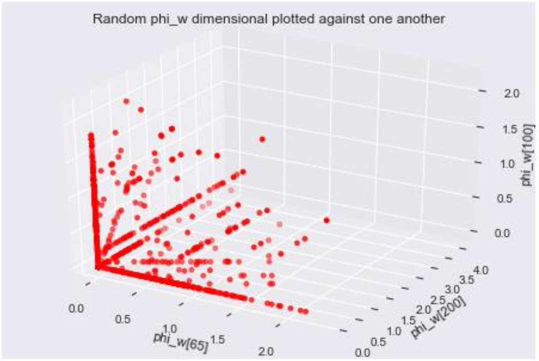
Orysya Stus
28 May 2017

# Homework 4: Embedding of Words

## I. A description of your 100-dimentional embedding

The Brown corpus contains 1,161,192 separate strings. Following removal of punctuation (using string.punctuation) and stop words (using nltk.corpus.stopwords), 53,991 non-distinct words remain. The word list was fed through a frequency counter (function freq_counter) to extract the most frequent 5000 (V) and 1000 (C) words. Pr(c|w), Pr(c), and phi_w were computed using the documentation provided.

Both PCA and Isomap manifold learning were tested, PCA works for linear dimensional embedding while Isomap works for non-linear dimensional embedding. PCA(100) resulted in 22.52% of the variance explained and 3.504e-11 as the sum of residuals, while Isomap(100) resulted in a reconstruction_error of 30.37. The resulting performance metrics could not be compared directly to one another, therefore the transformed embedded matrixes were run through the same KMeans and NearestNeighbor algorithms, finding PCA(100) to be superior. PCA(100) resulted in more meaningful, equally distributed clusters and more meaningful nearest neighbors (ie. PCA: utopian nearest neighbor to communism vs. reading as nearest neighbor for communism), both of which will be discussed. Furthermore, mapping random attributes of the pre-transformed data showed that geodesic distances were not characteristic of the data, but rather linear relationships between features were seen. Therefore, PCA is an appropriate, less computationally expensive (1.11 s to fit PCA vs. 5min 1s to fit Isomap) dimensional embedding.



## II. Nearest neighbor results

Using the PCA(100) transformed phi_w matrix, the NearestNeighbors(n_neighbors=1, algorithm='brute', metric='cosine') unsupervised learning algorithm was used in order to find the nearest neighbor for 25 words. The results are as follows:

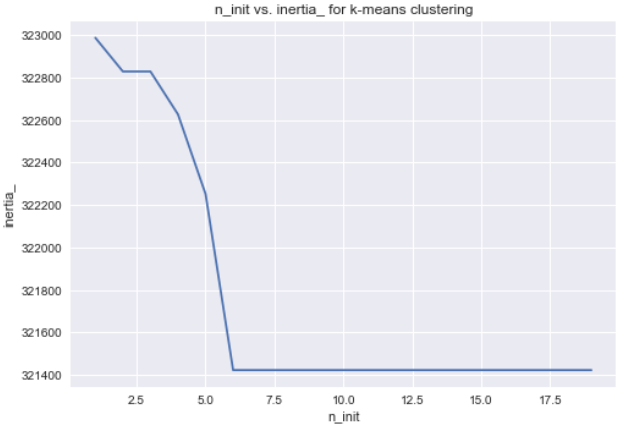
```
Word: communism -- Nearest neighbor: utopian -- Distance: 0.618788623478
Word: autumn -- Nearest neighbor: storm -- Distance: 0.515481120469
Word: cigarette -- Nearest neighbor: bullet -- Distance: 0.509093821293
Word: pulmonary -- Nearest neighbor: artery -- Distance: 0.256579170491
Word: mankind -- Nearest neighbor: world -- Distance: 0.538521909205
Word: africa -- Nearest neighbor: asia -- Distance: 0.358373805583
Word: chicago -- Nearest neighbor: portland -- Distance: 0.483564007709
Word: revolution -- Nearest neighbor: modern -- Distance: 0.633185238013
Word: september -- Nearest neighbor: july -- Distance: 0.233969560906
Word: chemical -- Nearest neighbor: drugs -- Distance: 0.476249972519
Word: detergent -- Nearest neighbor: fabrics -- Distance: 0.4881330922
Word: dictionary -- Nearest neighbor: text -- Distance: 0.276896053425
Word: storm -- Nearest neighbor: saturday -- Distance: 0.515346722166
Word: worship -- Nearest neighbor: christian -- Distance: 0.498514492007
Word: money -- Nearest neighbor: pay -- Distance: 0.450026079941
Word: red -- Nearest neighbor: hair -- Distance: 0.369553830758
Word: vacation -- Nearest neighbor: time -- Distance: 0.516709200467
Word: missile -- Nearest neighbor: submarines -- Distance: 0.521776137033
```

```
Word: player -- Nearest neighbor: palmer -- Distance: 0.381354486192
Word: velocity -- Nearest neighbor: fluid -- Distance: 0.500602222468
Word: reality -- Nearest neighbor: human -- Distance: 0.505323261859
Word: education -- Nearest neighbor: public -- Distance: 0.282687141521
Word: strong -- Nearest neighbor: enough -- Distance: 0.524498957838
Word: churches -- Nearest neighbor: members -- Distance: 0.509190852885
Word: world -- Nearest neighbor: war -- Distance: 0.403722008098
```

The algorithm returned the nearest neighbor and the distance between the two words using cosine distance. Most the results make sense, demonstrating that the NearestNeighbor algorithm found relevant neighboring words.

**III. Clustering**

KMeans(n_init=15, n_clusters=100, max_iter= 1000, init='k-means++', random_state=0) was used to cluster the words in V into 100 groups. The method of initialization chosen was k-means++, since it initialized the cluster seeds by means of choosing outliers as cluster centroids thus yielding better results. Describing the PCA transformed phi_w (showing count, mean, standard deviation, etc.) shows that outliers exist, thus k-means++ is appropriate. The algorithm used was "elkan" since the PCA transformed phi_w is dense. Furthermore, 6 different centroid seeds were used since there were no changes to inertia seen at higher n_init values. The clusters were reviewed and the following clusters analyzed:



```
This cluster is associated with scientific studies.
 ['information', 'study', 'data', 'results', 'methods', 'reaction',
'described', 'studies', 'cells', 'selected']

This cluster is associated with numeric values or measurement metrics.
 ['two', 'years', 'three', 'several', 'four', 'five', 'ago', 'six',
'minutes', 'miles', 'hundred', 'ten', 'couple', 'seven', 'eight', 'dollars',
'thousand', 'nine', 'twenty', 'fifty', 'thirty', 'fifteen', 'twelve',
'eleven', 'forty', 'fourteen']

This cluster is associated with predominanatly with relationship status.
 ['man', 'old', 'young', 'wife', 'mother', 'father', 'son', 'friend', 'met',
'husband', 'lived', 'poor', 'hospital', 'married', 'jack', 'spoke', 'died',
'captain', 'named', 'remembered', 'lady', 'murder', 'brother', 'daughter',
'mercer', 'smiled', 'sweet', 'fellow', 'baby', 'wilson', 'talked', 'lewis',
'wondered', 'fathers', 'uncle', 'alive', 'loved', 'joe', 'wished', 'dear',
'alfred', 'warren', 'cousin', 'sick', 'lucy', 'younger', 'adam', 'lawyer',
'anne', 'kate', 'papa', 'handed', 'thompson', 'sister', 'harry', 'bride',
'johnnie', 'blanche', 'aunt']

This cluster is associated with government associated terminology.
 ['program', 'national', 'education', 'defense', 'medical', 'aid',
'planning', 'activities', 'assistance', 'educational', 'policies',
'longterm']
```

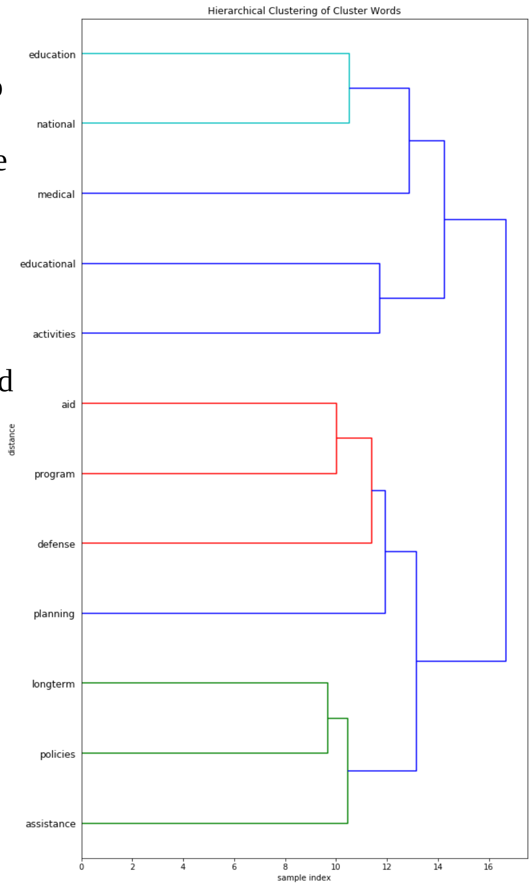
```
This cluster is associated with economic terminology.
 ['tax', 'pay', 'paid', 'sales', 'income', 'rates', 'share', 'annual',
'workers', 'capital', 'gain', 'increases', 'du', 'estimated', 'employees',
'gross', 'sets', 'rising', 'wage', 'vehicles', 'bills', 'raise', 'expense',
'extra', 'bonds', 'insurance', 'dollar', 'shares', 'percentage', 'taxes',
'load', 'excess', 'wages', 'spending', 'estimate', 'consumer', 'license',
'retired', 'dealers', 'adjustment', 'producing', 'net', 'adjusted',
'household', 'reducing', 'builders', 'decline', 'buying', 'utility',
'proportion', 'customer', 'revenues', 'marginal', 'allowances', 'dealer',
'prospects', 'monthly', 'saving', 'retail', 'stocks', 'earnings']
```

The clustering performed produced coherent results. K-means clustering assumes that the cluster is spherical based on convergence, while EM soft assigns a point to clusters (giving a probability of any point belonging to any centroid). The simpler model, K-means, was chosen for modeling, but EM could be used as a comparison of cluster consistency.

## IV. Supplementary: Hierarchical Clustering of an Individual Cluster

The words belonging to the clustering associated with government associate terminology were fed into scipy.cluster.hierarchy.linkage(subset, 'ward') to perform agglomerative/hierarchical clustering using ‘ward’ which minimized variance in increments. The goal of hierarchical clustering is to visualize the association of each word and their associated similarities within the clusters.

Here, John Firth’s idea that “You shall know a word by the company it keeps” is visualized and confirmed as each word is incrementally clusters next to its nearest neighbor recursively.



## IV. Summary

Using PCA 100-dimensional embedding appropriately projects the phi_w data to an appropriate 100-dimensional space, using k-means clusters V into meaningful clusters which a domain expert can classify, using nearest neighbors locates the nearest neighbor based on cosine distance, and using hierarchical clustering demonstrates the incremental clustering of a kmeans cluster for further vocabulary understanding.