

## K Means Clustering, Agglomerative Clustering, DBSCAN Clustering:

Objective :

1. Perform featurization BoW,TFIDF, Avg Word2Vec, tf-idf-Word2Vec.
2. At the end of the assignment, please elaborate on the step by step procedure you followed to solve the assignment.

### 3. DBSCAN Clustering:

1. Find the best 'Eps' using the elbow-knee method.
2. Try representing those reviews in a cluster as a word cloud so that it would be more comprehensible what a particular cluster represents.

#### Step-By-Step procedure

1. In this, we need to work with all 4-vectorizers (BOW, TFIDF,Avg w2v and TF-IDF weighted w2v) where we will convert our texted review into numerical(vector) form in order to apply any Model on it.
2. After that we will take our cleandtext(i.e cleand text means we have already cleaned our data by removing stops words, other this which are going to affect our model) and then we will take only test data not their respective class label because as we know we are going to apply K-means clustering algo which do not require class labels. what it does is it mainly group/clusters the similar data points
3. As we know DBSCAN is density based clustering algo and in this we have to hyperparameters i.e min\_pts and EPS what are these min\_pts i.e min\_samples is The number of samples (or total weight) in a neighborhood for a point to be considered as a core point. and eps is the maximum distance between two samples for them to be considered as in the same neighborhood. and we typically try to choose larger min\_points because which helps to remove the outliers
4. We typically choose min\_pts is 2\*dim-of-our data and EPS using elbow-knee methods.
5. After getting min\_points and EPS we try to implement DBSCAN model with different EPS near by to best EPS and try to read and represent the reviews for each EPS

In [1]:

```
%matplotlib inline
import warnings
warnings.filterwarnings("ignore")
import warnings
warnings.filterwarnings("ignore", category=DeprecationWarning)

import sqlite3
import pandas as pd
import numpy as np
import nltk
import string
import matplotlib.pyplot as plt
import seaborn as sns
from sklearn.feature_extraction.text import TfidfTransformer
from sklearn.feature_extraction.text import TfidfVectorizer

from sklearn.feature_extraction.text import CountVectorizer
from sklearn.metrics import confusion_matrix
from sklearn import metrics
from sklearn.metrics import roc_curve, auc
from nltk.stem.porter import PorterStemmer

import re
# Tutorial about Python regular expressions: https://pymotw.com/2/re/
import string

from gensim.models import Word2Vec
from gensim.models import KeyedVectors
import pickle
#taking cleaned data i.e in Reviews table from final sql database
#making connection with database
conn = sqlite3.connect('final.sqlite')
final = pd.read_sql_query(""" SELECT * FROM Reviews""", conn)
```

```
C:\Users\nisha\Anaconda3\lib\site-packages\gensim\utils.py:1212: UserWarning: detected Windows; aliasing chunkize to chunkize_serial
warnings.warn("detected Windows; aliasing chunkize to chunkize_serial")
```

In [2]:

```
final = final[:5000]
print(len(final))
```

5000

In [3]:

```
CleanedText = final['CleanedText'];
text=final.CleanedText.values
```

In [ ]:

```
def dbscan_implement_diff_eps(n_eps,min_pts,data):
    print("Note: Cluster labels for each point in the dataset given to fit(). Noisy samples are gi
ven the label -1.\n")
    for eps_val in n_eps:
        model = DBSCAN(eps=eps_val, min_samples=min_pts).fit(data)
        print("#" * 25, "DBSCAN with EPS = ", eps_val, " ", "#" * 25)
        print("\nReviews for the clusters -> 1: \n")
        print("Clusters",set(model.labels_))
        n_clusters = len(set(model.labels_))
        print("DBSCAN with EPS =%.2f and min_samples = %d the no of Clusters we get is = %d"%(eps_
val,min_pts,n_clusters))
        print('\n')
        review_count = 1
        for i in range(model.labels_.shape[0]):
            if model.labels_[i] == 0:
                if review_count == 3:
                    break;
                else:
                    print('Review - ',review_count)
                    print(reviews[i])
                    cluster_wordcloud_generated_image_fun(reviews[i])
                    review_count = review_count+1

        print("_" * 100)
        print("\n")
```

In [21]:

```
from tqdm import tqdm
import os

from wordcloud import WordCloud, STOPWORDS
import matplotlib.pyplot as plt
stopwords = set(STOPWORDS)
import seaborn as sns;

# Reference http://iopsience.iop.org/article/10.1088/1755-1315/31/1/012012/pdf
# function to determinethe distance of nth-nearest neighbour to all points in a multi-dimensional
array
reviews = final['Text'].values
from sklearn.cluster import DBSCAN
from sklearn.decomposition import PCA

def n_neighbour(vectors , n):
    distance = []
    for point in vectors:
        temp = np.sort(np.sum((vectors-point)**2,axis=1),axis=None)
        distance.append(temp[n])
    return np.sqrt(np.array(distance))

def dbscan_implement_diff_eps(n_eps,min_pts,data):
    print("\nNote: Cluster labels for each point in the dataset given to fit(). Noisy samples are
given the label -1.\n")
    for eps_val in n_eps:
        model = DBSCAN(eps=eps_val, min_samples=min_pts).fit(data)
        print("#" * 30, "DBSCAN with EPS = ", eps_val, " ", "#" * 30)
```

```

clusters_list = list(set(model.labels_))
n_clusters = len(set(model.labels_))
print("DBSCAN with EPS =%.2f and min_pts = %d the no of Clusters we get is = %d"%(eps_val,
min_pts,n_clusters))
#         print('\n')

for cluster_name in clusters_list:
    review_count = 1
    print("*" * 100)
    if cluster_name == -1:
        print("Noisy samples are given the label -> ",cluster_name)
    else:
        print("Reviews of cluster -> ",cluster_name+1)
    print("*" * 100)
    for i in range(model.labels_.shape[0]):
        if model.labels_[i] == cluster_name:
            if review_count == 3:
                break;
            else:
                print('Review - ',review_count)
                print(reviews[i])
                cluster_wordcloud_generated_image_fun(reviews[i])
                review_count = review_count+1

    print("_" * 100)
    print("\n")

def cluster_wordcloud_generated_image_fun(text):
    wordcloud = WordCloud(max_font_size=50, max_words=100,stopwords=stopwords ,background_color="white").generate(text)
    plt.figure()
    plt.imshow(wordcloud, interpolation="bilinear")
    plt.axis("off")
    plt.show()

```

## Bow

Applying Bow vectorizer on data

In [5]:

```

#BOW
from sklearn.feature_extraction.text import CountVectorizer
vectorizer = CountVectorizer(min_df=200)
vocabulary= vectorizer.fit(text)
#print("the shape of out text BOW vectorizer ",vocabulary.get_shape())

bow_data= vectorizer.transform(text)
print("the shape of out text BOW vectorizer ",bow_data.get_shape())

```

the shape of out text BOW vectorizer (5000, 173)

**Find the best 'Eps' using the elbow-knee method.**

In [7]:

```

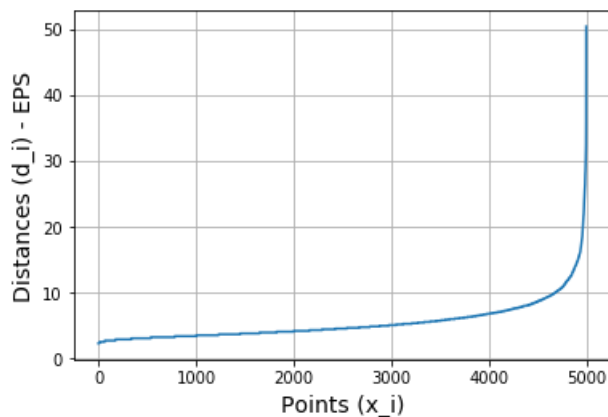
#bow_data.shape[1]
min_pts = 2*bow_data.shape[1]
# Computing distances of nth-nearest neighbours
dist = n_neighbour(bow_data.toarray(),min_pts)
sorted_distance = np.sort(dist)
# bow_data.shape[0]
n_points = [i for i in range(bow_data.shape[0])]

# Draw distances(d_i) VS points(x_i) plot
plt.plot(n_points, sorted_distance)
plt.xlabel('Points (x_i)',size=14)
plt.ylabel('Distances (d_i) - EPS',size=14)
plt.title('Distances and Points Graph Plot\n',size=18)

```

```
plt.grid()
plt.show()
```

## Distances and Points Graph Plot



## Implementing DBSCAN

So after getting best EPS lets try different nearby eps to see that the change in eps what will happen to our clusters

In [22]:

```
#So after getting best EPS lets try different nearby eps to see that the change in eps what will h
appen to our clusters
#Cluster labels for each point in the dataset given to fit(). Noisy samples are given the label -1
.
eps_list = [8,10,12,14,17]
dbscan_implement_diff_eps(eps_list,min_pts,bow_data.toarray())
```

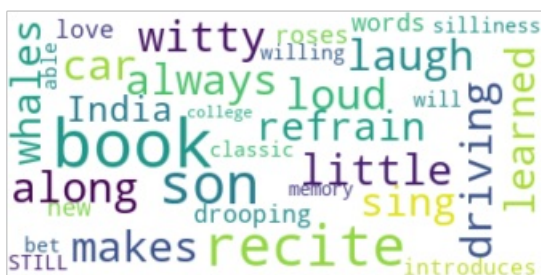
Note: Cluster labels for each point in the dataset given to fit(). Noisy samples are given the label -1.

```
##### DBSCAN with EPS = 8 #####
DBSCAN with EPS =8.00 and min_pts = 346 the no of Clusters we get is = 2
*****
```

```
Reviews of cluster -> 1
*****
```

Review - 1

this witty little book makes my son laugh at loud. i recite it in the car as we're driving along and he always can sing the refrain. he's learned about whales, India, drooping roses: i love all the new words this book introduces and the silliness of it all. this is a classic book i am willing to bet my son will STILL be able to recite from memory when he is in college



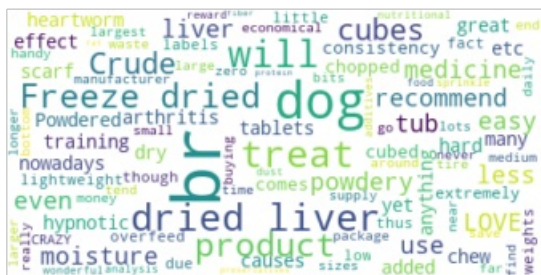
Review - 2

I grew up reading these Sendak books, and watching the Really Rosie movie that incorporates them, and love them. My son loves them too. I do however, miss the hard cover version. The paperbacks seem kind of flimsy and it takes two hands to keep the pages open.





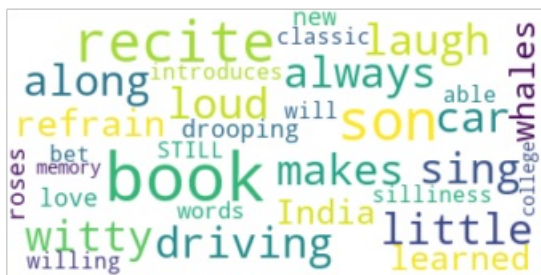
easy to use.<br /><br />~~~~~<br />Here's the nutritional analysis:<br /><br />-100% freeze dried liver<br />-NO additives or preservatives.<br /><br />-Crude protein..... not less than 50%<br />-Crude fat..... not less than 5%<br />-Crude fiber..... not more than 3%<br />-Moisture..... not more than 6%



```
##### DBSCAN with EPS = 10 #####
DBSCAN with EPS =10.00 and min_pts = 346 the no of Clusters we get is = 2
*****
```

```
Reviews of cluster -> 1
*****
```

Review - 1  
this witty little book makes my son laugh at loud. i recite it in the car as we're driving along and he always can sing the refrain. he's learned about whales, India, drooping roses: i love all the new words this book introduces and the silliness of it all. this is a classic book i am willing to bet my son will STILL be able to recite from memory when he is in college



Review - 2  
I grew up reading these Sendak books, and watching the Really Rosie movie that incorporates them, and love them. My son loves them too. I do however, miss the hard cover version. The paperbacks seem kind of flimsy and it takes two hands to keep the pages open.



```
*****
Noisy samples are given the label -> -1
*****
```

Review - 1  
These days, when a person says, "chicken soup" they're probably going to follow up those words with, "for the soul" or maybe "for the teenaged soul". Didn't used to be that way. Why I can remember

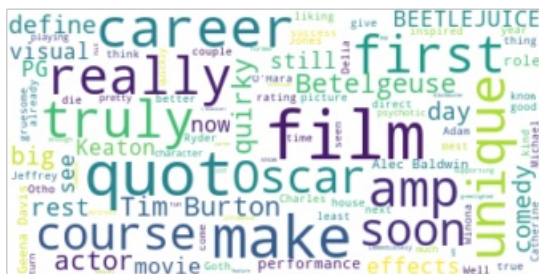








with their chubby, eerie chauffer Otto (Glenn Shadix) have already moved in, and inspire true disgust in the newly-dead couple. However, they seem to have an ally in the older couple's death-obsessed Goth daughter Lydia (Winona Ryder) see an afterlife advertisement for a "Bio-Exorcist" named Betelgeuse (Michael Keaton) and decide that their only hope in driving out the detestable Charles, Delia, and Otto is in using the wild, unpredictable Betelgeuse. Of course, good ol' Betelgeuse will gladly provide his unique brand of service, but at a very high price. <p> Alec Baldwin & Geena Davis fill their roles very well. The ironic thing is that this was the first film I'd ever seen Alec Baldwin in, so my first impression of him as the nerdy, good-natured Adam ended up being turned around by 180 degrees by the psychotic, tough-guy roles that would define the rest of his career. That turnabout came pretty quickly, actually, as the very next film I would see him in was MARRIED TO THE MOB (1988), in which he was a mobster unbeknownst to his wife who gets offed early on, but it was his psychotic turn in the extremely quirky and violent MIAMI BLUES (1989) that really shocked me and made me realize that Alec really had some range to him (unfortunately, he would end up being typecast as psychos for the rest of his career). However, Geena Davis, herself at the cusp of stardom (and Oscar acclaim) is playing pretty much the quirky kind of character that would eventually define her career. In fact, it was just the following year that she would win the Oscar for Best Supporting Actress for playing the truly quirky and inspired Muriel Pritchett in the otherwise uninspired THE ACCIDENTAL TOURIST (1989). This film, of course, also boosted the career of the now-infamous Winona Ryder, who quickly became the hot new "it" girl in the late 80's with this movie and with others soon to follow: 1969 (1988), HEATHERS (1989), GREAT BALLS OF FIRE! (also 1989), EDWARD SCISSORHANDS (1990, and another Tim Burton film) and MERMAIDS (also 1990). Of course, former AMADEUS (1984) emperor and FERRIS BUELLER'S DAY OFF (1986) principal Jeffrey Jones, now infamous for his own reasons, is also terrific here, as is former "SC-TV" alumnus Catherine O'Hara. We also get a welcome supporting performance from the great old character actress, Sylvia Sidney, as the chain-smoking receptionist Juno!<p>But it was Michael Keaton, who was already an up-and-coming comedic actor known for quirky hit comedies such as the Ron Howard-directed NIGHT SHIFT (1982), MR. MOM (1983), JOHNNY DANGEROUSLY (1984) and GUNSHO (1986), who really stole the show and hit paydirt in the process. He would immediately be elevated to "A" list status with his truly brilliant, maniacal and hilarious performance as the titular Betelgeuse, going immediately on to bigger & better things with Tim Burton himself, starring soon after as BATMAN (1989)! It's funny; I think that in 1988, Keaton was robbed of Oscar nominations not only for this truly unique comedic role, but for also his surprising dramatic turn as a recovering alcoholic in CLEAN AND SOBER. It just shows that, for whatever reason, comedians just don't get recognized enough for their acting ability.<p>BEETLEJUICE is a lot of fun. It looks like it must have been a lot of fun to make. Despite the liberal PG rating, I would recommend it only for ages 9 & up, as the humor is quite adult and the rather gruesome visuals are likely to give any young child nightmares. It is still a truly unique movie with incredible special effects, and it still makes me laugh to this day. Oddly enough, whenever I see it, it makes me that much less afraid to die!<p>HIGHLY RECOMMENDED



## Review - 2

Picked up a set of 2 at the local home improvement store for ~\$11. The design of this trap seems to count on the animal pushing dirt in front of him and the dirt is what sets off the trap. In these instances, the gopher will likely be caught behind the neck or shoulders. I see a lot of holes on our property where the hole is not plugged and plenty of sign that he has been out scampering around. In these cases, he'll get caught in the hip area with his nose up against where the trigger plate would be set. This just makes it a little more awkward in emptying the trap. The cinch type trap looks to be just as effective but much easier to empty. The cinch trap will cost up to 2-3 times more, though. I might get one or two cinch traps, but 9/10 will be the 610 type.<br /><br />I have read at least one review where the sweep of the jaws could be an issue, preventing the trap from being seated in small holes and from closing effectively. This is really easy to work around. I first thought to shorten the jaws, but decided to just make the hole bigger when setting it. I learned that gophers are not shy of these traps or your activity around the hole. In very short order you can excavate the hole to a size that allows the trap to operate freely...and the gopher will come...especially if you notice fresh activity around the hole.<br /><br />Much of the literature on setting traps recommends digging back from the plugged hole to access the main tunnel and then set the traps in pairs so the gopher has to enter a trap from either direction in order to replug the hole. I have found this is not really necessary. If the hole I am working was freshly plugged, one trap in the hole works every time.<br /><br />I also noticed that a small plug doesn't necessarily mean a small gopher. In one instance, I almost passed on setting a trap in what I thought was just going to be wasted effort. I cleared dirt from a plug the size of a 50 cent piece and it opened up into a tunnel about 2 1/2 inches! The gopher I got out of that hole was a normal sized gopher.<br /><br />I also read that a 3 degree difference in the gopher's comfort zone in the burrow is enough to cause him to do something about it. The fact that light appears





\*\*\*\*\*

Noisy samples are given the label -> -1

\*\*\*\*\*

Review - 1

Tim Burton had come off the blockbuster success of his first feature film PEE-WEE'S BIG ADVENTURE (1985) when he was greenlighted by Warner Bros. to direct the project he REALLY wanted to make. It was the picture that would define the rest of his career to this day: the supernaturally weird black comedy (and weirdly-titled) BEETLEJUICE (1988). It was so unique, so strange to me that I didn't know what to make of it when I first saw it on video in 1989. I remember from that first time liking parts of it, but not liking the picture as a whole. The movie was so frenetic, so odd in its mixture of Goth Horror, Comedy, gruesome visual effects (I'm still kind of amazed that it got a PG rating instead of PG-13) and cartoonish slapstick that I really didn't know what to make of it. Well, now that I've seen it at least a half-dozen times since (three of them in the past year alone), I can say that it has held up better than many of the most popular 80's films (TOP GUN, etc) and has become a classic all unto itself. <p>With BEETLEJUICE, Tim Burton showed not only that he could direct effective comedy and illicit inspired performances from his actors, but also bring a true artist's touch to the big screen. Of course, the film won the Oscar for Best Makeup (as if there was any real question about it), but I also think that it should have also been at least nominated for Visual Effects and Art Direction, as those truly mesmerizing aspects all come together to create a product that is so unique. Not only that, but this film helped make the careers of several actors who would go on to major success soon after. <p>Two of those actors are Alec Baldwin and Geena Davis, as the ill-fated newlyweds Adam and Barbara Maitland who, after buying a big Victorian dream house at the film's outset, die in a bizarre car accident soon after. Slowly realizing that they've passed on to the next world, they don't want to give up their house to the pathetically bourgeois Charles & Delia Deetz (Jeffrey Jones & Catherine O'Hara), who along with their chubby, effete chauffeur Otho (Glenn Shadix) have already moved in, and inspire true disgust in the newly-dead couple. However, they seem to have an ally in the older couple's death-obsessed Goth daughter Lydia (Winona Ryder) see an afterlife advertisement for a "Bio-Exorcist" named Betelgeuse (Michael Keaton) and decide that their only hope in driving out the detestable Charles, Delia, and Otho is in using the wild, unpredictable Betelgeuse. Of course, good ol' Betelgeuse will gladly provide his unique brand of service, but at a very high price. <p>Alec Baldwin & Geena Davis fill their roles very well. The ironic thing is that this was the first film I'd ever seen Alec Baldwin in, so my first impression of him as the nerdy, good-natured Adam ended up being turned around by 180 degrees by the psychotic, tough-guy roles that would define the rest of his career. That turnabout came pretty quickly, actually, as the very next film I would see him in was MARRIED TO THE MOB (1988), in which he was a mobster unbeknownst to his wife who gets offed early on, but it was his psychotic turn in the extremely quirky and violent MIAMI BLUES (1989) that really shocked me and made me realize that Alec really had some range to him (unfortunately, he would be end up being typecast as psychos for the rest of his career). However, Geena Davis, herself at the cusp of stardom (and Oscar acclaim) is playing pretty much the quirky kind of character that would eventually define her career. In fact, it was just the following year that she would win the Oscar for Best Supporting Actress for playing the truly quirky and inspired Muriel Pritchett in the otherwise uninspired THE ACCIDENTAL TOURIST (1989). This film, of course, also boosted the career of the now-infamous Winona Ryder, who quickly became the hot new "it" girl in the late 80's with this movie and with others soon to follow: 1969 (1988), HEATHERS (1989), GREAT BALLS OF FIRE! (also 1989), EDWARD SCISSORHANDS (1990, and another Tim Burton film) and MERMAIDS (also 1990). Of course, former AMADEUS (1984) emperor and FERRIS BUELLER'S DAY OFF (1986) principal Jeffrey Jones, now infamous for his own reasons, is also terrific here, as is former "SC-TV" alumnus Catherine O'Hara. We also get a welcome supporting performance from the great old character actress, Sylvia Sidney, as the chain-smoking receptionist Juno! <p>But it was Michael Keaton, who was already an up-and-coming comedic actor known for quirky hit comedies such as the Ron Howard-directed NIGHT SHIFT (1982), MR. MOM (1983), JOHNNY DANGEROUSLY (1984) and GUNG HO (1986), who really stole the show and hit paydirt in the process. He would immediately be elevated to "A" list status with his truly brilliant, maniacal and hilarious performance as the titular Betelgeuse, going immediately on to bigger & better things with Tim Burton himself, starring soon after as BATMAN (1989)! It's funny; I think that in 1988, Keaton was robbed of Oscar nominations not only for this truly unique comedic role, but for also his surprising dramatic turn as a recovering alcoholic in CLEAN AND SOBER. It just shows that, for whatever reason, comedians just don't get recognized enough for their acting ability. <p>BEETLEJUICE is a lot of fun. It looks like it must have been a lot of fun to make. Despite the liberal PG rating, I would recommend it only for ages 9 & up, as the humor is quite adult and the rather gruesome visuals are likely to give any young child nightmares. It is still a truly unique movie with incredible special effects, and it still makes me laugh to this day. Oddly enough, whenever I see it, it makes me that much less afraid to die! <p>HIGHLY RECOMMENDED







Betelgeuse, named Betelgeuse (Michael Keaton), and decide that their only hope in driving out the  
 e detestable Charles, Delia, and Otho is in using the wild, unpredictable Betelgeuse. Of course,  
 good ol' Betelgeuse will gladly provide his unique brand of service, but at a very high price. <p>  
 Alec Baldwin & Geena Davis fill their roles very well. The ironic thing is that this was the f  
 irst film I'd ever seen Alec Baldwin in, so my first impression of him as the nerdy, good-natured  
 Adam ended up being turned around by 180 degrees by the psychotic, tough-guy roles that would  
 define the rest of his career. That turnabout came pretty quickly, actually, as the very next film  
 I would see him in was MARRIED TO THE MOB (1988), in which he was a mobster unbeknownst to his wif  
 e who gets offed early on, but it was his psychotic turn in the extremely quirky and violent MIAMI  
 BLUES (1989) that really shocked me and made me realize that Alec really had some range to him (un  
 fortunately, he would be end up being typecast as psychos for the rest of his career). However, Ge  
 ena Davis, herself at the cusp of stardom (and Oscar acclaim) is playing pretty much the quirky ki  
 nd of character that would eventually define her career. In fact, it was just the following year t  
 hat she would win the Oscar for Best Supporting Actress for playing the truly quirky and inspired  
 Muriel Pritchett in the otherwise uninspired THE ACCIDENTAL TOURIST (1989). This film, of course,  
 also boosted the career of the now-infamous Winona Ryder, who quickly became the hot new &quot;it&  
 quot; girl in the late 80's with this movie and with others soon to follow: 1969 (1988), HEATHERS  
 (1989), GREAT BALLS OF FIRE! (also 1989), EDWARD SCISSORHANDS (1990, and another Tim Burton film)  
 and MERMAIDS (also 1990). Of course, former AMADEUS (1984) emperor and FERRIS BUELLER'S DAY OFF (1  
 986) principal Jeffrey Jones, now infamous for his own reasons, is also terrific here, as is forme  
 r &quot;SC-TV&quot; alumnus Catherine O'Hara. We also get a welcome supporting performance from th  
 e great old character actress, Sylvia Sidney, as the chain-smoking receptionist Juno!<p>But it was  
 Michael Keaton, who was already an up-and-coming comedic actor known for quirky hit comedies such  
 as the Ron Howard-directed NIGHT SHIFT (1982), MR. MOM (1983), JOHNNY DANGEROUSLY (1984) and GUNG  
 HO (1986), who really stole the show and hit paydirt in the process. He would immediately be eleva  
 ted to &quot;A&quot; list status with his truly brilliant, maniacal and hilarious performance as t  
 he titular Betelgeuse, going immediately on to bigger & better things with Tim Burton himself,  
 starring soon after as BATMAN (1989)! It's funny; I think that in 1988, Keaton was robbed of Oscar  
 nominations not only for this truly unique comedic role, but for also his surprising dramatic turn  
 as a recovering alcoholic in CLEAN AND SOBER. It just shows that, for whatever reason, comedians j  
 ust don't get recognized enough for their acting ability.<p>BEETLEJUICE is a lot of fun. It looks  
 like it must have been a lot of fun to make. Despite the liberal PG rating, I would recommend it o  
 nly for ages 9 & up, as the humor is quite adult and the rather gruesome visuals are likely to  
 give any young child nightmares. It is still a truly unique movie with incredible special effects,  
 and it still makes me laugh to this day. Oddly enough, whenever I see it, it makes me that much  
 less afraid to die!<p>HIGHLY RECOMMENDED



Review - 2

Picked up a set of 2 at the local home improvement store for ~\$11. The design of this trap seems  
 to count on the animal pushing dirt in front of him and the dirt is what sets off the trap. In th  
 ese instances, the gopher will likely be caught behind the neck or shoulders. I see a lot of hole  
 s on our property where the hole is not plugged and plenty of sign that he has been out scampering  
 around. In these cases, he'll get caught in the hip area with his nose up against where the trigg  
 er plate would be set. This just makes it a little more awkward in emptying the trap. The cinch  
 type trap looks to be just as effective but much easier to empty. The cinch trap will cost up to  
 2-3 times more, though. I might get one or two cinch traps, but 9/10 will be the 610 type.<br /><br />  
 I have read at least one review where the sweep of the jaws could be an issue, preventing the  
 trap from being seated in small holes and from closing effectively. This is really easy to work a  
 round. I first thought to shorten the jaws, but decided to just make the hole bigger when setting  
 it. I learned that gophers are not shy of these traps or your activity around the hole. In very  
 short order you can excavate the hole to a size that allows the trap to operate freely...and the g  
 opher will come...especially if you notice fresh activity around the hole.<br /><br />Much of the  
 literature on setting traps recommends digging back from the plugged hole to access the main  
 tunnel and then set the traps in pairs so the gopher has to enter a trap from either direction in  
 order to replug the hole. I have found this is not really necessary. If the hole I am working wa  
 s freshly plugged, one trap in the hole works every time.<br /><br />I also noticed that a small p  
 lug doesn't necessarily mean a small gopher. In one instance, I almost passed on setting a trap i  
 n what I thought was just going to be wasted effort. I cleared dirt from a plug the size of a 50  
 cent piece and it opened up into a tunnel about 2 1/2 inches! The gopher I got out of that hole w  
 as a normal sized gopher!<br /><br />I also read that a 3 degree difference in the gopher's comfor  
 t zone in the burrow is enough to cause him to do something about it. The fact that light appears  
 at a point he previously plugged is also enough for him to quickly come back and see what  
 happened. One the first day I used these, I set one, then walked about 30 yards away and set anot  
 her. I checked the first one on the way back a few minutes later and found it had been tripped an



Next, I checked the first one on the way back a few minutes later and found it had been clipped and completely covered. I had to dig it out and found the little bugger had pushed a rock the size of a bird's egg into it! My first catch was a rock! A Bill Murray moment flashed through my mind. That gopher is now history, by the way.  
I am working with only 2 traps at the moment and secure them with a piece of butcher twine to my tunnel probe or a 12" piece of 1/4 in re-bar. This size hardware is probably overkill. You can probably tie the traps to those marking flags on wire type stems available at your local home improvement store, especially if you are setting a lot of traps. They may not need to be secured at all. When that gopher's body is in the trap, he isn't really going anywhere. If he can go anywhere with it, it might be best to let THAT gopher keep it as a souvenir!  
If you are not having any success at all with these traps, it would be a good idea to re-examine how and where the traps are being set. I have never used these things before and only bought this set after 4 years of poison pellets simply didn't work. Look for FRESH activity and don't be shy about providing enough room for these traps to work. So far, have set these only at holes that have been plugged, but will set the next time where there is fresh sign but no plug. I don't bait mine and the trigger plate can ALWAYS be seen from above after I set them...easier to check them. From all the mounds on our 2 acres, am guessing we have at least 40 gophers or more. These traps will be busy and I'll be keeping score.  
I give the trap 5 stars for effectiveness and 4 stars for the emptying issue. From my experience, no need to spend a lot of money...these things work.



## TF-IDF

In [29]:

```
#tfidf
# tf_idf_vect = TfidfVectorizer(ngram_range=(1,2))
tf_idf_vect = TfidfVectorizer(min_df=150)
vocabulary = tf_idf_vect.fit(text)
#print("the shape of out text TF-IDF vectorizer ",tf_idf_x_tr.get_shape())
tf_idf_data = tf_idf_vect.transform(text)
print("the shape of out text TF-IDF vectorizer ",tf_idf_data.get_shape())
```

```
the shape of out text TF-IDF vectorizer (5000, 240)
```

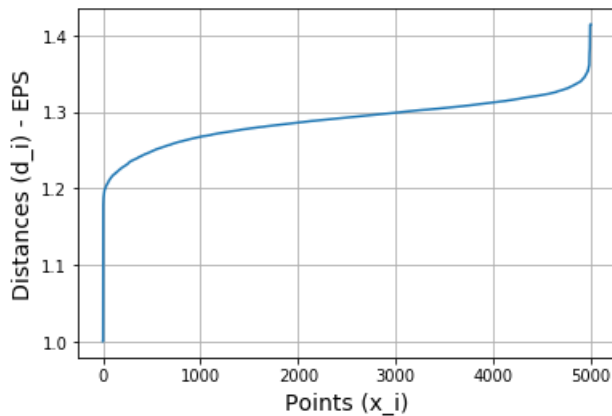
### Find the best 'Eps' using the elbow-knee method.

In [30]:

```
#tf_idf_data.shape[1]
min_pts = 2*tf_idf_data.shape[1]
# Computing distances of nth-nearest neighbours
dist = n_neighbour(tf_idf_data.toarray(),min_pts)
sorted_distance = np.sort(dist)
# tf_idf_data.shape[0]
n_points = [i for i in range(tf_idf_data.shape[0])]

# Draw distances(d_i) VS points(x_i) plot
plt.plot(n_points, sorted_distance)
plt.xlabel('Points (x_i)',size=14)
plt.ylabel('Distances (d_i) - EPS',size=14)
plt.title('Distances and Points Graph Plot\n',size=18)
plt.grid()
plt.show()
```

## Distances and Points Graph Plot



## Implementing DBSCAN

So after getting best EPS lets try different nearby eps to see that the change in eps what will happen to our clusters

In [31]:

```
#So after getting best EPS lets try different nearby eps to see that the change in eps what will h
appen to our clusters
#Cluster labels for each point in the dataset given to fit(). Noisy samples are given the label -1
.
min_pts = 2*tf_idf_data.shape[1]
eps_list = [0.5,0.8,1.2,2,3]
dbscan implement diff eps(eps list,min pts,tf idf data.toarray())
```

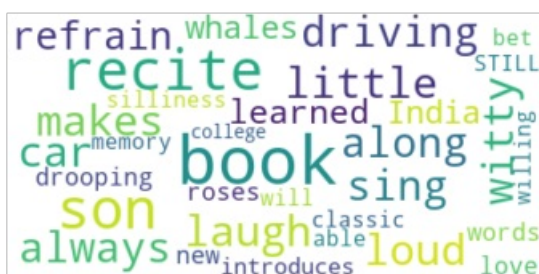
Note: Cluster labels for each point in the dataset given to `fit()`. Noisy samples are given the label -1.

```
##### DBSCAN with EPS = 0.5 #####
DBSCAN with EPS = 0.5 and min_pts = 480 the no of Clusters we get is = 1
*****
```

```
Noisy samples are given the label -> -1
*****
```

Review - 1

this witty little book makes my son laugh at loud. i recite it in the car as we're driving along and he always can sing the refrain. he's learned about whales, India, drooping roses: i love all the new words this book introduces and the silliness of it all. this is a classic book i am willing to bet my son will STILL be able to recite from memory when he is in college

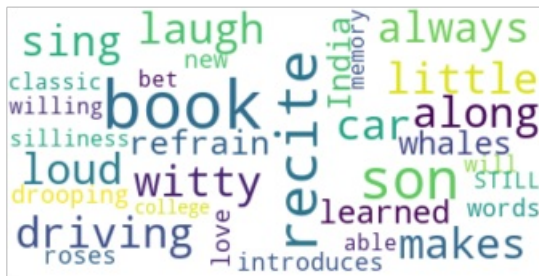


Review - 2

I grew up reading these Sendak books, and watching the Really Rosie movie that incorporates them, and love them. My son loves them too. I do however, miss the hard cover version. The paperbacks seem kind of flimsy and it takes two hands to keep the pages open.

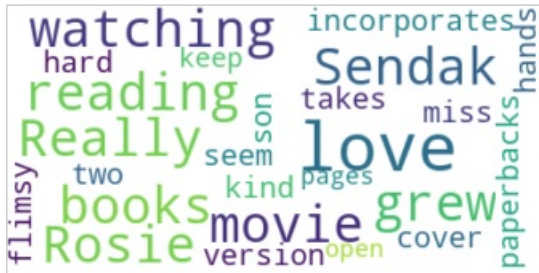






Review - 2

I grew up reading these Sendak books, and watching the Really Rosie movie that incorporates them, and love them. My son loves them too. I do however, miss the hard cover version. The paperbacks seem kind of flimsy and it takes two hands to keep the pages open.



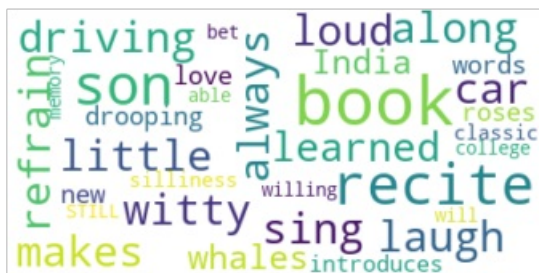
```
##### DBSCAN with EPS = 2 #####
DBSCAN with EPS =2.00 and min_pts = 480 the no of Clusters we get is = 1
*****
```

```
Reviews of cluster -> 1
```

\*\*\*\*\*

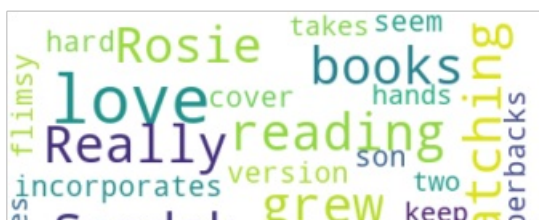
Review - 1

this witty little book makes my son laugh at loud. i recite it in the car as we're driving along a  
nd he always can sing the refrain. he's learned about whales, India, drooping roses: i love all t  
he new words this book introduces and the silliness of it all. this is a classic book i am  
willing to bet my son will STILL be able to recite from memory when he is in college



Review - 2

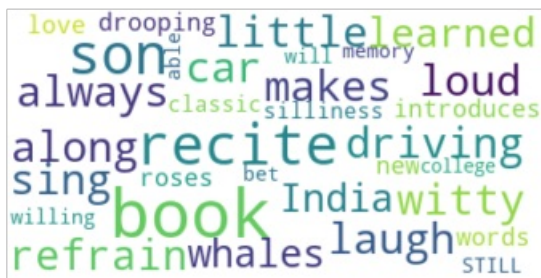
I grew up reading these Sendak books, and watching the Really Rosie movie that incorporates them, and love them. My son loves them too. I do however, miss the hard cover version. The paperbacks seem kind of flimsy and it takes two hands to keep the pages open.



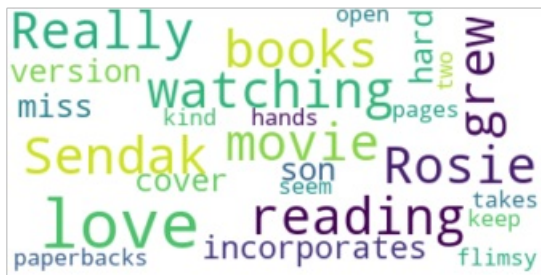
```
##### DBSCAN with EPS = 3 #####
DBSCAN with EPS =3.00 and min_pts = 480 the no of Clusters we get is = 1
*****
```

```
Reviews of cluster -> 1
*****
```

Review - 1  
 this witty little book makes my son laugh at loud. i recite it in the car as we're driving along a  
 nd he always can sing the refrain. he's learned about whales, India, drooping roses: i love all t  
 he new words this book introduces and the silliness of it all. this is a classic book i am  
 willing to bet my son will STILL be able to recite from memory when he is in college



Review - 2  
 I grew up reading these Sendak books, and watching the Really Rosie movie that incorporates them,  
 and love them. My son loves them too. I do however, miss the hard cover version. The paperbacks se  
 em kind of flimsy and it takes two hands to keep the pages open.



\*\*TF-IDF ENDS \*

## Word2Vec

In [32]:

```
#Word2Vec mode
#splitting train sentence in words
# Train your own Word2Vec model using your own text corpus
i=0
list_of_sent=[]
for sent in text:
    list_of_sent.append(sent.split())
```

◀ ▶

5000

```
#The Word to Vec model produces a vocabulary, with each word being represented by
#an n-dimensional numpy array
data_w2v_model=Word2Vec(list_of_sent,min_count=1,size=50, workers=4)
data_w2v_model.wv['man']
wlist =list(data_w2v_model.wv.vocab)
# wlist is a list of words
len(wlist)
```

10429

```
# CALCULATE AVG WORD2VEC FOR data
w2v_words = list(data_w2v_model.wv.vocab)
# compute average word2vec for each review.
data_sent_vectors = []; # the avg-w2v for each sentence/review is stored in this list
for sent in tqdm(list_of_sent): # for each review/sentence
    sent_vec = np.zeros(50) # as word vectors are of zero length
    cnt_words = 0; # num of words with a valid vector in the sentence/review
    for word in sent: # for each word in a review/sentence
        if word in w2v_words:
            vec = data_w2v_model.wv[word]
            sent_vec += vec
            cnt_words += 1
    if cnt_words != 0:
        sent_vec /= cnt_words
    data_sent_vectors.append(sent_vec)
print(len(data_sent_vectors))
print(len(data_sent_vectors[0]))
```

[illegible]

## Find the best 'Eps' using the elbow-knee method

```
min_pts = 2*data_sent_vectors[0].shape[0]
# Computing distances of nth-nearest neighbours
dist = n_neighbour(data_sent_vectors,min_pts)
sorted_distance = np.sort(dist)

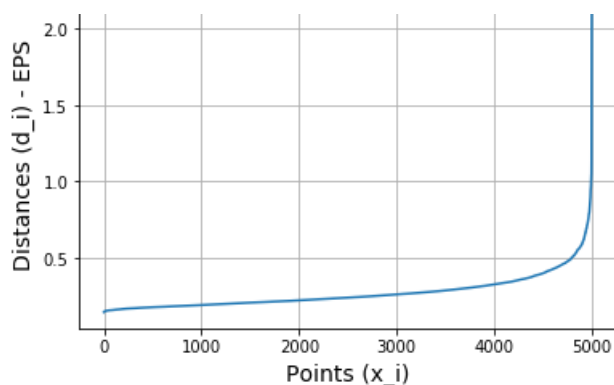
n_points = [i for i in range(len(data_sent_vectors))]

plt.plot(n_points, sorted_distance)
plt.xlabel('Points (x_i)',size=14)
plt.ylabel('Distances (d_i) - EPS',size=14)
plt.title('Distances and Points Graph Plot\n',size=18)
plt.grid()
plt.show()
```

Distances and Points Graph Plot







## Implementing DBSCAN

So after getting best EPS lets try different nearby eps to see that the change in eps what will happen to our clusters

In [38]:

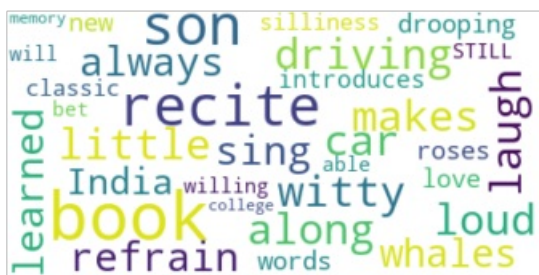
```
#So after getting best EPS lets try different nearby eps to see that the change in eps what will h
appen to our clusters
#Cluster labels for each point in the dataset given to fit(). Noisy samples are given the label -1
.
min_pts = 2*data_sent_vectors[0].shape[0]
eps_list = [0.5,0.8,1.2,2,3]
dbscan implement diff eps(eps list,min pts,data sent vectors)
```

Note: Cluster labels for each point in the dataset given to `fit()`. Noisy samples are given the label -1.

```
##### DBSCAN with EPS = 0.5 #####
DBSCAN with EPS=0.50 and min_pts = 100 the no of Clusters we get is = 2
#####
```

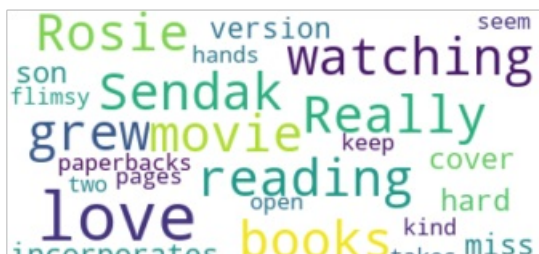
```
Reviews of cluster -> 1
*****
```

Review - 1  
this witty little book makes my son laugh at loud. i recite it in the car as we're driving along and he always can sing the refrain. he's learned about whales, India, dropping roses: i love all the new words this book introduces and the silliness of it all. this is a classic book i am willing to bet my son will STILL be able to recite from memory when he is in college



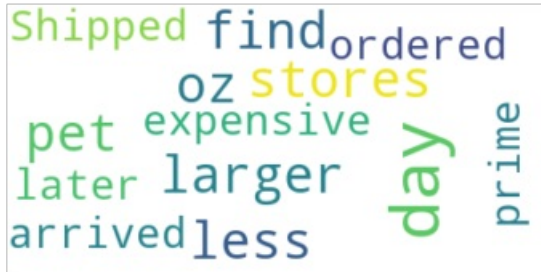
Review - 2

I grew up reading these Sendak books, and watching the Really Rosie movie that incorporates them, and love them. My son loves them too. I do however, miss the hard cover version. The paperbacks seem kind of flimsy and it takes two hands to keep the pages open.

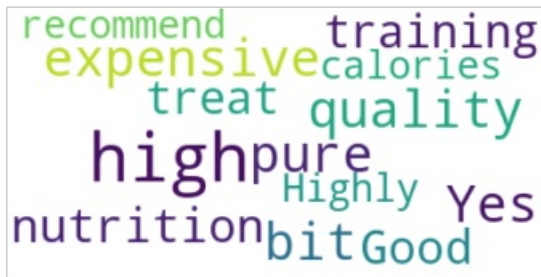


\*\*\*\*\*  
 Noisy samples are given the label -> -1  
 \*\*\*\*\*

Review - 1  
 This is 2.5 oz larger than what you can find in pet stores and less expensive. Shipped day the it was ordered and arrived 2 days later with prime.



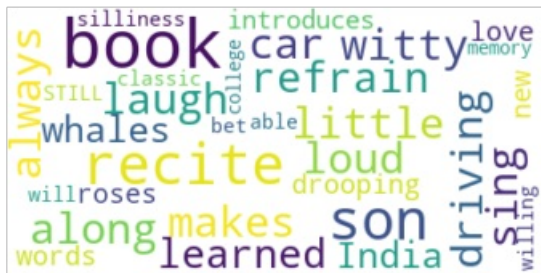
Review - 2  
 Yes, they are a bit expensive but, they are high quality, and pure nutrition. Good training treat and not too high in calories. Highly recommend this!



##### DBSCAN with EPS = 0.8 #####  
 DBSCAN with EPS =0.80 and min\_pts = 100 the no of Clusters we get is = 2  
 \*\*\*\*\*

Reviews of cluster -> 1  
 \*\*\*\*\*

Review - 1  
 this witty little book makes my son laugh at loud. i recite it in the car as we're driving along a nd he always can sing the refrain. he's learned about whales, India, drooping roses: i love all t he new words this book introduces and the silliness of it all. this is a classic book i am willing to bet my son will STILL be able to recite from memory when he is in college

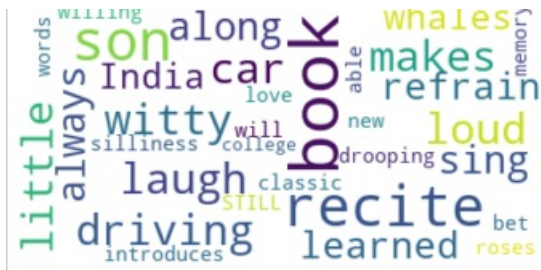


Review - 2  
 \*\*\*\*\*

Really Sendak Rosie Love grew books watching reading movie version paperbacks miss pages sontakes kind keep open hard incorporates seem two flimsy hands cover flimsy

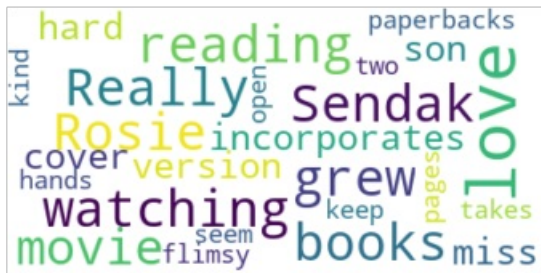
price  
Dr  
Smith Foster

Review - 1  
this witty little book makes my son laugh at loud. i recite it in the car as we're driving along and he always can sing the refrain. he's learned about whales, India, drooping roses: i love all the new words this book introduces and the silliness of it all. this is a classic book i am willing to bet my son will STILL be able to recite from memory when he is in college



Review - 2

I grew up reading these Sendak books, and watching the Really Rosie movie that incorporates them, and love them. My son loves them too. I do however, miss the hard cover version. The paperbacks seem kind of flimsy and it takes two hands to keep the pages open.



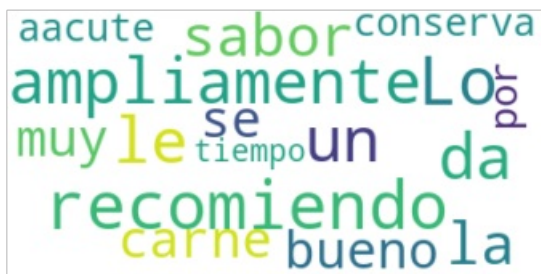
\*\*\*\*\*

Noisy samples are given the label  $\rightarrow -1$

\*\*\*\*\*

Review - 1

Lo recomiendo ampliamente, le da un sabor a la carne muy bueno, se conserva por m&acute;s tiempo.



```
##### DBSCAN with EPS = 2 #####
```

```
DBSCAN with EPS =2.00 and min pts = 100 the no of Clusters we get is = 1
```

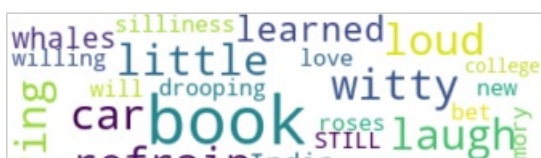
\*\*\*\*\*

Reviews of cluster -> 1

\*\*\*\*\*

Review - 1

review 1  
this witty little book makes my son laugh at loud. i recite it in the car as we're driving along a  
nd he always can sing the refrain. he's learned about whales, India, drooping roses: i love all t  
he new words this book introduces and the silliness of it all. this is a classic book i am  
willing to bet my son will STILL be able to recite from memory when he is in college





◀ ▶

## In [39]:

the shape of out text TF-IDF vectorizer (5000, 10429)

```
# TF-IDF weighted Word2Vec
tfidf_feat = tf_idf_vect.get_feature_names() # tfidf words/col-names
# final_tf_idf is the sparse matrix with row= sentence, col=word and cell_val = tfidf

data_tfidf_sent_vectors = []; # the tfidf-w2v for each sentence/review is stored in this list
row=0;
for sent in tqdm(list_of_sent): # for each review/sentence
    sent_vec = np.zeros(50) # as word vectors are of zero length
    weight_sum =0; # num of words with a valid vector in the sentence/review
    for word in sent: # for each word in a review/sentence
        if word in w2v_words:
            vec = data_w2v_model.wv[word]
            # tfidf = tf_idf_matrix[row, tfidf_feat.index(word)]
            # to reduce the computation we are
            # dictionary[word] = idf value of word in whole corpus
            # sent.count(word) = tf value of word in this review
            tf_idf = dictionary[word]*(sent.count(word)/len(sent))
            sent_vec += (vec * tf_idf)
            weight_sum += tf_idf
    if weight_sum != 0:
        sent_vec /= weight_sum
    data_tfidf_sent_vectors.append(sent_vec)
    row += 1

print(len(data_tfidf_sent_vectors))
print(len(data_tfidf_sent_vectors[0]))
```

5000  
50

## In [42]:

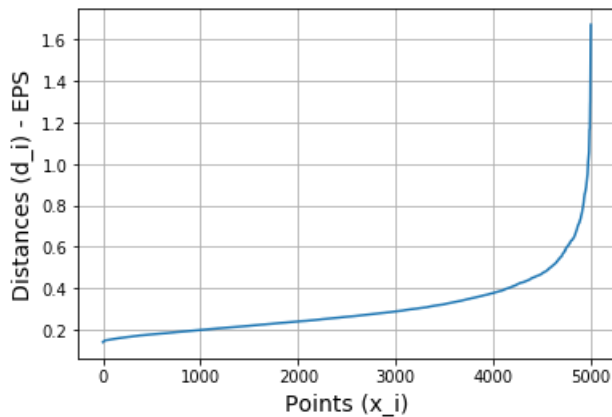
```
min_pts = 2*data_tfidf_sent_vectors[0].shape[0]
# Computing distances of nth-nearest neighbours
dist = n_neighbour(data_tfidf_sent_vectors,min_pts)
sorted_distance = np.sort(dist)

n_points = [i for i in range(len(data_tfidf_sent_vectors))]
plt.plot(n_points, sorted_distance)
```



```
plt.plot(n_points, sorted_distance,
plt.xlabel('Points (x_i)',size=14)
plt.ylabel('Distances (d_i) - EPS',size=14)
plt.title('Distances and Points Graph Plot\n',size=18)
plt.grid()
plt.show()
```

Distances and Points Graph Plot



## Implementing DBSCAN

So after getting best EPS lets try different nearby eps to see that the change in eps what will happen to our clusters

In [43]:

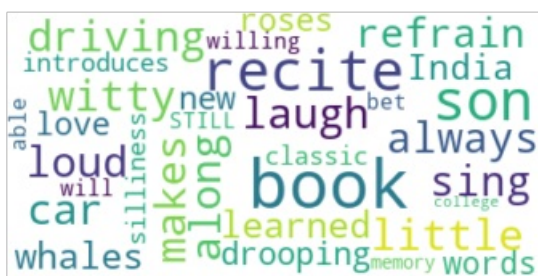
```
#So after getting best EPS lets try different nearby eps to see that the change in eps what will h
appen to our clusters
#Cluster labels for each point in the dataset given to fit(). Noisy samples are given the label -1
.
min_pts = 2*data_tfidf_sent_vectors[0].shape[0]
eps_list = [0.5,0.8,1.2,2,3]
dbscan_implement_diff_eps(eps_list,min_pts,data_tfidf_sent_vectors)
```

Note: Cluster labels for each point in the dataset given to fit(). Noisy samples are given the label -1.

```
##### DBSCAN with EPS = 0.5 #####
DBSCAN with EPS =0.50 and min_pts = 100 the no of Clusters we get is = 2
*****
```

```
Reviews of cluster -> 1
*****
```

Review - 1  
this witty little book makes my son laugh at loud. i recite it in the car as we're driving along and he always can sing the refrain. he's learned about whales, India, drooping roses: i love all the new words this book introduces and the silliness of it all. this is a classic book i am willing to bet my son will STILL be able to recite from memory when he is in college



Review - 2  
I grew up reading these Sendak books, and watching the Really Rosie movie that incorporates them, and love them. My son loves them too. I do however, miss the hard cover version. The paperbacks seem kind of flimsy and it takes two hands to keep the pages open.



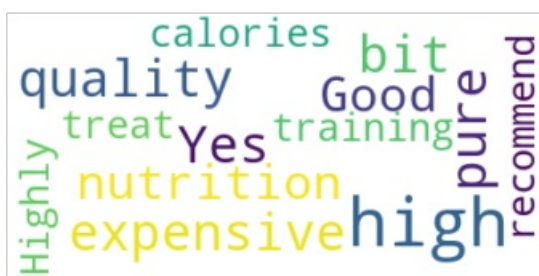
\*\*\*\*\*

Noisy samples are given the label -> -1

\*\*\*\*\*

Review - 1

Yes, they are a bit expensive but, they are high quality, and pure nutrition. Good training treat and not too high in calories. Highly recommend this!



Review - 2

Delishus, pleeze give mee more Gimborn Livur Treetts. Everee dog should havve these. I don't know aboutt kats, they probabbly don't deserve them. Kats skare me.... Anyway these treetts are delishus and yoo should buy manee buckets for yore dog. If yore dog doesn't want his bucket of livur treet s, I will take them. Thank yoo, love, Sonar.



##### DBSCAN with EPS = 0.8 #####

DBSCAN with EPS =0.80 and min\_pts = 100 the no of Clusters we get is = 2

\*\*\*\*\*

Reviews of cluster -> 1

\*\*\*\*\*

Review - 1

this witty little book makes my son laugh at loud. i recite it in the car as we're driving along a nd he always can sing the refrain. he's learned about whales, India, drooping roses: i love all t he new words this book introduces and the silliness of it all. this is a classic book i am willing to bet my son will STILL be able to recite from memory when he is in college





Review - 2

I grew up reading these Sendak books, and watching the Really Rosie movie that incorporates them, and love them. My son loves them too. I do however, miss the hard cover version. The paperbacks seem kind of flimsy and it takes two hands to keep the pages open.



\*\*\*\*\*

Noisy samples are given the label -> -1

\*\*\*\*\*

Review - 1

Yes, they are a bit expensive but, they are high quality, and pure nutrition. Good training treat and not too high in calories. Highly recommend this!



Review - 2

El producto me llegó; en una semana hasta Costa Rica, llegó; en buen estado y es de buena calidad, le quito una estrella ya que me llegó; un dinosaurio diferente al que ponen en la foto pero aun así estoy satisfecho con la compra.



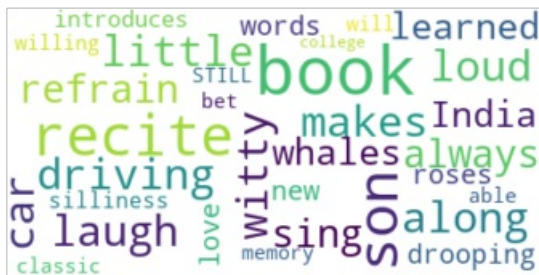


I grew up reading these Sendak books, and watching the Really Rosie movie that incorporates them, and love them. My son loves them too. I do however, miss the hard cover version. The paperbacks seem kind of flimsy and it takes two hands to keep the pages open.



```
Reviews of cluster -> 1
*****
```

this witty little book makes my son laugh at loud. i recite it in the car as we're driving along a nd he always can sing the refrain. he's learned about whales, India, drooping roses: i love all t he new words this book introduces and the silliness of it all. this is a classic book i am willing to bet my son will STILL be able to recite from memory when he is in college



I grew up reading these Sendak books, and watching the Really Rosie movie that incorporates them, and love them. My son loves them too. I do however, miss the hard cover version. The paperbacks seem kind of flimsy and it takes two hands to keep the pages open.

