Bioinformatics LAB 10

GAN & Autoencoders in genomics

Prof.ssa Elisa Ficarra
Prof.ssa Santa Di Cataldo
Eng. Marta Lovino
Eng. Alessio Mascolini

Politecnico di Torino
DAUIN
Dept. of Control and Computer Engineering

LAB 10 - Objectives

Objectives

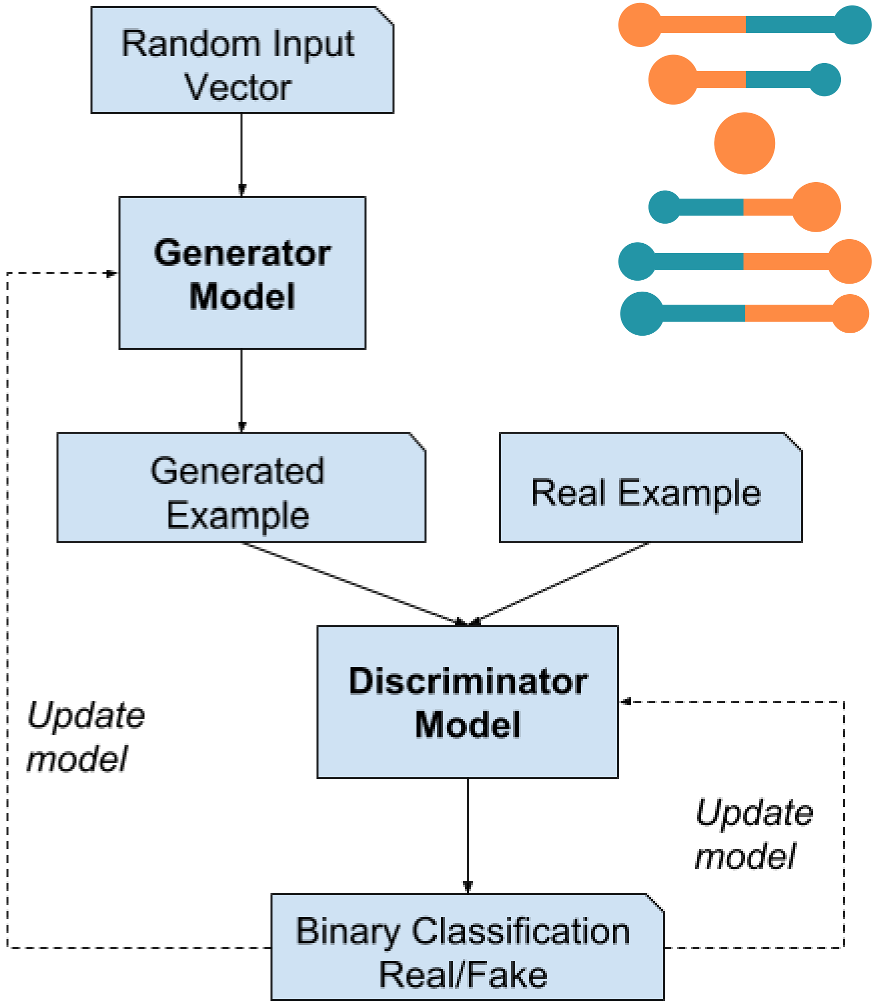
- Familiarize with GAN and Autoencoders
- Understand the main biological drawbacks

Adversarial Training

How can we know how similar the images made by our generator are to the original data?

Generative Adversarial Networks infer this metric using an heuristic, which is how difficult a second neural network is to deceive.

This approach works without paired data, which makes acquiring datasets significantly easier.
GANs represent the state of the art in image generation and image translation.



Random Input Vector

Generator Model

Generated Example

Real Example

Discriminator Model

Binary Classification Real/Fake

Update model

Update model

Models by subclassing

```
class MyModel(tf.keras.Model):

def __init__(self,...):
super(MyModel,self).__init__()
#Constructor code

def compile(self,...):
super(MyModel,self).compile()
#Code to run on compile

def train_step(self,data):
x,y = data
#Code for training
return {‘Metric name’: metricVariable}
```

Automatic differentiation

tf.keras.Model needs __init__ and compile
We can overload train_step to define custom training behaviour

tensorflow gives us the GradientTape to do automatic differentiation

Can calculate gradients for any tf.Variable, provided operations are differentiable

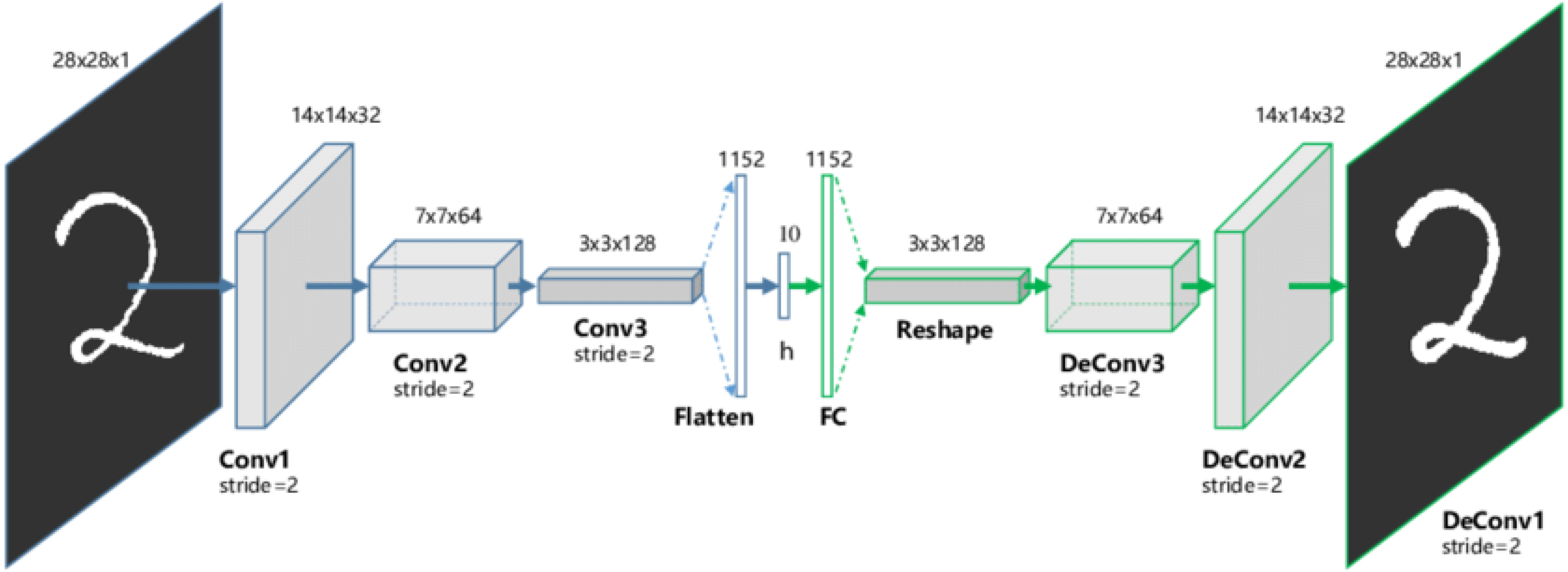
```
with tf.GradientTape() as tape:
    y_pred = self(x)
    loss = self.loss(y, y_pred)
gradients = tape.gradient(loss, self.trainable_variables)
self.optimizer.apply_gradients(zip(gradients, self.trainable_variables))
```

Autoencoders summary

Encoder-Decoder architecture

An Autoencoder encodes an image as a point in a latent space and decodes the point to the same image

A bottleneck ensures the network learns a compressed representation of the data



VAE summary

We assume a prior N(0,1)
The encoder predicts a posterior (μ and σ) instead of a point.

Then we sample a point from the distribution given by the encoder and feed it to the decoder.

Loss is: Reconstruction Loss + KLD from our prior

$$- \frac{1}{2}\sum_{j=1}^{J}\left(1+\log((\sigma_j)^2)-(\mu_j)^2-(\sigma_j)^2\right)$$

We can’t propagate gradients through the sampling operation, so we make our sampling deterministic by defining it as μ + εσ. ε is stochastic.

LAB 10 - Assignments

Assignment 1: Implement a GAN for sample generation

Download *dataset_2000samples.rar* from the Teaching Portal. After decompressing the file, you will have: mRNA.*txt*, *meth.txt*, *prot.txt* which contain respectively transcriptome, genome and proteome of 2000 samples divided in two classes. In *clusters.txt* you can find the label number for each sample. For this assignment you need only the *mRNA.txt* file.

Implement a GAN for each class in order to generate new samples starting from *mRNA.txt* file.

Implement an upper sampling technique using SMOTE.

Which are the differences between sample generation using GANs and SMOTE upper sampling?

Assignment 2: Implement a Variational Autoencoder for domain translation

Download *dataset_5000samples.rar* from the Teaching Portal. After decompressing the file, you will have: mRNA*.txt*, *meth.txt*, *prot.txt* which contain respectively transcriptome, genome and proteome of 2000 samples divided in 5 classes. In *clusters.txt* you can find the label number for each sample. For this assignment you need *mRNA.txt* and *meth.txt* files.

- Implement a Variational Autoencoder on mRNA samples
- Implement a Variational Autoencoder on meth samples
- Perform the domain translation using the encoder for mRNA and the decoder for math samples.
- Compare the new translated sample with the true ones.

Yang, Karren Dai, Anastasiya Belyaeva, Saradha Venkatachalapathy, Karthik Damodaran, Adityanarayanan Radhakrishnan, Abigail Katcoff, G. V. Shivashankar, e Caroline Uhler. «Multi-Domain Translation between Single-Cell Imaging and Sequencing Data Using Autoencoders». *BioRxiv*, 18 dicembre 2019, 2019.12.13.875922. https://doi.org/10.1101/2019.12.13.875922.

Project extension 1: Multi-omics GANs

Develop a multi-omic sample generation method using GANS. You have to look at the GANs for genomics in the literature and compare your method with at least an existing one.

- Which are the advantages of your approach?
- How much your approach is new?
- How much your approach is reliable?

Test your method on the real biological dataset from GDC data portal (details on that part will be provided in a lesson after Christmas break).

Project extension 2: Multi-omic Variational Autoencoder for domain translation

Develop a multi-omic domain translator using Variational Autoencoders. You have to look at the Variational autoencoder for genomics in the literature and compare your method with at least an existing one.

- Which are the advantages of your approach?
- How much your approach is new?
- How much your approach is reliable?

Test your method on the real biological dataset from GDC data portal (details on that part will be provided in a lesson after Christmas break).

LAB10 – Take home message

- Appropriately handling data spaces
- Set GAN and Autoencoders in the genomic context
- When is a GAN better than an autoencoder?
- Consider the biological implications of the computational choices

Questions?

Remember:
no question is
stupid