Derivation MPMR-Simple Learning Algorithm

We start out from equation (17) of the MPMC paper, which simplifies for MPMR to only one term (because the symmetric $\pm\epsilon$-shift of the output coordinate implies $\Sigma_u = \Sigma_v$):
min $||\Sigma_u^{1/2}(a_0 + Ft)||_2^2$ with respect to $t \in R^N$

Let $B = \Sigma_u^{1/2}$, since the covariance matrix $\Sigma_u = Cov(Y, Z_1, ..., Z_N) \in R^{(N+1)\times(N+1)}$ is symmetric and nonnegative definite, it has a square root that is a symmetric matrix. Thus, we have the following equations (write u_{ij} for entries of Σ_u):
(1) $u_{kj} = \sum_{i=1}^{N+1} b_{ki}b_{ij} = \sum_{k=1}^{N+1} b_{ik}b_{ij} \quad$ (symmetry: $b_{ki} = b_{ik}$)

Back to the minimization problem with $a_0 \in R^{N+1}, F \in R^{(N+1)\times N}, t \in R^N$:

$$a_0 = \begin{pmatrix} 1/2\epsilon \\ 0 \\ \vdots \\ 0 \end{pmatrix} \quad F = \begin{pmatrix} 0 & 0 & \dots & 0 \\ 1 & 0 & \dots & 0 \\ 0 & 1 & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & 1 \end{pmatrix} \quad t = \begin{pmatrix} t_1 \\ t_2 \\ \vdots \\ t_N \end{pmatrix} \quad Ft = \begin{pmatrix} 0 \\ t_1 \\ t_2 \\ \vdots \\ t_N \end{pmatrix}$$

$$||B(a_0 + Ft)||_2^2 = \left\| \begin{pmatrix} b_{11} & b_{12} & \dots & b_{1,N+1} \\ b_{21} & b_{22} & \dots & b_{2,N+1} \\ \dots & \dots & \dots & \dots \\ b_{N+1,1} & b_{N+1,2} & \dots & b_{N+1,N+1} \end{pmatrix} \begin{pmatrix} 1/2\epsilon \\ t_1 \\ t_2 \\ \vdots \\ t_N \end{pmatrix} \right\|_2^2$$

$$= \left\| \begin{pmatrix} b_{11}/2\epsilon + \sum_{j=1}^N b_{1,j+1}t_j \\ b_{21}/2\epsilon + \sum_{j=1}^N b_{2,j+1}t_j \\ \vdots \\ b_{N+1,1}/2\epsilon + \sum_{j=1}^N b_{N+1,j+1}t_j \end{pmatrix} \right\|_2^2$$

$= \sum_{i=1}^{N+1}(b_{i1}/2\epsilon + \sum_{j=1}^N b_{i,j+1}t_j)^2 =: g(t)$

Setting the derivatives of $g(t)$ to zero yields:
$\frac{d}{dt_k}g(t) = \sum_{i=1}^{N+1} 2(b_{i1}/2\epsilon + \sum_{j=1}^N b_{i,j+1}t_j)b_{i,k+1} = 0$
$\Leftrightarrow 2\sum_{i=1}^{N+1}\sum_{j=1}^N b_{i,k+1}b_{i,j+1}t_j = -\frac{1}{\epsilon}\sum_{i=1}^{N+1} b_{i1}b_{i,k+1}$

This defines a linear system – call it $St = r$ – that has N equations ($\frac{d}{dt_k}g(t) = 0$ for $k = 1, ..., N$) and N variables $(t_1, ..., t_N)$. From (1) we get:
$S_{kj} = 2\sum_{i=1}^{N+1} b_{i,k+1}b_{i,j+1} = 2u_{k+1,j+1} \quad k = 1, ..., N \; j = 1, ..., N$
$r_k = -\frac{1}{\epsilon}\sum_{i=1}^{N+1} b_{i1}b_{i,k+1} = -\frac{1}{\epsilon}u_{1,k+1} \quad k = 1, ..., N$

So we solve the following system:

$$2\begin{pmatrix} u_{2,2} & u_{2,3} & \dots & u_{2,N+1} \\ u_{3,2} & u_{3,3} & \dots & u_{3,N+1} \\ \dots & \dots & \dots & \dots \\ u_{N+1,2} & u_{N+1,3} & \dots & u_{3,N+1} \end{pmatrix} \begin{pmatrix} t_1 \\ t_2 \\ \vdots \\ t_N \end{pmatrix} = -\frac{1}{\epsilon} \begin{pmatrix} u_{12} \\ u_{13} \\ \vdots \\ u_{1,N+1} \end{pmatrix}$$

After computing t we obtain a as a $(N+1)$-dimensional vector defining the regression hyperplane: $a_1 = 1/2\epsilon$, $a_{i+1} = t_i \ \ i = 1, ..., N$. The linear combination of inputs and output looks like this:

$a_1 y + a_2 z_1 + a_3 z_2 + ... a_{N+1} z_N = 0 \Leftrightarrow y = -\sum_{i=1}^{N} \frac{a_{i+1}}{a_1} z_i = \sum_{i=1}^{N} (-2\epsilon t_i) z_i$

Thus the coefficients of the inputs z_i are $\beta_i = -2\epsilon t_i$. Substituting this linear relationship ($t_i = -\frac{1}{2\epsilon}\beta_i$) into our linear system $St = r$ yields:

$$2\begin{pmatrix} u_{2,2} & u_{2,3} & \dots & u_{2,N+1} \\ u_{3,2} & u_{3,3} & \dots & u_{3,N+1} \\ \dots & \dots & \dots & \dots \\ u_{N+1,2} & u_{N+1,3} & \dots & u_{3,N+1} \end{pmatrix} - \frac{1}{2\epsilon} \begin{pmatrix} \beta_1 \\ \beta_2 \\ \vdots \\ \beta_N \end{pmatrix} = -\frac{1}{\epsilon} \begin{pmatrix} u_{12} \\ u_{13} \\ \vdots \\ u_{1,N+1} \end{pmatrix}$$

which simplifies to:

$$\begin{pmatrix} u_{2,2} & u_{2,3} & \dots & u_{2,N+1} \\ u_{3,2} & u_{3,3} & \dots & u_{3,N+1} \\ \dots & \dots & \dots & \dots \\ u_{N+1,2} & u_{N+1,3} & \dots & u_{3,N+1} \end{pmatrix} \begin{pmatrix} \beta_1 \\ \beta_2 \\ \vdots \\ \beta_N \end{pmatrix} = \begin{pmatrix} u_{12} \\ u_{13} \\ \vdots \\ u_{1,N+1} \end{pmatrix}$$

and which looks like this in covariance notation:

$$\begin{pmatrix} \sigma_{z_1 z_1} & \sigma_{z_1 z_2} & \dots & \sigma_{z_1 z_N} \\ \sigma_{z_2 z_1} & \sigma_{z_2 z_2} & \dots & \sigma_{z_2 z_N} \\ \dots & \dots & \dots & \dots \\ \sigma_{z_N z_1} & \sigma_{z_N z_2} & \dots & \sigma_{z_N z_N} \end{pmatrix} \begin{pmatrix} \beta_1 \\ \beta_2 \\ \vdots \\ \beta_N \end{pmatrix} = \begin{pmatrix} \sigma_{y z_1} \\ \sigma_{y z_2} \\ \vdots \\ \sigma_{y z_N} \end{pmatrix}$$