



Integrating BI tools with R and Python

INTEGRATING TABLEAU AND MICROSOFT POWER BI WITH R AND PYTHON,
DR. PETER BRENNAN, B.SC, H.DIP COMP.SCI, M.SC, PH.D, brennap3@yahoo.ie

Aim of the talk

- ▶ To outline (as I see it), why you would integrate R and Python with Tableau and Power BI.
- ▶ To illustrate how to integrate R and Python with Power BI and Tableau respectively.
- ▶ Using Gapminder and NVD (national vulnerability data) I will show how to take advantage of either Python and R to enhance data, process it and create analytical visualizations not easy to create in these tools.
- ▶ Why is this important? BI and analytics (stats, machine learning, data engineering) is merging into a single field.
- ▶ When it comes to analysis I am complete ~~where~~ where social butterfly and I will use whatever is easy to use and cheap or better still free, a nicer way of putting this is we should try and be tool agnostic for benefits stated above.

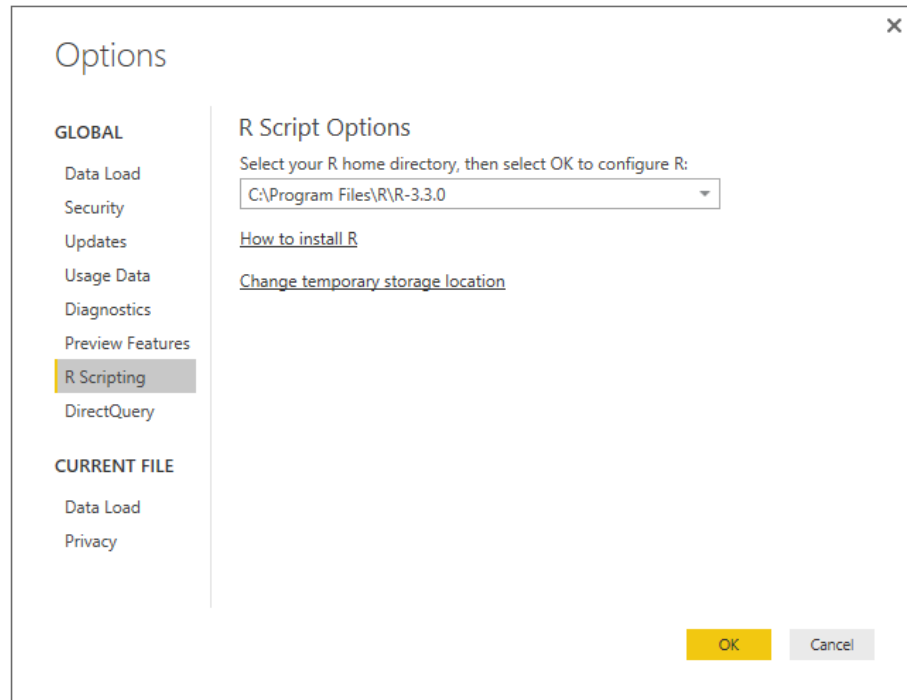
Why is it useful to integrate R and Python with BI tools

- ▶ Micro ETL's, R and Python let you merge data from multiple data sets clean it, process it to far more an extent then Tableau or Power BI can do.
- ▶ This allows you to bi-pass traditional data engineering functions to get access to the data, enables autonomous explorative experimentation.
- ▶ Process the data, we will use PCA (Principal Component Analysis) and CA (correspondence analysis) to carry out dimension reduction on both numerical and categorical data.
- ▶ We will then use this data to create some custom visualizations of the data.
- ▶ We will also showcase some useful visualizations that may not be available to us in either Tableau or Microsoft Power BI

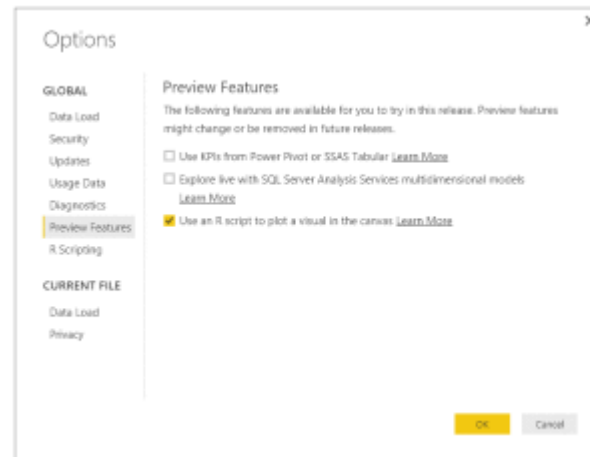
Setting up R with Microsoft Power BI (Windows 10)

- ▶ Follow this tutorial: <https://powerbi.microsoft.com/en-us/documentation/powerbi-desktop-r-visuals/>
- ▶ It involves the following steps.
 1. Install R (go ahead and install R-studio while you are at it)
 2. Install Power BI
 3. Go to File > Options and settings > Options and point R scripting to your R installation.
 4. Enable preview feature tab
- ▶ Don't know if this is available on Linux (maybe you can run it under Wine)
, it's available on Mac and all free for personnel use and enterprise licenses are really cheap.

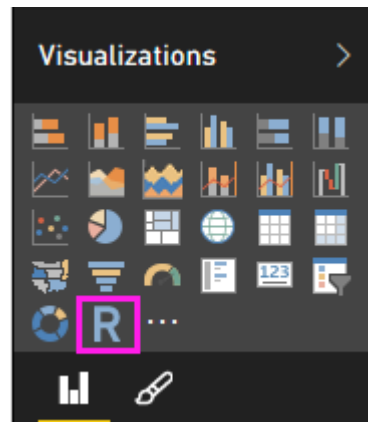
Setting up R with Microsoft Power BI (Windows 10)



Enable preview feature tab



R ICON is available



Create Your dataset

- ▶ We will use the following script to do the following.
- ▶ The script joins data from a number of disparate sources:
 - ▶ Vfeed (A copy of NISTS NVD, National Vulnerability Database accessed via executing a Python script which builds a Sqlite database holding the most update copy of NISTS VFEED)
 - ▶ exploitdb (Blackducks online repository)
- ▶ We will build our micro-etl script as an R-markdown file, this allows the script to be annotated clearly and makes hand off support very easy.

100



Power BI

File

Home

Transform

Add Column

View

Group By

Use First Row As Headers

Count Rows

Table

Transpose

Reverse Rows

Data Type: Text

Detect Data Type

Rename

Replace Values

Replace Errors

Fill

Pivot Column

Unpivot Columns

Move

Split Column

Format

Parse

Merge Columns

Extract

Statistics

Standard

Scientific

Information

Trigonometry

Rounding

Date

Time

Duration

Expand

Aggregate

Run R Script

Queries [2]

exploitsca

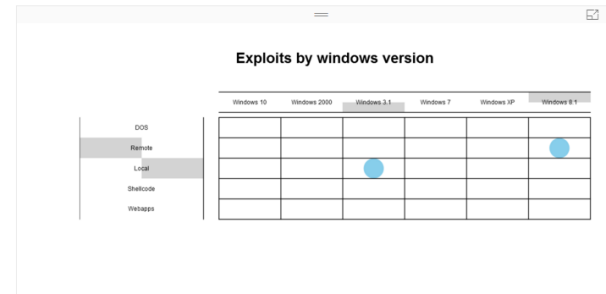
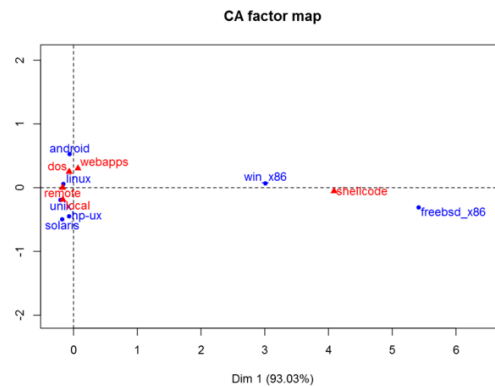
databasedesktopos

	A ₁	A ₂ count_REMOTE_EXPLOIT_T...	A ₃ count_LOCAL_EXPLOIT_T...	A ₄ count_WEBAPPS_EXPLOIT_T...	A ₅ count_DOS_EXPLOIT_T...	A ₆ count_SHELLCODE_EXPLOIT_T...
1	Mac OS X 10.5	0	0	0	3	0
2	Mac OS X 10.6	0	0	0	11	0
3	Mac OS X 10.7	0	0	0	2	0
4	Mac OS X 10.9	0	0	0	2	0
5	Windows 3.1	0	1	0	0	0
6	Mac OS X 10.10	1	1	0	0	0
7	Windows 8.1	1	0	0	0	0
8	Cassandra	0	0	0	0	0
9	DB2	0	0	0	0	0
10	Elastic	0	0	0	0	0
11	Filemaker	0	0	0	0	0
12	Hbase	0	0	0	0	0
13	Mac OS X 10.11	0	0	0	0	0
14	Mac OS X 10.8	0	0	0	0	0
15	Microsoft Access	0	0	0	0	0
16	Microsoft SQL Server	0	0	0	0	0
17	MongoDB	0	0	0	0	0
18	MySQL	0	0	0	0	0
19	Oracle	0	0	0	0	0
20	PostgreSQL	0	0	0	0	0
21	Redis	0	0	0	0	0
22	Solr	0	0	0	0	0
23	SQLite	0	0	0	0	0
24	Teradata	0	0	0	0	0
25	Windows 10	0	0	0	0	0
26	Windows 2000	0	0	0	0	0
27	Windows 7	0	0	0	0	0
28	Windows XP	0	0	0	0	0

Querying a dataset

Queries [2]							
		A ^B _C Software	A ^B _C Remote	A ^B _C Local	A ^B _C Webapps	A ^B _C DOS	A ^B _C Shellcode
exploitsca		1 Windows 3.1	0	1	0	0	0
databasedesktopos		2 Windows 8.1	1	0	0	0	0
		3 Windows 10	0	0	0	0	0
		4 Windows 2000	0	0	0	0	0
		5 Windows 7	0	0	0	0	0
		6 Windows XP	0	0	0	0	0

Exploits Analytical Dashboard built in Power BI with R's Help



Exploits by windows Version

DOS	Local	Remote	Shellcode	Software	Webapps
0	0	0	0	Windows 10	0
0	0	0	0	Windows 2000	0
0	0	0	0	Windows 7	0
0	0	0	0	Windows XP	0
0	0	1	0	Windows 8.1	0
0	1	0	0	Windows 3.1	0

Now for Python and Tableau

- ▶ So before getting started lets look at our setup:
 - ▶ Curtis Harris's blog does a pretty good job here:
 - ▶ Some requisites
 - ▶ Python 2.7.x save yourself a lot of messing and install Anaconda, you get all python packages you need (scikit, pandas, numpy, seaborn etc)
 - ▶ Download the Python extract API.
 - ▶ And probably install Tableau (the nice people at tableau will give you a copy of this for free) as that will be quite useful.

Python and tableau –looking at democracy around the world

- ▶ This use case involves downloading data from Gapminder and transparency international data to build a political and economic freedom index.
- ▶ We will carry out cluster a k-means cluster analysis of the data and create a dashboard showing choropleth and the results of PCA and clustering to get a better understanding of what makes countries democratic or undemocratic
- ▶ We will use the Tableau Data Extract, to achieve this

```
##data['Years_In_Nato'] = data.apply (lambda row: AGE_YEARS (row),axis=1)
data2extract['polityscore_cat_alph'] = data2extract.apply (lambda row: polityscore_cat (row),axis=1)

##

data2extract.head()

##

##

try:
    tdefile=tde.Extract('Extractp.tde')
    tableDef=tde.TableDefinition()
    tableDef.addColumn("country",tde.Types.Type.CHAR_STRING)
    tableDef.addColumn("polityscore",tde.Types.Type.DOUBLE)
    tableDef.addColumn("ploityscore_cat_alpha",tde.Types.Type.CHAR_STRING)
    table = tdefile.addTable('Extract',tableDef)
except:
    os.remove('Extractp.tde')
    tdefile=tde.Extract('Extractp.tde')

#step 2 create tabledef

##create the table in the image of the tableDef

##

newrow = tde.Row(tableDef)

##
for i in range(1,(len(data2extract.index)-1)):
    newrow.setCharString(0,data2extract.iloc[i,0])
    newrow.setDouble(1,data2extract.iloc[i,11])
    newrow.setCharString(2,data2extract.iloc[i,13])
    table.insert(newrow)

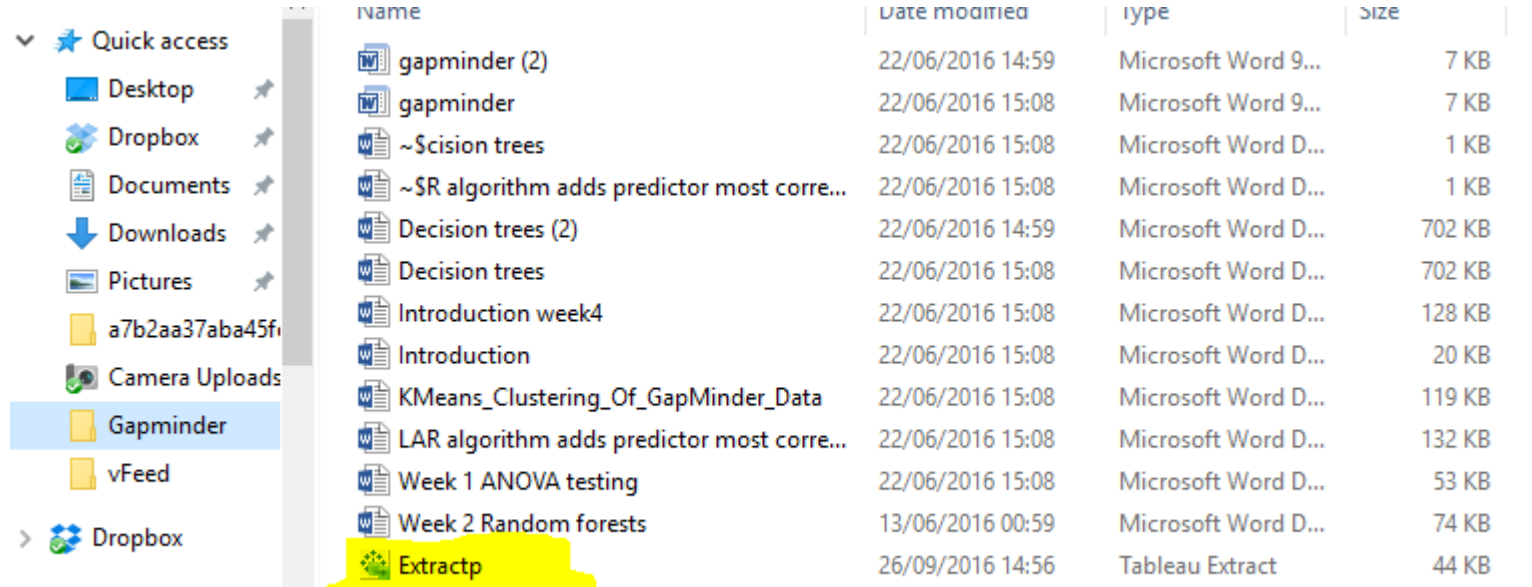
##close the file
tdefile.close()

## ----
```

Code to create an extract

- ▶ Python Code
 - ▶ Try to create an extract and defines it definition
 - ▶ Add the appropriate columns to the extract
 - ▶ Loop over the dataframe adding the appropriate cell

What you get ? Tableau dataset

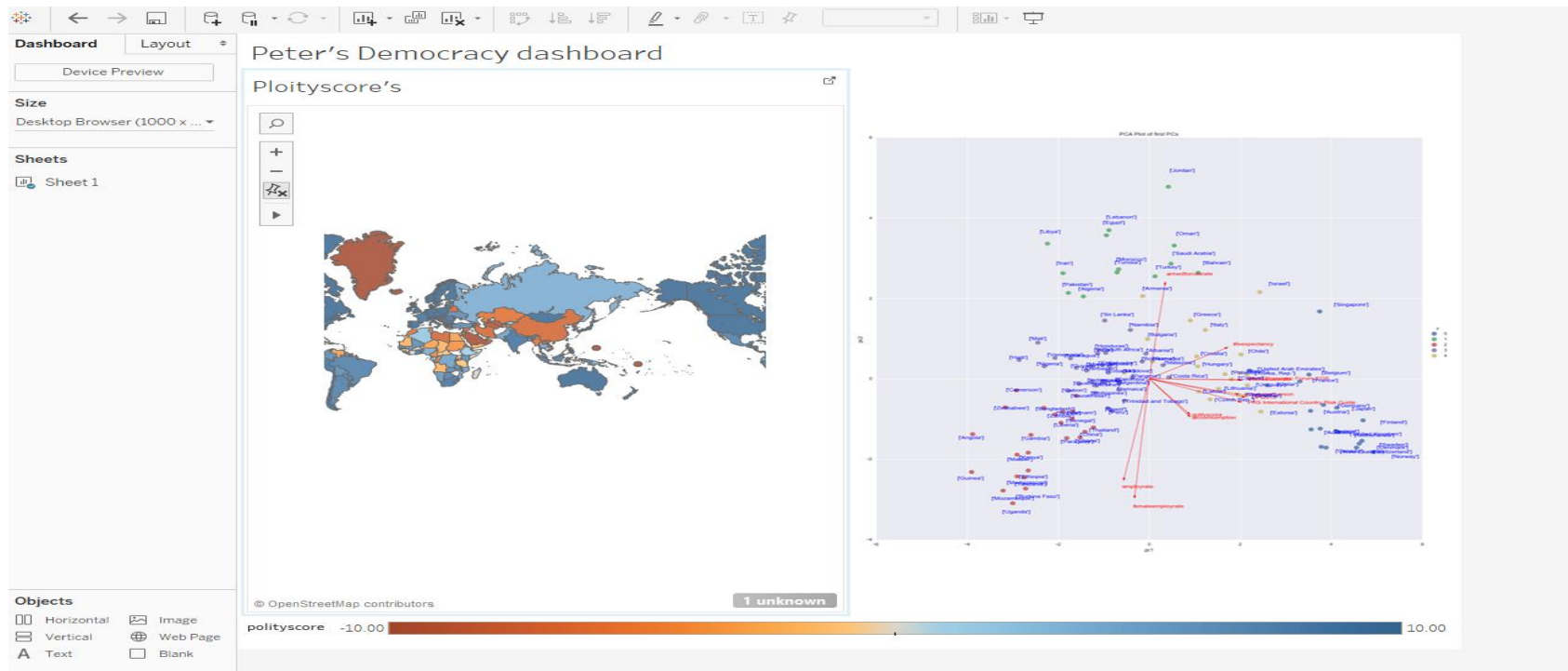


Name	Date modified	Type	Size
gapminder (2)	22/06/2016 14:59	Microsoft Word 9...	7 KB
gapminder	22/06/2016 15:08	Microsoft Word 9...	7 KB
~Scision trees	22/06/2016 15:08	Microsoft Word D...	1 KB
~SR algorithm adds predictor most corre...	22/06/2016 15:08	Microsoft Word D...	1 KB
Decision trees (2)	22/06/2016 14:59	Microsoft Word D...	702 KB
Decision trees	22/06/2016 15:08	Microsoft Word D...	702 KB
Introduction week4	22/06/2016 15:08	Microsoft Word D...	128 KB
Introduction	22/06/2016 15:08	Microsoft Word D...	20 KB
KMeans_Clustering_Of_GapMinder_Data	22/06/2016 15:08	Microsoft Word D...	119 KB
LAR algorithm adds predictor most corre...	22/06/2016 15:08	Microsoft Word D...	132 KB
Week 1 ANOVA testing	22/06/2016 15:08	Microsoft Word D...	53 KB
Week 2 Random forests	13/06/2016 00:59	Microsoft Word D...	74 KB
Extractp	26/09/2016 14:56	Tableau Extract	44 KB

So what's the benefit of this?

- ▶ You can provision (large) data rapidly from any source join it process and display it in Tableau.
- ▶ You can schedule the ETL's to run of a Jupyter (IPython) notebook giving large visibility to the data operations used to run the

Python derived dashboard in Tableau



Factor plot explained

- ▶ The factor plot shows the original dimensions overlayed on to the factor plot.
- ▶ Please follow my Tumblr blog for more detailed explanation (<http://brennap3.tumblr.com/>)
- ▶ We get to look at a number of dimensions from Gapminder and Transparency international and distill them down to two dimensions, overlay the original data dimensions and volia we can explain the correlations for the different groups.
- ▶ So for democracies, they start to break out into two tiers (tier one are categorized by low corruption, open business, median armed forces rates and employment rates), tier are less economically free, higher amounts of corruption and higher armed forces rates and lower employment rates.
- ▶ Why is this useful, can be used for a number of business use cases including, risk management, site strategy etc.

Conclusions

- ▶ Integration of Python and R into BI tools is straightforward.
- ▶ The utility of doing this is:
 - ▶ Allows you to provision data from complex data sources, clean it and join it, process it analytically and display it as part of BI dashboard.
 - ▶ Allows you to include visualization types which are not readily available in these tools such as factor plots of dimensions provided by dimension reduction techniques, in the case of numerical data (Principal Component Analysis, PCA) and categorical data (Correspondence analysis, CA).