Generative Model for Decision Trees

Riccardo Guidotti[1,2] riccardo.guidotti@unipi.it
Anna Monreale[1] anna.monreale@unipi.it
Mattia Setzu[1] mattia.setzu@unipi.it
Giulia Volpi[1] giuliavolpi25.93@gmail.com

[1]University of Pisa, Italy [2]ISTI-CNR, Italy

Decision Tree induction

When growing Decision Trees we go...

- **greedy:** Trees are induced on a greedy *suboptimal* optimization (CART, C4.5);
- **optimal:** Trees are induced with *prohibitively costly* induction algorithms (Optimal Trees).

Can we find a middle ground?

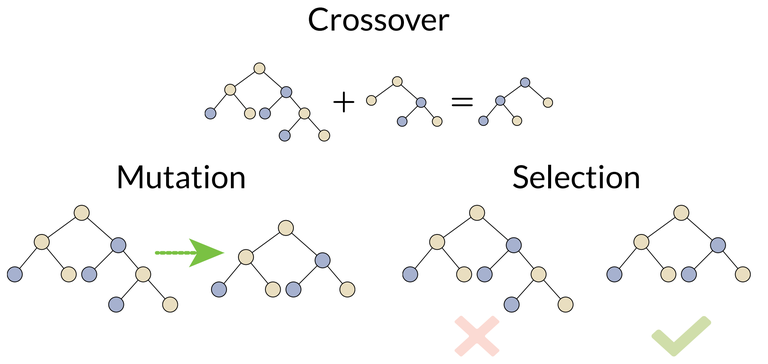
Growing Decision Trees

Genetic (Decision) Trees offer a great middle ground both in performance and induction times, but are limited by their encoding!

Growing Trees by using Chromosomes

Genetic Trees leverage human-defined Decision Tree representation called *chromosomes*, which are the seeds to a *genetic* optimization algorithm, which evolves them into a single Decision Tree.

As in natural evolution, genetic algorithms repeatedly evolve a population by emulating an environment where subjects reproduce, optimize, mutate, adapt to the environment, and less fit subjects are culled in favor of better ones.

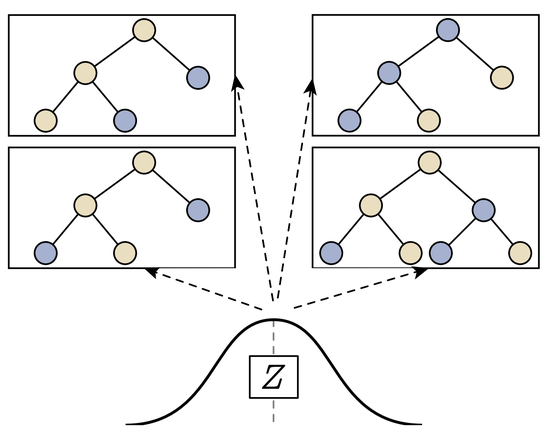


Likewise, Genetic Trees are induced by iteratively *i)* mixing, *ii)* mutating, and *iii)* culling an initial population of candidate trees. Evolution consists in tree manipulation, e.g., adding subtrees to another tree, trimming a branch, or changing a split feature.

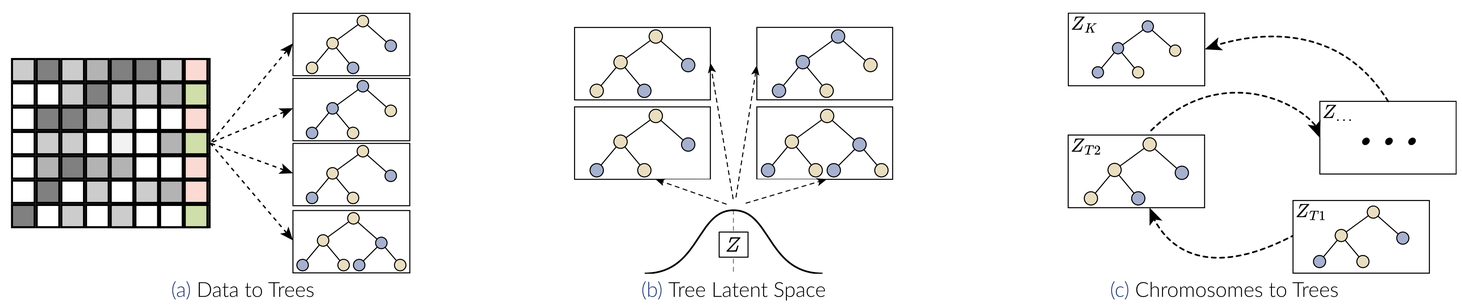
Chromosomes, the Deep Learning way

Genetic algorithms are only as good as the chromosomes they use, and i) choosing encodings, and ii) optimizing over such predefined representations has been found to be difficult. Idea: do not define representations, **learn** them!

GenTree introduces the Tree-VAE, a Variational Autoencoder learning a sample-able dense space of Decision Trees.



GenTree at a glance



(a) Data to Trees (b) Tree Latent Space (c) Chromosomes to Trees

Figure: The GenTree algorithm: given a dataset, GenTree generates a set of Decision Trees, learns a continuous representation of them through the T-VAE, then samples representations and optimizes them through a genetic algorithm.

GenTree

GenTree is a Decision Tree induction algorithm, like CART and C5, which induces by:

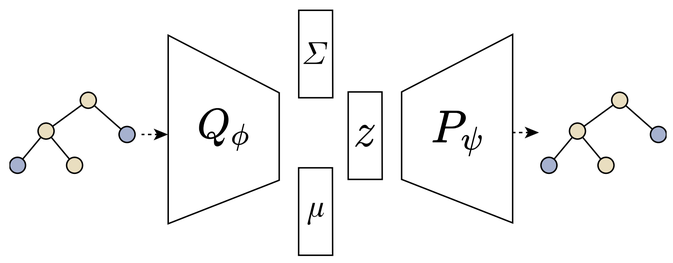
1. generating a set of Decision Trees T from a given training dataset;
2. learning a Decision Tree latent space $\mathcal{Z}$ and thus representations of T;
3. optimizing trees Z_T sampled out of this space.

1: Data to Trees: Generating a Population

Given a training dataset, GenTree leverages greedy induction algorithms to generate a large set of Decision Trees, either through ensambling (boosting/bagging) or perturbation.

2: Learning a Decision Tree Latent Space

The latent space is learned through a *novel* Decision Tree Variational Autoencoder (T-VAE) which, given a Tree t, first embeds it into a matrix, then learns an *encoder* $Q_\phi : \mathcal{T} \to \mathcal{Z}$, which maps Trees to a continuous representation; and a *decoder* $P_\psi : \mathcal{Z} \to \mathcal{T}$, which maps a representation back into a Tree.



GenTree samples an initial population of Trees $z_1, \dots, z_k$ from $\mathcal{Z}$, which are then fed to a genetic algorithm, which in turn optimizes them to extract an accurate yet simple Tree.

3: Chromosomes to Trees

Sampling from the latent space, a genetic algorithm looks to find chromosomes yielding simple yet accurate Decision Trees. When evolving, Trees are rewarded for low complexity and high accuracy through the following fitness function:

$$1 - accuracy(T) + \omega \cdot size(T).$$

where ω controls the importance of the complexity. Trees with low fitness value are culled.

Experiments

Results

- **DT**: greedy Decision Tree
- **GT**: GenTree
- **GS**: GeneSim
- **ODT**: Optimal Decision Tree
- **RF**: Random Forest

	Accuracy ↑					Complexity ↓				
	DT	GT	GS	ODT	RF	DT	GT	GS	ODT	RF
avg	.753	.779	*.813*	.808	**.858**	128.1	62.7	*32.5*	**30.5**	124.3
rank	2.10	1.78	-	*1.43*	**0.99**	2.10	*1.10*	-	**1.06**	1.86

Table: GenTree and competing Decision Tree induction algorithms.

	Accuracy ↑				Complexity ↓			
	DT	GT	ODT	RF	DT	GT	ODT	RF
aus	.829	*.855*	*.855*	**.858**	29.0	**3.0**	**3.0**	26.7
bank	.825	*.891*	**.894**	.881	31.0	*11.4*	**11.0**	30.4
bnk	.916	.915	*.978*	**.996**	43.0	**11.0**	*19.0*	40.8
brst	.864	.914	.929	**.957**	41.0	**3.4**	*7.0*	42.1
car	.866	.875	*.913*	**.954**	79.0	**19.6**	*49.0*	147.0
dnb	.817	.665	**.902**	.897	183.0	**73.6**	*75.0*	169.7
ecoli	.850	.760	**.940**	*.916*	25.0	**3.0**	15.0	26.4
glass	.716	.777	.682	**.855**	23.0	**5.8**	*7.0*	23.8
heart	.764	.784	.780	**.920**	29.0	*6.2*	**3.0**	26.8
iris	.913	**.967**	.933	.933	13.0	**5.0**	*7.0*	12.0
iso	.730	.779	.822	**.933**	*526.2*	815.2	**163.0**	597.4
led7	.733	.753	.795	**.804**	217.0	**51.2**	*57.0*	195.8
lymph	.797	*.800*	*.800*	**.920**	25.0	**5.0**	**5.0**	21.5
pima	.644	.726	.766	**.797**	31.0	*3.6*	**3.0**	28.5
sonar	.711	.695	*.762*	**.900**	39.0	**23.0**	*31.0*	40.0
VCL	.649	.678	**.776**	*.764*	217.8	**41.8**	*49.0*	194.7
wine	.482	.481	.503	*.595*	660.2	*39.6*	*11.0*	528.6
yeast	.509	.519	*.530*	**.572**	95.0	**28.4**	*35.0*	86.3

Table: Methods accuracy and complexity. Average score and average rank position are reported on the bottom.

T-VAE & Decision Tree Latent Space

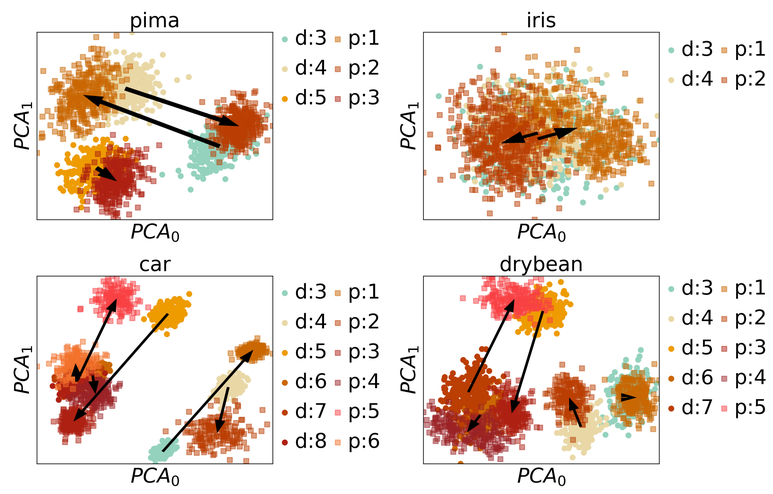


Figure: GenTree latent tree spaces shown with two principal components for 1000 decision trees and their pruned counterpart. Different colors for different depths. Depths (after ``:'') are colored in different ways.

Highlights

- Two-step approach to Tree induction: representation and optimization!
- Better than greedy Trees...
- ...and faster than optimal Trees!

Future Artificial Intelligence Research
SOBIGDATA RESEARCH INFRASTRUCTURE
EXPLANATION OF AI DECISION MAKING.
MIMOSA