I'm not a robot

In this post I would like to write and give some intuition about the Kullback-Leibler Divergence, which is a measure of how different two probability distributions over the same random variable are. I'll start by giving an intuitive explanation of the entropy and then derive the Kullback-Leibler Divergence from it. Lastly, I will calculate the KL Divergence for a small toy example. One of the main goals of information theory is to quantify the information content of data, or in other words, how surprising an event is. For example, consider an unfair coin that lands heads up in 95% of the time. You wouldn't be surprised if heads are up on a throw. In contrast, you would be very surprised if tails is up. This intuition, where an unlikely event is more informative than a likely event, is formalized in the Self-Information or Shannon-Information which can be rewritten to So for our coin example, the probability of heads coming up is , whereas the probability of tails coming up is . When calculating the Self-Information for both, we can clearly see that the outcome tails is way more surprising than the outcome heads However, usually we are not interested in the surprise of a particular event, but on the surprise we can expect on average for a whole probability distribution. This is where the entropy comes into place. Entropy is the expected value of the Self-Information and is defined for a discrete random variable with possible values as One important aspect here is the choice of the logarithm. If it is base 2, the unit of the entropy is in bits. With this in mind, entropy can also be explained as the minimum number of bits (or yes-no-questions) needed, on average, to encode our data. Consider a fair coin, where each value is equally likely (). The entropy of one toss would be 1. This indicates, that we need one bit, and thus, one yes-no-question, to encode the data (the bit is either turned on or off). This example can of course be extended to a random variable with multiple possible outcome values. Let's take a random variable that can take a value between 1 and 8 () each with equal probability . How many yes-no-questions do we need? Calculating the entropy gives us Hence, we need 3 yes-no-questions to find our value, but of course not simply asking "Is the value a 1?", "Is the value a 2?", etc. The questions asked are: "Is the value in the first or in the second half?" and that 3 times. Besides giving a lower bound of the number of bits needed to encode information, entropy is also often given as a measure of the amount of disorder in a system. A high entropy indicates a big amount of disorder, whereas a low entropy indicates a low amount of disorder. For a better intuition of this notion of entropy, let's assume a dataset which contains several objects, each belonging to one of two classes. The frequency of class 1 objects can be denoted with the parameter , so that and . In simple terms, if the dataset contains 100 objects and , then 60 objects belong to class 1 and the rest to class 2. The entropy (or in a loose notion the disorder of the dataset) can then be written as a function of the parameter : Note: For those of you who are familiar with machine learning, this equation should already be well-known, as it is nothing more than the binary cross-entropy loss. Plotting this equation results in the following function: This plot further visualizes what we've discussed so far. If all objects of the dataset belong to only one class (or) the entropy or disorder of the dataset equals zero. This means loosely spoken, there is "no disorder" in the dataset. However, if we are dealing with a balanced dataset (), the "disorder" is maximal with . Note: I have taken a dataset for the example here to illustrate the concept of "disorder". However, this example can also be rewritten so that it represents the outcome of a Bernoulli random variable with binary outcome: success or failure . This would be more appropriate in terms of entropy being a measure of uncertainty of an outcome, with higher entropy indicating higher uncertainty. Up to this point we have seen that self-information is a measure to quantify the information of a specific event, and that the entropy is the expected information for a whole probability distribution. With this we have finally worked out the prerequisites to understand the Kullback-Leibler divergence, which is a measure to quantify how different two distributions are, or in other words, how much information is lost when we approximate one distribution with another. For two probability distributions and over the same random variable , the Kullback-Leibler Divergence is defined as which can be rewritten to For a discrete random variable, the KL divergence is the extra amount bits needed to encode samples from using a code that was designed for samples drawn from . If both probability distributions are the same, then we would not need any extra bit and thus, the KL-divergence equals zero (the probability distributions do not differ). For a more intuitive understanding, let's take the coin example again. Consider a fair coin with equal probability for heads and tails coming up and a biased coin where heads is coming up 95% of the time One way of comparing these two distributions is to sample some sequences and compare what probability each distribution would assign to each sequence. If both distributions assign similar probabilities to the same sample sequence, it implies that the two distributions might be similar. Let's clarify it further with an example, where we draw a sample sequence from the fair coin distribution by tossing the fair coin 10 times. This could generate a sequence like with and . The probability of getting this sequence given the fair coin is whereas the probability of generating this sequence when sampling from the biased coin distribution is When comparing these two probabilities one can clearly see that the likelihood of generating the sequence given the biased coin is way smaller than generating the sequence with the fair coin. This suggests that the two distributions might not be similar, as they assign different probabilities to the same sequence. For a direct comparison we can take the ratio of both probabilities as a measure for similarity. Expressing this in a more formal way with , for the fair coin and , for the biased coin gives us Taking the log of this equation and normalizing it by the length of the sequence (sample size) gives us By applying logarithm rules we can refine this equation If the generated sequence from the fair coin is infinitely long the proportion of heads coming up and the proportion of tails coming up tends towards and , respectively. Hence, Rearranging this equation gives us which is equivalent to the discrete Kullback-Leibler Divergence Finally I would like to calculate the Kullback-Leibler Divergence for a small toy example. Assume we conducted a survey where we asked 100 persons how many plants they have in their living room. The empirical probability distribution is Observed probability. How much information would we lose if we approximate this empirical distribution with a uniform distribution where each outcome is equally likely? Approximate with uniform probability distribution. We can calculate that by using the KL divergence with representing the uniform distribution: How much information would we lose if we approximate this empirical distribution with a Binomial distribution with parameter equal to the normalized expectation of our empirical distribution? Approximate with binomial distribution. Again, calculating the Kullback-Leibler divergence with being the binomial distribution gives us: By comparing both values we can conclude, that the Binomial distribution approximates our empirical distribution better than the uniform distribution, as the KL divergence is smaller. Summing up, we have seen that self-information is used to quantify the information of a specific outcome/event and the expected information for a probability distribution is determined by the entropy. The Kullback-Leibler divergence is based on the entropy and a measure to quantify how different two probability distributions are, or in other words, how much information is lost if we approximate one distribution with another distribution. However, one drawback of the Kullback-Leibler divergence is that it is not a metric, since (not symmetric). If a symmetric measurement is preferred, I would recommend to read more about the Jensen-Shannon divergence which is bound to values between 0 and 1. If you have any comments, feel free to leave a comment below or write a short message. Ian J. Goodfellow, Ian J.; Bengio, Yoshua; and Courville, Aaron (2016). Deep Learning. MIT Press.Mezard, M., & Montanari, A. (2009). Information, physics, and computation. Oxford University Press.Stone, J. V. (2015). Information theory: a tutorial introduction. Kullback–Leibler divergence (KL divergence), also known as relative entropy, is a fundamental concept in statistics and information theory. It measures how one probability distribution diverges from a second, reference probability distribution. This article delves into the mathematical foundations of KL divergence, its interpretation, properties, applications across various fields, and practical considerations for its implementation.1. IntroductionIntroduced by Solomon Kullback and Richard Leibler in 1951, KL divergence quantifies the information lost when one distribution is used to approximate another. It is widely utilized in various fields such as machine learning, statistics, fluid mechanics, neuroscience, and bioinformatics. Understanding KL divergence is essential for model comparison and inference in these domains.2. Mathematical Foundations2.1 Discrete VariablesFor discrete random variables PP and QQ with supports XX and probability mass functions p(x)p(x) and q(x)q(x), the KL divergence from QQ to PP is defined as:This formula captures the expected number of extra bits required to encode samples from PP using a code… I'm starting a new series of blog articles following a beginner friendly approach to understanding some of the challenging concepts in machine learning. To start with, we will start with KL divergence. Code:Here The other articles of this series can be found below. *A B C D E F G H I J K L M N O P Q R S T U V W X Y Z** denotes articles behind the Medium paywall First of all let us build some ground rules. We will define few things we need to know like the back of our hands to understand KL divergence. By distribution we refer to different things such as data distributions or probability distributions. Here we are interested in probability distributions. Imagine you draw two axis (that is, X and Y) on a paper, I like to imagine a distribution as a thread dropped between the two axis; X and Y. X represents different values you are interested in obtaining probabilities for. Y represents the probability of observing some value on the X axis (that is, y=p(x)). I visualize this below. This is a continuous probability distribution. For example think of axis X as the height or a human and Y as the probability of finding a person with that height. If you want to make this probability distribution discrete, you cut this thread to fixed length pieces and turn the pieces in such a way that they are horizontal. And then create rectangles connecting the edges of each piece of thread and the x-axis. That is a discrete probability distribution. For a discrete probability distribution, an event is you observing X taking some value (e.g. X=1). Let us call P(X=1) probability of the event X=1. In continuous space you can think of this as a range of values (e.g. 0.95< X