Homework

- Read chapter 1, section 2.1.
- Do problems 2.1, 2.3, 2.4, 2.5, 2.7 (pp. 59–60)
- Due Wednesday, February 9
- Move on to section 2.2
Probability so far

- Classical vs. frequentist vs. subjectivist probability

- Sum rule:
  \[ P(A \text{ or } B) = P(A) + P(B) - P(AB) \]
  or, for mutually exclusive events:
  \[ P(A \text{ or } B) = P(A) + P(B) \]

- Product rule:
  \[ P(A \text{ and } B) = P(A) \times P(B|A) \]
  or, for independent events:
  \[ P(A \text{ and } B) = P(A) \times P(B) \]

- Bayes Theorem:
  \[ P(A|B) = \frac{P(A) \times P(B|A)}{P(B)} \]
Expected value

- Expected value:
  \[ E[X] = \sum_i x_i P(x_i) \]

- Variance:
  \[ \text{Var}[X] = E[(X - E[X])^2] = E[X^2] - (E[X])^2 \]

- \[ E[X + Y] = E[X] + E[Y] \]

- \[ E[XY] = E[X]E[Y], \text{ when } X \text{ and } Y \text{ are independent} \]
Suppose you want to perform a sequence of Bernoulli trials: independent random trials with two possible outcomes

Call one outcome ‘success’

The probability of getting $r$ successes out of $n$ trials is given by the binomial distribution:

\[ p(r; n, p) = \binom{n}{r} p^r (1 - p)^{n-r} \]

This is a parametric distribution: $r$ is a random variable, $n$ and $p$ are parameters

$p(r; n, p)$ is sometimes written $p(r|n, p)$, $p_{n,p}(r)$, or $p(r)$
Expected value

• In European Roulette, the wheel is numbered 0–36, with outside bets paying even money. The catch is that 0 is neither odd nor even, neither high nor low, and neither black nor red.

• layout
Expected value

- Suppose you play six spins of the wheel for $1. The chance of at least breaking even is:

\[
P(\text{even}) = p(3; 6, \frac{18}{37}) + p(4; 6, \frac{18}{37}) + p(5; 6, \frac{18}{37}) + p(6; 6, \frac{18}{37})
\]
\[
= 0.311 + 0.222 + 0.084 + 0.013
\]
\[
= 0.631
\]

- But, the expected value of your profit on a $1 bet is:

\[
E[\text{profit}] = \frac{18}{37}(+1) + \frac{19}{37}(-1) = -0.027
\]
Multinomial Distribution

- When there are more than two possible outcomes, the binomial distribution doesn’t apply

- Instead, we can use the *multinomial* distribution:

\[
P(X_1 = x_1, \ldots, X_n = x_n; p_1, \ldots, p_n) = \frac{N!}{\prod_i x_i!} \prod_i p_i^{x_i}
\]

where

\[
\sum_i x_i = N
\]

with \( p_i > 0 \):

\[
\sum_i p_i = 1
\]
The binomial and multinomial distributions turn up frequently in NLP.

Parametric distributions allow us to generalize across different but related cases.

They also allow us to say something about cases where we don’t know or aren’t sure about the parameter values.

Probability distributions save us from having to work everything out from first principles every time.

Other distributions: geometric, hypergeometric, Poisson, normal.
A binary code for transmitting poker hands:

- straight flush: 0000
- four of a kind: 0001
- full house: 0010
- flush: 0100
- straight: 1000
- three of a kind: 0011
- two pair: 0101
- pair: 1001
- high card: 0111
Information theory

- This encoding will require $C = 4$ bits per hand.
- An encoding using $n$ bits can represent $2^n$ distinct possibilities.
- Recall that:
  \[ \log_2(2^n) = n \]
- In general, to encode $k$ possibilities we will need $C = \lceil \log_2(k) \rceil$ bits.
Information theory

• An improved code, taking advantage of uneven probabilities:

<table>
<thead>
<tr>
<th>Hand</th>
<th>Probability</th>
<th>Code</th>
</tr>
</thead>
<tbody>
<tr>
<td>straight flush</td>
<td>0.0000154</td>
<td>111111111</td>
</tr>
<tr>
<td>four of a kind</td>
<td>0.000240</td>
<td>111111110</td>
</tr>
<tr>
<td>full house</td>
<td>0.00144</td>
<td>11111110</td>
</tr>
<tr>
<td>flush</td>
<td>0.00196</td>
<td>111110</td>
</tr>
<tr>
<td>straight</td>
<td>0.00393</td>
<td>11110</td>
</tr>
<tr>
<td>three of a kind</td>
<td>0.0211</td>
<td>1110</td>
</tr>
<tr>
<td>two pair</td>
<td>0.0475</td>
<td>110</td>
</tr>
<tr>
<td>pair</td>
<td>0.422</td>
<td>10</td>
</tr>
<tr>
<td>high card</td>
<td>0.501</td>
<td>0</td>
</tr>
</tbody>
</table>

• Now the expected value of the message length $E[C]$ is 1.61 bits.
Entropy

- The *entropy* of a random variable $H(X)$ is a lower bound on the expected value of the message length $E_P[C(X)]$:

$$H(X) = - \sum_{x \in X} p(x) \log_2 p(x)$$

- Shannon’s (1948) *Source Coding Theorem* tells us that there exists an optimal encoding such that:

$$H(X) \leq C(X) < H(X) + 1$$
Information Theory

- Alan Turing, Claude Shannon

- A mathematical theory of communication and how messages convey information

- Originally developed for decoding messages during WW2, later applied to improving telegraph and telephone communication

- Based on probabilities: how would an optimal betting strategy change after you find out a roulette wheel is rigged?

- Fundamental concept = information entropy
Entropy

- Entropy is the average surprise on finding out the outcome of some random variable $X$

$$H(X) = - \sum_{x \in X} P(x) \log_2 P(x)$$

- If $P(x)$ is the probability that the outcome is $x$, then $-\log P(x)$ is how surprised we would be if the outcome were $x$

- Since $P(x)$ ranges from 0 (for impossible events) to 1 (for certain events), the surprise ranges from infinity (for impossible events) to 0 (for certain events)

- Entropy is the weighted average of the surprise for all possible outcomes
Entropy

- Entropy can be used to evaluate compression schemes, etc.

- It can also be used as a measure of information content

- ‘Surprise’ on receiving message $x$ is $-\log_2 p(x)$

- Since $p(x)$ ranges from 0 (for impossible events) to 1 (for certain events), the surprise ranges from infinity (for impossible events) to 0 (for certain events)

- Increasing the number of possible outcomes increases the entropy

- For a given number of outcomes, the uniform distribution has the greatest entropy
Entropy

- Entropy is never less than zero (no surprise, total information) but has no upper limit.
- More possible outcomes and more even probabilities raises surprise and entropy, fewer outcomes or skewed probabilities lowers surprise and entropy.
- Source Coding Theorem: an optimal binary encoding for S uses on average $H(S) \pm 1$ bits.
- One such optimal coding scheme is Huffman coding, created in 1954 as homework in an MIT information theory class.
**Entropy**

- **Shannon game:** guess the next letter in an English text

<table>
<thead>
<tr>
<th>Model</th>
<th>Entropy (bits)</th>
</tr>
</thead>
<tbody>
<tr>
<td>zeroth order</td>
<td>4.76</td>
</tr>
<tr>
<td>first order</td>
<td>4.03</td>
</tr>
<tr>
<td>second order</td>
<td>2.8</td>
</tr>
<tr>
<td>humans</td>
<td>1.3</td>
</tr>
</tbody>
</table>

- More information helps a lot – ‘optimal’ code depends on the amount of context

- Note that typical text encodings use 8 bits per character, so there’s lots of room for compression

- Also compare with the number of bits required for audio (speech) recordings