

DAT565/DIT407 Assignment 4

Vaibhav Talari
talari@chalmers.se

Stefán Ólafur Ingimarsson
stefanla@chalmers.se

December 2, 2024

This report is submitted by group 25 for **assignment four** in *Introduction to Data Science & AI*.

1 Problem 1: Splitting the data

The purpose of our train-test split was to predict life expectancy at birth. The target variable of our predictive model was "Life expectancy at birth, both sexes(years)." This value represents the average number of years that a newborn is expected to live, assuming that current mortality rates remain constant throughout their life. Life expectancy is based on a variety of factors that influence human life. The training process involves teaching the model to learn patterns like health care and literacy rates.

We divide the data into 80% training data and 20% testing data. Since the overall data set has 5554 rows, it is appropriate to allocate more data points to training.

2 Problem 2: Single-variable model

Life expectancy at birth increases as human development increases. The Human Development Index is a statistic developed and compiled by the United Nations to measure various countries' levels of social and economic development. One of the main areas of interest is life expectancy. The country's education level is also considered when calculating the HDI, as well as the economic metric of the country.[1]

The plot shows that life expectancy increases as the human development index increases, which makes sense since life expectancy is a key factor in the calculation of the human development index. We have included the resulting graph in figure 1.

3 Problem 3: Nonlinear relationship

To select a variable that has a nonlinear relationship with the target variable, we perform an analysis to find the Spearman coefficient, which gives the relationship between the candidate variable and the target variable as a monotonic

function. The target variable is **life expectancy at birth**. We further narrowed down the search by finding a variable such that the difference between Spearman and Pearson correlation is high (Spearman - Pearson) because if the difference is high then Pearson correlation is low stating a non-linear relationship. Based on this, the following observations are made.

Selected candidate variable **Gross National Income Per Capita (2017 PPP\$)**

Function used: logarithmic transformation. In logarithmic transformation, we simply normalize all the values of the column by minimal value to avoid errors during log calculation if the value is negative or zero.

Pearson **before** applying the transformation: 0.6514708331957301

Spearman **before** applying the transformation: 0.8648281379988499

Pearson **after** applying the transformation: 0.8286706182772113

Spearman **after** applying the transformation: 0.8648281379988499

4 Problem 4: Multiple linear regression

To find a subset of variables suitable for the multi-variable linear regression model, we first get the list of all columns, excluding the columns mentioned as described in the question. We calculate the coefficient of determination (R^2), mean squared error, Pearson coefficient, and intercept for each of the candidate columns against the target variable. Once we calculate all this data we sort by the coefficient of determination as this tells us how good the of a linear relation the candidate variable has with the target variable. For example, if we get a coefficient of determination as 0.9 then there is a strong linear relationship. Once we have sorted the data based on the coefficient of determinant, we run a loop which keeps adding all the candidate variables from the sort list in accessing order to add all the strongest linear relationship variables into the multi-variable model. Furthermore, we keep checking if the coefficient of determination outperforms the single variable model and once it does we break out of the loop. Name of variables:

β_1 Crude Birth Rate (births per 1,000 population)

β_2 Total Fertility Rate (live births per woman)

β_3 log_transform (*this is the transformed variable from problem 3*)

β_4 Adolescent Birth Rate (births per 1,000 women ages 15-19)

β_5 Expected Years of Schooling (years)

β_6 Expected Years of Schooling, female (years)

β_7 Median Age, as of 1 July (years)


β_8 Net Reproduction Rate (surviving daughters per woman)

Coefficients of the model:

$\beta_1 -3.89724681e - 01$

β_2 $-9.88362950e + 00$


β_3 $1.90638381e + 00$

β_4 $-1.07051217e - 02$ 

β_5 $2.93376972e - 01$

β_6 $-2.61906186e - 01$

β_7 $2.98152842e - 01$

β_8 $2.84437400e + 01$ 

Model intercept: 45.087181883644604

Model coefficient determination: 0.8815633649159162

Pearson correlation coefficient between predicated and actual by the model:
0.9344246974943984

'MSE between predicated and actual by the model': 10.399592319797634

5 Assignment code

Here is our code that shows how we performed the train-test of the data. As well as calculating the non linear relationship.

```
1 # train-test split
2 # handle missing values
3 if df.isnull().any().sum() > 0:
4     print("Handling missing values...")
5     # fill missing values for numeric columns with median
6     numeric_cols = df.select_dtypes(include="number").columns
7     df[numeric_cols] = df[numeric_cols].fillna(df[numeric_cols].median())
8
9 base_data = df.drop
10 (columns=["Life_Expectancy_at_Birth, both_sexes (years)"])
11 base_data_label = df["Life_Expectancy_at_Birth, both_sexes (years)"]
12
13 train_data, test_data, train_data_label,
14     test_data_label = train_test_split(
15     base_data, base_data_label, test_size=0.2, random_state=42
16 )
17
18 print("Train-test split successful")
19
20 # prob3.
21 # selecting a non linear but monotonic relation ship with target variable
22 def find_nonlinear_rel(col):
23     pearson_corr = train_data[col].corr
24     (train_data_label, method="pearson")
25     spearman_corr = train_data[col].corr
26     (train_data_label, method="spearman")
```

```

27
28     return {
29         "Variable": col,
30         "Spearman": spearman_corr,
31         "Pearson": pearson_corr,
32         "Difference": abs(spearman_corr) - abs(pearson_corr),
33     }
34
35 results = []
36 # using juse train_data.columns gives
37 # error: could not convert string to float: 'Mauritius'
38 for col in train_data.select_dtypes(include=["number"]).columns:
39     results.append(find_nonlinear_rel(col))
40 results_df = pd.DataFrame(results)
41
42 # we check spearman greate than 0.7 to
43 # have a strong monotonic relationship
44 # pearson cooefficient gives the linear cooreleation
45 # difference of spearman-pearson if higher means
46 # pearson cooefficient is less meaning
47 # less linear relationship
48 non_linear_monotonic_vars = results_df[
49     (results_df["Spearman"].abs() > 0.7) & (results_df["Difference"] > 0.2)
50 ]
51
52 # Sort by Spearman correlation
53 non_linear_monotonic_vars = non_linear_monotonic_vars.sort_values(
54     by="Spearman", ascending=False
55 )
56
57 non_linear_monotonic_vars

```

References

- [1] The Investopedia team. “What is the human development index(HDI)?” In: (2024). URL: <https://www.investopedia.com/terms/h/human-development-index-hdi.asp>.

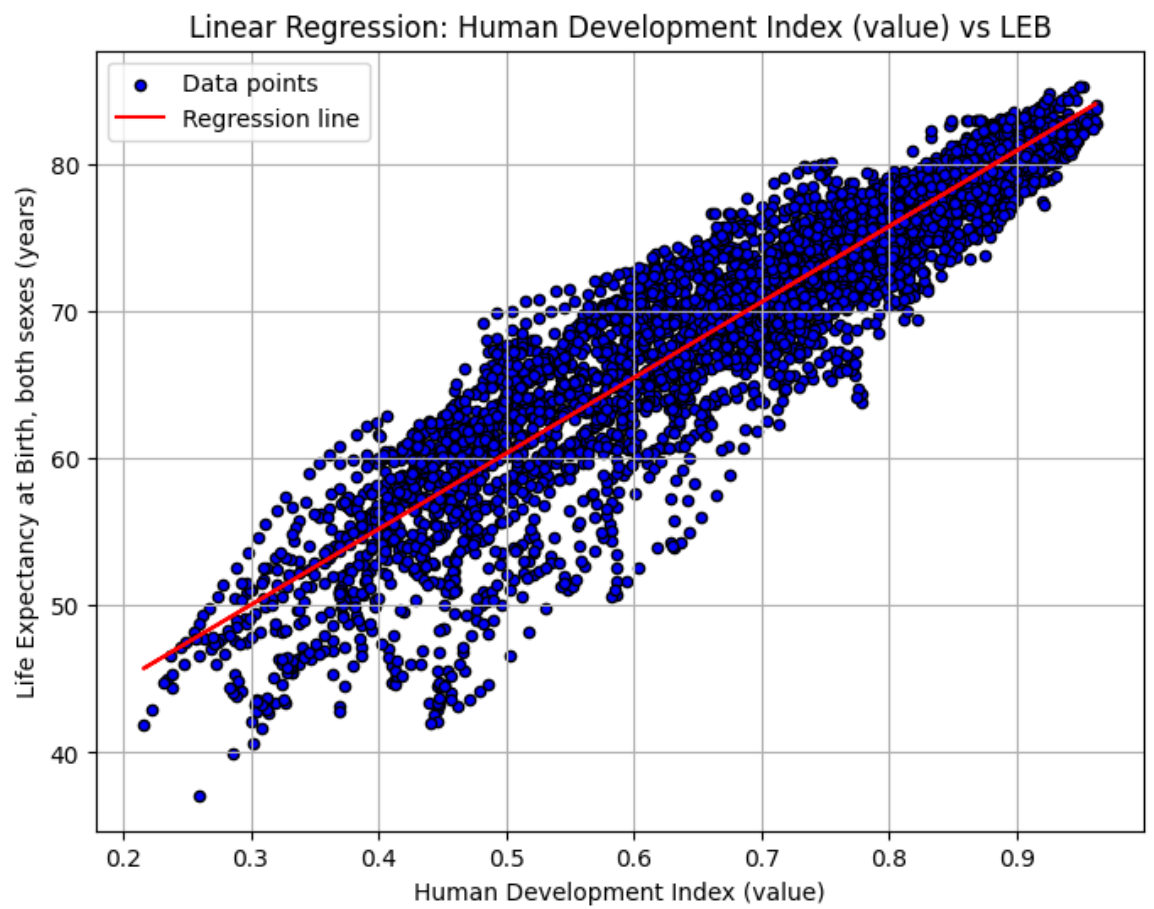


Figure 1: Scatter Plot of LEB against HDI