Category learning based on features extracted by fastText model

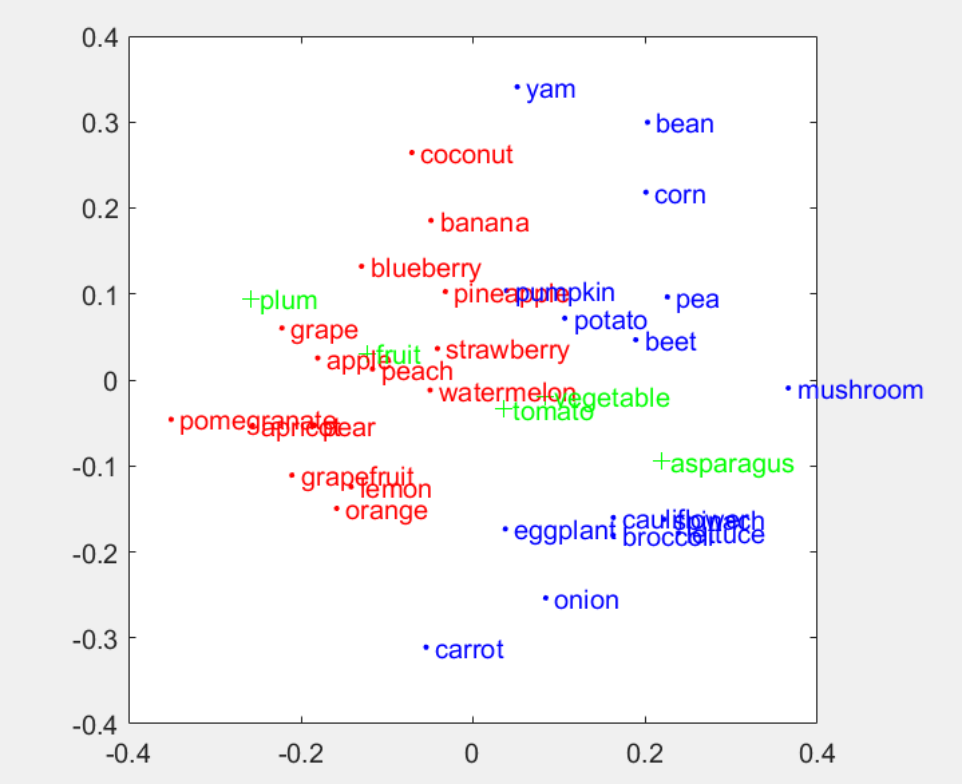
We used the fastText model (developed by Facebook research group, now included in MATLAB Text Analytics Toolbox) to get word embeddings with 300-dim features for 15 fruit words (including 'orange';'apple';'banana';'peach';'pear';'apricot';lemon;'grape';'strawberry';'grapefruit'; 'pineapple';'blueberry';'watermelon'; 'pomegranate';'coconut'), 15 vegetable words (including 'pea';'carrot';'bean';'spinach';'broccoli';pumpkin';'corn'; cauliflower';'lettuce'; 'beet';'eggplant'; 'onion';'potato';'yam'; 'mushroom'), and 5 more words for categorization task (including "plum", "tomato", "asparagus", "fruit", 'vegetable").

Load the input file **wordsinput.mat**. You will get the name list in the cell-array "wordlist", and word embeddings saved in the matrix "wordvec" with the size of 30 by 300. Rows correspond to words, and columns are 300 embedding features. Load the file **testwordsinput.mat** to get the input for the five test words with the name list in "testwordlist" and embeddings in "testwordvec".

1) Run MDS algorithm. Compute the distance matrix based on the provided word embedding inputs. Use the matlab function pdist() to compute pairwise cosine distances. Take the computed distance matrix (with size of 35 by 35) as the input to run nonmetric multidimensional scaling method and compute the coordinates for each object in a 2-dimensional space. Plot all the objects in the 2D space computed by MDS.

2) Category learning: compute P(X|category) for the fruit and vegetable categories based on provided exemplars. Use the MDS-calculated 2D coordinates as the feature representation of each object for inputs. Implement a categorization model with the parametric method of prototype theory. Use 15 exemplars for fruit category and 15 exemplars for vegetable category. [Hint: you may find matlab function mean(), cov() useful].

3) Categorization task: compute the probability of new objects belonging to the fruit category. The test words are "plum", "tomato", "asparagus", "fruit", 'vegetable". Use the learned category representations to determine the probability that each test word belongs to the fruit category. We assume the prior probabilities are 0.5 for the fruit and the vegetable category.




```
>> wordembeddingEx

testwordlist =

  1×5 cell array

    {'plum'}    {'tomato'}    {'asparagus'}    {'fruit'}    {'vegetable'}


postprobcat1 =

    0.9993    0.1196    0.0000    0.9878    0.0211
```