**1. 인공지능에서 지능에 해당하는 기능은 무엇인가?**

• 학습 (Learning)

• 추론 (Reasoning)

• 문제 해결 (Problem Solving)

• 지각/인식 (Perception)

• 자연어 처리 (Natural Language Processing)

• 계획 (Planning)

**2. 인공지능의 종류 3가지 (지도학습, 반지도학습, 강화학습)**

• **지도학습**: 입력-정답 쌍으로 학습 (예: 이메일 분류, 집값 예측)

• **반지도학습**: 일부만 라벨이 있는 데이터 사용

• **강화학습**: 보상을 바탕으로 최적의 행동 학습 (예: 게임, 알파고)

**3. 전통적인 프로그래밍 vs 인공지능 프로그래밍**

• **전통적**: 규칙(로직)과 데이터를 프로그래머가 정의 → 결과 생성

• **AI 방식**: 데이터와 정답을 제공 → 모델이 규칙(패턴)을 스스로 학습

**4. 딥러닝 vs 머신러닝**

• **머신러닝**: 데이터 기반 학습 일반 알고리즘 (SVM, Decision Tree 등)

• **딥러닝**: 인공신경망 기반, 대규모 데이터/복잡한 문제에 강함

**5. Classification vs Regression**

• **분류(Classification)**: 카테고리 예측 (예: 고양이/강아지)

• **회귀(Regression)**: 연속적인 수치 예측 (예: 집값, 온도)

**6. 차원의 저주 (Curse of Dimensionality)**

• 데이터의 차원이 많아질수록, 계산량 증가 / 데이터 희소해짐 → 학습 어려움

**7. Dimensionality Reduction가 필요한 이유**

• 고차원 데이터를 저차원으로 줄여서 계산 효율성, 시각화, 노이즈 제거

**8. Ridge vs Lasso**

• 공통: 정규화(Regularization) 기법 → 과적합 방지

• **Ridge**: L2, 모든 특성 유지하되 가중치 축소

• **Lasso**: L1, 불필요한 특성의 가중치를 0으로 만들어 **변수 선택 효과**

**9. Overfitting vs Underfitting**

• **Overfitting**: 훈련 데이터에 너무 맞춰져 일반화 실패

• **Underfitting**: 너무 단순한 모델 → 훈련조차 잘 안됨

**10. Feature Engineering vs Feature Selection**

• **Engineering**: 새 특성 만드는 과정

• **Selection**: 중요하지 않은 특성을 제거

**11. 전처리(Preprocessing)의 목적과 방법**

• 목적: 모델 성능 향상, 이상치/결측치/노이즈 제거

• 방법: 스케일링, 결측값 대체, 이상치 처리 등

**12. EDA (탐색적 데이터 분석)**

• 데이터 분포, 통계치, 상관관계 등 파악

• 시각화(히스토그램, 박스플롯), 통계 요약 등을 포함

**13. 회귀에서 절편과 기울기의 의미**

• **절편**: 입력이 0일 때 출력값

• **기울기**: 입력 1 증가 시 출력 변화량

• 딥러닝에서도 가중치(weight), 편향(bias)으로 동일 개념 적용됨

**14. 교차검증 vs K-Fold 교차검증**

• **교차검증**: 데이터를 나눠서 훈련/평가 반복

• **K-Fold**: 데이터를 K등분 → K번 훈련/평가 수행 → 평균 성능 계산

**15. 하이퍼파라미터 튜닝이란?**

• 모델 외부 설정값 (예: 학습률, 트리 깊이 등)을 조절하여 성능 최적화

**16. 결정트리에서 불순도 - 지니 계수(Gini Index)**

• 노드의 혼합 정도(불순도)를 나타냄

• 낮을수록 더 순수함 (즉, 한 클래스가 지배적일수록 좋음)

**17. 앙상블이란?**

• 여러 모델을 조합하여 더 나은 예측을 수행하는 기법

• 예: 배깅, 부스팅, 랜덤 포레스트

**18. 부트스트랩핑 (Bootstrapping)**

• 원본 데이터를 **중복 허용**하며 여러 샘플을 만들어 학습에 활용하는 방식

**19. 배깅 (Bagging)**

• 부트스트랩 데이터를 사용해 여러 모델 학습 → 예측 결과를 평균/투표

• **분산 감소**, 과적합 줄임

**20. PCA (주성분 분석)**

• 고차원 데이터를 저차원으로 축소하는 방법

• 가장 많은 정보를 담는 축(주성분)을 찾아 데이터 투영

• 데이터 압축, 시각화, 노이즈 제거에 유용

필기로 정리할 땐 각 항목을 **요점 위주로 요약**해서 쓰는 게 좋아.

이제 이걸 참고해서 손으로 정리하고 사진 찍어 GitHub에 올리면 돼.