KDD 2024 – DELTA Workshop

On the Impact of Industrial Delays when Mitigating Distribution Drifts: an Empirical Study on Real-world Financial Systems

Thibault Simonetto[1], Maxime Cordy[1], Salah Ghamizi[2], Yves Le Traon[1],

Clément Lefebvre[3], Andrey Boystov[3], and Anne Goujon[3]

[1] University of Luxembourg, Luxembourg

[2] Luxembourg Institute of Science and Technology, Luxembourg

[3] BGL BNP Paribas, Luxembourg

Context

Transaction system to detect and quarantine suspicious transaction

Transaction 1
Transaction 2
Transaction …
Transaction n

Client history

+

T= Transaction n+1

Incoming transaction

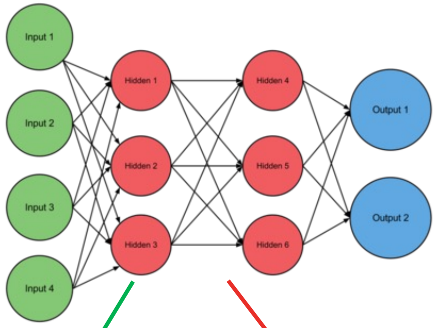
→

Feature vector [f1, f2, f3, f4, f5]

Feature vector

→

ML Model

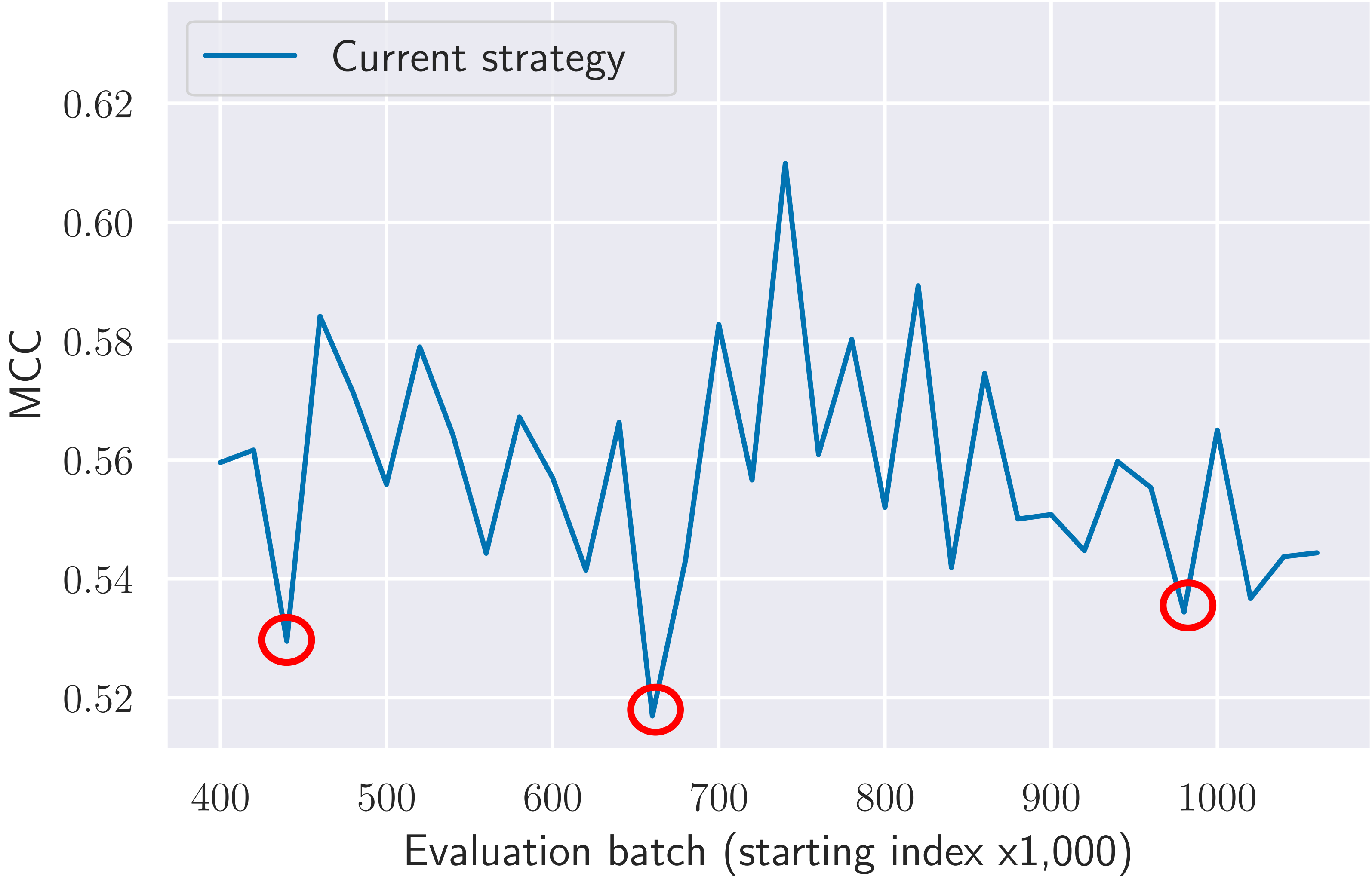


Accept

Reject

Correctly classifies up to 80% of the transactions

Problem: Performance drift

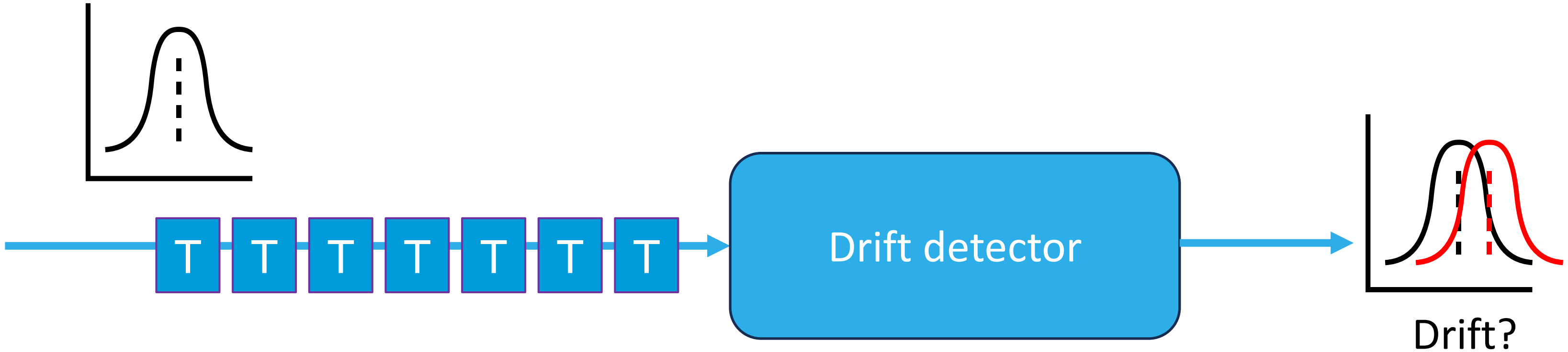


Time

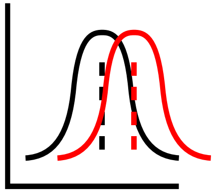
Drift detectors

Distribution

Sample stream

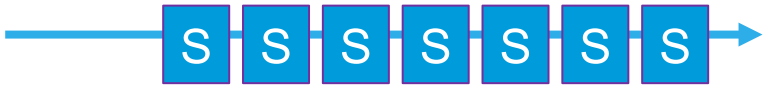


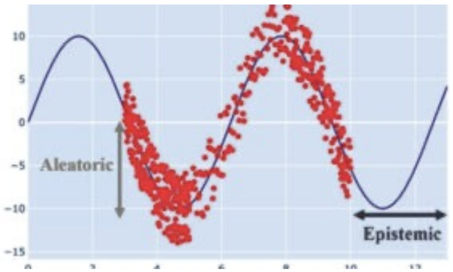
Drift?



Training data

Testing data





Statistical test (p-value)

Distance metric (threshold)

Data-based detection

Based on score function

Requires the label

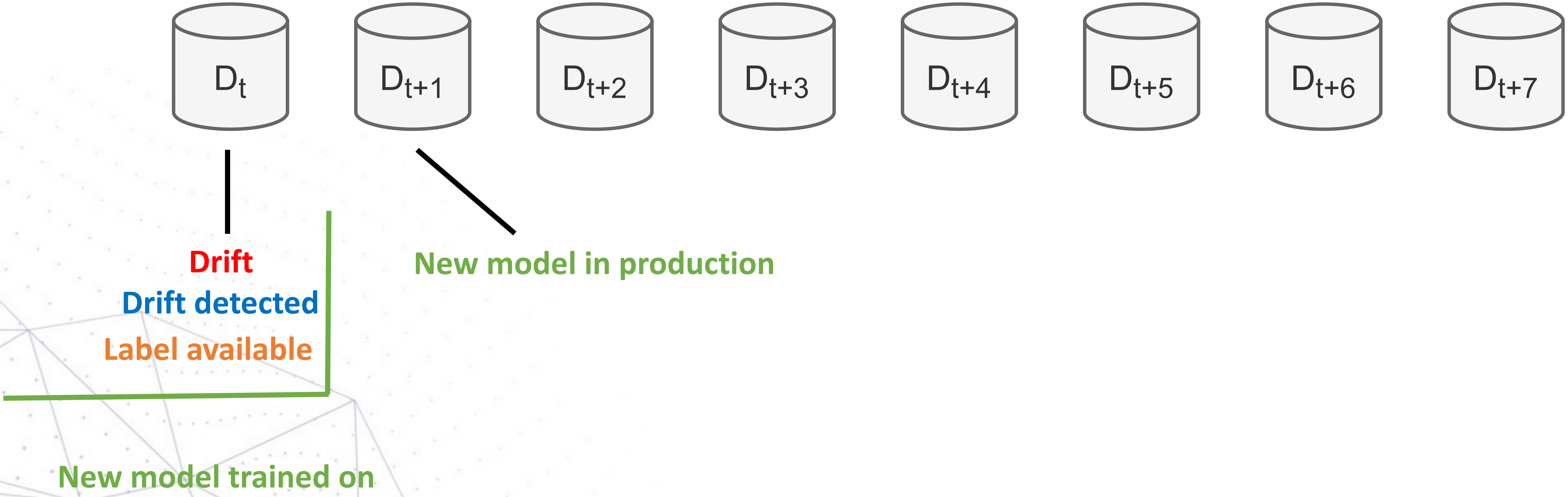
Error-based detection

Based on model behaviour for

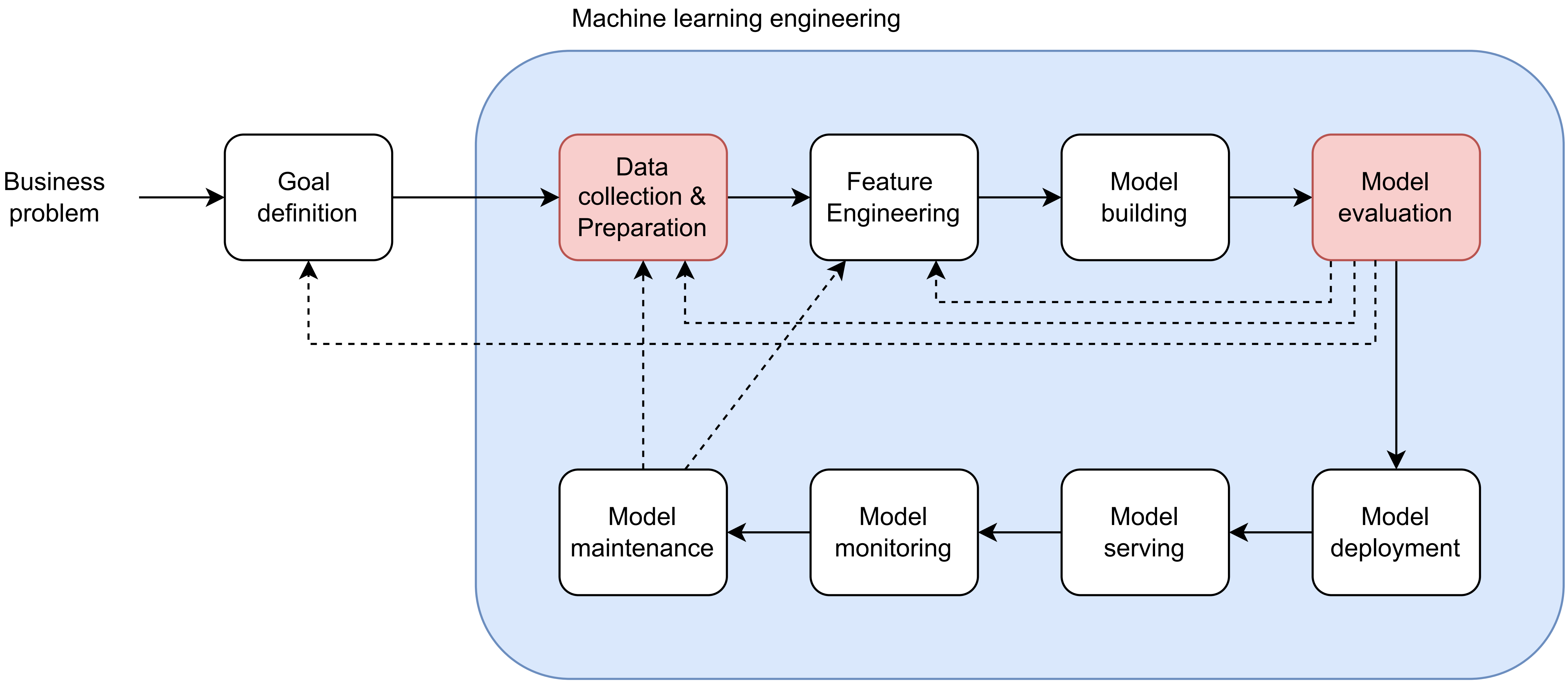
Given samples without labels

Predictive-based detection

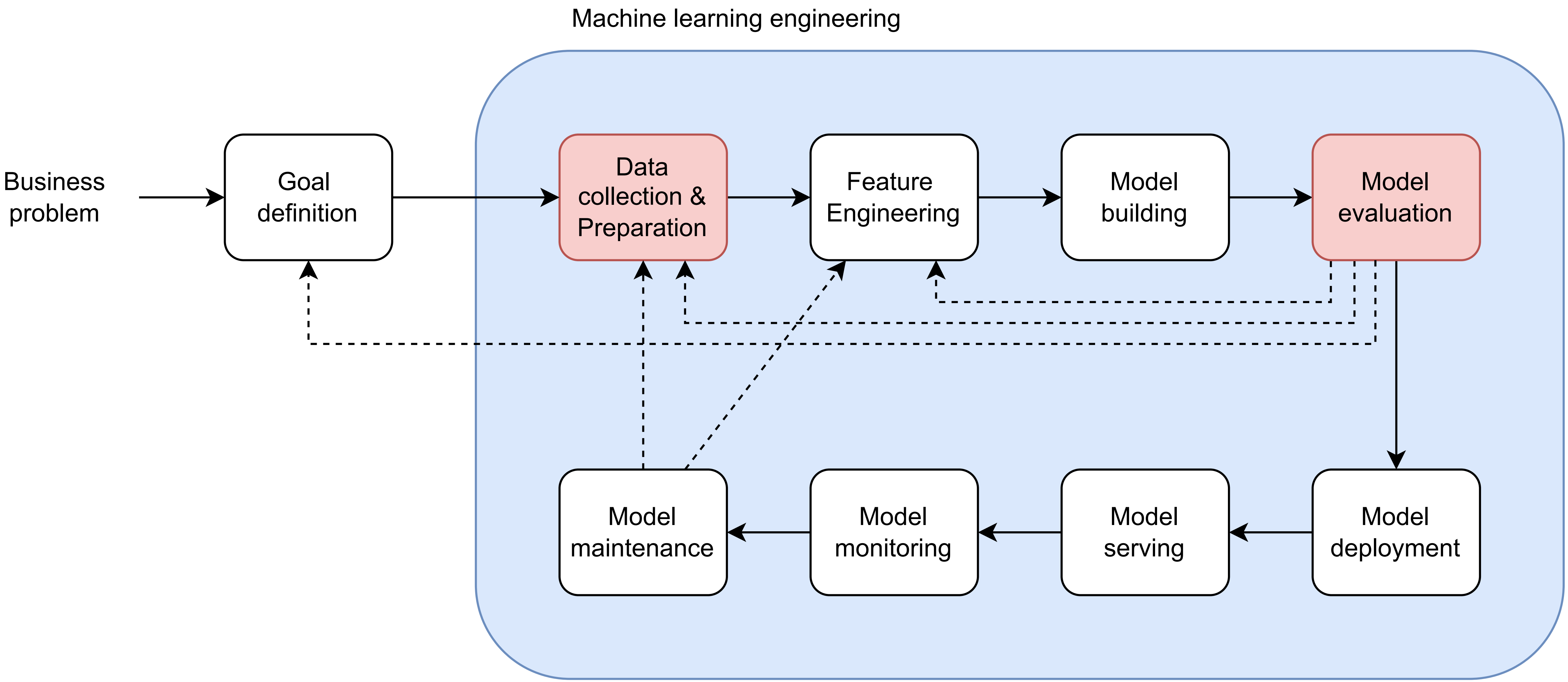
Ideal scenario



ML model lifecycle



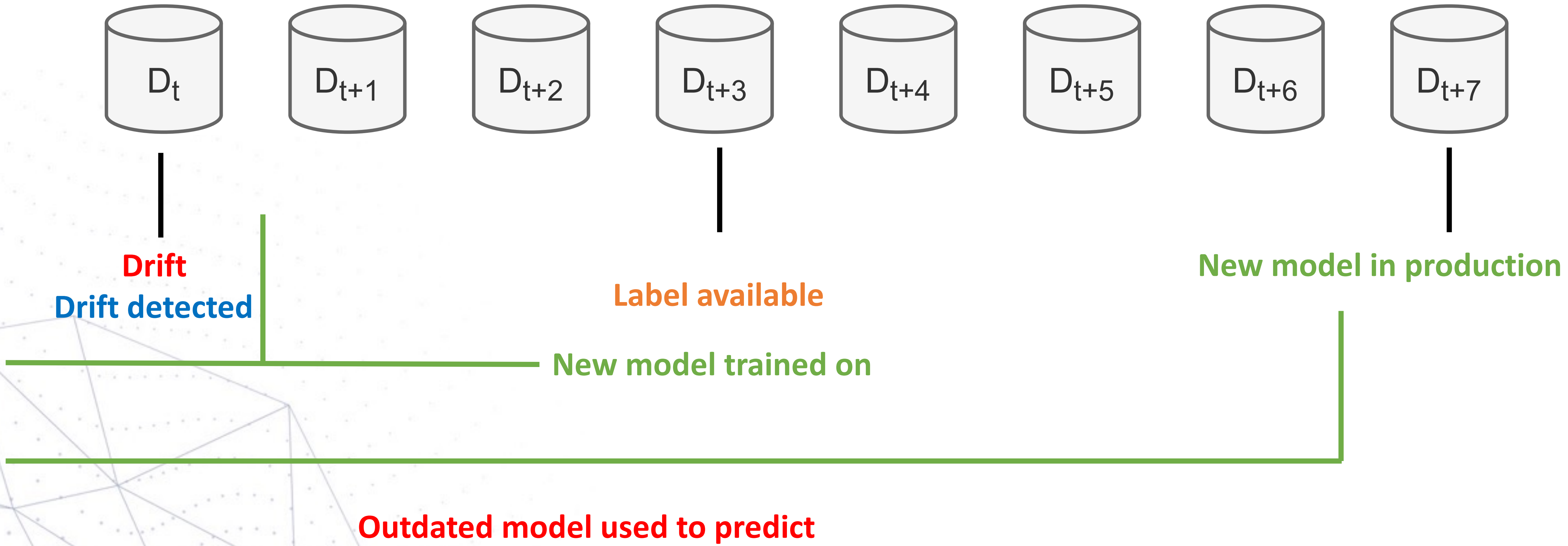
ML model lifecycle



Delays in data collection (labels) and model evaluation

Cost of model evaluation

Real-world Systems



Real-world Systems

δ_d Labelling delay δ_l Deployment delay

Problem

δ_d Labelling delay δ_l Deployment delay

$\mathrm{h}_{t_j} : \mathcal{X} \rightarrow \mathcal{Y}$ Classification model trained a time t_j Using data $\{\mathrm{x}_i, y_i\}$ such that $t_i + \delta_l \leq t_j$

h_{t_j} can only make prediction on $\{\mathrm{x}_k\}$ such that $t_j + \delta_d \leq t_k$

Problem

δ_d Labelling delay δ_l Deployment delay

$\mathrm{h}_{t_j} : \mathcal{X} \rightarrow \mathcal{Y}$ Classification model trained a time t_j Using data $\{\mathrm{x}_i, y_i\}$ such that $t_i + \delta_l \leq t_j$

h_{t_j} can only make prediction on $\{\mathrm{x}_k\}$ such that $t_j + \delta_d \leq t_k$

$\mathrm{sched} = \{\mathrm{t}_1 \ldots t_n\}$ Retraining schedule determining the sequence of models $\mathrm{H} = \{\mathrm{h}_{t_1} \ldots h_{t_j} \ldots h_{t_n}\}$

Problem

δ_d Labelling delay δ_l Deployment delay

$\mathrm{h}_{t_j} : \mathcal{X} \rightarrow \mathcal{Y}$ Classification model trained a time t_j Using data $\{\mathrm{x}_i, y_i\}$ such that $t_i + \delta_l \leq t_j$

h_{t_j} can only make prediction on $\{\mathrm{x}_k\}$ such that $t_j + \delta_d \leq t_k$

$\mathrm{sched} = \{\mathrm{t}_1 \ldots t_n\}$ Retraining schedule determining the sequence of models $\mathrm{H} = \{\mathrm{h}_{t_1} \ldots h_{t_j} \ldots h_{t_n}\}$

X_i is classified as $\hat{y}_i = h_{t_*}(x_i)$ where $t_* = \max\{t_k \in sched \text{ s.t. } t_k + \delta_{prod} \leq t_i\}$

Problem

δ_d Labelling delay δ_l Deployment delay

$\mathrm{h}_{t_j} : \mathcal{X} \rightarrow \mathcal{Y}$ Classification model trained a time t_j Using data $\{\mathrm{x}_i, y_i\}$ such that $t_i + \delta_l \leq t_j$

h_{t_j} can only make prediction on $\{\mathrm{x}_k\}$ such that $t_j + \delta_d \leq t_k$

$\mathrm{sched} = \{\mathrm{t}_1 \ldots t_n\}$ Retraining schedule determining the sequence of models $\mathrm{H} = \{\mathrm{h}_{t_1} \ldots h_{t_j} \ldots h_{t_n}\}$

X_i is classified as $\hat{y}_i = h_{t_*}(x_i)$ where $t_* = \max\{t_k \in sched \text{ s.t. } t_k + \delta_{prod} \leq t_i\}$

We evaluate the schedule $\mathrm{sched} = \{\mathrm{t}_1 \ldots t_n\}$ We evaluate the schedule of effectiveness and cost

$$s = score(Y, \hat{Y}) = MCC(Y, \hat{Y})$$

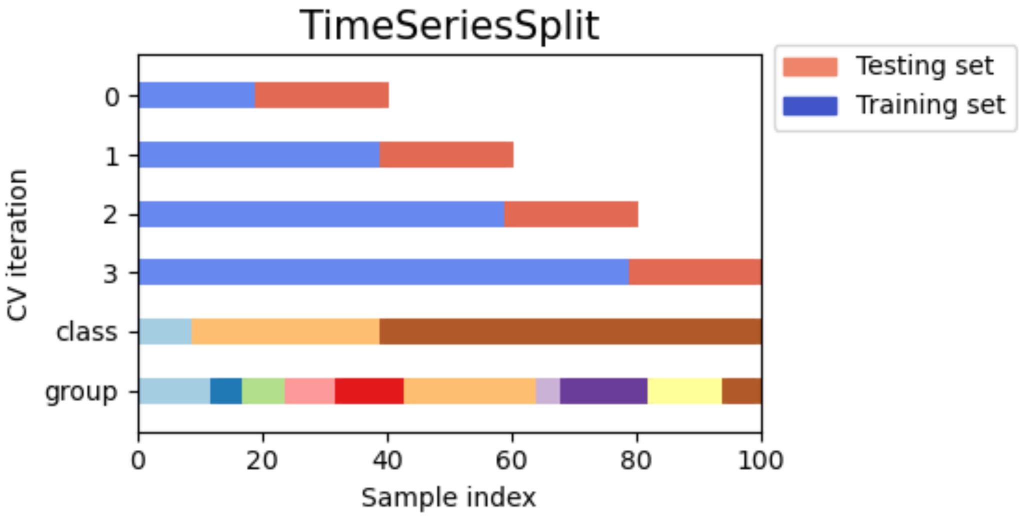
$$c = cost(sched) = |H|$$

Evaluating drift detectors

Train | Test

Train Drift Detectors Hyper-Pameters

25 Search iterations on with 5-fold timeseries validation



Figures from sklearn documentation:
https://scikit-learn.org/stable/auto_examples/model_selection/plot_cv_indices.html

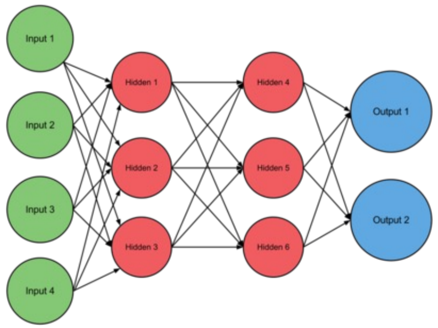
BGL BNP PARIBAS

Evaluating drift detectors

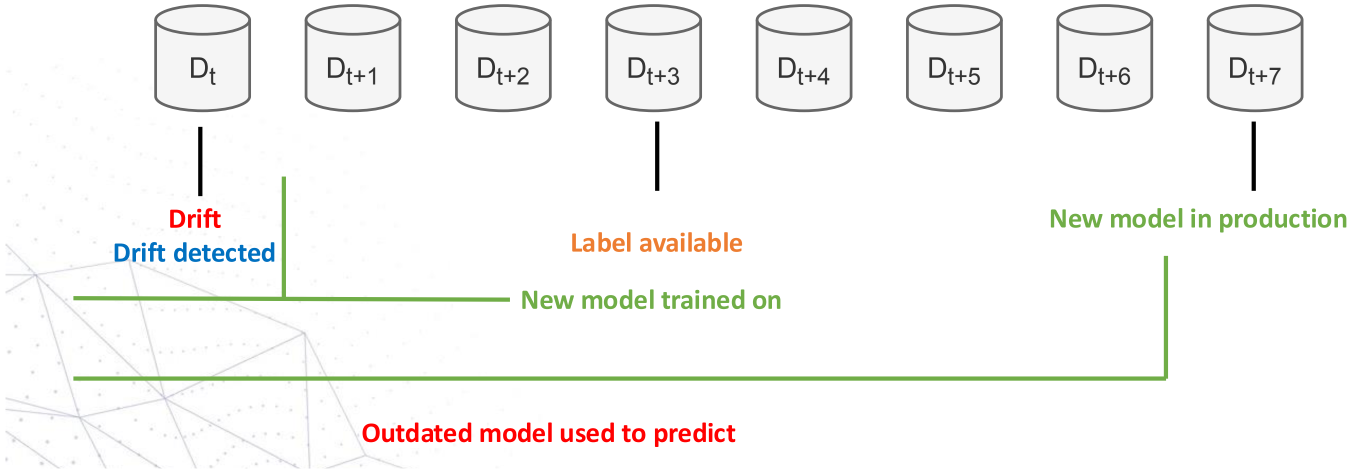
Split 0 to 5

Train	Test

Train model on the train dataset



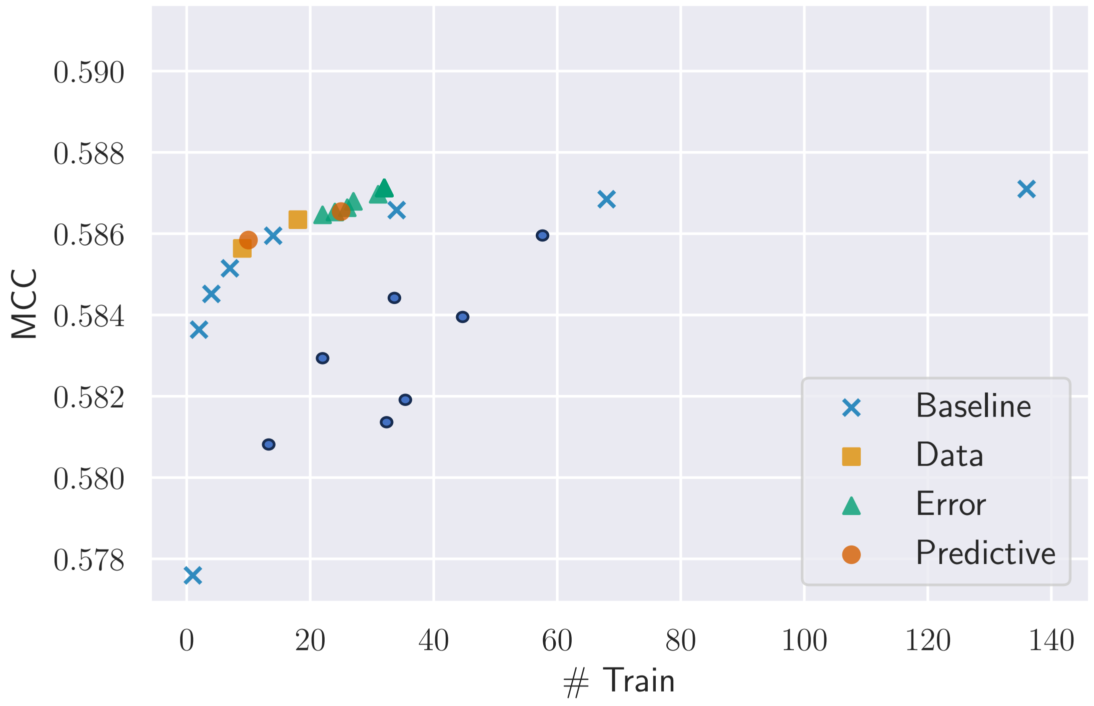
Evaluate a single drift detector and parameters with realistic delays



Evaluating drift detectors

Train	Test

Evaluate the solution using best training parameters for each detector

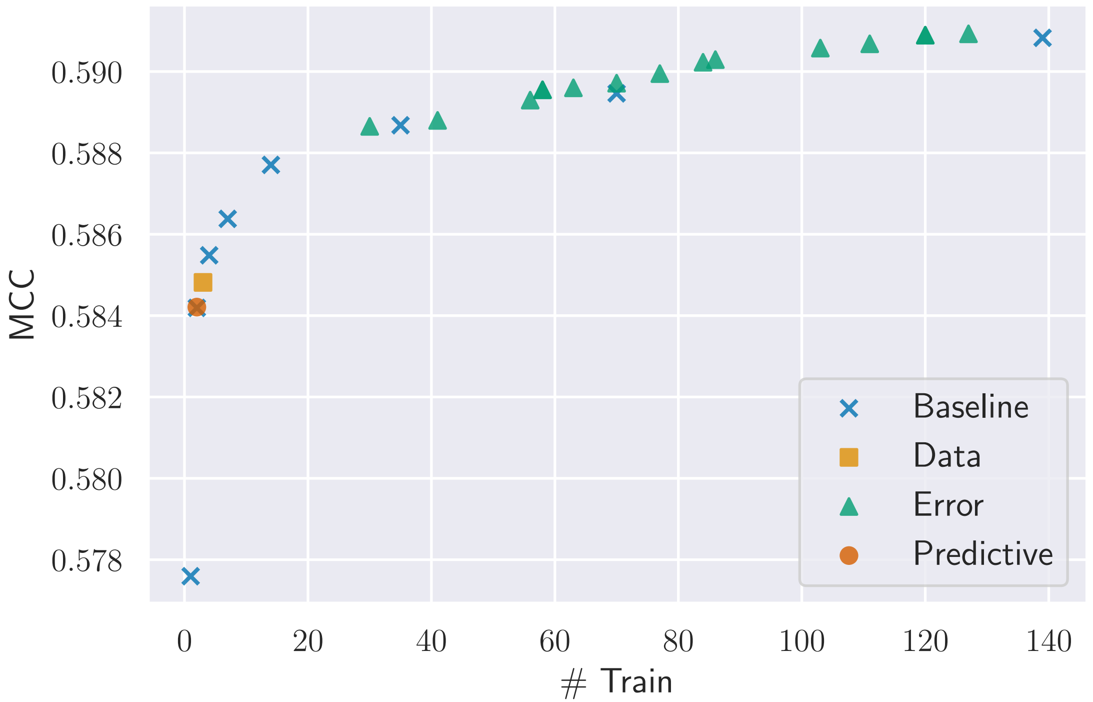


Empirical results on real-world financial system

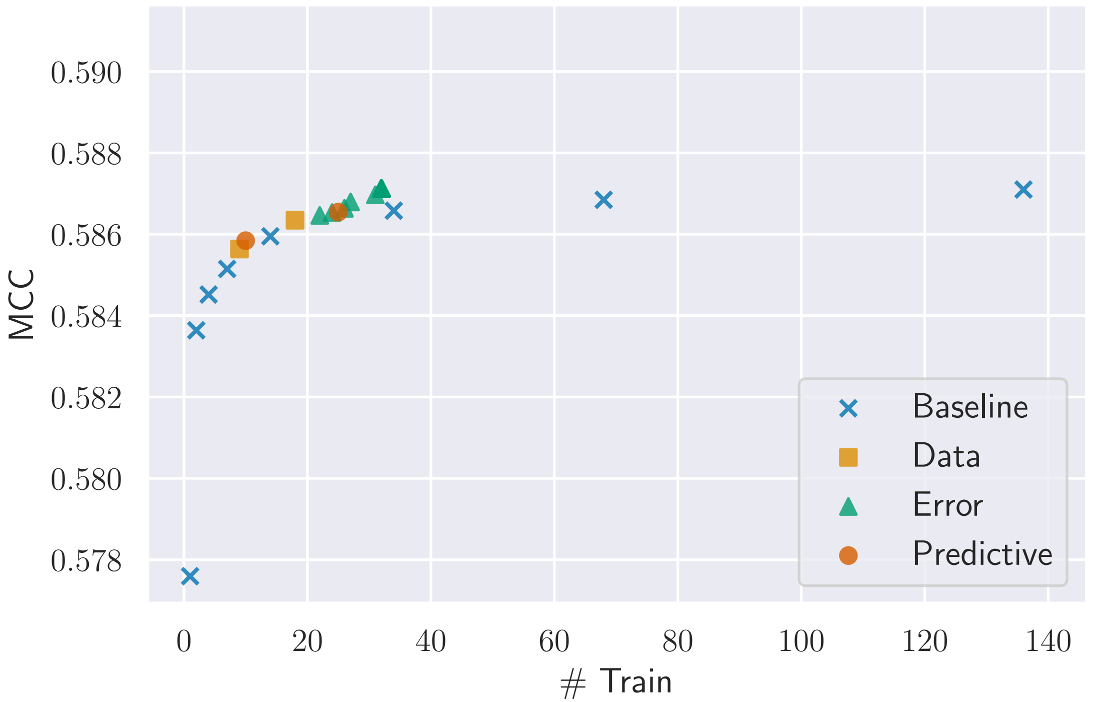
Train	Test
2 years of data	4 years of data

Type	Detector
Baseline	No detection Periodic
Data-based detector	Statistical test Divergence PC -CD
Error-based detector	DWIN (CE) DWIN (PE) DDM EDDM HDDM- HDDM-W KSWIN (CE) KSWIN (PE) Page-Hinkley (CE) Page-Hinkley (PE)
Predictive-based detector	Uncertainty ries DWIN

Empirical results on real-world financial system



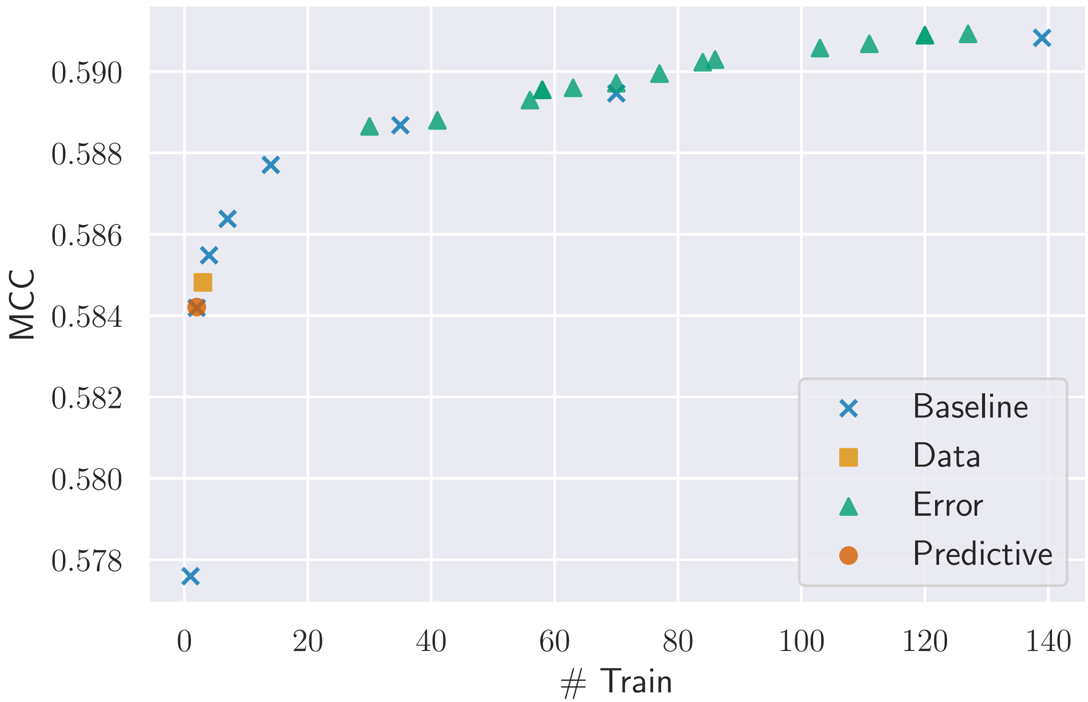
Without delays



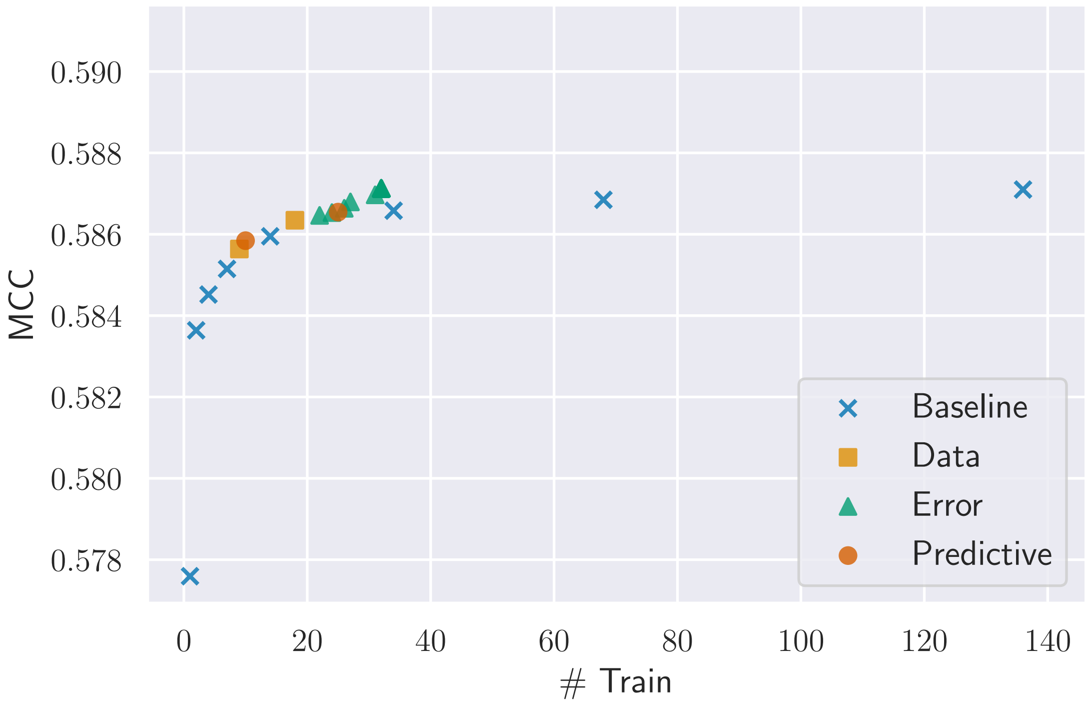
With delays

Label: 10days, Validation/Deployment: 4 weeks

Empirical results on real-world financial system



Without delays

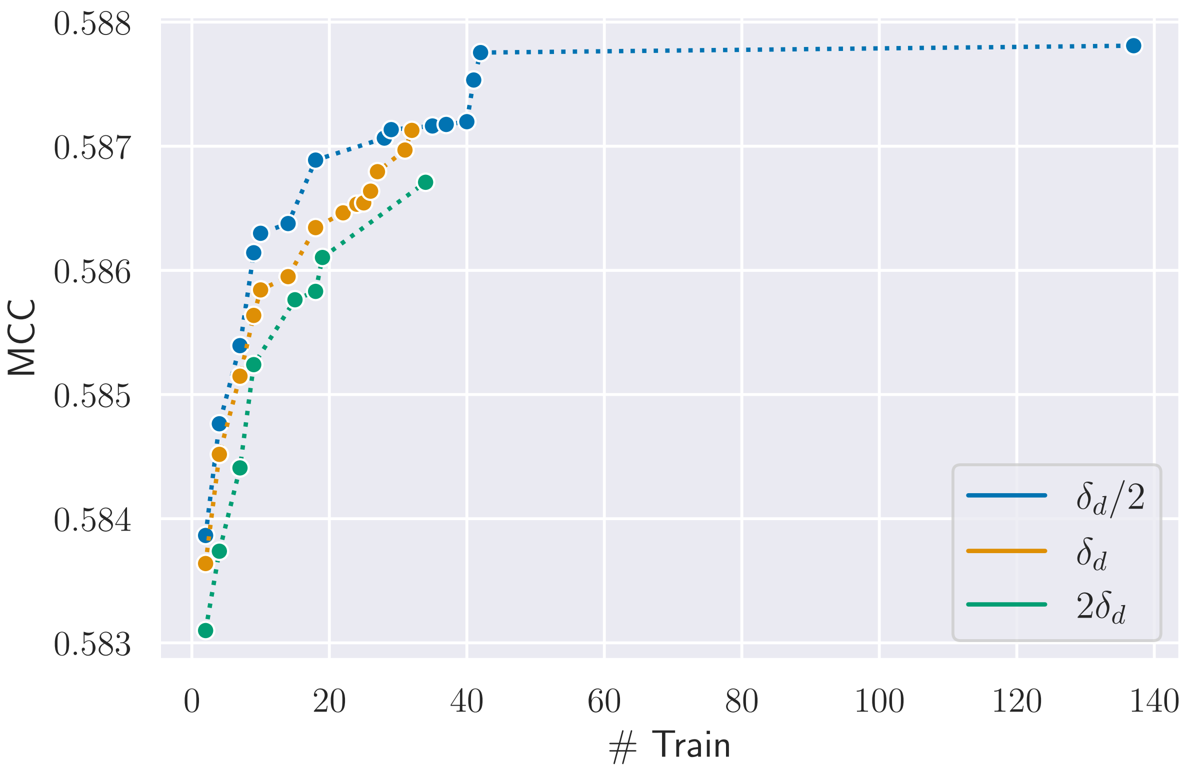


With delays

Label: 10days, Validation/Deployment: 4 weeks

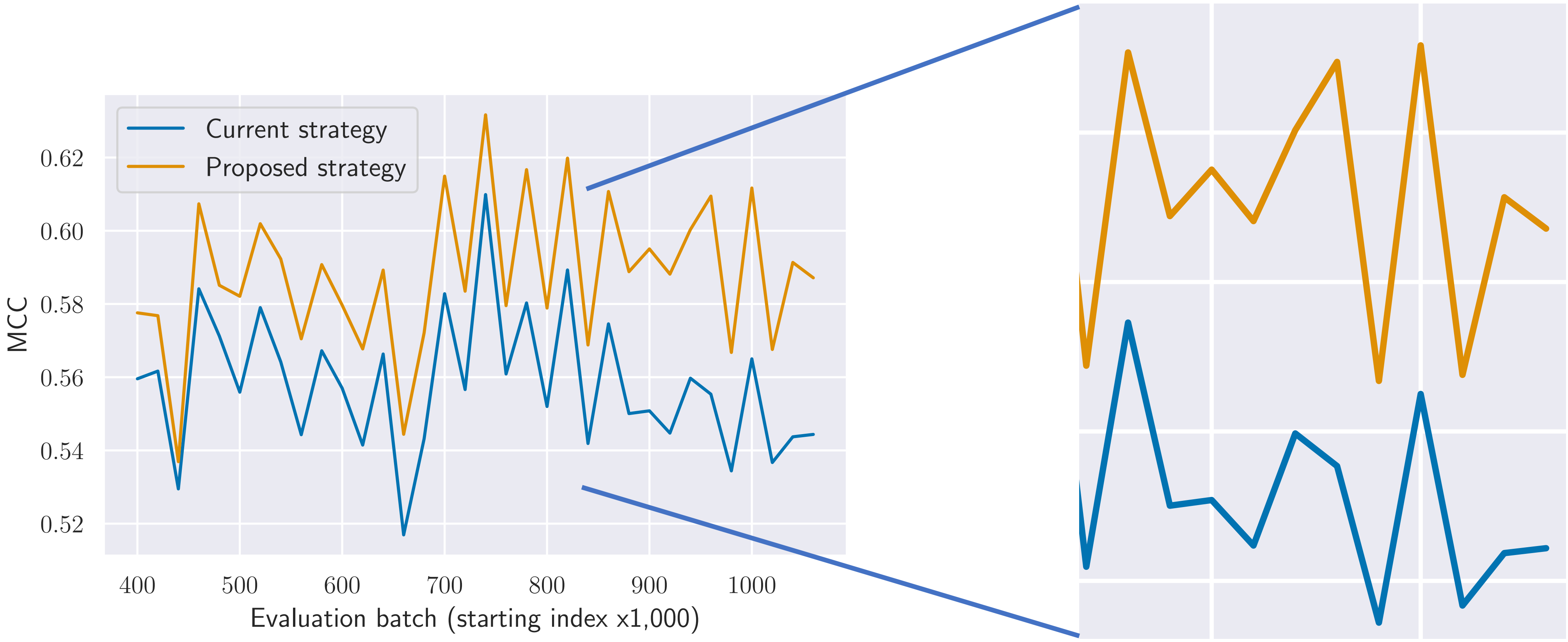
Not considering delay overestimates the effectiveness/efficiency trade-off of retraining schedules
Change in delay disrupts the ranking of drift detectors

Empirical results on real-world financial system



Change in delay has an inverse effect on efficiency/effectiveness.

Empirical results on real-world financial system



Our method help to mitigate performance drift

Conclusion

δ_d Labelling delay δ_l Deployment delay

$h_{t_j} : \mathcal{X} \rightarrow \mathcal{Y}$ Classification model trained a time t_j Using data $\{x_i, y_i\}$ such that $t_i + \delta_l \leq t_j$

h_{t_j} can only make prediction on $\{x_k\}$ such that $t_j + \delta_d \leq t_k$

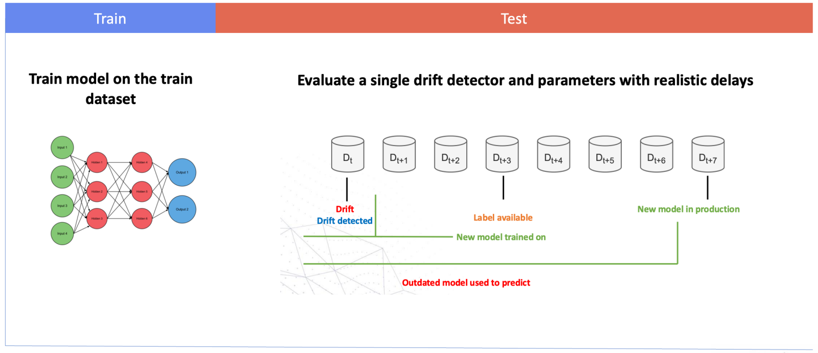
$\text{sched} = \{t_1 \ldots t_n\}$ Retraining schedule determining the sequence of models $H = \{h_{t_1} \ldots h_{t_j} \ldots h_{t_n}\}$

x_i is classified as $\hat{y}_i = h_{t_*}(x_i)$ where $t_* = \max\{t_k \in sched \text{ s.t. } t_k + \delta_{prod} \leq t_i\}$

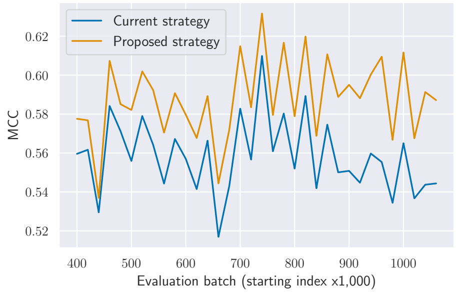
We evaluate the schedule $\text{sched} = \{t_1 \ldots t_n\}$ We evaluate the schedule of effectiveness and cost

$$s = score(Y, \hat{Y}) = MCC(Y, \hat{Y})$$
$$c = cost(sched) = |H|$$

Problem definition of delays in drift mitigation

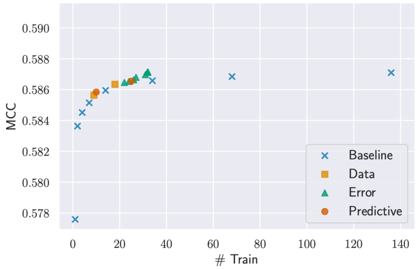


Evaluation protocol



BGL BNP PARIBAS

Empirical study on a real world financial system



With delays
Label: 10days, Validation/Deployment: 4 weeks

Change in delay disrupts the ranking of drift detectors

Conclusion

δ_d Labelling delay δ_l Deployment delay

$h_{t_j} : \mathcal{X} \to \mathcal{Y}$ Classification model trained a time t_j Using data $\{x_i, y_i\}$ such that $t_i + \delta_l \leq t_j$

h_{t_j} can only make prediction on $\{x_k\}$ such that $t_j + \delta_d \leq t_k$

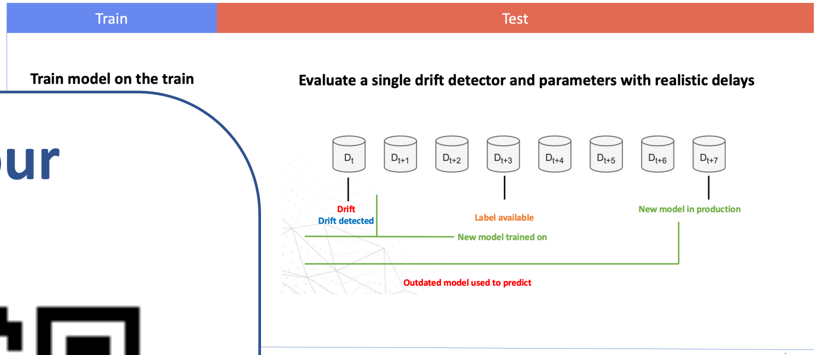
$sched = \{t_1 \ldots t_n\}$ Retraining schedule determining the sequence of models $H = \{h_{t_1}$

x_i is classified as $\hat{y}_i = h_{t_*}(x_i)$ where $t_* = \max\{t_k \in sched$ s.

We evaluate the schedule $sched = \{t_1 \ldots t_n\}$ We evaluate the schedule of effectiv

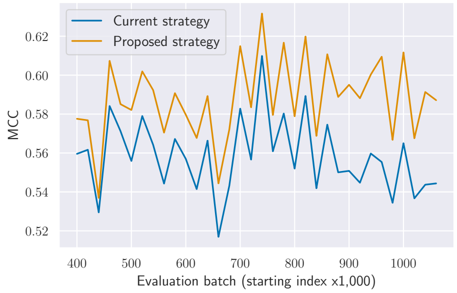
$$s = score(Y, \hat{Y}) = MCC(Y, \hat{Y})$$
$$c = cost(sched) = |H|$$

Problem definition of delay mitigation

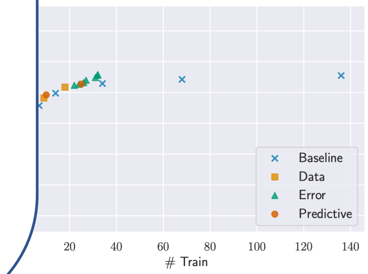


valuation protocol

Checkout our repo



Empirical study on a real world financial system



With delays
Label: 10days, Validation/Deployment: 4 weeks

Change in delay disrupts the ranking of drift detectors

BGL BNP PARIBAS