Multivariate Statistics

Term Project

현대자동차 및 기아자동차의
미국 내 차량 판매량 유지 및 증가를 위한 방향성 제시

담당교수　최 용 석

2020년　12월

부산대학교 자연과학대학 통계학과

201611531 정호재

목 차

1. Abstract

다음 분석은 현대자동차그룹이 차량판매에 있어서 앞으로 미국시장에서 나아갈 방향과 아직 미국 내에 출시되지 않은 신차들의 성적을 예측하는 것을 목표로 하여 2015년 1월부터 2020년 5월까지 미국 내 판매된 차량에 대하여 분석하였다.

미국 내 판매된 차량의 연비, 출력, 가격 등 차량의 특성과 제조사별 판매비율, 판매금액과 같은 판매성적에 대한 자료를 바탕으로 다변량 자료분석 기법인 PCA, FA를 통해서 어떤 변수가 판매성적에 더 큰 기여를 하는지 살펴보았다. 그리고 CA를 통해 그룹화를 시켜 그룹들의 특성을 알아보았다.

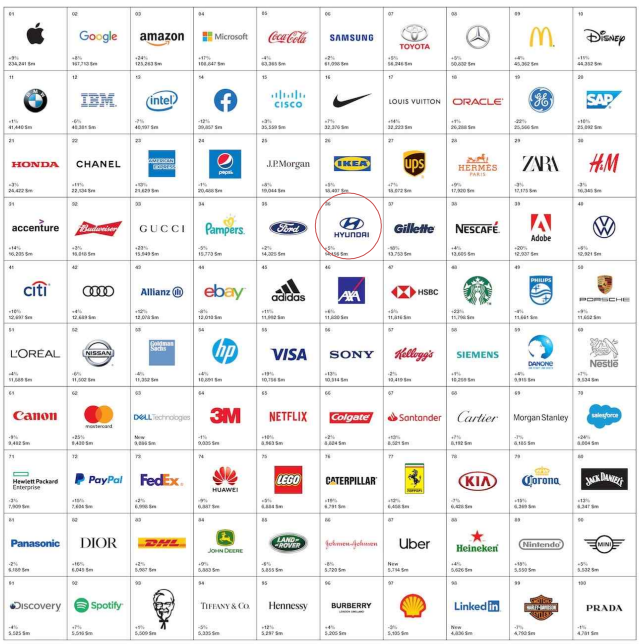
분석 결과 배기량과 출력, 가격, 차량종류는 판매성적에 영향을 미친다. 그리고 PC차량보다는 RV차량에 대한 선호도가 더 높았다. RV차량의 경우에는 출력량과 배기량이 판매성적에 가장 큰 영향을 주었고 PC차량은 연비가 판매량에 가장 큰 영향을 주었다.

2. Introduction

현대자동차그룹은 현대자동차를 모회사로 하는 글로벌 자동차 그룹으로 2000년 9월 현대그룹으로부터 분리돼 출범했다. 주요 사업 분야는 완성차, 철강, 건설, 자동차부품, 금융 부문이다. 주력 계열사는 현대 · 기아자동차를 비롯해 현대모비스, 현대위아, 현대제철, 현대글로비스 등이 있다. 그 중 완성차 생산/판매업체인 현대자동차는 2019 글로벌 100대 브랜드에서 브랜드 가치 약 141억 달러로 36위에 오른 대한민국을 대표하는 기업으로 계열사들 중 가장 규모가 크다. 게다가 그룹 내에서 현대자동차의 판매량이 각 부품, 철강 등 타계열사들에 영향을 준다. 이는 2001년 4월 현대자동차그룹으로 편입된 완성차 생산/판매업체 기아자동차 또한 마찬가지이다. 따라서 현대자동차그룹에서 현대 · 기아자동차의 역할이 중요하다.

현대 · 기아자동차의 2015년 1월부터 2020년 5월까지 각 국가별 판매량을 살펴봤을 때, 미국시장은 에서 항상 전체 판매량의 30%이상을 유지하는 중요한 지역이다. 현재 국내 자산총액 2위로 계속 자리를 유지하며 세계적인 성장을 위해선 미국시장에서의 선호도를 파악하며 미국시장에서의 판매량을 유지 및 늘려가는 것이 가장 빠른 방법이다. 코로나바이러스로 국제경기가 침체되어있는 현재 어떤 방향의 자동차를 생산하는 것이 시장에서 유리한지 31가지의 차종별로 PCA와 FA로 분석하여 각 특성의 영향력을 분석하였고, CA를 이용하여 전체 31가지의 차량이 어떤 비슷한 특성을 가진 차량들로 묶일 수 있는지를 알아보았다. 분석결과를 이용해 현대자동차그룹이 앞으로 미국시장에서 나아갈 방향과 아직 미국 내에 출시되지 않은 GV80, 더 뉴 산타페, 4세대 카니발 등 신차들의 성적을 예측하는데 도움이 될 것이다.

<그림 1-1> 2019년 베스트 글로벌 브랜드



자료 : Best Global Brands 2019 (interbrand)

<표 1-1> 2020년 지정집단 대등비교(계열사수, 자산총액)

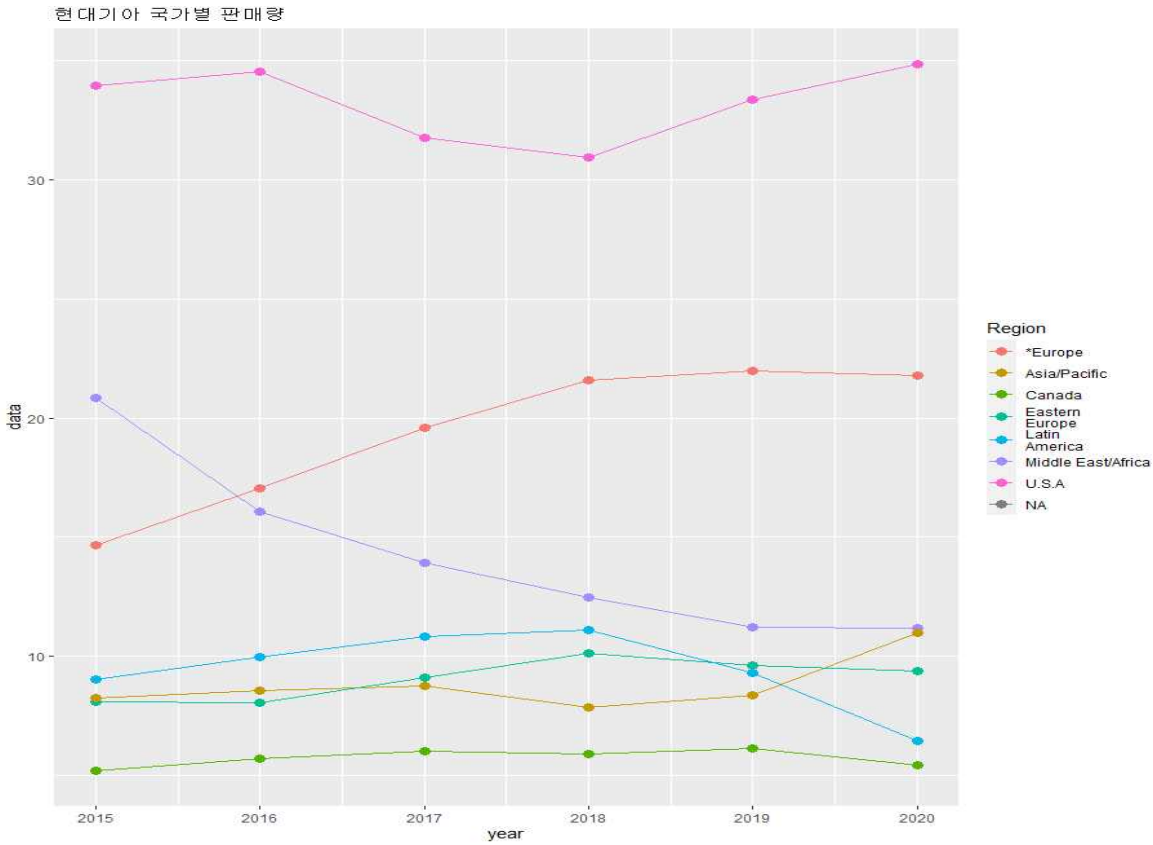
2020년 지정집단 대등비교(계열사수, 자산총액)

(202005 기준,단위:개,십억원)

구분	기업집단명	동일인	계열회사 수	자산총액
			202005	202005
계속지정집단	삼성	이재용	59	424,848
	현대자동차	정몽구	54	234,706
	에스케이	최태원	125	225,526
	엘지	구광모	70	136,967
	롯데	신동빈	86	121,524
	포스코	(주)포스코	35	80,340
	한화	김승연	86	71,686
	지에스	허창수	69	66,753
	현대중공업	정몽준	30	62,863
	농협	농업협동조합중앙회	58	60,596

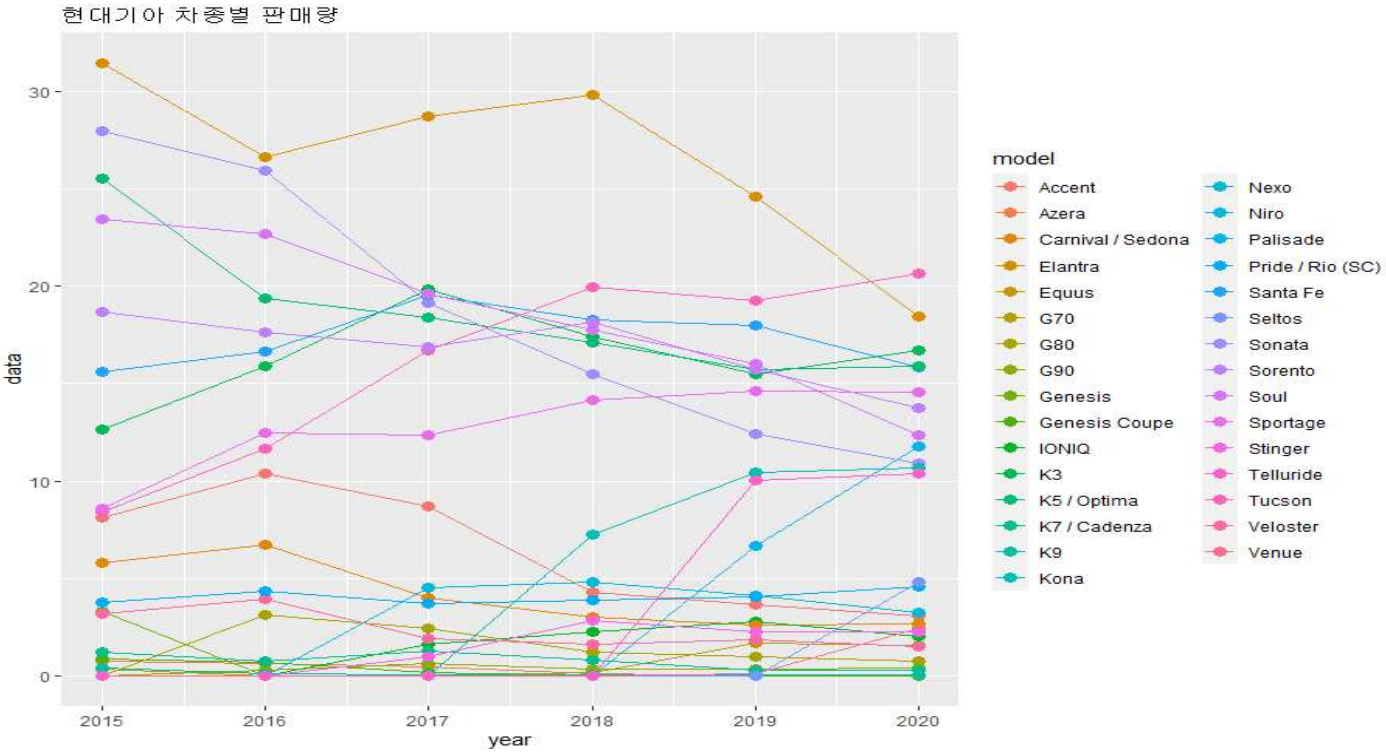
자료 : 기업진단포털

<그림 1-2> 현대 · 기아자동차 국가별 판매량



자료 : 현대자동차 및 기아자동차 판매실적

<그림 1-3> 현대 · 기아자동차 차종별 판매량



자료 : 현대자동차 및 기아자동차 판매실적

3. Data Description

<표 1-2> Multivarate data

model	연비_ HWYMPG	가격_ 시작가	제조사별 월 판매율	월 판매금액	출력_hp	배기량_cc	인승	유형
Nexo	37	58,735	0.012407	42,091,566	161	1600	5	-1
Equus	23	61,500	0.052396	186,127,334	429	3800	5	1
Genesis Coupe	24	21,000	0.282544	342,721,079	348	2000	4	1
K9	20	59,900	0.13039	388,912,977	365	3300	5	1
Venue	35	17,350	0.431701	432,631,574	121	1600	5	-1
Azera	29	34,100	0.325495	641,112,838	293	2400	5	1
G70	30	35,450	0.560307	1,147,301,838	252	2000	5	1
Genesis	29	38,750	0.553931	1,239,832,059	311	3300	5	1
K7 / Cadenza	24	33,100	0.760049	1,252,716,113	290	3000	5	1
G90	25	72,200	0.336912	1,405,042,009	365	2500	5	1
IONIQ	59	23,200	1.434922	1,922,877,119	139	1500	5	1
Veloster	34	18,800	2.338112	2,538,975,269	201	1400	5	1
Pride / Rio / SC	36	16,490	4.06579	3,338,476,867	120	1600	5	1
Stinger	24	33,090	2.089325	3,442,592,140	255	2000	5	1
G80	25	47,700	1.415016	3,898,657,780	300	2500	5	1
Niro	46	24,590	4.171657	5,107,989,010	139	1600	5	-1
Seltos	31	21,990	4.780508	5,234,583,217	147	1600	5	-1
Kona	33	20,300	4.722787	5,537,703,818	175	1600	5	-1
Accent	41	15,295	6.361791	5,620,361,448	120	1500	5	1
Carnival / Sedona	20	27,600	4.131935	5,678,653,136	276	3300	8	-1
Palisade	26	31,975	3.075129	5,679,488,139	291	3800	8	-1
K3 / Forte	37	17,890	16.34298	14,558,760,739	147	1600	5	1
Sportage	33	23,990	12.79092	15,279,699,516	181	2000	5	-1
Telluride	23	31,890	10.20461	16,204,429,228	291	3800	8	-1
Soul	31	17,490	18.64185	16,235,348,608	147	1600	5	-1
K5 / Optima	35	23,390	18.6502	21,721,830,618	185	1600	5	1
Tucson	28	23,550	16.11499	21,920,791,289	181	1600	5	-1
Sorento	34	26,990	16.80015	22,578,701,968	185	2200	5	-1
Sonata	38	23,600	18.6434	25,413,963,848	191	2000	5	1
Santa Fe	29	26,275	17.31687	26,281,338,848	235	1500	5	-1
Elantra	41	19,300	26.60805	29,662,357,555	201	1600	5	1

사용한 데이터는 2015년 1월부터 2020년 5월까지 판매된 31가지의 차종과 8개의 변수로 이루어졌다.
각 변수에 대한 설명은 다음과 같다. 이때 각 변수에 대한 차량의 스펙들은 가장 최신모델의 스펙을 적용하였다.

<표 1-3> 변수에 대한 설명

연비_HWYMPG	연비는 열의 효율을 뜻하며, 연료내의 잠재적인 에너지를 운동에너지 또는 일로 바꾸는 효율이다. 데이터에서는 고속도로 기준 미국에서 사용 중인 연비단위 HWYMPG(Miles Per Gallon, 1갤런의 연료로 주행한 마일)를 사용하였다. 전기·수소자동차의 경우에는 MPG를 구할 수 없으므로 전기 및 수소의 충전금액에 비례하여 MPG를 설정해주었다. 값이 클수록 성능이 뛰어나다. (단위 : HWYMPG)
가격_시작가	현대자동차 · 기아자동차 · 제네시스 미국 홈페이지 기준으로 가장 낮은 등급의 가격을 기준으로 선정하였다. (단위 : $)
제조사별 월판매율	현대자동차 및 기아자동차 홈페이지에 공시된 월별 미국 내 판매량을 기준으로 각 회사별 판매비율을 측정하였다. (단위 : %)
월판매금액	월 평균 판매수량과 가격을 바탕으로 월 평균 판매금액을 선정하였다. (단위 : $)
출력_hp	차량 내연 기관의 일률을 나타내는 단위이다. hp은 제임스 와트가 개발한 증기 기관의 성능을 재기 위해 도입한 단위로 이때 1마력은 야드파운드법에 따라 피트(1ft=0.3048m)와 파운드(1lb=453.599082g)를 써서 550ft·lb/sec이다. 값이 클수록 성능이 뛰어나다. (단위 : hp)
배기량_cc	내연 기관의 연소 정도에 영향을 미치는 용적의 크기를 나타내는 수치로 엔진의 성능지표 중 하나이다. 값이 클수록 차량의 규모가 크다. (단위 : cc)
인승	차량 내 기준 승차인원이다. (단위 : 인승)
유형	pc(승용차)는 1으로 rv(레저용 차량)은 -1로 이진수로 나타내서 자료를 살펴본다.

4. Analysis and Interpretations

1) Summary Of Data

<표 1-4> Summary Of Data

중심화자료행렬

	연비_HWYMPG	가격_시작가	제조사별.월판매율	월판매금액	출력_hp	배기량_cc	인승	유형
Nexo	5.3870968	28171.129	-6.8955646	-8504233258	-66.16129	-574.19355	-0.2580645	-1.1612903
Equus	-8.6129032	30936.129	-6.8555753	-8360197490	201.83871	1625.80645	-0.2580645	0.8387097
Genesis Coupe	-7.6129032	-9563.871	-6.6254275	-8203603745	120.83871	-174.19355	-1.2580645	0.8387097
K9	-11.6129032	29336.129	-6.7775820	-8157411847	137.83871	1125.80645	-0.2580645	0.8387097
Venue	3.3870968	-13213.871	-6.4762702	-8113693250	-106.16129	-574.19355	-0.2580645	-1.1612903
Azera	-2.6129032	3536.129	-6.5824762	-7905211986	65.83871	225.80645	-0.2580645	0.8387097
G70	-1.6129032	4886.129	-6.3476645	-7399022986	24.83871	-174.19355	-0.2580645	0.8387097
Genesis	-2.6129032	8186.129	-6.3540403	-7306492766	83.83871	1125.80645	-0.2580645	0.8387097
K7 / Cadenza	-7.6129032	2536.129	-6.1479220	-7293608711	62.83871	825.80645	-0.2580645	0.8387097
G90	-6.6129032	41636.129	-6.5710593	-7141282815	137.83871	325.80645	-0.2580645	0.8387097
IONIQ	27.3870968	-7363.871	-5.4730498	-6623447705	-88.16129	-674.19355	-0.2580645	0.8387097
Veloster	2.3870968	-11763.871	-4.5698597	-6007349555	-26.16129	-774.19355	-0.2580645	0.8387097
Pride / Rio / SC	4.3870968	-14073.871	-2.8421818	-5207847957	-107.16129	-574.19355	-0.2580645	0.8387097
Stinger	-7.6129032	2526.129	-4.8186465	-5103732684	27.83871	-174.19355	-0.2580645	0.8387097
G80	-6.6129032	17136.129	-5.4929554	-4647667044	72.83871	325.80645	-0.2580645	0.8387097
Niro	14.3870968	-5973.871	-2.7363143	-3438335814	-88.16129	-574.19355	-0.2580645	-1.1612903
Seltos	-0.6129032	-8573.871	-2.1274632	-3311741607	-80.16129	-574.19355	-0.2580645	-1.1612903
Kona	1.3870968	-10263.871	-2.1851848	-3008621006	-52.16129	-574.19355	-0.2580645	-1.1612903
Accent	9.3870968	-15268.871	-0.5461803	-2925963376	-107.16129	-674.19355	-0.2580645	0.8387097
Carnival / Sedona	-11.6129032	-2963.871	-2.7760368	-2867671688	48.83871	1125.80645	2.7419355	-1.1612903
Palisade	-5.6129032	1411.129	-3.8328428	-2866836685	63.83871	1625.80645	2.7419355	-1.1612903
K3 / Forte	5.3870968	-12673.871	9.4350090	6012435915	-80.16129	-574.19355	-0.2580645	0.8387097
Sportage	1.3870968	-6573.871	5.8829478	6733374692	-46.16129	-174.19355	-0.2580645	-1.1612903
Telluride	-8.6129032	1326.129	3.2966385	7658104404	63.83871	1625.80645	2.7419355	-1.1612903
Soul	-0.6129032	-13073.871	11.7338782	7689023784	-80.16129	-574.19355	-0.2580645	-1.1612903
K5 / Optima	3.3870968	-7173.871	11.7422243	13175505794	-42.16129	-574.19355	-0.2580645	0.8387097
Tucson	-3.6129032	-7013.871	9.2070157	13374466465	-46.16129	-574.19355	-0.2580645	-1.1612903
Sorento	2.3870968	-3573.871	9.8921832	14032377144	-42.16129	25.80645	-0.2580645	-1.1612903
Sonata	6.3870968	-6963.871	11.7354257	16867639024	-36.16129	-174.19355	-0.2580645	0.8387097
Santa Fe	-2.6129032	-4288.871	10.4088990	17735014024	7.83871	-674.19355	-0.2580645	-1.1612903
Elantra	9.3870968	-11263.871	19.7000759	21116032730	-26.16129	-574.19355	-0.2580645	0.8387097

표준화자료행렬

	연비_HWYMPG	가격_시작가	제조사별.월판매율	월판매금액	출력_hp	배기량_cc	인승	유형
Nexo	0.65370539	1.90600449	-0.88776874	-0.9096869	-0.78593171	-0.73279370	-0.2775503	-1.1575623
Equus	-1.04514575	2.09307908	-0.88262032	-0.8942796	2.39764733	2.07487655	-0.2775503	0.8360172
Genesis Coupe	-0.92379924	-0.64707314	-0.85298996	-0.8775290	1.43544621	-0.22230820	-1.3530578	0.8360172
K9	-1.40918528	1.98482615	-0.87257907	-0.8725879	1.63738965	1.43676968	-0.2775503	0.8360172
Venue	0.41101237	-0.89402513	-0.83378672	-0.8679114	-1.26109276	-0.73279370	-0.2775503	-1.1575623
Azera	-0.31706669	0.23924770	-0.84746020	-0.8456104	0.78209976	0.28817730	-0.2775503	0.8360172
G70	-0.19572018	0.33058611	-0.81722939	-0.7914640	0.29505968	-0.22230820	-0.2775503	0.8360172
Genesis	-0.31706669	0.55385777	-0.81805025	-0.7815662	0.99592223	1.43676968	-0.2775503	0.8360172
K7 / Cadenza	-0.92379924	0.17158962	-0.79151357	-0.7801880	0.74646268	1.05390555	-0.2775503	0.8360172
G90	-0.80245273	2.81702053	-0.84599033	-0.7638939	1.63738965	0.41579867	-0.2775503	0.8360172
IONIQ	3.32332861	-0.49822537	-0.70462722	-0.7085017	-1.04727028	-0.86041508	-0.2775503	0.8360172
Veloster	0.28966586	-0.79592092	-0.58834611	-0.6425985	-0.31077065	-0.98803645	-0.2775503	0.8360172
Pride / Rio / SC	0.53235888	-0.95221108	-0.36591640	-0.5570768	-1.27297178	-0.73279370	-0.2775503	0.8360172
Stinger	-0.92379924	0.17091304	-0.62037613	-0.5459398	0.33069676	-0.22230820	-0.2775503	0.8360172
G80	-0.80245273	1.15939758	-0.70718996	-0.4971550	0.86525294	0.41579867	-0.2775503	0.8360172
Niro	1.74582398	-0.40418064	-0.35228649	-0.3677944	-1.04727028	-0.73279370	-0.2775503	-1.1575623
Seltos	-0.07437367	-0.58009164	-0.27390003	-0.3542528	-0.95223807	-0.73279370	-0.2775503	-1.1575623
Kona	0.16831935	-0.69443380	-0.28133138	-0.3218283	-0.61962534	-0.73279370	-0.2775503	-1.1575623
Accent	1.13909143	-1.03306249	-0.07031792	-0.3129866	-1.27297178	-0.86041508	-0.2775503	0.8360172
Carnival / Sedona	-1.40918528	-0.20052982	-0.35740056	-0.3067512	0.58015631	1.43676968	2.9489722	-1.1575623
Palisade	-0.68110622	0.09547428	-0.49345894	-0.3066618	0.75834171	2.07487655	2.9489722	-1.1575623
K3 / Forte	0.65370539	-0.85748977	1.21470923	0.6431426	-0.95223807	-0.73279370	-0.2775503	0.8360172
Sportage	0.16831935	-0.44477549	0.75739949	0.7202605	-0.54835118	-0.22230820	-0.2775503	-1.1575623
Telluride	-1.04514575	0.08972334	0.42442538	0.8191776	0.75834171	2.07487655	2.9489722	-1.1575623
Soul	-0.07437367	-0.88455300	1.51067691	0.8224850	-0.95223807	-0.73279370	-0.2775503	-1.1575623
K5 / Optima	0.41101237	-0.48537033	1.51175142	1.4093670	-0.50083507	-0.73279370	-0.2775503	0.8360172
Tucson	-0.43841320	-0.47454504	1.18535626	1.4306496	-0.54835118	-0.73279370	-0.2775503	-1.1575623
Sorento	0.28966586	-0.24180125	1.27356808	1.5010254	-0.50083507	0.03293455	-0.2775503	-1.1575623
Sonata	0.77505190	-0.47116214	1.51087614	1.8043097	-0.42956092	-0.22230820	-0.2775503	0.8360172
Santa Fe	-0.31706669	-0.29017677	1.34009260	1.8970917	0.09311624	-0.86041508	-0.2775503	-1.1575623
Elantra	1.13909143	-0.76209188	2.53628418	2.2587550	-0.31077065	-0.73279370	-0.2775503	0.8360172

공분산행렬S

	연비_HWYMPG	가격_시작가	제조사별.월판매율	월판매금액	출력_hp	배기량_cc	인승	유형
연비_HWYMPG	6.791183e+01	-5.411928e+04	1.522948e+01	1.307069e+10	-5.076688e+02	-4.163656e+03	-2.330108e+00	3.311828e-01
가격_시작가	-5.411928e+04	2.184543e+08	-5.318346e+04	-5.477105e+13	8.822129e+05	6.291003e+06	2.961344e+02	2.973688e+03
제조사별.월판매율	1.522948e+01	-5.318346e+04	6.033092e+01	7.048481e+10	-2.589182e+02	-2.056488e+03	-1.103765e-01	-1.559459e+00
월판매금액	1.307069e+10	-5.477105e+13	7.048481e+10	8.739496e+19	-2.437384e+11	-1.873347e+12	4.658131e+08	-2.340748e+09
출력_hp	-5.076688e+02	8.822129e+05	-2.589182e+02	-2.437384e+11	7.086606e+03	5.167430e+04	1.362366e+01	2.820645e+01
배기량_cc	-4.163656e+03	6.291003e+06	-2.056488e+03	-1.873347e+12	5.167430e+04	6.139785e+05	4.435484e+02	3.096774e+01
인승	-2.330108e+00	2.961344e+02	-1.103765e-01	4.658131e+08	1.362366e+01	4.435484e+02	8.645161e-01	-3.763441e-01
유형	3.311828e-01	2.973688e+03	-1.559459e+00	-2.340748e+09	2.820645e+01	3.096774e+01	-3.763441e-01	1.006452e+00

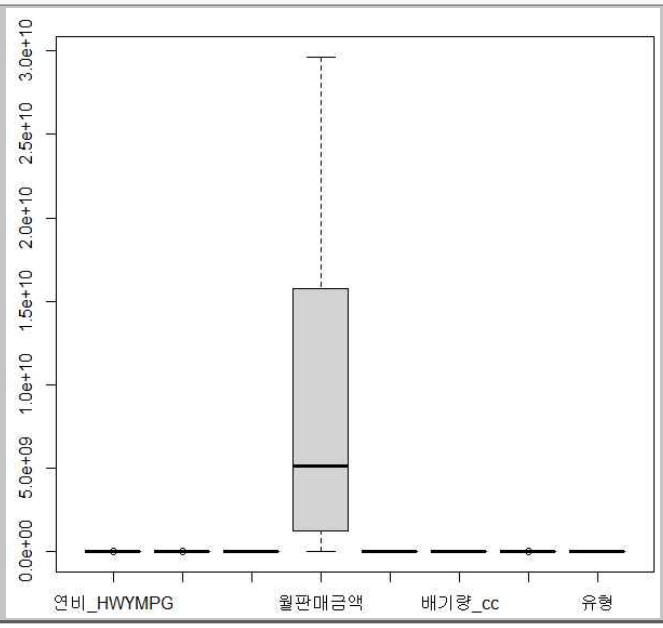
상관행렬R

	연비_HWYMPG	가격_시작가	제조사별.월판매율	월판매금액	출력_hp	배기량_cc	인승	유형
연비_HWYMPG	1.00000000	-0.44432322	0.23792622	0.16966117	-0.7317936	-0.64480076	-0.30410020	0.04005886
가격_시작가	-0.44432322	1.00000000	-0.46326158	-0.39639441	0.7090452	0.54320404	0.02154875	0.20054814
제조사별.월판매율	0.23792622	-0.46326158	1.00000000	0.97069406	-0.3959802	-0.33789339	-0.01528341	-0.20012785
월판매금액	0.16966117	-0.39639441	0.97069406	1.00000000	-0.3097146	-0.25573985	0.05358977	-0.24958306
출력_hp	-0.73179362	0.70904515	-0.39598019	-0.30971459	1.0000000	0.78339153	0.17405558	0.33398953
배기량_cc	-0.64480076	0.54320404	-0.33789339	-0.25573985	0.7833915	1.00000000	0.60880453	0.03939458
인승	-0.30410020	0.02154875	-0.01528341	0.05358977	0.1740556	0.60880453	1.00000000	-0.40346150
유형	0.04005886	0.20054814	-0.20012785	-0.24958306	0.3339895	0.03939458	-0.40346150	1.00000000

변동량척도

```
> detS; # 데이터의 일반화 분산
[1] 1.590896e+38
> detR; # 상관행렬의 일반화 분산
[1] 0.0005372194
> trS; # 데이터의 총 분산
[1] 8.739496e+19
> trR; # 상관행렬의 총 분산
[1] 8
```

Boxplot



일반화분산은 |S|=1.590896×10^38으로 매우 큰 값을 가진다. 상자그림을 보면 평균 또는 중위수를 중심으로 자료가 많이 흩어져 있는 것을 알 수 있다. 특히 월 판매금액에서의 분산은 8.739496×10^19으로 매우 높다. 이는 총 분산 tr(S)=8.739496×10^19에도 반영되어있다. 이와 반대로 상관행렬의 일반화분산 |R|=0.0005372194은 매우 작다. 각 변수들의 상관관계가 전체적으로 높다고 할 수 있다. 총 분산 tr(R)=8은 변수가 8개라서 이와 같은 값을 갖는다.
따라서 각 변수를 표준화한 표준화자료행렬으로 데이터

분석을 실시한다.

2) Multivarate Normality Test

다변량 통계학에서의 많은 기법과 접근방법들은 다변량 정규분포를 필요로 한다. 따라서 다변량 정규성을 Chisq Q-Q plot과 Mahalanobis distance로 검정하고자 한다.

Mahalanobis distance :

$$D_M(\vec{x}) = \sqrt{(\vec{x}-\vec{\mu})^T S^{-1}(\vec{x}-\vec{\mu})}.$$

```
> rq<-cor(cbind(q, m))[1,2]
> rq
[1] 0.9900863
```

Mahalanobis distance의 상관계수를 구하면 0.9900863으로 1에 가까운 값을 가지므로 다변량 정규성을 만족한다. Chi-Square Q-Q plot에서 점들이 직선에 가까우므로 시각적으로도 다변량 정규성을 만족한다고 할 수 있다.

3) DCA

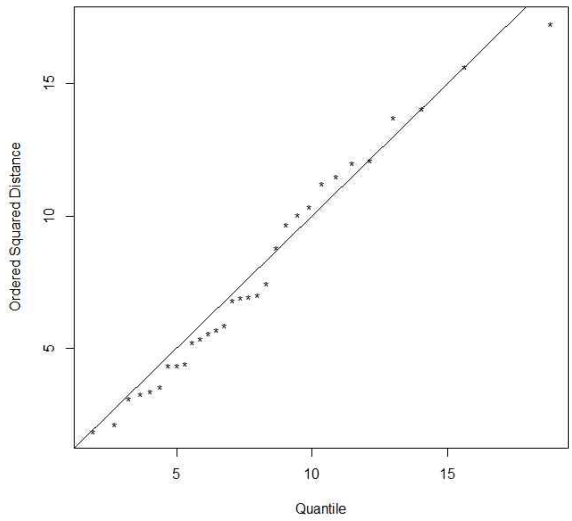
```
> rq #cedan
[1] 0.9883448

> rq #suv
[1] 0.9645605
```

Mahalanobis distance의 상관계수를 구하면 각각 0.9883448, 0.9645605으로 1에 가까운 값을 가지므로 다변량 정규성을 만족한다. Chi-Square Q-Q plot에서 점들이 직선에 가까우므로 시각적으로도 다변량 정규성을 만족한다고 할 수 있다.

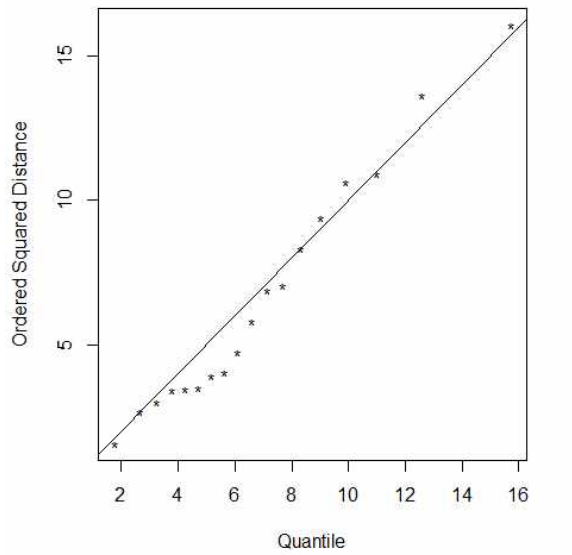
```
> cov.Mtest(x, ina)
$M.test
[1] 75.92568

$degrees
[1] 28

$critical
[1] 41.33714

$p.value
[1] 2.656586e-06
```

p.value가 0.05보다 작으므로 귀무가설 기각한다. 차량유형에 대한 분산이 동질하지 않다.

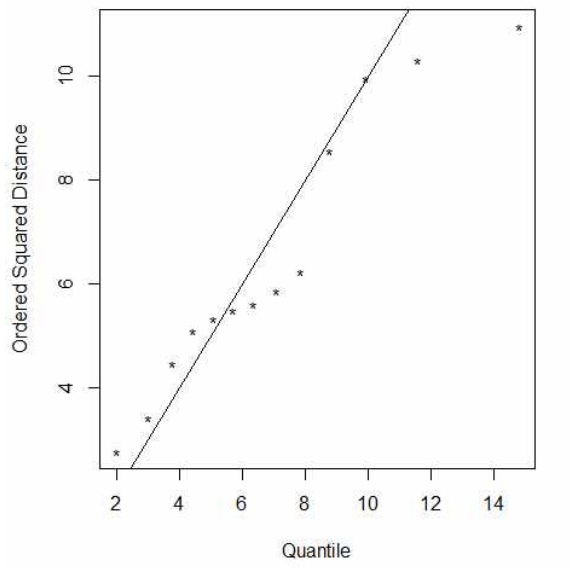
<그림 1-4> Chi-Square Q-Q plot (all data)



<그림 1-5> Chi-Square Q-Q plot (cedan data)



<그림 1-6> Chi-Square Q-Q plot (suv data)



다변량 정규성을 만족하고, 공분산행렬이 동질하지 않으므로 QDA 실시한다. (RSM)

```
> QDA
Call:
qda(유형 ~ ., data = X, prior = c(18, 13)/31)

Prior probabilities of groups:
       -1         1
0.5806452 0.4193548

Group means:
   연비_HWYMPG 가격_시작가 제조사별.월판매율  월판매금액  출력_hp 배기량_cc      인승
-1    31.23077    27132.69          8.707347 11247188455 194.6154  2138.462 5.692308
1     31.88889    33041.94          5.608422  6595701091 250.6667  2200.000 4.944444

> qct
    qcluster
     -1  1
  -1 11  2
  1   2 16

> (1-mean(X$유형==qcluster))*100
[1] 12.90323
```

오분류율이 12.9%로 생각보다 예측률이 높지는 않다.

CV를 적용해서 QDA를 적용하면 (CVM)

```
> QDA=qda(유형~., data=X, prior=c(18,13)/31, CV=TRUE)
> confusion=table(X$유형, QDA$class)
> confusion
     -1  1
  -1  6  7
  1   5 13
> EAER=(1-sum(diag(prop.table(confusion))))*100
> EAER
[1] 38.70968
```

오분류율이 38.7%로 이전보다 예측률이 많이 낮아졌다.

4) CRA

차량 유형과 판매비율이 10%보다 높은지 낮은지를 기준으로 이원분할표를 작성하여 이들 간의 관계를 알아보고자 한다.

```
> O
        a
         saledown saleup
  suv           7       6
  cedan        14       4

> chisq.test(O)

        Pearson's Chi-squared test with Yates'
        continuity correction

data:  O
X-squared = 1.0348, df = 1, p-value = 0.309
```

P-value가 0.309으로 유의수준 0.05보다 크다. 귀무가설을 기각할 수 없으므로 Pearson's Chi-squared test에서는 차량유형과 판매 비율간 서로 연관이 없다.

단순 CRA를 실시하면

[Step 1] n*p two-way table data matrix:

```
> O
        a
         saledown saleup
  suv           7       6
  cedan        14       4
```

[Step 2] Corresponence Matrix:

```
> F
        a
          saledown     saleup
  suv    0.2258065 0.1935484
  cedan 0.4516129 0.1290323
```

Row and Column centroids

```
> r;c;
      suv      cedan
0.4193548 0.5806452
 saledown     saleup
0.6774194 0.3225806
```

Centred correspondence matrix

```
> cF
        a
             saledown       saleup
  suv    -0.05827263  0.05827263
  cedan   0.05827263 -0.05827263
```

[Step 3] SVD of residual matrix

marginal sum of rows and columns f

```
> Dr;Dc
          [,1]      [,2]
[1,] 1.54422  0.000000
[2,] 0.00000  1.312335
          [,1]      [,2]
[1,] 1.214986  0.000000
[2,] 0.000000  1.760682
```

[Step 4] Coordinates of rows and columns of Simple CRA Map:

```
> A;B
              [,1]          [,2]
suv    -0.2972590  1.103488e-16
cedan   0.2146871  1.103488e-16
                [,1]          [,2]
saledown   0.1743255  1.103488e-16
saleup    -0.3660835  1.103488e-16
```

[Step 5] Goodness-of fit of s(>=2)-dimensional Simple CRA Map:

eigenvalue and GOF

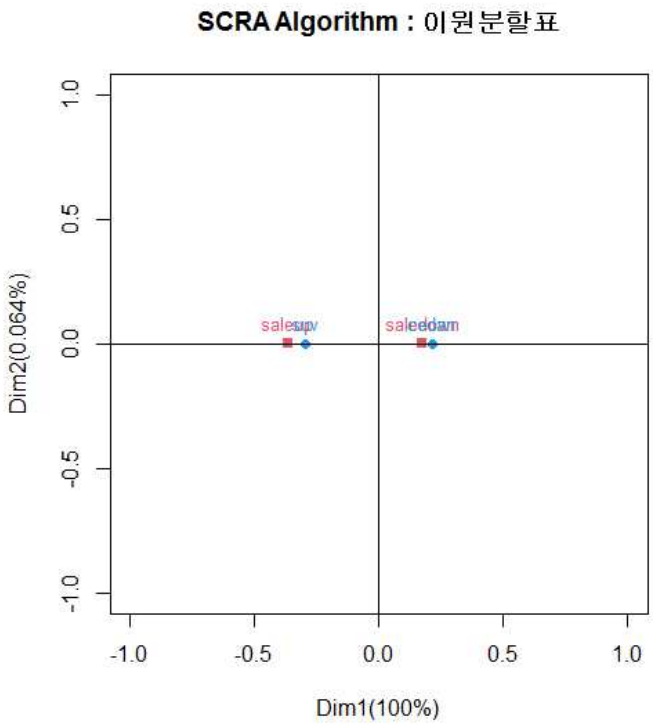
```
> rbind(round(eig, 3),round(per, 3))
          [,1] [,2]
[1,]    0.064    0
[2,] 100.000    0
```

1-dimsion's GOF은 100이다.

2-dimsion's GOF은 0.064이다.

simple CRA Plot을 보면 1-dimsion기준으로 SUV차량과 높은판매량이 Cedan차량과 낮은 판매량이 서로 관계가 있음을 알 수 있다.

<그림 1-7> Simple CRA Plot



5) MDS

데이터가 순서척도인 변수를 가지는 경우 비계량형 MDS를 사용하고 데이터가 연속형 변수(구간척도, 비율척도)인 경우 계량형 MDS를 사용한다. 본 데이터는 연속형 변수를 가지므로 계량형 MDS를 실시한다.

```
> con<-cmdscale(D, k=2, eig=T)
> con
$points
                          [,1]       [,2]
Nexo                -0.2765562 -0.9765762
Equus               -3.9530386 -0.2629272
Genesis Coupe       -1.1573063 -1.4510657
K9                  -3.4024149 -0.3978746
Venue                1.0276082 -0.6908454
Azera               -1.3603734 -0.9739997
G70                 -0.8727338 -1.1780743
Genesis             -2.0568060 -0.5822078
K7 / Cadenza        -1.8370294 -0.5120098
G90                 -3.0217431 -0.8971403
IONIQ                1.7696793 -2.3230981
Veloster             0.5191435 -1.3656764
Pride / Rio / SC     1.1229873 -1.2946712
Stinger             -0.9467988 -0.7821017
G80                 -1.8470350 -0.6623532
Niro                 1.5738896 -0.6947804
Seltos               0.9459246 -0.1303393
Kona                 0.9366751 -0.1533667
Accent               1.6295427 -1.2850782
Carnival / Sedona   -1.8917340  2.7643113
Palisade            -2.1479531  2.7020575
K3 / Forte           1.9543209 -0.2163463
Sportage             1.3028280  0.8214860
Telluride           -1.5765371  3.6233094
Soul                 2.1108923  1.0539362
K5 / Optima          1.8608884  0.2930954
Tucson               1.6983572  1.2935733
Sorento              1.5787141  1.3793932
Sonata               1.8667852  0.5244069
Santa Fe             1.6349604  1.5093650
Elantra              2.8148628  0.8655982

$eig
 [1]  1.084696e+02  5.653733e+01  3.515055e+01  1.832486e+01  1.397886e+01  4.644061e+00
2.296092e+00
 [8]  5.986138e-01  1.056366e-14  8.745634e-15  7.926608e-15  2.224531e-15  2.082600e-15
9.808090e-16
```

```
[15]  7.221023e-16  6.492945e-16  5.348755e-16  4.744873e-16  4.412324e-16  2.413783e-16
1.702979e-16
[22]  1.357512e-16 -2.527250e-16 -5.706093e-16 -1.022084e-15 -1.228215e-15 -1.832165e-15
-3.137808e-15
[29] -6.553734e-15 -1.301066e-14 -1.321434e-14

$x
NULL

$ac
[1] 0

$GOF
[1] 0.687529 0.687529
```

```
> which(con$eig<0)
[1] 23 24 25 26 27 28 29 30 31
```

음수의 개수가 전체의 1/3보다 크므로 음수인 eigenvalue가 많다. 따라서 비계량형 mds를 적용하여 살펴본다.

```
> con<-isoMDS(as.matrix(D), k=2)
initial  value 15.812371
iter   5 value 10.910419
final  value 10.756616
converged
> con
$points
                          [,1]         [,2]
Nexo                -0.2336736 -2.26090565
Equus               -3.9147572 -0.39544343
Genesis Coupe       -1.3895410 -1.31883225
K9                  -3.2171557 -0.40628305
Venue                0.9432439 -1.40732326
Azera               -1.1109589 -0.52012342
G70                 -0.6680302 -0.63302776
Genesis             -1.8594508 -0.21388867
K7 / Cadenza        -1.5829949 -0.20006541
G90                 -3.1766251 -1.14305726
IONIQ                2.1436916 -2.81678184
Veloster             0.3251544 -0.91662579
Pride / Rio / SC     0.8568226 -1.01972658
Stinger             -0.7752068 -0.28912269
G80                 -1.5999252 -0.48494528
Niro                 1.5958492 -1.31533428
Seltos               0.6787252 -0.19881071
```

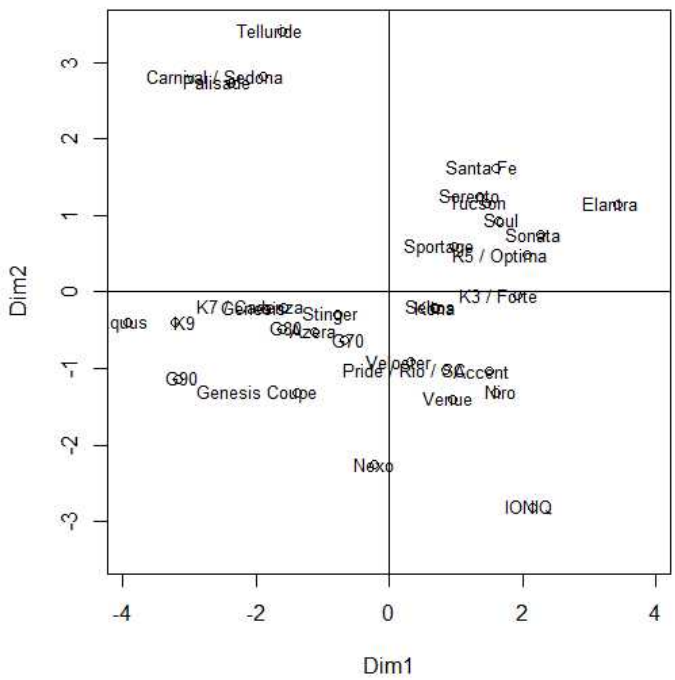
```
Kona                  0.6779679 -0.22071626
Accent                1.4888690 -1.03896441
Carnival / Sedona    -1.9005541  2.81182453
Palisade             -2.3784283  2.73605106
K3 / Forte            1.9158545 -0.05264649
Sportage              0.9770187  0.59212994
Telluride            -1.6049080  3.40629087
Soul                  1.6311900  0.92730762
K5 / Optima           2.0621177  0.47730873
Tucson                1.4589098  1.15433985
Sorento               1.3517682  1.24875687
Sonata                2.2607854  0.74111397
Santa Fe              1.6103748  1.61878152
Elantra               3.4338670  1.13871952

$stress
[1] 10.75662
```

stress가 10.75662으로 Kruskal's criterion of the number of dimensions에 의해 Goodness-of-fits는 Fair이다.

MDS MAPs을 살펴보자

1사분면에는 판매율이 높은 차량, 2사분면에는 대형 SUV, 3사분면에는 고급 Cedan, 4사분면에는 나머지 차량들이 분포해있다. 위쪽은 판매율이 높고 아래쪽은 판매율이 낮은 차량들이 분포해있고 왼쪽에는 판매가격이 높은 차량이 오른쪽에는 상대적으로 판매가격이 낮은 차량이 분포해있다.

<그림 1-8> MDS MAPs (non-metric MDS)



Shepard Diagram을 살펴봤을 때 단조 증가하므로 데이터나 알고리즘에 문제는 없다.

<그림 1-9> Shepard Diagram

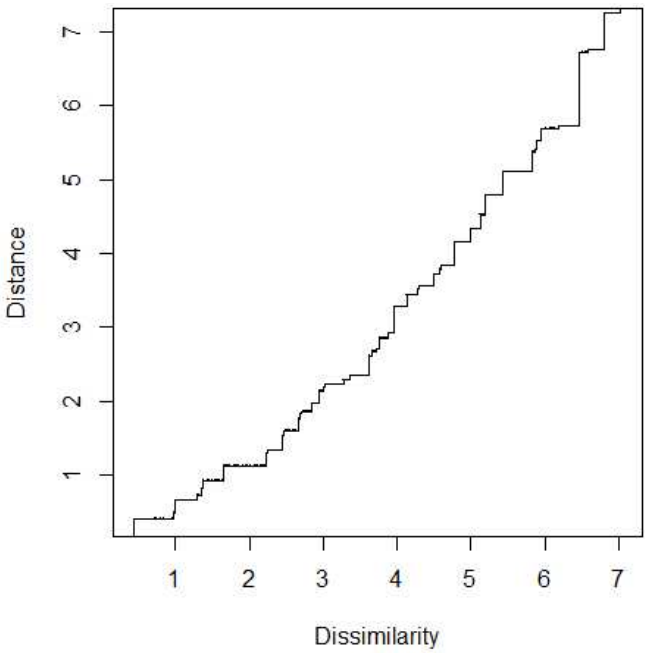
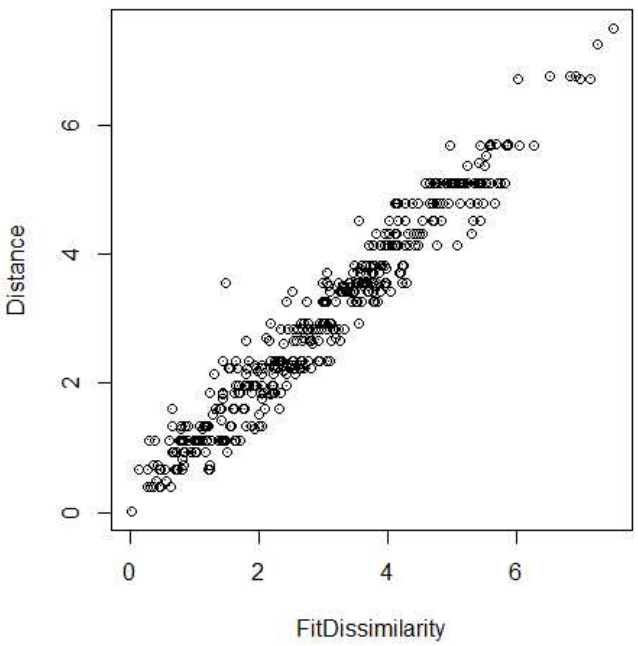


Image Diagram을 봤을 때 그래프의 점들이 y=x직선에 가까우므로 MDS MAPs는 잘 적합 되었다.

<그림 1-10> Image Diagram



6) PCA

표준화 자료행렬 Z를 사용하여 다양한 변수들 중에서 대표적인 변수를 구하여 차원을 줄이기 위해 주성분분석으로 데이터를 살펴본다.

상관행렬 R의 스펙트럼분해를 통해 고윳값과 고유벡터를 구하였다.

고윳값

```
> D_R
[1] 3.62 1.88 1.17 0.61 0.47 0.15 0.08 0.02
```

고유벡터

```
> V_R
      [,1]  [,2]  [,3]  [,4]  [,5]  [,6]  [,7]  [,8]
[1,]  0.38  0.28 -0.17  0.33 -0.67  0.39 -0.20  0.05
[2,] -0.41  0.08  0.19 -0.42 -0.69 -0.37  0.10  0.00
[3,]  0.37 -0.37  0.45  0.00 -0.08 -0.04  0.17  0.70
[4,]  0.33 -0.43  0.46 -0.04 -0.13  0.03 -0.12 -0.68
[5,] -0.47 -0.08  0.34 -0.01  0.06  0.39 -0.69  0.18
[6,] -0.44 -0.32  0.02  0.29 -0.13  0.50  0.59 -0.08
[7,] -0.16 -0.55 -0.37  0.48 -0.16 -0.45 -0.28  0.05
[8,] -0.13  0.44  0.52  0.63  0.07 -0.33  0.09 -0.07
```

고윳값의 총합에서 70% 이상 설명비율의 합을 갖는 m(<<p)개 고윳값을 선택하고자 한다.

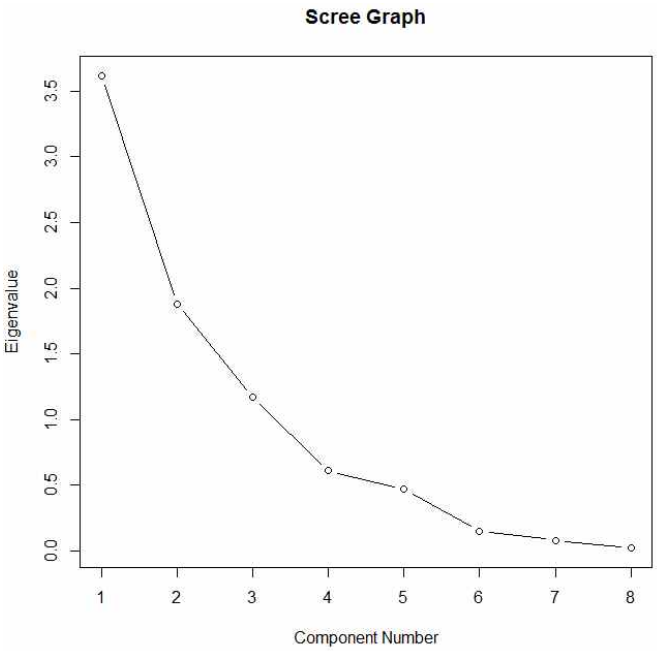
```
> round(gof_R,2)
[1] 45.25 23.50 14.62  7.62  5.88  1.88  1.00  0.25
```

45.25+23.50+14.62=76.11%

m=3개 고윳값의 설명비율이 83.37%이다.

스크리그림을 보면 팔꿈치가 4에서 이루어진다. 그러므로 주성분의 개수는 3개로 하는 것이 시각적으로도 타당해 보인다.

<그림 1-11> PCA Scree Graph



3개 고윳값 l_1, l_2, l_3에 대응하는 고유벡터 v_1, v_2, v_3을 활용하여 표준화 변수의 선형결합인 주성분 p_1, p_2, p_3을 구한다.

$p_1 = 0.38z_1 - 0.41z_2 + 0.37z_3 + 0.33z_4 - 0.47z_5 - 0.44z_6 - 0.16z_7 - 0.13z_8$

$p_2 = 0.28z_1 + 0.08z_2 - 0.37z_3 - 0.43z_4 - 0.08z_5 - 0.32z_6 - 0.55z_7 + 0.44z_8$

$p_3 = -0.17z_1 + 0.19z_2 + 0.45z_3 + 0.46z_4 + 0.34z_5 + 0.02z_6 - 0.37z_7 + 0.52z_8$

제1주성분은 제조사별 월 판매율, 월 판매금액과 연비는 부호가 양 나머지는 부호가 음으로 이들 간의 관계를 나타낸다.

제2주성분은 연비와 가격, 차량유형은 부호가 양 나머지는 부호가 음으로 이들 간의 관계를 나타낸다.

제3주성분은 연비와 차량인승이 음 나머지는 부호가 양으로 이들 간의 관계를 나타낸다.

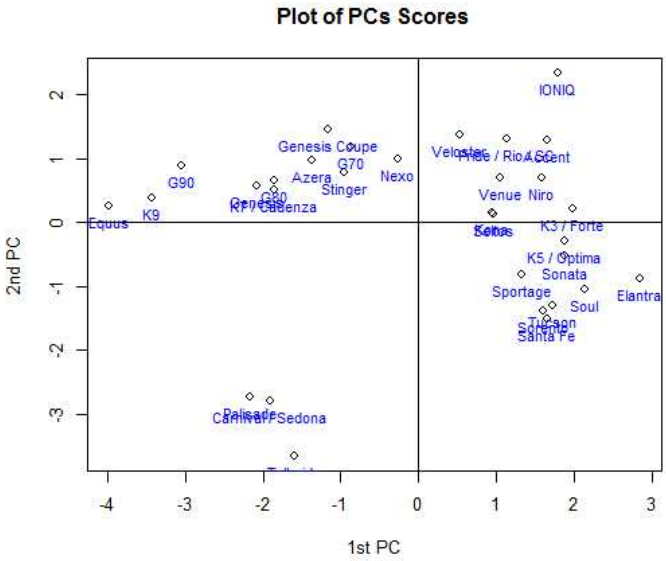
주성분점수를 차원 축소된 새로운 다변량 자료로
고려하여 분석을 한다. 31×3인 주성분점수를 구하였다.

```
> round(V_R2,2)
      [,1]  [,2]  [,3]
[1,]  0.38  0.28 -0.17
[2,] -0.41  0.08  0.19
[3,]  0.37 -0.37  0.45
[4,]  0.33 -0.43  0.46
[5,] -0.47 -0.08  0.34
[6,] -0.44 -0.32  0.02
[7,] -0.16 -0.55 -0.37
[8,] -0.13  0.44  0.52
```

```
> P_R
                      [,1]       [,2]       [,3]
Nexo             -0.2750166  0.9958515 -1.348052393
Equus            -3.9811137  0.2506432  1.160932214
Genesis Coupe    -1.1599717  1.4508464  0.665559042
K9               -3.4261034  0.3920240  0.945497882
Venue             1.0378571  0.7039711 -1.956846298
Azera            -1.3698477  0.9732494  0.138120545
G70              -0.8786086  1.1823837 -0.002446195
Genesis          -2.0726987  0.5753422  0.336262769
K7 / Cadenza     -1.8505476  0.5069377  0.286878298
G90              -3.0418106  0.8986192  1.042015043
IONIQ             1.7791451  2.3356565 -1.138479366
Veloster          0.5231791  1.3729717 -0.348819420
Pride / Rio / SC  1.1299303  1.3046477 -0.602382559
Stinger          -0.9527034  0.7844856  0.304631094
G80              -1.8599010  0.6617264  0.649700257
Niro              1.5869464  0.7065947 -1.571265254
Seltos            0.9562014  0.1404381 -1.221440742
Kona              0.9469275  0.1614428 -1.159764109
Accent            1.6397124  1.2945733 -0.478140447
Carnival / Sedona -1.9129439 -2.7839140 -1.567538512
Palisade          -2.1724618 -2.7245173 -1.622911188
K3 / Forte        1.9673653  0.2196179  0.767417502
Sportage          1.3146749 -0.8200704 -0.131096361
Telluride        -1.5972947 -3.6506367 -0.631183044
Soul              2.1296475 -1.0502096  0.065070605
K5 / Optima       1.8731732 -0.2940610  1.518987242
Tucson            1.7137080 -1.2927931  0.475541769
Sorento           1.5915625 -1.3820457  0.499527906
Sonata            1.8775776 -0.5295521  1.675522799
Santa Fe          1.6500708 -1.5123677  0.989684089
Elantra           2.8333440 -0.8718557  2.259016832
```

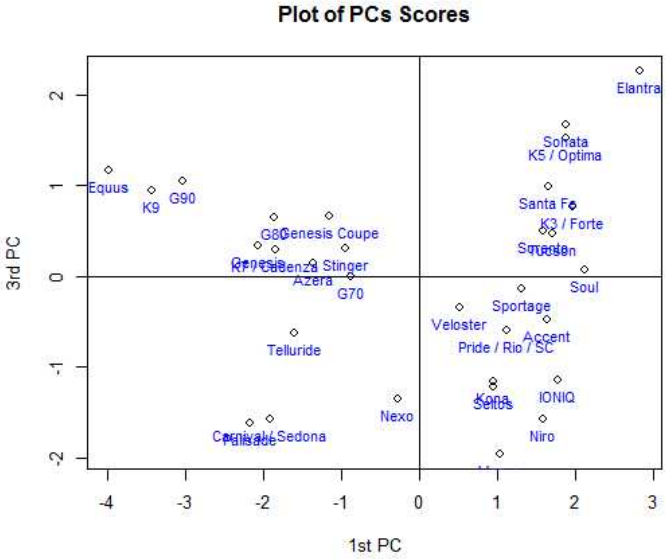
주성분 점수를 바탕으로 산점도를 그렸다.

<그림 1-12> Plot of PCs Scores p1 vs p2



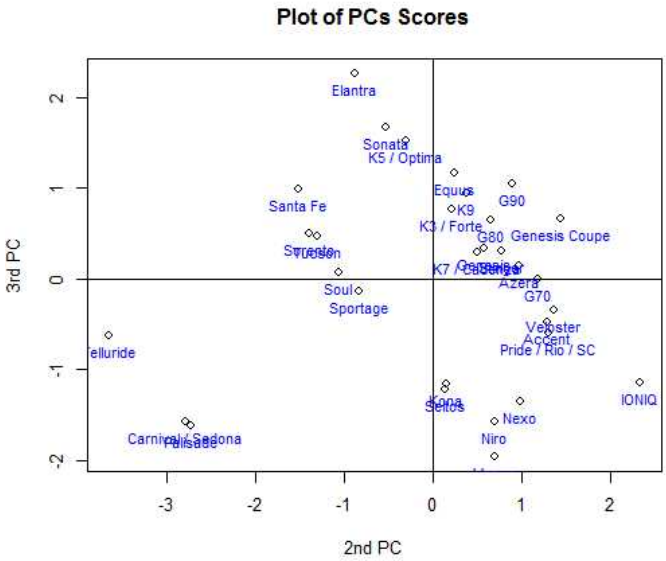
<그림 1-12>에서 연비가 높고 판매율 및 월 판매량이 높은 차량들은 오른쪽에 위치하고 있으며 연비와 가격이 높은 PC차량은 위쪽에 위치해있다. 1사분면에 위치한 점들은 연비가 높은 PC차량들이다. 반면 3사분면에 위치한 점들은 연비가 낮은 RV차량들이다.

<그림 1-13> Plot of PCs Scores p1 vs p3



<그림 1-13>에서 연비가 높고 판매율 및 월 판매량이 높은 개체들은 오른쪽에 위치하고 있으며 연비와 차량인승이 높은 차량은 아래쪽에 위치해있다. 2사분면에 위치한 점들은 연비와 판매율 차량인승이 낮은 차량들이다. 반면 4사분면에 위치한 점들은 연비가 높은 차량들이다.

<그림 1-14> Plot of PCs Scores p2 vs p3



<그림 1-14>에서 연비가 높은 PC차량은 오른쪽에 위치하고 있으며 연비와 차량인승이 높은 차량은 아래쪽에 위치해있다. 1사분면에 위치한 점들은 연비가 높은 PC차량들이다. 반면 4사분면에 위치한 자료들은 연비가 낮은 RV차량들이다.

Spectroal을 사용하는 princomp() 함수를 이용하여 Biplot을 그려본다.

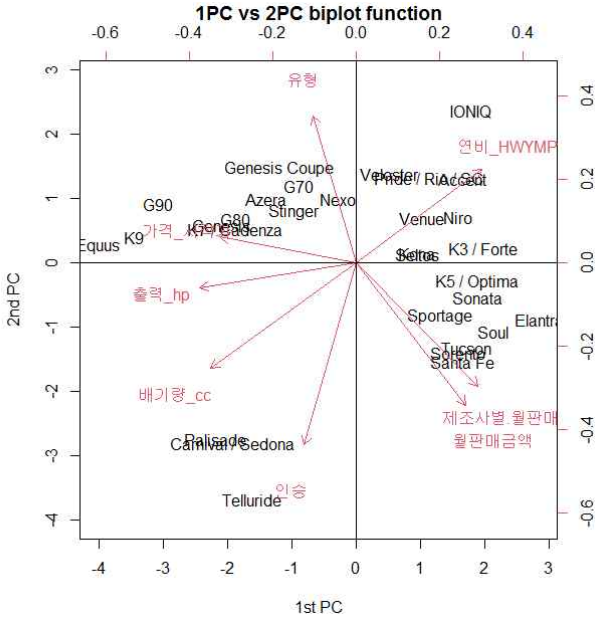
```
> round(pca.R$loadings[,1:3],2)
```

	Comp.1	Comp.2	Comp.3
연비_HWYMPG	0.38	0.28	0.17
가격_시작가	-0.41	0.08	-0.19
제조사별.월판매율	0.37	-0.37	-0.45
월판매금액	0.33	-0.43	-0.46
출력_hp	-0.47	-0.08	-0.34
배기량_cc	-0.44	-0.32	-0.02
인승	-0.16	-0.55	0.37
유형	-0.13	0.44	-0.52

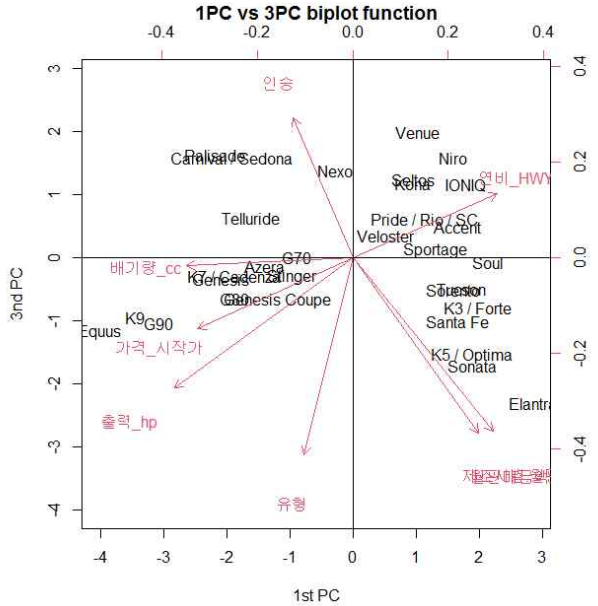
주성분은 제3주성분에서 부호가 바뀐 것 이외엔 앞의 자료와 동일하게 해석된다.

biplot을 작성하면 오른쪽 그림과 같다.
각 그림들의 설명력은 <그림1-15>부터 <그림1-16>까지 각각 68.75%, 59.89%, 38.12%이다. 각 변수간의 관계를 살펴보면 제조사별 월판매율과 월판매금액은 높은 상관을 보이고 각 그림 마다 차량 유형, 인승, 연비와 음의 상관을 보인다. <그림 1-17>은 다른 Biplot에 비해 설명력이 조금 떨어져 연비와의 관계를 제외한 나머지의 변수간의 관계를 비교해봤을 때 PC차량보다는 RV차량이 차량인승이 낮은 차량이 판매율이나 판매금액이 높은 것을 알 수 있다. 또한 가격과 출력, 배기량은 서로 약한 양의 상관이 있고 연비와는 음의 상관을 가진다.

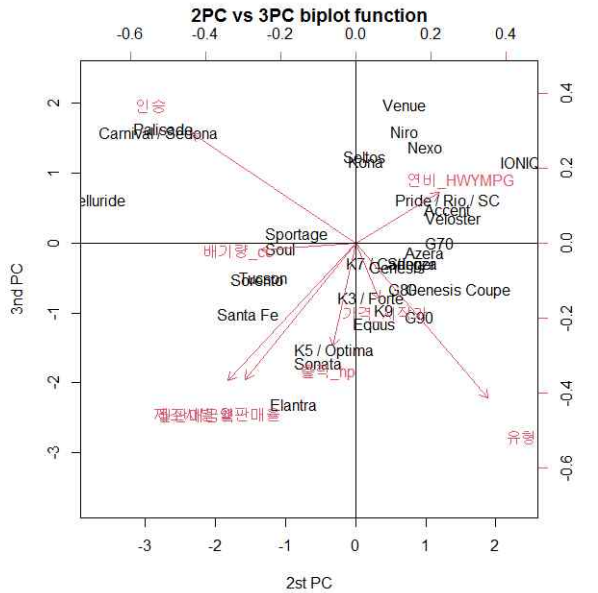
<그림 1-15> Biplot of PCs Scores p1 vs p2



<그림 1-16> Biplot of PCs Scores p1 vs p3



<그림 1-17> Biplot of PCs Scores p2 vs p3



7) FA

인자 분석 (factor analysis)은 인자 (factor)라고 불리는 잠재적으로 적은 숫자의 관찰되지 않은 변수 (variable)들로, 관찰된 서로 상관인 변수 (variable)들 사이에서의 분산 (variance)을 설명하기 위한 통계학적 방법이다. 주성분분석에서 통계적 모형을 설정하지 않았기 때문에 적절한 통계적 모형을 설정하여 많은 변수들 간의 상관관계를 알아보기 위해 인자분석으로 데이터를 확인하고자 한다. 또한 명확한 해석을 위하여 Varimax 인자회전을 하였다.

PCFA

Principal Components Analysis

Call: principal(r = Z, nfactors = 3, rotate = "varimax")

Standardized loadings (pattern matrix) based upon correlation matrix

	RC1	RC3	RC2	h2	u2	com
연비_HWYMPG	-0.82	0.04	0.16	0.70	0.302	1.1
가격_시작가	0.70	-0.34	0.21	0.65	0.348	1.7
제조사별.월판매율	-0.22	0.97	-0.04	0.98	0.015	1.1
월판매금액	-0.13	0.98	-0.10	0.98	0.019	1.1
출력_hp	0.94	-0.17	0.17	0.94	0.062	1.1
배기량_cc	0.87	-0.17	-0.31	0.87	0.127	1.3
인승	0.36	0.02	-0.82	0.81	0.192	1.4
유형	0.21	-0.12	0.82	0.74	0.263	1.2

	RC1	RC3	RC2
SS loadings	3.03	2.08	1.56
Proportion Var	0.38	0.26	0.20
Cumulative Var	0.38	0.64	0.83
Proportion Explained	0.45	0.31	0.23
Cumulative Proportion	0.45	0.77	1.00

Mean item complexity = 1.2

Test of the hypothesis that 3 components are sufficient.

The root mean square of the residuals (RMSR) is 0.07
with the empirical chi square 8.73 with prob < 0.27

Fit based upon off diagonal values = 0.97

MLFA

Call:

factanal(x = Z, factors = 3, rotation = "varimax")

Uniquenesses:

연비_HWYMPG	가격_시작가	제조사별.월판매율	월판매금액	출력_hp	배기량_cc	인승	유형
0.429	0.453	0.041	0.005	0.005	0.144	0.023	0.654

Loadings:

	Factor1	Factor2	Factor3
연비_HWYMPG	-0.749		
가격_시작가	0.652	-0.292	-0.193
제조사별.월판매율	-0.240	0.947	
월판매금액	-0.141	0.982	0.105
출력_hp	0.974	-0.161	-0.143
배기량_cc	0.829	-0.180	0.370
인승	0.316		0.937
유형	0.238	-0.160	-0.514

	Factor1	Factor2	Factor3
SS loadings	2.855	2.036	1.357
Proportion Var	0.357	0.254	0.170
Cumulative Var	0.357	0.611	0.781

Test of the hypothesis that 3 factors are sufficient.

The chi square statistic is 8.86 on 7 degrees of freedom.

The p-value is 0.263

Varimax를 설정한 PCFA와 MLFA중 인자 3개에 대한 기여율은 PCFA가 약 83% MLFA가 약 78.1%으로 PCFA의 기여율이 더 높으므로 PCFA로 분석을 실시한다.

<표 1-5> PCFA 수행결과

인자적재값

> L

	RC1	RC3	RC2
연비_HWYMPG	-0.8180691	0.04152139	0.16301600
가격_시작가	0.7012662	-0.33848010	0.21308222
제조사별.월판매율	-0.2243894	0.96565918	-0.04282018
월판매금액	-0.1349668	0.97609601	-0.09802965
출력_hp	0.9376447	-0.17298111	0.17139173
배기량_cc	0.8663154	-0.17068771	-0.30589501
인승	0.3608226	0.02494540	-0.82309292
유형	0.2107553	-0.12152938	0.82352643

인자적재행렬

> fpc

	RC1	RC3	RC2
Nexo	-0.61187789	-1.2728548	-0.3158921
Equus	2.11828162	-0.5263535	0.8553431
Genesis Coupe	0.38419835	-0.5413769	1.1908529
K9	1.75869355	-0.5582745	0.7928851
Venue	-1.33913723	-1.1446222	-0.8426609
Azera	0.40810679	-0.7045128	0.6060600
G70	0.08636108	-0.7383985	0.6280761
Genesis	0.89185921	-0.6164605	0.5180946
K7 / Cadenza	0.79323858	-0.5559523	0.4491375
G90	1.49412633	-0.6075866	1.1206469
IONIQ	-1.79587740	-1.1472333	0.5454935
Veloster	-0.70857088	-0.6336019	0.5169910
Pride / Rio / SC	-1.05193312	-0.5851895	0.3217815
Stinger	0.33085568	-0.4147325	0.6052249
G80	0.88827209	-0.4123304	0.7535822
Niro	-1.44541841	-0.7695909	-0.5983767
Seltos	-0.89511361	-0.5044051	-0.6873763
Kona	-0.87419751	-0.4798240	-0.6357262
Accent	-1.23105213	-0.3680680	0.3957510
Carnival / Sedona	0.99600054	-0.2710372	-2.4589999
Palisade	1.07545754	-0.3991224	-2.4593549
K3 / Forte	-0.65495189	0.8953837	0.5966636
Sportage	-0.41866097	0.6292216	-0.5193144
Telluride	1.41151321	0.7216414	-2.3377595
Soul	-0.64998270	1.0645527	-0.5202759
K5 / Optima	-0.21327090	1.5198581	0.7888978
Tucson	-0.25782352	1.2880316	-0.3968834
Sorento	-0.17286006	1.3062329	-0.4269540
Sonata	-0.10025568	1.7098467	0.7621212
Santa Fe	0.01058966	1.6605882	-0.1931062
Elantra	-0.22657034	2.4561709	0.9450774

특성분산

연비_HWYMPG	가격_시작가	제조사별.월판매율	월판매금액	출력_hp	배기량_cc	인승	유형
0.30246472	0.34825289	0.01531815	0.01941072	0.06152483	0.12679159	0.19170285	0.26261704

잔차행렬

> Rm

	연비_HWYMPG	가격_시작가	제조사별.월판매율	월판매금액	출력_hp	배기량_cc	인승	유형
연비_HWYMPG	-4.440892e-16	1.086794e-01	2.124502e-02	3.470054e-02	1.450735e-02	1.208581e-01	1.242191e-01	8.326934e-02
가격_시작가	1.086794e-01	-4.440892e-16	3.007579e-02	4.953070e-02	-4.356459e-02	-5.690728e-02	-4.765395e-02	-1.638615e-01
제조사별.월판매율	2.124502e-02	3.007579e-02	-6.661338e-16	-6.364793e-03	-1.120280e-02	8.226311e-03	6.347624e-03	-2.170745e-04
월판매금액	3.470054e-02	4.953070e-02	-6.364793e-03	-4.440892e-16	2.483971e-03	-2.195213e-03	-2.747774e-03	-2.178373e-02
출력_hp	1.450735e-02	-4.356459e-02	-1.120280e-02	2.483971e-03	-2.220446e-16	-6.002376e-03	-1.888139e-02	-2.579197e-02
배기량_cc	1.208581e-01	-5.690728e-02	8.226311e-03	-2.195213e-03	-6.002376e-03	-4.440892e-16	4.869624e-02	8.798308e-02
인승	1.242191e-01	-4.765395e-02	6.347624e-03	-2.747774e-03	-1.888139e-02	4.869624e-02	4.440892e-16	2.013636e-01
유형	8.326934e-02	-1.638615e-01	-2.170745e-04	-2.178373e-02	-2.579197e-02	8.798308e-02	2.013636e-01	1.110223e-16

인자 f1(RC1)의 적재 값을 보면 출력과 배기량, 가격이 양(+) 연비가 음(-)으로 다른 변수들에 비해 절댓값이 크고 이들 간의 관계를 보여주는 인자이다.
인자 f2(RC3)의 적재 값을 보면 제조사별 월판매율과 월판매금액이 양(+)으로 다른 변수들에 비해 절댓값이 크다. 즉, 판매량에 관한 인자이다.
인자 f3(RC2)의 적재 값을 보면 유형은 양(+) 인승은 음(-)으로 다른 변수들에 비해 절댓값이 크고 이들 간의 관계를 보여주는 인자이다.

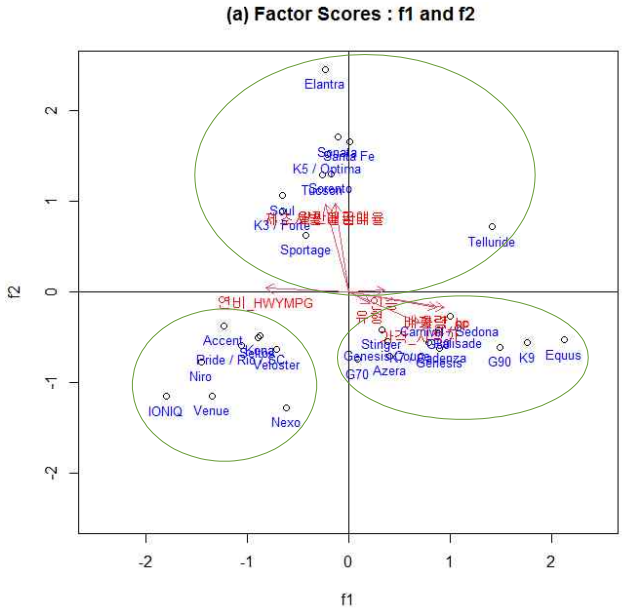
특성분산에서 각 변수들은 비교적 0에 가까워 설명력이 높다. 특히 제조사별 월판매율과 월판매금액, 출력은 다른 변수들에 비하여 설명력이 더 높다.

잔차행렬을 보았을 때 비대각원소들의 절댓값이 낮으므로 각각의 두 요인들을 잘 설명을 하고 있다.

위의 자료들을 바탕으로 인자행렬도를 작성한다.

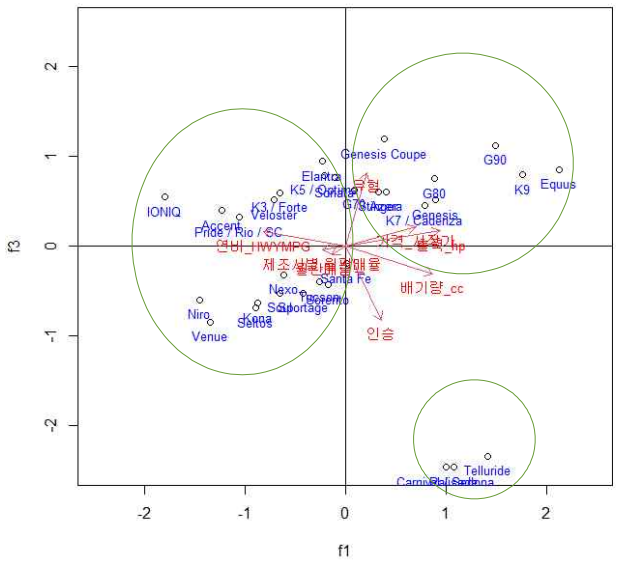
<그림 1-18>은 약 64%의 설명력을 가진다. 출력과 배기량, 가격이 높은 차량들은 오른쪽에 위치하고 있으며 연비가 높은 차량은 왼쪽에 위치해있다. 또한 제조사별 월판매율과 월판매금액이 높은 차량은 위쪽에 위치해있다. 변수들 간의 관계를 살펴보면 제조사별 월판매율과 월판매금액은 높은 연관이 있고 배기량과 출력, 가격, PC차량은 서로 높은 연관이 있다. 이 두 집단은 서로 약한 음의 상관관계를 가진다. 판매율이 높은 차량, 판매율은 낮지만 연비가 좋은 차량, 판매율은 낮지만 배기량, 출력이 좋은 차량 3가지의 집단으로 분류된다.

<그림 1-18> Biplot of Factor Scores f1 vs f2



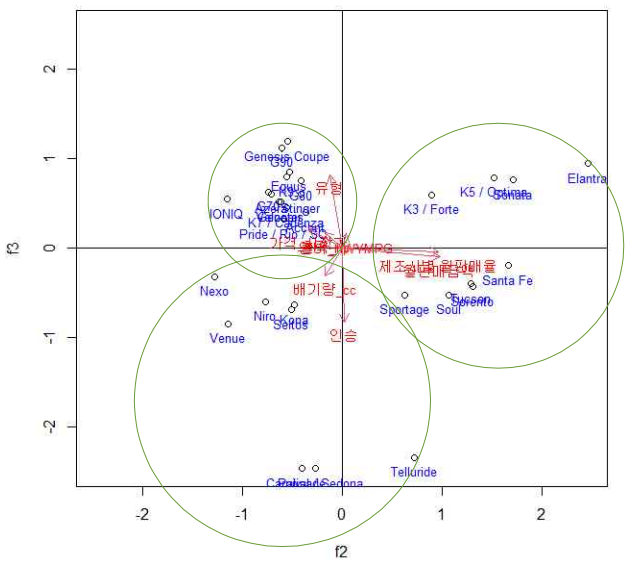
<그림 1-19>은 약 58%의 설명력을 가진다. 출력과 배기량, 가격이 높은 차량들은 오른쪽에 위치하고 있으며 연비가 높은 차량은 왼쪽에 위치해있다. 또한 탑승인승이 낮은 PC차량은 위쪽에 위치해있다. 변수들 간의 관계를 살펴보면 제조사별 월판매율과 월판매금액은 높은 연관이 있고 배기량과 출력, 가격은 서로 높은 연관이 있다. 이 두 집단은 서로 강한 음의 상관관계를 가진다. 연비가 높은 차량, 가격이 높고 출력이 좋은 차량, 탑승인승이 높은 차량 3가지의 집단으로 분류된다.

<그림 1-19> Biplot of Factor Scores f1 vs f3



<그림 1-20>은 약 46%의 설명력을 가진다. 제조사별 월판매율과 월판매금액이 높은 차량은 오른쪽에 위치해있다. 또한 탑승인승이 낮은 PC차량은 위쪽에 위치해있다. 변수들 간의 관계를 살펴보면 제조사별 월판매율과 월판매금액은 높은 연관이 있고 가격과 출력은 서로 높은 연관이 있다. 이 두 집단은 서로 약한 음의 상관관계를 가진다. 그리고 연비와 배기량, PC차량과 차량인승 또한 각각 음의 상관관계를 가진다. 판매율이 높은 차량, 판매율은 낮지만 연비가 좋은 PC차량, 배기량과 차량인승이 높은 차량 3가지의 집단으로 분류된다.

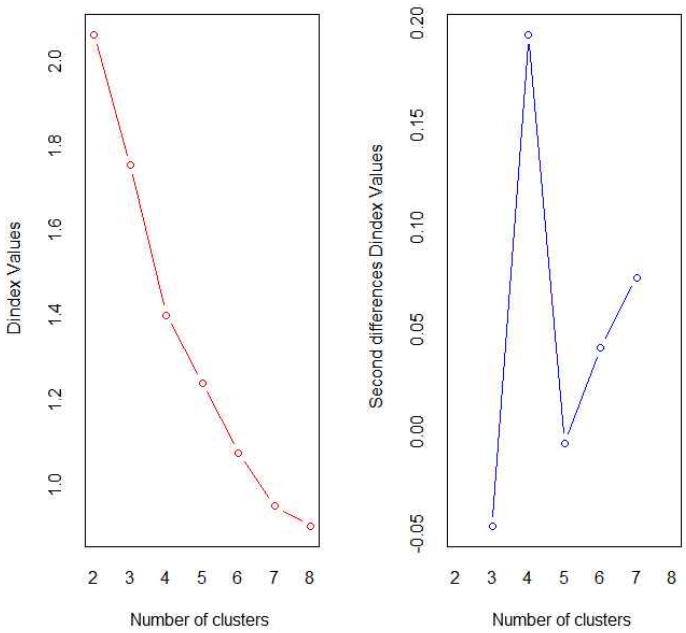
<그림 1-20> Biplot of Factor Scores f2 vs f3



8) CA

주성분 분석에서 얻은 데이터 분석을 바탕으로 좀 더 자세한 분석을 위해 군집분석을 진행한다. NbClust() 함수의 옵션 index="all"을 이용하여 주어진 모든 지수를 통해 추천된 빈도가 높은 군집의 수를 선택하여 계층 및 비계층 군집분석을 실시한다.

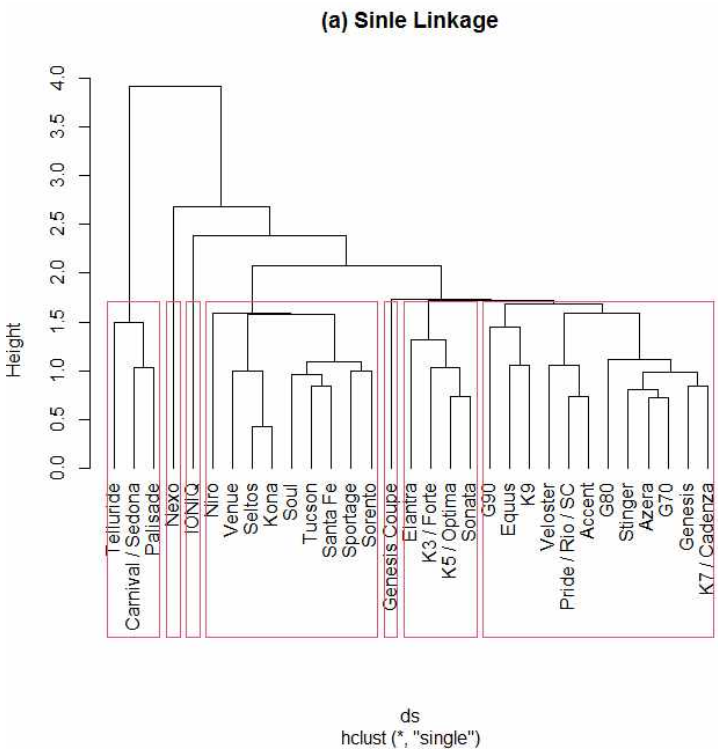
<그림 1-21> Dindex 그림



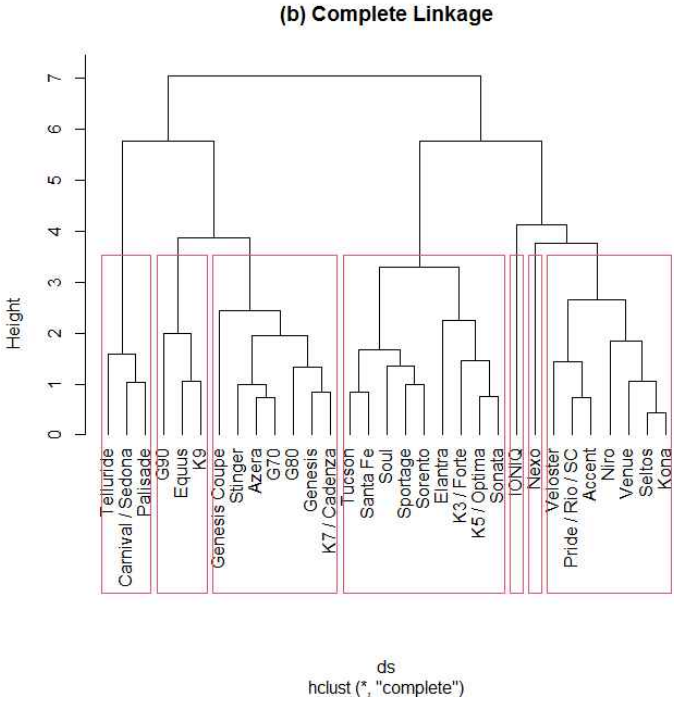

```
**********************************************
* Among all indices:
* 1 proposed 2 as the best number of clusters
* 2 proposed 3 as the best number of clusters
* 6 proposed 4 as the best number of clusters
* 1 proposed 5 as the best number of clusters
* 3 proposed 6 as the best number of clusters
* 7 proposed 7 as the best number of clusters
* 3 proposed 8 as the best number of clusters

                ***** Conclusion *****

* According to the majority rule, the best
number of clusters is  7
**********************************************
```

추천된 군집의 수는 7가지로 이를 바탕으로 단일연결법과 완전연결법, 평균연결법, 와드연결법, K-평균법, K-대표개체법, Allindex best partition으로 군집분석을 실시한다.

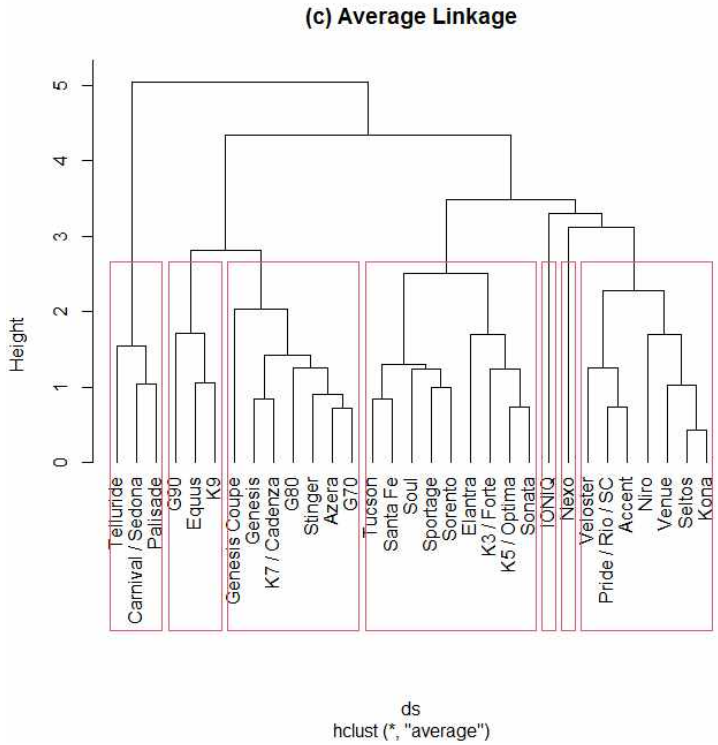
<그림 1-22> 단일연결법



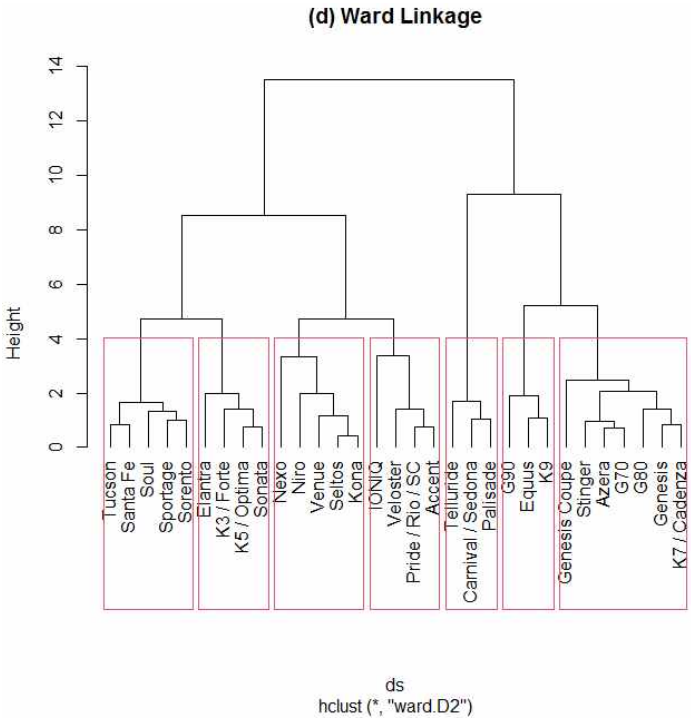
<그림 1-23> 완전연결법



<그림 1-24> 평균연결법



<그림 1-25> 와드연결법



K-평균법

> C1;C2;C3;C4;C5;C6;C7

	model	cluster
Equus	Equus	1
Genesis Coupe	Genesis Coupe	1
K9	K9	1
Azera	Azera	1
G70	G70	1
Genesis	Genesis	1
K7 / Cadenza	K7 / Cadenza	1
G90	G90	1
Stinger	Stinger	1
G80	G80	1

	model	cluster
Venue	Venue	2
Niro	Niro	2
Seltos	Seltos	2
Kona	Kona	2

	model	cluster
Sportage	Sportage	3
Soul	Soul	3
Tucson	Tucson	3
Sorento	Sorento	3
Santa Fe	Santa Fe	3

	model	cluster
IONIQ	IONIQ	4
Veloster	Veloster	4
Pride / Rio / SC	Pride / Rio / SC	4
Accent	Accent	4

	model	cluster
Nexo	Nexo	5

	model	cluster
Carnival / Sedona	Carnival / Sedona	6
Palisade	Palisade	6
Telluride	Telluride	6

	model	cluster
K3 / Forte	K3 / Forte	7
K5 / Optima	K5 / Optima	7
Sonata	Sonata	7
Elantra	Elantra	7

K-대표개체법

	model	cluster
Nexo	Nexo	1
Venue	Venue	1
Niro	Niro	1
Seltos	Seltos	1
Kona	Kona	1

	model	cluster
Equus	Equus	2
K9	K9	2
G90	G90	2

	model	cluster
Genesis Coupe	Genesis Coupe	3
Azera	Azera	3
G70	G70	3
Genesis	Genesis	3
K7 / Cadenza	K7 / Cadenza	3
Stinger	Stinger	3
G80	G80	3

	model	cluster
IONIQ	IONIQ	4
Veloster	Veloster	4
Pride / Rio / SC	Pride / Rio / SC	4
Accent	Accent	4

	model	cluster
Carnival / Sedona	Carnival / Sedona	5
Palisade	Palisade	5
Telluride	Telluride	5

	model	cluster
K3 / Forte	K3 / Forte	6
K5 / Optima	K5 / Optima	6
Sonata	Sonata	6
Elantra	Elantra	6

	model	cluster
Sportage	Sportage	7
Soul	Soul	7
Tucson	Tucson	7
Sorento	Sorento	7
Santa Fe	Santa Fe	7

K-평균법에 의한 군집의 각 변수별 평균

```
> aggregate(X, by=list(kmeans$cluster),FUN=mean)
  Group.1 연비_HWYMPG 가격_시작가 제조사별.월판매율  월판매금액 출력_hp 배기량_cc 인승 유형
1       1       25.30    43679.00       0.65063662  1394501617   320.8  2680.000  4.9    1
2       2       36.25    21057.50       3.52666340  4078226905   145.5  1600.000  5.0   -1
3       3       31.00    23659.00      16.33295631 20459176046   185.8  1780.000  5.0   -1
4       4       42.50    18446.25       3.55015360  3355172676   145.0  1500.000  5.0    1
5       5       37.00    58735.00       0.01240689    42091566   161.0  1600.000  5.0   -1
6       6       23.00    30488.33       5.80389116  9187523501   286.0  3633.333  8.0   -1
7       7       37.75    21045.00      20.06115525 22839228190   181.0  1700.000  5.0    1
```

```
> aggregate(Z, by=list(kmeans$cluster),FUN=mean)
  Group.1 연비_HWYMPG  가격_시작가 제조사별.월판매율 월판매금액    출력_hp  배기량_cc       인승       유형
1       1 -0.76604877  0.887344444       -0.8055999 -0.7650214  1.1123367  0.6455171 -0.3851011  0.8360172
2       2  0.56269551 -0.643182804       -0.4353262 -0.4779467 -0.9700566 -0.7327937 -0.2775503 -1.1575623
3       3 -0.07437367 -0.467170310        1.2134187  1.2743024 -0.4913319 -0.5030752 -0.2775503 -1.1575623
4       4  1.32111120 -0.819854964       -0.4323019 -0.5552909 -0.9759961 -0.8604151 -0.2775503  0.8360172
5       5  0.65370539  1.906004491       -0.8877687 -0.9096869 -0.7859317 -0.7327937 -0.2775503 -1.1575623
6       6 -1.04514575 -0.005110731       -0.1421447  0.0685882  0.6989466  1.8621743  2.9489722 -1.1575623
7       7  0.74471527 -0.644028530        1.6934052  1.5288936 -0.5483512 -0.6051723 -0.2775503  0.8360172
```

K-대표개체법에 의한 군집의 각 변수별 평균

```
> aggregate(X, by=list(kmedoids$cluster),FUN=mean)
  Group.1 연비_HWYMPG 가격_시작가 제조사별.월판매율  월판매금액  출력_hp 배기량_cc      인승 유형
1       1    36.40000    28593.00       2.8238121  3270999837 148.6000  1600.000 5.000000   -1
2       2    22.66667    64533.33       0.1732327   660027440 386.3333  3200.000 5.000000    1
3       3    26.42857    34741.43       0.8552383  1709276264 292.7143  2457.143 4.857143    1
4       4    42.50000    18446.25       3.5501536  3355172676 145.0000  1500.000 5.000000    1
5       5    23.00000    30488.33       5.8038912  9187523501 286.0000  3633.333 8.000000   -1
6       6    37.75000    21045.00      20.0611553 22839228190 181.0000  1700.000 5.000000    1
7       7    31.00000    23659.00      16.3329563 20459176046 185.8000  1780.000 5.000000   -1
```

```
> aggregate(Z, by=list(kmedoids$cluster),FUN=mean)
  Group.1 연비_HWYMPG  가격_시작가 제조사별.월판매율 월판매금액    출력_hp  배기량_cc       인승       유형
1       1  0.58089749 -0.133345345       -0.5258147 -0.5642948 -0.9332316 -0.7327937 -0.2775503 -1.1575623
2       2 -1.08559458  2.298308590       -0.8670632 -0.8435871  1.8908089  1.3091483 -0.2775503  0.8360172
3       3 -0.62910057  0.282645525       -0.7792585 -0.7313503  0.7787058  0.3611038 -0.4311943  0.8360172
4       4  1.32111120 -0.819854964       -0.4323019 -0.5552909 -0.9759961 -0.8604151 -0.2775503  0.8360172
5       5 -1.04514575 -0.005110731       -0.1421447  0.0685882  0.6989466  1.8621743  2.9489722 -1.1575623
6       6  0.74471527 -0.644028530        1.6934052  1.5288936 -0.5483512 -0.6051723 -0.2775503  0.8360172
7       7 -0.07437367 -0.467170310        1.2134187  1.2743024 -0.4913319 -0.5030752 -0.2775503 -1.1575623
```

Allindex best partition

```
$Best.partition
            Nexo            Equus   Genesis Coupe               K9
               4                5              3                5
           Venue            Azera              G70          Genesis
               4                3              3                3
     K7 / Cadenza              G90           IONIQ          Veloster
               3                5              7                7
 Pride / Rio / SC          Stinger              G80             Niro
               7                3              3                4
          Seltos             Kona         Accent Carnival / Sedona
               4                4              7                2
        Palisade       K3 / Forte       Sportage        Telluride
               2                6              1                2
            Soul      K5 / Optima         Tucson          Sorento
               1                6              1                1
          Sonata         Santa Fe       Elantra
               6                1              6
```

위 자료들을 취합하면 다음과 같다.

<표 1-6> 계층 및 비계층 군집분석의 군집 비교

군집 방법	C1	C2	C3	C4	C5	C6	C7
단일연결법	Telluride, Carnival/Sedona, Palisade	Nexo	IONIQ	Niro, Venue, Seltos, Kona, Soul, Tucson, Santa Fe, Sportage, Sorento	Genesis Coupe	Elantra, K3/Forte, K5/Optima, Sonata	G90, Equus, K9, Veloster, Pride/Rio/SC, Accent, G80, Stinger, Azera, G70, Genesis, K7/Cadenza
단일연결법 군집특성	차량인승과 배기량이 높은 RV차량	판매량이 가장 적고 가격이 높은 RV차량	연비가 가장 높은 차량	판매량이 높은 RV차량	차량인승이 가장 적은 차량	판매량이 높은 PC차량	판매량이 낮은 PC차량
완전연결법 및 평균연결법	Telluride, Carnival/Sedona, Palisade	Nexo	IONIQ	Tucson, Santa Fe, Soul, Sportage, Sorento, Elantra, K3/Forte, K5/Optima, Sonata	G90, Equus, K9	Genesis Coupe, Genesis, K7/Cadenza, G80, Stinger, Azera, G70	Veloster, Pride/Rio/SC, Accent, Niro, Venue, Seltos, Kona
완전연결법 및 평균연결법 군집특성	차량인승과 배기량이 높은 RV차량	판매량이 가장 적고 가격이 높은 RV차량	연비가 가장 높은 차량	판매량이 높은 차량	가격이 높아 판매량은 낮지만 출력과 배기량이 높은 프리미엄 PC차량	판매량이 낮고 가격이 낮은 PC차량	출력이 낮은 차량
와드연결법 · K-대표개체법 · Allindex best partition	Telluride, Carnival/Sedona, Palisade	Nexo, Niro, Venue, Seltos, Kona	IONIQ, Veloster, Pride/Rio/SC, Accent	Tucson, Santa Fe, Soul, Sportage, Sorento	G90, Equus, K9	Elantra, K3/Forte, K5/Optima, Sonata	Genesis Coupe, Genesis, K7/Cadenza, G80, Stinger, Azera, G70,
와드연결법 · K-대표개체법 · Allindex best partition 군집특성	차량인승과 배기량이 높은 RV차량	판매량이 적은 소형 RV차량	가격이 낮고 연비가 높은 PC차량	판매량이 높은 RV차량	가격이 높아 판매량은 낮지만 출력과 배기량이 높은 프리미엄 PC차량	판매량이 높은 PC차량	판매량이 낮고 가격이 낮은 PC차량
K-평균법	Telluride, Carnival/Sedona, Palisade	Nexo	IONIQ, Veloster, Pride/Rio/SC, Accent	Venue, Niro, Seltos, Kona	Sportage Soul Tucson Sorento Santa Fe	K3/Forte, K5/Optima, Sonata, Elantra	Equus, Genesis Coupe, K9, Azera, G70, Genesis, K7/Cadenza, G90, Stinger, G80
K-평균법 군집특성	차량인승과 배기량이 높은 RV차량	판매량이 가장 적고 가격이 높은 RV차량	가격이 낮은 PC차량	판매량이 적은 소형 RV차량	판매량이 높은 RV차량	판매량이 높은 PC차량	판매량이 적고 가격이 높은 PC차량

결과를 정리한 표를 참고하면, 그룹을 7가지로 나눌 수 있으며 각각 그룹은 앞의 PCA와 FA와 비슷하게 판매량, 연비, 차량인승에 따라 군집을 분류하고 있다. 가격이 차종끼리 비교했을 때 RV차량이 PC차량보다는 판매율이 높았고 가격이 높은 프리미엄 차량의 경우에 판매율이 낮았다. 판매율이 높은 RV차량은 연비가 30초반 가격은 2만 달러 초반, 출력은 18HP초반 배기량은 1780cc의 특성을 보인다. 판매율이 높은 PC차량은 연비가 30후반 가격은 2만 달러 초반, 출력은 180HP초반, 배기량은 1700cc의 특성을 보인다. 판매율이 낮은 PC 차량은 다른 PC차량과의 군집들과 비교했을 때 출력과 배기량이 좋지만 가격이 상대적으로 높았고 연비가 좋지 않았다. 판매율이 낮은 RV 차량도 마찬가지로 다른 RV차량과 비교해보았을 때 가격이 높거나 출력량과 배기량이 다른 RV 차량들에 비해 낮았다.

5. Conclusion

1. 미국 소비자들의 현대 · 기아자동차에 대한 소비 경향

PCA, FA를 통하여 변수들의 관계를 알아봤을 때 출력과 배기량, 가격, 차량종류가 판매성적에 영향을 미친다는 것을 알 수 있었다. CA를 통하여 나눈 7가지의 군집 대한 특징을 비교해본결과 RV차량은 30HWYMPG, 180HP, 1780cc의 스펙을 가진 2만 달러 초반의 차량이 강세를 보였고 PC차량은 30HWYMPG, 180HP, 1700cc의 스펙을 가진 2만 달러 초반의 차량이 강세를 보인다. RV차량은 출력량과 배기량이 판매량에 민감한 반응을 보였고 PC차량은 연비가 판매량에 민감한 반응을 보였다. 또한 두 종류의 차량 모두 준중형~준형(1500cc~2000cc)차량에 대한 선호도가 높았으며 가격 또한 판매량에 크게 영향을 미치고 있음을 보이고 있다. 또한 전반적으로 RV차량의 성적이 PC차량의 판매성적보다는 조금 더 안정적으로 RV차량에 대한 미국 내 소비자들의 선호도가 점점 높아지고 있다.

2. 현대 · 기아자동차의 출시예정 차량의 성적 기대

2020년 출시 예정인 차량들을 살펴보면 제네시스의 RV차량인 GV80과 GV70, 현대자동차의 RV차량 2020 산타페와 연식변경예정이 있는 팰리세이드, 기아자동차의 2021 카니발 등 다양한 RV차량이 대기 중이다. 미국 내 소비자들의 RV차량의 선호도가 높아지고 있는 현재 소비자의 선호도가 반영된 사항으로 유추해 볼 수 있다. 특히 이들 중에서 2020 산타페는 RV차량으로 판매율이 높은 RV차량의 특징과 가장 유사하고, 이전 모델에 비해 추가된 휠과 리플렉터 밴드, 버튼식 전자 변속기 탑재 등으로 소비자들의 가장 큰 관심을 이끌 수 있을 것으로 기대가 된다.

3. 현대 · 기아자동차의 미국시장에서의 방향성

현재 미국 내 소비자들의 RV차량의 선호도가 높아지고 있어 점차 RV차량의 판매성적이 전체의 판매성적에 큰 영향을 미칠 것이다. 하지만 현대 · 기아자동차의 판매성적에 판매성적이 감소하고 있는 PC차량들도 판매량의 큰 부분을 차지하고 있다. 게다가 제네시스 PC차량들의 부진과 Sonata와 Elantra의 판매성적이 감소하고 있는 상황에서 RV차량에만 의존하기보다는 PC차량의 성능 개선 및 디자인, 가격조정 등으로 PC차량의 판매량을 반드시 유지하여야만 한다. 또한 앞선 분석의 결과로 보여진 30HWYMPG, 180HP, 1700cc의 스펙에 2만 달러 초반의 가격을 기준으로 하여 판매성적이 높은 PC차량인 Sonata와 Elantra의 성능을 더 높인다면 높은 품질에 싼 가격으로 판매성적을 높일 수 있을 것이다. 이로 인하여 제네시스 차량에 대한 기대감도 불러올 수 있을 것이다. 그러기 위해선 제네시스 엔진결함문제, 기아자동차의 급발진문제, 현대자동차의 아반떼 제동 장치 결함 문제 등 품질적인 문제에서 결함을 꼭 바로 잡아야 할 것이다. 그렇게 된다면 현대자동차가 세계적인 자동차 시장에서 도요타, 벤츠, BMW등과 같은 세계적인 자동차 브랜드와 견줄 수 있을 것이다. 나아가 국내의 삼성, LG와 협력하여 배터리 문제를 바로 잡고 수소전기 자동차로 시장을 선도할 수 있을 것이다.

<그림 1-26> 2021 카니발

<그림 1-27> 2020 산타페

<그림 1-28> GV70

<그림 1-29> 정의선 부회장의 회동

6. References

현대자동차 그룹
https://www.hyundai.co.kr/Index.hub

현대자동차 주요판매실적
https://www.hyundai.com/kr/ko/company-intro/ir-information/sales/sales-record

인터브랜드 '2019년 베스트 글로벌 브랜드'
https://www.interbrand.com/kr/newsroom/best-global-brands-2019/

기업진단포털
https://www.egroup.go.kr/egps/wi/mainPage.do

네이버 자동차
https://auto.naver.com/index.nhn

현대자동차 USA
https://www.hyundaiusa.com/us/ko/vehicles

제네시스 USA
https://www.genesis.com/us/ko/genesis.html

2016 Azera 2016 Genesis 2016 Equus 자료
https://www.khaiyang.com/2982

Best cars U.S.News
https://cars.usnews.com/cars-trucks

기아자동차 주요판매실적
http://pr.kia.com/ko/company/ir/ir-library/sales-results.do

KIA USA
https://www.kia.com/us/en

7. R-code

```
install.packages('readxl')
library(readxl)
h_export<-read_excel("C:/Users/Administrator/Desktop/
현대기아 국가별 판매량.xlsx",
          sheet="rdata",
          range="A1:C49",
          col_names=TRUE,
          col_types="guess",
          na = "NA")
h_export

install.packages('ggplot2')
library(ggplot2)
ggplot(data=h_export, aes(x=year, y=data, colour=Region,
group=Region)) +
          geom_line() +
          geom_point(size=3) +
          ggtitle("현대기아 국가별 판매량")

h_usa_year<-read_excel("C:/Users/Administrator/Deskto
p/현기차 미국 제품별 판매현황.xlsx",
          sheet="rdata1",
          range="A1:D187",
          col_names=TRUE,
          col_types="guess",
          na = "NA")
h_usa_year

ggplot(data=h_usa_year, aes(x=year, y=data,
colour=model, group=model)) +
          geom_line() +
          geom_point(size=3) +
          ggtitle("현대기아 차종별 판매량")

h_usa<-read_excel("C:/Users/Administrator/Desktop/현기
차 미국 제품별 판매현황.xlsx",
          sheet="rdata2",
          range="A1:I32",
          col_names=TRUE,
          col_types="guess",
          na = "NA")
h_usa<-data.frame(h_usa)
h_usa
rownames(h_usa)<-h_usa[,1]
h_usa<-h_usa[,-1]
X<-h_usa
class(X)
X<-as.matrix(h_usa)

##############################################
boxplot(X)
n<-nrow(X)
p<-ncol(X)
I<-diag(n)
J<-matrix(1,n,n)
H<-I-1/n*J                    # 중심화행렬-원래의
데이터에서 평균을 뺀것
Y<-H%*%X                      # 중심화 자료행렬
S<-t(Y)%*%Y/(n-1)             # 공분산행렬
D<-diag(1/sqrt(diag(S)))      # 표준편차행렬의 역수 (1/s_ii)
Z<-Y%*%D                      # 표준화자료행렬
rownames(Y)<-rownames(X)
rownames(Z)<-rownames(X)
colnames(Y)<-colnames(X)
colnames(Z)<-colnames(X)
xbar<-colMeans(X) # 평균벡터
S<-cov(X)# 공분산행렬
R<-cor(X)# 상관행렬
head(Y)
head(Z)
Y;Z;S;R
detS <- det(S)
detR <- det(R)
trS <- sum(diag(S))
trR <- sum(diag(R))
detS; # 데이터의 일반화 분산
detR; # 상관행렬의 일반화 분산
trS; # 데이터의 총 분산
trR; # 상관행렬의 총 분산

##############################################
Zbar=colMeans(Z)
m<-mahalanobis(Z, Zbar, R)
m<-sort(m)
id<-seq(1, n)
pt<-(id-0.5)/n
q<-qchisq(pt, p)
plot(q, m, pch="*", xlab="Quantile", ylab="Ordered
Squared Distance")
abline(0, 1)
rq<-cor(cbind(q, m))[1,2]
rq
```

```
X<-h_usa
class(X)
X<-as.matrix(h_usa)
X

library(MVN)

up=X[which(X[,8]==1),]
do=X[which(X[,8]==-1),]
up=up[,1:7]
do=do[,1:7]

result_up = mvn(up)
result_do = mvn(do)

result_up
result_do
mvn(X)

##########################################
X=up
n<-nrow(X)
p<-ncol(X)
I<-diag(n)
J<-matrix(1,n,n)
H<-I-1/n*J                    # 중심화행렬-원래의
데이터에서 평균을 뺀것
Y<-H%*%X                      # 중심화 자료행렬
S<-t(Y)%*%Y/(n-1)             # 공분산행렬
D<-diag(1/sqrt(diag(S)))      # 표준편차행렬의 역수 (1/s_ii)
Z<-Y%*%D                      # 표준화자료행렬
rownames(Y)<-rownames(X)
rownames(Z)<-rownames(X)
colnames(Y)<-colnames(X)
colnames(Z)<-colnames(X)
xbar<-colMeans(X) # 평균벡터
S<-cov(X)# 공분산행렬
R<-cor(X)# 상관행렬
head(Y)
head(Z)
Y;Z;S;R
detS <- det(S)
detR <- det(R)
trS <- sum(diag(S))
trR <- sum(diag(R))
detS; # 데이터의 일반화 분산
detR; # 상관행렬의 일반화 분산
trS; # 데이터의 총 분산
trR; # 상관행렬의 총 분산

Zbar=colMeans(Z)
m<-mahalanobis(Z, Zbar, R)
m<-sort(m)
id<-seq(1, n)
pt<-(id-0.5)/n
q<-qchisq(pt, p)
plot(q, m, pch="*", xlab="Quantile", ylab="Ordered
Squared Distance")
abline(0, 1)
rq<-cor(cbind(q, m))[1,2]
rq

##################################
X=do
n<-nrow(X)
p<-ncol(X)
I<-diag(n)
J<-matrix(1,n,n)
H<-I-1/n*J                    # 중심화행렬-원래의
데이터에서 평균을 뺀것
Y<-H%*%X                      # 중심화 자료행렬
S<-t(Y)%*%Y/(n-1)             # 공분산행렬
D<-diag(1/sqrt(diag(S)))      # 표준편차행렬의 역수 (1/s_ii)
Z<-Y%*%D                      # 표준화자료행렬
rownames(Y)<-rownames(X)
rownames(Z)<-rownames(X)
colnames(Y)<-colnames(X)
colnames(Z)<-colnames(X)
xbar<-colMeans(X) # 평균벡터
S<-cov(X)# 공분산행렬
R<-cor(X)# 상관행렬
head(Y)
head(Z)
Y;Z;S;R
detS <- det(S)
detR <- det(R)
trS <- sum(diag(S))
trR <- sum(diag(R))
detS; # 데이터의 일반화 분산
detR; # 상관행렬의 일반화 분산
trS; # 데이터의 총 분산
trR; # 상관행렬의 총 분산

Zbar=colMeans(Z)
m<-mahalanobis(Z, Zbar, R)
m<-sort(m)
id<-seq(1, n)
pt<-(id-0.5)/n
q<-qchisq(pt, p)
plot(q, m, pch="*", xlab="Quantile", ylab="Ordered
```

```
Squared Distance")
abline(0, 1)
rq<-cor(cbind(q, m))[1,2]
rq

cov.Mtest=function(x,ina,a=0.05){
  ## x is the *data set*
  ## ina is a *numeric vector* indicating the groups
of the data set #그룹데이터를 numeric vector형태로
변환해서 사용
  ## a is the significance level, set to 0.05 by default
  x=as.matrix(x)
  p=ncol(x) ## dimension of the data set
  n=nrow(x) ## total sample size
  k=max(ina) ## number of groups
  nu=rep(0,k) ## the sample size of each group will be
stored here later
  pame=rep(0,k) ## the determinant of each covariance
will be stored here
  ## the next "for" function calculates the covariance
matrix of each group
  nu=as.vector(table(ina))
  mat=mat1=array(dim=c(p,p,k))
  for (i in 1:k) {
    mat[,,i]=cov(x[ina==i,])
    pame[i]=det(mat[,,i]) ## the detemirnant of each
covariance matrix
    mat1[,,i]=(nu[i]-1)*cov(x[ina==i,]) }
  ## the next 2 lines calculate the pooled covariance
matrix
  Sp=apply(mat1,1:2,sum)
  Sp=Sp/(n-k)
  for (i in 1:k)
    pamela=det(Sp) ## determinant of the pooled
covariance matrix
  test1=sum((nu-1)*log(pamela/pame))
  gama1=(2*(p^2)+3*p-1)/(6*(p+1)*(k-1))
  gama2=(sum(1/(nu-1))-1/(n-k))
  gama=1-gama1*gama2

  test=gama*test1 ## this is the M (test statistic)
  df=0.5*p*(p+1)*(k-1) ## degrees of freedom of the
chi-square distribution
  pvalue=1-pchisq(test,df) ## p-value of the test
statistic
  crit=qchisq(1-a,df) ## critical value of the chi-square
distribution
  list(M.test=test,degrees=df,critical=crit,p.value=pvalue) }

X<-h_usa
class(X)
X<-as.matrix(h_usa)
X

ina=as.numeric(as.factor(X[, 8]))
x=X[, 1:7]
cov.Mtest(x, ina)
# p.value가 0.05보다 작으므로 귀무가설 기각 분산이 이질
# 다변량 정규성 만족, 공분산행렬 동질성 성립하지 않으므로
QDA 실시

QDA=qda(유형~., data=X, prior=c(18,13)/31)
QDA
X=data.frame(X)

qcluster=predict(QDA, X)$class
qct=table(X$유형, qcluster)
qct

(1-mean(X$유형==qcluster))*100

QDA=qda(유형~., data=X, prior=c(18,13)/31, CV=TRUE)
QDA
confusion=table(X$유형, QDA$class)
confusion

# Expected actual error rate : EAER
EAER=(1-sum(diag(prop.table(confusion))))*100
EAER

####################################
#비계량형 MDS: 데이터가 순서척도 인 변수를 가지는 경우
사용
X<-scale(as.matrix(X))
D <-as.matrix(dist(X, method="euclidean", diag=T))
car=colnames(D)

# Metric MDS
con<-cmdscale(D, k=2, eig=T)
con
con$eig
which(con$eig<0)


# 음수의 개수가 전체의 1/3보다 크므로 음수인 eigenvalue가
많다. 비계량형 mds를 적용
# Nonmetric MDS
library(MASS)
con<-isoMDS(as.matrix(D), k=2)
```

```
con
x<-con$points[,1]
y<-con$points[,2]
lim1<-c(-max(abs(x)), max(abs(x)))
lim2<-c(-max(abs(y)), max(abs(y)))
plot(x,y, xlab="Dim1", ylab="Dim2", xlim=lim1, ylim=lim2)
text(x+0.5,y,car, cex=0.8, pos=2)
abline(v=0, h=0)

# Shepard  Diagram
dist_sh <- Shepard(D[lower.tri(D)], con$points)
dist_sh
cdist_sh=cbind(dist_sh$x, dist_sh$y, dist_sh$yf)
cdist_sh
plot(cdist_sh[,1], cdist_sh[,3], pch = ".", xlab =
"Dissimilarity", ylab = "Distance",
      xlim = range(cdist_sh[,1]), ylim =
range(cdist_sh[,1]))
lines(cdist_sh[,1],  cdist_sh[,3], type = "S")

# Image Diagram
plot(cdist_sh[,2], cdist_sh[,3], pch = ".", xlab =
"FitDissimilarity", ylab = "Distance",
      xlim = range(cdist_sh[,2]), ylim =
range(cdist_sh[,2]))
lines(cdist_sh[,2],  cdist_sh[,3], type = "p")


####################################
saleup=length(which(X[,3]>10))
saledown=length(which(X[,3]<10))
saleup
saledown
cedan=length(which(X[,8]==1))
suv=length(which(X[,8]==-1))
cedan
suv
a=ifelse(X[,3]>=10,1,0)
O=table(X[,8],a)

row.names(O)=c("suv","cedan")
colnames(O)=c("saledown","saleup")
O
chisq.test(O)

F <- O/sum(O)
r <- apply(F,1,sum)
c <- apply(F,2,sum)
r;c;

Dr<- diag(1/sqrt(r))
Dc<- diag(1/sqrt(c))
Dr;Dc
cF<- F-r%*%t(c)
cF
Y <- Dr%*%(cF)%*%Dc
svd.Y <- svd(Y)
U <- svd.Y$u
V <- svd.Y$v
D <- diag(svd.Y$d)

A <- (Dr%*%U%*%D)[,1:2]
B <- (Dc%*%V%*%D)[,1:2]
rownames(A) <- c("suv","cedan")
rownames(B) <- c("saledown","saleup")
A;B

eig <- (svd.Y$d)^2
per <- eig/sum(eig)*100
gof <- sum(per[1:2])
rbind(round(eig, 3),round(per, 3))

par(pty="s")
lim <-range(-1:1)
plot(B[, 1:2],
xlab="Dim1(100%)",ylab="Dim2(0.064%)",xlim=lim,ylim=lim,
pch=15,col=2,
      main="SCRA Algorithm : 이원분할표")
text(B[, 1:2],rownames(B),cex=0.8,col=2,pos=3)
points(A[, 1:2],pch=16, col=4)
text(A[, 1:2],rownames(A),cex=0.8,pos=3, col=4)
abline(v=0,h=0)


########################################
eigen.R<-eigen(R)
D_R<-round(eigen.R$values,2)
V_R<-round(eigen.R$vectors,2)

D_R
V_R

gof_R<-D_R/sum(D_R)*100
round(gof_R,2)
plot(D_R, type="b", main="Scree Graph",
xlab="Component Number", ylab="Eigenvalue")

V_R2<-V_R[,1:3]
round(V_R2,2)
```

```
P_R<-Z%*%V_R2
P_R<-as.matrix(P_R)
P_R

par(mfrow=c(2,2))
plot(P_R[,1], P_R[,2], main="Plot of PCs Scores",
xlab="1st PC", ylab="2nd PC")
text(P_R[,1], P_R[,2], labels=rownames(X), cex=0.8,
col="blue", pos=1)
abline(v=0, h=0)

plot(P_R[,1], P_R[,3], main="Plot of PCs Scores",
xlab="1st PC", ylab="3rd PC")
text(P_R[,1], P_R[,3], labels=rownames(X), cex=0.8,
col="blue", pos=1)
abline(v=0, h=0)

plot(P_R[,2], P_R[,3], main="Plot of PCs Scores",
xlab="2nd PC", ylab="3rd PC")
text(P_R[,2], P_R[,3], labels=rownames(X), cex=0,
col="blue", pos=1)
abline(v=0, h=0)

##############################################
pca.R<-princomp(X, cor=T)
round(pca.R$loadings[,1:3],2)
# Principle component biplot (SVD)

par(mfrow=c(2,2))
biplot(pca.R, choices=c(1,2), scale=0,  xlab="1st
PC",ylab="2nd PC", main="1PC vs 2PC biplot function")
abline(v=0, h=0)
biplot(pca.R, choices=c(1,3), scale=0,  xlab="1st
PC",ylab="3nd PC", main="1PC vs 3PC biplot function")
abline(v=0, h=0)
biplot(pca.R, choices=c(2,3), scale=0,  xlab="2st
PC",ylab="3nd PC", main="2PC vs 3PC biplot function")
abline(v=0, h=0)

##############################################
library(psych)
pcfa<-principal(Z, nfactors=3, rotate="varimax")
pcfa
mlfa<-factanal(Z, factors = 3, rotation="varimax")
mlfa

L<-pcfa$loadings[,1:3]
fpc<-pcfa$scores
Psi<-pcfa$uniquenesses
Rm<-R-(L%*%t(L)+diag(Psi))
L
fpc
Psi
Rm

gof<-pcfa$values/ncol(Z)*100
gof

par(pty="s")    # square figure.
lim<-c(min(fpc),max(fpc))

#
plot(fpc[,1], fpc[,2],main=" (a) Factor Scores : f1 and f2",
xlab="f1", ylab="f2",
      xlim=lim, ylim=lim)
text(fpc[,1], fpc[,2], labels=rownames(fpc), cex=0.8,
col="blue", pos=1)
abline(v=0, h=0)
text(pcfa$loadings[,1], pcfa$loadings[,2],
labels=rownames(pcfa$loadings), cex=0.8, col="red",
pos=1)
arrows(0,0, L[,1], L[, 2], col=2, code=2, length=0.1)

#
plot(fpc[,1], fpc[,3],main=" (b) Factor Scores : f1 and f3",
xlab="f1", ylab="f3",
      xlim=lim, ylim=lim)
text(fpc[,1], fpc[,3], labels=rownames(fpc), cex=0.8,
col="blue", pos=1)
abline(v=0, h=0)
text(pcfa$loadings[,1], pcfa$loadings[,3],
labels=rownames(pcfa$loadings), cex=0.8, col="red",
pos=1)
arrows(0,0,pcfa$loadings[,1], pcfa$loadings[,3], col=2,
code=2, length=0.1)

#
plot(fpc[,2], fpc[,3],main="(c) Factor Scores : f2 and f3",
xlab="f2", ylab="f3",
      xlim=lim, ylim=lim)
text(fpc[,2], fpc[,3], labels=rownames(fpc), cex=0.8,
col="blue", pos=1)
abline(v=0, h=0)
text(pcfa$loadings[,2], pcfa$loadings[,3],
labels=rownames(pcfa$loadings), cex=0.8, col="red",
pos=1)
arrows(0,0,pcfa$loadings[,2], pcfa$loadings[,3], col=2,
code=2, length=0.1)
```

```
##############################################
install.packages("NbClust")
library(NbClust)
allindex<-NbClust(Z, distance="euclidean", min.nc = 2,
max.nc = 8,
                  method = "kmeans", index = "all" )
allindex

ds <- dist(Z, method="euclidean")
round(ds, 3)
#단일연결법
sinle=hclust(ds, method="single")
plot(sinle, hang=-1, main="(a) Sinle Linkage")
rect.hclust(sinle,k=7)
#완전연결법
complete=hclust(ds, method="complete")
plot(complete,hang=-1, main="(b) Complete Linkage")
rect.hclust(complete,k=7)
#평균연결법
average=hclust(ds, method="average")
plot(average, hang=-1, main="(c) Average Linkage")
rect.hclust(average,k=7)
#와드연결법
ward=hclust(ds, method="ward.D2")
plot(ward, hang=-1, main="(d) Ward Linkage")
rect.hclust(ward,k=7)

# K-means Method
kmeans <- kmeans(Z, 7) # 7 cluster solution
cluster=data.frame(model,cluster=kmeans$cluster)
C1=cluster[(cluster[,2]==1),]
C2=cluster[(cluster[,2]==2),]
C3=cluster[(cluster[,2]==3),]
C4=cluster[(cluster[,2]==4),]
C5=cluster[(cluster[,2]==5),]
C6=cluster[(cluster[,2]==6),]
C7=cluster[(cluster[,2]==7),]
C1;C2;C3;C4;C5;C6;C7

# Get cluster means
aggregate(X, by=list(kmeans$cluster),FUN=mean)
aggregate(Z, by=list(kmeans$cluster),FUN=mean)


# K-medoids Method
kmedoids <- pam(Z, 7) # 7 cluster solution
cluster=data.frame(model,cluster=kmedoids$cluster)
C1=cluster[(cluster[,2]==1),]
C2=cluster[(cluster[,2]==2),]
C3=cluster[(cluster[,2]==3),]
C4=cluster[(cluster[,2]==4),]
C5=cluster[(cluster[,2]==5),]
C6=cluster[(cluster[,2]==6),]
C7=cluster[(cluster[,2]==7),]
C1;C2;C3;C4;C5;C6;C7

# Get cluster means
aggregate(X, by=list(kmedoids$cluster),FUN=mean)
aggregate(Z, by=list(kmedoids$cluster),FUN=mean)
```