

What really is

Machine Learning

And how can anyone
get involved in it?

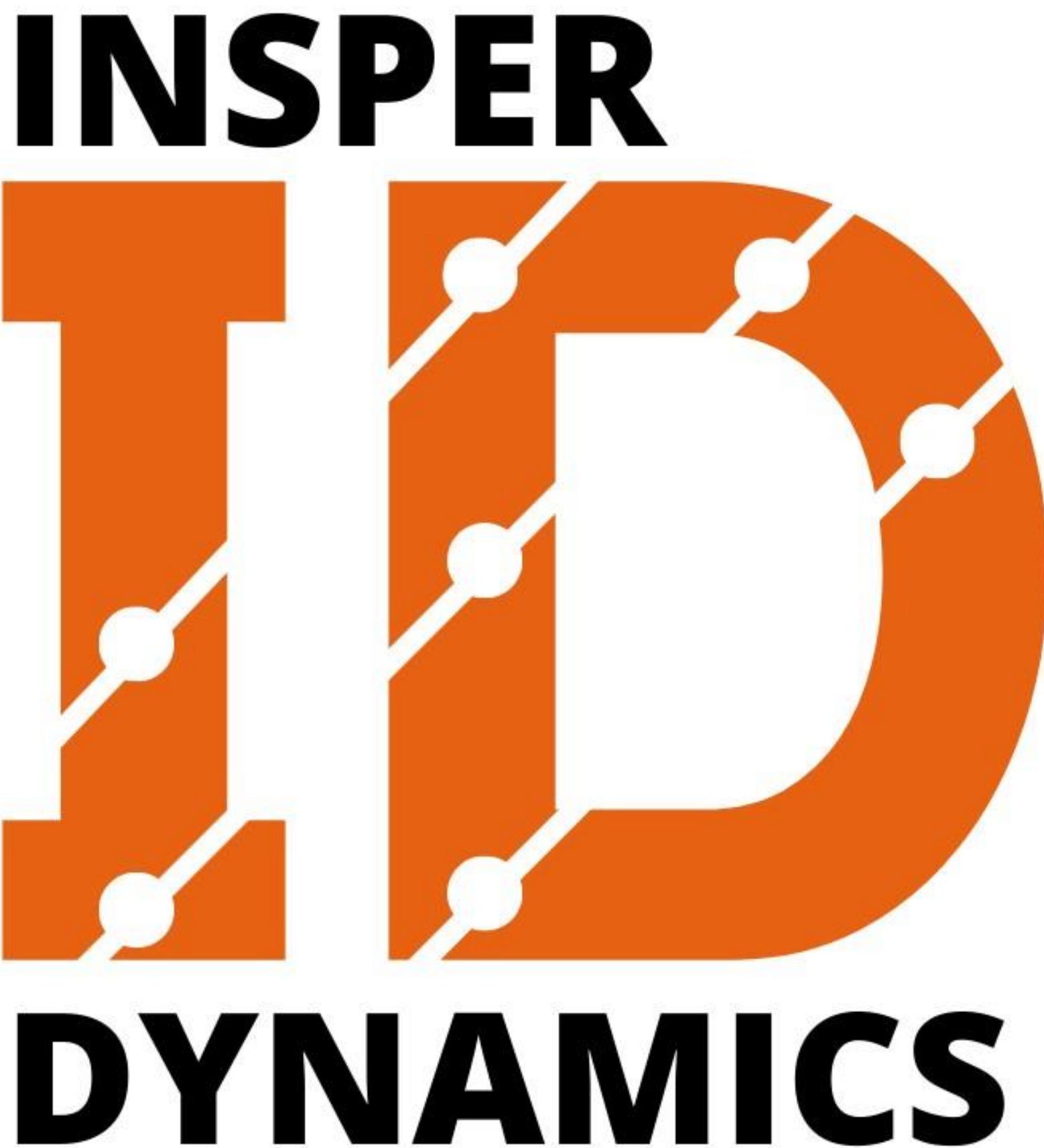


le wagon

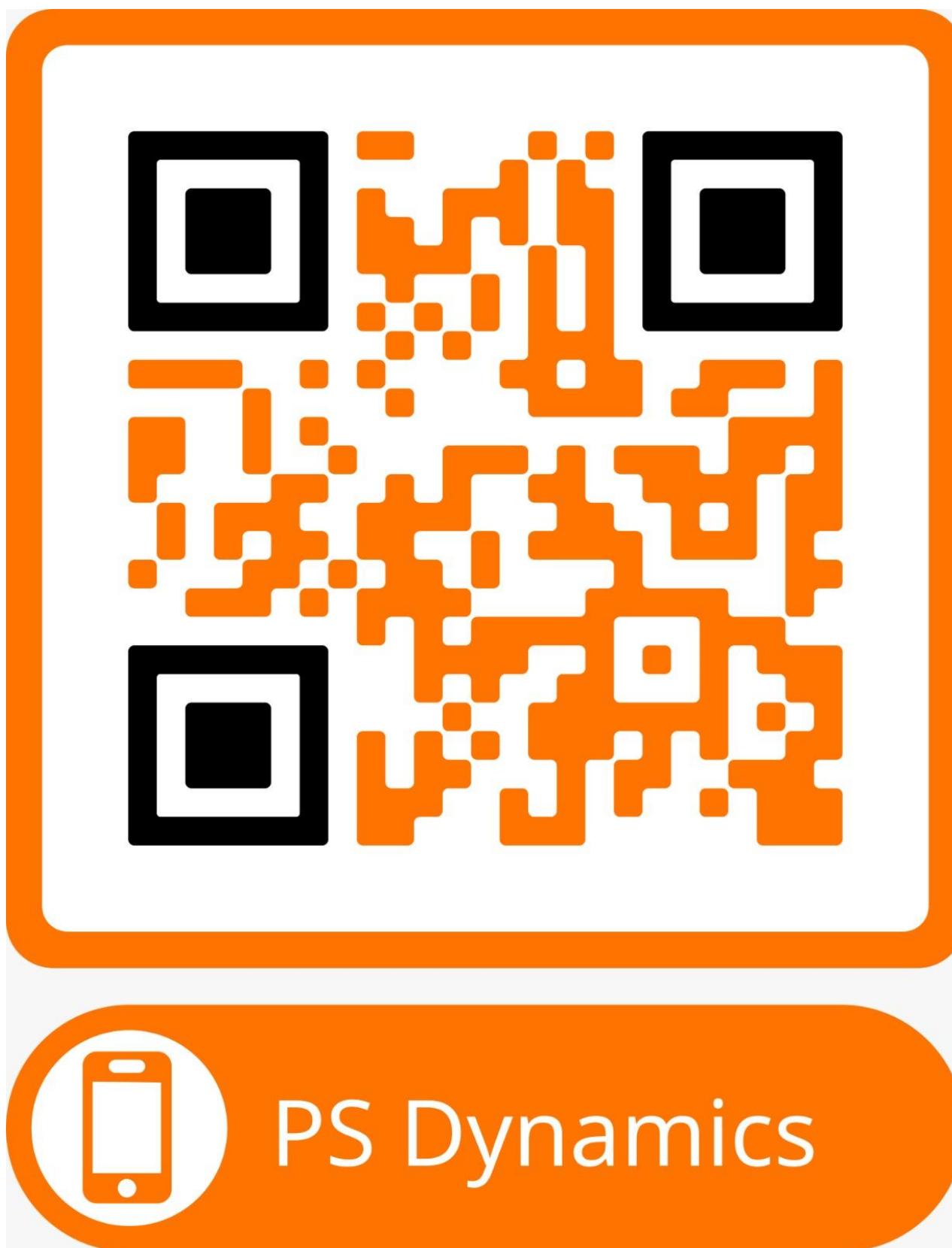


Nossa Entidade

- Machine Learning
 - Kaggle
 - ChatBOT
- Computação Embarcada
 - Proto E-enable
 - Robocup
- Web Development
 - INSPER App
 - Consulting



Processo Seletivo:



Since 2014

Le Wagon

Has been providing candidates with the skillset necessary to future-proof their career.

A teaching of excellence

Most voted

Voted bootcamp in the world



Official Partners

International campuses

16,000+

People educated at Le Wagon

43

Cities around the world

Focused on practice & product

2500+

Products build

100+

Startups launched



le wagon

Agenda

- ✓ What is Machine Learning
- ✓ What is **NOT** Machine Learning
- ✓ Who are the people building ML
- ✓ Let's code our own models!
- ✓ What we didn't cover



le wagon



- ✓ What is Machine Learning
- ✓ What is NOT Machine Learning
- ✓ Who are the people building ML
- ✓ Let's code our own models!
- ✓ What we didn't cover



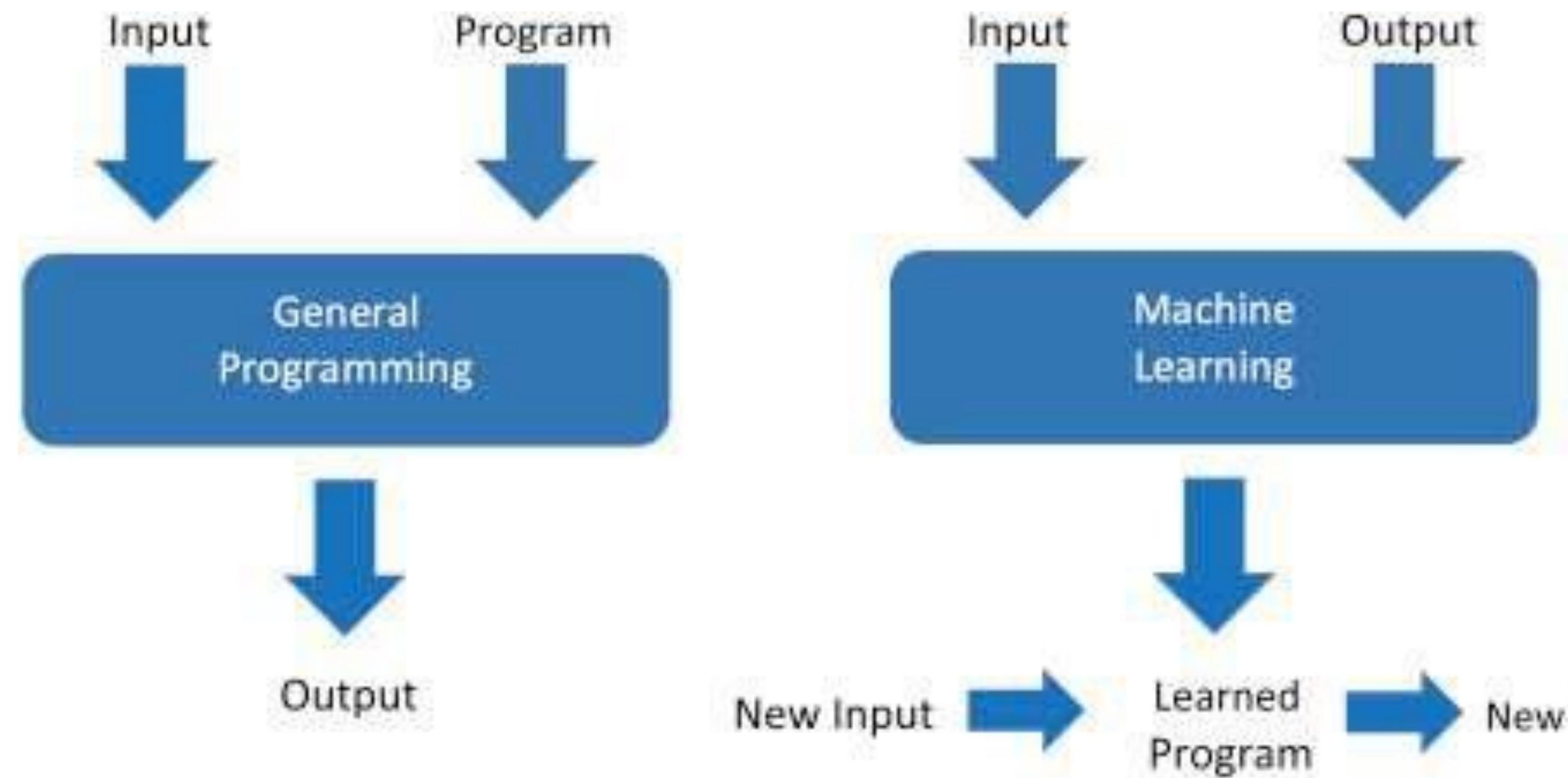
le wagon



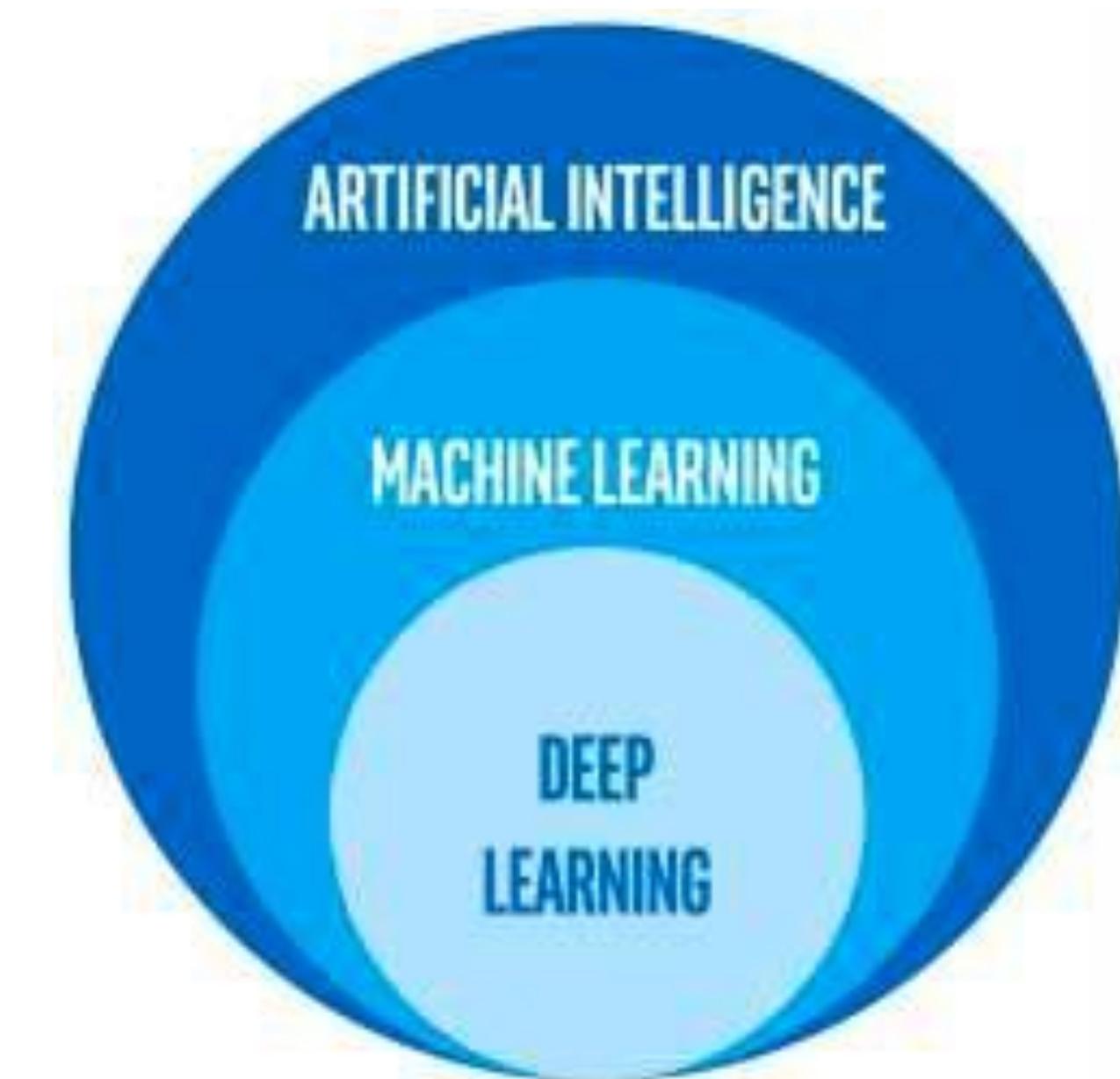
“[a] field of study that gives computers the ability to learn without being explicitly programmed”

Arthur Samuel
(1959)





What about terms like AI or Deep Learning?



Source: intel.com

How Machine Learning works ?



Take basics Observations

My Age / Height ratio ↴

Input

Output

Age	Height (cm)
5	95
6	103
7	110
8	114
9	121
10	126
11	130
12	135



Take basics Observations

My Age / Height ratio ↴

Feature
Input

Age	Height (cm)
5	95
6	103
7	110
8	114
9	121
10	126
11	130
12	135

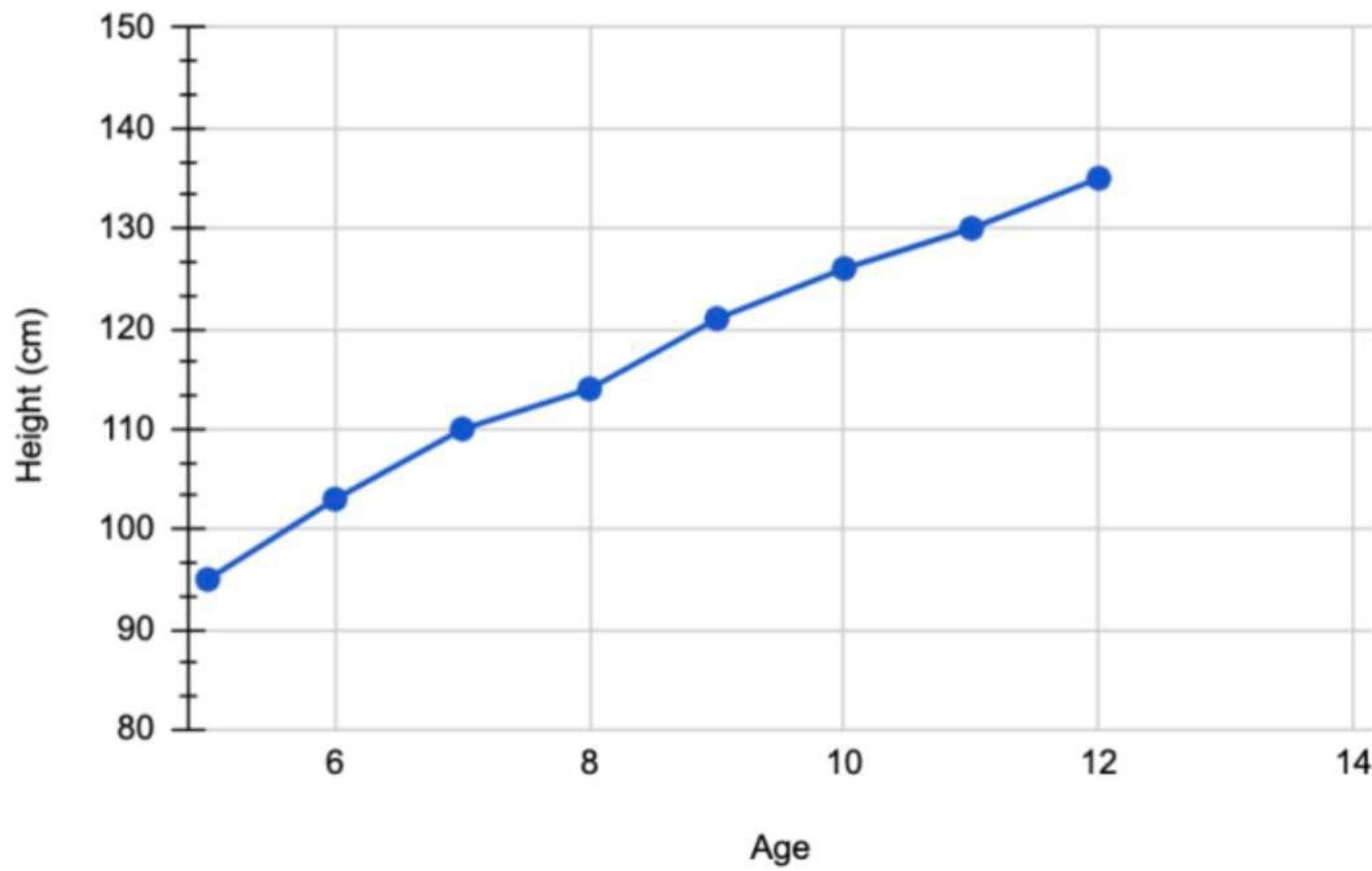
Target
Output



Infer a **Math** function

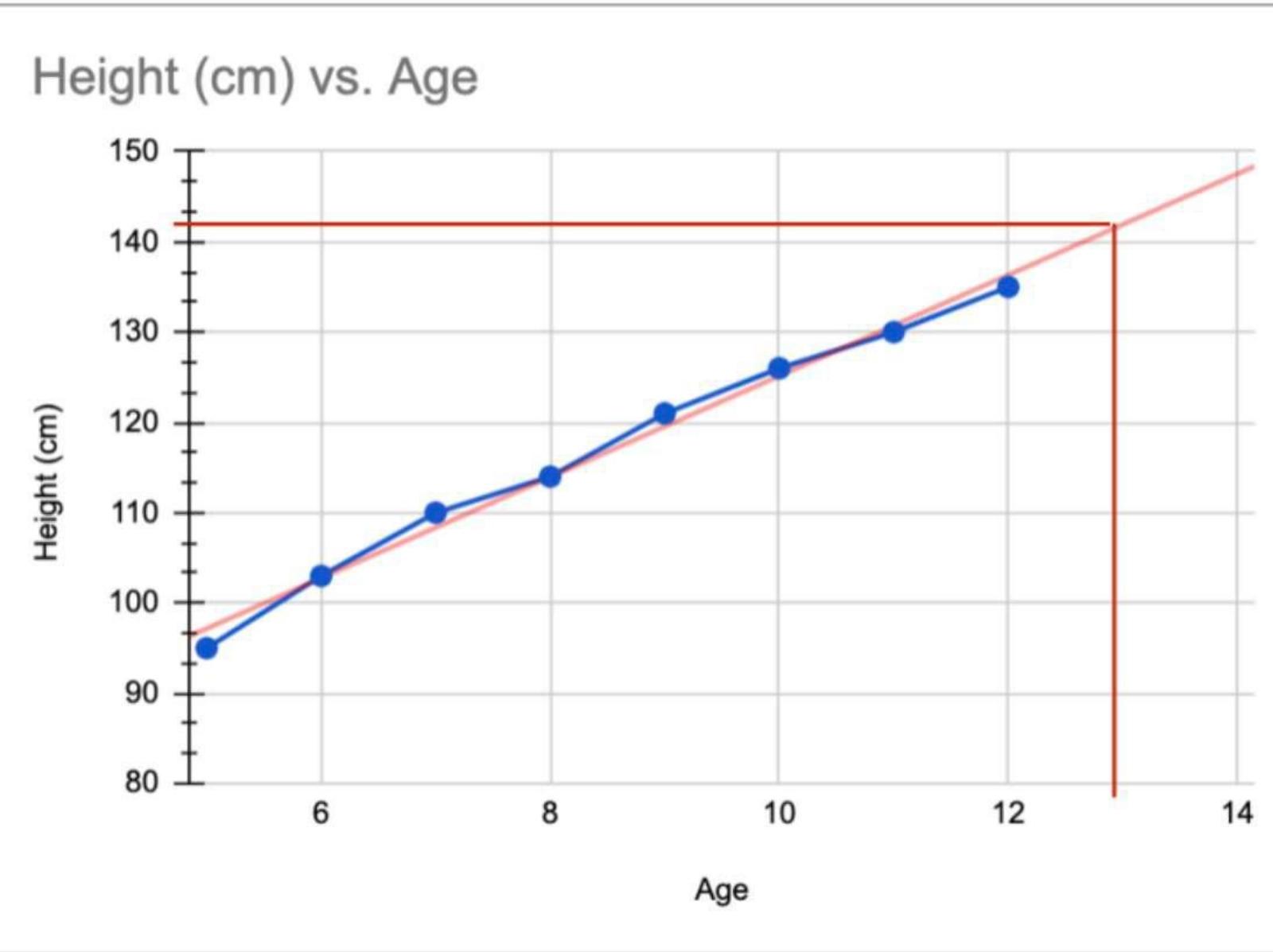
How tall was I at 13? 🧑

Height (cm) vs. Age



Extrapolate for **new** observations

Easy guess, isn't it ?



**What if there are 150 different features
and a million observations
for 2 million different people?**



**What if there are 150 different features
and a million observations
for 2 million different people?**

Every day.



Machine Learning is
already all **around us**



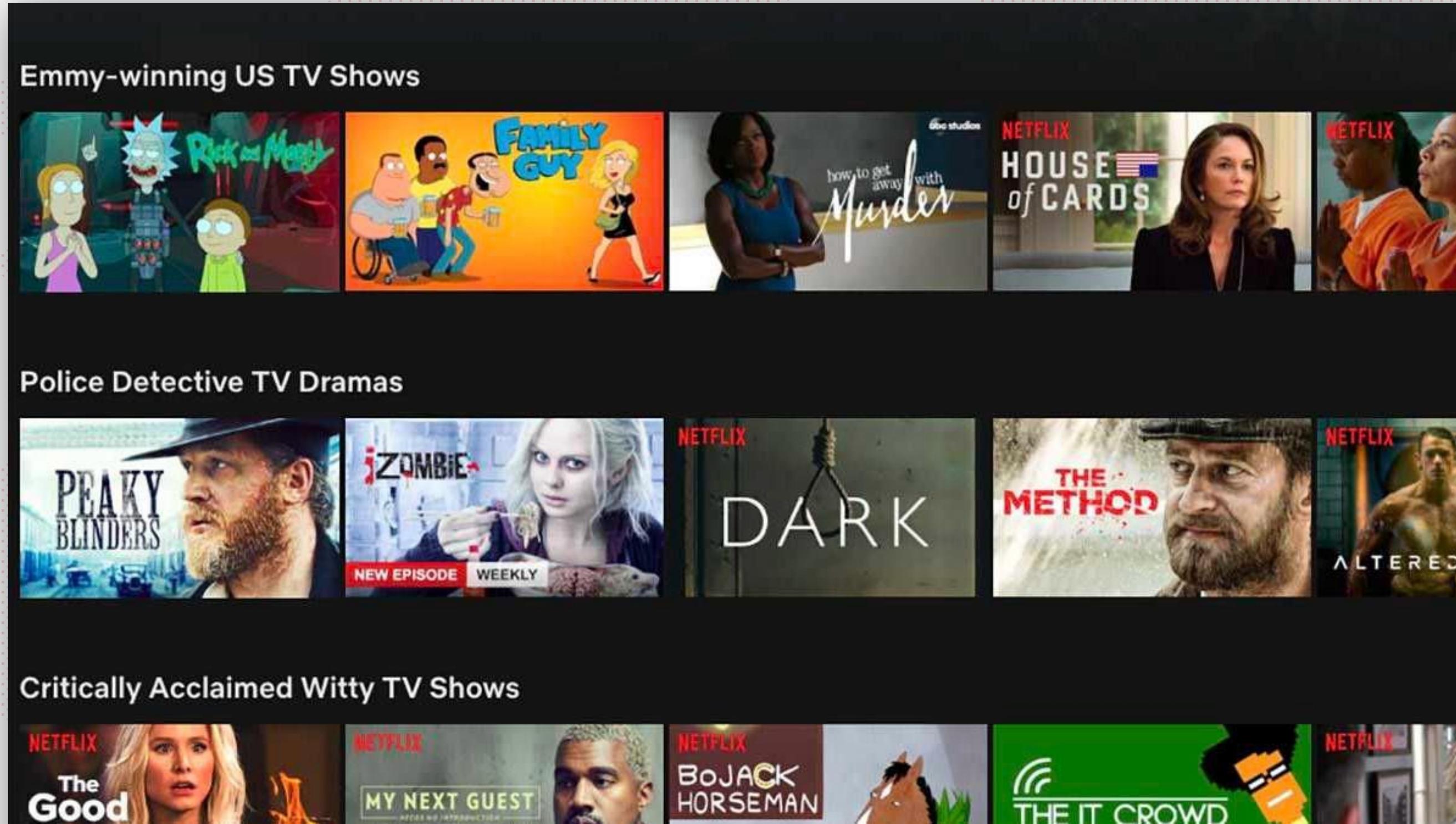
Computer Vision

Classifying and detecting visual data



Recommendation Engine

Predicting your next action



Natural Language Processing

Finding meaning in text



Time Series

Can past change predict future change

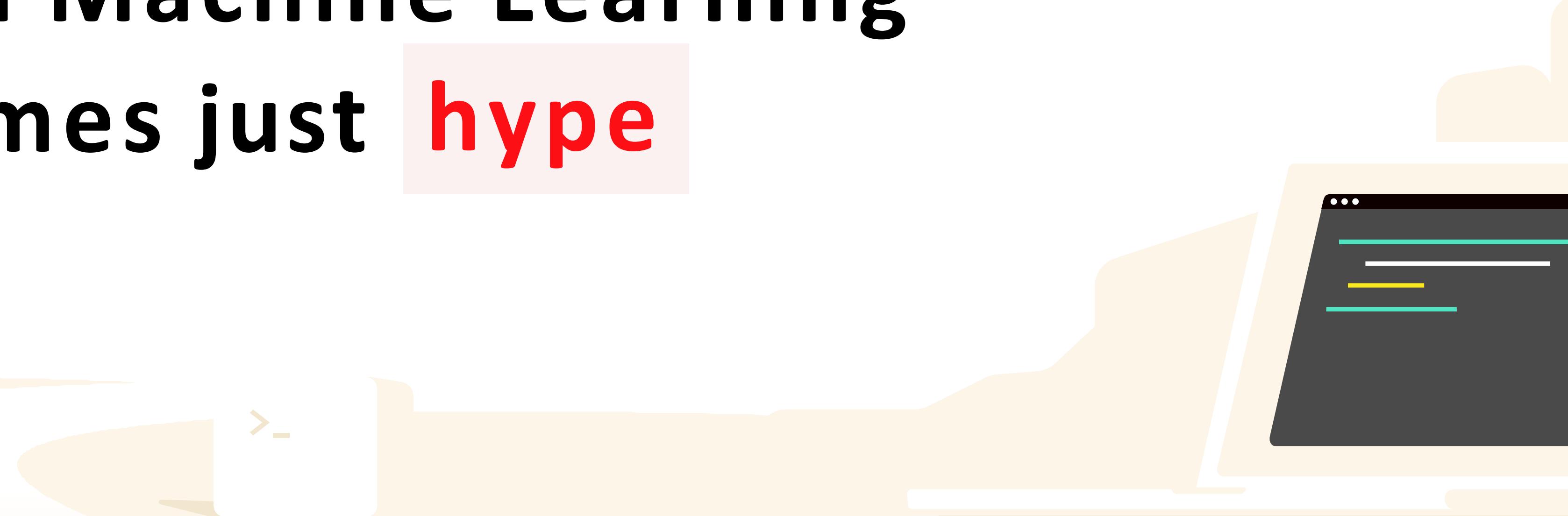


Anomaly Detection

“Predicting patterns” in reverse



When Machine Learning becomes just **hype**



- ✓ What is Machine Learning
- ✓ What is **NOT** Machine Learning
- ✓ Who are the people building ML
- ✓ Let's code our own models!
- ✓ What we didn't cover

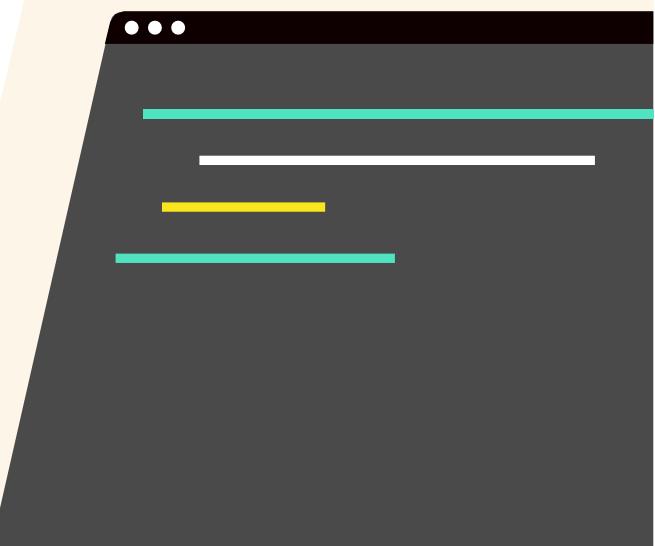


le wagon

Questions of **analytics**

Needs ML

Does **not** need ML



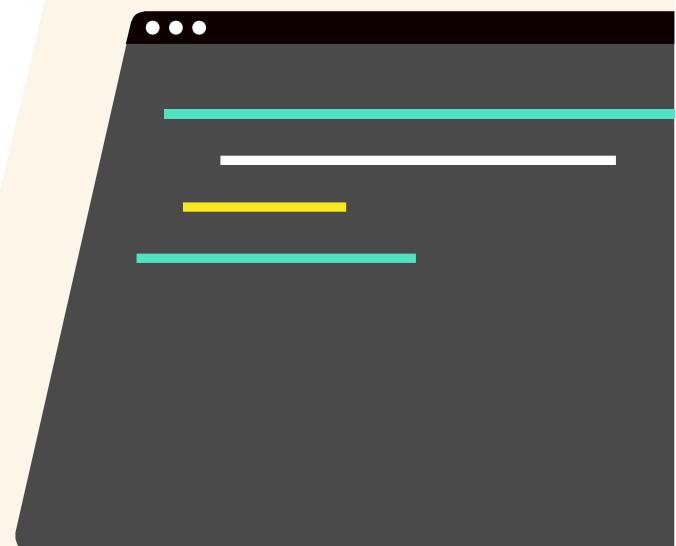
Questions of analytics

Needs ML

I want to predict salaries of potential hires

Does **not** need ML

Do women and men earn the same in our company?



Questions of **analytics**

Needs ML

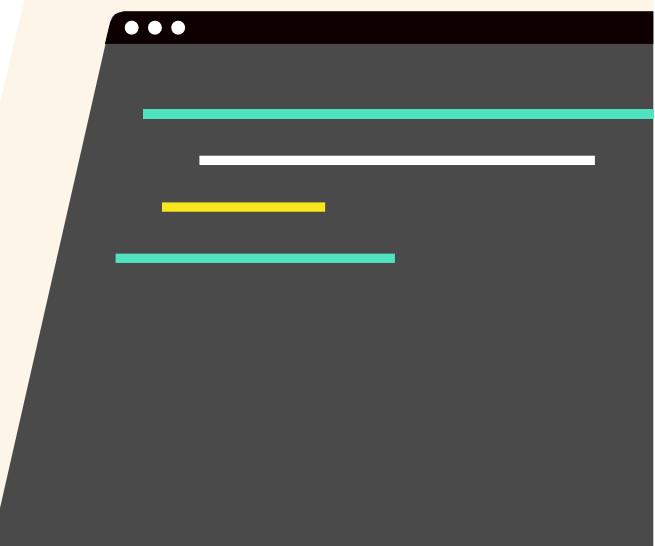
I want to predict salaries of potential hires

Why this marketing channel brings the most clicks?

Does **not need ML**

Do women and men earn the same in our company?

Which marketing channel brings the most clicks?



Questions of **analytics**

Needs ML

I want to predict salaries of potential hires

Why this marketing channel brings the most clicks?

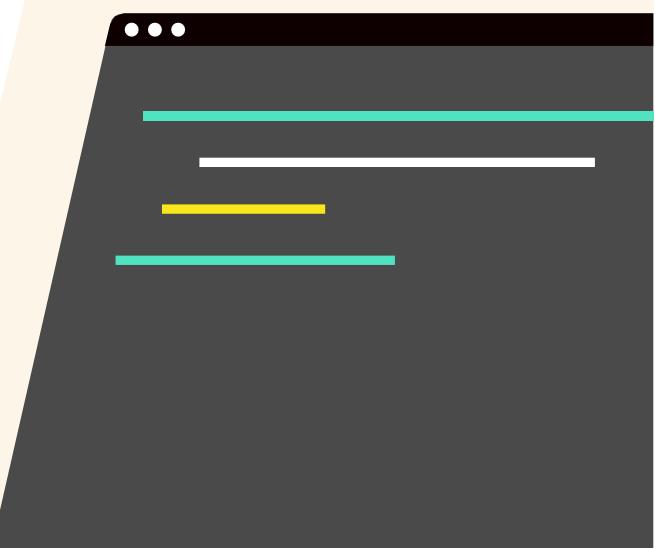
How are our sales looking if we open in this new country?

Does **not need ML**

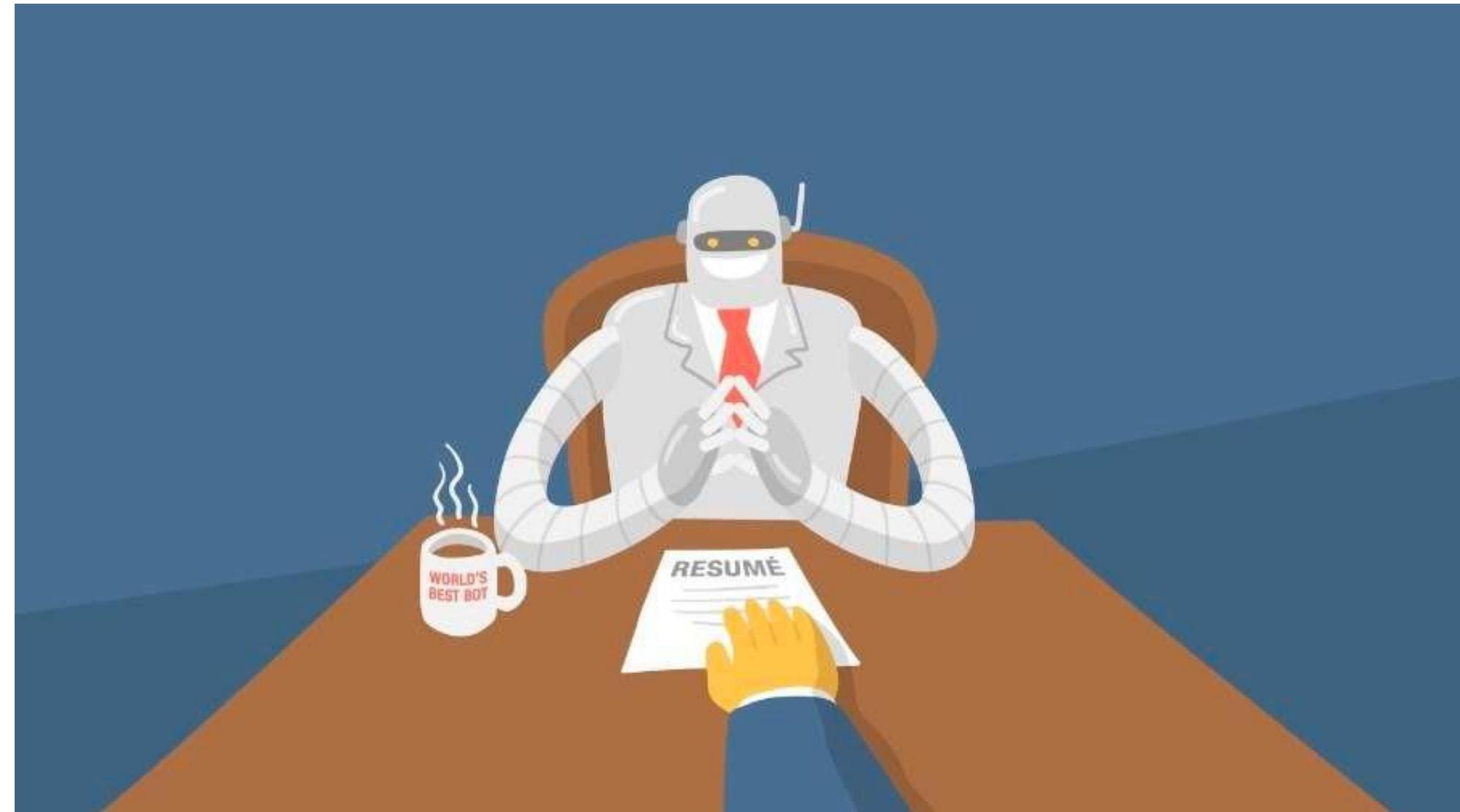
Do women and men earn the same in our company?

Which marketing channel brings the most clicks?

How are our sales looking for next quarter?



Missing ground truth



Machine Learning for hiring the perfect candidate!



Missing ground truth



vs



Missing ground truth

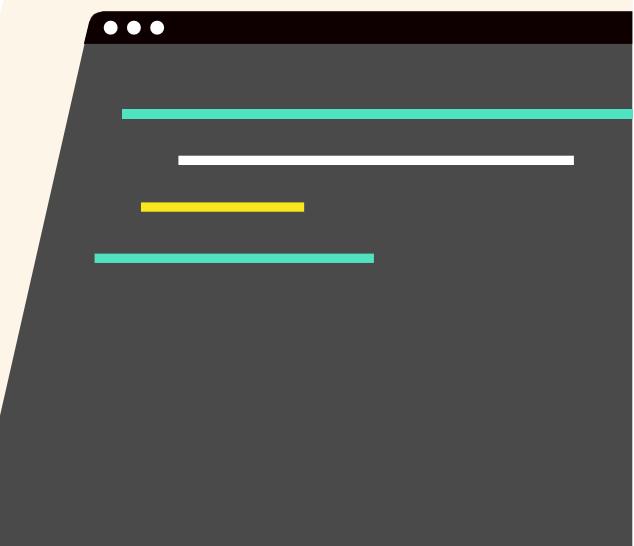


vs



Things have changed

Let's take a drug dose analysis 🍀



Things have changed

Let's take a drug dose analysis 🧬

Day 1 - 10mg

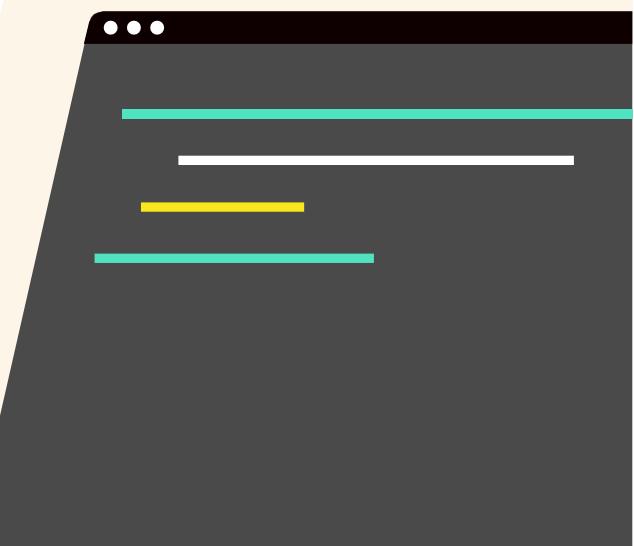
Day 2 - 15mg

Day 3 - 20mg

Day 6 - ?

Day 4 - 25mg

Day 5 - 30mg



Things have **really changed**

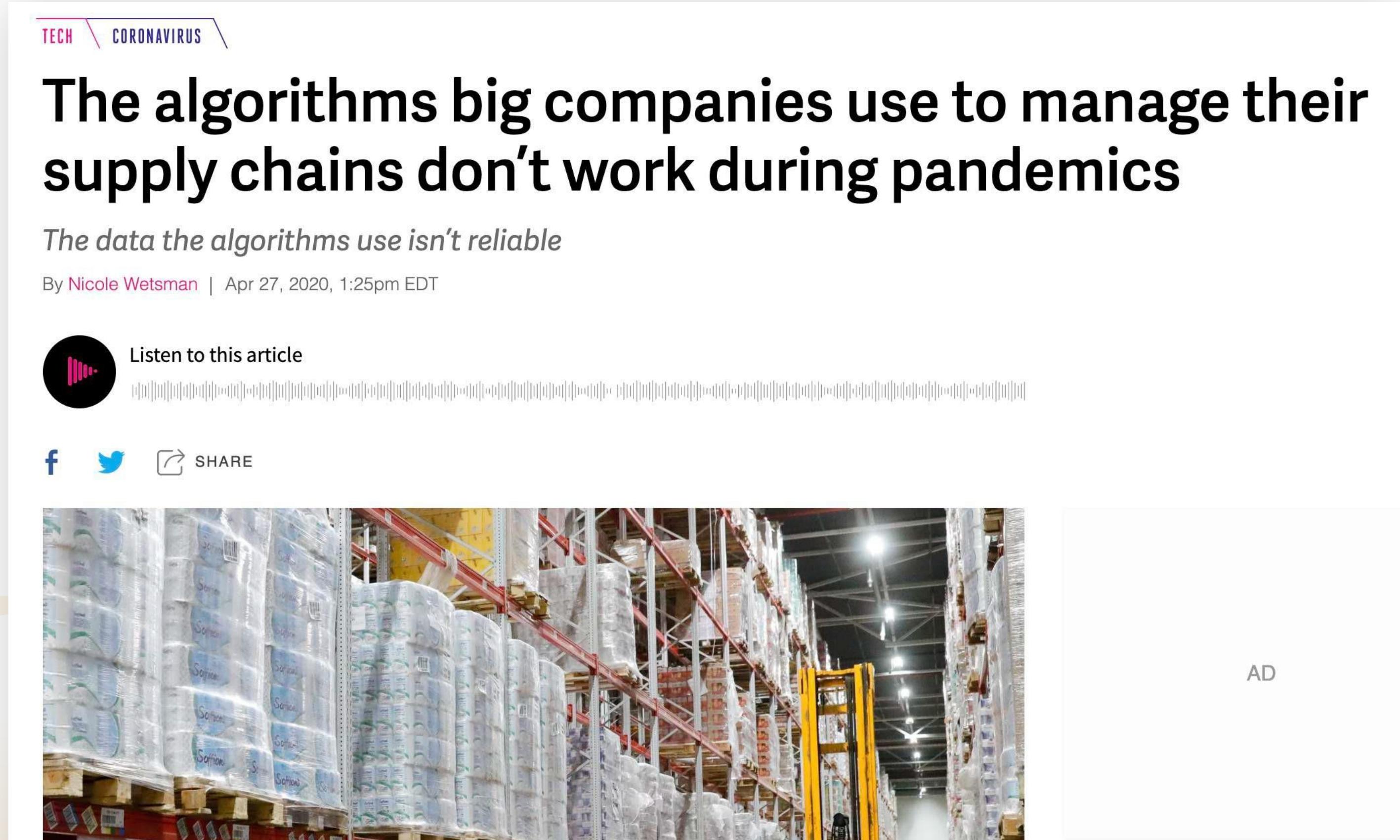
TECH \ CORONAVIRUS

The algorithms big companies use to manage their supply chains don't work during pandemics

The data the algorithms use isn't reliable

By Nicole Wetsman | Apr 27, 2020, 1:25pm EDT

Listen to this article



f    SHARE



AD

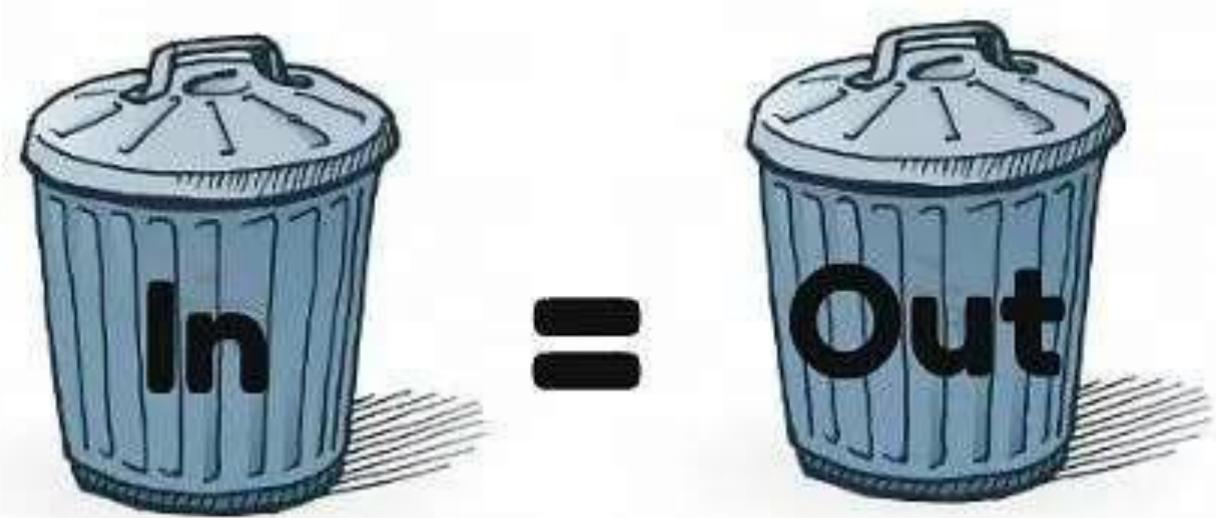
**Most firms that think they want advanced AI/ML really just
need linear regression on **cleaned-up data****

— Robin Hanson (@robinhanson) November 28, 2016

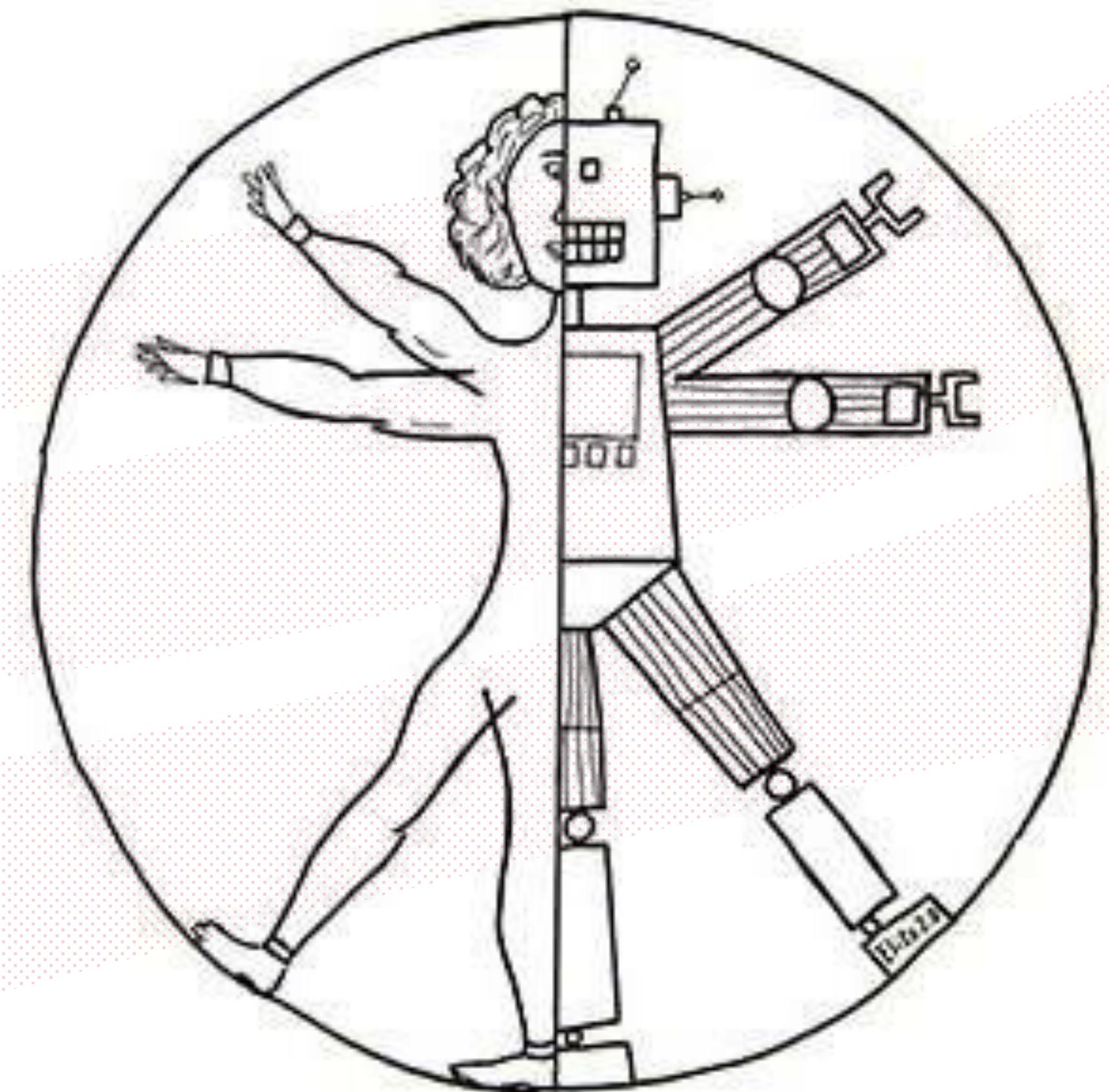


**Most firms that think they want advanced AI/ML really just
need linear regression on **cleaned-up data****

— Robin Hanson (@robinhanson) November 28, 2016



- ✓ What is Machine Learning
- ✓ What is NOT Machine Learning
- ✓ Who are the people building ML
- ✓ Let's code our own models!
- ✓ What we didn't cover



le wagon

MATH & STATS

**DATA
ANALYST**

INSIGHT &
COMMUNICATION

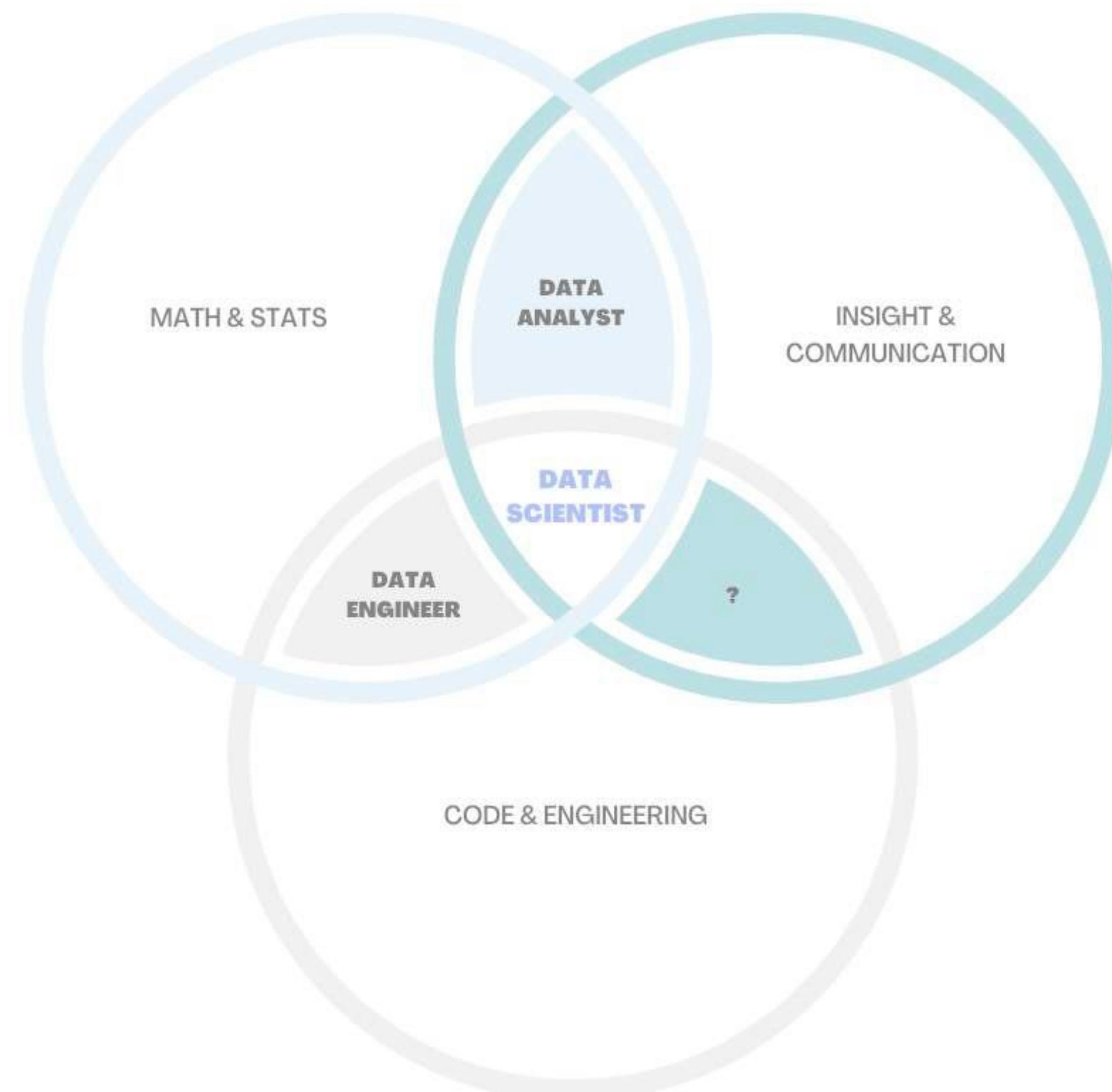
**DATA
ENGINEER**

**DATA
SCIENTIST**

?

CODE & ENGINEERING





Business Analyst

MATH & STATS

**DATA
ANALYST**

INSIGHT &
COMMUNICATION

**DATA
SCIENTIST**

**DATA
ENGINEER**

?

Data Product
Manager

CODE & ENGINEERING

Database
Admin





Finance & HR

Operations

Database Admin

Supply Chain

Business Analyst

DATA ANALYST

DATA SCIENTIST

DATA ENGINEER

CODE & ENGINEERING

INSIGHT &
COMMUNICATION

Data Product Manager

Digital Marketing

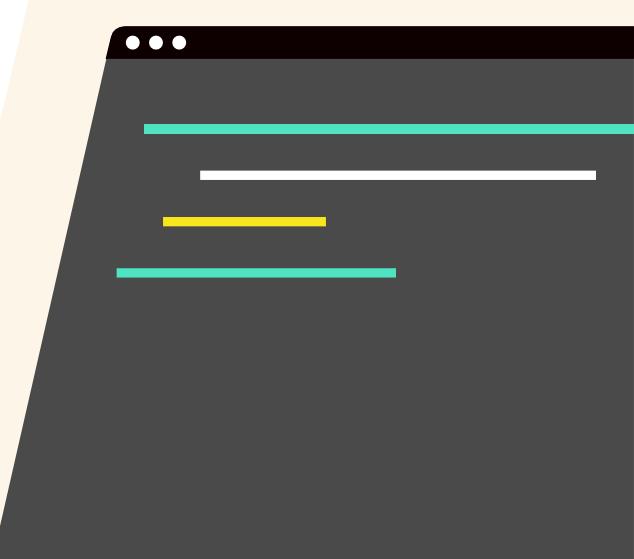
Strategy & Planning

Engineering

Data Science **as a skill**



>_

A small white rectangular box with a black border and a right-pointing arrow symbol (>) followed by a short horizontal line (_).

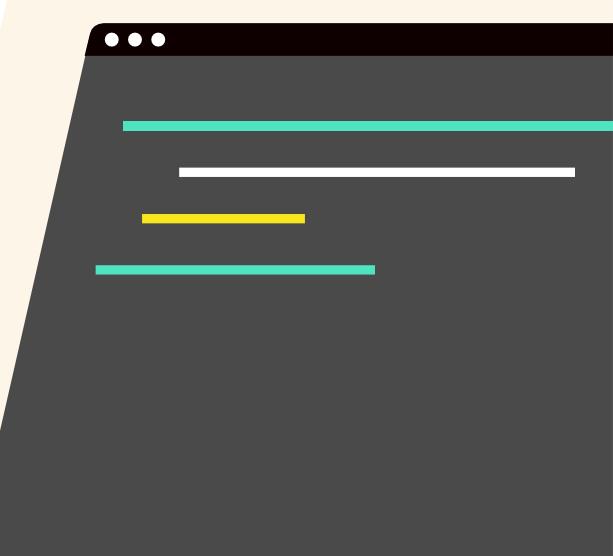
Data Science **as a skill**



Data Journalism



>_



Data Science **as a skill**



Data Journalism



Film Coloring



- ✓ What is Machine Learning
- ✓ What is NOT Machine Learning
- ✓ Who are the people building ML
- ✓ Let's code our own models!
- ✓ What we didn't cover



le wagon



Ready to get **nerdy**?



RStudio Connect

https://colorado.rstudio.com/rsc/jupyter-notebook-visualization/jupyter-static-visualization.html

Python Visualization Libraries

```
In [1]: import numpy as np
import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt
%matplotlib inline
```

Matplotlib

```
In [2]: np.random.seed(0)

mu = 200
sigma = 25
x = np.random.normal(mu, sigma, size=100)

fig, (ax0, ax1) = plt.subplots(ncols=2, figsize=(8, 4))

ax0.hist(x, 20, density=1, histtype='stepfilled', facecolor='g', alpha=0.75)
ax0.set_title('stepfilled')

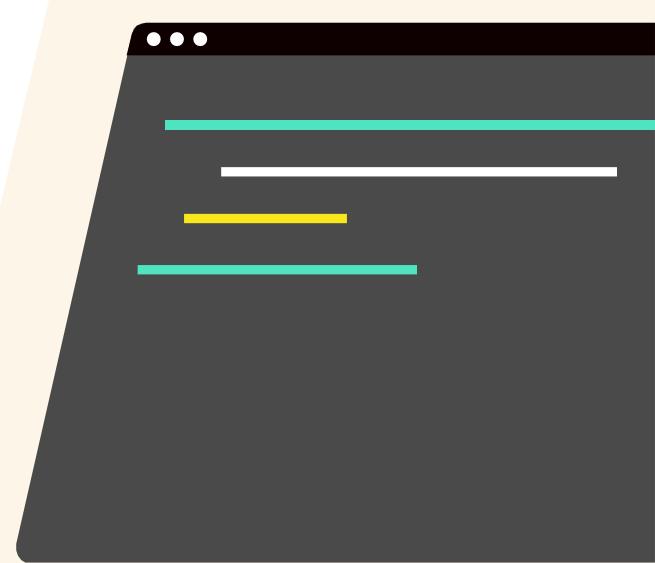
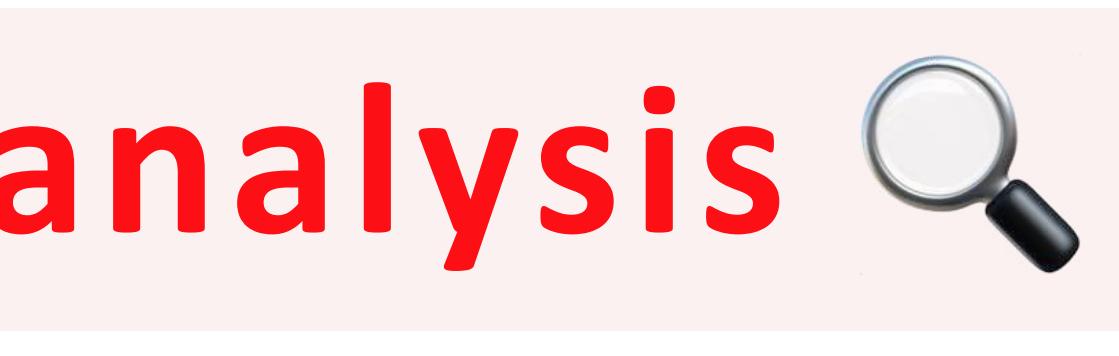
# Create a histogram by providing the bin edges (unequally spaced).
bins = [100, 150, 180, 195, 205, 220, 250, 300]
ax1.hist(x, bins, density=1, histtype='bar', rwidth=0.8)
ax1.set_title('unequal bins')

fig.tight_layout()
plt.show()
```

Two side-by-side histograms. The left histogram, titled 'stepfilled', shows a distribution of values with green bars that have a step-filled appearance. The right histogram, titled 'unequal bins', shows a distribution of values with blue bars of varying widths, illustrating an unequal binning scheme.

Before Machine Learning

We can do some analysis



What's the data

Understand what we have

- ✓ How many rows and columns?
- ✓ What are the columns we have?
- ✓ What are the averages?
- ✓ What are the minimums and maximums?

Methods to explore your data:

```
In [ ]: # to get the number of rows, columns  
salaries.shape
```

```
In [ ]: # to get the columns and their data types  
salaries.dtypes
```

```
In [ ]: # to get a readable summary of your data  
round(salaries.describe())
```

```
In [ ]: # to see only one column of the dataset  
salaries["Column Name"]
```

```
In [ ]: # to see multiple columns of the dataset  
salaries[["Column Name 1", "Column Name 2"]]
```



>_

Visualize the data

Get an intuition of what it tells you

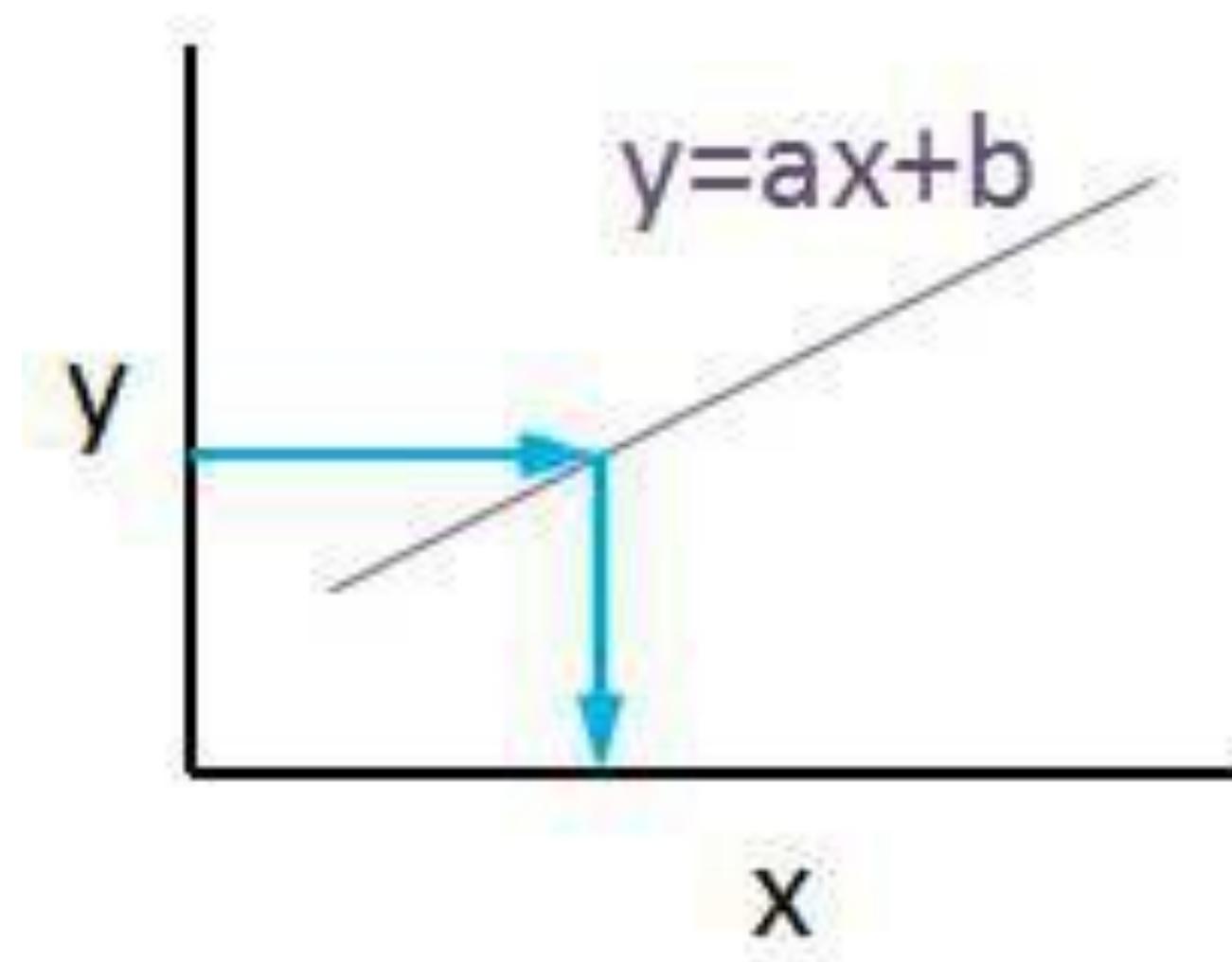
- ✓ How many of each category we have?
- ✓ Do any two columns relate to each other?
- ✓ Do some categories influence the output?
- ✓ First step of any Data Scientist!



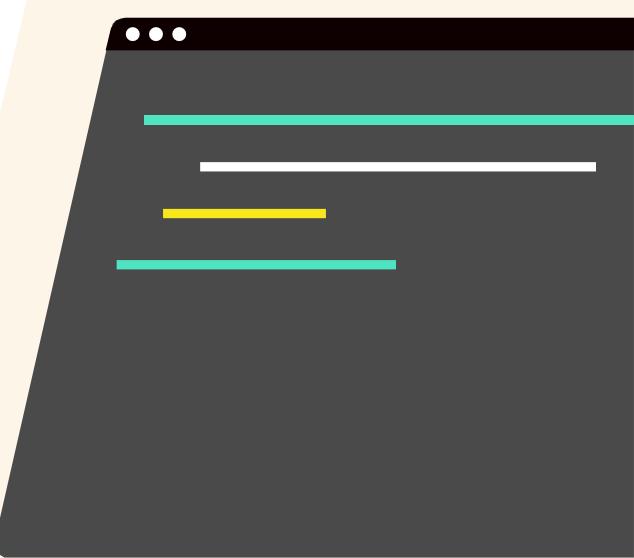
Challenge 1:

Predicting salaries 💰

Linear Regression

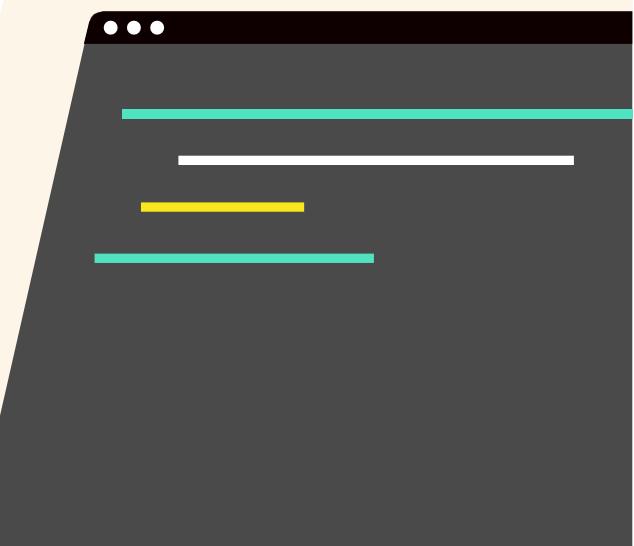


>

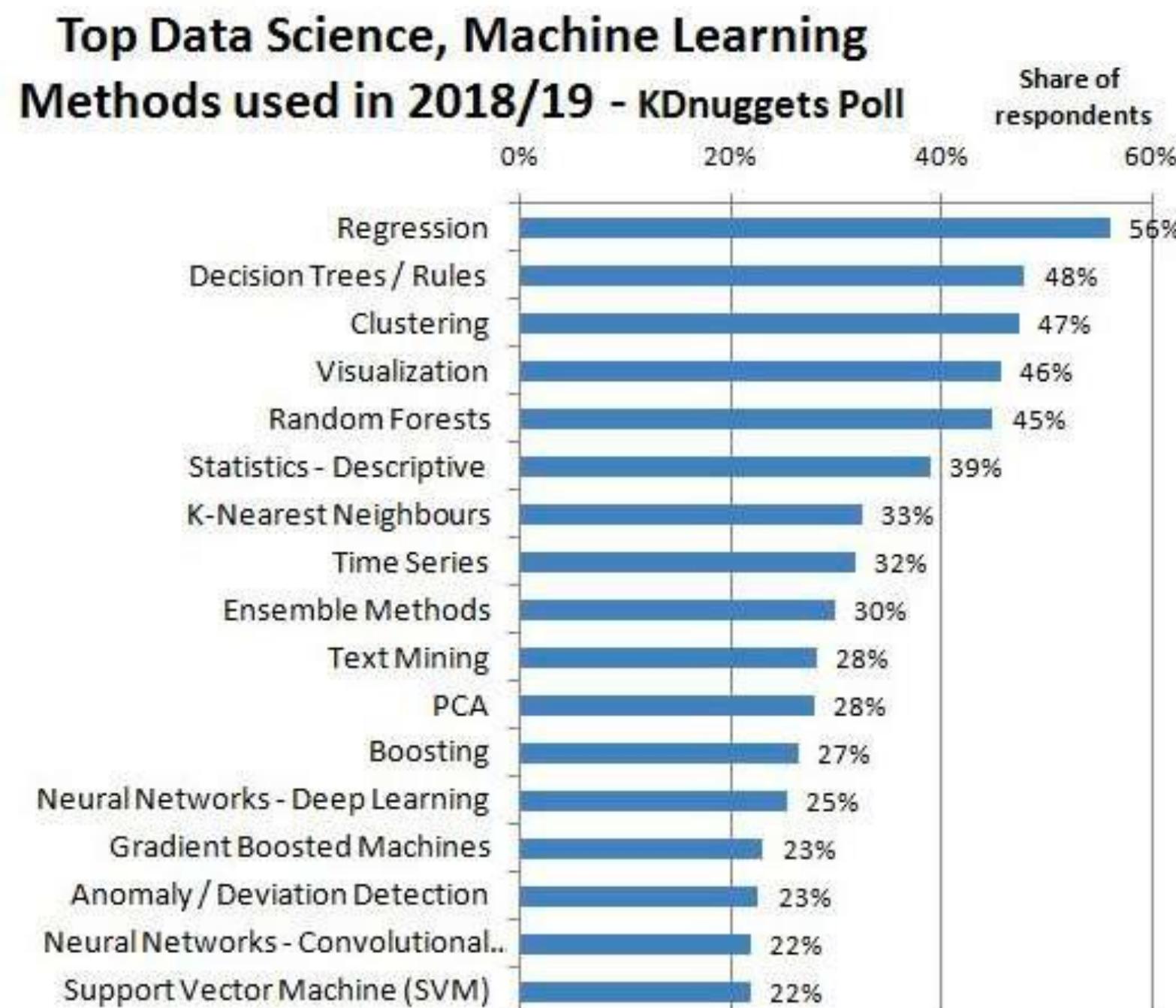


**Most firms that think they want advanced AI/ML really just
need linear regression on cleaned-up data**

— Robin Hanson (@robinhanson) November 28, 2016



Why Regression?



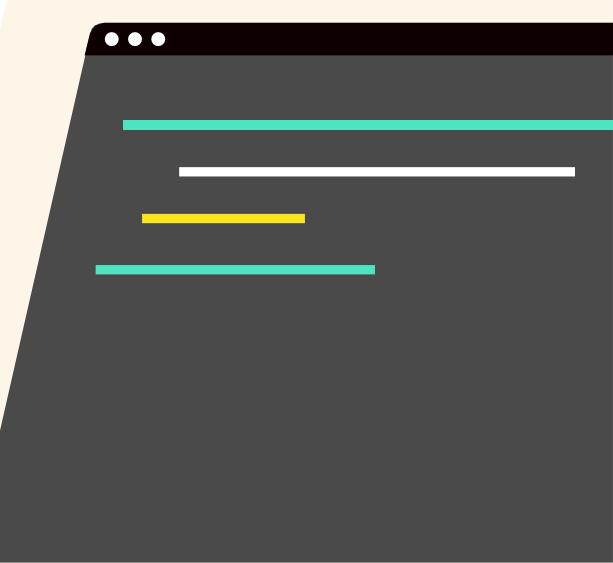
>_

Visual

demo on setosa.io!



>_



Road to Machine Learning

1. Select the **features** and **targets**
2. Import the model from Sklearn
3. Train the model
4. Score the model's performance
5. Predict with new data



Road to Machine Learning

1. Select the **features** and **target**

Setting our features (inputs):

```
In [ ]: # we can select all needed columns...
features = salaries[["Gender", "Age", "Department_code", "Years_exp", "Tenure (months)"]]
```

```
In [ ]: # ...or we can simply drop the not needed!
features = salaries.drop(["Department", "Gross"], axis="columns")
```

Setting our target (output):

```
In [ ]: # we can simply select the column we need
target = salaries["Gross"]
```



>_

Road to Machine Learning

1. Select the **features** and **targets**

2. Import the model from Sklearn

3. Train the model

4. Score the model's performance

5. Predict with new data

Once we find the model we need, it's easy to import

```
In [ ]: # the syntax looks like this  
from sklearn.MODEL_TYPES import MODEL_YOU_NEED
```

```
In [ ]: # with Linear Regression we need this  
from sklearn.linear_model import LinearRegression
```

After importing, we need to **initialize** the model, like this:

```
In [ ]: model = LinearRegression()
```



>_

Road to Machine Learning

1. Select the **features** and **targets**
2. Import the model from Sklearn
3. Train the model
4. Score the model's performance
5. Predict with new data

To train the model, we use the `.fit` method:

```
In [ ]: model.fit(features, target)
```

The `model` then finds the **best fitting** line between features and target



Road to Machine Learning

1. Select the **features** and **targets**
2. Import the model from Sklearn
3. Train the model
4. Score the model's performance
5. Predict with new data



To score the model, we use the `.score` method:

```
In [ ]: model.score(features, targets)
```

We need to give it some **test data** to do the scoring

Road to Machine Learning

1. Select the **features** and **targets**
2. Import the model from Sklearn
3. Train the model
4. Score the model's performance
5. Predict with new data



```
To predict -- you guessed it ;)  
We use the .predict method:  
In [ ]: model.predict(new_data)  
new_data is the info we want our model to use to predict a new target (output).  
In our case, it's a new hire !
```

Road to Machine Learning

1. Select the **features** and **targets**

2. Import the model from Sklearn

3. Train the model

4. Score the model's performance

5. Predict with new data

6. Explaining the model >

The things that a Linear Regression model "learns" are **coefficients** and **intercept**.

The **coefficients** show how each feature influences the target:

```
In [ ]: model.coef_
```

The **intercept** shows what would the target be when all features are at zero:

```
In [ ]: model.intercept_
```

That easy?? 😱
Yes... but no



That easy?? 😱

Yes... but no

```
model.fit(features, target)
```

```
model.score(features, targets)
```



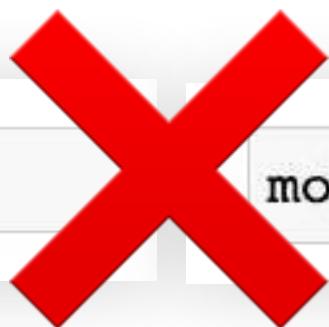
>_

That easy?? 😱

Yes... but no

```
model.fit(features, target)
```

```
model.score(features, targets)
```



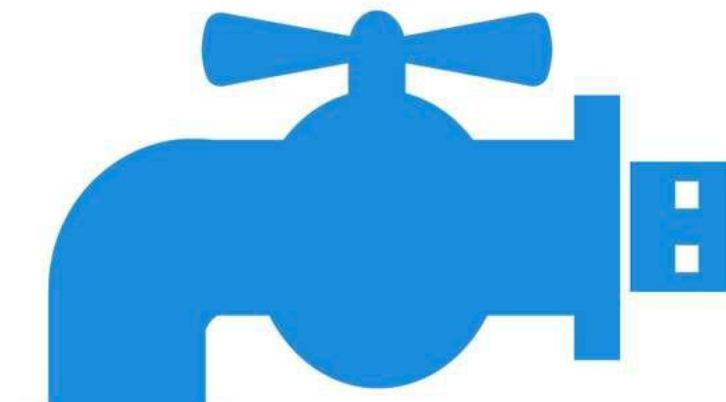
That easy?? 😱

More on this in Challenge 2

```
model.fit(features, target)
```



```
model.score(features, targets)
```



```
101  
1101  
01101  
110101  
110001  
1001
```

Data Leakage

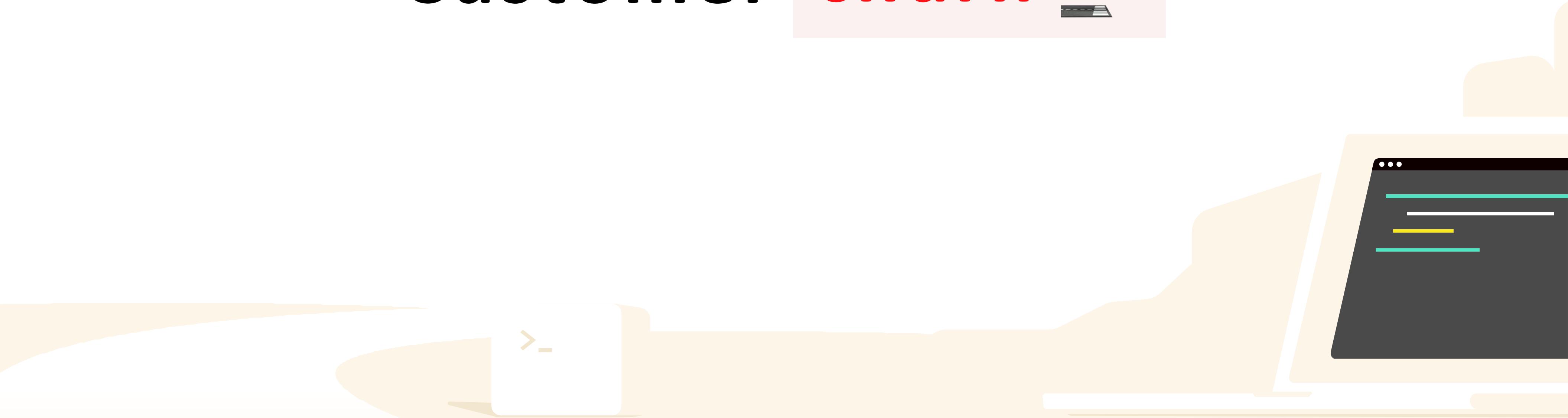


>_



Optional Challenge 2:

Customer **churn**



Regression vs. Classification



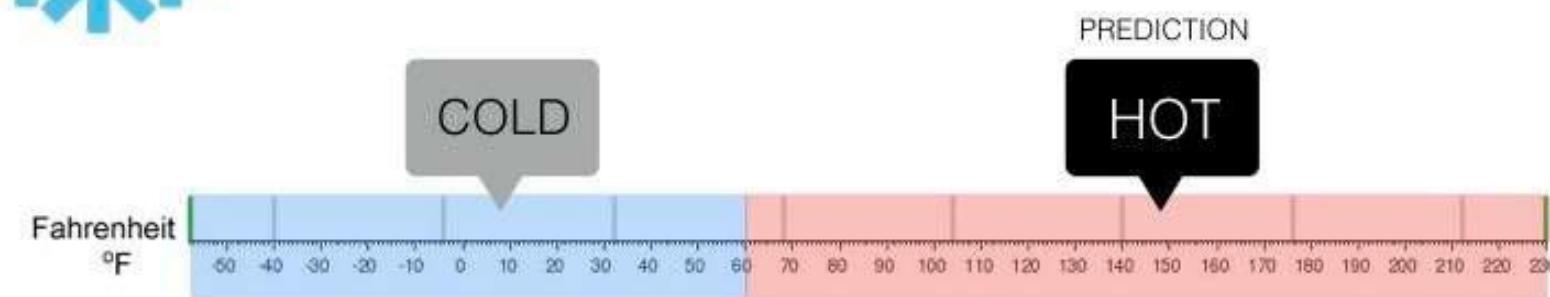
Regression

What is the temperature going to be tomorrow?



Classification

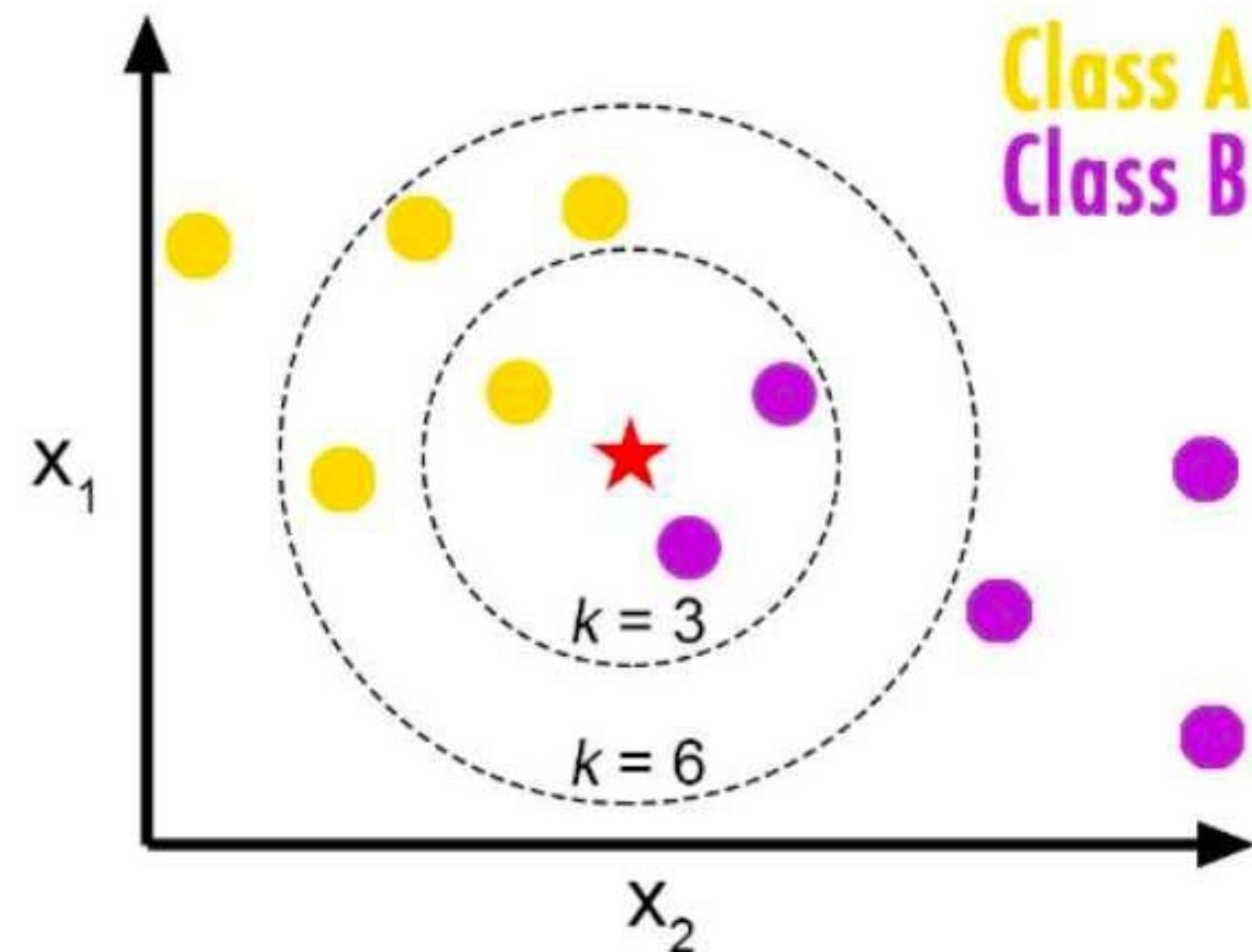
Will it be Cold or Hot tomorrow?



>_



K-Nearest Neighbor

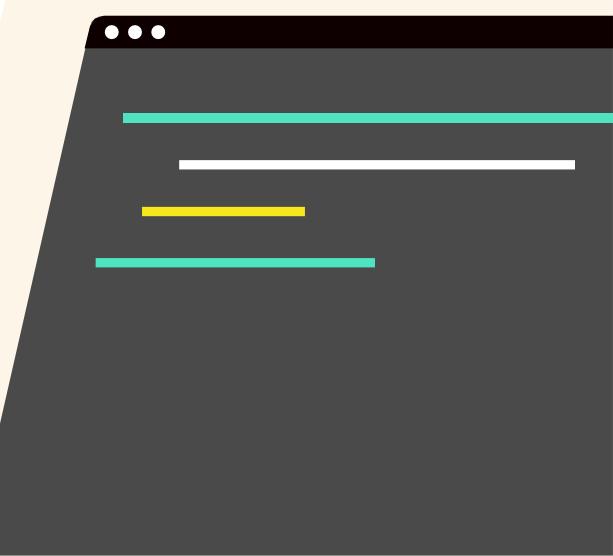


>-

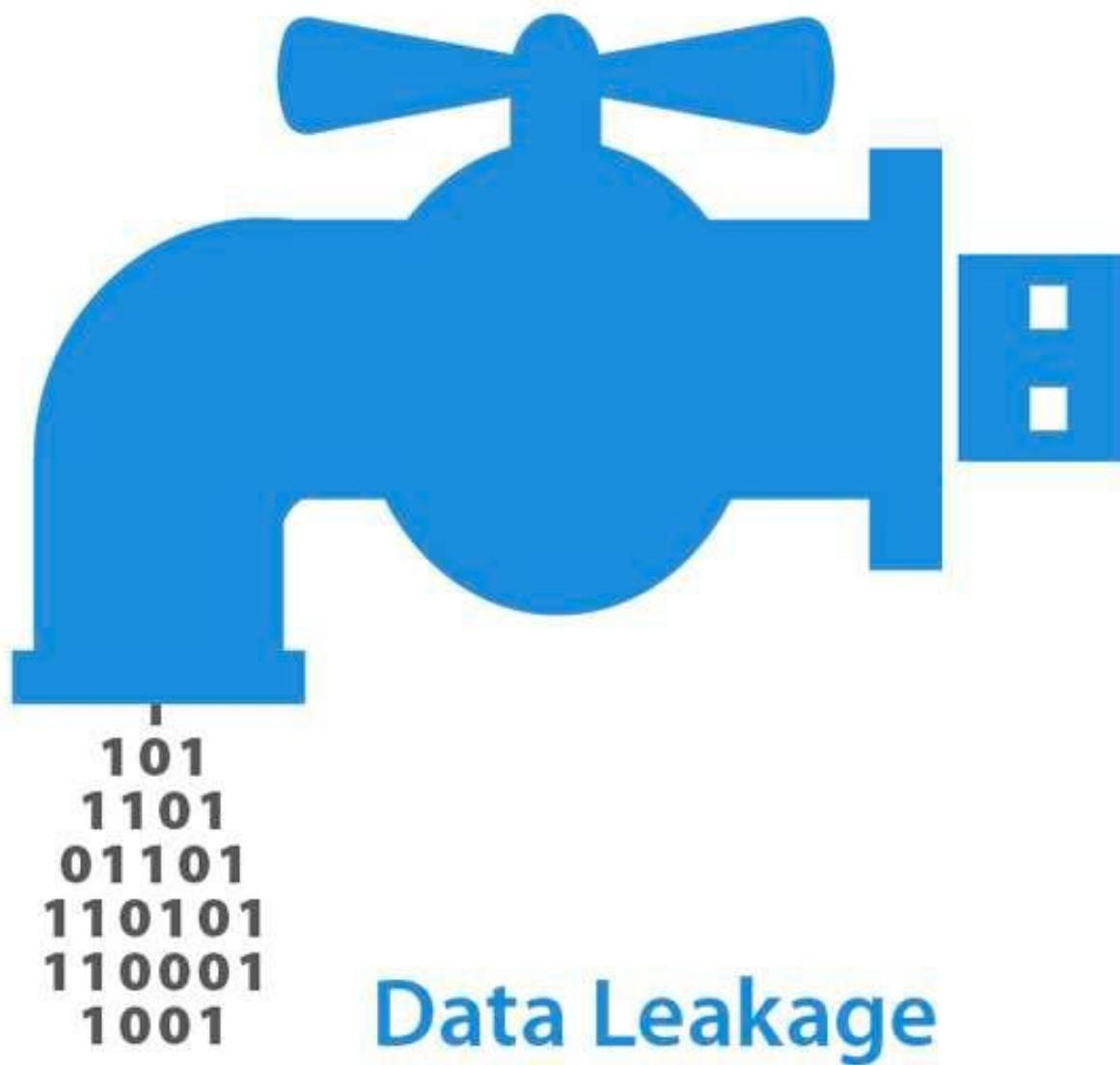
Visual demo on ml-playground!



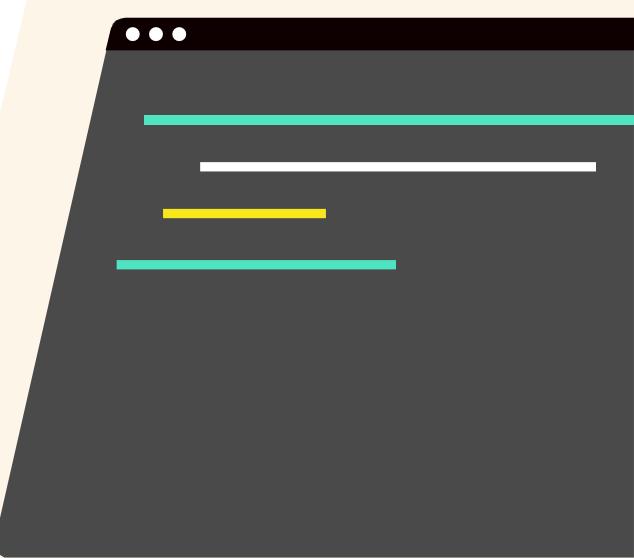
>_

A small white text element with a yellow outline, positioned between the red icon and the dark screen.

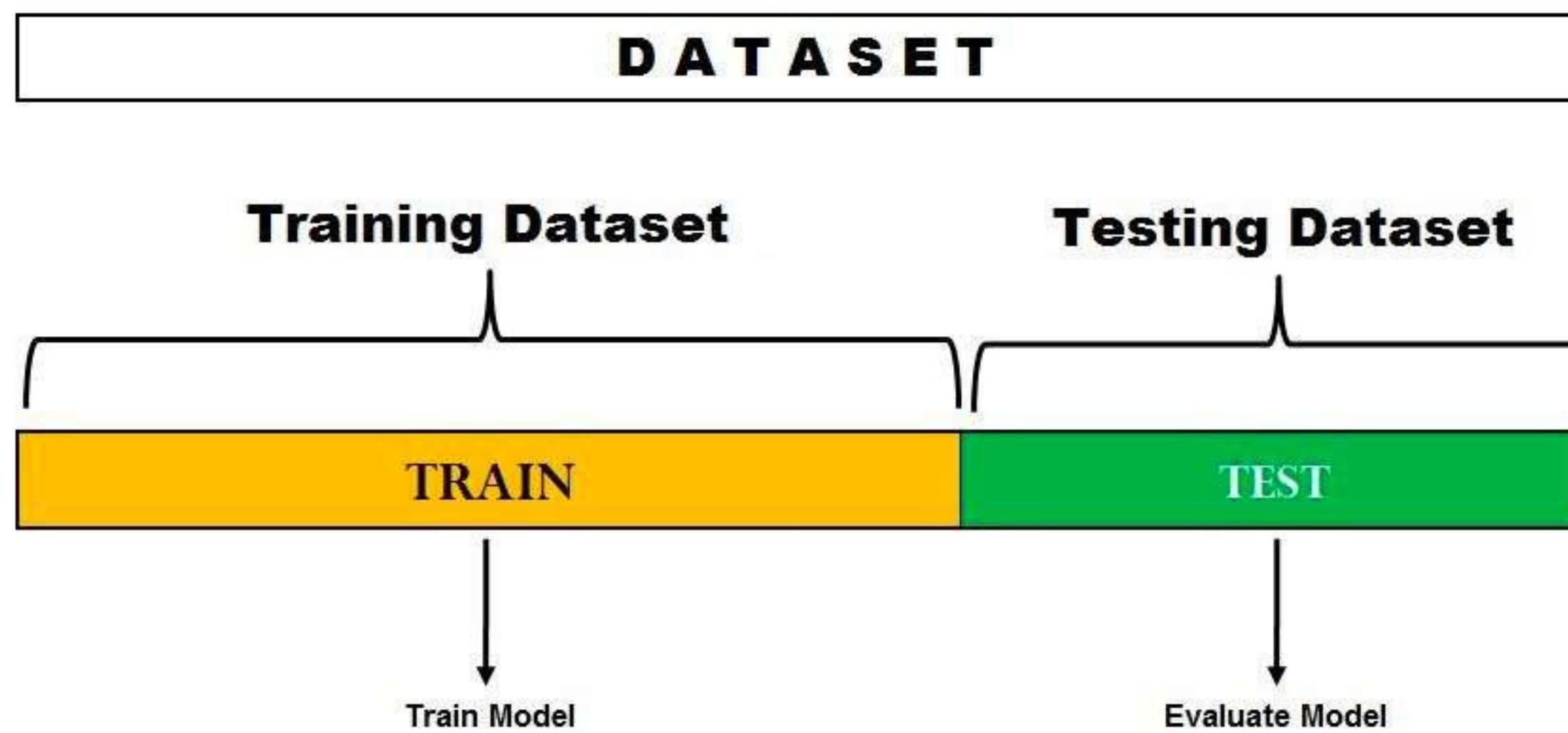
We have a leak



>_



Holdout Method



You now have the tools

Your turn! 



Congrats! 🙌 Now you know...



le wagon



Congrats! 🙌 Now you know...

What is the concept of Machine Learning

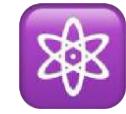


le wagon

Congrats! 🙌 Now you know...

What is the concept of Machine Learning 🤖

The basics of Jupyter Notebook - the #1 tool of any Data Scientist

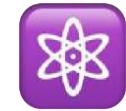


le wagon

Congrats! 🙌 Now you know...

What is the concept of Machine Learning 🤖

The basics of Jupyter Notebook - the #1 tool of any Data



Scientist How to use Python libraries - of which there are



thousands



le wagon

Congrats! 🙌 Now you know...

What is the concept of Machine Learning 🤖

The basics of Jupyter Notebook - the #1 tool of any Data



Scientist How to use Python libraries - of which there are 😂

thousands

12
34

How to import and visualize a CSV dataset

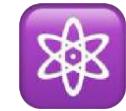


le wagon

Congrats! 🙌 Now you know...

What is the concept of Machine Learning 🤖

The basics of Jupyter Notebook - the #1 tool of any Data



Scientist How to use Python libraries - of which there are 😂

thousands

12
34

How to import and visualize a CSV dataset



le wagon

And of course - how to build your own ML models

- ✓ What is Machine Learning
- ✓ What is NOT Machine Learning
- ✓ Who are the people building ML
- ✓ Let's code our own models!
- ✓ What we did not cover



le wagon



Other model types

Decision tree based

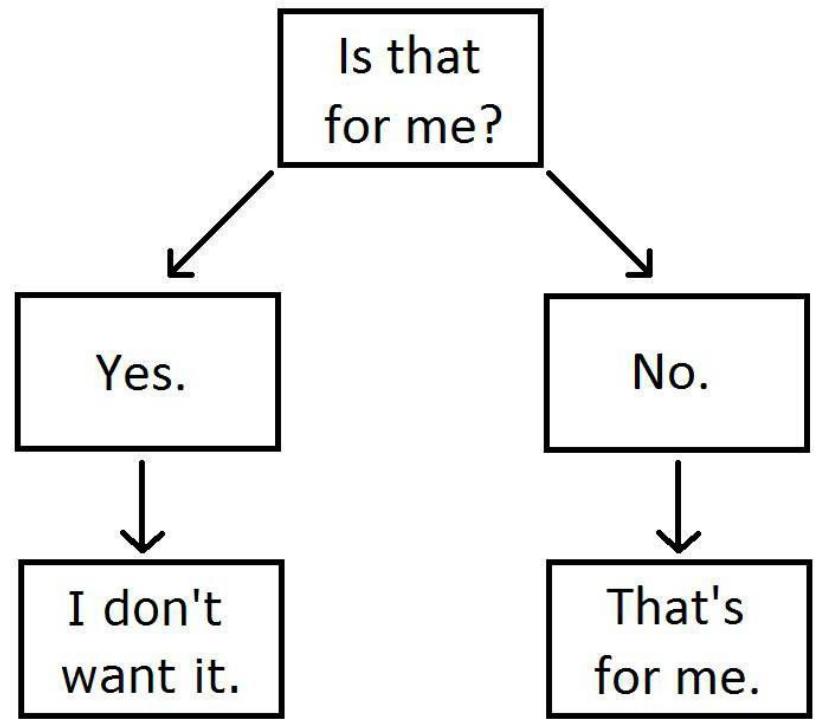


Probabilistic

Logistic

Other model types

My Cat's Decision-Making Tree.



Decision tree based

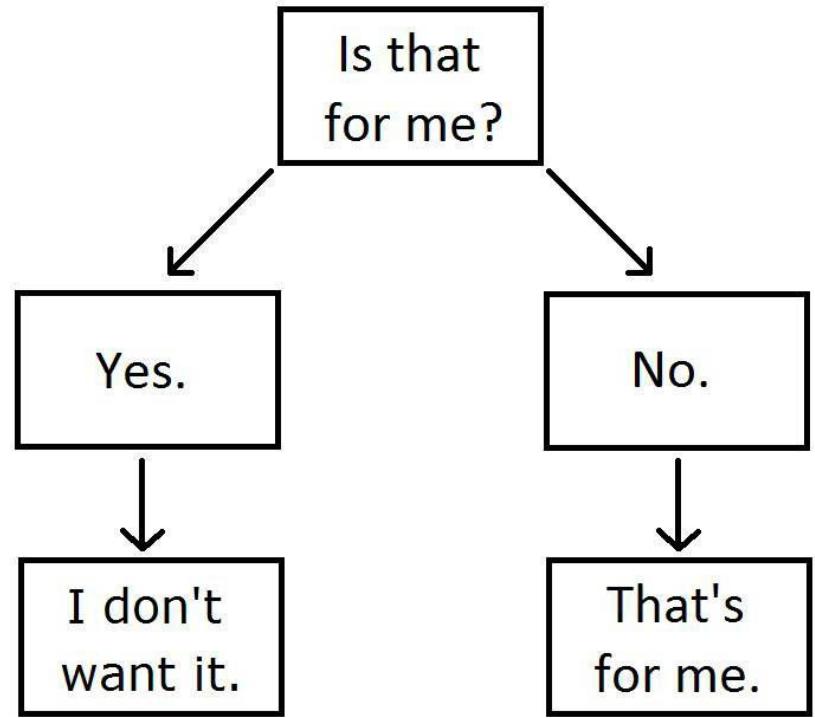


Probabilistic

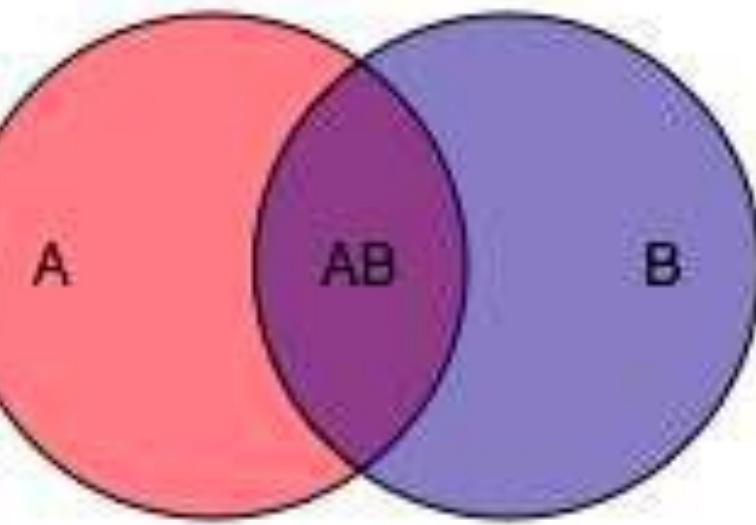
Logistic

Other model types

My Cat's Decision-Making Tree.



Decision tree based

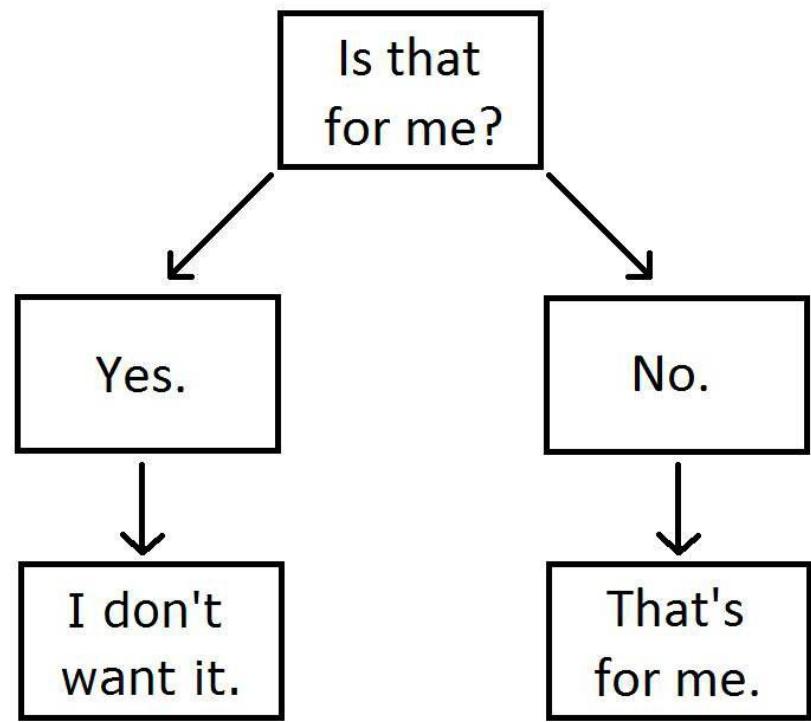


Probabilistic

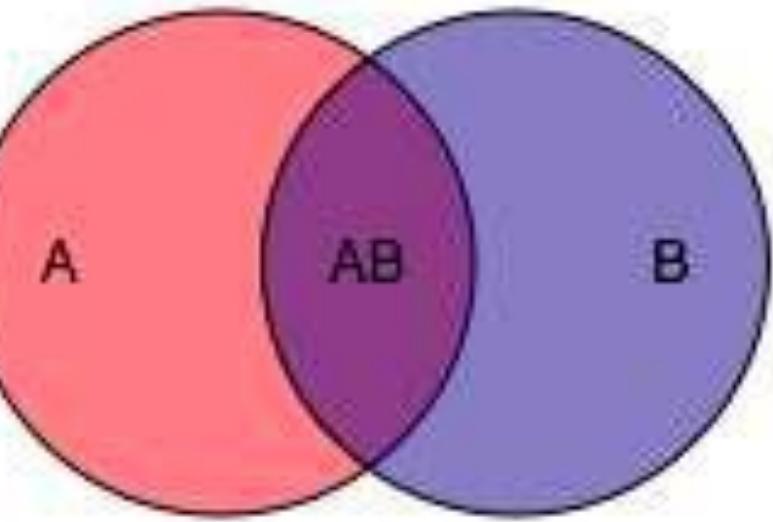
Logistic

Other model types

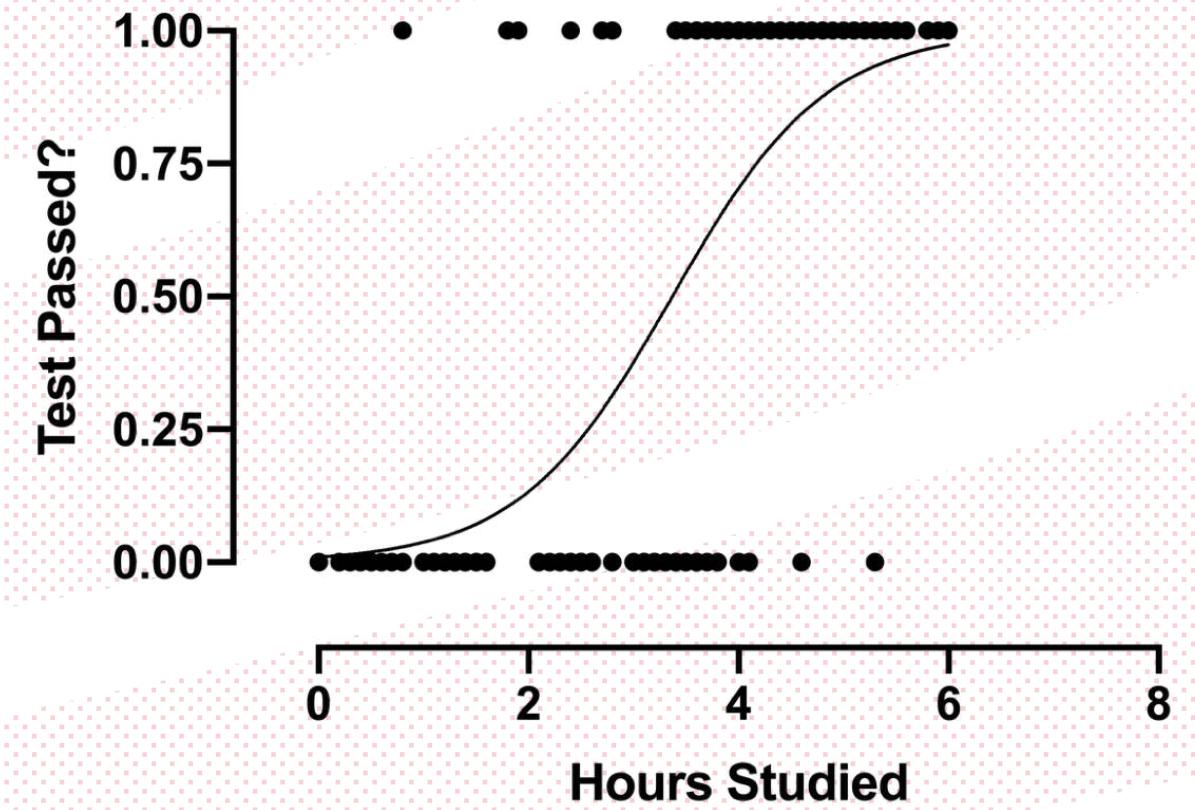
My Cat's Decision-Making Tree.



Decision tree based

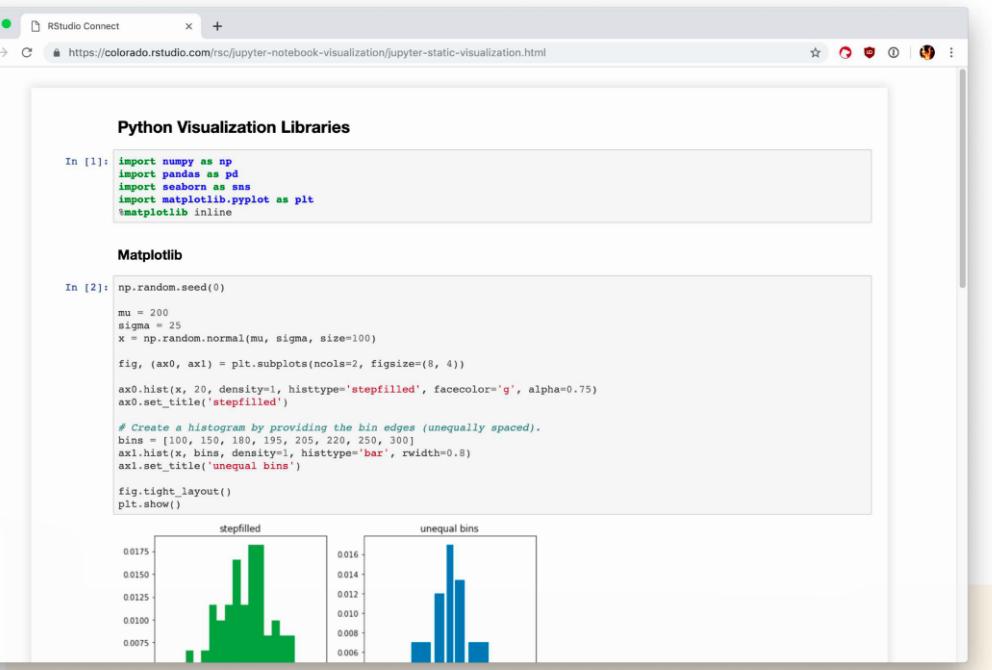


Probabilistic



Logistic

Making it a product



The screenshot shows a Jupyter Notebook interface with two code cells and their corresponding plots.

Python Visualization Libraries

```
In [1]: import numpy as np
import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt
%matplotlib inline
```

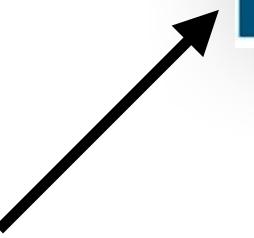
Matplotlib

```
In [2]: np.random.seed(0)
mu = 200
sigma = 25
x = np.random.normal(mu, sigma, size=100)
fig, (ax0, ax1) = plt.subplots(nrows=2, figsize=(8, 4))
ax0.hist(x, 20, density=1, histtype='stepfilled', facecolor='g', alpha=0.75)
ax0.set_title('stepfilled')
# Create a histogram by providing the bin edges (unequally spaced).
bins = [100, 150, 180, 195, 205, 220, 250, 300]
ax1.hist(x, bins, density=1, histtype='bar', rwidth=0.8)
ax1.set_title('unequal bins')
fig.tight_layout()
plt.show()
```

The notebook displays two histograms. The first histogram, titled "stepfilled", uses 20 equally spaced bins and is filled with green steps. The second histogram, titled "unequal bins", uses unequally spaced bins defined by the list [100, 150, 180, 195, 205, 220, 250, 300] and is shown as a bar chart with blue bars.



Making it a product

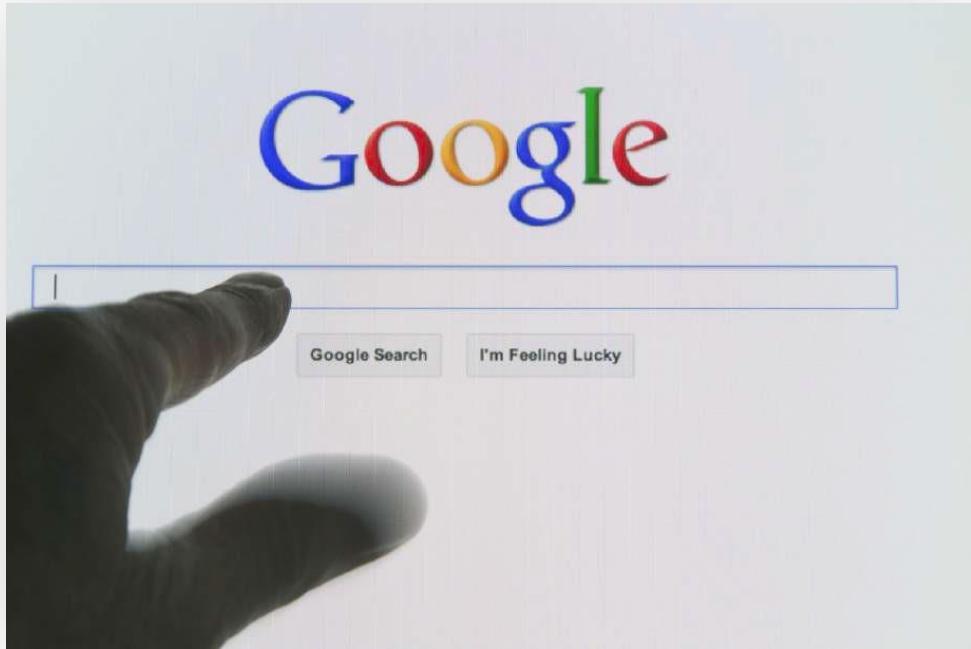


```
In [1]: import numpy as np
import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt
%matplotlib inline

In [2]: np.random.seed(0)
mu = 200
sigma = 25
x = np.random.normal(mu, sigma, size=100)
fig, (ax0, ax1) = plt.subplots(nrows=2, figsize=(8, 4))
ax0.hist(x, 20, density=1, histtype='stepfilled', facecolor='g', alpha=0.75)
ax0.set_title('stepfilled')
# Create a histogram by providing the bin edges (unequally spaced).
bins = [100, 150, 180, 195, 205, 220, 250, 300]
ax1.hist(x, bins, density=1, histtype='bar', rwidth=0.8)
ax1.set_title('unequal bins')
fig.tight_layout()
plt.show()
```



Making it a product



RStudio Connect

<https://colorado.rstudio.com/rsc/jupyter-notebook-visualization/jupyter-static-visualization.html>

Python Visualization Libraries

```
In [1]: import numpy as np
import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt
%matplotlib inline
```

Matplotlib

```
In [2]: np.random.seed(0)

mu = 200
sigma = 25
x = np.random.normal(mu, sigma, size=100)

fig, (ax0, ax1) = plt.subplots(ncols=2, figsize=(8, 4))

ax0.hist(x, 20, density=1, histtype='stepfilled', facecolor='g', alpha=0.75)
ax0.set_title('stepfilled')

# Create a histogram by providing the bin edges (unequally spaced).
bins = [100, 150, 180, 195, 205, 220, 250, 300]
ax1.hist(x, bins, density=1, histtype='bar', rwidth=0.8)
ax1.set_title('unequal bins')

fig.tight_layout()
plt.show()
```





ML + domain knowledge



ML + domain knowledge

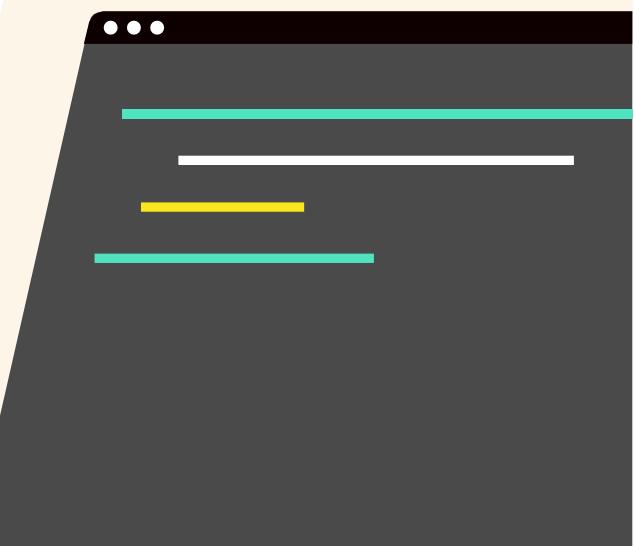
But...



ML + domain knowledge

But...

That's you! 🙌



Thank you!



Want to go further ? Head over to
www.lewagon.com



le wagon

Leave us **feedback.**

Le Wagon's workshops are driven by our community!

Please please **PLEASE** tell us what you think.

From the bottom of our hearts: thank you! ❤



le wagon

