solution.py

```
1   from __future__ import print_function
2   from __future__ import division
3   import os
4   import time
5   import pandas as pd
6   import numpy as np
7   from sklearn.cluster import MiniBatchKMeans, DBSCAN
8
9   # Get rid of sklearn deprecation warnings
10  import warnings
11  warnings.filterwarnings("ignore", category=DeprecationWarning)
12
13  # Load data from HDF5 file
14  # (I have loaded the JSON data into a pandas DataFrame and stored in
15  # an HDF5 file for faster access)
16  datadir = os.path.realpath('./data')
17  with pd.HDFStore(os.path.join(datadir, 'tweets_1M.h5')) as store:
18      tweets = store.tweets
19
20  # Constrain to Bay Area
21  tweets = tweets.loc[(tweets.lat > 36.903929764) &
22                      (tweets.lat < 38.853939589) &
23                      (tweets.lng > -123.528897483) &
24                      (tweets.lng < -121.213352822)]
25  print('Size of full dataset: {}'.format(len(tweets)))
26
27  # Set index to id for easy matching
28  tweets.set_index('id', inplace=True)
29
30  # Start timing implementation
31  t0 = time.time()
32
33  # MiniBatch section
34  mb = MiniBatchKMeans(n_clusters=100, init='k-means++', n_init=10, batch_size=1000)
35  data = tweets.as_matrix(columns=['lat', 'lng'])
36  mb.fit(data)
37  tweets['mb_cluster'] = mb.labels_   # Add labels back into DataFrame
38
39  # DBSCAN section
40  meters = 100      # Transform meters to degrees (roughly)
41  eps = meters / 100000
42
43  for i in tweets.mb_cluster.unique():
44      subset = tweets.loc[tweets.mb_cluster == i]
45      db = DBSCAN(eps=eps, min_samples=100)
46      data = subset.as_matrix(columns=['lat', 'lng'])
47      db.fit(data)
48      subset['db_cluster'] = db.labels_
49      tweets.loc[tweets.mb_cluster == i, 'db_cluster'] = subset['db_cluster']
50
51  # Set final cluster variable
52  tweets['cluster'] = tweets.mb_cluster + (tweets.db_cluster.replace(-1.0, np.nan) / 100)
53  print('Number of unique clusters generated: {}'.format(len(tweets.cluster.unique())))
54
55  t1 = time.time() - t0
56  print('Implementation time: {}'.format(t1))
57
58  # Save results
59  with pd.HDFStore(os.path.join(datadir, 'results.h5'), mode='w') as results:
60      results['results'] = tweets
```