assignment 1

Huan Zhang
Student ID: 1173919

September 10, 2021

Please refer to the notebook ("1.ipynb") for all codes for plotting and calculations.

Q1

1.1

The 6 subplots are shown in Figure 1. Since the original TC only consists of 0 and 1 then if we normalize each column of TC (via dividing by the L2 norm) the normalised results will be really small, so there's a risk of experiencing the 'rounding' error from the computer which causes incorrect outputs for future calculations (e.g. estimating the parameters) or wrong results when comparing the data (because if the number is too small, the computer may just treat all of them as 0 when make comparison). Furthermore, standardise the data will help mitigating multicollinearity problem. By making all predictors on the same scale, this will also prevent too much or too little penalisation on some particular coefficients.

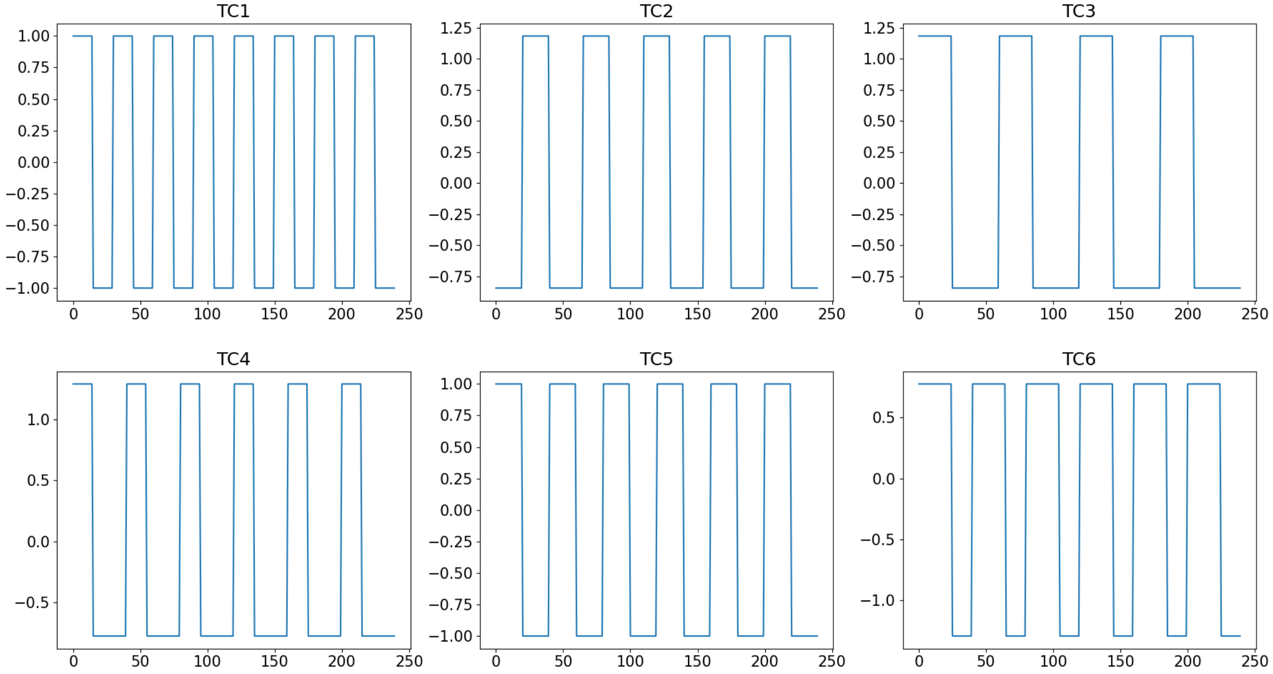

Figure 1

1.2

As shown in Figure 2, the correlation matrix(CM) for TCs are plotted via a heatmap. The CM is constructed by calculating the Pearson correlation coefficients between every two TC variables, as shown in the color map, the more red of the color, the more positively correlated between the two variables. Thus, we can find that 'TC5 & TC6'," TC4 & TC6" and "TC4 & TC5" are highly correlated, Rest of the variables have the correlation coefficient around 0 meaning that they are quite independent to each other.

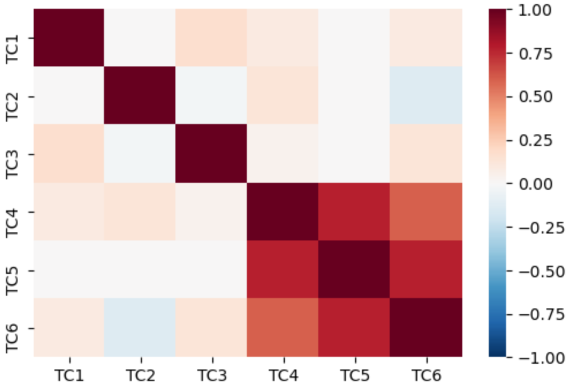

Figure 2

1.3

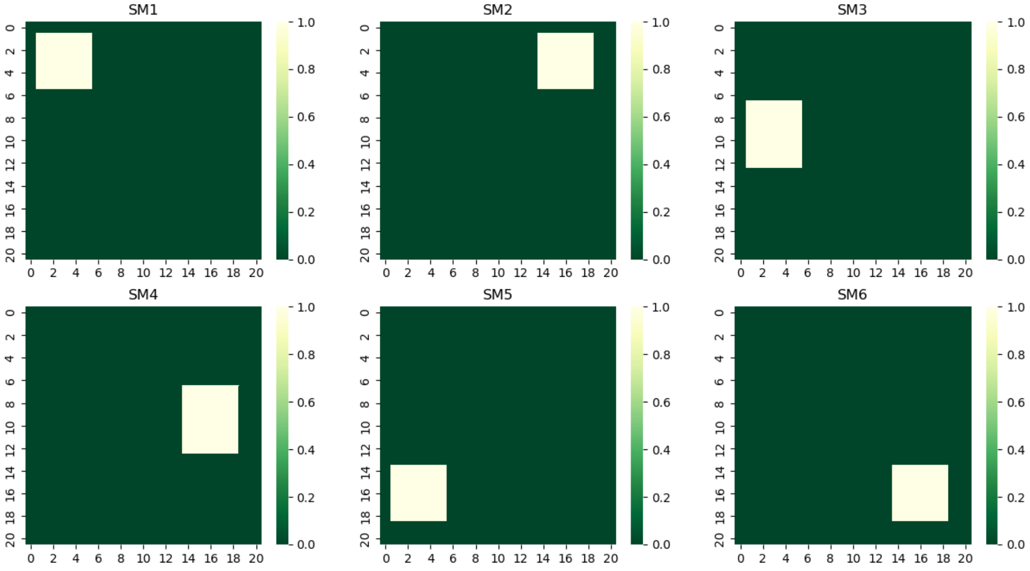

Figure 3

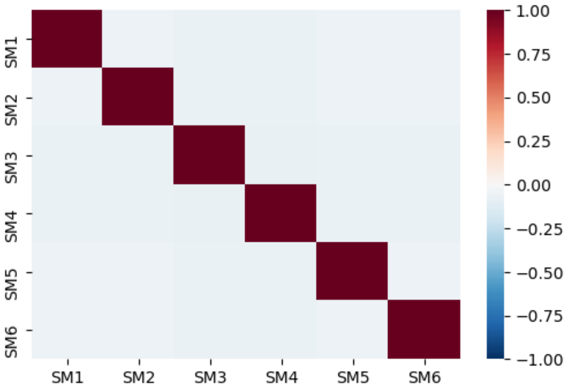

Figure 4

The six SMs are shown in Figure 3. From the visualisation of the CM for the 6 vectored SMs (Figure 4), we can see the correlation coefficient between every 2 different vectors are all around 0 (white/grey color) which means the 6 SMs are independent to each other.
The reasons to why we don't standardise SMs are:

- They are independent to each other so we don't need to care the multicollinearity problem on SMs (spatial dependence is avoided in our case)
- In our case, we don't care about how large the pixel values for a particular pixel because D = TC and we are using D to estimate A (SM in our case), so whatever what regression model we choose to fit, A(SMs) are just indicators to tell whether the spatial signals are activated at a particular pixel.
- Also, we don't need to concern if we overly penalise some parameters in SMs, because we only apply penalisation on D (TCs in our case)

1.4

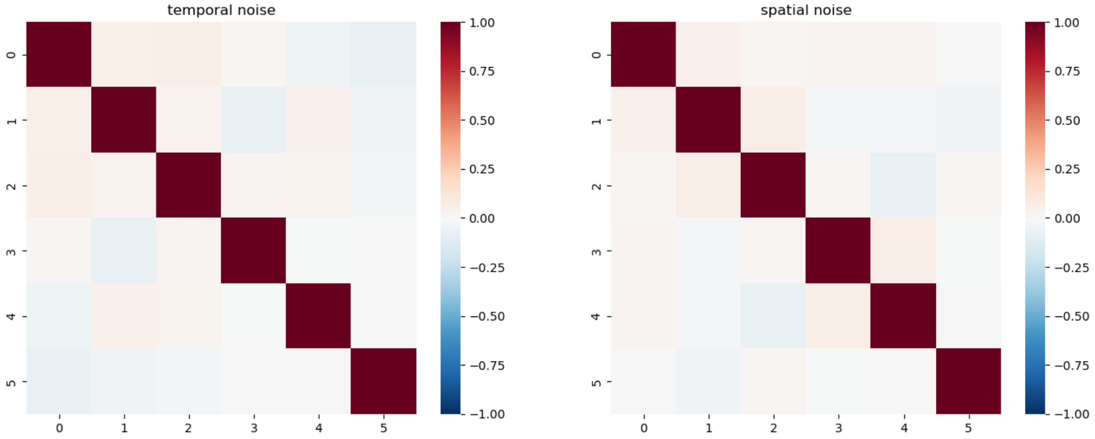

Figure 5: Left is CM for temporal noise and right is CM for spatial noise

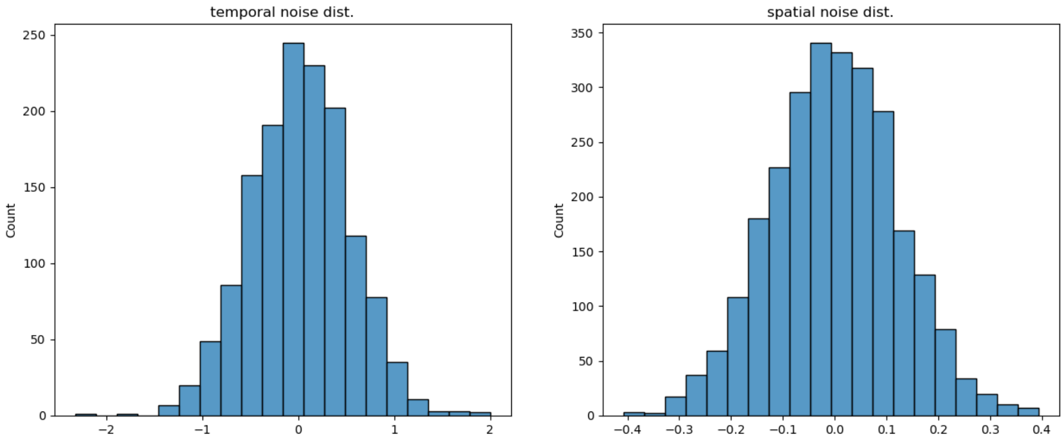

Figure 6: Left is temporal noise distribution and right is spatial noise

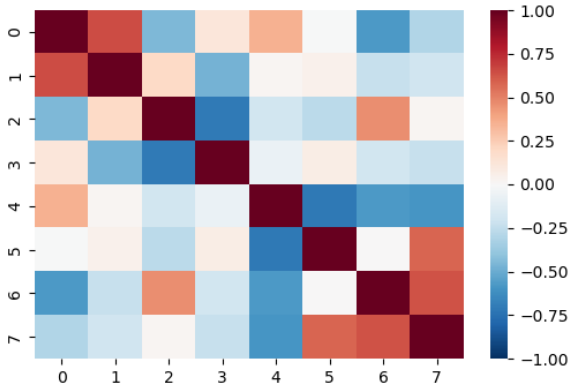

Figure 7: partial CM of the noise product

As shown in Figure 5, we can see most blocks in both plots have very light color which suggests that for both noise types, each noise are not correlated across their sources.
From the histograms in Figure 6, we can see both distribution fit quite well with the Gaussian distribution, and they centered at 0 shows that they fulfil mean 0, and if we calculate the cumulative density i.e. $\mathrm{P}(\mathrm{X} < 1.96 \times \sqrt{0.25})$ and $\mathrm{P}(\mathrm{X} < 1.96 \times \sqrt{0.015})$ for temporal and spatial noise respectively, we will get both results very close to 0.975 which is what we expected, and this suggests that they do have the expected variance.
A partial CM of size 8×8 for the product of the temporal and spatial noises is shown in Figure 7, and we see in this subsample, there are many blocks (excluding the diagonal) have dark blue or dark red color, and this means there are many pairs of variables which are either highly positively or negatively correlated, and thus the product matrix of the two noises are correlated across V number of variables.

1.5

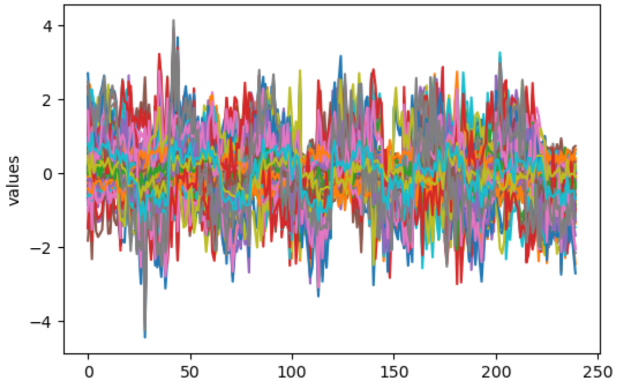

Figure 8

The two matrix products mentioned in the question can exist as they do have the correct shape for multiplication. Since these two terms can be treated as extra noises added to the data itself (TC and SM) which makes them look more realistic (not just 0 and 1s for all data points before standardisation), so we can just leave them there as they can be included in the model error term (E).
The 100 randomly selected time-series from X are plotted in Figure 8, and the variance plot for all 441 variables are shown in Figure 9 by dots. As we can see that most variables have a very small variance (around 0) which means they are quite stable across the timeline (lines that are more 'flat' and have smaller fluctuations in Figure 8). Also, we can see some variables of about (location)number 30-120 and 300-400 have greater variance which means these variables fluctuate more across the 240 time points, these particular variables are represented by those lines(in Figure 8) that have higher peaks and lower troughs across the timeline. In addition, these variables indicate where the pixel value equals to 1 in those SM matrices. As shown in Figure 10, if we reshape the 441 variables' variance into a 21 by 21 matrix and plot it by a heatmap, we can see this is exactly the positions of ones in all SMs (refer to Figure 3)

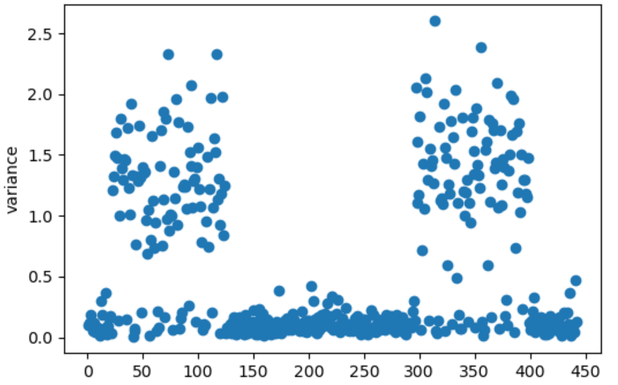

Figure 9

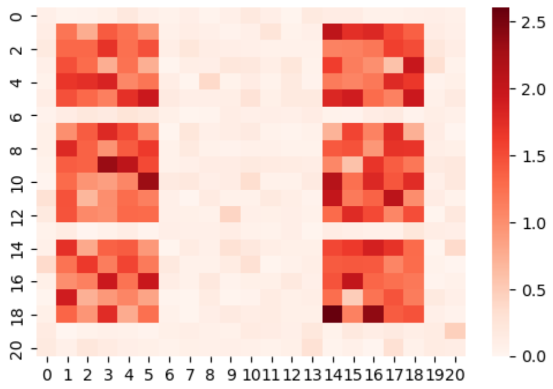

Figure 10

Q2

2.1

For least square regression(LSR), the 6 retrieved sources for both TC(D_lsr) and SM(A_lsr) are shown in Figure 12(next page). Note that the absolute operation is applied on retrieved SMS when plotting. On the left of Figure 11 (For better visualisation, the D_lsr for both plots are calculated from the **absolute** A_lsr), we can see a obvious linear relationship between 30th column of standardised X and 3rd column of D_Lsr, while on the right clearly there is no relationship with 4th column of D_Lsr. Since the 30th column of standardised X is actually 30th data element of SMs which is the pixel at 2nd column 9th row, and if we look at Figure 12, among all SMs, only SM3 has color yellow on this particular pixel (which means value of 1 in the matrix and the rest are all 0). Thus, only the corresponding TC3 is activated and this explains why 30th column of X only linearly depends on retrieved TC3 but not other columns.

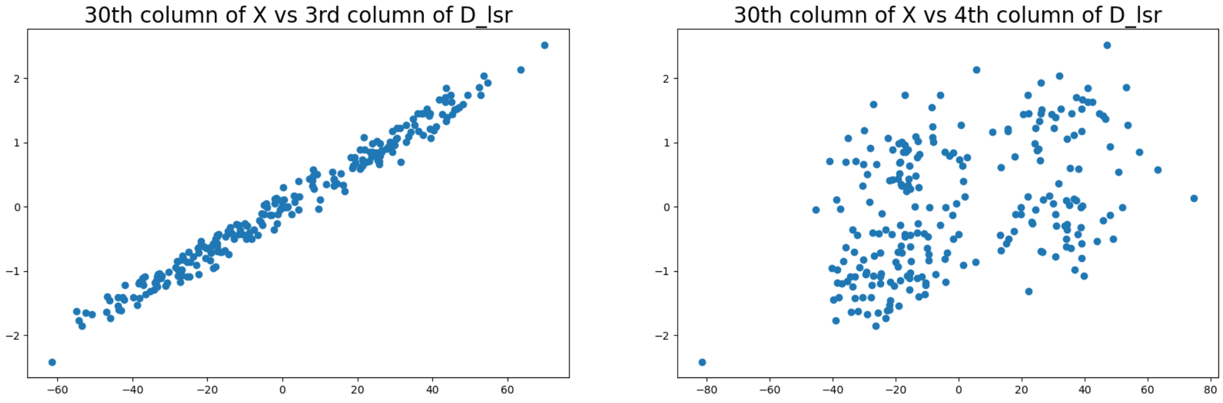

Figure 11

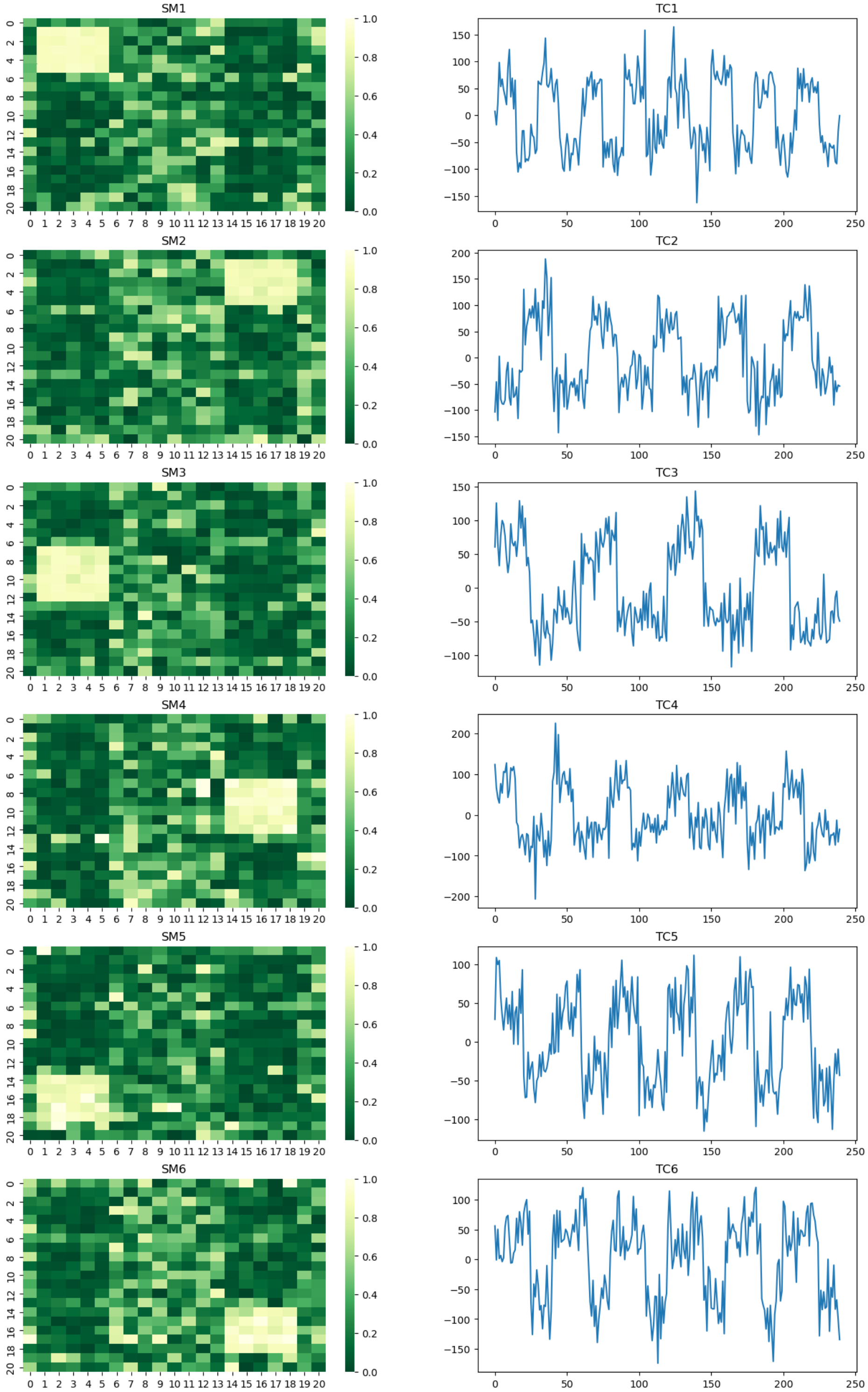

Figure 12: Left are 6 retrieved source SMs Right are retrieved TCs

2.2

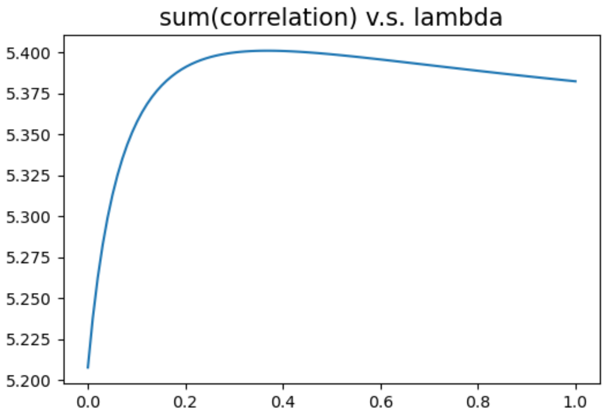

Figure 13

By using the "check and guess" method to estimate penalising term for ridge regression(RR), I end up with $\lambda = 0.37$ and $\widetilde{\lambda} = 163.17$ this λ also maximises the total maximum absolute correlation between each original TCs and D_rr(Refer to Figure 13). Note that if we estimate D_rr based on the **absolute** A_rr, $\lambda = 0.11$ will maximise the sum correlations (Refer to Figure 14: where blue line shows the sum correlations for different λ and orange line is the sum correlations for LSR) However, for both case, if such λ values are chosen, the sum maximum absolute correlations for RR ($\sum c\_TRR$) will always be greater than for LSR($\sum c\_TLSR$). Here, I've decided **not to** apply absolute operations(let $\lambda = 0.37$), I've got $\sum c\_TRR = 5.4011$ and $\sum c\_TLSR = 5.1700$. Please refer the notebook for more details of the calculation.

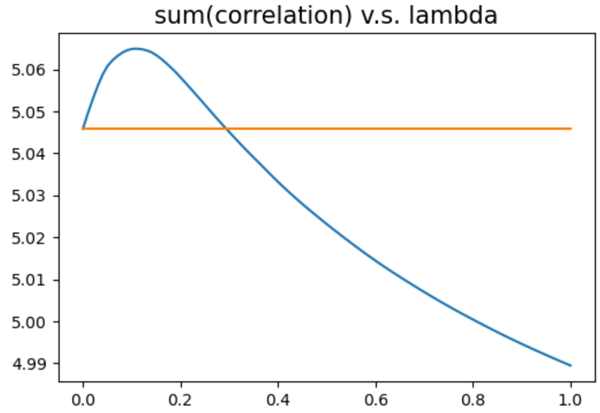

Figure 14

For $\lambda = 1000$, the plots for the 1st vector of A_rr and the 1st vector of A_lsr are shown in Figure 15. On the left, we can see from the y-axis' scale that most values are around zero, and we do find RR shrinks all values in the 1st vector towards zero (to a very small value maximum is only 0.0005) but not to exactly 0.

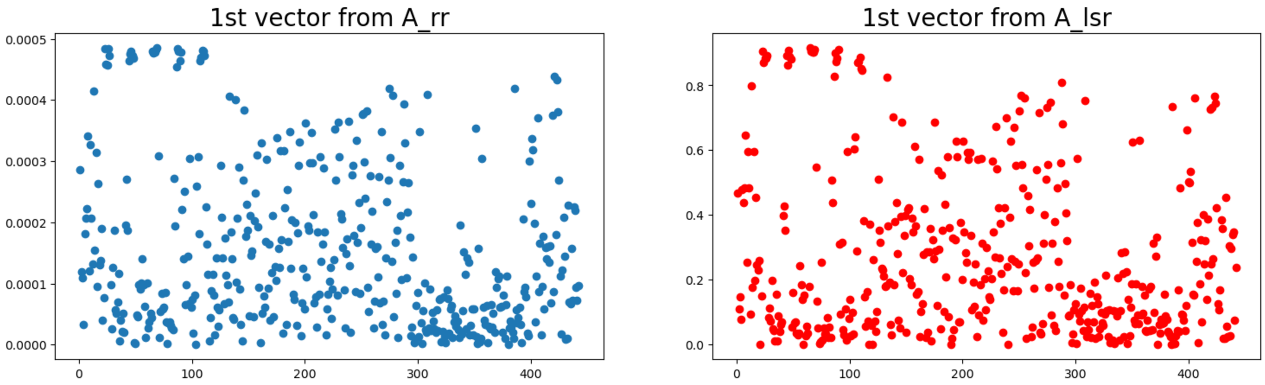

Figure 15: Left RR, Right LSR

2.3

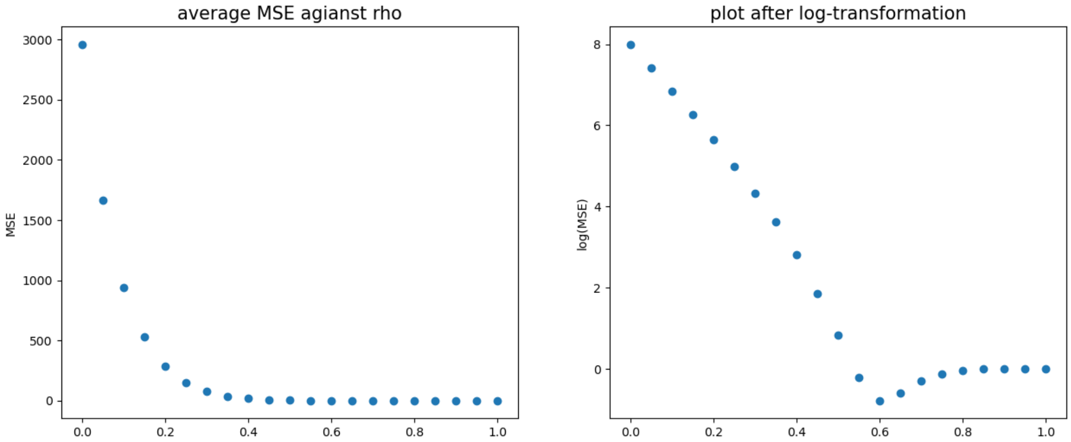

Figure 16: Left without log-transformation Right with log-transformation

On the left plot of Figure 16, we can see how the average MSE over 10 realisations are changed across ρ. For better visualisation, we can apply a log-transformation on the MSEs, from the right plot we can see the minimum MSE occurs at $\rho = 0.6$ with avg(MSE)=0.4589, and after the same point, MSE has slightly increased and seems converge to 1. Thus, I think it's fine to select this ρ value as this point does appear to be a global minimum (at least in the range of [0,1]).

2.4

By setting $\rho = 0.6$ and $\lambda = 0.37$ for RR, the LASSO regression(LR) estimates(D_lr and A_lr) do have greater sum(correlations) than RR with both original TCs and SMs:
$\sum c\_TLR = 5.4039$ and $\sum c\_TRR = 5.4011$,
$\sum c\_SLR = 5.0854$ and $\sum c\_SRR = 3.1260$
The two conditions in the question are both satisfied. Note that the sum correlations were calculated based on **no absolute operations applied on A**, we can see from Figure 17 the temporal plot for LR and RR look quite similar and this explains why $\sum c\_TLR$ is only slightly greater than $\sum c\_TRR$, but if you apply absolute operation on retrieved A and calculate retrieved D from it, you will find a little bit more difference between $\sum c\_TLR$ and $\sum c\_TRR$.
The estimates of D and A for LR and RR are plotted on Figure 17(next page). As shown clearly from

the spatial maps(retrieved A), there are much more false positives in RR's retrieved A than in LR's retrieved A. (In this case, false positive means expecting a 0 pixel-value in the original SM matrix but a non-zero pixel value given in the retrieved SM). The reason behind this difference is that Ridge Regression shrink values to a very small number (not exactly zero) while LASSO does shrink most of those insignificant pixels to exactly zero.

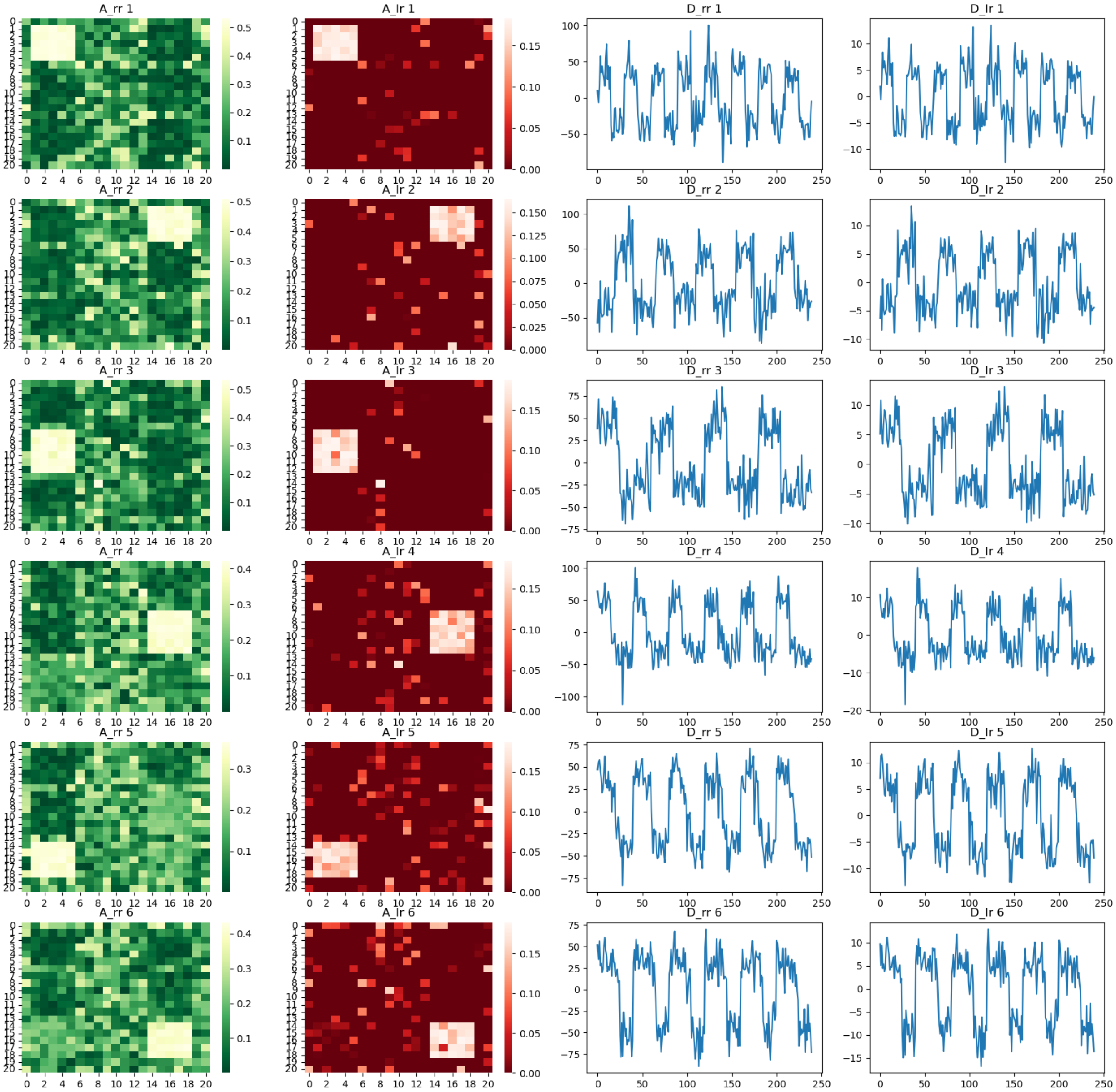

Figure 17: 1st column: A_rr 2nd column: A_lr 3rd column: D_rr 4th column: D_lr

2.5

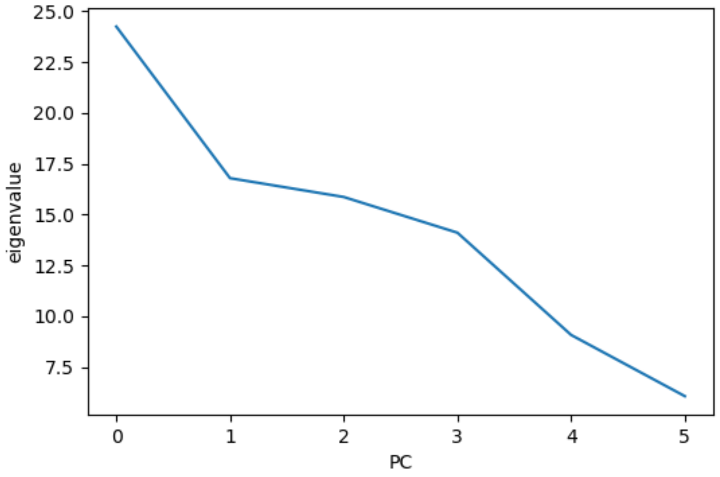

Figure 18

From the eigenvalues' plot (Figure 18), we can see the last principle component PC6 has the smallest eigenvalue since it also has the least percentage of explained variance to the model.

The regressors in Z and source TCs plot are shown in Figure 19(page 12), we can see the shape of PCs are deteriorated compare to the TCs. In fact, the first PC (Z1) does appear to be a reasonable shape even though it's not an exact match with TC1 either, but the other 5 PCs' shapes just don't make much sense, especially the last 3, and this is due to the way principal components are constructed such that the first PC explained the most variance of the mode, second PC explained the second most variance and so on.

The results of D_pcr and A_pcr are shown in Figure 20(page 13), we can see that among RR, LSR, LR and PCR, PCR has the worst performance. The first reason is mentioned above, only the first one or two principal components are useful. Also, since the principal components are the linear combination of the original variables in the design matrix, then the regressors in Z will have a completely different structure to TC which results in that the estimates of A and D from PCR will have completely different meanings to the original model (hard to interpret as well). Another important reason to why PCR has inferior performance is that PCR is agnostic to the response variable, in other words, PCR doesn't take the response (which is X in our case) into consideration at all when estimating D and A, while the other regression method all somehow include X in their process of making estimations.

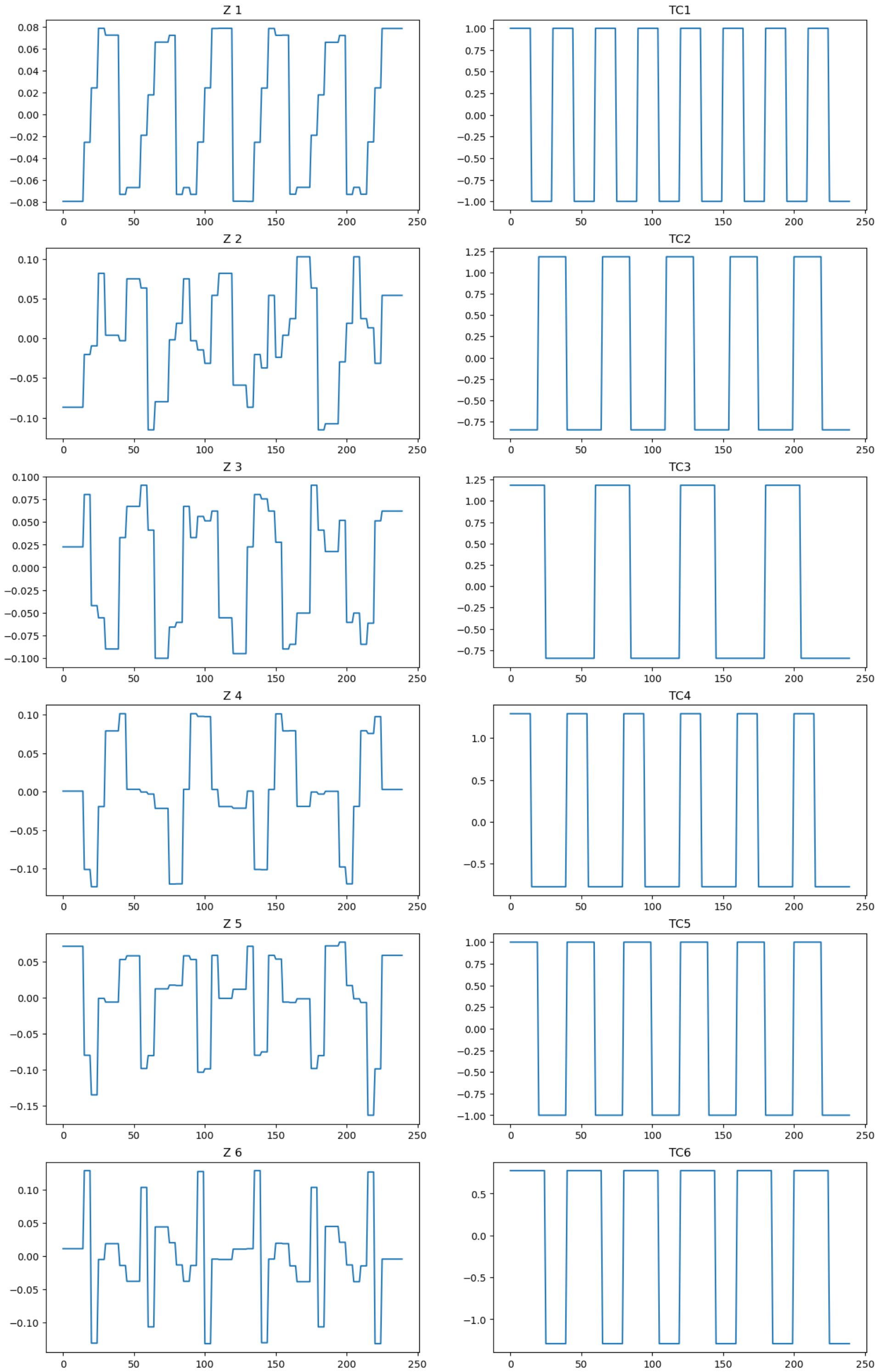

Figure 19: Left PCs, Right source TCs

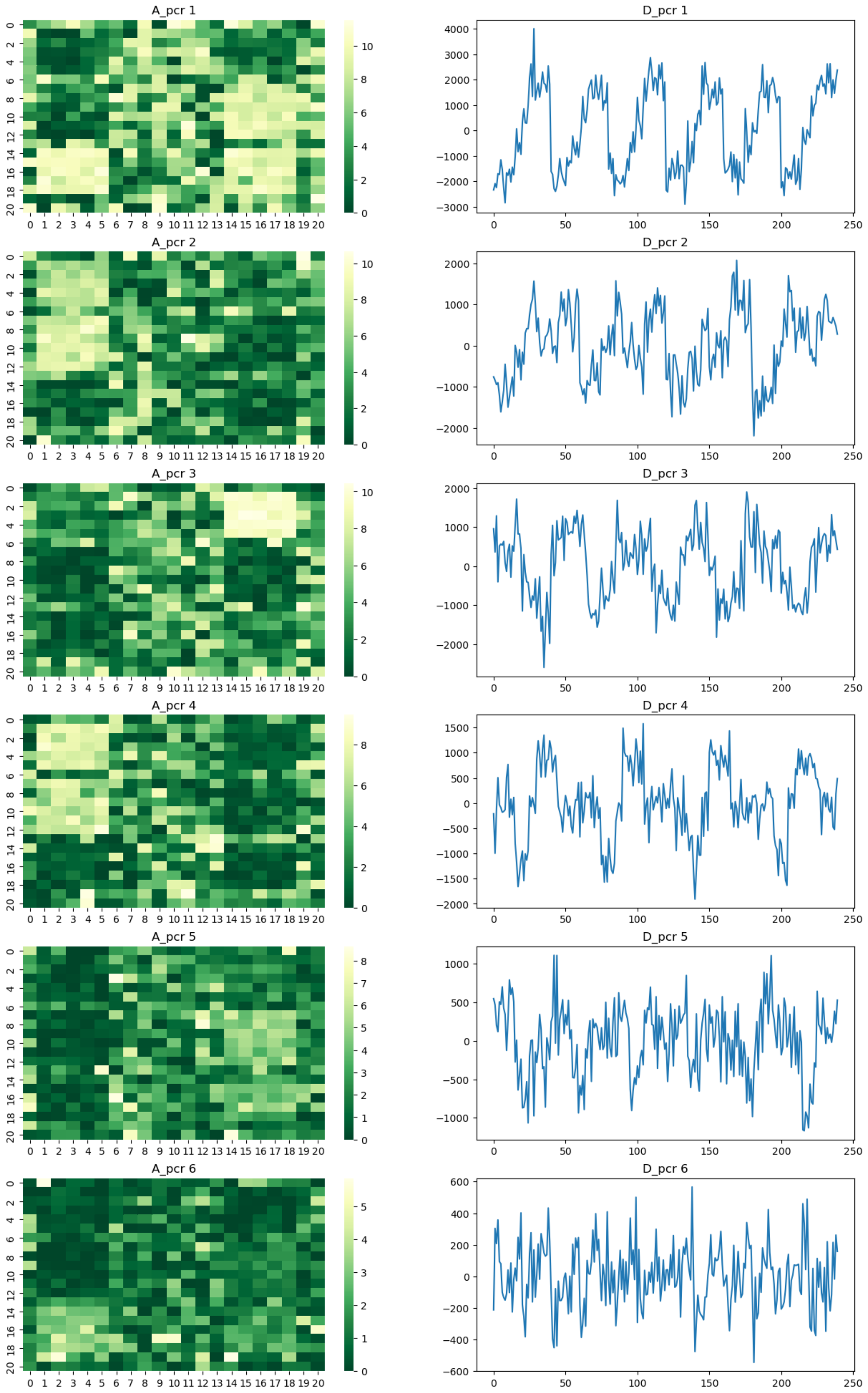
A_pcr 1
D_pcr 1
4000
3000
2000
1000
0
−1000
−2000
−3000
A_pcr 2
D_pcr 2
2000
1000
0
−1000
−2000
A_pcr 3
D_pcr 3
2000
1000
0
−1000
−2000
A_pcr 4
D_pcr 4
1500
1000
500
0
−500
−1000
−1500
−2000
A_pcr 5
D_pcr 5
1000
500
0
−500
−1000
A_pcr 6
D_pcr 6
600
400
200
0
−200
−400
−600
0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20
0 2 4 6 8 10 12 14 16 18 20
0 50 100 150 200 250
0 2 4 6 8 10
0 1 2 3 4 5 6 7 8

Figure 20