IDL - Appeal, Exam B.

David Ponarovsky

August 2025

Question 7 - RNN nets. Does a recursive net of type Elman, that gets the zero vector as input at each step, can count? Namely, outputs the value t at its t-th step?

1. Yes
2. No
3. In general No, Yet when given t as initial input, yes.

My answer: (3), Correct answer: (1) .

I believe that the confusion arises from the order of entities. The question: 'Is there an Elman cell that can count until t for an arbitrary t?' is a different question than: 'Fix t, does there exist an Elman cell that can count until t?'

For the **second** question: There is a family of unbounded fan-in/out circuits, at width $poly(|t|)$ (the length of the encoding of t), that implement addition: [addition in AC_0]. It's not hard to see that the implementation in the notes can be realized using an Elman cell, and therefore one can find such a realization that adds 1 to the input which is entered via the hidden channel.

For the **first** question, in which we first fix the Elman cell architecture, and therefore also fix the width of the output. Thus, the number that can be outputted using the cell has to have a suitable length and cannot be arbitrary.

I marked the third option since saying 'on given t as initial input' implies that t is a valid input/token to the cell, namely, it has a length that matches the cell's weights. For such t's, the RNN counts correctly, whereas for others, the behavior of giving the input is not well defined and is expected to be a failure.

Question 8 - Inception Score. Which of the following scenarios is expected to yield high IS score, although the generated images are at low quality?

1. The model generates a blurred images, yet with high number of categorical.
2. The model generates clear elements that are easy to classify, Yet the elements (inside them) are unrealistic.
3. The model generates good images and then add them a random noise.

4. The model generates images with high variance between the outputs at the pixels level, but their sementic content repeats on itself.

Question 14 - VAEs. What is the reason for the generated images by VAEs been blurred compared to the images generated by GANs ?

1. Usage of reconstruction loss that smooth sharp items.
2. KL-divergence element that impair the disentangle (or separation) of different samples in the latent space.
3. Low presentation ability of the VAEs architecture.
4. Entering too mach noise into the latent space, which after decoding comes into fact in burred image.

My answer: (1), Correct answer: (2) . Denote the loss function by $\mathcal{L} = |A(E) - I| + KL(p(z)|q)$, where the first term is the reconstruction loss and the second is the KL-divergence, which penalizes the distance between the distribution induced over latent space and a baseline family of distributions q (Gaussians).

I agree that, in general, the main reason for the results of the VEAs being blurred is the KL-divergence term, In particular it enforces the decoder to decode a sampled super-position over the latent space. Yet the question asks what is the main reason for blurriness compared to GANs.

When comparing to GANs, which, in our course, have an amorphic architecture and can be arbitrarily complex, one should also consider the case when the architecture of the VEAs is as exactly complicated as in GANs, surly if in that regime the VEAs products are still inferior.

In that regime, one could think on a **decoder** which expands the latent dimension so mach, such that for human eye (Or more correctly to latent-space eye) the interpolation $D(tz_1 + (1-t)z_2)$ is not a continues function. In that case, even though that the KL-element enforce the latent vector z be distributed according gaussian, any value of z that can be seen in experiment leads to other value x in the data space.

For example, consider that the latent space has a width of k bits (For the sake of the exercise, you can imagine that the numbers at the latent space are in $(0, 1)$), and the decoder decode each z to a space represented by $n = 2^k$ bits, such that $D(z) = 2^{z*2^k}$ (shifting by the integer suitable to z). Clearly, close samples at the latent space gets far in the domain space. That argument cancel any justification due to behavior or panellization in the latent space.

That bring us to ask if there is difference in the case which is the latent space is trivial. Namely, when it's dimension is 0 (it's size is 1), namely a constant machine ('generates only cat, and always the same cat'). In that case optimal generator and discriminator in GANs would be the generator which output the same cat, and discriminator which guess at probability $\frac{1}{2}$, while in the VEA scheme any generator which outputs a noise version of the same cat, would penalize by only $\frac{\varepsilon}{\sqrt{n}}$ if the reconstraction error is l_2.

Second, if the KL-divergence element is indeded the main source for the blurration, we would expect that WAEs for which the loss over the latent-space heads to advance $\mathbf{E}_{\sim x}\left[\mathbf{Pr}\left[Z|X\right]\right] \rightarrow\sim e^{-z^2}$ instead of $\mathbf{Pr}\left[Z|X\right] \rightarrow\sim e^{-z^2}$ (for any x) would generates a significantly less blurred images. Indeed they exhibit improvements when compared to VEAs, yet they are still blurred compared to Gans.