CS 6966: High-Dimensional Data Analysis **Due: Thursday 11.21 at 11:50pm**

Homework 4: Optimization and Coresets

Instructions: Your answers are due at 11:50pm submitted on canvas. You must turn in a pdf through canvas. I recommend using latex (`http://www.cs.utah.edu/~jeffp/teaching/latex/`, see also `http://overleaf.com`) for producing the assignment answers. If the answers are too hard to read you will lose points, entire questions may be given a 0 (e.g. **sloppy pictures with your phone's camera are not ok, but very careful ones are**)

Please make sure your name appears at the top of the page.

You may discuss the concepts with your classmates, but write up the answers entirely on your own. **Be sure to show all the work involved in deriving your answers! If you just give a final answer without explanation, you may not receive credit for that question.**

---

1. [**50 points**] Gilbert's algorithm (covered in Lecture 19) iteratively converges to the polytope distance. Assume both polytopes as input are represented by point sets $A, B \subset \mathbb{R}^d$, and their convex hulls do not intersect. Let the polytope distance be ρ. That lecture discussed how after C'/ε steps the algorithm finds a distance $\hat{\rho}$ which satisfies $\hat{\rho} \geq \rho \geq (1-\varepsilon)\hat{\rho}$.

   I want you to **prove** a different property. That after $T = C/\rho^2$ steps that Gilbert's algorithm finds a pair $a_T \in CH(A)$ and $b_T \in CH(B)$ so that the bisector h of a_T and b_T linearly separates A and B.

   Assume there exists a unit ball that contains $A \cup B$. Here C will be an absolute constant that you do not need to (but could) specify. You should start by considering the similar statement for the Perceptron algorithm, discussed in Lecture 18. Gilbert's algorithm and the Perceptron algorithm are similar, but not the same.

2. [**30 points**] You are given a point set $X \subset \mathbb{R}^d$ and a positive definite matrix $M \subset \mathbb{R}^{d \times d}$. Let M define a Mahalanobis distance $\mathsf{d}_M(x,p) = \sqrt{(x-p)^T M (x-p)}$. Define a Mahalanobis ball as $B_{r,M}(c) = \{x \in \mathbb{R}^d \mid d_M(x,c) \leq r\}$.

   1. Based on the approximate MEB algorithm (from Lecture 19), design an algorithm that finds a Mahalanobis ball $B_{\hat{r},M}(\hat{c})$ which contains all of X, and so $\hat{r} \leq (1+\varepsilon)r^*$ where $B_{r^*,M}(c^*)$ is the Mahalanobis ball (with M fixed) that contains X and has the smallest radius r^*.

   2. Analyze the runtime of the algorithm.

3. [**20 points**] Consider a distribution $\mathcal{D}$ defined on $\mathbb{R}^d$ that has mean μ and all points x in the support of $\mathcal{D}$ are within a distance $\|x - \mu\| \leq \beta$. Now draw n iid samples $X \sim \mathcal{D}$, and take their mean $\hat{\mu} = \frac{1}{n}\sum_{x_i \in X} x_i$.

   1. How many samples n do I need so that with probability at least $1-\delta$ that $\|\mu - \hat{\mu}\| \leq \varepsilon$? Write the answer in big-Oh notation as a function of ε, δ, d, and β.

   2. Now assume $n = 3000$ and $\beta = 2$, and the big-Oh notation has a constant of 1 *(so if your answer above was $n = O(d^2\varepsilon\delta^2\beta^7)$ [this is not the right answer] then it is precisely $n = 1 \cdot d^2\varepsilon\delta^2\beta^7$).* If the probability of failure I am willing to tolerate is $\delta = 0.1$, what ε can I guarantee?