Machine learning: k-mean clusters

author: bhavesh patel

We will use Wikipedia data for famous people. The data set contains around 50,000 famous people. We will create a cluster for the group of people. We will identify who are similar to given person.

We will go through unsupervised learning, as we don't have a label for each cluster. That is, we don't have famous sports vs. politician vs. actor, etc.

Typically, we extract features from training data, which goes to machine learning model. Training data is also passed to machine learning algorithm. In this case, we are not passing this data. Hence it becomes unsupervised learning.

We will use TF-IDF document representation. At high level, same words appearing in one document e.g. the, a should have less weighting compare to word appearing across different documents.

The TF = Term Frequency is counted by number of times the word apperas.

IDF = Inverse Document Frequency. That uses log function for inverse frequency.

log (number of docs / 1 + number of docs using word).

larger number of doc with same word approaces to log 1 which is close to zero.

smaller number of doc with same word approaches to log (low value) which is larger value.

In [ ]:

In [1]:
```
import graphlab
```

In [2]:
```
# Limit number of worker processes. This preserves system memory, which prevents hosted notebook
graphlab.set_runtime_config('GRAPHLAB_DEFAULT_NUM_PYLAMBDA_WORKERS', 4)
```

```
[INFO] graphlab.cython.cy_server: GraphLab Create v2.1 started. Logging: /tmp/graphlab_server_1479752866.log

This non-commercial license of GraphLab Create for academic use is assigned to bhaveshhk8@gmail.com and will expire on October 17, 2017.
```

In [4]:
```
people = graphlab.SFrame('people_wiki.gl/')
```

In [6]:
```
people.head()
```

Out[6]:

URI	name	text
<http://dbpedia.org/resource/Digby_Morrell> ...	Digby Morrell	digby morrell born 10 october 1979 is a former ...
<http://dbpedia.org/resource/Alfred_J._Lewy> ...	Alfred J. Lewy	alfred j lewy aka sandy lewy graduated from ...
<http://dbpedia.org/resource/Harpdog_Brown> ...	Harpdog Brown	harpdog brown is a singer and harmonica player who ...
<http://dbpedia.org/resource/Franz_Rottensteiner> ...	Franz Rottensteiner	franz rottensteiner born in waidmannsfeld lower ...
<http://dbpedia.org/resou	G-Enka	henry krvits born 30

rce/G-Enka> ...	G-Enka	henry krvits born 30 december 1974 in tallinn ...
<http://dbpedia.org/resource/Sam_Henderson> ...	Sam Henderson	sam henderson born october 18 1969 is an ...
<http://dbpedia.org/resource/Aaron_LaCrate> ...	Aaron LaCrate	aaron lacrate is an american music producer ...
<http://dbpedia.org/resource/Trevor_Ferguson> ...	Trevor Ferguson	trevor ferguson aka john farrow born 11 november ...
<http://dbpedia.org/resource/Grant_Nelson> ...	Grant Nelson	grant nelson born 27 april 1971 in london ...
<http://dbpedia.org/resource/Cathy_Caruth> ...	Cathy Caruth	cathy caruth born 1955 is frank h t rhodes ...

[10 rows x 3 columns]

The data has the URL for each person, name of the person and text about the person. Let's retrive data for Obama.

In [10]:
```
people[people['name'] == 'Barack Obama']['text']
```

Out[10]:
```
dtype: str
Rows: ?
['barack hussein obama ii brk husen bm born august 4 1961 is the 44th and current president o
f the united states and the first african american to hold the office born in honolulu hawaii
 obama is a graduate of columbia university and harvard law school where he served as preside
nt of the harvard law review he was a community organizer in chicago before earning his law d
egree he worked as a civil rights attorney and taught constitutional law at the university of
 chicago law school from 1992 to 2004 he served three terms representing the 13th district in
 the illinois senate from 1997 to 2004 running unsuccessfully for the united states house of
 representatives in 2000in 2004 obama received national attention during his campaign to repr
esent illinois in the united states senate with his victory in the march democratic party pri
mary his keynote address at the democratic national convention in july and his election to th
e senate in november he began his presidential campaign in 2007 and after a close primary cam
paign against hillary rodham clinton in 2008 he won sufficient delegates in the democratic pa
rty primaries to receive the presidential nomination he then defeated republican nominee john
 mccain in the general election and was inaugurated as president on january 20 2009 nine mont
hs after his election obama was named the 2009 nobel peace prize laureateduring his first two
 years in office obama signed into law economic stimulus legislation in response to the great
 recession in the form of the american recovery and reinvestment act of 2009 and the tax reli
ef unemployment insurance reauthorization and job creation act of 2010 other major domestic i
nitiatives in his first term included the patient protection and affordable care act often re
ferred to as obamacare the doddfrank wall street reform and consumer protection act and the d
ont ask dont tell repeal act of 2010 in foreign policy obama ended us military involvement in
 the iraq war increased us troop levels in afghanistan signed the new start arms control trea
ty with russia ordered us military involvement in libya and ordered the military operation th
at resulted in the death of osama bin laden in january 2011 the republicans regained control
 of the house of representatives as the democratic party lost a total of 63 seats and after a
 lengthy debate over federal spending and whether or not to raise the nations debt limit obam
a signed the budget control act of 2011 and the american taxpayer relief act of 2012obama was
 reelected president in november 2012 defeating republican nominee mitt romney and was sworn
 in for a second term on january 20 2013 during his second term obama has promoted domestic p
olicies related to gun control in response to the sandy hook elementary school shooting and h
as called for full equality for lgbt americans while his administration has filed briefs whic
h urged the supreme court to strike down the defense of marriage act of 1996 and californias
 proposition 8 as unconstitutional in foreign policy obama ordered us military involvement in
 iraq in response to gains made by the islamic state in iraq after the 2011 withdrawal from i
raq continued the process of ending us combat operations in afghanistan and has sought to nor
malize us relations with cuba', ... ]
```

In [11]:
```
# Let's find out how big is out database.

len(people)
```

Out[11]:
```
59071
```

In [24]:
```
# now let's explore words used in obama.  we need to find out frequencye of the words.
# that will be used in TD-IDF algorithm.
# nv for name value pair.

obama = people[people['name'] == 'Barack Obama']
obama['word_nv'] = graphlab.text_analytics.count_words(obama['text'])
```

In [25]:
```
print obama['word_nv']
```

```
[{'operations': 1, 'represent': 1, 'office': 2, 'unemployment': 1, 'is': 2, 'doddfrank': 1,
 'over': 1, 'unconstitutional': 1, 'domestic': 2, 'named': 1, 'ending': 1, 'ended': 1, 'propo
sition': 1, 'seats': 1, 'graduate': 1, 'worked': 1, 'before': 1, 'death': 1, '20': 2, 'taxpay
er': 1, 'inaugurated': 1, 'obamacare': 1, 'civil': 1, 'mccain': 1, 'to': 14, '4': 1, 'polic
y': 2, '8': 1, 'has': 4, '2011': 3, '2010': 2, '2013': 1, '2012': 1, 'bin': 1, 'then': 1, 'hi
s': 11, 'march': 1, 'gains': 1, 'cuba': 1, 'californias': 1, '1992': 1, 'new': 1, 'not': 1,
 'during': 2, 'years': 1, 'continued': 1, 'presidential': 2, 'husen': 1, 'osama': 1, 'term':
 3, 'equality': 1, 'prize': 1, 'lost': 1, 'stimulus': 1, 'january': 3, 'university': 2, 'righ
ts': 1, 'gun': 1, 'republican': 2, 'rodham': 1, 'troop': 1, 'withdrawal': 1, 'involvement':
 3, 'response': 3, 'where': 1, 'referred': 1, 'affordable': 1, 'attorney': 1, 'school': 3, 's
enate': 3, 'house': 2, 'national': 2, 'creation': 1, 'related': 1, 'hawaii': 1, 'born': 2, 's
econd': 2, 'street': 1, 'election': 3, 'close': 1, 'operation': 1, 'insurance': 1, 'sandy':
 1, 'afghanistan': 2, 'initiatives': 1, 'for': 4, 'reform': 1, 'federal': 1, 'review': 1, 're
presentatives': 2, 'debate': 1, 'current': 1, 'state': 1, 'won': 1, 'marriage': 1, 'victory':
 1, 'unsuccessfully': 1, 'reauthorization': 1, 'keynote': 1, 'full': 1, 'patient': 1, 'augus
t': 1, 'degree': 1, '44th': 1, 'bm': 1, 'mitt': 1, 'attention': 1, 'delegates': 1, 'lgbt': 1,
 'job': 1, 'protection': 2, 'address': 1, 'ask': 1, 'november': 2, 'debt': 1, 'by': 1, 'car
e': 1, 'on': 2, 'great': 1, 'defense': 1, 'signed': 3, 'libya': 1, 'receive': 1, 'of': 18, 'm
onths': 1, 'against': 1, 'foreign': 2, 'spending': 1, 'american': 3, 'harvard': 2, 'act': 8,
 'military': 4, 'hussein': 1, 'or': 1, 'first': 3, 'and': 21, 'major': 1, 'clinton': 1, '199
7': 1, 'campaign': 3, 'russia': 1, 'wall': 1, 'legislation': 1, 'into': 1, 'primary': 2, 'com
munity': 1, 'three': 1, 'down': 1, 'hook': 1, 'ii': 1, '63': 1, 'americans': 1, 'elementary':
 1, 'total': 1, 'earning': 1, 'often': 1, 'barack': 1, 'law': 6, 'from': 3, 'raise': 1, 'dist
rict': 1, 'representing': 1, 'nine': 1, 'reinvestment': 1, 'arms': 1, 'relations': 1, 'nobe
l': 1, 'start': 1, 'dont': 2, 'tell': 1, 'iraq': 4, 'convention': 1, 'strike': 1, 'served':
 2, 'john': 1, 'was': 5, 'war': 1, 'form': 1, 'that': 1, 'tax': 1, 'sufficient': 1, 'republic
ans': 1, 'resulted': 1, 'hillary': 1, 'taught': 1, 'honolulu': 1, 'filed': 1, 'regained': 1,
 'july': 1, 'hold': 1, 'with': 3, 'he': 7, '13th': 1, 'made': 1, 'brk': 1, '1996': 1, 'whethe
r': 1, 'reelected': 1, 'budget': 1, 'us': 6, 'nations': 1, 'recession': 1, 'while': 1, 'econo
mic': 1, 'limit': 1, 'policies': 1, 'promoted': 1, 'called': 1, 'at': 2, 'control': 4, 'supre
me': 1, 'ordered': 3, 'nominee': 2, 'process': 1, '2000in': 1, '2012obama': 1, 'received': 1,
 'romney': 1, 'briefs': 1, 'defeated': 1, 'general': 1, 'states': 3, 'as': 6, 'urged': 1, 'i
n': 30, 'sought': 1, 'organizer': 1, 'shooting': 1, 'increased': 1, 'normalize': 1, 'length
y': 1, 'united': 3, 'court': 1, 'recovery': 1, 'laden': 1, 'laureateduring': 1, 'peace': 1,
 'administration': 1, '1961': 1, 'illinois': 2, 'other': 1, 'which': 1, 'party': 3, 'primarie
s': 1, 'sworn': 1, '2007': 1, 'obama': 9, 'columbia': 1, 'combat': 1, 'after': 4, 'islamic':
 1, 'running': 1, 'levels': 1, 'two': 1, 'included': 1, 'president': 4, 'repeal': 1, 'nominat
ion': 1, 'the': 40, 'a': 7, '2009': 3, 'chicago': 2, 'constitutional': 1, 'defeating': 1, 'tr
eaty': 1, 'relief': 2, '2004': 3, 'african': 1, '2008': 1, 'democratic': 4, 'consumer': 1, 'b
egan': 1, 'terms': 1}]
```

In [ ]:
```
```

In [26]:
```
# let's sort it to make it easier.  There is stack function in graph lab
# to view data side by side like a table.

obama_word_table = obama[['word_nv']].stack('word_nv', new_column_name=['word','count'])
```

In [27]:
```
obama_word_table.head()
```

Out[27]:

word	count
cuba	1
relations	1
sought	1
combat	1
ending	1
withdrawal	1
state	1
islamic	1
by	1
gains	1

[10 rows x 2 columns]

In [28]:
```
obama_word_table.sort('count', ascending=False)
```

Out[28]:

word	count

the	40
in	30
and	21
of	18
to	14
his	11
obama	9
act	8
he	7
a	7

[273 rows x 2 columns]
Note: Only the head of the SFrame is printed.
You can use print_rows(num_rows=m, num_columns=n) to print more rows and columns.

In [29]:
```
# to find the cluster and K-nearest-neighbours, we need to calculate data for all people.
# nv = name value pair.

people['word_nv'] = graphlab.text_analytics.count_words(people['text'])
```

In [30]:
```
people.head()
```

Out[30]:

<http://dbpedia.org/resource/Alfred_J._Lewy> ...	Alfred J. Lewy	alfred j lewy aka sandy lewy graduated from ...	{'precise': 1, 'thomas': 1, 'closely': 1, ...
<http://dbpedia.org/resource/Harpdog_Brown> ...	Harpdog Brown	harpdog brown is a singer and harmonica player who ...	{'just': 1, 'issued': 1, 'mainly': 1, 'nominat ...
<http://dbpedia.org/resource/Franz_Rottensteiner> ...	Franz Rottensteiner	franz rottensteiner born in waidmannsfeld lower ...	{'all': 1, 'bauforschung': 1, ...
<http://dbpedia.org/resource/G-Enka> ...	G-Enka	henry krvits born 30 december 1974 in tallinn ...	{'they': 1, 'gangstergenka': 1, ...
<http://dbpedia.org/resource/Sam_Henderson> ...	Sam Henderson	sam henderson born october 18 1969 is an ...	{'currently': 1, 'less': 1, 'being': 1, ...
<http://dbpedia.org/resource/Aaron_LaCrate> ...	Aaron LaCrate	aaron lacrate is an american music producer ...	{'exclusive': 2, 'producer': 1, 'show' ...
<http://dbpedia.org/resource/Trevor_Ferguson> ...	Trevor Ferguson	trevor ferguson aka john farrow born 11 november ...	{'taxi': 1, 'salon': 1, 'gangs': 1, 'being': 1, ...
<http://dbpedia.org/resource/Grant_Nelson> ...	Grant Nelson	grant nelson born 27 april 1971 in london ...	{'houston': 1, 'frankie': 1, 'labels': 1, ...
<http://dbpedia.org/resource/Cathy_Caruth> ...	Cathy Caruth	cathy caruth born 1955 is frank h t rhodes ...	{'phenomenon': 1, 'deborash': 1, 'both' ...

word_nv
{'selection': 1, 'carltons': 1, 'being': ...
{'precise': 1, 'thomas': 1, 'closely': 1, ...
{'just': 1, 'issued': 1, 'mainly': 1, 'nominat ...
{'all': 1, 'bauforschung': 1, ...
{'they': 1, 'gangstergenka': 1, ...
{'currently': 1, 'less': 1, 'being': 1, ...
{'exclusive': 2, 'producer': 1, 'show' ...

{'taxi': 1, 'salon': 1, 'gangs': 1, 'being': 1, ...
{'houston': 1, 'frankie': 1, 'labels': 1, ...
{'phenomenon': 1, 'deborash': 1, 'both' ...

[10 rows x 5 columns]

In [31]:
```
# now compute TF-IDF for each person.

tfidf_for_wiki_people = graphlab.text_analytics.tf_idf(people['word_nv'])
```

In [33]:
```
tfidf_for_wiki_people.head()
```

Out[33]:
```
dtype: dict
Rows: 10
[{'selection': 3.836578553093086, 'carltons': 7.0744723837970485, 'being': 1.7938099524877322, '2005': 1.6425861253275964, 'coach': 5.444264118987054, 'its': 1.6875948402695313, 'before': 2.9935647453367427, '21': 2.797250863489293, 'northern': 3.310021742836038, 'bullants': 7.489987827758714, 'to': 0.23472468840899618, 'perth': 5.051601193605607, 'sydney': 3.5981675296480873, '2014': 2.2073995783446634, 'has': 0.428497539744039, '2011': 1.7023470901042916, '2013': 1.9545642372230505, 'division': 2.7906099979103978, 'his': 0.7878343656409721, 'rules': 3.8272034844276295, 'assistant': 2.5220702633476124, 'spanned': 5.531174273867493, 'early': 1.929422753652229, 'game': 2.4168995190159084, 'five': 2.2137301792754096, 'during': 1.3174651479035495, 'continued': 2.720588055069447, '44game': 9.887883100557085, 'kangaroos': 20.726873835958425, 'twice': 3.3301582227950113, 'round': 2.897933583948961, 'the': 0.0027426017494956603, 'parade': 5.510031837293684, 'born': 0.268196273764765, 'clubs': 3.4464050690798693, 'college': 1.5613662703175555, 'blues': 4.064837205074066, 'for': 0.29145011737314763, 'falcons': 5.868501576808439, 'currently': 1.637088969126014, 'hill': 3.794313330511949, 'drawn': 4.96062941539988, 'cause': 4.8023464982877115, 'kicked': 5.142950972193835, 'exchange': 4.113331555012676, 'mckernan': 9.600201028105303, '19982000': 6.509158574746988, 'brisbaneafter': 10.986495389225194, 'losing': 3.773463729390325, 'essendon': 6.016682089649193, 'along': 2.5088749729287803, 'teaches': 3.7712554104950966, 'by': 0.37455341206197373, 'box': 4.57
```

In [35]:
```
# let's add another column to people SFrame to store tfidf value.

people['tfidf_value'] = tfidf_for_wiki_people
```

In [36]:
```
people.head()
```

Out[36]:

URI	name	text	word_count
<http://dbpedia.org/resource/Digby_Morrell> ...	Digby Morrell	digby morrell born 10 october 1979 is a former ...	{'selection': 1, 'carltons': 1, 'being': ...
<http://dbpedia.org/resource/Alfred_J._Lewy> ...	Alfred J. Lewy	alfred j lewy aka sandy lewy graduated from ...	{'precise': 1, 'thomas': 1, 'closely': 1, ...
<http://dbpedia.org/resource/Harpdog_Brown> ...	Harpdog Brown	harpdog brown is a singer and harmonica player who ...	{'just': 1, 'issued': 1, 'mainly': 1, 'nominat ...
<http://dbpedia.org/resource/Franz_Rottensteiner> ...	Franz Rottensteiner	franz rottensteiner born in waidmannsfeld lower ...	{'all': 1, 'bauforschung': 1, ...
<http://dbpedia.org/resource/G-Enka> ...	G-Enka	henry krvits born 30 december 1974 in tallinn ...	{'they': 1, 'gangstergenka': 1, ...
<http://dbpedia.org/resource/Sam_Henderson> ...	Sam Henderson	sam henderson born october 18 1969 is an ...	{'currently': 1, 'less': 1, 'being': 1, ...
<http://dbpedia.org/resource/Aaron_LaCrate> ...	Aaron LaCrate	aaron lacrate is an american music producer ...	{'exclusive': 2, 'producer': 1, 'show' ...
<http://dbpedia.org/resource/Trevor_Ferguson> ...	Trevor Ferguson	trevor ferguson aka john farrow born 11 november ...	{'taxi': 1, 'salon': 1, 'gangs': 1, 'being': 1, ...
<http://dbpedia.org/resource/Grant_Nelson> ...	Grant Nelson	grant nelson born 27 april 1971 in london ...	{'houston': 1, 'frankie': 1, 'labels': 1, ...
<http://dbpedia.org/resource/Cathy_Caruth> ...	Cathy Caruth	cathy caruth born 1955 is frank h t rhodes ...	{'phenomenon': 1, 'deborash': 1, 'both' ...

word_nv	tfidf_value

{'selection': 1, 'carltons': 1, 'being': ...	{'selection': 3.836578553093086, ...
{'precise': 1, 'thomas': 1, 'closely': 1, ...	{'precise': 6.44320060695519, ...
{'just': 1, 'issued': 1, 'mainly': 1, 'nominat ...	{'just': 2.7007299687108643, ...
{'all': 1, 'bauforschung': 1, ...	{'all': 1.6431112434912472, ...
{'they': 1, 'gangstergenka': 1, ...	{'they': 1.8993401178193898, ...
{'currently': 1, 'less': 1, 'being': 1, ...	{'currently': 1.637088969126014, ...
{'exclusive': 2, 'producer': 1, 'show' ...	{'exclusive': 10.455187230695827, ...
{'taxi': 1, 'salon': 1, 'gangs': 1, 'being': 1, ...	{'taxi': 6.0520214560945025, ...
{'houston': 1, 'frankie': 1, 'labels': 1, ...	{'houston': 3.935505942157149, ...

In [38]:
```
obama
```

Out[38]:

URI	name	text	word_count
<http://dbpedia.org/resource/Barack_Obama> ...	Barack Obama	barack hussein obama ii brk husen bm born august ...	{'operations': 1, 'represent': 1, 'offi ...

word_nv
{'operations': 1, 'represent': 1, 'offi ...

[1 rows x 5 columns]

In [41]:
```
# now let's find out tf-idf for Obama.
# first read the obama values again, as it containts its own ididf value.
obama = people[people['name']=='Barack Obama']
```

In [47]:
```
# now let's get the tf-idf value for Obama.
obama[['tfidf_value']].stack('tfidf_value', new_column_name=['word','tfidf_value']).sort('tfidf_
```

Out[47]:

word	tfidf_value
obama	43.2956530721
act	27.678222623
iraq	17.747378588
control	14.8870608452
law	14.7229357618
ordered	14.5333739509
military	13.1159327785
involvement	12.7843852412
response	12.7843852412
democratic	12.4106886973

[273 rows x 2 columns]
Note: Only the head of the SFrame is printed.
You can use print_rows(num_rows=m, num_columns=n) to print more rows and columns.

In [ ]:
```
# this make more sense for the obama document itself.  Without tf-idf, it was showing that
# "the", "a" etc. had more counts.  But with tf-idf, it reduces the importance of that
# as it uses log (count) function.
```

In [51]:
```
# let's find out if Obama is closer to Clinton or Bill Gates.

clinton = people[people['name'] == 'Bill Clinton']

gates = people[people['name'] == 'Bill Gates']
```

In [56]:
```
# now let's use cosine function to understand if Obama is closer to Clinton or Gates.
# [0] is used for syntax even though it has only one row to access.

graphlab.distances.cosine(obama['tfidf_value'][0], clinton['tfidf_value'][0])
```

Out[56]: 0.8339854936884276

In [57]:
```
graphlab.distances.cosine(obama['tfidf_value'][0], gates['tfidf_value'][0])
```

Out[57]: 0.9900304363196061

In [58]:
```
# from distanct point of view, Obama is closer to Clinton than Gates.
```

In [61]:
```
# now let's build K nearest neighbour model for document retrieval.

knn_model = graphlab.nearest_neighbors.create(people,features=['tfidf_value'],label='name')
```

```
Starting brute force nearest neighbors model training.
```

In [62]:
```
# now find out who is similar to Obama?

knn_model.query(obama)
```

```
Starting pairwise querying.
+--------------+---------+-------------+--------------+
| Query points | # Pairs | % Complete. | Elapsed Time |
+--------------+---------+-------------+--------------+
| 0            | 1       | 0.00169288  | 30.755ms     |
| Done         |         | 100         | 509.943ms    |
+--------------+---------+-------------+--------------+
```

Out[62]:

query_label	reference_label	distance	rank
0	Barack Obama	0.0	1
0	Joe Biden	0.794117647059	2
0	Joe Lieberman	0.794685990338	3
0	Kelly Ayotte	0.811989100817	4
0	Bill Clinton	0.813852813853	5

[5 rows x 4 columns]

In [63]:
```
# who is similar to jolie?

jolie = people[people['name'] == 'Angelina Jolie']

knn_model.query(jolie)
```

```
Starting pairwise querying.
+--------------+---------+-------------+--------------+
| Query points | # Pairs | % Complete. | Elapsed Time |
+--------------+---------+-------------+--------------+
| 0            | 1       | 0.00169288  | 22.496ms     |
| Done         |         | 100         | 418.058ms    |
+--------------+---------+-------------+--------------+
```

Out[63]:

query_label	reference_label	distance	rank
0	Angelina Jolie	0.0	1

0	Brad Pitt	0.784023668639	2
0	Julianne Moore	0.795857988166	3
0	Billy Bob Thornton	0.803069053708	4
0	George Clooney	0.8046875	5

[5 rows x 4 columns]

In [69]:
```
# who is similar to scarlett?
scarlett = people[people['name'] == 'Scarlett Johansson']

knn_model.query(scarlett)
```

```
Starting pairwise querying.

+--------------+---------+-------------+--------------+
| Query points | # Pairs | % Complete. | Elapsed Time |
+--------------+---------+-------------+--------------+
| 0            | 1       | 0.00169288  | 38.117ms     |
| Done         |         | 100         | 442.503ms    |
+--------------+---------+-------------+--------------+
```

Out[69]:

query_label	reference_label	distance	rank
0	Scarlett Johansson	0.0	1
0	Jennifer Aniston	0.79	2
0	Jennifer Connelly	0.809210526316	3
0	Robert Downey, Jr.	0.811965811966	4
0	Chlo%C3%AB Sevigny	0.8125	5

[5 rows x 4 columns]