World Happiness Prediction Project - Year 2021 - under COVID-19

HarvardX Data Science Professional Certificate: PH125.9x

Data Science Initiative
The Palestinian Central Bureau of Statistics (PCBS)
Arab American University of Palestine (AAUP)

Mohammed K. Elhabbash

28 July, 2021

Contents

1. Introduction

Machine learning algorithms that predict the future are frequently at the forefront of current computer research. One of the most powerful applications of machine learning is guessing results autonomously given large amounts of simulated data. Exoplanet detection using light flux data to exploiting the flow of the US bond and stock markets are just a few examples. Prediction can be done using classification models, random forests, k-nearest neighbors, or a variety of other machine learning techniques, depending on the sort of data you're working with. In order to study the happiness of countries, this research uses regression analysis.

The Sustainable Development Solutions Network's World Happiness Report is characterized as "a major study of the condition of global happiness" on Kaggle. The most current poll was performed throughout 2021 and assigns a happiness score to each country. Specific societal elements are factored into the score, which aids in determining the country's standard of living. The following is a sample of our data

```
## Rows: 149
## Columns: 9
## $ Overall.rank                <int> 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13~
## $ Country.or.region           <chr> "Finland", "Denmark", "Switzerland", "Ice~
## $ Score                       <dbl> 7.842, 7.620, 7.571, 7.554, 7.464, 7.392,~
## $ GDP.per.capita              <dbl> 1.032, 1.039, 1.046, 1.037, 1.039, 1.043,~
## $ Social.support              <dbl> 0.954, 0.954, 0.942, 0.983, 0.942, 0.954,~
## $ Healthy.life.expectancy     <dbl> 0.72000, 0.72700, 0.74400, 0.73000, 0.724~
## $ Freedom.to.make.life.choices <dbl> 0.949, 0.946, 0.919, 0.955, 0.913, 0.960,~
## $ Generosity                  <dbl> -0.098, 0.030, 0.025, 0.160, 0.175, 0.093~
## $ Perceptions.of.corruption   <dbl> 0.186, 0.179, 0.292, 0.673, 0.338, 0.270,~
```

There are 149 nations mentioned in the data, with 9 useful columns. Happiness rank, nation name, total happiness score, and six criteria that help define the happiness score include GDP per capita, Social Support, Life Expectancy, Freedom, Generosity, and Perceptions ofcorruption.

The results are based on responses to the main life evaluation question in the Gallup World Poll. The *Cantril ladder* asks respondents to imagine a ladder with a 10 at the top and a 0 at the bottom, and then assess their present lives on that scale. The most closely linked with the lowest scoring countries is the iconic Orwellian dystopia, a society in which the above elements are harmful to the wellbeing of a free and open society. As a result, higher ratings effectively reflect a closer approach to a Utopia. There is some ambiguity about how the numbers for each element relate to the happiness score (debates may be found here. The influence of each societal variable on a conventional dystopian score is shown by these numbers. Higher factor scores will also result in a higher happiness score. Older versions of the happiness report included a dystopia residual that represented more broad improvements to the basic dystopian. This is not accessible in the data sets (2018, 2019), but reappears and can be accessible in the data sets (2020,2021). It's worth noting that the factor values aren't *directly* linked to total happiness. The purpose of this study is to see how closely they are linked. The total of the individual components, i.e. sum(data[$factors$]) = $data$Score, is the optimal model for estimating happiness scores, according to the previously cited arguments. This model is provided with a generalized linear model equation based on both 2020 and 2021 data. The following is how the content of the report is organized: In the **Data** phase, the data set is deconstructed and displayed. The data is partitioned in the third step, and the three modeling approaches are described in the **Methods** section. In the **Results** section, the three models are compared. Finally, in the **Conclusion** section, the findings are summarized and compared.

1.1 Dataset

The data's qualitative structure and overall connection are explained above. Quantitatively, the data provided here is normal. The happiness ratings have a roughly normal distribution, as shown by a histogram, and descriptive statistics back up this assertion.

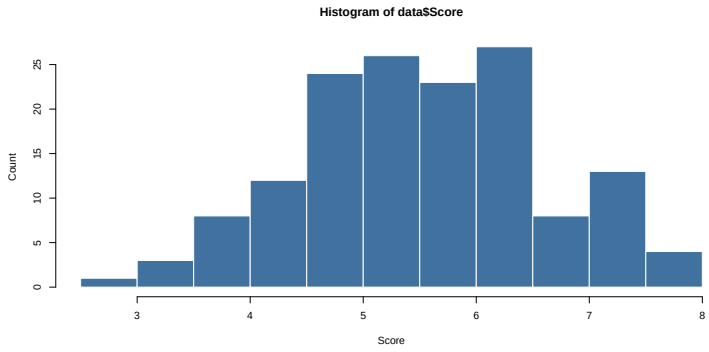

Figure 1: *Histogram of Happiness Scores for year2021*

Min.	Q_1	Median	Mean	Q_3	Max
2.523	4.852	5.534	5.5328389	6.255	7.842

Right-leaning histogram Comparing the mean (5.5328389) and median (5.534) scores helps demonstrate this quantitatively. There are more values larger than the middle when the mean exceeds the median. The possible outcomes are as follows: 2.523 ~ 7.842. Below are the top three and bottom three observations.

Overall.rank	Country.or.region	Score
1	**Finland**	**7.842**
2	**Denmark**	**7.620**
3	**Switzerland**	**7.571**

Overall.rank	Country.or.region	Score
147	**Rwanda**	**3.415**
148	**Zimbabwe**	**3.145**
149	**Afghanistan**	**2.523**

Please note that low happiness levels are more akin to dystopian civilizations. This implies that **lower happiness ratings are associated with lower factor scores**. As a result, high factor scores have a positive influence on the happiness score, whereas low factor scores have a negative effect. In the end, the most important thing is to figure out how each element interacts with the others. To investigate this, the ggcorrplot package is used.

```
## NULL
```

With the exception of Generosity, all factors are at least slightly linked (above zero). With a correlation of approximately zero, the Generosity component appears to have less of an influence on happiness scores, therefore *removing the Generosity term from the model may enhance accuracy.* By displaying particular variables and selecting the best fit line, the relationships between them may be depicted more clearly. For example, compare "Score" to "Generosity" and "Score" to "GDP per capita."

The absence of a link between score and generosity is demonstrated by a horizontal line of best fit. On the correlation map, the connection between score and generosity was close to zero. In contrast, the plot of Score and GDP per capita clearly demonstrates a positive relationship between the two as GDP per capita rises, so does Score. Further, in comparison to the GDP per capita plot, the line of best fit for generosity has unusually high confidence bands, indicating that it is a somewhat unstable characteristic. When undertaking analysis, visualizing data patterns may provide hints of interesting paths to take, but it does not provide any notable prediction capabilities. This is where machine learning approaches show their true worth. In the following part, we will construct many models to identify the most accurate technique to predict a happiness score.

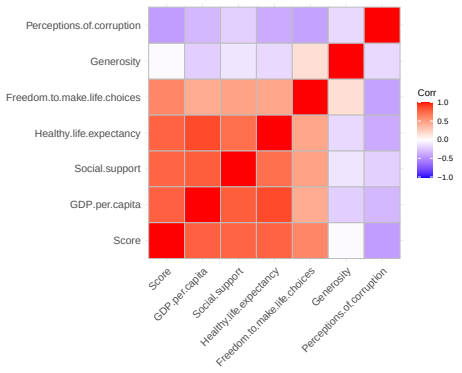

Figure 2: *Correlation among the six parameters*

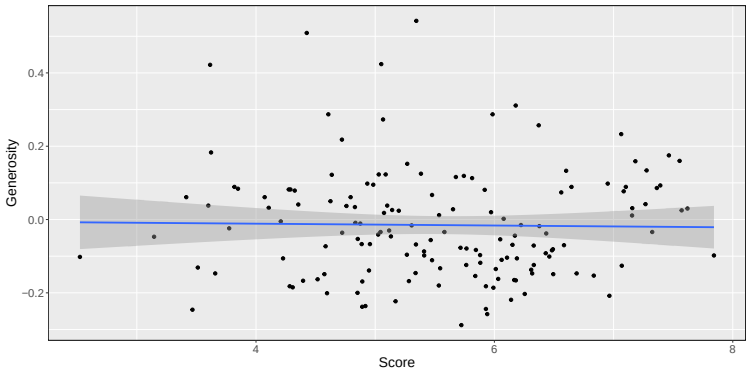

Figure 3: *Score vs. Generosity*

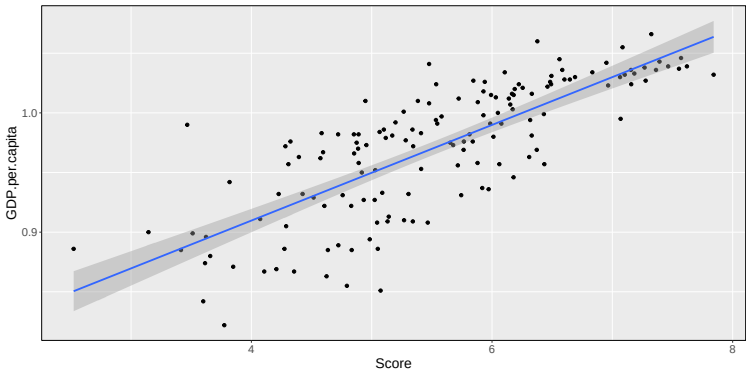

Figure 4: *Score vs. GDP per capita*

2. Methods

2.1 Model 1: The Sum of Factors

The forum was offered on the Kaggle website says that the "perfect" model prediction merely considers the total of all components as a good result.This approach is implemented with one drawback: in previous happiness surveys, a "standard dystopia score" was identified and assigned the value of 1.972. Because each component represents a judgment of how much better the country is than the typical dystopia, this value is also included to our predicted results. It's worth noting that the Root Mean Square Error (RMSE) is used as a success indicator. The results section goes into further detail about this decision.

Overall.rank	Country.or.region	Score	pred_score	RMSE
1	**Finland**	**7.842**	**5.715**	**1.003941**
2	**Denmark**	**7.620**	**5.847**	**1.003941**
3	**Switzerland**	**7.571**	**5.940**	**1.003941**
4	**Iceland**	**7.554**	**6.510**	**1.003941**
5	**Netherlands**	**7.464**	**6.103**	**1.003941**

The RMSE for all estimated scores is the same! Previous Happiness Report data sets contain a "dystopian residual" that adds to the happiness score in addition to the dystopian standard score. It is judged unsuitable to include the residual into the model since it is not presented or specified in this data set. The data set appears to be incomplete without this value.The dystopian residual column is computed despite its absence, and a sample of them is given below. This is derived on the basis of a perfect summation model. The residual is obtained by subtracting all variables and the standard dystopia value from the happiness score. We may infer that the sum of factors model, as depicted on the discussion boards, is not entirely correct; it has an RMSE of 1.003941. We noticed that RSME rose in the year 2021 due to the low value of *Healthy life expectancy* in the year 2021. Also the dystopian residual of 1.972 - which is computed from previous data-set i.e. 2020 data-set- contributes to the hike of RMSE value. A sharp glance at the 2020 data-set and 2021 data-set will lead us to infer that COVID-19 affects the *Healthy life expectancy* in all communities all over the world.

Overall.rank	Country.or.region	Score	pred_score	RMSE	residual
1	**Finland**	**7.842**	**5.715**	**1.003941**	**2.127**
2	**Denmark**	**7.620**	**5.847**	**1.003941**	**1.773**
3	**Switzerland**	**7.571**	**5.940**	**1.003941**	**1.631**
4	**Iceland**	**7.554**	**6.510**	**1.003941**	**1.044**
5	**Netherlands**	**7.464**	**6.103**	**1.003941**	**1.361**

2.2 Model 2: The GLM Model (2021)

The data must be partitioned into a training and test set before our first linear regression model can be implemented. When using machine learning methods that require a check on the quality of fit, this step is typical. It lowers the chances of our training data being overfitted at the expense of our prediction model. The equation `sum(data$[factors])` was not finished for our sum of factors model because a model did not need to be trained. There are a little less 150 nation observations in the Happiness Report, as well as six parameters that we will use to condition our model, *GDP per capita, Social support, Healthy life expectancy, Freedom to make life choices, Generosity, and Perceptions of corruption.* The model may have a propensity to overfit to the training data by overweighting irrelevant variables due to the small number of observations compared to the number of components. Working with little amounts of data makes this practically unavoidable. In the world of regression, you can *always* find a model that fits your training data

well, but it's usually useless for prediction. With this in mind, an optimum training-test data split ratio is first calculated, such as 70 training:30 test, 80 training:20 test, and so on.

Following this section, we'll look into the partitioning model that was tested. When plotting `RMSE` vs `p`, we see a tiny pattern: our RMSE drops as the amount of the training data increases. On the surface, this seems reasonable. As previously demonstrated, the data is highly correlated, and the model is being given additional training data in order to generate better predictions in the test set. 0.4606218 has the lowest RMSE, with a ratio of 0.82:0.18.

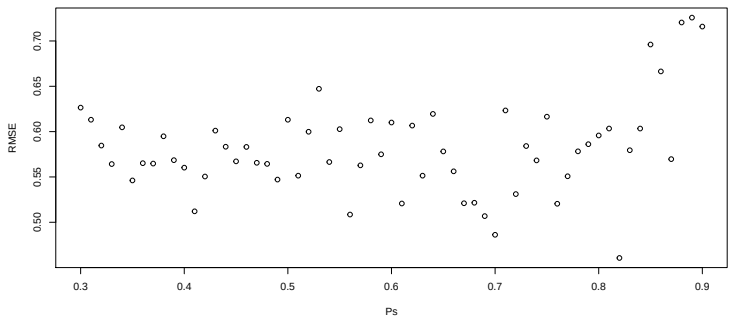

Figure 5: *RMSE vs. Ps.*

Though utilizing just 0.18 percent of our data to test is beneficial for attaining a low RMSE, it does not leave much in terms of prediction. RMSE appears to grow more irregular beyond this number, thus an arbitrary value of 0.70 is used. In the *methods* section, future models will also utilize 0.70. When the existing data is augmented with additional data, this ratio is maintained.

A generalized linear model is fitted using the `caret` package with our data divided 0.7 : 0.3. All six criteria are used to forecast the `Score`. These are the first five expected score.

	Overall.rank	Country.or.region	Score	pred_score
1	1	Finland	7.842	7.019225
3	3	Switzerland	7.571	7.035741
5	5	Netherlands	7.464	6.948880
9	9	New Zealand	7.277	6.993448
12	12	Israel	7.157	6.384657

The lowest five observations, as well as the actual score, are displayed below.

	Overall.rank	Country.or.region	Score	pred_score
135	135	Madagascar	4.208	3.909076
136	136	Togo	4.107	3.658418
138	138	Sierra Leone	3.849	3.946143
145	145	Lesotho	3.512	4.366559
148	148	Zimbabwe	3.145	4.438177

A line of best fit in blue and a reference line in red at `y = x` are plotted in the results data frame below. Because each predicted score would be identical to the score, the line of best fit would follow the reference line if the model worked correctly.

However, it is worth noting that RMSE was used in place of other success metrics. The root mean square error (RMSE) indicates how near (or how distant) your predicted values are to the real data you are attempting to model. The goal of using a success measure for this model, as well as others in the techniques section, is to understand the accuracy and precision of the model's predictions. As a result, RMSE is preferred above other success metrics. 0.4831122 is the RMSE of this model. To be thorough, the fitted model's coefficients are provided below.

```
##                   (Intercept)                GDP.per.capita
##                    -5.1051683                     5.7385780
##                Social.support      Healthy.life.expectancy
##                     2.6307721                     2.3796669
## Freedom.to.make.life.choices                    Generosity
##                     2.2107437                     0.2501214
##    Perceptions.of.corruption
##                    -0.5076351
```

\ The final model equation is given by the equation below. These coefficients and the following notation provide the following results: predicted score = $\hat{y}$, GDP per capita score = x_{GDP}, social support score = x_{SS}, life expectation = x_{HEA}, freedom score = x_{FRE}, generosity score = x_{GEN}, and truth score = x_{TRU}.

$$\hat{y} = (-5.105) + (5.739)x_{GDP} + (2.631)x_{SS} + (2.38)x_{HEA} + (2.211)x_{FRE} + (0.25)x_{GEN} + (-0.508)x_{TRU} \quad (1)$$

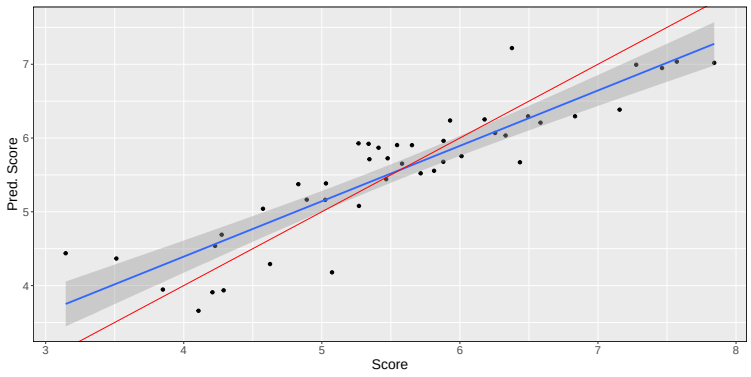

Figure 6: *Actual Scores vs. Predicted Scores for the GLM Model (2021)*

2.3 Model 3: The 2021-GLM without `Generosity` Model

The generosity component can be removed from the model in an effort to enhance it, since early studies indicated that it was the least associated factor. This model also uses the previously partitioned data (`p = 0.70`).

0.4831122 is the RMSE for this model. The model coefficients are presented below, along with the model equation, in the same manner as previously.

```
##                   (Intercept)                GDP.per.capita
##                    -4.9068750                     5.5197851
##                Social.support      Healthy.life.expectancy
##                     2.6507472                     2.2985661
## Freedom.to.make.life.choices    Perceptions.of.corruption
##                     2.3099785                    -0.5484228
```

$$\hat{y} = (-4.907) + (5.52)x_{GDP} + (2.651)x_{SS} + (2.299)x_{HEA} + (2.31)x_{FRE} + (-0.548)x_{TRU} \quad (2)$$

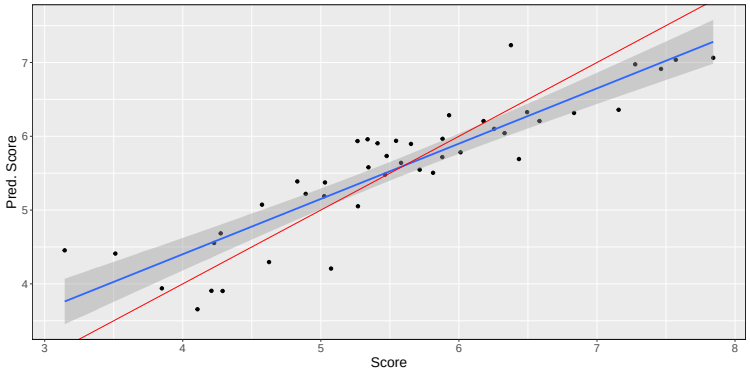

Figure 7: *Actual Scores vs. Predicted Scores for the GLM Model without the Generosity parameter for year 2021*

2.4 Model 4: The GLM Model (2020/2021)

Data sets would be infinitely huge in an ideal situation, and models could be adapted with as much precision as your CPU could manage. The following model is based on survey data from 2020 and 2021. While increasing the size of our train and test sets does not provide an endless data set, it does allow for additional training and testing. In this situation, combining data from previous polls i.e.2020 is suitable since these two years have the same grading method, and they both lack the dystopian residuals indicated above. The initial stage is to create fresh train and test sets, after which you'll run a model that takes into account all of the variables.

```
## Rows: 302
## Columns: 7
## $ Score                          <dbl> 7.842, 7.620, 7.571, 7.554, 7.464, 7.392,~
## $ GDP.per.capita                 <dbl> 1.032, 1.039, 1.046, 1.037, 1.039, 1.043,~
## $ Social.support                 <dbl> 0.954, 0.954, 0.942, 0.983, 0.942, 0.954,~
## $ Healthy.life.expectancy        <dbl> 0.72000, 0.72700, 0.74400, 0.73000, 0.724~
## $ Freedom.to.make.life.choices <dbl> 0.949, 0.946, 0.919, 0.955, 0.913, 0.960,~
## $ Generosity                     <dbl> -0.098, 0.030, 0.025, 0.160, 0.175, 0.093~
## $ Perceptions.of.corruption      <dbl> 0.186, 0.179, 0.292, 0.673, 0.338, 0.270,~
```

Before merging, columns containing no data, such as country name and rank, were deleted. When changing the corruption column of the 2020 data set from factor to numeric, coercion introduces NAs. These are replaced with zeros to indicate data that was not found. The above image shows a sample of the 2020/2021 data set.

Here, the model fits well, especially around the median, where the confidence bands are tiny. The RMSE for this model is 0.4984779. The model coefficients, as well as the model equation in the same format as previously, are presented below.

$$\hat{y} = -4.418 + 4.852x_{GDP} + 2.879x_{SS} + 3.769x_{HEA} + 1.296x_{FRE} + 0.581x_{GEN} + -0.78x_{TRU} \qquad (3)$$

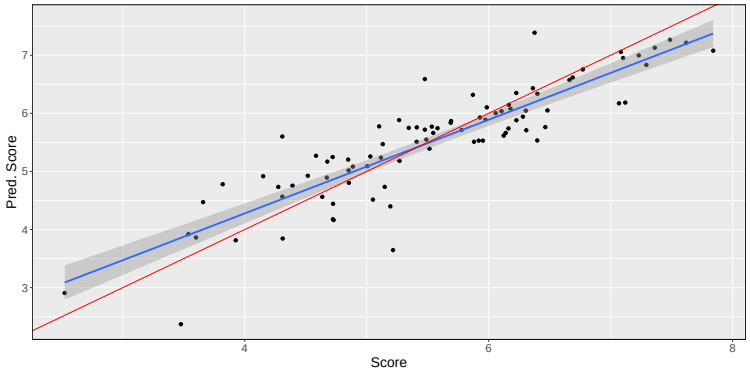

Figure 8: *Predicted Scores vs. Actual Scores for GLM Model (2020/2021)*

2.5 Model 5: The 2020/2021-GLM without `Generosity` Model

The preceding model may be improved by eliminating the relatively uncorrelated generosity component observed in earlier parts. The 2018-2019 data set, `full_data`, completes the set of information.

Especially near the median, when confidence bands are tiny, we observe a model that is better fitting. 0.4902188 is the RMSE of this model. Following is a list of the model coefficients and the model equation in the same notation as previously.

```
##                    (Intercept)                 GDP.per.capita
##                     -4.0137052                      4.3317482
##                 Social.support       Healthy.life.expectancy
##                      2.9481182                      3.7244327
## Freedom.to.make.life.choices     Perceptions.of.corruption
##                      1.4781944                     -0.8856348
```

$$\hat{y} = -4.014 + 4.332x_{GDP} + 2.948x_{SS} + 3.724x_{HEA} + 1.478x_{FRE} + -0.886x_{TRU} \tag{4}$$

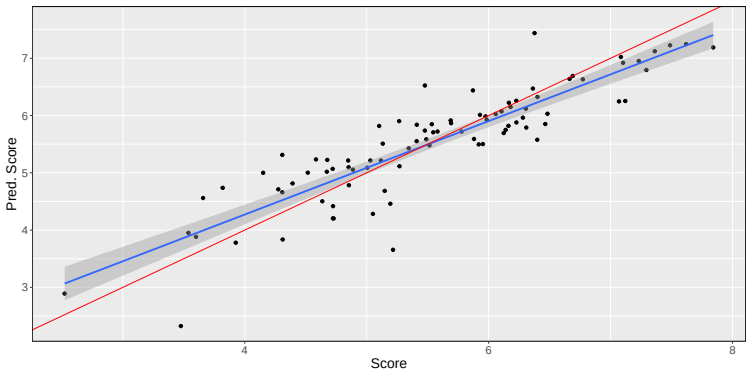

Figure 9: *Predicted Scores vs. Actual Scores for GLM Model without the Generosity parameter for years 2020/2021*

3. Results

The table below summarizes the outcomes of our five models. The GLM Model (2021) is clearly proven to be the best model in terms of RMSE. Despite the fact that the data with dystopian residuals appeared to

be incomplete, the GLM Model (2021) and the conventional dystopian scores yields the most accurate score forecasts.

Method	RMSE
Sum of Factors Model (2021)	1.004
GLM Model (2021)	0.483
GLM Model (2020/2021)	0.498
GLM No Generosity Model (2021)	0.488
GLM No Generosity Model (2020/2021)	0.49

For completeness, it may be useful to check the summation model with the full 2020-2021 data set. This is shown below and reported in the final table.

The best model, as can be noticed, is the GLM Model (2021).

Method	RMSE
Sum of Factors Model (2021)	1.004
Sum of Factors Model (2020/2021)	0.976
GLM Model (2021)	0.483
GLM Model (2020/2021)	0.498
GLM No Generosity Model (2021)	0.488
GLM No Generosity Model (2020/2021)	0.49

4. Conclusion

In computer science and analysis, M.L. algorithms have become indispensable. This study uses regression analysis on a dataset of national happiness to investigate this utilisation. Unexpected results were discovered: a glaring mistake in the data had no influence on the simplest model, increasing the size of the training data set had no effect on the error, and lowering the efficacy of uncorrelated variables had no effect on the error.

The amount of data that was trained and tested in this study seems restricted. I believe that the summation model would beat the GLM models if more consistent data was supplied. The missing residuals created a gap in the data that may have revealed further information about the summation model's validity. The discussion boards were useful in this regard since they provided insight into the data's purpose. Calculating and utilising dystopian residuals for data sets when they were lost, integrating prior years reports, and comparing more adaptable models on big data sets are all things that might be done with further effort on a project like this.

Appendix A:

A.1 Project Overview: Choose Your Own!

For this project, you will be applying machine learning techniques that go beyond standard linear regression. You will have the opportunity to use a publicly available dataset to solve the problem of your choice. You are strongly discouraged from using well-known datasets, particularly ones that have been used as examples in previous courses or are similar to them (such as the iris, titanic, mnist, or movielens datasets, among others) - this is your opportunity to branch out and explore some new data! The UCI Machine Learning Repository and Kaggle are good places to seek out a dataset. Kaggle also maintains a curated list of datasets that are cleaned and ready for machine learning analyses. Your dataset must be automatically downloaded in your code or included with your submission. You may not submit the same project for both the MovieLens and Choose Your Own project submissions.

The ability to clearly communicate the process and insights gained from an analysis is an important skill for data scientists. You will submit a report that documents your analysis and presents your findings, with supporting statistics and figures. The report must be written in English and uploaded as both a PDF document and an Rmd file. Although the exact format is up to you, the report should include the following at a minimum:

an introduction/overview/executive summary section that describes the dataset and variables, and summarizes the goal of the project and key steps that were performed; a methods/analysis section that explains the process and techniques used, including data cleaning, data exploration and visualization, insights gained, and your modeling approaches (you must use at least two different models or algorithms); a results section that presents the modeling results and discusses the model performance; and a conclusion section that gives a brief summary of the report, its potential impact, its limitations, and future work. Your project submission will be graded both by your peers and by a staff member. The peer grading will give you an opportunity to check out the projects done by other learners. You are encouraged to give your peers thoughtful, specific feedback on their projects (i.e., more than just "good job" or "not enough detail").

A.2 Choose Your Own Instructions

The submission for the choose-your-own project will be three files: a report in the form of both a PDF document and Rmd file and the R script that performs your machine learning task. You must also provide access to your dataset, either through automatic download in your script or inclusion in a GitHub repository. (Remember, you are strongly discouraged from using well-known datasets, particularly ones that have been used as examples in previous courses or are similar to them. Also remember that you may not submit the same project for both the MovieLens and Choose Your Own project submissions.) We recommend submitting a link to a GitHub repository with these three files and your dataset. Your grade for the project will be based on your report and your script.

A.3 Report and Script

Your report and script will be graded by your peers, based on a rubric defined by the course staff, as well as by the course staff. The staff grade will be your final grade for the project. Note that due to the volume of submissions and the number of graders, it can take up to four weeks to receive your staff grade, although we strive for a faster turnaround time! To receive your grade, you must review and grade the reports of five of your fellow learners after submitting your own. This will give you the chance to learn from your peers. You are encouraged to give your peers thoughful and specific feedback on their projects.

Please pay attention to the due dates listed! The project submission is due before the end of the course to allow time for peer grading. Also note that you must grade the reports of your peers by the course close date in order to receive your grade.

A.4 Honor Code

You are welcome to discuss your project with others, but all submitted work must be your own. Your participation in this course is governed by the terms of the edX Honor Code. If your report is found to violate the terms of the honor code (for example, if you copy a project from another learner), you will receive a zero on the project, may be unenrolled from the course, and will not be eligible to receive a certificate.

A.5 Project Due Date

Submissions for the project are due one week before course close, on July 28, 2021, at 23:59 UTC. This allows time for peer grading to occur! Peer grades are due at course close, on August 4, 2021, at 23:59 UTC.

Appendix B:

B.1 MovieLens Project Submission

B.2.1 Your Response due Jul 29, 2021 02:59 EEST (in 2 weeks, 3 days)

Enter your response to the prompt. You can save your progress and return to complete your response at any time before the due date (Thursday, Jul 29, 2021 02:59 EEST). After you submit your response, you cannot edit it.

B.2.2 The prompt for this section Your submission for this project is three files:
1.Your report in Rmd format
2.Your report in PDF format (knit from your Rmd file)
3.A script in R format that generates your predicted movie ratings and RMSE score (should contain all code and comments for your project)

You may upload the three files directly to the edX platform or submit a GitHub link in the text response box below.

To upload and submit your files press the "Choose Files" button, select three files at once (using the control key on a Windows machine or command key on a Mac) and press "Choose," type a description for each (PDF, Rmd, R), and then press the "Upload files" button. If uploading files, we recommend also providing a link to a GitHub repository containing the three files above in case there is a problem with the upload process.

Note that when downloading files for peer assessments, R and Rmd files will be downloaded as txt files by default.

B.2.3 Grading Rubric
Note: after you submit your project, please check immediately after submitting to make sure that all files were correctly uploaded. Occasionally, there are file upload failures, and it's easiest to fix if these are caught early. ##### B.2.3.a Files (5 points possible)
The appropriate files are submitted in the correct formats: a report in both PDF and Rmd format and an R script in R format. The PDF version of the report should be knit from your Rmd file, and the R script should contain all of the code and comments for your project.

- 0 points: No files provided.
- 3 points: At least one file is missing and/or not in the correct format.
- 5 points: All 3 files were submitted in the requested formats.

B.2.3.b Report (25 points possible)
The report documents the analysis and presents the findings, along with supporting statistics and figures. In order to demonstrate your understanding of course material, please provide thorough explanation or justification for various steps of your project, such as why a specific train/test split (e.g. 50/50 vs 90/10) or algorithm was used. The report must be written in English and uploaded. The report should assume that the reader is not familiar with the project or the dataset. The report must include at least the following sections: **1. An introduction/overview/executive summary section that describes the dataset and variables, and summarizes the goal of the project and key steps that were performed.**
2. A methods/analysis section that explains the process and techniques used, including data cleaning, data exploration and visualization, any insights gained, and your modeling approach.

At least two different models or algorithms must be used, with at least one being more advanced than linear or logistic regression for prediction problems.
3. A results section that presents the modeling results and discusses the model performance.
4. A conclusion section that gives a brief summary of the report, its potential impact, its limitations, and future work.

- 0 points: The report is either not uploaded or contains very minimal information OR the report appears to violate the terms of the edX Honor Code.

- 5 points: One or more required sections of the report are missing.
- 10 points: The report includes all required sections, but the report is significantly difficult to follow or missing significant supporting detail in multiple sections.
- 15 points: The report includes all required sections, but the report is difficult to follow or missing supporting detail in one section (or has minor flaws in multiple sections).

- 20 points: The report includes all required sections and is easy to follow, but with minor flaws in one section.

- 25 points: The report includes all required sections, is easy to follow with good supporting detail throughout, and is insightful and innovative.

B.2.3.c Code (20 points)

The code in the R script should run without errors and should be well-commented and easy to follow. It should also use relative file paths and automatically install missing packages. The dataset you use should either be automatically downloaded by your code or provided in your GitHub repo along with the rest of your files (Rmd, PDF, R). If your dataset is provided as a zip file in GitHub, your code should automatically unzip and load it. The R script should contain all of the code and comments for your project. + 0 points: Code does not run and produces many errors OR code appears to violate the terms of the edX Honor Code.
+ 5 points: Code runs but does not produce output consistent with what is presented in the report OR there is overtraining (the test set is used for training steps).
+ 10 points: Code runs but is difficult to follow and/or may not produce output entirely consistent with what is presented in the report. + 15 points: Code runs, can be followed, is at least mostly consistent with the report, but is lacking (sufficient) comments and explanation OR uses absolute paths instead of relative paths OR does not automatically install missing packages External link OR does not provide easy access to the dataset (either via automatic download or inclusion in a GitHub repository)
+ 20 points: Code runs easily, is easy to follow, is consistent with the report, and is well-commented. All file paths are relative and missing packages are automatically installed External link with if(!require) statements.

Have a question about the choose your own project? Want to bounce some ideas for an analysis to do or a dataset to pick off someone else? Need some feedback on the best approach to take or some troubleshooting for a snippet of your code? You can ask your questions here!

You are encouraged to discuss general approaches to your project. It is okay to post small snippets of code if you're having trouble getting a particular piece of code to run. However, you may not post your entire R script for the project.