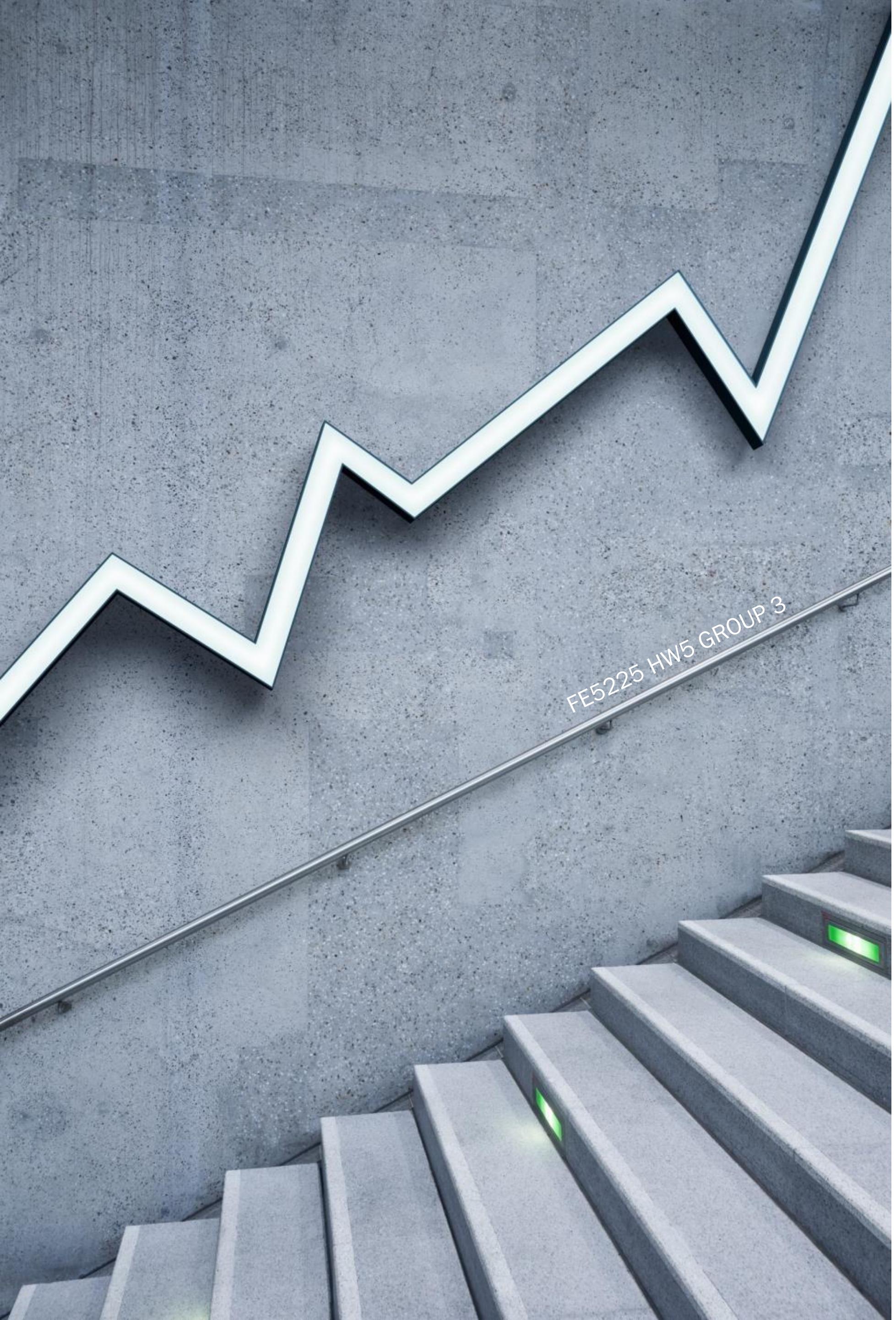


Entropy & Cross-Entropy

IN MATHEMATICS

FE5225 HW5 Group 3

Understanding Entropy

"The increase of disorder or entropy is what distinguishes the past from the future, giving a direction to time."

— Stephen Hawking, A Brief History of Time

Spontaneous change for an irreversible process in an isolated system always proceeds in the direction of increasing entropy.

— Rudolf Clausius, The Second Law of Thermodynamics

The Shannon entropy of a distribution is the expected amount of information in an event drawn from that distribution.

— Claude Shannon, father of information theory

Calculate Information for Events

- **Low Probability Event: High Information (surprising).**
- **High Probability Event: Low Information (unsurprising).**
- The amount of information of a discrete event is calculated using the probability of the event.

$$H(x) = -\log(\, p(x) \,)$$

- Example:
  - $p(x)=1$, information: $H(x) = 0$. Information will be zero when an event is certain, e.g. there is no surprise.
  - $p(x)=0.500$, information: $H(x) = 1.0$ bits
  - $p(x)=0.100$, information: $H(x) = 3.322$ bits
- If the base-e or natural logarithm is used instead, the result will have the units called nats.

Calculate Entropy for a Random Variable

- Skewed Probability Distribution (unsurprising): Low entropy.
- Balanced Probability Distribution (surprising): High entropy.
- Entropy can be calculated for a random variable X with k in K discrete states as follows:

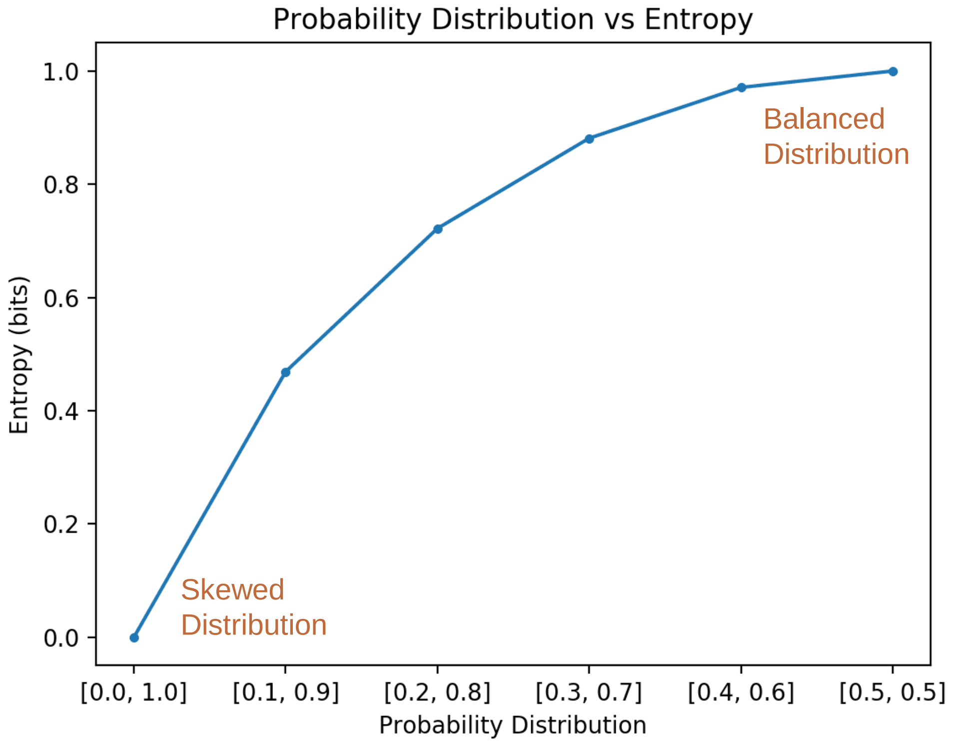
H(**X**) = -sum(p(k) * log(p(k)) : each k in K)

Probability Distribution vs Entropy

```python
# compare probability distributions vs entropy
from math import log2
from matplotlib import pyplot

# calculate entropy
def entropy(events, ets=1e-15):
    return -sum([p * log2(p + ets) for p in events])

# define probabilities
probs = [0.0, 0.1, 0.2, 0.3, 0.4, 0.5]
# create probability distribution
dists = [[p, 1.0 - p] for p in probs]
# calculate entropy for each distribution
ents = [entropy(d) for d in dists]
# plot probability distribution vs entropy
pyplot.plot(probs, ents, marker='.')
pyplot.title('Probability Distribution vs Entropy')
pyplot.xticks(probs, [str(d) for d in dists])
pyplot.xlabel('Probability Distribution')
pyplot.ylabel('Entropy (bits)')
pyplot.show()
```



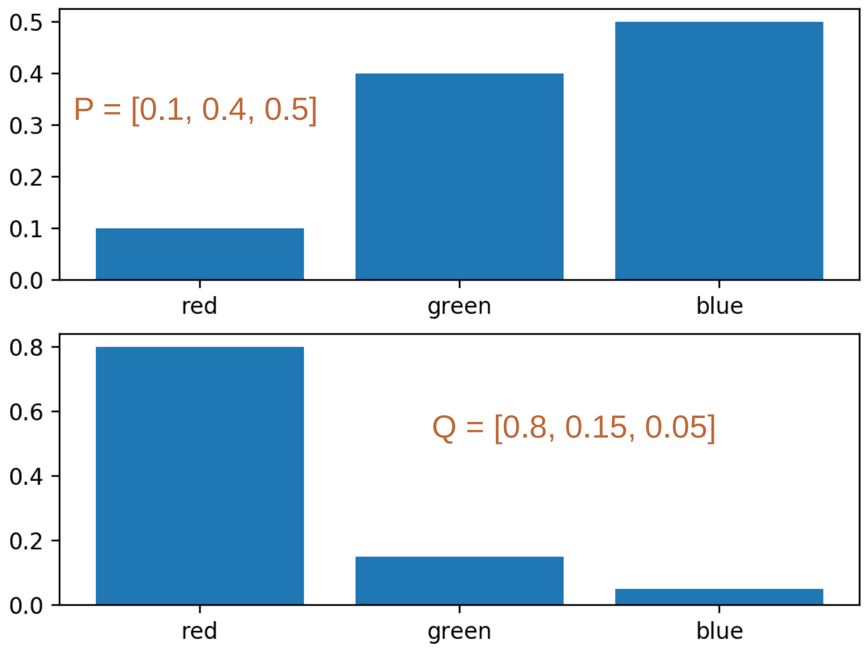
What is Cross-Entropy?

- Cross-entropy is a measure of the **difference between two probability distributions** for a given random variable or set of events.
- "The cross entropy is the average number of bits needed to encode data coming from a source with distribution P when we use model Q …"
- The cross-entropy between two probability distributions, such as Q from P, can be stated formally as: **H(P, Q).** Where H() is the cross-entropy function, P may be the target distribution and Q is the approximation of the target distribution.

$$H(P, Q) = \text{sum } \{-P(x) * \log(Q(x)) : x \text{ in } \mathbf{X} \}$$

Cross-Entropy Example

- Two Discrete Probability Distributions



- Calculate Cross-Entropy Between Distributions

H(P, Q) = sum {–P(x) * log(Q(x)): x in X }

H(P, P): 1.36 bits

H(Q, Q): 0.88 bits

H(P, Q): 3.29 bits

H(Q, P): 2.91 bits

e.g. 3.29 bits are needed to encode data coming from a source with distribution P when we use model Q.

Cross-Entropy as a Loss Function

- Cross-entropy is widely used as a loss function when **optimizing classification models (e.g. Logistic Regression, Artificial Neural Networks)**.
- "Using the cross-entropy error function instead of the sum-of-squares for a classification problem leads to **faster training** as well as improved generalization."
- E.g. Cross-entropy in Binary case:

$$\text{Loss} = -\frac{1}{\substack{\text{output} \\ \text{size}}} \sum_{i=1}^{\substack{\text{output} \\ \text{size}}} y_i \cdot \log \hat{y}_i + (1 - y_i) \cdot \log (1 - \hat{y}_i)$$

- Expected Probability (y): The known probability of each class label for an example in the dataset (P).
- Predicted Probability ($\hat{y}$): The probability of each class label an example predicted by the model (Q).

References

- https://machinelearningmastery.com/cross-entropy-for-machine-learning/
- https://machinelearningmastery.com/cross-entropy-for-machine-learning/
- https://peltarion.com/knowledge-center/documentation/modeling-view/build-an-ai-model/loss-functions/binary-crossentropy
- https://en.wikipedia.org/wiki/Claude_Shannon
- https://fs.blog/2018/11/entropy/