

ACÀMICA

---

# ¡Bienvenidos/as a Data Science!



# Agenda

---

Presentación del Equipo Docente y de la metodología Académica

Actividad: Presentación de estudiantes

Carrera: lineamientos, estructura y proyectos

Herramientas a utilizar: plataforma Académica, Slack, Trello

Break

Introducción a Python

Actividad: instalamos Python y creamos nuestro primer proyecto

Cierre

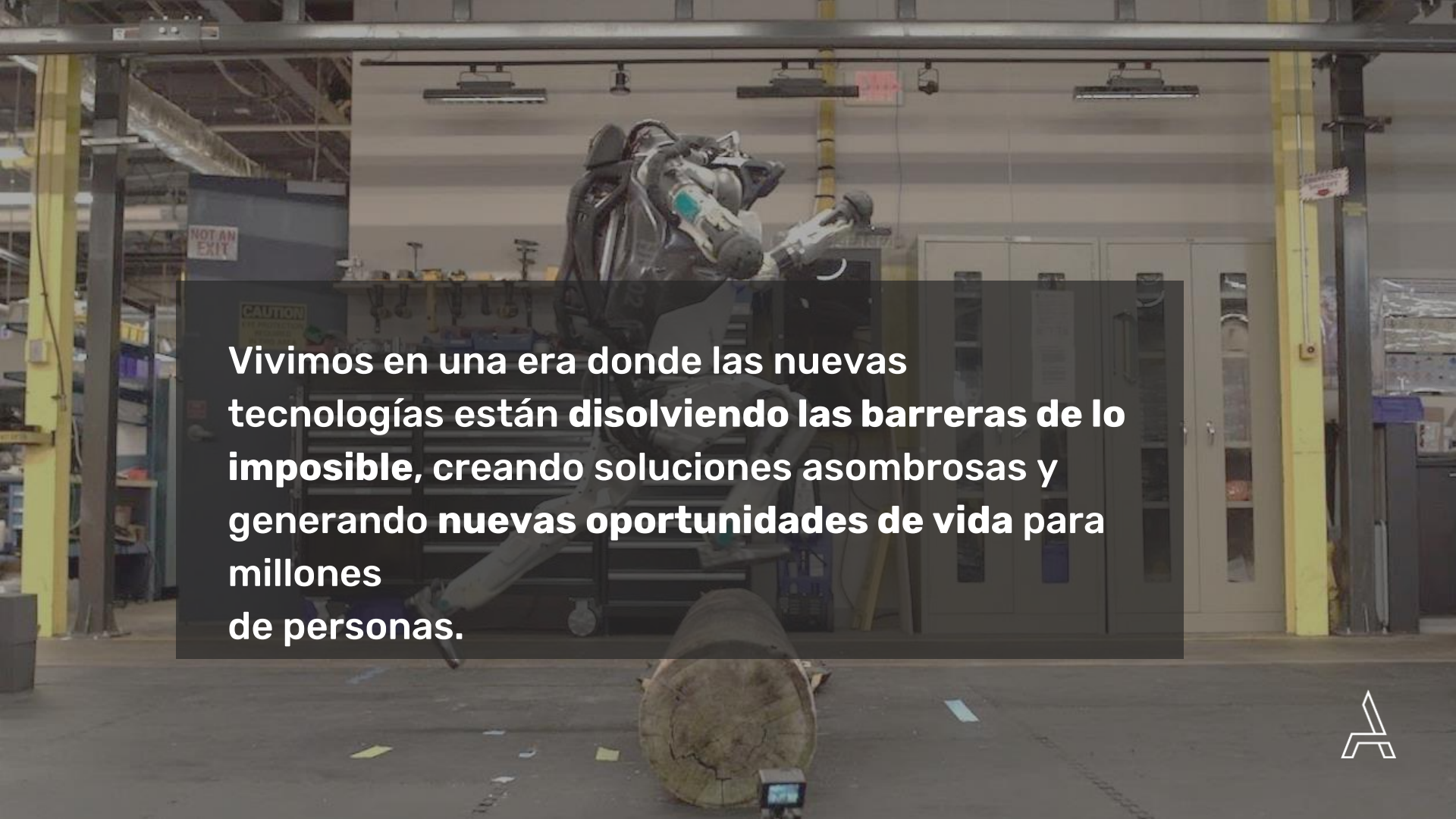


# Equipo Docente



# Acámica





Vivimos en una era donde las nuevas tecnologías están **disolviendo las barreras de lo imposible**, creando soluciones asombrosas y generando **nuevas oportunidades de vida** para millones de personas.



A large industrial robot arm is positioned in a factory setting, with a red semi-transparent text box overlaid on the image. The robot arm is grey and white, with a blue light on its joint. It is surrounded by various industrial equipment, including a yellow crane, a blue storage unit, and a large metal cylinder in the foreground. The background shows a factory floor with yellow pillars and a grey door.

**Pero la educación tradicional no logra acompañar la velocidad de esta transformación, dejando atrás a personas y organizaciones, provocando una enorme brecha de talento digital.**



**Somos una**  
**academia tecnológica**  
**que acompaña a las personas**  
**en su transformación digital.**





**Buscamos empoderar  
a las personas para  
ser protagonistas  
de la transformación  
del mundo.**





# nuestro MINDSET

---

## **#Student First**

Todas las decisiones que tomamos están enfocadas en mejorar la experiencia de aprendizaje de nuestros estudiantes y darles una educación de primer nivel.

## **#Tech Driven**

Nos apoyamos en la tecnología para dar una educación personalizada a escala, tanto en la distribución de conocimiento como en la interacción con nuestros alumnos y cuerpo de mentores.

## **#Community Based**

Contratamos a mentores que trabajan de lo que enseñan, y diseñamos planes de estudio en función de las necesidades de las empresas de tecnología.

## **#Global Mindset**

Nuestros planes de estudio y metodologías están diseñados para que cualquier persona en cualquier parte del mundo pueda aprender y convertirse en un/a profesional competente en el mundo digital.



## nuestros VALORES

---

**Pasión por aprender**

**Motivación por hacer**

**Entusiasmo por colaborar**



# nuestros VALORES

---

## **Pasión por aprender**

**Aprendemos continuamente con curiosidad y humildad.**

**Desafiamos constantemente nuestros límites.**

**Capitalizamos aciertos y aprendemos de nuestros errores.**

**Compartimos información abierta y proactivamente.**

**Valoramos todos los puntos de vista.**



# nuestros VALORES

---

## **Motivación por hacer**

**Innovamos para generar valor.**

**Nos hacemos cargo de los resultados y los problemas.**

**Inspiramos a otros trabajando con excelencia.**

**Tomamos decisiones basadas en datos.**

**Ejecutamos con agilidad.**



# nuestros VALORES

---

## **Entusiasmo por colaborar**

**Disfrutamos y valoramos trabajar en equipo.**

**Nos fortalecemos con nuestras diferencias.**

**Colaboramos de manera organizada e inteligente.**

**Damos feedback sincero y constructivo.**

**Reconocemos y celebramos nuestros logros.**

**Compartimos junto a nuestros estudiantes.**



# nuestra METODOLOGÍA

---

## **Aprendemos haciendo**

La creación de productos concretos permite a los/as estudiantes generar soluciones creativas a problemáticas específicas.

## **Damos feedback que forma**

Más allá de certificar que hayan aprendido los contenidos técnicos, buscamos ofrecer un feedback que los ayude a entender sus aciertos y oportunidades de mejora a lo largo del proceso de aprendizaje.

## **Aprendemos con otros/as**

Promovemos clases con multiplicidad de disciplinas en las que se aprende co-creando y colaborando con pares y equipo docente.

## **Formamos profesionales**

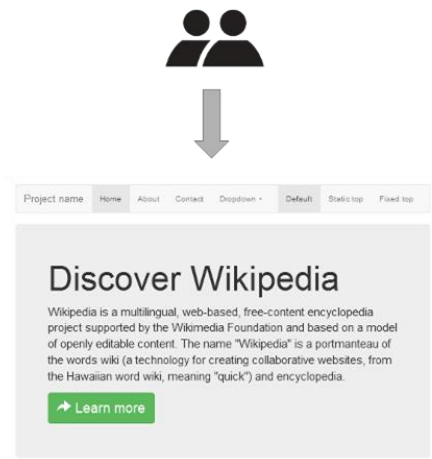
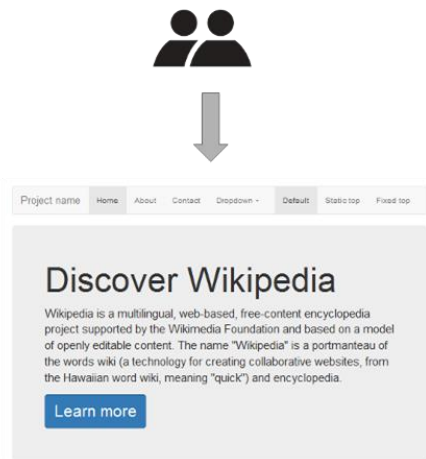
Trabajaremos la comunicación, la empatía y la colaboración en distintas instancias para formar egresados que aporten mucho más que saber técnico en los lugares de trabajo a los que ingresen.

# Actividad: Nos vamos conociendo...





# Veamos un ejemplo: A/B Testing



**¿CÓMO HARÍAN  
para saber cuál  
logra un mayor  
número de *clics*  
(conversiones)?**

# ¿Qué es Data Science?

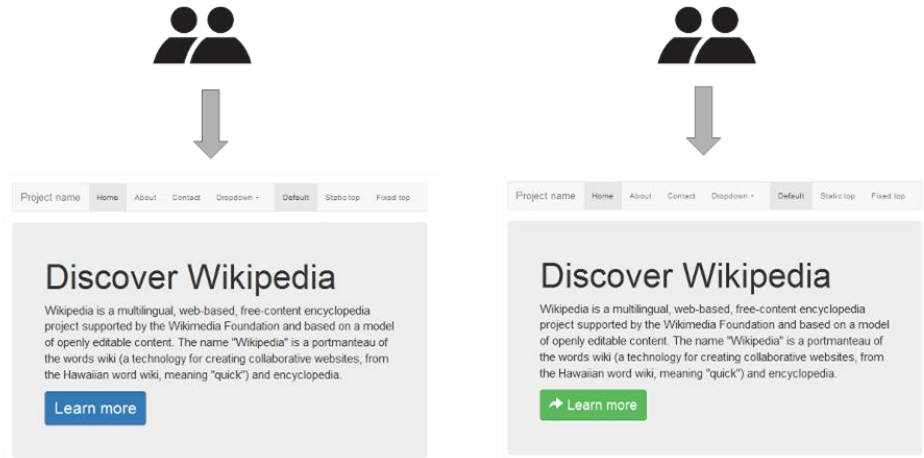


# Veamos un ejemplo: A/B Testing

Wikipedia quiere  
modificar su página  
Web de forma tal de  
obtener más  
donaciones.

Consultan con  
expertos en UX/UI que  
les proponen dos  
modelos parecidos  
pero con algunas  
diferencias mínimas.

# Veamos un ejemplo: A/B Testing



# Veamos un ejemplo: A/B Testing

**Propuesta:** dividir algunos usuarios en dos grupos al azar. Al primer grupo le mostramos la versión **A** del diseño de la página, y al segundo grupo les mostramos el diseño **B**. Luego, contamos cuántos usuarios de cada grupo hace *click*.

¿Convirtió?	A	B
Sí	3	2
No	7	8
Total	10	10

**¿Se animan a decir que la versión A obtiene más conversiones que la B?**

# Veamos un ejemplo: A/B Testing

**Propuesta:** dividir algunos usuarios en dos grupos al azar. Al primer grupo le mostramos la versión **A** del diseño de la página, y al segundo grupo les mostramos el diseño **B**. Luego, contamos cuántos usuarios de cada grupo hace *click*.

¿Y ahora?

¿Convirtió?	A	B
Sí	3	2
No	7	8
Total	10	10

¿Convirtió?	A	B
Sí	350	175
No	650	825
Total	1000	1000

# Veamos un ejemplo: A/B Testing

Intuitivamente, esperamos que cuantas más personas vean cada diseño, más **confiable** va a ser la diferencia que encontremos.

Y, cuánto más *marcada* la diferencia que veamos, menos probable que se deba a un **efecto aleatorio**.

# El Data Scientist ayuda a ...



## **Decidir cómo hacer el experimento.**

¿Cómo dividimos los grupos?

¿Cuántos miembros necesitamos en cada grupo?

¿De qué forma nos aseguramos que los resultados que obtengamos sean **generales**?

## **Analizar sus resultados.**

Las diferencias que encontramos, ¿son significativas o son aleatorias?

¿Necesitamos hacer más experimentos o estos resultados son concluyentes?

## **Comunicar los resultados de una forma eficiente. Y más.**



# ¿Qué es Data Science?

Es un campo **interdisciplinario** tanto en sus objetivos como en sus metodologías que busca:

## **DEFINICIÓN**

Se definen las preguntas que queremos responder.

¿Cuáles datos necesitamos para responder esas preguntas?

## **INVESTIGACIÓN**

Se obtienen los datos, se “limpian” y se procede a explorarlos.

## **ANÁLISIS**

Los datos obtenidos se analizan con modelos (estadísticos, Machine Learning, etc.). Interpretamos los resultados y transformamos datos en información.

## **PRESENTACIÓN**

Presentamos los resultados obtenidos y las conclusiones a las que llegamos. Puesta en producción.

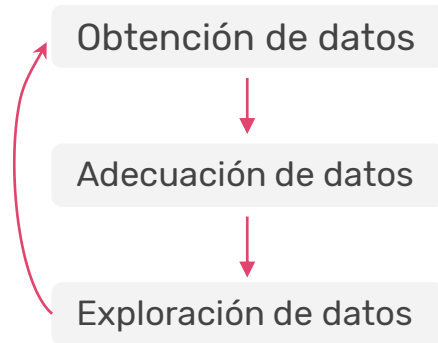
# Es un proceso iterativo

DEFINICIÓN

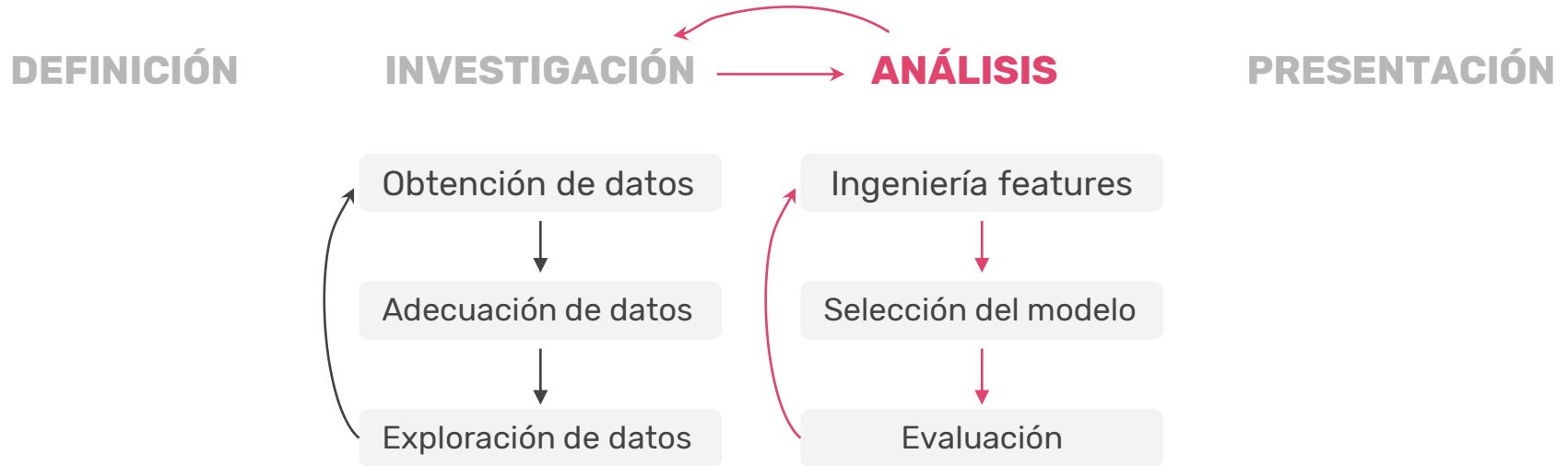
**INVESTIGACIÓN**

ANÁLISIS

PRESENTACIÓN



# Es un proceso iterativo



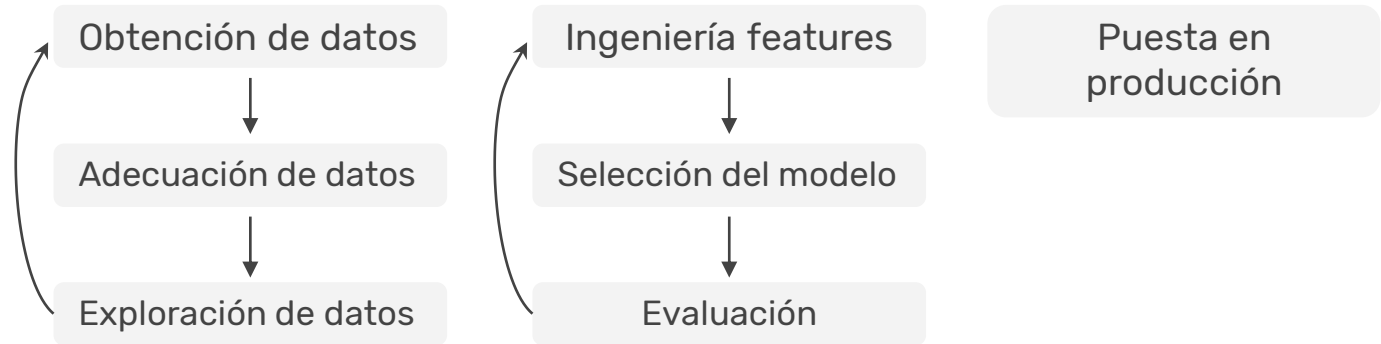
# Es un proceso iterativo

## DEFINICIÓN

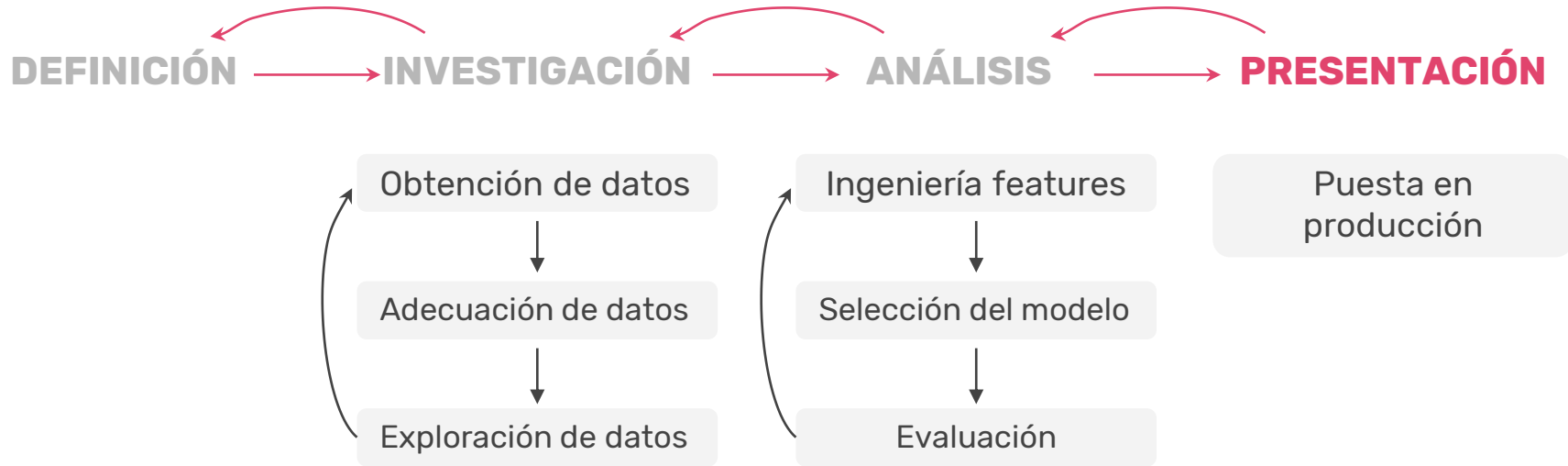
## INVESTIGACIÓN

## ANÁLISIS

## PRESENTACIÓN



# Es un proceso iterativo





DATA

# Data Scientist: The Sexiest Job of the 21st Century

by [Thomas H. Davenport](#) and [D.J. Patil](#)

FROM THE OCTOBER 2012 ISSUE

*“More than anything, what data scientists do is make discoveries while swimming in data. [...].*

*[...] they are able to bring structure to large quantities of formless data and make analysis possible. They identify rich data sources, join them with other, potentially incomplete data sources, and clean the resulting set.*

*In a competitive landscape where challenges keep changing and data never stop flowing, data scientists help decision makers shift from ad hoc analysis to an ongoing conversation with data.”*



# Herramientas fundamentales de un DATA SCIENTIST



- 1 Conocimientos de **matemática y estadística**
- 2 Conocimientos de **programación**
- 3 Algunos conocimientos del **tema específico**
- 4 Habilidad de **comunicar** resultados



# Parientes y amigos

Probabilidad y Estadística

Bases de Datos

Data Mining, Big Data

Machine Learning - Aprendizaje Automático

Deep Learning - Redes Neuronales

Inteligencia Artificial

# Carrera



6 meses

2 encuentros por semana

4 proyectos (7 entregas)



# Cronograma de carrera

fase	BLOQUE 1 ADQUISICIÓN Y EXPLORACIÓN		BLOQUE 2 MODELADO				BLOQUE 3 DEPLOY
	Exploración de datos	Feature Engineering	Regresión	Optimización de parámetros	Procesam. del lenguaje natural	Sistema de recomendación	Publicación de modelos
tiempo	SEM 1	SEM 5	SEM 7	SEM 11	SEM 13	SEM 18	SEM 22
	SEM 2	SEM 6	SEM 8	SEM 12	SEM 14	SEM 19	SEM 23
	SEM 3		SEM 9		SEM 15	SEM 20	SEM 24
	SEM 4		SEM 10		SEM 16	SEM 21	
					SEM 17		



# ¿Tengo que saber algo de antemano?

**¡SÍ!**

- 1) Probabilidad y estadística (básico)
- 2) Programar (cualquier lenguaje, básico)
- 3) Usar la terminal (Windows, Mac, Ubuntu, etc.)

**MUY  
ÚTIL**

Saber leer inglés



# Cursada • Modalidad

---



## ENCUENTROS

Teoría & Ejercitación práctica

Repaso de temas vistos en encuentros anteriores y/o en plataforma

Diremos lo que tienen que ver en plataforma para el encuentro siguiente  
+ tareas adicionales según corresponda

+



## PLATAFORMA

Teoría introductoria

Guías y checklists para realizar proyectos

Entrega de proyectos

## **Proyectos • ¿Cómo me gradúo en Acámica?**

---

**Para graduarse / certificarse,  
ES OBLIGATORIO tener aprobados  
todos los proyectos de la carrera.**

# Proyectos • Condiciones de aprobación

---

Los/as evaluadores/as considerarán una entrega como **Aprobada** cuando el/la estudiante haya cumplido satisfactoriamente el 100% de los puntos que pide el checklist (aunque no los haya hecho a todos perfectos).

Caso contrario, el/la evaluador/a considerará la entrega como **Para rehacer**.

*Nota: no hay un límite de iteraciones (el/la estudiante puede tener que rehacer su trabajo todas las veces que sea necesario hasta aprobar).*



# Proyectos • Evaluadores

---

**Sus proyectos serán corregidos por evaluadores externos con el fin de:**

- Asegurar una **evaluación uniforme** respecto de estas habilidades técnicas, que mantenga los estándares acordados en los programas académicos definidos por Acámica
- Emular metodologías laborales en las cuales **quien recepciona los trabajos finales no es parte del equipo que desarrolló el proyecto**
- Sumar **otra voz profesional** en el proceso de aprendizaje de la persona evaluada
- Promover un sistema de **evaluación justo, neutral, a la vez que preciso y profundo**

# Proyectos • Evaluadores

---

## ¿Tienes dudas sobre la corrección de alguno de tus proyectos?

- Si no comprendes algo de la evaluación que recibiste, podrás escribirle al evaluador/a que corrigió tu proyecto por Slack para despejar dudas. Nota: no es responsabilidad del evaluador/a hacer mentoreo sobre los proyectos o contestar preguntas no relacionadas a una entrega corregida por él/ella.
- Si consideras que tu evaluación debe ser revisada, envía un correo a [hola@acamica.com](mailto:hola@acamica.com) solicitando la revisión.
- Si la devolución de una entrega que realizaste tarda más de 5 días hábiles, escribe a [hola@acamica.com](mailto:hola@acamica.com) para elevar el reclamo (si sabes el nombre del evaluador/a que tomó tu proyecto, puedes también escribirle por Slack para ver el estado de la evaluación).

# Recomendaciones para proyectos

- 1) No podrás entregar un proyecto si no entregaste el anterior.
- 2) Agenda las fechas de entrega en tu calendario para no atrasarte en la clase.
- 3) Descarga los “Checklist” antes de comenzar tus proyectos para tener claridad sobre qué esperamos que entregues.
- 4) Los “Checklist” son la base. ¡Recomendamos ejercitar tu creatividad y personalizar los proyectos con tu impronta!

**Extra:** durante la cursada incentivamos el trabajo en proyectos propios, ya sea de forma individual o grupal, y la práctica de presentaciones orales.



# Bibliografía MUY útil

Introducción a python:

<https://learnxinyminutes.com/docs/python3/>



Introducción a algunas de las librerías (y *bastante* más):

<https://jakevdp.github.io/PythonDataScienceHandbook/>

Más en <https://github.com/acamica/biblio-ds>



Libro Divulgativo/Introdutorio pero muy cercano a los contenidos de la carrera: *big data*, de Walter Sosa Escudero



# Herramientas



## ACÁMICA

---

### VIDEOS Y RECURSOS

Aquí podrán ver videos con contenido teórico de la carrera, y materiales para realizar los proyectos.



### REPOSITORIO DE MATERIAL DE ENCUENTROS

Aquí subimos lo visto en encuentros, el Plan de estudios, links, bibliografía.



### COMUNICACIÓN CLASE

¡Aquí nos comunicamos!

# ACÁMICA

---

## VIDEOS Y RECURSOS

Aquí podrán ver videos con contenido teórico de la carrera, y materiales para realizar los proyectos.



ACÁMICA

---



---

**REPOSITORIO DE  
MATERIAL DE  
ENCUENTROS**

Aquí subimos lo visto en  
encuentros, el Plan de  
estudios, links, bibliografía.



ACÁMICA

---



---

**COMUNICACIÓN CLASE**

¡Aquí nos comunicamos!

ACÁMICA



**¿Tienen acceso?**

# ¿Cómo dar feedback sobre tu experiencia en Acámica?



# ¡Queremos escucharte!

## Tu equipo docente

Puedes plantearles tus dudas, sugerencias o reportar problemas que tengas con los encuentros, contenidos, sedes, conectividad... ¡ESTAMOS PARA ACOMPAÑARTE!

## Encuestas

Haremos 4 encuestas de satisfacción a lo largo de la carrera:

- **1 checkpoint** en 2 semanas, para saber cómo comenzó tu experiencia
- **3 encuestas modulares** (al finalizar cada bloque)

## Community Care

Escríbenos a [hola@acamica.com](mailto:hola@acamica.com) si tienes dudas o problemas que excedan a los encuentros o no puedas resolver con los docentes.

A close-up photograph of a white ceramic cup filled with a latte. The surface of the milk is decorated with intricate latte art, featuring a central heart shape surrounded by concentric, wavy lines. The cup is placed on a matching white saucer. In the background, a blurred white napkin and a silver fork are visible on a dark, textured surface. The overall lighting is soft and focused on the cup.

**¡BREAK!**

---

# ¿Qué es programar?



*“Programar es darle instrucciones a la computadora para que realice una función específica.”*

*“Programar es darle instrucciones a la computadora para que realice una función específica.”*



**¡ESTO NO ES UNA  
CARRERA DE  
PROGRAMACIÓN!**



**¿Y cómo lo  
vamos a hacer  
en esta carrera?**





General purpose and high level programming language.



- Fácil de usar
- Rápido y eficiente
- Gran comunidad online
- Ampla cantidad de **librerías** específicas (¡pronto veremos qué son!)

¡EMPEZAMOS!

## 1. Instalar Python

Vamos a instalar una distribución particular: [Miniconda](#).

Si ya tienen instalado Anaconda está perfecto. Si tienen otra distribución, instalar Miniconda preferiblemente.

1) Ir al link y descargar la versión correspondiente a su sistema operativo. **Importante:** Python 3

[Miniconda](#) 🔗

	Windows	Mac OS X	Linux
Python 3.7	<a href="#">64-bit (exe installer)</a>	<a href="#">64-bit (bash installer)</a>	<a href="#">64-bit (bash installer)</a>
	<a href="#">32-bit (exe installer)</a>	<a href="#">64-bit (.pkg installer)</a>	<a href="#">32-bit (bash installer)</a>

2) ¡Instalar!

3) Poner que “sí” a las preguntas que les haga

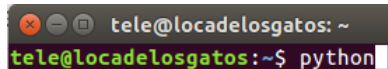


¡EMPEZAMOS!

## 2. Comprobar instalación

Vamos a instalar una distribución particular: [Miniconda](#).

1) Abrir una terminal (¡¿Qué es eso?!)



```
tele@locadelosgatos: ~  
tele@locadelosgatos:~$ python
```

2) Tipear “python”

3) Fijarse qué versión de Python les aparece

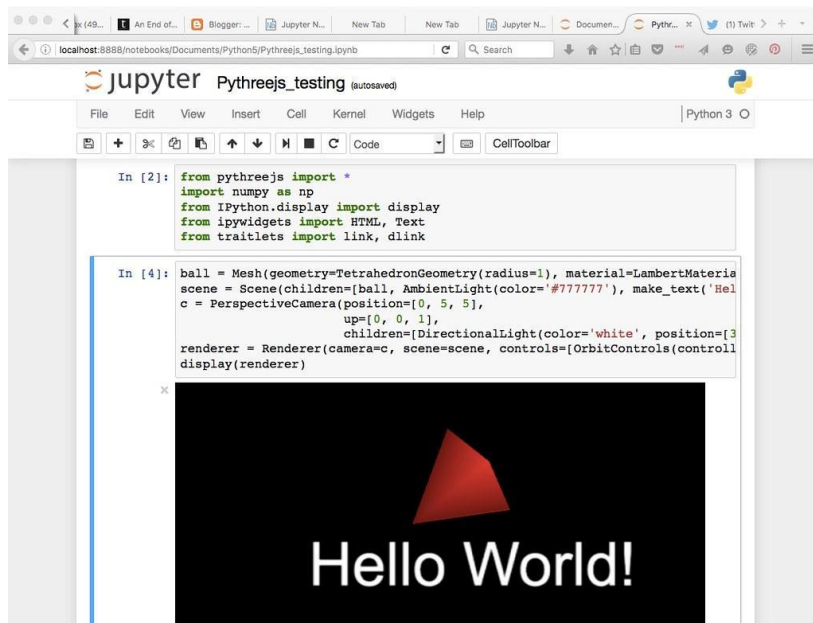
4) Poner “2+5”



¡EMPEZAMOS!

### 3. Abrir entorno de trabajo en Jupyter

Como trabajar desde la terminal es incómodo, vamos a usar notebooks de Jupyter:



**¡EMPEZAMOS!**

## 4. Crear un environment

Conda nos permite crear ambientes (environments) de trabajo e instalar librerías.

No es obligatorio para trabajar con Python, pero sí conveniente.

Al principio parece un poco oscuro. No se preocupen, ya va a quedar más claro.



¡EMPEZAMOS!

## 4. Crear un environment (cont.)

¿Cómo se hace?

1. Poner en la terminal “conda create --name datascience”.
2. Activar el ambiente: “conda activate datascience”
3. Instalar las librerías Jupyter, Notebook y JupyterLab: “conda install jupyter notebook jupyterlab”
4. Comprobar que anduvo: “jupyter lab”. Debería abrirles una pestaña en su navegador.





¡EMPEZAMOS!

## 4. Crear un environment (cont.)



The screenshot displays the JupyterLab interface. On the left, a file browser shows the directory structure of a project named 'Acamica / DS-BUE-4-2019'. It lists various files, including datasets, notebooks, and scripts, along with their last modified dates. On the right, the 'Launcher' tab is active, showing options to create a new environment. The 'Notebook' section features a 'Python 3' icon. The 'Console' section also features a 'Python 3' icon. Below these, the 'Other' section includes icons for 'Terminal', 'Text File', 'Markdown File', and 'Contextual Help'.

Name	Last Modified
Datasets	17 days ago
imagenes_nst	6 days ago
clase_02_python_respuestas.ipynb	7 days ago
clase_02_python.ipynb	2 days ago
clase_03_numpy_docentes.ipynb	6 months ago
clase_03_numpy_respuestas.ipynb	2 days ago
clase_03_numpy.ipynb	2 days ago
clase_04_matplotlib_estadistica_do...	5 months ago
clase_04_matplotlib_estadistica_res...	5 months ago
clase_04_matplotlib_estadistica.ipynb	2 days ago
clase_05_Pandas_docentes.ipynb	5 months ago
clase_05_Pandas.ipynb	5 months ago
clase_06_titanic_docentes.ipynb	3 months ago
clase_06_titanic_resuelto.ipynb	3 months ago
clase_06_titanic.ipynb	3 months ago
clase_07_nacimientos.ipynb	3 months ago
clase_08_impares.ipynb	3 months ago
clase_08_pares.ipynb	3 months ago
clase_09_features_docentes.ipynb	3 months ago
clase_09_features.ipynb	2 days ago
clase_10_features_docentes.ipynb	3 months ago
clase_10_features.ipynb	2 days ago
clase_11_outliers&norm.ipynb	2 days ago

Launcher

Dropbox/Acamica/DS-BUE-4-2019

Notebook

Python 3

Console

Python 3

Other

Terminal

Text File

Markdown File

Contextual Help

# Primeros pasos con Python

## 1. Crear un Notebook vacío

# Primeros pasos con Python

## 1. Crear un Notebook vacío

## 1. Crear una variable *nombre* y asignarle su nombre

```
[ ]: nombre = 'Esteban'
```

¿Qué ocurre si no ponemos las comillas? ¿Y si en lugar de comillas simples usamos dobles?

# Primeros pasos con Python

## 1. Crear un Notebook vacío

### 1. Crear una variable *nombre* y asignarle su nombre

```
[ ]: nombre = 'Esteban'
```

¿Qué ocurre si no ponemos las comillas? ¿Y si en lugar de comillas simples usamos dobles?

### 1. Crear una variable *edad* y asignarle su edad

```
[ ]: edad = 31
```

¿Qué ocurre si ponemos el número entre comillas? ¿Cuál será la diferencia? ¿Y si ponemos un número “con coma”?

# Primeros pasos con Python

## 1. Crear un Notebook vacío

### 1. Crear una variable *nombre* y asignarle su nombre

```
[ ]: nombre = 'Esteban'
```

¿Qué ocurre si no ponemos las comillas? ¿Y si en lugar de comillas simples usamos dobles?

### 1. Crear una variable *edad* y asignarle su edad

```
[ ]: edad = 31
```

¿Qué ocurre si ponemos el número entre comillas? ¿Cuál será la diferencia? ¿Y si ponemos un número “con coma”?

### 1. Imprimir en pantalla su *nombre* y *edad*

```
[ ]: print(nombre, edad)
```

¿Alguna forma es mejor?

```
[ ]: print('Mi nombre es', nombre, '. Mi edad es ', edad, 'años')
```

```
[ ]: print('Mi nombre es {}. Mi edad es {} años'.format(nombre, edad))
```

# Hands-on training: Ejercicio de código



# 1. La edad de mi compañero@

Copiar la edad del compañero/a que esté más cerca.

1. Obtener la diferencia entre tu edad y la de él/ella.

2. Decidir (usando Python) cuál edad es mayor.

```
[ ]: edad < edad_compa
```

3. Sumarle 50 años a la edad menor

```
[ ]: edad = edad + 50
```

→ Supusimos que  
la edad menor  
era *edad*

¿Sigue siendo mayor nuestra *edad* o *edad\_compa*?



## 2. Lista numérica

1. Imprimir en pantalla los primeros 10 números naturales (0,...,9)

¿Siempre tendremos que escribir los números en una lista "a mano"?

```
[ ]: primeros_10 = [0,1,2,3,4,5,6,7,8,9]
```

2. Imprimir únicamente los cinco primeros números pares.

¿Habrá una forma mejor de hacerlo?





# Recomendaciones para programar

---

- 1) Comentar el código en voz alta ayuda a aprender y a entender lo que estás haciendo.
- 2) No tengas miedo de hacer, romper y arreglar.
- 3) La frustración es una buena señal (“Get things done”).
- 4) Pedir la opinión de tus compañeros/as y mentores/as sobre tu código.
- 5) Busca crecer en comunidad (Medium, Github, Slack Stackoverflow, etc).
- 6) Pide ayuda a tu mejor amigo:

Google Search

I'm Feeling Lucky

Search by voice

# Para la próxima

---

- 1) Suscribirse a los canales (Plataforma, Trello y Slack)
- 2) Ver los videos de la plataforma hasta Herramientas de Data Science (nivel II)
- 3) Si no lo hicieron, instalar una versión de Python y crear el ambiente.
  - a) Abrir un entorno de trabajo en Jupyter



A person with curly hair and glasses, wearing a backpack, is seen from behind, looking out over a vast mountain range. The scene is misty and atmospheric, with dense evergreen forests in the foreground and snow-capped peaks in the distance.

*“Para lograr lo posible, hay  
que intentar lo imposible.”*

— Hermann Hesse



ACÀMICA