# Unit 3 Dimensionality Reduction Techniques

## (1) Principal Components Regression (PCR)

October 30, 2019

* Explain most of variability using the smaller # of predictors.
(Find a tool to find representation of a data that contains as much of the variation as possible)

### 1. Introduction

coefficients ← OLS estimates which have relatively low bias and low variability especially when the relationship between the response and predictors is linear and $n >> p$

if $n$ is NOT much larger than $p$,

- OLS fit can have high variance ⇒ may result in over-fitting and poor estimates.
- many predictors have no or little effects on the respone.
  ↑
  ① irrelevant variable leads to unnecessary complexity
  ② harder to see the effect of the important variables.

∴ It is better to remove the irrelevant variables from the model.

How?

① variable selection < stepwise / best-subset.

: Identify a subset of the p-predictors that we believe to be related to the response; then fit a model using OLS on the reduced set.

② Dimension Reduction < PCR (principal component regression) / PLS (Partial Least squares)

: Involves projecting the p-predictors into a M-dimensional subspace, where $M < p$, and fit the linear regression model using M projections as predictors.

③ Shrinkage (Regulation) < Ridge / Lasso

: Involves shrinking the estimated coefficients toward '0' relative to the OLS estimates. has the effect of reducing variance and performs variable selection.

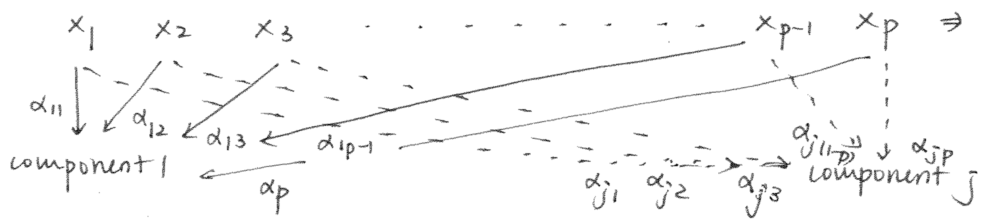
2. Performing Principal Components Regression (PCR)

3.2.1 Principal Components Analysis (PCA) : summarize the information in the predictors into a smaller set of $\boxed{\text{variables}}$ and then try to predict Y

→ linear combinations of the original variables.

Assume we have a data set where p-variables are observed.

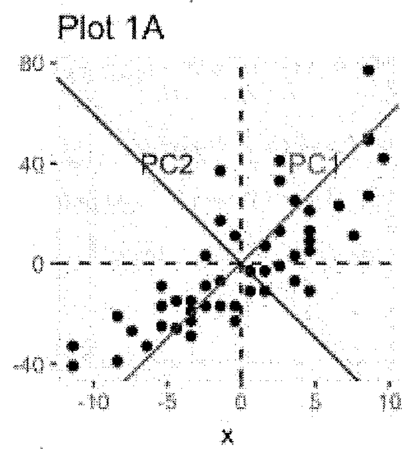
$x_i$ : $i^{th}$ observation of p-dimensional vector $x$.



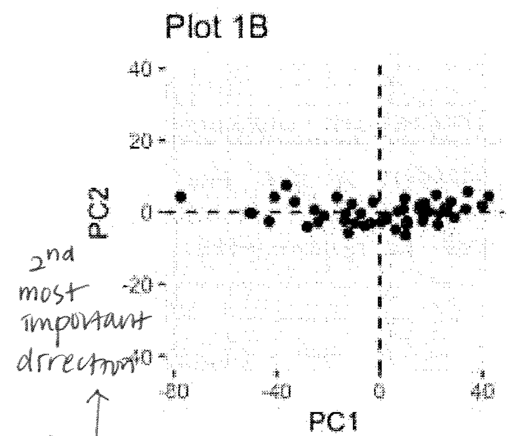

⇒ $PC_j = (\alpha_{j1} \times \text{predictor } 1) + (\alpha_{j2} \times \text{predictor } 2) + \cdots + (\alpha_{jp} \times \text{predictor } p)$

where $\alpha_{ji}$ = component weight

* PCA assumes that the directions w/ the largest variances are the 'most' important.




data are presented in the X-Y coordinate system.

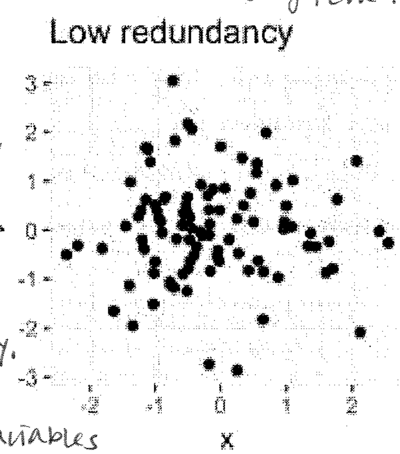


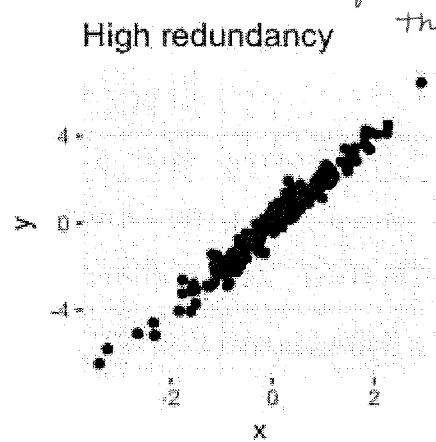

"orthogonal" → : 1st principal direction. (along which the samples show the largest variation)

PCA is useful when the variables within the dataset are highly correlated.

↓

"redundancy"




∴ Due to this redundancy, PCA can be used to reduce the original variables into a smaller number of new variables (= PCs) explaining most of the variance in the original variables.

→ In other words, PCA ① identify hidden pattern in a data set.

② reduce the dimensionality of the data by removing the noise and redundancy in the data

③ identify correlated variables

Regression technique based on principal component analysis

### 3.2.2 Assumption of PCR

- the directions in which the predictors show the most variation are the exact directions associated w/ the response variable.
  the directions in which $x_1, \cdots, x_p$ show the most variation are the direction that are associated w/ other Xs.

⇒ on one hand, this assumption is <u>NOT</u> guaranteed to hold 100% of the time, but, even though the assumption is not completely true it can be a good approximation and yield interesting results.

### 3.2.3 Advantages and Disadvantages

- Advantages
  1) Dimensionality reduction

     By using PCR, you can easily perform dimensionality reduction a high dimensional dataset, and then fit a linear regression model to a smaller set of variables, while at the same time keep most of the variability of the original predictors.

     ⇒ use only some of the principal components.
     ↓
     Help to reduce the model complexity

  2) Avoiding multicollinearity

     A significant benefit of PCR is that by using the PCs, if there is some degree of multicollinearity between the variables in data set, this procedures should be able to avoid this problem since performing PCA on the raw data produces linear combinations of the predictors that are uncorrelated.

  3) Overfitting mitigation

     If all the assumptions underlying PCR hold, then fitting a least squares model to the principal components will lead to better results than fitting a least squares models to the original data since the most of the variation and information related to the dependent variable is condensend in 3 the principal components and by estimating less coefficients you can reduce the risk of overfitting.

- Disadvantages

① PCR ≠ a feature selection method

∵ each of the calculated principal components is a linear combination of the original variables

∴ Hard to explain what is affecting what.

② The directions that best represent each predictor are obtained in an unsupervised way.

∵ The dependent variable is <u>NOT</u> used to identify each principal component direction.

∴ the directions found from PCR are not be the optimal directions to use when making predictions on the dependent variable.

3. Performing PCR in R

## Men's Decathlon Athletes In 2012

We will use the demo data set "decathlon2" from the *factoextra* package. The data used here describes athletes' performance during two sporting events (Desctar and OlympicG). It contains 27 individuals (athletes) described by 13 variables.

| Variable Name | Description |
|---|---|
| X100m | Points scored in 100 metres |
| Long.jump | Points scored in long jump |
| Shot.put | Points scored in shot put |
| High.jump | Points scored in high jump |
| X400m | Points scored in 400 metres |
| X110m.hurdle | Points scored in 110 metres hurdles |
| Discus | Points scored in discus throw |
| Pole.vault | Points scored in polt vault |
| Javeline | Points scored in javelin throw |
| X1500m | Points scored in 1500 metres |
| Rank | Final Ranking in 2012 |
| Points | Total scores |
| Competition | Name of athletic competition in 2012 |

R packages:
Several functions from different packages are available in the *R software* for computing PCA/PCR:

- *prcomp*() and *princomp*() [built-in R *stats* package],
- *PCA*() [*FactoMineR* package],
- *dudi.pca*() [*ade4* package],
- and *epPCA*() [*ExPosition* package]

Performing PCA

```
PCA(X, scale.unit = TRUE, ncp = 5, graph = TRUE)
```

Results

```
## **Results for the Principal Component Analysis (PCA)**
## The analysis was performed on 23 individuals, described by 10 variables
## *The results are available in the following objects:
##
##    name               description
## 1  "$eig"             "eigenvalues"
## 2  "$var"             "results for the variables"
## 3  "$var$coord"       "coord. for the variables"
## 4  "$var$cor"         "correlations variables - dimensions"
## 5  "$var$cos2"        "cos2 for the variables"
## 6  "$var$contrib"     "contributions of the variables"
## 7  "$ind"             "results for the individuals"
## 8  "$ind$coord"       "coord. for the individuals"
## 9  "$ind$cos2"        "cos2 for the individuals"
## 10 "$ind$contrib"     "contributions of the individuals"
## 11 "$call"            "summary statistics"
## 12 "$call$centre"     "mean of the variables"
## 13 "$call$ecart.type" "standard error of the variables"
## 14 "$call$row.w"      "weights for the individuals"
## 15 "$call$col.w"      "weights for the variables"
```

measurement to show
the amount of variance retained by each pc

1) Eigenvalues/Variances

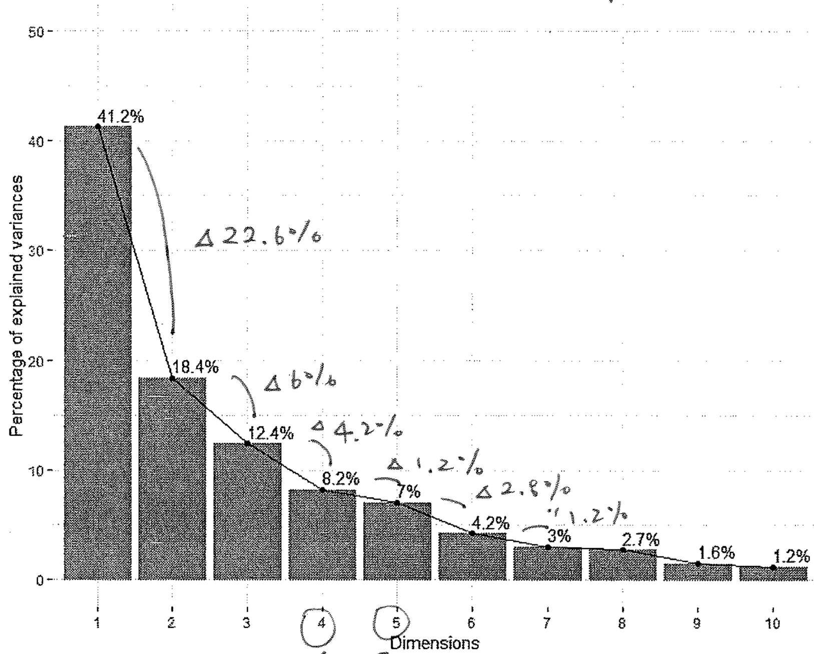
```
##         eigenvalue variance.percent cumulative.variance.percent
## Dim.1        4.124            41.24                        41.2
## Dim.2        1.839            18.39                        59.6
## Dim.3        1.239            12.39                        72.0
## Dim.4        0.819             8.19                        80.2
## Dim.5        0.702             7.02                        87.2
## Dim.6        0.423             4.23                        91.5
## Dim.7        0.303             3.03                        94.5
## Dim.8        0.274             2.74                        97.2
## Dim.9        0.155             1.55                        98.8
## Dim.10       0.122             1.22                       100.0
```

41% of variance explained by 1st PC.

72% of variance explained by 3 PCs (PC1, PC2, and PC3)

2) Scree plot

← Find elbow point (compare the difference in % variance).

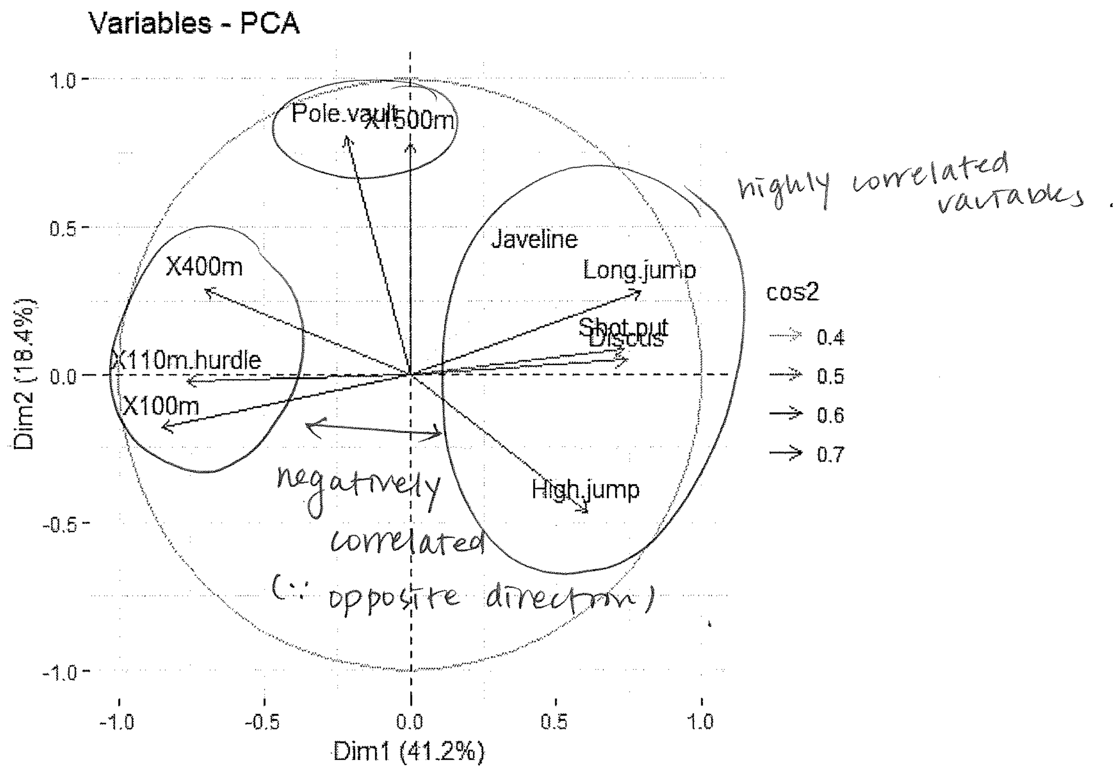



4 or 5 PCs.

3) Result of variables

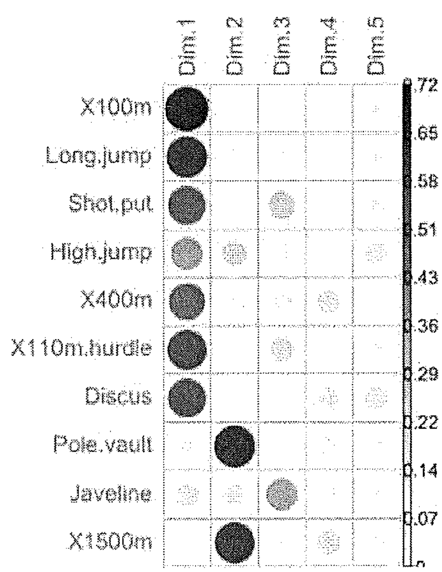
```
## Principal Component Analysis Results for variables
##  ===================================================
##   Name       Description
## 1 "$coord"   "Coordinates for the variables"
## 2 "$cor"     "Correlations between variables and dimensions"
## 3 "$cos2"    "Cos2 for the variables"
## 4 "$contrib" "contributions of the variables"
```

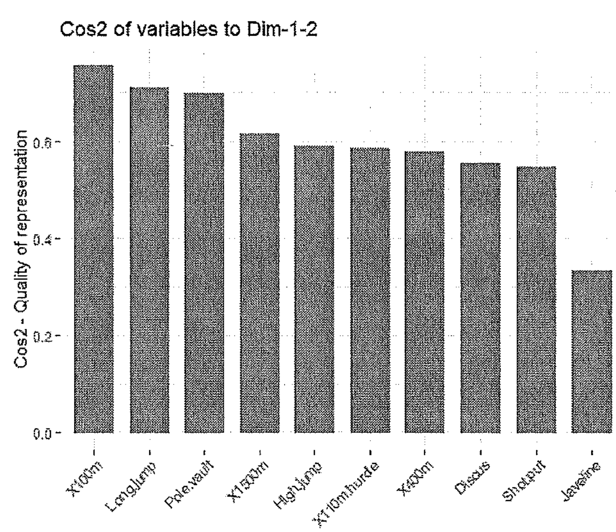
4) Correlation Circle

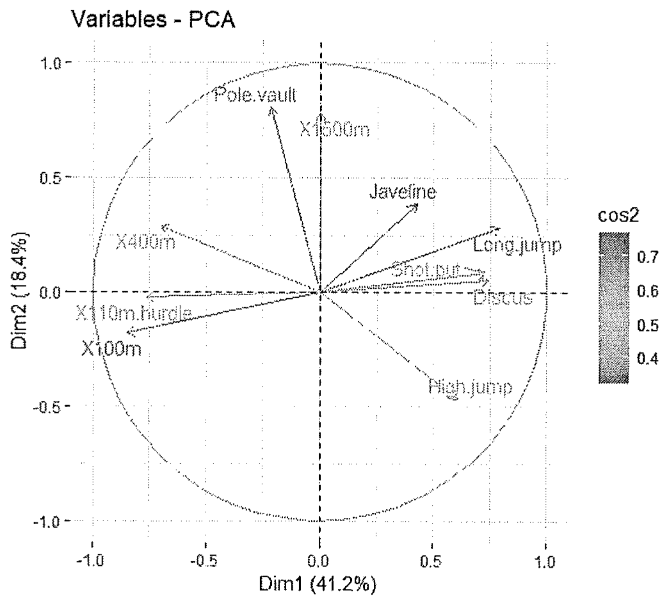



5) Quality of representation

```
##              Dim.1   Dim.2   Dim.3   Dim.4   Dim.5
## X100m        0.724 0.03218 0.0909 0.00113 0.0378
## Long.jump    0.631 0.07888 0.0363 0.01331 0.0544
## Shot.put     0.539 0.00729 0.2679 0.01650 0.0619
## High.jump    0.372 0.21642 0.1090 0.02089 0.1622
```



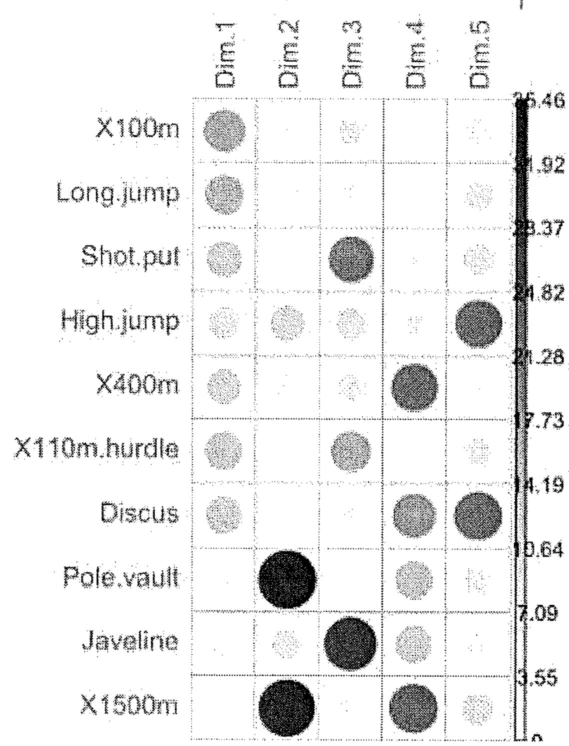


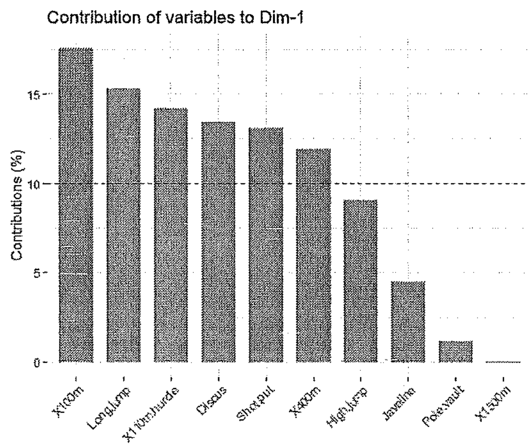

6) Contribution of variables to PCs

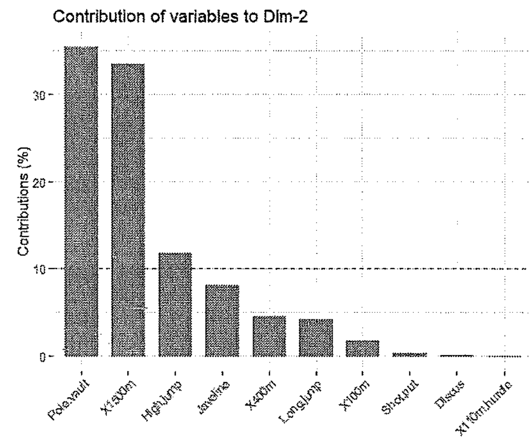
first 3 variables contribute on 1st PC.

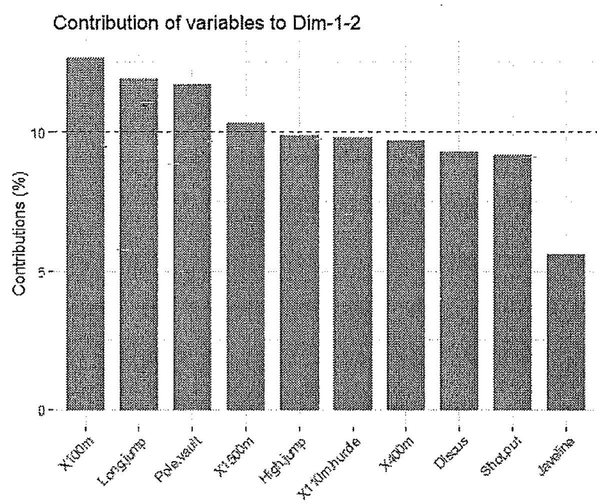
```
##            Dim.1  Dim.2 Dim.3 Dim.4 Dim.5
## X100m      17.54  1.751  7.34 0.138  5.39
## Long.jump  15.29  4.290  2.93 1.625  7.75
## Shot.put   13.06  0.397 21.62 2.014  8.82
## High.jump   9.02 11.772  8.79 2.550 23.12
```

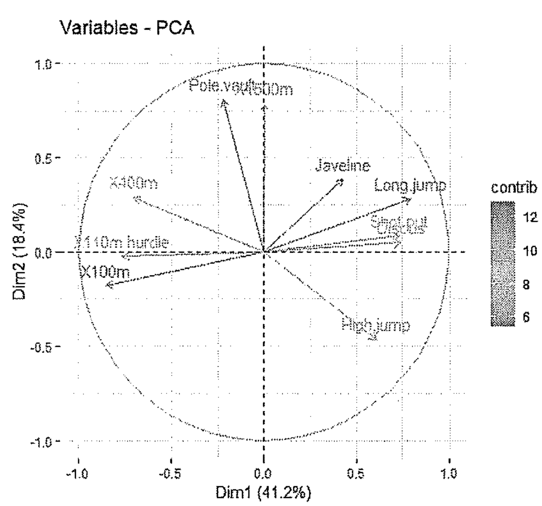
most of 2nd PC explained by High.Jump.

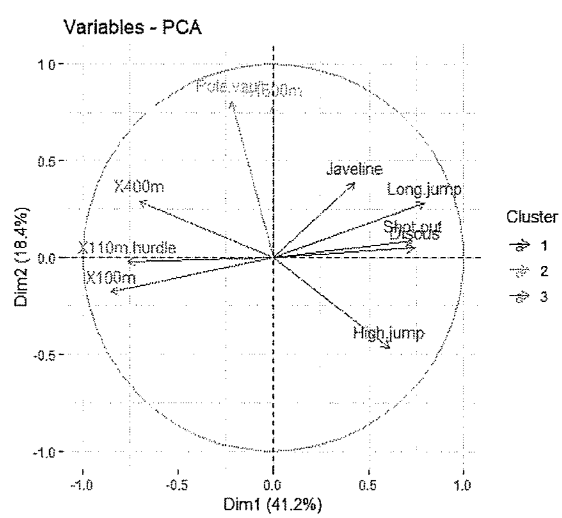



Contribution of variables to Dim-1
Contributions (%)
X100m
Long.jump
X110m.hurdle
Discus
Shot.put
X400m
High.jump
Javeline
Pole.vault
X1500m


Contribution of variables to Dim-2
Contributions (%)
Pole.vault
X1500m
High.jump
Javeline
X400m
Long.jump
X100m
Shot.put
Discus
X110m.hurdle


Contribution of variables to Dim-1-2
Contributions (%)
X100m
Long.jump
Pole.vault
X1500m
High.jump
X110m.hurdle
X400m
Discus
Shot.put
Javeline


Variables - PCA
Pole.vault
X1500m
Javeline
X400m
Long.jump
Shot.put
Discus
X110m.hurdle
X100m
High.jump
contrib
Dim2 (18.4%)
Dim1 (41.2%)


Variables - PCA
Pole.vault
X1500m
Javeline
X400m
Long.jump
Shot.put
Discus
X110m.hurdle
X100m
High.jump
Cluster
Dim2 (18.4%)
Dim1 (41.2%)

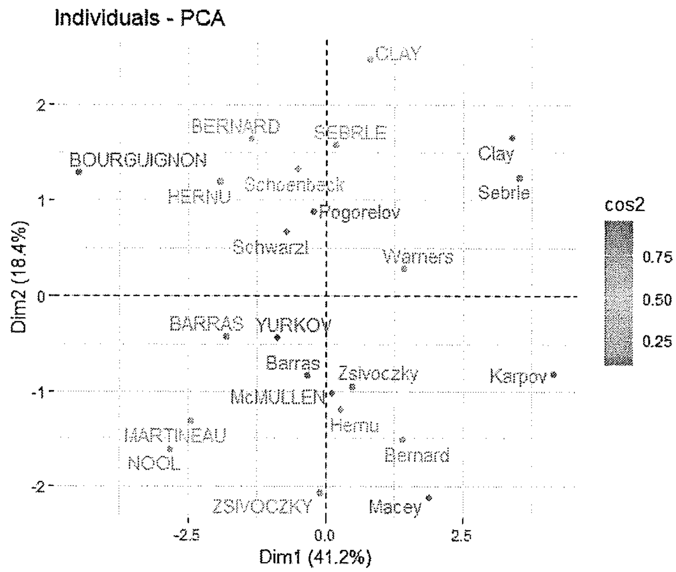
7) Dimension description

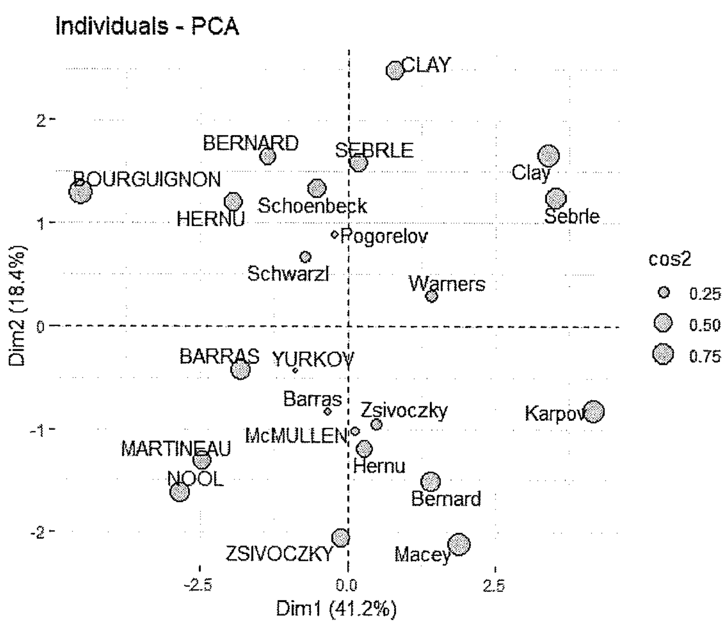
one group

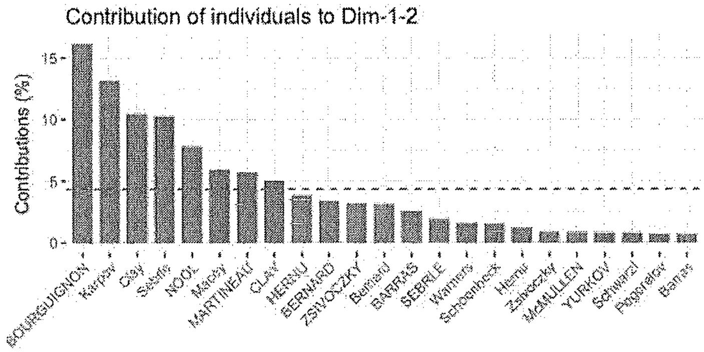
```
## $quanti
##               correlation  p.value
## Long.jump           0.794 6.06e-06
## Discus              0.743 4.84e-05
## Shot.put            0.734 6.72e-05
## High.jump           0.610 1.99e-03
## Javeline            0.428 4.15e-02
## X400m              -0.702 1.91e-04
## X110m.hurdle       -0.764 2.20e-05
## X100m              -0.851 2.73e-07
```

```
## $quanti
##              correlation  p.value
## Pole.vault         0.807 3.21e-06
## X1500m             0.784 9.38e-06
## High.jump         -0.465 2.53e-02
```

8) Graph individuals

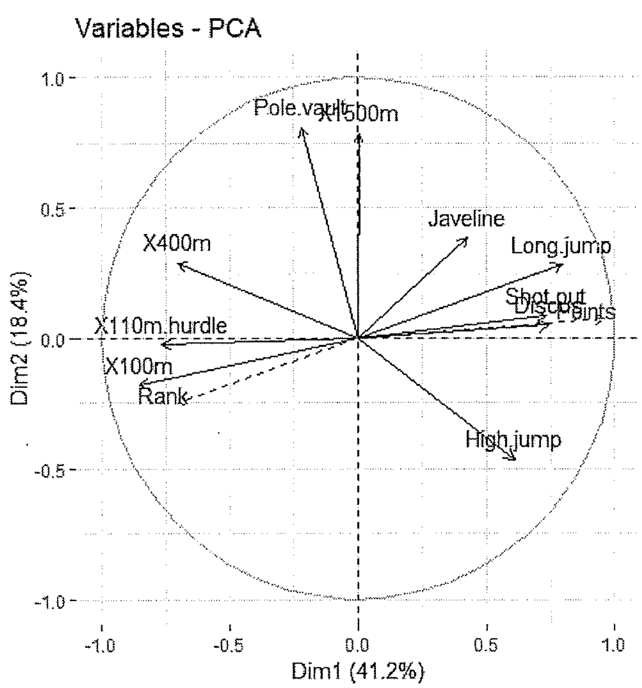


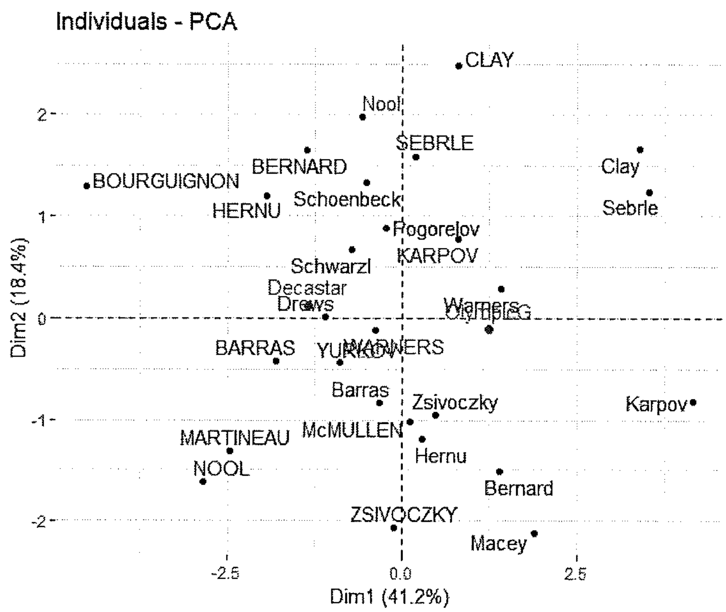



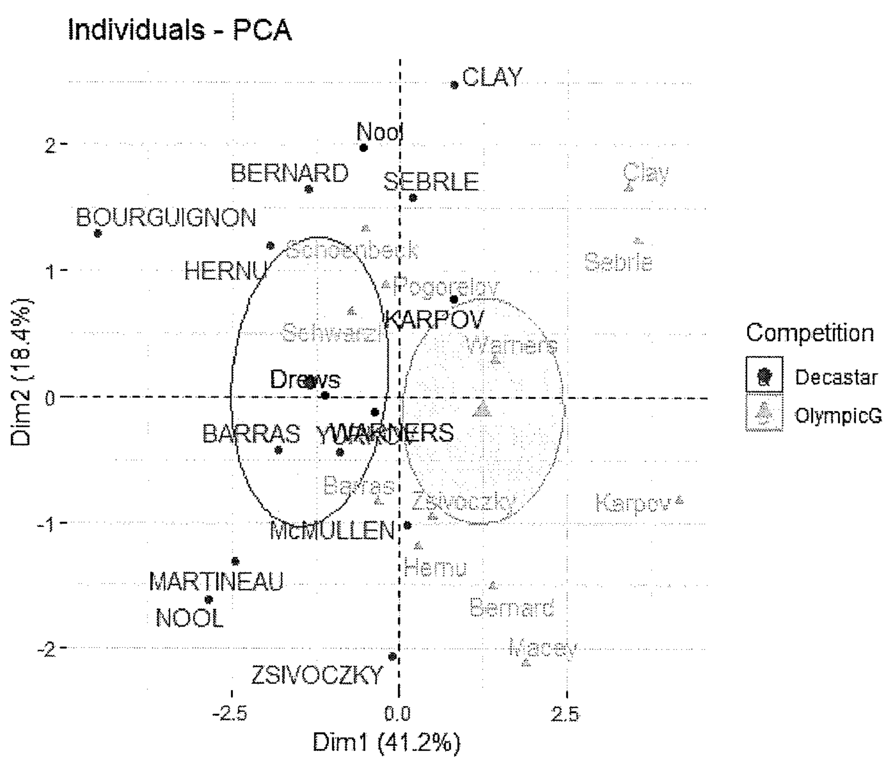



9) Specification in PCA

```
## $coord
##          Dim.1   Dim.2  Dim.3   Dim.4   Dim.5
## Rank    -0.701 -0.2452 -0.183  0.0558 -0.0738
## Points   0.964  0.0777  0.158 -0.1662 -0.0311
##
## $cor
##          Dim.1   Dim.2  Dim.3   Dim.4   Dim.5
## Rank    -0.701 -0.2452 -0.183  0.0558 -0.0738
## Points   0.964  0.0777  0.158 -0.1662 -0.0311
##
## $cos2
##         Dim.1   Dim.2  Dim.3   Dim.4   Dim.5
## Rank    0.492 0.06012 0.0336 0.00311 0.00545
## Points  0.929 0.00603 0.0250 0.02763 0.00097
```




Individuals - PCA
CLAY
Nool
SEBRLE
BERNARD
BOURGUIGNON
Clay
Schoenbeck
HERNU
Sebrle
Pogorelov
KARPOV
Schwarzl
Decastar
Drews
Warners
OlympicG
BARRAS
YURKOV
WARNERS
Barras
Zsivoczky
Karpov
MARTINEAU
McMULLEN
Hernu
NOOL
Bernard
ZSIVOCZKY
Macey
Dim2 (18.4%)
Dim1 (41.2%)
-2.5
0.0
2.5
2
1
0
-1
-2


Individuals - PCA
CLAY
Nool
BERNARD
SEBRLE
Clay
BOURGUIGNON
Schoenbeck
HERNU
Sebrle
Pogorelov
KARPOV
Schwarzl
Warners
Drews
BARRAS
YURKOV
WARNERS
Barras
Zsivoczky
Karpov
McMULLEN
Hernu
MARTINEAU
NOOL
Bernard
Macey
ZSIVOCZKY
Competition
Decastar
OlympicG
Dim2 (18.4%)
Dim1 (41.2%)
-2.5
0.0
2.5
2
1
0
-1
-2

# Unit 3. Dimensionality Reduction Techniques

## (2) Partial Least Squares (PLS) Regression

dimensional reduction technique w/ some similarities to principle component regression.

1. Introduction

* Unit 3 presents regression methods based on dimension reduction techniques, • ignore" outcome variable.
  which can be very useful when you have a large data set with multiple correlated predictor variables.

* All the dimension reduction methods work by first summarizing the original predictors into few new variables called principal components (PCs), which are then used as predictors to fit the linear regression model.

* Generally recommend to standardize each predictor to make them comparable.

□ PCR

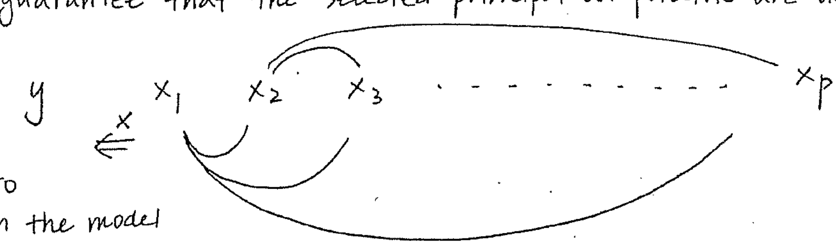
① first applies Principal Component Analysis on the data set to summarize the original predictor variables into few new variables.

② These PCs are then used to build the linear regression model.
  # of PCs to incorporate in the model is chosen by cross-validation (CV).

③ Suitable when the data set contains highly correlated predictors.

④ No guarantee that the selected principal components are associated w/ outcome.

⇓




∴ the PCs to incorporate in the model is NOT supervised by the outcomes.

□ Alternative PCR: PLS identifies linear combinations and directions that best represent the predictors.
Partial least squares regression is a form of regression that involves the development of components of the original variables in a supervised way.

what this means that: ⇒ the dependent variable is used to help create the new components form the original variables.

∴ help to explain both the independent and dependent variables in the model.

2. Performing Partial Least Squares (PLS) Regression

new variable

① The weights of the first linear combination ($Z_1$) is defined by the regression of $Y$ onto each of the $X$'s

⇒ Large weights are going to placed on the X-variables most related to $Y$ in the univariate case (most strongly related to the response).

② Regress each $X$ variable onto $Z_1$ and compute residuals

③ repeat step 1 using the residuals from ② in place of $X$.

④ iterate.

As always, the choice of where to stop (i.e., how many z-variables to use) should be done by comparing the out-of-sample predictive performance.

CVs of testing set.

(MSE or RMSE)

3. Advantages and Disadvantages

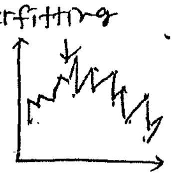
o Advantages

- Identifing new features in a supervised way.
- ~~Att~~ Attempting to find directions that help explain both response and predictors
- Reduce the bias. ⇒ more predictive accuracy.
- Reduce (control) the collinearity
  ⇒ much lower risk of chace correlation.

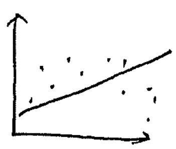
* Bias : the difference between the expected prediction of the model and the correct value which we are trying to predict.

▫ Bias - variance trade off.

*overfitting



· complex model ⇒ model variance ↑
w/ large # of Xs.
· perfectly predict the value ⇒ low bias



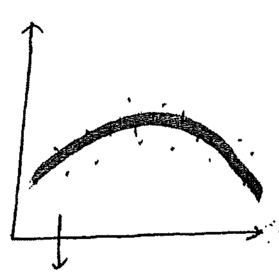
· simple model ↗ model variance ↓
w/ small # of Xs
· poorly predict values
⇒ High bias.

o Disadvantages

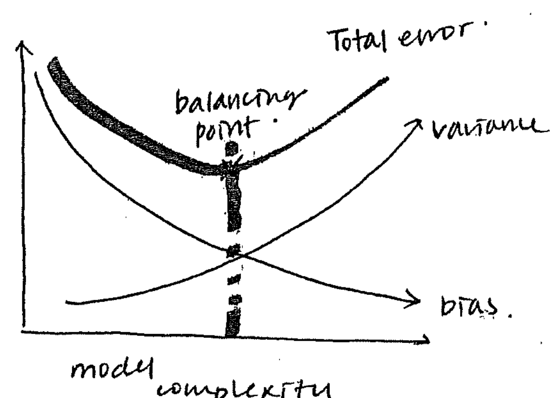
* variance : the variability of model prediction for a given data point.

- Potentially increase the variance.
- Higher risk of overlooking 'real' correlations.
- Sensitivity to the relative scaling of the predictor variables (in different scale)
  ⇒ cannot apply for complex mixture sample.

* Good balanced model



· relatively small # of predictors.
↓
controll variance
· follow the data pattern → reduce the bias.



4. Performing PLS in R: Using the Labor Supply Data

Income data

The Labor Supply Data is a cross-sectional data containing 753 observations with 18 predictors.

| Variables | Descriptions |
|---|---|
| work | participation in 1975 |
| hoursw | wife's hours of work in 1975 |
| child6 | number of children less than 6 years old in household |
| child618 | number of children between ages 6 and 18 in household |
| agew | wife's age |
| educw | wife's educational attainment, in years |
| hearnw | wife's average hourly earnings, in 1975 dollars |
| wagew | wife's wage reported at the time of the 1976 interview (not= 1975 estimated wage) |
| hoursh | husband's hours worked in 1975 |
| ageh | husband's age |
| educh | husband's educational attainment, in years |
| wageh | husbad's wage, in 1975 dollars |
| income | family income, in 1975 dollars |
| educwm | wife's mother's educational attainment, in years |
| educwf | wife's father's educational attainment, in years |
| unemprate | unemployment rate in county of residence, in percentage points |
| city | lives in large city (SMSA) |
| experience | actual years of wife's previous labor market experience |

**Source:** Mroz, T. (1987) "The sensitivity of an empirical model of married women's hours of work to economic and statistical assumptions", *Econometrica*, 55, 765-799.1976 Panel Study of Income Dynamics.

- 50/50 Split dataset

set.seed : In order to assure reduplization.
scale =T ; standardize the scale

1) Perform PLS

```
Data:    X dimension: 392 17
         Y dimension: 392 1
Fit method: kernelpls
Number of components considered: 17
```

∵ # of predictors =17.

After this point ΔCV is smaller.

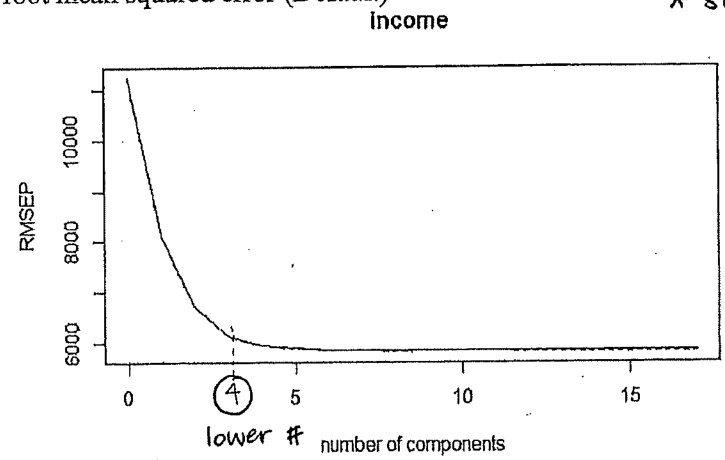
```
VALIDATION: RMSEP
Cross-validated using 10 random segments.
       (Intercept)  1 comps  2 comps  3 comps  4 comps  5 comps  6 comps  7 comps  8 comps  9 comps
CV           11218     8121     6701     6127     5952     5886     5857     5853     5849     5854
adjCV        11218     8114     6683     6108     5941     5872     5842     5837     5833     5837
       10 comps  11 comps  12 comps  13 comps  14 comps  15 comps  16 comps  17 comps
CV         5853      5853      5852      5852      5852      5852      5852      5852
adjCV      5836      5836      5835      5835      5835      5835      5835      5835
```

After 3 or 4 components. there is little improvement in PVE.

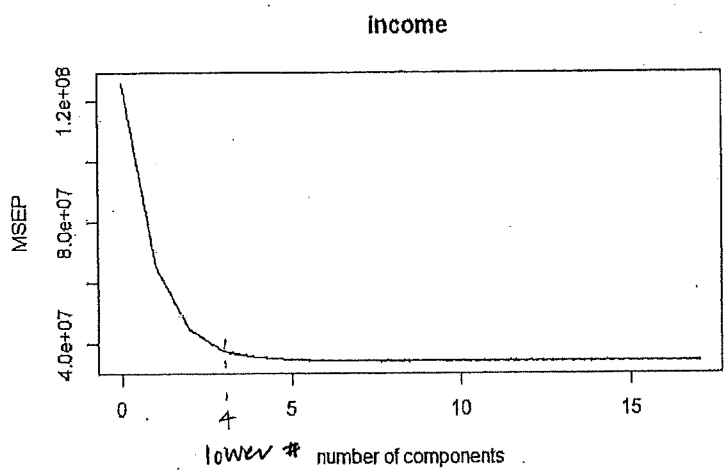
```
TRAINING: % variance explained
        1 comps  2 comps  3 comps  4 comps  5 comps  6 comps  7 comps  8 comps  9 comps  10 comps
X         17.04    26.64    37.18    49.16    59.63    64.63    69.13    72.82    76.06     78.59
income    49.26    66.63    72.75    74.16    74.87    75.25    75.44    75.49    75.51     75.51
        11 comps  12 comps  13 comps  14 comps  15 comps  16 comps  17 comps
X          81.79     85.52     89.55     92.14     94.88     97.62    100.00
income     75.52     75.52     75.52     75.52     75.52     75.52     75.52
```

2) Plot the root mean squared error (Default)

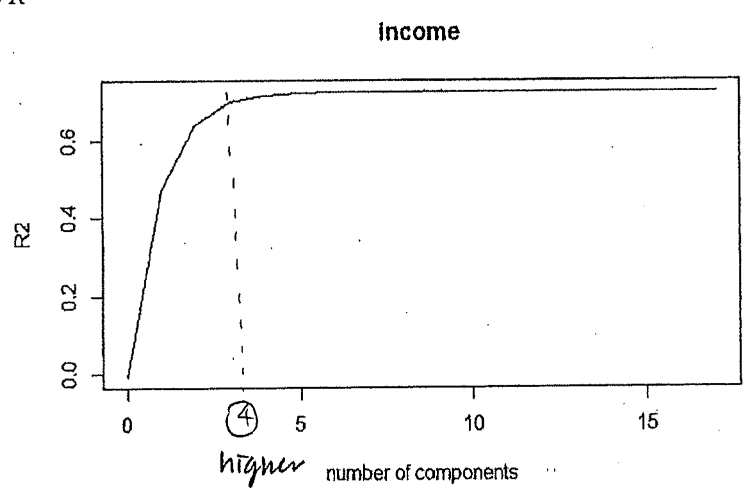
* still 5 or more components are considerable.
↓
bias-variance trade off.



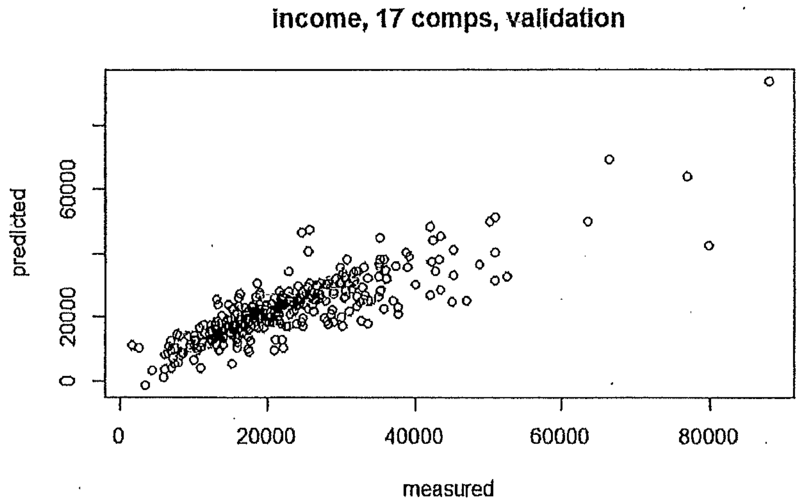

3) Plot the cross validation MSE



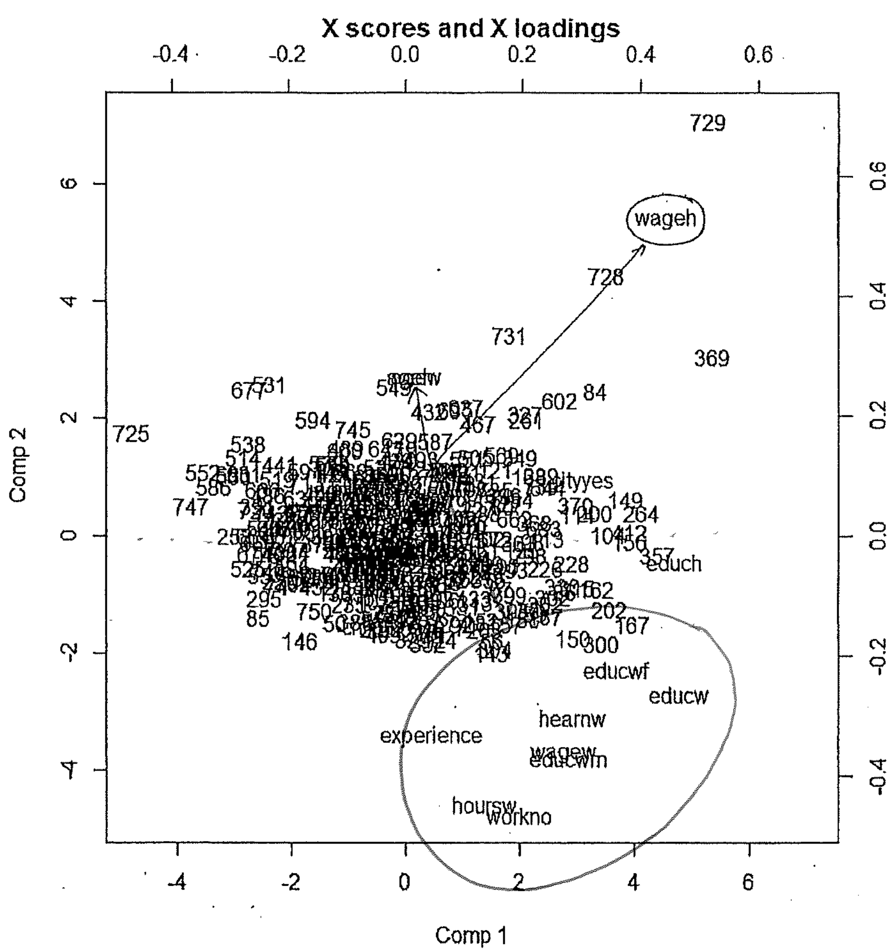

4) Plot the $R^2$



5) Plot the predicted vs measured values




5) Biplot



6) OLS

```
Call:
lm(formula = income ~ ., data = Mroz, subset = train)

Residuals:
   Min     1Q Median     3Q    Max
-20131  -2923  -1065   1670  36246

Coefficients:
              Estimate Std. Error t value Pr(>|t|)
(Intercept) -1.946e+04  3.224e+03  -6.036 3.81e-09 ***
workno      -4.823e+03  1.037e+03  -4.651 4.59e-06 ***
hoursw       4.255e+00  5.517e-01   7.712 1.14e-13 ***
child6      -6.313e+02  6.694e+02  -0.943 0.346258
child618     4.847e+02  2.362e+02   2.052 0.040841 *
agew         2.782e+02  8.124e+01   3.424 0.000686 ***
educw        1.268e+02  1.889e+02   0.671 0.502513
hearnw       6.401e+02  1.420e+02   4.507 8.79e-06 ***
wagew        1.945e+02  1.818e+02   1.070 0.285187
hoursh       6.030e+00  5.342e-01  11.288  < 2e-16 ***
ageh        -9.433e+01  7.720e+01  -1.222 0.222488
educh        1.784e+02  1.369e+02   1.303 0.193437
wageh        2.202e+03  8.714e+01  25.264  < 2e-16 ***
educwm      -4.394e+01  1.128e+02  -0.390 0.697024
educwf       1.392e+02  1.053e+02   1.322 0.186873
unemprate   -1.657e+02  9.780e+01  -1.694 0.091055 .
cityyes     -3.475e+02  6.686e+02  -0.520 0.603496
experience  -1.229e+02  4.490e+01  -2.737 0.006488 **
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 5668 on 374 degrees of freedom
Multiple R-squared:  0.7552,    Adjusted R-squared:  0.744
F-statistic: 67.85 on 17 and 374 DF,  p-value: < 2.2e-16
```

7) Comparing models

| | PLS | OLS1 | OLS2 | PCR |
|---|---|---|---|---|
| Bias | 63386682 | 59432814 | **57839715** | |
| Variance | **3** | 17 | 8 | |

MSE ⇐

\# of independent variables.

Full model w/ All predictors.

Best!

after dropping N.S effects. variance ↓ · (17→8).

but, model bias ↓.

# Unit 3 Dimensionality Reduction Techniques

## (3) Ridge and LASSO Regression

November 6, 2019

1. A large number of predictors

Potential multicollinearity problem ; predictors are highly correlated.

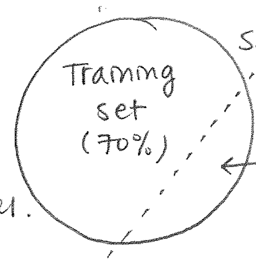
↓

① SE of $\beta_j$ is smaller than they should be.

↓

overfitting problem for training set of data; Perfectly fit the training data set.
But,
Poorly predict unknown values .

using to fit models for finding the best model.




Sample (n)

Testing (Validation) set (30%) .

↓

w/ many predictors

↓

model variances ↑.

↓

performing Good : $\boxed{\hat{y} = \hat{\beta}_0 + \hat{\beta}_1 x_1 + \cdots + \hat{\beta}_p x_p}$.

Best Model .

refit the Best Model using a test set .
① produce predicted model
② compare predicted values to actual values .

- High $R^2$
- Small MSE .
- Large F.
- Small BIC .

② "OLS" fails to find the unique solution to minimize $\sum_{i=1}^{n} (SSE_i)^2$ .

unbiased estimator .

Solution . ① variable selection

② Dimentional reduction ; PCR or PLS (Based on PCA)

③ Shrinkage models

- Ridge
- Lasso
- Elastic Net Regression .

## 2. Overview of the Shrinkage Methods

Under the assumptions, the coefficients estimated by "OLS" are unbiased and of all unbiased linear techniques also has the lowest variance ($\sigma^2$) ⇒ $\boxed{\text{MVUE (Minimum variance unbiased Estimator)}}$

※ Best

$MSE\ (=\sigma^2) = \sigma^2 + \text{Model Bias}^2 + \text{Model variance}$.

① we can effectively reduce "Total MSE" by controlling Model Bias & Model variance. ※ Bias-variance trade off.

② Introducing $\boxed{\text{small amount of Bias}}$.

⇒ Substantial drop in the variance.

※ Goal is to get a model w/ smaller MSE as a 2nd option.

∴ Shrinkage Model = Penalized regression model

= Regularization. regression model.

- imposes a penalty to the model for too many predictors.
- This results in shrinking the coefficients of the less contributive variables toward zero.

3. Ridge Regression

when # $p \uparrow$, $S.S.R = \sum_{i=1}^{n}(y_i - \hat{y}_i)^2$ under estimated.

↓

Sig. of $\beta_i$ over-estimated.

insignificant effect → significant.

∴ Estimated coefficients may become inflates.

↓

Need to control the magnitude of coefficient.

$t = \beta / SE(\beta)$, $t \uparrow$ if $\underline{\underline{\beta \uparrow}}$ or $SE(\beta) \downarrow$.

↓

achieve the smaller MSE (or SSR)

$$\underbrace{S.S.R_{norm2}}_{\text{2nd-order term}} = \sum_{i=1}^{n}(y_i - \hat{y}_i)^2 + \boxed{\lambda} \sum_{j=1}^{p} \beta_j^2$$

$i = 1, \cdots, n$ ; sample

$j = 1, \cdots, p$ ; predictors.

↓

controll the magnitude of coefficient.

- Choosing $\lambda$ (Tuning Parameter).
  - $\lambda$ controls the magnitute of coefficients.
  - " the degree of regularization.
  - As $\lambda \downarrow 0$, we obtain the least squares solution

    $\therefore \lambda = 0$, we are back to the regular regression model using OLS.

  - As $\lambda \uparrow \infty$, we obtain $\beta^{Ridge}_{\lambda=\infty} \approx 0$ ; close to "zero"

    Not equal to 0

In R, choosing the best $\lambda$.

1. Plot the components of all $\hat{\beta}_\lambda$ against $\lambda$
2. choose $\lambda$ for which the coefficients are not rapidly changing and have "sensible" sign.
3. Fit the model w/ best $\lambda$ using training set.
4. Refit the model using a testing set.

   $\downarrow$

   calculate C.V.

4. LASSO Regression (Least Absolute Shrinkage and Selecti Operator).

$$S.S.R._{norm_1} = \sum_{i=1}^{n} (y_i - \hat{y}_i)^2 + \lambda \sum_{j=1}^{p} |\beta_j|$$

Small $\lambda \Rightarrow$ "OLS"

Large $\lambda \Rightarrow$ less important $\beta_j = 0$.
$(\lambda \to \infty)$

Summary

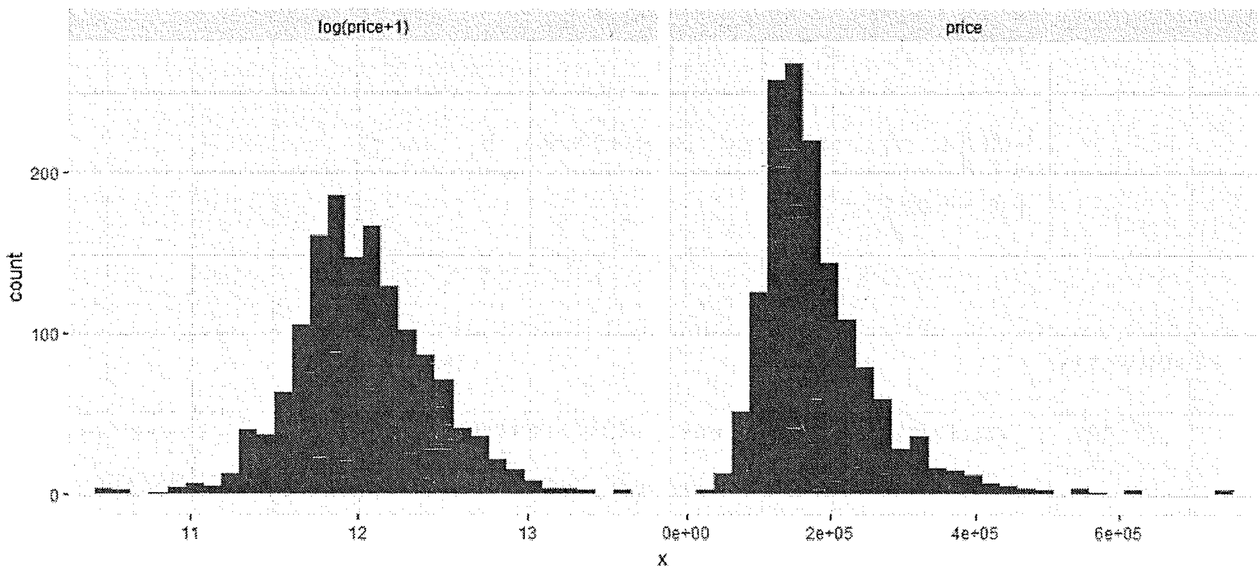
Ridge: variables w/ minor contribution have their coefficients close to zero. However, all the variables are incorporated in the model.

Lasso: the coefficients of some less contributive variables are forced to be exactly zero. Only the most sig. variables are kept in the final model.

Elastic net regression: combination of Ridge and Lasso.
Some of coefficient toward to zero
set some coefficients to exactly zero.

5. Performing Ridge and LASSO in R
   Data: Boston Housing Dataset

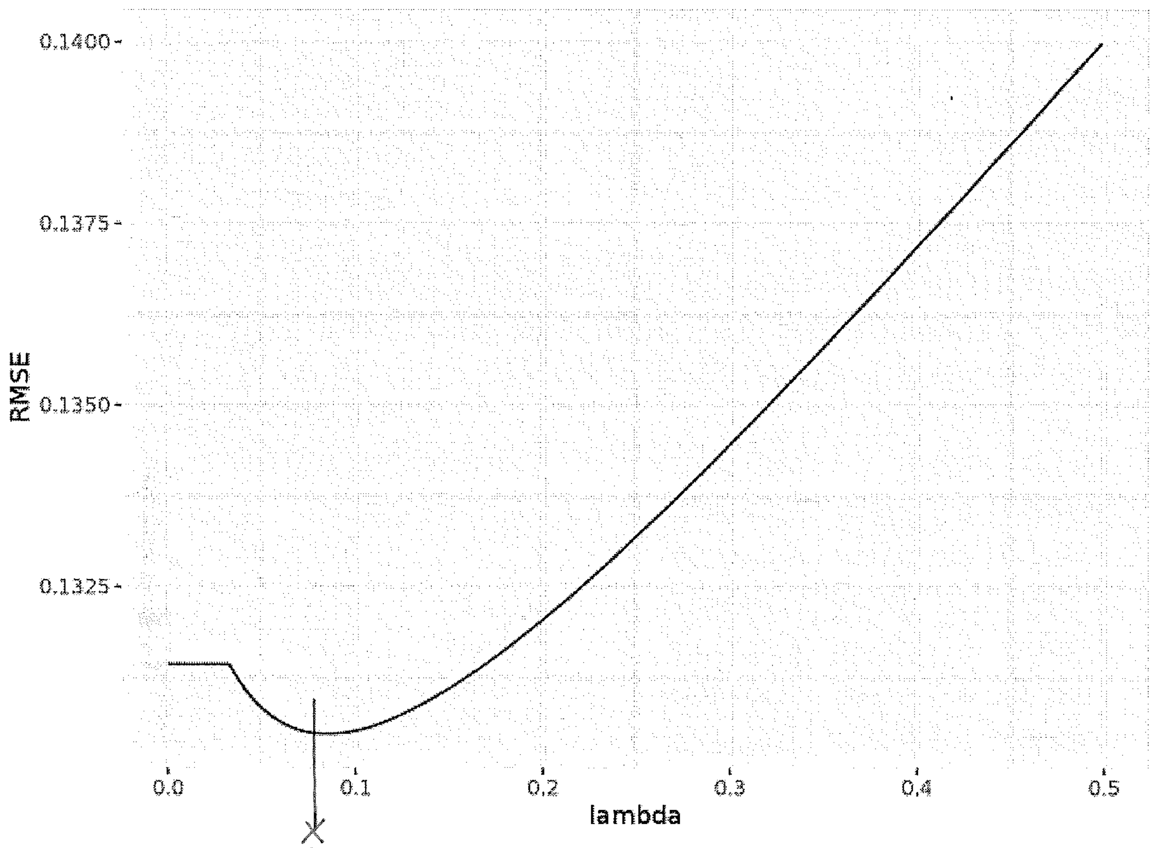
```
> dim(train)
[1] 1460   81
```




```
# test out Ridge regression model

            max  min   decending
lambdas <- seq(1,0,-0.001) <- λ     ∵ λ=0 ⇒ OLS.

# train model
set.seed(123)  # for reproducibility
model_ridge <- train(x=X_train,y=y,
                     method="glmnet", <- Generalized Linear model
                     metric="RMSE",  <- choose the best model based on MSE.
                     maximize=FALSE,
                     trControl=CARET.TRAIN.CTRL,
                     tuneGrid=expand.grid(alpha=0, # Ridge regression
                                          lambda=lambdas))
```




```
mean(model_ridge$resample$RMSE)
```

```
## [1] 0.1304773
```

```
# test out Lasso regression model

# train model
set.seed(123)  # for reproducibility
model_lasso <- train(x=X_train,y=y,
                     method="glmnet",
                     metric="RMSE",
                     maximize=FALSE,
                     trControl=CARET.TRAIN.CTRL,
                     tuneGrid=expand.grid(alpha=1,  # Lasso regression
                                          lambda=c(1,0.1,0.05,0.01,seq(0.009,0.001,-0.001),
                                                   0.00075,0.0005,0.0001)))
model_lasso
```

alternative way to generate λ.

```
## glmnet
##
## 1460 samples
##  288 predictor
##
## No pre-processing
## Resampling: Cross-Validated (5 fold, repeated 5 times)
## Summary of sample sizes: 1169, 1168, 1168, 1167, 1168, 1168, ...
## Resampling results across tuning parameters:
##
##   lambda   RMSE       Rsquared
##   0.00010  0.1334558  0.8890952
##   0.00050  0.1296718  0.8945323
##   0.00075  0.1284374  0.8963014
##   0.00100  0.1275134  0.8976080
##   0.00200  0.1251501  0.9009834
##   0.00300  0.1240240  0.9026627
##   0.00400  0.1238925  0.9029669
##   0.00500  0.1242215  0.9026495
##   0.00600  0.1247290  0.9021018
##   0.00700  0.1253763  0.9013804
##   0.00800  0.1262039  0.9004245
##   0.00900  0.1272133  0.8992018
##   0.01000  0.1283448  0.8977959
##   0.05000  0.1731301  0.8406120
##   0.10000  0.2154212  0.7968063
##   1.00000  0.3990550        NaN
##
## Tuning parameter 'alpha' was held constant at a value of 1
## RMSE was used to select the optimal model using  the smallest value.
## The final values used for the model were alpha = 1 and lambda = 0.004.
```

```
mean(model_lasso$resample$RMSE)
```

```
## [1] 0.1238925
```

CV

select / keep only "20" important predictors in the final model.