|  |  |
| --- | --- |
| 교육 제목 | AI 프로젝트 기반 S/W 전문가 양성과정 |
| 교육 일시 | 21.12. |
| 교육 장소 | 집 |
| **교육 내용** | |
| 오전 | 1. 경사 하강법 복습 편미분(기울기)를 0을 찾아가는 것 lr = step 학습률 = (느림 : overshooting) 사이를 잘 찾아야 함 w = (기울기)가중치, b = bias(절편) 생성 f(x) = ax + b 기본적인 선형회귀 함수 MSE 구하기 tf.reduce\_mean(tf.square(hypothesis – y) 평균 제곱인데 루트 학습률 지정 – learning\_rate 모델 객체 생성 – Sequential() 레이어 추가 – model.add(Dense()) 모델학습 후 생성 – compile 학습 – fit() 예측 – predict() 2. 다중 선형 회귀 은닉층을 넣음(hidden layer) 3. 로지스틱 회귀 정의 0 or 1 시그모이드 함수를 사용 = sigmoid(x) = 1 / 1 + e^-1(ax + b) < x로 편미분 0과 1사이의 값을 가지는 특성 때문에 로그 함수를 써야함 np.e = 로그 4. 로지스틱 회귀에서 퍼셉트론으로 5. 퍼셉트론 – 신경망 활성망에서 일정 값을 넘으면 true, 못 넘으면 false 여러 층을 조합해서 입력 값을 판단 입력 -> sum(가중치 + bias) -> 활성화 함수를 함수로 적용 기존 : y = ax + b(a는 기울기, b는 y절편) 앞으로 : y = wx + b(a는 가중치, b는 bias) 6. 퍼셉트론 하나만으로는 XOR 논리를 풀 수 없음 XOR gate : t t = 0, t f = 1, f t = 0, t t = 1 다중 퍼셉트론을 사용하여 은닉 층(hidden layer, h)을 넣으면 XOR gate를 풀 수 있음 중간 층을 node라고 함 7. XOR gate 실습 8. 은닉 계층 – hidden layer 은닉 계층은 입력 계층과 연결된 전결합 계층이다 얕은 신경망에서는 1개의 은닉 계층만을 사용한다. 9. 출력 계층 – output layer 출력 계층의 활성 함수에 의해 결정 10. 은닉 계층이 여러 개 인 것을 심층 신경망이라고 함 보통 5개 이상을 deep neural network(DNN) 11. 입력 🡪 은닉계층 🡪 활성 함수 🡪 출력 계층 12. 오차 역전파 = 🡨 방향으로 |
| 오후 | 1. 오차 역전파 엡실론 = 노차, 노이즈 엡실론이 충분히 작으면 2. 동적 계획법 중복 계산이 발생하면 저장해 놓고 갖다 씀 3. 연쇄 법칙 = chain rule 4. 결국 오차를 업데이트하면서 오차를 최소화한 값을 찾아내는 것 5. 델타 식 6. 오차역전파 = 🡨 편미분, 오차가 0이 될 때까지 가중치 수정 7. 시그모이드 미분하면 최대치가 0.3, 계속 미분 하다보면 기울기가 사라짐 8. 렐루(relu)함수를 활성화 함수로 많이 씀 미분했을 때 1이하로 떨어지는 일이 없음, 하지만 0은 있음 소프트 플러스 함수 = 렐루 함수를 미분했을 때 0으로 떨어지는걸 완화하고자 만들었음 9. SGD = 확률적 경사 하강법 경사 하강법은 하나하나 다 해야 돼서 느리니까 적당한 random 값으로 찾아 내려 감, 아주 약간의 정확도 손실이 있음 모멘텀을 적용하면 지그재그가 덜 함(방향성) Keras의 최적화 함수를 이용 파라미터 lr = 0.1을 자주 씀 파라미터 momentum을 0.9 정확도 개선 nesterpy = True : 네스테로프, 정확도 개선 adagrad = 학습률을 적게 해서 보폭 크기 개선 keras.optimizers.Adagrad adagrad 파라미터 (lr = 0.1를 조절하고 epsilon = 1e – 6) RMSProp = keras.optimizers.RMSProp() 보폭 크기 개선 Adam = 모멘텀과 RMSProp을 합침, 정확도와 보폭 크기 개선 10. 모델의 정의 딥러닝 실행 model.compile(loss = “mean\_squared\_error”, optimizer = “adam”, metrics = [“accuracy”], binary\_crossentropy 이항 교차 정말 중요!!!!!! : activation에 따라 loss 함수를 뭘 줄꺼냐 잘 선택해야 함 model.fit() : 출력 층 11. 실습 |