5장 서포트 벡터 머신

감사의 글

자료를 공개한 저자 오렐리앙 제롱과 강의자료를 지원한 한빛아카데미에게 진심어린 감사를 전합니다.

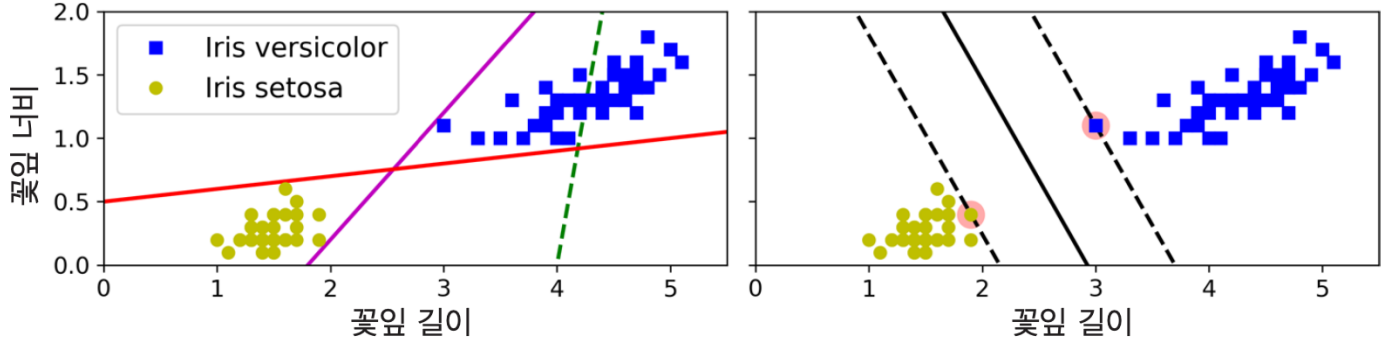
주요 내용

- 선형 SVM 분류
- 비선형 SVM 분류
- SVM 회귀
- SVM 이론

5.1 선형 SVM 분류

기본 아이디어

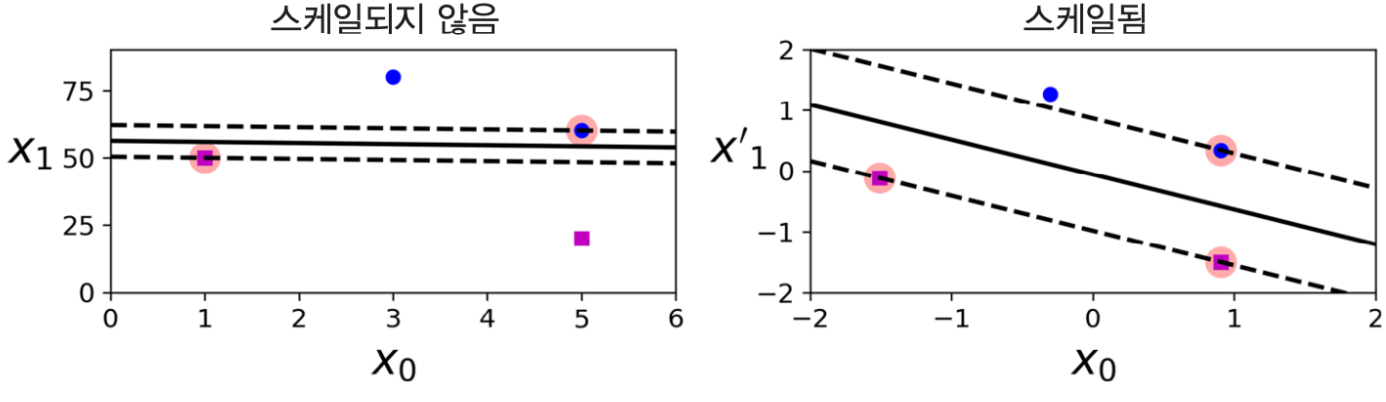
- 마진(margin): 클래스를 구분하는 도로의 경계
- 라지 마진 분류: 마진 폭을 최대로 하는 클래스 분류



	왼편 그래프	오른편 그래프
분류기:	선형 분류	라지 마진 분류
실선:	결정 경계	결정 경계
일반화:	일반화 어려움	일반화 쉬움

서포트 벡터

- 도로의 양쪽 경계에 위치하는 샘플 (아래 그림에서 동그라미 표시됨)
- 서포트 벡터 사이의 간격, 즉 도로의 폭이 최대가 되도록 학습
- 특성 스케일을 조정하면 결정경계가 훨씬 좋아짐.

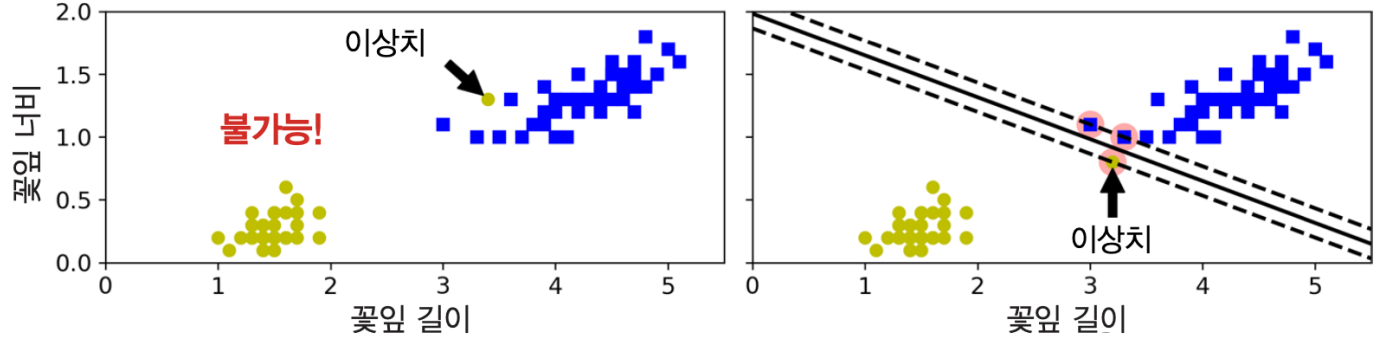


서포트 벡터 머신(SVM) 모델

- 두 클래스로부터 최대한 멀리 떨어져 있는 결정 경계를 찾는 분류기
- 목표: 특정 조건을 만족하면서 동시에 클래스를 분류하는 가능한 넓은 도로의 결정 경계 찾기

하드 마진 분류

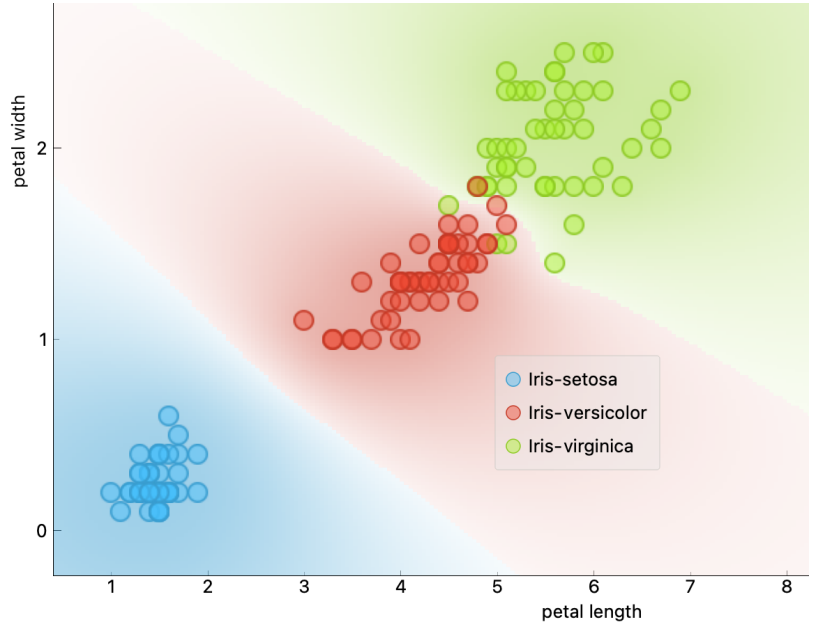
- 모든 훈련 샘플이 도로 바깥쪽에 올바르게 분류되도록 하는 마진 분류
- 훈련 세트가 선형적으로 구분되는 경우에만 가능
- 이상치에 민감함



	왼편 그래프	오른편 그래프
이상치:	타 클래스에 섞임	타 클래스에 매우 가까움
하드 마진 분류:	불가능	가능하지만 일반화 어려움

소프트 마진 분류

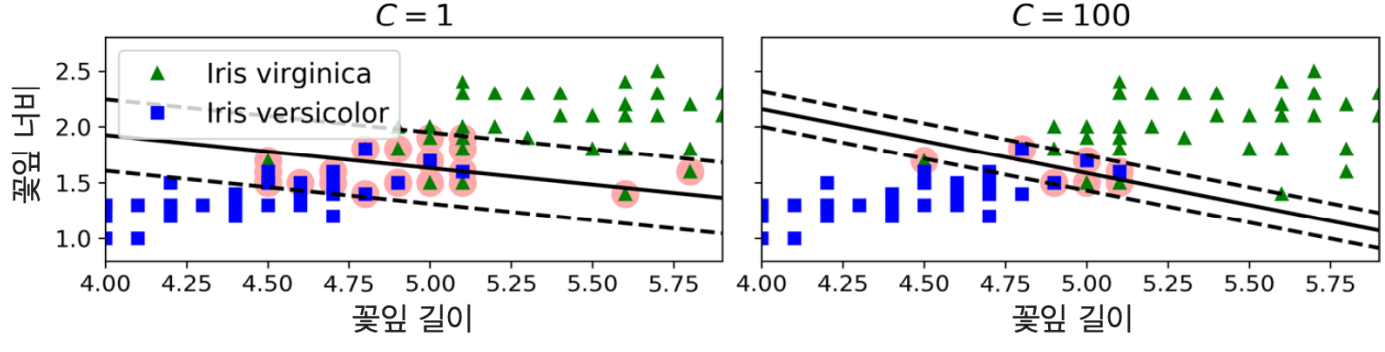
- 마진 위반(margin violation) 사례의 발생 정도를 조절하면서 도로의 폭을 최대로 넓게 유지하는 마진 분류
- **마진 위반:** 훈련 샘플이 도로 상에 위치하거나 결정 경계를 넘어 해당 클래스 반대편에 위치하는 샘플
- 하드 마진 분류 불가능 예제: 꽃잎 길이와 너비 기준의 버지니카와 버시컬러 품종



예제: 버지니까 품종 여부 판단

- 사이킷런의 선형 SVM 분류기 `LinearSVC` 활용
    - `C` : 무조건 양수이어야 하며 클 수록 마진 위반을 적게, 즉 도로폭을 작게 만듦. 결국 규제를 덜 가하게 되어 모델의 자유도를 올려 과대적합 가능성을 키움.
    - `hinge` : 힌지 손실. 예측값과 실제 라벨 사이의 차이가 클 수록 큰 손실이 가해짐.

```
svm_clf1 = LinearSVC(C=1, loss="hinge", random_state=42)
```



	왼편 그래프	오른편 그래프
C	작게	크게
도로폭(마진 위반 수)	크게	작게
분류	덜 정교하게	보다 정교하게

기타 선형 SVM 지원 모델 예제

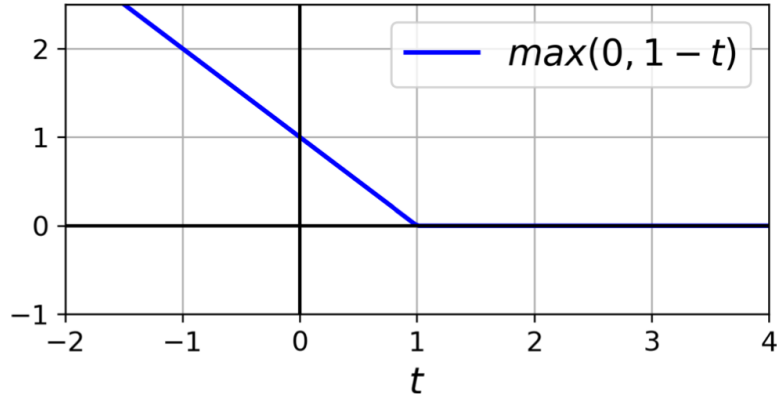
- SVC + 선형 커널

```
SVC(kernel="linear", C=1)
```

- SGDClassifier + hinge 손실함수 활용 + 규제: 규제 강도가 훈련 샘플 수(`m`)에 반비례.
    - 힌지 손실함수: 어긋난 예측 정도에 비례하여 손실값이 선형적으로 커짐. 로그 손실과 다름에 주의.

```
SGDClassifier(loss="hinge", alpha=1/(m*C))
```

- 경첩 손실(hinge loss) 함수



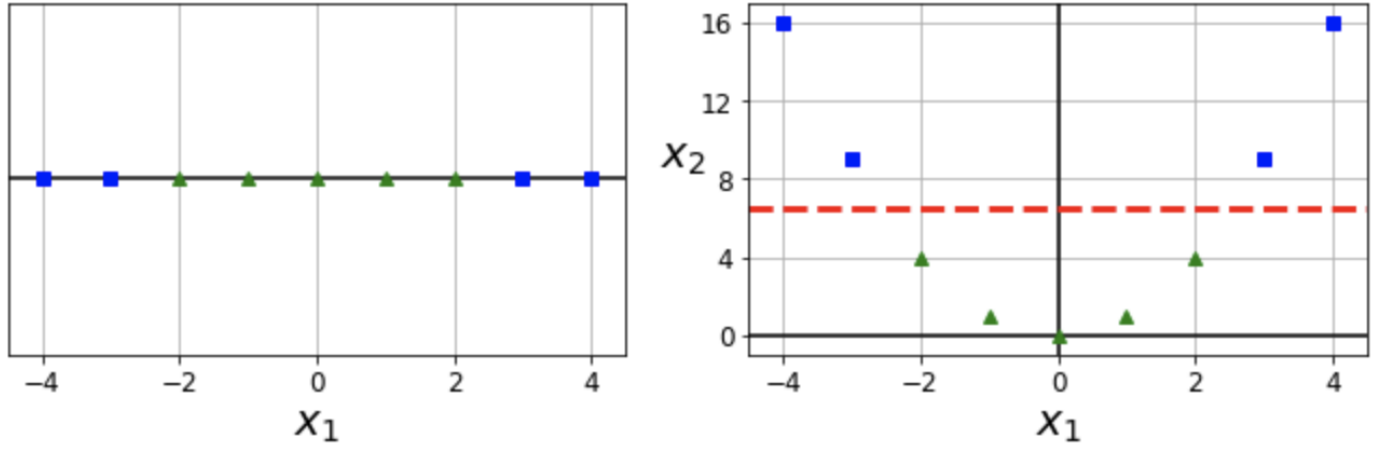
5.2 비선형 분류

- 방식 1: 특성 추가 + 선형 SVC
    - 다항 특성 활용: 다항 특성을 추가한 후 선형 SVC 적용
    - 유사도 특성 활용: 유사도 특성을 추가한 후 선형 SVC 적용
- 방식 2: `SVC` + 커널 트릭
    - 커널 트릭: 새로운 특성을 실제로 추가하지 않으면서 동일한 결과를 유도하는 방식
    - 예제 1: 다항 커널 (**주의**: 책에서는 다항식 커널로 불림)
    - 예제 2: 가우시안 RBF(방사 기저 함수) 커널

5.2.1 다항 커널

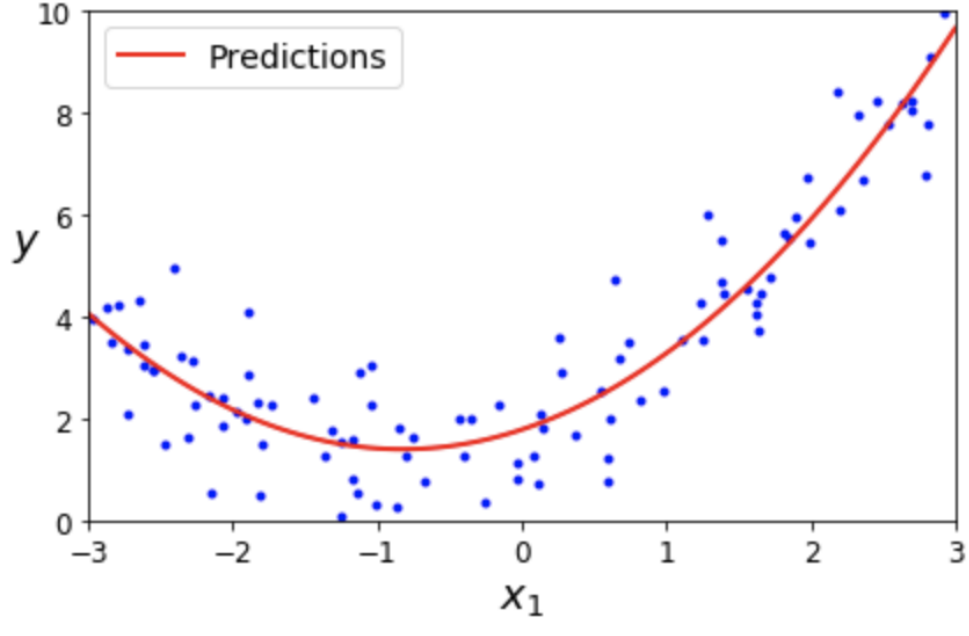
다항 특성 추가 + 선형 SVM

- 예제 1: 특성 x_1 하나만 갖는 모델에 새로운 특성 x_1^2을 추가한 후 선형 SVM 분류 적용

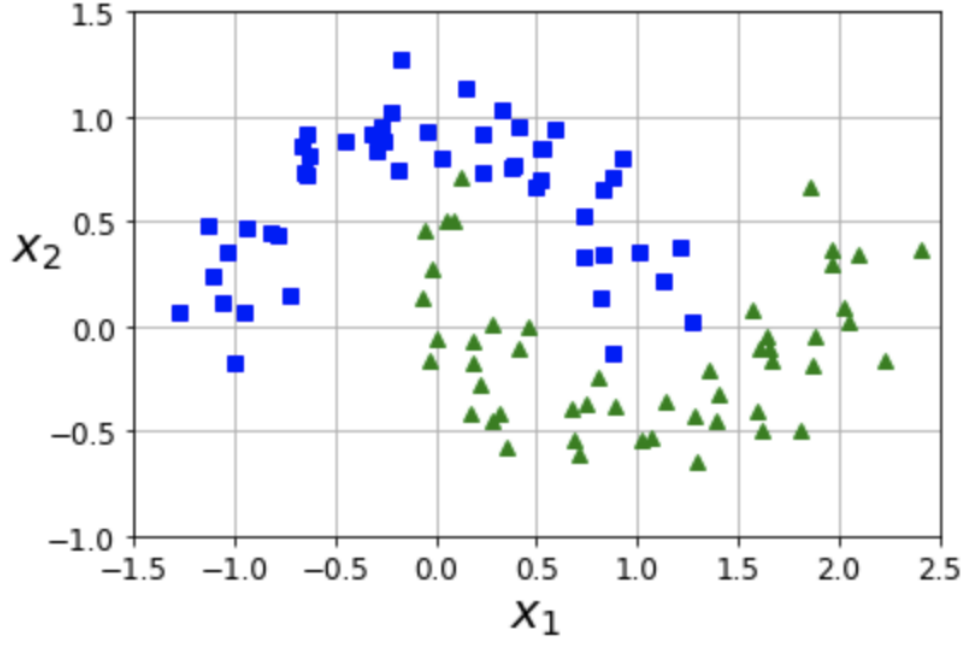


- 다항 특성 + 선형 회귀(4장): 특성 x_1 하나만 갖는 모델에 새로운 특성 x_1^2을 추가한 후 선형회귀 적용

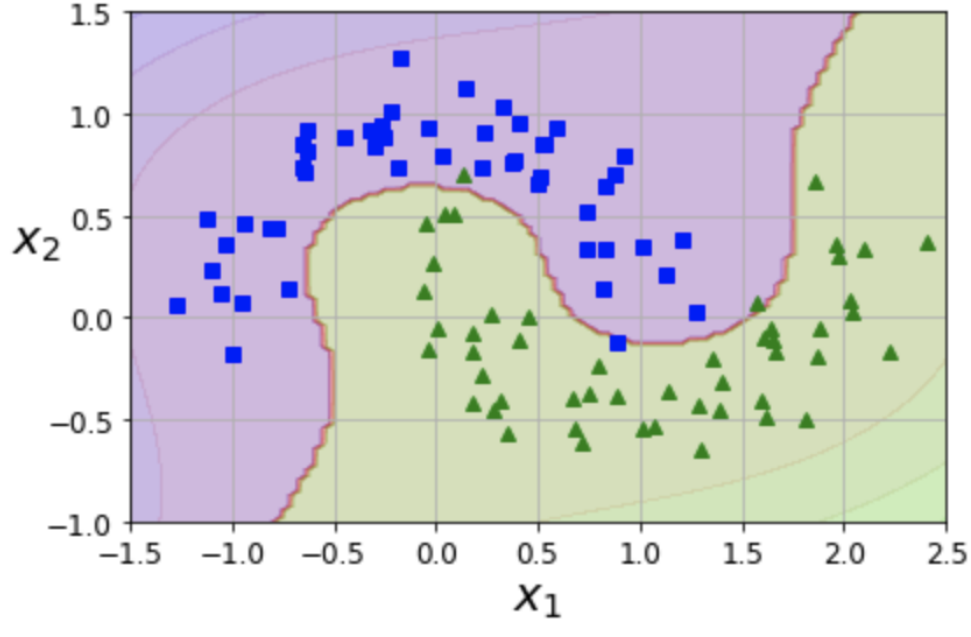
$$\hat{y} = \theta_0 + \theta_1 x_1 + \theta_2 x_1^2$$



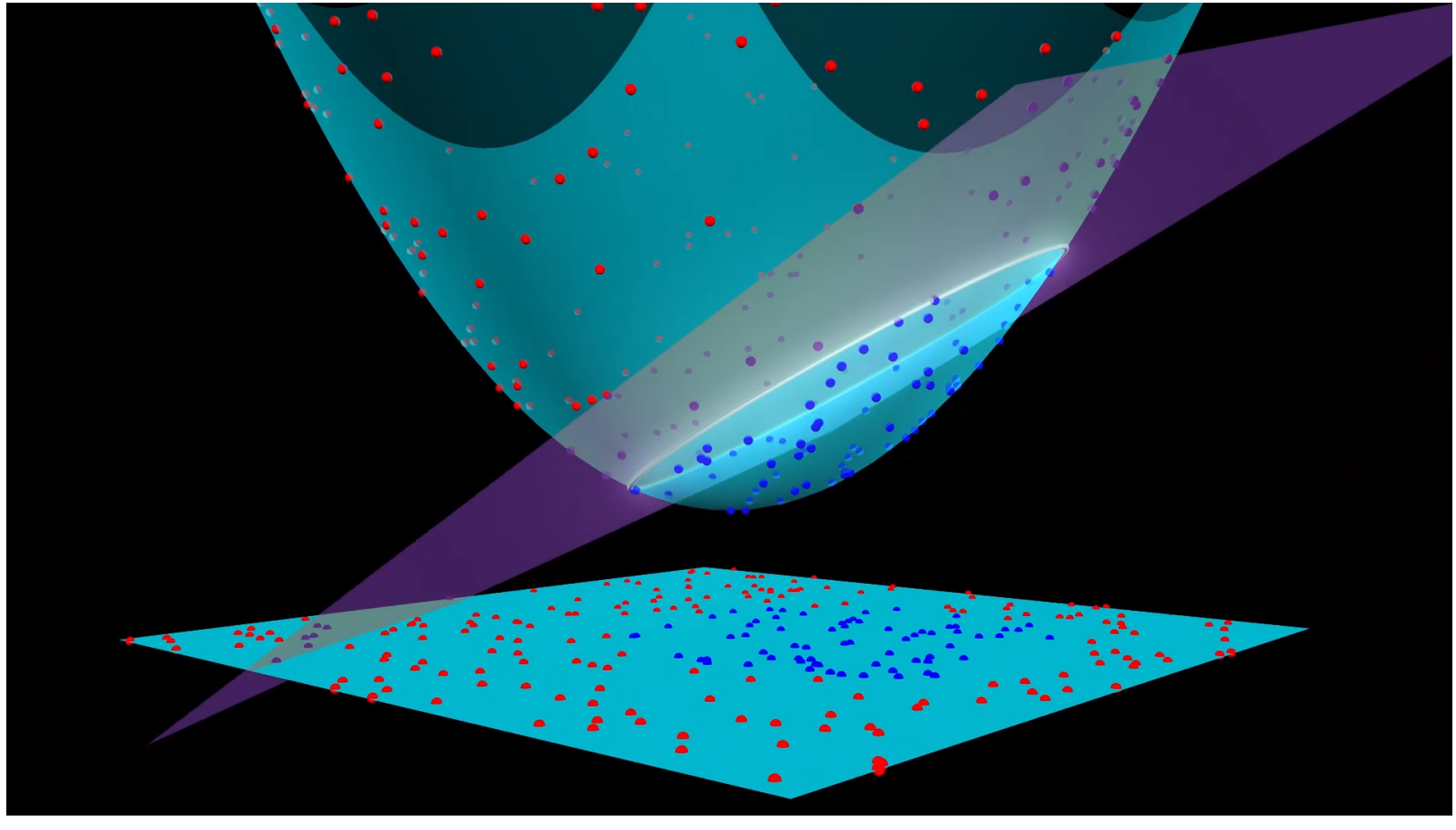
- 예제 2: moons 데이터셋. 마주보는 두 개의 반원 모양으로 두 개의 클래스로 구분되는 데이터



```
# 3차 항까지 추가
polynomial_svm_clf = Pipeline([
        ("poly_features", PolynomialFeatures(degree=3)),
        ("scaler", StandardScaler()),
        ("svm_clf", LinearSVC(C=10, loss="hinge", random_state=42))
    ])
```

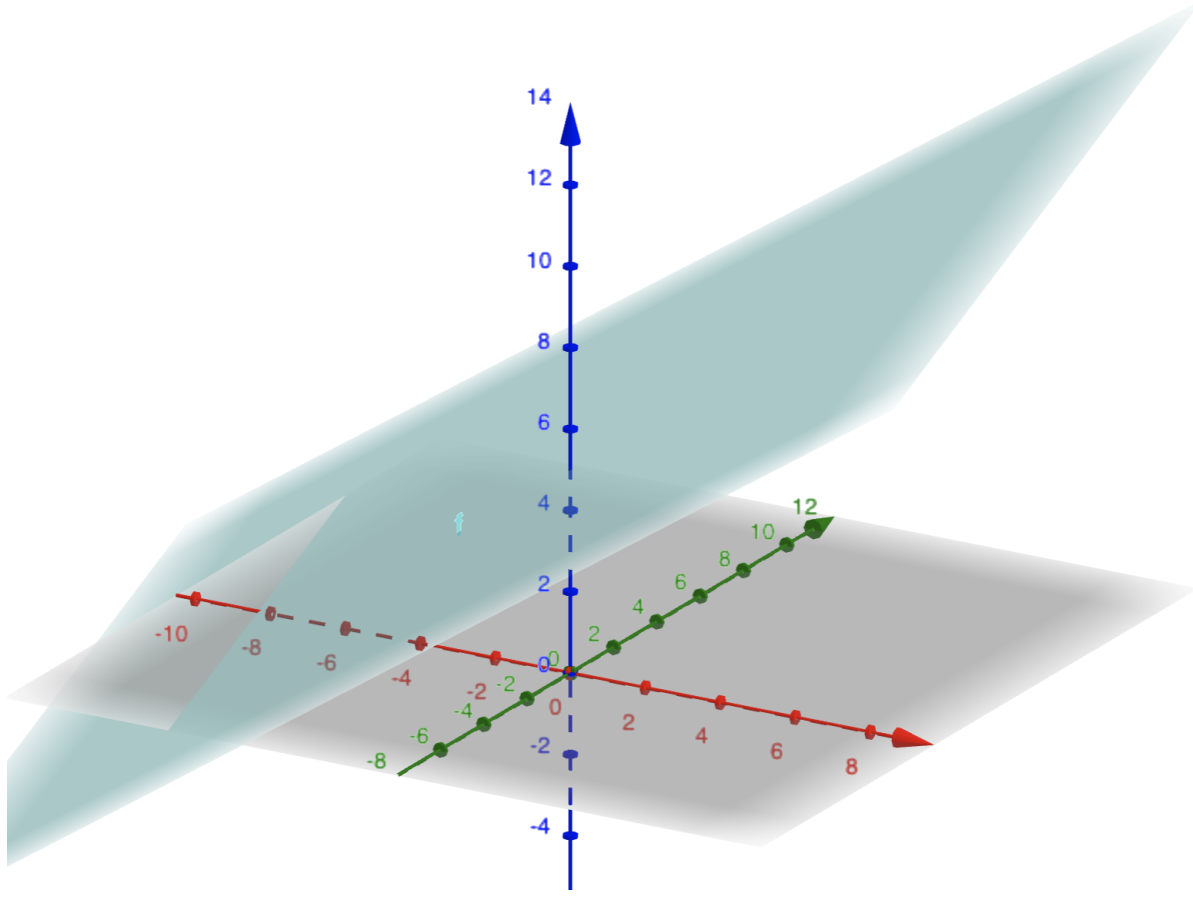


참고 영상: SVM + 다항 커널



- 3차원의 선형 방정식 그래프 예제:

$$z = \frac{3}{5}x + \frac{1}{5}y + 5 \quad \Longleftrightarrow \quad 3x + y - 5z + 25 = 0$$

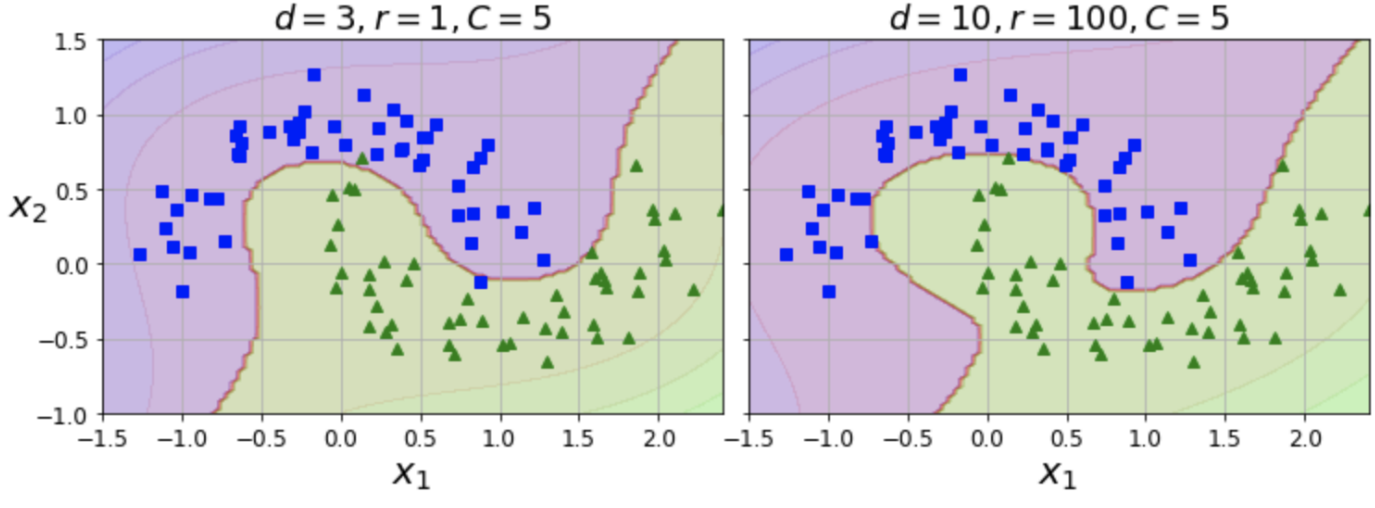


<그림 출처: 지오지브라(GeoGebra)>

SVC + 다항 커널

- 예제: moons 데이터셋

```
poly_kernel_svm_clf = Pipeline([
            ("scaler", StandardScaler()),
            ("svm_clf", SVC(kernel="poly", degree=3, coef0=1, C=5)) ])
```



	왼편 그래프	오른편 그래프
degree	3차 다항 커널	10차 다항 커널
coef0(r)	높은 차수 강조 조금	높은 차수 강조 많이

적절한 하이퍼파라미터 선택

- 모델이 과대적합이면 차수를 줄여야 함
- 적절한 하이퍼파라미터는 그리드 탐색 등을 이용하여 찾음
- 처음에는 그리드의 폭을 크게, 그 다음에는 좀 더 세밀하게 검색
- 하이퍼파라미터의 역할을 잘 알고 있어야 함

5.2.2 유사도 특성

유사도 함수

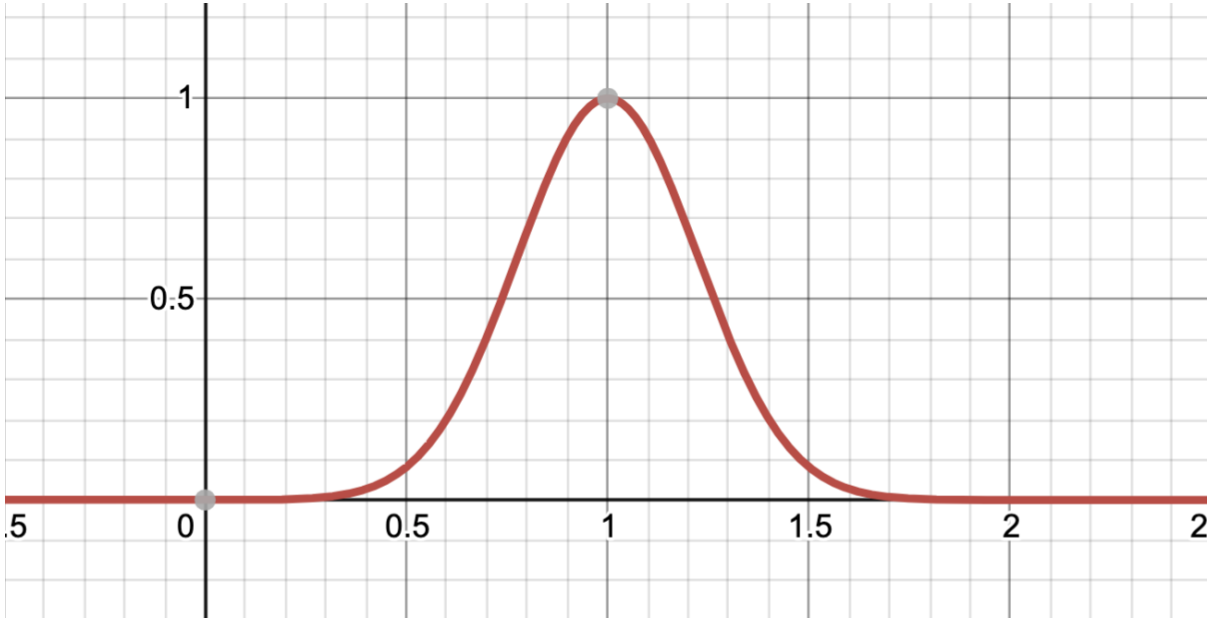
- 유사도 함수: 각 샘플에 대해 특정 **랜드마크**(landmark)와의 유사도를 측정하는 함수
- 예제: **가우시안 방사 기저 함수**(RBF, radial basis function)

$$\phi(\mathbf{x}, \ell) = \exp(-\gamma \left\|\mathbf{x} - \ell\right\|^2)$$

  - ℓ: 랜드마크
  - γ: 랜드마크에서 멀어질 수록 0에 수렴하는 속도를 조절함
    - γ 값이 클수록 가까운 샘플 선호하며 과대적합 위험 커짐.
    - 0: 랜드마크에서 아주 멀리 떨어진 경우
    - 1: 랜드마크와 같은 위치인 경우

- 예제

$$y = \phi(\mathbf{x}, 1) = \exp(-10 \left\| \mathbf{x} - 1 \right\|^2)$$

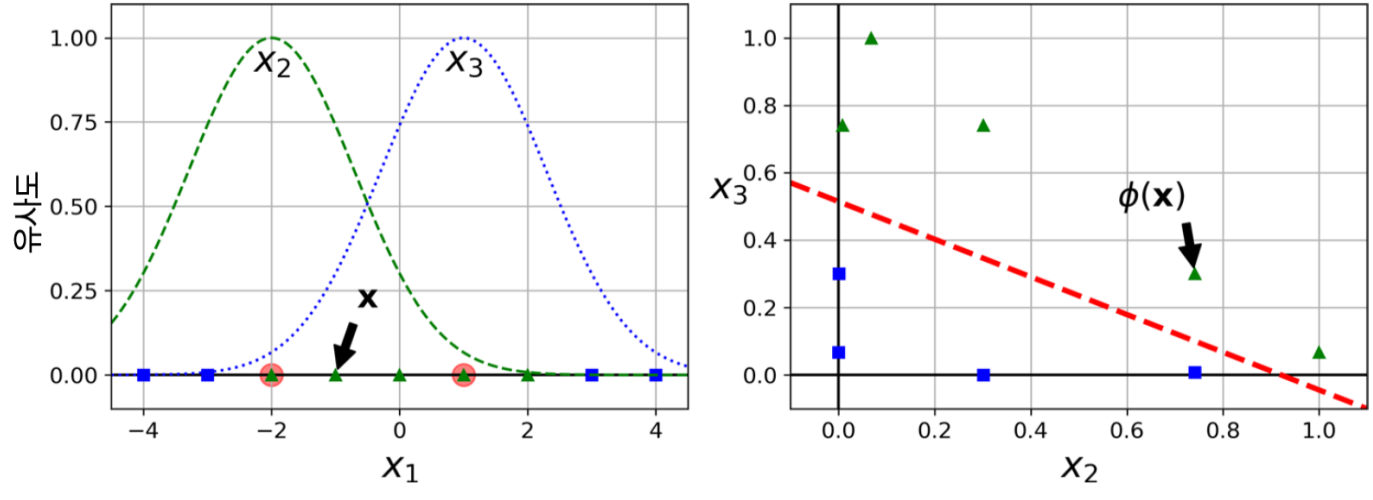


<그림 출처: 데스모스(desmos)>

유사도 특성 추가 + 선형 SVC

- 각 샘플을 랜드마크로 지정 후 **유사도 특성** 추가
- (n 개의 특성을 가진 m 개의 샘플) $\Rightarrow$ (m 개의 특성을 가진 m 개의 샘플)
- 장점: 차원이 커지면서 선형적으로 구분될 가능성이 높아짐.
- 단점: 훈련 세트가 매우 클 경우 동일한 크기의 아주 많은 특성이 생성됨.

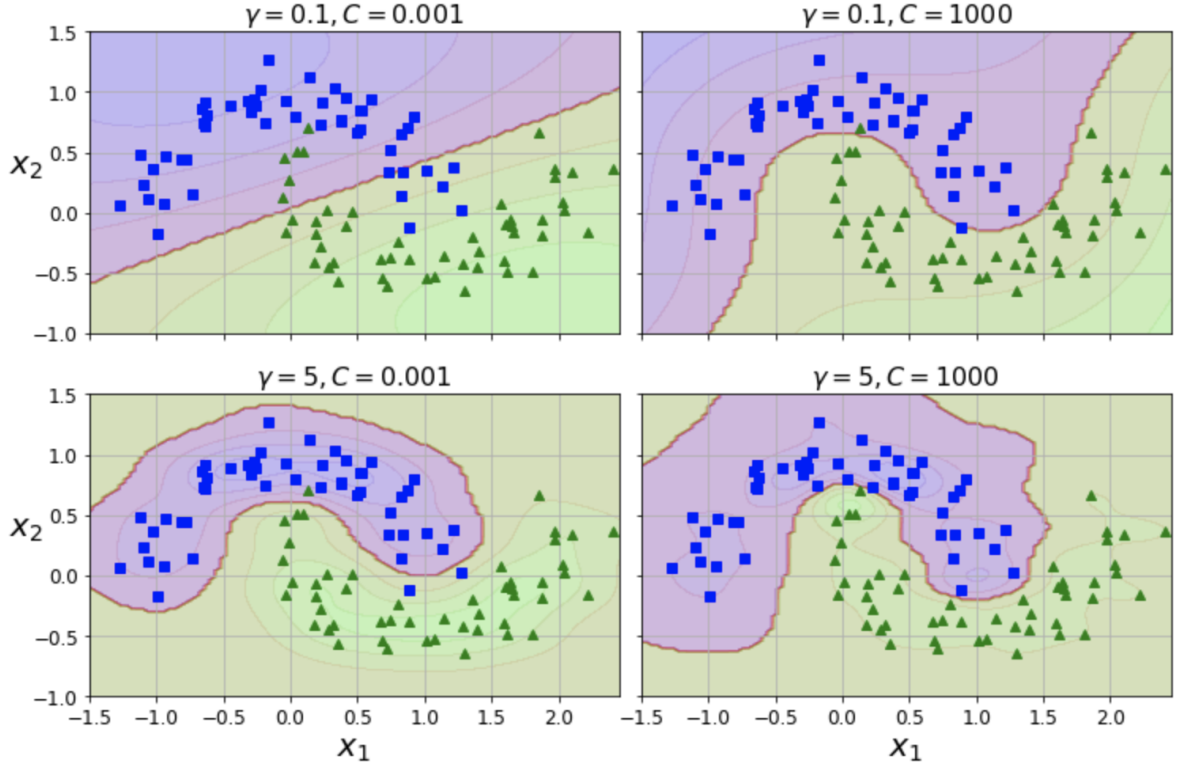
- 예제
  - 랜드마크: -2와 1
  - x_2과 x_3: 각각 -2와 1에 대한 가우시안 RBF 함수로 계산한 유사도
  - 아래 이미지: $\mathbf{x} = -1$



5.2.3 가우시안 RBF 커널

SVC + RBF 커널 예제: moons 데이터셋

```
rbf_kernel_svm_clf = Pipeline([
        ("scaler", StandardScaler()),
        ("svm_clf", SVC(kernel="rbf", gamma=0.1, C=0.001)) ])
```



	상단 그래프	하단 그래프
gamma	랜드마크에 조금 집중	랜드마크에 많이 집중

추천 커널

- `SVC`의 `kernel` 기본값은 `"rbf"` => 대부분의 경우 이 커널이 잘 맞음
- 선형 모델이 예상되는 경우 `SVC`의 `"linear"` 커널을 사용할 수 있음
- 훈련 세트가 크거나 특성이 아주 많을 경우 `LinearSVC`가 빠름
- 시간과 컴퓨팅 성능이 허락한다면 교차 검증, 그리드 탐색을 이용하여 적절한 커널을 찾아볼 수 있음
- 훈련 세트에 특화된 커널이 알려져 있다면 해당 커널을 사용

5.2.4 계산 복잡도

분류기	시간 복잡도(m 샘플 수, n 특성 수)	외부 메모리 학습	스케일 조정	커널 트릭	다중 클래스 분류
LinearSVC	$O(m \times n)$	미지원	필요	미지원	OvR 기본
SGDClassifier	$O(m \times n)$	지원	필요	미지원	지원
SVC	$O(m^2 \times n) \sim O(m^3 \times n)$	미지원	필요	지원	OvR 기본

5.3 SVM 회귀

- SVM 분류 목표: 마진 위반 발생 정도를 조절하면서 두 클래스 사이의 도로폭을 최대한 넓게 하기
- SVM 회귀 목표: 마진 위반 발생 정도를 조절하면서 도로폭을 최대한 넓혀서 도로 위에 가능한 많은 샘플 포함하기
- 회귀 모델의 마진 위반 사례: 도로 밖에 위치한 샘플

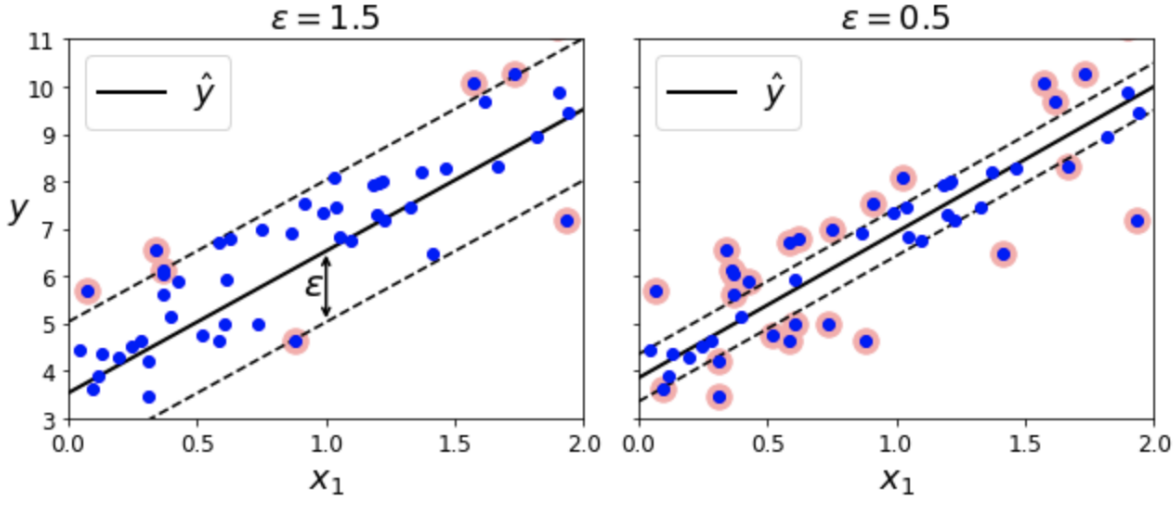
선형 SVM 회귀

- 선형 회귀 모델을 SVM을 이용하여 구현

예제: LinearSVR 활용

```
# LinearSVR 클래스 지정
from sklearn.svm import LinearSVR

svm_reg = LinearSVR(epsilon=e)
```



	왼편 그래프	오른편 그래프
epsilon=e:	e=1.5: 마진 크게	e=0.5 마진 작게

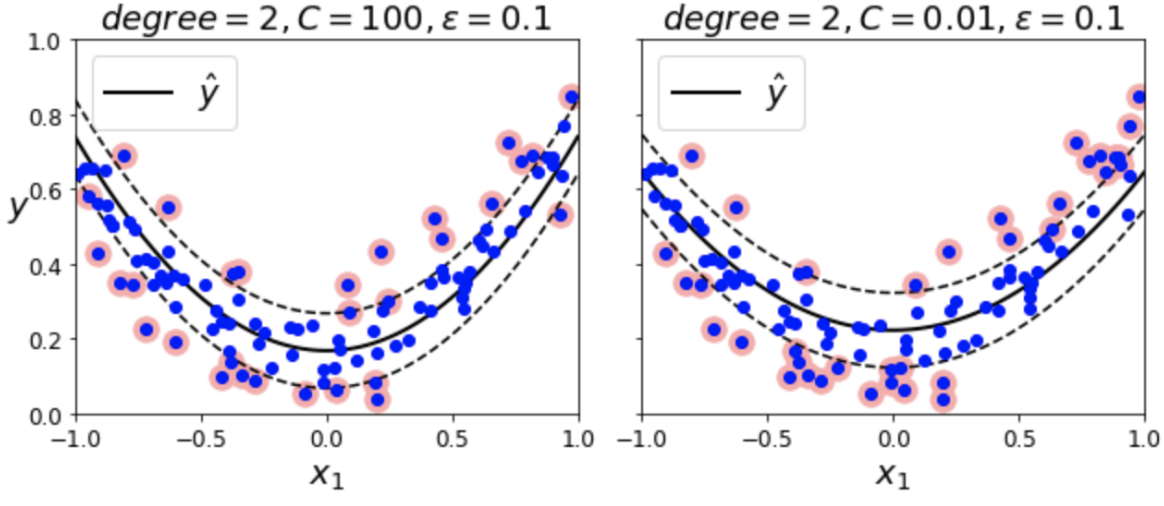
비선형 SVM 회귀

- 커널 트릭을 활용하여 비선형 회귀 모델 구현

예제: SVR + 다항 커널

```
# SVR + 다항 커널
from sklearn.svm import SVR

svm_poly_reg = SVR(kernel="poly", degree=d, C=C, epsilon=e, gamma="scale")
```



	왼편 그래프	오른편 그래프
degree=d:	d=2: 2차 다항 커널	d=2: 2차 다항 커널
epsilon=e:	e=0.1: 마진 작게	e=0.1 마진 작게
C=C:	C=100: 가중치 규제 거의 없음	C=0.01: 가중치 규제 많음
	샘플에 더 민감	샘플에 덜 민감
	도록폭을 보다 넓게	도로폭을 보다 좁게

회귀 모델 시간 복잡도

- `LinearSVR` : `LinearSVC` 의 회귀 버전
    - 시간 복잡도가 훈련 세트의 크기에 비례해서 선형적으로 증가
- `SVR` : `SVC` 의 회귀 버전
    - 훈련 세트가 커지면 매우 느려짐

5.4 SVM 이론

- (선형) SVM 작동 원리
    - 결정 함수와 예측
    - 목적 함수
    - 2차 계획법(QP, quadratic programming)
    - 쌍대 문제

- 커널 SVM 작동원리
    - 쌍대 문제를 해결할 때 커널 기법 활용 가능

- 온라인 SVM
    - 온라인 선형 SVM
    - 온라인 커널 SVM

(선형) SVM 작동 원리: 결정 함수와 예측

선형 SVM 분류기 모델의 결정 함수

$$\begin{aligned} h(\mathbf{x}) &= \mathbf{w}^T \mathbf{x} + b \\ &= w_1 x_1 + \cdots + w_n x_n + b \end{aligned}$$

선형 SVM 분류기 예측

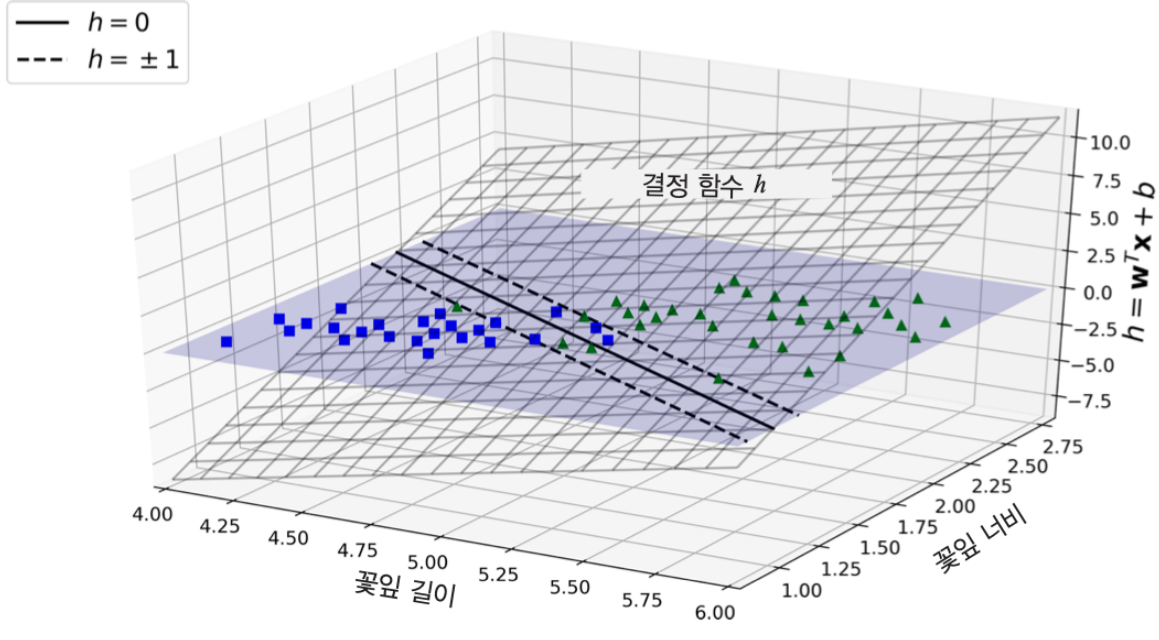
$$\hat{y} = \begin{cases} 0 & \text{if } h(\mathbf{x}) < 0 \\ 1 & \text{if } h(\mathbf{x}) \geq 0 \end{cases}$$

결정 경계

- 결정 함수의 값이 0인 점들의 집합

$$\{\mathbf{x} \mid h(\mathbf{x}) = 0\}$$

- 결정 경계 예제
    - 붓꽃 분류: 꽃잎 길이와 너비를 기준으로 Iris-Virginica(초록색 삼각형) 품종 여부 판단

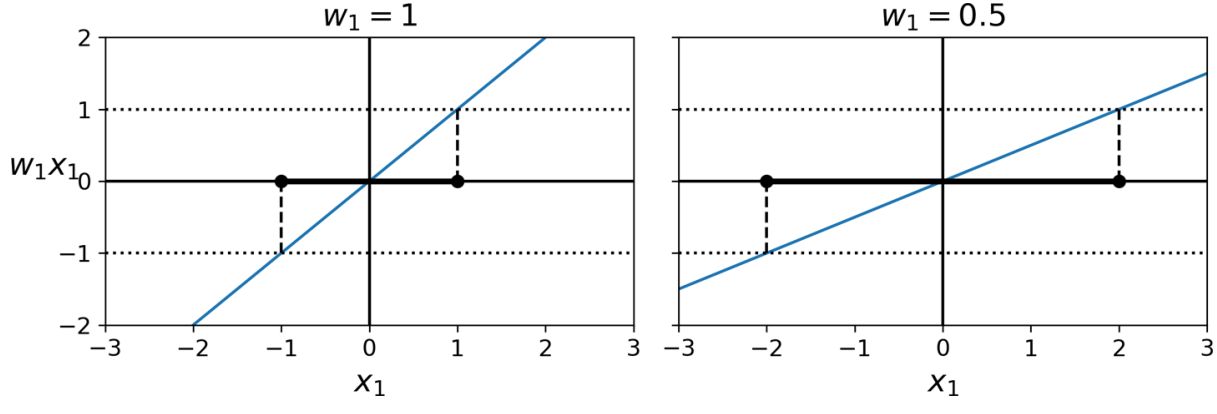


- 두 점선에 유의할 것
    - $h(\mathbf{x})$가 1 또는 -1인 샘플들의 집합
    - 마진과 밀접하게 관련됨.

(선형) SVM 작동 원리: 목적 함수

결정 함수의 기울기와 마진 폭

- 결정 함수의 기울기가 작아질 수록 마진 폭이 커짐. 아래 그림 참조
- 결정 함수의 기울기가 $\|\mathbf{w}\|$에 비례함.



- 마진을 크게 하기 위해 $\|\mathbf{w}\|$를 최소화 해야 함.
    - 하드 마진: 모든 양성(음성) 샘플에 대한 결정 함수의 값이 1(-1)보다 크다(작다)
    - 소프트 마진: 모든 샘플에 대한 결정 함수의 값이 지정된 값 이상 또는 이하 이어야 한다.

하드 마진 선형 SVM 분류기의 목적 함수

- 목적 함수:

$$\frac{1}{2}\mathbf{w}^T\mathbf{w}$$

- 아래 조건 하에서 목적 함수를 최소화 시키는 $\mathbf{w}$와 b를 구해야 함:

$$t^{(i)}(\mathbf{w}^T\mathbf{x}^{(i)} + b) \geq 1$$

- 단, 다음이 성립:
    - $x^{(i)}$: i 번째 샘플
    - $t^{(i)}$: 양성 샘플일 때 1, 음성 샘플일 때 -1

소프트 마진 선형 SVM 분류기의 목적 함수

- 목적 함수:

$$\frac{1}{2}\mathbf{w}^T\mathbf{w} + C\sum_{i=0}^{m-1}\zeta^{(i)}$$

- 아래 조건 하에서 목적 함수를 최소화 시키는 $\mathbf{w}$와 b를 구해야 함:

$$t^{(i)}(\mathbf{w}^T\mathbf{x}^{(i)} + b) \geq 1 - \zeta^{(i)}$$

- 단, 다음이 성립:
    - $x^{(i)}$: i 번째 샘플
    - $t^{(i)}$: 양성 샘플일 때 1, 음성 샘플일 때 -1
    - $\zeta^{(i)} \geq 0$: **슬랙 변수**. i 번째 샘플이 얼마나 마진을 위반할지 정함.
- C: 아래 두 목표 사이의 트레이드오프를 조절하는 하이퍼파라미터
    - 목표 1: 슬랙 변수의 값을 작게 만들기
    - 목표 2: 마진을 크게 하기 위해 $\frac{1}{2}\mathbf{w}^T\mathbf{w}$ 값을 가능하면 작게 만들기

(선형) SVM 작동 원리: 2차 계획법(QP)

- 하드(소프트) 마진 문제: 선형 제약조건이 있는 블록 2차 최적화 문제
- 2차 계획법(QP, quadratic programming) 문제로 알려짐.
- 해법에 대한 설명은 이 책의 수준을 벗어남.

(선형) SVM 작동 원리: 쌍대 문제

- 쌍대 문제(dual problem): 주어진 문제의 답과 동일한 답을 갖는 문제
- 하드(소프트) 마진과 관련된 2차 계획법 문제의 답을 보다 쉽게 해결할 수 있는 쌍대 문제를 이용하여 해결 가능

선형 SVM 목적 함수의 쌍대 문제

- 아랙 식을 최소화하는 α 찾기. 단, $\alpha^{(i)} > 0$:

  $$ \frac 1 2 \sum*{i=0}^{m-1} |sum*{j=0}^{m-1} \alpha^{(i)} \alpha^{(j)} t^{(i)} t^{(j)} \mathbf x^{{(i)}^T} \mathbf x^{(j)}

    - \sum_{i=0}^{m-1} \alpha^{(i)} $$

- 쌍대 문제의 답 $\hat{\alpha}$를 이용하여 $\hat{\mathbf{w}}$ 와 $\hat{b}$를 선형 SVM 모델의 파라미터로 활용
    - n_s: 서포트 벡터 수, 즉, $\hat{\alpha}^{(i)} > 0$ 인 샘플 수

$$\hat{\mathbf{w}} = \sum_{i=0}^{m-1} \hat{\alpha}^{(i)} t^{(i)} \mathbf{x}^{(i)}$$

$$\hat{b} = \frac{1}{n_s} \sum_{i=0,\ \hat{\alpha}^{(i)}>0}^{m-1} \left(t^{(i)} - \hat{\mathbf{w}}^T \mathbf{x}^{(i)} \right)$$

커널 SVM 작동 원리

쌍대 문제와 커널 SVM

- 커널 SVM이 작동 원리는 원래의 문제가 아닌 쌍대 문제 해결과 관련됨.

- 특히 아래 쌍대 목적 함수에서 사용된 $\mathbf{x}^{(i)^T}\mathbf{x}^{(j)}$에 주의해야 함.

$$\frac{1}{2}\sum_{i=0}^{m-1}\sum_{j=0}^{m-1}\alpha^{(i)}\alpha^{(j)}t^{(i)}t^{(j)}\boxed{\mathbf{x}^{(i)^T}\mathbf{x}^{(j)}}-\sum_{i=0}^{m-1}\alpha^{(i)}$$

예제: 2차 다항 커널 작동 아이디어

- 원래 아래 2차 다항식 함수를 적용한 후에 쌍대 목적 함수의 최적화 문제를 해결해야 함.

$$\phi(\mathbf{x}) = (x_1^2, \sqrt{2}x_1x_2, x_2^2)^T$$

- 원래 아래 식의 최적화 문제를 해결해야 함.

$$\frac{1}{2}\sum_{i=0}^{m-1}\sum_{j=0}^{m-1}\alpha^{(i)}\alpha^{(j)}t^{(i)}t^{(j)}\boxed{\phi(\mathbf{x}^{(i)})^T\phi(\mathbf{x}^{(j)})}-\sum_{i=0}^{m-1}\alpha^{(i)}$$

- 하지만 다음이 성립함

$$\phi(\mathbf{a})^T\phi(\mathbf{b}) = (\mathbf{a}^T\mathbf{b})^2$$

- 따라서 2차 다항식 함수 ϕ 전혀 적용할 필요 없이 아래 함수에 대한 최적화 문제를 해결하면 됨.

$$\frac{1}{2}\sum_{i=0}^{m-1}\sum_{j=0}^{m-1}\alpha^{(i)}\alpha^{(j)}t^{(i)}t^{(j)}\left(\mathbf{x}^{(i)^T}\mathbf{x}^{(j)}\right)^2 - \sum_{i=0}^{m-1}\alpha^{(i)}$$

- 커널 기법으로 구해진 쌍대문제의 해 $\hat{\alpha}$를 이용하여 예측값 $h(\phi(\mathbf{x}))$ 또한 $\phi(\mathbf{x})$ 없이 계산할 수 있음.

예제: 지원되는 커널

- 선형:

$$K(\mathbf{a}, \mathbf{b}) = \mathbf{a}^T \mathbf{b}$$

- 다항식:

$$K(\mathbf{a}, \mathbf{b}) = \left(\gamma \mathbf{a}^T \mathbf{b} + r\right)^d$$

- 가우시안 RBF:

$$K(\mathbf{a}, \mathbf{b}) = \exp\left(-\gamma \|\mathbf{a} - \mathbf{b}\|^2\right)$$

- 시그모이드:

$$K(\mathbf{a}, \mathbf{b}) = \tanh\left(\gamma \mathbf{a}^T \mathbf{b} + r\right)$$

온라인 SVM

- 온라인 학습: 새로운 샘플에 대해 점진적으로 학습하기

선형 온라인 SVM

- 특정 비용함수를 최소화하기 위한 경사하강법 사용
- 예제: 사이킷런의 SGDClassifier
    - `loss` 하이퍼파라미터를 `hinge` 로 설정하면 선형 SVM 모델 지정

비선형 온라인 SVM

- 온라인 커널 SVM 구현 가능.
- 하지만 신경망 알고리즘 사용 추천