

www.0xdata.com

H2O on Tableau

September 4, 2014

# H2O on Tableau

# Introduction

H2O is a powerful open source math platform developed by 0xdata. The goal of the machine learning engine is to bring distributed and parallelized algorithms into the workflow of engineers and data scientists without the growing pains. H2O is offered as an easy to integrate modular component to the infrastructure already in place. It is accessible by widely used languages like R and JSON via REST API and can ingest data from HDFS, S3, SQL, NoSQL, or from a local disk. The connection to R shared by H2O and Tableau made it possible to integrate the two.

This whitepaper is made available for H2O, R, and Tableau users or prospective users of any of three. The documentation of H2O + R is already available on 0xdata, henceforth the paper will assume some basic knowledge and previous use or R on H2O.

# Perquisites

* 64-bit Java 1.6 +

# Overview

* Get and install R
* Download [H2O](http://0xdata.com/downloadtable/)
* Have Tableau 8.1 running

Tableau is the frontend visual organizer that utilizes all the available statistic tools from open source R and H2O. Tableau will connect to R via a socket server using a library package already built for R. The H2O client package available for installation allows R to connect and communicate to H2O via a REST API. So by connecting Tableau to R, Tableau essentially can launch or initiate H2O and run any of the features already available for R which is heavily supported.

Note: In H2O’s github repository there is two Tableau demo workbook. One for Tableau 8.1 and one for the newer version of Tableau 8.2, both of which currently only works on a Windows version of Tableau. There is currently an issue running calculating fields on a Mac that has been submitted as a [question](http://community.tableausoftware.com/thread/148067) on Tableau’s forum. Generally it seems like the issue is how states are saved or not saved between a Windows and Mac running on Rserve.

# Metadata

Tableau requires a data input of some sort in order to run but when dealing with immensely big data as a H2O user would typically do, it is impractical and unnecessarily time consuming to feed all the data into Tableau. So the solution is to create a small metadata file that can be easily loaded from disk.

Note: in particular, geo attributes like cities and latitude/longitude coordinates need to be imported into Tableau. Aggregates and calculated fields cannot be converted so even if you have a vector of cities (New York City, San Francisco, Houston, Austin, Seattle) as a output from R into Tableau as a new measure the user cannot convert the measure into geographical points.

The metadata is just a file with factor levels of the data frame that’ll give a quick overview of the unique input values and cardinality of each column. This format will also allow the user to calculate across the table making it easier actually visualize some of the data coming from H2O. Example: if a column has a 352 categorical values and a binomial GLM is ran with 352 coefficients in the output, Tableau will allow the plotting of the coefficients against the column.

The following data is a truncated metadata file for 120million rows airline data set. This file in particular is only about 352 rows big. A extra column was prepend to the data file to grab the column names of the actual data set, this column will help populate certain parameters in the workbook.

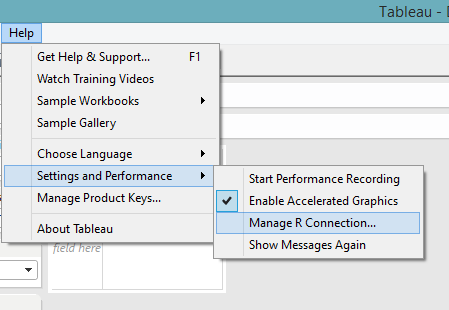
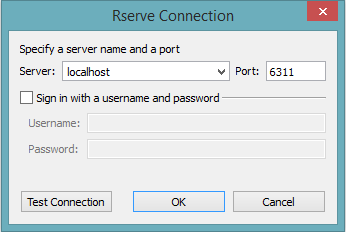


# Start R + Tableau Connection

1. Install Rserve which is the package built to create R servers
2. Once installed load up the Rserve library and then run run.Rserve or Rserve (run.Rserve is recommended) with your desired port:

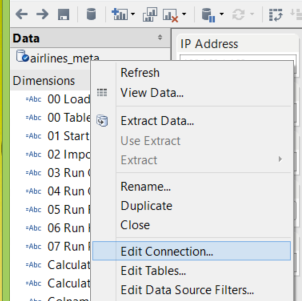
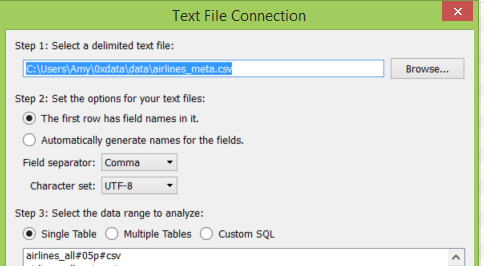


1. Open Tableau workbook template with most of the code already encoded
2. Navigate to “Help>Settings and Performance>Manage R Connection” to establish connection to the R server:

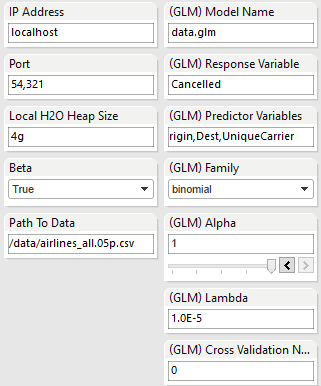
 

# Tableau Workbook Set Up

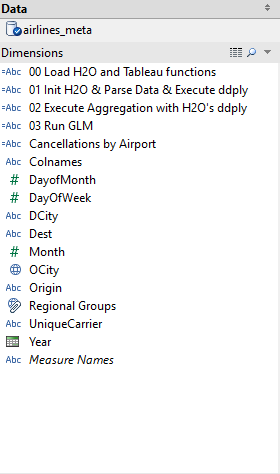
1. Edit the data source to point to your meta data:

1. Configure the IP Address and Port then start H2O, if an H2O instance has not been launched configure the heap size before using calculated field “01 Start H2O in R”. Do the same for importing a file into H2O by dictating the path to the data as well as the import type (i.e. from disk or from HDFS) :



# Execute Script in Tableau 8.1



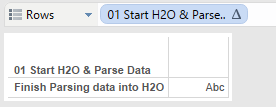
Double click on the calculated fields under Dimensions in the order they are listed:

1. Execute “00 Load H2O and Tableau functions” to run library(h2o) and define other R functions to be used in Tableau.
2. Execute “01 Init H2O & Parse Data” will run h2o.init() and h2o.importFile().
3. Execute “02 Compute Aggregation with H2O’s ddply” will run h2o.ddply() and create measures for use in “Flights By Month” and “Flights By Airport” worksheets.
4. Execute “03 Run GLM” to build a GLM model in H2O and grab back coefficient values that will be plotted in multiple worksheets.

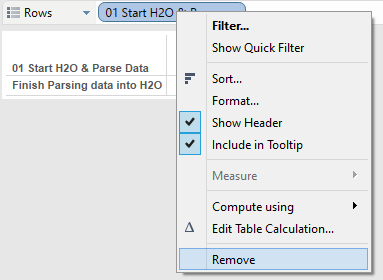
# Execute Script in Tableau 8.2

Move a calculated field under Measures to the worksheet in order to execute the R scripts:

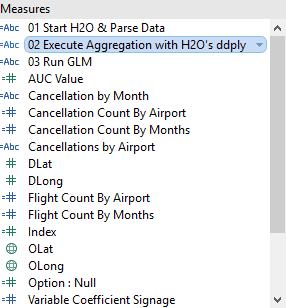
1. Move “01 Start H2O & Parse Data” to the Row shelf which will trigger an execution.



1. Right click the field sitting on the Row shelf and select Remove to remove it from the worksheet.

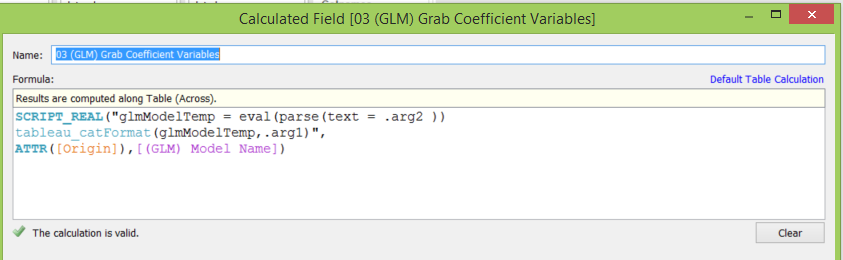


1. Repeat this calculated field “02 Execute Aggregation with H2O’s ddply” and “03 Run GLM” to finish all the calculations necessary for the dashboard.

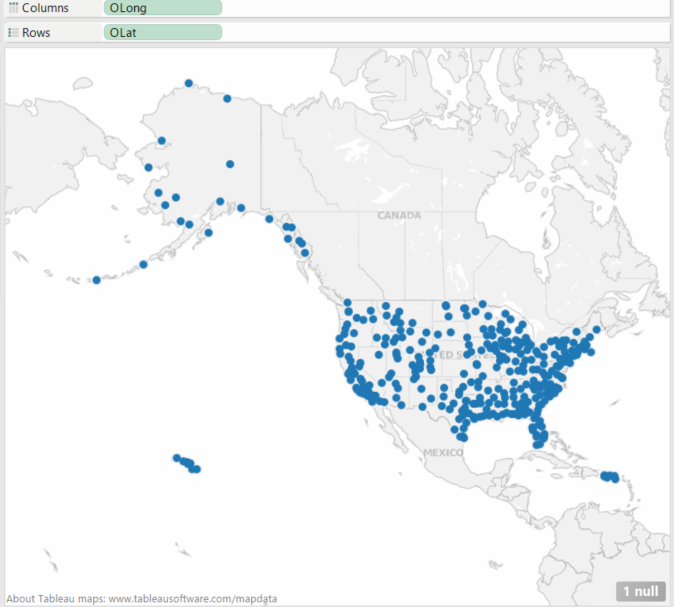


# Writing R Script in Calculated Fields

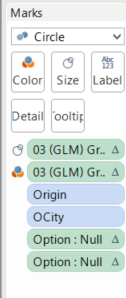
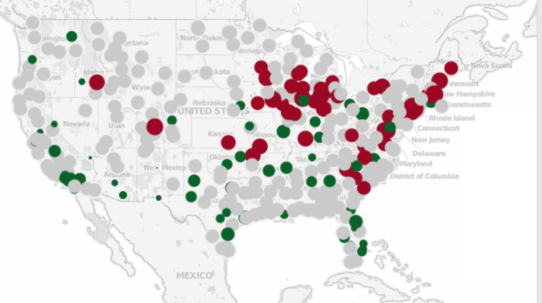
1. Create a calculated field that is an aggregate vector in Tableau and run it. Keep in mind that there needs to be input in the form of Dimensions or Measures from Tableau to R in order for the users to plot the calculated field. Tableau is doing a calculation across the table for all the unique values of that Measure, so the output will be of the same length.



1. For a map of the log odds coefficients simply create a new sheet, Longitude measure goes in the columns and Latitude in the rows. When plotting points remember to treat values as discrete so if Tableau tries to automatically change the measure into an aggregate, right click and select dimension.



1. Move coefficient measures (written as a calculated field) into Tableau. The plot below is a plot of log odds coefficients for each airport so the grey values have coefficients of 0 so it’s still relevant to include if you want to see the full spectrum of cancellation odds.

1. However if the goal of the map was to showcase the airports least likely and most likely to get cancellation, you can remove the null or 0 coefficients by utilizing the Tableau’s filter:

