1.

a) Given rule:

Prove

Starting with RHS, we have:

(i)

// Using given rule

// Linearity of Expectations

// Law of Iterated Expectations

(ii)

// Definition of variance

// Law of Iterated Expectations

Add parts (i) and (ii) together.

// Definition of variance

b) Recall the proven formula in part a:

Here, and .

(i)

// Condition on N

// Variance of sum of iid RV

// Placing into expected value and using fact that all come from same distribution and has same variance

// Simplify

(ii)

// Linearity of Expectations

// Variance of Multiplied Number

Add parts (i) and (ii) together.

2. As a base case, suppose there is only coins. Since there is only one coin to flip, that very coin must be the one fair coin with , where *p* and *q* denote the probability for a specific coin to turn heads or tails, respectively. The number of even heads can only be zero i.e. the coin turns tail. That means , where denotes the probability of getting an even number of heads in a total of 1 coin flip.

Now suppose there is a total of an arbitrary *n* coins. It follows that:

This says that the probability of even coins being heads in *n* flips is the probability of *n*-1 coins having even heads with the *nth* coin being tails or *n*-1 coins having odd heads with the *nth* coin being heads to make a total of even number of heads.

Suppose that the fair coin is already present in the first *n*-1 coins. This would indicate . Recall for , the corresponding as proven before in the base case. By induction, for etc. The other scenario is if the fair coin is the last, or *nth* , coin. This means that is not necessarily, and most likely not, . The probability for even number of heads now is . Therefore, regardless of the position that the fair coin comes in the sequence of *n* coins, the probability of the even heads is still .

3.

a)

// Joint probability

// Inner integral

// Simplify previous step

b) Let *t* denote an arbitrary value such that both and are true. Then logically, the minimum between *X* and *Y* is also greater than *t*.

The CDF would be .

Following the form for CDF of the exponential distribution, the mean is easily seen being .

c) // Definition of conditional probability

(i) numerator

// Based on CDF computed from part (b)

(ii) denominator

Note that in part (a), we proved that . The denominator can be verified by .

Combine numerator and denominator into fraction.

Since it is conditioned that , the above equation can also replace *z* with *x*.

d) Just like in part (c), conditioning on would only consider a distribution when . Similar to part (c), a convolution of the individual density functions, in this case denoted by as the integrand, would be computer such that is taken into account. This would yield another exponential distribution in the end by looking at the previous part.

Due to the memoryless property of exponential distribution, the difference comes from the same distribution as *Y*, without needing to known where the minimum i.e. *X* even occurred prior to *Y*. Since distribution of *Y* is known to have mean , this is also the case for .

4.

a) //Definition of EV

// Integration by parts

Note that

b) Use the identity:

The identity can be re-arranged to compute

The following parts are known or computed using properties of the exponential distribution.

// Definition of expected value

// Integration by parts

Putting the formula together.

5.

a) Because of the memoryless property of the exponential distribution, the first person between A and B to be done being served still has mean at the point C begins being served. Person C also has mean during the start of being served, as stated by the problem. Since both the first person to be finished and person C have the same mean time being server i.e. , the probability that person C leaves last is .

b) Let R1 and R2 be the remaining time left for clerks 1 and 2 with their customers.

The third customer, C, would leave after addition time spent on top of the first of 2 customers to be finished.

c)

6.

a) Removing the assumption of independent features, the same model becomes:

Let .

With the assumption of independent features, we can arrive at by simply multiply the feature weights of *D* features together per class, yielding parameters. We cannot do that now since independence is not valid for feature. For each of *C* classes, we can instead lookup given using the *D* features as a binary index or binary string as key. For a feature space of length *D*, there are 2D unique binary indexes that are possible. Therefore, the number of parameters over *C* classes is .

b) For one instance :

Note that each instance has its own unique that is looked up in the table described in part (a). Because features are not iid random variables, cannot be broken down into a product along feature subscript *j* as seen in NBC.

Likelihood function:

There can only be one specific *D*-bit vector and class per instance , therefore the two (2) inner product operators are dismissed for simplification.

Log-likelihood function:

c) The class distribution is given as uniform, so .

d) Runtime complexity is for both NBC and the full model. Both algorithms require iteration through *N* instances, under which there is an iteration through *D* features. Note that this accounts for the formulation of binary index in time.

Memory space is for NBC. can be stored in a matrix. And then there is the dataset of *N* instances. For the full model, memory space is . The only difference here is that for each class table, there can be up to binary indexes.

e) To use plug-in approximation, test each of *C* classes to find

We are already given that , so no need to compute from scratch.

Knowing , find estimate that maximizes the expression, which can be re-expressed below.

Maximizing yields 1 as before.

For a single instance to classify, runtime is for NBC and for full model.

7.

a) Note that we can use the formula .

b) It can be shown that

Using this formula, we have