Clustering

September 11, 2018

1 Clustering

1.1 Basic Clustering Task

Use the following dataset:

```
In [24]: # Pkg.add("RDatasets")
         using RDatasets
         iris = dataset("datasets", "iris")
```

```
INFO: Package RDatasets is already installedINFO: METADATA is out-of-date  you may not have th
```

```
Out[24]: 150Œ5 DataFrames.DataFrame
          Row  SepalLength  SepalWidth  PetalLength  PetalWidth  Species

          1    5.1          3.5         1.4          0.2         "setosa"
          2    4.9          3.0         1.4          0.2         "setosa"
          3    4.7          3.2         1.3          0.2         "setosa"
          4    4.6          3.1         1.5          0.2         "setosa"
          5    5.0          3.6         1.4          0.2         "setosa"
          6    5.4          3.9         1.7          0.4         "setosa"
          7    4.6          3.4         1.4          0.3         "setosa"
          8    5.0          3.4         1.5          0.2         "setosa"
          9    4.4          2.9         1.4          0.2         "setosa"
          10   4.9          3.1         1.5          0.1         "setosa"
          11   5.4          3.7         1.5          0.2         "setosa"

          139  6.0          3.0         4.8          1.8         "virginica"
          140  6.9          3.1         5.4          2.1         "virginica"
          141  6.7          3.1         5.6          2.4         "virginica"
          142  6.9          3.1         5.1          2.3         "virginica"
          143  5.8          2.7         5.1          1.9         "virginica"
          144  6.8          3.2         5.9          2.3         "virginica"
          145  6.7          3.3         5.7          2.5         "virginica"
          146  6.7          3.0         5.2          2.3         "virginica"
          147  6.3          2.5         5.0          1.9         "virginica"
          148  6.5          3.0         5.2          2.0         "virginica"
          149  6.2          3.4         5.4          2.3         "virginica"
          150  5.9          3.0         5.1          1.8         "virginica"
```

Use Clustering.jl to cluster using the `SepalLength`, `PetalLength`, and `PetalWidth`features via K-means clustering. Make a scatter plot of the resulting clusters.

Hint: You will need to index the dataframe, convert it to an array, and transpose it. In addition, you will need to use the `assignments` field of the return to get the cluster assignments.

1.2 Advanced Clustering Task

For the the example presented here, we will use a subhset of Word Embedding, trained using Word2Vec.jl. These are 100 dimentional vectors, which encode syntactic and semantic information about words.

```
In [ ]: using Embeddings
        countries = ["Afghanistan", "Algeria", "Angola", "Arabia", "Argentina", "Australia", "
        usa_cities = ["Albuquerque", "Atlanta", "Austin", "Baltimore", "Boston", "Charlotte", "
        world_capitals = ["Accra", "Algiers", "Amman", "Ankara", "Antananarivo", "Athens", "Bag
        animals = ["alpaca","camel","cattle","dog","dove","duck","ferret","goldfish","goose","
        sports = ["archery","badminton","basketball","boxing","cycling","diving","equestrian","

        words_by_class = [countries, usa_cities, world_capitals, animals, sports]
        all_words = reduce(vcat, words_by_class)
        embedding_table = load_embeddings(Word2Vec; keep_words = all_words)
        @assert Set(all_words) == Set(embedding_table.vocab)

        embeddings = embedding_table.embeddings
        all_words = embedding_table.vocab
        classes = map(all_words) do word
            findfirst(col -> word  col, [countries, usa_cities, world_capitals, animals, sport
        end;
```

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

- Use Affinity Propagraion from Clustering.jl, to cluster word2vec word embeddings, according to meaning.
- Done right this will seperate locations from sports
- Done finely and it will seperate ball-sports from other sports, and will seperate locations according to regions, etc
- Affinity propagraion 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 as 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.