|  |  |
| --- | --- |
| 교육 제목 | **Logistic Regression** |
| 교육 일시 | 21.10.19 (화) |
| 교육 장소 | 오프라인 (영우글로벌러닝) |
| **교육 내용** | |
| 오전 | **▣ Logistic Regression**  -------------------------------------------------------------------------------------------  **∇ Logistic Regression**  \* 분류의 문제인데 회귀로 해석할 때 필요  -------------------------------------------------------------------------------------------  **∇ Sigmoid 함수**  \* 1/(1+exp(-z)) 꼴의 함수로 표현.  z = np.arange(-5, 5, 0.1) # z는 e의 지수 부분  prob\_y = 1/(1+np.exp(-z))  plt.plot(z, prob\_y)  -------------------------------------------------------------------------------------------  **∇ Logistic Regression (Probability) 구해보기**  \* print(lr.predict\_proba(train\_bream\_smelt[:5]))  # 예측 확률을 구해줌, 1에 가까운게 smelt일 확률을 나타냄.  ------------------------------------------------------------------------------------------- |
| 오후 | **▣ Grdient Descent & Decision Tree, Random Forest**  -------------------------------------------------------------------------------------------  **∇ Grdient Descent**  \* from sklearn.linear\_model import SGDClassifier  sc = SGDClassifier(loss = 'log', max\_iter = 100)  # learning\_rate 는 Gradient Descent에서 감마를 나타냄  # max\_iter 는 100번까지만 움직이겠다.  # loss는 loss function J(세타)의 term을 설정해줌  sc.fit(train\_scaled, train\_target)  -------------------------------------------------------------------------------------------  **∇ Partial fit**  \* 업데이트 된 데이터들로 fit함.  -------------------------------------------------------------------------------------------  **∇ Decision Tree**  \* Root Node, Interal Node, Branch, Leaf Node로 구성  \* Gini Index = 1-시그마(P^2)  Tree의 우선순위 및 연속형일땐 어느 구간에서 나눌지 분류일땐 우월성 보여줌.  \* Information Gain  Information Gain이 큰 값일수록 우선순위가 높아짐. (0.5일땐 의미 없음.)  ------------------------------------------------------------------------------------------- |