Week 7

Progress during 10-16th Aug

Hao SUN

August 17, 2017

Contents

Progress in this week

- Segmentation task
  1. Tried residual connection to improve the reproduce accuracy
  2. Tried to use 2*2 strides to replace maxpooling
  3. Tried to use r band
  4. Summary and problem for this task
- Classification task
  1. On the difference between unsipervised/supervised learning (Obj function)
  2. On the number of hidden variables
  3. Going to the mathematical part
  4. An approximation
  5. Promising directions

Residual connection and use strides to replace pooling

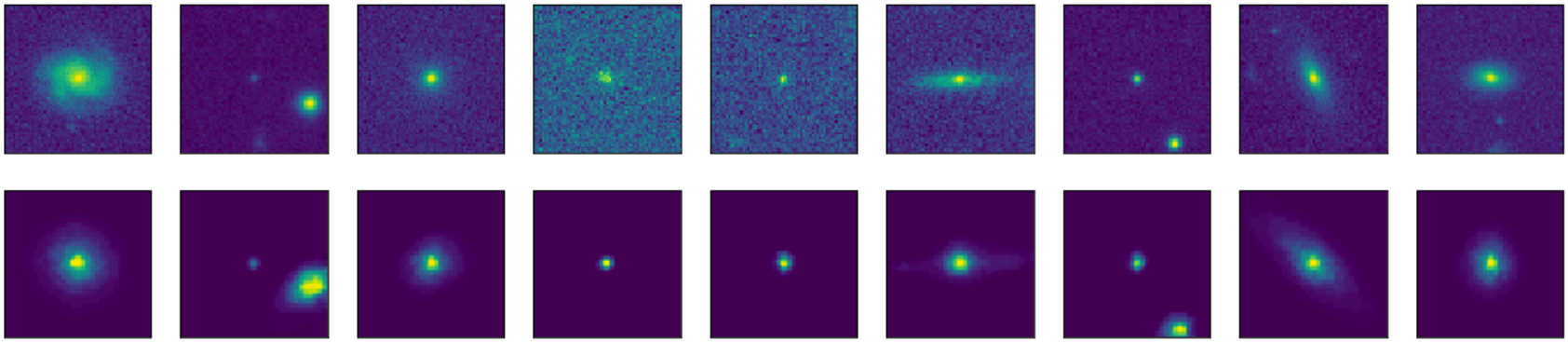


Figure: VAE with 30 hidden units

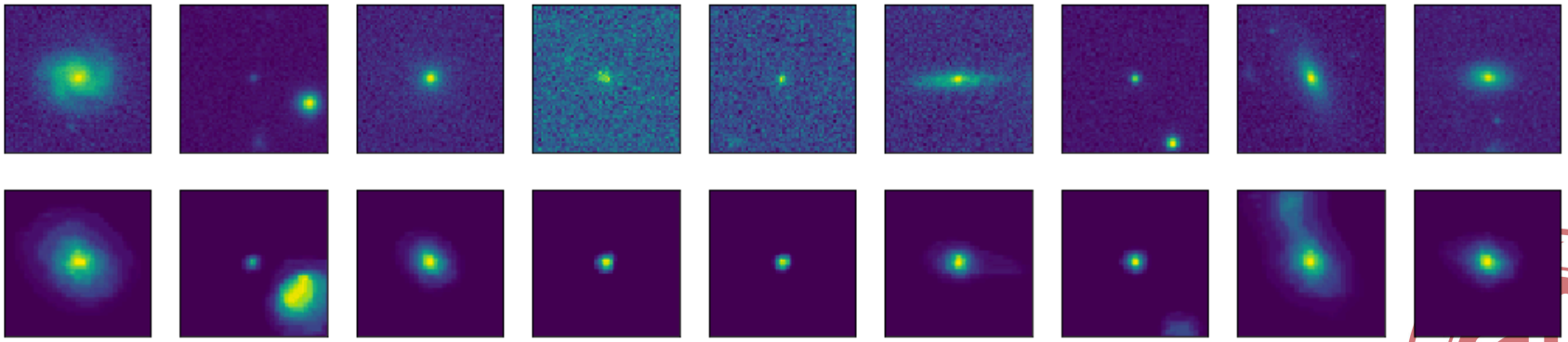


Figure: VAE with 100 hidden units

Residual connection and use strides to replace pooling

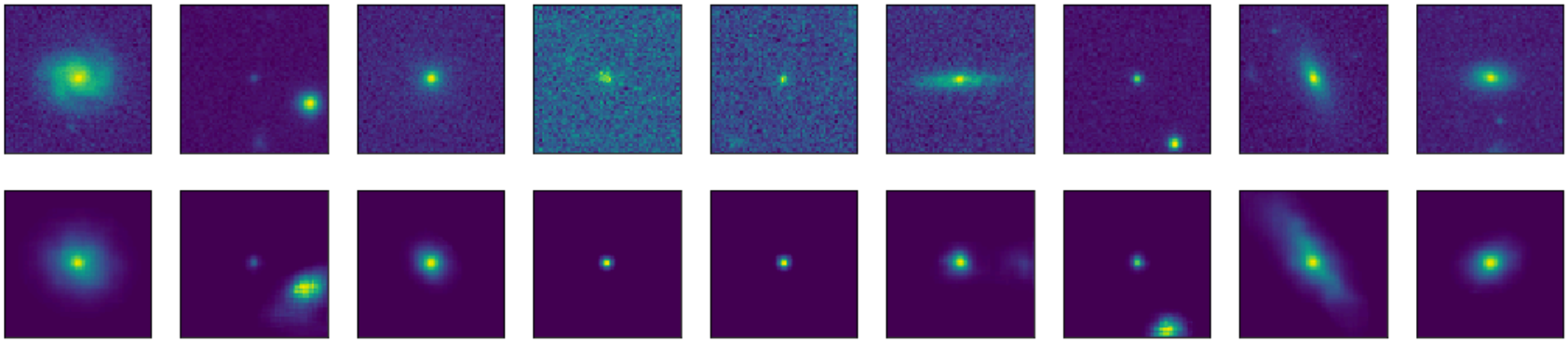


Figure: VAE; use 2*2 strides in first 2 down-sample layers

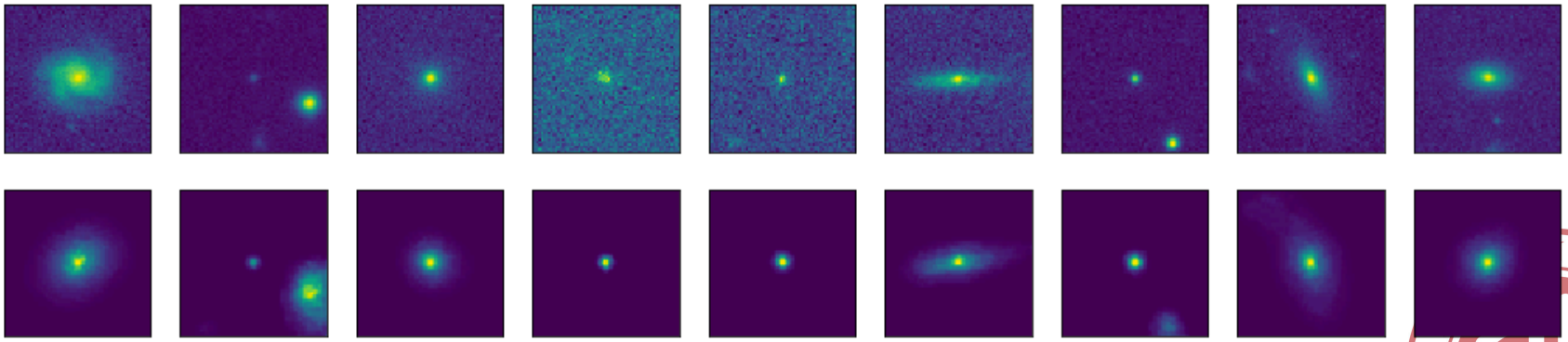


Figure: VAE; use 2*2 strides in all down-sample layers

Residual connection and use strides to replace pooling

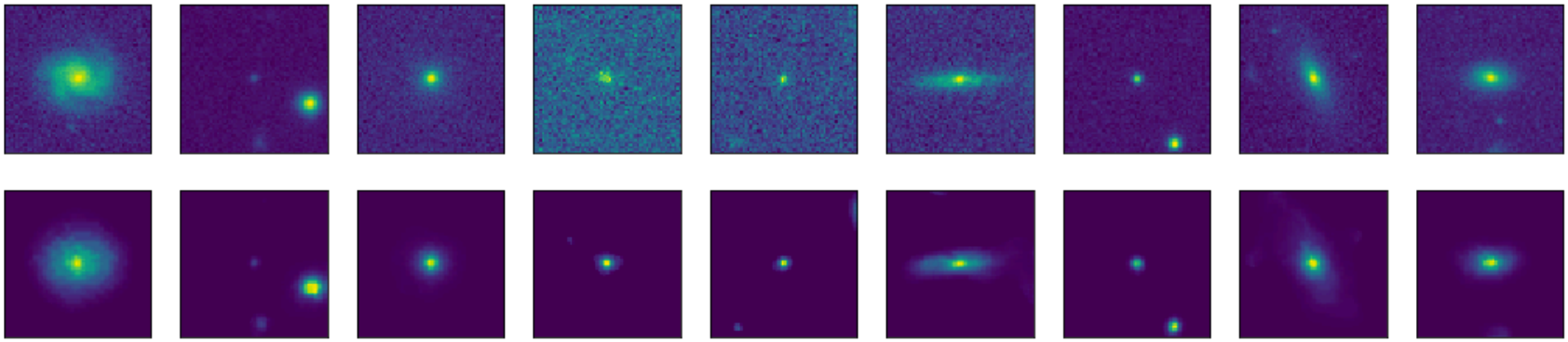


Figure: AE

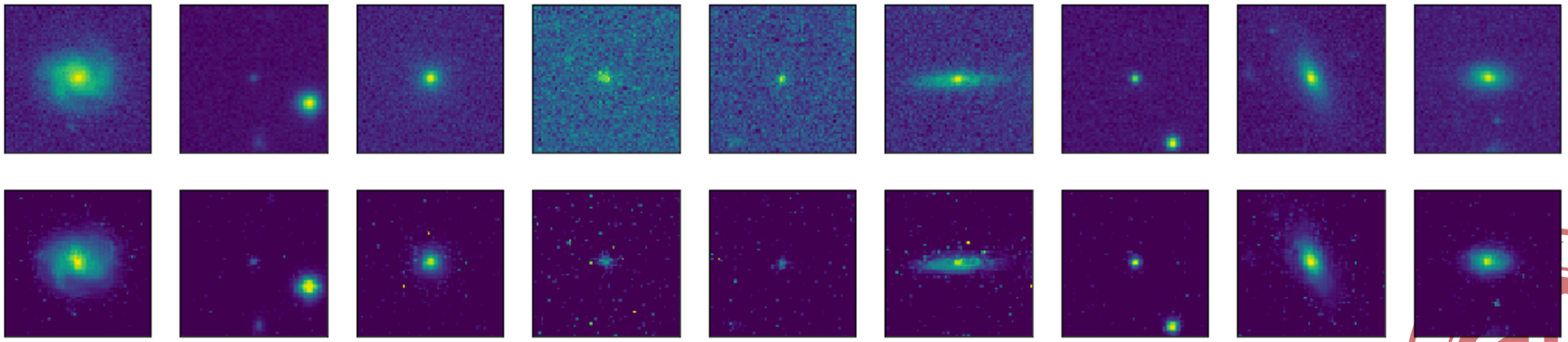


Figure: AE; Use residual connections

Residual connection and use strides to replace pooling

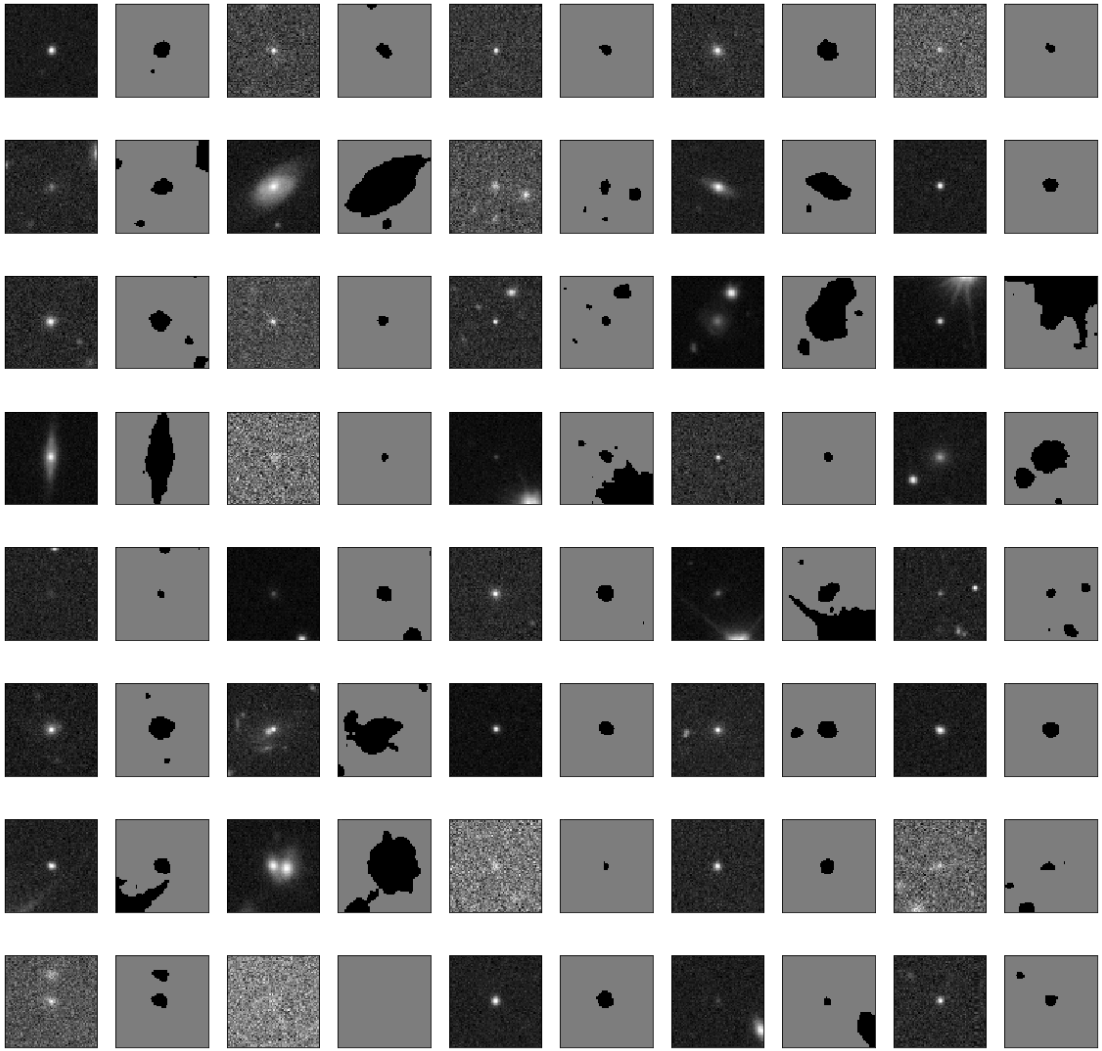


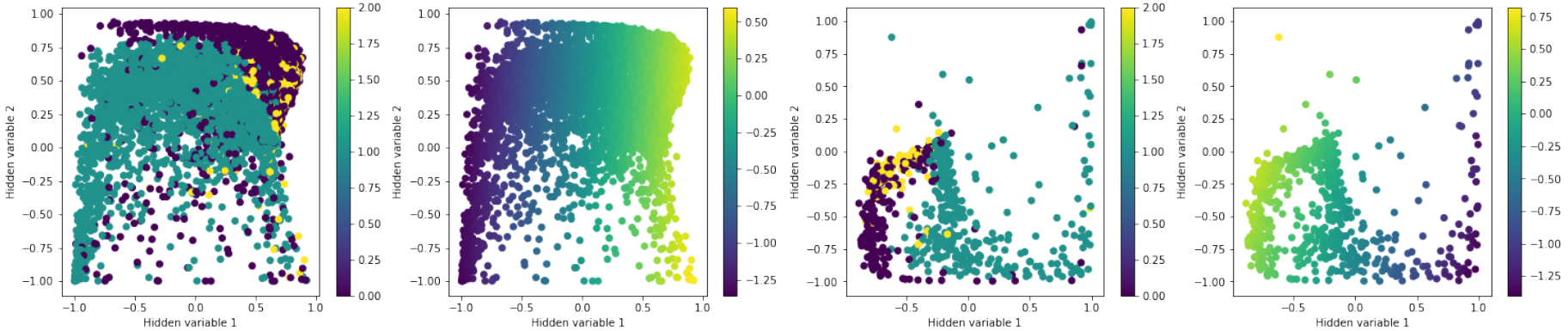
Figure: Segmentation; Use residual connections

Summary and problem for this task

1. TODOs:
   - Get better performance and only use AE (Sophie will focus on this task later)
2. Problems:
   - A trade-off between noises and faint stars
   - Standard or benchmark

On the number of hidden variables

In experiment, I found using more hidden varialbes(5 or more) always lead to worse result than using less (2 or 3).
Such result comes from the geometry structure of high dimensional hidden space.
The manifold clustering method I use is extremly sensitive to outliers and "tails"



In higher dimensional space, it's more likely to have more "tails", which lead to worse manifold learning result.
Here in the classification task, what we should do is to separate those clusters, or broaden the gap between different clusters for better performance in clustering algorithms.

On mathematical perspective

KL divergence:

$$D_{KL} = \int_{-\infty}^{\infty} P(x) \log \frac{P(x)}{Q(x)} dx$$

In traditional Variational Autoencoders, the priori distribution is always a normal distribution. So that we can generate a new image easily with an given point in the hidden space.

$$P(x) \sim \sum_{i}^{n} N(\mu_i, \sigma_i^2) \quad Q(x) \sim N(0, 1)$$

In this project, I want to map the images into two clusters in hidden space(stars and galaxies). I should use a double peak gaussian instead of a single peak one.

$$Q(x) \sim \frac{1}{2} N(-1, 1) + \frac{1}{2} N(1, 1)$$

An approximation

Before get analytic solutions (by working out a pretty complex integral problem), here is an approximation:

$$
\begin{aligned}
& D_{KL}\left(N(\mu,\sigma^2)\|\frac{1}{2}N(-1,1)+\frac{1}{2}N(1,1)\right) \\
=& D_{KL}\left(2*\frac{1}{2}N(\mu,\sigma^2)\|\frac{1}{2}N(-1,1)+\frac{1}{2}N(1,1)\right) \\
\leq& D_{KL}\left(\frac{1}{2}N(\mu,\sigma^2)\|\frac{1}{2}N(-1,1)\right)+D_{KL}\left(\frac{1}{2}N(\mu,\sigma^2)\|\frac{1}{2}N(1,1)\right) \\
=& -\frac{1}{2}\log 2\left(1+\log\sigma^2-\frac{1}{2}(\mu-1)^2-\frac{1}{2}(\mu+1)^2-\sigma^2\right)
\end{aligned}
$$

This approximation is quite rough. I'm trying to use Mathematica to solve this intigral. But I'm not sure if I can get analytic solution.

Defects of KL-divergence

Here are lots of defects of using KL-divergence: asymmetry(troubles are like figures below) [1], gradient disappearance.

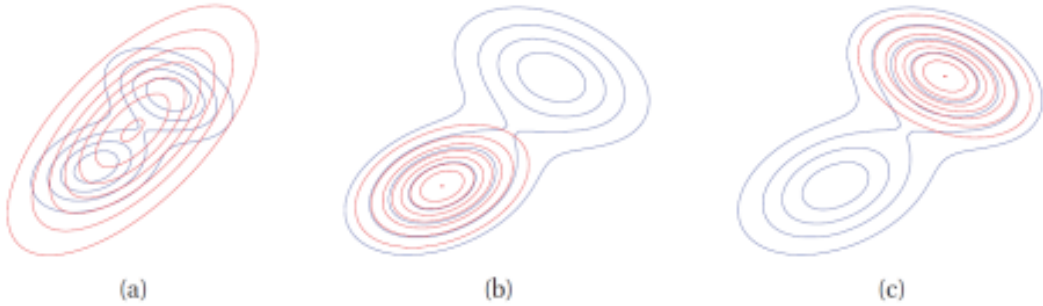


Figure 21.1 Illustrating forwards vs reverse KL on a bimodal distribution. The blue curves are the contours of the true distribution p. The red curves are the contours of the unimodal approximation q. (a) Minimizing forwards KL: q tends to "cover" p. (b-c) Minimizing reverse KL: q locks on to one of the two modes. Based on Figure 10.3 of (Bishop 2006b). Figure generated by `KLfwdReverseMixGauss`.

Figure: Optimize $D_{KL}(P, Q)$ or $D_{KL}(Q, P)$, with blue as P and red as Q

Wasserstein divergence works pretty good in WGAN, but I only have a conceptual understanding of it. There are some exciting similarity between VAE classification and GAN - both are unstable and sensitive to initial values.

[1] Machine Learning: A Probabilistic Perspective, p734

On the stability and sensitivity to initial values

When using KL-divergence:

1. N(0,1) priori, in normal VAEs. **Gradient exist**. Optimization is easy for we use normal initializers.
2. double-peak priori, in classification tasks. **Gradient disappear**. Optimization is hard, especially when those two peaks are far away from each other.

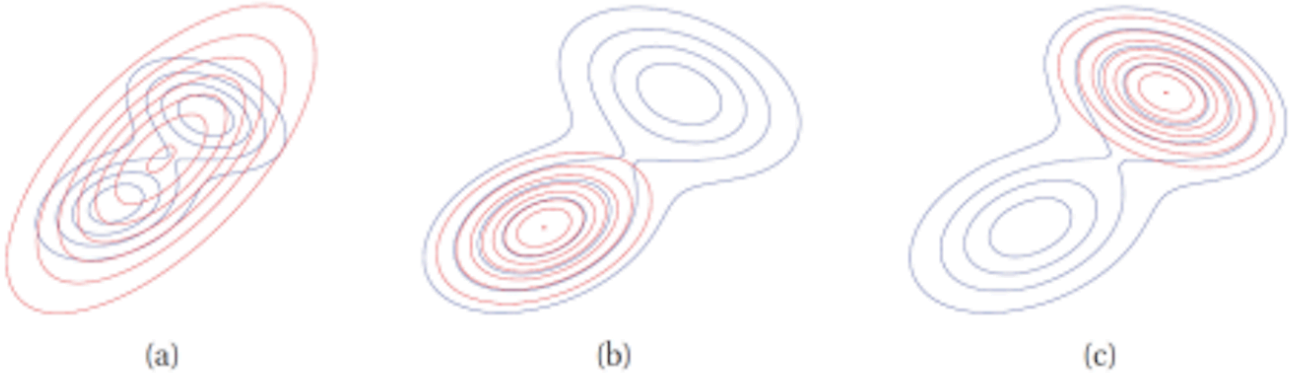


Figure: Optimize $D_{KL}(P, Q)$ or $D_{KL}(Q, P)$, with blue as P and red as Q

An naive W-loss VAE (designed for classification)

I created 2 potential wells in each hidden dimension with a naive L2 loss (L1 loss has uniformly large gradient, too strong). Only an analogy of W-loss.