Uso de R en Ambientes Productivos

Tomás León – Head of Analytics Consulting & Factory
@tomasleon2

Motivación

Motivación

R como versión Libre de S, nace en la academia para la **investigación**

La generación R llega a la industria y comienza a **reemplazar algunos softwares** privativos por R

Motivación

Se hacen populares áreas como analytics/ BI/Data Science. El mundo entiende que necesita **procesar datos para obtener información**. Los modelos Estadísticos/ Matemáticos son cada día más confiables.

Las empresas grandes comienzan a usar software libre para sus análisis.

Propuestas para R en producción

H2O + R

H2O es una plataforma de aprendizaje automático distribuida en memoria y de código abierto con escalabilidad lineal. H2O es compatible con los algoritmos estadísticos y de aprendizaje automático más utilizados, incluidos GBM, deep learning entre otros.

Se puede integrar con Spark, Python, ecosistemas de big data, etc.

H2O + R


Installation
• Java 7 or later; R3.1 and above; Linux, Mac, Windows
• The easiest way to install the h2o R package is CRAN
• Latests version: http://www.h2o.ai/download/h2o/r


Design
All computations are performed in highly optimized Java code in the H2O cluster, initiated by REST calls from R.

H2O + R

```
tomas@tomas-Lenovo-B590: ~/h2o_Prueba/Implementacion
File  Edit  View  Search  Terminal  Help
(generico) tomas@tomas-Lenovo-B590:~/h2o_Prueba/Implementacion$ time java -cp .:h2o-genmodel.jar main
Entra en default (1=yes; 0=no): 0
Class probabilities: 0.9816356290027105,0.018364370997289523

real    0m0,149s
user    0m0,171s
sys     0m0,032s
(generico) tomas@tomas-Lenovo-B590:~/h2o_Prueba/Implementacion$
```

Usando un MOJO como objeto

(donde se guarda un modelo Stacked Ensemble de alta complejidad)

se logra calificar un individuo nuevo en 0.149s

Docker + H2O + R

Docker es un **proyecto de código abierto** que automatiza el despliegue de **aplicaciones** dentro de **contenedores de software**, proporcionando una capa adicional de abstracción y automatización de virtualización de aplicaciones en múltiples sistemas operativos.

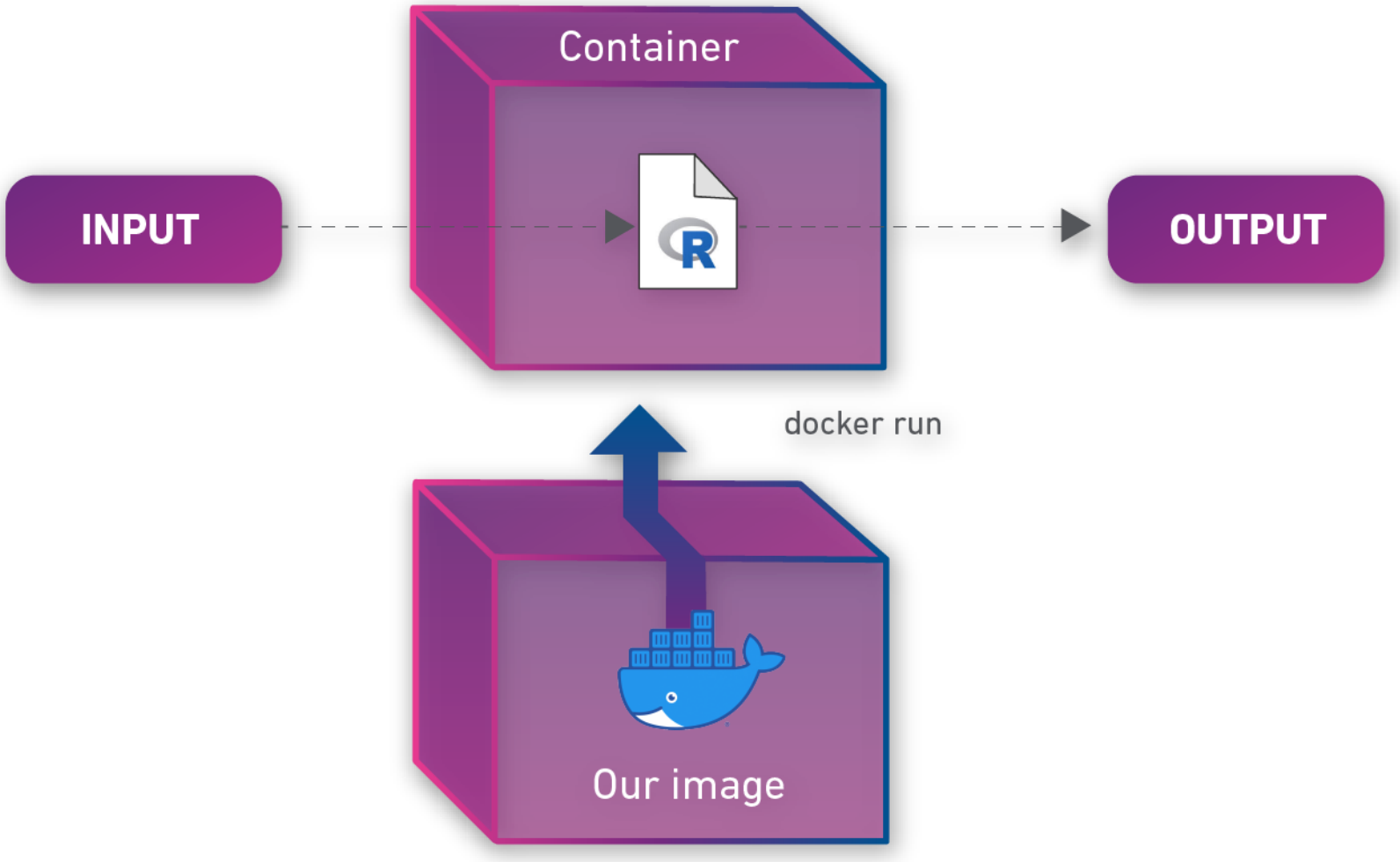
Docker utiliza características de aislamiento de recursos del kernel Linux.

Dentro de docker, podemos aislar completamente nuestro ambiente desarrollo

(versión de h2o, spark, R, librerías R, scripts, etc, etc ,etc)

Docker + H2O + R

Running a Container Based on our Image



Docker + H2O + R

```
FROM r-base:3.4.4

RUN apt-get update && apt-get install -y \
        libpq-dev \
        build-essential \
        libcurl4-gnutls-dev \
        libxml2-dev \
        libssl-dev \
        libc6-dev \
        default-jdk \
        default-jre

RUN R CMD javareconf
RUN mkdir -p /opt/software/setup/R
ADD install_packages.R /opt/software/setup/R/
RUN Rscript /opt/software/setup/R/install_packages.R

RUN mkdir /home/produccion
RUN mkdir /home/produccion/data
ADD Binning.rds /home/produccion/
ADD StackedEnsemble_BestOfFamily_AutoML_20181227_141429 /home/produccion/
ADD proceso_bach.R /home/produccion/

CMD ["Rscript","/home/produccion/proceso_bach.R"]
```

Docker con:

1. R 3.4.4
2. Java
3. H2O
4. Modelo SE
5. Script para variables
6. Vínculo con carpeta de entrada
7. Vínculo con carpeta de salida

Docker + H2O + R

Para calificar una base batch de:

- 3.356.708 registros.
- Donde es necesario mantener:
  - Un script para cálculo de 19 variables.
  - Una versión inmutable de H2O - R por modelo SE
  - Una versión “inmutable” de Java por dependencia de H2O

Conclusiones

Si bien existen muchas formas de “implementar” modelos R en producción.

Pocos logran cumplir los tiempos establecidos para un procesamiento en tiempo real,

o incluso en procesamientos por lotes.

Hoy día en Experian ya contamos con plataformas que se integran con R, de la misma manera, ya algunos de los cores a nivel mundial usan R para procesar data bajo estos esquemas

H2O

Docker

Por eso y por ahora una buena aproximación es usar R en conjunto con:

Kubernets

Plumber*

*No hemos realizado pruebas en producción

¿Preguntas?

@tomasleon2