Please follow the homework policies on the course website.

---

1. **(10 pt.) Poissonization**

   For this problem, we will study a simplified variant of Wordle. We consider an n-letter alphabet, and a language containing all the possible k-letter words. One k-letter word is chosen to be the secret word of the day, and the goal is to identify all the letters that occur in this word. The player guessing guesses a single letter at a time, and it is revealed to them whether the letter appears in the secret word. (Note, in contrast to the real game Wordle, we only care about guessing the letters, not their order, and each guess consists of a single letter, not a full word).

   Throughout, you should assume that as n goes to infinity, $\omega(1) = k = o(n)$.

   Greg and Mary have been competing at Wordle for the past two weeks, and naturally, they both use a randomized algorithm to make their guesses.

   (a) **(2 pt.)** Greg is lazy (or lets his daughter do the typing) and makes his guesses completely at random. (His guesses might repeat themselves.) Give an upper bound on the number of guesses he needs to make until with probability 0.9, he learns all the letters in the secret word. Express your answer in terms of n and k with an exact leading term. You may use big-O notation for any lower order terms.

   Mary knows that the secret word comes from the following distribution: each letter $i \in 1, \ldots, n$ has a frequency p_i, where $\sum_{i=1}^{n} p_i = 1$, and the secret word is chosen by selecting each letter independently to be letter i with probability p_i. For simplicity, for the rest of the problem you may assume that the secret word has length $\text{Poi}(k)$.

   (b) **(2 pt.)** What is the distribution of number of occurrences of each letter in the secret word?

   Clarification: You do not need to justify your answer (if it is correct), but think about it enough to convince yourself of it! Feel free to include a justification for partial credit.

   (c) **(2 pt.)** Mary chooses her guesses independenty to come from exactly the same letter distribution as the secret word (given by the p_i). What is the probability that after she has guessed $\text{Poi}(cn)$ total guesses that she knows all the letters? Express your answer in terms of c and the p_i (and n and k).

   (d) **(3 pt.)** Suppose the following distribution p_i: A third of the letters are vowels, and each of their probabilities of appearing are $\frac{3}{2n}$. Two thirds of the letters are consonants, and their probability of appearing is $\frac{3}{4n}$. What is the number of guesses until Mary knows all the letters with probability more that 90%? Give the exact leading term. [**HINT:** *First find some c such that if Mary makes $\text{Poi}(cn)$ guesses, with probability 90% she guesses all the letters. Then argue that if she makes a deterministic and slightly higher number of guesses, she succeeds with probability 90%. Similarly, if she makes a deterministic and slightly lower guess, she fails with probability at least 10%.*]

(e) **(1 pt.)** Whose strategy is better for this letter distribution, Greg's or Mary's?

Clarification: It's okay that Greg and Mary are working in slightly different models (since Greg's word has exactly k letters and Mary's does not). We're just looking for a one-bit answer here.

(f) **(0 pt.)** Can you come up with a better strategy for the guesses, assuming you are still guessing letters independently at random? Find a distribution $\{p_i\}$ on which your strategy performs better than both Mary and Greg's strategies.

Formally, find some probabilities $\{p_i\}$ and $\{g_i\}$ and a value k such that if the letters of the secret word are chosen independently from the distribution $\{p_i\}$ and your guesses are chosen independently from the distribution $\{g_i\}$, then the number of guesses required to know the letters in the secret word with probability 90% is a constant factor smaller for your strategy than for either Greg or Mary's.

Can you find an optimal way to chose the g_i as a function of the p_i?

2. **(4 pt.)** Prove that $(\mathbb{R}^3, \ell_2)$ cannot be embedded into $(\mathbb{R}^2, \ell_2)$ with bounded distortion. In other words, there are no functions $f : \mathbb{R}^3 \to \mathbb{R}^2$ and constants $\alpha, \beta > 0$ such that the following inequality holds for all $x, y \in \mathbb{R}^3$:

$$\beta \|x - y\|_2 \le \|f(x) - f(y)\|_2 \le \alpha\beta \|x - y\|_2.$$

[**HINT:** *Try a proof by contradiction. How should the grid $G_n := \{(i, j, k) : i, j, k \in \{0, 1, \ldots, n\}\}$ be embedded?*]
[**HINT:** *A disc of radius r has area πr^2.*]

3. **(4 pt.)** We showed that Bourgain's embedding allows us to embed an arbitrary metric space (X, d) with $|X| = n$ into $(\mathbb{R}^k, \ell_1)$ with target dimension k being $O((\log n)^2)$ and distortion being $O(\log n)$. Moreover, the embedding can be computed efficiently using a randomized algorithm. Prove that the exact same embedding computed by the randomized algorithm also achieves $O(\log n)$ distortion with high probability when the target metric is ℓ_p for $p > 1$. We encourage you to emphasize only the differences from the proof in the lecture notes rather than copying the entire proof.
[**HINT:** *Let $f : X \to \mathbb{R}^k$ denote the relevant embedding. For any two points $x, y \in X$, we showed that $\|f(x) - f(y)\|_1 \le k \cdot d(x, y)$. Can we say something similar about $\|f(x) - f(y)\|_p$?*]
[**HINT:** *For any two points $a, b \in \mathbb{R}^k$ and $p > 1$, it holds that $\|a - b\|_p \ge k^{(1/p)-1} \|a - b\|_1$. This is a special case of Hölder's inequality.*]