UNIVERSITÉ BISHOP'S UNIVERSITY

Statistical Learning-CS507

Madjid Allili ∼ J125 ∼ ext. 2740 ∼ mallili@ubishops.ca

Assignment 2: Due October 11th, 2017

1. Now that you have answered the questions of the first assignment, you should submit the program in a R-script file that allowed you to determine all the answers starting from the instruction that loads the dataset "data.csv".

   The R-script file should take the form:

   ```
   #Reading the data file:
   Data <- ...
   #Question 1 - displaying the column names:
   ...
   #Question 2 - the first two rows are:
   ⋮
   ```

2. Calculate the following probabilities using R

   (a) Probability that a normal random variable with mean 22 and variance 25

   i. lies between 16.2 and 27.5
   ii. is greater than 29
   iii. is less than 17
   iv. is less than 15 or greater than 25

   (b) Probability that in 60 tosses of a fair coin the head comes up

   i. 20, 25 or 30 times
   ii. less than 20 times
   iii. between 20 and 30 times

   (c) A random variable X has Poisson distribution with mean 7. Find the probability that

   i. X is less than 5 less or equal is:
   ii. less than is
   iii. X is greater than 10 (strictly)
   iv. X is between 4 and 16

3. Suppose that Joe draws k balls from an urn containing n red balls and n green balls, without replacing the balls after they are drawn. Similarly, Mary draws k balls from an urn containing m red balls and m green balls, without replacing the balls after they are drawn. We want to compute the probability that Joe and Mary will draw the the same number of red balls.

   (a) Write an R function to compute this, which takes n, m, and k as arguments. (These arguments must be positive integers, and $2n$ and $2m$ must be at least as big as k, but you don't have to check for this in your program).

   This function should use one or more of the `sum`, `prod`, `factorial`, and `choose` functions.

   Note that `factorial` and `choose` can take vectors as arguments, and then return a vector of results.

   Note also that a vector that is a sequence can be created with an expression like `i:j`.

   (b) Test your function on at least the following values for the arguments:

      i. n = 20, m = 30, k = 1
      ii. n = 20, m = 30, k = 2
      iii. n = 200, m = 300, k = 2
      iv. n = 2000, m = 3000, k = 2
      v. n = 50, m = 60, k = 17

   (c) Comment on the results. Can you see why some of them are what they are (at least approximately) with simple calculations?

   (d) Try the function you wrote above with `n = 600`, `m = 500`, and `k = 400`. You should see a result of `NaN`, indicating that floating point overflow occurred as some point in the computation, so the final result was meaningless.

   Write a new version of the function that avoids this problem by working in terms of the logarithms of the values, until the final result needs to be computed.

   The (natural) logarithm is computed with the log function, and its inverse, the exponential function, is computed with `exp`.

   The `lfactorial` and `lchoose` functions compute the log of the factorial and choose functions.

   Test your new function on the same sets of arguments as above, for which it should produce the same (or very close to the same) answer, as well as on `n = 600`, `m = 500`, and `k = 400` and on `n = 6000`, `m = 5000`, and `k = 4000`, for which it should not produce `NaN`.

4. You roll a fair six-sided die 100 times. Let X be the random variable that is the number of these rolls that show 1, 2, or 3. You then roll the die another 100 times. Let Y be the random variable that is the number of these rolls that show 1 or 2. With simple calculations in R, using the `dbinom` and `pbinom` functions, find the following:

   (a) $P(X \leq 60)$.

   (b) $P(Y \geq 60)$.

   (c) $P(X = Y)$.

   (d) $P(X > Y)$.

   Hand in your R commands (which should consist of one line for each question) and their numerical outputs.