Tyler Larsen
April 24, 2018
Data- X Homework 04

Part 1: Entropy

1. The best feature to do the first split is HasFamily. If the individual has a family (HasFamily=1), there is a 75% chance they are not a defaulter (Defaulter=0), while if they do not have a family (HasFamily=0), there is a 75% chance they are a defaulter (Defaulter=1). This split maximizes information gain.

HasJob		
HasJob=1	P(Defaulter=1)	$\frac{2}{5}$
HasJob=1	P(Defaulter=0)	$\frac{3}{5}$
HasJob=0	P(Defaulter=1)	$\frac{2}{3}$
HasJob=0	P(Defaulter=0)	$\frac{1}{3}$

HasFamily		
HasFamily=1	P(Defaulter=1)	$\frac{1}{4}$
HasFamily=1	P(Defaulter=0)	$\frac{3}{4}$
HasFamily=0	P(Defaulter=1)	$\frac{3}{4}$
HasFamily=0	P(Defaulter=0)	$\frac{1}{4}$

IsAbove30years		
IsAbove30years=1	P(Defaulter=1)	$\frac{1}{2}$
IsAbove30years=1	P(Defaulter=0)	$\frac{1}{2}$
IsAbove30years=0	P(Defaulter=1)	$\frac{1}{2}$
IsAbove30years=0	P(Defaulter=0)	$\frac{1}{2}$

2. The entropy of S is 1.568. According to the Source Coding Theorem, 1.568 is the smallest codeword length that is theoretically possible for signal S.

$$H(S) = \sum_x p(x) \log_2 \left(\frac{1}{p(x)} \right),$$
$$H(S) = p(A) \log_2 \left(\frac{1}{p(A)} \right) + p(B) \log_2 \left(\frac{1}{p(B)} \right) + p(C) \log_2 \left(\frac{1}{p(x)} \right)$$
$$H(S) = 0.7 \log_2(7) + 0.2 \log_2(2) + 0.1 \log_2(1)$$
$$H(S) = 1.568$$

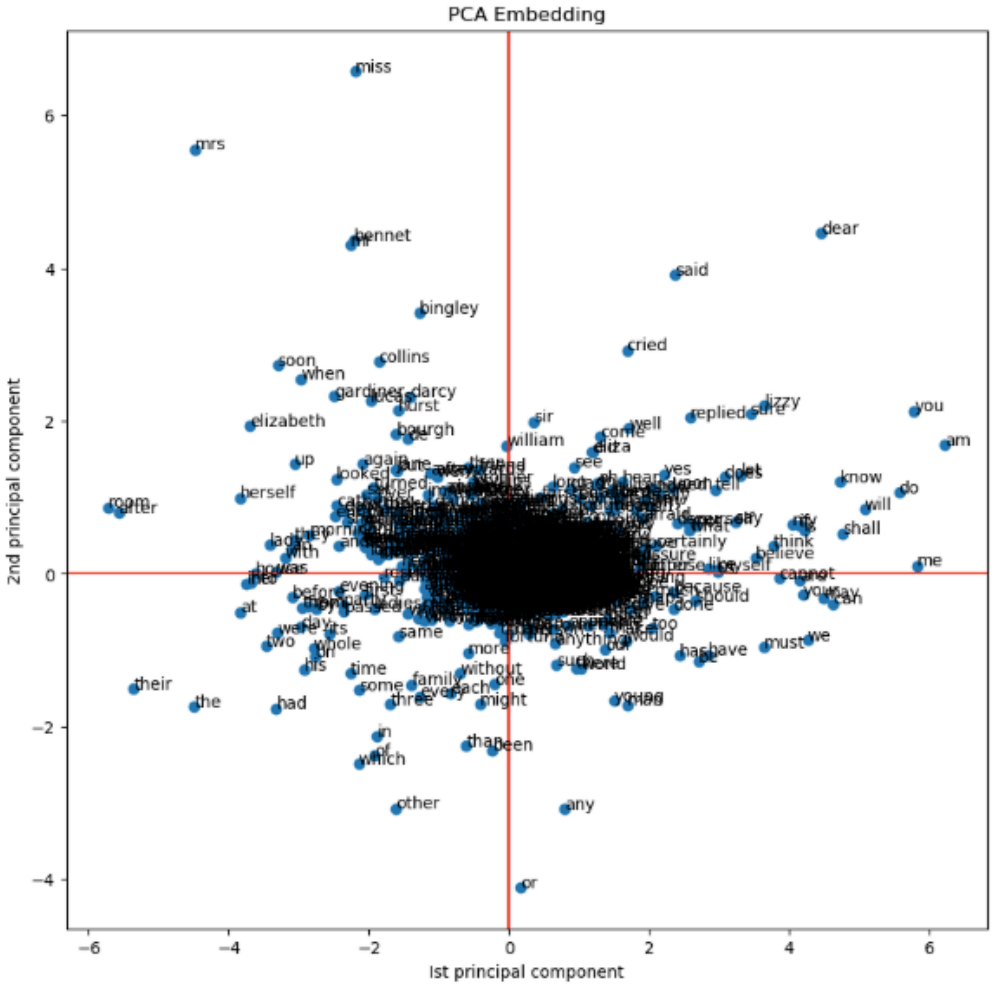
Part 2: Natural Language Processing

1. A Bag of Words model in NLP is a model in which the entire corpus is tokenized, and then each word's respective count is recorded. For a corpus of n words, a basic Bag of Words model will create an n by 2 matrix. Each new corpus added will add one additional column. A Word2vec model, on the other hand, is a neural network approach to NLP. It takes a corpus of words as an input, produces a matrix assigning each word a unique vector, and if the CBOW model is being used it will predict the next word given the previous words, and if a skip gram model is being used it will attempt to predict the context, given the previous words. A major disadvantage of the Bag of Words model is that context, or the ordering of words, does not affect the prediction. An advantage of Word2Vec over Bag of Words is that it is highly scalable, it understands word

positioning, syntax, and semantics, and it takes into account the context in which the word is in.

2. A word vector is a word encoding in which the column corresponding to the word is set to one and the rest of the elements are set to zero. Word embeddings Embedding
3. In NLP, a corpus is a piece of text that is inputted into the NLP model. The vocabulary of the model is different than that of the corpus because the model only contains the vocabulary of the corpus it was trained on, and so the vocabulary of the testing corpus may be different than that of the model.
4. See Jupyter Notebook for code.
    a. Preprocessing/modeling technique: For all of my data preprocessing, I used the self defined function review_cleaner, much the same was as we did in lecture. First, I removed the punctuation from each sentence. Then, I tokenized the words, and converted them to lowercase. Then, I set stopwords using the stopwords package from NLTK. After taking these pre-processing steps, I looped through all the words in each review. For each word in the corpus, I then lemmatized and stemmed all the words that were not part of the English stopwords. After creating this function, I looped through every sentence in the corpus and applied review_cleaner to each sentence, and added all of the resulting lists to a larger list of lists. Next, I used the gensim library's models.Word2Vec function to create a Word2Vec model. For my hyperparameters, I set size equal to 300, window equal to 10, min_count equal to 5, workers equal to 10, and iter equal to 5.
    b. Reporting
        i. Vocabulary Count: 435
        ii. Embedding Size: 300
        iii. Number of Training Iterations: 10

c. Visualization



This plot visualizes the relationship among words in the entire corpus. The closer the words are together on the plot, more similar they are. For example, in the top left quadrant we see the words miss, mrs, Elizabeth, mom, and herself. In the top right quadrant know and shall are right next to each other, me and am are next to eachother, think and believe are next to each other, and so on.

d. Intrinsic Evaluation Results
   i. Similarity (brother, sister): 0.8668104698432741
   ii. Similarity (happy, sad): 0.807388939816033
   iii. Similarity (pride, prejudice): 0.9725077469861934
   iv. Similarity (house, home): 0.856531421484642
   v. Similarity (street, road): 0.9200372061877513

Part 3: Simple SQL Queries

1. Simple SELECTS

Num.	Query	Output
1	SELECT * FROM parents;	abraham|barack abraham|clinton delano|herbert eisenhower|fillmore fillmore|abraham fillmore|delano fillmore|grover
2	SELECT child, parent FROM parents WHERE parent="abraham";	barack|abraham clinton|abraham
3	SELECT child FROM parents WHERE child LIKE '%e%';	herbert fillmore delano grover
4	SELECT DISTINCT parent FROM parents ORDER BY parent DESC;	fillmore eisenhower delano abraham
5	SELECT a.child, b.child FROM parents a, parents b WHERE a.parent = b.parent AND a.child < b.child;	barack|clinton abraham|delano abraham|grover delano|grover

2. Joins

Num.	Query	Output
1	SELECT COUNT(*) FROM dogs WHERE fur="curly";	2
2	SELECT parents.parent as child FROM parents JOIN dogs ON child = name WHERE dogs.fur='curly';	eisenhower delano
3	SELECT a.parent, b.child FROM (SELECT parents.parent, dogs.fur FROM parents LEFT JOIN dogs ON parents.parent = dogs.name) as a, (SELECT parents.child, parents.parent, dogs.fur FROM parents JOIN dogs ON parents.child = dogs.name) as b WHERE a.parent = b.parent AND a.fur = b.fur	abraham|clinton

	GROUP BY a.parent;	

3.

Num.	Query	Output
1	SELECT kind, MIN(weight) FROM animals;	parrot|6
2	SELECT AVG(legs), AVG(weight) from animals;	3.0|2009.3333333
3	SELECT kind FROM animals WHERE legs > 2 AND weight < 20;	cat ferret
4	SELECT AVG(weight) FROM animals GROUP BY legs;	4005.333333333 13.33333333333