

# Componentes Principales 2

Andrea Piñeiro

2022-10-18

## Componentes Principales 2

```
D = read.csv("países_mundo.csv")
D
```

##	CrecPobl	MortInf	PorcMujeres	PNB95	ProdElec	LinTelf	ConsAgua	PropBosq
## 1	1.0	30	41	2199	3903	12	94	53
## 2	3.0	124	46	4422	955	6	57	19
## 3	4.3	21	13	133540	91019	96	497	1
## 4	2.5	34	24	44609	19883	42	180	2
## 5	1.3	22	31	278431	65962	160	1043	22
## 6	1.4	6	43	337909	167155	510	933	19
## 7	0.6	6	41	216547	53259	465	304	47
## 8	2.0	79	42	28599	9891	2	220	6
## 9	0.3	8	40	250710	72236	457	917	20
## 10	3.0	95	48	2034	6	5	26	45
## 11	0.4	13	49	21356	31397	190	295	31
## 12	2.3	69	37	5905	2824	35	201	45
## 13	1.6	44	35	579787	260682	75	246	66
## 14	-0.6	15	48	11225	381333	335	1544	33
## 15	2.9	56	38	8615	2740	4	38	44
## 16	1.3	6	45	573695	554227	590	1602	49
## 17	1.8	26	37	70263	43354	100	174	52
## 18	3.1	90	43	1784	435	8	20	58
## 19	1.8	26	45	12870	38000	47	687	74
## 20	0.9	10	40	435137	164993	415	632	66
## 21	3.4	86	33	9248	2305	8	66	34
## 22	2.5	13	30	8884	4772	164	780	28
## 23	0.9	9	38	7150	10982	32	870	16
## 24	1.6	12	32	59151	25276	132	1626	12
## 25	1.3	34	45	744890	928083	34	461	13
## 26	0.2	6	46	156027	40097	613	233	12
## 27	2.0	37	29	11390	6182	79	446	22
## 28	2.3	36	26	15997	8256	61	581	43
## 29	2.2	56	29	45507	51947	46	956	0
## 30	1.8	36	34	9057	3211	53	245	6
## 31	5.8	16	13	42806	18870	283	884	0
## 32	0.3	11	48	15848	24740	208	337	38
## 33	0.2	7	36	532347	161654	385	781	51
## 34	2.6	112	41	5722	1293	2	51	13

## 35	2.3	39	37	71865	27062	21	686	26
## 36	0.4	5	48	105174	65546	550	440	77
## 37	0.5	6	44	1451051	476200	558	665	25
## 38	2.9	89	44	3759	933	30	57	71
## 39	3.0	73	51	6719	6115	4	35	42
## 40	0.5	8	36	85885	40623	493	523	47
## 41	2.9	44	26	14255	3161	27	139	39
## 42	2.0	72	43	1777	362	8	7	1
## 43	0.6	6	40	371039	79647	525	518	10
## 44	3.0	45	30	3566	2672	29	294	41
## 45	-0.3	11	44	42129	33486	185	661	18
## 46	1.9	68	32	319660	386500	13	612	17
## 47	1.7	51	40	190105	53414	17	96	60
## 48	2.7	108	18	24600	27060	33	4575	4
## 49	3.2	45	24	113400	79128	79	1362	11
## 50	0.1	6	33	52765	17105	365	233	6
## 51	1.1	4	44	6686	4780	555	636	1
## 52	2.7	8	40	87875	32781	418	408	6
## 53	4.7	31	21	6354	5076	73	173	1
## 54	2.9	58	46	7583	3539	9	87	2
## 55	-0.3	11	28	28941	22798	230	525	0
## 56	2.3	32	28	10673	5184	82	271	8
## 57	3.6	61	21	23400	17800	59	880	0
## 58	2.5	12	37	78321	39093	166	768	54
## 59	2.0	55	35	29545	11100	43	427	20
## 60	2.1	33	31	304596	147926	96	899	25
## 61	1.8	113	48	1353	490	3	55	22
## 62	1.8	83	43	35840	3500	3	101	44
## 63	2.5	91	40	4391	927	4	150	37
## 64	3.1	46	36	1659	1688	23	367	50
## 65	2.9	80	36	28411	15530	4	41	17
## 66	0.5	5	46	136077	113488	556	488	31
## 67	1.0	7	44	51655	35135	479	589	28
## 68	4.5	18	15	10578	6187	77	564	19
## 69	3.0	90	26	59991	58529	16	2053	2
## 70	1.9	23	34	7253	3380	114	754	42
## 71	2.7	41	29	8158	36415	31	109	32
## 72	2.1	47	29	55019	15563	47	300	53
## 73	0.4	14	46	107829	135347	148	321	28
## 74	-0.1	7	43	96829	31380	361	739	34
## 75	0.3	6	43	1094734	325383	502	205	10
## 76	0.0	8	47	39990	58705	236	266	34
## 77	0.0	23	44	33488	55136	131	1134	27
## 78	2.8	62	42	5070	1002	10	202	39
## 79	1.8	4	38	79831	20046	478	84	7
## 80	3.1	32	26	15780	15186	63	435	4
## 81	1.3	16	35	12616	4387	11	503	27
## 82	2.2	77	28	7510	1333	3	633	18
## 83	0.6	4	48	209720	142895	681	341	68
## 84	0.8	6	40	286014	65724	613	173	30
## 85	2.3	50	37	130918	189316	95	359	4
## 86	1.3	35	46	159630	71177	59	602	25
## 87	3.1	82	49	3703	1913	3	40	38
## 88	2.1	39	30	16369	6714	58	381	4

## 89	1.9	48	35	169452	78322	212	585	26
## 90	0.1	15	49	84084	202995	157	673	16
## 91	0.6	18	40	16458	7617	196	241	4
## 92	2.4	23	33	65382	73116	111	382	52
## 93	2.2	41	49	17634	12270	11	414	26
## 94	4.2	100	29	4044	2159	12	335	8
## 95	2.6	109	45	3605	7785	8	186	43
## 96	2.8	55	44	5933	7334	14	136	23
##	PropDeFor	ConsEner	EmisCO2					
## 1	0.0	341	1.2					
## 2	0.7	89	0.5					
## 3	0.0	4566	13.1					
## 4	0.8	906	3.0					
## 5	0.1	1504	3.5					
## 6	0.0	5341	15.3					
## 7	-0.4	3301	7.2					
## 8	4.1	64	0.2					
## 9	-0.3	5120	10.1					
## 10	1.3	20	0.1					
## 11	-0.4	2392	9.9					
## 12	1.2	373	1.0					
## 13	0.6	718	1.4					
## 14	-0.2	2438	6.4					
## 15	0.6	103	0.2					
## 16	-1.1	7854	14.4					
## 17	0.7	622	1.8					
## 18	0.2	331	1.6					
## 19	0.0	1129	11.2					
## 20	0.1	2982	6.6					
## 21	1.0	103	0.5					
## 22	3.0	558	1.2					
## 23	1.0	923	2.6					
## 24	-0.1	1012	2.6					
## 25	0.7	664	2.3					
## 26	0.0	3977	10.4					
## 27	2.9	337	1.4					
## 28	1.8	565	1.8					
## 29	0.0	600	1.5					
## 30	2.3	370	0.7					
## 31	0.0	10531	33.9					
## 32	0.1	3243	7.0					
## 33	0.0	2458	5.7					
## 34	0.3	22	0.1					
## 35	3.4	316	0.8					
## 36	0.0	5997	8.2					
## 37	-0.1	4042	6.3					
## 38	0.6	652	5.5					
## 39	1.4	93	0.2					
## 40	0.0	2260	7.2					
## 41	1.8	210	0.6					
## 42	5.1	29	0.1					
## 43	-0.3	4580	9.2					
## 44	2.2	204	0.6					
## 45	-0.5	2383	5.8					

## 46	0.6	248	0.9
## 47	1.1	366	1.0
## 48	0.1	1213	3.4
## 49	0.0	1505	4.0
## 50	-1.2	3137	8.7
## 51	0.0	7932	6.8
## 52	-0.3	2717	8.1
## 53	-1.0	1067	3.0
## 54	0.6	110	0.2
## 55	0.0	8622	11.2
## 56	0.6	964	2.9
## 57	-1.4	2499	8.1
## 58	2.1	1699	3.8
## 59	-1.4	327	1.1
## 60	1.3	1561	3.8
## 61	0.8	40	0.1
## 62	1.3	49	0.1
## 63	1.0	28	0.1
## 64	1.9	300	0.6
## 65	0.7	162	0.9
## 66	-1.4	5318	14.1
## 67	0.0	4245	7.6
## 68	0.0	2392	5.3
## 69	3.5	254	0.6
## 70	1.9	618	1.7
## 71	2.8	299	0.6
## 72	0.4	367	1.0
## 73	-0.1	2401	8.9
## 74	-0.5	1827	4.8
## 75	-1.1	3732	9.8
## 76	0.0	3868	13.1
## 77	0.0	1733	5.4
## 78	0.7	97	0.4
## 79	2.3	8103	17.7
## 80	-4.3	997	3.3
## 81	1.4	97	0.3
## 82	1.1	66	0.1
## 83	0.0	5723	6.6
## 84	-0.6	3629	6.4
## 85	-0.8	2146	7.5
## 86	3.5	769	2.0
## 87	1.2	34	0.1
## 88	-1.9	595	1.6
## 89	0.0	957	2.5
## 90	-0.3	3180	11.7
## 91	-0.6	629	1.6
## 92	1.2	2186	5.7
## 93	1.5	101	0.3
## 94	0.0	206	0.7
## 95	1.1	149	0.3
## 96	0.7	438	1.8

## Parte I

1. Calcule las matrices de varianza-covarianza  $S$  con  $\text{cov}(X)$  y la matriz de correlaciones con  $\text{cor}(X)$

```
S = cov(D)
cat('Matriz de varianza-covarianza\n')
```

```
## Matriz de varianza-covarianza
```

```
S
```

```
##          CrecPobl      MortInf  PorcMujeres      PNB95
## CrecPobl    1.538298e+00  2.195026e+01 -6.078026e+00 -8.933379e+04
## MortInf     2.195026e+01  1.032859e+03 -9.249342e+00 -2.269332e+06
## PorcMujeres -6.078026e+00 -9.249342e+00  7.698322e+01  2.813114e+05
## PNB95       -8.933379e+04 -2.269332e+06  2.813114e+05  4.999786e+10
## ProdElec   -4.973964e+04 -1.043435e+06  2.260248e+05  2.247791e+10
## LinTelf    -1.369079e+02 -4.381366e+03  4.499750e+02  2.039550e+07
## ConsAgua   -4.827092e+01 -1.288211e+03 -1.568313e+03  1.097481e+07
## PropBosq   -3.887018e+00 -1.466316e+01  6.517895e+01  2.474311e+05
## PropDefor   3.361974e-01  1.276296e+01  2.680592e-01 -5.806203e+04
## ConsEner   -8.384169e+02 -4.442568e+04  2.855207e+02  1.415628e+08
## EmisC02    -1.137877e+00 -9.485500e+01 -2.150132e+00  2.501673e+05
##          ProdElec      LinTelf      ConsAgua      PropBosq
## CrecPobl   -4.973964e+04 -1.369079e+02 -4.827092e+01 -3.887018
## MortInf    -1.043435e+06 -4.381366e+03 -1.288211e+03 -14.663158
## PorcMujeres 2.260248e+05  4.499750e+02 -1.568313e+03  65.178947
## PNB95      2.247791e+10  2.039550e+07  1.097481e+07 247431.122807
## ProdElec   1.821909e+10  7.583050e+06  1.399817e+07 70359.785965
## LinTelf     7.583050e+06  3.841247e+04  1.193110e+04 248.715789
## ConsAgua    1.399817e+07  1.193110e+04  3.301981e+05 -2220.757895
## PropBosq    7.035979e+04  2.487158e+02 -2.220758e+03 401.003509
## PropDefor  -3.180340e+04 -9.940461e+01 -6.743793e+01  2.625263
## ConsEner    6.801296e+07  3.426262e+05  2.092242e+05 -5153.438596
## EmisC02     1.392779e+05  6.385700e+02  4.869328e+02 -12.897193
##          PropDefor      ConsEner      EmisC02
## CrecPobl    3.361974e-01 -8.384169e+02 -1.137877
## MortInf     1.276296e+01 -4.442568e+04 -94.855000
## PorcMujeres 2.680592e-01  2.855207e+02 -2.150132
## PNB95       -5.806203e+04  1.415628e+08 250167.323509
## ProdElec   -3.180340e+04  6.801296e+07 139277.888640
## LinTelf    -9.940461e+01  3.426262e+05  638.570000
## ConsAgua   -6.743793e+01  2.092242e+05  486.932763
## PropBosq    2.625263e+00 -5.153439e+03 -12.897193
## PropDefor   1.817253e+00 -1.051522e+03 -2.632487
## ConsEner   -1.051522e+03  5.014395e+06 10286.159781
## EmisC02    -2.632487e+00  1.028616e+04  27.268614
```

```
corD = cor(D)
cat('\n\nMatriz de correlaciones\n')
```

```
##
##
## Matriz de correlaciones
```

```
corD
```

```
##          CrecPobl      MortInf PorcMujeres      PNB95      ProdElec
## CrecPobl      1.00000000  0.55067948 -0.55852711 -0.32212154 -0.29711119
## MortInf       0.55067948  1.00000000 -0.03280139 -0.31579250 -0.24053689
## PorcMujeres  -0.55852711 -0.03280139  1.00000000  0.14338826  0.19085114
## PNB95         -0.32212154 -0.31579250  0.14338826  1.00000000  0.74476081
## ProdElec     -0.29711119 -0.24053689  0.19085114  0.74476081  1.00000000
## LinTelf      -0.56321228 -0.69558922  0.26167018  0.46539599  0.28664508
## ConsAgua     -0.06772953 -0.06975563 -0.31106243  0.08541500  0.18047653
## PropBosq     -0.15650281 -0.02278415  0.37096694  0.05525919  0.02603078
## PropDefor     0.20107881  0.29459348  0.02266339 -0.19262327 -0.17478434
## ConsEner     -0.30187731 -0.61731132  0.01453216  0.28272492  0.22501894
## EmisCO2      -0.17568860 -0.56520778 -0.04692837  0.21425123  0.19760017
##          LinTelf      ConsAgua      PropBosq      PropDefor      ConsEner
## CrecPobl     -0.56321228 -0.06772953 -0.15650281  0.20107881 -0.30187731
## MortInf     -0.69558922 -0.06975563 -0.02278415  0.29459348 -0.61731132
## PorcMujeres  0.26167018 -0.31106243  0.37096694  0.02266339  0.01453216
## PNB95        0.46539599  0.08541500  0.05525919 -0.19262327  0.28272492
## ProdElec     0.28664508  0.18047653  0.02603078 -0.17478434  0.22501894
## LinTelf      1.00000000  0.10593934  0.06337138 -0.37623801  0.78068385
## ConsAgua     0.10593934  1.00000000 -0.19299225 -0.08705811  0.16259804
## PropBosq     0.06337138 -0.19299225  1.00000000  0.09725032 -0.11492480
## PropDefor    -0.37623801 -0.08705811  0.09725032  1.00000000 -0.34833836
## ConsEner     0.78068385  0.16259804 -0.11492480 -0.34833836  1.00000000
## EmisCO2      0.62393719  0.16227447 -0.12333592 -0.37396154  0.87965517
##          EmisCO2
## CrecPobl     -0.17568860
## MortInf      -0.56520778
## PorcMujeres  -0.04692837
## PNB95        0.21425123
## ProdElec     0.19760017
## LinTelf      0.62393719
## ConsAgua     0.16227447
## PropBosq     -0.12333592
## PropDefor    -0.37396154
## ConsEner     0.87965517
## EmisCO2      1.00000000
```

2. Calcule los valores y vectores propios de cada matriz. La función en R es: `eigen()`.

```
EVar = eigen(S)
eigValVar = EVar$values
eigVecVar = EVar$vectors
cat('Valores y vectores propios de la varianza-covarianza\n')
```

```
## Valores y vectores propios de la varianza-covarianza
```

EVar

```
## eigen() decomposition
## $values
## [1] 6.163576e+10 6.581612e+09 4.636256e+06 3.107232e+05 1.216015e+04
## [6] 5.137767e+02 3.627885e+02 4.542082e+01 5.800868e+00 1.438020e+00
## [11] 4.768083e-01
##
## $vectors
##           [,1]           [,2]           [,3]           [,4]           [,5]
## [1,] -1.658168e-06 4.706785e-07 0.0001263736 -1.928408e-05 -0.0055373971
## [2,] -4.048139e-05 -1.774254e-05 0.0082253821 -2.493257e-03 -0.0944030203
## [3,] 5.739096e-06 -1.084543e-05 0.0001318149 5.538307e-03 0.0314036410
## [4,] 8.880376e-01 4.597632e-01 0.0026022071 -3.893588e-04 -0.0003327409
## [5,] 4.597636e-01 -8.880405e-01 0.0005694896 1.096305e-03 0.0002207819
## [6,] 3.504341e-04 4.016179e-04 -0.0619424889 7.641174e-03 0.9921404486
## [7,] 2.625508e-04 -1.122118e-03 -0.0401453227 -9.991411e-01 0.0057795144
## [8,] 4.089564e-06 7.790843e-06 0.0012719918 6.435797e-03 0.0419331615
## [9,] -1.073825e-06 2.350808e-07 0.0001916177 4.043796e-05 -0.0018090751
## [10,] 2.547156e-03 7.126782e-04 -0.9972315499 3.973568e-02 -0.0625729475
## [11,] 4.643724e-06 -1.315731e-06 -0.0020679047 -5.626049e-05 -0.0042367120
##           [,6]           [,7]           [,8]           [,9]           [,10]
## [1,] 1.243456e-02 5.359089e-03 -8.390810e-02 -6.778358e-02 -1.158091e-01
## [2,] 9.917515e-01 2.258020e-02 -7.891128e-02 -1.637836e-02 4.264872e-04
## [3,] 8.552992e-02 -1.136481e-01 9.856498e-01 -1.468464e-02 8.241465e-03
## [4,] -8.621005e-06 -7.566477e-06 1.217248e-05 -3.971469e-07 4.274451e-07
## [5,] 1.955408e-05 1.544658e-05 -2.558998e-05 1.059471e-06 -1.353881e-06
## [6,] 9.109622e-02 4.748682e-02 -3.416812e-02 -5.379549e-03 -3.409423e-03
## [7,] -1.087229e-03 -6.863294e-03 4.698731e-03 7.965261e-05 3.621425e-05
## [8,] 1.721948e-02 -9.920538e-01 -1.169638e-01 1.416566e-03 5.891758e-03
## [9,] 1.758667e-03 -7.455427e-03 1.811443e-02 1.283039e-01 -9.859317e-01
## [10,] 2.639673e-03 -3.764707e-03 1.267052e-03 2.262931e-03 2.672618e-04
## [11,] -1.877994e-02 -1.709137e-03 -5.204823e-03 -9.891529e-01 -1.200519e-01
##           [,11]
## [1,] 9.872887e-01
## [2,] -2.092491e-02
## [3,] 8.344324e-02
## [4,] 2.723996e-07
## [5,] -2.086857e-07
## [6,] 4.944397e-04
## [7,] 4.780416e-04
## [8,] -3.748976e-03
## [9,] -1.052934e-01
## [10,] 5.906241e-05
## [11,] -8.221371e-02
```

```
ECor = eigen(corD)
eigValCor = ECor$values
eigVecCor = ECor$vectors
cat('\n\nValores y vectores propios de la correlación\n')
```

```
##
##
```

```
## Valores y vectores propios de la correlación
```

```
ECor
```

```
## eigen() decomposition
## $values
## [1] 4.02987902 1.92999195 1.37041115 0.86451597 0.79414057 0.72919997
## [7] 0.57130511 0.32680096 0.16806846 0.14632819 0.06935866
##
## $vectors
##          [,1]      [,2]      [,3]      [,4]      [,5]      [,6]
## [1,] -0.314119414  0.34835747 -0.07352541  0.44028717  0.32972147 -0.18392437
## [2,] -0.392395442 -0.04136238 -0.17759254  0.13398483 -0.08340489 -0.08656390
## [3,]  0.116546319 -0.58283641  0.16686305 -0.05865031 -0.18654100  0.16835650
## [4,]  0.295393771 -0.17690839 -0.53343025  0.26248209  0.14110658  0.04653378
## [5,]  0.258964724 -0.17356372 -0.61438847  0.17389644  0.07521971  0.02821905
## [6,]  0.446082934 -0.02719077  0.15177250 -0.04959796  0.05416498  0.02442175
## [7,]  0.092410503  0.32060987 -0.37024258 -0.73603097 -0.02671021 -0.30940890
## [8,]  0.005692925 -0.45742697  0.16480339 -0.04024882  0.41531702 -0.75356463
## [9,] -0.243652293 -0.15408201 -0.02961449 -0.33650345  0.73261463  0.50894232
## [10,] 0.415029554  0.23286257  0.20608749  0.06730166  0.23100421  0.05806466
## [11,] 0.374531032  0.29168698  0.20631751  0.14843513  0.24028756 -0.02809233
##
##          [,7]      [,8]      [,9]      [,10]      [,11]
## [1,]  0.1628974320 -0.09481963  0.52181220  0.34674573 -0.10062784
## [2,]  0.6398040762 -0.32307802 -0.29031618 -0.38959240  0.17487096
## [3,]  0.5310867107  0.05209889  0.23599758  0.42854658 -0.16786800
## [4,] -0.1490207046 -0.44913216 -0.36995675  0.34911534 -0.15247432
## [5,]  0.1082745817  0.50343911  0.30681318 -0.33770404  0.12366382
## [6,] -0.0008501608 -0.56975094  0.44733110 -0.20997673  0.44992596
## [7,]  0.2357666690 -0.05962470  0.08358225  0.20561803 -0.07067780
## [8,] -0.0806036686  0.04275404 -0.07438520 -0.08671232 -0.01493710
## [9,]  0.0112333588 -0.01607505 -0.01868615 -0.03209758  0.07259619
## [10,] 0.2711228006 -0.05023582 -0.04339752 -0.36147417 -0.67912543
## [11,] 0.3352822144  0.30978009 -0.37666244  0.28779437  0.46737561
```

3. Calcule la proporción de varianza explicada por cada componente. Se sugiere dividir cada lambda entre la varianza total (las lambdas están en `eigen(S)[1]`). La varianza total es la suma de las varianzas de la diagonal de S. Una forma es `sum(diag(S))`.

```
propVar = eigValVar / sum(diag(S))
cat('Proporción de varianza explicada de la matriz de varianza-covarianza\n')
```

```
## Proporción de varianza explicada de la matriz de varianza-covarianza
```

```
propVar
```

```
## [1] 9.034543e-01 9.647298e-02 6.795804e-05 4.554567e-06 1.782429e-07
## [6] 7.530917e-09 5.317738e-09 6.657763e-10 8.502887e-11 2.107843e-11
## [11] 6.989035e-12
```



```
propCor = eigValCor / sum(diag(corD))
cat('\n\nProporción de varianza explicada de la matriz de correlación\n')
```

```
##
##
## Proporción de varianza explicada de la matriz de correlación
```

```
propCor
```

```
## [1] 0.366352638 0.175453813 0.124582832 0.078592361 0.072194597 0.066290906
## [7] 0.051936828 0.029709178 0.015278951 0.013302563 0.006305332
```

#### 4. Acumule los resultados anteriores.

```
resVar = cumsum(propVar)
cat('Acumulativo de proporción de varianza para la matriz de varianza-covarianza\n')
```

```
## Acumulativo de proporción de varianza para la matriz de varianza-covarianza
```

```
resVar
```

```
## [1] 0.9034543 0.9999273 0.9999953 0.9999998 1.0000000 1.0000000 1.0000000
## [8] 1.0000000 1.0000000 1.0000000 1.0000000
```

```
resCor = cumsum(propCor)
cat('\n\nAcumulativo de proporción de varianza para la matriz correlación\n')
```

```
##
##
## Acumulativo de proporción de varianza para la matriz correlación
```

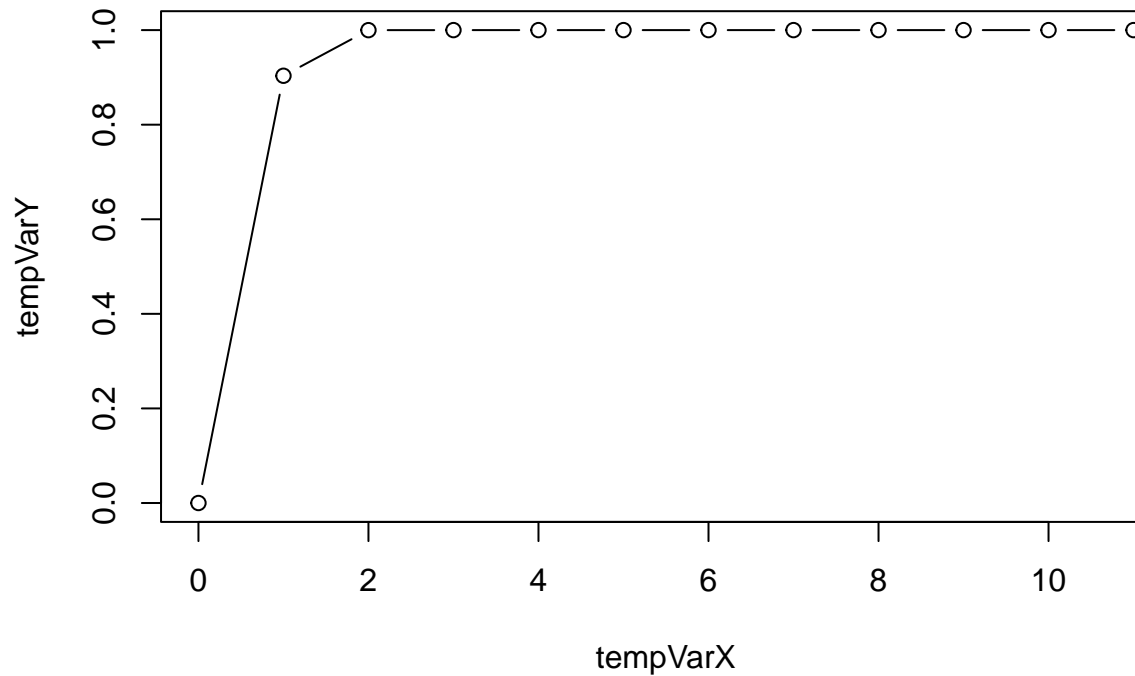
```
resCor
```

```
## [1] 0.3663526 0.5418065 0.6663893 0.7449816 0.8171762 0.8834671 0.9354040
## [8] 0.9651132 0.9803921 0.9936947 1.0000000
```

5 y 6. Según los resultados anteriores, ¿qué componentes son los más importantes? ¿qué variables son las que más contribuyen a la primera y segunda componentes principales? ¿por qué lo dice? ¿influyen las unidades de las variables?

```
tempVarY = c(0, resVar)
tempVarX = seq(0, 11)
```

```
plot(tempVarX,
     tempVarY,
     type = "b")
```



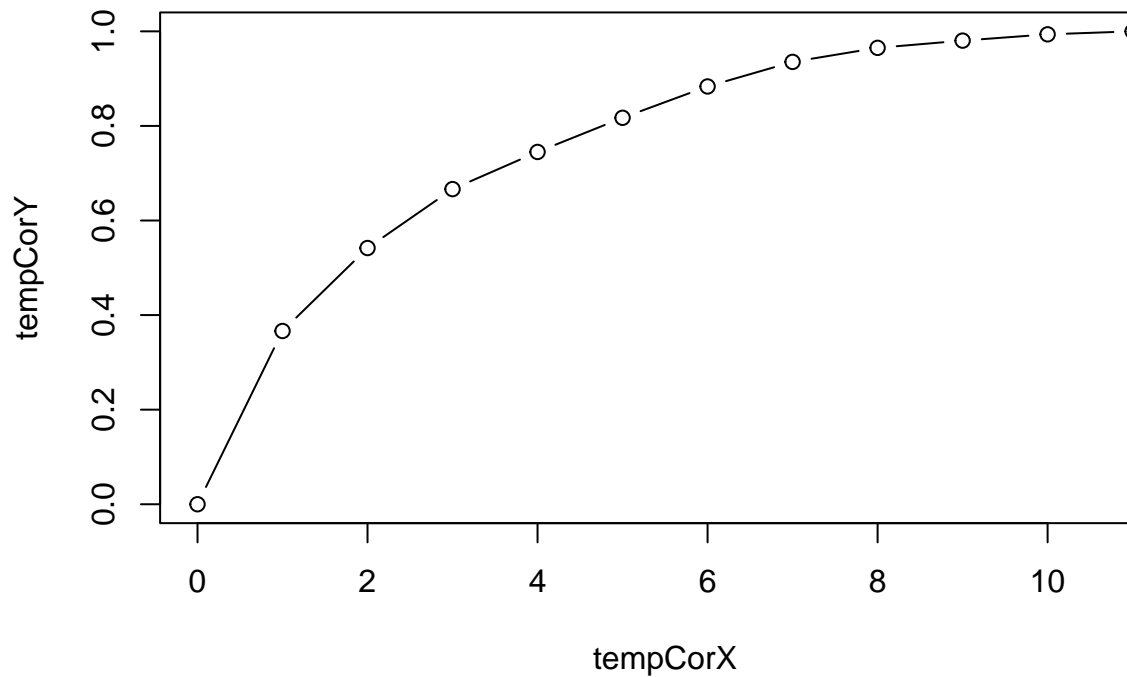
### Matriz de Covarianza

Como podemos observar anteriormente los componentes principales de la covarianza de los datos son el 1 y 2. El 1 explica el 90% de los datos, y con el 2 componente se explica el 99% de los datos, por lo que estos serían los componentes más importantes.

De igual manera como podemos ver con los vectores propios del componente 1 y 2, las variables que más influyen son la 4 y la 5, en dicho orden para ambos componentes. Si influirían las unidades debido a que las variables 4 y 5 son las que tienen unidades más grandes, por lo que son las que dominan. Para obtener las variables sin que afecten las unidades, sería necesario escalar los datos para que todos estén en el mismo rango.

```
tempCorY = c(0, resCor)
tempCorX = seq(0, 11)
```

```
plot(tempCorX,
     tempCorY,
     type = "b")
```



### Matriz de Correlación

Como podemos observar para esta matriz los componentes 1 y 2 también son los más importantes, sin embargo tienen menor relevancia, ya que con ellos se explica el 54% de los datos.

Para el componente principal 1, las variables que más influyen son la 6 y 10. Mientras que para el componente principal 2, las variables que más influyen son la 3 y 8 pero con signo negativo, por lo que mientras menores seán estas variables, mayor será el componente principal. En este caso no influyen las unidades, pues la covarianza esta dada en un rango de 0 a 1, por lo que todas tienen las mismas unidades.

**7. Compare los resultados de los incisos 6 y 7. ¿qué concluye?** Obtenemos diferentes resultados para los incisos. En la matriz de covarianzas si influyen las unidades debido a que las escalas entre variables son muy diferentes. Al contrario, en la matriz de correlación todas las variables están en el mismo rango, por lo que podemos tener mayor confianza en que estas variables contribuyen de mayor manera a los componentes principales y no esta sesgado por las unidades.

Sin embargo, con la correlación tenemos un menor porcentaje de explicación de datos de solo 54% a diferencia del 99%.

## Parte II

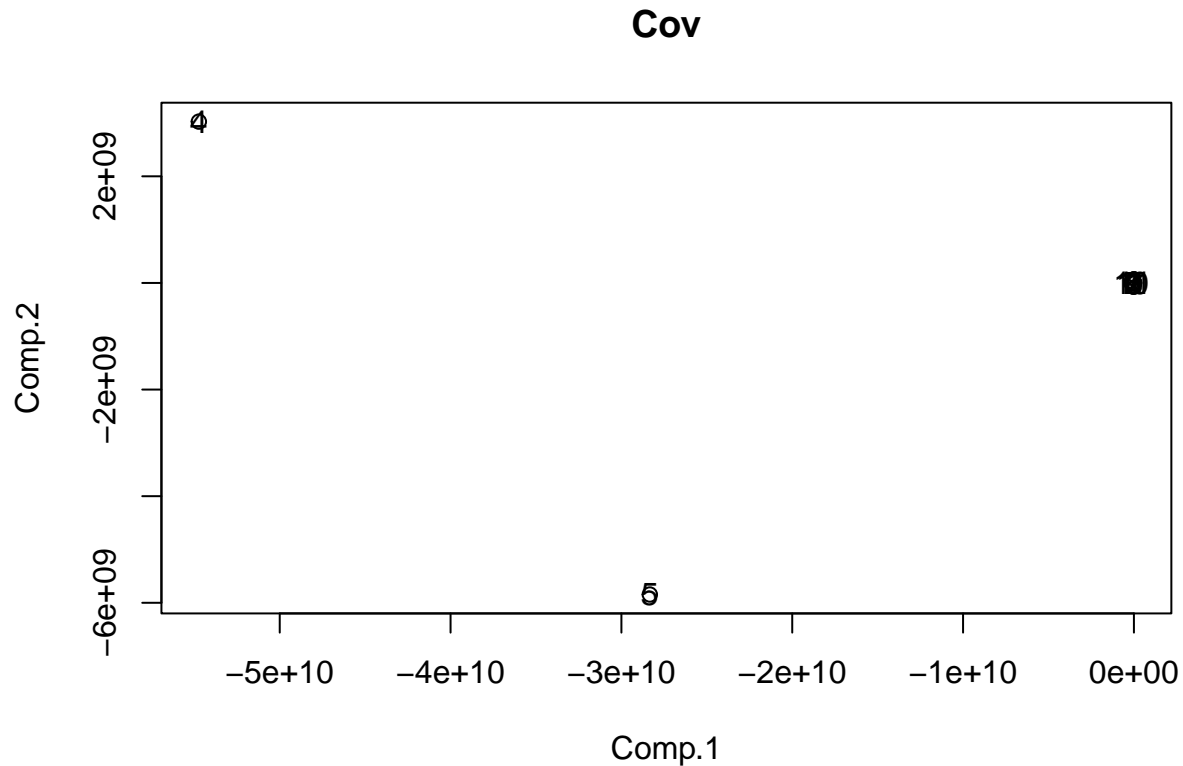
### Matriz de varianzas-covarianzas

```
library(stats)
library(factoextra)
```

```
## Loading required package: ggplot2
```

## Welcome! Want to learn more? See two factoextra-related books at <https://goo.gl/ve3WBa>

```
library(ggplot2)
cpSCov = princomp(D, cor=FALSE)
cpaSCov = as.matrix(S) %*% cpSCov$loadings
plot(cpaSCov[,1:2], type="p", main = "Cov")
text(cpaSCov[,1], cpaSCov[,2], 1:nrow(cpaSCov))
```



```
biplot(cpSCov)
```

```
## Warning in arrows(0, 0, y[, 1L] * 0.8, y[, 2L] * 0.8, col = col[2L], length =  
## arrow.len): zero-length arrow is of indeterminate angle and so skipped
```

```
## Warning in arrows(0, 0, y[, 1L] * 0.8, y[, 2L] * 0.8, col = col[2L], length =  
## arrow.len): zero-length arrow is of indeterminate angle and so skipped
```

```
## Warning in arrows(0, 0, y[, 1L] * 0.8, y[, 2L] * 0.8, col = col[2L], length =  
## arrow.len): zero-length arrow is of indeterminate angle and so skipped
```

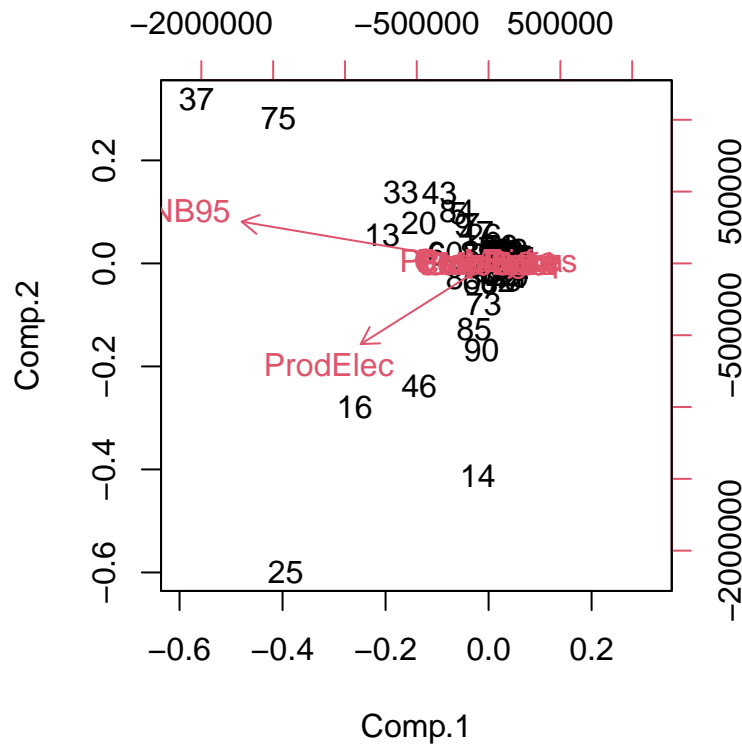
```
## Warning in arrows(0, 0, y[, 1L] * 0.8, y[, 2L] * 0.8, col = col[2L], length =  
## arrow.len): zero-length arrow is of indeterminate angle and so skipped
```

```
## Warning in arrows(0, 0, y[, 1L] * 0.8, y[, 2L] * 0.8, col = col[2L], length =  
## arrow.len): zero-length arrow is of indeterminate angle and so skipped
```

```
## Warning in arrows(0, 0, y[, 1L] * 0.8, y[, 2L] * 0.8, col = col[2L], length =
## arrow.len): zero-length arrow is of indeterminate angle and so skipped

## Warning in arrows(0, 0, y[, 1L] * 0.8, y[, 2L] * 0.8, col = col[2L], length =
## arrow.len): zero-length arrow is of indeterminate angle and so skipped

## Warning in arrows(0, 0, y[, 1L] * 0.8, y[, 2L] * 0.8, col = col[2L], length =
## arrow.len): zero-length arrow is of indeterminate angle and so skipped
```



Como podemos observar estamos obteniendo el análisis de los componentes principales en la matriz de datos con princomp. Con esta función podemos obtener los eigenvectors que se almacenan en la variable de loadings, la cual omite los loadings que sean demasiado pequeños.

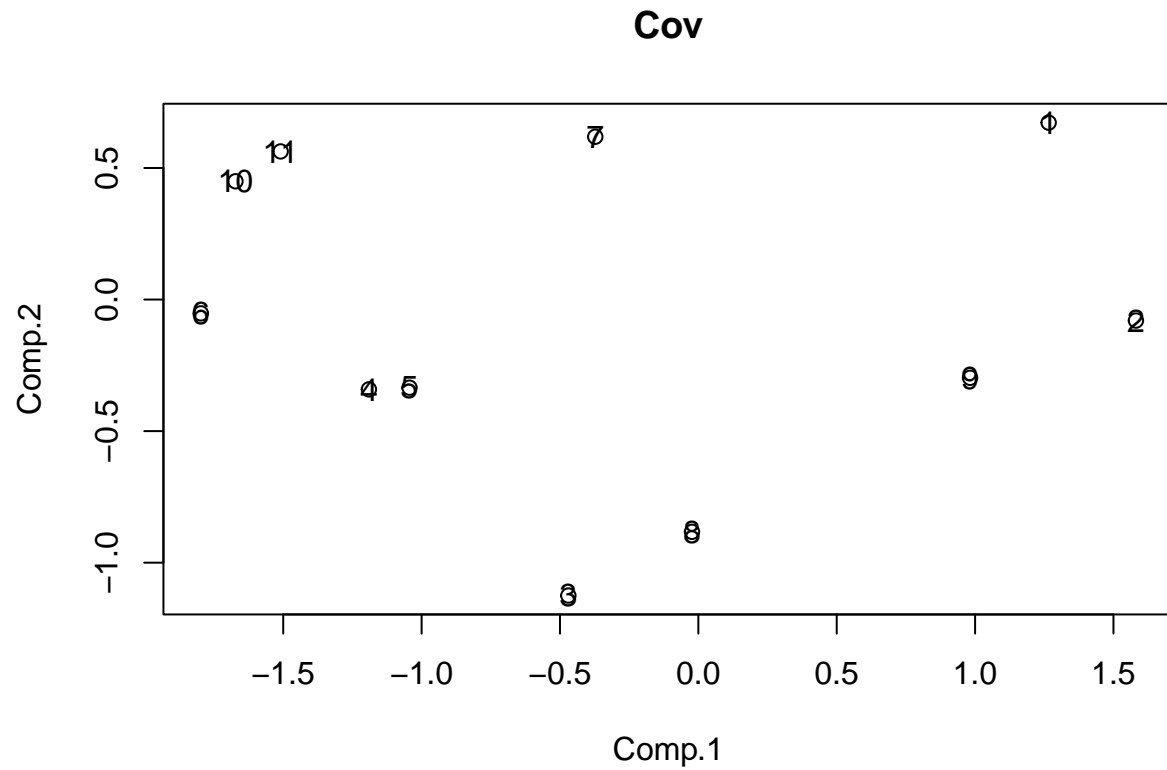
Como observamos en la primera gráfica, están muy bien divididos los resultados en 3 diferentes conjuntos, por lo que es posible que una clasificación en el dataset sea posible. Y la explicación de los datos será muy buena.

Como podemos observar en la segunda gráfica las variables que tienen mayor influencia en los componentes principales son 'NB95' y 'ProdElec', las variables número 4 y 5, pues tienen mayor variabilidad para explicar los datos. Estas son las mismas variables que obtuvimos en la parte 1 con la matriz de covarianzas.

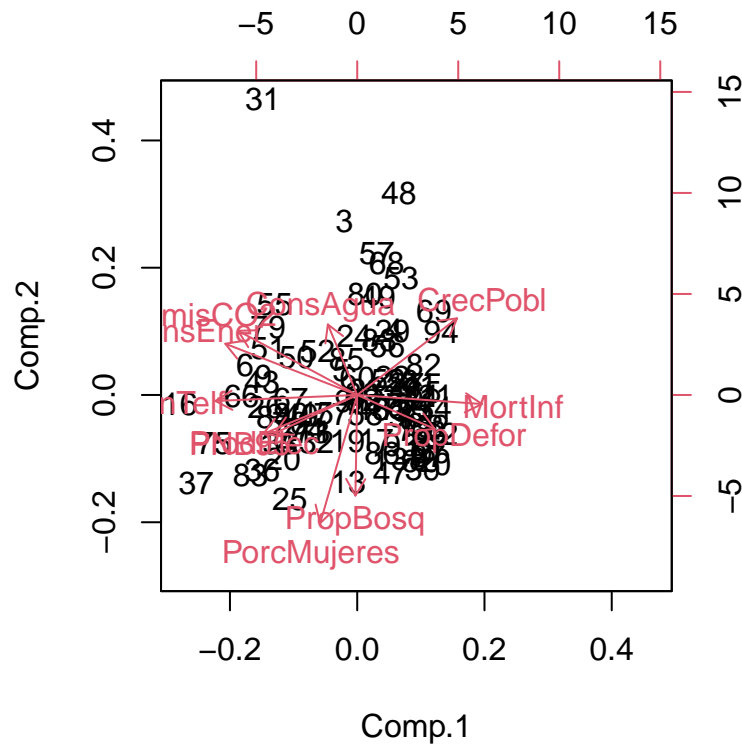
## Matriz de correlación

```
library(stats)
library(factoextra)
```

```
library(ggplot2)
cpSCor = princomp(D, cor=TRUE)
cpaSCor = as.matrix(corD) %*% cpSCor$loadings
plot(cpaSCor[,1:2], type="p", main = "Cov")
text(cpaSCor[,1], cpaSCor[,2], 1:nrow(cpaSCor))
```



```
biplot(cpSCor)
```



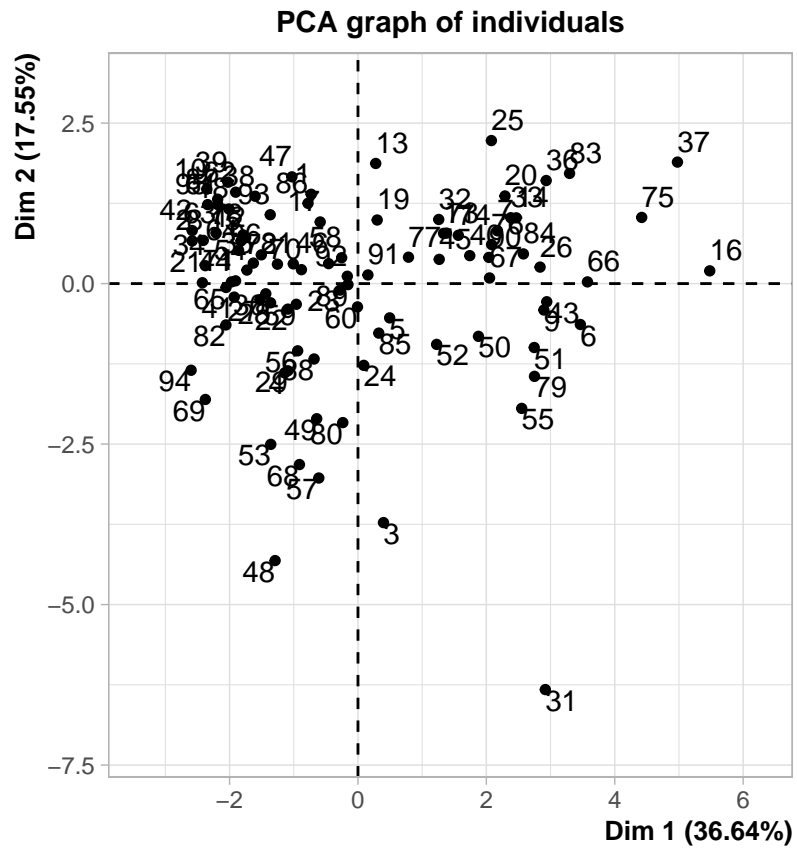
Como observamos en la primera gráfica, a diferencia de la matriz de covarianzas, tenemos mayor dispersión en los datos. En este caso no podemos ver la segmentación entre grupos de manera sencilla.

Como podemos observar en la segunda gráfica es mucho más difícil identificar cuáles son las variables que mayor efecto tienen en los componentes principales. Según lo obtenido en la parte 1 eran: \* PorcMujeres \* LinTelf \* PropBosq \* ConsEner

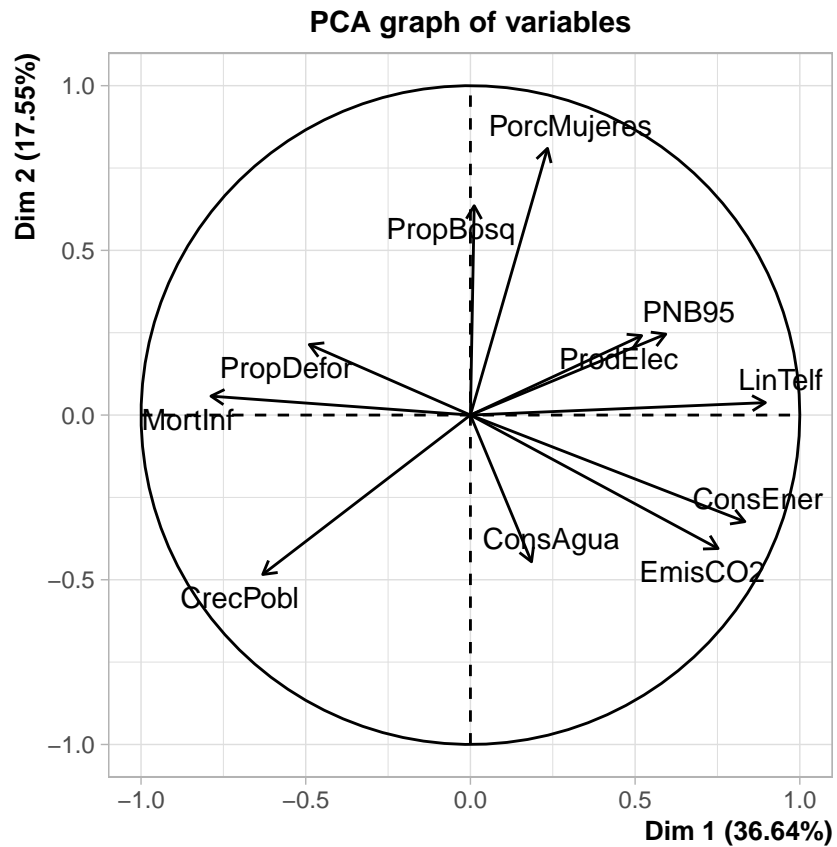
Si observamos la gráfica podemos ver que estas variables si tienen mayor variabilidad de los datos, aunque no de manera muy notoria a diferencia de la matriz de covarianzas.

### Parte III

```
library(FactoMineR)
library(factoextra)
library(ggplot2)
cp3 = PCA(D)
```



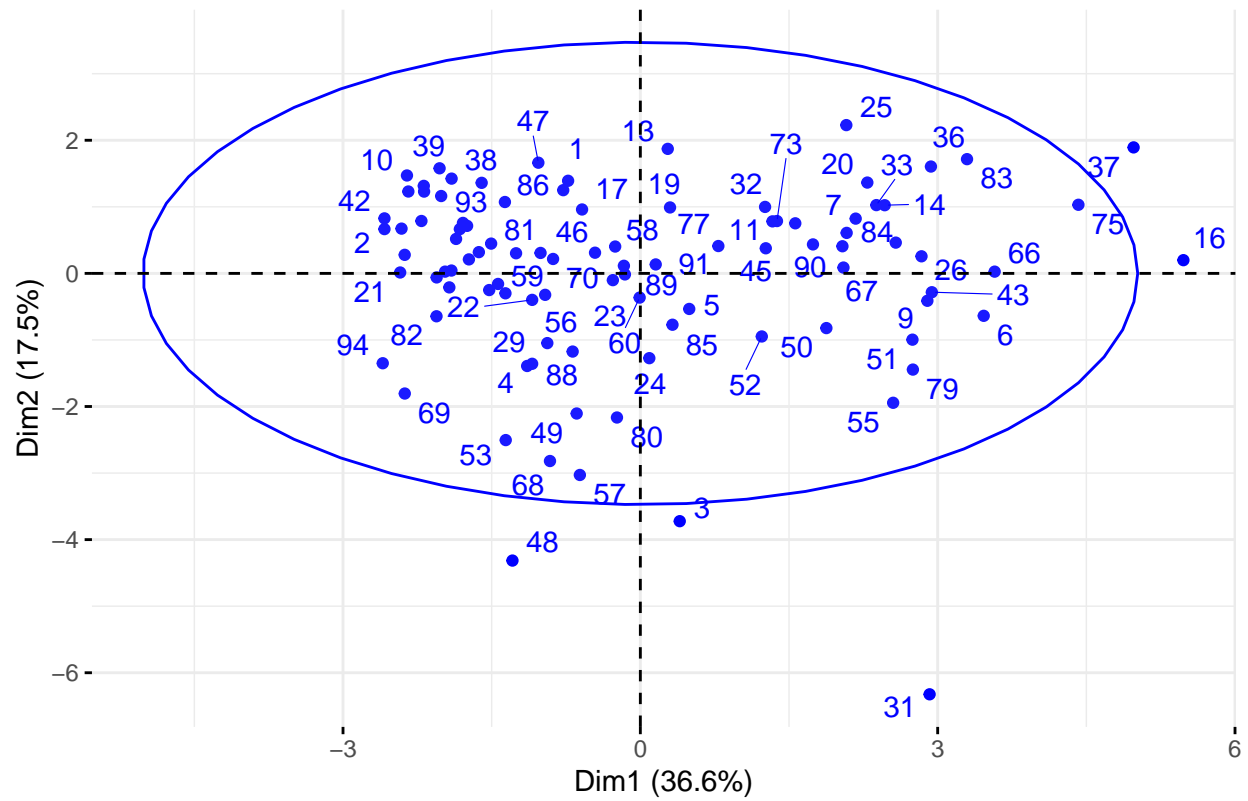




```
fviz_pca_ind(cp3, col.ind = "blue", addEllipses = TRUE, repel = TRUE)
```

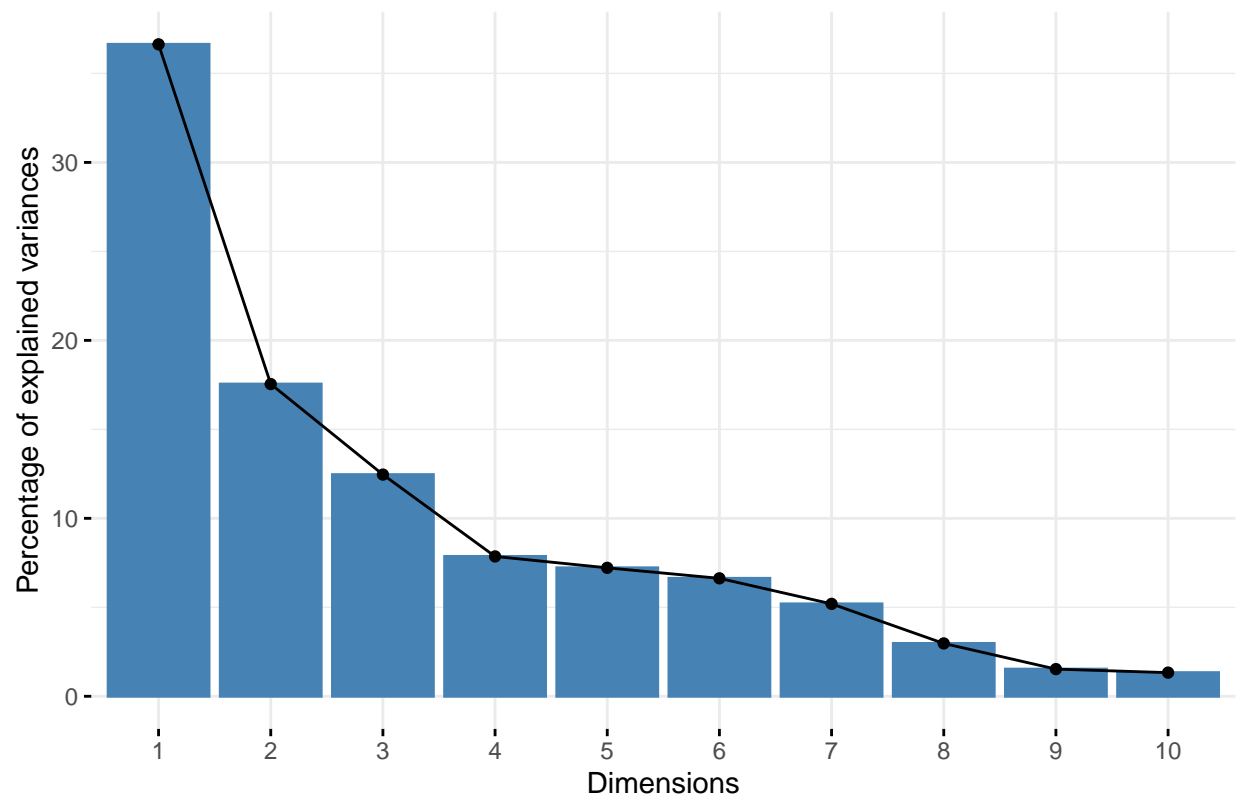
```
## Warning: ggrepel: 27 unlabeled data points (too many overlaps). Consider
## increasing max.overlaps
```

## Individuals – PCA

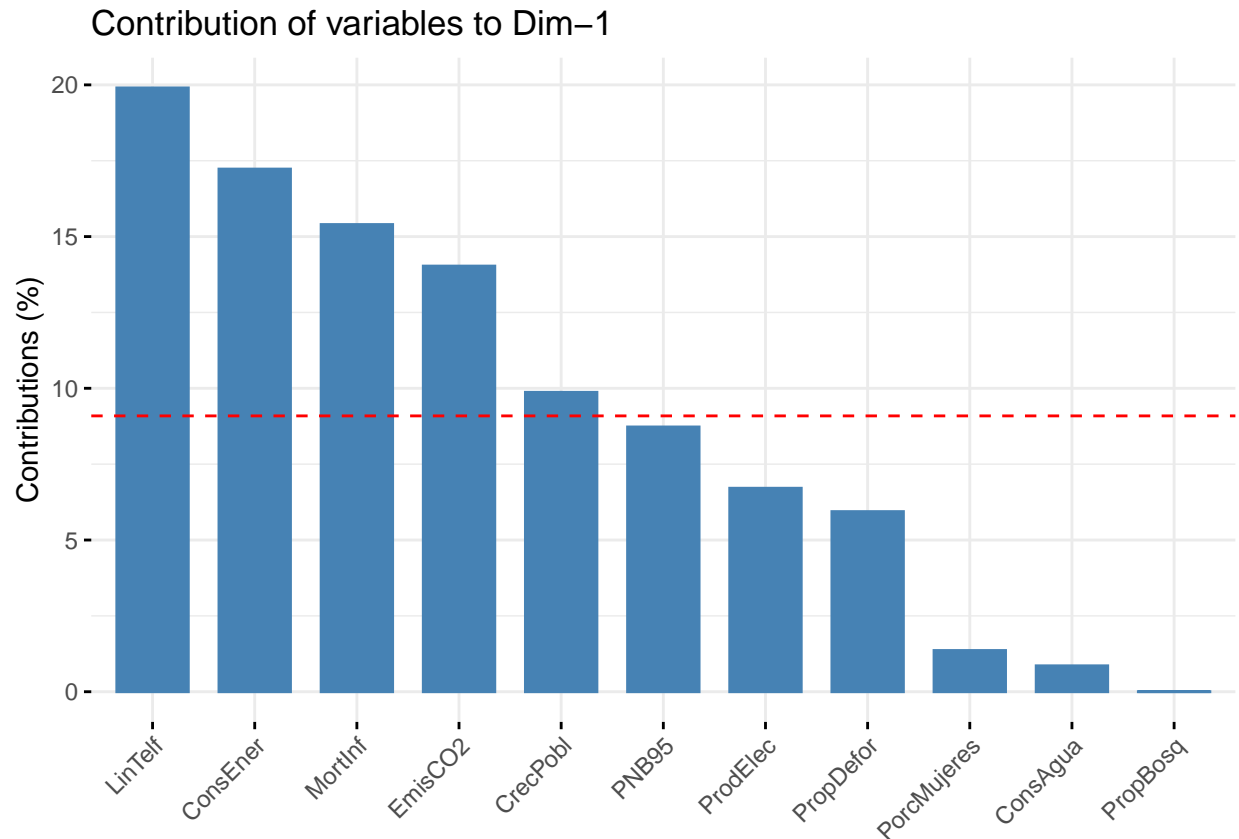


```
fviz_screepplot(cp3)
```

Scree plot



```
fviz_contrib(cp3, choice = c("var"))
```



En esta parte estamos realizando PCA (Principal Component Analysis). Con esto obtenemos un análisis mucho más completo y realizado de manera mucho más sencilla.

En la **gráfica 1** podemos observar que los datos están muy dispersos y no es fácil agruparles o segmentarles, por lo que la explicabilidad de los datos no es muy buena.

En la **gráfica 2** podemos observar que las variables que contribuyen de mayor manera a los componentes principales son: \* PorcMujeres \* LinTelf \* CrecPol \* ConsEner

Algo muy similar a lo obtenido con la matriz de correlación. Sería útil imprimir el tamaño de los vectores para estar seguros de cuales son los que más influyen.

En la **gráfica 2** podemos observar una elipse en la gráfica con la dispersión de los datos, lo que nos indica la variabilidad de los datos y podemos ver que hay algunos que salen fuera de la elipse.

En la **gráfica 4** podemos observar que el componente 1 y 2 explican poco menos del 60% de los datos.

Y finalmente en la **gráfica 5** podemos observar la contribución de las variables en los componentes principales de una mejor manera. Como podemos ver las variables que más contribuyen son 'LinTelf' y 'ConsEner'.