CHAPTER 9

Multicollinearity

Introduction

- **Multicollinearity** is a problem that plagues many regression models
- **Multicollinearity** impacts the estimates of the individual regression coefficients
- Uses of regression:
  - Identifying the relative effects of the regressor variables
  - Prediction and/or estimation
  - Selection of an appropriate set of variables for the model

Introduction

- If all regressors are **orthogonal**, then multicollinearity does not exist
- This is a rare situation in regression analysis
- Often there are *near-linear dependencies* among the regressors such that

$$\sum_{j=1}^{p} t_j \mathbf{X}_j \approx \mathbf{0}$$

- If this sum holds exactly for a subset of regressors, then $(\mathbf{X'X})^{-1}$ does not exist.

Sources of Multicollinearity

- Four primary sources
  - The data collection method employed
  - Constraints on the model or in the population
  - Model specification
  - An overdefined model

Sources of Multicollinearity

- **Data collection method employed**
  - Occurs when only a subsample of the entire sample space has been selected.
  - Example: Soft drink delivery: number of cases and distance tend to be correlated
  - Have data where only a small number of cases are paired with short distances, large number of cases paired with longer distances
  - May be able to reduce this multicollinearity through the sampling technique used

Sources of Multicollinearity

- **Constraints on the model or in the population**
  - Electricity consumption: two variables x_1 – family income and x_2 – house size
  - Physical constraints are present, multicollinearity will exist *regardless* of collection method

Sources of Multicollinearity

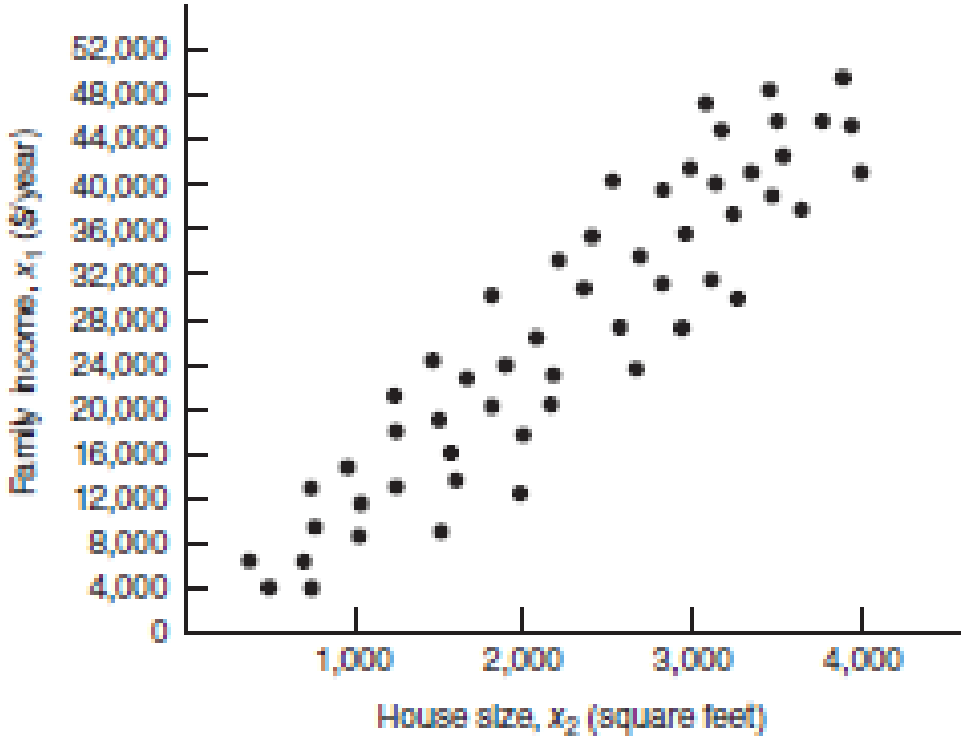


Figure 9.1 Levels of family income and house size for a study on residential electricity consumption.

Sources of Multicollinearity

- **Model Specification**
  - Polynomial terms can cause ill-conditioning in the X'X matrix
  - This is especially true if range on a regressor variable, *x*, is small

Sources of Multicollinearity

- **Overdefined model**
  - More regressor variables than observations
  - The best way to counter this is to remove regressor variables
  - Recommendations:
    - Redefine the model using smaller set of regressors
    - Do preliminary studies using subsets of regressors
    - Use principal components type regressor methods to remove regressors

Effects of Multicollinearity

- Strong multicollinearity can result in large variances and covariances for the least squares estimates of the coefficients

- Recall **C** = (**X'X**)$^{-1}$ and $C_{jj} = \dfrac{1}{1 - R_j^2}$

- Strong multicollinearity between x_j and any other regressor variable will cause R_j^2 to be large, and thus C_{jj} to be large.
- Thus the variance of the least squares estimate of the coefficient will be very large

Effects of Multicollinearity

- Strong multicollinearity can also produce least-squares estimates of the coefficients that are too large in absolute value

Effects of Multicollinearity

- Informal Diagnostics
  - Large changes in the estimated regression coefficients when a predictor variable is added or deleted
  - Nonsignificant results in individual tests of regression coefficients for important predictors
  - Regression coefficients with a sign that is opposite of that expected
  - Large coefficients of simple correlation between pairs of predictor variables
  - Wide confidence intervals for the regression coefficients for important predictors

Multicollinearity Diagnostics

- Ideal characteristics of a multicollinearity diagnostic:
  1. Want the procedure to correctly indicate if multicollinearity is present
  2. Want the procedure to provide some insight as to which regressors are causing the problem

Multicollinearity Diagnostics

- Correlation Coefficients
  - The pairwise correlation between two variables x_i and x_j is denoted r_{ij}
  - $|r_{ij}|$ close to unity is an indication of multicollinearity
  - However,there may be instances when multicollinearity is present, but the pairwise correlations do not indicate a problem

TABLE 9.4 Unstandardized Regressor and Response Variables from Webster, Gunst, and Mason [1974]

Observation, i	y_i	x_{i1}	x_{i2}	x_{i3}	x_{i4}	x_{i5}	x_{i6}
1	10.006	8.000	1.000	1.000	1.000	0.541	–0.099
2	9.737	8.000	1.000	1.000	0.000	0.130	0.070
3	15.087	8.000	1.000	1.000	0.000	2.116	0.115
4	8.422	0.000	0.000	9.000	1.000	–2.397	0.252
5	8.625	0.000	0.000	9.000	1.000	–0.046	0.017
6	16.289	0.000	0.000	9.000	1.000	0.365	1.504
7	5.958	2.000	7.000	0.000	1.000	1.996	–0.865
8	9.313	2.000	7.000	0.000	1.000	0.228	–0.055
9	12.960	2.000	7.000	0.000	1.000	1.380	0.502
10	5.541	0.000	0.000	0.000	10.000	–0.798	–0.399
11	8.756	0.000	0.000	0.000	10.000	0.257	0.101
12	10.937	0.000	0.000	0.000	10.000	0.440	0.432

Multicollinearity Diagnostics

- Mason, Gunst & Webster data

$$\mathbf{X'X} = \begin{bmatrix} 1.000 & 0.052 & -0.343 & -0.498 & 0.417 & -0.192 \\ & 1.000 & -0.432 & -0.371 & 0.485 & -0.317 \\ & & 1.000 & -0.355 & -0.505 & 0.494 \\ & & & 1.000 & -0.215 & -0.087 \\ & & & & 1.000 & -0.123 \\ & & & & & 1.000 \\ \text{Symmetric} & & & & & \end{bmatrix}$$

Variance Inflation Factors

- Variance inflation factors are very useful in determining if multicollinearity is present.

$$VIF_j = C_{jj} = (1 - R_j^2)^{-1}$$

- VIFs > 5 to 10 are considered significant
- The regressors that have high VIFs may have poorly estimated regression coefficients

R code

```
#Webster data
W <- read.csv("Webster.csv",h=T)
pairs(W,pch=20)
cor(W)
library(car)
model1=lm(y~x1+x2+x3+x4+x5+x6,data=W)
vif(model1)

# Acetylene data
A <- read.csv("Acetylene.csv",h=T)
pairs(A,pch=20)
A$x12=A$x1*A$x2
A$x13=A$x1*A$x3
A$x23=A$x2*A$x3
A$x1sq=A$x1^2
A$x2sq=A$x2^2
A$x3sq=A$x3^2
names(A)
cor(A)
library(car)
library(MASS)
model1=lm(y~x1+x2+x3+x12+x13+x23+x1sq+x2sq+x3sq,data=A)
vif(model1)

# standardized Acetylene
A_standard=as.data.frame(apply(A,2,function(x){(x-mean(x))/sd(x)}))
A_standard$x12=A_standard$x1*A_standard$x2
A_standard$x13=A_standard$x1*A_standard$x3
A_standard$x23=A_standard$x2*A_standard$x3
A_standard$x1sq=A_standard$x1^2
A_standard$x2sq=A_standard$x2^2
A_standard$x3sq=A_standard$x3^2
A_standard$y=A$y
model1=lm(y~x1+x2+x3+x12+x13+x23+x1sq+x2sq+x3sq,data=A_standard)
vif(model1)
model1=lm(y~x1+x2+x3+x12+x13+x23+x1sq+x2sq+x3sq,data=A)
vif(model1)
```

Variance Inflation Factors

TABLE 9.5 VIFs for Acetylene Data and Webster, Gunst, and Mason Data

Data, (A) Acetylene Centered Term VIF	Data, (B) Acetylene Uncentered Term VIF	Data, (C) Webster, Gunst, and Mason Term VIF
$x_1 = 374$	$x_1 = 2{,}856{,}749$	$x_1 = 181.83$
$x_2 = 1.74$	$x_2 = 10{,}956.1$	$x_2 = 161.40$
$x_3 = 679.11$	$x_3 = 2{,}017{,}163$	$x_3 = 265.49$
$x_1x_2 = 31.03$	$x_1x_2 = 2{,}501{,}945$	$x_4 = 297.14$
$x_1x_3 = 6565.91$	$x_1x_3 = 65.73$	$x_5 = 1.74$
$x_2x_3 = 35.60$	$x_2x_3 = 12{,}667.1$	$x_6 = 1.44$
$x_1^2 = 1762.58$	$x_1^2 = 9802.9$	
$x_2^2 = 3.17$	$x_2^2 = 1{,}428{,}092$	
$x_3^2 = 1158.13$	$x_3^2 = 240.36$	
Maximum VIF = 6565.91	Maximum VIF = 2,856,749	Maximum VIF = 297.14

Variance Inflation Factors

VIFs: A Second Look and Interpretation

- The square root of the *j*th VIF provides a measure of how much longer the confidence interval for the *j*th regression coefficient is because of multicollinearity
- For example, if $VIF_3 = 10$, then $\sqrt{VIF_3} \cong 3.3$ and the confidence interval is 3.3 times longer than if the regressors had been *orthogonal*

Methods for Dealing with Multicollinearity

- Collect more data
- Respecify the model
- Use Ridge Regression

Methods for Dealing with Multicollinearity

- Least squares estimation gives an unbiased estimate,

$$E(\hat{\beta}) = \beta$$

with minimum variance but may be very large

- Alternative: Find an estimate that is biased but with a smaller variance

Methods for Dealing with Multicollinearity

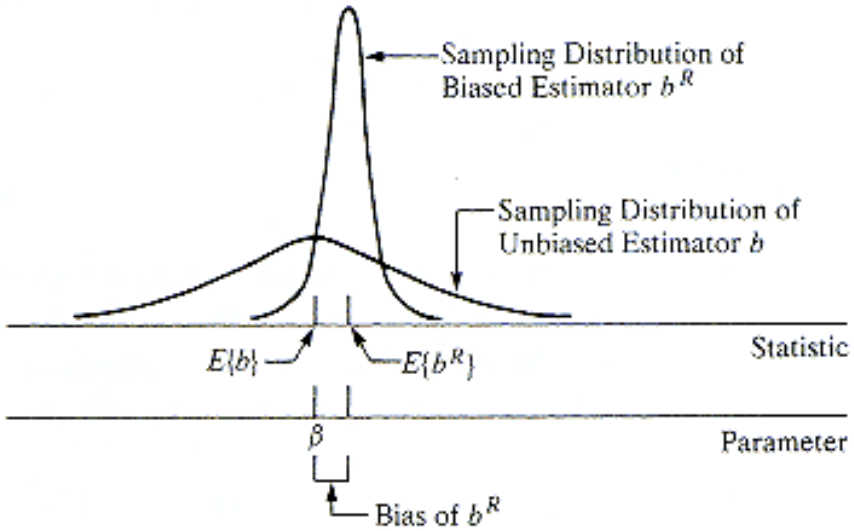
- Ridge Estimator $\hat{\beta}_R$

$$\hat{\beta}_R = (\mathbf{X}'\mathbf{X} + k\mathbf{I})^{-1}\mathbf{X}'\mathbf{y}$$
$$= (\mathbf{X}'\mathbf{X} + k\mathbf{I})^{-1}\mathbf{X}'\mathbf{X}\hat{\boldsymbol{\beta}}$$
$$= \mathbf{Z}_k\hat{\boldsymbol{\beta}}$$

  - k is a “biasing parameter”
  - k usually between 0 and 1

Methods for Dealing with Multicollinearity

- Ridge Regression
  - Modifies method of least squares to biased estimators of the regression coefficients



Methods for Dealing with Multicollinearity

- **Ridge Trace**
  - Plots *k* against the coefficient estimates
  - Will show impact of multicollinearity in the stability of the coefficients
  - Choose *k* such that $\hat{\beta}_R$ is stable and the MSE is acceptable

- Ridge regression is a good alternative if the model user wants to have all regressors in the model

Methods for Dealing with Multicollinearity

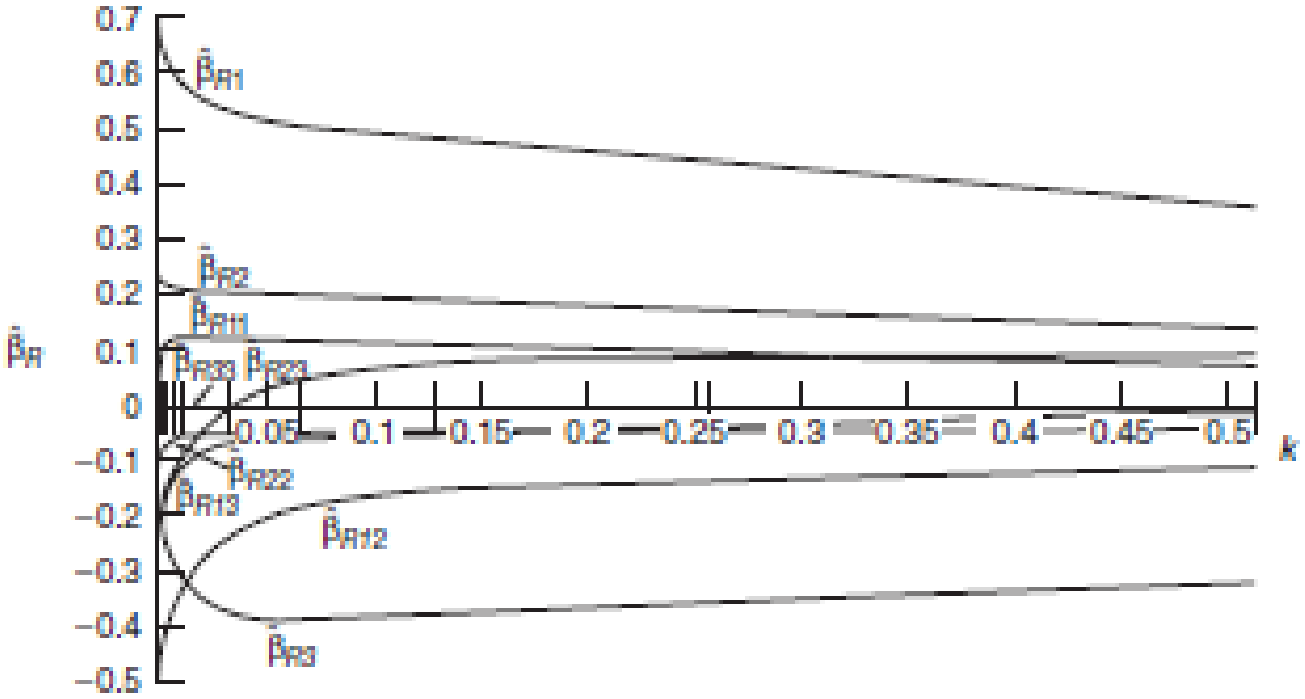
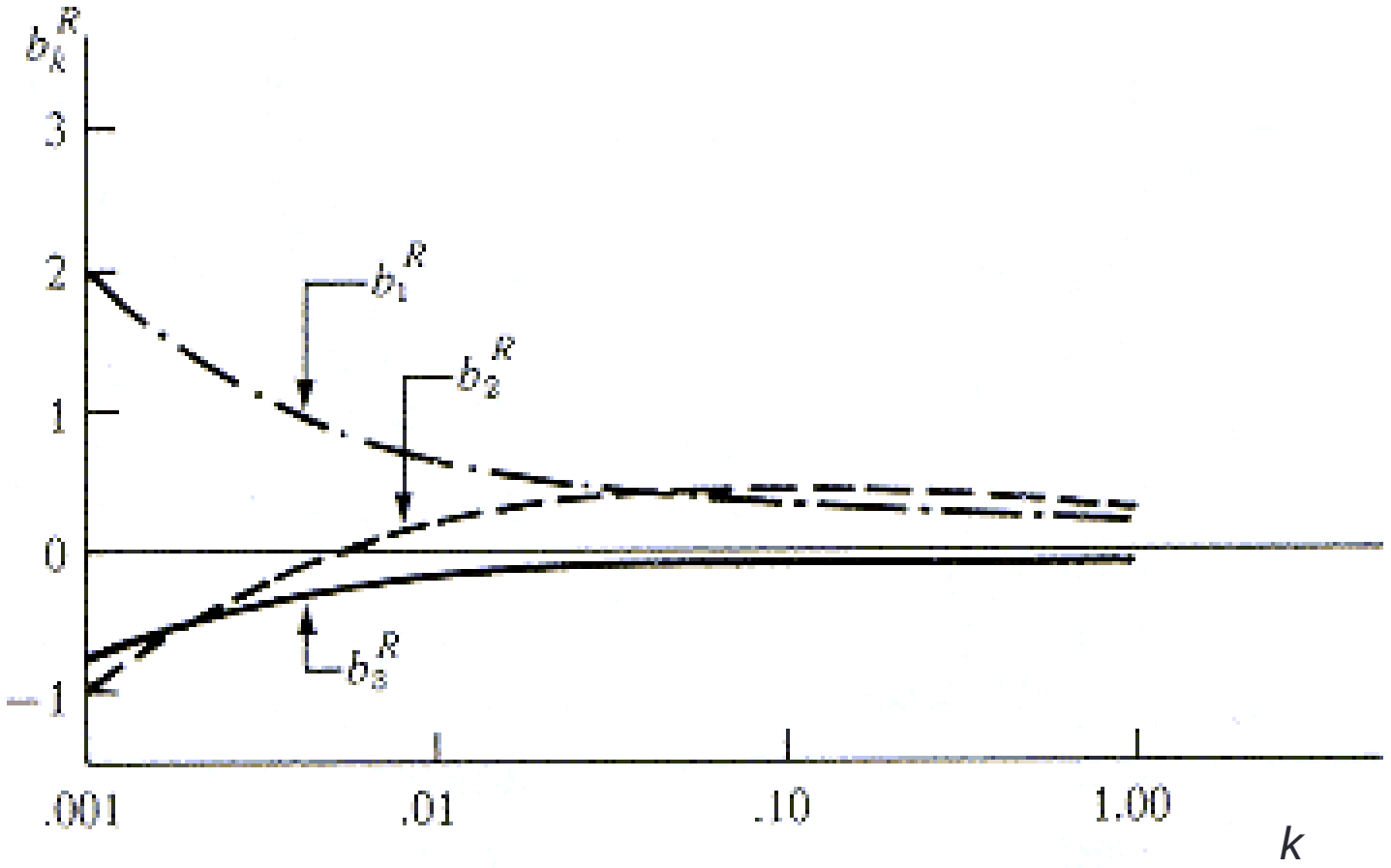


Figure 9.5 Ridge trace for acetylene data using nine regressors.

Methods for Dealing with Multicollinearity

- **Ridge Trace**



Methods for Dealing with Multicollinearity

- **Ridge Regression**

k	$(VIF)_1$	$(VIF)_2$	$(VIF)_3$	R^2
.000	708.84	564.34	104.61	.8014
.002	50.56	40.45	8.28	.7901
.004	16.98	13.73	3.36	.7864
.006	8.50	6.98	2.19	.7847
.008	5.15	4.30	1.62	.7838
.010	3.49	2.98	1.38	.7832
.020	1.10	1.08	1.01	.7818
.030	.63	.70	.92	.7812
.040	.45	.56	.88	.7808
.050	.37	.49	.85	.7804
.100	.25	.37	.76	.7784
.500	.15	.21	.40	.7427
1.000	.11	.14	.23	.6818

k	b_1^R	b_2^R	b_3^R
.000	4.264	−2.929	-1.561
.002	1.441	−.4113	−.4813
.004	1.006	−.0248	−.3149
.006	.8300	.1314	−.2472
.008	.7343	.2158	−.2103
.010	.6742	.2684	−.1870
.020	.5463	.3774	−.1369
.030	.5004	.4134	−.1181
.040	.4760	.4302	−.1076
.050	.4605	.4392	−.1005
.100	.4234	.4490	−.0812
.500	.3377	.3791	−.0295
1.000	.2798	.3101	−.0059

Methods for Dealing with Multicollinearity

- Ridge Regression
  - Comments
    - R^2 can be defined analogously to OLS
    - Estimates are more stable than OLS estimates and little affected by small changes in the data
    - Estimates provide good estimates of mean responses for levels of the predictor variables outside the region of the observations
    - Inferential procedures are not applicable since exact distributions are not known
    - Selection of k is judgmental
    - Exploratory tool used to reduce the number of predictor variables

R code

```
install.packages("genridge")
install.packages("pls")
library (genridge)
library(pls)
library(MASS)
library(car)

# body fat example
rm(list=ls())
BF <- read.csv("BodyFat.csv",h=T)
pairs(BF,pch=20)
cor(BF)
model1=lm(y~x1+x2+x3,data=BF)
vif(model1)
rreg <- lm.ridge(y~x1+x2+x3, BF, lambda=seq(0,1,.05))
select(rreg)
temp <- lm(y~x1+x2+x3, data=BF)
y <- BF[,"y"]
X0 <- model.matrix(temp)[,-1]
lambda <- seq(0, 1, 0.05)
aridge <- ridge(y, X0, lambda=lambda)
traceplot(aridge)
coef(aridge)
vridge <- vif(aridge)
vridge
rreg <- lm.ridge(y~x1+x2+x3, BF, lambda=.31)
temp <- princomp(bodyfat)
summary(temp)
outpcr <- pcr(y~x1+x2+x3, data=BF,ncomp=2)
summary(outpcr)
outpcr <- pcr(y~x1+x2+x3, data=BF,ncomp=3)
summary(outpcr)

# table B21
B21 <- read.csv("data-table-B21.csv",h=T)
pairs(B21,pch=20)
cor(B21)
library(car)
model1=lm(y~x1+x2+x3+x4,data=B21)
vif(model1)
rreg <- lm.ridge(y~x1+x2+x3+x4, B21, lambda=seq(0,1,.05))
select(rreg)
temp <- lm(y~x1+x2+x3+x4, data=B21)
y <- B21[,"y"]
X0 <- model.matrix(temp)[,-1]
lambda <- seq(0, 1, 0.05)
aridge <- ridge(y, X0, lambda=lambda)
traceplot(aridge)
coef(aridge)
vridge <- vif(aridge)
vridge
rreg <- lm.ridge(y~x1+x2+x3+x4, B21, lambda=.08)
```

R code

```
# webster
W <- read.csv("Webster.csv",h=T)
pairs(W,pch=20)
cor(W)
library(car)
model1=lm(y~x1+x2+x3+x4+x5+x6,data=W)
vif(model1)
rreg <- lm.ridge(y~x1+x2+x3+x4+x5+x6, W, lambda=seq(0,1,.05))
select(rreg)
temp <- lm(y~x1+x2+x3+x4+x5+x6, data=W)
y <- W[,"y"]
X0 <- model.matrix(temp)[,-1]
lambda <- seq(0, 1, 0.05)
aridge <- ridge(y, X0, lambda=lambda)
traceplot(aridge)
coef(aridge)
vridge <- vif(aridge)
vridge
rreg <- lm.ridge(y~x1+x2+x3+x4+x5+x6, W, lambda=.25)

# acetylene
A <- read.csv("Acetylene.csv",h=T)
pairs(A,pch=20)
A$x12=A$x1*A$x2
A$x13=A$x1*A$x3
A$x23=A$x2*A$x3
A$x1sq=A$x1^2
A$x2sq=A$x2^2
A$x3sq=A$x3^2
names(A)
cor(A)
library(car)
library(MASS)
model1=lm(y~x1+x2+x3+x12+x13+x23+x1sq+x2sq+x3sq,data=A)
vif(model1)
rreg <- lm.ridge(y~x1+x2+x3+x12+x13+x23+x1sq+x2sq+x3sq, A, lambda=seq(0,1,.05))
select(rreg)
y <- A[,"y"]
X0 <- model.matrix(model1)[,-1]
lambda <- seq(0, 1, 0.05)
library(genridge)
aridge <- ridge(y, X0, lambda=lambda)
traceplot(aridge)
coef(aridge)
vridge <- vif(aridge)
vridge
rreg <- lm.ridge(y~x1+x2+x3+x4+x5+x6, W, lambda=.25)
```

R code

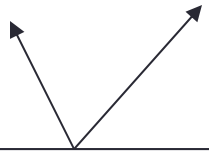
- A_standard=as.data.frame(apply(A,2,function(x){(x-mean(x))/sd(x)}))
- A_standard$x12=A_standard$x1*A_standard$x2
- A_standard$x13=A_standard$x1*A_standard$x3
- A_standard$x23=A_standard$x2*A_standard$x3
- A_standard$x1sq=A_standard$x1^2
- A_standard$x2sq=A_standard$x2^2
- A_standard$x3sq=A_standard$x3^2
- A_standard$y=A$y
- model1=lm(y~x1+x2+x3+x12+x13+x23+x1sq+x2sq+x3sq,data=A_standard)
- vif(model1)
- model1=lm(y~x1+x2+x3+x12+x13+x23+x1sq+x2sq+x3sq,data=A)
- vif(model1)

Principal-Component Regression

- Provides composites that are linear combinations of the original variables
- Creates composites are uncorrelated
- Utilizes a subset of the components as the regressors that explain much of the variance
- May be limited by inability to attach concrete meanings to the composites

TABLE 9.12 Principal Components Regression for the Acetylene Data

Parameter	Principal Components in Model									
	A		B		C		D		E	
	z_1		z_1, z_2		z_1, z_2, z_3		z_1, z_3, z_3, z_4		z_1, z_3, z_3, z_4, z_5	
	Standarized Estimate	Original Estimate	Standardized Estimate	Original Estimate	Standardized Estimate	Original Estimate	Standardized Estimate	Original Estimate	Standardized Estimate	Original Estimate
β_0	.0000	42.1943	.0000	42.2219	.0000	36.6275	.0000	34.6688	.0000	34.7517
β_1	.1193	1.4194	.1188	1.4141	.5087	6.0508	.5070	6.0324	.5056	6.0139
β_2	.0466	.5530	.0450	.5346	.0409	.4885	.2139	2.5438	.2195	2.6129
β_3	−.1457	−1.7327	−.1453	−1.7281	−.4272	−5.0830	−.4100	−4.8803	−.4099	4.8757
β_{12}	−.0772	−1.0369	−.0798	−1.0738	−.0260	−.3502	−.1123	−1.5115	−.1107	−1.4885
β_{13}	.1583	2.0968	.1578	2.0922	−.0143	−.1843	−.0597	−.7926	−.0588	−.7788
β_{23}	.0889	1.2627	.0914	1.2950	.0572	.8111	.1396	1.9816	.1377	1.9493
β_{11}	−.1429	−2.1429	−.1425	−2.1383	.1219	1.8295	.1751	2.6268	.1738	2.6083
β_{22}	.0091	.0968	.0065	.0691	−.1280	−1.3779	−.0460	−.4977	−.0533	−.5760
β_{33}	−.1644	−1.9033	−.1639	−1.8986	−.0786	−.9125	−.0467	−.5392	−.0463	−.5346
R^2	.5217		.5218		.9320		.9914		.9915	
MS_{Res}	.079713		.079705		.011333		.001427		.00142	



Models D and E are similar

R code

```
install.packages("genridge")
install.packages("pls")
library (genridge)
library(pls)
library(MASS)
library(car)

# body fat example
rm(list=ls())
BF <- read.csv("BodyFat.csv",h=T)
pairs(BF,pch=20)
cor(BF)
model1=lm(y~x1+x2+x3,data=BF)
vif(model1)
rreg <- lm.ridge(y~x1+x2+x3, BF, lambda=seq(0,1,.05))
select(rreg)
temp <- lm(y~x1+x2+x3, data=BF)
y <- BF[,"y"]
X0 <- model.matrix(temp)[,-1]
lambda <- seq(0, 1, 0.05)
aridge <- ridge(y, X0, lambda=lambda)
traceplot(aridge)
coef(aridge)
vridge <- vif(aridge)
vridge
rreg <- lm.ridge(y~x1+x2+x3, BF, lambda=.31)
temp <- princomp(bodyfat)
summary(temp)
outpcr <- pcr(y~x1+x2+x3, data=BF,ncomp=2)
summary(outpcr)
outpcr <- pcr(y~x1+x2+x3, data=BF,ncomp=3)
summary(outpcr)
```