





# Introducción a la Ciencia de Datos con R Miguel Jorquera

Educación Profesional Escuela de Ingeniería

El uso de apuntes de clases estará reservado para finalidades académicas. La reproducción total o parcial de los mismos por cualquier medio, así como su difusión y distribución a terceras personas no está permitida, salvo con autorización del autor.





# RESUMEN

#### **RESUMEN**



#### Manipulación de tablas

- Lectura de archivos locales (csv con read.csv)
- dplyr:
  - Consultas a tablas
    - filter() : Para filtrar los datos (por filas).
    - arrange() : Para ordenar un dataset.
    - select() : Para seleccionar y renombrar columnas.
    - mutate() : Para crear (o modificar) columnas.
    - group\_by() : Para agrupar tablas
    - summarise() : Para generar medidas agregadas.
    - sample\_n() y sample\_frac() : Para generar muestras aleatorias.
  - Cruce entre tablas
    - left\_join
    - right\_join
    - Inner\_join
    - full\_join

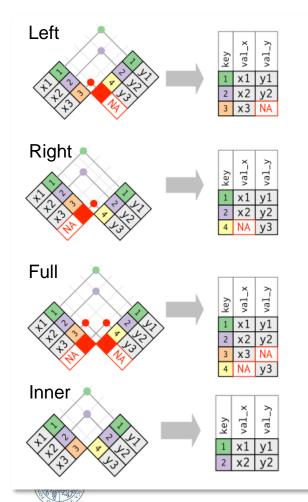


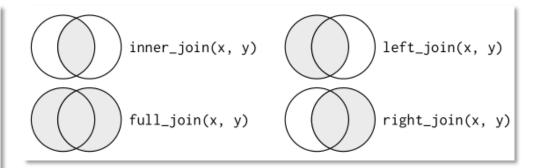


#### **RESUMEN**



#### Manipulación de tablas





dplyr	merge
<pre>inner_join(x, y)</pre>	merge(x, y)
<pre>left_join(x, y)</pre>	merge(x, y, all.x = TRUE)
<pre>right_join(x, y)</pre>	merge(x, y, all.y = TRUE),
<pre>full_join(x, y)</pre>	merge(x, y, all.x = TRUE, all.y = TRUE)

dplyr	SQL
nner_join(x, y, by = "z")	SELECT * FROM $x$ INNER JOIN $y$ USING $(z)$
eft_join(x, y, by = "z")	SELECT * FROM x LEFT OUTER JOIN y USING (z)
ght_join(x, y, by = "z")	SELECT * FROM x RIGHT OUTER JOIN y USING (z)
ll_join(x, y, by = "z")	SELECT * FROM $\times$ FULL OUTER JOIN $y$ USING $(z)$



https://r4ds.had.co.nz/relational-data.html







# TEMAS PARA HOY

#### TEMAS PAR HOY

#### EDUCACIÓN PROFESIONAL

#### Agenda

- Como consumir datos desde una API con R
- Nociones básicas de web scraping con R
- Interacción con base de datos





# Conceptos básicos

- ¿Qué es una API?
- HTTP
- Interacción con APIs
- Package httr
- Request-response







# ¿Que es una API?

- API: Application Programming Interface
- Son instrucciones programadas sobre cómo interactuar con una pieza de software, por ejemplo:
  - Un package e R/Python u otro lenguaje,
  - Otra API (pública por ejemplo)
  - Una base de datos
  - Un sistema operativo





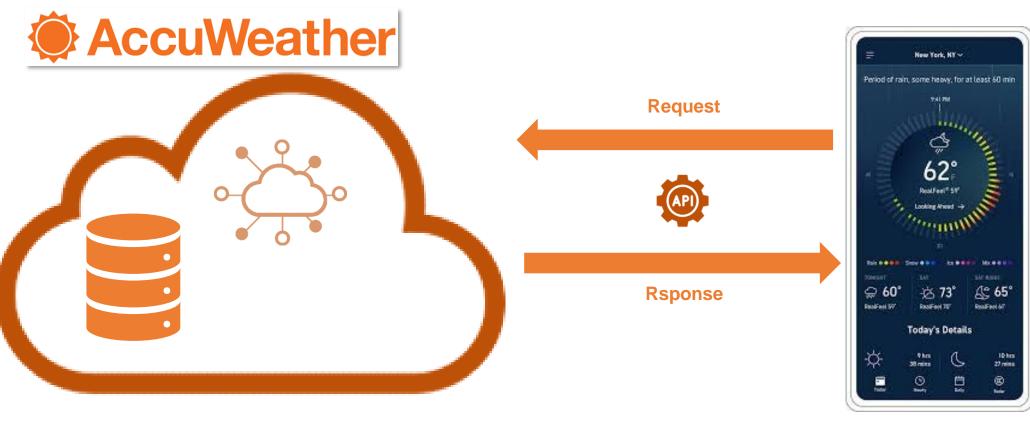
#### **Protocolo HTTP**

- HTTP: Hyper Text Transfer Protocol
- Es un protocolo de comunicación que permite la transferencia de información en la web (World Wide Web)
- Algunas características
  - Existen verbos o métodos para diferentes acciones (GET, HEAD, POST, PUT, PATCH, DELETE, CONNECT, OPTIONS y TRACE.)
  - Usualmente se requiere una autenticación para poder interactuar con la web
  - Existe códigos de estados
  - La mayoría de las (REST) APIs usan HTTP para la transferencia de datos.



#### EDUCACIÓN PROFESIONAL

# Protocolo HTTP y APIs







#### Package httr

 Desde R, podemos utilizar métodos del protocolo HTTP, mediante el package httr

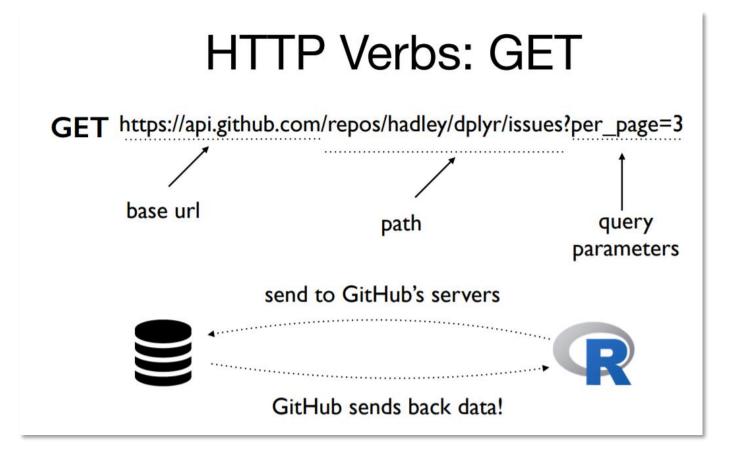
```
library(httr)
x <- GET('https://google.com/')</pre>
x$status_code
#>[1] 200
x$headers
#> $date
#> [1] "Thu, 23 Jun 2016 23:05:27 GMT"
#> ...
x$content
#>[1] 3c 21 64 6f 63 74 79 70 65 20 68 ...
```





#### Request

 Cuando consultamos una API, esta usualmente requiere parámetros según el método (verbo) que utilicemos. Por ejemplo GET:







#### Response

- La respuesta ante la utilización de un verbo posee una estructura de salida, la cual puede venir contenida en un json, código xml, o en formato raw. Los compontes de la resuesta son:
  - status: Estado de la respuesta
  - headers: metada de la respuesta (tipo de contenido, info de la página, tamaño de la consulta, etc)
  - body/content: Contenido de salida



#### **EJEMPLO**







API REFERENCE GENERAL INFO →

FAQS PACI

PACKAGES & PRICING

Search this site...

#### Packages

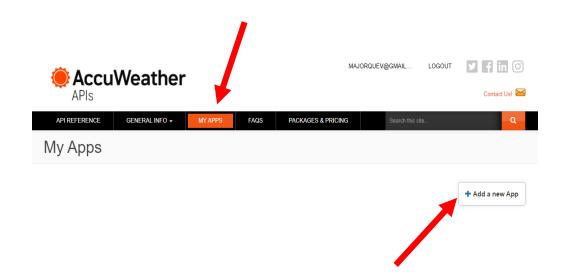
Limited Trial	Standard		Prime		Elite	
Locations	Locations		Locations		Locations	
Current Conditions	Current Conditions		Current Conditions		Current Conditions	
24-Hour Historical Current Conditions	24-Hour Historical Current Conditions		24-Hour Historical Current Conditions		24-Hour Historical Current Conditions	
Forecast 5-day	Forecast 5-day		Forecast 10-day		Forecast 15-day	
Forecast 12-hour	Forecast 12-hour		Forecast 72-hour		Forecast 120-hour	
Indices 5-day	Indices 5-day		Indices 10-day		Indices 15-day	
	Alarms 5-day		Alarms 10-day		Alarms 15-day	
	Translations		Translations		Translations	
			Tropical		Tropical	
			Alerts		Alerts	
			Imagery		Imagery	
	Standard Pricing		Prime Pricing		Elite Pricing	
50 calls/day	Base Monthly Fee	CPM Rate	Base Monthly Fee	CPM Rate	Base Monthly Fee	CPM Rate
Limit: 1 key/developer Free: Get started now!	\$25.00 per month	\$0.12 CPM over 225,000 calls per month	\$250.00 per month	\$0.15 CPM over 1,800,000 calls per month	\$500.00 per month	\$0.22 CPM over 2,400,000 calls per month
	Purchase Standard Package		Purchase Prime Package		Purchase Elite Package	

Need more? To hear about additional content, climatology data, high volume CPM rates, or other business opportunities, please contact our sales team at sales@accuweather.com to learn about what else AccuWeather has to offer!

BRANDING REQUIREMENTS: You are required to display proper attribution when using AccuWeather APIs. Please include the AccuWeather logo, linked to our website (http://www.accuweather.com) in clear view on your app or product. AccuWeather logo with linking must appear on all pages/screens where AccuWeather content appears, as stated in the Terms of Use.

ESCUELA DE INGENIERÍA
FACULTAD DE INGENIERÍA

- API de Accuweather <u>https://developer.accuweather.com/</u>
  - Es necesario regitrarse (pide solo mail)
  - Para usar la API deben crear un API KEY en la sección de Aplicaciones.



#### **EJEMPLO**



#### **Accuweather API**

Documentación y referencias de uso (sintaxis, parámetros, formatos, etc)



GENERAL INFO ▼

MAJORQUEV@GMAIL....

LOGOUT









API REFERENCE

MY APPS

**FAQS** 

**PACKAGES & PRICING** 

a

#### **API** Reference

#### Locations API

Get a location key for your desired location. Use the location key to retrieve weather data from the Forecast or Current Conditions API.

#### Forecast API

Get forecast information for a specific location.

#### **Current Conditions API**

Get Current Conditions data for a specific location.

#### Indices API

Get daily index values for a specific location. Index availability varies by location.

#### Weather Alarms API

Get Weather Alarms for a specific location. AccuWeather Weather Alarms are determined using the daily forecasts for a location. An alarm exists for a location if the forecast weather meets or exceeds specific thresholds.





#### EDUCACIÓN PROFESIONAL

# Accuweather API





#### INTRO: WEB SCRAPING CON R



# Conceptos básicos

Para esta sección utilizaremos el siguiente recurso online

https://cfss.uchicago.edu/notes/web-scraping/



### INTERACCIÓN CON BASE DE DATOS



#### **Conexiones**

Existe una gran variedad de opciones de conexión a bases de datos. Algunas de ellas reciben soporte de Rstudio.com y entre ellas destacan:

- Microsoft SQL Server
- Oracle
- Teradata
- PostgreSQL
- Apache Hive
- Apache Impala
- Apache Cassandra

- Amazon Athena
- Amazon Redshift
- MongoDB
- Google BigQuery
- IBM Netezza
- Salesforce
- MySQL



https://db.rstudio.com/

