**데이타마이닝(CSI4117-01)**

**최종보고서**

**과목명**  데이타마이닝

**담당교수**  조성배교수님

**학과명**  컴퓨터과학과

**조이름**  한뚝배기조

**조원**  류재훈

배지민

최인영

****



**R E P O R T**

**과목명** OOOO

**담당교수** OOO 교수님

**이름** OOO

**학과명** OOOOO과

**학번**  2000112345

**데이타마이닝(CSI4117-01)**

**최종보고서**

정의한 문제

**사용자의 타자 패턴에 따른 연령대 분류**

수집된 데이터 셋으로부터 사용자의 타자 패턴에 따라 연령대를 분류한다. 분류된 연령대와 해당 사용자의 사용 환경에 따라서 현재 스마트폰 사용자에게 더 편리한 맞춤 환경을 제공하거나(예를 들어 자판 크기 추천 등), 사용자에게 필요한 서비스를 제안하는 등 사용자 분석이 필요한 여러 분야에 활용될 수 있을 것으로 기대된다. 우리는 이번 프로젝트에서 정의한 문제에 대해서 각종 데이터마이닝 기법을 사용해 분류 모델을 구축하고 결과를 도출하였다. 또한 이 결과의 타당성과 사용성 검증을 통해 유의미한 결론을 얻는 것을 목표로 하였다.

프로젝트 진행

이번 텀프로젝트에서 우리는 데이터를 수집한 이후 다음과 같은 단계로 문제를 해결하였다.

1. 데이터 수집

수집된 데이터는 소프트컴퓨팅 연구실에서 제공한 SCLab 어플리케이션에서 얻은 것이다. SCLab 어플리케이션 에서는 입력에 사용하는 손 3종류(왼손, 오른손, 양손)와 입력 자세 4종류(누움, 앉음, 서 있음, 걸음)로 총 12가지 입력 상황에 따른 데이터가 수집된다. 사용자의 연령, 성별 및 스마트폰 사용 경험 등과 같은 정보는 사용자가 직접 입력한다. 위 조건에서 수집된 데이터는 csv 파일 형태로 제공되었다. 앞서 명시한 데이터 수집 환경에서 사용자의 실수에서 비롯된 잘못된 데이터가 일부 존재할 수 있음을 인식하고 이를 바탕으로 프로젝트를 진행하였다.

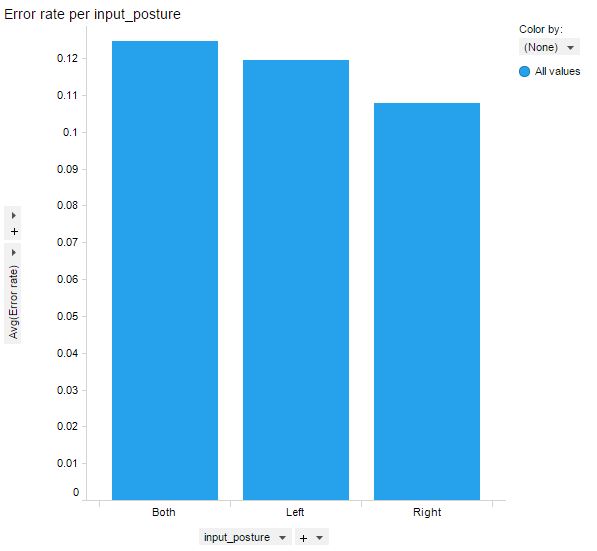
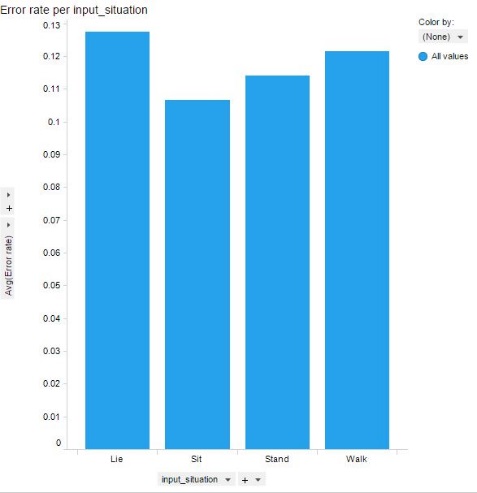
1. 데이터 탐색 & 전처리

데이터마이닝 기법을 원활이 적용할 수 있도록 수집된 데이터를 탐색하고 적절한 전처리를 거친다.

1. 데이터 탐색

수집한 데이터를 더 잘 이해하기 위해서 시각화 등을 이용해 데이터를 탐색한다. 변수 간의 상관관계를 파악하기 위해 histogram을 주로 사용하였다. 탐색 결과는 다음과 같다.

1. 탐색한 변수 간 상관관계

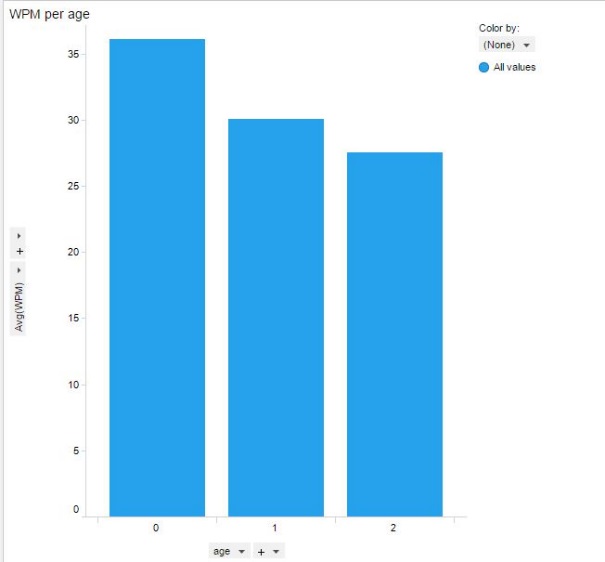
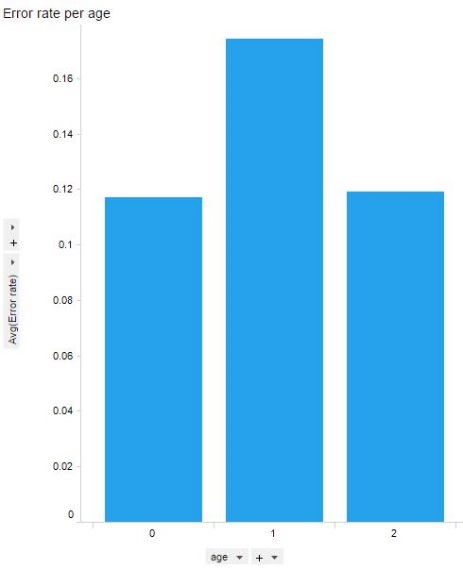
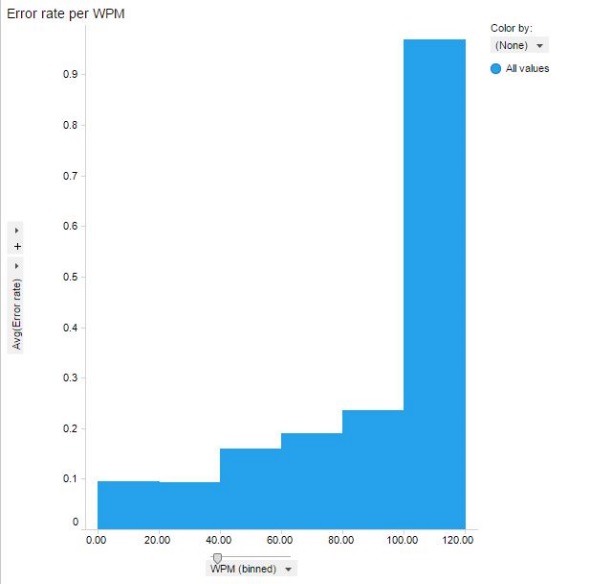


1. 입력 상황에 따른 평균 에러율(Error rate per input\_situation)

앉아있을 경우 가장 에러율이 낮고, 누워있을 때 가장 에러율이 높은 것으로 판단된다.

1. 입력 손에 따른 평균 에러율(Error rate per input\_posture)

오른손으로 입력할 때 가장 에러율이 낮으며, 양손으로 입력할 때 가장 에러율이 높다. 양손으로 입력할 때 가장 에러율이 낮을 것으로 생각되었으나 예상과 달랐다.



1. 입력 속도에 따른 평균 에러율(Error rate per WPM)

입력 속도(WPM)가 빨라질수록 에러율이 높아지는 것으로 보인다.

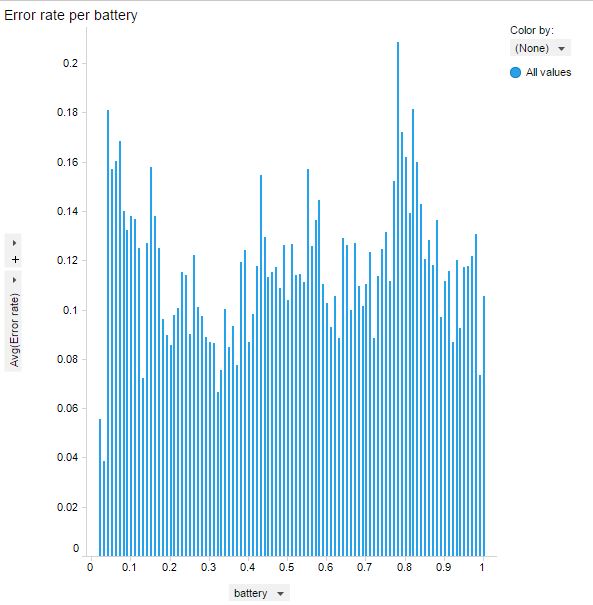
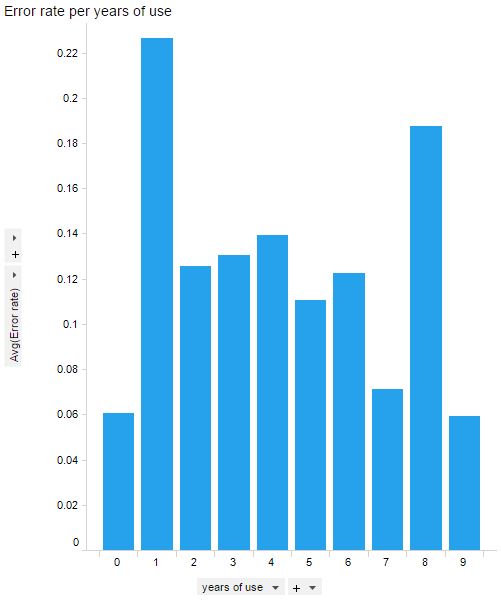
1. 연령대에 따른 평균 에러율(Error rate per age)

장년층(16세~45세)가 가장 에러율이 높다. 다른 두 연령대에 비해서 데이터의 양이 상대적으로 적은 것이 이유가 될 수 있을 것으로도 추정된다.

1. 연령대에 따른 입력 속도(WPM per age)

연령대가 낮을수록 입력 속도가 빠른 것으로 파악되었다. d. 연령대에 따른 평균 에러율에서 중년층(46세~60세)의 에러율은 청년층(16세~30세)과 비슷한 정도이지만, 입력 속도가 느린 것의 영향을 받았을 것으로 보인다. 또한 청년층은 다른 연령층에 비해 입력이 빠르지만 에러율은 가장 낮은 것으로 파악이 되었다.

1. 아래의 변수들은 에러율과 큰 상관관계가 없는 것으로 파악하고 데이터 전처리 과정에서 제외하기로 하였다.

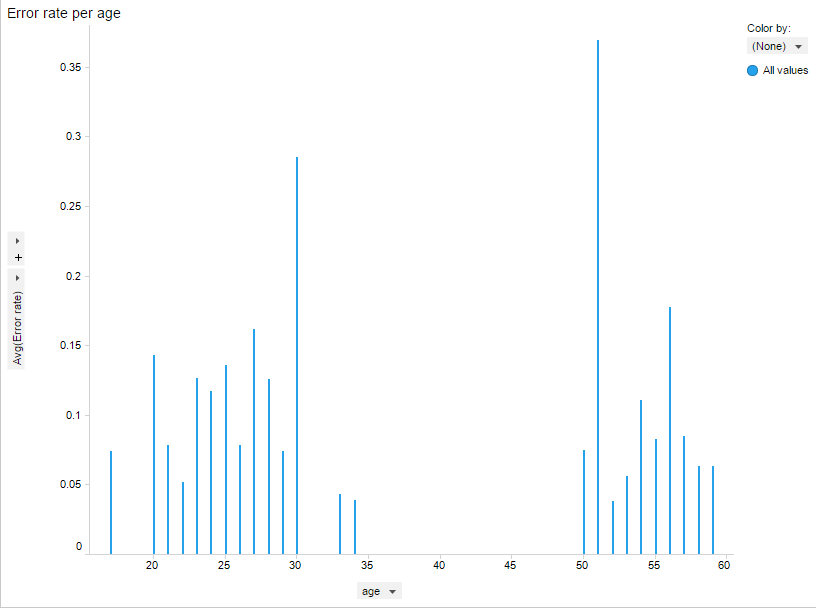


1. 사용 기간에 따른 평균 에러율(Error rate per year of use)
2. 배터리 잔여량에 따른 평균 에러율(Error rate per battery)

두 경우 모두 평균 에러율과 큰 상관관계가 없는 것으로 파악된다. year of use에는 잘못된 데이터가 상당 수 있을 것으로도 추정하였다. 두 경우에서 변수에 따른 높낮이가 있으나, 추세를 파악할 수 없기에 모델 구축에서 제외하기로 결정하였다. (이 외에도 유의미한 상관관계가 없다고 판단된 변수 역시 제외하였다.)

1. 데이터 전처리

1) 정의한 문제의 해결에 앞서 ‘age’ 변수의 값을 범주형으로 변환하였다. 데이터 탐색 결과 연령에 따른 에러율은 아래 [그림2]와 같은 분포를 보였다.

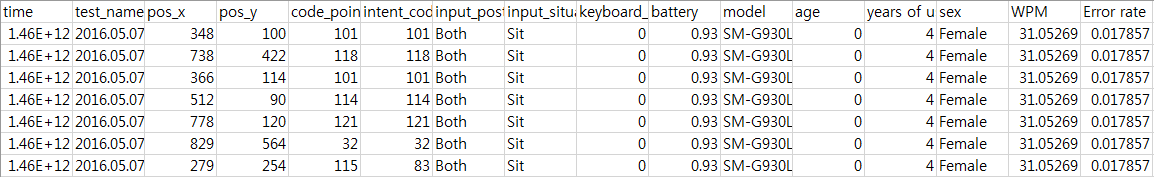


1. **연령에 따른 에러율**

데이터 분포에서 연령 구간을 크게 세 구간, 15~29세(청년), 30~45세(장년), 45~60세로(중년) 으로 나눌 수 있을 것으로 생각되어 연령(age) 변수를 범주형 변수로 변환하였다. 세 구간에서 변수의 값은 각각 0, 1, 2를 가진다.

|  |  |  |
| --- | --- | --- |
| 나이 | 연령대 | 범주형 변수 |
| 15~29 | 청년 | 0 |
| 30~44 | 장년 | 1 |
| 45~60 | 중년 | 2 |

**[표 1] 연령대로의 범주형 변수 변환**

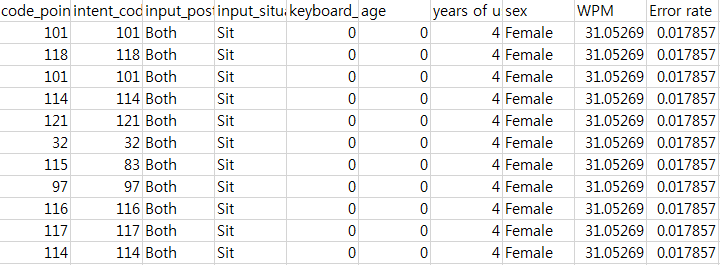


1. **연령대로의 범주형 변수 변환 결과**

2) 정의한 문제를 해결하는 과정에서 불필요한 변수를 제거하였다. 데이터 탐색 결과와 데이터에 대한 팀원들의 지식을 바탕으로 결과값 도출에 유의미하지 않다고 판단한 변수들을 불필요하다고 간주하고 제거하였다. 그 결과 모델 구축에 사용한 변수들은 다음과 같다.

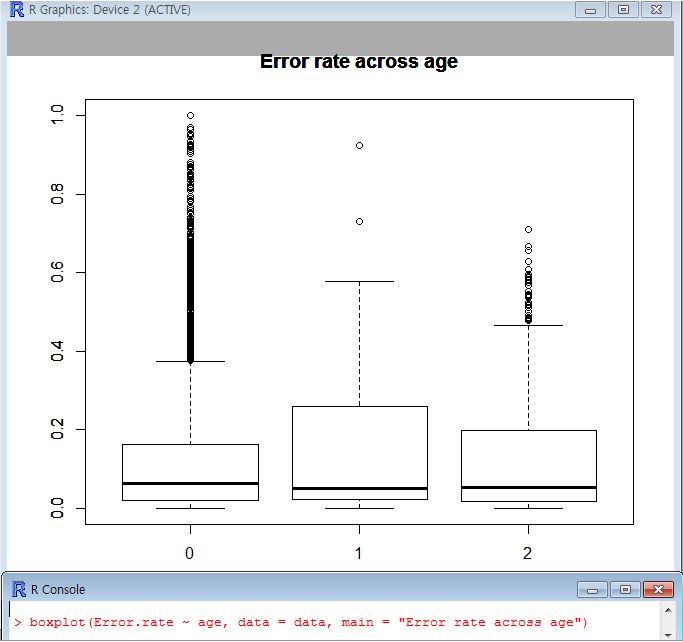
|  |  |
| --- | --- |
| 기존 변수들(불필요한 변수) | 불필요한 변수 제거 후 변수들 |
| (Time)  (test\_name)  (pos\_x)  (pos\_y)  code\_point  intent\_code\_point  input\_posture  input\_situation  keyboard\_condition  (battery)  (model)  age  years of use  sex  WPM  Error rate | code\_point  intent\_code\_point  input\_posture  input\_situation  keyboard\_condition  age  years of use  sex  WPM  Error rate |

**[표 2] 불필요한 변수 제거**



1. **변수 제거 결과**

3) 에러율의 이상치로 판단되는 데이터를 제거하였다. 모델 구축에 사용할 툴인 R에서 Box plot으로부터 이상치를 파악하였고, 사용할 데이터로부터 제외하였다.



1. **연령대 별 에러율의 Box plot**

각 연령층 별 이상치 상한은 음수로 [그림 3]과 같이 이상치가 존재하지 않았다. 이상치는 다음과 같이 구했다.

|  |
| --- |
| IF ( 값 > 제3사분위수 + 1.5\*사분위범위 ) THEN  RETURN 이상치  ELSE  RETURN 보통치 |

여기서 사분위범위는 각 조건에서 사분위3, 사분위 1에서의 값 차이(Q3-Q1)를 뜻한다. 위 기준으로 구한 이상치 범위는 아래와 같다. 아래 범위에 속하는 데이터는 이상치로 간주하고 데이터셋에서 제거하였다.

|  |
| --- |
| 청년층( age = 0 )의 이상치 >= ( 0.16176 + ( 1.5\*0.14298 ) ) = 0.37623  장년층( age = 1 )의 이상치 >= ( 0.25926 + ( 1.5\*0.23752 ) ) = 0.61554  중년층( age = 2 )의 이상치 >= ( 0.19718 + ( 1.5\*0.18023 ) ) = 0.467525 |

1. 데이터 축소 & 분할

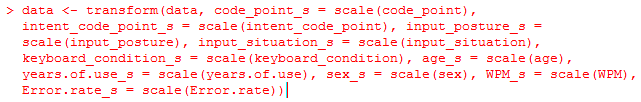
불필요한 변수와 데이터 수를 줄여서 데이터를 단순한 형태로 정제한다. PCA를 사용해 분석할 데이터의 차원을 축소하였으며, 구축한 데이터모델을 검증하기 위해서 주어진 데이터셋을 학습데이터(67), 검증데이터(33)로 분할하였다.

**01. 데이터 축소**

우리는 데이터를 축소하는 방법으로 주성분 분석(Principal Components Analysis, 이하 PCA) 방법을 사용하였다. PCA는 기존 변수들의 선형 결합으로 이루어진 새로운 변수를 생성해서 적은 수의 변수들만 사용하여 전체 데이터를 표현하도록 하는 기법이다. 우리는 다음과 같은 과정으로 PCA를 진행하였다.

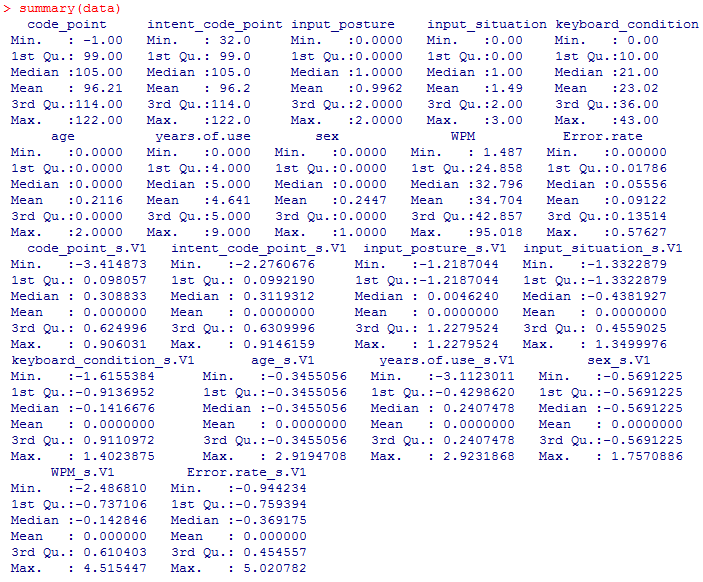
1) 표준화 작업

주성분 분석에서는 변수 별로 단위가 다르면 데이터 왜곡이 생길 수 있기 때문에 표준화 작업을 거칠 필요가 있다. 이 과정에서 transform과 scale이라는 함수를 사용하였다.



1. **표준화에 사용한 R 스크립트**

위와 같이 명령을 R에서 실행해 본 후에는 함수 transform과 scale이 잘 수행되었는지 확인해 볼 필요가 있다. 이 때 아래와 같이 summary 함수를 사용하여 각 변수들의 종류와 그 변수들의 정보를 살펴볼 수 있다. 그 정보들을 살펴보면 표준화가 되어 있지 않은 변수들에 표준화가 된 변수들이 추가된 것을 확인할 수 있다. 또 그 추가된 변수들의 내용을 살펴보면 평균(mean)값이 0으로 설정되고 Min, 1stQu, Median, 3rdQu, Max 값이 각각의 표준편차에 따라 바뀐 것 또한 확인할 수 있다.

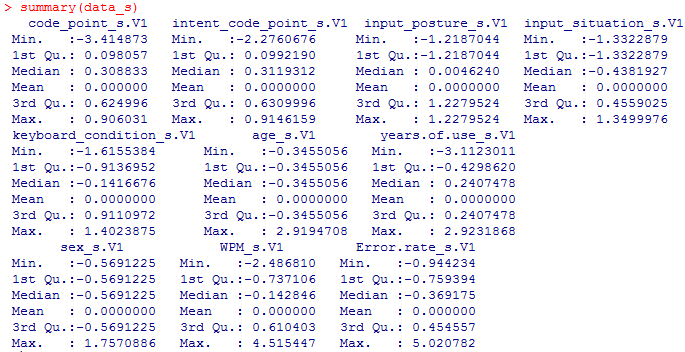


1. **표준화 결과 기존 변수와 새로 생성된 변수의 정보**

이후 새로운 변수 data\_s를 정의해 표준화 되어 있는 데이터만을 분리하였다. 그 과정은 다음과 같다.



1. **새로운 변수 data\_s에 표준화된 데이터만 분리하는 R 스크립트**



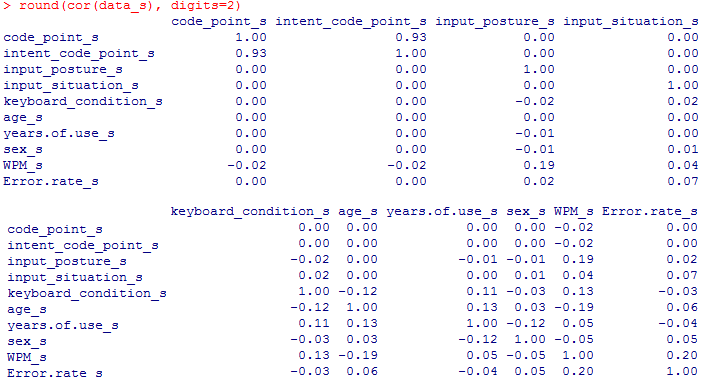
1. **표준화된 데이터**

이후 상관관계 분석과 PCA 과정에서는 여기서 표준화된 데이터를 사용해 진행하였다.

2) 상관계수 분석

주성분 분석을 실시하는 가장 큰 목적은 데이터의 차원을 축소하고자 하는 데에 있다. 이 때 차원을 축소하는 과정은 상관관계가 높은 변수들이 합쳐지면서 이루어진다. 즉, 상관관계가 높은 변수들이 주성분 하나로 축소되어 표현되는 것이다. 따라서 주성분 분석을 하기에 앞서 어떤 변수들이 상관관계가 높은지 확인해보기 위해서 상관관계 분석을 실시해 보았다.

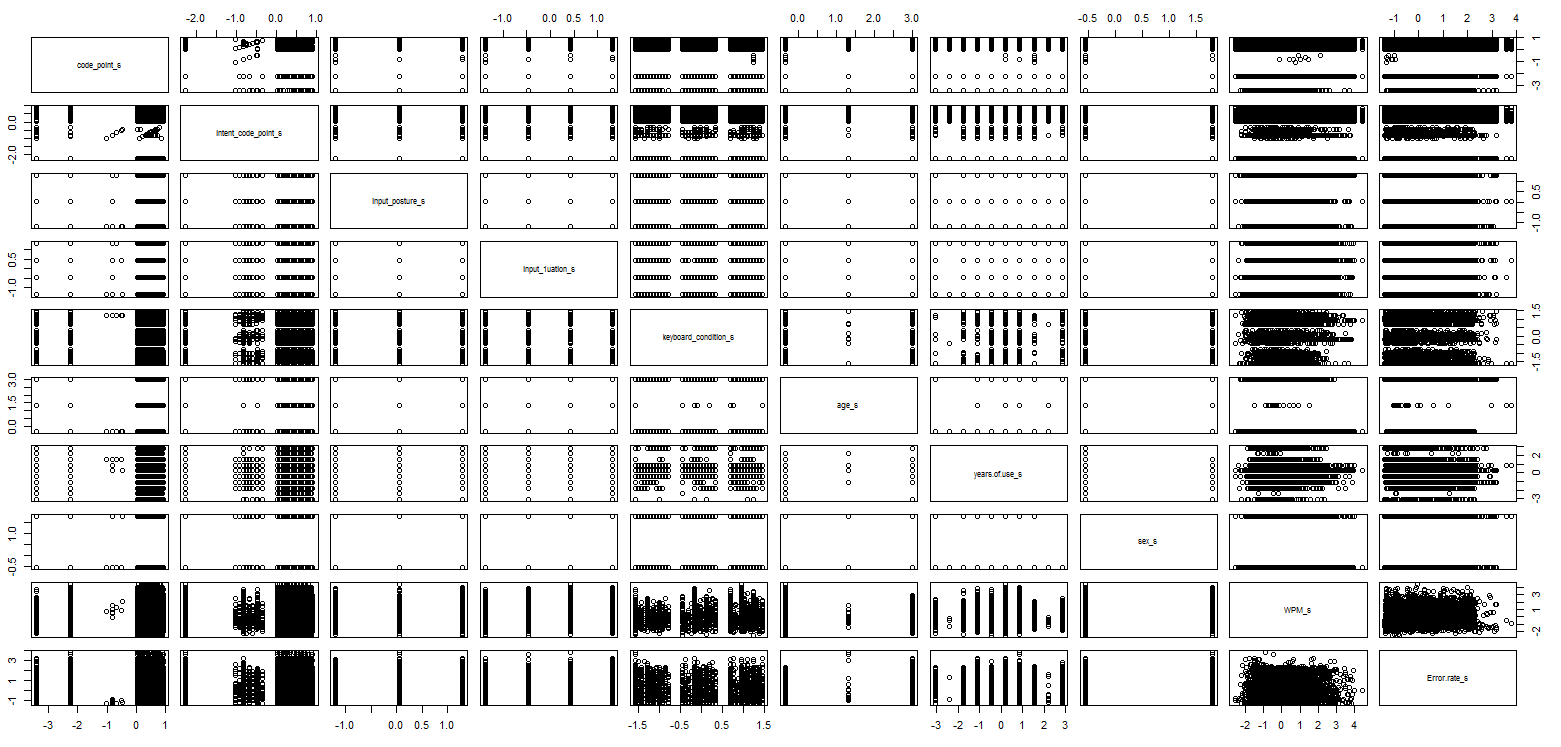
우리는 이 분석을 R에서 제공하는 cor 함수를 사용하여 그 과정을 진행해 보았다. cor 함수를 사용하여 상관관계를 비교해봤을 때 그 결과값이 1에 가까울수록 상관관계가 높고 음수가 나온다면 역의 상관관계가 나온다고 볼 수 있다. 아래 그림은 cor 함수를 사용하여 상관관계를 비교해보는 과정과 그 결과를 나타낸 것이다. 이 때, 상관관계 결과값이 소수점 아래 숫자까지 길게 나타나는 것을 방지하기 위해서 round 함수도 함께 사용하였다. round 함수를 사용하여 소수점 아래 두번째(digits=2) 자리까지만 표시하였다.



1. **변수 간 상관관계 분석 결과**

이를 보면 code\_point\_s(터치한 문자)와 intent\_code\_point\_s(눌러야하는 문자)가 높은 상관관계, age\_s(나이)와 WPM\_s(타자속도)가 역의 상관관계를 가지고 있는 것을 알 수 있었다. 서로 상관관계가 없는 변수들은 cor 함수 결과값도 0에 가깝게 나오는 것 또한 확인할 수 있었다.

이를 시각화를 통해 한 눈에 비교하기 위해 산점도 행렬도 이용해 보았는데 이 때는 R의 plot 함수를 이용하였다. plot 함수를 사용하면 아래와 같이 n\*n 그래프가 나오는데 이 각각의 그래프는 cor 함수값이 1에 가까울수록 y=x 그래프와 가깝게 그려지는 것을 확인할 수 있다.

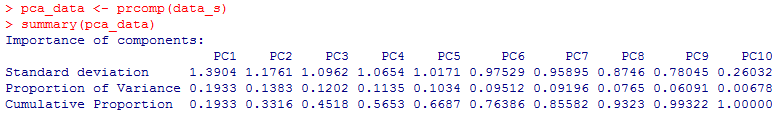


1. **상관관계 산점도 행렬**

위에서 분석한 상관관계를 바탕으로 PCA 과정을 진행하였다.

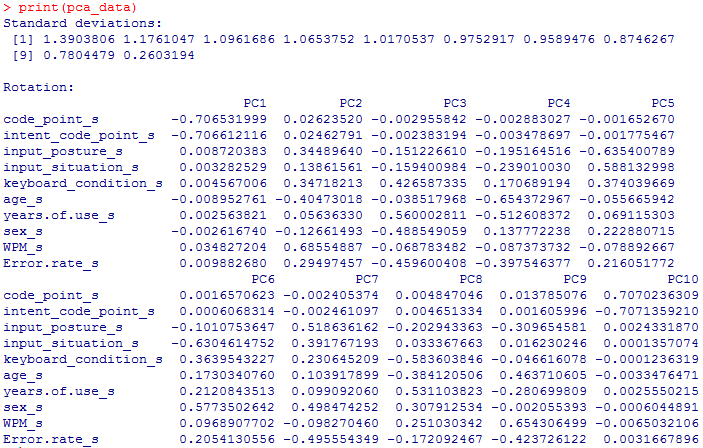
3) 주성분 분석(PCA)

우리는 주성분 분석을 하기 위해서 R의 prcomp 함수를 사용하였다. prcomp는 주성분 분석을 한 결과값을 반환한다. 이 함수의 반환값을 새로운 변수에 저장해 아래와 같이 주성분 분석의 결과를 확인하였다.



1. **주성분 분석(PCA) 결과**

주성분이 PC1부터 PC10까지 각 주성분의 표준편차(Standard deviation), 기여율(Proportion of Variance), 누적 기여율(Cumulative Proportion)까지 나열되어 있다. 이 때 데이터에 영향을 많이 끼치는 순서대로 주성분이 나열된다. 이와 별개로 print 함수를 사용하여 아래 그림과 같이 주성분끼리의 관계도 알아볼 수 있다.

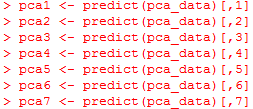


1. **주성분에서 각 변수의 계수**

이 결과를 보면 Rotation 후의 고유벡터의 계수를 확인할 수 있고 이로써, 주성분들과 변수들의 상관관계를 알 수 있다. 이를 통해 어떤 변수들이 주성분들로 축소되어 표시될 지 알 수 있다. 제 1요인(PC1)은 code\_point\_s, intent\_code\_point\_s와 관련(음의 방향)이 있고, 제 2요인(PC2)은 WPM\_s와 관련(양의 방향), age\_s와 관련(음의 방향)이 있음을 알 수 있다. 이와 같은 방법으로 각 주성분들이 어떤 변수들과 관련이 있는지 알 수 있다.

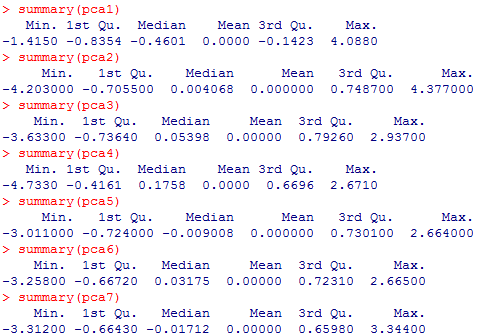
우리는 [그림 11]에서 얻은 주성분들의 누적기여율(cumulative proportion)을 참고하여 주성분의 개수를 선택하였다. 누적기여율이 80퍼센트 이상이면 전체 데이터를 충분히 표현한다고 생각되어 주성분 7개를 사용하기로 하였다. 따라서, summary 함수를 사용해서는 누적기여율을 참고하여 주성분 변수를 몇 개까지 사용할 지를 결정하였고 print 함수를 사용해서는 주성분들과 관련된 변수들이 어떤 것인지를 알아내는 분석을 하였다.

주성분들의 개수와 주성분이 어떤 변수와 관련이 있는지 확인하였으니 이제 그 각각의 주성분들이 어떤 정보를 담고 있는지 확인해 볼 필요가 있다. 이를 수행하기 위해서 우리는 아래와 같이 R의 predict함수를 사용하였다.



1. **주성분 데이터의 확인 및 이동에 필요한 predict 함수**

predict 함수를 사용해서 그 결과값을 각각 pca1, pca2, … , pca7에 저장한 후에는 그 주성분의 내용을 다시 summary 함수를 이용하여 알아볼 수 있다. 그 진행 과정은 아래와 같이 나타내어진다.



1. **각 주성분 내용 확인 결과**

이후 모델 구축에는 앞서 얻은 주성분들을 사용한다.

**02. 데이터 분할**

수집한 데이터에 대한 마지막 전처리 과정으로, 사용할 데이터를 학습 데이터:검증 데이터 = 67:33 의 비율로 나누었다. 분리된 데이터 중 학습 데이터는 각각의 모델을 구축하는 데에 사용되고 검증 데이터는 가장 좋은 모델을 찾아내는 데에 사용된다. 평가데이터는 오로지 최종적으로 결정된 모델의 성능을 평가하는 데에만 쓰일 뿐, 그 최종적인 모델을 보완하거나 교정하는 데에는 쓰이지 않는다. 이번 프로젝트에서 우리는 모델 구축에 있어서 데이터를 학습시키고 검증하는 데에 더 중점을 두기로 하여서 평가데이터를 따로 분리하지는 않았다.

1. 분류 모델 구축과 모델 평가

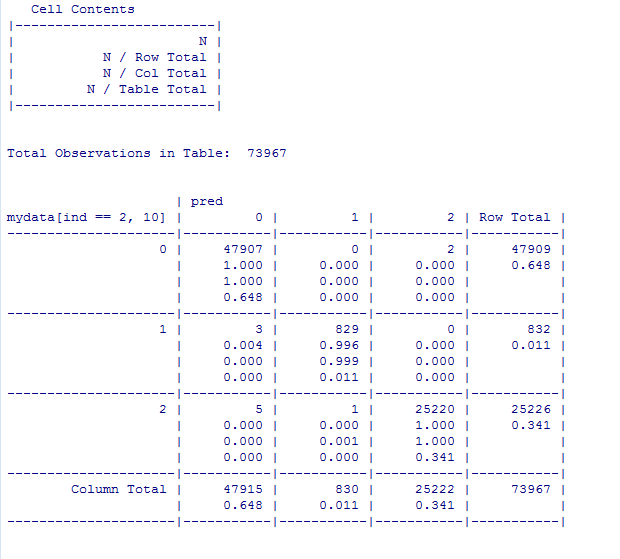
앞서 정의한 대로 ‘연령대’라는 클래스에 새로운 기록을 분류하기 위해서는 분류모델의 구축이 필요하다. 우리는 이러한 모델 구축에 K-최근접 이웃 알고리즘(이하 K-NN)을 사용하기로 결정했다. K-NN 알고리즘이란 분류나 회귀분석에 사용되는 비모수적 기법으로, 입력된 데이터에 가장 근접한 K개의 학습 데이터를 이용하여 분류 및 회귀분석을 수행하는 알고리즘이다.

또한 K-NN은 오직 지역적으로만 근사하고, 모든 계산이 분류 시점까지 연기되는 인스턴스 기반의 학습 방식으로, 모든 기계 학습 알고리즘 중 가장 간단한 알고리즘의 하나이기도 하다. 가장 적절한 K값은 데이터에 의존적이기 때문에 모델마다 다르게 선택되며, 일반적으로 K값이 커질수록 잡음의 영향이 줄어들지만, 항목 간의 경계가 불분명해진다는 단점이 있다. 이 K값은 다양한 휴리스틱 기법들을 이용하여 선정 할 수 있다. 우리는 이러한 K-NN알고리즘을 학습 데이터에 적용하여 분류모델을 구축한 뒤, 검증 데이터를 통해 가장 적합한 모델을 선택 하였다.

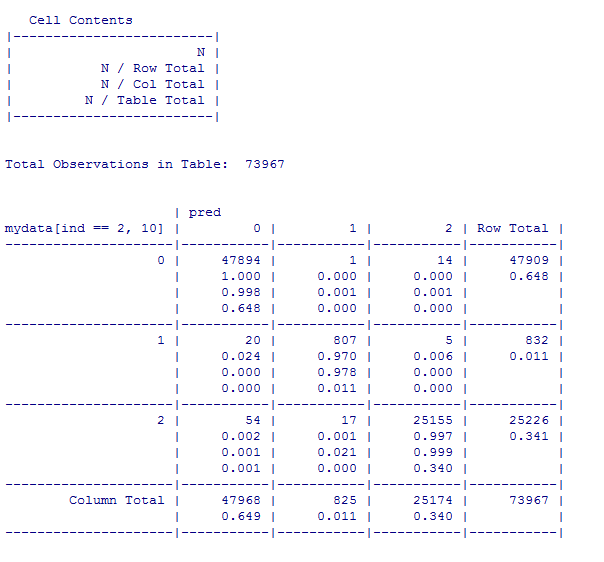
이 때 위에서 언급한 바와 같이 가장 근접한 K개의 학습 데이터에서 다수가 속하는 클래스에 새로운 데이터를 분류한다. 그렇기 때문에 만약 클래스의 개수가 n이라고 할 때, K가 n의 배수일 경우 모든 클래스에 같은 개수의 근접 학습 데이터가 속하게 되는 경우가 발생할 수 있다. 이러한 오류를 발생시키지 않기 위해서는 적절한 K값을 선정하여야 하는데, 우리의 경우 연령대에 3개의 클래스가 존재하기 때문에 3의 배수를 제외하고 K값이 각각 1, 5, 7, 11, 17일 때의 K-NN분류 모델을 구축하여 각각의 모델을 평가했다. 분류 모델을 구축에 R을 사용하였으며, 실제 코드는 다음과 같다.

|  |
| --- |
| **#데이터를 분류하기 위한 랜덤 시드를 생성**  **set.seed(1234)**  **#데이터를 분류하기 위한 index설정**  **ind <- sample(2, nrow(pca\_s), replace=TRUE, prob=c(0.67, 0.33))**  **#PCA결과 생성된 새로운 데이터셋을 각각 학습 데이터와 검증 데이터로 분류**  **data.training <- pca\_s[ind==1,1:7]**  **data.validation <- pca\_s[ind==2, 1:7]**  **#학습 데이터와 검증 데이터 부분에 해당하는 원본 자료의 연령대 값들**  **data.trainLabels <- mydata[ind==1,10]**  **data.validationLabels <- mydata[ind==2,10]**  **#KNN분류를 위한 라이브러리를 include하고, 서로 다른 K값에 대한 분류 실시(K값을 변경시켜가며 반복적으로 수행)**  **library(class)**  **pred <- knn(train = data.training, test = data.validation, cl = as.factor(data.trainLabels), k=11)**  **result <- cbind(pred, mydata[ind==2,10])**  **#분류 결과와 원본 자료의 연령대 값을 비교하여 표로 나타냄**  **library(gmodels)**  **CrossTable(x = mydata[ind==2,10], y = pred, prop.chisq=FALSE)** |

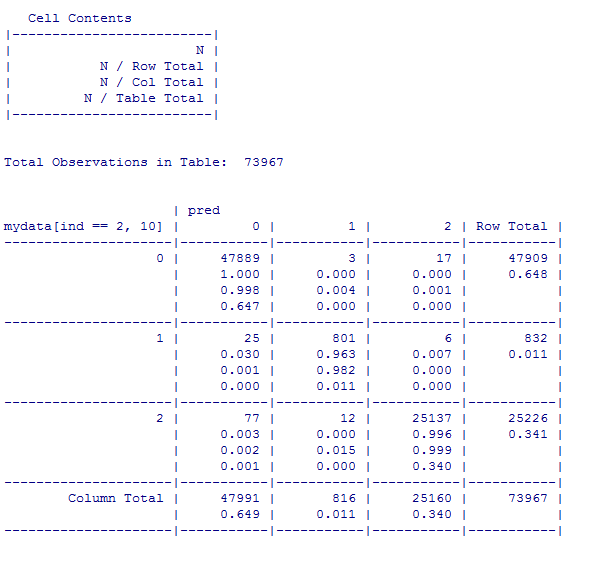
이에 따라 얻어진 검증 데이터의 분류 결과는 각각의 K값에 따라 다음과 같다. 가로 행의 경우 예측된 레코드의 클래스를 나타내며 세로 열의 경우 실제 검증 데이터의 클래스를 나타낸다.



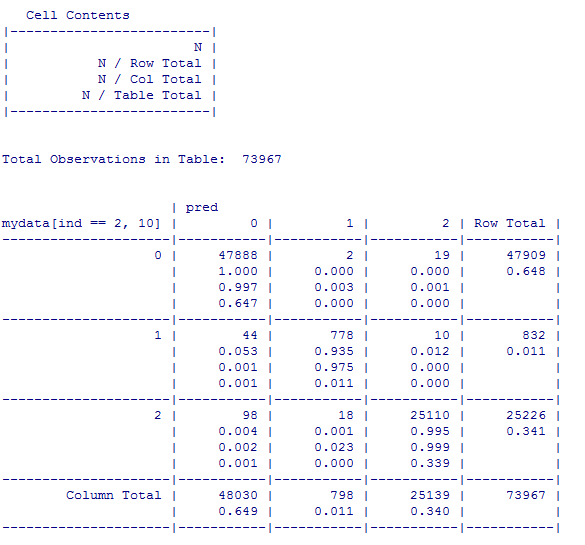
1. **K = 1**



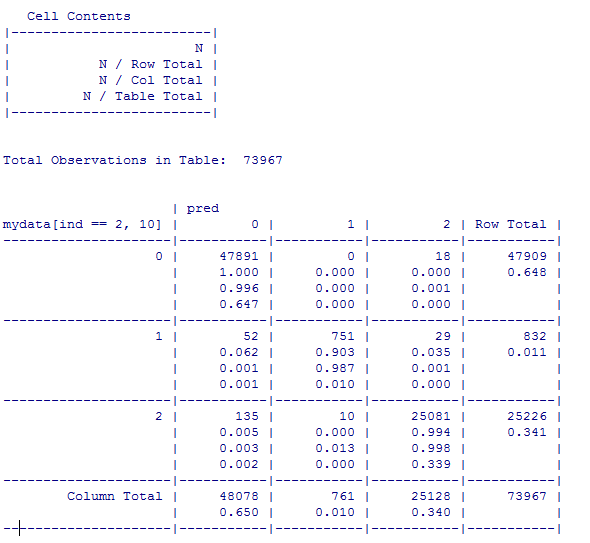
1. **K = 5**



1. **K = 7**



1. **K = 11**



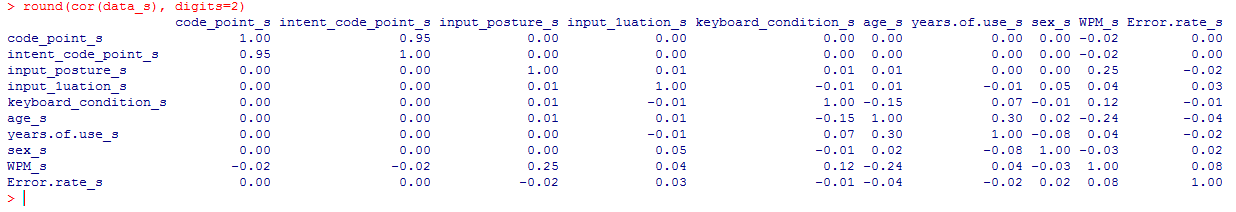
1. **K = 17**

지난번 1차 발표 이전에 테스트 한 결과에서는 K가 1일 때 성능이 가장 좋은 것으로 나타났으며, 잘못 분류한 인스턴스가 0개로 나와 에러율이 완전한 0퍼센트가 나왔었다. 그러나 그 이유를 추가적으로 분석하기 위해 다시 한 번 다양한 K값에 대한 테스트를 진행하니, 데이터를 학습 데이터와 검증 데이터로 분리하는 과정에서 인덱스 설정에 오류가 있어 데이터의 분리가 제대로 이루어지지 않았던 것을 확인할 수 있었다. 그래서 새롭게 제대로 분리된 데이터를 이전과 마찬가지로 0.67대 0.33의 비율로 분리하여 테스트를 진행해보니 여전히 K값이 1일 때 성능이 가장 좋은 것으로 나타났으나 에러율이 완전한 0퍼센트가 아니라 0.015퍼센트로 나타났다. K값을 증가시켜 테스트한 경우 K가 5, 7, 11, 17일 때 각각 에러율이 0.149, 0.185, 0.256, 0.330퍼센트로 점차 증가하는 추세를 보였다.

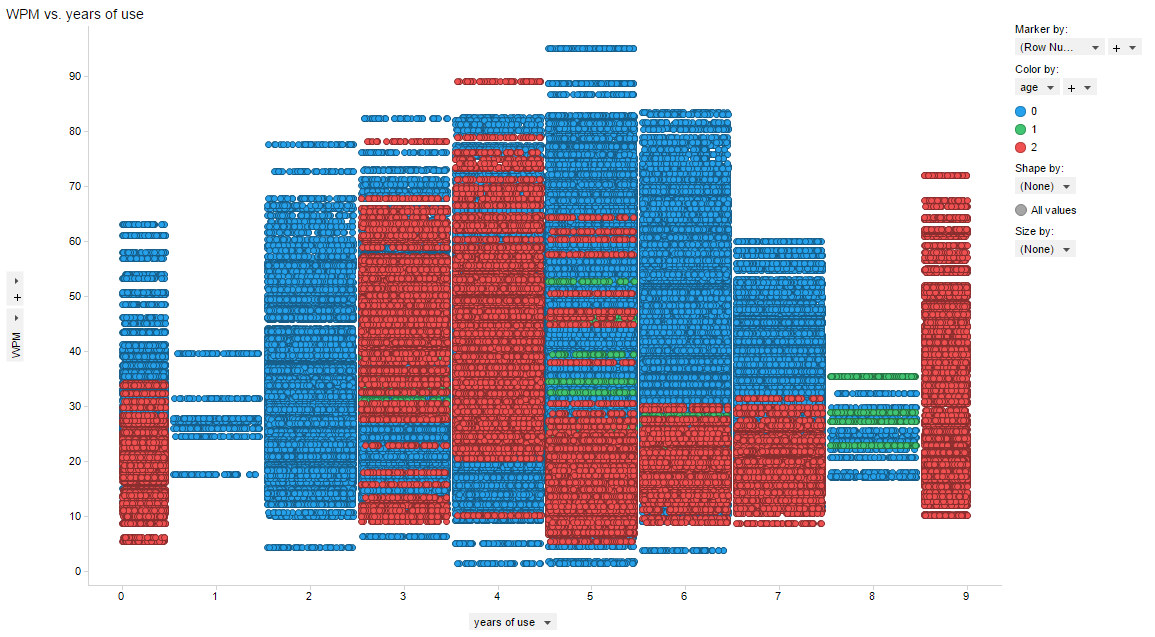
프로젝트 결과 분석

1. 결과의 타당성 및 실용성 검증
2. **결과의 타당성**

분류 모델을 구축한 후, 얻어진 결과에 대한 추가적인 분석을 위하여 K-NN분류법에서 K가 1일 때 가장 성능이 좋게 나오는 경우에 대하여 조사를 시행하였다. 그 결과 세 가지의 경우가 존재할 수 있는 것으로 나타났는데, 첫 번째는 학습 데이터와 검증 데이터가 같을 경우이다. 두 번째 이유로는 학습 데이터와 검증 데이터가 매우 유사한 경우가 있을 수 있고, 세 번째는 각 클래스간의 경계가 뚜렷한 경우이다. 이 세 가지 이유 중, 가장 가능성이 높은 이유를 찾기 위해 추가적인 분석을 실시하였다. 이 때, 1차 발표 이전에 도출한 결과만을 두고 생각했을 때에는 첫 번째 이유 때문에 K값이 1이 나왔을 것이라 예측할 수도 있으나 새로이 데이터를 분리하여 분류 모델을 테스트 했을 때에도 여전히 K값이 1이었기 때문에 학습 데이터와 검증 데이터가 동일하기 때문은 아닐 것이라고 이야기 할 수 있다. 따라서 두 번째와 세 번째 이유에 대한 추가적인 분석을 수행하기 위하여 자료에 대한 상관관계 분석을 실시한 후 연령대와 관련성이 높은 상위 두 변수인 Years of use(사용 기간)와 WPM(분당 입력 단어 수)을 사용하여 산점도를 그린 뒤, 연령대에 따라 다른 색을 주어 보았더니 다음과 같은 결과를 얻을 수 있었다



1. **상관관계 분석 결과**

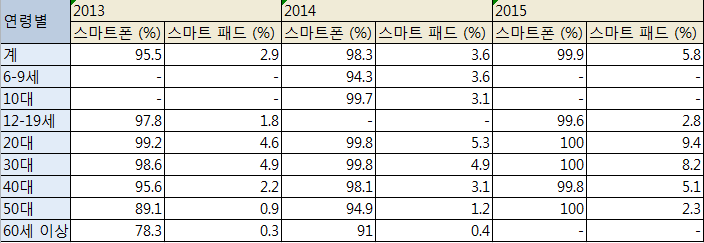


1. **사용 기간(year of use)에 따른 입력 속도(WPM) 산점도**

이 결과에서, 각각의 연령대 클래스가 뚜렷하게 구분되고 있음을 확인할 수 있다. 이를 통해 K값이 1인 결과를 얻은 데에 세 번째 이유가 크게 작용하였음을 알 수 있으며, K-NN 분류 모델을 이용하여 얻은 결과가 타당함을 확인할 수 있다.

**02. 실용성 검증**

현대 사회는 사람들의 평균 수명이 증가함에 따라, 이전보다 다양한 연령대의 고객들이 존재한다. 따라서 회사들은 스스로의 이익을 증대시키기 위해 각 연령대 별 특성을 이해하고, 이에 따른 타겟 마케팅을 실시해야 할 필요성을 느끼고 있다.



1. **국내 스마트 기기 사용자 실태 (KOSIS 제공)**

또한, 한국 국가 통계 포탈인 KOSIS에서 제공하고 있는 통계 자료에 따르면, 국민 대다수가 스마트 기기를 사용하고 있기 때문에 키보드 입력 패턴을 통해 해당 기기 사용자의 연령대를 알 수 있다면, 스마트 기기 내의 어플리케이션이나 메일을 통한 마케팅 자료 전송 등에 유용하게 사용 될 수 있을 것으로 예상된다. 특히나 앞서 검증하였듯이, 연령대 클래스 별로 키보드 입력 패턴에서 뚜렷한 차이를 보이고 있으며 K-NN 모델을 이용한 연령대 분류의 정확도가 높기 때문에 이 모델에 대한 신뢰도가 높다고 할 수 있다. 결과적으로, 우리는 이러한 데이터 분석이 서로 다른 연령대 별 고객을 타겟으로 마케팅을 수행하고자 하는 수많은 회사들에게 고객 분류를 위한 데이터 분석 시간을 줄여줄 수 있는 도구가 될 것으로 예상한다.