DEEP ANALYSES OF MESSY PERIOPERATIVE DATA
Parisa Rashidi, PhD
UF
HOW TO MAKE PERIOPERATIVE DATA SCIENCE MORE AFFECTIVE

THIS MORNING
Assembling a Deep EHR
Tighe
Deep Learning Mechanisms
Rashidi
Applying Deep Insights
Price

DISCLOSURES
• I have no financial conflicts of interest to report
• Research support from NIH and NSF
OBJECTIVES

1. Learn about limitations of conventional models
2. What do deep learning models look like?
3. Deep learning in action

AI, MACHINE LEARNING, DEEP LEARNING

MACHINE LEARNING VS. STATISTICS

- Machine learning deals with high-dimensional data
  - Data can be unstructured
    - E.g., video, images, natural language data
MACHINE LEARNING TASKS

Supervised Learning
- Input: data, labels
- Output: label prediction for new data points

Unsupervised Learning
- Input: data
- Output: discover the latent structure of data points

Reinforcement Learning
- Input: data, occasional feedback
- Output: learned policy as series of actions

CENTRAL PROBLEM

- To transform data to learn useful representations of the input data (encode), i.e. representation learning

CONVENTIONAL MACHINE LEARNING MODELS

- There are already many different machine learning (ML) and statistical learning models
  - Logistic regression
  - Support Vector Machines (SVM)
  - Random forests
LIMITATIONS OF CONVENTIONAL ML MODELS

- Shallow models that are unable to model highly complex data
  - E.g. video or image analysis
- Features need to be provided to the model, but what features for each problem?
  - Feature engineering

Conventional Machine Learning

Input → Manual Feature Extraction → Classification → Output (Benign, Malignant)

Deep Learning

Input → Automated Feature Extraction & Classification → Output (Benign, Malignant)

NEURAL NETWORKS HISTORY

GOES BACK TO 1940, WITH SEVERAL DARK AI WINTERS

ELECTRONIC BRAIN - 1943

- Warren McCulloch and Walter Pitts published the first concept of a simplified brain cell (1943).
  - A simple logic gate with binary outputs
  - Multiple signals arrive at the dendrites.
  - Signals then integrated into the cell body.
  - If the accumulated signal exceeds a certain threshold, a 1 is generated and passed by the axon.

\[ \Phi(x) = \begin{cases} 
1 & \text{if } x \geq \theta \\
-1 & \text{otherwise}
\end{cases} \]
PERCEPTRON - 1957
• The basic processing element (Rosenblat, 1957)
• Associated with each input is a connection weight $w_i$
• Rosenblatt proposed an algorithm to automatically learn the optimal weights

The Mark I Perceptron machine was the first implementation of the perceptron algorithm. The machine was connected to a camera and to two 1024-element digital memory units. The computation inside the machine was performed by a group of input features that allowed comparisons with different combinations of input features. To the right of that are arrays of potentiometers that implemented the adaptive weights.

ADALINE - 1960
• Proposed by Widrow & Hoff
• Illustrates the key concept of defining and minimizing a cost function
• The groundwork for understanding more advanced techniques
• Key difference from perceptron
  • Compared to perceptron, we can use a continuous function to compute the error
  • Differentiable

$E(w) = \frac{1}{2} \sum_{i} (y_i - \theta(z))^2$


Winter is coming

Winter is coming

Winter is coming

Winter is coming

Winter is coming

Winter is coming

Winter is coming

Winter is coming

Winter is coming

Winter is coming
BACKPROPAGATION - 1986

- Proposed by Rumelhart, Hinton, and Williams
- Paved the way for training deep networks

Rumelhart, Hinton, Williams (1986)


Winter is coming... Again

DEEP LEARNING IS REBORN

REIGNITED DEEP LEARNING IN 2006

Ruslan Salakhutdinov and Geoffrey Hinton, 2006
FIRST IMPRESSIVE RESULTS

STARTING IN 2010-2012

REGRESSION

\[ y = \sum_{j=1}^{k} w_j x_j + w_0 = \mathbf{w}^T \mathbf{x} \]

CLASSIFICATION

\[ \Phi(x) = \begin{cases} 1 & \text{if } x \geq \theta \\ -1 & \text{otherwise} \end{cases} \]
**PERCEPTRON ONLINE LEARNING**

- We do not write the error function over the whole sample at once
  - But on individual instances at each step

1. Start from random weights
2. At each iteration, adjust parameters a little bit to minimize error based on current input

\[
\Delta w_j = \eta(y - \hat{y}) x_i
\]

**FULLY CONNECTED NETWORKS (FCN)**

- Composed of multiple hidden layers
  - Every neuron in layer \( i \) is connected to every other neuron in layer \( i + 1 \)

Input  | Hidden Layers  | Output
--- | --- | ---
\( x_1 \) | \( W_{11} \) | \( y_1 \)
\( x_2 \) | \( W_{12} \) | \( y_2 \)
\( \ldots \) | \( \ldots \) | \( \ldots \)
\( x_n \) | \( W_{n1} \) | \( y_n \)

GRADIENT DESCENT

• A simple, yet powerful optimization algorithm
• Think of it as climbing down a hill until a local or global minimum is reached.
• In each iteration, we take a step away from the gradient where the step size is determined by the value of the learning rate as well as the slope of the gradient.

$$\Delta w = -\eta \nabla E(w)$$ Update

$$\frac{dE}{dw} = -\nabla E \left( \sum_i (y_i - z_i)^2 \right)$$ Gradient

FULLY CONNECTED NETWORKS (FCN)

• Backpropagation algorithm is key

$$h_k = \sigma \left( \sum_j x_j w_{jk} + b_j \right)$$

1. Sample a batch of data
2. Forward through network, compute loss
3. Backpropagate to calculate the gradients
4. Update the weights using the gradients

DEEP LEARNING

• Learning successive layers of increasingly meaningful representations
  – Mostly learned via models known as neural networks
INCREASING NETWORK SIZE

- It is not just about the number of neurons.
- Brain is much more complex.

Many Deep Learning Models

- CNN layers extract patches from input feature maps and apply the same transformation to all patches, producing an output feature map.
- Tool of choice in almost every vision tasks these days
- Many variants
**CNNs Have Been Around for a Long Time**

- Goes back to 1989, first successful application LeNet
- In recent years, it scaled up due to hardware improvements, algorithmic advances, and data availability

![LeNet (1989)](image)

**Impressive Progress**

- Error rates for ImageNet challenge have fallen from 28.5% to below 2.5% since 2010
- Many biomedical imaging applications
  - Image segmentation & registration
  - Automatic labeling (e.g., lesion detection)
  - Computer-aided diagnosis

![Image](image)

**Recurrent Neural Networks (RNN)**

- CNNs useful when data has spatial structure
- Recurrent neural networks (RNNs) useful for sequentially ordered data
  - Time series, natural language
- Popular RNN variants
  - Long short-term memory (LSTM)
  - Gated recurrent unit (GRU)
RECURRENT NEURAL NETWORKS (RNN)

- Physiological signals can be processed by LSTMs

AUTOENCODERS

- Typically used for unsupervised representation learning
- To encode input into a lower dimensional space
  - Reconstruction and decoding

AUTOENCODERS

Deep Patient: An Unsupervised Representation to Predict the Future of Patients from the Electronic Health Records
COMBINING DATA TYPES

WE CAN CREATE MULTI-INPUT AND MULTI-OUTPUT MODELS


REINFORCEMENT LEARNING

- Which actions to take to maximize cumulative reward
- Tradeoff between exploration & exploitation

\[ a = \pi(s) \]

SUCCESS STORIES

- Near-human-level
  - Image classification, speech recognition, handwriting transcription
- Improved
  - Machine translation, text-to-speech, autonomous driving
- Superhuman
  - Go playing
RECENT PROGRESS - 2017

- Capsule net
- One model for all
- Alpha-zero


OUR USE CASES

- Time series: LSTM – SOFA Score Prediction
- Video feed: CNN – Functional status detection
- Text: LSTM – Mental health, clinical notes

AI FUTURE

- Human-level general intelligence: still far far away
- Many opportunities for AI in medicine
- Best to moderate our expectations, there is a short-term hype, but there is also a long-term vision
WHERE TO GO FROM HERE?

• Our recent survey paper on use of deep learning with EHR data

• Look for the latest developments on ArXiv

• Many open software tools
  - Keras
  - TensorFlow
  - CNTK
  - Theano
  - MXNet