INFORMATION THEORY

Master of Logic, Master AI, Master CS, University of Amsterdam, 2018

TEACHER: Christian Schaffner, TAs: Yfke Dulek, Esteban Landerreche, Kyah Smaal

Team homework set 1

Problem 1: Two coins and a die (7pt)

You have two (fair) coins and a (fair) 4-sided die with outcomes $\{1, 2, 3, 4\}$. Let X be the number of heads after flipping the two coins and let Y be the result of rolling the die. Let Z be the average of X and Y.

(a) **(2pt)** What is the distribution P_Z of Z?

(b) **(3pt)** Compute the variances of X, Y and Z.

(c) **(2pt)** You play the following game. If $2X \geq Y$, you win X^2 euros and otherwise you lose 1 euro. What is your expected total gain or loss after playing this game 40 times?

Problem 2: Expectation and variance (7pt)

(a) **(2pt)** Show that expectation is linear: for arbitrary $a, b \in \mathbb{R}$ and arbitrary real random variables X, Y,

$$\mathbb{E}[aX + bY] = a\mathbb{E}[X] + b\mathbb{E}[Y].$$

(b) **(2pt)** Show that for arbitrary $a \in \mathbb{R}$ and an arbitrary real random variable X,

$$Var[aX] = a^2 Var[X].$$

(c) **(3pt)** Show that if two real random variables X and Y are independent, then

$$Var[X + Y] = Var[X] + Var[Y].$$

Hint: start by showing that if X and Y are independent, then $\mathbb{E}[X]\mathbb{E}[Y] = \mathbb{E}[XY]$.
Does the converse also hold, i.e., are X and Y independent whenever $Var[X + Y] = Var[X] + Var[Y]$? If so, give a proof. If not, give a counterexample.

Problem 3: Deriving the weak law of large numbers (9pt)

(a) **(2pt)** (Markov's inequality) For any real non-negative random variable X and any $t > 0$, show that

$$P[X \geq t] \leq \frac{\mathbb{E}[X]}{t} .$$

(b) **(1pt)** Exhibit a random variable (which can depend on t) that achieves this inequality with equality.

(c) **(3pt)** (Chebyshev's inequality.) Let Y be a random variable with mean μ and variance σ^2. Show that for any $\varepsilon > 0$,

$$P[|Y - \mu| \geq \varepsilon] \leq \frac{\sigma^2}{\varepsilon^2} .$$

Hint: Define a random variable $X := (Y - \mu)^2$.

(d) **(3pt)** (The weak law of large numbers.) Let $Z_1, Z_2, ..., Z_n$ real i.i.d. random variables with mean $\mu = \mathbb{E}[Z_i]$ and variance $\sigma^2 = \mathbb{E}[(Z_i - \mu)^2] < \infty$. Define the random variables $S_n = \frac{1}{n}\sum_{i=1}^{n} Z_i$. Show that

$$P[|S_n - \mu| \geq \varepsilon] \leq \frac{\sigma^2}{n\varepsilon^2} .$$

Thus, $P[|S_n - \mu| \geq \varepsilon] \to 0$ as $n \to \infty$. This is known as the weak law of large numbers (which we will use heavily in Week 03).

Problem 4: Multiple-choice test (3pt)

A multiple-choice exam has 4 choices for each question. A student has studied enough so that the probability she will know the answer to a question is 0.5, the probability that she will be able to eliminate one choice is 0.25, otherwise all 4 choices seem equally plausible. If she knows the answer she will get the question right. If not she has to guess from the 3 or 4 choices.

As the teacher you want the test to measure what the student knows. If the student answers a question correctly, what is the probability she knew the answer? Give your answer with three decimals of precision.

Problem 5: Fermi estimates (8pt)

Read up on what a Fermi estimate is. Give a Fermi estimate for the *daily number of git commits in the world*.

In this exercise, it is more important to be clear about your arguments and assumptions rather than getting the correct result.

(a) **(4pt)** Give your Fermi estimate without looking up any numbers on the internet.

(b) **(4pt)** Look up the number for your estimates from the previous subexercise. Possibly revise your strategy to do the estimate. Compare the new final outcome to what you got previously as endresult in (a).

One team member should post your answers to this discussion forum, don't forget to state your team name.

Problem 6: Programming: Computing Variational Distance (8pt)

The *total variation distance* between two probability distributions P and Q over the finite alphabet $\mathcal{X}$ is defined as

$$\|P - Q\| := \frac{1}{2}\sum_{x\in\mathcal{X}} |P(x) - Q(x)|$$

This distance measure is symmetric, fulfills the triangle inequality and is normalized, i.e. it is 0 iff $P = Q$ and 1 iff P and Q have disjoint support.

The *collision probability* of a distribution P over finite alphabet $\mathcal{X}$ is defined as

$$Coll(P) := \sum_{x\in\mathcal{X}} P(x)^2$$

In this exercise, we are going to analyze the letter frequencies of *Alice in Wonderland* in five different languages: English, German, Esperanto, Italian and Finnish. You can find all necessary files here: `https://github.com/cschaffner/InformationTheory/tree/master/Problems/HW1`. Hereby, we are going to consider only the 26 English letters (without space) and ignore that languages like German and Finnish have important other letters such as ä, ö, ü.

For $lang \in \{$eng, ger, esp, ita, fin$\}$, let P_{lang} be the frequency distribution of the 26 English letters (without space) of Alice in Wonderland.

(a) **(2pt)** Compute all pairwise variational distances $\|P_{lang} - P_{lang'}\|$ for $lang \neq lang' \in \{$eng, ger, esp, ita, fin$\}$. Which two languages are closest, which two are furthest apart in terms of variational distance?

Note: for this exercise and any future programming exercises, you do not have to submit your code. In the pdf that you hand in, describe in a few sentences which general strategy you used (e.g., what quantities did you compute and in what order?), any choices you made (e.g., how did you treat uppercase letters? How did you deal with edge cases?), and any 'sanity check' computations you may have performed (e.g., did you check what the variational distance between a text file and itself was?) By adding this information, you may still receive partial credit for your approach, even if your final numerical answer is incorrect.

(b) **(2pt)** Compute the five collision probabilities $Coll(P_{lang})$ for $lang \in \{$eng, ger, esp, ita, fin$\}$.

Note: You do not have to submit your code.

(c) **(1pt)** Why is it called collision probability?

(d) **(2pt)** You are given the file `permuted_cipher.txt` that has been encrypted by (first removing spaces and then) shuffling around the characters (i.e. by applying a permutation cipher). Note that this kind of encryption preserves the letter frequencies. Compute the frequency distribution P_{cipher} and figure out which language the original text was by picking the one that minimizes by the variational distance $\|P_{cipher} - P_{lang}\|$ with $lang \in \{$eng, ger, esp, ita, fin$\}$ as above.

Note: You do not have to submit your code.

(e) **(1pt)** Would you have picked the same language when comparing the collision probability $Coll(P_{cipher})$ to the ones above?