# First-time Qwiklab Account Setup

- Go to https://h2oai.qwiklab.com/
- Click on “JOIN”
- Create a new account with a valid email address
- You will receive a confirmation email
  - Click on the link in the confirmation email
- Go back to http://h2oai.qwiklab.com and log in
- Go to the Catalog on the left bar
- Choose “H2O and Sparking Water Workshop”
- Wait for instructions

# H2O-3 and Sparkling Water Workshop

Megan Kurka / Tom Kraljevic
H2O.ai

# Agenda

- Introduction to H2O-3
- Hands On
  - Data Cleaning and Supervised Learning
- Introduction to Sparkling Water
- Hands On
  - Feature Engineering for Time Series and AutoML
- Questions and Answers

# H2O Overview

# H2O Products

$H_2O$.ai

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

Spark + $H_2O$ SPARKLING WATER

H2O AI Open Source Engine Integration with Spark

$H_2O$4GPU

Lightning Fast machine learning on GPUs

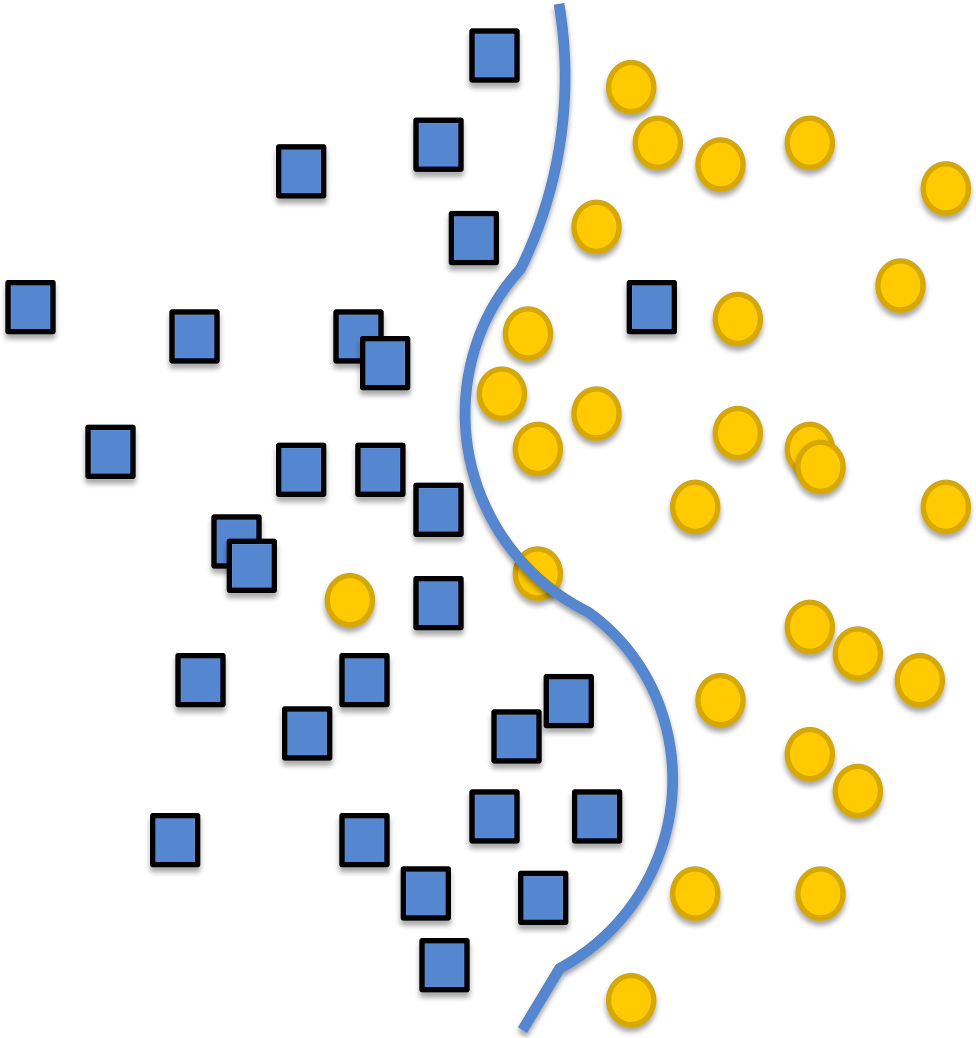
DRIVERLESSAI

Automatic feature engineering, machine learning and interpretability

Steam

Secure multi-tenant H2O clusters

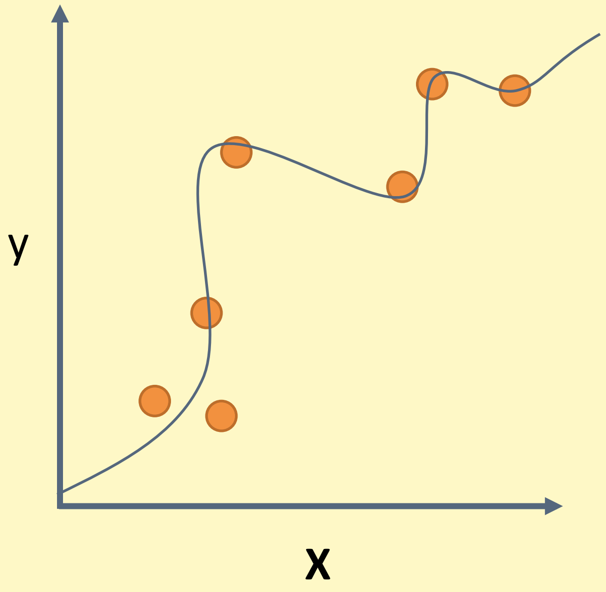
# What is Machine Learning?



# Supervised Learning

**Regression:**

How much will a customer spend?

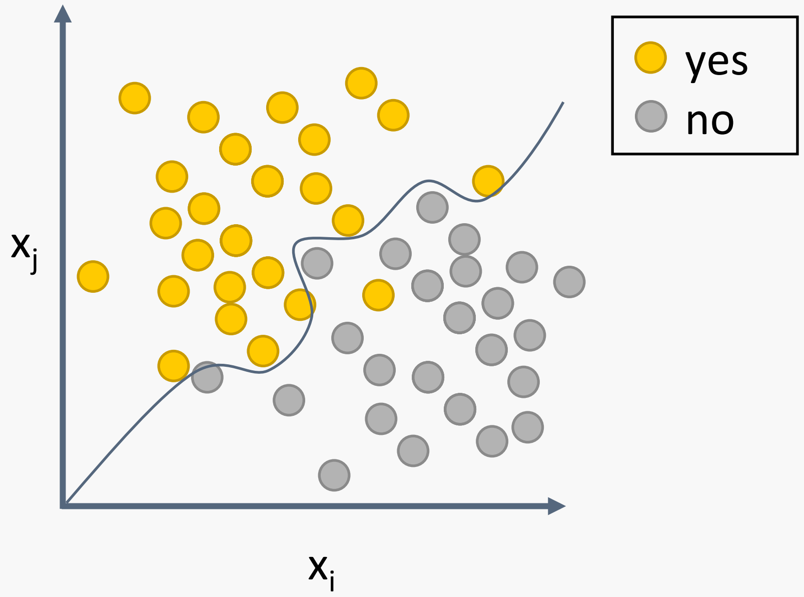


**$H_2O$ algos:**

- Penalized Linear Models
- Random Forest
- Gradient Boosting
- XGBoost
- Neural Networks
- Stacked Ensembles

**Classification:**

Will a customer churn?



**$H_2O$ algos:**

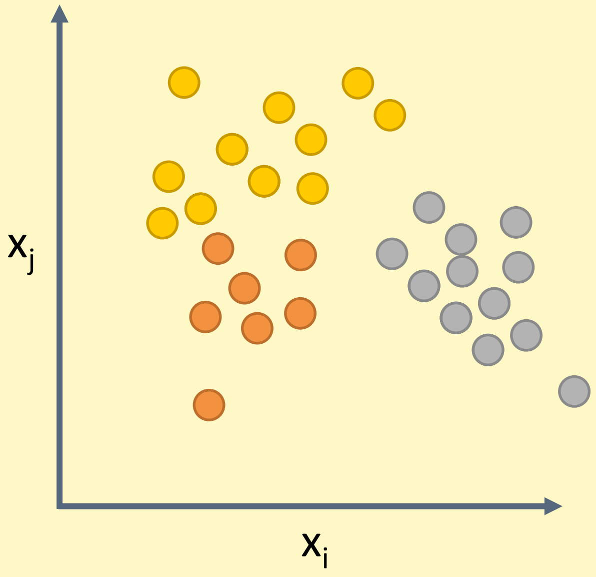
- Penalized Linear Models
- Naïve Bayes
- Random Forest
- Gradient Boosting
- XGBoost
- Neural Networks
- Stacked Ensembles

H2O.ai

# Unsupervised Learning

**Clustering:**

Grouping rows – e.g. creating groups of similar customers

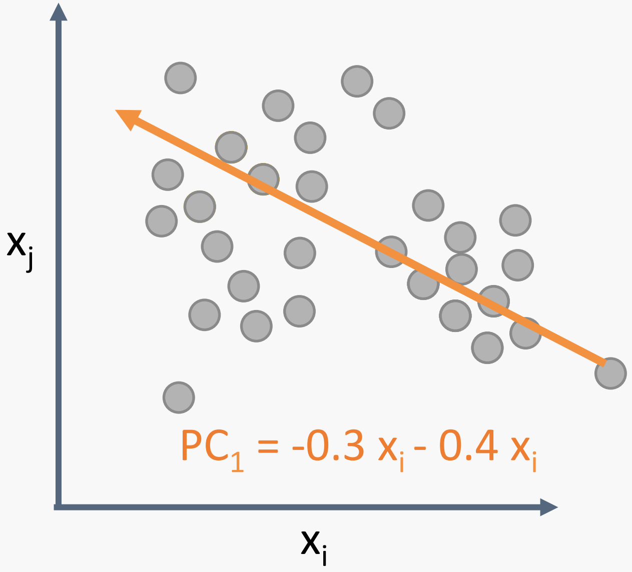


**$H_2O$ algos:**

k – means

**Feature extraction:**

Grouping columns – Create a small number of new representative dimensions



**$H_2O$ algos:**

Principal components

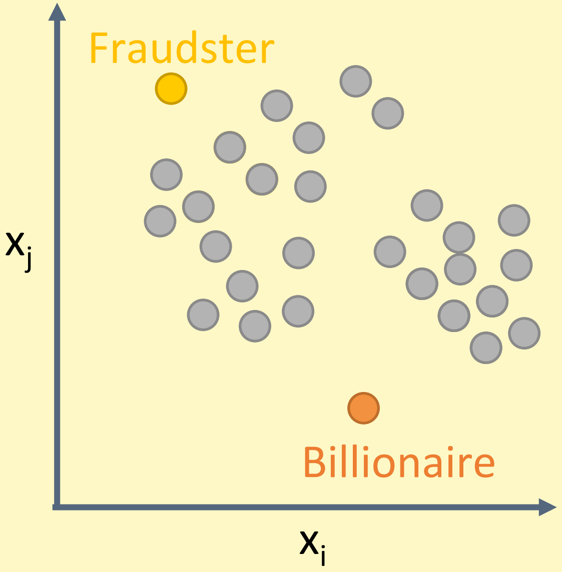
Generalized low rank models

Autoencoders

Word2Vec

**Anomaly detection:**

Detecting outlying rows - Finding high-value, fraudulent, or weird customers



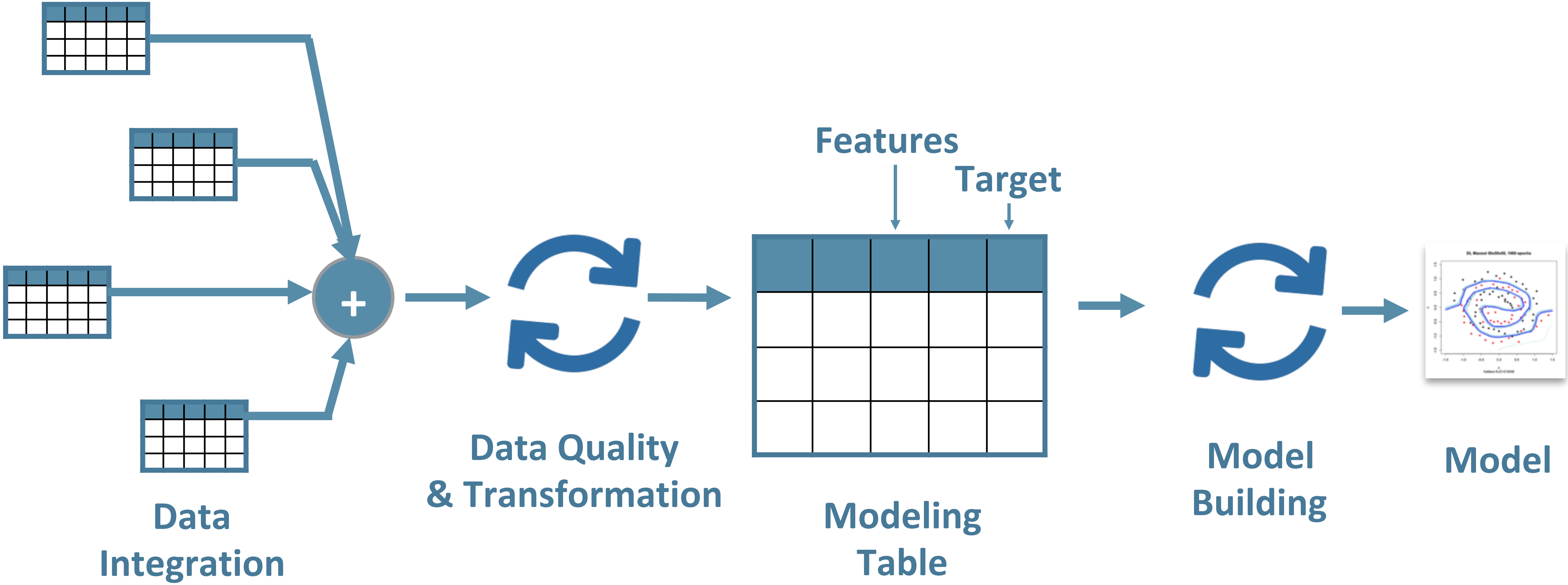
**$H_2O$ algos:**

Principal components

Generalized low rank models

Autoencoders

# Simplified Typical Machine Learning Pipeline



Data Integration

Data Quality & Transformation

Features

Target

Modeling Table

Model Building

Model

# H2O-3

# What is H2O?

## Math Platform — Open source in-memory AI engine

- Parallelized and distributed algorithms
- GLM, Random Forest, GBM, Deep Learning, etc.

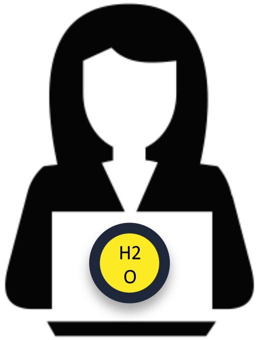
## Tech and API — Easy to use and adopt

- Written in Java – perfect for Java Programmers
- Install is lightweight
- REST API (Java) – run H2O from R, Python, WebUI

## Big Data — More data? Or better models? BOTH

- Use all of your data – model without sampling
- More Data + Better Models = Better Predictions

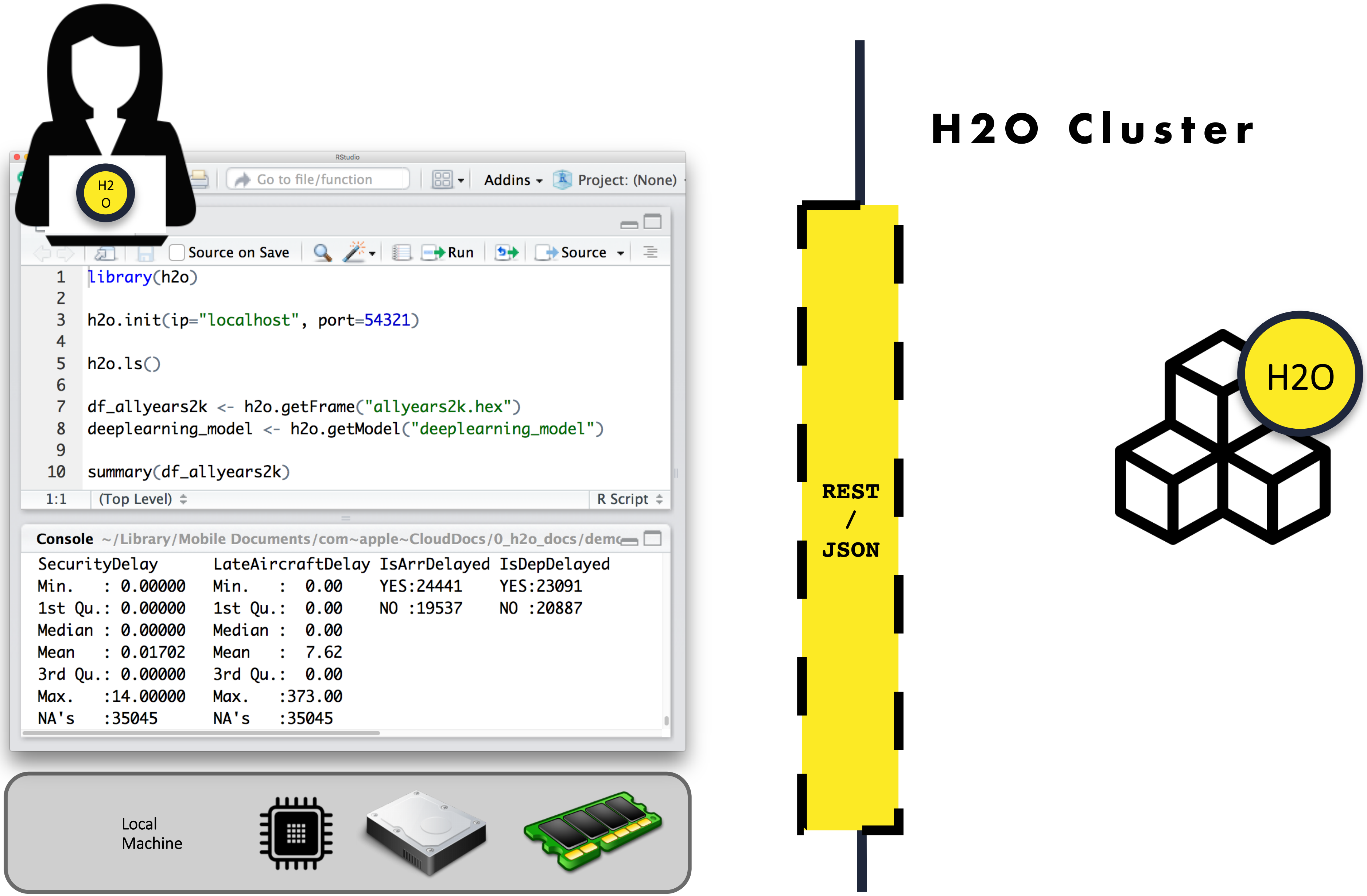
# H2O Cluster



# H2O Clients

## H2O Cluster

# H2O Cluster

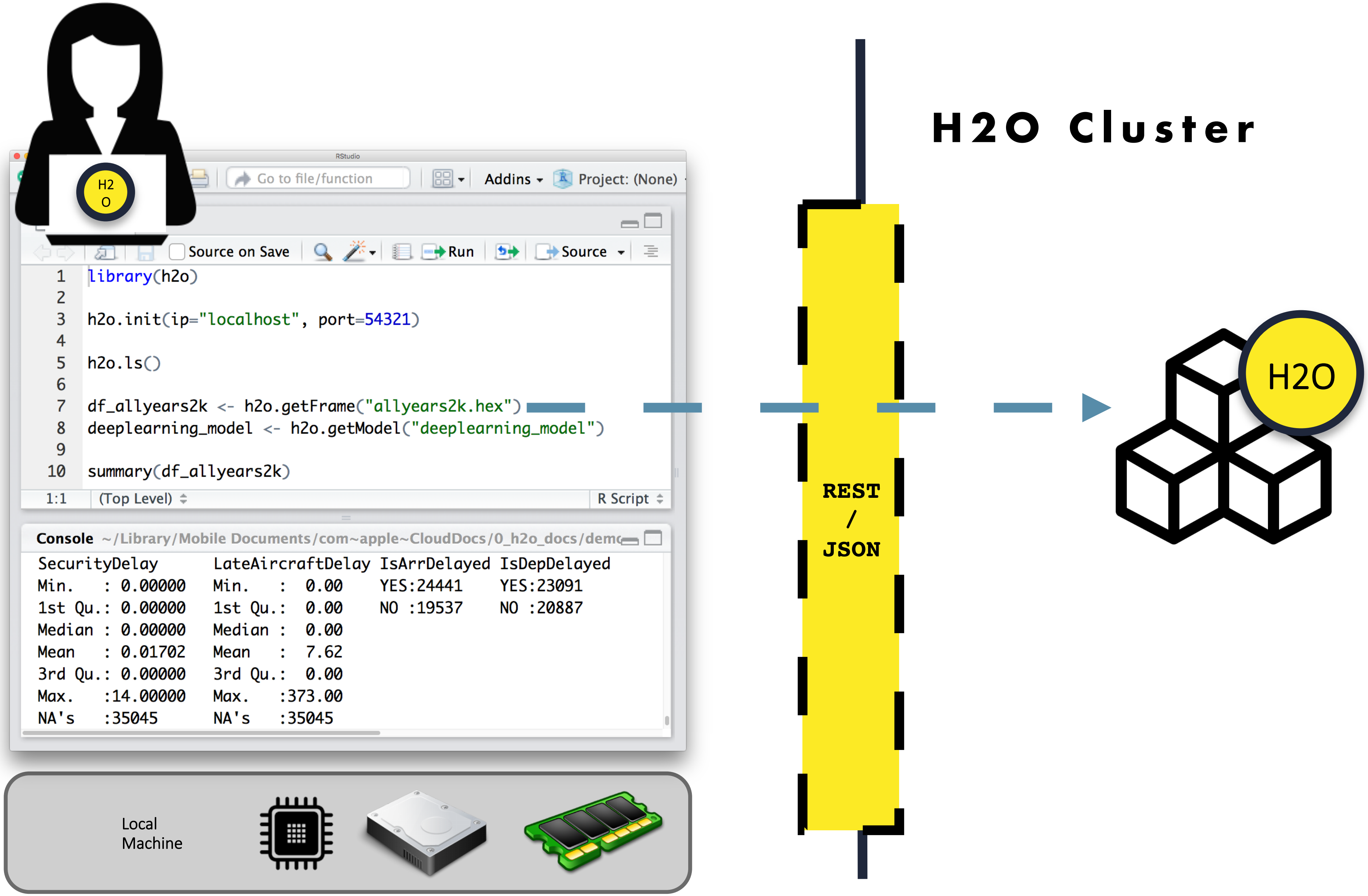


```
library(h2o)

h2o.init(ip="localhost", port=54321)

h2o.ls()

df_allyears2k <- h2o.getFrame("allyears2k.hex")
deeplearning_model <- h2o.getModel("deeplearning_model")

summary(df_allyears2k)
```

Console ~/Library/Mobile Documents/com~apple~CloudDocs/0_h2o_docs/demo

```
SecurityDelay      LateAircraftDelay IsArrDelayed IsDepDelayed
Min.   : 0.00000   Min.   :  0.00    YES:24441    YES:23091
1st Qu.: 0.00000   1st Qu.:  0.00    NO :19537    NO :20887
Median : 0.00000   Median :  0.00
Mean   : 0.01702   Mean   :  7.62
3rd Qu.: 0.00000   3rd Qu.:  0.00
Max.   :14.00000   Max.   :373.00
NA's   :35045      NA's   :35045
```

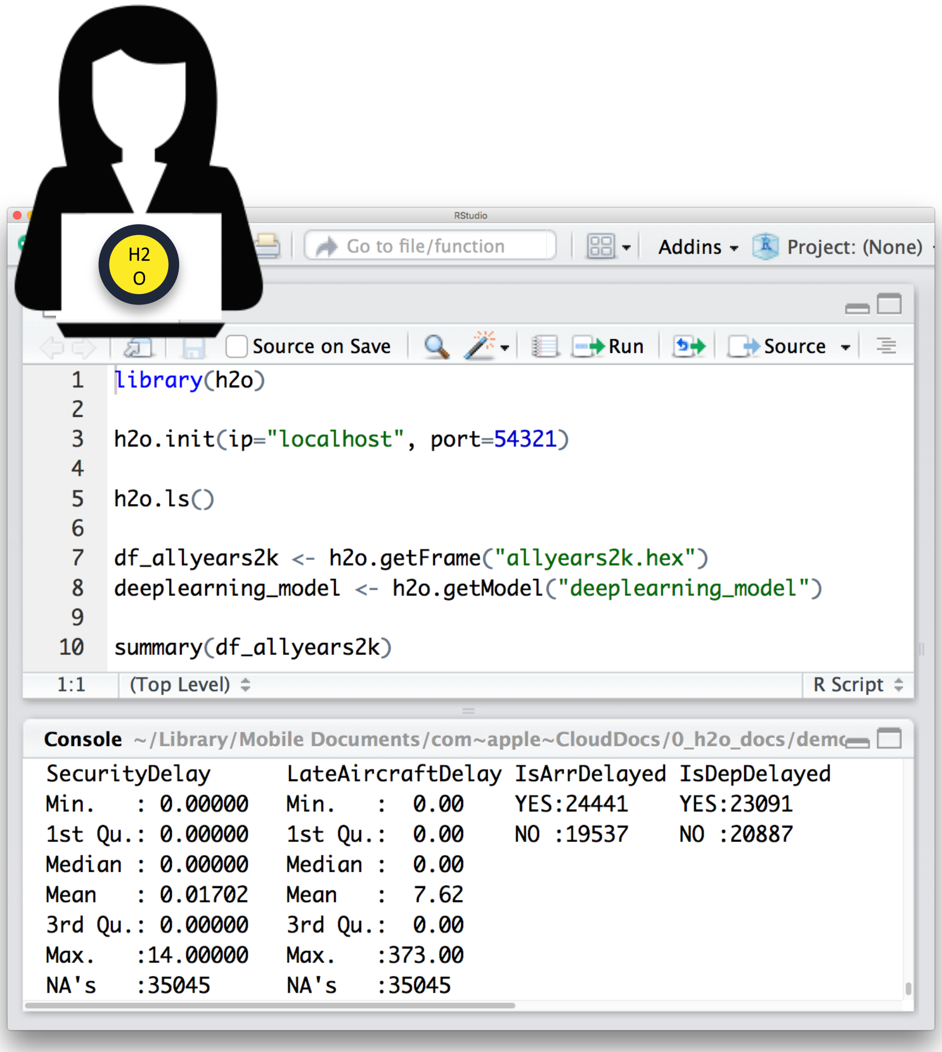
# H2O Cluster



```r
library(h2o)

h2o.init(ip="localhost", port=54321)

h2o.ls()

df_allyears2k <- h2o.getFrame("allyears2k.hex")
deeplearning_model <- h2o.getModel("deeplearning_model")

summary(df_allyears2k)
```

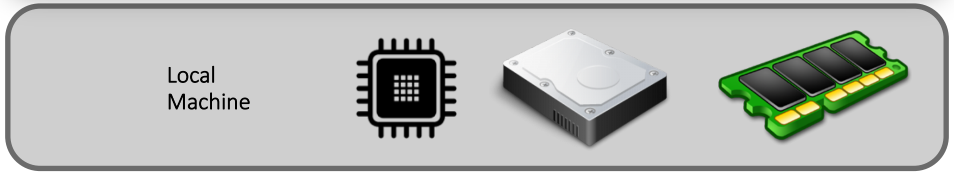
```
SecurityDelay      LateAircraftDelay IsArrDelayed IsDepDelayed
Min.   : 0.00000   Min.   :  0.00    YES:24441    YES:23091
1st Qu.: 0.00000   1st Qu.:  0.00    NO :19537    NO :20887
Median : 0.00000   Median :  0.00
Mean   : 0.01702   Mean   :  7.62
3rd Qu.: 0.00000   3rd Qu.:  0.00
Max.   :14.00000   Max.   :373.00
NA's   :35045      NA's   :35045
```

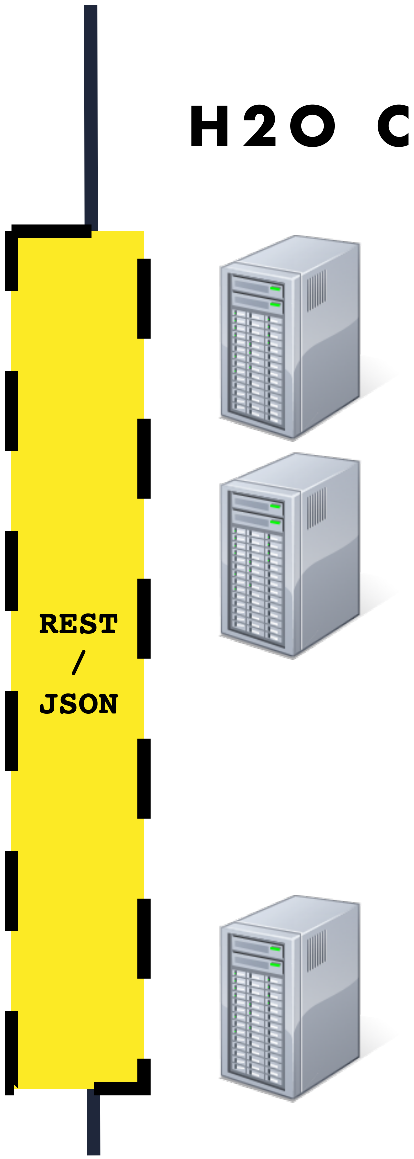
# H2O Cluster

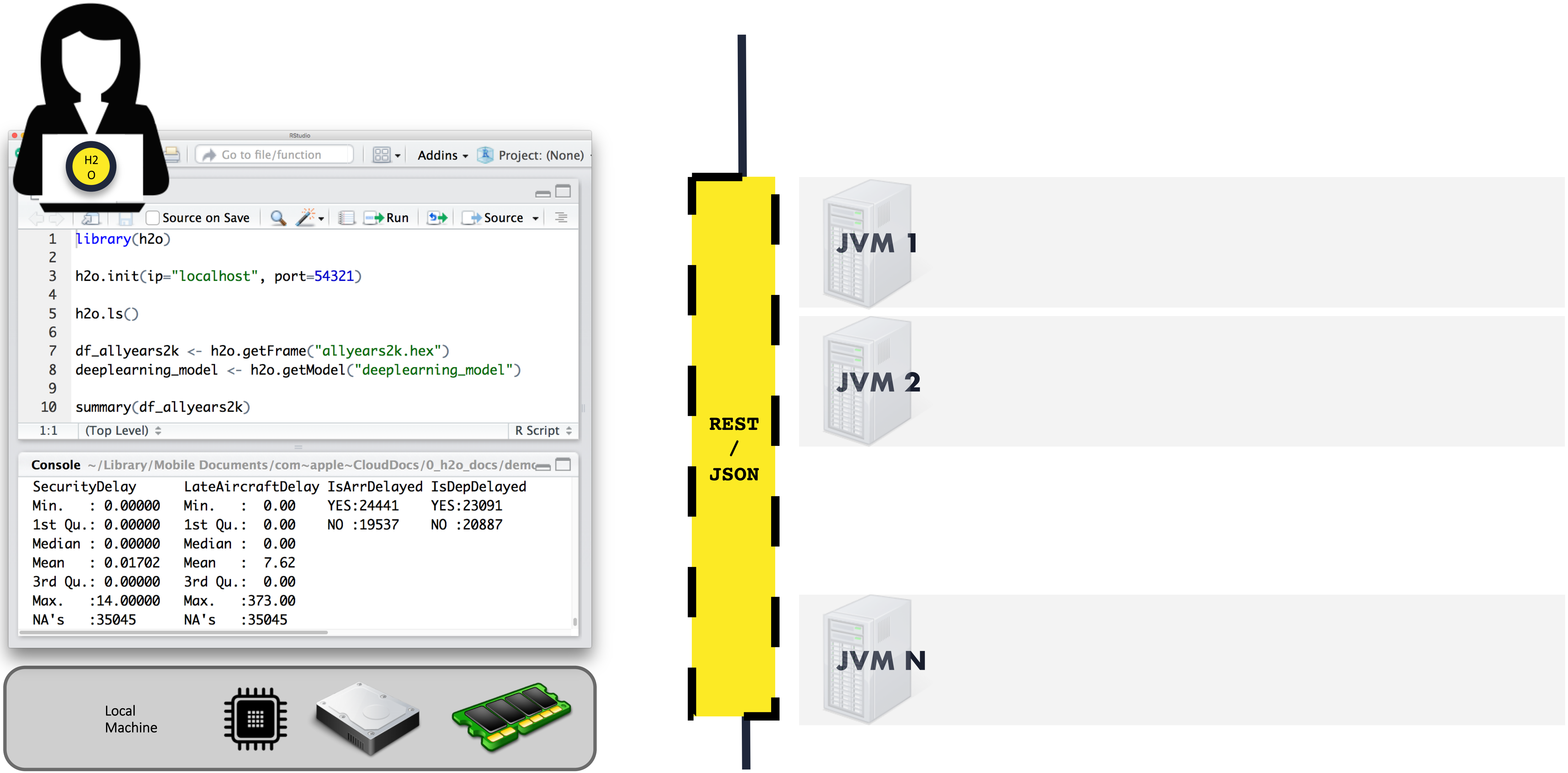


```r
library(h2o)

h2o.init(ip="localhost", port=54321)

h2o.ls()

df_allyears2k <- h2o.getFrame("allyears2k.hex")
deeplearning_model <- h2o.getModel("deeplearning_model")

summary(df_allyears2k)
```

```
SecurityDelay      LateAircraftDelay IsArrDelayed IsDepDelayed
Min.   : 0.00000   Min.   :  0.00    YES:24441    YES:23091
1st Qu.: 0.00000   1st Qu.:  0.00    NO :19537    NO :20887
Median : 0.00000   Median :  0.00
Mean   : 0.01702   Mean   :  7.62
3rd Qu.: 0.00000   3rd Qu.:  0.00
Max.   :14.00000   Max.   :373.00
NA's   :35045      NA's   :35045
```



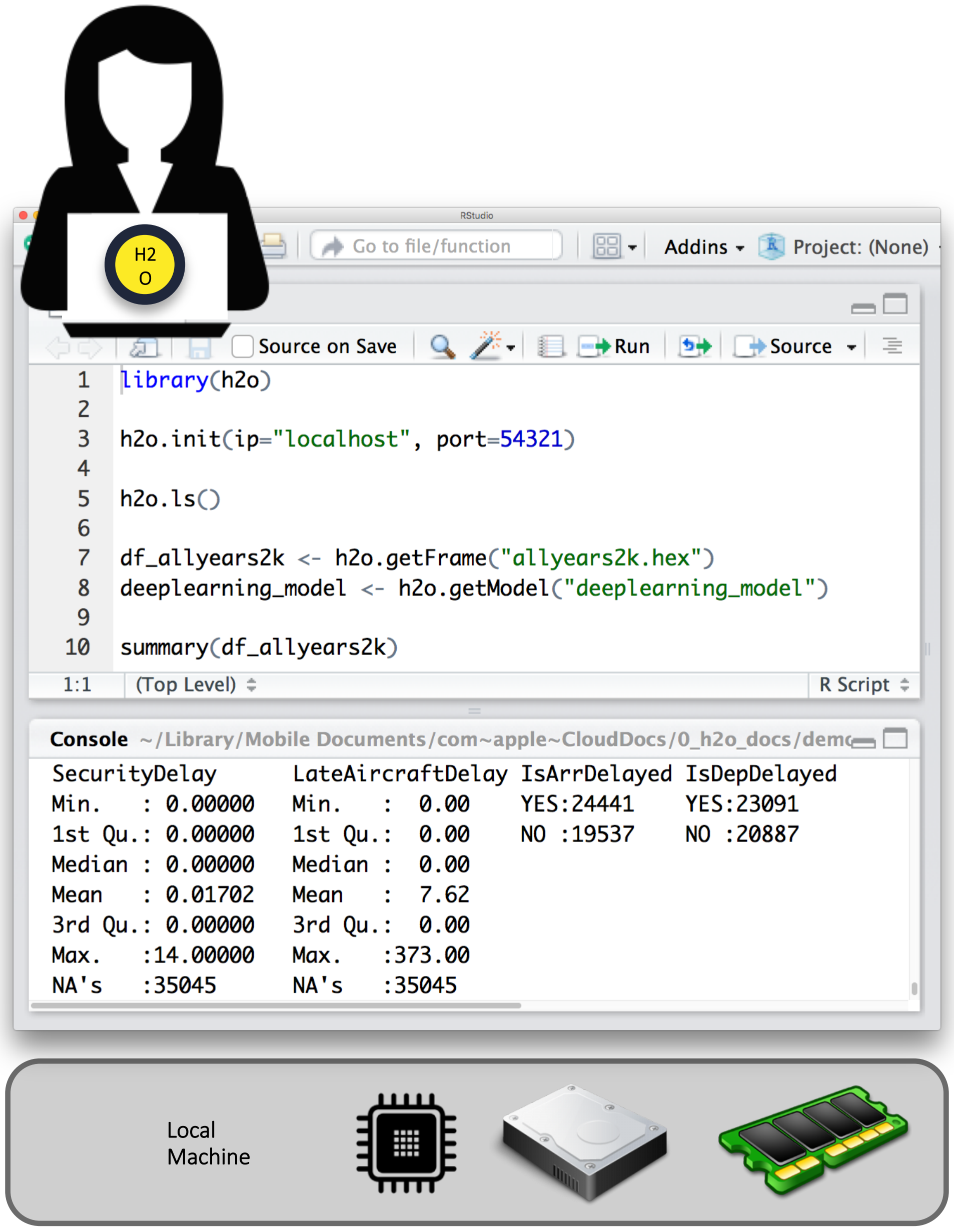




```r
library(h2o)

h2o.init(ip="localhost", port=54321)

h2o.ls()

df_allyears2k <- h2o.getFrame("allyears2k.hex")
deeplearning_model <- h2o.getModel("deeplearning_model")

summary(df_allyears2k)
```

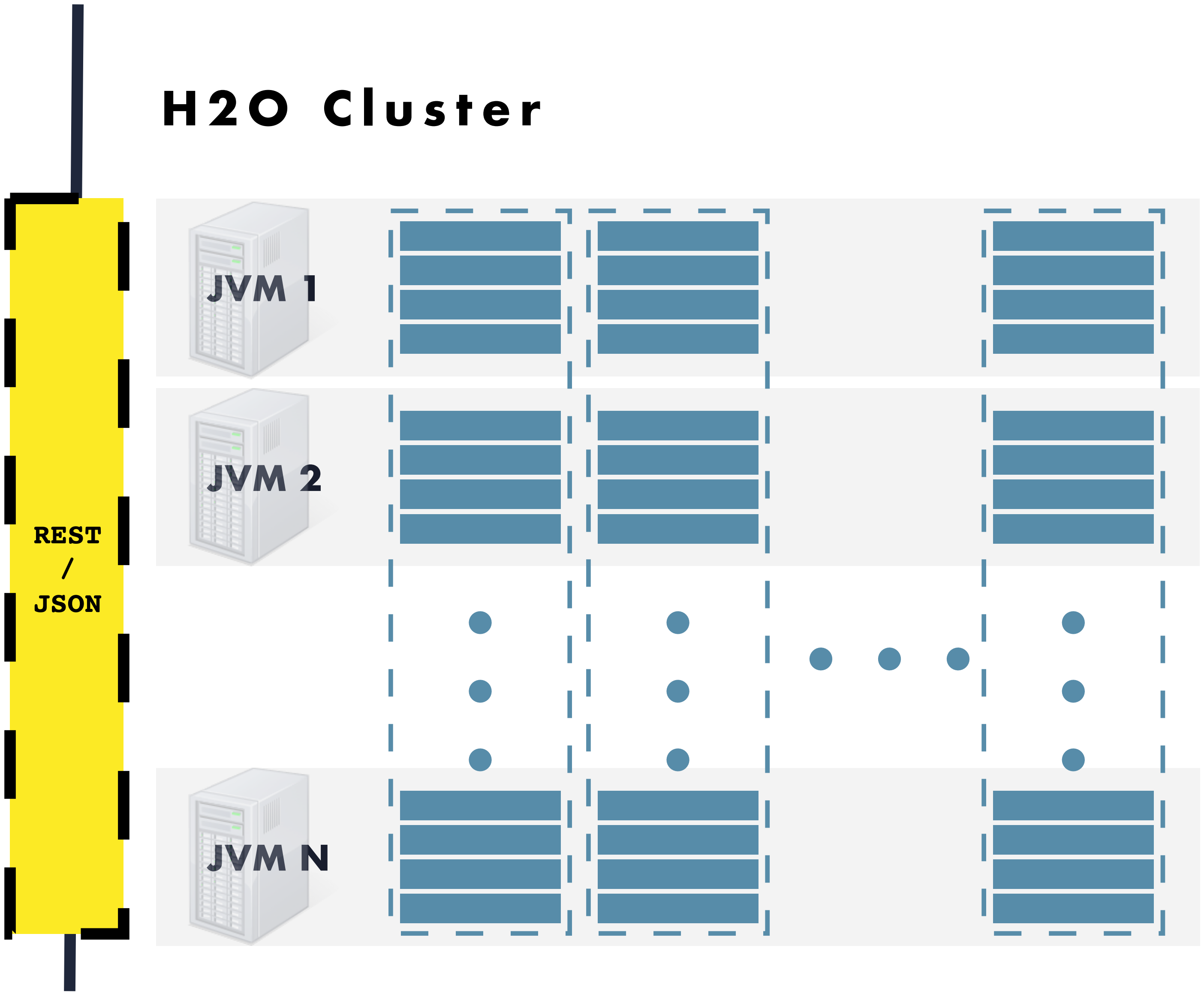
Console ~/Library/Mobile Documents/com~apple~CloudDocs/0_h2o_docs/demo

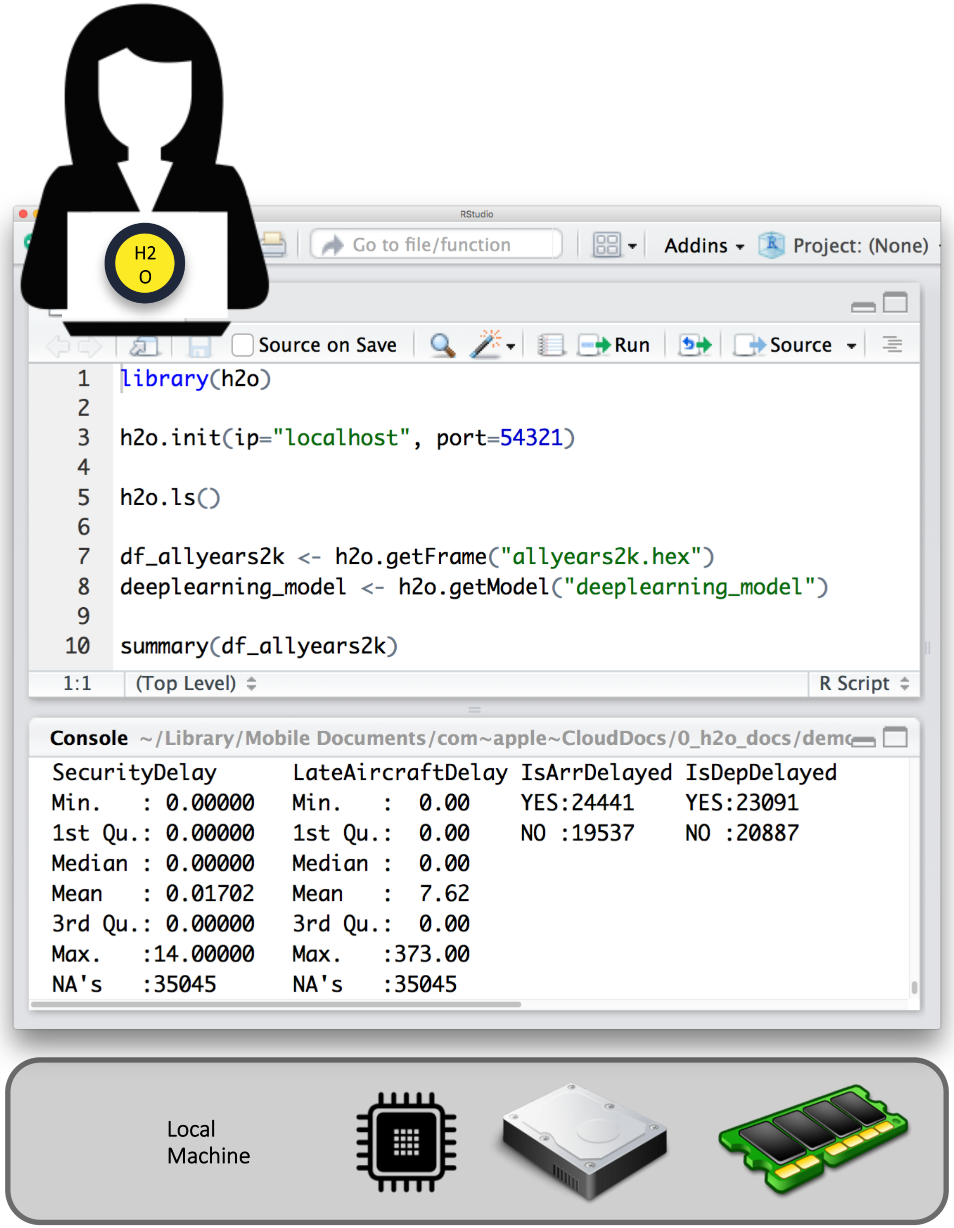
```
SecurityDelay        LateAircraftDelay IsArrDelayed IsDepDelayed
Min.   : 0.00000     Min.   :  0.00    YES:24441    YES:23091
1st Qu.: 0.00000     1st Qu.:  0.00    NO :19537    NO :20887
Median : 0.00000     Median :  0.00
Mean   : 0.01702     Mean   :  7.62
3rd Qu.: 0.00000     3rd Qu.:  0.00
Max.   :14.00000     Max.   :373.00
NA's   :35045        NA's   :35045
```



```r
library(h2o)

h2o.init(ip="localhost", port=54321)

h2o.ls()

df_allyears2k <- h2o.getFrame("allyears2k.hex")
deeplearning_model <- h2o.getModel("deeplearning_model")

summary(df_allyears2k)
```

```
 SecurityDelay     LateAircraftDelay IsArrDelayed IsDepDelayed
 Min.   : 0.00000  Min.   :  0.00    YES:24441    YES:23091
 1st Qu.: 0.00000  1st Qu.:  0.00    NO :19537    NO :20887
 Median : 0.00000  Median :  0.00
 Mean   : 0.01702  Mean   :  7.62
 3rd Qu.: 0.00000  3rd Qu.:  0.00
 Max.   :14.00000  Max.   :373.00
 NA's   :35045     NA's   :35045
```

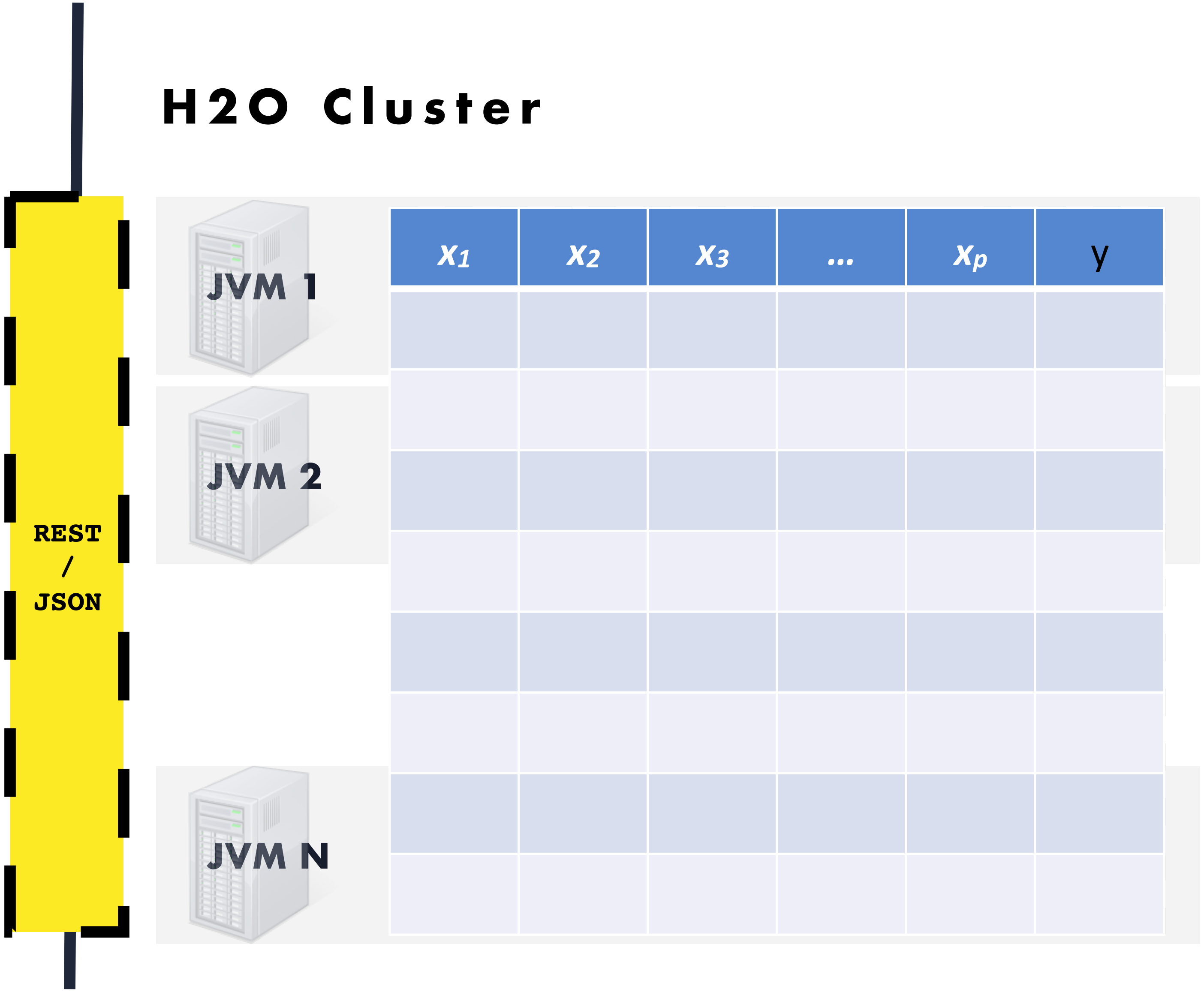




```
library(h2o)

h2o.init(ip="localhost", port=54321)

h2o.ls()

df_allyears2k <- h2o.getFrame("allyears2k.hex")
deeplearning_model <- h2o.getModel("deeplearning_model")

summary(df_allyears2k)
```

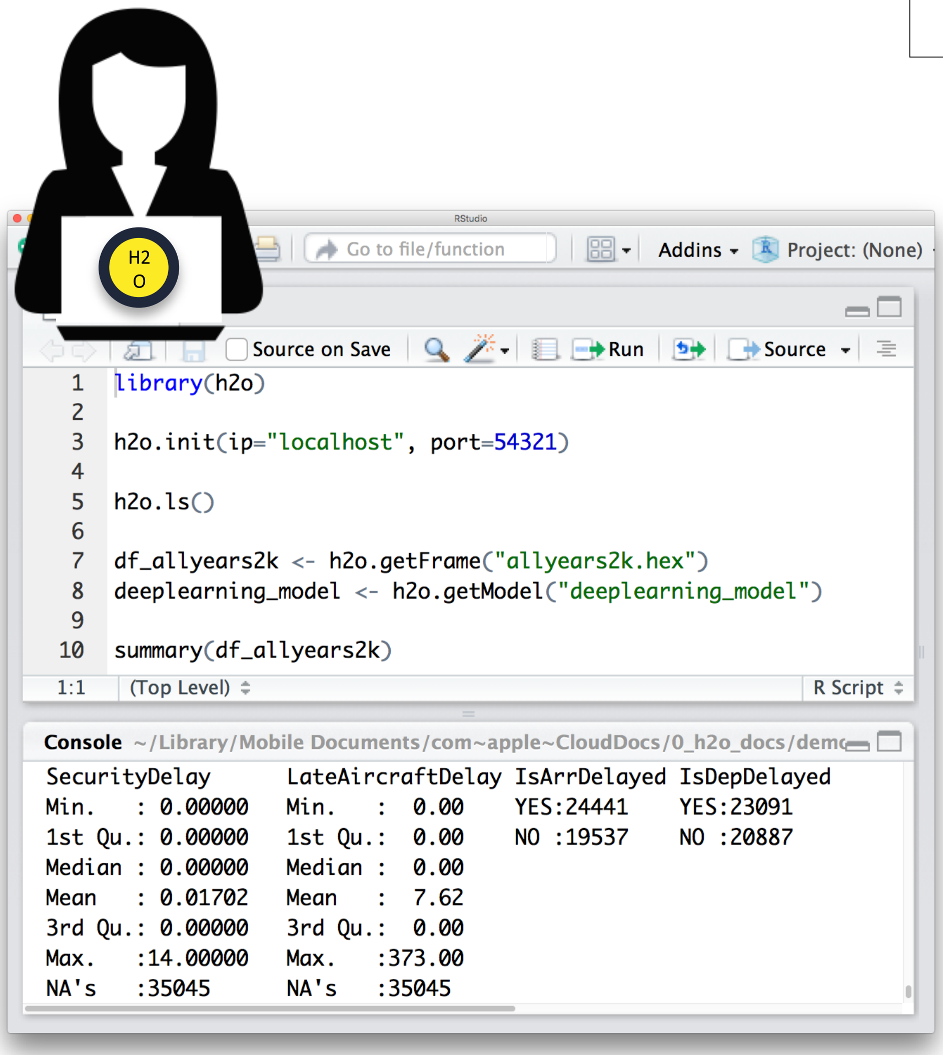
```
SecurityDelay      LateAircraftDelay IsArrDelayed IsDepDelayed
Min.   : 0.00000   Min.   :  0.00    YES:24441    YES:23091
1st Qu.: 0.00000   1st Qu.:  0.00    NO :19537    NO :20887
Median : 0.00000   Median :  0.00
Mean   : 0.01702   Mean   :  7.62
3rd Qu.: 0.00000   3rd Qu.:  0.00
Max.   :14.00000   Max.   :373.00
NA's   :35045      NA's   :35045
```

# H2O Cluster



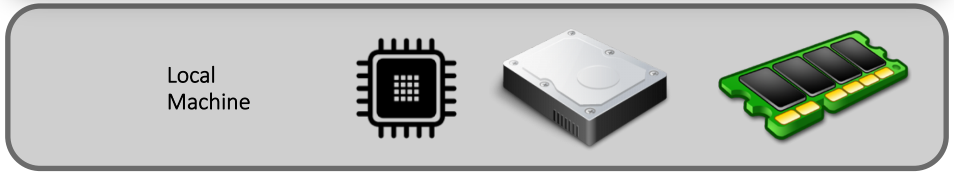
cluster returns frame key

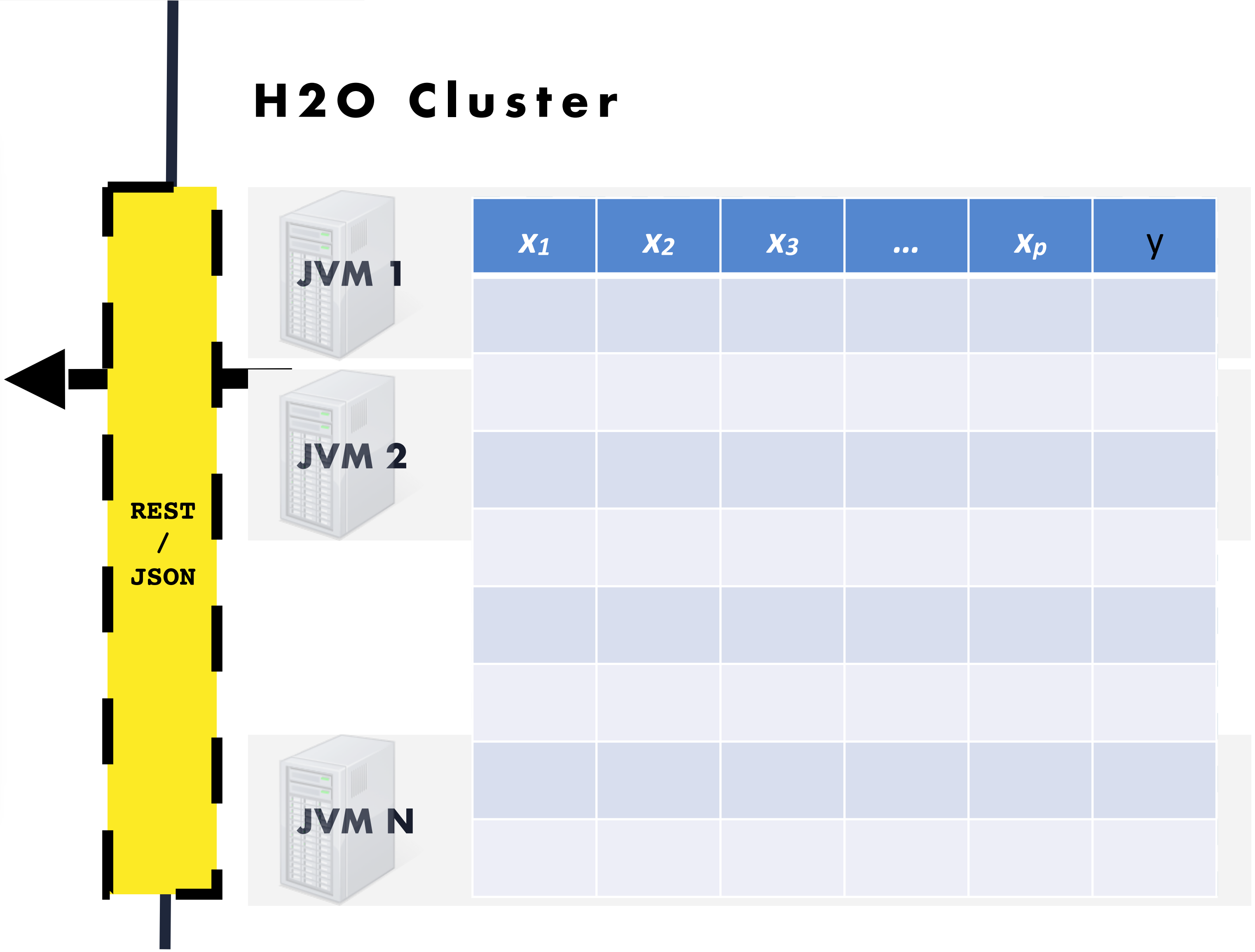
# H2O Cluster



```r
library(h2o)

h2o.init(ip="localhost", port=54321)

h2o.ls()

df_allyears2k <- h2o.getFrame("allyears2k.hex")
deeplearning_model <- h2o.getModel("deeplearning_model")

summary(df_allyears2k)
```

```
SecurityDelay      LateAircraftDelay IsArrDelayed IsDepDelayed
Min.   : 0.00000   Min.   :  0.00    YES:24441    YES:23091
1st Qu.: 0.00000   1st Qu.:  0.00    NO :19537    NO :20887
Median : 0.00000   Median :  0.00
Mean   : 0.01702   Mean   :  7.62
3rd Qu.: 0.00000   3rd Qu.:  0.00
Max.   :14.00000   Max.   :373.00
NA's   :35045      NA's   :35045
```





# H2O-3 Hands On

# The Data

- Lending Club peer-to-peer loan default dataset
- Covers loans issued in years 2007-2011
- 42,538 loans

| Type of information | Column names |
|---|---|
| Demographic Information | annual_inc, home_ownership, emp_length |
| Loan Information | purpose, term, desc, int_rate |
| **Response Column** | **bad_loan** |