ABE6933 SML HW1

Directions

Please submit **ONE PDF** file including all your reports (answer + code + figures + comments; must be easily readable and have file size under a few megabytes) and **ONE R code script**. The R script is supplementary material to ensure that your code runs correctly. If you are using RMarkdown, please also include your `.Rmd` file.

Place these two (or three) files in a folder, make a zip or rar archive, and submit the archive electronically via Dropbox file request at `tinyurl.com/nbliznyuk-submit-files` (on the landing page, enter your name so that we know it is you and email so that you get a confirmation).

For the full list of rules/policies/expectations, please visit "hw.rules.pdf" document.

Deadline: 24-Sep-2020, 11:59 PM EST.

Practice/Optional Problems (do not submit)

1. Practice with pdfs and cdfs: integrate uniform and exponential pdfs to obtain the cdfs, then differentiate your cdf expressions; make sure you get the same results (the correct pdfs); be super cautious about the range of possible values (aka support of a rv). This should be done by whatever means necessary, e.g., by analytical or numerical integration/differentiation.
2. Try to repeat the preceding exercise for normal and general gamma pdfs (can you find closed-form math expressions?); appreciate the fact that we can work with pdfs rather than cdfs
3. Verify the factorization criterion for independence in the pdf form (i.e., establish equivalence of pdf & cdf versions) for 2 rvs.
4. (a) MLE by hand: try to do analytical maximization of the likelihood rather than the log-likelihood for a problem of your choice (any sample size of 10 or greater);

   (b) MLE graphically/numerically using R: try to plot your likelihood function in 4(a); compare with the plots of log-likelihood; which one is better behaved?
5. Implement (by hand, without using external R libraries) the log-likelihood for the bivariate normal data model. Use numerical optimization to obtain the MLE of the 5-dimensional parameter vector (muv, sigv, rho). Refer to "2020.09.15.demo.MLE.R" under the code folder for details on how to generate the data.

Required Problems (for submission)

1. (Inefficient) Implementation by hand a nearest-neighbors-like classifier with 2d features.

Your goal is to provide your own implementation of a NN method "by hand", i.e., writing your own functions rather than using existing third-party libraries. The data for this problem is in file "SML.NN.data.csv". The columns are Y (the response - class 0 or 1), X_1 and X_2 (features, horizontal and vertical "coordinates") and set identification ("train", "valid" or "test"). Approach: calibrate/tune the parameter(s) using the calibration ("valid") set; report performance on the left-out "test" set.

1.1: Write a function `getClass1Prop(x, r)` with inputs x and r that outputs the proportion of class 1 among observations of the training data that are within radius r from point x. The function would return

NA if there are no points within radius r. Use the Euclidean distance when defining proximity. You will use this function in later subproblems to make predictions (0 or 1) using thresholding: predict 1 if the class 1 proportion is 0.5 or higher; else predict class 0.

1.2: Write a function that, for a fixed radius r, computes misclassification rate over the validation data. I.e., for each x in the "valid" set, obtain prediction $\widehat{y}(x)$, compare it with the true $y(x)$ and compute the proportion of incorrect predictions.

1.3: Explore the "train" and "valid" data. What would be your guess(es) about the good values of r for accurate out-of-sample classification? (Record those for future comparisons in 1.4).

1.4: Compute the misclassification rate (from 1.2) for a grid of r values (e.g., from 0.01 to 1 with step size 0.01) and plot. Find the value of r that achieves the lowest misclassification rate; call it r^*. Use r^* to obtain misclassification rate on "test" set. Compare it with the misclassification rates using your guesses in 1.3; briefly discuss.

1.5*: Optional - optimize your code organization to reduce the use of loops.

1.6*: Optional - contrast this "fixed radius r" approach with the "fixed number of neighbors K" approach.

2. Factorization criterion in action in the special case of the bivariate normal pdf.

2.1. Find the marginals (i.e., the marginal pdfs of X and Y from the joint pdf on slide 11 from 2019/08/29 under "old/lectures" or p.18 of 2016.08.24.statlearn.review.pack.01.pdf under "[3].handouts.to.print.or.comment.electronically"). If you are unable to do this analytically (which is fine, no penalties), assume $\mu_X = \mu_Y = 0$, $\sigma_X = \sigma_Y = 1$, and $\rho = 0.5$; specifically, use numerical integration to find the values of the marginal pdfs on a fine grid from on the interval [-3,3], plot those and compare with Normal(0,1) pdf.

2.2. Show that if $\rho = 0$ then the rvs are independent. You can use the fact established in 2.1 that, marginally, X is normal with mean μ_X and variance σ_X^2 (similarly, for Y).

2.3. Assume $\mu_X = \mu_Y = 0$ and $\sigma_X = \sigma_Y = 1$; let ρ be general (strictly between -1 and 1). For values of ρ on the grid from -0.75 to 0.75 with step size 0.25, plot the conditional pdf of X given that $Y = 1$. If $Y = 1$ is the observed value, does the correlation (positive or negative) help one predicting X, relative to the case of $\rho = 0$? Briefly discuss.

3. MLE with data from exponential distribution.

Let $X_1, \ldots, X_{100}$ be independent rvs from the exponential distribution with rate λ (i.e., rate is the reciprocal of the population mean here). Nature uses the following code to generate the data:

```
set.seed(0); x = rexp(100,10);
```

I.e., in the game theory setup, Nature choses $\lambda = 10$, but this is not known to the Statistician.

3.1. (Computational) Use the examples from class to estimate λ using the method of maximum likelihood. Show all steps.

3.2. (Optional - analytical) Use calculus to obtain the (expression for) MLE of lambda. Show all steps (including checking the second-order conditions presented in class).

4. Exact and approximate small-sample CIs for the mean.

For each of the cases 1-3 below, complete the steps (a)-(e) below. For j=1,...,1000,

(a) set the random seed to j,

(b) generate a random sample of size 4 from Normal(0,1)

(c) compute a 2-sided 95% CI for the mean.

(d) record whether the CI contains the true mean (=0).

(e) for cases 2 and 3, also store the length of the CI

Case 1: sig2 is known and is equal to 1.

Case 2: sig2 is unknown and exact small sample CI is computed in (c) [using t-distribution quantiles]

Case 3: sig2 is unknown and an approximate large-sample CI is computed in (c) [using Normal(0,1) quantiles]

Report

(1) the "empirical frequency of coverage" (i.e., the average of (d))

(2) histogram of interval widths (when appropriate).

Discuss your findings (particularly, try to relate (1) and (2)).