Research Internship
Visual Information Processing Lab - ECE, IISc
(Coding Questions)

1 Spearman's Rank-Order Correlation

One of the research topics that our lab mainly focuses on is Image/Video Quality Assessment (IQA/VQA). We develop methods that estimate the perceptual quality of a given input image. The main goal is to ensure that the estimated quality score correlates well with human judgment. For this, we collect opinion scores from human subjects asking them to rate the perceptual quality of a given image on a continuous scale of 0 to 1. We quantify the final quality score of an image as the mean of opinion scores over multiple human subjects (Mean Opinion Score (MOS)). Therefore, while evaluating our methods, we compute a correlation metric between the model's estimate of the quality and the MOS.

One such correlation metric is Spearman's Rank-Order Correlation (SROCC). Let's say we have N images, $\{x_1, x_2, ..., x_N\}$ are the MOS values for each image and $\{y_1, y_2, ..., y_N\}$ are the estimated quality scores from the model on each image. First, we generate a list of ranks for each set. Let x'_i be the rank of x_i over $\{x_1, x_2, ..., x_N\}$ and similarly for y'_i. The expression for computing SROCC is:

$$\rho = \frac{\sum_i (x'_i - \bar{x'})(y'_i - \bar{y'})}{\sqrt{\sum_i (x'_i - \bar{x'})^2 \sum_i (y'_i - \bar{y'})^2}} \tag{1}$$

where, $\bar{x'}$ is the sample mean over $\{x'_1, x'_2, ..., x'_N\}$. Refer to the following link to get a better idea. **Spearman's Rank-Order Correlation**

Write the `spr(x, y)` function in the `srocc_on_np.py` file to compute the SROCC between x and y.

2 Failure Cases - Data analysis

During the research, after developing a model and testing it on different images, we analyze the estimated quality predictions to understand better where the model was performing poorly. We call them failure cases. One way to investigate such failure cases is by computing the error between MOS and estimated quality and selecting the images with the highest error. We then qualitatively analyze the perceptual quality distortions the model cannot capture.

The file `contrique_livefb_test.csv` contains a table with columns that specify **image location**, its corresponding **MOS**, and estimated quality **prediction**. Write the function `worst_perf_images(df, n)` in `failure_cases.py`, which takes in a pandas DataFrame of this CSV file and returns a list of image locations of the top n worst performing images.

3 Parameter Estimate of Distribution

You are given $n = 1000$ samples of a random normal vector $\mathbf{X}$ of size 3. In `parameter_estimation.py`, the variable `x` contains a matrix of size 3×1000 where each column corresponds to an individual sample. You have to estimate the sample mean μ and sample covariance matrix Σ of $\mathbf{X}$ as described below.

$$\mu(x) = \frac{1}{n} \sum_{i=1}^{n} x_i \tag{2}$$

$$\Sigma(x) = \frac{1}{n-1} \sum_{i=1}^{n} (x_i - \mu(x))(x_i - \mu(x))^T \tag{3}$$

where x_i is a column vector of size 3 representing the i^{th} sample.