

Principal Components Regression

Predictive Modeling & Statistical Learning

Gaston Sanchez

CC BY-SA 4.0

Data set cars2004

Data cars2004

Data file cars2004 in the data/ folder of the github repo

	price	engine	cyl	hp	city_mpg
Acura 3.5 RL 4dr	43755	3.5	6	225	18
Acura 3.5 RL w/Navigation 4dr	46100	3.5	6	225	18
Acura MDX	36945	3.5	6	265	17
Acura NSX coupe 2dr manual S	89765	3.2	6	290	17
Acura RSX Type S 2dr	23820	2.0	4	200	24
Acura TL 4dr	33195	3.2	6	270	20
	hwy_mpg	weight	wheel	length	width
Acura 3.5 RL 4dr	24	3880	115	197	72
Acura 3.5 RL w/Navigation 4dr	24	3893	115	197	72
Acura MDX	23	4451	106	189	77
Acura NSX coupe 2dr manual S	24	3153	100	174	71
Acura RSX Type S 2dr	31	2778	101	172	68
Acura TL 4dr	28	3575	108	186	72

Data cars2004

```
'data.frame': 385 obs. of 10 variables:  
 $ price      : int  43755 46100 ...  
 $ engine     : num   3.5 3.5 ...  
 $ cyl        : int   6 6 ...  
 $ hp         : int  225 225 ...  
 $ city_mpg   : int   18 18 ...  
 $ hwy_mpg    : int   24 24 ...  
 $ weight     : int  3880 3893 ...  
 $ wheel      : int  115 115 ...  
 $ length     : int  197 197 ...  
 $ width      : int   72 72 ...
```

Predicting Price of cars

- ▶ Response: price
- ▶ Predictors: cyl, hp, ..., length, and width

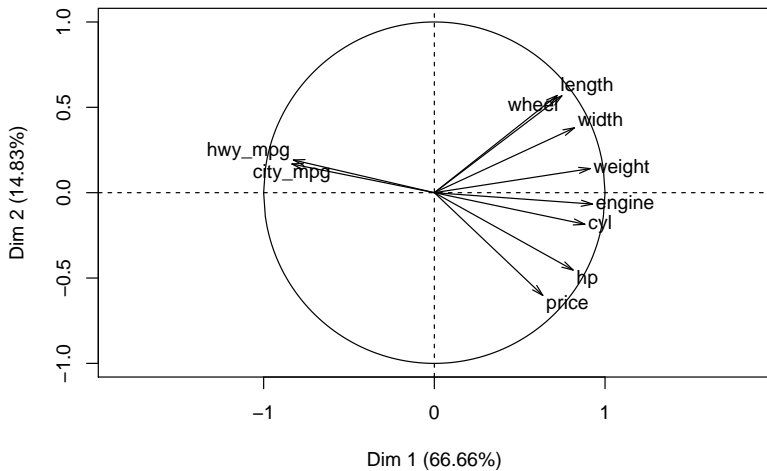
$$\text{price} = \beta_0 + \beta_1 \text{ cyl} + \beta_2 \text{ hp} + \dots + \beta_9 \text{ width} + \varepsilon$$

Correlation Matrix

	engine	cyl	hp	city_mpg	hwy_mpg	weight	wheel	length	width
price	0.6	0.654	0.836	-0.485	-0.469	0.476	0.204	0.210	0.314
engine		0.912	0.778	-0.706	-0.708	0.812	0.631	0.624	0.727
cyl			0.792	-0.670	-0.664	0.731	0.553	0.547	0.621
hp				-0.672	-0.652	0.631	0.396	0.381	0.500
city_mpg					0.941	-0.736	-0.481	-0.468	-0.590
hwy_mpg						-0.789	-0.455	-0.390	-0.585
weight							0.751	0.653	0.808
wheel								0.867	0.760
length									0.752

PCA

Variables factor map (PCA)



PCA: Eigenvalues

	eigenvalues	proportion	cum_prop
comp1	6.30	70.04	70.04
comp2	1.21	13.43	83.47
comp3	0.61	6.76	90.23
comp4	0.28	3.06	93.30
comp5	0.21	2.37	95.67
comp6	0.19	2.13	97.79
comp7	0.09	0.99	98.79
comp8	0.07	0.79	99.58
comp9	0.04	0.42	100.00

Predicting Price of cars

$$\text{price} = \beta_0 + \beta_1 \text{cyl} + \beta_2 \text{hp} + \dots + \beta_9 \text{width} + \varepsilon$$

OLS regression:

```
ols_reg <- lm(price ~ ., data = cars2004)
```

```
ols_reg_sum <- summary(ols_reg)
```

OLS Regression

Regress price on 9 predictors

	Estimate	Std. Error	t value	Pr(> t)
(Intercept)	32536.025	17777.488	1.8302	6.802e-02
engine	-3273.053	1542.595	-2.1218	3.451e-02
cyl	2520.927	896.202	2.8129	5.168e-03
hp	246.595	13.201	18.6797	1.621e-55
city_mpg	-229.987	332.824	-0.6910	4.900e-01
hwy_mpg	979.967	345.558	2.8359	4.817e-03
weight	9.937	2.045	4.8584	1.741e-06
wheel	-695.392	172.896	-4.0220	6.980e-05
length	33.690	89.660	0.3758	7.073e-01
width	-635.382	306.344	-2.0741	3.875e-02

```
Call:
lm(formula = price ~ ., data = cars2004)
```

```
Residuals:
```

Min	1Q	Median	3Q	Max
-21534	-5411	-352	4054	92763

```
Coefficients:
```

	Estimate	Std. Error	t value	Pr(> t)
(Intercept)	32536.025	17777.488	1.830	0.06802 .
engine	-3273.053	1542.595	-2.122	0.03451 *
cyl	2520.927	896.202	2.813	0.00517 **
hp	246.595	13.201	18.680	< 2e-16 ***
city_mpg	-229.987	332.824	-0.691	0.48998
hwy_mpg	979.967	345.558	2.836	0.00482 **
weight	9.937	2.045	4.858	1.74e-06 ***
wheel	-695.392	172.896	-4.022	6.98e-05 ***
length	33.690	89.660	0.376	0.70731
width	-635.382	306.344	-2.074	0.03875 *

```
---
```

```
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
Residual standard error: 10110 on 375 degrees of freedom
```

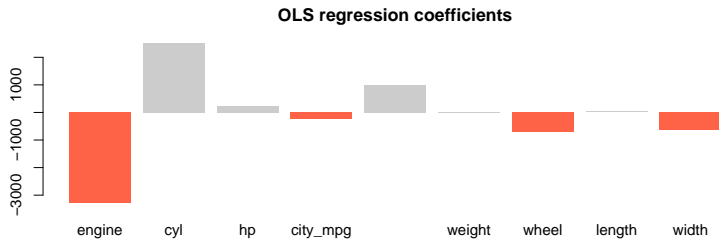
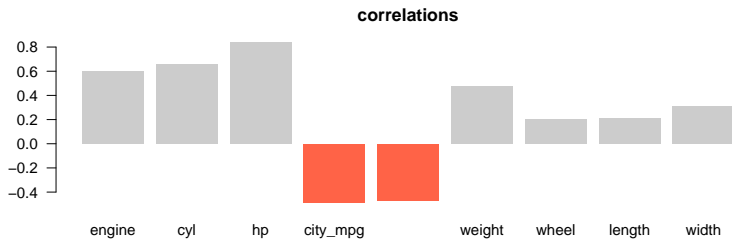
```
Multiple R-squared:  0.745, Adjusted R-squared:  0.7389
```

```
F-statistic: 121.7 on 9 and 375 DF,  p-value: < 2.2e-16
```

Correlations and OLS Coefficients

	correlation	coefficient
engine	0.5997873	-3273.05304
cyl	0.6544123	2520.92691
hp	0.8360930	246.59496
city_mpg	-0.4854130	-229.98735
hwy_mpg	-0.4694315	979.96656
weight	0.4760867	9.93652
wheel	0.2035464	-695.39157
length	0.2096682	33.69009
width	0.3135383	-635.38224

Some correlation signs don't match regression coeff. signs



Principal Components Regression (PCR)

Uses of PCA

PCA can be used to ...

- ▶ Reduce the dimensionality of a data set (i.e. reduce the number of variables)
- ▶ For data visualization and exploration purposes to obtain:
 - a map to visualize the objects in terms of their proximities, and
 - another map to visualize the variables in terms of their correlations.
- ▶ Summarize the systematic patterns of variation within observations, within variables, and between observations and variables.

More uses of PCA

When fitting a regression model:

$$\hat{\mathbf{y}} = \mathbf{X}(\mathbf{X}^T \mathbf{X})^{-1} \mathbf{X}^T \mathbf{y}$$

We can also use PCA as a *pre-processing* step on the predictors before fitting a regression equation.

PCA Regression Idea

Decompose the predictors data matrix \mathbf{X} into PC's, and then regress \mathbf{y} onto the PCs

PCA Formula Decomposition

Decompose the predictors data matrix \mathbf{X} into PC's

$$\mathbf{X} = \mathbf{ZV}^T$$

where:

- ▶ \mathbf{Z} is the matrix of PCs
- ▶ \mathbf{V} is the matrix of loadings

Without loss of generality suppose the predictors and response are standardized.

PC Regression

Regress y onto the PC's:

$$\hat{y} = \hat{\beta}_1 \mathbf{z}_1 + \hat{\beta}_2 \mathbf{z}_2 + \cdots + \hat{\beta}_p \mathbf{z}_p$$

where:

- ▶ \mathbf{z}_j is the j -th PC
- ▶ $\hat{\beta}_j$ is the estimate of the j -th PCR-coefficient

PC Regression

In matrix notation:

$$\hat{\mathbf{y}} = \mathbf{Z}\hat{\boldsymbol{\beta}}_{PCR}$$

The vector of PCR coefficients is obtained via OLS:

$$\hat{\boldsymbol{\beta}}_{PCR} = (\mathbf{Z}^T \mathbf{Z})^{-1} \mathbf{Z}^T \mathbf{y}$$

PC Regression

	Estimate	Std. Error	t value	Pr(> t)
(Intercept)	0.000	515.0450	0.0000	1.0000
PC1	-4470.761	205.4021	-21.7659	0.0000
PC2	7608.419	469.1001	16.2192	0.0000
PC3	-9650.324	661.0626	-14.5982	0.0000
PC4	-1768.547	981.9554	-1.8010	0.0725
PC5	10528.146	1116.5683	9.4290	0.0000
PC6	-5593.736	1179.2391	-4.7435	0.0000
PC7	-5746.452	1723.8664	-3.3335	0.0009
PC8	-7606.196	1928.9437	-3.9432	0.0001
PC9	5473.090	2663.9282	2.0545	0.0406

PC Regression

Usually, you don't use all p PCs, but just $k < p$ of them i.e.

$$\underset{(n,p)}{\hat{\mathbf{X}}} = \underset{(n,k)}{\mathbf{Z}} \underset{(k,p)}{\mathbf{V}^T}$$

The regression of \mathbf{y} onto the k PCs is:

$$\hat{\mathbf{y}} = \hat{\beta}_1 \mathbf{z}_1 + \hat{\beta}_2 \mathbf{z}_2 + \cdots + \hat{\beta}_k \mathbf{z}_k$$

Regressions with few PCs

Regression with first two PCs

	Estimate	Std. Error	t value	Pr(> t)
(Intercept)	0.000	713.7648	0.0000	1
PC1	-4470.761	284.6524	-15.7060	0
PC2	7608.419	650.0930	11.7036	0

Regressions with few PCs

Regression with first two PCs

	Estimate	Std. Error	t value	Pr(> t)
(Intercept)	0.000	713.7648	0.0000	1
PC1	-4470.761	284.6524	-15.7060	0
PC2	7608.419	650.0930	11.7036	0

Regression with first three PCs

	Estimate	Std. Error	t value	Pr(> t)
(Intercept)	0.000	602.0143	0.0000	1
PC1	-4470.761	240.0858	-18.6215	0
PC2	7608.419	548.3113	13.8761	0
PC3	-9650.324	772.6881	-12.4893	0

PC Regression property

Because of uncorrelatedness, the contributions and estimated coefficient of a PC are unaffected by which other PCs are also included in the regression.

How does PCR work?

PC Regression Idea

Decompose the predictors data matrix \mathbf{X} into PC's

$$\mathbf{X} = \mathbf{Z}\mathbf{V}^T$$

Replace \mathbf{X} with $\mathbf{Z}\mathbf{V}^T$ in the fitted values:

$$\hat{\mathbf{y}} = \mathbf{X}(\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\mathbf{y}$$

OLS Regression in terms of PCs

$$\begin{aligned}\hat{\mathbf{y}} &= \mathbf{X}(\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\mathbf{y} \\ &= \mathbf{ZV}^T ((\mathbf{ZV}^T)^T\mathbf{ZV}^T)^{-1} (\mathbf{ZV}^T)^T\mathbf{y} \\ &= \mathbf{ZV}^T (\mathbf{VZ}^T\mathbf{ZV}^T)^{-1} \mathbf{VZ}^T\mathbf{y} \\ &= \mathbf{ZV}^T(\mathbf{V}\mathbf{\Lambda}\mathbf{V}^T)^{-1}\mathbf{VZ}^T\mathbf{y} \\ &= \mathbf{ZV}^T(\mathbf{V}\mathbf{\Lambda}^{-1}\mathbf{V}^T)\mathbf{VZ}^T\mathbf{y} \\ &= \mathbf{Z}\mathbf{\Lambda}^{-1}\mathbf{Z}^T\mathbf{y} \\ &= \mathbf{Z}\hat{\boldsymbol{\beta}}_{PCR}\end{aligned}$$

OLS and PCR coefficients

Relationship between OLS coefficients and PCR coefficients when using all PCs, that is, $\mathbf{X} = \mathbf{ZV}^T$:

$$\begin{aligned}\hat{\mathbf{y}} &= \mathbf{X}\hat{\boldsymbol{\beta}}_{OLS} \\ &= \mathbf{ZV}^T\hat{\boldsymbol{\beta}}_{OLS} \\ &= \mathbf{Z}\hat{\boldsymbol{\beta}}_{PCR}\end{aligned}$$

OLS and PCR coefficients

Relationship between OLS coefficients and PCR coefficients when using all PCs, that is, $\mathbf{X} = \mathbf{ZV}^T$:

$$\begin{aligned}\hat{\mathbf{y}} &= \mathbf{X}\hat{\boldsymbol{\beta}}_{OLS} \\ &= \mathbf{ZV}^T\hat{\boldsymbol{\beta}}_{OLS} \\ &= \mathbf{Z}\hat{\boldsymbol{\beta}}_{PCR}\end{aligned}$$

$$\begin{aligned}\hat{\mathbf{y}} &= \mathbf{Z}\hat{\boldsymbol{\beta}}_{PCR} \\ &= \mathbf{XV}\hat{\boldsymbol{\beta}}_{PCR} \\ &= \mathbf{X}\hat{\boldsymbol{\beta}}_{OLS}\end{aligned}$$

Notice when using all PCs, OLS $\hat{\mathbf{y}}$ is the same as PCR $\hat{\mathbf{y}}$

PCR Coefficients transition formula

In general, you can always reexpress the PCR coefficients in terms of the original variables:

$$\begin{aligned}\hat{\mathbf{y}} &= \mathbf{Z}\hat{\boldsymbol{\beta}}_{PCR} \\ &= \mathbf{XV}\hat{\boldsymbol{\beta}}_{PCR}\end{aligned}$$

by premultiplying $\hat{\boldsymbol{\beta}}_{PCR}$ by the loadings \mathbf{V}

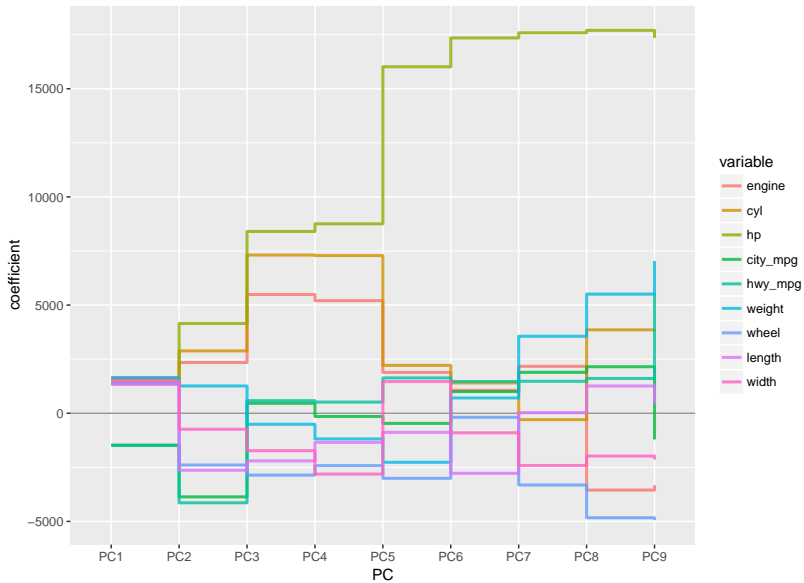
PC Regression Coefficients

Regression coefficients in terms of the original variables, for each of the 9 PC regressions:

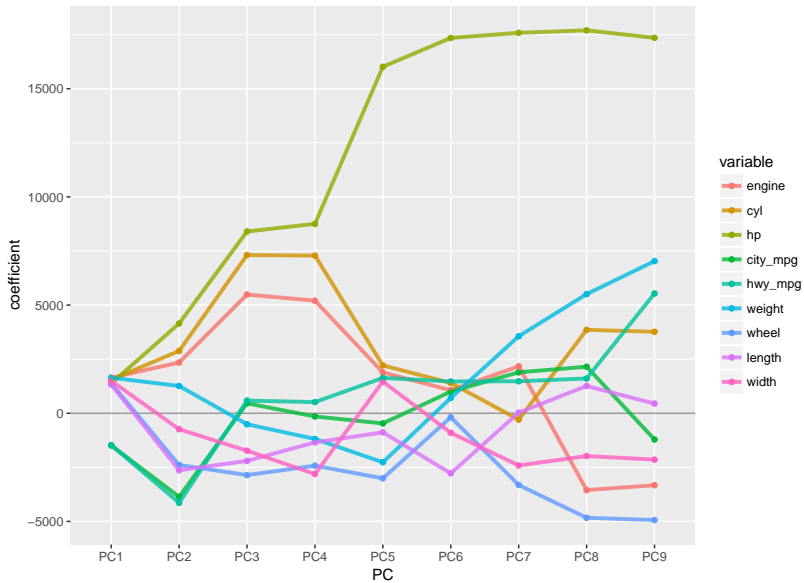
	PC1	PC2	PC3	PC4	PC5	PC6	PC7	PC8	PC9
engine	1641	2345	5487	5203	1887	1064	2168.5	-3551	-3327
cyl	1544	2881	7312	7289	2213	1401	-295.5	3856	3766
hp	1377	4148	8402	8755	16016	17348	17585.7	17700	17352
city_mpg	-1487	-3864	459	-149	-470	999	1891.3	2151	-1213
hwy_mpg	-1472	-4140	583	516	1633	1468	1477.9	1607	5536
weight	1642	1261	-510	-1187	-2266	707	3557.7	5507	7033
wheel	1385	-2392	-2860	-2422	-3009	-187	-3315.7	-4832	-4938
length	1332	-2634	-2202	-1346	-883	-2778	28.2	1260	447
width	1501	-738	-1731	-2811	1464	-904	-2413.8	-1978	-2142

The solution with 9 PCs matches the OLS solution.

Path of PCR Standardized Coefficients



Path of PCR Standardized Coefficients



PCR with all PCs equals OLS solution

With \mathbf{y} and \mathbf{X} standardized, the OLS solution:

Xengine	Xcyl	Xhp	Xcity_mpg	Xhwy_mpg	Xweight	Xwheel
-3326.7904	3766.1495	17352.2185	-1213.0142	5535.9355	7032.5383	-4937.8106
Xlength	Xwidth					
446.9752	-2141.8196					

matches the PCR solution using all PCs:

engine	cyl	hp	city_mpg	hwy_mpg	weight	wheel
-3326.7904	3766.1495	17352.2185	-1213.0142	5535.9355	7032.5383	-4937.8106
length	width					
446.9752	-2141.8196					

How to choose an
“optimal” number of PCs?

Number of PCs

- ▶ The main challenge in PCR is the choice for k , the number of PCs
- ▶ k is also known as the tuning parameter
- ▶ k is usually chosen via cross-validation

Regression with PC1

$$\hat{\mathbf{y}} = \hat{\beta}_1 \mathbf{z}_1 = \mathbf{X}(\hat{\beta}_1 \mathbf{v}_1)$$

	correlation	coefficient
engine	0.5997873	1640.623
cyl	0.6544123	1543.678
hp	0.8360930	1376.699
city_mpg	-0.4854130	-1487.103
hwy_mpg	-0.4694315	-1471.922
weight	0.4760867	1641.840
wheel	0.2035464	1385.197
length	0.2096682	1331.796
width	0.3135383	1500.565

Regression with PC1 and PC2

$$\hat{y} = \hat{\beta}_1 \mathbf{z}_1 + \hat{\beta}_2 \mathbf{z}_2 = \mathbf{XV}_{1:2} \hat{\boldsymbol{\beta}}_{1:2}$$

	correlation	coefficient
engine	0.5997873	2345.1254
cyl	0.6544123	2881.3726
hp	0.8360930	4148.0459
city_mpg	-0.4854130	-3864.4992
hwy_mpg	-0.4694315	-4139.9870
weight	0.4760867	1260.7474
wheel	0.2035464	-2391.5824
length	0.2096682	-2634.3174
width	0.3135383	-738.1835

Number of PCs via Cross-Validation

- ▶ Optionally: divide the data into training and test sets.
- ▶ If you don't have a test set, then all your data is the training set.
- ▶ Divide the training set into m parts (of similar size).
- ▶ For each possible number k of PCs do:
 - For each part, use the rest of the data to fit the PC regression, and use the part to compute the MSE.
 - Repeat for each part, and average the MSE's.
- ▶ Select the number of PCs with smallest MSE.

PCR Coefficients

PCR without training-and-test sets:

```
library(pls)
set.seed(2)

pcr_fit <- pcr(price ~ ., data = cars2004, scale = TRUE, validation = "CV")

summary(pcr_fit)

validationplot(pcr_fit, val.type = "MSEP")
```

`pcr()` standardizes the predictors, but not the response

PCR Coefficients

Data: X dimension: 385 9
Y dimension: 385 1
Fit method: svdpc
Number of components considered: 9

VALIDATION: RMSEP

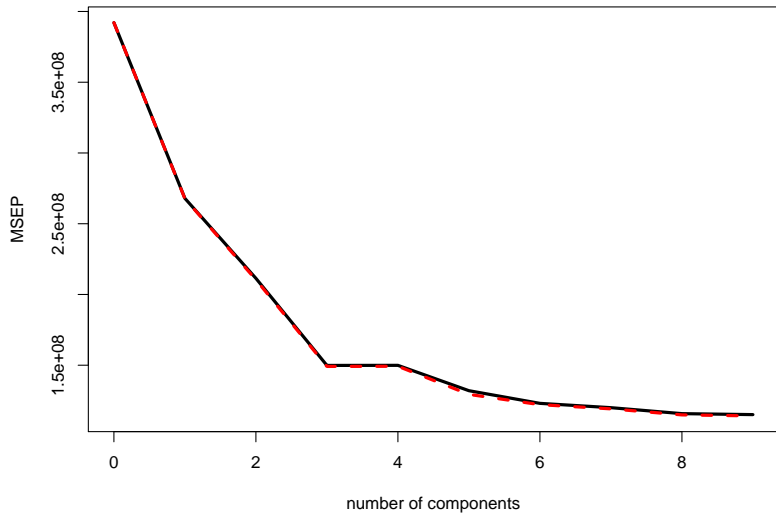
Cross-validated using 10 random segments.

	(Intercept)	1 comps	2 comps	3 comps	4 comps	5 comps	6 comps	7 comps	8 comps
CV	19802	16369	14540	12242	12244	11489	11092	10921	10921
adjCV	19802	16363	14506	12212	12217	11374	11051	10921	10921
	8 comps	9 comps							
CV	10760	10728							
adjCV	10719	10686							

TRAINING: % variance explained

	1 comps	2 comps	3 comps	4 comps	5 comps	6 comps	7 comps	8 comps	9 comps
X	70.04	83.47	90.23	93.30	95.67	97.79	98.79	99.58	99.58
price	32.22	50.11	64.60	64.82	70.87	72.40	73.15	74.21	74.21

price



PCR Coefficients

PCR with training-and-test sets:

```
# train set
set.seed(1)
train <- sample(c(TRUE, FALSE), nrow(cars2004), replace = TRUE)
test <- (!train)

pcr_fit <- pcr(price ~ ., data = cars2004, subset = train,
               scale = TRUE, validation = "CV")

summary(pcr_fit)

validationplot(pcr_fit, val.type = "MSEP")
```

PCR Coefficients

Data: X dimension: 208 9
Y dimension: 208 1
Fit method: svdpc
Number of components considered: 9

VALIDATION: RMSEP

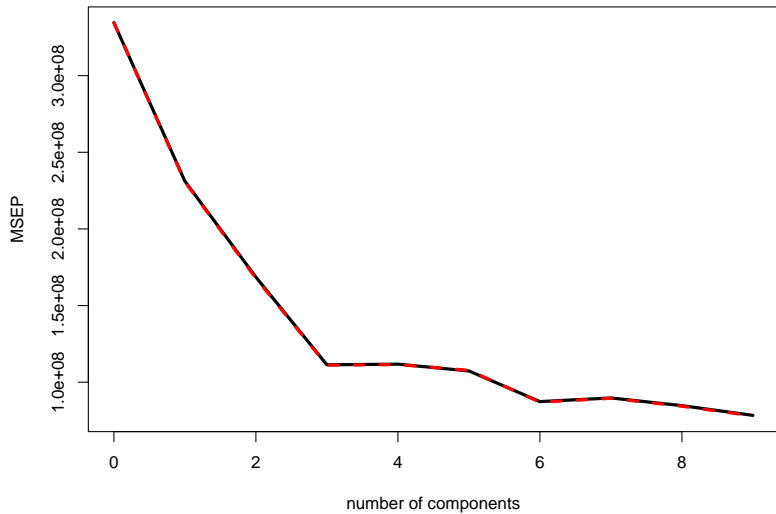
Cross-validated using 10 random segments.

	(Intercept)	1 comps	2 comps	3 comps	4 comps	5 comps	6 comps	7 comps	8 comps
CV	18291	15203	12983	10552	10572	10361	9346	94	94
adjCV	18291	15190	12952	10540	10560	10381	9327	94	94
	8 comps	9 comps							
CV	9203	8852							
adjCV	9182	8832							

TRAINING: % variance explained

	1 comps	2 comps	3 comps	4 comps	5 comps	6 comps	7 comps	8 comps	9 comps
X	68.61	82.49	90.05	93.57	95.98	97.91	98.80	99.58	99.98
price	32.49	53.06	67.70	67.80	68.57	75.55	75.66	76.96	77.06

price

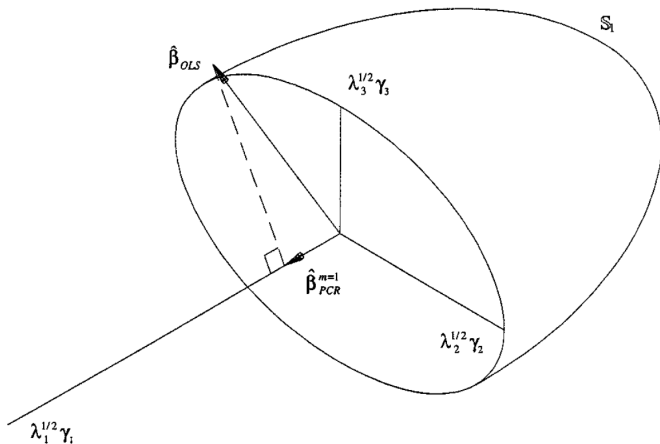


Number of PCs via Cross-Validation

- ▶ PCR with CV suggests using all $k = 9$ PCs.
- ▶ The PCR solution coincides with the OLS.
- ▶ There is not really an advantage of PCR in this particular example.
- ▶ As this example shows: the PCs with large variation may not necessarily be good predictors.
- ▶ However, a regression on PC1 may still be an interesting case.

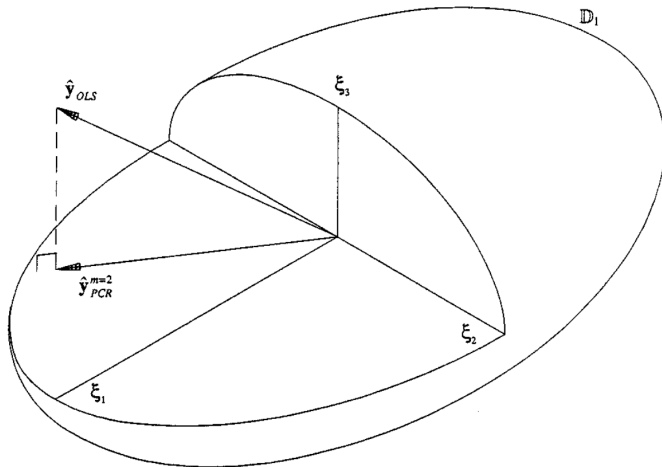
Geometry of PCR

PCR Geometry



PCR fitted vector with first PC (Phatak & de Jong, 1997)

PCR Geometry



PCR fitted vector with two first PCs (Phatak & de Jong, 1997)

PCR Geometry

“The geometric interpretation of principal component regression is quite straightforward. $\hat{\beta}_{PCR}^m$ is simply the orthogonal projection of $\hat{\beta}_{OLS}$ onto the subspace spanned by the eigenvectors corresponding to the PCs retained in the regression.”

(Phatak & de Jong, 1997)

PC Regression rationale

- ▶ The main advantage of PC regression occurs when multicollinearities are present.
- ▶ Finding regression coefficients is more straightforward with \mathbf{Z} because its columns are orthogonal.
- ▶ Calculation of OLS estimates via PC regression may be numerically more stable.
- ▶ PCR assumes that the new predictors (i.e. the PCs), explain the response, especially those PCs with larger variance.

PC Regression Issues

- ▶ The first PCs may not be necessarily related to the response.
- ▶ PCR is not a feature selection method.
- ▶ Each of the PCs used in the regression is a linear combination of all p predictor variables.
- ▶ PCR is not scaling-invariant.

References

- ▶ **Principal Component Analysis** by Ian Jolliffe (2004). *Chapter 8: Principal Components in Regression Analysis*. Springer.
- ▶ **Linear Models with R** by Julian J. J. Faraway (2015). *Chapter 11: Shrinkage Methods*. CRC Press.
- ▶ **Modern Multivariate Statistical Techniques** by Julian J. J. Izenman (2008). *Chapter 5, sec 6: Biased Regression Methods*. Springer.
- ▶ **Modern Regression Methods** by Thomas Ryan (1997). *Chapter 12, sec 9: Other Biased Estimators*. Wiley.
- ▶ **The Geometry of Partial Least Squares** by Alok Phatak and Sijmen de Jong (1997). *Journal of Chemometrics, Vol. 11, 311-338*.