I have split the practical into three questions to ensure each part of the question is answered in depth.

Q1. Take the simple sentences we used in our word example from the lecture. Put these into the program and compute the K-L divergence scores for them, in both directions.

We will firstly start this question by explaining what KL divergence is. KL or Kullback–Leibler divergence can also be called relative entropy. It quantifies how much one probability distribution differs from another probability distribution. KL is named after Solomon Kullback and Richard A.Leibler The KL divergence between two distributions Q and P is often stated using the following notation:

- KL(P || Q)

In this case, the "||" operator indicates "divergence of Ps divergence from Q. It is not a metric; it can be asymmetric, (also it does not satisfy the triangle inequality) and the divergence of P from Q may not be the same as Q from P. KL divergence can be computed as follows: The negative sum of probability of each event in P multiplied by the log of the probability of the event in Q over the probability of the event in P. KL Divergence aids us in measuring how much information is lost when an approximation is chosen. The origins of KL Divergence are in information theory. We can say the main goal of information theory is to compute how much information is in data. Entropy is the most important metric in information theory. Entropy is typically denoted as H. The formula for Entropy for a probability distribution can be seen below:

$$H = -\sum_{i=1}^{N} p(x_i) \cdot \log p(x_i)$$

Entropy can be interpreted as the minimum number of bits it would take us to encode our information. Kullback-Leibler Divergence is just a slight modification to our formula for entropy. Then we look at the difference of the log values for each rather than just having our probability distribution (p), we add in our approximating distribution (q) :

$$D_{KL}(p||q) = \sum_{i=1}^{N} p(x_i) \cdot log\frac{p(x_i)}{q(x_i)}$$

If the two distributions are shown to match perfectly, then DKL(p || q) = 0. If not, the values can range anywhere from 0 to an infinite value. The lower the value we get from our KL divergence score, the better we can say we have matched the true distribution with our approximation.

Now that we have explained what KL divergence is in depth, we will use the word example from our lecture to compute the KL divergence scores in both directions.

The word example from the lecture can be seen below:

```
d1 = """john fell down harry fell as-well down by
the stream the sun shone before it went down"""

d2 = """bill fell down jeff fell too down by the
river the sun shone until it sunk down belinda was ill"""
```

We have put these words into the program and the following is the K-L divergence score obtained in both directions:

```
KL-divergence between d1 and d2: 3.0246436601452777
KL-divergence between d2 and d1: 3.459887986316757
```

The scores obtained here for DKL(d1 || d2) and DKL(d2 || d1) are quite **low** which indicates that each of these word examples have less divergence and also their probabilities match to a great extent. Due to the fact that DKL(d1 || d2) does not equal DKL(d2 || d1), it proves that the KL divergence score is **asymmetric**.

Q2. Now create a third story that is very different to the other two, add it to the program and report how its score changes relative to the first two. Comment on whether the scores make sense.

The third story I have created can be seen below:

```
d3 = """The intelligence of a female polar bear can be credited
to their indulgence of fish"""
```

The K-L divergence scores obtained from passing our third story through the program relative to the first two stories can be seen below:

```
KL-divergence between d1 and d3: 7.236637718952577
KL-divergence between d2 and d3: 7.194663944546398
```

We will now comment on whether we believe the scores makes sense. I believe the K-L divergence scores do in fact make sense for this text. Based on the results, we can see the divergence scores are **high** when compared to the results for DKL(d1 || d2) and DKL(d2 || d1) which were considered **low** when compared together. This is due to the fact that our third story (d3) is dissimilar and unlike the previous two stories (d1 & d2). We can say that we have not matched the true distribution with our approximation due to these high KL divergence values. The terms contained in d1 and d2 do not resemble the terms used in d3 and as a result we get our logical high divergence scores.

Q3. Explain what role epsilon and gamma play in the computation of K-L.

In terms of the computation of K-L, **epsilon** is the probability of a term which is not in a document. The value is set to a small number instead of 0 to avoid the distance being an infinite value. Whereas **gamma** is a normalization coefficient to account of **epsilon**, so a probability of a term in a category satisfies the properties of a probability (sum to 1). As well as this, the article "Computing the Kullback-Leibler Divergence between two Generalized Gamma Distributions ", Christian Bauckhage describes how a closed form solution can be derived for the Kullback-Leibler by using two generalized **gamma** distributions and finding the divergence between them.

In the article "Using Kullback-Leibler Distance for Text Categorization", Brigitte Bigi proposes a back-off smoothing model where term frequencies that appear in the text are weakened and all the expressions which do not appear in the text are given an epsilon probability equal to the probability of the unknown words. The formula for the back-off method can be seen below:

$$P(t_k, d_j) = \begin{cases} \beta P(t_k \mid d_j) & if\ t_k\ occurs\ in\ the\ document\ d_j \\ \varepsilon & else \end{cases}$$

Where:

$$P(t_k \mid d_j) = \frac{tf(t_k, d_j)}{\sum_{x \in d_j} tf(t_x, d_j)}$$

The process of assigning weights is called the **epsilon probability** (ϵ) The value of ϵ (epsilon) is defined such that it is smaller than the smallest distance between the samples.

This explains the roles of both epsilon and gamma for the computation of K-L.

References:

Bauckhage, Christian. (2014). Computing the Kullback-Leibler Divergence between two Generalized Gamma Distributions. arXiv. 1401.6853.

Brigitte Bigi. Using Kullback-Leibler Distance for Text Categorization. Advances in Information Retrieval, 2633, Springer Berlin Heidelberg, pp.305-319, 2003, ⟨10.1007/3-540-36618-0_22⟩. ⟨hal-01392500⟩

Eguchi, S., & Copas, J. (2006). Interpreting Kullback–Leibler divergence with the Neyman–Pearson lemma. Journal Of Multivariate Analysis, 97(9), 2034-2040. doi: 10.1016/j.jmva.2006.03.007

Thirumagal, E., & Saruladha, K. (2021). GAN models in natural language processing and image translation. Retrieved from https://www.sciencedirect.com/topics/engineering/kullback-leibler-divergence