Do Simpler Models Exist and How Can We Find Them?

Cynthia Rudin
Professor of Computer Science, Electrical and Computer Engineering, and Statistical Science
Duke University
Perhaps we are using complicated models when we don't need them.

The New York Times

OP-ED CONTRIBUTOR

When a Computer Program Keeps You in Jail

By Rebecca Wexler

June 13, 2017

Glenn Rodriguez was denied parole because of a miscalculated “COMPAS” score.

How accurate is COMPAS? Data from Florida can tell us...
COMPAS vs. CORELS

COMPAS: (Correctional Offender Management Profiling for Alternative Sanctions)

CORELS: (Certifiably Optimal Rule Lists, with Elaine Angelino, Nicholas Larus-Stone, Daniel Alabi, and Margo Seltzer, KDD 2017 & JMLR 2018)

Here is the machine learning model:

If age=19-20 and sex=male, then predict arrest
else if age=21-22 and priors=2-3 then predict arrest
else if priors >3 then predict arrest
else predict no arrest
Prediction of re-arrest within 2 years
If age=19-20 and sex=male, then predict arrest
else if age=21-22 and priors=2-3 then predict arrest
else if priors >3 then predict arrest
else predict no arrest
Perhaps we are using complicated models when we don't need them.

There’s no benefit from complicated models for re-arrest prediction in criminal justice.


There’s no benefit from complicated models for lots of problems.

- Sleep apnea screening
- Energy grid reliability (underground power events)
- Adult ADHD screening
- Seizure prediction in ICU patients
- Financial risk assessment
- Crime series detection

Depends on data representation.
Perhaps we are using complicated models when we don't need them.

There’s no benefit from complicated models for lots of problems.
Perhaps we are using complicated models when we don't need them. There’s no benefit from complicated models for lots of problems.
Perhaps we are using complicated models when we don't need them.

There’s no benefit from complicated models for lots of problems.

So why are we using complicated models?

- They are profitable
- They are much easier to construct than “simpler” models.
Perhaps we are using complicated models when we don't need them. There’s no benefit from complicated models for lots of problems.

So why are we using complicated models?

• They are profitable
• They are much easier to construct than “simpler” models.
Perhaps we are using complicated models when we don't need them. There's no benefit from complicated models for lots of problems.

So why are we using complicated models?

- They are much easier to construct than “simpler” models.
Optimization for accurate models:

\[
\min_{f \in \mathcal{F}} \sum_{i=1}^{n} \text{loss}\left(f(x_i), y_i\right)
\]

Optimization for accurate and simple models:

\[
\min_{f \in \mathcal{F}} \sum_{i=1}^{n} \text{loss}\left(f(x_i), y_i\right)
\text{ such that } \text{Complexity}(f) \leq \theta
\]
Optimization for accurate models:

\[
\min_{f \in \mathcal{F}} \sum_{i=1}^{n} \text{loss}(f(x_i), y_i)
\]

Optimization for accurate and simple models:

\[
\min_{f \in \mathcal{F}} \sum_{i=1}^{n} \text{loss}(f(x_i), y_i) \quad \text{such that Complexity}(f) \leq \theta
\]

Accurate Decision Tree
Accurate Linear Model
Accurate Neural Network or Boosted Decision Tree

Accurate & Sparse Decision Tree
Accurate & Sparse Linear Model
Accurate & Sparse Decision Tree?
Accurate & Sparse Linear Model?
Interpretable Neural Network?
Optimization for accurate models:

\[
\min_{f \in F} \sum_{i=1}^{n} \text{loss}(f(x_i), y_i) = \min_{f \in F} \sum_{i=1}^{n} \text{loss}(f(x_i), y_i)
\]

such that Complexity(f) \leq \theta

Optimization for accurate and simple models:

Accurate Neural Network or Boosted Decision Tree?  
Accurate & Sparse Decision Tree?  
Accurate & Sparse Linear Model?  
Interpretable Neural Network?
Optimization for accurate models:

\[
\min_{f \in \mathcal{F}} \sum_{i=1}^{n} \text{loss}(f(x_i), y_i)
\]

Optimization for accurate and simple models:

\[
\min_{f \in \mathcal{F}} \sum_{i=1}^{n} \text{loss}(f(x_i), y_i) \quad \text{such that Complexity}(f) \leq \theta
\]

Question: Can we determine this without solving that?

Can we determine the existence of a simple accurate model without actually finding one?
In this talk

• Define a condition under which a simple-yet-accurate model is likely to exist.

• Computationally efficient solutions to some of these hard optimization problems.

Joint work with: Lesia Semenova, Ron Parr, Sean Xiyang Hu, Margo Seltzer, Chaofan Chen, Oscar Li, Berk Ustun, and many others.
• Define a condition under which a simple-yet-accurate model is likely to exist

If the true Rashomon set is large, a simple-yet-accurate model is likely to exist.

The true Rashomon set is the set of models with low true loss.

$$\{ f \in \mathcal{F} \text{ such that } \mathbb{E}_{(x,y)} \text{loss}(f(x), y) \leq \theta \}$$
Rashomon Set theory: In a sea of equally accurate models, maybe there’s a good one.

If the **true Rashomon set** is large, a simple-yet-accurate model is likely to exist.
If the true Rashomon set is large, a simple-yet-accurate model is likely to exist.

The true Rashomon set is the set of models with low true loss.

\[ \{ f \in \mathcal{F} \text{ such that } \mathbb{E}_{(x,y)} \text{loss}(f(x), y) \leq \theta \} \]
If the true Rashomon set is large, a simple-yet-accurate model is likely to exist.

\[
\mathbb{E}_{(x,y)} \text{loss}(f(x), y)
\]

The true Rashomon set is the set of models with low true loss.

\[
\left\{ f \in \mathcal{F} \text{ such that } L(f) \leq \theta \right\}
\]
If the true Rashomon set is large, a simple-yet-accurate model is likely to exist.

The true Rashomon set is the set of models with low true loss. 

\[ \{ f \in \mathcal{F} \text{ such that } L(f) \leq \theta \} \]
Finite hypothesis spaces: $\mathcal{F}_1$ (simple models) and $\mathcal{F}_2$ (all models) $\mathcal{F}_1 \subset \mathcal{F}_2$, where $\mathcal{F}_1$ is uniformly drawn from $\mathcal{F}_2$ without replacement loss bounded by $b$

$$f_2^* \in \arg\min_{f_2 \in \mathcal{F}_2} \mathbb{E}_{(x,y)} \text{loss}(f_2(x), y), \quad \hat{f}_1 \in \arg\min_{f_1 \in \mathcal{F}_1} \frac{1}{n} \sum_{i=1}^{n} \text{loss}(f_1(x_i), y_i)$$

$$f_2^* \in \arg\min_{f_2 \in \mathcal{F}_2} L(f_2), \quad \hat{f}_1 \in \arg\min_{f_1 \in \mathcal{F}_1} \hat{L}(f_1)$$

Want best true risk of the complex class to be close to best empirical risk of the simpler class.

$$\left| L(f_2^*) - \hat{L}(\hat{f}_1) \right| \leq ...$$
Finite hypothesis spaces: $\mathcal{F}_1$ (simple models) and $\mathcal{F}_2$ (all models) 
$\mathcal{F}_1 \subset \mathcal{F}_2$, where $\mathcal{F}_1$ is uniformly drawn from $\mathcal{F}_2$ without replacement
loss bounded by $b$

\[
\hat{f}_1 \in \arg\min_{f_1 \in \mathcal{F}_1} \frac{1}{n}\sum_{i=1}^{n} \text{loss}(f_1(x_i), y_i) \\
\hat{f}_2 \in \arg\min_{f_2 \in \mathcal{F}_2} \mathbb{E}_{(x,y)} \text{loss}(f_2(x), y), \quad f_2^* \in \arg\min_{f_2 \in \mathcal{F}_2} L(f_2), \quad \hat{f}_1 \in \arg\min_{f_1 \in \mathcal{F}_1} \hat{L}(f_1)
\]

Want best true risk of the complex class to be close to best empirical risk of the simpler class.

Rashomon Ratio $(\mathcal{F}_2, \theta) := \frac{\#\{f_2 \in \mathcal{F}_2 \text{ such that } L(f_2) \leq \theta \}}{|\mathcal{F}_2|}$

Rashomon Ratio is the fraction of models that are good.
Finite hypothesis spaces $\mathcal{F}_1$ and $\mathcal{F}_2$, where $\mathcal{F}_1$ is uniformly drawn from $\mathcal{F}_2$ without replacement loss bounded by $b$

Want best true risk of the complex class to be close to best empirical risk of the simpler class.

$$ L(f_2^*), \text{ where } f_2^* \in \arg\min_{f_2 \in \mathcal{F}_2} L(f_2), \quad \hat{L}(\hat{f}_1) \text{ where } \hat{f}_1 \in \arg\min_{f_1 \in \mathcal{F}_1} \hat{L}(f_1) $$

Rashomon Ratio $(\mathcal{F}_2, \theta) := \frac{\#\{f_2 \in \mathcal{F}_2 \text{ such that } L(f_2) \leq \theta \}}{|\mathcal{F}_2|}$

**Theorem.** For any $\epsilon > 0$, with probability at least $(1 - \epsilon)p$, with respect to the random draw of functions from $\mathcal{F}_2$ to form $\mathcal{F}_1$, and with respect to the random draw of iid data:

$$ \left| L(f_2^*) - \hat{L}(\hat{f}_1) \right| \leq \theta + 2b \sqrt{\frac{\log|\mathcal{F}_1| + \log 2}{2n}} \epsilon, \text{ where } p = 1 - \left( \frac{1 - \text{Rashomon Ratio}(\mathcal{F}_2, \theta)}{|\mathcal{F}_2|} \right) \left( \frac{|\mathcal{F}_1|}{|\mathcal{F}_2|} \right) \left( \frac{|\mathcal{F}_2|}{|\mathcal{F}_1|} \right). $$
Example: If $|\mathcal{F}_2| = 100000$, Rashomon Ratio $\geq 1\%$, bound holds with 99% probability when $|\mathcal{F}_1| \geq 526$.

Example: If $|\mathcal{F}_2| = 100000$, Rashomon Ratio $\geq 0.5\%$, bound holds with 99% probability when $|\mathcal{F}_1| \geq 1051$.

**Theorem.** For any $\epsilon > 0$, with probability at least $(1 - \epsilon)p$, with respect to the random draw of functions from $\mathcal{F}_2$ to form $\mathcal{F}_1$, and with respect to the random draw of iid data:

$$
\left| L(f^*_2) - \hat{L}(\hat{f}_1) \right| \leq \theta + 2b \sqrt{\frac{\log|\mathcal{F}_1| + \log 2}{2n} \frac{1}{\epsilon}}, \quad \text{where } p = 1 - \frac{1 - \text{Rashomon Ratio}(\mathcal{F}_2, \theta) |\mathcal{F}_2|}{|\mathcal{F}_1|}
$$
If the Rashomon Ratio is sufficiently large, then with high prob, the best empirical risk over the simpler class $\mathcal{F}_1$ is close to the best possible true risk over the larger class $\mathcal{F}_2$.

The generalization guarantee comes from $\mathcal{F}_1$.

**Theorem.** For any $\epsilon > 0$, with probability at least $(1 - \epsilon)p$, with respect to the random draw of functions from $\mathcal{F}_2$ to form $\mathcal{F}_1$, and with respect to the random draw of iid data:

$$
\left| L(f^*_2) - \hat{L}(\hat{f}_1) \right| \leq \theta + 2b\sqrt{\frac{\log|\mathcal{F}_1| + \log 2}{2n}} + \frac{1 - \text{Rashomon Ratio}(\mathcal{F}_2, \theta)}{|\mathcal{F}_1|} \left( \frac{|\mathcal{F}_2|}{|\mathcal{F}_1|} \right),
$$

where $p = 1 - \frac{1}{|\mathcal{F}_2|}$. 


Rashomon ratio compared to complexity measures?

Large Rashomon ratios pertain to the existence of models with good generalization

I showed one (of several) generalization bounds that includes the Rashomon Ratio.

The Rashomon ratio is:

- Not the (geometric) margin...
  Margin is measured with respect to one model.  
  Rashomon ratio is measured with respect to many.

- Not the VC dimension...
  VC dimension is data independent.  
  Rashomon ratio is a property of a loss function on a dataset.

- Not algorithmic stability...
  Stability is a property of an algorithm.  
  Rashomon ratio is a property of a loss function on a dataset.

- Not Rademacher complexity...
  Rademacher complexity measures ability to fit noisy targets.  
  Rashomon ratio uses fixed labels.

- Not a flat minimum...
  Rashomon set could consider many local minima
• The next theorem also guarantees the existence of a simpler model that is in the Rashomon set and generalizes well.

• The conditions of the theorem are: Lipschitz loss, the simpler class $\mathcal{F}_1$ is a good approximating set for $\mathcal{F}_2$, and the Rashomon set is large.

• The proof is almost trivial (it’s self-evident).
Theorem

For a $K$-Lipschitz loss $l$ bounded by $b$, hypothesis spaces $\mathcal{F}_1$ and $\mathcal{F}_2$, $\mathcal{F}_1 \subset \mathcal{F}_2$, if for each $f_2 \in$ Rashomon set($\mathcal{F}_2, \theta$) there exists a model $f_1 \in \mathcal{F}_1$ such that $\| f_2 - f_1 \|_p \leq \delta$, and if the Rashomon set is large, in that it contains an $\ell_p$ ball of size at least $\delta$, then there exists $\overline{f}_1 \in$ Rashomon set($\mathcal{F}_2, \theta$) such that for a fixed parameter $\epsilon \in (0,1)$:

1. $\overline{f}_1$ is from simpler class $\mathcal{F}_1$.

2. With prob $\geq 1-\epsilon$ w.r.t. the random draw of training data,

$$
\left| L(\overline{f}_1) - \hat{L}(\overline{f}_1) \right| \leq 2KR_n(\mathcal{F}_1) + b\sqrt{\frac{\log(2/\epsilon)}{2n}},
$$

where $R_n(\mathcal{F}_1)$ is Rademacher complexity.
These results suggest:

- If $\mathcal{F}_1$ is a good approximating set for $\mathcal{F}_2$, and the Rashomon set is large, then we might as well work with the simpler class.

- E.g., if decision trees approximate SVM-RBF and for my problem, and the Rashomon set is large, I can work with decision trees.

These results can be generalized (under similar conditions) to say that all functions in $\mathcal{F}_2$ generalize, based on proximity to functions in $\mathcal{F}_1$. 
What do Rashomon Ratios look like in reality?

• Let’s find out!  

\[
\hat{\text{Rashomon Ratio}} (\mathcal{F}, \theta) := \frac{\# \{ f \in \mathcal{F} \text{ such that } \hat{L}(f) \leq \theta \}}{|\mathcal{F}|}
\]

You’d never calculate this in reality… But we will today!
An experiment

• Calculate the size of the empirical Rashomon set using decision trees of depth 7.
  • Why? Decision trees can be sampled. Depth 7 trees are a good approximating set because they are able to fit our datasets.

\[ \mathcal{F} \supset \mathcal{F}_{\text{tree}}, \mathcal{F}_{\text{svm}}, \mathcal{F}_{\text{boosting}}, \mathcal{F}_{\text{forest}}, \text{etc} \]
An experiment

• Calculate the size of the empirical Rashomon set using decision trees of depth 7.
  • Why? Decision trees can be sampled. Depth 7 trees are a good approximating set because they are able to fit our datasets.

• Look at the performance of many popular ML methods on the dataset.

• Evaluate: do all the methods perform similarly?

• Evaluate: do the methods generalize?

Short answer: When the Rashomon ratio is large, all methods perform similarly and generalize.
(The reverse isn’t always true.)

$\mathcal{F} \supset \mathcal{F}_{\text{tree}}, \mathcal{F}_{\text{svm}}, \mathcal{F}_{\text{boosting}}, \mathcal{F}_{\text{forest}}, \text{etc}$
If the Rashomon ratio is large, do all the methods perform similarly?

**Large Rashomon ratio:**

- HTRU_2: Rratio 1.0419E-37 %
- Voting: Rratio 9.2687E-38 %
- Nursery-1: Rratio 1.7433E-38 %
- Mammographic masses: Rratio 1.3817E-37 %

**Small Rashomon ratio:**

- Diabetic Retinopathy: Rratio 0.0000E+00 %
- Eye state: Rratio 0.0000E+00 %
- Mushroom: Rratio 1.5649E-39 %
- Wine: Rratio 3.9183E-41 %

36 datasets: 15 UCI categorical + 21 UCI real-valued, with number of features between [3,784]; number of classes between [2,6]
If the Rashomon ratio is large, all the methods perform similarly, generalize well.

Can’t measure Rashomon ratio in practice? That’s ok.

If all methods perform similarly, they tend to generalize well. (Could be due to a large Rashomon ratio.)

If methods perform differently, likely a small Rashomon ratio.
The Rashomon Curve

\[ \hat{R}_{\mathrm{Rashomon}}(\mathcal{F}_i, \theta) := \frac{\# \{ f \in \mathcal{F}_i \text{ such that } \hat{L}(f) \leq \theta \}}{|\mathcal{F}_i|} \]

- \( \mathcal{F}_1 \) simplest class
- \( \mathcal{F}_2 \)
- ...
- \( \mathcal{F}_{t-1} \)
- \( \mathcal{F}_t \) most complex class

\( \mathcal{F}_1 \subset \mathcal{F}_2 \subset \ldots \subset \mathcal{F}_{t-1} \subset \mathcal{F}_t \)
The Rashomon Curve

\[ F_1 \subset F_2 \subset \ldots \subset F_{t-1} \subset F_t \]

- $F_1$ simplest class
- $F_2$
- ...
- $F_{t-1}$
- $F_t$ most complex class

log Rashomon Ratio, %

Empirical risk, $\hat{L}$
The Rashomon Curve

\[ \text{log Rashomon Ratio, } \% \]
\[ \text{Empirical risk, } \hat{L} \]

\[ \text{log Rashomon Ratio, } \% \]
\[ \text{Empirical risk, } \hat{L} \]

\[ \text{log Rashomon Ratio, } \% \]
\[ \text{Empirical risk, } \hat{L} \]

\[ * F_1 \text{ simplest class} \]
\[ F_2 \]
\[ \ldots \]
\[ F_{i-1} \]
\[ F_i \text{ most complex class} \]
The Rashomon Curve

The Rashomon Curve is a graphical representation that shows the behavior of the log Rashomon ratio, $\log R$, as a function of the empirical risk, $\hat{L}$, for different complexity classes $\mathcal{F}_i$. The graphs illustrate how the log Rashomon ratio changes with increasing empirical risk for the simplest class $\mathcal{F}_1$, the intermediate class $\mathcal{F}_2$, and the most complex class $\mathcal{F}_i$. The data points indicate the empirical risk values for each complexity class, with different markers for each class.
The Rashomon Curve

The diagram illustrates the Rashomon curve, which shows the relationship between the empirical risk $\hat{L}$ and the log Rashomon ratio for different classes $F_i$. The classes are ordered from the simplest $F_1$ to the most complex $F_i$. The graph compares the performance of each class on the same dataset, highlighting how the empirical risk varies with increasing complexity.
The Rashomon Curve

- $\mathcal{F}_1$ simplest class
- $\mathcal{F}_2$
- ...
- $\mathcal{F}_{i-1}$
- $\mathcal{F}_i$ most complex class

Empirical risk, $\hat{L}$

log Rashomon ratio, %
Experiments to investigate Rashomon Curve

- A hierarchy of hypothesis spaces: decision trees, depths 1 through 7
- Sampled 2.25 million trees for each decision tree depth
- **36 datasets**: 15 UCI categorical + 21 UCI real-valued, with number of features between [3,784]; number of classes between [2,6]
- Rashomomon parameter: $\theta = 5\%$ above best across all spaces
36 datasets: 15 UCI categorical + 21 UCI real-valued
36 datasets: 15 UCI categorical + 21 UCI real-valued
That was the training set, what about the test set?
That was the training set, what about the test set?
Experiments to investigate Rashomon Curve Generalization

- A hierarchy of hypothesis spaces: linear models with increasing number of polynomial features, regression with squared loss
- Rashomon Ratio computed analytically. (It’s the volume of an ellipsoid.)
Sometimes theory is insightful, but sometimes we always generalize and sometimes we never generalize.

The elbow model always seems to be a good choice for model selection.
Where are we on this curve?
ML algorithms perform differently

Where are we on this curve?
Where are we on this curve?

ML algorithms perform similarly
Are Different Problems on Their Own Rashomon Curves?

“Rashomon Ratio”

High accuracy

“Empirical Risk”

Low accuracy

MNIST (~100%)

ImageNet (~97%)

Recidivism Prediction (~74%)

Credit Scoring

Stroke Prediction

Diabetes prediction

Pneumonia prediction

Various medical imaging problems
An Easy Check for the Possible Presence of a Simpler-Yet-Accurate Model
An Easy Check for the Possible Presence of a Simpler-Yet-Accurate Model

Pick several of your favorite ML methods. Run them all on the data.
An Easy Check for the Possible Presence of a Simpler-Yet-Accurate Model

Pick several of your favorite ML methods. Run them all on the data.

If they all perform differently: model class too small to include elbow solution.

**Broaden the horizon**: use a more complex model class.
An Easy Check for the Possible Presence of a Simpler-Yet-Accurate Model

Pick several of your favorite ML methods. Run them all on the data.

If they all perform similarly: model class might be larger than necessary.

Delve in: find specialized models with specific properties, such as interpretability.

Look up: improve generalization without reducing accuracy, move towards the elbow by decreasing complexity.
In this talk

- Define a condition under which a simple-yet-accurate model is likely to exist.

  A study in Rashomon curves and volumes: A new perspective on generalization and model simplicity in machine learning (Semenova, Rudin, Parr, in progress)

- Computationally efficient solutions to some of these hard optimization problems.
FICO Explainable ML Challenge 2018

Play around with our entry at http://dukedatasciencefico.cs.duke.edu
Play around with our entry at http://dukedatasciencefico.cs.duke.edu
Play around with our entry at http://dukedatasciencefico.cs.duke.edu
Play around with our entry at http://dukedatasciencefico.cs.duke.edu
FICO ML Challenge

Play around with our entry at http://dukedatasciencefico.cs.duke.edu
The team representing Duke University, which included Chaofan Chen, Kangcheng Lin, Cynthia Rudin, Yaron Shaposhnik, Sijia Wang and Tong Wang, received the FICO Recognition Award acknowledging their submission for going above and beyond expectations with a fully transparent global model and a user-friendly dashboard to allow users to explore the global model and its explanations. The Duke team took home $3,000.
Dear XXX@Stanford.edu,

<blah>... We don’t know whether our paper fits into the scope of the special issue. It’s not a traditional methodology paper, it’s contribution is an analysis of the FICO data, including a machine learning model that is interpretable, ... The paper’s content actually won the FICO Challenge Award for the first Explainable Machine Learning Challenge.

Response:
Dear Cynthia,
Thanks for reaching out. This is an interesting paper... . But I’m afraid its not a good fit for the special issue. ... Its also related to my own recent work on explainability of neural nets. ... Is the FICO data still available? If so, could you share it?

My (silent) response: Whoa, can I frame this ridiculous email?
Projects in the Prediction Analysis Lab @ Duke

• Optimal Decision Trees and Decision Lists

• Interpretable Neural Networks for Computer Vision

• Optimal Sparse Linear Models with Integer Coefficients

• Fast Large-Scale Learning-to-Match for Causal Inference
Optimal Sparse Decision Trees

\[ \hat{L}(\text{tree}, \{(x_i, y_i)\}_i) = \frac{1}{n} \sum_{i=1}^{n} 1_{\text{tree}(x_i) \neq y_i} + C(\# \text{leaves in tree}) \]

CART & C4.5 do not minimize this.

It’s hard.

The key: very efficient branch & bound combined with computer systems.

Here is an optimal model on the re-arrest data.
Optimal Sparse Decision Trees

With Xiyang Sean Hu, Jimmy Lin, & Margo Seltzer

The Iris Data Set (Predictions)
Optimal Sparse Decision Trees

- Depth first search (but with bounds)

**Theorem** (Lower bound on node support): For an optimal tree, the support of each node must be above $2C$. 

Node support not sufficient to lead to an optimal solution.
Optimal Sparse Decision Trees

**Theorem** (Lower bound on classification accuracy): Each leaf of an optimal tree correctly classifies at least fraction $C$ of the data.
Theorem (Permutation bound): If two trees have the same leaves, up to a permutation, all their child trees will be the same, so one of them can be pruned.

Optimal Sparse Decision Trees

Two “symmetry-aware maps” to keep track of leaf and tree symmetries.
Optimal Sparse Decision Trees

• Several other theorems
• Bit-vector representation of trees as a collection of leaves
• Specialized bit-vector library to evaluate decision trees fast
• We observe runtime to be linear in the amount of data
• We observe runtime to be exponential in the first few features, and subexponential for more features because of our theorems.
• This extends our work on Certifiably Optimal Rule Lists (CORELS).
  • https://corels.eecs.harvard.edu
  • Python and R interfaces to C++.
Optimal Sparse Decision Trees

- Several other theorems
- Bit-vector representation of trees as a collection of leaves
- Specialized bit-vector library to evaluate decision trees fast
- We observe runtime to be linear in the amount of data
- We observe runtime to be exponential in the first few features, and subexponential for more features because of our theorems.
- This extends our work on Certifiably Optimal Rule Lists (CORELS).
  - https://corels.eecs.harvard.edu
  - Python and R interfaces to C++.
Projects in the Prediction Analysis Lab @ Duke

• Optimal Decision Trees and Decision Lists
  OSDT (Hu, Rudin Seltzer, in progress)

• Interpretable Neural Networks for Computer Vision

• Optimal Sparse Linear Models with Integer Coefficients

• Fast Large-Scale Learning-to-Match for Causal Inference
Optimization for accurate models:

\[
\min_{f \in \mathcal{F}} \sum_{i=1}^{n} \text{loss}(f(x_i), y_i) = \min_{f \in \mathcal{F}} \sum_{i=1}^{n} \text{loss}(f(x_i), y_i)
\]

such that Complexity\((f) \leq \theta\)

Optimization for accurate and simple models:

\[
\min_{f \in \mathcal{F}} \sum_{i=1}^{n} \text{loss}(f(x_i), y_i) = \min_{f \in \mathcal{F}} \sum_{i=1}^{n} \text{loss}(f(x_i), y_i)
\]

Accurate Neural Network vs. Interpretable Neural Network?

"K-nearest patches of prototypical neighbors"
"This Looks Like That: deep learning for interpretable image recognition"

Chaofan Chen, Oscar Li, Alina Barnett, Jonathan Su, Cynthia Rudin

(Submitted on 27 Jun 2018)

When we are faced with challenging image classification tasks, we often explain our reasoning by dissecting the image, and pointing out prototypical aspects of one class or another. The mounting evidence for each of the classes helps us make our final decision. In this work, we introduce a deep network architecture that reasons in a similar way: the network dissects the image by finding prototypical parts, and combines evidence from the prototypes to make a final classification. The algorithm thus reasons in a way that is qualitatively similar to the way ornithologists, physicians, geologists
looks like looks like looks like looks like

Leftmost column: a test image of a clay-colored sparrow

Second column: same test image, each with a bounding box generated by our model -- the content within the bounding box is considered by our model to look similar to the prototypical part (same row, third column) learned by our algorithm

Third column: prototypical parts learned by our algorithm

Fourth column: source images of the prototypical parts in the third column

Rightmost column: activation maps indicating how each prototypical part resembles part of the test bird
Convolutional layers $f$  
Prototype layer $g_p$  
Fully connected layer $h$  
Output logits

Similarity score

Black footed albatross
Indigo bunting
Cardinal
Clay colored sparrow
Common yellowthroat
Why is this bird classified as a red-bellied woodpecker?

Evidence for this bird being a red-bellied woodpecker:

<table>
<thead>
<tr>
<th>Original image</th>
<th>Prototype (box showing part that looks like prototype)</th>
<th>Training image where prototype comes from</th>
<th>Activation map</th>
<th>Similarity score</th>
<th>Class connection points</th>
<th>contributed</th>
</tr>
</thead>
<tbody>
<tr>
<td><img src="image1.png" alt="Original Image" /></td>
<td><img src="image2.png" alt="Prototype" /></td>
<td><img src="image3.png" alt="Training Image" /></td>
<td><img src="image4.png" alt="Activation Map" /></td>
<td>6.499</td>
<td>1.180</td>
<td>7.669</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>4.392</td>
<td>1.127</td>
<td>4.950</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>3.890</td>
<td>1.108</td>
<td>4.310</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Total points to red-bellied woodpecker: 32.736

Evidence for this bird being a red-cockaded woodpecker:

<table>
<thead>
<tr>
<th>Original image</th>
<th>Prototype (box showing part that looks like prototype)</th>
<th>Training image where prototype comes from</th>
<th>Activation map</th>
<th>Similarity score</th>
<th>Class connection points</th>
<th>contributed</th>
</tr>
</thead>
<tbody>
<tr>
<td><img src="image5.png" alt="Original Image" /></td>
<td><img src="image6.png" alt="Prototype" /></td>
<td><img src="image7.png" alt="Training Image" /></td>
<td><img src="image8.png" alt="Activation Map" /></td>
<td>2.452</td>
<td>1.046</td>
<td>2.565</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>2.125</td>
<td>1.091</td>
<td>2.318</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>1.945</td>
<td>1.069</td>
<td>2.079</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Total points to red-cockaded woodpecker: 16.886
Projects in the Prediction Analysis Lab @ Duke

• Optimal Decision Trees and Decision Lists
  OSDT (Hu, Rudin Seltzer, in progress)

• Interpretable Neural Networks for Computer Vision
  This Looks Like That (Chen et al, in progress)

• Optimal Sparse Linear Models with Integer Coefficients
  RiskSLIM (Ustun, Rudin, KDD’17 & JMLR’19)

• Fast Large-Scale Learning-to-Match for Causal Inference
A critical care patient in the ICU

- Work with Aaron Struck, Brandon Westover & Berk Ustun
- Seizure are common (20%)\(^1\)
- Seizure → Brain Damage\(^2\)
- Need EEG to detect seizures\(^3\)
- EEG is expensive and not well-utilized
- Want to predict seizures in advance to determine who needs EEG monitoring.

Credit: Aaron Struck and Brandon Westover
\(^1\)Ruiz et al JAMA Neuro 2017, \(^2\)De Marchis et al Neuro 2016, \(^3\)Claassen et al Neuro 2004
• Using data from the Critical Care EEG Monitoring Research Consortium (CCEMRC) on 5427 consecutive patients, 70+ EEG variables, we developed the 2HELPS2B score...

### 2HELPS2B Score

<table>
<thead>
<tr>
<th></th>
<th>Score Item</th>
<th>Points</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Any cEEG Pattern with Frequency 2 Hz</td>
<td>1 point</td>
</tr>
<tr>
<td>2</td>
<td>Epileptiform Discharges</td>
<td>1 point</td>
</tr>
<tr>
<td>3</td>
<td>Patterns include [LPD, LRDA, BIPD]</td>
<td>1 point</td>
</tr>
<tr>
<td>4</td>
<td>Patterns Superimposed with Fast or Sharp Activity</td>
<td>1 point</td>
</tr>
<tr>
<td>5</td>
<td>Prior Seizure</td>
<td>1 point</td>
</tr>
<tr>
<td>6</td>
<td>Brief Rhythmic Discharges</td>
<td>2 points</td>
</tr>
</tbody>
</table>

**SCORE**

<table>
<thead>
<tr>
<th>SCORE</th>
<th>0</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6+</th>
</tr>
</thead>
<tbody>
<tr>
<td>RISK</td>
<td>&lt;5%</td>
<td>11.9%</td>
<td>26.9%</td>
<td>50.0%</td>
<td>73.1%</td>
<td>88.1%</td>
<td>95.3%</td>
</tr>
</tbody>
</table>

**Risk:**

- 0-2: <5%
- 2-4: 11.9%
- 4-6: 26.9%
- 6+: 50.0%
- 6+: 73.1%
- 6+: 88.1%
- 6+: 95.3%
Using data from the Critical Care EEG Monitoring Research Consortium (CCEMRC) on 5427 consecutive patients, 70+ EEG variables, we developed the 2HELPS2B score...

- It is just as accurate as black box models.
- Doctors can decide themselves whether to trust it
- **63.6%** reduction in duration of EEG monitoring per patient in 2018
- **2.82 X** More Patients Monitored
- **$6.1M savings** in 2018 at two major hospitals (MGH, Wisconsin)

2HELPS2B Score

<table>
<thead>
<tr>
<th>Score Criteria</th>
<th>Points</th>
</tr>
</thead>
<tbody>
<tr>
<td>Any cEEG Pattern with Frequency 2 Hz</td>
<td>1 point</td>
</tr>
<tr>
<td>Epileptiform Discharges</td>
<td>1 point</td>
</tr>
<tr>
<td>Patterns include [LPD, LRDA, BIPD]</td>
<td>1 point</td>
</tr>
<tr>
<td>Patterns Superimposed with Fast or Sharp Activity</td>
<td>1 point</td>
</tr>
<tr>
<td>Prior Seizure</td>
<td>1 point</td>
</tr>
<tr>
<td>Brief Rhythmic Discharges</td>
<td>2 points</td>
</tr>
</tbody>
</table>

**Score Table**

<table>
<thead>
<tr>
<th>Score</th>
<th>0</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6+</th>
</tr>
</thead>
<tbody>
<tr>
<td>RISK</td>
<td>&lt;5%</td>
<td>11.9%</td>
<td>26.9%</td>
<td>50.0%</td>
<td>73.1%</td>
<td>88.1%</td>
<td>95.3%</td>
</tr>
</tbody>
</table>

2HELPS2B Score

It is just as accurate as black box models. Doctors can decide themselves whether to trust it. **63.6%** reduction in duration of EEG monitoring per patient in 2018. **2.82 X** More Patients Monitored. **$6.1M savings** in 2018 at two major hospitals (MGH, Wisconsin).
Solution to a mixed-integer nonlinear program

Solved with the Lattice Cutting Plane Algorithm (LCPA)

2HELPS2B Score

1. Any cEEG Pattern with Frequency 2 Hz  1 point
2. Epileptiform Discharges  1 point + ...
3. Patterns include [LPD, LRDA, BIPD]  1 point + ...
4. Patterns Superimposed with Fast or Sharp Activity  1 point + ...
5. Prior Seizure  1 point + ...
6. Brief Rhythmic Discharges  2 points + ...

<table>
<thead>
<tr>
<th>SCORE</th>
<th>0</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6+</th>
</tr>
</thead>
<tbody>
<tr>
<td>RISK</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>&lt;5%</td>
<td>11.9%</td>
<td>26.9%</td>
<td>50.0%</td>
<td>73.1%</td>
<td>88.1%</td>
<td>95.3%</td>
<td></td>
</tr>
</tbody>
</table>

Score Table:

- Score 0: Risk <5%
- Score 1: 11.9%
- Score 2: 26.9%
- Score 3: 50.0%
- Score 4: 73.1%
- Score 5: 88.1%
- Score 6+: 95.3%

Sparse linear model with integer coefficients
Risk-Calibrated Supersparse Linear Integer Models (Risk-SLIM)

\[
\min_{\lambda \in \mathcal{L}} \frac{1}{n} \sum_{i=1}^{n} \log \left( 1 + e^{-y_i x_i \lambda} \right) + C_0 \| \lambda \|_0
\]

\( \lambda \in \mathcal{L} \) means that \( \forall j, \lambda_j \in \{-10, -9, \ldots, 0, \ldots, 9, 10\} \)

Ustun and Rudin. Learning Optimized Risk Scores from Large-Scale Datasets. KDD 2017, JMLR 2019 (accepted)
Code: https://github.com/ustunb/risk-slim
Projects in the Prediction Analysis Lab @ Duke

• Optimal Decision Trees and Decision Lists
  OSDT (Hu, Rudin Seltzer, in progress)

• Interpretable Neural Networks for Computer Vision
  This Looks Like That (Chen et al, in progress)

• Optimal Sparse Linear Models with Integer Coefficients
  RiskSLIM (Ustun, Rudin, KDD’17 & JMLR’19)

• Fast Large-Scale Learning-to-Match for Causal Inference
  FLAME/DAME/MALTS (with Volfovsky, Roy & others, AISTATS’19, UAI’19)
FLAME – Fast Large-Scale Almost Matching Exactly

• Match treatment units to control units as closely as possible.
• When data are categorical, for each treatment unit, find a control unit that matches on as many (weighted) covariates as possible.
• Use a training set to determine the weights using machine learning.
• For treatment unit $t$,

\[ v^t \in \arg\max_{v \in \{0,1\}^p} v^t w \text{ such that } \exists \text{ control unit } c \text{ with } x_c^\circ v = x_t^\circ v. \]

Choose as many weighted features as possible so that...

there exists a control unit $c$ that $t$ matches on covariates where $v=1$
In summary...

- Define a condition under which a simple-yet-accurate model is likely to exist.
  - ...which is that the Rashomon set is large
  - I showed a simple check for large Rashomon sets
  - I also introduced Rashomon curves

- Computationally efficient solutions to some of these hard optimization problems.
  - Optimal sparse decision trees (OSDT, CORELS)
  - Interpretable neural networks ("This looks like that")
  - Optimal scoring systems (RiskSLIM)
  - Learning to Match for Causal Inference (FLAME, DAME, MALTS)
Berk Ustun

A study in Rashomon curves and volumes: A new perspective on generalization and model simplicity in machine learning, arXiv 2019

Elaine Angelino

Optimal Sparse Decision Trees, arXiv 2019

Certifiably Optimal Rule Lists for Categorical Data, KDD’17, JMLR’18.

Lesia Semenova

Nicholas Larus-Stone

Sean Xiyang Hu

Margo Seltzer

Ron Parr

This Looks Like That, arXiv 2018

Oscar Li

Chaofan Chen

Learning Optimized Risk Scores from Large-Scale Datasets, KDD’17, JMLR’19 (accepted)