

# Reinforcement Learning for Data Cleaning and Data Preparation

**Laure Berti-Equille**

IRD, Aix Marseille Université

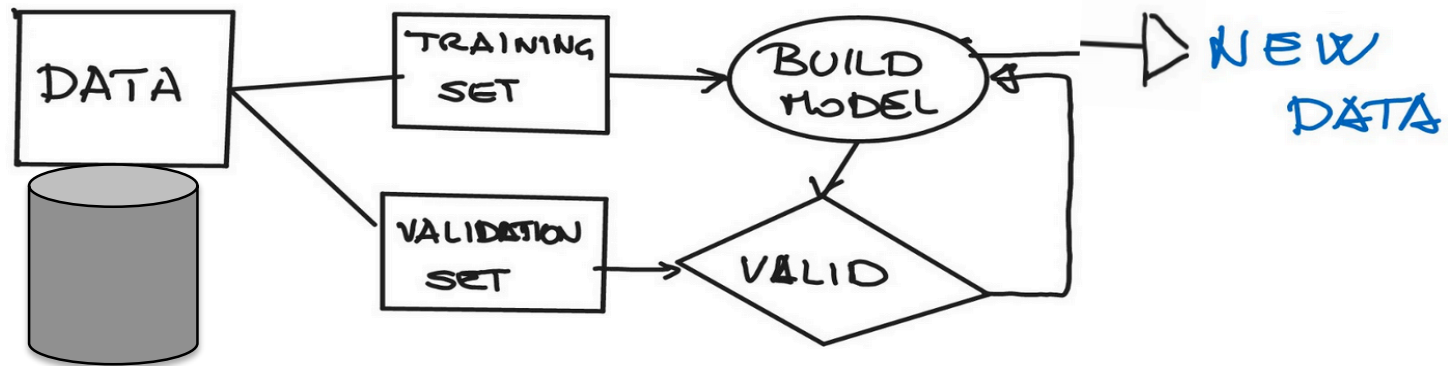
DIAMS, LIS, France

[laure.berthi@ird.fr](mailto:laure.berthi@ird.fr)

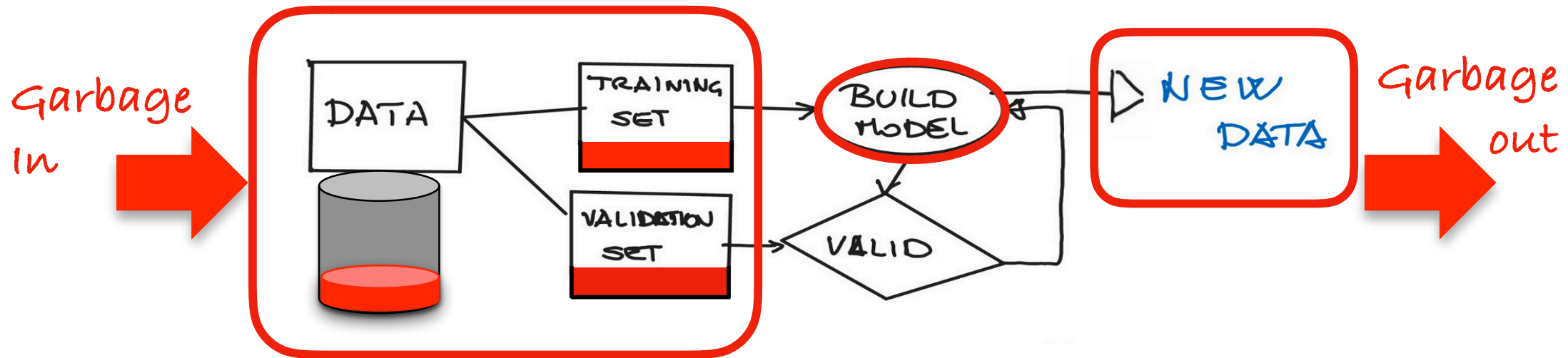
HILDA 2019 @ SIGMOD 2019



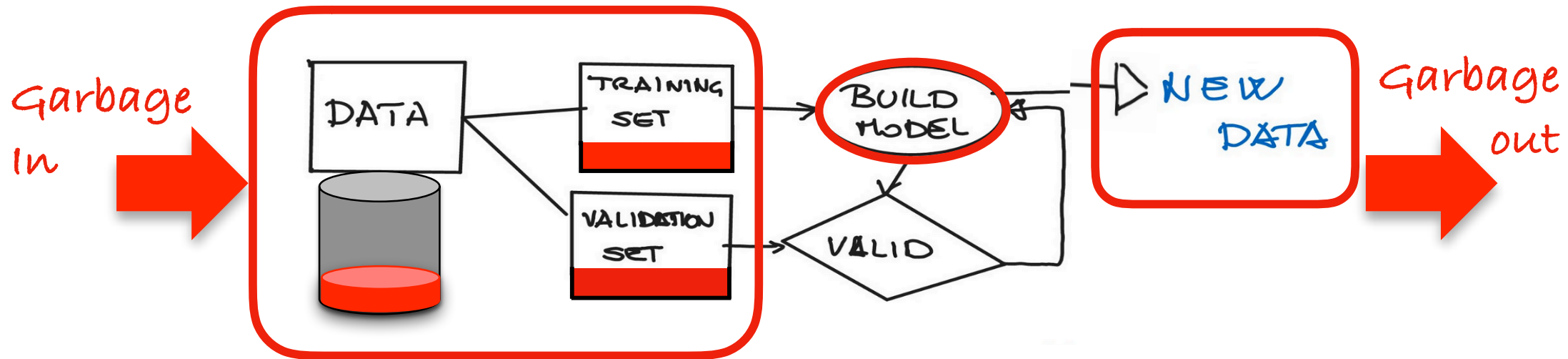
# Learning from dirty data is risky



# Learning from dirty data is risky

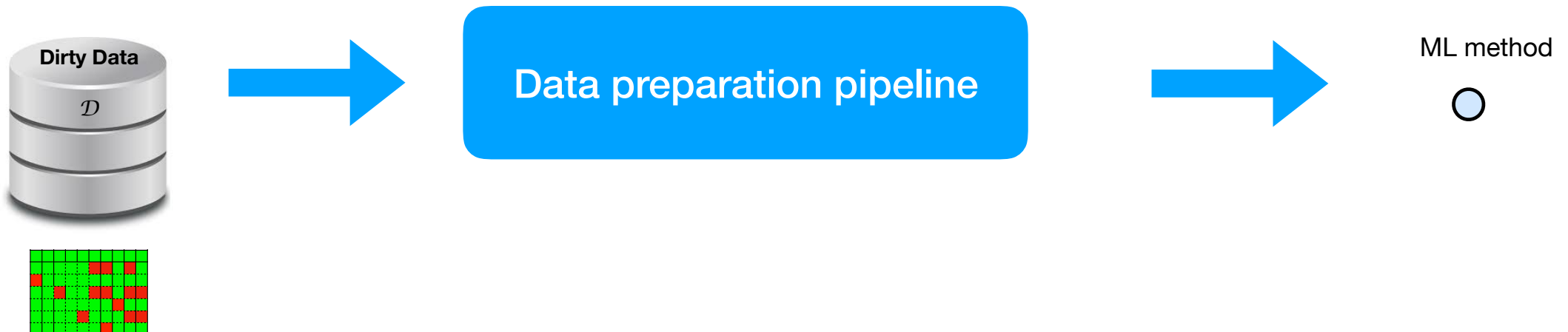


# Learning from dirty data is risky

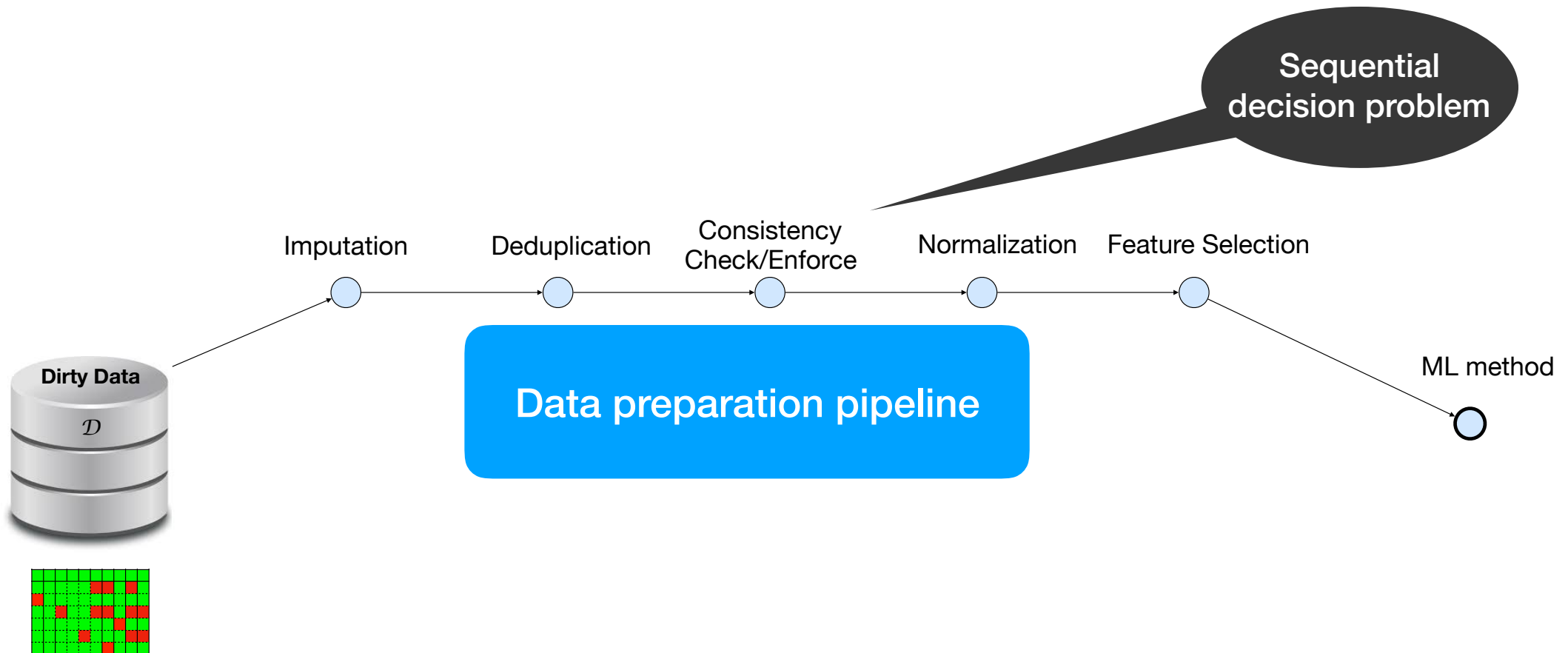


How to clean and prepare the data at their best ?

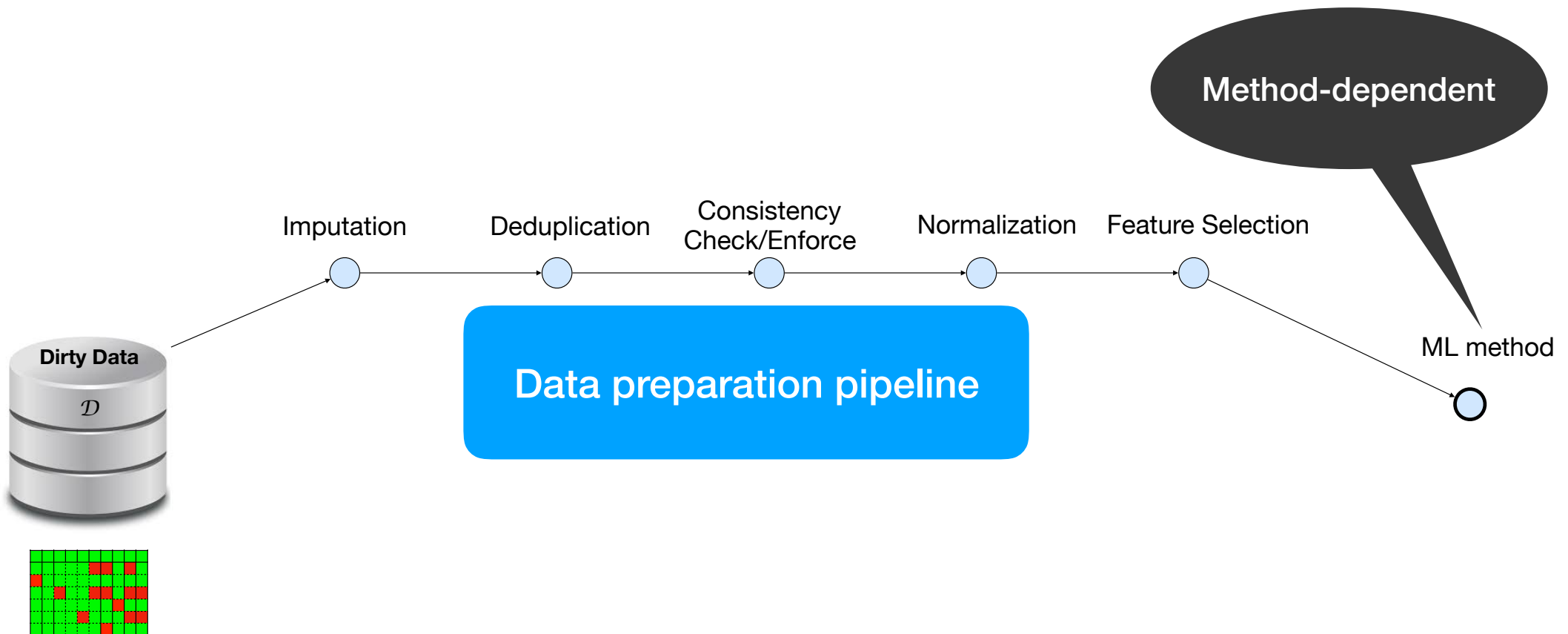
# Data preparation is challenging



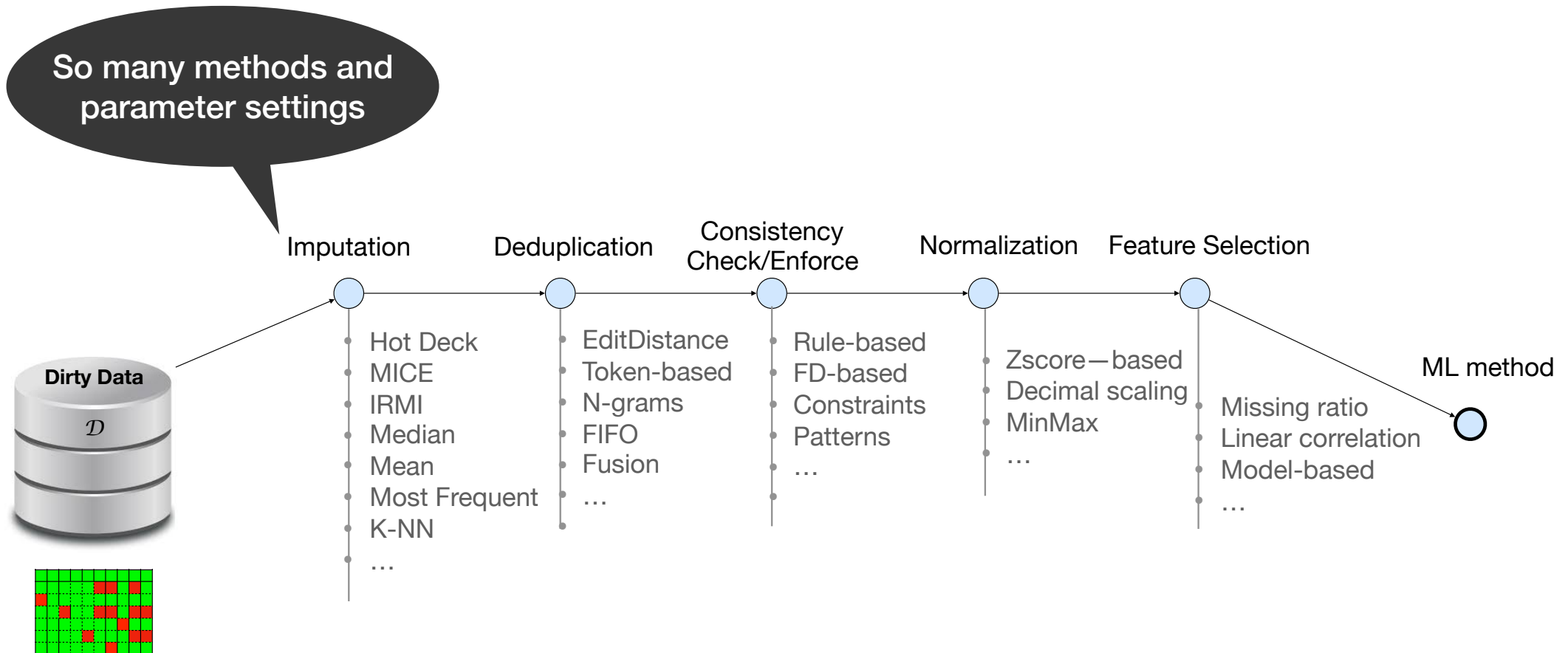
# Data preprocessing is challenging



# Data preprocessing is challenging

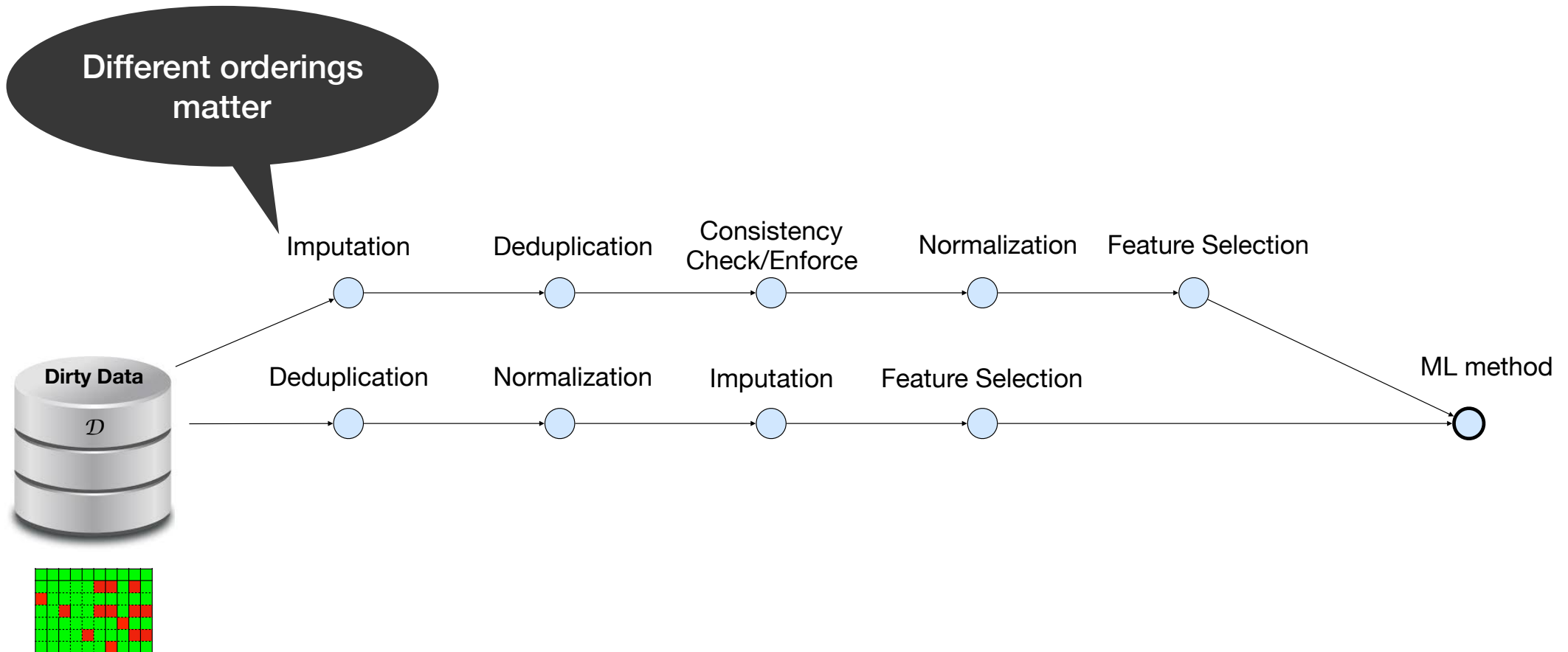


# Data preprocessing is challenging

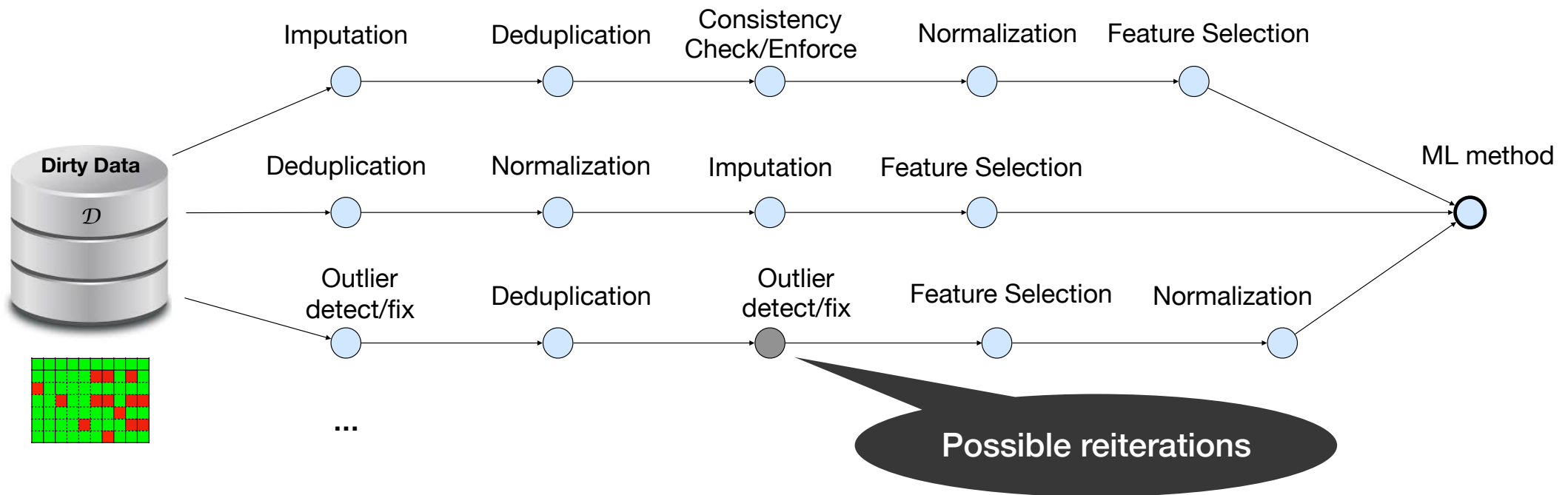




# Data preprocessing is challenging

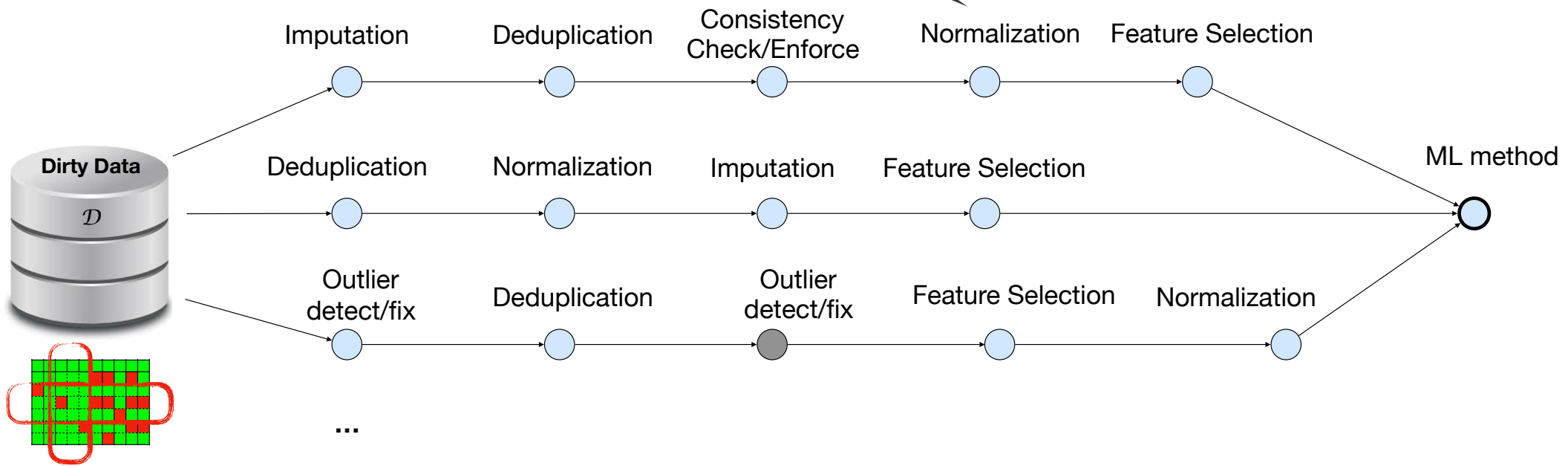


# Data preprocessing is challenging



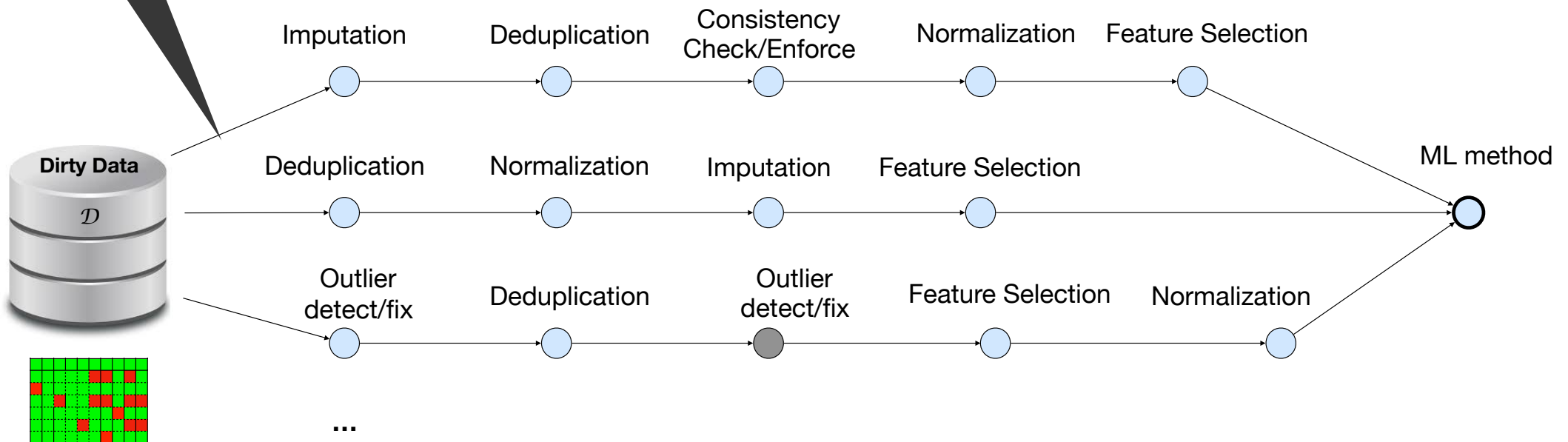
# Data preprocessing is challenging

Selective processing of some parts of the dataset



# Data preprocessing is challenging

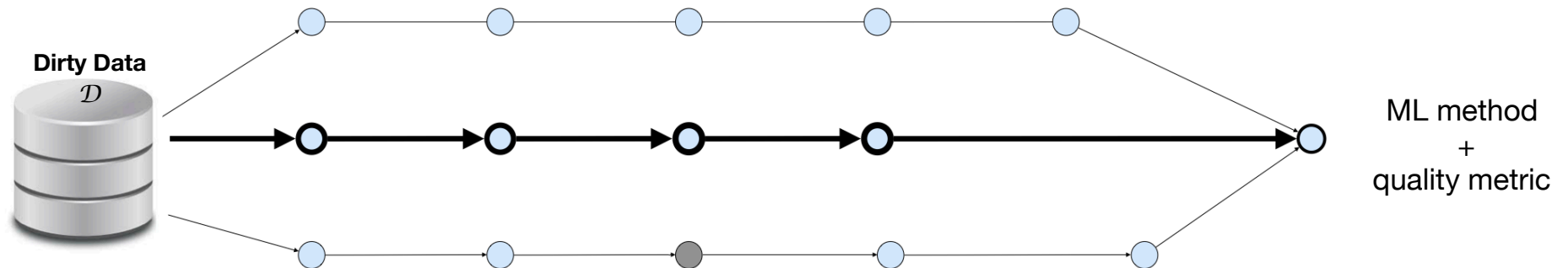
Infinite space of possible strategies



# Optimization Problem



Can we help the user in composing the data preparation pipeline that maximizes the quality performance of the ML method ?



# Optimization Problem



Can we help the user in composing the data preparation pipeline that maximizes the quality performance of the ML method ?

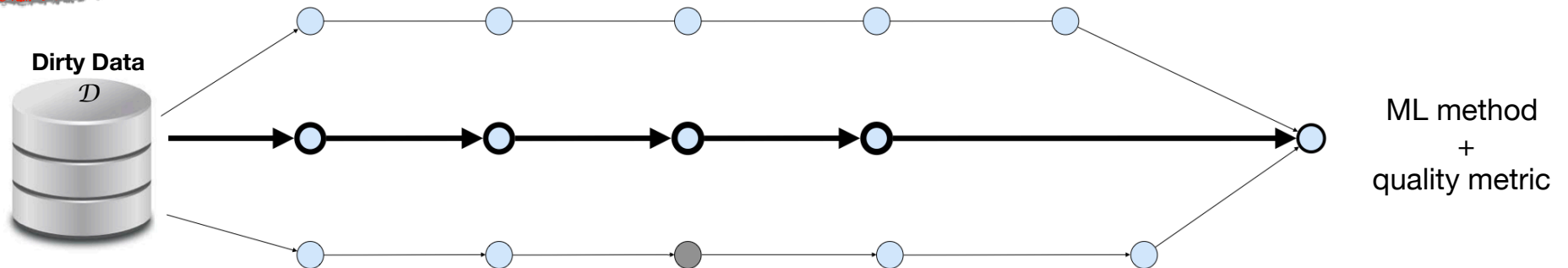
No training example for "good" data cleaning

AutoML-like approach

Metric-dependent

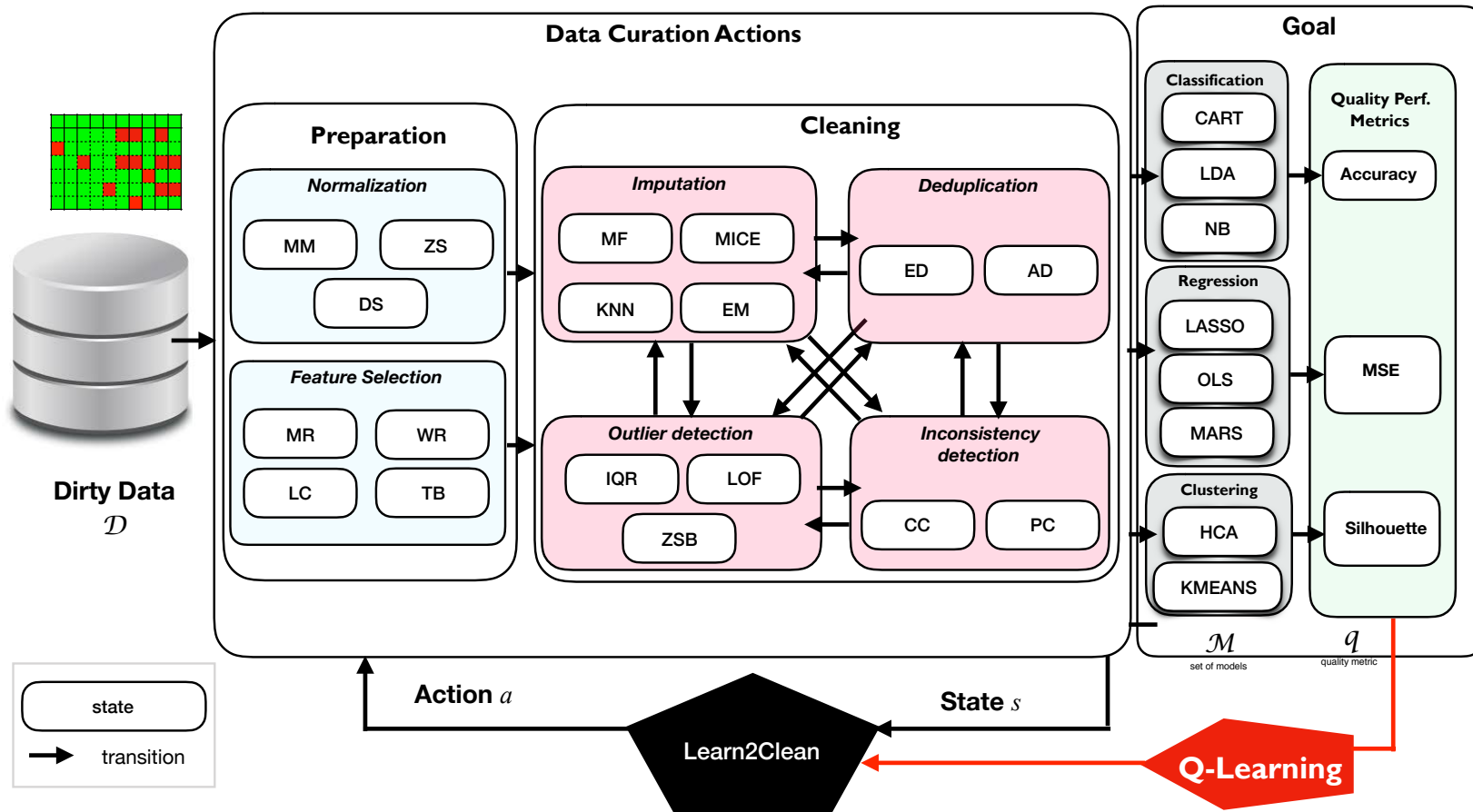
No model a priori

Human-In-The Loop



# First Solution: Learn2Clean

[The Web Conf 2019]



*AutoML-like approach for curation*

# Reinforcement Learning Framework

Markov Decision Process

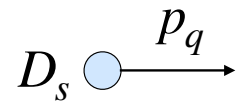
State

Action

Transition

Reward

Learn2Clean





# Reinforcement Learning Framework

Markov Decision Process

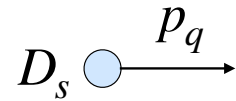
State

Action

Transition

Reward

Learn2Clean



# Reinforcement Learning Framework

Markov Decision Process

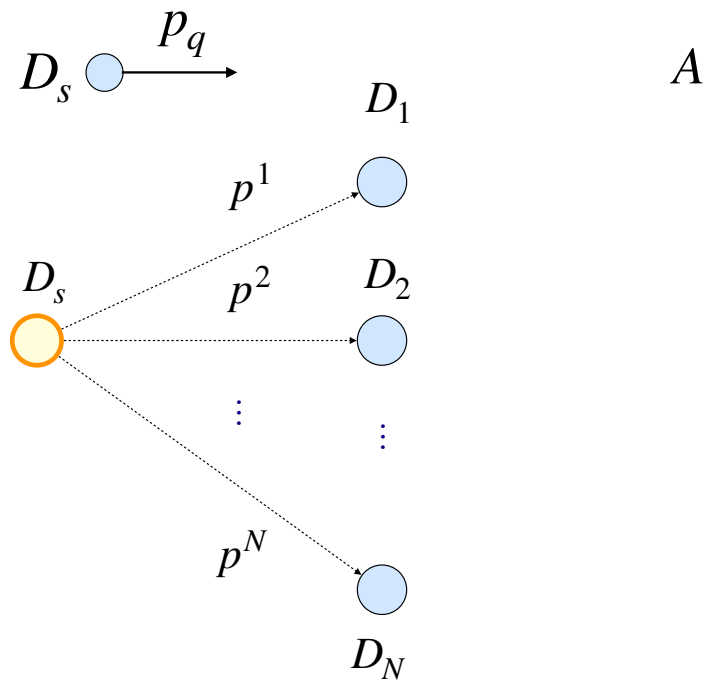
State

Action

Transition

Reward

Learn2Clean



# Reinforcement Learning Framework

Markov Decision Process

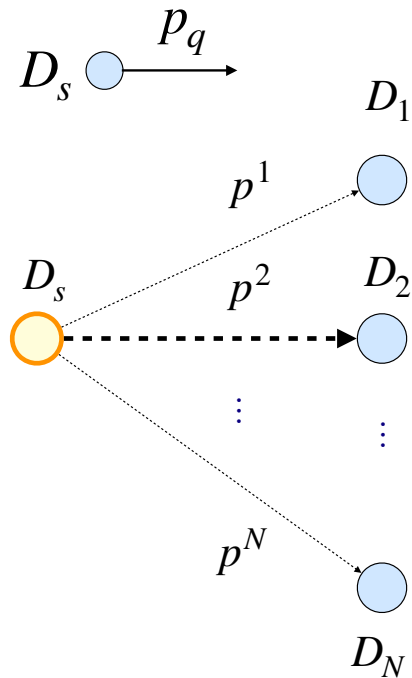
State

Action

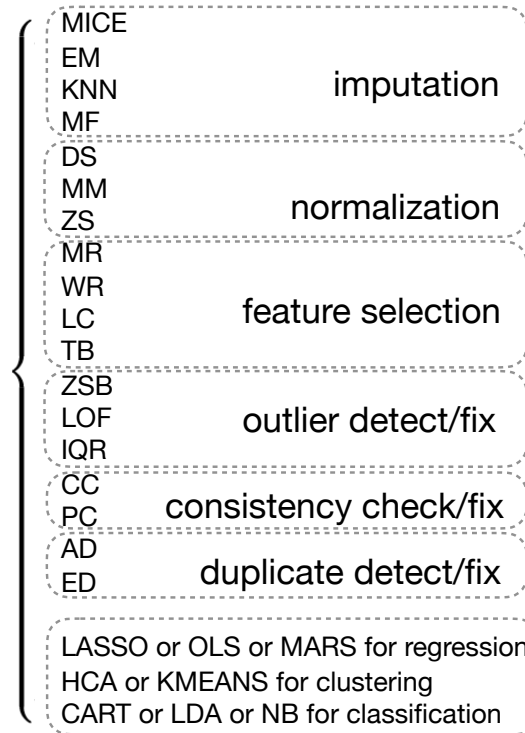
Transition

Reward

Learn2Clean



A



# Reinforcement Learning Framework

Markov Decision Process

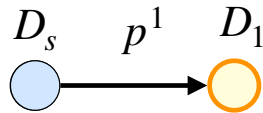
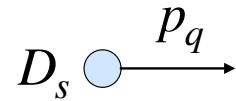
State

Action

Transition

Reward

Learn2Clean



# Reinforcement Learning Framework

Markov Decision Process

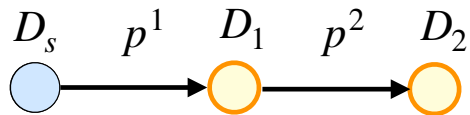
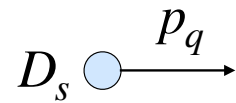
State

Action

Transition

Reward

Learn2Clean



# Reinforcement Learning Framework

Markov Decision Process

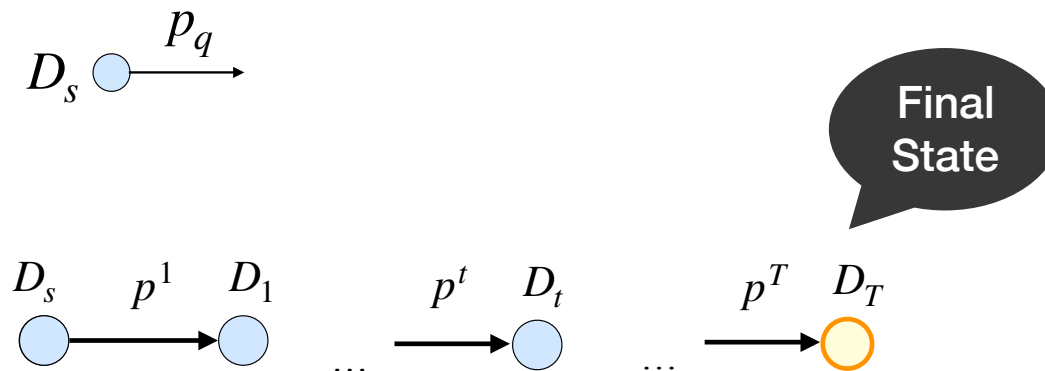
State

Action

Transition

Reward

Learn2Clean



LASSO or OLS or MARS for regression  
HCA or KMEANS for clustering  
CART or LDA or NB for classification

# Reinforcement Learning Framework

Markov Decision Process

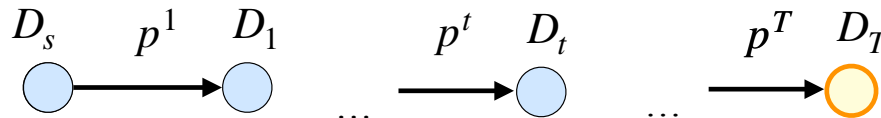
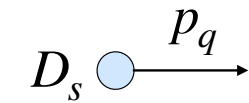
State

Action

Transition

Reward

Learn2Clean



*deterministic*

$R =$

MICE EM KNN MF DS MM ZS MR WR LC TB ZSB LOF IQR CC PC AD ED LASSO

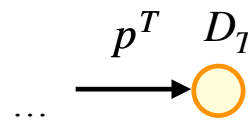
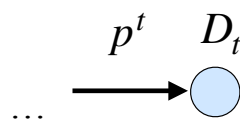
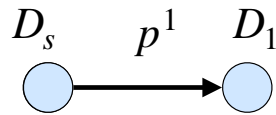
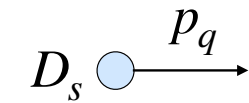
-1	-1	-1	-1	0	0	0	-1	0	0	0	0	0	0	0	0	-1	-1	0	0	100
-1	-1	-1	-1	0	0	0	-1	0	0	0	0	0	0	0	0	-1	-1	0	0	100
-1	-1	-1	-1	0	0	0	-1	0	0	0	0	0	0	0	0	-1	-1	0	0	100
-1	-1	-1	-1	0	0	0	-1	0	0	0	0	0	0	0	0	-1	-1	0	0	100
-1	-1	-1	-1	-1	-1	-1	0	0	0	0	0	0	0	0	0	-1	-1	0	0	-1
-1	-1	-1	-1	-1	-1	-1	0	0	0	0	0	0	0	0	0	-1	-1	0	0	-1
0	0	0	0	-1	-1	-1	0	0	0	0	0	0	0	0	0	-1	-1	0	0	-1
0	0	0	0	0	0	0	-1	-1	-1	-1	0	0	0	0	0	-1	-1	0	0	-1
0	0	0	0	0	0	0	-1	-1	-1	-1	0	0	0	0	0	-1	-1	0	0	-1
0	0	0	0	0	0	0	-1	-1	-1	-1	0	0	0	0	0	-1	-1	0	0	-1
0	0	0	0	-1	-1	-1	0	0	0	0	-1	-1	-1	-1	0	-1	-1	0	0	100
-1	-1	-1	-1	-1	-1	-1	0	0	0	0	-1	-1	-1	-1	-1	-1	-1	0	0	100
0	0	0	0	-1	-1	-1	0	0	0	0	-1	-1	-1	-1	-1	-1	-1	0	0	100
-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	0	0	0	0	0	-1	-1	0	0	100
-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	0	0	0	0	0	-1	-1	0	0	100
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	-1	-1	100
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	-1	100
-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1

# Reinforcement Learning Framework

Markov Decision Process



Learn2Clean



*deterministic*

MICE EM KNN MF DS MM ZS MR WR LC TB ZSB LOF IQR CC PC AD ED LASSO

$$R = \begin{bmatrix} -1 & -1 & -1 & -1 & 0 & 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -1 & -1 & 0 & 0 & 100 \\ -1 & -1 & -1 & -1 & 0 & 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -1 & -1 & 0 & 0 & 100 \\ -1 & -1 & -1 & -1 & 0 & 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -1 & -1 & 0 & 0 & 100 \\ -1 & -1 & -1 & -1 & -1 & -1 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -1 & -1 & 0 & 0 & -1 \\ -1 & -1 & -1 & -1 & -1 & -1 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -1 & -1 & 0 & 0 & -1 \\ 0 & 0 & 0 & 0 & -1 & -1 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -1 & -1 & 0 & 0 & -1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & -1 & -1 & -1 & -1 & 0 & 0 & 0 & 0 & 0 & -1 & -1 & 0 & 0 & -1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & -1 & -1 & -1 & -1 & 0 & 0 & 0 & 0 & 0 & -1 & -1 & 0 & 0 & -1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & -1 & -1 & -1 & -1 & 0 & 0 & 0 & 0 & 0 & -1 & -1 & 0 & 0 & -1 \\ 0 & 0 & 0 & 0 & -1 & -1 & -1 & 0 & 0 & 0 & 0 & -1 & -1 & -1 & -1 & 0 & 0 & 0 & 0 & -1 & 100 \\ -1 & -1 & -1 & -1 & -1 & -1 & -1 & 0 & 0 & 0 & 0 & -1 & -1 & -1 & -1 & 0 & 0 & 0 & 0 & 100 \\ 0 & 0 & 0 & 0 & -1 & -1 & -1 & 0 & 0 & 0 & 0 & -1 & -1 & -1 & -1 & 0 & 0 & 0 & 0 & 100 \\ -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & 0 & 0 & 0 & 0 & -1 & -1 & 0 & 0 & 100 \\ -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & 0 & 0 & 0 & 0 & -1 & -1 & 0 & 0 & 100 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -1 & -1 & 100 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -1 & -1 & 100 \\ -1 & -1 \end{bmatrix}$$

- > MSE
- > Silhouette
- > Accuracy

Quality metric



# Reinforcement Learning Framework

Markov Decision Process

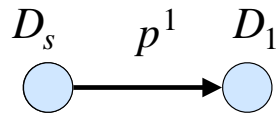
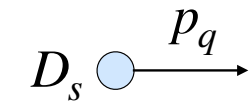
State

Action

Transition

Reward

Learn2Clean



Final State

*deterministic*

MICE EM KNN MF DS MM ZS MR WR LC TB ZSB LOF IQR CC PC AD ED LASSO

$R =$

-1	-1	-1	-1	0	0	0	-1	0	0	0	0	0	0	0	0	0	0	0	0	-1	-1	0	0	100	
-1	-1	-1	-1	0	0	0	-1	0	0	0	0	0	0	0	0	0	0	0	0	-1	-1	0	0	100	
-1	-1	-1	-1	0	0	0	-1	0	0	0	0	0	0	0	0	0	0	0	0	-1	-1	0	0	100	
-1	-1	-1	-1	0	0	0	-1	0	0	0	0	0	0	0	0	0	0	0	0	-1	-1	0	0	100	
-1	-1	-1	-1	-1	-1	-1	0	0	0	0	0	0	0	0	0	0	0	0	0	-1	-1	0	0	-1	
0	0	0	0	-1	-1	-1	0	0	0	0	0	0	0	0	0	0	0	0	0	-1	-1	0	0	-1	
0	0	0	0	0	0	0	-1	-1	-1	-1	-1	0	0	0	0	0	0	0	0	-1	-1	0	0	-1	
0	0	0	0	0	0	0	-1	-1	-1	-1	-1	-1	0	0	0	0	0	0	0	-1	-1	0	0	-1	
0	0	0	0	-1	-1	-1	0	0	0	0	0	0	-1	-1	-1	-1	-1	-1	-1	-1	-1	0	0	100	
-1	-1	-1	-1	-1	-1	-1	0	0	0	0	0	0	-1	-1	-1	-1	-1	-1	-1	-1	-1	0	0	100	
0	0	0	0	-1	-1	-1	0	0	0	0	0	0	-1	-1	-1	-1	-1	-1	-1	-1	-1	0	0	100	
-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	0	0	0	0	0	0	0	-1	-1	0	0	100	
-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	0	0	0	0	0	0	0	-1	-1	0	0	100	
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	-1	-1	100
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	-1	-1	100
-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1

LASSO or OLS or MARS for regression  
HCA or KMEANS for clustering  
CART or LDA or NB for classification

→ MSE  
→ Silhouette  
→ Accuracy

Quality metric

Learn2Clean selects the sequence of preprocessing actions that maximizes the quality metric (resp. minimizes the error)

# Reinforcement Learning Framework

Markov Decision Process

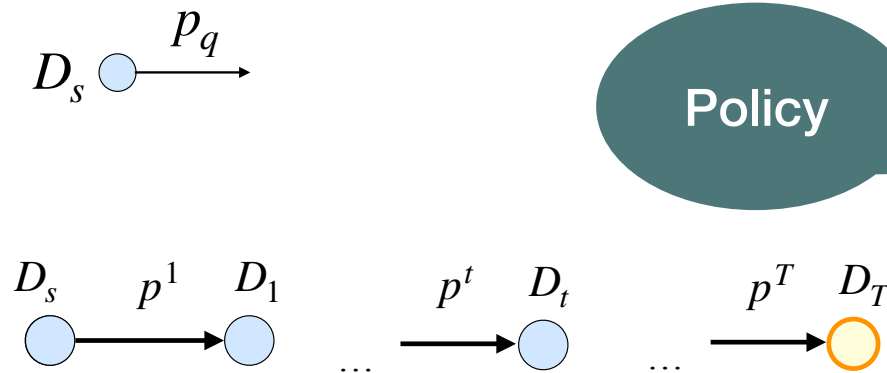
State

Action

Transition

Reward

Learn2Clean



**Softmax action selection**

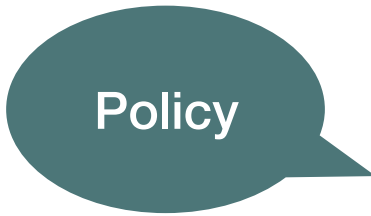
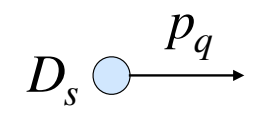
$$\pi = P(a | s) = \frac{e^{Q(s,a)/k}}{\sum_j e^{Q(s,a_j)/k}}$$

# Reinforcement Learning Framework

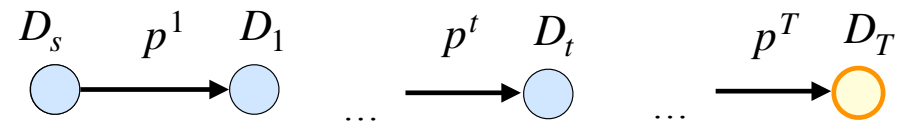
Markov Decision Process



Learn2Clean



**Softmax action selection**

$$\pi = P(a | s) = \frac{e^{Q(s,a)/k}}{\sum_j e^{Q(s,a_j)/k}}$$


**Q-table**

*value iteration update*

$$Q^\pi(s, a) \leftarrow (1 - \alpha) \cdot Q(s, a) + \alpha \cdot \left( R(s, a) + \gamma \cdot \max_{a'} Q(s', a') \right)$$

↑ new value      ↗ learning rate      ↑ old value      ↑ reward      ↑ discount factor      ↘ optimal future value

learned value

# Experiment Setup

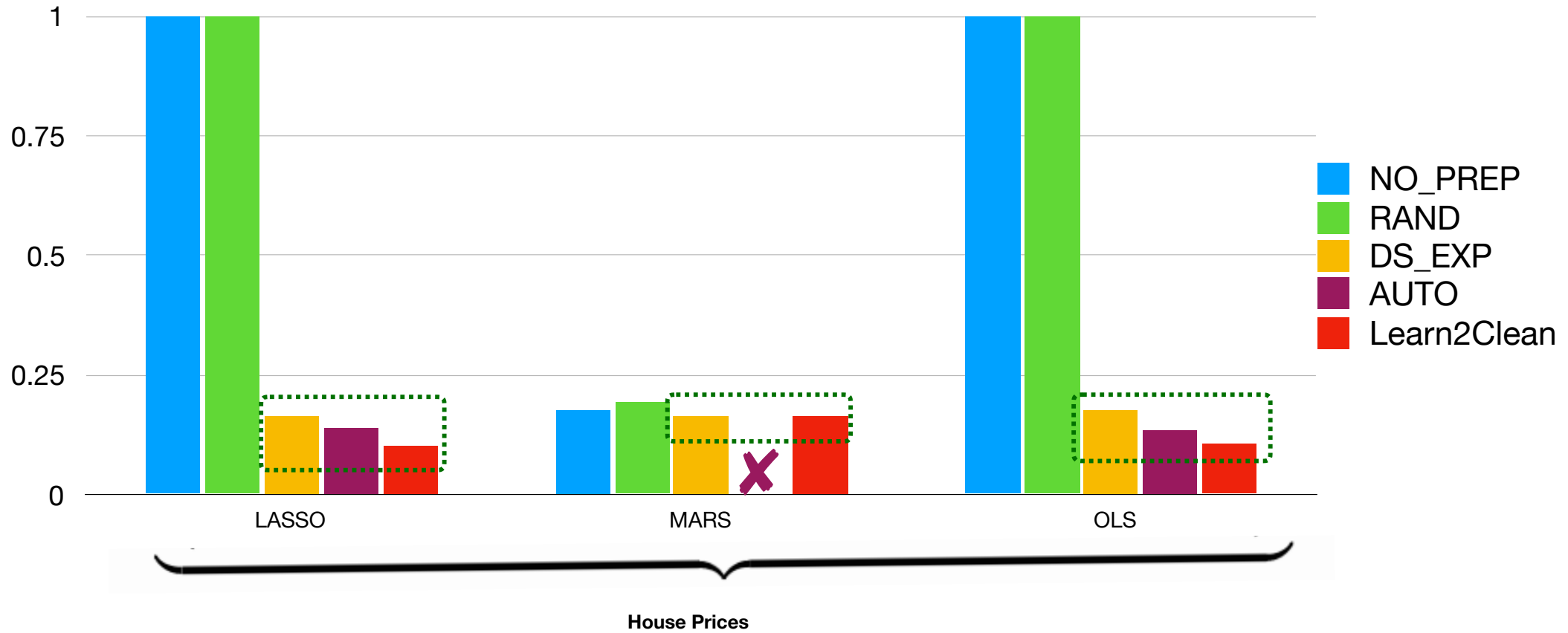
## Datasets

Name	# Att.	# Rows	Clustering	Regression	Classification
House Prices	81	1.46k	✓	✓	✓
Google Playstore Users	5	64.3k	✓		
Google Playstore Apps	13	10.8k	✓		✓

**Evaluation** : Silhouette for Clustering  
MSE for Regression  
Accuracy for Classification

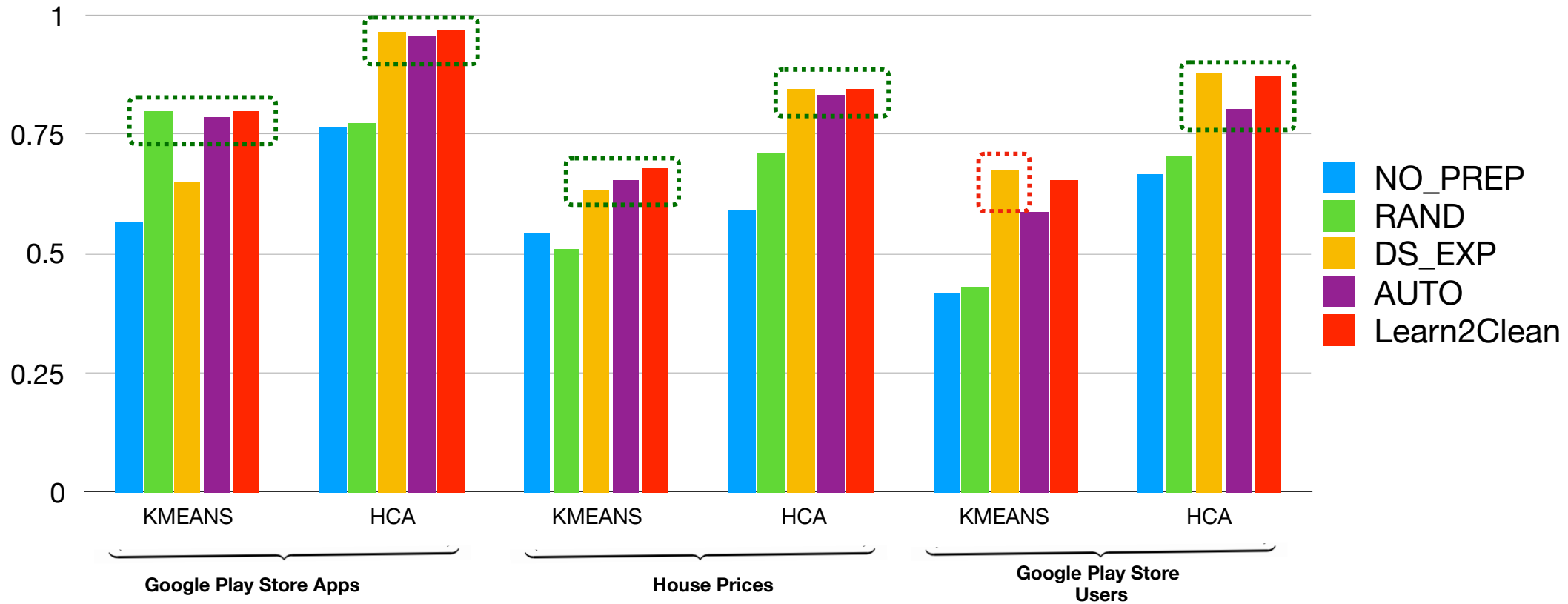
# Experimental Results

## Regression (MSE)



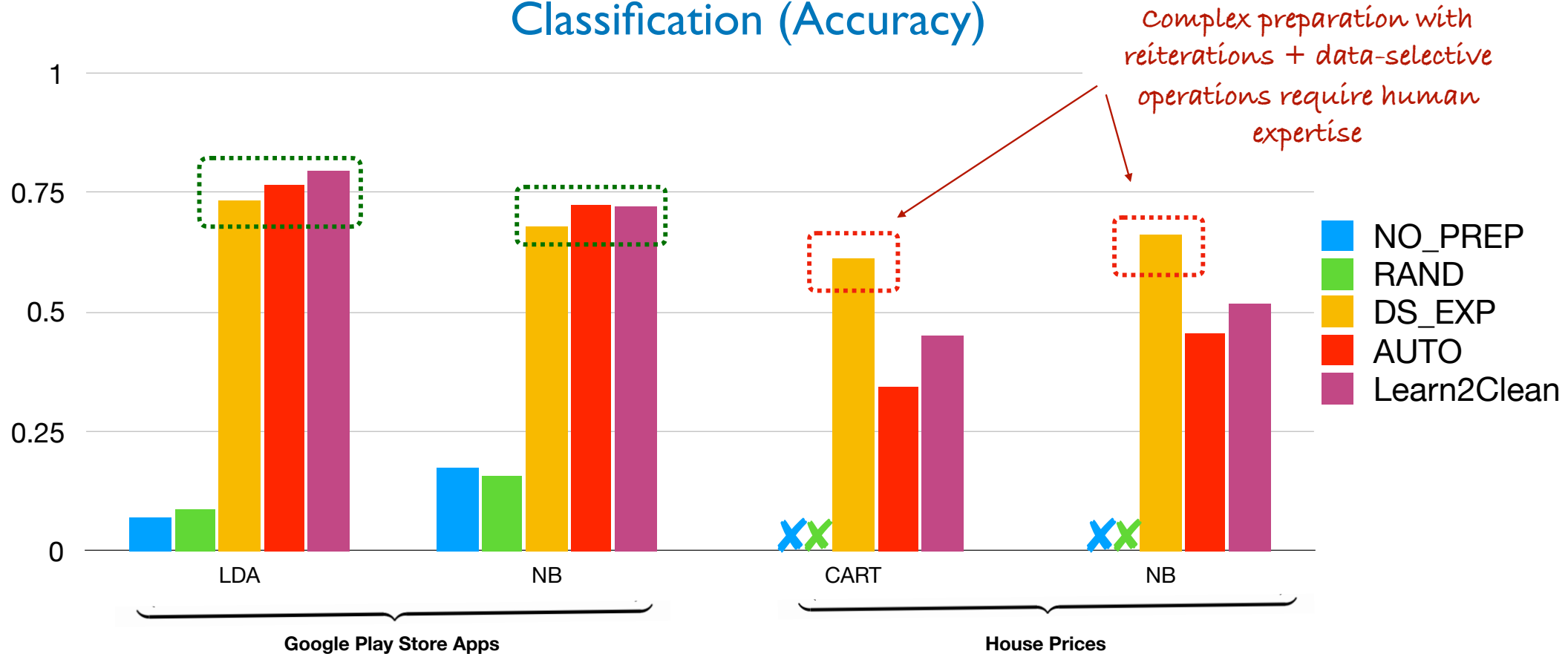
# Experimental Results

## Clustering (Silhouette)

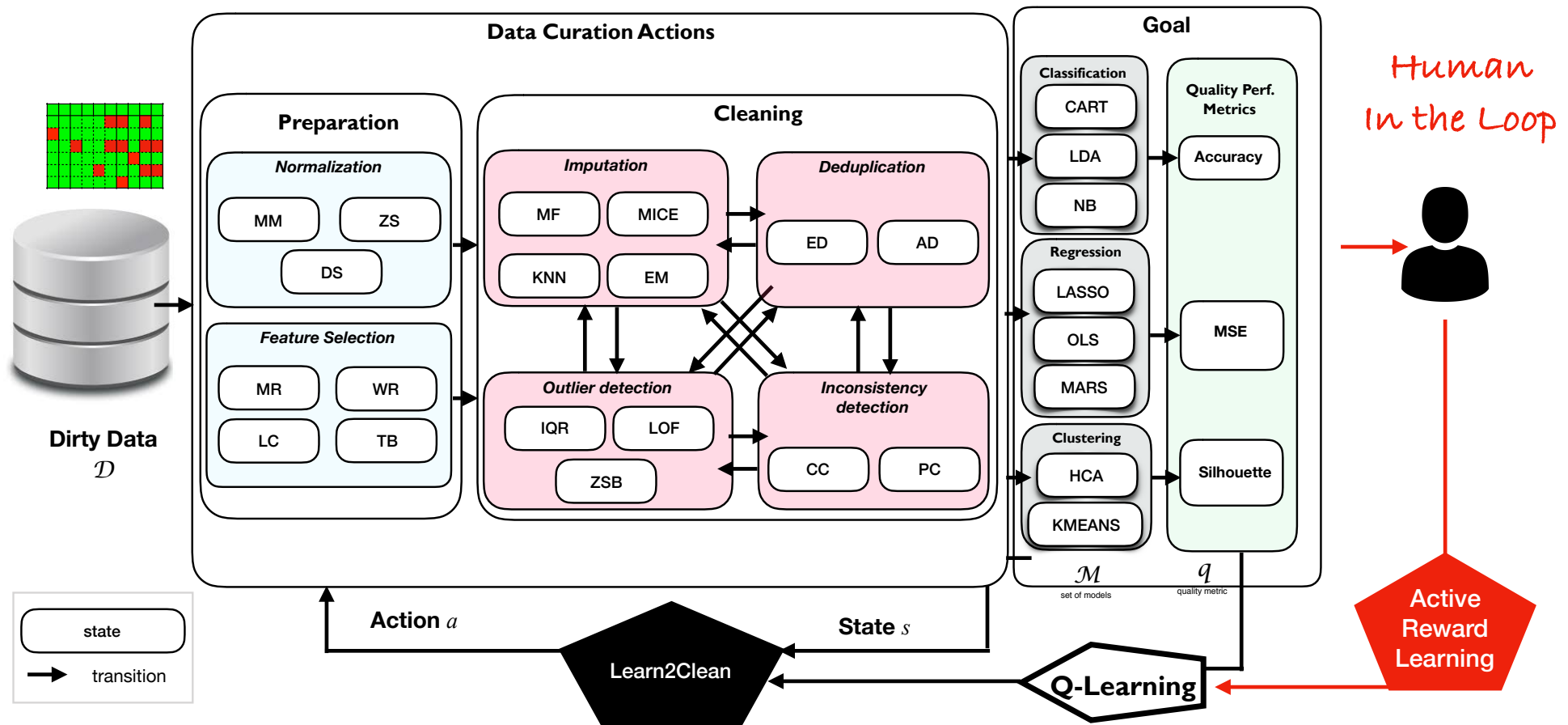


# Experimental Results

## Classification (Accuracy)



# New Version: HIL with Active Reward Learning

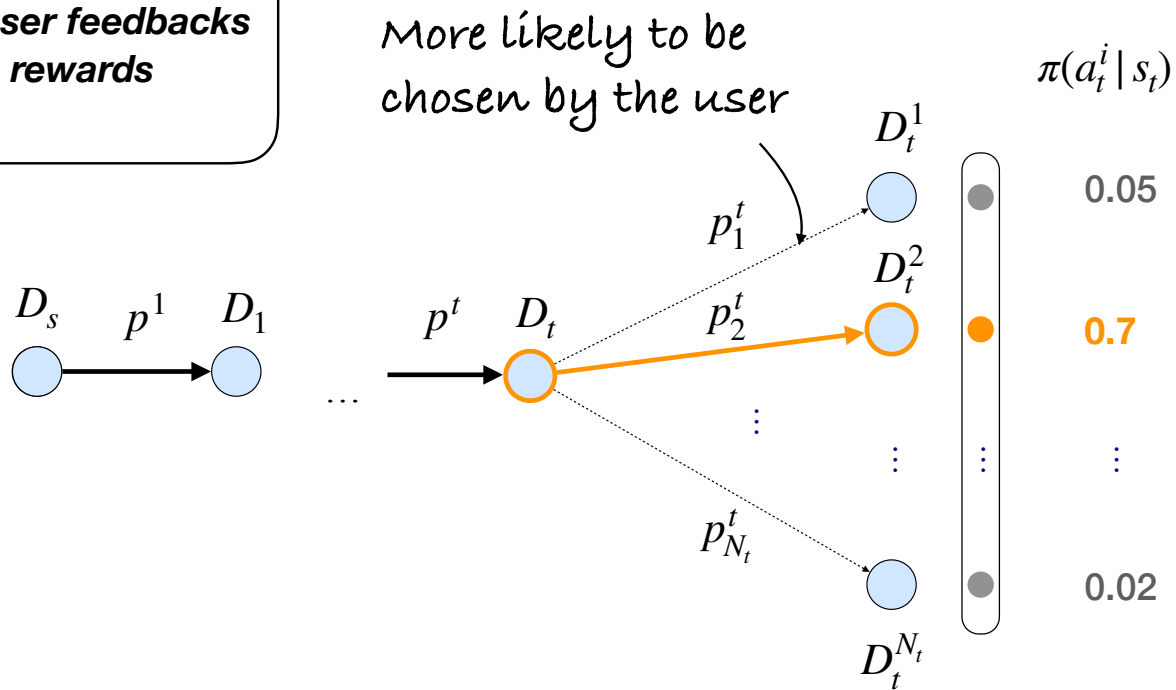




# Active Reward Learning

Learn2Clean  
+  
HIL

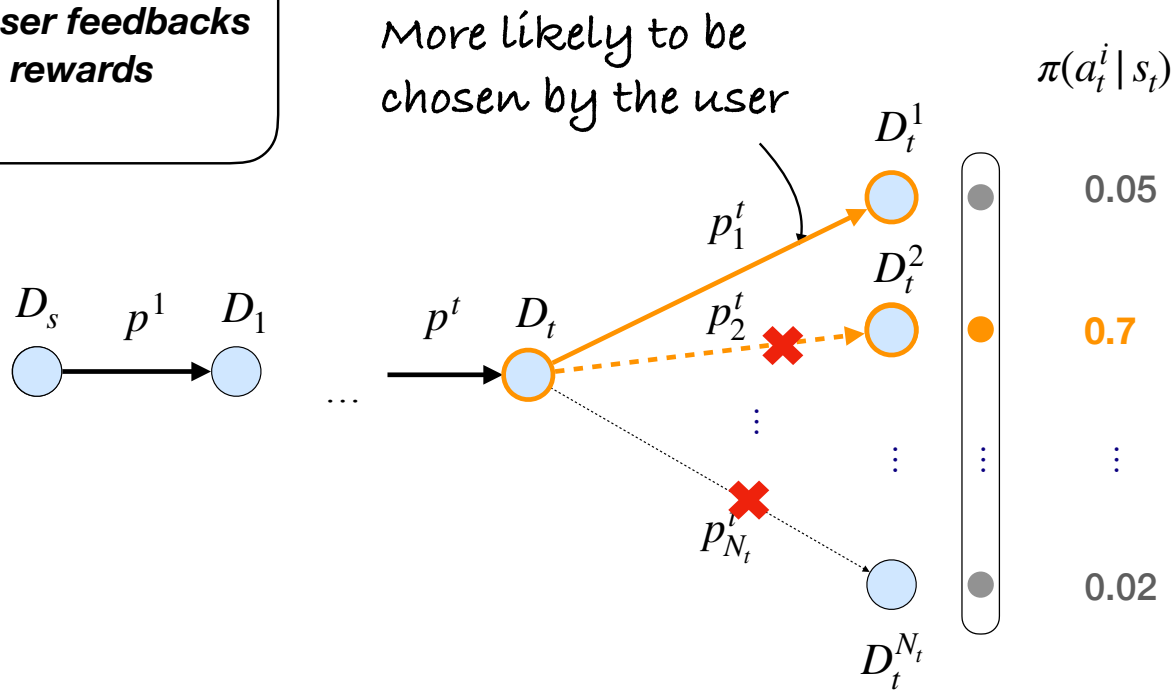
Goal: *learn from user feedbacks to adapt the rewards*



# Active Reward Learning

Learn2Clean  
+  
HIL

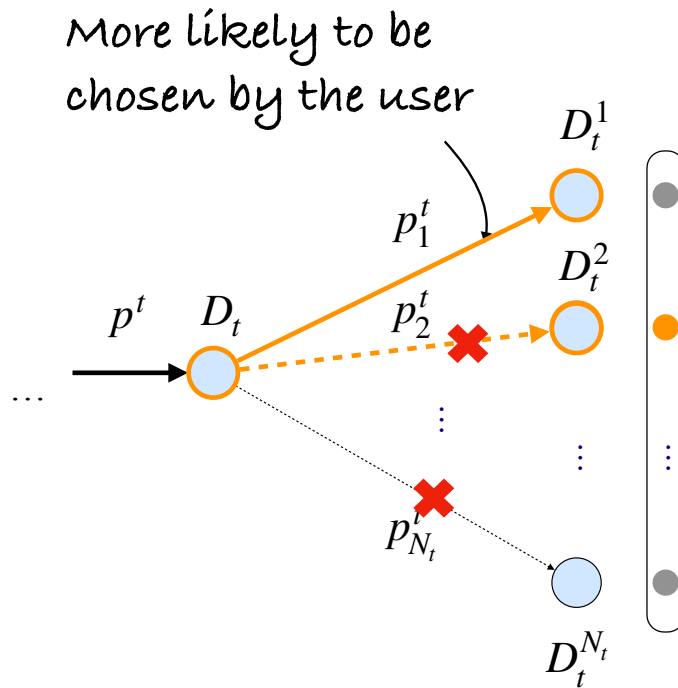
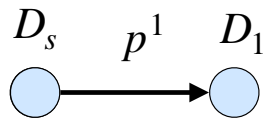
Goal: *learn from user feedbacks to adapt the rewards*



# Active Reward Learning

Learn2Clean  
+  
HIL

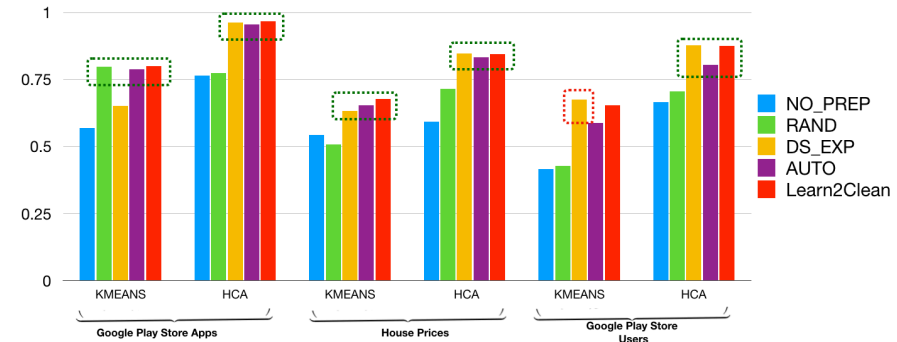
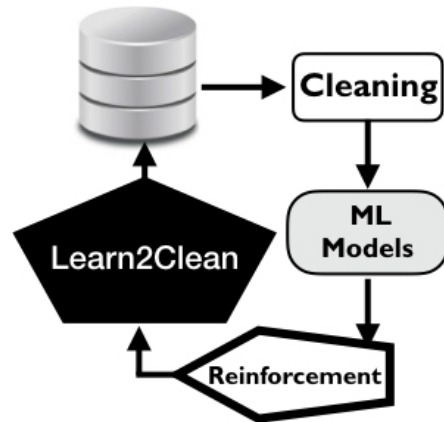
Goal: *learn from user feedbacks to adapt the rewards*



$\tilde{\pi}(a_t^i   s_t)$	$\pi(a_t^i   s_t)$
0.9	0.05
0	0.7
⋮	⋮
0	0.02

Force exploration

Code: <https://github.com/LaureBerti/Learn2Clean>



## Future directions

- Combine AutoML and AutoCuration
- Learn better reward functions
- Extend the library of ML and preparation methods
- Investigate other RL techniques (e.g., deep RL, on-policy, model)
- Extend experiments with more intricate data glitches and various glitch distributions

Thank you!