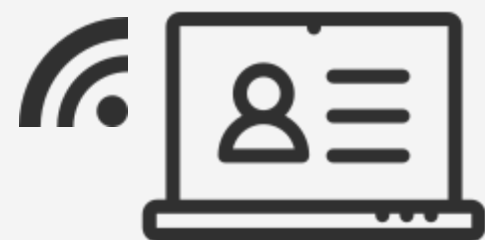




# VEEM VR 데이터를 이용한 예측 및 분류 모델 개발

인공지능응용학과  
오승민, 정다예나, 정지훈, 최예지

---



팀원 간 역할 분담

프로젝트 의사결정 및  
수행 과정



데이터 변수 특성 결정

시선, 몸  
공통



데이터 전처리 과정

특정 전처리 및 분석기법을 사용  
한 근거에 대한 설명



모델 설명

회귀모델  
분류모델



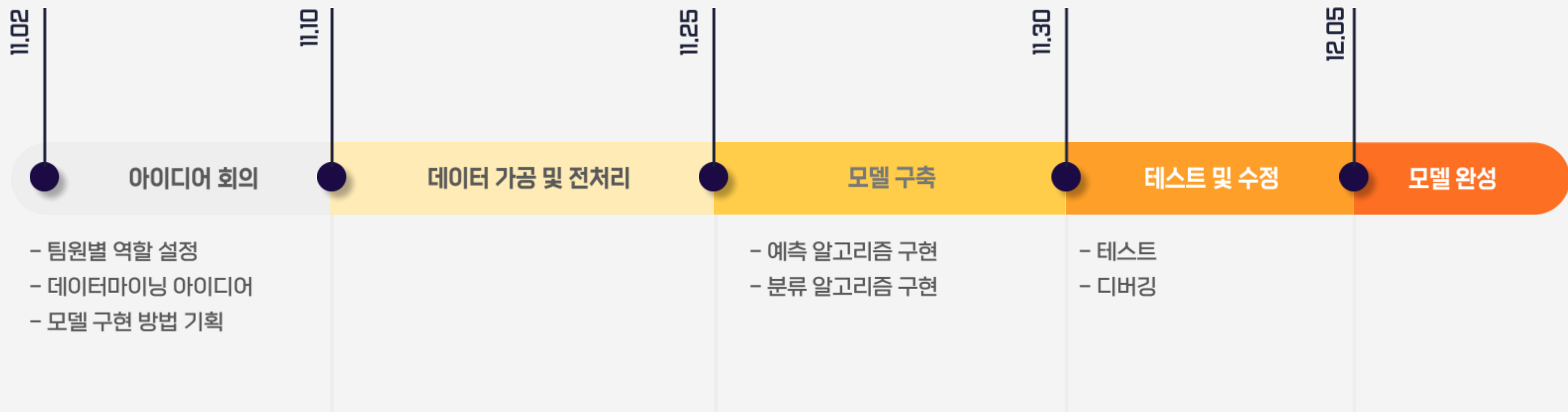
결과 설명

회귀모델  
분류모델

# 01. 팀원 간 역할 분담

| 프로젝트 의사결정 및 수행 과정

| 오승민  | 정다예나  | 정지훈   | 최예지   |
|--|---|---|---|
| <ul style="list-style-type: none"><li>데이터 전처리 코드 구현</li><li>분류 모델 코드 검토 및 조언</li><li>코드 오류 수정</li><li>발표</li></ul> | <ul style="list-style-type: none"><li>데이터 마이닝 아이디어 제공</li><li>데이터 전처리 코드 구현</li><li>분류 모델 코드 구현 및 검토</li><li>발표 자료 제작</li></ul> | <ul style="list-style-type: none"><li>데이터 전처리 코드 구현</li><li>예측 모델 코드 구현 및 검토</li><li>코드 오류 수정</li></ul> | <ul style="list-style-type: none"><li>데이터 마이닝 아이디어 제공</li><li>데이터 전처리 코드 구현</li><li>예측 모델 코드 구현 및 검토</li><li>발표 자료 제작</li></ul> |



## 02. 데이터 변수 특성 결정 | ① 시선 움직임



### 단계 별 소요 시간

- 1단계부터 6단계까지의 소요 시간 (s)

패널 별로 걸린 시간과 **주의집중력**과 관련이 있다고 생각하여 추출  
특히, **DST\_F+B 점수 예측**에 패널 6 소요시간 사용하기 위해 추출



### 이상치 개수 및 비율

- 키오스크 화면을 벗어난 지점을 본 좌표점 개수 및 비율

키오스크화면을 벗어난 지점을 많이 본다는 것은 수  
행과정에서 **주의집중력**이 떨어졌다는 의미로 해석



### 목표 패널 기준 좌표 산만도

- 목표 패널의 중심점을 기준으로 하여 좌표들과의 거리  
측정하여 변수로 지정

산만도와 **기억력**이 관련 있다고 생각하여 변수 추출  
즉, 산만도가 높을수록 기억력이 낮다고 해석

## 02. 데이터 변수 특성 결정 | ② 몸 움직임



### 테스트 동안 움직인 총 거리

- 머리를 움직인 총 거리 구하기(m)
- 손을 움직인 총 거리 구하기(m)
- 각 panel별 움직인 총 거리 구하기(m) - 머리, 손

머리와 손을 얼마나 많이 움직였는지는  
**주의집중력, 산만한 정도**와 관련이 있다고  
생각하여 변수 추출



### 평균 움직임 속도 구하기

- 머리의 움직임 속도(m/s)
- 머리의 움직임 각속도( $\Delta\theta/\Delta t$ )
- 손의 움직임 속도(m/s)

평균적인 움직임 **속도와**  
**수행 능력**이 관련이 있다고 생각하여  
(전체거리/총시간)으로 속도 계산

## 02. 데이터 변수 특성 결정 | ③ 공통 변수 설정



### error 총 횟수

목표 항목을 누르지 않은 횟수는 **주의집중력**,  
**기억력** 등 수행 능력과 관련이 있다고 생각하여 추출



### 뒤로가기 없이 한 번에 맞은 횟수

틀리지 않고 한 번에 넘어간 패널의 개수는  
기억력 등 수행 능력과 관련이 있다고 생각하여 추출



### 테스트 수행하는 데 걸린 시간

총 걸린 시간(s)  
전체적으로 걸린 시간은 K-TMT-E\_B 점수 뿐만 아니라  
**전반적인 수행 능력**과 관련이 있다고 생각하여 추출



### 비밀번호 틀린 횟수

비밀번호를 틀린 횟수는 주의 집중력, 기억력 등  
수행 능력과 관련이 있다고 생각하여 추출

### 03. 데이터 전처리 과정 | 시선 움직임 데이터



#### 단계 별 소요 시간 구하기(s)

'Panel num'열과 'Panel X'열 추출



panel num에 따른 panel X row  
개수 추출



곱 계산으로 단계 별 소요 시간 추출



#### 이상치 개수 및 비율

키오스크 화면의 끝 4점 좌표축 불러오기



Panel X와 Panel Y 정보 불러오기



키오스크 화면 끝 4점 좌표에서 벗어나는 좌표  
추출



좌표 개수 추출하여 전체 row 개수로 나눠주어  
비율 추출



#### 목표 패널 기준 산만도 추출

받은 자료에서 **목표 패널의 중심점**  
좌표 추출



Panel X와 Panel Y 정보 불러오기



패널 별로 나누어 목표 패널의  
**중심점에서의 거리 추출**

### 03. 데이터 전처리 과정 | 몸 움직임 데이터



#### 총 거리 구하기(m)

좌표값 추출

unit 단위를 **meter**로 변환

**distance.euclidean()** 함수를 이용하여  
각 좌표 간의 거리를 구한 뒤,  
모두 합치는 과정 시행

머리, 손, 각 panel별 머리, 손  
모두 동일 방법으로 처리



#### 움직임 속도 구하기(m/s)

계산한 총 거리에서 걸린 시간 나누기

머리, 손 동일 방법으로 처리



#### 머리의 각속도 구하기( $\Delta\theta/\Delta t$ )

각 x,y,z 축별 데이터를 불러와  
만약 0~90도 사이면 +360도 더해준다  
(빠기 연산 용이하게 하기 위함)

각 좌표 간의 각거리를 구하고  
절댓값을 취한 뒤, 모두 합치는 과정 시행

계산한 총 각거리에서 걸린 시간을 나눈 뒤,  
x,y,z에 대해 반복 후 평균 계산



### 03. 데이터 전처리 과정 | 공통 변수 전처리



#### error 총 횟수

error값 열 추출



마지막 최종 error값 추출



#### 뒤로가기 없이 한 번에 맞은 횟수

error열에서 max값 구하기



(비밀번호 틀린 횟수-1)를 빼  
비밀번호 panel 부분 중복 제거



(전체 panel 개수 6개)  
- (틀린 횟수)  
= 한번에 맞은 횟수



#### 비밀번호 틀린 횟수

비밀번호 panel 부분만 추출



(마지막 error값)  
- (처음 error값)  
= 비밀번호 틀린 횟수



#### 총 걸린 시간

100ms당 하나의 row가 기록되므로  
대상자 별 row 개수 구하기



곱 계산으로  
**총 걸린 시간** 구하기



ms 단위를 s로 변환

# VEEM VR 데이터



### 예측 모델

: 사용자 기반 협업 필터링에  
기반한 모델



### 분류 모델

: 랜덤포레스트의  
가장 높은 퍼포먼스를 기반으로 선택

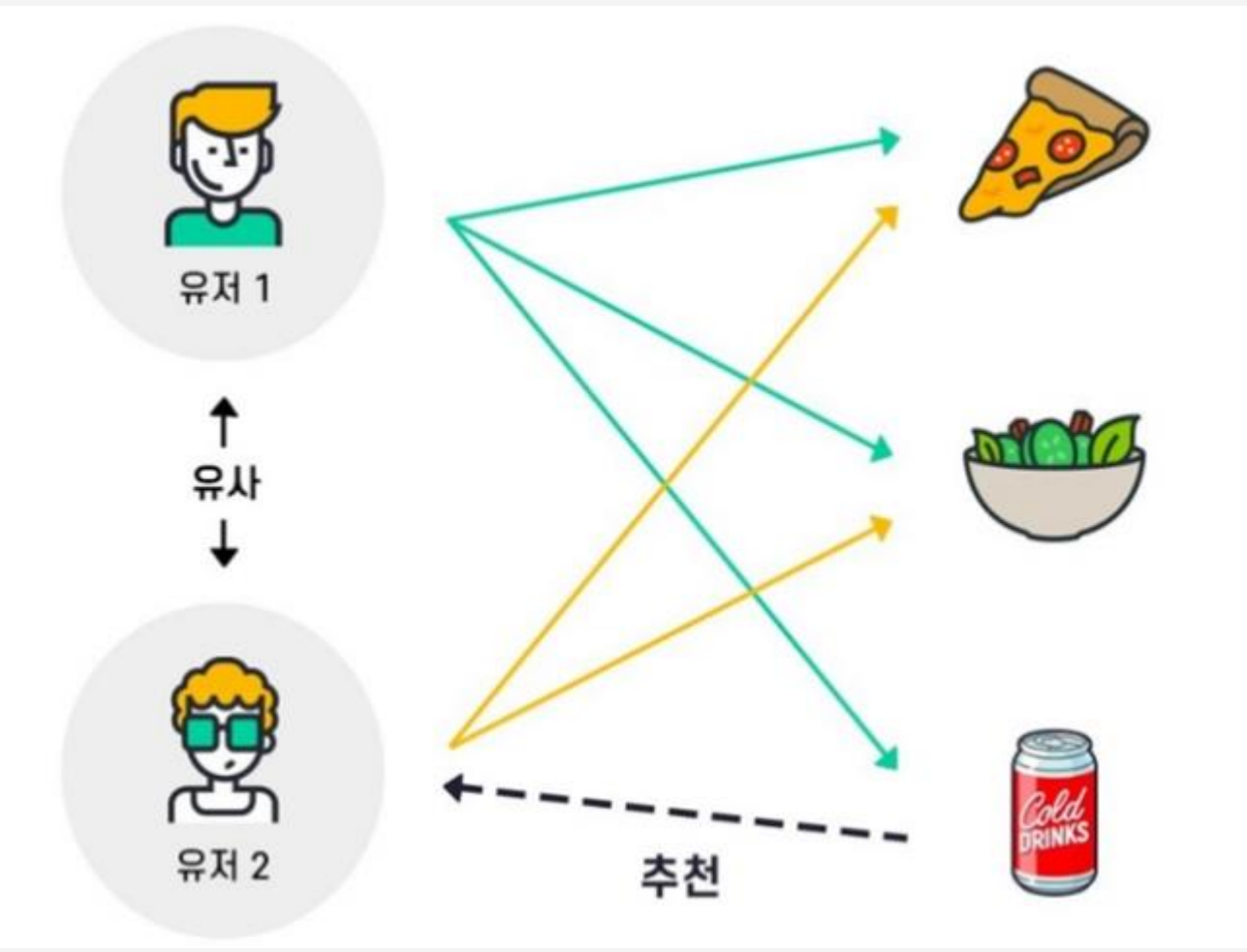
04. 모델 구축 | ① 예측 모델

예측 모델

아이디어 모델

Collaborative Filtering - User based Filtering

Collaborative Filtering 구조



- 나와 성향이 비슷한 사람들이 사용한 아이템을 추천해 주는 방식

|      | feature 1 | feature 2 | feature3 |
|------|-----------|-----------|----------|
| 사용자1 |           |           |          |
| 사용자2 |           |           |          |
| 사용자3 |           |           |          |

1) 위쪽의 표 형식을 이용해 사용자별 벡터 구하여  
Cosine 유사도 측정

2) 나와 가장 유사도가 높은 사용자가 좋아한 아이템 추천

04. 모델 구축 | ① 예측 모델

예측 모델

예측 모델 구조

1) 입력 데이터

|   | feature 1 | feature 2 |
|---|-----------|-----------|
| 1 |           |           |
| 2 |           |           |
| 3 |           |           |

|  |  |  |
|--|--|--|
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |

- 다섯가지 시험 별 가장 중요하다고 판단한 feature 한 열씩 추출
- 조합을 사용하여 두가지씩 뽑아서 왼쪽의 형태로 행렬 추출
- 행렬을 기반으로 사용자별 Cosine 유사도 계산
- 계산 결과 3차원의 유사도 행렬 추출 (조합개수, 29, 29)  
(8명은 test set)

예측 모델

2) 유사도 값과 실제 시험 점수 값 가중치 평균 함수

|               |               |               |     |
|---------------|---------------|---------------|-----|
| 1,1의 유사도<br>값 | 1,2의 유사도<br>값 | 1,3의 유사도<br>값 | ... |
| 2,1의 유사도<br>값 | 2,2의 유사도<br>값 | 2,3의 유사도<br>값 | ... |
| ...           | ...           | ...           | ... |

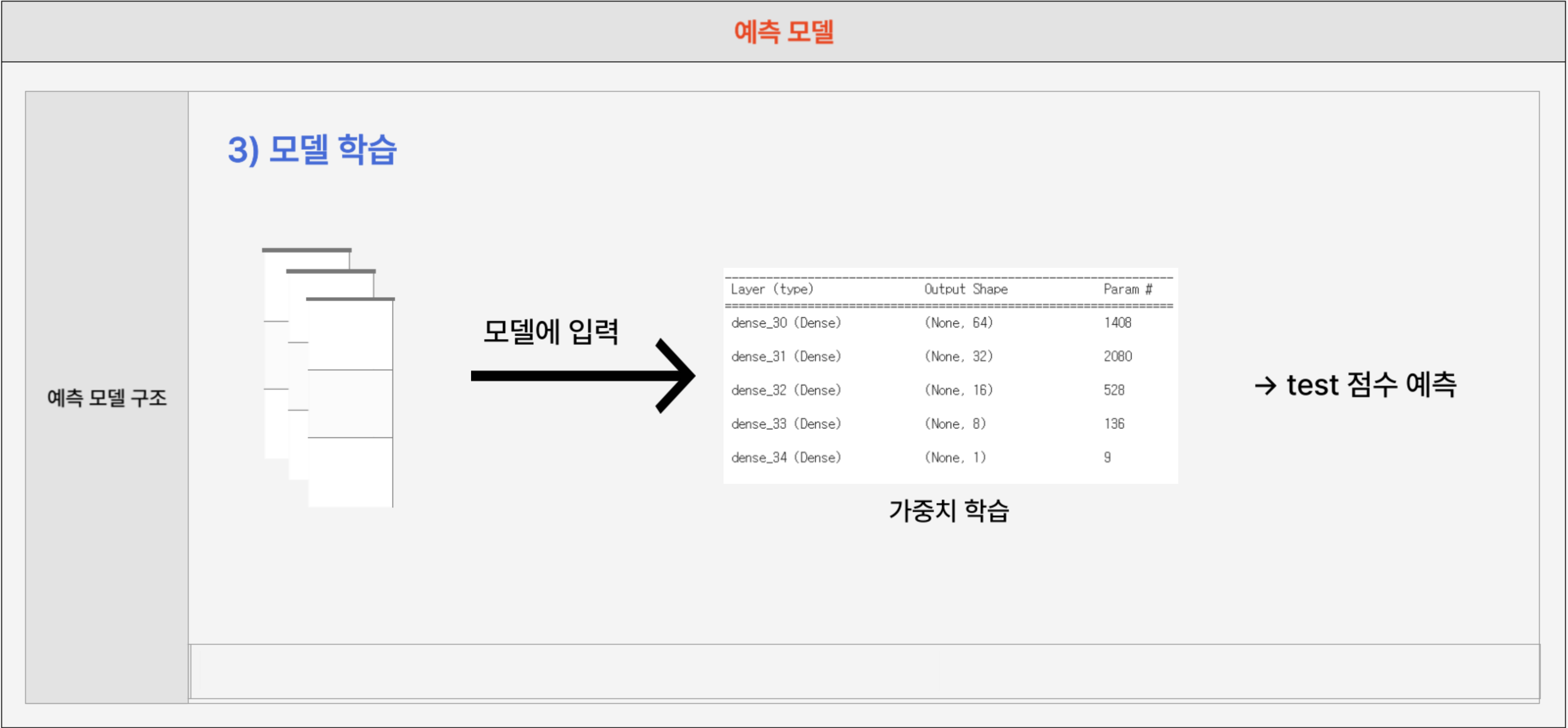
- 사용자1의 실제 시험 점수 곱하여 더한 후 평균
- 사용자2의 실제 시험 점수 곱하여 더한 후 평균
- 사용자 3의 실제 시험 점수 곱하여 더한 후 평균

예측 모델 구조



- 유사도 matrix별 전부 계산
- 대각선 즉, 자기 자신과 자기 자신의 유사도 값은 계산에서 제외
- (조합 개수, 29,1) 3차원 행렬 출력

04. 모델 구축 | ① 예측 모델



04. 모델 구축 | ② 분류 모델

| 분류 모델                                    |   |
|--|---|
| 사용한 모델                                   | 랜덤 포레스트(Random Forest)  |
| 데이터 전처리<br>및<br>Train-<br>Validation 나누기 | <ul style="list-style-type: none"><li>* 범주형 변수 → 레이블 인코딩</li><li>* 연속형 변수 → Feature Scaling: Min-Max Normalization 이용</li><li>* Train-Validation set 8:2 비율로 나누기</li><li>* Train-Validation set 안의 결과변수 비율: 약 6:4</li></ul> |
| 성능 평가                                    | '랜덤서치'를 이용해 최적의 하이퍼파라미터를 찾은 다음, <u>'RandomForestClassifier'</u> 를 이용하여 분류 모델 구축   |
|  | <ul style="list-style-type: none"><li>□ 정확도 87.5%</li><li>□ F-1 점수 88.9%</li><li>□ 재현율 80%</li><li>□ 정밀도 100%</li><li>□ 특이도 100%</li><li>□ AUC 값: 0.9333</li></ul>  |

## 05. 결과 설명 | ① 예측 모델

### 1. DST\_F+B 검사 방법의 예측 결과

```
1/1 [=====] - 0s 132ms/step - loss: 6.5323 - mse: 6.5323  
acc : 6.532321929931641
```

```
1/1 [=====] - 0s 66ms/step  
[[ 9.119286 ]  
 [10.46184 ]  
 [ 7.827271 ]  
 [ 6.257837 ]  
 [ 6.5273576]  
 [ 6.6519747]  
 [ 5.8075857]  
 [ 6.988053 ]]  
[11.  9. 10.  7. 10. 12.  5.  7.]
```

#### - 사용 변수

: 6단계 소요시간 및 머리와 손 움직인 거리, 비밀번호 틀린 횟수

- DST\_F+B검사 : 주의집중력 측정 위한 숫자 검사

→ 비밀번호를 외워서 입력해야 하는 6단계와 관련된 feature 추출

- 소요시간이 길고, 틀린 횟수가 많으며 머리와 손이 움직인 거리가 많을수록 주의집중력 낮은 것이라고 판단



## 05. 결과 설명 | ① 예측 모델

### 2. S-K-BNT 검사 방법의 예측 결과

```
1/1 [=====] - 0s 127ms/step - loss: 5.2921 - mse: 5.2921  
acc : 5.292051792144775
```

```
1/1 [=====] - 0s 80ms/step  
[[11.867567 ]  
 [11.206017 ]  
 [12.095689 ]  
 [11.203219 ]  
 [10.0026045]  
 [10.977524 ]  
 [11.913314 ]  
 [10.805043 ]]  
[12. 11. 14. 13. 14. 13. 8. 11.]
```

#### - 사용 변수

: 총 소요시간, 비밀번호 틀린 횟수, 손 움직임 속도, 손 움직임 거리, 이상치 비율

- S-K-BNT 검사 : 언어 능력 측정 위해 선 그림 본 후 단어 말하는 시각적 검사  
→ '시각'과 관련된 변수 추출

- 패널 그림이 모두 그림이므로 과제를 수행한 총 소요시간이 길고, 비밀번호 틀린 횟수가 많을수록 그림을 보고 단어를 연상하는 능력이 떨어진다고 판단

- 마찬가지로, 그림을 보고 단어를 연상하는 시간이 많이 걸릴수록 손 움직임 속도가 느리고 손 움직임 거리가 클 것이라 판단

- '시각'적 검사이기 때문에  
키오스크 화면 자체를 벗어난 좌표의 비율을 계산한 이상치 비율 사용

## 05. 결과 설명 | ① 예측 모델

### 3. RCFT\_copyscore 검사 방법의 예측 결과

```
1/1 [=====] - 0s 176ms/step - loss: 52.0343 - mse: 52.0343  
acc : 52.034339904785156
```

```
1/1 [=====] - 0s 66ms/step  
[[32.19238 ]  
 [31.898733]  
 [32.95236 ]  
 [43.63003 ]  
 [30.744814]  
 [31.451662]  
 [28.64703 ]  
 [31.364046]]  
[35.  31.  36.  28.  32.  36.  18.5 26. ]
```

#### - 사용 변수

: 비밀번호 틀린 횟수, 1단계에서의 좌표 산만도

- RCFT\_copyscore 검사 : 시공간능력 측정 위한 복잡한 선 그리기 따라하는 검사

- 검사 방법이 선 그리기를 인식한 다음 기억만으로 그림을 재현하는 검사이므로  
'기억력'과 관련된 비밀번호 틀린 횟수 사용

- '시공간능력'이 떨어질수록 패널을 인식하는 좌표가 넓게 분포해있을거라 판단하여  
좌표 산만도 사용

## 05. 결과 설명 | ① 예측 모델

### 4. SVLT\_delayedrecall 검사 방법의 예측 결과

```
1/1 [=====] - 0s 234ms/step - loss: 7.8186 - mse: 7.8186  
acc : 7.818583965301514
```

```
1/1 [=====] - 0s 96ms/step  
[[6.606761 ]  
 [4.4598722]  
 [9.114917 ]  
 [7.7264504]  
 [7.436287 ]  
 [9.577507 ]  
 [3.4531515]  
 [1.229886 ]]  
[ 4.  1.  7.  4.  4. 10.  0.  0.]
```

#### - 사용 변수

: 총 소요 시간, 한 번에 단계 넘어간 횟수, 비밀번호 틀린 횟수, 머리 움직임 속도, 손 움직임 속도

- SVLT\_delayedrecall 검사 : **기억력** 측정 위한 단어 기억력 검사

- 목표 패널을 모두 기억하고 과제를 수행해야 하기 때문에 총 소요시간이 길고, 한 번에 단계를 넘어간 횟수가 많으며 비밀번호 틀린 횟수가 많을수록 '**기억력**'이 안 좋다고 판단.

- 머리 움직임 속도와 손 움직임 속도도 마찬가지로 기억력이 높을수록 빠르게 패널을 고르므로 두 속도가 빠를 것이라고 판단

## 05. 결과 설명 | ① 예측 모델

### 5. K-TMT-E\_B 검사 방법의 예측 결과

```
1/1 [=====] - 0s 126ms/step - loss: 18890.5918 - mse: 18890.5918  
acc : 18890.591796875
```

```
1/1 [=====] - 0s 67ms/step  
[[ 13.156064]  
 [ 9.688511]  
 [-27.363476]  
 [101.13249 ]  
 [355.00516 ]  
 [ 33.63048 ]  
 [109.667404]  
 [166.50801 ]]  
[ 40.  39.  20.  52.  29.  16. 300. 121.]
```

#### - 사용 변수

: 패널 1~6에서 걸린 시간, 총 소요 시간

#### - K-TMT-E\_B 검사 :

전두엽 실행기능 측정 위한 기호 잇기 검사, 시간 측정 검사

- '시간'을 측정하는 검사이므로 소요시간과 관련된 feature 사용

[예측 모델 결론]

- 범위가 비교적 작은 1,2,4번 모델의 mse 값은 작게 5-7점 사이로 도출
- 3번 검사 방법은 범위가 36점이므로 상대적으로 범위가 넓고 MCI 그룹에서의 표준편차가 7으로 4번째로 높음, HC 그룹의 최대값과 MCI 그룹의 최대값이 동일함.  
→ 따라서 분포가 겹치는 부분이 생기므로 mse 값이 높게 나왔다고 판단
- 5번 검사 방법은 MCI 그룹의 점수 분포가 22점에서 300점이므로 분포가 너무 넓을 뿐만 아니라 MCI 그룹의 표준편차가 105.5점으로 매우 높아 mse 값이 높게 나왔다고 판단

| HC | 최소값 | 최대값 | 평균          | 표준편차        |
|----|-----|-----|-------------|-------------|
| 1번 | 7   | 14  | 10.28571429 | 1.927248223 |
| 2번 | 11  | 15  | 13          | 1.095445115 |
| 3번 | 28  | 36  | 32.71428571 | 2.299844715 |
| 4번 | 4   | 11  | 6.714285714 | 2.390457219 |
| 5번 | 16  | 61  | 32.19047619 | 13.24620341 |

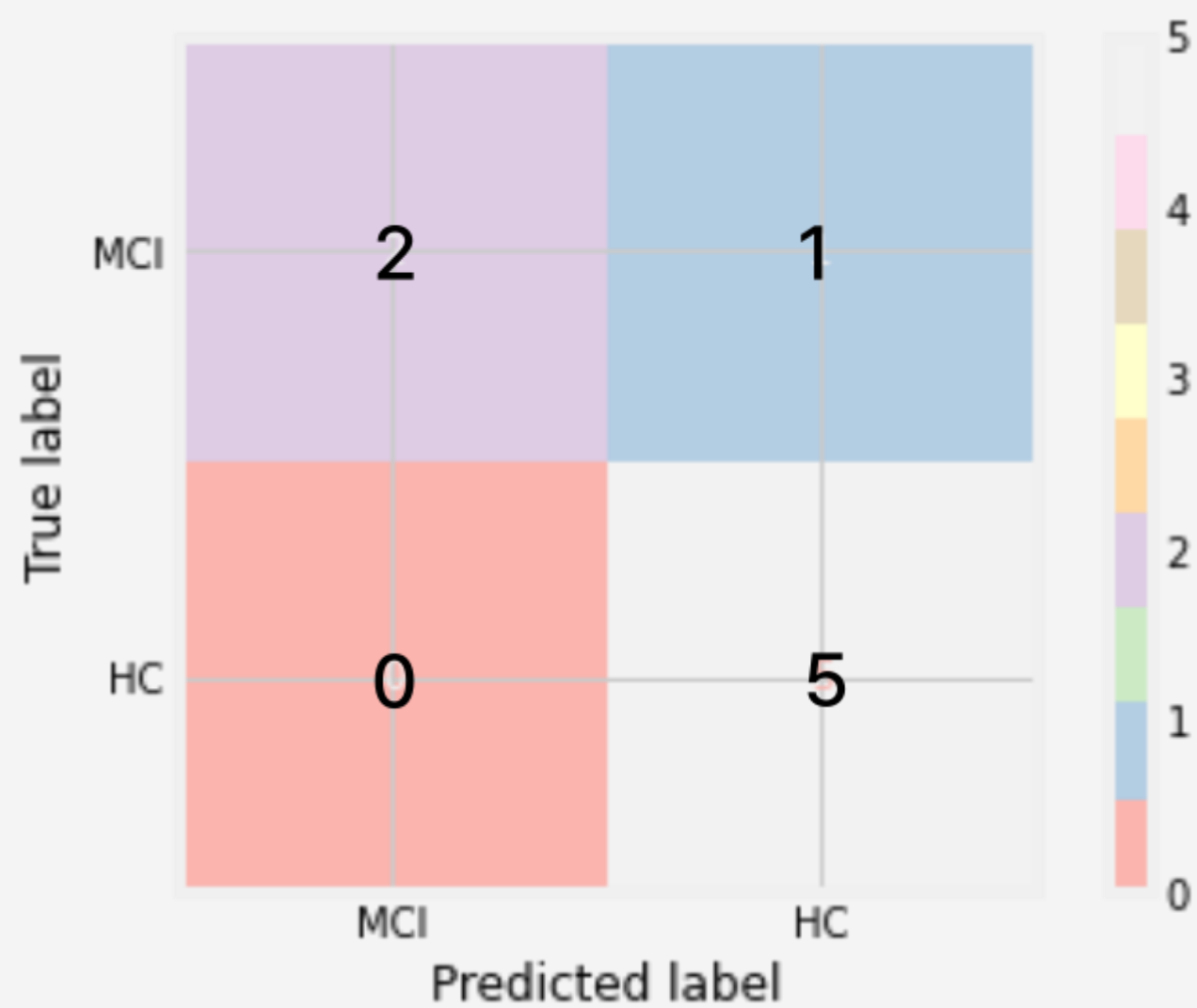
▸ HC 그룹의 점수 별 분석

| MCI | 최소값 | 최대값 | 평균          | 표준편차        |
|-----|-----|-----|-------------|-------------|
| 1번  | 4   | 12  | 8.545454545 | 2.197992    |
| 2번  | 5   | 15  | 10.68181818 | 2.41791653  |
| 3번  | 16  | 36  | 26.77272727 | 7.041927619 |
| 4번  | 0   | 9   | 2.636363636 | 2.71758742  |
| 5번  | 22  | 300 | 118.9090909 | 105.4644644 |

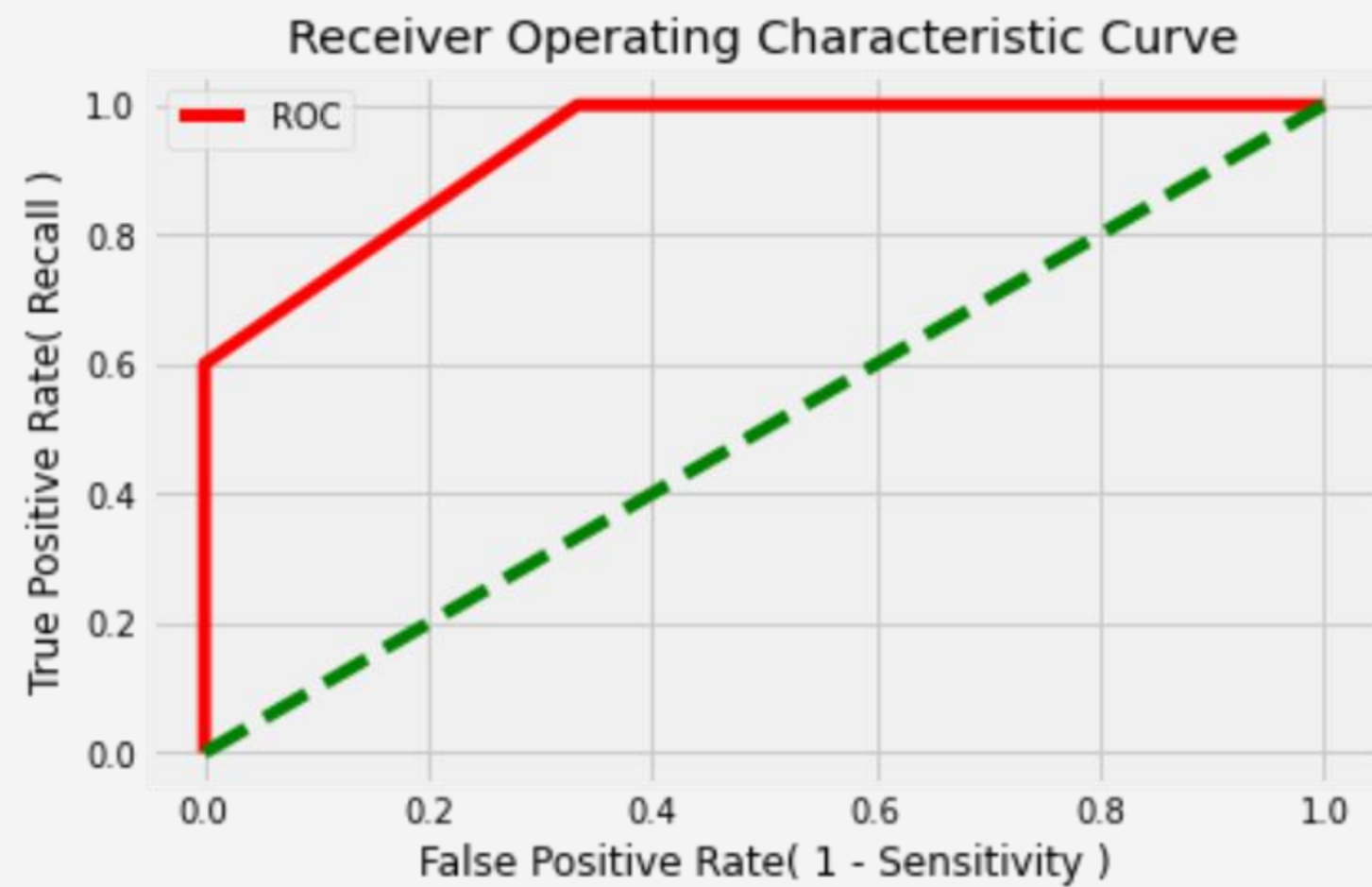
▸ MCI 그룹의 점수 별 분석

## 05. 결과 설명 | ② 분류 모델

### ▼ 오차행렬



### ▼ ROC 곡선

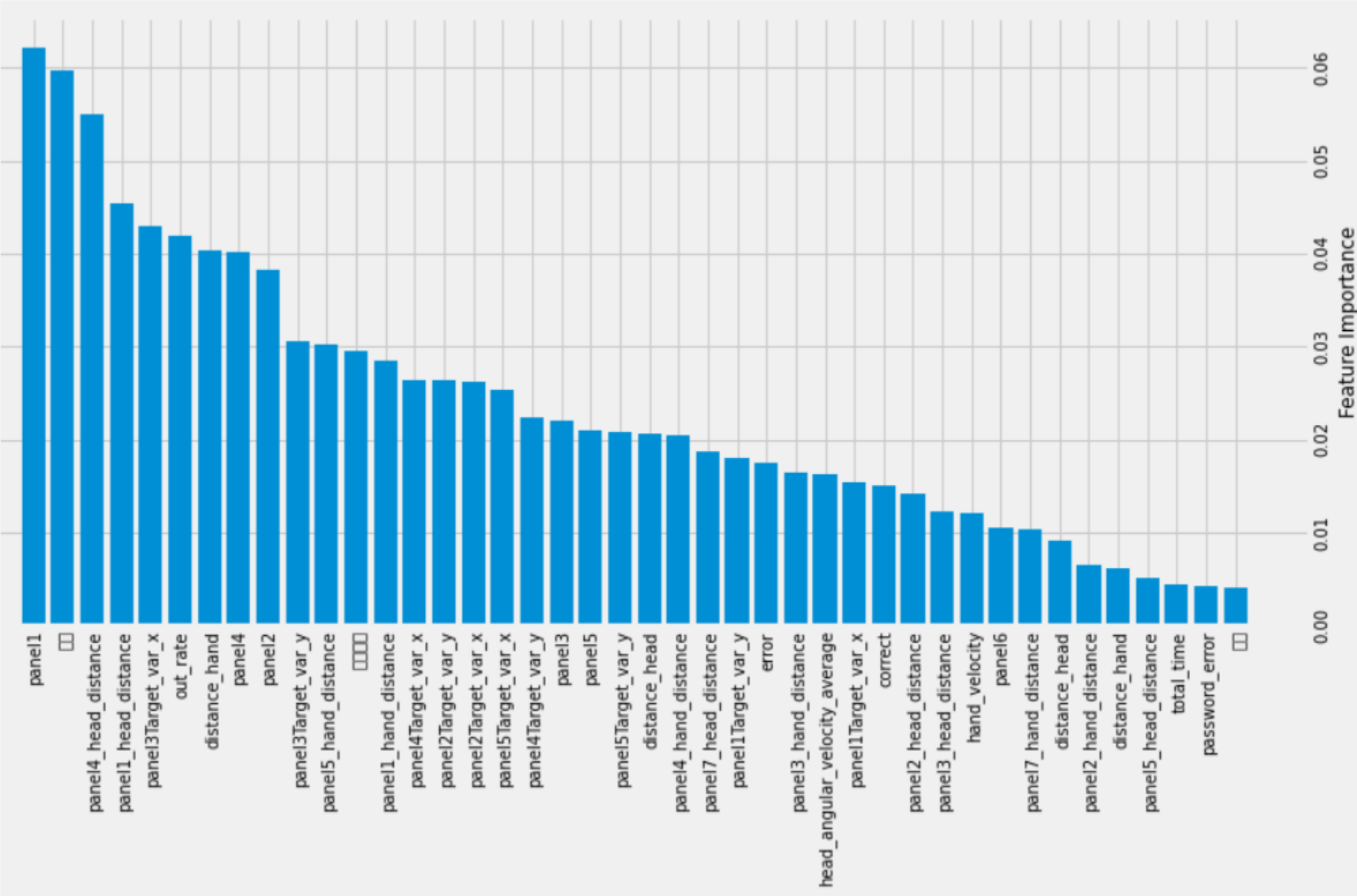




05. 결과 설명 | ② 분류 모델

▼ 분류 모델로 도출된 변수 중요도

|                                 |
|---------------------------------|
| TOP4 중요한 변수                     |
| 패널 별 걸린 시간                      |
| 패널 별 움직인 거리(머리, 손)              |
| 키오스크 화면을 벗어난 좌표의 비율             |
| 목표 버튼의 중심점을 기준으로 좌표 퍼진 정도 (x좌표) |



05. 결과 설명 | ② 분류 모델

| TOP4 중요한 변수                       | 추정  |
|-----------------------------------|---|
| 패널 별 걸린 시간                        | 목표 항목을 잘 외우지 못했다.<br>집중력이 상대적으로 낮다.                     |
| 패널 별 움직인 거리(머리)                   | 현재의 task에 온전히 집중하지 못한다.<br>총 걸린 시간이 많을 수록 움직인 거리도 많다.   |
| 키오스크 화면을 벗어난 좌표의 비율               | 현재의 task에 온전히 집중하지 못한다.                                 |
| 목표 버튼의 중심점을 기준으로<br>좌표 퍼진 정도(x좌표) | 목표 항목을 잘 외우지 못해 시선이 방황하고 있다.<br>현재의 task에 온전히 집중하지 못한다. |



05. 결과 설명 | ② 분류 모델

패널 별 걸린 시간 평균 비교

|        | HC        | MCI       |
|--------|-----------|-----------|
| Panel1 | 3.126667  | 8.195455  |
| Panel2 | 5.14      | 14.854545 |
| Panel3 | 13.706667 | 16.854545 |
| Panel4 | 4.56      | 10.436364 |

(단위: sec)

패널 별 움직인 거리(머리) 평균 비교

|        | HC       | MCI      |
|--------|----------|----------|
| Panel1 | 0.11215  | 0.342356 |
| Panel2 | 0.216326 | 0.661189 |
| Panel3 | 0.627484 | 0.780932 |
| Panel4 | 0.24073  | 0.472097 |
| Panel5 | 0.151222 | 0.495695 |
| Panel6 | 1.321546 | 2.03901  |

(단위: m)

화면을 벗어난 시선 좌표 비율의 평균 비교

|          | HC        | MCI      |
|----------|-----------|----------|
| out_rate | -0.111933 | 0.076318 |

\*표준화 한 값

시선 좌표의 퍼진 정도(x좌표) 평균 비교

|        | HC       | MCI      |
|--------|----------|----------|
| Panel1 | 0.011393 | 0.012927 |
| Panel2 | 0.019693 | 0.022345 |
| Panel3 | 0.015753 | 0.024868 |
| Panel4 | 0.025367 | 0.038445 |
| Panel5 | 0.012127 | 0.019305 |



# THANK YOU!

지금까지 발표를 들어주셔서 감사합니다!

---