

# Algorithms for Big Data

Fall Semester 2019

## Exercise Set 3

Sketching: we want to represent some large input (say  $M$ ) with a much shorter sketch  $s$ . Additionally we want to be able to produce some form of estimate based solely on sketch (sketches). For example: Alice and Bob both hold long input strings, respectively  $M_A$  and  $M_B$ . They want to compute (based only on their inputs) sketches  $s_A$  and  $s_B$  and send them to Charlie, who then computes e.g. a distance between  $M_A$  and  $M_B$  (approximately).

### Exercise 1:

Lets say  $M \in [0, 1]^n$  represents a vector of user preference. We can define a user cosine similarity score as

$$\text{sim}(A, B) = \cos(\angle(M_A, M_B)),$$

that is cosine of an angle the respective vectors make. Show an efficient way of sketching the vectors so that the cosine similarity can be computed with  $\pm \varepsilon$  precision.

The goal for next exercises is to derive a sketching scheme for estimating *Hamming distance*:  $\text{Ham}(x, y) = |\{i : x[i] \neq y[i]\}|$ .

### Exercise 2:

Consider binary alphabet  $\{0, 1\}$ . Use AMS sketches for to derive efficient sketching scheme binary inputs for estimating Hamming distance (up to  $1 \pm \varepsilon$  factor). What is the size of sketches?

### Exercise 3:

Consider random projection  $\varphi : \Sigma \rightarrow \{0, 1\}$ . Show that for some words  $x, y$ ,  $2 \cdot \text{Ham}(\varphi(x), \varphi(y))$  approximates  $\text{Ham}(x, y)$  *in expectation*. Improve the quality of estimation to multiplicative  $1 \pm \varepsilon$  by averaging over many independent choices of  $\varphi$ . (How many?)

### Exercise 4:

Show a sketching scheme for  $1 \pm \varepsilon$  approximating Hamming distance with sketches using  $\mathcal{O}(\frac{\log^2 \delta^{-1}}{\varepsilon^4})$  words and working with probability  $1 - \delta$ .

### Exercise 5:

Improve the sketches from previous exercise to  $\mathcal{O}(\frac{\log \delta^{-1}}{\varepsilon^2})$  words.

(2 pts)