Time Series to observe DAILY temperature variations

Daily temperature prediction using Random Forest

Going one step deeper than a tree, we use in the notebook below

Random Forests : We begin by loading all necessary libraries and paths to read the "pickles" as well as store image for the graph towards the end of our code. The pickles are read and the data is fed into an RF model.

Finally, we have two graphs showing the RF results vs. the fitted model as well as predicted results vs. actuals and test data

In [224]:
```
import warnings
warnings.filterwarnings('ignore')

%run helper_functions.py
%matplotlib inline
```

Create a folder for every run of the Random forest to store images

In [225]:
```
city='Miami' # New_York Atlanta Boston Dallas Houston Miami
analysis_type = 'Enhanced_Signals' # Basic, Inc_Signals, Enhanced_Signals
```

In [226]:
```
EXPERIMENT_DIR, EXPERIMENT_ID  = create_results_perrun()
print(f"Experiment ID: {EXPERIMENT_ID}")
print(f"Path of the results directory:{EXPERIMENT_DIR}")
```

```
Experiment ID: 36
Path of the results directory:../experiment_results/RUN-36
```

Here we are importing the train and test Data from pickle files created through the EDA file

In [227]:
```
X_train = pd.read_pickle(f'{PICKLE_PATH}/X_train_{city}_{analysis_type}.pkl')
Y_train = pd.read_pickle(f'{PICKLE_PATH}/Y_train_{city}_{analysis_type}.pkl')

X_test  = pd.read_pickle(f'{PICKLE_PATH}/X_test_{city}_{analysis_type}.pkl')
Y_test  = pd.read_pickle(f'{PICKLE_PATH}/Y_test_{city}_{analysis_type}.pkl')

print("Shape of Training Dataset " , X_train.shape)
print("Shape of Testing Dataset " , X_test.shape)
```

```
Shape of Training Dataset  (1399, 17)
Shape of Testing Dataset  (90, 17)
```

In [228]:
```
# Fitting a decision tree regressor with max depth and n_estimators
max_depth = 8
n_estimators = 50
fitted_model = RandomForestRegressor(max_depth=max_depth, random_state=0, n_estimators=n_estimators)
fitted_model.fit(X_train, Y_train)

# Dataframe to show features and their importances
top_features = 10
features_importances_df= show_feature_importances(X_train.columns.values.tolist(),
                                                  fitted_model.feature_importances_,top_features)

# Top 10 features
features_importances_df.head(10)

# Store results
features_importances_df.to_csv(f'{EXPERIMENT_DIR}/feature_importances.csv')
```

In [229]:
```
# Run the model on the training dataset
Y_train_pred = fitted_model.predict(X_train)
# Calculate mean squared error for the predicted values
mse_train = mean_squared_error(Y_train, Y_train_pred)
print('Mean Squared Error for the training dataset: %.3f' % mse_train)
```

```
Mean Squared Error for the training dataset: 0.757
```

In [230]:
```
# Run the model on the testing dataset
Y_test_pred = fitted_model.predict(X_test)
# Calculate mean squared error for the test vs predicted values
mse_test = mean_squared_error(Y_test, Y_test_pred)
print('Mean Squared Error for the testing dataset: %.3f' % mse_test)
```

```
Mean Squared Error for the testing dataset: 1.169
```

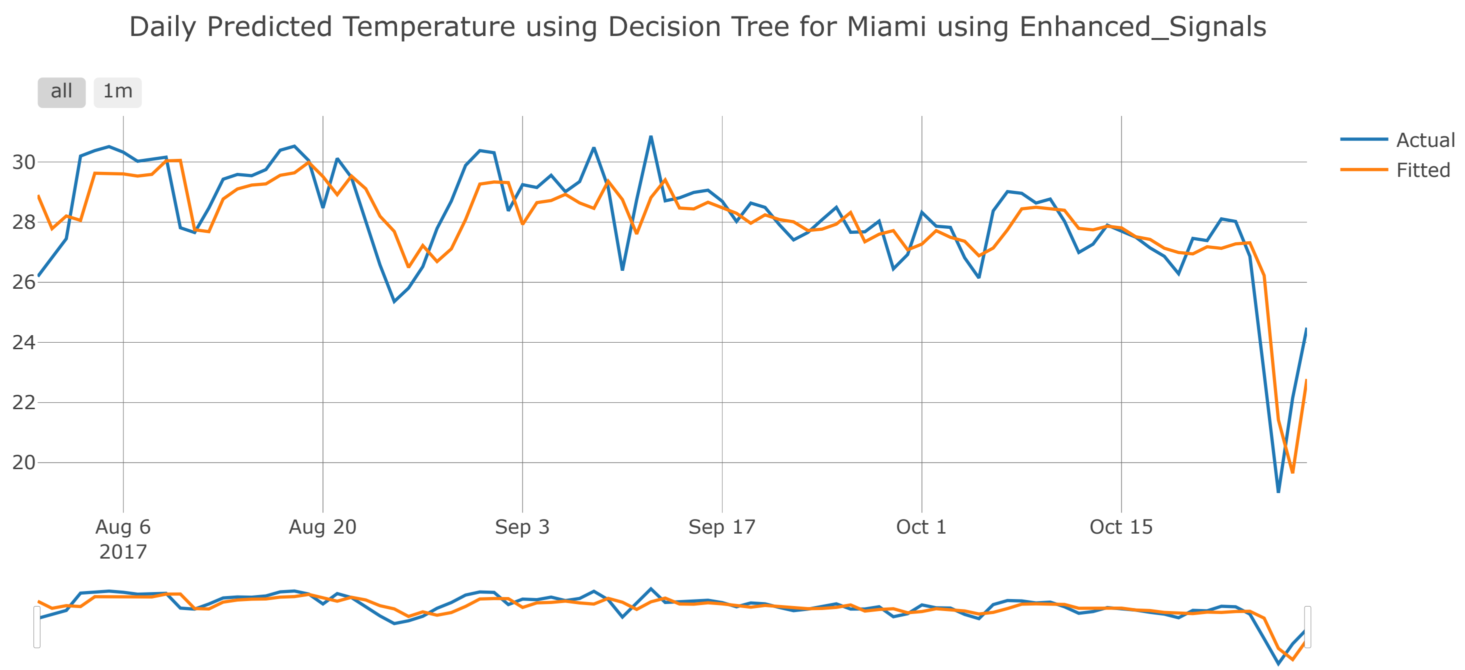
In [231]:

```python
# Creating a dataframe for predicted/fitted values
future_forecast = pd.DataFrame(Y_test_pred,index = Y_test.index,columns=['Fitted'])

# Concatenate the predicted/fitted values with actual values to display graphs
predictions = pd.concat([Y_test,future_forecast],axis=1)
predictions.columns = ["Actual","Fitted"]

# Displaying few of the predicted values
predictions.head(10)
```

Out[231]:

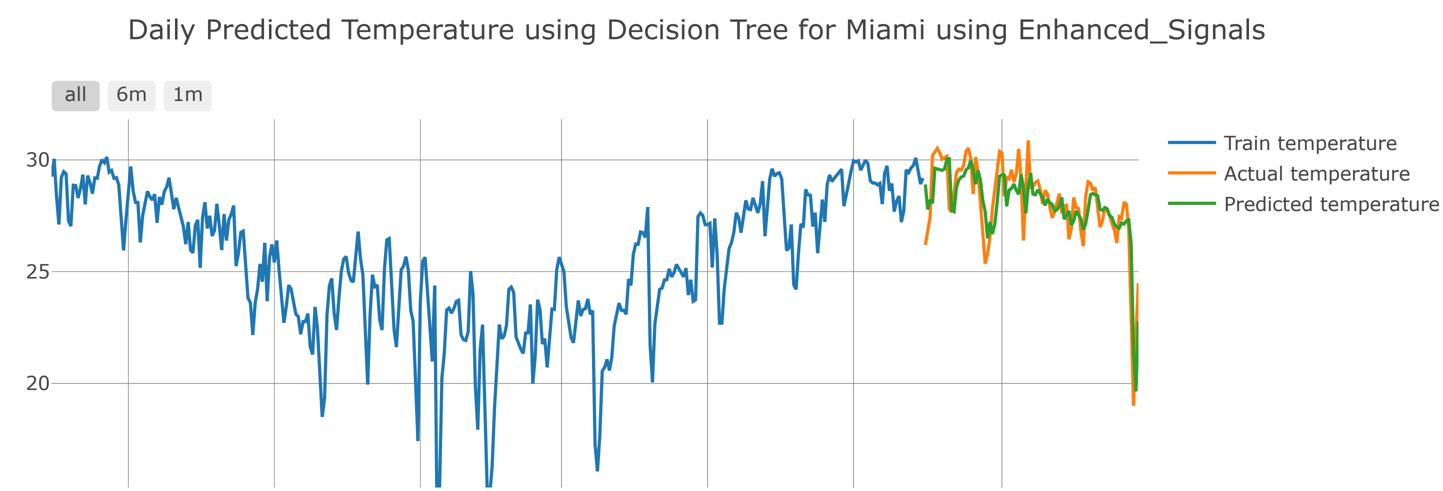
datetime	Actual	Fitted
2017-07-31	26.192917	28.903332
2017-08-01	26.835000	27.783733
2017-08-02	27.449583	28.204148
2017-08-03	30.198333	28.059736
2017-08-04	30.380000	29.627431
2017-08-05	30.513333	29.596322
2017-08-06	30.326667	29.604831
2017-08-07	30.024583	29.532938
2017-08-08	30.094583	29.588812
2017-08-09	30.160833	30.038199

In [232]:
```
city = city.replace('_',' ')
# Plotting the daily predicted temperature vs Actual Temperature - Decision Tree
fig = charter_helper_fitted(f"Daily Predicted Temperature using Decision Tree for {city} using {analysis_type}", predictions)
iplot(fig)

py.image.save_as(fig, f'{EXPERIMENT_DIR}/Daily_actual_vs_predict.png')
```



Export to plot.ly »

In [233]:
```
# Plotting the training data for past year, Actual/test data and predicted temperature - Decision Tree
fig = charter_helper_prediction(f"Daily Predicted Temperature using Decision Tree for {city} using {analysis_type}",
                                X_train,Y_train,X_test,Y_test,future_forecast)

iplot(fig)

py.image.save_as(fig, f'{EXPERIMENT_DIR}/Daily_predict.png')
```



In [234]:
```
results = update_results_function(EXPERIMENT_ID, 'RANDOM FOREST', city, analysis_type,
                                  {'max_depth': max_depth , 'n_estimators':n_estimators, 'Info': X_train._metadata},
                                  {'features' : X_train.columns.values.tolist(),
                                   'importances':fitted_model.feature_importances_,
                                   'mse_train' : mse_train},
                                   mse_test)
```

In [235]:
```
results.tail(1)
```

Out[235]:

	RUN_ID	DATETIME	MODEL_NAME	CITY	FEATURE_TYPE	HOST_MACHINE	MODEL_PARAMETERS	MODEL_RESULTS	MEAN_SQUARED_ERROR
35	36	2018-08-14 19:56:51.638468	RANDOM FOREST	Miami	Enhanced_Signals	DESKTOP-KN40C32	{'max_depth': 8, 'n_estimators': 50, 'Info': {...	{'features': ['temperature_lag1', 'temperature...	1.16879

In [236]:
```python
results = pd.read_pickle('../pickles/results.pkl')
results
```

Out[236]:

	RUN_ID	DATETIME	MODEL_NAME	CITY	FEATURE_TYPE	HOST_MACHINE	MODEL_PARAMETERS	MODEL_RESULTS	MEAN_SQUARED_ERROR
0	1	2018-08-14 19:41:45.275297	DECISION TREE	New York	Basic	DESKTOP-KN40C32	{'max_depth': 8, 'Info': {'feature_set_type': ...	{'features': ['temperature_lag1', 'temperature...	5.799872
1	2	2018-08-14 19:42:13.654060	DECISION TREE	Atlanta	Basic	DESKTOP-KN40C32	{'max_depth': 8, 'Info': {'feature_set_type': ...	{'features': ['temperature_lag1', 'temperature...	8.436149
2	3	2018-08-14 19:42:23.328525	DECISION TREE	Boston	Basic	DESKTOP-KN40C32	{'max_depth': 8, 'Info': {'feature_set_type': ...	{'features': ['temperature_lag1', 'temperature...	16.504857
3	4	2018-08-14 19:42:30.715058	DECISION TREE	Dallas	Basic	DESKTOP-KN40C32	{'max_depth': 8, 'Info': {'feature_set_type': ...	{'features': ['temperature_lag1', 'temperature...	12.523335
4	5	2018-08-14 19:42:41.506483	DECISION TREE	Houston	Basic	DESKTOP-KN40C32	{'max_depth': 8, 'Info': {'feature_set_type': ...	{'features': ['temperature_lag1', 'temperature...	7.705077
		2018-08-14	DECISION			DESKTOP-	{'max_depth': 8, 'Info':	{'features':	

In [ ]: