Problem Set 2

Issued: Thursday 12$^{\text{th}}$ October, 2017 **Due:** Friday 20$^{\text{th}}$ October, 2017

Notations: x, y are random variables. If they are discrete, $\mathcal{X}$ and $\mathcal{Y}$ are corresponding alphabets. If x is continuous, $p_{\mathsf{x}}(x)$ represents the PDF (Probability Density Function) of x. $\underline{\mathsf{x}}, \underline{\mathsf{y}}$ are random vectors. The notation "★" represents optional problems with bonus points.

2.4. *From probability to statistics.* The main difference between the "probability" and "statistics" is that statistics deal with samples of random variables. For example, the mathematical expectation for discrete random variable x which takes value from finite alphabet $\mathcal{X}$ is

$$\mathbb{E}[\mathsf{x}] = \sum_{x \in \mathcal{X}} P_{\mathsf{x}}(x)x, \tag{1}$$

while the empirical average based on n samples is

$$\hat{\mathbb{E}}[\mathsf{x}] = \frac{1}{n}\sum_{i=1}^{n} x^{(i)} \tag{2}$$

$$= \sum_{x \in \mathcal{X}} \hat{P}_{\mathsf{x}}(x)x, \tag{3}$$

where $x^{(i)}$ is the i-th sample of x, and $\hat{\mathbb{E}}[\cdot]$ means that this expectation is calculated based on empirical distribution $\hat{P}_{\mathsf{x}}(\cdot)$. $\hat{P}_{\mathsf{x}}(\cdot)$ is the empirical distribution, could be considered as the histogram of data samples:

$$\hat{P}_{\mathsf{x}}(x) \triangleq \frac{1}{n}\sum_{i=1}^{n} \mathbb{1}_{x^{(i)}=x}, \quad \forall x \in \mathcal{X}.$$

For continuous random variables, situations would be a little more complicated, but Equation (2) is still valid. Now we would like to conduct a series of simulations to process samples of random variables.

We start with a two dimensional standard Gaussian vector $\underline{\mathsf{z}} \sim \mathcal{N}(\underline{0}, I_2)$. Assume we have n independent samples of $\underline{\mathsf{z}}$: $\underline{z}^{(i)}, i = 1, 2, \cdots, n$. Use MATLAB to finish the following tasks.

(a) Let $\underline{\mathsf{x}} = (\mathsf{x}_1, \mathsf{x}_2)^{\mathrm{T}} = C\underline{\mathsf{z}}$, where $C \in \mathbb{R}^{2\times 2}$. The matrix C is chosen such that $\mathrm{Var}(\mathsf{x}_1) = \mathrm{Var}(\mathsf{x}_2) = \sigma^2, \rho(\mathsf{x}_1, \mathsf{x}_2) = \rho_{\mathsf{x}}$.

   i. Write a MATLAB function `K = getK(simga, rho)` to calculate K_{x} for given σ and ρ.

   ii. Write a MATLAB function `C = getC(K)` to calculate C for given K_{x}.

   iii. Now we could get $\underline{x}^{(i)}$ based on $\underline{x}^{(i)} = C\underline{z}^{(i)}$, as the samples of $\underline{\mathsf{x}}$. From these samples, we could plot the histogram of $\underline{\mathsf{x}}$. Since the histogram is 3 dimensional, we plot its contour instead. Write a function `plot_hist_contour(X)` to plot the contour of $\underline{\mathsf{x}}$'s histogram, where X is a data matrix formed by all samples $\underline{x}^{(i)}$:

$$X = \begin{bmatrix} \underline{x}^{(1)\mathrm{T}} \\ \underline{x}^{(2)\mathrm{T}} \\ \vdots \\ \underline{x}^{(n)\mathrm{T}} \end{bmatrix}. \tag{4}$$

In our MATLAB script, the samples are always presented by such data matrices. Now you can use your function `plot_hist_contour` to plot the contours of the histogram of $\underline{\mathsf{x}}$ for the following cases:

- $\sigma = 1, \rho_{\mathsf{x}} = 0.3$
- $\sigma = 1, \rho_{\mathsf{x}} = 0.8$
- $\sigma = 1, \rho_{\mathsf{x}} = -0.8$

Hint: You may use MATLAB's `hist3` and `contour` function.

For the following parts, we will assume $\sigma = 1, \rho_{\mathsf{x}} = -0.8$.

(b) From above practice, we have generated samples of $\underline{\mathsf{x}} \sim \mathcal{N}(\underline{0}, K_{\mathsf{x}})$. In fact, MATLAB has a built-in function `r = mvnrnd(MU,SIGMA,cases)` to generate independent samples of multivariate normal distribution with given mean vector and covariance matrix. We now use it to generate n samples of $\underline{\mathsf{w}}$, which has the same mean and covariance as $\underline{\mathsf{x}}$.

i. Plot the contours of the histogram of $\underline{\mathsf{w}}$. Does it have the same distribution as $\underline{\mathsf{x}}$?

ii. Write a function `Exy = getExy(X, Y)` to calculate $\hat{\mathbb{E}}[\underline{\mathsf{x}}^{\mathrm{T}}\underline{\mathsf{y}}]$ from corresponding data matrices X and Y, where the data matrices are formed by their samples as defined in (4). Similar to (2), $\hat{\mathbb{E}}[\underline{\mathsf{x}}^{\mathrm{T}}\underline{\mathsf{y}}]$ could be calculated using

$$\hat{\mathbb{E}}[\underline{\mathsf{x}}^{\mathrm{T}}\underline{\mathsf{y}}] = \frac{1}{n}\sum_{i=1}^{n}\langle \underline{x}^{(i)}, \underline{y}^{(i)}\rangle.$$

iii. Use your `getExy` to calculate $\hat{\mathbb{E}}[\underline{\mathsf{x}}^{\mathrm{T}}\underline{\mathsf{z}}]$ and $\hat{\mathbb{E}}[\underline{\mathsf{w}}^{\mathrm{T}}\underline{\mathsf{z}}]$ and briefly analysis the results.

(c) For given K_{x}, write a MATLAB function `V = getV(K)` to calculate the projection matrix $V = [\underline{v}_1, \underline{v}_2]$, where $\underline{v}_1, \underline{v}_2$ is defined as in Problem 2.2. As in Problem 2.2, $\underline{\mathsf{y}} \triangleq V^{\mathrm{T}}\underline{\mathsf{x}}$. Then we would get corresponding $\underline{y}^{(i)}$ as samples of $\underline{\mathsf{y}}$: $\underline{y}^{(i)} = V^{\mathrm{T}}\underline{x}^{(i)}$. Using these samples to plot the contour of the histogram of $\underline{\mathsf{y}}$.

Hint: You may use MATLAB's `svd` or `eig` function.

The following parts are based on Problem 2.3. You need to define your own MATLAB functions to finish these tasks.

(d) i. Write the expression of the conditional PDF $p_{\mathsf{x}_1|\mathsf{x}_2}(x|0)$.

ii. Based on samples $\{\underline{x}^{(i)}\}_{i=1}^{n}$ (i.e., the data matrix X) to plot the histogram $\hat{p}_{\mathsf{x}_1|\mathsf{x}_2}(x|0)$, as a estimation of $p_{\mathsf{x}_1|\mathsf{x}_2}(x|0)$.

iii. Plot the true PDF $p_{\mathsf{x}_1|\mathsf{x}_2}(x|0)$ in the same figure and compare your histogram with true PDF.

Hints:

- For real data, $x_2^{(i)} = 0$ is a too strong condition for you to find any point with $x_2^{(i)} = 0$. So you can just pick samples with $x_2^{(i)}$ being very close to 0.
- I have an example of plotting histogram and comparing it with actual PDF in the LaTeX template for homework. You can refer to it.

(e) (★) Since $\mathbb{E}[\mathsf{x}_1|\mathsf{x}_2]$ is a function of x_2, we could generate samples of $\mathbb{E}[\mathsf{x}_1|\mathsf{x}_2]$ based on $\{\underline{x}^{(i)}\}_{i=1}^n$.

  i. For each $\underline{x}^{(i)}$, calculate corresponding $\hat{\mathbb{E}}[\mathsf{x}_1|\mathsf{x}_2 = x_2^{(i)}]$ from data. So now you would get n samples of $\mathbb{E}[\mathsf{x}_1|\mathsf{x}_2]$.

  ii. Plot the histogram of above samples and compare this histogram with the actual PDF of $\mathbb{E}[\mathsf{x}_1|\mathsf{x}_2] = \rho\mathsf{x}_2$.

SUBMISSION REQUIREMENTS

- You need to submit all your source files and your report (a single PDF document that contains all figures and necessary interpretation) as a zip file through Web Learning. The contents of the zip file should be organized as follows:

  - `Problem2.4.zip` Your zip file.
    - `problem2.4.pdf` This is your report.
    - `src` Folder for all your .m files.
      - `iips2.m` Main script file
      - `getC.m` Matlab functions
      - `getK.m`
      - $\cdots$ Your other functions

  Where `iips2.m` is the main script that you need to work on. All results (figures and numerical results) should be given when running this script.

- **Do not** change the settings in `iips2.m`, such as `n = 1e6`. However, you could add more input arguments to functions to help your plotting. For example, my second input argument of `plot_hist_contour()` is a string used to labeling axises.

- Your total running time of the main script `iips2.m` should be short, say, within 10 seconds.

- The acknowledgment policy as stated before.

Below is the source code of main script `iips2.m`. You can also find a separate file in the folder.

```
%%%% iips2.m %%%%
n = 1e6;
dim = 2;
Z = randn(n, dim); % Generate Random Gaussian.
%% (a)
sigma = 1;
rho_vec = [0.3, 0.8, -0.8];
figure;
for i = 1 : 3
    subplot(1, 3, i);
    rho = rho_vec(i);
    K = getK(sigma, rho);
    C = getC(K);
    X = Z * C'; % (C * z')'
    plot_hist_contour(X, 'x');
    title(strcat('$\rho = ', num2str(rho_vec(i)), '$'), 'Interpreter', 'LaTeX', 'Fontsize', 15);
end
%% (b)
W = mvnrnd(zeros(2, 1), K, n);
Ezw = getExy(Z, W);
Ezx = getExy(Z, X);
disp([Ezw, Ezx]);
figure;
plot_hist_contour(W, 'w');
%% (c)
V = getV(K);
Y = X * V;  %(V' * x')'
figure;
plot_hist_contour(Y, 'y');
%% (d)
% Write your code here
%% (e) (Optional)
% Write your code here
```