Designing DAG-shaped Classifiers for Fast Triggers

Djalel Benbouzid
Vladimir Vava Gligorov
Michael Williams
Balázs Kégl

benbouzid @ lal.in2p3.fr
Laboratoire de l'Accélérateur Linéaire
Université Paris-Sud, CNRS/IN2P3

May, 17th 2013

Motivation

Fast Classification and Trigger Design

- High Level Triggers,
- Face detection,
- Web page ranking, . . .
- Imbalanced data
- Real-time classification
- Limited budget
- Multi-class

Fast Classification and Trigger Design

- High Level Triggers,
- Face detection,
- Web page ranking, ...
- Imbalanced data
- Real-time classification
- Limited budget
- Multi-class

What might come to mind...

Adaboost (Freund and Schapire 1995) [1]

Good properties both theoretical and practical

Multi-class flavors like Adaboost.MH

Versatile can be used with a plethora of base learners

Anytime Direct control over over the complexity

1. Freund, Yoav, and Robert E. Schapire. A desicion-theoretic generalization of on-line learning and an application to boosting. Computational learning theory. Springer Berlin Heidelberg, 1995.

Adaboost.MH

- Iteratively constructs a pool of base classifiers
- For K classes, returns a score function of the form

$$\mathbf{f}(\mathbf{x}) = \sum_{i=1}^{T} \alpha_i \mathbf{h}_i(\mathbf{x}) \in \mathbb{R}^K$$

$$\text{Prediction : } \widehat{\ell} = \arg\max_{\ell} f_\ell(\mathbf{x})$$

- Usually not applicable to real-time classification (big T)
- Representation

Adaboost.MH

- Iteratively constructs a pool of base classifiers
- For K classes, returns a score function of the form

$$\mathbf{f}(\mathbf{x}) = \sum_{i=1}^{T} \alpha_i \mathbf{h}_i(\mathbf{x}) \in \mathbb{R}^K$$

$$\text{Prediction : } \widehat{\ell} = \arg\max_{\ell} f_\ell(\mathbf{x})$$

- Usually not applicable to real-time classification (big T)
- Representation

Adaboost.MH

- Iteratively constructs a pool of base classifiers
- For K classes, returns a score function of the form

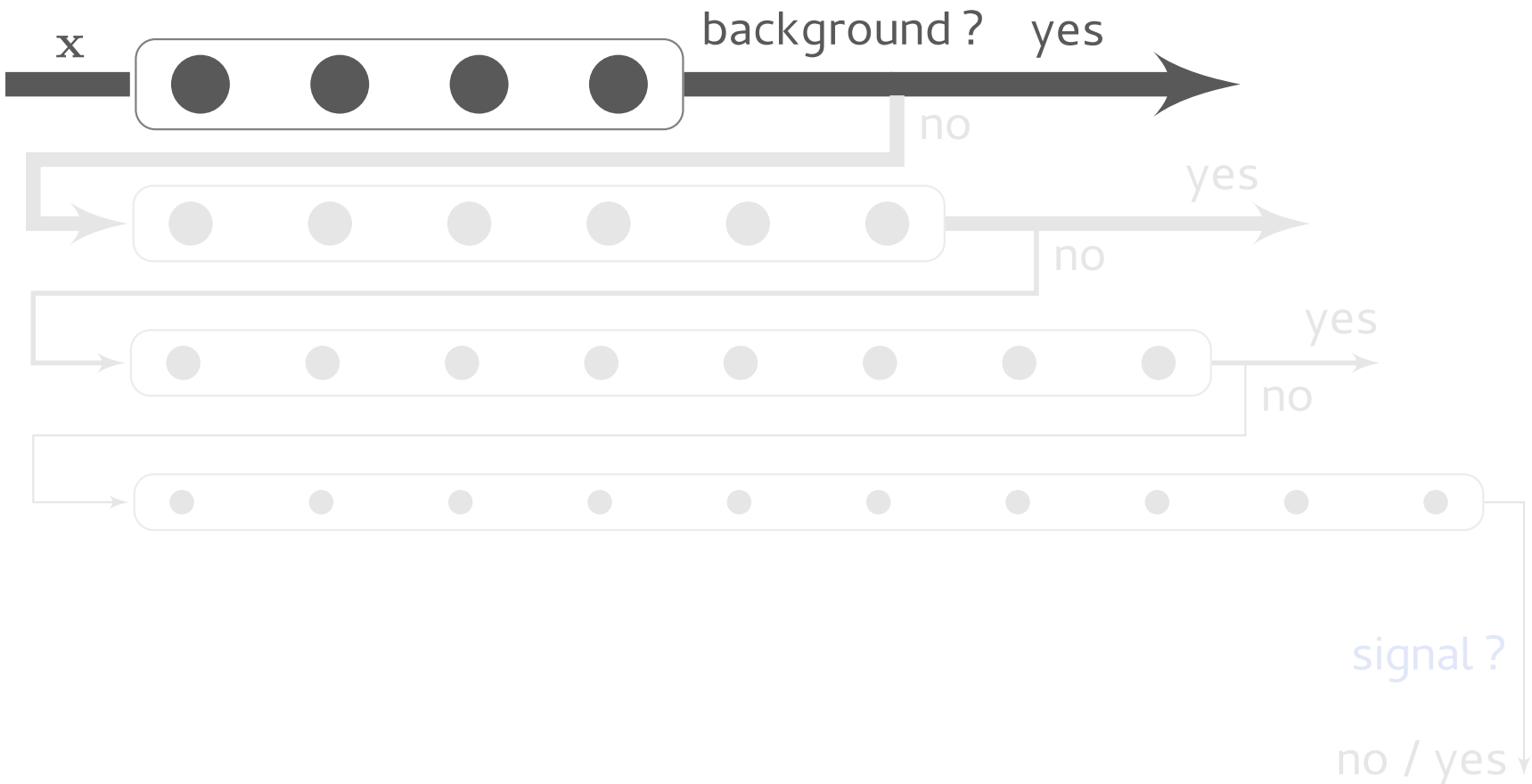
$$\mathbf{f}(\mathbf{x}) = \sum_{i=1}^{T} \alpha_i \mathbf{h}_i(\mathbf{x}) \in \mathbb{R}^K$$

$$\text{Prediction : } \widehat{\ell} = \arg\max_{\ell} f_\ell(\mathbf{x})$$

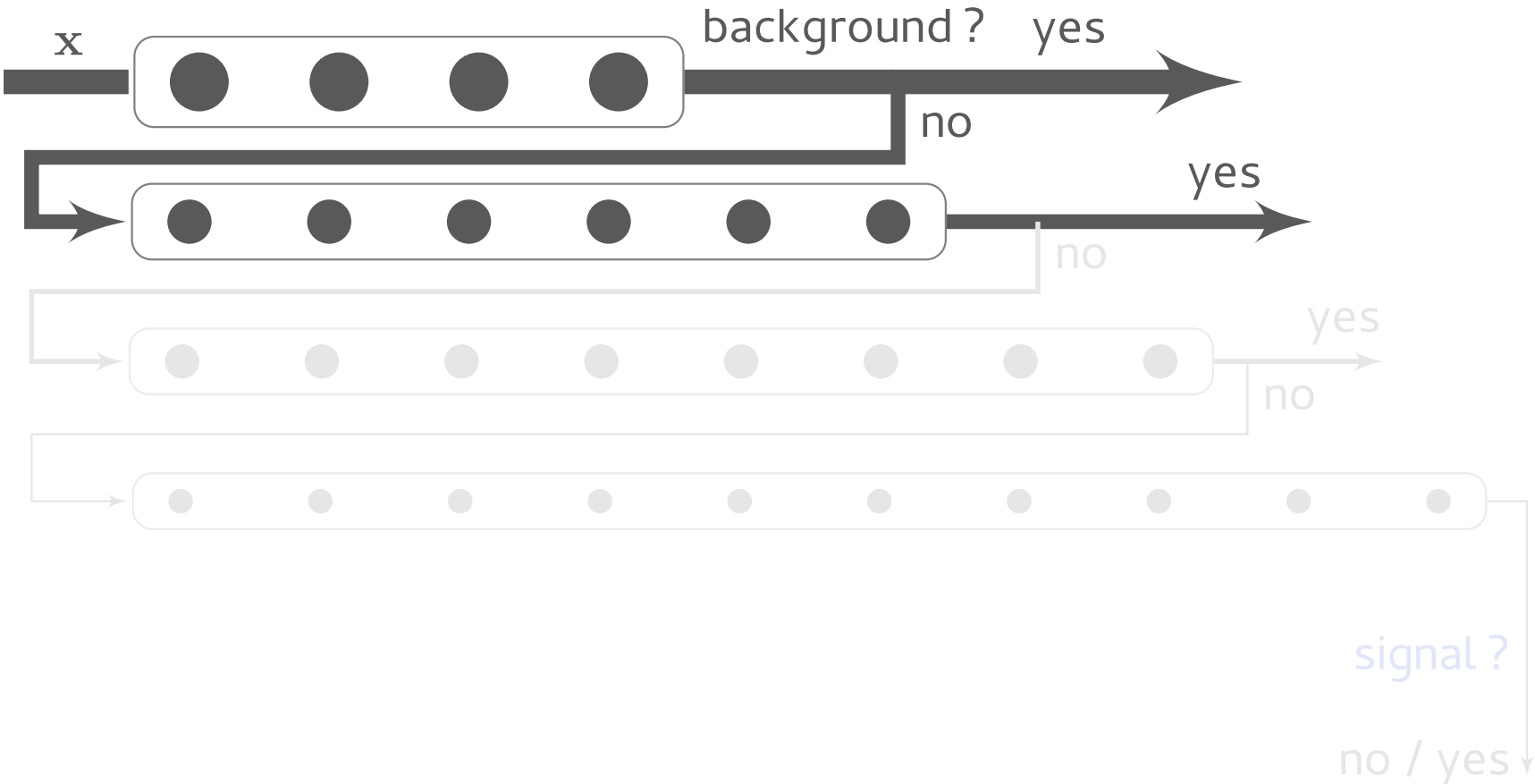
- Usually not applicable to real-time classification (big T)
- Representation

Cascade classifiers

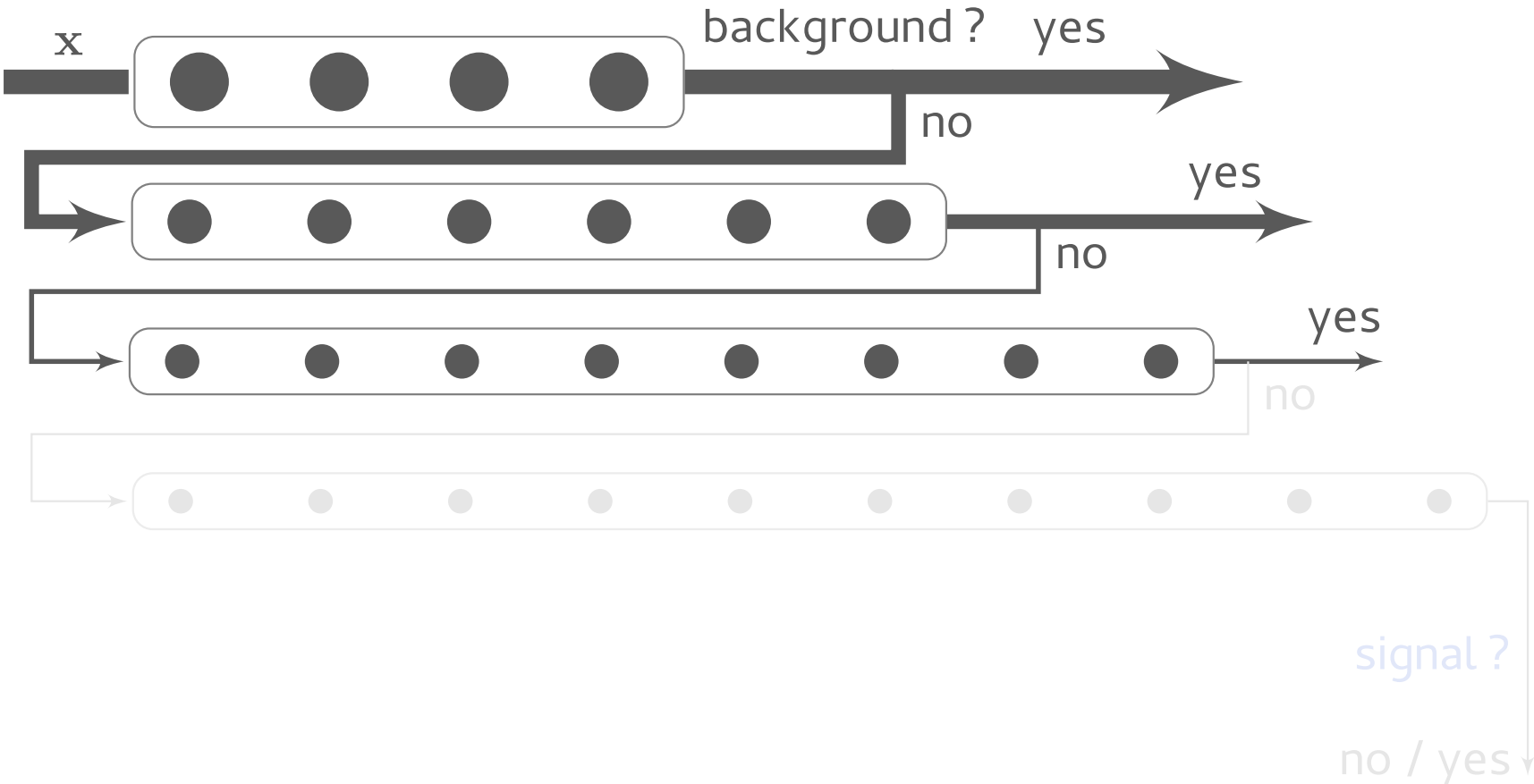
Chaining classifiers



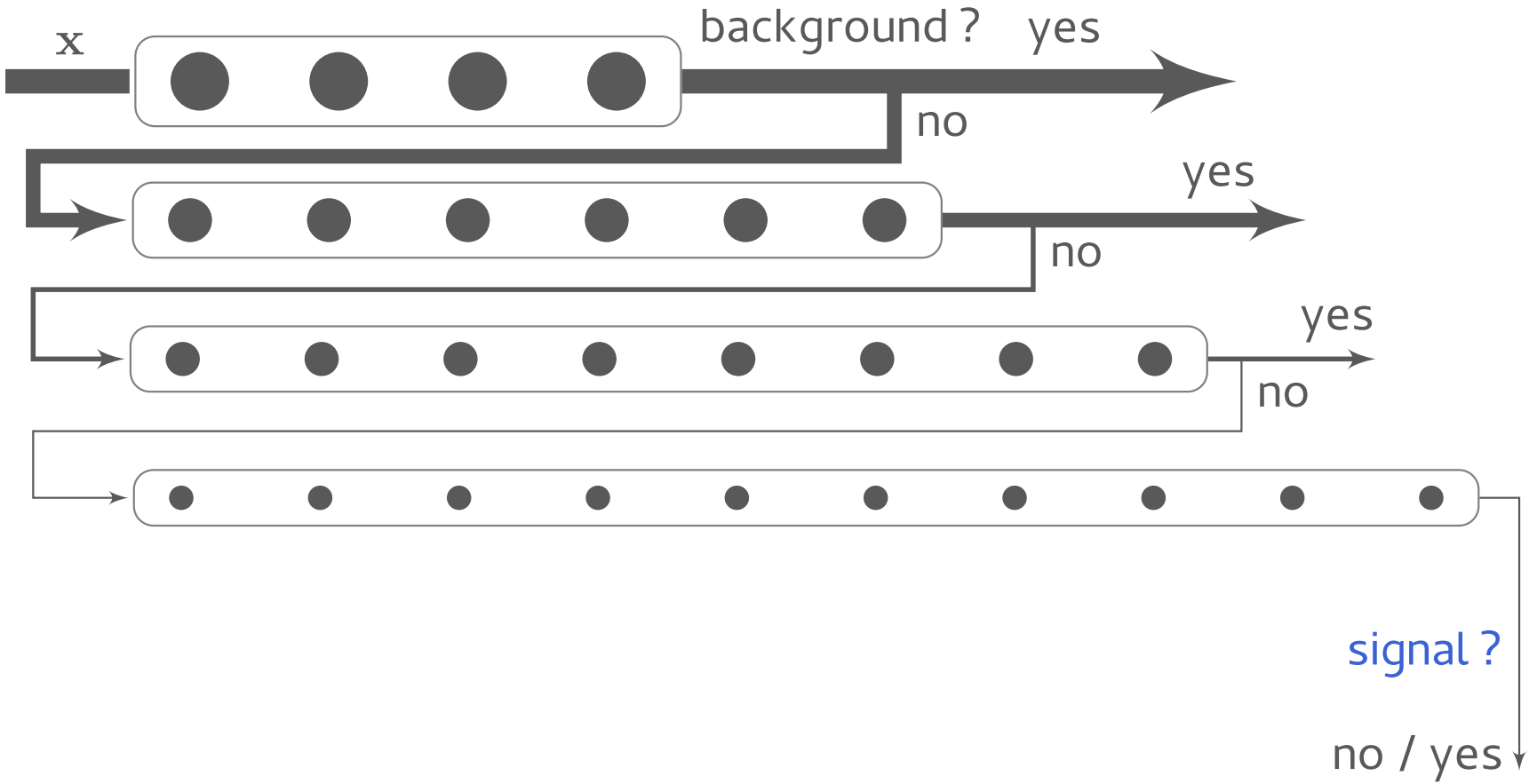
Chaining classifiers



Chaining classifiers



Chaining classifiers



But...

- Hand-tuning of the hyper-parameters.
- No early classification for signal.
- The margin information is lost.
- No straightforward extension to multi-class classification.

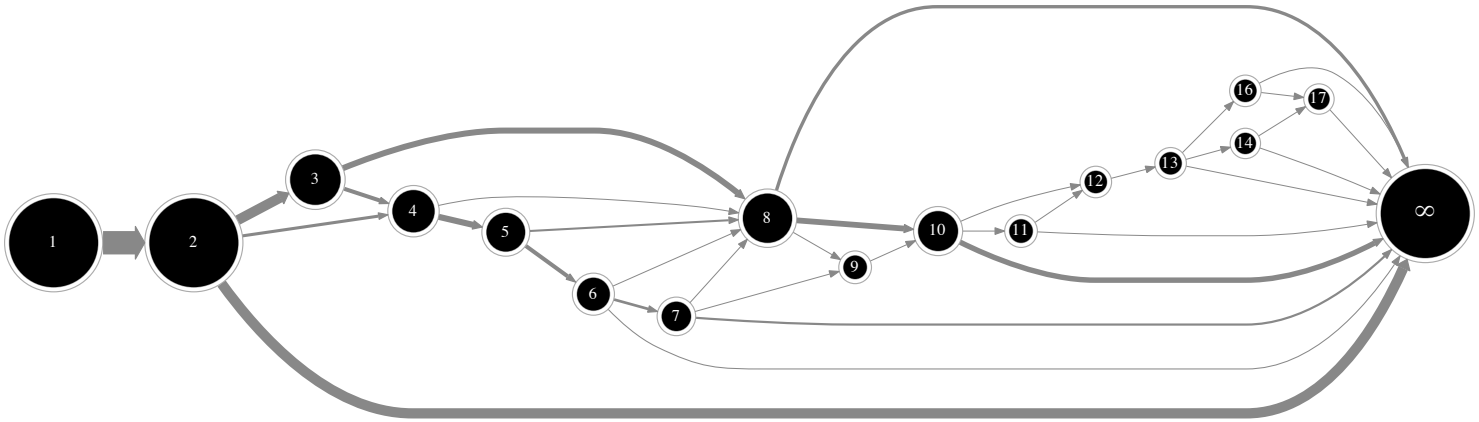
Cascade improvements

- Viola & Jones cascade (IJCV 2004)
- SoftCascade (Bourdev et. al CVPR 2005)
- FCBoost (Saberian et. al. NIPS 2010)
- SVMBoost (Xu et. al. ICML 2013)

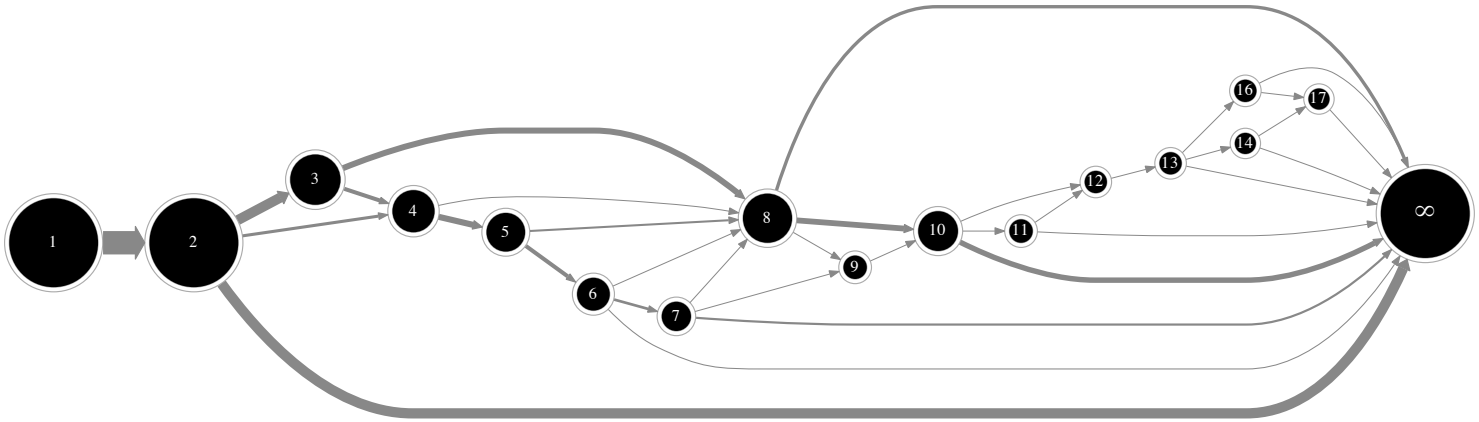
MDDAG

Markov Decision Directed Acyclic Graph

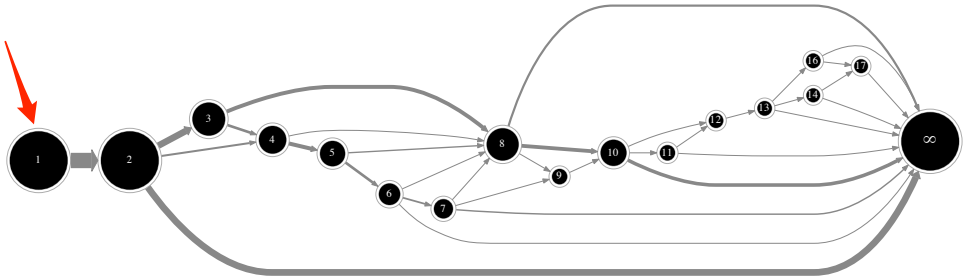
Example (trailer...)

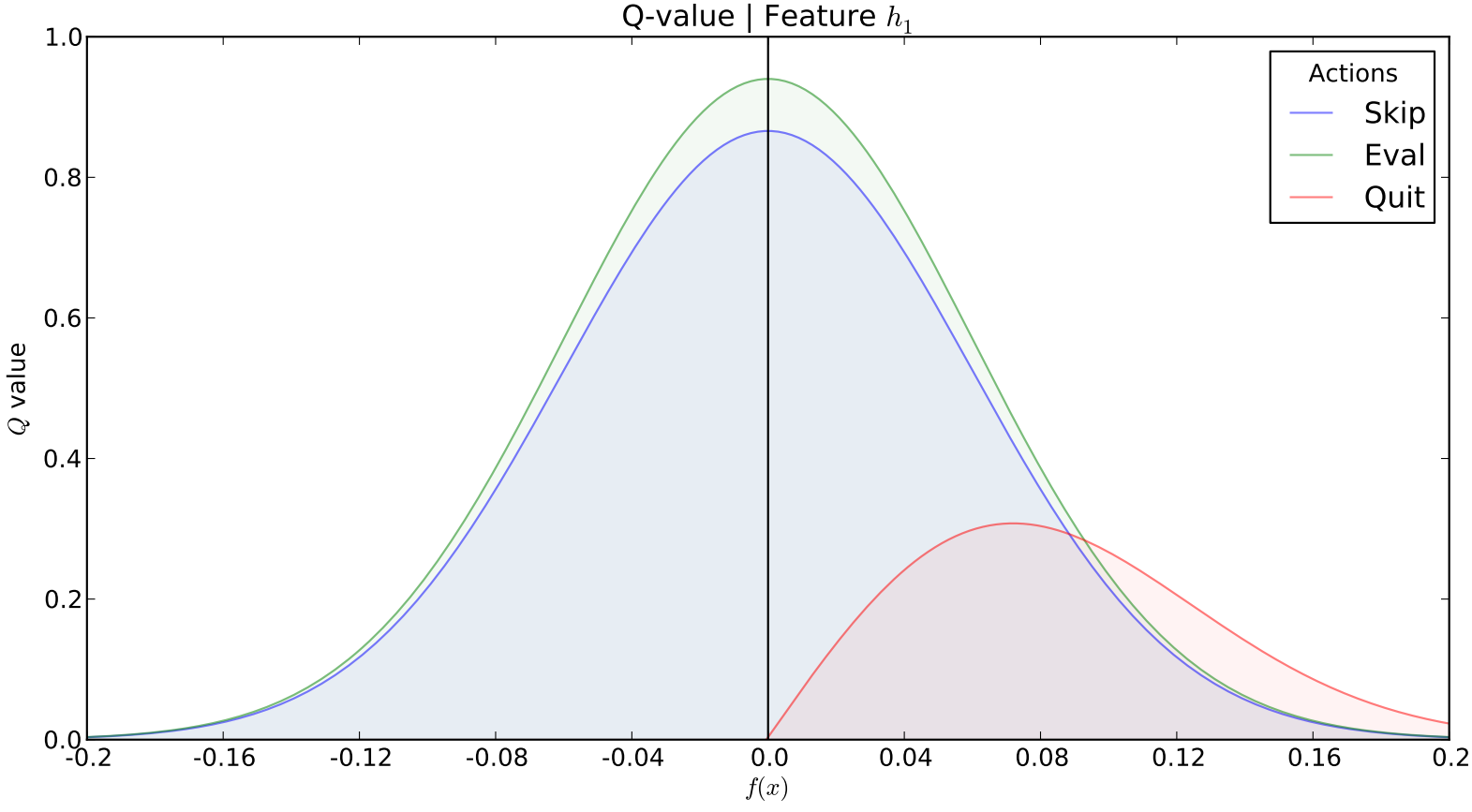


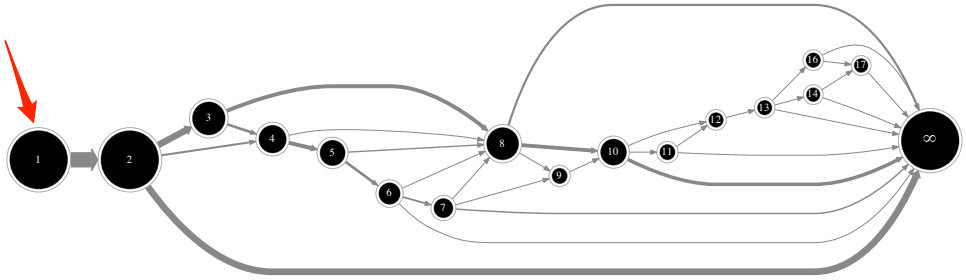
Example (trailer...)

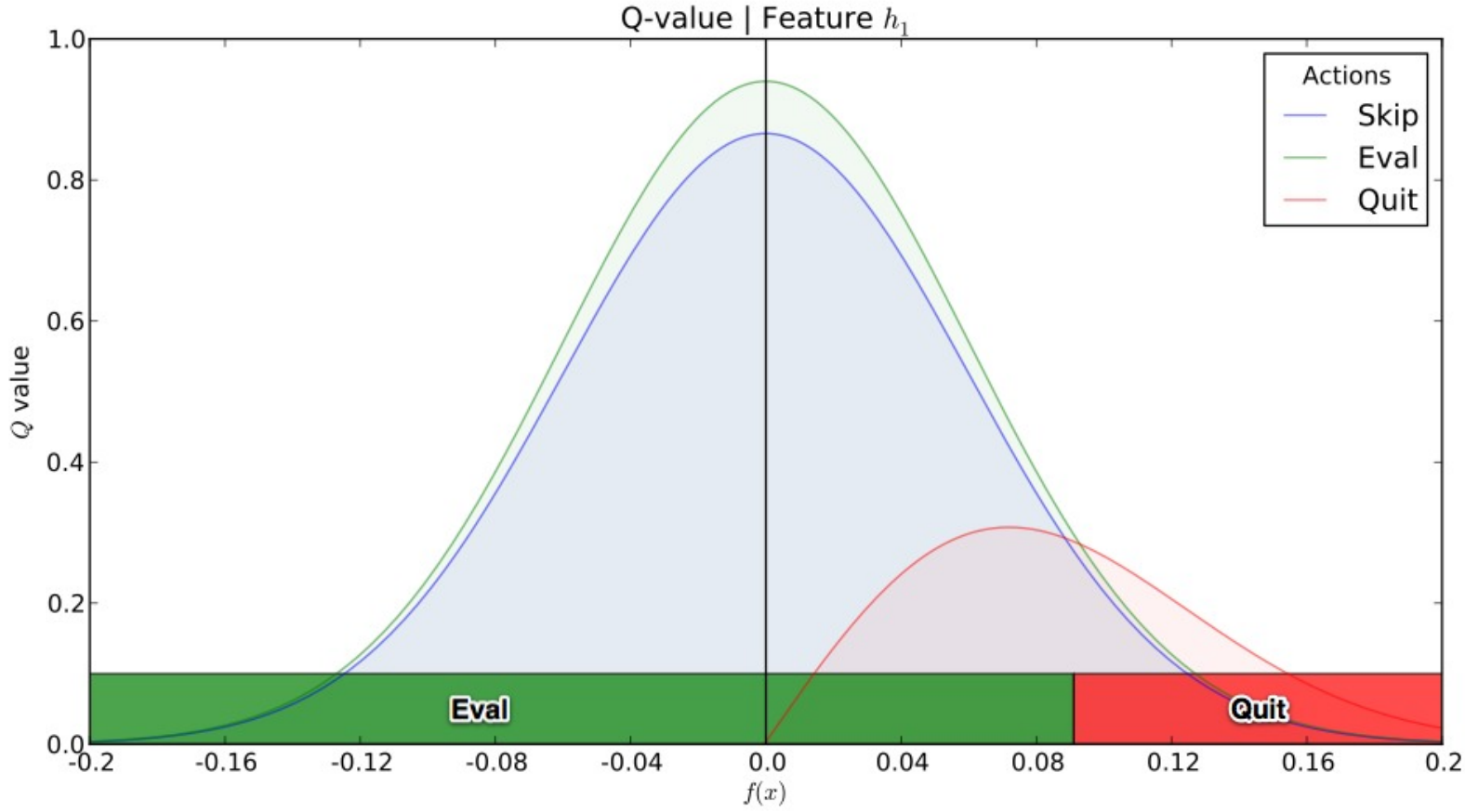


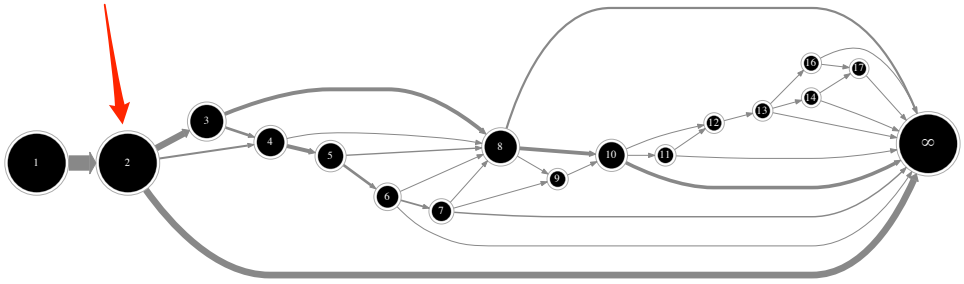
- Face instance 
- Path : 1, 2, 3, 8, ∞

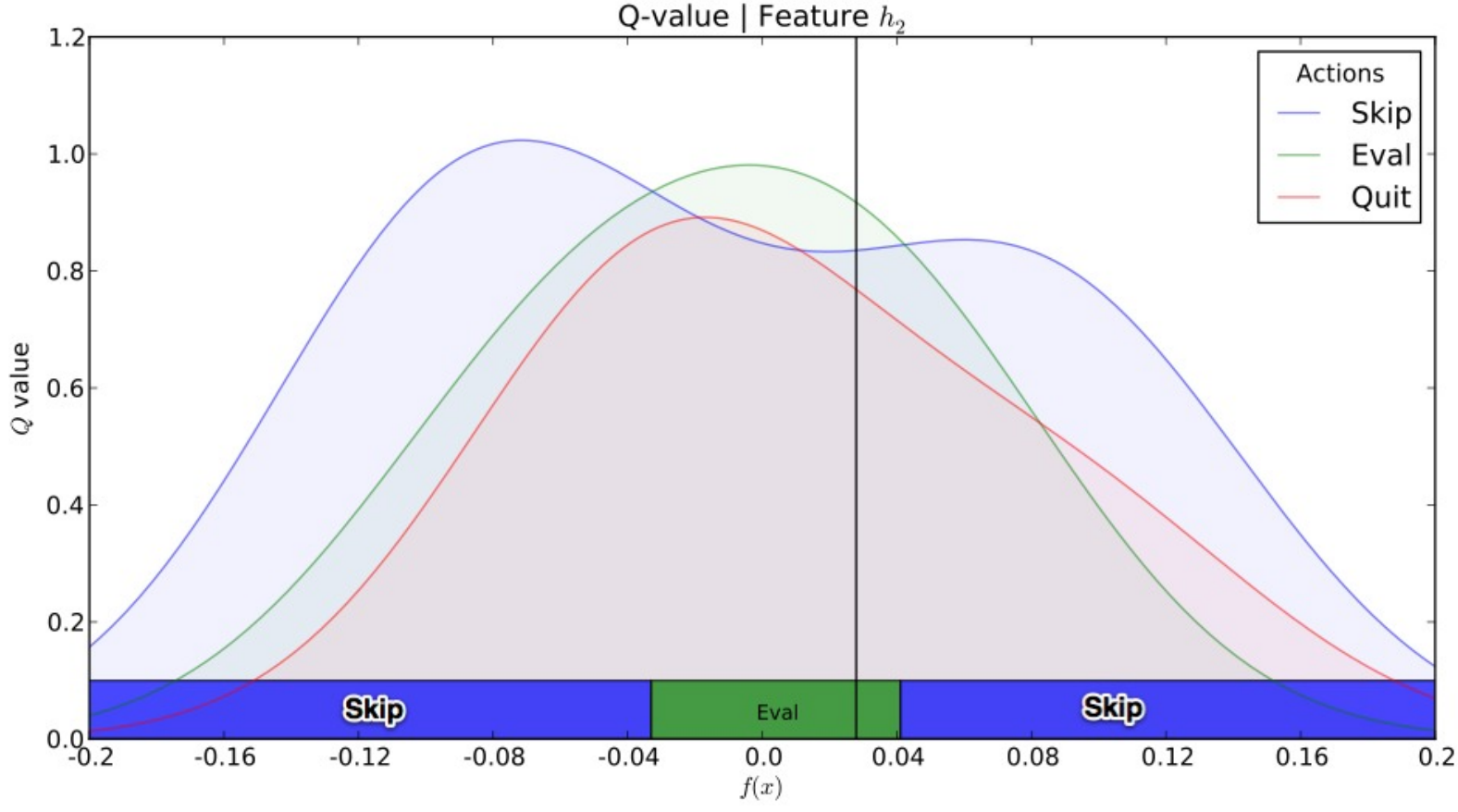

1
2
3
4
5
6
7
8
9
10
11
12
13
14
16
17
∞

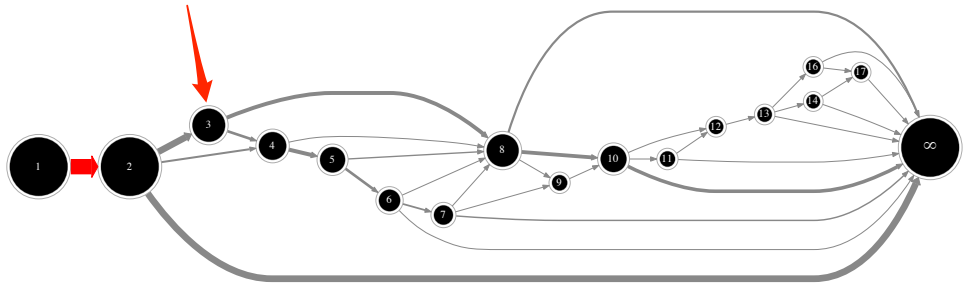

Q-value | Feature h_1
Actions
Skip
Eval
Quit
Q value
$f(x)$
1.0
0.8
0.6
0.4
0.2
0.0
-0.2
-0.16
-0.12
-0.08
-0.04
0.0
0.04
0.08
0.12
0.16
0.2

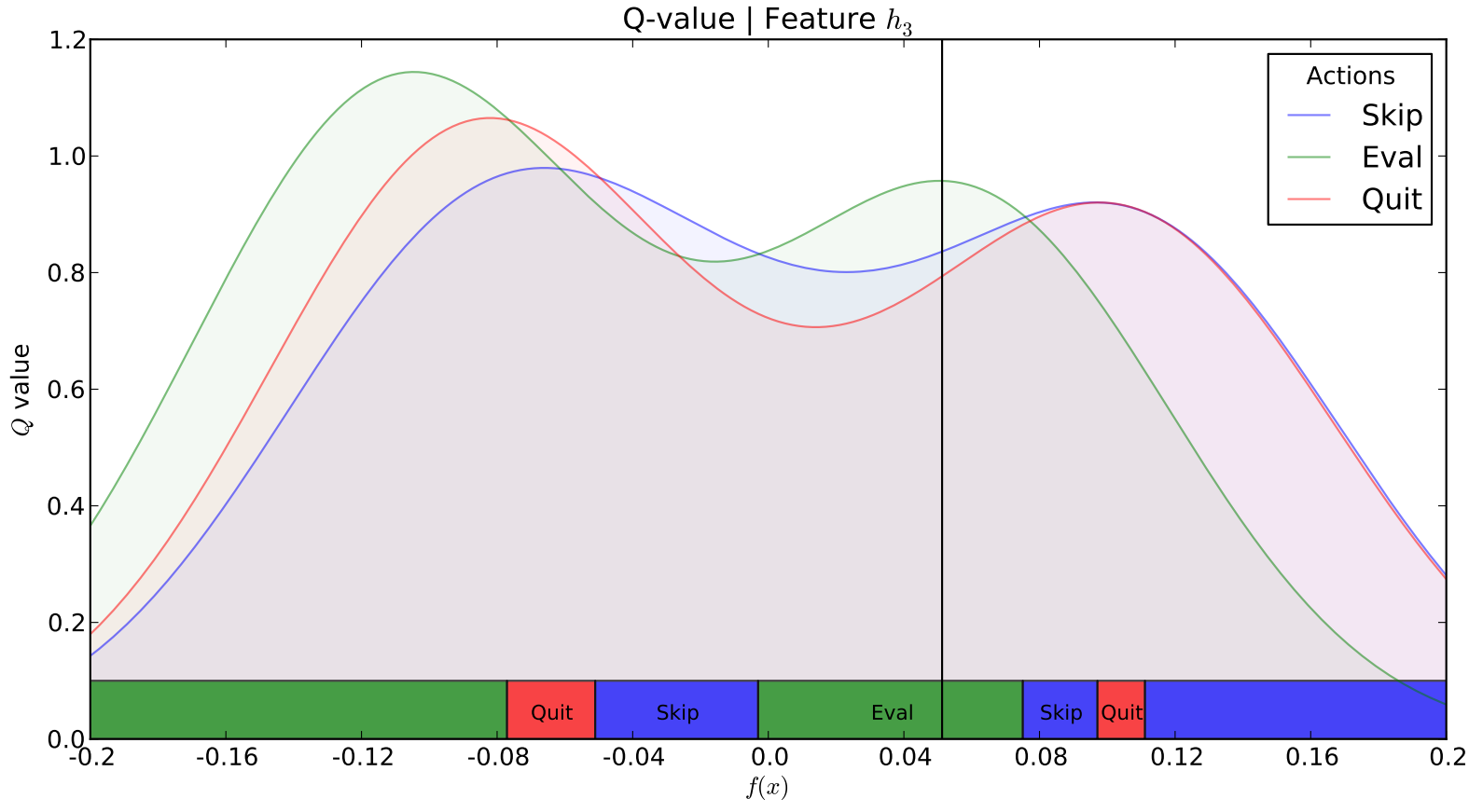

1
2
3
4
5
6
7
8
9
10
11
12
13
14
16
17
∞

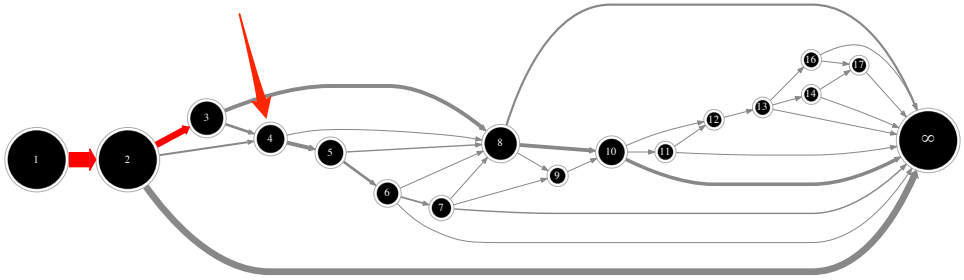

Q-value | Feature h_1
Actions
Skip
Eval
Quit
Eval
Quit
Q value
$f(x)$
1.0
0.8
0.6
0.4
0.2
0.0
-0.2
-0.16
-0.12
-0.08
-0.04
0.0
0.04
0.08
0.12
0.16
0.2

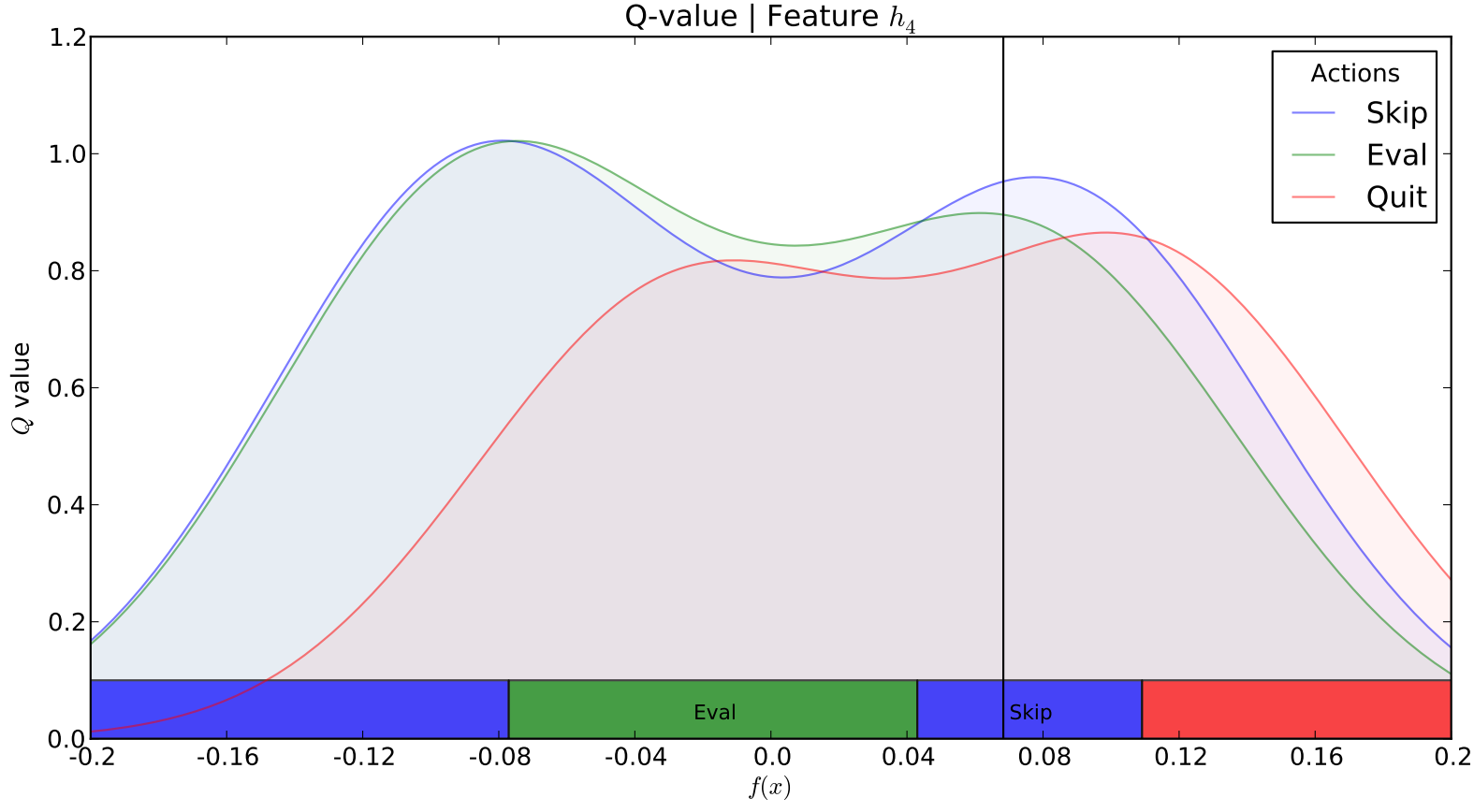

1
2
3
4
5
6
7
8
9
10
11
12
13
14
16
17
∞

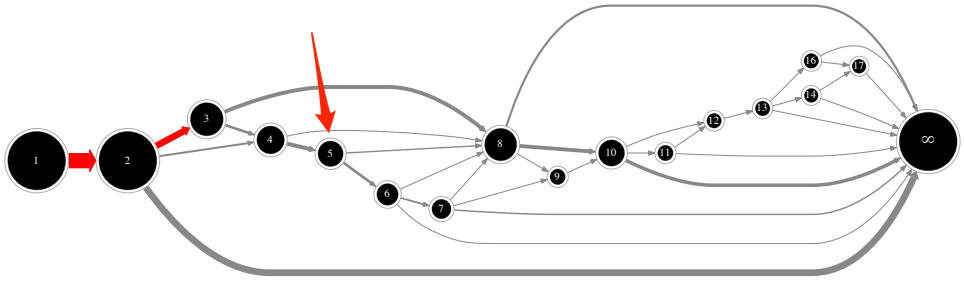

Q-value | Feature h_2
Actions
Skip
Eval
Quit
Skip
Eval
Skip
Q value
$f(x)$
1.2
1.0
0.8
0.6
0.4
0.2
0.0
-0.2
-0.16
-0.12
-0.08
-0.04
0.0
0.04
0.08
0.12
0.16
0.2

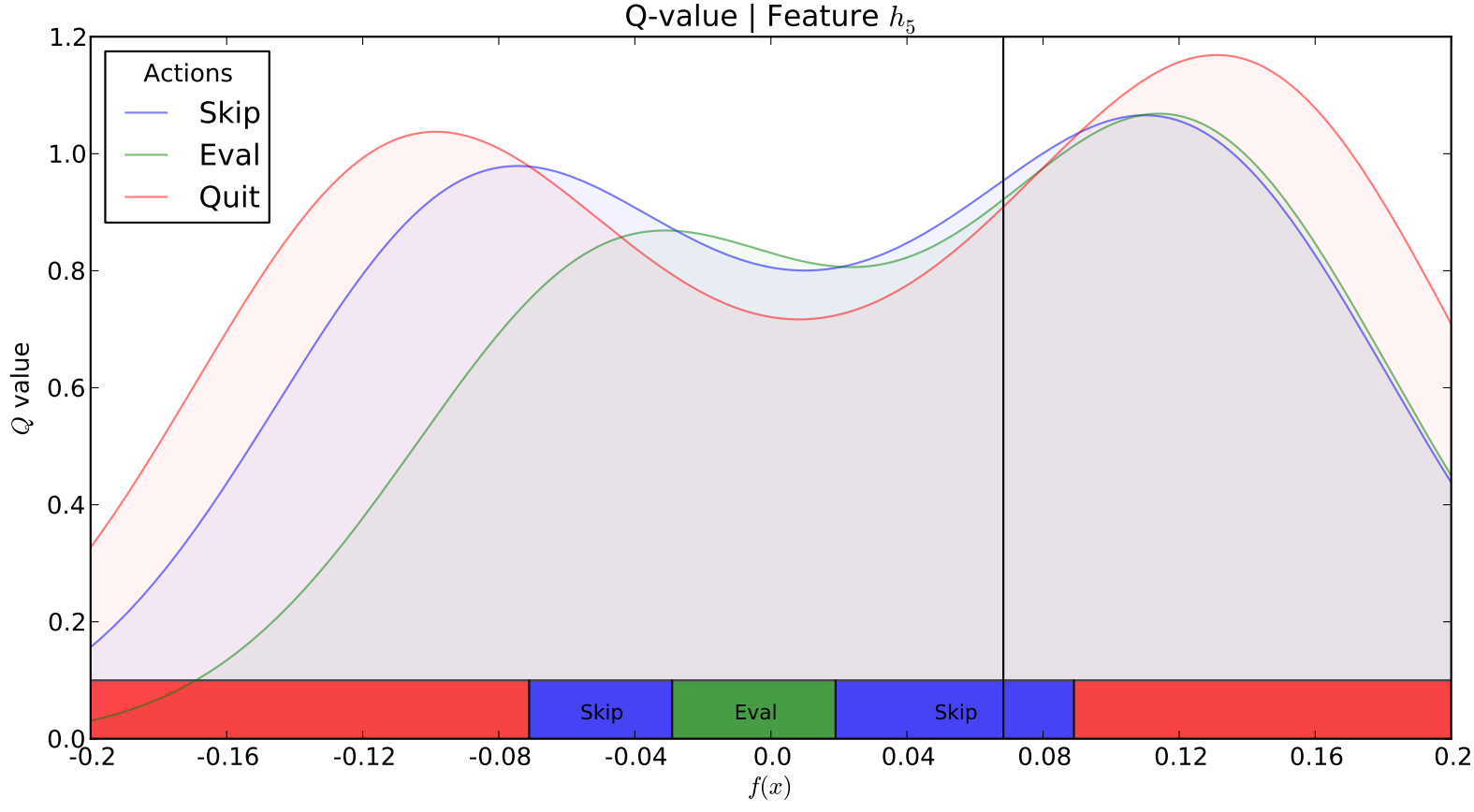

1
2
3
4
5
6
7
8
9
10
11
12
13
14
16
17
∞

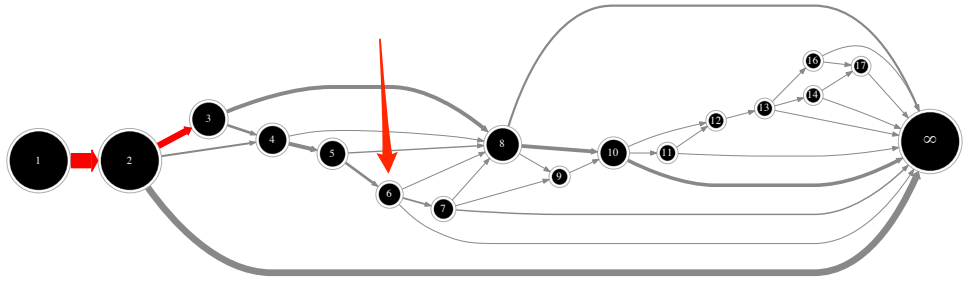

Q-value | Feature h_3
Actions
Skip
Eval
Quit
Q value
$f(x)$
1.2
1.0
0.8
0.6
0.4
0.2
0.0
-0.2
-0.16
-0.12
-0.08
-0.04
0.0
0.04
0.08
0.12
0.16
0.2
Quit
Skip
Eval
Skip
Quit

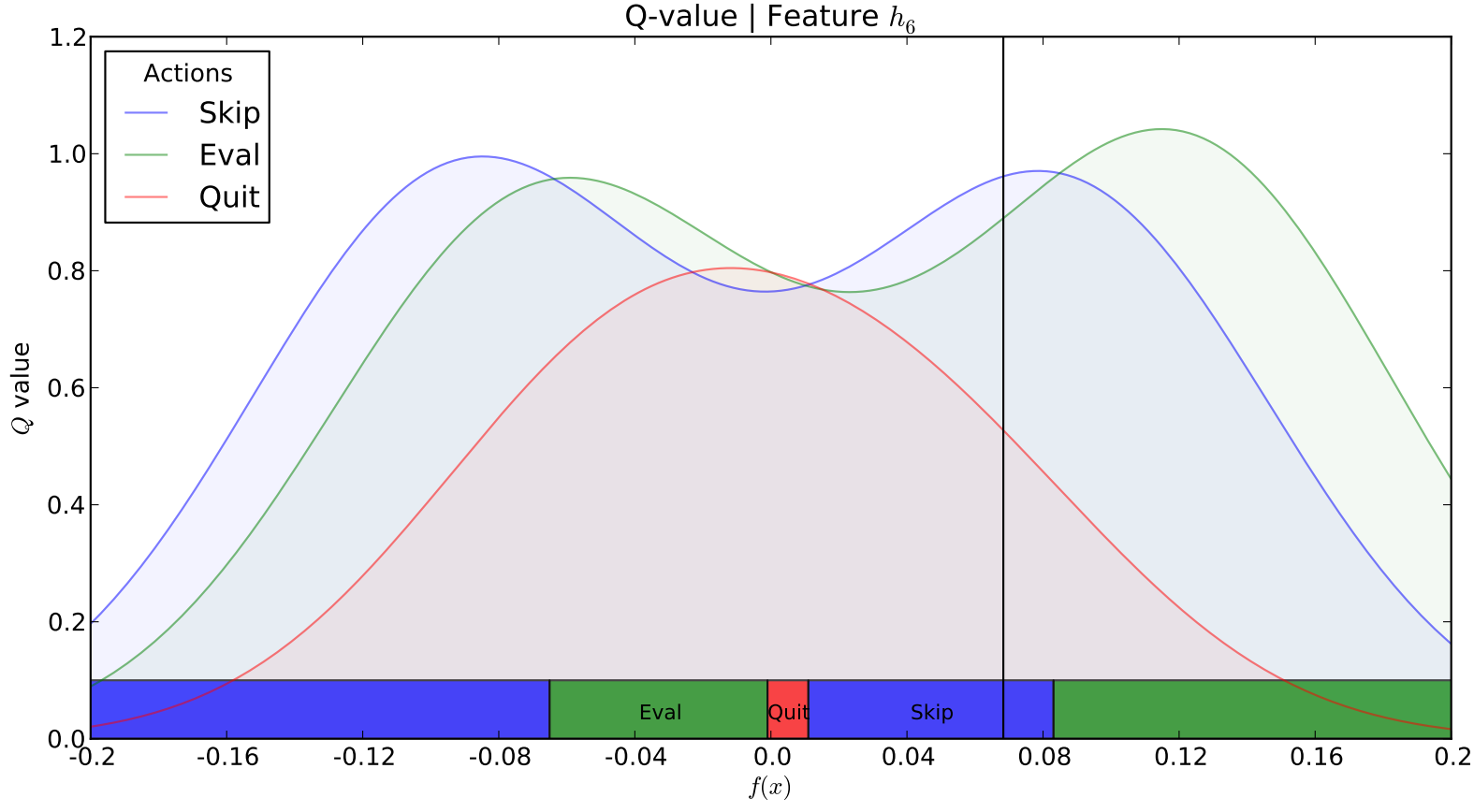

1
2
3
4
5
6
7
8
9
10
11
12
13
14
16
17
∞

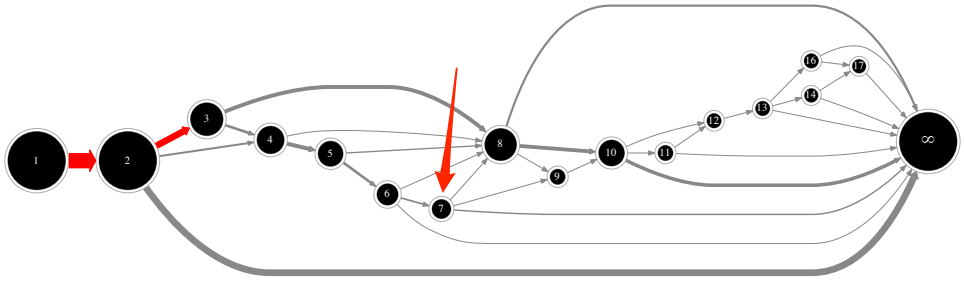

Q-value | Feature h_4
Actions
Skip
Eval
Quit
Eval
Skip
Q value
$f(x)$
1.2
1.0
0.8
0.6
0.4
0.2
0.0
-0.2
-0.16
-0.12
-0.08
-0.04
0.0
0.04
0.08
0.12
0.16
0.2

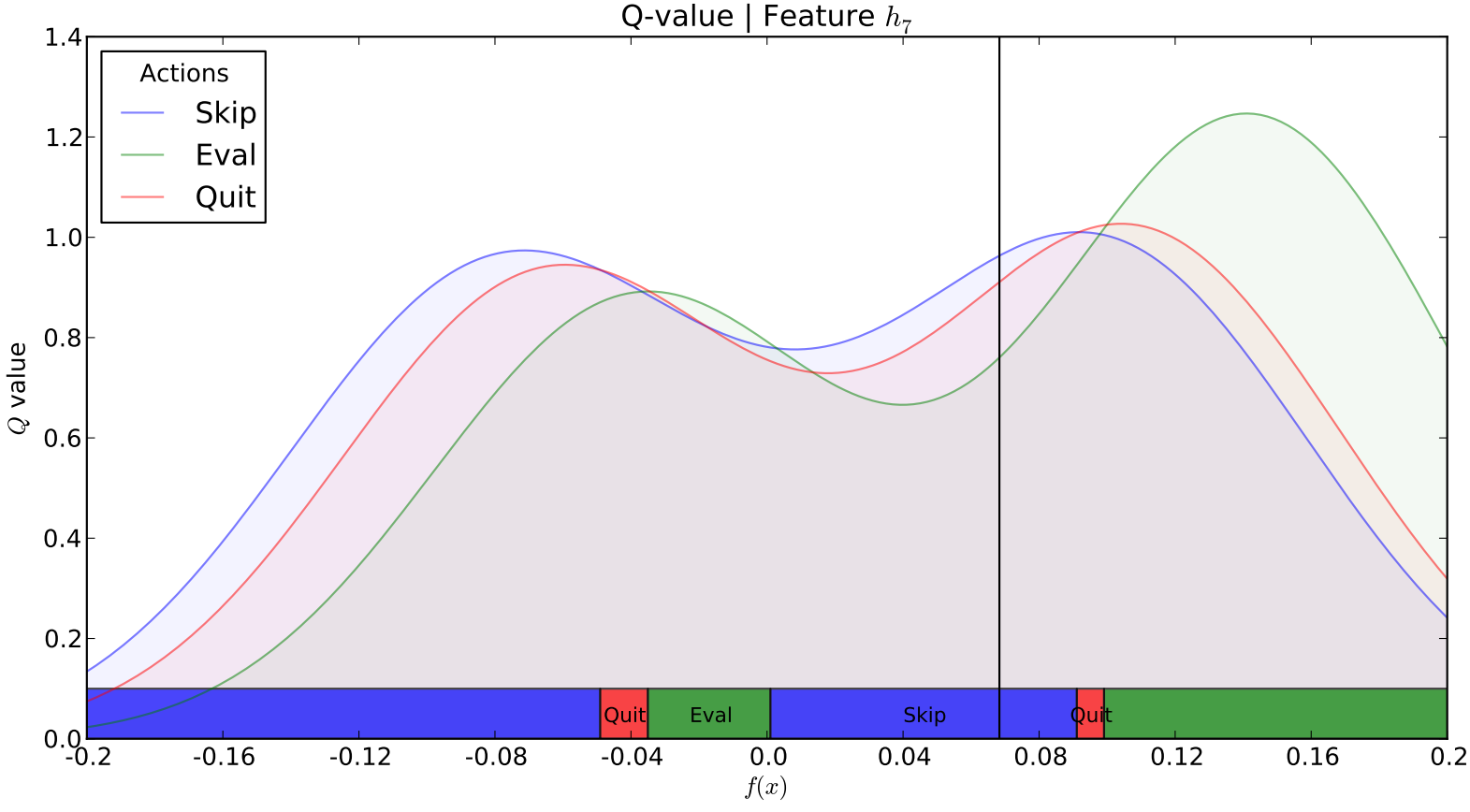

1
2
3
4
5
6
7
8
9
10
11
12
13
14
16
17
∞

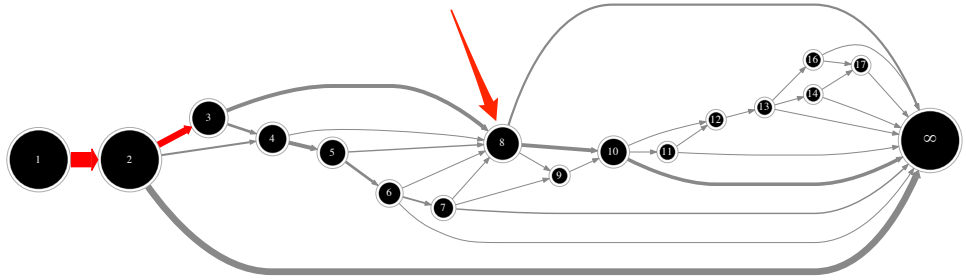

Q-value | Feature h_5
Actions
Skip
Eval
Quit
Skip
Eval
Skip
Q value
$f(x)$
1.2
1.0
0.8
0.6
0.4
0.2
0.0
-0.2
-0.16
-0.12
-0.08
-0.04
0.0
0.04
0.08
0.12
0.16
0.2

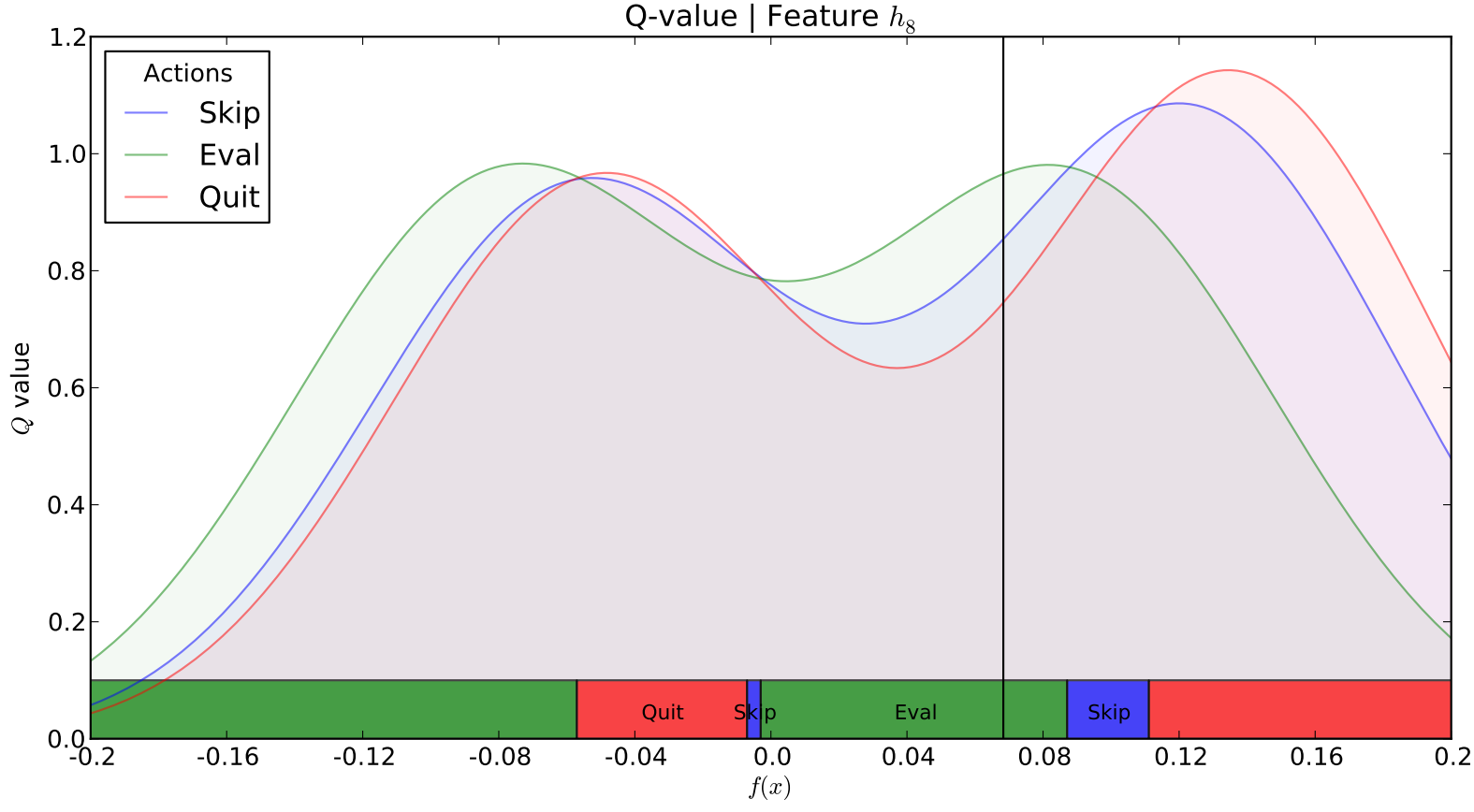

1
2
3
4
5
6
7
8
9
10
11
12
13
14
16
17
∞

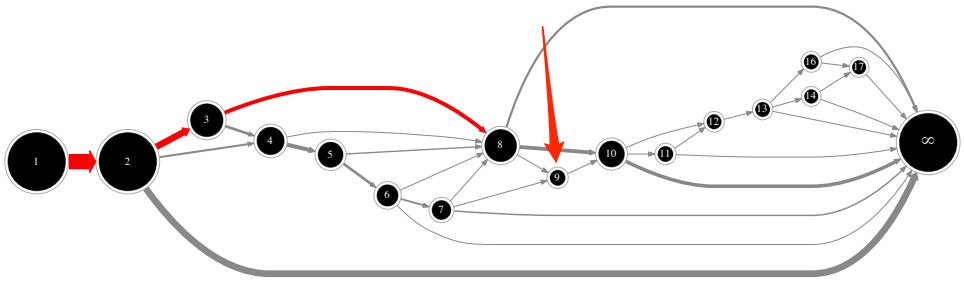

Q-value | Feature h_6
Actions
Skip
Eval
Quit
Q value
1.2
1.0
0.8
0.6
0.4
0.2
0.0
Eval
Quit
Skip
-0.2
-0.16
-0.12
-0.08
-0.04
0.0
0.04
0.08
0.12
0.16
0.2
$f(x)$

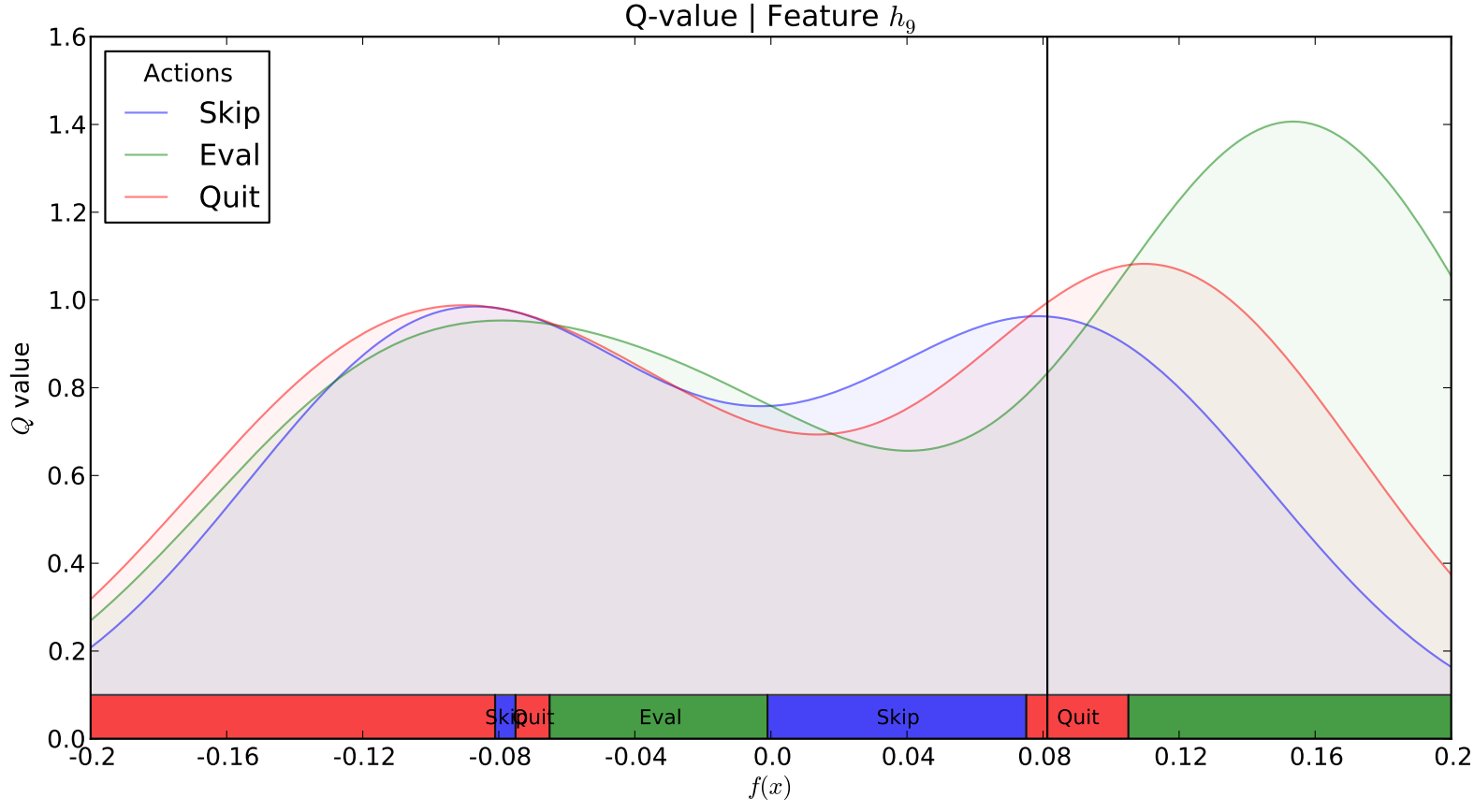

1
2
3
4
5
6
7
8
9
10
11
12
13
14
16
17
∞

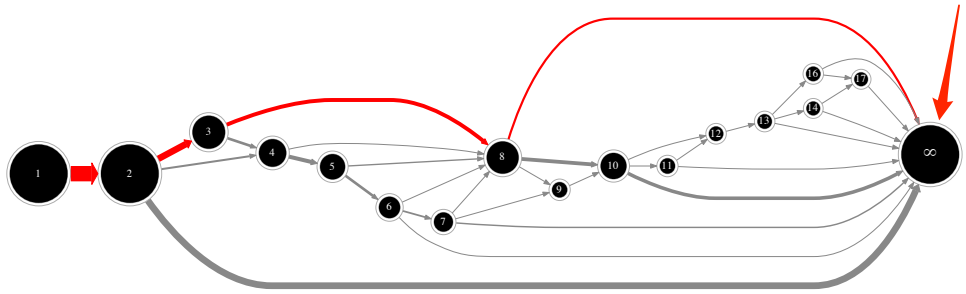

Q-value | Feature h_7
Actions
Skip
Eval
Quit
Q value
1.4
1.2
1.0
0.8
0.6
0.4
0.2
0.0
-0.2
-0.16
-0.12
-0.08
-0.04
0.0
0.04
0.08
0.12
0.16
0.2
$f(x)$
Quit
Eval
Skip
Quit

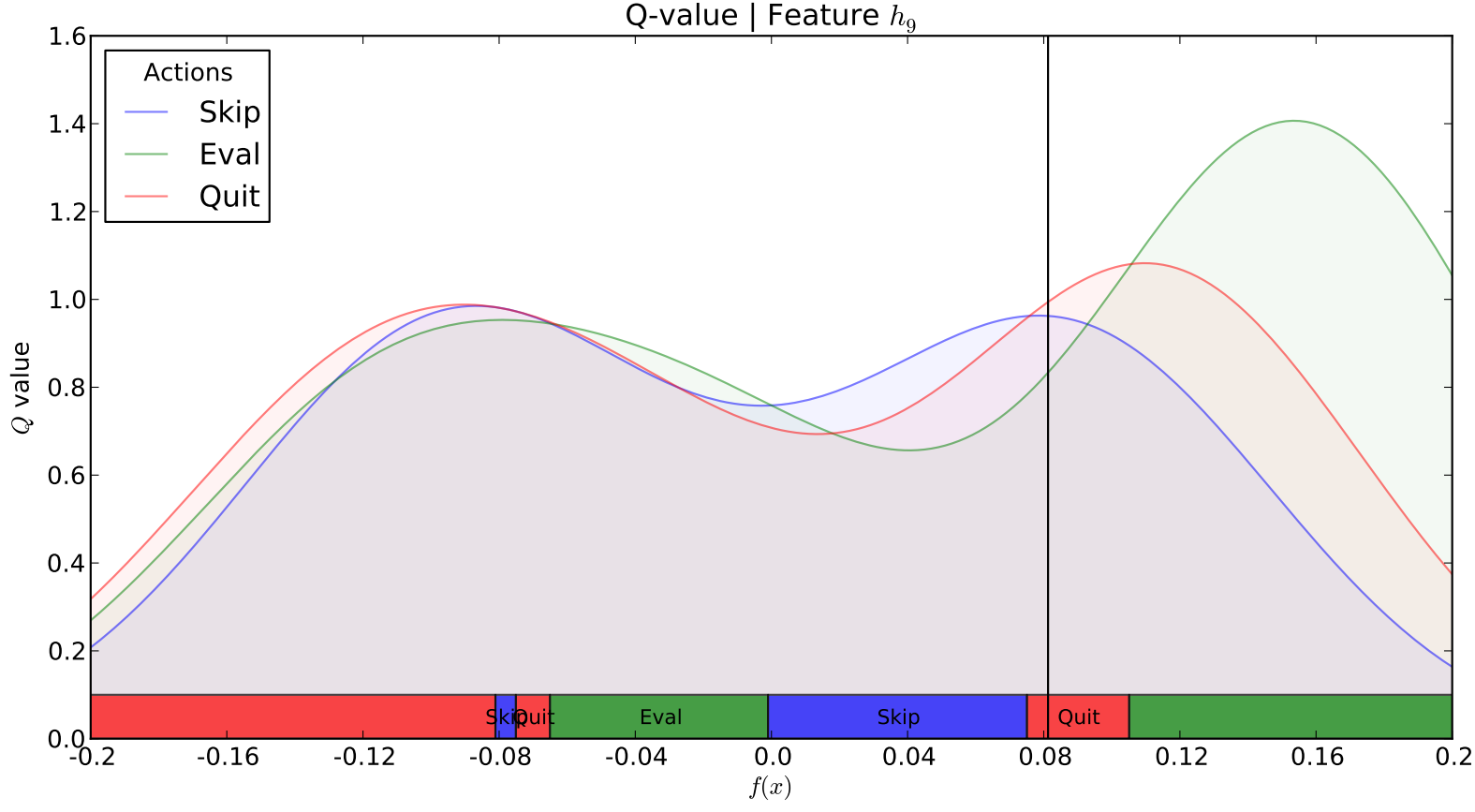

1
2
3
4
5
6
7
8
9
10
11
12
13
14
16
17
∞


Q-value | Feature h_8
Actions
Skip
Eval
Quit
Q value
$f(x)$
1.2
1.0
0.8
0.6
0.4
0.2
0.0
-0.2
-0.16
-0.12
-0.08
-0.04
0.0
0.04
0.08
0.12
0.16
0.2
Quit
Skip
Eval
Skip


1
2
3
4
5
6
7
8
9
10
11
12
13
14
16
17
∞

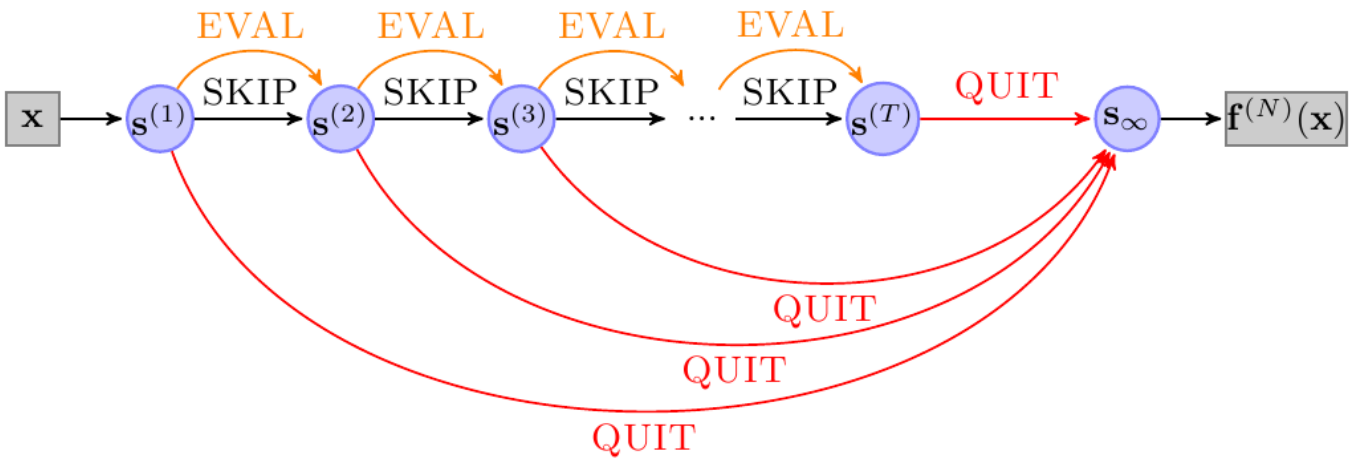

Q-value | Feature h_9
Actions
Skip
Eval
Quit
Q value
$f(x)$
Skip
Quit
Eval
Skip
Quit
1.6
1.4
1.2
1.0
0.8
0.6
0.4
0.2
0.0
-0.2
-0.16
-0.12
-0.08
-0.04
0.0
0.04
0.08
0.12
0.16
0.2


1
2
3
4
5
6
7
8
9
10
11
12
13
14
16
17
∞


Q-value | Feature h_9
Actions
Skip
Eval
Quit
Q value
$f(x)$
Skip
Quit
Eval
Skip
Quit
1.6
1.4
1.2
1.0
0.8
0.6
0.4
0.2
0.0
-0.2
-0.16
-0.12
-0.08
-0.04
0.0
0.04
0.08
0.12
0.16
0.2

MDDAG : The setup

The setup



Given Sequence of K-class base classifiers $(\mathbf{h}_1, \ldots, \mathbf{h}_N)$,

$$\mathbf{h}_j : \mathcal{X} \to \mathbb{R}^K, j = 1, \ldots, N$$

Goal Learn an agent π which takes decisions $\{\text{EVAL}, \text{SKIP}, \text{QUIT}\}$

Learning Episodic undiscounted Markov Decision Process

What the agent knows... (1)

The score function

$$\mathbf{f}^{(j)}(\mathbf{x}) = \sum_{j'=1}^{j} b_{j'}(\mathbf{x})\ \mathbf{h}_{j'}(\mathbf{x}) \in \mathbb{R}^K$$

$$b_j(\mathbf{x}) = \begin{cases} 1 & \text{if the feature } j \text{ was evaluated} \\ 0 & \text{otherwise} \end{cases}$$

Two top ranked labels

$$\ell_1^{(j)}(\mathbf{x}) = \arg\max_{\ell} \mathbf{f}^{(j-1)}(\mathbf{x})$$

$$\ell_2^{(j)}(\mathbf{x}) = \arg\max_{\ell, \ell \neq \ell_1^{(j)}(\mathbf{x})} \mathbf{f}^{(j-1)}(\mathbf{x})$$

Their score difference

$$\Delta^{(j)}(\mathbf{x}) = \max_{\ell} \mathbf{f}^{(j-1)(\mathbf{x})} - \max_{\ell, \ell \neq \ell_1^{(j)}(\mathbf{x})} \mathbf{f}^{(j-1)(\mathbf{x})}$$

What the agent knows... (2)

For a given instance

$$\Big(\underbrace{j}_{\substack{\text{base classifier} \\ \text{index}}} , \underbrace{(\ell_1^{(j)}, \ell_2^{(j)})}_{\text{winning labels}} , \underbrace{\Delta^{(j)}}_{\substack{\text{score} \\ \text{difference}}} \Big)$$

What the agent knows... (2)

For a given instance

$$\Big(\underbrace{j}_{\substack{\text{base classifier} \\ \text{index}}} , \underbrace{(\ell_1^{(j)}, \ell_2^{(j)})}_{\substack{\text{winning} \\ \text{labels}}} , \underbrace{\Delta^{(j)}}_{\substack{\text{score} \\ \text{difference}}} \Big)$$

The agent π

$$\pi\Big(\underbrace{j, \quad (\ell_1^{(j)}, \ell_2^{(j)}), \quad \Delta^{(j)}}_{\text{state descriptor}}\Big) \mapsto \{\underbrace{\textsc{Eval}, \textsc{Skip}, \textsc{Quit}}_{\text{actions}}\}$$

Learning from interaction

- The agent takes actions and receives rewards.
- QUIT action reward $\propto L(\mathbf{f}, (\mathbf{x}, \ell))$
- Penalize EVAL action : $r_t = -\beta, \quad 0 < \beta < 1$
- Control of the accuracy / complexity trade-off.
- The agent maximizes $\varrho = \mathbb{E}\left\{\sum_{t=1}^{T} r^{(t)}\right\}, \quad r_t \in \mathbb{R}$

Objective function

$$\varrho = \mathbb{E}_{(\mathbf{x},\ell)\sim\mathfrak{D}}\left\{\underbrace{-L\big(\mathbf{f}, (\mathbf{x}, \ell)\big)}_{error} \underbrace{- \beta \sum_{j=1}^{N} b_j(\mathbf{x})}_{L_0 \text{ penalty}}\right\}.$$

What motivates the agent

Multi-class 0-1 loss

$$L_{\mathbb{I}}(\mathbf{f}, (\mathbf{x}, \ell)) = \mathbb{I}\left\{ f_\ell(\mathbf{x}) - \max_{\ell' \neq \ell} f_{\ell'}(\mathbf{x}) < 0 \right\}$$

Multi-class exponential loss

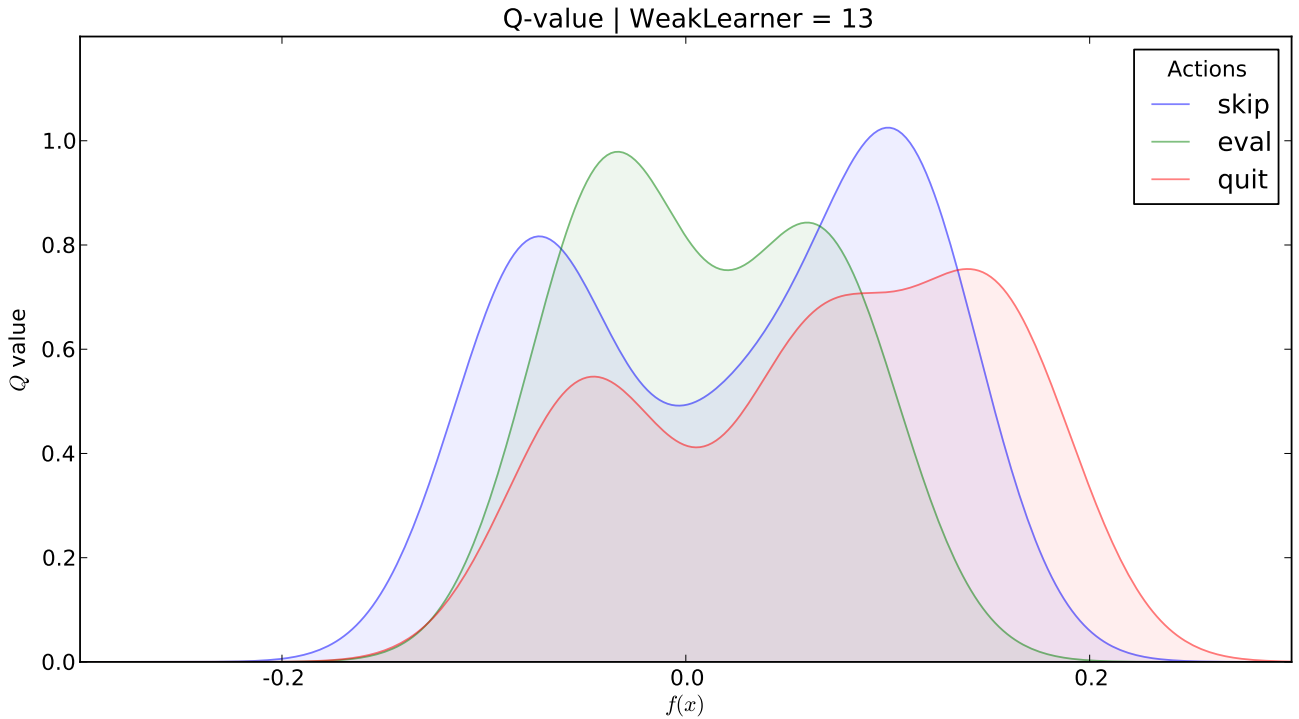
$$L_{\mathsf{EXP}}(\mathbf{f}, (\mathbf{x}, \ell)) = \exp\left(\sum_{\ell' \neq \ell}^{K} f_{\ell'}(\mathbf{x}) - f_\ell(\mathbf{x}) \right),$$

Feature costs by varying β

Asymmetric classification

What the agent learns

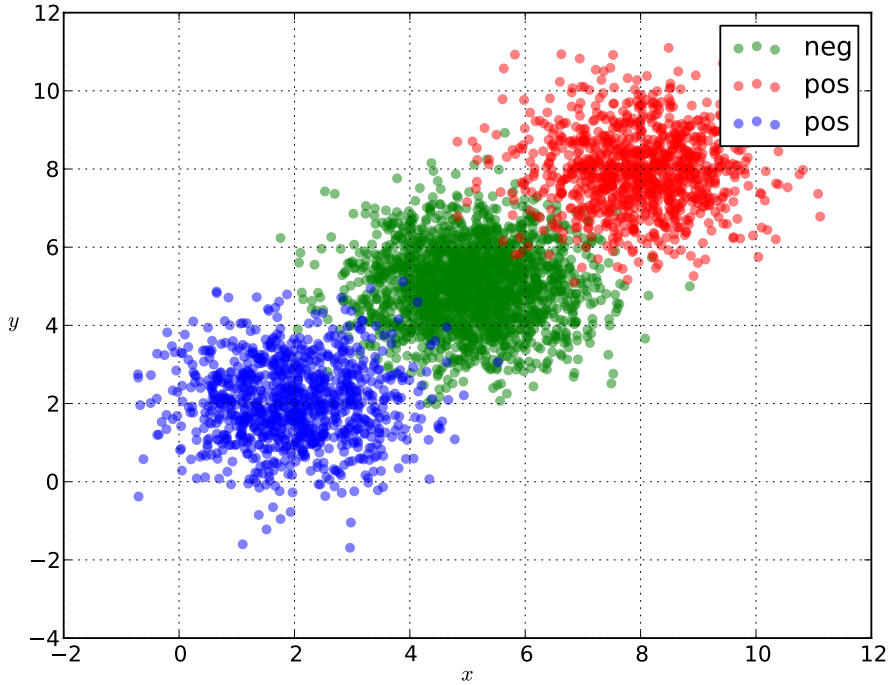
Estimate the value of taking a given action in a given state



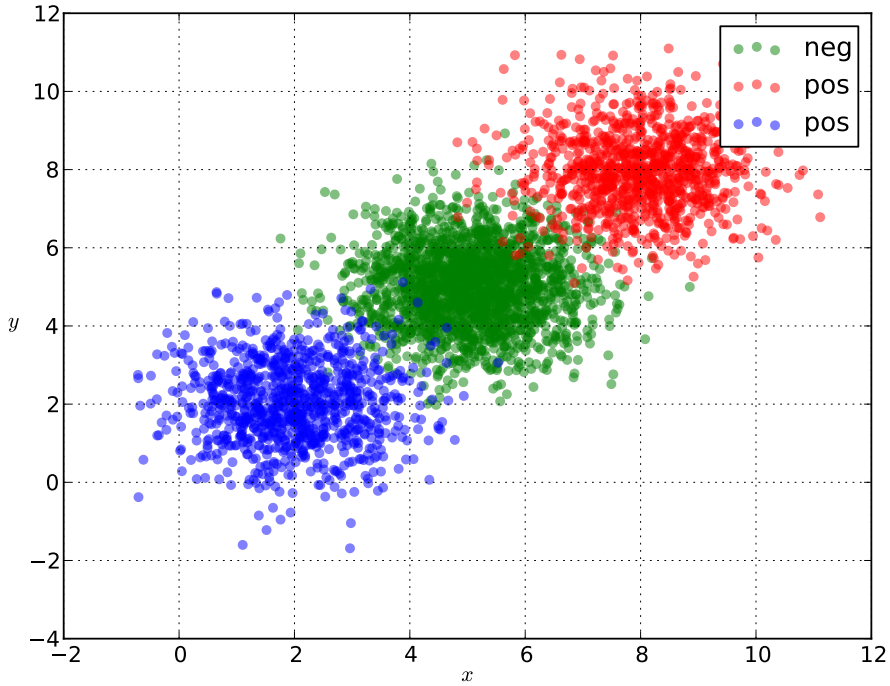
- Radial Basis Functions
- Simple discretization

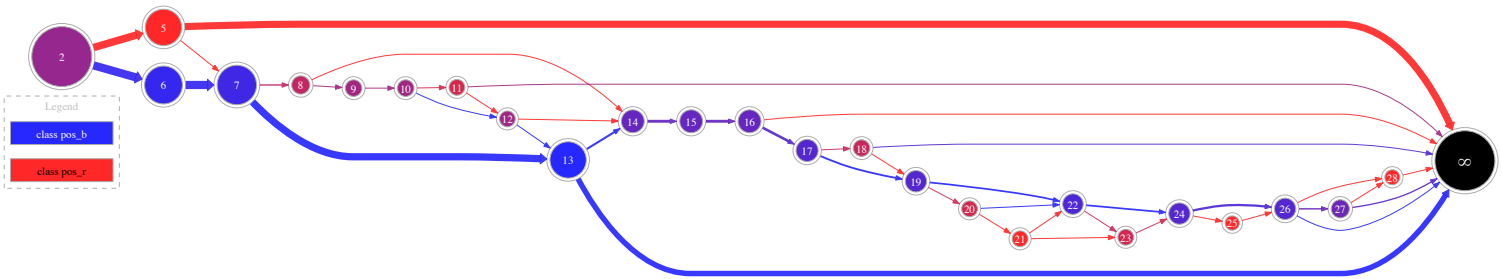
Data-dependent classification

A toy example

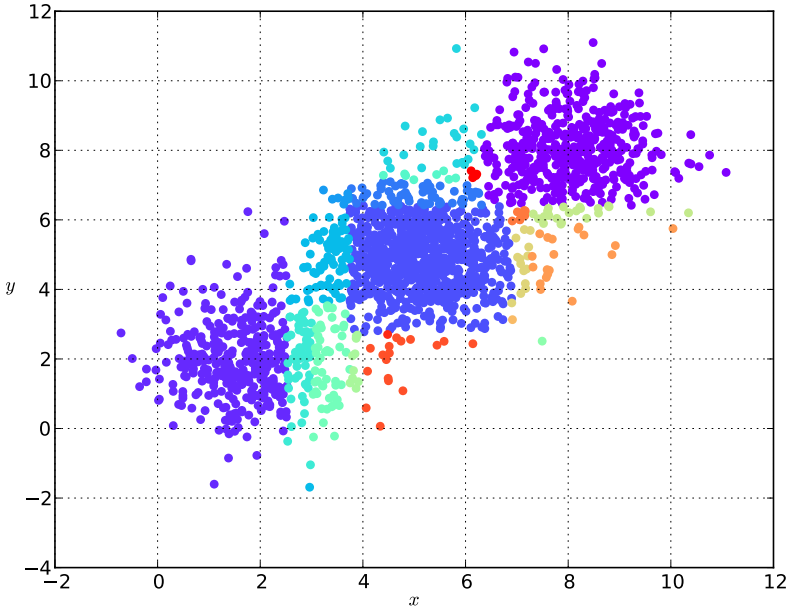


A toy example

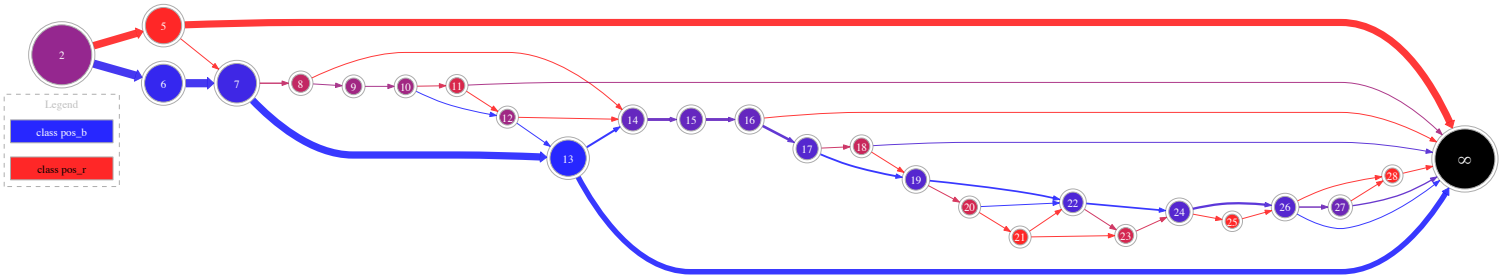




A toy example



- 2,5 (454)
- 2,6,7,13 (357)
- 2,6,7,8,9,10,11 (800)
- 2,5,7,8,9,10,11 (64)
- 2,5,7,13,14,15,16,17,18 (8)
- 2,6,7,13,14,15,16,17,18 (81)
- 2,5,7,8,9,10,11,12,14,15,16 (25)
- 2,6,7,13,14,15,16,17,19,22,24,26 (54)
- 2,5,7,8,9,10,11,12,14,15,16,17,18 (8)
- 2,6,7,13,14,15,16,17,19,22,24,26,27 (52)
- 2,6,7,8,14,15,16,17,19,20,21,22,24,26,27 (1)
- 2,6,7,8,9,10,12,13,14,15,16,17,19,22,24,26,27 (10)
- 2,6,7,8,14,15,16,17,18,19,20,21,22,23,24,26,27 (22)
- 2,6,7,8,14,15,16,17,18,19,20,21,23,24,25,26,28 (14)
- 2,6,7,8,14,15,16,17,19,20,21,23,24,25,26,27,28 (1)
- 2,6,7,8,14,15,16,17,18,19,20,21,22,23,24,26,27,28 (19)
- 2,6,7,8,14,15,16,17,18,19,20,21,23,24,25,26,27,28 (8)
- 2,6,7,8,9,10,12,13,14,15,16,17,19,20,22,23,24,26,27 (17)
- 2,6,7,8,9,10,12,13,14,15,16,17,19,20,22,23,24,26,27,28 (1)
- 2,5,7,8,9,10,11,12,14,15,16,17,18,19,20,21,23,24,25,26,27,28 (4)

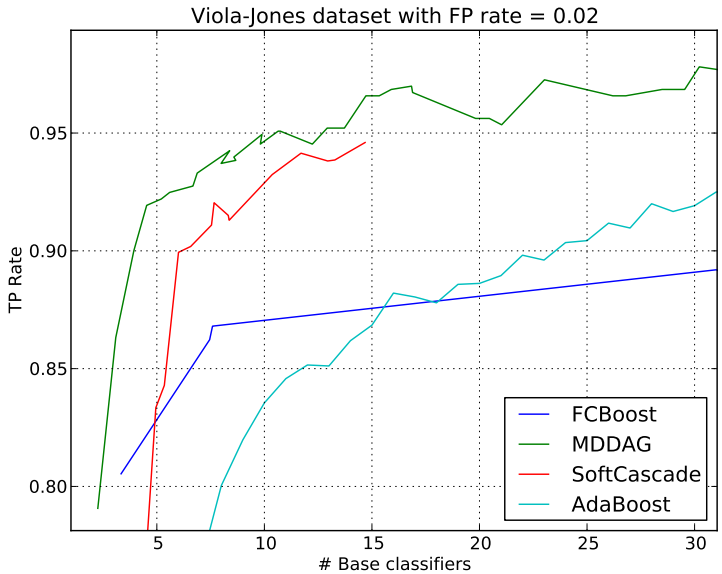


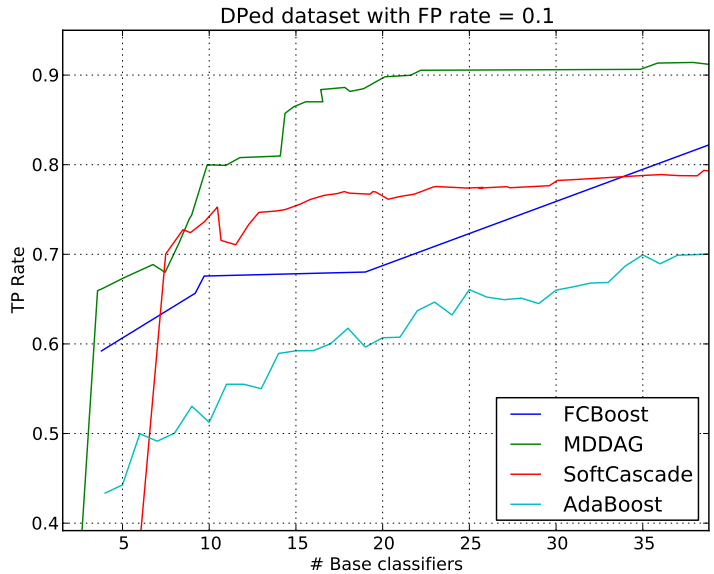
Related work

- Póczos et. al. (ICML, 2009)
- Gao and Koller (NIPS, 2011)
- Dulac-Arnold et. al. (Machine Learning, 2012)

Benchmarks

Object detection benchmarks





LHCb Toy Data (1)

Test set :

2body	1674 candidates, 1674 events
3body	4892 candidates, 1968 events
4body	9390 candidates, 1945 events
bkgd	1635092 candidates, 43252 events

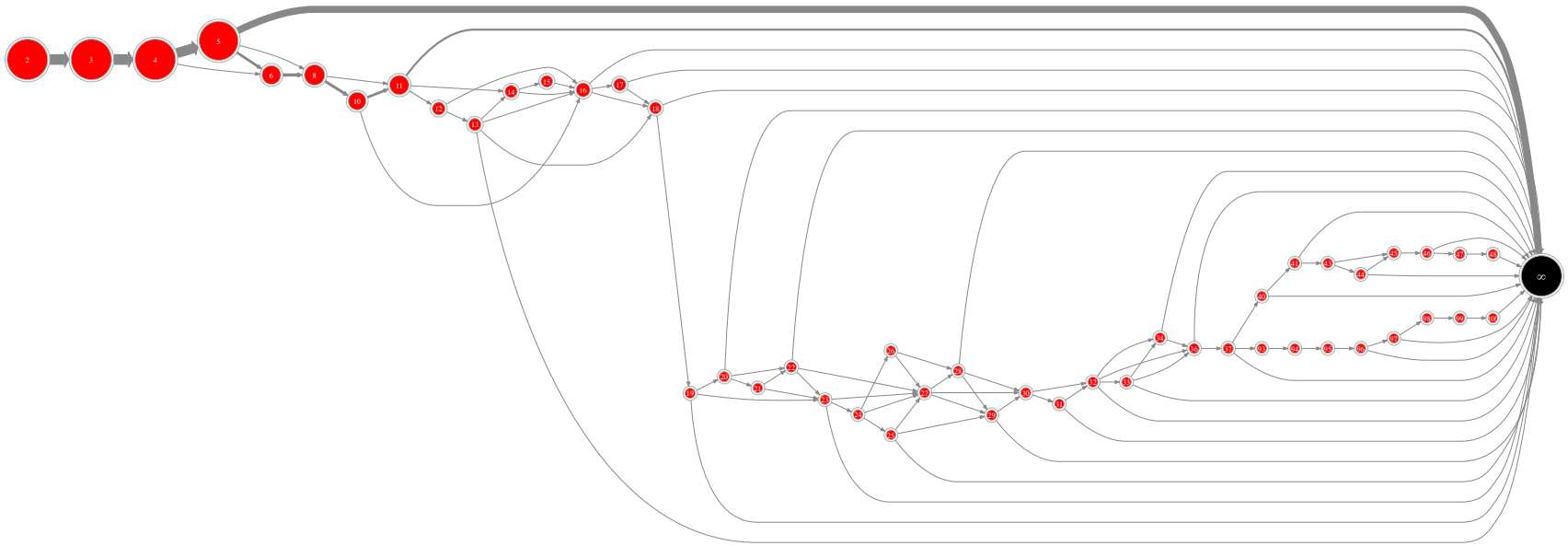
Average evaluation cost : 5.418

2body	18.092
3body	13.551
4body	11.357
bkgd	5.389

Error :

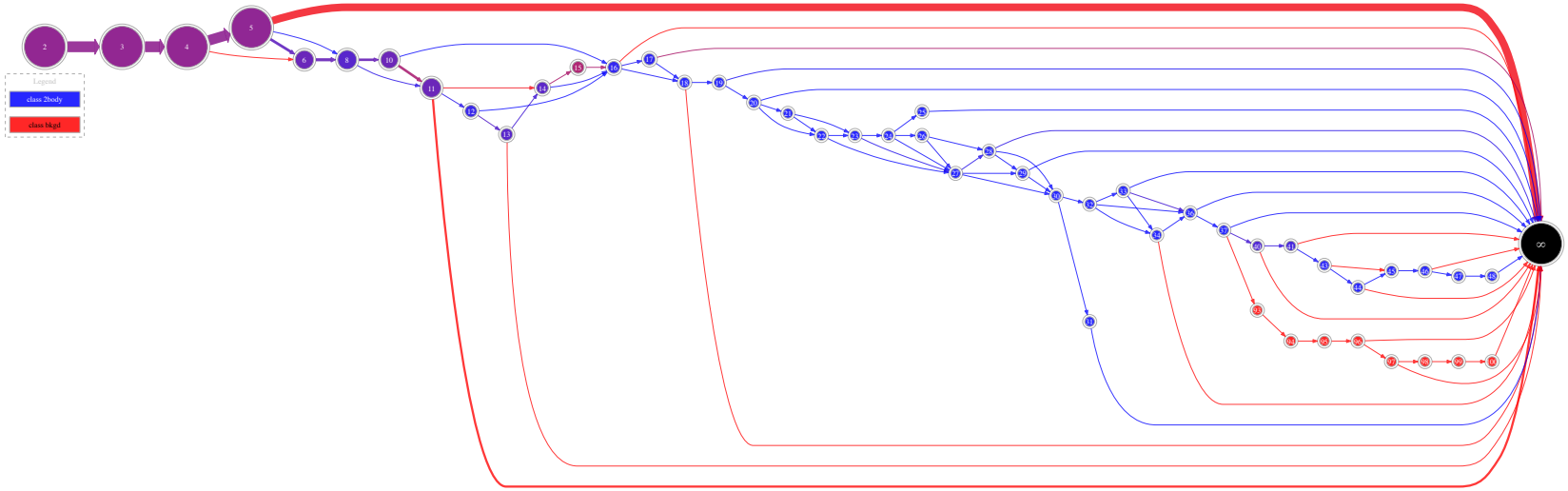
- Adaboost – 6 base classifiers : 0.09523
- MDDAG – 5.42 base classifiers : 0.0456
- Adaboost – 100 base classifiers : 0.03858

LHCb Toy Data (2)

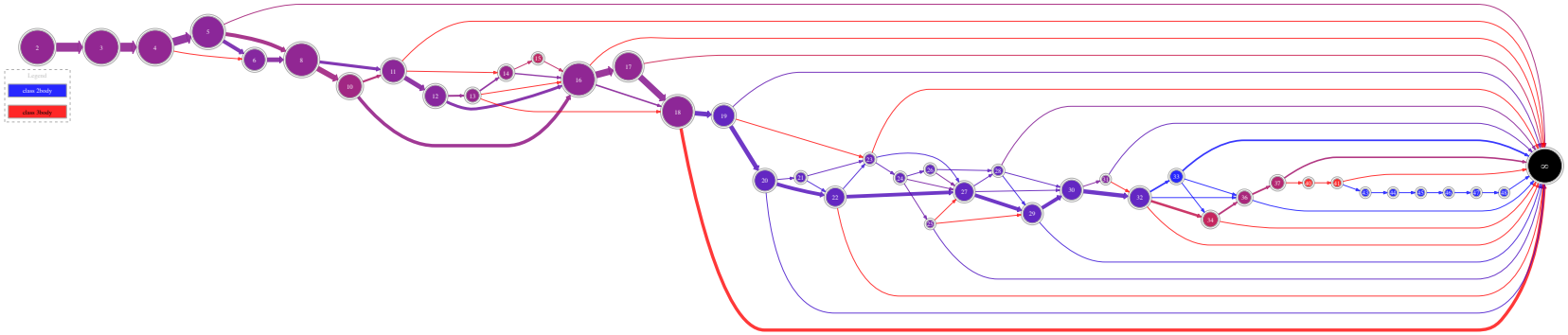


LHCb Toy Data (2)

2body vs background



2body vs 3body



Conclusion

Conclusion and next steps

- A sparse classifier of the form of a Directed Acyclic Graph.
- Adapt to many (multi-class) loss functions.
- Data-dependent classification.

Conclusion and next steps

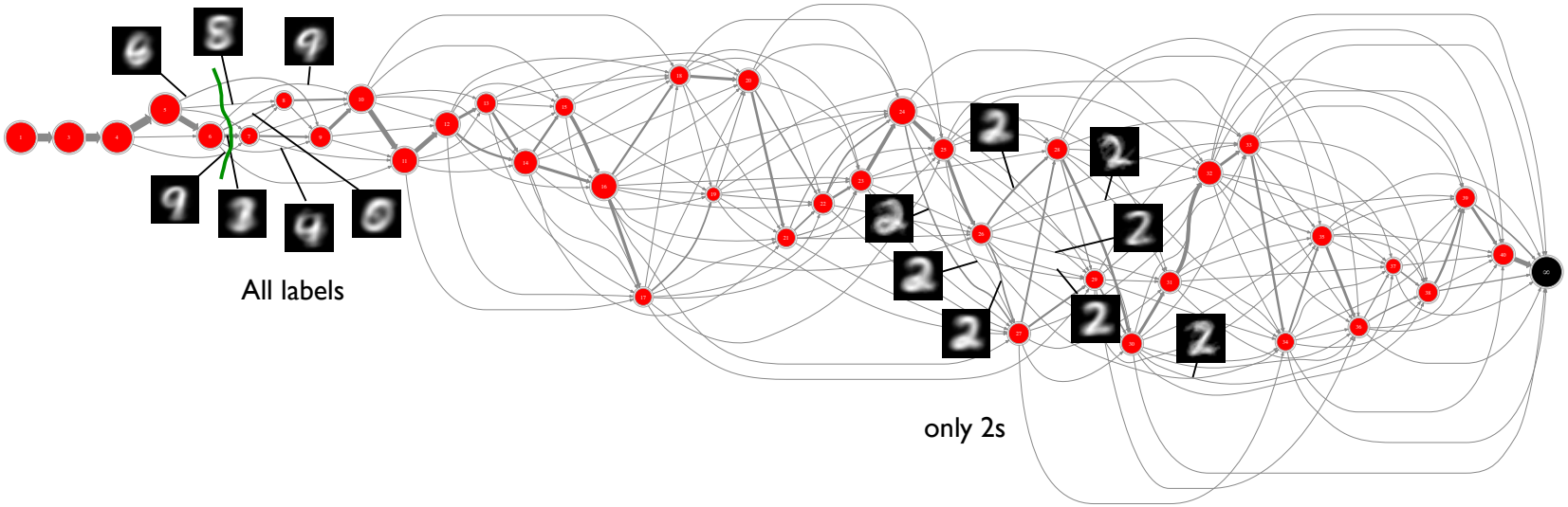
- A sparse classifier of the form of a Directed Acyclic Graph.
- Adapt to many (multi-class) loss functions.
- Data-dependent classification.

- Next : go further with triggers
- Features have acquisition cost
- Learn to manage the cost budget

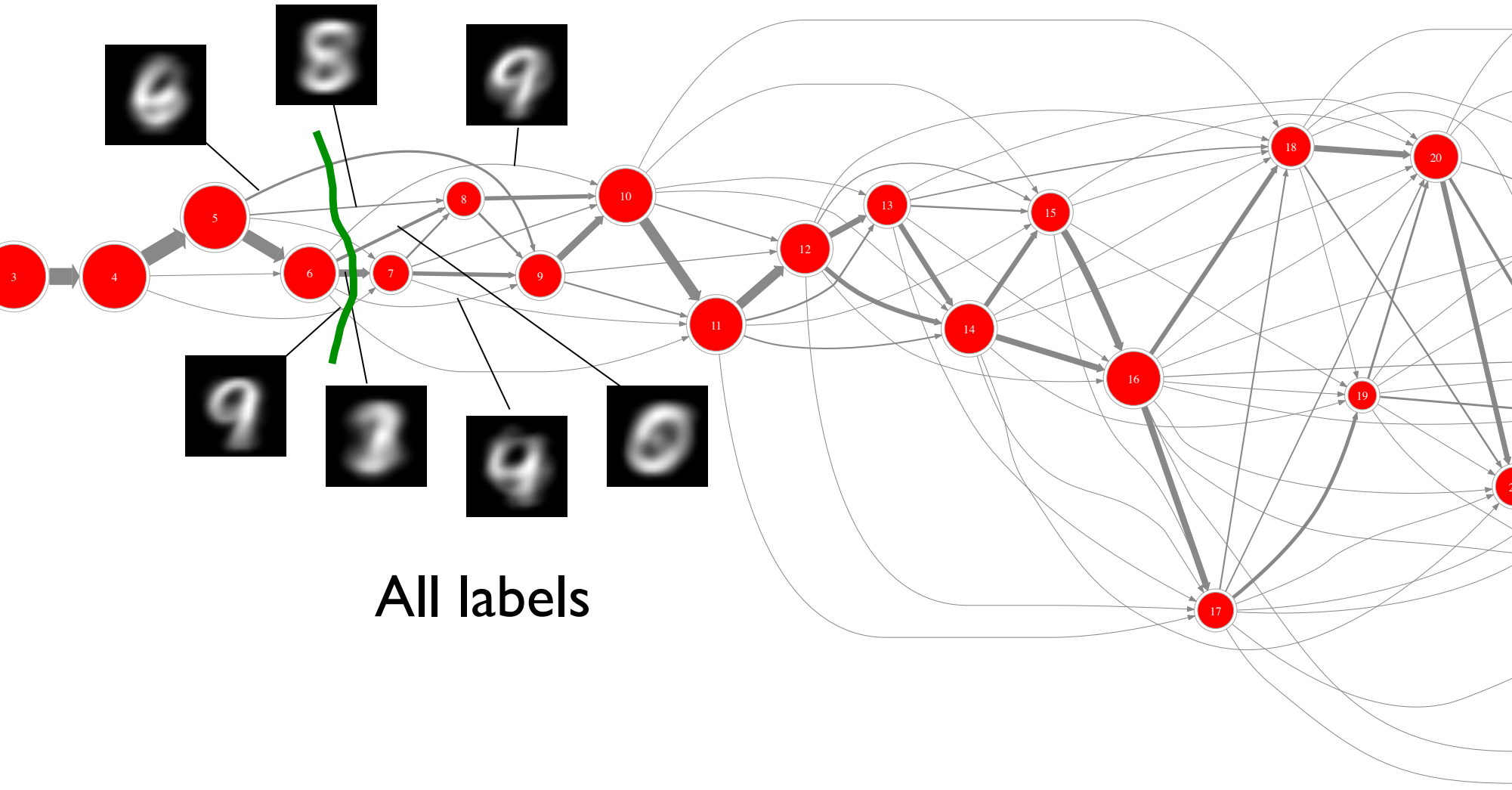
Thank you

Path clustering

- MNIST



Path clustering



All labels

Path clustering