深蓝学院 VIO 第三次课程作业

温焕宇

2019.6.27

1 题一

1.1 请绘制样例代码中 LM 阻尼因子 μ 随着迭代变化的曲线图

答：使用工具：python，课程给代码’CurveFitting-LM‘

步骤：

1. 修改并编译代码

在 problem.cc 文件里 while 循环下加入以下用于存储迭代次数和 λ：

```
ofstream save_lamda;
save_lamda.open("../../data/Lamda.txt",ios::app);
save_lamda << iter << "\t" << currentLambda_ << endl;
save_lamda.close();
```

编译并执行

```
mkdir build
cd build/
cmake ..
make
cd ..
./build/app/testCurveFitting
```

得到结果如下：

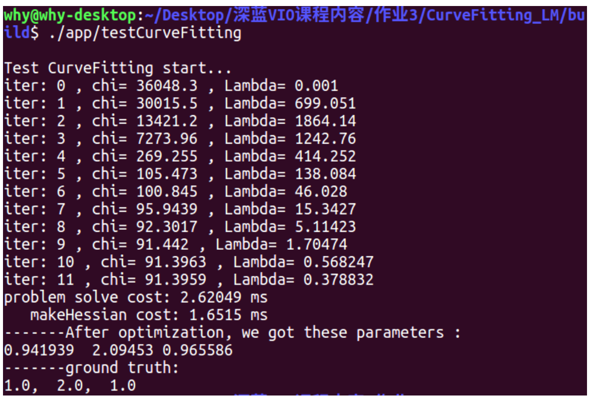

图 1: 没修改后的结果图

2. 写 python 画散点图：

```
import matplotlib.pyplot as plt
filename = 'Lamda.txt'
X, Y = [],[]
for line in open(filename,'r'):
    value = [float(s) for s in line.split()]
    X.append(value[0])
    Y.append(value[1])

plt.plot(X,Y,'ro')
plt.title('Lamda trends')
plt.xlabel('Iter')
plt.ylabel('Lamda')
plt.show()
```

结果图如下：

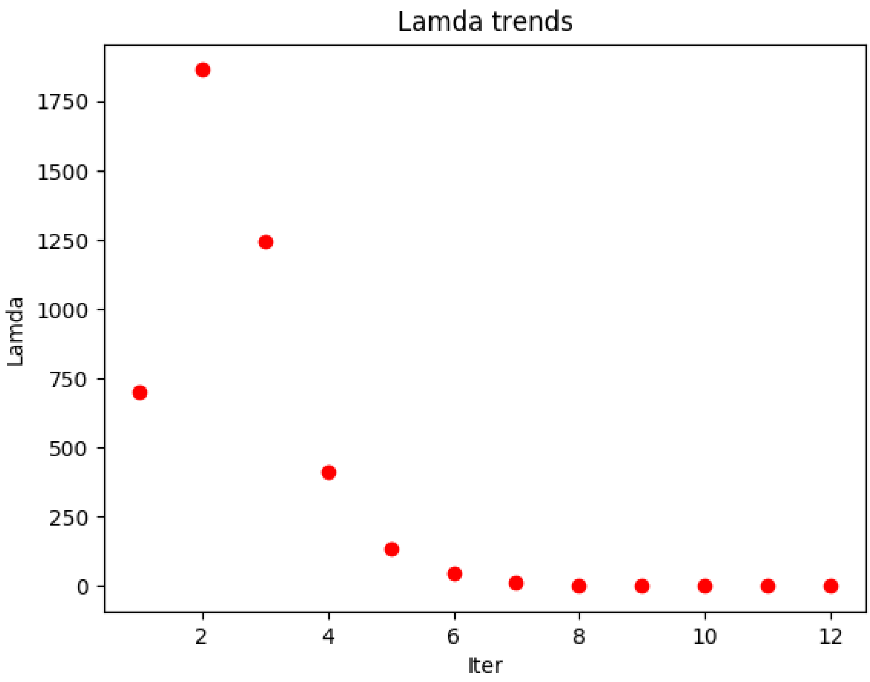

图 2: μ 随迭代变化的曲线图

1.2 将曲线函数改为 $y = ax^2 + bx + c$，请修改样例代码中残差计算，雅可比计算等函数，完成曲线估计

将 $y = \exp(ax^2 + bx + c)$ 改成 $y = ax^2 + bx + c$
残差为：

$$residual = \breve{a}x^2 + \breve{b}x + \breve{c} - y \tag{1}$$

其中ˇ上标表示预测值

对 a b c 求偏导得雅可比矩阵为：

$$[x^2 \quad x \quad 1] \tag{2}$$

修改代码如下：

```
residual_(0) = abc(0)*x_*x_ + abc(1)*x_ + abc(2) - y_; // 构建残差 = 预测值-测量值

jaco_abc << x_ * x_, x_, 1; // y = a*x*x + b*x + c 的雅可比矩阵
```

【注：由于对于此方程，原始设置的噪声过大，不好收敛，故采用增大观测数或者减小噪声，考虑实际情况，选择增大观测次数 N=1500】

得到结果如下：

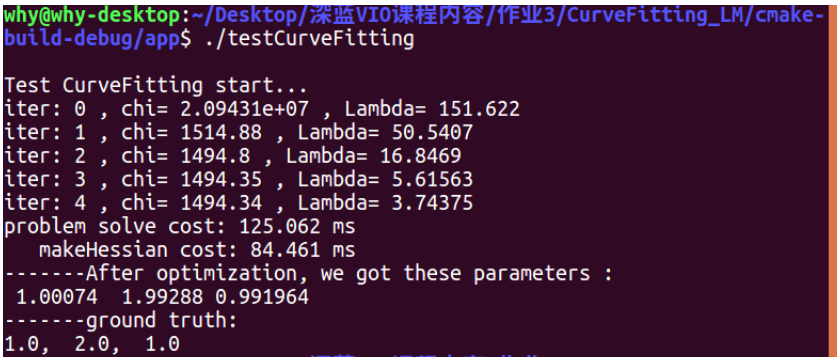

图 3: 修改后输出结果

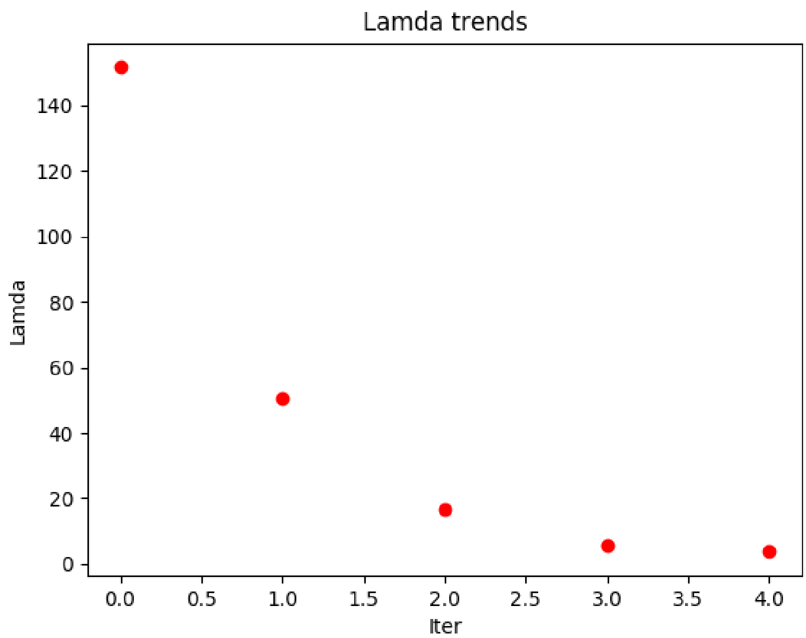

图 4: 修改后 λ 曲线图

1.3 实现其它阻尼因子更新策略

参考‘The Levenberg-Marquardt algorithm fornonlinear least squares curve-fitting problems’第一种阻尼因子更新策略：

1. $\lambda_0 = \lambda_o$; λ_o is user-specified [8].
   use eq'n (13) for $\boldsymbol{h}_{\mathsf{lm}}$ and eq'n (16) for ρ
   if $\rho_i(\boldsymbol{h}) > \epsilon_4$: $\boldsymbol{p} \leftarrow \boldsymbol{p} + \boldsymbol{h}$; $\lambda_{i+1} = \max[\lambda_i / L_{\downarrow}, 10^{-7}]$;
   otherwise: $\lambda_{i+1} = \min[\lambda_i L_{\uparrow}, 10^7]$;

其中 $L_{\downarrow}$ 取 9，$L_{\uparrow}$ 取 11 时效果最好。

修改代码如下：

```cpp
double maxDiagonal = 0;
ulong size = Hessian_.cols();
assert(Hessian_.rows() == Hessian_.cols() && "Hessian is not square");
for (ulong i = 0; i < size; ++i) {
    maxDiagonal = std::max(fabs(Hessian_(i, i)), maxDiagonal); //Hessian矩阵对角元素最大值
}

scale = delta_x_.transpose() * (currentLambda_ * maxDiagonal * delta_x_ + b_);
scale += 1e-3;
double rho = (currentChi_ - tempChi) / scale;
if(rho > 0 && isfinite(tempChi))
{
    double scaleFactor = (std::max)(currentLambda_/9, 10e-7);
    currentLambda_ = scaleFactor;
    currentChi_ = tempChi;
    return true;
} else{
    double scaleFactor = (std::min)(currentLambda_/11, 10e+7);
    currentLambda_ = scaleFactor;
    return false;
}
```

相比于第三种阻尼因子更新策略，第一种只需要迭代两次即可收敛，效果较好。结果如图 5：

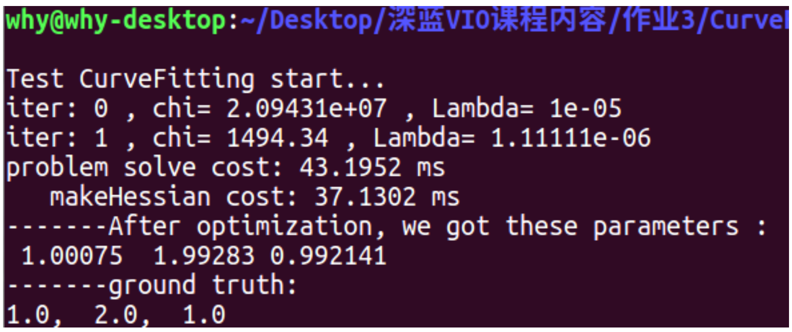

图 5: 采用第一种阻尼因子更新策略结果

2 题二

2.1 根据课程知识，完成 F，G 推导

答：根据课程 PPT 内容推导如下：

(1) $\mathbf{f}_{15} = \frac{\partial \boldsymbol{\alpha}_{b_i b_{k+1}}}{\partial \delta \mathbf{b}_k^g}$

$\mathbf{a} = \frac{1}{2}\Big(\mathbf{q}_{b_i b_k}(a^{b_k} - b_k^a) + \mathbf{q}_{b_i b_k} \otimes \begin{bmatrix} 1 \\ \frac{1}{2}\boldsymbol{\omega}\delta t \end{bmatrix} (\mathbf{a}^{b_{k+1}} - \mathbf{b}_k^a)\Big)$

$\boldsymbol{\omega} = \frac{1}{2}\Big((\boldsymbol{\omega}^{b_k} + \mathbf{n}_k^g - b_k^g) + (\boldsymbol{\omega}^{b_{k+1}} + \mathbf{n}_{k+1}^g - \mathbf{b}_k^g)\Big) = \frac{1}{2}\Big((\boldsymbol{\omega}^{b_k} + \mathbf{n}_k^g) + (\boldsymbol{\omega}^{b_{k+1}} + \mathbf{n}_{k+1}^g)\Big) - \mathbf{b}_k^g$

$\boldsymbol{\alpha}_{b_i b_{k+1}} = \boldsymbol{\alpha}_{b_i b_k} + \boldsymbol{\beta}_{b_i b_k} + \frac{1}{2}\mathbf{a}\delta t^2$

$\mathbf{b}_k^g$ 与 $\boldsymbol{\omega}$ 相关，误差传递过程，对 $\boldsymbol{\omega}$ 右乘微小扰动 $\delta \mathbf{b}_k^g \delta t$ 求以下偏导：

$$
\begin{aligned}
\mathbf{f}_{15} &= \frac{\partial \boldsymbol{\alpha}_{b_i b_{k+1}}}{\partial \delta \mathbf{b}_k^g} && (3)\\
&= \frac{\partial \frac{1}{2} \cdot \frac{1}{2}\boldsymbol{\alpha}\delta t^2}{\partial \delta \mathbf{b}_k^g} && (4)\\
&= \frac{\partial \frac{1}{2} \cdot \frac{1}{2}\mathbf{q}_{b_i b_k} \otimes \begin{bmatrix} 1 \\ \frac{1}{2}\boldsymbol{\omega}\delta t \end{bmatrix} (a^{b_{k+1}} - b_k^a)\delta t^2}{\partial \delta \mathbf{b}_k^g} && (5)\\
&= \frac{1}{4}\frac{\partial \mathbf{q}_{b_i b_k} \otimes \begin{bmatrix} 1 \\ \frac{1}{2}\boldsymbol{\omega}\delta t \end{bmatrix} \otimes \begin{bmatrix} 1 \\ -\frac{1}{2}\delta \mathbf{b}_k^g \delta t \end{bmatrix} (\mathbf{a}^{b_{k+1}} - \mathbf{b}_k^a)\delta t^2}{\partial \delta \mathbf{b}_k^g} && (6)\\
&= \frac{1}{4}\frac{\partial \mathbf{R}_{b_i b_{k+1}} \exp([-\frac{1}{2}\delta \mathbf{b}_k^g \delta t]_\times)(\mathbf{a}^{b_{k+1}} - \mathbf{b}_k^a)\delta t^2}{\partial \delta \mathbf{b}_k^g} && (7)\\
&= \frac{1}{4}\frac{\partial \mathbf{R}_{b_i b_{k+1}}(\mathbf{I} + [-\frac{1}{2}\delta \mathbf{b}_k^g \delta t]_\times)(\mathbf{a}^{b_{k+1}} - \mathbf{b}_k^a)\delta t^2}{\partial \delta \mathbf{b}_k^g} && (8)\\
&= -\frac{1}{4}\frac{\partial \mathbf{R}_{b_i b_{k+1}}[(\mathbf{a}^{b_{k+1}} - \mathbf{b}_k^a)]_\times \delta t^2(-\delta \mathbf{b}_k^g \delta t)}{\partial \delta \mathbf{b}_k^g} && (9)\\
&= -\frac{1}{4}\Big(\mathbf{R}_{b_i b_{k+1}}[(\mathbf{a}^{b_{k+1}} - \mathbf{b}_k^a)]_\times \delta t^2\Big)(-\delta t) && (10)\\
& && (11)
\end{aligned}
$$

(2) $\mathbf{g}_{12} = \frac{\partial \boldsymbol{\alpha}_{b_i b_{k+1}}}{\partial \delta \mathbf{n}_k^g}$

$\mathbf{a} = \frac{1}{2}\Big(\mathbf{q}_{b_i b_k}(a^{b_k} - b_k^a) + \mathbf{q}_{b_i b_k} \otimes \begin{bmatrix} 1 \\ \frac{1}{2}\boldsymbol{\omega}\delta t \end{bmatrix} (\mathbf{a}^{b_{k+1}} - \mathbf{b}_k^a)\Big)$

$\boldsymbol{\omega} = \frac{1}{2}\Big((\boldsymbol{\omega}^{b_k} + \mathbf{n}_k^g - b_k^g) + (\boldsymbol{\omega}^{b_{k+1}} + \mathbf{n}_{k+1}^g - \mathbf{b}_k^g)\Big) = \frac{1}{2}\Big((\boldsymbol{\omega}^{b_k} + \mathbf{n}_k^g) + (\boldsymbol{\omega}^{b_{k+1}} + \mathbf{n}_{k+1}^g)\Big) - \mathbf{b}_k^g$

$\boldsymbol{\alpha}_{b_i b_{k+1}} = \boldsymbol{\alpha}_{b_i b_k} + \boldsymbol{\beta}_{b_i b_k} + \frac{1}{2}\mathbf{a}\delta t^2$

同理：对 $\boldsymbol{\omega}$ 右乘微小扰动 $\frac{1}{2}\delta\mathbf{n}_k^g\delta t$ 得：

$$\begin{aligned}
\mathbf{g}_{12} &= \frac{\partial \boldsymbol{\alpha}_{b_i b_{k+1}}}{\partial \delta \mathbf{n}_k^g} && (12)\\
&= \frac{\partial \frac{1}{2}\boldsymbol{\alpha}\delta t^2}{\partial \delta \mathbf{n}_k^g} && (13)\\
&= \frac{\partial \frac{1}{2}\cdot\frac{1}{2}\mathbf{q}_{b_i b_k} \otimes \begin{bmatrix} 1 \\ \frac{1}{2}\boldsymbol{\omega}\delta t \end{bmatrix} (a^{b_{k+1}} - b_k^a)\delta t^2}{\partial \delta \mathbf{n}_k^g} && (14)\\
&= \frac{1}{4}\frac{\partial \mathbf{q}_{b_i b_k} \otimes \begin{bmatrix} 1 \\ \frac{1}{2}\boldsymbol{\omega}\delta t \end{bmatrix} \otimes \begin{bmatrix} 1 \\ \frac{1}{2}\cdot\frac{1}{2}\delta\mathbf{n}_k^g\delta t \end{bmatrix} (\mathbf{a}^{b_{k+1}} - \mathbf{b}_k^a)\delta t^2}{\partial \delta \mathbf{n}_k^g} && (15)\\
&= \frac{1}{4}\frac{\partial \mathbf{R}_{b_i b_{k+1}} \exp([\frac{1}{2}\delta\mathbf{n}_k^g\delta t]_\times)(\mathbf{a}^{b_{k+1}} - \mathbf{b}_k^a)\delta t^2}{\partial \delta \mathbf{n}_k^g} && (16)\\
&= \frac{1}{4}\frac{\partial \mathbf{R}_{b_i b_{k+1}}(\mathbf{I} + [\frac{1}{2}\delta\mathbf{n}_k^g\delta t]_\times)(\mathbf{a}^{b_{k+1}} - \mathbf{b}_k^a)\delta t^2}{\partial \delta \mathbf{n}_k^g} && (17)\\
&= -\frac{1}{4}\frac{\partial \mathbf{R}_{b_i b_{k+1}}[\mathbf{a}^{b_{k+1}} - \mathbf{b}_k^a)]_\times \delta t^2(\frac{1}{2}\delta\mathbf{n}_k^g\delta t)}{\partial \delta \mathbf{n}_k^g} && (18)\\
&= -\frac{1}{4}(\mathbf{R}_{b_i b_{k+1}}[(\mathbf{a}^{b_{k+1}} - \mathbf{b}_k^a)]_\times \delta t^2)(\frac{1}{2}\delta t) && (19)\\
& && (20)
\end{aligned}$$

3 题三

3.1 证明式（9）

$$\triangle \mathbf{x}_{lm} = -\sum_{j=1}^{n} \frac{\mathbf{v}_j^T \mathbf{F}'^T}{\lambda_j + \mu} \mathbf{v}_j \tag{21}$$

根据 GN-LM 论文推导：

$(\mathbf{J}^T\mathbf{J} + \mu\mathbf{I}) \triangle \mathbf{x}_{lm} = -\mathbf{J}^T\mathbf{f} = -\mathbf{F}'^T$
由 SVD 分解

$$\mathbf{A} = \mathbf{J}^T\mathbf{J} = \mathbf{U}\mathbf{\Lambda}\mathbf{V}^T \tag{22}$$

其中 $\mathbf{U} \in \mathbf{R}^{m\times m}$ 和 $\mathbf{V} \in \mathbf{R}^{n\times n}$。
$\{\mathbf{u_j}\}_{j=1}^m$ 和 $\{\mathbf{v_j}\}_{j=1}^n$ 分别表示的 $\mathbf{U}, \mathbf{V}$ 的列。
因为矩阵是正交的，矢量形成两个正交基：

$$\mathbf{n}_i^T \mathbf{u}_j = \mathbf{v}_i^T \mathbf{v}_i = 1, i = j \tag{23}$$

由（22）（23）得：

$$\mathbf{A} = \sum_{j=1}^{p} (\lambda_j + \mu)\mathbf{u}_j \mathbf{v}_j, \text{p 为矩阵}\mathbf{A}\text{的秩} \tag{24}$$

$$-\mathbf{F}'^T = -\mathbf{J}^T\mathbf{f} = \sum_{j=1}^{m} \beta_j \mathbf{u}_j \tag{25}$$

$$\triangle \mathbf{x}_{lm} = \sum_{i=1}^{n} \eta_i \mathbf{v}_i \tag{26}$$

联立（24-26）得：

$$\mathbf{r} = -\mathbf{F}'^T - \mathbf{A} \triangle \mathbf{x}_{lm} \tag{27}$$

$$= \sum_{j=1}^{p} (\beta_j - (\lambda_j + \mu)\eta_j)\mathbf{u}_j + \sum_{j=p+1}^{m} \beta_j \mathbf{u}_j \tag{28}$$

对上式取模得最小值时满足以下：

$$\beta_j - (\lambda_j + \mu)\eta_j = 0 \tag{29}$$

$$\frac{\beta_j}{\lambda_j + \mu} = \eta_j, j = 1, ..., p. \tag{30}$$

最小二乘结果表示为：

$$\triangle \mathbf{x}_{lm} = \sum_{j=1}^{p} (\frac{\beta_j}{\lambda_j + \mu}) \mathbf{v}_j + \sum_{j=p+1}^{n} \eta_j \mathbf{v}_j \tag{31}$$

$$\| \triangle \mathbf{x}_{lm} \| = \sum_{j=1}^{p} (\frac{\beta_j}{\lambda_j + \mu})^2 + \sum_{j=p+1}^{n} \eta_j^2 \tag{32}$$

当 $\triangle \mathbf{x}_{lm}$ 取最小值时,$\eta_{p+1} = ... = \eta_n = 0$, 由式 (25)(32), 得下式:

$$\triangle \mathbf{x}_{lm_{min}} = \sum_{j=1}^{p} (\frac{\beta_j}{\lambda_j + \mu}) \mathbf{v}_j = - \sum_{j=1}^{p} \frac{\mathbf{u}_j^T \mathbf{F'}^T}{\lambda_j + \mu} \mathbf{v}_j \tag{33}$$

根据矩阵运算推导:

i. 由于 $\mathbf{V}$ 是正交阵，具有性质：$\mathbf{V}^T = \mathbf{V}^{-1} \Rightarrow (\mathbf{V}(\mathbf{\Lambda})\mathbf{V}^T)^{-1} = \mathbf{V}(\mathbf{\Lambda})^{-1}\mathbf{V}^T$
ii. 由课件推导可知:
$\mathbf{J}^T\mathbf{J}$ 的特征值为 λ_j 和对应的特征向量为 $\mathbf{v}_j$
对 $\mathbf{J}^T\mathbf{J}$ 特征分解后得:
$\mathbf{J}^T\mathbf{J} = \mathbf{V}(\mathbf{\Lambda} + \mu\mathbf{I})\mathbf{V}^T$ 代入下式:

$$(\mathbf{J}^T\mathbf{J} + \mu\mathbf{I}) \triangle \mathbf{x}_{lm} = -\mathbf{J}^T\mathbf{f} \tag{34}$$

$$\triangle \mathbf{x}_{lm} = (\mathbf{J}^T\mathbf{J} + \mu\mathbf{I})^{-1}(-\mathbf{J}^T\mathbf{f}) \tag{35}$$

$$= (\mathbf{J}^T\mathbf{J} + \mu\mathbf{I})^{-1}(-\mathbf{F'}^T) \tag{36}$$

$$= (\mathbf{V}(\mathbf{\Lambda} + \mu\mathbf{I})\mathbf{V}^T)^{-1}(-\mathbf{F'}^T) \tag{37}$$

$$= \begin{bmatrix} \mathbf{v}_1 & \mathbf{v}_2 & ... & \mathbf{v}_n \end{bmatrix} \begin{bmatrix} \frac{1}{\lambda_1+\mu} & 0 & ... & 0 \\ 0 & \frac{1}{\lambda_2+\mu} & ... & 0 \\ . & . & . & . \\ . & . & . & . \\ . & . & . & . \\ 0 & 0 & ... & \frac{1}{\lambda_n+\mu} \end{bmatrix} \begin{bmatrix} \mathbf{v_1}^T \\ \mathbf{v_2}^T \\ . \\ . \\ . \\ \mathbf{v_n}^T \end{bmatrix} (-\mathbf{F'}^T) \tag{38}$$

$$= \begin{bmatrix} \frac{\mathbf{v}_1}{\lambda_1+\mu} & \frac{\mathbf{v}_2}{\lambda_2+\mu} & ... & \frac{\mathbf{v}_n}{\lambda_n+\mu} \end{bmatrix} \begin{bmatrix} -\mathbf{v_1}^T\mathbf{F'}^T \\ -\mathbf{v_2}^T\mathbf{F'}^T \\ . \\ . \\ . \\ -\mathbf{v_n}^T\mathbf{F'}^T \end{bmatrix} \tag{39}$$

$$= -\sum_{j=1}^{n} \frac{\mathbf{v}_j \mathbf{v}_j^T \mathbf{F'}^T}{\lambda_j + \mu} \tag{40}$$