Objective Function:

Let the power spectrum we are trying to fit be $P(e^{j\omega})$ and let the estimated parameterized spectrum be $P_\theta(e^{j\omega})$. The objective function

$$J(\theta) = \frac{1}{2\pi}\int_{-\pi}^{\pi} \mathcal{L}\{\log P(e^{j\omega}), \log P_\theta(e^{j\omega})\} d\omega$$

Where, $\mathcal{L}$ measures the dissimilarity between two spectrums.

This problem can be transformed into a stochastic optimization problem if we rewrite $J(\theta)$ as

$$J(\theta) = \underset{\omega \sim \mathrm{U}(-\pi,\pi)}{\mathrm{E}} \mathcal{L}\{\log P(e^{j\omega}), \log P_\theta(e^{j\omega})\}$$

Taking, $\mathcal{L}(t_1, t_2) = (t_1 - t_2)^4$

$$J(\theta) = \underset{\omega \sim \mathrm{U}(-\pi,\pi)}{\mathrm{E}} \{\log P(e^{j\omega}) - \log P_\theta(e^{j\omega})\}^4$$

Parameters of the ARMA system:

The ARMA signal:

$$x[n] = h[n] * v[n]$$

Where, $h[n]$ is the impulse response of the system and $v[n]$ is white Gaussian noise.

z-transform of the impulse response

$$H(z) = \frac{1 + \sum_{k=1}^{n_q} b_k z^{-1}}{1 + \sum_{k=1}^{n_p} a_k z^{-1}} = \frac{\prod_{k=1}^{n_q}(1 - q_k z^{-1})}{\prod_{k=1}^{n_p}(1 - p_k z^{-1})}$$

The estimated log power spectrum,

$$\begin{aligned}\log P(e^{j\omega}) &= \log\left\{\left|\frac{\prod_{k=1}^{n_q}(1 - q_k e^{-j\omega})}{\prod_{k=1}^{n_p}(1 - p_k e^{-j\omega})}\right| \sigma_v^2\right\} \\ &= 2\left[\sum_{k=1}^{n_q}\left|1 - q_k e^{-j\omega}\right| - \sum_{k=1}^{n_p}\left|1 - p_k e^{-j\omega}\right| + \sigma_v\right]\end{aligned}$$

σ_v can be left out if we by performing a normalization by dividing the ACFs by the zeroth lag coefficient ($R_{xx}[0]$) before estimating the power spectrum.

Then the parameter vector can be written as

$$\boldsymbol{\theta} = \begin{bmatrix} real(\boldsymbol{p}) \\ imag(\boldsymbol{p}) \\ real(\boldsymbol{q}) \\ imag(\boldsymbol{q}) \end{bmatrix}$$

Where, $\boldsymbol{p} = \left[p_1, p_2, \dots, p_{n_p}\right]^T$ and $\boldsymbol{q} = \left[q_1, q_2, \dots, q_{n_q}\right]^T$. The optimum value can found,

$$\boldsymbol{\theta}_{opt} = \underset{\theta}{argmin}\, J(\theta) = \underset{\theta}{argmin} \underset{\omega \sim \mathrm{U}(-\pi,\pi)}{\mathrm{E}} \left\{ \log P\left(e^{j\omega}\right) - \log P_{\theta}\left(e^{j\omega}\right) \right\}^4$$

Gradient Descent:

At its simplest form, stochastic Gradient Descent:

$$\boldsymbol{\theta}_{i+1} = \boldsymbol{\theta}_i - \nabla_{\theta} J(\boldsymbol{\theta}_i)$$

However, more complex optimization algorithms can have internal states $\boldsymbol{\phi_i}$ e.g. momentum. In that case,

$$\boldsymbol{\theta}_{i+1} = \boldsymbol{\theta}_i - f\{\nabla_{\theta} J(\boldsymbol{\theta}_i), \boldsymbol{\phi_i}\}$$

Re-parameterization:

In order to make the optimization procedure easier, we can solve for a higher dimensional θ' instead of θ

$$\theta = f(\theta')$$

For this we chose the linear transform equation [Perceptron Equation].