



Bài 1: Tổng quan Data Science

Phòng Lập Trình

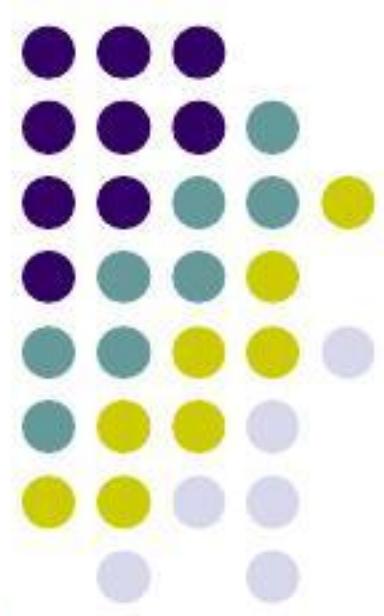




Nội dung

- **Giới thiệu Data Science**
- **Khác biệt giữa Data Science và Data Analytics**
- **Quy trình thực hiện dự án Data Science**
- **Python và các thư viện mở rộng**
- **Thiết lập môi trường làm việc**
- **Ngôn ngữ Markdown**



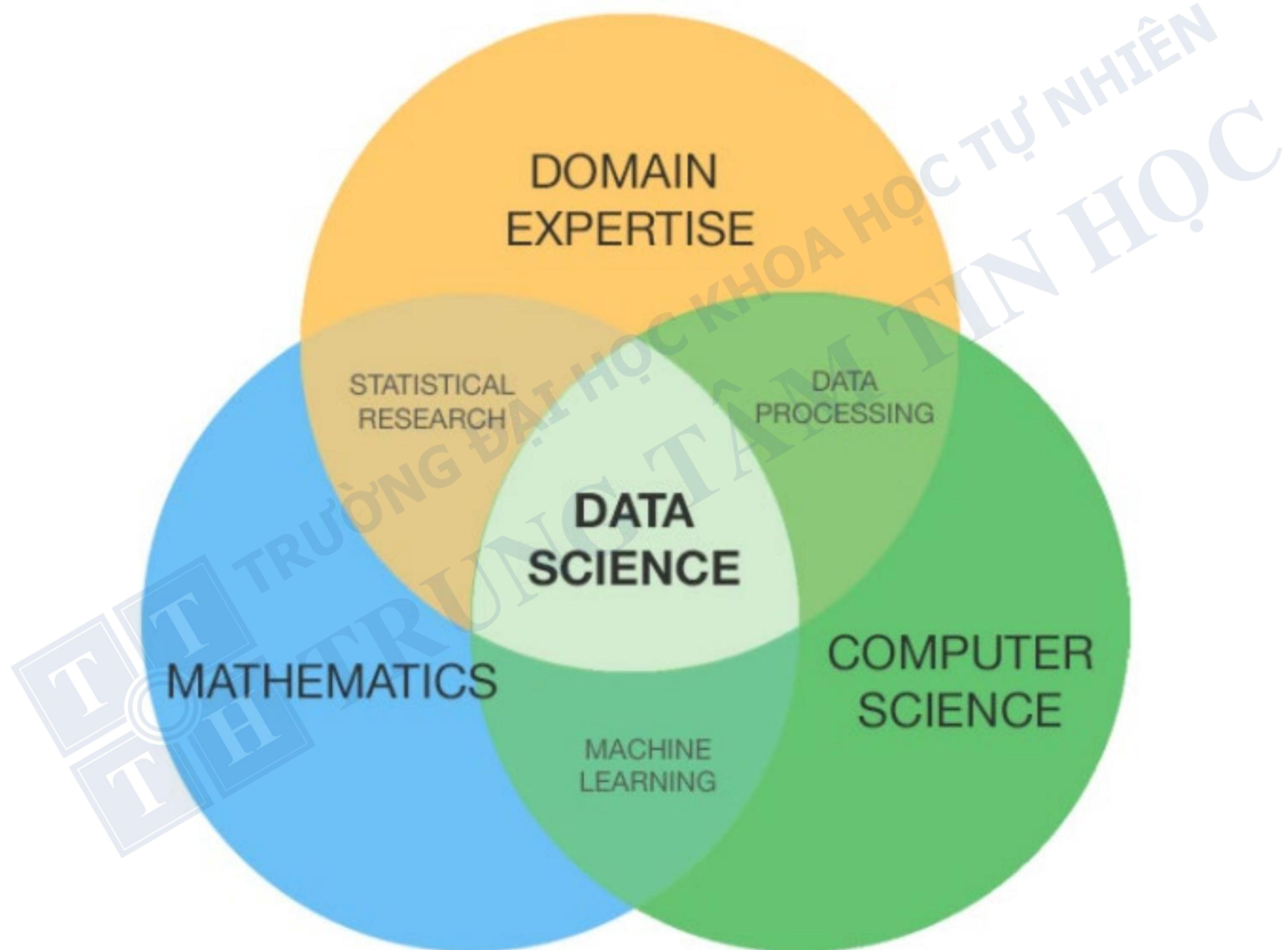
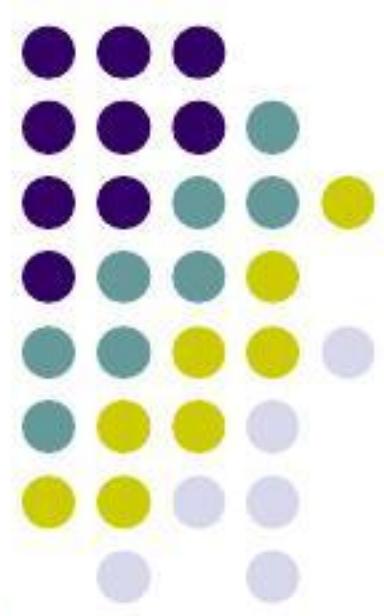


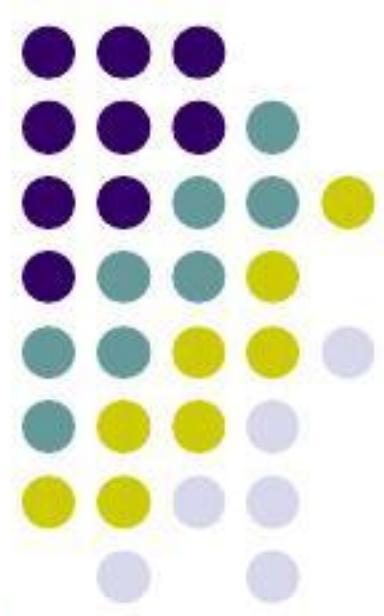
Giới thiệu Data Science

□ Data Science

- Data science (Khoa học dữ liệu) là lĩnh vực liên ngành sử dụng các phương pháp, quy trình, thuật toán và hệ thống khoa học để có được tri thức, thông tin giá trị từ dữ liệu.
- Data Science sử dụng các kỹ thuật từ toán học, thống kê, khoa học máy tính, và trí tuệ nhân tạo (AI).
- Data Science xử lý trên dữ liệu phức tạp và đa dạng, bao gồm cả dữ liệu có cấu trúc, bán cấu trúc và phi cấu trúc (unstructured data).
- Dữ liệu xử lý của Data Science có thể là Big Data, dữ liệu lớn, phức tạp, không thể xử lý hiệu quả bằng các phương pháp truyền thống.

Giới thiệu Data Science





Giới thiệu Data Science

□ Data Science – Xu hướng của tương lai

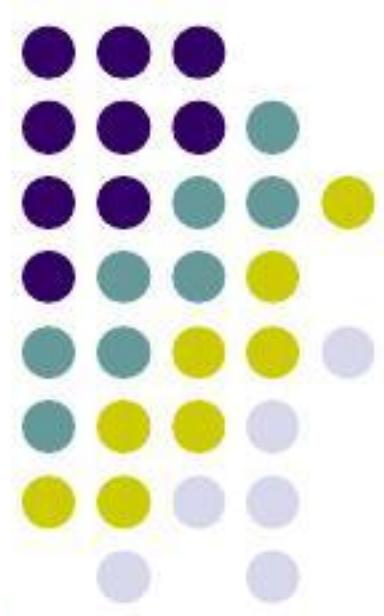
- Lượng dữ liệu được tạo ra đang tăng lên một cách chóng mặt.
- Dữ liệu không ngừng được thu thập thủ công, bán tự động và tự động.
- Dữ liệu từ nhiều nguồn ngày càng đa dạng và phức tạp.

Do đó:

- Cần có các phương pháp phân tích dữ liệu tiên tiến hơn, như máy học và trí tuệ nhân tạo.
- Kỹ thuật tích hợp và phân tích dữ liệu từ nhiều nguồn.
- Cần cơ chế bảo vệ an toàn dữ liệu.

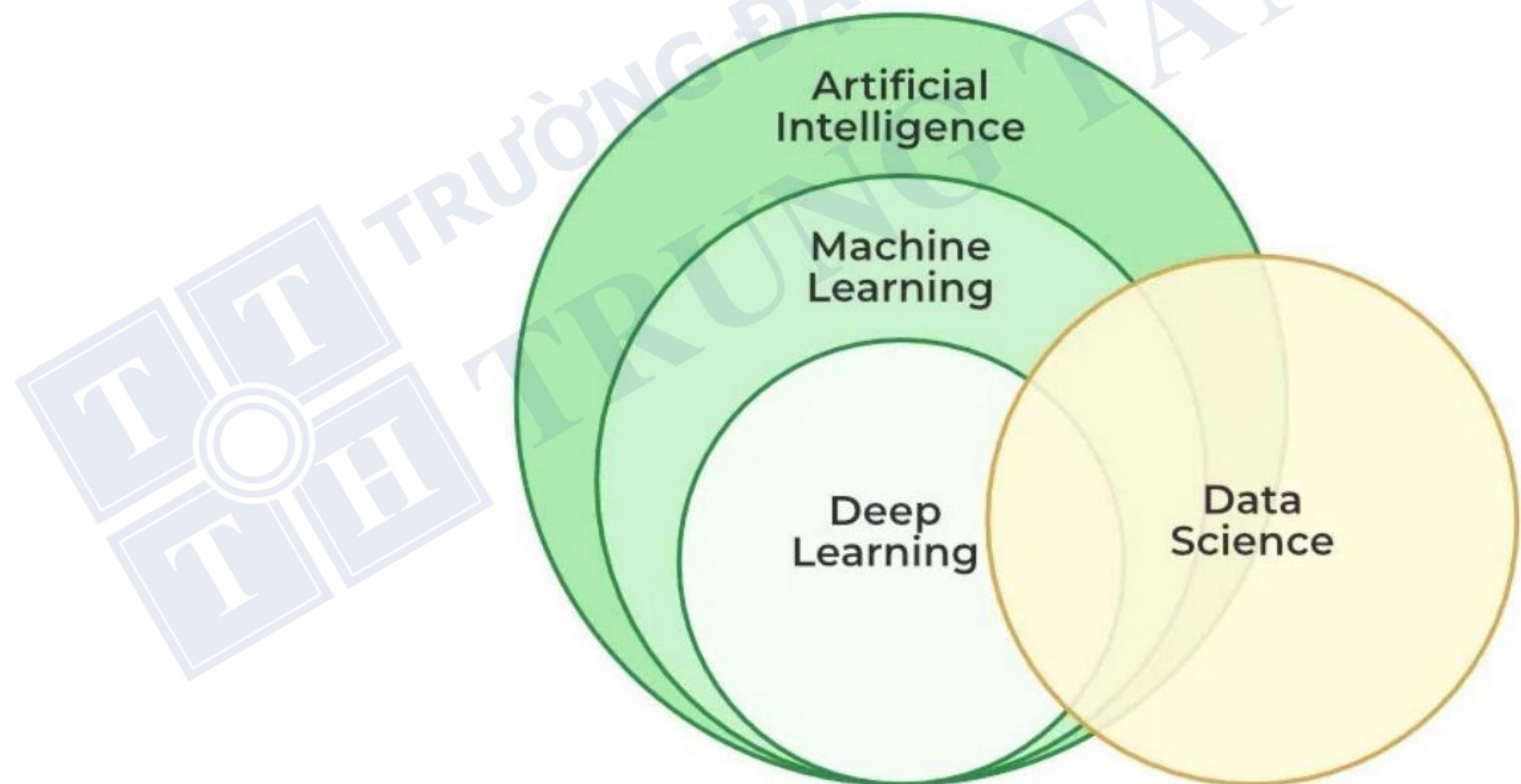
Các thuật toán học máy và AI sẽ ngày càng trở nên thông minh hơn, trở thành “trợ thủ” đắc lực trong xử lý và phân tích dữ liệu.

Giới thiệu Data Science



Với Data Science, dữ liệu không những cung cấp thông tin giá trị mà còn là nguồn “học liệu” quý giá. Dữ liệu được dùng để “học” và “dự đoán”.

- Dữ liệu – lợi thế cạnh tranh
- Dữ liệu – chủ động ứng phó tình huống
- Dữ liệu – giúp Doanh nghiệp, Tổ chức làm nên điều khác biệt



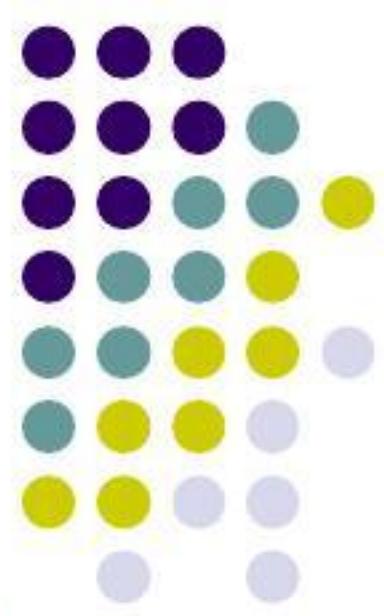


Giới thiệu Data Science

□ Các kỹ năng khoa học dữ liệu hiện đại

- Kỹ năng lập trình và xử lý dữ liệu với Python
- Trực quan hóa dữ liệu
- Toán học - Thống kê và ứng dụng
- Học máy (Machine Learning)
- Phân tích dữ liệu lớn (Scalable Big Data Analysis)
- Kỹ năng giao tiếp với dữ liệu

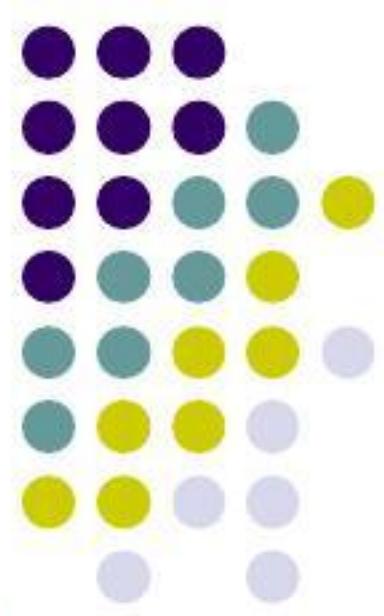




Nội dung

- Giới thiệu Data Science
- **Khác biệt giữa Data Science và Data Analytics**
- Quy trình thực hiện dự án Data Science
- Python và các thư viện mở rộng
- Thiết lập môi trường làm việc
- Ngôn ngữ Markdown

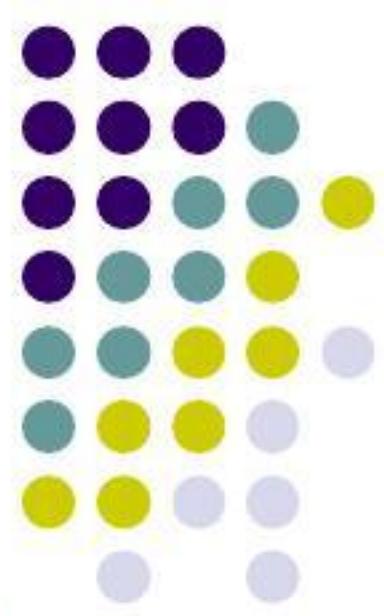




Khác biệt giữa Data Science và Data Analytics

□ Có 4 loại hình phân tích dữ liệu

- Phân tích Mô tả (Descriptive Analytics)
 - Hiểu dữ liệu quá khứ và hiện tại, phân tích **những gì đã xảy ra**.
 - Vd: Phân tích doanh thu, phân tích lượng truy cập website...
- Phân tích Chuẩn đoán (Diagnostic Analytics)
 - Tìm hiểu **nguyên nhân** của hiện tượng nào đó.
 - Vd: Tìm hiểu nguyên nhân doanh số giảm trong quý,...
- Phân tích Dự đoán (Predictive Analytics)
 - Sử dụng dữ liệu hiện tại và quá khứ để **dự đoán** tương lai
 - Vd: Dự đoán xu hướng doanh số trong tương lai,...
- Phân tích Tiên lượng (Prescriptive Analytics)
 - **Đưa ra các khuyến nghị, kịch bản hành động trong tương lai**.
 - Vd: Đưa ra các chiến lược kinh doanh dựa trên dự đoán nhu cầu.



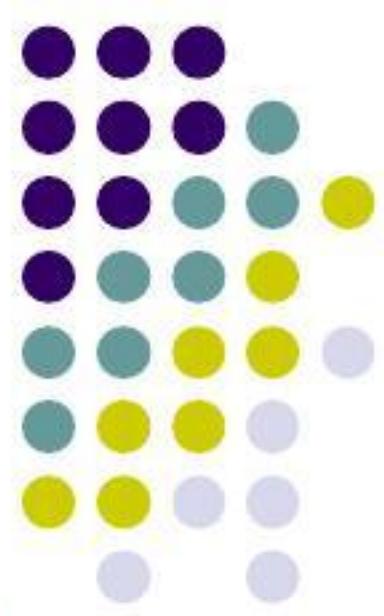
Khác biệt giữa Data Science và Data Analytics

□ Phân tích dữ liệu – Data Analytics

- Quá trình xử lý, làm sạch, biến đổi và mô hình hóa dữ liệu để tìm ra thông tin hữu ích giúp hỗ trợ việc ra quyết định.
- Phân tích dữ liệu ít tập trung vào xây dựng các mô hình máy học phức tạp.

□ Khoa học dữ liệu – Data Science

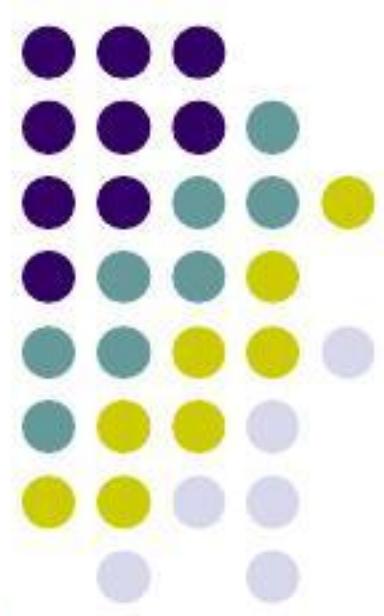
- Khoa học dữ liệu là một lĩnh vực rộng lớn hơn, kết hợp phân tích dữ liệu với lập trình, máy học và trí tuệ nhân tạo.
- Thường xử lý dữ liệu phức tạp hơn với quy mô lớn hơn.



Khác biệt giữa Data Science và Data Analytics

Data Science	Data Analytics
Phân tích dữ liệu hiện có và xây dựng các mô hình máy học dự đoán tương lai.	Phân tích dữ liệu hiện tại và quá khứ để rút ra thông tin giá trị
Xử lý dữ liệu phức tạp, từ nhiều nguồn và dữ liệu lớn (Big Data)	Xử lý với dữ liệu có cấu trúc, ít phức tạp.
Tập trung nhiều hơn vào ứng dụng và phát triển các thuật toán máy học	Chủ yếu sử dụng kỹ thuật thống kê và trực quan hóa dữ liệu
Áp dụng kiến thức từ nhiều ngành khác nhau và lĩnh vực cụ thể của dữ liệu	Các tình huống phân tích cụ thể

Trong một thế giới dữ liệu đang không ngừng thay đổi, với kỹ năng Khoa học dữ liệu bạn sẽ luôn thích ứng và đổi mới.

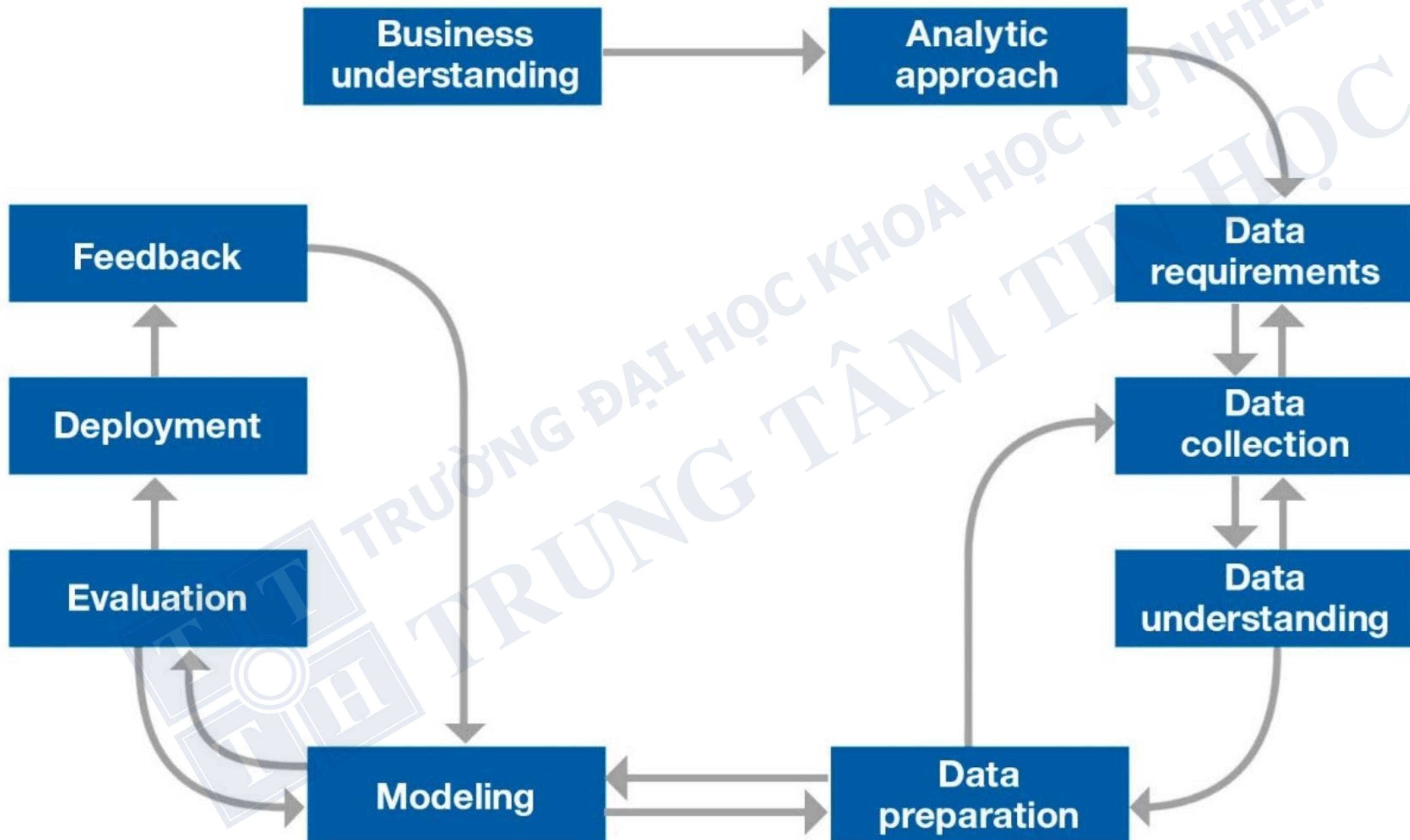


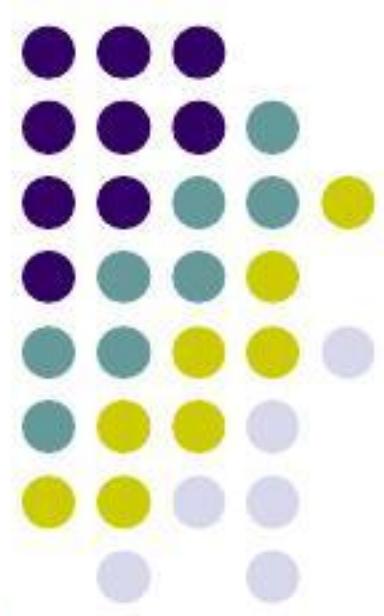
Nội dung

- Giới thiệu Data Science
- **Khác biệt giữa Data Science và Data Analytics**
- Quy trình thực hiện dự án Data Science
- Python và các thư viện mở rộng
- Thiết lập môi trường làm việc
- Ngôn ngữ Markdown



Quy trình thực hiện dự án Data Science





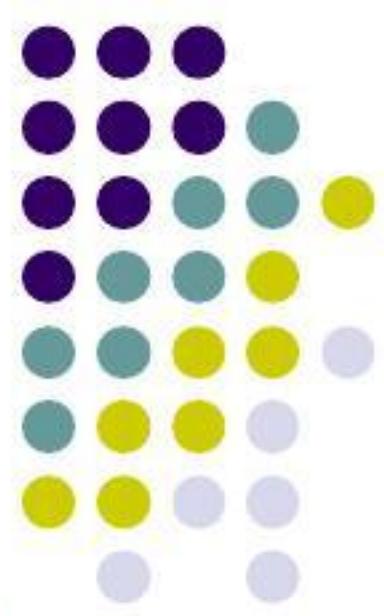
Quy trình thực hiện dự án Data Science

□ Business Understanding – Tìm hiểu vấn đề

- Vấn đề quan trọng cần giải quyết
- Các KPI (Key Performance Indicator) đo lường để đánh giá

□ Analytics Approach – Hướng tiếp cận phân tích

- Chiến lược phân tích phù hợp để giải quyết vấn đề
- Quyết định phương pháp phân tích nào sẽ được sử dụng
 - Phân tích mô tả (Descriptive Analytics)
 - Phân tích dự báo (Predictive Analytics)
 - ...



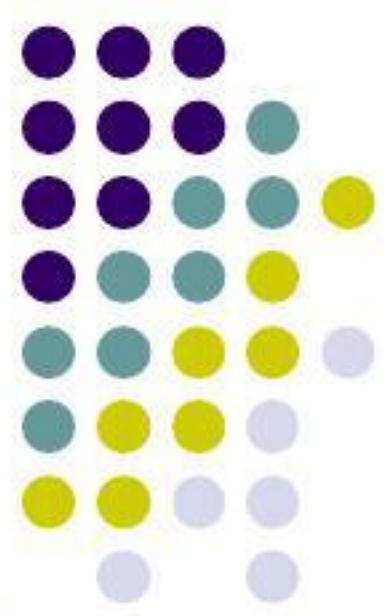
Quy trình thực hiện dự án Data Science

□ Data Requirements – Yêu cầu dữ liệu

- Dữ liệu nào cần?
- Nguồn dữ liệu? Định dạng? Mức độ chi tiết? Độ trễ dữ liệu...

□ Data Collection – Thu thập dữ liệu

- Thu thập dữ liệu từ nhiều nguồn
- Các yêu cầu về dữ liệu có thể được sửa đổi, có thể cần nhiều hơn hay ít dữ liệu hơn.



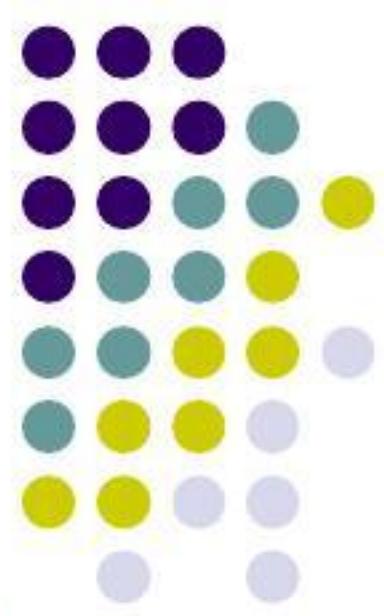
Quy trình thực hiện dự án Data Science

□ Data Understanding – Hiểu dữ liệu

- Thực hiện các phân tích sơ bộ để nhận biết các mô hình, vấn đề tiềm ẩn hoặc điểm dữ liệu nổi bật.
- Dữ liệu thu thập có phù hợp cho vấn đề cần giải quyết không?

□ Data Preparation – Chuẩn bị dữ liệu

- Giai đoạn chiếm khá nhiều thời gian và công sức.
- Làm sạch dữ liệu, chọn lọc các biến đặc trưng, xử lý chuyển đổi dữ liệu..
- Kết quả là tập dữ liệu sẵn sàng cho việc xây dựng mô hình.
- Data – Preprocessing (tiền xử lý dữ liệu).



Quy trình thực hiện dự án Data Science

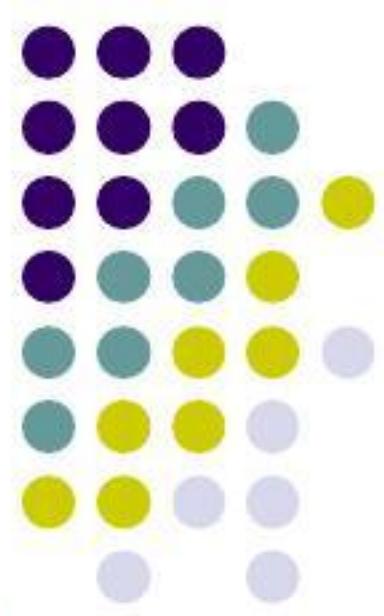
□ Data Modelling – Mô hình hóa dữ liệu

- Tạo các mô hình thống kê hoặc máy học trên dữ liệu đã được xử lý.
- Xây dựng mô hình phân tích mô tả, phân tích chuẩn đoán hoặc phân tích dự báo, dự đoán.

□ Evaluation – Deployment - Feedback

- Đánh giá hiệu quả của mô hình.
- Triển khai mô hình vào thực tế.
- Ghi nhận phản hồi và cải tiến.

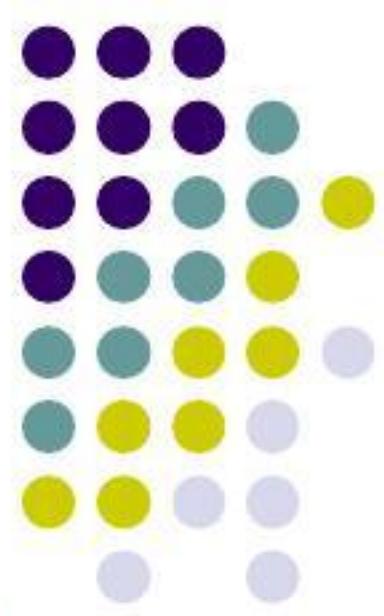
Quy trình thực hiện dự án Data Science



□ Ứng dụng bài toán Titanic

- Khả năng sống sót của hành khách phụ thuộc vào những yếu tố như độ tuổi, giới tính, giá vé, ... như thế nào ?

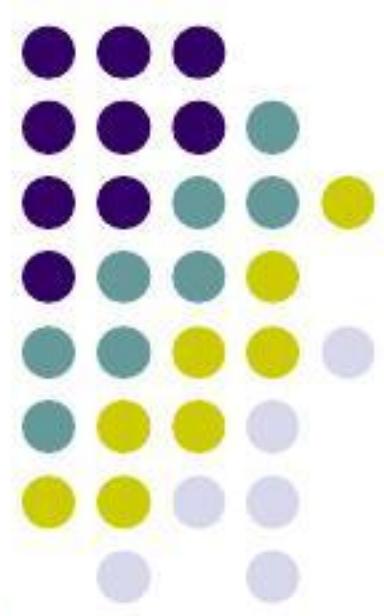




Nội dung

- Giới thiệu Data Science
- Khác biệt giữa Data Science và Data Analytics
- Quy trình thực hiện dự án Data Science
- Python và các thư viện mở rộng
- Thiết lập môi trường làm việc
- Ngôn ngữ Markdown



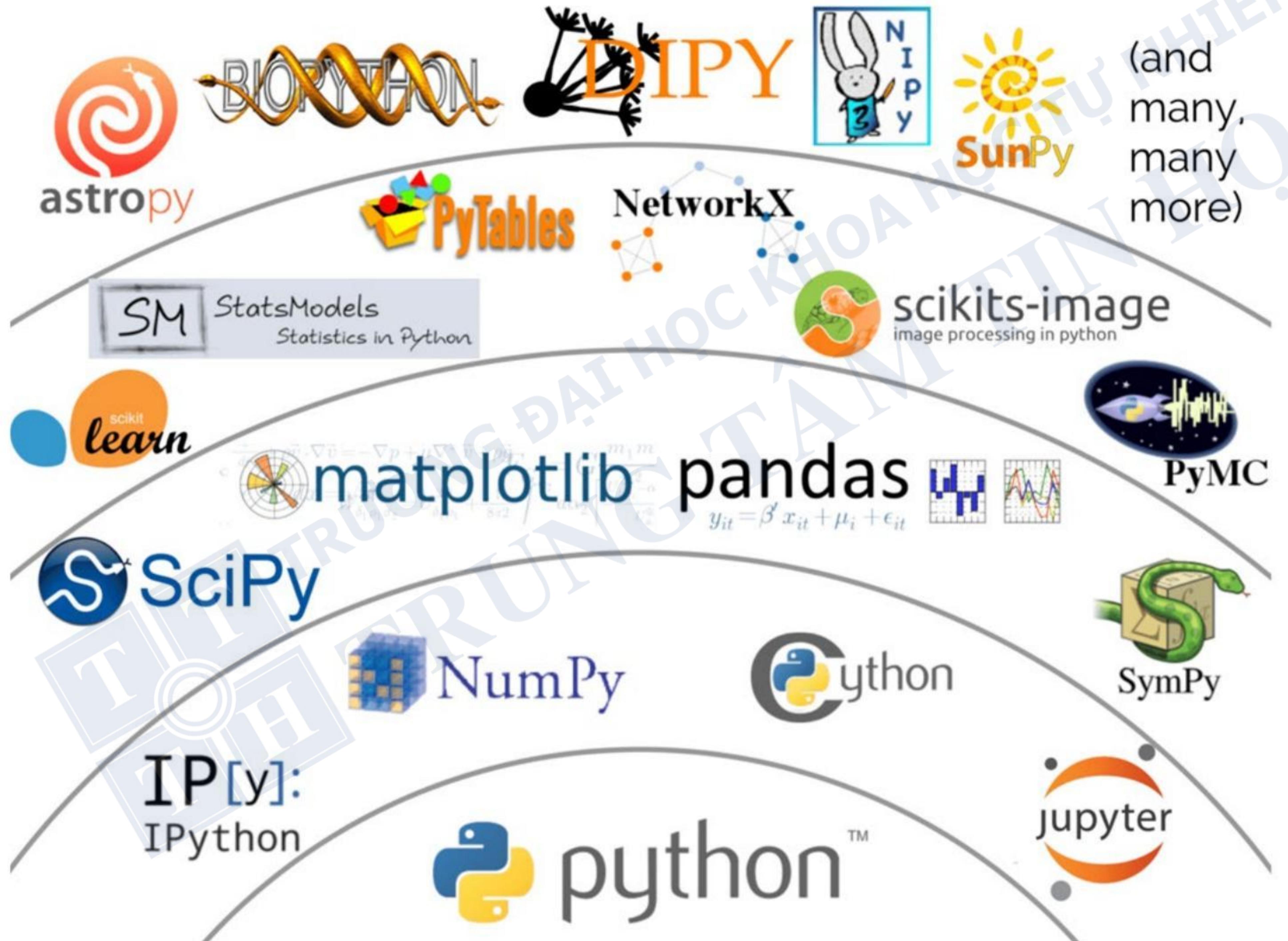
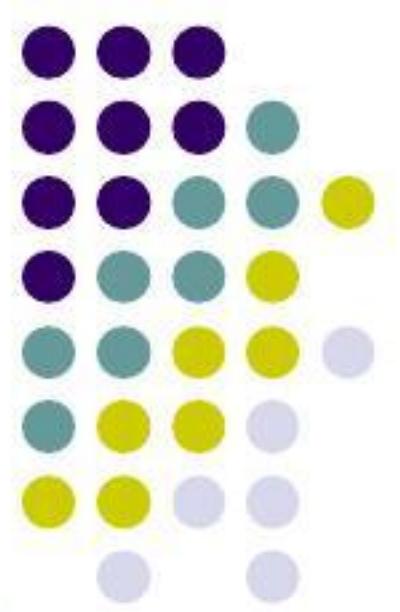


Python và các thư viện mở rộng

□ Python - ngôn ngữ lập trình chính trong Khoa học Dữ liệu

- Dễ học – dễ sử dụng.
- Thư viện phong phú và luôn phát triển.
- Cộng đồng người dùng lớn và sẵn sàng hỗ trợ.
- Dễ dàng tích hợp và mở rộng.
- Nhiều thư viện miễn phí, mạnh mẽ cho phân tích dữ liệu, trực quan hóa dữ liệu, học máy và AI.
- Được các dịch vụ Cloud Services (AWS, Azure, Google Cloud Platform...) hỗ trợ xử lý dữ liệu quy mô lớn và tính toán phân tán.
- Được tích hợp vào các trợ lý AI đắc lực hiện nay: ChatGPT, GitHub Copilot, DeepCode....

Python – Thư viện hỗ trợ phong phú

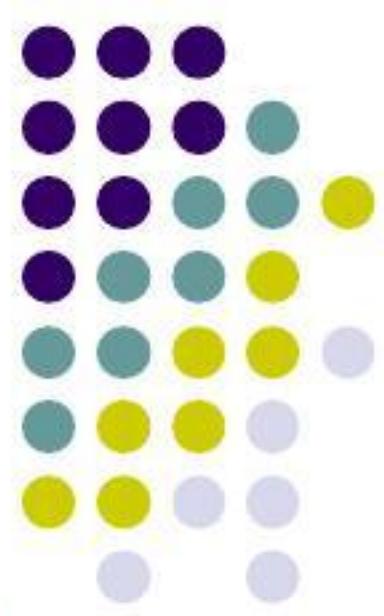




Nội dung

- Giới thiệu Data Science
- Khác biệt giữa Data Science và Data Analytics
- Quy trình thực hiện dự án Data Science
- Python và các thư viện mở rộng
- Thiết lập môi trường làm việc
- Ngôn ngữ Markdown





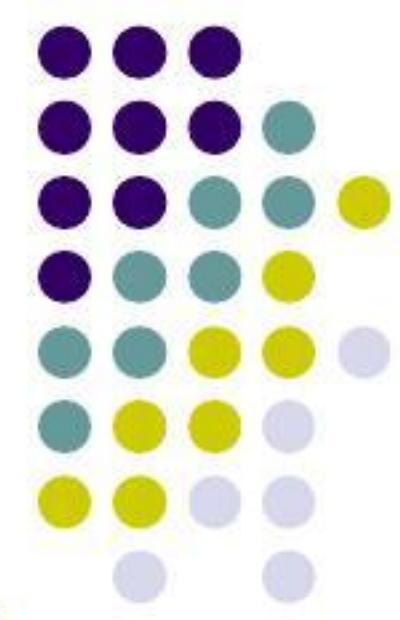
Thiết lập môi trường làm việc

Jupyter <https://jupyter.org/>

- Công cụ rất hữu ích lĩnh vực khoa học dữ liệu và máy học.
- Cho phép viết mã lệnh, thực thi lệnh, hiển thị kết quả, chèn thêm văn bản chú thích, hình ảnh,... một cách trực quan trên cùng tài liệu, tập tin lệnh.
- Dễ dàng chia sẻ tài liệu với người khác.
- Mặc định Jupyter hỗ trợ Python và R, để có thể thực thi các ngôn ngữ lập trình khác người dùng sẽ cài thêm các Kernel tương ứng.
- Cài đặt

```
pip install jupyter
```
- Jupyter Notebook và JupyterLab được cài đặt tự động sau khi cài đặt Jupyter thành công.
- Các notebook Jupyter được lưu dưới dạng tập tin **.ipynb**

Thiết lập môi trường làm việc



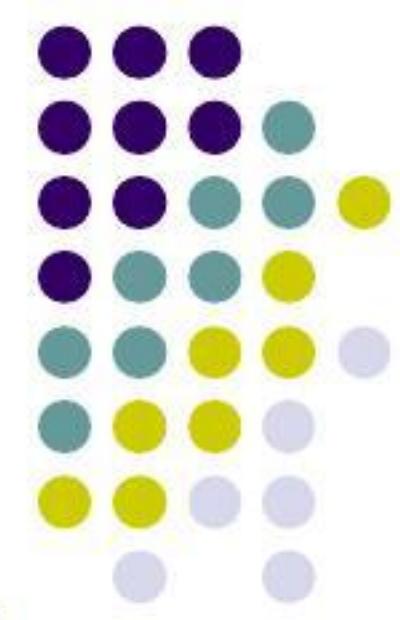
□ Giao diện người dùng của Jupyter

- Jupyter Notebook



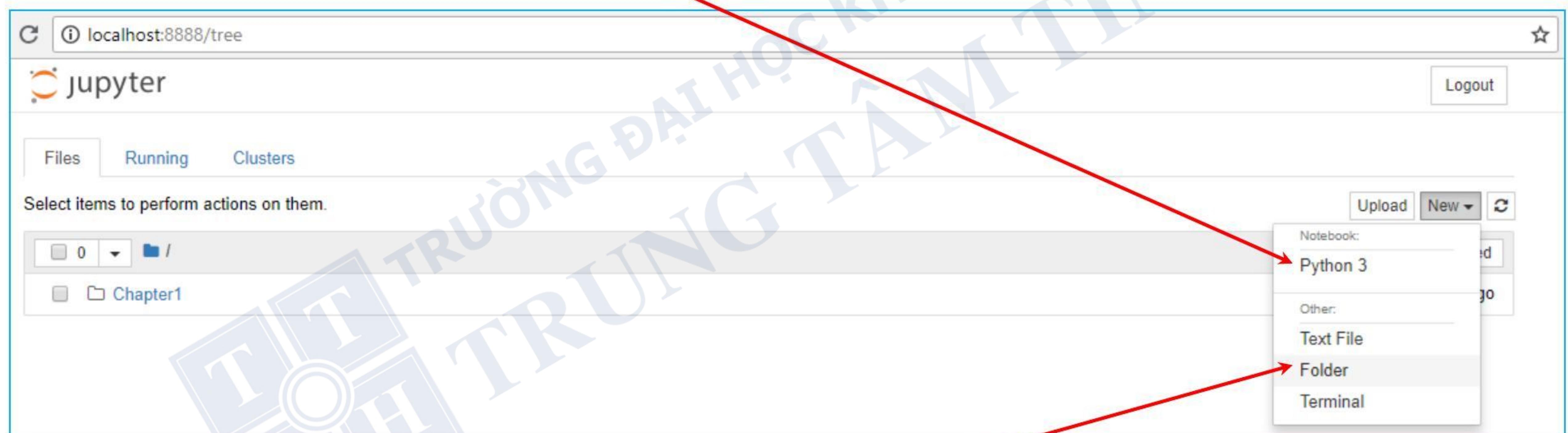
- Giao diện người dùng ban đầu, “cỗ điển” của Jupyter
- Cung cấp các tính năng cơ bản như tạo, chỉnh sửa, chạy các lệnh trong notebook.
- Mỗi notebook được mở trong một tab riêng biệt của trình duyệt
- Khởi động: mở command prompt hoặc tại đường dẫn thư mục làm việc, gõ `jupyter notebook`

Thiết lập môi trường làm việc



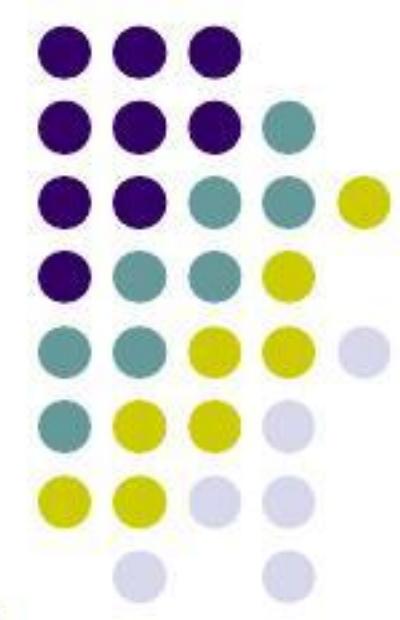
- Giao diện người dùng của Jupyter
 - Jupyter Notebook

Tạo tập tin làm việc (.ipynb)



Tạo folder

Thiết lập môi trường làm việc

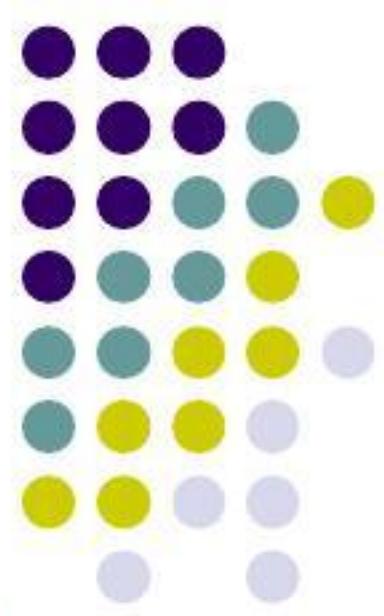


□ Giao diện người dùng của Jupyter

- JupyterLab



- Có đầy đủ tất cả tính năng của Jupyter Notebook.
- Cho phép mở nhiều notebook, tập tin khác cùng một lúc trong cùng một cửa sổ. Cho phép sắp xếp các tab và cửa sổ con một cách linh hoạt.
- Dễ dàng xem trước tập tin dữ liệu .csv dưới dạng bảng tính.
- Có thể thực thi các lệnh dạng Console.
- Dễ dàng sắp xếp, di chuyển các cell trong notebook.
- Khởi động: mở command prompt hoặc tại đường dẫn thư mục làm việc, gõ `jupyter-lab`.



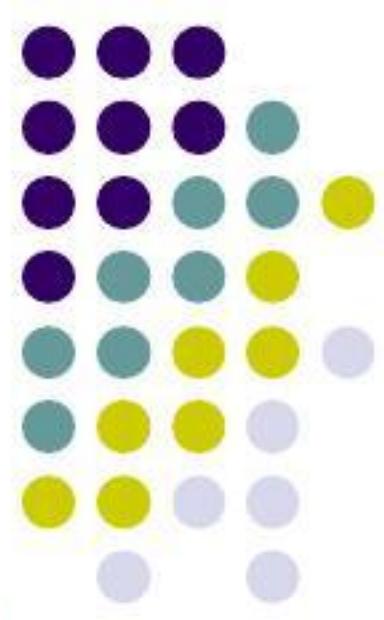
Thiết lập môi trường làm việc

Google Colaboratory

- Hay Google Colab, là một dịch vụ miễn phí của Google dựa trên nền tảng Jupyter.
- Cho phép viết và thực thi mã lệnh Python trực tuyến mà không cần cài đặt.
- Cung cấp quyền sử dụng tài nguyên máy chủ của Google (CPU, GPU) giúp xử lý các tác vụ tính toán phức tạp một cách nhanh chóng.
- Tất cả notebook trên Colab được lưu trữ trên Google Drive, giúp việc chia sẻ và cộng tác trở nên dễ dàng.
- Colab rất hữu ích khi huấn luyện mô hình machine learning, deep learning.



Thiết lập môi trường làm việc



Môi trường thực hành



Array



NumPy

Series và DataFrame

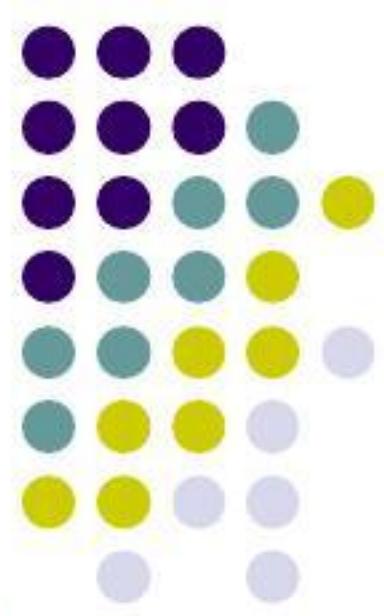


Trực quan hóa dữ liệu



seaborn





Thiết lập môi trường làm việc

□ Cài đặt và quản lý các thư viện

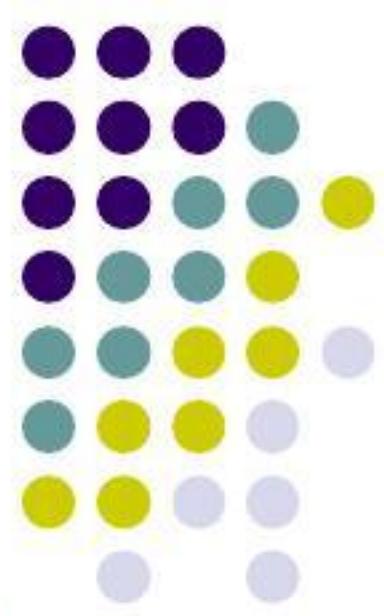
- Cài đặt jupyter: pip install jupyter
- Cài đặt các thư viện: numpy, pandas, matplotlib, seaborn, ...
 - pip install numpy
 - pip install pandas
 - pip install matplotlib
 - pip install seaborn





Nội dung

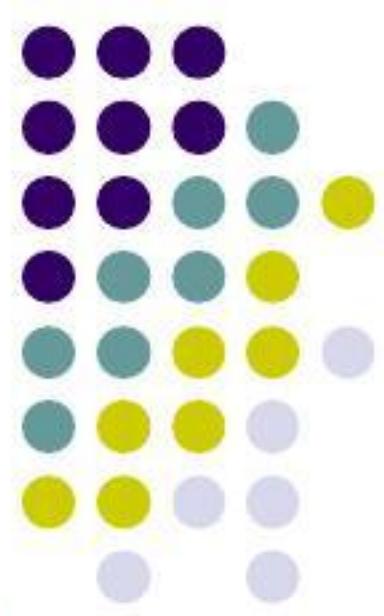
- Giới thiệu Data Science
- Khác biệt giữa Data Science và Data Analytics
- Quy trình thực hiện dự án Data Science
- Python và các thư viện mở rộng
- Thiết lập môi trường làm việc
- Ngôn ngữ Markdown



Ngôn ngữ Markdown

□ Markdown

- Là ngôn ngữ đánh dấu siêu nhẹ với định dạng văn bản plain text.
- Giúp chuyển đổi nội dung sang định dạng HTML một cách đơn giản và trực quan.
- Cú pháp đơn giản và gọn gàng, giúp người dùng tập trung vào nội dung chứ không phải định dạng.
- Thường dùng để định dạng file readme hoặc viết nội dung trong các kênh chia sẻ online như GitHub, Bitbucket, tập tin Jupyter Notebook và nhiều trình soạn thảo văn bản khác.
- Không hỗ trợ một số tính năng phức tạp như bảng màu, kích thước font, hoặc các yếu tố trang trí phức tạp khác mà HTML hỗ trợ.



Ngôn ngữ Markdown

☐ Một số cú pháp thông dụng

Type

Or

... to Get

Italic

Italic

Italic

Bold

 Bold

Bold

Heading 1

Heading 1
=====

Heading 1

Heading 2

Heading 2

Heading 2

[Link](<http://a.com>)

[Link][1]
:
[1]: <http://b.org>

[Link](#)

![Image](<http://url/a.png>)

![Image][1]
:
[1]: <http://url/b.jpg>



Ngôn ngữ Markdown



□ Một số cú pháp thông dụng

> Blockquote

- * List
- * List
- * List

- List
- List
- List

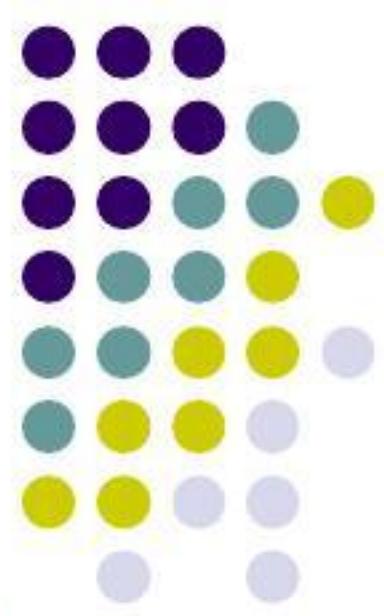
Blockquote

- List
- List
- List

1. One
2. Two
3. Three

- 1) One
- 2) Two
- 3) Three

1. One
2. Two
3. Three



Ngôn ngữ Markdown

□ Một số cú pháp thông dụng

Horizontal Rule

‘Inline code’ with backticks

...

```
# code block
print '3 backticks or'
print 'indent 4 spaces'
...
```

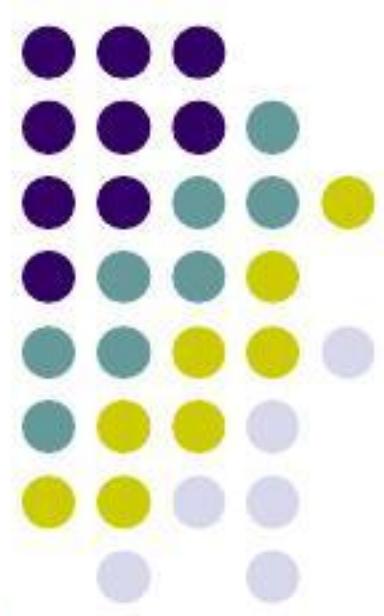
Horizontal Rule

```
....# code block
....print '3 backticks or'
....print 'indent 4 spaces'
```

Horizontal Rule

Inline code with backticks

```
# code block
print '3 backticks or'
print 'indent 4 spaces'
```



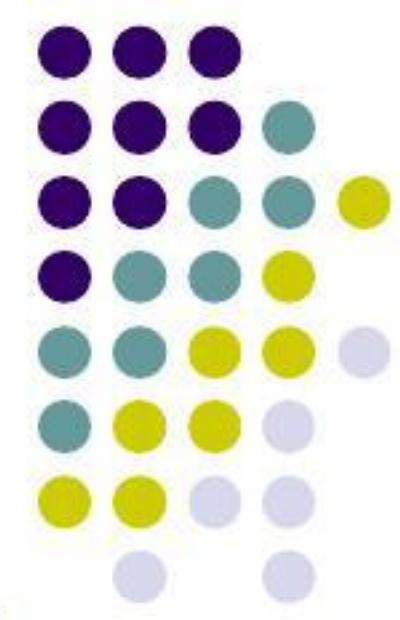
Ngôn ngữ Markdown

☐ Một số cú pháp thông dụng

- Table: sử dụng | để phân cách các cột

Python Operator	Description	Python Operator	Description
<code>---</code>	<code>---</code>	<code>+</code>	addition
<code>+</code>	addition	<code>-</code>	subtraction
<code>-</code>	subtraction	<code>*</code>	multiplication
<code>*</code>	multiplication	<code>/</code>	division
<code>/</code>	division	<code>**</code>	power
<code>**</code>	power		

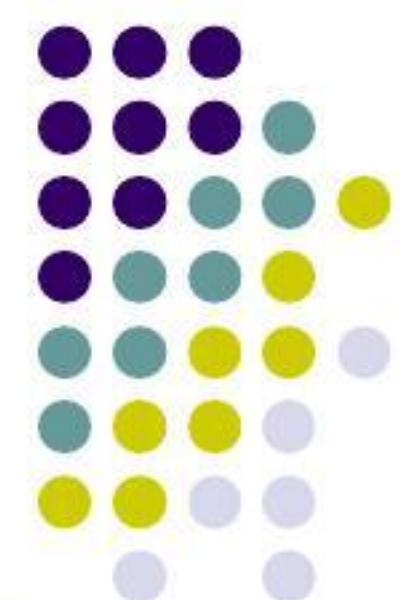
Ngôn ngữ Markdown



□ Ký tự Hy Lạp

Cú pháp: \$ Script \$
Ví dụ: \$\\alpha\$

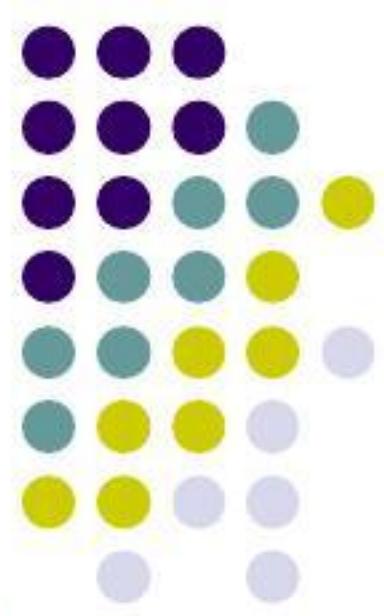
Symbol	Script
α	\$\\alpha\$
A	\$A\$
β	\$\\beta\$
B	\$B\$
γ	\$\\gamma\$
Γ	\$\\Gamma\$
π	\$\\pi\$
Π	\$\\Pi\$
ϕ	\$\\phi\$
Φ	\$\\Phi\$
φ	\$\\varphi\$
θ	\$\\theta\$



Markdown Text

□ Operator

Symbol	Script
cos	\cos
sin	\sin
lim	\lim
exp	\exp
→	\rightarrow
∞	\infty
≡	\equiv
mod	\bmod
×	\times



Ngôn ngữ Markdown

□ Chỉ số trên, dưới (power & indices)

- Dùng $^$ trước chỉ số trên và $_$ trước chỉ số dưới

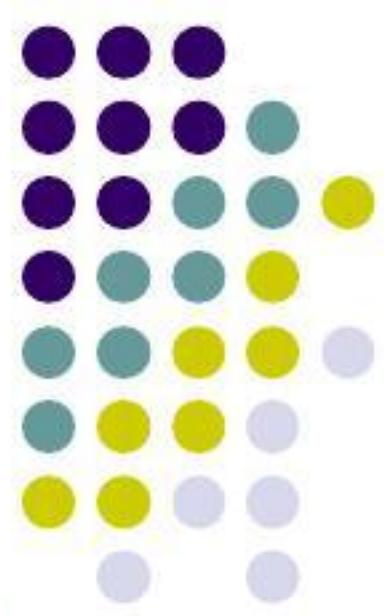
Symbol	Script
k_{n+1}	$k_{\{n+1\}}$
n^2	$n^{\wedge}2$
k_n^2	$k_n^{\wedge}2$



Ngôn ngữ Markdown

□ Phân số và nhị thức (fractions & binomials)

Symbol	Script
$\frac{n!}{k!(n-k)!}$	<code>\frac{n!}{k!(n-k)!}</code>
$\binom{n}{k}$	<code>\binom{n}{k}</code>
$\frac{\frac{x}{1}}{x-y}$	<code>\frac{\frac{x}{1}}{x-y}</code>
$\frac{3}{7}$	<code>\frac{3}{7}</code>



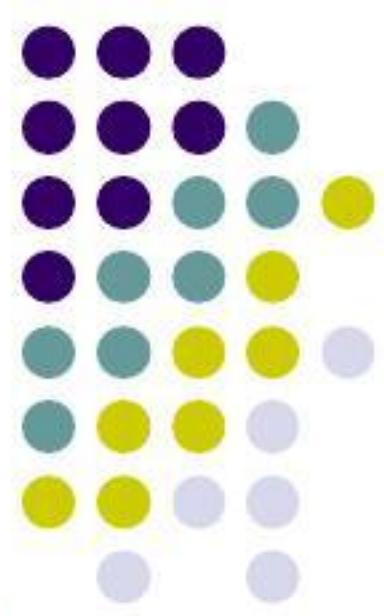
Ngôn ngữ Markdown

□ Căn (root)

Symbol	Script
\sqrt{k}	<code>\sqrt{k}</code>
$\sqrt[n]{k}$	<code>\sqrt[n]{k}</code>

□ Tổng và tích phân (sums & integrals)

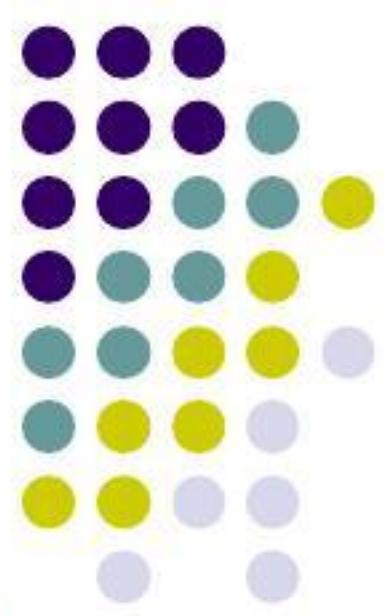
Symbol	Script
$\sum_{i=1}^{10} t_i$	<code>\sum_{i=1}^{10} t_i</code>
Σ	<code>\sum</code>
\int	<code>\int</code>
\oint	<code>\oint</code>
\iint	<code>\iint</code>
\int_a^b	<code>\int_a^b</code>



Ngôn ngữ Markdown

□ Các ký hiệu khác

Symbol	Script
(a)	(a)
$[a]$	$[a]$
a	$\{a\}$
$\langle f \rangle$	$\backslash\langle\!f\!\rangle$
$\lfloor f \rfloor$	$\lfloor f \rfloor$
$\lceil f \rceil$	$\lceil f \rceil$
$\lceil f \rceil$	$\lceil f \rceil$



Ngôn ngữ Markdown

□ Cú pháp tạo công thức

- \$ công thức \$

Trung bình của mẫu (mean)

$$\bar{x} = \frac{\sum_{i=1}^n x_i}{n}$$

Phương sai của mẫu (variance)

$$s^2 = \frac{\sum_{i=1}^n (x_i - \bar{x})^2}{n-1}$$

Độ lệch chuẩn (standard deviation)

$$s = \sqrt{s^2}$$

Trung bình của mẫu (mean)

$$\# \$ \bar{x} = \frac{\sum_{i=1}^n x_i}{n} \$$$

Phương sai của mẫu (variance)

$$\# \$ s^2 = \frac{\sum_{i=1}^n (x_i - \bar{x})^2}{n-1} \$$$

Độ lệch chuẩn (standard deviation)

$$\# \$ s = \sqrt{s^2} \$$$

