

Autoencoders

Deep Learning

Autoencoders

Aprendizagem supervisionada	Aprendizagem não supervisionada
Redes Neurais Artificiais classificação e regressão	Mapas auto organizáveis detecção de características e agrupamento
Redes Neurais Convolucionais visão computacional	Boltzmann machines sistemas de recomendação redução de dimensionalidade
Redes Neurais Recorrentes análise de séries temporais	Auto encoders redução de dimensionalidade
	Redes adversariais generativas geração de imagens

Material Complementar

Livro **Autoencoder Neural Network: A Performance Study Based on Image Reconstruction, Recognition and Compression** de **Chun Chet Tan**: livro teórico que apresenta as características, treinamento e como avaliar autoencoders

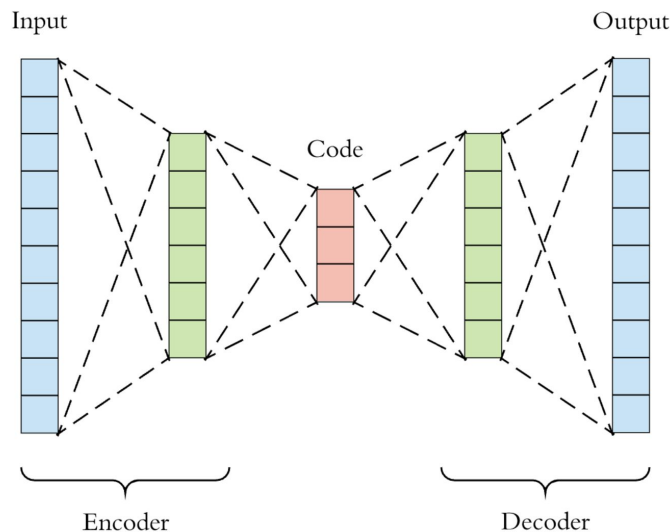
Tutorial **Introduction to Autoencoders** de **Jeremy Jordan**: mostra o funcionamento básico dos autoencoders e apresenta indicações de vários outros materiais complementares

UFLDL Tutorial: um dos materiais mais recomendados sobre a teoria e teoria dos autoencoders

Building Autoencoders in Keras: tutorial da documentação do Keras para criação de autoencoders (o material deste curso foi baseado neste tutorial)

Autoencoders

Autoencoders são arquiteturas neurais surpreendentemente simples. Eles são basicamente uma forma de compactação, semelhante à forma como um arquivo de áudio é compactado em MP3 ou um arquivo de imagem é compactado em JPEG.



Autoencoders

Os codificadores automáticos estão intimamente relacionados à análise de componentes principais (PCA). Na verdade, se a função de ativação usada no autoencoder é linear dentro de cada camada, as variáveis latentes presentes no gargalo (a menor camada na rede, também conhecida como código) correspondem diretamente aos componentes principais do PCA. Geralmente, a função de ativação usada em autoencoders é não linear, as funções de ativação típicas são ReLU (Rectified Linear Unit) e sigmóide.

Autoencoders

A matemática por trás das redes é bastante fácil de entender, portanto, vou analisá-la brevemente. Basicamente, dividimos a rede em dois segmentos, o codificador e o decodificador.

$$\phi : \mathcal{X} \rightarrow \mathcal{F}$$

$$\psi : \mathcal{F} \rightarrow \mathcal{X}$$

$$\phi, \psi = \arg \min_{\phi, \psi} \|X - (\psi \circ \phi)X\|^2$$

Autoencoders

A função de codificador, denotada por ϕ , mapeia os dados originais X , para um espaço latente F , que está presente no gargalo. A função decodificadora, denotada por ψ , mapeia o espaço latente F no gargalo para a saída. A saída, neste caso, é igual à função de entrada. Portanto, estamos basicamente tentando recriar a imagem original após alguma compressão não linear generalizada.

A rede de codificação pode ser representada pela função de rede neural padrão passada por uma função de ativação, onde \mathbf{z} é a dimensão latente.

$$\mathbf{z} = \sigma(\mathbf{W}\mathbf{x} + \mathbf{b})$$

Autoencoders

Da mesma forma, a rede de decodificação pode ser representada da mesma maneira, mas com peso, polarização e funções de ativação potencialmente diferentes sendo usadas.

$$\mathbf{x}' = \sigma'(\mathbf{W}'\mathbf{z} + \mathbf{b}')$$

A função de perda pode então ser escrita em termos dessas funções de rede, e é essa função de perda que usaremos para treinar a rede neural por meio do procedimento de retropropagação padrão.

$$\mathcal{L}(\mathbf{x}, \mathbf{x}') = \|\mathbf{x} - \mathbf{x}'\|^2 = \|\mathbf{x} - \sigma'(\mathbf{W}'(\sigma(\mathbf{W}\mathbf{x} + \mathbf{b})) + \mathbf{b}')\|^2$$

Autoencoders

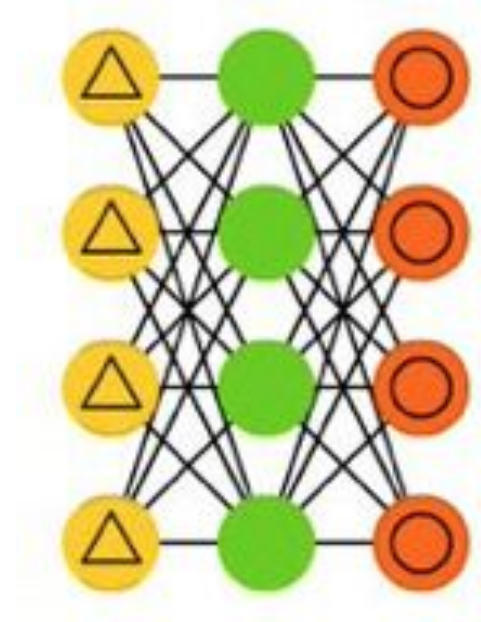
Como a entrada e a saída são as mesmas imagens, isso não é realmente um aprendizado supervisionado ou não supervisionado, portanto, normalmente chamamos isso de aprendizado autossupervisionado. O objetivo do autoencoder é selecionar nossas funções de codificador e decodificador de forma que exijamos o mínimo de informação para codificar a imagem de forma que ela possa ser regenerada no outro lado.

Se usarmos poucos nós na camada de gargalo, nossa capacidade de recriar a imagem será limitada e regeneraremos imagens borradas ou irreconhecíveis do original. Se usarmos muitos nós, não haverá sentido em usar compactação.

Autoencoders Automáticos

Denoising Autoencoders

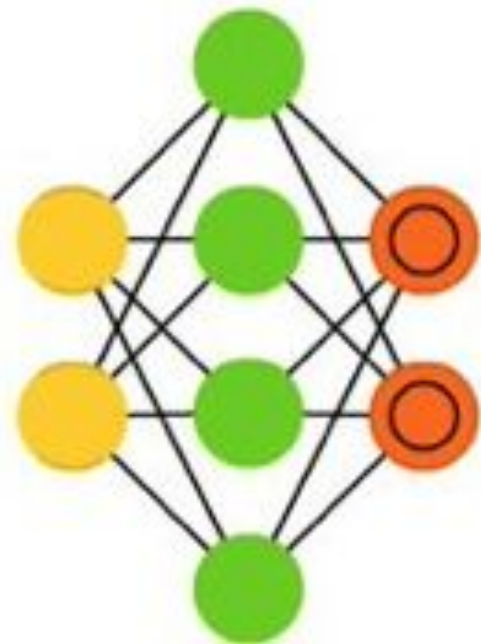
Esses codificadores automáticos adicionam algum ruído branco aos dados antes do treinamento, mas comparam o erro com a imagem original durante o treinamento. Isso força a rede a não se ajustar ao ruído arbitrário presente nas imagens.



Autoencoders

Autoencoders esparsos

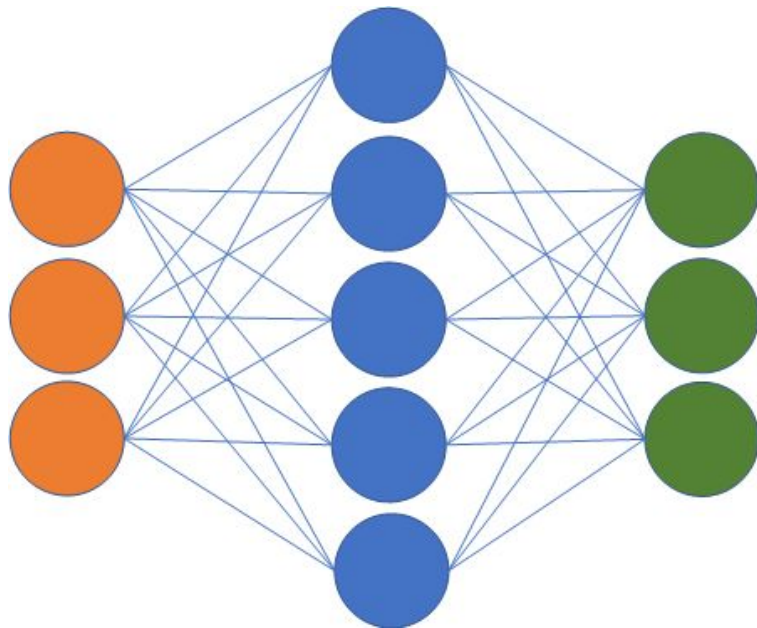
Um autencoder esperso, contra-intuitivamente, tem uma dimensão latente maior do que as dimensões de entrada ou saída. No entanto, cada vez que a rede é executada, apenas uma pequena fração dos neurônios dispara, o que significa que a rede é inerentemente 'esparsa'. Isso é semelhante a um autoencoder de eliminação de ruído, no sentido de que também é uma forma de regularização para reduzir a propensão para a rede ajustar-se excessivamente.



Autoencoders

Autoencoder de contração

Os codificadores de contração são praticamente os mesmos que os dois últimos procedimentos, mas, neste caso, não alteramos a arquitetura e simplesmente adicionamos um regularizador à função de perda. Isso pode ser considerado uma forma neural de regressão de cristas.

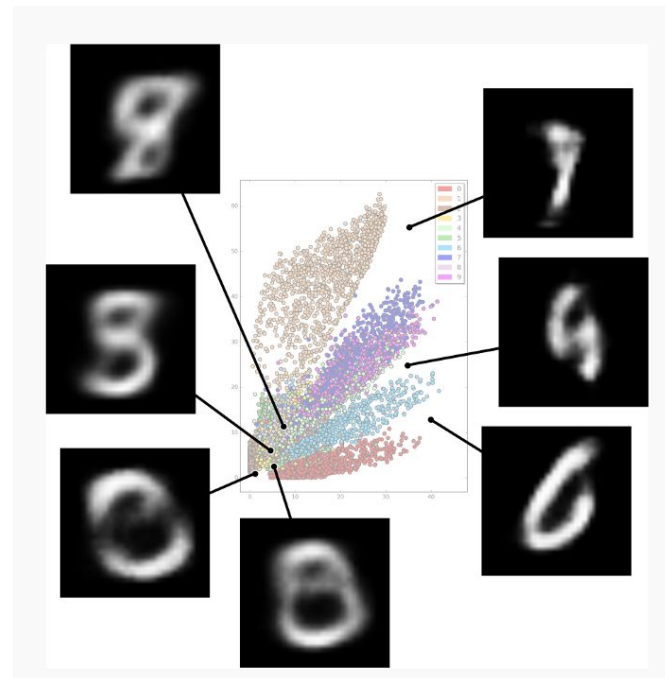


Autoencoders

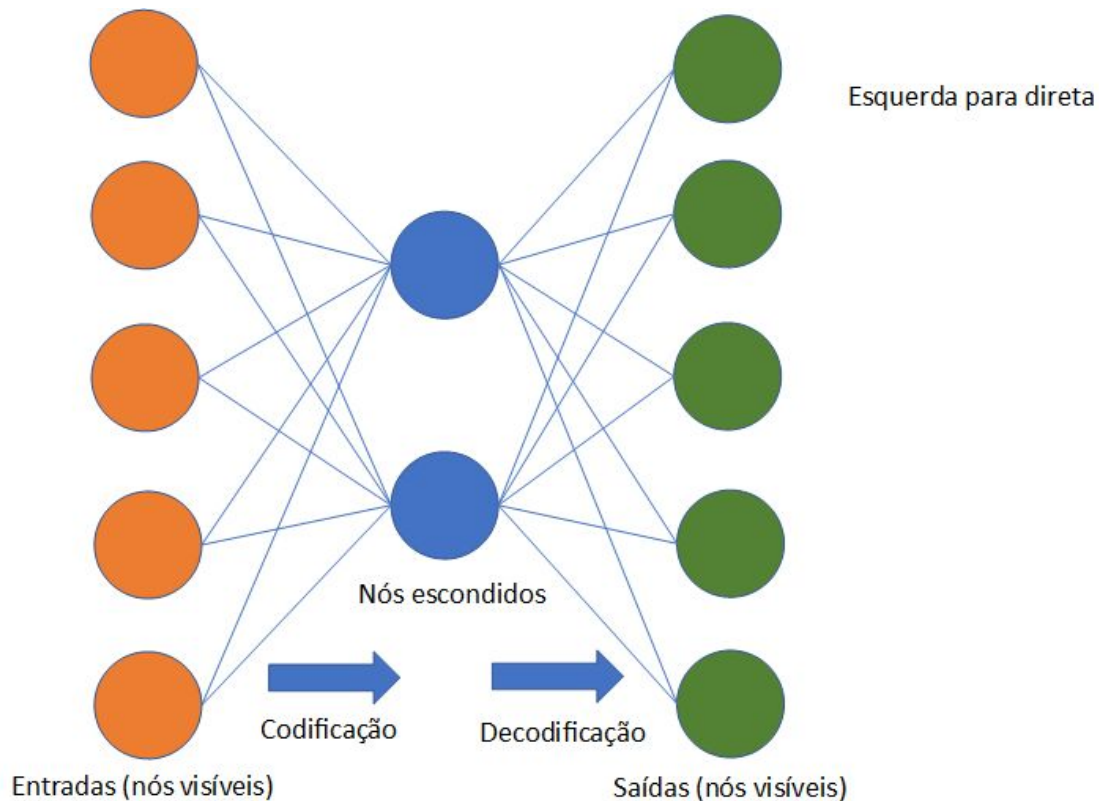
Portanto, agora que entendemos como os codificadores automáticos são, precisamos entender no que eles não são bons. Alguns dos maiores desafios são:

- Lacunas no espaço latente
- Separabilidade no espaço latente
- Espaço latente discreto

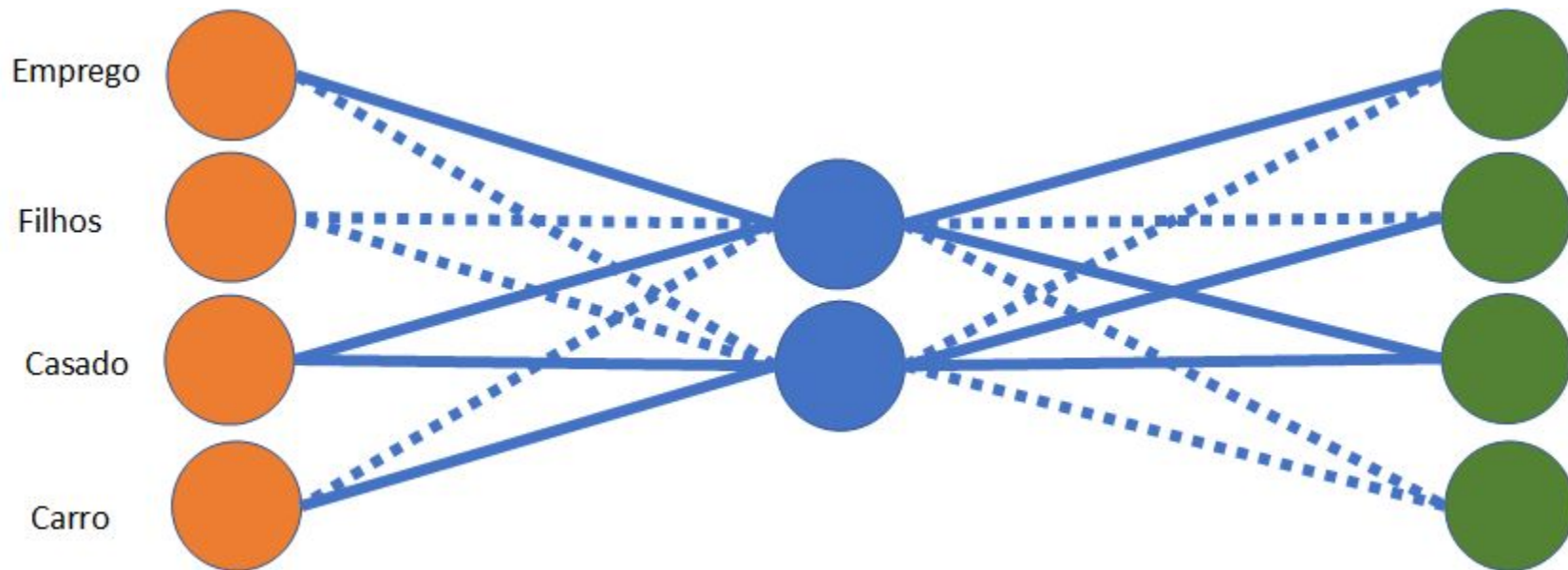
Todos esses problemas podem ser ilustrados neste diagrama.



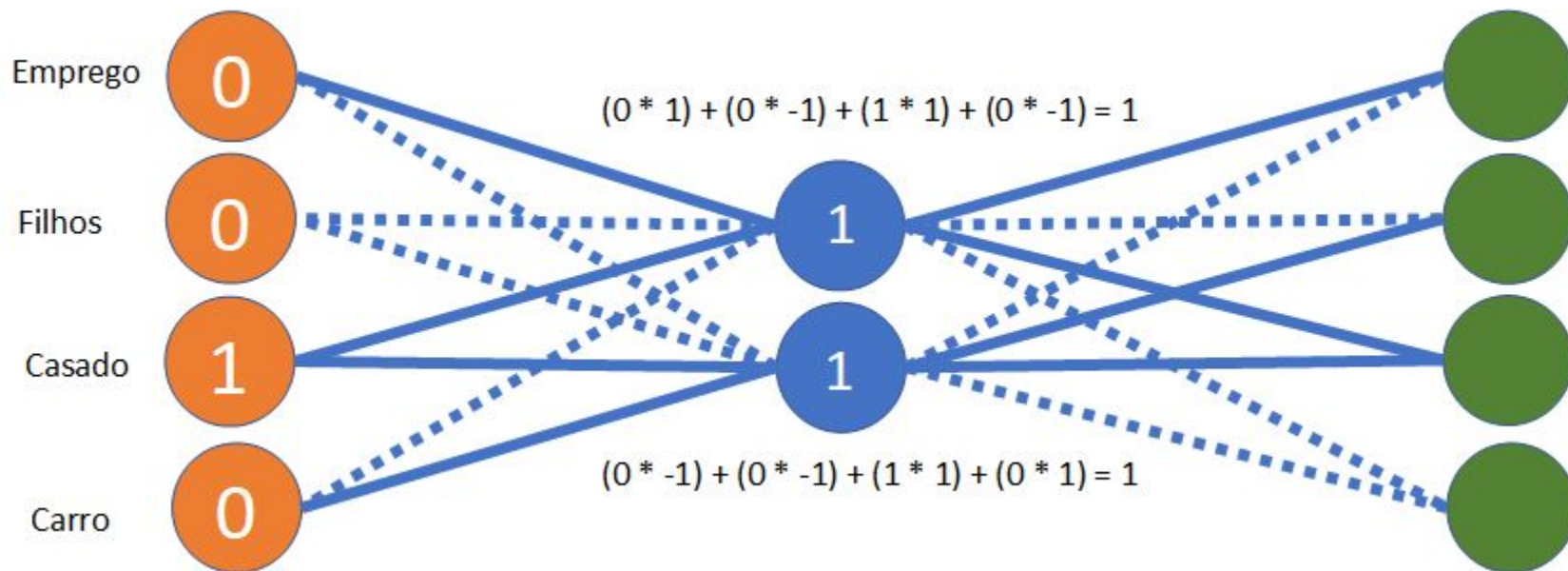
Autoencoders (*exemplo ilustrativo)



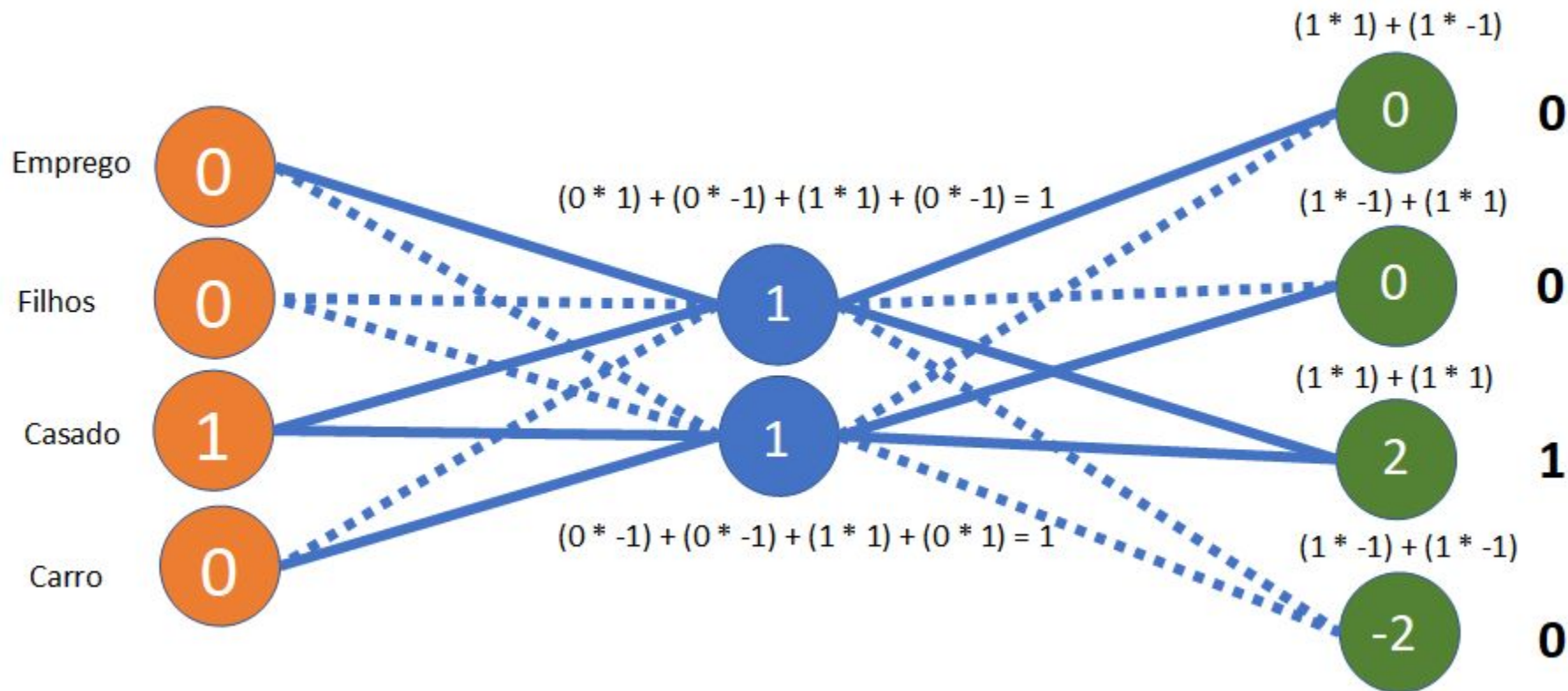
Autoencoders (*exemplo ilustrativo)



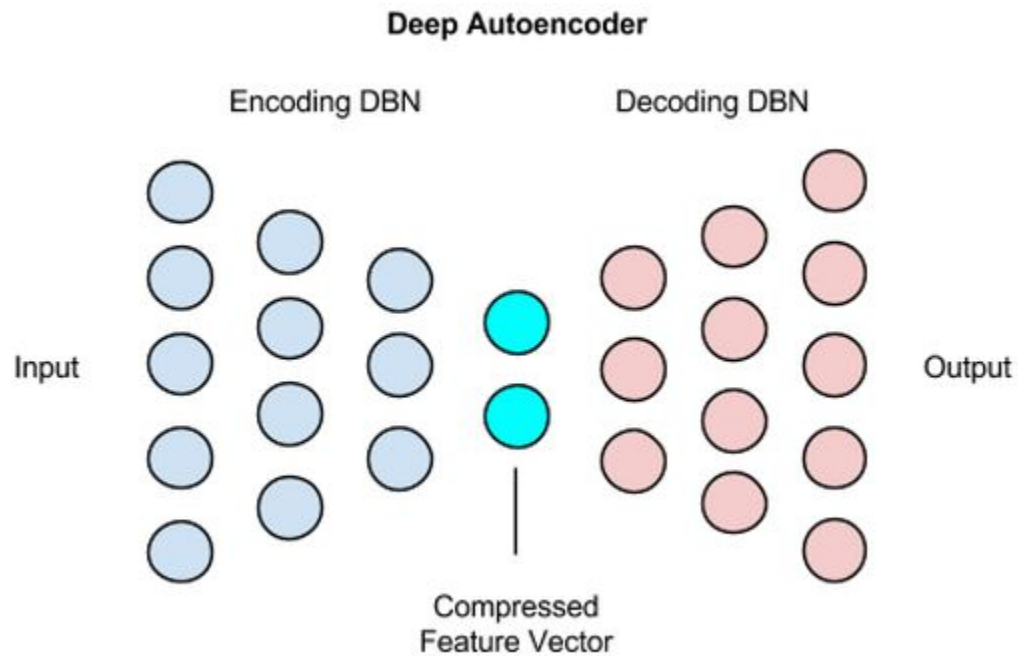
Autoencoders (*exemplo ilustrativo)



Autoencoders (*exemplo ilustrativo)



Deep autoencoder



Autoencoders (*esempio illustrativo)

