Learn to use RStudio in the Cloud

May 28, 2020

Your hosts

Ross Cruickshank

Developer Advocate

driving the screen

https://developer.ibm.com/profiles/ross.cruickshank/

Yamini Rao

Developer Advocate

driving the Q&A

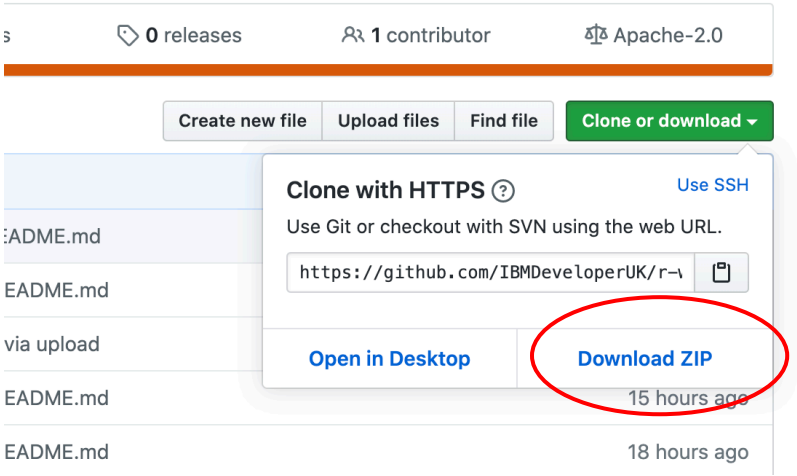
https://developer.ibm.com/profiles/yrao

https://ibm.biz/rstudio @CodeMeetup https://github.com/IBMdeveloperUK/r-workshops

Preparation
https://github.com/IBMdeveloperUK/r-workshops

For ease of use, and to avoid issues with file types/extensions, download the Git repository as a ZIP file and expand locally

When we get to the Rstudio stage, you can upload whole repo into a project folder, and access all the R notebooks directly.



An introduction to R

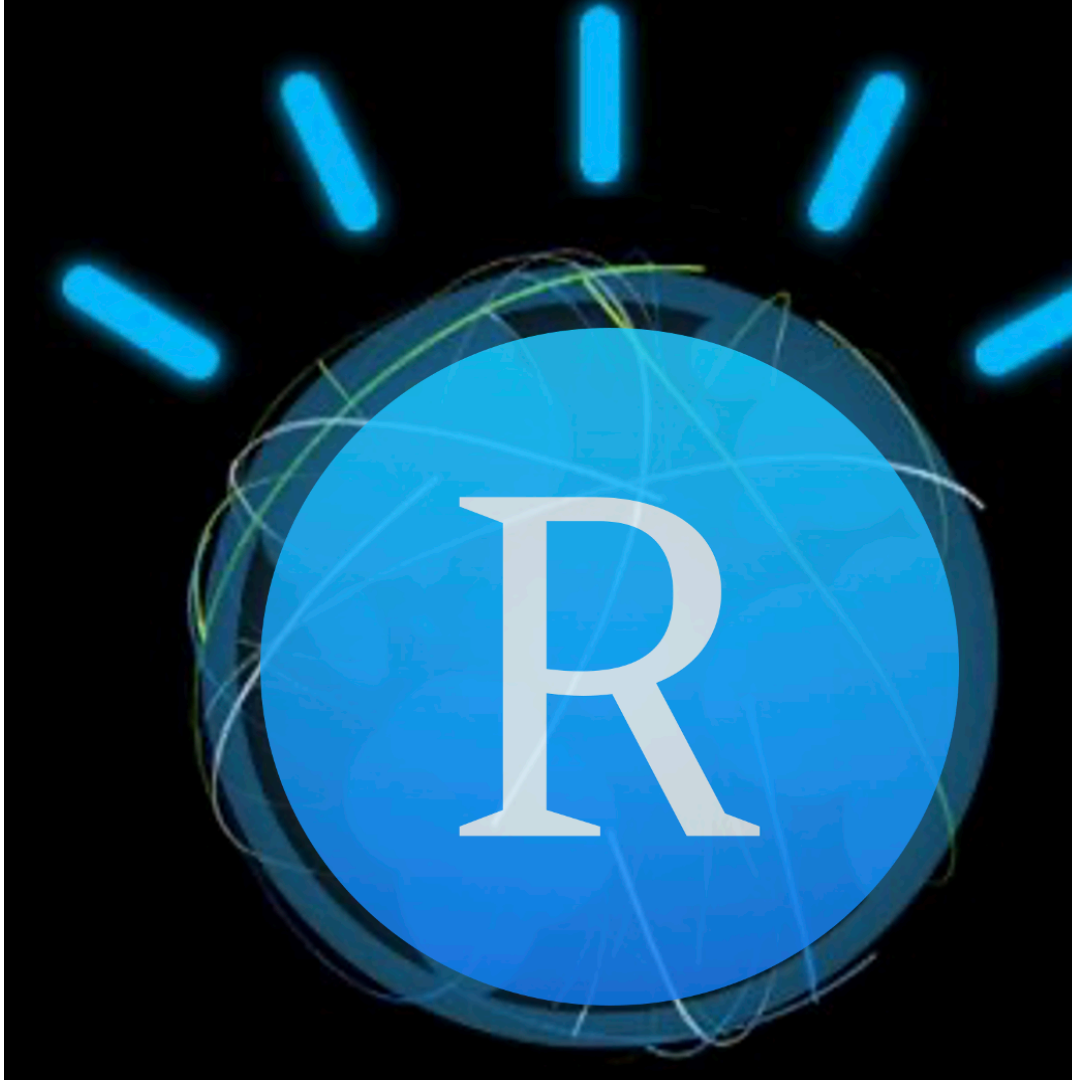
Basics	Charts	Mapping	Applications
Language components	Visualising data	Leaflet for R	Shiny framework
Data structures	Plots	Overlaying data on maps	Geo-data for water
Functions	Richer plots with ggplot	Linking geo-data to external information	Coronavirus data explorer

What's it for?

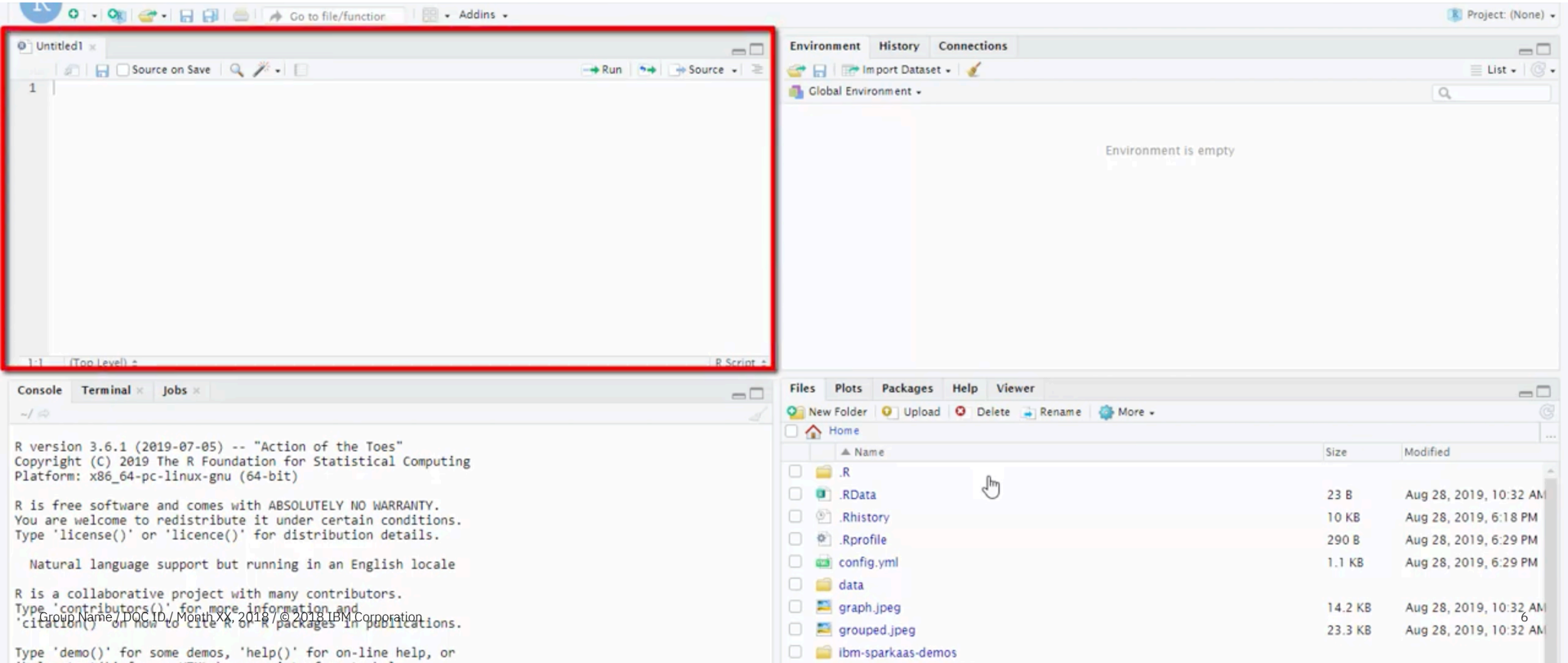
Data analysis, data science, statistical modelling, machine learning, data visualization, interactive data exploration

https://ibm.biz/rstudio @CodeMeetup https://github.com/IBMdeveloperUK/r-workshops

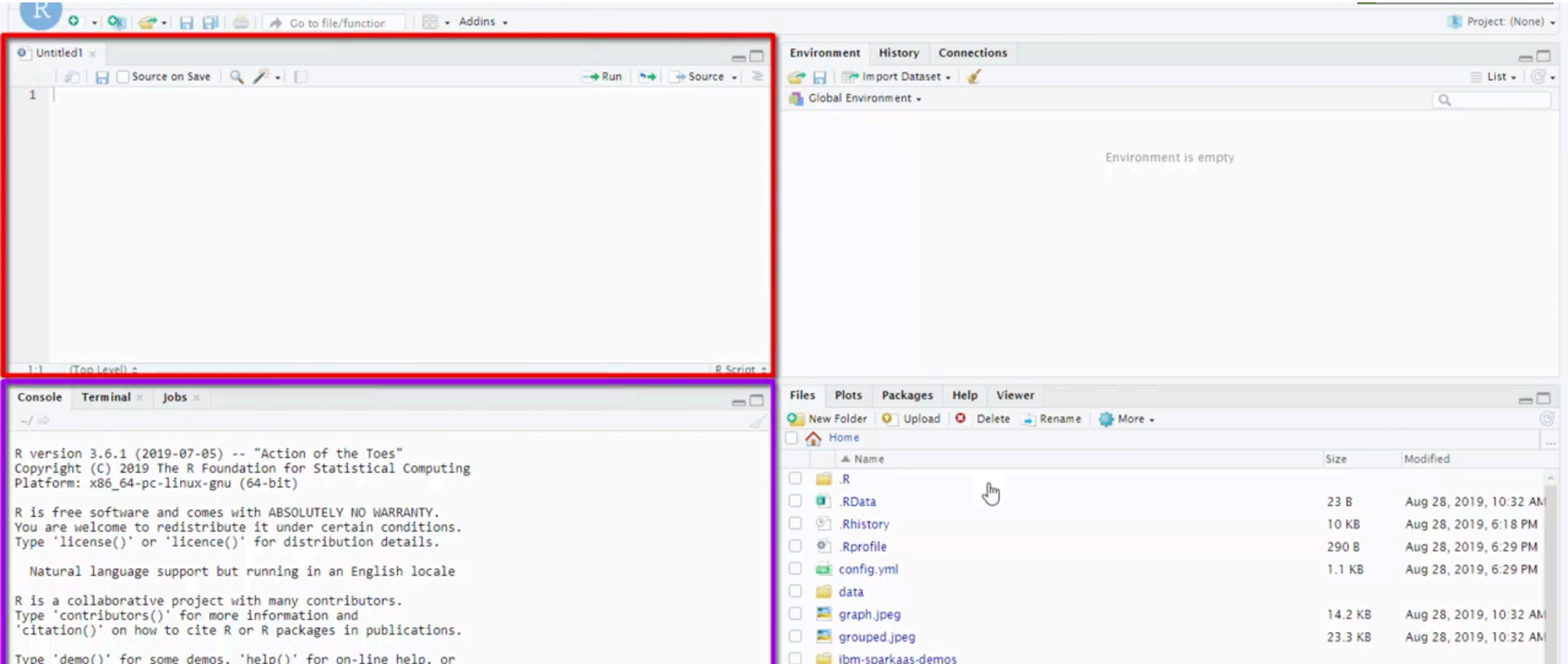
Introducing RStudio



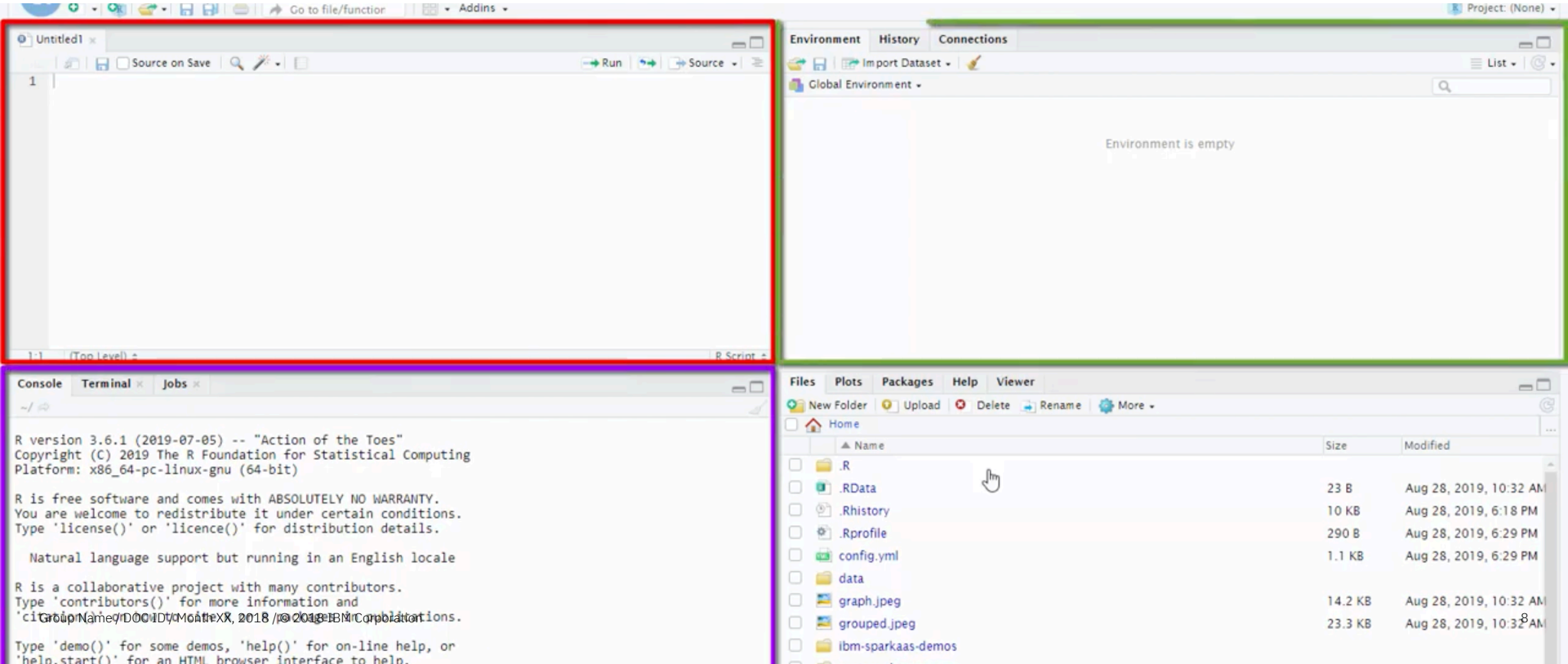
Coding pane
Source editing, running, debugging



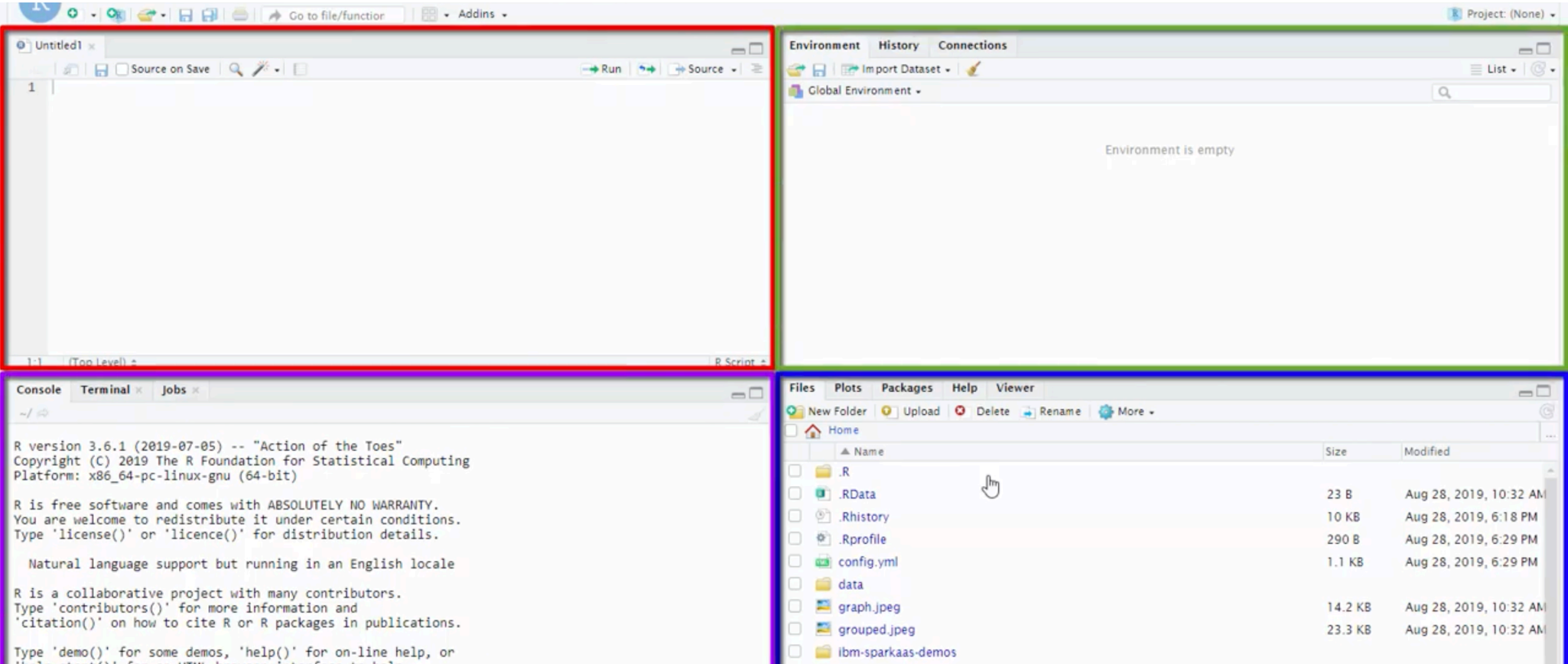
Console pane
Code I/O console, interpreter, Jobs



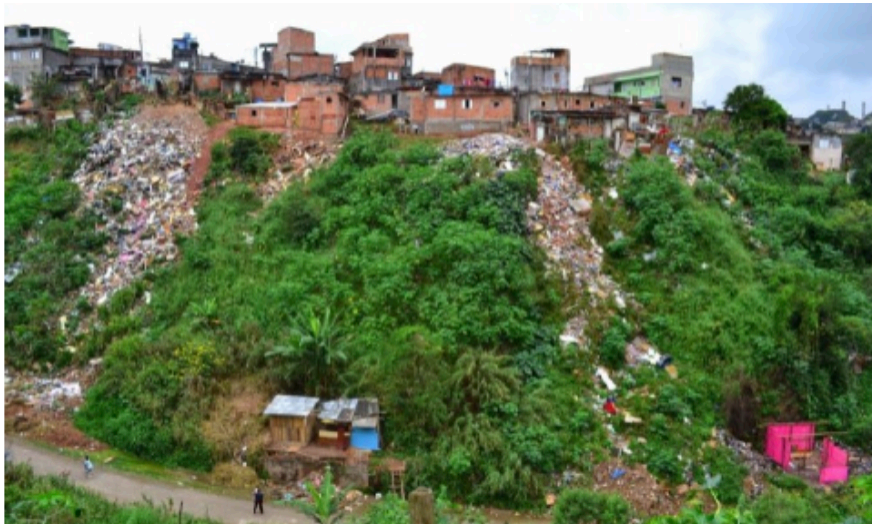
Environment pane
Variables, History, Connections



I/O pane
Files, Plots, Packages, Help



What's it for?



Data analysis, data science, statistical modelling, machine learning, data visualization, interactive data exploration

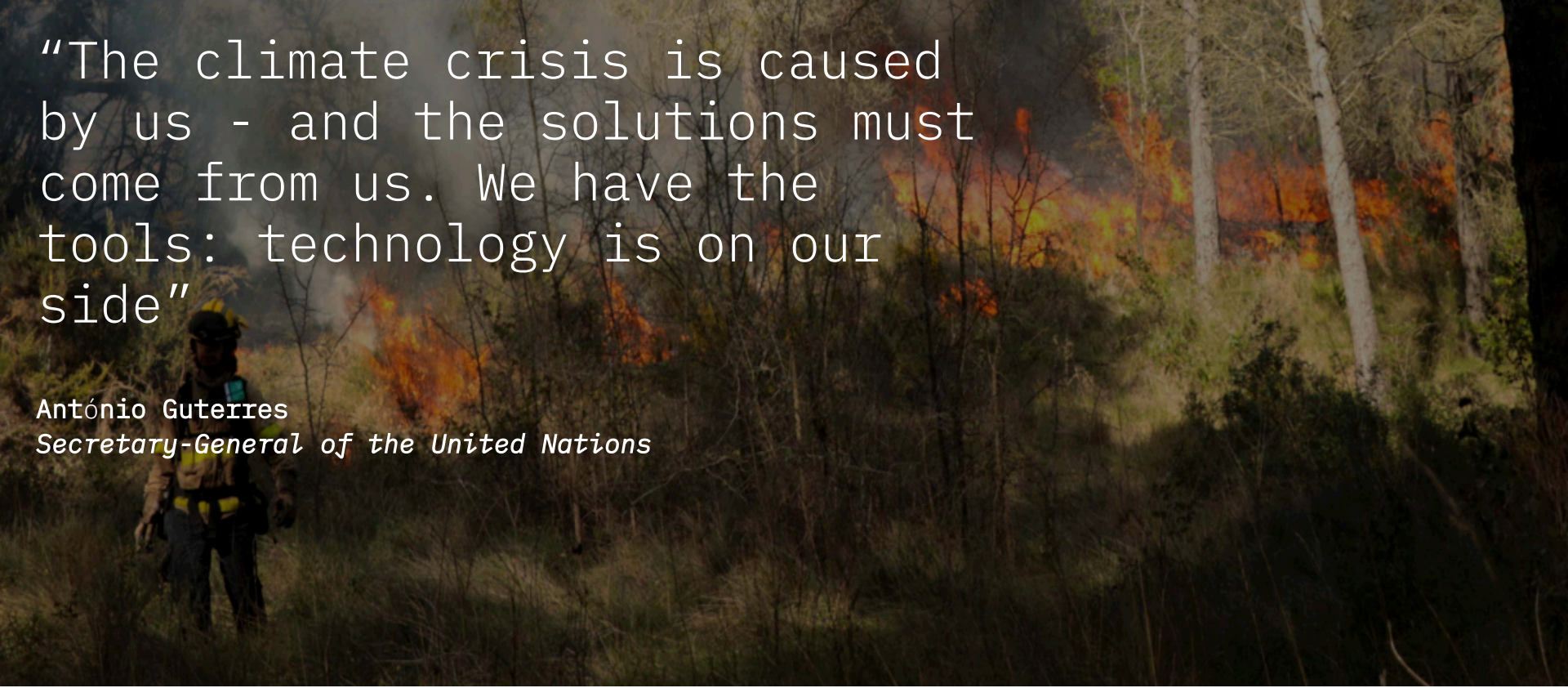
In short, an ideal tool for helpful to analyze data that could help with world societal and environmental challenges.

Example: Modelling and visualizing risk of landslide

https://www.researchgate.net/publication/328974779_A_tool_to_compute_the_landslide_degree_of_risk_using_R-Studio_and_R-Shiny

Build & deploy solutions to help halt & reverse the impact of climate change

Our reality



75%

of earth’s carbon emissions are caused by our cities .

1/4

of people are likely to live in chronic water shortage areas by 2050.

37m

people are severely effected by natural disasters every year.

What our reality needs

Energy sustainability

60%

of total greenhouse gas emissions are due to current energy methodologies

Water sustainability

2B+

people are living with risk of reduced access to freshwater resources

Disaster resiliency

1M

people have died because of natural disasters since the turn of the century

Action you can take now

Build & deploy solutions to help halt & reverse the impact of climate change

In 2020, Call for Code is aligned to the UN 75th anniversary global conversation theme of climate change, with a focus on **energy sustainability**, **water sustainability**, & **disaster resiliency**

- Join a **movement** of:
  - **210,000+** developers, data scientists & problem solvers
  - **165+** nations
  - **8,000+** applications built
- Have the chance to win:
  - **$200,000** USD
  - Open Source support from **The Linux Foundation**
  - Meetings with **mentors** & potential **investors**
  - **Solution implementation** support through **Code and Response™**

Call for Code
Founding Partner

Call for Code
Creator

Call for Code
Charitable Partner

Call for Code
Charitable Partner

20 CALL FOR 20 CODE®

Global Challenge

Energy sustainability

Technologies such as:

- AI
- IoT

can leverage **data** to pinpoint & **reduce** energy consumption

Water sustainability

Technology can address:

- Water quantity
- Water quality

through **data science** and **opensource platforms** to help at personal and global scale

Disaster Resiliency

Technology can help us:

- Reduce toxic exposure
- Improve preparedness

by developing hardware and software solutions with **machine learning** capabilities

Solving societal issues

Previous Call for Code Winners

Call for Code 2019 Winner, Prometeo

A first of its kind, hardware-software solution equipped with sensors that detect toxicity levels for firefighters battling wildfires in real time, offering a transformative solution to first-responders globally.

Technology: IBM Cloud IoT Platform, IBM Cloud services, Cloudant database

Call for Code 2018 Winner, Project Owl

A groundbreaking hardware-software solution that automatically constructs a mesh network via small Wi-Fi units allowing for sustained communications in peak times of need.

Technology: IBM Watson Studio, IBM Cloud IoT Platform

Other projects now in development

Call for Code runner-up, P3DR

The first of its kind software solution that provides displaced families with immediate access to engineering advice following natural disaster repercussions

Technology: IBM Watson Studio, IBM Watson Visual Recognition
Receiving Linux Foundation Support

Call for Code runner-up, DroneAID

An intuitive hardware-software solution that gathers aid needs from stranded communities with drones powered by visual recognition allowing for clear-cut communication signals and speedy assistance

Technology: IBM Watson Studio, IBM Cloud Object Storage, IBM Watson Visual Recognition
Receiving Linux Foundation Support

21

Take on climate change

Participants:
Register for the challenge, and start building

Supporters:
Bring Call for Code into your organization, socialize with your network or donate to the winning team

Sponsors:
Show your full support with a sponsorship

2020 Call for Code announcement	Submission portal opens	Submission portal closes
February 26	March 22	July 30

ibm.biz/callforcode

To the workshop!

https://github.com/IBMdeveloperUK/r-workshops

2020 CALL FOR 2020 CODE®
Global Challenge

Participants:
Register for the challenge, and start building

Supporters:
Bring Call for Code into your organization, socialize with your network or donate to the winning team

Sponsors:
Show your full support with a sponsorship

IBM Call for Code Founding Partner
DCC David Clark Cause Call for Code Creator
United Nations Human Rights Office of the High Commissioner Call for Code Charitable Partner
The Linux Foundation Call for Code Charitable Partner

Join a **movement** of:

- **210,000** developers, data scientists & problem solvers
- **165+** nations
- **8,000+** applications built in the prior 2 years

In line with the UN 75th Anniversary global conversation, help halt & reverse climate change by addressing:

- **energy sustainability**
- **water sustainability**
- **disaster resiliency**

Have the chance to be **awarded:**

- **$200,000** USD
- Open Source support from **The Linux Foundation**
- Meetings with **mentors** & potential **investors**
- **Solution implementation** support through **Code and Response™**

2020 Call for Code announcement	Submission portal opens	Submission portal closes
February 26	March 22	July 30

ibm.biz/callforcode

Introduction to Machine Learning with R and Watson Studio

July 22, 2020

IBM Developer

IBM

Your hosts

Ross Cruickshank

Developer Advocate

driving the screen

https://developer.ibm.com/profiles/ross.cruickshank/

Margriet Groenendijk

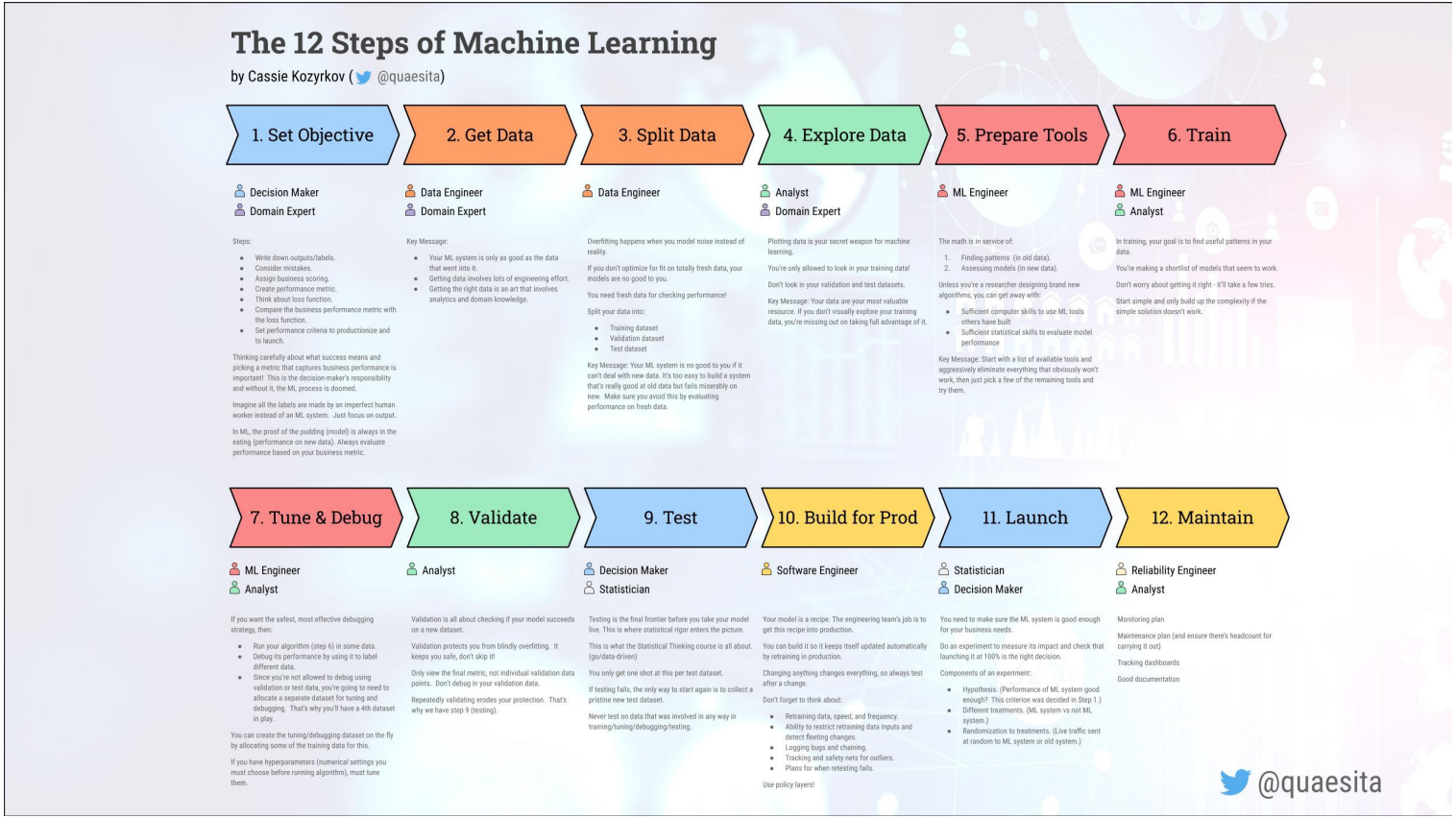
Developer Advocate

driving the Q&A

https://developer.ibm.com/profiles/mgroenen/

(a REAL data scientist!)

https://ibm.biz/mlrstudio @CodeMeetup https://github.com/IBMdeveloperUK/r-workshops


The 12 Steps of Machine Learning
by Cassie Kozyrkov (@quaesita)
1. Set Objective
Decision Maker
Domain Expert
Steps:
Write down outputs/labels.
Consider mistakes.
Assign business scoring.
Create performance metric.
Think about loss function.
Compare the business performance metric with the loss function.
Set performance criteria to productionize and to launch.
Thinking carefully about what success means and picking a metric that captures business performance is important! This is the decision-maker's responsibility and without it, the ML process is doomed.
Imagine all the labels are made by an imperfect human worker instead of an ML system. Just focus on output.
In ML, the proof of the pudding (model) is always in the eating (performance on new data). Always evaluate performance based on your business metric.
2. Get Data
Data Engineer
Domain Expert
Key Message:
Your ML system is only as good as the data that went into it.
Getting data involves lots of engineering effort.
Getting the right data is an art that involves analytics and domain knowledge.
3. Split Data
Data Engineer
Overfitting happens when you model noise instead of reality.
If you don't optimize for fit on totally fresh data, your models are no good to you.
You need fresh data for checking performance!
Split your data into:
Training dataset
Validation dataset
Test dataset
Key Message: Your ML system is no good to you if it can't deal with new data. It's too easy to build a system that's really good at old data but fails miserably on new. Make sure you avoid this by evaluating performance on fresh data.
4. Explore Data
Analyst
Domain Expert
Plotting data is your secret weapon for machine learning.
You're only allowed to look in your training data!
Don't look in your validation and test datasets.
Key Message: Your data are your most valuable resource. If you don't visually explore your training data, you're missing out on taking full advantage of it.
5. Prepare Tools
ML Engineer
The math is in service of:
1. Finding patterns (in old data).
2. Assessing models (in new data).
Unless you're a researcher designing brand new algorithms, you can get away with:
Sufficient computer skills to use ML tools others have built
Sufficient statistical skills to evaluate model performance
Key Message: Start with a list of available tools and aggressively eliminate everything that obviously won't work, then just pick a few of the remaining tools and try them.
6. Train
ML Engineer
Analyst
In training, your goal is to find useful patterns in your data.
You're making a shortlist of models that seem to work.
Don't worry about getting it right - it'll take a few tries.
Start simple and only build up the complexity if the simple solution doesn't work.
7. Tune & Debug
ML Engineer
Analyst
If you want the safest, most effective debugging strategy, then:
Run your algorithm (step 6) in some data.
Debug its performance by using it to label different data.
Since you're not allowed to debug using validation or test data, you're going to need to allocate a separate dataset for tuning and debugging. That's why you'll have a 4th dataset in play.
You can create the tuning/debugging dataset on the fly by allocating some of the training data for this.
If you have hyperparameters (numerical settings you must choose before running algorithm), must tune them.
8. Validate
Analyst
Validation is all about checking if your model succeeds on a new dataset.
Validation protects you from blindly overfitting. It keeps you safe, don't skip it!
Only view the final metric, not individual validation data points. Don't debug in your validation data.
Repeatedly validating erodes your protection. That's why we have step 9 (testing).
9. Test
Decision Maker
Statistician
Testing is the final frontier before you take your model live. This is where statistical rigor enters the picture.
This is what the Statistical Thinking course is all about. (go/data-driven)
You only get one shot at this per test dataset.
If testing fails, the only way to start again is to collect a pristine new test dataset.
Never test on data that was involved in any way in training/tuning/debugging/testing.
10. Build for Prod
Software Engineer
Your model is a recipe. The engineering team's job is to get this recipe into production.
You can build it so it keeps itself updated automatically by retraining in production.
Changing anything changes everything, so always test after a change.
Don't forget to think about:
Retraining data, speed, and frequency.
Ability to restrict retraining data inputs and detect fleeting changes.
Logging bugs and chaining.
Tracking and safety nets for outliers.
Plans for when retesting fails.
Use policy layers!
11. Launch
Statistician
Decision Maker
You need to make sure the ML system is good enough for your business needs.
Do an experiment to measure its impact and check that launching it at 100% is the right decision.
Components of an experiment:
Hypothesis. (Performance of ML system good enough? This criterion was decided in Step 1.)
Different treatments. (ML system vs not ML system.)
Randomization to treatments. (Live traffic sent at random to ML system or old system.)
12. Maintain
Reliability Engineer
Analyst
Monitoring plan
Maintenance plan (and ensure there's headcount for carrying it out)
Tracking dashboards
Good documentation
@quaesita

Machine learning with R in Watson Studio

Steps 3- 9

- Get data
- Split data
- Explore data
- Prepare tools
- Tools
- Tune and debug
- Validate
- Test

Datasets

Libraries/packages

Techniques

Hands-on
- Regression
- Neural networks

Dataset packages

Base R installation

- https://www.rdocumentation.org/packages/datasets/

Machine learning benchmarks – mlbench

- https://www.rdocumentation.org/packages/mlbench/

Top 10 – Jason Brownlee

- https://machinelearningmastery.com/machine-learning-datasets-in-r/

University of California, Irvine – public machine learning datasets

- ftp://ftp.ics.uci.edu/pub/machine-learning-databases

Mango Training datasets

- https://www.rdocumentation.org/packages/mangoTraining/versions/1.1/topics/mangoTraining-package

And many, many more …

Data sets

iris

Edgar Anderson's Iris Data

Plant variant classification by petal characteristics

	Sepal.Length	Sepal.Width	Petal.Length	Petal.Width	Species
1	5.1	3.5	1.4	0.2	setosa
2	4.9	3.0	1.4	0.2	setosa
3	4.7	3.2	1.3	0.2	setosa
4	4.6	3.1	1.5	0.2	setosa
5	5.0	3.6	1.4	0.2	setosa
6	5.4	3.9	1.7	0.4	setosa

Uses:

Sampling

Knn modelling

prediction

Data sets

mtcars

Motor Trend magazine fuel consumption data ‘73-74

	mpg	cyl	disp	hp	drat	wt	qsec	vs	am	gear	carb
Mazda RX4	21.0	6	160	110	3.90	2.620	16.46	0	1	4	4
Mazda RX4 Wag	21.0	6	160	110	3.90	2.875	17.02	0	1	4	4
Datsun 710	22.8	4	108	93	3.85	2.320	18.61	1	1	4	1
Hornet 4 Drive	21.4	6	258	110	3.08	3.215	19.44	1	0	3	1
Hornet Sportabout	18.7	8	360	175	3.15	3.440	17.02	0	0	3	2
Valiant	18.1	6	225	105	2.76	3.460	20.22	1	0	3	1

Uses:

Principal Component Analysis

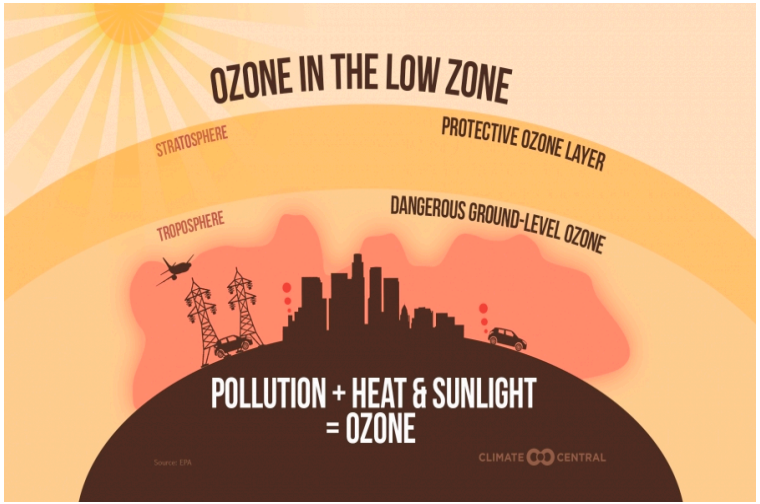
regression

Data sets

airquality

Daily air quality measurements in New York, May to September 1973

	Ozone	Solar.R	Wind	Temp	Month	Day
1	41	190	7.4	67	5	1
2	36	118	8.0	72	5	2
3	12	149	12.6	74	5	3
4	18	313	11.5	62	5	4
5	NA	NA	14.3	56	5	5
6	28	NA	14.9	66	5	6



Uses:

Principal Component Analysis

regression

Data sets

UKgas

UK quarterly gas consumption 1960-86

	date	MTherms
1	1960.00	160.1
2	1960.25	129.7
3	1960.50	84.8
4	1960.75	120.1
5	1961.00	160.1
6	1961.25	124.9

Uses:

Time series analysis - seasonality

Trending, prediction

Data sets

longley

USA economy data 1947-52

	GNP.deflator	GNP	Unemployed	Armed.Forces	Population	Employed
1947	83.0	234.289	235.6	159.0	107.608	60.323
1948	88.5	259.426	232.5	145.6	108.632	61.122
1949	88.2	258.054	368.2	161.6	109.773	60.171
1950	89.5	284.599	335.1	165.0	110.929	61.187
1951	96.2	328.975	209.9	309.9	112.075	63.221
1952	98.1	346.999	193.2	359.4	113.270	63.639

Uses:

Linear regression

Prediction

https://rstudio-pubs-static.s3.amazonaws.com/269760_abe16b38acec471e92021b0ee703efd8.html

Popular libraries

Machine Learning

CARET - https://topepo.github.io/caret/
Classification and Regression Toolkit

KERAS - https://keras.io/
High-level Neural network API

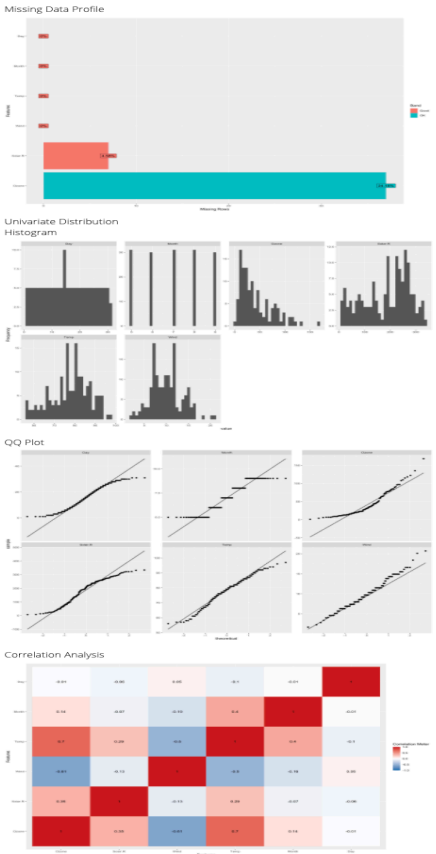
Data exploration, munging, wrangling

Exploratory Data Analysis (Wikipedia)

TIDYVERSE
DPLYR
DataExplorer

GGPLOT2

DataExplorer



https://www.rdocumentation.org/packages/DataExplorer/versions/0.8.0

https://www.rdocumentation.org/packages/DataExplorer/versions/0.8.0/vignettes/dataexplorer-intro.Rmd

There primary reasons for DataExplorer

- Exploratory Data Analysis (EDA)
- Feature Engineering
- Data Reporting

Caret

caret|

Classification **A**nd **RE**gression **T**raining

- data splitting
- pre-processing
- feature selection
- model training
- Model tuning

Aimed at making the ML developer's life easy

Max Kuhn

Keras

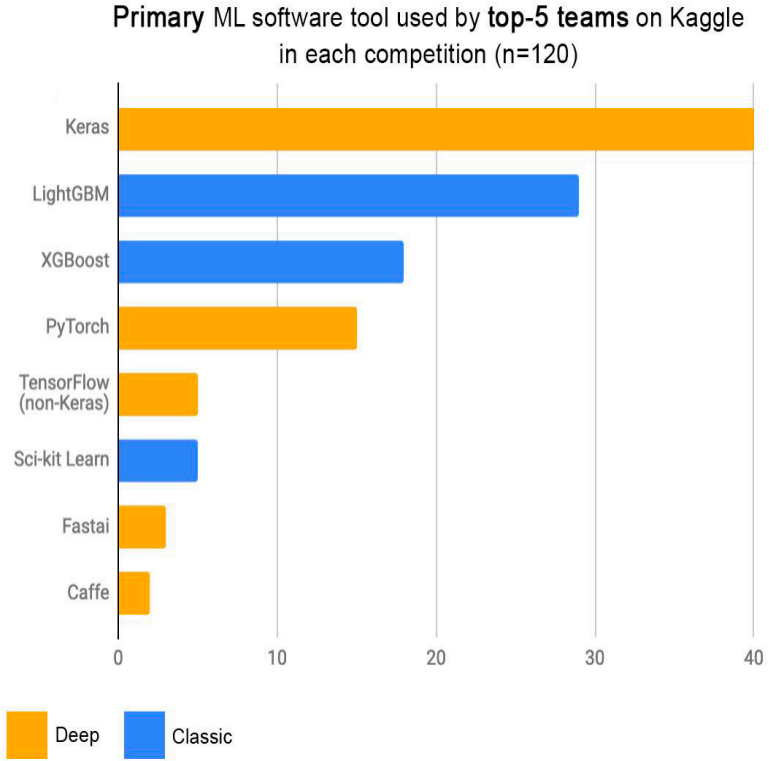
Keras

High-level Machine learning API

Supports multiple underlying frameworks:

- TensorFlow
- Theanos
- PyTorch
- Caffe
- XGBoost
- …

Aimed at making the ML developer's life easy



Route to Production

https://dataplatform.cloud.ibm.com/docs/content/wsj/analyze-data/ml-import-keras.html

Export your model from Rstudio as PMML (https://en.wikipedia.org/wiki/Predictive_Model_Markup_Language)

https://stackoverflow.com/questions/54670786/save-and-deploy-r-model-in-watson-studio

https://www.rdocumentation.org/packages/pmml/versions/1.5.7/topics/pmml.glm

EDA with DataExplorer

Lab

Built-in dataset - mtcars

"eyeball" the dataset

Principle component analysis

https://www.rdocumentation.org/packages/datasets/versions/3.6.2/topics/mtcars

Error rates from 12% (1998) down to 0.23% (2012)

https://arxiv.org/abs/1202.2745

Hands-on lab – hello-DataExplorer.R

(not everything is a notebook ..)

Predict fuel consumption with Caret

Lab

Built-in dataset - mtcars

lLinear regression problem

Principle component analysis

https://www.rdocumentation.org/packages/datasets/versions/3.6.2/topics/mtcars

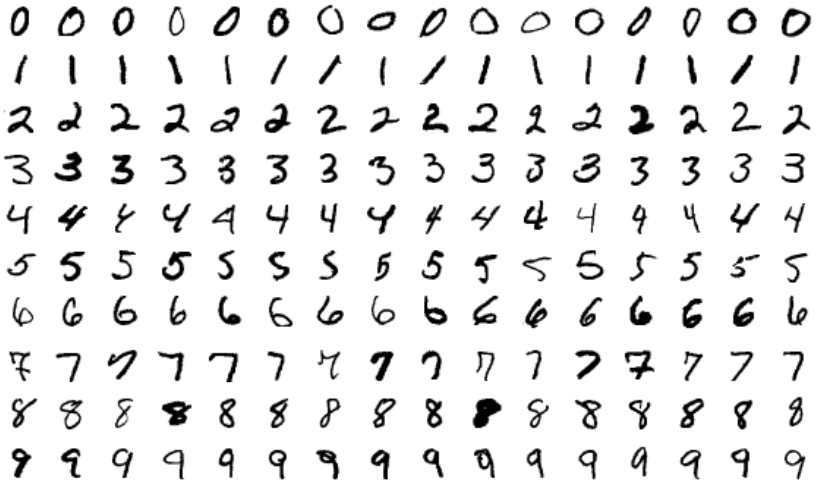
Error rates from 12% (1998) down to 0.23% (2012)

https://arxiv.org/abs/1202.2745

Hands-on lab – hello-caret.Rmd

(based on https://towardsdatascience.com/create-predictive-models-in-r-with-caret-12baf9941236)

Handwriting recognition with Keras



Lab

Very well-known image classification problem

Employ Convolutional neural network (CNN)

Improve accuracy

http://yann.lecun.com/exdb/mnist/

Error rates from 12% (1998) down to 0.23% (2012)

https://arxiv.org/abs/1202.2745

Hands-on lab – hello-keras.Rmd

(based on https://github.com/IBM/using-tensorflow-with-r)

Learning resources

Datacamp

- https://www.datacamp.com/community/tutorials/machine-learning-in-r

University of California, Irvine – public machine learning datasets

- ftp://ftp.ics.uci.edu/pub/machine-learning-databases

Time Series with R introduction

- https://a-little-book-of-r-for-time-series.readthedocs.io/

Keras with R

- https://github.com/IBM/using-tensorflow-with-r
- https://developer.ibm.com/components/keras/

Caret

- https://www.r-bloggers.com/machine-learning-with-r-caret-part-1/

To the workshops!

https://github.com/IBMdeveloperUK/r-workshops