STAT 408
Applied Regression Analysis

Nan Miles Xi

Department of Mathematics and Statistics

Loyola University Chicago

Fall 2022

Shrinkage Method

Issues with High-Dimensional Data

- In many modern applications, the datasets contain a large number of predictors
- For example, the gene sequencing data contain the gene expression of all human protein encoding genes (up to 20K)
- Such datasets are called “high-dimensional” data, because p is very large
- There are issues to conduct linear regression on high-dimensional dataset:
  1. Many predictors are redundant
  2. Many predictors are often highly correlated – collinearity
  3. If $p > n$, then X is not full-rank, $(X^T X)$ is not invertible, least square estimation does not have unique solution

Ridge Regression

- Instead of dropping predictors, ridge regression limits the size of parameters so that the “shrinking effect” is embedded in the model fitting process

- The ridge regression build a linear model $Y = X\beta + \varepsilon$, but estimate β by minimizing

$$(y - X\beta)^T (y - X\beta) + \lambda \sum_j \beta_j^2$$

  where λ is a constant and $\lambda > 0$

- Ridge regression tries to make a balance between model fit and model complexity

- Larger λ will generate smaller model

Ridge Regression

- The term $\sum_j \beta_j^2$ is called L2 penalty and can be written in vector from $\beta^T\beta$
- The estimation of β in ridge regression is similar to regular least square estimation

$$\frac{\partial[(y-X\beta)^T(y-X\beta)+\lambda\beta^T\beta]}{\partial\beta}=0$$

$$\frac{\partial[(y-X\beta)^T(y-X\beta)+\lambda\beta^T\beta]}{\partial\beta}=-X^T2(y-X\beta)+2\lambda\beta=0$$

$$-2X^Ty+2X^TX\beta+2\lambda\beta=0$$

$$(X^TX+\lambda I)\beta=X^Ty$$

$$\hat{\beta}=(X^TX+\lambda I)^{-1}X^Ty$$

Ridge Regression

- The name "ridge regression" is coming from the term λI introduces a "ridge" into the $X^T X$

$$(X^T X)^{-1} X^T y$$

- λI in $\hat{\beta} = (X^T X + \lambda I)^{-1} X^T y$ brings two effects:
  1. It adds a "positive number" to "matrix denominator", so the size of parameter $\hat{\beta}$ shrinks
  2. With this "positive number", the "matrix denominator" cannot be zero anymore, the matrix $(X^T X + \lambda I)$ must be invertible and $\hat{\beta}$ must have a unique solution

Ridge Regression

- Minimizing $(y-X\beta)^T(y-X\beta)+\lambda\sum_j \beta_j^2$ is equivalent to we choose β to minimize

$$(y-X\beta)^T(y-X\beta) \quad \text{subject to} \quad \sum_{j=1}^{p}\beta_j^2 \le t^2$$

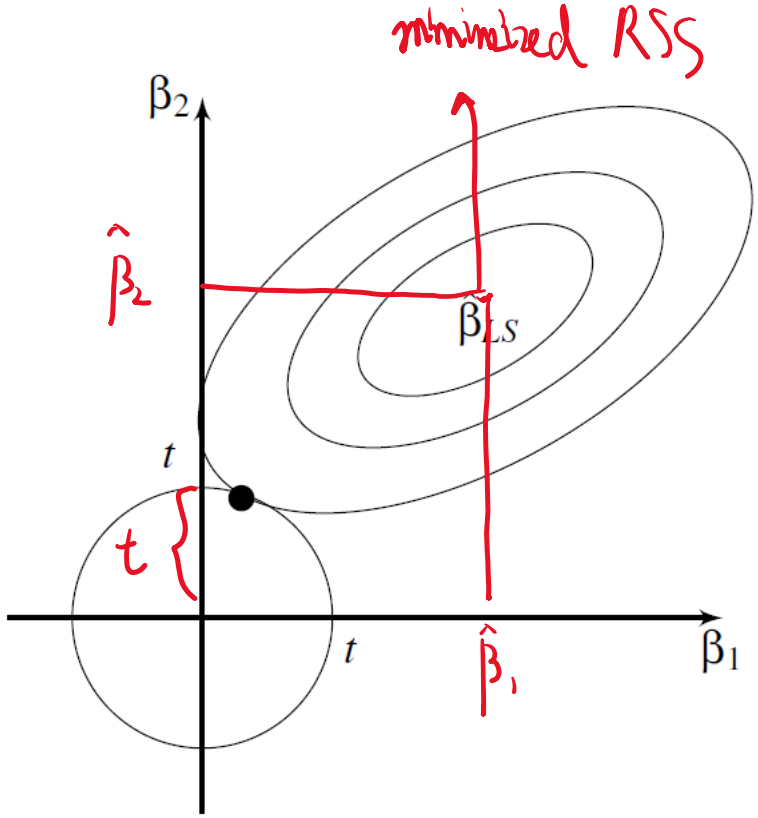
- In other words, ridge regression minimizes the RSS with the constraint that the total "size" of β is less than t^2, where t and λ is a one-to-one mapping
- With this form, we can use a figure to understand ridge regression

Ridge Regression

- This figure shows the optimization of ridge regression in two-dimensional case (β_1, β_2)
- The right circle is the contour of RSS (bottom at center)
- $\hat{\beta}_{LS}$ is the regular least square estimation that minimizes RSS
- The left circle defines the constraint in ridge regression

$$\sum_{j=1}^{p} \beta_j^2 \leq t^2$$

- Ridge regression finds a parameter estimation $\hat{\beta}$ that satisfies this constraint but also minimizes RSS as much as possible



Ridge Regression

- Let's use "meatspec" dataset to show the effect of penalty λ on $\hat{\beta}$

- The "meatspec" dataset contains a response variable of fat content in 215 samples of meat piece and 100 predictors

- Each predictor is the absorbances of near-infrared spectroscopy with one wavelength

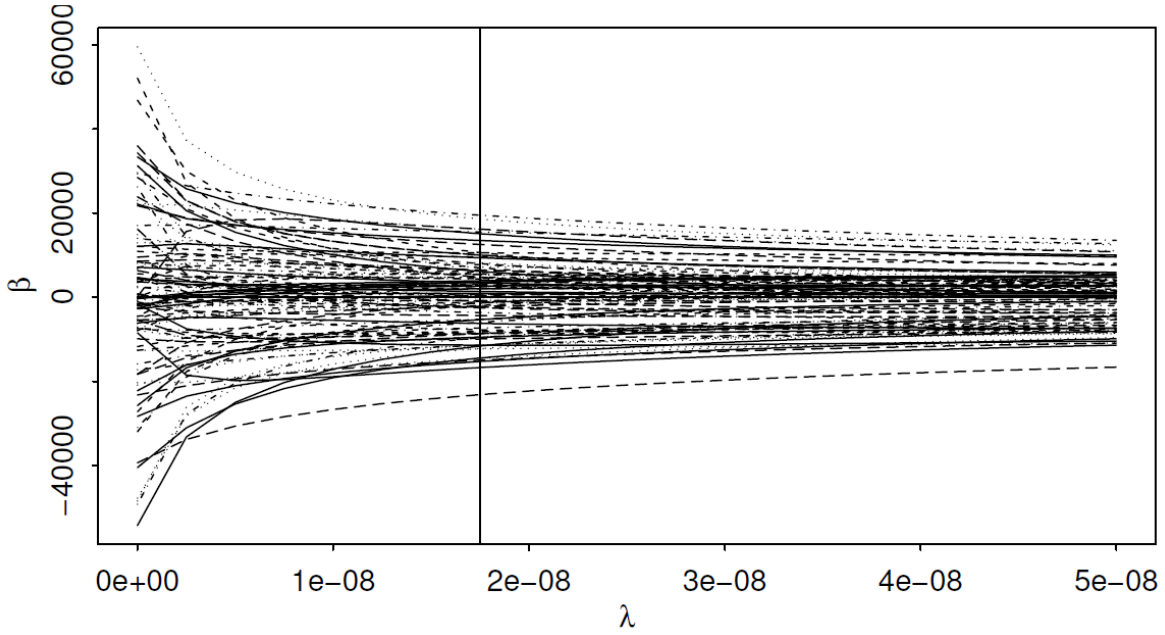
- We want to fit a linear model to predict the fat content of new samples using the 100 absorbances

Ridge Regression

- We set λ evenly increase from 0 to 5e-8 and fit 21 ridge regressions
- For each λ, we plot the value of 100 parameters to show the trend (ridge trace plot)

```
require(MASS)
rgmod <- lm.ridge(fat ~ ., meatspec, lambda = seq(0, 5e-8, len=21))
matplot(rgmod$lambda, coef(rgmod), type="l", xlab=expression(lambda)
        ,ylab=expression(hat(beta)),col=1)
```

Ridge Regression



1. $\lambda = 0$ mean no penalty – it is the regular least square
2. Larger λ generates "smaller" model
3. If $\lambda \to \infty$, then $\hat{\beta} \to 0$

Ridge Regression

- The λ in ridge regression is called “hyperparameter” and need to be selected by users

- The most common way to select λ is using cross-validation

- We start a group of alternatives of λ, then select the one that minimizes the test MSE in cross-validation

- See “coding 13.R” for implementations

Ridge Regression

- Ridge estimation $\hat{\beta}_{ridge} = (X^T X + \lambda I)^{-1} X^T y$ is different from regular least square estimation $(X^T X)^{-1} X^T y$

- Because least square estimation is unbiased, ridge estimation is biased $\mathrm{E}(\hat{\beta}_{ridge}) \neq \beta$

- Then why it provide more accuracy prediction than the unbiased $(X^T X)^{-1} X^T y$?

- The reason is biasness-variance tradeoff

Ridge Regression

- The accuracy of parameter estimation can be measured by the mean-squared-error (MSE)

$$E(\hat{\beta}-\beta)^2 = E[\hat{\beta}-E(\hat{\beta})+E(\hat{\beta})-\beta]^2$$

$$= E\left[\left(\hat{\beta}-E(\hat{\beta})\right)^2 + \left(E(\hat{\beta})-\beta\right)^2 + 2\left(\hat{\beta}-E(\hat{\beta})\right)\left(E(\hat{\beta})-\beta\right)\right]$$

$$= E\left(\hat{\beta}-E(\hat{\beta})\right)^2 + E\left(E(\hat{\beta})-\beta\right)^2$$

$$= Variance + Bias$$

$$V(X) = E(X-E(X))^2$$

Ridge Regression

- The accuracy of parameter estimation is determined by both biasness and variance of our estimation

- Although the least square estimation is unbiased (second term is zero), it may have larger variance compared with ridge regression

- If the decrease of variance overpasses the increase of biasness, the ridge regression will have a smaller final MSE and provides better prediction

- This is the trade-off that ridge regression makes — a reduction in variance at the price of an increase in bias

Lasso

- Lasso regression is to replace the L2 penalty in ridge regression by L1 penalty

$$(y - X\beta)^T (y - X\beta) + \lambda \sum_{i=1}^{p} |\beta_j|$$

- The penalty term now is sum of absolute values instead of sum of squares in ridge regression

- We can also write Lasso in another form

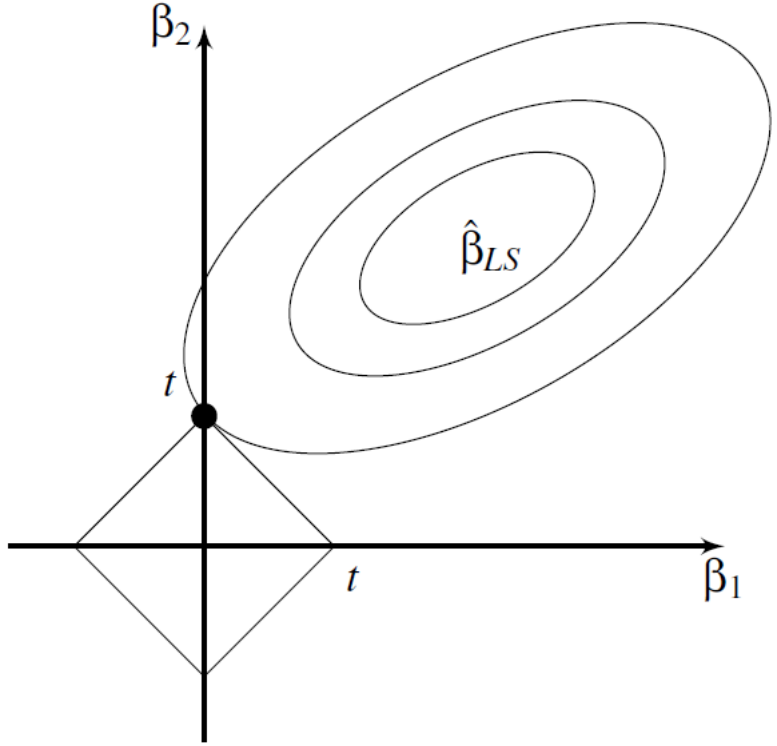
$$(y - X\beta)^T (y - X\beta) \quad \text{subject to} \quad \sum_{j=1}^{p} |\beta_j| \leq t$$

- Larger λ or smaller t give more penalty

Lasso

- The key difference between Lasso and ridge regression is that Lasso will force part of parameters to zero, which can be illustrated in the following figure

- The right circle is the contour of RSS in least square
- $\hat{\beta}_{LS}$ is the regular least square estimation that minimized RSS
- The left square defines the constraint in Lasso $\sum_{j=1}^{p} |\beta_j| \leq t$
- The solution of Lass will just cut the square at its corners, so that parameter would be zero



Lasso

- There is no closed-form solution for Lasso due to the absolute term
- Lasso will introduce sparsity into our model, which is usually a good property
- Now we perform Lasso on “state” dataset

	Population	Income	Illiteracy	Life.Exp	Murder	HS.Grad	Frost	Area
UT	1203	4022	0.6	72.90	4.5	67.3	137	82096
AK	365	6315	1.5	69.31	11.3	66.7	152	566432
NV	590	5149	0.5	69.03	11.5	65.2	188	109889
CO	2541	4884	0.7	72.06	6.8	63.9	166	103766
WA	3559	4864	0.6	71.72	4.3	63.5	32	66570
WY	376	4566	0.6	70.29	6.9	62.9	173	97203

Lasso

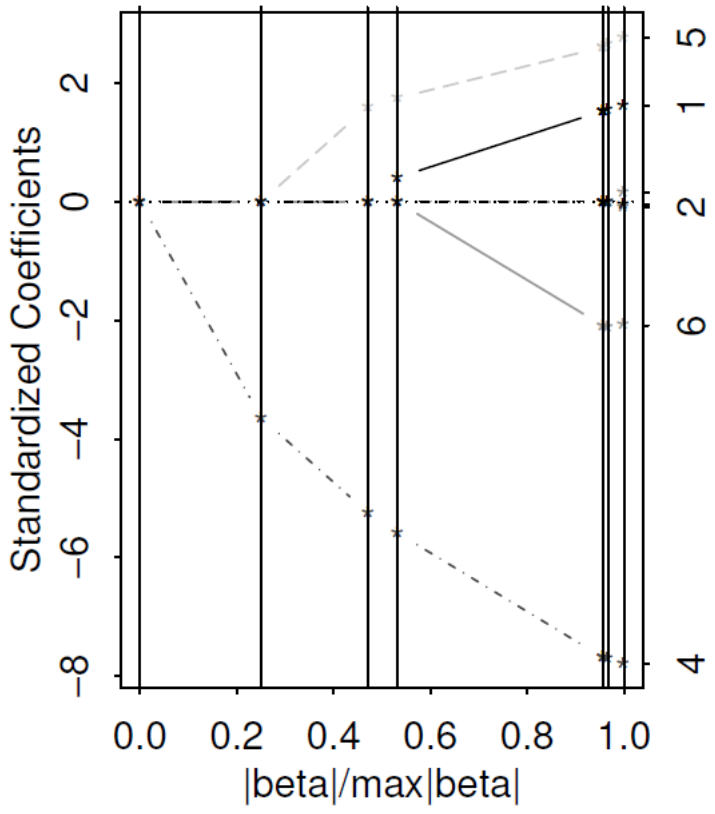
- The lars function requires the matrix of predictors as its first argument, and the vector of response as its second argument

```
require(lars)
state <- read.csv('state.csv')
lmod <- lars(as.matrix(state[,-4]),state$Life)
plot(lmod)
```

- We can plot the “trace” of Lasso parameters across difference degree of penalty

Lasso

- The x-axis is normalized constraint: $t/\max(\hat{\beta}_j)$; The y-axis is the estimated parameter
- For the smallest t (largest penalty), only predictor 4, the murder rate, is active
- As t increases, a second predictor, the high school graduation rate enters. The population and days of frost enter later
- The remaining three variables do not enter the model until t is very close to the least squares solution
- As t increases, we see the size of the estimated coefficients also increase.



Lasso

- We can use Lasso for prediction on fat dataset, by using cross-validation to select best degree of penalty t

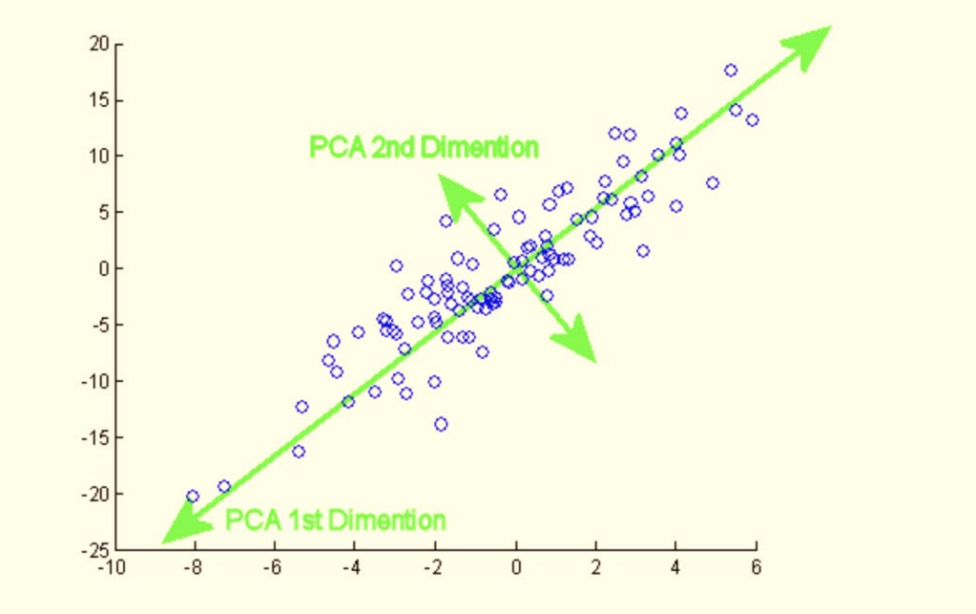
- See "coding 13.R" for implementations

Principle Component Analysis

Principle Component Analysis

- Principal components analysis (PCA) is a popular method for finding low-dimensional linear structure in high-dimensional data

- PCA is to project the original predictors to new directions such that the predictors after projection are orthogonal with maximum variance



An example of PCA with two-dimensions – the data are projected to two new orthogonal directions with maximum variance

Principle Component Analysis

- The general step of PCA is as following:
  1. Normalize the predictor matrix X by subtracting the mean and divide by standard deviation for each column
  2. Find a vector u_1 such that $\mathrm{Var}(Xu_1)$ is maximized subjected to $u_1^T u_1 = 1$
  3. Find another vector u_2 such that $\mathrm{Var}(Xu_2)$ is maximized (but less than $\mathrm{Var}(Xu_1)$) subjected to $u_1^T u_2 = 0$ and $u_2^T u_2 = 1$
  4. Keep finding directions of greatest variation orthogonal to those directions we have already found
  5. If the number of parameters is p, then we can find p different vectors u_i

- The transformed predictor $z_i = Xu_i$ is called principal component, $i = 1, 2, \dots, p$
- In matrix form, $Z = XU$ and U is called rotation matrix

Principle Component Analysis

- The general step of PCA is as following:
  1. Normalize the predictor matrix X by subtracting the mean and divide by standard deviation for each column
  2. Find a vector u_1 such that $\mathrm{Var}(Xu_1)$ is maximized subjected to $u_1^T u_1 = 1$
  3. Find another vector u_2 such that $\mathrm{Var}(Xu_2)$ is maximized (but less than $\mathrm{Var}(Xu_1)$) subjected to $u_1^T u_2 = 0$ and $u_2^T u_2 = 1$
  4. Keep finding directions of greatest variation orthogonal to those directions we have already found
  5. If the number of parameters is p, then we can find p different vectors u_i

- The transformed predictor $z_i = Xu_i$ is called principal component, $i = 1, 2, \dots, p$
- In matrix form, $Z = XU$ and U is called rotation matrix

Principle Component Analysis

- PCA essentially finds p orthogonal directions $u_1, u_2, \dots, u_p$ (with unit length), and projects the original X to those directions

- After PCA transformation, the new predictors z_i are orthogonal, so there is no collinearity anymore

- It turns out that we can use the eigenvectors of covariance matrix $Var(X)$ as $u_1, u_2, \dots, u_p$

$$Var(X)u_i = \lambda_i u_i$$

- In practice, we may only use the first several principal components if they explain the majority of variance in X

Principle Component Analysis

- PCA can be applied to both interpretation and prediction. Let's first see an example in interpretation

- Recall that "fat" dataset records the body fat (brozek) and 10 body size measurements for 252 men

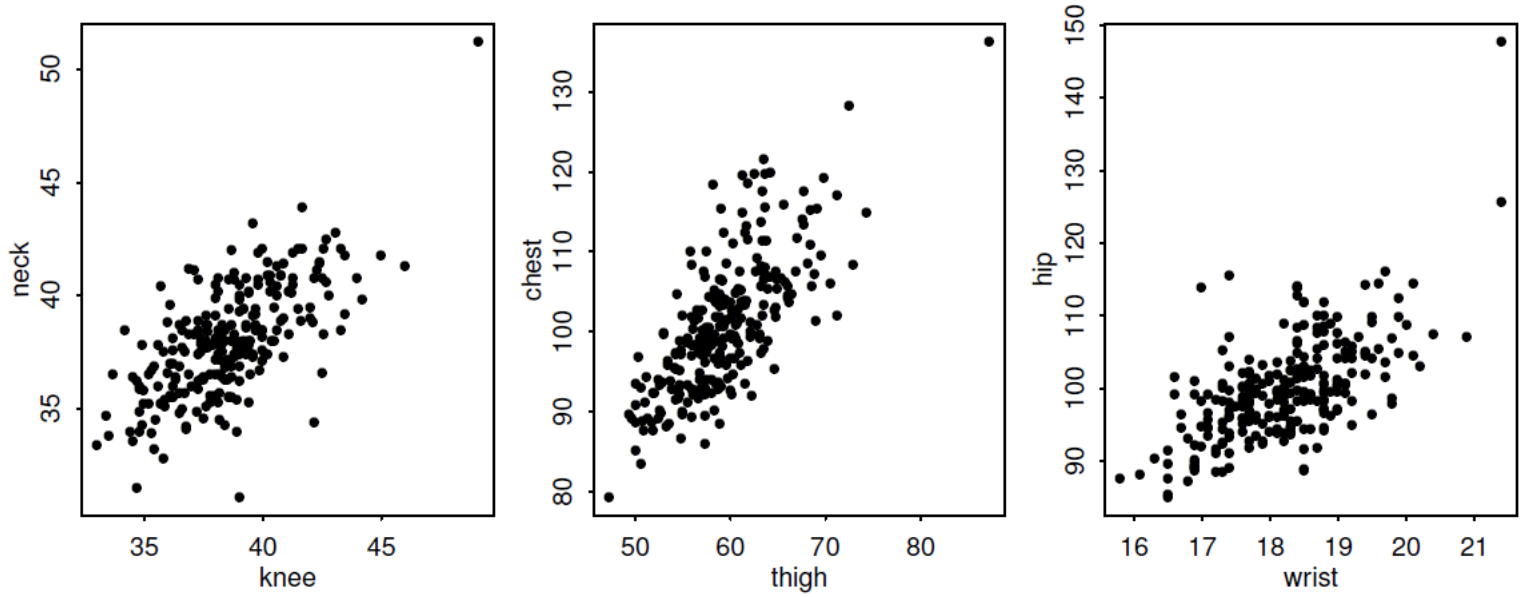
brozek	neck	chest	abdom	hip	thigh	knee	ankle	biceps	forearm	wrist
12.6	36.2	93.1	85.2	94.5	59.0	37.3	21.9	32.0	27.4	17.1
6.9	38.5	93.6	83.0	98.7	58.7	37.3	23.4	30.5	28.9	18.2
24.6	34.0	95.8	87.9	99.2	59.6	38.9	24.0	28.8	25.2	16.6
10.9	37.4	101.8	86.4	101.2	60.1	37.3	22.8	32.4	29.4	18.2
27.8	34.4	97.3	100.0	101.9	63.2	42.2	24.0	32.2	27.7	17.7
20.6	39.0	104.5	94.4	107.8	66.0	42.0	25.6	35.7	30.6	18.8

- We want to regress body fat on other variables to check their relation

Principle Component Analysis

- Many body part measurements are highly correlated

```
plot(neck ~ knee, fat)
plot(chest ~ thigh, fat)
plot(hip ~ wrist, fat)
```



- We use PCA to transform predictor X to a new space such that they are orthogonal after transformation

Principle Component Analysis

```
> cfat <- fat[,9:18]
> prfatc <- prcomp(cfat, scale=TRUE)
> summary(prfatc)
Importance of components:
                          PC1     PC2     PC3     PC4     PC5     PC6
Standard deviation     2.6498 0.85301 0.81909 0.70114 0.54708 0.52831
Proportion of Variance 0.7021 0.07276 0.06709 0.04916 0.02993 0.02791
Cumulative Proportion  0.7021 0.77490 0.84199 0.89115 0.92108 0.94899
                           PC7     PC8     PC9    PC10
Standard deviation     0.45196 0.40539 0.27827 0.2530
Proportion of Variance 0.02043 0.01643 0.00774 0.0064
Cumulative Proportion  0.96942 0.98586 0.99360 1.0000
```

- We need to set “scale = TRUE” to normalize each column of X, because PCA is sensitive to scale
- The output shows the variance of each principal component
- The proportion of variability explained by the first component is 70.2%, significantly larger than others

Principle Component Analysis

- Let's check the dimension of principal components and rotation matrix

$$z_i = Xu_i$$

where X: $n \times p$; u_i: $p \times 1$; z_i: $n \times 1$

$$Z = XU$$

where X: $n \times p$; U: $p \times p$; z_i: $n \times p$

- After PCA transformation, the new "data" Z still has p predictors and n observations, but in a rotated space defined by U

```
> cfat <- fat[,9:18]
> prfat <- prcomp(cfat)
> dim(prfat$rot)
[1] 10 10
> dim(prfat$x)
[1] 252  10
```

- The original X is 252×10
- The U matrix is $p \times p$ = 10×10
- The Z matrix is still 252×10

Principle Component Analysis

- By $z_i = Xu_i$, we can see that the principal component z_i (new predictor) is a linear combination of original p predictors

```
> round(prfatc$rot[,1],2)
   neck    chest    abdom      hip    thigh     knee    ankle   biceps
   0.33     0.34     0.33     0.35     0.33     0.33     0.25     0.32
forearm    wrist
   0.27     0.30
```

- We find the first principal component z_1 has very similar coefficients for all the p original predictors
- Therefore, we can interpret z_1 as the "overall size" of man's body since it is similarly proportional to all body circumferences

Principle Component Analysis

```
> round(prfatc$rot[,2],2)
   neck    chest    abdom      hip    thigh     knee    ankle   biceps
   0.00    -0.27    -0.40    -0.25    -0.19     0.02     0.62     0.02
forearm    wrist
   0.36     0.38
```

- The second principal component shows a contrast between the body center measures(chest, abdomen, hip, thigh) against the outer parts (forearm, wrist, ankle)
- Therefore, we can interpret z_2 as the relative measure of where the body is carrying its weight
- The other PC has even less variance and hard to interpret

Principle Components Regression

- Since the new predictor z_i is orthogonal to each other and is a linear combination of original predictor X, we can regress Y on Z to obtain better model fitting and interpretation
- This is called principal components regression (PCR); let's compare the regular linear regression with PCR

```
> lmoda <- lm(fat$brozek ~ ., data=cfat)
> sumary(lmoda)
             Estimate Std. Error t value Pr(>|t|)
(Intercept)  7.22875    6.21431    1.16  0.24588
neck        -0.58195    0.20858   -2.79  0.00569
chest       -0.09085    0.08543   -1.06  0.28866
abdom        0.96023    0.07158   13.41  < 2e-16
hip         -0.39135    0.11269   -3.47  0.00061
thigh        0.13371    0.12492    1.07  0.28554
knee        -0.09406    0.21239   -0.44  0.65828
ankle        0.00422    0.20318    0.02  0.98344
biceps       0.11120    0.15912    0.70  0.48533
forearm      0.34454    0.18551    1.86  0.06450
wrist       -1.35347    0.47141   -2.87  0.00445
```

- The high degree of collinearity makes the model interpretation difficult
- Many similar predictors have opposite signs (abdomen vs. hip)
- The standard error inflates

Principle Components Regression

- We conduct a PCR with first two PCs as predictors

```
> lmodpcr <- lm(fat$brozek ~ prfatc$x[,1:2])
> sumary(lmodpcr)
                    Estimate Std. Error t value Pr(>|t|)
(Intercept)           18.938      0.329   57.54   <2e-16
prfatc$x[, 1:2]PC1     1.842      0.124   14.80   <2e-16
prfatc$x[, 1:2]PC2    -3.551      0.387   -9.18   <2e-16
```

- The first two PCs are orthogonal, so there is no collinearity, the model interpretation is much easier
- The first PC can be viewed as a measure of overall size, which is associated with higher body fat
- The second PC shows a negative association, meaning that men who carry more of outer body parts tend to be leaner

PCR for Prediction

- We can make better predictions using linear model with a small number of principal components in Z than a much larger number of predictors in X

- The key of using PCR for regression is to select appropriate number of PCs

- We can use cross-validation to select the best number PCs for prediction