

Principal Component Regression

Shrihari Hudli, Abhishek Sanghavi, Gabriel Urrutia, Jingshen Zhao

August 11, 2015

Administrative Announcements

- The deadline for HW2 has been extended to Wednesday, August 19, at 12:00 noon. This is a hard deadline. 12:01 would be considered late.
- Office hours will be modified to account for this deadline extension.
- Rmd and Github will be rigorously enforced for HW2, as this is a basic course competency.
- If you have trouble with Rmd or Github, test with a “Hello, world” program to troubleshoot.

Outline: Latent Features

- Principal component analysis and regression (PCR) - discussed in this module
- Factor analysis (FA)
- Canonical correlation analysis (CCA)

Introduction to Principal Component Regression

- The aim of this analysis is to find the “privileged variable(s)” in our data set
- Synthetic features are generated from all existing features.
- A set of observations of possibly correlated variables is converted into a smaller set of values of linearly uncorrelated variables, called **principal components**.
- PCR can be used as a good beginning of a pipeline, chained together with linear regression, or any other regression for that matter.
- This procedure of layering between raw data and final results is used often in data science

Linear Regression

Scalar output y_i (scaled octane level for the gasoline example)

Feature vector x_i - this is a d -dimensional feature vector

$$x \in R^D$$

Basic Regression

$$\dots [1] y_i = \beta_0 + \beta^T x_i + \varepsilon$$

$$\dots [2] y_i = \beta_0 + \sum_{j=1}^D \beta_j x_j + \varepsilon$$

$X \in R^{n \times D}$ (feature matrix)

$$X = \begin{pmatrix} x_1^T \\ x_2^T \\ x_3^T \\ \vdots \\ x_N^T \end{pmatrix}$$

Principal Component Regression

Step 1: Run PCA on X

D = number of features, K = number of PC's, N = number of observations

Principal components: v_1, \dots, v_K where $K \leq D$

Scores: $s_{ij} = X_i^T v_j$ for $i = 1, \dots, N$ feature vector i on PC(j) $v = 1, \dots, K$

$$S_i \in R^K = \begin{pmatrix} s_{i1} \\ s_{i2} \\ \vdots \\ s_{iK} \end{pmatrix}$$

Step 2: Run Regression

Treat S_i as a feature vector and regress y_i on S_i .

$$\underline{y_i = \gamma_0 + \gamma^T s_i + \delta}$$

$$\gamma \in R^K$$

$$s \in R^K$$

Although principal component regression uses ordinary least squares regression, methods like tree, random forests, ridge, and lasso can also be applied.

Questions:

Q. Why would we do this? Why generate synthetic features?

A. You'd be able to get similar R^2 (predictive power) with significantly fewer variables. This reduces the complexity of the model.

In addition, by usually regressing on only a subset of all the principal components, PCR can result in dimension reduction through substantially lowering the effective number of parameters characterizing the underlying model. This can be particularly useful in settings with high-dimensional covariates.

Also, through appropriate selection of the principal components to be used for regression, PCR can lead to efficient prediction of the outcome based on the assumed model.

Q.)Are there any downsides to this method?

A.)Interpretability is made more difficult than linear regression.

Q.)Should one use PCR for a sparse vector?

A.) A sparse vector is a vector with only a few non-zero values, like a few needles in a giant haystack. It has a lot of nice properties, such as the ability to use sparse algebra to get vector products, etc. It has a huge advantage: very low storage cost. However, it's usually a bad idea to run PCR on a sparse vector, as it would be a regression with sparse features and negate the value and convenience of PCR.

For example, imagine a matrix where: rows = users of Netflix, columns = movies, possible values = didn't see it (NA) / didn't like it (0) / liked it (1).

This would be a sparse matrix as the number of movies seen by any individual would be far less than the total number of movies, so most of the cells would have the value NA.

PRCOMP would be disastrous, waiting a long time. In this case you should use the recent advances in PCR such as sparse principal components.

Gasoline Dataset

Download /data/gasoline.csv and /R/gasoline.R from [STA 380](#)

```
gasoline = read.csv('../data/gasoline.csv', header=TRUE)
dim(gasoline)

## [1] 60 402

names(gasoline)
```

```
## [1] "octane" "X900.nm" "X902.nm" "X904.nm" "X906.nm" "X908.n
nm"
## [7] "X910.nm" "X912.nm" "X914.nm" "X916.nm" "X918.nm" "X920.n
nm"
## [13] "X922.nm" "X924.nm" "X926.nm" "X928.nm" "X930.nm" "X932.n
nm"
## [19] "X934.nm" "X936.nm" "X938.nm" "X940.nm" "X942.nm" "X944.n
nm"
## [25] "X946.nm" "X948.nm" "X950.nm" "X952.nm" "X954.nm" "X956.n
nm"
## [31] "X958.nm" "X960.nm" "X962.nm" "X964.nm" "X966.nm" "X968.n
nm"
## [37] "X970.nm" "X972.nm" "X974.nm" "X976.nm" "X978.nm" "X980.n
nm"
## [43] "X982.nm" "X984.nm" "X986.nm" "X988.nm" "X990.nm" "X992.n
nm"
## [49] "X994.nm" "X996.nm" "X998.nm" "X1000.nm" "X1002.nm" "X1004.
nm"
## [55] "X1006.nm" "X1008.nm" "X1010.nm" "X1012.nm" "X1014.nm" "X1016.
nm"
## [61] "X1018.nm" "X1020.nm" "X1022.nm" "X1024.nm" "X1026.nm" "X1028.
nm"
## [67] "X1030.nm" "X1032.nm" "X1034.nm" "X1036.nm" "X1038.nm" "X1040.
nm"
## [73] "X1042.nm" "X1044.nm" "X1046.nm" "X1048.nm" "X1050.nm" "X1052.
nm"
## [79] "X1054.nm" "X1056.nm" "X1058.nm" "X1060.nm" "X1062.nm" "X1064.
nm"
## [85] "X1066.nm" "X1068.nm" "X1070.nm" "X1072.nm" "X1074.nm" "X1076.
nm"
## [91] "X1078.nm" "X1080.nm" "X1082.nm" "X1084.nm" "X1086.nm" "X1088.
nm"
## [97] "X1090.nm" "X1092.nm" "X1094.nm" "X1096.nm" "X1098.nm" "X1100.
nm"
## [103] "X1102.nm" "X1104.nm" "X1106.nm" "X1108.nm" "X1110.nm" "X1112.
nm"
## [109] "X1114.nm" "X1116.nm" "X1118.nm" "X1120.nm" "X1122.nm" "X1124.
nm"
## [115] "X1126.nm" "X1128.nm" "X1130.nm" "X1132.nm" "X1134.nm" "X1136.
nm"
## [121] "X1138.nm" "X1140.nm" "X1142.nm" "X1144.nm" "X1146.nm" "X1148.
nm"
## [127] "X1150.nm" "X1152.nm" "X1154.nm" "X1156.nm" "X1158.nm" "X1160.
nm"
## [133] "X1162.nm" "X1164.nm" "X1166.nm" "X1168.nm" "X1170.nm" "X1172.
nm"
## [139] "X1174.nm" "X1176.nm" "X1178.nm" "X1180.nm" "X1182.nm" "X1184.
nm"
## [145] "X1186.nm" "X1188.nm" "X1190.nm" "X1192.nm" "X1194.nm" "X1196.
nm"
```

[151] "X1198.nm" "X1200.nm" "X1202.nm" "X1204.nm" "X1206.nm" "X1208.nm"
[157] "X1210.nm" "X1212.nm" "X1214.nm" "X1216.nm" "X1218.nm" "X1220.nm"
[163] "X1222.nm" "X1224.nm" "X1226.nm" "X1228.nm" "X1230.nm" "X1232.nm"
[169] "X1234.nm" "X1236.nm" "X1238.nm" "X1240.nm" "X1242.nm" "X1244.nm"
[175] "X1246.nm" "X1248.nm" "X1250.nm" "X1252.nm" "X1254.nm" "X1256.nm"
[181] "X1258.nm" "X1260.nm" "X1262.nm" "X1264.nm" "X1266.nm" "X1268.nm"
[187] "X1270.nm" "X1272.nm" "X1274.nm" "X1276.nm" "X1278.nm" "X1280.nm"
[193] "X1282.nm" "X1284.nm" "X1286.nm" "X1288.nm" "X1290.nm" "X1292.nm"
[199] "X1294.nm" "X1296.nm" "X1298.nm" "X1300.nm" "X1302.nm" "X1304.nm"
[205] "X1306.nm" "X1308.nm" "X1310.nm" "X1312.nm" "X1314.nm" "X1316.nm"
[211] "X1318.nm" "X1320.nm" "X1322.nm" "X1324.nm" "X1326.nm" "X1328.nm"
[217] "X1330.nm" "X1332.nm" "X1334.nm" "X1336.nm" "X1338.nm" "X1340.nm"
[223] "X1342.nm" "X1344.nm" "X1346.nm" "X1348.nm" "X1350.nm" "X1352.nm"
[229] "X1354.nm" "X1356.nm" "X1358.nm" "X1360.nm" "X1362.nm" "X1364.nm"
[235] "X1366.nm" "X1368.nm" "X1370.nm" "X1372.nm" "X1374.nm" "X1376.nm"
[241] "X1378.nm" "X1380.nm" "X1382.nm" "X1384.nm" "X1386.nm" "X1388.nm"
[247] "X1390.nm" "X1392.nm" "X1394.nm" "X1396.nm" "X1398.nm" "X1400.nm"
[253] "X1402.nm" "X1404.nm" "X1406.nm" "X1408.nm" "X1410.nm" "X1412.nm"
[259] "X1414.nm" "X1416.nm" "X1418.nm" "X1420.nm" "X1422.nm" "X1424.nm"
[265] "X1426.nm" "X1428.nm" "X1430.nm" "X1432.nm" "X1434.nm" "X1436.nm"
[271] "X1438.nm" "X1440.nm" "X1442.nm" "X1444.nm" "X1446.nm" "X1448.nm"
[277] "X1450.nm" "X1452.nm" "X1454.nm" "X1456.nm" "X1458.nm" "X1460.nm"
[283] "X1462.nm" "X1464.nm" "X1466.nm" "X1468.nm" "X1470.nm" "X1472.nm"
[289] "X1474.nm" "X1476.nm" "X1478.nm" "X1480.nm" "X1482.nm" "X1484.nm"
[295] "X1486.nm" "X1488.nm" "X1490.nm" "X1492.nm" "X1494.nm" "X1496.nm"

```
## [301] "X1498.nm" "X1500.nm" "X1502.nm" "X1504.nm" "X1506.nm" "X1508.
nm"
## [307] "X1510.nm" "X1512.nm" "X1514.nm" "X1516.nm" "X1518.nm" "X1520.
nm"
## [313] "X1522.nm" "X1524.nm" "X1526.nm" "X1528.nm" "X1530.nm" "X1532.
nm"
## [319] "X1534.nm" "X1536.nm" "X1538.nm" "X1540.nm" "X1542.nm" "X1544.
nm"
## [325] "X1546.nm" "X1548.nm" "X1550.nm" "X1552.nm" "X1554.nm" "X1556.
nm"
## [331] "X1558.nm" "X1560.nm" "X1562.nm" "X1564.nm" "X1566.nm" "X1568.
nm"
## [337] "X1570.nm" "X1572.nm" "X1574.nm" "X1576.nm" "X1578.nm" "X1580.
nm"
## [343] "X1582.nm" "X1584.nm" "X1586.nm" "X1588.nm" "X1590.nm" "X1592.
nm"
## [349] "X1594.nm" "X1596.nm" "X1598.nm" "X1600.nm" "X1602.nm" "X1604.
nm"
## [355] "X1606.nm" "X1608.nm" "X1610.nm" "X1612.nm" "X1614.nm" "X1616.
nm"
## [361] "X1618.nm" "X1620.nm" "X1622.nm" "X1624.nm" "X1626.nm" "X1628.
nm"
## [367] "X1630.nm" "X1632.nm" "X1634.nm" "X1636.nm" "X1638.nm" "X1640.
nm"
## [373] "X1642.nm" "X1644.nm" "X1646.nm" "X1648.nm" "X1650.nm" "X1652.
nm"
## [379] "X1654.nm" "X1656.nm" "X1658.nm" "X1660.nm" "X1662.nm" "X1664.
nm"
## [385] "X1666.nm" "X1668.nm" "X1670.nm" "X1672.nm" "X1674.nm" "X1676.
nm"
## [391] "X1678.nm" "X1680.nm" "X1682.nm" "X1684.nm" "X1686.nm" "X1688.
nm"
## [397] "X1690.nm" "X1692.nm" "X1694.nm" "X1696.nm" "X1698.nm" "X1700.
nm"
```

Near-infrared spectroscopy is a technique used to measure the octane level of fuel without having to burn the fuel.

What makes this dataset different from the ones seen so far is that the dataset has more features (402) than observations (60).

```
X = as.matrix(gasoline[,-1])
y = gasoline[,1]
lm1 = lm(y ~ X)
summary(lm1)

##
## Call:
## lm(formula = y ~ X)
##
## Residuals:
```

```

## ALL 60 residuals are 0: no residual degrees of freedom!
##
## Coefficients: (342 not defined because of singularities)
##           Estimate Std. Error t value Pr(>|t|)
## (Intercept)    483.2           NA      NA      NA
## XX900.nm   -24260.3           NA      NA      NA
## XX902.nm   -18267.5           NA      NA      NA
## XX904.nm    15989.9           NA      NA      NA
## XX906.nm   -14110.5           NA      NA      NA
## XX908.nm    21111.9           NA      NA      NA
## XX910.nm    17721.0           NA      NA      NA
## XX912.nm    -6133.9           NA      NA      NA
## XX914.nm    -486.7           NA      NA      NA
## XX916.nm    6511.4           NA      NA      NA
## XX918.nm   -4253.3           NA      NA      NA
## XX920.nm   -32548.4           NA      NA      NA
## XX922.nm    24675.4           NA      NA      NA
## XX924.nm    1871.8           NA      NA      NA
## XX926.nm   -22699.9           NA      NA      NA
## XX928.nm    37521.8           NA      NA      NA
## XX930.nm   -4189.9           NA      NA      NA
## XX932.nm   -1284.8           NA      NA      NA
## XX934.nm    26831.8           NA      NA      NA
## XX936.nm    24394.2           NA      NA      NA
## XX938.nm    6157.6           NA      NA      NA
## XX940.nm   -62075.6           NA      NA      NA
## XX942.nm   -76762.4           NA      NA      NA
## XX944.nm    51875.9           NA      NA      NA
## XX946.nm   -20214.5           NA      NA      NA
## XX948.nm   -26627.7           NA      NA      NA
## XX950.nm    37529.8           NA      NA      NA
## XX952.nm   -15734.7           NA      NA      NA
## XX954.nm    87375.8           NA      NA      NA
## XX956.nm   -39180.5           NA      NA      NA
## XX958.nm   -23517.8           NA      NA      NA
## XX960.nm    21175.5           NA      NA      NA
## XX962.nm    28376.4           NA      NA      NA
## XX964.nm    56671.2           NA      NA      NA
## XX966.nm   -46634.1           NA      NA      NA
## XX968.nm   -17618.9           NA      NA      NA
## XX970.nm    76039.5           NA      NA      NA
## XX972.nm   -7797.7           NA      NA      NA
## XX974.nm   -68962.9           NA      NA      NA
## XX976.nm   -54896.1           NA      NA      NA
## XX978.nm    18649.7           NA      NA      NA
## XX980.nm    22834.2           NA      NA      NA
## XX982.nm   -57093.1           NA      NA      NA
## XX984.nm    18209.8           NA      NA      NA
## XX986.nm    37030.0           NA      NA      NA
## XX988.nm     8335.5           NA      NA      NA

```

## XX990.nm	17110.0	NA	NA	NA
## XX992.nm	-34620.8	NA	NA	NA
## XX994.nm	-91226.0	NA	NA	NA
## XX996.nm	96890.9	NA	NA	NA
## XX998.nm	20509.7	NA	NA	NA
## XX1000.nm	53535.2	NA	NA	NA
## XX1002.nm	4488.4	NA	NA	NA
## XX1004.nm	49531.4	NA	NA	NA
## XX1006.nm	-65202.8	NA	NA	NA
## XX1008.nm	9533.4	NA	NA	NA
## XX1010.nm	-88366.9	NA	NA	NA
## XX1012.nm	45037.9	NA	NA	NA
## XX1014.nm	15693.9	NA	NA	NA
## XX1016.nm	-28076.9	NA	NA	NA
## XX1018.nm	NA	NA	NA	NA
## XX1020.nm	NA	NA	NA	NA
## XX1022.nm	NA	NA	NA	NA
## XX1024.nm	NA	NA	NA	NA
## XX1026.nm	NA	NA	NA	NA
## XX1028.nm	NA	NA	NA	NA
## XX1030.nm	NA	NA	NA	NA
## XX1032.nm	NA	NA	NA	NA
## XX1034.nm	NA	NA	NA	NA
## XX1036.nm	NA	NA	NA	NA
## XX1038.nm	NA	NA	NA	NA
## XX1040.nm	NA	NA	NA	NA
## XX1042.nm	NA	NA	NA	NA
## XX1044.nm	NA	NA	NA	NA
## XX1046.nm	NA	NA	NA	NA
## XX1048.nm	NA	NA	NA	NA
## XX1050.nm	NA	NA	NA	NA
## XX1052.nm	NA	NA	NA	NA
## XX1054.nm	NA	NA	NA	NA
## XX1056.nm	NA	NA	NA	NA
## XX1058.nm	NA	NA	NA	NA
## XX1060.nm	NA	NA	NA	NA
## XX1062.nm	NA	NA	NA	NA
## XX1064.nm	NA	NA	NA	NA
## XX1066.nm	NA	NA	NA	NA
## XX1068.nm	NA	NA	NA	NA
## XX1070.nm	NA	NA	NA	NA
## XX1072.nm	NA	NA	NA	NA
## XX1074.nm	NA	NA	NA	NA
## XX1076.nm	NA	NA	NA	NA
## XX1078.nm	NA	NA	NA	NA
## XX1080.nm	NA	NA	NA	NA
## XX1082.nm	NA	NA	NA	NA
## XX1084.nm	NA	NA	NA	NA
## XX1086.nm	NA	NA	NA	NA
## XX1088.nm	NA	NA	NA	NA

## XX1090.nm	NA	NA	NA	NA
## XX1092.nm	NA	NA	NA	NA
## XX1094.nm	NA	NA	NA	NA
## XX1096.nm	NA	NA	NA	NA
## XX1098.nm	NA	NA	NA	NA
## XX1100.nm	NA	NA	NA	NA
## XX1102.nm	NA	NA	NA	NA
## XX1104.nm	NA	NA	NA	NA
## XX1106.nm	NA	NA	NA	NA
## XX1108.nm	NA	NA	NA	NA
## XX1110.nm	NA	NA	NA	NA
## XX1112.nm	NA	NA	NA	NA
## XX1114.nm	NA	NA	NA	NA
## XX1116.nm	NA	NA	NA	NA
## XX1118.nm	NA	NA	NA	NA
## XX1120.nm	NA	NA	NA	NA
## XX1122.nm	NA	NA	NA	NA
## XX1124.nm	NA	NA	NA	NA
## XX1126.nm	NA	NA	NA	NA
## XX1128.nm	NA	NA	NA	NA
## XX1130.nm	NA	NA	NA	NA
## XX1132.nm	NA	NA	NA	NA
## XX1134.nm	NA	NA	NA	NA
## XX1136.nm	NA	NA	NA	NA
## XX1138.nm	NA	NA	NA	NA
## XX1140.nm	NA	NA	NA	NA
## XX1142.nm	NA	NA	NA	NA
## XX1144.nm	NA	NA	NA	NA
## XX1146.nm	NA	NA	NA	NA
## XX1148.nm	NA	NA	NA	NA
## XX1150.nm	NA	NA	NA	NA
## XX1152.nm	NA	NA	NA	NA
## XX1154.nm	NA	NA	NA	NA
## XX1156.nm	NA	NA	NA	NA
## XX1158.nm	NA	NA	NA	NA
## XX1160.nm	NA	NA	NA	NA
## XX1162.nm	NA	NA	NA	NA
## XX1164.nm	NA	NA	NA	NA
## XX1166.nm	NA	NA	NA	NA
## XX1168.nm	NA	NA	NA	NA
## XX1170.nm	NA	NA	NA	NA
## XX1172.nm	NA	NA	NA	NA
## XX1174.nm	NA	NA	NA	NA
## XX1176.nm	NA	NA	NA	NA
## XX1178.nm	NA	NA	NA	NA
## XX1180.nm	NA	NA	NA	NA
## XX1182.nm	NA	NA	NA	NA
## XX1184.nm	NA	NA	NA	NA
## XX1186.nm	NA	NA	NA	NA
## XX1188.nm	NA	NA	NA	NA

## XX1190.nm	NA	NA	NA	NA
## XX1192.nm	NA	NA	NA	NA
## XX1194.nm	NA	NA	NA	NA
## XX1196.nm	NA	NA	NA	NA
## XX1198.nm	NA	NA	NA	NA
## XX1200.nm	NA	NA	NA	NA
## XX1202.nm	NA	NA	NA	NA
## XX1204.nm	NA	NA	NA	NA
## XX1206.nm	NA	NA	NA	NA
## XX1208.nm	NA	NA	NA	NA
## XX1210.nm	NA	NA	NA	NA
## XX1212.nm	NA	NA	NA	NA
## XX1214.nm	NA	NA	NA	NA
## XX1216.nm	NA	NA	NA	NA
## XX1218.nm	NA	NA	NA	NA
## XX1220.nm	NA	NA	NA	NA
## XX1222.nm	NA	NA	NA	NA
## XX1224.nm	NA	NA	NA	NA
## XX1226.nm	NA	NA	NA	NA
## XX1228.nm	NA	NA	NA	NA
## XX1230.nm	NA	NA	NA	NA
## XX1232.nm	NA	NA	NA	NA
## XX1234.nm	NA	NA	NA	NA
## XX1236.nm	NA	NA	NA	NA
## XX1238.nm	NA	NA	NA	NA
## XX1240.nm	NA	NA	NA	NA
## XX1242.nm	NA	NA	NA	NA
## XX1244.nm	NA	NA	NA	NA
## XX1246.nm	NA	NA	NA	NA
## XX1248.nm	NA	NA	NA	NA
## XX1250.nm	NA	NA	NA	NA
## XX1252.nm	NA	NA	NA	NA
## XX1254.nm	NA	NA	NA	NA
## XX1256.nm	NA	NA	NA	NA
## XX1258.nm	NA	NA	NA	NA
## XX1260.nm	NA	NA	NA	NA
## XX1262.nm	NA	NA	NA	NA
## XX1264.nm	NA	NA	NA	NA
## XX1266.nm	NA	NA	NA	NA
## XX1268.nm	NA	NA	NA	NA
## XX1270.nm	NA	NA	NA	NA
## XX1272.nm	NA	NA	NA	NA
## XX1274.nm	NA	NA	NA	NA
## XX1276.nm	NA	NA	NA	NA
## XX1278.nm	NA	NA	NA	NA
## XX1280.nm	NA	NA	NA	NA
## XX1282.nm	NA	NA	NA	NA
## XX1284.nm	NA	NA	NA	NA
## XX1286.nm	NA	NA	NA	NA
## XX1288.nm	NA	NA	NA	NA

## XX1290.nm	NA	NA	NA	NA
## XX1292.nm	NA	NA	NA	NA
## XX1294.nm	NA	NA	NA	NA
## XX1296.nm	NA	NA	NA	NA
## XX1298.nm	NA	NA	NA	NA
## XX1300.nm	NA	NA	NA	NA
## XX1302.nm	NA	NA	NA	NA
## XX1304.nm	NA	NA	NA	NA
## XX1306.nm	NA	NA	NA	NA
## XX1308.nm	NA	NA	NA	NA
## XX1310.nm	NA	NA	NA	NA
## XX1312.nm	NA	NA	NA	NA
## XX1314.nm	NA	NA	NA	NA
## XX1316.nm	NA	NA	NA	NA
## XX1318.nm	NA	NA	NA	NA
## XX1320.nm	NA	NA	NA	NA
## XX1322.nm	NA	NA	NA	NA
## XX1324.nm	NA	NA	NA	NA
## XX1326.nm	NA	NA	NA	NA
## XX1328.nm	NA	NA	NA	NA
## XX1330.nm	NA	NA	NA	NA
## XX1332.nm	NA	NA	NA	NA
## XX1334.nm	NA	NA	NA	NA
## XX1336.nm	NA	NA	NA	NA
## XX1338.nm	NA	NA	NA	NA
## XX1340.nm	NA	NA	NA	NA
## XX1342.nm	NA	NA	NA	NA
## XX1344.nm	NA	NA	NA	NA
## XX1346.nm	NA	NA	NA	NA
## XX1348.nm	NA	NA	NA	NA
## XX1350.nm	NA	NA	NA	NA
## XX1352.nm	NA	NA	NA	NA
## XX1354.nm	NA	NA	NA	NA
## XX1356.nm	NA	NA	NA	NA
## XX1358.nm	NA	NA	NA	NA
## XX1360.nm	NA	NA	NA	NA
## XX1362.nm	NA	NA	NA	NA
## XX1364.nm	NA	NA	NA	NA
## XX1366.nm	NA	NA	NA	NA
## XX1368.nm	NA	NA	NA	NA
## XX1370.nm	NA	NA	NA	NA
## XX1372.nm	NA	NA	NA	NA
## XX1374.nm	NA	NA	NA	NA
## XX1376.nm	NA	NA	NA	NA
## XX1378.nm	NA	NA	NA	NA
## XX1380.nm	NA	NA	NA	NA
## XX1382.nm	NA	NA	NA	NA
## XX1384.nm	NA	NA	NA	NA
## XX1386.nm	NA	NA	NA	NA
## XX1388.nm	NA	NA	NA	NA

## XX1390.nm	NA	NA	NA	NA
## XX1392.nm	NA	NA	NA	NA
## XX1394.nm	NA	NA	NA	NA
## XX1396.nm	NA	NA	NA	NA
## XX1398.nm	NA	NA	NA	NA
## XX1400.nm	NA	NA	NA	NA
## XX1402.nm	NA	NA	NA	NA
## XX1404.nm	NA	NA	NA	NA
## XX1406.nm	NA	NA	NA	NA
## XX1408.nm	NA	NA	NA	NA
## XX1410.nm	NA	NA	NA	NA
## XX1412.nm	NA	NA	NA	NA
## XX1414.nm	NA	NA	NA	NA
## XX1416.nm	NA	NA	NA	NA
## XX1418.nm	NA	NA	NA	NA
## XX1420.nm	NA	NA	NA	NA
## XX1422.nm	NA	NA	NA	NA
## XX1424.nm	NA	NA	NA	NA
## XX1426.nm	NA	NA	NA	NA
## XX1428.nm	NA	NA	NA	NA
## XX1430.nm	NA	NA	NA	NA
## XX1432.nm	NA	NA	NA	NA
## XX1434.nm	NA	NA	NA	NA
## XX1436.nm	NA	NA	NA	NA
## XX1438.nm	NA	NA	NA	NA
## XX1440.nm	NA	NA	NA	NA
## XX1442.nm	NA	NA	NA	NA
## XX1444.nm	NA	NA	NA	NA
## XX1446.nm	NA	NA	NA	NA
## XX1448.nm	NA	NA	NA	NA
## XX1450.nm	NA	NA	NA	NA
## XX1452.nm	NA	NA	NA	NA
## XX1454.nm	NA	NA	NA	NA
## XX1456.nm	NA	NA	NA	NA
## XX1458.nm	NA	NA	NA	NA
## XX1460.nm	NA	NA	NA	NA
## XX1462.nm	NA	NA	NA	NA
## XX1464.nm	NA	NA	NA	NA
## XX1466.nm	NA	NA	NA	NA
## XX1468.nm	NA	NA	NA	NA
## XX1470.nm	NA	NA	NA	NA
## XX1472.nm	NA	NA	NA	NA
## XX1474.nm	NA	NA	NA	NA
## XX1476.nm	NA	NA	NA	NA
## XX1478.nm	NA	NA	NA	NA
## XX1480.nm	NA	NA	NA	NA
## XX1482.nm	NA	NA	NA	NA
## XX1484.nm	NA	NA	NA	NA
## XX1486.nm	NA	NA	NA	NA
## XX1488.nm	NA	NA	NA	NA

## XX1490.nm	NA	NA	NA	NA
## XX1492.nm	NA	NA	NA	NA
## XX1494.nm	NA	NA	NA	NA
## XX1496.nm	NA	NA	NA	NA
## XX1498.nm	NA	NA	NA	NA
## XX1500.nm	NA	NA	NA	NA
## XX1502.nm	NA	NA	NA	NA
## XX1504.nm	NA	NA	NA	NA
## XX1506.nm	NA	NA	NA	NA
## XX1508.nm	NA	NA	NA	NA
## XX1510.nm	NA	NA	NA	NA
## XX1512.nm	NA	NA	NA	NA
## XX1514.nm	NA	NA	NA	NA
## XX1516.nm	NA	NA	NA	NA
## XX1518.nm	NA	NA	NA	NA
## XX1520.nm	NA	NA	NA	NA
## XX1522.nm	NA	NA	NA	NA
## XX1524.nm	NA	NA	NA	NA
## XX1526.nm	NA	NA	NA	NA
## XX1528.nm	NA	NA	NA	NA
## XX1530.nm	NA	NA	NA	NA
## XX1532.nm	NA	NA	NA	NA
## XX1534.nm	NA	NA	NA	NA
## XX1536.nm	NA	NA	NA	NA
## XX1538.nm	NA	NA	NA	NA
## XX1540.nm	NA	NA	NA	NA
## XX1542.nm	NA	NA	NA	NA
## XX1544.nm	NA	NA	NA	NA
## XX1546.nm	NA	NA	NA	NA
## XX1548.nm	NA	NA	NA	NA
## XX1550.nm	NA	NA	NA	NA
## XX1552.nm	NA	NA	NA	NA
## XX1554.nm	NA	NA	NA	NA
## XX1556.nm	NA	NA	NA	NA
## XX1558.nm	NA	NA	NA	NA
## XX1560.nm	NA	NA	NA	NA
## XX1562.nm	NA	NA	NA	NA
## XX1564.nm	NA	NA	NA	NA
## XX1566.nm	NA	NA	NA	NA
## XX1568.nm	NA	NA	NA	NA
## XX1570.nm	NA	NA	NA	NA
## XX1572.nm	NA	NA	NA	NA
## XX1574.nm	NA	NA	NA	NA
## XX1576.nm	NA	NA	NA	NA
## XX1578.nm	NA	NA	NA	NA
## XX1580.nm	NA	NA	NA	NA
## XX1582.nm	NA	NA	NA	NA
## XX1584.nm	NA	NA	NA	NA
## XX1586.nm	NA	NA	NA	NA
## XX1588.nm	NA	NA	NA	NA

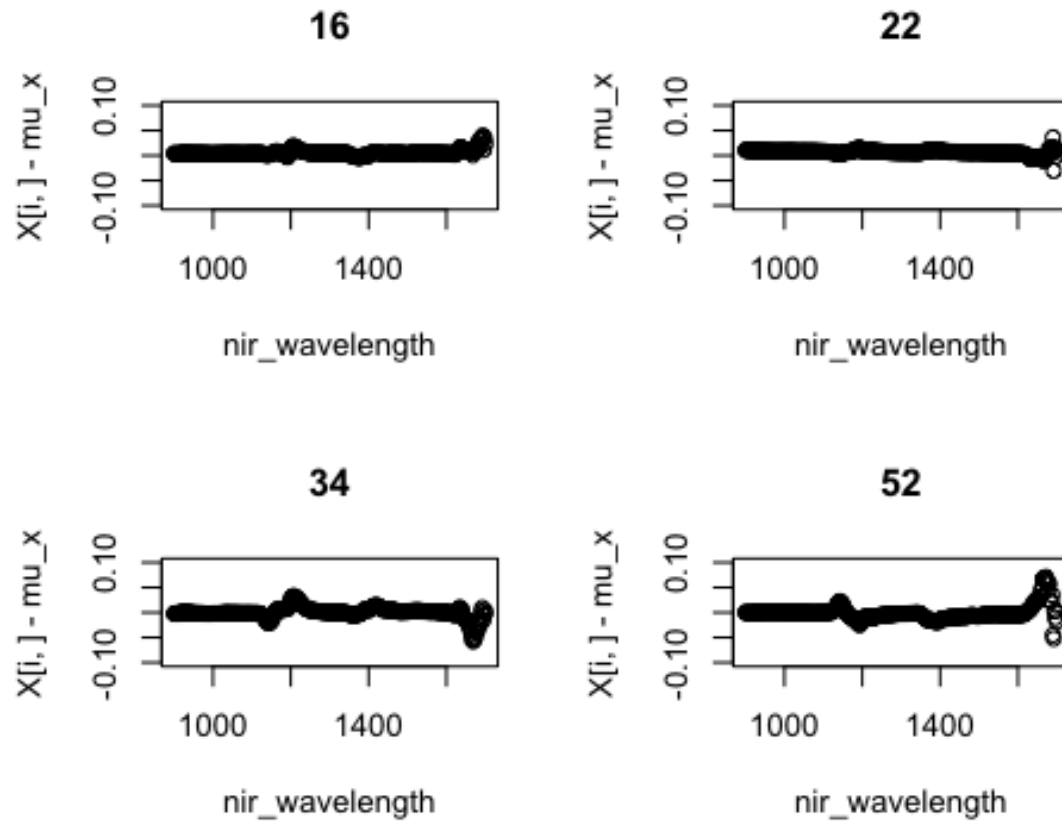
## XX1590.nm	NA	NA	NA	NA
## XX1592.nm	NA	NA	NA	NA
## XX1594.nm	NA	NA	NA	NA
## XX1596.nm	NA	NA	NA	NA
## XX1598.nm	NA	NA	NA	NA
## XX1600.nm	NA	NA	NA	NA
## XX1602.nm	NA	NA	NA	NA
## XX1604.nm	NA	NA	NA	NA
## XX1606.nm	NA	NA	NA	NA
## XX1608.nm	NA	NA	NA	NA
## XX1610.nm	NA	NA	NA	NA
## XX1612.nm	NA	NA	NA	NA
## XX1614.nm	NA	NA	NA	NA
## XX1616.nm	NA	NA	NA	NA
## XX1618.nm	NA	NA	NA	NA
## XX1620.nm	NA	NA	NA	NA
## XX1622.nm	NA	NA	NA	NA
## XX1624.nm	NA	NA	NA	NA
## XX1626.nm	NA	NA	NA	NA
## XX1628.nm	NA	NA	NA	NA
## XX1630.nm	NA	NA	NA	NA
## XX1632.nm	NA	NA	NA	NA
## XX1634.nm	NA	NA	NA	NA
## XX1636.nm	NA	NA	NA	NA
## XX1638.nm	NA	NA	NA	NA
## XX1640.nm	NA	NA	NA	NA
## XX1642.nm	NA	NA	NA	NA
## XX1644.nm	NA	NA	NA	NA
## XX1646.nm	NA	NA	NA	NA
## XX1648.nm	NA	NA	NA	NA
## XX1650.nm	NA	NA	NA	NA
## XX1652.nm	NA	NA	NA	NA
## XX1654.nm	NA	NA	NA	NA
## XX1656.nm	NA	NA	NA	NA
## XX1658.nm	NA	NA	NA	NA
## XX1660.nm	NA	NA	NA	NA
## XX1662.nm	NA	NA	NA	NA
## XX1664.nm	NA	NA	NA	NA
## XX1666.nm	NA	NA	NA	NA
## XX1668.nm	NA	NA	NA	NA
## XX1670.nm	NA	NA	NA	NA
## XX1672.nm	NA	NA	NA	NA
## XX1674.nm	NA	NA	NA	NA
## XX1676.nm	NA	NA	NA	NA
## XX1678.nm	NA	NA	NA	NA
## XX1680.nm	NA	NA	NA	NA
## XX1682.nm	NA	NA	NA	NA
## XX1684.nm	NA	NA	NA	NA
## XX1686.nm	NA	NA	NA	NA
## XX1688.nm	NA	NA	NA	NA

```
## XX1690.nm      NA      NA      NA      NA
## XX1692.nm      NA      NA      NA      NA
## XX1694.nm      NA      NA      NA      NA
## XX1696.nm      NA      NA      NA      NA
## XX1698.nm      NA      NA      NA      NA
## XX1700.nm      NA      NA      NA      NA
##
## Residual standard error: NaN on 0 degrees of freedom
## Multiple R-squared:      1, Adjusted R-squared:      NaN
## F-statistic:      NaN on 59 and 0 DF,  p-value: NA
```

The ordinary least squares method has an R^2 of 1 but many NA's. This is an absurdly overfit model. It would not have any predictive power.

As has been discussed before, the dataset has more features than observations, leading to an infinite number of possible solutions.

```
set.seed(1)
mu_x = colMeans(X)
nir_wavelength = seq(900, 1700, by=2)
par(mfrow=c(2,2))
for(i in sample.int(nrow(X), 4)) {
  plot(nir_wavelength, X[i,] - mu_x, main=i, ylim=c(-0.1,0.1))
}
```



The above are plots of near-infrared spectra from four random samples. The spectra are deviating from the mean significantly and in highly structured ways.

```
sigma_X = cor(X)
sigma_X[1:10,1:10]
```

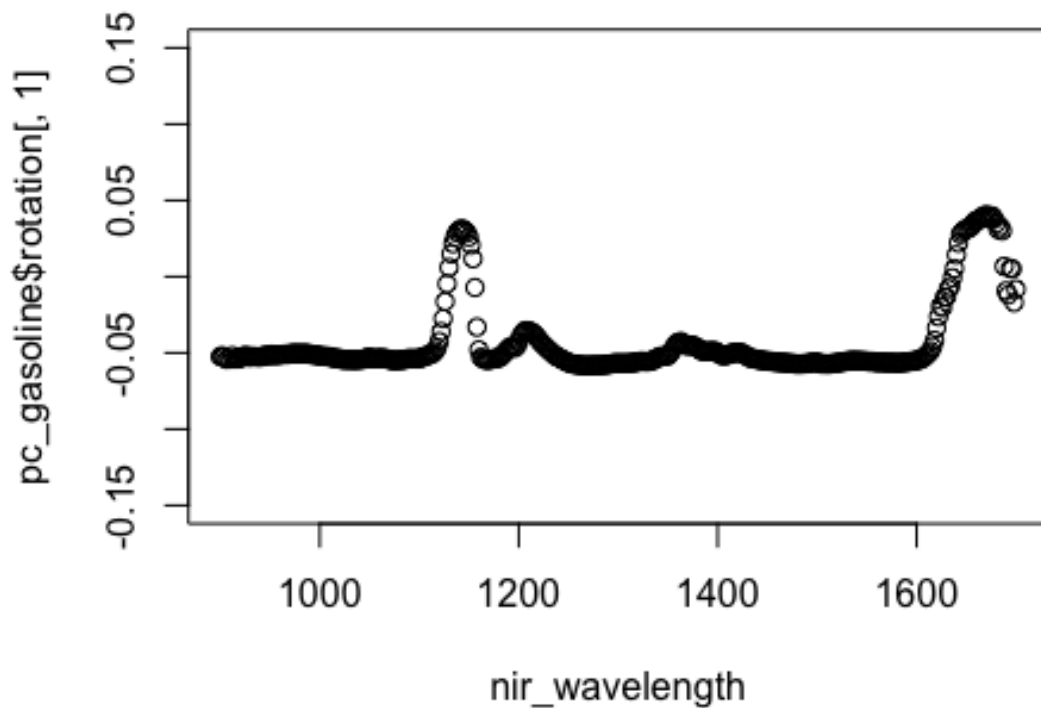
	X900.nm	X902.nm	X904.nm	X906.nm	X908.nm	X910.nm
## X900.nm	1.0000000	0.9946449	0.9932377	0.9822598	0.9781325	0.9694727
## X902.nm	0.9946449	1.0000000	0.9967327	0.9856213	0.9830355	0.9763599
## X904.nm	0.9932377	0.9967327	1.0000000	0.9902624	0.9891465	0.9844652
## X906.nm	0.9822598	0.9856213	0.9902624	1.0000000	0.9945963	0.9911308
## X908.nm	0.9781325	0.9830355	0.9891465	0.9945963	1.0000000	0.9961097
## X910.nm	0.9694727	0.9763599	0.9844652	0.9911308	0.9961097	1.0000000
## X912.nm	0.9650130	0.9723883	0.9774316	0.9800814	0.9910579	0.9920536
## X914.nm	0.9699421	0.9742834	0.9820720	0.9845183	0.9900193	0.9949961
## X916.nm	0.9809475	0.9809886	0.9855817	0.9864597	0.9886122	0.9882587
## X918.nm	0.9792130	0.9766522	0.9763888	0.9680528	0.9666960	0.9640762
##	X912.nm	X914.nm	X916.nm	X918.nm		
## X900.nm	0.9650130	0.9699421	0.9809475	0.9792130		
## X902.nm	0.9723883	0.9742834	0.9809886	0.9766522		
## X904.nm	0.9774316	0.9820720	0.9855817	0.9763888		
## X906.nm	0.9800814	0.9845183	0.9864597	0.9680528		
## X908.nm	0.9910579	0.9900193	0.9886122	0.9666960		


```
## X910.nm 0.9920536 0.9949961 0.9882587 0.9640762
## X912.nm 1.0000000 0.9893823 0.9849166 0.9536579
## X914.nm 0.9893823 1.0000000 0.9906084 0.9705865
## X916.nm 0.9849166 0.9906084 1.0000000 0.9849058
## X918.nm 0.9536579 0.9705865 0.9849058 1.0000000
```

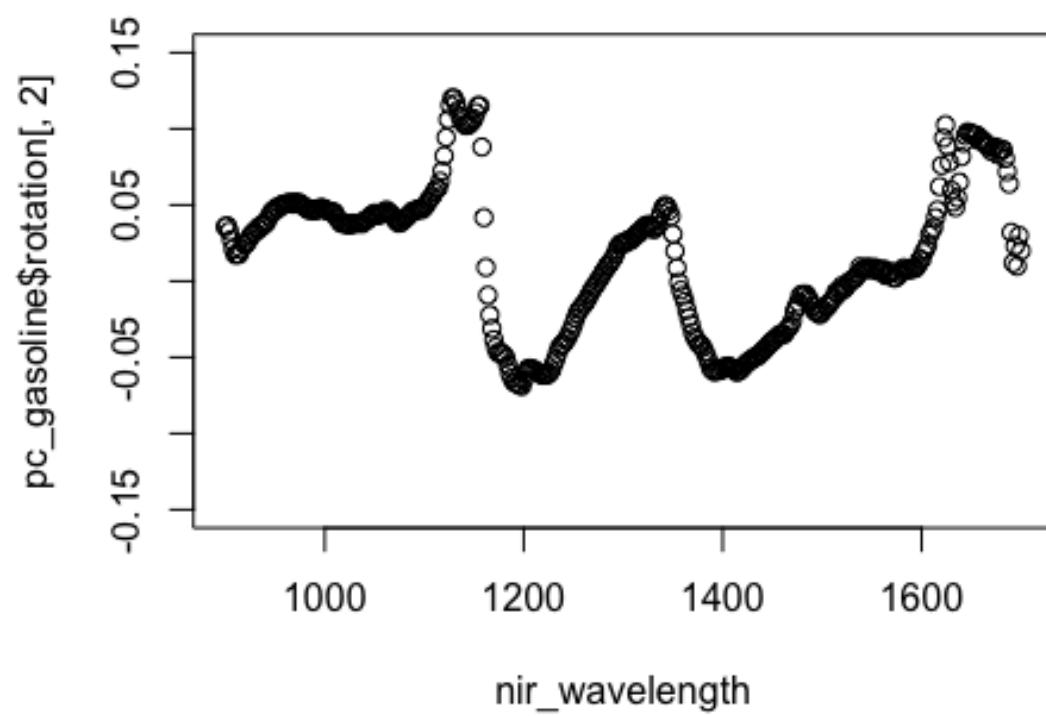
Correlation among features: correlation of the first 10 features: lines sigma_x (25, 26)
- “multicollinearity is really biting us here.”

The multicollinearity problem arises when two or more of the explanatory variables are close to being collinear. PCR can deal with such situations by excluding some of the low-variance principal components in the regression step.

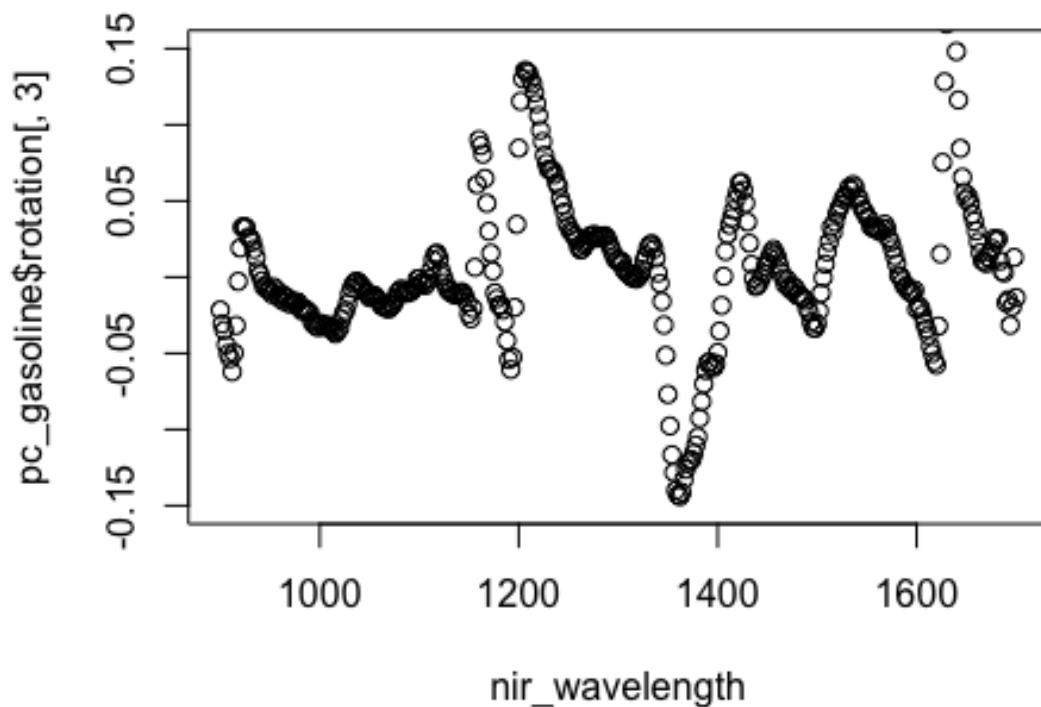
```
pc_gasoline = prcomp(X, scale=TRUE)
par(mfrow=c(1,1))
plot(nir_wavelength, pc_gasoline$rotation[,1], ylim=c(-0.15,0.15))
```



```
plot(nir_wavelength, pc_gasoline$rotation[,2], ylim=c(-0.15,0.15))
```



```
plot(nir_wavelength, pc_gasoline$rotation[,3], ylim=c(-0.15,0.15))
```



The above is the Step 1 we discussed earlier.

PC1: Below average in most wavelengths, with two peaks: 1150 and 1700
 “Interpretable signatures.”

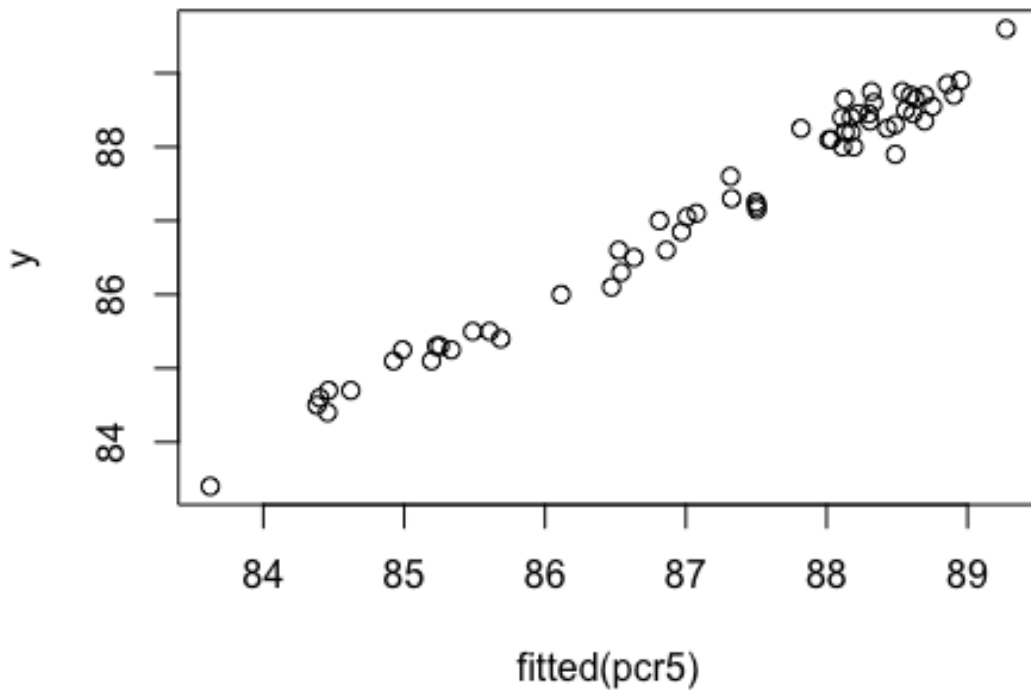
PC2 (second graph): more interesting looking, this is some other sort of signature.

```
K = 5
V = pc_gasoline$rotation[,1:K]
scores = X %%% V
pcr5 = lm(y ~ scores)
summary(pcr5)

##
## Call:
## lm(formula = y ~ scores)
##
## Residuals:
```

	Min	1Q	Median	3Q	Max
##	-0.58797	-0.16758	0.01553	0.15580	0.52314

```
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  99.3743     1.8300  54.302 < 2e-16 ***
## scoresPC1    0.0649     0.4000   0.162 0.871705
## scoresPC2    6.8278     0.5492  12.432 < 2e-16 ***
## scoresPC3   -25.3141     0.5979 -42.336 < 2e-16 ***
## scoresPC4    2.2640     1.3885   1.631 0.108801
## scoresPC5   -4.1484     1.0808  -3.838 0.000327 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.2319 on 54 degrees of freedom
## Multiple R-squared:  0.979, Adjusted R-squared:  0.977
## F-statistic:  503 on 5 and 54 DF,  p-value: < 2.2e-16
plot(fitted(pcr5), y)
```



Taking $K = 5$ gives us an R^2 value of 97.9 . The variables PC2, PC3 and PC5 are significant with very low p values.

Note:

%*% = R's notation for vector or matrix multiplication.

The first K components are obtained from D. The value of K = 5 was chosen arbitrarily. Cross validation should be used to pick K.

```
K = 1
V1 = pc_gasoline$rotation[,1:K]
scores1 = X %*% V1
pcr1 = lm(y ~ scores1)
summary(pcr1)

##
## Call:
## lm(formula = y ~ scores1)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -2.5610 -1.1634  0.2324  1.1455  3.0992
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)   87.748      0.257  341.472  < 2e-16 ***
## scores1       4.626      1.456   3.176  0.00239 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.424 on 58 degrees of freedom
## Multiple R-squared:  0.1482, Adjusted R-squared:  0.1335
## F-statistic: 10.09 on 1 and 58 DF,  p-value: 0.002389
```

K = 1 -> $R^2 = 14.8\%$

```
K = 2
V2 = pc_gasoline$rotation[,1:K]
scores2 = X %*% V2
pcr2 = lm(y ~ scores2)
summary(pcr2)

##
## Call:
## lm(formula = y ~ scores2)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -2.5942 -1.0613  0.2316  0.7734  3.5048
##
```

```
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)   84.922      1.432   59.307  <2e-16 ***
## scores2PC1     1.156      2.239    0.517   0.6074
## scores2PC2     4.295      2.142    2.005   0.0498 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.389 on 57 degrees of freedom
## Multiple R-squared:  0.2043, Adjusted R-squared:  0.1764
## F-statistic: 7.317 on 2 and 57 DF,  p-value: 0.001485
```

K = 2 -> $R^2 = 20.4\%$

```
K = 3
V3 = pc_gasoline$rotation[,1:K]
scores3 = X %>% V3
pcr3 = lm(y ~ scores3)
summary(pcr3)

##
## Call:
## lm(formula = y ~ scores3)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.49568 -0.19652 -0.03459  0.15126  0.68871
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)   95.3705      0.3987 239.230  <2e-16 ***
## scores3PC1     0.1582      0.4446   0.356   0.723
## scores3PC2     7.0836      0.4313 16.424  <2e-16 ***
## scores3PC3    -25.2991      0.6776 -37.336  <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.2753 on 56 degrees of freedom
## Multiple R-squared:  0.9693, Adjusted R-squared:  0.9676
## F-statistic: 588.7 on 3 and 56 DF,  p-value: < 2.2e-16
```

K = 3 -> $R^2 = 96.9\%$ (wow!)

Q: Why is PC3 so crucial in increasing the value of R^2 ? Isn't K = 1 i.e PC1 the most important component with the highest proportion of variation?

A: PC1 has highest proportion of variation with respect to the original variables x but doesn't explain anything with respect to y. PC3 has the most power to forecast y.

PCA answers the question:

“Which direction in x explains the most variation in x?”

PCR answers the question

“Which direction in x explains the most variation in y?”

Thus PC1 explained variation in x but not in y. PC3 explains the most variation in y and has a higher predictive power

You could also use lasso regression or stepwise selection to find the principal components with the highest predictive power for y.

Recount: Step one was about generating synthetic features and reducing the number of variables to a few principal components. Step two was running the regression.

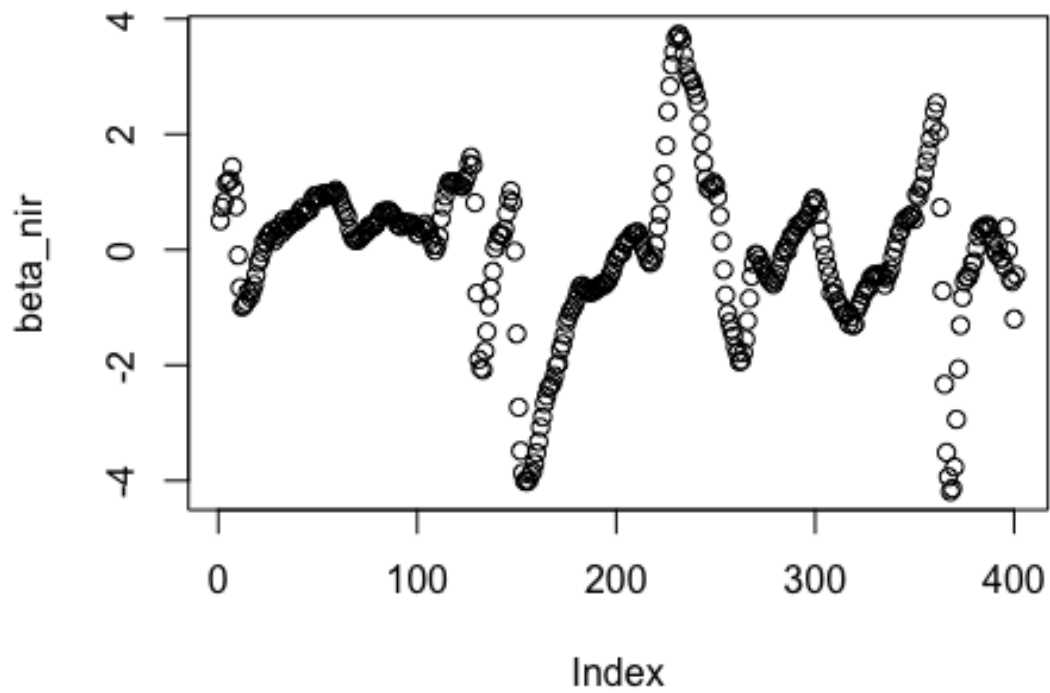
Partial least squares method was mentioned to address some issues in PC regression. Partial least squares finds the direction in x which explains variations in both x and y.

Q. If we were to take in Lasso, do we start with K or go back to D?

A. It depends on how many features you have in D and how many in K; you can certainly start Lasso at D, unless of course D is too big.

```
# Express the coefficients in terms of the original variables
beta_pcr = coef(pcr5)[-1]
beta_nir = rowSums(V %*% diag(beta_pcr))

plot(beta_nir)
```



Sources and Links

- [Wikipedia: Principal component analysis](#)
- [Wikipedia: Near-infrared spectroscopy](#)
- [Wikipedia: Principal component regression](#)
- [Wikipedia: Factor analysis](#)