Real-Time Machine Learning in Streaming Data Pipelines

Jakub Nowacki

DataMass Gdańsk 2017

whoami

Lead Data Scientist @ SigDelta (sigdelta.com)

Trainer @ Sages (sages.com.pl)

I can code, I do maths

@jsnowacki

j.nowacki@sigdelta.com

The rise of Big Data

The rise of streaming

DataMass

Flink Forward

7/16 talks about streaming

Separate track

Whole conference

The rise of Machine Learning (and AI)

amazon web services

- Amazon Lex
- Amazon Polly
- Amazon Rekognition
- Amazon Machine Learning
- Apache MXnet on AWS
- TensorFlow on AWS
- AWS Deep Learning AMIs

Google Cloud Platform

- Cloud Machine Learning Engine
- Clouds Jobs API
- Cloud Natural Language API
- Cloud Speech API
- Cloud Translation API
- Cloud Vision API
- Cloud Video Intelligence API

Microsoft Azure

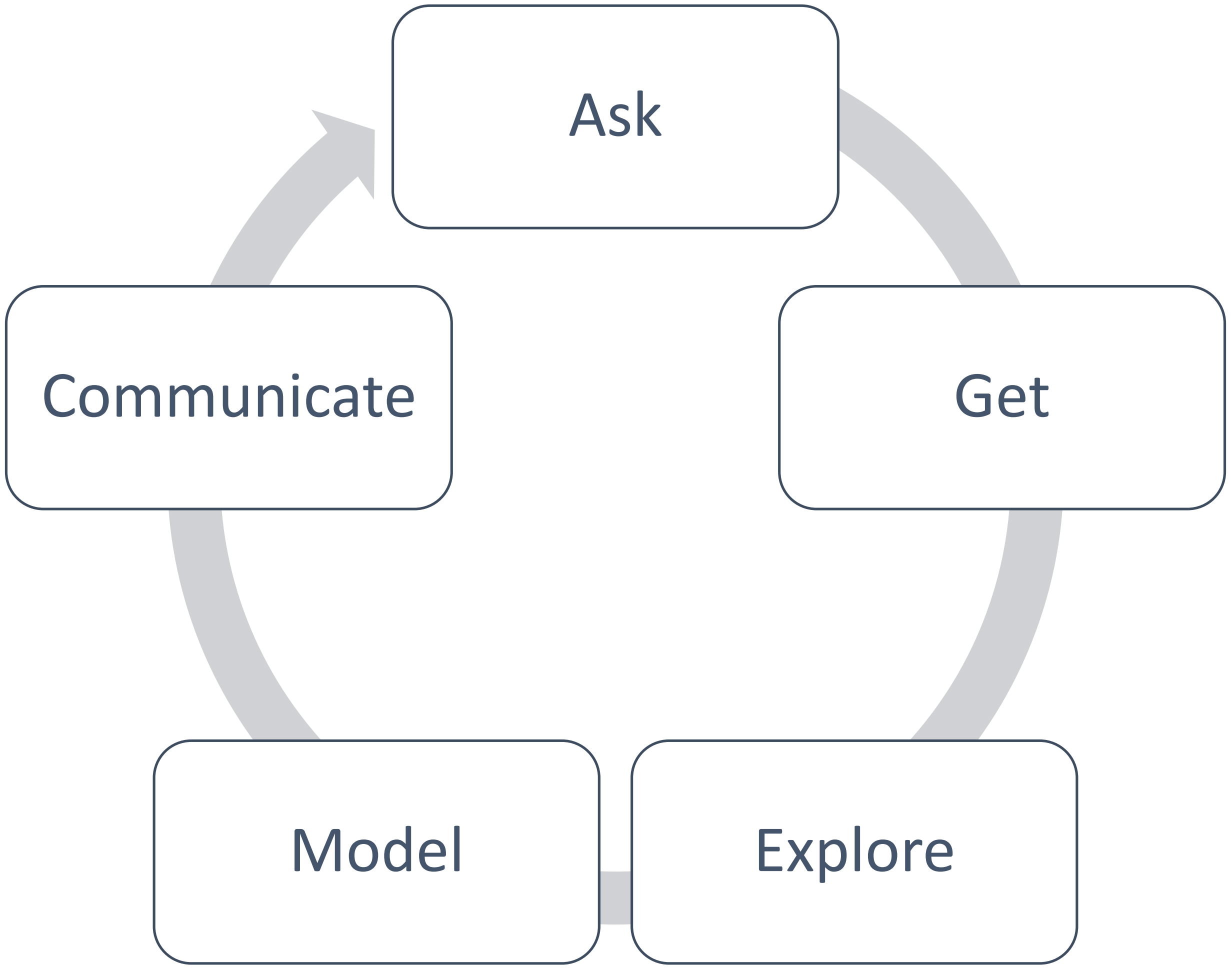
- Machine Learning
- Vision (7 APIs)
- Speech (4 APIs)
- Language (6 APIs)
- Knowledge (6 APIs)
- Search (7 APIs)
- Labs (6 APIs)

Where are we?

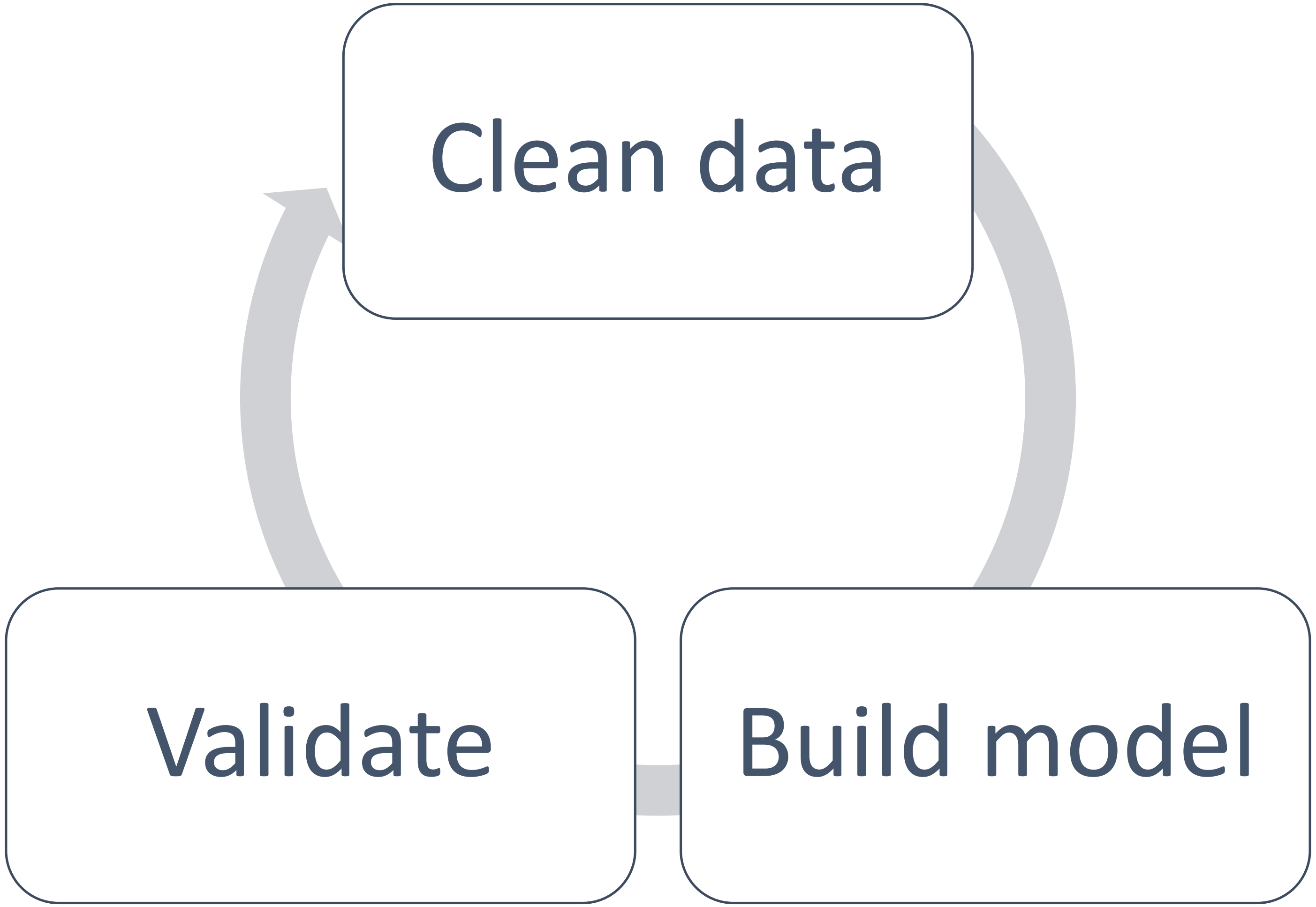
By 2020, predictive & prescriptive analytics will attract 40% of enterprises' net new investment.

100 Data and Analytics Predictions Through 2020, Gartner

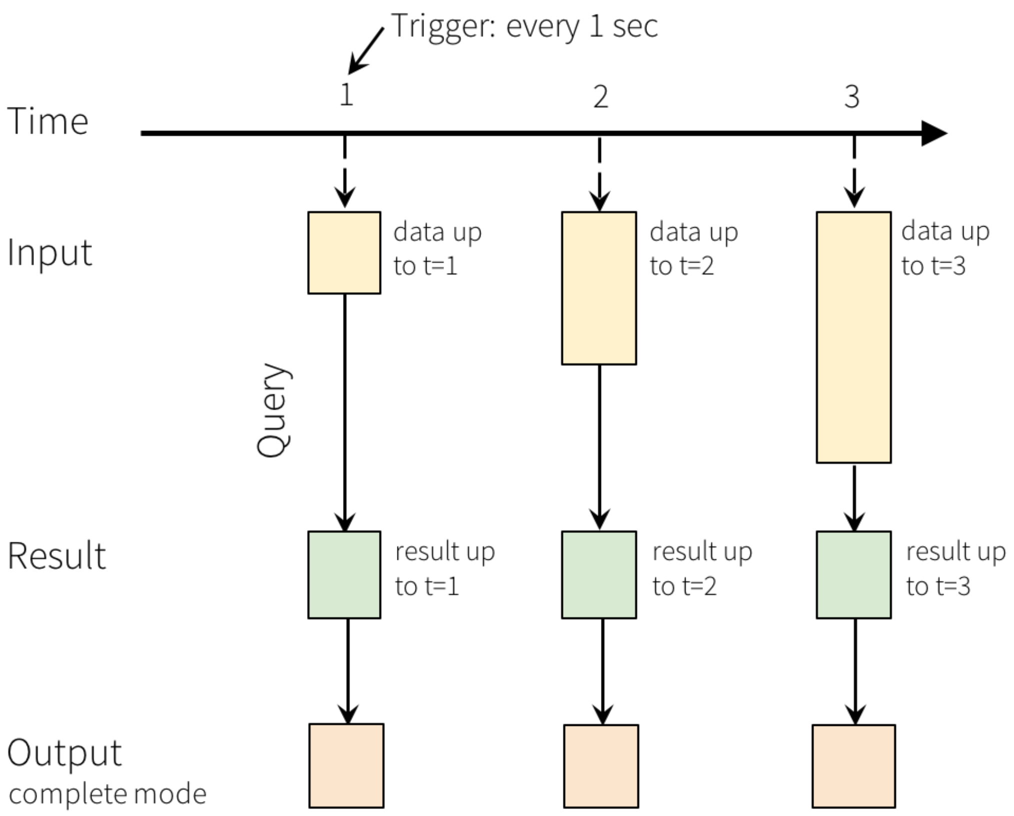
Data Science process



How Machine Learning usually works?



Streaming is not easy



Programming Model for Structured Streaming

Source: https://spark.apache.org/docs/latest/structured-streaming-programming-guide.html#basic-concepts

~~Hello World~~ K-means clustering

Algorithm 1 Mini-batch k-Means.

Given: k, mini-batch size b, iterations t, data set X
Initialize each $\mathbf{c} \in C$ with an $\mathbf{x}$ picked randomly from X
$\mathbf{v} \leftarrow 0$
for $i = 1$ to t **do**
  $M \leftarrow b$ examples picked randomly from X
  for $\mathbf{x} \in M$ **do**
    $\mathbf{d}[\mathbf{x}] \leftarrow f(C, \mathbf{x})$ // Cache the center nearest to $\mathbf{x}$
  end for
  for $\mathbf{x} \in M$ **do**
    $\mathbf{c} \leftarrow \mathbf{d}[\mathbf{x}]$ // Get cached center for this $\mathbf{x}$
    $\mathbf{v}[\mathbf{c}] \leftarrow \mathbf{v}[\mathbf{c}] + 1$ // Update per-center counts
    $\eta \leftarrow \frac{1}{\mathbf{v}[\mathbf{c}]}$ // Get per-center learning rate
    $\mathbf{c} \leftarrow (1 - \eta)\mathbf{c} + \eta\mathbf{x}$ // Take gradient step
  end for
end for

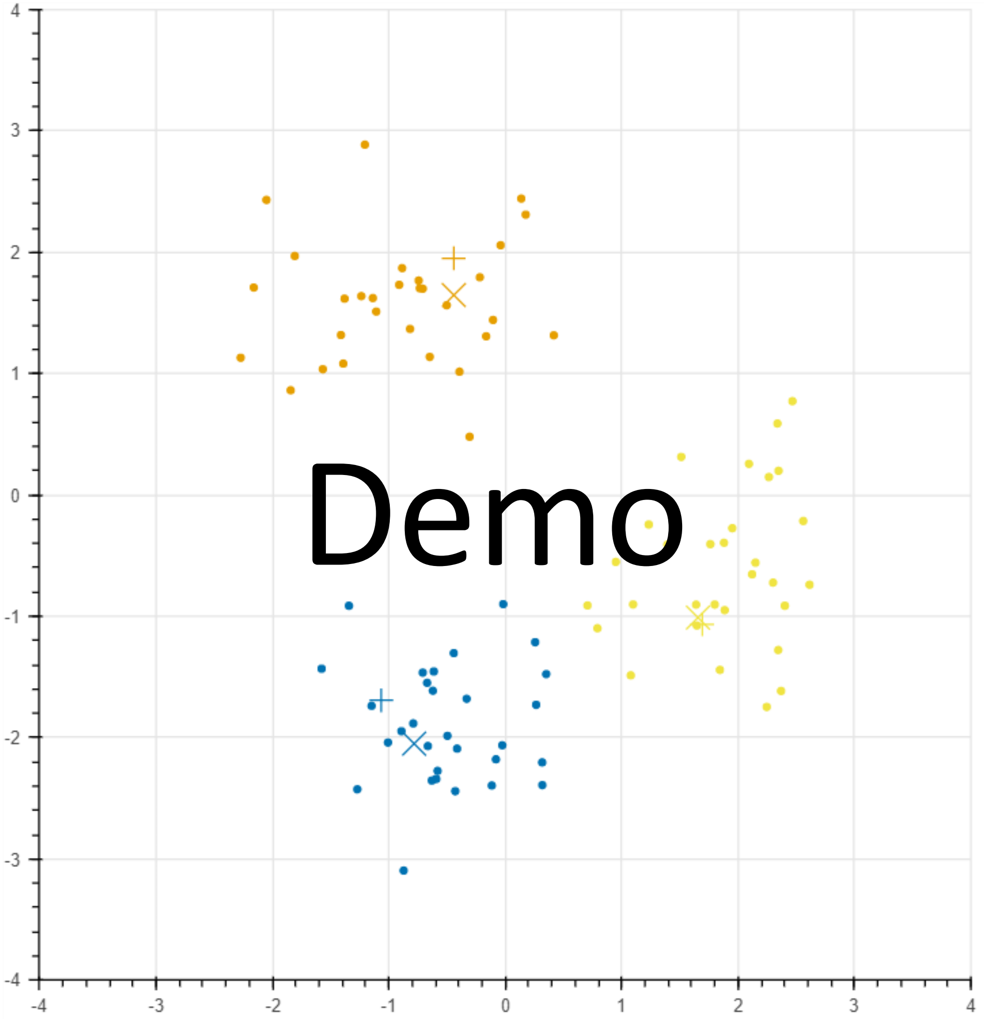
$$\eta \sim \frac{1}{n}$$

$$\eta \leftarrow const.$$

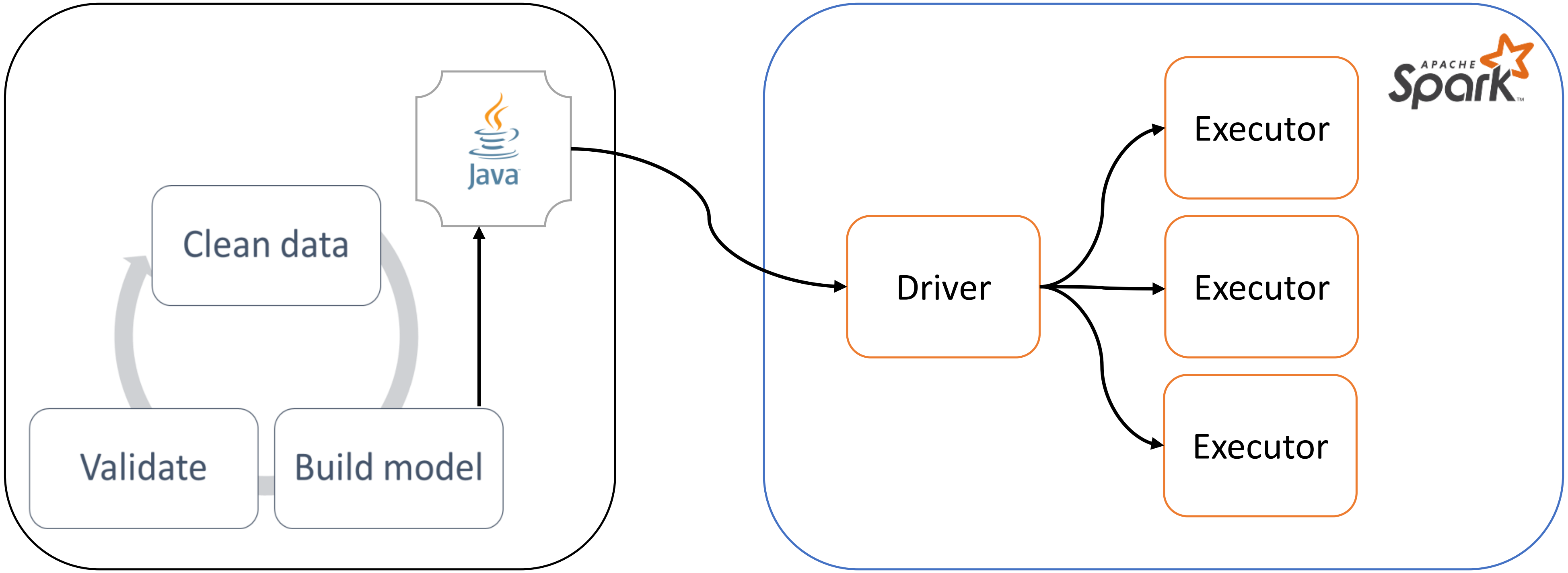
$$\eta \sim \alpha$$

Source: D. Sculley, *Web-Scale K-Means Clustering*, Google Inc. Pittsburgh. PA USA, 2010

How it works?



Trained models in streaming pipeline



Trained models in streaming pipeline

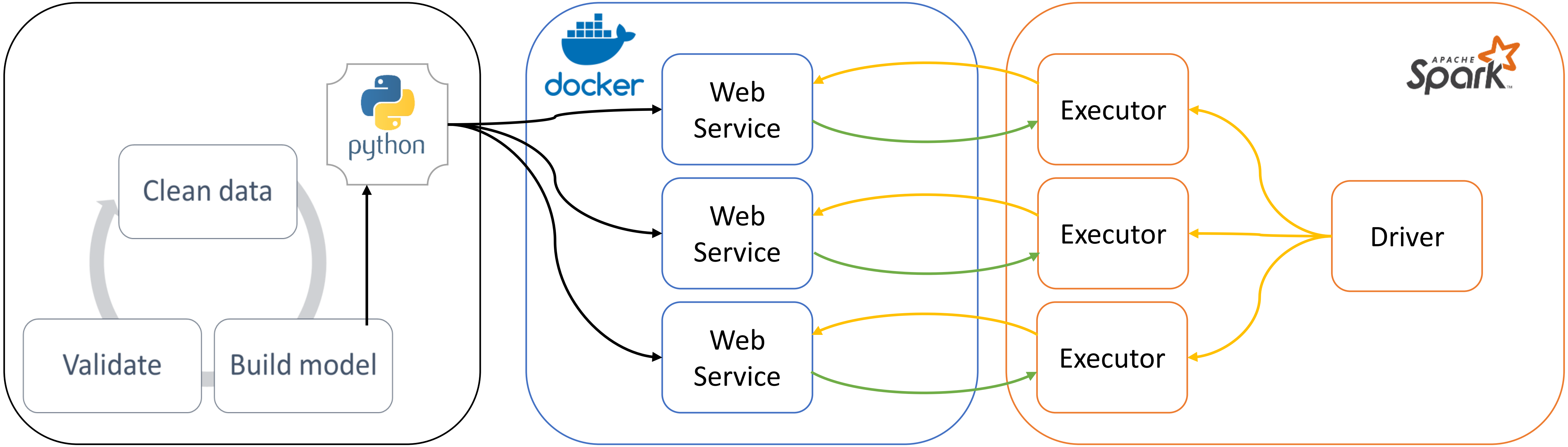
Pros

- Can develop models as usual, with proper validation etc.
- Models are part of processing pipeline, aka can't go any faster
- Models can be distributed efficiently
- Scale with the system

Cons

- Harder to update with a new model
- Programming languages should play well together
- Serialization can be problematic
- Models should be thread-safe
- Models often are passed down to engineers for productization

Containers et al.



Containers et al.

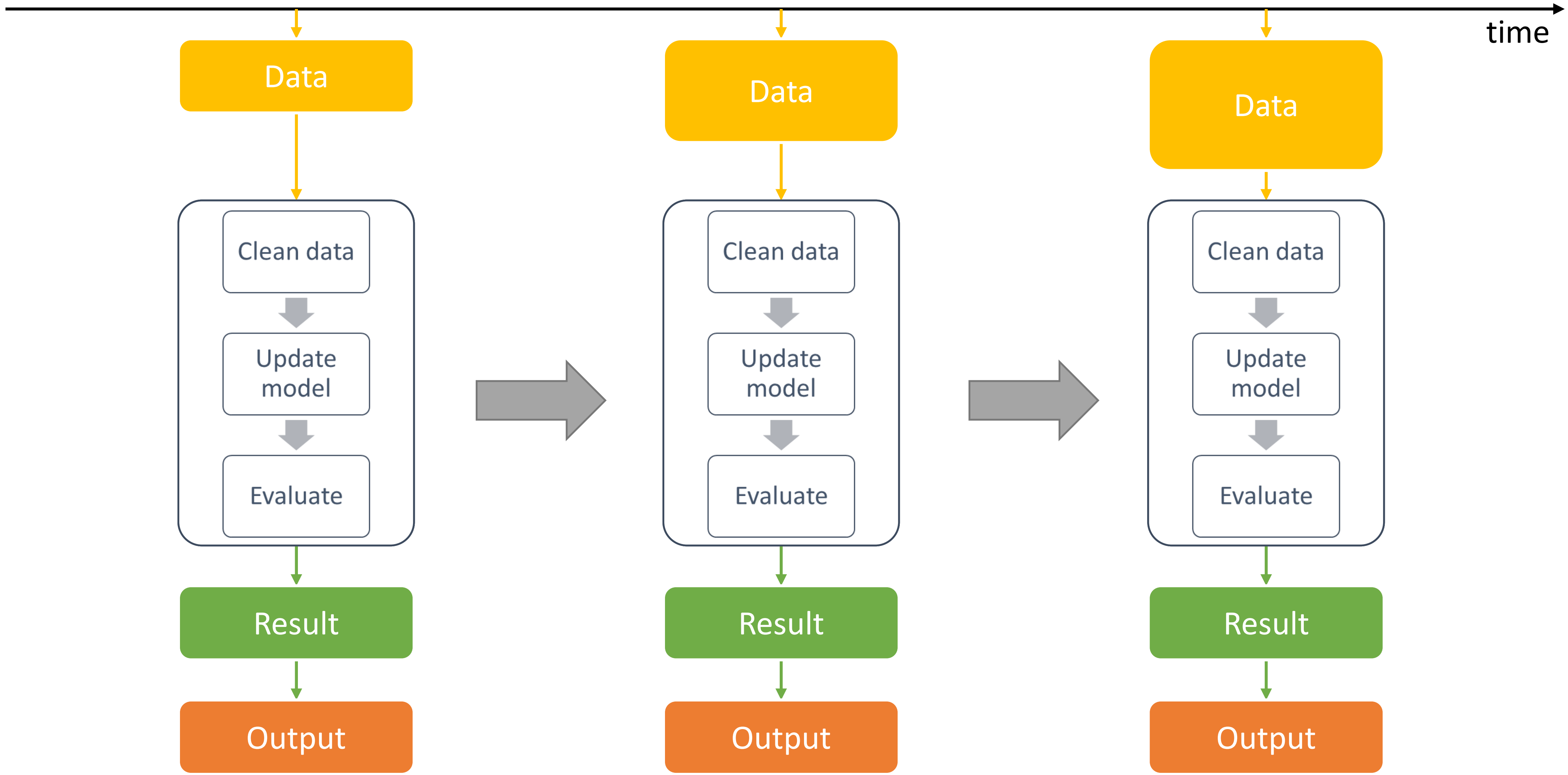
Pros

- Can develop models as usual, with proper validation etc.
- Can be done in almost any tools of choice
- Package and deployment can be done by model's creator
- Easy to incorporate into CI/CD process
- Can be utilized by any application which can reach its interface
- Scaling by cloning
- Multiple versions can be run at the same time
- Easy to update

Cons

- Extra work for modelling team
- Containers and web services have to be done properly, e.g. more bug prone
- Extra service to maintain
- May become a bottleneck
- Prone to network issues
- Require DevOps culture
- … (all other microservices' issues)

Online machine learning

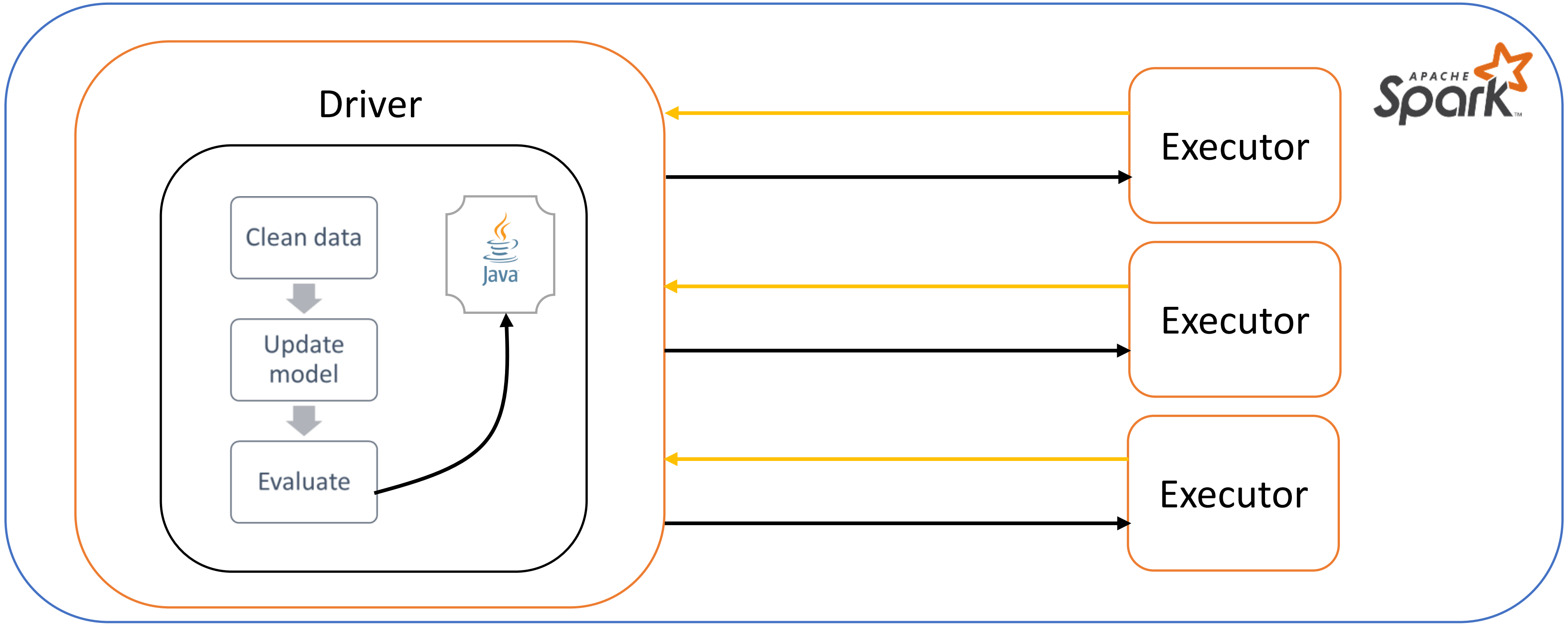


Models for online machine learning

- K-means (obviously)
- Generalized Linear Models
- Support Vector Machines
- Adaptive Boosting
- Neural-networks, including Deep Learning
- (anything that can be learned iteratively)

Source: https://www.coursera.org/learn/machine-learning

The state!


Driver
Clean data
Update model
Evaluate
Java
Executor
Executor
Executor
APACHE Spark

Online machine learning

Pros

- Adaptive
- Part of the streaming pipeline
- Update automatically
- Some models can be quite quick
- Instant results (compared to normal Data Science process)

Cons

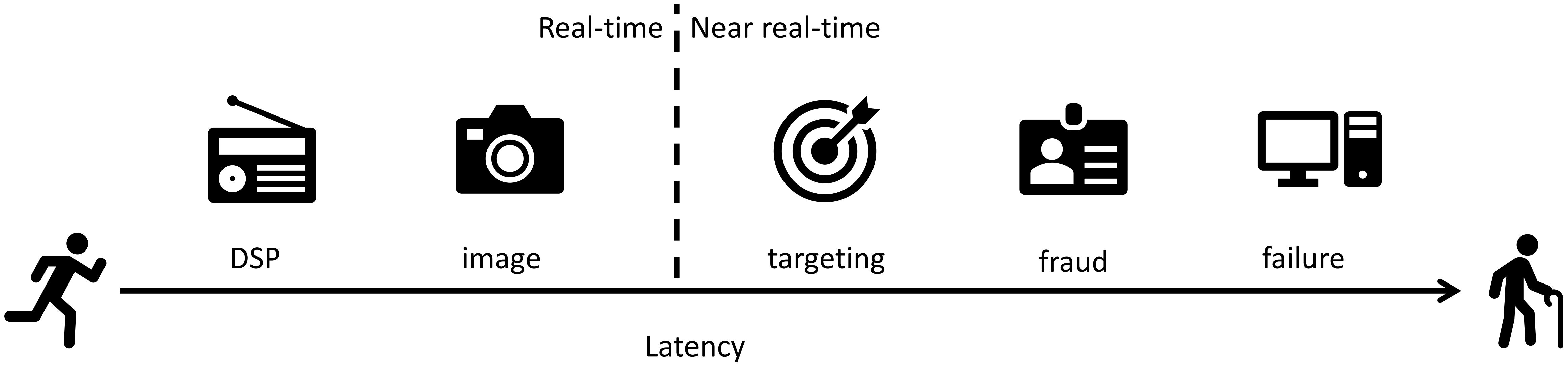
- Very little implemented models
- Much rely on the internals of the streaming system
- Tricky to implement
- Often require form of global state
- Validation only via monitoring
- Rely heavily on initial assumptions
- Training may be a bottleneck

What real-time means?

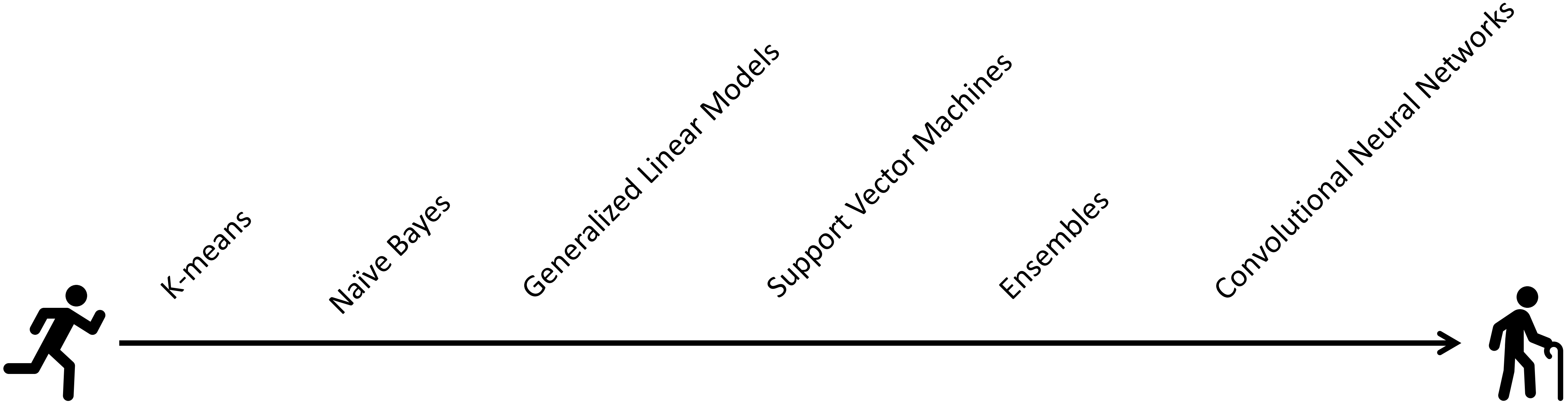
Real-time systems is hardware or software systems subject to a time constraint.

Source: https://en.wikipedia.org/wiki/Real-time_computing

Real-time time scales



Models vs time



Thank you!

Questions?

Codes available at GitHub:

https://github.com/jsnowacki/streaming-ml-talk