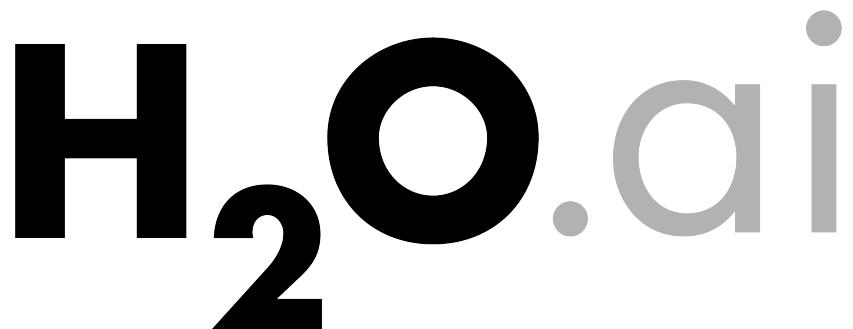


Introduction to Machine Learning with H₂O

- Introduction (30 mins)
 - From Engineering to Data Science – Why?
 - Machine Learning Basics
 - H₂O Machine Learning Platform
- Tutorial (60 mins)
 - Regression (House Price)
 - Classification (Human Activities)
 - Clustering (Water Treatment Plant)
- Extra: More About H₂O (15 mins)
 - H₂O on a Multi-Node Cluster
 - Next-Gen H₂O
- Q & A (15 mins)



Jo-fai (Joe) Chow
Data Scientist at H₂O.ai
joe@h2o.ai

Version 1.1

Download: http://bit.ly/h2o_exeter_2017

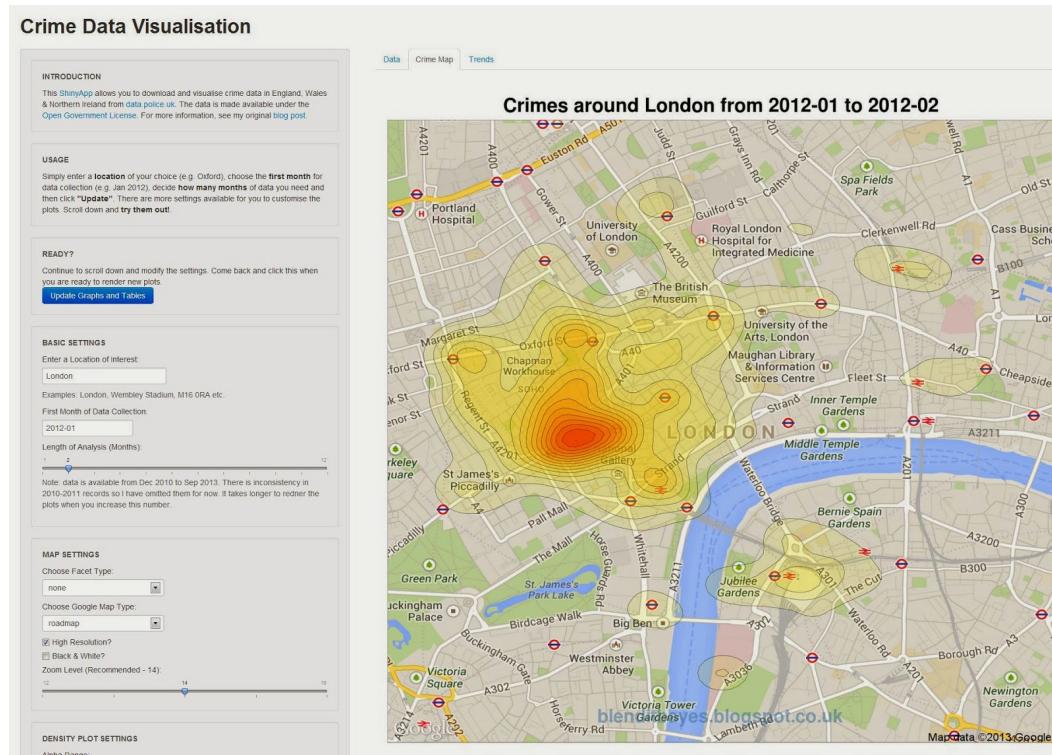
From Engineering to Data Science

- My Story

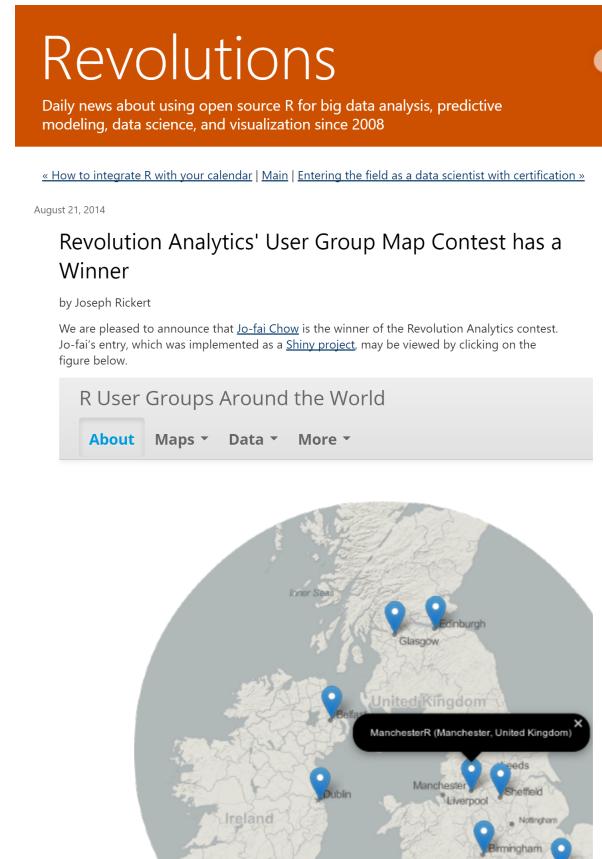
About Me

- Civil (Water) Engineer
 - 2005 – 2010
 - Consultant (SEAMS, UK)
 - Utilities / Asset Management
 - Constrained Optimization
 - 2010 – 2014
 - STREAM EngD (Exeter)
 - Infrastructure Design Optimisation
 - Machine Learning + Water Engineering
 - **Discovered H₂O in 2014**
- Data Scientist
 - 2015 – 2016
 - Virgin Media (UK)
 - Domino Data Lab (Silicon Valley)
 - 2016 – Present
 - H₂O.ai (Silicon Valley)
 - How?
 - bit.ly/joe_kaggle_story

About Me – I ❤️ DataViz



My First Data Viz & Shiny App Experience
[CrimeMap \(2013\)](#)



Revolution Analytics' Data Viz Contest
[RUGSMAPS \(2014\)](#)



Jo-fai (Joe) Chow
@matlabulous

Thank you very much @RevolutionR
@revodavid @RevoJoe #iloveR
bit.ly/rugsmaps #Shiny #rMaps

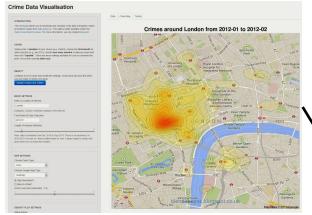


RETWEETS
3

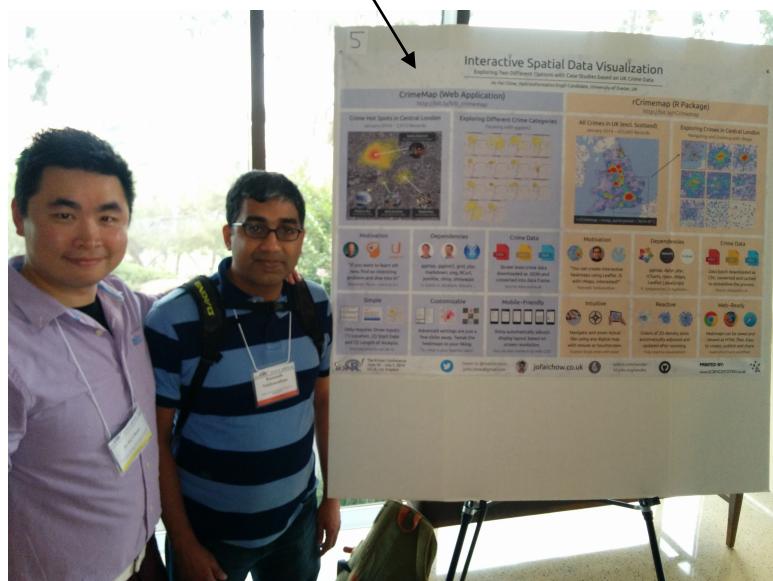


1:25 AM - 29 Aug 2014

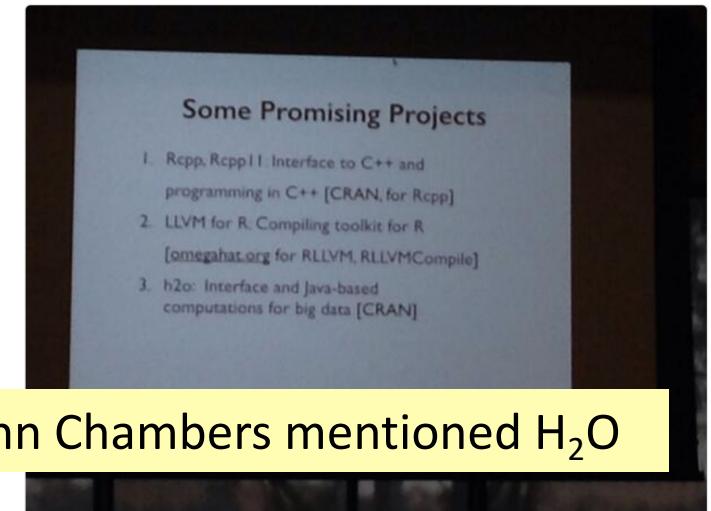
useR! 2014



CrimeMap -> poster



#SparkSummit ~~~> #useR2014
#h2o



John Chambers mentioned H₂O



Things to try after useR! – Part 1: Deep Learning wit...
Annual R User Conference 2014The useR! 2014 conference was a mind-blowing experience. Hundreds of R enthusiasts and the beautiful UCLA campus, I am rea...
r-bloggers.com

Jo-fai (Joe) Chow
@matlalobus

Replying to @h2oai

Hi @srisatish @ArnoCandel and every1
@hexadata thx 4 making and open-sourcing
the powerful #H2O shd hv tried it during (not
after) #user2014

LIKES
2

1:41 PM - 28 Jul 2014

H₂O.ai

About Me – I ❤️ Kaggle

Domino Data Lab
At the intersection of data science and engineering.
Domino App Site | [Twitter](#) | [Email](#)

19 Sep 2014 • [Facebook Like](#) 0 | [Twitter Tweet](#) 21 | [Google+1](#) 4

How to use R, H2O, and Domino for a Kaggle competition

Guest post by Jo-Fai Chow

The sample project (code and data) described below is [available on Domino](#).

If you're in a hurry, feel free to skip to:

- [Tutorial 1: Using Domino](#)
- [Tutorial 2: Using H2O to Predict Soil Properties](#)
- [Tutorial 3: Scaling up your analysis](#)

Introduction

This blog post is the sequel to [TTTAR1](#) a.k.a. [An Introduction to H2O Deep Learning](#). If the previous blog post was a brief intro, this post is a proper machine learning case study based on a recent [Kaggle competition](#): I am leveraging [R](#), [H2O](#) and [Domino](#) to compete (and do pretty well) in a real-world data mining contest.

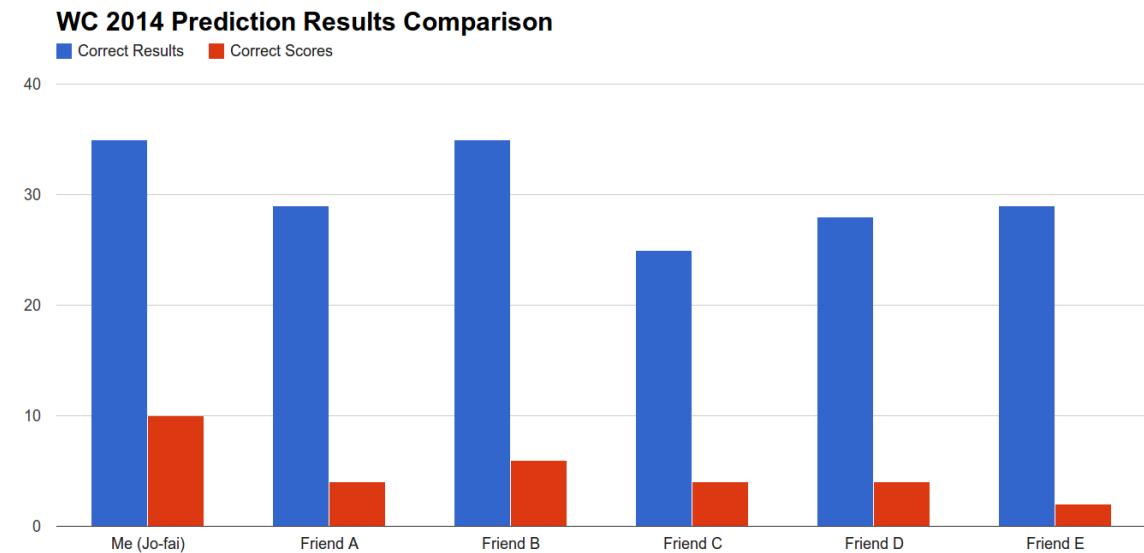
R + H₂O + Domino for Kaggle
[Guest Blog Post for Domino & H₂O \(2014\)](#)

- The Long Story
 - bit.ly/joe_kaggle_story

World Cup 2014

- Machine Learning vs My Friends

- Team performance data from web
- Simple machine learning models
- Objective: Predict Correct Score
- Me : 10 out of 64 (15.6%)
- Friends' Avg : 4 out of 64 (6.3%)



From Engineering to Data Science

- Knowledge Driven Decision Making
 - Write code to deal with data based on knowledge and assumptions.
 - Strong domain knowledge.
- Data Driven Decision Making
 - Define a problem and collect data related to the problem.
 - Write code to show data to computers and let computers (algorithms) find patterns in data.
 - Domain knowledge is helpful but not necessary.





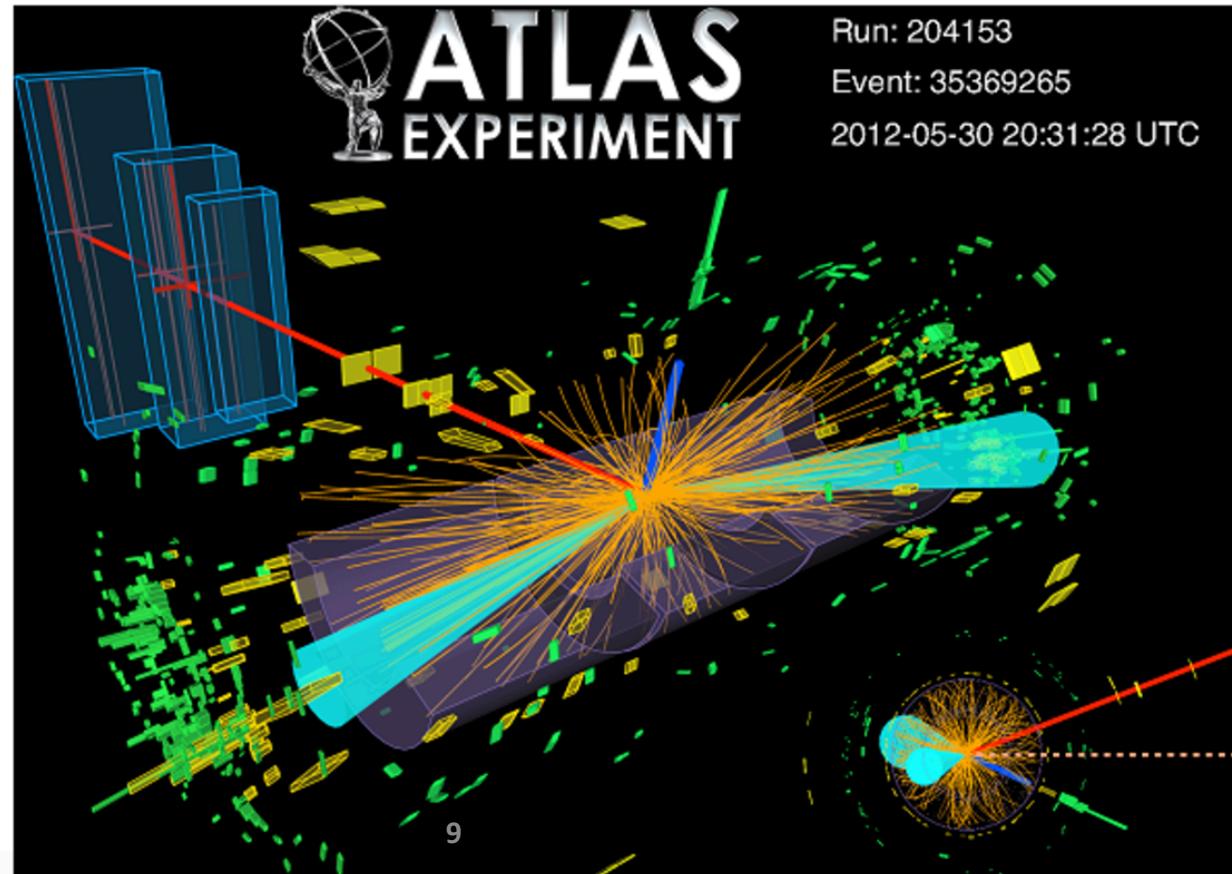
Higgs Boson Machine Learning Challenge

Use the ATLAS experiment to identify the Higgs boson

\$13,000 · 1,785 teams · 3 years ago

[Overview](#)[Data](#)[Discussion](#)[Leaderboard](#)[Rules](#)[Team](#)[My Submissions](#)[Late Submission](#)[Overview](#)

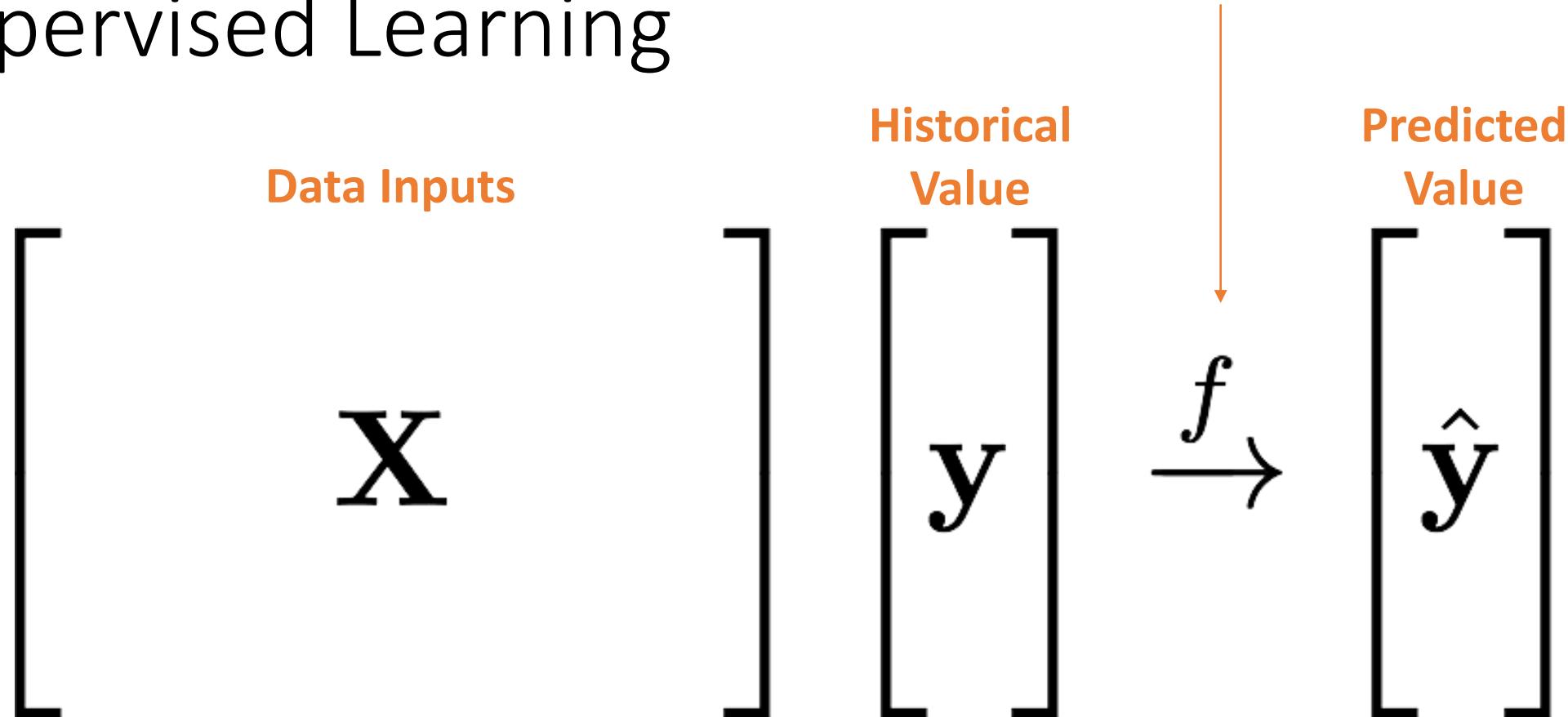
<https://www.kaggle.com/c/higgs-boson>

[Description](#)[Evaluation](#)[Prizes](#)[About The Sponsors](#)[Timeline](#)[Winners](#)

What is Machine Learning?

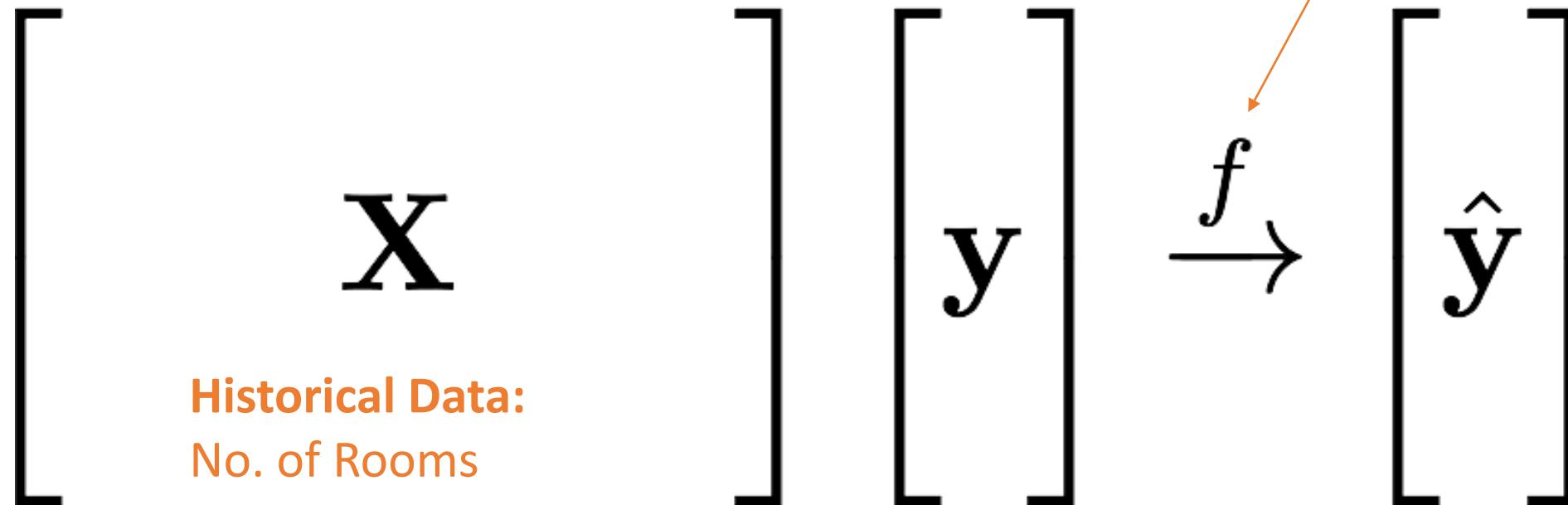
- Supervised Learning
- Unsupervised Learning

Supervised Learning



Supervised Learning Example

Machine Learning:
Learn Patterns
from Data



Target:
House Value

Predicted Value
(for evaluation)

Supervised Learning Example

[

X

New Data:

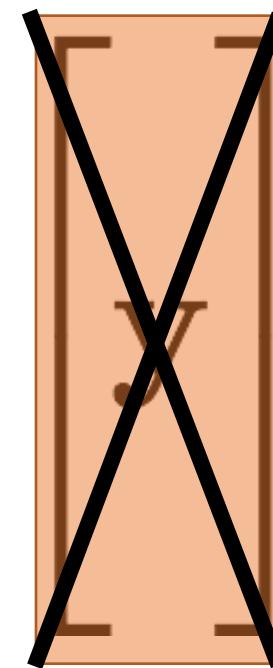
No. of Rooms

Crime Rate

Pupil-Teacher Ratio

...

]



Target:
Unknown

Patterns Learned
from Historical Data

$$f \rightarrow$$

[

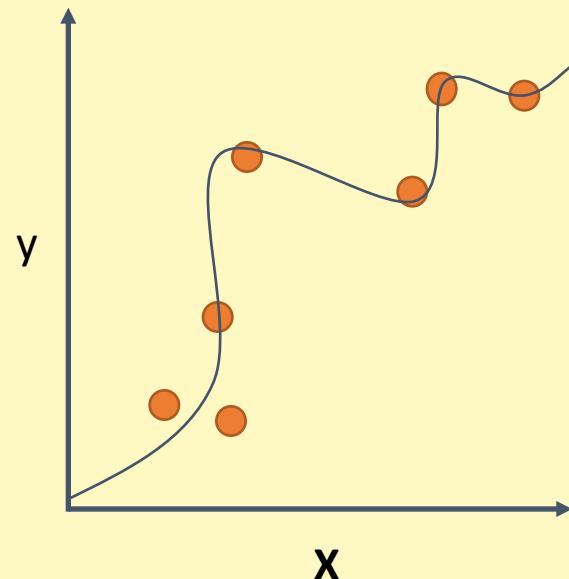
]

Predicted Value
(for decision making)

Supervised Learning – You Already Have Target Data

Regression:

How much will a customer spend?

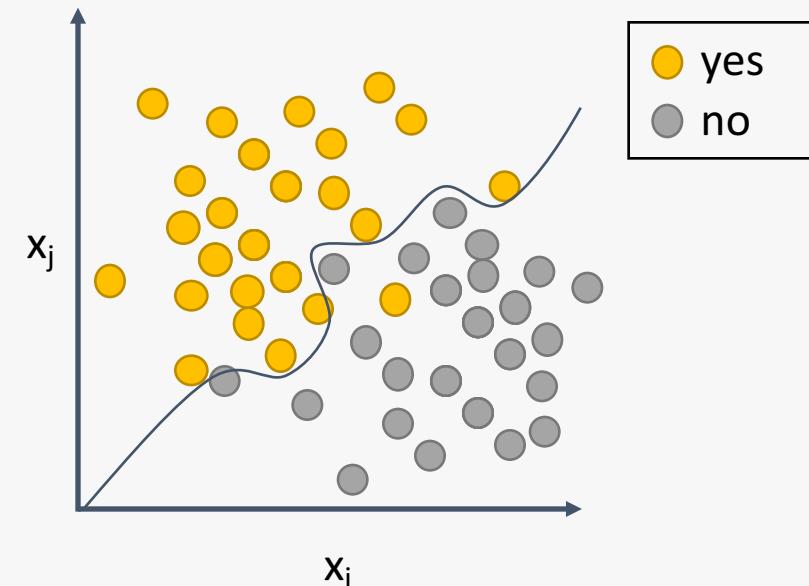


H₂O algos:

Penalized Linear Models
Random Forest
Gradient Boosting
Neural Networks
Stacked Ensembles

Classification:

Will a customer make a purchase? Yes or No

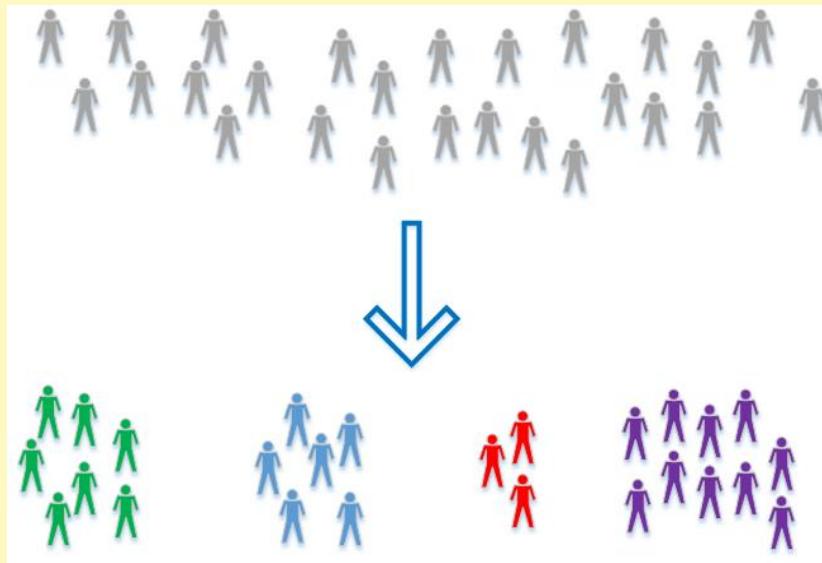


H₂O algos:

Penalized Linear Models
Naïve Bayes
Random Forest
Gradient Boosting
Neural Networks
Stacked Ensembles

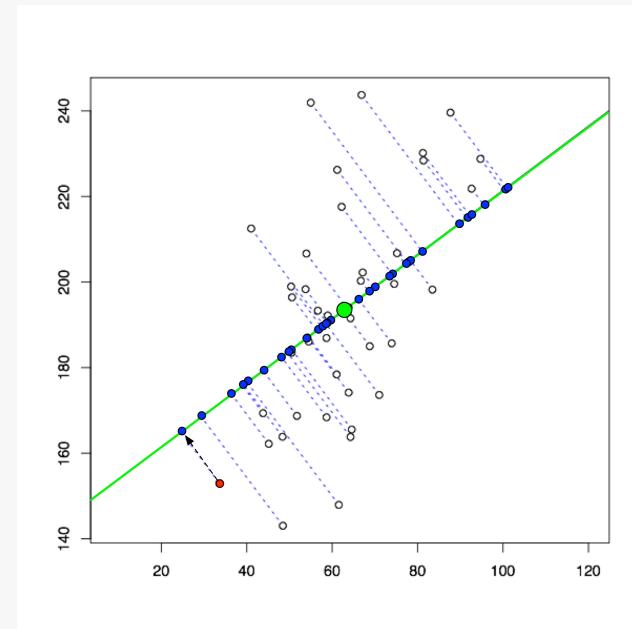
Unsupervised Learning – Discover Hidden Patterns

Clustering:
Customer Segmentation



H₂O algos:
K-Means
Generalised Low Rank Model

Dimensionality Reduction:
Linear Transformation of Variables



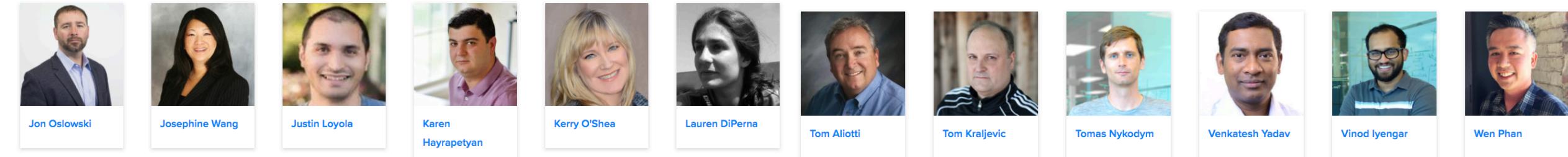
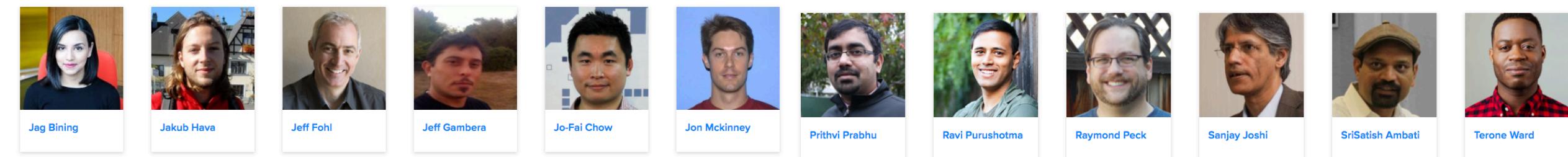
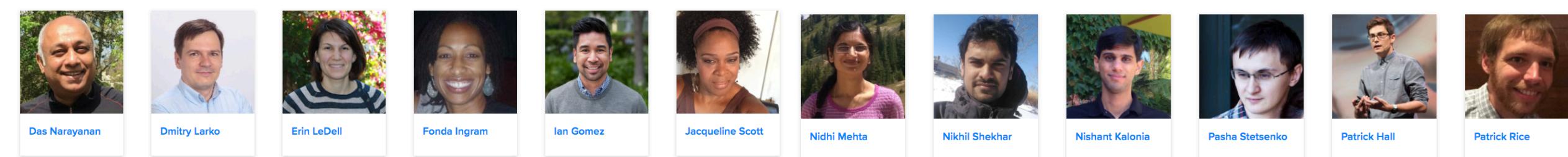
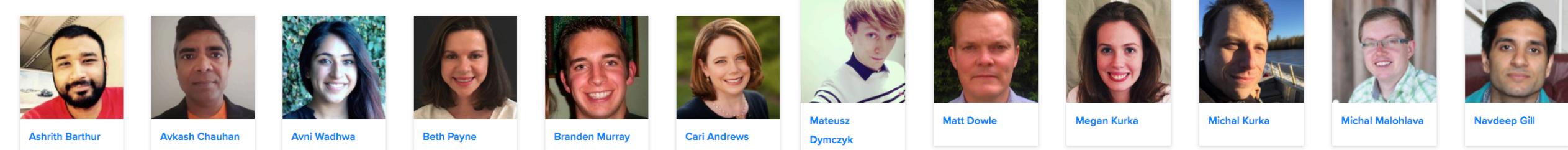
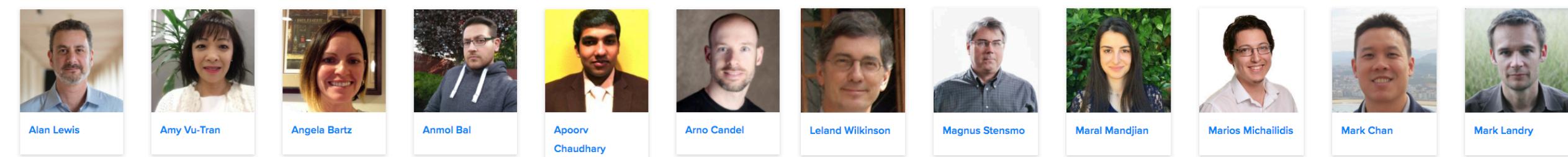
H₂O algos:
Principal Component Analysis
Generalised Low Rank Model

About H₂O.ai

Company Overview

Founded	2011 Venture-backed, debuted in 2012
Products	<ul style="list-style-type: none">• H₂O Open Source In-Memory AI Prediction Engine• Sparkling Water (H₂O + Spark)• Deep Water (H₂O + Other Deep Learning Frameworks)• Driverless AI (Next-Gen H₂O)
Mission	Operationalize Data Science, and provide a platform for users to build beautiful data products
Team	<p>75+ employees</p> <ul style="list-style-type: none">• Distributed Systems Engineers doing Machine Learning• World-class visualization designers
Headquarters	Mountain View, CA





Scientific Advisory Council



Dr. Trevor Hastie

- John A. Overdeck Professor of Mathematics, Stanford University
- PhD in Statistics, Stanford University
- Co-author, *The Elements of Statistical Learning: Prediction, Inference and Data Mining*
- Co-author with John Chambers, *Statistical Models in S*
- Co-author, *Generalized Additive Models*



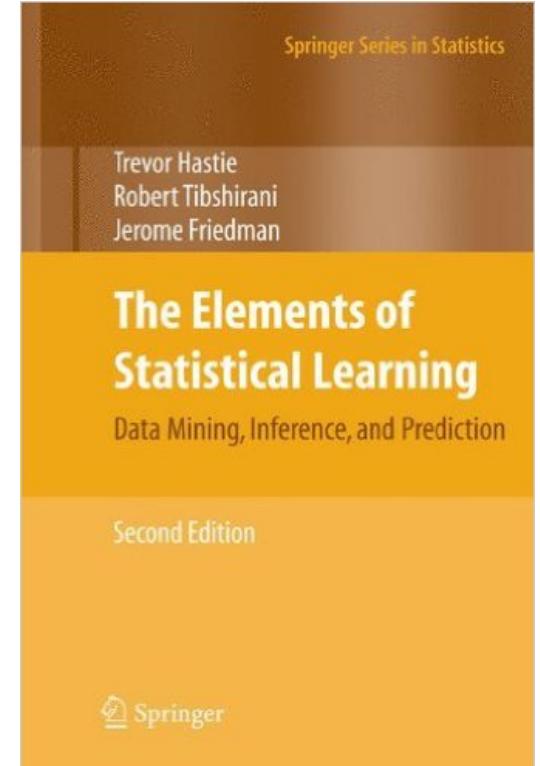
Dr. Robert Tibshirani

- Professor of Statistics and Health Research and Policy, Stanford University
- PhD in Statistics, Stanford University
- Co-author, *The Elements of Statistical Learning: Prediction, Inference and Data Mining*
- Author, *Regression Shrinkage and Selection via the Lasso*
- Co-author, *An Introduction to the Bootstrap*

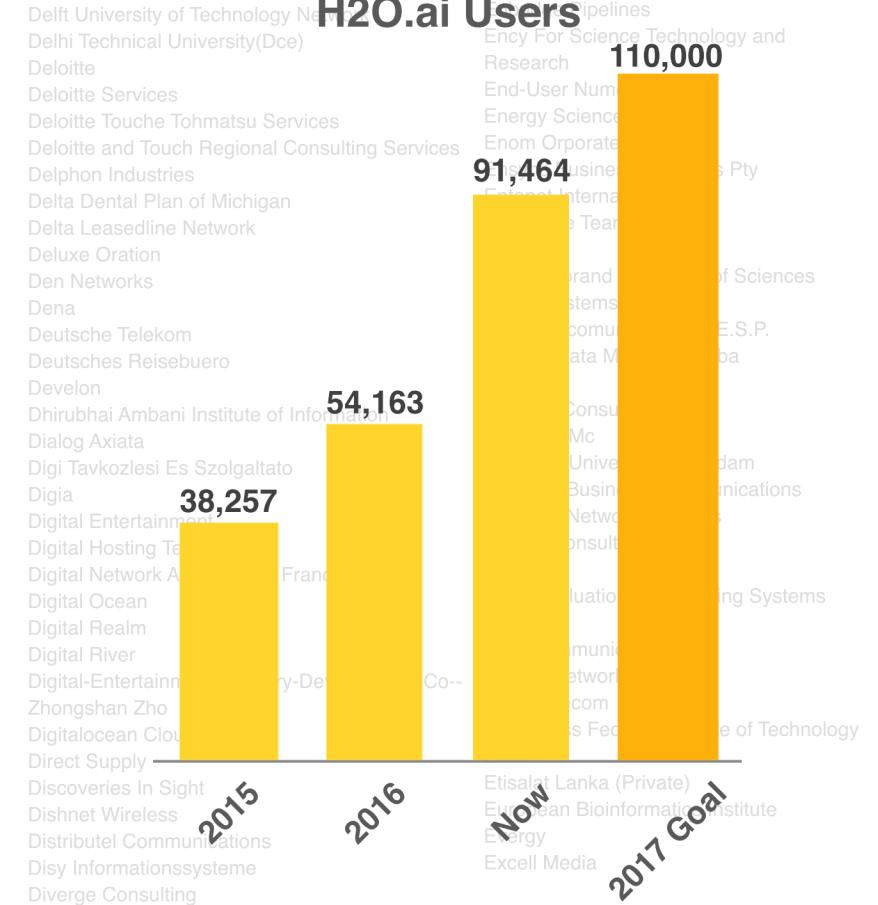
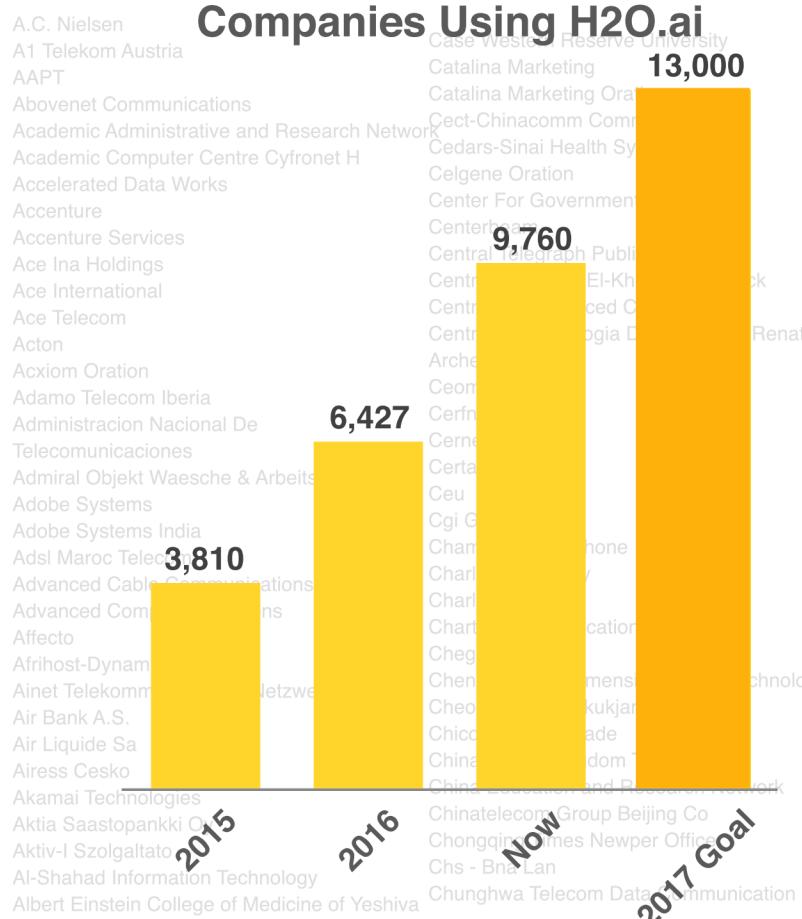


Dr. Steven Boyd

- Professor of Electrical Engineering and Computer Science, Stanford University
- PhD in Electrical Engineering and Computer Science, UC Berkeley
- Co-author, *Distributed Optimization and Statistical Learning via the Alternating Direction Method of Multipliers*
- Co-author, *Linear Matrix Inequalities in System and Control Theory*
- Co-author, *Convex Optimization*



H2O Community & Fortune 100 customers



Select Reference Customers:

“Overall customer satisfaction is very high.” - Gartner



Harnessing the power of AI to transform the detection of fraud and error

Setting the scene

PwC has invested significantly in pioneering the use of artificial intelligence for the audit and has partnered with H2O.ai, a leading Silicon Valley-based AI company.

Following 18 months of development, the first outcome of this partnership is PwC's GL.ai, the first module of PwC's Audit.ai - a revolutionary bot that does what humans can't. Its AI analyses billions of different data points in seconds and applies judgement to detect anomalies in general ledger transactions.



"The reason this is such a brilliant tool is the ability to look at different risks in context at the same time. For example, it would be uneconomical for an auditor to look at every single user's pattern of activity and decide what was unusual. With GL.ai, the algorithms do it for us."

Laura Needham partner, PwC UK



Follow

Exciting night at this year's [@WAI_News](#) Awards: PwC wins 2017 Audit Innovation of the Year! pwc.to/Glai17 #taandab17



10:15 PM - 4 Oct 2017

Community Expansion

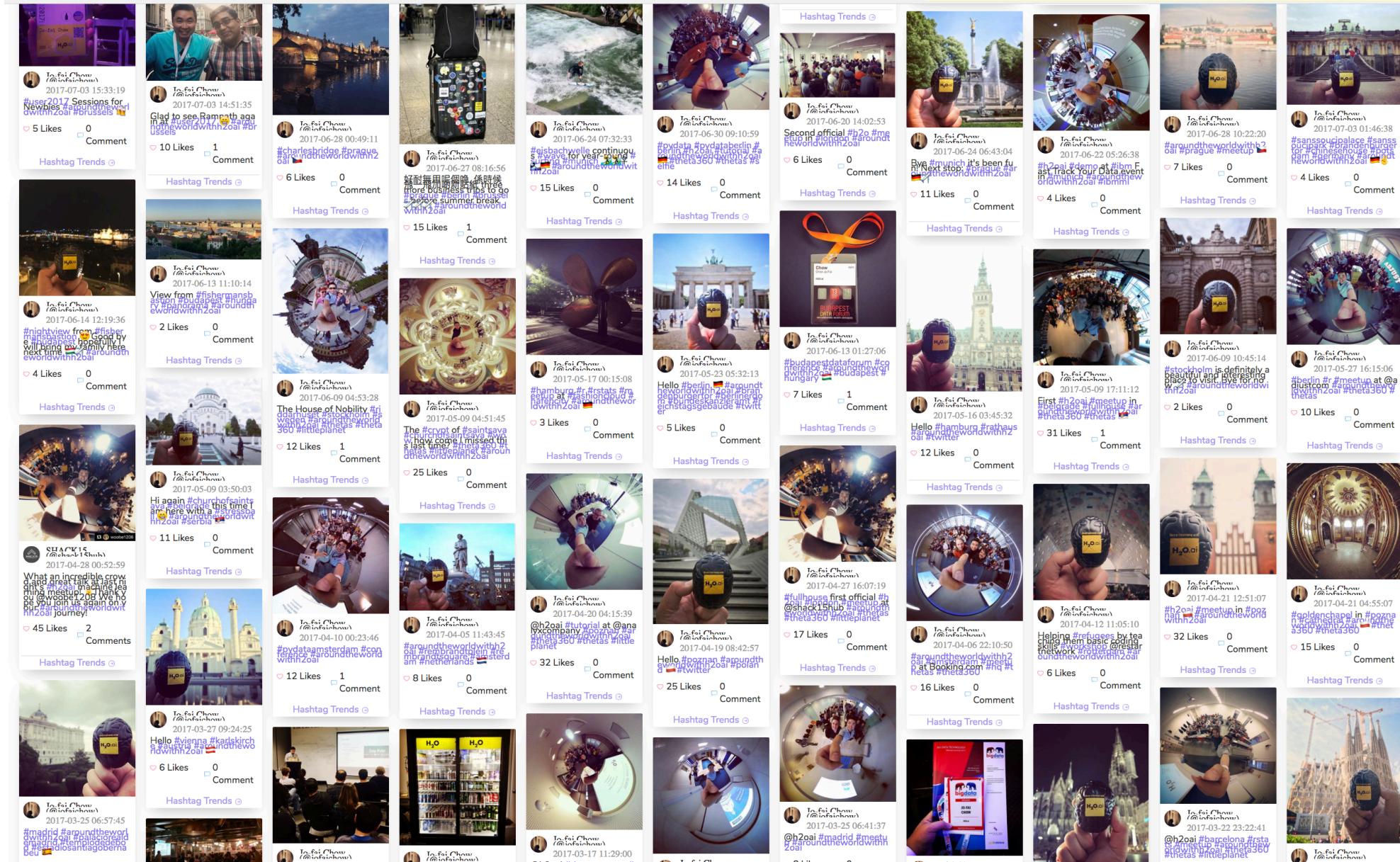


H₂O.ai

56,536 members 32 interested 50 Meetups 44 cities 18 countries

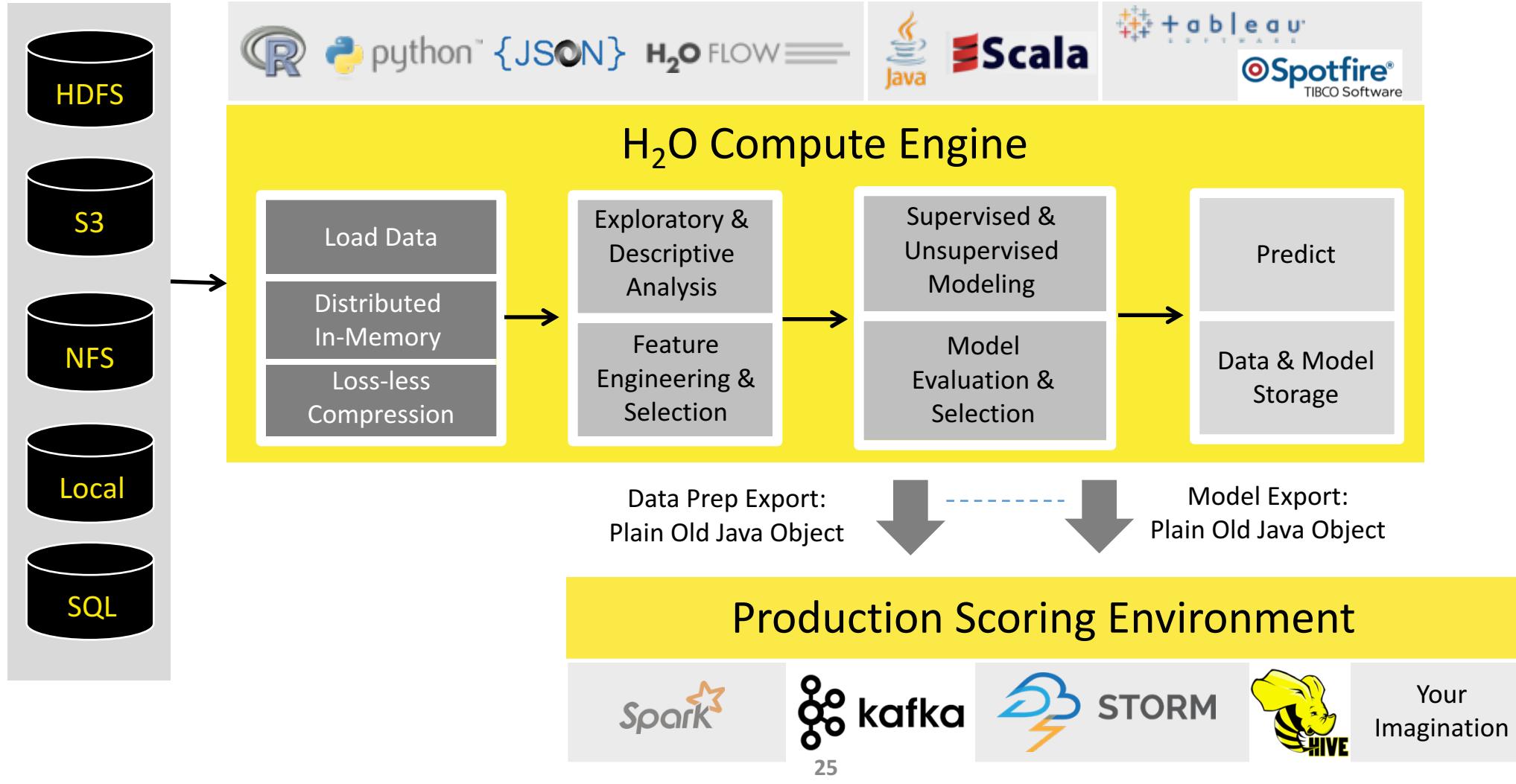
15 Meetups a month
Oct 17 – Amsterdam
Nov 17 – London

#AroundTheWorldWithH2Oai



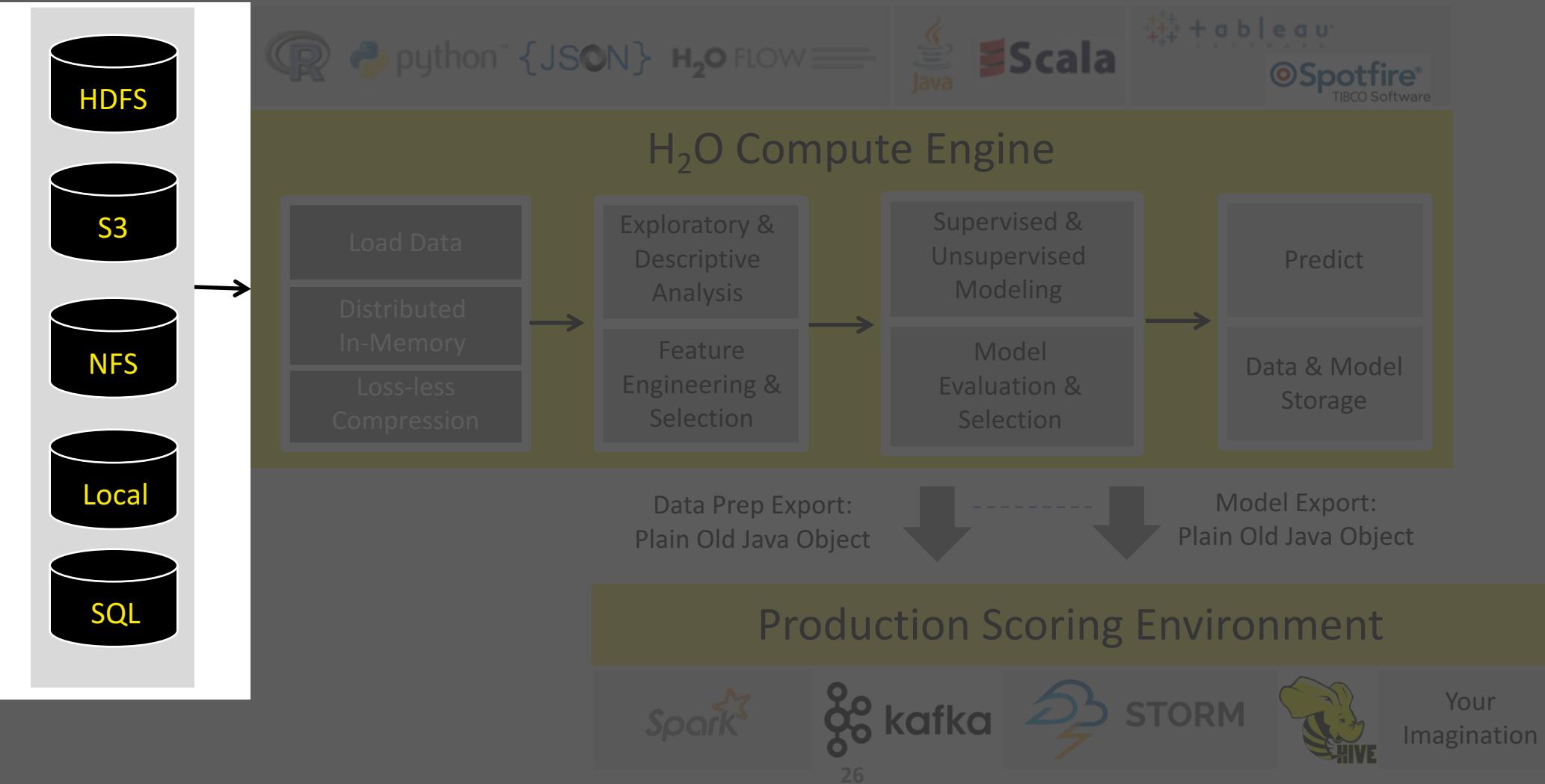
H₂O Machine Learning Platform

High Level Architecture



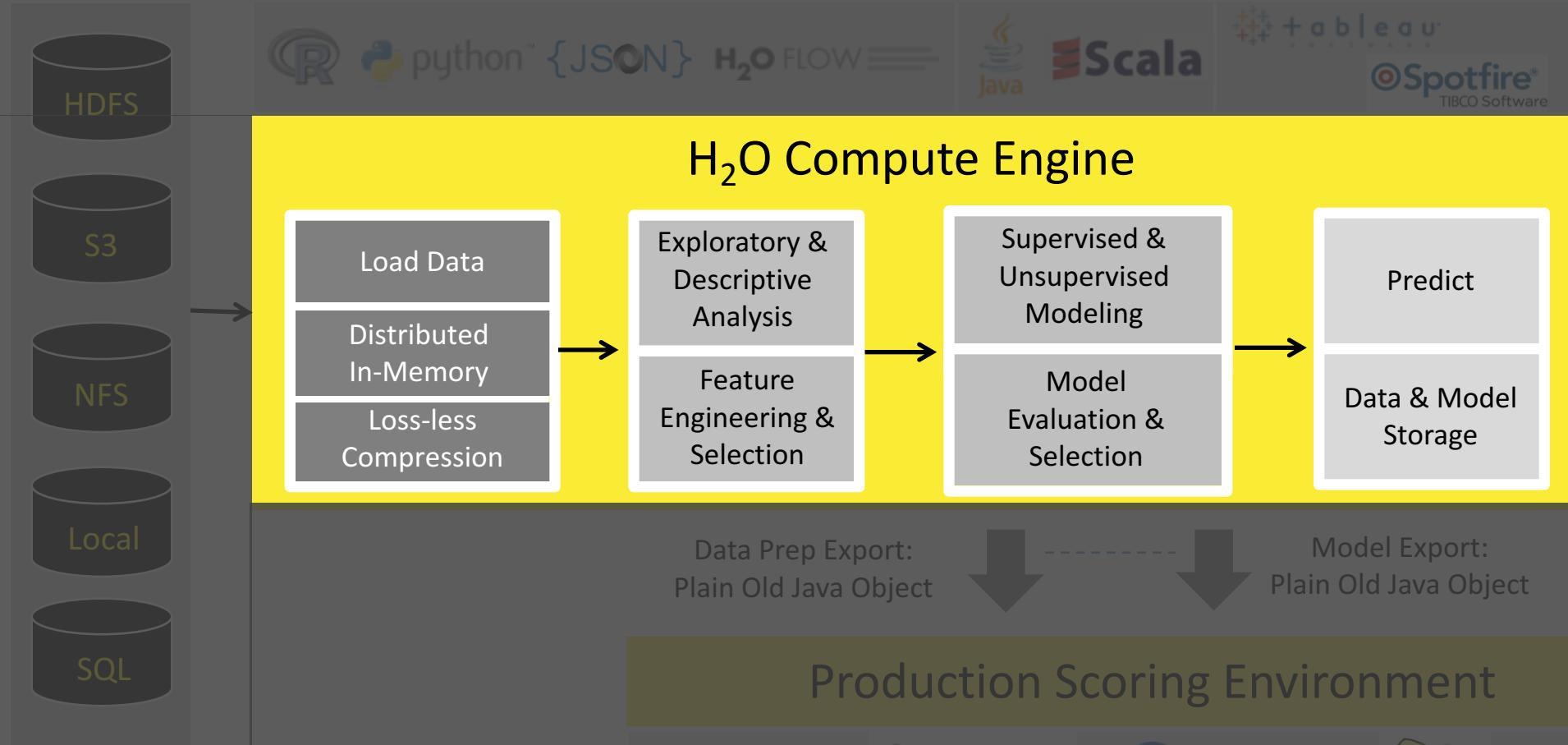
High Level Architecture

Import Data from
Multiple Sources



High Level Architecture

Fast, Scalable & Distributed
Compute Engine Written in
Java



Your
Imagination

Algorithms Overview

Supervised Learning

Statistical Analysis

- **Generalized Linear Models:** Binomial, Gaussian, Gamma, Poisson and Tweedie
- **Naïve Bayes**

Ensembles

- **Distributed Random Forest:** Classification or regression models
- **Gradient Boosting Machine:** Produces an ensemble of decision trees with increasing refined approximations

Deep Neural Networks

- **Deep learning:** Create multi-layer feed forward neural networks starting with an input layer followed by multiple layers of nonlinear transformations

Unsupervised Learning

Clustering

- **K-means:** Partitions observations into k clusters/groups of the same spatial size. Automatically detect optimal k

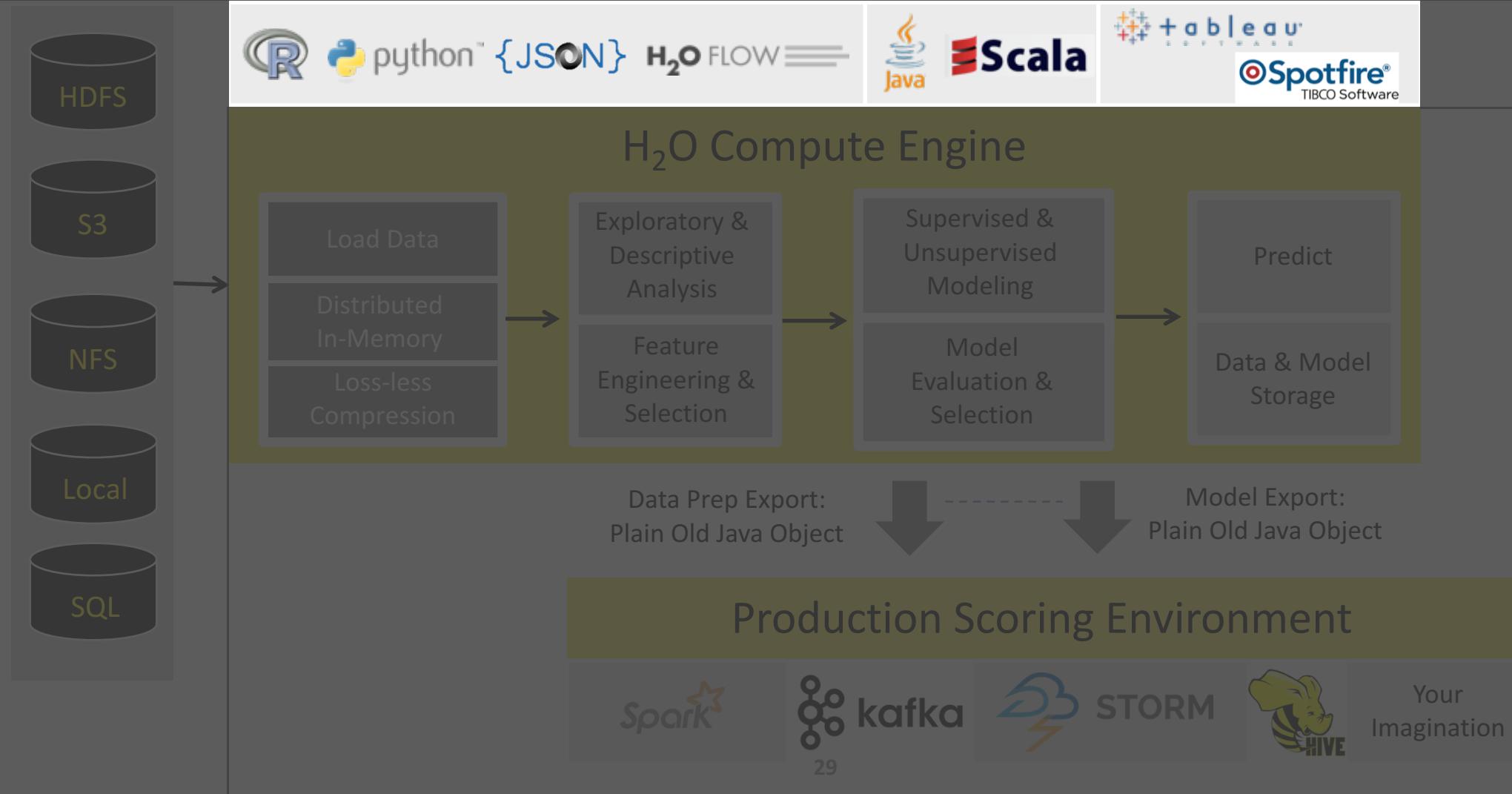
Dimensionality Reduction

- **Principal Component Analysis:** Linearly transforms correlated variables to independent components
- **Generalized Low Rank Models:** extend the idea of PCA to handle arbitrary data consisting of numerical, Boolean, categorical, and missing data

Anomaly Detection

- **Autoencoders:** Find outliers using a nonlinear dimensionality reduction using deep learning

High Level Architecture



Interface – H₂O Flow (Web)

The screenshot shows the H2O Flow web interface running in a browser window titled "H2O Flow". The URL in the address bar is "localhost:54321/flow/index.html". The browser's top bar includes standard icons for back, forward, and search, along with a user profile icon for "Jo-fai". The main interface has a header with tabs for "Flow", "Cell", "Data", "Model" (which is highlighted in yellow), "Score", "Admin", and "Help". Below the header is a toolbar with various icons for file operations like import, export, and search. A sidebar on the left is titled "Untitled Flow" and contains a "CS" section with a single entry labeled "assist". To the right of the sidebar is a large panel titled "Assistance" which lists various H2O routines with their descriptions. A context menu is open over the "Model" tab, listing options such as Aggregator..., Deep Learning..., Distributed Random Forest..., Gradient Boosting Machine..., Generalized Linear Modeling..., Generalized Low Rank Modeling..., K-means..., Naive Bayes..., Principal Components Analysis..., Stacked Ensemble..., Word2Vec..., XGBoost..., List All Models, List Grid Search Results, Import Model..., Export Model..., and Run AutoML... . On the far right, there is a "HELP" panel with sections for "Using Flow for the first time?", "Quickstart Videos", "view example Flows", "STAR H2O ON GITHUB!", "GENERAL" (with links to Flow Web UI, Importing Data, Building Models, Making Predictions, Using Flows, and Troubleshooting Flow), and "EXAMPLES" (describing Flow packs and linking to installed packs). The bottom right corner of the interface shows "Connections: 0" and the H2O logo.

Interface – R and Python

The screenshot shows the RStudio Source Editor window with the file `credit_card_example.R` open. The code is a script for training a GBM model on a credit card dataset. It includes imports, data loading from S3, model training, predictions, and a brief look at datasets.

```
~/Documents/repo_h2o/sales-engineering - master - RStudio Source Editor
credit_card_example.R
Source on Save Run Source Cell Toolbar
1 # Credit Card Example
2
3 # Datasets:
4 # https://s3.amazonaws.com/h2o-training/credit_card/credit_card_train.csv
5 # https://s3.amazonaws.com/h2o-training/credit_card/credit_card_test.csv
6
7 # Start and connect to a local H2O cluster
8 library(h2o)
9 h2o.init(nthreads = -1)
10
11 # Import datasets from s3
12 df_train = h2o.importFile("https://s3.amazonaws.com/h2o-training/credit_card/credit_card_train.csv")
13 df_test = h2o.importFile("https://s3.amazonaws.com/h2o-training/credit_card/credit_card_test.csv")
14
15 # Look at datasets
16 summary(df_train)
17 summary(df_test)
18
19 # Define features and target
20 features = colnames(df_test)
21 target = "DEFAULT_PAYMENT_NEXT_MONTH"
22
23 # Train a GBM model
24 model_gbm = h2o.gbm(x = features,
25                      y = target,
26                      training_frame = df_train,
27                      seed = 1234)
28 print(model_gbm)
29
30 # Use GBM model for making predictions
31 yhat_test = h2o.predict(model_gbm, newdata = df_test)
32 head(yhat_test)
33
34 # (Extra) Use H2O's AutoML
35 aml = h2o.automl(x = features,
36                   y = target,
37                   training_frame = df_train,
38                   max_runtime_secs = 60,
39                   seed = 1234)
40
41 # Print leaderboard
42 print(aml@leaderboard)
43
44 # Use best model for making predictions
45 best_model = aml@leader
46 yhat_test = h2o.predict(best_model, newdata = df_test)
47 head(yhat_test)
48
49
```

The screenshot shows a Jupyter Notebook interface with the notebook `credit_card_example.ipynb` open. The notebook contains Python code for connecting to a local H2O cluster, importing datasets, and summarizing them. The output cell shows the successful connection to the H2O server and the summary statistics for the datasets.

In [2]:

```
# Start and connect to a local H2O cluster
import h2o
h2o.init(nthreads = -1)

Checking whether there is an H2O instance running at http://localhost:54321.... not found.
Attempting to start a local H2O server...
Java Version: java version "1.8.0_72"; Java(TM) SE Runtime Environment (build 1.8.0_72-b15); Java HotSpot(TM) 64-Bit Server VM (build 25.72-b15, mixed mode)
Starting server from /Users/jofaichow/anaconda/lib/python2.7/site-packages/h2o/backend/bin/h2o.jar
Ice root: /var/folders/4z/p7yt7_4n4fjijlyg6g4qfbw000gn/T/tmpPdP3Av
JVM stdout: /var/folders/4z/p7yt7_4n4fjijlyg6g4qfbw000gn/T/tmpPdP3Av/h2o_jofaichow_started_from_python.out
JVM stderr: /var/folders/4z/p7yt7_4n4fjijlyg6g4qfbw000gn/T/tmpPdP3Av/h2o_jofaichow_started_from_python.err
Server is running at http://127.0.0.1:54321
Connecting to H2O server at http://127.0.0.1:54321... successful.
```

H2O cluster uptime:	02 secs
H2O cluster version:	3.13.0.3981
H2O cluster version age:	29 days
H2O cluster name:	H2O_from_python_jofaichow_id7qa
H2O cluster total nodes:	1

In [3]:

```
# Import datasets from s3
df_train = h2o.import_file("https://s3.amazonaws.com/h2o-training/credit_card/credit_card_train.csv")
df_test = h2o.import_file("https://s3.amazonaws.com/h2o-training/credit_card/credit_card_test.csv")
```

Parse progress: | 100%
Parse progress: | 100%

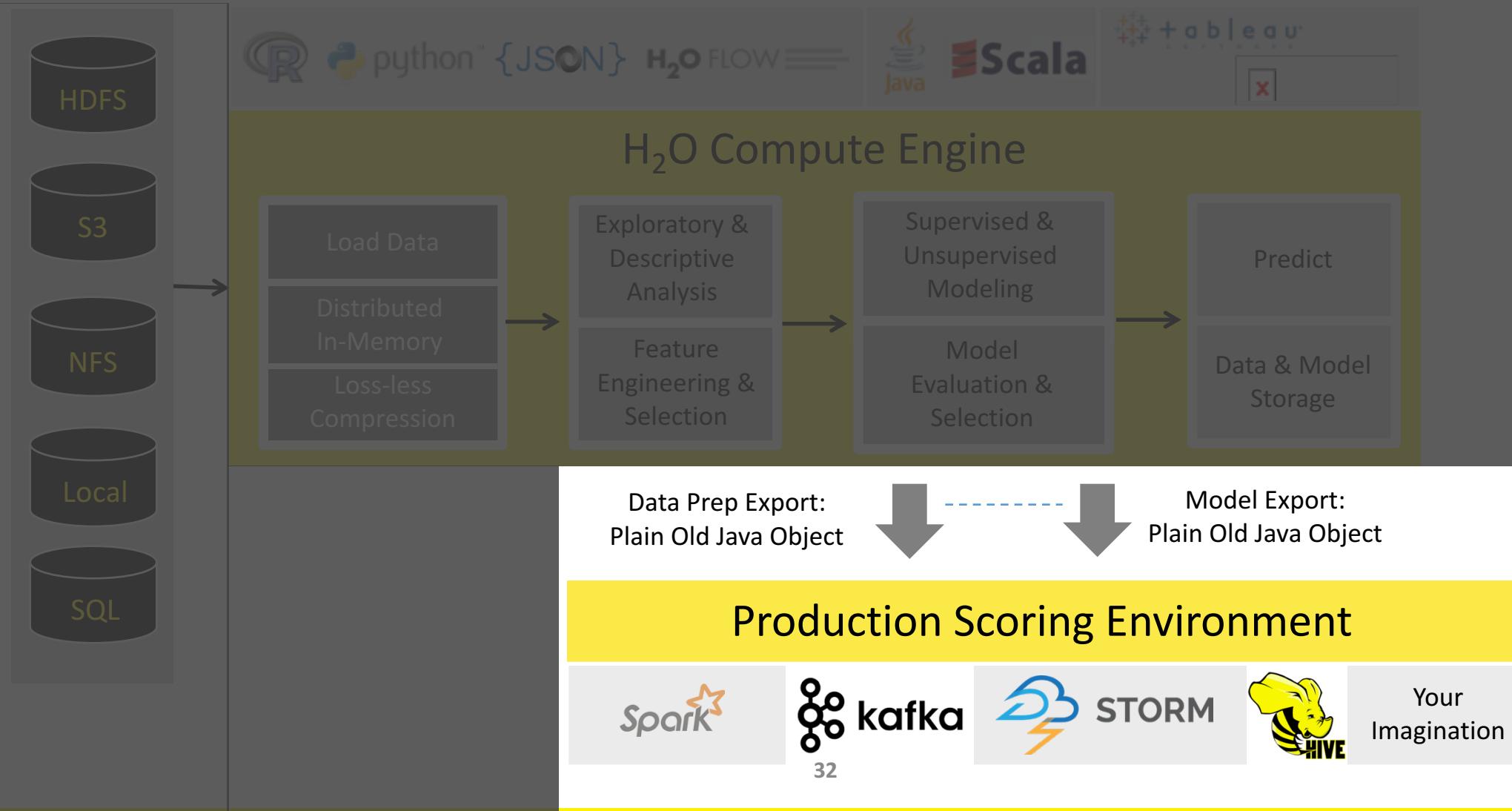
In [4]:

```
# Look at datasets
df_train.summary()
df_test.summary()
```

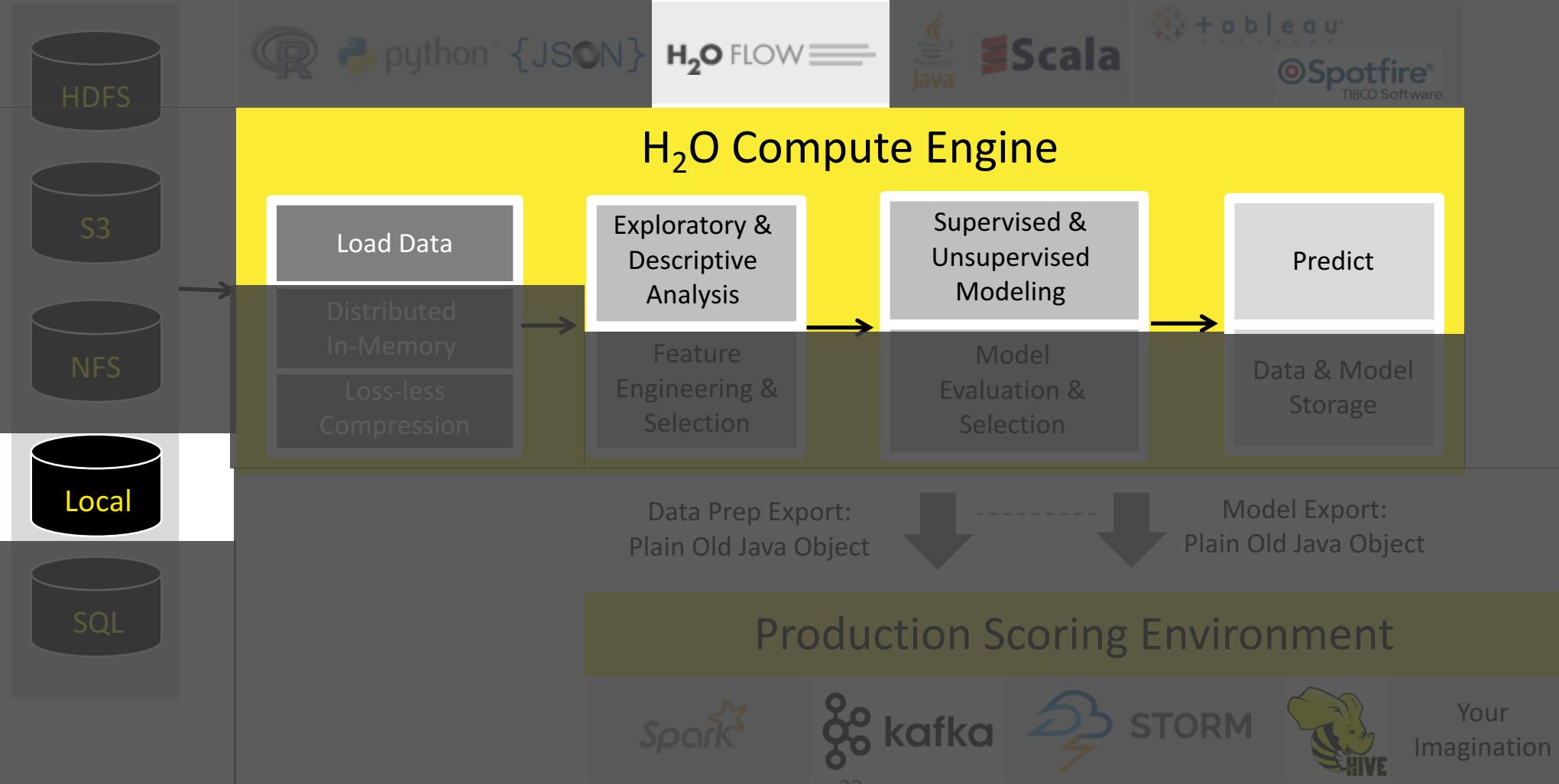
	LIMIT_BAL	SEX	EDUCATION	MARRIAGE	AGE	PAY_0	PAY_2	PAY_3	PAY_4
type	int	enum	int	int	int	int	int	int	int
mins	10000.0		0.0	0.0	21.0	-2.0	-2.0	-2.0	-2.0
mean	165471.466667		1.85	1.55578703704	35.4053240741	-0.00523148148148	-0.122361111111	-0.15537037037	-0.210601
maxs	1000000.0		6.0	3.0	79.0	8.0	8.0	8.0	8.0
sigma	128853.314839		0.779559696278	0.522505078476	9.27675421641	1.12668964211	1.20086854503	1.20727030901	1.172176
zeros	0		9	37	0	10563	11284	11309	11905
missing	0		0	0	0	0	0	0	0

High Level Architecture

Export Standalone Models
for Production



High Level Architecture



Getting Started & User Guides

H₂O

[What is H₂O?](#)
[H₂O User Guide](#) (Main docs)
[H₂O Book \(O'Reilly\)](#)
[Recent Changes](#)
[Open Source License \(Apache V2\)](#)

[Quick Start Video - Flow Web UI](#)
[Quick Start Video - R](#)
[Quick Start Video - Python](#)

[Download H₂O](#)

Sparkling Water

[What is Sparkling Water?](#)
[Sparkling Water Booklet](#)
[PySparkling Readme](#) 2.0 | 2.1 | 2.2
[RSparkling Readme](#)
[Open Source License \(Apache V2\)](#)

[Quick Start Video - Scala](#)

[Download Sparkling Water](#)

Steam

[What is Steam?](#)
[Steam User Guide](#)
[Recent Changes](#)
[Open Source License \(AGPL\)](#)

[Download Steam](#)

Deep Water (preview)

[Deep Water Readme](#)
[Deep Water Booklet](#)
[Deep Water AMI Guide](#)
[Deep Water Docker Image](#)
[Open Source License \(Apache V2\)](#)

[Launch Deep Water AMI
\(choose p2.xlarge\)](#)

Q & A

[FAQ](#)
[Issue Tracking \(JIRA\)](#)
[Stack Overflow](#)
[h2ostream Google Group](#)
[Gitter](#)
[Cross Validated](#)

For Supported Enterprise Customers
[Enterprise Support Web | Email](#)

Algorithms

Supervised Learning

Generalized Linear Modeling (GLM)	Tutorial	Booklet	Reference	Tuning
Gradient Boosting Machine (GBM)	Tutorial	Booklet	Reference	Tuning
Deep Learning	Tutorial	Booklet	Reference	Tuning
Distributed Random Forest	Tutorial	Booklet	Reference	Tuning
Naive Bayes	Tutorial	Booklet	Reference	Tuning
Stacked Ensembles	Tutorial	Booklet	Reference	Tuning
XGBoost	Tutorial	Booklet	Reference	Tuning

Unsupervised Learning

Generalized Low Rank Models (GLRM)	Tutorial	Reference
K-Means Clustering	Tutorial	Reference
Principal Components Analysis (PCA)	Tutorial	Reference

Miscellaneous

Word2vec	Tutorial	Reference
----------	--------------------------	---------------------------

TOP

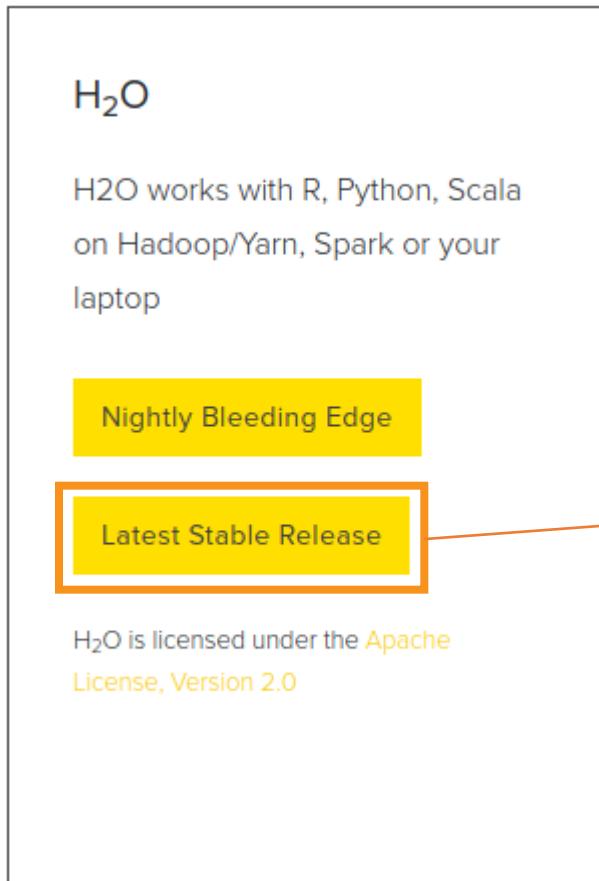
H₂O Tutorials

- Quick Start
- Regression
- Classification
- Clustering

H₂O Quick Start

www.h2o.ai/download

Prerequisite: Java version 7 or 8
(Note: Java version 9 is not yet supported)



H₂O
Version 3.14.0.3

Fast Scalable Machine Learning API
For Smarter Applications

DOWNLOAD AND RUN INSTALL IN R INSTALL IN PYTHON INSTALL ON HADOOP USE FROM MAVEN

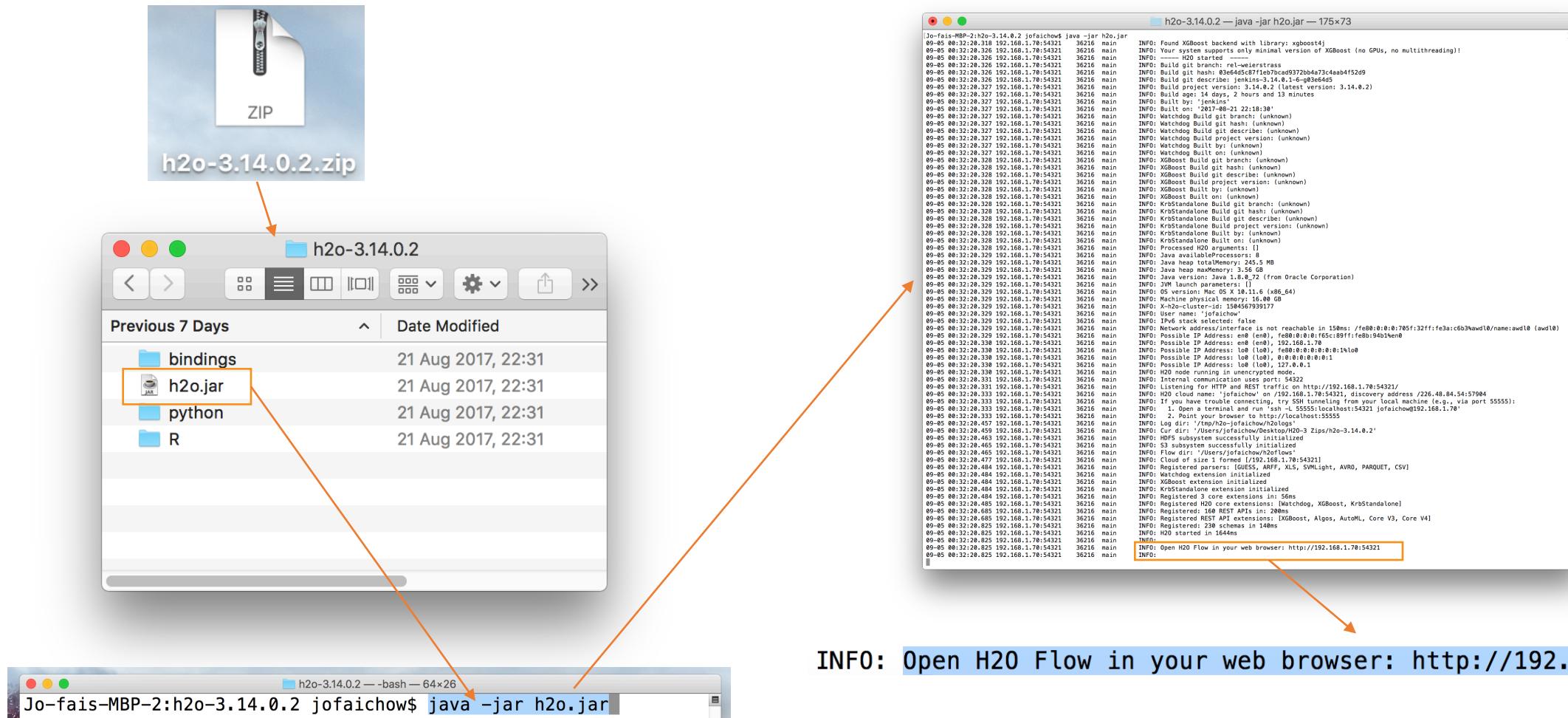
DOWNLOAD H₂O

Get started with H₂O in 3 easy steps

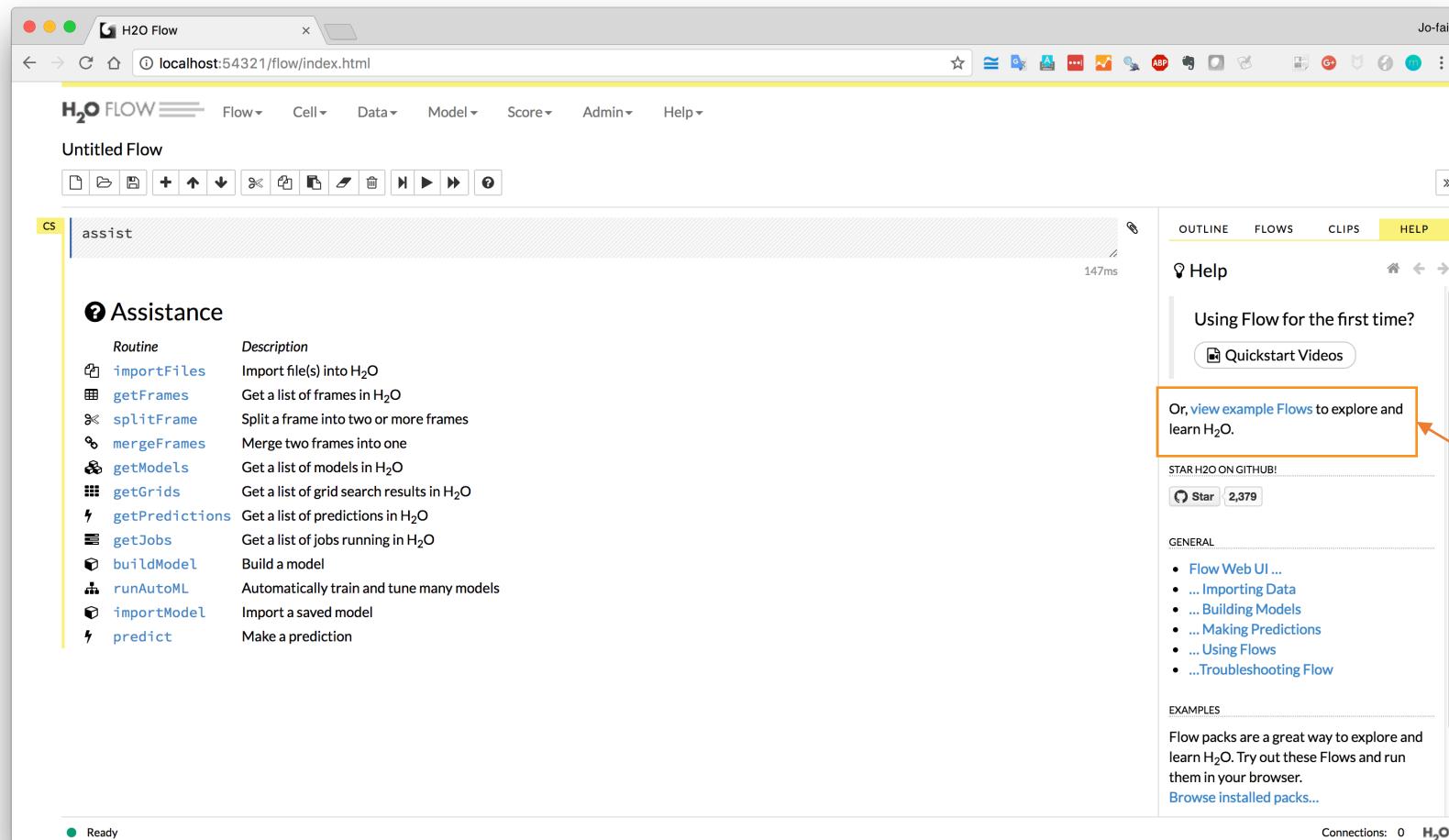
1. Download H₂O. This is a zip file that contains everything you need to get started.
2. From your terminal, run:

```
cd ~/Downloads
unzip h2o-3.14.0.3.zip
cd h2o-3.14.0.3
java -jar h2o.jar
```
3. Point your browser to <http://localhost:54321>

Install and Start H₂O Flow (Web Interface)

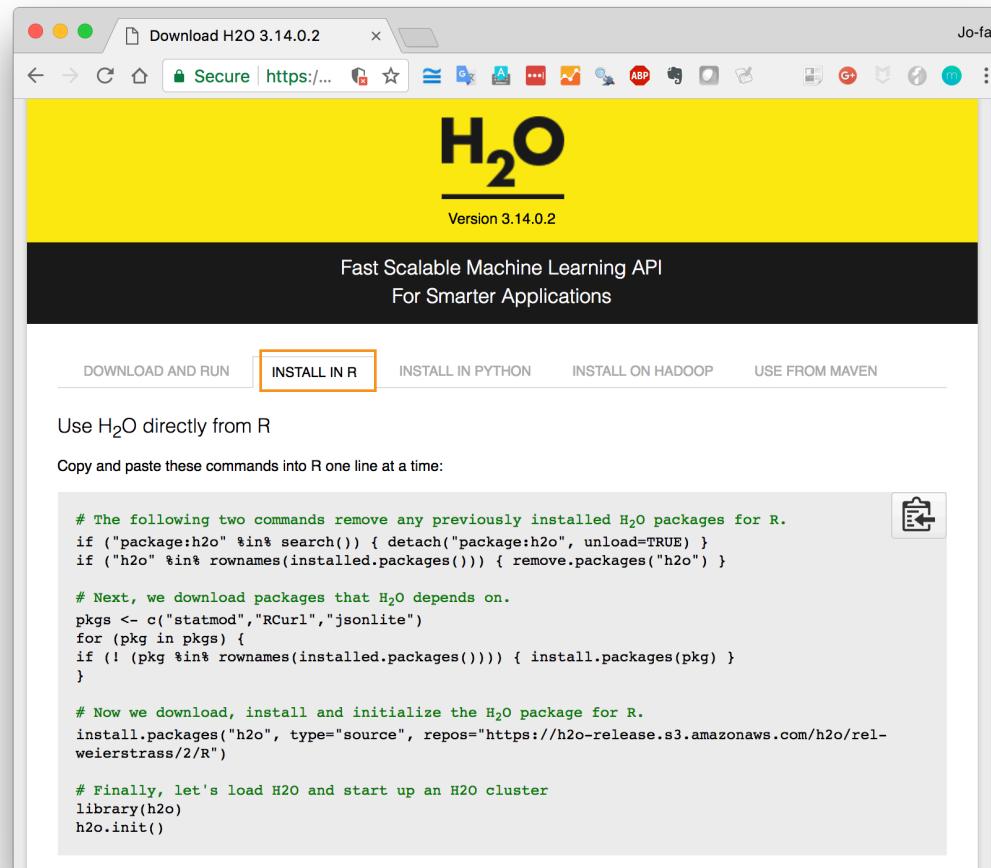


H₂O Flow (Web Interface)



More Examples

Install and Start H₂O in R / Python



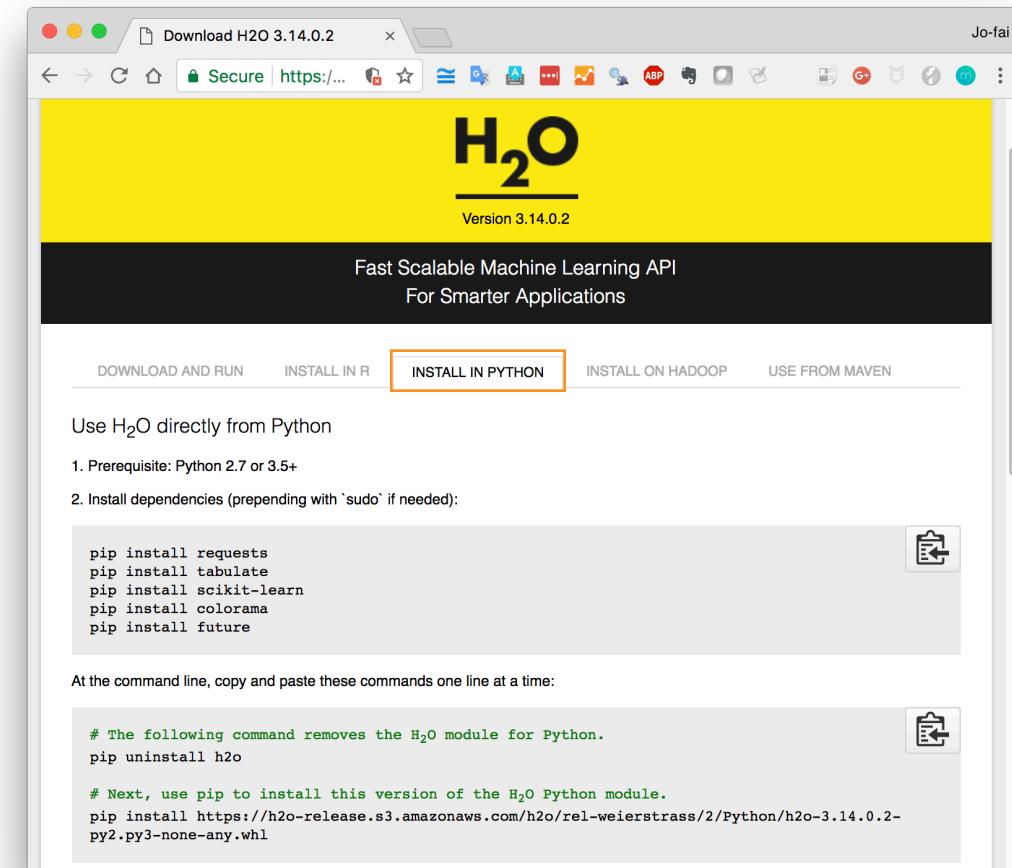
The screenshot shows the H2O download page for version 3.14.0.2. The top navigation bar includes links for 'Download H2O 3.14.0.2', 'Secure | https://...', and 'Jo-fai'. Below the header, the H2O logo and version information are displayed. A yellow banner reads 'Fast Scalable Machine Learning API For Smarter Applications'. Below the banner, there are four buttons: 'DOWNLOAD AND RUN' (disabled), 'INSTALL IN R' (highlighted in orange), 'INSTALL IN PYTHON', 'INSTALL ON HADOOP', and 'USE FROM MAVEN'. A section titled 'Use H2O directly from R' contains R code for package installation and cluster initialization. A clipboard icon is located next to the code block.

```
# The following two commands remove any previously installed H2O packages for R.
if ("package:h2o" %in% search()) { detach("package:h2o", unload=TRUE) }
if ("h2o" %in% rownames(installed.packages())) { remove.packages("h2o") }

# Next, we download packages that H2O depends on.
pkgs <- c("statmod", "RCurl", "jsonlite")
for (pkg in pkgs) {
  if (! (pkg %in% rownames(installed.packages()))) { install.packages(pkg) }
}

# Now we download, install and initialize the H2O package for R.
install.packages("h2o", type="source", repos="https://h2o-release.s3.amazonaws.com/h2o/rel-weierstrass/2/R")

# Finally, let's load H2O and start up an H2O cluster
library(h2o)
h2o.init()
```



The screenshot shows the H2O download page for version 3.14.0.2. The top navigation bar includes links for 'Download H2O 3.14.0.2', 'Secure | https://...', and 'Jo-fai'. Below the header, the H2O logo and version information are displayed. A yellow banner reads 'Fast Scalable Machine Learning API For Smarter Applications'. Below the banner, there are five buttons: 'DOWNLOAD AND RUN' (disabled), 'INSTALL IN R' (disabled), 'INSTALL IN PYTHON' (highlighted in orange), 'INSTALL ON HADOOP', and 'USE FROM MAVEN'. A section titled 'Use H2O directly from Python' contains instructions and command-line code. It lists prerequisites (Python 2.7 or 3.5) and dependencies (requests, tabulate, scikit-learn, colorama, future). A clipboard icon is located next to the dependency list. Another section provides Python installation commands at the command line.

1. Prerequisite: Python 2.7 or 3.5+
2. Install dependencies (prepending with `sudo` if needed):

```
pip install requests
pip install tabulate
pip install scikit-learn
pip install colorama
pip install future
```

At the command line, copy and paste these commands one line at a time:

```
# The following command removes the H2O module for Python.
pip uninstall h2o

# Next, use pip to install this version of the H2O Python module.
pip install https://h2o-release.s3.amazonaws.com/h2o/rel-weierstrass/2/Python/h2o-3.14.0.2-py2.py3-none-any.whl
```

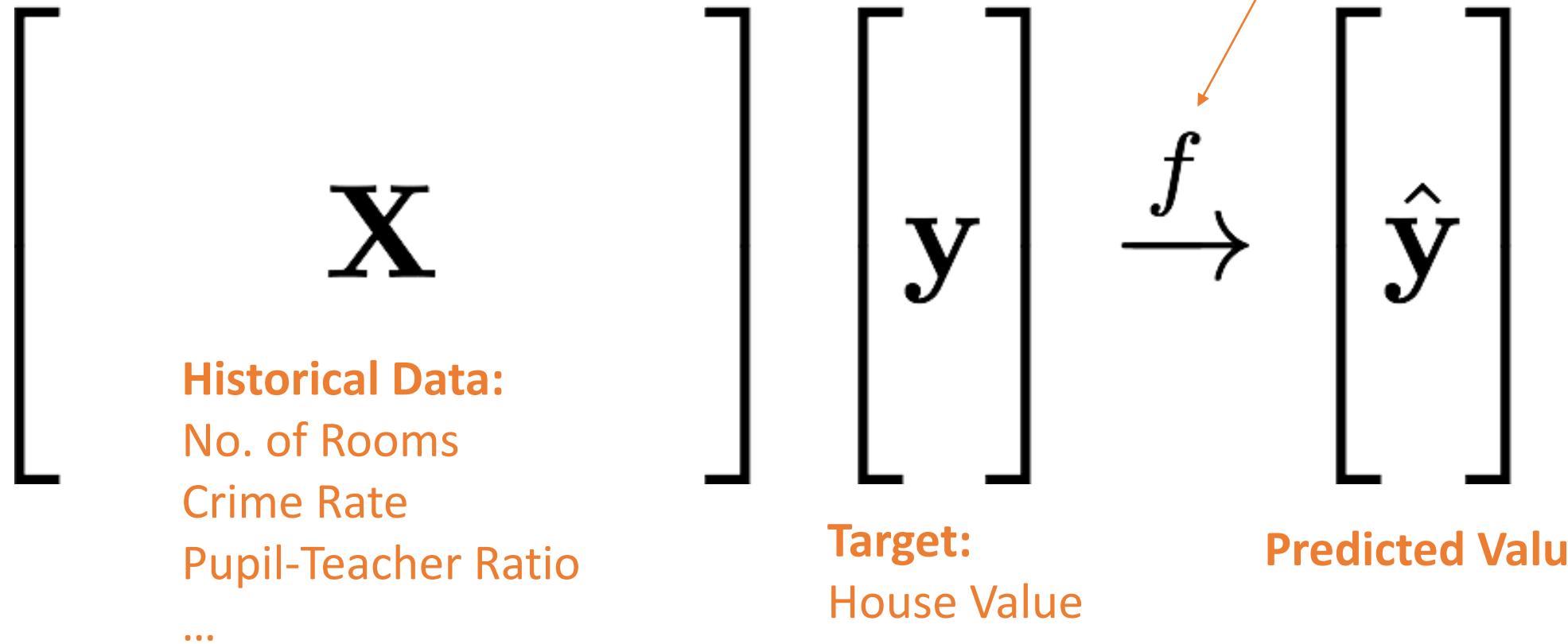
Tutorial 1 - Regression

- **Data:** Boston Housing (1978)
- **Source:** <http://www.cs.toronto.edu/~delve/data/boston/bostonDetail.html>



Supervised Learning Example

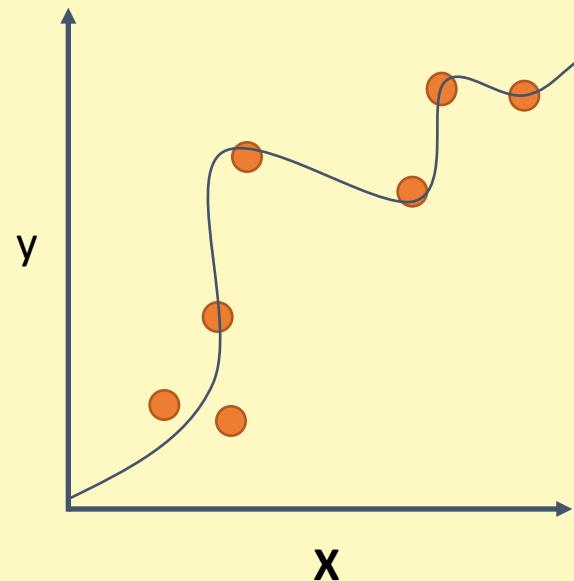
Machine Learning:
Learn Patterns
from Data



Supervised Learning – You Already Have Target Data

Regression:

How much will a customers spend?

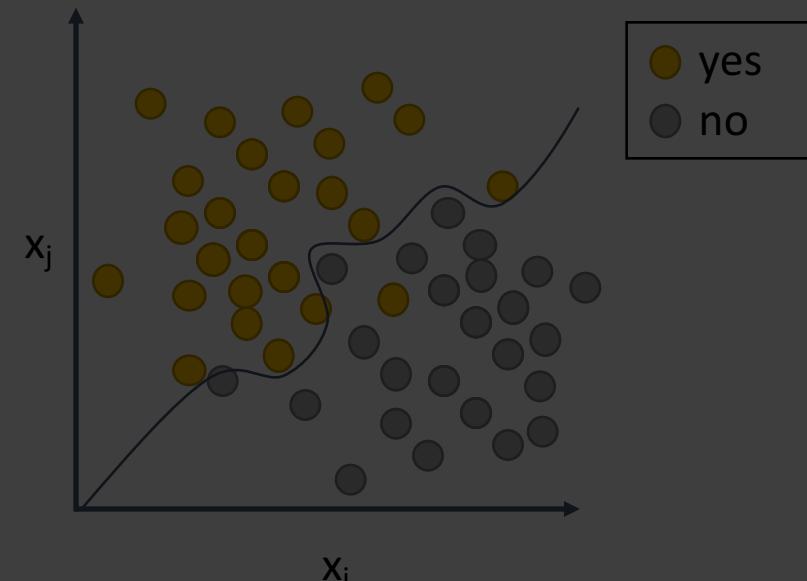


H₂O algos:

Penalized Linear Models
Random Forest
Gradient Boosting
Neural Networks
Stacked Ensembles

Classification:

Will a customer make a purchase? Yes or No



H₂O algos:

Penalized Linear Models
Naïve Bayes
Random Forest
Gradient Boosting
Neural Networks
Stacked Ensembles

Regression – Key Steps

- Import Data (CSV Files)
 - ./data/regression/ ...
 - Have a quick look
- Train a Random Forest Model
 - Look at variable importance
- Make Predictions
 - Compare with ground truth
- Build Partial Dependence Plots
 - Explain model behaviour



1. CRIM - per capita crime rate by town
2. ZN - proportion of residential land zoned for lots over 25,000 sq.ft.
3. INDUS - proportion of non-retail business acres per town.
4. CHAS - Charles River dummy variable (1 if tract bounds river; 0 otherwise)
5. NOX - nitric oxides concentration (parts per 10 million)
6. RM - average number of rooms per dwelling
7. AGE - proportion of owner-occupied units built prior to 1940
8. DIS - weighted distances to five Boston employment centres
9. RAD - index of accessibility to radial highways
10. TAX - full-value property-tax rate per \$10,000
11. PTRATIO - pupil-teacher ratio by town
12. B - $1000(Bk - 0.63)^2$ where Bk is the proportion of blacks by town
13. LSTAT - % lower status of the population
14. MEDV - Median value of owner-occupied homes in \$1000's

Target:
House Value

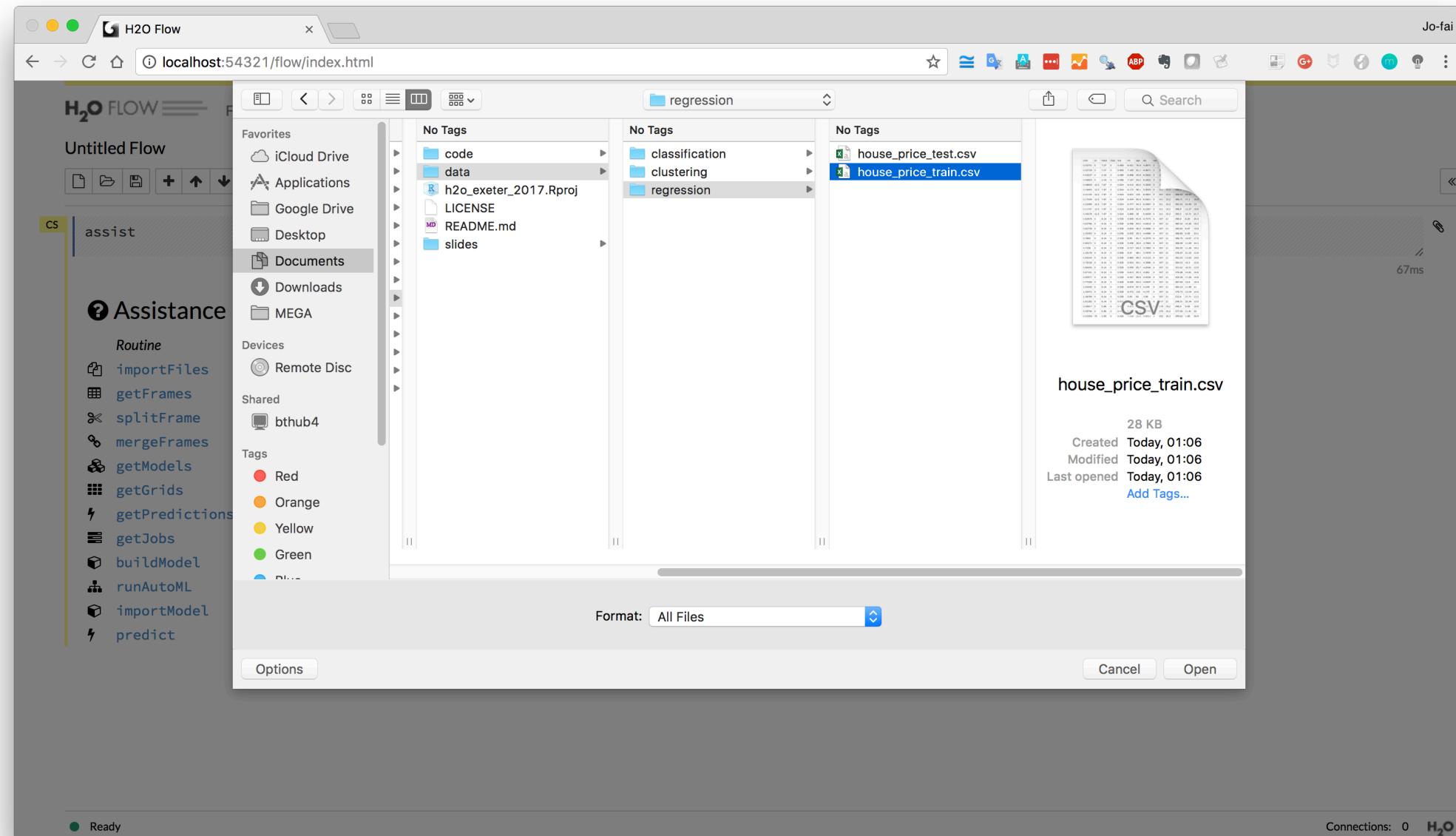
./data/regression/
house_price_train.csv

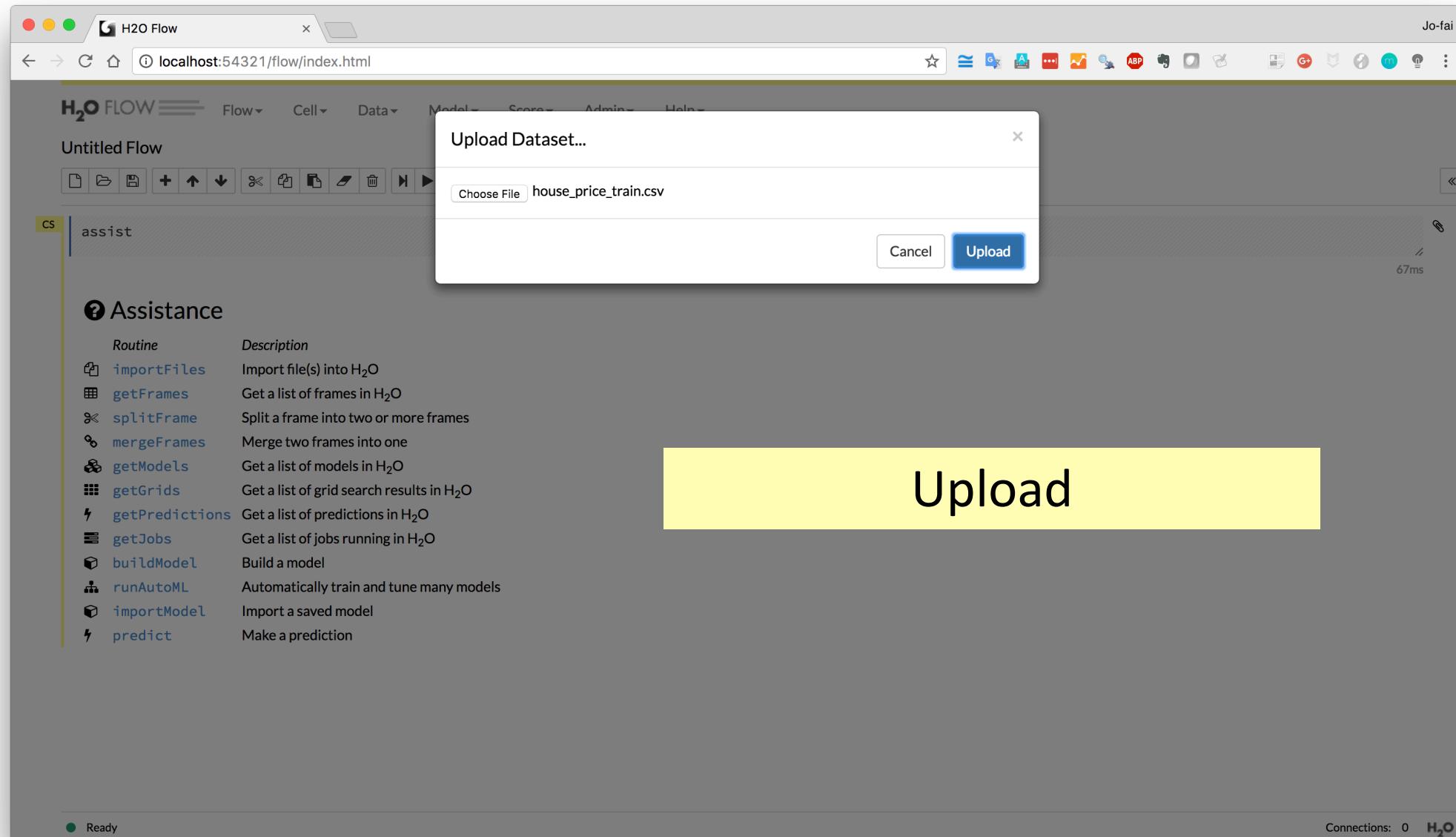
	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
1	crim	zn	indus	chas	nox	rm	age	dis	rad	tax	ptratio	b	lstat	medv	
2	0.02731	0	7.07	0	0.469	6.421	78.9	4.9671	2	242	17.8	396.9	9.14	21.6	
3	0.02729	0	7.07	0	0.469	7.185	61.1	4.9671	2	242	17.8	392.83	4.03	34.7	
4	0.03237	0	2.18	0	0.458	6.998	45.8	6.0622	3	222	18.7	394.63	2.94	33.4	
5	0.06905	0	2.18	0	0.458	7.147	54.2	6.0622	3	222	18.7	396.9	5.33	36.2	
6	0.08829	12.5	7.87	0	0.524	6.012	66.6	5.5605	5	311	15.2	395.6	12.43	22.9	
7	0.14455	12.5	7.87	0	0.524	6.172	96.1	5.9505	5	311	15.2	396.9	19.15	27.1	
8	0.21124	12.5	7.87	0	0.524	5.631	100	6.0821	5	311	15.2	386.63	29.93	16.5	
9	0.17004	12.5	7.87	0	0.524	6.004	85.9	6.5921	5	311	15.2	386.71	17.1	18.9	
10	0.22489	12.5	7.87	0	0.524	6.377	94.3	6.3467	5	311	15.2	392.52	20.45	15	
11	0.11747	12.5	7.87	0	0.524	6.009	82.9	6.2267	5	311	15.2	396.9	13.27	18.9	
12	0.09378	12.5	7.87	0	0.524	5.889	39	5.4509	5	311	15.2	390.5	15.71	21.7	
13	0.62976	0	8.14	0	0.538	5.949	61.8	4.7075	4	307	21	396.9	8.26	20.4	
14	0.63796	0	8.14	0	0.538	6.096	84.5	4.4619	4	307	21	380.02	10.26	18.2	
15	0.62739	0	8.14	0	0.538	5.834	56.5	4.4986	4	307	21	395.62	8.47	19.9	
16	1.05393	0	8.14	0	0.538	5.935	29.3	4.4986	4	307	21	386.85	6.58	23.1	
17	0.7842	0	8.14	0	0.538	5.99	81.7	4.2579	4	307	21	386.75	14.67	17.5	
18	0.80271	0	8.14	0	0.538	5.456	36.6	3.7965	4	307	21	288.99	11.69	20.2	
19	0.7258	0	8.14	0	0.538	5.727	69.5	3.7965	4	307	21	390.95	11.28	18.2	
20	1.25179	0	8.14	0	0.538	5.57	98.1	3.7979	4	307	21	376.57	21.02	13.6	
21	0.85204	0	8.14	0	0.538	5.965	89.2	4.0123	4	307	21	392.53	13.83	19.6	
22	0.75026	0	8.14	0	0.538	5.924	94.1	4.3996	4	307	21	394.33	16.3	15.6	
23	0.84054	0	8.14	0	0.538	5.599	85.7	4.4546	4	307	21	303.42	16.51	13.9	
24	0.67191	0	8.14	0	0.538	5.813	90.3	4.682	4	307	21	376.88	14.81	16.6	
25	0.95577	0	8.14	0	0.538	6.047	88.8	4.4534	4	307	21	306.38	17.28	14.8	
26	0.77299	0	8.14	0	0.538	6.495	94.4	4.4547	4	307	21	387.94	12.8	18.4	
27	1.00245	0	8.14	0	0.538	6.674	87.3	4.239	4	307	21	380.23	11.98	21	

The screenshot shows the H2O Flow web interface running on localhost:54321. The main window title is "H2O Flow". The top navigation bar includes "Data", "Model", "Score", "Admin", and "Help". The "Data" menu is open, showing options: Import Files..., Upload File..., Split Frame..., Merge Frames..., List All Frames, and Impute... . The "Upload File..." option is highlighted. On the left, there's a sidebar titled "Assistance" with a list of routines and their descriptions:

Routine	Description
importFiles	Import file(s) into H ₂ O
getFrames	Get a list of frames in H ₂ O
splitFrame	Split a frame into two or more frames
mergeFrames	Merge two frames into one
getModels	Get a list of models in H ₂ O
getGrids	Get a list of grid search results in H ₂ O
getPredictions	Get a list of predictions in H ₂ O
getJobs	Get a list of jobs running in H ₂ O
buildModel	Build a model
runAutoML	Automatically train and tune many models
importModel	Import a saved model
predict	Make a prediction

A large yellow callout box covers the right side of the screen, containing the text "Data → Upload File ...". At the bottom of the browser window, the URL "localhost:54321/flow/index.html#" is visible, along with "Connections: 0" and the H₂O logo.





H2O Flow Jo-fai

localhost:54321/flow/index.html

Untitled Flow

Flow Cell Data Model Score Admin Help

Setup Parse

PARSE CONFIGURATION

Sources house_price_train.csv
ID Key_Frame_house_price_train.hex
Parser CSV
Separator ;'44'
Column Headers Auto First row contains column names First row contains data
Options Enable single quotes as a field quotation character Delete on done

EDIT COLUMN NAMES AND TYPES

Search by column name...

	crim	Numeric	0.02731	0.02729	0.03237	0.06905	0.08829	0.14455	0.21124	0.17004	0.22489
1	zn	Numeric	0	0	0	0	12.5	12.5	12.5	12.5	12.5
2	indus	Numeric	7.07	7.07	2.18	2.18	7.87	7.87	7.87	7.87	7.87
3	chas	Numeric	0	0	0	0	0	0	0	0	0
4	nox	Numeric	0.469	0.469	0.458	0.458	0.524	0.524	0.524	0.524	0.524
5	rm	Numeric	6.421	7.185	6.998	7.147	6.012	6.172	5.631	6.004	6.377
6	age	Numeric	78.9	61.1	45.8	54.2	66.6	96.1	100	85.9	94.3
7	dis	Numeric	4.9671	4.9671	6.0622	6.0622	5.5605	5.9505	6.0821	6.5921	6.3467

Ready Connections: 0 H2O

Scroll down

H₂O Flow Jo-fai

localhost:54321/flow/index.html

H₂O FLOW Flow Cell Data Model Score Admin Help

Untitled Flow

Delete on done

EDIT COLUMN NAMES AND TYPES

Search by column name...

	Column Name	Type	Value 1	Value 2	Value 3	Value 4	Value 5	Value 6	Value 7	Value 8	Value 9
1	crim	Numeric	0.02731	0.02729	0.03237	0.06905	0.08829	0.14455	0.21124	0.17004	0.22489
2	zn	Numeric	0	0	0	0	12.5	12.5	12.5	12.5	12.5
3	indus	Numeric	7.07	7.07	2.18	2.18	7.87	7.87	7.87	7.87	7.87
4	chas	Numeric	0	0	0	0	0	0	0	0	0
5	nox	Numeric	0.469	0.469	0.458	0.458	0.524	0.524	0.524	0.524	0.524
6	rm	Numeric	6.421	7.185	6.998	7.147	6.012	6.172	5.631	6.004	6.377
7	age	Numeric	78.9	61.1	45.8	54.2	66.6	96.1	100	85.9	94.3
8	dis	Numeric	4.9671	4.9671	6.0622	6.0622	5.5605	5.9505	6.0821	6.5921	6.3467
9	rad	Numeric	2	2	3	3	5	5	5	5	5
10	tax	Numeric	242	242	222	222	311	311	311	311	311
11	ptratio	Numeric	17.8	17.8	18.7	18.7	15.2	15.2	15.2	15.2	15.2
12	b	Numeric	396.9	392.83	394.63	396.9	395.6	396.9	386.63	386.71	392.52
13	lstat	Numeric	9.14	4.03	2.94	5.33	12.43	19.15	29.93	17.1	20.45
14	medv	Numeric	21.6	34.7	33.4	36.2	22.9	27.1	16.5	18.9	15

Click "Parse"

Ready Connections: 0 H₂O

H2O Flow Jo-fai

localhost:54321/flow/index.html

Untitled Flow

Flow Cell Data Model Score Admin Help

File Edit Run View

CS parseFiles

```
source_frames: ["house_price_train.csv"]
destination_frame: "Key_Frame__house_price_train.hex"
parse_type: "CSV"
separator: 44
number_columns: 14
single_quotes: false
column_names: ["crim","zn","indus","chas","nox","rm","age","dis","rad","tax","ptratio","b","lstat","medv"]
column_types:
["Numeric","Numeric","Numeric","Numeric","Numeric","Numeric","Numeric","Numeric","Numeric","Numeric","Numeric","Numeric"]
delete_on_done: true
check_header: 1
chunk_size: 4194304
```

1.1s

Job

Run Time 00:00:00.107
Remaining Time 00:00:00.0

Type Frame
Key [Q Key_Frame_house_price_train.hex](#)

Description Parse
Status DONE
Progress 100% Done.

Actions [Q View](#)

Click “View”

Ready Connections: 0 H2O

H2O Flow Jo-fai

localhost:54321/flow/index.html

Untitled Flow

getFrameSummary "Key_Frame__house_price_train.hex"

86ms

Key_Frame__house_price_train.hex

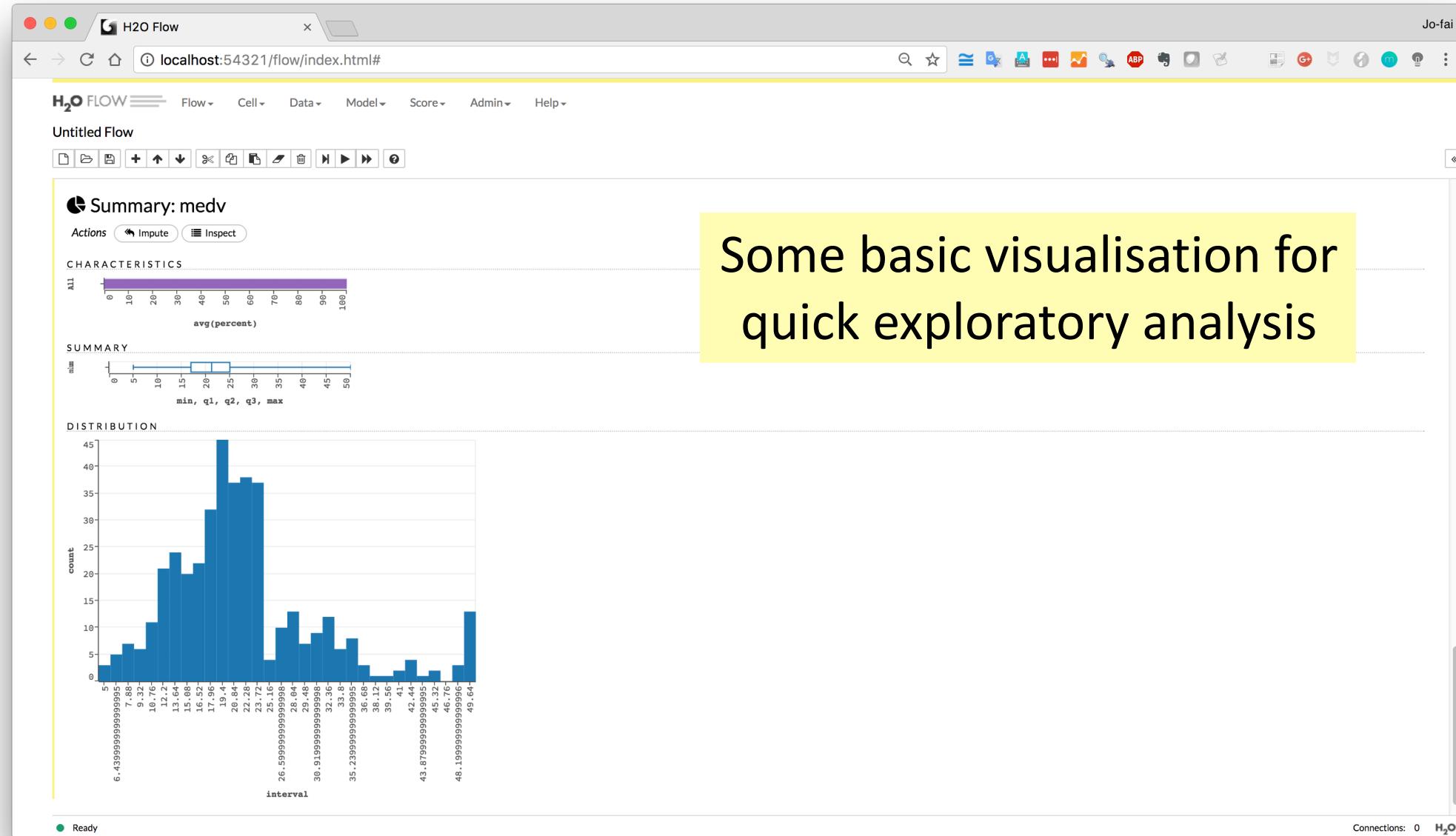
Actions: View Data Split... Build Model... Predict Download Export Delete

label	type	Missing	Zeros	+Inf	-Inf	min	max	mean	sigma	cardinality	Actions
crim	real	0	0	0	0	0.0091	73.5341	3.5674	7.9480	...	
zn	real	0	301	0	0	0	100.0	10.5872	22.2598	...	
indus	real	0	0	0	0	0.4600	27.7400	11.4093	6.8145	...	
chas	int	0	379	0	0	0	1.0	0.0688	0.2534	...	Convert to enum
nox	real	0	0	0	0	0.3850	0.8710	0.5568	0.1156	...	
rm	real	0	0	0	0	3.8630	8.7250	6.2866	0.6909	...	
age	real	0	0	0	0	2.9000	100.0	69.3889	27.8179	...	
dis	real	0	0	0	0	1.1296	10.7103	3.7177	2.0152	...	
rad	int	0	0	0	0	1.0	24.0	9.8378	8.7844	...	Convert to enum
tax	int	0	0	0	0	188.0	711.0	412.3784	170.4474	...	Convert to enum
ptratio	real	0	0	0	0	12.6000	22.0	18.4474	2.1618	...	
b	real	0	0	0	0	0.3200	396.9000	354.4032	94.1752	...	
lstat	real	0	0	0	0	1.7300	37.9700	12.7920	7.0987	...	
medv	real	0	0	0	0	5.0	50.0	22.6248	9.1850	...	

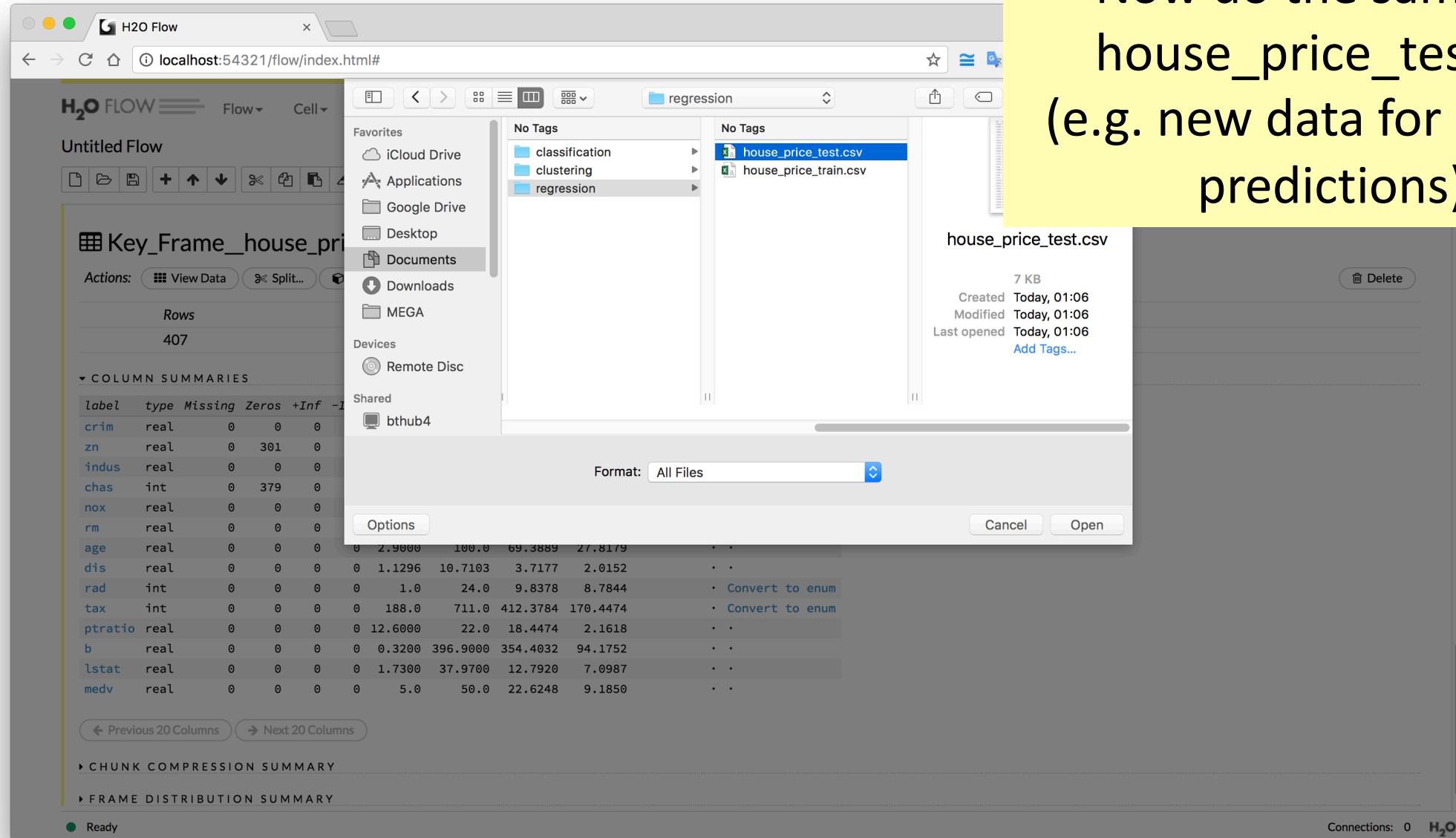
Previous 20 Columns Next 20 Columns

Ready Connections: 0 H2O

Click on any variable
e.g. crim, medv



Now do the same for
`house_price_test.csv`
(e.g. new data for making
predictions)



The screenshot shows the H2O Flow interface on a Mac OS X desktop. The window title is "H2O Flow" and the URL is "localhost:54321/flow/index.html#". The main pane displays a flow named "Untitled Flow" containing a step titled "Key_Frame_house_price". Below it, a table titled "COLUMN SUMMARIES" shows details for 14 columns: label, type, Missing, Zeros, +Inf, -Inf, crim, zn, indus, chas, nox, rm, age, dis, rad, tax, ptratio, b, lstat, and medv. The "Format:" dropdown is set to "All Files". A modal dialog box is open over the flow, showing the file browser. The left sidebar lists "Favorites" (iCloud Drive, Applications, Google Drive, Desktop, Documents, Downloads, MEGA) and "Devices" (Remote Disc). The right sidebar shows a list of files under "No Tags": "classification", "clustering", "regression", "house_price_test.csv" (selected), and "house_price_train.csv". The "house_price_test.csv" file is highlighted with a yellow background and has its details displayed: size 7 KB, created, modified, and last opened all at "Today, 01:06", and a "Add Tags..." button. The "Cancel" and "Open" buttons are visible at the bottom of the modal.

Show all data
Data → List All Frames

Untitled Flow

Type Frame
Key **Q Key_Frame_house_price_hex**

Description Parse
Status DONE
Progress 100% Done.

Actions **Q View**

getFrames

29ms

Frames

	Rows	Columns	Size
Key_Frame_house_price_hex	99	14	4KB
Key_Frame_house_price_train.hex	407	14	12KB

Q Predict on selected frames... **Delete selected frames**

localhost:54321/flow/index.html#

Connections: 0 **H₂O**

The screenshot shows the H2O Flow web application interface. The top navigation bar includes the H2O logo, a sidebar icon, and dropdown menus for Flow, Cell, Data, Model (which is currently selected), Score, Admin, and Help. The title bar displays the URL `localhost:54321/flow/index.html#`. The main workspace is titled "Untitled Flow" and contains a "Frames" section. Under "Frames", there are two entries: "Key_Frame_house_price_test.hex" and "Key_Frame_house_price_train.hex". Each entry has a "Build Model..." button, a "Predict..." button, and an "Inspect" button. Below the frames, there are buttons for "Predict on selected frames..." and "Delete selected frames". A progress bar indicates "Progress 100%" and "Done.". On the right side, there is a table showing the structure of the "Key_Frame_house_price_train.hex" frame, with 99 rows, 14 columns, and a size of 4KB. The "Model" dropdown menu is open, listing various modeling options: Aggregator..., Deep Learning..., Distributed Random Forest..., Gradient Boosting Machine..., Generalized Linear Modeling..., Generalized Low Rank Modeling..., K-means..., Naive Bayes..., Principal Components Analysis..., Stacked Ensemble..., Word2Vec..., XGBoost.... A yellow callout box on the right side of the screen contains the text "Train a predictive model" on the first line and "Start with Random Forest" on the second line.

H2O Flow Jo-fai

localhost:54321/flow/index.html#

H2O FLOW Flow Cell Data Model Score Admin Help

Untitled Flow

Build a Model

Select an algorithm: Distributed Random Forest

PARAMETERS

model_id my_random_forest
training_frame Key_Frame_house_price_train.hex
validation_frame (Choose...)
nfolds 0
response_column medv
ignored_columns Search...

GRID ?

Destination id for this model; auto-generated if not specified.

Id of the training data frame (Not required, to allow initial validation of model parameters).

Id of the validation data frame.

Number of folds for N-fold cross-validation (0 to disable or >= 2).

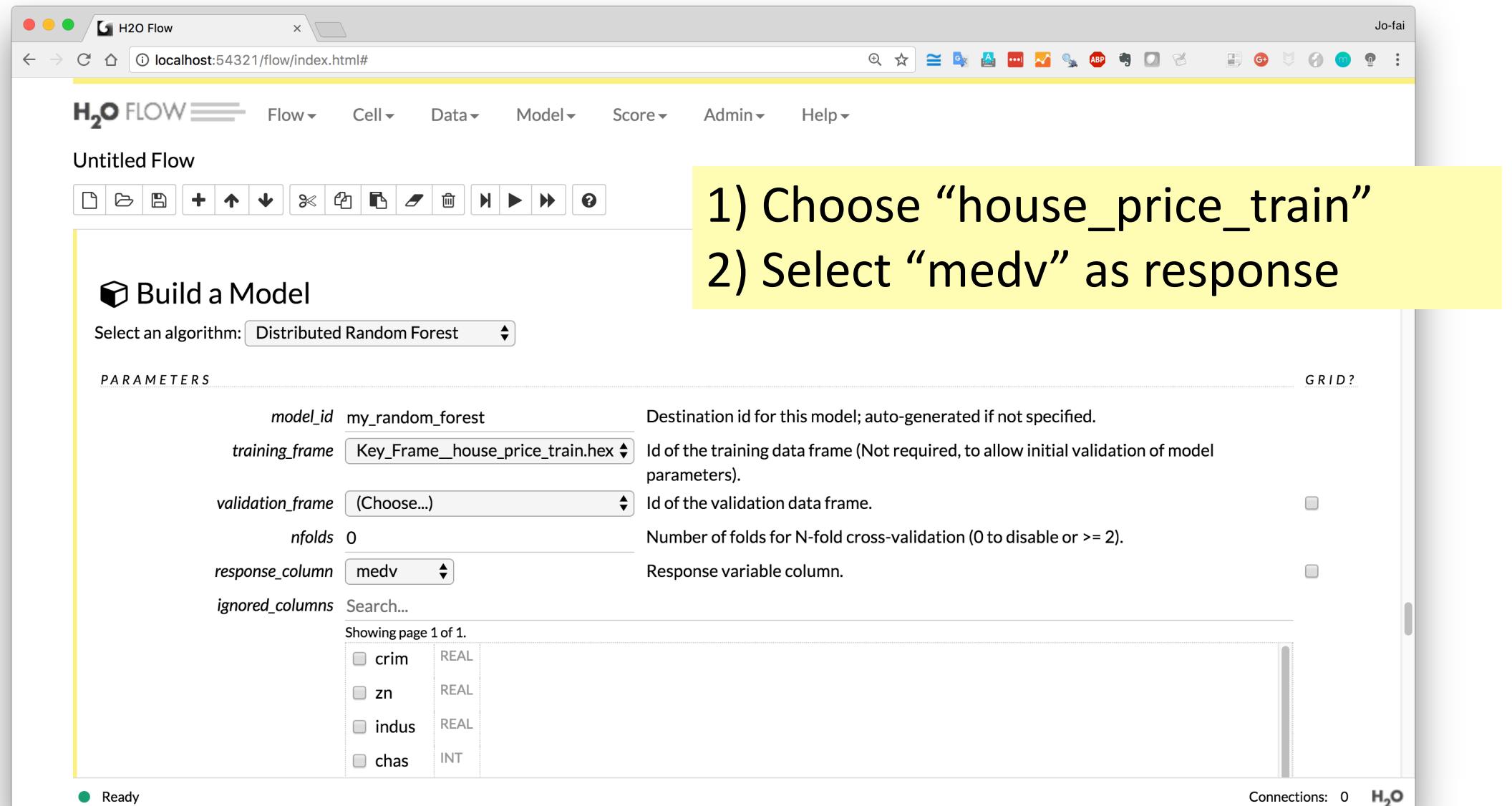
Response variable column.

Showing page 1 of 1.

crim	REAL
zn	REAL
indus	REAL
chas	INT

Connections: 0 H2O

1) Choose “house_price_train”
2) Select “medv” as response



Untitled Flow

Flow ▾ Cell ▾ Data ▾ Model ▾ Score ▾ Admin ▾ Help ▾

Only show columns with more than % missing values.

ignore_const_cols All None

Ignore constant columns.

ntrees 50 Number of trees.

max_depth 20 Maximum tree depth.

min_rows 1 Fewest allowed (weighted) observations in a leaf.

nbins 20 For numerical columns (real/int), build a histogram of (at least) this many bins, then split at the best point

seed 54321 Seed for pseudo random number generator (if applicable)

mtries -1 Number of variables randomly sampled as candidates at each split. If set to -1, defaults to \sqrt{p} for classification and $p/3$ for regression (where p is the # of predictors)

sample_rate 0.6320000290870667 Row sample rate per tree (from 0.0 to 1.0)

ADVANCED GRID ?

score_each_iteration Whether to score during each iteration of model training.

score_tree_interval 0 Score the model after every so many trees. Disabled if set to 0.

fold_column (Choose...) Column with cross-validation fold index assignment per observation.

offset_column (Choose...) Offset column. This will be added to the combination of columns before applying the link function.

weights_column (Choose...) Column with observation weights. Giving some observation a weight of zero is equivalent to excluding it from the dataset; giving an observation a relative weight of 2 is equivalent to repeating that row twice. Negative weights are not allowed.

nbins_top_level 1024 For numerical columns (real/int), build a histogram of (at most) this many bins at the root level, then decrease by factor of two per level

nbins_cats 1024 For categorical columns (factors), build a histogram of this many bins, then split at the best point. Higher values can lead to more overfitting.

r2_stopping 1.7976931348623157e+308 r2_stopping is no longer supported and will be ignored if set - please use stopping_rounds, stopping_metric and stopping_tolerance

Connections: 0 H₂O

Enter a seed number
(if you want to reproduce
your results in future)

Untitled Flow

Flow ▾ Cell ▾ Data ▾ Model ▾ Score ▾ Admin ▾ Help ▾

histogram_type: AUTO What type of histogram to use for finding optimal split points

categorical_encoding: AUTO Encoding scheme for categorical features

EXPERT

build_tree_one_node: Run on one node only; no network overhead but fewer cpus used. Suitable for small datasets.

sample_rate_per_class: A list of row sample rates per class (relative fraction for each class, from 0.0 to 1.0), for each tree

binomial_double_trees: For binary classification: Build 2x as many trees (one per class) - can lead to higher accuracy.

col_sample_rate_change_per_level: 1 Relative change of the column sampling rate for every level (from 0.0 to 2.0)

calibrate_model: Use Platt Scaling to calculate calibrated class probabilities. Calibration can provide more accurate estimates of class probabilities.

calibration_frame: (Choose...) Calibration frame for Platt Scaling

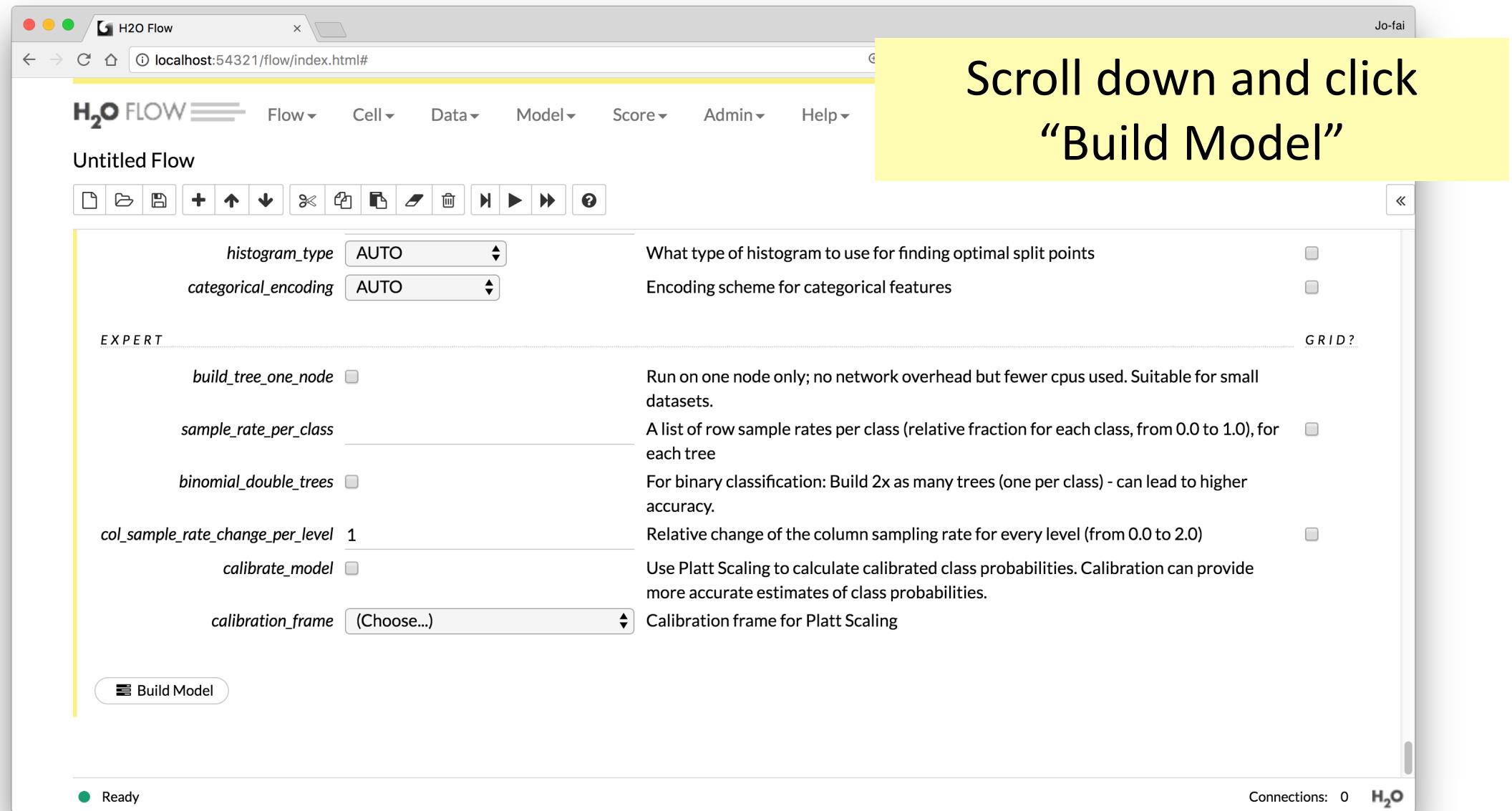
Build Model

Ready

Connections: 0 H₂O

Jo-fai

Scroll down and click “Build Model”



H2O Flow Jo-fai

localhost:54321/flow/index.html#

H2O FLOW Flow Cell Data Model Score Admin Help

Untitled Flow

Build Model

```
buildModel 'drf', {"model_id":"my_random_forest","training_frame":"Key_Frame__house_price_train.hex","nfolds":0,"response_column":"medv","ignored_columns":[],"ignore_const_cols":true,"ntrees":50,"max_depth":20,"min_rows":1,"nbins":20,"seed":54321,"mtries":-1,"sample_rate":0.6320000290870667,"score_each_iteration":false,"score_tree_interval":0,"nbins_top_level":1024,"nbins_cats":1024,"r2_stopping":1.7976931348623157e+308,"stopping_rounds":0,"stopping_metric":"AUTO","stopping_tolerance":0.001,"max_runtime_secs":0,"checkpoint":"","col_sample_rate_per_tree":1,"min_split_improvement":0.00001,"histogram_type":"AUTO","categorical_encoding":"AUTO","build_tree_one_node":false,"sample_rate_per_class":[],"binomial_double_trees":false,"col_sample_rate_change_per_level":1,"calibrate_model":false}
```

1.1s

Job

Run Time 00:00:00.654

Remaining Time 00:00:00.0

Type Model

Key my_random_forest

Description DRF

Status DONE

Progress 100%

Done.

Actions View

View the model

Ready Connections: 0 H2O

H2O Flow Jo-fai

localhost:54321/flow/index.html#

Untitled Flow

Model

Model ID: my_random_forest

Algorithm: Distributed Random Forest

Actions: Refresh, Predict..., Download POJO, Download Model Deployment Package (MOJO), Export, Inspect, Delete, Download Gen Model

MODEL PARAMETERS

SCORING HISTORY - DEVIANCE

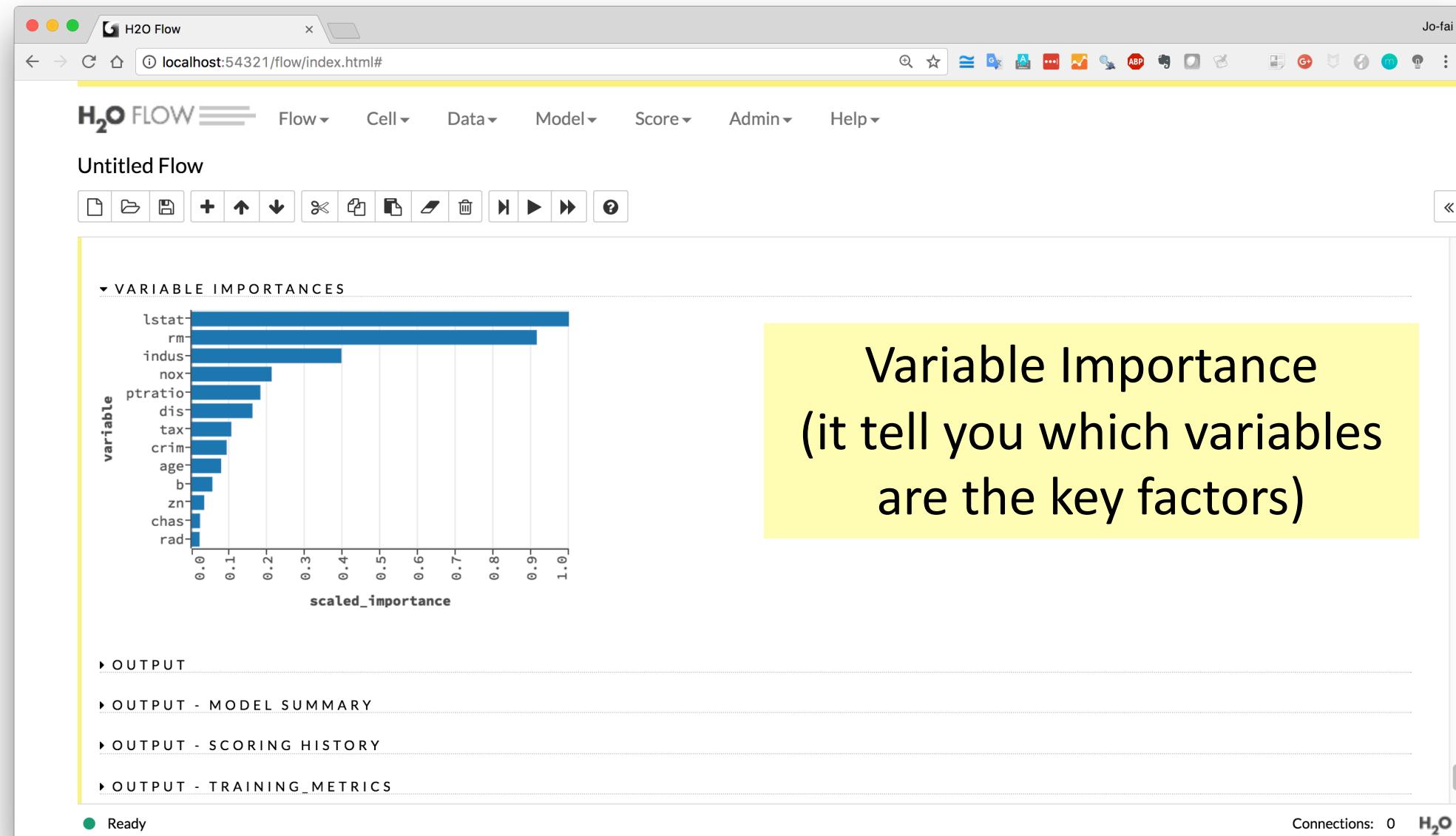
A line graph titled 'SCORING HISTORY - DEVIANCE'. The y-axis is labeled 'training_deviance' and ranges from 10 to 26. The x-axis is labeled 'number_of_trees' and ranges from 0 to 50. The data points show a sharp initial drop in deviance as the number of trees increases from 0 to about 10, followed by a more gradual, nearly linear decrease towards a minimum around 11.5.

VARIABLE IMPORTANCES

Ready

Connections: 0 H2O

Performance metric
Deviance
(Mean Squared Error)



The screenshot shows the H2O Flow interface on a Mac OS X desktop. The title bar says "H2O Flow". The menu bar includes "Flow", "Cell", "Data", "Model", "Score", "Admin", and "Help". A toolbar below the menu has icons for file operations like Open, Save, Copy, Paste, and various selection tools. The main workspace is titled "Untitled Flow" and contains a single step labeled "predict". This step has a yellow border and a status bar indicating "17ms". To the left of the step is a vertical yellow bar labeled "cs". Below the step, there's a "Predict" section with fields: "Name: my_predictions", "Model: my_random_forest", "Frame: Key_Frame_house_price_test.hex", and an "Actions" button labeled "⚡ Predict". A context menu is open over the "predict" step, with options "Predict...", "Partial Dependence Plots...", and "List All Predictions". The status bar at the bottom shows "localhost:54321/flow/index.html#" and "H2O".

We have a model.
We can now make some predictions on the test dataset.

- 1) Score → Predict
- 2) Select the model
- 3) Select “test” data
- 4) Click “Predict”

The screenshot shows the H2O Flow web application running in a browser window. The title bar reads "H2O Flow" and the address bar shows "localhost:54321/flow/index.html#". The main interface has a top navigation bar with "Flow", "Cell", "Data", "Model", "Score", "Admin", and "Help" dropdowns. Below the navigation is a toolbar with various icons for file operations like open, save, and copy. The main workspace contains a code editor with the following content:

```
predict model: "my_random_forest", frame: "Key_Frame__house_price_test.hex", predictions_frame: "my_predictions"
```

A yellow callout box is overlaid on the right side of the workspace, containing the text:

Combine predictions with original test data so we can compare the predictions with ground truth.

The code editor shows a timestamp "53ms" at the bottom right. On the left, there's a "Prediction" section with an "Inspect" button. Below it is a "PREDICTION" section listing various metrics and parameters:

- model: my_random_forest
- model_checksum: 2387713827283948032
- frame: Key_Frame__house_price_test.hex
- frame_checksum: -3619964009561634304
- description: .
- model_category: Regression
- scoring_time: 1507165392633
- predictions: my_predictions
- MSE: 10.860009
- RMSE: 3.295453
- nobs: 99
- r2: 0.872714
- mean_residual_deviance: 10.860009
- mae: 2.415040
- rmsle: 0.162378

At the bottom left of the workspace, there is a button labeled "Combine predictions with frame". The bottom status bar indicates "Ready" and "Connections: 0".

H2O Flow Jo-fai

localhost:54321/flow/index.html#

H2O FLOW Flow Cell Data Model Score Admin Help

Untitled Flow

combined-my_predictions

Prediction

Ground Truth

DATA

← Previous 20 Columns → Next 20 Columns

Row	predict	crim	zn	indus	chas	nox	rm	age	dis	rad	tax	ptratio	b	lstat	medv
1	27.0670	0.0063	18.0	2.3100	0	0.5380	6.5750	65.2000	4.0900	1.0	296.0	15.3000	396.9000	4.9800	24.0
2	25.9460	0.0299	0	2.1800	0	0.4580	6.4300	58.7000	6.0622	3.0	222.0	18.7000	394.1200	5.2100	28.7000
3	16.9850	1.2325	0	8.1400	0	0.5380	6.1420	91.7000	3.9769	4.0	307.0	21.0	396.9000	18.7200	15.2000
4	15.5270	0.9884	0	8.1400	0	0.5380	5.8130	100.0	4.0952	4.0	307.0	21.0	394.5400	19.8800	14.5000
5	15.0130	1.1308	0	8.1400	0	0.5380	5.7130	94.1000	4.2330	4.0	307.0	21.0	360.1700	22.6000	12.7000
6	15.6990	1.1517	0	8.1400	0	0.5380	5.7010	95.0	3.7872	4.0	307.0	21.0	358.7700	18.3500	13.1000
7	21.8920	0.0801	0	5.9600	0	0.4990	5.8500	41.5000	3.9342	5.0	279.0	19.2000	396.9000	8.7700	21.0
8	20.3540	0.1751	0	5.9600	0	0.4990	5.9660	30.2000	3.8473	5.0	279.0	19.2000	393.4300	10.1300	24.7000
9	29.0590	0.0276	75.0	2.9500	0	0.4280	6.5950	21.8000	5.4011	3.0	252.0	18.3000	395.6300	4.3200	30.8000
10	20.9370	0.0887	21.0	5.6400	0	0.4390	5.9630	45.7000	6.8147	4.0	243.0	16.8000	395.5600	13.4500	19.7000
11	23.3900	0.0205	85.0	0.7400	0	0.4100	6.3830	35.7000	9.1876	2.0	313.0	17.3000	396.9000	5.7700	24.7000
12	19.1160	0.1717	25.0	5.1300	0	0.4530	5.9660	93.4000	6.8185	8.0	284.0	19.7000	378.0800	14.4400	16.0
13	23.7010	0.1103	25.0	5.1300	0	0.4530	6.4560	67.8000	7.2255	8.0	284.0	19.7000	396.9000	6.7300	22.2000
14	21.9820	0.0839	0	12.8300	0	0.4370	5.8740	36.6000	4.5026	5.0	398.0	18.7000	396.0600	9.1000	20.3000
15	23.2093	0.0355	25.0	4.8600	0	0.4260	6.1670	46.7000	5.4007	4.0	281.0	19.0	390.6400	7.5100	22.9000
16	22.9100	0.0715	0	1.1900	0	0.1100	6.1210	56.8000	3.7476	3.0	217.0	18.5000	395.1500	8.1100	22.2000

Ready Connections: 0 H2O

H2O Flow Jo-fai

localhost:54321/flow/index.html#

Untitled Flow

Actions: View Data Split... Build Model... Predict Download Export Delete

combined-my_predictions

Rows: 99 Columns: 15 Compressed Size: 5KB

▼ COLUMN SUMMARIES

label	type	Missing	Zeros	+Inf	-Inf	min	max	mean	sigma	cardinality	Actions
predict	real	0	0	0	0	8.1564	46.3040	22.7819	7.9191	...	
crim	real	0	0	0	0	0.0063	88.9762	3.8033	10.9319	...	
zn	real	0	71	0	0	0	95.0	14.5556	27.1545	...	
indus	real	0	0	0	0	0.7400	27.7400	10.0166	6.9690	...	
chas	int	0	92	0	0	0	1.0	0.0707	0.2576	...	Convert to enum
nox	real	0	0	0	0	0.3890	0.8710	0.5462	0.1174	...	
rm	real	0	0	0	0	3.5610	8.7800	6.2766	0.7525	...	
age	real	0	0	0	0	9.9000	100.0	65.2283	29.3788	...	
dis	real	0	0	0	0	1.1781	12.1265	4.1131	2.4283	...	
rad	int	0	0	0	0	1.0	24.0	8.3636	8.3207	...	Convert to enum
tax	int	0	0	0	0	187.0	711.0	391.2121	160.1494	...	Convert to enum
ptratio	real	0	0	0	0	12.6000	22.0	18.4889	2.1887	...	
b	real	0	0	0	0	6.6800	396.9000	366.0096	78.0843	...	
lstat	real	0	0	0	0	2.8800	34.4100	12.0817	7.3215	...	
medv	real	0	0	0	0	5.6000	50.0	22.1545	9.2839	...	

◀ Previous 20 Columns ▶ Next 20 Columns

► CHUNK COMPRESSION SUMMARY

● Ready Connections: 0 H2O

Download the results

H2O Flow Jo-fai

localhost:54321/flow/index.html#

H2O FLOW Flow Cell Data Model Score Admin Help

Untitled Flow

buildPartialDependence

cs

Partial Dependence

Save Destination PDP as:

Model: my_random_forest

Frame: Key_Frame_house_price_train.hex

nbins 20

Select columns?

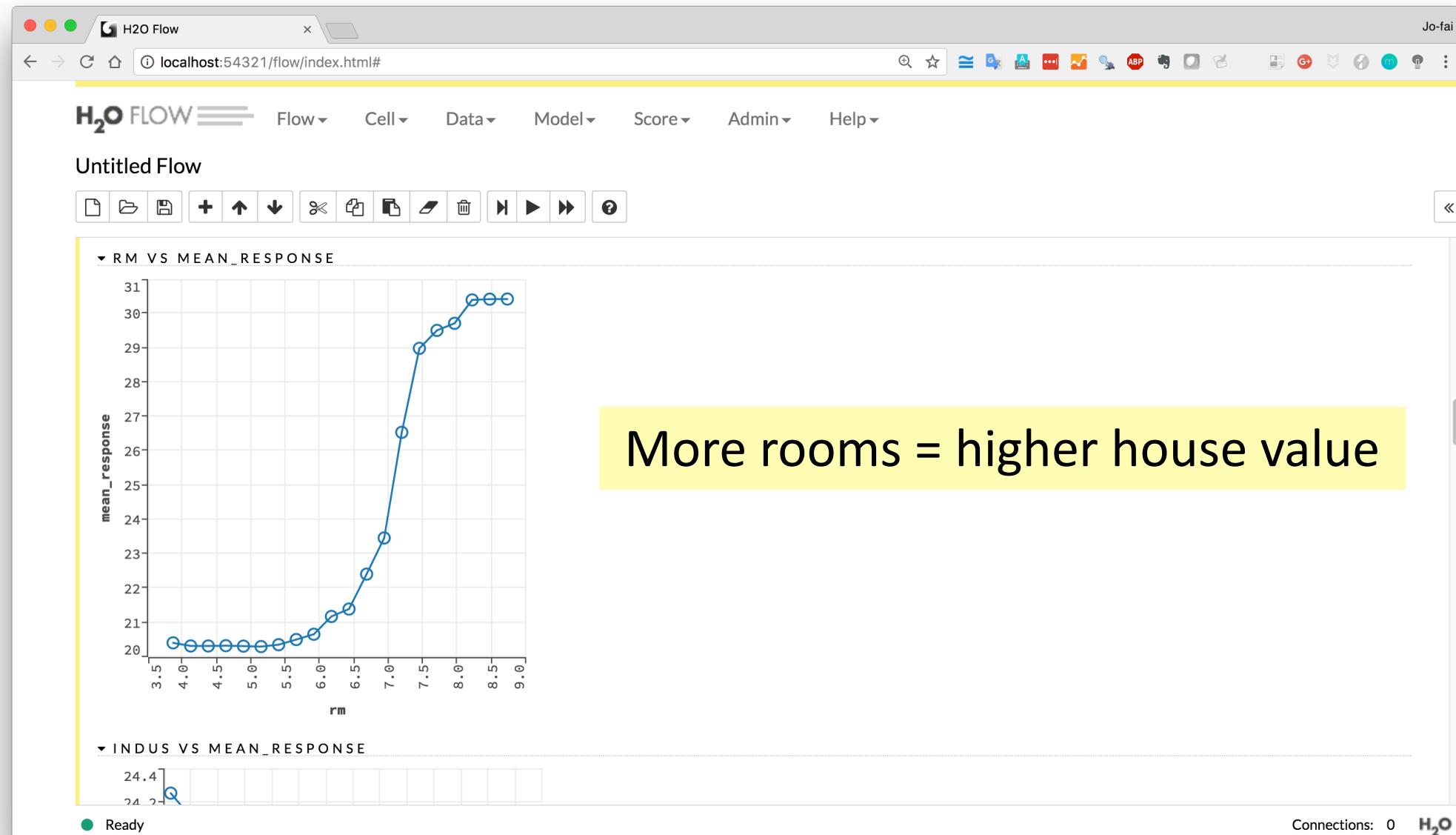
Actions: Compute

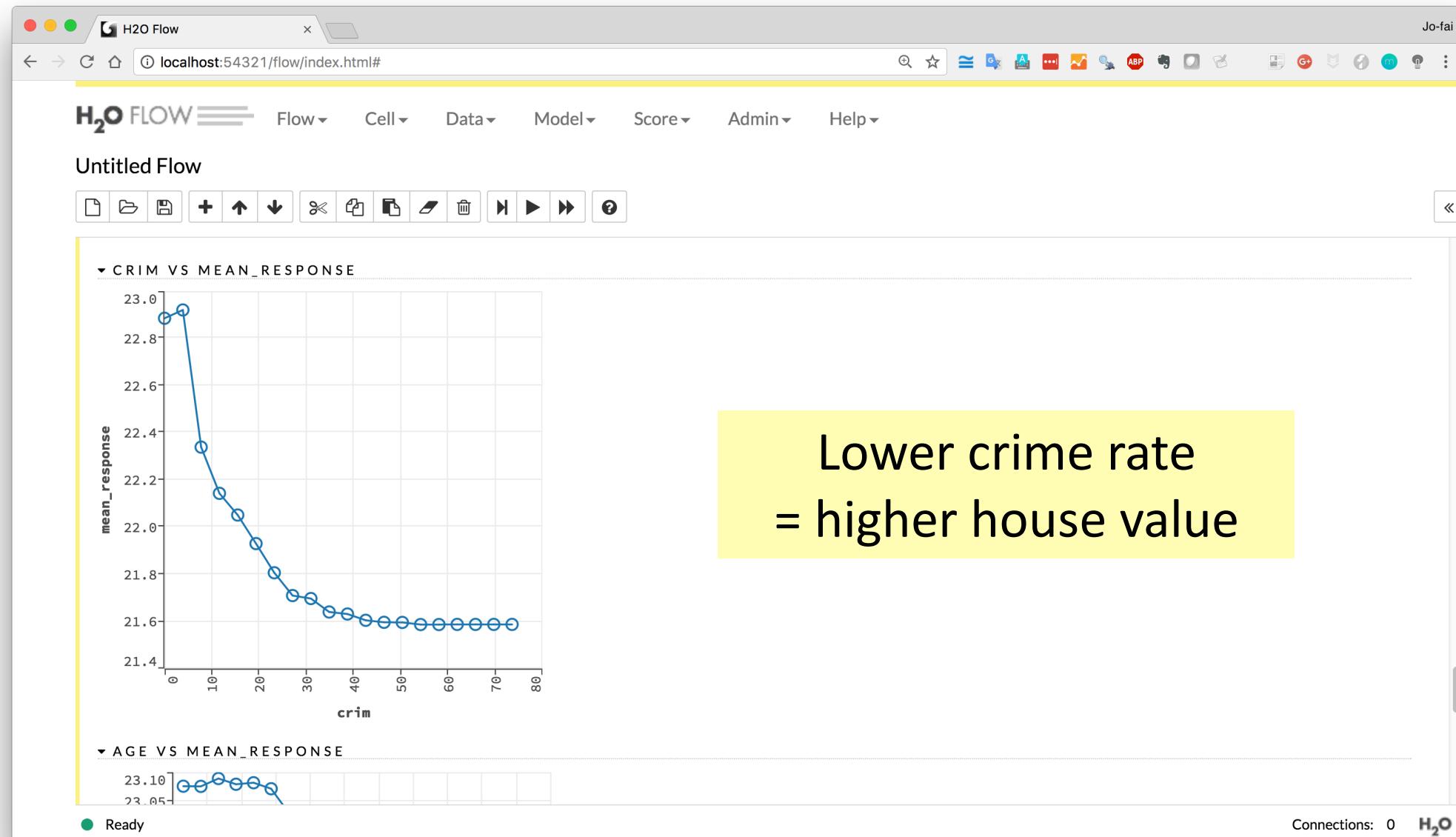
How many levels should PDP compute. More levels will make it slower.

Checking this will allow you to select custom columns for PDP. By default, the top 10 features are used. Those features are sorted by variable importance.

Connections: 0 H2O

Using Partial Dependence Plots to further explain results



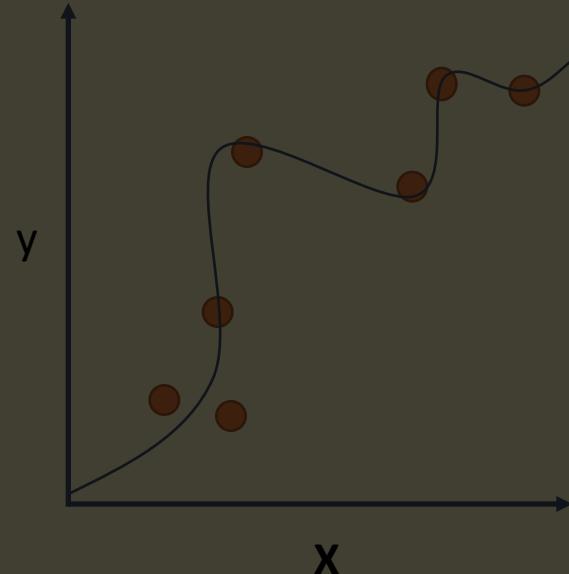


Tutorial 2 - Classification

- **Data:** Human Activity Recognition Using Smartphone Sensors (2012)
- **Source:**
<https://archive.ics.uci.edu/ml/datasets/human+activity+recognition+using+smartphones>

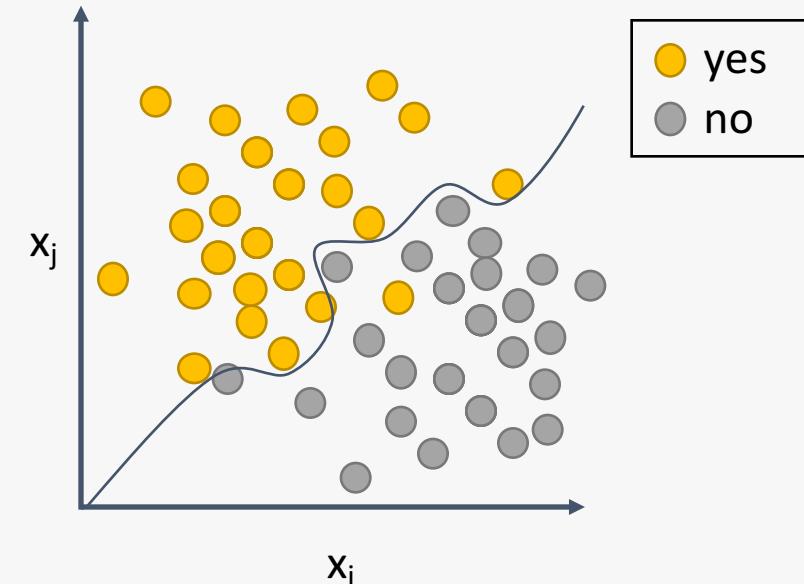
Supervised Learning – You Already Have Target Data

Regression:
How much will a customers spend?



H₂O algos:
Penalized Linear Models
Random Forest
Gradient Boosting
Neural Networks
Stacked Ensembles

Classification:
Will a customer make a purchase? Yes or No



H₂O algos:
Penalized Linear Models
Naïve Bayes
Random Forest
Gradient Boosting
Neural Networks
Stacked Ensembles

- Smartphone Sensor Data
 - 561 Features
 - 6 Activities
 - Walking
 - Walking Upstairs
 - Walking Downstairs
 - Sitting
 - Standing
 - Laying
 - Train (7k)
 - Test (3k)

[About](#) [Citation Policy](#) [Donate a Data Set](#) [Contact](#)

Repository Web [Search](#)

[View ALL Data Sets](#)



Machine Learning Repository
Center for Machine Learning and Intelligent Systems

Human Activity Recognition Using Smartphones Data Set

[Download](#) [Data Folder](#) [Data Set Description](#)

Abstract: Human Activity Recognition database built from the recordings of 30 subjects performing activities of daily living (ADL) while carrying a waist-mounted smartphone with embedded inertial sensors.

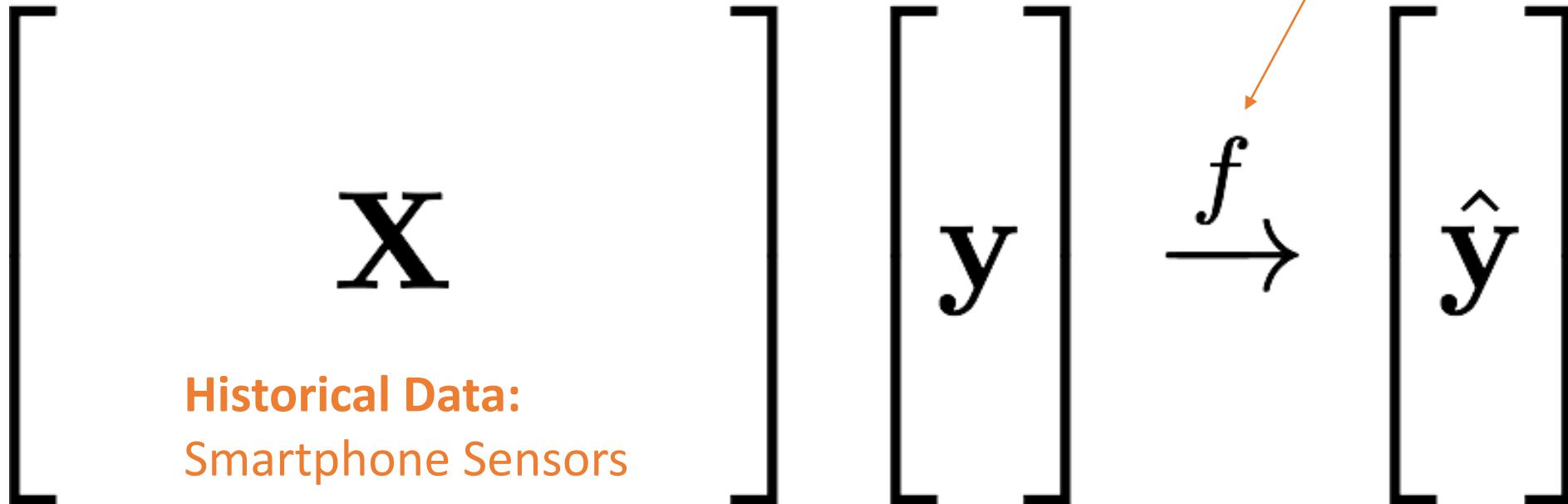
Data Set Characteristics:	Multivariate, Time-Series	Number of Instances:	10299	Area:	Computer
Attribute Characteristics:	N/A	Number of Attributes:	561	Date Donated:	2012-12-10
Associated Tasks:	Classification, Clustering	Missing Values?	N/A	Number of Web Hits:	505091

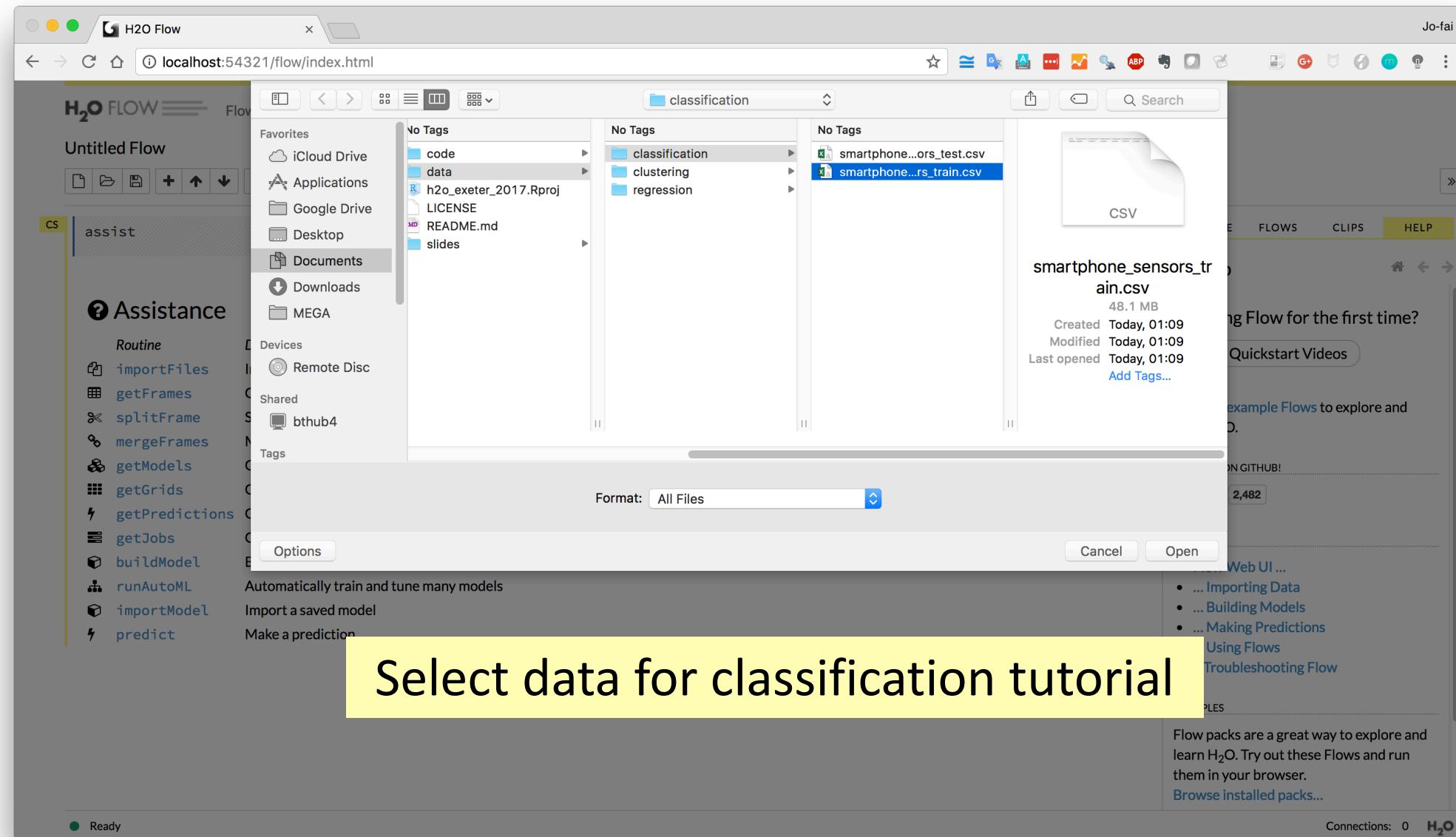
Source:

Jorge L. Reyes-Ortiz(1,2), Davide Anguita(1), Alessandro Ghio(1), Luca Oneto(1) and Xavier Parra(2)
 1 - Smartlab - Non-Linear Complex Systems Laboratory
 DITEN - Università degli Studi di Genova, Genoa (I-16145), Italy.
 2 - CETeD - Technical Research Centre for Dependency Care and Autonomous Living
 Universitat Politècnica de Catalunya (BarcelonaTech), Vilafranca i la Geltrú (08800), Spain
 activityrecognition @' smartlab.ws



Supervised Learning Example





Enum = Categorical Value (in Java)

The screenshot shows the H2O Flow web interface. In the 'PARSE CONFIGURATION' section, the 'Sources' field is set to 'smartphone_sensors_train.csv'. The 'ID' field is set to 'Key_Frame__smartphone_sensors_train.hex'. The 'Parser' dropdown is set to 'CSV'. The 'Separator' dropdown is set to ';' or '44'. The 'Column Headers' section has 'Auto' selected. Under 'First row contains', 'First row contains column names' is checked. Under 'Options', 'Delete on done' is checked. In the 'EDIT COLUMN NAMES AND TYPES' section, there is a table with 8 rows and 11 columns. The first column is labeled 'activity'. The second column is labeled 'f1_tBodyAccmeanX' and its type is set to 'Enum'. The other columns are labeled f2 through f7 and their types are set to 'Numeric'. An arrow points from the text 'Enum = Categorical Value (in Java)' to the 'Enum' dropdown in the table.

	activity	f1_tBodyAccmeanX	f2_tBodyAccmeanY	f3_tBodyAccmeanZ	f4_tBodyAccstdX	f5_tBodyAccstdY	f6_tBodyAccstdZ	f7_tBodyAccmadX	STANDING	STANDING	STANDING	STANDING	STANDING	STANDING	STANDING	STANDING	STANDING	STANDING	STANDING	STANDING
1	activity	Enum	Numeric	Numeric	Numeric	Numeric	Numeric	Numeric	0.28858451	0.27841883	0.27965306	0.27917394	0.27662877	0.27719877	0.27945388	0.27743247	0.27729342	0.27729342	0.27729342	
2	f1_tBodyAccmeanX								-0.020294171	-0.016410568	-0.019467156	-0.026200646	-0.016569655	-0.01009785	-0.019640776	-0.030488303	-0.021750698			
3	f2_tBodyAccmeanY								-0.13290514	-0.12352019	-0.11346169	-0.12328257	-0.11536185	-0.10513725	-0.11002215	-0.12536043	-0.12075082			
4	f3_tBodyAccmeanZ								-0.9952786	-0.99824528	-0.99537956	-0.99609149	-0.99813862	-0.99733496	-0.99692104	-0.99655926	-0.99732847			
5	f4_tBodyAccstdX								-0.98311061	-0.97530022	-0.96718701	-0.9834027	-0.98081727	-0.99048681	-0.96718593	-0.96672843	-0.96124532			
6	f5_tBodyAccstdY								-0.91352645	-0.96032199	-0.97894396	-0.9906751	-0.99048163	-0.99542003	-0.98311783	-0.98158533	-0.98367156			
7	f6_tBodyAccstdZ								-0.99511208	-0.99880719	-0.99651994	-0.99709947	-0.99832113	-0.9976274	-0.99700268	-0.99648525	-0.99759576			
8	f7_tBodyAccmadX																			

H2O Flow Jo-fai

localhost:54321/flow/index.html#

Untitled Flow

Key_Frame_smartphone_sensors_train.hex

Actions: View Data Split... Build Model... Predict Download Export Delete

	Rows	Columns	Compressed Size
	7352	562	25MB

▼ COLUMN SUMMARIES

label	type	Missing	Zeros	+Inf	-Inf	min	max	mean	sigma	cardinality	Actions
activity	enum	0	1407	0	0	0	5.0	.	.	6	Convert to numeric
f1_tBodyAccmeanX	real	0	0	0	0	-1.0	1.0	0.2745	0.0703	.	.
f2_tBodyAccmeanY	real	0	0	0	0	-1.0	1.0	-0.0177	0.0408	.	.
f3_tBodyAccmeanZ	real	0	0	0	0	-1.0	1.0	-0.1091	0.0566	.	.
f4_tBodyAccstdX	real	0	0	0	0	-1.0	1.0	-0.6054	0.4487	.	.
f5_tBodyAccstdY	real	0	0	0	0	-0.9999	0.9162	-0.5109	0.5026	.	.
f6_tBodyAccstdZ	real	0	0	0	0	-1.0	1.0	-0.6048	0.4187	.	.
f7_tBodyAccmadX	real	0	0	0	0	-1.0	1.0	-0.6305	0.4241	.	.
f8_tBodyAccmadY	real	0	0	0	0	-1.0	0.9677	-0.5269	0.4859	.	.
f9_tBodyAccmadZ	real	0	0	0	0	-1.0	1.0	-0.6062	0.4141	.	.
f10_tBodyAccmaxX	real	0	0	0	0	-1.0	1.0	-0.4686	0.5445	.	.
f11_tBodyAccmaxY	real	0	0	0	0	-1.0	1.0	-0.3060	0.2822	.	.
f12_tBodyAccmaxZ	real	0	0	0	0	-1.0	1.0	-0.5571	0.2939	.	.
f13_tBodyAccminX	real	0	0	0	0	-1.0	1.0	0.5236	0.3636	.	.
f14_tBodyAccminY	real	0	0	0	0	-1.0	1.0	0.3874	0.3436	.	.
f15_tBodyAccminZ	real	0	0	0	0	-1.0	1.0	0.5944	0.2978	.	.
f16_tBodyAccsma	real	0	0	0	0	-1.0	1.0	-0.5476	0.4718	.	.
f17_tBodyAccenergyX	real	0	0	0	0	-1.0	1.0	-0.8200	0.2596	.	.
f18_tBodyAccenergyY	real	0	0	0	0	-1.0	1.0	-0.9019	0.1263	.	.
f19_tBodyAccenergyZ	real	0	0	0	0	-1.0	1.0	-0.8458	0.2220	.	.

Ready Connections: 0 H2O

Exploratory analysis
(click on any variable)

H2O Flow Jo-fai

localhost:54321/flow/index.html#

Untitled Flow

Flow ▾ Cell ▾ Data ▾ Model ▾ Score ▾ Admin ▾ Help ▾

File ▾ + New ▾ Open ▾ Save ▾ Undo ▾ Redo ▾ Copy ▾ Paste ▾ Delete ▾ Find ▾ Help ▾

◀ Previous 20 Columns ▶ Next 20 Columns

▶ CHUNK COMPRESSION SUMMARY

▶ FRAME DISTRIBUTION SUMMARY

CS | getColumnSummary "Key_Frame__smartphone_sensors_train.hex", "activity"

133ms

Summary: activity

Actions Impute Inspect

CHARACTERISTICS

All

avg(percent)

DOMAIN (MAX 1000 LEVELS)

label

label	count
LAYING	~1300
STANDING	~1300
SITTING	~1200
WALKING	~1100
WALKING_UPSTAIRS	~1000
WALKING_DOWNSTAIRS	~1000

Connections: 0 H2O

H2O Flow Jo-fai

localhost:54321/flow/index.html#

H2O FLOW Flow Cell Data Model Score Admin Help

Untitled Flow

buildModel "drf"

Build a Model

Select an algorithm: Distributed Random Forest

PARAMETERS

model_id my_random_forest
training_frame Key_Frame_smartphone_sensors_train.hex
validation_frame (Choose...)
nfolds 0
response_column activity
ignored_columns Search...

GRID?

Destination id for this model; auto-generated if not specified.

Id of the training data frame (Not required, to allow initial validation of model parameters).

Id of the validation data frame.

Number of folds for N-fold cross-validation (0 to disable or >= 2).

Response variable column.

Showing page 1 of 6.

activity	ENUM(6)
f1_tBodyAccmeanX	REAL
f2_tBodyAccmeanY	REAL
f3_tBodyAccmeanZ	REAL
f4_tBodyAccstdX	REAL
f5_tBodyAccstdY	REAL

Connections: 0 H2O

Build Random Forest Model

1) Select "..._train" dataset

2) Select "activity" as response

H2O Flow Jo-fai

localhost:54321/flow/index.html#

Untitled Flow

Model

Model ID: my_random_forest
Algorithm: Distributed Random Forest

Actions: Refresh, Predict..., Download POJO, Download Model Deployment Package (MOJO), Export, Inspect, Delete, Download Gen Model

MODEL PARAMETERS

SCORING HISTORY - LOGLOSS

The plot shows the relationship between the number of trees (x-axis, ranging from 0 to 50) and training logloss (y-axis, ranging from 0.0 to 4.0). The data points show a clear downward trend, indicating that the model's performance is improving as more trees are added.

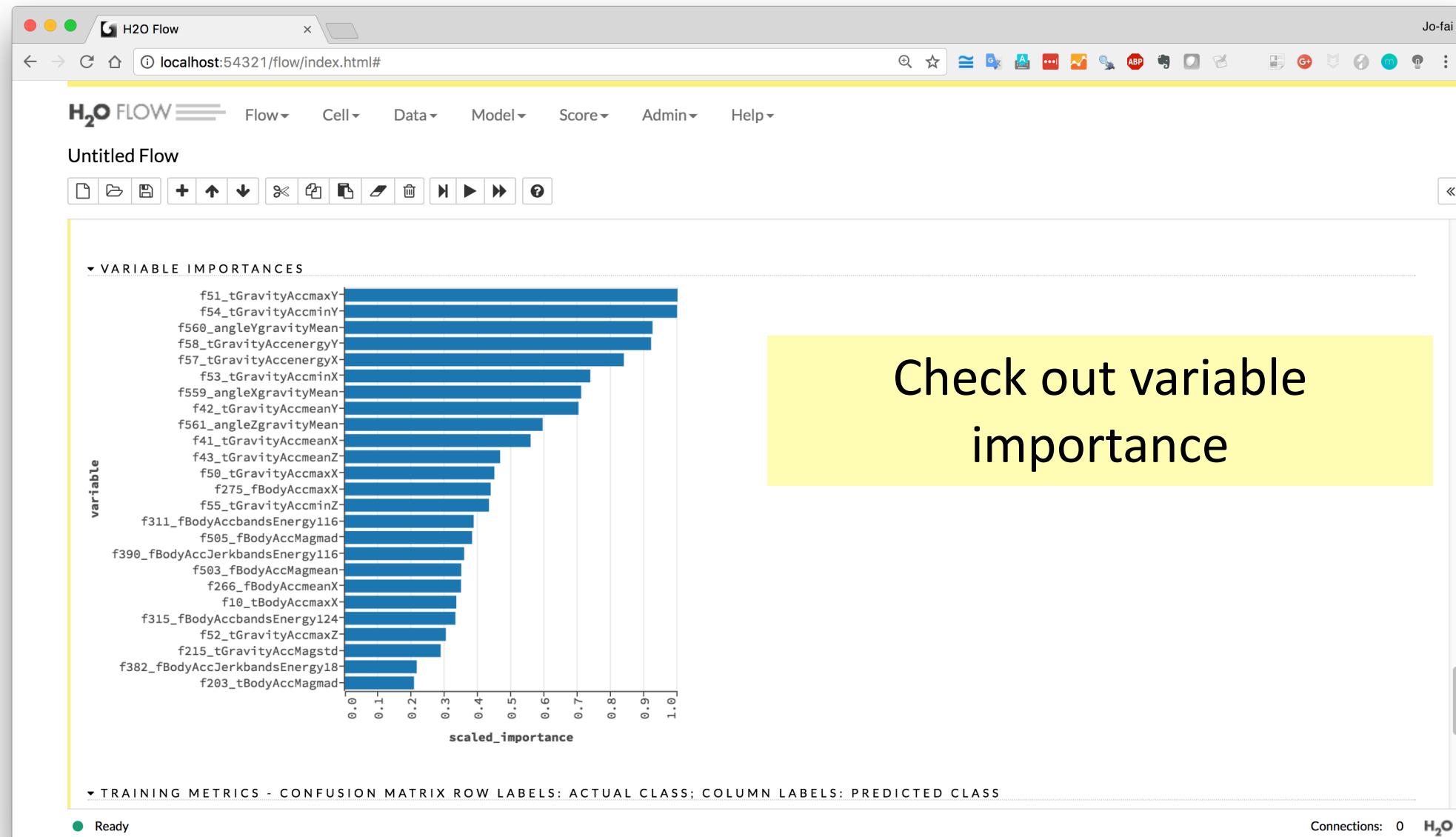
number_of_trees	training_logloss
0	3.5
5	2.0
10	0.8
15	0.4
20	0.2
25	0.1
30	0.08
35	0.06
40	0.04
45	0.03
50	0.02

VARIABLE IMPORTANCES

Ready

Connections: 0 H2O

Performance metric
Logarithmic Loss (logloss)
A loss function for classification



H2O Flow Jo-fai

localhost:54321/flow/index.html#

H2O FLOW Flow ▾ Cell ▾ Data ▾ Model ▾ Score ▾ Admin ▾ Help ▾

Untitled Flow

File ▾ + New ▾ Save ▾ Undo ▾ Redo ▾ Cut ▾ Copy ▾ Paste ▾ Delete ▾ Run ▾ Stop ▾ Help ▾

◀ ▶

▼ TRAINING METRICS - CONFUSION MATRIX ROW LABELS: ACTUAL CLASS; COLUMN LABELS: PREDICTED CLASS

	LAYING	SITTING	STANDING	WALKING	WALKING_DOWNSTAIRS	WALKING_UPSTAIRS	Error	Rate
LAYING	1407	0	0	0	0	0	0	0 / 1,407
SITTING	0	1216	69	0	0	1	0.0544	70 / 1,286
STANDING	0	43	1331	0	0	0	0.0313	43 / 1,374
WALKING	0	0	2	1213	7	4	0.0106	13 / 1,226
WALKING_DOWNSTAIRS	0	1	0	5	970	10	0.0162	16 / 986
WALKING_UPSTAIRS	0	0	1	1	5	1066	0.0065	7 / 1,073
Total	1407	1260	1403	1219	982	1081	0.0203	149 / 7,352

▶ OUTPUT

▶ OUTPUT - MODEL SUMMARY

▶ OUTPUT - SCORING HISTORY

▶ OUTPUT - TRAINING_METRICS

▶ OUTPUT - TRAINING_METRICS - TOP-6 HIT RATIOS

▶ OUTPUT - VARIABLE IMPORTANCES

▼ PREVIEW POJO

● Ready

Connections: 0 H2O

Confusion Matrix
(Yellow cells = correct predictions)

H2O Flow Jo-fai

localhost:54321/flow/index.html#

H2O FLOW Flow Cell Data Model Score Admin Help

Untitled Flow

OUTPUT - TRAINING_METRICS - TOP-6 HIT RATIOS

OUTPUT - VARIABLE IMPORTANCES

PREVIEW POJO

Preview POJO

Make predictions on test set

CS predict 41ms

Predict

Name: my_predictions

Model: my_random_forest

Frame: Key_Frame_smartphone_sensors_test.hex

Actions: ⚡ Predict

Ready Connections: 0 H2O

The screenshot shows the H2O Flow web application. At the top, there's a navigation bar with tabs for Flow, Cell, Data, Model, Score, Admin, and Help. Below the navigation is a toolbar with icons for file operations (New, Open, Save, etc.) and process management (Run, Stop, etc.). A large yellow callout box on the right side contains the text "Make predictions on test set". In the main workspace, there's a step labeled "predict" under the "CS" category. This step has several configuration fields: "Name: my_predictions", "Model: my_random_forest", "Frame: Key_Frame_smartphone_sensors_test.hex", and an "Actions" section containing a button labeled "⚡ Predict". The status bar at the bottom indicates the step is "Ready" and shows "Connections: 0" along with the H2O logo.

H2O Flow Jo-fai

localhost:54321/flow/index.html#

Compare Results

Untitled Flow

combined-my_predictions

Predictions (with prob. of each class)

◀ Previous 20 Columns ▶ Next 20 Columns

Row	predict	LAYING	SITTING	STANDING	WALKING	WALKING_DOWNSTAIRS	WALKING_UPSTAIRS	activity	f1_tBodyAccmeanX	f2_tBodyAccmeanY	f3_tBodyAccmeanZ	f4_t
1	STANDING	0.0401	0.0401	0.9199	0	0	0	STANDING	0.2572	-0.0233	-0.0147	
2	STANDING	0	0.1698	0.8302	0	0	0	STANDING	0.2860	-0.0132	-0.1191	
3	STANDING	0.0182	0.2000	0.7818	0	0	0	STANDING	0.2755	-0.0261	-0.1182	
4	STANDING	0.0408	0.0816	0.8776	0	0	0	STANDING	0.2703	-0.0326	-0.1175	
5	STANDING	0	0.0579	0.9421	0	0	0	STANDING	0.2748	-0.0278	-0.1295	
6	STANDING	0	0.0394	0.9606	0	0	0	STANDING	0.2792	-0.0186	-0.1139	
7	STANDING	0	0.0965	0.9035	0	0	0	STANDING	0.2797	-0.0183	-0.1040	
8	STANDING	0	0.0446	0.9554	0	0	0	STANDING	0.2746	-0.0250	-0.1168	
9	STANDING	0	0.1429	0.8571	0	0	0	STANDING	0.2725	-0.0210	-0.1145	
10	STANDING	0	0.1569	0.8431	0	0	0	STANDING	0.2757	-0.0104	-0.0998	
11	STANDING	0	0.1667	0.8333	0	0	0	STANDING	0.2786	-0.0152	-0.0989	
12	STANDING	0.0179	0.1607	0.8214	0	0	0	STANDING	0.2792	-0.0219	-0.1097	
13	STANDING	0	0	1.0	0	0	0	STANDING	0.2745	-0.0231	-0.1125	
14	STANDING	0	0.1325	0.8675	0	0	0	STANDING	0.2691	-0.0277	-0.1102	
15	STANDING	0	0.1115	0.8885	0	0	0	STANDING	0.2756	-0.0189	-0.0974	

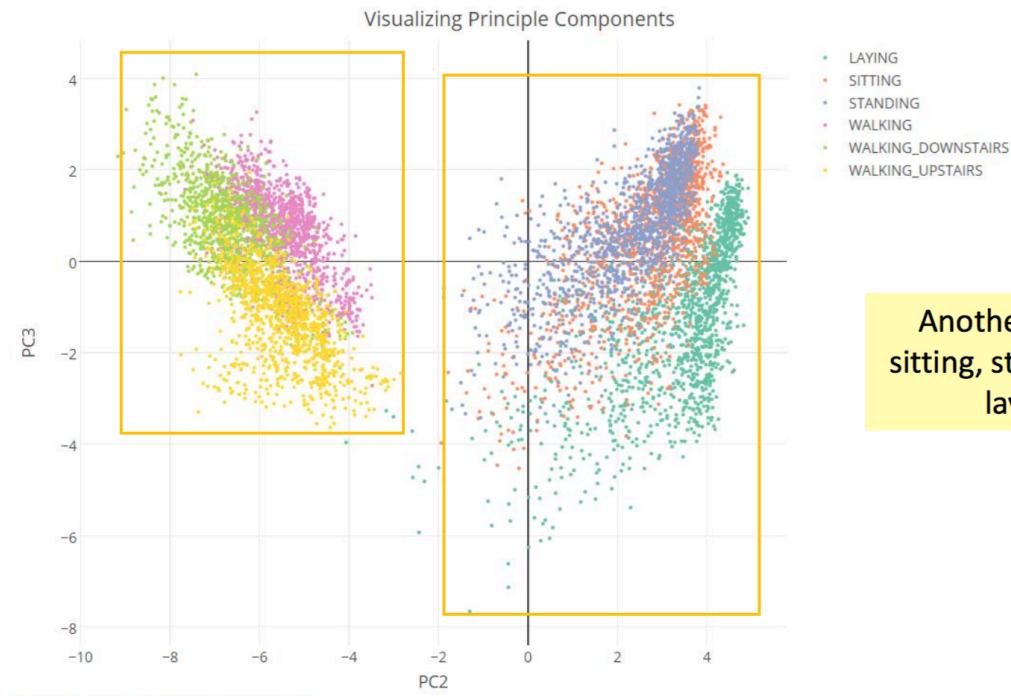
● Ready

Ground Truth

Connections: 0 H2O

H₂O Demo at IBM Conference

First, we can see two major clusters. One for movement (walking, walking upstairs and walking downstairs)



From the graph above, we can see that:

- it could be difficult to distinguish between **Standing** and **Sitting** as there are large overlaps in their sensor data.
- Laying has its own cluster so it should be easy to classify.
- Walking, **Walking Upstairs** and **Walking Downstairs** are understandably closer to each other yet they are quite different to **Sitting**, **Standing** and **Laying**.

Another one for sitting, standing and laying.

IBM Fast Track Your Data (ML Conference)
Munich



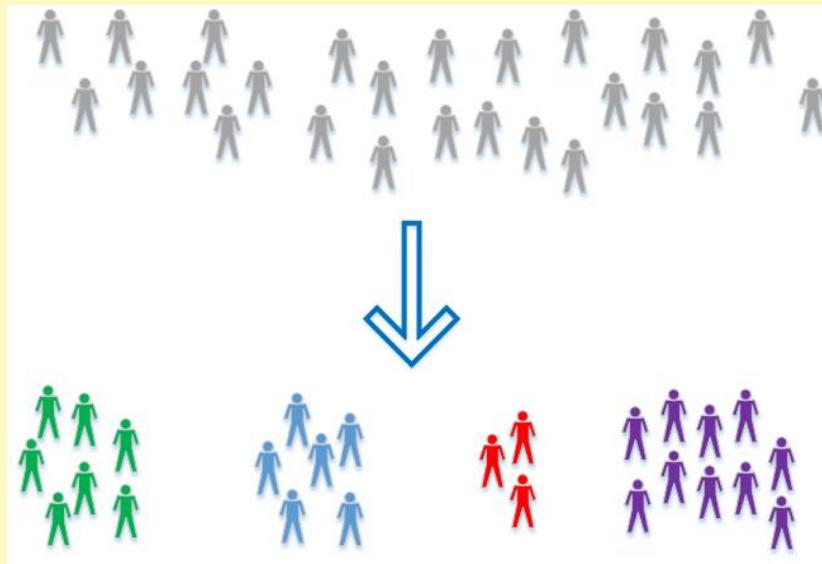
https://github.com/woobe/h2o_demo_for_ibm_dsx

Tutorial 3 - Clustering

- **Data:** Water Treatment Plant (1993)
- **Source:** <https://archive.ics.uci.edu/ml/datasets/water+treatment+plant>

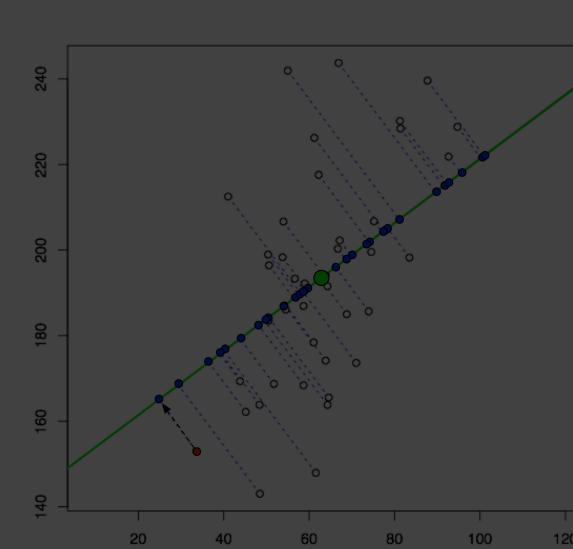
Unsupervised Learning – Discover Hidden Patterns

Clustering:
Customer Segmentation



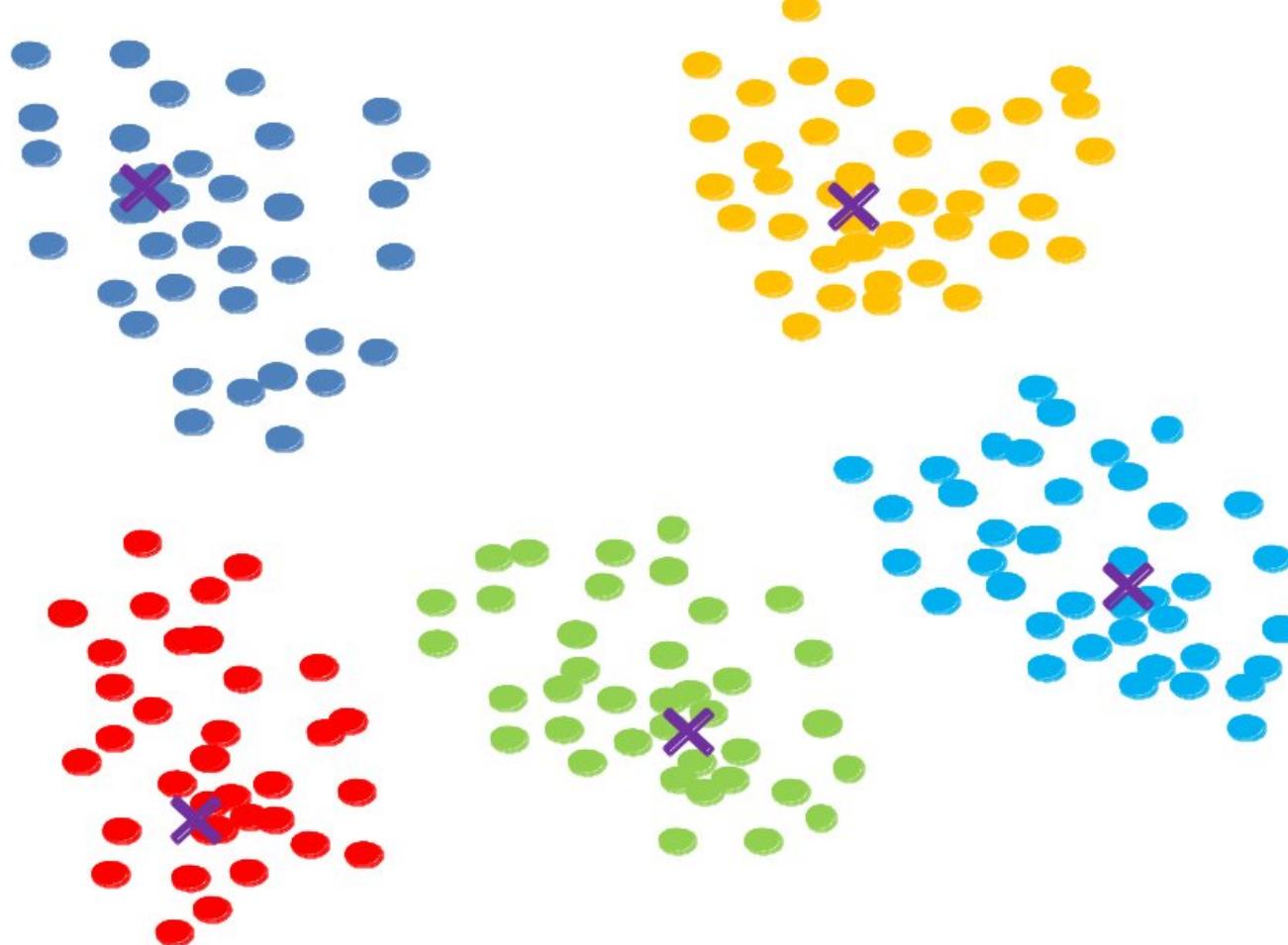
H₂O algos:
K-Means
Generalised Low Rank Model

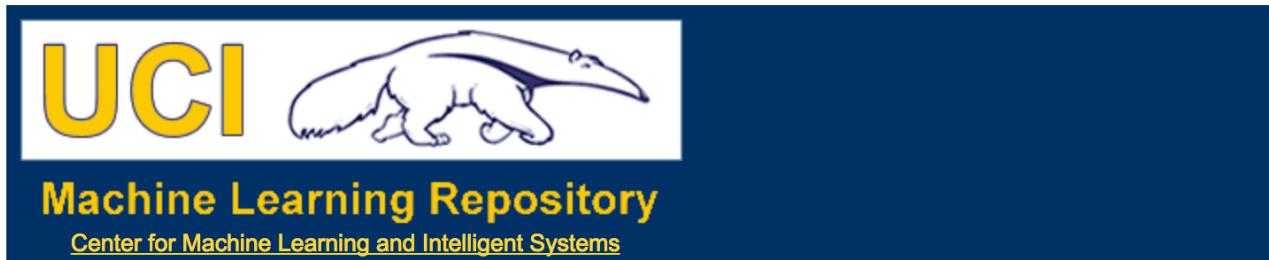
Dimensionality Reduction:
Linear Transformation of Variables



H₂O algos:
Principal Component Analysis
Generalised Low Rank Model

K-Means Clustering





Machine Learning Repository
Center for Machine Learning and Intelligent Systems

Water Treatment Plant Data Set

Download: [Data Folder](#), [Data Set Description](#)

Abstract: Multiple classes predict plant state

Data Set Characteristics:	Multivariate	Number of Instances:	527	Area:	Physical
Attribute Characteristics:	Integer, Real	Number of Attributes:	38	Date Donated	1993-06-01
Associated Tasks:	Clustering	Missing Values?	N/A	Number of Web Hits:	91704

Source:

Creators:

Manel Poch (igte2 '@' cc.uab.es)

Unitat d'Enginyeria Química

Universitat Autònoma de Barcelona. Bellaterra. Barcelona; Spain

Donor:

Javier Bejar and Ulises Cortes (bejar '@' lsi.upc.es)

Dept. Llenguatges i Sistemes Informàtics;

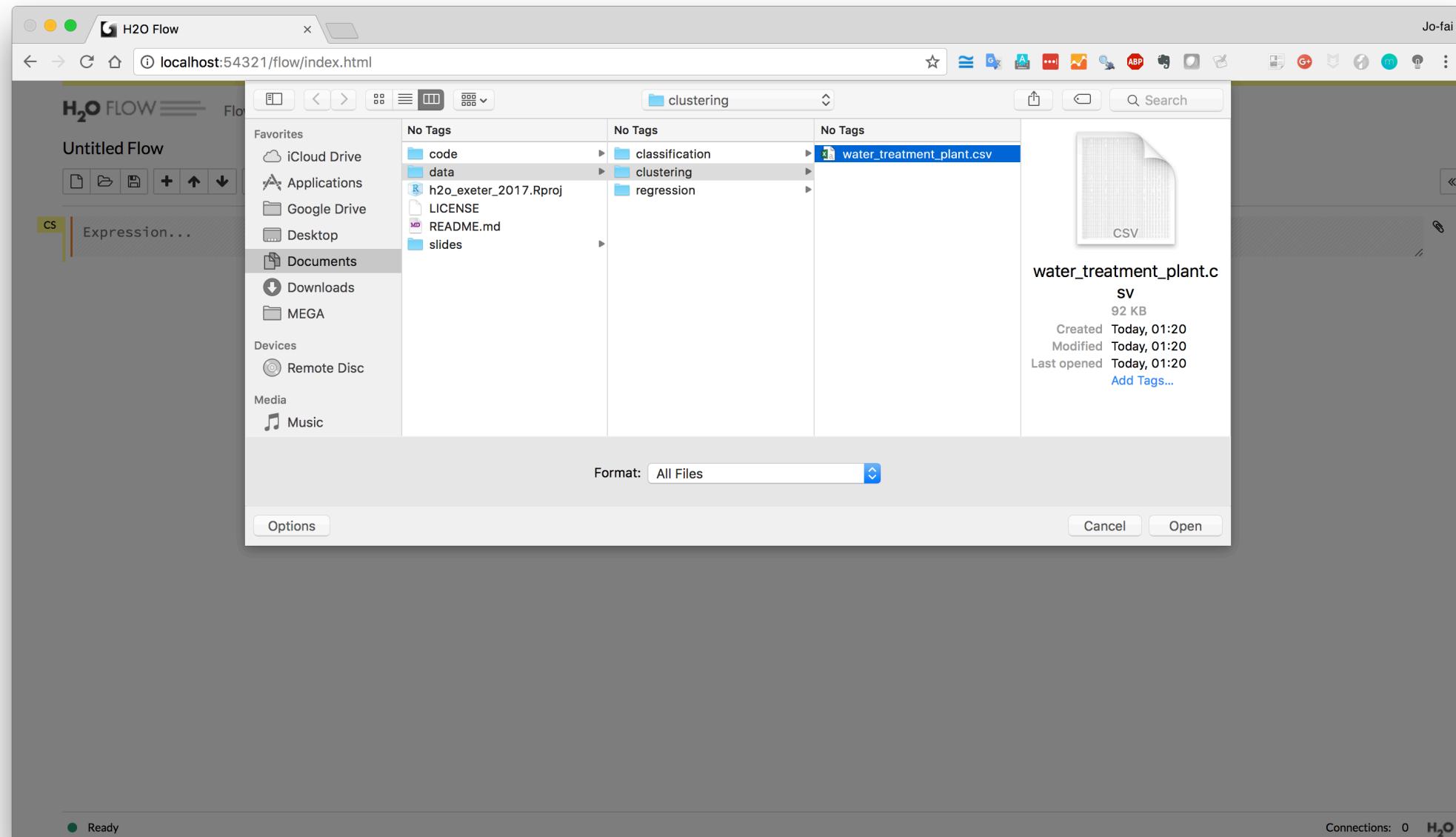
Universitat Politècnica de Catalunya. Barcelona; Spain

Attribute Information:

All attributes are numeric and continuous

N. Attrib.

- 1 Q-E (input flow to plant)
- 2 ZN-E (input Zinc to plant)
- 3 PH-E (input pH to plant)
- 4 DBO-E (input Biological demand of oxygen to plant)
- 5 DQO-E (input chemical demand of oxygen to plant)
- 6 SS-E (input suspended solids to plant)
- 7 SSV-E (input volatile suspended solids to plant)
- 8 SED-E (input sediments to plant)
- 9 COND-E (input conductivity to plant)
- 10 PH-P (input pH to primary settler)
- 11 DBO-P (input Biological demand of oxygen to primary settler)
- 12 SS-P (input suspended solids to primary settler)
- 13 SSV-P (input volatile suspended solids to primary settler)
- 14 SED-P (input sediments to primary settler)
- 15 COND-P (input conductivity to primary settler)
- 16 PH-D (input pH to secondary settler)
- 17 DBO-D (input Biological demand of oxygen to secondary settler)
- 18 DQO-D (input chemical demand of oxygen to secondary settler)
- 19 SS-D (input suspended solids to secondary settler)
- 20 SSV-D (input volatile suspended solids to secondary settler)
- 21 SED-D (input sediments to secondary settler)
- 22 COND-D (input conductivity to secondary settler)
- 23 PH-S (output pH)
- 24 DBO-S (output Biological demand of oxygen)
- 25 DQO-S (output chemical demand of oxygen)
- 26 SS-S (output suspended solids)
- 27 SSV-S (output volatile suspended solids)
- 28 SED-S (output sediments)
- 29 COND-S (output conductivity)
- 30 RD-DBO-P (performance input Biological demand of oxygen in primary settler)
- 31 RD-SS-P (performance input suspended solids to primary settler)
- 32 RD-SED-P (performance input sediments to primary settler)
- 33 RD-DBO-S (performance input Biological demand of oxygen to secondary settler)
- 34 RD-DQO-S (performance input chemical demand of oxygen to secondary settler)
- 35 RD-DBO-G (global performance input Biological demand of oxygen)
- 36 RD-DQO-G (global performance input chemical demand of oxygen)
- 37 RD-SS-G (global performance input suspended solids)
- 38 RD-SED-G (global performance input sediments)



H2O Flow Jo-fai

localhost:54321/flow/index.html

Untitled Flow

Setup Parse

PARSE CONFIGURATION

Sources: water_treatment_plant.csv
 ID: Key_Frame_water_treatment_plant.hex
 Parser: CSV
 Separator: ;'44'
 Column Headers: Auto
 First row contains column names
 First row contains data
 Options: Enable single quotes as a field quotation character
 Delete on done

EDIT COLUMN NAMES AND TYPES

Search by column name...

1	name	String	D-1/3/90	D-2/3/90	D-4/3/90	D-5/3/90	D-6/3/90	D-7/3/90	D-8/3/90	D-9/3/90	D-11/3/90
2	Q-E	Numeric	44101	39024	32229	35023	36924	38572	41115	36107	29156
3	ZN-E	Numeric	1.50	3.00	5.00	3.50	1.50	3.00	6.00	5.00	2.50
4	PH-E	Numeric	7.8	7.7	7.6	7.9	8	7.8	7.8	7.7	7.7
5	DBO-E	Numeric	?	?	?	205	242	202	?	215	206
6	DQO-E	Numeric	407	443	528	588	496	372	552	489	451
7	SS-E	Numeric	166	214	186	192	176	186	262	334	194
8	SSV-E	Numeric	66.3	69.2	69.9	65.6	64.8	68.8	64.1	40.7	69.1

Ready Connections: 0 H2O

Model → K-means

The screenshot shows the H2O Flow interface on a Mac OS X desktop. The window title is "H2O Flow". The main menu bar includes "File", "Flow", "Cell", "Data", "Model" (which is highlighted in yellow), "Score", "Admin", and "Help". Below the menu is a toolbar with icons for file operations like Open, Save, and Print, along with icons for Split, Build Model, and others.

The main workspace displays a table of column summaries for a dataset. The columns include "label", "type", "Missing", "Zeros", "+Inf", "-Inf", and "mean". The rows list various feature names: name, Q-E, ZN-E, PH-E, DBO-E, DQO-E, SS-E, SSV-E, SED-E, COND-E, PH-P, DBO-P, SS-P, SSV-P, SED-P, COND-P, PH-D, DBO-D, DQO-D, and SS-D. The "type" column shows values like string, int, real, and double.

A context menu is open over the table, with "K-means..." selected. Other options in the menu include Aggregator..., Deep Learning..., Distributed Random Forest..., Gradient Boosting Machine..., Generalized Linear Modeling..., Generalized Low Rank Modeling..., Naive Bayes..., Principal Components Analysis..., Stacked Ensemble..., Word2Vec..., and XGBoost... . The menu also contains sections for "Cardinality Actions" (with Convert to enum options) and "Actions" (List All Models, List Grid Search Results, Import Model..., Export Model..., Run AutoML...).

The status bar at the bottom shows the URL "localhost:54321/flow/index.html#" and connection information "Connections: 0".

H2O Flow UCI Machine Learning Repository Jo-fai

localhost:54321/flow/index.html

H2O FLOW

Flow ▾ Cell ▾ Data ▾ Model ▾ Score ▾ Admin ▾ Help ▾

Untitled Flow

Build a Model

Select an algorithm: K-means

PARAMETERS

model_id my_kmeans Destination id for this model; auto-generated if not specified.

training_frame Key_Frame_water_treatment_plant.hex Id of the training data frame (Not required, to allow initial validation of model parameters).

validation_frame (Choose...) Id of the validation data frame.

nfold 0 Number of folds for N-fold cross-validation (0 to disable or >= 2).

ignored_columns Search... Showing page 1 of 1. 1 ignored.

<input checked="" type="checkbox"/>	name	STRING
<input type="checkbox"/>	Q-E	INT
<input type="checkbox"/>	ZN-E	REAL
<input type="checkbox"/>	PH-E	REAL
<input type="checkbox"/>	DBO-E	INT
<input type="checkbox"/>	DQO-E	INT

Connections: 0 H2O

1) Enter $k = 10$
2) Choose “estimate_k”
3) max_iterations = 100

The max. number of clusters. If estimate_k is disabled, the model will find k centroids, otherwise it will find up to k centroids.

Whether to estimate the number of clusters ($\leq k$) iteratively and deterministically.

Maximum training iterations (if estimate_k is enabled, then this is for each inner Lloyds iteration)

Standardize columns before computing distances

Initialization mode

Column with cross-validation fold index assignment per observation.

Whether to score during each iteration of model training.

RNG Seed

H2O Flow UCI Machine Learning Repository Jo-fai

localhost:54321/flow/index.html

Untitled Flow

Model

Model ID: my_kmeans
Algorithm: K-means

Actions: Refresh, Predict..., Download POJO, Download Model Deployment Package (MOJO), Export, Inspect, Delete, Download Gen Model

► MODEL PARAMETERS

► SCORING HISTORY

► OUTPUT

► OUTPUT - MODEL SUMMARY

► OUTPUT - SCORING HISTORY

► OUTPUT - TRAINING_METRICS

► OUTPUT - TRAINING_METRICS - CENTROID STATISTICS

► OUTPUT - CLUSTER MEANS

centroid	qe	zne	phe	dboe	dqoe	sse	ssve	sede	conde	php	dbop	ssp	ssvp	sedp	condp	phd	dbod	dqod	ssd	ssvd	se	
1	38115.3858	2.2676	7.7007	159.6578	340.9922	214.3935	57.6919	3.7877	1293.8806	7.7250	168.6287	234.5485	57.0164	3.9748	1305.7052	7.7243	101.1276	228.7470	85.8673	71.2329	0.30	
2	36255.1751	2.4893	7.9333	219.5125	477.4861	241.6190	65.3837	5.4669	1680.3095	7.9456	246.1699	275.8571	63.9487	6.1777	1704.9881	7.9119	144.7682	321.1317	102.7619	74.9277	0.52	
3	38167.6719	1.1714	7.5571	192.4286		389.0	216.8571	59.4429	4.0286	1290.7143	7.6857	206.2857	208.2857	59.9571	4.3857	1260.5714	7.5714	127.7143	313.2923	106.8571	68.8857	0.72

► OUTPUT - STANDARDIZED CLUSTER MEANS

► PREVIEW POJO

</> Preview POJO

Ready Connections: 0 H2O

H₂O finds optimal k = 3

H2O Flow UCI Machine Learning Repository Jo-fai

localhost:54321/flow/index.html

H2O FLOW Flow ▾ Cell ▾ Data ▾ Model ▾ Score ▾ Admin ▾ Help ▾

Untitled Flow

combined-my_clusters

DATA

Previous 20 Columns Next 20 Columns

Row	predict	name	Q-E	ZN-E	PH-E	DBO-E	DQO-E	SS-E	SSV-E	SED-E	COND-E	PH-P	DBO-P	SS-P	SSV-P	SED-P	COND-P	PH-D	DBO-D	DQO-D
1	1.0	0-1/3/90	44101.0	1.5000	7.8000	.	407.0	166.0	66.3000	4.5000	2110.0	7.9000	.	228.0	70.2000	5.5000	2120.0	7.9000	.	280.0
2	1.0	0-2/3/90	39024.0	3.0	7.7000	.	443.0	214.0	69.2000	6.5000	2660.0	7.7000	.	244.0	75.4000	7.7000	2570.0	7.6000	.	474.0
3	1.0	0-4/3/90	32229.0	5.0	7.6000	.	528.0	186.0	69.9000	3.4000	1666.0	7.7000	.	220.0	72.7000	4.5000	1594.0	7.7000	.	272.0
4	1.0	0-5/3/90	35023.0	3.5000	7.9000	205.0	588.0	192.0	65.6000	4.5000	2430.0	7.8000	236.0	268.0	73.1000	8.5000	2280.0	7.8000	158.0	376.0
5	1.0	0-6/3/90	36924.0	1.5000	8.0	242.0	496.0	176.0	64.8000	4.0	2110.0	7.9000	.	236.0	57.6000	4.5000	2020.0	7.8000	.	372.0
6	1.0	0-7/3/90	38572.0	3.0	7.8000	202.0	372.0	186.0	68.8000	4.5000	1644.0	7.8000	.	248.0	66.1000	8.5000	1762.0	7.7000	150.0	460.0
7	1.0	0-8/3/90	41115.0	6.0	7.8000	.	552.0	262.0	64.1000	5.0	1603.0	7.8000	.	320.0	67.5000	6.5000	1608.0	7.8000	192.0	376.0
8	1.0	0-9/3/90	36107.0	5.0	7.7000	215.0	489.0	334.0	40.7000	6.0	1613.0	7.6000	.	304.0	53.9000	8.0	1557.0	7.6000	181.0	350.0
9	0	0-11/3/90	29156.0	2.5000	7.7000	206.0	451.0	194.0	69.1000	4.5000	1249.0	7.7000	206.0	220.0	61.8000	4.0	1219.0	7.7000	111.0	282.0
10	1.0	0-12/3/90	39246.0	2.0	7.8000	172.0	506.0	200.0	69.0	5.0	1865.0	7.8000	208.0	248.0	66.1000	6.5000	1929.0	7.8000	164.0	463.0
11	2.0	0-13/3/90	42393.0	0.7000	7.9000	189.0	478.0	230.0	67.0	5.5000	1410.0	8.1000	173.0	192.0	62.5000	5.0	1406.0	7.7000	172.0	412.0
12	2.0	0-14/3/90	42857.0	1.5000	7.7000	238.0	319.0	292.0	33.8000	3.5000	1261.0	7.6000	170.0	268.0	31.3000	4.2000	1204.0	7.6000	116.0	276.0
13	2.0	0-15/3/90	42911.0	0.7000	7.6000	114.0	252.0	116.0	58.6000	1.2000	1238.0	7.9000	148.0	136.0	64.7000	3.0	1208.0	7.7000	79.0	216.0
14	1.0	0-16/3/90	40376.0	.	8.1000	204.0	333.0	174.0	67.8000	3.0	2390.0	7.8000	231.0	156.0	74.4000	2.5000	2540.0	7.8000	136.0	325.0
15	0	0-18/3/90	40923.0	3.5000	7.6000	146.0	329.0	188.0	57.4000	2.5000	1300.0	7.6000	162.0	132.0	63.6000	2.0	1324.0	7.6000	109.0	243.0
16	0	0-19/3/90	43830.0	1.5000	7.8000	177.0	512.0	214.0	58.9000	5.5000	1605.0	7.7000	164.0	256.0	71.9000	5.5000	1599.0	7.7000	118.0	320.0
17	0	0-20/3/90	39165.0	1.2000	7.4000	250.0	447.0	252.0	61.1000	7.0	1533.0	7.4000	275.0	216.0	57.4000	6.5000	1501.0	7.4000	138.0	269.0
18	1.0	0-21/3/90	35791.0	1.2000	7.8000	277.0	466.0	246.0	63.4000	4.0	1556.0	7.7000	.	288.0	65.3000	6.0	1846.0	7.7000	166.0	419.0
19	1.0	0-22/3/90	37419.0	1.2000	7.6000	219.0	446.0	222.0	61.3000	5.5000	1600.0	7.7000	266.0	240.0	70.0	5.0	1645.0	7.6000	172.0	345.0
20	0	0-23/3/90	40983.0	3.0	7.6000	182.0	431.0	214.0	57.0	7.0	1591.0	7.5000	219.0	248.0	58.1000	5.5000	1473.0	7.5000	175.0	376.0

Ready Connections: 0 H2O

Remember ...

- All models are wrong but some are useful.

More About H₂O

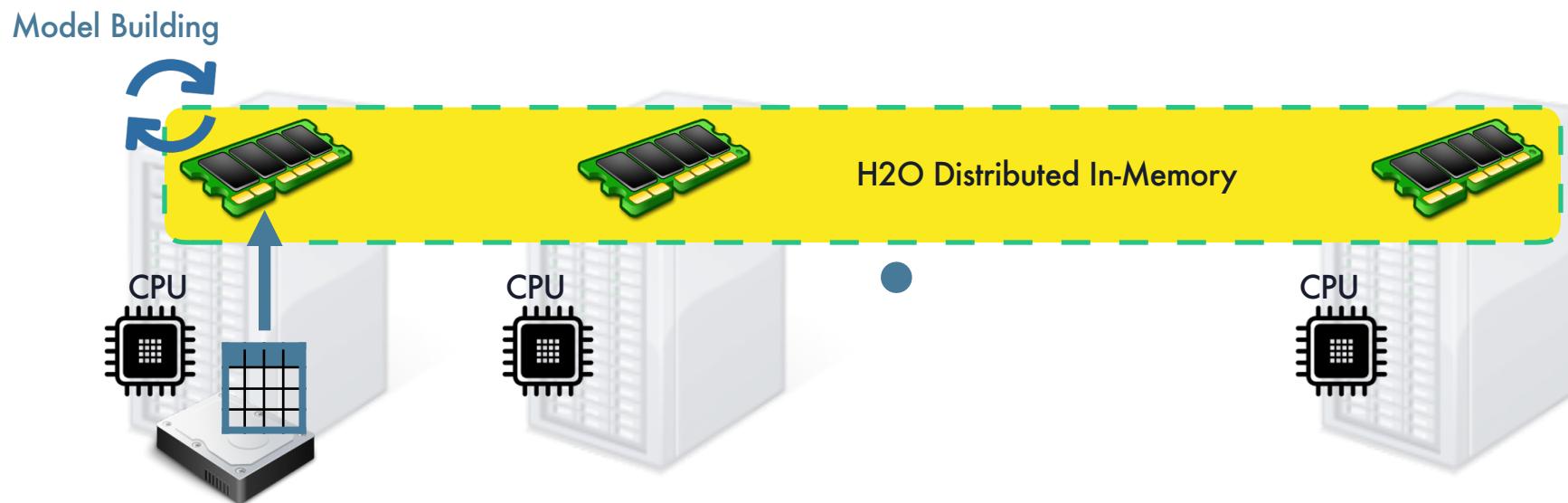
- Running H₂O on Multi-Node Cluster
- Next-Gen H₂O Products

H₂O on Multi-Node Cluster

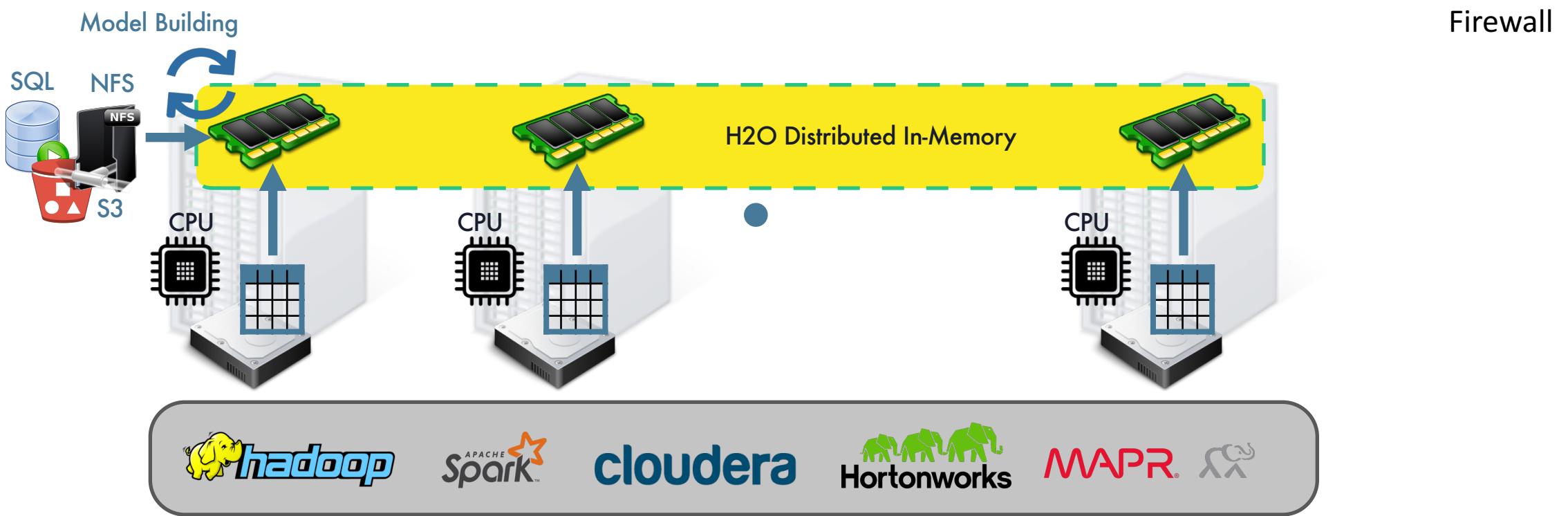
H₂O on Single Machine



H₂O on Multi-Node Cluster



H₂O with Distributed Data Storage Systems





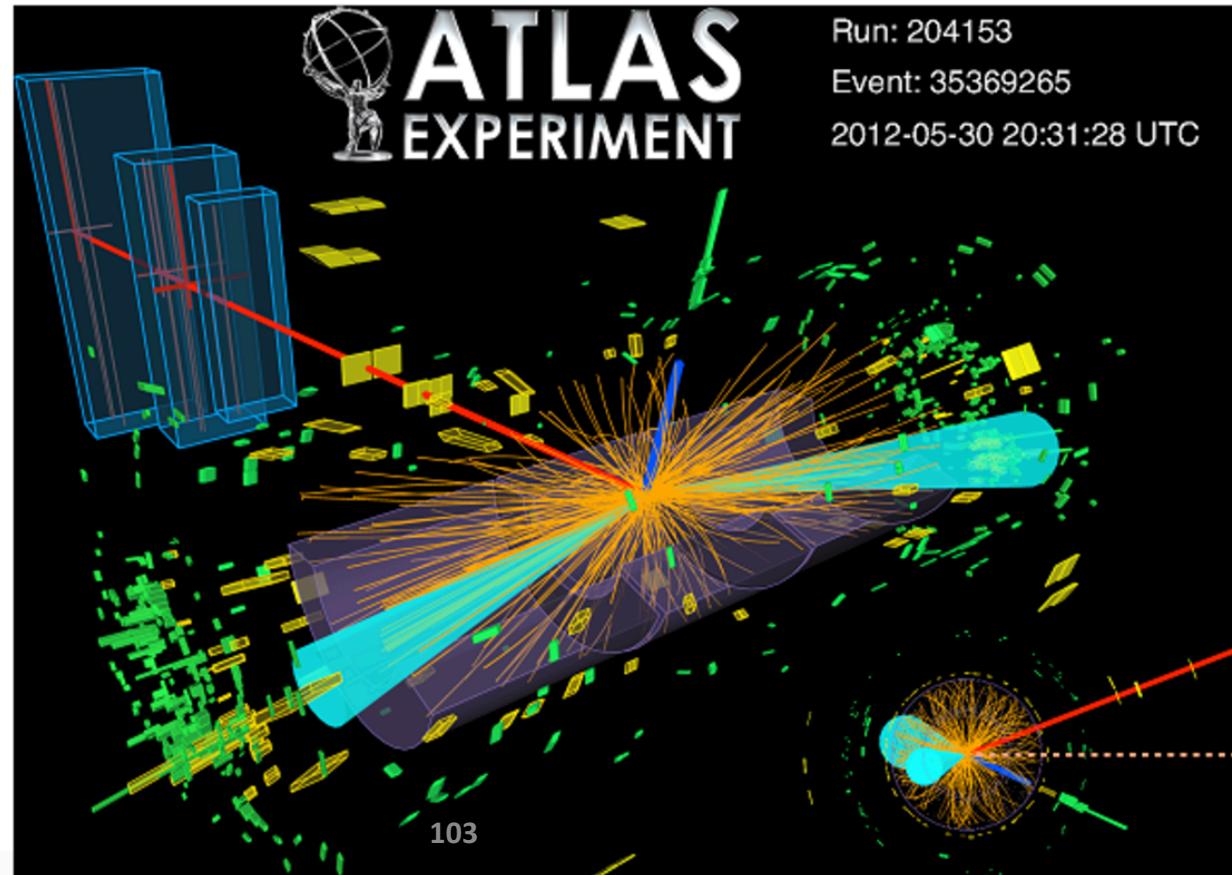
Higgs Boson Machine Learning Challenge

Use the ATLAS experiment to identify the Higgs boson

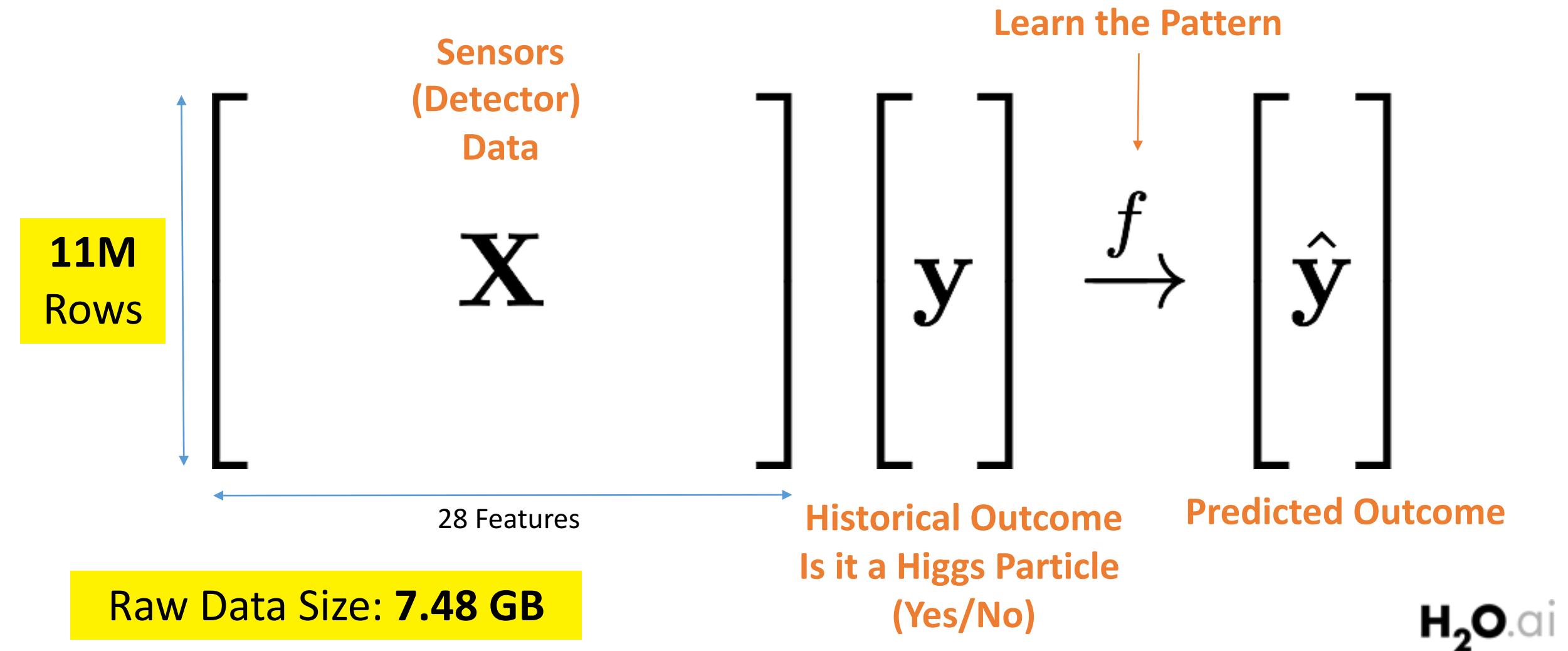
\$13,000 · 1,785 teams · 3 years ago

[Overview](#)[Data](#)[Discussion](#)[Leaderboard](#)[Rules](#)[Team](#)[My Submissions](#)[Late Submission](#)[Overview](#)

<https://www.kaggle.com/c/higgs-boson>

[Description](#)[Evaluation](#)[Prizes](#)[About The Sponsors](#)[Timeline](#)[Winners](#)

Learning from Higgs Boson Dataset



11M Rows**Size (Raw): 7.48 GB****Compressed: 2.00 GB (\approx 27% of Raw)**

HIGGS.hex

Actions:

[View Data](#)[Split...](#)[Build Model...](#)[Predict](#)[Download](#)[Export](#)

Rows	Columns	Compressed Size
11000000	29	2GB

▼ COLUMN SUMMARIES

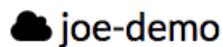
label	type	Missing	Zeros	+Inf	-Inf	min	max	mean	sigma	cardinality	Actions
C1	enum	0	5170877	0	0	0	1.0	0.5299	0.4991	2	Convert to numeric
C2	real	0	0	0	0	0.2747	12.0989	0.9915	0.5654	· ·	
C3	real	0	0	0	0	-2.4350	2.4349	-0.0	1.0088	· ·	
C4	real	0	0	0	0	-1.7425	1.7432	-0.0	1.0063	· ·	
C5	real	0	0	0	0	0.0002	15.3968	0.9985	0.6000	· ·	
C6	real	0	0	0	0	-1.7439	1.7433	0.0	1.0063	· ·	
C7	real	0	0	0	0	0.1375	9.9404	0.9909	0.4750	· ·	
C8	real	0	0	0	0	-2.9697	2.9697	-0.0	1.0093	· ·	
C9	real	0	0	0	0	-1.7412	1.7415	0.0	1.0059	· ·	
C10	real	0	5394611	0	0	0	2.1731	1.0	1.0278	· ·	
C11	real	0	0	0	0	0.1890	11.6471	0.9927	0.5000	· ·	
C12	real	0	0	0	0	-2.9131	2.9132	-0.0	1.0093	· ·	
C13	real	0	0	0	0	-1.7424	1.7432	-0.0	1.0062	· ·	
C14	real	0	5523912	0	0	0	2.2149	1.0	1.0494	· ·	
C15	real	0	0	0	0	0.2636	14.7090	0.9923	0.4877	· ·	
C16	real	0	0	0	0	-2.7297	2.7300	0.0	1.0087	· ·	
C17	real	0	0	0	0	-1.7421	1.7429	0.0	1.0063	· ·	
C18	real	0	6265240	0	0	0	2.5482	1.0	1.1937	· ·	
C19	real	0	0	0	0	0.3654	12.8826	0.9861	0.5058	· ·	
C20	real	0	0	0	0	-2.4973	2.4980	-0.0	1.0077	· ·	

Untitled Flow



CS

getCloud



CLOUD STATUS

HEALTHY	CONSENSUS	LOCKED
Version	Started	Nodes (Used / All)
3.13.0.3981	a minute ago	10 / 10

NODES

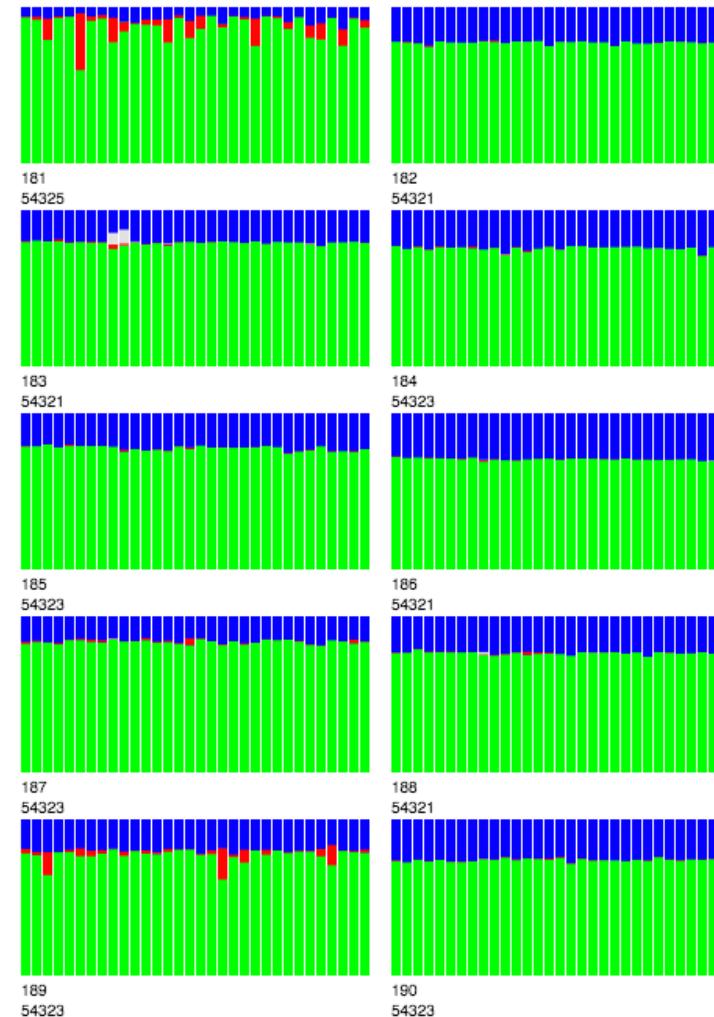
Name	Ping	Cores	Load	My CPU %	Sys	Shut Down	Data (Used/Total)	Data (% Cached)	GC (Free / Total / Max)	Disk (Free / Max)	Disk (% Free)
✓ 172.16.2.181:54323	a few seconds ago	32	6.110	0	8	-	40.603	33.82 GB / s	29.46 GB / NaN undefined / 29.58 GB	339.08 GB / 1.70 TB	19%
✓ 172.16.2.182:54321	a few seconds ago	32	0.240	7	8	-	44.566	39.59 GB / s	29.43 GB / NaN undefined / 29.58 GB	225.64 GB / 1.70 TB	12%
✓ 172.16.2.183:54321	a few seconds ago	32	9.820	0	3	-	44.883	42.09 GB / s	29.34 GB / NaN undefined / 29.58 GB	450.18 GB / 1.70 TB	25%
✓ 172.16.2.184:54323	a few seconds ago	32	0.990	0	0	-	44.656	41.67 GB / s	29.51 GB / NaN undefined / 29.58 GB	254.96 GB / 1.70 TB	14%
✓ 172.16.2.185:54323	a few seconds ago	32	0.440	8	8	-	43.128	38.33 GB / s	29.43 GB / NaN undefined / 29.58 GB	501.02 GB / 1.70 TB	28%
✓ 172.16.2.186:54321	a few seconds ago	32	1.750	0	0	-	44.589	42.46 GB / s	29.42 GB / NaN undefined / 29.58 GB	331.27 GB / 1.70 TB	18%
✓ 172.16.2.187:54323	a few seconds ago	32	1.490	0	10	-	43.993	42.00 GB / s	29.46 GB / NaN undefined / 29.58 GB	367.40 GB / 1.70 TB	21%
✓ 172.16.2.188:54321	a few seconds ago	32	0.610	0	8	-	41.977	18.63 GB / s	28.30 GB / NaN undefined / 29.58 GB	218.27 GB / 1.70 TB	12%
✓ 172.16.2.189:54323	a few seconds ago	32	4.420	6	9	-	48.590	38.91 GB / s	29.34 GB / NaN undefined / 29.58 GB	477.97 GB / 1.70 TB	27%
✓ 172.16.2.190:54323	a few seconds ago	32	2.970	10	12	-	43.931	22.15 GB / s	29.51 GB / NaN undefined / 29.58 GB	274.50 GB / 1.70 TB	15%
✓ TOTAL	-	320	28.840	-	-	-	440.916	359.62 GB / s	293.18 GB / NaN undefined / 295.83 GB	3.36 TB / 17.04 TB	19%

$$10 \times 32 = \\ 320 \text{ Cores}$$

$$10 \times 29.6 = 296 \\ \text{GB Memory}$$

H₂O Water Meter (CPU Usage Monitor)

10 x 32 = 320 Cores



Legend

Each bar represents one CPU.

Blue: idle time

Green: user time

Red: system time

White: other time (e.g. i/o)

H2O4GPU

H2O4GPU – H2O.ai Blog

Secure | https://blog.h2o.ai/tag/h2o4gpu/

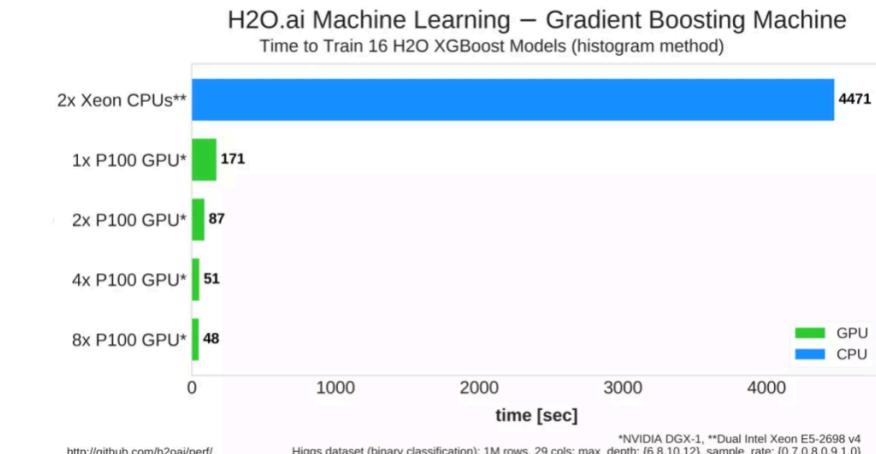
H2O.ai Releases H2O4GPU, the Fastest Collection of GPU Algorithms on the Market, to Expedite Machine Learning in Python

BY AVNI WADHWA ON SEPTEMBER 26, 2017 – 0 COMMENTS

H2O4GPU is an open-source collection of GPU solvers created by H2O.ai. It builds on the easy-to-use scikit-learn Python API and its well-tested CPU-based algorithms. It can be used as a drop-in replacement for scikit-learn with support for GPUs on selected (and ever-growing) algorithms. H2O4GPU inherits all the existing scikit-learn algorithms and falls back to CPU algorithms when the GPU algorithm does not support an important existing scikit-learn class option. It utilizes the efficient parallelism and high throughput of GPUs. Additionally, GPUs allow the user to complete training and inference much faster than possible on ordinary CPUs.

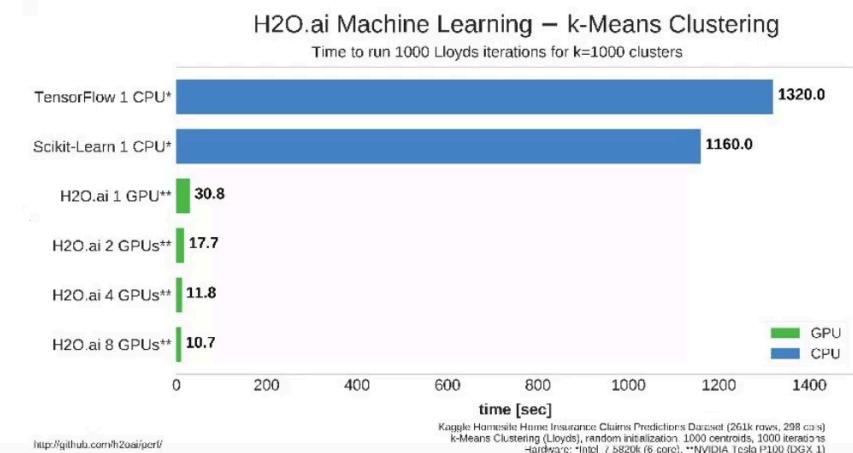
Today, select algorithms are GPU-enabled. These include Gradient Boosting Machines (GBM's), Generalized Linear Models (GLM's), and K-Means Clustering. Using H2O4GPU, users can unlock the power of GPU's through the scikit-learn API that many already use today. In addition to the scikit-learn Python API, an R API is in development.

<https://blog.h2o.ai/tag/h2o4gpu/>



k-Means Clustering

- Based on NVIDIA prototype of k-Means algorithm in CUDA
- Improvements to original implementation:
 - Significantly faster than scikit-learn implementation (50x) and other GPU implementations (5-10x)
 - Supports multiple GPUs



H₂O.ai

Driverless AI

H2O.ai and NVIDIA Bring Fast, Accurate and Interpretable Driverless AI with Automated Machine Learning and Feature Engineering

Businesses Can Leapfrog Data Scientist Shortage and Accelerate Adoption of AI with H2O.ai Driverless AI on NVIDIA DGX Systems

September 26, 2017 04:57 PM Eastern Daylight Time

NEW YORK--(BUSINESS WIRE)--Strata Data Conference — H2O.ai today announced an offering built on NVIDIA DGX Systems to further democratize machine learning and address the growing demands placed on a limited number of trained data scientists.

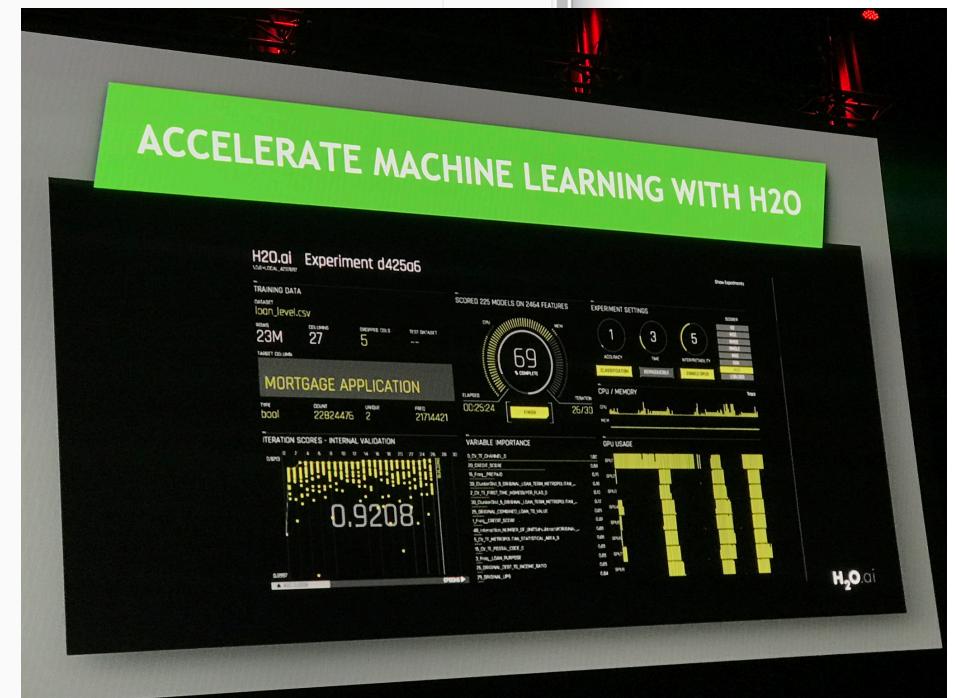
"Interpretable AI builds trust in AI and automates report generation for regulatory purposes. Driverless AI on GPU Computing Platforms accelerates learning and data products with AI in enterprises."

[Tweet this](#)

Engineering and quickly develop hundreds of machine learning models to help businesses mitigate risks and maximize revenue

[View All](#)

The screenshot shows a news article from Business Wire. The header features the Business Wire logo and navigation links for HOME, SERVICES, NEWS, EDUCATION, and ABOUT US. A search bar and user account links for Log In and Sign Up are also present. The main content discusses H2O.ai's announcement at the Strata Data Conference, highlighting their offering built on NVIDIA DGX Systems. It emphasizes the need to democratize machine learning and address the shortage of data scientists. A sidebar on the left includes social sharing icons for LinkedIn, Facebook, Twitter, Reddit, Email, and a Plus sign. At the bottom, there are file download options for 'DKOKfLaUMAEp...jpg-large' and a 'Show All' button.



<http://www.businesswire.com/news/home/20170926006769/en/H2O.ai-NVIDIA-Bring-Fast-Accurate-Interpretable-Driverless>

Thank you!

- My STREAM Supervisors
 - Prof. Dragan Savić
 - Prof. Zoran Kapelan
- Code, Slides & Documents
 - bit.ly/h2o_exeter_2017
 - docs.h2o.ai
 - bit.ly/h2o_meetups
- Contact
 - joe@h2o.ai
 - [@matlabulous](https://twitter.com/matlabulous)
 - github.com/woobe
- Please search/ask questions on **Stack Overflow**
 - Use the tag `h2o` (not H2 zero)