5. Pollster: Regularized Least Squares

(a) $\mathbf{A} = \mathbb{R}^{90\times 10}$, $\vec{x} = \mathbb{R}^{10\times 1}$, $\vec{b} = \mathbb{R}^{90\times 1}$

$\mathbf{A}$ is constructed as the first 90 examples in the dataset, i.e. the 90 feature vectors of the first 90 counties. It has dimensions 90×10.

$\vec{x}$ is the model vector we're trying to train and optimize, i.e. the weight vector that would optimize the predictions of the voting decisions of any county. It has dimensions 10×1.

$\vec{b}$ is constructed as the 90 actual voting decisions of the first 90 counties. It has dimensions 90×1.

(b)

Using IPython, I found the solved for $\vec{\tilde{x}}$ (vector indicated on IPython), and using this linear model, I evaluated that the total prediction error on the training data is: 0.8633; the total prediction error on the testing data is: 0.3203.

(c) Direct Proof

First, since the eigenvalues of $\mathbf{A}^T\mathbf{A}$ are denoted as λ_i, so as we proved in HW earlier, we have that the eigenvalues of $\mathbf{A}^T\mathbf{A}^{-1}$ are $\frac{1}{\lambda_i}$.

The solution $\vec{\tilde{x}}$ to the least squares problem $\mathbf{A}\vec{x} = \vec{b}$ is, by our formula:

$$\vec{\tilde{x}} = (\mathbf{A}^T\mathbf{A})^{-1}\mathbf{A}^T\vec{b}$$

Thus, $\vec{e} = \vec{b} - \mathbf{A}\vec{\tilde{x}} = \vec{b} - \mathbf{A}(\mathbf{A}^T\mathbf{A})^{-1}\mathbf{A}^T\vec{b}$, and since $\mathbf{A}^T\vec{b} = \sum_{i=1}^{N} \beta_i\vec{v_i}$, so we can further simplify the error vector (along with the properties of eigenvalues and eigenvectors) as:

$$\vec{e} = \vec{b} - \mathbf{A}(\mathbf{A}^T\mathbf{A})^{-1}\sum_{i=1}^{N}\beta_i\vec{v_i} = \vec{b} - \mathbf{A}\sum_{i=1}^{N}\frac{\beta_i}{\lambda_i}\vec{v_i}$$

Therefore,

$$\|\vec{e}\| = \|\vec{b} - \mathbf{A}\sum_{i=1}^{N}\frac{\beta_i}{\lambda_i}\vec{v_i}\| = \left\|\vec{b} - \mathbf{A}\left(\frac{\beta_1}{\lambda_1}\vec{v_1} + \frac{\beta_2}{\lambda_2}\vec{v_2} + \cdots + \frac{\beta_N}{\lambda_N}\vec{v_N}\right)\right\|$$

as desired. Q.E.D.

(d) Yes, we do.

The maximum eigenvalue is about 526, and all other eigenvalues (except one at about 377) is less than 10, which means that we're encountering the issue described in part (c).

(e) $\tilde{\mathbf{A}} = \mathbb{R}^{100\times 10}$, $\vec{\tilde{b}} = \mathbb{R}^{100\times 1}$

By the nature of concatenation, since the identity matrix $\mathbf{I}$ would have dimensions 10×10, so $\tilde{\mathbf{A}}$ has dimensions 100×10, and $\vec{\tilde{b}}$ has dimensions 100×1. (Implemented with IPython.)

(f) Direct Proof

First, define some variables for notation: let

$$\tilde{\mathbf{A}} = \begin{bmatrix} \mathbf{A} \\ \hline \sqrt{\gamma}\,\mathbf{I} \end{bmatrix}, \qquad \tilde{\vec{b}} = \begin{bmatrix} \vec{b} \\ \hline \vec{0} \end{bmatrix}$$

So, the solution $\tilde{\vec{x}}$ to the least squares problem $\tilde{\mathbf{A}}\vec{x} = \tilde{\vec{b}}$ is, by our formula:

$$\tilde{\vec{x}} = (\tilde{\mathbf{A}}^T\tilde{\mathbf{A}})^{-1}\tilde{\mathbf{A}}^T\tilde{\vec{b}}$$

Thus, using the hint regarding matrix-matrix multiplication handled in blocks, we have that:

$$\tilde{\mathbf{A}}^T\tilde{\mathbf{A}} = \left[\mathbf{A}^T \middle| \sqrt{\gamma}\,\mathbf{I}\right]\begin{bmatrix} \mathbf{A} \\ \hline \sqrt{\gamma}\,\mathbf{I} \end{bmatrix} = \mathbf{A}^T\mathbf{A} + \gamma\mathbf{I}$$

$$\tilde{\mathbf{A}}^T\tilde{\vec{b}} = \left[\mathbf{A}^T \middle| \sqrt{\gamma}\,\mathbf{I}\right]\begin{bmatrix} \vec{b} \\ \hline \vec{0} \end{bmatrix} = \mathbf{A}^T\vec{b} + \vec{0} = \mathbf{A}^T\vec{b}$$

Therefore, we can conclude that solution to the modified linear least squares problem is:

$$\tilde{\vec{x}} = (\mathbf{A}^T\mathbf{A} + \gamma\mathbf{I})^{-1}\mathbf{A}^T\vec{b}$$

as desired. Q.E.D.

(g) Direct Proof

First, since the eigenvalues of $\mathbf{A}^T\mathbf{A}$ are denoted as λ_i, so for any arbitrary eigenvector $\vec{u_k}$ of $\mathbf{A}^\mathbf{T}\mathbf{A}$, we have that

$$(\mathbf{A}^\mathbf{T}\mathbf{A} + \gamma\mathbf{I})\ \vec{u_k} = \mathbf{A}^\mathbf{T}\mathbf{A}\vec{u_k} + \gamma\mathbf{I}\vec{u_k} = \lambda_k\vec{u_k} + \gamma\vec{u_k} = (\lambda_k + \gamma)\ \vec{u_k}$$

Thus, by definition, all the eigenvectors $\vec{u_i}$ of $\mathbf{A}^\mathbf{T}\mathbf{A}$ are also eigenvectors $\vec{v_i}$ of $\mathbf{A}^\mathbf{T}\mathbf{A} + \gamma\mathbf{I}$ (this result could also be achieved via the diagonalization of $\mathbf{A}^T\mathbf{A}$, which I've used as a sanity check for myself), and with our calculation, the corresponding eigenvalues are $(\lambda_i + \gamma)$. Then, similarly, as we proved in HW earlier, we have that the eigenvalues of $(\mathbf{A}^T\mathbf{A} + \gamma\mathbf{I})^{-1}$ are $\frac{1}{\lambda_i + \gamma}$.

Then, the solution $\vec{\tilde{x}}$ to the least squares problem $\mathbf{A}\vec{x} = \vec{b}$ is, by our formula:

$$\vec{\tilde{x}} = (\mathbf{A}^T\mathbf{A})^{-1}\mathbf{A}^T\vec{b}$$

Thus, $\vec{e} = \vec{b} - \mathbf{A}\vec{\tilde{x}} = \vec{b} - \mathbf{A}(\mathbf{A}^T\mathbf{A})^{-1}\mathbf{A}^T\vec{b}$, and given that $\mathbf{A}^T\vec{b} = \sum_{i=1}^{N}\beta_i\vec{v_i}$, so we can further simplify the error vector (along with the properties of eigenvalues and eigenvectors) as:

$$\vec{e} = \vec{b} - \mathbf{A}(\mathbf{A}^T\mathbf{A})^{-1}\sum_{i=1}^{N}\beta_i\vec{v_i} = \vec{b} - \mathbf{A}\sum_{i=1}^{N}\frac{\beta_i}{\lambda_i + \gamma}\vec{v_i}$$

Therefore,

$$\|\vec{e}\| = \|\vec{b} - \mathbf{A}\sum_{i=1}^{N}\frac{\beta_i}{\lambda_i}\vec{v_i}\| = \left\|\vec{b} - \mathbf{A}\left(\frac{\beta_1}{\lambda_1 + \gamma}\vec{v_1} + \frac{\beta_2}{\lambda_2 + \gamma}\vec{v_2} + \cdots + \frac{\beta_N}{\lambda_N + \gamma}\vec{v_N}\right)\right\|$$

as desired. Q.E.D.

(h)

As γ increases, the eigenvalues of $\mathbf{A}^T\mathbf{A}$ increases as well, and they shift directly up the y-axis.

(i) $\gamma = 9.326$; Error slightly decreased.

The best choice, optimal γ is $\gamma = 9.326$. Here, using the modified least squares with the optimal γ, we achieved Total Prediction Error on testing data as 0.3065, which is just a bit smaller than the total prediction error on testing data calculated in part (b), 0.3203.