Moduł 11
Przypadek użycia 2
Dane Covid a rekomendacje turystyczne

iBigWorld:
Innovations for Big Data in a Real World

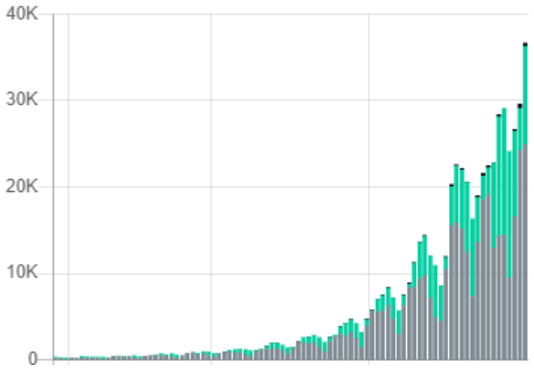
Zespół UBB

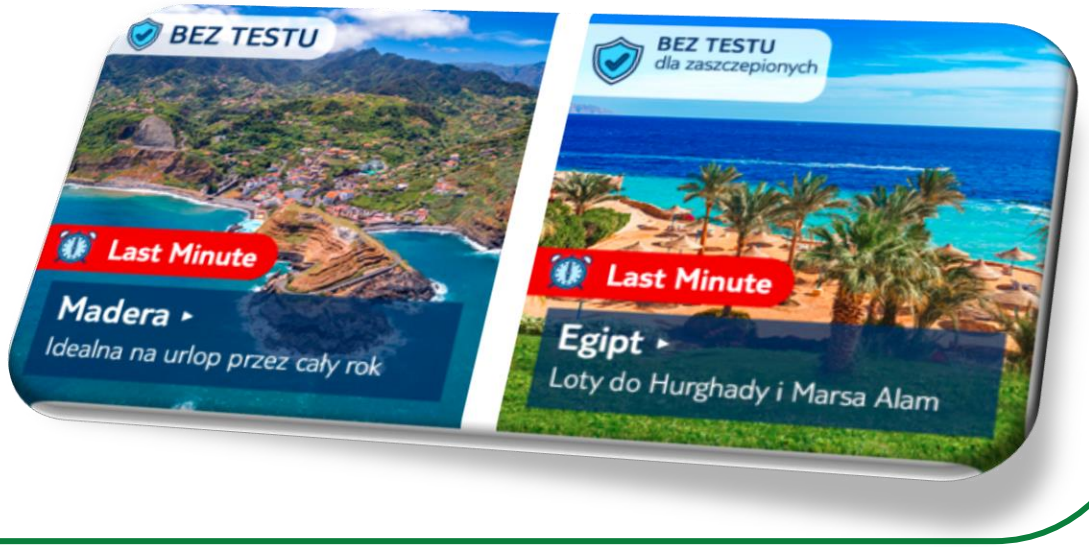
iBigWorld:2020-1-PL01-KA203-082197

Co-funded by the Erasmus+ Programme of the European Union

Covid Big Data i podróże

- Covid Big Data Hub data
- System dla podróżnych



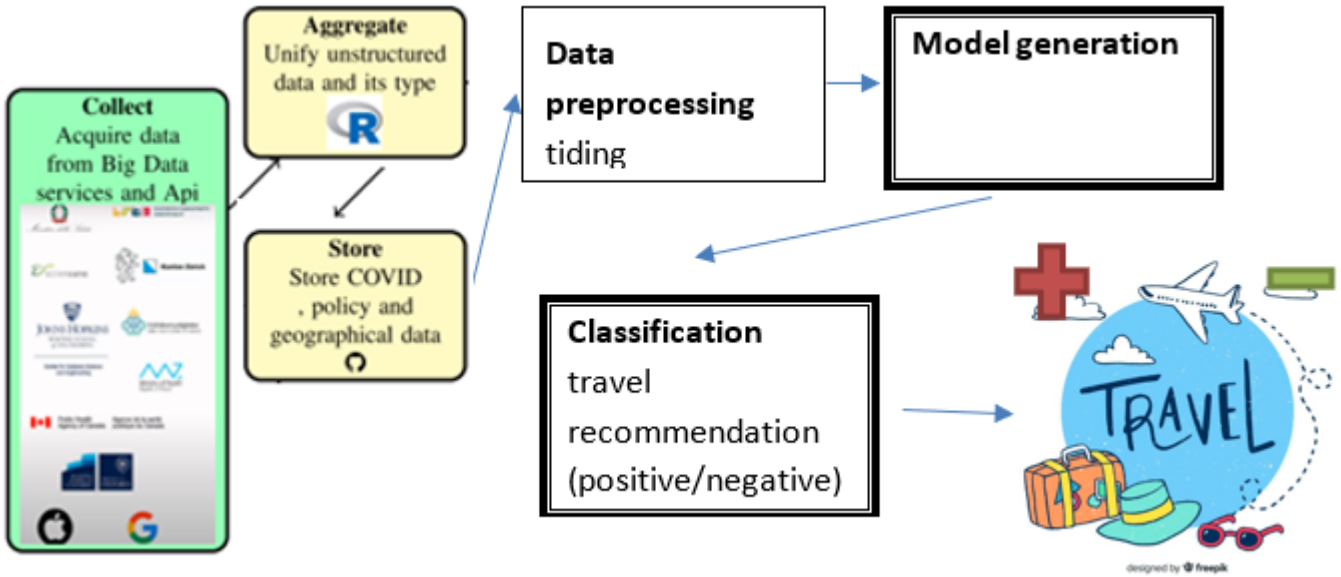


iBigWorld:2020-1-PL01-KA203-082197

Co-funded by the
Erasmus+ Programme
of the European Union

Przykładowy potok

- Big Data służy do analizy aktualnej sytuacji nosicielstwa w każdym kraju



Co-funded by the Erasmus+ Programme of the European Union

Akwizycja danych

- Akwizycja zostanie przeprowadzona przez połączenie z centrum danych Covid-19. To zadanie można wykonać poprzez bezpośrednie połączenie API w Pythonie.

```
#Create SparkSession

spark = SparkSession.builder.appName("SparkByExamples.com").getOrCreate()

sc=spark.sparkContext


from pyspark.sql import SQLContext
```

Jeśli ta operacja zostanie sfinalizowana bez błędu. Dane można uzyskać:

```
spark.sparkContext.addFile('https://storage.covid19datahub.io/level/1.csv')

df = spark.read.csv(SparkFiles.get("1.csv"), header=True)
```


Co-funded by the Erasmus+ Programme of the European Union

Weryfikacja

- Na każdym kroku dane powinny być weryfikowane
- Przykładową ramkę danych można wyświetlić:
  - df.show()

id	date	confirmed	deaths	recovered	tests	vaccines	people_vaccinated	people_fully_vaccinated	hosp	icu	vent	school_closing
8dea749d	2021-04-27	null	null	null	null	2400	2400	null	null	null	null	null	null
8dea749d	2021-04-28	null	null	null	null	null	null	null	null	null	null	null	null
8dea749d	2021-04-29	null	null	null	null	null	null	null	null	null	null	null	null
8dea749d	2021-04-30	null	null	null	null	null	null	null	null	null	null	null	null
8dea749d	2021-05-01	null	null	null	null	null	null	null	null	null	null	null	null
8dea749d	2021-05-02	null	null	null	null	null	null	null	null	null	null	null	null
8dea749d	2021-05-03	null	null	null	null	null	null	null	null	null	null	null	null

iBigWorld:2020-1-PL01-KA203-082197

Co-funded by the Erasmus+ Programme of the European Union

Wstępne przetwarzanie

- Zestaw danych zawiera wiele wartości null, dlatego chcielibyśmy zredukować zestaw do wartości bez tekstu. Wykonujemy to poprzez operację:

```
df = df.na.drop().dropna()
```

- Przed użyciem danych do klasyfikacji przydatne byłoby przeanalizowanie rodzaju atrybutów, które można wykorzystać. Wybierając atrybuty używamy następującej operacji:

```
df.describe().toPandas().transpose()
```

deaths	114901	9070.355784544956	38019.575144957475	0	9998
recovered	114901	221425.94937380875	1307143.0714442944	0	99997
tests	114901	4119115.3995961742	2.4387220316502348E7	0	9999227
vaccines	114901	6320555.4125638595	7.73516276024037E7	0	999990
people_vaccinated	114901	2292638.470753083	2.229261468069429E7	0	999981
people_fully_vaccinated	114901	1344458.3581256908	1.3182489977202881E7	0	9999427
hosp	114901	595.6956939452225	4546.180546998785	0	999
icu	114901	80.29464495522232	417.26058991208157	0	999
vent	114901	15.922454982985352	157.4678695747927	0	999

iBigWorld:2020-1-PL01-KA203-082197

Co-funded by the Erasmus+ Programme of the European Union

Weryfikacja

- Wykorzystane zostaną dane poszczególnych krajów.

```
df.where("administrative_area_level_1 = 'Poland'")\
.describe().show()
```

```
-+-----------------+-----------------+-----------------+-----------------+------------------+-----------------+----------------------+
e|        confirmed|           deaths|        recovered|            tests|          vaccines|people_vaccinated|people_fully_vaccinated|
-+-----------------+-----------------+-----------------+-----------------+------------------+-----------------+----------------------+
6|              666|              666|              666|              666|               666|              666|                   666|
l|1320986.3828828828|32845.993993993994|730024.8858858859|8304218.719219219|  8176074.402402403|4657407.614114114|      3805497.03003003|
l|1257534.8457850073|32217.860283281472| 997022.047550355|7475097.679367347|1.3744549895910164E7|7510132.642690689|     6873304.652630386|
2|                0|                0|                0|                0|                 0|                0|                     0|
7|           999924|              996|           999610|          9972993|           9806966|          9987919|               9929923|
-+-----------------+-----------------+-----------------+-----------------+------------------+-----------------+----------------------+
```

```
df.where("administrative_area_level_1 = 'Germany'")\
.describe().show()
```

```
+-----------------+-----------------+-----------------+-----------------+-----------------+-----------------+----------------------+
|        confirmed|           deaths|        recovered|            tests|          vaccines|people_vaccinated|people_fully_vaccinated|
+-----------------+-----------------+-----------------+-----------------+------------------+-----------------+----------------------+
|              663|              663|              663|              663|               663|              663|                   663|
|1841591.8612368023|47535.858220211165|1787275.1478129714|4858724.852187029|2.7697745815987933E7|1.5930731660633484E7|  1.228045838612368E7|
|1673659.1870337324| 39329.44948264935|1622866.8072382798|1.5726457632497796E7| 4.123837671737309E7|2.2516564205440015E7|  1.991327742588989E7|
|                0|                0|                0|                0|                 0|                0|                     0|
|            99896|             9993|           995021|          9260128|          99970160|          9966023|                981705|
+-----------------+-----------------+-----------------+-----------------+------------------+-----------------+----------------------+
```

iBigWorld:2020-1-PL01-KA203-082197

Co-funded by the Erasmus+ Programme of the European Union

Przetwarzanie danych

- Tworzymy trzy kolumny, w których obliczona zostanie względna liczba osób zaszczepionych, chorych i liczba zgonów.

```
dfb = df.withColumn("people_vaccinatedI",col("people_fully_vaccinated").cast("double")/col("population").cast("double"))\
.withColumn("confirmedI",col("confirmed").cast("double")/col("population").cast("double"))\
.withColumn("deathsI",col("deaths").cast("double")/col("population").cast("double"))\
.where("population>1000")
```

```
+------------------+----------+-------+
|people_vaccinatedI|confirmedI|deathsI|
+------------------+----------+-------+
|               0.0|       0.0|    0.0|
|               0.0|       0.0|    0.0|
|               0.0|       0.0|    0.0|
|               0.0|       0.0|    0.0|
|               0.0|       0.0|    0.0|
|               0.0|       0.0|    0.0|
```

iBigWorld:2020-1-PL01-KA203-082197

Co-funded by the Erasmus+ Programme of the European Union

Weryfikacja

- Poszczególne kraje możemy zobaczyć według średniej sytuacji za pomocą:

```
dfb.groupBy("administrative_area_level_1")\
  .agg(\
  avg("people_vaccinatedI"),\
  avg("confirmedI"),\
  avg("deathsI"))\
  .orderBy("avg(deathsI)")\
```

administrative_area_level_1	avg(people_vaccinatedI)	avg(confirmedI)	avg(deathsI)
Macao	0.10073494040050726	0.0	0.0
Guernsey	0.002527788033813...	0.0	0.0
Kiribati	8.203711579716138E-4	4.704869742376127E-6	0.0
Tokelau	0.005155572598770758	0.0	0.0
Micronesia	0.0	3.999027436527444E-6	0.0
Niue	0.013744186046511626	0.0	0.0
Palau	0.0	3.882253840370779E-5	0.0
Greenland	0.04691682311494895	0.001850230417834...	0.0
Tuvalu	0.008428433559942754	0.0	0.0
Turkmenistan	0.003012738677586...	0.0	0.0
Jersey	0.13796394439727838	0.0	0.0
Nauru	0.014189792450226009	0.0	0.0
Samoa	0.006271029097725...	7.609710357887618E-6	0.0
Tonga	0.004141078198463587	2.764472661853399...	0.0
Burundi	7.107538503791945E-8	3.361368587672341E-4	7.345352173956666E-7
Laos	0.013052139546809874	6.13219707245954E-4	7.754684624650199E-7
Bhutan	0.020955242232850488	0.0012111724128256 59	1.064832571976055...
Vanuatu	0.001179782693722...	5.807365380961095E-6	1.077338100708330...
Tanzania	7.545922225450822E-5	4.11770071988088E-5	1.240987935057866...
Eritrea	0.0	4.194834913669479E-4	1.884177214974042...

Co-funded by the Erasmus+ Programme of the European Union

Klasyfikacja

- Przygotowywanie danych dla klasyfikatora Big Data odbywa się w krokach:
- W pierwszym kroku musimy zebrać tylko potrzebne informacje. W naszym przypadku będą to trzy lub dwa parametry

```
Cdata = dfb.select(col("deathsI").alias("D"),col("people_vaccinatedI").a
lias("V"), col("confirmedI").alias("C"))\
.where("C>0")
```

```
CdataW = dfb.select(col("deathsI").alias("D"), col("confirmedI").alias("
C"))\
.where("C>0")
```

iBigWorld:2020-1-PL01-KA203-082197

Co-funded by the
Erasmus+ Programme
of the European Union

Przekształcanie danych dla MLLIb

- Modyfikacja dla biblioteki

```
from pyspark.ml.feature import VectorAssembler

vectorAssembler = VectorAssembler(inputCols = ['D','V','C'], outputCol='features')

vCdata = vectorAssembler.transform(Cdata)

vCdata = vCdata.select(['features'])

vCdata.show()
```

- Standaryzacja danych

```
from pyspark.ml.feature import StandardScaler

scale=StandardScaler(inputCol='features',outputCol='standardized')

data_scale=scale.fit(assembled_data)

data_scale_output=data_scale.transform(assembled_data)

data_scale_output.show(2)
```

iBigWorld:2020-1-PL01-KA203-082197

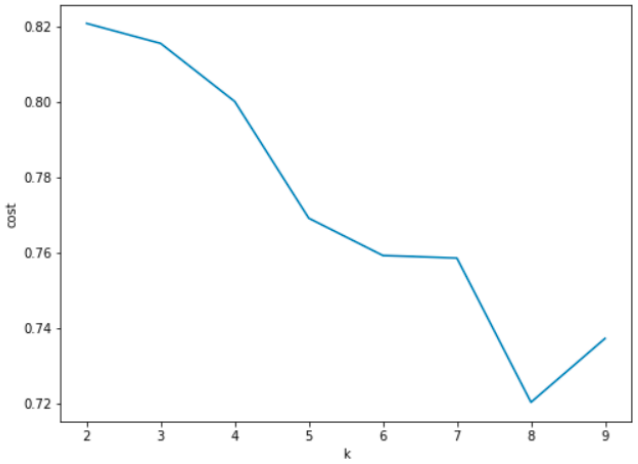
Co-funded by the Erasmus+ Programme of the European Union

Rekomendacja przez grupowanie

- Trening przy użyciu algorytmu k-średnich (przy użyciu k=2)

```
KMeans_algo=KMeans(featuresCol='standardized', k=i
KMeans_fit=KMeans_algo.fit(data_scale_output)
output=KMeans_fit.transform(data_scale_output)
score=evaluator.evaluate(output)
silhouette_score.append(score)
print("Silhouette Score for:",i," is:",score)
```

```
Silhouette Score for: 2  is: 0.8207894303443868
Silhouette Score for: 3  is: 0.8154741212163766
Silhouette Score for: 4  is: 0.8000668811544062
Silhouette Score for: 5  is: 0.7690749121371553
Silhouette Score for: 6  is: 0.7591842970143277
```



iBigWorld:2020-1-PL01-KA203-082197

Co-funded by the Erasmus+ Programme of the European Union

Prezentacja danych

- Teraz wytrenowany model może służyć do udzielania rekomendacji. Pierwsza klasa będzie prezentować rekomendację „-”, a druga rekomendację „+”.

```
recom = df.withColumn("people_vaccinatedI",col("people_fully_vaccinated").cast("double
")/col("population").cast("double"))\

.withColumn("confirmedI",col("confirmed").cast("double")/col("population").cast("doubl
e"))\

.withColumn("deathsI",col("deaths").cast("double")/col("population").cast("double"))\

.where("population>1000 AND date='2021-11-16'") # change to current date



Recom = recom.select(col("administrative_area_level_1").alias("Name"),col("deathsI").a
lias("D"),col("people_vaccinatedI").alias("V"), col("confirmedI").alias("C"))\

.where("C>0")
```

iBigWorld:2020-1-PL01-KA203-082197

Co-funded by the Erasmus+ Programme of the European Union

Prezentacja wyników

Name	features
Lithuania	[0.00228624011839...
Ireland	[0.00114354769750...
Croatia	[0.00244285311348...
Latvia	[0.00196972354338...
Liechtenstein	[0.00152993932999...
Australia	[7.62528035413963...
Switzerland	[0.00128223997786...
Austria	[0.00129924469383...
Netherlands	[0.00109014681378...
Italy	[0.00219942285693...
Portugal	[0.00177696580123...
Brazil	[0.00292047046333...
Turkey	[8.98606025452660...
Chile	[0.00202918870894...
France	[0.00176548383912...
Colombia	[0.00298827652736...
China	[4.09052723787094...
Peru	[0.00627382518680...
Taiwan	[3.59275617315373...
Thailand	[2.90125712596165...

Name	Recomendation
Lithuania	-
Ireland	-
Croatia	-
Latvia	-
Liechtenstein	-
Australia	-
Switzerland	-
Austria	-
Netherlands	-
Italy	-
Portugal	-
Brazil	-
Turkey	-
Chile	-
France	-
Colombia	-
China	+
Peru	-
Taiwan	+
Thailand	-

iBigWorld:2020-1-PL01-KA203-082197

Co-funded by the
Erasmus+ Programme
of the European Union