

AUTOENCODER: REDUÇÃO DE DIMENSIONALIDADE DE DADOS

Caio Ueno

743516

Claudia Sanches

743521

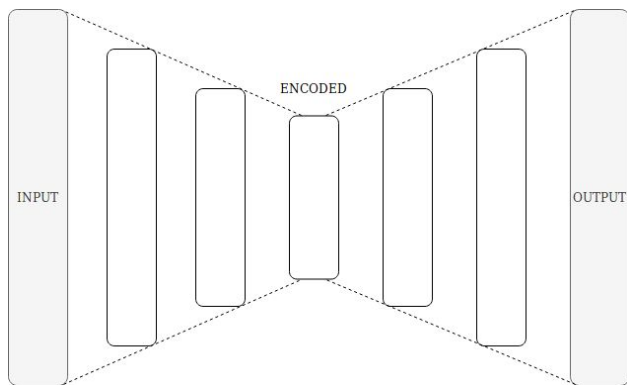
INTRODUÇÃO

Dados com alta complexidade, vetor de características com muitos valores, são comuns nas aplicações atuais.

Mas nem sempre muita informação é necessária, ou manipulável. Normalmente as observações, os dados, trazem consigo ruído, que em muitos casos não contribuem com a tarefa a ser desempenhada.

AUTOENCODER

- Rede neural que reconstrói os dados de entrada na camada final.
- Entretanto, reduz a dimensionalidade (quantidade de neurônios) e depois volta ao tamanho original.
- Encoder: primeira metade da rede, recebe o dado e reduz para uma representação menor e enxuta.



BASE DE DADOS PARA OS EXPERIMENTOS

De forma a observar a representação - mantém uma similaridade com as imagens originais? - foram utilizadas imagens.

Dois datasets foram utilizados:

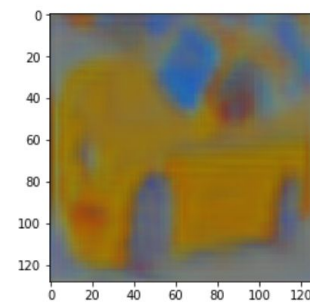
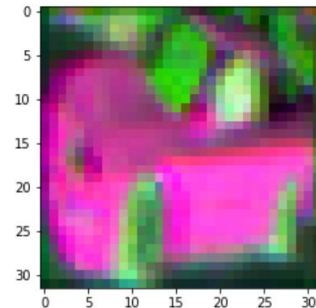
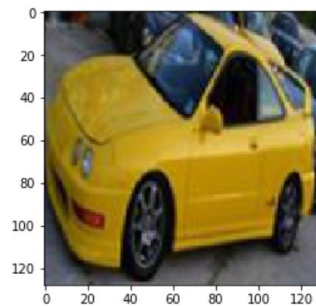
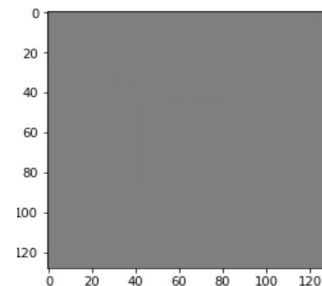
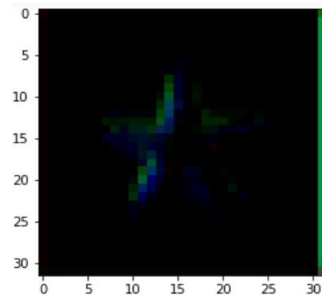
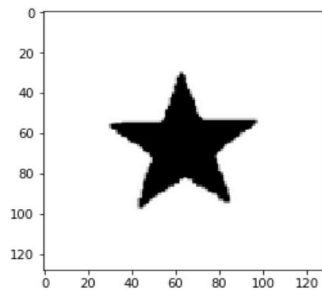
1. Conjunto de figuras geométricas simples - 4 classes.
2. Conjunto de fotos mais elaboradas - 8 classes.

CLASSIFICADORES

Para que fosse possível fazer uma comparação justa entre o classificador das imagens originais e o das representações, foram implementados dois classificadores com arquiteturas similares, exceto pelo tamanho da entrada.



REPRESENTAÇÕES OBTIDAS



EFICIÊNCIA E EFICÁCIA DA REPRESENTAÇÃO

Imagens originais:

- Treino do classificador mais lento;
- Acurácia maior;
- Mais memória.

Representações das imagens:

- Treinar o autoencoder - processo demorado;
- Treino do classificador mais rápido;
- Acurácia ligeiramente menor, dependendo da complexidade das imagens.

RESULTADOS

Acurácia e tempo de execução dos classificadores em cada *dataset*.

Classificador	D1	D2
Original	0.8310	0.9987
Encoded	0.7584	0.9732
Perda de acurácia	0.0726	0.0255

Classificador	D1	D2
Original	24m46s	20m25s
Encoded	6m3s	1m50s
Autoencoder	1h18m	19m11s
Diferença de tempo	18m43s	18m35s

CONCLUSÕES

Usar ou não Autoencoder? Depende do que é possível abrir mão:

- ocupa menos memória;
- maior velocidade de treinamento;
- a representação é um reflexo do dado original;
- reduz a complexidade do classificador.

REFERÊNCIAS

- Ana Caroline Gomes Vargas, Aline Paes e Cristina Nader Vasconcelos (2016). Um estudo sobre redes neurais convolucionais e sua aplicação em detecção de pedestres. *Proceedings of the XXIX Conference on Graphics, Patterns and Images, pages 1-4*.
- François Chollet (2016). Building Autoencoders in Keras. Disponível em: <https://blog.keras.io/building-autoencoders-in-keras.html>.
- Aditya Sharma (2018). Implementing Autoencoders in Keras: Tutorial. Disponível em: <https://www.datacamp.com/community/tutorials/autoencoder-keras-tutorial>.