Assignment 4

Machine Learning
MSc Business Analytics

Wolfram Wiesemann

1 Individual Assignment

Instructions: *This exercise should be done "by hand", that is, not using R or Python. All necessary calculations should be included in the submission, as well as brief explanations of what you do.*

Consider the following 7 two-dimensional observations:

	Observation i						
	$i = 1$	$i = 2$	$i = 3$	$i = 4$	$i = 5$	$i = 6$	$i = 7$
x_{i1}	1	1	1	5	2	6	4
x_{i2}	4	3	2	1	3	2	1

1. Plot the observations in a two-dimensional graph.
2. Perform K-means clustering with $K = 2$ using the Euclidean norm. Toss a coin 7 times to initialise the algorithm.
3. Cluster the data using hierarchical clustering with complete linkage and the Euclidean norm. Draw the resulting dendrogram.

2 Group Assignment

For this assignment, we are going to 'scrape' data on Welsh soccer players from the EA sports FIFA games website. It is important that you know how to scrape data from the web that is not available in a convenient form such as a CSV file.

1. Explore manually the website http://sofifa.com. Under the tab 'All', press on the any of the Welsh flags (e.g. those corresponding to G. Bale or A. Ramsey). Notice how the URL of the opened webpage changes to http://sofifa.com/players?na=50. Scrolling down, notice that not all players fit in one page. If you press 'Next', the new URL is http://sofifa.com/players?na=52&offset=80. Can you see the pattern? Next, select

an individual player and notice how the URL changes. We want to download the numerical attributes available for the first 640 Argentinian players (as appeared in the website.

2. Explain in detail the code below. In order to better understand the code, you may want to look at the following websites:

   - https://www.crummy.com/software/BeautifulSoup/
   - http://www.aivosto.com/vbtips/regex.html
   - https://docs.python.org/2/library/re.html

```
import pandas as pd
from bs4 import BeautifulSoup
import requests
import re
import unicodedata

attributes=['Crossing','Finishing','Heading␣accuracy',
 'Short␣passing','Volleys','Dribbling','Curve',
 'Free␣kick␣accuracy','Long␣passing','Ball␣control','Acceleration',
 'Sprint␣speed','Agility','Reactions','Balance',
 'Shot␣power','Jumping','Stamina','Strength',
 'Long␣shots','Aggression','Interceptions','Positioning',
 'Vision','Penalties','Composure','Marking',
 'Standing␣tackle','Sliding␣tackle','GK␣diving',
 'GK␣handling','GK␣kicking','GK␣positioning','GK␣reflexes']

links=[]  #get all argentinian players
for offset in ['0','80','160','240','320','400','480','560']:
    page=requests.get('http://sofifa.com/players?na=52&offset='+offset)
    soup=BeautifulSoup(page.content,'html.parser')
    for link in soup.find_all('a'):
        links.append(link.get('href'))
links=['http://sofifa.com'+l for l in links if 'player/'in l]

#pattern regular expression
pattern=r"""\s*([\w\s]*?)\s*FIFA"""  #file starts with empty spaces...
   players name...FIFA...other stuff
for attr in attributes:
    pattern+=r""".*?(\d*\s*"""+attr+r""")"""  #for each attribute we have
        other stuff..number..attribute..other stuff
pat=re.compile(pattern, re.DOTALL)  #parsing multiline text

rows=[]
links=links[10:]

for j,link in enumerate(links):
    print j,link
    row=[link]
    playerpage=requests.get(link)
    playersoup=BeautifulSoup(playerpage.content,'html.parser')
    text=playersoup.get_text()
    text=unicodedata.normalize('NFKD', text).encode('ascii','ignore')
    a=pat.match(text)
    row.append(a.group(1))
    for i in range(2,len(attributes)+2):
        row.append(int(a.group(i).split()[0]))
    rows.append(row)
    print row[1]
df=pd.DataFrame(rows,columns=['link','name']+attributes)
df.to_csv('ArgentinaPlayers.csv',index=False)
```

3. How would you change the code to download the first 480 English players instead?

4. Use the `sklearn.cluster.KMeans` Python class to cluster the players into 4 clusters.

5. By inspecting the clusters and looking up individual players online, try to assign meaningful labels to the clusters.

6. For a new and unknown player, the following attributes are available:

Crossing	45
Sprint Speed	40
Long Shots	35
Aggression	45
Marking	60
Finishing	40
GK_Handling	15

   For each of your 4 clusters from Step 4, compute the cluster centroid. Assign the new player to the nearest cluster based on the distance to the cluster centroids, using only the available attributes.