A Project Report On

Personalized Medicine

Redefining Cancer Treatment

By
Garima Singh

TABLE OF CONTENT

S. No	Graph Name	Page no.
1.	Problem Statement	4
2.	Data Used	(4-5)
2.1	File Descriptions	4
3.	Exploration of Variables	5
4	Text Mining Process	5
5	Visualizing Data	(5-11)
5.1	Most frequent words in Text	6
5.2	Class wise most frequent Terms	7
6.	Model Building	11
7.	Result	(11-12)
7.1	Overall Statistics	11
7.2	Statistics by Class	12

LIST OF GRAPHS

S. No	Graph Name	Page no.
(a)	Plot for Word and Frequency	7
(b)	Plot for Word and Frequency for Class 1	7
(c)	Plot for Word and Frequency for Class 2	8
(d)	Plot for Word and Frequency for Class 3	8
(e)	Plot for Word and Frequency for Class 4	8
(f)	Plot for Word and Frequency for Class 5	9
(g)	Plot for Word and Frequency for Class 6	9
(h)	Plot for Word and Frequency for Class 7	10
(i)	Plot for Word and Frequency for Class 8	10
(j)	Plot for Word and Frequency for Class 9	11

LIST OF FIGURES

S. No	Figure Name	Page no.
(a)	Entity Relationship Diagram	4
(b)	Word Chart displaying most common terms	6

LIST OF TABLES

S. No	Table Name	Page no.
(a)	Confusion Marix for XGBoost Model	12

1. Problem Statement

During the past several years it has been found that how precision medicine and, more concretely, how genetic testing is going to disrupt the way diseases like cancer are treated. But this is only partially happening due to the huge amount of manual work still required. Once sequenced, a cancer tumor can have thousands of genetic mutations. But the challenge is distinguishing the mutations that contribute to tumor growth (drivers) from the neutral mutations (passengers).

Currently this interpretation of genetic mutations is being done manually. This is a very time-consuming task where a clinical pathologist has to manually review and classify every single genetic mutation based on evidence from text-based clinical literature.

Memorial Sloan Kettering Cancer Center (MSKCC) has made available an expert-annotated knowledge base where world-class researchers and oncologists have manually annotated thousands of mutations. We need to develop a Machine Learning algorithm that, using this knowledge base as a baseline, automatically classifies genetic variations based on clinical evidence (text).

2. Data Used

There are nine different classes a genetic mutation can be classified on. Both, training and test, data sets are provided via two different files. One (training/test_variants) provides the information about the genetic mutations, whereas the other (training/test_text) provides the clinical evidence (text) that our human experts used to classify the genetic mutations. Both are linked via the ID field.

Therefore the genetic mutation (row) with ID=15 in the file training_variants, was classified using the clinical evidence (text) from the row with ID=15 in the file training_text.

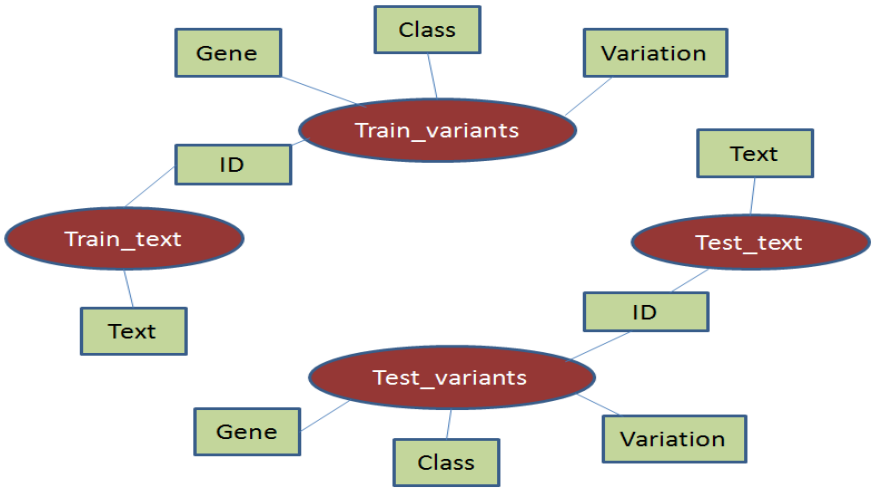


Figure (a) : Entity Relationship Diagram

2.1 File Descriptions

- **training_variants** - a comma separated file containing the description of the genetic mutations used for training.

- **training_text -** a double pipe (||) delimited file that contains the clinical evidence (text) used to classify genetic mutations.
- **test_variants -** a comma separated file containing the description of the genetic mutations used for training.
- **test_text -** a double pipe (||) delimited file that contains the clinical evidence (text) used to classify genetic mutations.

3. Exploration of Variables

- ID : the id of the row used to link the mutation to the clinical evidence
- Gene : the gene where this genetic mutation is located
- Variation : the aminoacid change for this mutations
- Class : 1-9 the class this genetic mutation has been classified on
- Text : the clinical evidence used to classify the genetic mutation

4. Text Mining Process

Text Mining / Natural Language Processing helps computers to understand text and derive useful information from it. It consists of pre-defined set of commands used to clean the data. Since, text mining is mainly used to verify sentiments, the incoming data can be loosely structured, multilingual, textual or might have poor spellings.

Techniques used in text mining are:

- **Bag of Words** : This techniques creates a 'bag' or group of words by counting the number of times each word has appear and use these counts as independent variables.
- **Change the text case** : Data is often received in irregular formats. For example: 'CyCLe' & 'cycle'. Both means the same thing but is represented in an irregular manner. Hence, it is advisable to change the case of text. Either to upper or lower case.
- **Deal with Punctuation** : This can be tricky at times. Your tool(R or Python) would read 'data mining' & 'data-mining' as two different words. But they are same. Hence, we should remove the punctuation elements also.
- **Remove Stopwords** : Stopwords are nothing but the words which add no value to text. They don't describe any sentiment. Examples are 'i','me','myself','they','them' and many more. Hence, we should remove such words too. In addition to stopwords, you may find other words which are repeated but add no value. Remove them as well.
- **Stemming or Lemmatization** : This suggests bringing a word back to its root. It is generally used of words which are similar but only differ by tenses. For example: 'play', 'playing' and 'played' can be stemmed into one word 'play', since all three connotes the same action.
- **Remove White Spaces** : Remove extra white spaces.
- **Remove Numerical Values** : We remove the numbers and only process the character strings
- **Create Document Term Matrix** : After processing the text, we now create a Document term matrix to check which all words have how many occurrences/frequencies.
- **Remove Sparse Items** : remove the less frequent data (sparse data)

5. Visualizing Data

Now let's visualize the data based on most frequent terms in the whole text and based on class level distinction.

5.1. **Most frequent words in Text**

From the wordcloud (figure (a) and (b)), it can be seen that the most valid common/frequent words in the text are mutat, cell, cancer, tumor, activ, gene, patient, protein, analysis, bind

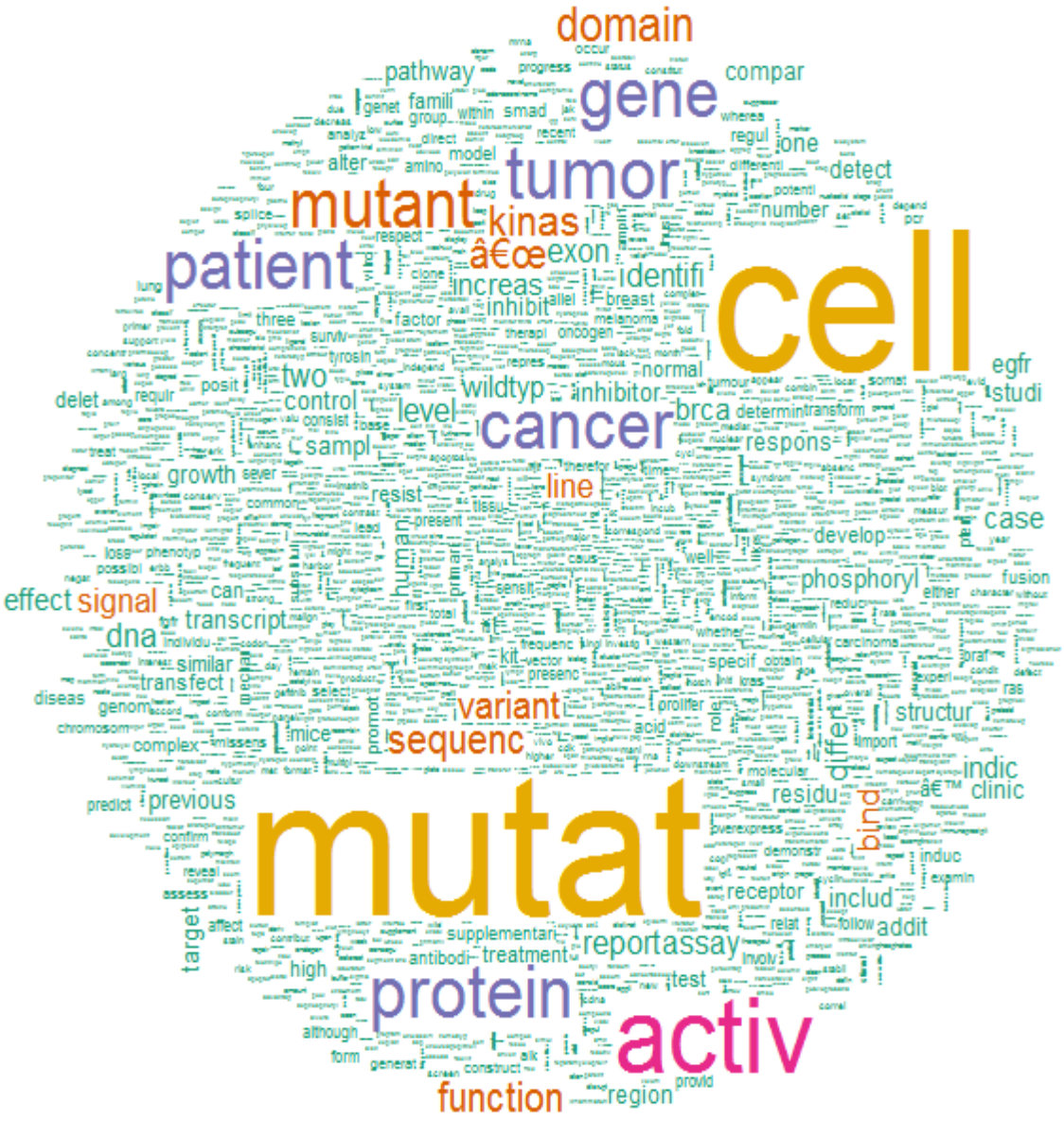
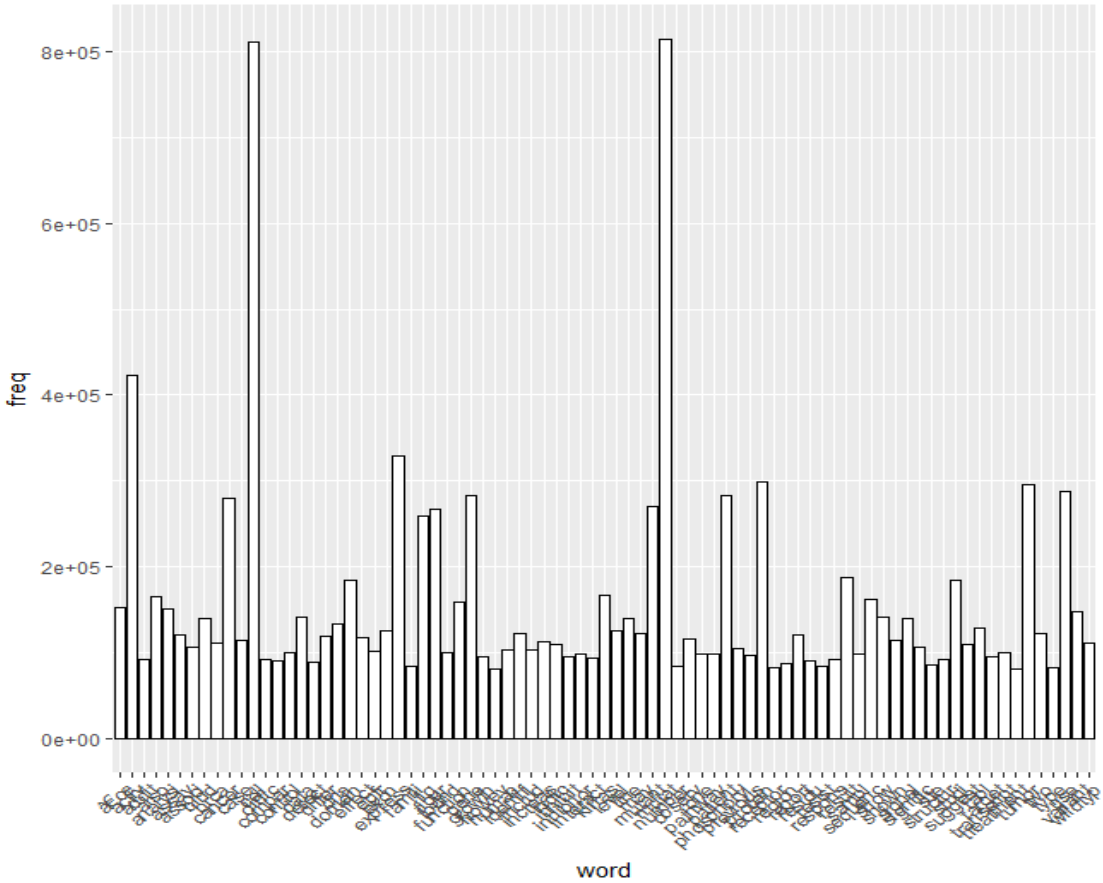
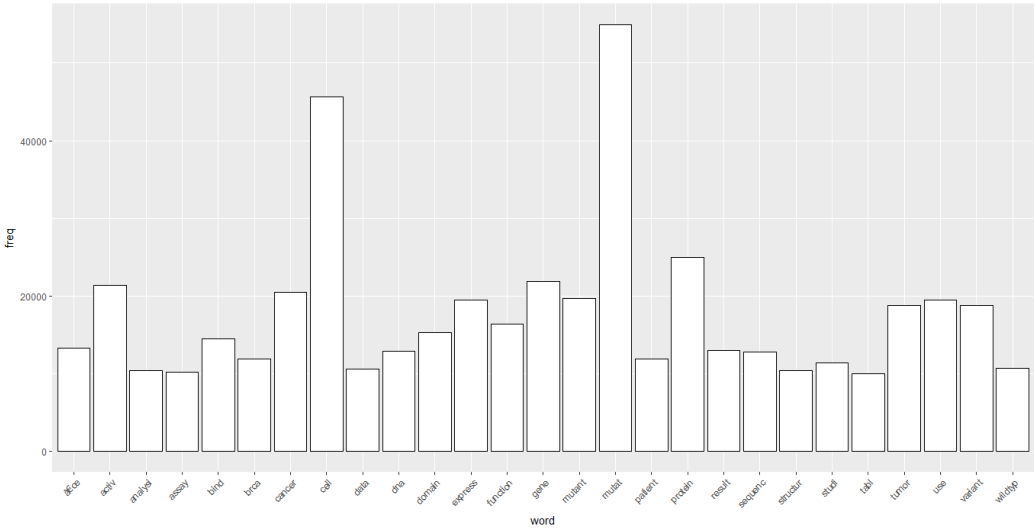


Figure (b) : Word chart displaying most common terms

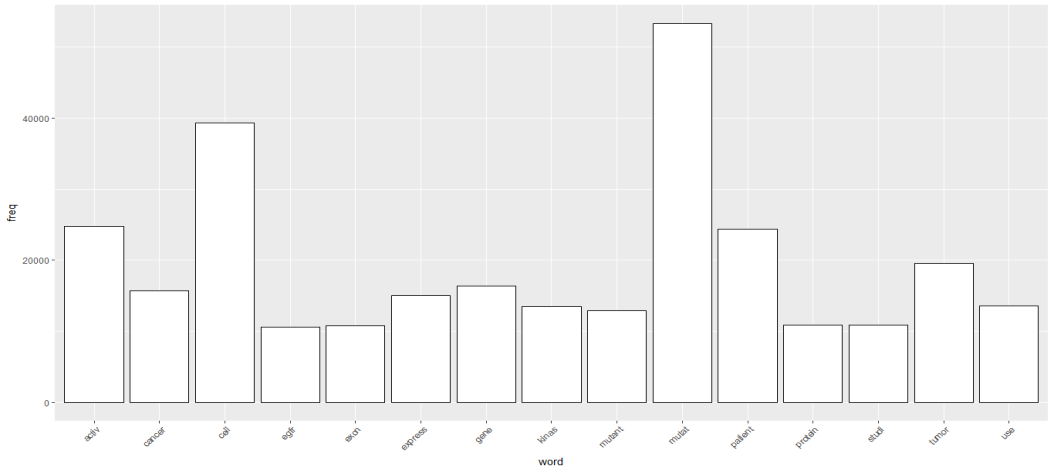


Graph (a) : Plot for word and frequency

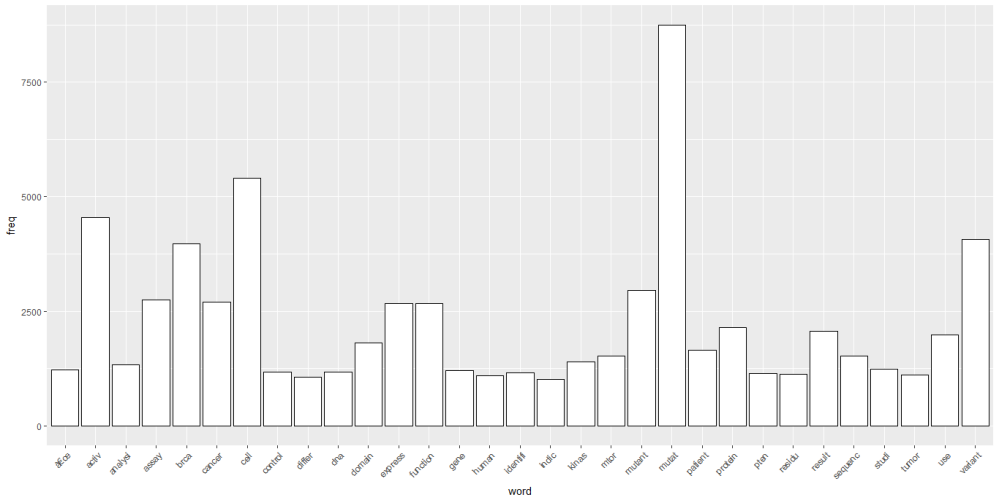
5.2 Class wise most frequent terms



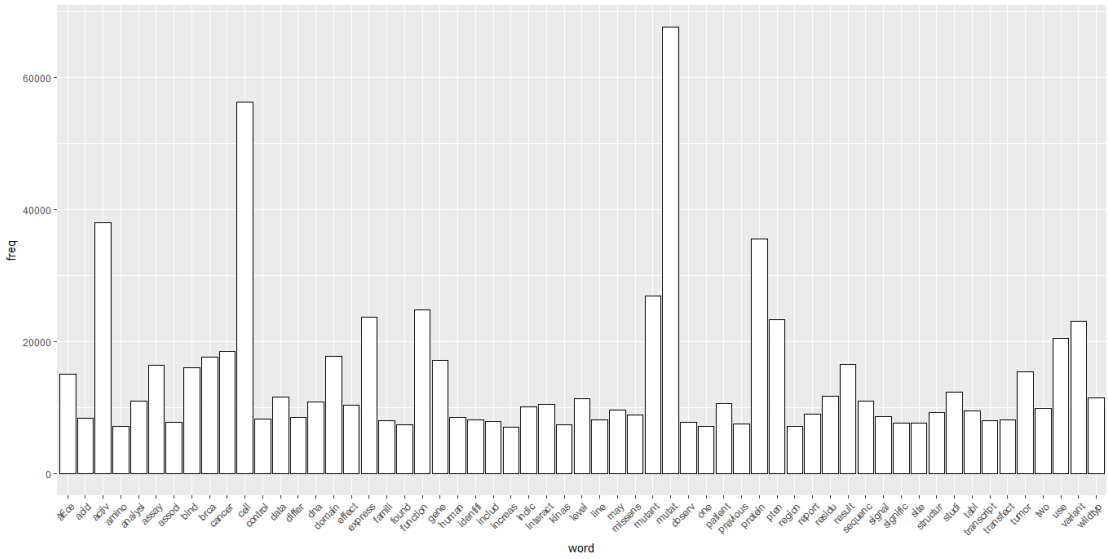
Graph (b) : Plot for word and frequency for Class 1



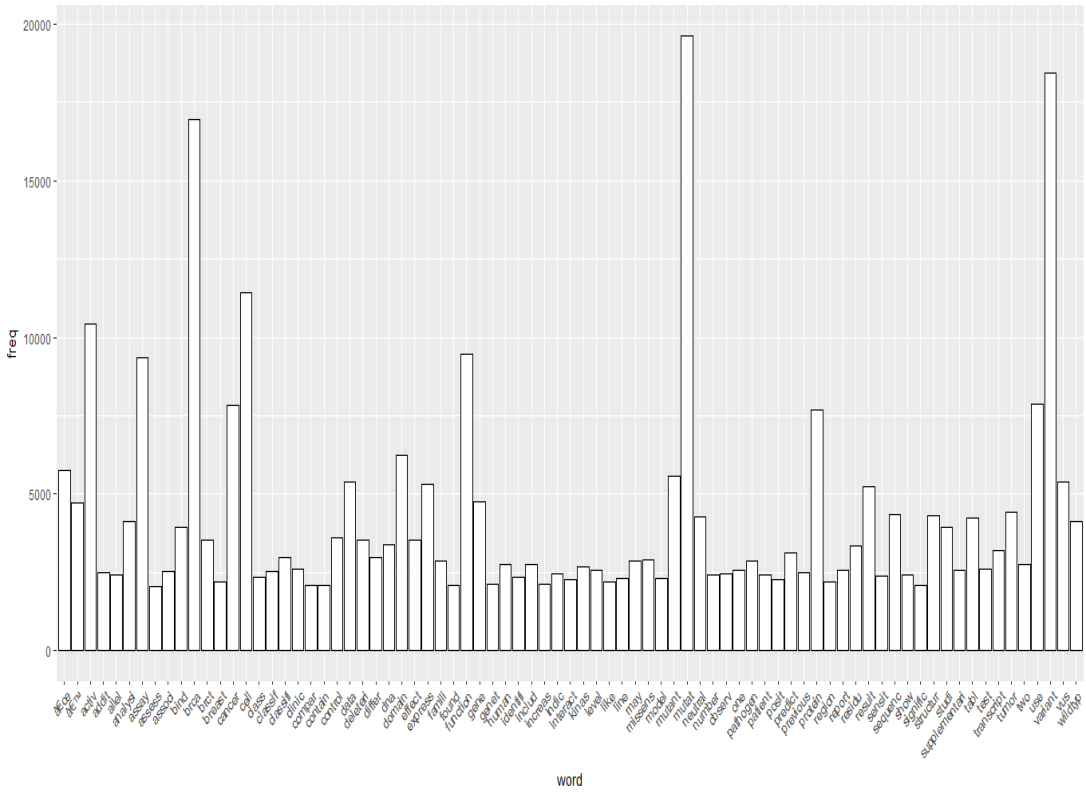
Graph (c) : Plot for word and frequency for Class 2



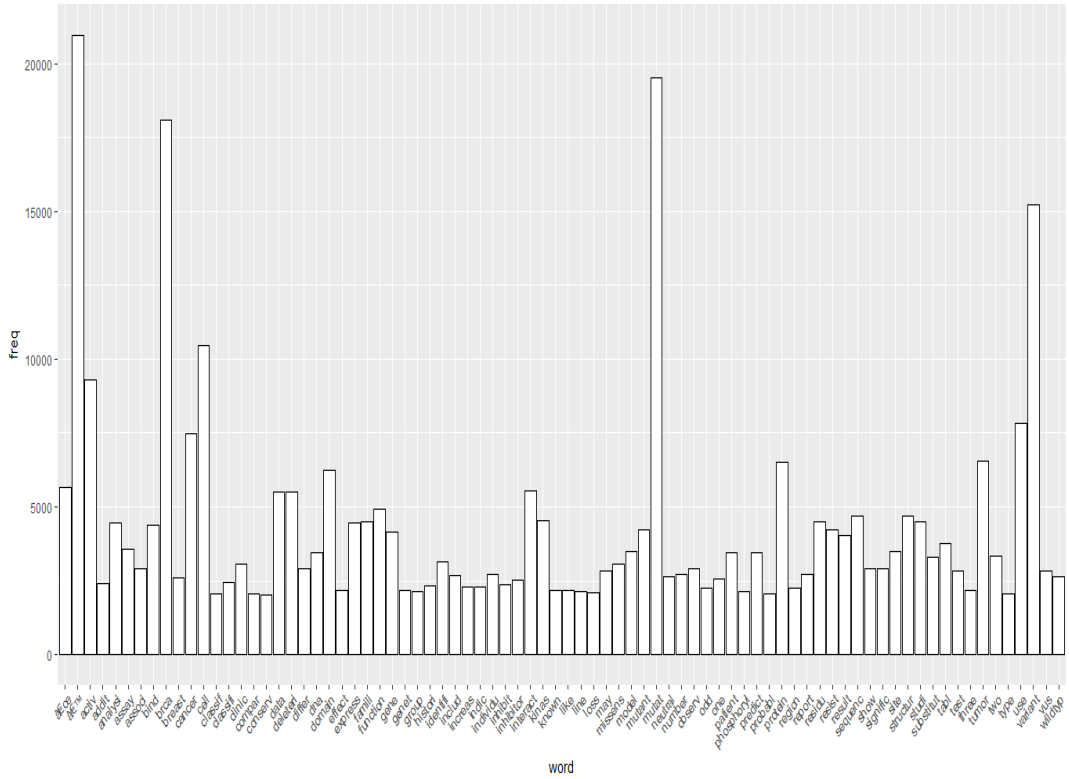
Graph (d) : Plot for word and frequency for Class 3



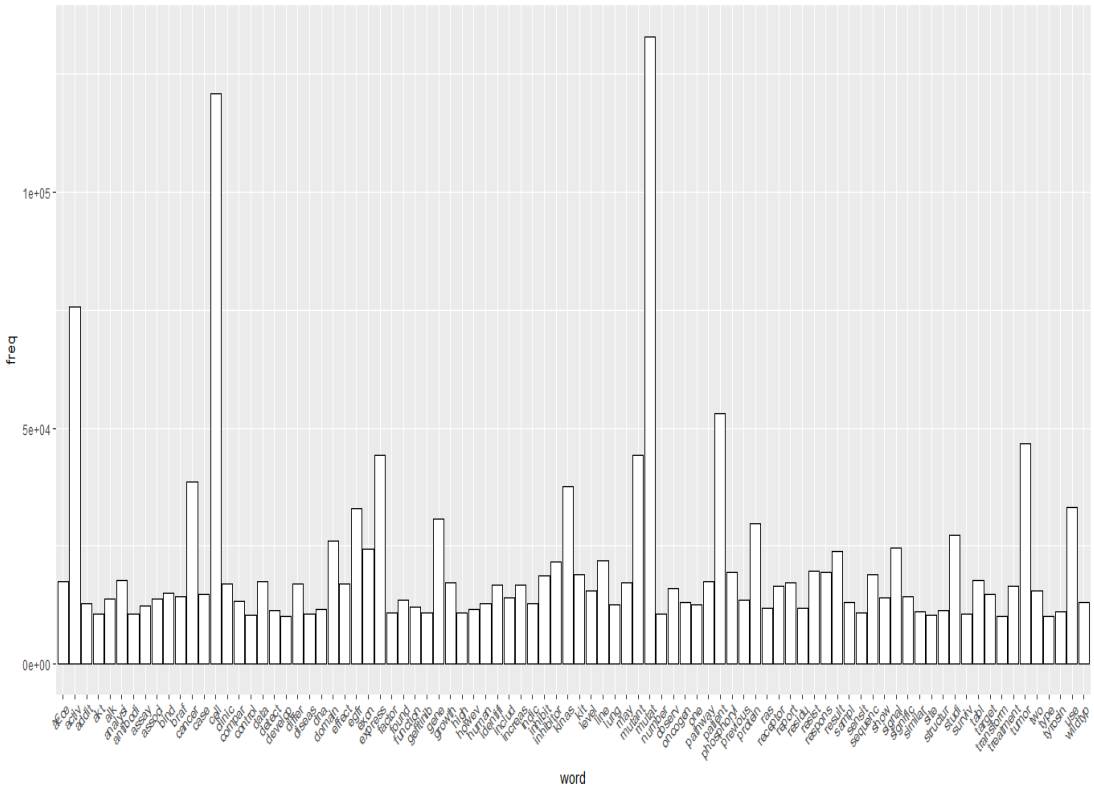
Graph (e) : Plot for word and frequency for Class 4



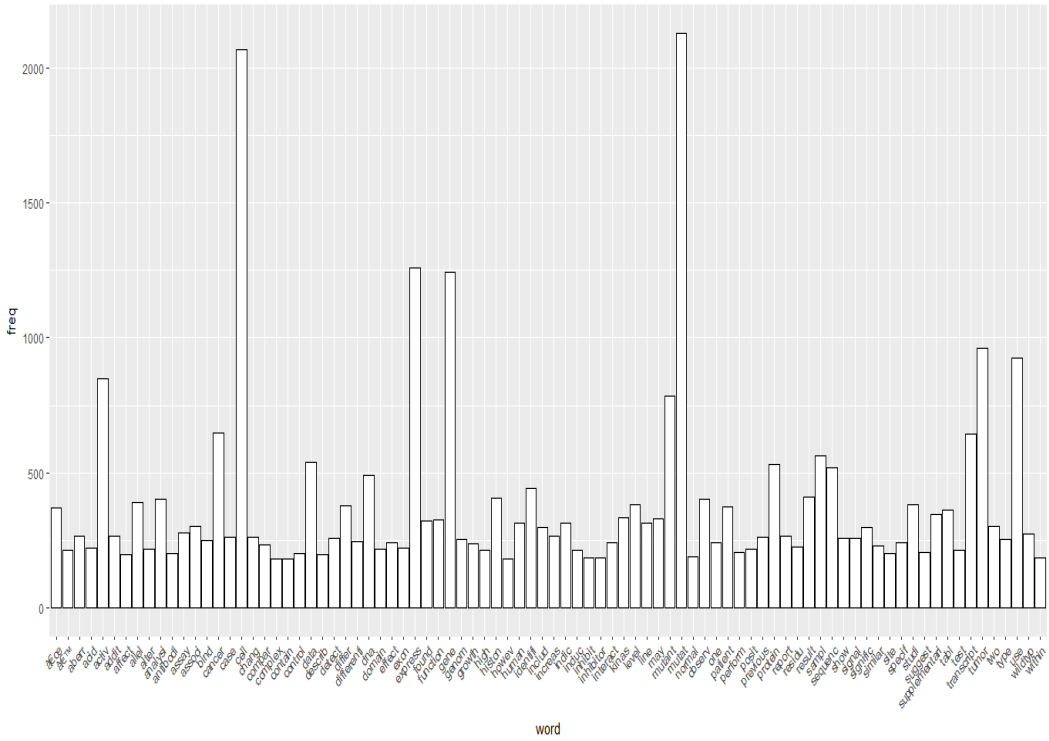
Graph (f) : Plot for word and frequency for Class 5



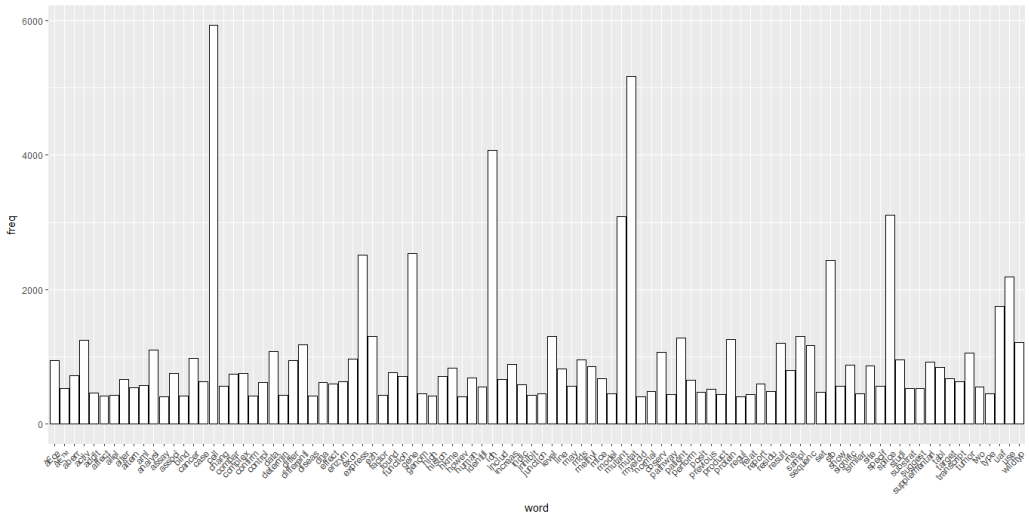
Graph (g) : Plot for word and frequency for Class 6



Graph (h) : Plot for word and frequency for Class 7



Graph (i) : Plot for word and frequency for Class 8



Graph (j) : Plot for word and frequency for Class 9

From the graphs above, one may see the following frequent distinct words in each class:
Class 1(Graph(b)) : brca
Class 2(Graph(c)) : egfr, kinas
Class 3(Graph(d)) : brca
Class 4(Graph(e)) : missens, pten, brca, kinas
Class 5(Graph(f)) : Breast Cancer, brca1
Class 6(Graph(g)) : Breast Cancer, brca, brct
Class 7(Graph(h)) : braf, egfr, tyrosin, lung cancer
Class 8(Graph(i)) : pten
Class 9(Graph(j)) : sfb

6. Model Building

To test the data we divided the training set of 3321 observations into train(3000 observations) and test(321 observations) data set. We trained the model using the train dataset and predicted the 'Class' of test set by applying XGBoost Model. We ran the model thrice to predict the value of "Class" variable and found the mode of all the three results obtained and this decided the final values of the Class.

7. Results

By applying the trained model to our test data, we attained an accuracy of 47.66%.
7.1 Overall Statistics

```
Accuracy : 0.4766
                 95% CI : (0.4209, 0.5328)
    No Information Rate : 0.4611
    P-Value [Acc > NIR] : 0.3068

                  Kappa : 0.2158
 Mcnemar's Test P-Value : NA
```

7.2 Statistics by Class

	Class:7	Class:2	Class:1	Class:4	Class:6	Class:9	Class:5	Class:8	Class:3
Sensitivity	0.6284	0.5375	0.37143	0.14815	0.000000	0.000000	0.00000	0.000000	0.00000
Specificity	0.6243	0.6680	0.96154	0.98980	0.996743	0.975000	1.00000	1.000000	1.00000
Pos Pred Value	0.5886	0.3496	0.54167	0.57143	0.000000	0.000000	NaN	NaN	NaN
Neg Pred Value	0.6626	0.8131	0.92593	0.92675	0.956250	0.996805	0.96885	0.993769	0.98754
Prevalence	0.4611	0.2492	0.10903	0.08411	0.043614	0.003115	0.03115	0.006231	0.01246
Detection Rate	0.2897	0.1340	0.04050	0.01246	0.000000	0.000000	0.00000	0.000000	0.00000
Detection Prevalence	0.4922	0.3832	0.07477	0.02181	0.003115	0.024922	0.00000	0.000000	0.00000
Balanced Accuracy	0.6263	0.6028	0.66648	0.56897	0.498371	0.487500	0.50000	0.500000	0.50000

Predicted/ Observed	**1**	**2**	**4**	**6**	**7**	**8**	**9**
1	17	0	1	1	4	8	4
2	5	7	0	5	39	22	2
3	1	0	0	0	1	2	0
4	2	0	1	2	12	4	6
5	0	0	0	0	4	6	0
6	2	1	0	1	8	1	1
7	4	8	1	1	100	34	0
8	0	0	1	0	0	1	0
9	0	0	0	0	0	1	0

Table (a) : Confusion Matrix for XGBoost Model