Adversarially Constrained Autoencoder Interpolation using Wasserstein Autoencoder

Machine Learning

Lorenzo Palloni

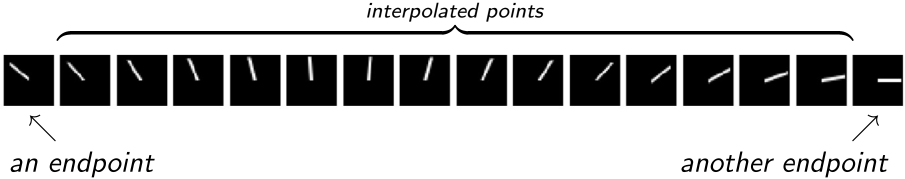
University of Florence

lorenzo.palloni@stud.unifi.it

April 20, 2020

Introduction

- **Unsupervised Learning** context.
- We aim to obtain "high-quality" **interpolations**.
- An interpolation example:



- An "high-quality" interpolation point have two characteristics:
  - is indistinguishable from real data;
  - represents a semantically smooth morphing between the endpoints.

Motivation and Techniques

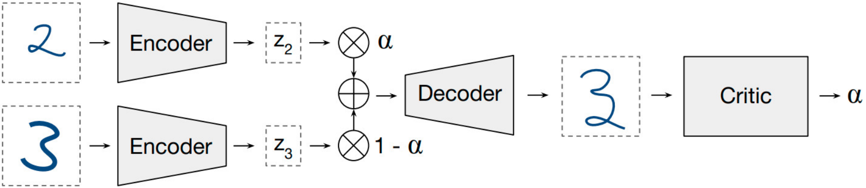
- Uncover underlying structure of dataset.
- Better representations $\rightarrow$ better results in other tasks.

- Implemented[1] frameworks:
  - Adversarially Constrained Autoencoder Interpolation (ACAI) [1];
  - Wasserstein Autoencoder (WAE) [2];

[1] All the models are implemented using pytorch [4]

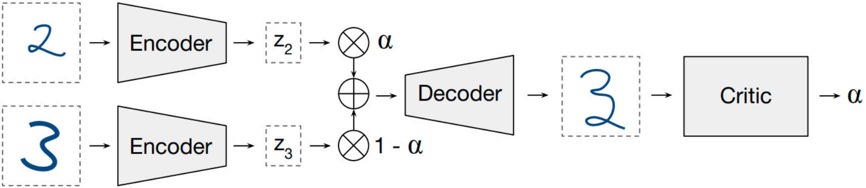
ACAI

- Graphical representation of ACAI structure:



ACAI

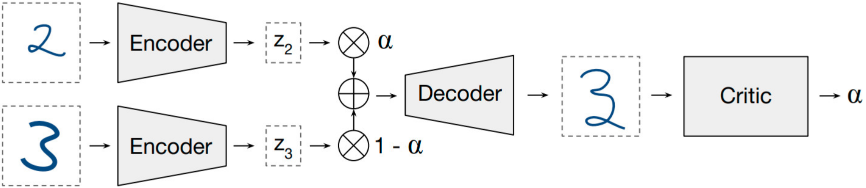
- Graphical representation of ACAI structure:



- Loss functions:

ACAI

- Graphical representation of ACAI structure:

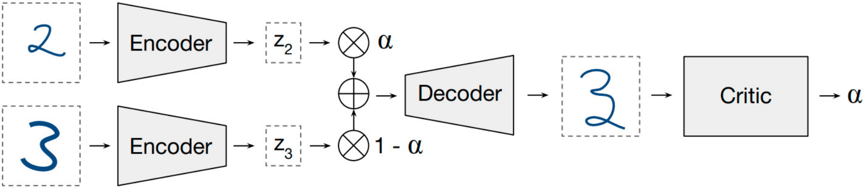


- Loss functions:

$$\mathcal{L}_d := \left\| d_\omega \left(\hat{x}_\alpha \right) - \alpha \right\|^2 + \left\| d_\omega \left(\gamma x + (1 - \gamma) g_\phi \left(f_\theta(x) \right) \right\|^2$$

ACAI

- Graphical representation of ACAI structure:



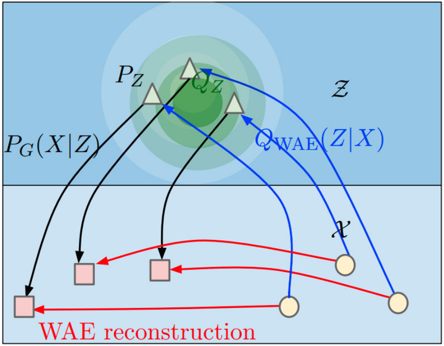
- Loss functions:

$$\mathcal{L}_d := \left\|d_\omega\left(\hat{x}_\alpha\right) - \alpha\right\|^2 + \left\|d_\omega\left(\gamma x + (1-\gamma) g_\phi\left(f_\theta(x)\right)\right\|^2$$

$$\mathcal{L}_{f,g} := \left\|x - g_\phi\left(f_\theta(x)\right)\right\|^2 + \lambda \cdot \left\|d_\omega\left(\hat{x}_\alpha\right)\right\|^2$$

WAE

- Graphical representation of WAE structure:



- Loss function:

$$D_{WAE}(P_X, P_G) := \inf_{Q(Z|X)\in\mathcal{Q}} \mathbb{E}_{P_X}\mathbb{E}_{Q(Z|X)}[c(X, G(Z))] + \lambda \cdot \mathcal{D}_Z(Q_Z, P_Z)$$

Different penalties, different WAEs (1/2)

- Setting

$$\mathcal{D}_Z(Q_Z, P_Z) = MMD_k(Q_Z, P_Z)$$

  $\rightarrow$ **we obtain WAE-MMD**.

- where $MMD_k(\cdot, \cdot)$ is the Maximum Mean Discrepancy [10]:

$$MMD_k(Q_Z, P_Z) := \left\| \int_{\mathcal{Z}} k(z, \cdot) dQ_Z(z) - \int_{\mathcal{Z}} k(z, \cdot) dP_Z(z) \right\|_{\mathcal{H}_k}$$

- where k is a positive-definite kernel k: $\mathcal{Z} \times \mathcal{Z} \rightarrow \mathcal{R}$.

Different penalties, different WAEs (1/2)

- Setting

$$\mathcal{D}_Z(Q_Z, P_Z) = MMD_k(Q_Z, P_Z)$$

$\rightarrow$ **we obtain WAE-MMD**.

- where $MMD_k(\cdot, \cdot)$ is the Maximum Mean Discrepancy [10]:

$$MMD_k(Q_Z, P_Z) := \left\| \int_{\mathcal{Z}} k(z, \cdot) dQ_Z(z) - \int_{\mathcal{Z}} k(z, \cdot) dP_Z(z) \right\|_{\mathcal{H}_k}$$

- where k is a positive-definite kernel k: $\mathcal{Z} \times \mathcal{Z} \rightarrow \mathcal{R}$.
- Kernel used in the WAE paper:

Different penalties, different WAEs (1/2)

- Setting

$$\mathcal{D}_Z(Q_Z, P_Z) = MMD_k(Q_Z, P_Z)$$

  $\rightarrow$ **we obtain WAE-MMD**.

- where $MMD_k(\cdot, \cdot)$ is the Maximum Mean Discrepancy [10]:

$$MMD_k(Q_Z, P_Z) := \left\| \int_{\mathcal{Z}} k(z, \cdot) dQ_Z(z) - \int_{\mathcal{Z}} k(z, \cdot) dP_Z(z) \right\|_{\mathcal{H}_k}$$

- where k is a positive-definite kernel k: $\mathcal{Z} \times \mathcal{Z} \rightarrow \mathcal{R}$.
- Kernel used in the WAE paper:
  - Radial Basis Function: $k(x, y) = \exp(-\gamma \|x - y\|^2)$

Different penalties, different WAEs (1/2)

- Setting

$$\mathcal{D}_Z(Q_Z, P_Z) = MMD_k(Q_Z, P_Z)$$

$\rightarrow$ **we obtain WAE-MMD**.

- where $MMD_k(\cdot, \cdot)$ is the Maximum Mean Discrepancy [10]:

$$MMD_k(Q_Z, P_Z) := \left\| \int_{\mathcal{Z}} k(z, \cdot) dQ_Z(z) - \int_{\mathcal{Z}} k(z, \cdot) dP_Z(z) \right\|_{\mathcal{H}_k}$$

- where k is a positive-definite kernel k: $\mathcal{Z} \times \mathcal{Z} \rightarrow \mathcal{R}$.
- Kernel used in the WAE paper:
  - Radial Basis Function: $k(x, y) = \exp(-\gamma \left\| x - y \right\|^2)$
  - Inverse Multiquadratic: $k(x, y) = \frac{C}{C + \|x - y\|^2}$, $C \in \mathcal{R}$

Different penalties, different WAEs (2/2)

- Setting

$$\mathcal{D}_Z(Q_Z, P_Z) = \underbrace{\|\mu_{P_Z} - \mu_{Q_Z}\|^2 + \mathrm{Tr}\left(\Sigma_{P_Z} + \Sigma_{Q_Z} - 2(\Sigma_{P_Z}\Sigma_{Q_Z})^{\frac{1}{2}}\right)}_{\text{2-Wasserstein distance between two multivariate normal distributions [6]}}$$

- and assuming the prior P_Z normal distributed,

  $\rightarrow$ **we obtain WWAE** (Wasserstein-Wasserstein Autoencoder) [3].

Roadmap of hands-on approach

- We aim to achieve "high-quality" interpolations on two datasets:
  - MNIST (images)
  - NESMDB [12] (music)

Roadmap of hands-on approach

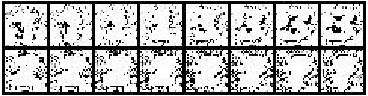
- We aim to achieve "high-quality" interpolations on two datasets:
  - MNIST (images)
  - NESMDB [12] (music)

1. ACAI + WAE applied on MNIST $\rightarrow$ poor results:

Roadmap of hands-on approach

- We aim to achieve "high-quality" interpolations on two datasets:
    - MNIST (images)
    - NESMDB [12] (music)

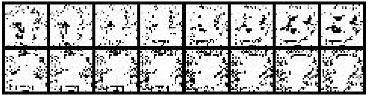
1. ACAI + WAE applied on MNIST → poor results:



Roadmap of hands-on approach

- We aim to achieve "high-quality" interpolations on two datasets:
  - MNIST (images)
  - NESMDB [12] (music)

1. ACAI + WAE applied on MNIST → poor results:

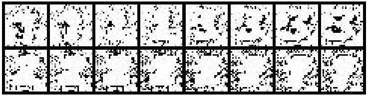


2. ACAI + WAE applied on NESMDB → poor results too.

Roadmap of hands-on approach

- We aim to achieve "high-quality" interpolations on two datasets:
  - MNIST (images)
  - NESMDB [12] (music)

1. ACAI + WAE applied on MNIST → poor results:

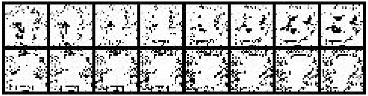


2. ACAI + WAE applied on NESMDB → poor results too.
3. ACAI + WWAE applied on MNIST → good results (next slide).

Roadmap of hands-on approach

- We aim to achieve "high-quality" interpolations on two datasets:
  - MNIST (images)
  - NESMDB [12] (music)

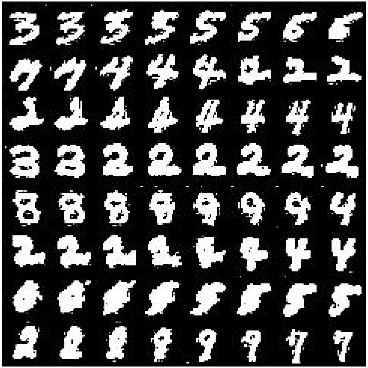
1. ACAI + WAE applied on MNIST $\rightarrow$ poor results:



2. ACAI + WAE applied on NESMDB $\rightarrow$ poor results too.
3. ACAI + WWAE applied on MNIST $\rightarrow$ good results (next slide).
4. ACAI + WWAE applied on NESMDB $\rightarrow$ requires too time to collect results.

Results on MNIST

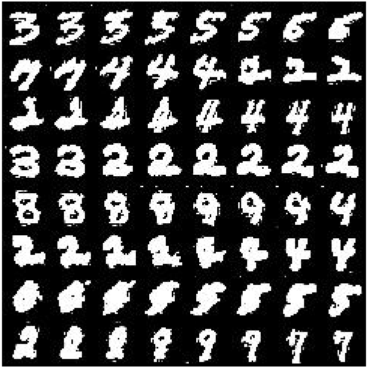
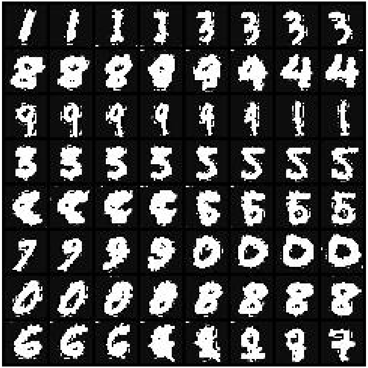
- Example interpolations on MNIST made with ACAI + WWAE:



- Each row is made by reconstructing (a sample) of the linear convex combination of the latent representation between the first and the last element of the row.

Results on MNIST

- Example interpolations on MNIST made with ACAI + WWAE:

- Each row is made by reconstructing (a sample) of the linear convex combination of the latent representation between the first and the last element of the row.

Final discussions

- We achieve pretty good quality interpolations on image data using ACAI + WWAE.
- All models discussed so far require:
  - high computational resources for training;
  - have the lack of an objective way to assess the performance.
- Raw audio in waveform probably needs:
  - another type of representation (e.g. MIDI) [7] [8];
  - another autoencoder structure (e.g. Wavenet Autoencoder [9])

References I

Berthelot, David, et al. "Understanding and improving interpolation in autoencoders via an adversarial regularizer." arXiv preprint arXiv:1807.07543 (2018).

Tolstikhin, Ilya, et al. "Wasserstein auto-encoders." arXiv preprint arXiv:1711.01558 (2017).

Zhang, Shunkang, et al. "Wasserstein-Wasserstein auto-encoders." arXiv preprint arXiv:1902.09323 (2019).

Paszke, Adam, et al. "Automatic differentiation in pytorch." (2017).

Goodfellow, Ian, et al. "Generative adversarial nets." Advances in neural information processing systems. 2014.

Dowson, D. C., and B. V. Landau. "The Fréchet distance between multivariate normal distributions." Journal of multivariate analysis 12.3 (1982): 450-455.

Borghuis, Tijn, et al. "Off the Beaten Track: Using Deep Learning to Interpolate Between Music Genres." arXiv preprint arXiv:1804.09808 (2018).

References II

- Borghuis, Tijn, et al. "Full-Band Music Genres Interpolations with Wasserstein Autoencoders." Ital-ia (2019).
- Engel, Jesse, et al. "Neural audio synthesis of musical notes with wavenet autoencoders." Proceedings of the 34th International Conference on Machine Learning-Volume 70. JMLR. org, 2017.
- Gretton, Arthur, et al. "A kernel two-sample test." Journal of Machine Learning Research 13.Mar (2012): 723-773.
- Radford, Alec, Luke Metz, and Soumith Chintala. "Unsupervised representation learning with deep convolutional generative adversarial networks." arXiv preprint arXiv:1511.06434 (2015).
- Donahue, Chris, Huanru Henry Mao, and Julian McAuley. "The NES music database: A multi-instrumental dataset with expressive performance attributes." arXiv preprint arXiv:1806.04278 (2018).

Appendix - Wasserstein distance

- Kantorovich's formulation of the *optimal transport* problem:

$$W_c(P_X, P_G) := \inf_{\Gamma \in \mathcal{P}(X \sim P_X, Y \sim P_G)} \mathbb{E}_{(X,Y)\sim\Gamma}[c(X, Y)]$$

- with some weak constrains it could be written as the following:

$$W_1(P_X, P_G) = \sup_{f \in \mathcal{F}_{L_1}} \mathbb{E}_{X \sim P_X}[f(X)] - \mathbb{E}_{Y \sim P_G}[f(Y)]$$

- Moreover defining P_G in two steps:
  1. $Z \sim P_Z$
  2. $G : \mathcal{Z} \rightarrow \mathcal{R}^d$
- we have:

$$\inf_{\Gamma \in \mathcal{P}(X \sim P_X, Y \sim P_G)} \mathbb{E}_{(X,Y)\sim\Gamma}[c(X, Y)] = \inf_{Q : Q_Z = P_Z} \mathbb{E}_{Q(X|Z)}[c(X, G(Z))]$$