Assignment 3: CS 754, Advanced Image Processing

Due: 22nd March 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: A3-IdNumberOfFirstStudent-IdNumberOfSecondStudent.zip. (If you are doing the assignment alone, the name of the zip file is A3-IdNumber.zip). Upload the file on moodle BEFORE 11:55 pm on 22nd March. No assignments will be accepted after a cutoff deadline of 10 am on 23rd March. 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. Your task here is to implement the ISTA algorithm for the following three cases:

   (a) Consider the image from the homework folder. Add iid Gaussian noise of mean 0 and variance 3 (on a [0,255] scale) to it, using the 'randn' function in MATLAB. Thus $\boldsymbol{y} = \boldsymbol{x} + \boldsymbol{\eta}$ where $\boldsymbol{\eta} \sim \mathcal{N}(0, 3)$ (earlier the variance was mistakenly marked as 4). You should obtain $\boldsymbol{x}$ from $\boldsymbol{y}$ using the fact that patches from $\boldsymbol{x}$ have a sparse or near-sparse representation in the 2D-DCT basis.

   (b) Divide the image shared in the homework folder into patches of size 8×8. Let $\boldsymbol{x_i}$ be the vectorized version of the i^{th} patch. Consider the measurement $\boldsymbol{y_i} = \boldsymbol{\Phi x_i}$ where $\boldsymbol{\Phi}$ is a 32×64 matrix with entries drawn iid from $\mathcal{N}(0, 1)$. Note that $\boldsymbol{x_i}$ has a near-sparse representation in the 2D-DCT basis $\boldsymbol{U}$ which is computed in MATLAB as 'kron(dctmtx(8)',dctmtx(8)')'. In other words, $\boldsymbol{x_i} = \boldsymbol{U\theta_i}$ where $\boldsymbol{\theta_i}$ is a near-sparse vector. Your job is to reconstruct each $\boldsymbol{x_i}$ given $\boldsymbol{y_i}$ and $\boldsymbol{\Phi}$ using ISTA. Then you should reconstruct the image by averaging the overlapping patches. You should choose the α parameter in the ISTA algorithm judiciously. Choose $\lambda = 1$ (for a [0,255] image). Display the reconstructed image in your report. State the RMSE given as $\|X(:) - \hat{X}(:)\|_2 / \|X(:)\|_2$ where $\hat{X}$ is the reconstructed image and X is the true image. [15 points]

   Solution: See sample code in the homework folder.

2. Download the book 'Statistical Learning with Sparsity: The Lasso and Generalizations' from `https://web.stanford.edu/~hastie/StatLearnSparsity_files/SLS_corrected_1.4.16.pdf`, which is the website of one of the authors. (The book can be officially downloaded from this online source). Your task is to trace through the steps of the proof of Theorem 11.1(b). This theorem essentially derives error bounds on the minimum of the following objective function: $J(\boldsymbol{\beta}) = \frac{1}{2N}\|\boldsymbol{y} - \boldsymbol{X\beta}\|^2 + \lambda_N \|\boldsymbol{\beta}\|_1$ where λ_N is a regularization parameter, $\boldsymbol{\beta} \in \mathbb{R}^p$ is the unknown sparse signal, $\boldsymbol{y} = \boldsymbol{X\beta} + \boldsymbol{w}$ is a measurement vector with N values, $\boldsymbol{w}$ is a zero-mean i.i.d. Gaussian noise vector whose each element has standard deviation σ and $\boldsymbol{X} \in \mathbb{R}^{N \times p}$ is a sensing matrix whose every column is unit normalized. This particular estimator (i.e. minimizer of $J(\boldsymbol{x})$ for $\boldsymbol{x}$) is called the LASSO in the statistics literature. The theorem derives a statistical bound on λ also. Your task is split up in the following manner:

   (a) Define the restricted eigenvalue condition (the answer's there in the book and you are allowed to read it, but you also need to understand it).

Answer: Refer equation 11.10 of the book.

(b) Starting from equation 11.20 on page 309 - explain why $G(\hat{v}) \le G(0)$.
Answer: The true signal is β^* and the estimate $\hat{\beta}$ is the minimum of $G(\hat{\nu})$ where $\hat{\nu} = \hat{\beta} - \beta^*$. Since $\hat{\beta}$ is the minimum, we clearly must have $G(\hat{\nu}) = \frac{1}{2N}\|\boldsymbol{y} - \boldsymbol{X}\hat{\boldsymbol{\beta}}\|^2 + \lambda_N\|\hat{\boldsymbol{\beta}}\|_1 \le \frac{1}{2N}\|\boldsymbol{y} - \boldsymbol{X}\boldsymbol{\beta^*}\|^2 + \lambda_N\|\boldsymbol{\beta^*}\|_1 = G(0)$.

(c) Do the algebra to obtain equation 11.21.
Answer: We have $y = X\beta^* + w$. Hence from $G(\hat{v}) \le G(0)$, we have $\frac{1}{2N}\|w - X\hat{\nu}\|^2 + \lambda_N\|\beta^* + \hat{\nu}\|_1 \le \frac{1}{2N}\|w\|^2 + \lambda_N\|\beta^*\|_1$. Opening out the brackets, $\frac{1}{2N}\|X\hat{\nu}\|^2 \le \frac{1}{N}w^t X\hat{\nu} + \lambda_N(\|\beta^*\|_1 - \|\beta^* + \hat{\nu}\|_1)$ which is eqn. 11.21.

(d) Do the algebra in more detail to obtain equation 11.22 (state the exact method of application of Holder's inequality - check the wiki article on it, if you want to find out what this inequality states).
Answer: We use the reverse triangle inequality to prove that $\|\beta^* + \hat{\nu}\|_1 \ge \|\beta^*_S\|_1 - \|\hat{\nu}_S\|_1 + \|\hat{\nu}_{S_c}\|_1$. Plugging this into Eqn. 11.21, we get $\frac{1}{2N}\|X\hat{\nu}\|^2 \le \frac{w^t X\hat{\nu}}{N} + \lambda_N(\|\hat{\nu}_S\|_1 - \|\hat{\nu}_{S_c}\|_1) \le \frac{\|X^T w\|_\infty \|\hat{\nu}\|_1}{N} + \lambda_N(\|\hat{\nu}_S\|_1 - \|\hat{\nu}_{S_c}\|_1)$. The last step follows Holder's inequality which states that for vectors a, b, we have $a^t b \le \|a\|_p \|b\|_q$ where $p, q \in [1, \infty]$ and $1/p + 1/q = 1$. here we have $p = \infty, q = 1$.

(e) Derive equation 11.23.
Answer: This is straightforward, starting from Eqn. 11.22.

(f) Assuming Lemma 11.1 is true and now that you have derived equation 11.23, complete the proof for the final error bound for equation 11.14b.
Answer: Eqn. 11.23 yields $\|X\hat{\nu}\|^2/(2N) \le \frac{3\sqrt{k}\lambda_N\|\hat{\nu}\|_2}{2}$. But assuming that Lemma 11.1 is true, the restricted eigenvalue condition states that $\|X\hat{\nu}\|^2/(N) \ge \gamma\|\hat{\nu}\|^2$. Combining this with the earlier equation yields the correct bounds.

(g) In which part of the proof does the bound $\lambda_N \ge 2\frac{\|\boldsymbol{X}^T\boldsymbol{w}\|_\infty}{N}$ show up? Explain.
Answer: In two place. The first is when going from Eqn. 11.22 to Eqn. 11.23. This essentially tells us how to choose the regularization parameter in terms of the noise vector w. The second is in proving Lemma 11.1, which proves that the solution to Lasso obeys the cone constraint, needed to apply the RE condition.

(h) Why is the cone constraint required? You may read the rest of the chapter to find the answer.
Answer: Strong convexity is required for uniqueness of the solution of the given problem. Strong convexity is not possible in the given setting because the matrix $X^T X$ is low rank since $N < p$. Hence, we consider a restricted version of strong convexity, which is restricted to vectors that lie in some constraint set $\mathcal{C}$. It turns out that solutions to the Lagrangian Lasso problem obey a cone constraint of the form $\|\hat{\nu}_{S_c}\|_1 \le 3\|\hat{\nu}_S\|_1$ where $\hat{\nu} \triangleq \hat{\beta} - \beta^*$. In fact, the proof we just did, has an important step that proves this cone constraint. It is called a cone constraint because the condition gives you the equation of a cone. The set S is the support of β^* which is assumed to be sparse. Thus, we are restricting our requirement of strong convexity not to arbitrary vectors, but to vectors $\hat{\nu}$ that lie inside this cone.

(i) Read example 11.1 which tells you how to put a tail bound on λ_N assuming that the noise vector $\boldsymbol{w}$ is zero-mean Gaussian with standard deviation σ. Given this, state the advantages of this theorem over Theorem 3 that we did in class. You may read parts of the rest of the chapter to answer this question.

What are the advantages of Theorem 3 over this particular theorem?

Answer: The advantage of this theorem over that of Theorem 3 is that the error here in the case of noise from $\mathcal{N}(0, \sigma^2)$ is upper bounded by $\mathcal{O}(\sigma\sqrt{k \log p})$, whereas for Theorem 3 it was upper bounded by $\mathcal{O}(\sigma\sqrt{N})$. Therefore this theorem effectively predicts a tighter upper bound for sparser signals, something which was missing in theorem 3. Also, the upper bound on the error does not scale with the number of measurements in this new theorem. The advantage of Theorem 3 over this one is that Theorem 3 handles the case of signals that are not exactly sparse. However this new theorem does have an extension to handle the compressible case as seen in Eqn. 11.24 of the book. Moreover, this theorem gives bounds that are minimax optimal, i.e. there is no other algorithm which can yield a substantially better error bound.

(j) Now read Theorem 1.10 till corollary 1.2 and comments on it concerning an estimator called the 'Dantzig selector', in the tutorial 'Introduction to Compressed Sensing' by Davenport, Duarte, Eldar and Kuttyniok. You can find it here: `http://www.ecs.umass.edu/~mduarte/images/IntroCS.pdf` or at `https://webee.technion.ac.il/Sites/People/YoninaEldar/files/ddek.pdf`. What is the common thread between the bounds on the 'Dantzig selector' and the LASSO?

Answer: The common thread is that bounds for both estimators scale as $\mathcal{O}(\sigma\sqrt{k \log p})$. Morover, both are proved to be minimax estimators, i.e. you cannot invent another estimator whose bounds are substantially better than these bounds (apart from constant factors). The part on minimax estimators is not required for the grade for this sub-question. [2 x 8 + 4 + 4 = 24 points]

3. In this task, you will you use the well-known package L1_LS from `https://stanford.edu/~boyd/l1_ls/`. This package is often used for compressed sensing solution, but here you will use it for the purpose of tomographic reconstruction. The homework folder contains images of two slices taken from an MR volume of the brain. Create measurements by parallel beam tomographic projections at any 18 randomly angles chosen from a uniform distribution on $[0, \pi)$. Use the MATLAB function 'radon' for this purpose. Now perform tomographic reconstruction using the following method: (a) filtered back-projection using the Ram-Lak filter, as implemented in the 'iradon' function in MATLAB, (b) independent CS-based reconstruction for each slice by solving an optimization problem of the form $J(\boldsymbol{x}) = \|\boldsymbol{y} - \boldsymbol{A}\boldsymbol{x}\|^2 + \lambda\|\boldsymbol{x}\|_1$, (c) a coupled CS-based reconstruction that takes into account the similarity of the two slices using the model given in the lectures notes on tomography. For parts (b) and (c), use the aforementioned package from Stanford. For part (c), make sure you use a different random set of 18 angles for each of the two slices. The tricky part is careful creation of the forward model matrix $\boldsymbol{A}$ or a function handle representing that matrix, as well as the corresponding adjoint operator $\boldsymbol{A}^T$. Use the 2D-DCT basis for the image representation. Modify the objective function from the lecture notes for the case of three similar slices. Carefully define all terms in the equation but do not re-implement it. For ease of implementation, use square images. For this zero-pad the original images to make them square-shaped before getting the radon projections. You can also specify the output size in the iradon function. You may work with uniformly spaced angles instead of randomly generated angles as the former can give better results. [3+7+8+7 = 25 points]

Answer: The operator A acting on θ is basically RU where U is the 2D-DCT basis, i.e. it computes the Radon of the inverse 2D-DCT of θ. The adjoint operator A^T acting on the projection vector y basically computes the 2D-DCT of the inverse Radon (with a Ram-Lak filter as stated here, though this filter is not really required) of y. The modified function for three slices is as follows:

$J(\boldsymbol{\beta_1}, \boldsymbol{\Delta\beta_{12}}, \boldsymbol{\Delta\beta_{13}}) = \|\boldsymbol{y_1} - \boldsymbol{R_1 U \beta_1}\|^2 + \|\boldsymbol{y_2} - \boldsymbol{R_2 U}(\boldsymbol{\beta_1} + \boldsymbol{\Delta\beta_{12}})\|^2 + \|\boldsymbol{y_3} - \boldsymbol{R_3 U}(\boldsymbol{\beta_1} + \boldsymbol{\Delta\beta_{13}})\|^2 + \lambda_1\|\boldsymbol{\beta_1}\|_1 + \lambda_2\|\boldsymbol{\Delta\beta_{12}}\|_1 + \lambda_3\|\boldsymbol{\Delta\beta_{13}}\|_1$.

Another modified form, which is also correct, is: $J(\boldsymbol{\beta_1}, \boldsymbol{\Delta\beta_{12}}, \boldsymbol{\Delta\beta_{23}}) = \|\boldsymbol{y_1} - \boldsymbol{R_1 U \beta_1}\|^2 + \|\boldsymbol{y_2} - \boldsymbol{R_2 U}(\boldsymbol{\beta_1} + \boldsymbol{\Delta\beta_{12}})\|^2 + \|\boldsymbol{y_3} - \boldsymbol{R_3 U}(\boldsymbol{\beta_1} + \boldsymbol{\Delta\beta_{12}} + \boldsymbol{\Delta\beta_{23}})\|^2 + \lambda_1\|\boldsymbol{\beta_1}\|_1 + \lambda_2\|\boldsymbol{\Delta\beta_{12}}\|_1 + \lambda_3\|\boldsymbol{\Delta\beta_{23}}\|_1$.

In general, better results are expected with the coupling effects as compared to without. This requires careful selection of regularization parameters. If the student has presented results with more than 18 angles, that is fine too.

4. Here is our Google search question again. You know of the applications of tomography in medicine (CT scanning) and virology/structural biology. Your job is to search for a journal paper from any other field

which requires the use of tomographic reconstruction (examples: seismology, agriculture, gemology). State the title, venue and year of publication of the paper. State the mathematical problem defined in the paper. Take care to explain the meaning of all key terms clearly. State the method of optimization that the paper uses to solve the problem. [16 points]

Solution: A very nice (and unexpected) application of tomography is in non-parametric probability density estimation, as seen in the paper 'Multi-dimensional density estimation by tomography' published in the Journal of the Royal Statistical Society (B) in 1993. Let $X = (X_1, X_2)$ be a two-dimensional random variable. Instead of estimating the density of X directly, the paper estimates the density of various quantities of 1D quantities of the form $\phi_1 X_1 + \phi_2 X_2$ where $\phi_1^2 + \phi_2^2 = 1$ and ϕ_1, ϕ_2 are chosen randomly from a unit circle. It turns out (see Theorem 1) that the density of these 1D quantities is equal to the Radon transform of the original 2D density taken in the direction (ϕ_1, ϕ_2). Thus given such 1D densities, the 2D density is computed by filtered backprojection with a Ram-Lak filter (equation 18). **Marking scheme:** There are many applications of tomography in agriculture, food science, seismology, minerology. One example in agriculture is `https://plantmethods.biomedcentral.com/articles/10.1186/s13007-019-0468-y`. Many of these papers do not define the mathematical problem. In such cases, merely stating that so and so algorithm was used for reconstruction is enough.

5. Let $R_\theta(f)$ be the Radon transform of the image $f(x,y)$ in the direction given by θ. Derive a formula for the Radon transform of the scaled image $f(ax, ay)$ where $a \neq 0$ is a scalar. [10 points]
   Solution: We know that $R_\theta[f(x,y)](\rho) = \int \delta(\rho - x\cos\theta - y\sin\theta) f(x,y)dxdy$. Now, we have
   $R_\theta[f(ax,ay)](\rho) = \int \delta(\rho - x\cos\theta - y\sin\theta) f(ax,ay)dxdy$. Putting in $ax = x_1, ay = y_1$, we have $dx_1 = adx, dy_1 = ady$. This gives us
   $LHS = R_\theta[f(ax,ay)](\rho) = \frac{1}{a^2}\int \delta(\rho - \frac{x_1}{a}\cos\theta - \frac{y_1}{a}\sin\theta) f(x_1,y_1)dx_1dy_1 =$
   $\frac{1}{a^2}\int \delta(\frac{a\rho - x_1\cos\theta - y_1\sin\theta}{a}) f(x_1,y_1)dx_1dy_1 = \frac{|a|}{a^2}\int \delta(\frac{a\rho - x_1\cos\theta - y_1\sin\theta}{a}) f(x_1,y_1)dx_1dy_1 = \frac{1}{|a|} R_\theta[f(x,y)](a\rho)$. The second last step involves the scaling property of the Dirac delta function as follows: $\delta(\alpha x) = \frac{\delta(x)}{|\alpha|}$ or equivalently $\delta(x/\alpha) = |\alpha|\delta(x)$. See `https://en.wikipedia.org/wiki/Dirac_delta_function`, in particular under the sub-heading 'scaling'.

6. Derive the Radon transform of the unit impulse $\delta(x,y)$ and the shifted unit impulse $\delta(x - x_0, y - y_0)$. [10 points]
   Solution: Let us consider the shifted impulse for which $R_\theta(f)(\rho) = \int \delta(\rho - x\cos\theta - y\sin\theta)\delta(x - x_0, y - y_0)dxdy = \delta(\rho - x_0\cos\theta - y_0\sin\theta)$. If $x_0 = y_0 = 0$, we get just $\delta(\rho)$.