VSB TECHNICAL UNIVERSITY OF OSTRAVA | FACULTY OF ELECTRICAL ENGINEERING AND COMPUTER SCIENCE | DEPARTMENT OF COMPUTER SCIENCE

Algorithms for Big Data

Autoencoders

Jan Platoš

October 11, 2020

Department of Computer Science
Faculty of Electrical Engineering and Computer Science
VŠB - Technical University of Ostrava

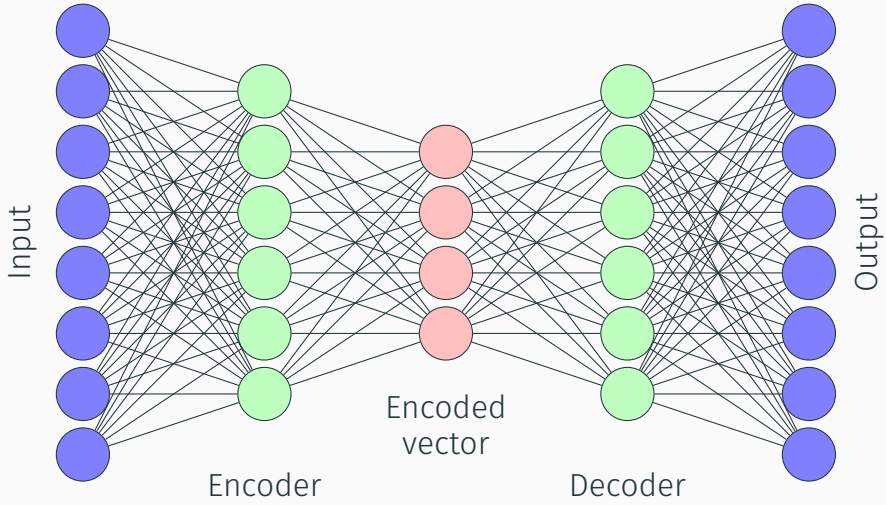
Table of contents

Autoencoders

Autoencoders

- Autoencoder is a neural networks that learn how to ”copy” its input to its output.
- The learned copy function may modify the image during the copy process.
- The middle layer contains reduced representation of the input.
- The layer need to learn how to maintain the main input image properties in the compressed space to be able to reproduce it in the output.

Autoencoders



Autoencoder

- The main goal of the autoencoder is to learn properly the output reconstruction.
- *Encoded vector* is defined in the form that is most siutable for this task.
- The weights learned focuses on this only, therefore, small *encoders* may not be well in classification.
- Autoencoder is an **unsupervised** model.

Autoencoder

- Autoencoder may be realised using Dense, Convolution, Recurrent or any other type or its combination.
- **Padding** is very important in Convolution autoencoders, because the decoder needs all the data. The option **same** maintain the original image size.
- *Encoder* and *Decoder* are usually symmetric but it is not a rule.

Autoencoder

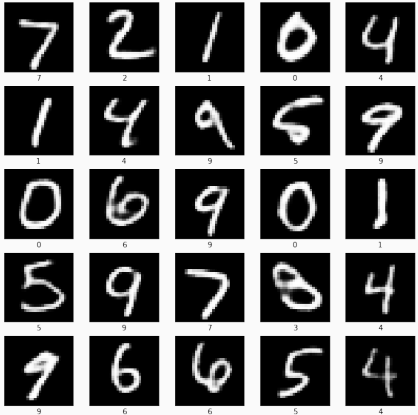
```
autoencoder = keras.Sequential([
    keras.layers.Conv2D(64, (3,3), activation='relu', padding='same',
        input_shape=(28,28,1)),
    keras.layers.MaxPooling2D((2, 2), padding='same'),
    keras.layers.Conv2D(32, (3,3), padding='same', activation='relu'),
    keras.layers.MaxPooling2D((2, 2), padding='same'),
    keras.layers.Conv2D(8, (3,3), padding='same', activation='relu'),
    keras.layers.MaxPooling2D((2, 2), padding='same'),
    # a 128 values of the minimized knowledge / features
    keras.layers.Conv2D(8, (3,3), padding='same', activation='relu'),
    keras.layers.UpSampling2D((2,2)),
    keras.layers.Conv2D(32, (3,3), padding='same', activation='relu'),
    keras.layers.UpSampling2D((2,2)),
    keras.layers.Conv2D(64, (3,3), activation='relu'),
    keras.layers.UpSampling2D((2,2)),
    # final output layer
    keras.layers.Conv2D(1, (3,3), activation='sigmoid', padding='same')
])
```

Autoencoder - Outlier analysis

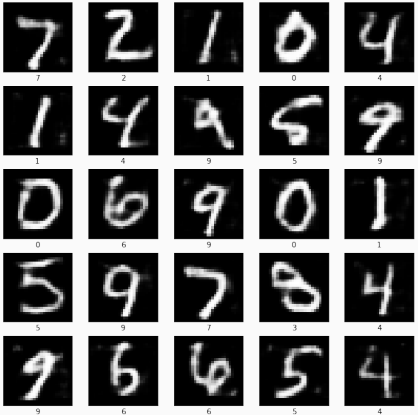
- Autoencoder is learned to produce original images.
- The difference between output and input is minimized during the training.
- Summary statistic about the output error on the test data may be collected.
- Such data may be used to identify outliers, i.e. images that differs from the testing data.
- The output on the outliers should be outside the expected error distribution of the real data.

Autoencoder - Outlier analysis

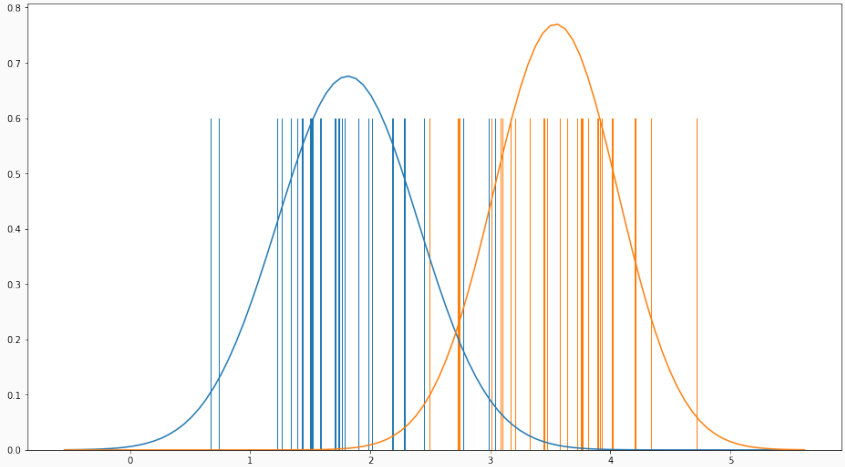
Test data



Modified data

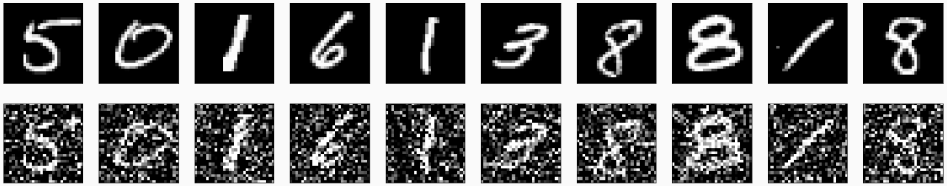


Autoencoder - Outlier analysis

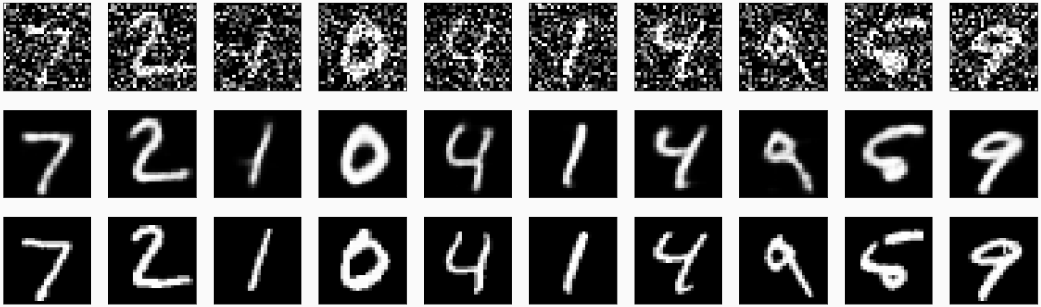


Autoencoder - Denoising autoencoder

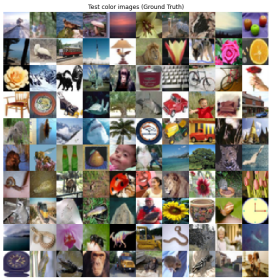
- The mapping function that transforms input into input may improve the image a bit.
- E.g. the reconstructed image may remove the noise from the input.
- In other words, we train the autoencoder to be resistant to small but finite-sized perturbations.

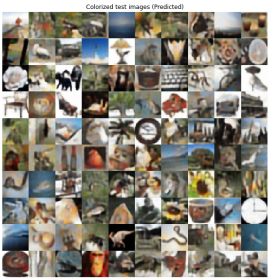


Autoencoder - Denoising autoencoder



Autoencoder - Image Coloring

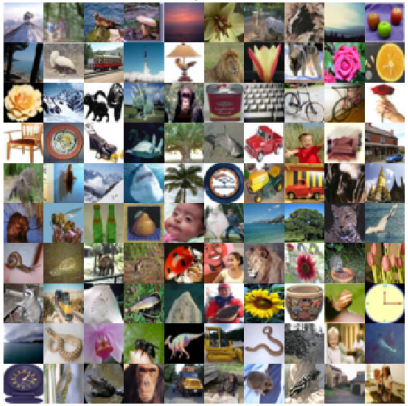




https://www.geeksforgeeks.org/colorization-autoencoders-using-keras/

Autoencoder - Image Coloring

Test color images (Ground Truth)



Colorized test images (Predicted)

Questions?