

Oddawanie Projektu

Paweł Benkowski s16569

Link do aplikacji

<https://projekt-dot-still-primer-271314.ey.r.appspot.com/>

Aplikacja agreguje 2 datasety i zbiera informacje do utworzenia raportu na rok 2020

<https://www.kaggle.com/unsdsn/world-happiness>

<https://www.kaggle.com/fernandol/countries-of-the-world>



Mortality GDP per capita Survey 2020

Partake in World Happiness Survey 2020

Country
Afghanistan

Overall Happiness
The worst

Economical Situation
The worst

Family support
The worst

Health Care
The worst

Civil Liberties
The worst

Trust in Government
The worst

Social Support
The worst

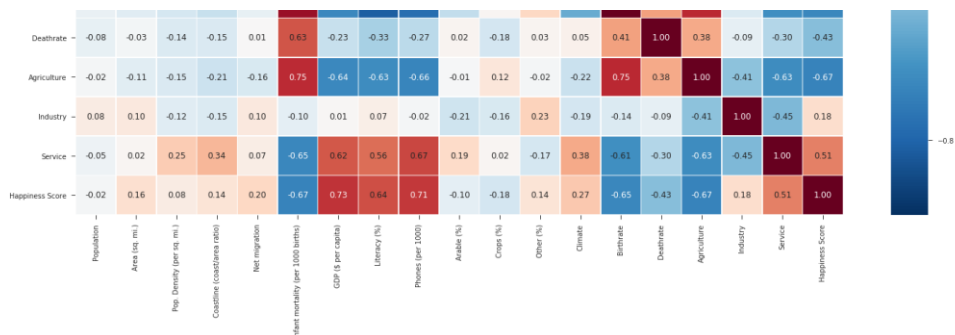
Submit

Figure 1 Formularz do zbierania danych na rok 2020

Są one normalizowane przez organizatorów względem liczby ankiet. Dlatego w obecnej postaci jeszcze nie da się ich wykorzystać.

Analiza została przeprowadzona w datalabie

<https://console.cloud.google.com/compute/instancesDetail/zones/europe-north1-a/instances/countries-info-exploration?project=still-primer-271314&cloudshell=false&supportedpurview=project>



Interpretacja Heatmap'y

Najważniejsza dla tego projektu jest ostatni wiersz pokazujący korelacje Happiness Score / x. Wysokie dodatnie wartości przy GDP, Literacy i Phones oznaczają "jedną rośnię gdy rośnie druga" tzn. Im wyższe GDP kraju tym szczęśliwsi mieszkańcy. Analogicznie wysokie ujemne wartości przy Infant Mortality, Deathrate i Agriculture pokazują "im więcej ludzi umiera i im więcej rolników tym ludzie mniej szczęśliwi".

Ploty zależności

Część analizy z datalab'a

Plik ipynb i wygenerowany html znajdują się w załącznikach zadania Analiza i Eksploracja danych, dołączy je również tutaj



countries_info_exploration.html



countries_info_exploration.ipynb

Przetworzenie danych i umieszczenie w BigQuery

```
def export_to_bq(dataset_name, table_name, data):  
    # Define BigQuery dataset and table  
    dataset = bq.Dataset(dataset_name)  
    table = bq.Table(dataset_name + '.' + table_name)  
    # Create BigQuery dataset  
    if not dataset.exists():  
        dataset.create()  
    # Set schema  
    table_schema = bq.Schema.from_data(data)  
    # Create table  
    table.create(schema = table_schema, overwrite = True)  
    # Insert data  
    table.insert(data)
```

```
1 bucket_object = export_to_gcs('data_2015.csv', mh_15)  
2 %storage write --variable mh_15 --object $bucket_object --content_type text/csv  
3 bucket_object = export_to_gcs('data_2016.csv', mh_16)  
4 %storage write --variable mh_16 --object $bucket_object --content_type text/csv  
5 bucket_object = export_to_gcs('data_2017.csv', mh_17)  
6 %storage write --variable mh_17 --object $bucket_object --content_type text/csv  
7 bucket_object = export_to_gcs('data_2018.csv', mh_18)  
8 %storage write --variable mh_18 --object $bucket_object --content_type text/csv  
9 bucket_object = export_to_gcs('data_2019.csv', mh_19)  
10 %storage write --variable mh_19 --object $bucket_object --content_type text/csv  
11  
12 export_to_bq('country_info', 'data_2015', mh_15)  
13 export_to_bq('country_info', 'data_2016', mh_16)  
14 export_to_bq('country_info', 'data_2017', mh_17)  
15 export_to_bq('country_info', 'data_2018', mh_18)  
16 export_to_bq('country_info', 'data_2019', mh_19)  
17 export_to_bq('country_info', 'data_all', all_years)  
18  
19 survey_mock = pd.DataFrame({"country": "Poland", "overall_happiness": [6], "economy": [  
20 export_to_bq("surveys", 'survey_2020', survey_mock)
```

Rysunek 2 Fragment kodu z analizy odpowiedzialny za export do bq

The screenshot shows the BigQuery web interface. On the left is a sidebar with navigation options like 'Query history', 'Saved queries', 'Job history', 'Transfers', 'Scheduled queries', 'Reservations', 'BI Engine', and 'Resources'. The main area is titled 'Query editor' and contains a SQL query: `SELECT * FROM `still-primer-271314.country_info.data_all` LIMIT 1000`. Below the editor, the 'Query results' section shows that the query is complete (0.2 sec elapsed, 157.5 KB processed). A table of results is displayed with 10 rows and 14 columns. The columns are: Row, agriculture_share, arable_percent, area_sq_mi, birthrate_per_1000, climate_type, coastline_coast_by_area_ratio, country, crops_percent, deathrate_per_1000, economy_gdp_per_capita, family, freedom, and an unlabeled column with values 0, 1.265, 0.972, 0, 0.474, 0.592, 0, 0, 0.26135, and 0.14622.

Row	agriculture_share	arable_percent	area_sq_mi	birthrate_per_1000	climate_type	coastline_coast_by_area_ratio	country	crops_percent	deathrate_per_1000	economy_gdp_per_capita	family	freedom	
1	0.059000000000000004	49.26	2040	15.43	2.0	8.68	Mauritius	2.96	6.86	1.0076100000000001	0.98521	0	
2	0.033	2.67	92300	21.25	1.0	0.03	Jordan	1.83	2.65	0.8220000000000001	1.265	0.4679999999999999	
3	0.13	0.37	163270	18.02	2.0	0.24	Suriname	0.06	7.27	0.99534	0.972	0.5965699999999999	
4	0.258	3.09	47000	33.65	2.0	0.0	Bhutan	0.43	12.7	0.8527	0.9083600000000001	0	
5	0.395	46.15	56785	37.01	2.0	0.1	Togo	2.21	9.83	0.259	0.474		
6	0.018000000000000002	54.02	43094	11.13	3.0	16.97	Denmark	0.19	10.36	1.383	1.5730000000000002	0.5920000000000000	
7	0.4	35.87	2170	36.93	2.0	15.67	Comoros	23.32	8.2	0.27509	0.60323	0	
8	0.25	0.48	1030700	40.99	1.0	0.07	Mauritania	0.01	12.16	0.45407	0.86908	0.24231999999999999	
9	0.65	1.67	637657	45.13	1.0	0.47	Somalia	0.04	16.63	0.0	0.7120000000000001		
10	0.21600000000000003	27.87	803940	29.74	1.0	0.13	Pakistan	0.87	8.23	0.68816	0.26135	0.14622000000000000	

Figure 2 Widok danych w bq

Wizualizacja wniosków

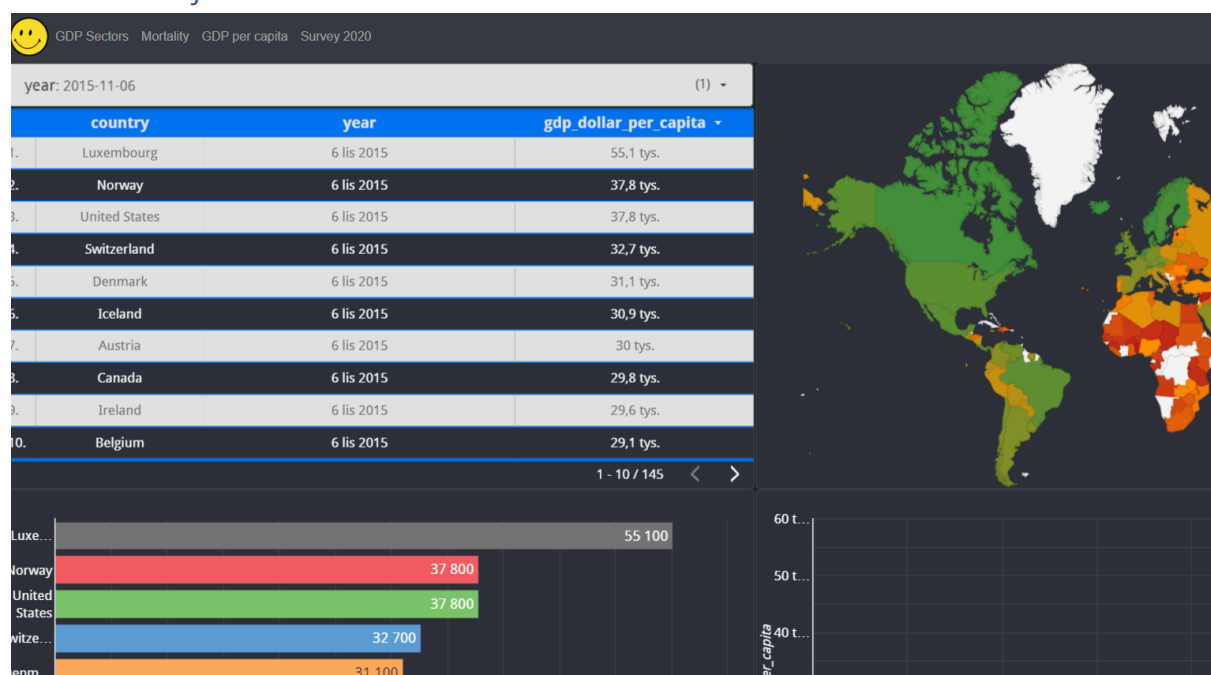


Figure 3 Jeden z 3 raportów zembedowanych na stronie