Notes for Experiment 1

1. Reading input data into python:

The input data is given in *.mat* format. *Use* the codes in the file *expt1_data_io.py* to read them into Python.

2. Addition of noise

The signal to noise ratio (SNR) of a noisy signal is computed as:

$$SNR\ (in\ dB) = 10log_{10}\frac{P_{signal}}{P_{noise}}$$

Hence, given the SNR, the noise power (P_{noise}) in the noisy signal can be calculated as:

$$P_{noise} = P_{signal}10^{\frac{-SNR}{10}}$$

To generate the noisy signal, $x_{noisy}(t)$, at the specified SNR:

$$\boldsymbol{x_{noisy}(t) = x(t) + A * w(t)}$$

Where $x(t)$ and $w(t)$ are the original signal and noise respectively and,

$$A = \sqrt{\frac{P_{noise}}{var(w(t))}}$$

Here, variance of a signal yields its power.

3. Calculating improvement in SNR

Let $x_{noisy}(t)$ be the noisy signal. Let $d(t)$ and $y(t)$ be the desired output and the filtered output respectively. The residual noise $v(t)$ in the filtered output is computed as:

$$v(t) = y(t) - d(t)$$

Hence, SNR after filtering can be computed as:

$$SNR_{new} = 10\log_{10}\frac{var(d(t))}{var(v(t))}$$

The improvement in SNR is then calculated as:

$$\boldsymbol{Improvement = SNR_{new} - SNR_{original}}$$

4. Calculating Signal to Distortion Ratio

When a noisy signal is filtered, it reduces the noise and also introduces slight distortions in the filtered signal because of filter artifacts. The distortion (along with residual noise) in the filtered signal can be obtained as below which in turn is used to compute the Signal to Distortion Ratio (SDR).

$$x_{distort}(t) = x_{original}(t) - x_{filtered}(t)$$

$$\boldsymbol{SDR = 10 \log_{10} \frac{var\left(x_{original}(t)\right)}{var\left(x_{distort}(t)\right)}}$$