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

Introduction

This week's report documents the model packaging process and identifies performance indicators that can be used to monitor the model's performance, validity, and relevance. We will go over risk mitigation strategies for different thresholds of performance metrics, as well as discussing the impact that data drift or concept drift may have on our data and model.

Model Packaging

Because we have a separate ARIMA model for every individual sector, we shall simply create a folder storing each of the pickle files for each model distinguishing every sector by the name of the file.

Environment Dependencies

Our models have been trained and operated under Windows OS. The corresponding Python version we have used is 3.12.7 and the list of packages that were used for our final models are as follows:

pandas : 2.2.3 — storing/data manipulation
matplotlib : 3.9.2 — plotting visualizations
numpy : 1.26.4 — used in our metric MAPE calculations
statsmodels : 0.14.4 — ARIMA models

Model Monitoring

The main method of performance monitoring will be observing the MAPE value over three months of forecasting associated with each separate model. In previous weeks and analysis, we used the average MAPE across all sectors to discuss the appropriateness and performance of ARIMA models, but the end goal was to create models for every operating US Customs and Border Protection field office/sector. As such, we'll have to monitor the forecasting performance for each deployed model separately. As for the metrics used to monitor the model, the MAPE shall be used to stay consistent with the previous model evaluation while accounting for the different scales of the model. Additionally, for the data drift of the model, the change in mean and variance shall also be used to monitor the data's distribution and determine whether the model must be retrained as a percentage of the previously trained mean.

Thresholds for Error Detection		
0-25%	**26-50%**	**>50%**

Risk Mitigation Strategies

For future implementation of statistical anomaly detection, we would use a tool such as MLFlow to continuously monitor the status of the model versus the true data. There would need to be some workarounds with regards to pipelining the data for input because there was not an API that we were able to find with regards to the USCBP data. However, the general plan for what would need to be done for continuous deployment would be to refit the models on a smaller, more recent portion of the data to more accurately reflect the recent trends. This would be a necessity because of the relatively recent nature of the drop in immigration due to the change in administration and policy. As it pertains to the model, though, this would be our strategy to mitigate the threat of data drift through continuously monitoring the model.

The major strategy that should be taken is understanding the current policies and actions taken within this space to understand the major changes since the model's last training date and how the new policy and enforcement may change the effects of concept drift. Keeping updated on current news will be crucial to making any changes in the model training process, such as going back over to ARIMAX (exogenous variables included).

For the strategy to mitigate the green threshold, there would not need to be much done here except alert the developer that the model is in the green and that the predictions have been doing relatively well.

As for the yellow, the developer should be given the choice for whether they want to retrain the model having more context than we are able to provide such as outside news sources (NLP would be great input for future iterations). If the developer feels as if the model is not trained on recent enough context, they should give the command to retrain the model to try and get it back into the green.

As for the red, MLFlow would be used to completely retrain the model upon detecting this much of an anomaly with a more recent dataset. The predictions have become so off at this point as well as the change in mean and variance of the dataset that retraining is the best possible way to account for the data drift at this point. The model would automatically be retrained and redeployed.

Retraining and Model Decay

Considering how quickly immigration numbers have plummeted since January 2025, this shows that our model will require routine retraining due to concept drift. We've identified that the effects of data drift are unlikely, due to the underlying reasons on why encounters might change drastically. For example, if there were a massive increase in the number of encounters from a specific region without altering the overall pattern, this may be considered data drift. But we have to consider the underlying reasons as to why such a shift occurred. Concept drift may be more relevant due to changes in policy (enacting another Title 42-adjacent policy, new immigration laws) and global crises occurring (geopolitical events that we've discussed such as Venezuela and the Russian invasion of Ukraine). These changes fundamentally alter the trend or seasonal behavior of the data, thus indicating concept drift. The best practice would likely be to retrain monthly right after USCBP releases the data, as this aligns with ARIMA's significant emphasis and reliance on the most recent data to predict on. However, should there be any limitations on our ability to retrain and deploy an updated model, we could work in conjunction with our defined thresholds to determine when to retrain our models. Sometimes, there is no reason to update a model unnecessarily if it still holds up to our standards.