

# Clustering

May 13, 2018

## 1 Clustering

### 1.1 Clustering Task

- Use Affinity Propagation from [Clustering.jl](#), to cluster word2vec word embeddings, according to meaning.
- Done right this will separate locations from sports
- Done finely and it will separate ball-sports from other sports, and will separate locations according to regions, etc
- Affinity propagation requires a similarity matrix, which you can set as a negated distance matrix.
- For this you'll also want [Distances.jl](#) for all your distance metric needs.
- It is traditional with word2vec to use cosine distance.
- You will also need to set each item's availability. This is the diagonal of the similarity matrix. Decreasing it roughly corresponds to decreasing the amount each node wants to be in a cluster on its own.

## 2 First we load up some data

For the example presented here, we will use a subset of Word Embedding, trained using [Word2Vec.jl](#). These are 100 dimensional vectors, which encode syntactic and semantic information about words.

You can download the dataset from [here](#), and load it up with [JLD](#) as shown below. (or just load it directly if you have cloned the notebooks)

```
In [ ]: using JLD
         embeddings = load("../assets/ClusteringAndDimensionalityReduction.jld", "em
```