Assignment 4: CS 754, Advanced Image Processing

Due: 4th April before 11:55 pm

Remember the honor code while submitting this (and every other) assignment. All members of the group should work on and understand all parts of the assignment. We will adopt a zero-tolerance policy against any violation.

Submission instructions: You should ideally type out all the answers in Word (with the equation editor) or using Latex. In either case, prepare a pdf file. Create a single zip or rar file containing the report, code and sample outputs and name it as follows: A4-IdNumberOfFirstStudent-IdNumberOfSecondStudent.zip. (If you are doing the assignment alone, the name of the zip file is A4-IdNumber.zip). Upload the file on moodle BEFORE 11:55 pm on the due date. The cutoff is 10 am on 5th April after which no assignments will be accepted. Note that only one student per group should upload their work on moodle. Please preserve a copy of all your work until the end of the semester. If you have difficulties, please do not hesitate to seek help from me.

1. Consider a signal $\boldsymbol{x}$ which is sparse in the canonical basis and contains n elements, which is compressively sensed in the form $\boldsymbol{y} = \boldsymbol{\Phi x} + \boldsymbol{\eta}$ where $\boldsymbol{y}$, the measurement vector, has m elements and $\boldsymbol{\Phi}$ is the $m \times n$ sensing matrix. Here $\boldsymbol{\eta}$ is a vector of noise values that are distributed by $\mathcal{N}(0, \sigma^2)$. One way to recover $\boldsymbol{x}$ from $\boldsymbol{y}, \boldsymbol{\Phi}$ is to solve the LASSO problem, based on minimizing $J(\boldsymbol{x}) \triangleq \|\boldsymbol{y} - \boldsymbol{\Phi x}\|^2 + \lambda \|\boldsymbol{x}\|_1$. A crucial issue is to how to choose λ. One purely data-driven technique is called cross-validation. In this technique, out of the m measurements, a random subset of (say) 90 percent of the measurements is called the reconstruction set R, and the remaining measurements constitute the validation set V. Thus V and R are always disjoint sets. The signal $\boldsymbol{x}$ is reconstructed using measurements only from R (and thus only the corresponding rows of $\boldsymbol{\Phi}$) using one out of many different values of λ chosen from a set Λ. Let the estimate using the g^{th} value from Λ be denoted $\boldsymbol{x_g}$. The corresponding validation error is computed using $VE(g) \triangleq \sum_{i \in V} (y_i - \boldsymbol{\Phi^i x_g})^2 / |V|$. The value of λ for which the validation error is the least is chosen to be the optimal value of λ. Your job is to implement this technique for the case when $n = 500, m = 200, \|\boldsymbol{x}\|_0 = 18, \sigma = 0.05 \times \sum_{i=1}^{m} |\boldsymbol{\Phi^i x}|/m$. Choose $\Lambda = \{0.0001, 0.0005, 0.001, 0.005, 0.01, 0.05, 0.1, 0.5, 1, 2, 5, 10, 15, 20, 30, 50, 100\}$. Draw the non-zero elements of $\boldsymbol{x}$ at randomly chosen location, and let their values be drawn randomly from Uniform(0, 1000). The sensing matrix $\boldsymbol{\Phi}$ should be drawn from $\pm 1/\sqrt{m}$ Bernoulli with probability of $+1/\sqrt{m}$ being 0.5. Now do as follows. Use the L1-LS solver from `https://web.stanford.edu/~boyd/l1_ls/` for implementing the LASSO.

   (a) Plot a graph of VE versus the logarithm of the values in Λ. Also plot a graph of the RMSE versus the logarithm of the values in Λ, where RMSE is given by $\|\boldsymbol{x_g} - \boldsymbol{x}\|_2 / \|\boldsymbol{x}\|_2$. Comment on the plots. Do the optimal values of λ from the two plots agree?

   (b) What would happen if V and R were not disjoint but coincident sets?

   (c) The validation error is actually a proxy for actual mean squared error. Note that you can never determine the mean squared error since the ground truth $\boldsymbol{x}$ is unknown in an actual application. Which theorem/lemma from the paper `https://ieeexplore.ieee.org/document/6854225` (On the theoretical analysis of cross-validation in compressed sensing) refers to this proxying ability? Explain how.

   (d) In your previous assignment, there was a theorem from the book by Tibshirani and others which gave you a certain value of λ. What is the advantage of this cross-validation method compared to the choice of λ using that theorem? Explain. [10+5+5+5=25 points]

Part (a): A sample of plots of VE and RMSE versus $\log \lambda$ are in the homework folder ('crossval.png'). The plots show that the optimal value of λ in terms of least validation error is close to the one in terms of least RMSE, though they may not be exactly equal. In both cases, the RMSE/VE curves decrease with increase in λ, then dip down to a minimum and then increase w.r.t. λ.
Part (b): It is important to have R and V that are disjoint from one another. Suppose you instead had $R = V$. Since the problem is inherently ill-posed without a sparsity constraint, you could find some $\boldsymbol{x}$ such that $\sum_{i \in R}(y_i - \boldsymbol{\Phi^i x})^2$ is zero or close to zero. This is easy to do by choosing some vector from the nullspace of $\boldsymbol{\Phi_R}$. If V and R are the same, this will produce a low validation error, but such an estimate of $\boldsymbol{x}$ will be a very bad one as no sparsity prior has been used, and there are no theoretical guarantees for such an estimate. On the other hand, if V is disjoint from R, the validation error will likely be very high, as is also inferred from theorem 1 (see solution to part (c) as well).
Part (c): Theorem 1 from this paper refers to the proxying ability. It clearly gives a confidence interval for ε_x (the true MSE) in terms of ϵ_{cv} (the cross-validation error), σ_n (the noise standard deviation) and m_{cv} (the number of measurements used for cross-validation). This shows that ϵ_{cv} gives us some indication of the value of ε_x.
Part (d): That theorem indeed gave a lower bound for the value of λ. This derived lower bound is sufficient for the upper bounds on the estimation error, i.e. $\|\boldsymbol{x} - \hat{\boldsymbol{x}_\lambda}\|_2$ to hold. Larger values of λ produce looser upper bounds, as per that theorem. However, for specific instances, the value of λ predicted from that theorem need not produce the best MSE, nor do we have any confidence interval for the MSE. The present cross-validation approach instead gives us such a confidence interval.

2. Consider that you learned a dictionary $\boldsymbol{D}$ to sparsely represent a certain class $\mathcal{S}$ of images - say handwritten alphabet or digit images. How will you convert $\boldsymbol{D}$ to another dictionary which will sparsely represent the following classes of images? Note that you are not allowed to learn the dictionary all over again, as it is time-consuming.

   (a) Class $\mathcal{S}_1$ which consists of images obtained by applying a known derivative filter to the images in $\mathcal{S}$.

   (b) Class $\mathcal{S}_2$ which consists of images obtained by rotating a subset of the images in class $\mathcal{S}$ by a known fixed angle α, and the other subset by another known fixed angle β.

   (c) Class $\mathcal{S}_3$ which consists of images obtained by applying an intensity transformation $I^i_{new}(x, y) = \alpha(I^i_{old}(x, y))^2 + \beta(I^i_{old}(x, y)) + \gamma$ to the images in $\mathcal{S}$, where α, β, γ are known.

   (d) Class $\mathcal{S}_4$ which consists of images obtained by applying a known blur kernel to the images in $\mathcal{S}$.

   (e) Class $\mathcal{S}_5$ which consists of images obtained by applying a blur kernel which is known to be a linear combination of blur kernels belonging to a known set $\mathcal{B}$, to the images in $\mathcal{S}$.

   (f) Class $\mathcal{S}_6$ which consists of 1D signals obtained by applying a Radon transform in a known angle θ to the images in $\mathcal{S}$.

   (g) Class $\mathcal{S}_7$ which consists of images obtained by translating a subset of the images in class $\mathcal{S}$ by a known fixed offset (x_1, y_1), and the other subset by another known fixed offset (x_2, y_2). Assume appropriate zero-padding and increase in the size of the image canvas owing to the translation. [4+4+4+4+4+6+4=30 points]

   Solution:

   (a) Consider image $\boldsymbol{f} \in \mathcal{S}$, we have $\boldsymbol{f} = \boldsymbol{D\theta_f}$. If d is a derivative filter, we have $d * \boldsymbol{f} = \boldsymbol{D_1 \theta_f}$, where $\boldsymbol{D_1}$ is obtained by applying the derivative filter d to every column of $\boldsymbol{D}$.

   (b) Create a new dictionary D_2 which will have $2K$ columns if D has K columns. The first K columns are obtained by rotating each of the columns of D (reshaped to form images of the desired size) by angle α, and the remaining K columns are obtained by rotating each of the columns of D (reshaped to form images of the desired size) by angle β. In this case, it is advisable not to the crop the images after rotation and instead it is better to increase their size by suitable zero-padding. Finally, the rotated images are reshaped to form vectors while creating the columns of $\boldsymbol{D_3}$.

(c) Class $\mathcal{S}_3$ which consists of images obtained by applying an intensity transformation $I^i_{new}(x, y) = \alpha(I^i_{old}(x, y))^2 + \beta(I^i_{old}(x, y)) + \gamma$ to the images in $\mathcal{S}$, where α, β, γ are known. In this case we see that $\text{vec}(I^i_{new}) = \alpha(\boldsymbol{D\theta}).^2 + \beta\boldsymbol{D\theta} + \gamma = \alpha(\sum_{k=1}^{K} \boldsymbol{d_k}\theta_k) \cdot (\sum_{l=1}^{K} \boldsymbol{d_l}\theta_l) + \beta \sum_{k=1}^{K} \boldsymbol{d_k}\theta_k + \gamma\boldsymbol{1}$. Therefore, $\text{vec}(I^i_{new})$ can be expressed as a sparse linear combination of the columns of a dictionary $\boldsymbol{D_3}$ which contains the columns of $\boldsymbol{D}$, element-wise products of all pairs of columns of $\boldsymbol{D}$ (including pairs with identical members), and a column vector containing all ones.

(d) Consider image $\boldsymbol{f} \in \mathcal{S}$, we have $\boldsymbol{f} = \boldsymbol{D\theta_f}$. If b is a blur filter kernel, we have $b * \boldsymbol{f} = \boldsymbol{D_4\theta_f}$, where $\boldsymbol{D_4}$ is obtained by applying the blur filter b to every column of $\boldsymbol{D}$.

(e) Class $\mathcal{S}_5$ which consists of images obtained by applying a blur kernel which is known to be a linear combination of blur kernels belonging to a known set $\mathcal{B}$, to the images in $\mathcal{S}$. Lete $L = |\mathcal{B}|$. Here we have
$\text{vec}(I_{5,i}) = \text{vec}[(\sum_{l=1}^{L} b_l) * I_i] = \text{vec}[(\sum_{l=1}^{L} b_l) * \text{reshape}(\boldsymbol{D\theta})] = \text{vec}[(\sum_{l=1}^{L} b_l) * \text{reshape}(\sum_{k=1}^{K} \boldsymbol{d_k}\theta_k)]$
$= \text{vec}[(\sum_{l=1}^{L} \sum_{k=1}^{K} b_l * \text{reshape}(\boldsymbol{d_k})\theta_k]$. The new dictionary $\boldsymbol{D_5}$ is created by convolving a reshaped version of each column vector of $\boldsymbol{D}$ with every blur kernel belong to $\mathcal{B}$.

(f) Class $\mathcal{S}_6$ which consists of 1D signals obtained by applying a Radon transform in a known angle θ to the images in $\mathcal{S}$. The new dictionary $\boldsymbol{D_6}$ is obtained as follows: $\boldsymbol{d_{6,i}} = \mathcal{R}_\theta(\text{reshape}(\boldsymbol{d_i}))$ where 'reshape' converts a column vector to a 2D image.

(g) Class $\mathcal{S}_7$ which consists of images obtained by translating a subset of the images in class $\mathcal{S}$ by a known fixed offset (x_1, y_1), and the other subset by another known fixed offset (x_2, y_2). Assume appropriate zero-padding and increase in the size of the image canvas owing to the translation. In this case, the dictionary $\boldsymbol{D_7}$ is created by (i) reshaping the column vectors of $\boldsymbol{D}$ to form images, (ii) translating each of them by (x_1, y_1) to create new zero-padded images and repeating this for (x_2, y_2), (iii) reshaping all these zero-padded images to form column vectors. The resultant dictionary will thus have twice the number of columns as $\boldsymbol{D}$.

3. How will you solve for the minimum of the following objective functions: (1) $J(\boldsymbol{A_r}) = \|\boldsymbol{A} - \boldsymbol{A_r}\|^2_F$, where $\boldsymbol{A}$ is a known $m \times n$ matrix of rank greater than r, and $\boldsymbol{A_r}$ is a rank-r matrix, where $r < m, r < n$. (2) $J(\boldsymbol{R}) = \|\boldsymbol{A} - \boldsymbol{RB}\|^2_F$, where $\boldsymbol{A} \in \mathbb{R}^{n \times m}, \boldsymbol{B} \in \mathbb{R}^{n \times m}, \boldsymbol{R} \in \mathbb{R}^{n \times n}, m > n$ and $\boldsymbol{R}$ is constrained to be orthonormal. Note that $\boldsymbol{A}$ and $\boldsymbol{B}$ are both known.
In both cases, explain briefly any one situation in image processing where the solution to such an optimization problem is required. [5+5+5+5=20 points]
Solution: First part: by Eckart-Young theorem, you need to compute the SVD of $\boldsymbol{A} = \boldsymbol{USV}^T$. Then consider the r largest singular values and their corresponding singular vectors from $\boldsymbol{U}$ and $\boldsymbol{V}$ - respectively named as $\boldsymbol{U_r}, \boldsymbol{V_r}$. Then $\boldsymbol{A_r} = \boldsymbol{U_r S_r}(\boldsymbol{V_r})^T$.
Application of first part: A rank-one approximation to the error matrix is computed in KSVD - see slides 110 and 111 of the lecture slides on Dictionary Learning. In PCA, we consider a rank-k approximation to the covariance matrix effectively. In orthographic structure from motion, a rank 3 approximation to the matrix of data-points across the frames is considered. (any one application is enough, but some more details of the matrix and its dimensions need to be mentioned in the answer.)

Second part: This is the orthogonal procrustes problem. The solution is obtained from the SVD of $\boldsymbol{BA}^T$. If $\boldsymbol{BA}^T = \boldsymbol{USV}^T$, then $\boldsymbol{R} = \boldsymbol{VU}^T$.
Application of second part: If $n = 3$, this is useful in determining a rotation/reflection transformation between two sets of points (assuming $m > 3$). We saw this in tomography (where exactly?). Another application is in the dictionary learning algorithm called union of orthonormal bases (any one application is enough, but some more details of the matrix and its dimensions need to be mentioned in the answer.)

4. We have studied the non-negative matrix factorization (NMF) technique in our course and examined applications in face recognition. I also described the application to hyperspectral unmixing. Your job is to find a research paper which explores an application of NMF in any task apart from these. You may look up the wikipedia article on this topic. Other interesting applications include stain normalization in pathology. Your job is to answer the following: (1) Mention the title, author list, venue and year of publication of the paper and include a link to it. (2) Which task does the paper apply NMF to? (3) How exactly does the paper

solve the problem using NMF? What is the significance of the dictionary and the dictionary coefficients in solving the problem at hand? [15 points]

Solution: I am consider the paper 'Structure-Preserving Color Normalization and Sparse Stain Separation for Histological Images' published in the IEEE Transactions on Medical Imaging in August 2016. Link: `https://ieeexplore.ieee.org/stamp/stamp.jsp?arnumber=7460968`.

The paper applies NMF for the task of stain normalization, i.e. normalizing the colored appearance of a histopathology image. Given an image $I^{3\times n}$ with n pixels, we first determine its relative optical density given by $V = \log\frac{I_o}{I}$ where I_o is illuminating light intensity. This is as per equations 1 to 3 of the paper, using the Beer-Lambert law. Then, we factorize V into the product $V = WH$ where W is the $3\times r$ matrix of stains (non-negative) and H is the $r\times n$ matrix of proportions of each stain in a given pixel. The cost function in equation 5 is used for the factorization with imposition of additional constraints such as unit-norm constraint on every column of W. Given an image s whose appearance has to be normalized w.r.t. target image t, we factorize their density maps V_s and V_t into $V_s = W_sH_s$ and $V_t = W_tH_t$ using the proposed factorization scheme from equation 5. The normalized version of s is obtained as $V_s^{norm} = W_tH_s^{norm}$ where H_s^{norm} is computed from equation 8. See equations 8, 9, 10 for more mathematical details of the normalization.

5. In parallel bean computed tomography, the projection measurements are represented as a single vector $\boldsymbol{y} \sim \text{Poisson}(I_o\exp(-\boldsymbol{Rf}))$, where $\boldsymbol{y}\in\mathbb{R}^m$ with m = number of projection angles × number of bins per angle; I_o is the power of the incident X-Ray beam; $\boldsymbol{R}$ represents the Radon operator (effectively a $m\times n$ matrix) that computes the projections at the pre-specified known projection angles; and $\boldsymbol{f}$ represents the unknown signal (actually tissue density values) in $\mathbb{R}^n$. If $m < n$, write down a suitable objective function whose minimum would be a good estimate of $\boldsymbol{f}$ given $\boldsymbol{y}$ and $\boldsymbol{R}$ and which accounts for the Poisson noise in $\boldsymbol{y}$. State the motivation for each term in the objective function. Recall that if $z\sim\text{Poisson}(\lambda)$, then $P(z=k) = \lambda^k e^{-\lambda}/k!$ where k is a non-negative integer. Now suppose that apart from Poisson noise, there was also iid additive Gaussian noise with mean 0 and known standard deviation σ, in $\boldsymbol{y}$. How would you solve this problem (eg: appropriate preprocessing or suitable change of objective function)? [6+ 4 = 10 points]

   Solution: We have $y_i \sim \text{Poisson}(I_o\exp(-\boldsymbol{R^i f}))$. The cost function to solve the problem of determining $\boldsymbol{f}$ from $\boldsymbol{y}, I_o, R$ is given as:

$$J(\boldsymbol{f};\boldsymbol{y}) = \sum_{i=1}^{m}\left(I_o e^{-\boldsymbol{R^i f}} - y_i\log I_o + y_i\boldsymbol{R^i f}\right) + \rho\|\boldsymbol{\Psi}^t\boldsymbol{f}\|_1 \text{ s. t. } \boldsymbol{f}\succeq \boldsymbol{0}, \tag{1}$$

where the $y_i\log I_o$ term can be dropped off as it remains unaffected by $\boldsymbol{f}$, and $\boldsymbol{f}\succeq\boldsymbol{0}$ represents the constraint that $\boldsymbol{f}$ is elementwise non-negative. Here the first term is the data fidelity term which accounts for the Poisson noise in the measurement vector $\boldsymbol{y}$ (it is the negative log-likelihood of the Poisson distribution of y_i given mean $I_o\exp(-\boldsymbol{R^i f})$), and the second one is a regularizer (with regularization parameter ρ) which imposes sparsity of $\boldsymbol{f}$ in the basis $\boldsymbol{\Psi}$.

Another equivalent way of expressing the same cost function using $\boldsymbol{f} = \boldsymbol{\Psi\theta}$ is:

$$J(\boldsymbol{\theta};\boldsymbol{y}) = \sum_{i=1}^{m}\left(I_o e^{-\boldsymbol{R^i\Psi\theta}} - y_i\log I_o + y_i\boldsymbol{R^i\Psi\theta}\right) + \rho\|\boldsymbol{\theta}\|_1 \text{ s. t. } \boldsymbol{\Psi\theta}\succeq\boldsymbol{0}. \tag{2}$$

If the noise in y_i is Gaussian in addition to the Poisson noise, then we have $y_i \sim \text{Poisson}(I_o\exp(-\boldsymbol{R^i f})) + \eta_i$ where $\eta_i \sim \mathcal{N}(0,\sigma^2)$. Thus the noise in y_i is obtained from the addition of Poisson as well as Gaussian random variables. The PDF of the sum of two random variables is obtained from the convolution of their individual PDFs. Hence, denoting $k_i \triangleq I_o\exp(-\boldsymbol{R^i f})$, we have $p(y_i|k_i) = \sum_{n=0}^{\infty}\dfrac{e^{-k_i}k_i^n}{n!}\dfrac{e^{-(y_i-n)^2/(2\sigma^2)}}{\sigma\sqrt{2\pi}}$ which is the convolution of a Poisson PMF and a Gaussian PDF. In this case, the likelihood function of all the y_i values (assuming statistical independence given $I_oe^{-\boldsymbol{Rf}}$) will be given by $\Pi_{i=1}^m p(y_i|k_i)$. The overall cost function to be minimized in this case is given as:

$$J(\boldsymbol{f};\boldsymbol{y}) = \sum_{i=1}^{m}\log\Big(\sum_{n=0}^{\infty}\frac{e^{-I_o\exp(-\boldsymbol{R^i f})}(I_o)^n\exp(-n\boldsymbol{R^i f})}{n!}\Big) + \rho\|\boldsymbol{\Psi}^t\boldsymbol{f}\|_1 \text{ s. t. } \boldsymbol{f}\succeq\boldsymbol{0}. \tag{3}$$

In practice, the infinite series needs to be truncated for computational efficiency. (No points to be deducted for not mentioning this.)