RUTGERS
UNIVERSITY | NEW BRUNSWICK

Towards Explainable Artificial Intelligence

Yongfeng Zhang
Department of Computer Science, Rutgers University
yongfeng.zhang@rutgers.edu
The WISE Lab at Rutgers
https://wise.cs.rutgers.edu/

6/20/2020

Outline

- Basic Concepts and History of AI
- How did the Explainable AI Problem Emerge
- Why should we Care about the Problem
- Different Explainable AI models
- Summary and Future Directions

About AI and ML

- AI ≠ ML, AI ⊃ ML [1]

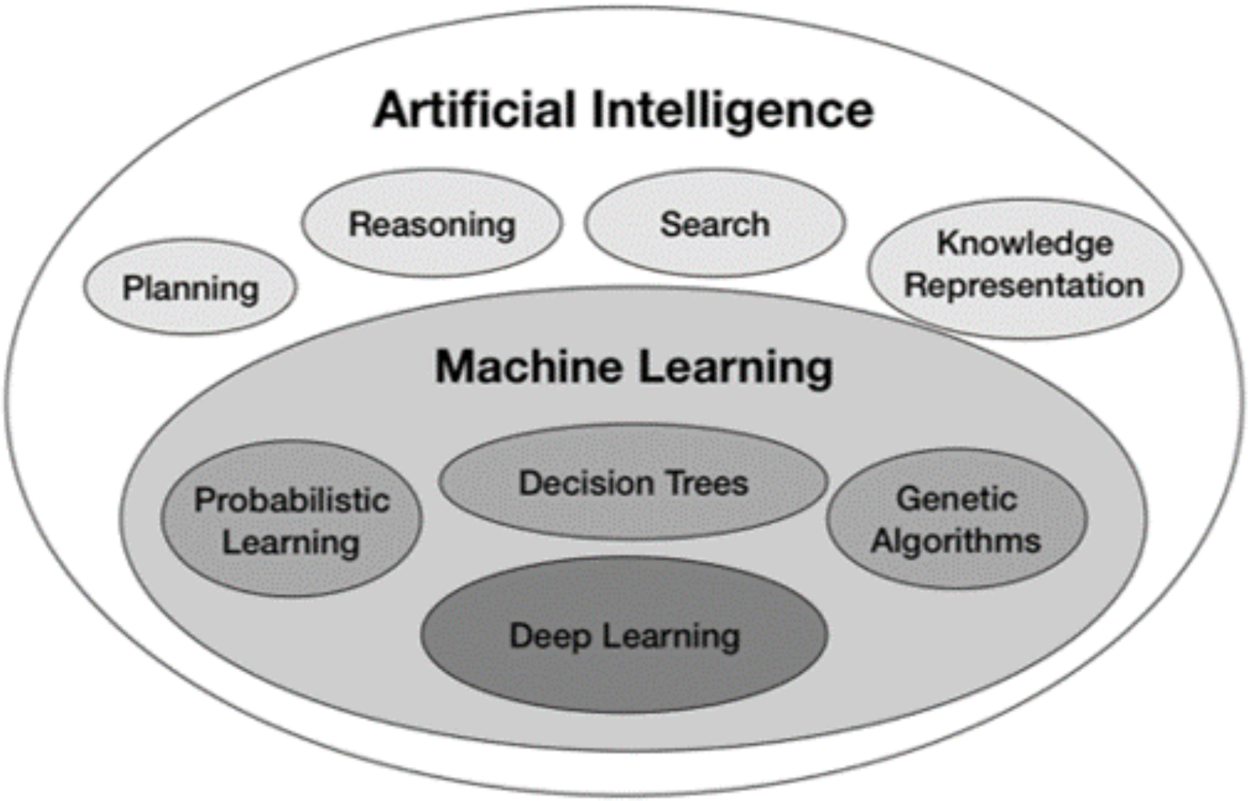


Image from: [1] Marcus G, Davis E. (2019). Rebooting AI: Building artificial intelligence we can trust. Pantheon; 2019 Sep 10.

A (very rough) history of AI research

- Symbolic Reasoning Approach to AI
  - Mid-1950s to late 1980s

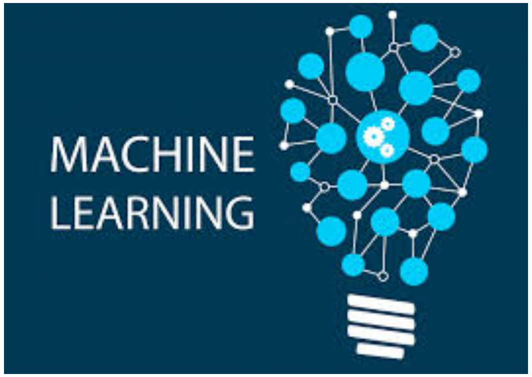
GOFAI

means

Good Old Fashioned Artificial Intelligence

>Representative methods:
  Graph search algorithms (e.g., A*),
  Production rules, Knowledge reasoning, etc.
>Representative systems:
  Expert systems (If-Then production rules)
  Chess game AI (IBM DeepBlue)

- Machine Learning Approach to AI
  - Early 1990s to date



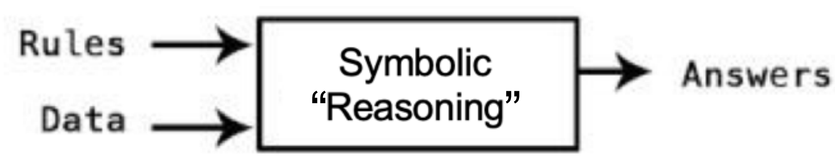
>Representative methods:
  Support vector machines, Logistic regression, Matrix factorization, Deep neural networks
>Representative systems:
  Recommender systems, Computer vision, NLP, etc.

Symbolism vs Connectionism - A comparison

- a.k.a. Rationalism vs Empirism approaches to AI

Symbolism/Rationalism

A top-down design approach

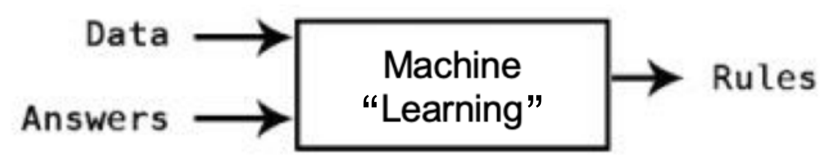


Advantages:
Accurate decision
Highly explainable & human readable

Disadvantages:
Extensive expert human efforts
Difficult to generalize
(handle expected inputs)

Connectionism/Empirism

A bottom-up design approach



Advantages:
Less human efforts
Great generalization ability

Disadvantages:
Decision are usually approximate
Difficult to explain (black-box model)

How ML Approaches Advanced to Date

- From shallow, to deep, and deeper
- Too many models
  - We will introduce some representative methods to build a timeline

- Early approaches, easily explainable
  - Linear Regression
  - Support Vector Machines (non-linearity helps)
- Bi-linear models, somewhat explainable
  - Matrix Factorization
- ML + Big Data => Deep Models, hardly explainable
  - Two directions: Representation Learning vs Similarity Learning
  - A.k.a: Feature Learning vs Function Learning

Linear Regression

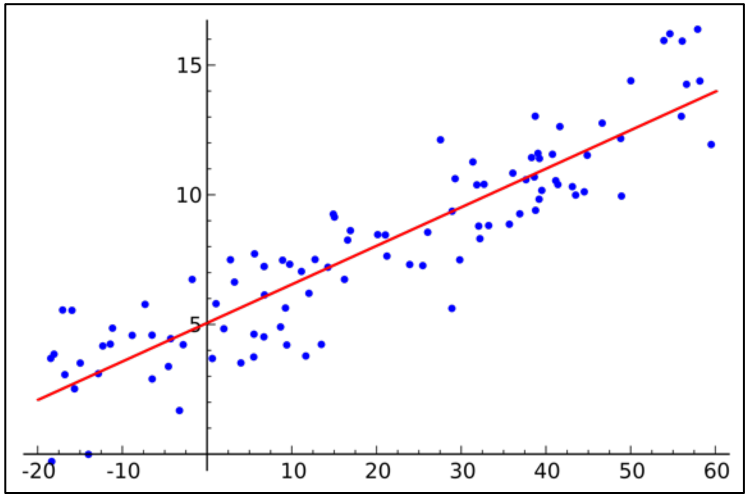


Image from: https://en.wikipedia.org/wiki/Linear_regression

$$y = \boldsymbol{w}^T \boldsymbol{x} + b$$

x: a high-dimensional feature vector
w: weight vector to be learned
b: bias scalar to be learn
y: the output model prediction

w and b can be easily "learned" by some "cost function" (e.g., minimizing the square loss)

$$\min_{\boldsymbol{w},b} \sum_i (y - \hat{y})^2 \quad , \hat{y} = \boldsymbol{w}^T \boldsymbol{x} + b$$

Application:
Widely applied in many research areas

Pros:
Easily explainable: the learned $\boldsymbol{w}$ vector actually tells us the influence of each dimension (i.e., factor) in $\boldsymbol{x}$

e.g., in student class performance prediction, dimensions in $\boldsymbol{x}$ could be:
x=[GPA, attendance, age, gender, major, etc.]
e.g. [3.5, 70%, 20, 1/0, 1/0, …]
y = 1/0 (Pass/Fail)

After model learning, the learned $\boldsymbol{w}$ could be:
w = [0.6, **0.8**, 0.5, **0.01**, 0.6, …]

Very successful in Econometrics

Cons:
(1) Needs manual feature design/collection
(2) Limited expressiveness power

How to Solve the Two Major Problems?

- 1. Needs manual feature design/collection
- 2. Limited expressiveness power

- A Typical ML model (no matter simple or complex) consists two parts:

$$y = \boldsymbol{w}^T \boldsymbol{x} + b$$

$\Uparrow$

Linear (inner product)

$\Uparrow$

The **x** vector

Output/Prediction

$\Uparrow$

Function (Simple or complex)

$\Uparrow$

Features (usually as vectors)

Why don't we use more "expressive" functions?
Similarity Learning!
(aka Metric Learning)

Why don't we "learn" the features?
Representation Learning!

What happened in the last 30 years? (1990s to date)

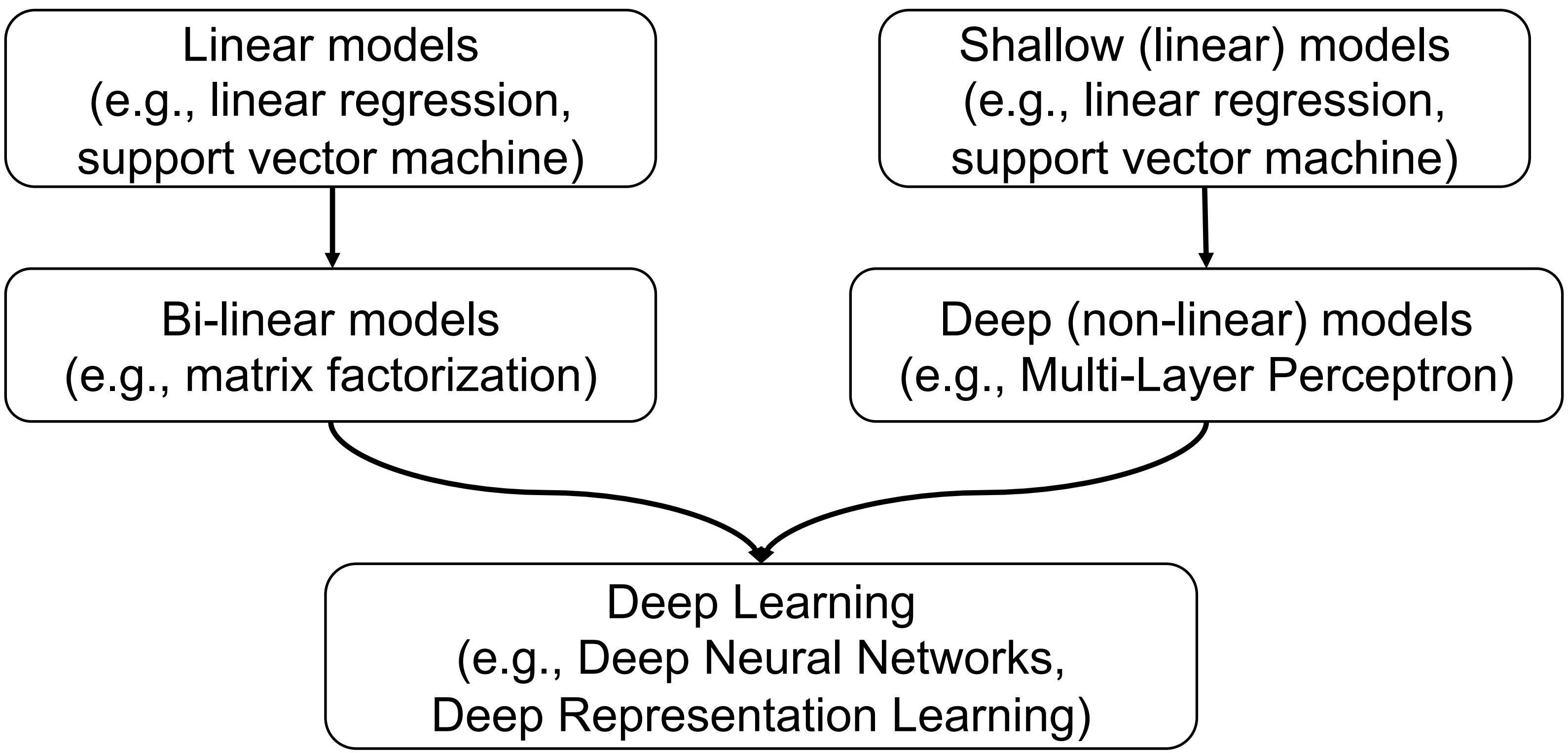
- Two lines of research in the AI community

Representation Learning

Better Features!

Similarity/Metric Learning

Better Functions!



Smarter AI Round 1: Automatic Feature Learning!

- To solve problem 1 of the "toy" linear regression model
  - i.e., Needs manual feature design/collection

$$y = \boldsymbol{w}^T \boldsymbol{x} + b$$

- Solution: Representation Learning

- Use Matrix Factorization as an example
  - Still use linear inner-product function, but "learn" the features

Recommender System: A Typical application of MF

- A partially observed matrix

Items

Users

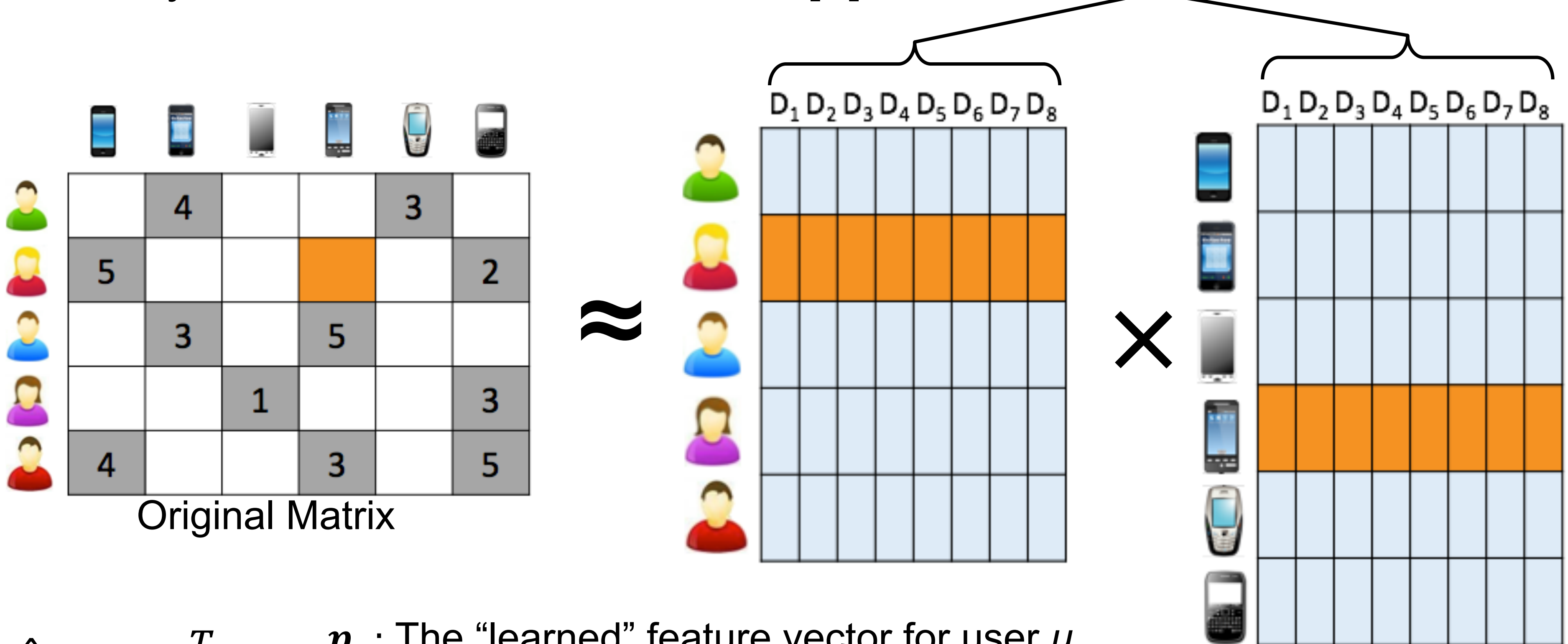
?	4	?	?	3	?
5	?	?	?	?	2
?	3	?	5	?	?
?	?	1	?	?	3
4	?	?	?	?	2

A key task:
Predict the missing ratings

Predict the Missing Ratings

Matrix Factorization for Recommendation

- Key idea of latent factor models [2]



$$\hat{r}_{ui} = \boldsymbol{p}_u^T \boldsymbol{q}_i$$

$\boldsymbol{p}_u$: The “learned” feature vector for user *u*
$\boldsymbol{q}_i$: The “learned” feature vector for item *i*

$$\min_{\boldsymbol{p},\boldsymbol{q}} \underbrace{\sum_{(u,i)\in R} (r_{ui} - \boldsymbol{p}_u^T \boldsymbol{q}_i)^2}_{\text{Goodness of fit}} + \underbrace{\lambda_1 \sum_u \| \boldsymbol{p}_u \|^2 + \lambda_2 \sum_i \| \boldsymbol{q}_i \|^2}_{\text{Regularization}}$$

Compare:

$$y = \boldsymbol{w}^T \boldsymbol{x} + b$$

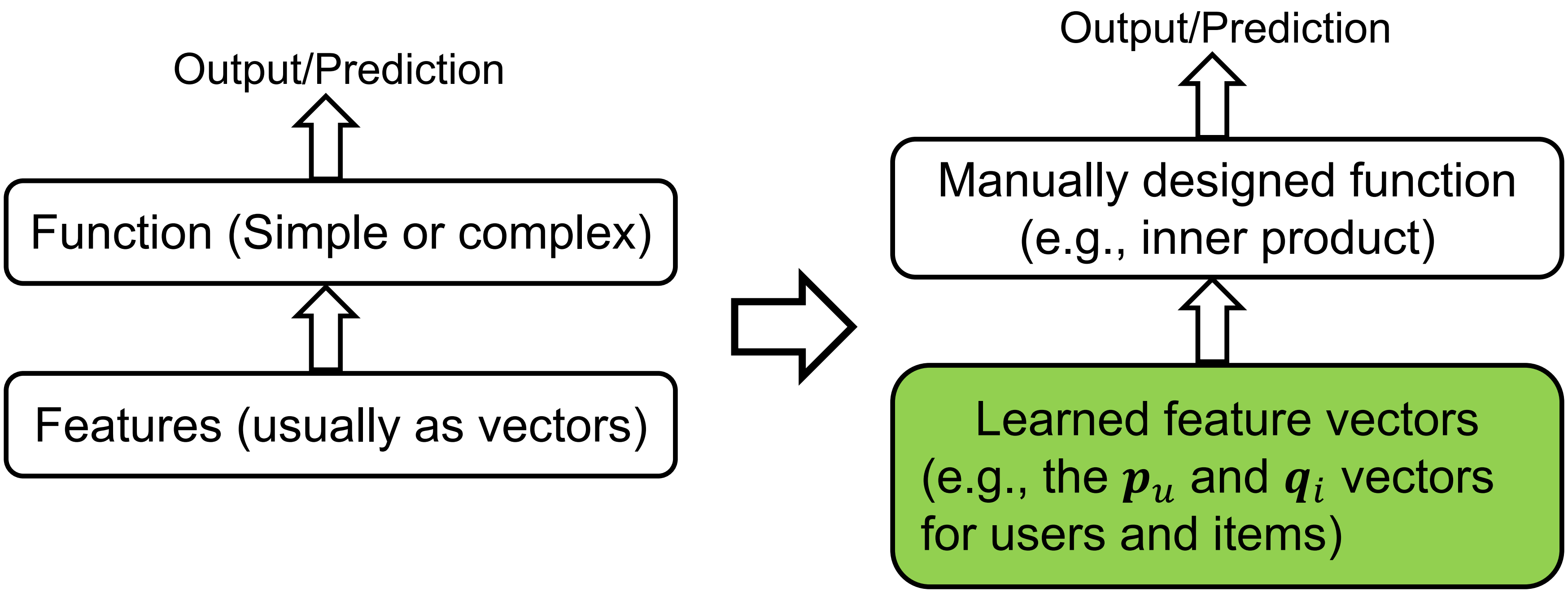
[2] Koren, Y., Bell, R. and Volinsky, C. (2009). Matrix factorization techniques for recommender systems. Computer, 42(8), pp.30-37.

Personalized Recommendations

4.5	4	2.7	4.7	3	3.6
5	3.8	1.2	4.2	3.5	2
3.3	3	2.6	5	2.8	3.7
4.2	4.5	1	2.4	3.2	3
4	3.7	3.9	3.2	4.6	2

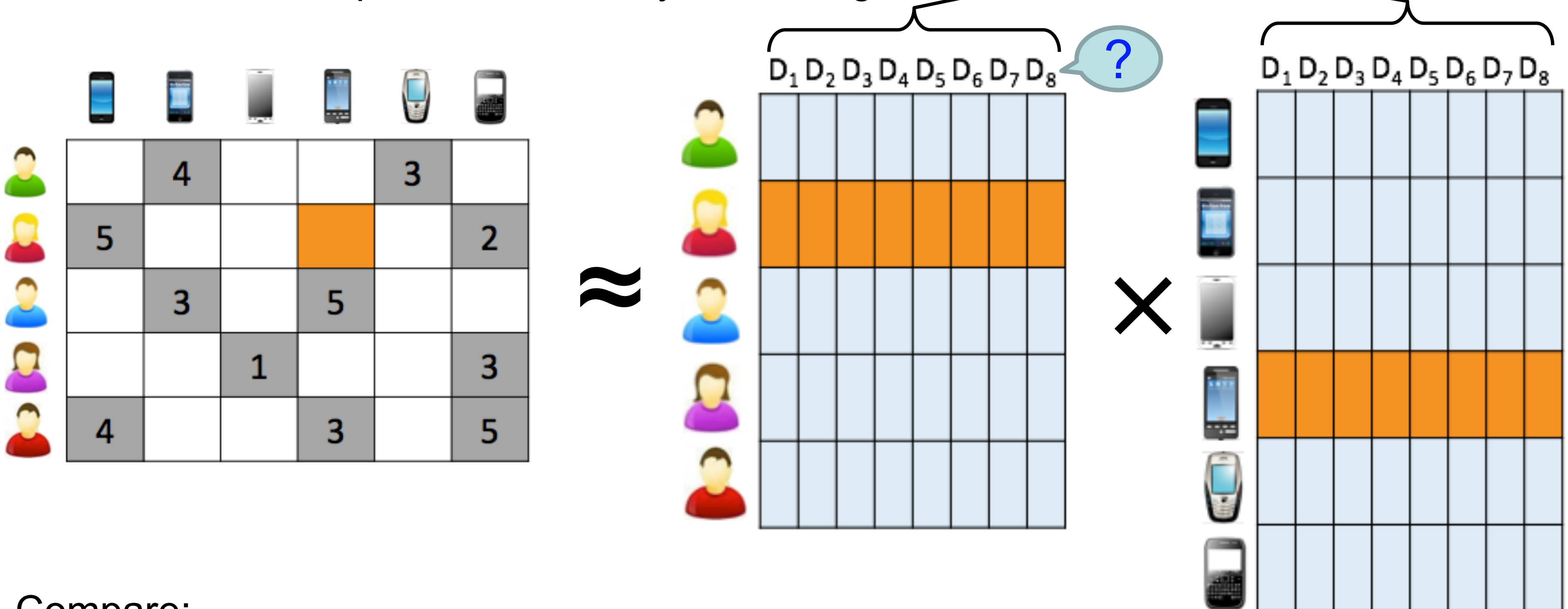
RUTGERS

Take away: Representation Learning



Less Explainability!

- The meaning of each dimension of the learned feature vectors are unknown to us
- **Latent** factor models
  - More accurate (directly minimize prediction error)
  - But less explainable (due to the “latent” factors)
  - Unable to explain to users why something is recommended



Compare:

$y = \boldsymbol{w}^T \boldsymbol{x} + b$ **x**=[GPA, attendance, age, gender, major, etc.]

Smarter AI Round 2: Automatic Function Learning!

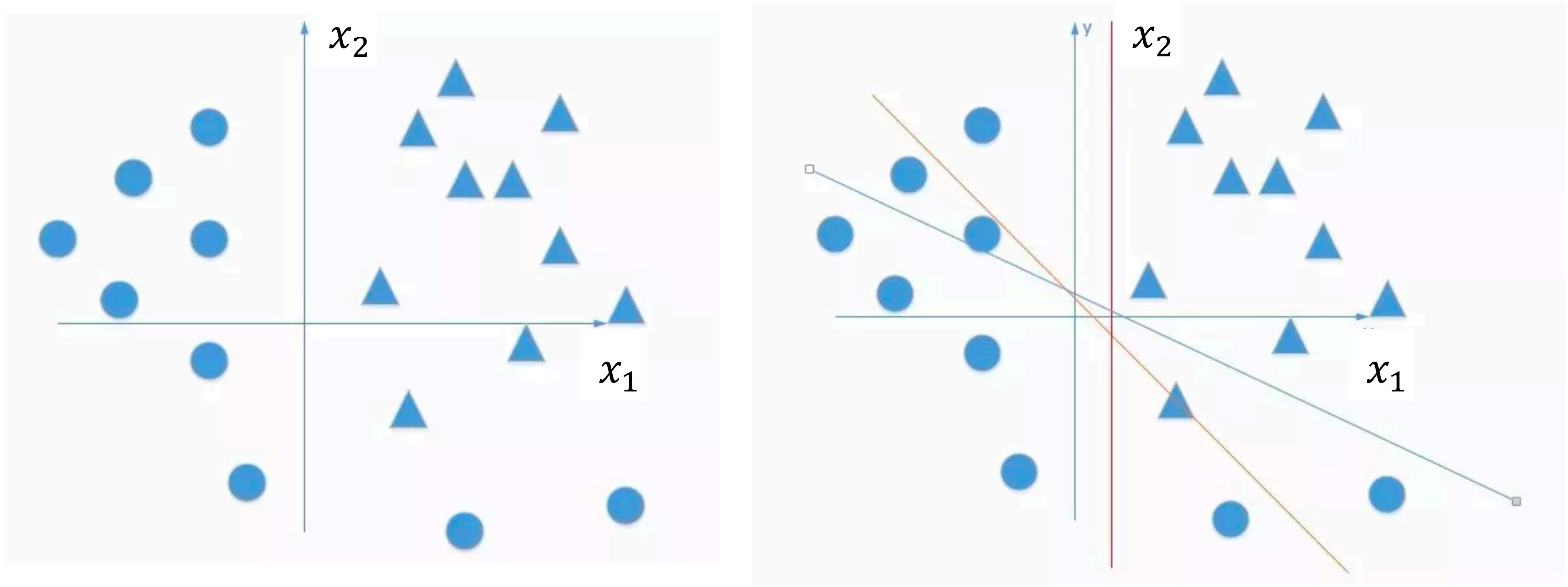
- To solve problem 2 of the "toy" linear regression model
  - i.e., Limited expressiveness power

$$y = \boldsymbol{w}^T \boldsymbol{x} + b$$

- Solution: Similarity/Metric Learning
  - To learn more powerful functions
  - How? Two key ingredients
    - 1. Non-linearity
    - 2. Deeper functions
  - (Side comment: Both leads to difficulty in explainability)

- Use Multi-Layer Perceptron (MLP) as an example
  - The power of non-linear and deep models

A Classification Problem



$$\boldsymbol{x} = [x_1, x_2]$$

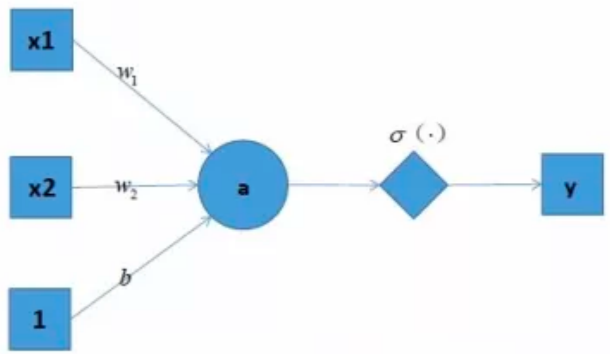
$$y = \boldsymbol{w}^T \boldsymbol{x} + b$$

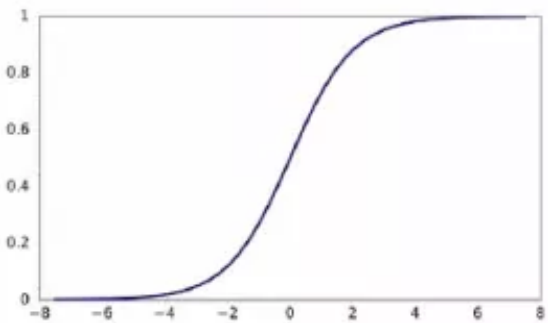
●: y=0
▲: y=1

$$\min_{w,b} \sum_i (y - \hat{y})^2 \quad , \hat{y} = \boldsymbol{w}^T \boldsymbol{x} + b$$

Reference: https://www.cnblogs.com/LittleHann/p/11385756.html

Perceptron with non-linear activation function





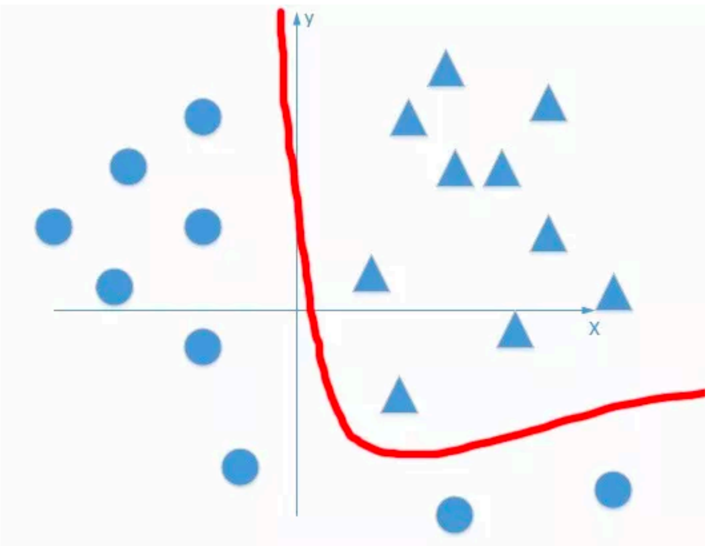
$\sigma(\cdot)$ is a non-linear activation function, sigmoid was the most popular one,

$$\sigma(y) = \frac{1}{1+e^{-y}}$$

$$y = \sigma(\boldsymbol{w}^T \boldsymbol{x} + b)$$

●: y=0
▲: y=1

$$\min_{\boldsymbol{w},b} \sum_i (y - \hat{y})^2 \quad , \hat{y} = \sigma(\boldsymbol{w}^T \boldsymbol{x} + b)$$

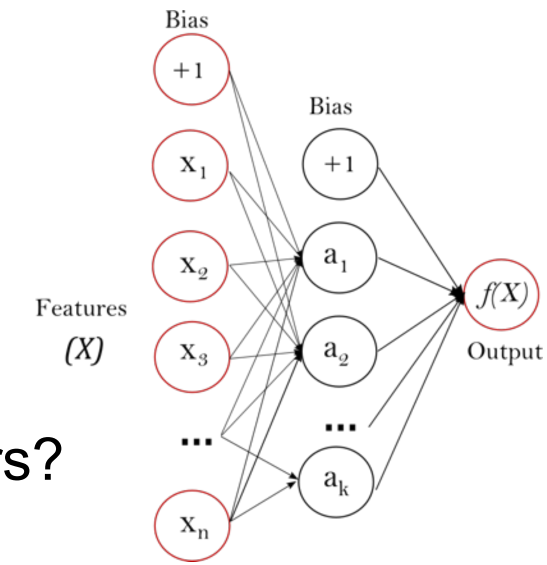


More expressive than a simple linear function

Reference: https://www.cnblogs.com/LittleHann/p/11385756.html

RUTGERS

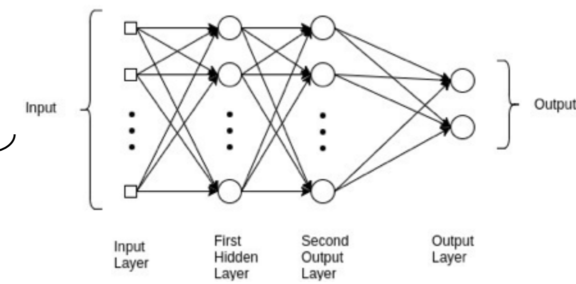
What is $y = \sigma(\boldsymbol{w}^T\boldsymbol{x} + b)$?



- This is actually a (one-layer) neural network
  - with a 1-dimential hidden layer

- What if we extend one-layer to multiple layers?

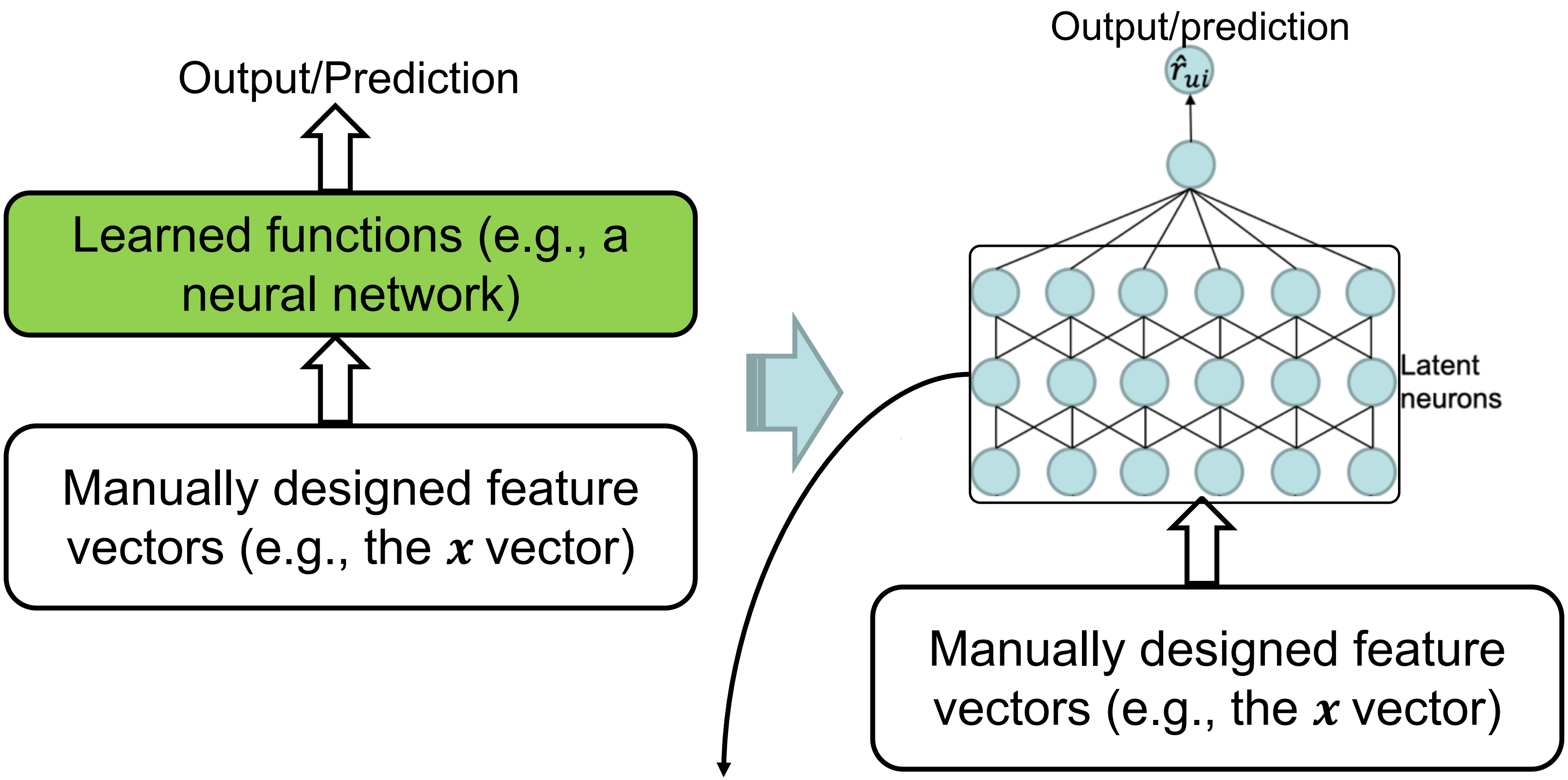
$$y = \sigma(\underbrace{\boldsymbol{w}_3^T\sigma(\underbrace{\boldsymbol{W}_2^T\underbrace{\sigma(\boldsymbol{W}_1^T\boldsymbol{x} + \boldsymbol{b}_1)}_{1^{st}\text{ Layer}} + \boldsymbol{b}_2)}_{2^{nd}\text{ Layer}} + b_3)}_{3^{rd}\text{ Layer}}$$



- A (very nice) Theorem
  - Universal Approximation Theorem (UAT) [3,4,5]
  - A network containing a finite number of neurons can approximate arbitrary well any real-valued continuous functions on compact subsets of $\mathbf{R}^n$.
  - Neural networks are very powerful functions!

[3] Balazs Csanad Csaji (2001). Approximation with Artificial Neural Networks. Faculty of Sciences, Etvs Lornd University, Hungary 24(48:7).
[4] Cybenko, G. (1989). Approximations by superpositions of sigmoidal functions. Mathematics of Control, Signals, and Systems, 2(4):303–314.
[5] Hornik, Kurt (1991). Approximation capabilities of multilayer feedforward networks. Neural networks, 4(2): 251-257.

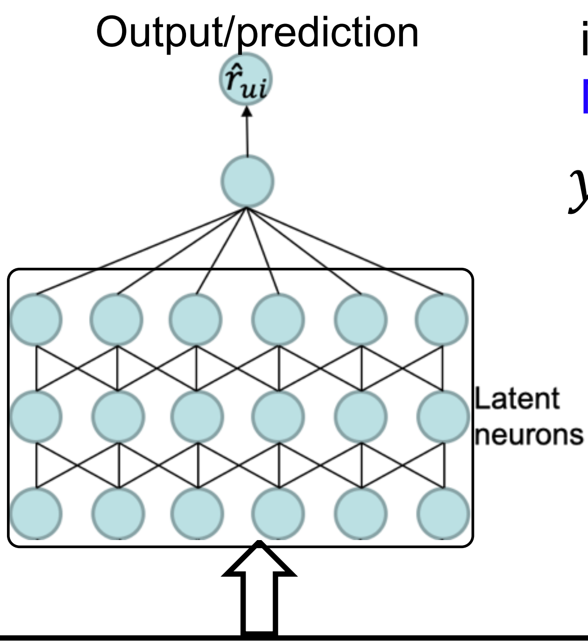
Take away: Similarity Learning



We don’t know what is the correct prediction function, but we don’t care, because whatever it is, our neural network can approximate it.

(However, it requires large amount of training data to learn the many parameters in the neural network, that’s why Deep Learning didn’t prosper until early 2010s.
In the 2000s and 2010s, the prospering of the Internet brings us **Big Data**)

Brings More Problems on Explainability!



Manually designed feature vectors (e.g., the $\boldsymbol{x}$ vector)

Non-linearity: weights in **W** no longer tell us the influence of factors in **x**.
Multiple layers: Influence of **x** changes across layers.

$$y = \sigma(\boldsymbol{w}_3^T \sigma(\boldsymbol{W}_2^T \sigma(\boldsymbol{W}_1^T \boldsymbol{x} + \boldsymbol{b}_1) + \boldsymbol{b}_2) + b_3)$$

The “learned” function may work well,
But it’s a **black box**, hard to explain!

However, sometimes we really want to know the meaning of the function, and why the model makes a certain output for a certain input.

That’s very important in many scenarios.

Modern ML: Representation Learning + Similarity Learning (End-to-End learning)

Output/Prediction

⇧

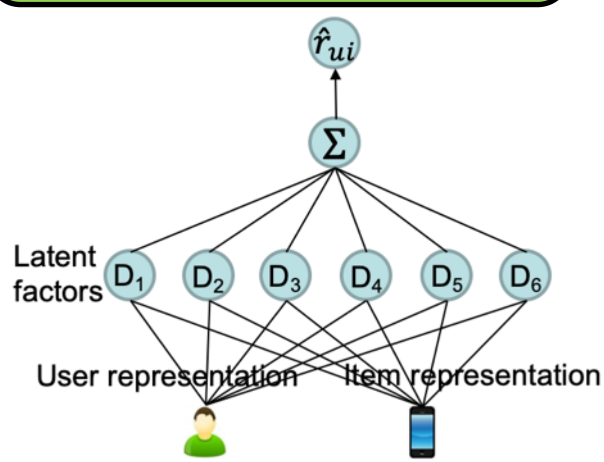
Learned functions (e.g., a neural network)

Hard to explain due to non-linearity, multiple layers

⇧

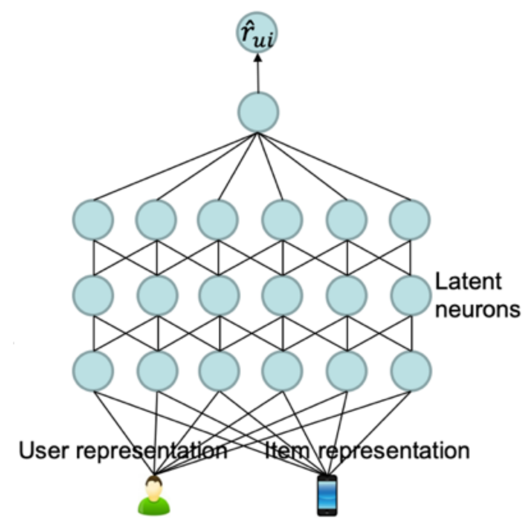
Learned feature vectors (e.g., the $\boldsymbol{p}_u$ and $\boldsymbol{q}_i$ vectors for users and items)

Hard to explain due to the unknown meaning of the learned "latent" feature vectors



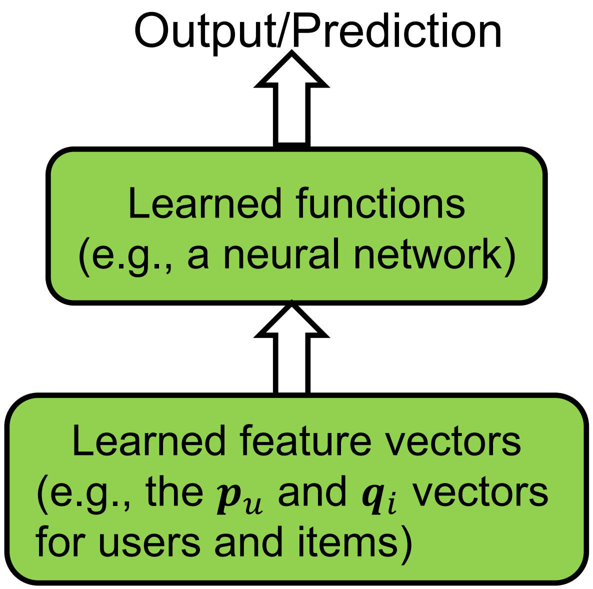
MF Shallow, Bi-Linear

⇨



DNN Deep, Non-Linear

Explainable Artificial Intelligence (XAI)



Hard to explain due to non-linearity, multiple layers

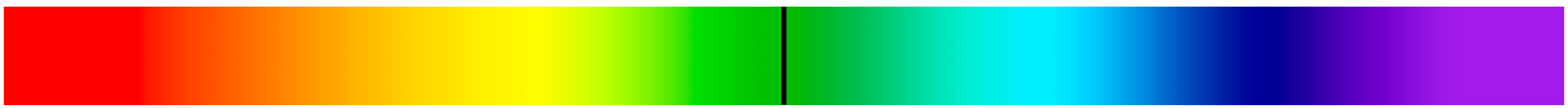
Hard to explain due to the unknown meaning of the learned “latent” feature vectors

We want to know **WHY** the model gives certain output for certain inputs.

Why XAI Matters

- AI/ML has been widely used in many research disciplines.
  - A (very rough) spectrum of research discipline system

Human/Arts Nature/Science



… Arts Literature Sociology Education Economics CS/AI Engineering Biology Medical Chemistry Physics …

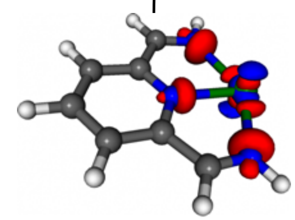
Computer Painting


WHAT STARTS OUT AS TRIUMPH SOON BECOMES MANIPULATED INTO A CACOPHONY OF FUTILITY, LEAVING ONLY A SENSE OF DREAD AND THE UNLIKELIHOOD OF A NEW ORDER.

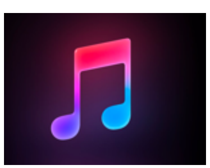
BY TEXT GENERATOR

Computer-generated poems/computer writing

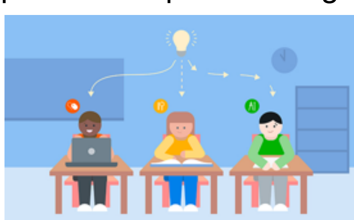
Computational Biology



Computational Chemistry



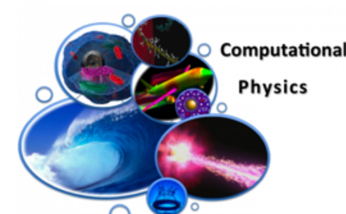
Computer Music Composing



Personalized education based on ML

Quantitative and Computational Econ

Computational drug discovery



Computational Physics

RUTGERS

Why XAI Matters

- In almost all of the areas
  - We not only want to know that a model works (e.g., make accurate predictions)
  - We also want to know **why** it works (e.g., why the model produce this output, why the model produced this drug structure)

- Even more important in **high-stake applications** related to health, safety, and law

Healthcare

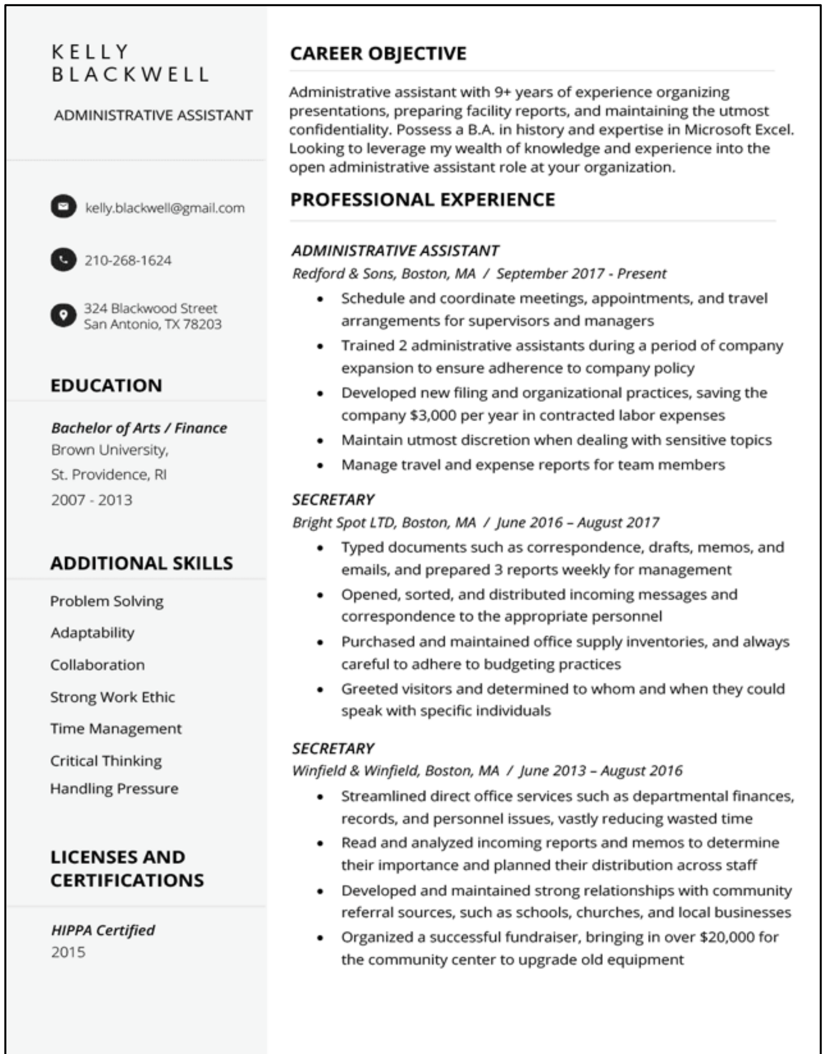
Self Driving

Legal Assistants

  - Errors/bias may cause severe loss in life, money, and reputation

- Explanations help humans to make better decisions
- Also help scientists in making more insightful science discoveries

Case Study I: XAI in AI-assisted Hiring Decision



KELLY BLACKWELL

ADMINISTRATIVE ASSISTANT

kelly.blackwell@gmail.com

210-268-1624

324 Blackwood Street
San Antonio, TX 78203

EDUCATION

Bachelor of Arts / Finance
Brown University,
St. Providence, RI
2007 - 2013

ADDITIONAL SKILLS

Problem Solving
Adaptability
Collaboration
Strong Work Ethic
Time Management
Critical Thinking
Handling Pressure

LICENSES AND CERTIFICATIONS

HIPPA Certified
2015

CAREER OBJECTIVE

Administrative assistant with 9+ years of experience organizing presentations, preparing facility reports, and maintaining the utmost confidentiality. Possess a B.A. in history and expertise in Microsoft Excel. Looking to leverage my wealth of knowledge and experience into the open administrative assistant role at your organization.

PROFESSIONAL EXPERIENCE

ADMINISTRATIVE ASSISTANT
Redford & Sons, Boston, MA / September 2017 - Present

- Schedule and coordinate meetings, appointments, and travel arrangements for supervisors and managers
- Trained 2 administrative assistants during a period of company expansion to ensure adherence to company policy
- Developed new filing and organizational practices, saving the company $3,000 per year in contracted labor expenses
- Maintain utmost discretion when dealing with sensitive topics
- Manage travel and expense reports for team members

SECRETARY
Bright Spot LTD, Boston, MA / June 2016 – August 2017

- Typed documents such as correspondence, drafts, memos, and emails, and prepared 3 reports weekly for management
- Opened, sorted, and distributed incoming messages and correspondence to the appropriate personnel
- Purchased and maintained office supply inventories, and always careful to adhere to budgeting practices
- Greeted visitors and determined to whom and when they could speak with specific individuals

SECRETARY
Winfield & Winfield, Boston, MA / June 2013 – August 2016

- Streamlined direct office services such as departmental finances, records, and personnel issues, vastly reducing wasted time
- Read and analyzed incoming reports and memos to determine their importance and planned their distribution across staff
- Developed and maintained strong relationships with community referral sources, such as schools, churches, and local businesses
- Organized a successful fundraiser, bringing in over $20,000 for the community center to upgrade old equipment

- Big companies receive thousands of applications for a position
- Impossible to manually screen every resume
- Use ML (e.g., Natural Language Processing) algorithms for pre-screening
- Input: texts in your resume
- Output: Pass or not
- You will have a chance of next-round interview only if the machine (i.e., algorithm) agrees
- Explanations of the machine decision is important!

Case Study II: XAI in Science Discovery

- A traditional paradigm of science discovery

- Step 1: Make an observation
- Step 2: Ask a question
- Step 3: Form a hypothesis, or testable explanation
- Step 4: Make a prediction based on the hypothesis
- Step 5: Test the prediction
- Step 6: Iterate: use the results to make new hypotheses or predictions

Kepler's Law of Planetary Motion

We can
Obverse it!

We can
Predict it!

Tycho Brahe (1546-1610)
Demark astronomer

Good at astro-observation
Observed and recorded a lot of data about how planets circle around the sun.

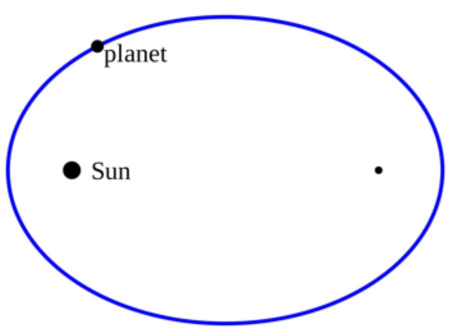
Johannes Kepler (1571-1630)
German astronomer, student of Tycho Brahe.

Analyzed Tycho's data, and discovered a rule hidden in the data.
The "Kepler's laws of planetary motion":

$$\frac{\tau^2}{r^3} = K$$

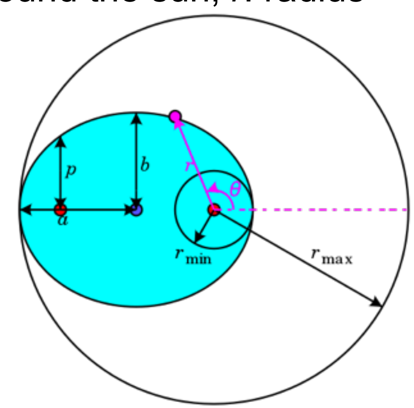
τ: period of circling around the sun, r: radius

Time, Position
1, (a, b)
2, (c, d)
3, (e, f)
…



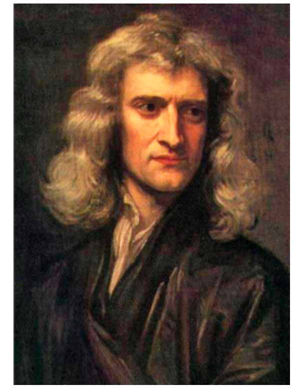
Time, Position
1, (a, b)
2, (c, d)
3, (e, f)
…

$$\frac{\tau^2}{r^3} = K$$



Is the Story Over? No!

We can
Predict it!



We **Understand** it!
We know **Why**!

Johannes Kepler (1571-1630)
German astronomer, student of Tycho Brahe.

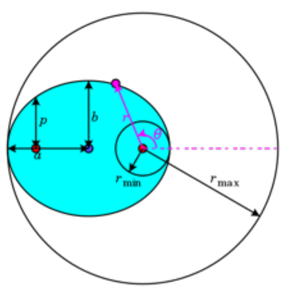
Analyzed Tycho's data, and discovered a rule hidden in the data.
The "Kepler's laws of planetary motion":

$$\frac{\tau^2}{r^3} = K$$

τ: period of circling around the sun, r: radius

Time, Position
1, (a, b)
2, (c, d)
3, (e, f)
…

$$\frac{\tau^2}{r^3} = K$$



Isaac Newton (1643-1727)
English mathematician, physicist, astronomer, theologian, and author.

Proposed the Newton's law of universal gravitation + differential calculus:

Naturally derives out the Kepler's laws of planetary motion!

$\frac{\tau^2}{r^3} = K$ is because $F = G\frac{m_1 m_2}{r^2}$

Three Key Roles in AI/ML

Tycho Brahe (1546-1610)

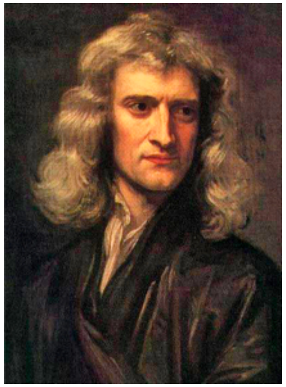
Data Collection

Time, Position
1, (a, b)
2, (c, d)
3, (e, f)
…

Johannes Kepler (1571-1630)

Model Learning

$$\frac{\tau^2}{r^3} = K$$



Isaac Newton (1643-1727)

Model Interpretation

$$F = G\frac{m_1 m_2}{r^2}$$

What if Kepler had DL in the 16-17th Century?

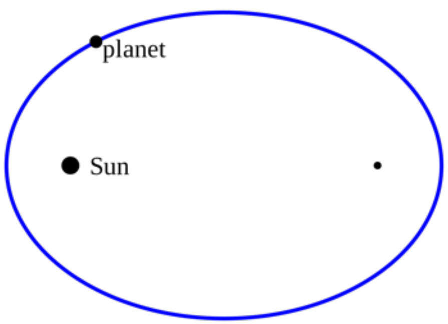
We can
Obverse it!

We can
Predict it!

Tycho Brahe (1546-1610)
Demark astronomer

Johannes Kepler (1571-1630)
German astronomer, student of Tycho Brahe.

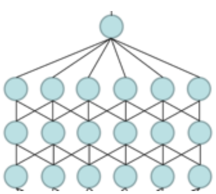
Time, Position
1, (a, b)
2, (c, d)
3, (e, f)
…



There must be some rules underlying the data.

I don't know what it is, but NN can fit any function.

So I'm going to train a NN to fit the data!



It fits the data pretty well!

I can make predictions!

$$\tau = some\ NN(r)$$

But wait: can this be called science discovery?
Science is not only about know HOW, but also know WHY!

Challenges in Modern Science Research

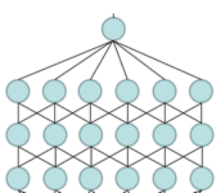
We can
Predict it!

Johannes Kepler (1571-1630)
German astronomer, student of Tycho Brahe.

There must be some rules underlying the data.

I don’t know what it is, but NN can fit any function.

So I’m going to train a NN to fit the data!



It fits the data pretty well!

I can make predictions!

$$\tau = some\ NN(r)$$

But wait: can this be called science discovery?
Science is not only about know HOW, but also know WHY!

- However, manually analyzing data like Kepler did is almost impossible in modern science research, because tons of data is being produced.
  - E.g., By astronomical telescope and particle colliders.
- We indeed need AI for data analyses and model learning

Challenges in Modern Science Research

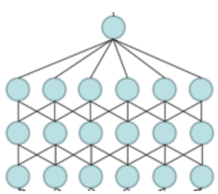
We can
Predict it!

Johannes Kepler (1571-1630)
German astronomer, student of Tycho Brahe.

There must be some rules underlying the data.

I don't know what it is, but NN can fit any function.

So I'm going to train a NN to fit the data!

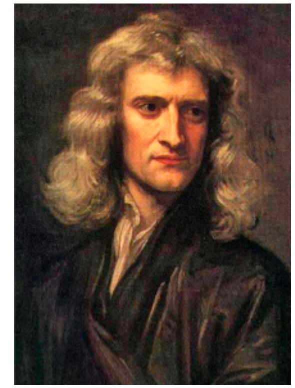


It fits the data pretty well!

I can make predictions!

$$\tau = some\ NN(r)$$

But wait: can this be called science discovery?
Science is not only about know HOW, but also know WHY!



We **Understand** it!
We know **Why**!

Isaac Newton (1643-1727)

Explainable AI (XAI) plays the role of Newton!

Interpret and **explain** the learned (black-box) model, derive insightful discoveries/theories.

Help us better understand the nature.

A New Paradigm of (AI-assisted) Science Discovery

Step 1: Data Collection
Step 2: (Black-box) model learning
Step 3: Model Interpretation based on XAI
Step 4: Science experts derive scientific insights
Step 5: Iterate if necessary, e.g., more data required

Tycho Brahe (1546-1610)

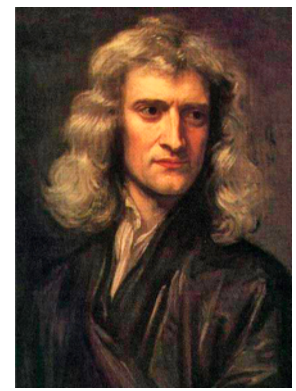
Data Collection

Almost automatic

Johannes Kepler (1571-1630)

Model Learning

Many available methods



Isaac Newton (1643-1727)

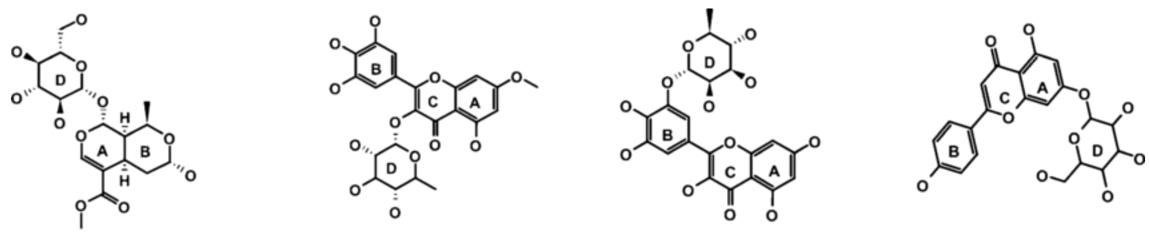
Model Interpretation (XAI)

Still needs much exploration

An Example in Medical Research: Drug Discovery

- Step1: Target a disease.

- Step2: Thousands of (or even more) candidate drug molecular structures. Impossible to conduct clinical trial for all of them



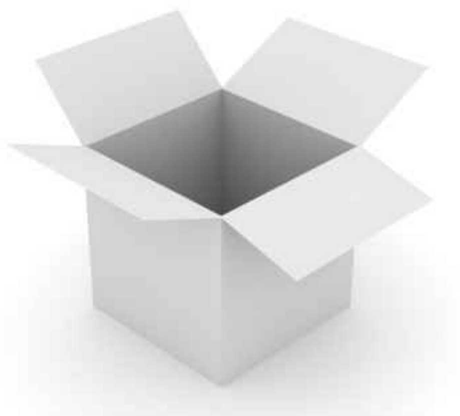
- Step3: Train a ML model (e.g., Graph Neural Network) for pre-selection
  - Selects only a few drugs that the model thinks are potentially effective

- Step 4: Use XAI model to explain the results
  - Why the model believes the selected drugs are potentially effective
  - E.g., which functional group of the molecular could be active

- Step5: Medical experts analyze the results and make decisions

Current XAI Methods

- (In general) Two types of XAI methods
  - Model Intrinsic Explanation and Model Agnostic Explanation



Intrinsic explainable models

Sub-type1: Model is a “white box”,
we naturally know how the model works
e.g., Linear regression, decision trees,
(neural) symbolic AI

Sub-type2: Certain information in the model reveals how the model works
e.g., attention mechanism
explicit factor models

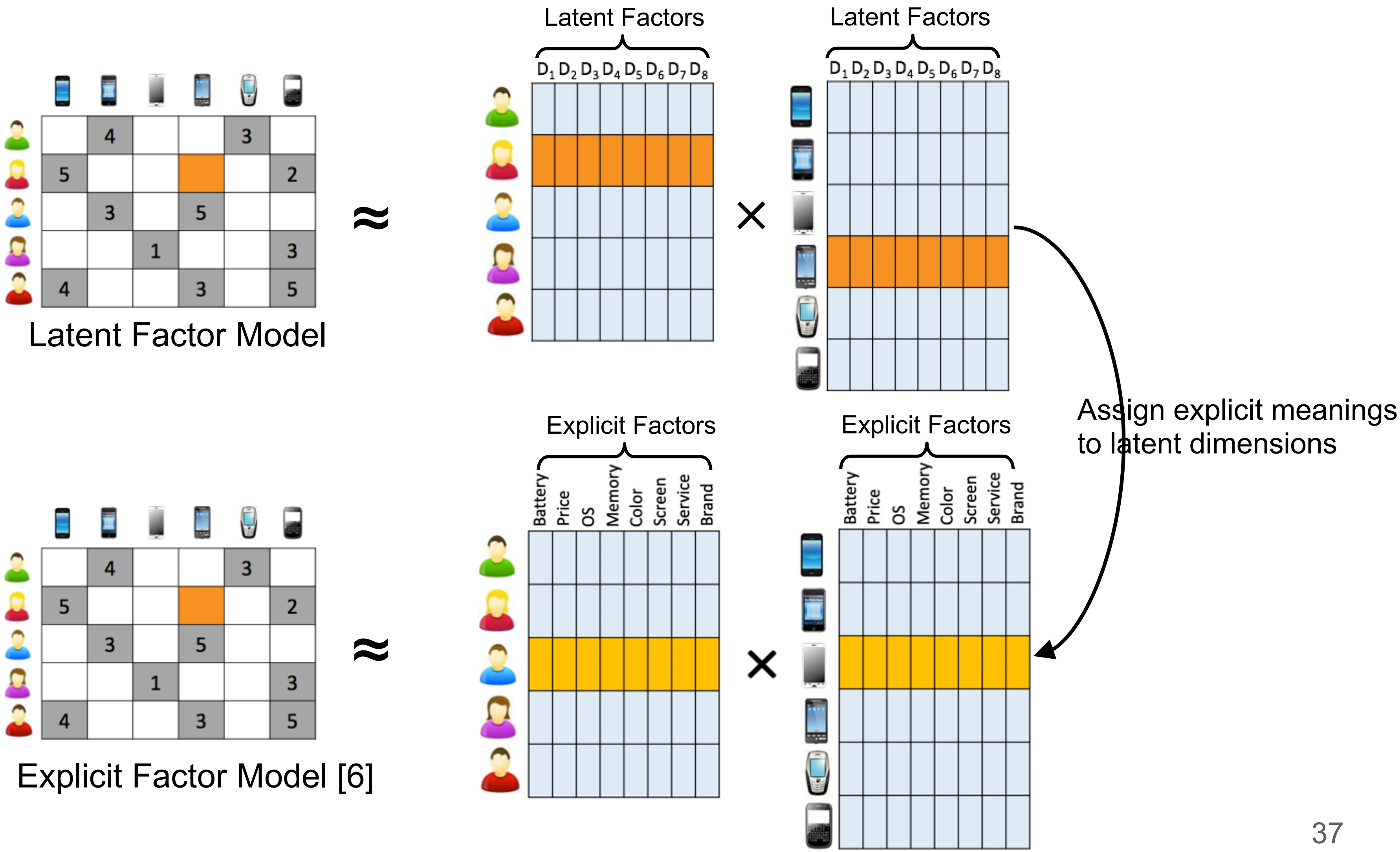
Agnostic/post-hoc explanation models

The prediction model is still a “black box”
We develop other models to “explain” the black box

Sub-type1: (Local or global) function approximation
e.g., Local Interpretable Model-Agnostic Explanations (LIME)

Sub-type2: Function influence analysis
e.g., Causal analysis of the black-box model

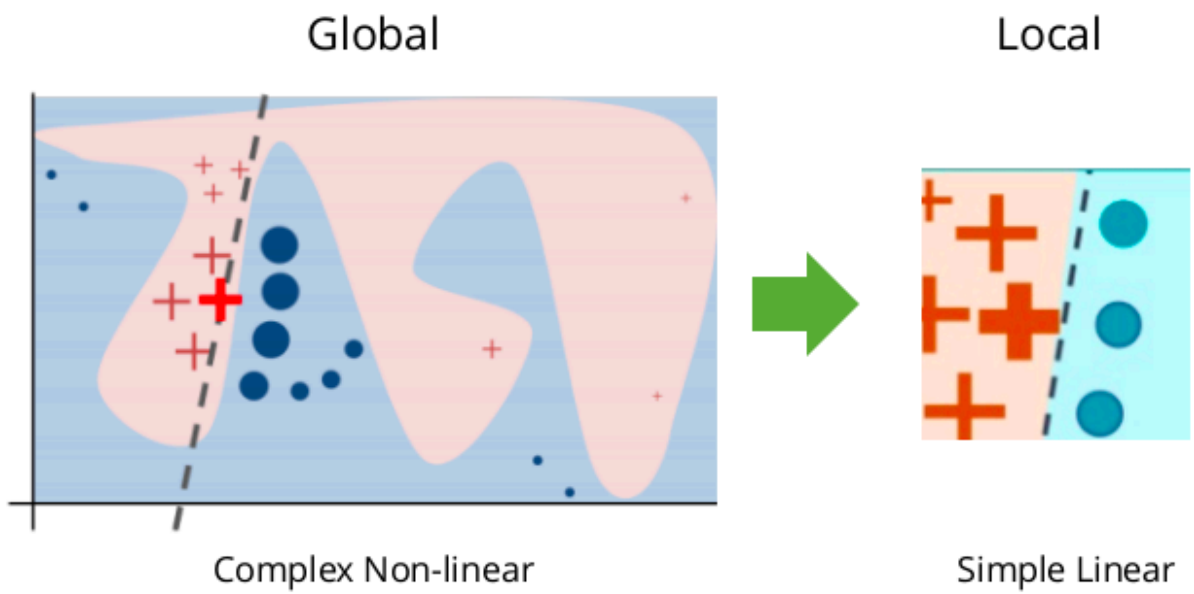
Intrinsic Explanation Model – an Example



[6] Zhang, Y., Lai, G., Zhang, M., Zhang, Y., Liu, Y. and Ma, S. (2014). Explicit factor models for explainable recommendation based on phrase-level sentiment analysis. ACM SIGIR 2014.

Agnostic Explanation Model – an Example

LIME - Local Interpretable Model-Agnostic Explanations [7]



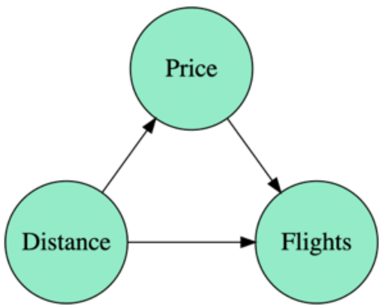
Basic idea: use a simple linear model to approximate the complex model in a small local area (i.e., around a data point). Similar to local differential analysis.

The simple linear model is a local explanation of the complex model in that area.

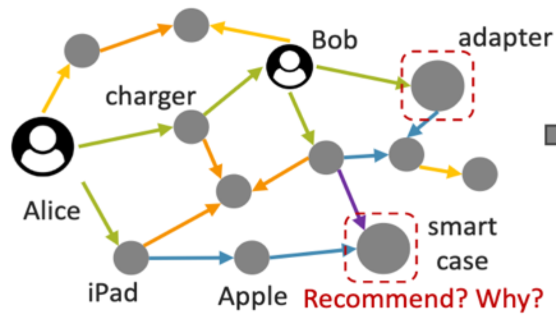
[7] Ribeiro, M.T., Singh, S. and Guestrin, C. (2016). "Why should i trust you?" Explaining the predictions of any classifier. ACM SIGKDD 2016.

Our Recent Research on Explainable AI (XAI)

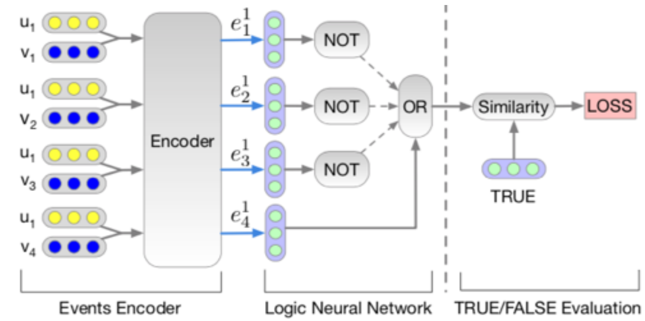
- Explainable Machine Learning Methods



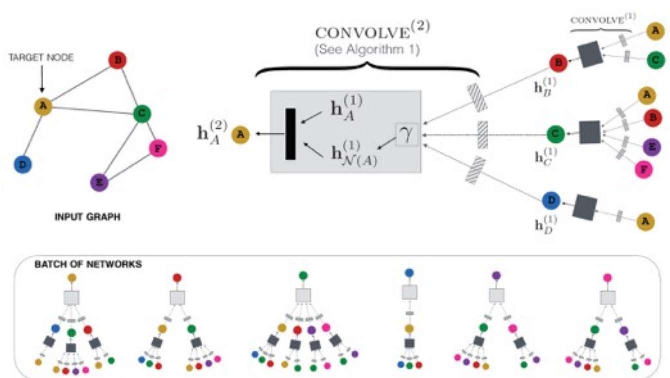
Causal Machine Learning



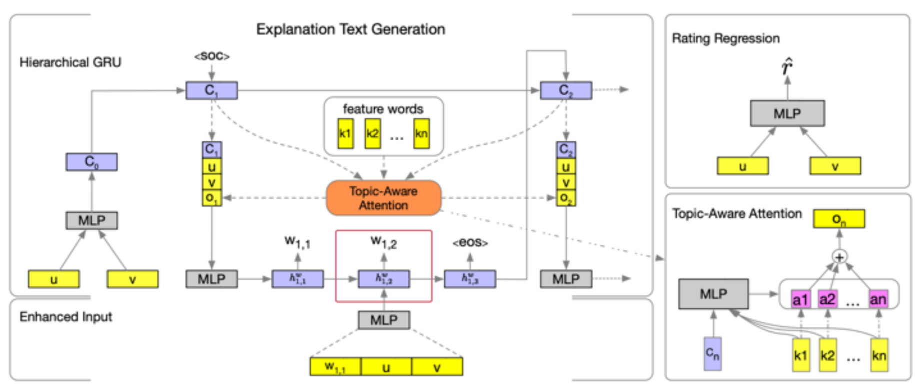
Knowledge Graph Reasoning



Neural Logic Reasoning



Explainable Graph Neural Networks



Generating Natural Language Explanations

Influence in Industry and Academia

- Our XAI/ML methods are adopted in many industry companies

- Our XAI toolkits are adopted by industry/academic institutions worldwide

- Explainable AI will be an essential part of intelligent systems
  - Search engines, recommender systems, social networks, chatbots, healthcare systems, autonomous driving, multimedia processing, science discovery, and more.
  - An important and promising direction.

Yongfeng Zhang

Department of Computer Science, Rutgers University

yongfeng.zhang@rutgers.edu

The WISE Lab at Rutgers

https://wise.cs.rutgers.edu/