What is machine learning? How can it help?

Praneeth Netrapalli
Microsoft Research India
What is machine learning?

Identifying patterns in data

• Very broad – data may be of different types, from different domains

• Interplay between
  1. Statistics: how much data do we need to see to be sure of patterns?
  2. Computer science: mainly algorithms & optimization
Structure of the talk

1. Frameworks/problems studied in machine learning
2. Real world examples
3. Algorithms/methods to solve these problems
4. Some applications in Physics
I. Supervised learning

**Training data**

Cat  Cat  Cat  Cat  Cat  Cat  Cat

Dog  Dog  Dog  Dog  Dog  Dog  Dog

**Test data**

?  ?  ?
Problem formulation

\[ x_1 \in \mathbb{R}^{16} \qquad y_1 \in \{ -1, 1 \} \]

\[ \text{Cat} \quad (x_2, y_2) \]

\[ \text{Cat} \quad (x_3, y_3) \]

\[ \text{Dog} \quad (x_4, y_4) \]

\[ \text{Dog} \quad (x_5, y_5) \]

\[ \text{Dog} \quad (x_6, y_6) \]

\[ ? (x_7, ?) \]
Approach I – Linear classifiers

• **Problem**: Given \((x_1, y_1), \ldots, (x_n, y_n)\), predict \(y\) for a new \(x\)

• **Main idea**: approximate \(y\) using a **linear** function of \(x\)

• Perhaps \(y \approx \text{sign}(\langle w, x \rangle + b)\) for some \(w, b\)

• **Formulation**: Find \(w, b\) such that for \(i = 1, \ldots, n\)

\[
y_i = \text{sign}(\langle w, x_i \rangle + b)
\]
Approach I – Linear classifiers

• **Formulation**: Find \( w, b \) such that for \( i = 1, \ldots, n \)
  \[
y_i = \text{sign}(\langle w, x_i \rangle + b)
  \]

• (Computationally) easy to find such \( w, b \) if they exist

• Unlikely that \( y \) is such a simple function – definitely not the case in most practical applications

• **Reformulation**: \[
\max_{w,b} \sum_{i=1}^{n} y_i \cdot \text{sign}(\langle w, x_i \rangle + b)
\]

• Computationally very hard (NP hard)
Approach I – Linear classifiers

• Yet another reformulation: \( \max_{w,b} \sum_{i=1}^{n} \min(1, y_i \cdot (\langle w, x_i \rangle + b)) \)

• Performance may be worse than that of original reformulation, but can be solved efficiently

• This is the essential idea behind support vector machines (SVM)

• Efficiency vs performance: recurring theme in machine learning
Approach II – Nonlinear classifiers

• Recall linear classifiers assume $y_i \approx \text{sign}(<w, x_i> + b)$

• A better approximation would be $y_i \approx \text{sign}(p_d(x_i))$, where $p_d(\cdot)$ is a polynomial of degree $d$

• Formulation: $\max_{p_d} \Sigma_{i=1}^{n} y_i \cdot \text{sign}(p_d(x_i))$

• Computation time is exponential in $d$. People use what is known as “kernel trick”. Works well in practice – kernel SVMs.
Approach III – Neural networks

• In general, can approximate $y_i \approx \text{sign}(f_w(x_i))$ for some function class parameterized by $w$.

• For perceptual tasks such as vision, speech and language, function classes inspired by neuroscience make sense.

• Each neuron computes a simple function.
Are we done with supervised learning?

• **No!** If we apply this directly on pixel values, the results are not very impressive.

• Lots of symmetries in images – translation invariance, scale invariance, rotations/points of view etc.

• Other domains present their own set of symmetries.

• The functions we are using to approximate (SVMs, fully connected neural networks etc.) do not have these symmetries built into them.

• Major reason for poor performance on direct data
How do we fix it?

- E.g., magnitude of Fourier transform for translation invariance
  \[ \hat{f} \rightarrow \hat{f} e^{i2\pi t} \]
- Earlier approach was to use such hand-crafted features
- Very difficult and time consuming
- Can we minimize our effort in doing this?
How do we fix it?

- E.g., magnitude of Fourier transform for translation invariance
  \[ \hat{f} \rightarrow \hat{f}e^{i2\pi t} \]
- Earlier approach was to use such hand-crafted features
- Very difficult and time consuming
- Can we minimize our effort in doing this?
Convolutional neural networks (CNN)

- Convolutional filters $w$; recall that we are optimizing over $w$
- Filters can be thought of as templates; high inner product $\Rightarrow$ good match
- Lower level filters thought of as edge detectors, higher level filters thought of as object detectors
Current status

- CNNs and derivatives are best performing models for image data
- Require larger amounts of data compared to hand-crafted feature based methods
- Mostly empirical at the moment
- Both theory and applied work are major areas of research
- Several deep learning packages available in open source. Possible for new entrants to quickly learn these techniques and try on their problems.
Other settings and approaches

• Target values could be real valued – regression; easy extension

• Several other methods for classification studied in the literature. E.g., nearest neighbors, decision trees etc.

• Meta-techniques to improve performance: ensembles, bagging, distillation, boosting etc.
Phase transitions

- Study of phases and their transitions important in Physics

- Physical quantities (order parameters) that illustrate phase transitions are motivated by Physics intuition/experiments

- Can we distinguish phases and learn order parameters from data?

- Ferromagnetic 2d-Ising model; $H = -\sum_{i,j} \sigma_i \sigma_j$

- Training data: (spin configurations, above/below critical temp)

\[
\{\pm 1\}^{\sqrt{n} \times \sqrt{n}}, \pm 1
\]
Phase transitions

**Input**
Training data
\((\text{Spins}_1, \text{Label}_1), (\text{Spins}_2, \text{Label}_2), \ldots (\text{Spins}_n, \text{Label}_n)\)

**Learning algorithm**

**Output**
Prediction algorithm

**Input**
Spins

**Prediction algorithm**

**Output**
Label
Looking inside the prediction algorithm

Penultimate layer captures net magnetization
Order parameter

Label
Penultimate layer
Input spins
Prediction algorithm
Fully connected neural network
Predicting properties of molecules

**Input**
Molecule i.e., atoms, bonds, perhaps geometry of the molecule

**Prediction algorithm**

**Output**
Various properties such as
- Atomization energies
- Electron energy gap
- Fundamental vibrations

---

### Neural Message Passing for Quantum Chemistry

Justin Gilmer, Samuel Schoenholz, Patrick Riley, Oriol Vinyals, George Dahl

**Abstract**
Supervised learning on molecules has incredible potential to be useful in chemistry, drug discovery, and materials science. Luckily, several promising and closely related neural network models invariant to molecular symmetries have already been described in the literature. These models learn a message passing algorithm and aggregation procedure to compute a function of their entire input graph. At this point, the next step is to find a particularly effective variant of this general approach and apply it to chemical prediction benchmarks until we either solve them or reach the limits of the approach. In this paper, we reformulate existing models into a single common framework we call Message Pass-
Why prediction algorithm?

• Gold standard: physical measurement – very expensive/time consuming

• Next best: simulation using density functional theory (DFT) – time consuming

• Proposed approach: much faster and inexpensive compared to both
Evaluation

- QM9 dataset – details for 134,000 molecules
- Target values computed using DFT
- Train a graph neural network for predicting the target values
- Predictions are very accurate (up to chemical accuracy)
Predicting tensorial properties

- Dipole moment, polarizability etc.
- Reformulate the problem to take advantage of rotational symmetries in tensors
- Improved results over naïve approach
Anderson impurity model

- Quantum many body problem – properties of large number of strongly correlated particles
- Exact solutions impractical to compute
- Dynamical mean field theory (DMFT) computes approximate solutions
- DMFT subproblem – Anderson impurity model
Predicting functions

• Penalization for double occupancy – $U$
• Hybridization between bath and impurity – $V$
• Onsite energy of impurity – $\epsilon_d$
• Want to predict Green’s function

• Training data generated for various values of $U, V, \epsilon_d$ using exact diagonalization
• Green’s function learned in a particular basis
• Legendre polynomials used as a basis

Open direction

Green’s function at different points correlated but not enforced by the algorithm (scope for improvement)
II. Unsupervised learning

Learn from unlabeled data

Given: Training data points, we would like to do

<table>
<thead>
<tr>
<th>Method</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Clustering</td>
<td>Group data points into separate clusters</td>
</tr>
<tr>
<td>Density estimation</td>
<td>Given a new data point, estimate its likelihood</td>
</tr>
<tr>
<td>Generative models</td>
<td>Generate new data points from the same distribution</td>
</tr>
</tbody>
</table>
Algorithms/models for clustering

• Mixture of simple distributions e.g., mixture of Gaussians
  • K-means, Expectation-maximization (EM) etc.

• Subspace/manifold clustering
  • Principal components analysis (PCA)

• Graph clustering

• Hierarchical clustering
Density estimation

\[ P[\text{ow]} = ? \]
\[ P[\text{umbrella}] = ? \]
\[ P[\text{apple}] = ? \]
\[ P[\text{plane}] = ? \]
Algorithms/models for density estimation

• Fit parametric/nonparametric distributions e.g., Gaussians etc.

• Use neural networks/other functions to approximate the density

• Subspace/manifold learning
Generative models
Algorithms/models for generating examples

• Variational autoencoders (VAE)
• Generative adversarial networks (GAN)
• Many more...

Picture from Radford et al. 2015
Challenges in unsupervised learning

• Much harder problem than supervised learning

• No ground truth ⇒ difficult to evaluate performance

• Much wider applicability than supervised learning
  • Lot more data available
Phase transitions – unsupervised learning

• Can we distinguish phases and learn order parameters from unlabeled data?

• Training data: Spin configurations at various temp. $[\text{Spins}_1 | \text{Spins}_2 | \cdots | \text{Spins}_n]$

• Much more realistic setup

• Applicable to situations where there is less intuition about phases and order parameters
Technique: PCA + Clustering

• Compute the top- \( k \) singular directions of the matrix \([\text{Spins}_1 | \text{Spins}_2 | \cdots | \text{Spins}_n]\). Denote them \( w_1, \ldots, w_k \).

• Project each \( \text{Spins}_i \) onto \( \text{Span}(w_1, \ldots, w_k) \). Denote the projection \((y^i_1, \ldots, y^i_k)\).

• Cluster \( \{(y^i_1, \ldots, y^i_k)| i = 1, \ldots, n\} \)

• For a new test point, find out which cluster it belongs to

2d-Ising model
COP Ising model

• Conserved order parameter (COP)

\[ H = -\sum_{i,j} \sigma_i \sigma_j \text{ subject to } \sum_i \sigma_i = 0 \]

• Describes classical lattice gases

Phase transition and order parameter

Typical configurations below (a) & (b); and above (c) critical temperature

Singular value spectrum and singular vectors
Flurry of recent work...


Generating jet images

- Jets are streams of protons and hadrons formed due to scattering quarks/gluons
- Jet image – 2-dim radiation pattern
- Very sparse and NOT translation invariant
- Physicists use lots of simulated data which is very (computationally) expensive to generate
- Develops a framework to learn jet images
III. Reinforcement learning – example
Formal setup

Learn in dynamic & unknown environments

Components: agent & environment

Repeat

1. Agent performs an action
2. Action changes the state of the environment
3. Action also produces a reward which the agent observes

Goal: Maximize cumulative reward
Applications

• Robotics

• Planning & execution (manufacturing etc.)

• Human interaction systems e.g., conversational agents etc.

• Health and medicine e.g., design of clinical trials
Approaches

• Wide variety depending on the context
  • Do we have a model of the environment?
  • Can rewards be observed instantaneously?

• Learn the state transitions of the environment

• Learn the value function

• Learn the best policy for the agent
New quantum experiments

- Given some pieces of equipment, how to design experiments that let us create interesting phenomena?

- Toolbox: Beam splitters, mirrors, holograms and Dove prisms

- Goal: experimental setup that produces high-dimensional multipartite entanglement
Use reinforcement learning

- Learning proceeds in episodes
- Each episode ends when either
  1. Maximum number of components used or
  2. Reward obtained
- Agent chooses actions based on previous experience
- Reward given for producing desired state
- Finds orbital angular momentum (3,3,3) states
More results

Next task: Find as many experimental setups as possible for observing desired state

Agent *independently discovered* interesting techniques such as
1. nonlocal parity sorter and
2. a setup to increase dimensionality of photons

Both are modern quantum optical gadgets that have only been recently discovered by Physicists!
Variations

- Learning from sequences (time series data)
- Semi-supervised learning
- Active learning
- Transfer learning
- Learning under constraints
  - E.g., time constraints, fairness constraints etc.
- Interpretable learning
- And many more
Conclusion

• Use of machine learning for solving Physics problems has just begun

• Shows promise but ways to go

• Requires coming together of Physicists and ML researchers

• Potential to create new directions in both fields