

Appendix

Here we first give the theoretical formulation of our work DoPaNet to show that the modes captured should be different for each categorical code c_i and that the standard GAN can be considered as its lower bound on mode collapse. We then give some more experimental insights from the 2D task. We also give some more generations using CIFAR. Later we provide implementation details of the network architectures we used.

A. Theoretical formulation

Here we present the theoretical formulation for our proposed method DoPaNet.

Lemma 3.1. *For optimal Q and fixed G , the optimal D_i , $\forall i \in [1, \dots, N]$ is*

$$D_i^*(x) = \frac{\rho_{S_i} p_d^i(x)}{\rho_{S_i} p_d^i(x) + \frac{1}{N} p_g^i(x)} \quad (6)$$

where $S_i = \{x \in \text{Supp}(p_d) | Q(x) = c_i\}$, $\rho_{S_i} = \int_{x \in S_i} p_d(x) dx$, p_d^i is a probability distribution such that $p_d^i(x) = \frac{p_d(x)}{\rho_{S_i}}$ and $\text{Supp}(p_d^i) = S_i$, and $p_g^i(x) = p_z(z)$ such that $G(z, c_i) = x$.

Proof. Let us consider a case where we have $N = 2$ discriminators. The theoretical formulation for this case can be trivially extended to more number of discriminators. The objective being optimized by the generator and the discriminators is (Obj. 3):

$$\min_G \max_{D_1, D_2} M(\{D_1, D_2\}, G) := \left[\mathbb{E}_{\substack{x \sim p_d \\ \sigma \sim Q(x)}} [\log D_\sigma(x)] + \mathbb{E}_{\substack{z \sim p_z, c \sim p_c \\ \hat{\sigma} \sim Q(G(z, c))}} [\log (1 - D_{\hat{\sigma}}(G(z, c)))] \right] \quad (7)$$

When the classifier Q has converged to its optimal form, the above Equation 7 can be rewritten as:

$$\min_G \left[\max_{D_1} \left[\mathbb{E}_{\substack{x \sim p_d \\ x \in S_1}} [\log D_1(x)] + p_c(c_1) \mathbb{E}_{z \sim p_z} [\log (1 - D_1(G(z, c_1)))] \right] + \max_{D_2} \left[\mathbb{E}_{\substack{x \sim p_d \\ x \in S_2}} [\log D_2(x)] + p_c(c_2) \mathbb{E}_{z \sim p_z} [\log (1 - D_2(G(z, c_2)))] \right] \right] \quad (8)$$

where $x \in S_i$ if $Q(x) = c_i$. p_c is the categorical distribution and in our case equal probability is assigned to both the values c_1 and c_2 . Here c_i is that code vector which leads the classifier Q to pass $G(z, c_i)$ to D_i for $i \in [1, 2]$. Please note that we can therefore consider $G(\cdot, c_1)$ as $G_1(\cdot)$ and $G(\cdot, c_2)$ as $G_2(\cdot)$, where G_1 and G_2 have shared weights except the bias weights in the initial layer. Bias weights in the initial layer are independently trained for G_1 and G_2 .

The Objective 8 can be rewritten as:

$$\min_{G_1, G_2} \left[\max_{D_1} \left[\rho_{S_1} \mathbb{E}_{x \sim p_d^1} [\log D_1(x)] + \frac{1}{2} \mathbb{E}_{z \sim p_z} [\log (1 - D_1(G_1(z)))] \right] + \max_{D_2} \left[\rho_{S_2} \mathbb{E}_{x \sim p_d^2} [\log D_2(x)] + \frac{1}{2} \mathbb{E}_{z \sim p_z} [\log (1 - D_2(G_2(z)))] \right] \right] \quad (9)$$

where $\rho_{S_i} = \int_{x \in S_i} p_d(x) dx$. p_d^i is a probability distribution such that $p_d^i(x) = \frac{p_d(x)}{\rho_{S_i}}$ and $\text{Supp}(p_d^i) = S_i$ where $S_i = \{x \in \text{Supp}(p_d) | Q(x) = c_i\}$ is the set of samples in the i^{th} mode of the real distribution. So, sampling from p_d^i is same as sampling from the i^{th} mode of the real distribution p_d . Therefore, $\text{Supp}(p_d) = \text{Supp}(p_d^1) \cup \text{Supp}(p_d^2)$ and $\text{Supp}(p_d^1) \cap \text{Supp}(p_d^2) = \emptyset$.

For a fixed generator G , G_1 and G_2 are also fixed. For a given G_1 and G_2 , the discriminator D_i tries to maximize the quantity (using Objective 9):

$$\rho_{S_i} \int_x p_d^i(x) \log D_i(x) dx + \frac{1}{2} \int_z p_z(z) \log (1 - D_i(G_i(z))) dz = \int_x \rho_{S_i} p_d^i(x) \log D_i(x) + \frac{1}{2} p_g^i(x) \log (1 - D_i(x)) dx \quad (10)$$

where $p_g^i(x) = p_z(z)$ such that $G_i(z) = x$ for $i = 1, 2$. Therefore, for a fixed generator we get the optimal discriminator D_i as:

$$D_i^*(x) = \frac{\rho_{S_i} p_d^i(x)}{\rho_{S_i} p_d^i(x) + \frac{1}{2} p_g^i(x)} \quad (11)$$

In case of N discriminators, the optimal discriminator D_i can be similarly obtained as:

$$D_i^*(x) = \frac{\rho_{S_i} p_d^i(x)}{\rho_{S_i} p_d^i(x) + \frac{1}{N} p_g^i(x)} \quad (12)$$

□

Theorem 3.2. In case of N discriminators, the global minimum of $U(G)$ is achieved if and only if $p_g^i(x) = p_d^i(x)$, $\forall i \in [1, \dots, N]$. When $\rho_{S_i} = 1/N$, the global minimum value of $U(G)$ is $-\log(4)$.

Proof. Given the optimal discriminators D_1^* and D_2^* , we can reformulate the Objective 9 as:

$$\begin{aligned} \min_{G_1, G_2} & \left[\rho_{S_1} \mathbb{E}_{x \sim p_d^1} [\log D_1^*(x)] \right. \\ & + \frac{1}{2} \mathbb{E}_{z \sim p_z} [\log (1 - D_1^*(G_1(z)))] \\ & + \rho_{S_2} \mathbb{E}_{x \sim p_d^2} [\log D_2^*(x)] \\ & \left. + \frac{1}{2} \mathbb{E}_{z \sim p_z} [\log (1 - D_2^*(G_2(z)))] \right] \end{aligned} \quad (13)$$

As noted earlier, bias weights in the initial layer of G_1 and G_2 are independently trained with all the other weights shared. As it empirically turns out, the shared weights help learn the similar features, which are essential in low-level image formation and should be similar even if G_1 and G_2 were trained independently. So, we can rather relax the restriction and consider G_1 and G_2 to be independent of each other. So, the objective 13 can be rewritten as:

$$\min_{G_1} W(G_1) + \min_{G_2} W(G_2) \quad (14)$$

where,

$$\begin{aligned} W(G_i) := & \left[\rho_{S_i} \mathbb{E}_{x \sim p_d^i} [\log D_i^*(x)] \right. \\ & \left. + \frac{1}{2} \mathbb{E}_{z \sim p_z} [\log (1 - D_i^*(G_i(z)))] \right] \end{aligned} \quad (15)$$

This is same as optimizing different G_i - D_i pairs on dataset distributions p_d^i decided by the classifier Q based on the target real distribution p_d . Figure 3(b) and 3(c) empirically show that the assumption of disjoint support of the distributions p_d^1 and p_d^2 is valid. The Equation 15 can be rewritten as:

$$\begin{aligned} W(G_i) := & \left[\rho_{S_i} \mathbb{E}_{x \sim p_d^i} [\log D_i^*(x)] \right. \\ & \left. + \frac{1}{2} \mathbb{E}_{x \sim p_g^i} [\log (1 - D_i^*(x))] \right] \\ = & \left[\rho_{S_i} \mathbb{E}_{x \sim p_d^i} \left[\log \frac{\rho_{S_i} p_d^i(x)}{\rho_{S_i} p_d^i(x) + \frac{1}{N} p_g^i(x)} \right] \right. \\ & \left. + \frac{1}{2} \mathbb{E}_{x \sim p_g^i} \left[\log \left(\frac{\frac{1}{N} p_g^i(x)}{\rho_{S_i} p_d^i(x) + \frac{1}{N} p_g^i(x)} \right) \right] \right] \end{aligned} \quad (16)$$

We can further reformulate Equation 16 as:

$$\begin{aligned} W(G_i) := & \rho_{S_i} \left[-\log(c_1^i) + KL \left(p_d^i \left\| \frac{p_d^i}{c_1^i} + \frac{p_g^i}{2\rho_{S_i} c_1^i} \right\| \right) \right. \\ & + c_2^i \left(-\log(c_3^i) \right. \\ & \left. \left. + KL \left(\frac{p_g^i}{2\rho_{S_i} c_2^i} \left\| \frac{p_d^i}{c_2^i c_3^i} + \frac{p_g^i}{2\rho_{S_i} c_2^i c_3^i} \right\| \right) \right) \right] \end{aligned} \quad (17)$$

where KL is the Kullback-Leibler divergence, c_1^i , c_2^i and c_3^i are constants such that $2\rho_{S_i} c_2^i = 1$ ($\implies c_2^i = 1/2\rho_{S_i}$) for $p_g^i/2\rho_{S_i} c_2^i$ (the first distribution of second KL term) to be a probability distribution. The Kullback-Leibler divergence between two distributions is always non-negative and, zero iff the two distributions are equal. In above equation, the two KL terms are zero simultaneously when $(c_1^i - 1)(c_2^i - 1) = 1$ and the generator distribution is

$$p_g^i = 2(c_1^i - 1)\rho_{S_i} p_d^i$$

where $2(c_1^i - 1)\rho_{S_i} = 1$ ($\implies c_1^i = 1 + 1/2\rho_{S_i}$) for p_g^i to be a probability distribution. Therefore, the global minimum of Eq. 17 is achieved iff $p_g^i = p_d^i$. The constants in Eq. 17 are chosen such that:

$$c_1^i = 1 + \frac{1}{2\rho_{S_i}}, \quad c_2^i = \frac{1}{2\rho_{S_i}}, \quad c_3^i = 1 + 2\rho_{S_i} \quad (18)$$

Please note that when $\rho_{S_i} = 1/2$, the Eq. 17 can be reformulated as:

$$W(G_i) := -\log(2) + JSD(p_{\text{data}} \| p_g) \quad (19)$$

and the global minimum of $U(G)$ obtained is $-\log(4)$. This global minimum value is the same in general case for N discriminators when $\rho_{S_i} = 1/N$. $\square$

1188
1189
1190
1191
1192
1193
1194
1195
1196
1197
1198
1199
1200
1201
1202
1203
1204
1205
1206
1207
1208
1209
1210
1211
1212
1213
1214
1215
1216
1217
1218
1219
1220
1221
1222
1223
1224
1225
1226
1227
1228
1229
1230
1231
1232
1233
1234
1235
1236
1237
1238
1239
1240
1241

Corollary 3.2.1. *At global minimum of $U(G)$, the generative model G replicates the real distribution p_d , categorized into different modes.*

Proof. As noted in the proof of Lemma 3.1, sampling from p_d^i is same as sampling from the i^{th} mode of the real distribution p_d . At global minimum of $U(G)$, we have $p_g^i = p_d^i$ so $G_i(\cdot) = G(\cdot, c_i)$ is able to sample from the i^{th} mode of the real distribution. As the real distribution is categorized into N modes in total and each of $\{G(\cdot, c_i)\}_{i=1}^N$ can samples from the corresponding modes, so G can replicate the real distribution p_d , categorized into different modes. $\square$

1242
1243
1244
1245
1246
1247
1248
1249
1250
1251
1252
1253
1254
1255
1256
1257
1258
1259
1260
1261
1262
1263
1264
1265
1266
1267
1268
1269
1270
1271
1272
1273
1274
1275
1276
1277
1278
1279
1280
1281
1282
1283
1284
1285
1286
1287
1288
1289
1290
1291
1292
1293
1294
1295

Algorithm 1 DoPaNet training algorithm

- 1: **for** number of training iterations **do**
- 2: Sample minibatch of m noise samples $\{z^{(1)}, \dots, z^{(m)}\}$ from the noise prior $p_z(z)$.
- 3: Sample minibatch of m code samples $\{c^{(1)}, \dots, c^{(m)}\}$ from the code prior $p_c(c)$.
- 4: Sample minibatch of m examples $\{x^{(1)}, \dots, x^{(m)}\}$ from the data generating distribution $p_d(x)$.
- 5: Update Q by ascending its stochastic gradient:

$$\nabla_{\theta_a} \frac{1}{m} \sum_{i=1}^m c^{(i)} \cdot \log Q \left(G \left(z^{(i)}, c^{(i)} \right) \right)$$

- 6: Decide for every input which of the N discriminators to use by sampling from the likelihood distribution of Q :

$$\sigma(i) \sim Q \left(x^{(i)} \right) \quad \hat{\sigma}(i) \sim Q \left(G \left(z^{(i)}, c^{(i)} \right) \right)$$

- 7: $\forall n \in [1, \dots, N]$, define the set of samples that are assigned to the n^{th} discriminator D_n as:

$$\mathcal{D}_n = \left\{ x^{(i)} \mid \sigma(i) = n \right\} \quad \hat{\mathcal{D}}_n = \left\{ G \left(z^{(i)}, c^{(i)} \right) \mid \hat{\sigma}(i) = n \right\}$$

- 8: $\forall n \in [1, \dots, N]$, update the n^{th} discriminator by ascending its stochastic gradient:

$$\nabla_{\theta_{d_n}} \left(\frac{1}{|\mathcal{D}_n|} \sum_{x \in \mathcal{D}_n} \log D_n(x) + \frac{1}{|\hat{\mathcal{D}}_n|} \sum_{\hat{x} \in \hat{\mathcal{D}}_n} \log (1 - D_n(\hat{x})) \right)$$

- 9: Sample minibatch of m noise samples $\{z^{(1)}, \dots, z^{(m)}\}$ from the noise prior $p_z(z)$.
- 10: Sample minibatch of m code samples $\{c^{(1)}, \dots, c^{(m)}\}$ from the code prior $p_c(c)$.
- 11: Decide for every i^{th} input which of the N discriminators to use by sampling from likelihood distribution of Q :

$$\hat{\sigma}(i) \sim Q \left(G \left(z^{(i)}, c^{(i)} \right) \right)$$

- 12: Update the generator by descending its stochastic gradient:

$$\nabla_{\theta_g} \frac{1}{m} \sum_{i=1}^m \log \left(1 - D_{\hat{\sigma}(i)} \left(G \left(z^{(i)}, c^{(i)} \right) \right) \right)$$

- 13: **end for**
-

B. 2D GMM

As discussed in the section 4.1.2, here we show qualitative results with $N = M = 8$ (where N is the number of discriminators, M is the number of modes) whose corresponding quantitative results are mentioned in the Table 3. We illustrate our findings in Figure 9 and Figure 10.

We argue that GMAN is not able to utilize multiple discriminators in this experiment setup and that most of the learning is done by just a few discriminators rather than their effective ensemble (see Appendix B).

In (4.1.2), under Error Analysis paragraph we claimed that GMAN fails to utilize multiple discriminators to their full potential. In Figure 5 we already have visual proof: the gradient field of the first two discriminators (top row, red and blue) are almost identical to each other, while the gradient of the third network (top row, green) is pointing towards a completely different mode (lower left) in its non-adjacent area, while around this distant mode the magnitude of the gradient is relatively small.

C. CIFAR-10

As discussed in the Section 4.2.2, here we present some more results obtained for each of the 10 classes of CIFAR-10 in Figure 11 and Figure 12.

D. Implementation details

Here we present the way we structured our experiments and the details about the network architecture we used in the experiments.

D.1. Synthetic low dimensional distributions

First, we reproduced the 1D setting in [13] with 5 modes at $[10, 20, 60, 80, 110]$ and standard deviations $[3, 3, 2, 2, 1]$ respectively and we compare to the numbers reported in that paper in Table 1. Second, we compared DoPaNet directly to GMAN [11] qualitatively in Figure 5 and quantitatively in Table 3 using a circular 2D GMM distribution with 3 and 8 modes respectively, around the unit circle. To illustrate the advantage of DoPaNet over GMAN [11] we plotted the gradient field to visualize the benefit of using multiple discriminators. The gradient field of this setup can be seen in Figure 5. To get quantitative results, we estimated the probability density distribution using a histogram with 1400 bins over the real and generated samples and computed the Chi-square and KL-divergence between the two histograms.

Comparing against other GAN variants When comparing against other GAN variants, we run the 1D experiments using a fixed set of 200,000 samples from the real distribution and generate 65,536 elements from each model.

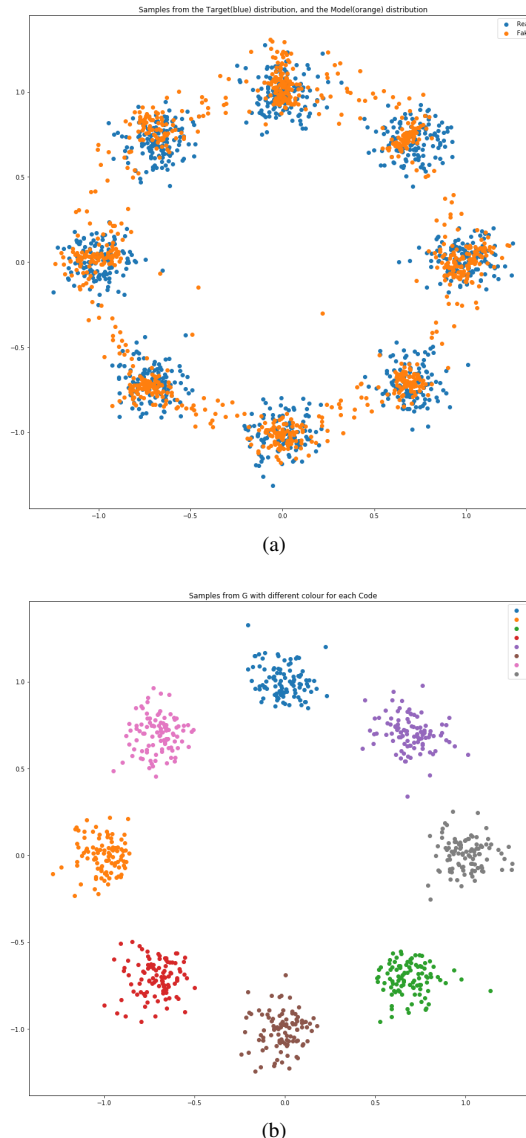


Figure 9. (a) GMAN: scatter plot of the data distribution (blue) and model distribution (orange). Although it is covering all the modes, there are a lot of false positives lying between the actual modes. (b) DoPaNet: scatter plot of the model distribution with different colors assigned for samples generated using different code. We sample c from uniform categorical distribution. Notice that the modes are clearly separated from each other as compared to GMAN.

Since DoPaNet is directly designed to separate different modes, we outperform all the other methods as shown in Table 1.

In our case, we sample the code vectors for the generator from a categorical distribution with uniform probability. For the best results, we use 5 discriminators in both GMAN [11] and DoPaNet. For both, we train 3 instances and select the best score from each of them.

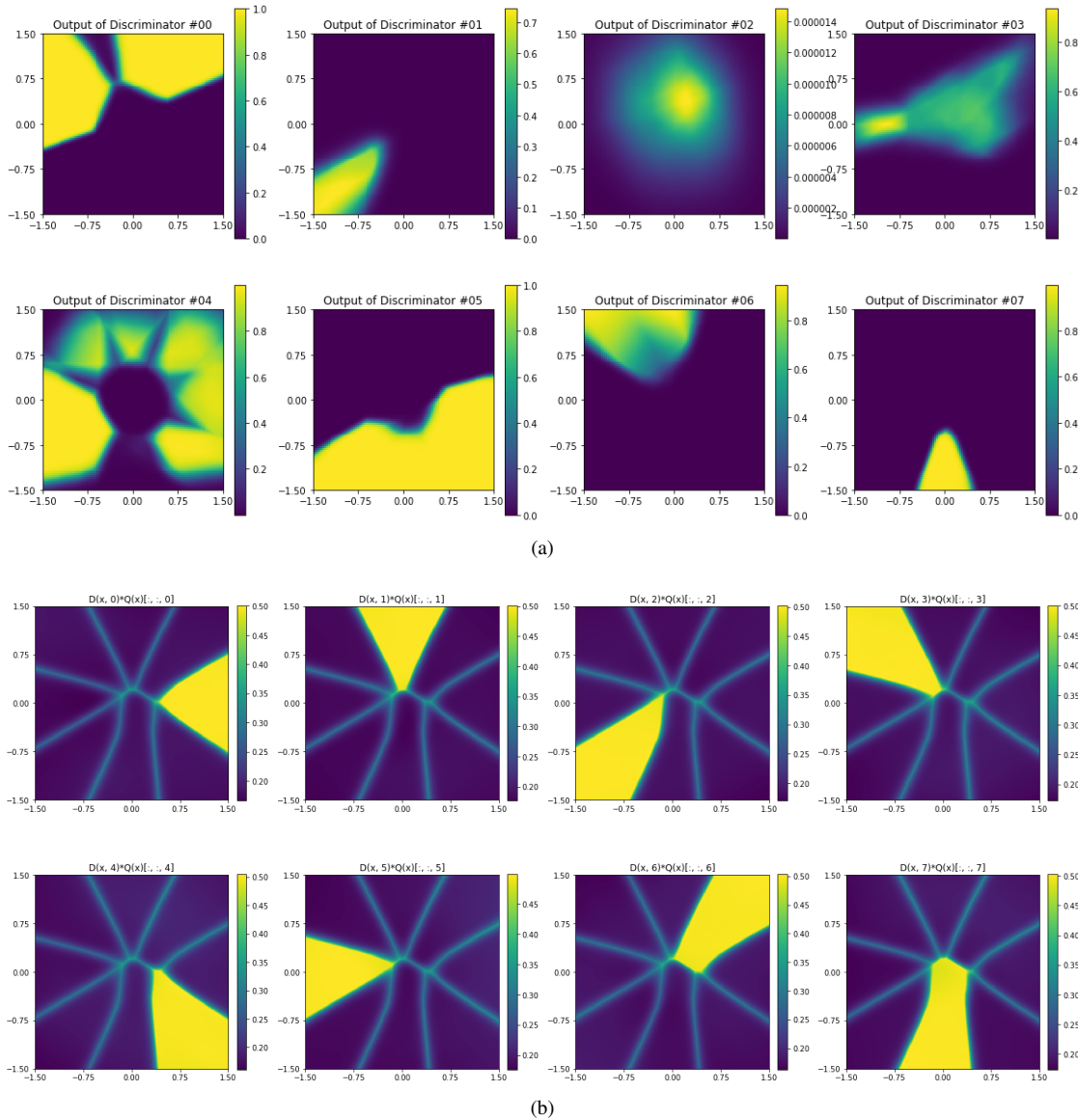


Figure 10. Heat map of discriminator scores $\in [0, 1]$ (0 signifies fake while 1 means real) when evaluated for every data-point pair lying in $[-1.5, 1.5]^2$ (corresponding to Figure 9). (a) GMAN: it is clear that discriminator #04 already covers the majority of the modes while the other discriminators give high scores for obviously fake samples (#03 and #05). (b) DoPaNet: here we multiplied the discriminator scores with the probability of each point being assigned to that discriminator (obtained from classifier Q). Although the capacity of each discriminator in DoPaNet is identical to the discriminators in the GMAN experiment, the DoPaNet framework reduces the complexity of each discriminator’s task by making it work only on a different identifiable mode.

Benchmarking the number of discriminators In 1D for better non-parametric probability density estimation, we increased the number of generated samples from 65,536 to 1,000,000 samples as done in [13]. For more reliable results on the implied mechanism of both approaches, we run the training 20 times for each algorithm with number of discriminators $N = 2, \dots, 8$, totaling 320 training. As in the previous experiment, we chose the best results from each run.

2D experiments In 2D, for both variants we experiment with $M = N = 8$ (where N is the number of discriminators, M is the number of Gaussians we used in the mixture) for quantitative results, listed in Table 3 and $M = N = 3$ setting for qualitative results, illustrated in Figure 5. For each experiment we use a fixed set of 1,000,000 samples and take 5 run per each algorithm, then report the best run. We took effort to make sure that the comparison was fair, and used the same settings of hyper-parameters as it was done in the 1D experiments: network structure, optimization method,

learning rate, mini-batch size, number of training steps, etc.

D.2. Image generation

For both the generator and discriminator we use ResNet-architectures [18], with 18 layers each in the CIFAR-10 experiments, and 26 layers each in the CelebA experiments. As was done in [27] we multiply the output of the ResNet blocks with 0.1, use 256-dimensional unit Gaussian distribution. For categorical conditional image generation we use an embedding network that projects category indices to 256 dimensional label vector, normalized to the unit sphere. We embed the code vector similar to the ground truth labels in this setting for CIFAR-10. We use Leaky-RELU nonlinearities everywhere, without BatchNorm.

Following the considerations in [27] for optimizing parameters of Q , D , G we use the RMSProp with $\alpha = 0.99$, $\epsilon = 10^{-8}$, and initial learning rate of 10^{-3} . We use a batch size of 64, and train the algorithm for 700,000 and 400,000 iterations for CIFAR-10 and CelebA tasks respectively. Similar to work that provided state of the art results on image generation tasks [21, 27] for visualizing the generator’s progress we use an exponential moving average of the parameters of G with decay 0.999.

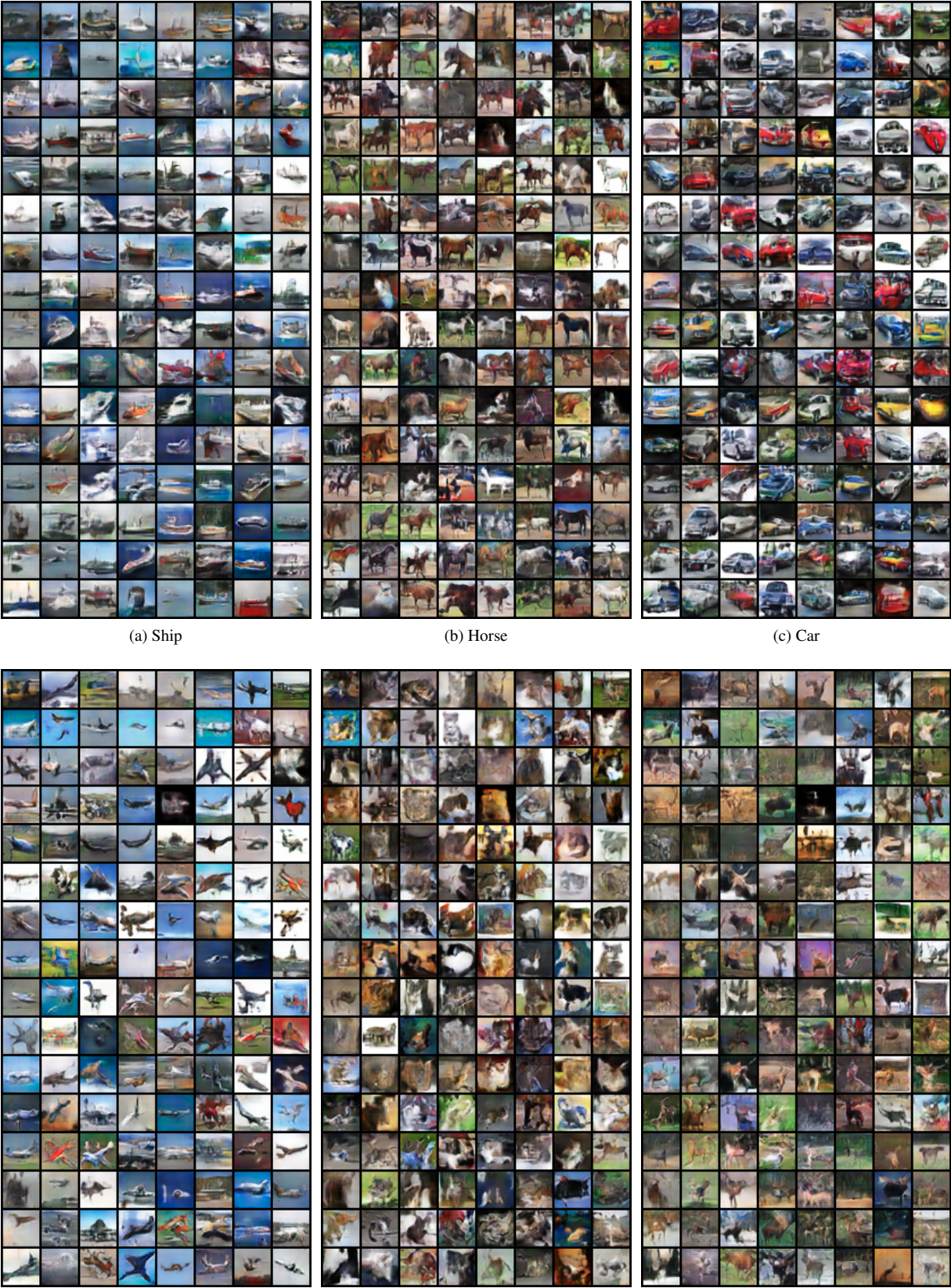


Figure 11. 32×32 CIFAR-10 samples drawn from DoPaNet trained with $N = 5$ discriminators for 700k iterations. The generations in each subfigure correspond to $y = 1, \dots, 6$ respectively. We call the attention to the various different details learnt for each class which can cause mode collapse in other GAN variants.

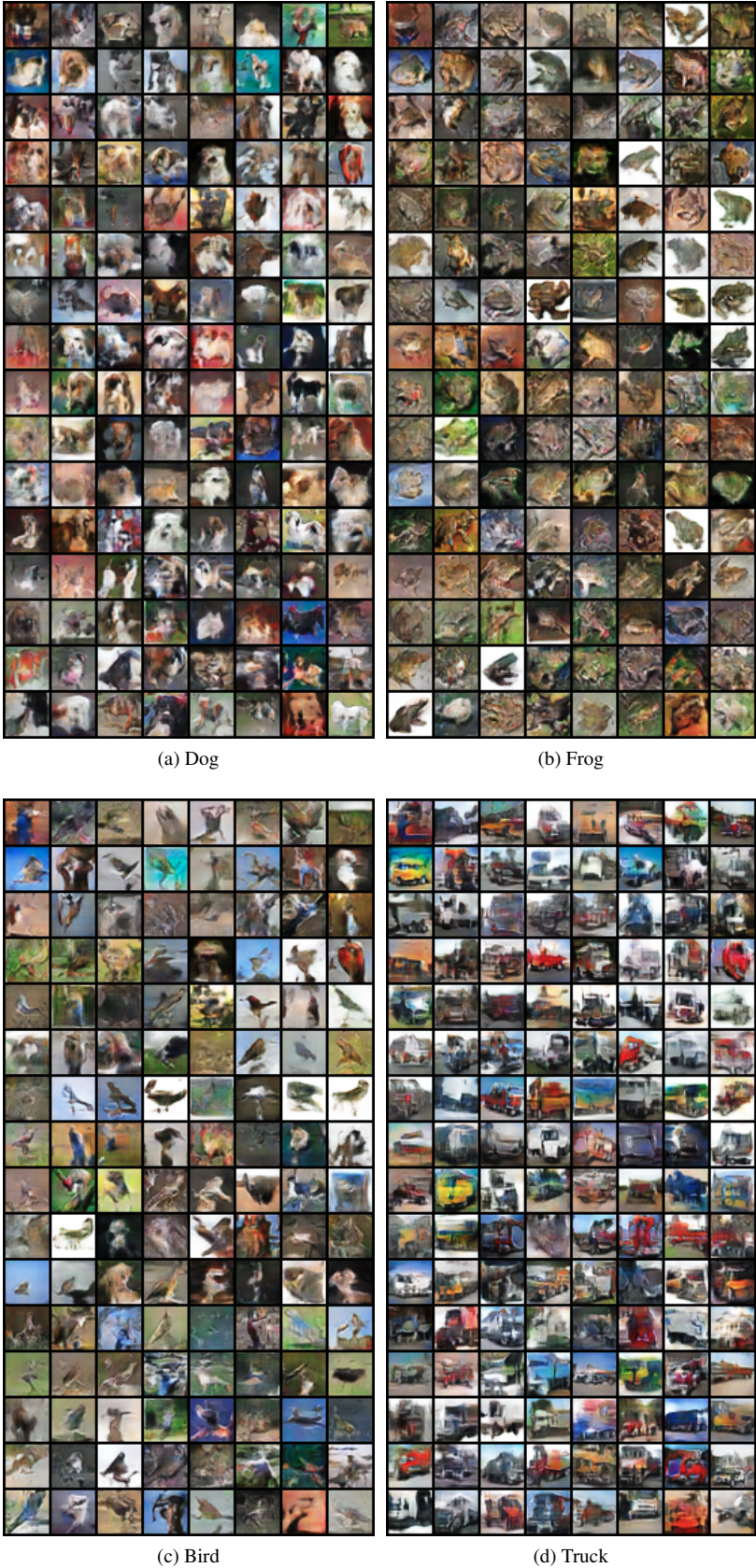


Figure 12. 32×32 CIFAR-10 samples drawn from DoPaNet trained with $N = 5$ discriminators for 700k iterations. The generations in each subfigure correspond to $y = 7, 8, 9$ and 10 respectively. We call the attention to the various different details learnt for each class which can cause mode collapse in other GAN variants.