

Introduction

I have chosen to look at the **World Happiness Report 2015**. This is a landmark survey report which was released on March 20th, 2015, also known as the International Day of Happiness, constructed by the United Nations. This report had been done previously in 2012 and 2013 as well prior to the data I am using. The report continues to be created by the UN and has been redone on a yearly basis after the 2015 report. The report has gained global recognition over time and now is used by many world governments, organizations and civil societies for policy making decisions.

I have found this report and data inside it to be extremely interesting. This is because one of the main goals in life is for people to live happily. Happiness has been scientifically proven to have huge benefits for your health. Happiness promotes a healthy lifestyle as it combats stress, boosts your immune system, reduces pain, and even increases your life expectancy.

My interest is to look at and compare 3 main questions.

- 1) Which region of the world is the happiest?
- 2) Do countries with a better GDP per capita have the happiest people?
- 3) How dose freedom and trust in government affect a countries happiness?

I shall use SQL queries on the dataset to answer my questions. I will do this using both SQLite and MS SQL.

Dataset:

This dataset named **World Happiness Report 2015** was published by the United Nations on March 20th, 2015. It was constructed using surveys sent out to 158 countries. There was no mention of the sample size of people surveyed however as this data is from a credible source (UN) I trust the data is accurate.

Schema and limitations:

The dataset is comprised of 12 fields total. One for country, region, happiness rank, happiness score, standard error, economy, family, life expectancy, freedom, government trust, generosity, and dystopia residual. The last 8 fields mentioned were the main indicators used to determine happiness. The main field in the dataset is the happiness field, it is ranked from 0 to 10. In this dataset there are 158 rows one for each country surveyed. A limitation in the data is that there is a standard error for each country. In my examination of the data, I will not be taking the standard error into consideration as processing the data for my questions using error this is out of my level of understanding, therefore I shall only use the averaged results used for the rest of the dataset.

Rohan Mankame

Data Sci 220

June 24th, 2022

Methods:

How I imported the csv file to SQLite and MS MySQL.

SQLite:

- 1) Created an empty database in SQLite.
- 2) Create new table and entered each field from the dataset and set a corresponding datatype to field.
- 3) Choose CSV file from import option
- 4) Imported dataset from CSV file to fill the table I created.

MS MySQL:

- 1) Created an empty database in MySQL.
- 2) Use "Table Data Import Wizard" tool and choose CSV file
- 3) Imported dataset from CSV file to database.

Table Data Import Wizard

(No issues importing CSV file)

How table looked:

SQLite:

Table: 2015									
Filter in any column									
	Country	Region	HappinessRank	HappinessScore	StandardError	Economy_GDP_per_Capita	Family	LifeExpectancy	Fre
	Filter	Filter	Filter	Filter	Filter	Filter	Filter	Filter	Filter
1							Family	LifeExpectancy	Fre
2	Switzerland	Western Europe	1	7.587	0.03411	1.39651	1.34951	0.94143	0.
3	Iceland	Western Europe	2	7.561	0.04884	1.30232	1.40223	0.94784	0.
4	Denmark	Western Europe	3	7.527	0.03328	1.32548	1.36058	0.87464	0.
5	Norway	Western Europe	4	7.522	0.0388	1.459	1.33095	0.88521	0.
6	Canada	North America	5	7.427	0.03553	1.32629	1.32261	0.90563	0.
7	Finland	Western Europe	6	7.406	0.0314	1.29025	1.31826	0.88911	0.
8	Netherlands	Western Europe	7	7.378	0.02799	1.32944	1.28017	0.89284	0.
9	Sweden	Western Europe	8	7.364	0.03157	1.33171	1.28907	0.91087	C

MySQL:

	Country	Region	HappinessRank	HappinessScore	StandardError	Economy_GDP_per_Capita	Family	LifeExpectancy	Freedom	Trust_Gov	Generosity
▶	Switzerland	Western Europe	1	7.587	0.03411	1.39651	1.34951	0.94143	0.66557	0.41978	0.29678
	Iceland	Western Europe	2	7.561	0.04884	1.30232	1.40223	0.94784	0.62877	0.14145	0.4363
	Denmark	Western Europe	3	7.527	0.03328	1.32548	1.36058	0.87464	0.64938	0.48357	0.34139
	Norway	Western Europe	4	7.522	0.0388	1.459	1.33095	0.88521	0.66973	0.36503	0.34699
	Canada	North America	5	7.427	0.03553	1.32629	1.32261	0.90563	0.63297	0.32957	0.45811
	Finland	Western Europe	6	7.406	0.0314	1.29025	1.31826	0.88911	0.64169	0.41372	0.23351
	Netherlands	Western Europe	7	7.378	0.02799	1.32944	1.28017	0.89284	0.61576	0.31814	0.4761
	Sweden	Western Europe	8	7.364	0.03157	1.33171	1.28907	0.91087	0.6598	0.43844	0.36262
	New Zealand	Australia and New Zealand	9	7.286	0.03371	1.25018	1.31967	0.90837	0.63938	0.42922	0.47501
	Australia	Australia and New Zealand	10	7.284	0.04083	1.33358	1.30923	0.93156	0.65124	0.35637	0.43562

Rohan Mankame

Data Sci 220

June 24th, 2022

How I imported the csv file to MongoDB.

MongoDB:

- 1) Connected to a open cluster on MongoDB Compass on local machine.
- 2) Create a new database and give it a name and collection name.
- 3) Open database on MongoDB Compass and click import, choose the CSV file with the dataset. After choosing file to import make sure to edit all field datatypes from string to what they are meant to be. Should look like the image shown to the right ----->
- 4) Click import and done once imported

(No issues importing CSV file)

What kind of database is it?


































MongoDB uses Non-Rational Databases also usually referred to as NoSQL. MongoDB can be used as a general data store.

How is it different from a relational database tool?

Non-Rational databases are able to be schema-agnostic, meaning they can be unstructured or even semi-structured and still able to be stored and processed.

How table looked:

MongoDB (NoSQL):

_id	ObjectId	Country String	Region String	HappinessRank Int32	HappinessScore Double	StandardError Doub	
1	ObjectId('62b62494c51da73d5ff...	"Switzerland"	"Western Europe"	1	7.587	0.03411	  
2	ObjectId('62b62494c51da73d5ff...	"Iceland"	"Western Europe"	2	7.561	0.04884	  
3	ObjectId('62b62494c51da73d5ff...	"Denmark"	"Western Europe"	3	7.527	0.03328	  
4	ObjectId('62b62494c51da73d5ff...	"Norway"	"Western Europe"	4	7.522	0.0388	  
5	ObjectId('62b62494c51da73d5ff...	"Canada"	"North America"	5	7.427	0.03553	  
6	ObjectId('62b62494c51da73d5ff...	"Finland"	"Western Europe"	6	7.406	0.0314	  
7	ObjectId('62b62494c51da73d5ff...	"Netherlands"	"Western Europe"	7	7.378	0.02799	  
8	ObjectId('62b62494c51da73d5ff...	"Sweden"	"Western Europe"	8	7.364	0.03157	  
9	ObjectId('62b62494c51da73d5ff...	"New Zealand"	"Australia and New Zealand"	9	7.286	0.03371	  
10	ObjectId('62b62494c51da73d5ff...	"Australia"	"Australia and New Zealand"	10	7.284	0.04083	  
11	ObjectId('62b62494c51da73d5ff...	"Israel"	"Middle East and Northern Afr"	11	7.272	0.0347	  

Using the Command line my data looked like this:

```
DBS1 0.000GB
admin 0.000GB
config 0.000GB
local 0.000GB
> use DBS1
switched to db DBS1
> show collections
Happiness
> db.Happiness.find()
{ "_id" : ObjectId("62b62494c51da73d5ff7c6bd"), "Country" : "Switzerland", "Region" : "Western Europe", "HappinessRank" : 1, "HappinessScore" : 7.587, "StandardError" : 0.03411, "Family" : 1.34951, "LifeExpectancy" : 0.94143, "Freedom" : 0.66557, "Trust_Gov" : 0.66557 },
{ "_id" : ObjectId("62b62494c51da73d5ff7c6be"), "Country" : "Iceland", "Region" : "Western Europe", "HappinessRank" : 2, "HappinessScore" : 7.561, "StandardError" : 0.04884, "Family" : 1.40223, "LifeExpectancy" : 0.94784, "Freedom" : 0.62877, "Trust_Gov" : 0.62877 },
{ "_id" : ObjectId("62b62494c51da73d5ff7c6bf"), "Country" : "Denmark", "Region" : "Western Europe", "HappinessRank" : 3, "HappinessScore" : 7.527, "StandardError" : 0.03328, "Family" : 1.36058, "LifeExpectancy" : 0.87464, "Freedom" : 0.64938, "Trust_Gov" : 0.64938 },
{ "_id" : ObjectId("62b62494c51da73d5ff7c6c0"), "Country" : "Norway", "Region" : "Western Europe", "HappinessRank" : 4, "HappinessScore" : 7.522, "StandardError" : 0.0388, "Family" : 1.33095, "LifeExpectancy" : 0.88521, "Freedom" : 0.66973, "Trust_Gov" : 0.66973 },
{ "_id" : ObjectId("62b62494c51da73d5ff7c6c1"), "Country" : "Canada", "Region" : "North America", "HappinessRank" : 5, "HappinessScore" : 7.427, "StandardError" : 0.03553, "Family" : 1.32261, "LifeExpectancy" : 0.90563, "Freedom" : 0.63297, "Trust_Gov" : 0.63297 },
{ "_id" : ObjectId("62b62494c51da73d5ff7c6c2"), "Country" : "Finland", "Region" : "Western Europe", "HappinessRank" : 6, "HappinessScore" : 7.406, "StandardError" : 0.0314, "Family" : 1.31826, "LifeExpectancy" : 0.88911, "Freedom" : 0.64169, "Trust_Gov" : 0.64169 },
{ "_id" : ObjectId("62b62494c51da73d5ff7c6c3"), "Country" : "Netherlands", "Region" : "Western Europe", "HappinessRank" : 7, "HappinessScore" : 7.378, "StandardError" : 0.02799, "Family" : 1.31826, "LifeExpectancy" : 0.88911, "Freedom" : 0.64169, "Trust_Gov" : 0.64169 },
{ "_id" : ObjectId("62b62494c51da73d5ff7c6c4"), "Country" : "Sweden", "Region" : "Western Europe", "HappinessRank" : 8, "HappinessScore" : 7.364, "StandardError" : 0.03157, "Family" : 1.31826, "LifeExpectancy" : 0.88911, "Freedom" : 0.64169, "Trust_Gov" : 0.64169 },
{ "_id" : ObjectId("62b62494c51da73d5ff7c6c5"), "Country" : "New Zealand", "Region" : "Australia and New Zealand", "HappinessRank" : 9, "HappinessScore" : 7.286, "StandardError" : 0.03371, "Family" : 1.31826, "LifeExpectancy" : 0.88911, "Freedom" : 0.64169, "Trust_Gov" : 0.64169 },
{ "_id" : ObjectId("62b62494c51da73d5ff7c6c6"), "Country" : "Australia", "Region" : "Australia and New Zealand", "HappinessRank" : 10, "HappinessScore" : 7.284, "StandardError" : 0.04083, "Family" : 1.31826, "LifeExpectancy" : 0.88911, "Freedom" : 0.64169, "Trust_Gov" : 0.64169 }
```

SQL Queries used for each question;

- 1) Which region of the world is the happiest?

```
SELECT Region, avg(HappinessScore)
FROM '2015'
GROUP BY Region
order by avg(HappinessScore) DESC
```

- 2) Do countries with a better GDP per capita have the happiest people?

```
SELECT Region, avg(Economy_GDP_per_Capita)
FROM '2015'
GROUP BY Region
order by avg(HappinessScore) DESC
```

- 3) How dose freedom and trust in government affect a countries happiness?

```
SELECT Country, Trust_Gov + Freedom , HappinessScore
FROM `2015`
GROUP BY Country
order by avg(HappinessScore) DESC
```

Rohan Mankame

Data Sci 220

June 24th, 2022

MongoDB(NoSQL) Queries used for each question;

1) Which region of the world is the happiest?

```
db.Happiness.find([{"HappinessScore" : {"$gt": 6.5}}]).pretty()
```

I chose to look for data where the Happiness Score is over 6.5 to solve this in NoSQL, I used .pretty() to make the data more readable

2) Do countries with a better GDP per capita have the happiest people?

```
db.Happiness.find( $and : [{HappinessScore : {"$gt": 6.5}, Economy_GDP_per_Capita : {"$gt : 1} } ]).pretty()
```

Here I made looked for only data with a good economy and high happiness scores.

3) How dose freedom and trust in government affect a countries happiness?

```
db.Happiness.aggregate(($project: (_id:0, HappinessScore:1, Trust_Gov:1, Freedom:1, TrustNFreedom: ($add: ["$Trust_Gov", "$Freedom"]))))
```

Unlucky my solution for this question was not working it kept coming up with errors this can be seen in my results section as well

```
> db.Happiness.aggregate(($project: (_id:0, HappinessScore:1, Trust_Gov:1, Freedom:1, TrustNFreedom: ($add: ["$Trust_Gov", "$Freedom"]))))
uncaught exception: SyntaxError: missing ) in parenthetical :
@(shell):1:32
>
```

Results:

1) Which region of the world is the happiest?

```

1
2 SELECT Region, avg(HappinessScore)
3 FROM '2015'
4 GROUP BY Region
5 order by avg(HappinessScore) DESC
6
7
8
9

```

	Region	avg(HappinessScore)
1	Australia and New Zealand	7.285
2	North America	7.273
3	Western Europe	6.68961904761905
4	Latin America and Caribbean	6.14468181818182
5	Eastern Asia	5.62616666666667
6	Middle East and Northern Africa	5.4069
7	Central and Eastern Europe	5.33293103448276
8	Southeastern Asia	5.31744444444445
9	Southern Asia	4.58085714285714
10	Sub-Saharan Africa	4.2028

```

1 • SELECT Region, avg(HappinessScore)
2 FROM new_schema22.`2015`
3 GROUP BY Region

```

```

> db.Happiness.find({"HappinessScore" : {"$gt":6.5}}).pretty()
{
  "_id" : ObjectId("62b62494c51da73d5ff7c6bd"),
  "Country" : "Switzerland",
  "Region" : "Western Europe",
  "HappinessRank" : 1,
  "HappinessScore" : 7.587,
  "StandardError" : 0.03411,
  "Economy_GDP_per_Capita" : 1.39651,
  "Family" : 1.34951,
  "LifeExpectancy" : 0.94143,
  "Freedom" : 0.66557,
  "Trust_Gov" : 0.41978,
  "Generosity" : 0.29678,
  "DystopiaResidual" : 2.51738
}
{
  "_id" : ObjectId("62b62494c51da73d5ff7c6be"),
  "Country" : "Iceland",
  "Region" : "Western Europe",
  "HappinessRank" : 2,
  "HappinessScore" : 7.561,
  "StandardError" : 0.04884,
  "Economy_GDP_per_Capita" : 1.30232,
  "Family" : 1.40223,
  "LifeExpectancy" : 0.94784,
  "Freedom" : 0.62877,
  "Trust_Gov" : 0.14145,
  "Generosity" : 0.4363,
  "DystopiaResidual" : 2.70201
}
{
  "_id" : ObjectId("62b62494c51da73d5ff7c6bf"),
  "Country" : "Denmark",
  "Region" : "Western Europe",

```

From my results I found that Australia and New Zealand were on average the happiest region to live in, with its happiness score being 7.285. North America was a close second with its happiness score averaging 7.273. Western Europe came third. Further the least happy places in the world were Sub-Saharan Africa, South Asia, and Southeast Asia. From the NoSQL data we can see that top 2 countries ranked according to happiness were both in Western Europe. Switzerland having a happiness score of 7.587 and Ireland having a score of 7.56.

These results show western nations are happier, and southern Asian countries being the least happy. This may be due to the western vs eastern lifestyles people have. However, I thought about the economy in all regions. Does more money make us happier? Therefore, I chose my next question to build on this was;

2) Do countries with a better GDP per capita have the happiest people?

```
1  
2 SELECT Region, avg(Economy_GDP_per_Capita)  
3 FROM '2015'  
4 GROUP BY Region  
5 order by avg(HappinessScore) DESC  
6  
7  
8  
9
```

	Region	avg(Economy_GDP_per_Capita)
1	Australia and New Zealand	1.29188
2	North America	1.3604
3	Western Europe	1.29859571428571
4	Latin America and Caribbean	0.876815454545455
5	Eastern Asia	1.15178
6	Middle East and Northern Africa	1.0669735
7	Central and Eastern Europe	0.942438275862069
8	Southeastern Asia	0.789054444444444
9	Southern Asia	0.560485714285714
10	Sub-Saharan Africa	0.380473

```
1  
2 • SELECT Region, avg(Economy_GDP_per_Capita)  
3 FROM new_schema22.`2015`  
4 GROUP BY Region  
5 order by avg(HappinessScore) DESC  
6
```

```
> db.Happiness.find( {$and :[ {HappinessScore :{"$gt":6.5}, Economy_GDP_per_Capita: {$gt:1}} ]} ).pretty()  
{  
  "_id" : ObjectId("62b62494c51da73d5ff7c6bd"),  
  "Country" : "Switzerland",  
  "Region" : "Western Europe",  
  "HappinessRank" : 1,  
  "HappinessScore" : 7.587,  
  "StandardError" : 0.03411,  
  "Economy_GDP_per_Capita" : 1.39651,  
  "Family" : 1.34951,  
  "LifeExpectancy" : 0.94143,  
  "Freedom" : 0.66557,  
  "Trust_Gov" : 0.41978,  
  "Generosity" : 0.29678,  
  "DystopiaResidual" : 2.51738  
}  
{  
  "_id" : ObjectId("62b62494c51da73d5ff7c6be"),  
  "Country" : "Iceland",  
  "Region" : "Western Europe",  
  "HappinessRank" : 2,  
  "HappinessScore" : 7.561,  
  "StandardError" : 0.04884,  
  "Economy_GDP_per_Capita" : 1.30232,  
  "Family" : 1.40223,  
  "LifeExpectancy" : 0.94784,  
  "Freedom" : 0.62877,  
  "Trust_Gov" : 0.14145,  
  "Generosity" : 0.4363,  
  "DystopiaResidual" : 2.70201  
}  
{  
  "_id" : ObjectId("62b62494c51da73d5ff7c6bf"),  
  "Country" : "Denmark",  
  "Region" : "Western Europe",  
  "HappinessRank" : 3,  
  "HappinessScore" : 7.527
```

Here I got very similar results with Australia, and New Zealand coming first, followed by North America and Western Europe. Also, the places with the least GDP per capita were the least happy. My noSQL data was ordered by the happiness rank of a country it shows that both Switzerland and Ireland have the happiest people overall looking at their GDP it is in the 1.39 and 1.30 respectively. It shows that a good economy dose in fact have a happier population.

3) How dose freedom and trust in government affect a countries happiness?

```

1
2 SELECT Country, Trust_Gov + Freedom , HappinessScore
3 FROM '2015'
4 GROUP BY Country
5 order by avg(HappinessScore) DESC
6
7
8
9

```

```

1
2 • SELECT Country, Trust_Gov + Freedom , HappinessScore
3 FROM new_schema22.`2015`
4 GROUP BY Country
5 order by avg(HappinessScore) DESC
6
7

```

	Country	Trust_Gov + Freedom	HappinessScore
1	Switzerland	1.08535	7.587
2	Iceland	0.77022	7.561
3	Denmark	1.13295	7.527
4	Norway	1.03476	7.522
5	Canada	0.96254	7.427
6	Finland	1.05541	7.406
7	Netherlands	0.9339	7.378
8	Sweden	1.09824	7.364
9	New Zealand	1.0686	7.286
10	Australia	1.00761	7.284

```

> db.Happiness.aggregate(($project: (_id:0, HappinessScore:1, Trust_Gov:1, Freedom:1, TrustNFreedom: ($add: ["$Trust_Gov", "$Freedom"]))))
uncaught exception: SyntaxError: missing ) in parenthetical :
@(shell):1:32
>

```

These results show all countries with the most trust in government and most freedom all had the happiest. Looking back at my first question we can see the happiest region had the happiness score of 7.285. All top 9 countries with high freedom and trust in government rank above that average happiness score for a region. I was unable to examine the NoSQL data.

After looking at my results I can see why many governments have started using the **World Happiness Report 2015** into consideration when making policy changes.

Discussion:

In MySQL when importing the CSV file, it was easier. This is as when importing the file, the Data table Import Wizard tool is used. This allows the file data to be read and MySQL can most often predict the correct datatype of the fields. However, in SQLite this was not the case, you must first create a table in a database and manually setup fields and their datatype. However, I felt SQLite was much easier if you must create your own table as MySQL requires to program the table using SQL script if you were to create your own. I would use MySQL if importing a database and use SQLite if creating my own database. This was for relational databases.

Comparing relational to the non-relational database I used, It was as easy to import the dataset as it was in the relational database tools. I found that the non-relational database was going to be more suited for larger datasets. This is because of its ability to use unstructured data and large data objects. In the case of this dataset, it is relatively small and therefore more suited for the 2 relational database tools as compared to MongoDB. It became much harder to look through the data in the CMD as compared to the outputs in MySQL and SQLite. I found it was also much more difficult to program for MongoDB as the relational database tools allowed simple SQL querying language. Also, because all this data was structured it was much easier on those tools.

Next Steps:

The dataset may be able to be scaled into a larger dataset by having more questions on the survey given out. Currently all the data is well structured and all either one string or numbers. Here as nearly all countries are already included it will be most likely scale horizontally, by more questions asked. Data such as when people are asked more about their feelings could scale the dataset.

The main elements that allow NoSQL to be more applicable for larger datasets are the fact that it is easily scalable as compared to relational databases, they are flexible allowing the use of unstructured or semi-structured data and the fact that they support multiple data models.

Rohan Mankame

Data Sci 220

June 24th, 2022

Collaboration report:

I had a partner to work on this project with however as our schedules did not work well together, he had already started worked with another member. I was unable to find a new partner therefore I decided to work alone.

I have done this project individually.

All sections were produced by me (Rohan Mankame).

Tasks I performed were,

- Creating questions of interest for dataset
- Importing of dataset into SQLite, MS MySQL, and MongoDB
- Creating and performing queries on dataset
- Analysis of results
- Entire write up and documentation