

PyTorch Cheatsheet

Tensor Fundamentals (Recitation 0C)

- **Library name:** torch
- **Check GPU availability:** torch.cuda.is_available()
- **Tensors creation:**
 - **Ones tensor:** torch.ones(size=(m, n))
 - **Zero tensor:** torch.zeros(size=(m, n))
 - **Identity matrix:** torch.eye(m)
 - **Random tensor - Uniform distribution [0, 1) :** torch.rand(size=(m, n))
 - **Random int tensor:** torch.randint(<low>, <high>, <shape>)
- **Conversion:**
 - torch.tensor(<list/array>)
 - torch.from_array(<array>)
- **Push tensor and model to device ("cuda" or "cpu"):**
 - tensor.to(device)
 - model.to(device)
- **Flattening (Nd to 1d tensor):** tensor.flatten()
- **Add an extra dimension at position d:** torch.unsqueeze(tensor, dim= d)
- **Remove extra dimension at position d:** torch.squeeze(tensor, dim= d)
- **Reshape dimension:**
 - tensor.reshape(<new_shape>)
 - tensor.view(<new_dim>)
- **Reorder dimensions:**
 - **Swap dimensions d1 and d2:** torch.transpose(tensor, d1, d2)
 - **Swap multiple dimensions:** torch.permute(tensor, <new_dim_order>)
eg: torch.permute(x, (1,2,0))
Put dim1 at 0th; Put dim2 at 1st, Put dim0 at 3rd
- **Combining tensors:**
 - **With existing axis at d:** torch.cat([tensor1, tensor2, ...], dim= d)
 - **With new axis at d:** torch.stack([tensor1, tensor2, ...], dim= d)
- **Padding:** torch.nn.functional.pad(tensor, (left, right, top, bottom), mode= <mode>, value= <value>)
- **Mathematical Operations:**
 - torch.sum(), torch.mean(), torch.max(), torch.min(), torch.argmax(), torch.std(), etc. Visit [torchdocs](https://pytorch.org/docs)

