Intuition Report 7

Ramtin Mojtahedi Saffari - 20307293

Github link: https://github.com/Ramtin-Mojtahedi/intuition_Report_6/blob/main/Intuition%20Report%206.pdf

Assessed link 1: Variational Autoencoder Interactive Demos with Deeplearn.js

Self-reflection

- **Short Summary**

The link provides a great interactive tool for exploring and experimenting with an example of a variational autoencoder using digits of [0-9]. The demo offers three different interactive services, including working with ten variants that are represented Z (latent variables), clicking on ten piles of numbers and randomly generating ten pairs of σ and μ, and gives the possibility of drawing a number by ourselves and then generate ten teams σ and μ to be imported to the decoder section and reconstruct the image.

- **Hypothesis and Expectation**

I hypothesized that the mean of all assigned Z values to each number decreases. In contrast, we reduce the number, whereas the standard deviation of the Z values for the number increases while we increase the numbers. Also, the generated random Z values for a specific number have a relative standard deviation compared to the other generated numbers.

- **What I Achieved and Learned**

As I learned, the latent space(Z) represents compressed data in which related data points are closer in space. Latent space may be used to learn data attributes and to develop simpler data representations for analysis. In addition, by examining data in the latent space, whether by manifolds, clustering, or other methods, we may uncover patterns or structural similarities between data points. We may use our model's decoder to 'create' data samples by interpolating data in the latent space [1].

Towards my experiments, I experiment with the achieved Z set of values for the number and calculate the mean and standard deviation values for each set. For the number "0," the mean and standard deviation values achieved as -0.02 and 1.14, respectively. However, for the number "9," they achieved 0.04 and 0.75 for the mean and standard deviation, respectively. The same trend is detected for the numbers between zero and nine. As another example, for the number 8, it is achieved as 0.36 and 0.95, respectively, while for number -0.84 and 0.69 for the mean and standard values, respectively. In addition, I tried specific numbers several times to understand their pattern and understand that a particular number has a relative standard deviation among those randomly generated Z values in each trial. For example, I tried number "0" three times and got the standard deviation results of [1.09, 1.13, 1.2]. However, the results for the number "9" were [0.72, 0.77, 0.79]. This shows that although the Z values are generated randomly, there is a distinction between the range f the generated values for each specific number. This distinction and patterns can be used for determining the numbers. As mentioned earlier, the Z values are a representation f the input data. This means that the more distinction between the standard deviation of the set of Z values, the better and more specific results of the reconstructed number can be achieved in the output. This finding aligns with the fact that Z is the latent space formed by multiplying the encoder weights by the input and passing through the function. Then the z is multiplied with decoder weights in the decoder to produce the reconstructed picture and calculate the reconstruction loss values [2].

Suggestion and Filling the Gaps

I found the interactive link very interesting and helpful in learning how a generative text model works. It is recommended to add an in-detail explanation for the Z value and how it was achieved and used from reference [3]. This brief description is really helpful to improve users' understanding of the rationale behind the toolbox.

Assessed link 2: Digit Fantasies by a Deep Generative Model

Self-reflection

- **Short Summary**

The provided link is an interactive tool for generating digits from zero to nine using different values of Z that are given in twelve sliding controller bars to visualize the outcome of the generated numbers. Also, the toolbox provides a dream mode, which changes the Z values to make fancy generated digits randomly.

- **Hypothesis**

I hypothesized that by decreasing the Z values, the generated digits become more thick and bold. Also, the values around the middle make the numbers look perpendicular. However, as the Z values go towards the low or high boundaries, the generated digits got oriented to the left or right.

- **Testing the hypothesize and what I have learned**

As mentioned in the first part, Z values represent the compressed data, and making a tradeoff between its values is significant in generating decent results. I tried a different set of values to understand the effect of Z values in the generated digits. As Z values go to the lower boundary, the results become bolder, and the line of numbers gets thick. However, increasing Z values to the higher boundary makes them look thinner. Also, it seems the higher Z values are responsible for the edge sides of the digits, and the lower values are responsible for the base structure of the digits. This should be more distinguished by comparing the standard deviations as I experimented with in the previous trial. In addition, increasing the Z values towards the high boundary or decreasing it to the lower boundary makes the digits oriented, and this is important for the digits that there has less deviation like “1” compared to a number that has more deviation like “7”. Therefore, always there should be a tradeoff to generate the best possible digit number.

As we know, the average probability is then used as an anomaly score, which is called the reconstruction probability. Here digits with high reconstruction probability are classified as anomalies, and the major benefit of VAEs compared to traditional AE is the use of probabilities to detect anomalies. In this way, we can even enhance the performance of the VAEs as recommended in the reference [4]. As the authors proposed, in addition to enforcing the deep feature consistency principle, which ensures that the VAE output and its related input pictures have equivalent deep features, they used a generative adversarial training technique to drive the VAE to create realistic and natural images or digits. Also, effective strategies can be used to extract effective features used for image representation. All of these methods can be used to improve the performance of the generated digits [4].

Suggestion and Filling the Gaps

Similar to what I mentioned before, I found the interactive link very interesting and helpful in learning-by-experiment with different given sentences. It is recommended to explain the information in the toolbox and how the dream option works. Also, provide some details about the Z values, such as using the information in the reference [5].

Self-evaluation:

In this intuition report, I have gone through an in-depth analysis of two of the provided links. In my assessment, I have completely considered the required expectation for deep exploration, including proposing a hypothesis and what I expected, reporting on what I achieved and explored, in-depth discussion of what I have learned, and providing gaps and recommendations to fill them. Considering the quality and assessment level, I deserve to get the full mark (4 points) for this intuition report.

In advance, thank you very much for your time and consideration of this report.

Best regards,

Ramtin

References

[1] Jordan, J. (2018, July 16). *Variational autoencoders.* Jeremy Jordan. https://www.jeremyjordan.me/variational-autoencoders/

[2] Ijaz, E. (2021, July 9). *Facial Image Reconstruction using Autoencoders in Keras*. Medium. https://medium.com/geekculture/face-image-reconstruction-using-autoencoders-in-keras-69a35cde01b0

[3] Stewart, M. P. R. (2020, July 30). *Comprehensive Introduction to Autoencoders - Towards Data Science*. Medium. https://towardsdatascience.com/generating-images-with-autoencoders-77fd3a8dd368

[4] Hou, X., Sun, K., Shen, L., & Qiu, G. (2019). Improving variational autoencoder with deep feature consistent and generative adversarial training. *Neurocomputing*, *341*, 183–194. https://doi.org/10.1016/j.neucom.2019.03.013

[5] Tiu, E. (2021, January 8). *Understanding Latent Space in Machine Learning - Towards Data Science*. Medium.

https://towardsdatascience.com/understanding-latent-space-in-machine-learning-de5a7c687d8d#:%7E:text=The%20latent%20space%20is%20simply,representations%20of%20data%20for%20analysis.