

H2O Machine Learning & Deep Learning London Workshop



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

Data Science for IoT Meetup
Barclays Eagle Venture Labs
21st November, 2016

About Me: Civil Engineer → Data Scientist

- 2005 - 2015
 - Water Engineer
 - Consultant for Utilities
 - Industrial PhD
 - Water Engineering + Machine Learning
- 2015 - Present
 - Data Scientist
 - Virgin Media (UK)
 - Domino Data Lab (US)
 - H2O.ai (US)



Why? Long story – see bit.ly/joe_h2o_talk2

Agenda

- Workshop 1
 - H2O Introduction
 - Quick Demo
 - Step-by-Step Example 1
 - Binary Classification
 - Import & Explore
 - Build & Evaluate
 - Parameters Tuning
- Workshop 2
 - Example 1 (cont'd)
 - Grid Search
 - Model Stacking
 - Example 2
 - Anomaly Detection
 - Other H2O Features

About H2O.ai

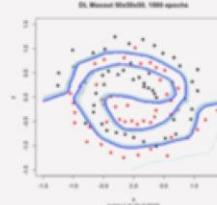
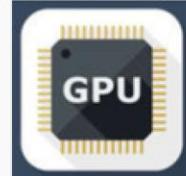


About H2O.ai

- H2O.ai, the Company
 - Team: 80
 - Founded in 2012
 - HQ: Mountain View, California
- H2O, the Platform
 - Open Source (Apache 2.0)
 - Algorithms written in Java
 - Fast, distributed and scalable
 - Multiple interfaces to suit different users
 - Web, R, Python, Java, Scala, REST/JSON
 - Works with desktop/laptop, cloud, Spark and Hadoop



H₂O.ai Makes A Difference as an AI Platform

Open Source	Integration with Big Data Frameworks	Flexible Interface	Smart and Fast Algorithms
	 And More...	   H₂O Flow	
Scalability and Performance	Rapid Model Deployment	GPU Enablement	Cloud Integration
 <ul style="list-style-type: none">Distributed In-Memory Computing PlatformDistributed AlgorithmsFine-Grain MapReduce	<ul style="list-style-type: none">Highly portable models deployed in Java (POJO)Automated and streamlined scoring service deployment with Rest API <p><small>"Confidential" REST API of H2O.ai. All rights reserved"</small></p>		  

H₂O.ai Offers AI Open Source Platform

Product Suite to Operationalize Data Science with Visual Intelligence



Visual Intelligence and UX Framework For
Beautiful Data Products

100% Open Source



In-Memory, Distributed
Machine Learning Algorithms
with H₂O Flow GUI

Deep
Water

Best in class
Deep Learning on GPUs with
TensorFlow, MXNet or Caffe
with the ease of use of H₂O

Spark + H₂O
SPARKLING
WATER

H₂O AI Open Source Engine
Integration with Spark

Steam

Operationalize and Streamline
Model Building, Training and
Deployment Automatically
and Elastically

"Confidential and property of H₂O.ai. All rights reserved"

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*



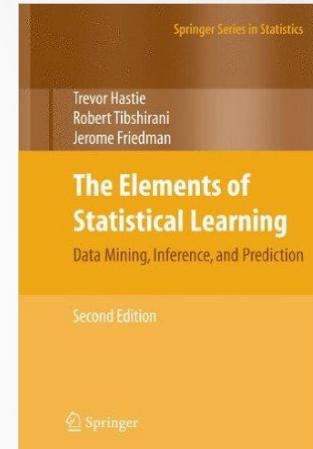
Dr. Robert Tibshirani

- 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*



Current Algorithm Overview

Statistical Analysis

- Linear Models (GLM)
- Naïve Bayes

Ensembles

- Random Forest
- Distributed Trees
- Gradient Boosting Machine
- R Package - Stacking / Super Learner

Workshop

Deep Neural Networks

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

Clustering

- K-Means

Dimension Reduction

- Principal Component Analysis
- Generalized Low Rank Models

Joe's Strata Hadoop
London Talk
bit.ly/joe_h2o_talk4

Solvers & Optimization

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

Data Munging

- Scalable Data Frames
- Sort, Slice, Log Transform

Joe's LondonR Talk
bit.ly/joe_h2o_talk3



H2O.ai allows people to be
human again





H2O is an open source platform
empowering business
transformation

Bring AI To Business Empower Transformation

Financial Services as Our Vertical Focus

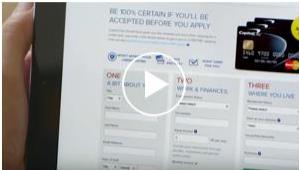


Community as Our Foundation

H2O In Action

www.h2o.ai/customers

Capital One



Capital One uses H2O open source machine learning for various use cases.

MarketShare



Predicting Marketing Results Through Analytics

H2O predictive analytics helps boost the impact and results of digital marketing.

Kaiser



Kaiser uses H2O machine learning to save lives.

Zurich Insurance



Zurich turned to H2O as a strategic differentiator for commercial insurance.

Progressive



Progressive uses H2O predictive analytics for user-based insurance.

Comcast



Comcast uses H2O to improve customer experience.

Hospital Corporation of America



HCA uses H2O to predict patient outcomes in real-time.

McKesson



McKesson discusses the adoption of artificial intelligence in healthcare.

Macy's



Macy's uses H2O for personalized site recommendations.

Transamerica



Transamerica turns to H2O to develop a product recommendation platform for insurance.

Paypal



Paypal turned to H2O Deep Learning for fraud detection and customer churn.

eBay



eBay chose H2O for open source machine learning.



H2O for Kaggle Competitions

CIFAR-10 Competition
Winners: Interviews with Dr.
Ben Graham, Phil Culliton, &
Zygmunt Zajac

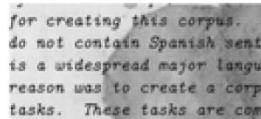
Triskelion | 01.02.2015

[READ MORE](#)

“I did really like H2O’s deep learning implementation in R, though - the interface was great, the back end extremely easy to understand, and it was scalable and flexible. Definitely a tool I’ll be going back to.”

Kaggle challenge
2nd place winner
Colin Priest

[READ MORE](#)



Completed • Knowledge • 161 teams

Denoising Dirty Documents

Mon 1 Jun 2015 – Mon 5 Oct 2015 (3 months ago)

“For my final competition submission I used an ensemble of models, including 3 deep learning models built with R and h2o.”

H2O for Academic Research

European Journal of Operational Research

Available online 22 October 2016

In Press, Accepted Manuscript — Note to users

Innovative Applications of O.R.

Deep neural networks, gradient-boosted trees, random forests:
Statistical arbitrage on the S&P 500

Christopher Krauss^{a, b}, Xuan Anh Do^a, Nicolas Huck^{a, b}

Received 15 April 2016, Revised 22 August 2016, Accepted 18 October 2016, Available online 22 October 2016

Highlights

- Latest machine learning techniques are deployed in a statistical arbitrage context.
- Deep neural networks, gradient-boosted trees, and random forests are considered.
- An equal-weighted ensemble of these techniques produces the best performance.
- Daily returns are substantial though declining over time.
- The system is especially effective at times of financial turmoil.

<http://www.sciencedirect.com/science/article/pii/S0377221716308657>

Cornell University Library

We gratefully acknowledge support from the Simons Foundation and member institutions

arXiv.org > physics > arXiv:1509.01199

Search or Article-Id (Help | Advanced search) All papers ▾ Go!

Physics > Physics and Society

Download:

- PDF
- Other formats

(license)

Current browse context: physics.soc-ph
< prev | next >
new | recent | 1509

Change to browse by:

- cs
- cs.CY
- physics
- physics.data-an
- stat
- stat.AP
- stat.ML

References & Citations

- INSPIRE HEP (refers to | cited by)
- NASA ADS

Bookmark (what is this?)

<https://arxiv.org/abs/1509.01199>

H2O Community Support

Google forum – h2osteam

The screenshot shows the Google forum interface for the group 'h2osteam'. The sidebar on the left includes links for 'Groups', 'My groups', 'Home', 'Starred', 'Favourites' (with a note to click the star icon), 'Recently viewed' (including 'H2O Open Sour...', 'sparkr-d...'), 'Recent searches' (including 'spark streaming (i...'), and 'Recently posted to H2O Open Sour...'). The main content area displays a post from 'H2O Open Source Scalable Machine Learning - h2ostream' with 30 of 2055 topics (99+ unread). A note at the bottom encourages users to shift their energy toward the new community website.

community.h2o.ai

Please try

The screenshot shows the community.h2o.ai website. The sidebar on the right includes links for 'Algorithms', 'Announcements', 'Artificial Intelligence', 'Deep Water', 'Demos', 'H2O', 'Java', 'Machine Learning', 'Python', 'R', 'Source Code', 'Sparkling Water', 'Steam', 'Tools', and 'Troubleshooting'. The main content area displays a feed of posts from 'All Posts', including topics like 'When is Steam going to be released?', 'H2O Python Modules', 'H2O Installation', 'PySparkling launch problem with Python 2.6 or older', 'Predicted Values', and 'Combining holdout predictions, while keep_cross_validation_predictions parameter is active in Python'. A 'Sparkling Water Release' announcement is also visible.

#AroundTheWorldWithH2Oai

London Kaggle Meetup



Strata Hadoop London



Chelsea FC



Big Data London



PyData Amsterdam



useR! 2016 Stanford



satRdays Budapest



Paris ML Meetup



Data Science Milan

Machine Learning Demo



A Typical Machine Learning Task

- Demo
 - Dataset – MNIST
 - LeCun et al. (1999)
 - Hand-written Digits
 - Import & Explore Data
 - Build & Evaluate Models
 - Make Predictions



MNIST Hand-Written Digits

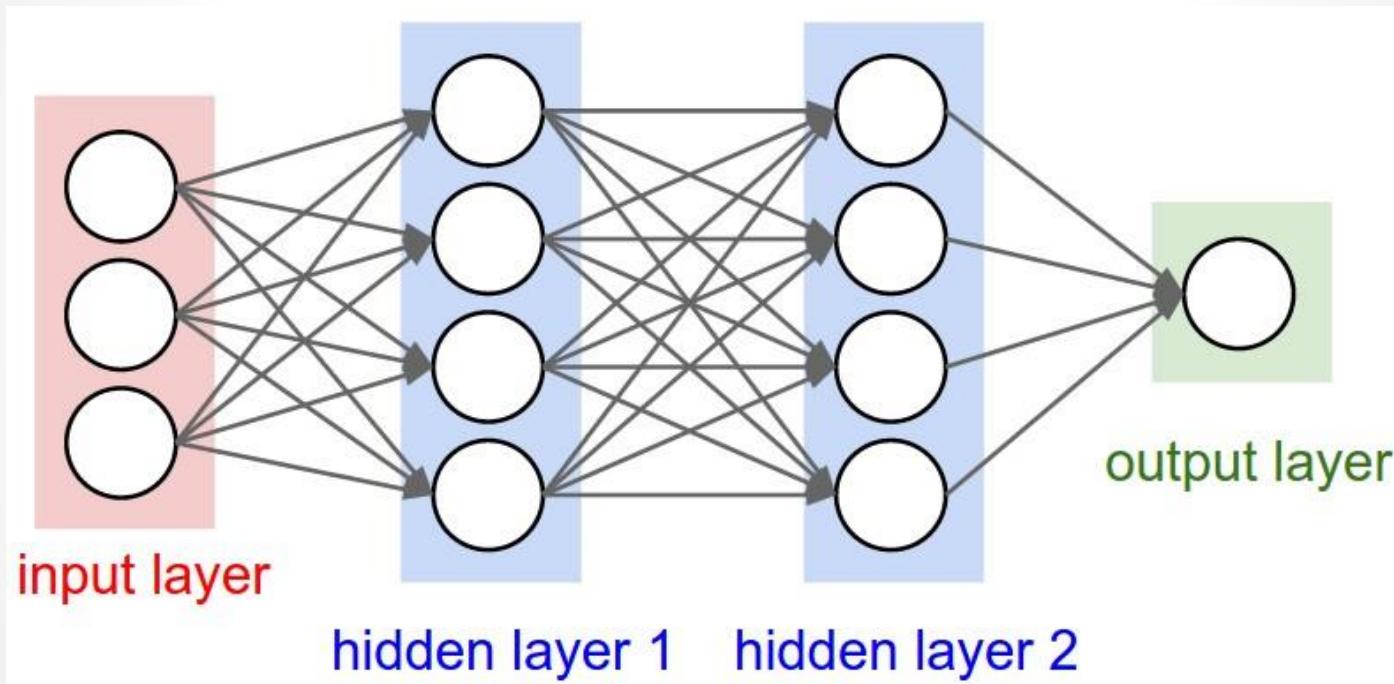
- **784 Inputs**
 - $28 \times 28 = 784$ pixels
 - **1 Output**
 - 0, 1, 2, 3, 4, 5, 6, 7, 8 or 9
 - Classification
 - **Files**
 - Train (42k Records)
 - Test (28k)
 - **Links**
 - <https://www.kaggle.com/c/digit-recognizer/data>



$$= 784 \text{ pixels}$$

Photo credit: https://ml4a.github.io/ml4a/neural_networks/

A Simple Neural Network



H2O Deep Learning in Action

116M rows, 6GB CSV file
800+ predictors (numeric + categorical)

airlines_all_selected_cols.hex

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

Rows	Columns	Compressed Size
116695259	12	2GB



Job

Run Time: 00:00:36.712

Remaining Time: 00:00:17.188

Type: Model

Key: Q_deeplearning-dd2f42f7-81f7-42e8-9d98-e34437309828

Description: DeepLearning

Status: RUNNING

Progress: 69%

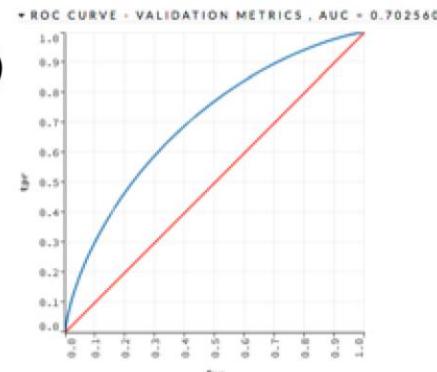
Iterations: 12. Epochs: 0.628821. Speed: 2,243,735 samples/sec. Estimated time left: 21.849 sec

Actions: View Cancel Job

* OUTPUT - STATUS OF NEURON LAYERS (PREDICTING ISDEPDELAYED, 2-CLASS CLASSIFICATION, BERNoulli DISTRIBUTION, CROSSENTROPY LOSS, 17,462 WEIGHTS/BIASES, 221.3 KB, 106,585,365 TRAINING SAMPLES, MINI-BATCH SIZE 1)

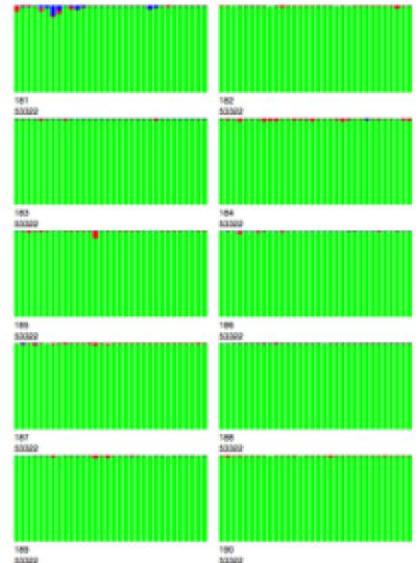
layer	units	type	dropout	l1	l2	mean_rate	rate_rms	momentum	mean_weight	weight_rms	mean_bias	bias_rms
1	897	Input	0									
2	20	Rectifier	0	0	0	0.0463	0.2020	0	-0.0021	0.2111	-0.9139	1.0036
3	20	Rectifier	0	0	0	0.0197	0.2027	0	-0.1053	0.5362	-1.3908	1.5259
4	20	Rectifier	0	0	0	0.0517	0.0446	0	-0.1575	0.3068	-0.0846	0.6046
5	20	Rectifier	0	0	0	0.0761	0.0844	0	-0.0374	0.2275	-0.2647	0.2481
6	2	Softmax	0	0	0	0.0161	0.0083	0	0.0741	0.7260	0.4269	0.2056

model trained in <1 min:
2M+ samples/second



Threshold: Choose... Criterion: Choose...

VARIABLE IMPORTANCES



Legend

Each bar represents one CPU.

Blue: idle time

Green: user time

Red: system time

White: other time (e.g. Vo)

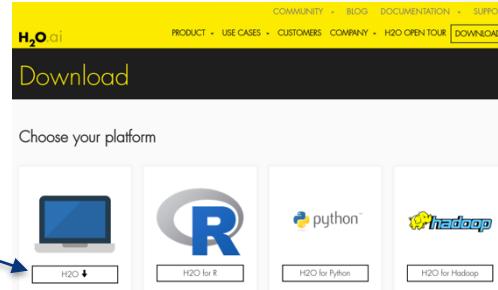
10 nodes: all 320 cores busy

H2O Flow Demo



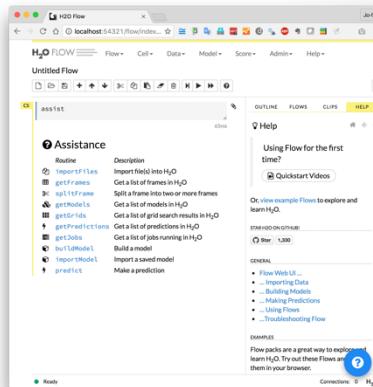
H2O Flow (Web Interface) Demo

- Download and unzip jar from www.h2o.ai



- In terminal:
 - java -jar h2o.jar
- Web browser:
 - localhost:54321

```
Jo-fais-MacBook-Pro-2:~ jofaichow$ cd h2o-3.10.0.6
Jo-fais-MacBook-Pro-2:h2o-3.10.0.6 jofaichow$ java -jar h2o.jar
09-18 13:16:13.620 192.168.0.6:54321 8620 main INFO: ----- H2O started -----
09-18 13:16:13.636 192.168.0.6:54321 8620 main INFO: Build git branch: rel-turing
09-18 13:16:13.636 192.168.0.6:54321 8620 main INFO: Build git hash: 3b286dea7b719b6ef2c2f5f7728648f2440a1502
09-18 13:16:13.636 192.168.0.6:54321 8620 main INFO: Build git describe: jenkins-rel-turing-6
09-18 13:16:13.636 192.168.0.6:54321 8620 main INFO: Build project version: 3.10.0.6 (latest version: 3.10.0.6)
```



H2O Flow Examples

The screenshot shows the H2O Flow web application running in a browser. The title bar reads "H2O Flow". The main menu includes "Flow", "Cell", "Data", "Model", "Score", "Admin", and "Help". Below the menu, the title "Untitled Flow" is displayed, followed by a toolbar with various icons for file operations and navigation.

The central workspace has a search bar containing "assist" and a status bar indicating "57ms". To the right of the workspace is a sidebar with tabs: OUTLINE, FLOWS, CLIPS, and HELP. The HELP tab is highlighted with a yellow background and a black arrow points to it from the top right.

The help content area displays a section titled "Assistance" with a question mark icon. It lists several routines with their descriptions:

Routine	Description
importFiles	Import file(s) into H2O
getFrames	Get a list of frames in H2O
splitFrame	Split a frame into two or more frames
getModels	Get a list of models in H2O
getGrids	Get a list of grid search results in H2O
getPredictions	Get a list of predictions in H2O
getJobs	Get a list of jobs running in H2O
buildModel	Build a model
importModel	Import a saved model
predict	Make a prediction

At the bottom left, a green circle indicates "Ready". At the bottom right, it says "Connections: 0" and there is a blue circular icon with a white question mark.

H2O Flow Demo

- Tasks:
 - Import data
 - Visualize data
 - Split data
 - 80% training
 - 20% validation
 - Build models

- Tasks:
 - Evaluate models
 - Use models for predictions
 - Grid search

H2O + R Demo



Using H2O's R Interface

```
## H2O R Demo - MNIST
# Load H2O's R package
library(h2o)
# Start and connect to a local H2O cluster
h2o.init(nthreads = -1)
# Import CSV files
d_train <- h2o.importFile("MNIST_Kaggle_train.csv")
d_test <- h2o.importFile("MNIST_Kaggle_test.csv")
# Convert "label" column to categorial values
d_train$label <- as.factor(d_train$label)
# Split training data into train/valid
d_split <- h2o.splitFrame(d_train, ratios = 0.8, seed = 1234)
# Define features and target
features <- setdiff(colnames(d_train), "label")
target <- "label"
# Train a deep learning model with 80% data
model <- h2o.deeplearning(x = features,
                           y = target,
                           training_frame = d_split[[1]], # 80% split
                           standardize = TRUE,
                           activation = "RectifierWithDropout",
                           hidden = c(50, 50),
                           epochs = 10)
print(model)
# Evaluate performance with 20% data
h2o.performance(model, d_split[[2]])
# Make predictions using model
yhat_test <- h2o.predict(model, d_test)
print(head(yhat_test))
```

Start a local cluster from R

Import data

Convert integers (0...9) to categorical values

Split dataset for simple validation

Define features and target

R API for H2O Deep Learning

Evaluate performance

Making predictions

Other H2O Interfaces

- R

```
1 # Load H2O R package
2 library(h2o)
3
4 # Initialize and Connect to H2O
5 h2o.init()
```

- Python

```
1 # Import H2O Python module
2 import h2o
3
4 # Initialize and Connect to H2O
5 h2o.init()
```

- **docs.h2o.ai**

Key Resources

H2O, Sparkling Water, and Steam Documentation

Getting Started Data Science Algorithms Languages Tutorials, Examples, & Presentations For Developers For the Enterprise

Getting Started

H2O

What is H2O?
H2O User Guide
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
RSparkling Readme
Open Source License (Apache V2)

Quick Start Video - Scala
Quick Start Video - Python

Download Sparkling Water

Steam

What is Steam?
Steam User Guide
Recent Changes
Open Source License (AGPL)

Download Steam

Questions and Answers

FAQ
Community Forum
H2Ostream Google Group
Issue Tracking (JIRA)
Gitter
Stack Overflow
Cross Validated

For Supported Enterprise Customers
Enterprise Support via Web | Email

Data Science 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
Ensembles (Stacking)	Tutorial	Booklet	Reference	Tuning

Unsupervised Learning

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

Languages

R

Quick Start Video - R
R Package Docs
R Booklet
Examples and Demos
R FAQ

Python

Quick Start Video - Python
Python Module Docs
Python Booklet
Examples and Demos
Python FAQ

Java

POJO Model Javadoc
H2O Core Javadoc
H2O Algorithms Javadoc

Scala

Sparkling Water API
Sparkling Water Scaladoc
H2O Scaladoc