Machine Learning Perspectives and Challenges

Michael I. Jordan
University of California, Berkeley
Machine Learning (aka, AI)

- First Generation (‘90-’00): the **backend**
  - e.g., fraud detection, search, supply-chain management
- Second Generation (‘00-’10): the **human side**
  - e.g., recommendation systems, commerce, social media
- Third Generation (‘10-now): **end-to-end**
  - e.g., speech recognition, computer vision, translation
- Fourth Generation (emerging): **markets**
  - not just one agent making a decision or sequence of decisions
  - but a huge interconnected web of data, agents, decisions
  - many new challenges!
Perspectives on AI

• The classical “human-imitative” perspective
  – cf. AI in the movies, interactive home robotics

• The “intelligence augmentation” (IA) perspective
  – cf. search engines, recommendation systems, natural language translation
  – the system need not be intelligent itself, but it reveals patterns that humans can make use of

• The “intelligent infrastructure” (II) perspective
  – cf. transportation, intelligent dwellings, urban planning
  – large-scale, distributed collections of data flows and loosely-coupled decisions
Human-Imitative AI: Where Are We?

• Computer vision
  – \textit{Possible}: labeling of objects in visual scenes
  – \textit{Not Yet Possible}: common-sense understanding of visual scenes

• Speech recognition
  – \textit{Possible}: speech-to-text and text-to-speech in a wide range of languages
  – \textit{Not Yet Possible}: common-sense understanding of auditory scenes

• Natural language processing
  – \textit{Possible}: minimally adequate translation and question-answering
  – \textit{Not Yet Possible}: semantic understanding, dialog

• Robotics
  – \textit{Possible}: industrial programmed robots
  – \textit{Not Yet Possible}: robots that interact with humans and can operate autonomously over long time horizons
Human-Imitative AI Isn’t the Right Goal

• Problems studied from the “human-imitative” perspective aren’t necessarily the same as those that arise in the IA or II perspectives
  – unfortunately, the “AI solutions” being deployed for the latter are often those developed in service of the former
Human-Imitative AI Isn’t the Right Goal

• Problems studied from the “human-imitative” perspective aren’t necessarily the same as those that arise in the IA or II perspectives
  – unfortunately, the “AI solutions” being deployed for the latter are often those developed in service of the former

• To make an overall system behave intelligently, it is neither necessary or sufficient to make each component of the system be intelligent
Human-Imitative AI Isn’t the Right Goal

• Problems studied from the “human-imitative” perspective aren’t necessarily the same as those that arise in the IA or II perspectives
  – unfortunately, the “AI solutions” being deployed for the latter are often those developed in service of the former

• To make an overall system behave intelligently, it is neither necessary or sufficient to make each component of the system be intelligent

• “Autonomy” shouldn’t be our main goal; rather our goal should be the development of small intelligences that work well with each other and with humans
Near-Term Challenges in II

- Error control for **multiple** decisions
- Systems that create **markets**
- Designing systems that can provide meaningful, calibrated notions of their **uncertainty**
- Managing **cloud-edge** interactions
- Designing systems that can find **abstractions** quickly
- **Provenance** in systems that learn and predict
- Designing systems that can **explain** their decisions
- Finding causes and performing **causal** reasoning
- Systems that pursue **long-term goals**, and actively collect data in service of those goals
- Achieving **real-time** performance goals
- Achieving **fairness** and **diversity**
- Robustness in the face of **unexpected situations**
- Robustness in the face of **adversaries**
- **Sharing data** among individuals and organizations
- Protecting **privacy** and data ownership
Multiple Decisions: The Load-Balancing Problem

• In many problems, a system doesn’t make just a single decision, or a sequence of decisions, but huge numbers of linked decisions in each moment
  – those decisions often interact
Multiple Decisions: The Load-Balancing Problem

• In many problems, a system doesn’t make just a single decision, or a sequence of decisions, but huge numbers of linked decisions in each moment
  – those decisions often interact
• They interact when there is a scarcity of resources
• To manage scarcity of resources at large scale, with huge uncertainty, algorithms (“AI”) aren’t enough
Multiple Decisions: The Load-Balancing Problem

• In many problems, a system doesn’t make just a single decision, or a sequence of decisions, but huge numbers of linked decisions in each moment
  – those decisions often interact
• They interact when there is a scarcity of resources
• To manage scarcity of resources at large scale, with huge uncertainty, algorithms (“AI”) aren’t enough

• There is an emerging need to build AI systems that create markets; i.e., blending statistics, economics and computer science
Multiple Decisions: Load Balancing

• Suppose that recommending a certain movie is a good business decision (e.g., because it’s very popular)
Multiple Decisions: Load Balancing

• Suppose that recommending a certain movie is a good business decision (e.g., because it’s very popular)
• Is it OK to recommend the same movie to everyone?
Multiple Decisions: Load Balancing

• Suppose that recommending a certain movie is a good business decision (e.g., because it’s very popular)
• Is it OK to recommend the same movie to everyone?
• Is it OK to recommend the same book to everyone?
Multiple Decisions: Load Balancing

• Suppose that recommending a certain movie is a good business decision (e.g., because it’s very popular)
• Is it OK to recommend the same movie to everyone?
• Is it OK to recommend the same book to everyone?
• Is it OK to recommend the same restaurant to everyone?
Multiple Decisions: Load Balancing

• Suppose that recommending a certain movie is a good business decision (e.g., because it’s very popular)
• Is it OK to recommend the same movie to everyone?
• Is it OK to recommend the same book to everyone?
• Is it OK to recommend the same restaurant to everyone?
• Is it OK to recommend the same street to every driver?
Multiple Decisions: Load Balancing

• Suppose that recommending a certain movie is a good business decision (e.g., because it’s very popular)
• Is it OK to recommend the same movie to everyone?
• Is it OK to recommend the same book to everyone?
• Is it OK to recommend the same restaurant to everyone?
• Is it OK to recommend the same street to every driver?
• Is it OK to recommend the same stock purchase to everyone?
Multiple Decisions: The Statistical Problem

JELLY BEANS CAUSE ACNE!

SCIENTISTS! INVESTIGATE!

BUT WE'RE PLAYING MINECRAFT!

...FINE.

WE FOUND NO LINK BETWEEN JELLY BEANS AND ACNE (P > 0.05).

THAT SETTLES THAT.

I HEAR IT'S ONLY A CERTAIN COLOR THAT CAUSES IT.

SCIENTISTS!

BUT MINECRAFT!
WE FOUND NO LINK BETWEEN
PURPLE JELLY
BEANS AND ACNE
(\(P > 0.05\)).

WE FOUND NO LINK BETWEEN
BROWN JELLY
BEANS AND ACNE
(\(P > 0.05\)).

WE FOUND NO LINK BETWEEN
PINK JELLY
BEANS AND ACNE
(\(P > 0.05\)).

WE FOUND NO LINK BETWEEN
BLUE JELLY
BEANS AND ACNE
(\(P > 0.05\)).

WE FOUND NO LINK BETWEEN
TEAL JELLY
BEANS AND ACNE
(\(P > 0.05\)).

WE FOUND NO LINK BETWEEN
SALMON JELLY
BEANS AND ACNE
(\(P > 0.05\)).

WE FOUND NO LINK BETWEEN
RED JELLY
BEANS AND ACNE
(\(P > 0.05\)).

WE FOUND NO LINK BETWEEN
TURQUOISE JELLY
BEANS AND ACNE
(\(P > 0.05\)).

WE FOUND NO LINK BETWEEN
MAGENTA JELLY
BEANS AND ACNE
(\(P > 0.05\)).

WE FOUND NO LINK BETWEEN
YELLOW JELLY
BEANS AND ACNE
(\(P > 0.05\)).

WE FOUND NO LINK BETWEEN
GREY JELLY
BEANS AND ACNE
(\(P > 0.05\)).

WE FOUND NO LINK BETWEEN
TAN JELLY
BEANS AND ACNE
(\(P > 0.05\)).

WE FOUND NO LINK BETWEEN
CYAN JELLY
BEANS AND ACNE
(\(P > 0.05\)).

WE FOUND A LINK BETWEEN
GREEN JELLY
BEANS AND ACNE
(\(P < 0.05\)).

WE FOUND NO LINK BETWEEN
MAUVE JELLY
BEANS AND ACNE
(\(P > 0.05\)).

WE FOUND NO LINK BETWEEN
BEIGE JELLY
BEANS AND ACNE
(\(P > 0.05\)).

WE FOUND NO LINK BETWEEN
LILAC JELLY
BEANS AND ACNE
(\(P > 0.05\)).

WE FOUND NO LINK BETWEEN
BLACK JELLY
BEANS AND ACNE
(\(P > 0.05\)).

WE FOUND NO LINK BETWEEN
PEACH JELLY
BEANS AND ACNE
(\(P > 0.05\)).

WE FOUND NO LINK BETWEEN
ORANGE JELLY
BEANS AND ACNE
(\(P > 0.05\)).
GREEN JELLY BEANS LINKED TO ACNE!

95% CONFIDENCE

ONLY 5% CHANCE OF COINCIDENCE!

SCIENTISTS...
DAGGER

(Ramdas, Chen, Wainwright & Jordan, 2018)

\[ \textbf{Inputs:} \quad \text{DAG } \mathcal{G}, \text{ p-values attached to each node, target FDR } \alpha. \]

\textbf{Procedure:}

Partition the DAG into \( \mathcal{H}_1, \ldots, \mathcal{H}_D. \)

\textbf{for} \( d = 1, 2, \ldots, D \) \textbf{do}

Run a step-up procedure to test the hypotheses in \( \mathcal{H}_d \) with threshold functions \( \{\alpha_{d,i}(r)\}_{i=1}^{|\mathcal{H}_d|} \) defined by

\[ \alpha_{d,i}(r) = 1 \left\{ \bigcap_{j \in \text{Par}(i)} \mathcal{H}_{d,j} \in \mathcal{R}_{1:d-1} \right\} \alpha \frac{\ell_i \beta_d (m_i + r + R_{1:d-1} - 1)}{m_i}, \]

where \( R_{1:d-1} \) is the number of rejected hypotheses in the first \( d - 1 \) layers.

\textbf{end for}
Data and Markets

• Where data flows, economic value can flow
• Data allows prices to be formed, and offers and sales to be made
• The market can provide load-balancing, because the producers only make offers when they have a surplus

• Load balancing isn’t the only consequence of creating a market
• It’s also a way that AI can create jobs
Example: Music in the Data Age

• More people are making music than ever before
• More people are listening to music than ever before
Example: Music in the Data Age

- More people are making music than ever before
- More people are listening to music than ever before
- But there is no economic value being exchanged
- And most people who make music cannot do it as their full-time job
An Example: United Masters

- *United Masters* partners with sites such as Spotify, Pandora and YouTube, using ML to figure out which people listen to which musicians
- They provide a dashboard to musicians, letting them learn where their audience is
- The musician can give concerts where they have an audience
- And they can make offers to their fans
An Example: United Masters

- *United Masters* partners with sites such as Spotify, Pandora and YouTube, using ML to figure out which people listen to which musicians.
- They provide a dashboard to musicians, letting them learn where their audience is.
- The musician can give concerts where they have an audience.
- And they can make offers to their fans.
- I.e., consumers and producers become linked, and value flows: a market is created.
- The company that creates this market profits.
Learning with Long-Term Goals

• Current deep-learning technology is based mostly on supervised learning
  – this requires enormous numbers of labels
• It’s also based mostly on short-term temporal relationships (or snapshots)
• Moving beyond this requires the kinds of concepts that are found in optimal-control theory, specifically its sampled-based version known as reinforcement learning (RL)
Reinforcement Learning (RL)

- Reinforcement learning involves trying out sequences of actions and seeing what the outcome is.
- A sequence of actions is referred to as a “roll-out”.
  - Actions from a successful roll-out are “backed-up” in time, so that the subsequences of that roll-out are more probable in the future.
Reinforcement Learning (RL)

• **Reinforcement learning** involves trying out sequences of actions and seeing what the outcome is
• A sequence of actions is referred to as a “roll-out”
  – actions from a successful roll-out are “backed-up” in time, so that the subsequences of that roll-out are more probable in the future
• Most of the successes to date (e.g., AlphaGo) have been done using simulators
• When one has a simulator, one can do many many millions or billions of roll-outs
  – some roll-outs terminate quickly, others terminate much more slowly
• This setting yields major new requirements on distributed hardware and software platforms
Roll-Outs

Try lots of different policies and see which one works best…
Ray: A Distributed Execution Framework for Emerging RL Applications

Moritz, Nishihara, Wang, Tumanov, Liaw, Liang, Paul, Jordan and Stoica

https://github.com/ray-project/ray
About Ray

**Goal:** Make it easy to write high-performance, real-time distributed applications, especially AI/ML applications.

**Example use cases:**

- Reinforcement learning
- Distributed stochastic gradient descent (training neural networks)
- Hyperparameter search
- General purpose parallel/distributed Python
- Streaming
About Ray

**Goal:** Make it easy to write high-performance distributed applications, especially AI/ML applications.

**Problems with existing solutions:**

- **Spark**
  - Not sufficiently expressive (limited to bulk synchronous parallel (BSP) model)
  - Insufficient performance (target sub-second as opposed to sub-millisecond latencies)
  - Doesn’t handle numerical data well
  - Difficulty to integrate with third-party libraries

- **MPI**
  - Not fault tolerant
  - Difficult to write correct code
  - User has to implement scheduling and communication logic
About Ray

• Generality
  ○ Combines two key ingredients of a modern programming language: functions and objects
  ○ These are called tasks (stateless) and actors (stateful)
  ○ Cf. the Map-Reduce paradigm, which dispensed with objects
  ○ Can create tasks within tasks

• Ease of use
  ○ Integrates easily with arbitrary Python libraries (e.g., TensorFlow, PyTorch)
  ○ Easy to implement/customize new algorithms
  ○ Easy to parallelize existing Python code
  ○ Transparent fault tolerance
Ray performance

One million tasks per second
Ray architecture

Driver
Worker
Object Store
Local Scheduler

Worker
Worker
Object Store
Local Scheduler

Worker
Worker
Object Store
Local Scheduler

Global Scheduler
Global Control Store

Debugging Tools
Profiling Tools
Web UI
```python
@ray.remote

class Worker(object):
    def do_simulation(self, policy, seed):
        # perform simulation and return reward

workers = [Worker.remote() for i in range(20)]
policy = initial_policy()

for i in range(200):
    seeds = generate_seeds(i)
    rewards = [workers[j].do_simulation.remote(policy, seeds[j])
               for j in range(20)]
policy = compute_update(policy, ray.get(rewards), seeds)
```
Video 1

Iteration 0
Ray is Open Source

- [https://github.com/ray-project/ray](https://github.com/ray-project/ray)
- You can install Ray with
  `pip install ray`
Summary

• ML (AI) has come of age
• But it is far from being a solid engineering discipline that can yield robust, scalable solutions to modern data-analytic problems
• There are many hard problems involving uncertainty, inference, decision-making, robustness and scale that are far from being solved
  – not to mention economic, social and legal issues
Near-Term Challenges in II

- Error control for multiple decisions
- Systems that create markets
- Designing systems that can provide meaningful, calibrated notions of their uncertainty
- Managing cloud-edge interactions
- Designing systems that can find abstractions quickly
- Provenance in systems that learn and predict
- Designing systems that can explain their decisions
- Finding causes and performing causal reasoning
- Systems that pursue long-term goals, and actively collect data in service of those goals
- Achieving real-time performance goals
- Achieving fairness and diversity
- Robustness in the face of unexpected situations
- Robustness in the face of adversaries
- Sharing data among individuals and organizations
- Protecting privacy and data ownership
Thank you!