Notes to exam project

Bilateral Migration Prediction using Deep Learning methods

Ludwig-Maximilians-Universität

Master of Economics

Machine Learning Applications

WS2020/21 Term Paper

By: Yunting LIU, 15.03.2021

Table of Content

A. My research question

For my exam project I wish to continue in the subject area of international migration - this was the topic of my research paper review - and specifically apply deep learning model techniques to the prediction of bilateral international migration. In my review task I have come across some recent work with similar objective, and I wish to fit some models similar to those in published work hands-on.

Method-wise I aim to specifically fit two neural-network-based models (ANN and TabNet) and one tree-based model (Gradient Boosting Tree Regressor) for comparison. I extend beyond the scope of current literature in two ways. Firstly, I go beyond comparison of model results by studying the differences in model insights. Secondly, I am particularly interested in the in-built interpretation technique for TabNet models and examine this in detail.

This note is intended to accompany my project repository - my project code, data etc are all hosted here, so all results can be replicated: https://github.com/jacquell/Bilateral_migration_prediction

B. Methodology and results

The code use to implement the main model methodology discussed in this section is available here:
Repository -> Bilateral_migration_prediction/Exam_bilateral_migration_predictions_deep_learning_v5.ipynb

1. High level methodology

I apply machine learning models to forecast bilateral international migration in a particular year, that is, the number of migrants T from a particular origin country i to a particular destination country j in a particular year t. So the exercise is to predict:

$$\hat{T}_{i,j,t+1} = f(features_{i,t}, features_{j,t}, features_{i,j,t})$$

I apply three machine learning models:
- Artificial Neural Network (ANN)
- TabNet, a new deep learning method targeted for developing models on tabular data
- Gradient Boosting Tree Regressor

in parallel and compare prediction results. Each model is individually optimised. Common Part of Commuters (CPC) is used as the common performance evaluation metric, although a full range of other metric are included for comparison. With the three fitted models I apply result interpretation techniques, predominantly based on Shapley method, to gain insights into the respective model fitting.

2. Data and pre-processing

The modelling is based on 11 years of historical bilateral international migration data from 2004 to 2014, across 102 origin countries and 36 destination countries, with each origin-destination country pair available for between 2 to 11 years data period. This makes a total of 23,947 data samples. Demographic and economic statistics on the origin and destination countries as well as registers of events are also included. Finally the stock of immigrant from an origin country residing in a destination country, $mig\_stock_{i,j,t}$, is also included and prove to be a very significant predictor.

I took the data set initially assembled by [2] and subsequently re-used by [1][1]. This data set additionally covers google search terms which I am currently not using, so I have excluded these columns. I also collate some data columns which are simple transformations (e.g. log transformation) of other features. From the resulting data set I have made use of some of the following data for modelling:

Data field description	**Dimension**	**Data column name**	**Use in model**
Origin-destination country pair-identifier	*i,j*	pair_id	As control feature (expect no predictive effect)
Origin country identifier	*i*	iso3_o, iso3n_o, source_country	As control feature (expect no predictive effect)
Destination country identifier	*j*	iso3_d, iso3n_d	As control feature (expect no predictive effect)
Year	*t*	year	As feature (would be the time-dimension feature in a time series model)
Total migration in same year t	*i,j,t*	tot_mig , log_mig	Not used
Total migration in next year t+1	*i,j,t*	fwd_tot_mig , fwd_log_mig	As target
Measure of origin country unemployment rate	*i,t*	o_sl_uem_totl_zs	As feature
Measure of origin country share of young population	*i,t*	o_sp_pop_0014_to_zs	As feature
Measure of origin country political stability (State Fragility Index)	*i,t*	sfi_sfi	As feature
Measure of origin country political stability (Polity IV Autocracy Score)	*i,t*	pol4_autoc	As feature
Origin country mobile phone subscriptions (per 100 people)	*i,t*	o_IT_CEL_SETS_P2	As feature
Origin country share of internet users (per 100 people)	*i,t*	o_IT_NET_USER_P2	As feature
Origin country number of weather disasters from the EM-DAT database	*i,t*	o_number_weather	As feature
Origin country number of non-weather disasters from the EM-DAT database	*i,t*	o_number_nonweather	As feature
Measure of existing migrant network Migrant stock in year t in destination country j from origin country i	*i,j,t*	mig_stock_tot	As feature
Origin country share of population speaking French	*i*	o_French	As feature
Origin country share of population speaking Spanish	*i*	o_Spanish	As feature
Origin country share of population speaking English	*i*	o_English	As feature
Destination country GDP	*j,t*	d_GDP	As feature
Destination country population	*j,t*	d_pop	As feature
Origin country GDP	*i,t*	o_GDP	As feature
Origin country population	*i,t*	o_pop	As feature

Table 1: Overview of data set and individual data fields used for modelling

[1] The data set is available for public download here: https://www.sciencedirect.com/science/article/pii/S0304387819304900. In this data set, the google search terms are denoted by columns starting with “GTI”.

Aligning to the practice taken by the two research papers, I have split the data set into three sets: train-set using observations from 2004 to 2012 (with associated forecast year 2005 to 2013), validation-set using observations from 2013 (with associated forecast year 2014), and test set using observations from 2014 (with associated forecast year 2015). Overall this roughly represents a 80-10-10 split between train, validation and test data set.

Data is treated with simple pre-processing. Categorical features are turned into numerical format using Label Encoding. Missing values have been filled with "most likely" values for categorical features and mean value for numerical features.

3. Choice of target and features

In keeping with the research question I used next period's bilateral migration volume as target. As we are predicting annual bilateral emigration volume the predicted number must be non-negative. After several model experiments I found that even well-calibrated models will generate occasional negative predictions. To eliminate this I have decided to use the log transformed next period migration volume, specifically the "fwd_log_mig" variable, as the target of my model.

The list of features I used for modelling are those marked as "as control feature" and "as feature" in Table 1.
I did intend to put to test the various machine learning models' (especially deep learning models') ability to correctly deal with non-informative features and correlation among features, therefore apart from very basic de-duplication of features, I have fed most of the available data fields as model features.

Data fields marked as "as control feature" are usually ones which represent some kind of identifier, e.g. country code, which I do not expect to be informative to the model fitting. At the same time I look out for any model fitted impact on these non-informative features as a potential sign of over-fitting.

In general there is strong correlation present among the fitted features, e.g. between the countries' GDP and population. Figure 1 shows the Pearson Correlation between the data fields. For this model iteration I have not taken actions on account of this correlation - in particular I have not manually filtered out correlated features. The intention is to see if the models can correctly work with correlated input features.

4. Evaluation metric and loss objective for model training

In keeping with common practice in migration research I have used Common Part of Commuters (CPC) as the main performance evaluation metric. CPC measures commonality between the observed and predicted migration flow. Specifically its value is 0 when the ground truth bilateral migration flow matrix T and the predicted matrix $\hat{T}$ have no entries in common, and 1 when they are identical. As secondary performance metric I have included Maximum Absolute Error (MAE), Mean Square Error (MSE), Root Mean Square Error (RMSE) and Coefficient of determination (R-squared).

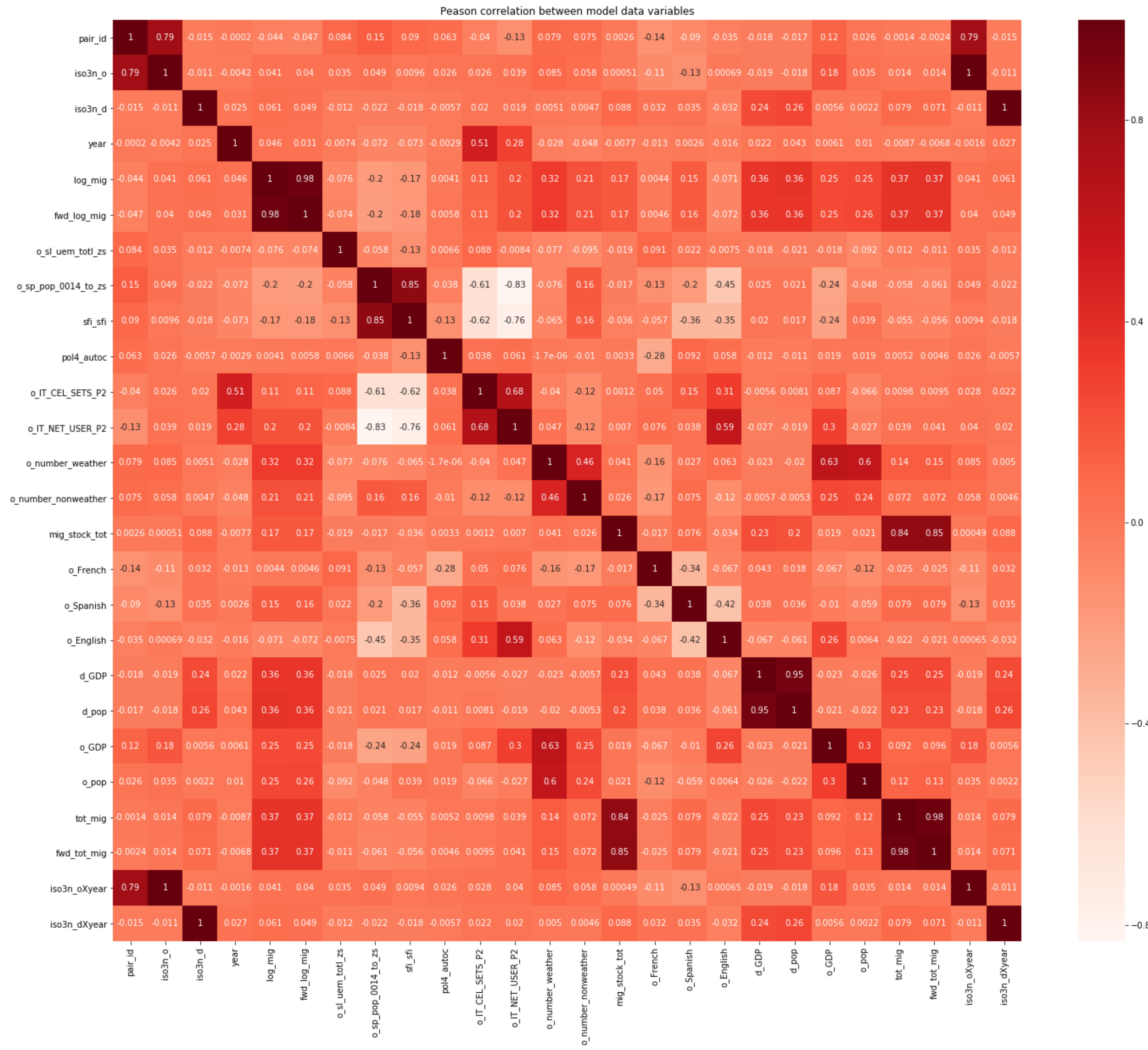


Figure 1: Pearson correlation between data features and target

For loss objective I have implemented custom loss function “CPC_loss” and used this in model training. It is simply defined as CPC_loss = 1 - CPC. As we target a model which maximises CPC we aim to minimise CPC_loss.

5. Model Tuning

The three models are implemented using available Python packages and further fine tuned. In particular:

- Gradient Boosting Tree Regressor is implemented using Xgboost (XGB)
- Artificial Neural Network using Keras (ANN)
- TabNet using PyTorch (TabNet)

For the deep learning model some hyper-parameters are set on available benchmarks, while some more sensitive parameters of batch size and number of hidden layers are optimised using manual grid-search. As a result for TabNet and ANN the “small” versions of the respective models are used, i.e. relatively few and small sized layers are used, compared to TabNet author hyper parameter range suggestions and to existing research paper ANN models. Small learning rates are also found to be advantageous to the fitting and adopted. Gradient clipping is used to avoid potential exploding gradient problems (although I did not observe this to be an issue during the fitting). Finally, drop-out is used for ANN to avoid neural network overfitting.

Further model hyper-parameter choice details and comments are included throughout the main modelling Jupyter notebook. A separate Jupyter notebook contains code used to run hyper-parameter grid search, here: Repository -> Bilateral_migration_prediction/hyper_param_tuning_v2.ipynb

The hyper parameter combinations I used are:

- For ANN, 2 hidden layers each sized 20, Relu activation function, dropout=0.15, learning rate = 0.002, Adam optimiser with gradient clip value = 2, batch_size=256.
- For TabNet, layers {n_d=8, n_a=8, n_steps=5}, learning rate = 0.002, Adam optimiser with gradient clip value = 2, batch_size=4096, virtual_batch_size=1024, and early stopping with patience=50.
- For XGB, max_depth=8, learning_rate=0.05, n_estimators=1000, min_child_weight=1, booster = gbtree, and early stopping with patience=50.

6. Result summary

After tuning, the two deep learning models and the tree based comparison model all yield reasonable predictions for the bilateral migration prediction. The actual bilateral migration and the predicted numbers from the three models on test data set (i.e. predicting 2015 bilateral migration using origin and destination country information as of 2014) is given in Figure 2. The distribution of predictions generated by all three models appear reasonable compared to the actual, where the TabNet and XGB prediction ranges appear to be the closer.

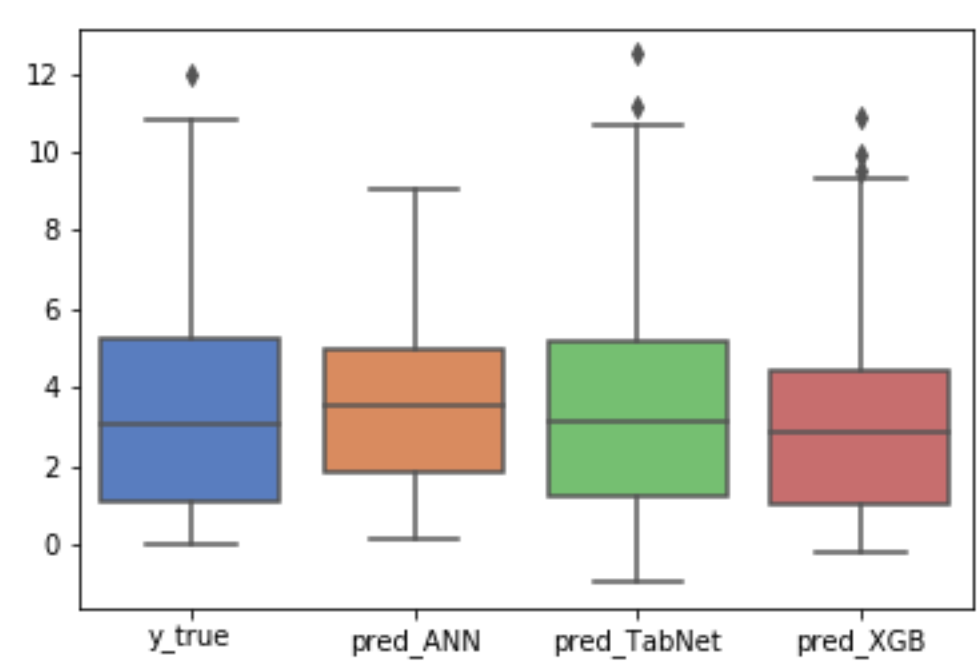


Figure 2: Actual versus predicted (from three models) forward bilateral migration from Test data set

The performance of the three model evaluated on a range of performance metrics is given in Table 2.

Metrics	Performance on Test Set			Performance on Train-validation Set		
	ANN	TabNet	XGB	ANN	TabNet	XGB
mae	1.0165	0.7670	0.7066	0.9542	0.4806	0.1535
mse	1.9158	1.2648	1.0745	1.7487	0.6342	0.0842
rmse	1.3841	1.1246	1.0366	1.3224	0.7963	0.2902
r_squared	0.7253	0.8187	0.8460	0.7384	0.9051	0.9874
CPC	0.8558	0.8862	0.8904	0.8546	0.9237	0.9756

Table 2: Performance metrics from three fitted models

On all metrics the performance of the tree-based XGBoost model is superior than the two deep learning models. TabNet comes to a close second on CPC score. I would have initially expected the deep learning models to performance better given their structural flexibility.

The performance metrics for ANN is consistent between the Train-validation set and the Test set, which is desirable. This outcome probably benefits from the use of drop-out in ANN fitting process. For both TabNet and XGB the performance on Test set is inferior to the performance on Train-validation set, suggesting a possibility of model overfitting.

7. Interpretations and difference between models

Beyond the performance metrics I am interested in how the three respective models consider the drivers of migration. Here I use Shapley interpretation methods[2] to derive insights into the three models and compare results. All exhibits shown in this section plus additional exhibits (including complete partial dependency plots on every individual feature) are available from:
Repository -> Bilateral_migration_prediction/exhibits/

SHAP feature importance based on magnitude of feature attributions to model outcome is available across three models, shown in Figure 3. There is good overlap in what the three models consider as most important features - existing migrant network (measured by migrant stock, the number of existing immigrant from a particular origin country living in a particular destination country at start of the year) and destination country GDP are ranked as top two most important drives across all three models, which origin country population and origin country GDP following closely after.

Since some non-informative control features are included in the models, it appears that all three models gave some weight to such control features: to country pair (“paid_id”) in TabNet and to destination country (“iso3_d”) in

[2] Specifically I used Shapley Kernel Explainer which is able to work all three models.

ANN and XGB. I think this outcome hints at potential missing country features and potential overfitting issues. I examine these two factors in the detailed dependency plots later.

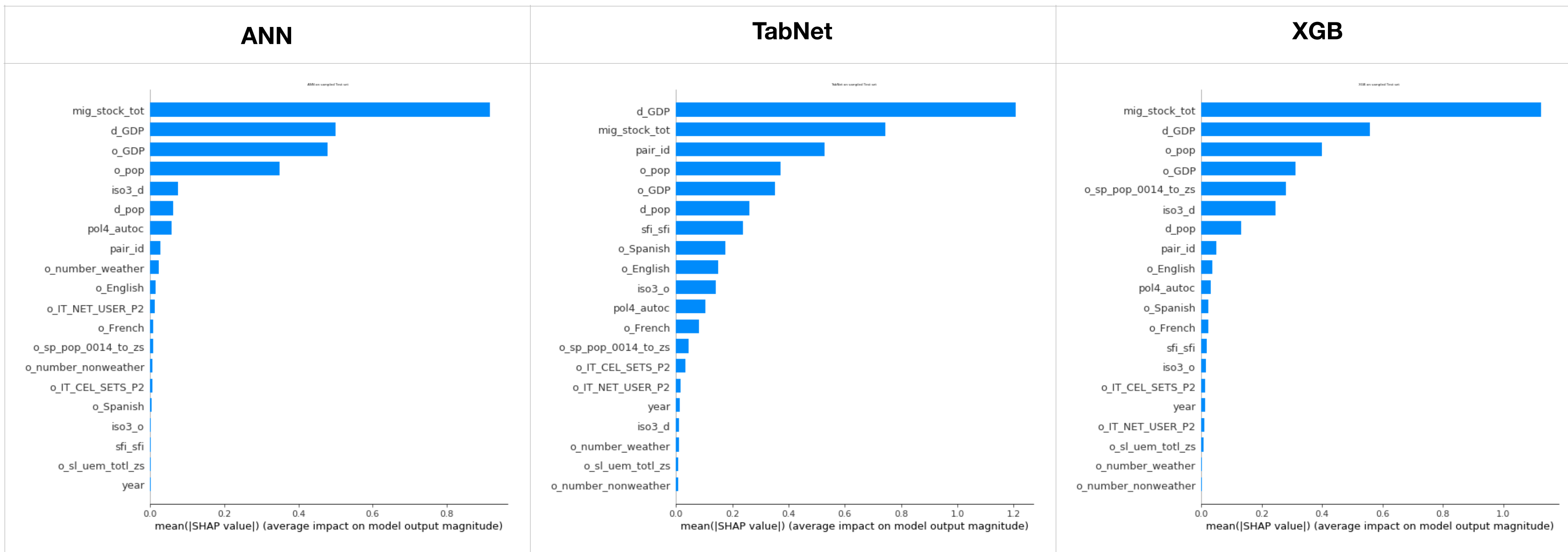


Figure 3: SHAP feature importance from three models

The effects on the most important drivers are consistent between the three models. This is shown in Figure 4.

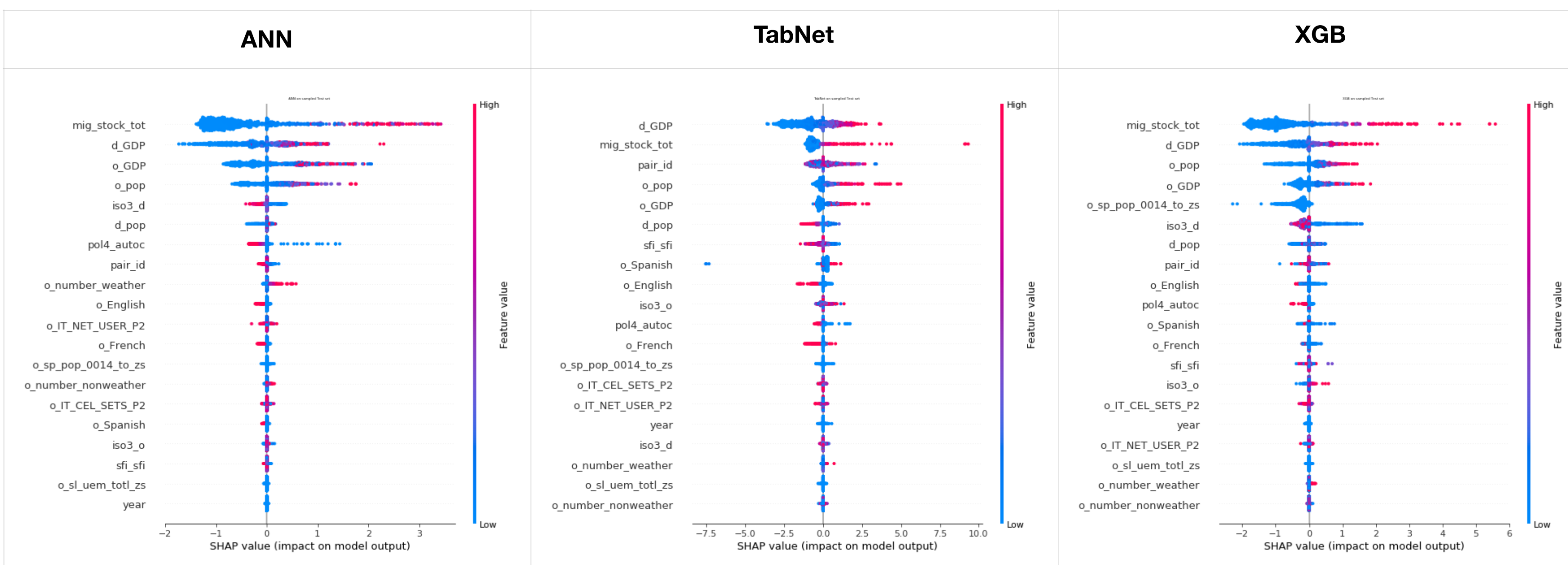


Figure 4: SHAP feature effect overview from three models.

The summary plot combines feature importance with feature effects. Each point on the summary plot is a Shapley value for a feature and an instance. The position on the y-axis is determined by the feature and on the x-axis by the Shapley value. The colour represents the value of the feature from low to high. Overlapping points are jittered in y-axis direction, so we get a sense of the distribution of the Shapley values per feature. The features are ordered according to their importance. [8]

Concretely, high values in existing migrant network (migrant stock”), destination country GDP and origin country population are expected to contribute to higher next year bilateral migration and this expectation is common across all three models. The magnitude of assumed effect differs, for example, TabNet expects a much higher positive effect from large existing migrant network compare to the other two models.

These effects are become clearer when we zoom to SHAP dependency plot level, showing partial-dependency-plot like impact of individual feature values on the predicted outcome. (Such plots for each model and for each feature is available in my repository.) From Figure 5, we can see that all three models expect higher migrant stock to relate to higher next year migration. The trend is most visible on TabNet and XGB model, and from the scale of y-axis we can clearly see that TabNet expect a larger magnitude of impact compared to XGB model, consistent with Figure 4 insights. From Figure 6 we can see across all three models the positive effect of destination country GDP on predicted forward migration. Again, the expected magnitude of impact under TabNet model is higher than for the other two models.

ANN	TabNet	XGB

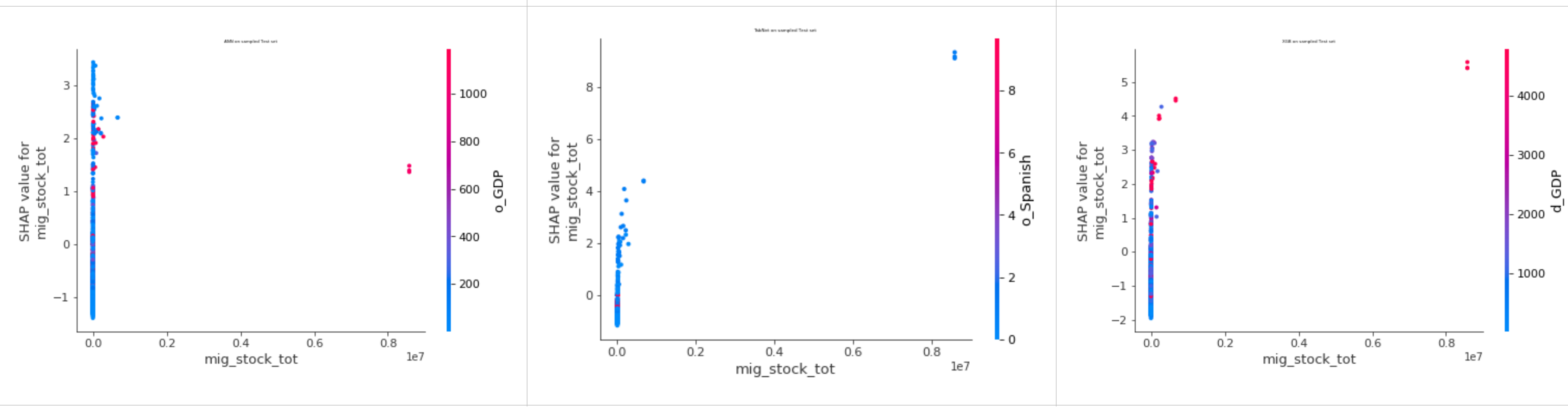


Figure 5: SHAP dependency plot on feature Bilateral Migrant Stock from three models.

SHAP dependence plots are an alternative to partial dependence plots and accumulated local effects. While PDP and ALE plot show average effects, SHAP dependence also shows the variance on the y-axis. This visualisation also shows potential interaction effect with a high-dependency variable, the value of which given in colour scale. Especially in case of interactions, the SHAP dependence plot will be much more dispersed in the y-axis than classical Partial Dependency Plots.

ANN	TabNet	XGB

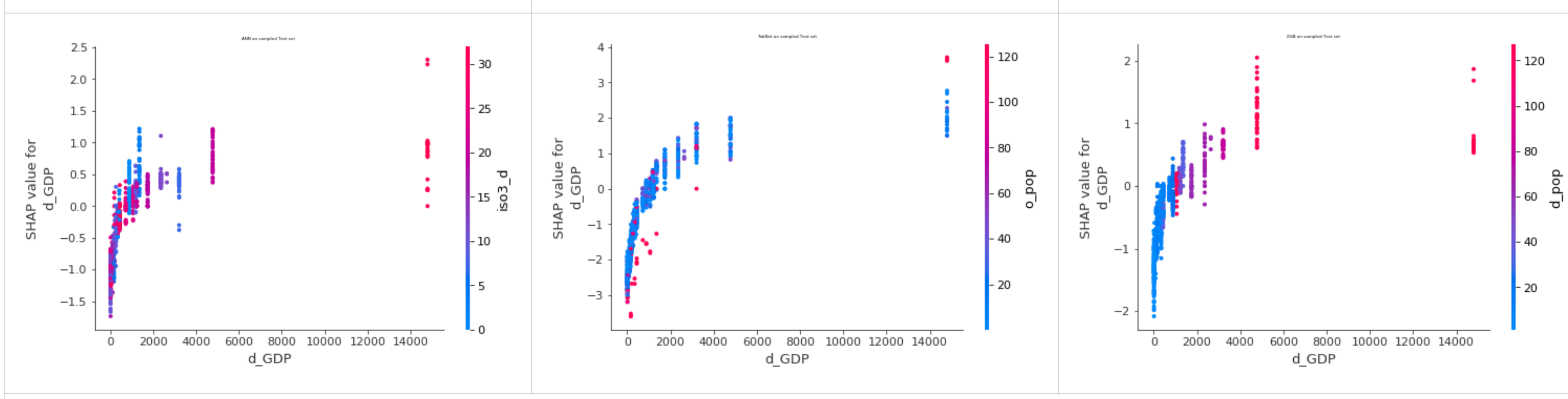


Figure 6: SHAP dependency plot on feature Destination County GDP from three models.

SHAP dependence plots are an alternative to partial dependence plots and accumulated local effects. While PDP and ALE plot show average effects, SHAP dependence also shows the variance on the y-axis. This visualisation also shows potential interaction effect with a high-dependency variable, the value of which given in colour scale. Especially in case of interactions, the SHAP dependence plot will be much more dispersed in the y-axis than classical Partial Dependency Plots.

As the models leverages destination country identifier and country-pair identifier, what I would expect to be non-informative features, I examine the model dependency plots on these two factors in detail in Figure 7 and Figure 8. Interestingly ANN and XGB model both find a negative impact on predicted migration from high values of label encoded destination country identifier and origin-destination country paid identifier. TabNet meanwhile does not suggest this negative overall impact, although TabNet does in general overfit to origin-destination country pair identifier with fairly strong assumed prediction effect.

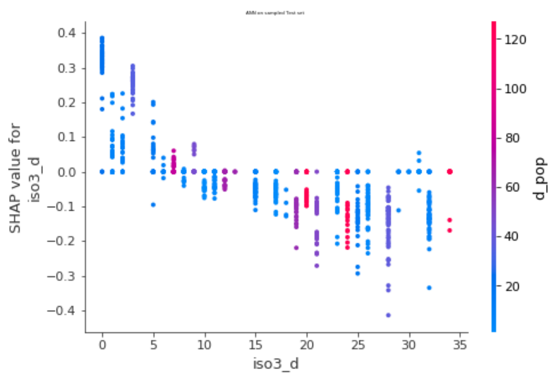
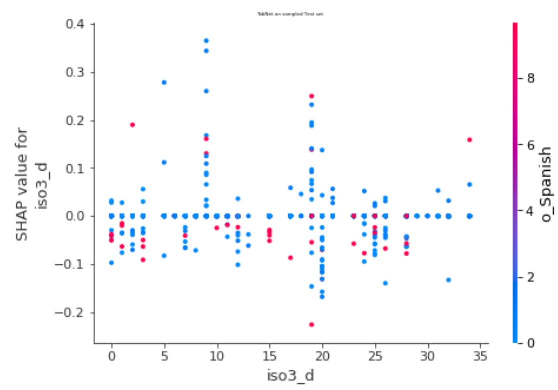
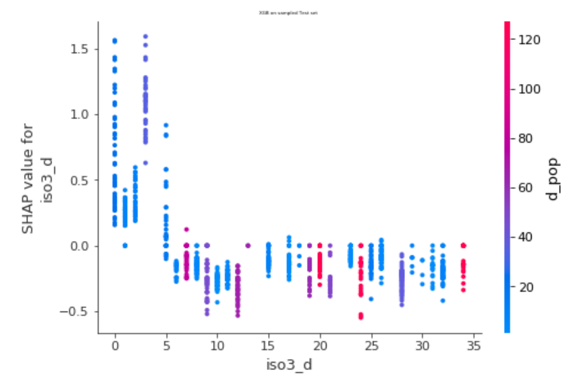
ANN	TabNet	XGB
		

Figure 7: SHAP dependency plot on feature Destination Country Identifier from three models.

SHAP dependence plots are an alternative to partial dependence plots and accumulated local effects. While PDP and ALE plot show average effects, SHAP dependence also shows the variance on the y-axis. This visualisation also shows potential interaction effect with a high-dependency variable, the value of which given in colour scale. Especially in case of interactions, the SHAP dependence plot will be much more dispersed in the y-axis than classical Partial Dependency Plots. [8]

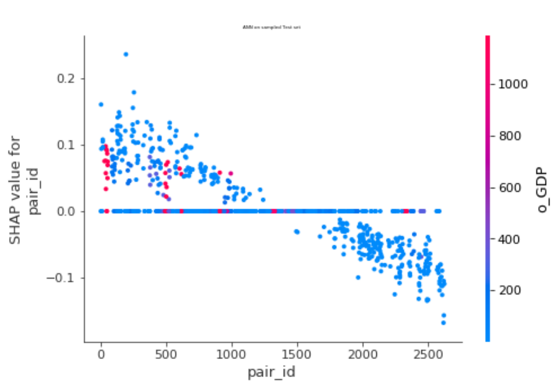
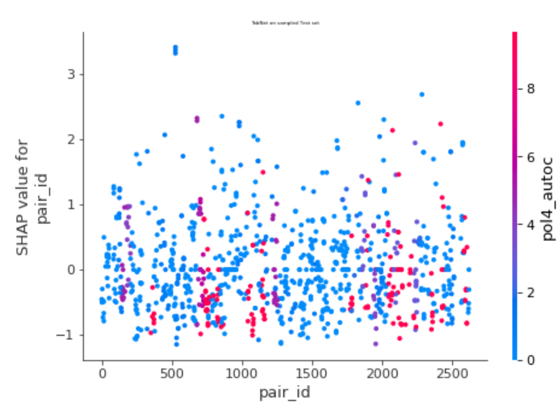
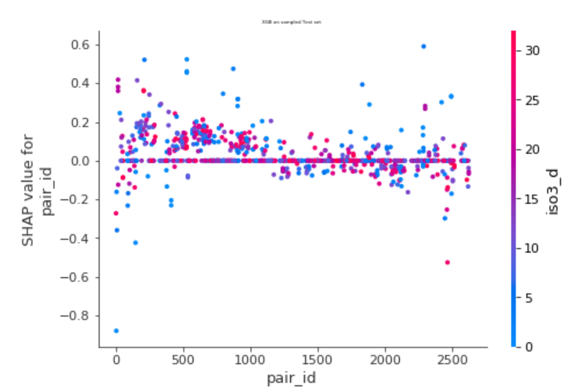
ANN	TabNet	XGB
		

Figure 8: SHAP dependency plot on feature Origin-Destination Country Pair Identifier from three models.

SHAP dependence plots are an alternative to partial dependence plots and accumulated local effects. While PDP and ALE plot show average effects, SHAP dependence also shows the variance on the y-axis. This visualisation also shows potential interaction effect with a high-dependency variable, the value of which given in colour scale. Especially in case of interactions, the SHAP dependence plot will be much more dispersed in the y-axis than classical Partial Dependency Plots. [8]

The model fitting has probably suffered from use of label encoding in the pre-processing step, where categorical features have been converted to numerical using label encoding. This has introduced artificial relation/comparison between the categories. It appears that ANN and XGB models both incorrectly captured the artificial relationship thus introduced (albeit small magnitude of net impact). If I repeat the modelling exercise I would like to try target encoding instead of label encoding to mitigate this issue. TabNet is not affected by this issue because I have had to explicitly specified these features as categorical features for its fitting.

8. Detailed study on TabNet model interpretation

One highlight of TabNet model is its result interpretability, where its feature selection masks can shed light on the selected features at each step [3]. Figure 9 shows the mask I derived from the TabNet model for each of the five decision steps (this was dictated by the model hyper-parameter) fitted on 100 random Test samples, showing which features are most significant in each of the five steps and for each individual sample. The higher the value of the mask for a particular sample, the more important the corresponding feature is for that sample.

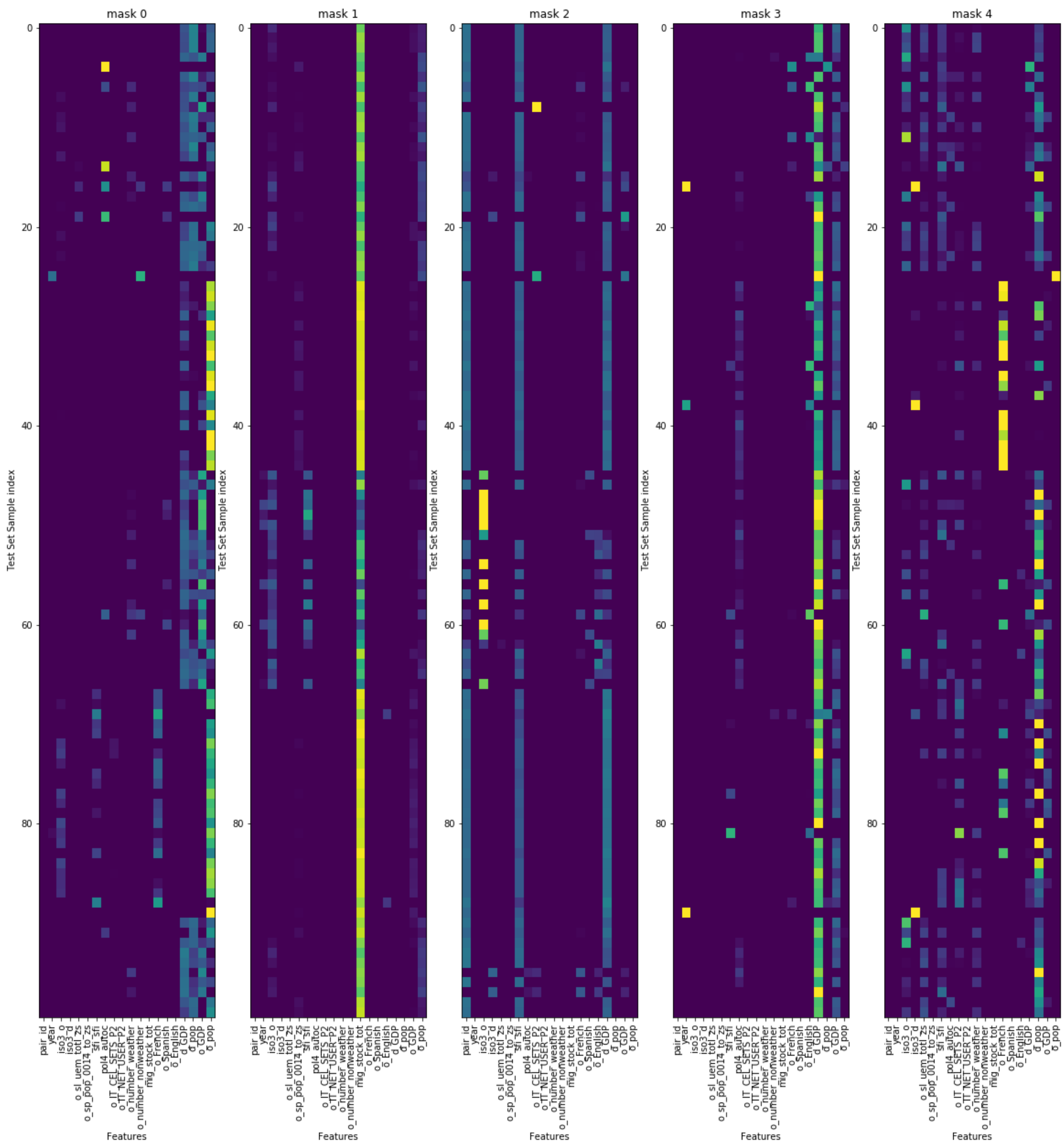


Figure 9: TabNet model interpretation heatmap showing feature importance per sample at each decision step

Very clear is the consistent high impact of migrant stock (on mask 1) and destination country GDP (on mask 3) as two key features on model predictions. Some interesting interaction effects are also visible. For example, on mask 4, the effect of destination country GDP seem to differ for origin countries with French speaking populations.

C. Some model variants explored

I subsequently explored three variants to the main model, the code and available for each is available.

1. Using current year migration volume as a feature for predicting next year migration

Repository -> Bilateral_migration_prediction/variant_use_lag_migration_as_feature.ipynb
Bilateral_migration_prediction/exhibits/variant_lag/

I explored adding current year migration, “log_mig”, as an additional feature to predict next year migration, “fwd_mig_log”. Initially I excluded this feature from the main model as this factor tends to dominate all other migration drivers, and I thought I would have preferred to study autocorrelation effects through a model suited for time series analysis, such as LSTM.

Adding current year migration as a feature improves the three model performances significantly, as shown in Table V1. Dependency plots shown in Figure V1 also show highly positive correlation between current year migration and predicted next year migration - the expected effect in TabNet and XGB models are close to linear.

Metrics	Performance on Test Set			Performance on Train-validation Set		
	ANN	TabNet	XGB	ANN	TabNet	XGB
mae	1.3708	0.5724	0.3999	1.3316	0.4580	0.2989
mse	2.7513	1.0505	0.3278	2.4653	0.6779	0.1889
rmse	1.6587	1.0249	0.5725	1.5701	0.8234	0.4346
r_squared	0.6056	0.8494	0.9530	0.6312	0.8986	0.9717
CPC	0.8269	0.9153	0.9401	0.8215	0.9269	0.9522

Table V.1: Performance metrics from three fitted models, adding current year migration as an additional feature

ANN	TabNet	XGB

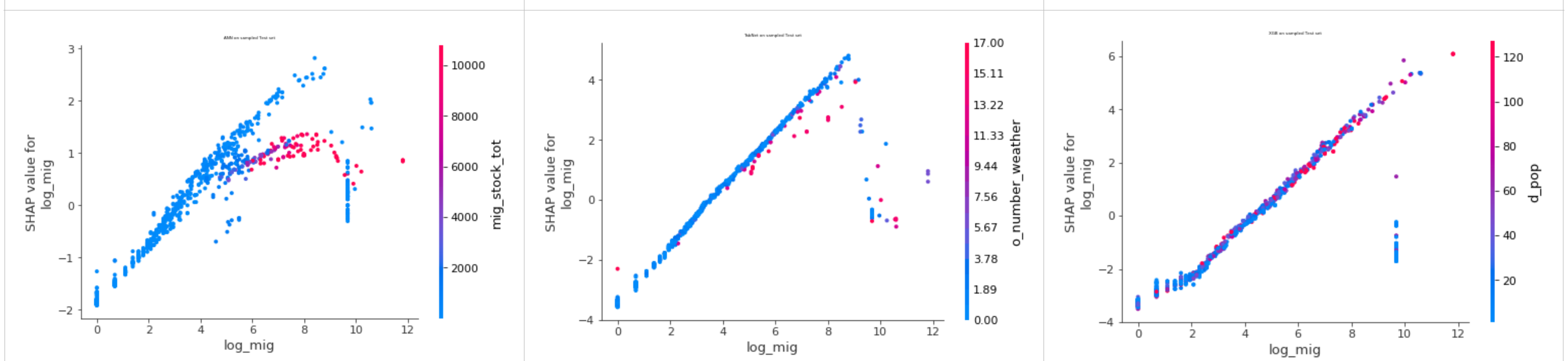


Figure V.1: SHAP dependency plot on feature Current year migration from three models.

SHAP dependence plots are an alternative to partial dependence plots and accumulated local effects. While PDP and ALE plot show average effects, SHAP dependence also shows the variance on the y-axis. This visualisation also shows potential interaction effect with a high-dependency variable, the value of which given in colour scale. Especially in case of interactions, the SHAP dependence plot will be much more dispersed in the y-axis than classical Partial Dependency Plots.

2. Using engineered features and a manually filtered features list

Repository -> Bilateral_migration_prediction/variant_filtered_and_engineered_features.ipynb
Bilateral_migration_prediction/exhibits/variant_features/

In my main model I intentionally included several dependent features as well as non-informative features, since in principle good deep learning models should be able to cater for these considerations. Here I explore one model variant where I removed non-informative features and included some engineered features to replace the highly correlated features. Specifically I did the following relative to the initial list of features presented in Table A:

- I removed non-informative features: “pair_id”, “iso3_o” and “iso3_d”, which are largely identifiers for the origin and destination country pairs.
- I created “GDP_per_capita”, calculated as GDP / population, and used this to replace “GDP” as a feature. “GDP” in the initial feature list was highly correlated to “Population”.
- I created “o_mainlanguage” as a new feature, which takes the value {“English”, “French", “Spanish”, “Other”} depending on the language spoken by majority population. I used this to replace the three correlated features “o_english”, “o_french” and “o_spanish” which captured the share of population speaking each of the three languages.

Model performances, shown in Table V.2, and are worse than the base model. However, some interesting insights came out of this model variant. For example, by examining the TabNet model heatmap in Figure V.2, it appears to have identified a group of country pairs (roughly the last ten samples shown) where the origin country’s main language, political stability and share of internet users played a major role in determination of next year migration, in contrast to the other country pairs examined where a stronger effect is observed on GDP per capita.

Metrics	Performance on Test Set			Performance on Train-validation Set		
	ANN	TabNet	XGB	ANN	TabNet	XGB
mae	1.4558	1.3421	0.7810	1.3732	0.6080	0.1660
mse	3.2512	3.9178	1.3260	2.9395	0.8543	0.0907
rmse	1.8031	1.9793	1.1515	1.7145	0.9243	0.3011
r_squared	0.5339	0.4383	0.8099	0.5603	0.8722	0.9864
CPC	0.8160	0.7951	0.8828	0.8150	0.9051	0.9736

Table V.2: Performance metrics from three fitted models, after manually filtering and engineering features

3. Using randomised train-val-test split, not restricted to year

Repository -> Bilateral_migration_prediction/variant_randomised_train_test_split.ipynb
Bilateral_migration_prediction/exhibits/variant_randomsplit/

I re-run the main model with a random split of data into train, validation and test data set using 80-10-10 ratio, therefore not restricting the split by year, but maintaining a similar split ratio as in the main model run. This would reduce ay potentially unusual development in years 2013 (used as main model validation set) and 2014 (used as

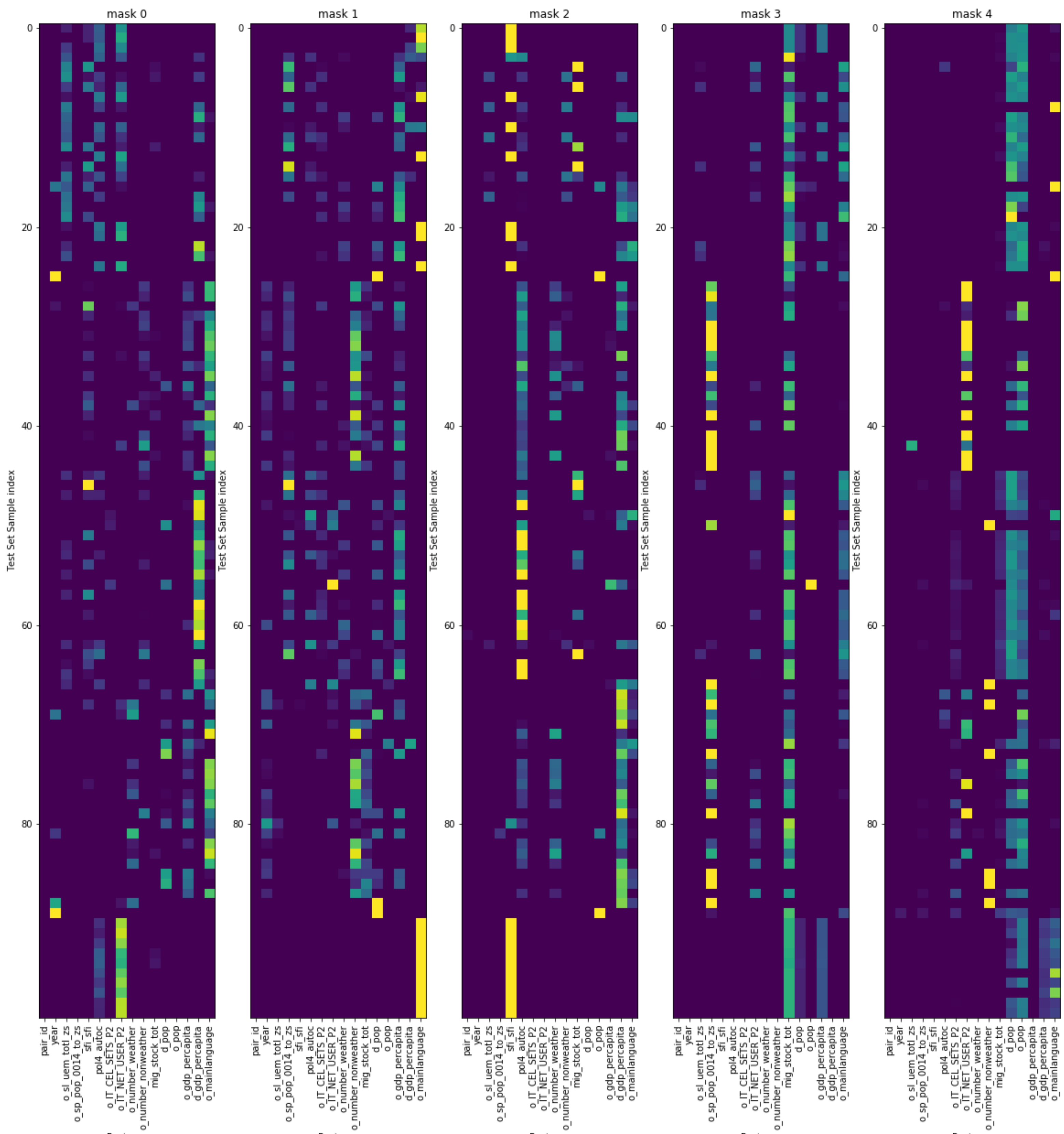


Figure V.2: TabNet model interpretation heatmap showing feature importance per sample at each decision step, after manually filtering and engineering features The model identifies a group of country pairs (roughly the last ten samples shown) where the origin country's main language, political stability and share of internet users play a major role in determination of next year migration, in contrast to the more common migration drivers applicable to other country pairs examined.

main model test set) which would unduly penalise our model performance evaluation.

As seen from Table V.3, the results using a randomised data split show better performance compared to the main model results. This indicates that there is some year-related effects, including for years 2013 and 2014, can be picked up if the model is supplied with the data during training stage. The main model's procedure of splitting data into train, validation and test sets by year is very demanding on model performance - even if the main model would have captured all drivers fo migration to perfection, the predictions on test set (=2014 year data in the main model) would still have been negatively influenced by unpredictable development in the year 2014 which differed to the earlier years.

Metrics	**Performance on Test Set**			**Performance on Train-validation Set**		
	ANN	**TabNet**	**XGB**	**ANN**	**TabNet**	**XGB**
mae	1.2060	0.6126	0.3604	1.1923	0.5483	0.1467
mse	2.4771	0.9688	0.3010	2.3447	0.7683	0.0537
rmse	1.5739	0.9843	0.5487	1.5312	0.8765	0.2317
r_squared	0.6239	0.8529	0.9543	0.6513	0.8857	0.9920
CPC	0.8267	0.9019	0.9413	0.8325	0.9146	0.9768

Table V.3: Performance metrics from three fitted models, splitting data randomly across the years into train, validation and test set

D. Extension opportunities

Due to time limitations I will subsequently list several extension ideas I have if I were to work further on this project.

1. Extending list of features, pre-processing and systematic feature selection

The list of features tested in model fitting can be extended by incorporating additional country factors. I can for example think of including:

- religion of origin and destination countries, from World Bank database
- distance between two countries, derived from country geo-coordinates
- destination country immigration policy restrictiveness, from DEMIG data [9]

as features with potentially good explanatory power.

With additional features included we can also develop a systematic feature selection method, e.g. by using Backward Elimination or Recursive Feature Elimination, to develop a short list of features which are the most relevant ones. Having such feature selection before model fitting could reduce the risk of overfitting.

2. Exploring Time-series models, e.g. LSTM

The data used in this analysis is a cross-sectional time series data spanning from 2004 to 2014, but treated completely as a cross-sectional data in the study. None of the three fitted models - ANN, TabNet nor Gradient Boosting Tree - structurally cater for autocorrelation effects. In my view autocorrelation effect should be introduced to improve prediction performance. One way is to use a Recurrent Neural Network or Long Short-term Memory (LSTM) Network [6, 7], which models time series data as a sequence and remembers past period factor and outcome values in predicting next period outcome.

I did attempt to develop a LSTM model but without success. The model did not fit, and here either I encountered an exploding gradient problem, or else made mistakes in other ways. My attempted model is available at:
Repository -> Bilateral_migration_prediction/hyper_param_tuning_v2.ipynb

3. Exploring causality

I have focused my effort entirely on developing a predictive model without analysing causality. Several methods, e.g. Causal Tree models, use clustering and appropriate data transformations to allow estimation of unobserved individual treatment effects. Other emerging methods to undercover underlying causalities from observational data include Causal Bayes Networks and Structural Equation models. These represent extension opportunities to the current predictive model.

E. References

[1] N. Golenvaux, P. G. Alvarez, H. S. Kiossou and P. Schaus, "An LSTM approach to Predict Migration based on Google Trends" ArXiv, abs/2005.09902. Published 2020.

[2] M. H. Böhme, A. Gröger, and T. Stöhr, “Searching for a better life: Predicting international migration with online search keywords,” Journal of Development Economics, vol. 142, p. 102347, Jan. 2020, doi: 10.1016/j.jdeveco.2019.04.002.

[3] S. O. Arik, and T. Pfister, "TabNet: Attentive Interpretable Tabular Learning". 2019. arXiv preprint arXiv:1908.07442, 2019. URL: https://arxiv.org/pdf/1908.07442.pdf

[4] A. P. Masucci, J. Serras, A. Johansson, and M. Batty, “Gravity versus radiation models: On the importance of scale and heterogeneity in commuting flows,” Physical Review E, vol. 88, no. 2, p. 022812, 2013.

[5] C. Robinson and B. Dilkina, “A machine learning approach to modeling human migration,” in Proceedings of the 1st ACM SIGCAS Conference on Computing and Sustainable Societies, 2018, pp. 1–8.

[6] F. A. Gers, J. Schmidhuber, and F. Cummins, “Learning to Forget: Continual Prediction with LSTM,” Neural Computation, vol. 12, no. 10, pp. 2451–2471, Oct. 2000, doi: 10.1162/089976600300015015.

[7] S. Hochreiter and J. Schmidhuber, “Long short-term memory,” Neural computation, vol. 9, no. 8, pp. 1735–1780, 1997.

[8] C. Molnar, “Interpretable Machine Learning - A Guide for Making Black Box Models Explainable,” https://christophm.github.io/interpretable-ml-book/index.html

[9] International Migration Institute,”Determinants of International Migration (DEMIG) - POLICY data”. 2021.