Build Both:

–A Leaps automatic model selector using Cp, and PRESS

- Best k models using Cp
- From those k calculate best PRESS

–And a lars automatic model selector using both Cp and cross validation MSE

- Best k models using Cp
- Best k using cross validated MSE
- Combine and choose 1

–Combine the two in a single function

–Compare predictions of each operating on Full 2nd order set of variables for Auto data, and on Each country of origin separately with full 2nd order set of variables.

CAlculate actual predictions X%*%lars.beta, and X%*%leaps.beta

And compare actual regression coefficients across countries to see if you can understand any country differences

–Anything interesting in you can pick up on what is going on in the different countries

Output below:

[1] "=========================="
[1] "AUTO(ALL COUNTRIES) MODEL COMPARISON"
[1] "=========================="
[1] "running leaps"
[1] 1
[1] "Press= 2965.80019718738"
[1] "MPSE= 7.82533033558675"
[1] "Cp= 5.48237932178012"
[1] 2
[1] "Press= 2973.46969745552"
[1] "MPSE= 7.84556648405151"
[1] "Cp= 6.41193394713218"
[1] 3
[1] "Press= 2963.97198199211"
[1] "MPSE= 7.82050654879183"
[1] "Cp= 6.58803898665906"

[1] 4
[1] "Press= 2969.00950290619"
[1] "MPSE= 7.81318290238471"
[1] "Cp= 6.72893176567476"

[1] 5
[1] "Press= 2985.56308580604"
[1] "MPSE= 7.89831504181491"
[1] "Cp= 6.80629604880136"

[1] 6
[1] "Press= 2984.94880916756"
[1] "MPSE= 7.89668997134276"
[1] "Cp= 6.96352987517361"

[1] 7
[1] "Press= 2989.50660643415"
[1] "MPSE= 7.90874763606918"
[1] "Cp= 7.13707916585133"

[1] 8
[1] "Press= 2973.77358535725"
[1] "MPSE= 7.86712588718849"
[1] "Cp= 7.15494914255413"

[1] 9
[1] "Press= 2975.27618155834"

[1] "MPSE= 7.87110100941359"
[1] "Cp= 7.19668547867724"

[1] 10
[1] "Press= 2989.26324829287"
[1] "MPSE= 7.90810383146261"
[1] "Cp= 7.21576451870499"

[1] "==============="
[1] "Best Model"
[1] "Press= 2963.97198199211"
[1] "MPSE= 7.82050654879183"
[1] "Cp= 6.58803898665906"
[1] " Model Statistics "
Residual Standard Error=2.699
R-Square=0.8841
F-statistic (df=12, 379)=240.8927
p-value=0

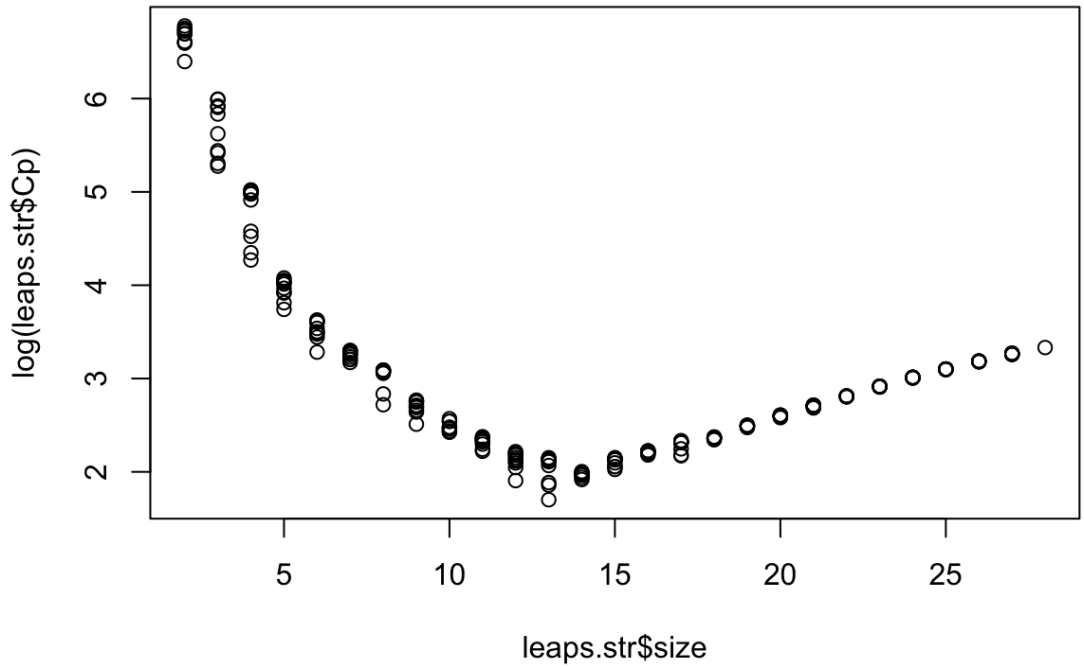
	Estimate	Std.Err	t-value	Pr(>|t|)
Intercept	386.6185	78.6636	4.9148	0.0000
displacement	-0.1649	0.0504	-3.2731	0.0012
acceleration	-5.0507	1.4386	-3.5109	0.0005
year	-8.5842	1.9851	-4.3242	0.0000
cylinderscylinders	-0.3815	0.0685	-5.5675	0.0000
cylindersweight	0.0016	0.0002	7.2242	0.0000
displacementhorsepower	0.0002	0.0001	3.5759	0.0004
displacementyear	0.0016	0.0006	2.5798	0.0103
horsepoweracceleration	-0.0071	0.0013	-5.6464	0.0000
weightacceleration	0.0002	0.0001	1.8989	0.0583
weightyear	-0.0002	0.0000	-7.1678	0.0000
accelerationyear	0.0652	0.0182	3.5939	0.0004
yearyear	0.0567	0.0129	4.3863	0.0000

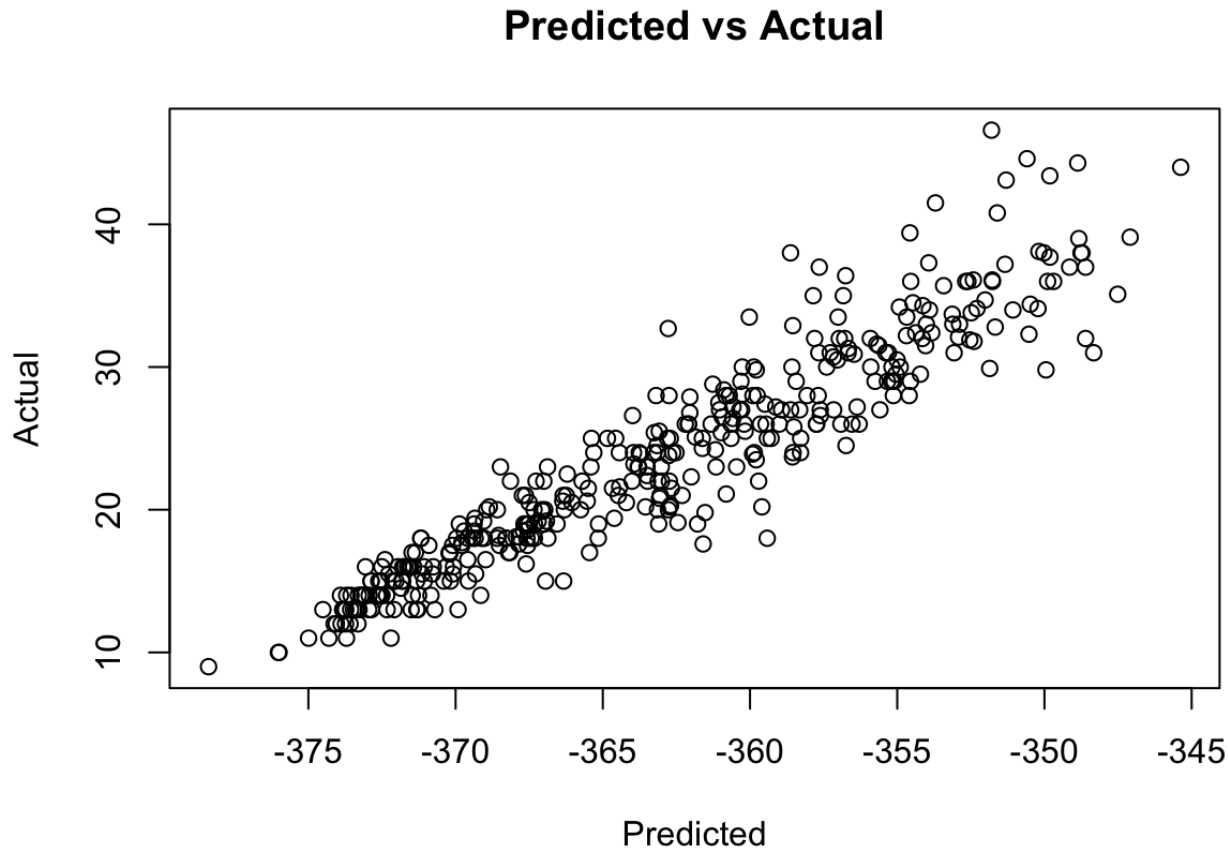
[1] "coeff without int"

displacement	acceleration	year
-0.1649241183	-5.0507390590	-8.5842494919
cylinderscylinders	**cylindersweight**	**displacementhorsepower**
-0.3815419206	0.0016003397	0.0001817800
displacementyear	**horsepoweracceleration**	**weightacceleration**
0.0016456187	-0.0071187139	0.0002029652
weightyear	**accelerationyear**	**yearyear**
-0.0002126282	0.0652495741	0.0566974152

[1] "correlation 0.940259364558168"

Leaps Plots:





[1] "running lars"
[1] "beta.out"

```
           cylinders         displacement
        2.580089e+00        -2.357853e-01
          horsepower               weight
        6.664148e-01        0.000000e+00
        acceleration                 year
       -1.655712e+00        0.000000e+00
  cylinderscylinders  cylindersdisplacement
       -3.452062e-01        0.000000e+00
 cylindershorsepower       cylindersweight
        2.347345e-02        2.704416e-04
cylindersacceleration        cylindersyear
        1.894657e-01       -5.713333e-02
displacementdisplacement displacementhorsepower
        0.000000e+00        0.000000e+00
 displacementweight displacementacceleration
        1.287367e-05       -2.548747e-03
   displacementyear   horsepowerhorsepower
        2.851404e-03       -4.914653e-04
  horsepowerweight horsepoweracceleration
       -7.500449e-06       -7.331224e-03
     horsepoweryear         weightweight
       -8.164636e-03        3.742530e-07
 weightacceleration           weightyear
        5.868171e-05       -1.455232e-04
accelerationacceleration     accelerationyear
        1.495249e-02        1.368659e-02
           yearyear
        1.021711e-02
[1] "lnd.out"
           cylinders         displacement
                TRUE                 TRUE
          horsepower               weight
                TRUE                FALSE
        acceleration                 year
                TRUE                FALSE
  cylinderscylinders  cylindersdisplacement
                TRUE                FALSE
 cylindershorsepower       cylindersweight
                TRUE                 TRUE
cylindersacceleration        cylindersyear
                TRUE                 TRUE
displacementdisplacement displacementhorsepower
               FALSE                FALSE
 displacementweight displacementacceleration
```

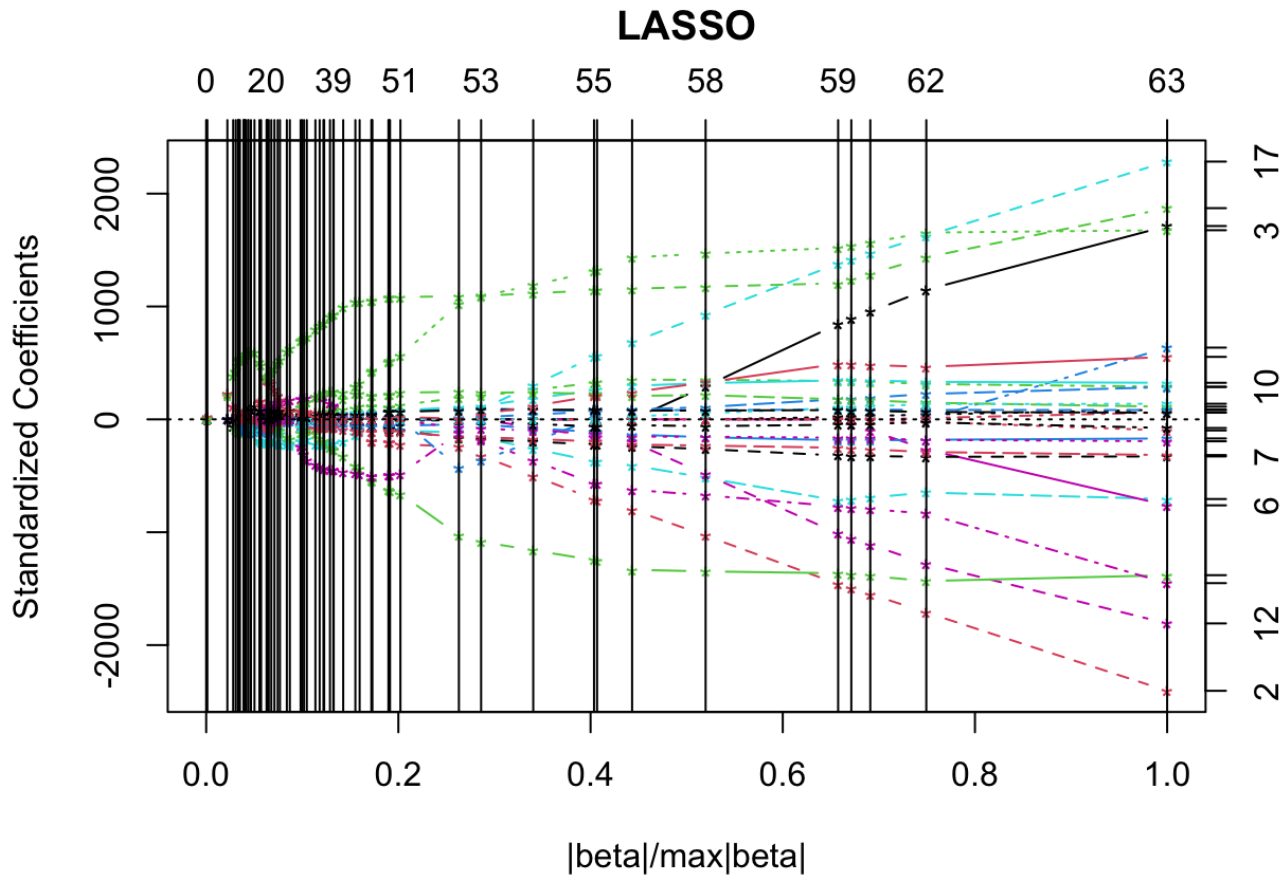
TRUE TRUE
displacementyear horsepowerhorsepower
TRUE TRUE
horsepowerweight horsepoweracceleration
TRUE TRUE
horsepoweryear weightweight
TRUE TRUE
weightacceleration weightyear
TRUE TRUE
accelerationacceleration accelerationyear
TRUE TRUE
yearyear
TRUE

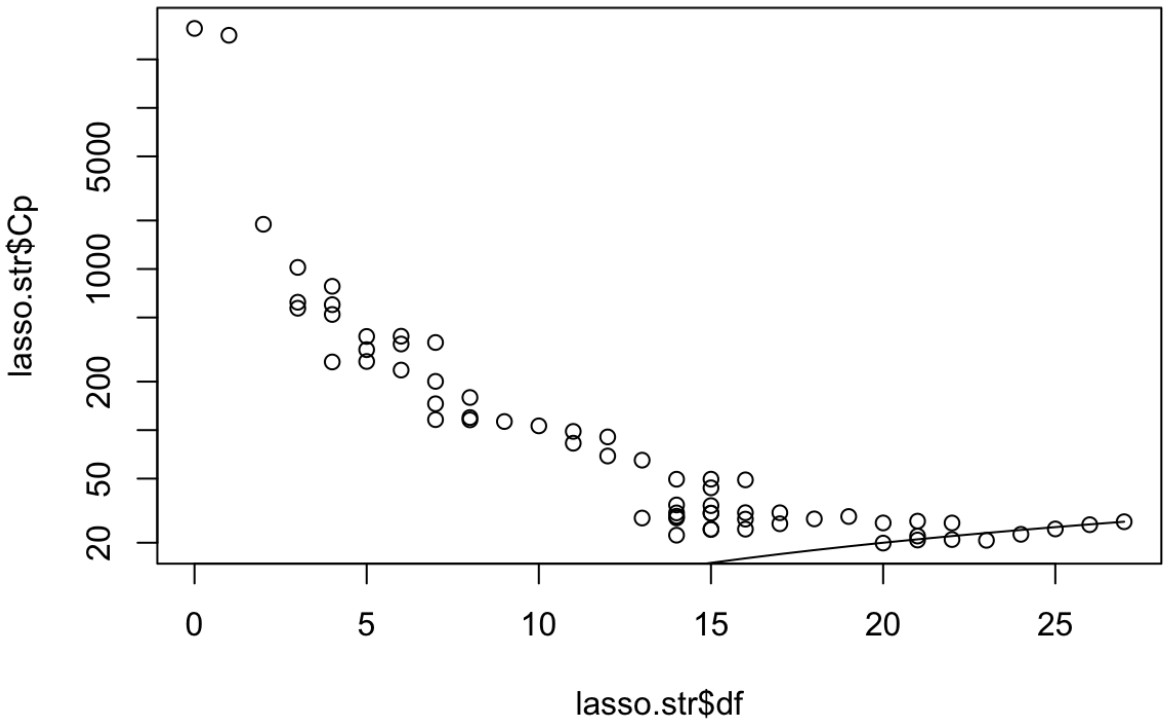
[1] "Cp= 20.9276348024184"

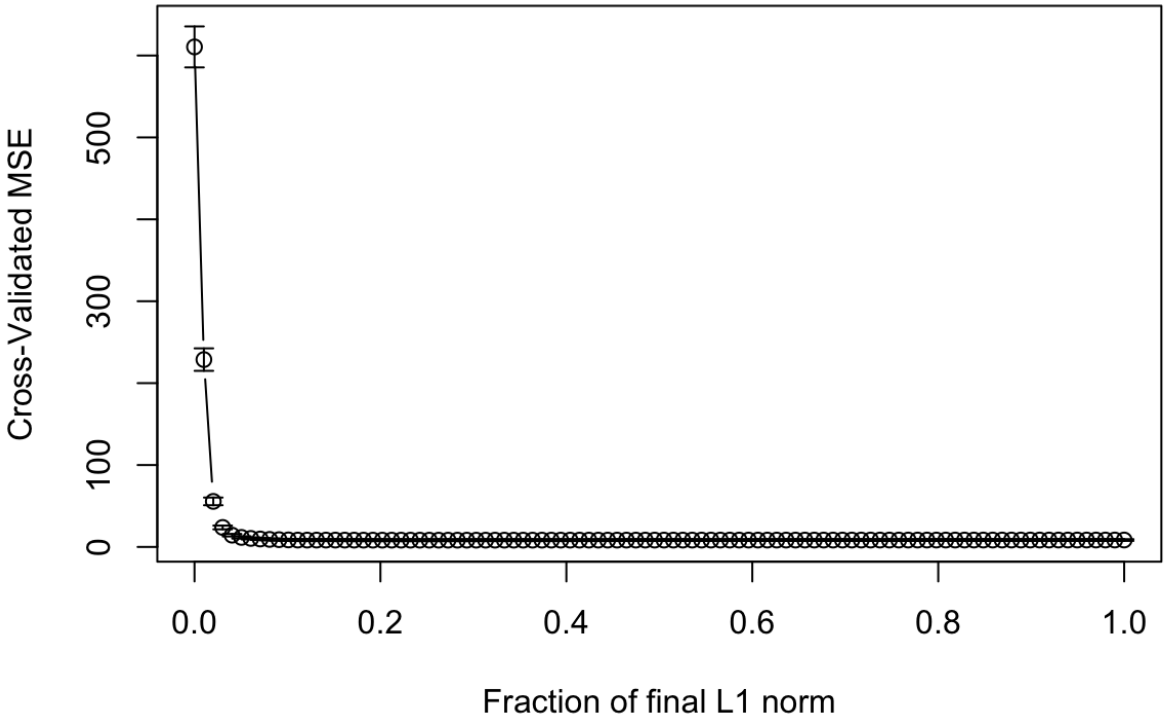
[1] "correlation 0.939378943903837"

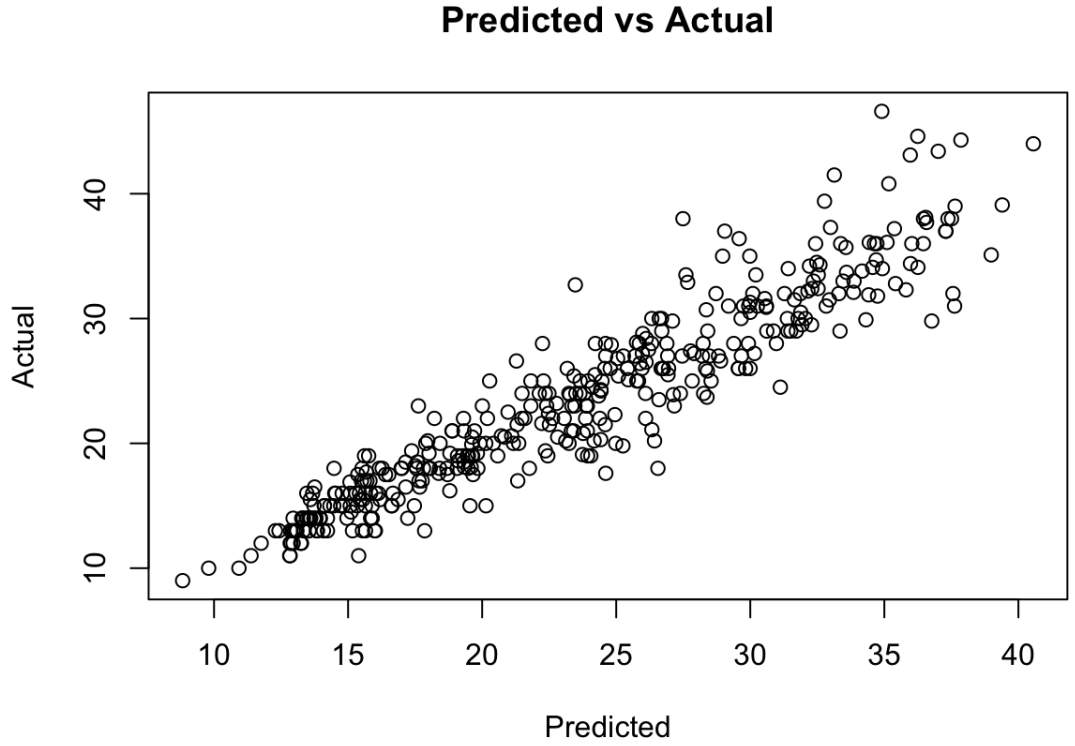
[1] "Leaps method produces the best model"

Lars Plots:




lasso.str$Cp
20
50
200
1000
5000
0
5
10
15
20
25
lasso.str$df


Cross-Validated MSE
0
100
300
500
0.0
0.2
0.4
0.6
0.8
1.0
Fraction of final L1 norm



[1] "=========================="
[1] "USA MODEL COMPARISON"
[1] "=========================="
[1] "running leaps"
[1] 1
[1] "Press= 1048.0470934059"
[1] "MPSE= 4.53700040435454"
[1] "Cp= 7.78048836274067"

[1] 2
[1] "Press= 1048.65844889037"
[1] "MPSE= 4.53964696489337"
[1] "Cp= 8.64637819138494"

[1] 3
[1] "Press= 1054.61322959921"
[1] "MPSE= 4.5654252363602"
[1] "Cp= 8.77043440443421"

[1] 4
[1] "Press= 1055.16553663311"
[1] "MPSE= 4.56781617590091"
[1] "Cp= 8.96361960412543"

[1] 5
[1] "Press= 1055.09219803246"
[1] "MPSE= 4.58735738274981"
[1] "Cp= 9.23735674234135"

[1] 6
[1] "Press= 1052.84469395977"
[1] "MPSE= 4.57758562591204"
[1] "Cp= 9.24846890662906"

[1] 7
[1] "Press= 1059.48426862413"
[1] "MPSE= 4.60645334184402"
[1] "Cp= 9.2698427017892"

[1] 8
[1] "Press= 1053.95232070415"
[1] "MPSE= 4.54289793406962"
[1] "Cp= 9.27002018824754"

[1] 9
[1] "Press= 1056.49658699534"
[1] "MPSE= 4.59346342171886"
[1] "Cp= 9.35541259256857"

[1] 10
[1] "Press= 1057.99930536437"
[1] "MPSE= 4.58008357300592"
[1] "Cp= 9.41433743620254"

[1] "==============="
[1] "Best Model"
[1] "Press= 1048.0470934059"
[1] "MPSE= 4.53700040435454"
[1] "Cp= 7.78048836274067"
[1] " Model Statistics "
Residual Standard Error=1.9563
R-Square=0.9126
F-statistic (df=13, 231)=185.6512
p-value=0

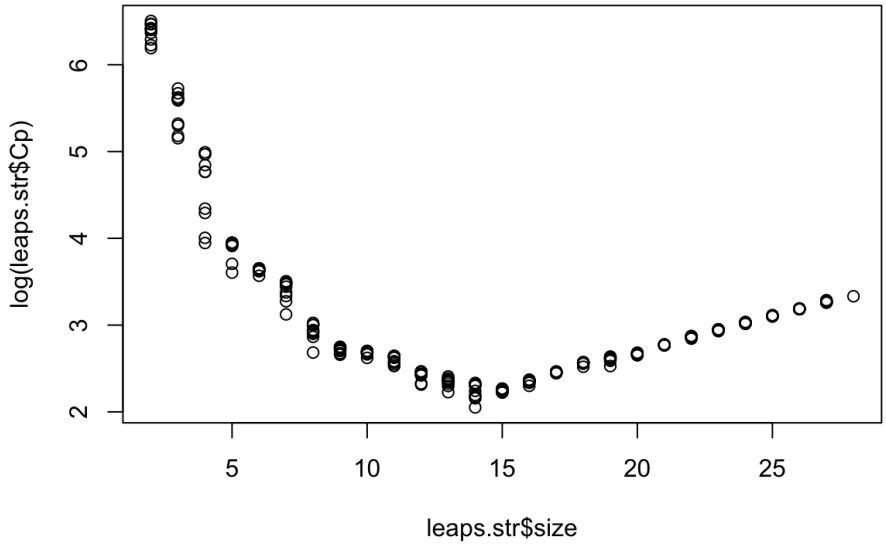
	Estimate	Std.Err	t-value	Pr(>|t|)
Intercept	234.4534	80.4294	2.9150	0.0039
displacement	-0.4656	0.1005	-4.6329	0.0000

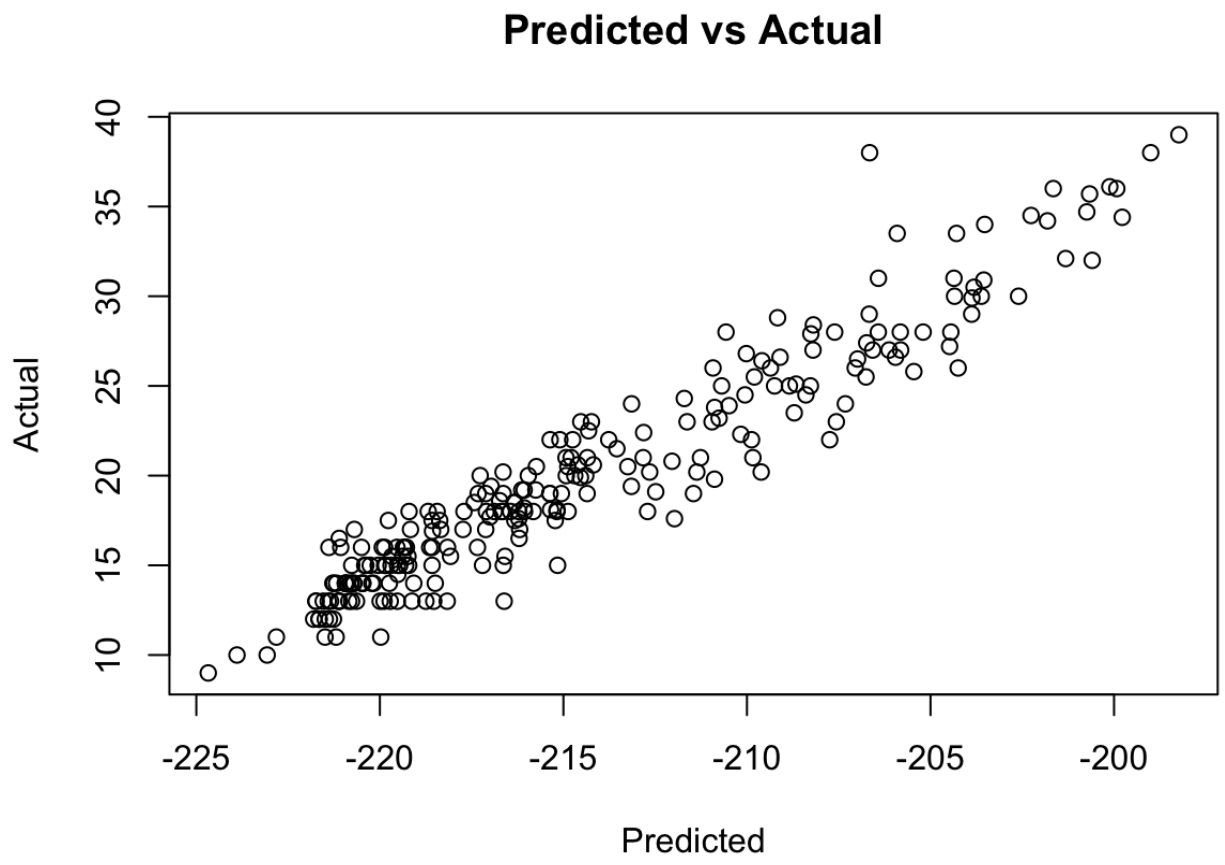
```
weight                    0.0331  0.0121  2.7322   0.0068
acceleration             -6.8417  1.6910 -4.0461   0.0001
year                     -4.5147  2.0424 -2.2105   0.0281
cylindersweight           0.0005  0.0003  1.8937   0.0595
cylindersacceleration     0.2319  0.0428  5.4203   0.0000
cylindersyear            -0.0693  0.0160 -4.3169   0.0000
displacementdisplacement  0.0001  0.0000  3.6849   0.0003
displacementyear          0.0053  0.0013  4.1804   0.0000
horsepoweracceleration   -0.0031  0.0007 -4.4225   0.0000
weightyear               -0.0005  0.0001 -3.5803   0.0004
accelerationyear          0.0717  0.0211  3.3970   0.0008
yearyear                  0.0324  0.0134  2.4230   0.0162
```

```
[1] "coeff without int"
          displacement                 weight
         -0.4656436060           0.0331472166
          acceleration                   year
         -6.8417304550          -4.5147448620
       cylindersweight    cylindersacceleration
          0.0005360273           0.2319231122
        cylindersyear displacementdisplacement
         -0.0692848340           0.0001228204
     displacementyear   horsepoweracceleration
          0.0052783043          -0.0030964437
           weightyear         accelerationyear
         -0.0005351085           0.0716677125
              yearyear
          0.0323750335
```

[1] "correlation 0.955326000142054"

Leaps Plots:



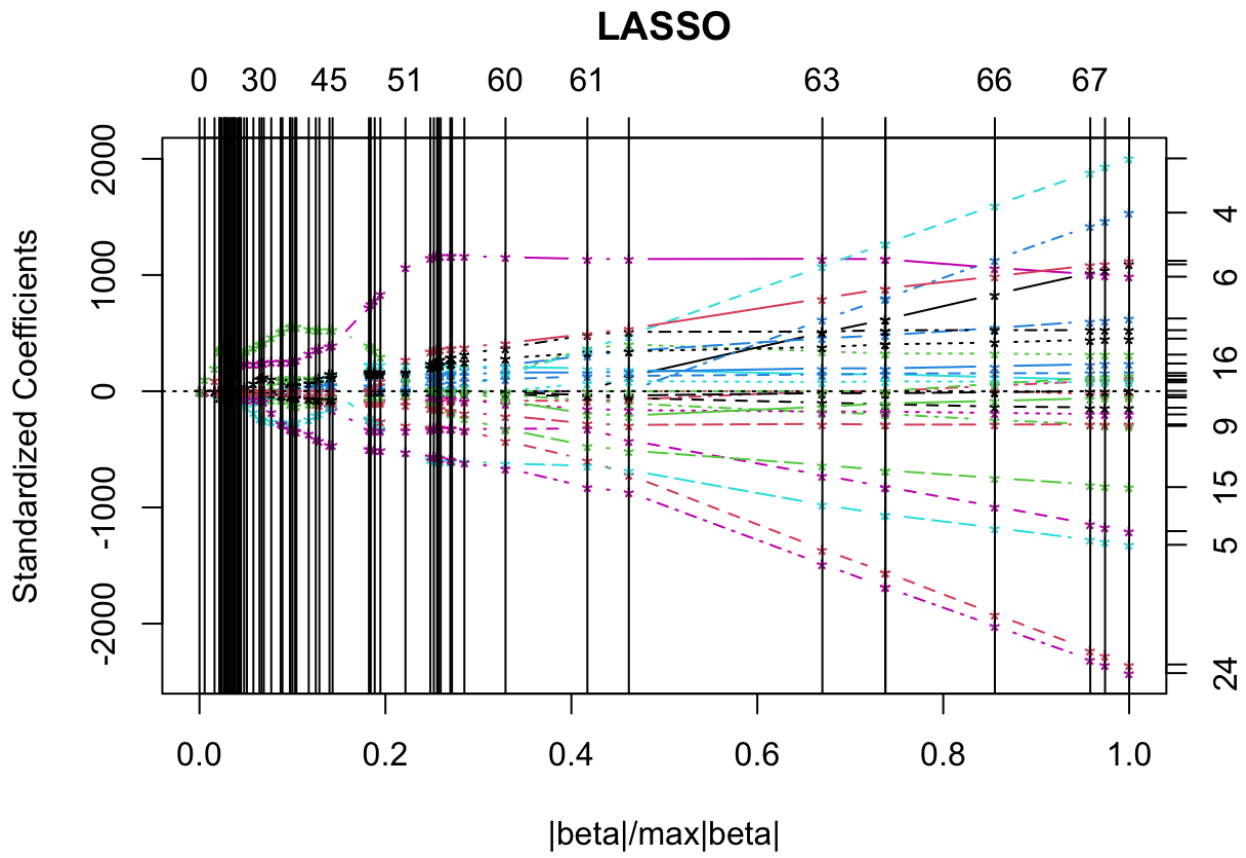


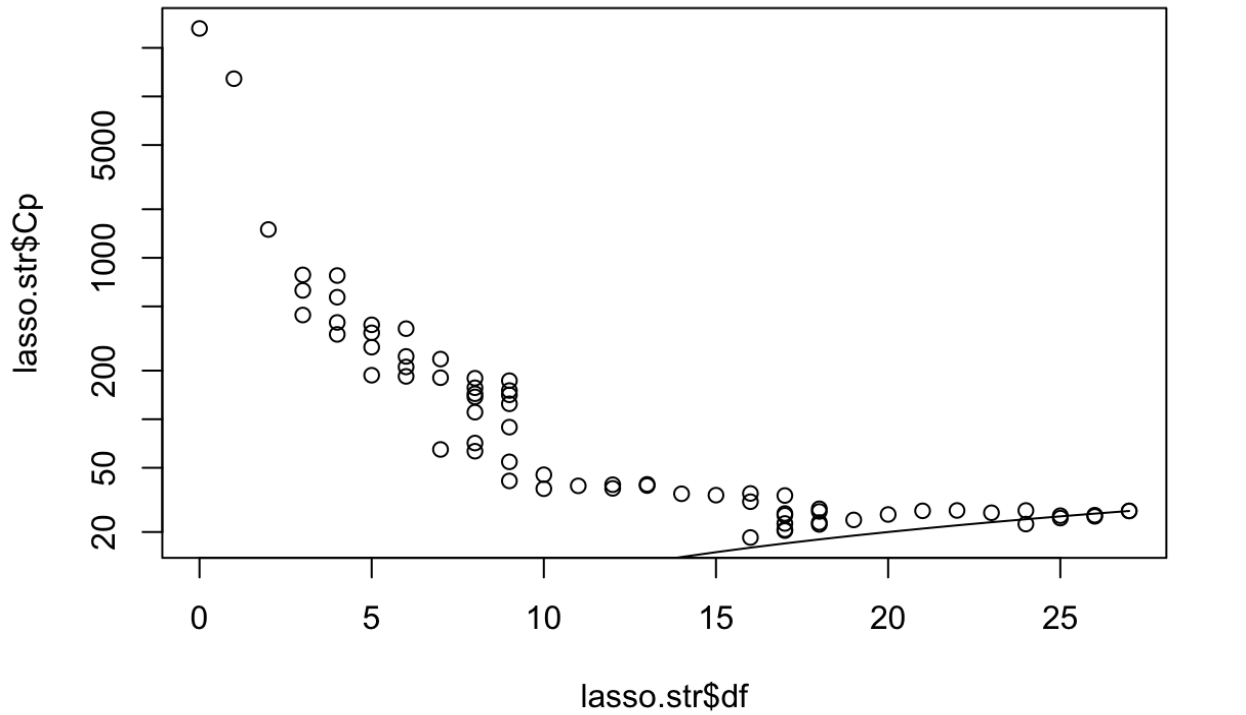
[1] "running lars"
[1] "beta.out"

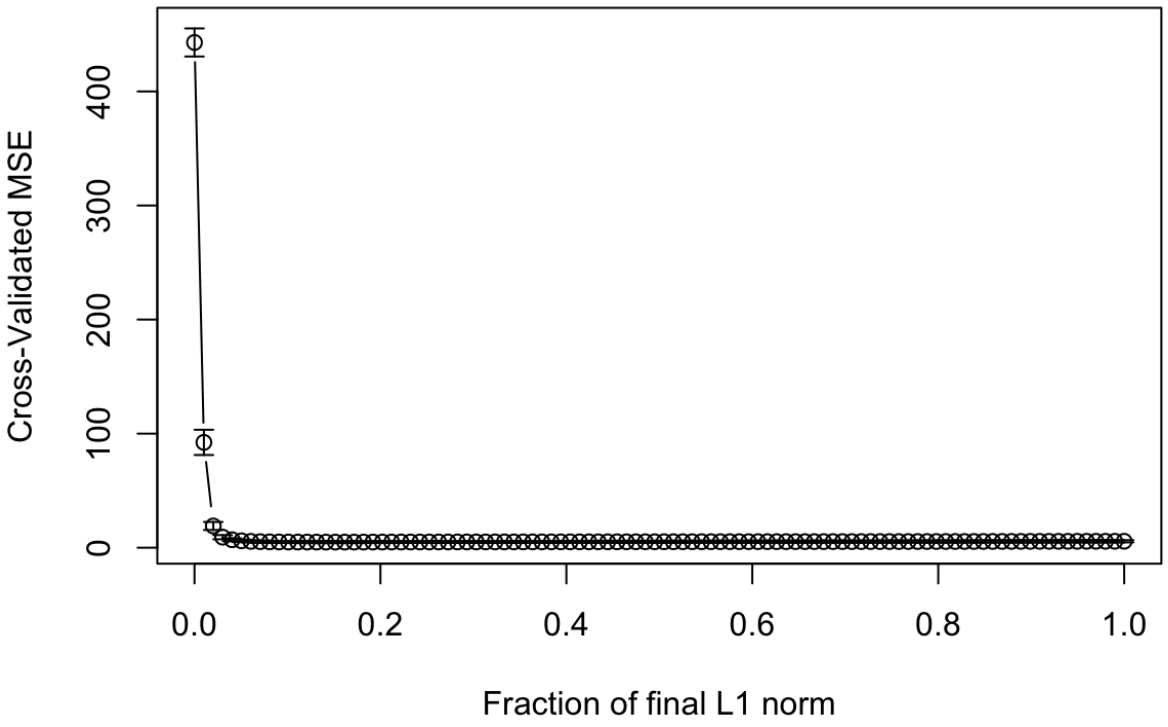
cylinders	displacement
0.000000e+00	-7.037797e-02
horsepower	weight
0.000000e+00	0.000000e+00
acceleration	year
-2.184701e+00	9.020811e-01
cylinderscylinders	cylindersdisplacement
0.000000e+00	0.000000e+00
cylindershorsepower	cylindersweight
0.000000e+00	3.692864e-04
cylindersacceleration	cylindersyear
1.527216e-01	-4.406635e-02
displacementdisplacement	displacementhorsepower
1.198031e-04	-1.134540e-04
displacementweight	displacementacceleration
0.000000e+00	1.370534e-03
displacementyear	horsepowerhorsepower
0.000000e+00	-4.724491e-05
horsepowerweight	horsepoweracceleration
2.112690e-05	-4.113511e-03

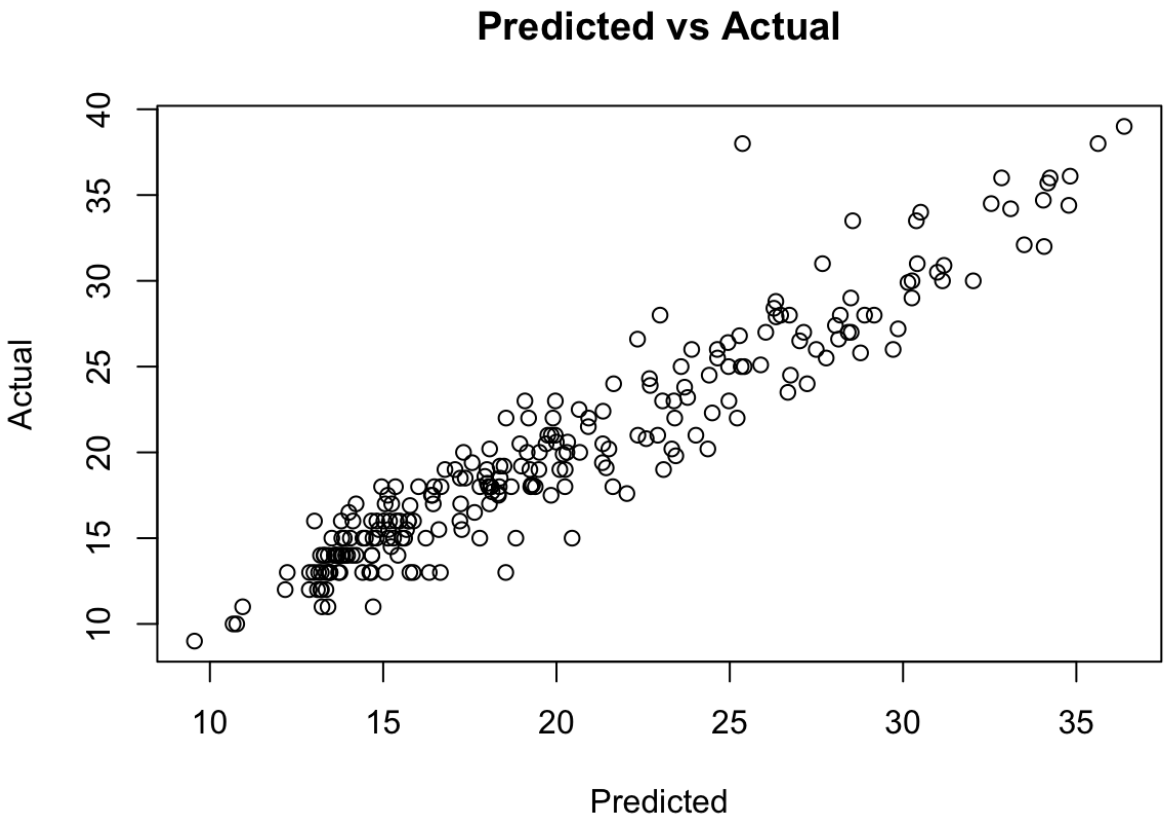
```
         horsepoweryear           weightweight
          -1.622748e-04           8.867796e-08
      weightacceleration             weightyear
           0.000000e+00          -1.295372e-04
accelerationacceleration       accelerationyear
           0.000000e+00           1.500302e-02
             yearyear
           1.026810e-03
[1] "Ind.out"
              cylinders             displacement
                  FALSE                     TRUE
             horsepower                   weight
                  FALSE                    FALSE
           acceleration                     year
                   TRUE                     TRUE
     cylinderscylinders     cylindersdisplacement
                  FALSE                    FALSE
    cylindershorsepower          cylindersweight
                  FALSE                     TRUE
  cylindersacceleration           cylindersyear
                   TRUE                     TRUE
displacementdisplacement    displacementhorsepower
                   TRUE                     TRUE
     displacementweight displacementacceleration
                  FALSE                     TRUE
       displacementyear      horsepowerhorsepower
                  FALSE                     TRUE
       horsepowerweight    horsepoweracceleration
                   TRUE                     TRUE
        horsepoweryear           weightweight
                   TRUE                     TRUE
     weightacceleration             weightyear
                  FALSE                     TRUE
accelerationacceleration       accelerationyear
                  FALSE                     TRUE
             yearyear
                   TRUE
[1] "Cp= 22.5953733104583"
[1] "correlation 0.951851510527563"
[1] "Leaps method produces the best model"
```

Lars Plots:


LASSO
0 30 45 51 60 61 63 66 67
Standardized Coefficients
2000
1000
0
-1000
-2000
4
6
16
9
15
5
24
0.0 0.2 0.4 0.6 0.8 1.0
|beta|/max|beta|


lasso.str$Cp
5000
1000
200
50
20
0 5 10 15 20 25
lasso.str$df


Cross-Validated MSE
0
100
200
300
400
0.0
0.2
0.4
0.6
0.8
1.0
Fraction of final L1 norm


Predicted vs Actual
Actual
10
15
20
25
30
35
40
10
15
20
25
30
35
Predicted

```
[1] "==========================="
[1] "Japan MODEL COMPARISON"
[1] "==========================="
[1] "running leaps"
[1] 1
[1] "Press= 751.265259710616"
[1] "MPSE= 10.434239718203"
[1] "Cp= 5.23147464526618"

[1] 2
[1] "Press= 754.372770620213"
[1] "MPSE= 10.4773995919474"
[1] "Cp= 5.3590981457485"

[1] 3
[1] "Press= 758.10468631658"
[1] "MPSE= 10.3849957029668"
[1] "Cp= 5.45344203020603"

[1] 4
[1] "Press= 770.558472031847"
[1] "MPSE= 10.7022010004423"
[1] "Cp= 5.71752005474353"

[1] 5
[1] "Press= 761.410121741233"
[1] "MPSE= 10.5751405797394"
[1] "Cp= 5.77598558661691"

[1] 6
[1] "Press= 760.556979629632"
[1] "MPSE= 10.5632913837449"
[1] "Cp= 5.8090825656937"

[1] 7
[1] "Press= 761.914359179202"
[1] "MPSE= 10.8844908454172"
[1] "Cp= 5.81746391405915"

[1] 8
[1] "Press= 770.048156329502"
[1] "MPSE= 11.1601182076739"
[1] "Cp= 5.81752798327379"
```

[1] 9
[1] "Press= 766.974898978378"
[1] "MPSE= 10.8024633658926"
[1] "Cp= 5.82483546102829"

[1] 10
[1] "Press= 760.996000511971"
[1] "MPSE= 10.4246027467393"
[1] "Cp= 5.84979686732029"

[1] "==========="
[1] "Best Model"
[1] "Press= 751.265259710616"
[1] "MPSE= 10.434239718203"
[1] "Cp= 5.23147464526618"
[1] " Model Statistics "
Residual Standard Error=2.9801
R-Square=0.779
F-statistic (df=6, 72)=42.2916
p-value=0

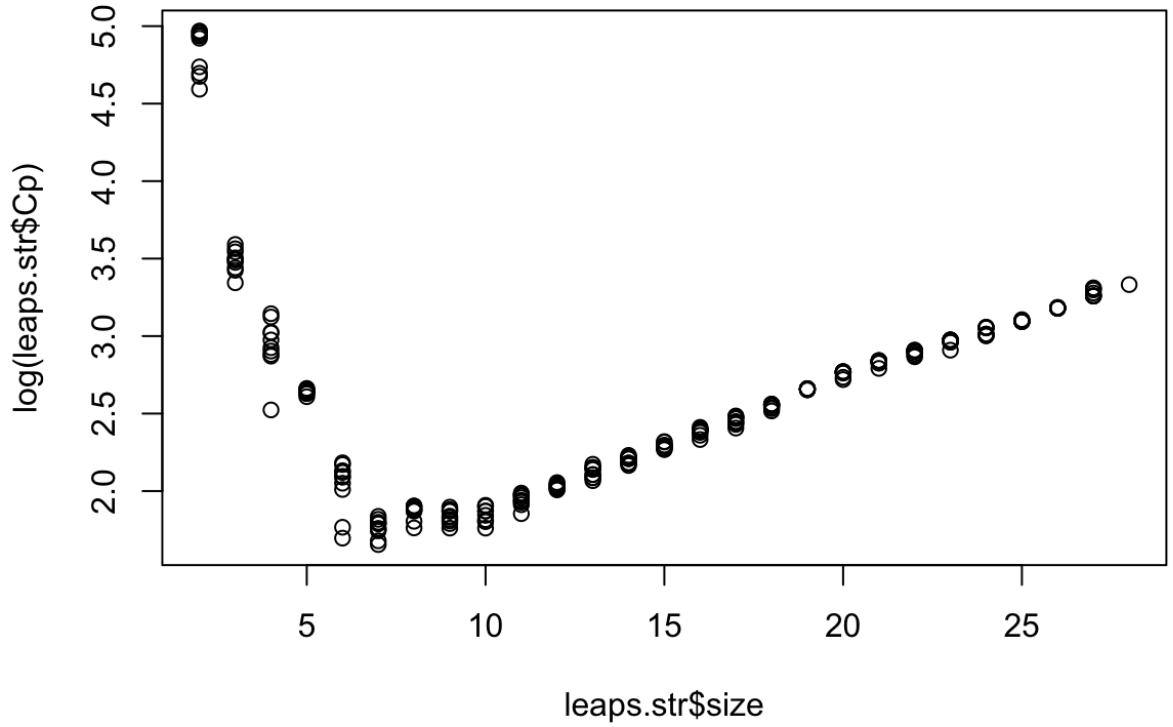
	Estimate	Std.Err	t-value	Pr(>|t|)
Intercept	-114.2973	20.8043	-5.4939	0
cylinders	54.2106	8.3938	6.4584	0
cylinderscylinders	-5.5297	0.8663	-6.3828	0
displacementacceleration	-0.0110	0.0024	-4.5164	0
horsepoweracceleration	-0.0267	0.0032	-8.2797	0
horsepoweryear	0.0048	0.0007	6.9069	0
accelerationyear	0.0314	0.0036	8.6041	0

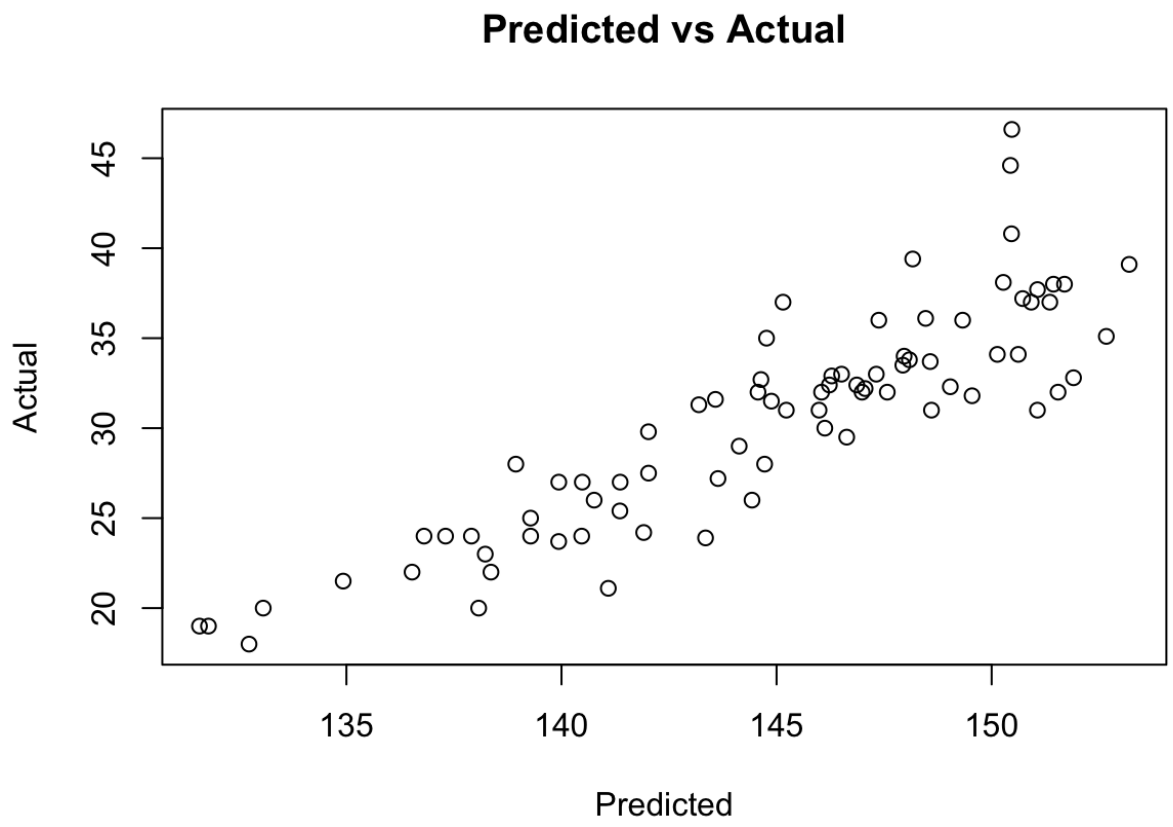
[1] "coeff without int"

cylinders	cylinderscylinders
54.210604617	-5.529733425
displacementacceleration	**horsepoweracceleration**
-0.010994889	-0.026658461
horsepoweryear	**accelerationyear**
0.004829747	0.031375884

[1] "correlation 0.882593532687259"

Leaps Plots:


log(leaps.str$Cp)
2.0
2.5
3.0
3.5
4.0
4.5
5.0
5
10
15
20
25
leaps.str$size


Predicted vs Actual
Actual
20
25
30
35
40
45
135
140
145
150
Predicted

```
[1] "running lars"
[1] "beta.out"
            cylinders          displacement
         3.212834e+00          0.000000e+00
          horsepower                weight
         0.000000e+00          0.000000e+00
         acceleration                  year
         0.000000e+00          0.000000e+00
   cylinderscylinders   cylindersdisplacement
        -8.019438e-01         -6.084117e-02
  cylindershorsepower        cylindersweight
         2.943885e-02          0.000000e+00
 cylindersacceleration          cylindersyear
         0.000000e+00          1.290488e-01
displacementdisplacement  displacementhorsepower
         0.000000e+00          2.033090e-03
   displacementweight displacementacceleration
         0.000000e+00         -4.889455e-03
     displacementyear    horsepowerhorsepower
         1.258454e-03         -3.801437e-04
     horsepowerweight  horsepoweracceleration
        -7.945983e-05         -1.069959e-02
       horsepoweryear          weightweight
         0.000000e+00          0.000000e+00
   weightacceleration            weightyear
         0.000000e+00          1.989211e-05
accelerationacceleration       accelerationyear
         3.002218e-02          0.000000e+00
           yearyear
         1.059581e-03
[1] “lnd.out"
            cylinders          displacement
                 TRUE                 FALSE
          horsepower                weight
                FALSE                 FALSE
         acceleration                  year
                FALSE                 FALSE
   cylinderscylinders   cylindersdisplacement
                 TRUE                  TRUE
  cylindershorsepower        cylindersweight
                 TRUE                 FALSE
 cylindersacceleration          cylindersyear
                FALSE                  TRUE
displacementdisplacement  displacementhorsepower
```

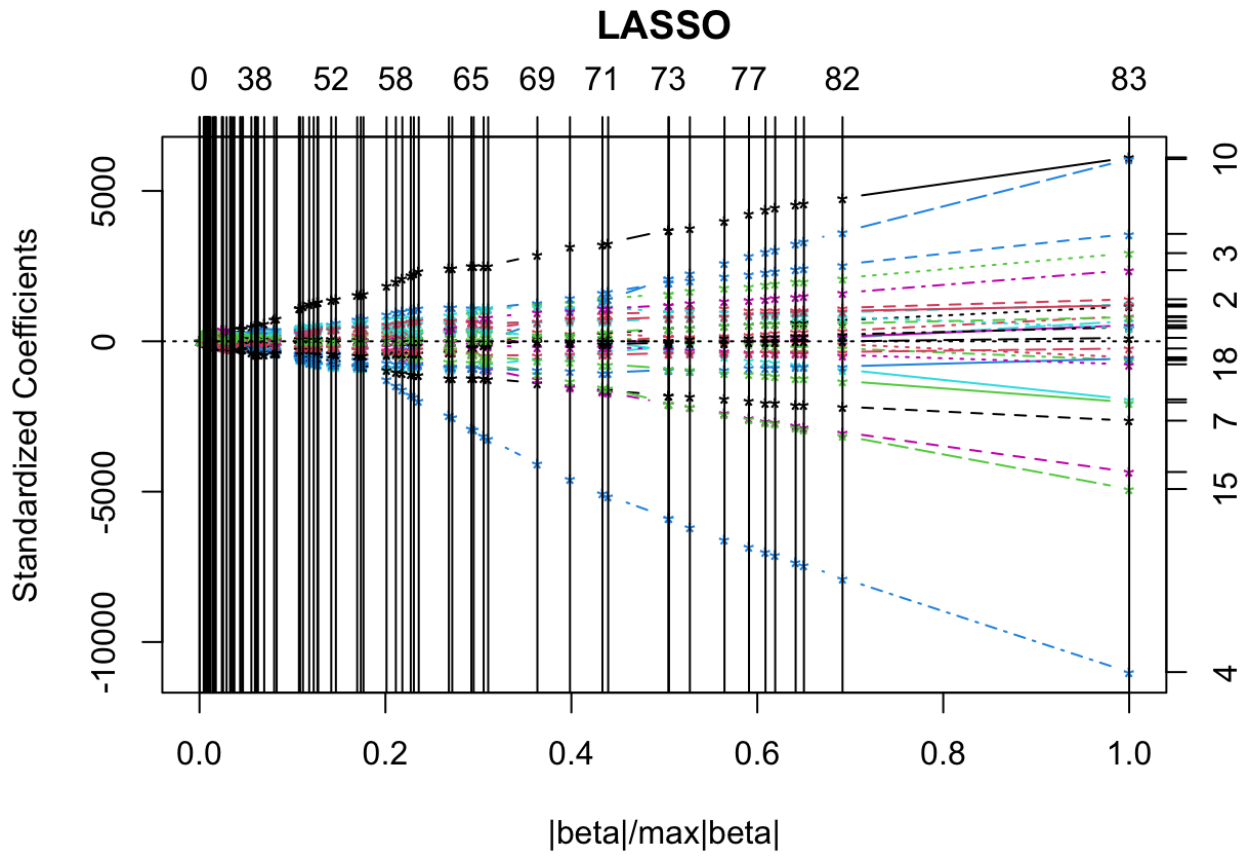
FALSE TRUE
displacementweight displacementacceleration
FALSE TRUE
displacementyear horsepowerhorsepower
TRUE TRUE
horsepowerweight horsepoweracceleration
TRUE TRUE
horsepoweryear weightweight
FALSE FALSE
weightacceleration weightyear
FALSE TRUE
accelerationacceleration accelerationyear
TRUE FALSE
yearyear
TRUE

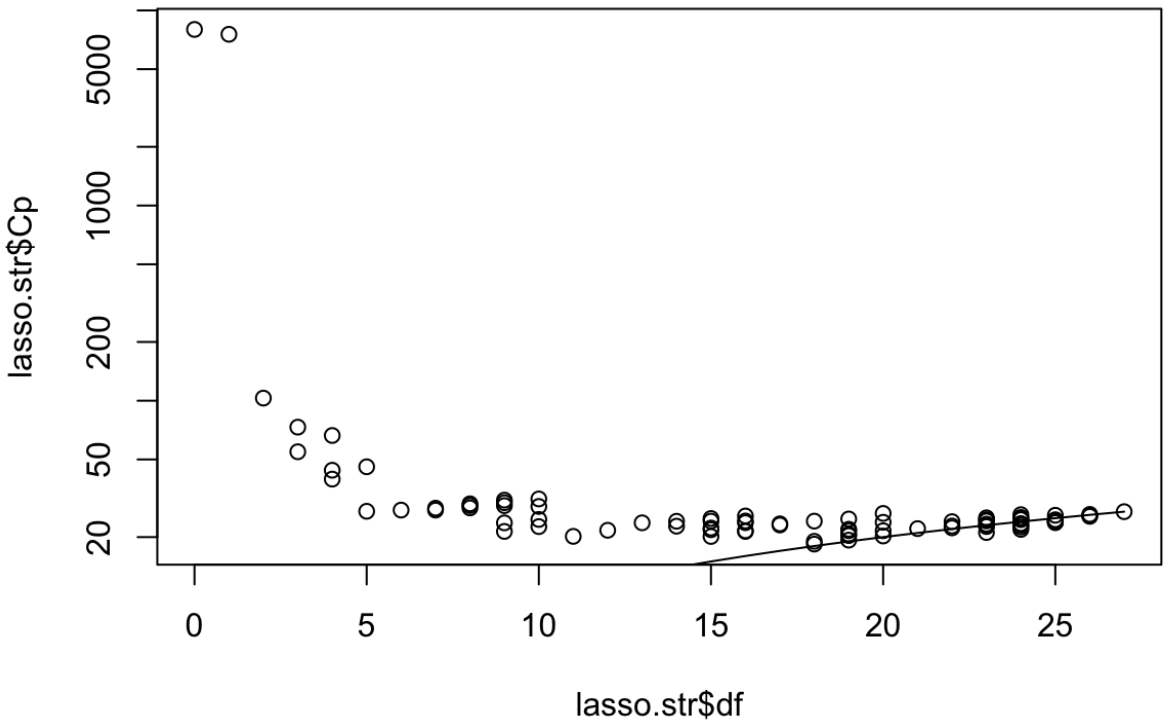
[1] "Cp= 24.1105712897532"

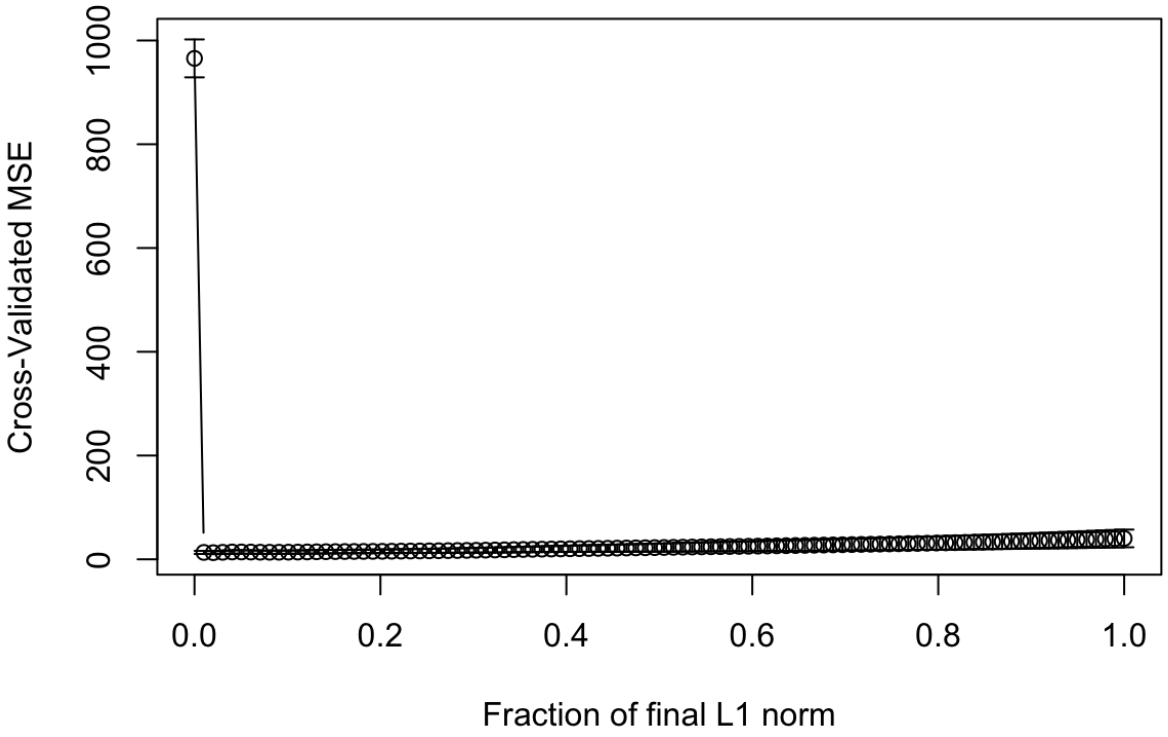
[1] "correlation 0.869160151428529"

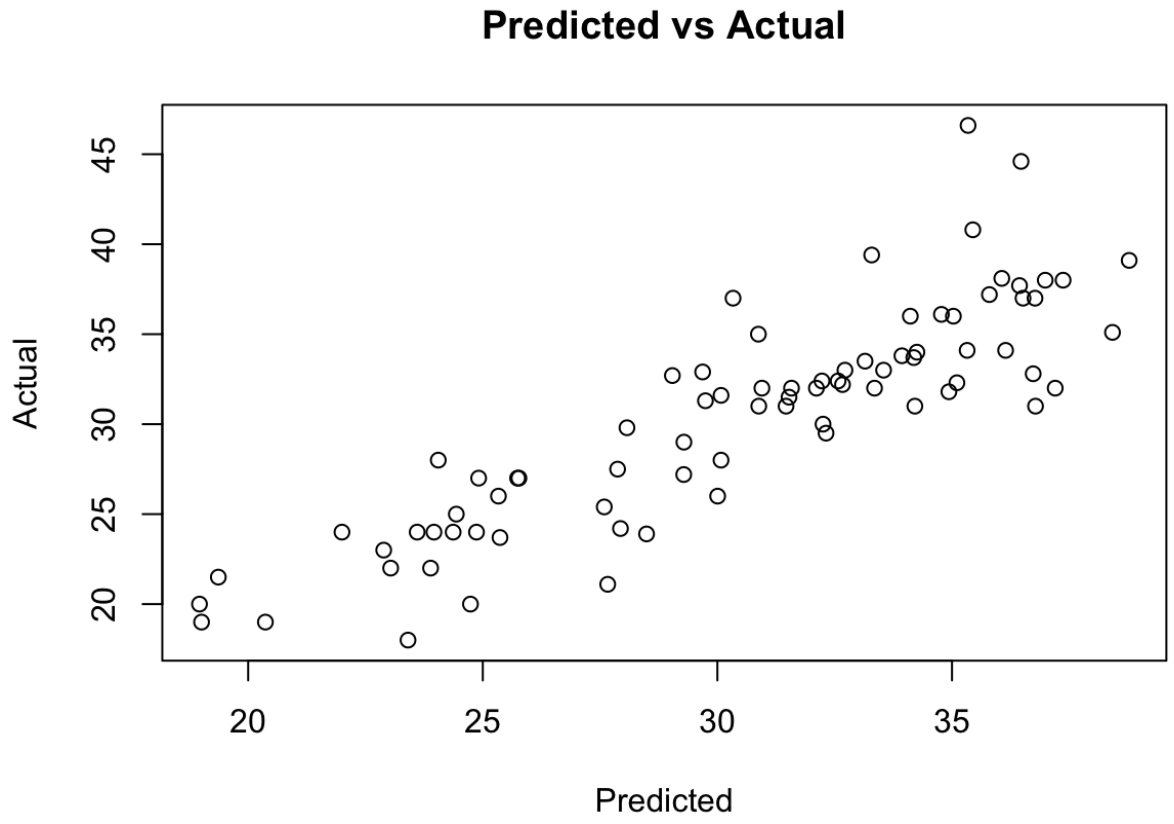
[1] "Leaps method produces the best model"

Lars Plots:




lasso.str$Cp
20
50
200
1000
5000
0
5
10
15
20
25
lasso.str$df


Cross-Validated MSE
0
200
400
600
800
1000
0.0
0.2
0.4
0.6
0.8
1.0
Fraction of final L1 norm



[1] "=========================="
[1] "Germany MODEL COMPARISON"
[1] "=========================="
[1] "running leaps"
[1] 1
[1] "Press= 528.995880853274"
[1] "MPSE= 9.28062948865393"
[1] "Cp= -0.21333031676"

[1] 2
[1] "Press= 515.885936066794"
[1] "MPSE= 9.05063045731217"
[1] "Cp= -0.210563636704251"

[1] 3
[1] "Press= 527.273544760542"
[1] "MPSE= 9.25041306597442"
[1] "Cp= -0.169169471838316"

[1] 4
[1] "Press= 509.335214833983"
[1] "MPSE= 9.09527169346398"
[1] "Cp= -0.147365890956877"

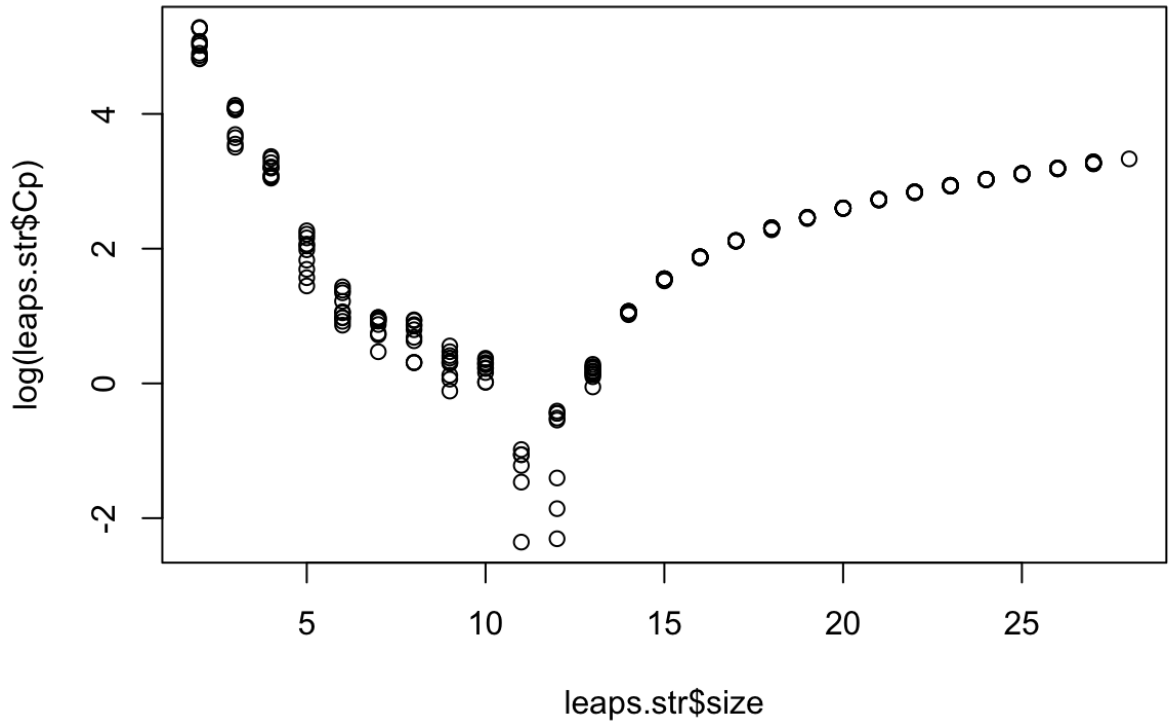
```
[1] 5
[1] "Press= 510.16400715496"
[1] "MPSE= 9.11007155633858"
[1] "Cp= -0.115670169533793"

[1] 6
[1] "Press= 516.565843435566"
[1] "MPSE= 9.06255865676432"
[1] "Cp= -0.106299573339051"

[1] 7
[1] "Press= 540.792070135161"
[1] "MPSE= 9.48758017780984"
[1] "Cp= 0.0948726505992283"

[1] 8
[1] "Press= 503.294412378813"
[1] "MPSE= 8.98740022105023"
[1] "Cp= 0.0996891582491983"

[1] 9
[1] "Press= 502.524096828047"
[1] "MPSE= 8.97364458621512"
[1] "Cp= 0.155815095804925"

[1] 10
[1] "Press= 528.698221379031"
[1] "MPSE= 9.27540739261458"
[1] "Cp= 0.231083849738766"

[1] "==============="
[1] "Best Model"
[1] "Press= 502.524096828047"
[1] "MPSE= 8.97364458621512"
[1] "Cp= 0.155815095804925"
[1] " Model Statistics "
Residual Standard Error=2.5539
R-Square=0.8741
F-statistic (df=11, 56)=35.3441
p-value=0

                Estimate  Std.Err t-value Pr(>|t|)
Intercept        475.0403 201.9493  2.3523   0.0222
```

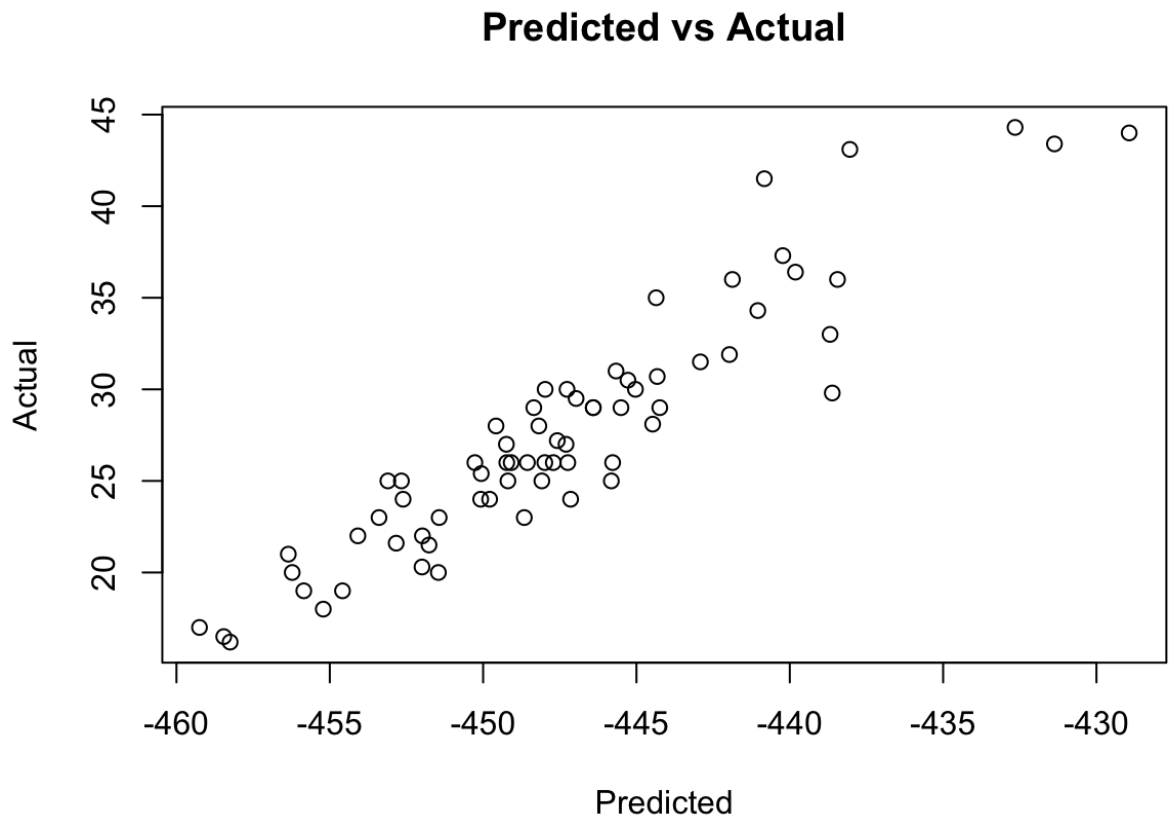
```
displacement           -0.1849    0.0751 -2.4615    0.0169
horsepower              2.2325    0.4779  4.6711    0.0000
weight                 -0.0744    0.0271 -2.7430    0.0082
year                  -12.2297    5.0397 -2.4266    0.0185
cylinderscylinders     -3.1334    1.0783 -2.9057    0.0052
cylindersweight         0.0097    0.0033  2.9475    0.0047
displacementhorsepower  0.0014    0.0009  1.4846    0.1433
horsepoweryear         -0.0330    0.0065 -5.1133    0.0000
weightweight            0.0000    0.0000 -3.0738    0.0033
weightyear              0.0009    0.0003  2.8431    0.0062
yearyear                0.0906    0.0324  2.7959    0.0071
```

```
[1] "coeff without int"
        displacement           horsepower               weight
       -1.849315e-01         2.232471e+00        -7.444464e-02
                year     cylinderscylinders        cylindersweight
       -1.222970e+01        -3.133354e+00         9.740884e-03
displacementhorsepower        horsepoweryear          weightweight
        1.388639e-03        -3.300719e-02        -7.352173e-06
          weightyear             yearyear
        9.139102e-04         9.056092e-02
```

[1] "correlation 0.934931220793042"

Leaps Plots:



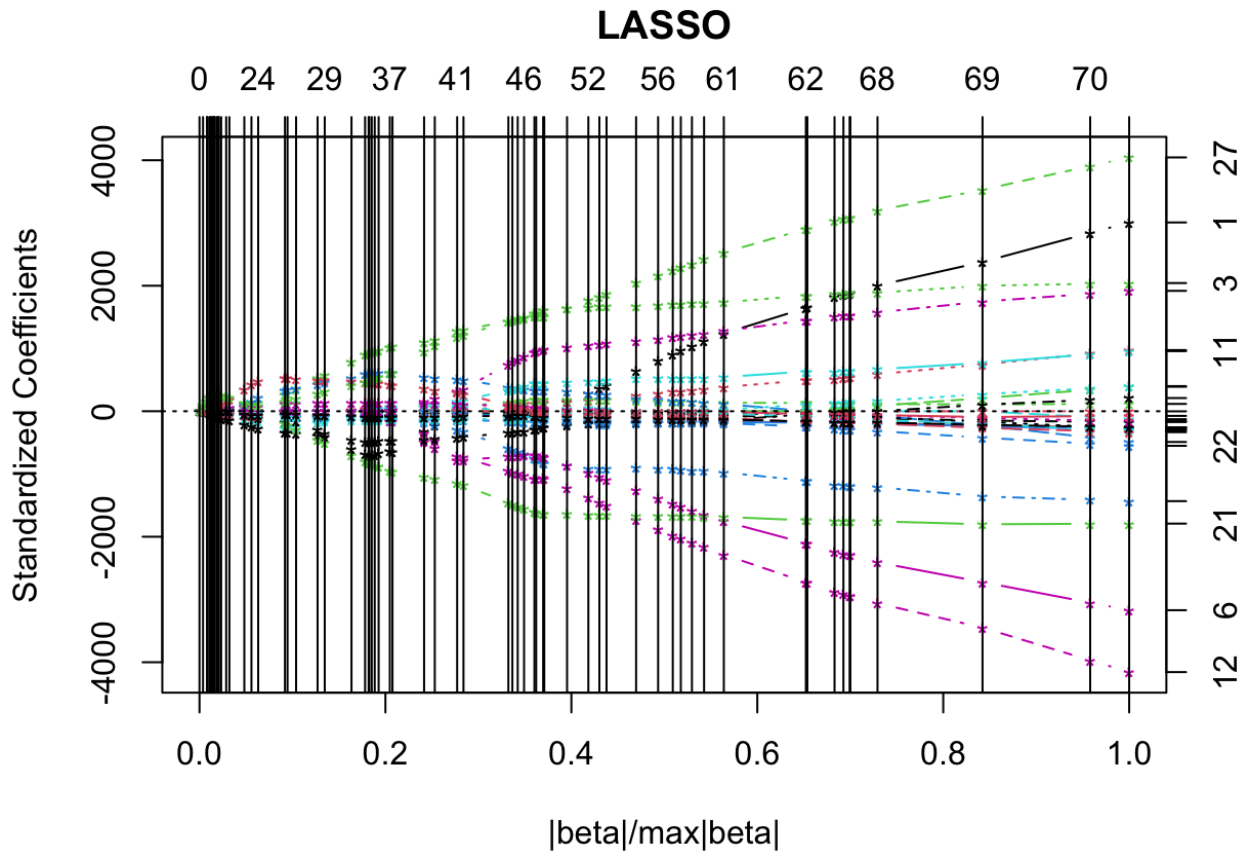


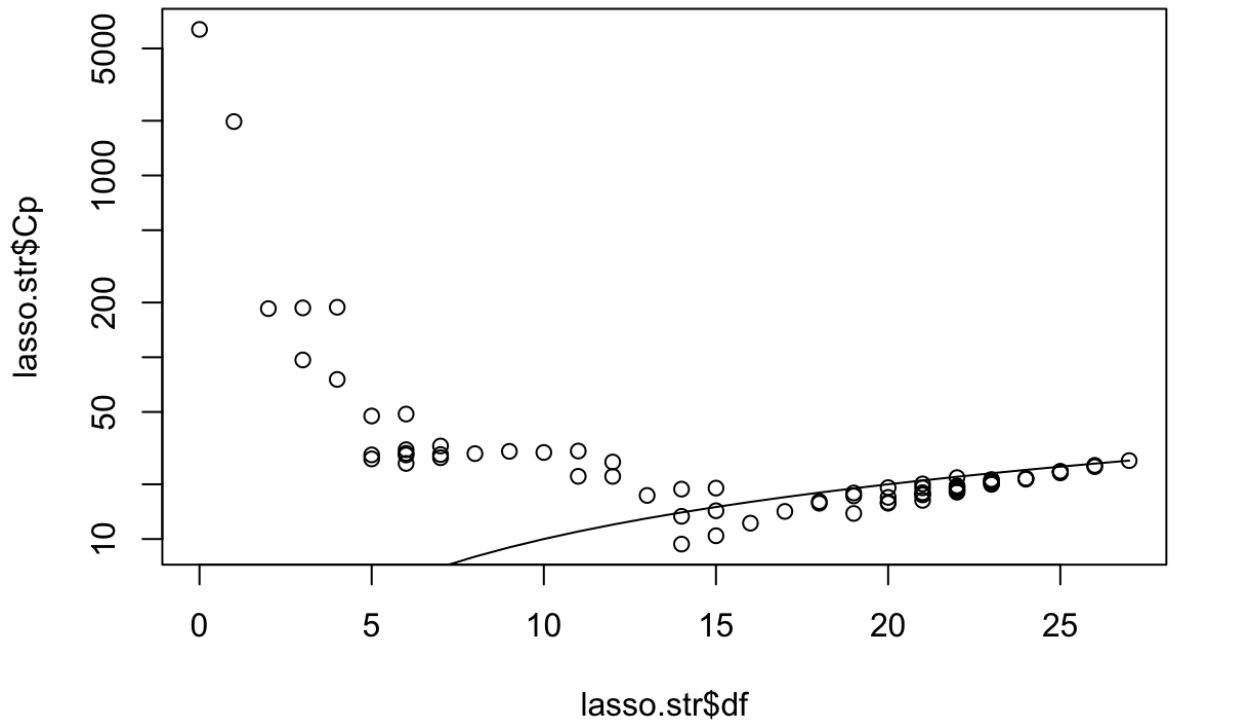
[1] "running lars"
[1] "beta.out"

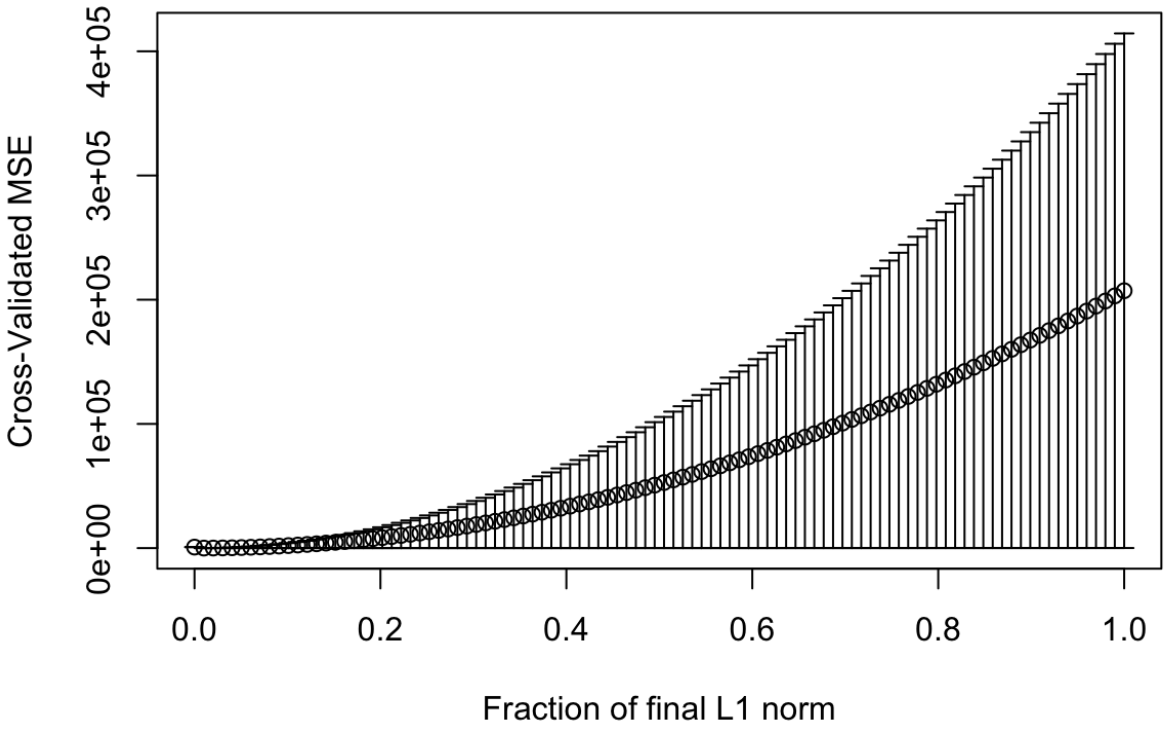
cylinders	displacement
0.000000e+00	0.000000e+00
horsepower	weight
7.643294e-02	0.000000e+00
acceleration	year
-5.112903e-01	0.000000e+00
cylinderscylinders	cylindersdisplacement
-7.432713e-02	0.000000e+00
cylindershorsepower	cylindersweight
-6.879455e-03	5.315393e-04
cylindersacceleration	cylindersyear
0.000000e+00	0.000000e+00
displacementdisplacement	displacementhorsepower
0.000000e+00	0.000000e+00
displacementweight	displacementacceleration
0.000000e+00	-3.167234e-03
displacementyear	horsepowerhorsepower
0.000000e+00	7.802120e-04
horsepowerweight	horsepoweracceleration
-7.883678e-05	-3.644475e-03

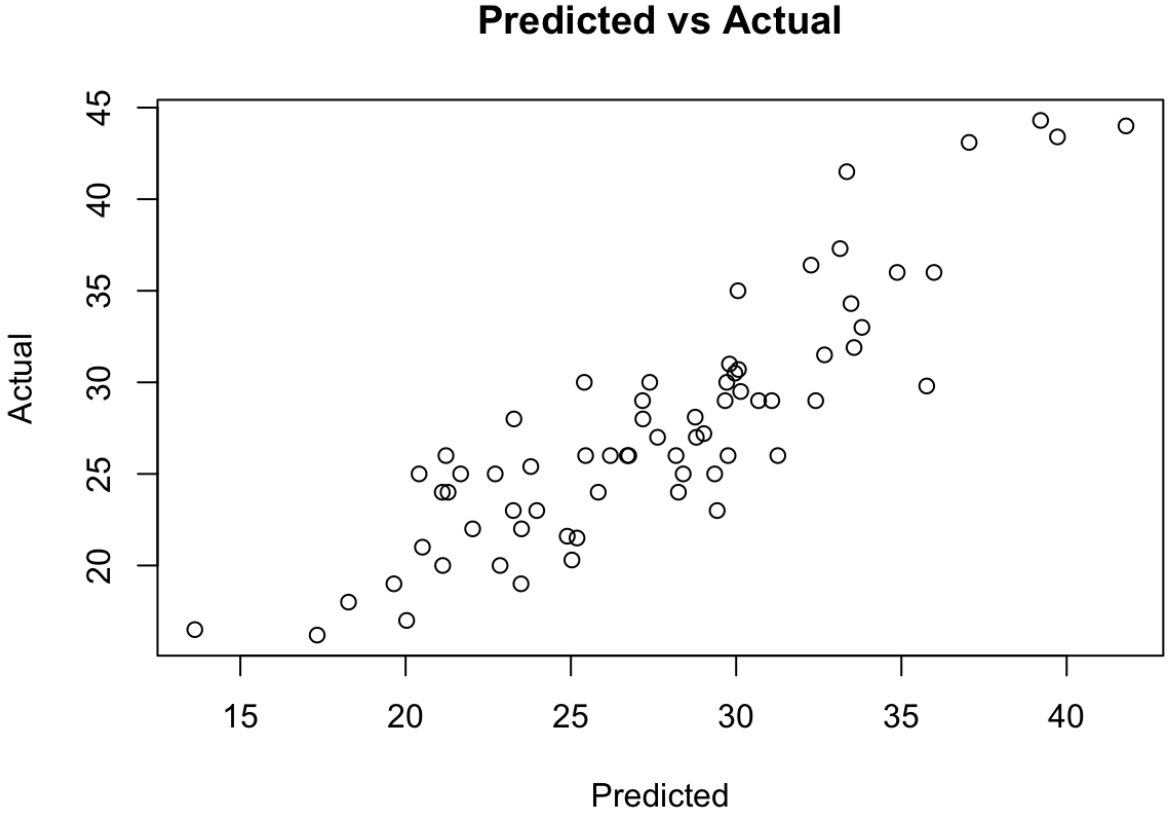
```
        horsepoweryear          weightweight
          0.000000e+00          0.000000e+00
     weightacceleration          weightyear
          0.000000e+00          0.000000e+00
accelerationacceleration       accelerationyear
          0.000000e+00          1.792401e-02
            yearyear
          4.636311e-03
[1] "Ind.out"
            cylinders           displacement
                FALSE                 FALSE
           horsepower                weight
                TRUE                 FALSE
         acceleration                  year
                TRUE                 FALSE
    cylinderscylinders    cylindersdisplacement
                TRUE                 FALSE
   cylindershorsepower         cylindersweight
                TRUE                  TRUE
 cylindersacceleration          cylindersyear
                FALSE                FALSE
displacementdisplacement   displacementhorsepower
                FALSE                FALSE
    displacementweight displacementacceleration
                FALSE                 TRUE
      displacementyear     horsepowerhorsepower
                FALSE                 TRUE
      horsepowerweight   horsepoweracceleration
                TRUE                  TRUE
       horsepoweryear          weightweight
                FALSE                FALSE
    weightacceleration          weightyear
                FALSE                FALSE
accelerationacceleration       accelerationyear
                FALSE                 TRUE
            yearyear
                TRUE
[1] "Cp= 30.4810739791194"
[1] "correlation 0.881100660239298"
[1] "Leaps method produces the best model"
```

Lars Plots:


LASSO
0 24 29 37 41 46 52 56 61 62 68 69 70
Standardized Coefficients
4000
2000
0
-2000
-4000
0.0 0.2 0.4 0.6 0.8 1.0
|beta|/max|beta|
27 1 3 11 22 21 6 12


lasso.str$Cp
5000
1000
200
50
10
0 5 10 15 20 25
lasso.str$df


Cross-Validated MSE
0e+00
1e+05
2e+05
3e+05
4e+05
0.0
0.2
0.4
0.6
0.8
1.0
Fraction of final L1 norm


Predicted vs Actual
Actual
20
25
30
35
40
45
15
20
25
30
35
40
Predicted

Anything interesting you can pick up on what is going on in the different countries?

Germany had interesting plots, specifically the cross validated mse plot and the leaps cp vs leaps size plot, they both differed greatly in appearance from the other countries. The cross validated mse plot looked so different for germany because they all had one outlier at the first index of the list of cv mse values but germany didnt have that so we get a clear view of where in the list the lowest cv mse value is. Germany and Usa had leaps model predicted vs actual plots with negative intercepts since when we were calculating predicted we didn't include intercept so that could be a possible explanation. The auto leaps and lars models that included all countries had the least subsets removed to create the best model for regression.

Following is code edited/ran/commented:

```
matrix.2ndorder.make<-function(x, only.quad=F){
  x0<-x
  dimn<-dimnames(x)[[2]] #extract the names of the variables
  num.col<-length(x[1,]) # how many columns
  for(i in 1:num.col){
    # if we are doing all 2nd order
    if(!only.quad){
      for(j in i:num.col){
        x0<-cbind(x0,as.numeric(x[,i])*as.numeric(x[,j]))
        dimn<-c(dimn,paste(dimn[i],dimn[j],sep=""))
        #create interaction dimnames
      }
    }
    else{
      #in here only if doing only squared terms
      x0<-cbind(x0,as.numeric(x[,i])*as.numeric(x[,i]))
      dimn<-c(dimn,paste(dimn[i],"2",sep="")) # squared dimmension names
```

```
    }
  }
  dimnames(x0)[[2]]<-dimn
  x0
}
#sum of the absolute values of x
sumabs<-function(x){sum(abs(x))}
betanorm.lars<-function(str){
  v1<-apply(str$beta,1,sumabs)
  v1/max(v1)
}
regpluspress<-function(x,y){
  #least squares fit of x and y
  str<-lsfit(x,y)
  #calculates press statistic
  press<-sum((str$resid/(1-hat(x)))^2)
  #assigns press and coeff to attributes of str
  str$press<-press
  beta<-str$coefficients
  str$beta<-beta
  str
}
leaps.then.press<-function(xmat,yvec,ncheck=10,print.ls=F)
{
  #performs leaps on xmat and yvec
  leaps.str<-leaps(xmat,yvec)
```

```
# plots the size against corrosponding cps
plot(leaps.str$size, log(leaps.str$Cp))
#saves cpstatistics
z1<-leaps.str$Cp
#orders cp and saves the order
o1<-order(z1)
#saves ncheck models with respect to cp in order
matwhich<-(leaps.str$which[o1,])[1:ncheck,]
z2<-z1[o1][1:ncheck]
#initializes press vec to use in following code
pressvec<-NULL
#calculates the press for the ncheck best models with regard to cp
for(i in 1:ncheck){
  ls.str0<-regpluspress(xmat[,matwhich[i,]],yvec)
  print(i)
  print(paste("Press=",ls.str0$press))
  #saves each press value into press vec
  pressvec<-c(pressvec,ls.str0$press)
  parvec<-matwhich[i,]
  #number of parameters
  npar<-sum(parvec)
  #calculates the mean predicted square error
  print(paste("MPSE=",ls.str0$press/(length(yvec)-(npar+1))))
  print(paste("Cp=",z2[i]))
  if(print.ls){
    print(" Model Statistics ")
```

```
    ls.print(ls.str0)
  }
 }
 #saves the index of the model with the smallest press stat
 modin<-which.min(pressvec)
 ls.strB<-regpluspress(xmat[,matwhich[modin,]],yvec)
 print("===============")
 print("Best Model")
 print(paste("Press=",ls.strB$press))
 parvec<-matwhich[modin,]
 #number of parameters
 npar<-sum(parvec)
 #calculates the mean predicted square error
 print(paste("MPSE=",ls.strB$press/(length(yvec)-(npar+1))))
 print(paste("Cp=",z2[modin]))
 print(" Model Statistics ")
 ls.print(ls.strB)
 #predictive analysis
 print("coeff without int")
 print(ls.strB$beta[-1])
 #plots the predicted y hat against y vec
 plot(xmat[,matwhich[modin,]]%*%ls.strB$beta[-1],yvec,main="Predicted vs
Actual",xlab="Predicted",ylab="Actual")
 #calculates the correlation between predicted and actual
 cor<-cor(xmat[,matwhich[modin,]]%*%ls.strB$beta[-1],yvec)
 print(paste("correlation",cor))
```

```
  outlist1<-list(correlation=cor)
  outlist1
}

lars.select<-function(xmat,y,int=F,ncheck=10)
{
  #calculates the lars and stores
  lasso.str<-lars(xmat,y,intercept=int)
  #plots the lasso
  plot(lasso.str)
  #plots the lasso df and lasso cp to see which model has cp about equal to df
indicaticing a good model
  plot(lasso.str$df,lasso.str$Cp,log="y")
  lines(lasso.str$df,lasso.str$df)
  #plots the cross validated MSE with the fraction of final L1 norm
  cv.str<-cv.lars(xmat,y,plot.it=T,intercept=int)
  #orders cross validated mse stats and stores the order
  o1<-order(cv.str$cv)
  #stores the index of the smallest cv value
  mindex<-cv.str$index[o1][1]
  #extracts coeffecients from lars
  beta<-coef(lasso.str)
  #finds the sum of the absolute value of every row in the beta matrix
  index0<-apply(beta,1,sumabs)
  #divide each number by the max index and orders them
  index0<-index0/max(index0)
```

```
o1<-order(abs(index0-mindex))
#find minimum index of I1
I1<-(abs(index0-mindex)==min(abs(index0-mindex)))
n1<-length(beta[,1])
beta.out<-beta[I1,]
#orders using Cp if sumabs is zero
if(sum(abs(beta.out))==0){
  v1<-lasso.str$Cp
  o2<-order(v1)
  beta.out<-beta[o1[1:ncheck],]
}
Ind.out<-beta.out!=0
#outputs
print(beta.out)
print(Ind.out)
print(paste("Cp=",lasso.str$Cp[I1]))
#outlist<-list(beta.out=beta.out,ind.out=Ind.out,Cp=lasso.str$Cp[I1])
if(int){
  Int.out1<-mean(y)-mean(xmat%*%beta.out[i])
  print(beta.out)
  print(Ind.out)
  print(Ind.out1)
  print(paste("Cp=",lasso.str$Cp[I1]))
}
#plots predicted vs actual
plot(xmat%*%beta.out,y,main="Predicted vs Actual",xlab="Predicted",ylab="Actual")
```

```
  #calculates correlation
  cor<-cor(xmat%*%beta.out,y)
  print(paste("correlation",cor))
  outlist2<-list(correlation=cor)
  outlist2
}

bestmodelcomp<-function(xmat,y,run.leaps = T,run.lars=T)
{
  if(run.leaps)
  {
    print("running leaps")
    hi<-leaps.then.press(xmat,y)
    corleaps<-hi$correlation
  }
  if(run.lars)
  {
    print("running lars")
    lol<-lars.select(xmat,y)
    corlars<-lol$correlation
  }
  #compares correlations of both models if both are True
  if(run.leaps&&run.lars)
  {
    if (corleaps<corlars)
    {
```

```
      print("Lars method produces the best model")
    }
    else
    {
      print("Leaps method produces the best model")
    }
  }
}

library(ISLR)
library(lars)
library(leaps)

Auto$mpg <- as.numeric(Auto$mpg)

auto.mat<-as.matrix(Auto[,-9])
y.auto<-data.matrix(as.numeric(auto.mat[,1]))
x.auto<-data.matrix(auto.mat[,2:7])
x.auto2<-data.matrix(matrix.2ndorder.make(x.auto))

auto.mat.japan<-data.matrix(auto.mat[auto.mat[,8]==3,])
auto.mat.germany<-data.matrix(auto.mat[auto.mat[,8]==2,])
auto.mat.usa<-data.matrix(auto.mat[auto.mat[,8]==1,])

y.auto.japan<-data.matrix(auto.mat.japan[,1])
x.auto.japan<-data.matrix(auto.mat.japan[,2:7])
```

```
x.auto2.japan<-data.matrix(matrix.2ndorder.make(x.auto.japan))

y.auto.germany<-data.matrix(auto.mat.germany[,1])
x.auto.germany<-data.matrix(auto.mat.germany[,2:7])
x.auto2.germany<-data.matrix(matrix.2ndorder.make(x.auto.germany))

y.auto.usa<-data.matrix(auto.mat.usa[,1])
x.auto.usa<-data.matrix(auto.mat.usa[,2:7])
x.auto2.usa<-data.matrix(matrix.2ndorder.make(x.auto.usa))
print("==========================")
print("AUTO (ALL COUNTRIES) MODEL COMPARISON")
print("==========================")
bestmodelcomp(x.auto2,y.auto)
print("==========================")
print("USA MODEL COMPARISON")
print("==========================")
bestmodelcomp(x.auto2.usa,y.auto.usa)
print("==========================")
print("Japan MODEL COMPARISON")
print("==========================")
bestmodelcomp(x.auto2.japan,y.auto.japan)
print("==========================")
print("Germany MODEL COMPARISON")
print("==========================")
bestmodelcomp(x.auto2.germany,y.auto.germany)
```