Analysis of World Happiness Index

Aditya Firman Ihsan

Introduction

The World Happiness Report is a landmark survey of the state of global happiness, firstly published in 2012.

The reports review the state of happiness in the world today and show how the new science of happiness explains personal and national variations in happiness.

We analyze the 2018-2019 report to see correlations between some aspects of national happiness.

Goal:

To understand the effects between aspects in regard of national happiness

Outcome:

Most influential factor of national happiness

Metadata of The Dataset

1. Overall rank
2. Country
3. Score
4. GDP per capita
5. Social support
6. Healthy life expectancy
7. Freedom to make life choices
8. Generosity
9. Perceptions of corruption

World Happiness Report 2018-2019

Owner: UN Sustainable Development Solutions Network

License: CC0 Public Domain

Last Updated: Nov 27th, 2019

Dimension: 156 countries x 9 columns

Data Preparation

Implementation

1) Missing Data Imputation

A few data in some features of the dataset has value 0 and N/A, Of course this value doesn't have realistic meaning, so its need to be filled over.

I used median value of each feature to fill the missing data.

2) Grouping Countries

The dataset have 156 unique values of country/region name. It won't be useful.

I applied bucketing techniques to generate continents data as additional categorical feature

Implementation

3) One Hot Encoding Continents

Continents feature from previous step is one-hot encoded to 6 sparse feature.

Note: I used another dataset from kaggle to obtain continents data (https://www.kaggle.com/folaraz/world-countries-and-continents-details)

4) Binning GDP per Capita

GDP per capita represents the developing level of respective countries. It will be more useful to use it as binned value, which divides countries based on its income (low, lower-mid, upper-min, high).

I used 4 bins with equal range.

Implementation

5) One Hot Encoding GDP Bins

Binned value of GDP from previous step is one-hot encoded to 4 sparse features.

6) Range Scaling Score

Score data have quite different range than the rest of the numerical features.

Thus, I rescaled the score data so that it has 0-1 numerical value.

Implementation

7) Clipping Generosity

I detected that Generosity data has two outliers (has value greater than triple of standard deviation).

Rather than dropping it, I clipped these outliers to preserve the number of data.

8) Log-Scaling Perception of Corruption

Perceptions of Corruption have quite large range of values. I log-scaled it to reduce the range magnitude.

Preparation Results

Grouping

After

	Country or region ▲	AF	AS	EU	OC	SA	Other
1.	Afghanistan	0	1	0	0	0	0
2.	Albania	0	0	1	0	0	0
3.	Algeria	1	0	0	0	0	0
4.	Angola	1	0	0	0	0	0
5.	Argentina	0	0	0	0	1	0
6.	Armenia	0	1	0	0	0	0
7.	Australia	0	0	0	1	0	0
8.	Austria	0	0	1	0	0	0
9.	Azerbaijan	0	1	0	0	0	0
10.	Bahrain	0	1	0	0	0	0
11.	Bangladesh	0	1	0	0	0	0

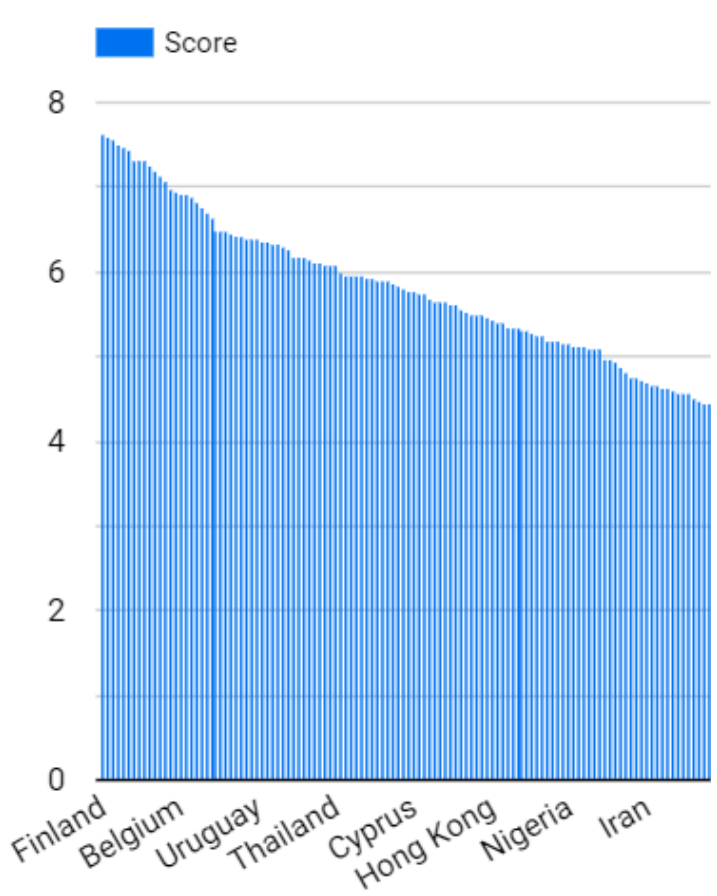
After

Binning

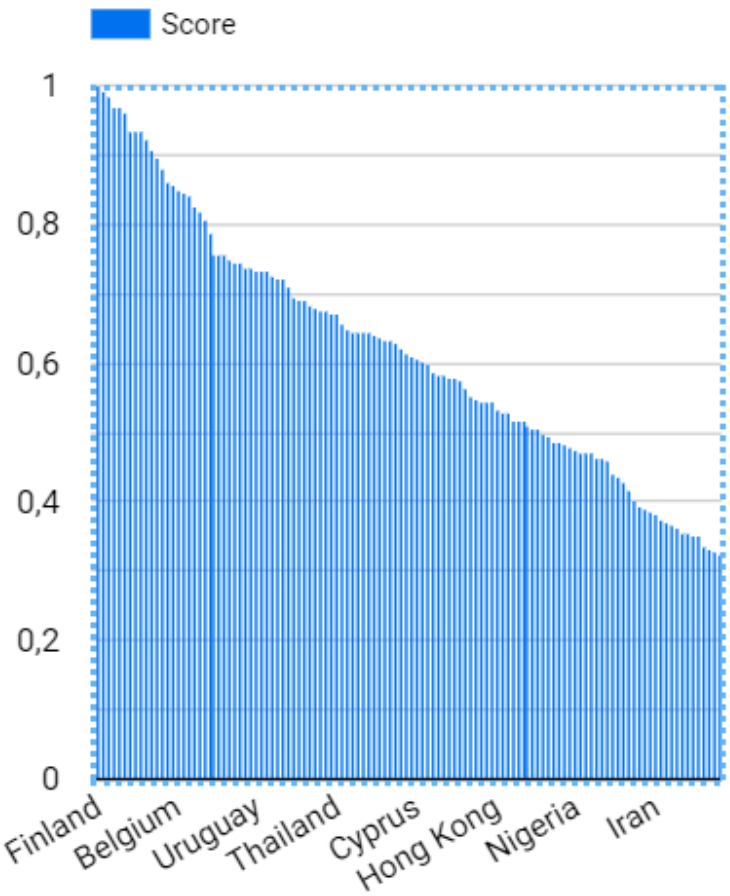
	Country or region	Low Income	Lower-Middle...	Upper-middle In...	High Income
1.	Afghanistan	1	0	0	0
2.	Albania	0	1	0	0
3.	Algeria	0	1	0	0
4.	Angola	0	1	0	0
5.	Argentina	0	0	1	0
6.	Armenia	0	1	0	0
7.	Australia	0	0	1	0
8.	Austria	0	0	1	0
9.	Azerbaijan	0	1	0	0
10.	Bahrain	0	0	1	0
11.	Bangladesh	1	0	0	0

Range Scaling

Before

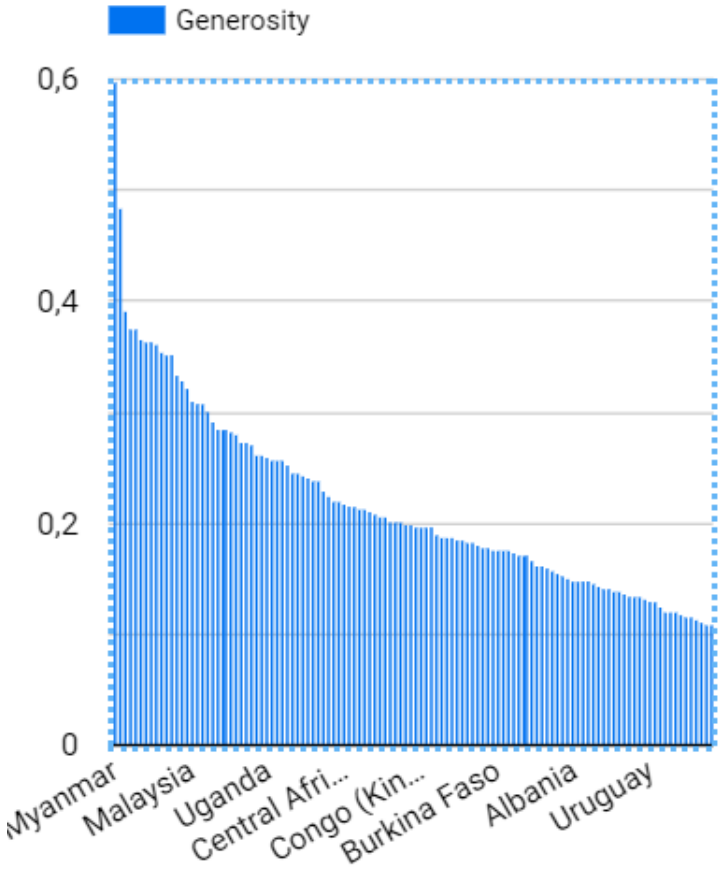


After

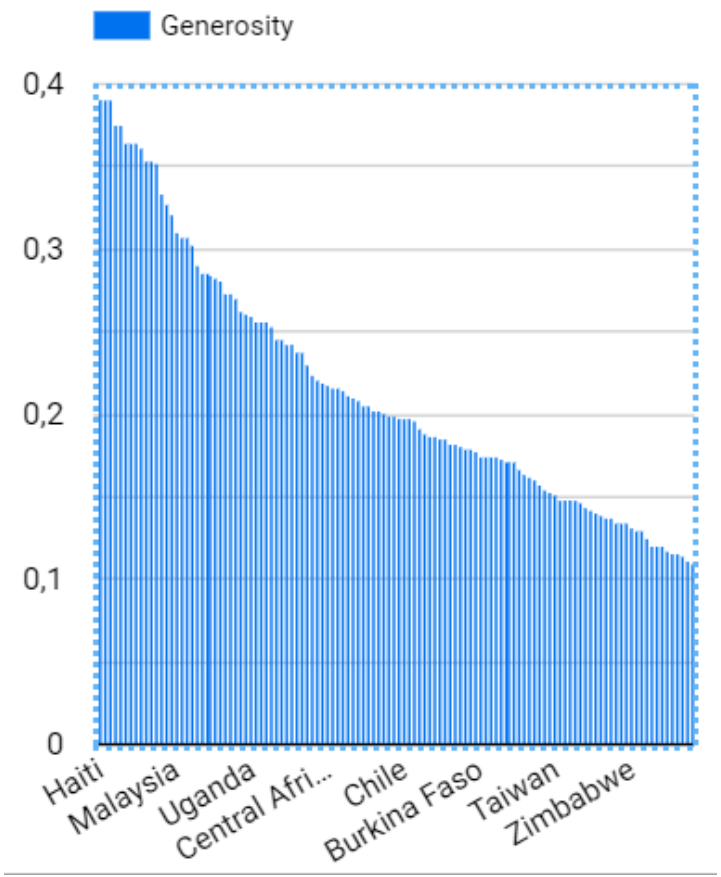


Clipping

Before

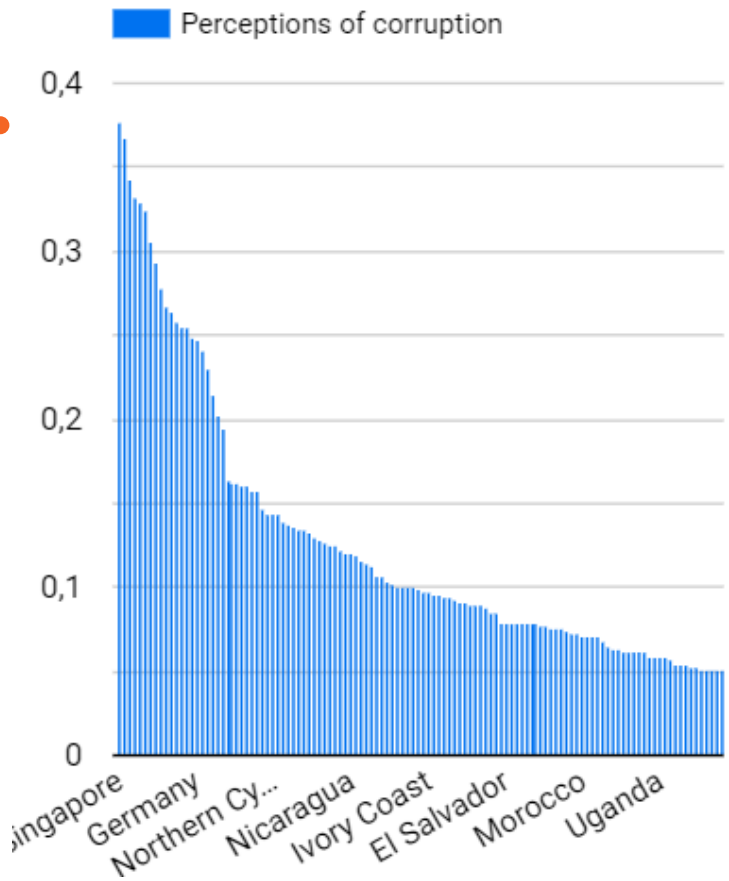


After

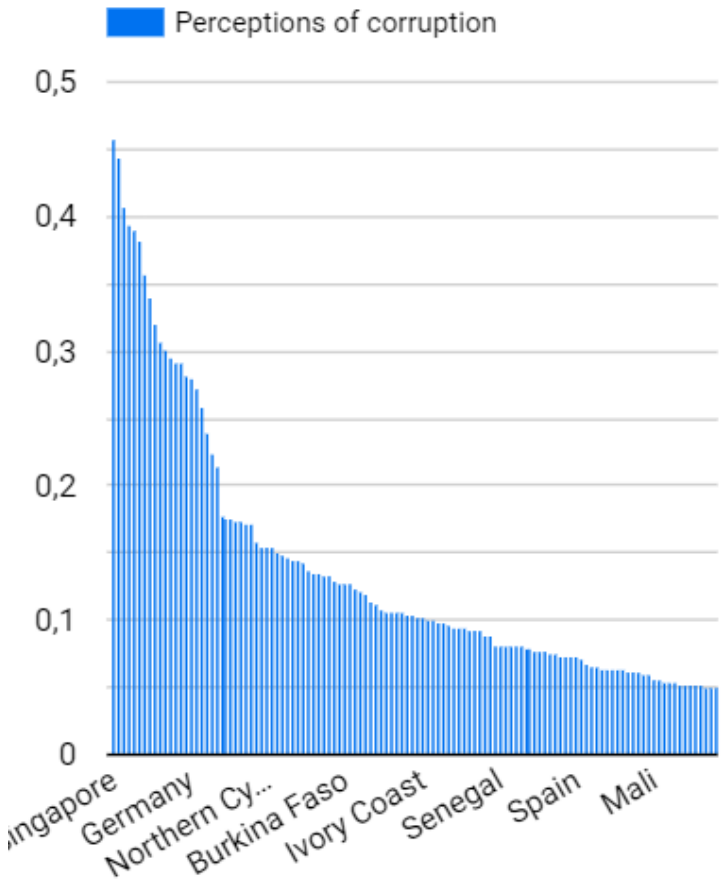


Log-Scaling

Before



After



Preparation Result Insight

1) Unique data identifier, such as country name, usually is not very useful. It turns out grouping it to continents will utilize it to be useful features.

2) Outliers aren't always to be dropped or ignored.

3) Binning should be applied carefully. GDP bins I created are not equally distributed. Proportion of high income countries is very small

4) Log-scale is not very effective to be applied in small scaled data, such as my result in perceptions of corruption feature.

Regression Analysis

Preparation

- Preparing Data

The data is already in clean table of csv file for every year. We combine the dataset in 2018 csv file with the 2019 one to obtain 312 data. We apply data preparation mentioned before.

- Deciding best feature

The data consists of happiness score with 6 key factors. We tried each factor as a feature to model a linear regression of happiness index and observed the error.

- Tuning the hyperparameter

We tweaked the learning rate, steps number, and batch size of the model to effectivize the regression process.

Techniques

Linear Regression model using TensorFlow Estimator v2.1.0

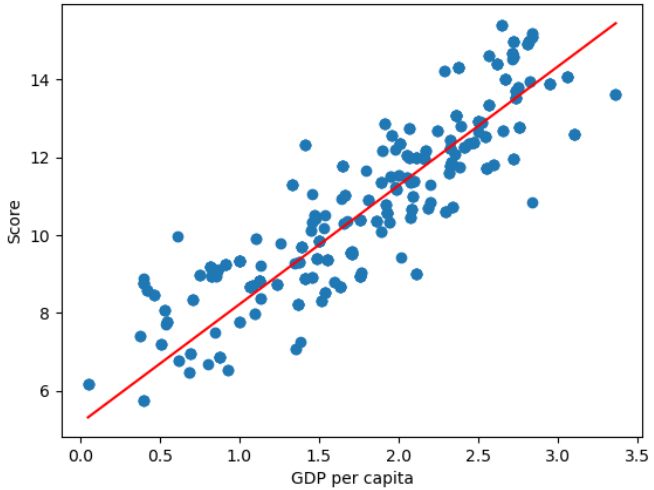
Stochastic Gradient Descent Optimization

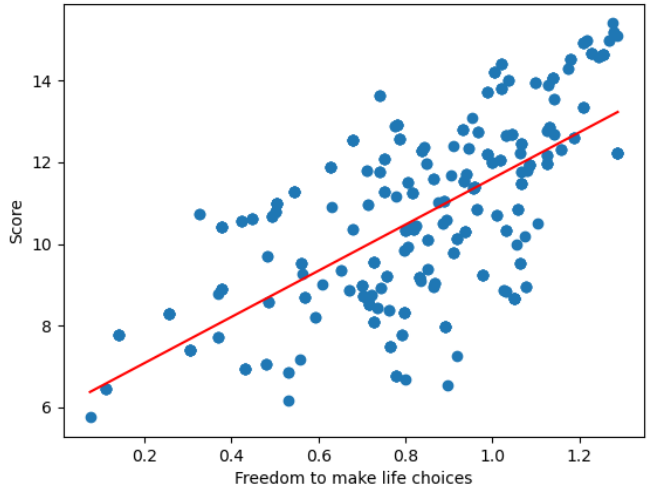
Result (Factor comparison)

Factor variable	Root Mean Square Error	Weight
Healthy Life Expectancy	1.185	4.4635
Social Support	1.329	3.2135
GDP per capita	1.128	3.064
Freedom of life choice	1.669	5.648
Generosity	2.303	5.1499
Perception of corruption	2.111	4.664

Result (Feature analysis)

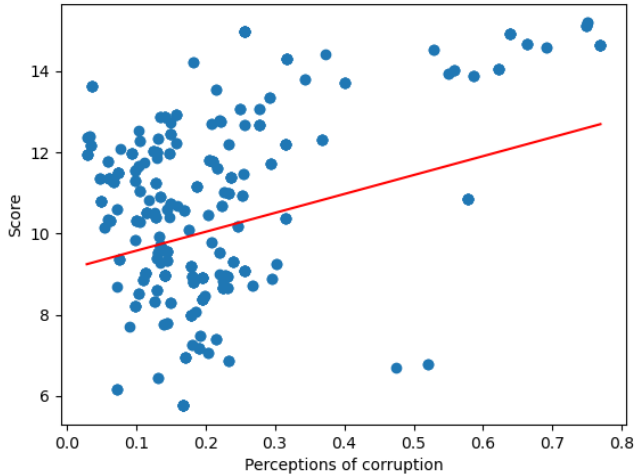
Freedom of life choice has the greatest weight, means it influence happiness index the most .
GDP per capita has the lowest RMSE, means that it follows a linear trend nicely. Its effect on Happiness score is more direct. But, it has the lowest weight. small increase in GDP only affects little increase in happiness score.

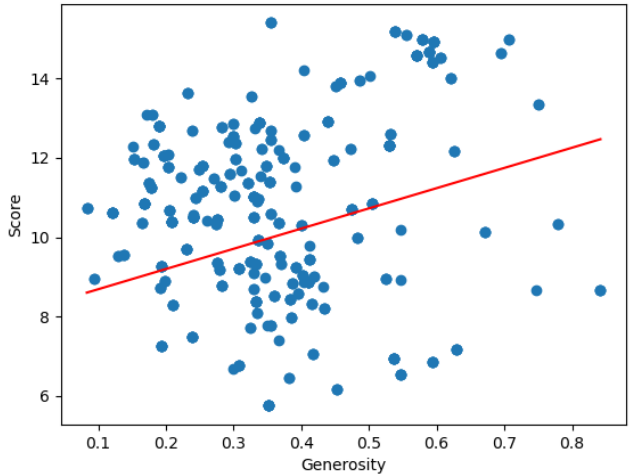




Result (Feature analysis)

On the other hand, **generosity** and **perception of corruption** have the biggest errors. It means their data spreads widely. We can see that they even don't have any readable trends.





Conclusion

By the model result, we conclude,

- GDP per capita is one factor that directly affects happiness score
- Freedom of choice has the most effect on happiness score
- Generosity and perceptions of corruption don’t have any meaningful correlation to happiness score.

It means,

- Individual wealth is a direct cause of happiness
- People are happy when they are in liberty
- People don’t care much about any aspect outside self needs

—

Notes!

The dataset can be accessed in
https://www.kaggle.com/unsdsn/world-happiness

Some interactive graphics of the result can be accessed in
https://datastudio.google.com/reporting/9d631348-8f00-4fca-8b1f-6618d446e803

All this analyses are done directly in Python. The code can be accessed in
https://github.com/phoenixfin/tf_basics/