Supplementary Material: Max-margin learning with the Bayes factor

Rahul G. Krishnan
MIT

Arjun Khandelwal
MIT

Rajesh Ranganath
NYU

David Sontag
MIT

1 MODEL ARCHITECTURES

We detail the neural architectures for each of the experimental setups. We use Keras (Chollet *et al.* , 2015) to implement the models described.

1.1 Pinwheel Dataset

Encoder: $p(z|x)$**:**

- $x \rightarrow$ Dense(20, 'relu')
- $h_1 \rightarrow$ Dense(20, 'relu') $\rightarrow h_2$
- $h_2 \rightarrow$ Dense(1) $\rightarrow \mu$
- $h_2 \rightarrow$ Dense(1) $\rightarrow \log \Sigma$

Decoder: $p(x|z)$**:**

- $z \rightarrow$ Dense(20, 'relu')
- $h_1 \rightarrow$ Dense(20, 'relu')
- $h_2 \rightarrow$ Dense(2) $\rightarrow \mu_{\text{obs}}$

Reasoning Model: $p(z|Q)$**:**

- $\{x_1, \ldots, x_Q\} \rightarrow p(z|x)$ **(Elementwise)**
- $\{[\mu_1, \log \Sigma_1], \ldots, [\mu_Q, \log \Sigma_Q]\}$ $\rightarrow$ PermutationEquivariant(20,'elu')
- $\{h_1^1, \ldots, h_Q^1\} \rightarrow$ PermutationEquivariant(20,'elu')
- $\{h_1^2, \ldots, h_Q^2\} \rightarrow$ PermutationInvariant(1) $\rightarrow \mu$
- $\{h_1^2, \ldots, h_Q^2\} \rightarrow$ PermutationInvariant(1) $\rightarrow \log \Sigma$

1.2 MiniImagenet Dataset

Embedding Network $f(x) \rightarrow x'$**:**

- $x \rightarrow$ ResNet18 (He *et al.* , 2016) Conv Layers (see below) $\rightarrow h_1$
- $h_1 \rightarrow$ AveragePooling $\rightarrow x'$

Encoder: $p(z|x')$**:**

- $x' \rightarrow$ Dense(512, 'relu') $\rightarrow h_1$
- $h_1 \rightarrow$ Dense(128, 'linear') $\rightarrow \mu$
- $h_1 \rightarrow$ Dense(128, 'linear') $\rightarrow \sigma$

Decoder: $p(x'|z)$**:**

- $z \rightarrow$ Dense(512, 'relu') $\rightarrow h_1$
- $h_1 \rightarrow$ Dense(256, 'linear') $\rightarrow \mu_{obs}$

Reasoning Model: $p(z|Q)$**:**

- $\{x_1, \ldots, x_Q\} \rightarrow p(z|x)$ **(Elementwise)**
- $\{[\mu_1, \log \Sigma_1], \ldots, [\mu_Q, \log \Sigma_Q]\}$ $\rightarrow$ PermutationEquivariant(2048,'linear')
- $\{h_1^2, \ldots, h_Q^2\} \rightarrow$ PermutationInvariant(128) $\rightarrow \mu$
- $\{h_1^2, \ldots, h_Q^2\} \rightarrow$ PermutationInvariant(128) $\rightarrow \log \Sigma$

Training Details:

We take $|Qs| = 1, |Q_{ns}| = 5$, learning rate $= 5e - 5$.

1.3 MNIST Dataset

Encoder: $p(z|x)$**:**

- $x \rightarrow$ Flatten() $\rightarrow h_1$
- $h_1 \rightarrow$ Dense(500, 'relu') $\rightarrow h_2$
- $h_2 \rightarrow$ Dense(500, 'relu') $\rightarrow h_3$
- $h_3 \rightarrow$ Dense(2) $\rightarrow \mu$
- $h_3 \rightarrow$ Dense(2) $\rightarrow \sigma$

Decoder: $p(x|z)$**:**

- $z \rightarrow$ Dense(500, 'relu') $\rightarrow h_1$
- $h_1 \rightarrow$ Dense(784, 'sigmoid') $\rightarrow h_2$
- $h_2 \rightarrow$ Reshape((28,28)) $\rightarrow \mu$

Reasoning Model: $p(z|Q)$**:**

- $\{x_1, \ldots, x_Q\} \rightarrow p(z|x)$ **(Elementwise)**
- $\{[\mu_1, \log \Sigma_1], \ldots, [\mu_Q, \log \Sigma_Q]\}$
  $\rightarrow$ PermutationEquivariant(20,'relu')
- $\{h_1^2, \ldots, h_Q^2\} \rightarrow$ PermutationInvariant(2) $\rightarrow \mu$
- $\{h_1^2, \ldots, h_Q^2\} \rightarrow$ PermutationInvariant(2) $\rightarrow \log \Sigma$

Training Details: We take $|Qs| = 5, |Q_{ns}| = 5$, learning rate $= 1e - 4$.

References

Chollet, François, *et al.* . 2015. *Keras*. `https://github.com/keras-team/keras`.

He, Kaiming, Zhang, Xiangyu, Ren, Shaoqing, & Sun, Jian. 2016. Deep residual learning for image recognition. *In: CVPR*.