In the Class BayesianLayer, the priors were initialized as tensor in the dimension input times the output. The weights were defined as parameters of the neural network. So that, the weights will be optimized during the optimization step. In addition, the weights were randomly initialized as a normal distribution.

For the method feedforward, the weights were sampled with the reparametrization trick. After, the weights are multiplied with the inputs, to calculate the outputs.

In the method \_kl\_divergence, the formula 13 from the paper GRAVES 2011 was implemented to calculate the Kulback-Leiber divergence between the two gaussian prior and the weights as posterior.

In the class Bayesian net, the class probability prediction was implanted in a for loop. The mean of the outputs, after the number of different forward passes were transformed with Softmax, so that the number could be interpreted as a probability.

In addition, the calculation for the complete loss were implemented.

In the method train-network, the additional loss from the Bayesian net was added to the loss. However, before the KL-Loss was divided by the batch size as recommended in the paper from GRAVES 2011.

Different number of epochs and batch sizes were tested until the program scored good results.