Cross Validated

Variational Autoencoder and validation loss

Asked 2 years, 2 months ago Active 2 years, 2 months ago Viewed 2k times

3

2

I'm experimenting with Keras (Tensorflow backend) and Variational Autoencoder. My loss function is the classical one, i.e, the sum of reconstruction error and KL divergence.

When I provide my model with validation data, Tensorboard shows a 'validation_loss', computed on validation data. I'm wondering what the correct form of 'loss' function on validation data should be; I think it should only contain the reconstruction error, not the KL part: is it correct?

In other words, what would be the meaning of KL divergence on validation data?

deep-learning validation loss-functions autoencoders

asked Aug 29 '17 at 8:41

user38320

111 10

2 Answers

3

✓

The short answer is: Don't drop the KL term.

The reconstruction error plus KL term optimized by a VAE is a lower bound on the log-likelihood (also called the "evidence lower bound", or ELBO) [1]. Log-likelihood is one way to measure how well your model explains the data. If that's what you're after, it makes sense to try to evaluate the log-likelihood. This is not straight-forward, but possible [2].

You can use the ELBO as a conservative estimate of the log-likelihood. It therefore makes sense to use reconstruction error plus KL term as your `validation_loss` .

Ask yourself *why* you are training a variational autoencoder (VAE). If you can answer this question, the right way to evaluate (and train) your model will become much clearer. Is the reconstruction error important for your application? If yes, then reconstruction error would be okay to use for validation, but then I would question why you are optimizing for log-likelihood.

[1] Kingma & Welling, Auto-Encoding Variational Bayes, 2014 (Equation 2)

[2] Wu et al., On the Quantitative Analysis of Decoder Based Generative Models, 2017

4,632 18 32

Thanks for the reply and the references! So, I can use the same loss function on training and validation data. Moreover, if I suppose that both P(z) and Q(z|X) are Gaussian, I can use close form of equation (10) in [1] to compute loss function (the first addendum should be the D_KL(Q(z|X) || P(z|X))). Is it correct? My doubts were related to the fact that I thought to the KL part as a regularization term and, usually, regularization terms are cut off in test phase (e.g., dropout). – user38320 Aug 31 '17 at 22:37

You can use the closed form for the KL term, correct. – Lucas Sep 11 '17 at 18:11

1

If you ignore the regularization part (KL divergence) you will not be able to compare it with the train loss. It is true that regularization is added to better optimize the parameters of the model and not for better approximation of the loss function. You could add 'mse' metric for this,

```
Model.compile(...,metric=['mse'])
```

You can have look at this keras issues page for more detailed discussion.

answered Aug 31 '17 at 8:30

curio1729

121 1 5