

Auto-Encoders

Los auto-encoders son un tipo de redes neuronales de aprendizaje no supervisado en donde

básicamente el OUTPUT es igual al INPUT de los pesos neuronales.

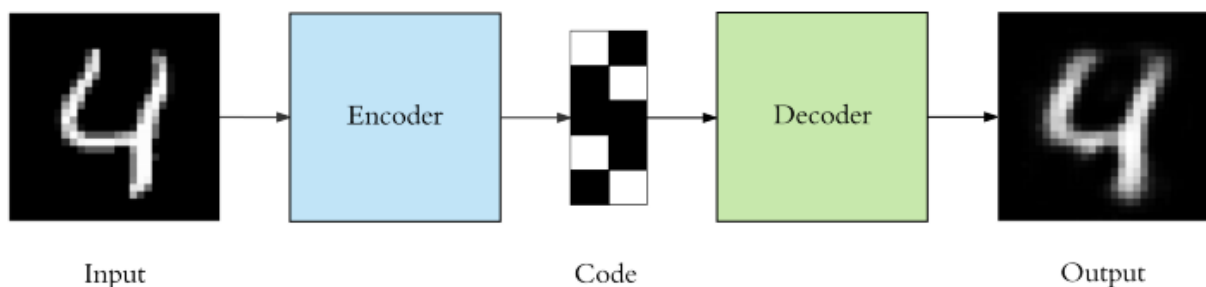
Los auto-encoders comprimen el INPUT en un código de menor dimensión y posteriormente reconstruyen la salida a partir de esta representación.

Codificador (Encoder): Es la porción de la red neuronal encargada de comprimir los datos de entrada,

típicamente en forma de vector, conocido como bottleneck feature.

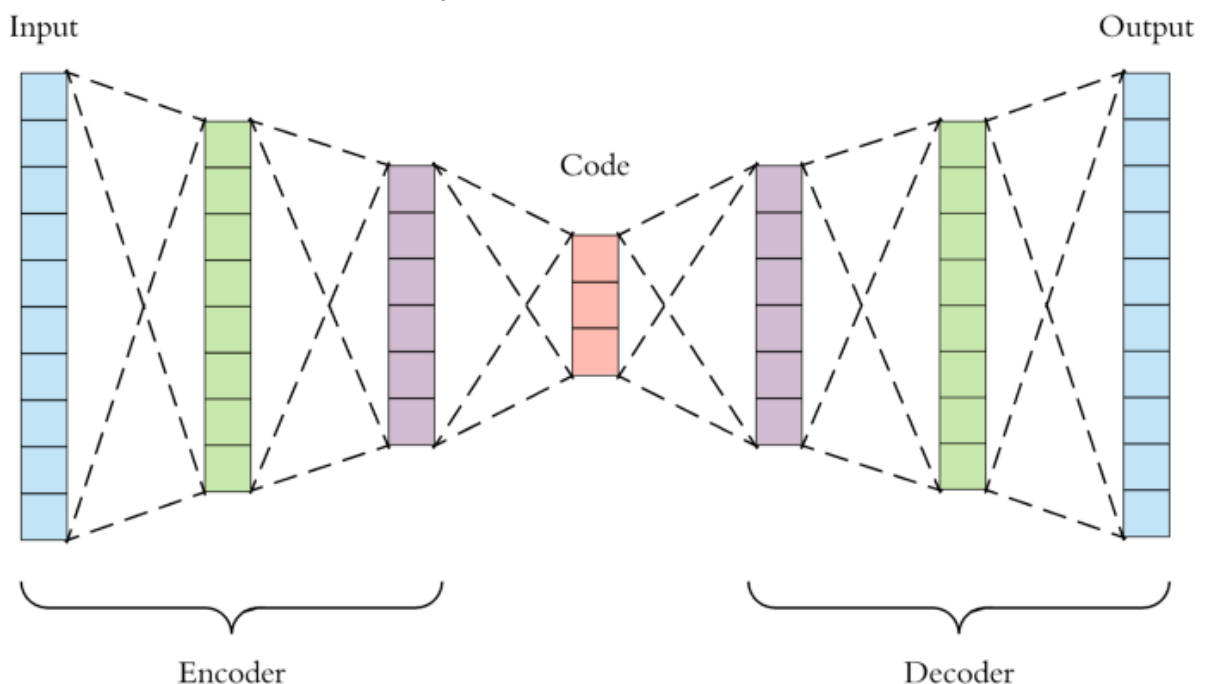
Bottleneck feature: Un feature cuello de botella es la capa media que divide al encoder y al decoder. Es la representación comprimida de la entrada, usualmente bastante más pequeña en cuanto a dimensiones e información.

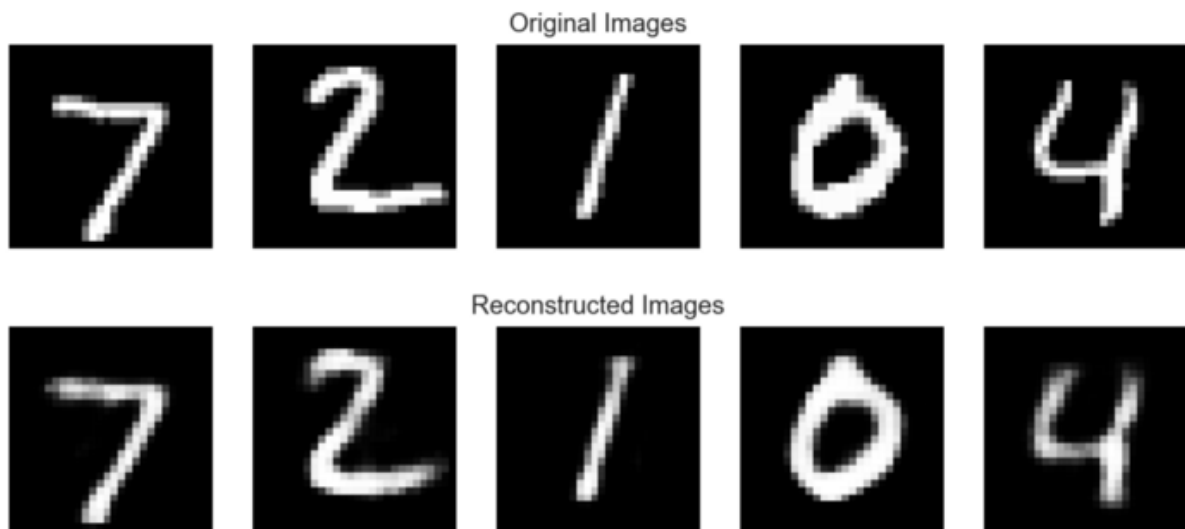
Decodificador (Decoder): Es la segunda parte de la red neuronal, a cargo de reconstruir los datos de entrada originales a partir de la representación comprimida resultante.



La utilidad de los auto-encoders está radicada en distintas áreas, pero es predominante en el deep learning orientado a imágenes y reconocimiento, el aprendizaje supervisado es aquel a partir de una data, ya sea con valores continuos o discretos, se busca hallar una correlación o básicamente encontrar el Y para cada X.

Por lo tanto un auto-encoder trabaja con datos no correlacionados.





La principal ventaja de los autoencoders es que no tienen ninguna restricción en cuanto al tipo de relaciones que pueden aprender, por lo tanto, a diferencia del PCA (Es un método matemático que se utiliza para reducir el número de variables de forma que pasemos a tener el mínimo número de nuevas variables y que representen a todas las antiguas variables de la forma más representativa posible.) La reducción de dimensionalidad puede incluir relaciones no lineales. La desventaja es su alto riesgo de sobreentrenamiento (overfitting), por lo que se recomienda emplear muy pocas épocas y siempre evaluar la evolución del error con un conjunto de validación.

Para mejorar el auto-encoder:

Tenemos control total sobre la arquitectura del autoencoder. Podemos hacerlo muy potente aumentando el número de capas, nodos por capa y lo más importante el tamaño del código. El aumento de estos hiperparámetros permitirá al autoencoder aprender codificaciones más complejas. Pero debemos tener cuidado de no hacerlo demasiado poderoso. De lo contrario, el autoencoder simplemente aprenderá a copiar sus entradas a la salida, sin aprender ninguna representación significativa. Simplemente imita la función de identidad. El autoencoder reconstruirá los datos de entrenamiento perfectamente, pero estará sobreajustado sin poder generalizar a nuevas instancias, que no es lo que queremos. Esta es la razón por la que preferimos una arquitectura "sandwich" y deliberadamente mantenemos el tamaño del código pequeño. Dado que la capa de codificación tiene una dimensionalidad más baja que los datos de entrada, se dice que el autoencoder está subcompletado. No podrá copiar directamente sus entradas a la salida y se verá obligado a aprender características inteligentes. Si los datos de entrada tienen un patrón, por ejemplo el dígito "1" generalmente contiene una línea algo recta y el dígito "0" es circular, aprenderá este hecho y lo codificará en una forma más compacta. Si los datos de entrada eran completamente aleatorios sin ninguna correlación o dependencia interna, entonces un autoencoder subcompletado no podrá recuperarlos perfectamente. Pero por suerte en el mundo real hay mucha dependencia.

Integrantes:

Angel Paillalef Silva.

Ignacio Durán Vasquez.