

## CodeAlive 전반적인 내용

### Practice & Challenge 부분

- Debugging이 어려움
- 변수 이름, 슬라이싱 방법 등이 정답코드와 **정확히** 일치해야 정답처리되는 부분
- 거의 모든 실습문제가 데이터셋을 필요로 하고 데이터셋의 구조가 실습에 핵심적인데 `show(data)` 등의 코드로 **Visualization** 탭에 띄워놓지 않는 이상 데이터셋을 보면서 진행하기에 많이 불편함
- 코드를 실행하는 도중 다른 페이지로 넘어가게 되면 **응답 없음** 현상이 생길 때가 있음 (항상은 아님)
- 코드 활용 예제 부분에 반복문이 등장할 때가 있는데 예제 부분 특성상 한줄씩 실행해가면서 설명해주기 때문에 breakpoint 기능을 가르쳐주어야함 (예: `for i in range(20)` 안의 코드를 모두 20번씩 반복실행해주어야함)
- 코드 실행시 애니메이션 화면은 대부분 아무런 변화도 없고 캐릭터 이름과 실습문제에 등장하는 인물의 이름과 매칭도 안 되어있을 때가 있음. 학생들이 CA과정을 선택하게되는 이유는 활동적이고 코드 실행 결과를 직관적으로 볼 수 있기 때문인데 SI과정부터는 그러한 부분이 거의 없다시피 하기 때문에 흥미요소가 줄어드는 이유가 될 수 있음
- **문제 정의하기**와 활용 예제 부분에서 기능적인 부분이 짧게 설명되어 있기 때문에 실습문제풀이에 가장 많이 참고하는 부분은 코드에 미리 작성되어 있는 주석임. 그러나 주석 설명에 오류가 있는 부분이 가끔 있음.
- **문제 해결하기**에서 정확도 또는 정확한 수치를 요구하는데 지금까지 구성한 코드를 재실행해서 확인해야하는 점이 불편함. 처음에 코드 구성을 할 때부터 문제 취지와 맞게 코드 작성해야 한다는 점을 학생들에게 인지시켜주어야 함.
- 주차마다 새로운 머신 러닝 기법이 등장하는데 지도/비지도 학습인지, 분류/회귀인지 설명이 바로바로 나와있지 않기 때문에 추가설명 필요.
- 학생들 입장에서느 대부분 처음 배우는 내용이고 용어나 개념 자체도 생소하기 때문에 주석이나 설명에 의존하게되니 정확한 설명과, 오타없는 주석 필요.

## CA-AI I

프로젝트 주차는 제외하였습니다.

### 1~4주차

- `pandas` 모듈과 함께 이전 과정에서 배웠던 데이터 분석과 시각화 복습은 꼭 필요하다고 생각합니다.
- 데이터 시각화 부분에서 지난 과정에서 거의 항상 학습하던 내용이기 때문에 학생들이 지루해할 수도 있을 것 같습니다.
  - 단순히 막대, 꺾은선, 산점도 뿐만 아니라 `matplotlib` 또는 `seaborn`에서 제공하는 다양한 그래프를 사용하여 시각화해보는 것도 흥미 유지에 좋을 것 같습니다.
  - 특히 `seaborn` 모듈은 꾸준히 업데이트되고 3D 그래프도 지원하고 있으니 간단하게라도 실습해보면 좋을 것 같습니다.

- 4주차마다 진행하는 프로젝트는 성취도가 낮은 학생들은 스스로 진행하기 굉장히 어려울 것으로 생각되어 난이도를 조금 낮추거나 가이드라인을 상세히 기재할 필요가 있을 것 같습니다.

## 5~8주차

- **Linear Regression**: 1차함수와 비슷한 내용으로 1차함수까지 배운 학생들은 수학적으로 설명해주는 것이 효과적일 것 같습니다. 결정계수, 상관계수까지는 불필요. `numpy`도 `pandas`와 같이 자주 사용하는 유용한 모듈이라고 간단히 설명해주면 좋을 것 같습니다. 첫 모델링 수업이니 `model.fit(x_train, y_train), model = LinearRegression()` 등의 모델을 만들고 학습시키는 코드를 기억할 수 있도록 강조하여 수업해야할 것 같습니다.
- **Logistic Regression**: `accuracy_score`에 관한 활용법이나 설명이 부족하다고 생각합니다. Challenge 부분 교안 내용이 오류가 많아서 검토가 필요할 것 같습니다.

## 9~12주차

- **Decision Tree**: MP과정에서 많이 다루어왔던 순서도, 결정트리라서 익숙할 수 있지만 머신 러닝 데이터 분류 기법이라고 설명해주어야 합니다.
- **KNN**: 현재 제공된 내용만으로는 ML, DL 관련 경험이나 지식이 부족한 강사는 개념 설명에 어려움이 있을 수 있습니다.
- **Clustering**: 군집화 기법은 여러가지가 있는데 그중에 특정해서 `Kmeans`만 소개한다는 점 알려주어야 합니다.

## CA-AI II

9~12주차는 복습내용(AI Project)이어서 생략하였습니다.

## 1~4주차

- **Ensemble**: 크게 Overfitting 이슈로 실무에서 사용을 꺼려하는 내용입니다. **Pipeline**과 같은 보다 효과적인 모델 비교기법이 있기 때문에 `RandomForestRegressor()` 부분을 보다 집중적으로 설명하는 것이 좋을 것 같습니다.
- **Hyperparameters**: 모델 평가하기 부분 Runtime이 너무 오래 걸리는 이슈가 있습니다. `RandomForestClassifier()`가 등장하기 때문에 이전 주차인 **Ensemble** 내용과 합쳐도 괜찮을 것 같습니다. (RFC와 RFR 차이점?)
- **Boosting**: 학생들도 대부분 이정도 주차까지 학습하게 되면 모델링 하는 방법에 많이 익숙해지기 때문에 머신러닝 알고리즘을 사용하는 방법보다 어떤 데이터에 어떠한 알고리즘을 선택하여 사용하여야 하는지 알아가는 방식도 설명해주면 좋을 것 같습니다.

## 5~8주차

- 5~6주차 내용을 합쳐서 간소화하고 7~8주차에 걸쳐서 CNN 내용을 진행하는 것이 개인적으로 보다 나은 구성이 될 수 있을 것 같습니다.
- **Tensorflow**와 **Keras**: 이 두가지는 각자만의 문법이 있을 정도로 배울 내용이 방대하기 때문에 **Tensor**가 무엇이고 어떻게 생성하는 지 정도의 기본적인 이론 학습이 필요할 수 있다고 생각합니다.