Khipu 2019 notes

Juan Cruz Barsce

jbarsce@frvm.utn.edu.ar

Event webpage: `https://khipu.ai/`

This document contain the notes I took during Khipu 2019, Montevideo, Uruguay. Please contact me if there is something that needs to be corrected.

Contents

1 Day 1

1.1 ML Challenges and Opportunities of Computational Behavioral Phenotyping in Developmental Health (Guillermo Sapiro)

Video

- 1/9 of children (2 in each elementary school classroom) have development problems.

Problems of detecting autism with a clinician:

- The system does not scale (not enough doctors to diagnose all). Autism can be detected when child is 18 months old, but the average age when it is detected is when child is 5 years old.

Solution: Computer Vision applied on children to see if they have autism when they watch videos.

- To preserve privacy, we can use information theory and e.g. random priors and filters. We can reach a 'no-harm fairness' based on Pareto optimality.
- Difficulty: metrics on clinics are **not** the same metrics for ML.

Also learned:

- There is a possible connection with autism and parkinson, regarding to facial rigidity.
- Nested learning: use ML to provide ontology when not knowing exactly the class.

1.2 Machine Learning Fundamentals (Luciana Ferrer)

Video
Slides

- Check for iid when deciding e.g. if to randomly split a dataset
- Consider statistical significance for small datasets. Do **not** ignore correlations.
- Calibration is very important e.g. when priors in testing are different that in training e.g. when training a medical data when a disease is rare or when we care more about some kind of errors than other.

- We are used to assume that training prior = test prior.
- ROC curves are very useful when not committing to a threshold.
- The cost can be measured as a sum between discrimination cost + calibration cost.
- Interesting: prior-weighting cross entropy.
- PAV algorithm link. Not trivial: how to obtain the minimum of the PAV algorithm? Also: no PAV when N ¿ 1 classes.
- When to fix bad calibration? whenever we have a good score in a system **and** we have confidence that it **is not overfitting**
- 'Calibration should be the first focus when starting ML a new task'.
- 'Keep a few metrics in mind when optimizing'.
- Also batch norm/dropout can cause miscalibration.

Good reference: Gun et al. 2017 'On Calibration of Modern NN' Also, to start with calibration check Nico Brummer's material.

1.3 Deep Learning Fundamentals (René Vidal)

`http://tv.vera.com.uy/video/55357`Video

- '(in deep learning) we have gone from hand crafted features to hand crafted architectures'
- Number of training samples grows polinomially with network size.
- Current research is in interplay between optimization and generalization.
- 'Droput induces specific regularizer' - it 'introduces' convexity!
- 'NNs, to work in scarse datasets need some domain modeling, introducing expert knowledge among the inputs'.
- 'People aren't using dropout these days anymore, batch normalization is being used instead'
- 'Dropout can be thought as a specific case of batch norm where you let weights be free'

Good reference: Bengio et al. '05 - Convex Neural Networks
Another notes from the day: interesting potential applications of reinforcement learning:

- Knowledge transfer
- Multi-criteria RL
- Hypertune: hyper-parameter optimization library from Google

2 Day 2

2.1 Convolutional Neural Networks II (Juan Carlos Niebles)

Video
Slides
Human event understanding in models: From actions to tasks
Hierarchical structure of events
Problem: temporal action detection

- To detect how much time does an action takes place. e.g. detect action ending and remember when it started.
- This is very possible in domains such as sports. It has some limitations, does not capture complexity/scale.

Complex actions to input language

- A composite event can be broken down into actions using language.
- It detects where a word happens in a video
- Problems with simpler or complicated uses of language

Now, can we extract structure in a video?

- For that, we need context.
- Solution: graphs! weakly supervised learning that connects graphs

2.2 Keynote from María José Escobar

Video

Slides

- Interesting: first convolutional layer in deep RL replaced by a retina module that is a lot more efficient.

2.3 Generative Models and Unsupervised Learning (Ian Goodfellow)

The talk was about generative models and unsupervised learning.

- Generative models fits points to a density.
- Unsupervised learning generate samples from a distribution. There is a blur in the term unsupervised learning e.g.:
- an autoregressive model turn unsupervised into N supervised tasks
- how we decide if a variable is x or y?

Downsides of GANs:

- Tends to drop modes of the distribution.
- Generate discrete data.
- Can some sort of adversarial loss be used instead of MSE loss?
- 'You can think GANs as a weird form of RL that gets reward from the discriminator'

Interpretability and security are really hard for current ML. Regarding interpretability, 'best papers right now are those that debunks another interpretability papers'

Importance of rigorous testing on ML models and discarding models that perform bad.

Interesting reference: In Abeel et al. 2006, a RL environment is learned by a generative model of the world

2.4 Panel: How to write a great research paper

Slides

Members: Nando de Freitas, Claire Monteleoni, David Lopez-Paz and Martin Arjovsky

A discussion took place in the panel based on seven principles. Source

1. Don't wait. Write.

   - Write a small 1-page draft 'cheat-sheet / key bullet points' of your idea and ask feedback to, say, 10 people.

2. Identify your key idea.

   - 'Why do you claim this as publishable?' It should not sound ad hoc

3. Tell **one story**.

   - A nice-real quick to the point story.
   - Do put negative results (1), but only those that are related to your main story you want to tell.

(1) such results are not negative in the sense that they invalidate algorithm of the paper. If that were the case, then there are bigger problems to be addressed.

4. Nail your contributions to the mast.
5. Related work: later in the paper, not at first

   - Except some of the related work is the basis of your paper. In such case it should be included in the preliminaries.
   - Also: related work later in the process of writing, when it is clearer.

6. Put your readers first

   - Really think who you are writing for.
   - Generate the 'aha!' moment early on, in the beginning

7. Listen to your readers

   - Listen to feedback.
   - If anything seem unclear or questionable, really listen to that.

- ’Write the paper for grad students’

Additional tips (open panel):

- Measure progress in terms of learning, not in terms of results.
- You must be able to summarize your paper in 1 sentence.
- ’Writing papers is like a GAN, where your are the generator and the reviewer is the discriminator/adversarial’
- ’First thing when writing: download the reviewers guideline’.
- ’Plot the beleived curve even before running the experiments. If the plot matches with the experiments, great! if not, you’ve learned a lot, even if you were wrong’.
- It is awesome if reviewer/colleagues can look at the figure, read the captions and get the paper right there.
- Use macros on latex symbols/notations/conference names/etc.
- Have a baseline to compare with!
- Use active voice, so you are in charge of the claims and not hiding behind them.
- Avoid use of qualifiers, adverbs, passive voice, etc.
- ’your research will be falsified, its the history of science’
- Put one sentence per line in the editor, so it is clearer and easier to follow through version control.

3 Day 3

3.1 Machine Learning for Health Care (Danielle Belgrave)

Current challenges

- Heterogeneous patient app.
- Tradeoff between accuracy and fairness
- Endotype discovery: identify subgroups of complex disease risk.

- Probabilistic modeling.
- Individualized disease progression model that includes population, subpopulation and disease encoded in random variables.

3.2 Perspectives on AI (Yoshua Bengio)

Current state:

- Higher-level conscious cognition seems currently out of reach.
- Attention revolution in deep learning.

Limitations of deep learning:

- Too high sample complexity (even worse for reinforcement learning).
- Some very important concepts still hard-coded (labels, for instance).
- Errors made by trained systems reveals 'very shallow' understanding.

Agent's perspective for deep learning

- Neccessity to build actual models of the world, uncovering underlying causal relationships.
- Learn generative models in latent space, not pixel space.
- Priors that depends on the goal context.

Good reference: Bengio's conscious prior paper
Additional notes:

- 'The problem [in ML] is what to predict, because much of it can be meaningless'
- 'Missing currently in ML: understand and generalize beyond the training distribution.'
- 'Meta-learning can help a lot in current ML'
- 'Both high-level and low-level concepts should be learned at the same time'
- 'Think beyond iid (independent and identically distributed). Most of changes in a distribution are because of agent's actions'

Recommended paper: Recurrent Independent Mechanisms
Big emphasis on **AI for social good**

3.3 Khipus

Slides

3.4 Reinforcement Learning (Nando de Freitas)

Video

Slides

RL summed up in three equations:

$$
\begin{aligned}
\nabla_\theta \rho_\pi(\theta) &= \mathbb{E}_\tau[(\sum_{t=0}^{T} R_t \nabla_\theta \log \pi_\theta(a_t \mid s_t))] && (1)\\
\nabla_\theta \rho_\pi(\theta) &= \mathbb{E}_{d_\pi(s)\pi(a|s)}[Q_\pi(s,a)\nabla \log \pi_\theta(a \mid s)] && (2)\\
Q^*(s,a) &= \mathbb{E}_{s'}[r(s,a,s') + \gamma max_{a'} Q^*(s',a') \mid s,a] && (3)
\end{aligned}
$$

- '[Notion of] agent as something that can have many actors'

Deep Distributed Distributional Deterministic Policy Gradient (D4PG)

- DDPG where Q is replaced with expected reward and that can use a distributed version of the agent.

- 'TRPO / PPO / ACER / Impala works great when you can simulate at many times real time. In other words, on-policy methods like them shine in simulation'.
- MPO 'DQN-like algorithm with KL divergence for policy'. Paper
- R2D3: 'the latest DQN'. It performs Batch RL. 'If we have data of the robot doing things (demonstration), we can use that data to compute a policy instead of using the robot (considering that the robot of a factory is hard-coded)' Paper
- AlphaNPI - DeepMind algorithm that solves Tower of Hanoi Source

David Silver's principles: Source

1. Evaluation Drives Progress Direction of progress is determined by the choice of the evaluation metric; this is the most important single decision in the course of a project.

2. Scalability Determines Success Algorithm scalability as a 'performance gradient': how does its performance increase given an increase of the resources (e.g. computation, memory, data)?

   A good algorithm should be optimal given infinite resources.

3. Generality Future-Proofs Algorithms Agents should be tested against a diverse but realistic set of RL environments.

4. Trust in the Agent's Experience Experience is the data in RL, and it should be trusted as the sole source of knowledge, avoiding the temptation of hard-coding with features, heuristics, environment changes, and so on.

5. State is Subjective Agents should construct their **own** state from their experience.

6. Control the Stream Agents live in rich streams of data; observation streams into the agent and agent, in turn, streams out actions. Controlling the features allows to control the stream, thus the future, and thus maximizing any reward.

7. Value Functions Model the World Why to use a value function?

   (a) They efficiently summarize the future

   (b) Reduce planning to constant-time look-up, rather than exponential lookahead

   (c) Can be computed and learned independent of their span

   Multiple value functions may be learned, at multiple timescales or to model many aspects of the world.

8. Planning: Learn from Imagined Experience Planning by imagining what will happen next, and learn from that imagined experience. Value funtion approximation should be focused on the moment *now*.

9. Empower the Function Approximator Algorithmic complexity should be pushed into the network architecture.

10. Learn to Learn Meta-learning as a way to finish handcrafting models, and everything is learned, from algorithms to features and end-to-end predictions.

Interesting RL PhD thesis:

1. John Schulman thesis

2. Peter Abeel thesis

4 Day 4

4.1 Climate and Artificial Intelligence (Claire Monteleoni)

Video

- There are lots of remaining open quests from climate change scientists
- High class imbalance when studying extremes in climate
- In the past 20-50 years, climate change was a ’small-data’ problem, and small-scale simulations were performed. Now climate change is a massive high-dimensional big data problem, where we look at a low-dimensional manifold.
- Looking at the pair atmosphere (more sensible to change) and ocean (less sensible to change) to see differences.

Workshop of climate informatics: `http://climateinformatics.org` This page also has very nice resources, datasets and materials.
Another workshop: `https://www.climatechange.ai/`

4.2 Sponsor Talk: DeepMind (Federico Carnevale)

Simulated agent trained with RL explores a room and learn representations. Then, it is asked questions, in order to understand what it learned.
Network consisting on

- Predicted loss:
  - Action-conditional CPC: compares predicted future with what really happened in future.
  - Simcore: uses a conditional generative model to do convolutional draw.
- QA Decoder: answers using supervised learning

4.3 Robotics and Continuous Control (Chelsea Finn)

Video
Slides

Moravec's Paradox: Simpler and broader capabilities are really hard This is the case in robotics.

Imitation learning...

- ... does not work, except when you collect a lot of data and fit them into a very powerful model.
- ... need lots of human supervision.
- ... scales well to image observation.

In model-based RL:

- Models can be reused for multiple tasks.
- Models don't optimize for task performance.
- Makes several assumption

4.4 Causality and Generalization (David Lopez-Paz)

Video

Slides

ML is full of 'learning horses' (see Clever Hans for context)

'The big lie in machine learning:' $P_{train}(X, Y) \neq P_{test}(X, Y)$

Your data is always from $P(X, Y \mid \text{observed} = 1)$ (example)

'Correlation does not imply causation' - Reichenbach's Principle of Common Cause states that correlation between X and Y is due to one of these four causes:

1. X caused Y (causal)
2. Y caused X (anticausal)
3. X and Y are both caused by Z (confounding)
4. X and Y both causes Z (selection bias)

But there is an additional possibility! coincidence

Recommended book: Judea Perl - The Book of Why

4.5 Microdata Anonymization for Learning Analytics (Lorena Etcheverry)

Video
Slides
Micro anonymization: data anonymization before it is passed to an ML model. Two views:

1. Anonymity (not revealing a person)
2. Confidentiality (not revealing an attribute)

Group aggregation (clustering) of features to preserve cluster info of the feature and allow generalization (minimizing info loss while maximizing utility). Is an NP-HARD problem

Micro aggregation: instead of replacing as above, replace with cluster centroid. Problem of this: it does not ensure confidentiality.
Takeaways:

1. Similar correlations can arise from different causal sources.
2. Different causal structures react differently to the same interventions.
3. Correlation are either spurious or invariant regarding environment changes.
4. Current machine learning models absorb spurious correlations recklessly.
5. Invariant correlations **are related to causality** and enable out-of-distribution generalization.
6. Correlation glues variables together, causation glues **distributions** together.
7. **Data from diverse environments allow to learn invariant correlations.**

5 Day 5

5.1 NLP (Luciana Benotti)

Video
Slides

Application: predict success in programming languages problems using NLP. There is a bias in language models that can be used for good. For that, Blocky is used, where learner's code is treated as natural language.

(Blocky features cool open-source resources to learn programming and computer science)

6 Practicals

Practicals repo, with the notebooks and background content for preparing for Khipu.

- Hackathon, an initiative from Cruzar that consisted in a text-recognition task to obtain text (from images, sometimes that have very poor quality) from Uruguayan civic-military dictatorship in 1973-1985. For that, a crowdsource-based tool named Luisa was used.
- Convolutional networks
- Optimization for Deep Learning, where different optimization algorithms are implemented to finding the minimum of the Rosenbrock's banana function, and then tried in Fashion MNIST.
- Recurrent neural networks
- The Transformer for Natural Language Processing and extras
- Reinforcement learning
- Generative models

7 Additional content

The 8 main takeaways from Khipu 2019, a recap from Tryolabs.