머신러닝 CHAPTER 04 객관식 퀴즈

다양한 분류 알고리즘

Part 1. 이론 및 개념 (1~50번)

1. 로지스틱 회귀(logistic regression)에 대한 설명으로 올바른 것은?

① 이름 그대로 회귀 문제를 푸는 알고리즘이다
② 이름은 회귀이지만 실제로는 분류 모델이다
③ 비선형 방정식을 학습하는 알고리즘이다
④ k-최근접 이웃의 다른 이름이다
⑤ 비지도 학습 알고리즘이다

2. 로지스틱 회귀가 학습하는 것은?

① 가장 가까운 이웃
② 의사결정 트리
③ 선형 회귀와 동일한 선형 방정식
④ 확률 분포 모델
⑤ 클러스터 중심

3. 로지스틱 회귀에서 선형 방정식의 출력(z)을 확률(0~1)로 변환하기 위해 이진 분류에서 사용하는 함수는?

① 소프트맥스 함수
② 시그모이드 함수
③ ReLU 함수
④ 로그 함수
⑤ 지수 함수

4. 시그모이드 함수의 출력 범위는?

① 음의 무한대 ~ 양의 무한대
② -1 ~ 1
③ 0 ~ 1
④ 0 ~ 100
⑤ -0.5 ~ 0.5

5. 이진 분류에서 시그모이드 함수의 출력이 0.5보다 크면 어떤 클래스로 판단하는가?

① 음성 클래스

② 양성 클래스

③ 판단 불가

④ 두 클래스 모두

⑤ 클래스 없음

6. 시그모이드 함수에 z=0을 입력하면 출력값은?

① 0

② 0.25

③ 0.5

④ 1

⑤ -1

7. 로지스틱 회귀에서 다중 분류를 위해 사용하는 함수는?

① 시그모이드 함수

② 소프트맥스 함수

③ ReLU 함수

④ tanh 함수

⑤ 로그 함수

8. 소프트맥스 함수의 핵심 특징으로 올바른 것은?

① 출력값의 합이 항상 0이 된다

② 출력값의 합이 항상 1이 된다

③ 출력값이 음수가 될 수 있다

④ 하나의 z값만 입력받는다

⑤ 이진 분류에서만 사용된다

9. 타깃 클래스가 2개 이상인 분류 문제를 무엇이라 하는가?

① 이진 분류

② 다중 분류(multiclass classification)

③ 회귀

④ 클러스터링

⑤ 차원 축소

10. 사이킷런에서 로지스틱 회귀 모델을 만드는 클래스와 그 위치는?

① sklearn.neighbors의 LogisticRegression

② sklearn.linear_model의 LogisticRegression

③ sklearn.preprocessing의 LogisticRegression

④ sklearn.metrics의 LogisticRegression

⑤ sklearn.tree의 LogisticRegression

11. 사이킷런의 분류 모델에서 클래스별 확률값을 반환하는 메서드는?

① predict()

② score()

③ predict_proba()

④ decision_function()

⑤ fit()

12. decision_function() 메서드가 반환하는 것은?

① 클래스별 확률

② 모델이 학습한 선형 방정식의 출력(z값)

③ 정확도

④ 손실 함수 값

⑤ 표준화된 데이터

13. 사이킷런에서 타깃값을 모델에 전달하면 클래스의 순서가 어떻게 정렬되는가?

① 입력 순서 그대로 유지

② 알파벳 순으로 자동 정렬

③ 빈도순으로 정렬

④ 랜덤 순서

⑤ 숫자 크기순 정렬

14. 모델에 저장된 정렬된 타깃 클래스 목록을 확인할 수 있는 속성은?

① targets_

② labels_

③ classes_

④ categories_

⑤ names_

15. 로지스틱 회귀의 다중 분류에서 coef_ 배열의 행 수는 무엇에 의해 결정되는가?

① 샘플 수

② 특성 수

③ 클래스 수

④ 에포크 수

⑤ alpha 값

16. 로지스틱 회귀의 다중 분류에서 예측 클래스를 결정하는 방법은?

① 가장 작은 z값을 가진 클래스

② 가장 높은 z값(또는 확률)을 가진 클래스

③ 랜덤으로 선택

④ 첫 번째 클래스를 항상 선택

⑤ z값의 평균을 사용

17. LogisticRegression에서 규제를 제어하는 매개변수는?

① alpha

② C

③ penalty

④ max_iter

⑤ tol

18. LogisticRegression의 C 매개변수와 릿지 회귀의 alpha 매개변수의 관계는?

① 둘 다 값이 커지면 규제가 강해진다

② C는 작을수록 규제가 커지고, alpha는 커질수록 규제가 커진다(서로 반대)

③ 둘은 완전히 동일하다

④ C는 분류에만, alpha는 회귀에만 적용된다

⑤ 두 매개변수는 관련이 없다

19. LogisticRegression의 C 매개변수 기본값은?

① 0

② 0.1

③ 1

④ 10

⑤ 100

20. LogisticRegression이 기본적으로 적용하는 규제 방식은?

① L1 규제 (라쏘)

② L2 규제 (릿지)

③ 규제 없음

④ L1과 L2를 함께 사용

⑤ 드롭아웃

21. LogisticRegression의 max_iter 매개변수의 기본값은?

① 10

② 50

③ 100

④ 1000

⑤ 10000

22. 불리언 인덱싱(Boolean indexing)의 설명으로 올바른 것은?

① 숫자 인덱스로 행을 선택하는 방법

② True/False 값을 전달하여 행을 선택하는 방법

③ 문자열로 열을 선택하는 방법

④ 랜덤으로 행을 선택하는 방법

⑤ 슬라이싱으로 행을 선택하는 방법

23. 이진 분류에서 predict_proba() 메서드가 반환하는 배열의 열 구성은?

① 양성 클래스 확률만 1열

② 첫 번째 열이 음성 클래스, 두 번째 열이 양성 클래스 확률

③ 첫 번째 열이 양성 클래스, 두 번째 열이 음성 클래스 확률

④ 클래스 이름이 첫 번째 열

⑤ z값이 반환됨

24. 소프트맥스 함수가 '정규화된 지수 함수'라고도 불리는 이유는?

① 로그 함수를 사용하기 때문에

② 지수 함수를 사용하여 값을 변환하고 합이 1이 되도록 정규화하기 때문에

③ 표준편차로 나누기 때문에

④ 제곱근을 사용하기 때문에

⑤ 절댓값을 사용하기 때문에

25. scipy 라이브러리에서 시그모이드 함수를 제공하는 함수 이름은?

① sigmoid()

② logistic()

③ expit()

④ softmax()

⑤ exp()

26. 점진적 학습(온라인 학습)이 필요한 상황은?

① 데이터가 적고 고정되어 있을 때

② 새로운 데이터가 지속적으로 추가되어 모델을 실시간으로 갱신해야 할 때

③ 테스트 세트가 없을 때

④ 하이퍼파라미터 튜닝이 필요할 때

⑤ GPU가 없을 때

27. 확률적 경사 하강법(SGD)에서 '확률적'이란 말의 의미는?

① 확률을 예측한다는 뜻

② 무작위하게 또는 랜덤하게라는 기술적 표현

③ 확률 분포를 학습한다는 뜻

④ 정확한 계산을 한다는 뜻

⑤ 결정적(deterministic)이라는 뜻

28. 확률적 경사 하강법의 핵심 원리는?

① 전체 데이터를 한 번에 사용하여 최적화

② 훈련 세트에서 랜덤하게 하나의 샘플을 골라 경사를 따라 내려감

③ 가장 가까운 이웃을 찾아 평균을 계산

④ 결정 트리를 반복적으로 만듦

⑤ 클러스터 중심을 이동시킴

29. 미니배치 경사 하강법의 설명으로 올바른 것은?

① 하나의 샘플만 사용

② 여러 개의 샘플을 사용해 경사 하강법을 수행

③ 전체 샘플을 한 번에 사용

④ 샘플을 사용하지 않음

⑤ 테스트 세트만 사용

30. 배치 경사 하강법의 설명으로 올바른 것은?

① 하나의 샘플만 사용

② 여러 개의 샘플을 사용

③ 한 번 경사로를 따라 이동하기 위해 전체 샘플을 사용

④ 샘플을 전혀 사용하지 않음

⑤ 랜덤하게 한 개를 선택

31. 에포크(epoch)의 정의로 올바른 것은?

① 하나의 샘플을 사용하는 과정

② 모델을 처음부터 다시 훈련하는 과정

③ 확률적 경사 하강법에서 훈련 세트를 한 번 모두 사용하는 과정

④ 테스트 세트를 평가하는 과정

⑤ 하이퍼파라미터를 조정하는 과정

32. 손실 함수(loss function)에 대한 설명으로 올바른 것은?

① 값이 클수록 좋은 모델이다

② 머신러닝 알고리즘이 얼마나 엉터리인지 측정하는 기준으로, 값이 작을수록 좋다

③ 모델의 파라미터 수를 측정하는 함수이다

④ 데이터의 크기를 측정하는 함수이다

⑤ 항상 정확도와 동일하다

33. 손실 함수가 반드시 갖추어야 할 수학적 조건은?

① 정수값만 출력해야 한다

② 미분 가능해야 한다

③ 항상 양수여야 한다

④ 최대 1이어야 한다

⑤ 불연속이어야 한다

34. 정확도(accuracy)를 손실 함수로 직접 사용하기 어려운 이유는?

① 계산이 너무 복잡해서

② 정확도는 불연속적이라 미분이 불가능하기 때문에

③ 정확도는 항상 0이기 때문에

④ 정확도는 회귀에만 쓸 수 있기 때문에

⑤ 정확도는 음수가 될 수 있기 때문에

35. 이진 분류에서 사용하는 손실 함수의 이름은? (2가지 이름 중 하나 선택)

① 평균 제곱 오차 손실 함수

② 로지스틱 손실 함수(= 이진 크로스엔트로피 손실 함수)

③ 힌지 손실 함수

④ MAE 손실 함수

⑤ R^2 손실 함수

36. 회귀 문제에서 사용하는 대표적인 손실 함수는?

① 로지스틱 손실 함수

② 힌지 손실 함수

③ 크로스엔트로피 손실 함수

④ 평균 제곱 오차 손실 함수

⑤ 시그모이드 손실 함수

37. 사이킷런에서 확률적 경사 하강법을 사용한 분류 모델을 만드는 클래스는?

① LogisticRegression

② SGDClassifier

③ KNeighborsClassifier

④ SGDRegressor

⑤ Ridge

38. SGDClassifier의 loss 매개변수 기본값은?

① log

② squared_loss

③ hinge

④ cross_entropy

⑤ mse

39. SGDClassifier에서 로지스틱 회귀를 수행하려면 loss 매개변수를 어떻게 설정해야 하는가?

① loss='hinge'

② loss='log'

③ loss='squared_loss'

④ loss='logistic'

⑤ loss='cross_entropy'

40. SGDClassifier에서 점진적 학습(추가 학습)을 수행하는 메서드는?

① fit()

② predict()

③ partial_fit()

④ score()

⑤ transform()

41. partial_fit() 메서드와 fit() 메서드의 차이로 올바른 것은?

① partial_fit()은 모델을 처음부터 다시 훈련한다

② partial_fit()은 기존 모델에 1 에포크씩 이어서 훈련하고, fit()은 처음부터 새로 훈련한다

③ fit()이 partial_fit()보다 항상 느리다

④ 둘은 완전히 동일한 기능이다

⑤ partial_fit()은 테스트 세트에만 적용된다

42. 조기 종료(early stopping)의 정의로 올바른 것은?

① 훈련을 시작하지 않는 것

② 과대적합이 시작하기 전에 훈련을 멈추는 것

③ 과소적합 상태에서 훈련을 멈추는 것

④ 손실 함수가 0이 되면 멈추는 것

⑤ 에포크를 1로 설정하는 것

43. 에포크 횟수에 따른 과대적합과 과소적합의 관계로 올바른 것은?

① 에포크가 적으면 과대적합, 많으면 과소적합

② 에포크가 적으면 과소적합, 많으면 과대적합 가능성이 높다

③ 에포크 수와 적합도는 관련 없다

④ 에포크가 많을수록 항상 성능이 좋아진다

⑤ 에포크가 적을수록 항상 성능이 좋아진다

44. 힌지 손실(hinge loss)은 어떤 머신러닝 알고리즘을 위한 손실 함수인가?

① 로지스틱 회귀

② k-최근접 이웃

③ 서포트 벡터 머신(SVM)

④ 랜덤 포레스트

⑤ 신경망

45. SGDClassifier의 penalty 매개변수 기본값은?

① l1

② l2

③ elasticnet

④ none

⑤ ridge

46. SGDClassifier의 max_iter 매개변수 기본값은?

① 10

② 100

③ 300

④ 1000

⑤ 10000

47. SGDClassifier의 tol 매개변수의 역할은?

① 규제 강도를 설정한다

② 학습률을 설정한다

③ 반복을 멈출 조건(손실 감소량 기준)을 지정한다

④ 에포크 횟수를 설정한다

⑤ 배치 크기를 설정한다

48. 사이킷런에서 확률적 경사 하강법을 사용한 회귀 모델을 만드는 클래스는?

① SGDClassifier

② SGDRegressor

③ LinearRegression

④ Ridge

⑤ Lasso

49. SGDRegressor의 loss 매개변수 기본값은?

① log

② hinge

③ squared_loss

④ cross_entropy

⑤ mse

50. k-최근접 이웃 분류기에서 주변 이웃 10개 중 사각형 3개, 삼각형 5개, 원 2개일 때, 삼각형으로 예측될 확률은?

① 20%

② 30%

③ 50%

④ 70%

⑤ 100%

Part 2. 코드 및 응용 (51~100번)

51. 다음 코드에서 pd.unique()가 반환하는 것은?

```
print(pd.unique(fish['Species']))
```

① Species 열의 모든 값(중복 포함)

② Species 열의 고유한(중복 제거된) 값 목록

③ Species 열의 값 개수

④ 데이터프레임 전체

⑤ Species 열의 평균값

52. 다음 코드에서 fish_input에 저장되는 것은?

```
fish_input = fish[['Weight','Length','Diagonal','Height','Width']]
```

① Species 열만 포함된 데이터프레임

② 5개 특성 열로 구성된 새로운 데이터프레임

③ 전체 데이터프레임의 복사본

④ 1차원 넘파이 배열

⑤ 타깃 데이터

53. StandardScaler를 사용할 때 주의할 점은?

① 테스트 세트의 통계값으로 훈련 세트를 변환해야 한다

② 훈련 세트의 통계값으로 테스트 세트를 변환해야 한다

③ 훈련 세트와 테스트 세트를 각각 독립적으로 표준화해야 한다

④ 표준화는 타깃 데이터에만 적용해야 한다

⑤ 표준화 후 원본 데이터를 삭제해야 한다

54. 다음 코드에서 kn.classes_의 출력이 알파벳순인 이유는?

```
print(kn.classes_)
# 출력: ['Bream' 'Parkki' 'Perch' 'Pike' 'Roach' 'Smelt' 'Whitefish']
```

① 데이터가 원래 알파벳순으로 정렬되어 있어서

② 사이킷런이 타깃값을 자동으로 알파벳순으로 정렬하기 때문에

③ 사용자가 정렬 코드를 작성했기 때문에

④ 판다스가 자동 정렬하기 때문에

⑤ 넘파이가 자동 정렬하기 때문에

55. predict_proba() 메서드의 출력에서 각 열의 순서를 결정하는 기준은?

① 입력 데이터의 순서

② classes_ 속성의 순서와 동일

③ 랜덤 순서

④ 확률이 높은 순

⑤ 알파벳 역순

56. 다음 코드에서 bream_smelt_indexes의 역할은?

```
bream_smelt_indexes = (train_target == 'Bream') | (train_target == 'Smelt')
```

① 도미와 빙어 데이터를 삭제한다

② 도미 또는 빙어인 행을 True로, 나머지를 False로 표시하는 불리언 배열을 만든다

③ 도미와 빙어의 평균을 계산한다

④ 전체 데이터를 정렬한다

⑤ 새로운 데이터프레임을 생성한다

57. 이진 분류에서 lr.classes_가 ['Bream', 'Smelt']일 때, 양성 클래스는?

① Bream (첫 번째)

② Smelt (두 번째, 알파벳순 뒤)

③ 둘 다 양성

④ 양성 클래스 없음

⑤ 랜덤으로 결정

58. 다음 코드에서 lr.coef_와 lr.intercept_가 의미하는 것은?

```
print(lr.coef_, lr.intercept_)
```

① R^2 점수와 MAE

② 로지스틱 회귀가 학습한 선형 방정식의 계수와 절편

③ 시그모이드 함수의 매개변수

④ 훈련 세트의 평균과 표준편차

⑤ 클래스 확률

59. 다음 코드에서 decision_function()이 반환하는 값에 expit()을 적용하면 얻을 수 있는 것은?

```
decisions = lr.decision_function(train_bream_smelt[:5])
from scipy.special import expit
print(expit(decisions))
```

① R^2 점수

② z값의 절댓값

③ predict_proba()의 양성 클래스 확률과 동일한 값

④ 손실 함수 값

⑤ 표준화된 z값

60. 다중 분류에서 coef_.shape가 (7, 5)이고 intercept_.shape가 (7,)일 때, 7과 5는 각각 무엇을 의미하는가?

① 7은 샘플 수, 5는 에포크 수

② 7은 클래스 수, 5는 특성 수

③ 7은 특성 수, 5는 클래스 수

④ 7은 이웃 수, 5는 샘플 수

⑤ 7은 에포크 수, 5는 배치 크기

61. 다음 코드에서 softmax 함수를 사용할 때 axis=1로 지정하는 이유는?

```
from scipy.special import softmax
proba = softmax(decision, axis=1)
```

① 열 방향으로 소프트맥스를 계산하기 위해

② 각 행(샘플)에 대해 소프트맥스를 계산하기 위해

③ 전체 배열에 대해 계산하기 위해

④ 행 방향으로 합을 구하기 위해

⑤ 열을 삭제하기 위해

62. 다음 코드에서 np.round(proba, decimals=4)의 역할은?

```
import numpy as np
print(np.round(proba, decimals=4))
```

① 확률값을 4자리 정수로 변환

② 소수점 네 번째 자리까지 유지하고 다섯 번째 자리에서 반올림

③ 확률값을 4배로 곱함

④ 상위 4개 값만 출력

⑤ 4개의 클래스만 선택

63. 다음 코드에서 SGDClassifier의 두 매개변수의 의미는?

```
sc = SGDClassifier(loss='log', max_iter=10, random_state=42)
```

① loss='log'는 회귀 손실, max_iter=10은 샘플 수

② loss='log'는 로지스틱 손실 함수, max_iter=10은 에포크 횟수 10회

③ loss='log'는 로그 스케일, max_iter=10은 이웃 수

④ loss='log'는 자연로그, max_iter=10은 특성 수

⑤ loss='log'는 출력 형식, max_iter=10은 배치 크기

64. 다음 코드에서 sc.partial_fit()을 호출하면 어떤 일이 일어나는가?

```
sc.partial_fit(train_scaled, train_target)
```

① 모델을 처음부터 다시 훈련한다

② 기존 모델에 1 에포크만큼 추가 훈련한다

③ 테스트 세트를 평가한다

④ 모델을 삭제한다

⑤ 하이퍼파라미터를 자동 조정한다

65. 다음 코드에서 classes=classes를 partial_fit()에 전달하는 이유는?

```
classes = np.unique(train_target)
sc.partial_fit(train_scaled, train_target, classes=classes)
```

① 클래스 수를 제한하기 위해

② 훈련 세트에 포함된 전체 클래스 목록을 알려주기 위해

③ 클래스별 가중치를 설정하기 위해

④ 특정 클래스만 훈련하기 위해

⑤ 클래스 이름을 변경하기 위해

66. 에포크 그래프에서 백 번째 에포크 근처가 적절한 반복 횟수인 이유는?

① 훈련 점수가 가장 낮아지는 지점이므로

② 훈련 세트와 테스트 세트의 점수가 모두 높으면서 과대적합이 시작되기 직전이므로

③ 테스트 점수가 0이 되는 지점이므로

④ 에포크 수가 짝수이므로

⑤ 손실 함수가 최대인 지점이므로

67. 다음 코드에서 tol=None으로 설정하는 이유는?

```
sc = SGDClassifier(loss='log', max_iter=100, tol=None, random_state=42)
```

① 에러를 무시하기 위해

② 자동 조기 종료를 비활성화하고 정확히 max_iter만큼 반복하기 위해

③ 손실 함수를 사용하지 않기 위해

④ 규제를 비활성화하기 위해

⑤ 모든 샘플을 사용하기 위해

68. SGDClassifier에서 loss='hinge'로 설정하면 어떤 알고리즘을 위한 손실 함수를 사용하는가?

① 로지스틱 회귀

② 선형 회귀

③ 서포트 벡터 머신(SVM)

④ 랜덤 포레스트

⑤ k-최근접 이웃

69. k-최근접 이웃 분류기로 확률을 예측하는 방식의 한계점은?

① 확률을 전혀 계산할 수 없다

② 이웃 수에 의해 확률 종류가 제한되어 다양한 확률을 만들기 어렵다

③ 항상 100% 확률만 출력한다

④ 음수 확률이 나올 수 있다

⑤ 분류만 가능하고 확률은 불가능하다

70. 로지스틱 회귀가 선형 회귀와 다른 핵심적인 차이점은?

① 선형 방정식을 사용하지 않는다

② 계수를 학습하지 않는다

③ 선형 방정식의 출력을 그대로 사용하지 않고 시그모이드/소프트맥스 함수로 확률로 변환한다

④ 비지도 학습이다

⑤ 규제를 적용할 수 없다

71. 이진 분류에서 음성 클래스의 확률을 구하는 방법은?

① 별도의 모델을 훈련해야 한다

② 양성 클래스 확률을 제곱한다

③ 1에서 양성 클래스의 확률을 빼면 된다

④ z값에 -1을 곱한다

⑤ 소프트맥스 함수를 적용한다

72. LogisticRegression에서 C=20으로 설정한 이유는?

① 규제를 강하게 적용하기 위해

② 규제를 조금 완화하기 위해 (C가 클수록 규제가 약함)

③ 학습률을 높이기 위해

④ 에포크 수를 설정하기 위해

⑤ 클래스 수를 지정하기 위해

73. 확률적 경사 하강법에서 '경사'가 의미하는 것은?

① 데이터의 크기

② 손실 함수의 기울기

③ 특성의 수

④ 에포크의 수

⑤ 클래스의 수

74. 경사 하강법에서 한 번에 걸음이 너무 크면 발생할 수 있는 문제는?

① 항상 더 빠르게 수렴한다

② 경사를 따라 내려가지 못하고 오히려 올라갈 수 있다

③ 메모리가 부족해진다

④ 데이터가 삭제된다

⑤ 과소적합이 발생한다

75. 로지스틱 손실 함수에서 예측 확률에 로그 함수를 적용하는 이유는?

① 계산을 복잡하게 만들기 위해

② 예측이 0에 가까울 때 손실을 매우 크게 만들어 모델에 큰 영향을 미치도록 하기 위해

③ 손실 값을 항상 음수로 만들기 위해

④ 확률을 정수로 바꾸기 위해

⑤ 소프트맥스를 대체하기 위해

76. 로지스틱 손실 함수에서 타깃이 양성(1)이고 예측 확률이 0.9일 때, 손실은 높은가 낮은가?

① 높다 — 예측이 틀렸으므로

② 낮다 — 예측이 타깃에 가까우므로

③ 정확히 0이다

④ 정확히 1이다

⑤ 판단할 수 없다

77. 로지스틱 손실 함수에서 타깃이 음성(0)일 때 손실 계산 방법은?

① 예측 확률을 그대로 사용한다

② 타깃을 1로, 예측 확률을 (1-예측확률)로 바꿔서 계산한다

③ 손실을 계산하지 않는다

④ 양성 클래스와 동일하게 계산한다

⑤ 예측 확률에 2를 곱한다

78. 다중 분류에서 사용하는 손실 함수는?

① 평균 제곱 오차

② 힌지 손실

③ 크로스엔트로피 손실 함수

④ MAE 손실

⑤ R^2 손실

79. 다음 코드에서 ConvergenceWarning이 발생했다면 어떻게 해야 하는가?

```
sc = SGDClassifier(loss='log', max_iter=10)
sc.fit(train_scaled, train_target)
```

① 코드를 삭제해야 한다

② max_iter 값을 늘려서 모델이 충분히 수렴하도록 해야 한다

③ loss를 변경해야 한다

④ 데이터를 줄여야 한다

⑤ 무시하면 안 되는 심각한 오류이다

80. 확률적 경사 하강법이 대량의 데이터를 다루는 데 적합한 이유는?

① GPU가 반드시 필요하므로

② 데이터를 조금씩 사용해 점진적으로 학습할 수 있어 전체를 메모리에 올릴 필요가 없으므로

③ 데이터를 압축하므로

④ 데이터를 삭제하면서 학습하므로

⑤ 배치 크기가 항상 전체 데이터이므로

81. 다음 중 SGDClassifier의 loss 매개변수에 지정할 수 있는 값의 올바른 조합은?

① 'log'와 'mse'

② 'log'와 'hinge'

③ 'sigmoid'와 'softmax'

④ 'cross_entropy'와 'mse'

⑤ 'logistic'와 'svm'

82. 다음 중 SGDClassifier를 임포트하는 올바른 코드는?

① from sklearn.preprocessing import SGDClassifier

② from sklearn.linear_model import SGDClassifier

③ from sklearn.neighbors import SGDClassifier

④ from sklearn.metrics import SGDClassifier

⑤ from sklearn.tree import SGDClassifier

83. SGDClassifier의 alpha 매개변수의 역할과 기본값은?

① 학습률 설정, 기본값 0.01

② 규제 강도 설정, 기본값 0.0001

③ 에포크 수 설정, 기본값 1000

④ 손실 함수 선택, 기본값 'hinge'

⑤ 배치 크기 설정, 기본값 32

84. 시그모이드 함수와 소프트맥스 함수의 사용 구분으로 올바른 것은?

① 시그모이드는 다중 분류, 소프트맥스는 이진 분류

② 시그모이드는 이진 분류, 소프트맥스는 다중 분류

③ 둘 다 이진 분류에서만 사용

④ 둘 다 다중 분류에서만 사용

⑤ 둘은 동일한 함수의 다른 이름

85. LogisticRegression에서 C 값을 기본값(1)에서 20으로 올리면 모델에 어떤 변화가 생기는가?

① 규제가 강해져 모델이 단순해진다

② 규제가 약해져 모델이 더 자유롭게 학습한다

③ 에포크 수가 20으로 변경된다

④ 클래스 수가 20으로 제한된다

⑤ 학습률이 20배로 증가한다

86. SGDClassifier에서 n_iter_no_change 매개변수의 기본값은?

① 1

② 3

③ 5

④ 10

⑤ 100

87. 다음 코드에서 올바른 실행 순서를 고르시오.

```
가. lr.predict_proba(test_scaled[:5])
나. lr.fit(train_scaled, train_target)
다. lr = LogisticRegression(C=20, max_iter=1000)
```

① 다 → 나 → 가

② 가 → 나 → 다

③ 나 → 가 → 다

④ 다 → 가 → 나

⑤ 나 → 다 → 가

88. 다음 중 경사 하강법에 대해 잘못 설명한 것은?

① 손실 함수는 샘플 하나에 대한 손실을 정의한다

② 확률적 경사 하강법은 훈련 샘플을 하나씩 사용해서 손실 함수의 최솟값을 찾는다

③ 미니 배치 경사 하강법은 여러 개의 샘플을 사용한다

④ SGDClassifier는 한 번에 훈련 세트를 모두 사용하는 배치 경사 하강법을 수행한다

⑤ 배치 경사 하강법은 전체 샘플을 사용하여 경사를 계산한다

89. 다음 중 표준화(StandardScaler) 같은 데이터 전처리를 수행하지 않아도 되는 방식으로 구현된 클래스는?

① LogisticRegression

② SGDClassifier

③ KNeighborsClassifier

④ Ridge

⑤ Lasso

90. 다중 분류에서 이진 분류와 달리 클래스마다 별도의 선형 방정식을 학습하는 이유는?

① 계산 속도를 높이기 위해

② 각 클래스에 대한 z값을 따로 구해야 소프트맥스로 클래스별 확률을 계산할 수 있기 때문에

③ 이진 분류보다 정확도가 항상 낮아서 보완하기 위해

④ 사이킷런의 제한 때문에

⑤ 다중 분류에서는 시그모이드를 사용하지 않기 때문에

91. 다음 중 LogisticRegression 클래스의 설명으로 올바른 것은?

① 회귀 문제에 사용하는 모델이다

② 매개변수 C의 값을 증가시키면 규제가 강해진다

③ decision_function() 메서드는 클래스별 확률을 반환한다

④ 이진 분류와 다중 분류를 모두 수행할 수 있다

⑤ sklearn.preprocessing 패키지에 포함되어 있다

92. SGDClassifier에서 에포크 횟수를 지정하는 매개변수는?

① epochs

② max_iter

③ n_epochs

④ iterations

⑤ n_iter

93. 시그모이드 함수의 수식 φ(z) = 1/(1+e^(-z))에서 z가 매우 큰 양수일 때 출력값은?

① 0에 가까움

② 0.5

③ 1에 가까움

④ 음의 무한대

⑤ 양의 무한대

94. 시그모이드 함수의 수식 φ(z) = 1/(1+e^(-z))에서 z가 매우 큰 음수일 때 출력값은?

① 0에 가까움

② 0.5

③ 1에 가까움

④ 음의 무한대

⑤ -1에 가까움

95. 넘파이에서 시그모이드 함수를 직접 계산하는 코드로 올바른 것은?

```
z = np.arange(-5, 5, 0.1)
```

① phi = np.log(z)

② phi = 1 / (1 + np.exp(-z))

③ phi = np.sqrt(z)

④ phi = z ** 2

⑤ phi = np.abs(z)

96. 사이킷런에서 시그모이드 함수의 출력이 정확히 0.5일 때 어떤 클래스로 판단하는가?

① 양성 클래스

② 음성 클래스

③ 판단을 보류한다

④ 에러가 발생한다

⑤ 랜덤으로 결정한다

97. 소프트맥스 함수의 계산 과정으로 올바른 것은?

① 각 z값을 전체 z값의 합으로 나눈다

② 각 z값에 e^z를 계산한 후, e^z들의 전체 합(e_sum)으로 나눈다

③ 각 z값에 로그를 취한 후 합산한다

④ 각 z값의 절댓값을 구한다

⑤ 각 z값에 시그모이드를 개별 적용한다

98. 확률적 경사 하강법의 대표적인 점진적 학습 알고리즘이라 불리는 이유는?

① 한 번에 모든 데이터를 학습하므로

② 샘플을 하나씩(또는 조금씩) 사용해 기존 모델을 조금씩 업데이트할 수 있으므로

③ 데이터를 삭제하면서 학습하므로

④ 모델을 매번 처음부터 다시 만들어야 하므로

⑤ 테스트 세트로 학습하므로

99. 다음 보기에서 predict_proba()와 decision_function()의 차이로 올바른 것은?

① predict_proba()는 z값을, decision_function()은 확률을 반환한다

② predict_proba()는 클래스별 확률을, decision_function()은 선형 방정식의 출력(z값)을 반환한다

③ 둘은 동일한 값을 반환한다

④ predict_proba()는 훈련에, decision_function()은 예측에 사용된다

⑤ predict_proba()는 회귀에만, decision_function()은 분류에만 사용된다

100. 다음 중 fit() 메서드와 partial_fit() 메서드를 모두 제공하는 클래스는?

① LinearRegression

② KNeighborsClassifier

③ SGDClassifier

④ StandardScaler

⑤ PolynomialFeatures

정답 및 해설

1번 정답: ②

[해설] 교재에서 '로지스틱 회귀는 이름은 회귀이지만 분류 모델'이라고 명시하고 있다. 선형 방정식을 학습하되, 그 출력을 시그모이드 또는 소프트맥스 함수로 변환하여 클래스 확률을 출력하는 분류 알고리즘이다.

2번 정답: ③

[해설] 교재에서 '이 알고리즘은 선형 회귀와 동일하게 선형 방정식을 학습'한다고 설명하고 있다. 다만 그 출력값을 그대로 사용하지 않고 시그모이드나 소프트맥스 함수를 통과시켜 확률로 변환한다.

3번 정답: ②

[해설] 교재에서 '시그모이드 함수를 사용하여 z가 아주 큰 음수일 때 0이 되고, z가 아주 큰 양수일 때 1이 되도록 바꿀 수 있다'고 설명하고 있다. 시그모이드 함수는 이진 분류에서 사용된다.

4번 정답: ③

[해설] 시그모이드 함수는 어떤 입력이든 0과 1 사이의 값으로 압축한다. 교재 그래프에서도 z값이 아무리 커도 1에 수렴하고, 아무리 작아도 0에 수렴하는 것을 보여주고 있다.

5번 정답: ②

[해설] 교재에서 '이진 분류일 경우 시그모이드 함수의 출력이 0.5보다 크면 양성 클래스, 0.5보다 작으면 음성 클래스로 판단'한다고 설명하고 있다.

6번 정답: ③

[해설] 시그모이드 함수 φ(z) = 1/(1+e^(-z))에서 z=0이면 φ(0) = 1/(1+e^0) = 1/(1+1) = 0.5이다. 교재 확인 문제(슬라이드 29)에서도 동일한 문제를 다루고 있다.

7번 정답: ②

[해설] 교재에서 '다중 분류는 이와 달리 소프트맥스(softmax) 함수를 사용하여 7개의 z 값을 확률로 변환'한다고 설명하고 있다. 시그모이드는 이진 분류, 소프트맥스는 다중 분류에 사용된다.

8번 정답: ②

[해설] 교재에서 '소프트맥스 함수는 여러 개의 선형 방정식의 출력값을 0~1 사이로 압축하고 전체 합이 1이 되도록 한다'고 설명하고 있다. 이를 통해 각 출력을 클래스별 확률로 해석할 수 있다.

9번 정답: ②

[해설] 교재에서 '다중 분류(multiclass classification): 타깃 데이터에 2개 이상의 클래스가 포함'되는 것이라고 정의하고 있다. 교재 확인 문제(슬라이드 28)에서도 동일한 질문을 다루고 있다.

10번 정답: ②

[해설] 교재에서 'LogisticRegression 클래스는 선형 모델 sklearn.linear_model 패키지 안에 있다'고 명시하고 있다.

11번 정답: ③

[해설] 교재에서 '사이킷런의 분류 모델은 predict_proba() 메서드로 클래스별 확률값을 반환'한다고 설명하고 있다.

12번 정답: ②

[해설] 교재에서 'decision_function(): 모델이 학습한 선형 방정식의 출력을 반환'한다고 설명하고 있다. 이 z값에 시그모이드나 소프트맥스를 적용하면 확률이 된다.

13번 정답: ②

[해설] 교재에서 '타깃값을 그대로 사이킷런 모델에 전달하면 순서가 자동으로 알파벳 순으로 매겨진다'고 설명하고 있다. 정렬된 타깃값은 classes_ 속성에 저장된다.

14번 정답: ③

[해설] 교재에서 'KNeighborsClassifier에서 정렬된 타깃값은 classes_ 속성에 저장된다'고 설명하고 있다. LogisticRegression에서도 동일하게 classes_ 속성을 사용한다.

15번 정답: ③

[해설] 교재에서 '5개의 특성을 사용하므로 coef_ 배열의 열은 5개, 행이 7'이라고 설명하고 있다. 다중 분류에서는 클래스마다 선형 방정식 하나씩을 학습하므로 행 수가 클래스 수와 같다.

16번 정답: ②

[해설] 교재에서 '다중 분류는 클래스마다 z 값을 하나씩 계산하고 가장 높은 z 값을 출력하는 클래스가 예측 클래스가 된다'고 설명하고 있다.

17번 정답: ②

[해설] 교재에서 'LogisticRegression에서 규제를 제어하는 매개변수는 C'라고 명시하고 있다. C는 릿지의 alpha와 반대로 작을수록 규제가 커진다.

18번 정답: ②

[해설] 교재에서 '릿지 회귀에서는 alpha가 커지면 규제도 커지지만, LogisticRegression의 C는 alpha와 반대로 작을수록 규제가 커진다'고 설명하고 있다.

19번 정답: ③

[해설] 교재에서 'C의 기본값은 1'이라고 명시하고 있다. 교재 예제에서는 규제를 조금 완화하기 위해 C=20으로 증가시켰다.

20번 정답: ②

[해설] 교재에서 'LogisticRegression은 기본적으로 릿지 회귀와 같이 계수의 제곱을 규제하며, 이를 L2 규제라고 한다'고 설명하고 있다.

21번 정답: ③

[해설] 교재에서 'max_iter 매개변수에서 반복 횟수를 지정하며 기본값은 100'이라고 명시하고 있다. 데이터에 따라 부족할 수 있어 교재에서는 1000으로 늘려 사용했다.

22번 정답: ②

[해설] 교재에서 '불리언 인덱싱: 넘파이 배열은 True, False 값을 전달하여 행을 선택할 수 있다'고 설명하고 있다. 예를 들어 ['A','B','C']에서 [True, False, True]를 전달하면 'A'와 'C'만 선택된다.

23번 정답: ②

[해설] 교재에서 '첫 번째 열이 음성 클래스(0)에 대한 확률, 두 번째 열이 양성 클래스(1)에 대한 확률'이라고 설명하고 있다. 출력 순서는 classes_ 속성의 순서와 동일하다.

24번 정답: ②

[해설] 교재에서 소프트맥스 함수는 '지수 함수를 사용하기 때문에 정규화된 지수 함수라고도 부른다'고 설명하고 있다. e^z를 계산한 후 전체 합으로 나누어 정규화한다.

25번 정답: ③

[해설] 교재에서 '파이썬의 사이파이(scipy) 라이브러리 시그모이드 함수 expit()으로 decisions 배열의 값을 확률로 변환'한다고 설명하고 있다.

26번 정답: ②

[해설] 교재에서 '새로운 생선이 도착하는 대로 가능한 즉시 훈련 데이터를 제공해야 하는' 상황을 소개하며, 점진적 학습(온라인 학습)의 필요성을 설명하고 있다.

27번 정답: ②

[해설] 교재에서 '확률적 경사 하강법에서 확률적이란 말은 무작위하게 혹은 랜덤하게의 기술적인 표현'이라고 설명하고 있다.

28번 정답: ②

[해설] 교재에서 '전체 샘플을 사용하지 않고 훈련 세트에서 랜덤하게 하나의 샘플을 고르는 것이 확률적 경사 하강법'이라고 설명하고 있다.

29번 정답: ②

[해설] 교재에서 '미니배치 경사 하강법: 여러 개의 샘플을 사용해 경사 하강법을 수행하는 방식'이라고 정의하고 있다. 교재 확인 문제(슬라이드 49)에서도 동일한 질문을 다루고 있다.

30번 정답: ③

[해설] 교재에서 '배치 경사 하강법: 극단적으로 한 번 경사로를 따라 이동하기 위해 전체 샘플을 사용'한다고 정의하고 있다.

31번 정답: ③

[해설] 교재에서 '에포크(epoch): 확률적 경사 하강법에서 훈련 세트를 한 번 모두 사용하는 과정'이라고 정의하고 있다. 일반적으로 수십~수백 번 이상 에포크를 수행한다.

32번 정답: ②

[해설] 교재에서 '손실 함수는 어떤 문제에서 머신러닝 알고리즘이 얼마나 엉터리인지를 측정하는 기준이며, 손실 함수의 값이 작을수록 좋다'고 설명하고 있다.

33번 정답: ②

[해설] 교재에서 '손실 함수는 미분 가능해야 한다'고 설명하고 있다. 경사 하강법은 손실 함수의 기울기(미분)를 이용하여 경사를 따라 내려가므로, 미분이 불가능하면 사용할 수 없다.

34번 정답: ②

[해설] 교재 예시에서 4개 예측 중 2개만 맞으면 정확도 0.5인데, 이런 불연속적 값은 미분할 수 없다. 그래서 연속적인 손실 함수(예: 로지스틱 손실)를 대신 사용한다.

35번 정답: ②

[해설] 교재에서 '이진 분류에는 로지스틱 회귀(또는 이진 크로스엔트로피) 손실함수를 사용'한다고 설명하고 있다. '로지스틱 손실 함수'와 '크로스엔트로피 손실 함수'는 같은 함수의 다른 이름이다.

36번 정답: ④

[해설] 교재에서 '회귀 문제에는 평균 제곱 오차 손실 함수를 사용'한다고 설명하고 있다.

37번 정답: ②

[해설] 교재에서 'SGDClassifier는 확률적 경사 하강법을 사용한 분류 모델을 만든다'고 설명하고 있다. sklearn.linear_model 패키지에 포함되어 있다.

38번 정답: ③

[해설] 교재에서 'SGDClassifier의 loss 매개변수의 기본값은 hinge'라고 명시하고 있다. 힌지 손실은 서포트 벡터 머신(SVM)을 위한 손실 함수이다. 로지스틱 회귀를 사용하려면 'log'로 지정해야 한다.

39번 정답: ②

[해설] 교재에서 'loss=log로 지정하여 로지스틱 손실 함수를 지정'한다고 설명하고 있다.

40번 정답: ③

[해설] 교재에서 'partial_fit() 메서드를 사용하여 SGDClassifier 객체를 다시 만들지 않고 훈련한 모델을 추가로 더 훈련'할 수 있다고 설명하고 있다. 호출할 때마다 1 에포크씩 이어서 훈련한다.

41번 정답: ②

[해설] 교재에서 'partial_fit() 메서드는 fit() 메서드와 사용법이 같지만 호출할 때마다 1 에포크씩 이어서 훈련'한다고 설명하고 있다. 기존 모델을 버리지 않고 추가 학습이 가능하다.

42번 정답: ②

[해설] 교재에서 '조기 종료(early stopping): 과대적합이 시작하기 전에 훈련을 멈추는 것'이라고 정의하고 있다. 에포크 그래프에서 테스트 점수가 떨어지기 직전이 최적의 멈춤 지점이다.

43번 정답: ②

[해설] 교재에서 '적은 에포크 횟수는 과소적합된 모델일 가능성이 높고, 많은 에포크 횟수는 과대적합된 모델일 가능성이 높다'고 설명하고 있다.

44번 정답: ③

[해설] 교재에서 '힌지 손실(hinge loss)은 서포트 벡터 머신(support vector machine)이라 불리는 또 다른 머신러닝 알고리즘을 위한 손실 함수'라고 설명하고 있다.

45번 정답: ②

[해설] 교재에서 'penalty 매개변수에서 규제의 종류를 지정하며, 기본값은 L2 규제를 위한 l2'라고 설명하고 있다.

46번 정답: ④

[해설] 교재에서 'max_iter 매개변수는 에포크 횟수를 지정하며, 기본값은 1000'이라고 명시하고 있다.

47번 정답: ③

[해설] 교재에서 'tol 매개변수는 반복을 멈출 조건으로, n_iter_no_change 에포크 동안 손실이 tol만큼 줄어들지 않으면 알고리즘이 중단된다'고 설명하고 있다. 기본값은 0.001이다.

48번 정답: ②

[해설] 교재에서 'SGDRegressor는 확률적 경사 하강법을 사용한 회귀 모델을 만든다'고 설명하고 있다. SGDClassifier와 동일한 매개변수를 사용한다.

49번 정답: ③

[해설] 교재에서 'SGDRegressor의 loss 매개변수 기본값은 제곱 오차를 나타내는 squared_loss'라고 명시하고 있다.

50번 정답: ③

[해설] 교재에서 '이웃한 샘플의 클래스를 확률로 삼는다면 삼각형일 확률은 50%'라고 설명하고 있다. 5/10 = 50%이다. k-최근접 이웃은 이웃의 클래스 비율을 확률로 사용한다.

51번 정답: ②

[해설] 교재에서 '판다스의 unique() 함수를 사용하여 Species 열에서 고유한 값을 추출'한다고 설명하고 있다. 결과로 7종의 생선 이름이 반환된다.

52번 정답: ②

[해설] 교재에서 '데이터프레임에서 여러 열을 선택하면 새로운 데이터프레임이 반환된다'고 설명하고 있다. Weight, Length, Diagonal, Height, Width 5개 열이 입력 데이터로 사용된다.

53번 정답: ②

[해설] 교재에서 '훈련 세트의 통계 값으로 테스트 세트를 변환해야 한다는 점에 주의'하라고 강조하고 있다. ss.fit(train_input)으로 학습한 후, 동일한 기준으로 test_input도 변환한다.

54번 정답: ②

[해설] 교재에서 '타깃값을 그대로 사이킷런 모델에 전달하면 순서가 자동으로 알파벳 순으로 매겨진다'고 설명하고 있다. classes_ 속성에서 이 정렬된 순서를 확인할 수 있다.

55번 정답: ②

[해설] 교재에서 'predict_proba() 메서드의 출력 순서는 앞서 보았던 classes_ 속성과 같다'고 설명하고 있다. 즉 classes_의 첫 번째 클래스가 첫 번째 열의 확률이 된다.

56번 정답: ②

[해설] 불리언 인덱싱을 위해 train_target이 'Bream'이거나(|) 'Smelt'인 행을 True로 표시하는 불리언 배열을 만든다. 이 배열로 훈련 세트에서 도미와 빙어 데이터만 골라낼 수 있다.

57번 정답: ②

[해설] 교재에서 알파벳순으로 정렬했을 때 Bream이 먼저, Smelt가 뒤에 오므로, '빙어(Smelt)가 양성 클래스'라고 설명하고 있다. 사이킷런은 classes_의 두 번째 클래스를 양성으로 취급한다.

58번 정답: ②

[해설] 교재에서 '로지스틱 회귀가 학습한 계수 확인'이라고 설명하며 coef_와 intercept_를 출력하고 있다. 이 값들로 선형 방정식 z = a×Weight + b×Length + ... + f를 구성한다.

59번 정답: ③

[해설] 교재에서 decision_function()으로 z값을 구한 뒤 expit()(시그모이드 함수)을 적용하면 양성 클래스에 대한 확률이 된다고 설명하고 있다. 이 값은 predict_proba()의 두 번째 열(양성 클래스 확률)과 동일하다.

60번 정답: ②

[해설] 교재에서 '5개의 특성을 사용하므로 coef_ 배열의 열은 5개, 행이 7'이라고 설명하고 있다. 행 7은 클래스 수(7종 생선), 열 5는 특성 수(Weight, Length, Diagonal, Height, Width)이다.

61번 정답: ②

[해설] 교재에서 'axis=1로 지정하여 각 행, 즉 각 샘플에 대해 소프트맥스를 계산'한다고 설명하고 있다. axis를 지정하지 않으면 배열 전체에 대해 계산되어 잘못된 결과가 나온다.

62번 정답: ②

[해설] 교재에서 '넘파이 round() 함수는 decimals 매개변수로 유지할 소수점 아래 자릿수를 지정할 수 있다'고 설명하고 있다. decimals=4이면 소수점 네 번째 자리까지 표기하고 다섯 번째 자리에서 반올림한다.

63번 정답: ②

[해설] 교재에서 'loss=log로 지정하여 로지스틱 손실 함수를 지정하고, max_iter는 수행할 에포크 횟수를 지정하며 10으로 지정하여 전체 훈련 세트를 10회 반복'한다고 설명하고 있다.

64번 정답: ②

[해설] 교재에서 'partial_fit() 메서드는 호출할 때마다 1 에포크씩 이어서 훈련'한다고 설명하고 있다. 기존에 학습한 가중치를 유지하면서 추가 학습이 이루어진다.

65번 정답: ②

[해설] 교재에서 'partial_fit() 메서드에 훈련 세트에 있는 전체 클래스의 레이블을 전달'한다고 설명하고

있다. 점진적 학습 시 한 번에 일부 클래스만 보일 수 있으므로, 전체 클래스 목록을 미리 알려줘야 한다.

66번 정답: ②

[해설] 교재에서 에포크 그래프를 보고 '이 모델의 경우 백 번째 에포크가 적절한 반복 횟수'라고 판단하고 있다. 이 지점 이후로는 과대적합이 시작되어 테스트 점수가 떨어지기 시작한다.

67번 정답: ②

[해설] tol 매개변수는 손실이 일정량 이상 줄어들지 않으면 조기 종료하는 조건이다. tol=None으로 설정하면 이 자동 조기 종료를 비활성화하여, max_iter에 지정한 에포크 수만큼 정확히 반복한다.

68번 정답: ③

[해설] 교재에서 '힌지 손실(hinge loss)은 서포트 벡터 머신이라 불리는 또 다른 머신러닝 알고리즘을 위한 손실 함수'라고 설명하고 있다.

69번 정답: ②

[해설] 교재에서 k=3이면 가능한 확률이 0/3, 1/3, 2/3, 3/3으로 4가지뿐이라는 한계를 암시하고 있다. 이웃 수에 따라 확률 해상도가 제한되므로, 이것이 로지스틱 회귀가 필요한 이유이다.

70번 정답: ③

[해설] 교재에서 '선형 회귀처럼 계산한 값을 그대로 출력하는 것이 아니라, 로지스틱 회귀는 이 값을 0~1 사이로 압축한다'고 설명하고 있다. 이것이 회귀와 분류의 결정적 차이이다.

71번 정답: ③

[해설] 교재에서 '음성 클래스의 확률은 1에서 양성 클래스의 확률을 빼면 된다'고 설명하고 있다. 이진 분류에서 두 클래스의 확률 합은 항상 1이기 때문이다.

72번 정답: ②

[해설] 교재에서 C는 '작을수록 규제가 커지고' 기본값은 1이다. '규제를 조금 완화하기 위해 20으로 증가시켰다'고 설명하고 있다. 즉 C를 크게 하면 규제가 약해진다.

73번 정답: ②

[해설] 교재에서 '경사는 기울기를 뜻한다'고 설명하고 있다. 경사 하강법은 손실 함수의 기울기(미분)를 따라 가장 가파른 방향으로 내려가면서 최적의 모델을 찾는 알고리즘이다.

74번 정답: ②

[해설] 교재에서 '한번에 걸음이 너무 크면 경사를 따라 내려가지 못하고 오히려 올라갈 수도 있다'고 설명하고 있다. 이것이 적절한 학습률(걸음 크기)이 중요한 이유이다.

75번 정답: ②

[해설] 교재에서 '로그 함수는 0에 가까울수록 아주 큰 음수가 되기 때문에 손실을 아주 크게 만들어 모델에 큰 영향을 미칠 수 있다'고 설명하고 있다. 또한 0~1 사이에서 로그 함수가 음수이므로 부호를 바꾸면 손실이 양수가 되어 해석이 쉬워진다.

76번 정답: ②

[해설] 교재에서 타깃이 1이고 예측이 0.9이면 '예측이 1에 가까울수록 손실이 작아진다'고 설명하고 있다. 0.9는 1에 가까우므로 손실이 낮다.

77번 정답: ②

[해설] 교재에서 '타깃을 마치 양성 클래스처럼 바꾸어 1로 만들고, 대신 예측값도 양성 클래스에 대한 예측으로 바꾼다(1-예측확률)'고 설명하고 있다. 그다음 곱하고 음수로 바꾸는 과정은 양성 클래스와 동일하다.

78번 정답: ③

[해설] 교재에서 '다중 분류에는 크로스엔트로피 손실 함수를 사용한다'고 설명하고 있다.

79번 정답: ②

[해설] 교재에서 'max_iter 매개변수의 값을 늘려 주는 것이 바람직하다'고 설명하고 있다. ConvergenceWarning은 오류가 아닌 경고이므로 실습은 진행되지만, 충분히 수렴하지 않았다는 신호이다.

80번 정답: ②

[해설] 교재에서 '데이터가 매우 크기 때문에 전통적인 방식으로 모델을 만들기 어렵고, 데이터를 조금씩 사용해 점진적으로 학습하는 방법이 필요하다'고 설명하며, 확률적 경사 하강법이 그 핵심 열쇠라고 설명하고 있다.

81번 정답: ②

[해설] 교재에서 SGDClassifier의 loss에 'log'(로지스틱 회귀)와 'hinge'(서포트 벡터 머신)를 모두 사용하는 예제를 보여주고 있다. 기본값은 'hinge'이다.

82번 정답: ②

[해설] 교재에서 'SGDClassifier를 sklearn.linear_model 패키지 아래에서 임포트'한다고 설명하고 있다. LogisticRegression과 같은 패키지에 있다.

83번 정답: ②

[해설] 교재에서 '규제 강도는 alpha 매개변수에서 지정하며, 기본값은 0.0001'이라고 명시하고 있다. LogisticRegression의 C와 달리 alpha는 값이 클수록 규제가 강해진다.

84번 정답: ②

[해설] 교재에서 '이진 분류에서는 시그모이드 함수를 사용하고, 다중 분류에서는 소프트맥스 함수를 사용'한다고 명확히 구분하고 있다.

85번 정답: ②

[해설] C는 작을수록 규제가 강해지므로, C를 1에서 20으로 올리면 규제가 약해진다. 교재에서도 '규제를 조금 완화하기 위해 20으로 증가시켰다'고 설명하고 있다.

86번 정답: ③

[해설] 교재에서 'n_iter_no_change 매개변수의 기본값은 5'라고 명시하고 있다. tol과 함께 사용되어, 5 에포크 동안 손실이 tol(0.001)만큼 줄어들지 않으면 훈련이 중단된다.

87번 정답: ①

[해설] 사이킷런 모델 사용 순서는 항상 ① 객체 생성(다) → ② 훈련(나: fit) → ③ 예측/확률(가: predict_proba)이다. 교재의 모든 코드 예제에서 이 순서를 일관되게 따르고 있다.

88번 정답: ④

[해설] 교재 확인 문제(슬라이드 50)에서 동일한 질문을 다루고 있다. SGDClassifier는 이름 그대로 확률적(Stochastic) 경사 하강법을 수행하는 클래스이지, 배치 경사 하강법이 아니다.

89번 정답: ③

[해설] 교재 확인 문제(슬라이드 49)에서 동일한 질문을 다루고 있다. k-최근접 이웃은 거리 기반이지만 사이킷런의 KNeighborsClassifier는 전처리 없이도 동작하도록 구현되어 있다. 나머지 선형 모델 기반 클래스들은 스케일에 민감하므로 표준화가 필요하다.

90번 정답: ②

[해설] 교재에서 '다중 분류는 클래스마다 z 값을 하나씩 계산'해야 하고, 이 여러 개의 z값을 소프트맥스 함수에 넣어 전체 합이 1이 되는 확률로 변환한다고 설명하고 있다.

91번 정답: ④

[해설] 교재 확인 문제(슬라이드 29)에서 동일한 질문을 다루고 있다. LogisticRegression은 이름은 회귀이지만 분류 모델이고(①틀림), C가 커지면 규제가 약해지고(②틀림), decision_function()은 z값을 반환하며(③틀림), sklearn.linear_model에 있다(⑤틀림). 이진 분류와 다중 분류를 모두 수행할 수 있다(④맞음).

92번 정답: ②

[해설] 교재 확인 문제(슬라이드 50)에서 동일한 질문을 다루고 있다. max_iter 매개변수로 에포크 횟수를 지정하며, 기본값은 1000이다.

93번 정답: ③

[해설] z가 매우 큰 양수이면 e^(-z)는 0에 가까워지고, φ(z) = 1/(1+0) = 1에 가까워진다. 교재에서도 '시그모이드 함수를 사용하여 z가 아주 큰 양수일 때 1이 되도록 바꿀 수 있다'고 설명하고 있다.

94번 정답: ①

[해설] z가 매우 큰 음수이면 e^(-z)는 매우 큰 양수가 되어, φ(z) = 1/(1+매우 큰 수) ≈ 0에 가까워진다. 교재에서도 '시그모이드 함수를 사용하여 z가 아주 큰 음수일 때 0이 되도록 바꿀 수 있다'고 설명하고 있다.

95번 정답: ②

[해설] 교재에서 '지수 함수 계산은 np.exp() 함수를 사용'하여 phi = 1 / (1 + np.exp(-z))로 시그모이드 함수를 계산하고 있다. 이 공식은 시그모이드 함수의 수학적 정의 φ(z) = 1/(1+e^(-z))를 그대로 코드로 옮긴 것이다.

96번 정답: ②

[해설] 교재에서 '정확히 0.5일 때는 라이브러리마다 다를 수 있지만, 사이킷런은 0.5일 때 음성 클래스로 판단'한다고 설명하고 있다.

97번 정답: ②

[해설] 교재에서 소프트맥스 계산을 'e_sum = e^z1 + e^z2 + ... + e^z7이고, s1 = e^z1/e_sum, s2 = e^z2/e_sum, ...'으로 설명하고 있다. 각 z값에 지수 함수를 적용한 후 전체 합으로 나누어 정규화한다.

98번 정답: ②

[해설] 교재에서 '대표적인 점진적 학습 알고리즘은 확률적 경사 하강법'이라고 소개하고 있다. 새로운 데이터가 도착할 때마다 partial_fit()으로 기존 모델에 추가 학습시킬 수 있기 때문이다.

99번 정답: ②

[해설] 교재에서 'predict_proba() 메서드: 예측 확률을 반환'하고, 'decision_function(): 모델이 학습한 선형 방정식의 출력을 반환'한다고 구분하고 있다. z값에 시그모이드/소프트맥스를 적용하면 predict_proba()의 결과와 같아진다.

100번 정답: ③

[해설] 교재에서 SGDClassifier는 fit()으로 처음부터 훈련할 수도 있고, partial_fit()으로 기존 모델에 1 에포크씩 이어서 훈련(점진적 학습)할 수도 있다고 설명하고 있다. 이는 확률적 경사 하강법 기반 클래스의 특징이다.