

Interpretable Machine Learning with rsparkling

2019 Symposium on Data Science and Statistics

Navdeep Gill, Senior Data Scientist/Software Engineer @ H2O.ai



Agenda

- What/who is H2O?
- H2O Platform
- H2O Sparkling Water
- sparklyr
- rsparkling
- Interpretable Machine Learning
- Demo

H2O.ai

H2O Company

- Team : 100+. Founded in 2012, Mountain View, CA
 - Stanford Math & Systems Engineers
-

H2O Software

- Open Source Software (<https://github.com/h2oai/h2o-3>)
- Ease of Use via Web Interface (H2O Flow)
- R, Python, Scala, Spark, and Hadoop Interfaces
- Distributed Algorithms Scale to Big Data

Current Algorithm Overview

Statistical Analysis

- Linear Models (GLM)
- Naïve Bayes

Ensembles

- Random Forest
- Distributed Trees
- Gradient Boosting Machine
- Super Learner Ensembles

Deep Neural Networks

- Multi-layer Feed-Forward Neural Network
- Auto-encoder
- Anomaly Detection
- Deep Features

Clustering

- K-Means

Dimension Reduction

- Principal Component Analysis
- Generalized Low Rank Models

Solvers & Optimization

- Generalized ADMM Solver
- L-BFGS (Quasi Newton Method)
- Ordinary Least-Square Solver
- Stochastic Gradient Descent

Data Munging

- Scalable Data Frames
- Sort ,Slice, Log Transform

H2O Components

H2O Cluster

Distributed Key
Value Store

H2O Frame

- Multi-node cluster with share memory model
- All computations are in memory
- Each node only sees some rows of the data
- No limit on cluster size
- Objects in the H2O cluster such as data frames, models and results are all reference by key
- Any node in the cluster can access any object in the cluster by key.
- Distributed data frames (collection of vectors).
- Columns are distributed (across nodes) arrays
- Each node must be able to see the entire dataset (achieved by HDFS, S3, or multiple copies of the data if it is a CSV file).

H2O in Spark


Spark + H₂O

**SPARKLING
WATER**

H2O Sparkling Water

Spark Integration

- Sparkling Water is a transparent integration of H2O into the Spark ecosystem.
- H2O runs inside of the Spark Executor JVM.

Benefits

- Provides advanced machine learning algorithms to Spark workflows.
- Alternative to default Mllib library in Spark.

Sparkling Shell

- Sparkling Shell is just a standard Spark shell with addition Sparkling Water classes.
- Export MASTER="local-cluster[3,2,1024]"
- Spark-shell –jars sparkling-water.jar

<https://github.com/h2oai/sparkling-water>



Sparkling Water Ecosystem

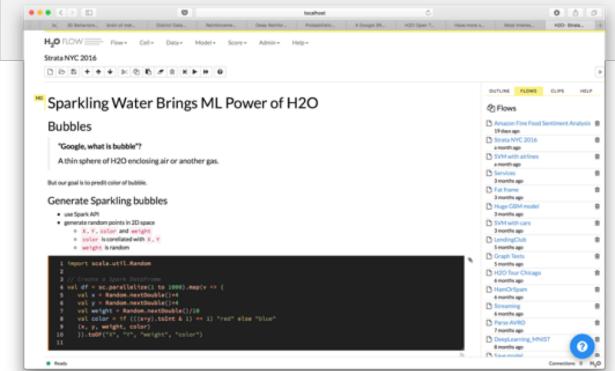
Scala: Sparkling Water

Spark

```
val df = sc.parallelize(1 to 10).toDF
```

```
val h2oContext =  
  H2OContext.getOrCreate(sc)
```

```
val hf = h2oContext.asH2OFrame(df)
```



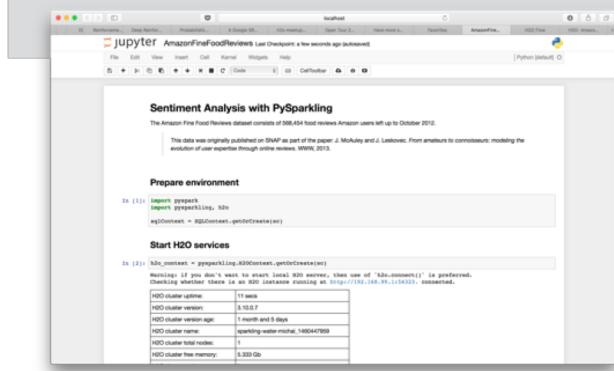
Python: PySparkling Water

```
sc = SparkContext()
```

```
df = sc.parallelize(range(1,11))  
    .toDF("int")
```

```
h2o_context =  
H2OContext.getOrCreate(sc)
```

```
hf = h2o context.as h2o frame(df)
```



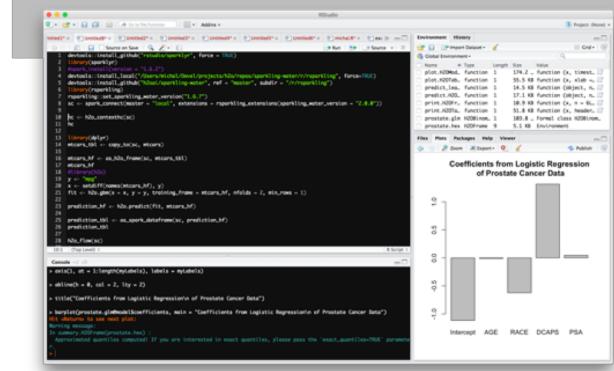
R: RSparkling Water

```
sc <- spark_connect(...)
```

```
tbl <- data_frame(c(1:10))  
df <- copy_to(sc, tbl)
```

```
hc <- h2o_context(sc)
```

```
hf <- as_h2o_frame(sc, df)
```



sparklyr

sparklyr

dplyr

ML

Extensions

Apache Spark

sparklyr

- Connect to Spark from R.
- The sparklyr package provides a complete dplyr backend.
- Filter and aggregate Spark datasets then bring them into R for analysis and visualization.
- Use Spark's distributed machine learning library from R.
- Create extensions that call the full Spark API and provide interfaces to Spark packages.

```
library(sparklyr)
spark_install(version = "2.1.1")
sc <- spark_connect(master = "local")
my_tbl <- copy_to(sc,iris)
```

<https://github.com/rstudio/sparklyr>

rsparkling



+

Spark + H₂O
SPARKLING
WATER

rsparkling

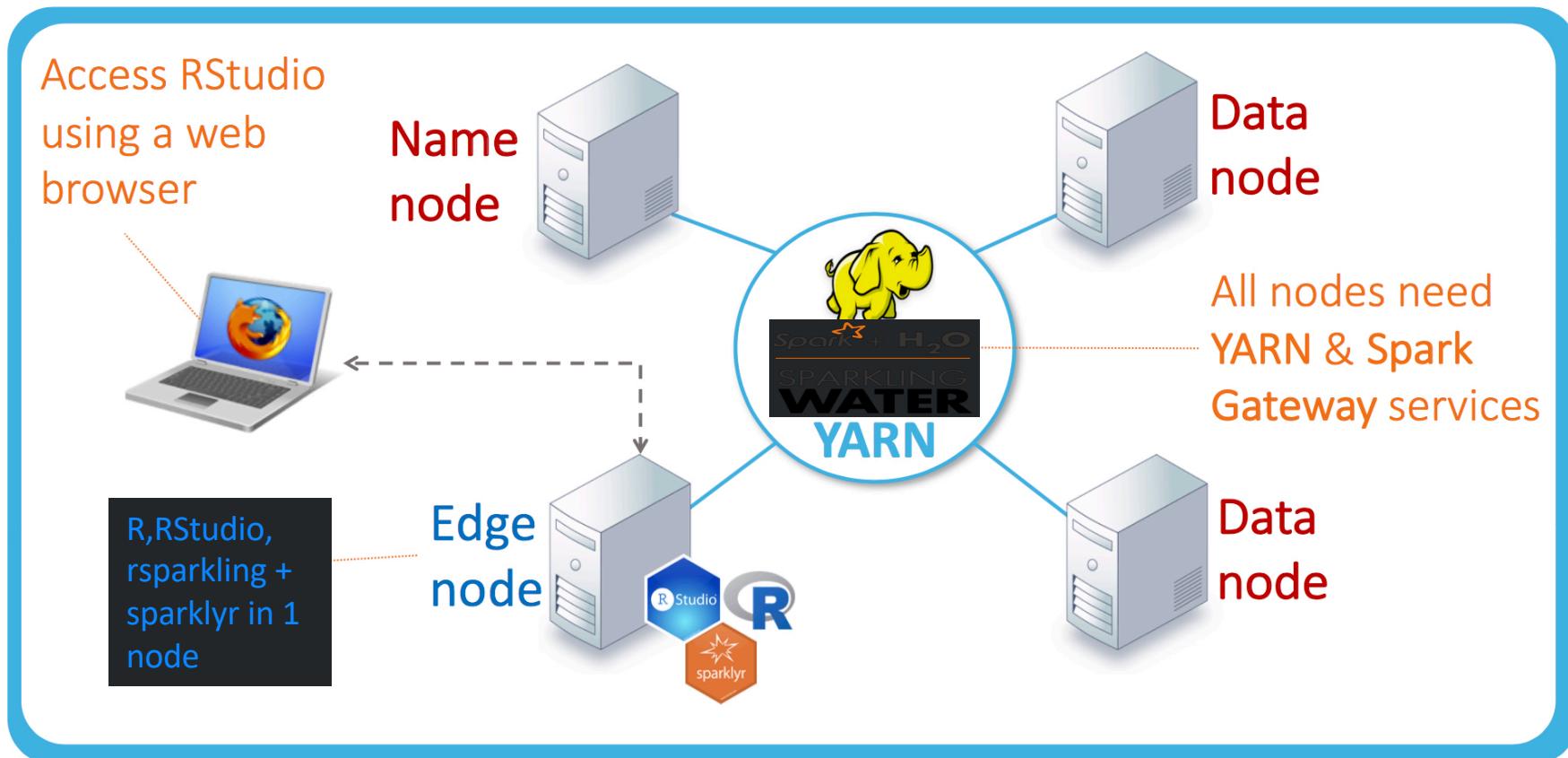
- The rsparkling R package is an extension package for sparkapi / sparklyr that creates an R front-end for a Spark package (Sparkling Water from H2O) .
- This provides an interface to H2O's machine learning algorithms on Spark, using R.
- This package implements basic functionality (creating an H2OContext, showing the H2O Flow interface, and converting between Spark DataFrames and H2O Frames).

```
library(sparklyr)
spark_install(version = "2.0.0")
options(rsparkling.sparklingwater.version = "2.0.0")
library(rsparkling)
sc <- spark_connect(master = "local")
```

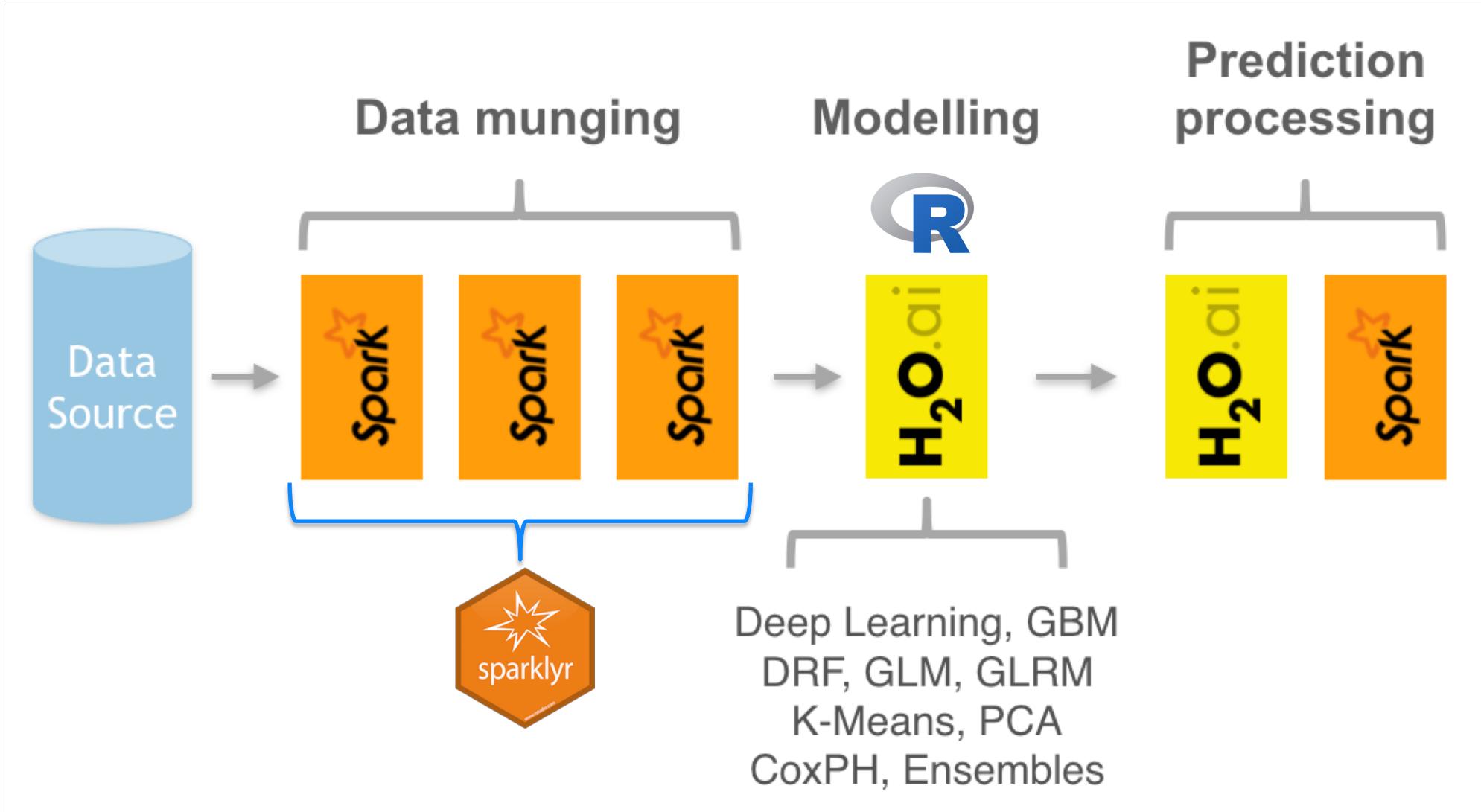
<https://github.com/h2oai/sparkling-water/tree/master/r>

rsparkling

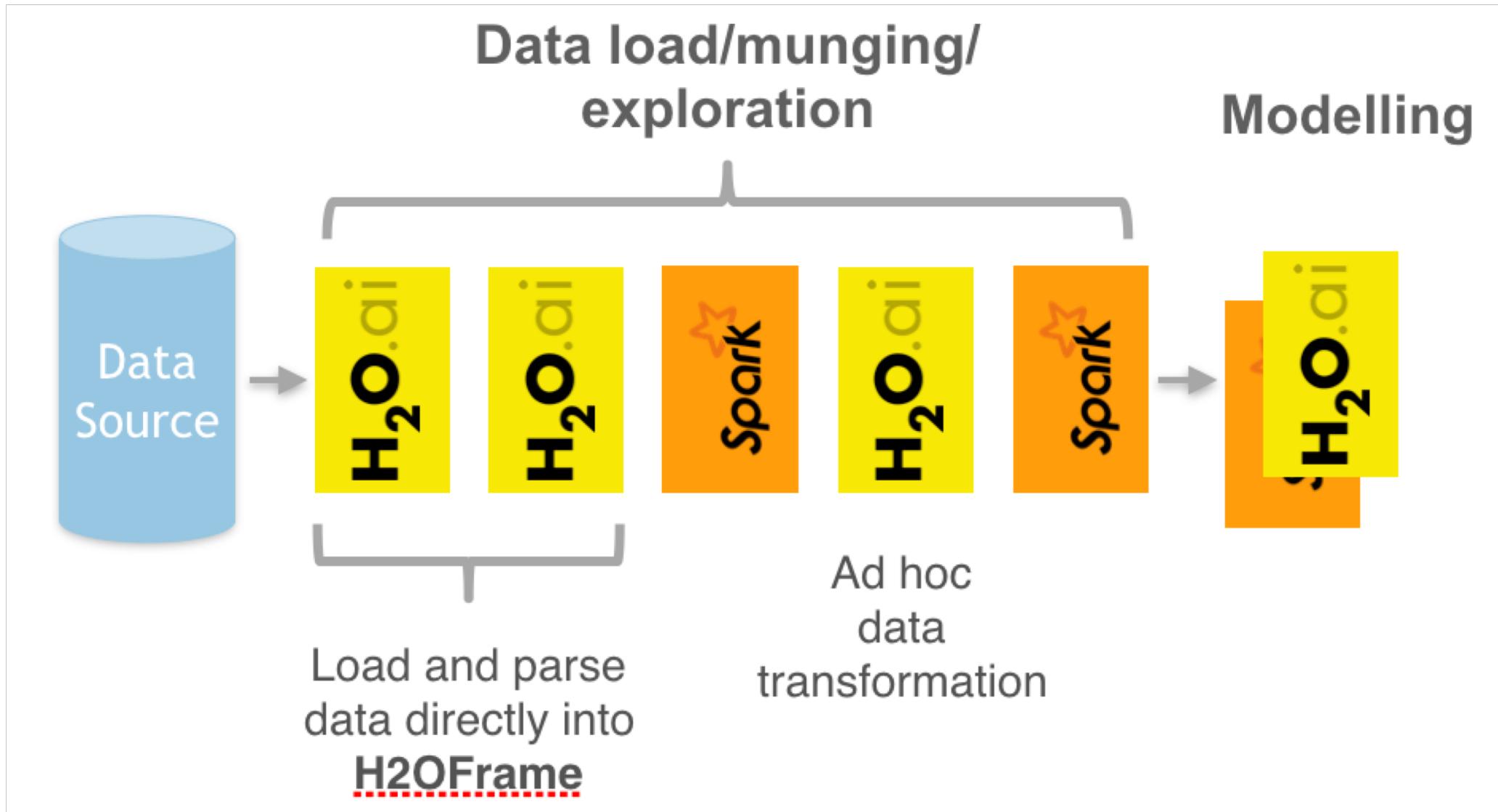
Cluster setup



Use Case



Use Case



Interpretable Machine Learning

- **Intro**
 - Context and Scope.
- Why
 - Why does explainability matter?
- What
 - Steps to build human-centered, low-risk models.
- How
 - Explaining models with rsparkling (H2O-3).

Interpretable Machine Learning

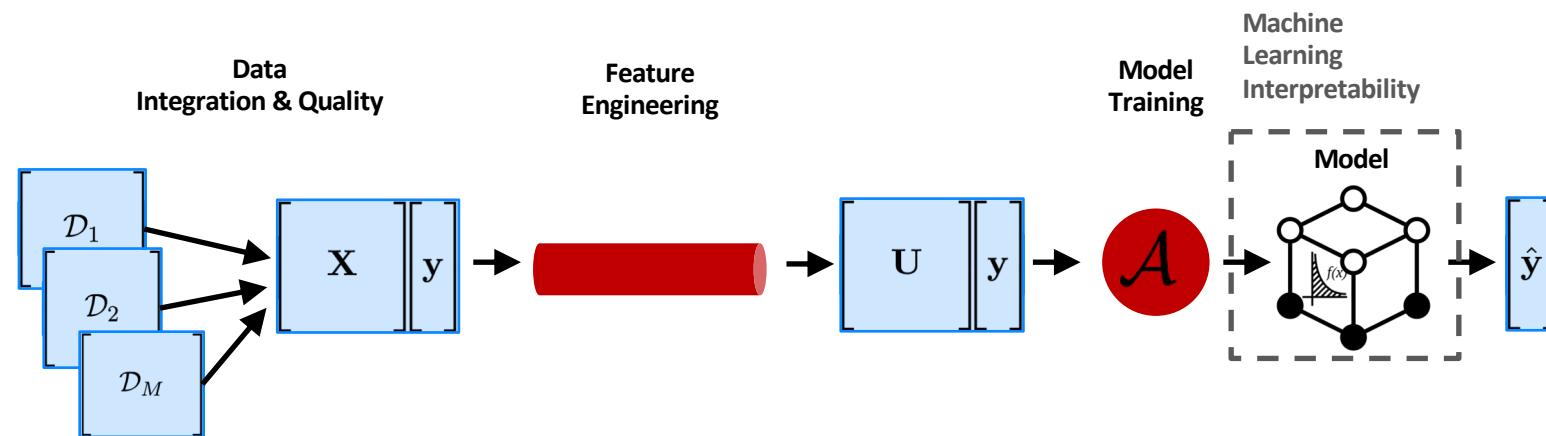
Context and Scope

“[Machine learning interpretability] is the ability to explain or present in understandable terms to a human.” –

<https://arxiv.org/pdf/1702.08608.pdf>

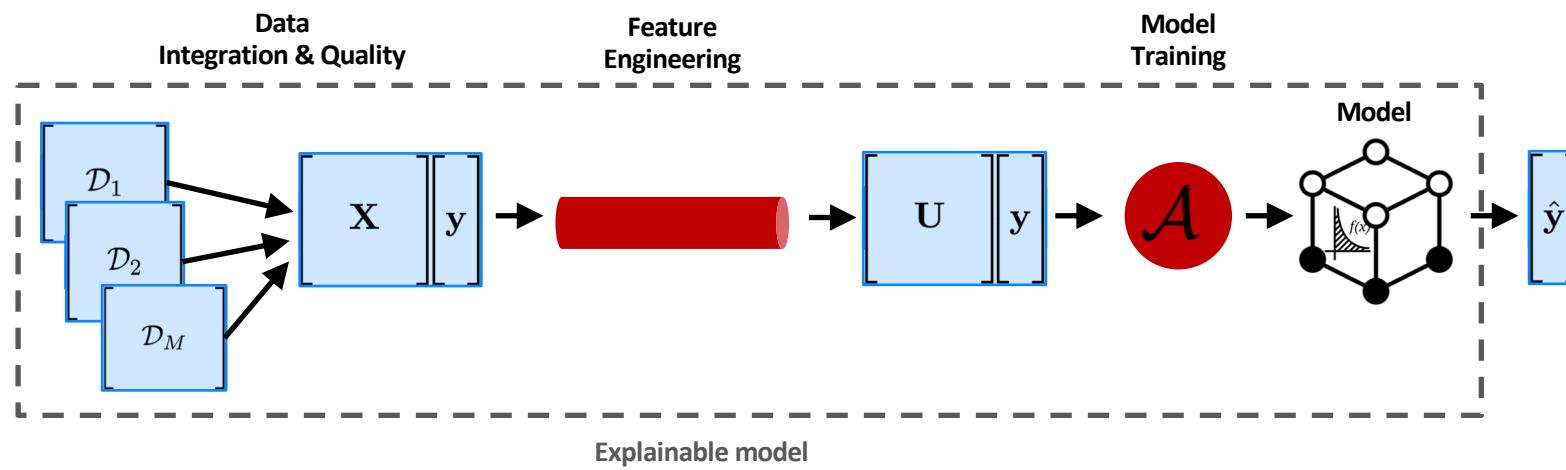
Interpretable Machine Learning

Context and Scope



Interpretable Machine Learning

Context and Scope



Interpretable Machine Learning

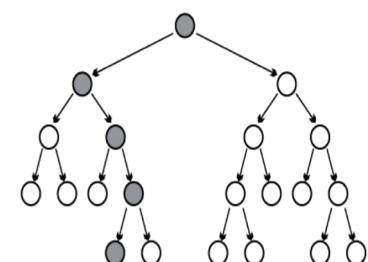
- **Intro**
 - Context and Scope.
- **Why**
 - Why does explainability matter?
- **What**
 - Steps to build human-centered, low-risk models.
- **How**
 - Explaining models with rsparkling (H2O-3).

Interpretable Machine Learning

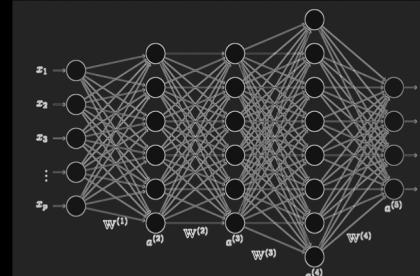
Why does explainability matter?

Potential Performance and Interpretability **Trade-off**

White box model



Black box model



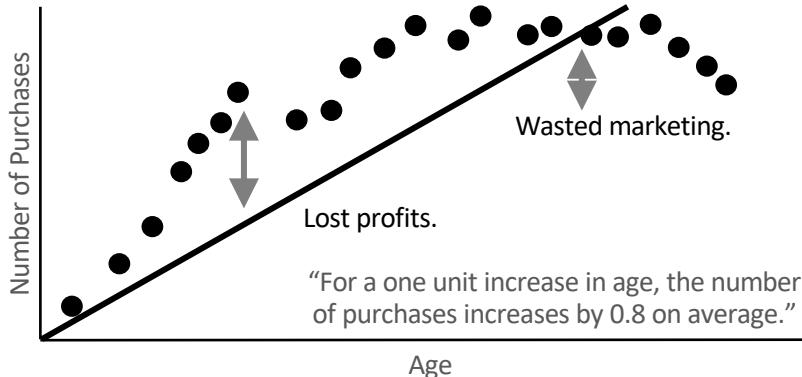
Interpretable Machine Learning

Why does explainability matter?

Potential Performance and Interpretability **Trade-off**

Exact explanations for
approximate models.

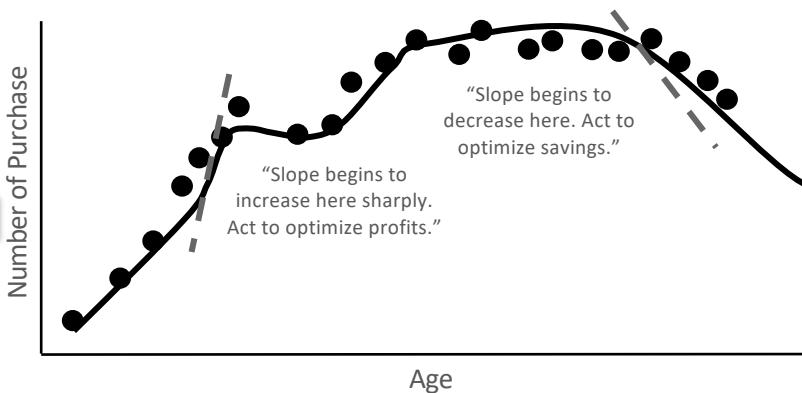
Linear models



Approximate explanations for
exact models.

Sometimes...

Machine learning models

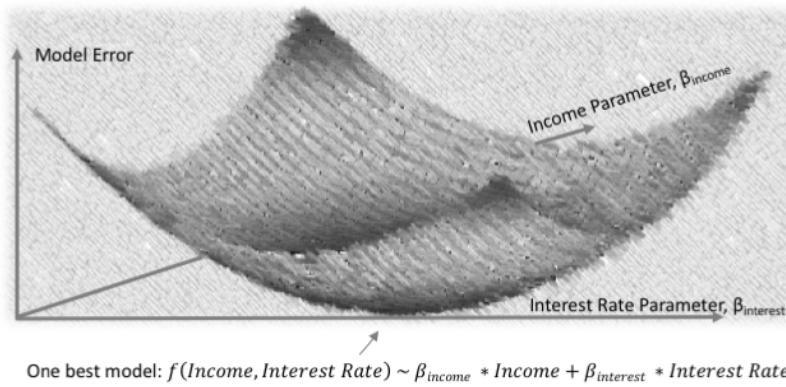


Interpretable Machine Learning

Why does explainability matter?

Multiplicity of Good Models

- For a given well-understood dataset there is usually **one** best linear model, but...

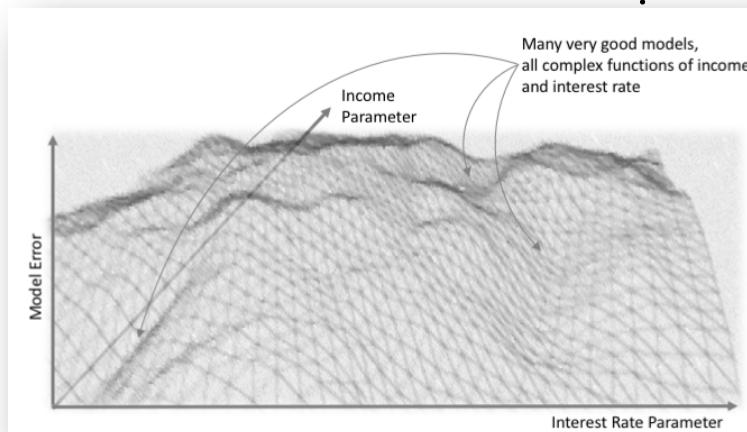


Interpretable Machine Learning

Why does explainability matter?

Multiplicity of Good Models

- ... for a given well-understood dataset there are usually **many good** ML models. Which one to **choose**?
- Same **objective metrics** values, **performance**, ...
- This is often referred to as “the **multiplicity** of good models.” -- [Leo Breiman](#)

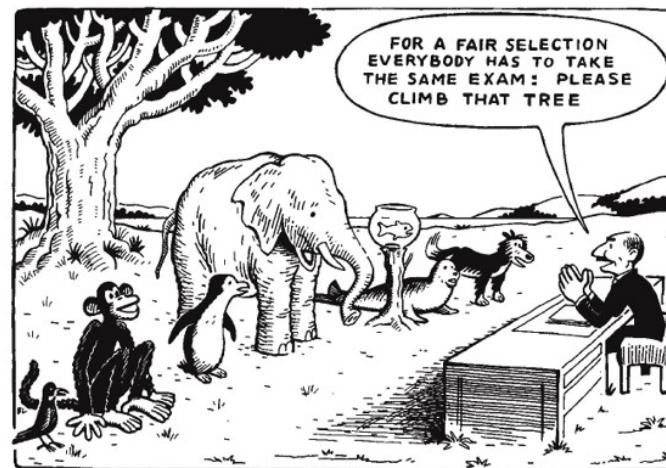


Interpretable Machine Learning

Why does explainability matter?

Fairness and Social Aspects

- Gender
- Age
- Ethnicity
- Health
- Sexual behavior



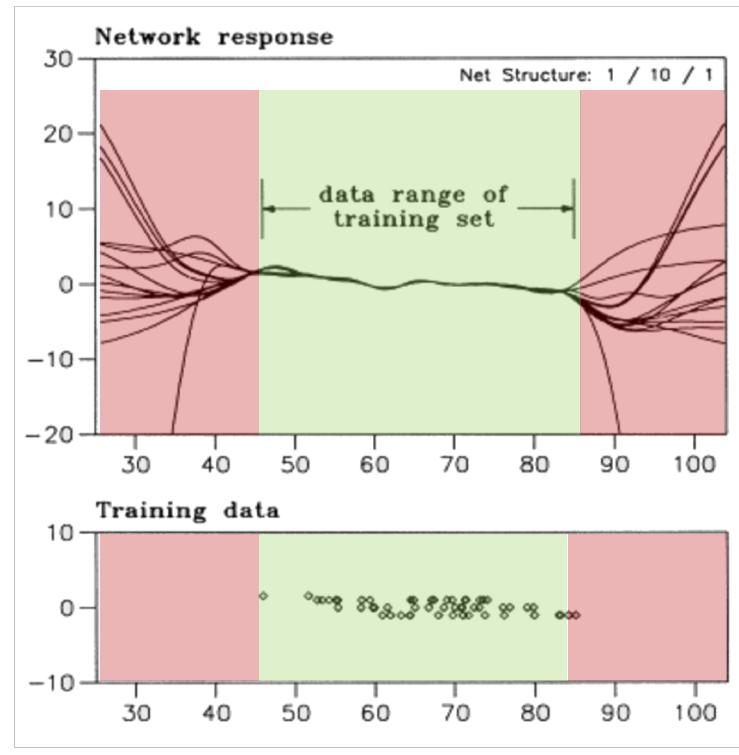
- Avoid **discriminatory models** and remediate [disparate impact](#).

Interpretable Machine Learning

Why does explainability matter?

Trust of model producers & consumers

- Dataset vs. real world
- ML adoption
- Introspection
- Sensitivity
- OOR
- Diagnostics
- “Debugging”



Source: <http://www.vias.org/tmdataanaleng/>

Interpretable Machine Learning

Why does explainability matter?

Security and Hacking

- Goal: **compromise** model integrity
- Attack types:
 - **Exploratory**
 - Surrogate model trained to identify vulnerabilities ~ MLI.
 - Trial and error (for specific class) x indiscriminate attacks.
 - **Causative**
 - Models trained w/ adversary datasets.
 - Local model > adversarial instance > target model.
 - Standard / continuous learning.
 - **Integrity** (compromise system integrity)
 - False negative instance e.g. fraud passes check.
 - **Availability** (compromise system availability)
 - False positive instance e.g. blocks access to legitimate instances.

Interpretable Machine Learning

Why does explainability matter?

Regulated & Controlled Environments

- Legal requirements
 - Banking, insurance, healthcare, ...
- Predictions explanation
 - Decisions justification (reason codes*, ...).
- Fairness
- Security
- Accuracy first vs. **interpretability** first
 - Competitions vs. real world.

Interpretable Machine Learning

So, why does explainability matter?

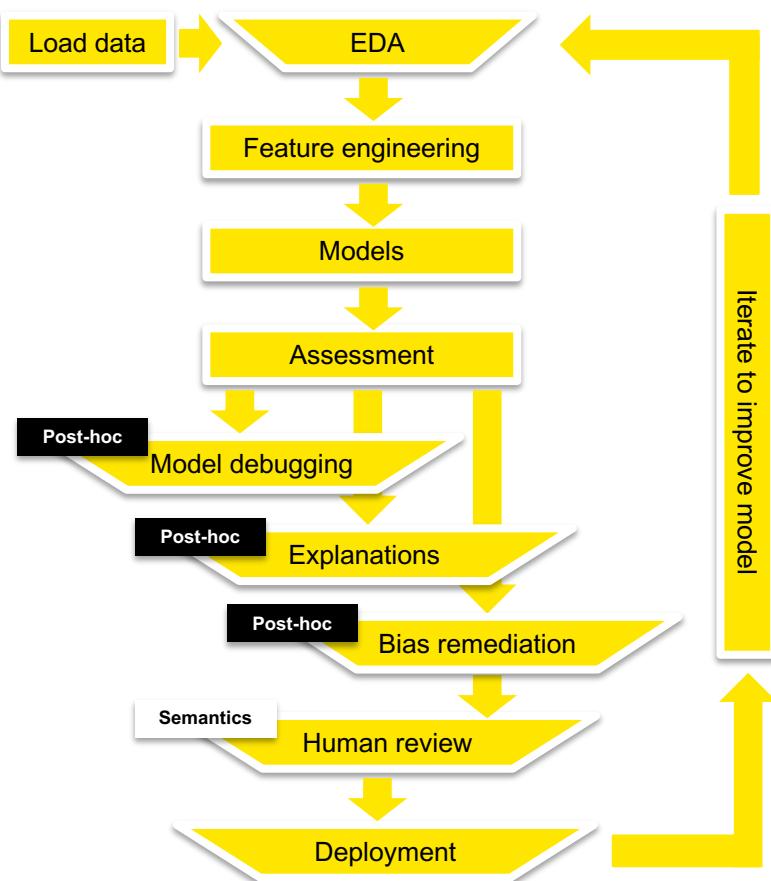
- **Balance** Performance and interpretability.
- **Multiplicity** of good models.
- **Fairness** and **social** aspects.
- **Trust** of model producers and consumers.
- **Security** and **hacking**.
- **Regulated/controlled** environments .

Interpretable Machine Learning

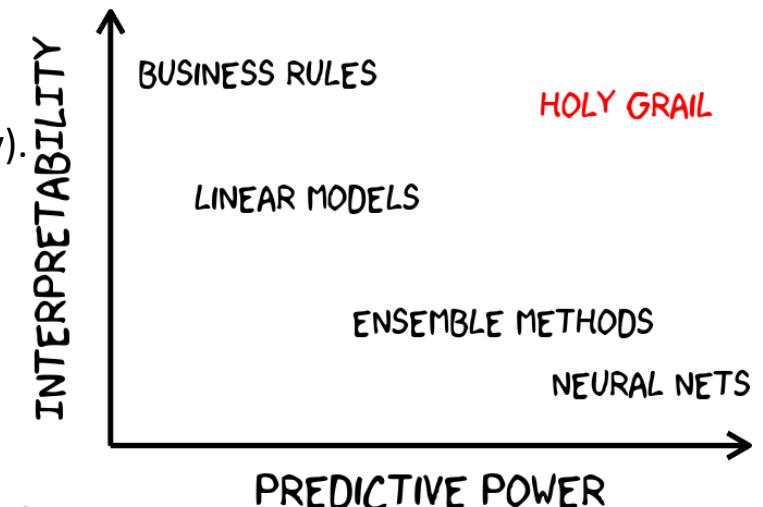
- **Intro**
 - Context and Scope.
- **Why**
 - Why does explainability matter?
- **What**
 - Steps to build human-centered, low-risk models.
- **How**
 - Explaining models with rsparkling (H2O-3).

Interpretable Machine Learning

Steps to build human centered, low-risk models

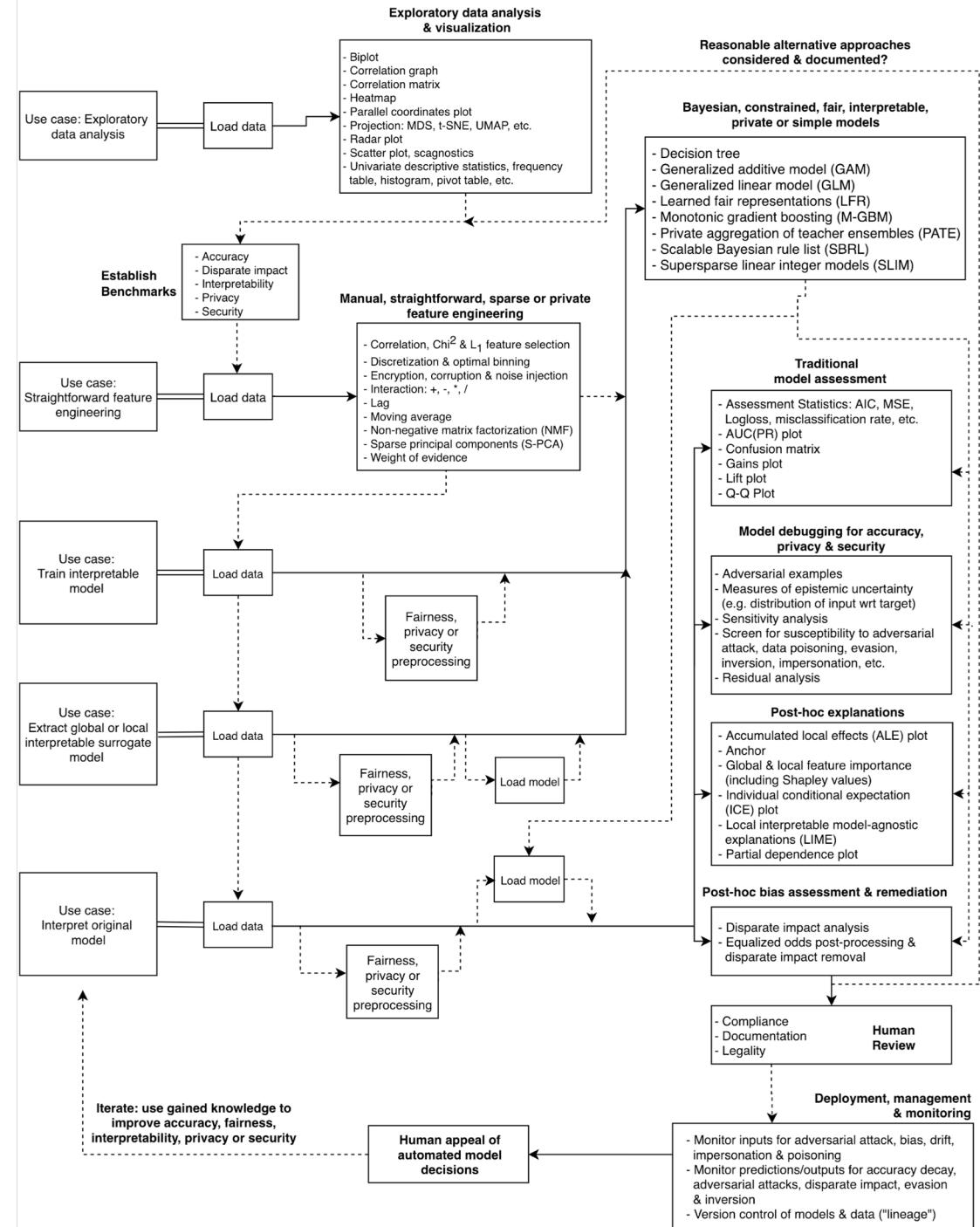


- **Post-hoc model debugging**
 - What-if, sensitivity analysis (accuracy).
- **Post-hoc explanations**
 - Reason codes.
- **Post-hoc bias assessment and remediation**
 - Disparate impact analysis.



Interpretable Machine Learning

Detailed steps to build human centered, low-risk models...

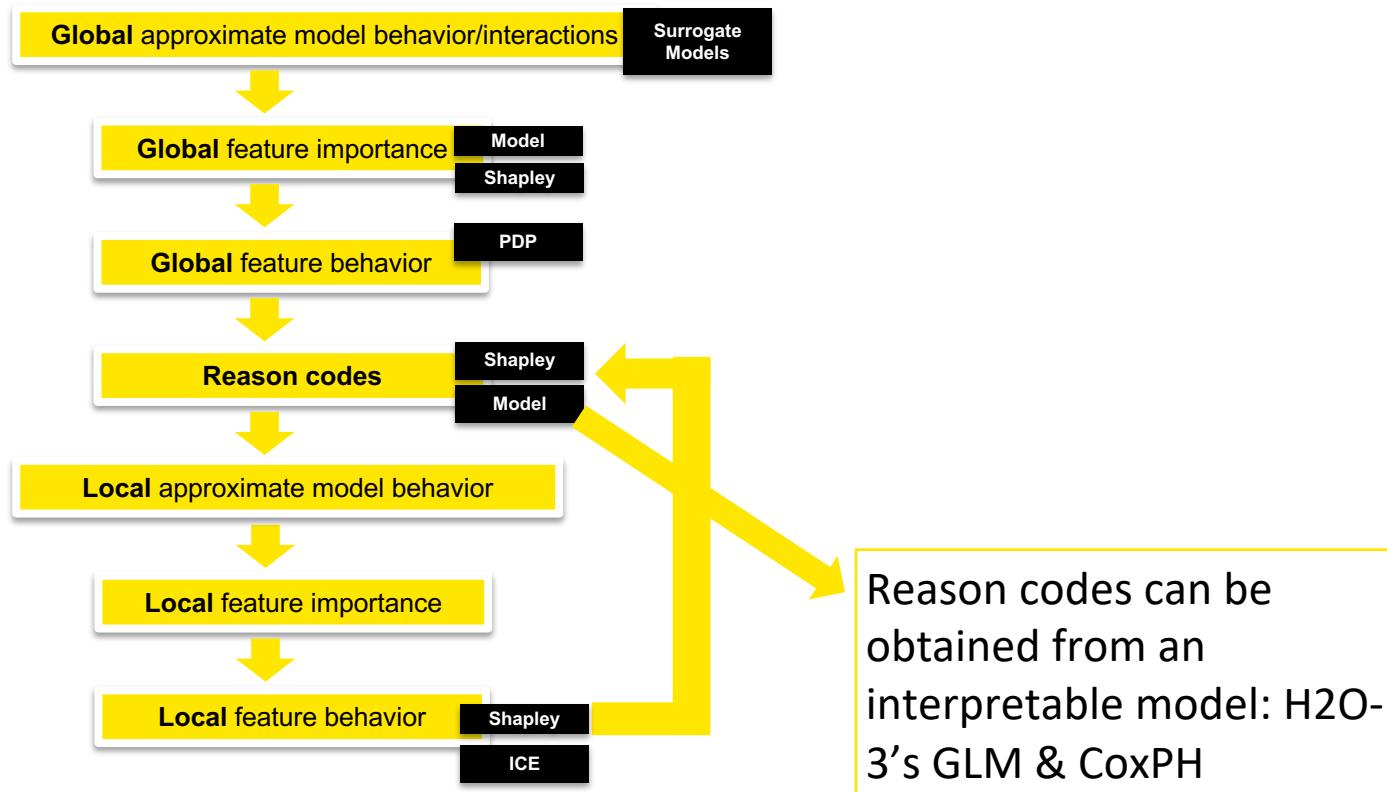


Interpretable Machine Learning

- **Intro**
 - Context and Scope.
- **Why**
 - Why does explainability matter?
- **What**
 - Steps to build human-centered, low-risk models.
- **How**
 - Explaining models with rsparkling (H2O-3).

Interpretable Machine Learning

Explaining models with H2O-3



Interpretable Machine Learning

Explaining models with rsparkling (H2O-3)

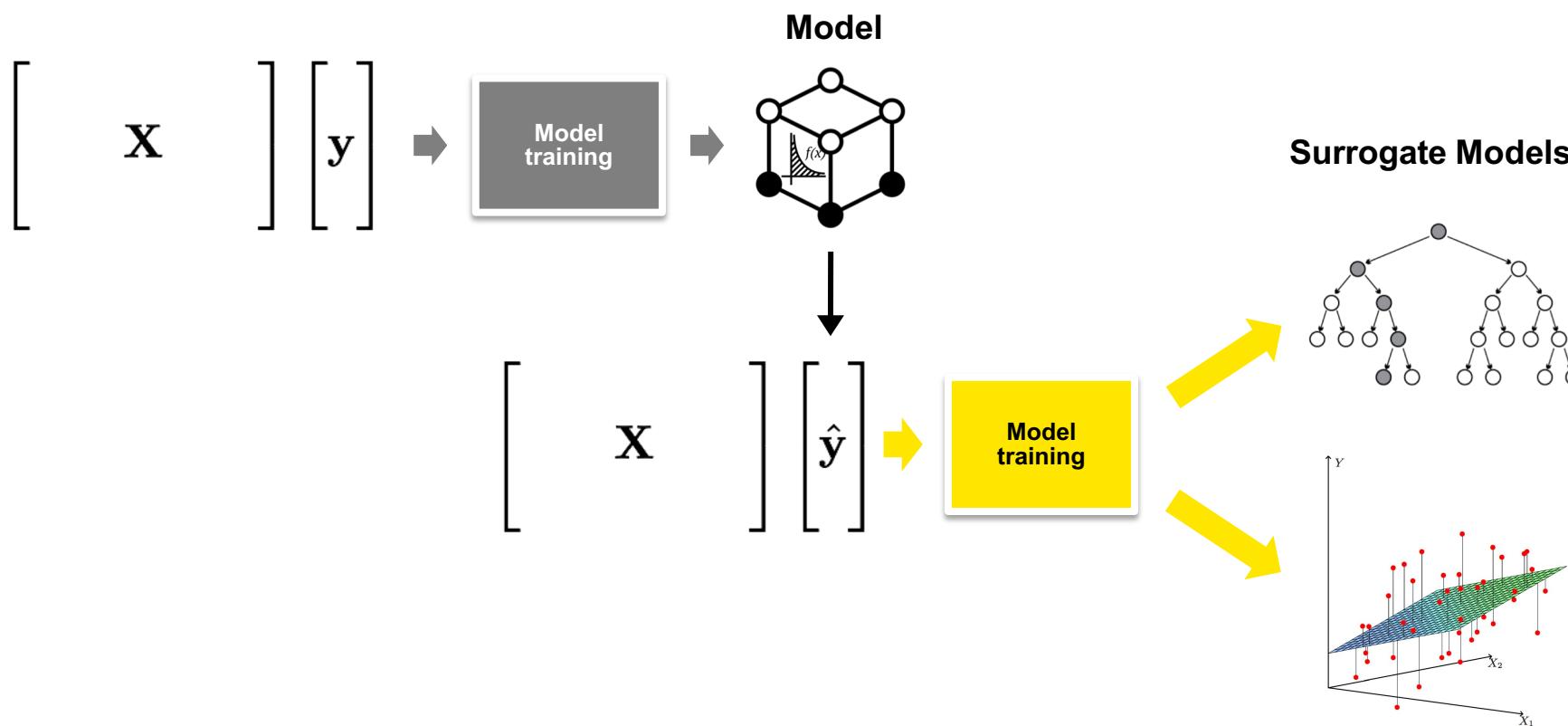
Global Approximate Model Behavior/Interaction

- **Challenge:**
 - Black-box models
 - Original vs. transformed features.
- **Solution:** Surrogate models
 - **Pros**
 - Increases any black-box model's interpretability
 - Time complexity
 - **Cons**
 - Accuracy

Interpretable Machine Learning

Explaining models with rsparkling (H2O-3)

Surrogate Models



Interpretable Machine Learning

Explaining models with rsparkling (H2O-3)

Global Feature Importance: Original Model

- **Challenges:**
 - Black-box models
 - Original vs. transformed features
- **Solutions:**
 - Model Introspection
 - **Pros:**
 - Accuracy
 - **Cons:**
 - Global only

Interpretable Machine Learning

Explaining models with rsparkling (H2O-3)

Global Feature Importance: Shapley Values

- **Challenge**
 - Black-box models
- **Solutions:**
 - Shapley values
 - Pros:
 - Accuracy
 - Math correctness
 - Global and local
 - Cons:
 - Time complexity

Interpretable Machine Learning

Explaining models with rsparkling (H2O-3)

Shapley Values

- [Lloyd Shapley](#)
 - American mathematician who won **Nobel** prize in 2012 (Economics).
 - Shapley values was his Ph.D. thesis written in **50s**.
- **Shapley values:**
 - Supported by **solid** mathematical (game) theory.
 - Calculation has **exponential** time complexity (number of coalitions).
 - Typically **unrealistic to compute** in real world.
 - Can be computed in **global** or **local** scope.
 - **Guarantee fair distribution** among features in the instance.
 - Does **not** work well in **sparse** cases, **all** features must be used.
 - Return **single value per feature**, not a model.



ALGORITHM: Shapley value or contribution of feature f in example e

Method:

- i) have dataset and chose example e and feature f
- ii) compute marginal contribution of feature f in e for every feature coalition

for \forall coalition c :

- a) eliminate all features which are not in current coalition c & using value from other randomly selected example e'
- b) predict...
 - with feature f in coalition $\Rightarrow p^w$
 - without feature f in coalition $\Rightarrow p^{w/o}$
 - (random select other example e' w/o f and take value of f from there)
- c) marginal f contribution in e : $p^w - p^{w/o} = \pi_c$

iii) marginal feature contribution is $\text{SHAPLEY}(f)^e = \overline{\text{AVG}}(\pi_c)_{c=1}^{2^F}$ \sim number of coalitions \sim response $O(2^F)$

SHAPLEY VALUES
GATE \sim single instance i
CAIN \sim individual - avg prediction
PLAYERS \sim feature is player
players cooperate in coalition to receive gains
Global
With all local
With only one active
Shapley values

Interpretable Machine Learning

Explaining models with rsparkling (H2O-3)

Global Feature Behavior:

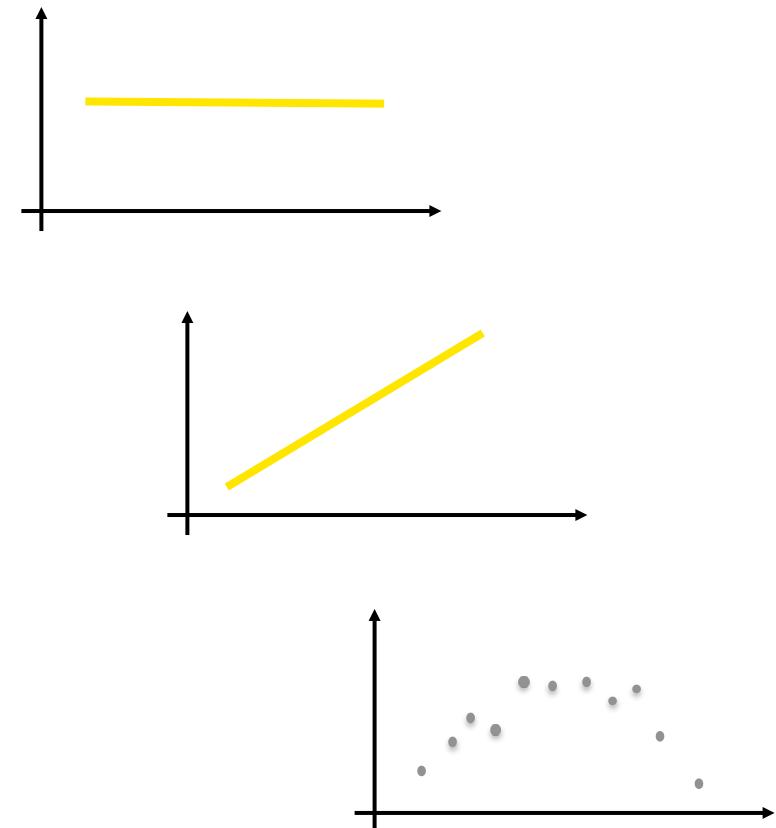
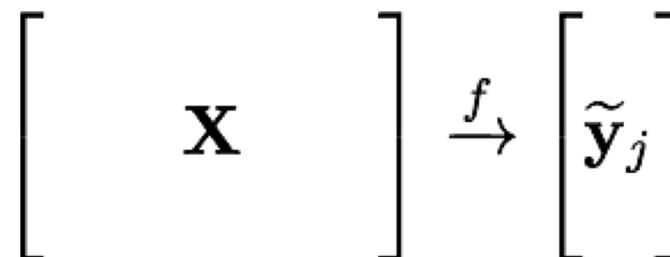
- **Solution:** PDP (Partial Dependency Plots)

- **Pros**

- Time complexity
 - Original features
 - White/black model interpretability

- **Cons**

- Accuracy



Interpretable Machine Learning

Explaining models with rsparkling (H2O-3)

Reason codes: Local Feature Importance

- **Use Cases:**
 - Predictions explanations
 - Legal
 - Debugging
 - Drill-down
- From **global** to local scope
 - Shapley

Interpretable Machine Learning

Explaining models with rsparkling (H2O-3)

Local Feature Behavior:

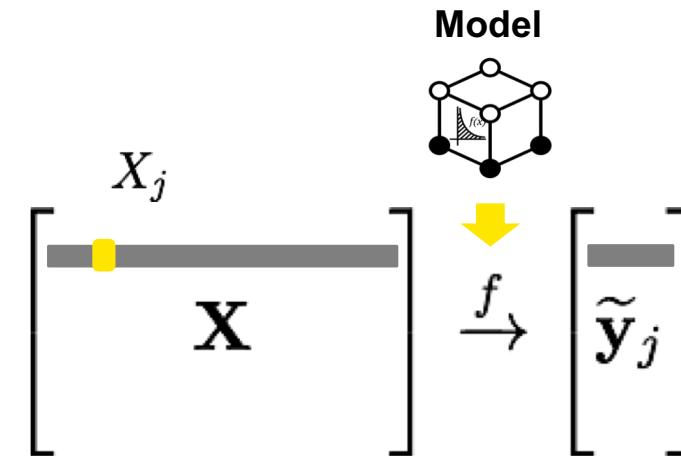
- **Solution:** ICE (Individual Conditional Expectation)

- **Pros**

- Time complexity
 - White/black model interpretability

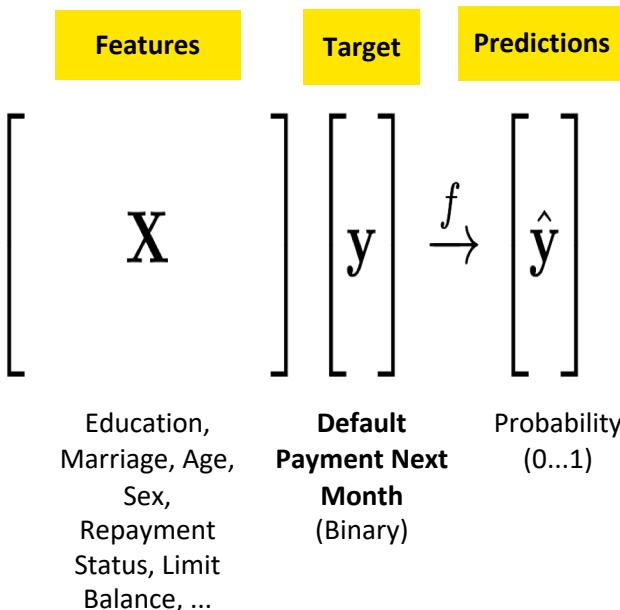
- **Cons**

- Accuracy



Demo

Dataset: Credit Card



Column Name	Description
ID	ID of each client
LIMIT_BAL	Amount of given credit in NT dollars (includes individual and family/supplementary credit)
SEX	Gender (1=male, 2=female)
EDUCATION	(1=graduate school, 2=university, 3=high school, 4=others, 5=unknown, 6=unknown)
MARRIAGE	Marital status (1=married, 2=single, 3=others)
AGE	Age in years
PAY_x {1, ..., 6}	Repayment status in August, 2005 – April, 2005 (-1=paid duly, 1=payment delay for 1 month, ..., 8=payment delay for 8 months)
BILL_AMTx {1, ..., 6}	Amount of bill statement in September, 2005 – April, 2005 (NT dollar)
PAY_AMTx {1, ..., 6}	Amount of previous payment in September, 2005 – April, 2005 (NT dollar)
default_payment_next_month	Default payment (1=yes, 0=no)

Demo: https://github.com/navdeep-G/sdss-2019/blob/master/r/rsparkling_mli.R

Interpretable Machine Learning Resources

- Booklets/Books:
 - [Ideas on Interpreting Machine Learning](#) by Patrick Hall, Wen Phan, & SriSatish Ambati
 - [An Introduction to Machine Learning Interpretability](#) by Patrick Hall & Navdeep Gill
 - [Interpretable Machine Learning](#) by Christoph Molnar
 - Of course, there are many more ...
- Presentations:
 - [Human Friendly Machine Learning](#) by Patrick Hall
 - [Ideas on Machine Learning Interpretability](#) by Navdeep Gill
 - Of course, there are many more ...
- GitHub repositories:
 - [Awesome Machine Learning Interpretability](#) (Contains many resources)
 - [MLI Resources](#)
 - Of course, there are many more ...

Thank You!

@Navdeep_Gill_ on Twitter

navdeep-G on Github

navdeep.gill@h2o.ai

H₂O