**1. 서포트 벡터 머신의 근본적인 아이디어는?**

두 클래스를 가장 멀리 떨어뜨리는 최적의 결정 경계를 찾는 것.

이 최대 margin을 가지는 초평면이 결정 경계가 됨.

**2. 서포트 벡터가 무엇인가?**

결정 경계에 가장 가까운 훈련 샘플.

이 포인트들만 모델에 영향을 주고, 결정 경계를 정의함.

**3. SVM을 사용할 때 입력값의 스케일이 왜 중요한가?**

스케일이 다르면 거리 계산에 왜곡이 생겨 잘못된 결정 경계 생성됨.

해결 방법: StandardScaler로 전처리 (평균 0, 분산 1로 정규화)

**4. SVM 분류기가 샘플을 분류할 때 신뢰도 점수와 확률을 출력할 수 있는가?**

기본적으로는 decision\_function()으로 점수 출력 가능.

확률은 probability=True로 설정 후 predict\_proba() 사용하면 출력 가능함.

**5. RBF 커널을 사용해 SVM 분류기를 훈련시켰더니 과소적합되었다. 해결 방법은?**

C 값을 증가시키거나, gamma 값을 증가시켜 모델 복잡도를 높임.

* C ↑ : 규제 완화 → 마진 좁아짐
* gamma ↑ : 결정 경계 더 유연하게

**6. (심화) MNIST 데이터셋에 SVM 분류기 훈련**

**결과:**

train\_size=30000, test\_size=1000으로 훈련

classification\_report 결과 평균 정확도: **0.95**

다중 클래스 분류에 SVC() 사용, OvR 전략 자동 적용

SVM 분류기를 MNIST 손글씨 데이터에 적용하였다.

train size: 30,000 / test size: 1,000 샘플 사용.

정밀도, 재현율, f1-score 모두 약 0.95 수준의 높은 성능을 보였으며,

SVC 모델은 다중 클래스 분류에서 OvR 전략으로 잘 작동하였다.

7. SVM 회귀 (SVR with RBF kernel)를 캘리포니아 주택 가격 데이터에 적용하였고,

StandardScaler로 스케일링 후 RMSE는 약 0.57로 측정됨.

이는 평균적으로 약 0.57의 오차로 예측했음을 의미함.