

## MNIST Diffusion Project

Generated by Doxygen 1.9.1



<b>1 Namespace Index</b>	<b>1</b>
1.1 Packages	1
<b>2 Hierarchical Index</b>	<b>3</b>
2.1 Class Hierarchy	3
<b>3 Class Index</b>	<b>5</b>
3.1 Class List	5
<b>4 File Index</b>	<b>7</b>
4.1 File List	7
<b>5 Namespace Documentation</b>	<b>9</b>
5.1 ddpm_train Namespace Reference	9
5.1.1 Function Documentation	9
5.1.1.1 ddpm_schedules()	10
5.1.2 Variable Documentation	10
5.1.2.1 accelerator	10
5.1.2.2 avg_loss	10
5.1.2.3 dataloader	10
5.1.2.4 dataset	10
5.1.2.5 ddpm	10
5.1.2.6 grid	11
5.1.2.7 gt	11
5.1.2.8 loss	11
5.1.2.9 losses	11
5.1.2.10 n_epoch	11
5.1.2.11 optim	11
5.1.2.12 pbar	11
5.1.2.13 tf	11
5.1.2.14 xh	11
<b>6 Class Documentation</b>	<b>13</b>
6.1 ddpm_train.CNN Class Reference	13
6.1.1 Constructor & Destructor Documentation	14
6.1.1.1 __init__()	14
6.1.2 Member Function Documentation	14
6.1.2.1 forward()	14
6.1.2.2 time_encoding()	14
6.1.3 Member Data Documentation	15
6.1.3.1 blocks	15
6.1.3.2 time_embed	15
6.2 ddpm_train.CNNBlock Class Reference	15
6.2.1 Constructor & Destructor Documentation	16

---

6.2.1.1 <code>__init__()</code> . . . . .	16
6.2.2 Member Function Documentation . . . . .	16
6.2.2.1 <code>forward()</code> . . . . .	16
6.2.3 Member Data Documentation . . . . .	16
6.2.3.1 <code>net</code> . . . . .	16
6.3 <code>ddpm_train.DDPM</code> Class Reference . . . . .	17
6.3.1 Constructor & Destructor Documentation . . . . .	17
6.3.1.1 <code>__init__()</code> . . . . .	18
6.3.2 Member Function Documentation . . . . .	18
6.3.2.1 <code>forward()</code> . . . . .	18
6.3.2.2 <code>sample()</code> . . . . .	18
6.3.3 Member Data Documentation . . . . .	18
6.3.3.1 <code>criterion</code> . . . . .	18
6.3.3.2 <code>gt</code> . . . . .	18
6.3.3.3 <code>n_T</code> . . . . .	18
<b>7 File Documentation</b> . . . . .	<b>19</b>
7.1 <code>/home/jhughes2712/projects/m2_assessment/jh2284/src/ddpm_train.py</code> File Reference . . . . .	19
7.1.1 Detailed Description . . . . .	20
<b>Index</b> . . . . .	<b>21</b>

# Chapter 1

## Namespace Index

### 1.1 Packages

Here are the packages with brief descriptions (if available):

<a href="#">ddpm_train</a> . . . . .	9
--------------------------------------	---



## Chapter 2

# Hierarchical Index

### 2.1 Class Hierarchy

This inheritance list is sorted roughly, but not completely, alphabetically:

nn.Module	
ddpm_train.CNN . . . . .	<a href="#">13</a>
ddpm_train.CNNBlock . . . . .	<a href="#">15</a>
ddpm_train.DDPM . . . . .	<a href="#">17</a>





## Chapter 3

# Class Index

### 3.1 Class List

Here are the classes, structs, unions and interfaces with brief descriptions:

<a href="#">ddpm_train.CNN</a>	13
<a href="#">ddpm_train.CNNBlock</a>	15
<a href="#">ddpm_train.DDPM</a>	17



## Chapter 4

# File Index

### 4.1 File List

Here is a list of all files with brief descriptions:

<code>/home/jhughes2712/projects/m2_assessment/jh2284/src/ddpm_train.py</code>	
Runs DDPM diffusion model . . . . .	19



## Chapter 5

# Namespace Documentation

### 5.1 ddpm\_train Namespace Reference

#### Classes

- class [CNNBlock](#)
- class [CNN](#)
- class [DDPM](#)

#### Functions

- Dict[str, torch.Tensor] [ddpm\\_schedules](#) (float beta1, float beta2, int T)  
*Returns pre-computed schedules for [DDPM](#) sampling with a linear noise schedule.*

#### Variables

- [tf](#) = transforms.Compose([transforms.ToTensor(), transforms.Normalize((0.5,), (1.0))])
- [dataset](#) = MNIST("./data", train=True, download=True, transform=[tf](#))
- [dataloader](#) = DataLoader([dataset](#), batch\_size=128, shuffle=True, num\_workers=4, drop\_last=True)
- [gt](#) = [CNN](#)(in\_channels=1, expected\_shape=(28, 28), n\_hidden=(16, 32, 32, 16), act=nn.GELU)
- [ddpm](#) = [DDPM](#)([gt](#)=[gt](#), betas=(1e-4, 0.02), n\_T=1000)
- [optim](#) = torch.optim.Adam([ddpm](#).parameters(), lr=2e-4)
- [accelerator](#) = Accelerator()
- int [n\\_epoch](#) = 100
- list [losses](#) = []
- [pbar](#) = tqdm([dataloader](#))
- [loss](#) = [ddpm](#)(x)
- [avg\\_loss](#) = np.average([losses](#)[min(len([losses](#))-100, 0):])
- [xh](#) = [ddpm](#).sample(16, (1, 28, 28), accelerator.device)
- [grid](#) = make\_grid([xh](#), nrow=4)

#### 5.1.1 Function Documentation

#### 5.1.1.1 `ddpm_schedules()`

```
Dict[str, torch.Tensor] ddpm_train.ddpm_schedules (
    float beta1,
    float beta2,
    int T )
```

Returns pre-computed schedules for [DDPM](#) sampling with a linear noise schedule.

### 5.1.2 Variable Documentation

#### 5.1.2.1 `accelerator`

```
ddpm_train.accelerator = Accelerator()
```

#### 5.1.2.2 `avg_loss`

```
ddpm_train.avg_loss = np.average(losses[min(len(losses)-100, 0):])
```

#### 5.1.2.3 `dataloader`

```
ddpm_train.dataloader = DataLoader(dataset, batch_size=128, shuffle=True, num_workers=4, drop↵
    _last=True)
```

#### 5.1.2.4 `dataset`

```
ddpm_train.dataset = MNIST("./data", train=True, download=True, transform=tf)
```

#### 5.1.2.5 `ddpm`

```
ddpm_train.ddpm = DDPM(gt=gt, betas=(1e-4, 0.02), n_T=1000)
```

#### 5.1.2.6 grid

```
ddpm_train.grid = make_grid(xh, nrow=4)
```

#### 5.1.2.7 gt

```
ddpm_train.gt = CNN(in_channels=1, expected_shape=(28, 28), n_hidden=(16, 32, 32, 16), act=nn.L↵  
GELU)
```

#### 5.1.2.8 loss

```
ddpm_train.loss = ddpm(x)
```

#### 5.1.2.9 losses

```
list ddpm_train.losses = []
```

#### 5.1.2.10 n\_epoch

```
int ddpm_train.n_epoch = 100
```

#### 5.1.2.11 optim

```
ddpm_train.optim = torch.optim.Adam(ddpm.parameters(), lr=2e-4)
```

#### 5.1.2.12 pbar

```
ddpm_train.pbar = tqdm(dataloader)
```

#### 5.1.2.13 tf

```
ddpm_train.tf = transforms.Compose([transforms.ToTensor(), transforms.Normalize((0.5, ), (1.↵  
0))])
```

#### 5.1.2.14 xh

```
ddpm_train.xh = ddpm.sample(16, (1, 28, 28), accelerator.device)
```



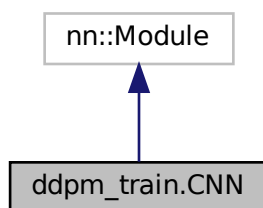


## Chapter 6

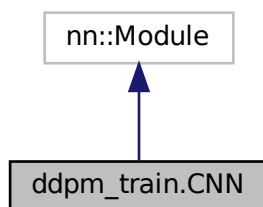
# Class Documentation

### 6.1 ddpm\_train.CNN Class Reference

Inheritance diagram for ddpm\_train.CNN:



Collaboration diagram for ddpm\_train.CNN:



## Public Member Functions

- None `__init__` (self, in\_channels, expected\_shape=(28, 28), n\_hidden=(64, 128, 64), kernel\_size=7, last\_kernel\_size=3, time\_embeddings=16, act=nn.GELU)
- torch.Tensor `time_encoding` (self, torch.Tensor t)
- torch.Tensor `forward` (self, torch.Tensor x, torch.Tensor t)

## Public Attributes

- `blocks`
- `time_embed`

## 6.1.1 Constructor & Destructor Documentation

### 6.1.1.1 `__init__()`

```
None ddpm_train.CNN.__init__ (
    self,
    in_channels,
    expected_shape = (28, 28),
    n_hidden = (64, 128, 64),
    kernel_size = 7,
    last_kernel_size = 3,
    time_embeddings = 16,
    act = nn.GELU )
```

## 6.1.2 Member Function Documentation

### 6.1.2.1 `forward()`

```
torch.Tensor ddpm_train.CNN.forward (
    self,
    torch.Tensor x,
    torch.Tensor t )
```

### 6.1.2.2 `time_encoding()`

```
torch.Tensor ddpm_train.CNN.time_encoding (
    self,
    torch.Tensor t )
```

### 6.1.3 Member Data Documentation

#### 6.1.3.1 blocks

`ddpm_train.CNN.blocks`

#### 6.1.3.2 time\_embed

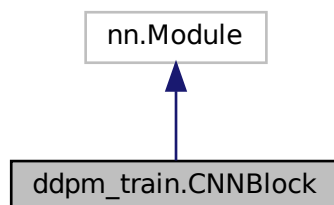
`ddpm_train.CNN.time_embed`

The documentation for this class was generated from the following file:

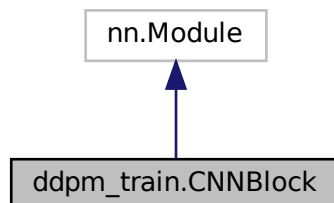
- [/home/jhughes2712/projects/m2\\_assessment/jh2284/src/ddpm\\_train.py](#)

## 6.2 ddpm\_train.CNNBlock Class Reference

Inheritance diagram for `ddpm_train.CNNBlock`:



Collaboration diagram for `ddpm_train.CNNBlock`:



## Public Member Functions

- def `__init__` (self, in\_channels, out\_channels, \*expected\_shape, act=nn.GELU, kernel\_size=7)
- def `forward` (self, x)

## Public Attributes

- `net`

## 6.2.1 Constructor & Destructor Documentation

### 6.2.1.1 `__init__()`

```
def ddpm_train.CNNBlock.__init__ (
    self,
    in_channels,
    out_channels,
    * expected_shape,
    act = nn.GELU,
    kernel_size = 7 )
```

## 6.2.2 Member Function Documentation

### 6.2.2.1 `forward()`

```
def ddpm_train.CNNBlock.forward (
    self,
    x )
```

## 6.2.3 Member Data Documentation

### 6.2.3.1 `net`

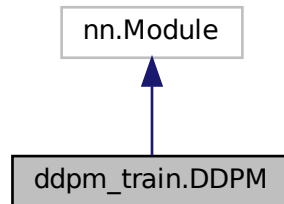
```
ddpm_train.CNNBlock.net
```

The documentation for this class was generated from the following file:

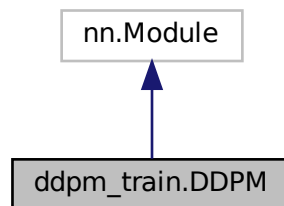
- [/home/jhughes2712/projects/m2\\_assessment/jh2284/src/ddpm\\_train.py](/home/jhughes2712/projects/m2_assessment/jh2284/src/ddpm_train.py)

## 6.3 ddpm\_train.DDPM Class Reference

Inheritance diagram for ddpm\_train.DDPM:



Collaboration diagram for ddpm\_train.DDPM:



### Public Member Functions

- None `__init__` (self, `gt`, Tuple[float, float] `betas`, int `n_T`, nn.Module `criterion`=nn.MSELoss())
- torch.Tensor `forward` (self, torch.Tensor `x`)

*Algorithm 18.1 in Prince.*

- torch.Tensor `sample` (self, int `n_sample`, size, device)

*Algorithm 18.2 in Prince.*

### Public Attributes

- `gt`
- `n_T`
- `criterion`

#### 6.3.1 Constructor & Destructor Documentation

### 6.3.1.1 `__init__()`

```
None ddpm_train.DDPM.__init__ (
    self,
    gt,
    Tuple[float, float] betas,
    int n_T,
    nn.Module criterion = nn.MSELoss() )
```

## 6.3.2 Member Function Documentation

### 6.3.2.1 `forward()`

```
torch.Tensor ddpm_train.DDPM.forward (
    self,
    torch.Tensor x )
```

Algorithm 18.1 in Prince.

### 6.3.2.2 `sample()`

```
torch.Tensor ddpm_train.DDPM.sample (
    self,
    int n_sample,
    size,
    device )
```

Algorithm 18.2 in Prince.

## 6.3.3 Member Data Documentation

### 6.3.3.1 `criterion`

```
ddpm_train.DDPM.criterion
```

### 6.3.3.2 `gt`

```
ddpm_train.DDPM.gt
```

### 6.3.3.3 `n_T`

```
ddpm_train.DDPM.n_T
```

The documentation for this class was generated from the following file:

- [/home/jhughes2712/projects/m2\\_assessment/jh2284/src/ddpm\\_train.py](/home/jhughes2712/projects/m2_assessment/jh2284/src/ddpm_train.py)

# Chapter 7

## File Documentation

### 7.1 /home/jhughes2712/projects/m2\_assessment/jh2284/src/ddpm\_train.py File Reference

Runs DDPM diffusion model.

#### Classes

- class [ddpm\\_train.CNNBlock](#)
- class [ddpm\\_train.CNN](#)
- class [ddpm\\_train.DDPM](#)

#### Namespaces

- [ddpm\\_train](#)

#### Functions

- Dict[str, torch.Tensor] [ddpm\\_train.ddpm\\_schedules](#) (float beta1, float beta2, int T)  
*Returns pre-computed schedules for [DDPM](#) sampling with a linear noise schedule.*

#### Variables

- [ddpm\\_train.tf](#) = transforms.Compose([transforms.ToTensor(), transforms.Normalize((0.5,), (1.0))])
- [ddpm\\_train.dataset](#) = MNIST("./data", train=True, download=True, transform=tf)
- [ddpm\\_train.dataloader](#) = DataLoader(dataset, batch\_size=128, shuffle=True, num\_workers=4, drop\_  
last=True)
- [ddpm\\_train.gt](#) = CNN(in\_channels=1, expected\_shape=(28, 28), n\_hidden=(16, 32, 32, 16), act=nn.GELU)
- [ddpm\\_train.ddpm](#) = DDPM(gt=gt, betas=(1e-4, 0.02), n\_T=1000)
- [ddpm\\_train.optim](#) = torch.optim.Adam(ddpm.parameters(), lr=2e-4)
- [ddpm\\_train.accelerator](#) = Accelerator()
- int [ddpm\\_train.n\\_epoch](#) = 100
- list [ddpm\\_train.losses](#) = []
- [ddpm\\_train.pbar](#) = tqdm(dataloader)
- [ddpm\\_train.loss](#) = ddpm(x)
- [ddpm\\_train.avg\\_loss](#) = np.average(losses[min(len(losses)-100, 0):])
- [ddpm\\_train.xh](#) = ddpm.sample(16, (1, 28, 28), accelerator.device)
- [ddpm\\_train.grid](#) = make\_grid(xh, nrow=4)

### 7.1.1 Detailed Description

Runs DDPM diffusion model.

#### Author

Created by J. Hughes on 18/03/2024.



# Index

/home/jhughes2712/projects/m2\_assessment/jh2284/src/ddpm\_train.py, 16  
19  
\_\_init\_\_  
    ddpm\_train.CNN, 14  
    ddpm\_train.CNNBlock, 16  
    ddpm\_train.DDPM, 17  
accelerator  
    ddpm\_train, 10  
avg\_loss  
    ddpm\_train, 10  
blocks  
    ddpm\_train.CNN, 15  
criterion  
    ddpm\_train.DDPM, 18  
dataloader  
    ddpm\_train, 10  
dataset  
    ddpm\_train, 10  
ddpm  
    ddpm\_train, 10  
ddpm\_schedules  
    ddpm\_train, 9  
ddpm\_train, 9  
    accelerator, 10  
    avg\_loss, 10  
    dataloader, 10  
    dataset, 10  
    ddpm, 10  
    ddpm\_schedules, 9  
    grid, 10  
    gt, 11  
    loss, 11  
    losses, 11  
    n\_epoch, 11  
    optim, 11  
    pbar, 11  
    tf, 11  
    xh, 11  
ddpm\_train.CNN, 13  
    \_\_init\_\_, 14  
    blocks, 15  
    forward, 14  
    time\_embed, 15  
    time\_encoding, 14  
ddpm\_train.CNNBlock, 15  
    \_\_init\_\_, 16  
    forward, 16  
    net, 16  
ddpm\_train.DDPM, 17  
    \_\_init\_\_, 17  
    criterion, 18  
    forward, 18  
    gt, 18  
    n\_T, 18  
    sample, 18  
forward  
    ddpm\_train.CNN, 14  
    ddpm\_train.CNNBlock, 16  
    ddpm\_train.DDPM, 18  
grid  
    ddpm\_train, 10  
gt  
    ddpm\_train, 11  
    ddpm\_train.DDPM, 18  
loss  
    ddpm\_train, 11  
losses  
    ddpm\_train, 11  
n\_epoch  
    ddpm\_train, 11  
n\_T  
    ddpm\_train.DDPM, 18  
net  
    ddpm\_train.CNNBlock, 16  
optim  
    ddpm\_train, 11  
pbar  
    ddpm\_train, 11  
sample  
    ddpm\_train.DDPM, 18  
tf  
    ddpm\_train, 11  
time\_embed  
    ddpm\_train.CNN, 15  
time\_encoding  
    ddpm\_train.CNN, 14  
xh  
    ddpm\_train, 11