6.857
NETWORK AND COMPUTER SECURITY
PROBLEM SET 3

JOHN WANG

Problem 3.1

Problem 3.1.a. To identify Z, we will take a number of samples of Z and average the results of the samples. We will choose X or Y depending on whether the average is closer to the expected value of X or Y. We can also do this by choosing an output depending on whether or not the average of the samples is greater or less than a particular threshold.

First, notice that the distributions of X and Y are identical binomial distributions, except that Y's probability density function is shifted up by 1 from X's probability density function. We want to be able to find a threshold value γ for which we will choose either X or Y depending on whether the sampled average is above or below the threshold γ. Since X and Y have symmetric distributions, the best threshold value that can be picked is the value at which the two distributions intersect.

This is because it minimizes the probability of an incorrect choice. To see this, say p_X and p_Y are the probability density functions of X and Y respectively. Then we want to choose a threshold value γ which minimizes $\int_{\gamma}^{\infty} p_X(x)dx + \int_{-\infty}^{\gamma} p_Y(x)dx$. We know that $p_Y(x) = p_X(x-1)$ for all x, so that we really want to minimize:

$$\min_{\gamma} \int_{\gamma}^{\infty} p_X(x)dx + \int_{-\infty}^{\gamma} p_X(x-1)dx \tag{1}$$

One could use the fact that $p_X(x) = \binom{t}{x} p^x (1-p)^{t-x}$ where $p = 1/2$ and explicitly minimize the above expression. However, it is clear that the minimum is achieved where the distributions intersect. Concretely, this is where the error of the Y distribution is equal to the error of the X distribution.

Thus, we want to choose a threshold $\gamma = (E[X] + E[Y])/2$. Since we know that $Y = X + 1$, we can use linearity of expectation to derive $\gamma = (2E[X] + 1)/2 = E[X] + 1/2$. Since X is a binomial random variable with $p = 1/2$, we have $E[x] = t/2$. Thus, our algorithm is as follows:

```
Take n samples of Z and compute the average q.
If q > t/2 + 1/2:
  return Z = Y
else:
  return Z = X
```

Now, let us determine how many samples n we need in order to be confident in our result. First, let us say we want to pick the incorrect Z with probability less than some c. For this, we shall use Chebyshev's inequality: $P(|K - E[K]| > \alpha) \leq \frac{Var(k)}{\alpha^2}$.

To do this, we define $K_X = \frac{1}{n}\sum_{i=1}^{n} X_i$ as the sample average of n trials of X and $K_Y = \frac{1}{n}\sum_{i=1}^{n} X_i$ as the sample average of n trials of Y. We note that our algorithm chooses an incorrect result when $K_X - E[K_X] \geq \frac{1}{2}$ or when $E[K_Y] - K_Y \geq \frac{1}{2}$. Firt, we will set $K = K_X$ and use the Chebyshev inequality to find the following:

$$P(incorrect) = P\left(|K_X - E[K_X]| \geq \frac{1}{2}\right) + P\left(|K_Y - E[K_Y]| \geq \frac{1}{2}\right) \tag{2}$$

$$\leq \frac{Var(K_X) + Var(K_Y}{(1/2)^2} \tag{3}$$

$$= \frac{2Var(K_X)}{1/4} \tag{4}$$

$$= 8Var(K_X) \tag{5}$$

Where we noticed that $Var(K_X) = Var(K_Y)$ by the fact that they are the same distribution just shifted. Thus, we simply need to find some n such that $8Var(K_X) = c$ in order to pick incorrectly with probability less than c. To choose n, we must find $Var(K_X)$. Since we know that each trail X_i is independent, we know their variances add:

$$Var(K_X) = Var\left(\frac{1}{n}\sum_{i=1}^{n} X_i\right) \tag{6}$$

$$= \frac{1}{n^2} Var\left(\sum_{i=1}^{n} X_i\right) \tag{7}$$

$$= \frac{1}{n^2}\sum_{i=1}^{n} Var(X_i) \tag{8}$$

$$= \frac{1}{n^2}\sum_{i=1}^{n} \frac{1}{4}t \tag{9}$$

$$= \frac{t}{4n} \tag{10}$$

Where we know that X_i is a binomial random variable so that its variance is given by $tp(1-p)$ where p is the probability of flipping a heads. Thus $Var(X_i) = t(1/2)(1-1/2) = \frac{1}{4}t$. Thus, we only need to solve the following equation for n:

$$8Var(K_X) = c \tag{11}$$

$$8\frac{t}{4n} = c \tag{12}$$

$$n = \frac{2t}{c} \tag{13}$$

Thus, if we choose $c = 0.01$ so that there is a 1% chance that we are incorrect with our choice of Z, then we must make $200t$ trials.

0.1. **Problem 3.1.b.** To recover the secret key k, we use the result from above. We know that there are 1408 total possible "coin flips" in the AES algorithm during the XOR stages. We expect 704 total ones bits. To get the first bit of the secret key, we select only triples where the first bit of the plaintext input is 1. In this case, if the secret key's first bit is a 1, then the first bit of the XOR state will be zero. If the secret key's first bit is a 0, then the first bit of the XOR state will be one. Thus the first XOR bit depends entirely on the first bit of the secret key, but there are $t = 1407$ other XOR bits which are randomly determined when we look at input with 1's in the first bit of the plaintext.

We expect there to be $1407/2 = 703.5$ ones if the first bit of the secret key is 1 and $703.5 + 1 = 704.5$ ones if the first bit of the secret key is 0. By the reasoning from the first problem, we know that can choose a threshold of 704 to determine whether the secret key is 1 or 0.

For the ith bit of the secret key, we do the same thing, except we select only inputs where the ith plaintext bit is 1. The algorithm is as follows:

```
for ith bit in the secret key:
  average the ones for triples where ith bit of plaintext is 1
  if average > 704:
    set ith bit of secret key to 0
  else:
    set ith bit of secret key to 1
```

0.2. **Problem 3.1.c.** We can run the algorithm, and it provides us with the following key (16 bytes in an array):

```
[89, 111, 117, 32, 109, 97, 100, 101, 32, 105, 116, 32, 116, 101, 97, 109]
```

Converted using ascii, this reads as:

```
['Y', 'o', 'u', ' ', 'm', 'a', 'd', 'e', ' ', 'i', 't', ' ', 't', 'e', 'a', 'm']
```

0.3. **Problem 3.1.d.** We used 123,830 triples. However, for each bit, only about half of those triples will be useful (since the other half will be 0s). Thus we can set $n = 123830/2 = 61915$. Using the result from first part of this problem that $\frac{2t}{c} = n$, we can compute the upper bound on the probability that we obtained an incorrect key c. Since $t = 1407$, we have $\frac{2t}{n} = \frac{2\times1407}{61915} = 0.044$. Thus, we only have a 4% chance of getting an incorrect answer.

To obtain a reasonable incorrectness probability (say 20%), we would only need to have used $\frac{2\times1407}{0.2} = 14070$ triples. However, the 14070 triples must be doubled because half of them will be unusable (will have a 0 digit in the ith bit). Thus, to get 20% incorrectness probability, we must obtain about 28140 triples.