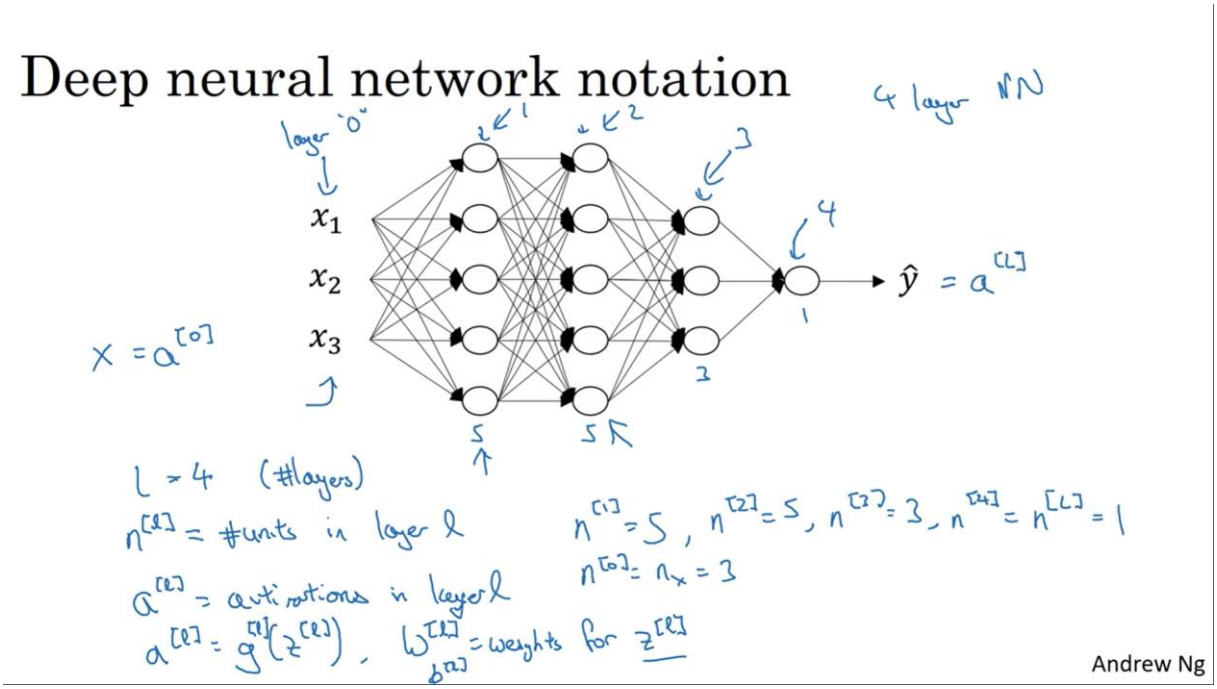
COURSE 1 W4

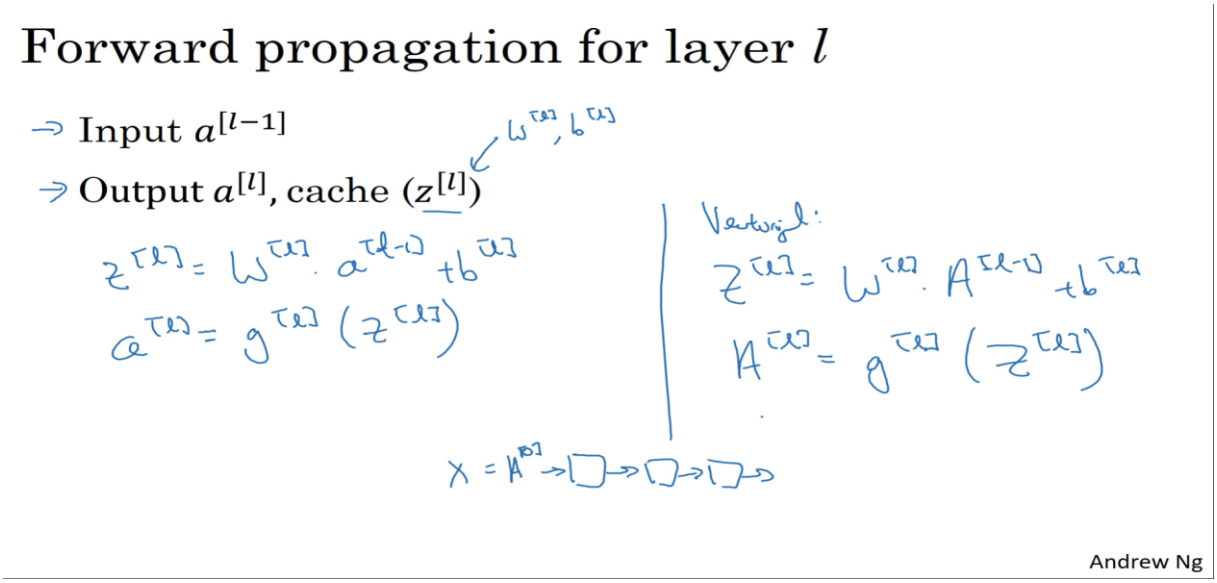
Vid 36

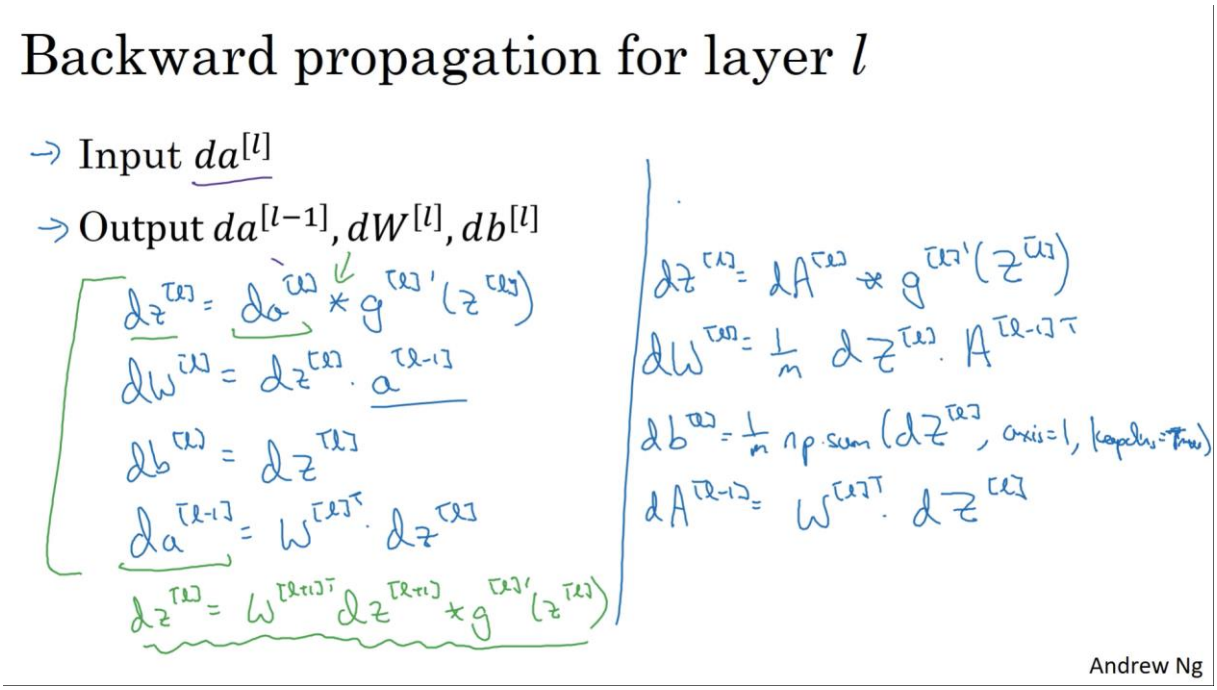
Deep L-Layer Neural Network



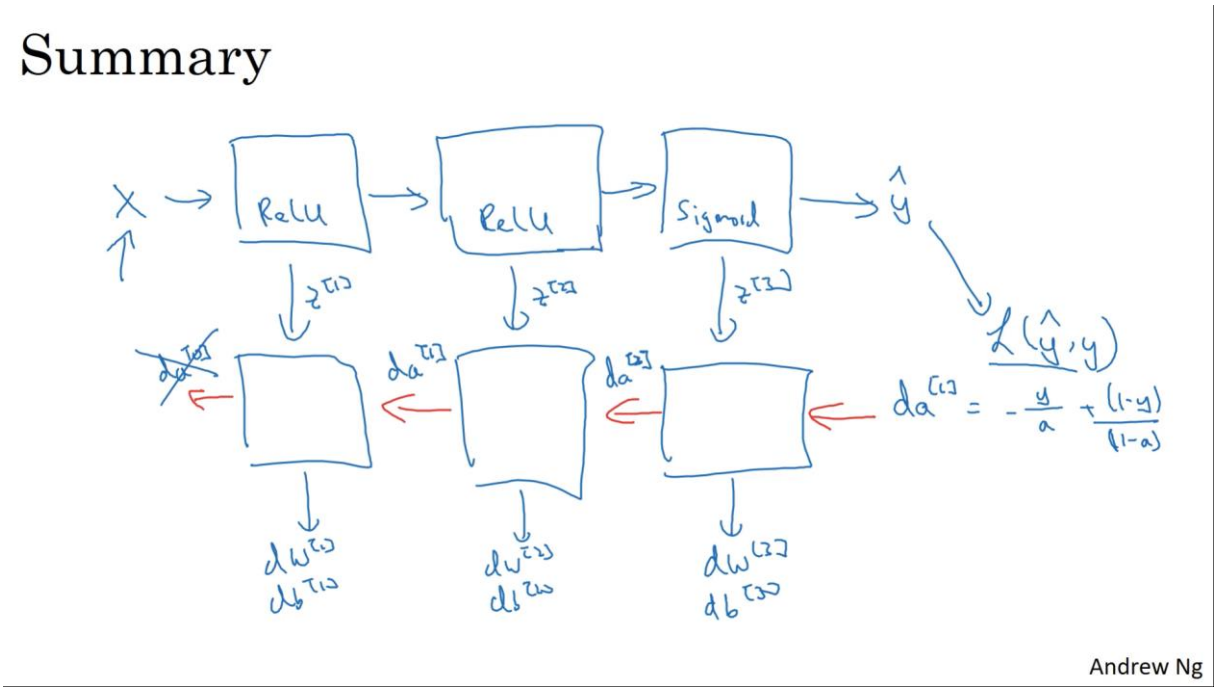
Vid 37

Forward and Backward Propagation





Sağdaki vectorized version.

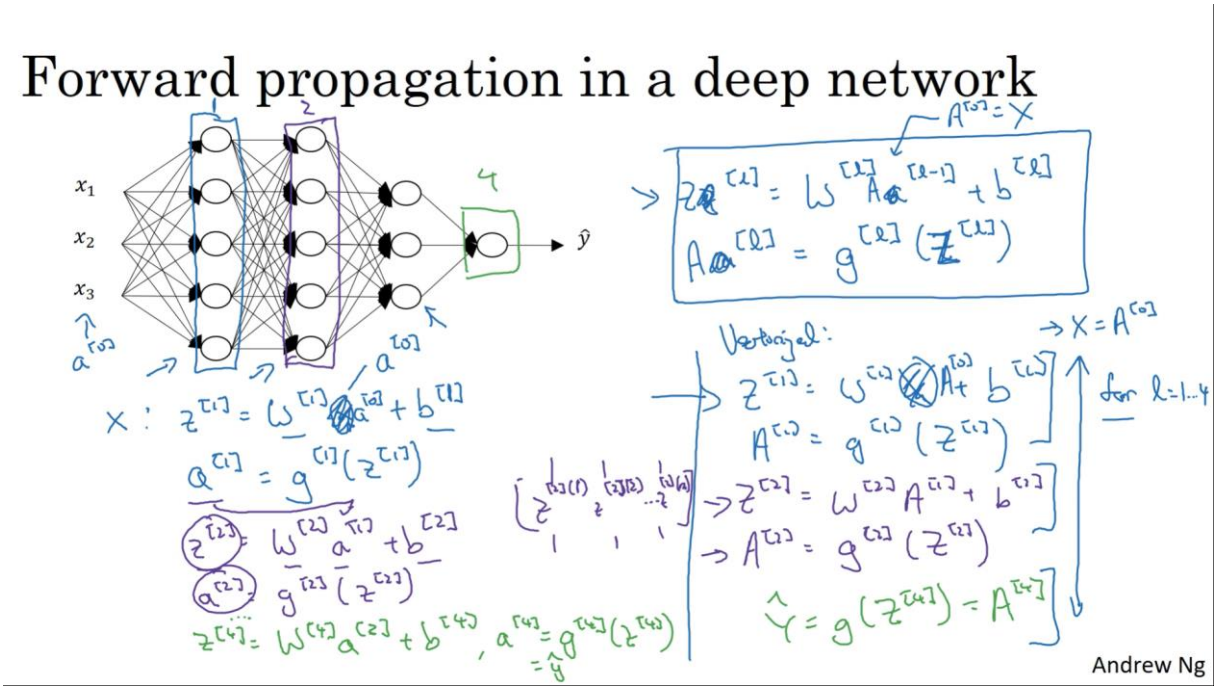


Özetle önce forward propagation ile y_hat ve loss hesaplanıyor daha sonra, aynı yapı üzerinde geriye doğru dw3 db3 den başlayarak geriye doğru geliniyor, z1,z2,z3 cache de tutuluyor sanırım türev hesaplamaları için kullanılıyor. Backwarda başlarken da(l) ile başlıyoruz, bu da loss'un y_hat'e göre türevinden başka bir şey değil.

Bunlar zaten bildiğimiz şeyler yeni bir şey değil. Özellikle bakman gerekirse assignment üzerinde dener bakarsın.

Vid 38

Forward Propagation in a Deep Neural Network

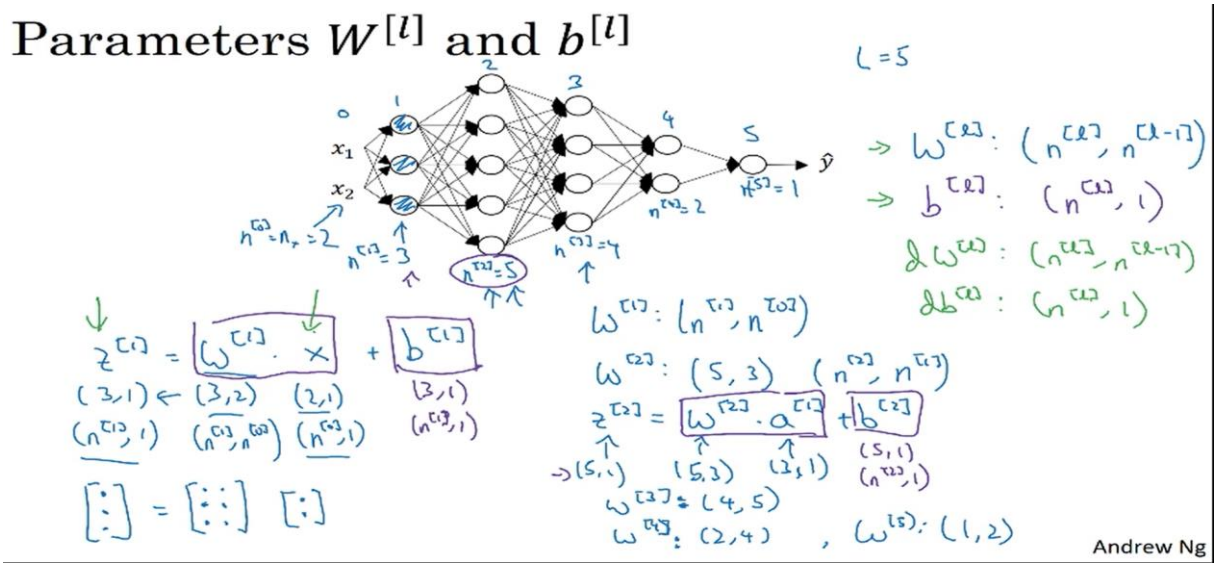


Burada sağ alttaki vectorized version için her layer için dahi her layer'ı dolandığımıza dikkat et yani burada bir for loop söz konusu, bunu vectorize edemiyoruz.

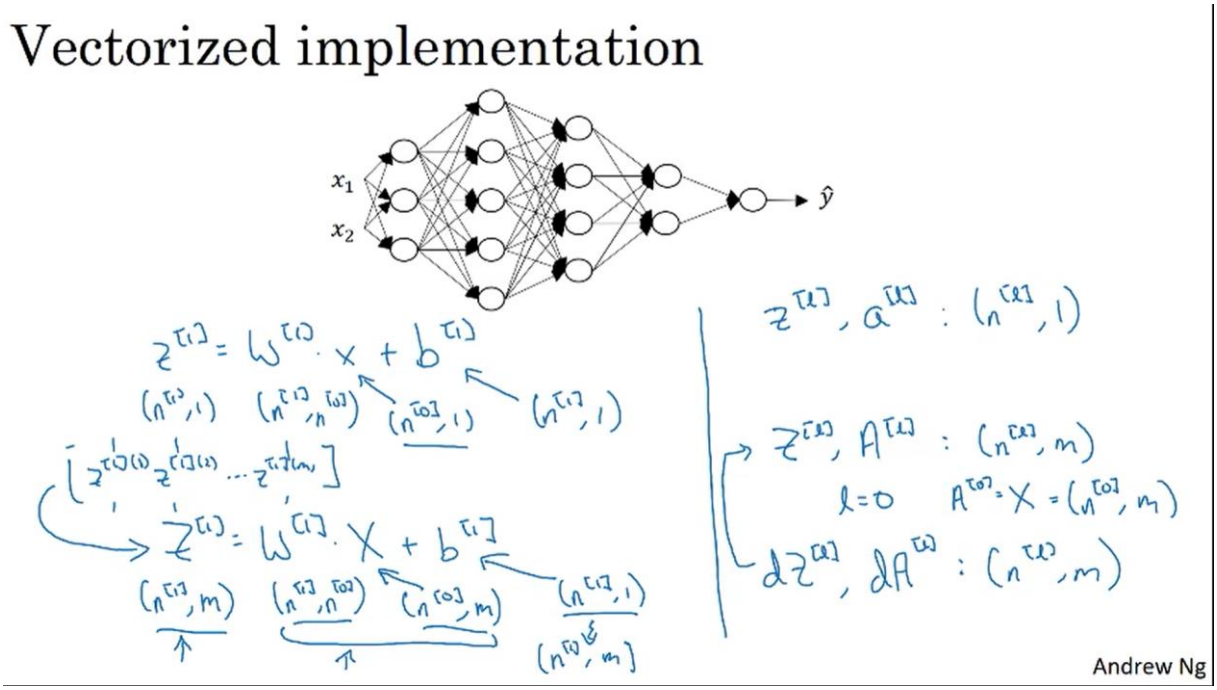
Vid 39

Getting Matrix Dimensions Right

NN implement ederken kullanılabilecek debugging tool'lardan biri, matrix dimensionları bir kağıda yazmaktır.



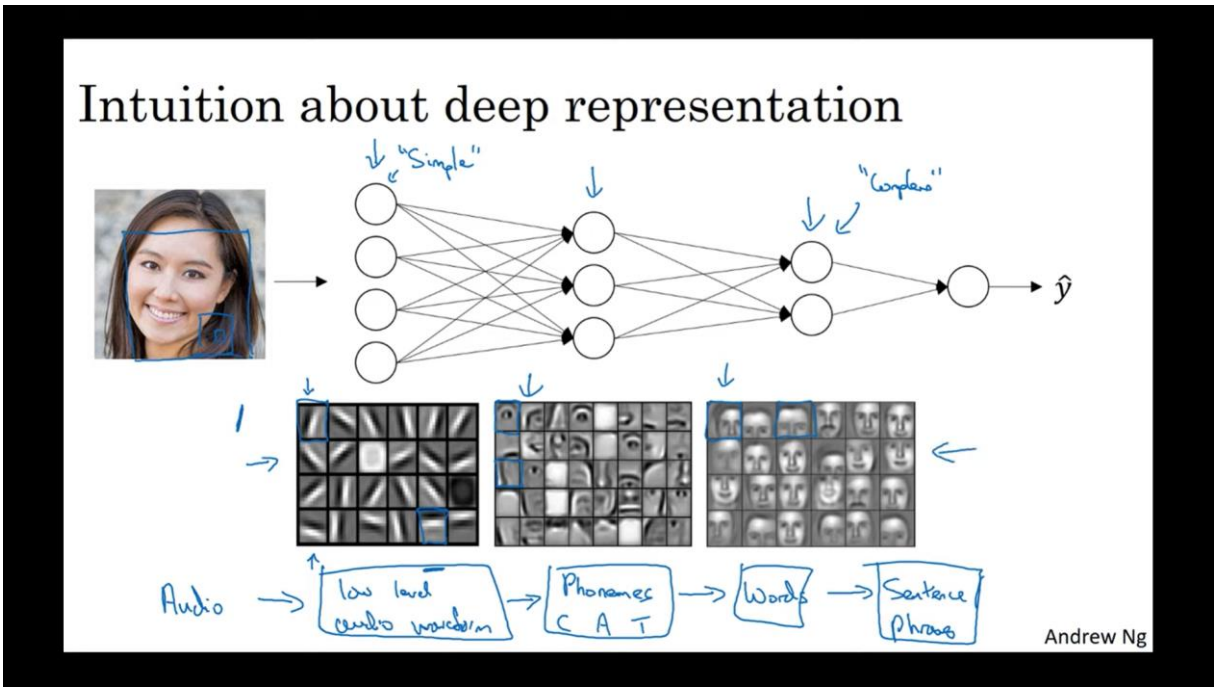
Yukarıda sağ üstte görülenler beklenen matris boyutları, kodlama sırasında bunları bilmek yardımcı olur.



Benzer şekilde yine sağdaki kısımda beklenen matrix boyutlarını görebiliyoruz.

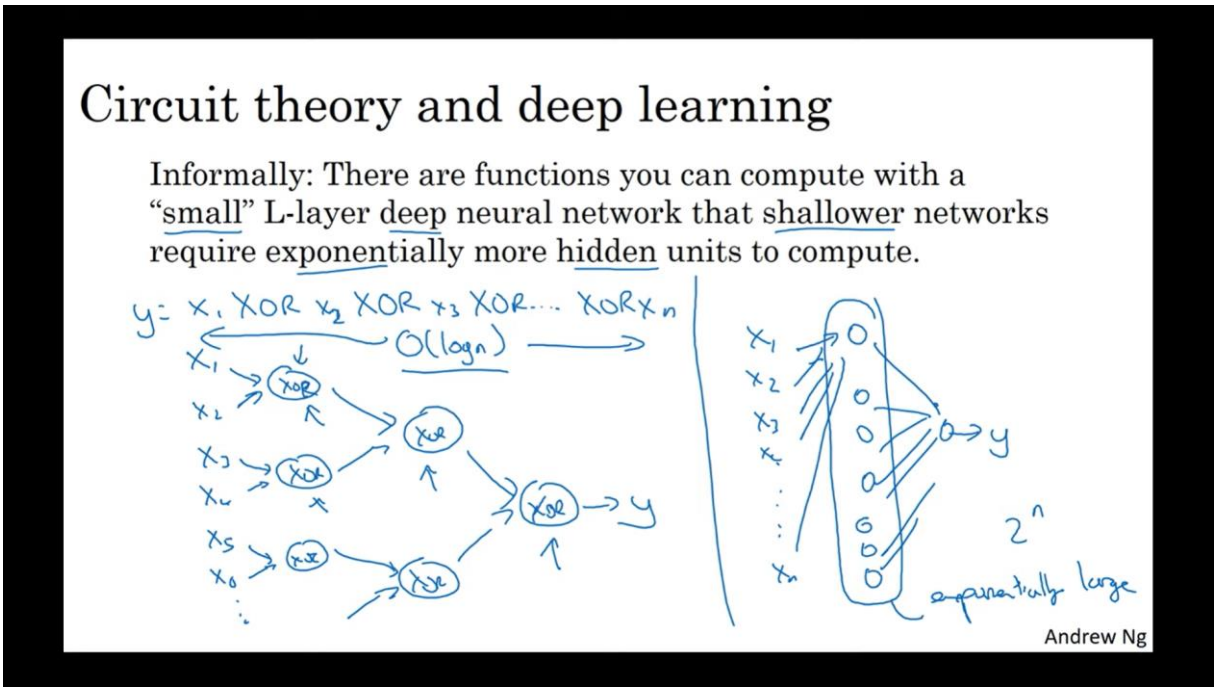
Vid 40

Why Deep Representations?



Özünde olay bildiğimiz gibi, NN derinleştikçe daha complex feature'lar ortaya çıkıyor.

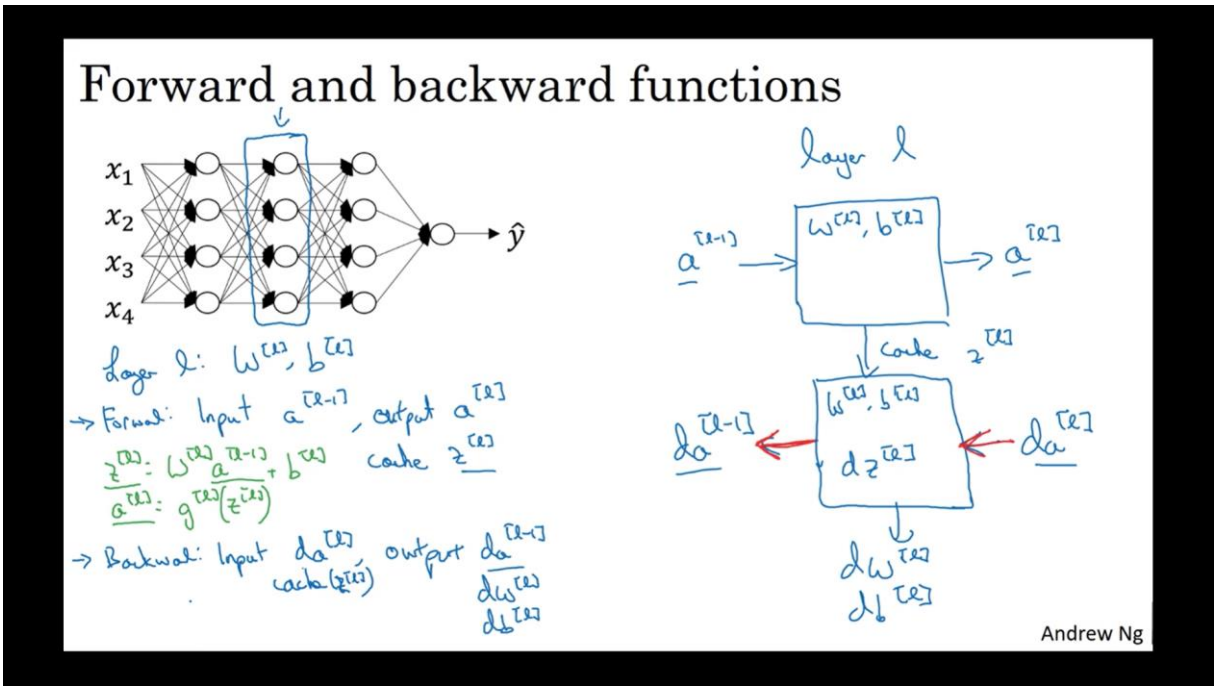
Bir başka sebep ise aşağıdaki:



Aynı fonksiyon için shallow NN'de birsürü unit gerekirken, deep NN için daha az unit ile işi halledebiliriz.

Vid 41

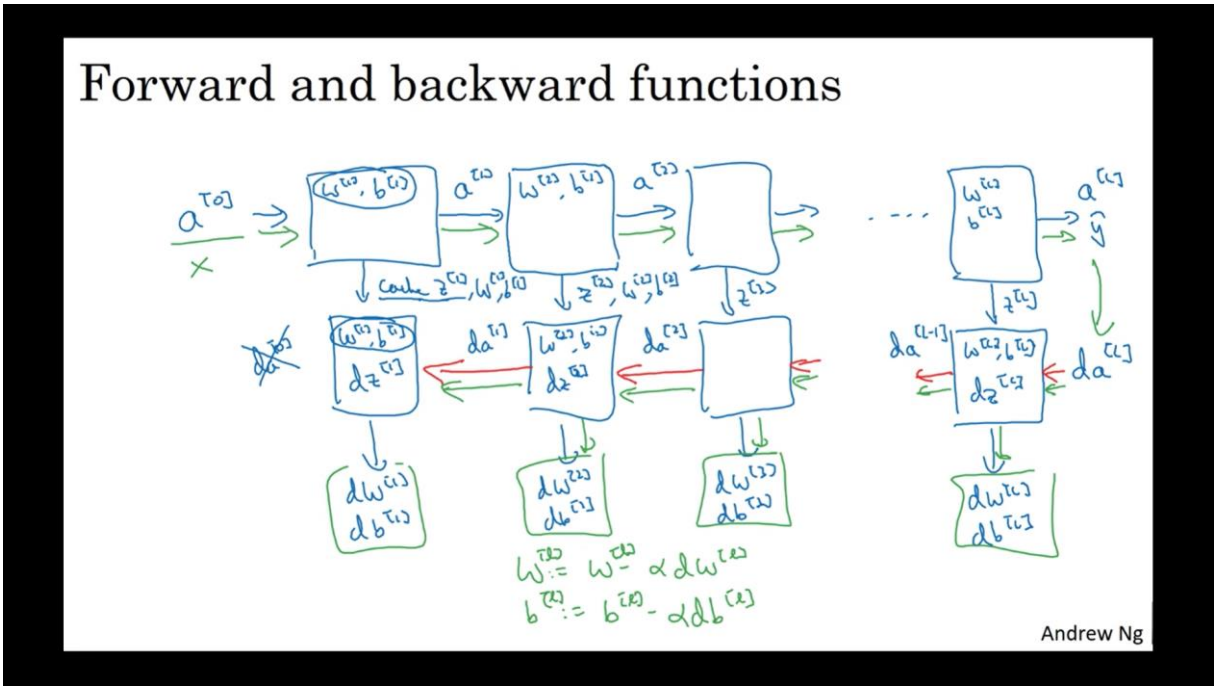
Building Blocks of a Deep Neural Network



Seçilen layer için forward ve backward computations'ı tekrar gözden geçiriyoruz. Sonuçta forward için al-1'den al hesaplanıyor ve bunun için wl ve bl bilgisini kullanıyoruz bu sırada backward computation için kullanılmak üzere zl'i cache'liyoruz.

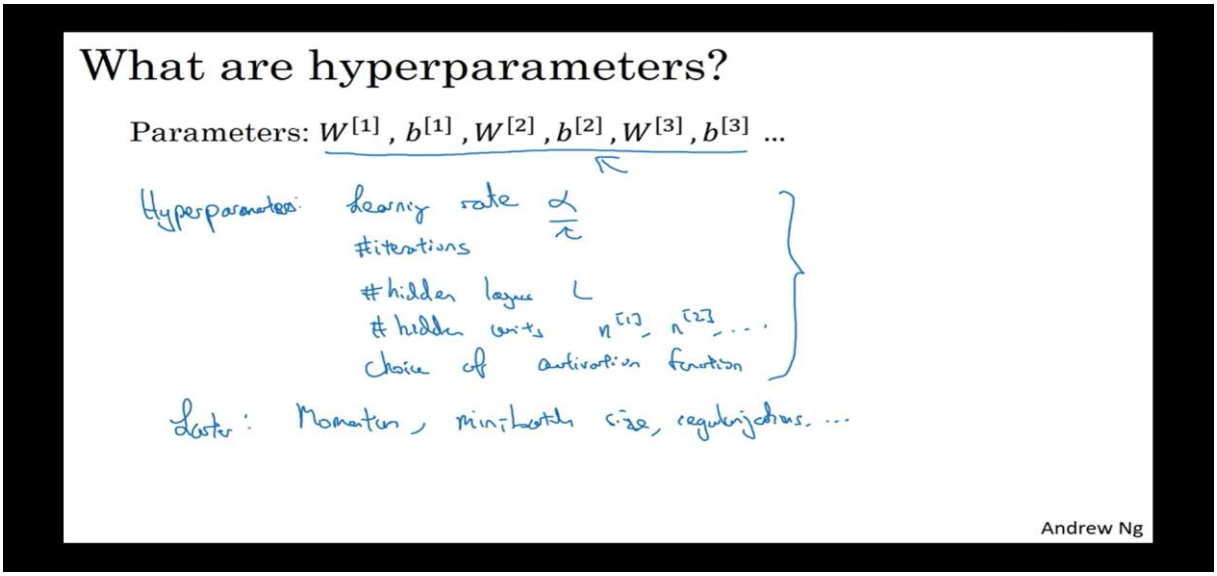
Backward için dal'den dal-1, hesaplıyoruz bu sırada wl,bl,dzl kullanılıyor ve dwl dbl de output olarak verilebilir.

Sonuçta forward ve backward computation'ı her layer için hesapladığımızı düşünersek aşağıdaki diagram ortaya çıkar.

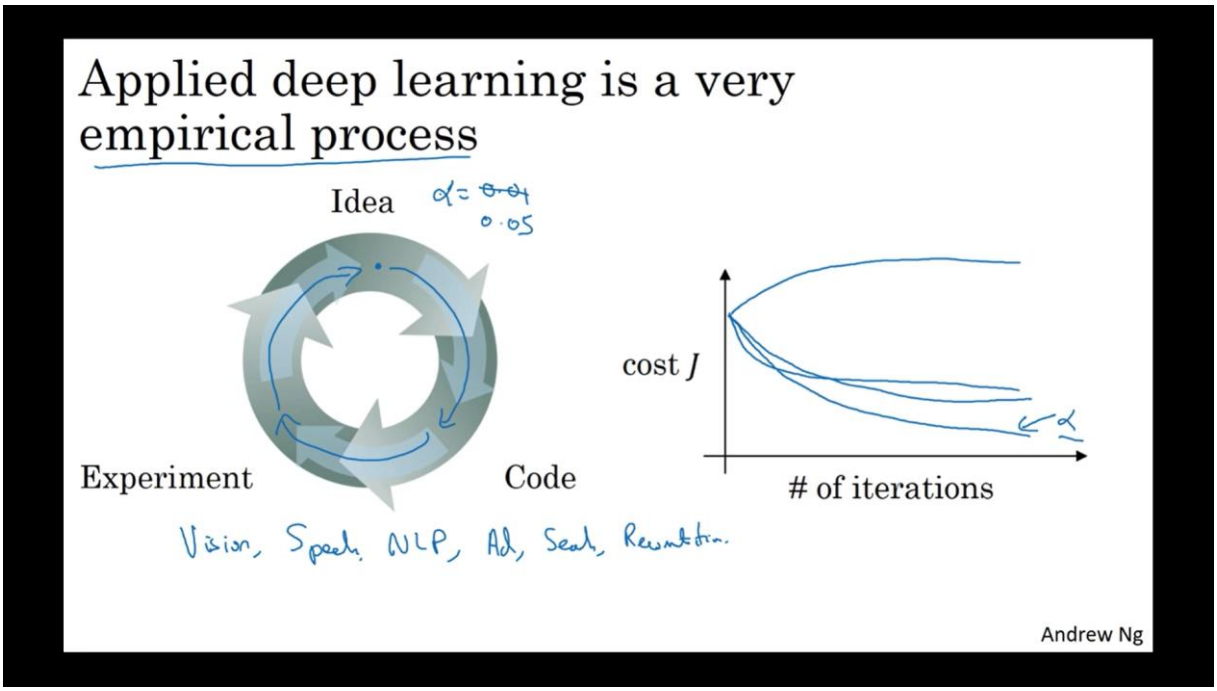


Vid 42

Parameters and Hyperparameters



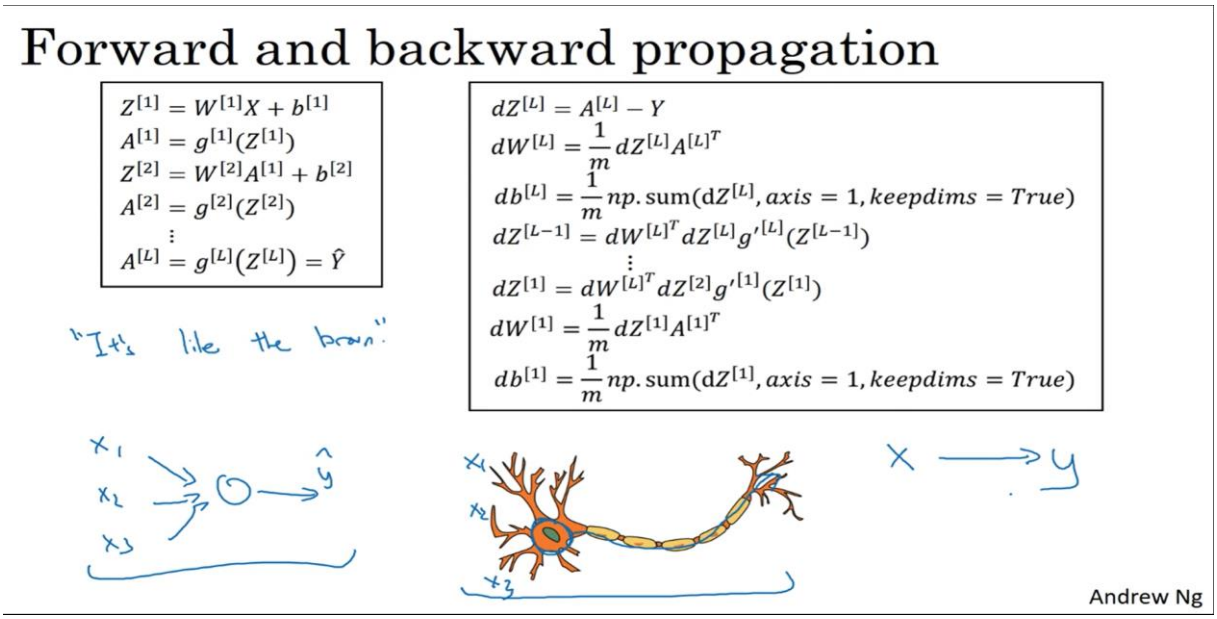
Hyperparameters dediğimiz parametreler, W ve b gibi final parametreleri etkileyen diğer parametreler olarak düşünülebilir, learning rate'e veya iteration'a göre benim W ve b parametrelerim değişecektir. Daha sonra göreceğimiz, momentum, minibatch size, regularization parameters gibi başka hyperparameters da söz konusudur.



Sonuçta uygulama hala empirik yani deneysel gerçekleşiyor, en iyi hyperparameterları bulmanın kesin bir yolu yok, deneme yanılma ile buluruz, daha sonra bu deneme yanılma için de sistematik bazı yollardan bahsedilecek. Ancak bulunan parametre bile bir süre sonra değiştirilebilir, CPUnun GPU'nun yapısından dolayı dahi sonuçlar değişebilir.

Vid 43

What does this have to do with brain?



Sonuçta dediği şey şu, bir log. Reg. Unit neuran'a benzese de bu pek de önemsenecek bir benzerlik değil, analoji kurmak doğru olmaz diyor. Tek bir neuronun bile henüz tam olarak neler yaptığı açık değil. Ayrıca beyinde backpropagation gibi bir mekanizma var mı yok mu belirsiz, nasıl öğreniyor belirsiz.

Bunun yerine supervised learning'i son derece komplex fonksiyonların öğrenilmesi olarak düşünmek daha mantıklı, en nihayetinde x ile y'yi map ediyoruz.