1. 일반적인 머신러닝(지도학습)의 학습에 대한 설명으로 가장 잘못된 것은?

① 학습의 목적은 예측을 잘하는 것에 있다. O

② 예측을 잘한다는 것은 trainset에 대한 타겟이 무엇인지 잘 맞추겠다는 의미다. X

③ 예측을 잘한다는 것은 Trainset의 X 와 y 의 관계를 잘 파악했다는 의미다.

④ X와 y의 관계를 잘 파악했다는 것은 hypothesis를 잘 찾았다는 의미다.

⑤ hypothesis를 잘찾았다는 의미는 cost가 낮을 때의 weight를 찾았다는 의미다.

2. 다음 중 지도학습이 아닌 모델을 모두 고르시오.

① ElasticNet

② Xgboost

③ VGGNet

④ PCA (비지도)

⑤ GAN (비지도)

3. 다음 중 회귀분석에 쓰이지 않는 모델을 모두 고르시오. (3, 4)

① ElasticNet

② Adaboost

③ LogisticRegression (분류)

④ K-Means (비지도 - Clustering)

⑤ Randomforest (트리 - 멀티플레이어)

4. 디시젼트리의 앙상블 모형 중 Boosting 모형에 해당하는 것을 모두 고르시오.

(4, 5)

① Decision Tree Classifier

② Decision Tree Regressor

③ Random Forest

④ AdaBoost

⑤ Xgboost

5. 다음 중 규제와 관련이 없는 것을 모두 고르시오.

① Standard Scaler (1. 스케일이 다르면, 거리를 재는 몇몇 알고리즘이 동등하게 특성들을 비교하지 못함. 2. 학습을 빠르게 시키기 위해서.)

② L1 Norm reg

③ L2 Norm reg

④ Drop-Out

⑤ pca (차원축소)

6. 최적화에 대한 설명 중 잘못된 설명을 모두 고르시오.

① 최적화는 trainset에 대해 cost를 최대한 낮추는 weight를 찾아가는 과정이다.

② cost가 가장 낮을 때의 weight를 찾기 위해선, weight에 대한 cost의 gradient를 계산해야 한다.

③ gradient descent 방법론에서 gradient가 양수가 나오면, weight는 감소할 때 cost가 낮아진다. **weight = (weight - lr \* gradient)**

④ gradient descent 방법론에서 gradient가 음수가 나오면, cost가 낮아지는 방향으로 weight를 업데이트하면 weight는 커진다.

⑤ 일반적으로 gradient **절대값**의 크기보다 weight의 크기가 크다.

7. 규제에 대한 설명 중 잘못된 설명을 모두 고르시오. **(모호) (1,2,3) or (2,3)**

① trainset 에 대해서 정확도가 높으면 오버피팅이다. **(1)**

② testset 에 대해서 정확도가 낮으면 오버피팅이다. (함께 비교)

③ testset 에 대해서 정확도가 낮으면 언더피팅이다. (

④ 문제의 복잡도보다 모델의 복잡도가 너무 높다면 trainset은 잘 맞추지만, testset은 잘 못맞출 가능성이 높다.

⑤ 문제에 복잡도 맞게 모델의 복잡도를 일반화시키는 것이 규제화라고 볼 수 있다.

8. 평가지표에 대한 설명으로 잘못된 것을 모두 고르시오.

① R2\_score는 회귀 분석에 쓰인다.

② R2\_score는 데이터의 분산도를 얼마나 설명하는지를 나타내는 지표라고 이해할 수 있다.

③ 정확도는 진짜 Positive 중에 Positive라고 예측한 비율을 의미한다.

④ Accuracy가 1이면 F1\_score도 1이다.

⑤ 정확도가 90%인 경우가 모든 경우에서, 정확도가 80%인 경우보다 항상 옳다.

9. 아래와 같은 데이터를 가지고 있을때, 아래 질문에 답해보세요

→ size로 price를 예측하기.

|  |  |
| --- | --- |
| **size** | **price** |
| **10평** | **1억** |
| **15평** | **2억** |
| **20평** | **2.5억** |

위의 데이터를 사용했을 때, 사용하기에 가장 적합하지 않은 모델은?

1. 리니어 리그레션
2. 로지스틱 리그레션
3. 릿지 리그레션
4. 엘라스틱넷
5. 라쏘 리그레션

10. 다음 중 딥러닝에 대한 설명으로 잘못된 것을 모두 고르시오.

① 일반적으로 **복잡도가 높은 문제에 대해 좋은 성능**을 보여준다.

② 일반적으로 **강한 비선형 문제를 풀 때 사용**된다.

③ 일반적으로 **underfitting 의 상황에서는 layer**를 높여준다.

④ 일반적으로 overfitting이 되면 각 layer의 node 수를 높여준다.

⑤ 일반적으로 **오버피팅이 잘 일어난다.**

11**.** 아래와 같은 데이터를 가지고 있을때, 아래 질문에 답해보세요.(10점)

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| X\_1 | X\_2 | X\_3 | y\_1 | y\_2 | y\_3 |
| 10 | 13 | 14 | 1 | 0 | 0 |
| 21 | 23 | 25 | 0 | 1 | 0 |
| 45 | 44 | 34 | 0 | 0 | 1 |

위의 데이터는 y값이 정답이고, 총 3 종류로 이루어져 있다. 정답이 0인 데이터는 (1, 0, 0) , 정답이 1이면, (0, 1, 0) , 정답이 2이면 (0, 0, 1)로 표현했다. 데이터 중에 대표적으로 3개만 표현했다. **(bias(W0) 포함)** (데이터 개수 1000개)

위의 데이터를 분류 예측한다고 했을 때, 로지스틱 리그레션을 사용한다면, 총 학습해야하는 weight의 개수는?(10점)

12. 클래스는 10이고, 차원이 224, 224, 3 인 이미지 데이터셋이 총 10만장이 있다. 다음 질문에 대답해보세요.

layer 구성이 다음과 같을 때, 각 층의 파라미터(weight)개수와 피처맵의 차원을 계산해보세요.

(**stride는 텐서플로에서 default값으로 주어지는 값**을(default 1) 사용하는 것으로 가정한다.)

layer 구성 : conv1 conv2 pool1 conv3 conv4 pool2 FFN1 FFN2 FFN3(output layer)

(224 x 224 x 3, )

conv1 : filter 32개, size=3x3, padding x (222 x 222 x 32, (((3\*3)\*3) + 1) \* 32 )

conv2 : filter 32개, size=3x3, padding x (220 x 220 x 32, (((3\*3)\*32) + 1) \* 32 )

pool1 : 2x2 (110x 110x32, 0 )

conv3 : filter 64개, size=3x3, padding x (108 x 108 x 64, (((3\*3)\*32) +1) \*64 )

conv4 : filter 64개, size=3x3, padding x ( 106 x 106 x 64, ((3\*3\*64 + 1) \*64) )

pool1 : 2x2 (53x 53 x 64, 0)

Flatten (53x 53 x 64, )

FFN1 : 128 ((53 \* 53 \* 64 + 1) \* 128)

FFN2 : 64 ((128+1) \* 64)

FFN3 : 10 (activation: ‘softmax’)

=> loss: (sparse\_)categorical\_crossentropy

(1). sigmoid

(2). Relu

(3) softmax