Making Machine Learning Accessible to Everyone

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

Big Data London
London Olympia Conference Centre
3rd November, 2016

About Me: Civil Engineer → Data Scientist

- 2005 - 2015
- Water Engineer
  - Consultant for Utilities
  - Industrial PhD
    - Water Engineering + Machine Learning
    - *Discovered H2O in 2014!*

- 2015 - Present
- Data Scientist
  - Virgin Media (UK)
  - Domino Data Lab (US)
  - H2O.ai (US)

Why? Long story – see bit.ly/joe_h2o_talk2

Agenda

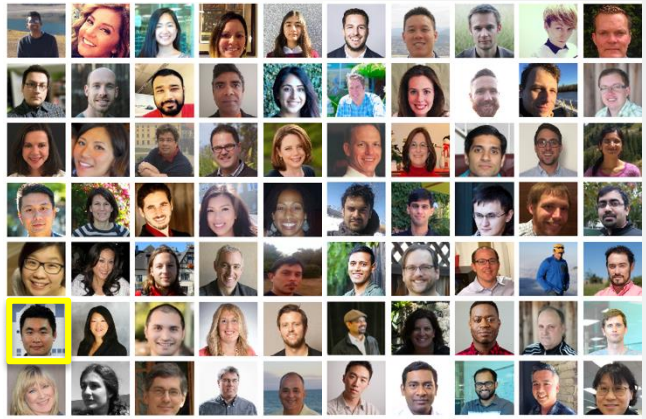
- This Talk
  - About H2O.ai
  - H2O's Web Interface
  - Why H2O?
  - What's Next?
- Data Science London Meetup
  - Main Auditorium
  - 6:30 pm
  - Deep Water
    - H2O's integration with other deep learning libraries
  - Live Demo

About H2O.ai

About H2O.ai

- H2O.ai, the Company
  - Team: 80 (70 shown)
  - 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



Scientific Advisory Council

Dr. Trevor Hastie

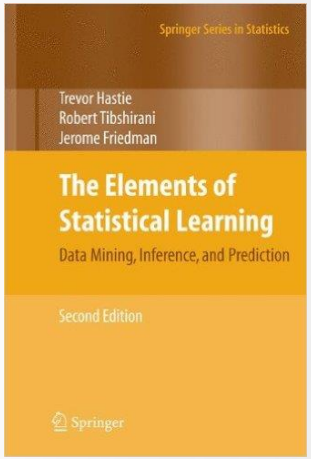
- John A. Overdeck Professor of Mathematics, Stanford University
- PhD in Statistics, Stanford University
- Co-author, *The Elements of Statistical Learning: Prediction, Inference and Data Mining*
- Co-author with John Chambers, *Statistical Models in S*
- Co-author, *Generalized Additive Models*

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

Deep Neural Networks

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

Clustering

- K-Means

Dimension Reduction

- Principal Component Analysis
- Generalized Low Rank Models

Solvers & Optimization

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

Data Munging

- Scalable Data Frames
- Sort, Slice, Log Transform

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

Joe's LondonR Talk
bit.ly/joe_h2o_talk3

H2O's Mission

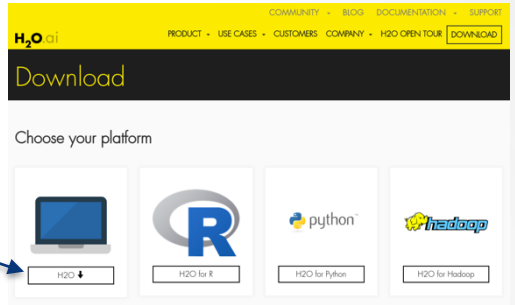
Making Machine Learning Accessible to Everyone

Photo credit: Virgin Media

H2O Web Interface

H2O Flow (Web Interface) Demo

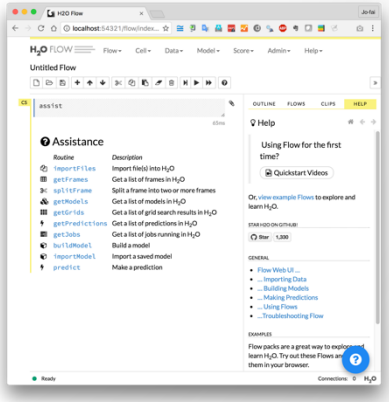
- Download and unzip **jar** from www.h2o.ai



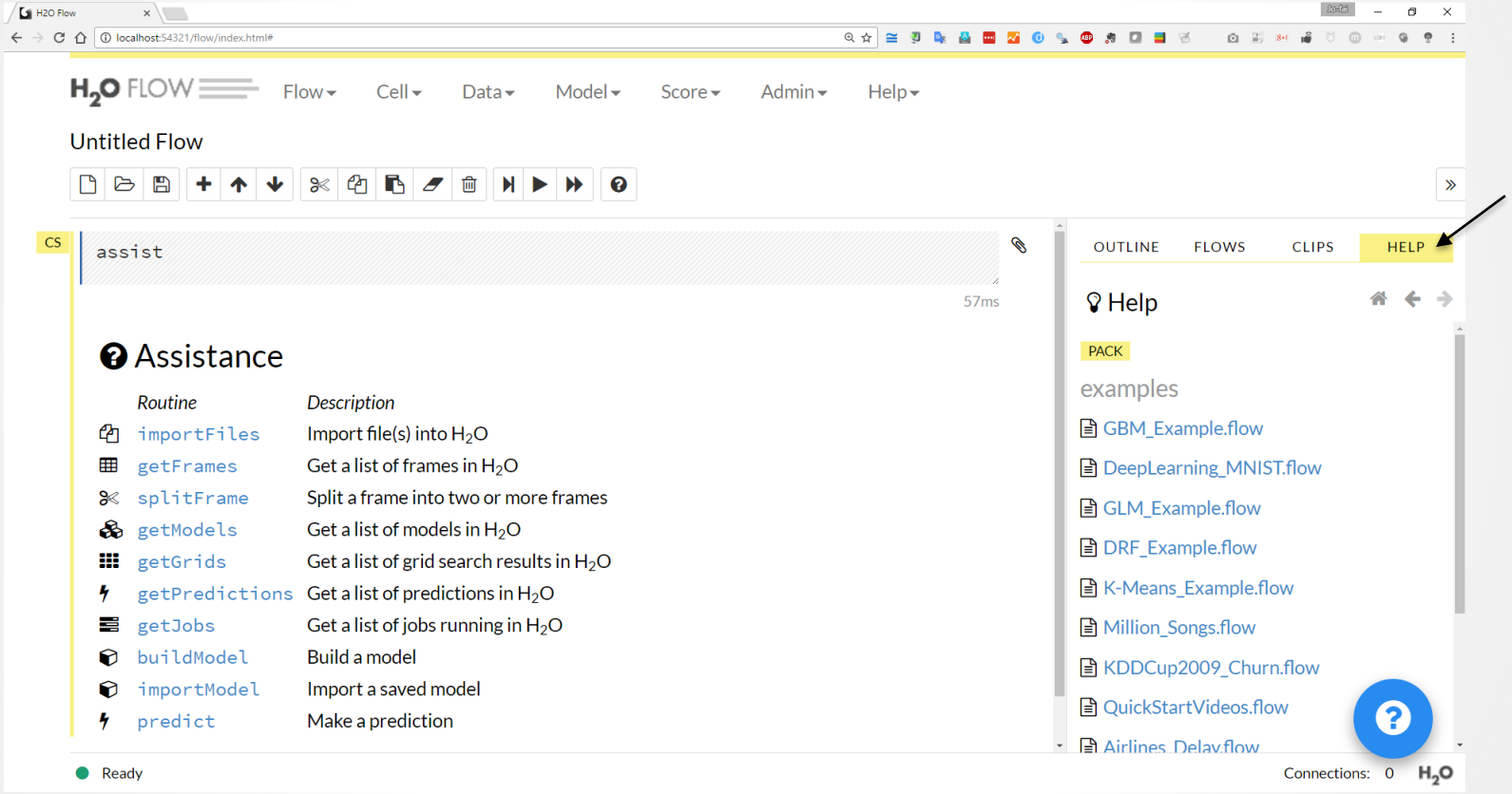
- In terminal:
  - java -jar h2o.jar

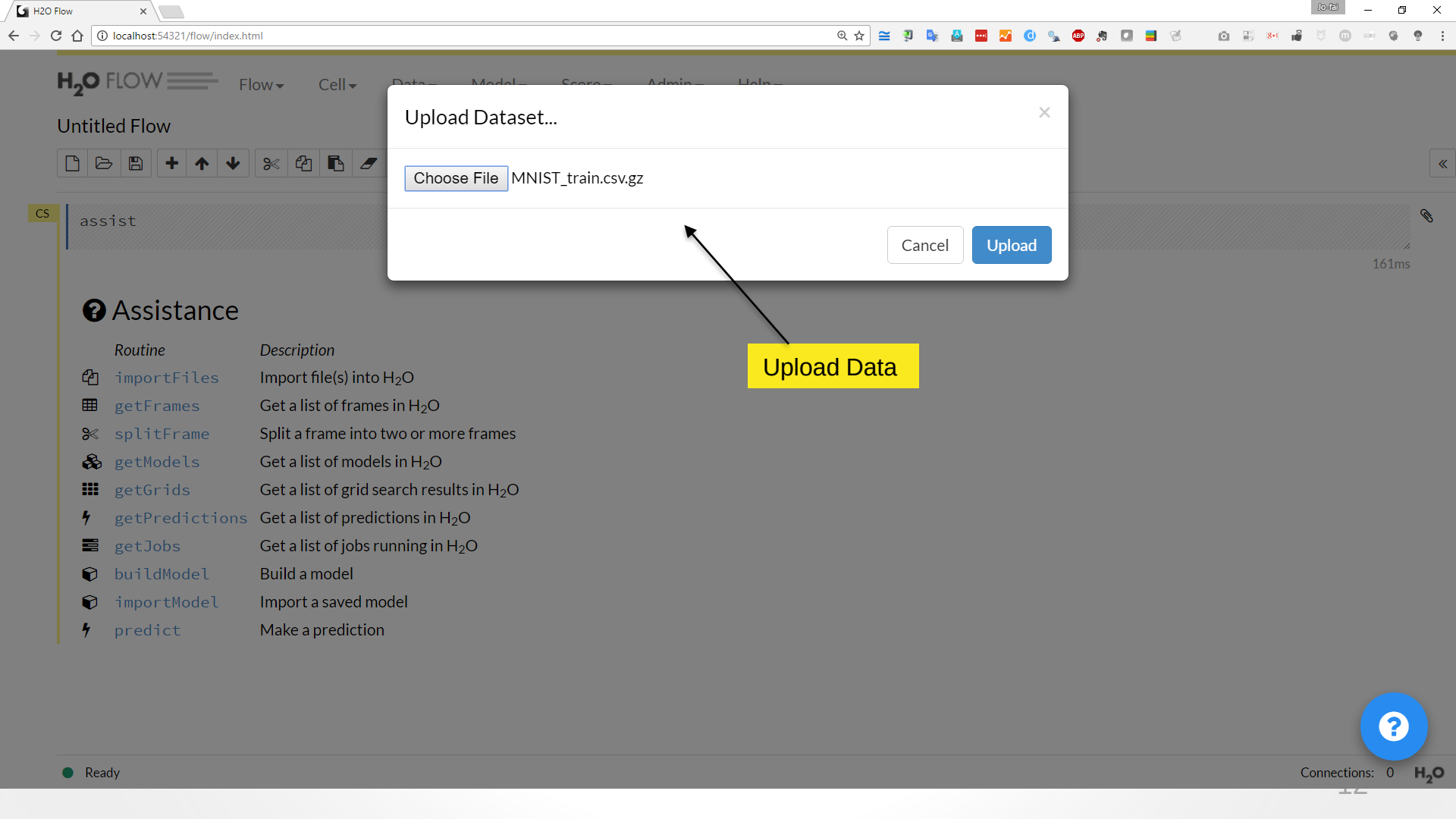
```
h2o-3.10.0.6 — java -jar h2o.jar — 123×24
Jo-fais-MacBook-Pro-2:~ jofaichow$ cd h2o-3.10.0.6
Jo-fais-MacBook-Pro-2:h2o-3.10.0.6 jofaichow$ java -jar h2o.jar
09-18 13:16:13.620 192.168.0.6:54321     8620   main      INFO: ----- H2O started  -----
09-18 13:16:13.636 192.168.0.6:54321     8620   main      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)
```

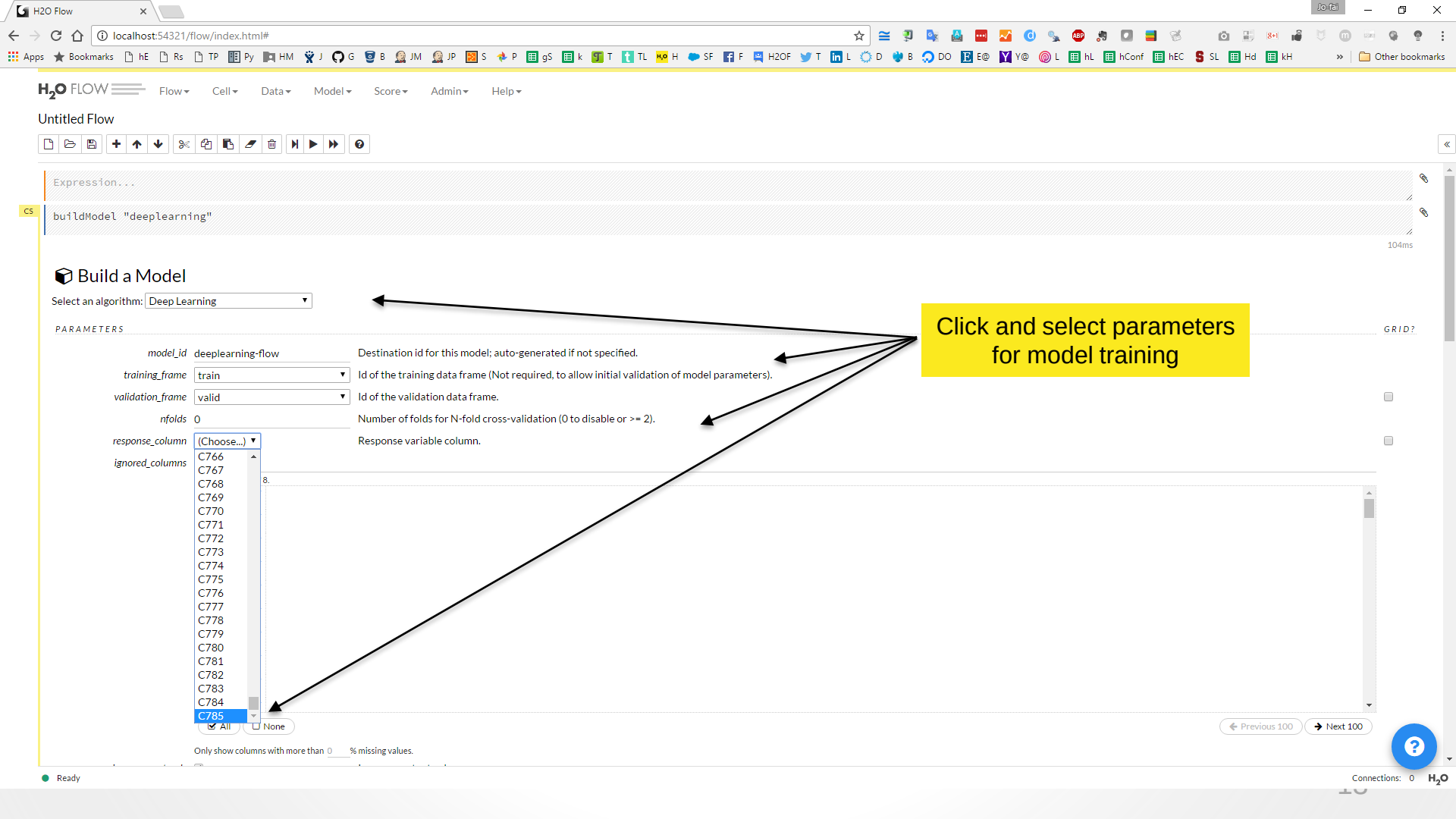
- Web browser:
  - localhost:54321

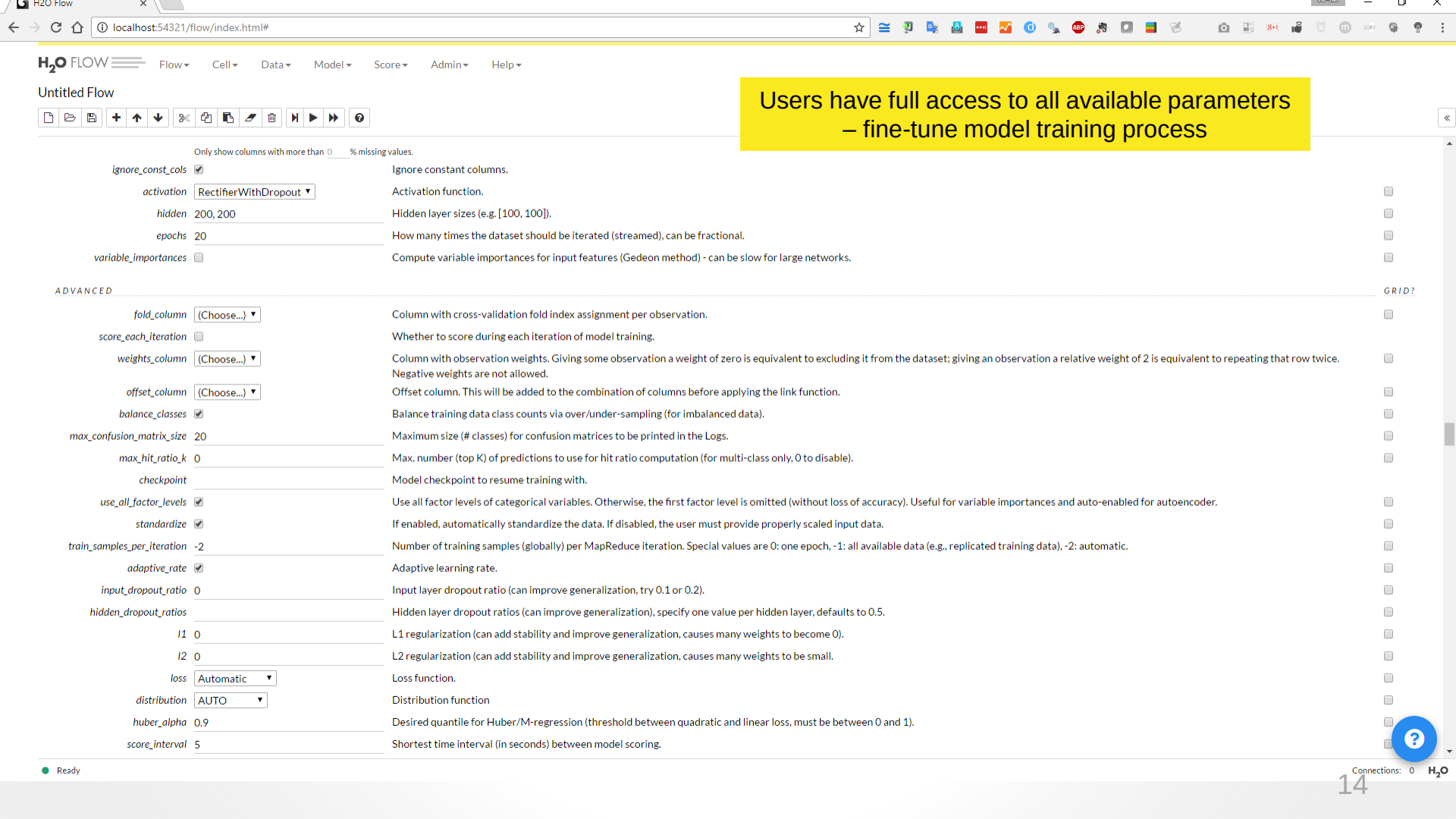


H2O Flow Examples




H2O Flow
localhost:54321/flow/index.html
H2O FLOW
Flow
Cell
Untitled Flow
Upload Dataset...
Choose File
MNIST_train.csv.gz
Cancel
Upload
Upload Data
CS
assist
161ms
Assistance
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
Ready
Connections: 0


H2O Flow
localhost:54321/flow/index.html#
Flow
Cell
Data
Model
Score
Admin
Help
Untitled Flow
Expression...
buildModel "deeplearning"
104ms
Build a Model
Select an algorithm: Deep Learning
PARAMETERS
GRID?
model_id
deeplearning-flow
Destination id for this model; auto-generated if not specified.
training_frame
train
Id of the training data frame (Not required, to allow initial validation of model parameters).
validation_frame
valid
Id of the validation data frame.
nfolds
0
Number of folds for N-fold cross-validation (0 to disable or >= 2).
response_column
(Choose...)
Response variable column.
ignored_columns
C766
C767
C768
C769
C770
C771
C772
C773
C774
C775
C776
C777
C778
C779
C780
C781
C782
C783
C784
C785
All
None
Previous 100
Next 100
Only show columns with more than 0 % missing values.
Click and select parameters for model training
Ready
Connections: 0


H2O Flow
localhost:54321/flow/index.html#
H2O FLOW
Flow
Cell
Data
Model
Score
Admin
Help
Untitled Flow
Users have full access to all available parameters – fine-tune model training process
Only show columns with more than 0 % missing values.
ignore_const_cols
Ignore constant columns.
activation
RectifierWithDropout
Activation function.
hidden
200, 200
Hidden layer sizes (e.g. [100, 100]).
epochs
20
How many times the dataset should be iterated (streamed), can be fractional.
variable_importances
Compute variable importances for input features (Gedeon method) - can be slow for large networks.
ADVANCED
GRID?
fold_column
(Choose...)
Column with cross-validation fold index assignment per observation.
score_each_iteration
Whether to score during each iteration of model training.
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.
offset_column
(Choose...)
Offset column. This will be added to the combination of columns before applying the link function.
balance_classes
Balance training data class counts via over/under-sampling (for imbalanced data).
max_confusion_matrix_size
20
Maximum size (# classes) for confusion matrices to be printed in the Logs.
max_hit_ratio_k
0
Max. number (top K) of predictions to use for hit ratio computation (for multi-class only, 0 to disable).
checkpoint
Model checkpoint to resume training with.
use_all_factor_levels
Use all factor levels of categorical variables. Otherwise, the first factor level is omitted (without loss of accuracy). Useful for variable importances and auto-enabled for autoencoder.
standardize
If enabled, automatically standardize the data. If disabled, the user must provide properly scaled input data.
train_samples_per_iteration
-2
Number of training samples (globally) per MapReduce iteration. Special values are 0: one epoch, -1: all available data (e.g., replicated training data), -2: automatic.
adaptive_rate
Adaptive learning rate.
input_dropout_ratio
0
Input layer dropout ratio (can improve generalization, try 0.1 or 0.2).
hidden_dropout_ratios
Hidden layer dropout ratios (can improve generalization), specify one value per hidden layer, defaults to 0.5.
l1
0
L1 regularization (can add stability and improve generalization, causes many weights to become 0).
l2
0
L2 regularization (can add stability and improve generalization, causes many weights to be small.
loss
Automatic
Loss function.
distribution
AUTO
Distribution function
huber_alpha
0.9
Desired quantile for Huber/M-regression (threshold between quadratic and linear loss, must be between 0 and 1).
score_interval
5
Shortest time interval (in seconds) between model scoring.
Ready
Connections: 0
H2O

H2O Flow

localhost:54321/flow/index.html#

H2O FLOW Flow Cell Data Model Score Admin Help

Untitled Flow

max_categorical_features	2147483647	Max. number of categorical features, enforced via hashing. #Experimental
reproducible	☐	Force reproducibility on small data (will be slow - only uses 1 thread).
export_weights_and_biases	☐	Whether to export Neural Network weights and biases to H2O Frames.
mini_batch_size	1	Mini-batch size (smaller leads to better fit, larger can speed up and generalize better).
elastic_averaging	☐	Elastic averaging between compute nodes can improve distributed model convergence. #Experimental

Build Model

CS

```
buildModel 'deeplearning', {"model_id":"deeplearning-flow","training_frame":"train","validation_frame":"valid","nfolds":0,"response_column":"C785","ignored_columns":
[],"ignore_const_cols":true,"activation":"RectifierWithDropout","hidden":
[200,200],"epochs":"20","variable_importances":false,"score_each_iteration":false,"balance_classes":true,"max_confusion_matrix_size":20,"max_hit_ratio_k":0,"checkpoint":"","use_all_factor_levels":true,"standardize":true,"train_samples_per_iteration":-2,"adaptive_rate":true,"input_dropout_ratio":0,"hidden_dropout_ratios":
[],"l1":0,"l2":0,"loss":"Automatic","distribution":"AUTO","huber_alpha":0.9,"score_interval":5,"score_training_samples":10000,"score_validation_samples":0,"score_duty_cycle":0.1,"stopping_rounds":5,"stopping_metric":"AUTO","stopping_tolerance":0,"max_runtime_secs":0,"autoencoder":false,"categorical_encoding":"AUTO","class_sampling_factors":
[],"max_after_balance_size":5,"pretrained_autoencoder":"","overwrite_with_best_model":true,"target_ratio_comm_to_comp":0.05,"seed":-1,"rho":0.99,"epsilon":1e-
8,"nesterov_accelerated_gradient":true,"max_w2":"Infinity","initial_weight_distribution":"UniformAdaptive","classification_stop":0,"score_validation_sampling":"Uniform","diagnostics":true,"fast_mode":true,"force_load_balance":true,"single_node_mode":false,"shuffle_training_data":false,"missing_values_handling":"MeanImputation","quiet_mode":false,"sparse":false,"col_major":false,"average_activation":0,"sparsity_beta":0,"max_categorical_features":2147483647,"reproducible":false,"export_weights_and_biases":false,"mini_batch_size":1,"elastic_averaging":false}
```

Started at 10:40:35 am

Job

Run Time	00:01:04.377
Remaining Time	00:02:45.486
Type	Model
Key	deeplearning-flow
Description	DeepLearning
Status	RUNNING
Progress	29%

Iterations: 5. Epochs: 5.00000. Speed: 5,553 samples/sec. Estimated time left: 3 min 10.601 sec

Actions View Cancel Job

Training the model with estimated remaining time – users can stop the process early if they want to

Ready

Connections: 0 H2O

H2O Flow

localhost:54321/flow/index.html#

H2O FLOW Flow Cell Data Model Score Admin Help

Untitled Flow

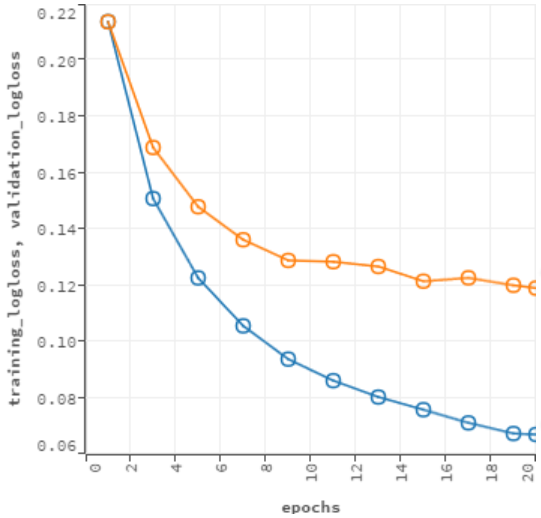
Model ID: deeplearning-flow

Algorithm: Deep Learning

Actions: Refresh | Predict... | Download POJO | Export | Inspect | Delete

▸ MODEL PARAMETERS

▾ SCORING HISTORY - LOGLOSS



Performance (logloss) on validation set

Performance (logloss) on training set

Ready

Connections: 0

Other H2O Interfaces

- R

```r
# Load H2O R package
library(h2o)

# Initialize and Connect to H2O
h2o.init()
```

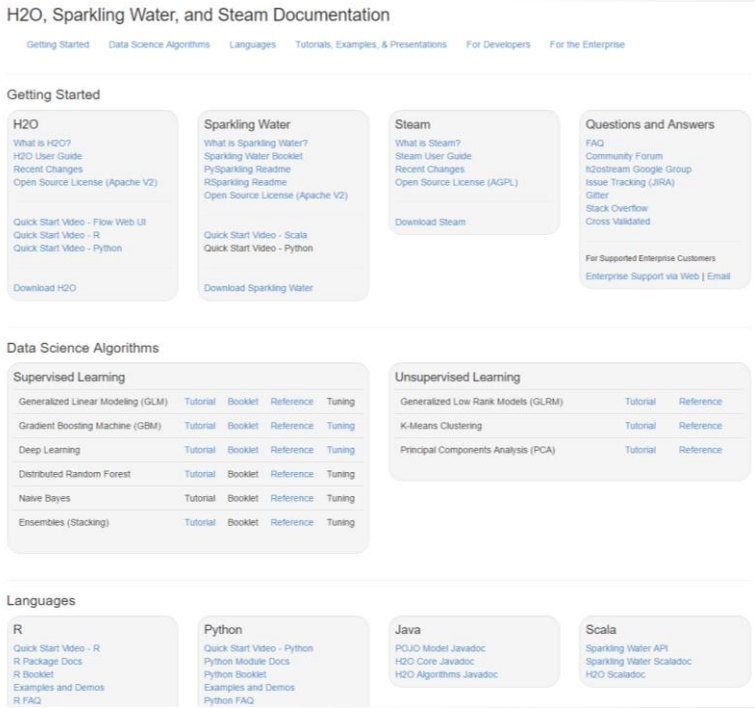
- Python

```python
# Import H2O Python module
import h2o

# Initialize and Connect to H2O
h2o.init()
```

Key Resources

- docs.h2o.ai



Advanced Features

- Advanced Features
  - Hyperparameters Tuning
  - Model Stacking
  - Saving/Loading Models
  - Export Plain Old Java Object (POJO)

- Key Resources
  - docs.h2o.ai

- Joe's Previous H2O Talks
  - bit.ly/joe_h2o_talk3
  - bit.ly/h2o_budapest_1
  - bit.ly/h2o_paris_1
  - bit.ly/h2o_milan_1

Why H2O?


H2O.ai
Products
Industries
Customers
Community
Documentation
Download
Stories of AI Transformation

H2O In Action

www.h2o.ai/customers

Capital One

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

MarketShare

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

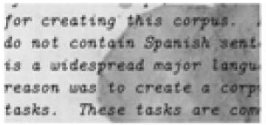
CIFAR-10 Competition Winners: Interviews with Dr. Ben Graham, Phil Culliton, & Zygmunt Zając

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



Completed • Knowledge • 161 teams

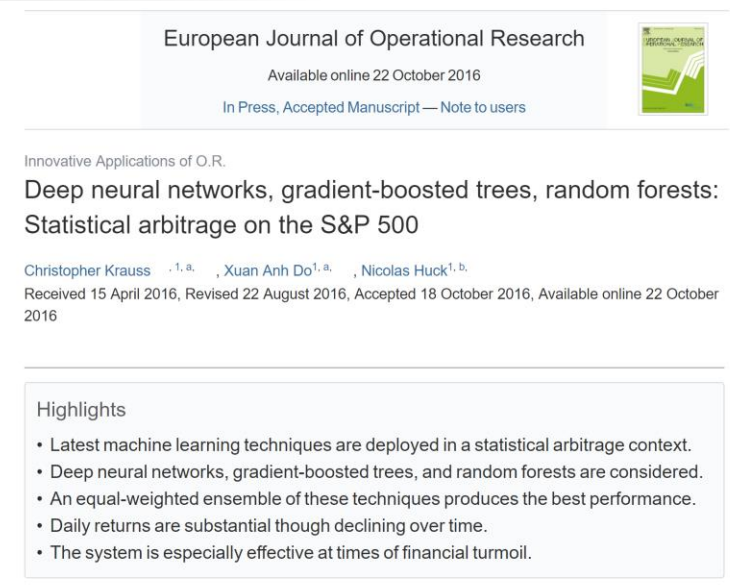
Denoising Dirty Documents

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

READ MORE

"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

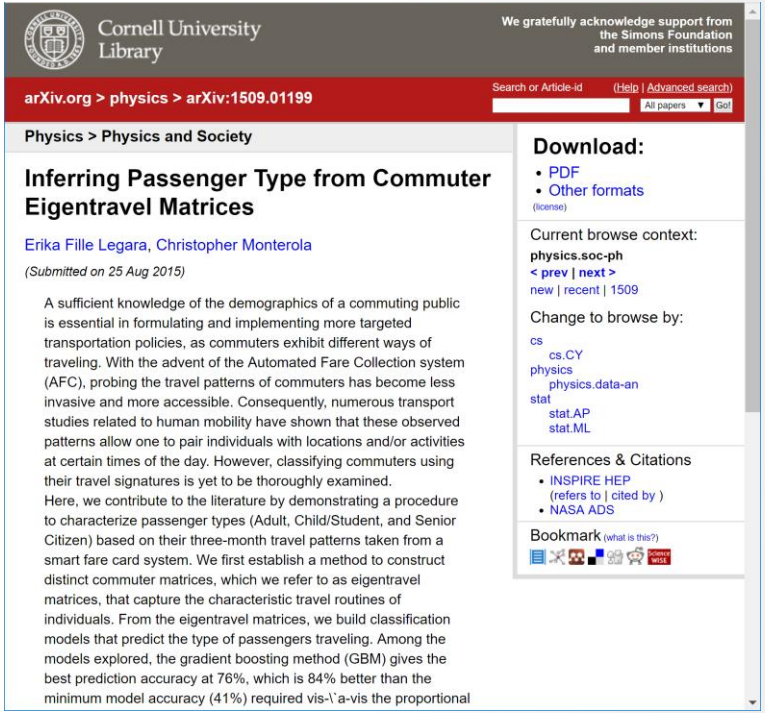
Christopher Krauss [1, a], Xuan Anh Do [1, a], Nicolas Huck [1, b]

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

Inferring Passenger Type from Commuter Eigentravel Matrices

Erika Fille Legara, Christopher Monterola

(Submitted on 25 Aug 2015)

A sufficient knowledge of the demographics of a commuting public is essential in formulating and implementing more targeted transportation policies, as commuters exhibit different ways of traveling. With the advent of the Automated Fare Collection system (AFC), probing the travel patterns of commuters has become less invasive and more accessible. Consequently, numerous transport studies related to human mobility have shown that these observed patterns allow one to pair individuals with locations and/or activities at certain times of the day. However, classifying commuters using their travel signatures is yet to be thoroughly examined. Here, we contribute to the literature by demonstrating a procedure to characterize passenger types (Adult, Child/Student, and Senior Citizen) based on their three-month travel patterns taken from a smart fare card system. We first establish a method to construct distinct commuter matrices, which we refer to as eigentravel matrices, that capture the characteristic travel routines of individuals. From the eigentravel matrices, we build classification models that predict the type of passengers traveling. Among the models explored, the gradient boosting method (GBM) gives the best prediction accuracy at 76%, which is 84% better than the minimum model accuracy (41%) required vis-\`a-vis the proportional

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

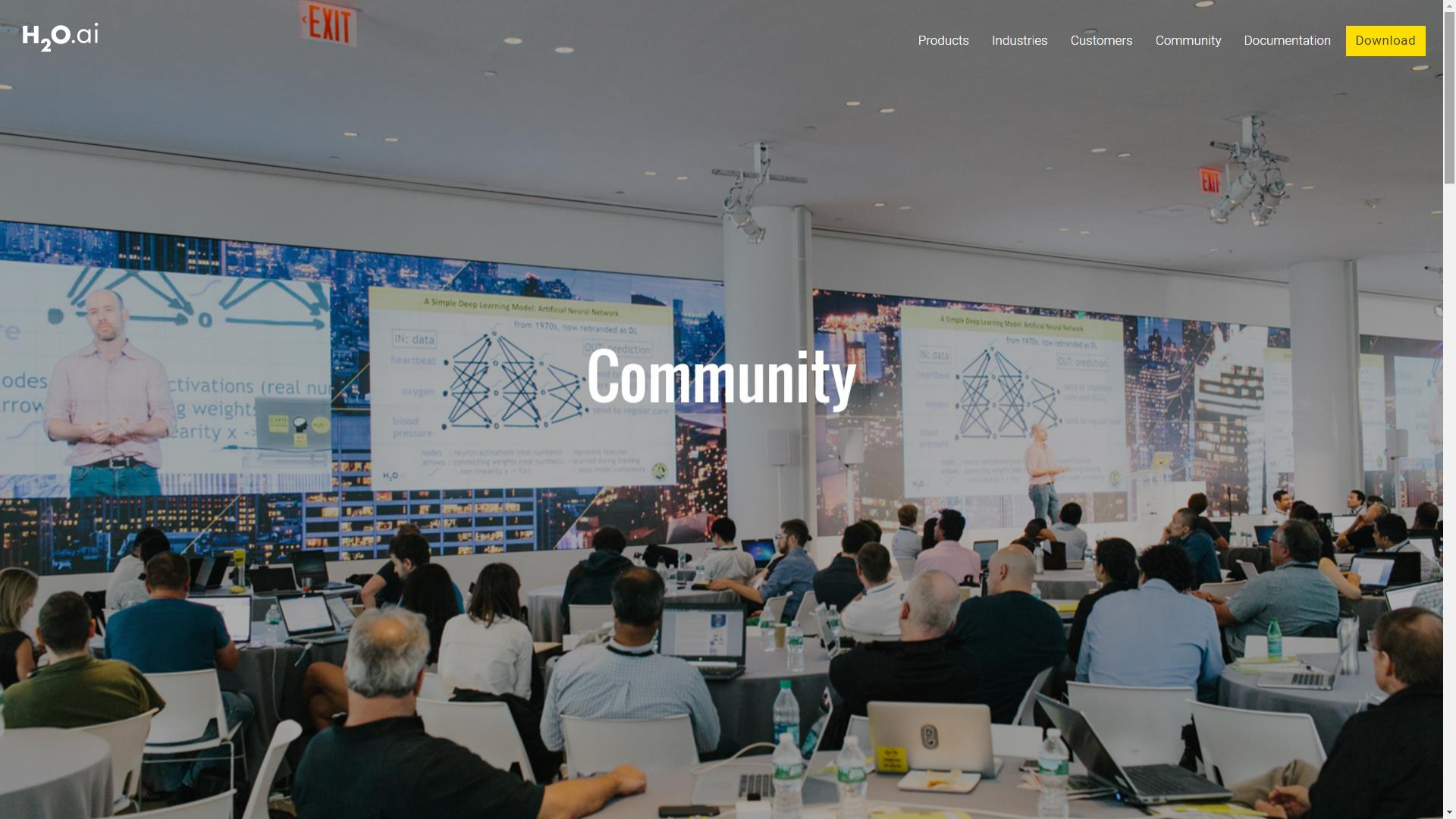
stat.ML

References & Citations

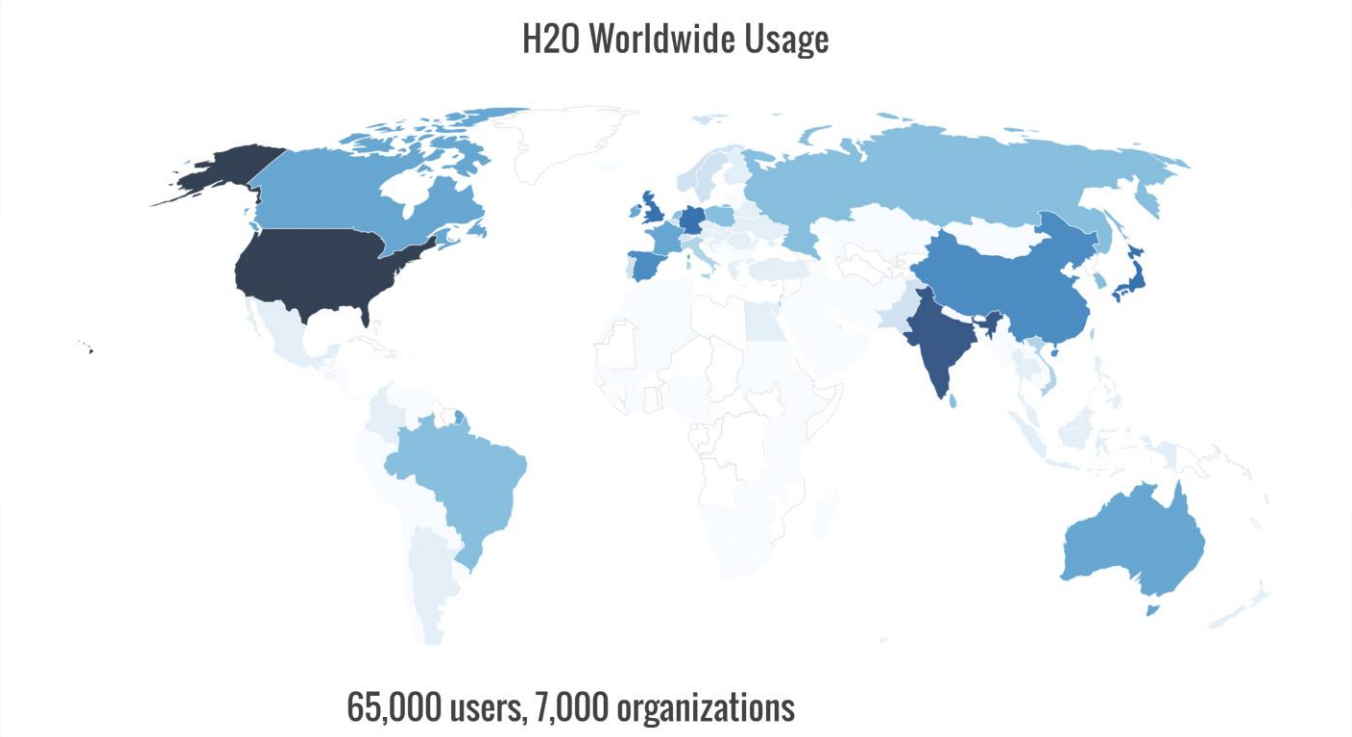
- INSPIRE HEP (refers to | cited by)
- NASA ADS

Bookmark (what is this?)

https://arxiv.org/abs/1509.01199


H2O.ai
Products
Industries
Customers
Community
Documentation
Download
EXIT
A Simple Deep Learning Model: Artificial Neural Network
from 1970s, now rebranded as DL
IN: data
heartbeat
oxygen
blood pressure
OUT: prediction
Community

H2O Worldwide Usage

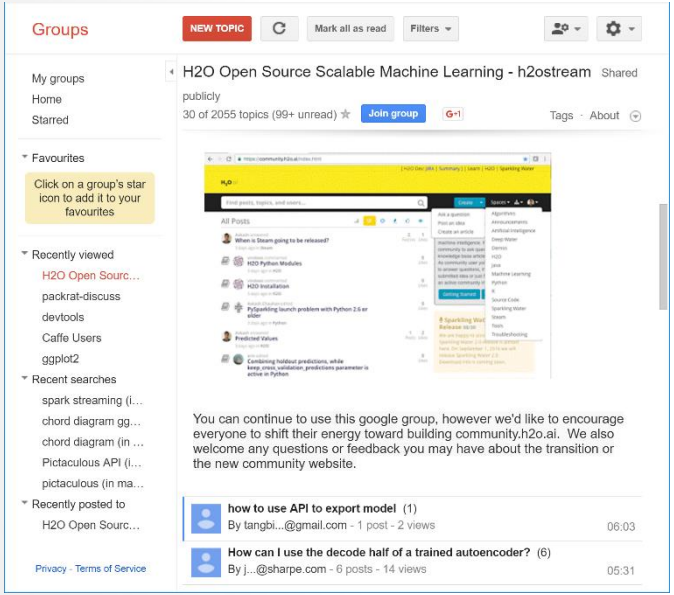


www.h2o.ai/community

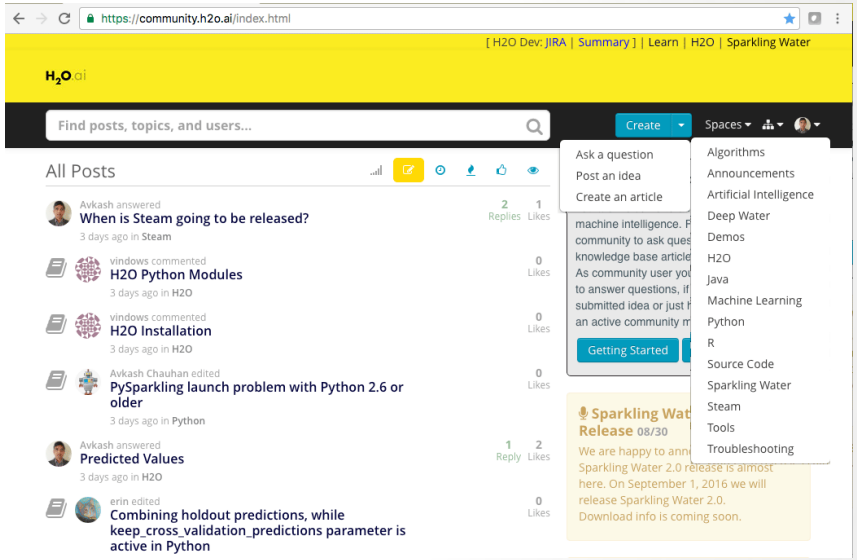
H2O Community Support

Please try

Google forum – h2osteam



community.h2o.ai



#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

What’s Next?

H2O is Evolving

- New Features
  - Advanced Data Munging
  - Deep Water
  - Auto ML
  - Steam
  - Sparkling Water
    - PySparkling
    - RSparkling (New)

- Find Out More
  - H2O YouTube Playlist
    - Open Tour New York
    - Open Tour Dallas
    - PyData DC 2016

H2O's Mission

Making Machine Learning Accessible to Everyone

Photo credit: Virgin Media

Thanks!

- Big Data London
- Data Science London
  - Meetup this evening
    - 6:30 pm
  - Deep Water
    - Motivation
    - Live Demo

- Resources
  - github.com/h2oai/h2o-meetups
  - www.h2o.ai
  - docs.h2o.ai
- Contact
  - joe@h2o.ai
  - @matlabulous

Extra Slides

(H2O Flow Demo Screenshots – just in case)

A Typical Machine Learning Task

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

Photo credit: http://www.opendeep.org/v0.0.5/docs/tutorial-classifying-handwritten-mnist-images

MNIST Hand-Written Digits

- 784 Inputs
  - 28 x 28 = 784 pixels
- 1 Output
  - 0, 1, 2, 3, 4, 5, 6, 7, 8 or 9
  - Classification
- Files
  - Train (60k Records)
  - Test (10k)
- Links
  - https://s3.amazonaws.com/h2o-public-test-data/bigdata/laptop/mnist/train.csv.gz
  - https://s3.amazonaws.com/h2o-public-test-data/bigdata/laptop/mnist/test.csv.gz

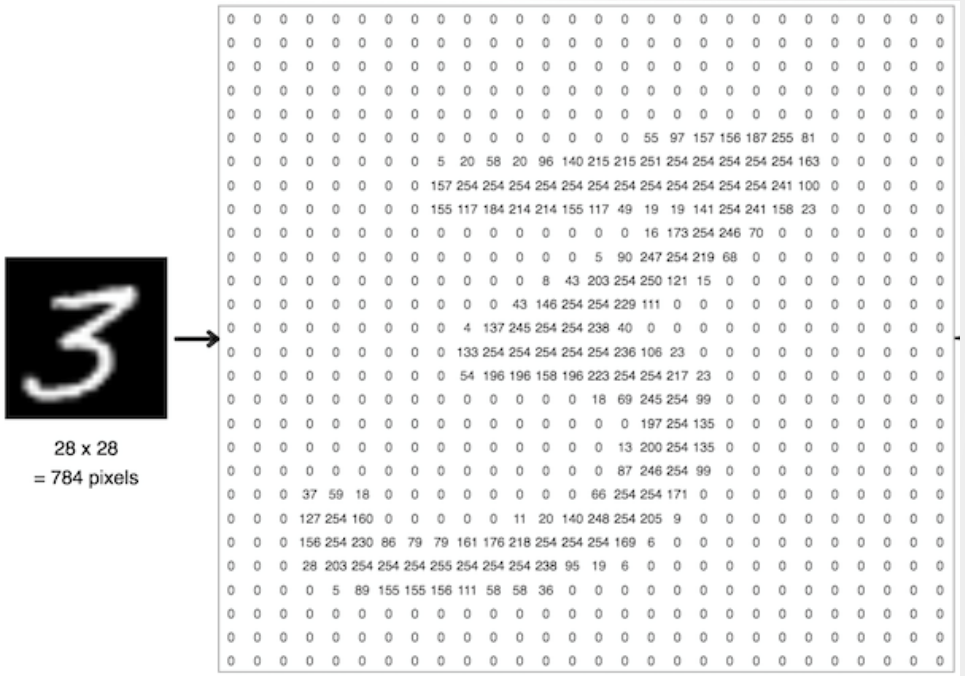
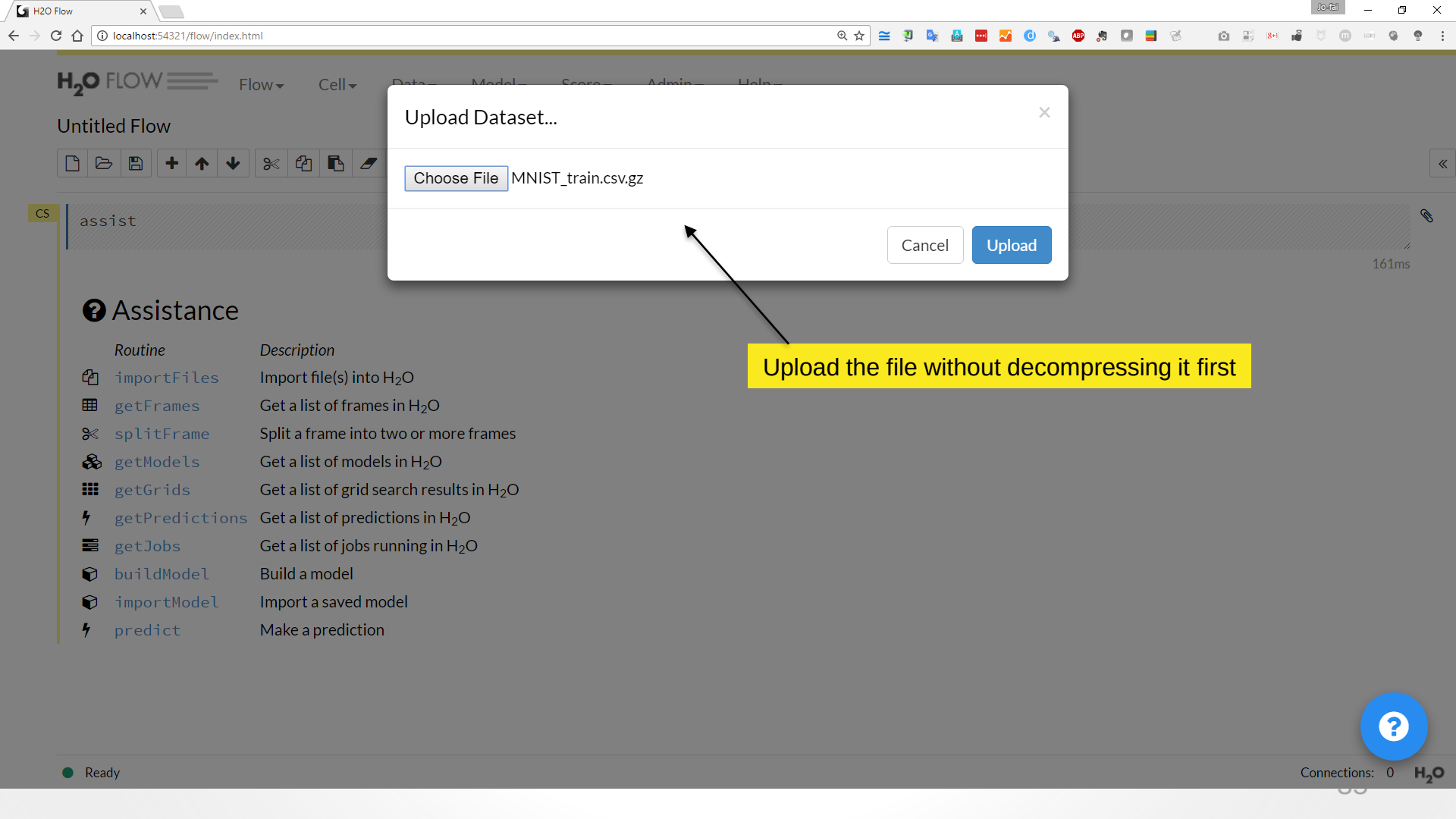
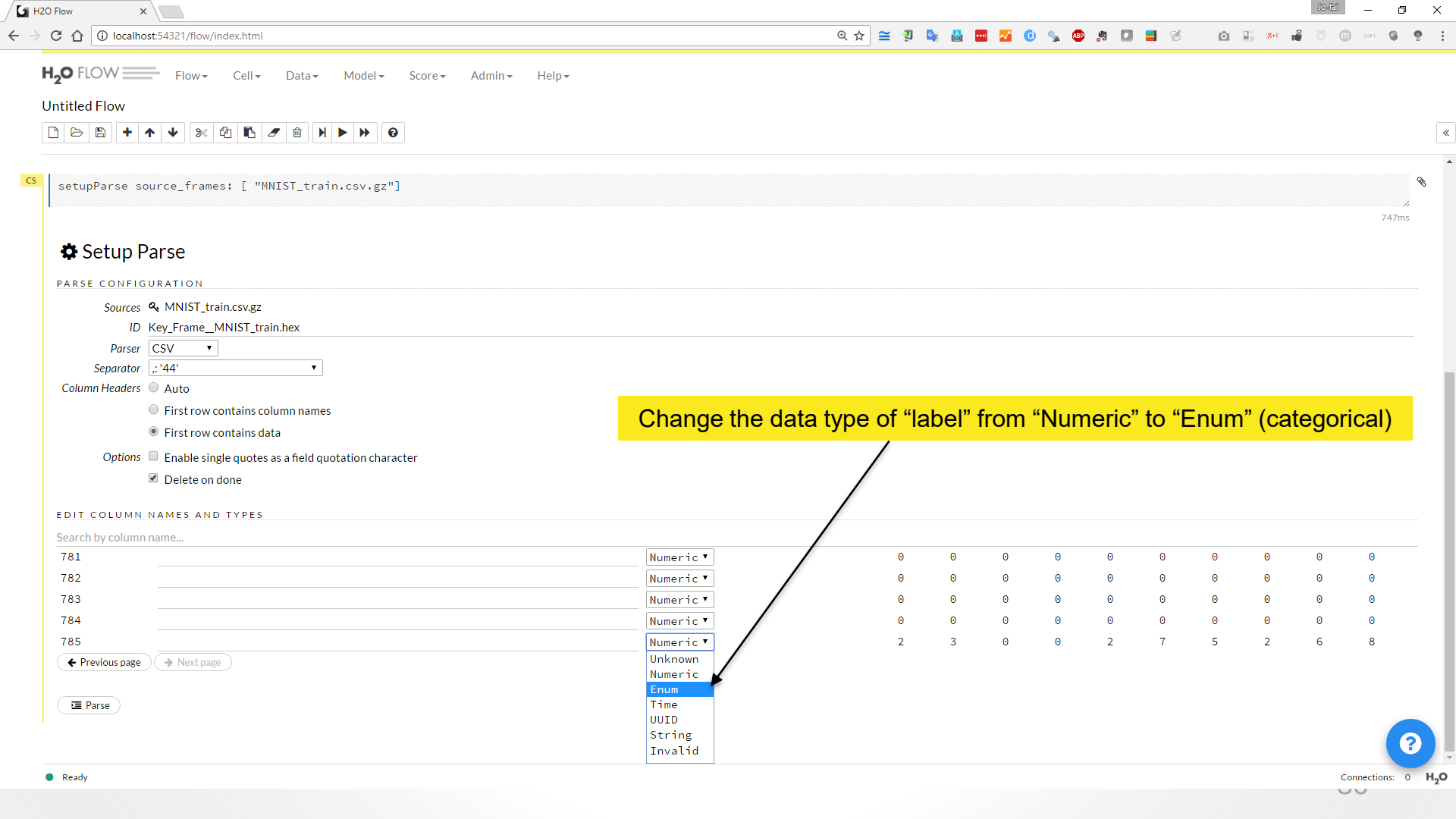


Photo credit: https://ml4a.github.io/ml4a/neural_networks/


H2O Flow
localhost:54321/flow/index.html
H2O FLOW
Flow
Cell
Untitled Flow
CS
assist
161ms
Upload Dataset...
Choose File
MNIST_train.csv.gz
Cancel
Upload
Upload the file without decompressing it first
Assistance
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
Ready
Connections: 0


H2O Flow
localhost:54321/flow/index.html
H2O FLOW
Flow
Cell
Data
Model
Score
Admin
Help
Untitled Flow
CS
setupParse source_frames: ["MNIST_train.csv.gz"]
747ms
Setup Parse
PARSE CONFIGURATION
Sources MNIST_train.csv.gz
ID Key_Frame__MNIST_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
Change the data type of “label” from “Numeric” to “Enum” (categorical)
EDIT COLUMN NAMES AND TYPES
Search by column name...
781 Numeric 0 0 0 0 0 0 0 0 0 0
782 Numeric 0 0 0 0 0 0 0 0 0 0
783 Numeric 0 0 0 0 0 0 0 0 0 0
784 Numeric 0 0 0 0 0 0 0 0 0 0
785 Numeric 2 3 0 0 2 7 5 2 6 8
Unknown
Numeric
Enum
Time
UUID
String
Invalid
Previous page
Next page
Parse
Ready
Connections: 0

Untitled Flow

CS

```
getFrameSummary "Key_Frame__MNIST_train.hex"
```

268ms

Key_Frame__MNIST_train.hex

Note: Size in Memory

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

Rows	Columns	Compressed Size
60000	785	21MB

COLUMN SUMMARIES

label	type	Missing	Zeros	+Inf	-Inf	min	max	mean	sigma	cardinality	Actions
C781	int	0	60000	0	0	0	0	0	0	·	Convert to enum
C782	int	0	60000	0	0	0	0	0	0	·	Convert to enum
C783	int	0	60000	0	0	0	0	0	0	·	Convert to enum
C784	int	0	60000	0	0	0	0	0	0	·	Convert to enum
C785	enum	0	5923	0	0	0	9.0	·	·	10	Convert to numeric

Previous 20 Columns | Next 20 Columns

Click on individual labels to explore data

CHUNK COMPRESSION SUMMARY

FRAME DISTRIBUTION SUMMARY

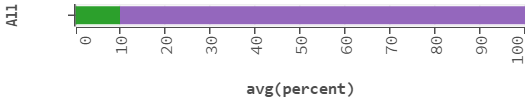
Untitled Flow

154ms

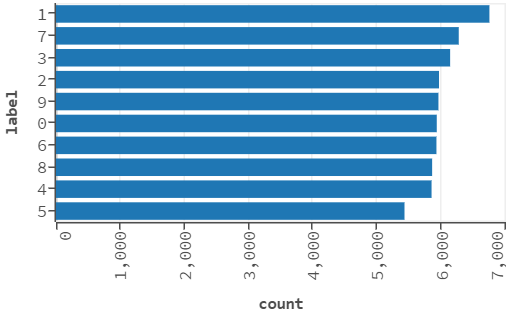
Summary: C785

Actions Impute Inspect

CHARACTERISTICS



DOMAIN (MAX 1000 LEVELS)



Untitled Flow

```
splitFrame
```

21ms

Split Frame

Frame: Key_Frame__MNIST_train.hex

Splits:	*Ratio*	*Key*
	0.8	train
	0.20	valid

Add a new split

Seed: 1234

Create

Split the full dataset into training (80% = 48k records) and validation (20% = 12k) – a common machine learning practice

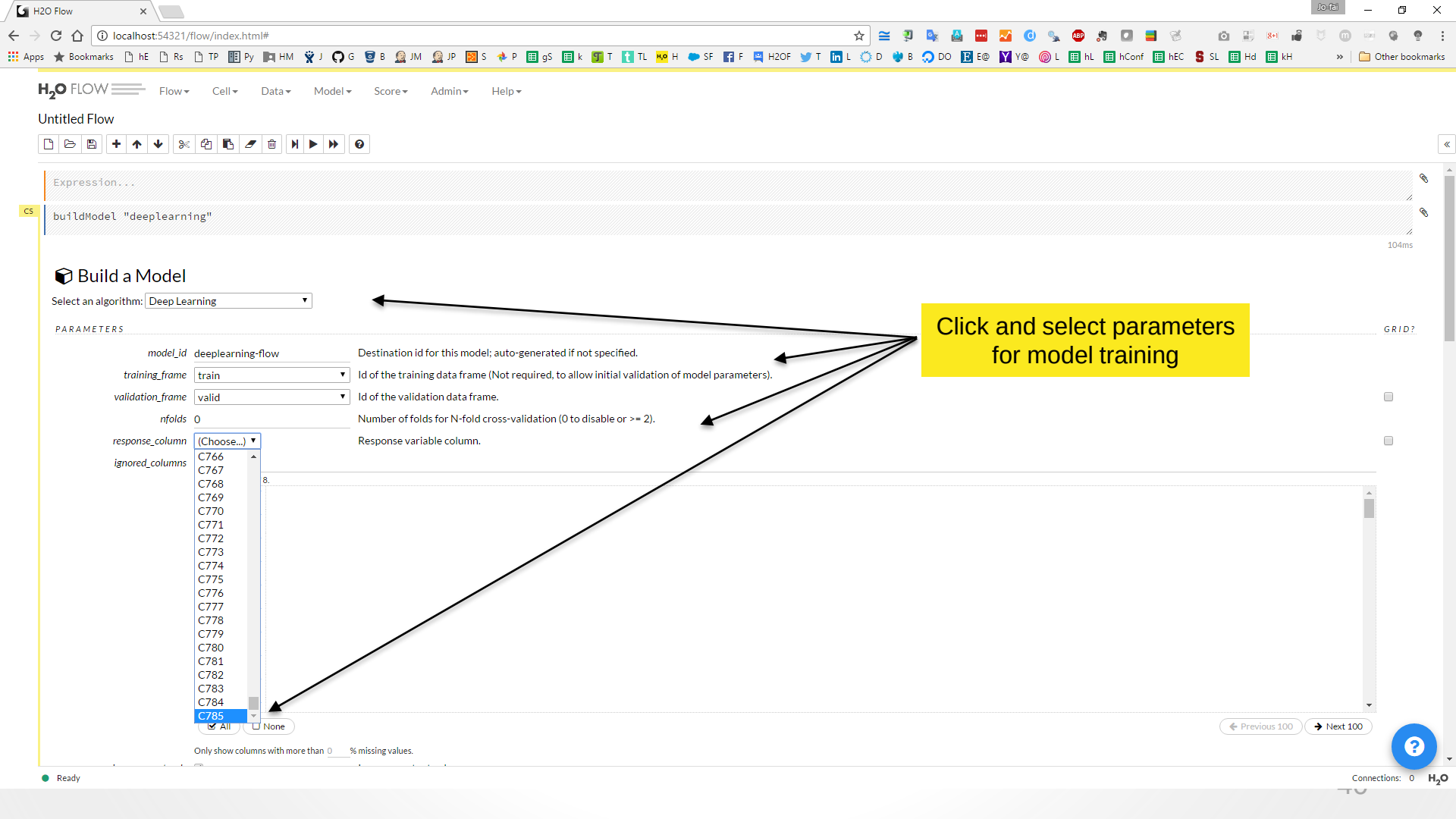
CS

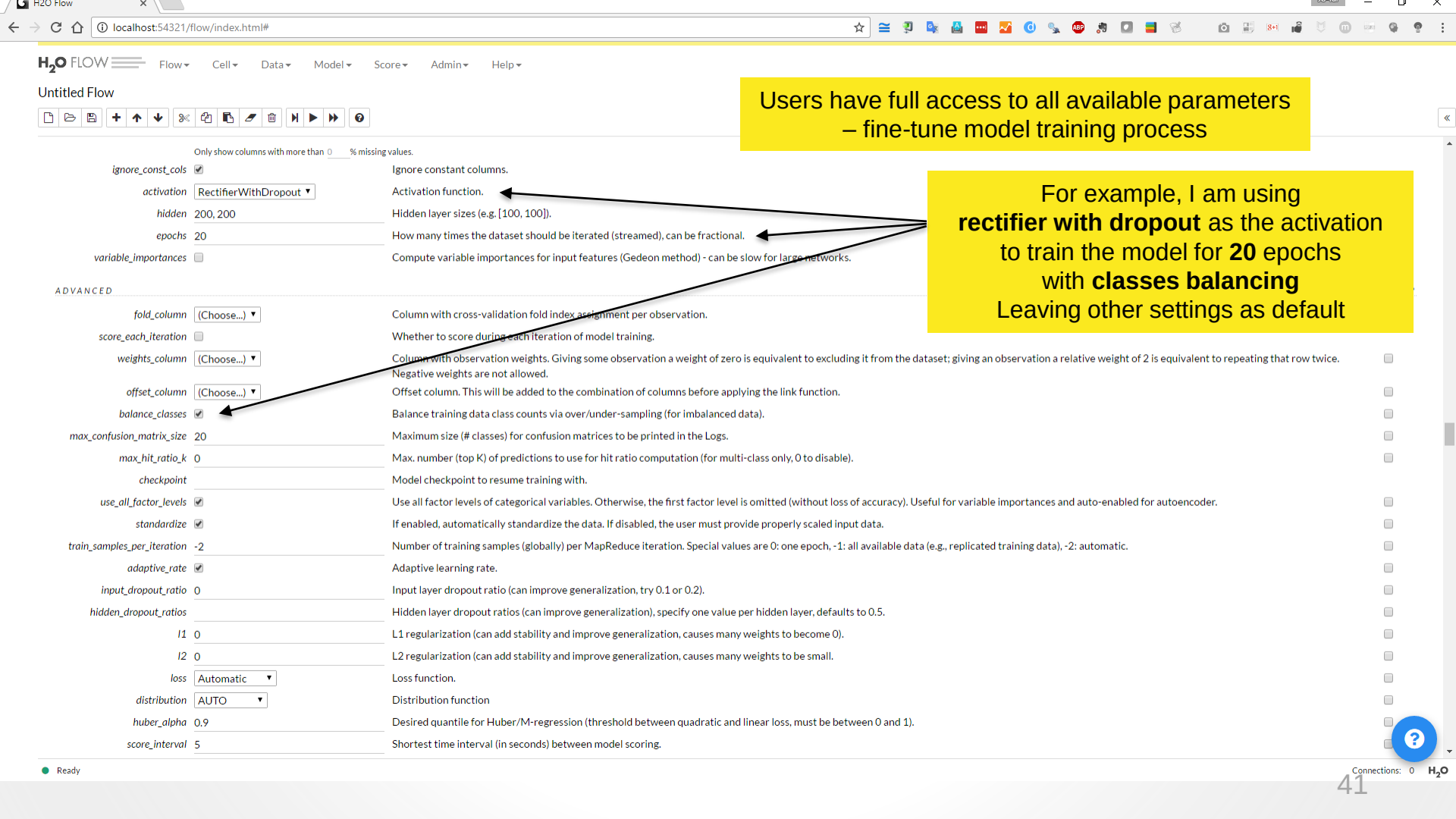
```
splitFrame "Key_Frame__MNIST_train.hex", [0.8], ["train","valid"], 1234
```

815ms

Split Frames

Type	*Key*	*Ratio*
	train	0.8
	valid	0.19999999999999996


H2O Flow
localhost:54321/flow/index.html#
Flow
Cell
Data
Model
Score
Admin
Help
Untitled Flow
Expression...
buildModel "deeplearning"
104ms
Build a Model
Select an algorithm: Deep Learning
PARAMETERS
GRID?
model_id
deeplearning-flow
Destination id for this model; auto-generated if not specified.
training_frame
train
Id of the training data frame (Not required, to allow initial validation of model parameters).
validation_frame
valid
Id of the validation data frame.
nfolds
0
Number of folds for N-fold cross-validation (0 to disable or >= 2).
response_column
(Choose...)
Response variable column.
ignored_columns
C766
C767
C768
C769
C770
C771
C772
C773
C774
C775
C776
C777
C778
C779
C780
C781
C782
C783
C784
C785
All
None
Previous 100
Next 100
Only show columns with more than 0 % missing values.
Click and select parameters for model training
Ready
Connections: 0


H2O Flow
localhost:54321/flow/index.html#
H2O FLOW
Flow
Cell
Data
Model
Score
Admin
Help
Untitled Flow
Users have full access to all available parameters – fine-tune model training process
For example, I am using **rectifier with dropout** as the activation to train the model for **20** epochs with **classes balancing** Leaving other settings as default
Only show columns with more than 0 % missing values.
ignore_const_cols
Ignore constant columns.
activation
RectifierWithDropout
Activation function.
hidden
200, 200
Hidden layer sizes (e.g. [100, 100]).
epochs
20
How many times the dataset should be iterated (streamed), can be fractional.
variable_importances
Compute variable importances for input features (Gedeon method) - can be slow for large networks.
ADVANCED
fold_column
(Choose...)
Column with cross-validation fold index assignment per observation.
score_each_iteration
Whether to score during each iteration of model training.
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.
offset_column
(Choose...)
Offset column. This will be added to the combination of columns before applying the link function.
balance_classes
Balance training data class counts via over/under-sampling (for imbalanced data).
max_confusion_matrix_size
20
Maximum size (# classes) for confusion matrices to be printed in the Logs.
max_hit_ratio_k
0
Max. number (top K) of predictions to use for hit ratio computation (for multi-class only, 0 to disable).
checkpoint
Model checkpoint to resume training with.
use_all_factor_levels
Use all factor levels of categorical variables. Otherwise, the first factor level is omitted (without loss of accuracy). Useful for variable importances and auto-enabled for autoencoder.
standardize
If enabled, automatically standardize the data. If disabled, the user must provide properly scaled input data.
train_samples_per_iteration
-2
Number of training samples (globally) per MapReduce iteration. Special values are 0: one epoch, -1: all available data (e.g., replicated training data), -2: automatic.
adaptive_rate
Adaptive learning rate.
input_dropout_ratio
0
Input layer dropout ratio (can improve generalization, try 0.1 or 0.2).
hidden_dropout_ratios
Hidden layer dropout ratios (can improve generalization), specify one value per hidden layer, defaults to 0.5.
l1
0
L1 regularization (can add stability and improve generalization, causes many weights to become 0).
l2
0
L2 regularization (can add stability and improve generalization, causes many weights to be small.
loss
Automatic
Loss function.
distribution
AUTO
Distribution function
huber_alpha
0.9
Desired quantile for Huber/M-regression (threshold between quadratic and linear loss, must be between 0 and 1).
score_interval
5
Shortest time interval (in seconds) between model scoring.
Ready
Connections: 0
H2O
41

Untitled Flow

max_categorical_features	2147483647	Max. number of categorical features, enforced via hashing. #Experimental
reproducible	☐	Force reproducibility on small data (will be slow - only uses 1 thread).
export_weights_and_biases	☐	Whether to export Neural Network weights and biases to H2O Frames.
mini_batch_size	1	Mini-batch size (smaller leads to better fit, larger can speed up and generalize better).
elastic_averaging	☐	Elastic averaging between compute nodes can improve distributed model convergence. #Experimental

Build Model

CS


```
buildModel 'deeplearning', {"model_id":"deeplearning-flow","training_frame":"train","validation_frame":"valid","nfolds":0,"response_column":"C785","ignored_columns":
[],"ignore_const_cols":true,"activation":"RectifierWithDropout","hidden":
[200,200],"epochs":"20","variable_importances":false,"score_each_iteration":false,"balance_classes":true,"max_confusion_matrix_size":20,"max_hit_ratio_k":0,"checkpoint":"","use_all_factor_levels":true,"standardize":true,"train_samples_per_iteration":-2,"adaptive_rate":true,"input_dropout_ratio":0,"hidden_dropout_ratios":
[],"l1":0,"l2":0,"loss":"Automatic","distribution":"AUTO","huber_alpha":0.9,"score_interval":5,"score_training_samples":10000,"score_validation_samples":0,"score_duty_cycle":0.1,"stopping_rounds":5,"stopping_metric":"AUTO","stopping_tolerance":0,"max_runtime_secs":0,"autoencoder":false,"categorical_encoding":"AUTO","class_sampling_factors":
[],"max_after_balance_size":5,"pretrained_autoencoder":"","overwrite_with_best_model":true,"target_ratio_comm_to_comp":0.05,"seed":-1,"rho":0.99,"epsilon":1e-8,"nesterov_accelerated_gradient":true,"max_w2":"Infinity","initial_weight_distribution":"UniformAdaptive","classification_stop":0,"score_validation_sampling":"Uniform","diagnostics":true,"fast_mode":true,"force_load_balance":true,"single_node_mode":false,"shuffle_training_data":false,"missing_values_handling":"MeanImputation","quiet_mode":false,"sparse":false,"col_major":false,"average_activation":0,"sparsity_beta":0,"max_categorical_features":2147483647,"reproducible":false,"export_weights_and_biases":false,"mini_batch_size":1,"elastic_averaging":false}
```


Started at 10:40:35 am

Job

Run Time	00:01:04.377
Remaining Time	00:02:45.486
Type	Model
Key	deeplearning-flow
Description	DeepLearning
Status	RUNNING
Progress	29%

Iterations: 5. Epochs: 5.00000. Speed: 5,553 samples/sec. Estimated time left: 3 min 10.601 sec

Actions View Cancel Job

Training the model with estimated remaining time – users can stop the process early if they want to

H2O Flow

localhost:54321/flow/index.html#

H2O FLOW Flow Cell Data Model Score Admin Help

Untitled Flow

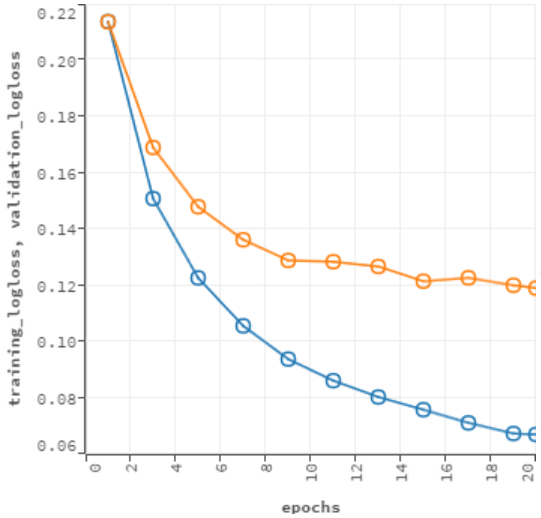
Model ID: deeplearning-flow

Algorithm: Deep Learning

Actions: Refresh | Predict... | Download POJO | Export | Inspect | Delete

▸ MODEL PARAMETERS

▾ SCORING HISTORY - LOGLOSS



Performance (logloss) on validation set

Performance (logloss) on training set

Ready

Connections: 0 H2O

Untitled Flow

TRAINING METRICS - CONFUSION MATRIX VERTICAL: ACTUAL; ACROSS: PREDICTED

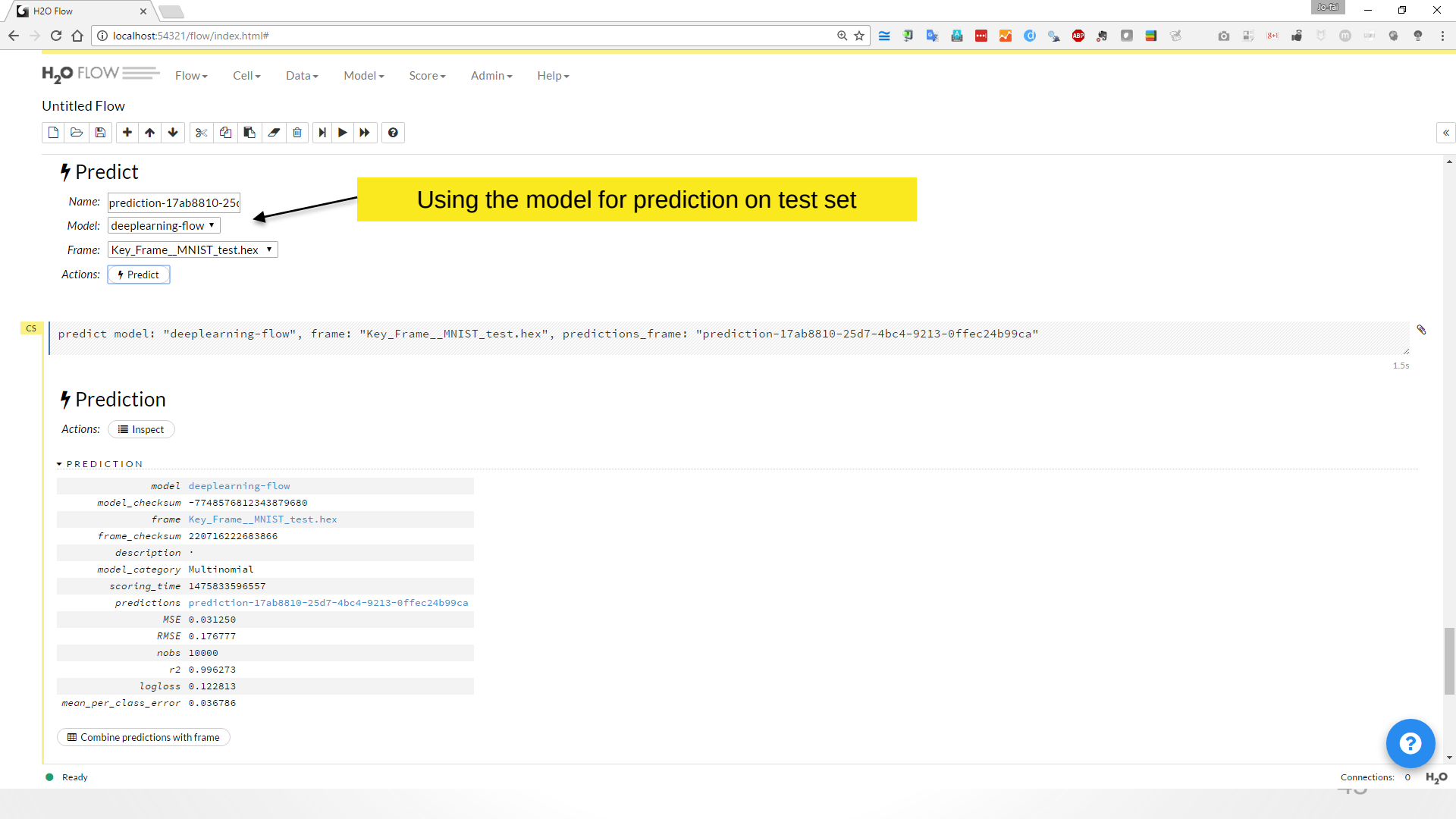
	0	1	2	3	4	5	6	7	8	9	Error	Rate
0	981	0	0	0	0	0	5	3	2	0	0.0101	10 / 991
1	0	966	4	2	0	0	4	2	1	0	0.0133	13 / 979
2	4	0	997	0	3	0	11	7	0	0	0.0245	25 / 1,022
3	0	0	10	928	0	4	7	12	3	2	0.0393	38 / 966
4	0	0	3	0	955	0	6	2	1	2	0.0144	14 / 969
5	0	0	2	1	0	919	19	1	1	2	0.0275	26 / 945
6	2	0	0	0	2	0	996	0	2	0	0.0060	6 / 1,002
7	0	1	2	1	0	1	6	1033	1	1	0.0124	13 / 1,046
8	1	4	6	4	0	8	9	3	984	3	0.0372	38 / 1,022
9	2	0	2	1	2	2	5	17	0	963	0.0312	31 / 994
Total	990	971	1026	937	962	934	1068	1080	995	973	0.0215	214 / 9,936

Confusion Matrix on Training Set (48k Records)
About 2% Error

VALIDATION METRICS - CONFUSION MATRIX VERTICAL: ACTUAL; ACROSS: PREDICTED

	0	1	2	3	4	5	6	7	8	9	Error	Rate
0	1175	0	5	4	0	2	14	2	2	0	0.0241	29 / 1,204
1	0	1320	9	5	2	1	5	7	2	0	0.0229	31 / 1,351
2	3	0	1107	5	1	1	26	9	4	1	0.0432	50 / 1,157
3	2	2	17	1179	0	25	8	17	8	4	0.0658	83 / 1,262
4	2	3	5	0	1092	0	13	5	3	19	0.0438	50 / 1,142
5	3	0	1	8	1	1073	17	3	3	4	0.0359	40 / 1,113
6	7	0	3	0	2	3	1139	0	3	0	0.0156	18 / 1,157
7	2	4	5	3	1	0	6	1247	1	5	0.0212	27 / 1,274
8	1	9	8	11	3	10	7	6	1092	8	0.0545	63 / 1,155
9	0	1	2	5	16	5	4	41	6	1096	0.0680	80 / 1,176
Total	1195	1339	1162	1220	1118	1120	1239	1337	1124	1137	0.0393	471 / 11,991

Confusion Matrix on Validation Set (12k Records)
About 4% Error


H2O Flow
localhost:54321/flow/index.html#
H2O FLOW
Flow
Cell
Data
Model
Score
Admin
Help
Untitled Flow
Predict
Name: prediction-17ab8810-25(
Model: deeplearning-flow
Frame: Key_Frame__MNIST_test.hex
Actions: Predict
Using the model for prediction on test set
CS
predict model: "deeplearning-flow", frame: "Key_Frame__MNIST_test.hex", predictions_frame: "prediction-17ab8810-25d7-4bc4-9213-0ffec24b99ca"
1.5s
Prediction
Actions: Inspect
PREDICTION
model deeplearning-flow
model_checksum -7748576812343879680
frame Key_Frame__MNIST_test.hex
frame_checksum 220716222683866
description ·
model_category Multinomial
scoring_time 1475833596557
predictions prediction-17ab8810-25d7-4bc4-9213-0ffec24b99ca
MSE 0.031250
RMSE 0.176777
nobs 10000
r2 0.996273
logloss 0.122813
mean_per_class_error 0.036786
Combine predictions with frame
Ready
Connections: 0

Untitled Flow

PREDICTION - CM - CONFUSION MATRIX

0	1	2	3	4	5	6	7	8	9	Error	Rate
962	0	1	2	0	2	10	1	1	1	0.0184	18 / 980
0	1121	6	1	0	0	3	1	3	0	0.0123	14 / 1,135
6	0	988	2	3	1	15	14	3	0	0.0426	44 / 1,032
0	0	10	971	0	6	7	11	3	2	0.0386	39 / 1,010
0	0	6	0	944	1	12	4	3	12	0.0387	38 / 982
3	0	2	9	2	849	18	1	5	3	0.0482	43 / 892
5	3	1	0	3	5	936	0	5	0	0.0230	22 / 958
1	4	14	1	1	0	2	1000	2	3	0.0272	28 / 1,028
5	1	4	6	7	9	9	7	922	4	0.0534	52 / 974
2	4	0	7	12	3	7	28	3	943	0.0654	66 / 1,009
984	1133	1032	999	972	876	1019	1067	950	968	0.0364	364 / 10,000

Confusion Matrix on Test Set (10k Records)
About 4% Error (similar to validation)

Untitled Flow

prediction-17ab8810-25d7-4bc4-9213-0ffec24b99ca

Full prediction outputs including individual probabilities and predicted label

DATA

Previous 20 Columns | Next 20 Columns

Row	predict	p0	p1	p2	p3	p4	p5	p6	p7	p8	p9
1	8	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	1.0	0.0
2	3	0.0	0.0	0.0	1.0	0.0	0.0	0.0	0.0	0.0	0.0
3	8	0.1102	0.0	0.0153	0.0039	0.0089	0.0461	0.1479	0.0	0.6660	0.0016
4	0	0.9990	0.0	0.0	0.0	0.0	0.0	0.0010	0.0	0.0	0.0
5	1	0.0	1.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
6	5	0.0	0.0	0.0	0.0	0.0	0.9594	0.0	0.0	0.0406	0.0
7	0	1.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
8	1	0.0	1.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
9	5	0.0	0.0	0.0	0.0092	0.0	0.9873	0.0	0.0	0.0002	0.0033
10	2	0.0	0.0003	0.9851	0.0011	0.0	0.0	0.0	0.0011	0.0123	0.0
11	4	0.0	0.0	0.0	0.0	1.0	0.0	0.0	0.0	0.0	0.0
12	8	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	1.0	0.0
13	1	0.0	0.9999	0.0	0.0	0.0	0.0	0.0	0.0	0.0001	0.0
14	4	0.0	0.0	0.0	0.0	1.0	0.0	0.0	0.0	0.0	0.0
15	3	0.0	0.0	0.0001	0.9983	0.0	0.0	0.0	0.0	0.0016	0.0
16	9	0.0	0.0	0.0	0.0002	0.0	0.0	0.0	0.0	0.0	0.9997
17	6	0.0	0.0	0.0	0.0	0.0	0.0	1.0	0.0	0.0	0.0

H2O Deep Learning beats MNIST

```
> install.packages("h2o")
> library(h2o)
> h2oServer <- h2o.init(ip="mr-0xd1", port=53322)
> train_hex <- h2o.importFile(h2oServer, "./train.csv.gz")
> test_hex  <- h2o.importFile(h2oServer, "./test.csv.gz")
> record_model <- h2o.deeplearning(x = c(1:784), y = 785, data = train_hex, validation = test_hex,
                                   activation = "RectifierWithDropout", hidden = c(1024,1024,2048),
                                   epochs = 2000, l1 = 1e-5, input_dropout_ratio = 0.2,
                                   train_samples_per_iteration = -1, classification_stop = -1)
|=================================================================================================
=================================================================================================| 100%
> record_model@model$confusion
```

	Predicted										
Actual	0	1	2	3	4	5	6	7	8	9	Error
0	975	1	1	0	0	0	1	1	1	0	0.00510
1	0	1135	0	0	0	0	0	0	0	0	0.00000
2	0	1	1028	0	1	0	0	2	0	0	0.00388
3	0	0	1	1001	0	1	0	3	2	2	0.00891
4	0	0	2	0	971	0	3	1	0	5	0.01120
5	2	0	0	6	0	881	1	1	1	0	0.01233
6	2	3	0	1	1	1	950	0	0	0	0.00835
7	1	2	6	0	0	0	0	1017	0	2	0.01070
8	1	0	3	4	0	1	1	2	958	3	0.01540
9	1	2	0	1	3	3	0	3	0	988	0.01288
Totals	982	1145	1041	1013	976	887	956	1030	961	997	0.00870

Standard 60k/10k data
No distortions
No convolutions
No unsupervised training
No ensemble

4 hours on 4 16-core nodes

World-class result!

0.87% test set error