CSE 255 : Assn 1 Report

Name: Sonali Rahagude
PID: A53051652

Introduction

The project is an on-going challenge on the DrivenData website, which hosts data mining competitions[1]. The project aims at using data aggregated and collected by the World Bank, create a model to predict progress towards the Millennium Development Goals.
Since its founding in 1944, the World Bank has been gathering data to help it alleviate poverty by focusing on foreign investment, international trade, and capital investment. In the year 2000, the member states of the United Nations agreed to a set of goals to measure the

progress of global development called Millenium Development Goals. These goals provide the big-picture perspective on international development. We can use these indicators to help understand where to focus development resources. Should we invest more in education? Healthcare? Clean drinking water? Access to capital?

Choice of Dataset

The dataset consists of various time series, each corresponding to an indicator defined by the United Nations for eg. Poverty ratio, Child mortality rate, Forest area etc.The data consists of various time series for the range 1972 - 2007 on over 1200 macroeconomic indicators in 214 countries around the world. Each row represents a time series for a specific indicator and country. The row has an id, a country name, a series code, a series name, and data for the years 1972 - 2007. The entire dataset consists of 195402 tuples. A random snapshot of the data looks like the below.

index,1972,1973,1974,1975,1976 ,1977, 1978, 1979, 1980, 1981, 1982, 1983, 1984, 1985, 1986, 1987, 1988, 1989, 1990,1991,1992,1993,1994,1995,1996,1997,1998,1999,2000,2001,2002,2003,2004,2005,2006,2007,Country Name,Series Code,Series Name
269296,,,,,,,,,,,,,,,,,0.997,0.997,0.997,0.997,0.997,0.997,0.997,0.997,0.997,0.997,0.997,0.997,0.997,0.997,0.997,0.997,0.997,0.997,0.996,United Arab Emirates,7.8,Ensure environmental sustainability

Prediction Task

The UN measures progress towards the Millenium Development goals (MDG) using indicators such as percent of the population making over one dollar per day, literacy rate, Incidence of HIV on population etc. Each goal has one or more targets, and each target has one or more indicators which are simply a subset of the 1200 macroeconomic indicators. The task is to predict the change in these MDG indicators [3] one year into the "future" from the year 2007 [1]. I will call these MDG indicators as the forecast indicators that need to be predicted.
This project deals with predicting the following indicators across different nations,

1. **Achieve universal primary education**
2. **Reduce child Mortality**
3. **Ensure environment sustainability**
4. **Combat HIV/AIDS**
5. **Develop a global partnership for development: Internet Use**

Predicting future progress will help to understand how we achieve these goals by uncovering complex relations between these goals and other economic indicators. The UN set 2015 as the target for measurable progress. Given the data from 1972 - 2007, the task is to predict a specific indicator for each of these goals in 2008.

Literature Survey

There have been various works on time series analysis but no particular has been done on this specific project yet. For time series analysis and prediction, I have drawn many concepts from [4], which will be discussed in the following sections.

Exploratory Analysis

The initial goal of exploratory analysis was to find out relationships between the different indicators based on time.I expected cases where a high cross correlation between 2 indicators was due to one being immediately computed from the other. Nonetheless, analyzing the cross correlations could possibly uncover indirect impacts of some of the indicators on the indicator being forecasted (latent factors) and it might be worthwhile to use these in the prediction model. It is clear from the problem description that we need to build prediction models per country since the data has been collected on a county by country basis. Also, for a given country, every indicator to be predicted has to be modelled separately for regression. Thus, for 247 countries and 5 indicators, we have to predict a total of 1235 variables. Each of them will have its own prediction model, with features defined according to the feature selection strategy described in the later sections.

Implementation Environment

I used Python, specifically pandas for exploratory analysis and scikit-learn for the modeling part.

Per Indicator Trend Across Different Countries

As an initial exploratory task, I decided to plot the time series of different indicators across different countries to see if there is a general trend that a specific indicator may follow across countries. The following figures show the trend across countries.

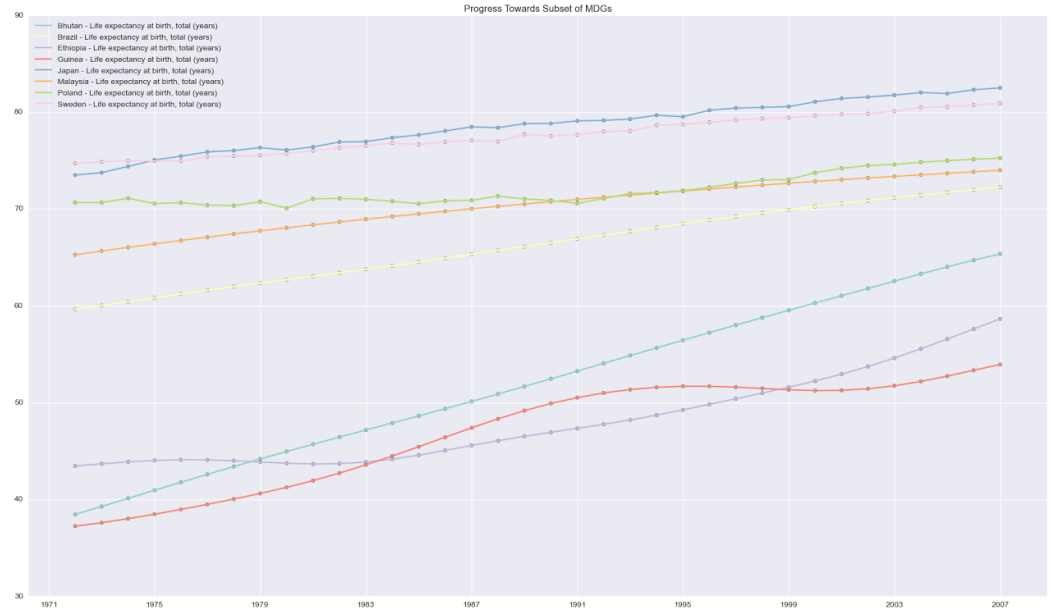


Fig 1: Life Expectancy of birth, total (years)

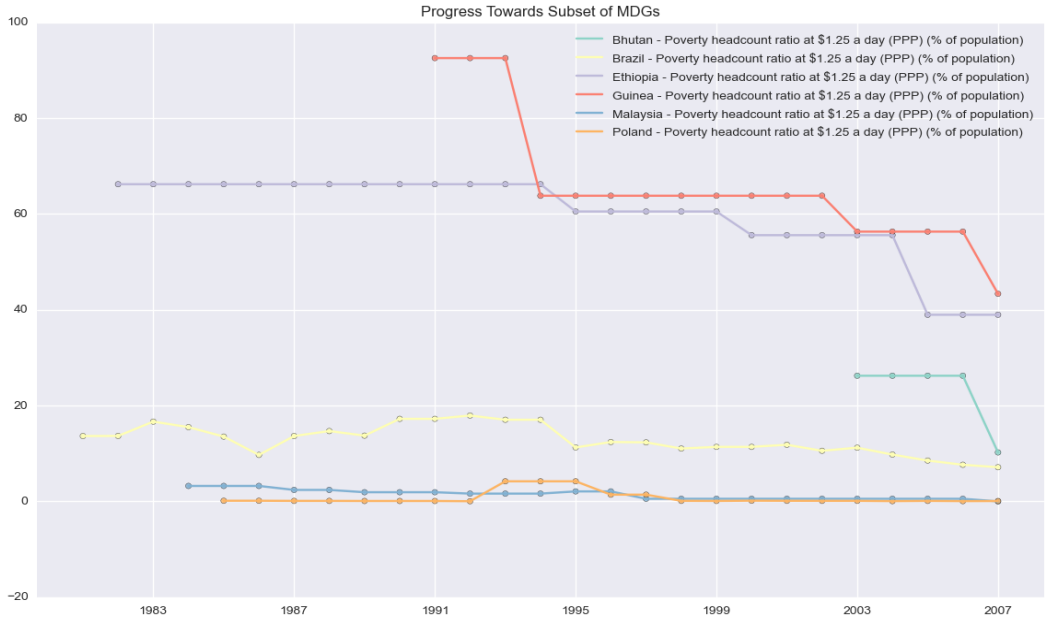


Fig 2: Poverty Headcount Ratio at 1.25$ a day PPP(% of population)

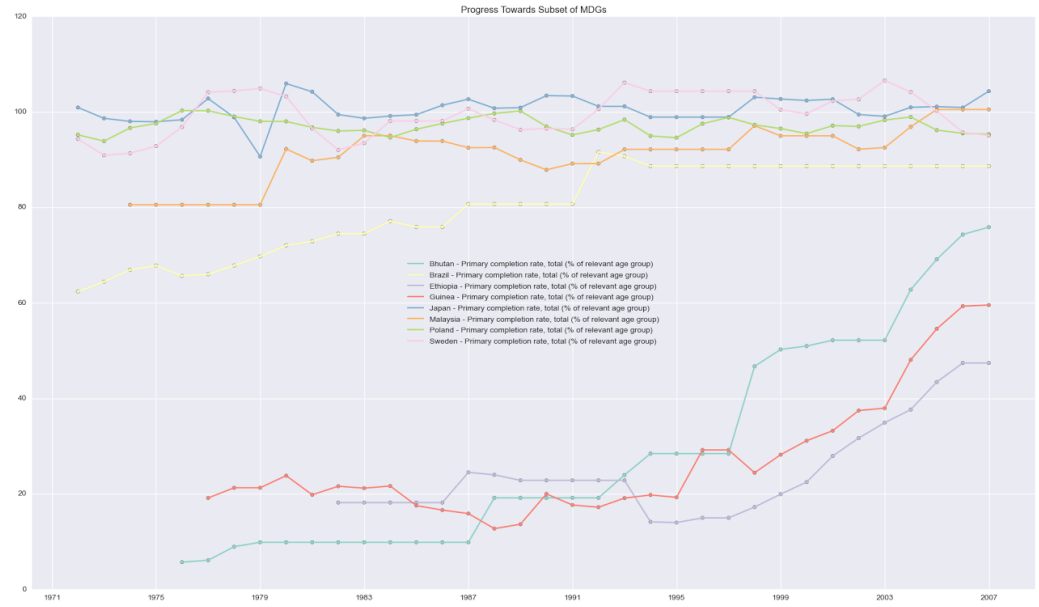


Fig 3: Primary completion Rate, total (% of total relevant age group)

Observations

As is seen from the figures, there is a general increasing or decreasing trend across various countries for a particular indicator. One can also notice that the plots are generally divided into 3 different regions. For example, in figure 2, the poverty head count ratio for countries of Ethiopia and Guinea are the the highest, in the range of 40% - 90%, that of Bhutan and Brazil are somewhere in the middle 10% - 30% while those of Malaysia and Poland are below 10%. This indicates a clear bifurcation in the trend of the indicators across least developed, developing and developed nations.

Cross correlation

To identify the most correlated indicators, I found the pair wise cross correlation of the forecast indicators (MDG indicators) with all the other existing indicators for a given. The idea was to use such indicators as features in the model in the hope of making better predictions for the forecast indicator. I used Pearson correlation coefficient for computation.

Detrending

Since two time series have an inherent mutual dependency introduced by the time, we need to remove this trend for the series before comparing them for correlation [4]. A non parametric method to remove a trend is first differences. With first differences, we subtract from each observation , the previous observation, so that the new series becomes,

$$y'(t) = y(t) - y(t-1)$$

Methodology

For a given country and forecast indicator, I cross correlate it with all the time series of all other indicators of that specific country. For every pair, the correlation coefficient is recorded. At the end, I chose the pair with the highest correlation coefficient and record the correlated indicator as well as the coefficient value in a CSV value. The format of the CSV file is, *Indicator; Country; Correlated Indicator; Corr Coefficient.* The CSV file thus contains an indicator that is most highly correlated to the forecast indicator and can be consulted later while building the prediction model.

Observations

The table below shows a subset of the CSV file mentioned above. As seen, some indicators are highly correlated such as Ensure environmental sustainability in Bhutan is highly correlated with the indicator 'Forest area (sq km)'. Also, notice that 'Reduce child mortality' in both El Salvador and Slovak Republic is highly correlated to 'Prevalence of anemia among children (% of children under 5).' Looking at the scatter plots for these in Fig.6 and Fig. 7 though, we can see there is a different trend in the countries for the 2 indicators. This may suggest that ''Prevalence of anemia' might be a very good predictor variable for 'Reduce child mortality'

Another interesting correlation, in for 'Reduce child mortality' for Japan which is negatively correlated to 'Survival to age 65 female'. The scatter plot is shown in Fig 9. Though the indicators are not directly related, the correlation is intuitive.

Indicator	Country	Correlated Indicator	Corr Coefficient
Combat HIV/AIDS	Paraguay	Improved water source, urban (% of urban population with access)	1
Ensure environmental sustainability	Bhutan	Forest area (sq. km)	0.991
Reduce child mortality	El Salvador	Prevalence of anemia among children (% of children under 5)	0.98
Reduce child mortality	Slovak Republic	Prevalence of anemia among children (% of children under 5)	0.956
Ensure environmental sustainability	Afghanistan	Arable land (hectares per person)	0.942
Reduce child mortality	Japan	Survival to age 65, female (% of cohort)	-0.948

Scatter Plots

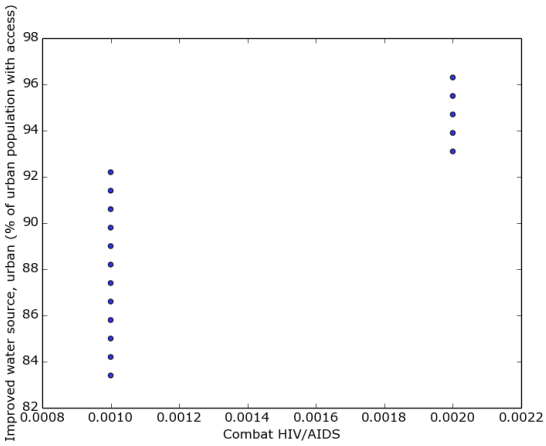


Fig 4. Paraguay

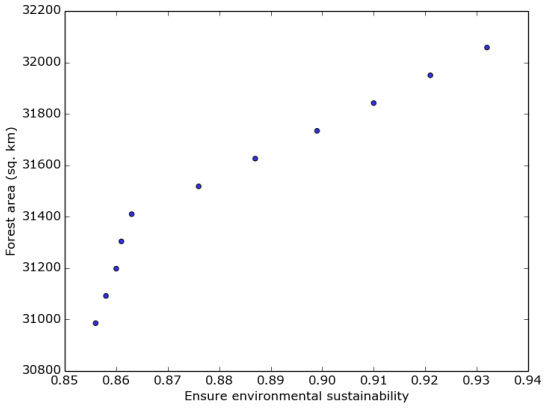


Fig 5. Bhutan

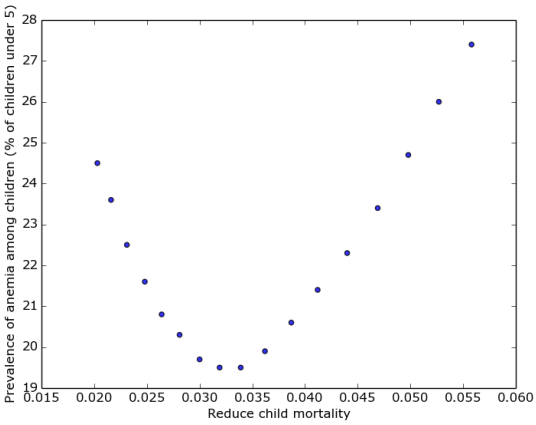


Fig 6. El Salvador

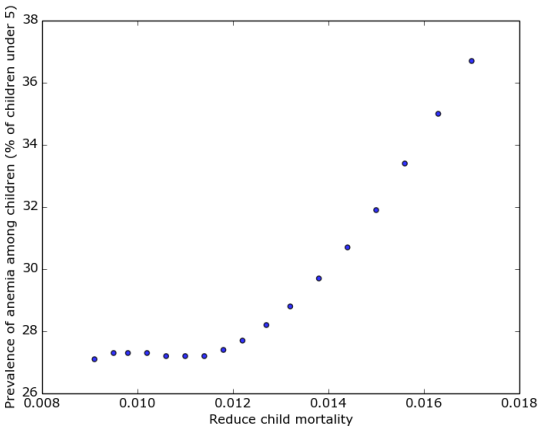


Fig 7. Slovak Republic

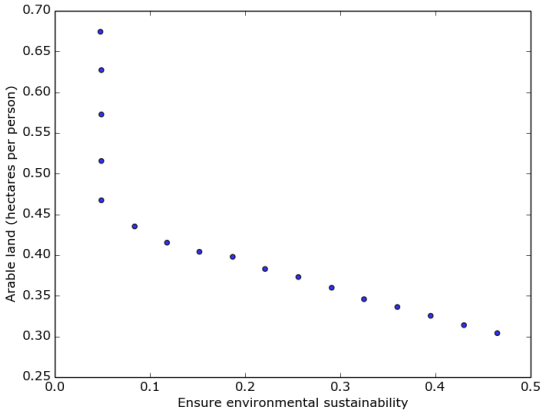


Fig 8. Afghanistan

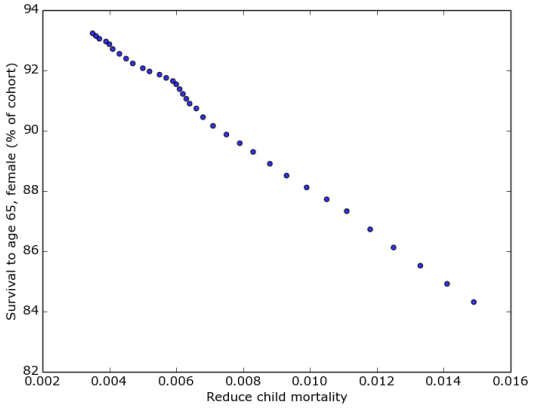


Fig 9. Japan

Autocorrelation

Just as correlation measures the extent of a linear relationship between two variables, autocorrelation measures the linear relationship between lagged values of a time series. In this case, I used a lag of one year so, the correlation measures the relationship between $y(t)$ and $y(t-1)$ [4]. I used Pearson correlation coefficient for computation.

Methodology

For all the forecast indicators for all countries, I calculate the autocorrelation. The correlation coefficient is recorded in a CSV value. The format of the CSV file is,
Indicator; Country; Corr Coefficient.
The CSV file thus contains values which may suggest high auto correlation for certain indicators. that is most highly correlated to the forecast indicator and can be consulted later while building the prediction model.

Observations

The table below shows a subset of the CSV file mentioned above. As seen, some indicators have a value of 'nan' which is infinity. A closer look at the data entry for 'Combat HIV/AIDS ' for Morocco revealed that the indicator has not changed over years at all, which leads to the infinity coefficient. Hence, the plot of such indicators (Fig. 10, 11) shows a single point.

Interestingly, many indicators have a high correlation coefficient. A quick file on the CSV file revealed that about 600 indicators out of a total 1235 total variables (indicator + country) have a coefficient > 0.5.

It is more interesting to look at indicators with low auto correlation. As we can see in the table, the autocorrelation for 'Achieve universal primary education' in Belize, Central African Republic, Dominican Republic and Macedonia is really low that there is no defined trend as can be seen in scatter plots Fig.14 and Fig. 15.

Indicator	Country	Auto Corr Coefficient
Combat HIV/AIDS	Morocco	nan
Ensure environmental sustainability	Seychelles	nan
Ensure environmental sustainability	Bangladesh	1
Develop a global partnership for development: Internet Use	Bermuda	0.998
Achieve universal primary education	Belize	0.18

Achieve universal primary education	Central African Republic	0.138
Achieve universal primary education	Dominican Republic	0.06
Achieve universal primary education	Macedonia, FYR	0.031

As seen above, the nan values actually indicator

Scatter Plots

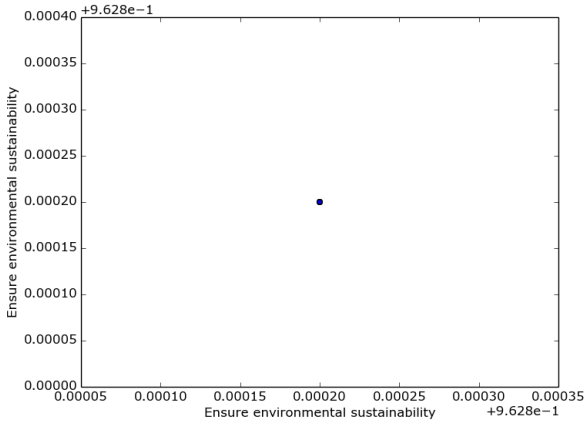


Fig 10. Morocco

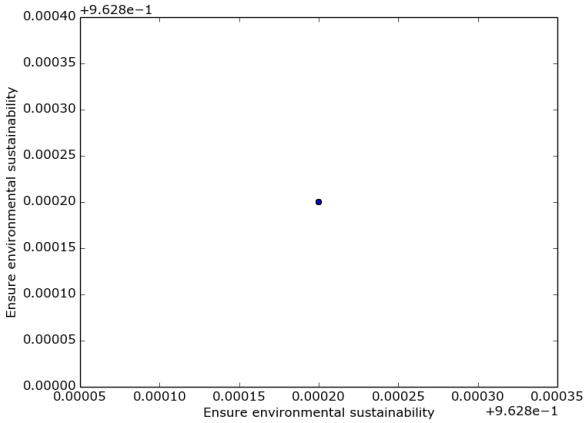


Fig 11. Seychelles

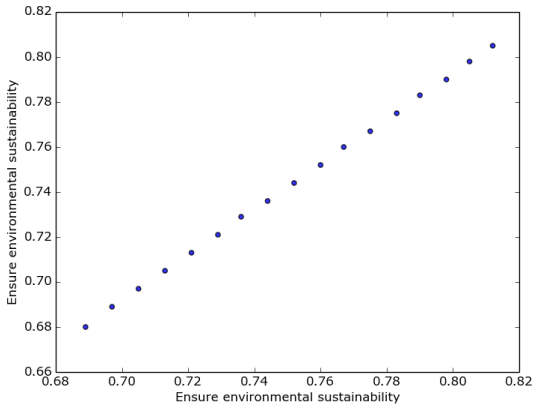


Fig 12. Bangladesh

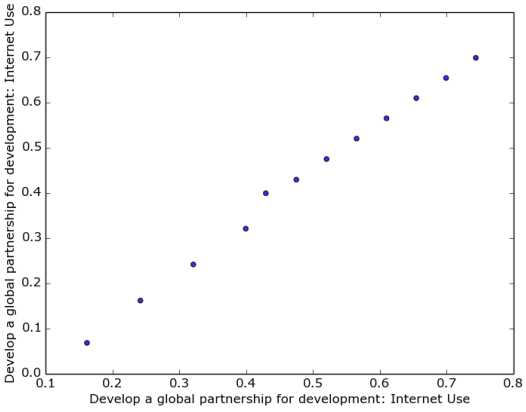


Fig 13. Bermuda

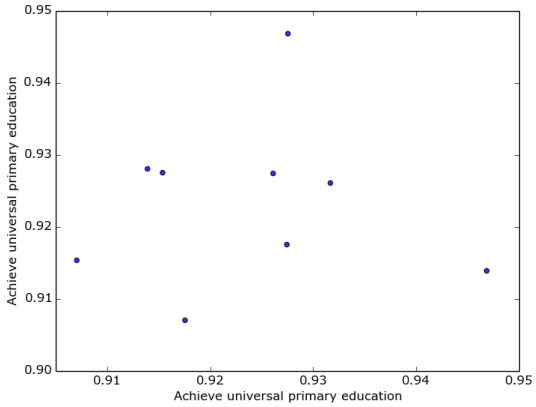


Fig 14. Central African Republic

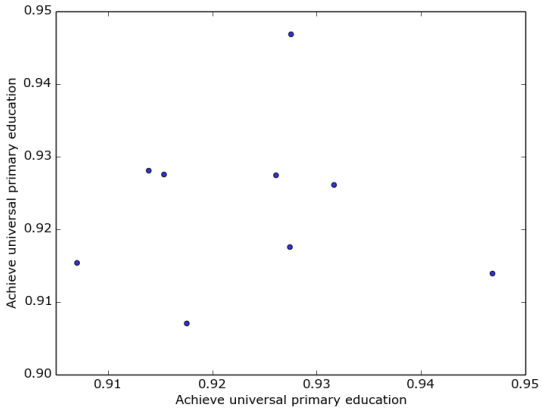
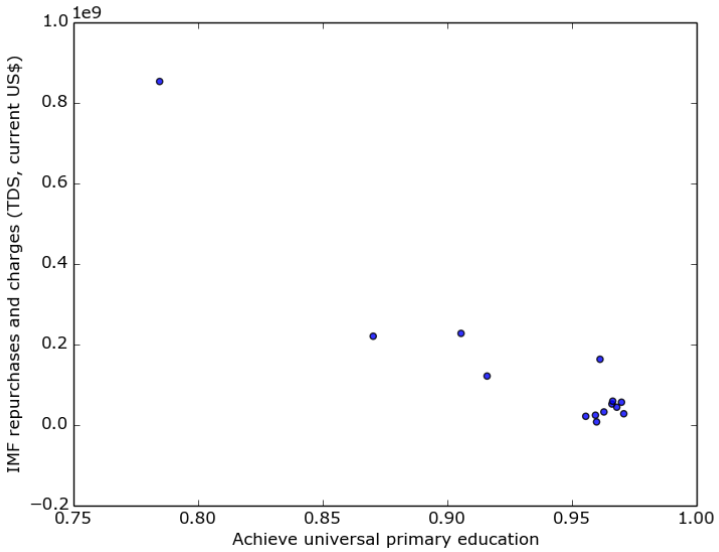


Fig 15. Macedonia

Common Pitfalls in Time Series Analysis

Correlation does not imply causation

Regressing non stationary time series may lead to spurious correlation as already mentioned. This can be clearly seen from the plot below for Hungary.



Feature Extraction

Given the small sized nature of the time series dataset, one might say that we can simply model all the indicators as features in the training model since one of the major motivations for feature extraction is to reduce the time for training the model by selecting only a subset of relevant features. However, feature selection often helps, especially when the classifier being

used is sensitive to noise or dependent features, as is the nearest neighbor classifier, neural network classifier etc. Since, I had small sized data, I could simply model every indicator and its year as a feature and then train the classifier. This will be my baseline predictor model.

Based on the results of the exploratory analysis, I used correlation to extract features from the time series. As is observed, in most cases, the auto correlation coefficient is sufficient high, indicating that we have simply model the relationship based on the last t observations.

A more complex case arises with indicators having lower auto correlation coefficients, indicating that the last t observations, may not be sufficient to accurately forecast the value of the indicator, in which case, it would be worthwhile to look at the cross correlation results and selecting correlated indicators as features in the prediction model

Model Selection

Cross Validation Technique

In order to evaluate all the hyperparameters, we use the Leave-One-Out Cross Validation [LOOCV] method which is generally to models for time series forecasting [4]. In general, leave-one-out cross-validation for regression can be carried out using the following steps.

1. Remove observation t from the data set, and fit the model using the remaining data.Say $y*(t)$ is the predicted value. Then compute the error $e(t) = y(t) - y*(t)$ for the omitted observation.
2. Repeat step 1 for t= m,…,N. Here, I start from m instead of 1, because I assume that $(m-1)$ years are minimally required to predict the time series correctly.
3. Compute the MSE from $e(m),\ e(m+1), ...e(N)$.

Baseline Model

As already mentioned, I used linear regression of all the indicators as the baseline prediction model.

Training Matrix Representation

Suppose we have different time series for 2 predictor variables *Ind1* and *Ind2,*

$Ind1 = [a1,\ a2,\ a3,\ ,\ a(t-1),\ a(t),\ a(t+1),\ ...\ ,\ aN]$,

$Ind2 = [b1,\ b2,\ b3,\ ,\ b(t-1),\ b(t),\ b(t+1),\ ...\ ,\ bN]$

Now, for the purposes of one iteration of the above mentioned LOOC technique, we need to remove one observation before regressing,

$Ind1(!t) = [a1,\ a2,\ a3,\ ,\ a(t-1),\ a(t+1),\ ...\ ,\ aN]$,

$Ind2(!t) = [b1,\ b2,\ b3,\ ,\ b(t-1),\ b(t+1),\ ...\ ,\ bN]$

Now, the training matrix for this iteration can be represented as the transpose of the concatenation of these 2 variables, ie. each row will indicate the year for which the values correspond. Thus, values of the indicators combined for a year constitute one data point.

$$Training\ Matrix\ =\ transpose\ (\ [Ind1(!t); Ind2(!t)\ ]\ )$$

Now, if *Ind* is the indicator to be predicted,

$$Training\ Labels\ =\ transpose\ (\ Ind(!t)\ )$$

Autocorrelation Model

In this model, I simply used autocorrelation to predict the value of the indicator. In this case, I required to chose a value m, the minimum number of observations needed to predict the value for a particular year. These would be my features in the training. Since, in this case, my prediction value depends on the previous years' observations, I can start predicting only after m years. Here, I chose my prediction years to 2006, 2007 only (the observations left out in cross validation). Thus, for predicting 2006, I would use values from 1972 - 2005 while for predicting 2007, I would use values from 1973 - 2006.

Selected Feature Model

For indicators with a low auto correlation coefficient, this model is used. In the selected feature model, I look up the most highly correlated indicator for the given indicator and country and use that as the training model. Similar to the autocorrelation model, I chose my prediction years to 2006, 2007 only (the observations left out in cross validation). Thus, for predicting 2006, I used values from 1972 - 2005 of the correlated indicator, and for predicting 2007, I used values from 1973 - 2006 of the correlated indicator.

Hyperparameters - Choice of Classifier

I tried two different linear classifiers - SVM and Nearest Neighbours. The decision to experiment only with linear classifiers was taken in light of the results of the auto correlations which showed a high Pearson coefficient for most indicators. Since Pearson assumes strong linear relationship, I think this is reasonable.

Results

As mentioned in the Autocorrelation section previously, because of most of the indicators for countries worked in most cases.Hence, we concentrate on the indicators with a low auto correlation coefficient. It would not make sense to report accuracy across countries or indicators as each one of them has a different prediction model. Hence, I only report MSE(mean squared error) per country and indicator here. All the countries mentioned below have low auto correlation coefficients.

Classifier/Model	Baseline	Autocorrelation	Selected Feature
Linear Regression	0.00001	0.000127	0.000032
SVM	0.00001	0.000127	0.00001
Nearest Neighbours(2)	0.000055	0.000127	0.000025

MSE for Belize: Achieve Primary Education (auto corr coeff = 0.18)

Classifier/Model	Baseline	Autocorrelation	Selected Feature
Linear Regression	2.053834	0.007236	0.00459
SVM	0.003687	0.007236	0.003687
Nearest Neighbours(2)	0.009182	0.007236	0.010658

MSE for Central African Republic: Achieve Primary Education (auto corr coeff = 0.138)

Classifier/Model	Baseline	Autocorrelation	Selected Feature
Linear Regression	**0.002885**	**0.000033**	**0.000029**
SVM	**0.000023**	**0.000033**	**0.000023**
Nearest Neighbours(2)	**0.000036**	**0.000033**	**0.000036**

MSE for Macedonia: Achieve Primary Education (auto corr coeff = 0.031)

Classifier/Model	Baseline	Autocorrelation	Selected Feature
Linear Regression	**0.006935**	**0.004059**	**0.002459**
SVM	**0.002007**	**0.004059**	**0.002007**
Nearest Neighbours(2)	**0.004471**	**0.004059**	**0.005571**

MSE for Dominican Republic: Achieve Primary Education (auto corr coeff = 0.06)

As in seen in most of the results above, the selected feature model performs much better than the autocorrelation model for low auto correlation values. It also performs better than the baseline method in some cases, which confirms my hypothesis, that baseline method may add noisy features to the prediction model, which are avoided by the selected feature model.

In terms of the classifier, we can clearly see that SVM performs better in almost all cases, across countries as well as across indicators. So, SVM is a good potential classifier for the prediction model.

Challenges & Future Work

Missing Data in Time Series

The given dataset had a lot of missing data, especially for older years. When dealing with time series data, one often has to make sure to handle missing values. For the project I used the *'backfill'* option of Pandas dataframe to fill in missing values. This is simply a heuristic and some of the discrepancies in the exploratory analysis can be attributed to this methodology of handling missing data. For example, one would expect the indicator 'Literacy rate' to be most correlated to 'Achieve Universal Primary Education' in any country. But this indicator has a lot of missing values, as seen below,

30083,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,74.4052,,,Bhutan,SE.ADT.1524.LT.ZS,"Literacy rate, youth total (% of people ages 15-24)"

Future work would include using 'Imputer' in scikit learn to better handle missing values. Another method could be using expectation maximization to build a model and then predicting the missing values from it.

Working with Small Scale Time Series Data

Though a linear regression model may seem to give the best prediction accuracy, it is difficult to validate it unless more data is available at hand. Also, in this project, I have only done one-step forecasting. Multi-step forecasting will require better classifier techniques. As many literatures suggest, Neural networks could be explored for the same.

Accounting for Outliers

Sometimes, sudden changes may affect the indicated to be forecast. For example, a natural calamity in a country may bring down the GDP for that particular year. Modeling such interventions would lead to better prediction accuracy. This can be done using a spike variable. This is a dummy variable taking value one in the period of the intervention and zero elsewhere [4].

Conclusion

Time Series analysis and prediction is often difficult because of small sized data, especially when the data is annual. In this project, though most of the time series indicators could be predicted well using auto correlation, it was interesting to look at the ones which did not have any auto correlation. The devised method to select features for such an indicator seemed to perform well than the baseline method. This was possible because it avoided the noisy feature that were incorporated by the baseline method.

References

[1] DrivenData, http://www.drivendata.org/competitions/1/
[2] World Bank Open Data, http://data.worldbank.org/
[3] Milleninum Development Goal Indicators, http://unstats.un.org/unsd/mdg/Host.aspx?Content=Indicators/OfficialList.htm
[4] Forecasting - Principles and Practice (https://www.otexts.org/fpp)
[5] Correlation based Feature Selection for Machine Learning (http://www.cs.waikato.ac.nz/~mhall/thesis.pdf)
[6] Correlation coefficient and p values: what they are and why you need to be very wary of them (http://www.eecs.qmul.ac.uk/~norman/blog_articles/p_values.pdf)
[7] G. Bontempi, S. Ben Taieb, and Y. Le Borgne. "Machine Learning Strategies for Time Series Forecasting." In Business Intelligence , pp. 62-77. Springer Berlin Heidelberg, 2013
[8] Honaker J, King G. 2010. What to do about missing values in time series cross-section data. Am.J.Polit.Sci. 54:561–81