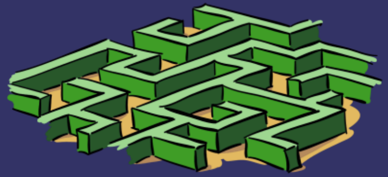
Overview of my work in Jul and Aug

- Classification of *Braking* and *Normal* Windows using PCA eigen vectors

- Classification of *Braking* and *Normal* Windows using AutoRegressive Model



Classification using PCA

- PCA data matrix consists of *variables* and *observations*

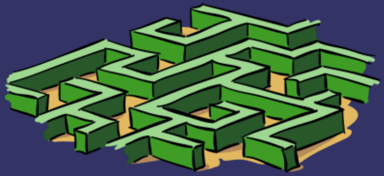
- I am considering channels as variables

- I am considering windows data samples as observations

- I consider 24 channels and 1.5sec window (300 samples) across these channels

- I reduce each window signal to Zero Mean and Unit Variance

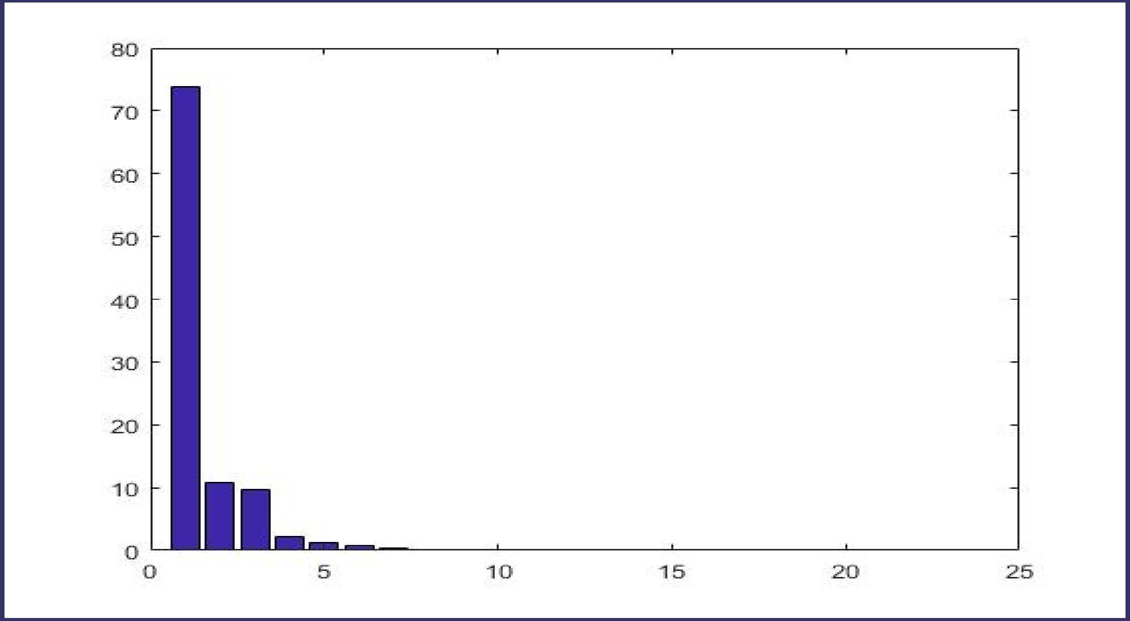
- So the data matrix fed to PCA is of size 300 * 24

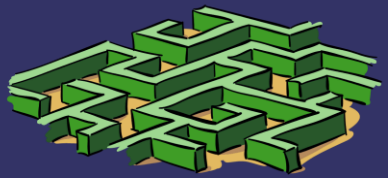


Classification using PCA (contd ...)

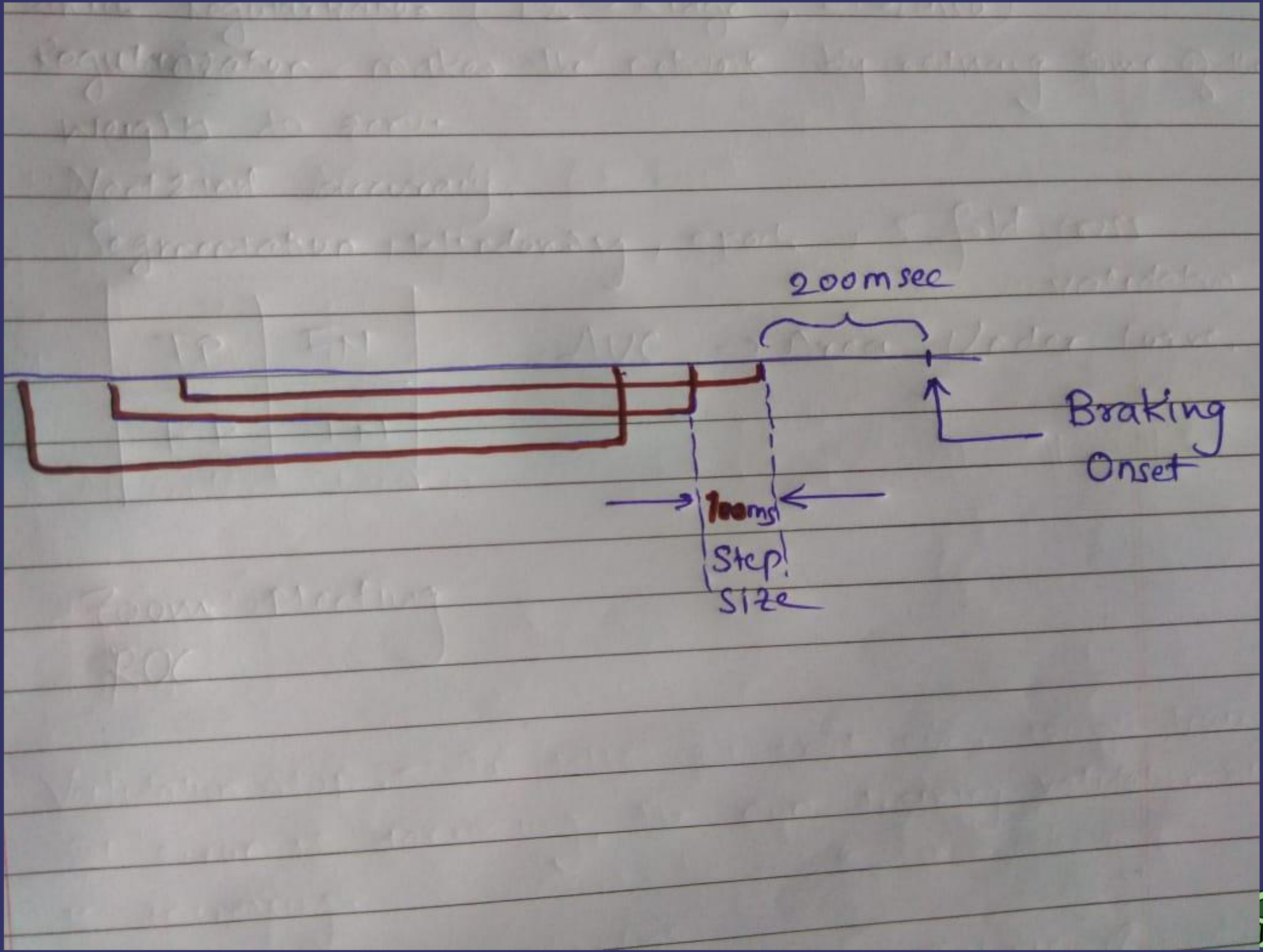
- PCA generates a 24*24 matrix with eigen vectors

- I am considering first 4 eigen vectors based on maximum information for each *point* (little 1.500 sec window)





Overlapping Windows



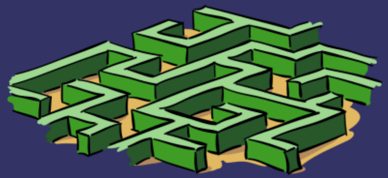
Classification using PCA (contd ...)

- There are totally such 30 braking and 70 normal points for each of 14 training subjects

- This gives me 14*30*6 + 14*70*6=8400 points (6 is number of overlapping windows)

- I shuffle the above points and their corresponding labels

- I feed the points to the neural network

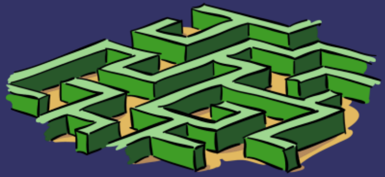


Over-Fitting issue

➲I faced an over-fitting issue where the <u>training accuracy is much higher than unseen data testing accuracy</u>

➲Solutions for Over-fitting issue

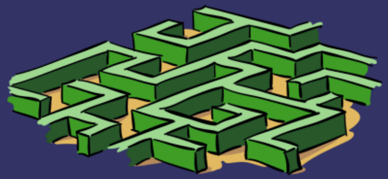
- Regularization
- Early Stopping
- More training data
- Less features



Regularization of Neural Network to avoid Over-Fitting

➲I implemented Regularization in two ways

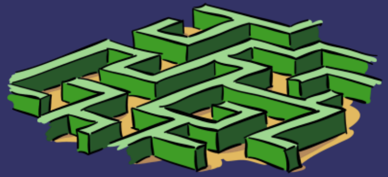
- Introducing a ReLU (Poslin) layer
- Make Mean Square Error (MSE) to be a function of weights



Training NN Multiple Times and Pick The Best Network

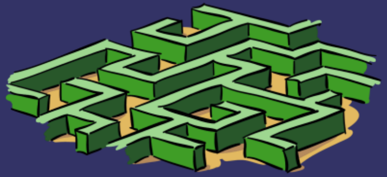
- I trained the neural network multiple times with Randomness in input data division and finite number of epochs

- I picked up the best network, based on the unseen data accuracy it provides

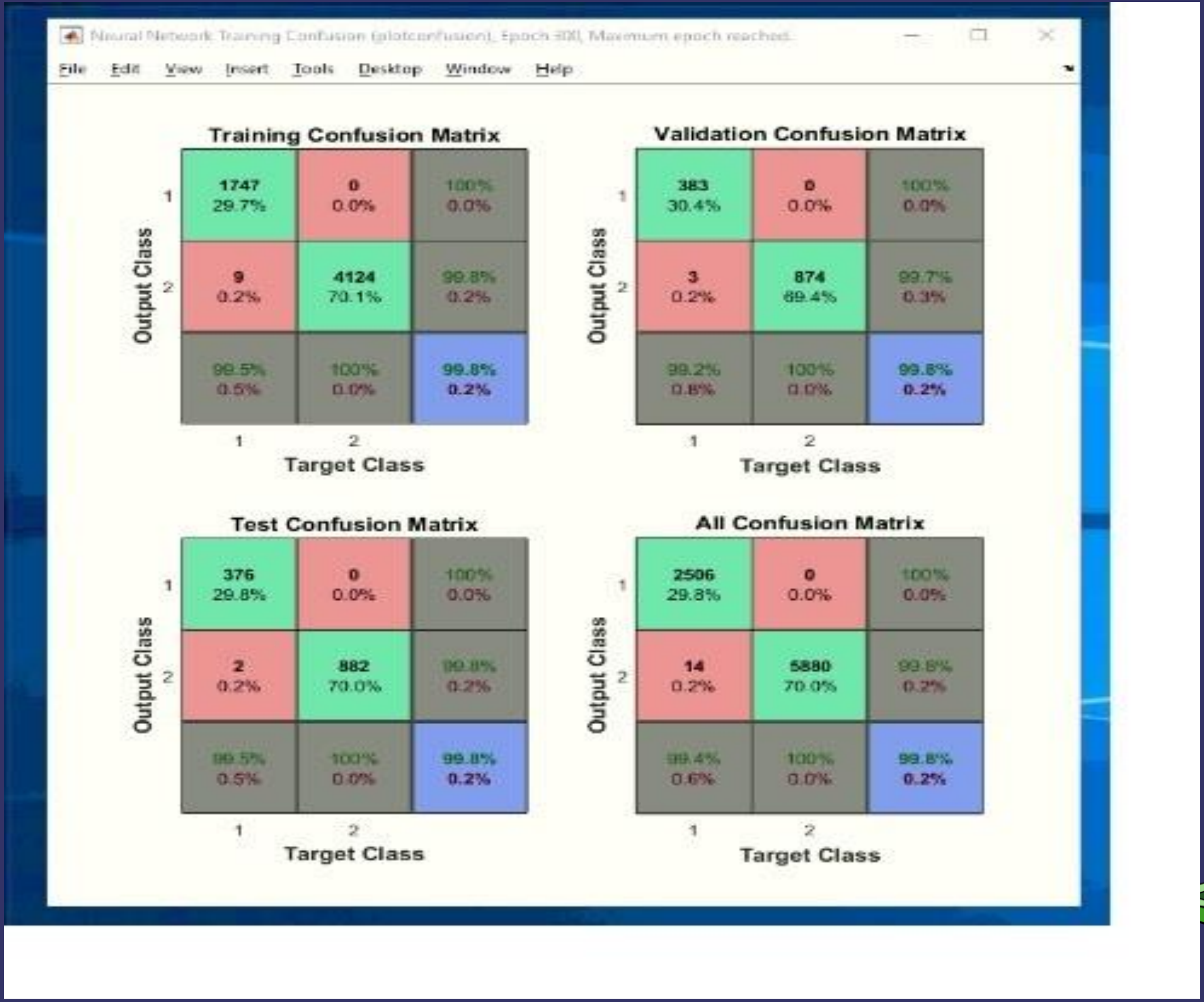


Add More Hidden Layers

- I added more hidden layers to represent a more complex non-linear function



Confusion Matrix for Training with PCA classification



Neural Network Training Confusion (plotconfusion), Epoch 300, Maximum epoch reached.

Training Confusion Matrix

Output Class

1	1747 29.7%	0 0.0%	100% 0.0%
2	9 0.2%	4124 70.1%	99.8% 0.2%
	99.5% 0.5%	100% 0.0%	99.8% 0.2%
	1	2	

Target Class

Validation Confusion Matrix

Output Class

1	383 30.4%	0 0.0%	100% 0.0%
2	3 0.2%	874 69.4%	99.7% 0.3%
	99.2% 0.8%	100% 0.0%	99.8% 0.2%
	1	2	

Target Class

Test Confusion Matrix

Output Class

1	376 29.8%	0 0.0%	100% 0.0%
2	2 0.2%	882 70.0%	99.8% 0.2%
	99.5% 0.5%	100% 0.0%	99.8% 0.2%
	1	2	

Target Class

All Confusion Matrix

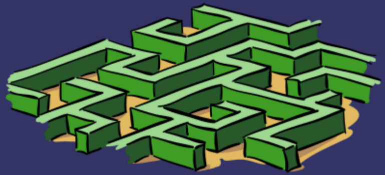
Output Class

1	2506 29.8%	0 0.0%	100% 0.0%
2	14 0.2%	5880 70.0%	99.8% 0.2%
	99.4% 0.6%	100% 0.0%	99.8% 0.2%
	1	2	

Target Class

Unseen data accuracy

- My unseen data accuracy is 70+ percent
- This accuracy is calculated as follows:
- Calculate the difference between the "target" and "predicted" labels.
- Error Percentage = ((Incorrect labels)/(Total number of labels)) * 100.
- Accuracy = 100 - Error Percentage



Results so far

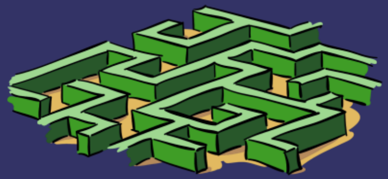
➲Early Detection:

- So far I am able to detect the braking signal 200ms early, which is less than mentioned in the paper which is 600ms

➲Braking Signal Detection Accuracy:

- I am able to detect the braking signal with 99% training accuracy and 70% testing accuracy with unseen data

➲Real Time Testing: Pending



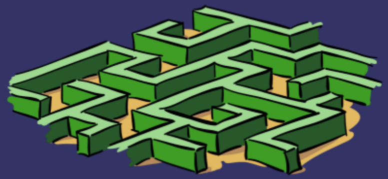
Auto-Correlation and Cross Correlation of Braking and Normal Windows

- I analyzed eight consecutive non-overlapping windows before the normal and braking points.

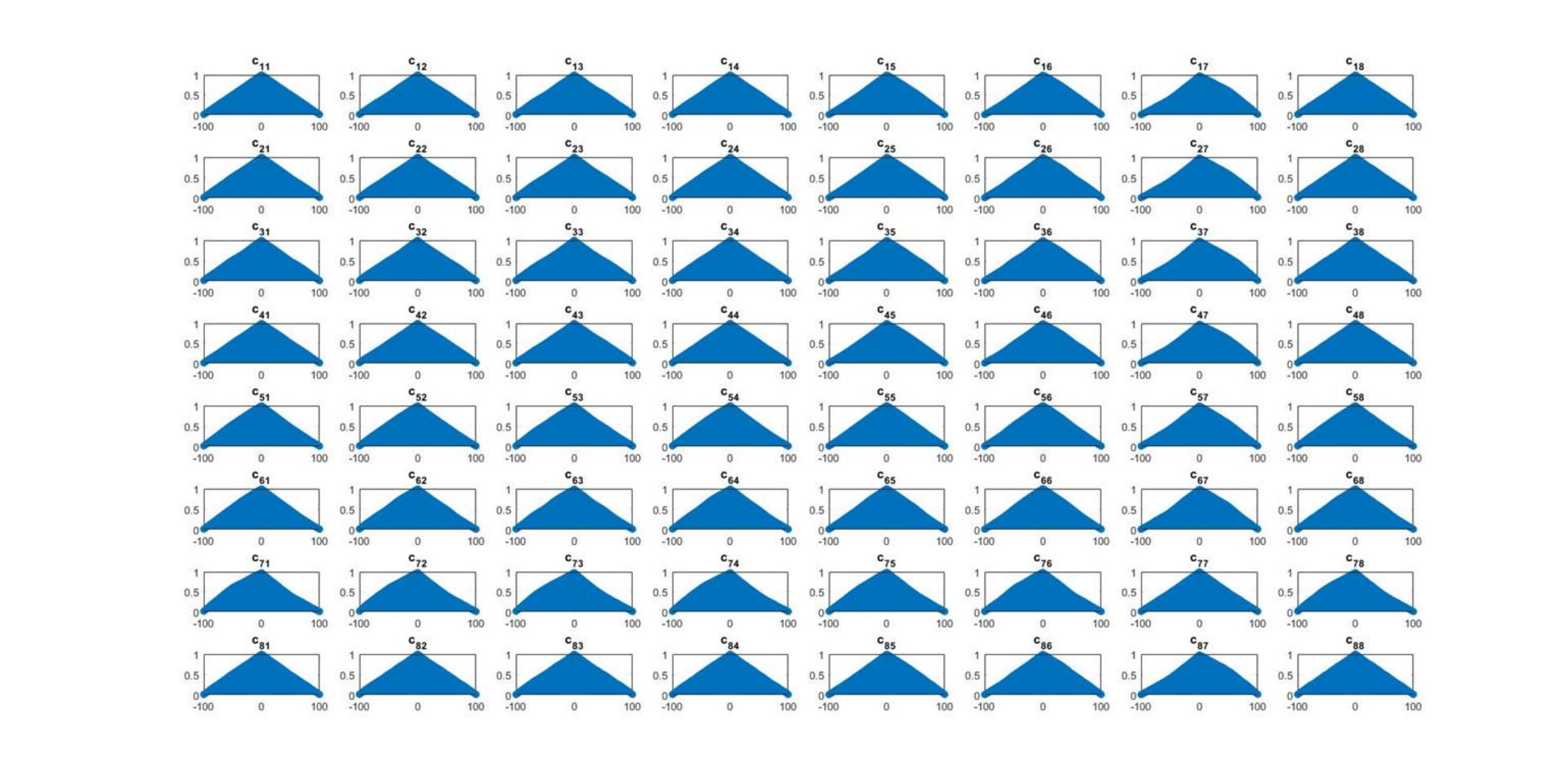
- I get the output as shown in the figures on the next slides

- My hypothesis is, whenever there is normal driving I get triangular auto/cross correlation without any abnormality.

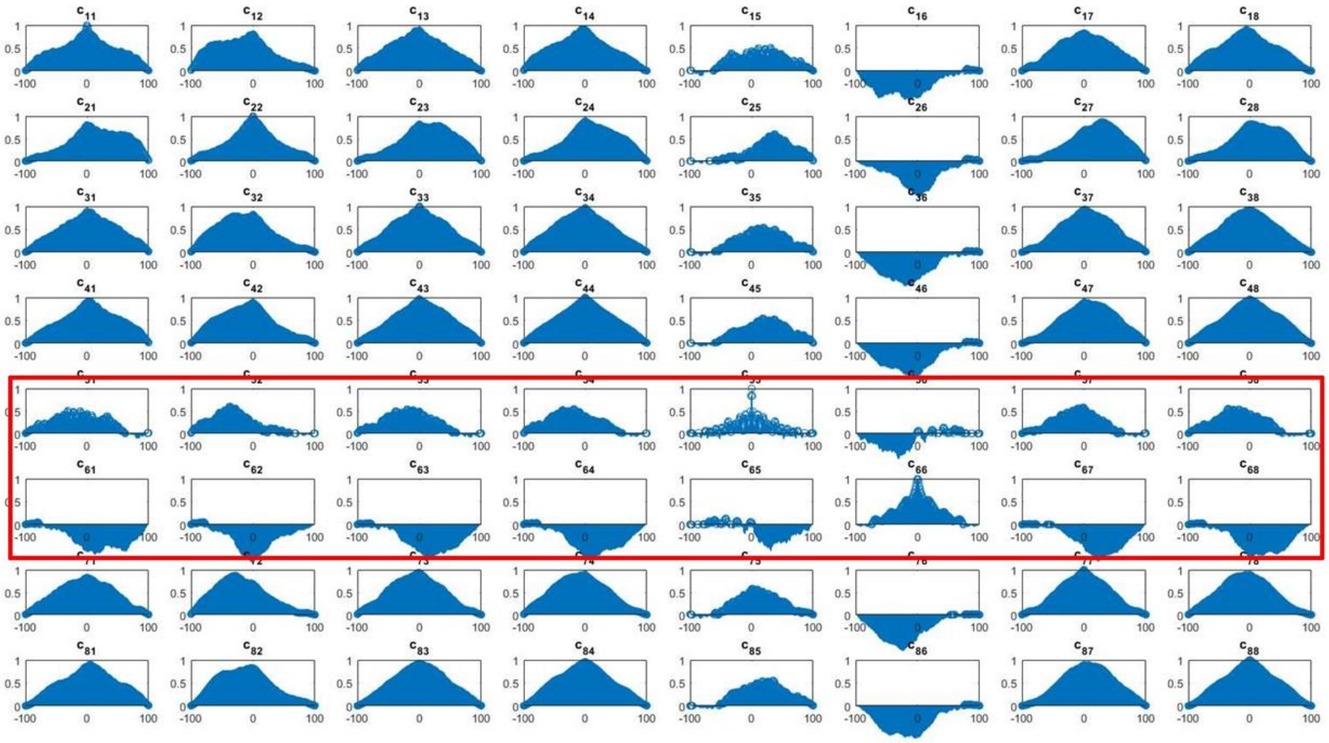
- However, when there is braking action, I get some abnormality in the auto/cross correlation



Auto and Cross Correlations before Normal



Auto and Cross Correlations before Braking


c11 c12 c13 c14 c15 c16 c17 c18
c21 c22 c23 c24 c25 c26 c27 c28
c31 c32 c33 c34 c35 c36 c37 c38
c41 c42 c43 c44 c45 c46 c47 c48
c51 c52 c53 c54 c55 c56 c57 c58
c61 c62 c63 c64 c65 c66 c67 c68
c71 c72 c73 c74 c75 c76 c77 c78
c81 c82 c83 c84 c85 c86 c87 c88
1
0.5
0
-100
0
100