Theorems

In this assignment we are trying to classify AML and ALL samples by use of penalized logistic regression. Before we indulge on the adventure of classification we should first explain the most important theorems behind the classification.

Least squares regression

First of all we are trying to model the outcome on the basis of the predictors. In this case the predictors are the gene expressions derived by means of microarrays. The model these outcomes we make use of a classical linear model:

$$Y = f(X) + \varepsilon \ (1.1)$$

We can model f(X) as follows:

$$\hat{y} = f(X) = \ X\vec{\beta} = \beta_0\vec{1} + \beta_1\overrightarrow{x_1} + \cdots + \beta_{p+1}\overrightarrow{x_p} \ (1.2)$$

Where $\vec{1}, \overrightarrow{x_1}, \ldots, \overrightarrow{x_p}$ are the column vectors of the matrix X and hence span the column space of X (denoted col(X)). These column vectors are also the predictor vectors gained from the microarray experiment. The vector $\beta =< \beta_0, \beta_1, \ldots, \beta_{p+1} >^T$ are the regression coefficient we are going to find. Now that we have derived this representation we can take a look at the following problem. First of all, if X is not an (p+1)X(p+1) matrix we have an inconsistent system. This means that the outcome vector $\vec{y}$ is not in the column space of the matrix X, hence we cannot find regression coefficients that would give a linear combination of the predictor vectors that constitutes the outcome vector. To overcome this problem we can make use of a least square solution. We do this by projecting the outcome vector $\vec{y}$ on the column space of the matrix X which has a least square error, this projection we can describe as a linear combination of the prediction vectors. We can now define the following definition:

If X is an nX(p+1) matrix and $\vec{y}$ is in $\mathbb{R}^n$, a least square solution of $X\vec{\beta} = \vec{y}$ is a vector $\hat{\beta}$ in $\mathbb{R}^{p+1}$ such that:

$$\left\|\vec{y} - X\hat{\beta}\right\| \leq \left\|\vec{y} - X\vec{\beta}\right\| \ (1.3)$$

Theorem 1.1 The least squares theorem:

The solution of the problem is given by:

$$X^TX\hat{\beta} = X^T\vec{y} \ \ or \ \ \hat{\beta} = (X^TX)^{-1}X^T\vec{y} \ (1.4)$$

Proof 1.1:

One can proof this derivation in the following two ways:

a) $RSS(\beta) = \left\|\vec{y} - X\vec{\beta}\right\|_2^2 = (\vec{y} - X\vec{\beta})^T(\vec{y} - X\vec{\beta}) = (X\vec{\beta})^T(X\vec{\beta}) - 2(X\vec{\beta})^T\vec{y} + \vec{y}^T\vec{y}$

Using matrix calculus and setting the derivative in terms of β to zero we derive the following formula:

$$\frac{\partial RSS(\beta)}{\partial \beta} = 2X^T X\vec{\beta} - 2X^T\vec{y} = 0$$
$$X^T X\vec{\beta} = X^T\vec{y}$$
$$\hat{\beta} = (X^T X)^{-1}X^T\vec{y}$$

b) We are trying to minimize:

$$\|\vec{y} - X\hat{\beta}\|$$

We know that:

$$X\hat{\beta} = proj_{col(A)}\vec{y}$$

and it is clear that:

$$\vec{y} - X\hat{\beta} = \vec{y} - proj_{col(X)}\vec{y} = perp_{col(X)}\vec{y}$$

This is orthogonal to col(X). So, if $\vec{x}_i$ is a column vector of X, we have

$$\vec{x}_i{}^T\left(\vec{y} - X\hat{\beta}\right) = 0$$

This is true if and only if:

$$X^T\left(\vec{y} - X\hat{\beta}\right) = 0$$

Which is equivalent to:

$$X^T\vec{y} - X^T X\hat{\beta} = 0$$
$$X^T X\vec{\beta} = X^T\vec{y}$$
$$\hat{\beta} = (X^T X)^{-1}X^T\vec{y}$$

Now that we know the formula for $\widehat{\beta}$, we now can define properties of what is called the Hat matrix which projects $\vec{y}$ on the column space of X:

$$X\hat{\beta} = X(X^T X)^{-1}X^T\vec{y} = H\vec{y}$$

Theorem 1.2:

a) The Hat matrix is symmetric
b) The Hat matrix is idempotent, meaning that the eigenvalues are 0 or 1. Which also constitutes that the Hat matrix is semi-positive definite
c) If $\vec{y}\,\epsilon\, null(X^T)$ it will be projected to the $\vec{0}$ vector
d) If $\vec{y}\epsilon\, col(X)$ it will remain $\vec{y}$ after projection
e) There is only a least square solution when the columns of X are linearly independent

Proof 1.2:

a) $H^T = (X(X^T X)^{-1}X^T)^T = X(X^T X)^{-1}X^T$
b) $HH = X(X^T X)^{-1}X^T X(X^T X)^{-1}X^T = X(X^T X)^{-1}X^T = H$
c) $H\vec{y} = X(X^T X)^{-1}X^T\vec{y} = X(X^T X)^{-1}\vec{0} = \vec{0}$
d) If $\vec{y}\epsilon\, col(X)$ it can be written as a linear combination such as: $\vec{y} = c_1\vec{x}_1 + \cdots + c_n\vec{x}_n = X\vec{c}$ this leads to $X(X^T X)^{-1}X^T\vec{y} = X(X^T X)^{-1}X^T X\vec{c} = X\vec{c} = \vec{y}$

e) If we go back to the form $X^T X\hat{\beta} = X^T\vec{y}$ we can prove that $X^T X$ is invertible if the columns of X are linearly independent.

$$\hat{\beta}^T X^T X\hat{\beta} = \vec{0}$$
$$\left(\beta_0\vec{1} + \beta_1\overrightarrow{x_1} + \cdots + \beta_{p+1}\overrightarrow{x_p}\right)^T\left(\beta_0\vec{1} + \beta_1\overrightarrow{x_1} + \cdots + \beta_{p+1}\overrightarrow{x_p}\right) = \vec{0}\ if\ and\ only\ if$$
$$\hat{\beta} = \vec{0}\ if\ the\ columns\ of\ X\ are\ linearly\ independent$$

The theory of least squares can also be viewed in another light. First of all we know that the Hat matrix projects the outcome vector on the column space of the matrix X. Orthogonal to the column space is the null space of the transpose of X. As shown in figure 1 these spaces are orthogonal to each other. Meaning that the basis vectors of these spaces are orthogonal to one another.

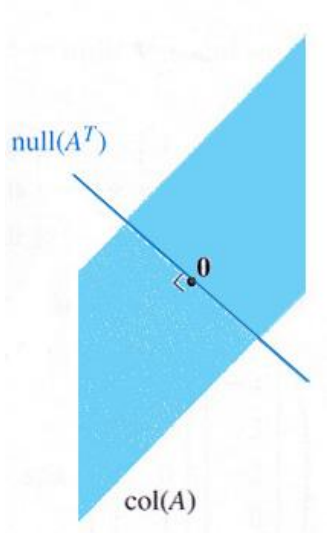

Figure 1: Orthogonal spaces

What the Hat matrix actually does is preserving the projection on the column space of X of the outcome vector. This statement comes from what is called the "Orthogonal decomposition" theorem which states that:

$$\vec{y} = proj_{col(X)}\vec{y} + perp_{col(X)}\vec{y} = \hat{y} + \varepsilon = H\vec{y} + (I - H)\vec{y}$$

An figure of this theorem one can observe in figure 2.

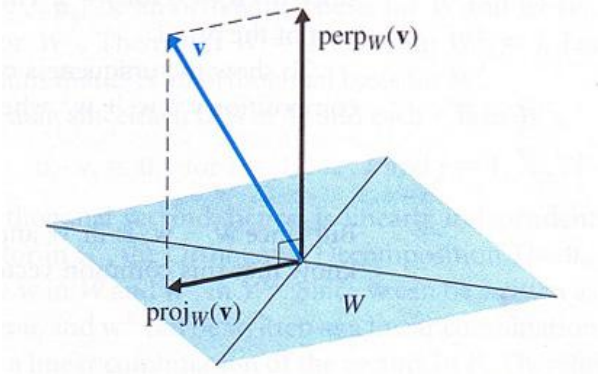

Figure 2: Orthogonal Decomposition of vector $\vec{v}$

It can easily be shown that the matrix (I-H) is also a projection matrix just like matrix H and that it has the same properties (not shown). This matrix is nothing more than a projection matrix for the null space of the transpose of X and can be used to get the residuals after projection. We know that $\vec{\varepsilon}$ is the projected vector on the null space of the transpose of X. To get the residual sum of squares we just have to calculate $||\vec{\varepsilon}||^2$.

Ridge regression

First we must clarify why we should not be satisfied by a least squares solution:

a) Prediction accuracy: The least squares estimates often have a low bias but large variance. The prediction accuracy sometimes can be improved by shrinking or setting some regression coefficients to zero. By doing so we sacrifice a little bit of bias to reduce the variance of the predicted values, and hence may improve the overall prediction accuracy.
b) Interpretation: With a large number of predictors, we often would like to determine a smaller subset that exhibits the strongest effects.

Both issues are occurring due to multi-collinearity. In regression when several predictors are highly correlated, the issue of multi-collinearity occurs. In a regression model we expect a high variance (R^2) explained. The higher the variance explained, the better the model. In a model where collinearity exists we expect that the model parameters and the variance are inflated. The high variance is not explained by independent good predictors, but is due to a misspecified model that caries mutually dependent and thus redundant predictors. To cope for these issues we can make use of Ridge regression on a continuous level.

Ridge regression shrinks the regression coefficients by imposing a penalty on their size. It does this by minimizing the following expression:

$$RSS(\hat{\beta}^{ridge}) = \underset{\beta}{argmax}\{\sum_{i=1}^{N}(y_i - \beta_0 - \sum_{j=0}^{p} x_{ij}\beta_j)^2 + \lambda\sum_{j=1}^{P}\beta_j^2\} = (\vec{y} - X\vec{\beta})^T(\vec{y} - X\vec{\beta}) + \lambda\vec{\beta}^T\vec{\beta}$$

Just like the least squares solution we can use Matrix calculus to minimize this expression. By taking the derivative of the residual sum of squares in terms of the regression coefficients and setting this equals to zero we get the following:

$$\frac{\partial RSS(\hat{\beta}^{ridge})}{\partial\beta} = 2X^TX\vec{\beta} - 2X^T\vec{y} + 2\lambda\vec{\beta} = 0$$
$$X^TX\vec{\beta} + \lambda\vec{\beta} = X^T\vec{y}$$
$$\hat{\beta}^{ridge} = (X^TX + \lambda I)^{-1}X^T\vec{y}$$

It is possible that there are many highly correlated variables in a linear regression model, these parameters can become poorly determined and exhibit high variance. A large positive coefficient for one variable can be canceled by a large negative coefficient due to another correlated variable. This can be prevented by imposing penalties on the size of the coefficients. Due to the ridge solutions not being equivariant under scaling of the inputs, one should first standardize the inputs. It should be clear that you don't want to punish the constant term. Now that the inputs are centered we need to know an estimate for the constant term β_0. This can be estimated by the following formula:

$$\beta_0 = \bar{y} = \frac{1}{n}\sum_{i=1}^{n} y_i$$

It is noted in the book of Friedman J., et al. [1] that the solution adds a positive constant to the diagonal of $X^T X$ before inversion. This should make the problem nonsingular, even if $X^T X$ is not of full rank, and it was the main motivation for Ridge regression when it was first introduced by Hoerl and Kennard in statistics. The basic idea behind this could be that one want to move the Gerschgorin disks such that the matrix becomes nonsingular.

Definition: Let X be a real or complex $n \times n$ matrix and let r_i denote the sum of the absolute values of the off-diagonal entries in the *i*th row of X. That is, $r_i = \sum_{j \neq i}^{n} |x_{ij}|$. The *i*th Gerschgorin disk is the circular disk D_i in the complex plane with center x_{ii} and radius r_i. That is,

$$D_i = \{z \; in \; \mathbb{C}: |z - x_{ii}| \leq r_i\}$$

Gerschgorin's Disk Theorem: Let X be an $n \times n$ real or complex matrix. Then every eigenvalues of X is contained with a Gerschgorin disk.

Proof:

Let λ be an eigenvalues X with corresponding eigenvector $\vec{v}$. Let v_i be the entry of $\vec{v}$ with the largest absolute values. Then $X\vec{v} = \lambda\vec{v}$, the *i*th row of which is:

$$\sum_{j=1}^{n} x_{ij} v_j = \lambda v_i$$

Rearranging we have:

$$(\lambda - x_{ii}) v_i = \sum_{j \neq i}^{n} x_{ij} v_j \quad => \quad \lambda - x_{ii} = \frac{\sum_{j \neq i}^{n} x_{ij} v_j}{v_i}$$

Because $v_i \neq 0$, we obtain:

$$|\lambda - x_{ii}| = \left| \frac{\sum_{j \neq i}^{n} x_{ij} v_j}{v_i} \right| = \frac{|\sum_{j \neq i}^{n} x_{ij} v_j|}{|v_i|} \leq \frac{\sum_{j \neq i}^{n} |x_{ij} v_j|}{|v_i|} = \frac{\sum_{j \neq i}^{n} |x_{ij}||v_j|}{|v_i|} \leq \sum_{j \neq i}^{n} |x_{ij}| = r_i$$

Because $|x_j| \leq |x_i| \; for \; j \neq i$. This means that the eigenvalues λ is contained within the Gerschgorin disk centered at x_{ii} with radius r_i. In figure 3 we can see some examples of Gerschgorin disks in the complex plane.

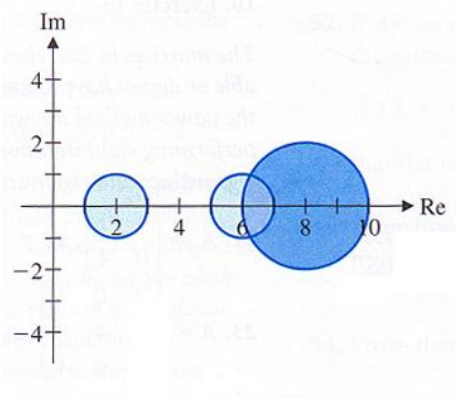

Figure 3: Gerschgorin disks in the complex plane

Because $X^T X$ is symmetric we also know that its eigenvalues are all real. This means that the Gerschgorin disks are just intervals on the real line. By adding constants to the diagonal entries of a matrix we are moving the Gerschgorin disks towards (larger) positive values while the radius stays the same. Eventually the matrix becomes what is called strictly diagonally dominant, where the absolute value of each diagonal entry is larger than the radius. This would mean that the matrix is always invertible as no disk overlaps the origin (0+i0). We know the following theorem:

$$\det(X) = \prod_{i=1}^{n} \lambda_i$$

By adding constants to the diagonal entries the eigenvalues will not become zero anymore and will turn the matrix in a nonsingular one.

As last part we can analyze the nature of Ridge regression. In the first part we have standardized the inputs which resulted in the new matrix X. Now we take the Singular Value Decomposition of this matrix, which is given by:

$$X = U\Sigma V^T$$
$$U: N \times p \; orthogonal \; matrix \; of \; which \; the \; columns \; spans \; the \; column \; space \; of \; X$$
$$\Sigma: P \times P \; diagonal \; matrix \; with \; the \; singular \; values$$
$$V: P \times P \; orthogonal \; matrix \; of \; which \; the \; column \; spans \; the \; row \; space \; of \; X$$

For the least squares solution this will give:

$$X\hat{\beta}^{LS} = X(X^T X)^{-1} X^T y = U\Sigma V^T (VDV^T)^{-1} V\Sigma \mathrm{U^T y} = U\Sigma V^T VD^{-1} V^T \, V\Sigma \mathrm{U^T y} = \mathrm{UU^T y}$$
$$= \sum_{\mathrm{i=1}}^{\mathrm{p}} \mathrm{u_i u_i^T y}$$

This would indicate that the solution is a linear combination of the basis vectors spanning the column space. We can also do this for Ridge regression as it has an analytic solution:

$$X\hat{\beta}^{ridge} = X(X^TX + \lambda I)^{-1}X^T\vec{y} = U\Sigma V^T(VDV^T + \lambda I)^{-1}V\Sigma U^T\vec{y} = U\Sigma V^T(VDV^T + \lambda VV^T)^{-1}V\Sigma U^T\vec{y} = U\Sigma V^T(V(D + \lambda I)V^T)^{-1}V\Sigma U^T\vec{y} = U\Sigma V^TV(D + \lambda I)^{-1}V^TV\Sigma U^T\vec{y} = U\Sigma(D + \lambda I)^{-1}\Sigma U^T\vec{y} = \sum_{i=1}^{p} u_i \frac{\delta_i^2}{\delta_i^2 + \lambda} u_i^T \vec{y}$$

Here the values δ_i^2 are the singular values squared which are just the eigenvalues of X^TX. We can clearly see that the basis vectors of the column space get shrunken. It is also clear from the formula that the lower the value δ_i^2 the more shrunken the basis vector gets. The basic idea behind Ridge regression is that when one centers the input vector on can perform principal component analysis on the matrix. The principal components are in this case the basis vectors that span the column space of the centered matrix X. This method wants to preserve the column vectors which exhibit the most variance according to the eigenvalues. The directions with the smallest variance get shrunken the most by this method.
As last step we also have the effective degrees of freedom statistic for a given λ, denoted by df(λ). This can be calculated as follows:

$$df(\lambda) = tr(X(X^TX + \lambda I)^{-1}X^T) = \sum_{i=1}^{p} u_i \frac{\delta_i^2}{\delta_i^2 + \lambda} u_i^T = \sum_{i=1}^{p} \frac{\delta_i^2}{\delta_i^2 + \lambda}$$

This is equal to the amount of retained variance of the basis vectors of the column space. If we would set $\lambda = 0$ we would just get the least squares dimension, namely p.

Logistic regression

To perform logistic regression we have the following assumptions and definitions:

Odds:

- If an event has probability P, it has odds $\frac{P}{1-P}$
- Odds go from 0 to ∞

Assumptions:

- Response y_i Bernoulli distributed: P(y_i=1)=P
- Logistic regression: $\ln(\frac{P}{1-P}) = \eta_i$
- Linear predictor $\eta_i = \alpha + \vec{x}^t\vec{\beta}$
- Or when offset incorporated $\eta_i = \vec{x}^t\vec{\beta}$

Because we are working with Bernoulli variables, we have a two class problem and can easily derive the respective formulas the following way:

$$\ln\left(\frac{P(x|\omega_1)}{P(x|\omega_2)}\right) = \alpha + \vec{x}^t\vec{\beta}$$

$$\ln\left(\frac{P(\omega_1|x)}{P(\omega_2|x)}\right) - \ln\left(\frac{P(\omega_1)}{P(\omega_2)}\right) = \alpha + \vec{x}^t\vec{\beta}$$

$$\ln\left(\frac{P(\omega_1|x)}{P(\omega_2|x)}\right) = \alpha' + \vec{x}^t\vec{\beta} \ , where\ \alpha' = \alpha + \ \ln\left(\frac{P(\omega_1)}{P(\omega_2)}\right)$$

$$\ln\left(\frac{1 - P(\omega_2|x)}{P(\omega_2|x)}\right) = \alpha' + \vec{x}^t\vec{\beta}$$

$$\frac{1}{P(\omega_2|x)} = 1 + e^{\alpha' + \vec{x}^t\vec{\beta}}$$

$$P(\omega_2|x) = \frac{1}{1 + e^{\alpha' + \vec{x}^t\vec{\beta}}}$$

$$P(x|\omega_1)\left(1 + e^{\alpha' + \vec{x}^t\vec{\beta}}\right) = e^{\alpha' + \vec{x}^t\vec{\beta}}$$

$$P(x|\omega_1) = \frac{e^{\alpha' + \vec{x}^t\vec{\beta}}}{1 + e^{\alpha' + \vec{x}^t\vec{\beta}}} = \frac{1}{1 + e^{-(\alpha' + \vec{x}^t\vec{\beta})}}$$

To find the optimal vector $\vec{\beta}$ we make use of the (log)-likelihood:

$$L(\vec{\beta}, \vec{y}) = \prod_{i=1}^{n} {P_i}^{Y_i}(1 - P_i)^{(1 - Y_i)}$$

$$l(\vec{\beta}, \vec{y}) = \sum_{i=1}^{n} Y_i \ln\left(P_i\right) + (1 - Y_i)\ln\left(1 - P_i\right)$$

The vector $\vec{x}\ and\ \vec{\beta}$ now both incorporate the offset:

$$l(\vec{\beta}, \vec{y}) = \sum_{i=1}^{n} y_i \vec{x}_i^t \vec{\beta} - \ln\left(1 + e^{\vec{x}_i^t\vec{\beta}}\right)$$

Taking the derivative (Matrix calculus) results in:

$$\frac{\partial l(\vec{\beta}\,)}{\partial \vec{\beta}} = \sum_{i=1}^{n} y_i\, \vec{x}_i - \frac{\vec{x}_i}{1 + e^{\vec{x}_i^t\vec{\beta}}} = X^T(\vec{y} - \vec{p})$$

Now that we have the gradient we still need the Hessian, which is:

$$\frac{\partial^2 l(\vec{\beta}\,)}{\partial \vec{\beta}\ \partial \vec{\beta}^T} = -\sum_{i=1}^{n} x_i x_i^t p_i(1 - p_i) = -\ X^T W X$$

$$W_{kk} = p_k(1 - p_k) and\ W_{kl} = 0\ when\ k \neq l$$

Now that we have the gradient and the Hessian, we now can use the Newton-Rhapson method to find the maximum likelihood estimator $\overrightarrow{\beta}$. Because the log likelihood is concave and everywhere twice differentiable we can find an estimate for $\overrightarrow{\beta}$ the following way:

$$\vec{\beta}_{i+1} = \vec{\beta}_i - (\frac{\partial^2 l(\vec{\beta})}{\partial\vec{\beta}\,\partial\vec{\beta}^T})^{-1}\frac{\partial l(\vec{\beta})}{\partial\vec{\beta}} = \vec{\beta}_i + (X^T W X)^{-1}X^T(\vec{y}-\vec{p})$$

To penalize the logistic regression we can define the following formula:

Maximize: $P(\beta) = l(\beta) - \frac{\lambda}{2}\beta^T\beta$

This yields the following Newton-Rhapson derivation:

$$\frac{\partial P(\vec{\beta})}{\partial\vec{\beta}} = X^T(\vec{y}-\vec{p}) - \lambda\vec{\beta}$$

$$\frac{\partial^2 P(\vec{\beta})}{\partial\vec{\beta}\,\partial\vec{\beta}^T} = -X^T W X - \lambda I$$

$$\vec{\beta}_{i+1} = \vec{\beta}_i - (\frac{\partial^2 P(\vec{\beta})}{\partial\vec{\beta}\,\partial\vec{\beta}^T})^{-1}\frac{\partial P(\vec{\beta})}{\partial\vec{\beta}} = \vec{\beta}_i + (X^T W X+\lambda I)^{-1}(X^T(\vec{y}-\vec{p}) - \lambda\beta_i)$$

Akaike information criterion and Leave-one-out Cross Validation

To calculate the Akaike information criterion (AIC) we must first define an appropriate Hat matrix for penalized logistic regression. This is given by:

$$H = WX(X^T W X + \lambda I)^{-1}X^T$$

To estimate a good λ, we can use the following formula:

$$AIC(\lambda) = -2P(\vec{\beta}) + 2edf(\lambda) = 2(edf(\lambda) - P(\vec{\beta}))$$

Where $edf(\lambda) = tr(H)$

Leave-one-out Cross Validation is based on another idea. Here we are trying to estimate λ based on the remaining data after one sample is taken away. We are trying to predict the observed value from the remaining observations and choose λ which yields the best prediction. If $\hat{y}_{-i}^{\lambda}(x_i)$ denotes the prediction of $y(x_i)$ when $y(x_i)$ is removed from the data the Leave-one-out Cross Validation criterion is:

$$LOOCV(\lambda) = \frac{1}{m}\sum_{i=1}^{m}(y(x_i) - \hat{y}_{-i}^{\lambda}(x_i))^2$$

Another way to find the optimal β vector

We know that when the algorithm converges that the gradient equals zero (extrema):

$$X^T(y-p)-\lambda\vec{\beta}=\vec{0}$$
$$X^T(y-p)=\lambda\vec{\beta}$$

We now can say that the beta vector is in the column space of X^T. This means that we can express the beta vector in the following expression:

$$\vec{\beta}=X^T\gamma$$

For some $n\times 1$ gamma vector. This leads to a reparameterization of the beta vector in terms of the gamma vector. Now we can optimize the gamma vector through the same Newton-Raphson method and gain the optimal beta vector.

$$\tilde{P}(\beta)=P(X^T\gamma)$$
$$\frac{\partial\tilde{P}}{\partial\gamma}=\frac{\partial P}{\partial\beta}\frac{\partial\beta}{\partial\gamma}=X(X^T(y-p)-\lambda\beta)$$
$$\frac{\partial\tilde{P}}{\partial\gamma\partial\gamma^T}=-XX^TWXX^T-\lambda XX^T=-X(X^TWX-\lambda I)X^T$$