

Derin Öğrenme

Derin Öğrenme Temelleri

Emir Öztürk

Yapay Sinir Ağları

Old but gold

- Deep learning'den önce daha kural tabanlı algoritmalar
 - Restricted Boltzman Machines
 - Deep belief Networks
 - Hopfield Networks
 - Self Organizing Maps

Derin Öğrenme Uygulama Alanları

Because... why not?

- Bilgisayarlı Görü
- Doğal Dil İşleme
- Ses İşleme
- Kontrol sistemleri ve otonom sistemler

Derin öğrenmenin yaygınlaşması

The problem is not you but me

- Veri
- Hesaplama gücü
- MODÜLARİTE
- Learning Curve düşük
 - Hızlı model üretme
 - YANLIŞ model üretme
- WEKA

Derin öğrenme başarısı

Success is an illusion

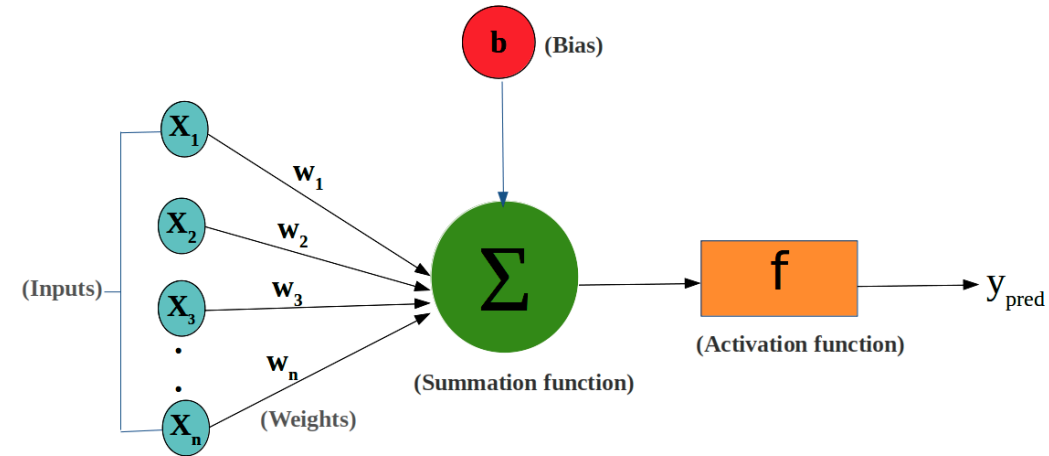
- Daha çok veri \neq Daha çok başarı
- Daha büyük model \neq Daha çok başarı
- Daha ham veri \neq Daha çok başarı

Yapay Sinir Ağları

Nothing like the real thing

N

- Sinir hücresi
- Aktivasyon
- Belirli durumların ağırlığı
- Hata değerinin eklenmesi



Yapay Sinir Ağları

Life is hard, calculating is harder

N

Vize	Final	Gerçek Değer	Geçiyor mu
50	50	50	1
30	70	58	1
70	30	42	0
37	43	41,2	0
85	76	78,7	1
100	28	49,6	1
42	100	82,6	1

Wv	Wf	B
0,5	0,5	5
0,6	0,4	5
0,8	0,2	5
0,3	0,7	5

Yapay Sinir Ağları

N

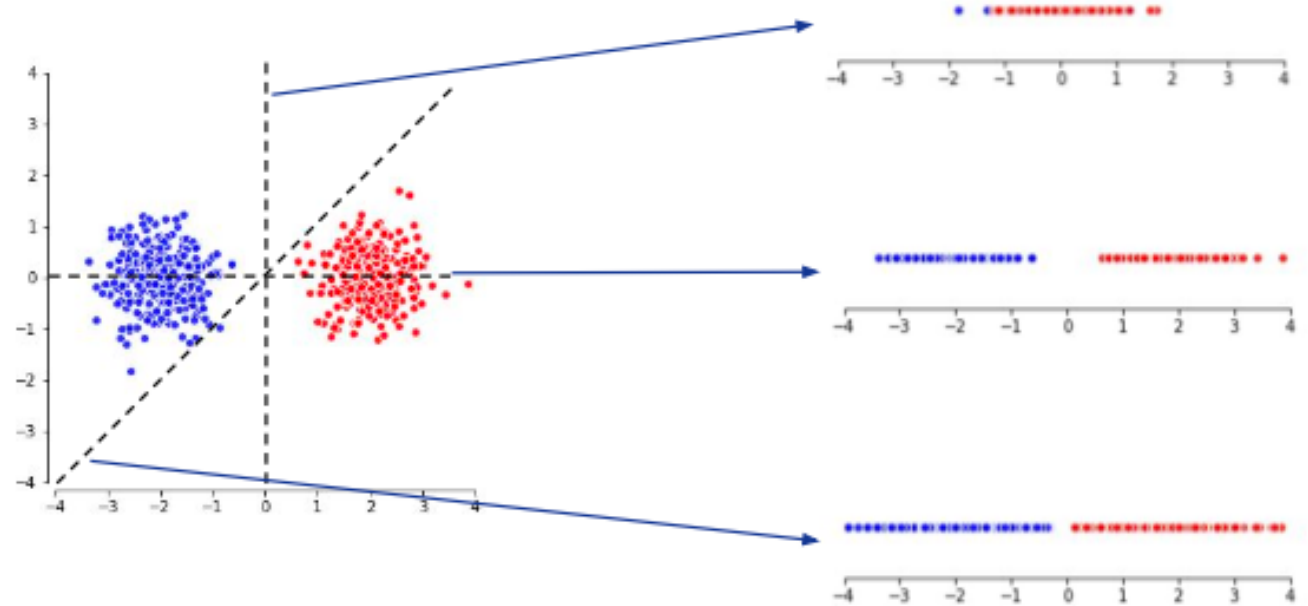
Life is hard, calculating is harder

Vize	Final	Gerçek Değer	Geçiyor mu	W1	W2	W3	W4
50	50	50	1	1	1	1	1
30	70	58	1	1	0	0	1
70	30	42	0	1	1	1	0
37	43	41,2	0	0	0	0	0
10	100	73	1	1	0	0	1
100	28	49,6	1	1	1	1	1
42	100	82,6	1	1	1	1	1
			1	0,86	0,57	0,57	1

Bir nöron ve çıktısı

Yes! I built an AI system

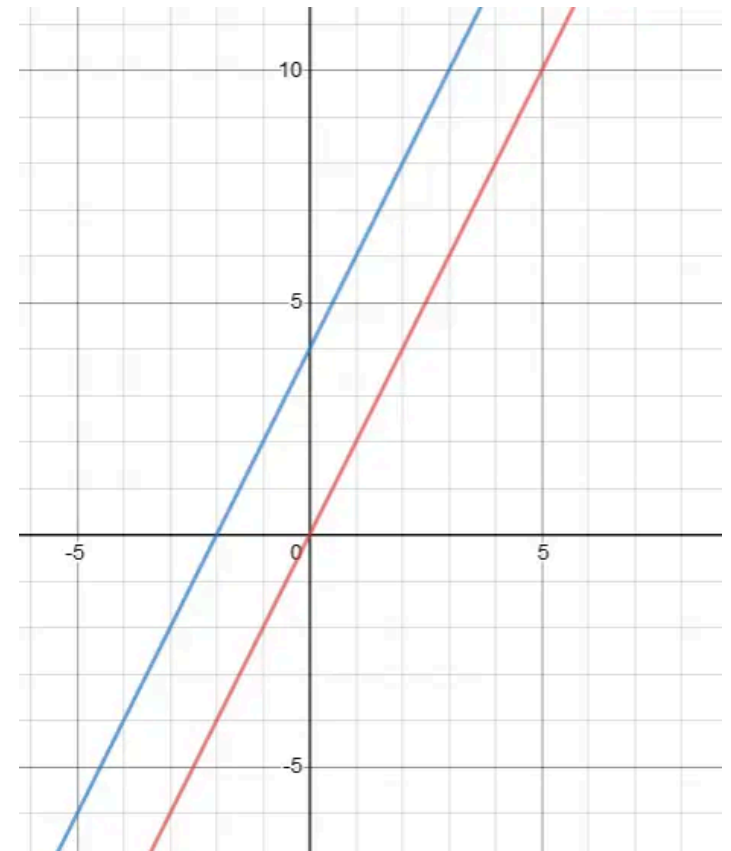
- $Ax+b$
- Lineer
- Regresyon



Linear Layer

Don't trust anybody

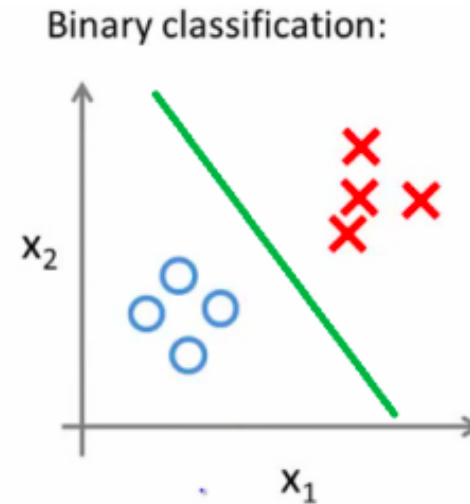
- Aslında lineer değil (ax)
 - Bias: Translation (b)
- Affine ($ax+b$)
- Her lineer fonksiyon affine
- Her affine fonksiyon lineer değil



Karmaşık sorunların çözümü

Life is complicated

- Lineer çok basit
- Kompleks yapılar modüler değil
- Stack edilmiş lineer yapılar
- Her düğümün birbiri ile kesişimi
- Non-lineerite için bir yapının eklenmesi

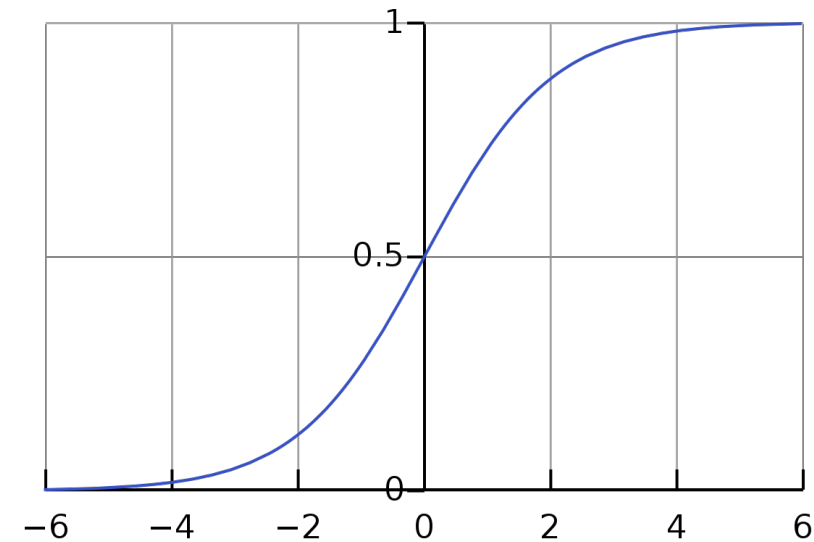


Sigmoid

A

We need to go deeper

- Non-linearitenin eldesi için lineer olmayan katman
- Çok yüksek ve düşük değerler için eğimin kaybolması
- Vanishing gradients
- Bilgi kaybı
- Tanh
- ReLU
- Leaky ReLU



Ağırlıkların düzenlenmesi - Öğrenme

A machine that learns



- Hesap sonucu kaybın bulunması
- Kayıp için bir fonksiyon
- İkili sınıflandırma için Cross-Entropy

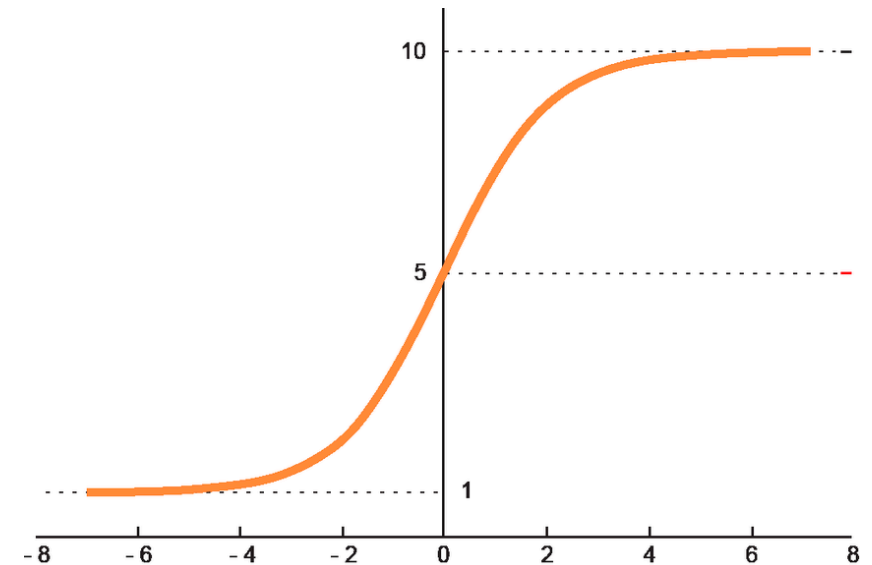
$$H(P^* | P) = - \sum_i \underbrace{P^*(i)}_{\text{TRUE CLASS DISTIRBUTION}} \log \underbrace{P(i)}_{\text{PREDICTED CLASS DISTIRBUTION}}$$

Ağırlıkların düzenlenmesi - Öğrenme

L

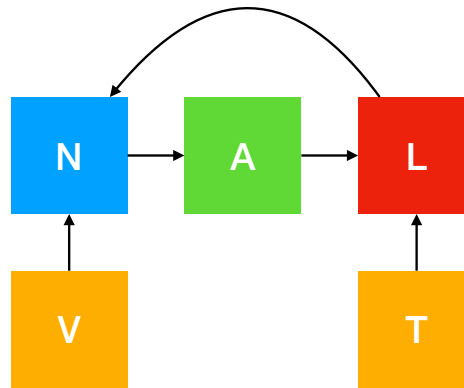
The more choices there are, the harder the decision is made

- Birden fazla sınıf olduğunda kayıp fonksiyonu
- Sigmoid yerine softmax
- Sınıf sayısı ile düzgün scale olma problemi



En temel yapay sinir ağı

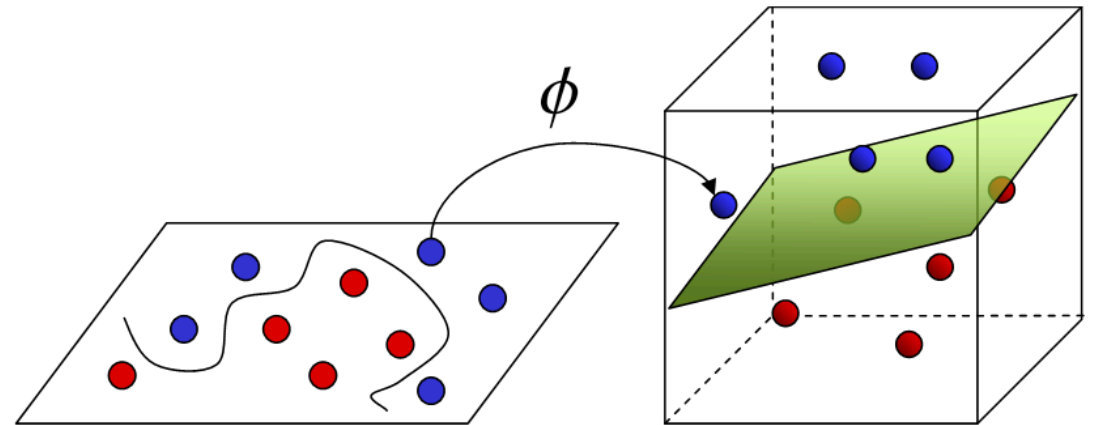
Back to basics



Boyut arttırımı

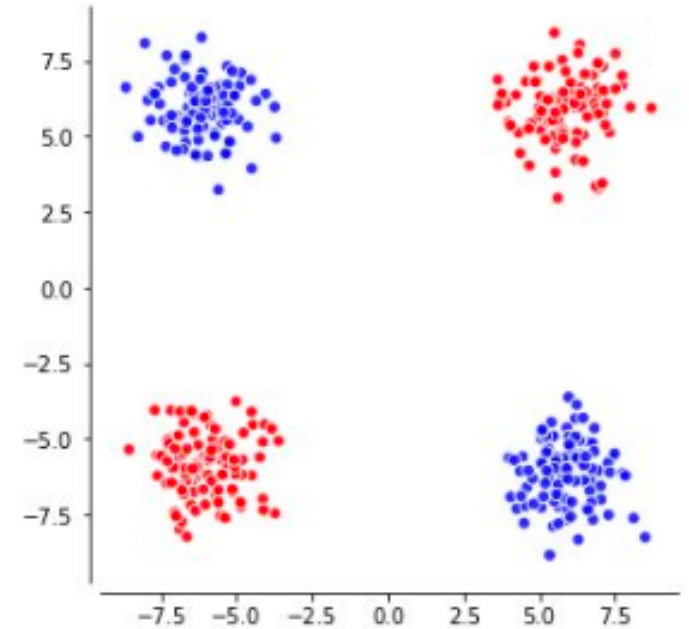
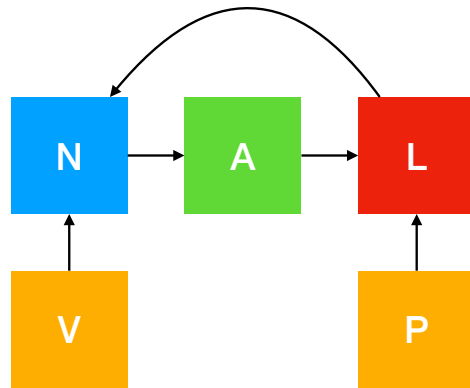
If can't figure it out why don't make it harder

- Basit problemlerin sınıflarının ayrıştırılmasında boyut arttırımı
- Hyperplane'ler



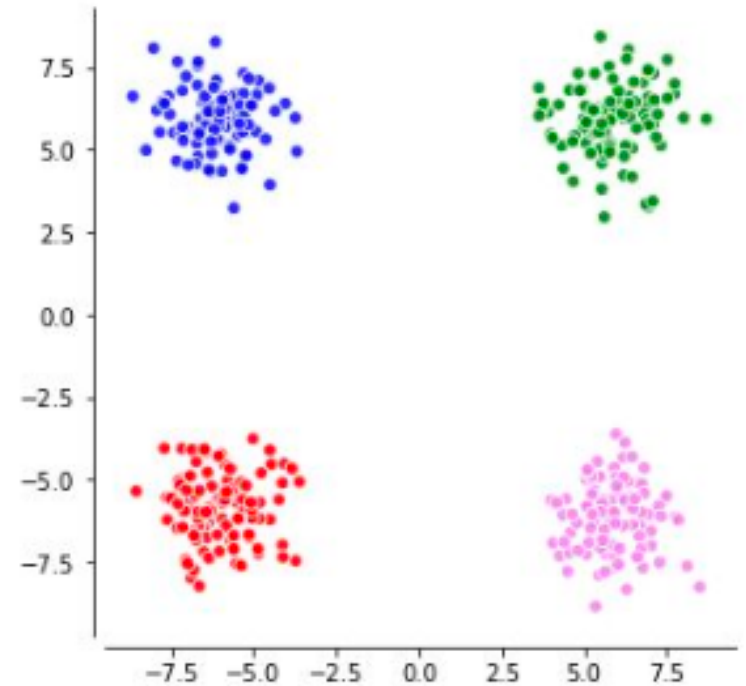
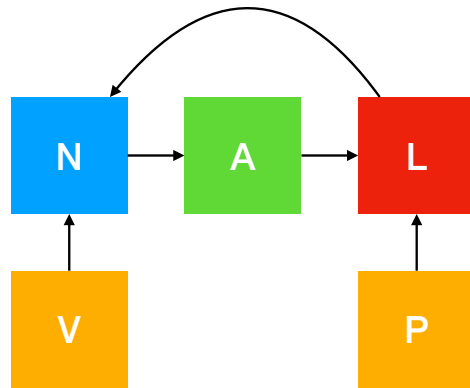
Daha derin sinir ağıları

Deep learning



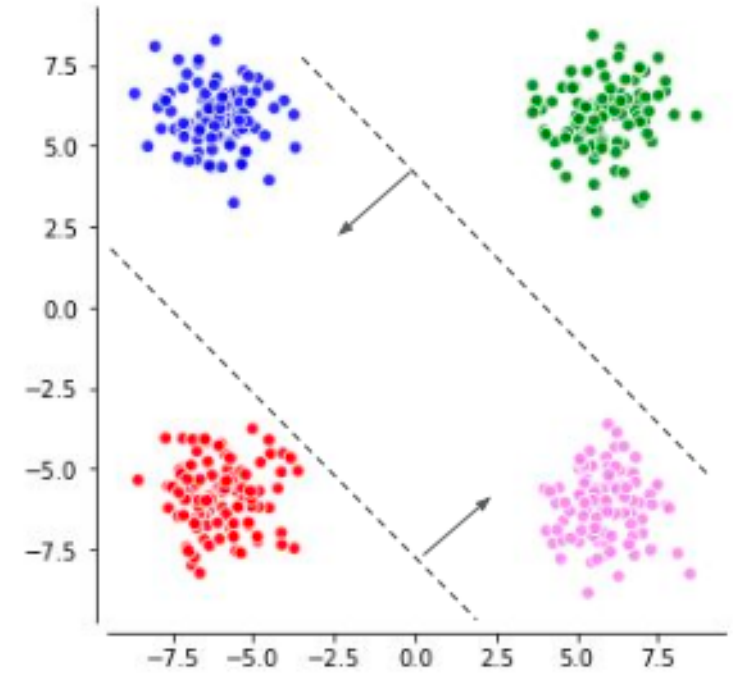
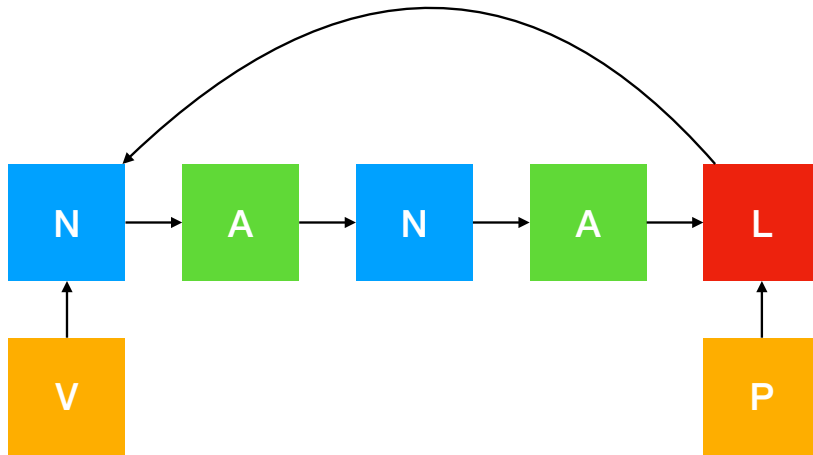
Daha derin sinir ağıları

Deeper learning



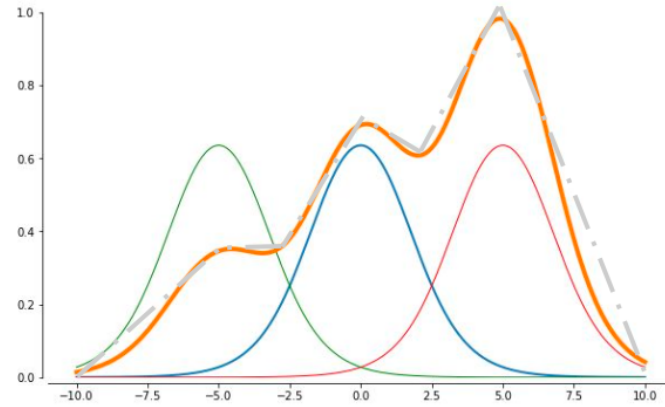
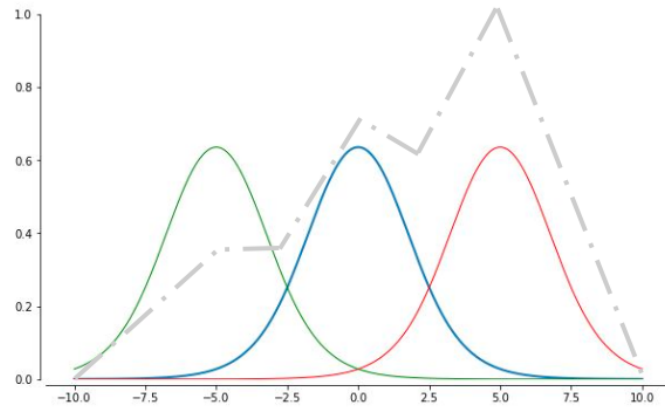
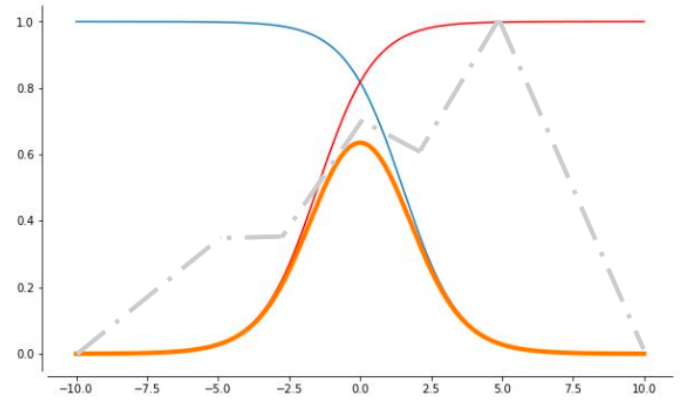
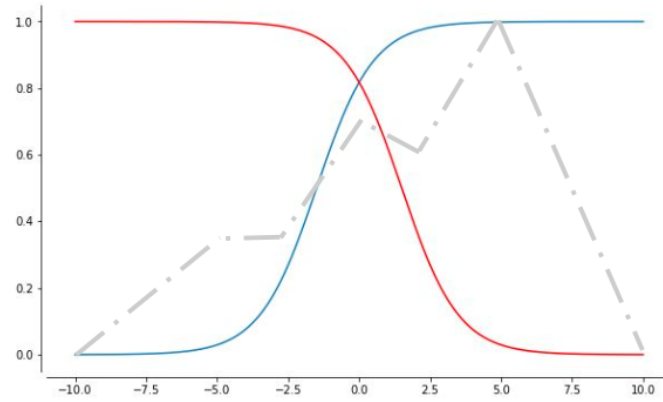
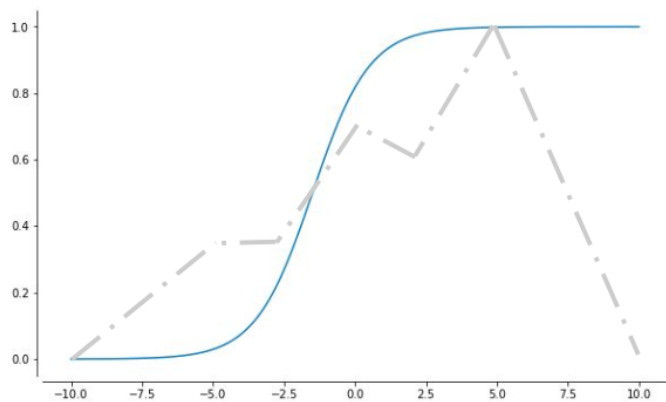
Daha derin sinir ağıları

That has no end I assume



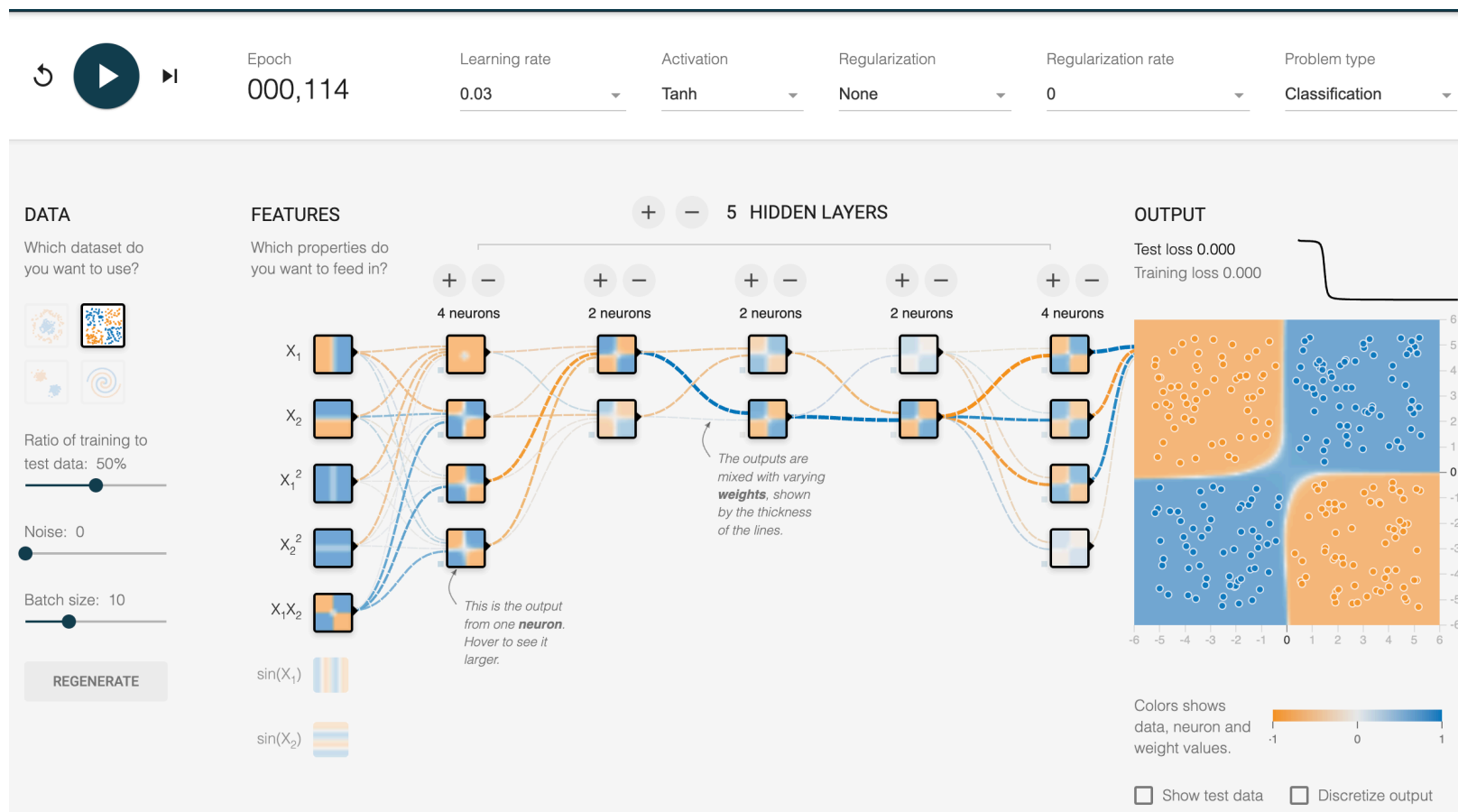
Daha derin sinir ağları

Lines everywhere



Tensorflow Playground

Now you know deep learning



Yapay sinir ağlarının özellikleri

Tips & Tricks

- Error rate olabildiğince küçültülebilir fakat 0 olamaz
 - Çarpma işlemi
- Bir modelin inputunu genişletmek polinomial karmaşıklık artışı sağlar
- Bir modelin derinliğini arttırmak eksponansiyel karmaşıklık artışı sağlar

Yapay sinir ağlarının özellikleri

Tips & Tricks

- Sigmoid yerine ReLU
 - Pozitif değerleri scale eder. Negatif değerleri sıfırlar
 - Negatif değerlere sahip düğümler olacaksa dikkatli kullanılmalı
- Learning gradyan hesaplamasıdır.
- Gradyan fonksiyonun türevidir.
- Gradyanların pozitif ya da negatife ilerlemesi nöronların önemini belirler.

Yapay sinir ağlarının özellikleri

Tips & Tricks

- Gradient Descent algoritması
 - Gradyanlarda minimizasyon sağlayamayana kadar devam etme amacı
- Parçaların gradyanları = Tüm gradyanın parçaları.
 - Deep learning'in tüm veriyi bir kerede vermeden eğitilmesini sağlayan temel
- Optimizer olarak Adam ile başlama
 - Çok nadir özelleştirmelerde değişmesi gerekir.

Yapay sinir ağlarının özellikleri

Tips & Tricks

- Model karmaşıklığı overfit'e sebep olabilir
 - Karmaşıklık azaltılabilir
 - L1 L2 Regülerizasyonları
 - Dropout

Yapay sinir ağlarının özellikleri

Tips & Tricks

- Overfit'i engellemek için
 - Veriye noise eklemek
 - Early stopping
 - Normalizasyon

Yapay sinir ağlarının özellikleri

Tips & Tricks

- İlk ağırlık değerlerinin seçimi önemli
 - Sıfır olması yanlış
 - Çok büyük değerler olması problem

Yapay sinir ağlarının özellikleri

Tips & Tricks

- Geliştirilen model küçük veride overfit olmalı
 - Olamıyorsa problem tanımında sıkıntı olabilir
 - Seçilen veride sıkıntı olabilir
 - Modelde sıkıntı olabilir
- Training loss takip edilmeli
- Accuracy her zaman doğru değeri vermez

Yapay sinir ağlarının özellikleri

Tips & Tricks

- Bakılabiliyorsa ağırlıkların normları incelenmeli
 - Sonsuza gidiyorsa bu düğümlerde sıkıntı var demektir
- Verilerin shape kontrolü yapılmalı
 - İstenilen şeyi yapmıyor olabilir
 - Makalelerin bir çoğu
 - CNN için küp boyutları

Yapay sinir ağlarının özellikleri

Tips & Tricks

- Aynı anda sadece bir parametre değiştirilmeli
- Bir çok parametre değiştirip sonuç gözlenmesi etki tespiti adına problem