

# PyTorch Cheatsheet

PyTorch is a popular open-source machine learning library based on the Torch library. It provides easy-to-use APIs for building and training deep neural networks. This cheatsheet provides a quick reference for some of PyTorch's unique features, including code blocks for variables, functions, loops, conditionals, file manipulation, and more. Additionally, it includes a list of resources for further learning.

## Variables

```
import torch

# Create a tensor
x = torch.tensor([1, 2, 3])

# Create a matrix
y = torch.tensor([[1, 2], [3, 4]])

# Get the shape of a tensor
x.shape

# Get the number of dimensions of a tensor
x.ndim

# Get the size of a tensor
x.size()

# Reshape a tensor
x.reshape(3, 1)

# Concatenate two tensors
torch.cat((x, x))

# Get the maximum value of a tensor
x.max()
```

## Functions

```
import torch.nn.functional as F

# Apply softmax function
F.softmax(x, dim=0)

# Apply ReLU activation function
F.relu(x)

# Apply cross-entropy loss function
F.cross_entropy(y_pred, y_true)
```

## Loops and Conditionals

```
# For loop
for i in range(10):
    print(i)

# While loop
while x < 10:
    print(x)
    x += 1

# If statement
if x > 0:
    print("x is positive")
elif x == 0:
    print("x is zero")
else:
    print("x is negative")
```

## File Manipulation

```
import torch

# Save a tensor to a file
torch.save(x, 'x.pt')

# Load a tensor from a file
x = torch.load('x.pt')
```

## Other Useful Features

```
import torch

# Set the random seed for reproducibility
torch.manual_seed(42)

# Move a tensor to the GPU
x.cuda()

# Compute gradients
x.requires_grad = True
y = x**2
y.backward()

# Define a neural network
import torch.nn as nn
```

```
class Net(nn.Module):
    def __init__(self):
        super(Net, self).__init__()
        self.fc1 = nn.Linear(10, 5)
        self.fc2 = nn.Linear(5, 1)

    def forward(self, x):
        x = F.relu(self.fc1(x))
        x = self.fc2(x)
        return x
```

## Resources

- [PyTorch documentation](#)
- [PyTorch tutorials](#)
- [Deep Learning with PyTorch book](#)