Introduction to Cryptography
Courant, Fall 2016

Homework 0

Instructor: Oded Regev
Student: YOUR NAME HERE

Homework is due by **7am of Sep 12**. Send by email to both "regev" (under the cs.nyu.edu domain) and "avt237@nyu.edu" with subject line "CSCI-GA 3210 Homework 0" and name the attachment "YOUR NAME HERE HW0.tex/pdf", and please also bring a printed copy to class. Start early!

Instructions. Solutions must be typeset in LaTeX (a template for this homework is available on the course web page). Your work will be graded on *correctness*, *clarity*, and *conciseness*. You should only submit work that you believe to be correct; if you cannot solve a problem completely, you will get significantly more partial credit if you clearly identify the gap(s) in your solution. It is good practice to start any long solution with an informal (but accurate) "proof summary" that describes the main idea.

You are expected to read all the hints either before or after submission, but before the next class.

You may collaborate with others on this problem set and consult external sources. However, you must ***write your own solutions***. You must also ***list your collaborators/sources*** for each problem.

1. Send email to Oded (`regev` at `cims`) with subject `CSCI-GA 3210 student` containing (1) a few sentences about yourself and your background (including your department, graduate program, how long in program), and (2) your comfort level with the following: mathematical proofs, elementary probability theory, big-O notation and analysis of algorithms, Turing machines, and P, BPP, NP, and NP-completeness. Please also mention any courses you've taken covering these topics.

2. *(Working with negligible functions.[1])* Recall that a non-negative function $\nu : \mathbb{N} \to \mathbb{R}$ is *negligible* if it decreases faster than the inverse of any polynomial (otherwise, we say that ν is *non-negligible*). More precisely, $\nu(n) = o(n^{-c})$ for every fixed constant $c > 0$, or equivalently, $\lim_{n\to\infty} \nu(n) \cdot n^c = 0$.

   State whether each of the following functions is negligible or non-negligible, and give a brief justification. In the following, $\mathrm{negl}(n)$ denotes some arbitrary negligible function, and $\mathrm{poly}(n)$ denotes some arbitrary polynomial in n. (If you are not comfortable with these notion, read Section 4.2 of Lecture 2 in Peikert's notes)

   (a) (1 point) $\nu(n) = 1/2^{100 \log n}$.

   > **Solution:** First we consider the limit $\lim_{n\to\infty} \frac{x^c}{e^{100log(n)}} = \lim_{n\to\infty} \frac{x^c}{n^{100}}$ for $c > 100$, this limit diverges to $+\infty$. Since $\lim_{n\to\infty} \frac{x^c}{e^{100log(n)}} < \lim_{n\to\infty} \frac{x^c}{2^{100log(n)}}$, we have that $\lim_{n\to\infty} \nu(n) \cdot n^c = +\infty$. Therefore the function is not negligible.

   (b) (1 point) $\nu(n) = n^{-\log\log\log n}$. (Compare with the previous item for "reasonable" values of n.)

   > **Solution:** for any c, we can pick $N_c = e^{e^{e^c}}$ so that $x^{-loglogloglogx} < x^{-c} \ \forall x > N_c$, satisfying the analytic definition of a negligible function. Therefore the function is negligible.

   (c) (1 point) $\nu(n) = \mathrm{poly}(n) \cdot \mathrm{negl}(n)$. (State whether ν is *always* negligible, or not necessarily.)

   > **Solution:** Since $\mathrm{negl}(n)$ is negligible, we have that $lim_{x\to\infty} \mathrm{negl}(n) \cdot x^c = 0 \ \forall c > 0$ The limit of the product $\mathrm{poly}(n) \cdot \mathrm{negl}(n)$ can be written as $\lim_{x\to\infty} \sum_{i=0}^{k} \alpha_i x^i * x^c * \mathrm{negl}(x) =$

[1] Based on a question from Peikert's class

$\sum_{i=0}^{k} \alpha_i \lim_{x\to\infty} x^{i+c} * \mathrm{negl}(x)$. Since each term in the summation is 0 due to the negligibility of $\mathrm{negl}(n)$, the whole limit is 0. Therefore, $\mathrm{poly}(n) \cdot \mathrm{negl}(n)$ is also negligible.

(d) (1 point) $\nu(n) = (\mathrm{negl}(n))^{1/\mathrm{poly}(n)}$. (Same instructions as previous item.)

Solution: Assume that $\mathrm{poly}(x)$ is a polynomial of finite degree but with degree at least 1, whose highest order coefficient is positive. Then we can write $lim_{x\to\infty} \mathrm{negl}(x)^{\frac{1}{\mathrm{poly}(x)}} x^c = (lim_{x\to\infty} \mathrm{negl}(x) x^{c\,\mathrm{poly}(x)})^{\frac{1}{\mathrm{poly}(x)}}$ Now, since $c\,\mathrm{poly}(x)$ is a positive number for large x $lim_{x\to\infty} \mathrm{negl}(x) x^{c\,\mathrm{poly}(x)}$ 0. Since the limit inside the exponent is 0 and the exponent is positive, the entire limit must be 0. Therefore, for polynomials with the above conditions, the function is negligible. If the polynomial diverges to $-\infty$ then the function is not negligible; if the polynomial is constant, it is negligible only when it is identically 0, and for the case of polynomials of infinite degree (which may be the series representation of any analytic function), the result may or may not be negligible.

(e) (1 point)

$$\nu(n) = \begin{cases} 2^{-n} & \text{if } n \text{ is composite} \\ 100^{-100} & \text{if } n \text{ is prime.} \end{cases}$$

Solution: Suppose that for any c, we could find a value of $x = N_c$ such that $\nu(x) < \frac{1}{x^c} \forall x > N_c$ We know that for any such N_c there is always a prime number $p > N_c$, and we can pick p such that $p > 100$. Suppose also that c=100. Then $\nu(p) = 100^{-100} > p^{-c}$, contradicting our assumption that no such $x > N_c$ exists. Thus $\nu(x)$ is not negligible.

3. *(Statistical distance.)* Recall that given two distributions over a (finite) set Ω, their statistical distance (also known as variational or L_1 distance) is defined as

$$\Delta(X, Y) := \frac{1}{2} \sum_{\omega\in\Omega} |X(\omega) - Y(\omega)| \ .$$

(a) (3 points) Show that Δ defines a metric (see here for the definition).

Solution: We can see that the statistical distance is never negative because the summand is non-negative due the presence of absolute value signs. To show that $\Delta(X, Y) = 0 -> X = Y$, suppose that $\exists a \in \Omega | X(a)! = Y(a)$ Then the summand is positive, and with no negative values to compensate, the sum itself is positive, contradicting our assumption. Symmetry is clearly upheld by the absoute value sign in the summand. To show subadditivity of the triangle inequality, we can simply fix a, and note that $|X(a) - Y(a)| \leq |X(a) - Z(a)| + |Z(a) - Y(a)|$ by the triangle inequality for real numbers. Since this is true for each term in the summand, it must also be true for the entire sum. Therefore all four conditions for being a metric are satisfied by $\Delta(X, Y)$.

(b) (3 points) Show that the following is an equivalent definition:

$$\Delta(X,Y) := \sup_{A\subseteq\Omega} |X(A) - Y(A)| \ ,$$

where $X(A)$ denotes the probability of X to be in A, and similarly for $Y(A)$. Give an "operational" interpretation to this definition (i.e., in terms of an algorithm trying to distinguish X and Y).

Solution: We car rewrite this alternative definition as $\sup_{A\subseteq\Omega} |\sum_{\omega\in A} X(\omega) - \sum_{\omega\in A} Y(\omega)|$ By the triangle inequality, we have that $|\sum_{\omega\in A} X(\omega) - \sum_{\omega\in A} Y(\omega)| \leq \sum_{\omega\in A} |X(\omega) - Y(\omega)|$ with equality holding when the rank ordering of $X(\omega)$ and $Y(\omega)$ are the same $\forall\omega \in A$. This hints at the fact that the choice of A which maximizes this expression is the set $A^* = \omega \in A : X(\omega) > Y(\omega)$ Note also that this sum is the same if we choose A^{*c} to be our set, so that $|X(A^*) - Y(A^*)| + |X(A^{*c}) - Y(A^{*c})| = 2|X(A^*) - Y(A^*)| = \sum_{\omega\in A^*} |X(\omega) - Y(\omega)| + \sum_{\omega\in A^{*c}} |X(\omega) - Y(\omega)| = \sum_{\omega\in\Omega} |X(\omega) - Y(\omega)|$. Rearranging terms, we get $|X(A^*) - Y(A^*)| = \frac{1}{2}\sum_{\omega\in\Omega} |X(\omega) - Y(\omega)|$, proving the desired equivalence. In order to implement this as an algorithm, we could add some efficiency by only making a single subtraction at the end, rather than subtracting the probabilities for every element of Ω. To do this we simply maintain two separate sums S_x and S_y. As we search trhough each element of Ω we simply check whether $X(\omega) > Y(\omega)$, and if it is, we add $X(\omega)$ to S_x and add $Y(\omega)$ to S_y, otherwise doing nothing. When we have iterated through all possible events, we simply take the difference $S_x - S_y$ in order to find $\Delta(X,Y)$ with only a single subtraction operation. (Note that while we have to make additional comparison operations, this is compensated by the fact that we never have to take the absolute value).

(c) (3 points) Let D_0 and D_1 be two distributions over the same support Ω. Suppose that we play the following game with an algorithm $\mathcal{A}$. First, we pick at random a bit $b \leftarrow \{0,1\}$ and then we pick $x \leftarrow D_b$ and we give x to $\mathcal{A}$. Finally, $\mathcal{A}$ returns a bit $\mathcal{A}(x)$. It wins if the bit returned is equal to b. Show that the highest success probability in this game is exactly $\frac{1}{2} + \frac{1}{2}\Delta(D_0, D_1)$.

Solution: If we consider a bayesian approach, we can imagine that there is some event ω that could have come from distribution D_0 or distribution D_1 with a prior probability of $\frac{1}{2}$ that it came from either one. If we assume perfect knowledge of both distributions, a perfect algorithm would output 0 whenever $D_0(\omega) > D_1(\omega)$, and vice versa. Then for any given event ω, we can use Baye's formula to conclude that the probability of guessing correctly is $\frac{1}{2}\frac{D_0(\omega)}{D_0(\omega)+D_1(\omega)}$, assuming that $D_0(\omega) > D_1(\omega)$. If we wan to take an average of this success probability, we must weight by the probability that the event occurrs in the first place, $D_0(\omega) + D_1(\omega)$. Thus our average success probability can be written as $S = \frac{1}{2}\sum_{\omega\in A^*} x + \frac{1}{2}\sum_{\omega\in A^{*c}} y = \frac{1}{2}\sum_{\omega\in A^*} x - y + \frac{1}{2}\sum_{\omega\in A^*} y + \frac{1}{2}\sum_{\omega\in A^{*c}} y = \frac{1}{2}\sum_{\omega\in A^*} x - y + \frac{1}{2}\sum_{\omega\in\Omega} y = \frac{1}{2}\Delta(D_0, D_1) + \frac{1}{2}$

4. (6 points) (Pairwise independence) Assume that $r_1, \ldots, r_t$ are independent uniform strings in $\{0,1\}^n$. Show that the collection of all $2^t - 1$ nontrivial XORs, $\{\bigoplus_{i\in S} r_i\}_{\emptyset\neq S\subset[t]}$ is pairwise independent, i.e., any two of them are jointly distributed like an independent uniform pair of strings in $\{0,1\}^n$.

Solution: For any given output of the XOR operation, there are 2^n possible pairs of input strings that could have produced it, since for each digit, if the result is 0, the strings could have both had a 0 or a 1, and if the result is 1, one of the input strings had a 0 and the other had a 1, in either order. Since the probability of picking a given pair that is uniformly distributed is 2^{-2n}, the probability of a given XOR result is $\frac{2^n}{2^{2n}} = \frac{1}{2^n}$. Thus the distribution of XOR results is the same as that of the input strings; namely, a uniform distribution.

5. *(Large deviation bounds.)* Assume that $X_1, \ldots, X_n$ are independent identically distributed (i.i.d.) random variables, each taking 1 with probability p and 0 with probability $1-p$. Recall that Chernoff's bound says that for all $\varepsilon > 0$,

$$\Pr\left[\left|\frac{1}{n}\sum_i X_i - p\right| > \varepsilon\right] \leq 2e^{-2n\varepsilon^2} .$$

If you are rusty on Chernoff's bound, read about it, e.g., here or search Google; there are lots of forms of the bound, the above being the most convenient for our applications.

(a) (2 points) How large should n be if we want the average of the X_i to be within $\pm\varepsilon$ of p with probability at least $1-\delta$? (asymptotic expression for n is enough)

Solution: We can easily see from the defintiion that the limiting case happens when $\delta = 2e^{-2n\varepsilon^2}$. Solving for n, we find that the limiting value of n occurrs when $n = \frac{-1}{2\varepsilon^2} log(\frac{\gamma}{2})$.

(b) (3 points) Imagine we used Chebyshev's bound instead of Chernoff's, and if you wish, assume for simplicity that $p = 1/2$. What bound on n would you get then? Do you see any advantage of Chebyshev's bound over Chernoff's?

Solution: Chebyshev's inequality tells us that $P[X - \mu| < k\sigma] \leq \frac{1}{k^2}$, or equivalently, $P[|X - \mu| < \varepsilon] \leq \frac{\sigma^2}{\varepsilon^2}$. Now if we define $X = \frac{1}{n}\sum_i X_i$, using the fact that $p = \frac{1}{2}$ we have from the central limit theorem that $\sigma^2 = \frac{1}{4n}$. Now we have the limiting case and can solve for n: $n = \frac{1}{4(1-\delta)(\varepsilon^2)}$ You should use Chebyshev's inequality when p is small, and Chernoff's bounds when p is large. Plotting on wolfram alpha shows that the point above which Chernoff's bounds become more advantageous occurrs at roughly p=0.6.

6. *(Error-correcting codes (optional, no credit).)* This is a bit off topic, but will give you an idea of the kind of math we use in this course. It will also give you a glimpse to an immensely important topic that also dates back to Shannon's seminal work. These ideas are used in pretty much all digital communication protocols: cell phones, Internet, satellites, etc.

(a) Assume we choose $2^{n/20}$ strings from the set $\{0,1\}^n$ uniformly at random. Show that with positive probability (in fact, high probability) the Hamming distance (i.e., number of different coordinates) between *any* two strings in the set is more than $n/4$. I need a hint! (ID 84542)

Solution:

(b) Show how Alice can communicate to Bob a message of k bits by sending only $n = 20k$ bits in such a way that Bob can recover the message even if an adversary flips up to $n/8$ bits of the communication. Would simply repeating the message 20 times be good enough?

Solution: