ISYE 8803 HDDA

Homework 4

July 13th, 2021

3.

a)

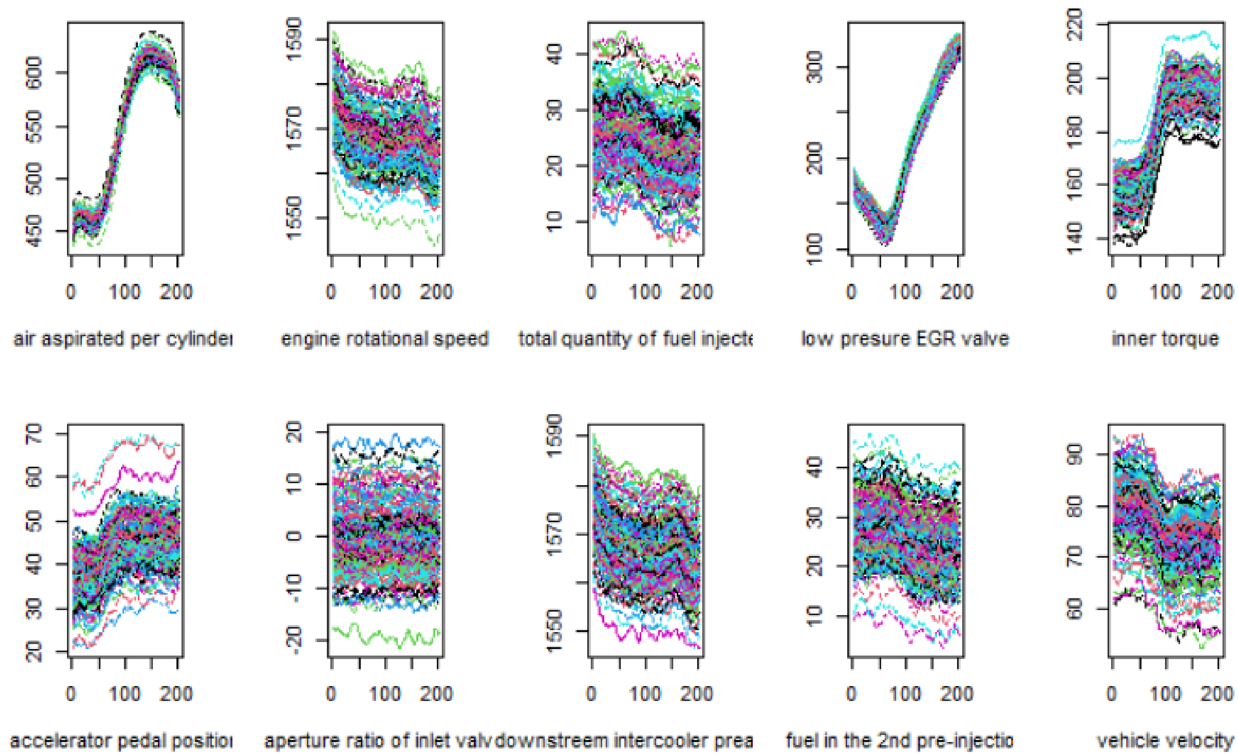

Figure 1: Sensor Training Dataset

b) Code

c)

From the lecture we can see the group lasso formula derived as following:

$$\min_{\hat{\boldsymbol{\beta}}_k} \frac{1}{2}\left\| \mathbf{y} - \sum_{k=1}^{K} \mathbf{X}_k \hat{\boldsymbol{\beta}}_k \right\|^2 + \lambda_1 \sum_{k=1}^{K} \left\| \boldsymbol{\beta}_k \right\|$$

$$\left\| \boldsymbol{\beta}_k \right\| = \sqrt{\sum_{j=1}^{p_k} \beta_{kj}^2}$$

$$(\underbrace{x_{11},\ldots,x_{1p_1}}_{\mathbf{X}_1}, \underbrace{x_{21},\ldots,x_{2p_2}}_{\mathbf{X}_2}, \ldots, \underbrace{x_{K1},\ldots,x_{Kp_K}}_{\mathbf{X}_K})$$

In our case, each $X_{1..K}$ matrix represents the coefficients for each sensor that we've then reduced using B – spline coefficients, where $K = 10$ in our case.

As seen in the lecture we can use B – spline to reduce dimensionality and derive the following formulation:

We can use b-splines to reduce the dimensionality:

$$\beta_j(t) = \sum_{k=1}^{10} b_{kj}\theta_{kj}(t) = \boldsymbol{\theta_j^T b_j}$$

With this, we have:

$$\int_0^1 x_{ij}(t)\beta_j(t)dt = \int_0^1 x_{ij}(t)\boldsymbol{\theta_j^T}(t)dt\boldsymbol{b_j} = \boldsymbol{z_{ij}b_j}$$

Therefore:

$$y_i = \boldsymbol{z_{i1}b_1} + \boldsymbol{z_{i2}b_2} + \cdots + \boldsymbol{z_{i10}b_{10}} + \boldsymbol{\epsilon_i}$$

Our goal is to estimate $\boldsymbol{b} \in \mathbb{R}^{10\times 10}$, using group lasso. The problem we want to solve is:

$$\min_{\boldsymbol{b}} \left|\left|\boldsymbol{y} - \boldsymbol{Zb}\right|\right|_2^2 + \sum_{j=1}^{10} \left|\left|\boldsymbol{b_j}\right|\right|_{\boldsymbol{2}}$$

Where $\beta_j(t)$ is our basis matrix that can be used to derive our b-spline coefficients z_{i1} *to* z_{i10}, for
all 10 sensors.

d)

When getting the coefficient values from our glasso fitted object, we can see that the model select all variables except for the 6th coefficient / feature. This can be represented by looking at the output of the group variables and seeing the coeffcient values that are set to 0. In our case this is the *accelerator pedal position variable.*

```
glasso_lambda <- min(glasso$lambda)
which(glasso$fit$beta[,glasso$min]==0)

V601 V602 V603 V604 V605 V606 V607 V608 V609 V610 V611 V612 V613
V614 V615 V616 V617 V618 V619 V620 V621 V622 V623 V624 V625
 602  603  604  605  606  607  608  609  610  611  612  613  614
615  616  617  618  619  620  621  622  623  624  625  626
V626 V627 V628 V629 V630 V631 V632 V633 V634 V635 V636 V637 V638
V639 V640 V641 V642 V643 V644 V645 V646 V647 V648 V649 V650
 627  628  629  630  631  632  633  634  635  636  637  638  639
640  641  642  643  644  645  646  647  648  649  650  651
V651 V652 V653 V654 V655 V656 V657 V658 V659 V660 V661 V662 V663
V664 V665 V666 V667 V668 V669 V670 V671 V672 V673 V674 V675
 652  653  654  655  656  657  658  659  660  661  662  663  664
665  666  667  668  669  670  671  672  673  674  675  676
V676 V677 V678 V679 V680 V681 V682 V683 V684 V685 V686 V687 V688
V689 V690 V691 V692 V693 V694 V695 V696 V697 V698 V699 V700
 677  678  679  680  681  682  683  684  685  686  687  688  689
690  691  692  693  694  695  696  697  698  699  700  701
```

All of the other coefficents are correlated with the predictor variable *air/fuel ratio,* however since we have 1000 observations the results are too long to show.

e)

After running the model against our test dataset we yield the following MSE:

$$MSE_{glasso} = 0.03853925$$