Analyzing and Forecasting U.S. Immigration Trends

Group 7: Alan Lin , Trevor Petrin , Ganesh Kumar Ramar , Mingyu Chen

Repository: alanklin/AA-Capstone: Boston College Applied Analytics Capstone Project

While we may be pivoting towards a different project following the conclusion of this one, we will be continuing to submit reports relating to the work we did on our poster. For Week 8, we'd like to dive deeper into our ARIMA models and how we tuned these models for the best average accuracy across all 41 sectors and field offices.

ARIMA models naturally have three parameters that can be adjusted and tuned for better model performance: *p, d, q*. The *p* parameter is for autoregressive terms, or the number of lag terms included in the equation. In autoregression models, the output is the future data point expressed as a linear combination of the past *p* data points. In general, the more lag terms there are, the better the model is at capturing a broader range of temporal dependencies and forecasting based on past values and patterns. However, there is a delicate balance to be met when choosing *p,* as too many terms can over-complicate the model and lead to overfitting. The *d* parameter is for the number of nonseasonal differences needed to achieve stationarity. This is the degree of differencing needed to make a time series stationary, which simplifies modeling and forecasting by ensuring statistical features such as mean and variance remain constant over time. This can help trends and patterns emerge that would not have been possible with the raw data. The differencing order generally never goes beyond 2, even though there is no theoretical upper limit. So, we limit our *d* values to [0, 1, 2]. Finally, the *q* parameter represents the order of the moving average, or the number of lagged forecast errors included in the model. Including past prediction errors (or residuals) into the model helps to capture the random noise or shocks in time series, which can improve prediction accuracy. We don't want to add too much noise to the model; otherwise the model won't be able to capture the underlying patterns.

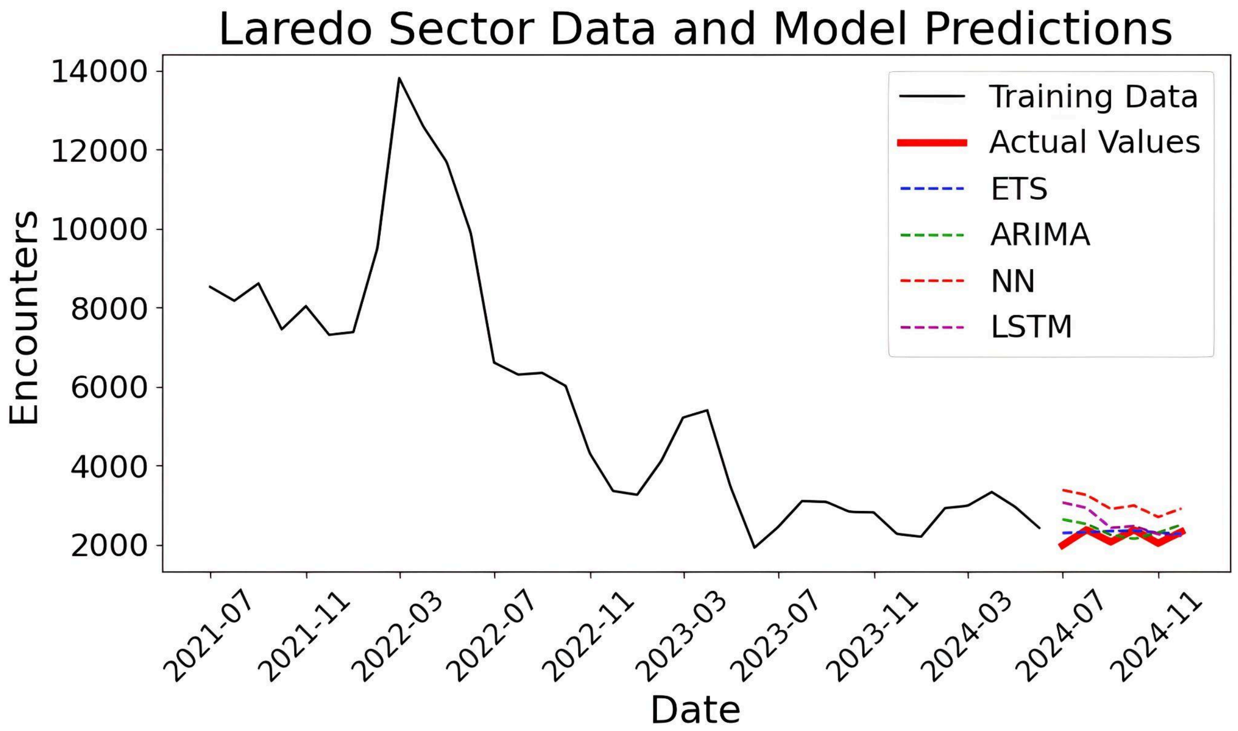
We also took last week's feedback into account as well for trying to implement the GridSearchCV package for the hyperparameter tuning process. That would be much more beneficial to go and redo the Neural Network or the LSTM model as there is a lone parameter, order, for the ARIMA model in the statsmodels package. Because it is a singular parameter, we decided it was better to implement it as a nested for loop rather than creating a dictionary and then passing that into the GridSearchCV because it's only one parameter. So yes, if we had to redo a different model with a different parameter input structure, we would use the GridSearchCV package, however, because of the way statsmodels has their ARIMA implementation structured as a tuple, we will use a for loop instead.

As for the predictive capabilities of the ARIMA model, because this is a multi-step prediction problem with the specifications calling for a 6 month forecast with monthly intervals, we will be using a recursive prediction method for this forecast. A brief intuition into this style of prediction is since the standard ARIMA is reliant upon the error term for the autoregressive portion of the model, which we don't have beyond the train data, this creates a problem for updating the model. To get around this, the model's output from prediction at time step t is used as input for time step t+1 to forecast the next prediction of the model.

ARIMA Tuning Parameters

Autoregressive (*p*)	Differencing (*d*)	Moving Average (*q*)
(0,4)	**(0-2)**	**(0-4)**

For training and tuning the models themselves, a similar approach is taken for training the models separately on each independent sector to get a more targeted estimation of what USCBP can expect for each individual sector. For tuning the models themselves, 41 models were tuned using a grid search to minimize the RMSE on the test data before using the models to forecast. Some of the main benefits of the model that we found included that the ARIMA models were much quicker to train as opposed to the Neural Network and LSTM models used in the past 2 weeks, using much more basic mathematical principles. Additionally, it was much less of a black box method, being able to better understand exactly what parameters were used to best fit the final model. There was also the issue of the Machine Learning models in the past 2 weeks not having a forecasting start point which was even close to the predictions as can be seen in the graph below.



Despite having a solid negative trend, the Neural Network and LSTM models both found a starting point at nearly double the actual value whereas the more basic models of ARIMA and ETS were able to better find an accurate starting point because they focused more upon recent values than overall patterns.

As for the results of the ARIMA relative to the other methods that were selected, we were surprised how much better the traditional forecasting methods outperformed the Machine Learning methods, especially in the first 2 months of predictions. Having a MAPE below 50% for the first 2 months of the data during a difficult season to forecast, and doing so in such a naive way, we were extremely satisfied with the ARIMA results. Considering the model does not take into account external elements such as push/pull migration factors, the reelection of Donald Trump, the continued gang conflict in Mexico, and other different happenings, the ARIMA performed extremely well. We will expand later upon how we would have been able to improve upon this model structure at the end of this report.

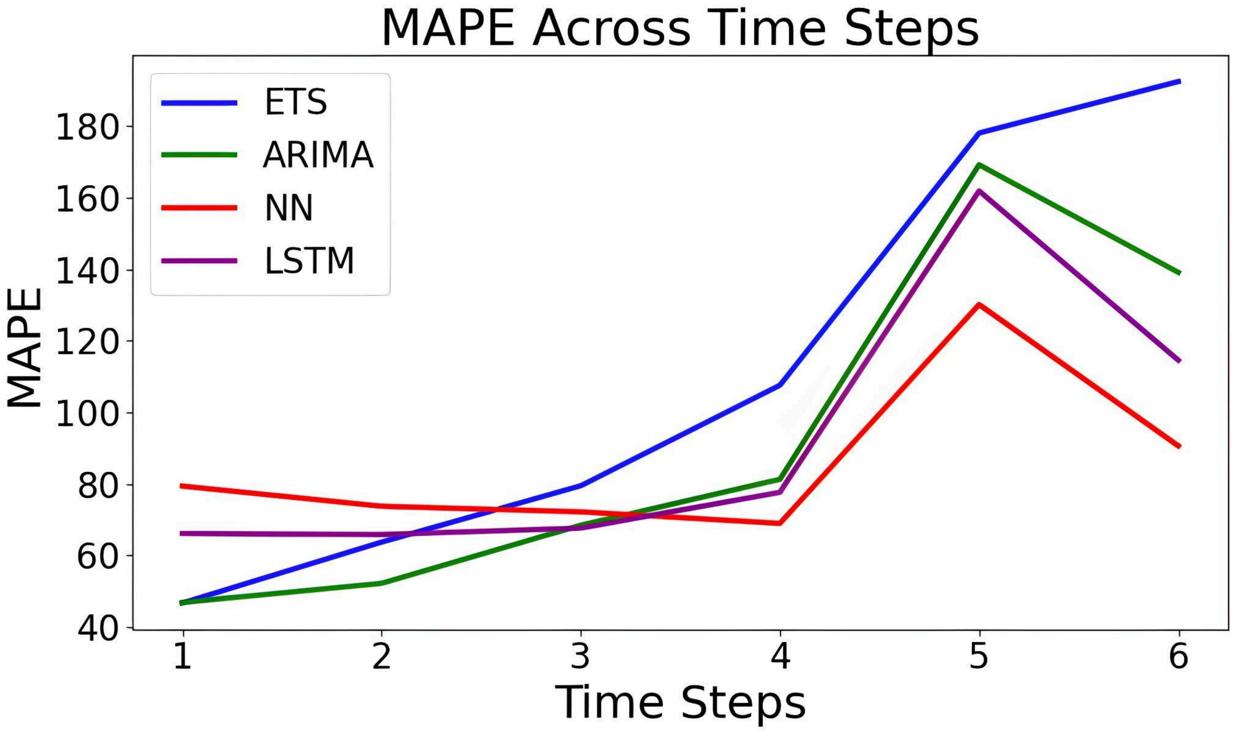
MAPE Average Results

	2 Month	6 Month
ARIMA	**49.6%**	**92.8%**
ETS	**55.3%**	**111.4%**
LSTM	**66.0%**	**92.3%**
Neural Network	**76.6%**	**85.9%**

For the overall MAPE performance across all models, it was straightforward that ARIMA was clearly the best performer for a short-term forecast with the MAPE of below 50% in the first two months of predictions on average across all sectors. It is clear as well that the Machine Learning models did not perform well out of the gate, though were slightly more effective in months 4-6, though there were not really valuable insights offered from those predictions because they were so far off. There was a difference in data with a sharp increase in MAPE at month 5 resulting in the models losing their effectiveness at that point, though the performance across the first 3 months for the ARIMA in particular was very sound.

As for speculating what may have caused the sharp dropoff in month 5, that would correspond with November 2024, when Donald Trump was reelected. President Trump was known for hardline immigration policies which were scrutinized on the global scale for humanitarian concerns such as family separation, inhumane living conditions, and expediting removals from the US without due process. His policies would be a major factor discouraging immigrants, particularly from Central and South America who are considering migrating to the United States. From speculation to what truly happened, further policies have been implemented by the current Trump administration to deter migration such as sending migrants to Guantanamo Bay as well as expediting deportations.

Ironically enough, though former President Joe Biden was known for his more welcoming immigration stance, his administration continued to expedite removing encountered migrants from the US as well as increasing the funding for USCBP and ICE, the two main departments associated with removing individuals from the country.



If we were to try and improve this model using the aforementioned external factors, we would need to take a separate modeling approach which would include a model input outside of a previous observation such as a regressive forecasting model or a neural network with a more diverse input structure than what we were trying to do. As for where we would be able to get more encompassing data, we could use NLP to text mine different articles by the date and associate them with the month in or before the number which the article was written in. This could help account for identifying when people will arrive such as the migrant caravans which were seen several times throughout our training period. Additional components may include migration from countries with known outward migration to the US such as Mexico and Venezuela and use a lag component to know when people will leave and when they will arrive.