General Flow

The latest version of the code uses **Wasserstein GANs** (`https://arxiv.org/abs/1701.07875`). The main differences between the WGANs and the standard (Vanilla) GANs are, WGANs : (i) use critic instead of Discriminator (just for convention); (ii) label real samples (-1) and fake samples (1) (instead of 1 nd 0); and (iii) use linear activation in the last layer (instead of sigmoid).

for number of training iterations **do** (see here):

Switch ON the training of the critic

// The first part trains the critic so make sure its training is set to TRUE

for a number of intermediate steps **do** (see here):

// Train the critic multiple times in one run.

- Draw samples of half-batch size from the PRIOR // samples of shape (half-batch, x-grid-size)
- Generate an array of (-1) as the target probability // shape (half-batch,)
- Train the critic on the PRIOR/TRUE samples
- Use the generator to generate/predict FAKE samples of a half-batch size // samples of shape (half-batch, x-grid-size)
- Associate the FAKE samples to an array of (1) // shape (half-batch,)
- Train the critic on the FAKE samples

end for

// As training progresses, this training procedure should allow the critic to distinguish between the TRUE and FAKE samples.

Freeze the critic

// Make sure that the critic is not trained (improving) when the generator is learning to reproduce the PRIOR samples. This procedure uses the previous trained status of the critic and engages it in an adversarial training with the generator.

for certain number of intermediate steps **do** (see here):

// As for the critic, this allows the generator to be trained multiple times during one run. The number of steps does not have to be the same as for the critic. For certain scenario, giving more training to the generator improves the performance of the model. (But is it always the case? And how to assess this apart from doing an hyperopt?)

- Generate FAKE samples of a batch size using the generator // shape (batch, x-grid-size)
- Associate the results to an array of (-1) // shape (batch,)
- Validate the results with the critic see here // outputs (-1) for TRUE and (1) for FAKE

end for

// As training progresses, this procedure should allow the generator to learn generating FAKE samples by improving through then critic's validation.

end for

Note: The model is a 1-dimensional one as he input PDF is formatted in this way. And the information concerning the x-grid is given to the generator here using a custom layer.

Side note: There is a proof of concept notebook (a concatenation of the whole) for quick testing.