IBM Machine Learning Certification
Supervised Machine Learning - Regression
Course Project
WHO Life expectancy
Gábor Sárosi
2020

Main objective

The main objective of this project is to find the best model (if exists) to predict the life expectancy in a given country at a given year based on different (medical, social, economical) factors, for example prevalence of deceases immunisation, alcohol consumption, schooling, etc.

Describing the data

Source: https://www.kaggle.com/mathchi/life-expectancy-who-with-several-ml-techniques/data

The publicly available Life Expectancy Data on Kaggle, collected from World Health Organization (WHO. Is a dataset with important health-related factors collected in 193 countries between 2000 and 2015.

The dataset has 22 columns:
The target variable is highlighted with yellow.

Name of column	Type of data	Dtype	Description
Country	Categorical	object	Name of country
Year	Numerical	int64	Year of observation
Status	Categorical	object	Development status
Life expectancy	Numerical	float64	Life expectancy in years
Adult Mortality	Numerical	float64	Probability of dying between 15 and 60 years per 1000 population
infant deaths	Numerical	int64	Number of infant deaths for every 1,000 live births
Alcohol	Numerical	float64	Alcohol, recorded per capita (15+) consumption (in litres of pure alcohol)
percentage expenditure	Numerical	float64	Expenditure on health as a percentage of Gross Domestic Product per capita(%)
Hepatitis B	Numerical	float64	Hepatitis B (HepB) immunization coverage among 1-year-olds (%)
Measles	Numerical	int64	Measles - number of reported cases per 1000 population
BMI	Numerical	float64	Average Body Mass Index of entire population
under-five deaths	Numerical	int64	Number of under-five deaths per 1000 population
Polio	Numerical	float64	Polio (Pol3) immunization coverage among 1-year-olds (%)
Total expenditure	Numerical	float64	General government expenditure on health as a percentage of total government expenditure (%)
Diphtheria	Numerical	float64	Diphtheria tetanus toxoid and pertussis (DTP3) immunization coverage among 1-year-olds (%)
HIV/AIDS	Numerical	float64	Deaths per 1 000 live births HIV/AIDS (0-4 years)
GDP	Numerical	float64	Gross Domestic Product per capita (in USD)
Population	Numerical	float64	Population of the country
thinness 1-19 years	Numerical	float64	Prevalence of thinness among children and adolescents for Age 10 to 19 (%)
thinness 5-9 years	Numerical	float64	Prevalence of thinness among children for Age 5 to 9(%)
Income composition of resources	Numerical	float64	Human Development Index in terms of income composition of resources (index ranging from 0 to 1)
Schooling	Numerical	float64	Number of years of Schooling(years)

fig.1: columns description of the original dataset

The dataset has 2938 rows (some countries have no data for some years) and 22 columns (see above).

Exploratory Data Analysis and Data Cleaning

Dealing with missing values.

The dataset has some columns with a high number of missing values

Population	652
Hepatitis B	553
GDP	448
Total expenditure	226
Alcohol	194
Income composition of resources	167
Schooling	163
thinness 5-9 years	34
thinness 1-19 years	34
BMI	34
Polio	19
Diphtheria	19
Life expectancy	10
Adult Mortality	10
HIV/AIDS	0
Country	0
Year	0
Measles	0
percentage expenditure	0
infant deaths	0
Status	0
under-five deaths	0

Fig. 2: Number of missing values by columns

I decided to drop columns with over 200 missing values. The columns: 'Population', 'Hepatitis B', 'GDP', 'Total expenditure', had to go.

Dealing with categorical data

The dataset had 2 columns with categorical data categorical data (see fig.1).
'Country' had to go, as it does not have any value for our purpose (It could be an interesting hypothesis though, but testing that is outside the goal of this paper).

To keep it simple, I dropped 'Status' as well, as it correlates with many features.

Multi-Collinearity

Then I checked the correlation matrix of the now smaller dataset (using only the lower half, as the tl-br axis mirrors it)

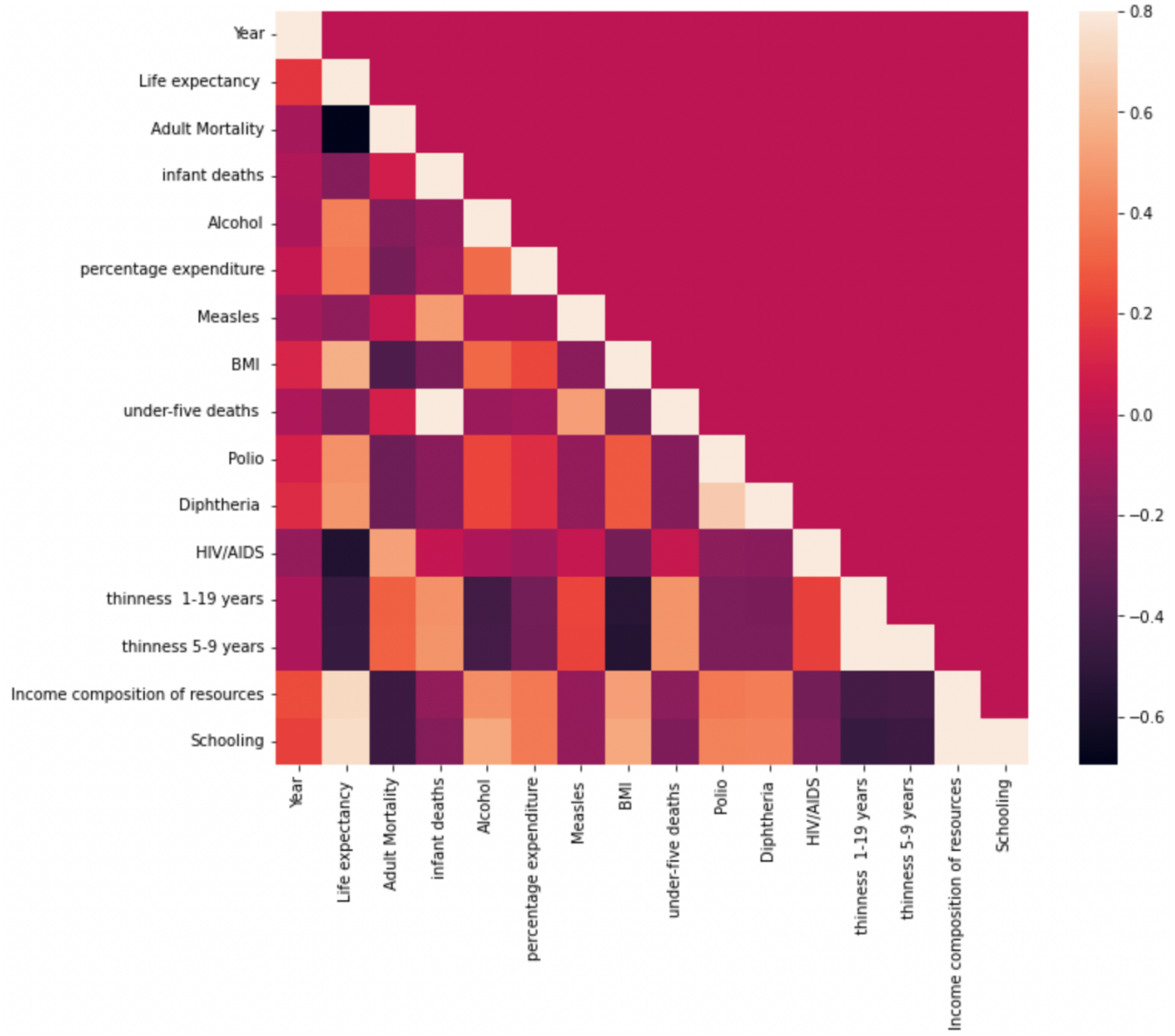

Fig.3: Correlation matrix

Pairs with a high (>.75) correlation score:

```
Features-pairs with over 0.8 Multi-Collinearity
under-five deaths  and infant deaths --> Correlation Score 0.996628882039801
 thinness 5-9 years and  thinness  1-19 years --> Correlation Score 0.93910199219146
Schooling and Life expectancy  --> Correlation Score 0.7519754627367001
Schooling and Income composition of resources --> Correlation Score 0.8000924203919638
```

Therefore I decided to drop these columns:
'infant deaths', ' thinness 5-9 years', 'Income composition of resources'

Before staring with the linear regression models, I decided to check if the target data (Life Expectancy) is normally distributed

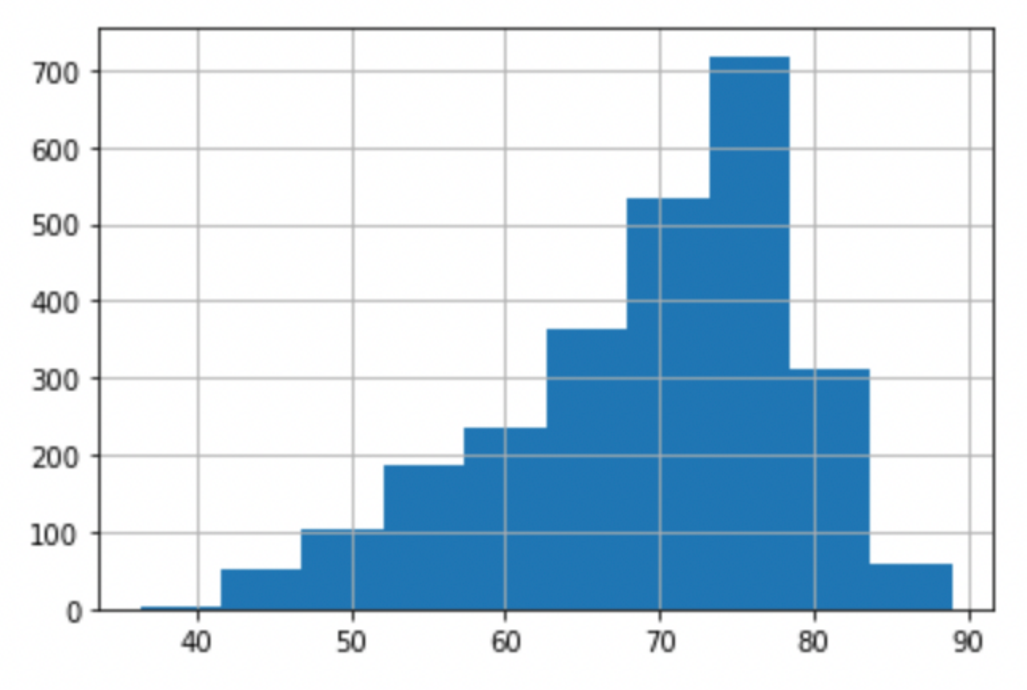

Fig.4: Histogram of the column ‘Life expectancy’

It is left-skewed, the normaltest shows a pvalue=2.1939621267620458e-37

Transforming tricks, like “squared”, or “boxcox” showed some small improvement, but the p-value of the normaltest always remained under the 0.5 threshold, so I decided to keep the target unchangend.

Linear Regression models

The baseline model is a simple linear regression with the using MinMax scaler on the data and a train-test split of 0.3.

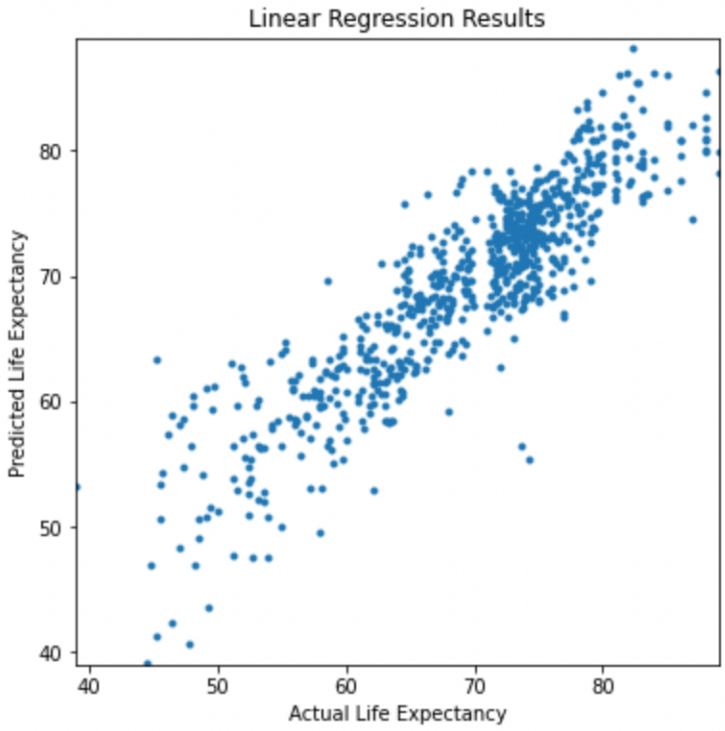

Fig.5: Plotting Baseline LR Results

The R-squared score was
0.7586518656967627

Not very high, but for the baseline it is ok.

Now, adding polynomial features with the degree of 2:

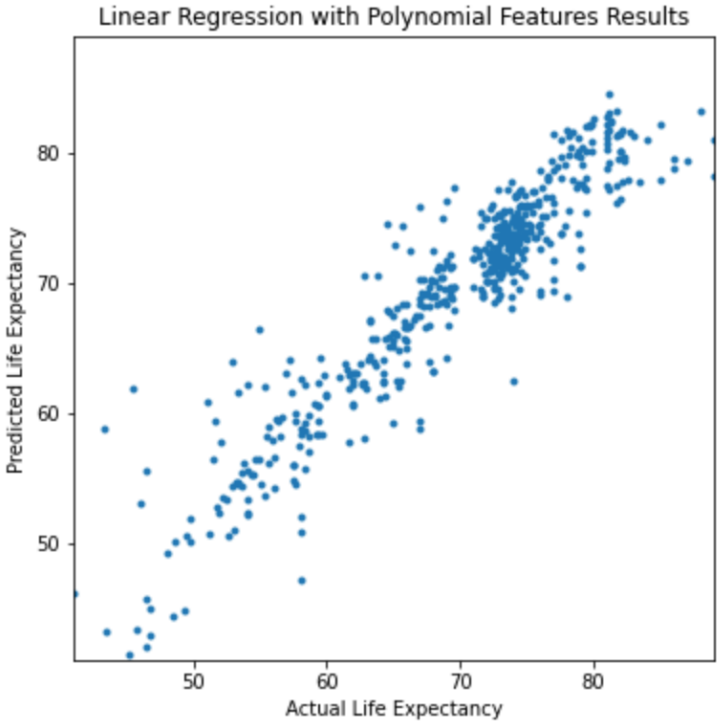

Fig.6: Plotting results of LR with Polynomial Features

r2-score now
0.8607548230989525

This is a significant increase! The plot also looks more aligned with the diagonal.

Let's see some regularisation!

My first choice was Lasso with alpha = 0.1

```
sum of coefficients: 61.132521563703136
number of coefficients not equal to 0: 8
```

Fig.5: Coefficients for Lasso with alpha=0.1

And now, the R-squared scores of some alphas (still Lasso)

```
Lasso alpha 1 R2: 0.27811290302835645
Lasso alpha 0.5 R2: 0.6065599388964806
Lasso alpha 0.1 R2: 0.79361012364358
Lasso alpha 0.05 R2: 0.8103530807041384
Lasso alpha 0.01 R2: 0.8553896576165068
Lasso alpha 0.005 R2: 0.8762366209120942
Lasso alpha 0.001 R2: 0.885813139015448
Lasso alpha 0.0005 R2: 0.884075970841822
Lasso alpha 0.0001 R2: 0.8782931984035807
Lasso alpha 5e-05 R2: 0.8758248927644665
Lasso alpha 1e-05 R2: 0.873779463455158
```

Fig.6: R2-scores for some alphas of Lasso regularisation

So far, the best model is lasso with alpha=0.001. Its plot looks like this:

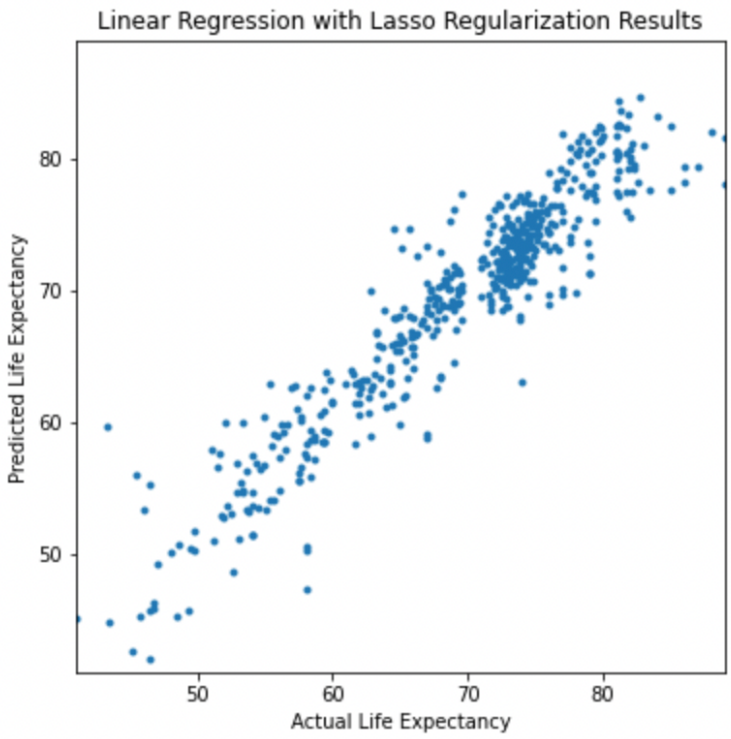

Fig.7: Plot for Lasso regularisation (alpha=0.001)

Now, the same with Ridge regularisation:

```
Ridge alpha 1 R2: 0.8693707282112473
Ridge alpha 0.5 R2: 0.8765300387371715
Ridge alpha 0.1 R2: 0.8817344590532453
Ridge alpha 0.05 R2: 0.8812040706441591
Ridge alpha 0.01 R2: 0.8784109856824199
Ridge alpha 0.005 R2: 0.8771531936786897
Ridge alpha 0.001 R2: 0.8747211948738214
Ridge alpha 0.0005 R2: 0.8740827177429323
Ridge alpha 0.0001 R2: 0.8734503838295158
Ridge alpha 5e-05 R2: 0.8733819073451943
Ridge alpha 1e-05 R2: 0.873351658623448
```

Fig.8: R2-scores for some alphas of Ridge regularisation

The best among the Ridge regularised models is the one with an alpha of 0.1. Here is the plot:

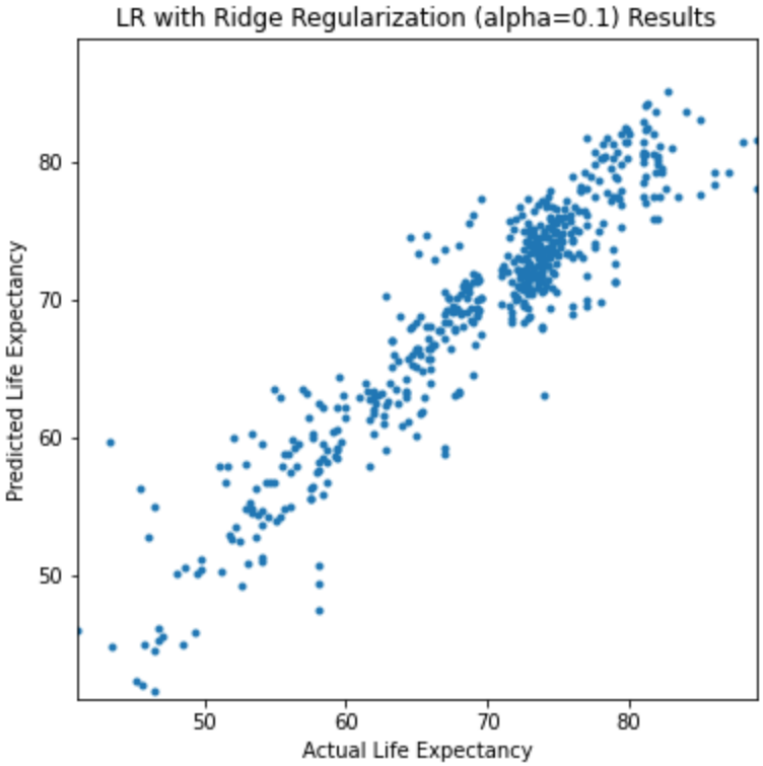

Fig.9: Plot for Ridge regularisation (alpha=0.1)

Conclusion

So far, the best model was the Linear Regression Model with degree 2 polynomial features and Lasso regularisation with an alpha of 0.001. It has an r-squared score of 0.885813 At this point I recommend this one as the final model.

Next steps

For further improvement I recommend the following:

1. Collect more data, and check the accuracy of the dataset. (Some governments might have lower credibility when they provide data, which can be an indication of their unsuccessful governance or hurt national pride)
2. In the EDA and Data Cleaning section, try different methods, for example filling the missing values with the mean, instead of getting rid of the “bad columns”. Trimming the outliers could also improve the prediction.
3. In the LR section adding Cross-validation methods (LassoCV, RidgeCV), and/or further fine-tuned alphas