Recursive Least Squares (RLS)Adaptive Filter

- LMS convergence slow
- Step size parameter is to be properly chosen
- LMS minimizes the instantaneous square error $e^2[n]$
  Where $e[n] = x[n] - \mathbf{h}'[n]\,\mathbf{y}[n] = x[n] - \mathbf{y}'[n]\,\mathbf{h}[n]$

The RLS algorithm considers all the available data for determining the filter parameters. The filter should be optimum with respect to all the available data in certain sense.

Minimizes the cost function

$$\varepsilon[n] = \sum_{k=0}^{n} \lambda^{\text{n-k}} \text{e}^2[k]$$

with respect to the filter parameter vector $\mathbf{h}[n] = \begin{bmatrix} h_0[n] \\ h_1[n] \\ h_{M-1}[n] \end{bmatrix}$

where λ is the weighing factor known as the forgetting factor

- Recent data is given more weightage
- For stationary case λ = 1 can be taken
- λ ≅ 0.95 is effective in tracking local nonstationarily

The minimization problem is

$$\text{Minimize } \ \varepsilon[n] = \sum_{k=0}^{n} \lambda^{n-k} \left(x[k] - \mathbf{y}'[k]\ \mathbf{h}[n]\right)^2$$

with respect to $\mathbf{h}[n]$

The minimum is given by

$$\frac{\partial \varepsilon(n)}{\partial \mathbf{h}(n)} = \mathbf{0}$$

$$\Rightarrow 2\left(\sum_{k=0}^{n} \lambda^{n-k} x[k]\,\mathbf{y}[k] - \mathbf{y}[k]\,\mathbf{y}'[k]\,\mathbf{h}[n]\right) = 0$$

$$\Rightarrow \ \mathbf{h}[n] = \left(\sum_{k=0}^{n} \lambda^{n-k} \mathbf{y}[k]\mathbf{y}'[k]\right)^{-1} \sum_{k=0}^{n} \lambda^{n-k} x[k]\mathbf{y}[k]$$

Let us define $\hat{\mathbf{R}}_{YY}[n] = \sum_{k=0}^{n} \lambda^{n-k} \mathbf{y}[k]\mathbf{y}'[k]$

which is an estimator for the autocorrelation matrix $\mathbf{R}_{\mathbf{YY}}$.

Similarly $\hat{\mathbf{r}}_{\mathbf{XY}}[n] = \sum_{k=0}^{n} \lambda^{n-k} x[k]\mathbf{y}[k] =$ estimator for the autocorrelation vector $\mathbf{r}_{XY}[n]$

Hence $\mathbf{h}[n] = \left(\hat{\mathbf{R}}_{\mathbf{XY}}[n]\right)^{-1} \hat{\mathbf{r}}_{\mathbf{XY}}[n]$

Matrix inversion is involved which makes the direct solution difficult. We look forward for a recursive solution.

Recursive representation of $\hat{\mathbf{R}}_{\mathbf{YY}}[n]$

$\hat{\mathbf{R}}_{\mathbf{YY}}[n]$ can be rewritten as follows

$$\hat{\mathbf{R}}_{\mathbf{YY}}[n] = \sum_{k=0}^{n-1} \lambda^{n-k} \mathbf{y}[k]\,\mathbf{y}'[k] + \mathbf{y}[n]\,\mathbf{y}'[n]$$
$$= \lambda \sum_{k=0}^{n-1} \lambda^{n-1-k} \mathbf{y}[k]\,\mathbf{y}'[k] + \mathbf{y}[n]\,\mathbf{y}'[n]$$
$$= \lambda \hat{\mathbf{R}}_{\mathbf{YY}}[n-1] + \mathbf{y}[n]\,\mathbf{y}'[n]$$

This shows that the autocorrelation matrix can be recursively computed from its previous values and the present data vector.

Similarly $\hat{\mathbf{r}}_{\mathbf{XY}}[n] = \lambda \hat{\mathbf{r}}_{\mathbf{XY}}[n-1] + x[n]\mathbf{y}[n]$

$$\mathbf{h}[n] = \left[\hat{\mathbf{R}}_{\mathbf{YY}}[n]\right]^{-1} \hat{\mathbf{r}}_{\mathbf{XY}}[n]$$
$$= \left(\lambda \hat{\mathbf{R}}_{\mathbf{YY}}[n-1] + \mathbf{y}[n]\mathbf{y}[n]'\right)^{-1} \hat{\mathbf{r}}_{\mathbf{XY}}[n]$$

For the matrix inversion above the matrix inversion lemma will be useful.

Matrix Inversion Lemma

If **A, B, C** and **D** are matrices of proper orders, **A** and **C** nonsingular

$$(\mathbf{A} + \mathbf{BCD})^{-1} = \mathbf{A}^{-1} - \mathbf{A}^{-1}\mathbf{B}\,(\mathbf{DA}^{-1}\mathbf{B} + \mathbf{C}^{-1})^{-1}\,\mathbf{DA}^{-1}$$

Taking $\mathbf{A} = \lambda \hat{\mathbf{R}}_{\mathbf{YY}}[n-1],\ \mathbf{B} = \mathbf{y}[n],\ \mathbf{C} = 1$ and $\mathbf{D} = \mathbf{y}'[n]$

we will have

$$\left(\hat{\mathbf{R}}_{yy}[n]^{-1}\right) = \frac{1}{\lambda}\hat{\mathbf{R}}_{\mathbf{YY}}^{-1}[n-1] - \frac{1}{\lambda}\hat{\mathbf{R}}_{\mathbf{YY}}^{-1}[n-1]\mathbf{y}[n]\left(\mathbf{y}'[n]\frac{1}{\lambda}\left[\hat{\mathbf{R}}_{\mathbf{YY}}^{-1}[n-1]\right]\mathbf{y}[n] + 1\right)^{-1}\mathbf{y}'[n]\hat{\mathbf{R}}_{\mathbf{YY}}^{-1}[n-1]$$
$$= \frac{1}{\lambda}\left(\hat{\mathbf{R}}_{\mathbf{YY}}^{-1}[n-1] - \frac{\hat{\mathbf{R}}_{\mathbf{YY}}^{-1}[n-1]\mathbf{y}[n]\mathbf{y}[n']\hat{\mathbf{R}}_{\mathbf{YY}}^{-1}[n-1]}{\lambda + \mathbf{y}'[n]\hat{\mathbf{R}}_{\mathbf{YY}}^{-1}[n-1]\mathbf{y}[n]}\right)$$

Rename $\mathbf{P}[n] = \hat{\mathbf{R}}_{\mathbf{YY}}^{-1}[n]$. Then

$$\mathbf{P}[n] = \frac{1}{\lambda}\left(\mathbf{P}[n-1] - \mathbf{k}[n]\mathbf{y}'[n]\mathbf{P}[n-1]\right)$$

where $\mathbf{k}[n]$ is called the 'gain vector' and given by

$$\mathbf{k}[n] = \frac{\mathbf{P}[n-1]\mathbf{y}[n]}{\lambda + \mathbf{y}'[n]\mathbf{P}[n-1]\mathbf{y}[n]}$$

$\mathbf{k}[n]$ important to interprete adaptation is also related to the current data vector $\mathbf{y}[n]$ by

$$\mathbf{k}[n] = \mathbf{P}[n]\mathbf{y}[n]$$

To establish the above relation consider

$$\mathbf{P}[n] = \frac{1}{\lambda}\left(\mathbf{P}[n-1] - \mathbf{k}[n]\mathbf{y}'[n]\mathbf{P}[n-1]\right)$$

Multiplying by λ and post-multiplying by $\mathbf{y}[n]$ and simplifying we get

$$
\begin{aligned}
\lambda \mathbf{P}[n]\mathbf{y}[n] &= \left(\mathbf{P}[n-1] - \mathbf{k}[n]\mathbf{y}'[n]\mathbf{P}[n-1]\right)\mathbf{y}[n] \\
&= \mathbf{P}[n-1]\mathbf{y}[n] - \mathbf{k}[n]\mathbf{y}'[n]\mathbf{P}[n-1]\mathbf{y}[n] \\
&= \lambda \mathbf{k}[n]
\end{aligned}
$$

Therefore

$$
\begin{aligned}
\mathbf{h}[n] &= \left(\hat{\mathbf{R}}_{\mathbf{YY}}[n]\right)^{-1} \hat{\mathbf{r}}_{\mathbf{XY}}[n] \\
&= \mathbf{P}[n]\left(\lambda \hat{\mathbf{r}}_{\mathbf{XY}}[n-1] + x[n]\mathbf{y}[n]\right) \\
&= \lambda \mathbf{P}[n]\hat{\mathbf{r}}_{\mathbf{XY}}[n-1] + x[n]\mathbf{P}[n]\mathbf{y}[n] \\
&= \lambda \tfrac{1}{\lambda}\left[\mathbf{P}[n-1] - \mathbf{k}[n]\mathbf{y}'[n]\mathbf{P}[n-1]\right]\hat{\mathbf{r}}_{\mathbf{XY}}[n-1] + x[n]\mathbf{P}[n]\mathbf{y}[n] \\
&= \mathbf{h}[n-1] - \mathbf{k}[n]\,\mathbf{y}'[n]\,\mathbf{h}[n-1] + x[n]\,\mathbf{k}[n] \\
&= \mathbf{h}[n-1] + \mathbf{k}[n]\left(x[n] - \mathbf{y}'[n]\mathbf{h}[n-1]\right)
\end{aligned}
$$

<u>RLS algorithm</u>

Initialization

At n = 0

$\mathbf{P}[0] = \delta\, \mathbf{I}_{\mathbf{MxM}}$, $\mathbf{y}[0] = \mathbf{0}$, $\mathbf{h}[0] = \mathbf{0}$,

Choose λ

Operation :

For 1 to n = Final do

1. Get $x[n], \mathbf{y}[n]$
2. Get $e[n] = x[n] - \mathbf{h}'[n-1]\mathbf{y}[n]$
3. Calculate gain vector $\mathbf{k}[n] = \dfrac{\mathbf{P}[n-1]\mathbf{y}[n]}{\lambda + \mathbf{y}'[n]\mathbf{P}[n-1]\mathbf{y}[n]}$
4. Update the filter parameters
   $\mathbf{h}[n] = \mathbf{h}[n-1] + \mathbf{k}[n]e[n]$
5. Update the $\mathbf{P}$ matrix
   $\mathbf{P}[n] = \dfrac{1}{\lambda}\left(\mathbf{P}[n-1] - \mathbf{k}[n]\mathbf{y}'[n]\mathbf{P}[n-1]\right)$

end do

Disscusion

(1) Relation with Wiener filter:

We have the optimality condition analysis for the RLS filters

$$\hat{\mathbf{R}}_{\mathbf{YY}}[n]\mathbf{h}[n] = \hat{\mathbf{r}}_{\mathbf{xy}}[n]$$

where $\hat{\mathbf{R}}_{\mathbf{yy}}[n] = \sum_{k=0}^{n} \lambda^{n-k}\mathbf{y}[k]\mathbf{y}'[k]$

Dividing by $n+1$

$$\frac{\hat{\mathbf{R}}_{\mathbf{YY}}[n]}{n+1} = \frac{\sum_{k=0}^{n} \lambda^{n-k}\mathbf{y}[k]\mathbf{y}''[k]}{n+1}$$

if we consider the elements of $\frac{\hat{\mathbf{R}}_{\mathbf{YY}}[n]}{n+1}$, we see that each is a estimator for the autocorreletion of specific lag.

$$\lim_{n\to\infty} \frac{\hat{\mathbf{R}}_{\mathbf{YY}}[n]}{n+1} = \mathbf{R}_{\mathbf{YY}}[n]$$

in the mean square sense. Weighted window sample autocorrelation is a consistent estimator.
Similarly

$$\lim_{n\to\infty} \frac{\hat{\mathbf{r}}_{\mathbf{yy}}[n]}{n+1} = \mathbf{r}_{\mathbf{yy}}[n]$$

Hence as $n \to \infty$, optimality condition can be written as

$$\mathbf{R}_{\mathbf{YY}}[n]\mathbf{h}[n] = \mathbf{r}_{\mathbf{xy}}[n] \quad .$$

(2) Dependence condition on the initial values

Consider the recursive relation

$$\hat{\mathbf{R}}_{YY}[n] = \lambda\hat{\mathbf{R}}_{YY}[n-1] + \mathbf{y}[n]\mathbf{y}'[n]$$

Corresponding to

$$\hat{\mathbf{R}}_{YY}^{-1}[-1] = \delta\mathbf{I}$$

we have $\hat{\mathbf{R}}_{YY}[-1] = \frac{\mathbf{I}}{\delta}$

With this inititial condition the matrix difference equation has the solution

$$\widetilde{\mathbf{R}}[n] = \lambda^{n+1}\hat{\mathbf{R}}_{YY}[-1] + \sum_{k=0}^{n} \lambda^{n-k}\mathbf{y}[k]\mathbf{y}'[k]$$
$$= \lambda^{n+1}\hat{\mathbf{R}}_{YY}[-1] + \hat{\mathbf{R}}_{YY}[n]$$
$$= \lambda^{n+1}\frac{\mathbf{I}}{\delta} + \hat{\mathbf{R}}_{YY}[n]$$

Hence the optimality condition is modified as

$$(\lambda^{n+1}\frac{\mathbf{I}}{\delta} + \hat{\mathbf{R}}_{YY}[n])\widetilde{\mathbf{h}}[n] = \hat{\mathbf{r}}_{XY}[n]$$

where $\widetilde{\mathbf{h}}[n]$ is the modified solution due to assumed initial value of the P-matrix.

$$\frac{\lambda^{n+1}\hat{\mathbf{R}}_{YY}^{-1}[n]\widetilde{\mathbf{h}}[n]}{\delta} + \widetilde{\mathbf{h}}[n] = \mathbf{h}[n]$$

If we take λ as less than 1,then the bias term in the left-hand side of the above equation will be eventually die down and we will get

$$\widetilde{\mathbf{h}}[n] = \mathbf{h}[n]$$

3. <u>Behaviour in stationary condition</u>

If the data is stationary, the algorithm is basically solving a set of linear equation. So the algorithm will converge at the best M iterations, where M is the filter depth.

4. <u>Tracking non-staionarity</u>

If λ is small $\lambda^{n-i} \cong 0$ for $i << n$

$\Rightarrow$ the filter is based on most recent values. This is also qualitatively explains that the filter can track non stationary in data.

5. Computational Complexity:

Several matrix multiplication results in $\cong 7M^2$ arithmetic operations, which is quite large, if the respective filter length high. So we have to go for the best implementation of the RLS algorithm.