Black Tree AutoML

SECURITIES: This booklet is not an offer, or a solicitation for an offer, to enter into any transaction. It is solely for informational purposes, only to describe a set of algorithms that implement machine learning and deep learning (the "algorithms").

Black Tree vs. Neural Networks (Runtime)

Black Tree is as far as I know the most efficient machine learning software known to mankind:

Dataset	Black Tree Runtime*	Neural Network Runtime**
UCI Iris	.045 SECONDS	1.47 HOURS
UCI Wine	0.961 SECONDS	1.73 HOURS
UCI Ionosphere	.145 SECONDS	2.28 HOURS

*SOURCE: "Vectorized Deep Learning"

**SOURCE: "Complexity Analysis of Multilayer Perceptron Neural Network Embedded into a Wireless Sensor Network"

Black Tree vs. Neural Networks (Accuracy)

Black Tree is as far as I know some of the most accurate machine learning software known to mankind:

Dataset	Black Tree Accuracy*
UCI Iris	94.168 %
UCI Wine	94.792 %
UCI Ionosphere	96.835 %

*SOURCE: "Vectorized Deep Learning"

For context, traditional machine learning methods achieved accuracies that were generally in the range of 90 % to 94 %, when applied to 121 UCI datasets.**

**SOURCE: "Do We Need Hundreds of Classifiers to Solve Real World Classification Problems?"

How did you do it?

How did you do it?

Black Tree makes maximum use of **parallel computing**, and **data compression**, producing what is as far as I know, mankind's fastest machine learning software, that is also among the most accurate software.

Serial vs. Parallel Computation

For example:

Calculate, A = 1+2+4+5+7+9

Serial Computation

Add each term in order

```
step 1:1+2
step 2: +4
step 3: +5
step 4: +7
step 5: +9

result: 28
```

Parallel Computation

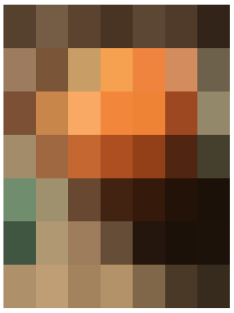
```
Add each pair simultaneously

step 1:          1+2; 4+5; 7+9
step 2:                    3+9
step 3:                   +16
-------------------------------
result:                     28
```

So in serial, 5 steps are required, whereas in parallel, only 3 steps are required. By making maximum use of **parallel computing**, Black Tree minimizes the number of calculations necessary to complete tasks.

Data Compression

A simple iPhone photo can contain millions of pixels. For example, the image on the left contains 7,990,272 pixels. This means comparing two images could easily require billions of calculations, which adds up when you're comparing thousands of images, which is common in machine learning. In contrast, the image on the right contains just 49 cells, after being compressed by my software, which is just 0.0006132457 % of the data in the original image.



So what happens after you get funding?

So what happens after you get funding?

1. I'm going to build a professional website, myself.

2. I'm going to rewrite three versions of my software for Apple's App Store (this is already started): a limited free version (almost complete), and two professional versions that will cost $5 and $50 per month, with more power for more money.

And that's really it.