"You may see a cup of tea fall off a table and break into pieces on the floor... But you will never see the cup gather itself back together and jump back on the table. The increase of disorder, or entropy, is what distinguishes the past from the future, giving a direction to time."
- Stephen Hawking

Entropy, Cross-Entropy and KL-Divergence

Entropy is a measure in Machine Learning that is used as an optimization function that describes just how certain the events in question are. If the events have equal probabilities, the entropy would be high; If the probabilities of the said events form a skewed distribution[1], for example, the entropy would be low.

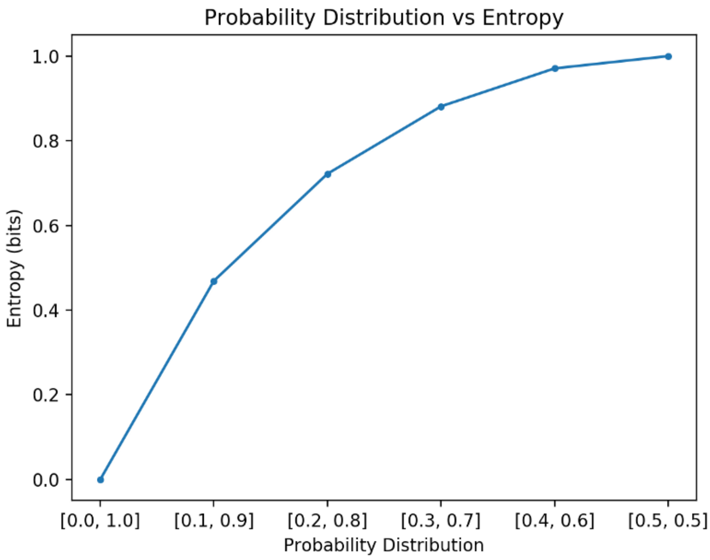

Fig 1: Probability Distribution vs Entropy Plot [2]

When we transmit bits to communicate, not every bit will be useful to the recipient. In order to optimize our system while communicating a message, we need more of the useful bits and less of the redundant bits or bits with error. According to Claude Shannon, who introduced the Information Theory in the late 1940s, transmitting one bit of data reduces the receipent's uncertainity by a factor of 2. In order to find the number of bits communicated in an equally likely possibility scenario, all we need to do is taking the log of our reduction factor.

However, if our scenario involves different possibilities, we will need to calculate the bits communicated on average – which we achieve by calculating the Entropy:

Entropy:

$$H(\mathbf{p}) = -\Sigma_i \, p_i \log_2(p_i)$$

Cross-Entropy, on the other hand, is used as a cost function and it indicates the average message length. It is used to adjust the model weights during the training process to minimize the loss. To calculate Cross-Entropy, we take the log of the predicted probability instead of the true probability:

Cross-Entropy:

$$H(\mathbf{p}, \mathbf{q}) = -\Sigma_i \, p_i \log_2(q_i)$$

If we get the perfect predictions for our model, Cross-Entropy will be equal to Entropy since p will be equal to q. But if our predictions are not exactly perfect, Cross-Entropy will be greater than Entropy. In the latter case, the difference between the Cross-Entropy and the Entropy gives us the KL-Divergence, which is also called the Relative Entropy:

KL Divergence:

$$D_{KL}(\mathbf{p} \,||\, \mathbf{q}) = H(\mathbf{p}, \mathbf{q}) - H(\mathbf{p})$$

We touched upon the basics of the Entropy, Cross-Entropy and KL-Divergence, learned about their practical usage and how they are calculated. To conclude, let's apply Cross-Entropy to an example case in order to visualize and check out how it changes based on true probability.

In this section, true and predicted probability values we will be using are directly taken from the weather forecast case from Aurélien Géron's video[3].

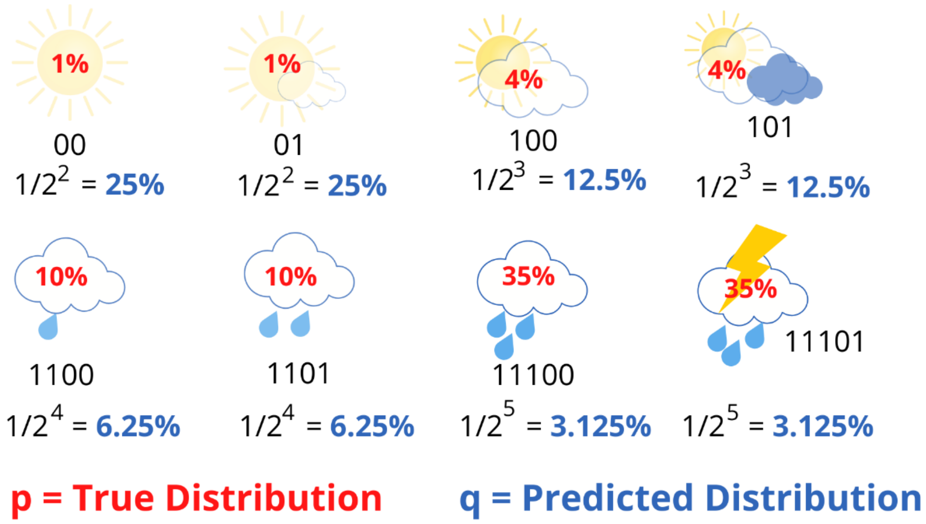

Fig 2: A Weather Forecast Case

Code:

```python
import seaborn as sns
import matplotlib.pyplot as plt
import math

def CrossEntropy(p, q):
    CE = []
    for i in range(len(p_list)):
        CE.append(-1*p_list[i]*math.log2(q_list[i]))
    return CE

# true probability distribution list for the weather forecast
p_list = [0.01, 0.01, 0.04, 0.04, 0.1, 0.1, 0.35, 0.35]
# predicted probability distribution for the weather forecast
q_list = [0.25, 0.25, 0.125, 0.125, 0.0625, 0.0625, 0.03125, 0.03125]

# calculating the cross entropy
CE = CrossEntropy(p_list, q_list)

# visualizing cross entropy vs true probability distribution
sns.set_style("darkgrid")
sns.lineplot(x=p_list, y=CE)
plt.rcParams['figure.dpi'] = 500
plt.xlabel("True Probability Distribution")
plt.ylabel("Cross Entropy")
plt.title("Weather Forecast Entropy Graph")
plt.show()
```

`Output:`

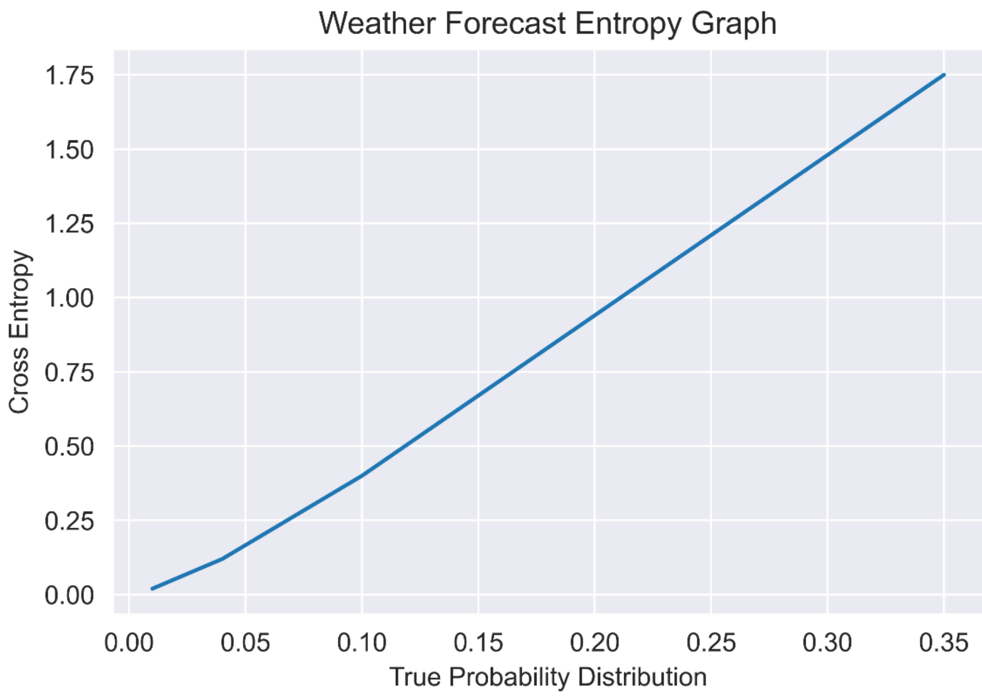

Fig 3: Output Plot of Our Code: Weather Forecast Entropy Graph

It is observed from the plot (Fig 3) that the closer the true probability of our weather type gets to the total probability of all the other types, the Cross-Entropy increases. This proves that the closer the events are to being equally likely, the harder time we will have making predictions about them.

References

[1] Koppert-Anisimova, I., 2021. *Cross-Entropy Loss in ML*. [online] Medium. Available at: <https://medium.com/unpackai/cross-entropy-loss-in-ml-d9f22fc11fe0> [Accessed 12 August 2021].

[2] Pandey, S., 2021. *A Friendly Introduction to Cross-Entropy for Machine Learning*. [online] Medium. Available at: <https://medium.com/analytics-vidhya/a-friendly-introduction-to-cross-entropy-for-machine-learning-b4e9f2b1f6> [Accessed 12 August 2021].

[3] *https://youtu.be/ErfnhcEV1O8*. 2018. [video] Aurélien Géron.