CS 754 Assignment 4 : Report

Mls Pragna - 180050064
Rahul Puli - 180050080

Question 1

(a)

To run the code : "Run Q1.m"

The plots VE vs logarithm of lambda and RMSE vs logarithm of lambda have very similiar pattern that is relatively they are same. Both Validation error and RMSE values decrease from $\lambda = 0.0001$ to $\lambda = 15$ and increase from $\lambda = 15$ to $\lambda = 100$.

The optimal values of λ from the two agree that is the minimum error for both plots is obtained at $\lambda = 15$.

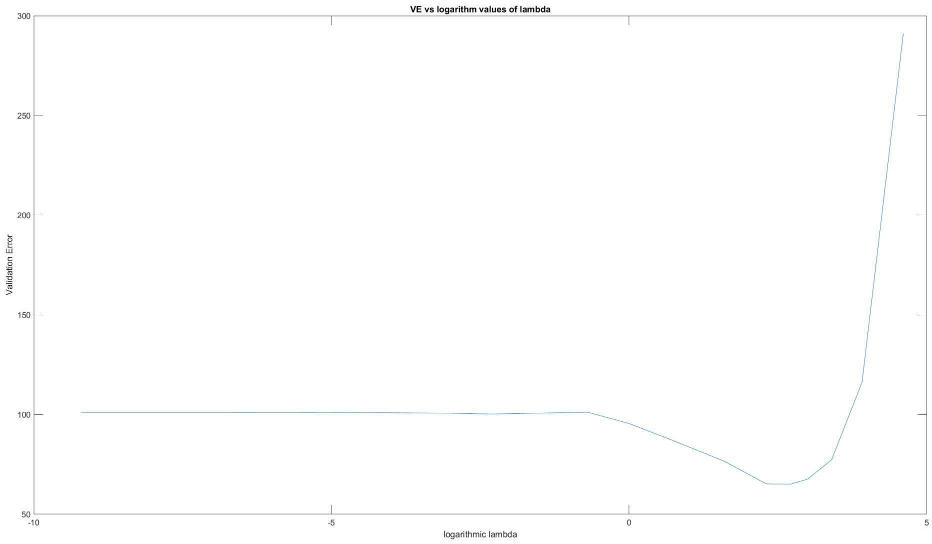

Figure 1: *
Validation Error Vs logarithm of lambda values

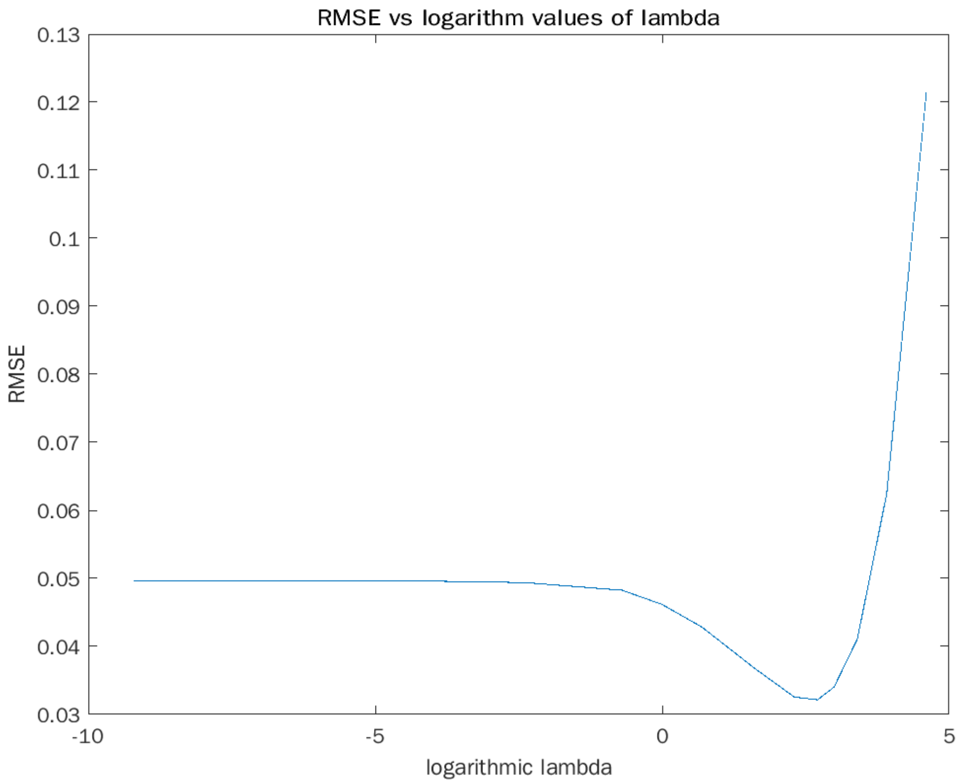

Figure 2: *
RMSE Vs logarithm of lambda values

(b)

If validation set and reconstruction set are not disjoint then during reconstructing the signal, the model has seen data that is used in validation, so the validation error will be overly optimistic. Therefore, a wrong lambda could give less validation error and hence producing the wrong reconstruction signal x which implies RMSE optimal lambda would not match with optimal lambda obtained by validation error that is the RMSE and VE plots would not have similar graphs.

(c)

The **theorem 1** from the paper `https://ieeexplore.ieee.org/document/6854225` refers to the proxying ability.

Theorem 1 ((Recovery error estimation) :

Provided that m_{cv} is sufficiently large, with probability $erf(\frac{\lambda}{\sqrt{(2)}})$ the following holds,

$$h(\lambda,+)\epsilon_{cv} - \sigma_n^2 \leq \epsilon_x \leq h(\lambda,-)\epsilon_{cv} - \sigma_n^2$$

where $h(\lambda,\pm) = \frac{m}{m_c v}\frac{1}{1\pm\lambda\sqrt{\frac{2}{m_{cv}}}}$ and $erf(u) = \frac{1}{\sqrt{\pi}\int_{-u}^{u} e^{-t^2}}dt$

Here ϵ_{cv} is validation error and ϵ_x is the mean squared error.

As, we don't know the ground truth signal x, mean squared error cannot be determined but this theorem 1 gives us the probability $erf(\frac{\lambda}{\sqrt{(2)}})$, with this probability we can bound the mean squared error ϵ_x by the interval,

$$[h(\lambda,+)\epsilon_{cv} - \sigma_n^2, h(\lambda,-)\epsilon_{cv} - \sigma_n^2]$$

The difference of these upper bound and lower bound is roughly proportional to $1/m_{cv}^{2/3}$ and becomes tighter as m_{cv}(no.of measurements used for validation) increases.

(d)

From the book by Tibshirani and others, the lambda is set to

$$\lambda_N = 2\sigma\sqrt{T\frac{logp}{N}}$$

for some $T > 2$

This certain value of λ assumes the noise is gaussian and it requires the noise level as σ is involved in estimation of λ_N whereas to obtain lambda from cross validation the prior knowlege on noise level or sparsity is not required.

Question 2

(a)

Let the derivative filter applied on class S images to get class S_1 be f.
Let D_1 be the dictionary that sparsely represents the class of images S_1.
Let I_1 be the image in class S_1 obtained by after applying the derivative filter to image I in class S.

$$I_1 = I * f$$

where * is convolution operator.
Let I(:) is obtained by reshaping the image I to a column vector.

$$I(:) = \sum_i x_i d_i$$

$$\Rightarrow (I * f)(:) = (reshape(\sum_i x_i d_i,\ size\ of\ image\ I)) * f$$

$$\Rightarrow I_1(:) = \sum_i x_i((reshape(d_i,\ size\ of\ image\ I)) * f)(:)$$

$$\Rightarrow I_1(:) = \sum_i x_i d_{1i}$$

$$\Rightarrow d_{1i} = ((reshape(d_i,\ size\ of\ image\ I)) * f)(:)$$

Hence each column of the dictionary D_1 is obtained by reshaping it into the size of image I and convolving it with the derivative filter f and then reshaping it to the column vector.

(b)

Let D_2 be the dictionary that sparsely represents the class of images S_2.
Let I_r be the image in class S_6 obtained by rotating the image I in class S.
Let A1 be the subset of images in S_2 that are rotated by an fixed angle α.
Let A2 be the subset of images in S_2 that are rotated by an fixed angle β.

$$I_r(x,y) = I(xcos\alpha + ysin\alpha, -xsin\alpha + ycos\alpha)\ \ \forall I \epsilon A_1$$

$$I_t(x,y) = I(xcos\beta + ysin\beta, -xsin\beta + ycos\beta)\ \ \forall I \epsilon A_2$$

For $I \epsilon A_1$
Let I(:) is obtained by reshaping the image I to a column vector.

$$I(:) = \sum_i a_i d_i$$

$$\Rightarrow I_t(:) = (reshape(\sum_i a_i d_i,\ size\ of\ image\ I)(xcos\alpha + ysin\alpha, -xsin\alpha + ycos\alpha))(:)$$

$$\Rightarrow I_t(:) = \sum_i a_i(reshape(d_i,\ size\ of\ image\ I)(xcos\alpha + ysin\alpha, -xsin\alpha + ycos\alpha))(:)$$

$$\Rightarrow I_t(:) = \sum_i a_i d_{A1_i}$$

$$\Rightarrow d_{A1_i} = (reshape(d_i,\ size\ of\ image\ I)(xcos\alpha + ysin\alpha, -xsin\alpha + ycos\alpha))(:)$$

Hence each column of the dictionary D_{A1} is obtained by reshaping it into the size of image I and rotating it by fixed angle α.

For $I \epsilon A_2$
Let I(:) is obtained by reshaping the image I to a column vector.

$$I(:) = \sum_i a_i d_i$$

$$\Rightarrow I_t(:) = (reshape(\sum_i a_i d_i,\ size\ of\ image\ I)(xcos\beta + ysin\beta, -xsin\beta + ycos\beta))(:)$$

$$\Rightarrow I_t(:) = \sum_i a_i(reshape(d_i,\ size\ of\ image\ I)(xcos\beta + ysin\beta, -xsin\beta + ycos\beta))(:)$$

$$\Rightarrow I_t(:) = \sum_i a_i d_{A2_i}$$

$$\Rightarrow d_{A2_i} = (reshape(d_i,\ size\ of\ image\ I)(xcos\beta + ysin\beta, -xsin\beta + ycos\beta))(:)$$

Hence each column of the dictionary D_{A2} is obtained by reshaping it into the size of image I and rotating it by fixed angle β.

The dictionary D_2 is obtained by column concatenating the dictionaries D_{A1} and D_{A2} that is $D_2 = [D_{A1}|D_{A2}]$ and appropriate padding need to done during creation of dictionaries D_{A1} and D_{A2}.

(c)

Let D_3 be the dictionary that sparsely represents the class of images S_1.

$$I^i_{new}(x,y) = \alpha(I^i_{old}(x,y))^2 + \beta(I^i_{old}(x,y)) + \gamma$$

Let I(:) is obtained by reshaping the image I to a column vector.

$$I(:) = \sum_i a_i d_i$$

$$\Rightarrow I^2(:) = (\sum_i a_i d_i)^2$$

$$\Rightarrow I^2(:) = \sum_{i,j} a_i a_j (d_i . * d_j) \quad .* \text{ } is\ point\ wise\ multiplication$$

$$\Rightarrow \alpha I^2(:) = \sum_{i,j} \alpha a_i a_j (d_i . * d_j)$$

$$\Rightarrow \alpha I^2(:) = \sum_{i,j} a'_{i,j} d_{term1_{i,j}}$$

$$\Rightarrow d_{term1_{i,j}} = d_i . * d_j$$

Hence each column of the dictionary D_{term1} is obtained by point wise multiplication of columns of dictionary D

$$I(:) = \sum_i a_i d_i$$

$$\Rightarrow \beta I(:) = \sum_i \beta a_i d_i$$

$$\Rightarrow \beta I(:) = \sum_i a'_i d_{term2_i}$$

$$\Rightarrow d_{term2_i} = d_i$$

Hence dictionary D_{term2} is equal to dictionary D

As, $I_{new}(:) = \alpha I^2_{old}(:) + \beta I_{old}(:) + \gamma$. The dictionary D_3 is obtained by column concatenating the dictionaries D_{term1} and D_{term2} and 1 that is $D_2 = [D_{term1}|D_{term2}|1]$

(d)

Let the blur kernel applied on class S images to get class S_4 be K.
Let D_4 be the dictionary that sparsely represents the class of images S_3.
Let I_k be the image in class S_4 obtained by after applying the blur kernel to image I in class S.

$$I_k = I * K$$

where * is convolution operator.
Let I(:) is obtained by reshaping the image I to a column vector.

$$I(:) = \sum_i a_i d_i$$

$$\Rightarrow (I * K)(:) = (reshape(\sum_i a_i d_i,\ size\ of\ image\ I)) * K$$

$$\Rightarrow I_k(:) = \sum_i a_i((reshape(d_i,\ size\ of\ image\ I)) * K)(:)$$

$$\Rightarrow I_k(:) = \sum_i a_i d_{ki}$$

$$\Rightarrow d_{ki} = ((reshape(d_i,\ size\ of\ image\ I)) * K)(:)$$

Hence each column of the dictionary D_4 is obtained by reshaping it into the size of image I and convolving it with the blur kernel K and then reshaping it to the column vector.

(e)

Let D_5 be the dictionary that sparsely represents the class of images S_5.
Let I_k be the image in class S_5 obtained by after applying the blur kernel to image I in class S.
Let set B consists of blur kernels K_1, K_2, K_3, K_4 .... ,K_n
Let the coefficients in the linear combination of blur kernels K_1, K_2, K_3, K_4 .... ,K_n be x_1, x_2, x_3, x_4 .... ,x_n
Let K be the blur kernel that is linear combination of kernels K_1, K_2, K_3, K_4 .... ,K_n.

$$k = x_1k_1 + x_2k_2 + ...x_nk_n = \sum_{j=1}^{n} x_ik_i$$

Let I(:) is obtained by reshaping the image I to a column vector.

$$I(:) = \sum_i a_id_i$$

$$\Rightarrow (I * K)(:) = (reshape(\sum_i a_id_i,\ size\ of\ image\ I)) * K$$

$$\Rightarrow I_k(:) = \sum_i a_i((reshape(d_i,\ size\ of\ image\ I)) * K)(:)$$

$$\Rightarrow I_k(:) = \sum_i a_i((reshape(d_i,\ size\ of\ image\ I)) * (\sum_{j=1}^{n} x_ik_i))(:)$$

$$\Rightarrow I_k(:) = \sum_{i,j} a_ix_j((reshape(d_i,\ size\ of\ image\ I)) * k_j)(:)$$

$$\Rightarrow I_k(:) = \sum_{i,j} a'_{i,j}((reshape(d_i,\ size\ of\ image\ I)) * k_j)(:)$$

$$\Rightarrow d_{ki,j} = (reshape(d_i,\ size\ of\ image\ I)) * k_j$$

Let D_{k_i} be the dictionary obtained by applying blur kernel k_i. (from part d).
Hence dictionary D_5 is obtained by column concatenation of dictionaries D_{k_1}, D_{k_2}, ...,D_{k_n}.

(f)

Let D_6 be the dictionary that sparsely represents the class of images S_6.
Let R be the 1D signal in class S_6 obtained by applying a radon transform in a known angle θ to image I in class S.

$$R = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} I(x,y)\delta(\rho - xcos\theta - ysin\theta)dxdy$$

Let I(:) is obtained by reshaping the image I to a column vector.

$$I(:) = \sum_i a_id_i$$

$$\Rightarrow R = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} (reshape(\sum_i a_id_i,\ size\ of\ image\ I))(x,y)\delta(\rho - xcos\theta - ysin\theta)dxdy$$

$$\Rightarrow R = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} (\sum_i a_i((reshape(d_i,\ size\ of\ image\ I)))(x,y)\delta(\rho - xcos\theta - ysin\theta)dxdy$$

$$\Rightarrow R = \sum_i a_i(\int_{-\infty}^{\infty}\int_{-\infty}^{\infty}(reshape(d_i,\ size\ of\ image\ I))(x,y)\delta(\rho - xcos\theta - ysin\theta)dxdy)$$

$$\Rightarrow R = \sum_i a_i d_{6_i}$$

$$\Rightarrow d_{6_i} = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty}(reshape(d_i,\ size\ of\ image\ I))(x,y)\delta(\rho - xcos\theta - ysin\theta)dxdy$$

Hence each column of the dictionary D_6 is obtained by reshaping it into the size of image I and applying the radon transform at an known angle θ.

(g)

Let D_7 be the dictionary that sparsely represents the class of images S_7.
Let I_t be the image in class S_7 obtained by translating the image I in class S.
Let A1 be the subset of images in S_6 that are translated by an fixed offset (x_1, y_1).
Let A2 be the subset of images in S_6 that are translated by an fixed offset (x_2, y_2).

$$I_t(x,y) = I(x - x_1, y - y_1)\ \ \forall I \epsilon A_1$$

$$I_t(x,y) = I(x - x_2, y - y_2)\ \ \forall I \epsilon A_2$$

For $I \epsilon A_1$
Let I(:) is obtained by reshaping the image I to a column vector.

$$I(:) = \sum_i a_i d_i$$

$$\Rightarrow I_t(:) = (reshape(\sum_i a_i d_i,\ size\ of\ image\ I)(x - x_1, y - y_1))(:)$$

$$\Rightarrow I_t(:) = \sum_i a_i(reshape(d_i,\ size\ of\ image\ I)(x - x_1, y - y_1))(:)$$

$$\Rightarrow I_t(:) = \sum_i a_i d_{A1_i}$$

$$\Rightarrow d_{A1_i} = (reshape(d_i,\ size\ of\ image\ I)(x - x_1, y - y_1))(:)$$

Hence each column of the dictionary D_{A1} is obtained by reshaping it into the size of image I and translating it by fixed offset (x_1, y_1).

For $I \epsilon A_2$
Let I(:) is obtained by reshaping the image I to a column vector.

$$I(:) = \sum_i a_i d_i$$

$$\Rightarrow I_t(:) = (reshape(\sum_i a_i d_i,\ size\ of\ image\ I)(x - x_2, y - y_2))(:)$$

$$\Rightarrow I_t(:) = \sum_i a_i(reshape(d_i,\ size\ of\ image\ I)(x - x_2, y - y_2))(:)$$

$$\Rightarrow I_t(:) = \sum_i a_i d_{A2_i}$$

$$\Rightarrow d_{A2_i} = (reshape(d_i,\ size\ of\ image\ I)(x - x_2, y - y_2))(:)$$

Hence each column of the dictionary D_{A2} is obtained by reshaping it into the size of image I and translating it by fixed offset (x_2, y_2).

The dictionary D_6 is obtained by column concatenating the dictionaries D_{A1} and D_{A2} that is $D_6 = [D_{A1}|D_{A2}]$ and appropriate padding need to done during creation of dictionaries D_{A1} and D_{A2}.

Question 3

a)

Given,

$$J(A_r) = ||A - A_r||_F^2$$

where A is a known mxn matrix of rank greater than r, and A_r is a rank-r matrix, where $r < m$, $r < n$.

The singular value decomposition of matrix A,

$$A = UDV^T$$

where the columns of U and V are orthonormal to each other and the matrix D is a diagonal matrix with positive real entries.

To minimize $J(A_r)$, by Eckart–Young–Mirsky theorem, we can construct A_r by using r largest singular values of A that is square root of eigenvalues of A^TA.
Let $U_r = U(:, 1:r)$, $D_r = D(1:r, 1:r)$ and $V_r = V(:, 1:r)$

$$A_r = U_r D_r V_r^T$$

where U_r and V_r are singular vectors corresponding to the r largest singular values of A and D_r has r largest singular values

This optimization is used for rank-1 approximation in KSVD method of learning the bases in dictionary learning.

$$min_{A,S}||Y - AS||^2 \;\; subject\ to\ \forall i, ||s_i||_0 \leq T_0$$

where Y is a signal, A is a dictionary and S consists if sparse vectors.
The dictionary is updated one column at a time,

$$\begin{aligned}
||Y - AS||_F^2 &= ||Y - \sum_{j=1}^{k} a_j s^j||_F^2 \\
&= ||Y - \sum_{j \neq k}^{k} a_j s^j - a_k s^k||_F^2 \\
&= ||E_k - a_k s^k||_F^2
\end{aligned}$$

Rank-1 approximation is done by using this optimization method for E_k by computing the SVD of E_k to get a_k.

b)

Given,

$$J(R) = ||A - RB||_F^2$$

where $A\epsilon R^{mxn}$, $B\epsilon R^{nxm}$, $R\epsilon R^{nxn}$, $m > n$, R is constrained to be orthonormal, A and B are both known.
Let at $R = R^*$, J(R) attains the minimum value.

$$\begin{aligned}
R* &= min_R \; ||A - RB||_F^2 \;\; such\ that\ R^T R = I \\
&= min_R \; trace((A - RB)^T (A - RB)) \\
&= min_R \; trace(A^T A - 2A^T RB + B^T B) \\
&= max_R \; trace(A^T RB) \\
&= max_R \; trace(RBA^T) \qquad\qquad SVD\ of\ BA^T\ gives\ BA^T = UDV^T \\
&= max_R \; trace(RUDV^T) \\
&= max_R \; trace(V^T RUD) \\
&= max_R \; trace(ZD) \\
&= max_R \; \sum_i z_{ii} d_{ii} \\
&\leq \sum_i d_{ii}
\end{aligned}$$

As, $max_R \sum_i z_{ii} d_{ii} \leq \sum_i d_{ii}$, the maximum is attained when $Z = I$. This implies,

$$V^T RU = I$$

$$R = VU^T$$

Therefore, the minimum value of J(R) is obtained at $R = VU^T$ where U and V are singular vectors of BA^T that is $BA^T = UDV^T$

This optimization is used in Union of orthonormal bases algorithm in dictionary learning where R is an over-complete dictionary, R ϵR^{mxn}, $m < n$. This is known as Orthogonal procrustes problem.
Union of Orthonormal bases method,

$$X = AS + \epsilon, \;\; (A, S) = min_{A,S} ||X - AS||^2 + \lambda ||S||_1$$

where A is orthonormal matrix and an over-complete dictionary, X is a signal and S consists if sparse vectors.

$$A* = min_A \; ||X - AS||_F^2 \;\; such\ that\ A^T A = I$$

Orthogonal procrustes optimization problem is used to obtain A^*.

Question 4

(1)

- **Title of the Paper :** Structure-Preserving Color Normalization and Sparse Stain Separation for Histological Images tomographic images reconstruction

- **Authors :** Abhishek Vahadane*, Tingying Peng*, Amit Sethi, Shadi Albarqouni, Lichao Wang, Maximilian Baust, Katja Steiger, Anna Melissa Schlitter, Irene Esposito, and Nassir Navab
- **Published on :** 2016
- **Venue :** Biomedical Imaging (ISBI), IEEE 12th International conference Symposium
- **Link :** `https://ieeexplore.ieee.org/stamp/stamp.jsp?arnumber=7460968`

(2)

Staining and scanning of tissue samples has undesirable color variations resulting from changes in stain vendors' raw materials and manufacturing procedures, lab staining protocols, and digital scanner colour sensitivities. For comparing tissue samples, color normalization and stain separation of the tissue images helpful for both pathologists and software. Non-negative matrix factorization(NMF) technique is used for stain separation, this derives image-specific stain color basis. NMF can be used because both color basis and density are non-negative as stains can only absorb but not emit light.

(3)

Let $I \epsilon R^{mxn}$ be the matrix of RGB intensities and I_0 be the illuminating light intensity on the sample where $m = 3$ for RGB channels, n is the number of pixels.
Let W ϵR^{mxr} be the stain color appereance matrix, the columns of which represent the color basis of each stain where r is the number of stains.
Let H ϵR^{rxn} is stain density maps, the rows of which represent the concentration of each stain.

Intensity I can be formulated as,

$$I = I_0 exp(-WH)$$

Let V be the relative optical density then,

$$V = log\frac{I_0}{I}$$

$$\Rightarrow V = WH$$

We have the matrix V and we find the W and H matrices using NMF.
Stain separation problem is modeled as a NMF to which a sparseness constraint is added.

$$min_{W,H}\frac{1}{2}||V - WH||_F^2, \quad such\ that\ W, H \geq 0$$

where the dictionary matrix is W ϵR^{mxr} which is the stain color matrix and dictionary cofficients is H ϵR^{rxn} which is stain density maps.

Question 5

Given, the projection measurements are represented as a single vector

$$y \sim Poisson(I_0 exp(-Rf))$$

where $y \epsilon R^m$ with m = no.of projection angles x number of bins per angle and I_0 is the power of the incident X-Ray beam; R ϵR^{mxn} represents the Radon operator and f represents the unknown signal in R^n.

Poisson noise

The probability of y_i given f_i,

$$P(y_i|(Rf)_i) = \frac{e^{-I_0 exp(-(Rf)_i)}(I_0 exp(-(Rf)_i))^{y_i}}{y_i!}$$

The objective function is the maximum log-likelihood,

$$P(f|Y) \propto P(Y|f)P(f)$$
$$\Rightarrow -logP(f|Y) = constant - logP(Y|f) - logP(f)$$
$$\Rightarrow -logP(f|Y) = constant - \sum_{i=1}^{n} log \frac{e^{-I_0 exp(-(Rf)_i)}(I_0 exp(-(Rf)_i))^{y_i}}{y_i!} - logP(f)$$
$$\Rightarrow -logP(f|Y) = constant + \sum_{i=1}^{n}(I_0 exp(-(Rf)_i) - y_i log(I_0 exp(-(Rf)_i)) - log(y_i!)) - logP(f)$$
$$\Rightarrow -logP(f|Y) = constant + \sum_{i=1}^{n}(I_0 exp(-(Rf)_i) + y_i(Rf)_i) - logP(f)$$

Assuming, f is coming from a truncated laplacian distribution that is $f_i \sim \frac{e^{-cf_i}}{constant}$

$$\Rightarrow -logP(f|Y) = constant + \sum_{i=1}^{n}(I_0 exp(-(Rf)_i) + y_i(Rf)_i) + \sum_{i=1}^{n}|f_i|$$

Hence, the objective function by maximizing loglikelihood is to minimize E[f],

$$E[f] = \sum_{i=1}^{n}(I_0 exp(-(Rf)_i) + y_i(Rf)_i) + \sum_{i=1}^{n}|f_i|$$

Poisson noise and Gaussian noise

Given, there is also Guassian noise of mean 0 and variance σ in addition to poisson noise.

$$y \sim Poisson(I_0 exp(-Rf)) + \mathcal{N}(I_0 exp(-Rf), \sigma^2)$$

The log-likelihood for guassian noise part is,

$$P(y_i|(Rf)_i) = \frac{e^{-\frac{(y_i - I_0 exp(-Rf)_i)^2}{2\sigma^2}}}{\sigma\sqrt{2\pi}}$$

$$\Rightarrow -logP(Y|f) = constant + \sum_{i=1}^{n}(y_i - I_0 exp(-Rf)_i)^2$$

In this case, the objective function is sum of maximizing loglikelihood of poisson and guassian distribution and prior of distribution of f,

$$E[f] = \sum_{i=1}^{n}(I_0 exp(-(Rf)_i) + y_i(Rf)_i) + \sum_{i=1}^{n}(y_i - I_0 exp(-Rf)_i)^2 + \sum_{i=1}^{n}|f_i|$$