Investigation into Human Trafficking in the UK

This report will focus on human trafficking within the UK. Analysis will be made using data that has been scrapped from National Crime Agency. I will seek to answer key statistical questions that have been formulated in consultations with an expert in the field.

Defining the Project Domain and Identifying Data Sources

Overview of the domain

I met with the founder of Red Light Campaign, a charity that helps human-trafficked victims. Key insights where gained about the nature of the domain and possible sources of data. I also found out what kinds of questions would be interesting to investigate.

The nature of the domain

Human Trafficking is the trading or movement of people for the purposes of forced labour, sexual slavery or exploitation and is currently a topical issue in the UK due to recently raised public awareness. Tackling this crime is considered a government priority and together with specialised charities, authorities seek to identify potential trafficked victims.

Once a potential victim of trafficking has been identified, specialist licensed agencies known as competent authorities (CAs) then refer the potential victim to the National Referral Mechanism (NRM) which is run by the National Crime Agency (NCA). The National Referral Mechanism (NRM) is a framework which seeks to identify victims of human trafficking or modern slavery to ensure they receive the appropriate support. The NRM is also the mechanism through which the Modern Slavery Human Trafficking Unit (MSHTU) collect data about victims. This information contributes to building a clearer picture about the scope of human trafficking and modern slavery in the UK. ("National Crime Agency - National Referral Mechanism," n.d.)

A person that is referred to the NRM is first assessed for 'reasonable grounds' status, i.e. does the assessor believe that they are victims, albeit without conclusive proof. This enables a victim to receive initial care and support. At a later stage, they are assessed for a 'conclusive decision' and if successful they may be granted discretionary leave to remain in the UK.

Data Sources

There are multiple sources that contain data for human trafficked victims including United Nations Reports. However as mentioned, I am choosing to focus on human trafficking within the UK, and particularly how it is handled by the police. Therefore, the most relevant data sets are the NRM data published by the National Crime Agency (NCA).

Problems

I was informed that although there is much to learn from the data recorded by the NCA, there is also many shortcomings in the level of detail that is provided. For example, criminal and labour trafficking numbers are placed in the same field in National Crime Agency data sets, making it hard for trafficking charities to gain a very detailed understanding of the nature of the domain. Additionally, the time it takes to gain a 'grounds' decisions have not been published.

Analytical questions

In consultation with the Red Light Campaign, I learned that a major area of interest is, how does the NCA handle human trafficking cases?
Typically, there are many decisions that constitute the NCA's handling of human trafficking cases.
This starts with the detection of victims, to victim's referral, treatment decisions, asylum decisions if necessary, decisions to prosecute the perpetrators.
What factors contribute to increased likelihood of achieving 'conclusive grounds' decisions for human trafficked victims?
How does geography, age, gender and language impact police outcomes as well as initial incidents?
If there are certain factors affecting a the 'grounds decisions', does this imply a bias?
Is there room to believe that external political events increase trafficking victims from specific countries?
How can this information then be used to improve outcomes who are hoping to achieve a successful grounds decision?
These are the analytical questions I sought to investigate using data.

Objectives

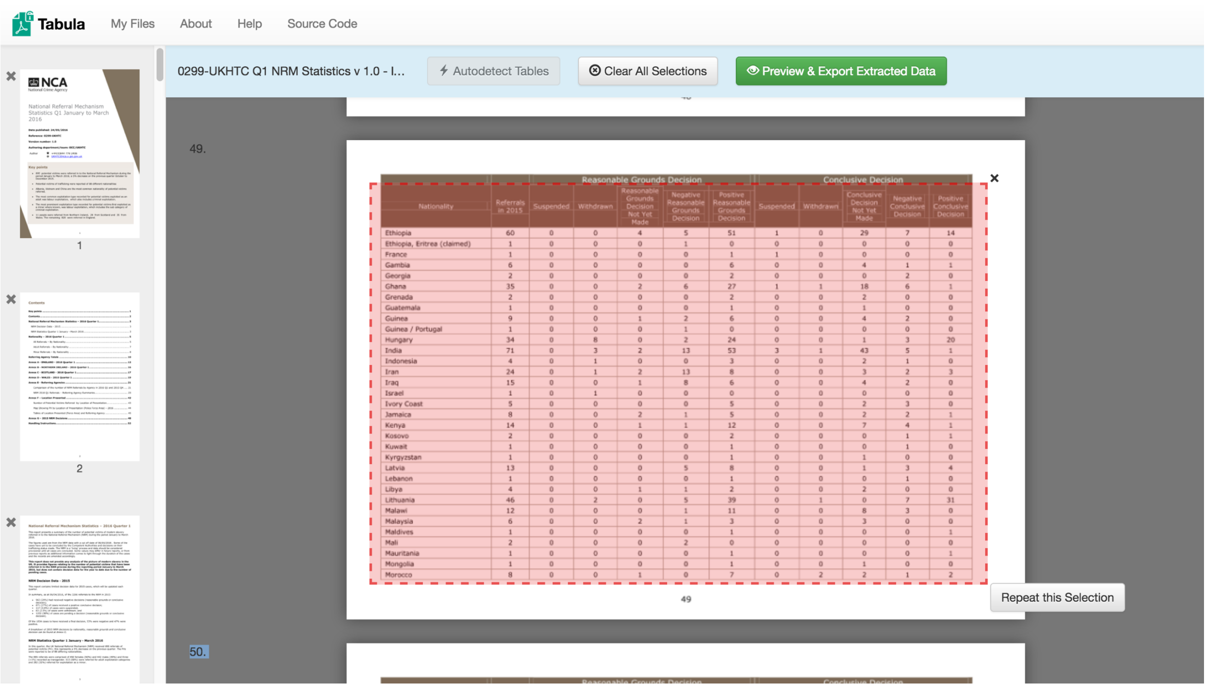
Once this data is investigated and analysed I hope to be able to contribute to the knowledge base that human trafficking victim's charities make use of. This will allow a greater understanding of where to focus their resources and a greater understanding of their own domain.
In particular, if there is found to be bias in the handling of 'grounds' decisions for people from specific countries, this can be addressed by government and NGOs.
Additionally, if there are external causative factors the cause human trafficking, maybe victims from specific countries of origin can be predicted and action can be taken via the international arena to counter this through police action and support.

Developing an Analysis Strategy

The first challenge arose when the NCA were unable to provide their data in a csv format. The only available sources of the data were in a pdf format. Therefore, the data needed to be scraped into a usable format.

Scraping Data from a pdf file

After investigating different data scraping solutions to accomplish this, which included manual selection and copying, pdftable, Tika and Tabula; Tabula ("Tabula," 2013) was chosen for its ease and effectiveness. Consequently, I was able to make a csv data file containing many of the data tables published by the NCA. This will be made public soon as a contribution to the research domain. See below for a screen shot of Tabula:



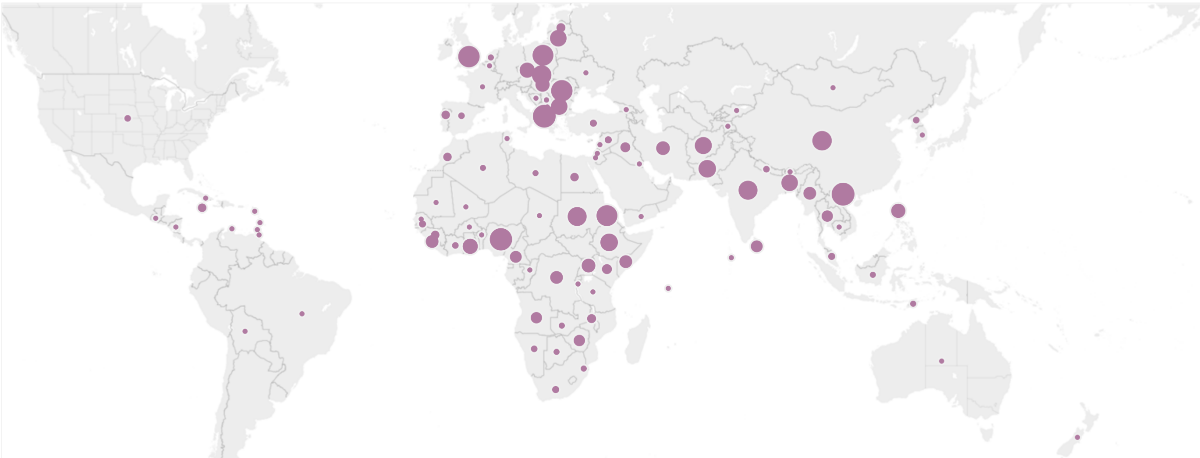
Initial Analysis

Since there was a variety of data gleaned from the NCA, it seems to sensible to do some exploratory data analysis.

I initially did this by observing the data in a tabular format. An initial visual observation of the data indicated that there is traffic victim's data categorised by country of origin and by type of trafficking, e.g. Domestic Servitude, Labour Exploitation. Additionally, there is data on the age and gender of the victim. However, there isn't any data containing tracking information of the timeline of how the cases are handled. There is useful data on "Grounds' Decisions such as NRM Decisions by Nationality, Reasonable Grounds (RG) and Conclusive Decision (CD). However, the data on 'grounds' decisions are only categorised by country and not by age, language or gender. Therefore, it seems the best initial course of action is to explore the 'grounds' decision data further. This data contains columns including Nationality, Referrals in 2015, and Positive Grounds Decision, see Appendix 3

I visualised referral data for 2015 by country and made an interactive dashboard using tableau.

Human Trafficking Referals in 2015 by Country of Origin



https://public.tableau.com/profile/marc.kendal#!/vizhome/HT1_0/Sheet2

I also visualised the data on a map using R as well with the same data. See appendix 1 for the code used.

This involved finding a world spatial map from http://www.naturalearthdata.com/. The Country label data was 'left joined' with the Nationality label data of trafficked victims. Then the map was plotted using the tmap package.

Performing the Analysis

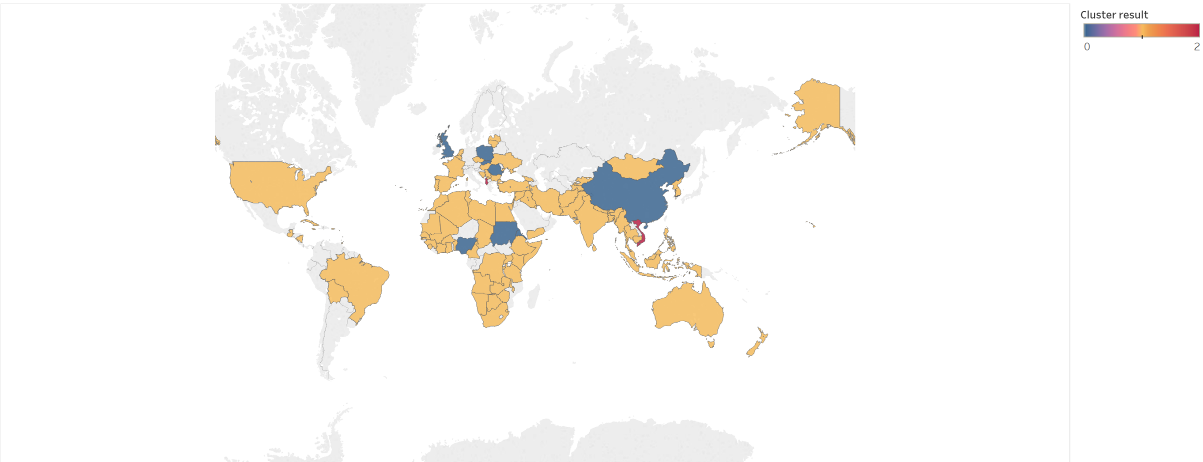
K-means clustering

To the answer the question of whether there is any inherent bias in 'grounds' decisions exist, which was one of the objectives, it made sense to use a k means algorithm to test if any meaningful clusters emerge that are different to the group as a whole. See Python code in Appendix 2.
After running the algorithm many times with different levels of k, there seemed to be one consistent cluster emerge.

This was that Albania and Vietnam are different from the other countries in the scale of potential victims from these origin countries. See Python code in Appendix 4
To visualise these results, I made a Pandas data frame and converted the results into a csv to visualise this data using tableau. See Python code in Appendix 5
The Country data was visualised by its K-means clustering group assignment. I left-joined the k-means results with the original 'Grounds' to generate the map below. By clicking on the dashboard link on can see more detailed information of each countries referrals and ground decisions data.

https://public.tableau.com/profile/marc.kendal#!/vizhome/KmeansClusterresults/Sheet1

K-means Clustering Results of Human Trafficking Origins recorded within UK 2015



Since it was clear that Albania and Vietnam were members of a unique and consistent occurring cluster, I tried to ascertain why this was. Straight away it was clear that the reason that Albania and Vietnam were grouped together was that they contained the largest number of trafficked victims for any country. Maybe the column containing the total number of referral victims was a leading the k-means algorithm to group clusters per the total number of referrals, by way of a compounding effect or it could just mean that there isn't evidence of any special treatment to a specific country and by implication all victims are treated equally irrespective of where they come from. So, in order test further I removed the Referral total column from the data set to be k-means clustered and ran the algorithm again to see if the clusters were still the same. In fact, the clustering results were still the same and based on this data and using a k-means algorithm I concluded that there isn't any evidence to suggest that there is a bias towards the handling of victims from specific countries.

I then thought of ways to explain the large numbers of victims from Albania and Vietnam. Was there a certain political event that happened or maybe other country indicators could help explain levels of human trafficking from origin countries. Maybe low income and unemployment in origin countries causes human trafficking?

Regression

I picked 3 relevant UN indicators: GDP per Capita 2015, Proportion of seats held by women in national parliaments (%) in 2015 and Unemployment, female (% of female labor force) in 2014 (2015 wasn't available) and joined that data with the existing 'grounds' data with the view to run an OLS regression and see if there were any significant coefficients. Unfortunately, recent general net migration data was not available to be tested as an independent variable. The data wrangling which was needed as a prerequisite to OLS analysis included: matching and renaming columns and merging csv files. See Python code in Appendix 6
After running an OLS multiple regression algorithm the following results were obtained: See Python code in Appendix 7

Params: OLS Regression Results
==

Dep. Variable:	y	R-squared:	0.105
Model:	OLS	Adj. R-squared:	0.068
Method:	Least Squares	F-statistic:	2.815
Date:	Wed, 14 Dec 2016	Prob (F-statistic):	0.0452

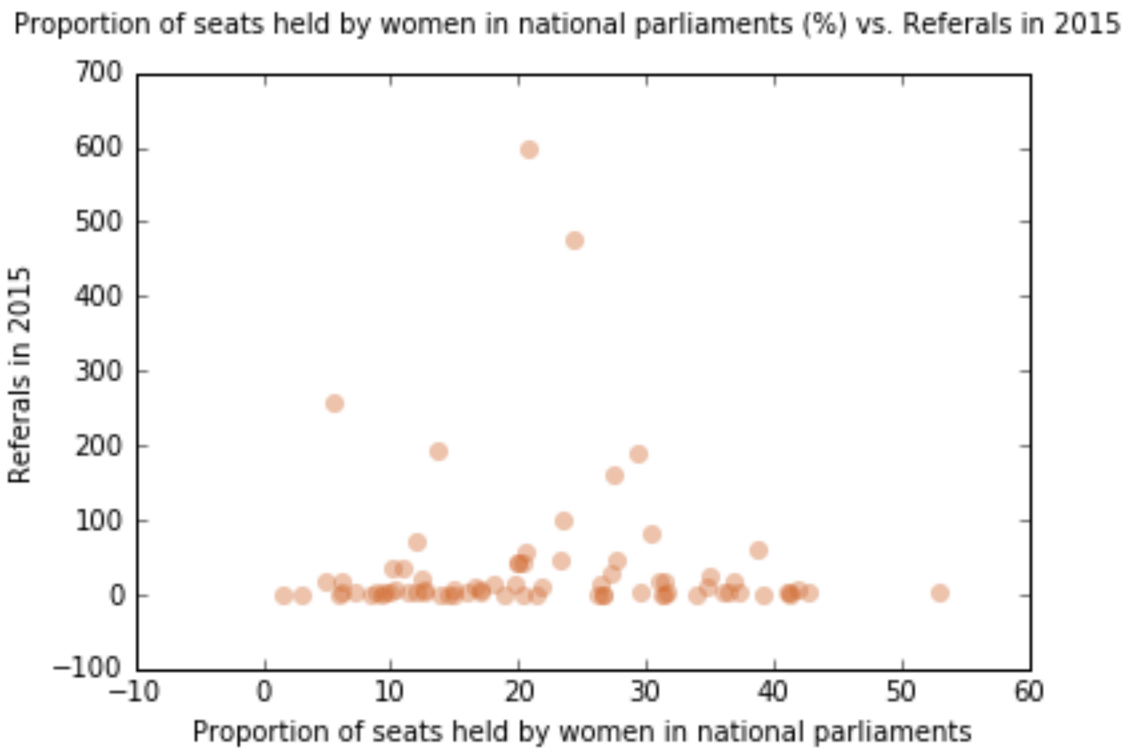
Time:	14:23:41	Log-Likelihood:	-450.03
No. Observations:	75	AIC:	906.1
Df Residuals:	72	BIC:	913.0
Df Model:	3		
Covariance Type:	nonrobust		

	coef	std err	t	P>|t|	[95.0% Conf. Int.]	
x1	0.0004	0.001	0.493	0.623	-0.001	0.002
x2	0.9947	0.765	1.300	0.198	-0.530	2.520
x3	0.3443	1.467	0.235	0.815	-2.580	3.268

Omnibus:	95.999	Durbin-Watson:	1.886
Prob(Omnibus):	0.000	Jarque-Bera (JB):	1246.477
Skew:	4.089	Prob(JB):	2.14e-271
Kurtosis:	21.221	Cond. No.	2.66e+03

There were three regressors that were tested, x1 : GDP per Capita 2015, x2: Proportion of seats held by women in national parliaments (%) in 2015 and x3: Unemployment, female (% of female labor force) in 2014. An interesting result to note is that regressor x2: Proportion of seats held by women in national parliaments (%) in 2015 has a high coefficient albeit with a high standard error. Additionally, the confidence interval is between -0.530 and 2.520 and therefore it wouldn't be correct to say that a positive or negative affect is expected. What can be said is based on this analysis, the estimated rate of change of the conditional mean of "Referals in 2015" (Y) with respect to " Proportion of seats held by women in national parliaments (%) in 2015 " (x2), when " GDP per Capita 2015" (x1) is fixed, and "Unemployment, female (% of female labor force) in 2014" (x3) is fixed is between -0.530 and 2.520 units.

However, when plotting and running correlation measures on variable x2 individually, the correlation values were low. See Python code in Appendix 7
Correlation Pearson: -0.0279147358285, 0.812086935757
Correlation Spearman: 0.0530301699146, 0.651370346067



It would be possible to conclude that the UN variables tested don't have strong explanatory power to explain why some countries to have many people being trafficked to the UK in 2015 for example Albania and Vietnam.

Reflection

The major conclusion is that there doesn't seem to be any bias as to how trafficked victims are handled from any specific country. This was based on using a k-means clustering algorithm. This result, of no bias found, is an insight of value to charities working with human trafficked victims as well as to the Home Office. Future work would include merging data sets to form a larger data set, using time series data and consequently utilising alternative clustering algorithms.
In conversation with the human trafficking charity, I was informed that there had in fact been natural disasters in Vietnam which could explain the large number of trafficked people from Vietnam.
Future work could include finding data on natural disasters in Vietnam and testing to see if they can explain the variability of victims from Vietnam over many years.
Indeed, in the initial investigation, it was noted that many countries of origin have large volatility in the number of victims, from year to year. Maybe it would be possible to create a general predictive model to estimate the likelihood of trafficked victims from a certain country by using news data.
Additionally, one could merge UK human trafficking data with UN human trafficking data to gain more confidence in any conclusion.

Furthermore, UN indicators were tested as explanatory variables to explain Human Trafficking Referrals in the UK. These included GDP, Proportion of seats held by women in national parliaments and female unemployment. There were mixed results which may be due to a small data set size. This small data set size could have caused the large confidence interval, which encompassed predictions of both positive and negative effects on Referrals based on changes to the UN indicators.
Future Work could include testing many more UN indicators and using a larger data set.

Bibliography

National Crime Agency - National Referral Mechanism [WWW Document], n.d. URL http://www.nationalcrimeagency.gov.uk/about-us/what-we-do/specialist-capabilities/uk-human-trafficking-centre/national-referral-mechanism (accessed 12.12.16).
Tabula: Extract Tables from PDFs [WWW Document], 2013. URL http://tabula.technology/ (accessed 12.12.16).

Supplementary Material

Appendix 1

```
library(rgdal)
htdata <- read.csv("Final_Decisions_2015_Next_Line.csv")
world <- readOGR(dsn = "ne_110m_admin_0_countries", layer = "ne_110m_admin_0_countries")
library(dplyr)
world@data <- left_join(world@data, htdata,  by=c("geounit" =  "Nationality"))
library(tmap)
tm_shape(world) +
  tm_fill(col="Referals.in.2015",style="cont",palette="BrBG", size=0.2, id="geo_label", title="") +
  tm_borders(col="#bdbdbd", lwd=0.5)
```

Appendix 2

```
# -*- coding: utf-8 -*-
"""
Created on Wed Dec  7 17:22:52 2016

@author: marckendal
"""

from pylab import plot,show
from numpy import vstack,array
from numpy.random import rand
from scipy.cluster.vq import kmeans,vq
from sklearn import datasets
import pandas as pd
from sklearn.decomposition import PCA
from sklearn.cluster import KMeans
import numpy as np

HTData = pd.read_csv('Final_Decisions_2015_Total_Removed.csv')
#print (HTData.head()) # check if everything is in place
HTData_Columns = len(HTData.columns)
print(HTData_Columns)
data = HTData.as_matrix()
#
classLabelsKnown = np.asarray(data[:,0])
##classLabelsKnown -= 1 # get the labels start from 0
#
##DataToCluster = HTData.as_matrix(['Referals in 2015','Positive Reasonable Grounds Decision'])
HTData_Minus_Countries = HTData.iloc[:,1:12]
DataToCluster = HTData_Minus_Countries.as_matrix()
#
print(DataToCluster)
# get all the data but not the class labels
#DataToCluster = data[:,1::]

#computing K-Means with K = 3 (3 clusters)
kmeansModel = KMeans(init='random', n_clusters=4, n_init=10)
kmeansModel.fit_predict(DataToCluster)
clusterResults = kmeansModel.labels_

## Let's check the results
for i, clustLabel in enumerate(clusterResults):
 print("Cluster result: ", clustLabel, " Known labels: ",classLabelsKnown[i])
```

Appendix 3

		Reasonable Grounds Decision					Conclusive Decision				
Nationality	Referals in 2015	Suspended	Withdrawn	Reasonable Grounds	Negative Reasona	Positive Reasonak	Suspended	Withdrawn	Conclusive Decis	Negative Conclu:	Positive Conclusive Decision
Afghanistan	48	2	0	1	18	27	2	0	15	8	2
Albania	600	2	2	20	36	540	3	0	329	145	63
Algeria	3	0	0	0	2	1	0	0	1	0	0
Angola	17	0	0	1	2	14	0	0	5	6	3
Australia	1	0	0	0	0	1	0	0	1	0	0
Bangladesh	44	0	0	1	6	37	0	1	29	6	1
Belgium	1	0	0	0	1	0	0	0	0	0	0
Benin	2	0	0	0	1	1	0	0	1	0	0
Bhutan	1	0	0	0	0	1	0	0	1	0	0
Bolivia	2	0	0	0	2	0	0	0	0	0	0
Bosnia and H	1	0	0	0	0	1	0	0	0	1	0
Botswana	4	0	0	0	1	3	0	0	3	0	0
Brazil	2	0	0	0	0	2	0	0	2	0	0
Bulgaria	44	1	0	0	4	39	0	1	0	7	31
Burkina Faso	1	0	0	0	0	1	0	0	1	0	0
Burma	20	0	0	0	4	16	0	0	7	7	2
Burundi	2	0	0	0	0	2	0	0	2	0	0
Cambodia	1	0	0	0	0	1	0	0	1	0	0
Cameroon	18	0	0	0	4	14	1	0	8	4	1
Chad	1	0	0	0	0	1	0	0	1	0	0
China	99	2	0	4	14	79	2	1	53	20	3
Congo	1	0	0	0	0	1	0	0	0	0	1
Cuba	2	0	0	1	0	1	0	0	0	1	0
Czech Repub	43	0	1	0	4	38	0	0	0	6	32
Democratic F	20	0	0	0	5	15	0	0	12	2	1

Appendix 4

K Means Clustering Results
Cluster result: 1 Known labels: Afghanistan
Cluster result: 0 Known labels: Albania
Cluster result: 1 Known labels: Algeria
Cluster result: 1 Known labels: Angola
Cluster result: 1 Known labels: Australia
Cluster result: 1 Known labels: Bangladesh
Cluster result: 1 Known labels: Belgium
Cluster result: 1 Known labels: Benin
Cluster result: 1 Known labels: Bhutan
Cluster result: 1 Known labels: Bolivia
Cluster result: 1 Known labels: Bosnia and Herzegovina
Cluster result: 1 Known labels: Botswana
Cluster result: 1 Known labels: Brazil
Cluster result: 1 Known labels: Bulgaria
Cluster result: 1 Known labels: Burkina Faso
Cluster result: 1 Known labels: Burma
Cluster result: 1 Known labels: Burundi
Cluster result: 1 Known labels: Cambodia
Cluster result: 1 Known labels: Cameroon
Cluster result: 1 Known labels: Chad
Cluster result: 2 Known labels: China
Cluster result: 1 Known labels: Congo
Cluster result: 1 Known labels: Cuba
Cluster result: 1 Known labels: Czech Republic
Cluster result: 1 Known labels: Democratic Republic of the Congo
Cluster result: 1 Known labels: Egypt
Cluster result: 1 Known labels: Egypt/Eritrean
Cluster result: 2 Known labels: Eritrea
Cluster result: 1 Known labels: Ethiopia
Cluster result: 1 Known labels: Ethiopia, Eritrea (claimed)
Cluster result: 1 Known labels: France
Cluster result: 1 Known labels: Gambia
Cluster result: 1 Known labels: Georgia
Cluster result: 1 Known labels: Ghana
Cluster result: 1 Known labels: Grenada
Cluster result: 1 Known labels: Guatemala
Cluster result: 1 Known labels: Guinea
Cluster result: 1 Known labels: Guinea / Portugal
Cluster result: 1 Known labels: Hungary
Cluster result: 1 Known labels: India
Cluster result: 1 Known labels: Indonesia
Cluster result: 1 Known labels: Iran
Cluster result: 1 Known labels: Iraq
Cluster result: 1 Known labels: Israel
Cluster result: 1 Known labels: Ivory Coast
Cluster result: 1 Known labels: Jamaica

```
Cluster result:  1  Known labels:  Kenya
Cluster result:  1  Known labels:  Kosovo
Cluster result:  1  Known labels:  Kuwait
Cluster result:  1  Known labels:  Kyrgyzstan
Cluster result:  1  Known labels:  Latvia
Cluster result:  1  Known labels:  Lebanon
Cluster result:  1  Known labels:  Libya
Cluster result:  1  Known labels:  Lithuania
Cluster result:  1  Known labels:  Malawi
Cluster result:  1  Known labels:  Malaysia
Cluster result:  1  Known labels:  Maldives
Cluster result:  1  Known labels:  Mali
Cluster result:  1  Known labels:  Mauritania
Cluster result:  1  Known labels:  Mongolia
Cluster result:  1  Known labels:  Morocco
Cluster result:  1  Known labels:  Namibia
Cluster result:  1  Known labels:  Nepal
Cluster result:  1  Known labels:  Netherlands
Cluster result:  1  Known labels:  Netherlands Antilles
Cluster result:  1  Known labels:  New Zealand
Cluster result:  1  Known labels:  Nicaragua
Cluster result:  2  Known labels:  Nigeria
Cluster result:  1  Known labels:  North Korea
Cluster result:  1  Known labels:  Pakistan
Cluster result:  1  Known labels:  Palestine
Cluster result:  1  Known labels:  Philippines
Cluster result:  2  Known labels:  Poland
Cluster result:  1  Known labels:  Portugal
Cluster result:  2  Known labels:  Romania
Cluster result:  1  Known labels:  Senegal
Cluster result:  1  Known labels:  Serbia
Cluster result:  1  Known labels:  Serbia/Kosovo
Cluster result:  1  Known labels:  Seychelles
Cluster result:  1  Known labels:  Sierra Leone
Cluster result:  2  Known labels:  Slovakia
Cluster result:  1  Known labels:  Somalia
Cluster result:  1  Known labels:  South Africa
Cluster result:  1  Known labels:  South Korea
Cluster result:  1  Known labels:  Spain
Cluster result:  1  Known labels:  Sri Lanka
Cluster result:  1  Known labels:  St Kitts & Nevis
Cluster result:  1  Known labels:  St Lucia
Cluster result:  2  Known labels:  Sudan
Cluster result:  1  Known labels:  Swaziland
Cluster result:  1  Known labels:  Syria
Cluster result:  1  Known labels:  Tajikistan
Cluster result:  1  Known labels:  Tanzania
Cluster result:  1  Known labels:  Thailand
Cluster result:  1  Known labels:  Timor - Leste
Cluster result:  1  Known labels:  Trinidad & Tobago
Cluster result:  1  Known labels:  Tunisia
Cluster result:  1  Known labels:  Turkey
Cluster result:  1  Known labels:  Uganda
Cluster result:  1  Known labels:  Ukraine
Cluster result:  2  Known labels:  United Kingdom
Cluster result:  1  Known labels:  United Kingdom/Vietnam
Cluster result:  1  Known labels:  USA
Cluster result:  0  Known labels:  Vietnam
Cluster result:  1  Known labels:  Yemen
Cluster result:  1  Known labels:  Zambia
Cluster result:  1  Known labels:  Zimbabwe
```

Appendix 5

```
d = []
for i, clustLabel in enumerate(clusterResults):
   d.append({'Cluster result': clustLabel, 'Known labels': classLabelsKnown[i], })

df_KMeans_Cluster = pd.DataFrame(d)
df_KMeans_Cluster.to_csv('Cluster_Results.csv')
```

Appendix 6

```
"""
Created on Thu Dec  8 15:34:48 2016

@author: marckendal
"""

# This code will merge UN indicator data with the existing  Human trafficking Data
import pandas as pd

#
#GDP Date MERGING
GDPData = pd.read_csv('GDP.csv', skiprows=4)
HTData = pd.read_csv('Final_Decisions_2015_Total_Removed.csv')
GDPData2015 = GDPData.loc[:,['Country Name','2015']]
# In order to merge files it is necessary to have the same common column names
GDPData2015_Rename = GDPData2015.rename(columns = {'2015':'GDP per Capita 2015'})
HTData_Column_Name_Change = HTData.rename(columns = {'Nationality':'Country Name'})
#Merging Data File
mergedGDP  = pd.merge(HTData_Column_Name_Change, GDPData2015_Rename, on='Country Name')

#GEN Data Merging
GENData = pd.read_csv('GEN.csv', skiprows=4)
GENData2015 = GENData.loc[:,['Country Name','2015']]
GENData2015_Rename = GENData2015.rename(columns = {'2015':'Proportion of seats held by women in national
parliaments (%)'})
mergedGEN  = pd.merge(mergedGDP, GENData2015_Rename, on='Country Name')


#UEM Data Merging
UEMData = pd.read_csv('UEM.csv', skiprows=4)
UEMData2015 = UEMData.loc[:,['Country Name','2014']]
UEMData2015_Rename = UEMData2015.rename(columns = {'2014':'Unemployment, female (% of female labor force)'})
mergedUEM  = pd.merge(mergedGEN, UEMData2015_Rename, on='Country Name')
print (mergedUEM .head())


mergedUEM.to_csv('UNMerged.csv',header=True, index=False)

print (mergedUEM .head())
```

Appendix 7

```
# -*- coding: utf-8 -*-
"""
Created on Wed Dec 14 13:13:02 2016

@author: marckendal
"""

import pandas as pd

mergedUEM = pd.read_csv('UNMerged.csv')
print(mergedUEM.head())

import scipy.stats  as sc
import matplotlib.pyplot as plt
```

```
# start with importing statsmodels
import statsmodels.api as sm

regressors = mergedUEM.as_matrix(['GDP per Capita 2015', 'Proportion of seats held by women in national parliaments
(%)', 'Unemployment, female (% of female labor force)'])

referrals_2015_matrix = mergedUEM.as_matrix(['Referals in 2015'])
referrals_2015 = referrals_2015_matrix[:,0]

# our dependent variable is Ref
# we use the OLS function from statsmodels
model = sm.OLS(referrals_2015,regressors)
results = model.fit()

# the summary function returns a very comprehensive report on the results
print ("Params: ", results.summary())




#gdp2015= matrix_array[:,0]
women_parliament = regressors[:,1]
unemployment =  regressors[:,2]
#
women_parliament_r, pvalue1 = sc.stats.pearsonr(women_parliament,referrals_2015)
wp_spearman_r, pvalue2 = sc.stats.spearmanr(women_parliament,referrals_2015)

print ("Correlation Pearson: ", women_parliament_r, pvalue1)
print("Correlation Spearman: ", wp_spearman_r, pvalue2)
##
##
plt.figure(1)
plt.suptitle('Proportion of seats held by women in national parliaments (%) vs. Referals in 2015')
plt.xlabel('Proportion of seats held by women in national parliaments')
plt.ylabel('Referals in 2015')
#plt.scatter(arr1, arr2 , c = "#D06B36", s = 50, alpha = 0.4, linewidth='0')
plt.scatter(women_parliament,referrals_2015, c = "#D06B36", s = 50, alpha = 0.4, linewidth='0')
```