  30 days
Testing:  14 days
   horizon         mse       rmse       mae      mape     mdape     smape  \
9  15 days  148.135507  12.171093  7.872756  0.077844  0.042252  0.079839
10 16 days  172.556799  13.136088  8.174087  0.083591  0.033080  0.084600
11 17 days  175.458554  13.246077  8.112177  0.084446  0.034546  0.084765
12 18 days  184.756423  13.592513  8.528508  0.089081  0.037090  0.089016
13 21 days  218.316969  14.775553  9.257935  0.096572  0.038644  0.096523

    coverage
9   0.552899
10  0.579477
11  0.577465
12  0.546299
13  0.531915
```

INFO:fbprophet:n_changepoints greater than number of observations. Using 19.
INFO:prophet:Making 93 forecasts with cutoffs between 2019-09-30 00:00:00 and 2021-07-05 00:00:00

```
Training:  30 days
Testing:  21 days
   horizon         mse       rmse       mae      mape     mdape     smape  \
9  15 days  152.882592  12.364570  8.505111  0.084849  0.046267  0.086016
10 16 days  141.638309  11.901189  8.264933  0.080857  0.045835  0.084386
11 17 days  135.183930  11.626862  7.808986  0.076689  0.038036  0.080377
12 18 days  151.097924  12.292190  8.121638  0.080683  0.038036  0.084503
13 21 days  185.208784  13.609143  9.075167  0.091176  0.041363  0.095391

    coverage
9   0.500709
10  0.568665
11  0.581117
12  0.545455
13  0.532580
```

INFO:prophet:Making 47 forecasts with cutoffs between 2019-09-30 00:00:00 and 2021-07-05 00:00:00

```
Training:  60 days
Testing:  7 days
   horizon         mse       rmse       mae      mape     mdape     smape  \
9  15 days  150.391556  12.263424  8.186059  0.081869  0.049688  0.083524
10 16 days  167.932397  12.958873  8.496005  0.085647  0.033589  0.087428
11 17 days  185.368517  13.615011  8.659401  0.088470  0.035517  0.089569
12 18 days  189.470868  13.764842  8.933823  0.090581  0.040094  0.091698
13 21 days  221.524957  14.883714  9.709794  0.098557  0.047357  0.099860

    coverage
9   0.527656
10  0.557134
11  0.556503
12  0.539364
13  0.516197
```

INFO:prophet:Making 31 forecasts with cutoffs between 2019-10-14 00:00:00 and 2021-07-05 00:00:00

```
Training:  60 days
Testing:  14 days
   horizon         mse       rmse       mae      mape     mdape     smape  \
9  15 days  154.158486  12.416058  8.089631  0.080146  0.038473  0.082208
10 16 days  179.251938  13.388500  8.361250  0.085744  0.038787  0.086776
11 17 days  182.458644  13.507725  8.293369  0.086600  0.033078  0.086924
12 18 days  192.497646  13.874352  8.771851  0.091833  0.044563  0.091758
13 21 days  227.252998  15.074913  9.513218  0.099489  0.033902  0.099412

    coverage
9   0.555237
```

```
10  0.597622
11  0.602941
12  0.555882
13  0.518627
```

```
INFO:prophet:Making 76 forecasts with cutoffs between 2020-01-27 00:00:00 and 2021-07-05
00:00:00
```

```
Training:  60 days
Testing:  21 days
   horizon         mse       rmse       mae      mape     mdape     smape  \
9  15 days  162.022589  12.728809  8.824706  0.088317  0.048394  0.089545
10 16 days  149.849616  12.241308  8.570642  0.084087  0.046863  0.087818
11 17 days  143.228218  11.967799  8.132377  0.080061  0.037563  0.083961
12 18 days  160.589820  12.672404  8.504751  0.084663  0.039700  0.088716
13 21 days  196.940193  14.033538  9.479702  0.095531  0.028937  0.099996

    coverage
9   0.534091
10  0.583578
11  0.563636
12  0.538123
13  0.534848
```

```
INFO:prophet:Making 38 forecasts with cutoffs between 2020-02-03 00:00:00 and 2021-07-05
00:00:00
```

```
Training:  180 days
Testing:  7 days
   horizon         mse       rmse        mae      mape     mdape     smape  \
9  15 days  181.423698  13.469361   9.399852  0.095019  0.052589  0.096981
10 16 days  200.257935  14.151252   9.675323  0.098503  0.067175  0.100596
11 17 days  221.355948  14.878036   9.835307  0.101658  0.055360  0.102919
12 18 days  229.541938  15.150642  10.231556  0.105146  0.041140  0.106427
13 21 days  268.983031  16.400702  11.195044  0.115007  0.051363  0.116529

    coverage
9   0.543083
10  0.570813
11  0.584211
12  0.580727
13  0.542510
```

```
INFO:prophet:Making 26 forecasts with cutoffs between 2020-01-27 00:00:00 and 2021-07-05
00:00:00
```

```
Training:  180 days
Testing:  14 days
   horizon         mse       rmse        mae      mape     mdape     smape  \
9  15 days  188.504648  13.729699   9.461738  0.094571  0.037106  0.097059
10 16 days  216.980745  14.730266   9.706716  0.100497  0.041013  0.101682
11 17 days  219.137797  14.803304   9.549325  0.100825  0.053373  0.101140
12 18 days  234.815239  15.323682  10.144251  0.107814  0.043962  0.107621
13 21 days  279.228843  16.710142  11.028881  0.117149  0.036950  0.116974

    coverage
9   0.559229
10  0.597129
11  0.634450
12  0.587715
13  0.573737


Training:  180 days
Testing:  21 days
   horizon         mse       rmse        mae      mape     mdape     smape  \
9  15 days  190.092640  13.787409   9.967261  0.100499  0.058849  0.101911
10 16 days  172.447831  13.131939   9.505450  0.093900  0.047891  0.098211
11 17 days  167.058279  12.925103   9.077842  0.090139  0.044952  0.094680
12 18 days  188.991149  13.747405   9.496460  0.095695  0.041363  0.100449
13 21 days  234.672015  15.319008  10.840796  0.110316  0.074291  0.115607
```

```
    coverage
9   0.554054
10  0.603950
11  0.642412
12  0.617464
13  0.551351
```

In [47]:

```
results_AAPL=tt_test_p('AAPL',trains,tests)
```

```
INFO:fbprophet:Disabling yearly seasonality. Run prophet with yearly_seasonality=True to
override this.
INFO:fbprophet:Disabling daily seasonality. Run prophet with daily_seasonality=True to ov
erride this.
```

```
Processing:  yfinance.Ticker object <AAPL>
```

```
INFO:prophet:Making 97 forecasts with cutoffs between 2019-09-02 00:00:00 and 2021-07-05
00:00:00
```

```
INFO:fbprophet:n_changepoints greater than number of observations. Using 19.
INFO:fbprophet:n_changepoints greater than number of observations. Using 23.
INFO:prophet:Making 49 forecasts with cutoffs between 2019-09-02 00:00:00 and 2021-07-05
00:00:00
```

```
Training:  30 days
Testing:  7 days
   horizon         mse       rmse       mae      mape     mdape     smape  \
9  15 days   91.161242   9.547840  7.533231  0.076999  0.062010  0.077448
10 16 days   97.392013   9.868739  7.739889  0.078440  0.067427  0.078868
11 17 days  104.926664  10.243372  8.048825  0.082479  0.053479  0.082713
12 18 days  110.730730  10.522867  8.185084  0.084839  0.066646  0.084962
13 21 days  121.565221  11.025662  8.633828  0.090164  0.066205  0.090282

    coverage
9   0.303531
10  0.335641
11  0.315038
12  0.301053
13  0.303727
```

```
INFO:fbprophet:n_changepoints greater than number of observations. Using 19.
INFO:prophet:Making 33 forecasts with cutoffs between 2019-09-02 00:00:00 and 2021-07-05
00:00:00
```

```
Training:  30 days
Testing:  14 days
   horizon         mse       rmse       mae      mape     mdape     smape  \
9  15 days   84.052630   9.168022  7.284547  0.074702  0.061041  0.075690
10 16 days   90.813322   9.529602  7.522705  0.076741  0.057568  0.077518
11 17 days   99.980688   9.999034  8.063382  0.082353  0.064846  0.083133
12 18 days  113.804634  10.667925  8.438426  0.087604  0.065909  0.087892
13 21 days  135.036355  11.620514  9.148186  0.095586  0.066190  0.095515

    coverage
9   0.276777
10  0.297212
11  0.246479
12  0.240935
13  0.241235
```

```
INFO:fbprophet:n_changepoints greater than number of observations. Using 19.
INFO:prophet:Making 93 forecasts with cutoffs between 2019-09-30 00:00:00 and 2021-07-05
00:00:00
```

```
Training:  30 days
Testing:  21 days
   horizon         mse       rmse       mae      mape     mdape     smape  \
9  15 days   80.304315   8.961268  7.307248  0.075624  0.065872  0.076113
10 16 days   79.684421   8.926613  7.222763  0.073578  0.058486  0.074504
```

```
11 17 days   88.077203   9.384946  7.481041  0.075664  0.062142  0.076783
12 18 days  103.954272  10.195797  7.855620  0.080186  0.055380  0.081320
13 21 days  115.724432  10.757529  8.319556  0.085444  0.067186  0.087007

    coverage
9   0.290780
10  0.299807
11  0.284574
12  0.299807
13  0.283245
```

```
INFO:prophet:Making 47 forecasts with cutoffs between 2019-09-30 00:00:00 and 2021-07-05
00:00:00
```

```
Training:  60 days
Testing:  7 days
   horizon         mse       rmse       mae      mape     mdape     smape  \
9  15 days   94.959375   9.744710  7.782065  0.078861  0.062010  0.079317
10 16 days  101.286972  10.064143  7.971594  0.079920  0.058027  0.080371
11 17 days  109.168280  10.448363  8.299680  0.084219  0.062217  0.084476
12 18 days  115.455790  10.745036  8.467386  0.087094  0.065141  0.087220
13 21 days  126.941645  11.266838  8.939798  0.092630  0.069225  0.092738

    coverage
9   0.302283
10  0.317044
11  0.299820
12  0.284648
13  0.290451
```

```
INFO:prophet:Making 31 forecasts with cutoffs between 2019-10-14 00:00:00 and 2021-07-05
00:00:00
```

```
Training:  60 days
Testing:  14 days
   horizon         mse       rmse       mae      mape     mdape     smape  \
9  15 days   87.357575   9.346527  7.473080  0.075549  0.056806  0.076548
10 16 days   94.075775   9.699267  7.681068  0.076997  0.058760  0.077811
11 17 days  103.670181  10.181855  8.242024  0.082730  0.062134  0.083568
12 18 days  118.388648  10.880655  8.671761  0.088899  0.067910  0.089197
13 21 days  140.582276  11.856740  9.414480  0.097272  0.066631  0.097175

    coverage
9   0.289096
10  0.310075
11  0.257353
12  0.251634
13  0.251961
```

```
INFO:prophet:Making 76 forecasts with cutoffs between 2020-01-27 00:00:00 and 2021-07-05
00:00:00
```

```
Training:  60 days
Testing:  21 days
   horizon         mse       rmse       mae      mape     mdape     smape  \
9  15 days   85.150392   9.227697  7.631006  0.077670  0.066400  0.078124
10 16 days   84.379420   9.185827  7.523253  0.075234  0.067944  0.076150
11 17 days   93.393758   9.664045  7.818838  0.077863  0.057028  0.078990
12 18 days  110.457649  10.509883  8.233022  0.082903  0.065219  0.084048
13 21 days  123.316756  11.104808  8.745368  0.088583  0.062909  0.090172

    coverage
9   0.299513
10  0.296921
11  0.270455
12  0.286657
13  0.268182
```

```
INFO:prophet:Making 38 forecasts with cutoffs between 2020-02-03 00:00:00 and 2021-07-05
00:00:00
```

```
Training:  180 days
Testing:  7 days
   horizon         mse       rmse        mae      mape     mdape     smape  \
9  15 days  114.138463  10.683560   8.915720  0.087632  0.066061  0.088185
10 16 days  120.489949  10.976791   9.117296  0.089181  0.076239  0.089740
11 17 days  129.836141  11.394566   9.473221  0.093833  0.075321  0.094092
12 18 days  139.037676  11.791424   9.688341  0.096781  0.089967  0.096846
13 21 days  152.930507  12.366507  10.208396  0.102552  0.087239  0.102624

    coverage
9   0.343742
10  0.310766
11  0.296292
12  0.313091
13  0.320743
```

```
INFO:prophet:Making 26 forecasts with cutoffs between 2020-01-27 00:00:00 and 2021-07-05
00:00:00
```

```
Training:  180 days
Testing:  14 days
   horizon         mse       rmse        mae      mape     mdape     smape  \
9  15 days  105.915841  10.291542   8.591393  0.083924  0.061428  0.085192
10 16 days  113.100673  10.634880   8.833780  0.086296  0.064846  0.087310
11 17 days  124.020052  11.136429   9.451284  0.092582  0.064846  0.093556
12 18 days  143.684568  11.986850  10.002018  0.099639  0.081322  0.099915
13 21 days  171.606097  13.099851  10.873775  0.108951  0.068349  0.108726

    coverage
9   0.330579
10  0.315789
11  0.248804
12  0.256511
13  0.277778


Training:  180 days
Testing:  21 days
   horizon         mse       rmse       mae      mape     mdape     smape  \
9  15 days   99.373294   9.968615  8.588606  0.085364  0.076971  0.085874
10 16 days   97.336710   9.865937  8.433699  0.082773  0.064846  0.083865
11 17 days  108.691985  10.425545  8.756846  0.085020  0.065909  0.086309
12 18 days  130.045188  11.403736  9.275773  0.090768  0.081322  0.092026
13 21 days  146.837301  12.117644  9.948414  0.098008  0.075448  0.099771

    coverage
9   0.315315
10  0.334719
11  0.319127
12  0.316008
13  0.310270
```

In [48]:

```
results_AMGN=tt_test_p('AMGN',trains,tests)
```

```
INFO:fbprophet:Disabling yearly seasonality. Run prophet with yearly_seasonality=True to
override this.
INFO:fbprophet:Disabling daily seasonality. Run prophet with daily_seasonality=True to ov
erride this.
```

```
Processing:  yfinance.Ticker object <AMGN>
```

```
INFO:prophet:Making 97 forecasts with cutoffs between 2019-09-03 00:00:00 and 2021-07-06
00:00:00
```

```
INFO:fbprophet:n_changepoints greater than number of observations. Using 19.
INFO:fbprophet:n_changepoints greater than number of observations. Using 23.
INFO:prophet:Making 49 forecasts with cutoffs between 2019-09-03 00:00:00 and 2021-07-06
00:00:00
```

```
Training:  30 days
Testing:  7 days
```

Testing: 7 days

	horizon	mse	rmse	mae	mape	mdape	smape	coverage
9	15 days	210.381650	14.504539	11.316671	0.050237	0.040550	0.050593	0.452430
10	16 days	238.651279	15.448342	12.148064	0.054184	0.044371	0.054582	0.392105
11	17 days	252.587655	15.893006	12.580768	0.056119	0.047504	0.056418	0.390602
12	20 days	290.460392	17.042899	13.335359	0.059548	0.049239	0.060138	0.403649
13	21 days	295.261254	17.183168	13.443803	0.059742	0.048664	0.060451	0.424238

INFO:fbprophet:n_changepoints greater than number of observations. Using 19.
INFO:prophet:Making 33 forecasts with cutoffs between 2019-09-03 00:00:00 and 2021-07-06 00:00:00

Training: 30 days
Testing: 14 days

	horizon	mse	rmse	mae	mape	mdape	smape	coverage
9	15 days	212.053531	14.562058	11.393647	0.050676	0.037737	0.050930	0.489221
10	16 days	245.621697	15.672323	12.143010	0.054065	0.044120	0.054470	0.401408
11	17 days	268.074251	16.372973	12.781424	0.057227	0.045092	0.057553	0.375787
12	20 days	303.129319	17.410609	13.607157	0.061125	0.051073	0.061538	0.382979
13	21 days	300.835415	17.344608	13.536050	0.060405	0.045296	0.060888	0.428529

INFO:fbprophet:n_changepoints greater than number of observations. Using 19.
INFO:prophet:Making 93 forecasts with cutoffs between 2019-10-01 00:00:00 and 2021-07-06 00:00:00

Training: 30 days
Testing: 21 days

	horizon	mse	rmse	mae	mape	mdape	smape	coverage
9	15 days	181.224012	13.461947	10.388174	0.046598	0.037465	0.046696	0.485493
10	16 days	197.927933	14.068686	10.701296	0.047879	0.039918	0.048155	0.446809
11	17 days	230.434798	15.180079	11.554360	0.051412	0.040680	0.051858	0.475822
12	20 days	277.772493	16.666508	12.566933	0.056274	0.048274	0.056835	0.436170
13	21 days	276.153799	16.617876	12.865214	0.057570	0.044564	0.058026	0.480851

INFO:prophet:Making 47 forecasts with cutoffs between 2019-10-01 00:00:00 and 2021-07-06 00:00:00

Training: 60 days
Testing: 7 days

	horizon	mse	rmse	mae	mape	mdape	smape	coverage
9	15 days	213.479132	14.610925	11.353894	0.049974	0.039947	0.050387	0.446317
10	16 days	241.973805	15.555507	12.166098	0.053797	0.044782	0.054243	0.421109
11	17 days	255.284342	15.977620	12.553776	0.055482	0.042986	0.055818	0.412006
12	20 days	292.679030	17.107865	13.260209	0.058633	0.049879	0.059250	0.418250
13	21 days	295.921922	17.202381	13.329741	0.058637	0.046164	0.059353	

```
13  0.443020
```

```
INFO:prophet:Making 31 forecasts with cutoffs between 2019-10-15 00:00:00 and 2021-07-06
00:00:00
```

```
Training:  60 days
Testing:  14 days
   horizon         mse       rmse        mae      mape     mdape     smape  \
9  15 days  212.277768  14.569755  11.339742  0.049917  0.042191  0.050282
10 16 days  246.516185  15.700834  12.072528  0.053178  0.038828  0.053695
11 17 days  268.644271  16.390371  12.671269  0.056136  0.046833  0.056559
12 20 days  302.982623  17.406396  13.449693  0.059785  0.049879  0.060274
13 21 days  299.471758  17.305252  13.341738  0.058859  0.043882  0.059411

    coverage
9   0.488736
10  0.441176
11  0.422222
12  0.407516
13  0.432353
```

```
INFO:prophet:Making 76 forecasts with cutoffs between 2020-01-28 00:00:00 and 2021-07-06
00:00:00
```

```
Training:  60 days
Testing:  21 days
   horizon         mse       rmse        mae      mape     mdape     smape  \
9  15 days  177.439347  13.320636  10.105360  0.044485  0.037855  0.044712
10 16 days  193.481025  13.909746  10.363979  0.045471  0.040640  0.045865
11 17 days  226.428664  15.047547  11.210858  0.048940  0.032130  0.049507
12 20 days  273.096693  16.525637  12.159529  0.053449  0.043563  0.054108
13 21 days  267.435067  16.353442  12.380942  0.054380  0.040147  0.054879

    coverage
9   0.494868
10  0.511364
11  0.505132
12  0.454545
13  0.490260
```

```
INFO:prophet:Making 38 forecasts with cutoffs between 2020-02-04 00:00:00 and 2021-07-06
00:00:00
```

```
Training:  180 days
Testing:  7 days
   horizon         mse       rmse        mae      mape     mdape     smape  \
9  15 days  236.847121  15.389838  11.892068  0.051854  0.044047  0.052354
10 16 days  270.876830  16.458336  12.780375  0.056071  0.050680  0.056610
11 17 days  279.347687  16.713698  12.947991  0.056669  0.045017  0.057208
12 20 days  317.880647  17.829208  13.621484  0.059576  0.048427  0.060509
13 21 days  319.547118  17.875881  13.586512  0.059113  0.044984  0.060104

    coverage
9   0.508134
10  0.477273
11  0.474545
12  0.451216
13  0.488933
```

```
INFO:prophet:Making 26 forecasts with cutoffs between 2020-01-28 00:00:00 and 2021-07-06
00:00:00
```

```
Training:  180 days
Testing:  14 days
   horizon         mse       rmse        mae      mape     mdape     smape  \
9  15 days  234.214454  15.304067  11.697205  0.050925  0.040879  0.051364
10 16 days  275.508988  16.598463  12.516005  0.054590  0.043303  0.055198
11 17 days  295.706906  17.196131  12.982092  0.056947  0.042561  0.057560
12 20 days  332.412504  18.232183  13.874050  0.061056  0.049914  0.061845
13 21 days  326.346411  18.065061  13.683571  0.059681  0.044673  0.060486
```

```
    coverage
9   0.560766
10  0.526316
11  0.486486
12  0.435859
13  0.474242

Training:  180 days
Testing:  21 days
   horizon         mse       rmse        mae      mape     mdape     smape  \
9  15 days  196.813477  14.029023  10.543567  0.046226  0.035674  0.046530
10 16 days  214.173353  14.634663  10.671314  0.046578  0.041362  0.047096
11 17 days  239.258672  15.467989  11.113101  0.048088  0.029545  0.048940
12 20 days  288.837238  16.995212  12.104710  0.052765  0.041185  0.053820
13 21 days  281.839450  16.788075  12.384649  0.054004  0.040655  0.054860

    coverage
9   0.565489
10  0.603950
11  0.632017
12  0.566486
13  0.560517
```

After looking over these results, it seems unlike ARIMA, Prophet likes the 180/7 split. Although it doesn't matter that much when using Prophet; it automatically takes all the data available and uses it. The train/test split is mainly used here for cross-validation.

Putting it all together

So let's make us a function! Ultimately, we'll want it to be able to select the best of 4 stocks input by the user. As always, let's start simple.

In [49]:

```
results_AMGN.tail()
```

Out[49]:

	ds	trend	yhat_lower	yhat_upper	trend_lower	trend_upper	additive_terms	additive_terms_lower	additive_terms
520	2021-08-13	248.486135	238.649420	261.246141	248.076161	248.865551	1.974628	1.974628	1
521	2021-08-14	248.569661	232.082666	255.521140	248.087700	249.042352	-4.464480	-4.464480	-4
522	2021-08-15	248.653187	232.641652	255.285831	248.117977	249.173462	-4.464480	-4.464480	-4
523	2021-08-16	248.736714	239.067346	261.492681	248.119817	249.314164	1.658290	1.658290	1
524	2021-08-17	248.820240	239.033445	261.719468	248.143942	249.457807	1.932887	1.932887	1

In [50]:

```
pred_cols=['ds','yhat']
```

In [51]:

```
AMGN_preds=results_AMGN[pred_cols]
```

In [52]:

```
AMGN_preds.tail()
```

Out[52]:

	ds	yhat
520	2021-08-13	250.460762
521	2021-08-14	244.105181
522	2021-08-15	244.188707
523	2021-08-16	250.395004
524	2021-08-17	250.753127

In [53]:

```
AMGN_preds=AMGN_preds.tail(21)
```

In [54]:

```
AMGN_preds.head()
```

Out[54]:

	ds	yhat
504	2021-07-28	248.929450
505	2021-07-29	248.816662
506	2021-07-30	249.291396
507	2021-07-31	242.935814
508	2021-08-01	243.019341

In [55]:

```
AMGN_preds.reset_index(level=0, drop=True,inplace=True)
```

In [56]:

```
AMGN_preds.head()
```

Out[56]:

	ds	yhat
0	2021-07-28	248.929450
1	2021-07-29	248.816662
2	2021-07-30	249.291396
3	2021-07-31	242.935814
4	2021-08-01	243.019341

In [57]:

```
AMGN_preds['yhat'][20]
```

Out[57]:

```
250.75312667813517
```

In [58]:

```
AMGN_preds['yhat'][0]
```

Out[58]:

```
248.9294504828349
```

In [59]:

```
change_percent=((AMGN_preds['yhat'][20]-AMGN_preds['yhat'][0])/AMGN_preds['yhat'][0])*10
```

```
change_percent=((AMGN_preds['yhat'][20]-AMGN_preds['yhat'][0])/AMGN_preds['yhat'][0])*10
0
```

In [60]:

```
change_percent=round(change_percent,2)
```

In [61]:

```
change_percent
```

Out[61]:

0.73

In [62]:

```
def pct_change(df):
    """This function will quickly calculate the percentage change from the predictions da
taframe.
    One that has been produced from the earlier Prophet function"""
    df1=df['yhat']
    df1=df1.tail(21)
    df1.reset_index(level=0, drop=True,inplace=True)
    change=((df1[20]-df1[0])/df1[0])*100
    change=round(change, 2)
    if change >0:
        print(f"According to the model, you stand to gain {change}% over the next 21 days
")
    else:
        change_abs=abs(change)
        print(f"According to the model, you stand to lose {change_abs}% over the next 21
days")
    return None
```

In [63]:

```
pct_change(results_AAPL)
```

```
According to the model, you stand to gain 0.72% over the next 21 days
```

In [64]:

```
pct_change(results_AXP)
```

```
According to the model, you stand to gain 2.97% over the next 21 days
```

In [65]:

```
pct_change(results_MMM)
```

```
According to the model, you stand to gain 1.77% over the next 21 days
```

In [66]:

```
pct_change(results_CAT)
```

```
According to the model, you stand to gain 0.48% over the next 21 days
```

In [67]:

```
def fcast(stock):
    """This function will take a stock and perform Prophet modeling on it and return the
forecast dataframe"""
    stock=stock.upper() #make sure the symbol is in uppercase
    prof=Prophet()
    stonk = yf.Ticker(stock)
    df1=stonk.history(period='1y')
    df1=df1['Close']
    df1=df1.to_frame()
    df1.index.names = ['ds']
    df1.columns=['y']
```

```
    df1.reset_index(level=0, inplace=True)
    prof.fit(df1)
    future = prof.make_future_dataframe(periods=21)
    forecast = prof.predict(future)
    #The next 2 lines are disabled for the final function
#    prof.plot(forecast)
 #   plt.show()
    return forecast
```

In [68]:

```
HAL=fcast('HAL')
```

```
INFO:fbprophet:Disabling yearly seasonality. Run prophet with yearly_seasonality=True to
override this.
INFO:fbprophet:Disabling daily seasonality. Run prophet with daily_seasonality=True to ov
erride this.
```

In [69]:

```
HAL.head()
```

Out[69]:

	ds	trend	yhat_lower	yhat_upper	trend_lower	trend_upper	additive_terms	additive_terms_lower	additive_terms_u
0	2020-07-27	15.561352	13.984150	17.510735	15.561352	15.561352	0.124212	0.124212	0.124
1	2020-07-28	15.526206	13.984599	17.547330	15.526206	15.526206	0.172389	0.172389	0.172
2	2020-07-29	15.491059	13.946429	17.635205	15.491059	15.491059	0.299438	0.299438	0.299
3	2020-07-30	15.455913	13.770879	17.314650	15.455913	15.455913	0.140359	0.140359	0.140
4	2020-07-31	15.420767	13.805283	17.336513	15.420767	15.420767	0.129428	0.129428	0.129

In [70]:

```
MMM=fcast('MMM')
```

```
INFO:fbprophet:Disabling yearly seasonality. Run prophet with yearly_seasonality=True to
override this.
INFO:fbprophet:Disabling daily seasonality. Run prophet with daily_seasonality=True to ov
erride this.
```

In [71]:

```
AXP=fcast('AXP')
```

```
INFO:fbprophet:Disabling yearly seasonality. Run prophet with yearly_seasonality=True to
override this.
INFO:fbprophet:Disabling daily seasonality. Run prophet with daily_seasonality=True to ov
erride this.
```

In [72]:

```
def pred_4():
    """This function will prompt the user to input 4 different stock/ETF symbols. It will
then use Prophet
    to forecast the next 21 days of the stock price, and return the values (in percent) o
f potential gain and loss."""
    stocks=[]
    pcts=[]
    stock1,stock2,stock3,stock4=input("Enter 4 stock symbols: ").split(",")
    stocks=[stock1,stock2,stock3,stock4]
    for stock in stocks:
        stock=stock.upper()
```

```
        df=fcast(stock)
        df1=df['yhat']
        df1=df1.tail(21)
        df1.reset_index(level=0, drop=True,inplace=True)
        change=((df1[20]-df1[0])/df1[0])*100
        change=round(change, 2)
        pcts.append(change)
    for i in range(0,4):
        print(f"Stock: {stocks[i].upper()}")
        print(f"Percent Change: {pcts[i]}")
    maxp=pcts.index(max(pcts))
    print()
    print(f'According to the model, {stocks[maxp].upper()} has the highest upside.')
    print("\n" * 3)
    print('FOR ENTERTAINMENT PURPOSES ONLY. This does not substitute for advise from a fi
nancial advisor.')
    print('The creator and affiliates are not responsible for any potential losses. But t
otally responsible for any gains.')
    return None
```

In [73]:

```
pred_4()
```

```
Enter 4 stock symbols: msft,amzn,tsla,spy
```

```
INFO:fbprophet:Disabling yearly seasonality. Run prophet with yearly_seasonality=True to
override this.
INFO:fbprophet:Disabling daily seasonality. Run prophet with daily_seasonality=True to ov
erride this.
INFO:fbprophet:Disabling yearly seasonality. Run prophet with yearly_seasonality=True to
override this.
INFO:fbprophet:Disabling daily seasonality. Run prophet with daily_seasonality=True to ov
erride this.
INFO:fbprophet:Disabling yearly seasonality. Run prophet with yearly_seasonality=True to
override this.
INFO:fbprophet:Disabling daily seasonality. Run prophet with daily_seasonality=True to ov
erride this.
INFO:fbprophet:Disabling yearly seasonality. Run prophet with yearly_seasonality=True to
override this.
INFO:fbprophet:Disabling daily seasonality. Run prophet with daily_seasonality=True to ov
erride this.
```

```
Stock: MSFT
Percent Change: 2.94
Stock: AMZN
Percent Change: 2.28
Stock: TSLA
Percent Change: -0.37
Stock: SPY
Percent Change: 1.25

According to the model, MSFT has the highest upside.



FOR ENTERTAINMENT PURPOSES ONLY. This does not substitute for advise from a financial adv
isor.
The creator and affiliates are not responsible for any potential losses. But totally resp
onsible for any gains.
```

Conclusion:

I like to call this the KISSSS (the Keep it Simple Stock Selector-The Last 'S' is a typo), and it's true to its name. It's simple, it provides an answer that investors can use to make their decisions. Now as the disclaimer states, this should not be the only factor used in making any investment decision; but for someone that doesn't want to do a lot of research into their stock purchases, this will fit them just fine.

In [ ]: