School of Computing and Information Systems
MAST30034: Applied Data Science
Assignment 1
Due date: No later than 11:59pm on Monday 30^{th} August 2021
Weight: 15%

Assignment Overview

Spatiotemporal datasets consists of both space and time dimensions and represent many real world applications such as medical imaging data, video data, weather patterns and so on. The aim of this assignment is to familiarize ourselves with dataset generation by time samples and pixel values, its preprocessing by statistical measures, its analysis by parameter estimation, its results visualization by graphs, and its evaluation by performance metrics. In this regard, you are free to choose any tools (R/Python/Matlab) to perform these tasks.

Assignment Details

In this assignment, you are going to attempt six main tasks given as follows:

1. Generate a spatiotemporal synthetic dataset (that follows a linear regression model),
2. Perform data preprocessing steps,
3. Apply least square regression and its variants on the generated (synthetic) dataset,
4. Visualize the generated dataset to address critical questions,
5. Select parameter by using mean square error (MSE) curve,
6. Use performance metrics to judge a better estimator.

Least Square Regression (LSR)

The least square estimator $\mathbf{A} = (\mathbf{D}^\top \mathbf{D})^{-1}\mathbf{D}^\top \mathbf{X}$ solves a linear regression problem of the type $\mathbf{X} = \mathbf{DA} + \mathbf{E}$ and in terms of cost function it can be written as

$$\min_{\mathbf{a}} ||\mathbf{x}_{\mathrm{v}} - \mathbf{D}\mathbf{a}_{\mathrm{v}}||^2, \quad \mathrm{v} = 1, ..., V \tag{1}$$

where $\mathbf{X} \in \mathbb{R}^{\mathrm{N}\times\mathrm{V}}$ is the observed dataset, $\mathbf{D} \in \mathbb{R}^{\mathrm{N}\times\mathrm{L}}$ contains the set of regressors, $\mathbf{A} \in \mathbb{R}^{\mathrm{L}\times\mathrm{V}}$ (known as coefficient matrix) measures the relationship between the observed responses and the regressors, $\mathbf{E}$ accounts for the model error, $\mathbf{x}_{\mathrm{v}}$ and $\mathbf{a}_{\mathrm{v}}$ are v-th column from $\mathbf{X}$ and $\mathbf{A}$, respectively, and $\mathbf{x}^{\mathrm{k}}$ is the k-th row. Also, N is the number of time points, L is the number of regressors, and V is the number of observed variables. Throughout this document capital bold letters represent matrices and small bold letters represent vectors.

Synthetic Dataset Generation

A spatiotemporal synthetic dataset $\mathbf{X}$ can be constructed using temporal sources called time courses **TC** and spatial sources called spatial maps **SM**. The model presented in eq. (1) can be used to generate a synthetic data where $\mathbf{D}$ becomes **TC**, and $\mathbf{A}$ becomes **SM**. On this

synthetic dataset $\mathbf{X}$ one can then perform LSR using regressors from $\mathbf{D}$ ($\mathbf{D} = \mathbf{TC}$ source TCs) to estimate response signal strength $\mathbf{A}$ (retrieved SMs), and then use $\mathbf{A}$ (retrieved SMs) to estimate $\mathbf{D}$ (retrieved TCs).
The source TCs and SMs can either be statistically dependent on each other or independent. For this assignment, a simple case is considered that avoids spatial dependence, however temporal correlation would be introduced to address the topic of multicollinearity (MC). In order to mitigate the problem of MC, variants of LSR that introduce regularization into their model making them more adaptable at handling MC can be used. Regularization allows to improve efficiency of the estimate (reduced variance or minimum mean square error) by introducing a bias into the model that shrinks the estimate of coefficients towards certain values (small values near zero).

Ridge Regression (RR)

An l-2 norm of $\mathbf{a}_{\mathrm{v}}$ in the objective function (1) is penalized by introducing a regularization term $\tilde{\lambda}$ as

$$\min_{\mathbf{a}} ||\mathbf{x}_{\mathrm{v}} - \mathbf{D}\mathbf{a}_{\mathrm{v}}||^2 + \tilde{\lambda} \sum_{\mathrm{l}=1}^{\mathrm{L}} ||\mathbf{a}_{\mathrm{v}}^{\mathrm{l}}||^2, \quad \mathrm{v} = 1, ..., V, \quad l = 1, ..., L \tag{2}$$

This results in a ridge regression estimate as $\mathbf{A} = (\mathbf{D}^\top \mathbf{D} + \tilde{\lambda}\mathbf{I})^{-1}\mathbf{D}^\top \mathbf{X}$. Here $\mathbf{I}$ is an identity matrix of size L $\times$ L, whereas penalty term $\tilde{\lambda}$ is a scalar value, which has no effect when its zero and RR produces LSR estimate, however as it reaches infinity its impact increases and the coefficients in $\mathbf{A}$ start getting close to zero. Generally, for our synthetic data case selecting λ between 0 and 1 would suffice only if we multiply whatever λ we choose between 0 and 1 by number of variables in the dataset, which are V in our case that is $\tilde{\lambda} = \lambda V$. We will use check and guess method for parameter selection (λ) in this case, but we will use sum of MSE versus regularization parameter plot in case of LASSO regression. Furthermore, RR will not produce a sparse solution as all coefficients are shrunk towards zero with the same factor so it fails to eliminate any of them.

LASSO Regression (LR)

LASSO regression on the other hand yields a sparse solution as it can eliminate less important coefficients by shrinking them towards zeros. Lets use this regression to further enhance the prediction accuracy for our case. Other than the inability of LSR to handle MC, the main reason behind the bad performance of LSR and RR (both of them producing many false positives while recovering coefficients) is that they incorporate the undesired (noise carrying) pixels into the estimate of $\mathbf{A}$ and this also effects the estimate of $\mathbf{D}$ in terms of overfitting. In order to mitigate the problem of both overfitting and multicollinearity, l-1 norm of $\mathbf{a}_{\mathrm{v}}$ in the objective function (1) is penalized by introducing a regularization term ρ as

$$\min_{\mathbf{a}} ||\mathbf{x}_{\mathrm{v}} - \mathbf{D}\mathbf{a}_{\mathrm{v}}||^2 + \rho \sum_{\mathrm{l}=1}^{\mathrm{L}} |\mathbf{a}_{\mathrm{v}}^{\mathrm{l}}|, \quad \mathrm{v} = 1, ..., V, \quad l = 1, ..., L \tag{3}$$

Luckily its closed form solution exists in soft thresholding as

$$s = \frac{1}{1.1||\mathbf{D}^\top \mathbf{D}||^2}, \quad \text{thr} = N\rho s$$
$$\mathbf{E} = \mathbf{a}_{(\mathbf{i})_\text{v}} + s(\mathbf{D}^\top(\mathbf{x}_\text{v} - \mathbf{D}\mathbf{a}_\text{v}))$$
$$\mathbf{a}_\text{v} = \frac{1}{1+\text{thr}}\Big[\text{sgn}\Big(\mathbf{E}\Big) \circ \Big(|\mathbf{E}| - \text{thr}\mathbf{1}_L\Big)_+\Big], \tag{4}$$

where i=1,...,I represents the i-th iteration of the algorithm, $0 < \rho < 1$ controls the thresholding of $\mathbf{a}_\text{v}$, sgn(.), |.|, $(\text{x})_+$, and $\circ$ define the component-wise sign, the component-wise absolute value, the component-wise max $(0, \text{x})$, and the Hadamard product, respectively. The R code to solve (4) is given below to answer Question 2.3 and 2.4.

R Code for LR

```
step <- 1/(norm(TC%*%t(TC))*1.1)
thr <- rho*N*step
Ao <- matrix(0,nsrcs,1)
A <- matrix(0,nsrcs,1)
Alr <- matrix(0,nsrcs,x1*x2)

for (k in 1:(x1*x2))
{
  A <- Ao+step*(t(TC)%*%(X[,k]-(TC%*%Ao)))
  A <- (1/(1+thr))*(sign(A)*pmax(replicate(nsrcs,0), abs(A) -thr))
  for (i in 1:10)
  {
    Ao <- A
    A <- Ao+step*(t(TC)%*%(X[,k]-(TC%*%Ao)))
    A <- (1/(1+thr))*(sign(A)*pmax(replicate(nsrcs,0), abs(A) -thr))
  }
  Alr[,k]<-A
}
```

Principal Component Regression (PCR)

The PCR, which also addresses the multicollinearity problem where instead of regressing the observed variable from $\mathbf{X}$ on the design matrix ($\mathbf{D}$) directly, the principal components (PC) of the design matrix are used as regressors. Only a subset of all the principal components are used for regression. We can apply PCA on the matrix of regressors $\mathbf{D}$ to obtain a new regressor matrix $\mathbf{Z}$ as

$$\min_{\mathbf{a}} ||\mathbf{x}_\text{v} - \mathbf{D}\mathbf{a}_\text{v}||^2 = \min_{\mathbf{b}} ||\mathbf{x}_\text{v} - \mathbf{Z}\mathbf{b}_\text{v}||^2, \quad v = 1, ..., V \tag{5}$$

In order to estimate PCs, you can use svd command in matlab, in python it is found in numpy library, and in R it is there in MASS library. For instance in Matlab

```
[U,V,W]=svds(X,5)
% here digit 5 points to the fact that we have only decomposed the data to first five PCs
% U is the eigen vector of XX' and it is where you obtain "Z" for your case
% W is the eigen vector of X'X,  V are the eigen values
```

You must note down that we did not take PCA of the source TCs in order to generate the dataset $\mathbf{X}$. We keep TCs intact for dataset generation and actually PCs of the TCs is going to be used as regressors in $\mathbf{Z}$ while estimating $\mathbf{B}$.

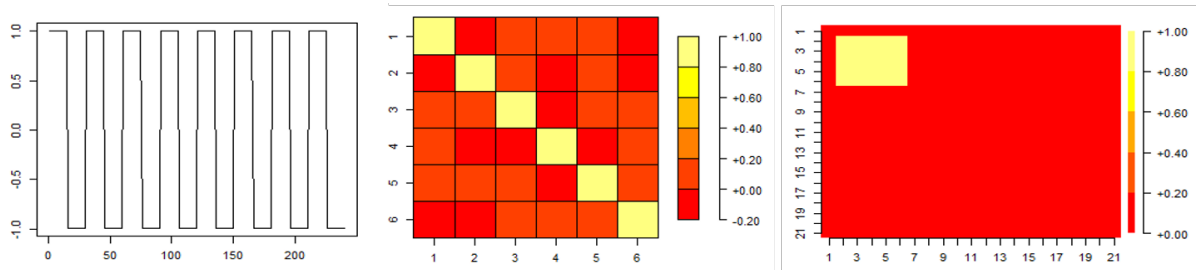

Figure 1: First TC source (left), random correlation matrix (middle), first SM source (right)

Assignment Questions

In both questions your marks carrying tasks are highlighted by italic text. For both questions and code given in LR section, value of N is 240, value of V is 441, x1=21 and x2=21 the size of each slice, value of ρ can be selected between 0 and 1, nsrcs is the number of sources =6, and $\mathbf{X}$ is the standardized generated dataset. For question 1.4, σ stands for standard deviation. For question 2, always apply absolute operation on retrieved coefficient values in $\mathbf{A}$ as this will make it easier to visualize its plots, and whatever the variant of LSR you encounter, you can always estimate $\mathbf{D}$ using $\mathbf{D} = \mathbf{X}\mathbf{A}^{\top}$

Question 1: Synthetic dataset generation, data preporcessing, & data visualization

1. Construct a matrix **TC** of size 240×6 consisting of six temporal sources using three vectors, 1) onsets arrival vector (AV) = [0,20,0,0,0,0], 2) increment vector (IV) = [30,45,60,40,40,40], and 3) duration of ones = [15,20,25,15,20,25]. For instance, you can generate first TC using these three vectors as AV:IV:N-20 =0:30:220 and ones stay active for 15 samples, and N here is 240. This TC is also shown in Figure 1 (left). Mean center each TC by subtracting its mean and standardize each TC by dividing it by its standard deviation. This will make TCs bias free (centered around the origin) and equally important (have unit variance). *Plot all TCs as six subplots. Why not normalize (divide by l-2 norm) the TCs instead of standardizing it?*

2. A randomly generated correlation matrix (CM) (illustrating uncorrelatedness among all variables) is shown as a sample in Figure 1 (middle). For your case, construct a CM that represents correlation values between 6 variables. *Show its plot, and can you tell visually which two TCs are highly correlated? If not, can you tell this from CM?*

3. Construct an array **tmpSM** of size $6 \times (21 \times 21)$ consisting of ones and zeros, by placing ones at these pixels along "vertical,horizontal" direction of the slice i) 02:06,02:06, ii) 02:06,15:19, iii) 08:13,02:06, iv) 08:13,15:19, v) 15:19,02:06, vi) 15:19,15:19. The first SM source is also shown in Figure 1 (right). *Plot these SMs in six subplots.* Reshape the array **tmpSM** into a two dimensional matrix and call it **SM** of size 6×441. *Using CM show if these 6 vectored SMs are independent? For our particular case, why standardization of SMs like TCs is not important?* Hint: Imagine one of the slice has pixel values of 5, while others remain at 1.

4. Generate zero mean white Gaussian noise for temporal and spatial sources denoted as $\mathbf{\Gamma}_t \in \mathbb{R}^{240\times 6}$ and $\mathbf{\Gamma}_s \in \mathbb{R}^{6\times 441}$. Besides their dimensions, another difference between spatial and temporal noise is the noise variance, which is 0.25 for $\mathbf{\Gamma}_t$, and 0.015 for $\mathbf{\Gamma}_s$. *Using a 6×6 CM for each noise type (spatial and temporal) can you show if they are correlated across sources? Also plot the histogram of both noise sources to see if they have a normal distribution? Does this normal distribution fulfils the mean and variance= 1.96σ criteria relating to 0.25, 0.015, and zero mean? Is there product $\mathbf{\Gamma}_t\mathbf{\Gamma}_s$ correlated across V number of variables?*

5. Generate a synthetic dataset $\mathbf{X}$ of size 240×441 as $\mathbf{X} = (\mathbf{TC}+\mathbf{\Gamma}_t)\times(\mathbf{SM}+\mathbf{\Gamma}_s)$. This builds a dataset that follows the model shown in eq (1). *Can these products $\mathbf{TC}\times\mathbf{\Gamma}_s$ and $\mathbf{\Gamma}_t\times\mathbf{SM}$ exist, If yes what happened to them because if we keep them then we cannot fit our model onto (1)? Plot atleast 100 randomly selected time-series from $\mathbf{X}$ as few of them are shown in Figure 2 (left). Also plot variance of all 441 variables on a separate plot. What information does this plot give you?* At the end standardize the dataset $\mathbf{X}$, because source TCs (regressors) are also standardized and so dataset should be too.

Question 2: Data analysis, results visualization, & performance metrics

1. The synthetic standardized dataset $\mathbf{X}$ that you have generated in Question 1 follows the linear regression model $\mathbf{X} = \mathbf{DA} + \mathbf{E}$ where the unknown $\mathbf{A}$ can be estimated using least squares. Since the set of regressors $\mathbf{D}$ are known as you have used them to generate $\mathbf{X}$, and these were the TCs so $\mathbf{D} = \mathbf{TC}$. Estimate $\mathbf{A}$ (retrieval of SMs) using least square solution $\mathbf{A}_{LSR} = (\mathbf{D}^\top\mathbf{D})^{-1}\mathbf{D}^\top\mathbf{X}$. Similarly, retrieve TCs in $\mathbf{D}_{LSR}$ for a known $\mathbf{A}_{LSR}$ using $\mathbf{D}_{LSR} = \mathbf{X}\mathbf{A}_{LSR}^\top$. *Plot six retrieved sources using $\mathbf{A}_{LSR}$ and $\mathbf{D}_{LSR}$ side by side as shown in Figure 2 (right) for one of the retrieved sources. Do a scatter plot between 3rd column of $\mathbf{D}_{LSR}$ and 30th column of standardized $\mathbf{X}$, you will find a linear relationship between them, why this does not exist between 4th column of $\mathbf{D}_{LSR}$ and same column of $\mathbf{X}$.* Hint: Look at the slices arrangement which source TCs do you think contributed in building 30th data element.

2. Estimate RR parameters $\mathbf{A}_{RR} = (\mathbf{D}^\top\mathbf{D} + \tilde{\lambda}\mathbf{I})^{-1}\mathbf{D}^\top\mathbf{X}$ and $\mathbf{D}_{RR}$ and then compare RR to LSR by estimating two correlation vectors retaining only maximum absolute correlations i) between each $\mathbf{TC}$ and $\mathbf{D}_{LSR}$ and store it in $\mathbf{c}_{TLSR}$, and ii) between each $\mathbf{TC}$ and $\mathbf{D}_{RR}$ and store it in $\mathbf{c}_{TRR}$. *Calculate the sum of these two correlation vectors.* If you have carefully selected the value of λ between 0 and 1 you must end up with $\sum\mathbf{c}_{TRR} > \sum\mathbf{c}_{TLSR}$, and remember $\tilde{\lambda} = \lambda V$. *Also, for $\lambda = 1000$, plot first vector from $\mathbf{A}_{RR}$ and the corresponding vector from $\mathbf{A}_{LSR}$, Do you find all values in $\mathbf{a}^1_{RR}$ shrinking towards zero?*

3. For 21 values of ρ selected between 0 and 1 with an interval of 0.05, estimate LR parameters $\mathbf{A}_{LR}$, $\mathbf{D}_{LR}$, and sum of MSE using LR parameters as $\sum_{\mathrm{v}=1}^{\mathrm{V}} ||\mathbf{X} - \mathbf{D}_{LR}\mathbf{A}^2_{LR}||_2^2/NV$, and repeat this process 10 times (10 realizations) each time with a new standardized $\mathbf{X}$ (new in terms of $\mathbf{\Gamma}_t$ and $\mathbf{\Gamma}_s$). *Then plot average of MSE over these 10 realizations*

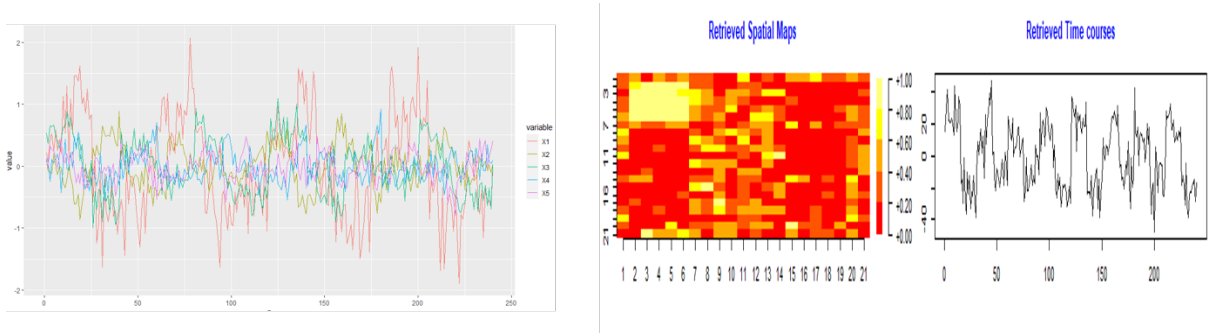

Figure 2: Few variables from $\mathbf{X}$ (left), first retrieved spatiotemporal source from $\mathbf{X}$ via SR (right)

against each value of ρ. At what value of ρ do you find the minimum MSE? Is it okay to select this value? At what value of ρ did MSE started to increase again (LR diverged) . Here $||.||_2$ is the Frobenius norm. Hint: Matlab code for MSE calculation over one realization and one lambda value, here rr=1, first realization, and kk=1, first lambda

$$\text{MSE(kk, rr)} = \text{sum(sum((Y–Dlr} * \text{Alr).^2))/(N} * \text{V)}$$

4. Estimate LR parameters for the value of ρ that you have selected in Question 2.3. Now, estimate four correlation vectors retaining only maximum absolute correlations i) between each **TC** and $\mathbf{D}_{RR}$ and store it in $\mathbf{c}_{TRR}$, ii) between each **SM** and $\mathbf{A}_{RR}$ and store it in $\mathbf{c}_{SRR}$, iii) between each **TC** and $\mathbf{D}_{LR}$ and store it in $\mathbf{c}_{TLR}$, and iv) between each **SM** and $\mathbf{A}_{LR}$ and store it in $\mathbf{c}_{SLR}$. *Calculate the sum of these four correlation vectors. If you have carefully selected the value of ρ you must end up with $\sum \mathbf{c}_{TLR} > \sum \mathbf{c}_{TRR}$ and $\sum \mathbf{c}_{SLR} > \sum \mathbf{c}_{SRR}$. Plot side by side in form of 4 columns estimates of $\mathbf{D}$ and $\mathbf{A}$ for both RR and LR to know the difference visually. You will see a major difference in estimates of $\mathbf{A}$ in terms of false positives. Can you mention the reason behind this difference?* 这个 题目还没画图

5. Estimate PCs of the TCs and *plot their eigen values. For which PC the eigen value is the smallest? Plot the regressors in $\mathbf{Z}$ and source TCs side by side. Did you notice deteriorated shape of PCs? Why the shape of TCs has been lost?* Now keeping all components in $\mathbf{Z}$ apply lasso regression on $\mathbf{X}$ using $\rho = 0.001$ and then *Plot the results of $\mathbf{D}_{PCR}$ and $\mathbf{A}_{PCR}$ side by side (note that $\mathbf{A}_{PCR} = \mathbf{B}$ and your regressors are in $\mathbf{Z}$ (PCs of the TCs)). Did you notice the inferior performance of PCR compared to the other three regression models? Why is that so?*