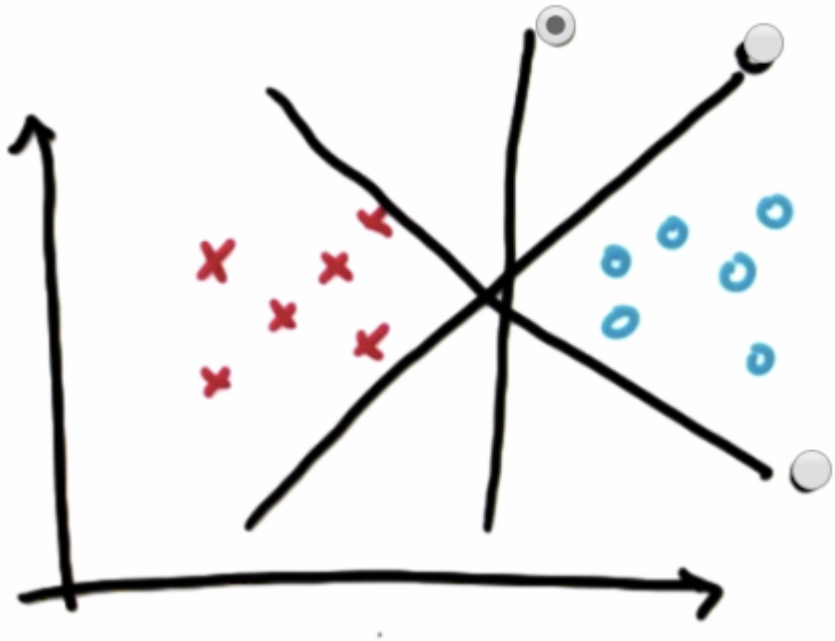
SVM(Support Vector Machine, 서포트 백터 머신)

1.설명

SVM은 주어진 데이터가 어느 카테고리에 속할지 판단하는 이진 선형 분류 모델입니다.



출처 : Udacity
위와 같이 빨간x 와 파랑 o를 구분하는 최적의 선을 구해 이를 통해 새로운 데이터의 카테고리를 구분하는 것입니다. 이때 3가지 구분선중 최적의 구분선을 구하는 몇가지 기준이 있다.

1-1. Margin 최대화

두 데이터를 구분하는 선을 Decision Boundary(구분선) 라고 한다. **구분선과 가장 가까운 점을 서포트 벡터(Support Vector)** 라고 하는데, **Margin은 구분선과 서포트 벡터와의 거리**를 의미한다.즉 Margin은 선과 가장 가까운 양 옆의 데이터와의 거리이다. 위의 3개 구분선 중 가운데 선의 Margin이 제일 크다.

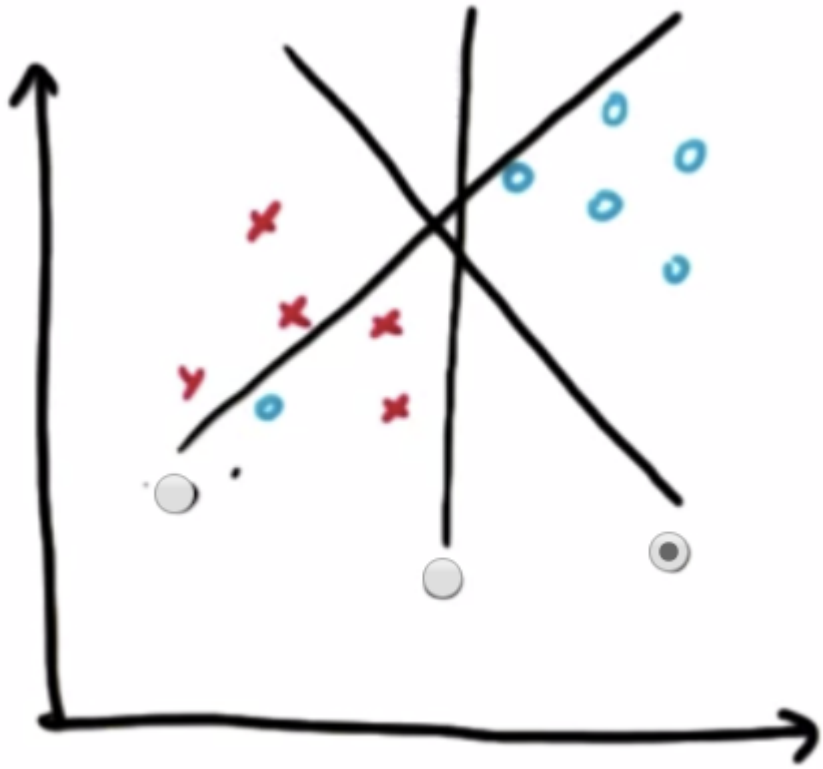
1-2. Robustness

Robustness는 건장한, 튼튼한 정도의 뜻으로, 아웃라이어(outlier)의 영향을 받지 않는다는 뜻이다.

> 1, 2, 3, 4, 5 평균 중앙값 모두 3이다.
> 1, 2, 3, 4, 100 평균 22, 중앙값 3이다.
> 평균은 100이라는 아웃라이어의 영향을 받아서 크게 바뀌었고, 중앙값은 영향을 받지 않았다.
> 이경우 **평균은 robust하지 않고 중앙값은 robust하다고 한다.**

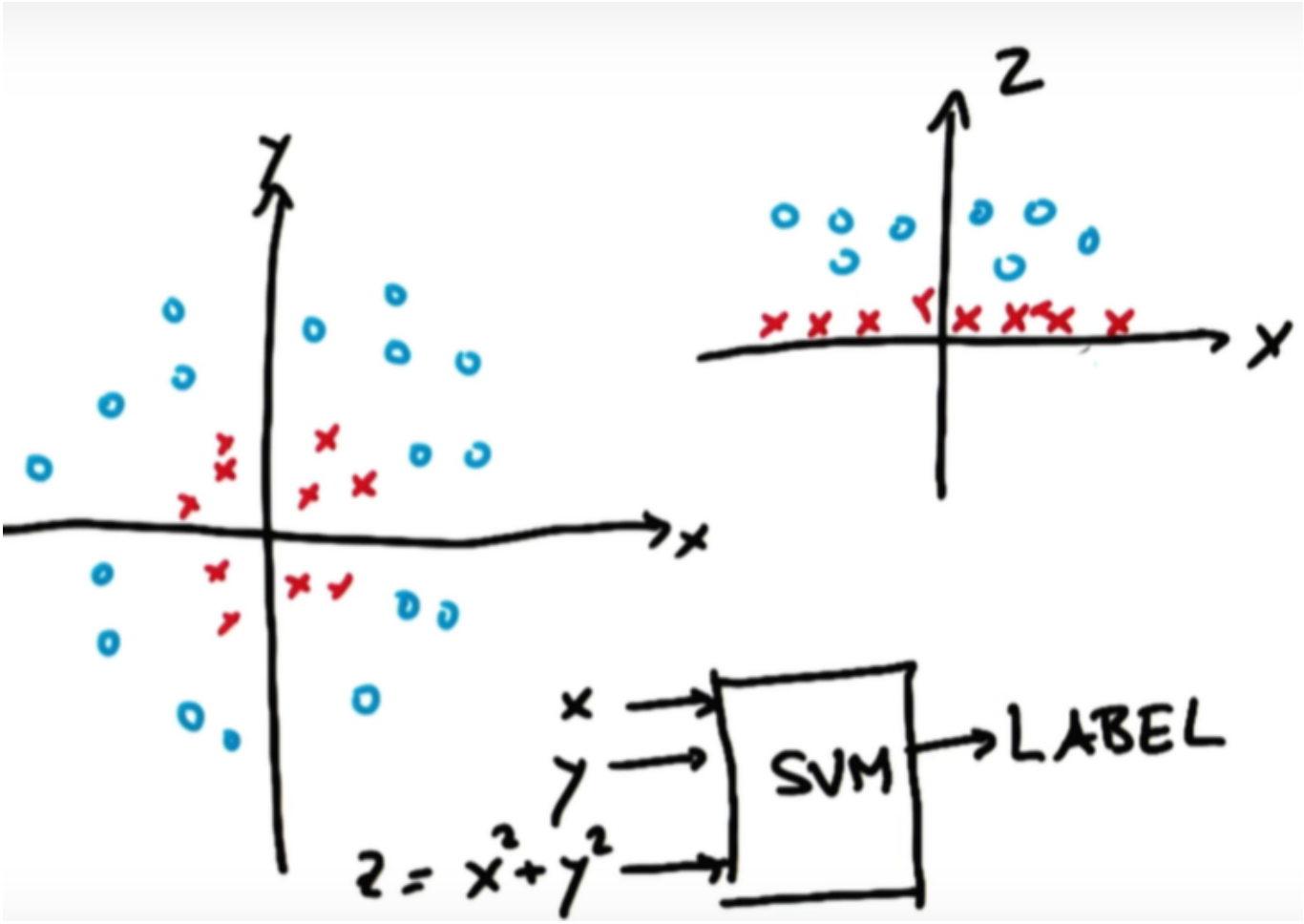
위의 3개의 구분선에서 가운데 선을 제외한 나머지 구분선은 비교적 노이즈로 인한 아웃라이어 발생시 제대로 구분하지 못하므로 로버스트하지 않는다. **Margin을 최대화 하면 Robustness도 최대화 된다.**

1-2. Outlier 처리

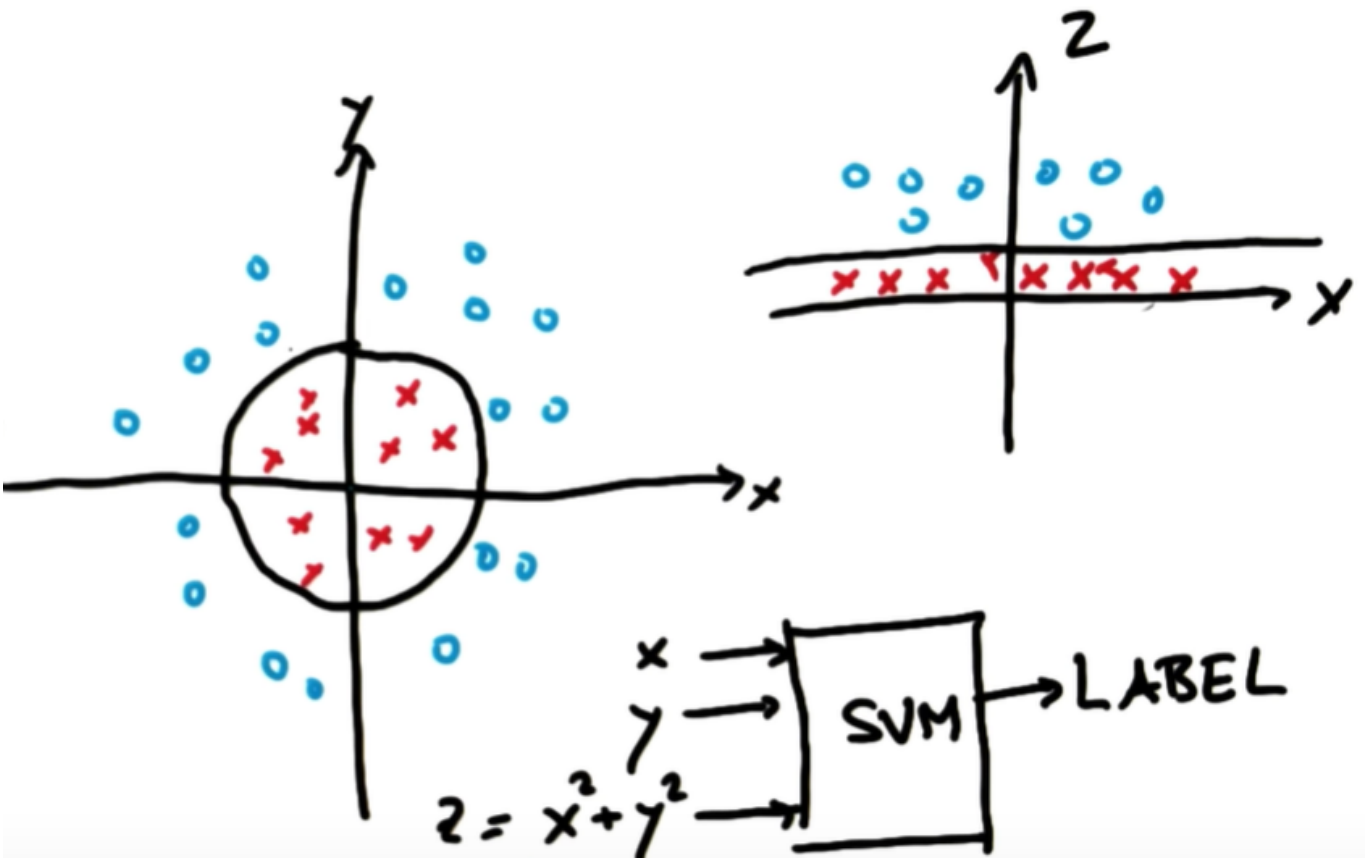


출처 : Udacity
위와 같은 outlier가 있는 데이터가 있을경우 두 데이터를 정확히 구분하는 직선이 없기 때문에 SVM이 어느정도 outlier를 무시하고 최적의 구분선을 찾는다. 위에서는 파란 점을 outlier로 취급해서 무시하고 Margin을 최대화하는 구분선을 찾는다.

1-3. 커널트릭(Kernel Trick)



위 그림처럼 구분을 위한 선형 구분선을 그릴 수 없을 경우가 발생할 수 있다. 위의 그림의 경우 원의 형태로 구분이 가능하므로 새로운 축 z를 x와 y로 정의하고 새로운 차원으로 바꾸어 구분선을 그릴수 있다.



그 결과 위와 같은 새로운 평면에서 새로운 선형 분류가 가능해진다. x,y로만 이뤄진 평면에서 x,y,z 평면으로 차

원 확대를 통해 구분선을 그려서 선형 분류가 가능하도록 할수 있다. **저차원 공간(low dimensional space)을 고차원 공간(high dimensional space)로 매핑해주는 작업을 커널 트릭(Kernal Trick)** 이라고 한다. 저차원 공간에서 비선형한 분포의 데이터를 커널 트릭으로 고차원 공간으로 매핑하여 선형 구분선을 구하고, 이 선을 다시 저차원 공간으로 매핑하면 비선형 구분선을 구할수 있다. **커널 트릭을 활용하여 고차원 공간의 선형한 해를 선행하여 구한후 비선형 공간에서 비선형한 해를 구하는것이 머신러닝의 중요기법 중 하나이다.**

---

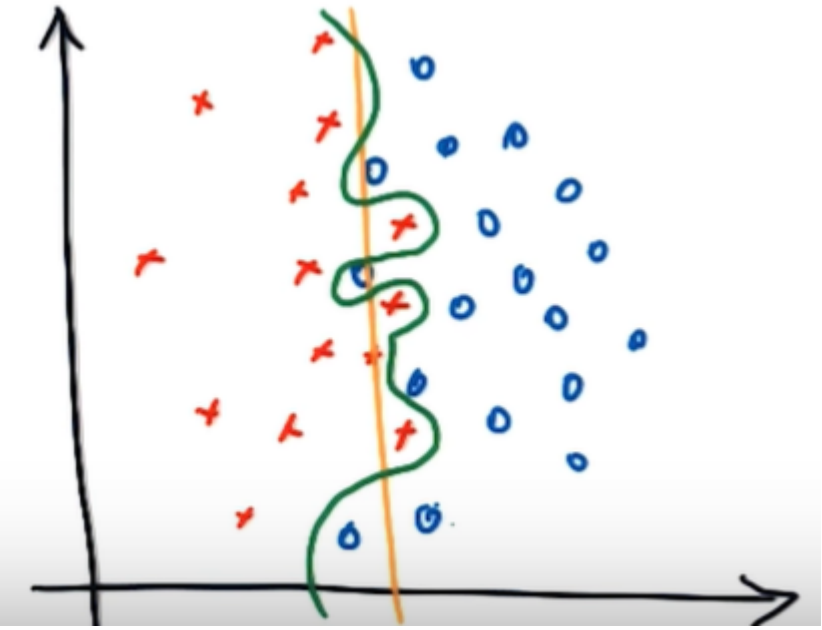
2 특징

SVM의 주요 파라미터인 Kernal, C, Gamma가 있다.

2-1. Kernal

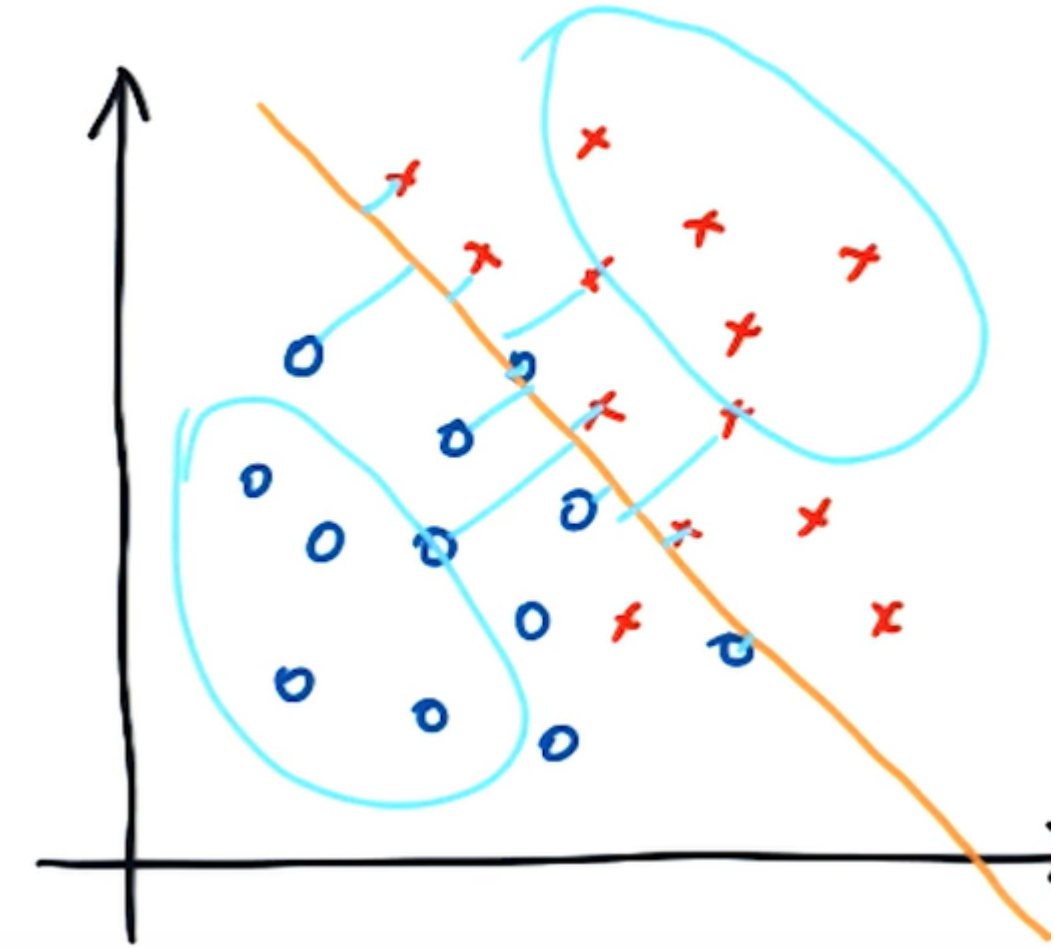
linear, polynomial(다항), sigmoid(아크탄젠트 같은 s형 곡선),rbf(Radial Basis Function 방사형)등 kernal이 선택 가능하다. Decision Boundary(구분선)의 묘양을 선택하는 것

2-2. C(Controls tradeoff between smooth decision boundary and classfying training points correctly)

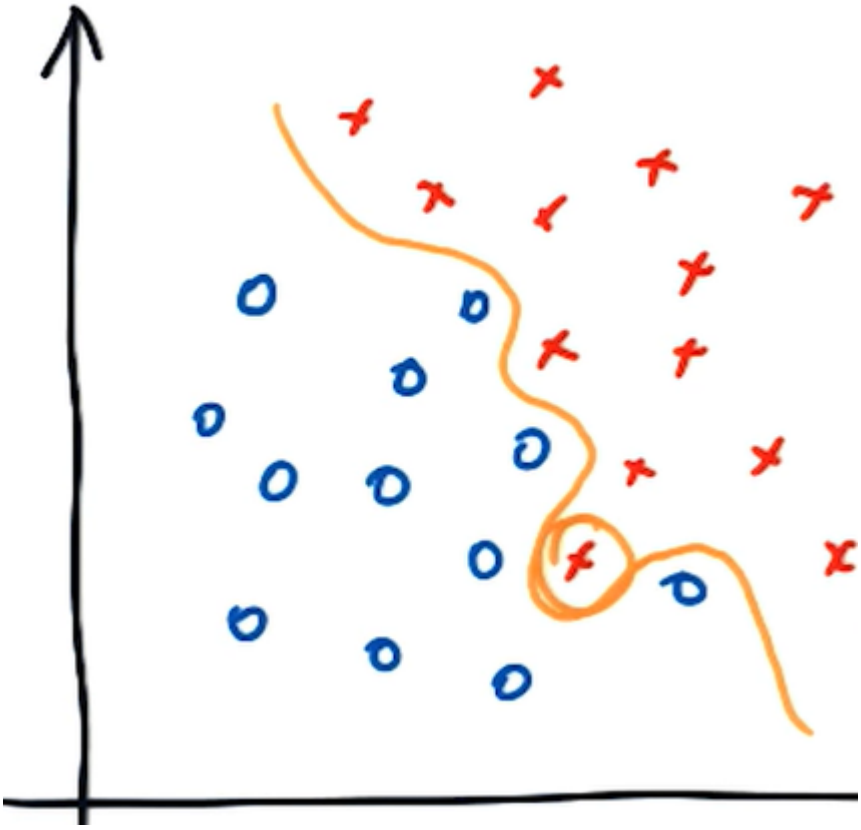


출처 : Udacity
초록색 구분선은 C가 큰 decision boundary이고, 주황색은 C값이 작은 decision boundary이다.C가 크면 더 정확히 구분해주지만 Margin값이 작아진다. **C가 크면 decision boundary가 더 굴곡지고 작으면 직선에 가까워진다.**

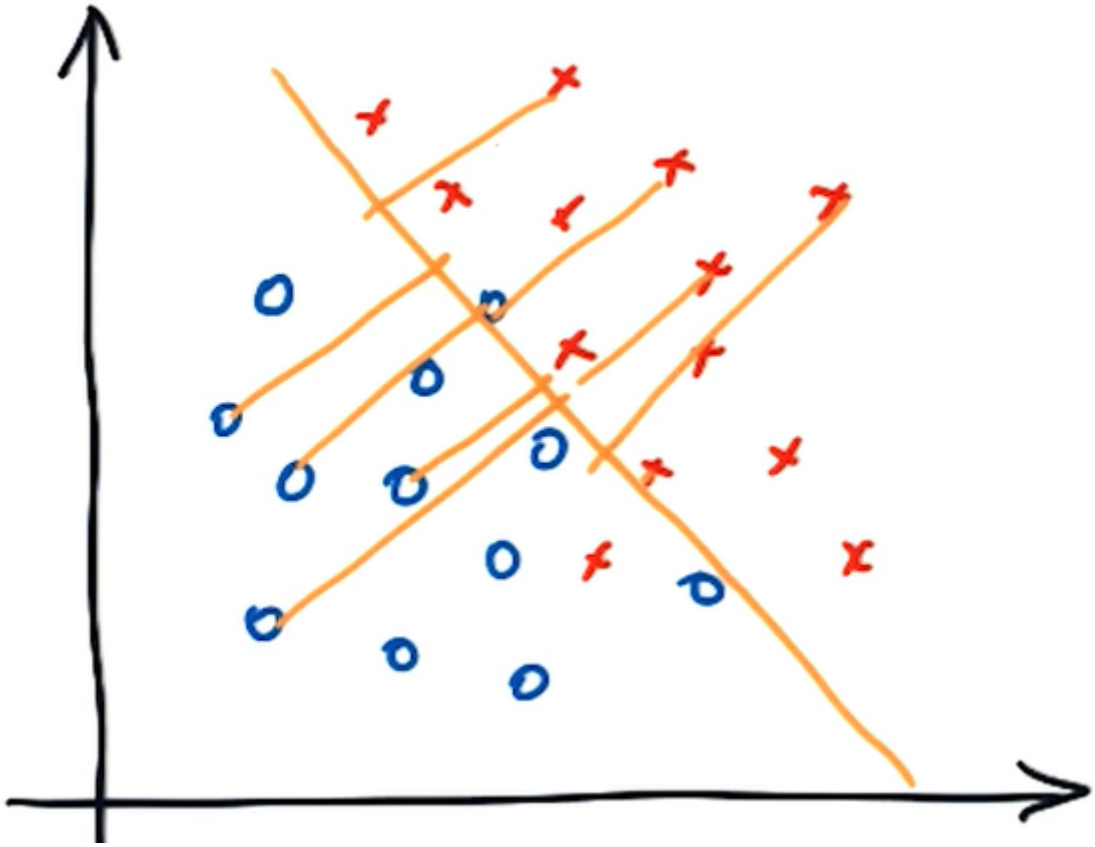
2-3 Gamma



출처 : Udacity
위와 같은 분포에서 Gamma값이 크면 reach가 좁기때문에 원안의 포인트들이 Decision Boundary에 영향을 주지 않고, Decision Boundary에 가까운 포인트들만 굴곡에 영향을 준다. 선 가까이 있는 포인트 하나하나의 영향이 커지고 멀리 있는 포인트들의 영향이 줄어 들어 아래와 같이 굴곡이지게 된다.



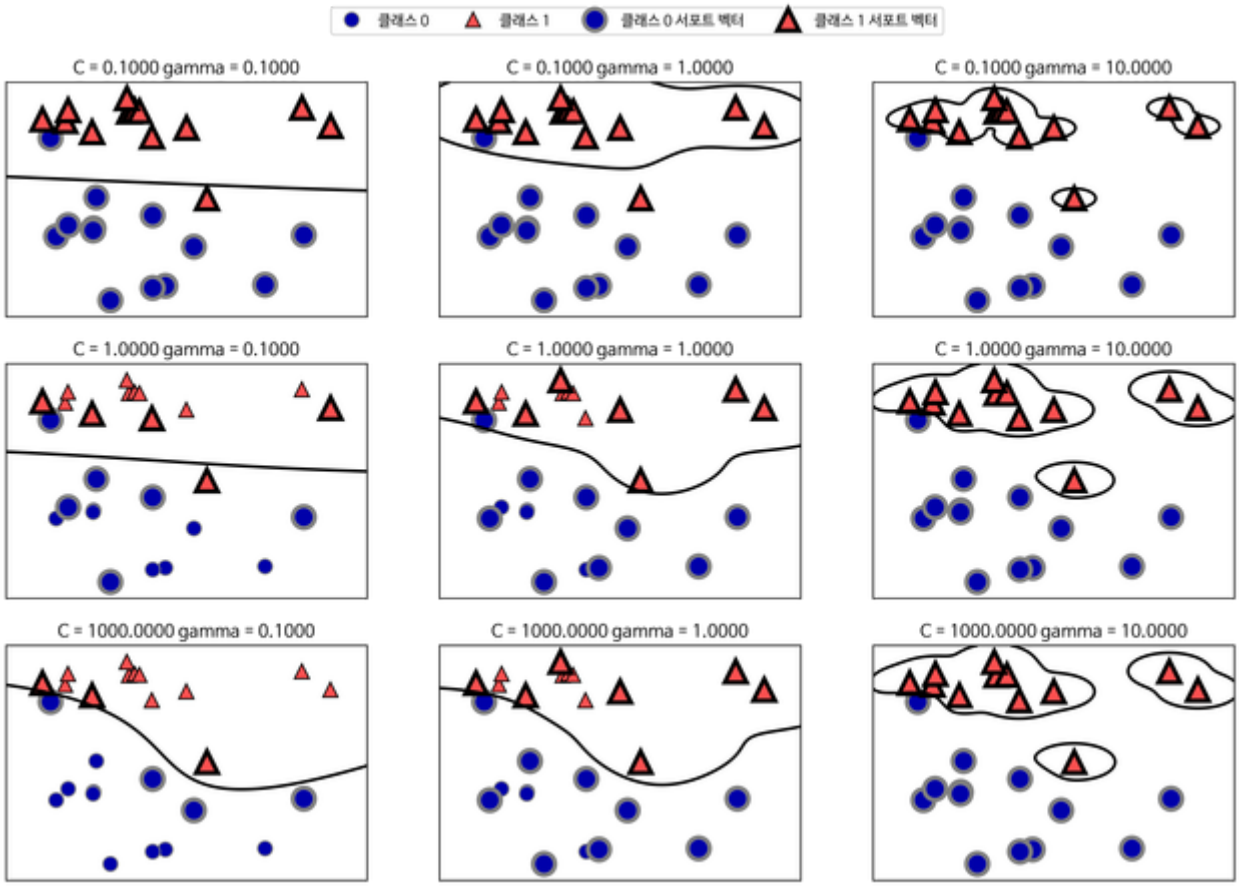
출처 : Udacity
반면 Gamma값이 작으면 reach값이 멀기 때문에 대부분의 포인트의 영향을 받아 가까이 있는 포인트 하나하나의 영향력이 줄어든다. 그러므로 아래와 같이 선이 포인트 하나 때문에 구부러지지 않는다.



출처 : Udacity

Gamma가 크면 decision boundary가 더 굴곡지고 작으면 직선에 가까워진다.

C와 Gamma에 따른 Decision Boundary



C는 두 데이터를 정확히 구분하는것이 목적이고, Gamma는 개별데이터 마다 Decision Boundary를 만드는것이 목적이다. Gamma는 커질 경우 Boundary가 여러개 생성될수 있다.

2-4 Overfitting(과적합)

Overfitting은 훈련데이터를 지나치게 학습하는 것으로 훈련데이터에 대해서는 100%의 성능을 내지만, 테스트 데이터에서는 성능이 떨어질 수 있다.

출처 : Udacity

훈련데이터를 과하게 학습하면 괴상한 구분선이 만들어 지고, 지나치게 훈련데이터에 적합한 구분선이 생성되어 새로운 데이터 구분시 성능이 떨어진다.
Kernal, C, Gamma 모두 과적합에 영향을 줄 수 있다. C, Gamma가 지나치게 높으면 과적합될 수 있다.
따라서 성능을 높히는 것과 과적합 사이 균형을 잘 지켜야 한다.

---

3 적합 데이터

SVM은 Training Time이 길기 때문에 사이즈가 큰 데이터 셋에는 적합하지 않다. 또한 추가적으로 노이즈가 많은 데이터의 경우 오버피팅 될 수 있으므로 적합하지 않다. 노이즈가 많을 경우 차라리 Naive Bayes가 적합할 수 있다.

2-4. 장단점

- 장점 : 범주나 수치 예측 문제에 사용가능, 오류 데이터 대한 영향이 없다. 과적합 되는 경우가 적고 사용하기 쉽다.
- 단점 : 여러 개의 조합 테스트가 필요하고, 최적의 모델을 찾기위해 커널과 모델에서 다양한 테스트 피룡, 데이터 셋이 많을 경우 학습속도가 느리다.해석이 복잡하고 어렵다.

적합한 사례

- 텍스트 하이퍼 텍스트 분류, 이미지 분류 필기 인식, 필기체 인식, 유전자 데이터 분류