



10-315 Introduction to ML

Practical and Responsible ML

Instructor: Pat Virtue

Great Power

Neural Network Toolkits

Pytorch

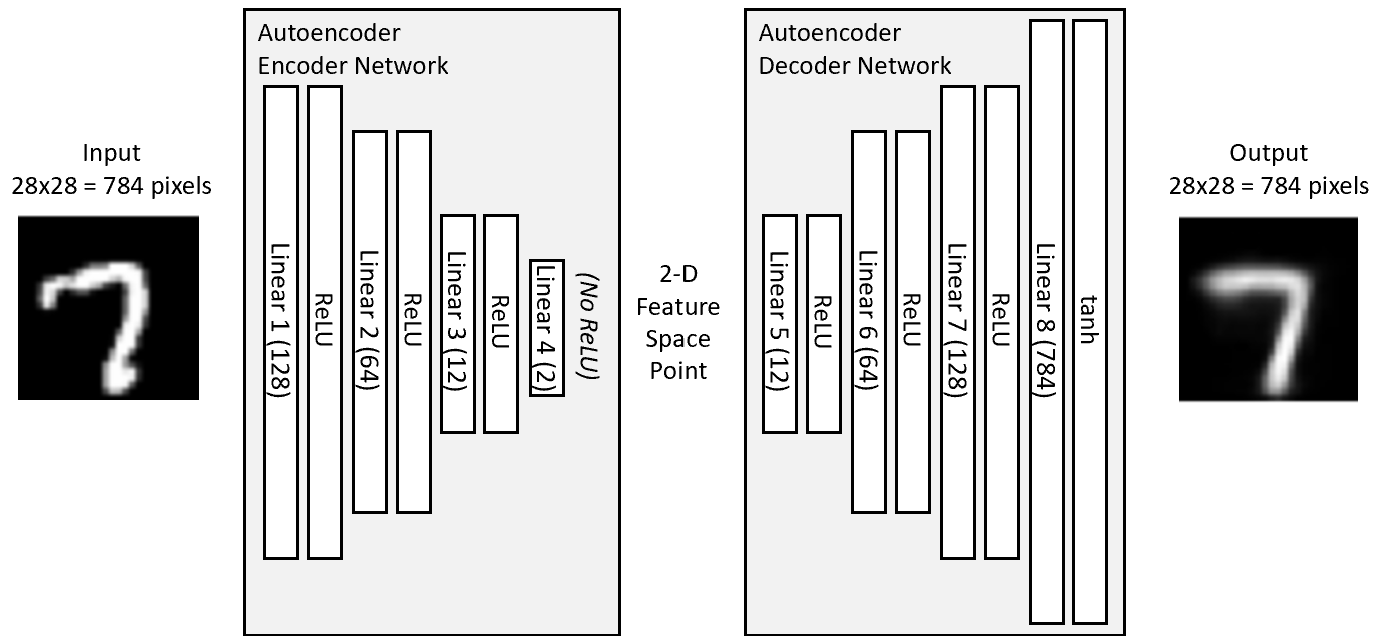
Network Network Toolkits

Pytorch in this course

- Already used behind the scenes in HW1
- HW7 transfer learning and exploration
- Mini-project if you want

Pytorch for HW1 Networks

Autoencoder



Source: https://github.com/L1aoXingyu/pytorch-beginner/blob/master/08-AutoEncoder/simple_autoencoder.py

```
class autoencoder(nn.Module):
    def __init__(self, bottleneck=2):
        super().__init__()
        self.encoder = nn.Sequential(
            nn.Linear(28 * 28, 128),
            nn.ReLU(True),
            nn.Linear(128, 64),
            nn.ReLU(True),
            nn.Linear(64, 12),
            nn.ReLU(True),
            # nn.Linear(12, 2))
            nn.Linear(12, bottleneck))

        self.decoder = nn.Sequential(
            # nn.Linear(2, 12),
            nn.Linear(bottleneck, 12),
            nn.ReLU(True),
            nn.Linear(12, 64),
            nn.ReLU(True),
            nn.Linear(64, 128),
            nn.ReLU(True),
            nn.Linear(128, 28 * 28),
            nn.Tanh())

    def forward(self, x):
        x = self.encoder(x)
        x = self.decoder(x)
        return x
```

Pytorch for HW1 Networks

Autoencoder

```
for epoch in range(num_epochs):
    for data in dataloader:
        img, _ = data
        img = img.view(img.size(0), -1)
        if torch.cuda.is_available():
            img = img.cuda()
        #
        =====forward=====
        output = model(img)
        loss = criterion(output, img)
        #
        =====backward=====
         optimizer.zero_grad()
        loss.backward()
        optimizer.step()
```

Source: https://github.com/L1aoXingyu/pytorch-beginner/blob/master/08-AutoEncoder/simple_autoencoder.py

```
class autoencoder(nn.Module):
    def __init__(self, bottleneck=2):
        super().__init__()
        self.encoder = nn.Sequential(
            nn.Linear(28 * 28, 128),
            nn.ReLU(True),
            nn.Linear(128, 64),
            nn.ReLU(True),
            nn.Linear(64, 12),
            nn.ReLU(True),
            # nn.Linear(12, 2))
            nn.Linear(12, bottleneck))

        self.decoder = nn.Sequential(
            # nn.Linear(2, 12),
            nn.Linear(bottleneck, 12),
            nn.ReLU(True),
            nn.Linear(12, 64),
            nn.ReLU(True),
            nn.Linear(64, 128),
            nn.ReLU(True),
            nn.Linear(128, 28 * 28),
            nn.Tanh())

    def forward(self, x):
        x = self.encoder(x)
        x = self.decoder(x)
        return x
```

Pytorch for HW1 Networks

Autoencoder

```
for epoch in range(num_epochs):
    for data in dataloader:
        img, _ = data
        img = img.view(img.size(0),
        if torch.cuda.is_available():
            img = img.cuda()
        #
        =====forward=====
        output = model(img)
        loss = criterion(output, img)
        #
        =====backward=====
        optimizer.zero_grad()
        loss.backward()
        optimizer.step()
```

```
class MiniImageCSVDataset:
    def __init__(self, csv_filename, has_labels=True, header=0):
        df = pd.read_csv(csv_filename, header=header)
        data = df.values
        if has_labels:
            self.data = data[:, 1:]
            self.labels = data[:, 0]
        else:
            self.data = data
            self.labels = None

        self.data = self.data.astype('float32')
        self.data = self.data / 255

    def __len__(self):
        return len(self.data)

    def __getitem__(self, idx):
        if self.labels is not None:
            return self.data[idx], self.labels[idx]
        else:
            return self.data[idx]
```



Network Network Toolkits

More PyTorch to come in recitation (and HW7)!

Great Responsibility

Example: Character.AI

The New York Times

Can A.I. Be Blamed for a Teen's Suicide?

The mother of a 14-year-old Florida boy says he became obsessed with a chatbot on Character.AI before his death.

On the last day of his life, Sewell Setzer III took out his phone and texted his closest friend: a lifelike A.I. chatbot named after Daenerys Targaryen, a character from “Game of Thrones.”



By **Kevin Roose**

Reporting from New York

Published Oct. 23, 2024 Updated Oct. 24, 2024

[Leer en español](#)

Example: Medical Imaging Systems

CT scanner without it's covers on



<https://www.youtube.com/watch?v=2CWpZKuy-NE>

Managing Risk

Design Controls

ML Model Cards

Full-cycle Accountability

Responsible Engineering: Design Controls

Responsible Engineering

Design controls

- Documentation
- Verification/Validation
- FMEA
- CAPA

Code of Federal Regulations

ECFR CONTENT	
▼ Title 21 Food and Drugs	Part / Section
▼ Chapter I Food and Drug Administration, Department of Health and Human Services	1 – 1299
▼ Subchapter H Medical Devices	800 – 898
▼ Part 820 Quality System Regulation	820.1 – 820.250
▼ Subpart C Design Controls	820.30
§ 820.30 Design controls.	

<https://www.ecfr.gov/current/title-21/chapter-I/subchapter-H/part-820/subpart-C?toc=1>

Responsible Engineering

Design controls

- Documentation
- Verification/Validation
- FMEA
- CAPA

Reg

— Use Cases

Responsible Engineering

FMEA (Failure Modes & Effects Analysis)

Process Step/Input	Potential Failure Mode	Potential Failure Effects	SEVERITY (1 - 10)	Potential Causes	OCCURRENCE (1 - 10)	Current Controls	DETECTION (1 - 10)
What is the process step, change or feature under investigation?	In what ways could the step, change or feature go wrong?	What is the impact on the customer if this failure is not prevented or corrected?		What causes the step, change or feature to go wrong? (how could it occur?)		What controls exist that either prevent or detect the failure?	

Responsible Engineering

FMEA (Failure Modes & Effects Analysis)

					Severity Scale		
Process Step/Input	Potential Failure Mode	Potential Failure Effects	SEVERITY (1 - 10)	Potential Cause	Adapt as appropriate		
What is the process step, change or feature under investigation?	In what ways could the step, change or feature go wrong?	What is the impact on the customer if this failure is not prevented or corrected?		What causes the step, change or feature to go wrong (how could it occur)?	Effect	Criteria: Severity of Effect	Ranking
					Hazardous - Without Warning	May expose client to loss, harm or major disruption - failure will occur without warning	10
					Hazardous - With Warning	May expose client to loss, harm or major disruption - failure will occur with warning	9
					Very High	Major disruption of service involving client interaction, resulting in either associate re-work or inconvenience to client	8
					High	Minor disruption of service involving client interaction and resulting in either associate re-work or inconvenience to clients	7
					Moderate	Major disruption of service not involving client interaction and resulting in either associate re-work or inconvenience to clients	6
					Low	Minor disruption of service not involving client interaction and resulting in either associate re-work or inconvenience to clients	5
					Very Low	Minor disruption of service involving client interaction that does not result in either associate re-work or inconvenience to clients	4
					Minor	Minor disruption of service not involving client interaction and does not result in either associate re-work or inconvenience to clients	3
					Very Minor	No disruption of service noticed by the client in any capacity and does not result in either associate re-work or inconvenience to clients	2
					None	No Effect	1

<https://goleansixsigma.com/failure-modes-effects-analysis>

Responsible Engineering

Design controls

- Documentation
- Verification/Validation
- FMEA
- CAPA

ECFR CONTENT

§ 820.100 Corrective and preventive action.

- (a) Each manufacturer shall establish and maintain procedures for implementing corrective and preventive action. The procedures shall include requirements for:
 - (1) Analyzing processes, work operations, concessions, quality audit reports, quality records, service records, complaints, returned product, and other sources of quality data to identify existing and potential causes of nonconforming product, or other quality problems. Appropriate statistical methodology shall be employed where necessary to detect recurring quality problems;
 - (2) Investigating the cause of nonconformities relating to product, processes, and the quality system;
 - (3) Identifying the action(s) needed to correct and prevent recurrence of nonconforming product and other quality problems;
 - (4) Verifying or validating the corrective and preventive action to ensure that such action is effective and does not adversely affect the finished device;
 - (5) Implementing and recording changes in methods and procedures needed to correct and prevent identified quality problems;
 - (6) Ensuring that information related to quality problems or nonconforming product is disseminated to those directly responsible for assuring the quality of such product or the prevention of such problems; and
 - (7) Submitting relevant information on identified quality problems, as well as corrective and preventive actions, for management review.
- (b) All activities required under this section, and their results, shall be documented.

<https://www.ecfr.gov/current/title-21/chapter-I/subchapter-H/part-820/subpart-J/section-820.100>

Managing Risk

Design Controls

ML Model Cards

Full-cycle Accountability

ML Model Cards

ML Model Cards

Mitchell, Margaret, et al.

"Model cards for model reporting."

Proceedings of the conference on fairness, accountability, and transparency. 2019.

ML Model Cards

Model Card

- **Model Details.** Basic information about the model.
 - Person or organization developing model
 - Model date
 - Model version
 - Model type
 - Information about training algorithms, parameters, fairness constraints or other applied approaches, and features
 - Paper or other resource for more information
 - Citation details
 - License
 - Where to send questions or comments about the model
- **Intended Use.** Use cases that were envisioned during development.
 - Primary intended uses
 - Primary intended users
 - Out-of-scope use cases
- **Factors.** Factors could include demographic or phenotypic groups, environmental conditions, technical attributes, or others listed in Section 4.3.
 - Relevant factors
 - Evaluation factors

- **Metrics.** Metrics should be chosen to reflect potential real-world impacts of the model.
 - Model performance measures
 - Decision thresholds
 - Variation approaches
- **Evaluation Data.** Details on the dataset(s) used for the quantitative analyses in the card.
 - Datasets
 - Motivation
 - Preprocessing
- **Training Data.** May not be possible to provide in practice. When possible, this section should mirror Evaluation Data. If such detail is not possible, minimal allowable information should be provided here, such as details of the distribution over various factors in the training datasets.
- **Quantitative Analyses**
 - Unitary results
 - Intersectional results
- **Ethical Considerations**
- **Caveats and Recommendations**

Poll 1: ML Model Hunt

Search the web to find the model card for a real-world model

URL

Missing

Model Card

- **Model Details.** Basic information about the model.
 - Person or organization developing model
 - Model date
 - Model version
 - Model type
 - Information about training algorithms, parameters, fairness constraints or other applied approaches, and features
 - Paper or other resource for more information
 - Citation details
 - License
 - Where to send questions or comments about the model
- **Intended Use.** Use cases that were envisioned during development.
 - Primary intended uses
 - Primary intended users
 - Out-of-scope use cases
- **Factors.** Factors could include demographic or phenotypic groups, environmental conditions, technical attributes, or others listed in Section 4.3.
 - Relevant factors
 - Evaluation factors

Full-cycle Accountability

AI Accountability

Slide from Alex London, CMU

I. System of **accountability** across the lifecycle?



Roles & responsibilities at every stage

1. Training in practices and procedures for safe, reliable deployment?
2. Feedback & monitoring for continuous improvement?
3. Contingency planning?

1. How are models validated?
 - a. Confounding & validity?
 - b. Parameters of reliable use?

Deployment

Task

Testing

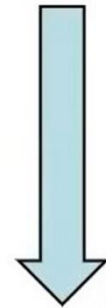
Data

Model

What policy is the system implementing?

1. What are we trying to do?
2. Definition of success / failure?
 - a. Metrics & comparator
3. Whose interests impacted?
 - a. Benefits / risks / fairness / autonomy
 - b. Strategies to promote / reduce, mitigate

1. Do we have permission / access to data that supports use case?
 - a. Suitable ontology?
 - b. Representative?
 - c. Direct measure of relevant features or use of proxies?
 - i. Are proxies valid?



Procedures & expectations



Reporting, coordination & oversight.

