Q1. a)

Binary Independence:

There are two sets of documents, in any retrieval model that assumes that relevance is binary. These sets are the relevant and non-relevant documents. The search engine's task is to decide whether the document belongs to the relevant or non-relevant set. That is,

if P(R I D) > P(NR I D) document is relevant and vice versa. Here P(R I D) = conditional probability representing the probability of relevance given the representation of that document.

To find, P(R I D), we first find, P(D I R) and then use Bayes' rule:

P(R I D) = P(D I R) P(R) / P(D)

Now we can classify a document relevant if,

P(D I R) / P(D I NR) > P(NR) / P(R)

Here,
P(D I R) / P (D I NR) is known as likelihood ratio and if this is used as a score, then higher rank documents will be those that have higher likelihood of being relevant.

In this model, documents are represented as a vector of binary features, D = (d1, d2, d3, …, dt) where di = 1, if the term i is present in the document, 0 otherwise.
We also assume, that the terms are independent. That is,

π(i = 1 to t) P(di I R)

Due to these two points, of binary features and term independence, this model is known as binary independence model.

$$\sum_{i:di = qi = 1} \log \frac{(ri + 0.5) / (R - ri + 0.5)}{(ni - ri + 0.5) / (N - ni - R + ri + 0.5)}$$

Here, the numerator is basically the number of relevant documents that contain a term upon the total number of relevant documents

The denominator is the number of non-relevant documents that contain a term divided by the total number of non-relevant documents.

We add 0.5 to avoid log (0) and we take log since multiplying lot of small values can result in inaccurate data.

Q1. b)

Significance of k1

The constant k1 determines how the tf component of the term weight changes as fi changes. If k1 = 0, the term frequency component would be ignored and only term presence or absence would matter. If k1 is large, the term weight component would increase nearly linearly with fi.

Significance of K
K is a parameter that is used to normalize the tf component by document length.

$K = k1 ((1 - b) + b * dl / avdl)$
where b is a parameter, that regulates the impact of the length normalization. b = 0 corresponds to no length normalization and b =1 is full normalization.

Plotting graph:
Since k1 and K = 1, the term-weighting component of a BM25 will be

$2fi / 1 + fi$. The graph for the same can be found in freqvsweight.xlsx

The x-axis is the frequency of the terms and the y-axis is the term weight associated with that frequency