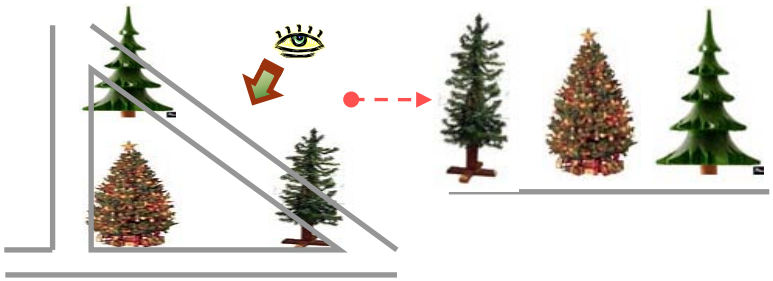
4장 주성분 분석 (PCA: Principal Component Analysis)

- 예 1) 바지구입-우리몸의 치수 모두를 알아야하는가?
-
- - 변수 : 허리둘레, 기장, 엉덩이둘레, 허벅지둘레, 무릎높이...
- - 허리둘레, 기장 두 변수면 충분
- (이것이 주성분 분석의 개념)
- 즉, 원 변수에 의해 주성분 변수를 구하고 일부 주성분에 의해 원 변수의 변동을 충분히 설명되는지 알아보는 것을 주성분 분석이라 한다.

- 예 2) 주성분 개념 : 차원의 축약
- 어느 곳에서 바라보면 희생되는 정보가 가장 적은가?



-

- 즉,주성분 분석은 원 변수의 정보(나무 위치; 실제분석에서는 변동)을 가장 잘 표현 할 수 있는 시점(주성분 변수)를 얻는 과정이다.

- 1) 주성분 분석은...

"분산을 최대"로 하는 방향(변량)을 찾아 그로부터 데이터를 축약하여 데이터의 개개 정보를 보다 보기 쉽게 표현해주는 분석 .

- 2) 데이터간 상관관계로 인한 정보손실의 최소화

-> 데이터의 분산의 최대화

- 3) 주성분분석은 p(≥3)개의 원변수를 m개(일반적 3개 이하)로 줄이기 위해

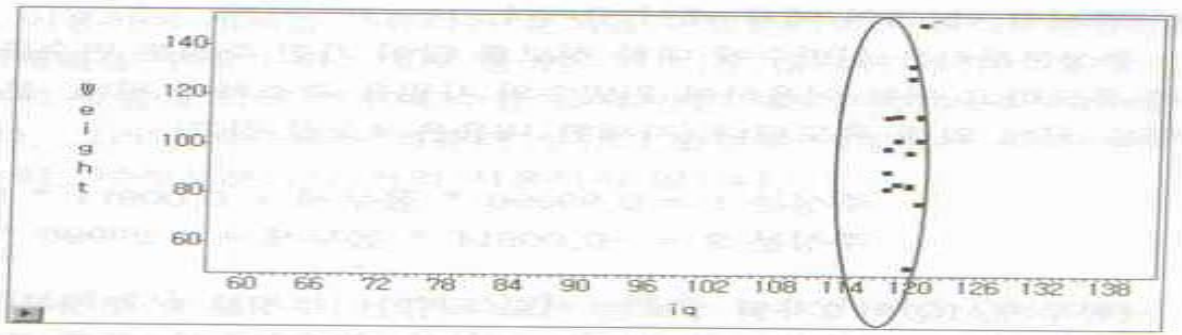
(1) 원변수의 선형결합으로 주성분 변수를 구하고,

(2) 선형계수(부하)값의 크기를 활용 적절한 주성분 변수이름 부여하는데 목적이 있다.

(변수 축약후, 일반적으로 손실정도는 20%이다)

4.1 맛보기 - 예제 (p 71)

19명의 학생, 몸무게(pound), IQ
1) 두 변수로부터 공분산행렬을 구해보면 IQ변동은 미미하고 몸무게 변동이 대부분 차지



다음은 위의 데이터로부터 공분산행렬을 구하고 그것을 이용하여 고유치오
는 고유벡터를 구한 결과이다.

Simple Statistics

	Weight	iq
Mean	100.0263158	119.6842105
StD	22.7739335	1.1081833

Covariance Matrix

	Weight	iq
Weight	518.6520468	4.7309942
iq	4.7309942	1.2280702

Eigenvalues of the Covariance Matrix

	Eigenvalue	Difference	Proportion	Cumulative
1	518.695300	517.510484	0.9977	0.9977
2	1.184817		0.0023	1.0000

Eigenvectors

	Prin1	Prin2
Weight	0.999958	-.009142
iq	0.009142	0.999958

▪4.1 맛보기

▪4. 1. 1 예제 (p 71)

- ▪19명의 학생, 몸무게(pound), IQ

1) 두 변수로부터 공분산행렬을 구해보면 IQ변동은 미미하고 몸무게 변동이 대부분 차지

$$S = \hat{\Sigma} = \begin{pmatrix} 518 & 4.7 \\ 4.7 & 1.22 \end{pmatrix}$$

2) 주성분 변수

- 원변수의 선형 결합
- 제일 주성분이 원변수의 변동을 가장 많이 설명: 원 변수 공분산 행렬 (상관계수 행렬)이 시작 단계
- 주성분의 원변수 설명력=고유치: 518.695, 1.1848
- 선형계수: 고유벡터

$$Y_1 = l_{11} \times Weight + l_{12} \times IQ$$
$$Y_2 = l_{21} \times Weight + l_{22} \times IQ$$

▪3) 주성분 1 = 0.99996*몸무게 + 0.00941*IQ

▪ (Prin1)

▪ 주성분 2 =-0.00914*몸무게 + 0.99996*IQ

▪ (Prin2)

▪4. 1. 2 주성분분석활용(페이지 72):

▪ 주성분 분석은 중간 단계이다.

즉, 데이터 축약을 통한 개체 분류나 변수 구조 탐색 도구

1)데이터 스크린

▪ 변수가 많을때 일반적 줄기잎그림, 산점도 등으로 한계가 있으므로 주성분분석에 의해

- Data 를 저차원 축소 => 개체 특성 및 이상치 탐지

2)군집

▪ 다변량인 경우 군집을 나누기 어려움 -> 원데이터의 저차원 축소(주성분북석 활용) => 주성분변수에 의해 군집개체 분류 혹은 해석

3)판별분석

- 주성분 분석에 의한 주성분 변수 사용(축약변수)에 의해 판별분석을 함

4)회귀분석에서

다중분석시 다중공선성 문제

- 문제되는 설명변수 제외
- 주성분변수를 설명변수로 하는 회귀분석실시
  - 주성분 변수의 의미가 명확하게 해석되지 않으면 회귀분석 결과해석이 어려움

4.2 주성분 분석(PCA) – 주성분 구하기

- 주성분 (변수)?
  - 주성분은 원 변수의 선형결합이다. (합성변량) Y=LX

$$\underline{y} = \begin{pmatrix} y_1 \\ y_2 \\ \vdots \\ y_p \end{pmatrix} = \begin{pmatrix} l_{11}\, l_{12} \cdots l_{1p} \\ l_{21}\, l_{12} \cdots l_{2p} \\ \vdots \\ l_{p1}\, l_{p2} \cdots l_{pp} \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ x_p \end{pmatrix} = L\underline{x}$$

  - P 개의 원 변수의 공분산 행렬을 이용 선형계수 구하고 이를 이용, 원 변수의 선형결합으로 주성분을 얻는 과정.
- y=주성분 벡터, L= 고유벡터결합, X=독립변수벡터
- 원칙
  - 원 변수의 선형결합
  - 주성분 변수 들은 서로 독립(상관계수 = 0)
  - 제1 주성분가 원 변수 변동 설명력 가장 큼
  - 원 변수의 개수만큼 주성분 변수가 존재: :
  - 원 변수 변동을 설명하는 비율 누적 80%인 주성분만 이용

- 원 변수 벡터 공분산 행렬 (covariance matrix, Σ, S)

$$\underline{x} = \begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ x_p \end{pmatrix} \qquad \Sigma = \begin{bmatrix} \sigma_{11} & \sigma_{12} & \cdots \sigma_{1p} \\ \sigma_{21} & \sigma_{22} & \cdots \sigma_{2p} \\ \cdots & & \\ \sigma_{p1} & \sigma_{p2} & \cdots \sigma_{pp} \end{bmatrix}$$

- 공분산 행렬(Σ)의 고유치
  - 계산 방법: $|\Sigma_{p\times p} - \lambda I_p| = \underline{0}$ 을 만족하는 λ들을 고유치라 한다.
  - 고유치는 행렬의 차수만큼 존재한다. $\lambda_1 \geq \lambda_2 . \geq .. \geq \lambda_p$
  - 고유치 의미: 주성분 변수의 분산 (슬라이드 9에서 타원의 폭과 높이에 해당)

- 공분산 행렬(Σ)의 고유벡터
  - 계산방법: $\Sigma\ \underline{e}_i = \lambda_i \underline{e}_i$을 만족하는 벡터 e를 고유벡터라 한다.
  - 고유벡터는 무수히 많이 존재 $\underline{e}_i' \underline{e}_j = 0 \ \ for\ i \neq j$
  - 고유벡터는 서로 orthogonal 하다.
  - 고유벡터를 주성분 계산 선형계수로 사용한다.

- 주성분 선형계수 추정 : 표본 공분산행렬 이용

주성분 분석 (PCA)

▪주성분 기여율

1) 주성분 변수의 원변수 총 변동을 설명하는 정도

2) 원 변수의 총변동은

$$\sum_i V(x_i) = \sum_i \hat{\sigma}_{ii} = \sum_i s_{ii}$$

- 각 주성분의 설명력의 합은 원변수 총 변동과 동일하다.
- 따라서, 주성분기여율은 총 변동 중 각 주성분의 설명력

3) 주성분 기여율 = $\dfrac{V(x_i)}{\sum_i V(x_i)} = \dfrac{\lambda_i}{\sum_i \lambda_i}$

▪ 주성분 구하기

▪제 1 주성분 (first principal component)

- $\underline{a}_1'\underline{a}_1 = 1$ 을 만족하는 벡터 중 $V(\underline{a}_1'(\underline{x}-\underline{\mu}))$ 을 최대화 하는 벡터 $\underline{a}_1$ 을 선형계수로 하여 구해진 합성변수.

$$\underline{y}_1 = \underline{a}_1'(\underline{x}-\underline{\mu})$$

- 공분산 행렬(S)로부터 얻어진 고유치 $\lambda 1$ 에 대응하는 고유벡터 e1 중 $\underline{e}_1'\underline{e}_1 = 1$을 만족하는 고유벡터

$$\underline{l}_1' = \underline{e}_1'$$

▪제2 주성분 (second PC)

- $\underline{a}_1'\underline{a}_2 = 0, \underline{a}_2'\underline{a}_2 = 1$을 만족하는 벡터 중 $V(\underline{a}'_2(\underline{x}-\underline{\mu}))$ 을 최대화 하는 벡터 $\underline{a}_2$를 선형계수로 하여 계산한 합성변수.
- 공분산 행렬(S)로부터 얻어진 고유치 $\lambda 2$에 대응하는 고유벡터 e2 중 $\underline{e}_1'\underline{e}_2 = 0, \underline{e}_2'\underline{e}_2 = 1$을 만족하는 고유벡터

$$\underline{l}_2' = \underline{e}_2'$$

▪이와 같은 방법으로 순차적으로 구한다.

▪ 주성분 부하 벡터

- ▪주성분을 만들때 사용되는 원 변수의 선형계수
- ▪ 주성분 이름 부여에 사용.
- ▪ 주의 사항
  - 원변수 단위는 유사해야함. (상관계수행렬사용)
  - 부하의 크기 비교는 주성분 내에서만 가능.
    (주성분 간 성분 부하 값 비교는 의미 없다)

주성분 분석 (PCA)

- 상관계수 행렬 (correlation matrix, R) 사용 (페이지 80)
  - 원변수의 측정 단위가 상이한 경우: 주성분 계산 시 단위 크기가 큰 원변수의 영향(분산이 크므로)이 크다.
  - 문제 해결을 위하여 상관계수 행렬 사용하여 고유치, 고유벡터를 구한다.
  - 주성분 구하는 절차는 공분산 행렬과 동일하다.

$$\hat{R} = \begin{bmatrix} 1 & r_{12} & \dots & r_{1p} \\ r_{21} & 1 & \dots & r_{2p} \\ & & \dots & \\ r_{p1} & r_{p2} & \dots & 1 \end{bmatrix}$$

- 주성분 점수
  - 주성분 변수는 계산 식을 실제 데이터에 의해 계산된 값을 주성분 점수(score)라 한다. k번째 개체의 j번째 주성분 점수 계산 식은 다음과 같다.
  - 선형계수에 의해 각 주성분 점수 계산
  - 좌표에 표현하여 이상치 발견, 주성분 변수간 관계 파악

$$\begin{pmatrix} y_1 \\ y_2 \\ \cdots \\ y_p \end{pmatrix} = \begin{pmatrix} l_{11}\, l_{12} \cdots l_{1p} \\ l_{21}\, l_{22} \cdots l_{2p} \\ \cdots \\ l_{p1}\, l_{p2} \cdots l_{pp} \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ \cdots \\ x_p \end{pmatrix} = \begin{pmatrix} e_{11}\, e_{12} \cdots e_{1p} \\ e_{21}\, e_{22} \cdots e_{2p} \\ \cdots \\ e_{p1}\, e_{p2} \cdots e_{pp} \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ \cdots \\ x_p \end{pmatrix}$$

- 주성분 성질 2

1) 주성분의 변동은 고유치와 같고, 변동의 크기는 제일, 제이, ... 순이다.

$$Var(Y_i) = \underline{e}_i' \Sigma \underline{e}_i = \lambda_i$$

2) 주성분 변수는 서로 독립이다.

$$Cov(Y_i, Y_k) = \underline{e}_i' \Sigma \underline{e}_k = 0, \ for\ i \neq k$$

3) 주성분 변수의 변동 합은 원변수 변동 합과 동일하다.

$$\sum_{i=1}^{p} V(Y_i) = \sum_{i=1}^{p} \lambda_i = \sum_{i=1}^{p} V(X_i)$$

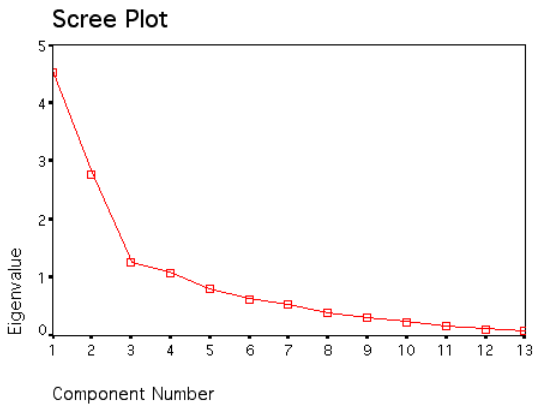
- 주성분 계수 시각적 표현
  - 주성분 변수(점수) 이름 부여에 도움
  - 실제 계수 값을 보고도 판단할 수 있으나 시각적 표현으로 인하여 손쉽게 이름 부여 가능

주성분 분석 (PCA)

- 주성분 개수 결정
  - 80% rule (공분산 행렬 사용 시)
  - 고유치 1 이상 (상관계수 행렬 사용 시)
- 총 변동 설명 비율

$$0.7 \le \frac{\lambda_1}{tr(S)} + \frac{\lambda_2}{tr(S)} + \cdots \le 0.9$$

  - 일반적으로 총 변동의 약 80% 설명가능하면 사용.
- Scree 도표 이용 (페이지 80)
  - Y축 고유치, x 축을 주성분 순차 번호 산점도
  - 고유치가 감소 경향을 시각적으로 표현
  - 급격히 감소하는 곳에서 주성분 개수 결정 (elbow)
  - 시각적 표현, 실제로는 80% 규칙 이용



- 예제

- 변수가 2개인 경우(PCA_example)
  - 데이터 코딩

```
data student;
input weight iq@@;
cards;
123 110 74 115 145 125 64 112 84 118 128 124 79 121
112 120 107 115 67 117 98 119 105 123 95 118 79 121
81 113 91 115 142 119 84 114  85 122 93 124
;
run;
```

  - 공분산행렬

```
proc corr data=student cov;
var weight iq;
run;
```

공분산행렬, DF = 19

	weight	iq
weight	542.9052632	35.1578947
iq	35.1578947	18.6184211

주성분 (계수) 구하기 - 예제에서..

- 고유치와 고유벡터

```
proc IML;
/*reset print all;*/
x={542.9 35.16, 35.16 18.62};
call eigen(m,e,x);
print m,e,x;
run;
```

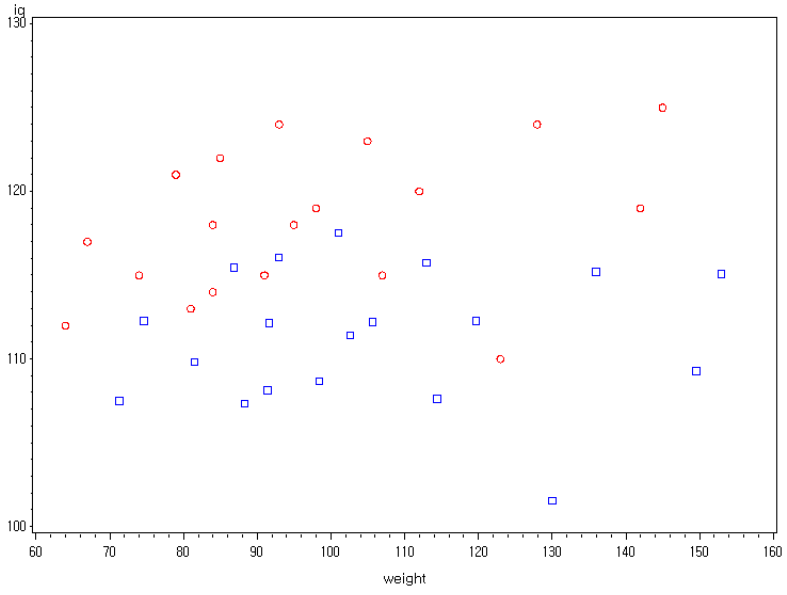
```
m

545.24744
16.272561
e

0.9977787 -0.066616
0.0666162 0.9977787
x

542.9     35.16
35.16     18.62
```

- 주성분 점수

```
data one;
set student;
y1=0.9978*weight+0.0666*iq;
y2=-0.0666*weight+0.9978*iq;
run;
```

- 원변수 관측치와 주성분점수의 산점도

```
proc gplot data=one;
symbol1 v=circle c=red;
symbol2 v=square c=blue;
plot iq*weight y2*y1/overlay;
run;
```



예제 2

applicant.txt 파일 (3장 p58)-회사 48명 지원, 6명 선발, 15개 변수

```
data app;
infile "C:₩data₩applicant.txt";
input id l ap aa li sc lc ho sm ex dr am gc po kj su;
run;

proc princomp data=app out=score covariance;
var l -- su;
run;
```

Covariance Matrix

	l	ap	aa	li	sc
l	7.14893617	1.25531915	0.23404255	2.29787234	0.59574468
ap	1.25531915	3.86524823	0.48226950	2.09397163	2.04787234
aa	0.23404255	0.48226950	3.95035461	0.00886525	0.00531915
li	2.29787234	2.09397163	0.00886525	7.87189716	2.05186170
sc	0.59574468	2.04787234	0.00531915	2.05186170	5.84707447
lc	1.93617021	2.31382979	0.48404255	4.29388298	6.19015957
ho	-0.72340426	1.76241135	-0.15248227	4.58953901	2.51329787
sm	2.48936170	3.31028369	0.37411348	3.48980496	6.65026596
ex	4.85106383	0.91666667	1.74645390	1.30629433	0.12101064
dr	2.72340426	1.97340426	0.54787234	3.25132979	5.01994681
am	2.23404255	3.17198582	0.25709220	2.85416667	5.97739362
gc	2.74468085	3.02127660	1.19148936	4.28191489	5.29255319
po	3.12765957	3.17553191	1.83510638	5.40824468	5.17154255
kj	3.31914894	1.48404255	-1.70744681	5.10771277	3.09973404
su	5.17021277	2.49290780	0.91843972	3.02748227	1.99734043

Covariance Matrix

	lc	ho	sm	ex	dr
l	1.93617021	-0.72340426	2.48936170	4.85106383	2.72340426
ap	2.31382979	1.76241135	3.31028369	0.91666667	1.97340426
aa	0.48404255	-0.15248227	0.37411348	1.74645390	0.54787234
li	4.29388298	4.58953901	3.48980496	1.30629433	3.25132979
sc	6.19015957	2.51329787	6.65026596	0.12101064	5.01994681
lc	10.04920213	2.85904255	8.91888298	1.54388298	6.51728723
ho	2.85904255	6.42375887	2.09131206	-1.30762411	2.09308511
sm	8.91888298	2.09131206	11.82934397	2.90647163	8.25930851
ex	1.54388298	-1.30762411	2.90647163	10.94636525	3.28856383
dr	6.51728723	2.09308511	8.25930851	3.28856383	8.68750000
am	7.04920213	1.59663121	8.67774823	1.89849291	6.75132979
gc	8.49468085	2.96808511	8.16489362	3.00531915	6.38829787

예제 2 결과 (고유벡터, 고유치)

Eigenvalues of the Covariance Matrix

	Eigenvalue	Difference	Proportion	Cumulative
1	66.5364216	48.3558875	0.5430	0.5430
2	18.1805340	7.5895485	0.1484	0.6914
3	10.5909855	3.8230376	0.0864	0.7778
4	6.7679478	2.7823032	0.0552	0.8330
5	3.9856446	0.3579742	0.0325	0.8656
6	3.6276704	0.7119224	0.0296	0.8952
7	2.9157480	0.0802174	0.0238	0.9190
8	2.8355306	0.8804081	0.0231	0.9421
9	1.9551225	0.3416313	0.0160	0.9581
10	1.6134912	0.4770192	0.0132	0.9712
11	1.1364720	0.2636018	0.0093	0.9805
12	0.8728702	0.1661951	0.0071	0.9876
13	0.7066751	0.1981398	0.0058	0.9934
14	0.5085353	0.2071663	0.0042	0.9975
15	0.3013690		0.0025	1.0000

Total Variance 122.53501773

적절한 주성분 수 = 4개

Eigenvectors

	Prin1	Prin2	Prin3	Prin4	Prin5	Prin6	Prin7	Prin8
l	0.149129	0.371461	0.200481	-.277311	0.636939	0.143537	0.054334	0.188839
ap	0.132250	-.029296	0.041918	0.134231	0.042210	0.757039	0.121848	-.269341
aa	0.029611	0.101846	-.131030	0.603168	0.167474	-.002238	0.117237	0.365044
ii	0.203126	-.093042	0.619733	0.126399	0.053473	-.018911	-.134527	0.046300
sc	0.231436	-.235740	-.189273	-.072088	-.025117	-.012126	0.247911	-.103262
ic	0.336870	-.195978	-.124714	0.052788	0.231817	-.367360	-.350780	-.318030
ho	0.120238	-.300549	0.447178	0.255587	-.334369	0.058714	0.177661	-.094947
sm	0.379017	-.090010	-.281581	-.172303	-.177778	0.093920	-.064038	-.086513
ex	0.164016	0.636212	0.025043	0.166245	-.191487	-.297426	0.480589	-.388399
dr	0.316050	0.012486	-.113315	-.134844	-.338054	-.118876	0.078867	0.577877
am	0.312106	-.122150	-.244517	-.147307	0.105416	0.226453	0.342601	0.037926
gc	0.338764	-.074347	-.050497	0.206271	0.258316	-.129935	-.121565	-.236845
po	0.357165	-.024920	0.041308	0.317232	0.108875	-.028784	0.023782	0.291193
kj	0.226076	-.044837	0.385206	-.459715	-.026846	-.148771	0.185820	0.030878
su	0.274483	0.470867	0.016815	-.015962	-.349972	0.254634	-.570749	0.003511

Eigenvectors

	Prin9	Prin10	Prin11	Prin12	Prin13	Prin14	Prin15
l	0.378998	0.224362	0.092656	-.190687	0.001965	-.135682	-.020959
ap	-.033120	-.028399	-.092894	0.086091	0.488649	0.206481	0.015430
aa	-.060954	0.450152	-.199223	0.387155	-.124848	0.099539	0.106257
ii	-.517090	0.135003	0.321951	0.069673	0.044180	-.182957	-.311511
sc	0.264633	0.150084	0.477289	0.176181	-.220675	0.431442	-.436694
ic	0.073486	0.160766	0.213320	0.123700	0.357672	0.037881	0.442122
ho	0.539060	0.137682	-.038002	-.139078	-.155159	-.256097	0.232601
sm	-.241327	0.543114	-.297876	-.458581	-.115634	-.094137	-.113068
ex	-.087106	0.006795	0.072428	-.064734	0.122574	-.031972	-.001808
dr	0.162946	-.080129	0.015671	0.113254	0.544965	-.180792	-.156981
am	-.242374	-.237856	0.204549	0.237351	-.309290	-.448256	0.338946
gc	0.168305	-.356813	-.470429	0.160937	-.086183	-.226942	-.467283
po	-.101284	-.414023	0.055047	-.524681	-.106388	0.411102	0.176385
kj	-.103965	0.010538	-.436794	0.318139	-.095263	0.416567	0.208082
su	0.132791	-.061706	0.116204	0.223043	-.306782	0.069812	0.075094

applicant.txt 파일

예제 2 결과

```
proc print data=score;
var prin1-prin15;
run;
```

OBS	Prin1	Prin2	Prin3	Prin4	Prin5	Prin6	Prin7	Prin8	Prin9	Prin10	Prin11	Prin12	Prin13	Prin14	Prin15
1	4.3040	-0.3819	-1.7574	-5.61558	-3.01763	1.46534	-2.12959	-0.65704	0.85108	-1.62724	0.39194	-0.27849	-0.79130	-1.48713	0.02523
2	10.1416	0.4179	0.0874	-3.08784	-0.75680	2.51241	-0.63342	-0.30638	0.63645	0.90685	0.92532	-0.78945	0.61561	0.14748	-0.48044
3	6.5297	-0.1686	-0.3237	-4.52532	-2.25663	1.20628	-1.18437	-0.48122	1.77632	-1.77433	0.70084	0.69604	0.09582	-0.80442	-0.07606
4	-1.3281	2.1759	1.0981	2.81460	-0.14716	-2.42384	2.47628	-1.22766	0.73820	-2.44594	-1.45842	0.11382	-0.76596	0.59535	0.13349
5	1.4804	3.4916	3.2469	2.45018	-0.93626	0.06325	1.18054	-0.26662	-1.60377	-1.31198	-2.78089	-1.19467	-0.18763	-0.07292	-0.32141
6	2.3832	2.4176	1.0401	1.18088	-0.72581	-2.06666	2.27863	-2.15546	1.97112	0.24681	-0.23031	-0.50451	0.67968	-0.08674	-0.82357
7	9.5935	4.9223	0.3070	-0.09617	-0.19334	0.72313	2.09228	-0.34014	-1.11458	0.06308	0.49247	0.14184	-0.12525	-0.23300	0.91097
8	11.0723	4.4857	0.0833	0.09941	0.07425	-0.05617	2.24977	-0.02448	-0.61013	0.39768	0.09231	1.42121	0.25041	0.34470	0.60647
9	7.5936	4.6013	1.4609	-0.51860	0.14971	0.54337	1.32018	-0.38636	0.10112	-1.37133	1.25330	1.48250	0.63960	0.01900	0.63040
10	8.2294	-1.2104	-8.8270	2.10878	-1.74097	0.29873	-2.27154	2.03244	1.48389	-0.31553	-0.52425	0.51187	-0.89341	-0.33875	0.34233
11	4.3825	-0.2660	-11.4878	1.81192	0.52774	-0.62679	1.86961	2.40398	-0.87885	-1.30960	-0.16404	-1.38261	0.43450	1.28613	-0.05546
12	5.9910	-3.0709	-6.3429	3.46936	0.53709	-0.63735	-2.00002	1.73552	0.67867	-0.73149	-0.09103	-0.26190	-0.35374	0.97534	-0.46557
13	-0.7646	1.4111	4.4261	1.70872	-0.99571	2.93693	-0.95003	1.14443	-2.31480	0.35858	0.61956	-0.40296	-0.51133	0.75182	0.47135
14	-1.9066	2.1314	3.7461	2.18465	2.49114	2.89800	1.22027	-0.05927	-1.38535	-1.23385	0.88981	-0.24152	-1.48368	0.37263	-0.78608
15	-2.9391	2.0457	2.6388	4.05528	-2.41299	-0.08645	0.86715	-0.52156	0.29992	-1.19660	-0.41182	-0.33007	1.20904	0.07981	-0.62649
16	6.8485	2.7686	-1.0857	-0.41740	-2.94363	0.54416	-0.72676	-1.30382	-0.90648	0.86512	1.20734	-1.48101	1.46841	0.72003	0.73070
17	3.2896	2.1463	0.3424	-1.81086	-0.72396	0.83018	1.68090	1.08430	0.15339	0.48915	1.16158	0.58469	-1.30259	-0.05121	-0.89714
18	-2.4699	-1.3436	-2.3240	-1.03799	2.76169	-0.01027	0.12545	-2.49319	0.45347	1.33920	-0.85498	2.93636	0.49277	0.60615	-0.00083
19	-2.6360	-0.2087	-2.8799	-0.07141	3.61957	-1.00240	-0.30385	-2.96844	-0.39617	0.38539	-1.12320	1.95330	-0.13974	0.14073	-0.15597
20	4.9529	-2.8031	1.6593	0.92051	-1.10014	0.30097	-2.82667	-0.32231	0.19497	-1.73531	-0.51384	1.88521	-0.92029	1.20177	0.73344
21	3.3098	-2.7063	0.6176	1.89694	-2.15566	0.55122	-2.03201	-0.80586	-0.27287	-2.17689	1.11644	0.16191	-0.35375	0.07548	-0.02785
22	10.4187	-1.8985	0.9701	-2.56450	0.51868	-0.11982	-1.01857	1.55046	1.37133	1.58731	-1.96990	0.13855	0.67158	-0.14502	-0.34827
23	10.6044	-2.7566	1.3076	-1.36303	-0.29678	1.77218	-0.37978	0.74967	-0.55009	0.48427	-1.98844	0.23750	0.37081	0.46210	-0.35681
24	9.8197	-2.3259	2.8235	-2.01274	0.92542	0.14548	-2.29225	2.04696	-0.03054	1.96121	-1.57046	-0.86977	0.60216	0.35669	-0.07109
25	-4.7128	-1.1959	2.6621	0.41022	0.48877	2.13173	-0.47629	0.33261	-0.35916	0.67360	-0.65573	-0.33718	1.88066	-0.51519	0.91857
26	-3.0377	0.7447	2.9647	0.63370	1.23168	1.85156	-0.73092	0.70737	-0.39929	-1.11224	0.07520	0.52175	0.05157	-0.67990	-0.69854
27	1.2640	-4.4574	0.7550	3.17409	-1.42208	0.99805	-1.03176	-2.44282	-1.40338	-0.03470	1.40234	0.65586	1.97748	-0.01363	-0.23997
28	-13.2762	-1.8724	-0.4129	-4.87070	1.61547	-2.02080	0.81868	2.13505	0.16962	0.68672	0.94021	0.29578	0.56985	0.11825	0.63224
29	-15.0505	-0.2647	-2.5238	-6.49352	0.96811	-1.36529	0.68245	3.64349	-4.05187	-2.00102	-0.23019	0.61253	1.07058	-0.03502	-0.86611
30	-6.7359	-4.1335	2.5080	-2.76280	-2.00590	-0.08252	0.95344	-0.58452	1.10171	1.93774	1.19759	0.41492	-0.85676	2.03986	-0.63709

이하 중략

OBS	id	Prin1	Prin2	Prin3	Prin4	total
1	39	13.1802	3.8946	2.3557	0.11512	19.5456
2	40	13.8044	3.6503	1.8667	-0.17950	19.1419
3	8	11.0723	4.4857	0.0833	0.09941	15.7408
4	7	9.5935	4.9223	0.3070	-0.09617	14.7266
5	9	7.5936	4.6013	1.4609	-0.51860	13.1372
6	5	1.4804	3.4916	3.2469	2.45018	10.6691
7	24	9.8197	-2.3259	2.8235	-2.01274	8.3045
8	16	6.8485	2.7686	-1.0857	-0.41740	8.1139
9	23	10.6044	-2.7566	1.3076	-1.36303	7.7923
10	2	10.1416	0.4179	0.0874	-3.08784	7.5591

<= 종합점수 순위에 따른 배열

주성분 분석 (PCA)

- 주성분 이름 붙이기
  - 공분산행렬 or 상관행렬로 부터 고유치, 고유벡터 계산
  - 고유벡터를 계수로하여 주성분 점수 계산.
  - 주성분 부하(선형계수)를 이용하여 이름 부여
  - 예제 – 지원자 예제에서..

	Prin1	Prin2	Prin3	Prin4
l	0.149129	0.371461	0.200481	-.277311
ap	0.132250	-.029296	0.041918	0.134231
aa	0.029611	0.101846	-.131030	0.603168
li	0.203126	-.093042	0.619733	0.126399
sc	0.231436	-.235740	-.169273	-.072088
lc	0.336870	-.195978	-.124714	0.052788
ho	0.[illegible]	-.300549	0.447176	0.255587
sm	0.379017	-.090010	-.281561	-.172303
ex	0.164016	0.636212	0.025043	0.166245
dr	0.316050	0.012466	-.113315	-.134844
am	0.312106	-.122150	-.244517	-.147307
gc	0.338764	-.074347	-.050497	0.206271
po	0.357165	-.024920	0.041308	0.317232
kj	0.226076	-.044827	0.385206	-.459715
su	0.274483	0.420867	0.016815	-.015962

- 제 1주성분의 계수의 크기에 의해..
  - LC, SM, DR, AM, GC, PO의 부하량이 크므로
  => 정신적 & 지적 능력
- 제 2주성분 : 경험 주성분
- 제 3주성분 : 심성변수
- 제 4주성분 : 학교성적