Homework 1

Problem 1 Consider a 2 dimensional random Gaussian vector,

$$\mathbf{x} \equiv \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} \sim \mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\Sigma}),$$

where $\boldsymbol{\Sigma} = \begin{bmatrix} \sigma_1^2 & \rho\sigma_1\sigma_2 \\ \rho\sigma_1\sigma_2 & \sigma_2^2 \end{bmatrix}$, and $-1 < \rho < 1$.

1. (10') Derive the joint entropy $H(\mathbf{x})$
2. (10') Derive the mutual information $I(x_1; x_2)$
3. (10') If ρ can be varied, when is the joint entropy maximized? what is the mutual information between x_1 and x_2 then?

Problem 2 We talked about cross-entropy in the class. That is,

$$H(p, q) = -\mathbb{E}_{z\sim p}[\log q(z)].$$

In a C-class image classification problem, we have N samples. Denote the i-th sample as $(\mathbf{x}_i, y_i)$, where $\mathbf{x}_i$ is image and $y_i \in \{0, \ldots, C-1\}$ is its class label. Suppose our model predicts class probability as $\hat{p}_i(y)$ where $y \in \{0, \ldots, C-1\}$.

1. (10') Show that (an empirical estimate of) cross-entropy is

$$\frac{1}{N}\sum_{i=1}^{N} -\log \hat{p}_i(y_i)$$

2. Let us denote the above cross-entropy loss as ℓ. We often report ℓ as an indicator of model quality. A model with lower ℓ is more accurate. However, in some applications, we also report a related metric, called *perplexity* (PPL), defined as

$$PPL = e^{\ell}.$$

   Show that:

   1) (5') A perfect model has $PPL = 1$.

   2) (5') Any reasonable model should have $PPL < C$.

Problem 3 Install numpy, matplotlib and pytorch/tensorflow in your laptop/desktop. Attach your code for all sub-problems below.

1. (10') Plot the following function

$$f(\mathbf{x}) = \frac{1}{2}\left\| \begin{bmatrix} 2 & 1 \\ -1 & 3 \\ 0 & -2 \end{bmatrix} \mathbf{x} - \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix} \right\|_2^2,$$

   on mesh grid defined upon $-1 \leq x_1, x_2 \leq 1$.

2. (10') Use pytorch/tensorflow's autograd ability to get the gradient at $\mathbf{x} = \begin{bmatrix} 1 \\ -1 \end{bmatrix}$
3. (30') Write a program (using pytorch/tensorflow's autograd) to search for the minimizer of $f(\mathbf{x})$. Call it $\mathbf{x}^*$. Also report the corresponding $f(\mathbf{x}^*)$.