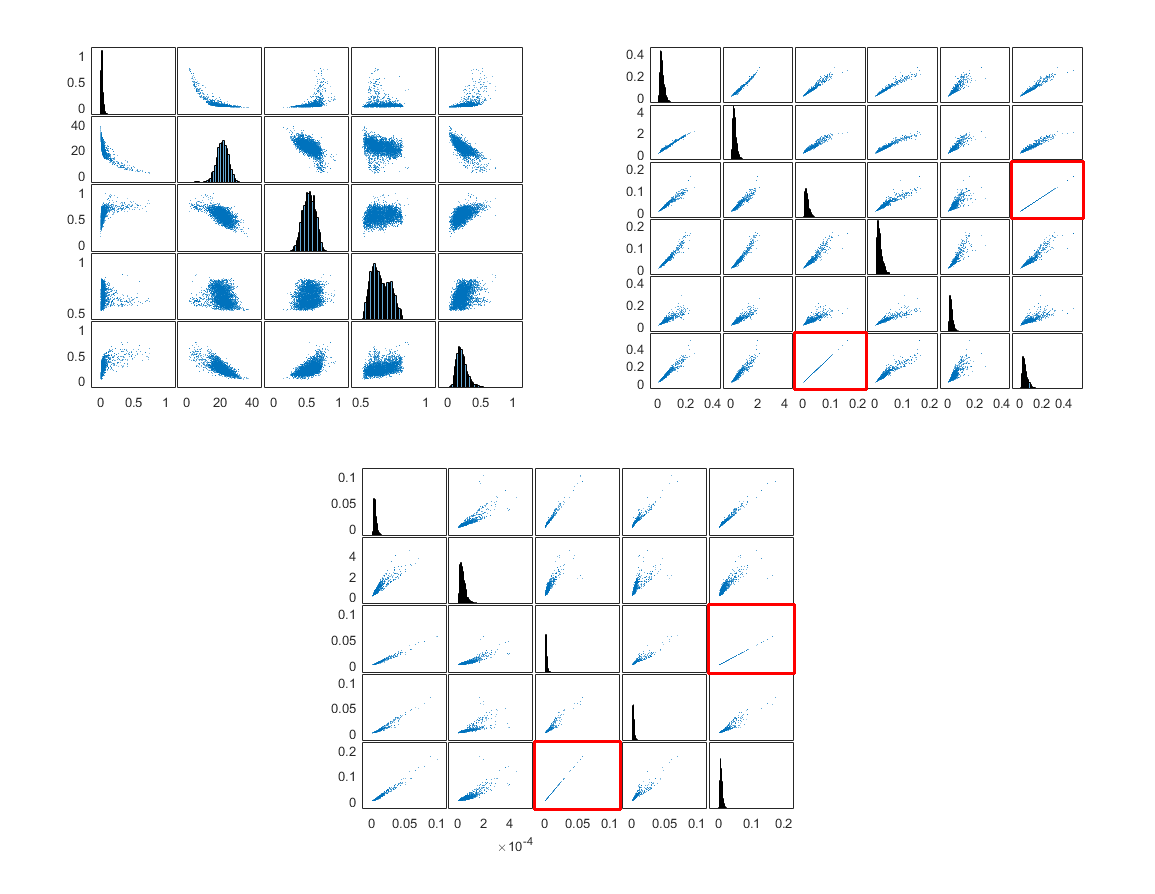
INCA Report

* Original Data had 22 variables
* This was then reduced to give three different subsets of the original data.
  + InputBase
    - Contains every variable but ones that would be inappropriate to include
  + Input
    - One with vars removed by me using plotmatrix
    - Show plot matrix
    - Why I chose those vars
    - Which ones I removed
    - % Plot matrices of Jitter attributes
    - % Plot matrices of shimmer attributes
    - % Plot matrices of rest
    - % From this can see Jitter:RAP and Jitter:DDP correlated
    - % From this can see Shimmer:APQ3 and Shimmer:DDA correlated
    - % So can remove Jitter:RAP and Shimmer:ADPQ3 vars



* + InputPCA
    - Same as input with further variables removed by PCA
    - How I performed PCA
    - Add in graph
    - What variables I ended up removing and why
    - % PCA - Remove more vars
    - % From this can see last 8 vars have an eigenvalue < 0.000
  + Why I did this:
    - Wanted to see the effect of removing variables
    - See if performance increased or decresased
  + What happened
* Validating the data
  + Need to check data was in the right format
* When using this data with each neural network I will split it (by hand) into the three sets; training, test and validate.
  + Training
    - Training will have the data from index 1:2928.
    - This is the first 21 patients.
  + Validation
    - Training will have the data from index 2929:4531.
    - This is the next 11 patients.
  + Test
    - Test will have the data from index 4531:5875
    - This is the last 10 patients
  + Why did I do this
    - Can’t let mat lab split the data itself.
    - May end up with the same patient in train/test/validate.
    - Wouldn’t be predicting the score then but learning what individual patients data looked like

MLP

* Can see that this is not great
* As an initial pass retry the ones highlighted with different training params

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Data Used | Hidden Layer Sizes | Run Number | MSE | Epoch | Training-Regression | Validation-Regression | Test-Regression |
| inputBase | 5 | avg | 81.436 | 2.666 | 0.377 | 0.628 | -0.045 |
|  | 10 | avg | 90.54 | 3 | 0.477 | 0.606 | -0.023 |
|  | 15 | avg | 107.01 | 6.666 | 0.608 | 0.499 | -0.144 |
|  | 20 | avg | 154.186 | 11 | 0.7 | 0.523 | 0.271 |
|  | 25 | avg | 90.354 | 5 | 0.562 | 0.534 | 0.053 |
|  | [5 10] | avg | 81.593 | 8.333 | 0.598 | 0.555 | 0.132 |
|  | [10 15] | avg | 97.147 | 5.333 | 0.459 | 0.526 | 0.015 |
|  | [15 20] | avg | 93.987 | 4.667 | 0.629 | 0.545 | 0.080 |
| input | 5 | avg | 87.719 | 5 | 0.45 | 0.52 | 0.018 |
|  | 10 | avg | 78.847 | 7.333 | 0.601 | 0.67 | 0.117 |
|  | 15 | avg | 99.955 | 3.666 | 0.579 | 0.499 | 0.252 |
|  | 20 | avg | 110.365 | 4.333 | 0.625 | 0.58 | 0.243 |
|  | 25 | avg | 128.796 | 3.333 | 0.488 | 0.505 | 0.006 |
|  | [5 10] | avg | 91.067 | 9.667 | 0.641 | 0.522 | -0.020 |
|  | [10 15] | avg | 83.656 | 10.333 | 0.646 | 0.565 | 0.092 |
|  | [15 20] | avg | 111.8816 | 4.667 | 0.551 | 0.544 | 0.081 |
| inputPCA | 5 | avg | 81.436 | 2.666 | 0.377 | 0.628 | -0.045 |
|  | 10 | avg | 90.54 | 3 | 0.477 | 0.606 | -0.023 |
|  | 15 | avg | 107.01 | 6.666 | 0.608 | 0.499 | -0.144 |
|  | 20 | avg | 154.186 | 11 | 0.7 | 0.523 | 0.271 |
|  | 25 | avg | 128.796 | 3.333 | 0.488 | 0.505 | 0.006 |
|  | [5 10] | avg | 90.278 | 8 | 0.531 | 0.548 | 0.182 |
|  | [10 15] | avg | 65.881 | 8.667 | 0.521 | 0.709 | 0.209 |
|  | [15 20] | avg | 195.3809 | 8 | 0.532 | 0.545 | 0.046 |

* IMPROVE BY SELECTING BEST RESULT AND TWEAKING WITH PARMAS

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Data Used | Hidden Layer Sizes | Number of runs | Training function | MSE | Epoch | Training-Regression | Validation-Regression | Test-Regression |
| inputPCA | 20 | 3 | trainbfg | 65.711 | 702.333 | 0.481 | 0.699 | 0.250 |
|  |  |  | trainscg | 110.093 | 6 | 0.230 | 0.411 | -0.087 |
|  |  |  | trainoss | 97.806 | 13.333 | 0.245 | 0.612 | -0.031 |
|  | [5 10] | 3 | trainbfg | 87.618 | 227.667 | 0.412 | 0.536 | 0.094 |
|  |  |  | trainscg | 78.863 | 16.667 | 0.331 | 0.617 | -0.060 |
|  |  |  | trainoss | 113.108 | 5 | 0.046 | 0.216 | 0.039 |
|  | [5 10] | 3 | trainbfg | 87.618 | 227.667 | 0.412 | 0.536 | 0.094 |
|  |  |  | trainscg | 78.863 | 16.667 | 0.331 | 0.617 | -0.060 |
|  |  |  | trainoss | 113.108 | 5 | 0.046 | 0.216 | 0.039 |
|  | [10 15] | 3 | trainbfg | 101.626 | 31.667 | 0.255 | 0.481 | -0.014 |
|  |  |  | trainscg | 64.427 | 24.667 | 0.459 | 0.679 | 0.240 |
|  |  |  | trainoss | 94.649 | 18.333 | 0.313 | 0.493 | 0.062 |

* DISCUSS CONCLUSIONS

Latent = [ 78.0850

18.1957

0.2173

0.0204

0.0056

0.0050

0.0027

0.0010

0.0001

0.0000

0.0000

0.0000

0.0000

0.0000

0.0000

0.0000 ]

