BIG DATA ANALYTICS REPORT - INDIVIDUAL PROJECT 2

STUDYING ABROAD: FACTORS INVOLVED

December 19, 2016

Pratik Shirish Kamath (N14671569)

Contents

1 ABSTRACT

Studying abroad is not only academically and culturally fulfilling, but also fosters personal growth. It is not only a good experience seeing and exploring a new country, but learning to interact with people from a variety of cultural background and learning from a curriculum you design is somewhat every student expects to take out from studying abroad. There are many factors, however, which affect the number of students going abroad for studying namely GDP, percentage of GDP invested by country on education etc. In this project, I have managed to venture into few of these factors and bring about conclusive results as to which factors are the most responsible[1].

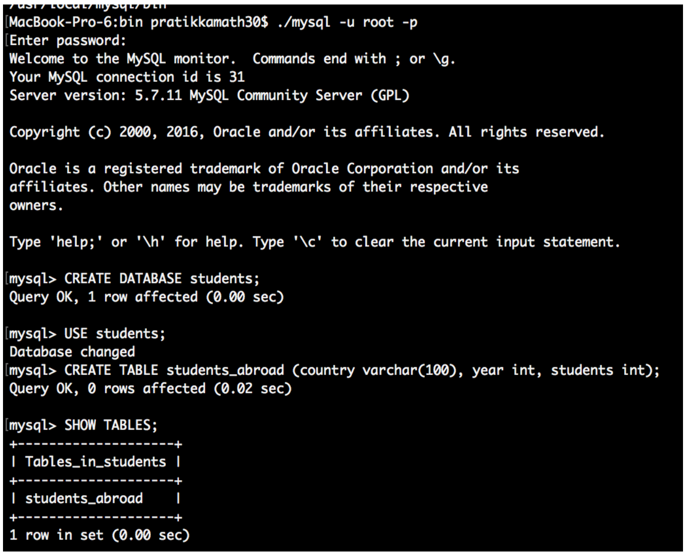
2 INTRODUCTION

For this project, I have used HIVE and MySQL for pre-processing the data i.e students abroad, gdp for countries, percentage of GDP invested in education datasets. I have created the tables in MySQL witht the same "skeleton-structure" as that of our datasets involved and loaded the data in the tables.After that I created an index(ID) for the tables using concatenation of country and year.

After that, I then imported them in HIVE and performed a join operation on them. This table was then used for Machine Learning and Information visualization analysis in R. Also, I have used Map Reduce analysis in Cloudera Hadoop environment for categorizing the output on basis of country using Apache Pig.

3 SQL PREPROCESSING

First of all I created tables in SQL for students studying abroad, GDP and %age of GDP invested in education as follows. Then I loaded the data in them using UNESCO datasets. After that, I created an index using country and year columns to create an ID for these tables so that the procedure of joining the tables becomes easier.Here in this way I have demonstrated effective data manipulation so that I will get all the statistics for a given country according to it's year because we need yearwise distribution for a country. I have concatenated two main factors which are important and made a new key which can be used as a primary index and can be used to join the tables which otherwise would have been very difficult to join. As seen from the figures, I have created skeletons of tables, loaded data from the UNESCO datasets and exported them as .txt files.



```
MacBook-Pro-6:bin pratikkamath30$ ./mysql -u root -p
Enter password:
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 31
Server version: 5.7.11 MySQL Community Server (GPL)

Copyright (c) 2000, 2016, Oracle and/or its affiliates. All rights reserved.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

mysql> CREATE DATABASE students;
Query OK, 1 row affected (0.00 sec)

mysql> USE students;
Database changed
mysql> CREATE TABLE students_abroad (country varchar(100), year int, students int);
Query OK, 0 rows affected (0.02 sec)

mysql> SHOW TABLES;
+--------------------+
| Tables_in_students |
+--------------------+
| students_abroad    |
+--------------------+
1 row in set (0.00 sec)
```

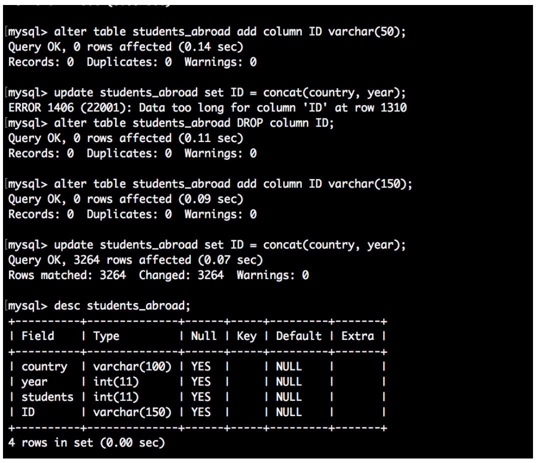
Figure 1

```
mysql> SELECT * FROM students_abroad;
Empty set (0.00 sec)

mysql> LOAD DATA LOCAL INFILE '/Users/pratikkamath30/Documents/Big Data 2/students_abroad.csv' into table students_abroad columns terminated by ',' optionally enclosed by '"' escaped by '"' lines terminated by '
\n' ignore 1 lines;
Query OK, 3264 rows affected (0.02 sec)
Records: 3264  Deleted: 0  Skipped: 0  Warnings: 0

mysql> SELECT * FROM students_abroad limit 20;
+-------------+------+----------+
| country     | year | students |
+-------------+------+----------+
| Afghanistan | 2013 |    12003 |
| Afghanistan | 2012 |     9686 |
| Afghanistan | 2011 |     9429 |
| Afghanistan | 2010 |     7865 |
| Afghanistan | 2009 |     5545 |
| Afghanistan | 2008 |     4484 |
| Afghanistan | 2007 |     3849 |
| Afghanistan | 2006 |     3308 |
| Afghanistan | 2005 |     3466 |
| Afghanistan | 2004 |     3175 |
| Afghanistan | 2003 |     3100 |
| Afghanistan | 2002 |     3086 |
| Afghanistan | 2001 |     2887 |
| Afghanistan | 2000 |     2930 |
| Afghanistan | 1999 |     2902 |
| Afghanistan | 1998 |     2934 |
| Albania     | 2013 |    24147 |
| Albania     | 2012 |    24470 |
| Albania     | 2011 |    25463 |
| Albania     | 2010 |    23813 |
+-------------+------+----------+
20 rows in set (0.00 sec)
```

Figure 2



```
mysql> alter table students_abroad add column ID varchar(50);
Query OK, 0 rows affected (0.14 sec)
Records: 0  Duplicates: 0  Warnings: 0

mysql> update students_abroad set ID = concat(country, year);
ERROR 1406 (22001): Data too long for column 'ID' at row 1310
mysql> alter table students_abroad DROP column ID;
Query OK, 0 rows affected (0.11 sec)
Records: 0  Duplicates: 0  Warnings: 0

mysql> alter table students_abroad add column ID varchar(150);
Query OK, 0 rows affected (0.09 sec)
Records: 0  Duplicates: 0  Warnings: 0

mysql> update students_abroad set ID = concat(country, year);
Query OK, 3264 rows affected (0.07 sec)
Rows matched: 3264  Changed: 3264  Warnings: 0

mysql> desc students_abroad;
+----------+--------------+------+-----+---------+-------+
| Field    | Type         | Null | Key | Default | Extra |
+----------+--------------+------+-----+---------+-------+
| country  | varchar(100) | YES  |     | NULL    |       |
| year     | int(11)      | YES  |     | NULL    |       |
| students | int(11)      | YES  |     | NULL    |       |
| ID       | varchar(150) | YES  |     | NULL    |       |
+----------+--------------+------+-----+---------+-------+
4 rows in set (0.00 sec)
```

Figure 3

```
mysql> SELECT * FROM students_abroad limit 30;
+-------------+------+----------+-----------------+
| country     | year | students | ID              |
+-------------+------+----------+-----------------+
| Afghanistan | 2013 |    12003 | Afghanistan2013 |
| Afghanistan | 2012 |     9686 | Afghanistan2012 |
| Afghanistan | 2011 |     9429 | Afghanistan2011 |
| Afghanistan | 2010 |     7865 | Afghanistan2010 |
| Afghanistan | 2009 |     5545 | Afghanistan2009 |
| Afghanistan | 2008 |     4484 | Afghanistan2008 |
| Afghanistan | 2007 |     3849 | Afghanistan2007 |
| Afghanistan | 2006 |     3308 | Afghanistan2006 |
| Afghanistan | 2005 |     3466 | Afghanistan2005 |
| Afghanistan | 2004 |     3175 | Afghanistan2004 |
| Afghanistan | 2003 |     3100 | Afghanistan2003 |
| Afghanistan | 2002 |     3086 | Afghanistan2002 |
| Afghanistan | 2001 |     2887 | Afghanistan2001 |
| Afghanistan | 2000 |     2930 | Afghanistan2000 |
| Afghanistan | 1999 |     2902 | Afghanistan1999 |
| Afghanistan | 1998 |     2934 | Afghanistan1998 |
| Albania     | 2013 |    24147 | Albania2013     |
| Albania     | 2012 |    24470 | Albania2012     |
| Albania     | 2011 |    25463 | Albania2011     |
| Albania     | 2010 |    23813 | Albania2010     |
| Albania     | 2009 |    22733 | Albania2009     |
| Albania     | 2008 |    20995 | Albania2008     |
| Albania     | 2007 |    19930 | Albania2007     |
| Albania     | 2006 |    17465 | Albania2006     |
| Albania     | 2005 |    15240 | Albania2005     |
| Albania     | 2004 |    13645 | Albania2004     |
| Albania     | 2003 |    11475 | Albania2003     |
| Albania     | 2002 |     8608 | Albania2002     |
| Albania     | 2001 |     7386 | Albania2001     |
| Albania     | 2000 |     5827 | Albania2000     |
+-------------+------+----------+-----------------+
30 rows in set (0.00 sec)
```

Figure 4

```
mysql> SELECT * FROM students_abroad INTO OUTFILE '/tmp/students_abroad.txt' FIELDS TERMINATED BY '\t' OPTIONALLY ENCLOSED BY '"' LINES TERMINATED BY '\n';
Query OK, 3264 rows affected (0.00 sec)
```

Figure 5

```
mysql> CREATE TABLE gdp(country varchar(100), year int, gdp bigint)
    -> ;
Query OK, 0 rows affected (0.02 sec)

mysql> desc gdp;
+---------+--------------+------+-----+---------+-------+
| Field   | Type         | Null | Key | Default | Extra |
+---------+--------------+------+-----+---------+-------+
| country | varchar(100) | YES  |     | NULL    |       |
| year    | int(11)      | YES  |     | NULL    |       |
| gdp     | bigint(20)   | YES  |     | NULL    |       |
+---------+--------------+------+-----+---------+-------+
3 rows in set (0.00 sec)
```

Figure 6

```
mysql> LOAD DATA LOCAL INFILE '/Users/pratikkamath30/Documents/Big Data 2/gdp.csv' INTO TABLE gdp COLUMNS TERMINATED BY ',' OPTIONALLY ENCLOSED BY '"' ESCAPED BY '"' LINES TERMINATED BY '\n' IGNORE 1 LINES;

Query OK, 10299 rows affected, 10304 warnings (0.06 sec)
Records: 10299  Deleted: 0  Skipped: 0  Warnings: 10304

mysql>
```

Figure 7

```
mysql> select * from gdp limit 30;
+-------------+------+-------------+
| country     | year | gdp         |
+-------------+------+-------------+
| Afghanistan | 2014 | 20038215159 |
| Afghanistan | 2013 | 20458939155 |
| Afghanistan | 2012 | 20536542737 |
| Afghanistan | 2011 | 17930239400 |
| Afghanistan | 2010 | 15936800636 |
| Afghanistan | 2009 | 12486943506 |
| Afghanistan | 2008 | 10190529882 |
| Afghanistan | 2007 |  9843842455 |
| Afghanistan | 2006 |  7057598407 |
| Afghanistan | 2005 |  6275076016 |
| Afghanistan | 2004 |  5285461999 |
| Afghanistan | 2003 |  4583648922 |
| Afghanistan | 2002 |  4128818042 |
| Afghanistan | 2001 |  2461666315 |
| Afghanistan | 1981 |  3478787910 |
| Afghanistan | 1980 |  3641723447 |
| Afghanistan | 1979 |  3697940345 |
| Afghanistan | 1978 |  3300000108 |
| Afghanistan | 1977 |  2953333419 |
| Afghanistan | 1976 |  2555555567 |
| Afghanistan | 1975 |  2366666615 |
| Afghanistan | 1974 |  2155555499 |
| Afghanistan | 1973 |  1733333265 |
| Afghanistan | 1972 |  1595555476 |
| Afghanistan | 1971 |  1831108972 |
| Afghanistan | 1970 |  1748886596 |
| Afghanistan | 1969 |  1408888923 |
| Afghanistan | 1968 |  1373333367 |
| Afghanistan | 1967 |  1673333419 |
| Afghanistan | 1966 |  1399999966 |
+-------------+------+-------------+
30 rows in set (0.00 sec)

mysql> alter table gdp add column ID varchar(150);
Query OK, 0 rows affected (0.34 sec)
Records: 0  Duplicates: 0  Warnings: 0

mysql> update gdp set ID = concat(country, year)
    -> ;
Query OK, 10299 rows affected (0.18 sec)
Rows matched: 10299  Changed: 10299  Warnings: 0
```

Figure 8

```
mysql> desc gdp;
+---------+--------------+------+-----+---------+-------+
| Field   | Type         | Null | Key | Default | Extra |
+---------+--------------+------+-----+---------+-------+
| country | varchar(100) | YES  |     | NULL    |       |
| year    | int(11)      | YES  |     | NULL    |       |
| gdp     | bigint(20)   | YES  |     | NULL    |       |
| ID      | varchar(150) | YES  |     | NULL    |       |
+---------+--------------+------+-----+---------+-------+
4 rows in set (0.00 sec)
```

Figure 9

```
mysql> CREATE TABLE education_expenditure(country varchar(100), year int, expenditure int);
Query OK, 0 rows affected (0.01 sec)

mysql> LOAD DATA FROM LOCAL INFILE '/Users/pratikkamath30/Documents/Big Data 2/students_abroad.csv' into table education_expenditure columns terminated by ',' optionally enclosed by '"' escaped by '"' lines term
inated by '\n' ignore 1 lines;
ERROR 1064 (42000): You have an error in your SQL syntax; check the manual that corresponds to your MySQL server version for the right syntax to use near 'FROM LOCAL INFILE '/Users/pratikkamath30/Documents/Big D
ata 2/students_abroad.cs' at line 1
mysql> LOAD DATA LOCAL INFILE '/Users/pratikkamath30/Documents/Big Data 2/students_abroad.csv' into table education_expenditure columns terminated by ',' optionally enclosed by '"' escaped by '"' lines terminate
d by '\n' ignore 1 lines;
Query OK, 3264 rows affected (0.06 sec)
Records: 3264  Deleted: 0  Skipped: 0  Warnings: 0

mysql> SELECT * FROM education_expenditure limit 20;
+-------------+------+-------------+
| country     | year | expenditure |
+-------------+------+-------------+
| Afghanistan | 2013 |       12003 |
| Afghanistan | 2012 |        9686 |
| Afghanistan | 2011 |        9429 |
| Afghanistan | 2010 |        7865 |
| Afghanistan | 2009 |        5545 |
| Afghanistan | 2008 |        4484 |
| Afghanistan | 2007 |        3849 |
| Afghanistan | 2006 |        3308 |
| Afghanistan | 2005 |        3466 |
| Afghanistan | 2004 |        3175 |
| Afghanistan | 2003 |        3100 |
| Afghanistan | 2002 |        3086 |
| Afghanistan | 2001 |        2887 |
| Afghanistan | 2000 |        2930 |
| Afghanistan | 1999 |        2902 |
| Afghanistan | 1998 |        2934 |
| Albania     | 2013 |       24147 |
| Albania     | 2012 |       24470 |
| Albania     | 2011 |       25463 |
| Albania     | 2010 |       23813 |
+-------------+------+-------------+
20 rows in set (0.01 sec)
```

Figure 10

4 HIVE PREPROCESSING

Here, similar to the skeleton created in SQL I have created skeleton-structure of the table. After that I used the three .txt files I had exported from MySQL and loaded them in the tables.Now we know that we have an ID field which is common in all the tables and the join operation can now be done easily and therefore I proceeded to do the join operation using the ID field as primary index for all the tables.

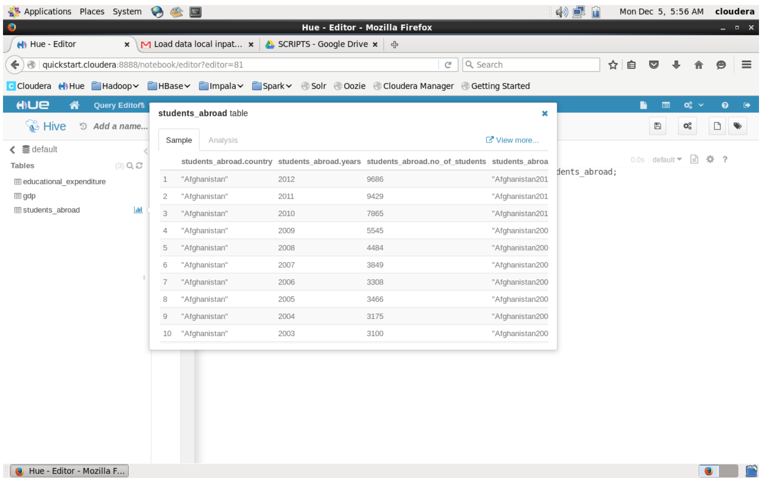


Figure 11

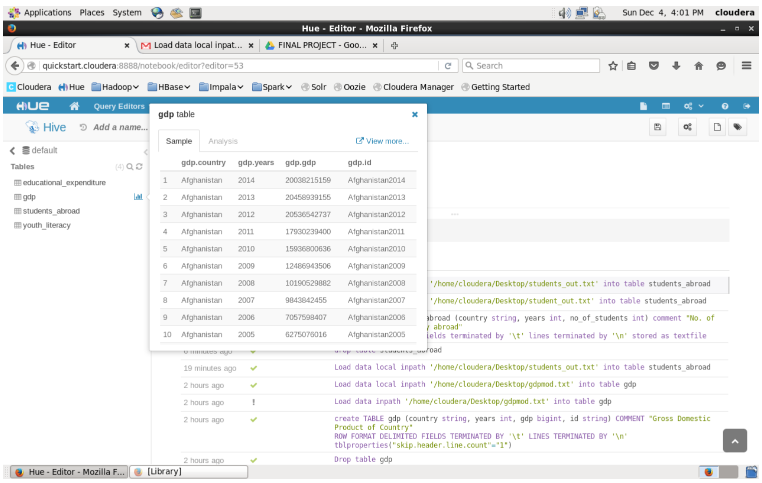
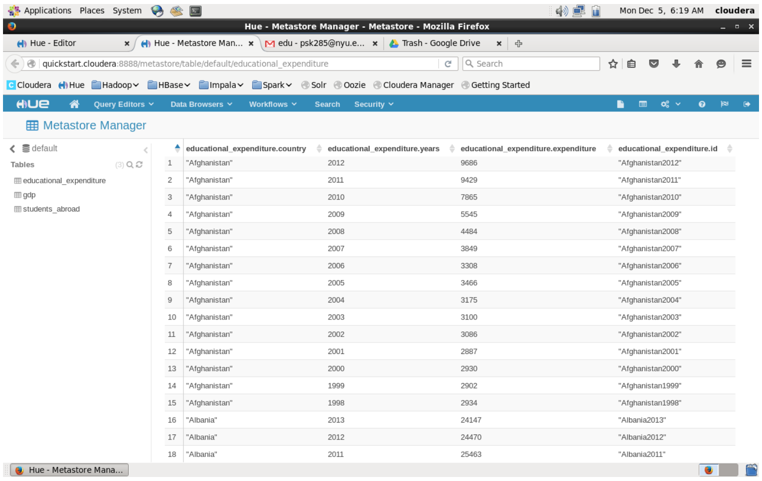


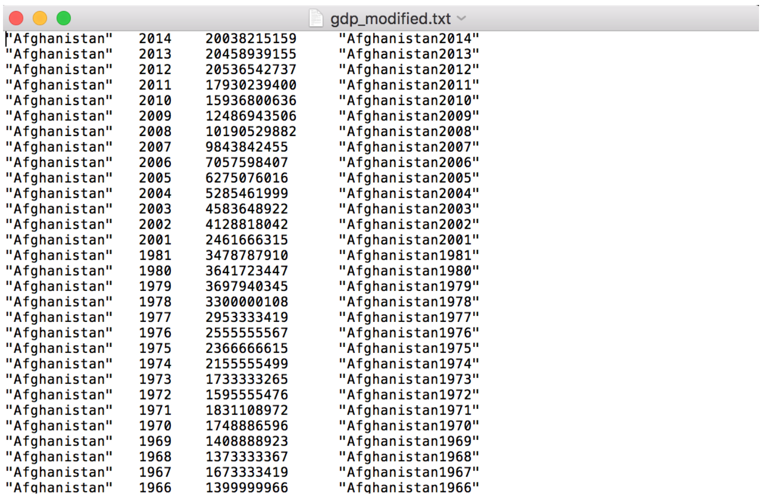
Figure 12



	educational_expenditure.country	educational_expenditure.years	educational_expenditure.expenditure	educational_expenditure.id
1	"Afghanistan"	2012	9686	"Afghanistan2012"
2	"Afghanistan"	2011	9429	"Afghanistan2011"
3	"Afghanistan"	2010	7865	"Afghanistan2010"
4	"Afghanistan"	2009	5545	"Afghanistan2009"
5	"Afghanistan"	2008	4484	"Afghanistan2008"
6	"Afghanistan"	2007	3849	"Afghanistan2007"
7	"Afghanistan"	2006	3308	"Afghanistan2006"
8	"Afghanistan"	2005	3466	"Afghanistan2005"
9	"Afghanistan"	2004	3175	"Afghanistan2004"
10	"Afghanistan"	2003	3100	"Afghanistan2003"
11	"Afghanistan"	2002	3086	"Afghanistan2002"
12	"Afghanistan"	2001	2887	"Afghanistan2001"
13	"Afghanistan"	2000	2930	"Afghanistan2000"
14	"Afghanistan"	1999	2902	"Afghanistan1999"
15	"Afghanistan"	1998	2934	"Afghanistan1998"
16	"Albania"	2013	24147	"Albania2013"
17	"Albania"	2012	24470	"Albania2012"
18	"Albania"	2011	25463	"Albania2011"

Figure 13

The .txt files exported were as follows:



gdp_modified.txt

"Afghanistan"	2014	20038215159	"Afghanistan2014"
"Afghanistan"	2013	20458939155	"Afghanistan2013"
"Afghanistan"	2012	20536542737	"Afghanistan2012"
"Afghanistan"	2011	17930239400	"Afghanistan2011"
"Afghanistan"	2010	15936800636	"Afghanistan2010"
"Afghanistan"	2009	12486943506	"Afghanistan2009"
"Afghanistan"	2008	10190529882	"Afghanistan2008"
"Afghanistan"	2007	9843842455	"Afghanistan2007"
"Afghanistan"	2006	7057598407	"Afghanistan2006"
"Afghanistan"	2005	6275076016	"Afghanistan2005"
"Afghanistan"	2004	5285461999	"Afghanistan2004"
"Afghanistan"	2003	4583648922	"Afghanistan2003"
"Afghanistan"	2002	4128818042	"Afghanistan2002"
"Afghanistan"	2001	2461666315	"Afghanistan2001"
"Afghanistan"	1981	3478787910	"Afghanistan1981"
"Afghanistan"	1980	3641723447	"Afghanistan1980"
"Afghanistan"	1979	3697940345	"Afghanistan1979"
"Afghanistan"	1978	3300000108	"Afghanistan1978"
"Afghanistan"	1977	2953333419	"Afghanistan1977"
"Afghanistan"	1976	2555555567	"Afghanistan1976"
"Afghanistan"	1975	2366666615	"Afghanistan1975"
"Afghanistan"	1974	2155555499	"Afghanistan1974"
"Afghanistan"	1973	1733333265	"Afghanistan1973"
"Afghanistan"	1972	1595555476	"Afghanistan1972"
"Afghanistan"	1971	1831108972	"Afghanistan1971"
"Afghanistan"	1970	1748886596	"Afghanistan1970"
"Afghanistan"	1969	1408888923	"Afghanistan1969"
"Afghanistan"	1968	1373333367	"Afghanistan1968"
"Afghanistan"	1967	1673333419	"Afghanistan1967"
"Afghanistan"	1966	1399999966	"Afghanistan1966"

Figure 14

students_abroad.txt

"Afghanistan"	2013	12003	"Afghanistan2013"
"Afghanistan"	2012	9686	"Afghanistan2012"
"Afghanistan"	2011	9429	"Afghanistan2011"
"Afghanistan"	2010	7865	"Afghanistan2010"
"Afghanistan"	2009	5545	"Afghanistan2009"
"Afghanistan"	2008	4484	"Afghanistan2008"
"Afghanistan"	2007	3849	"Afghanistan2007"
"Afghanistan"	2006	3308	"Afghanistan2006"
"Afghanistan"	2005	3466	"Afghanistan2005"
"Afghanistan"	2004	3175	"Afghanistan2004"
"Afghanistan"	2003	3100	"Afghanistan2003"
"Afghanistan"	2002	3086	"Afghanistan2002"
"Afghanistan"	2001	2887	"Afghanistan2001"
"Afghanistan"	2000	2930	"Afghanistan2000"
"Afghanistan"	1999	2902	"Afghanistan1999"
"Afghanistan"	1998	2934	"Afghanistan1998"
"Albania"	2013	24147	"Albania2013"
"Albania"	2012	24470	"Albania2012"
"Albania"	2011	25463	"Albania2011"
"Albania"	2010	23813	"Albania2010"
"Albania"	2009	22733	"Albania2009"
"Albania"	2008	20995	"Albania2008"
"Albania"	2007	19930	"Albania2007"
"Albania"	2006	17465	"Albania2006"
"Albania"	2005	15240	"Albania2005"
"Albania"	2004	13645	"Albania2004"
"Albania"	2003	11475	"Albania2003"
"Albania"	2002	8608	"Albania2002"
"Albania"	2001	7386	"Albania2001"
"Albania"	2000	5827	"Albania2000"

Figure 15

education_expenditure.txt

"Afghanistan"	2013	12003	"Afghanistan2013"
"Afghanistan"	2012	9686	"Afghanistan2012"
"Afghanistan"	2011	9429	"Afghanistan2011"
"Afghanistan"	2010	7865	"Afghanistan2010"
"Afghanistan"	2009	5545	"Afghanistan2009"
"Afghanistan"	2008	4484	"Afghanistan2008"
"Afghanistan"	2007	3849	"Afghanistan2007"
"Afghanistan"	2006	3308	"Afghanistan2006"
"Afghanistan"	2005	3466	"Afghanistan2005"
"Afghanistan"	2004	3175	"Afghanistan2004"
"Afghanistan"	2003	3100	"Afghanistan2003"
"Afghanistan"	2002	3086	"Afghanistan2002"
"Afghanistan"	2001	2887	"Afghanistan2001"
"Afghanistan"	2000	2930	"Afghanistan2000"
"Afghanistan"	1999	2902	"Afghanistan1999"
"Afghanistan"	1998	2934	"Afghanistan1998"
"Albania"	2013	24147	"Albania2013"
"Albania"	2012	24470	"Albania2012"
"Albania"	2011	25463	"Albania2011"
"Albania"	2010	23813	"Albania2010"
"Albania"	2009	22733	"Albania2009"
"Albania"	2008	20995	"Albania2008"
"Albania"	2007	19930	"Albania2007"
"Albania"	2006	17465	"Albania2006"
"Albania"	2005	15240	"Albania2005"
"Albania"	2004	13645	"Albania2004"
"Albania"	2003	11475	"Albania2003"
"Albania"	2002	8608	"Albania2002"
"Albania"	2001	7386	"Albania2001"
"Albania"	2000	5827	"Albania2000"

Figure 16

```
create table students_abroad (country string, years int, no_of_students int, id string) comment "No. of students going to study abroad"
row format delimited
fields terminated by '\t'
lines terminated by '\n'
tblproperties("skip.header.line.count"="1");

create TABLE gdp (country string, years int, gdp bigint, id string) COMMENT "Gross Domestic Product of Country"
ROW FORMAT DELIMITED
FIELDS TERMINATED BY '\t'
LINES TERMINATED BY '\n'
tblproperties("skip.header.line.count"="1");

create TABLE educational_expenditure (country string, years int, expenditure int, id string) COMMENT "Expenditure on Education in %"
ROW FORMAT DELIMITED
FIELDS TERMINATED BY '\t'
LINES TERMINATED BY '\n'
tblproperties("skip.header.line.count"="1");

Load data local inpath '/home/cloudera/Desktop/gdp_modified.txt' into table gdp;
Load data local inpath '/home/cloudera/Desktop/students_abroad.txt' into table students_abroad;
Load data local inpath '/home/cloudera/Desktop/education_expenditure.txt' into table educational_expenditure;

select students_abroad.country as country, students_abroad.years as years, students_abroad.number_of_students as students,
gdp as gdp, educational_expenditure.expenditure as expenditure
from students_abroad_mod
JOIN educational_expenditure on (educational_expenditure.id = students_abroad.id)
JOIN gdp on (gdp.id = students_abroad_mod.id);

#On Terminal: To export the joined data as a CSV.
hive -e 'select * from main_data' > /home/cloudera/Desktop/main_data.csv
```

Figure 17

Above I have mentioned all the commands I used in HIVE to do the preprocessing and after the join operation was completed I exported it as a csv for further Map Reduce and Machine Learning analysis.

country	years	students	gdp	expenditure
Afghanistan	2013	12003	2.0459E+10	4.57999992
Afghanistan	2012	9686	2.0537E+10	3.13000011
Afghanistan	2011	9429	1.793E+10	4.09000015
Afghanistan	2010	7865	1.5937E+10	4.51000023
Albania	2013	24147	1.2781E+10	3.5
Albania	2007	19930	1.0701E+10	3.26999998
Albania	2006	17465	8992642349	3.1099999
Albania	2005	15240	8158548717	3.1500001
Albania	2004	13645	7314865176	3.1099999
Albania	2003	11475	5746945913	3.11999989
Albania	2002	8608	4435078648	3.04999995
Albania	2001	7386	4060758804	3.31999993
Albania	2000	5827	3632043908	3.24000001
Albania	1999	4685	3414760915	3.3599999
Albania	1998	4596	2707123772	3.29999995
Algeria	2008	21987	1.71E+11	4.34000015
Andorra	2013	1177	3249100675	2.46000004
Andorra	2011	1377	3427235709	3.16000009
Andorra	2010	1249	3346317329	3.06999993
Andorra	2009	1314	3649863493	3.16000009
Andorra	2008	1200	4001349340	2.93000007
Andorra	2007	991	4010785102	2.06999993
Andorra	2006	305	3536451646	2.19000006
Andorra	2005	1085	3248134607	1.60000002
Andorra	2004	1224	2916913449	1.52999997
Andorra	2002	714	1717563533	1.70000005
Angola	2010	7916	8.3369E+10	3.48000002
Angola	2006	8268	5.2381E+10	2.8599999
Angola	2005	8020	3.6971E+10	2.77999997
Angola	2000	5359	9129634978	2.6099999

Figure 18

The final csv after SQL and HIVE preprocessing looks as shown above. However we need to do effecttive grouping now which can be done using Map Reduce analysis which I have achieved through Apache Pig in Cloudera Hadoop environment and Machine learning and Information visualization which I have done using R.

5 Map Reduce Analysis

For Map Reduce analysis, I used Apache Pig in Cloudera Hadoop environment which is basically an abstraction for the Java backend. It made my M/R easy to implement and made it equally effective. The basic need to do map reduce analysis was to group the data according to the factor of our choice. Here I have used country as the factor of choice and generated a count for the countries so that we come to know how many values are there for a particular country as well. The Apache Pig script I wrote for that was as follows:

```
A = LOAD './pig/main_data.csv' using PigStorage(',') AS (country:chararray, years:int, students:int, gdp:long,
expenditure:float);
B = GROUP A BY country;
C = FOREACH B GENERATE group as country, COUNT(A);
E = JOIN A by country, C by country;
F = FOREACH E GENERATE $0 as country,$1 as year,$2 as students,$3 as gdp, $4 as expenditure, $7 as counter;
G = FILTER F BY (counter > 2);
H = ORDER G BY country;;
STORE H INTO './PigScript_Result';
```

Figure 19

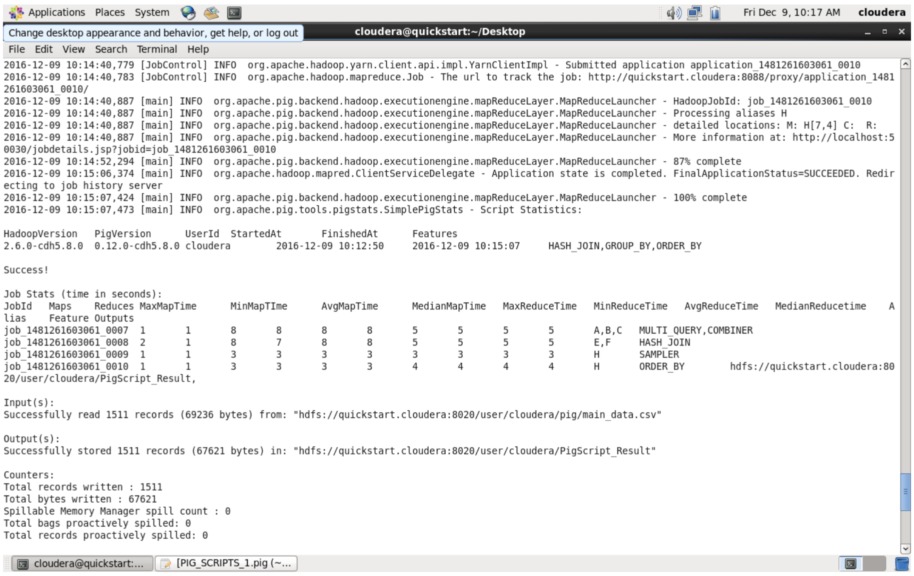


Figure 20

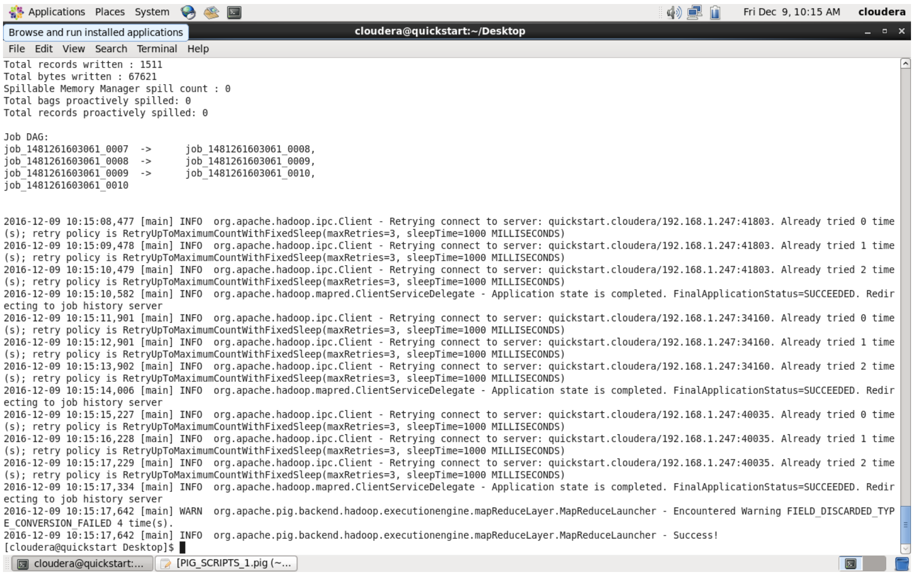


Figure 21

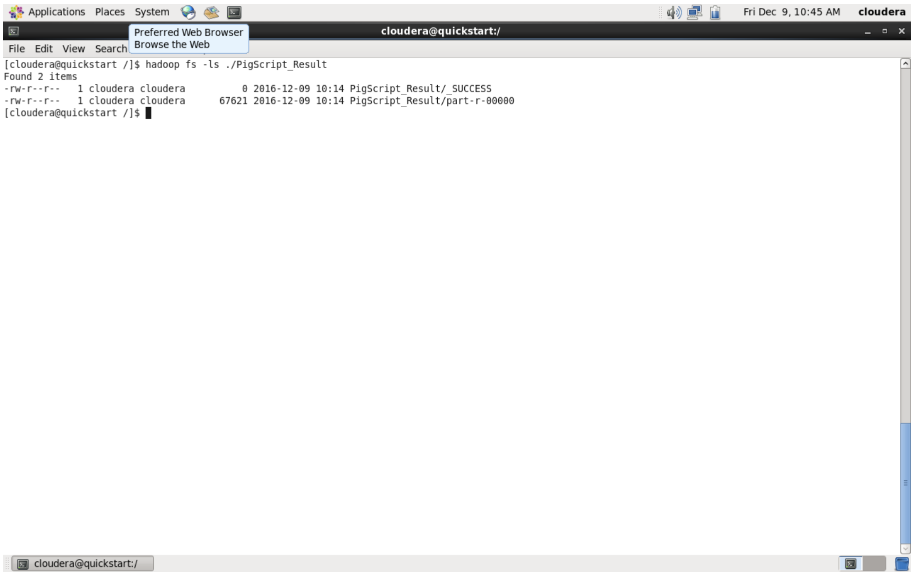


Figure 22

The final output that I generated through Pig is as follows which has count for each country as well. Thus I have successfully implemented Map Reduce analysis with my dataset. However I also wanted to do a machine learning analysis to analyze as to how the factors affect the number of students as I have done that analysis in R and all I have done is mentioned in my next section.

Cuba	2007	1442	58603500000	11.87	6	
Cuba	2010	2119	64328200000	12.84	6	
Cuba	2009	1755	62078600000	13.13	6	
Cuba	2008	1592	60806300000	14.06	6	
Cuba	2004	1199	38202800000	10.27	6	
Cuba	2006	1380	52742100000	9.06	6	
Peru	2013	14204	201848000000		3.28	6
Peru	2005	10134	74947898080	2.88	6	
Peru	2007	13374	102171000000		2.63	6
Peru	2009	15601	121192000000		3.13	6
Peru	2011	16424	170564000000		2.68	6
Peru	2012	15083	192680000000		2.92	6
Chile	1999	4696	72995286764	3.84	15	
Chile	2000	5070	79328640264	3.71	15	
Chile	2013	8937	276674000000		4.56	15
Chile	2012	8924	265232000000		4.57	15
Chile	2011	9860	250832000000		4.07	15
Chile	2010	9127	217538000000		4.18	15
Chile	2009	8242	171957000000		4.24	15
Chile	2008	7120	179627000000		3.79	15
Chile	2007	6177	173081000000		3.22	15

Figure 23

6 MACHINE LEARNING

As mentioned before I have done the Machine Learning and information visualization in R. Below are the graphs and explanation as to which Machine learning analysis I have applied to them.

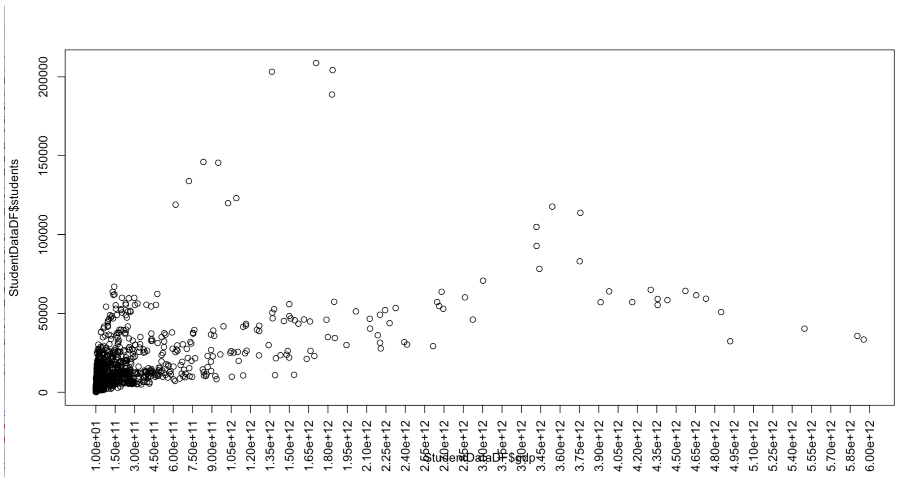


Figure 24

The graph above shows the variation of number of students each year w.r.t to GDP of country they belong to. this clearly shows that the number of students going abroad are densely populated in countries with lower GDP. We can say that no of students is inversely proportional to GDP of country which might be true because if there is high GDP there is high probability that there are adequate employment opportunities in those countries and vice-a-versa can be assumed too.

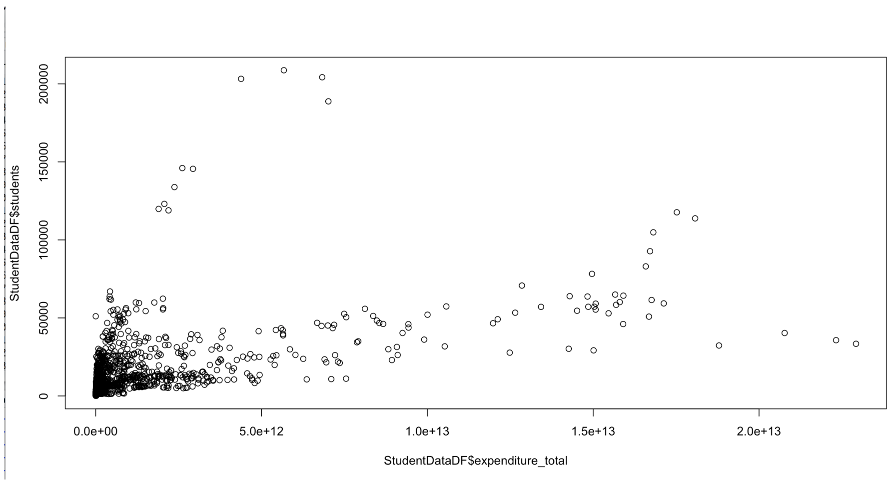


Figure 25

As the expenditure of the countries on education was given as a percentage of GDP, I calculated the expenditure for all the countries as expenditure_total = expenditure in percentage x GDP of the country and found out the values of expenditure and I plotted them as a function of students going abroad which is showing similar results.

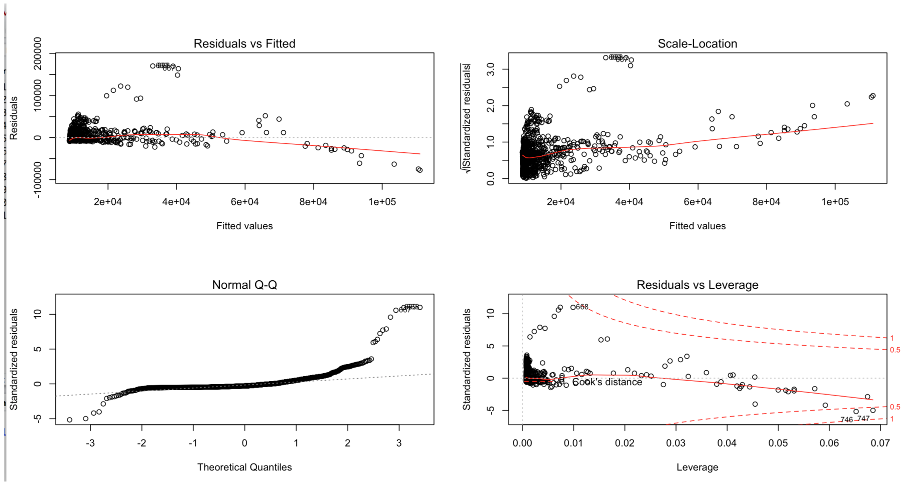


Figure 26

The above figure shows the plot of the regression analysis[4] I did using GDP and expenditure as factors of number of students going abroad. I had got R-squared value to be around 0.33 which shows there is good correlation but there might be other factors which might be needed to get better results. After that I did a 10 fold cross-validation[3] and still got a R-squared result more than 0.3 showing that my model is consistent.[5]

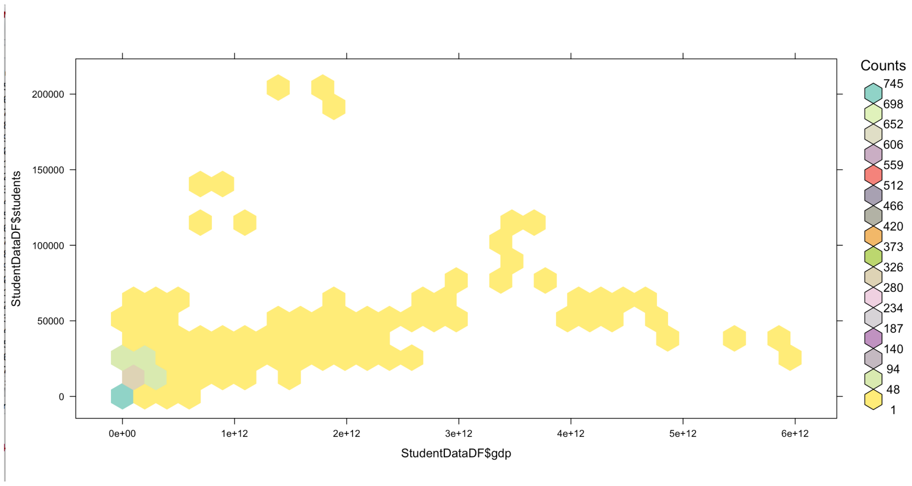


Figure 27

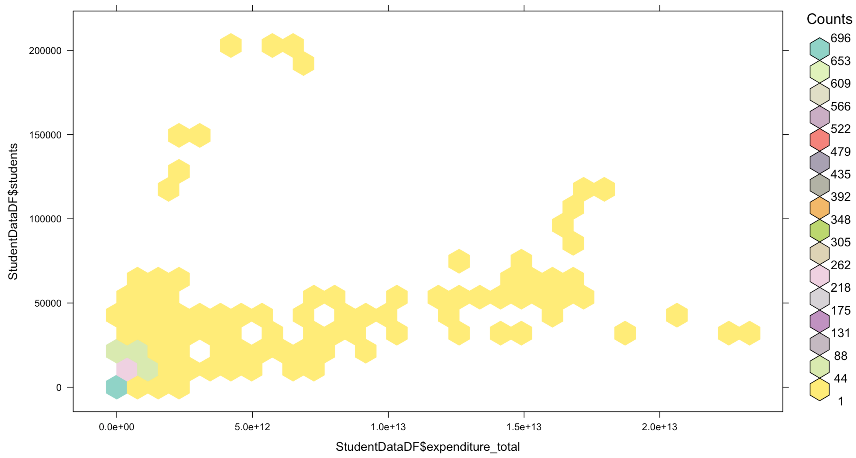


Figure 28

Above are hexbin plots which categorize on basis of colors and give an accurate result. I plotted number of students w.r.t GDP and expenditure respectively and the graphs above demonstrate that.As expected higher concentration of students going abroad lying in countries with lower gdp and lower expenditure on education.

The graph below shows variation of number of students w.r.t years with the mean plotted for all years showing it has almost been the same over the years[2].

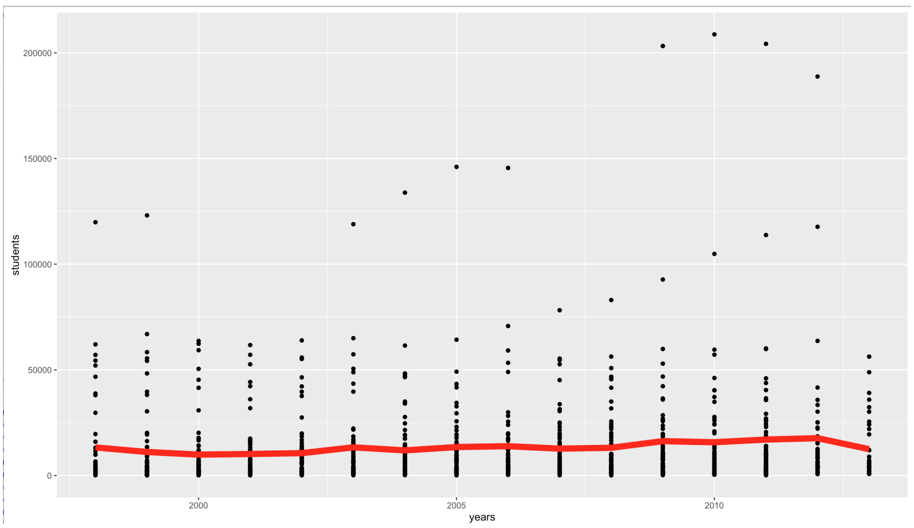


Figure 29

7 Conclusion

In this way I used UNESCO datasets for number of students going abroad, GDP of countries and expenditure of countries on education as percentage of their GDP for doing Map Reduce analysis in Apache Pig using Cloudera Hadoop environment and also managed to analyze trends of these factors using supervised Machine Learning in R and did some interesting information visualization as well.

References

[1] Melissa Banks and Rajika Bhandari. Global student mobility. *The Sage handbook of international higher education*, pages 379–397, 2012.

[2] Daniel A Keim. Information visualization and visual data mining. *IEEE transactions on Visualization and Computer Graphics*, 8(1):1–8, 2002.

[3] Ron Kohavi et al. A study of cross-validation and bootstrap for accuracy estimation and model selection. In *Ijcai*, volume 14, pages 1137–1145, 1995.

[4] Frederick Mosteller and John Wilder Tukey. Data analysis and regression: a second course in statistics. *Addison-Wesley Series in Behavioral Science: Quantitative Methods*, 1977.

[5] Colin Ware. *Information visualization: perception for design*. Elsevier, 2012.