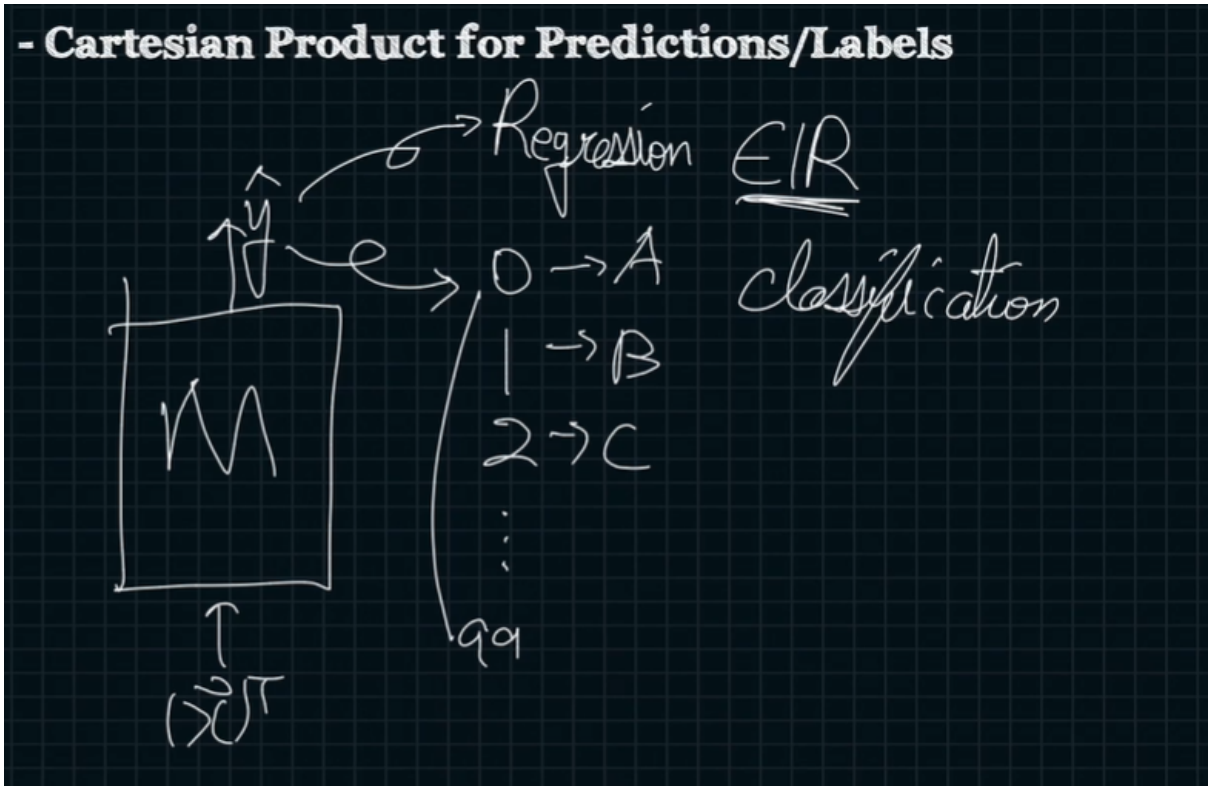
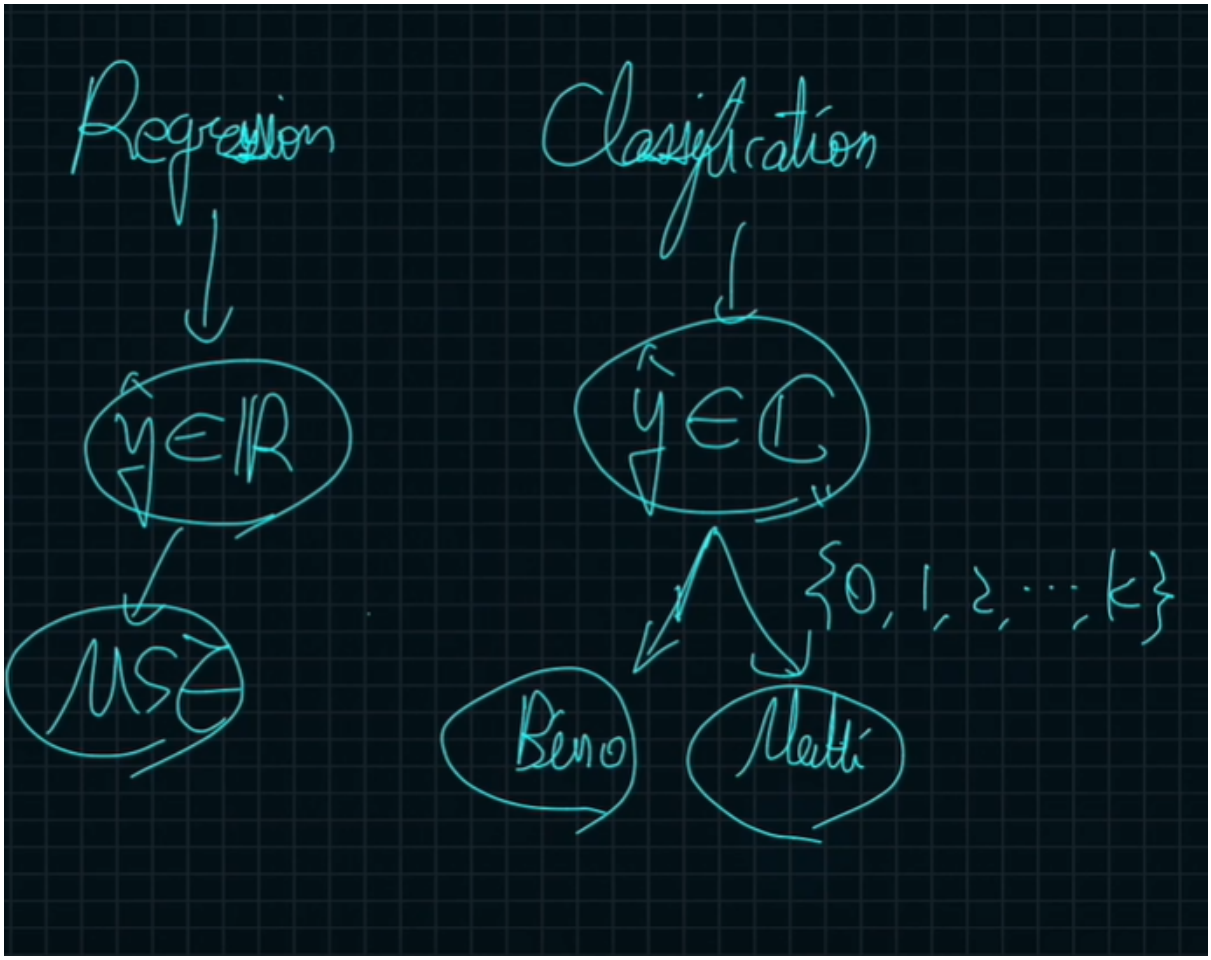
Loss Functions



- X_Transformation vector를 input으로 들어가서 Affine transformation연산을 하게 되고 예측값이 나오는데, 이 예측값은 연속이 될 수 있고 classification문제처럼 될 수 있다.
- Regression Problem에서 모델에서 나오는 예측값들이 대소관계가 존재한다.
- 하지만, Classfication문제는 0 → 강아지 1 ——>고양이 고양이가 강아지보다 크니까 고양이가 더 크다? 이런 개념이아니다. 대소관계가 존재하지 않는다.

우리는 사실 인공지능 모델이 실제 정답값이랑 차이를 최소화 되도록 하고싶다.예를들어서 공부시간이 있고 수학성적이 있다고 가정하자. 공부시간이 5시간 수학성적 100점 또다른 사람 공부시간 4시간 80점이라는 데이터가 있는데 모델이 4시간을 공부했을 때 80점이 나오도록 5시간이 공부했을 때 100점이 나오도록 학습을 하고싶다. 우리의 전체 데이터셋이 y값이 존재하고 x값이 존재하는데 x값을 받아서 y값에 비슷해지도록 하고싶은 거다.

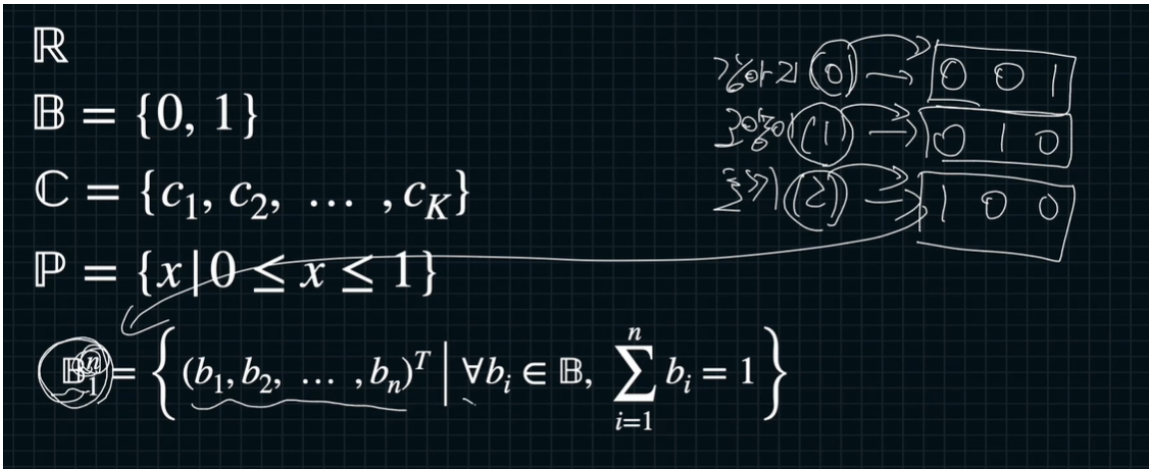
우리의 인공지능 모델은 x값에 들어가서 y값에 최대한 가깝게 출력이 되게 ㅎ학습을 하는것이다.



- Supervised Learning에서는 두가지가 있다고 했다. regression과 classification 일단, regression문제에서 예측값은 실수값이 나오고 classification문제에서는 class 값이 나오게 된다. 이거에 따라 Loss 사용 법이 달라진다.
- regression에서 loss는 MSE가 있고 classification에서는 이진 분류 문제에서는 binary cross entropy가 있고, 멀티 분류 문제에서는 Categorical cross entropy 가 있다.

- 위에서 말한 loss는 인공지능 모델에서 나온 예측값과 실제값의 차이를 어떻게 수치화를 해줄지에 대한 도구라고 생각하면된다.

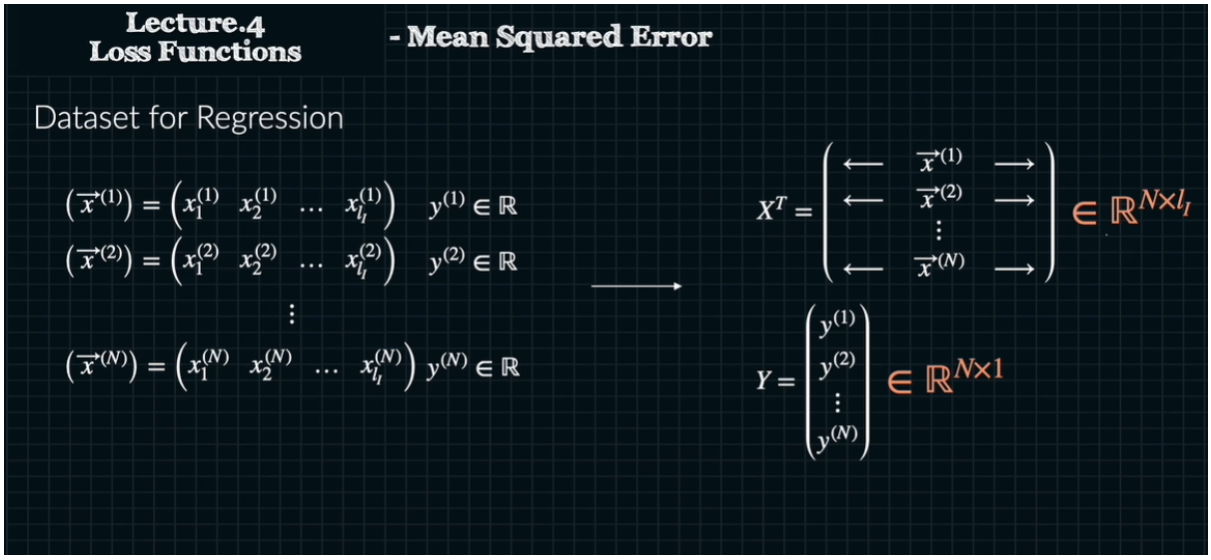
- 그래서 우리는 실제 값과 비슷한 출력을 만들기 위해서는 우리는 실제값고 예측값의 차이를 줄이기 위해 loss가 필요한것



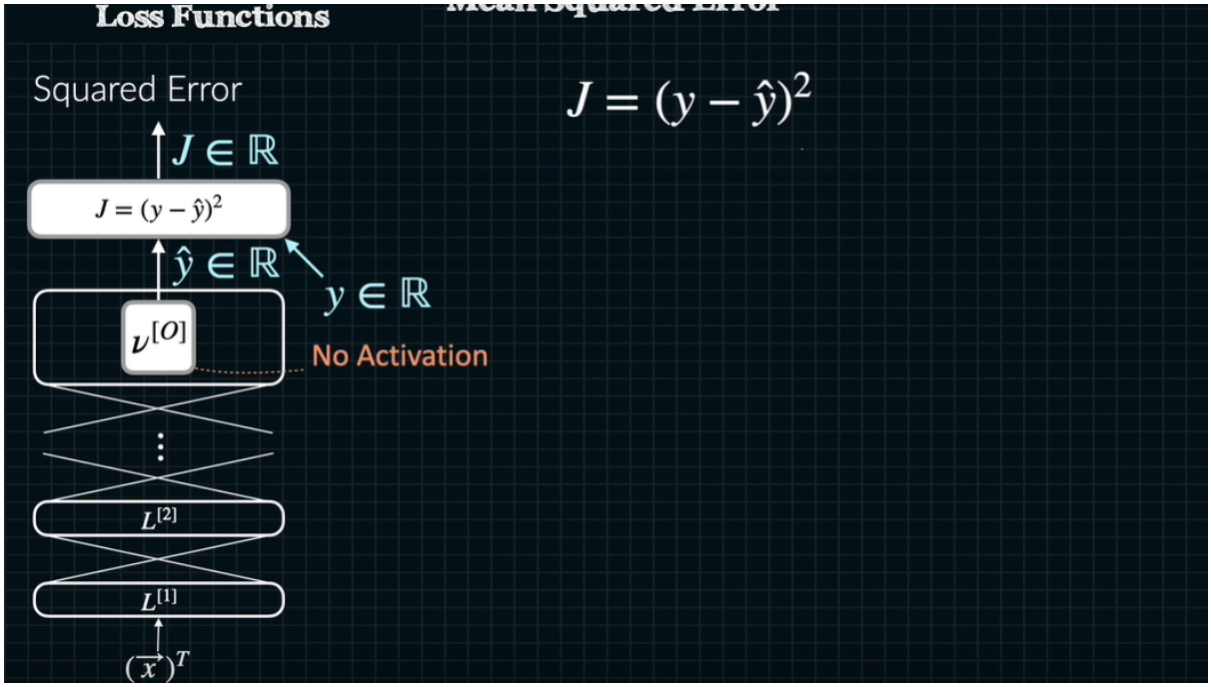
자기 위치에서만 1인 표현을 하는게 one hot vector이다.

- R → 실수 전체 집합이다.
- B - > Binary 집합
- C —→ Categorical Class 집합
- P → 확률 의 집합
- 마지막 B는 원핫 집합. 하나의 값만 1인 집합 B앞에 1이 붙어 있는 이유는 원핫 vector들의 원소들임

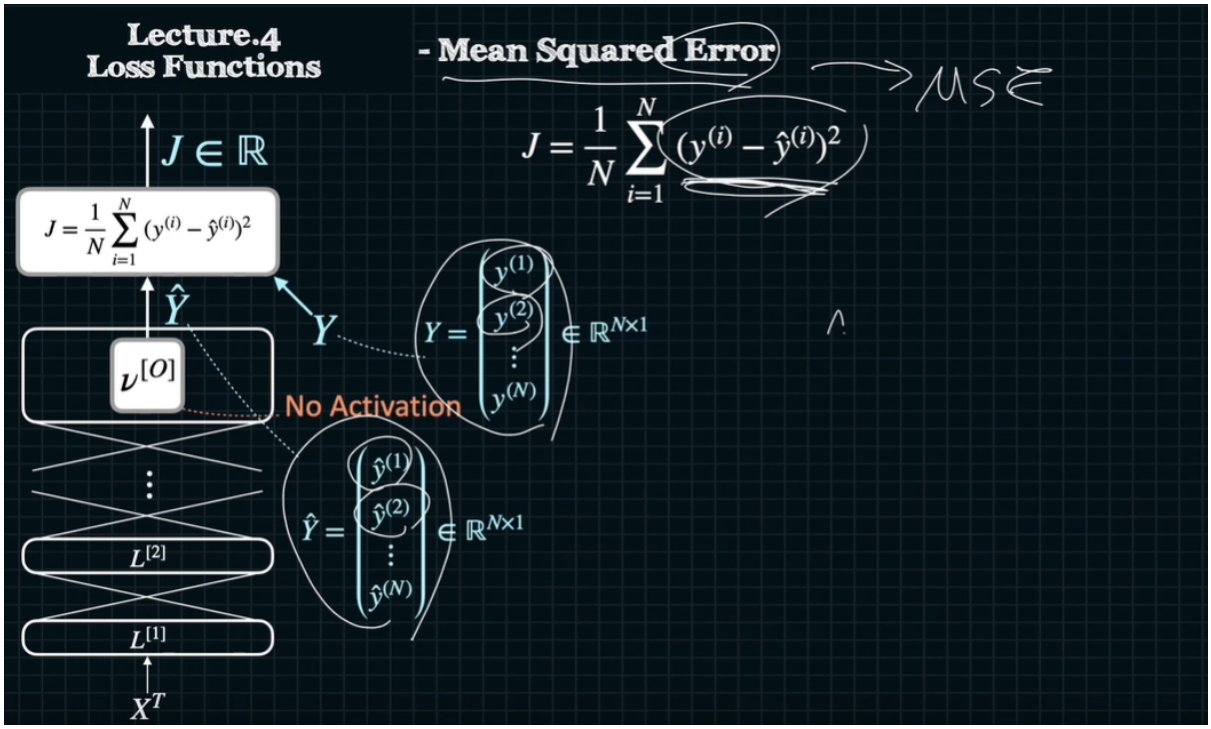
Mean Squared Error



- 첫 번째 sample에 대해서 y값을 가지는데 이 y값은 실수 전체 범위를 가진다. 두 번째도 마찬가지 마지막 번째도 마찬가지이다.
- 오른쪽보면 X_transformation인 행렬 metrix 데이터를 볼 수 있는데 이 metrics는 실수 범위를 가진다.
- y값도 마찬가지이다.



Minibath



- X 데이터의 mini batch형태의 데이터가 인풋으로 들어가고 dense layer을 통과한 후 Activation function을 통과하지 않는다. 그래서 미니배치만큼 들어간다고 해서 Weight와 bias가 늘어나는건 아니다. 마지막 layer를 보면 뉴런 하나만 있다 왜? 하나의 값을 예측하기위해 실수값을 예측을 해야하기 때문에 Linear function이나 no activation이다.
- 그래서 마지막 예측값을 보면 y_^과 실제값 y 들을 볼 수 있는데 이 각각의 원소들을의 차이를 모두 고려를 해줘야한다.
- 여러개의 미니베치로 들어가면 미니베치만큼 예측값들이 나오고 이 예측값들의 실제값들의 차이를 어떻게 수치화를 해주냐면 각각의 원소끼리 빼주고 갯수로 나눠주면된다.

Mean Squared Error의 특징

- 실제값과 예측값의 제곱의 값들이 비슷해질 수록 0에 가까워지고 차이가 심해지면 엄청 커진다. Loss는 0보다 큰값이되고 서로 비슷해지면 0 으로 가깝고 차이가 나면 무한대로 발산하고
- Mean squared error는 아래로 볼록한 그래프로 된다. 이차함수

Why model need loss?

- 그렇다면 왜 model이 loss가 필요할까?생각을 해보면, x_data가 input으로 들어가고 모델을 통과하면서 Affine transformation연산을하게되고 여기서 예측값이 나오는데 이 모델이 실제값이랑 얼마나 잘 맞았는지 혹은 얼마나 비슷한지를 수치화를 해줘야하는데, 그 방법이 loss인거 같다. 즉, loss는 모델이 실제값과 예측값이 얼마나 차이나는지를 수치화로 보는 방법인거 같다. 그래서 이 loss가 한 없이 작아지면 실제값이랑 예측값이랑 비슷하고 loss 값의 차이가 크다면 무한대로 발산할 가능성이 있다.
- 내가 생각했을 때 모델은 loss를 얼마나 최적화를 잘하는지가 중요한것 같다.
- 결국 batch size만큼 데이터가 들어가면 batch size만큼 예측값이 나오고 batch size만큼 loss를 구하는 방법이 mean squared error인거 같다. 그래서 MSE가 최소가 되면 가장 학습을 잘 했다고 판단하면 좋을 거 같다. 아직까진