Experiment Results

In [17]:

```
import warnings
warnings.filterwarnings('ignore')

%run helper_functions.py
%matplotlib inline
```

Here we review the results from all of our experiments across the various ML algorithms we investigated and the datasets we created.

We provide a summary below with a link to each of the notebooks used for the experiments.

Experiment Summary

We created the following datasets as part of our experiment design

Link	Dataset	Cleaned + Lags/Moving Averages	Including Signals	Enhanced Signals
link (DataViewer.ipynb?DATA=Atlanta)	Atlanta	x	x	x
link (DataViewer.ipynb?DATA=Boston)	Boston	x	x	x
link (DataViewer.ipynb?DATA=Dallas)	Dallas	x	x	x
link (DataViewer.ipynb?DATA=Houston)	Houston	x	x	x
link (DataViewer.ipynb?DATA=New_York)	New York	x	x	x
link (DataViewer.ipynb?DATA=Miami)	Miami	x	x	x

We created the following supplementary signals that where used to enhance the datasets above

Link	Dataset	Cleaned	Enhanced with Lags/Moving Averages	Reference
link (SignalViewer.ipynb?DATA=AO)	AO (Artic Oscillation)	x	x	x
link (SignalViewer.ipynb?DATA=NAO)	NAO (North American Oscilliation)	x	x	x
link (SignalViewer.ipynb?DATA=NINO3)	NINO3	x	x	x
link (SignalViewer.ipynb?DATA=NINO4)	NINO4	x	x	x
link (SignalViewer.ipynb?DATA=NINO12)	NINO1/2	x	x	x
link (SignalViewer.ipynb?DATA=NINO34)	NINO3/4	x	x	x

We also prepared other signals and locations to consider as part of our experimental design. We however limited ourselves to these datasets due to time constraints.

We ran the following models as part of our experiment design, these notebooks we used repeatedly to run each variant of the experiments and we included hooks to track the artifacts and results.

Link	Experiments

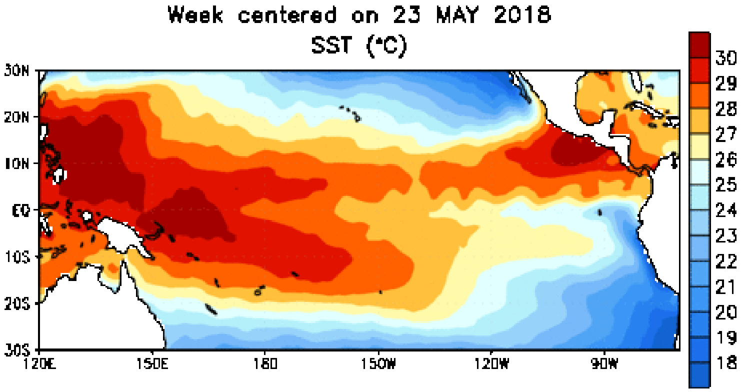
Link	Experiments
link (Daily_Temp_Analysis_ARIMA.ipynb)	ARIMA statistical models and techniques
link (Daily_Temp_Analysis_DT.ipynb)	Decision Tree
link (Daily_Temp_Analysis_RF.ipynb)	Random Forest
link (Daily_Temp_Analysis_RNN.ipynb)	Sequential Recurrent Neural Net

We systematically collected and archived all of the data created by each of the experiments and saved this assigning each experiment its own unique identifier. We also recorded who/when the experiment was run. We also collected various artifacts created in each experiement so we could repeat the experiment at a later date should it be required.

This diagram shows the NINO area of the ocean (courtesy of Climate Prediction Center, National Weather Service) and it shows how the temperature of the ocean is localized.

In [70]:

```
%%HTML
<img src="http://www.cpc.ncep.noaa.gov/products/analysis_monitoring/enso_update/sstanim.gif"></img>
```



Summary of results by location

Here is some quick links to our results

Link	Experiments Results for cities
link (Experiment_Results-Atlanta.ipynb)	Atlanta
link (Experiment_Results-Boston.ipynb)	Boston
link (Experiment_Results-Dallas.ipynb)	Dallas
link (Experiment_Results-Houston.ipynb)	Houston
link (Experiment_Results-New_York.ipynb)	New York
link (Experiment_Results-Miami.ipynb)	Miami

We have provided some pivot tables and charts to allow the reader to see the results of our experiments. We will focus initially on the Mean Squared Error metric as a means of illustrating how each model performed.

ARIMA has been excluded from these comparisons as we were unable to create a forecast that would extend any more than 6 days before the signal reverted to the mean trend.

On investigation, we would need to use an alternative model called SARIMAX which would allow us to extend the forecasts over a longer time period and thus we would be able to make a meaningful comparison.

In [4]:

```
# All runs presented in pivot table
results_df = get_results()
results_df.pivot_table(index=['CITY'], columns=['MODEL_NAME','FEATURE_TYPE'], values="MEAN_SQUARED_ERROR")
```

Out[4]:

MODEL_NAME	DECISION TREE			RANDOM FOREST			RNN		
FEATURE_TYPE	Basic	Enhanced_Signals	Inc_Signals	Basic	Enhanced_Signals	Inc_Signals	Basic	Enhanced_Signals	Inc_Signals
CITY									
Atlanta	8.436149	7.783687	7.992946	5.516615	5.434763	5.434763	5.105819	4.966304	5.628276
Boston	16.504857	25.109055	22.967065	11.877780	12.735153	12.735153	11.869203	11.645510	11.682041
Dallas	12.523335	15.432839	14.223161	8.983822	9.260334	9.260334	8.563057	8.632428	8.336575
Houston	7.705077	10.205057	10.519872	6.637421	6.041626	6.041626	5.581542	4.217506	4.588875
Miami	1.833555	1.556905	1.577366	1.206302	1.168790	1.168790	2.016882	2.012860	2.093093
New York	5.799872	6.767755	6.518000	4.369905	4.216172	4.216172	4.839433	4.095801	4.700272

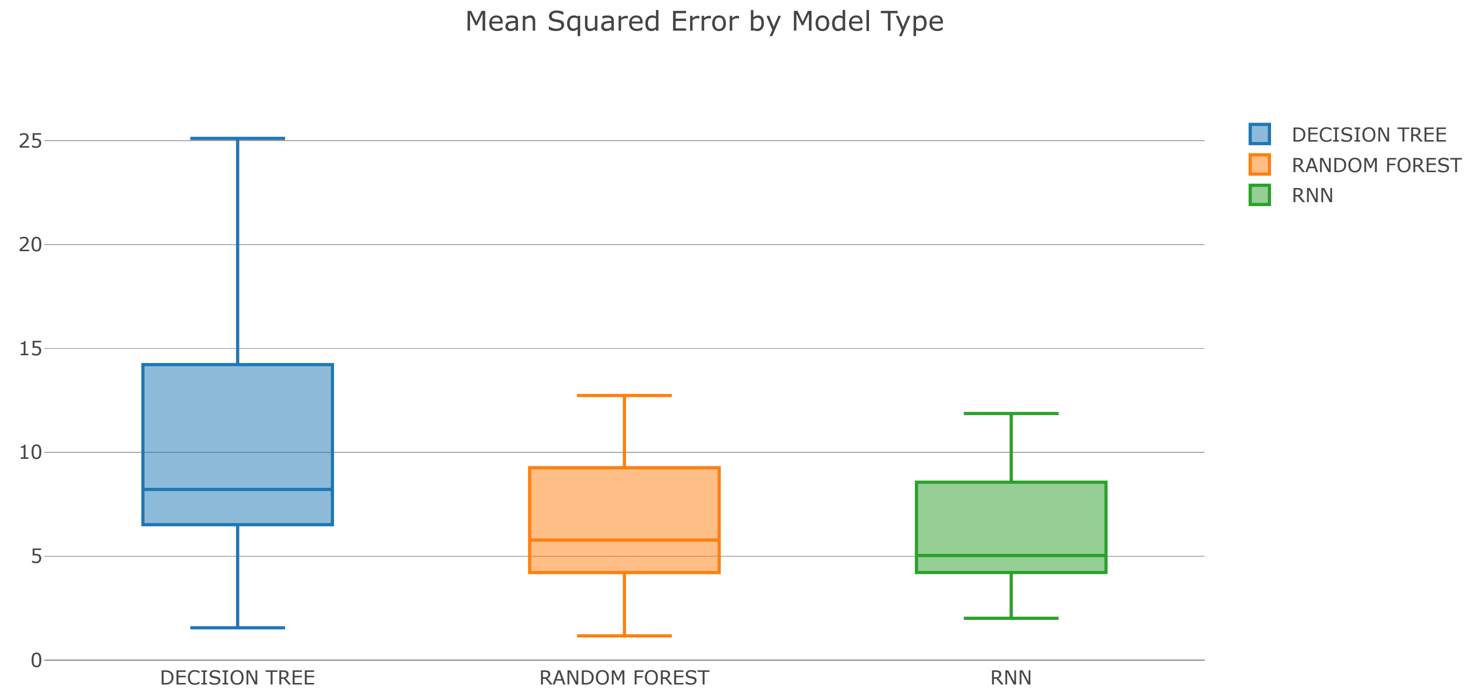
As we can see above, the enhanced signals appear in some instances to improve the forecasts. However this is not always the case, this could be because the cluster of signals choosen should be selected specifically for each location vs. just applying the same data set to each.

We will also review those features flagged in each city's analysis to determine the top 10 features used by the models.

Let's use some boxplots to review the performance of our models over all of our experiments

In [19]:

```
# All runs presented in pivot table
create_boxplot_traces_for_results(results_df,'MODEL_NAME','MEAN_SQUARED_ERROR',"Mean Squared Error by Model Type")
```

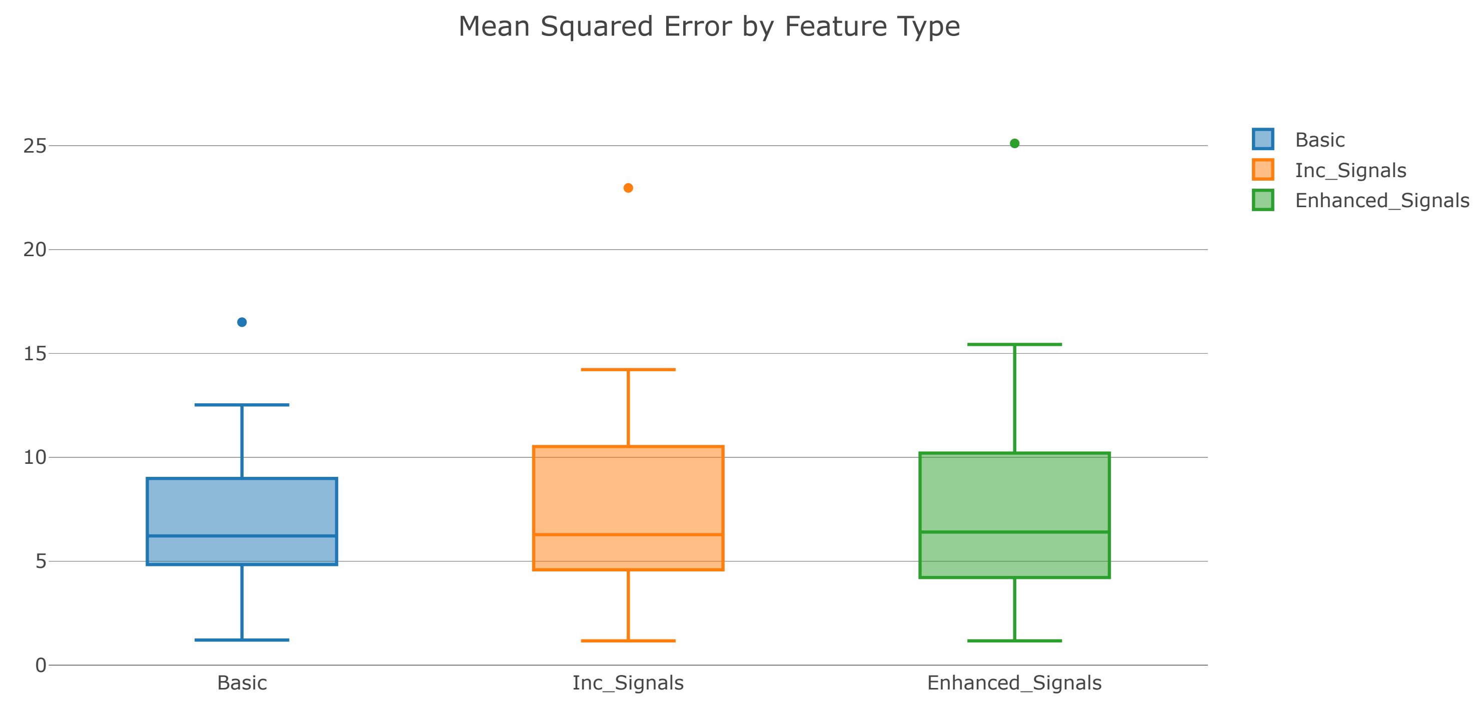

Mean Squared Error by Model Type

Export to plotly.att.com »

As expected the **RNN** performs the best, **Random forest** is the second, **Decision Tree** is the last

In [20]:

```
# All runs presented in pivot table
create_boxplot_traces_for_results(results_df,'FEATURE_TYPE','MEAN_SQUARED_ERROR',"Mean Squared Error by Feature Type")
```



Export to plotly.att.com »

In contrary to what we saw above the signals are not having a big of an effect on our prediction, this may be because we are missing signals or we have not tuned the hyper parameters of the models enough

Conclusion

The Arima model was unsuitable for forecasting as detailed in summary switching to a more advanced model for example SARIMAX or THETA would help us to forecast for a longer period of time

We also incorporated more signals like lags which is signal 1,2,7,30,90 and 365 days ago and moving averages which is signal averaged over 1 week, 30 days, 60 days etc from alternate locations (more locations for SST) and more weather measurement types like AO (Arctic Oscillation), NAO (North Atlantic Oscillation) and their lags and moving averages, also removed signals that are found to have no predictive power.

As per the results detailed in summary RNN performs better of all models and we intend to use LSTM model as a progression from our sequential RNN

Takeaways

- We used GIT as the version control tool, as we reached come to the end of the project we realised being organized with git came really handy with code merges and large csv or pickle file checkins.
- Serializing data, capturing results(pickle files) and images as we ran multiple experiments provided indispensible which helped us to quickly change artifacts / notebooks as we wanted to answer new questions or compare results.
- Notebooks is a great tool for quick analysis but are less efficient when you have to perform multiple iterations with different input parameters and datasets, A hybrid approach is recommended.
- We needed more time to tune our hyperparameters using grid searching or any other approaches.
- Now that we have good amount of results, It would be a good time to get the results reviewed by a domain expert so that we can identify weaknesses in our approach towards data and redefine our next steps, for example adding more signals or removing unnecessary signals, tune hyperparameters of our models.
- We could scale our data and no of experiments if we have faster compute resources.
- We would like to investigate a framework (i.e. MLFlow) or similiar to help manage the end to end process of data capture, running experiences, instrumentation and presenting our results.