Homework 6

Jeff Tilton

9/29/2018

Question 9.1

Using the same crime data set uscrime.txt as in Question 8.2, apply Principal Component Analysis and then create a regression model using the first few principal components. Specify your new model in terms of the original variables (not the principal components), and compare its quality to that of your solution to Question 8.2. You can use the R function prcomp for PCA. (Note that to first scale the data, you can include scale. = TRUE to scale as part of the PCA function. Don't forget that, to make a prediction for the new city, you'll need to unscale the coefficients (i.e., do the scaling calculation in reverse)!)

Goals

1. Perform Principal Component Analysis on Crime data
2. Build a linear model with components
3. Unscale coefficients of model
4. Compare with previous homework results
5. Predict crime for new city

Perform Principal Component Analysis on Crime data

Method

Use `prcomp(data, center = TRUE,  scale = TRUE)`, where center shifts variables to be zero centered and scale scales the variables to have unit variances before analysis. I will eliminate the inidcator variable for a southern state predictor because it is binary and pca works well on data with a high variance.

```
## Importance of components:
##                           PC1    PC2    PC3     PC4     PC5     PC6
## Standard deviation     2.3262 1.6513 1.4158 1.03670 0.96745 0.74049
## Proportion of Variance 0.3865 0.1948 0.1432 0.07677 0.06685 0.03917
## Cumulative Proportion  0.3865 0.5813 0.7244 0.80121 0.86806 0.90723
##                           PC7     PC8    PC9    PC10    PC11    PC12
## Standard deviation     0.56415 0.54675 0.4475 0.42747 0.35945 0.31852
## Proportion of Variance 0.02273 0.02135 0.0143 0.01305 0.00923 0.00725
## Cumulative Proportion  0.92996 0.95132 0.9656 0.97867 0.98790 0.99515
##                           PC13    PC14
## Standard deviation     0.25159 0.06802
## Proportion of Variance 0.00452 0.00033
## Cumulative Proportion  0.99967 1.00000
```

The `prcomp` summary output shows that the components have been ranked by variance. The proportion of variance can be interpreted as the percentage of variance that is explained by that component. Therefore, cumulative proportion sums to one as shown.

Build a linear model with components

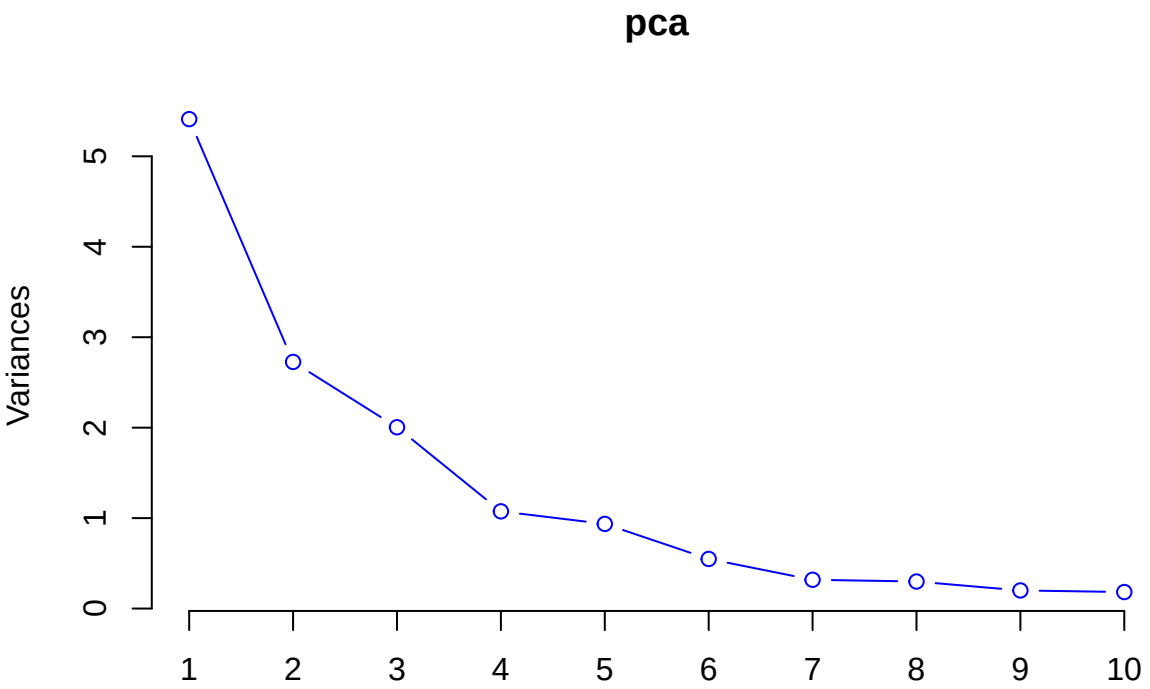
Method

1. Choose components to use
2. Build models using `lm` function.

Choose components

I will build 2 models to compare with the 2 homework 5 models I did. The first model will use all components to compare with the homework 5 model that used all predictors. The second model will use a subset of components to compare with the homework 5 model that limited the number of predictors by using the p-values. I will decide which components to use by using an elbow plot.

Model 2 Elbow Plot



I will use the first 7 principal components for the second model because that is where the elbow plot levels out and is where 93% of the variance is explained.

Unscale coefficients of model

Method

1. Convert the beta values to alphas
2. Unscale the alphas

Convert the beta values to alphas

Beta values are the principal component coefficients from the linear regression model. Thesee coefficients are related to the predictor coefficients (alphas), by

$$a_j = \sum_{k=1} b_k v_{jk}$$

,

Where v are the eigenvector

$$v_{jk} = X^T X$$

where X is the uscrime dataset. The eigenvectors can be retrieved from the `prcomp` output as `pca$rotation`. Scaled alphas are computed with matrix multiplication.

Unscale the alphas

Values are scaled by

$$X_{ij,scaled} = (X_{ij,} - \mu_{ij})/\sigma_j$$

Therefore

$$a_{scaled}(x - \mu)/\sigma = ax$$

I have previously centered and scaled the data so that the mean is equal to 0, and the x's cancel out therefore we are left with

$$a = a_{sclaed}/\sigma$$

Results are presented with the comparison to last week's homework.

Compare with previous homework results

I performed a 5-fold cross validation on each model presented at the end.

Results

	HW5 model 1	HW5 model 2	HW6 model 1	HW6 model 2
R^2	0.8	0.56	0.8	0.67
Adjusted R^2	0.71	0.49	0.72	0.61
CV MS	278973	90926	281898	478234

Coefficients

	Alpha.HW5	Alpha.HW6
M	87.83	87.73
Ed	188.32	188.23
Po1	192.80	192.73
Po2	-109.42	-109.22
LF	-663.83	-646.06
M.F	17.41	17.33
Pop	-0.73	-0.73
NW	4.20	4.13
U1	-5827.10	-5786.28
U2	167.80	167.33
Wealth	0.10	0.10
Ineq	70.67	70.45
Prob	-4855.27	-4863.63
Time	-3.48	-3.45

Predict crime for new city

New city predictors

- M = 14.0
- So = 0
- Ed = 10.0
- Po1 = 12.0
- Po2 = 15.5
- LF = 0.640
- M.F = 94.0
- Pop = 150
- NW = 1.1
- U1 = 0.120
- U2 = 3.6
- Wealth = 3200

Method

1. Rerun the pca analysis using only the predictors for the new city
2. Compute crime

The result was 628 offenses per 100,000 people.

Discussion

The results are interesting and not what I expected. The first thing that stands out to me are the alpha values. They are almost exactly the same as the previous homework. I am not sure what I expected, but I thought after performing OLS on the transformed data, transforming it back would result in dramatically different coefficients because some of the data were collinear, but they were nearly identical.

Secondly, the PCA was not as good as a predictor as the OLS model. The model with the lowest Mean Squared Error after cross validation was the predictor limited model from homework 5. This model reduced the number of predictors by cutting values with p-values greater than .075. Although it had the worst R-squared value it had a significantly better performance in cross validation. This suggests that the other models have been overfit. I really enjoyed this work. I have struggled to understand PCA in the past, but this crystalized it. The US crime data set does not seem to be the correct dataset to apply PCA to because it does not have a sufficient amount of data or predictors, but it has been an invaluable lesson to understand PCA.

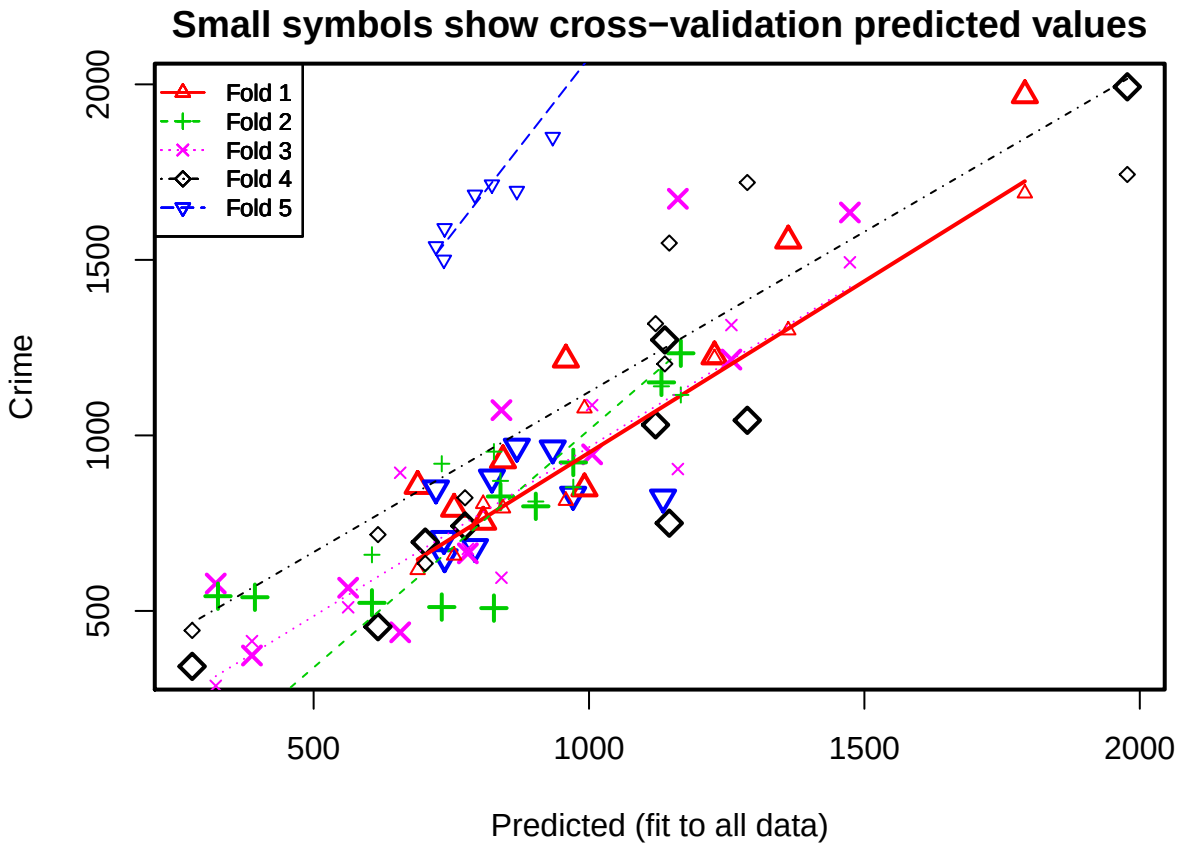
Cross Validation

```
## Warning in predict.lm(subs.lm, newdata = data[rows.out, ]): prediction from
## a rank-deficient fit may be misleading

## Warning in predict.lm(subs.lm, newdata = data[rows.out, ]): prediction from
## a rank-deficient fit may be misleading

## Warning in predict.lm(subs.lm, newdata = data[rows.out, ]): prediction from
## a rank-deficient fit may be misleading

## Warning in predict.lm(subs.lm, newdata = data[rows.out, ]): prediction from
## a rank-deficient fit may be misleading
```

```
## Warning in predict.lm(subs.lm, newdata = data[rows.out, ]): prediction from
## a rank-deficient fit may be misleading
```

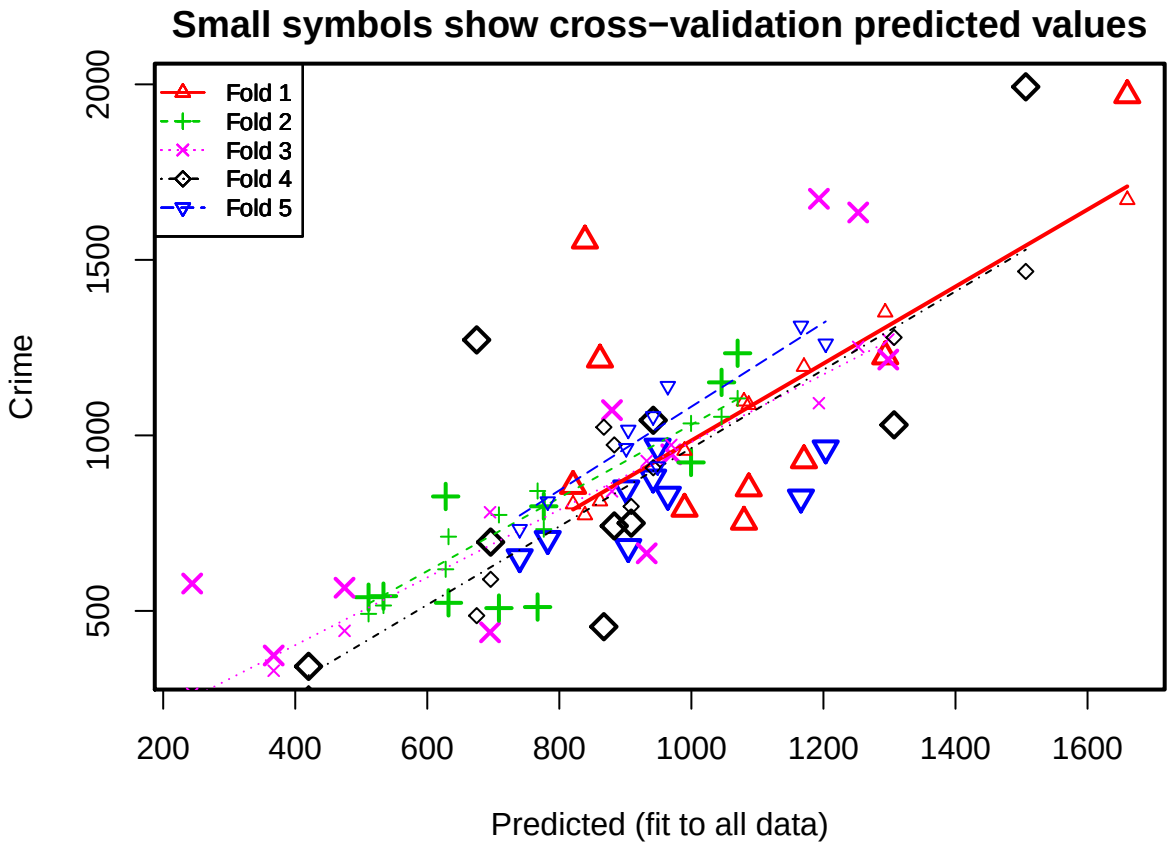
```
## Analysis of Variance Table
##
## Response: Crime
##           Df  Sum Sq Mean Sq F value  Pr(>F)
## M          1   55084   55084    1.26  0.2702
## So         1   15370   15370    0.35  0.5575
## Ed         1  905668  905668   20.72 7.7e-05 ***
## Po1        1 3076033 3076033   70.38 1.8e-09 ***
## Po2        1  153024  153024    3.50  0.0708 .
## LF         1   61134   61134    1.40  0.2459
## M.F        1  111000  111000    2.54  0.1212
## Pop        1   42649   42649    0.98  0.3309
## NW         1   14197   14197    0.32  0.5728
## U1         1    7065    7065    0.16  0.6904
## U2         1  269663  269663    6.17  0.0186 *
## Wealth     1   34748   34748    0.79  0.3795
## Ineq       1  547423  547423   12.52  0.0013 **
## Prob       1  222620  222620    5.09  0.0312 *
## Time       1   10304   10304    0.24  0.6307
## Residuals 31 1354946   43708
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
## Warning in cv.lm(data = raw, form.lm = form.lm.1, m = 5):
##
##  As there is >1 explanatory variable, cross-validation
##  predicted values for a fold are not a linear function
##  of corresponding overall predicted values.  Lines that
##  are shown for the different folds are approximate
```




```
##
## fold 1
## Observations in test set: 9
##                1    4    8   9  18      20   23    32   47
## Predicted    755 1791 1362 689 844 1227.84  958 807.8  992
## cvpred       658 1690 1300 617 792 1220.22  814 804.9 1077
## Crime        791 1969 1555 856 929 1225.00 1216 754.0  849
## CV residual  133  279  255 239 137    4.78  402 -50.9 -228
##
## Sum of squares = 453204    Mean square = 50356    n = 9
##
## fold 2
## Observations in test set: 10
##                5   13    15  17   25    34    39     40  42   46
## Predicted   1167  733 903.4 393  606 971.5 839.3 1131.5 326  827
## cvpred      1115  919 811.7  68  660 852.3 870.5 1139.8 -79  954
## Crime       1234  511 798.0 539  523 923.0 826.0 1151.0 542  508
## CV residual  119 -408 -13.7 471 -137  70.7 -44.5   11.2 621 -446
##
## Sum of squares = 1013064    Mean square = 101306    n = 10
##
## fold 3
## Observations in test set: 10
##                2   3   11    14   16   22     28    31   33    38
## Predicted   1474 322 1161 780.0 1006  657 1258.5 388.0  841 562.7
## cvpred      1493 287  904 676.5 1086  894 1314.3 413.9  595 510.4
## Crime       1635 578 1674 664.0  946  439 1216.0 373.0 1072 566.0
```

```
## CV residual  142 291  770 -12.5 -140 -455  -98.3 -40.9  477  55.6
##
## Sum of squares = 1166539    Mean square = 116654    n = 10
##
## fold 4
## Observations in test set: 9
##               19    21   26   27   29    30     36   44   45
## Predicted   1146 774.9 1977  279 1287 702.7 1137.6 1121  617
## cvpred      1548 822.3 1743  444 1720 635.2 1203.8 1318  717
## Crime        750 742.0 1993  342 1043 696.0 1272.0 1030  455
## CV residual -798 -80.3  250 -102 -677  60.8   68.2 -288 -262
##
## Sum of squares = 1335094    Mean square = 148344    n = 9
##
## fold 5
## Observations in test set: 9
##                 6    7   10   12   24   35    37   41    43
## Predicted     793  934  737  722  869  738   971  824  1134
## cvpred       1686 1850 1500 1539 1696 1590  2217 1715  2312
## Crime         682  963  705  849  968  653   831  880   823
## CV residual -1004 -887 -795 -690 -728 -937 -1386 -835 -1489
##
## Sum of squares = 9143814    Mean square = 1015979    n = 9
##
## Overall (Sum over all 9 folds)
##     ms
## 278973

## Analysis of Variance Table
##
## Response: Crime
##           Df  Sum Sq Mean Sq F value  Pr(>F)
## M          1   55084   55084    0.72  0.4001
## Ed         1  725967  725967    9.53  0.0037 **
## U2         1  736262  736262    9.67  0.0034 **
## Ineq       1   63813   63813    0.84  0.3655
## Wealth     1 2005043 2005043   26.33 7.8e-06 ***
## NW         1  248363  248363    3.26  0.0785 .
## Residuals 40 3046395   76160
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

## Warning in cv.lm(data = raw, form.lm = form.lm.2, m = 5):
##
##  As there is >1 explanatory variable, cross-validation
##  predicted values for a fold are not a linear function
##  of corresponding overall predicted values.  Lines that
##  are shown for the different folds are approximate
```




```
##
## fold 1
## Observations in test set: 9
##                 1    4    8   9   18   20   23   32   47
## Predicted     990 1660  839 821 1171 1294  862 1080 1087
## cvpred        957 1671  773 804 1196 1351  812 1097 1087
## Crime         791 1969 1555 856  929 1225 1216  754  849
## CV residual -166  298  782  52 -267 -126  404 -343 -238
##
## Sum of squares = 1154878    Mean square = 128320    n = 9
##
## fold 2
## Observations in test set: 10
##                  5   13    15    17   25   34  39     40    42   46
## Predicted   1070  767 776.4 511.0  632 1000 628 1045.7 533.6  709
## cvpred      1105  842 732.8 491.6  711 1034 619 1052.9 515.4  773
## Crime       1234  511 798.0 539.0  523  923 826 1151.0 542.0  508
## CV residual  129 -331  65.2  47.4 -188 -111 207   98.1  26.6 -265
##
## Sum of squares = 304064    Mean square = 30406    n = 10
##
## fold 3
## Observations in test set: 10
##                  2   3   11   14    16   22     28    31   33  38
## Predicted   1253 244 1193  932 969.5  695 1298.2 367.4  880 475
## cvpred      1252 264 1092  927 972.9  781 1272.9 329.5  841 443
## Crime       1635 578 1674  664 946.0  439 1216.0 373.0 1072 566
```

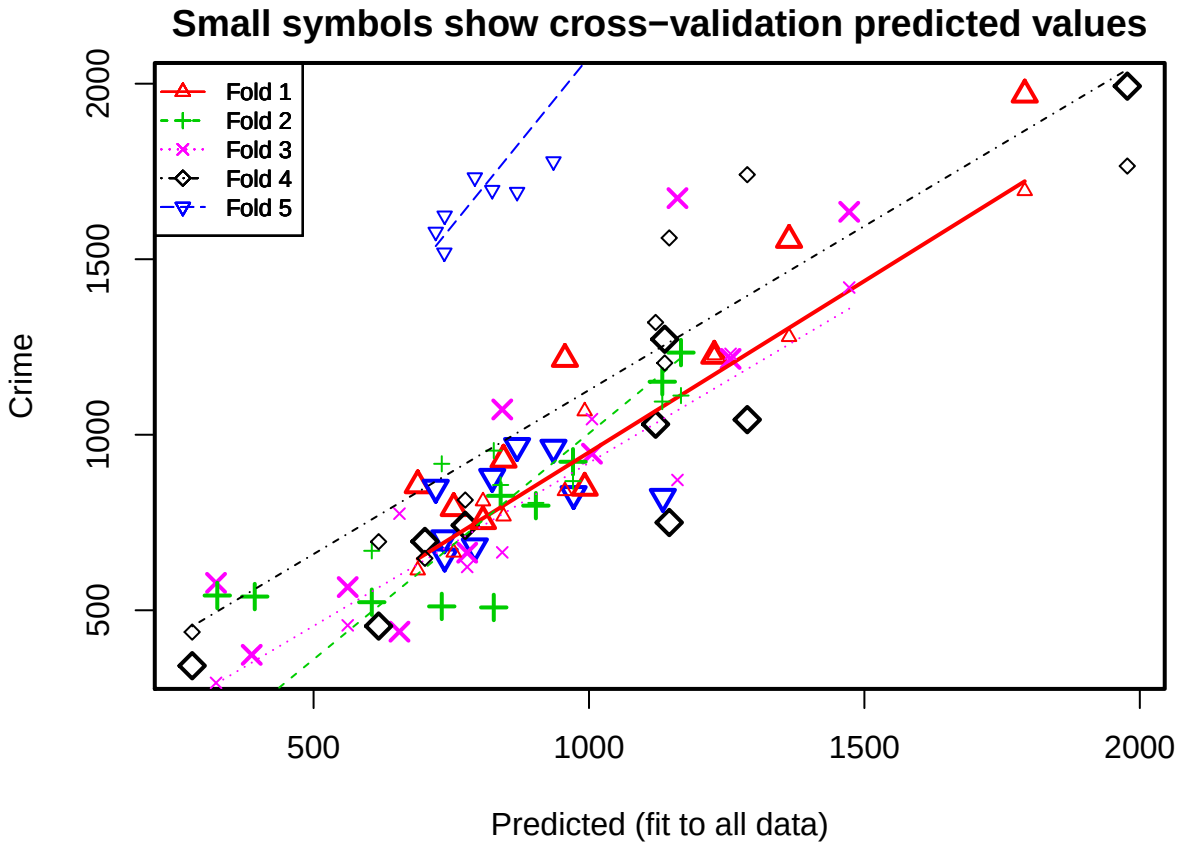
```
## CV residual  383 314  582 -263 -26.9 -342  -56.9  43.5  231 123
##
## Sum of squares = 844945    Mean square = 84495    n = 10
##
## fold 4
## Observations in test set: 9
##                19   21   26    27   29  30   36   44   45
## Predicted   908.9  883 1506 420.2  942 696  675 1307  867
## cvpred      797.6  973 1467 264.2  908 590  487 1279 1023
## Crime       750.0  742 1993 342.0 1043 696 1272 1030  455
## CV residual -47.6 -231  526  77.8  135 106  785 -249 -568
##
## Sum of squares = 1369735    Mean square = 152193    n = 9
##
## fold 5
## Observations in test set: 9
##                6    7   10   12    24    35   37   41   43
## Predicted    904 1203  782  901 948.9 739.9  964  942 1166
## cvpred      1016 1261  812  964 911.2 732.9 1140 1054 1312
## Crime        682  963  705  849 968.0 653.0  831  880  823
## CV residual -334 -298 -107 -115  56.8 -79.9 -309 -174 -489
##
## Sum of squares = 6e+05    Mean square = 66653    n = 9
##
## Overall (Sum over all 9 folds)
##    ms
## 90926
```

```
## Warning in predict.lm(subs.lm, newdata = data[rows.out, ]): prediction from
## a rank-deficient fit may be misleading

## Warning in predict.lm(subs.lm, newdata = data[rows.out, ]): prediction from
## a rank-deficient fit may be misleading

## Warning in predict.lm(subs.lm, newdata = data[rows.out, ]): prediction from
## a rank-deficient fit may be misleading

## Warning in predict.lm(subs.lm, newdata = data[rows.out, ]): prediction from
## a rank-deficient fit may be misleading

## Warning in predict.lm(subs.lm, newdata = data[rows.out, ]): prediction from
## a rank-deficient fit may be misleading
```

```
## Analysis of Variance Table
##
## Response: Crime
##           Df  Sum Sq Mean Sq F value  Pr(>F)
## PC1        1 1466202 1466202   34.63 1.5e-06 ***
## PC2        1  416830  416830    9.84 0.00364 **
## PC3        1   54504   54504    1.29 0.26499
## PC4        1     709     709    0.02 0.89789
## PC5        1 2394517 2394517   56.55 1.5e-08 ***
## PC6        1  103876  103876    2.45 0.12712
## PC7        1  150096  150096    3.54 0.06885 .
## PC8        1   53193   53193    1.26 0.27070
```

```
## PC9         1   19246   19246    0.45 0.50503
## PC10        1    6482    6482    0.15 0.69820
## PC11        1  606241  606241   14.32 0.00064 ***
## PC12        1  128601  128601    3.04 0.09098 .
## PC13        1   43778   43778    1.03 0.31687
## PC14        1   81679   81679    1.93 0.17446
## Residuals 32 1354974   42343
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
## Warning in cv.lm(data = model_1_data, form.lm = form.lm.1, m = 5):
##
##  As there is >1 explanatory variable, cross-validation
##  predicted values for a fold are not a linear function
##  of corresponding overall predicted values.  Lines that
##  are shown for the different folds are approximate
```



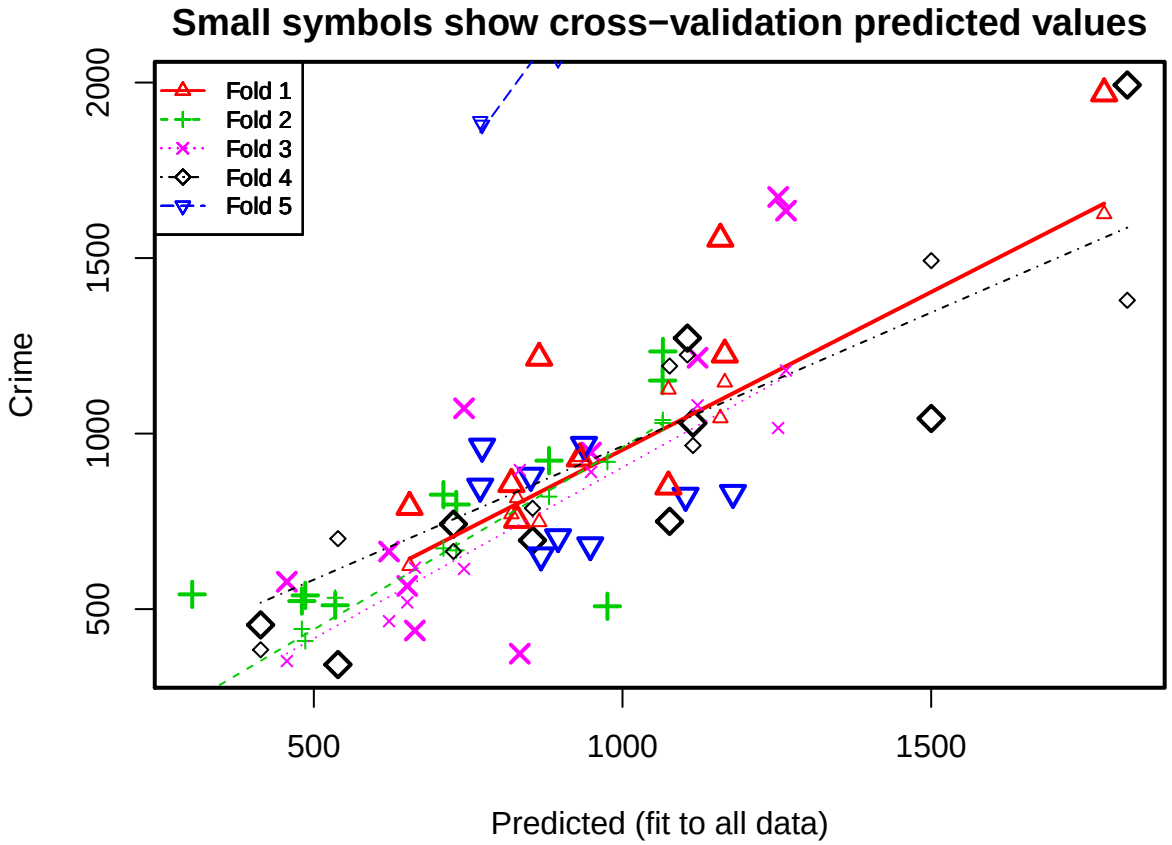

```
##
## fold 1
## Observations in test set: 9
##                1    4    8   9  18      20   23    32   47
## Predicted    754 1791 1363 689 845 1227.47  956 807.5  992
## cvpred       665 1695 1279 614 768 1226.68  840 810.7 1067
## Crime        791 1969 1555 856 929 1225.00 1216 754.0  849
## CV residual 126  274  276 242 161   -1.68  376 -56.7 -218
##
## Sum of squares = 444402    Mean square = 49378    n = 9
##
```

```
## fold 2
## Observations in test set: 10
##                5   13     15    17   25    34    39     40    42   46
## Predicted   1167  733 903.55 393.2  606 971.0 839.6 1133.1 325.2  827
## cvpred      1112  917 805.73  90.4  669 867.9 856.9 1094.5 -35.9  955
## Crime       1234  511 798.00 539.0  523 923.0 826.0 1151.0 542.0  508
## CV residual  122 -406  -7.73 448.6 -146  55.1 -30.9   56.5 577.9 -447
##
## Sum of squares = 943406    Mean square = 94341    n = 10
##
## fold 3
## Observations in test set: 10
##                2   3   11    14     16   22     28     31   33  38
## Predicted   1472 323 1161 778.9 1005.4  656 1257.1 387.70  843 562
## cvpred      1419 293  871 623.3 1044.4  775 1230.5 375.87  665 457
## Crime       1635 578 1674 664.0  946.0  439 1216.0 373.00 1072 566
## CV residual  216 285  803  40.7  -98.4 -336  -14.5  -2.87  407 109
##
## Sum of squares = 1074419    Mean square = 107442    n = 10
##
## fold 4
## Observations in test set: 9
##               19  21   26    27   29    30     36   44   45
## Predicted   1146 775 1977 279.6 1287 702.1 1137.5 1121  618
## cvpred      1561 814 1765 437.9 1741 647.9 1204.4 1320  695
## Crime        750 742 1993 342.0 1043 696.0 1272.0 1030  455
## CV residual -811 -72  228 -95.9 -698  48.1   67.6 -290 -240
##
## Sum of squares = 1358949    Mean square = 150994    n = 9
##
## fold 5
## Observations in test set: 9
##                 6    7   10   12   24   35    37   41    43
## Predicted     793  935  737  722  869  738   972  824  1134
## cvpred       1733 1779 1519 1578 1691 1624  2284 1697  2313
## Crime         682  963  705  849  968  653   831  880   823
## CV residual -1051 -816 -814 -729 -723 -971 -1453 -817 -1490
##
## Sum of squares = 9428008    Mean square = 1047556    n = 9
##
## Overall (Sum over all 9 folds)
##     ms
## 281898
## Warning in predict.lm(subs.lm, newdata = data[rows.out, ]): prediction from
## a rank-deficient fit may be misleading

## Warning in predict.lm(subs.lm, newdata = data[rows.out, ]): prediction from
## a rank-deficient fit may be misleading

## Warning in predict.lm(subs.lm, newdata = data[rows.out, ]): prediction from
## a rank-deficient fit may be misleading

## Warning in predict.lm(subs.lm, newdata = data[rows.out, ]): prediction from
```

```
## a rank-deficient fit may be misleading

## Warning in predict.lm(subs.lm, newdata = data[rows.out, ]): prediction from
## a rank-deficient fit may be misleading
```

```
## Analysis of Variance Table
##
## Response: Crime
##           Df  Sum Sq Mean Sq F value  Pr(>F)
## PC1        1 1466202 1466202   24.92 1.3e-05 ***
## PC2        1  416830  416830    7.09   0.011 *
## PC3        1   54504   54504    0.93   0.342
## PC4        1     709     709    0.01   0.913
## PC5        1 2394517 2394517   40.71 1.5e-07 ***
## PC6        1  103876  103876    1.77   0.192
## PC7        1  150096  150096    2.55   0.118
## Residuals 39 2294195   58826
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
## Warning in cv.lm(data = model_2_data, form.lm = form.lm.1, m = 5):
##
##  As there is >1 explanatory variable, cross-validation
##  predicted values for a fold are not a linear function
##  of corresponding overall predicted values.  Lines that
##  are shown for the different folds are approximate
```




```
##
```

```
## fold 1
## Observations in test set: 9
##                1    4    8     9     18     20   23    32   47
## Predicted    655 1780 1159 820.0 930.95 1165.7  865 829.3 1074
## cvpred       623 1626 1044 770.6 933.56 1146.1  748 816.6 1126
## Crime        791 1969 1555 856.0 929.00 1225.0 1216 754.0  849
## CV residual 168  343  511  85.4  -4.56   78.9  468 -62.6 -277
##
## Sum of squares = 719875    Mean square = 79986    n = 9
##
## fold 2
## Observations in test set: 10
##                5    13  15  17    25  34  39   40  42   46
## Predicted   1065 534.6 731 486 480.6 881 710 1065 303  976
## cvpred      1039 532.2 667 409 443.5 820 673 1030 203  919
## Crime       1234 511.0 798 539 523.0 923 826 1151 542  508
## CV residual  195 -21.2 131 130  79.5 103 153  121 339 -411
##
## Sum of squares = 411294    Mean square = 41129    n = 10
##
## fold 3
## Observations in test set: 10
##                2   3   11  14    16   22   28   31   33    38
## Predicted   1265 456 1252 622 949.0  664 1122  834  743 651.3
## cvpred      1180 352 1016 466 890.6  618 1080  896  615 519.2
## Crime       1635 578 1674 664 946.0  439 1216  373 1072 566.0
## CV residual  455 226  658 198  55.4 -179  136 -523  457  46.8
##
## Sum of squares = 1269578    Mean square = 126958    n = 10
##
## fold 4
## Observations in test set: 9
##                19    21   26   27   29    30     36   44    45
## Predicted   1076 726.2 1818  539 1500 854.3 1105.2 1114 413.8
## cvpred      1192 664.2 1380  701 1493 787.2 1224.1  966 384.1
## Crime        750 742.0 1993  342 1043 696.0 1272.0 1030 455.0
## CV residual -442  77.8  613 -359 -450 -91.2   47.9   64  70.9
##
## Sum of squares = 928486    Mean square = 103165    n = 9
##
## fold 5
## Observations in test set: 9
##                 6    7    10    12    24    35    37    41    43
## Predicted     948  773   896   769   938   868  1179   851  1102
## cvpred       2317 1880  2067  1890  2130  2102  2842  2136  2727
## Crime         682  963   705   849   968   653   831   880   823
## CV residual -1635 -917 -1362 -1041 -1162 -1449 -2011 -1256 -1904
##
## Sum of squares = 19147758    Mean square = 2127529    n = 9
##
## Overall (Sum over all 9 folds)
##     ms
## 478234
```