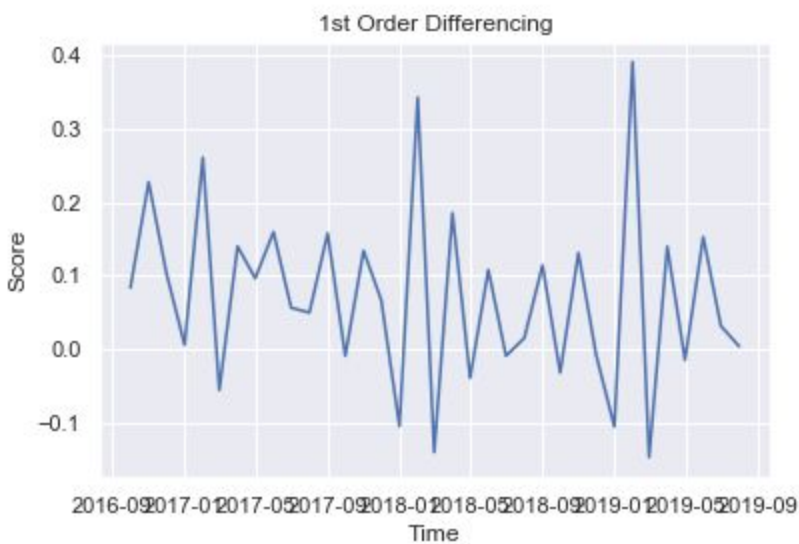


CFO ForeCasting Report

Revenue



Since, there is an uptrend , we need to remove it. Let's see how differencing looks like.



Now that the data is moving sideways , we should take this data to train our models.

I've used some insights given in the problem statement. Then, I applied GridSearchCV to get best parameters. I've found that `n_neurons = 110` and `n_steps = 19` gives the best result. The mape comes out to be 22% here.

Closing Base

Since, all three Products have the same behaviour , I've generalized some parameters for them.

```
model = Sequential()
model.add(LSTM(n_neurons, activation='tanh', return_sequences=True, input_shape=(n_steps,
n_features)))
model.add(Dropout(0.2))
model.add(LSTM(n_neurons, activation='tanh', return_sequences=False,
input_shape=(n_steps, n_features)))
model.add(Dropout(0.2))
model.add(Dense(1))
model.compile(optimizer='adam', loss='mse')
```

After Running Grid Search CV , here are the optimal parameters for Closing Base:

Panther : `n_steps = 5` , `n_neurons = 40` => mape = 10%

Leopard : `n_steps = 3` , `n_neurons = 80` => mape = 27%

Lion : `n_steps = 2` , `n_neurons = 120` => mape = 9%

Net Migrations

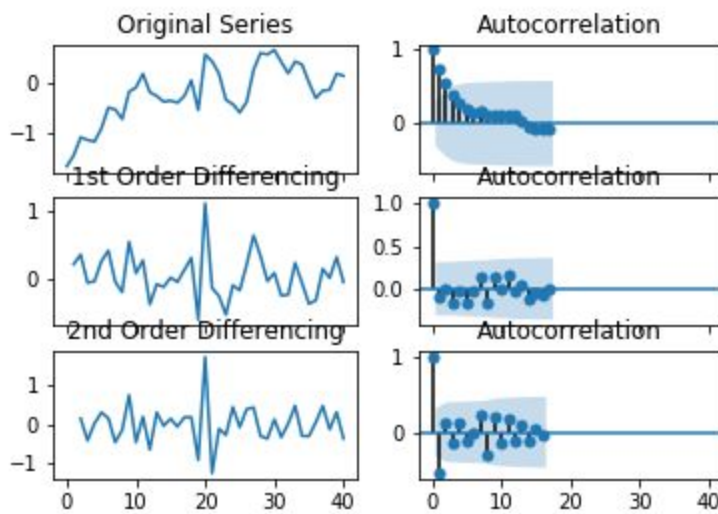
I've used the same configurations as above for this. And Grid Search CV gave me the following results:

Panther : `n_steps = 8` , `n_neurons = 90` => mape = 74%

Leopard : `n_steps = 2` , `n_neurons = 100` => mape = 86%

Lion : `n_steps = 4` , `n_neurons = 100` => mape = 51%

Leavers



The figure above shows you the data trend. We need to make it stationary so I've charted it's differencing and Autocorrelation. As you can see, the 1st order differencing is the best.

The mape I obtained using sarimax was 375%.