UQÀM

DEEL
DEpendable & Explainable Learning

PROBONITE : PRivate One-Branch-Only Non Interactive decision Tree Evaluation

Sofiane Azogagh, Victor Delfour, Sébastien Gambs and Marc-Olivier Killijian - UQÀM

WAHC 2022 | November 07, 2022

Summary

- Introduction
- State-of-the-art
- Preliminaries
    - Functional Bootstrapping
    - Private Information Retrieval
- Our proposal
- Conclusion and perspectives

Introduction

Introduction

Machine Learning as a Service (MLaaS)

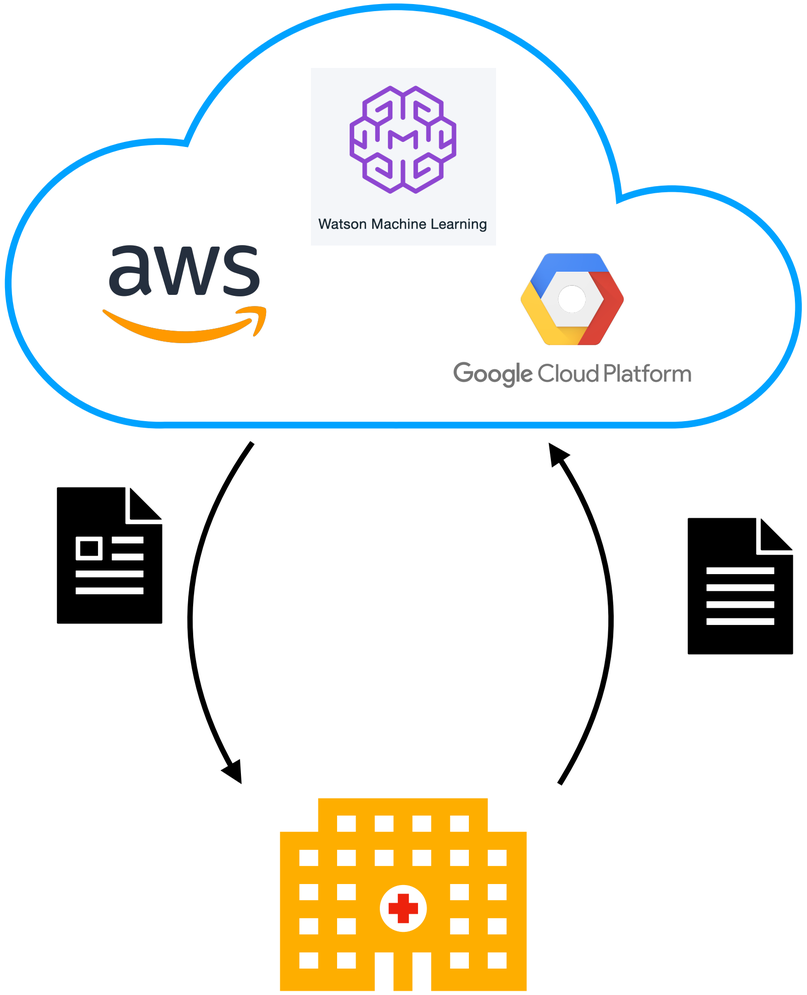
Exists in many platforms

Uses private or personal information

During the training phase or at inference time

Introduction

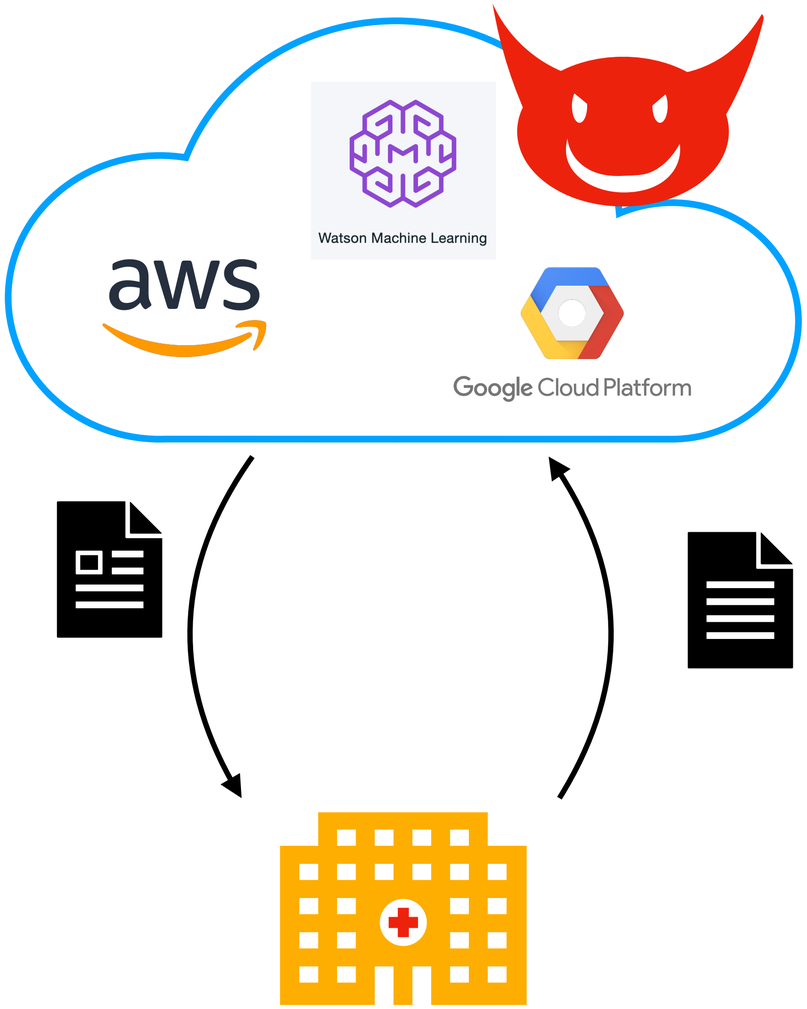
Motivation



Introduction

Motivation

If the cloud is compromised, some private information of the client will leak

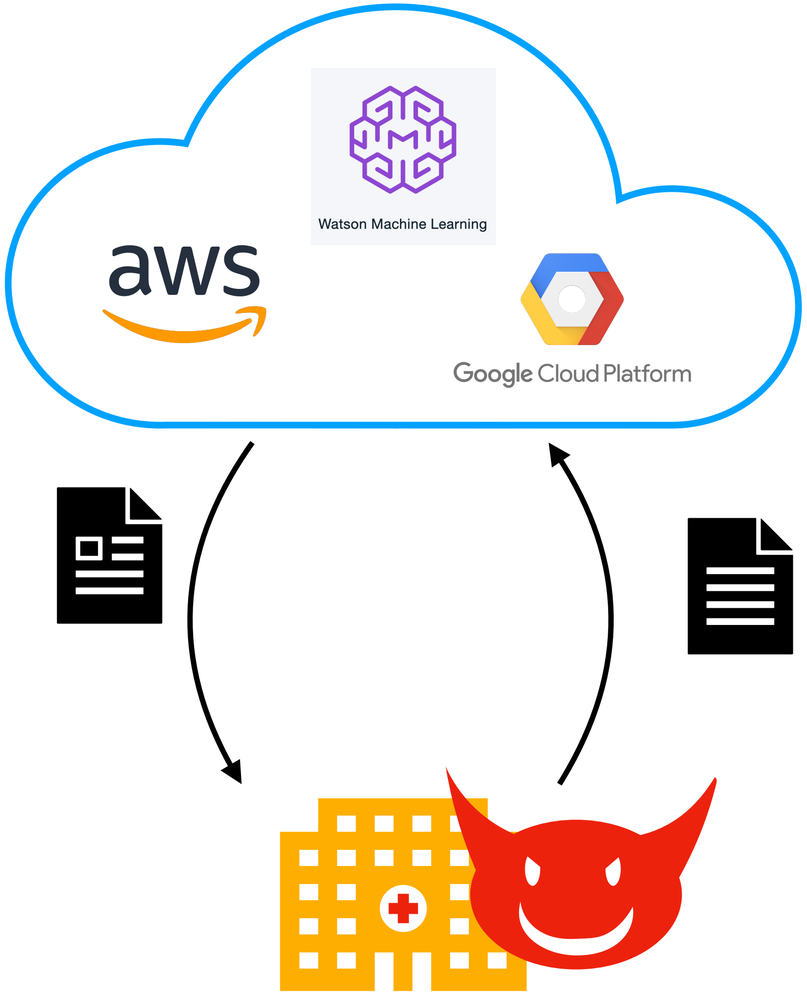


Introduction

Motivation

If the cloud is compromised, some private information of the client will leak

If the client is malicious, he might recover information about the model

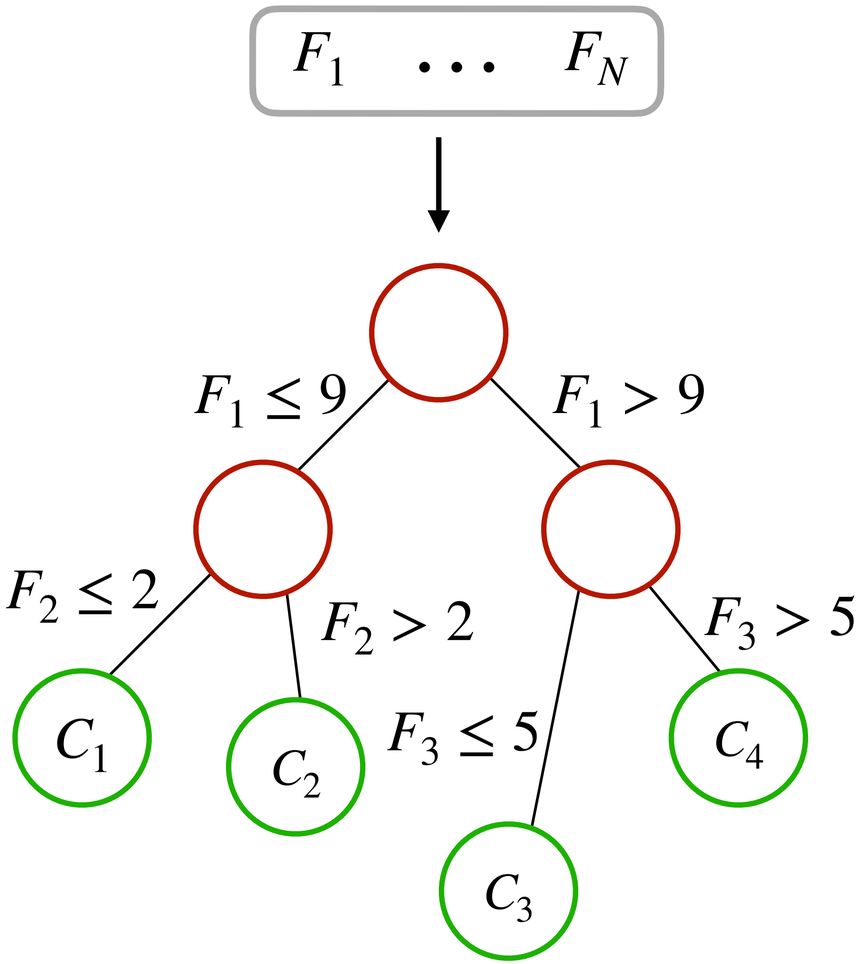


Introduction

Motivation

Decision trees are :

- Simpler to train
- Interpretable
- Used in post hoc explanation
- Building block for random forest

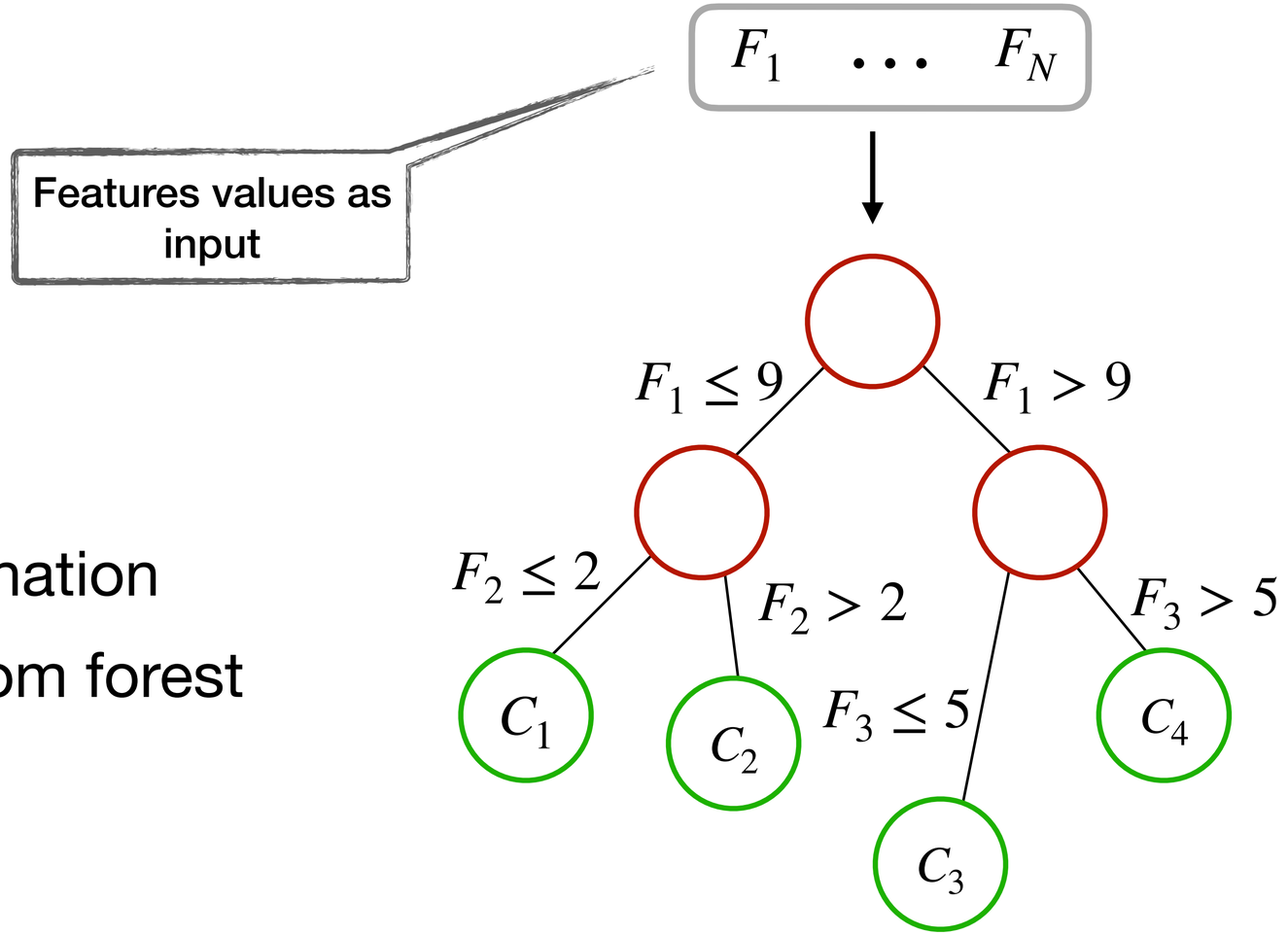


Introduction

Motivation

Decision trees are :

- Simpler to train
- Interpretable
- Used in post hoc explanation
- Building block for random forest

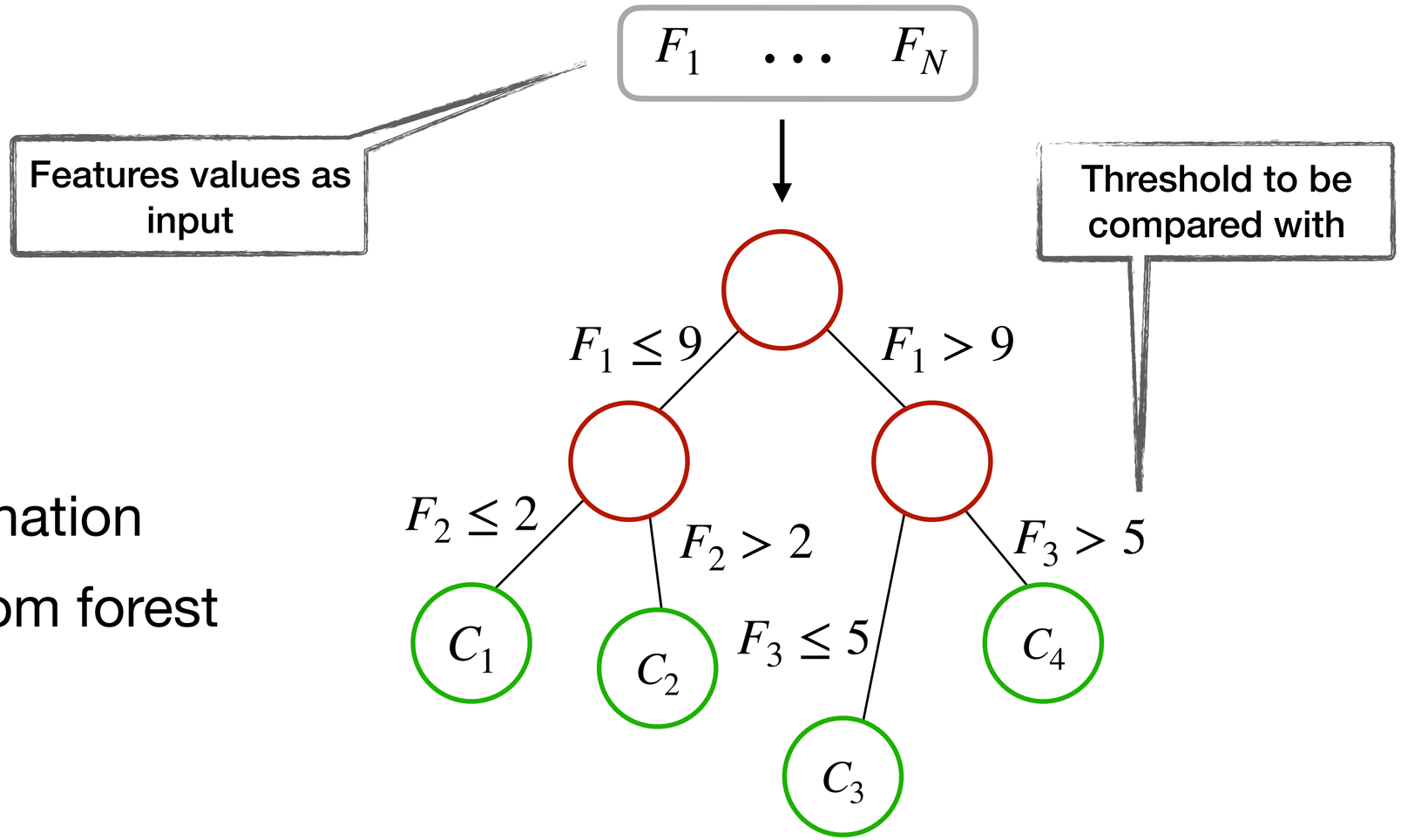


Introduction

Motivation

Decision trees are :

- Simpler to train
- Interpretable
- Used in post hoc explanation
- Building block for random forest

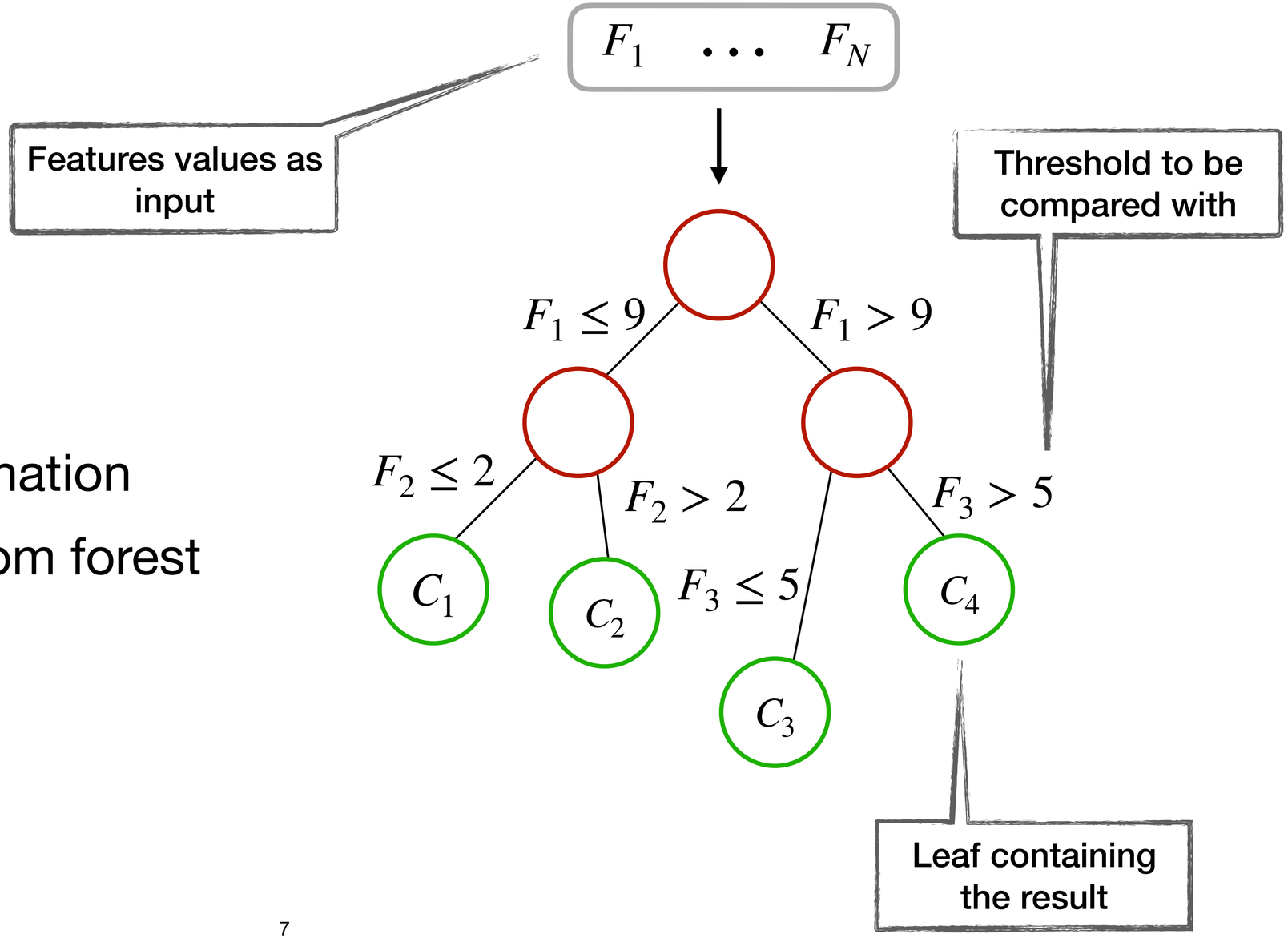


Introduction

Motivation

Decision trees are :

- Simpler to train
- Interpretable
- Used in post hoc explanation
- Building block for random forest



Introduction

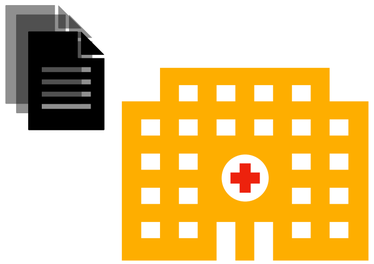
How to evaluate a decision tree on private data ?

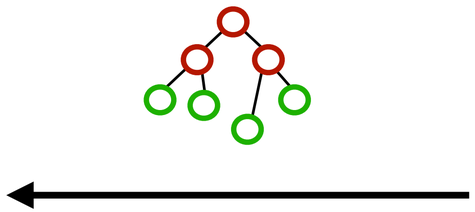
Introduction

How to evaluate a decision tree on private data ?

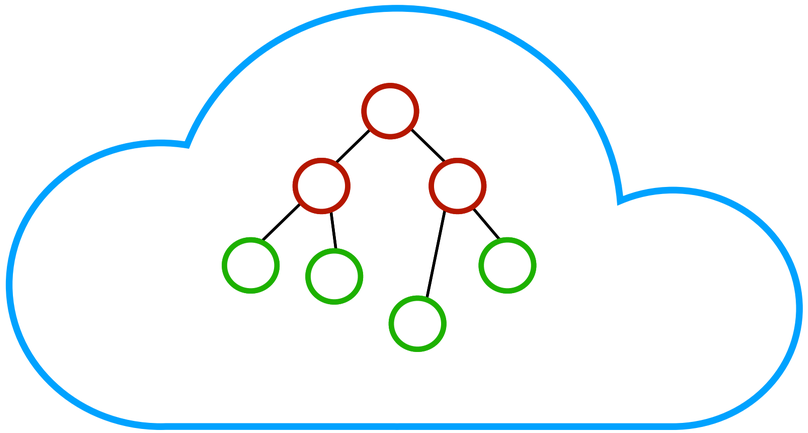
The naïve way

Client





Server



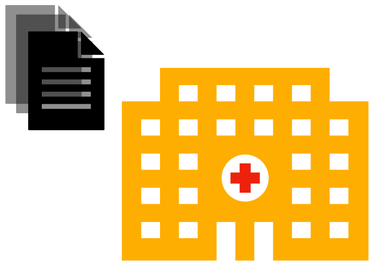
Preserves privacy of the client's data ✅
Preserves privacy of the server's model ❌
Needs one round of communication ✅

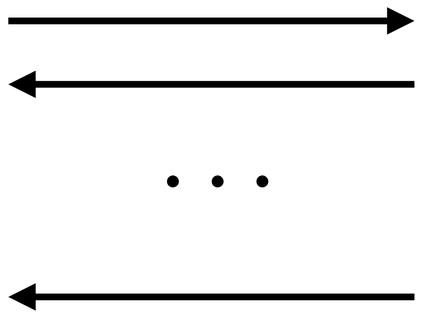
Introduction

How to evaluate a decision tree on private data ?

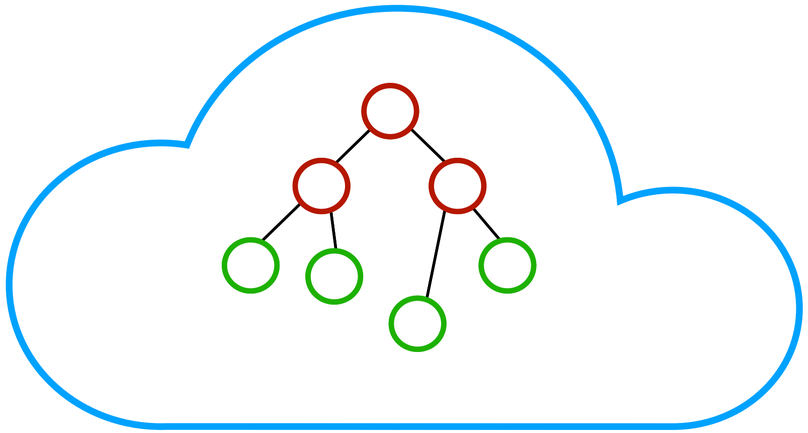
The SMPC way

Client





Server



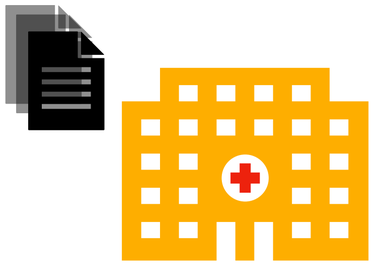
Preserves privacy of the client's data ✅
Preserves privacy of the server's model ✅
Needs one round of communication ❌

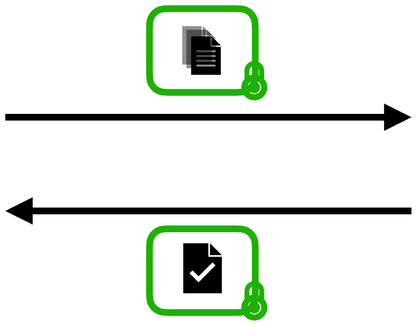
Introduction

How to evaluate a decision tree on private data ?

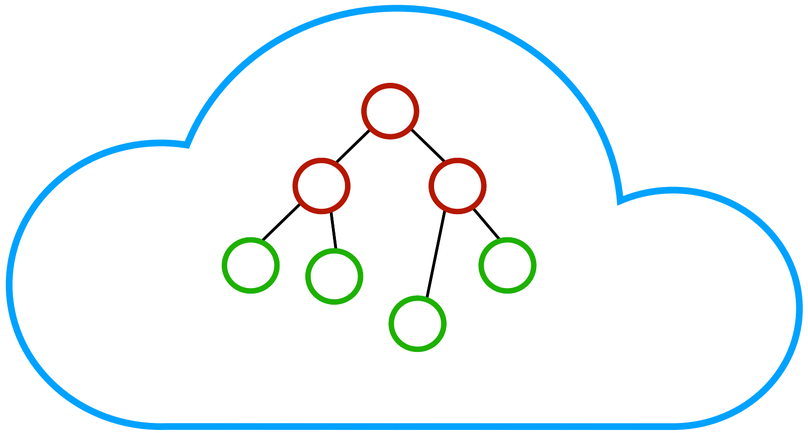
The HE way

Client





Server



Preserves privacy of the client's data ✅
Preserves privacy of the server's model ✅
Needs one round of communication ✅

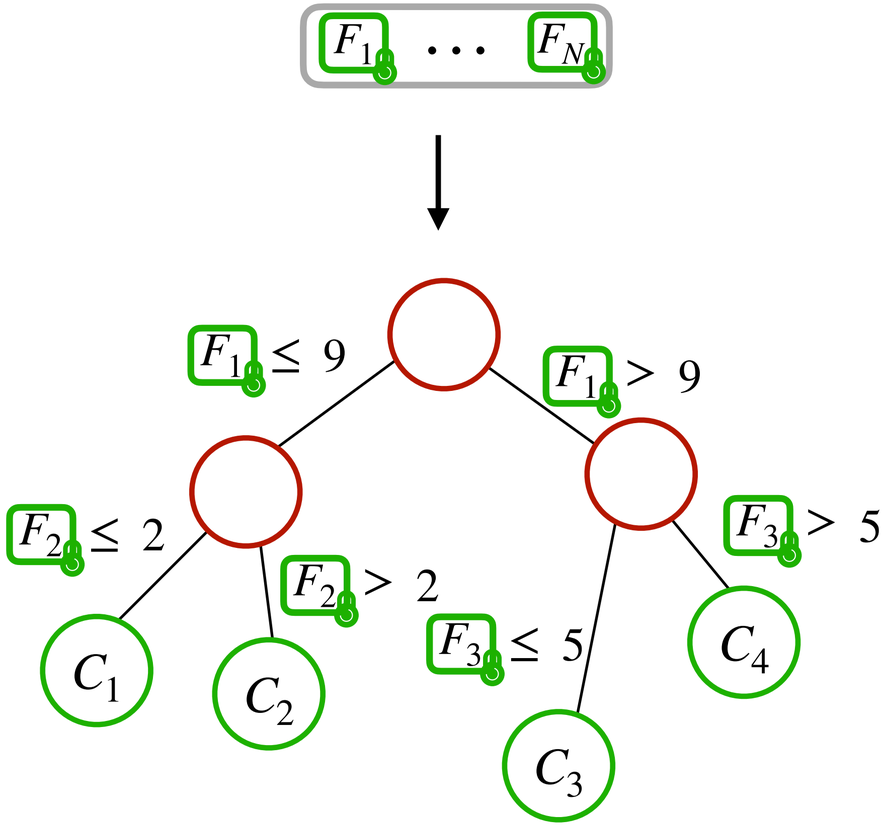
Introduction

How to evaluate a decision tree on private data ?

The client's features are encrypted

Server has to consider all the nodes ($\mathcal{O}(2^d)$)

Private comparison is the most expensive operation and there is one per node

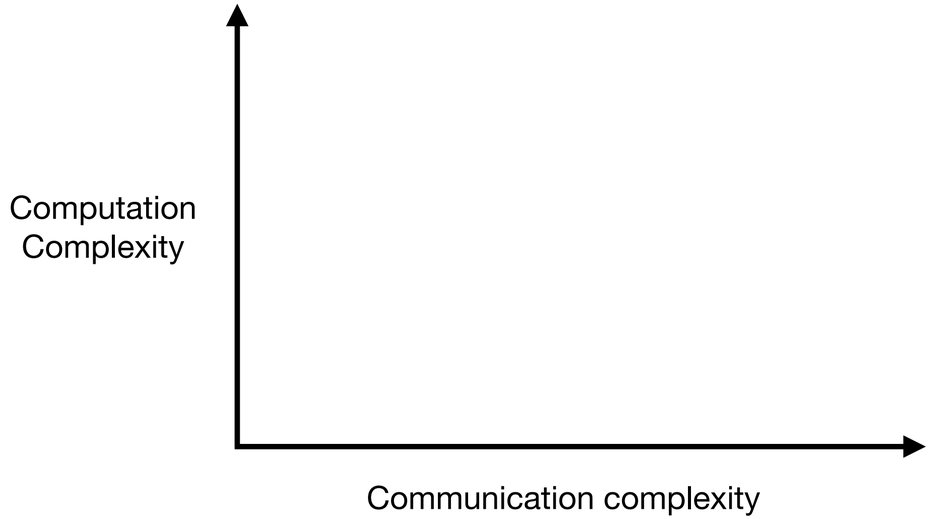


State-of-the-art

State-of-the-art

Comparisons of approaches

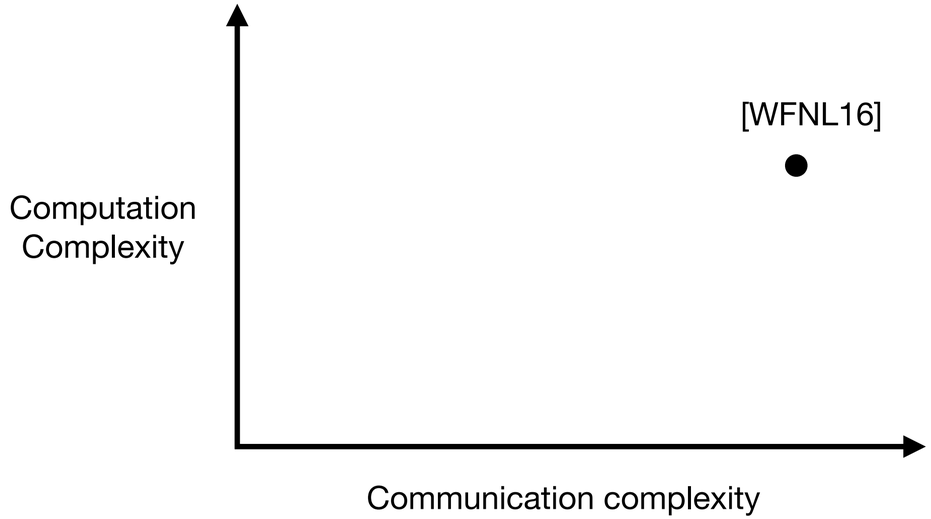
Non-interactive				
One branch				
Techniques used				



State-of-the-art

Comparisons of approaches

	[WFNL16]			
Non-interactive	❌			
One branch	❌			
Techniques used	OT + Add. HE			

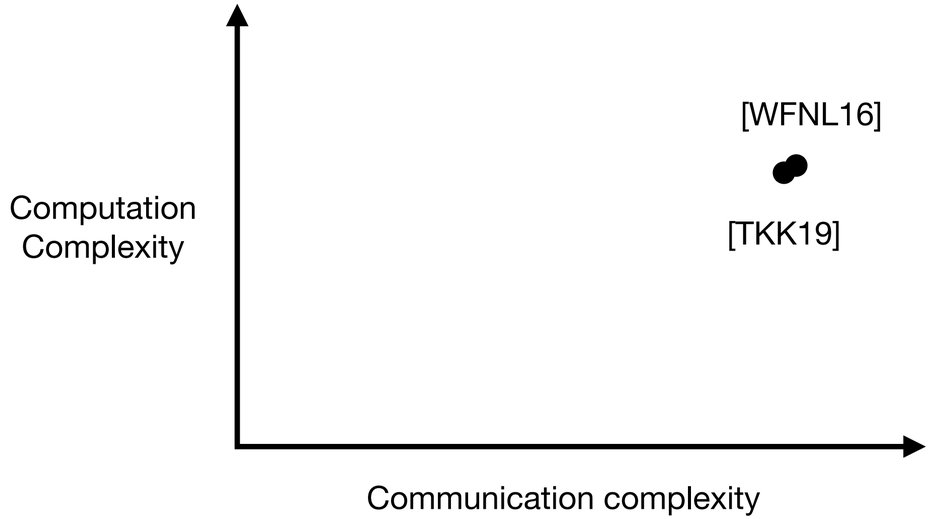


[WFNL16] : David J. Wu et al. Privately Evaluating Decision Trees and Random Forests. Proc. Priv. Enhancing Technol. 2016

State-of-the-art

Comparisons of approaches

	[WFNL16]	[TKK19]		
Non-interactive	❌	❌		
One branch	❌	✅		
Techniques used	OT + Add. HE	OT + ORAM		

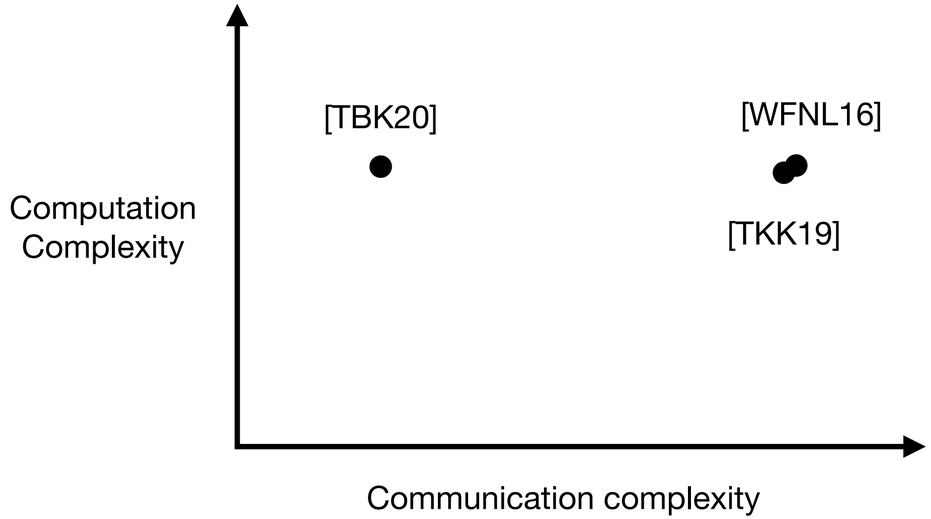


[WFNL16] : David J. Wu et al. Privately Evaluating Decision Trees and Random Forests. Proc. Priv. Enhancing Technol. 2016
[TKK19] : Anselme Tueno et al. Private Evaluation of Decision Trees using Sublinear Cost. Proc. Priv. Enhancing Technol. 2019

State-of-the-art

Comparisons of approaches

	[WFNL16]	[TKK19]	[TBK20]	
Non-interactive	❌	❌	✅	
One branch	❌	✅	❌	
Techniques used	OT + Add. HE	OT + ORAM	FHE	

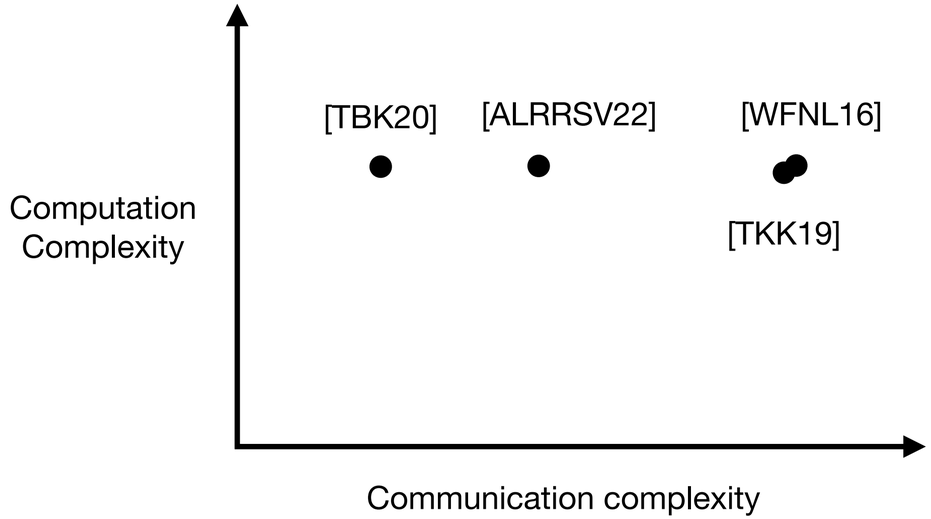


[WFNL16] : David J. Wu et al. Privately Evaluating Decision Trees and Random Forests. Proc. Priv. Enhancing Technol. 2016
[TKK19] : Anselme Tueno et al. Private Evaluation of Decision Trees using Sublinear Cost. Proc. Priv. Enhancing Technol. 2019
[TBK20] : Anselme Tueno et al. Non-interactive Private Decision Tree Evaluation. IFIP 2020

State-of-the-art

Comparisons of approaches

	[WFNL16]	[TKK19]	[TBK20]	[ALRRSV22]
Non-interactive	❌	❌	✅	✅
One branch	❌	✅	❌	❌
Techniques used	OT + Add. HE	OT + ORAM	FHE	FHE

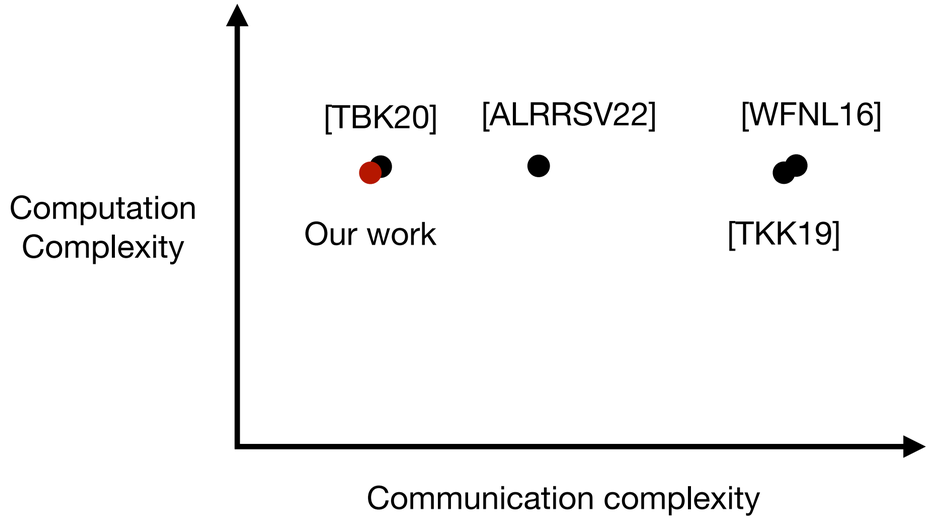


[WFNL16] : David J. Wu et al. Privately Evaluating Decision Trees and Random Forests. Proc. Priv. Enhancing Technol. 2016
[TKK19] : Anselme Tueno et al. Private Evaluation of Decision Trees using Sublinear Cost. Proc. Priv. Enhancing Technol. 2019
[TBK20] : Anselme Tueno et al. Non-interactive Private Decision Tree Evaluation. IFIP 2020
[ALRRSV22] : Adi Akavia et al. Privacy-Preserving Decision Trees Training and Prediction. ACM Trans. Priv. Secur. 2022

State-of-the-art

Comparisons of approaches

	[WFNL16]	[TKK19]	[TBK20]	[ALRRSV22]
Non-interactive	❌	❌	✅	✅
One branch	❌	✅	❌	❌
Techniques used	OT + Add. HE	OT + ORAM	FHE	FHE

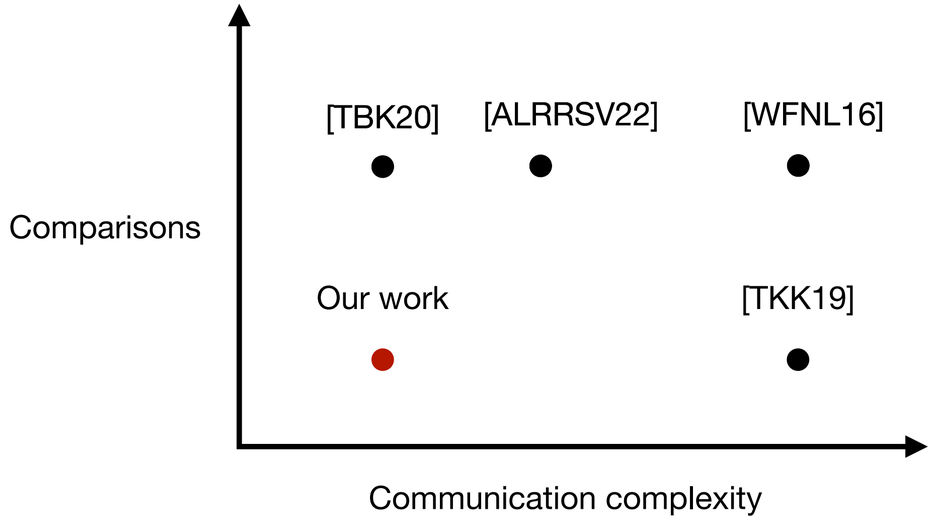


[WFNL16] : David J. Wu et al. Privately Evaluating Decision Trees and Random Forests. Proc. Priv. Enhancing Technol. 2016
[TKK19] : Anselme Tueno et al. Private Evaluation of Decision Trees using Sublinear Cost. Proc. Priv. Enhancing Technol. 2019
[TBK20] : Anselme Tueno et al. Non-interactive Private Decision Tree Evaluation. IFIP 2020
[ALRRSV22] : Adi Akavia et al. Privacy-Preserving Decision Trees Training and Prediction. ACM Trans. Priv. Secur. 2022

State-of-the-art

Comparisons of approaches

	[WFNL16]	[TKK19]	[TBK20]	[ALRRSV22]
Non-interactive	❌	❌	✅	✅
One branch	❌	✅	❌	❌
Techniques used	OT + Add. HE	OT + ORAM	FHE	FHE



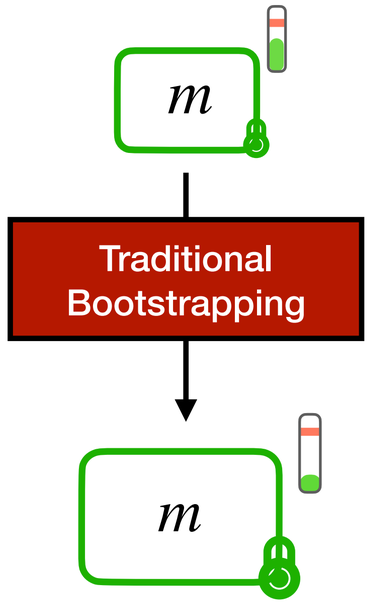
[WFNL16] : David J. Wu et al. Privately Evaluating Decision Trees and Random Forests. Proc. Priv. Enhancing Technol. 2016
[TKK19] : Anselme Tueno et al. Private Evaluation of Decision Trees using Sublinear Cost. Proc. Priv. Enhancing Technol. 2019
[TBK20] : Anselme Tueno et al. Non-interactive Private Decision Tree Evaluation. IFIP 2020
[ALRRSV22] : Adi Akavia et al. Privacy-Preserving Decision Trees Training and Prediction. ACM Trans. Priv. Secur. 2022

Preliminaries

Preliminaries

Functional Bootstrapping

Traditional bootstrapping allows to refresh the noise of a ciphertext

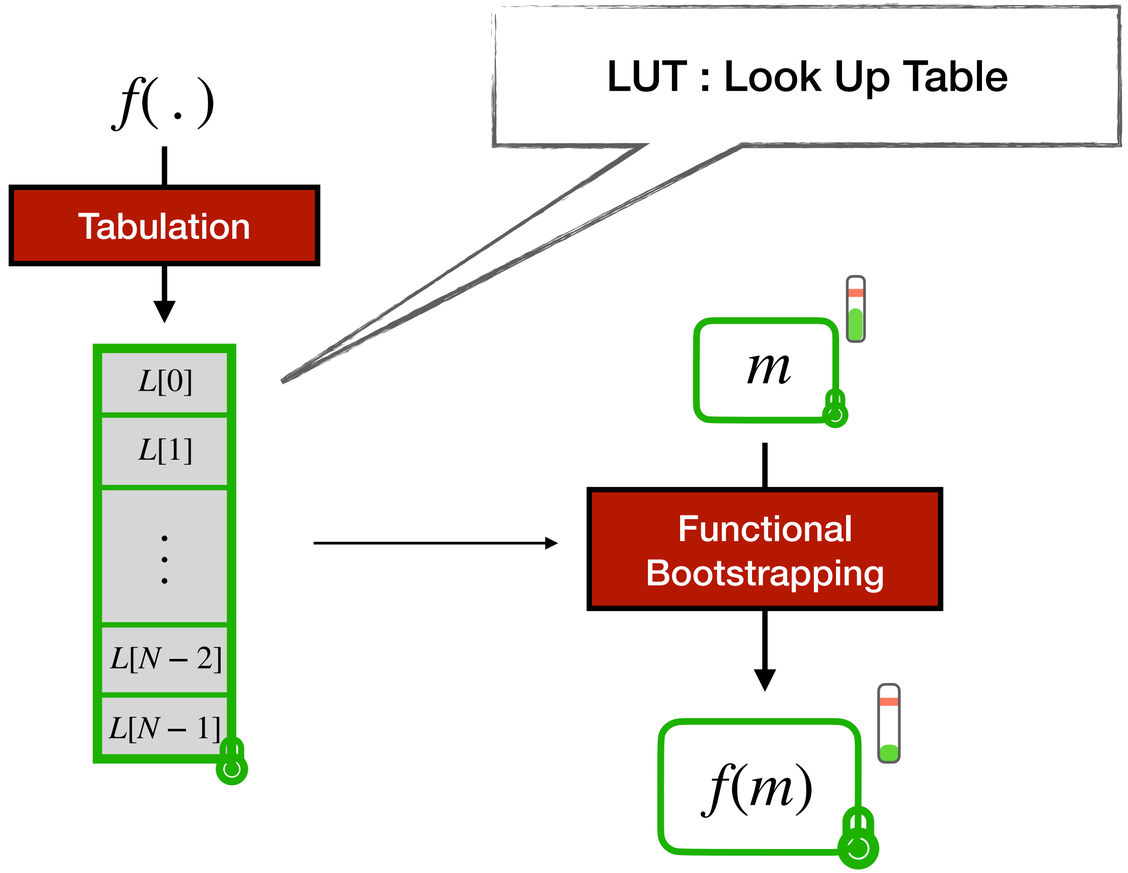


Preliminaries

Functional Bootstrapping

Traditional bootstrapping allows to refresh the noise of a ciphertext

Functional bootstrapping exploit the traditional one to compute arbitrary function



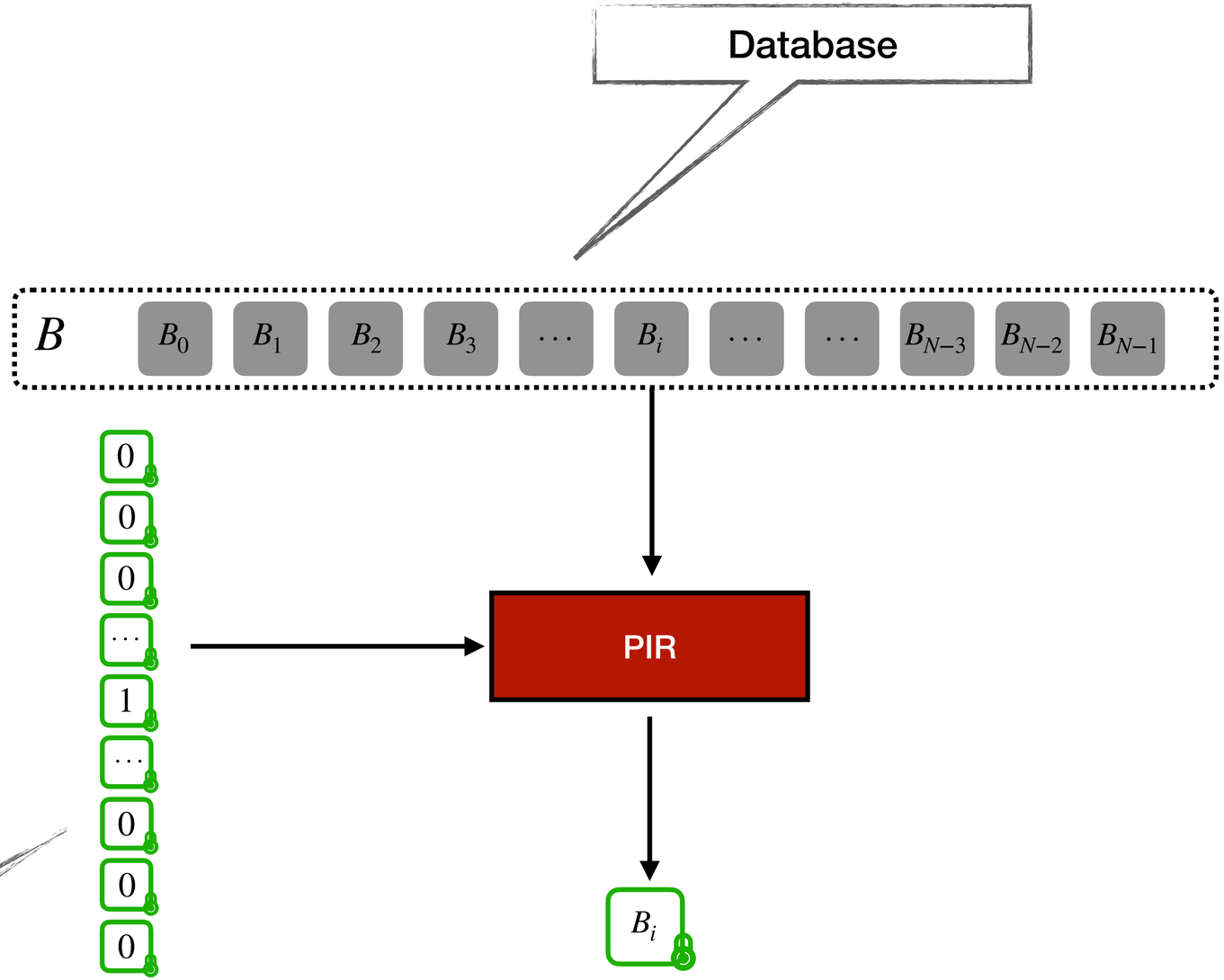
Preliminaries

Private Information Retrieval

A PIR can be done by an absorption between the encrypted request and the database

Homomorphic absorption :

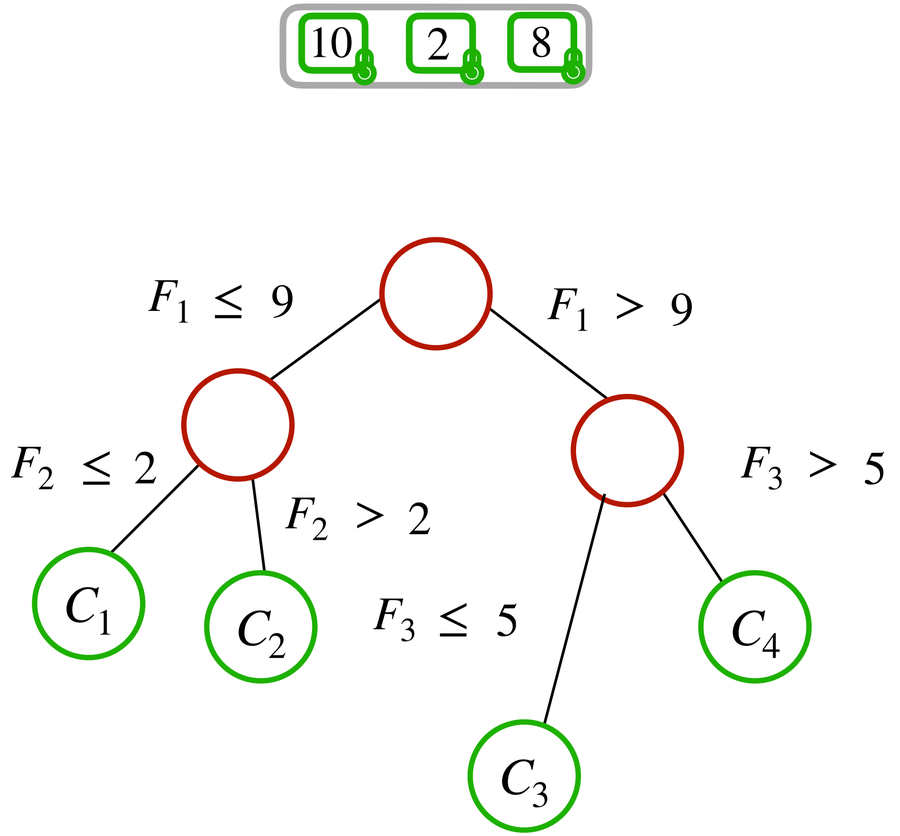
$$\boxed{1} \cdot B_i = \boxed{B_i}$$



Our proposal

Our proposal

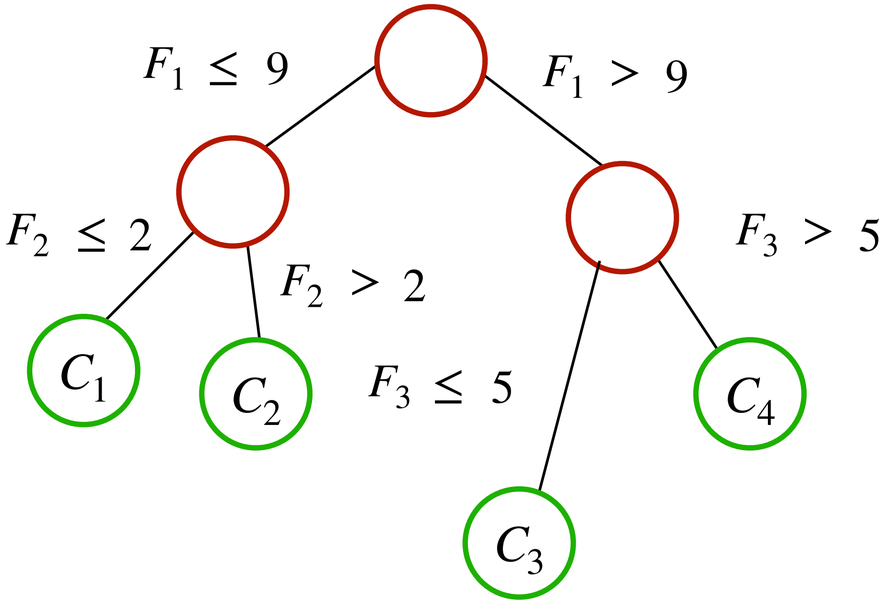
Challenge : reducing the number of comparisons



Our proposal

Challenge : reducing the number of comparisons

To address this challenge, the server has to accomplish two tasks :

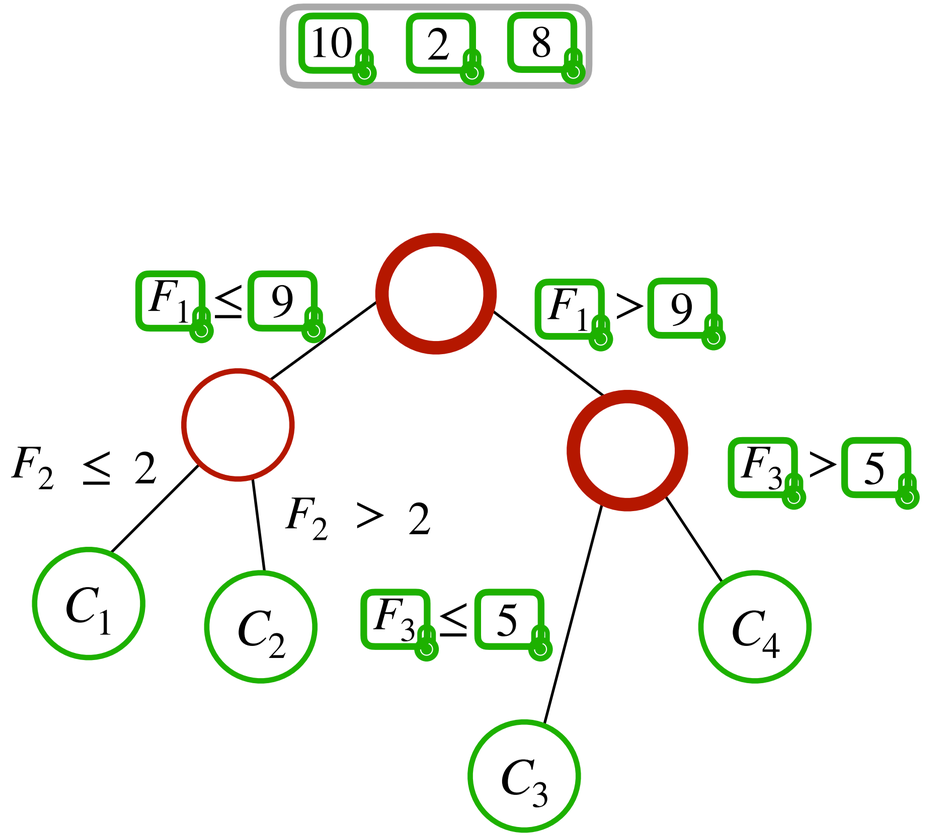


Our proposal

Challenge : reducing the number of comparisons

To address this challenge, the server has to accomplish two tasks :

1. Blindly select the node to evaluate



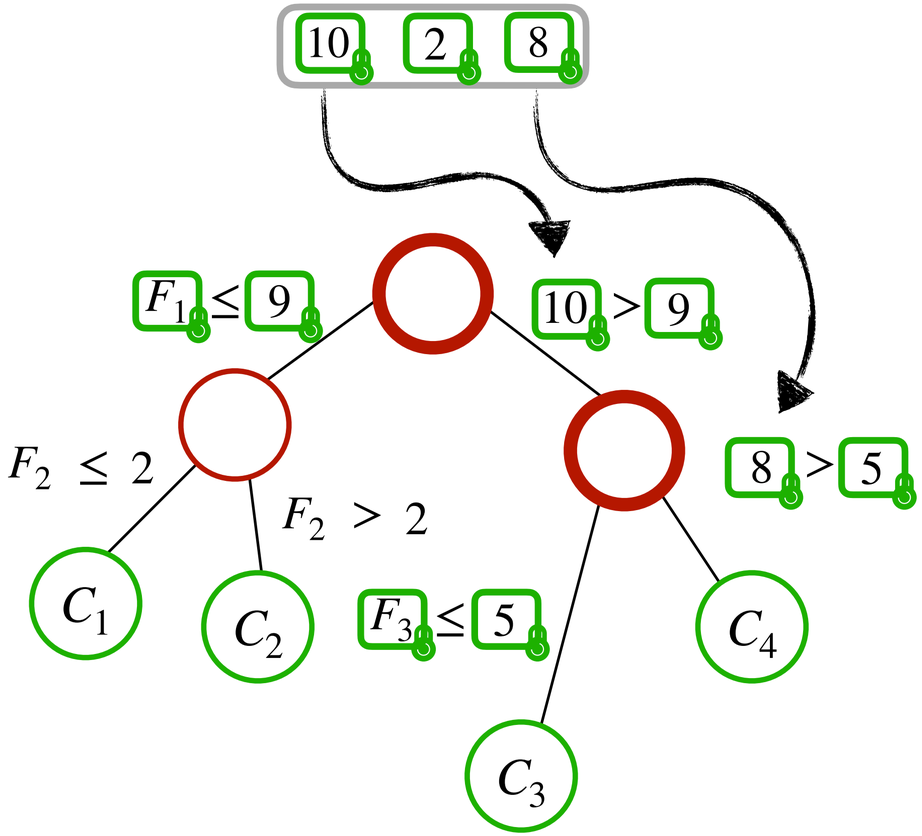
Our proposal

Challenge : reducing the number of comparisons

To address this challenge, the server has to accomplish two tasks :

1. Blindly select the node to evaluate

2. Blindly select the attribute without getting any knowledge



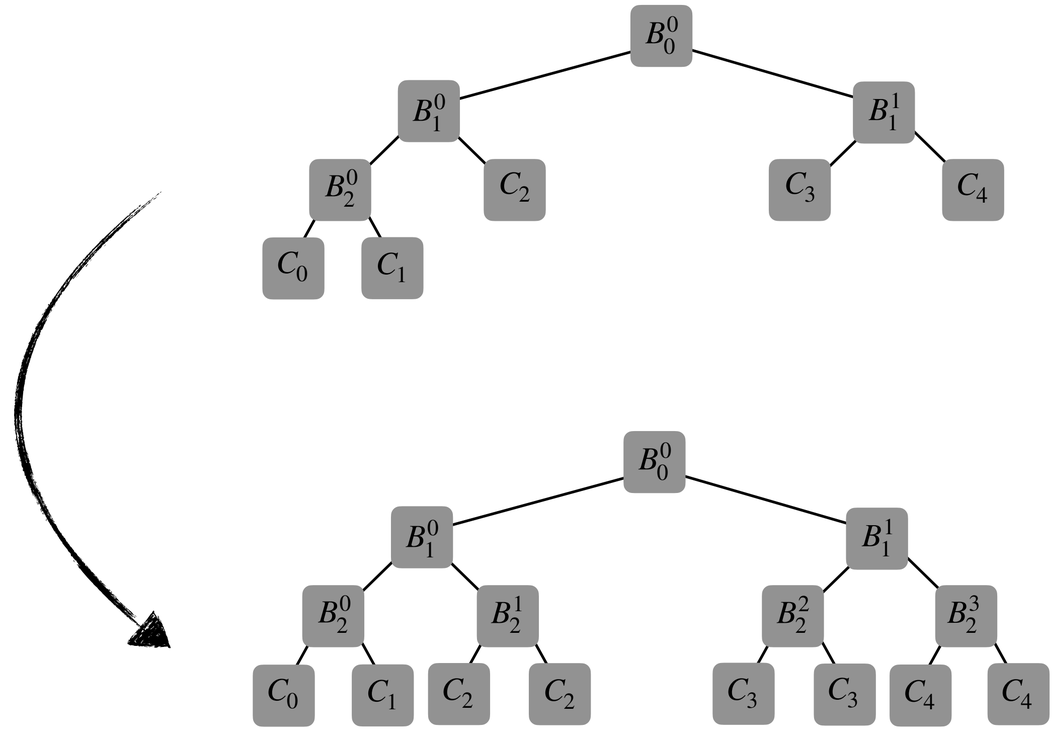
Our proposal

Impact on data structure

Our proposal

Impact on data structure

1. We complete the decision tree by adding some dummy nodes if necessary

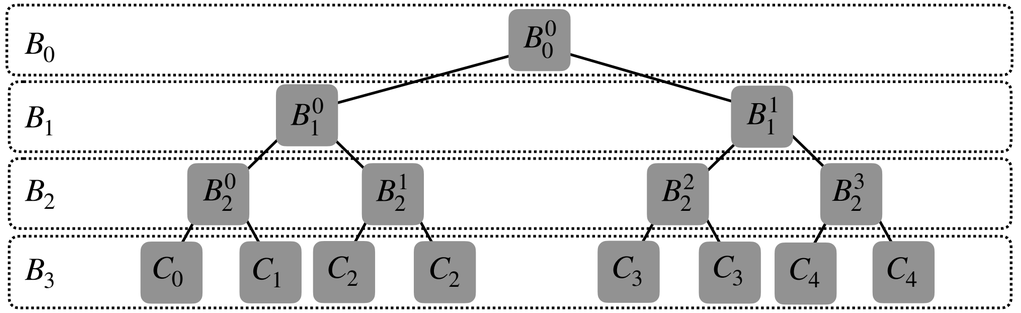


Our proposal

Impact on data structure

1. We complete the decision tree by adding some dummy nodes if necessary

2. We consider the tree as a database composed by d sub-databases called « levels »



Our proposal

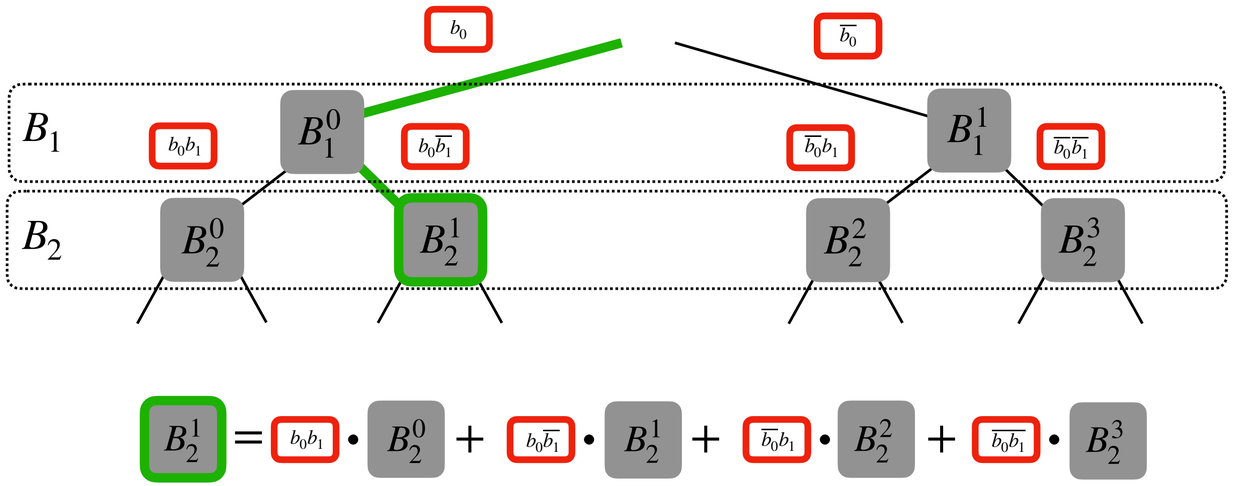
New primitive : Blind Node Selection

Each level, except the root, is associated to a new encrypted bit b_j

This bit is used to build an accumulator bit associated to each node

Only one of this accumulator bit is set

Finally, we get the correct node a-la-PIR

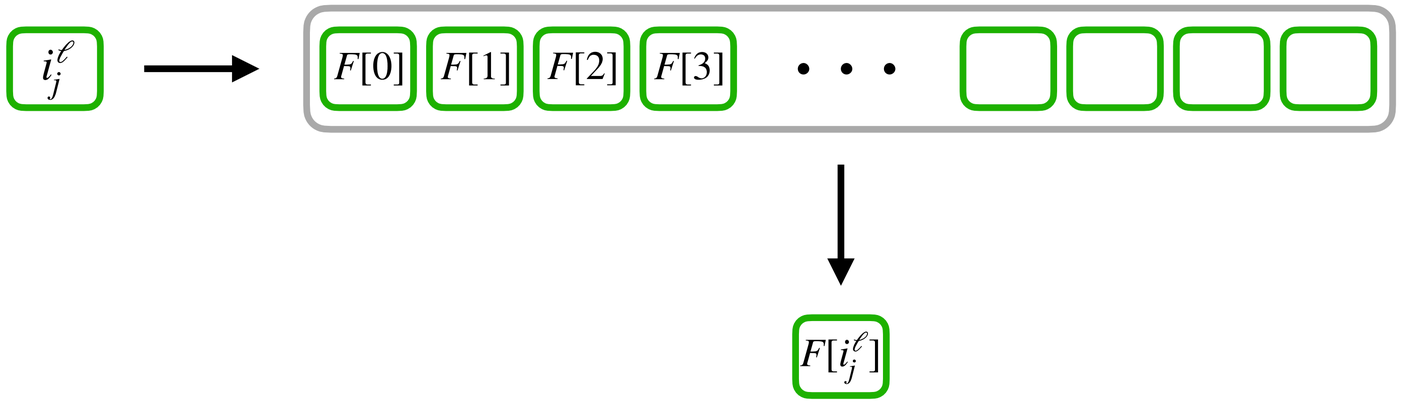


Our proposal

New primitive : Blind Array Access

The idea : use the feature vector as a LUT in the functional bootstrapping

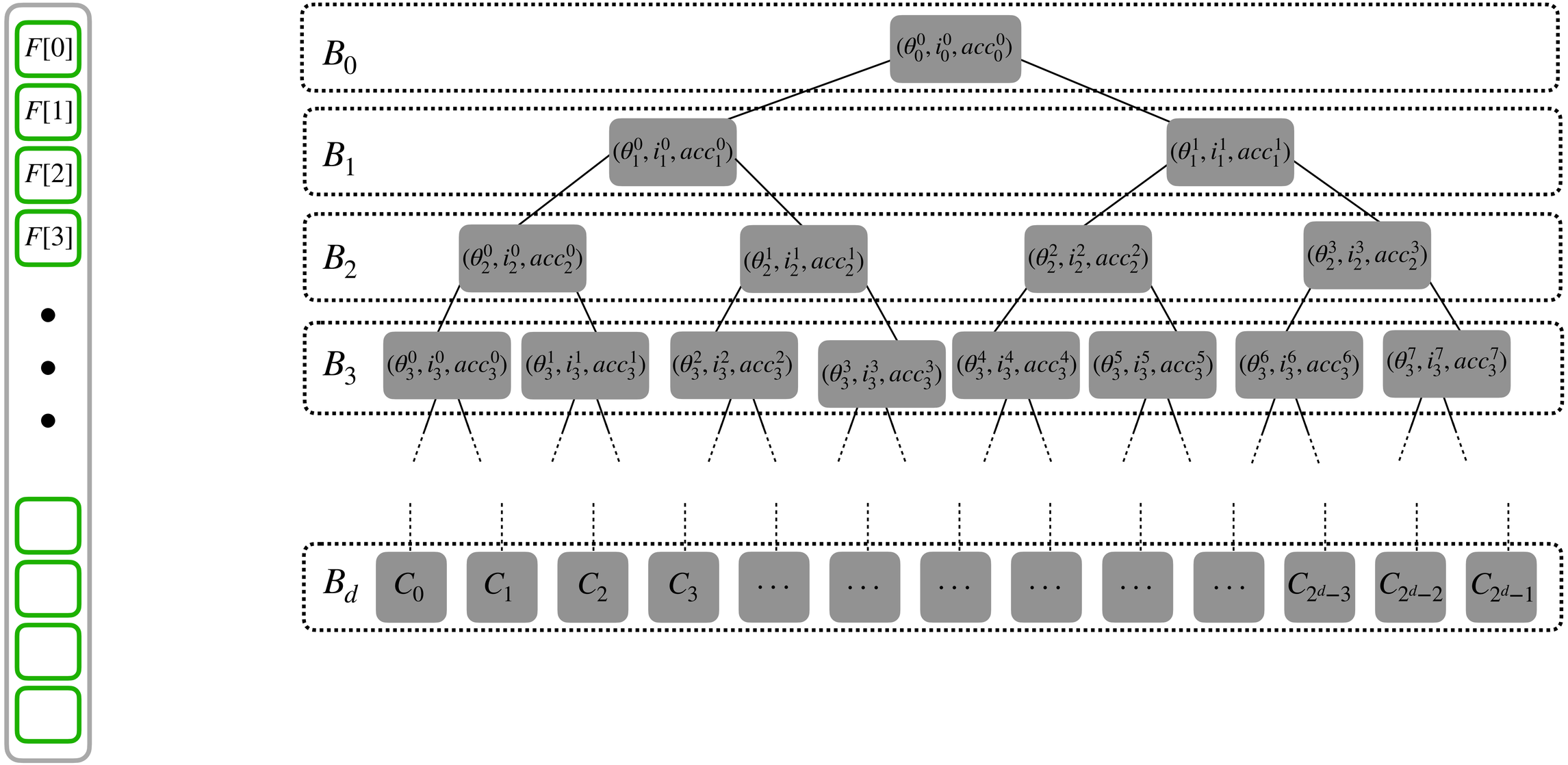
The message to be bootstrapped is the index encrypted



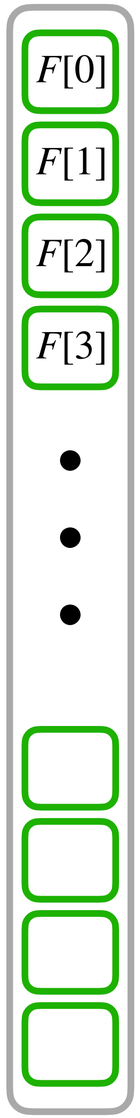
Improvement : use the binary decomposition of the index as a bootstrapping key

Client's Features

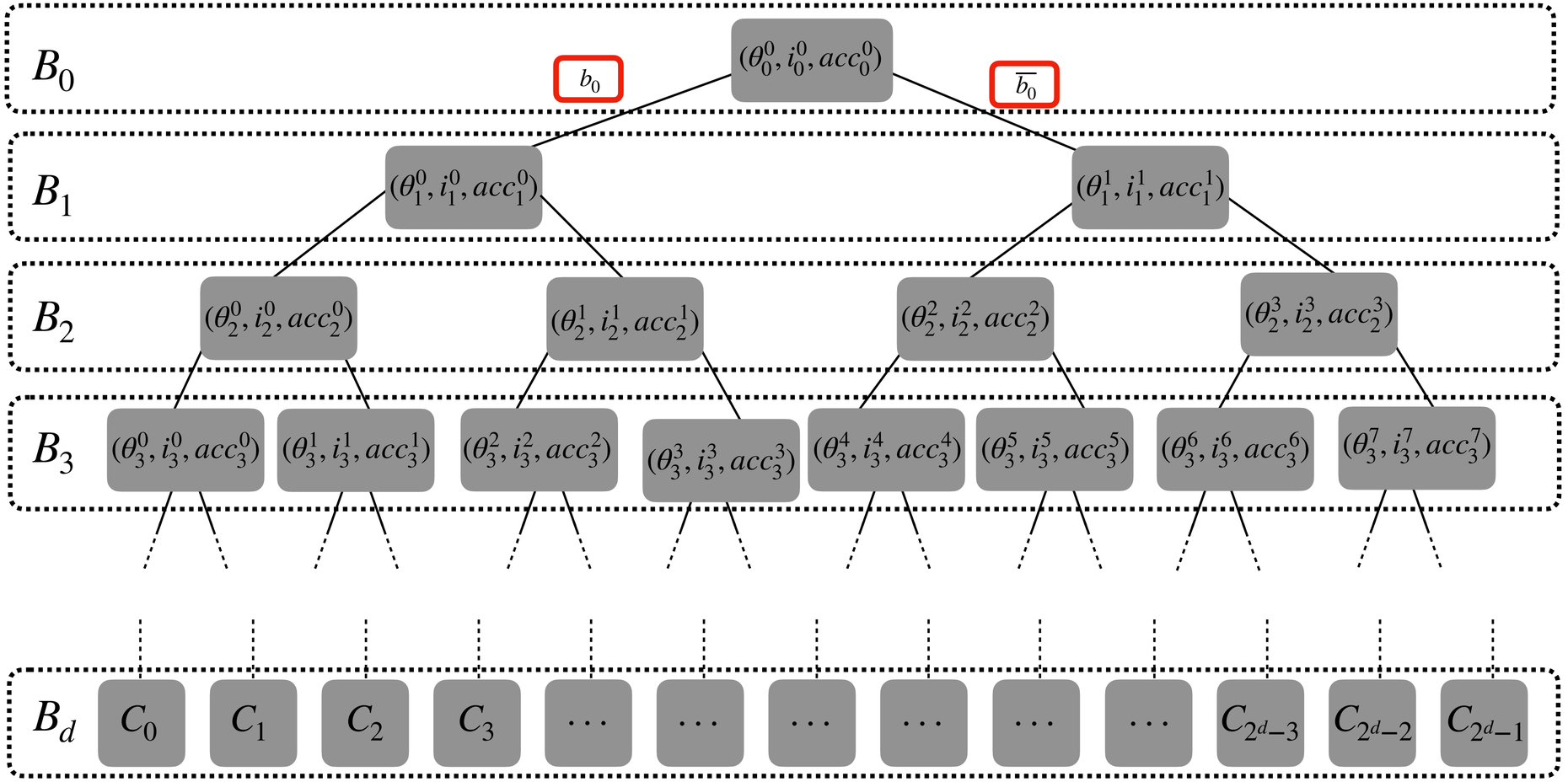
Server's Decision Tree



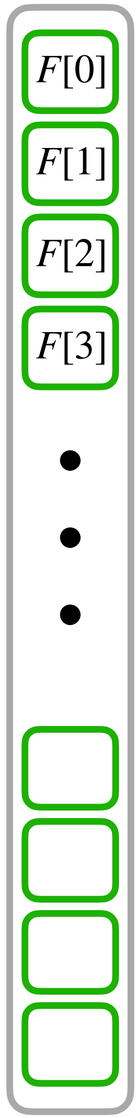
Client's Features



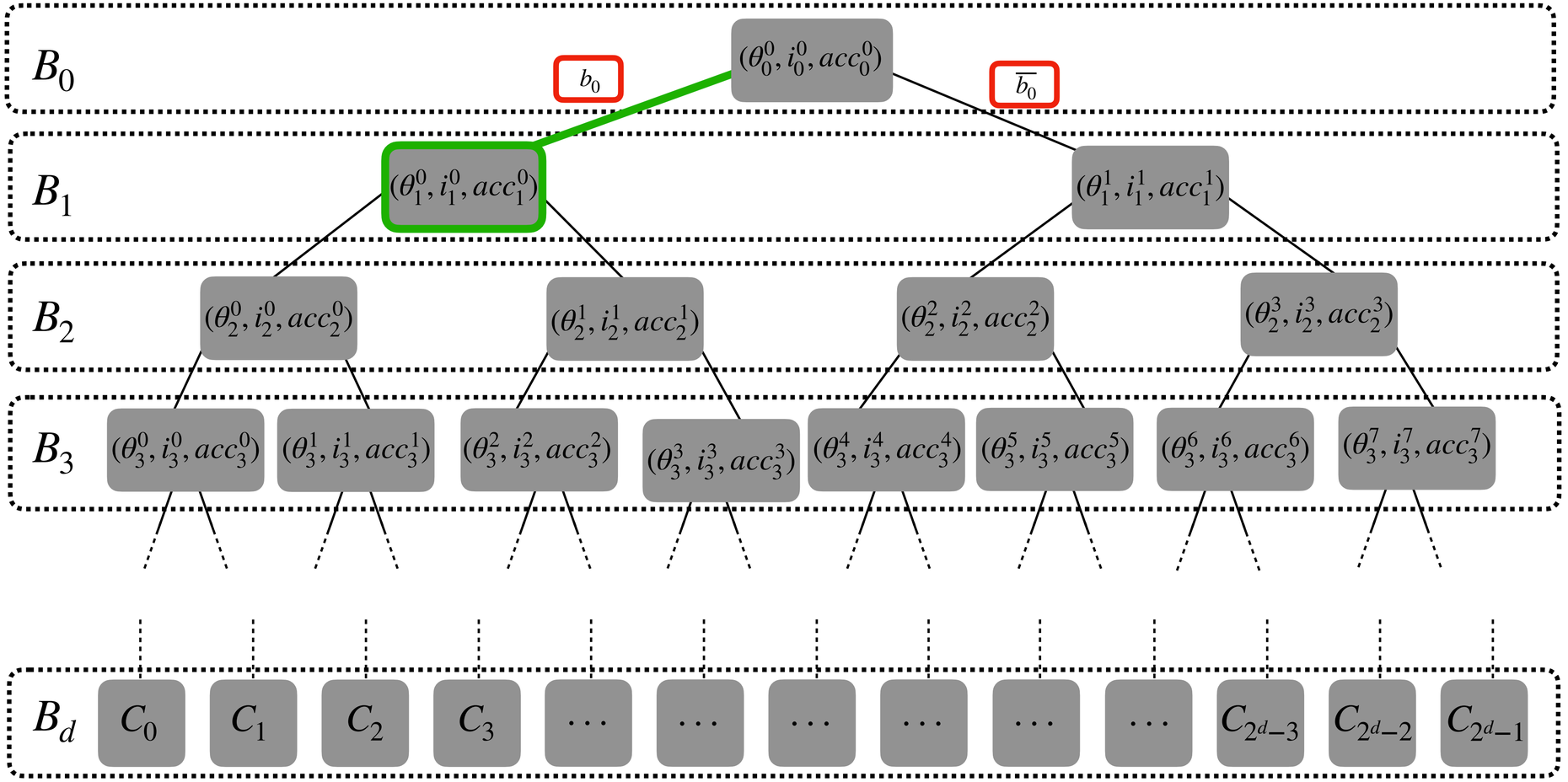
Server's Decision Tree



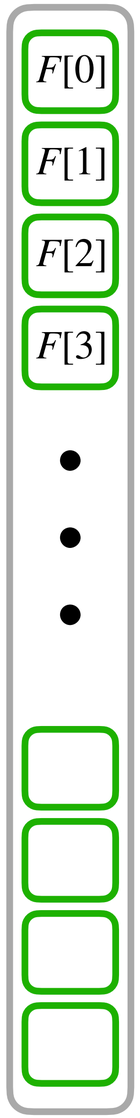
Client's Features



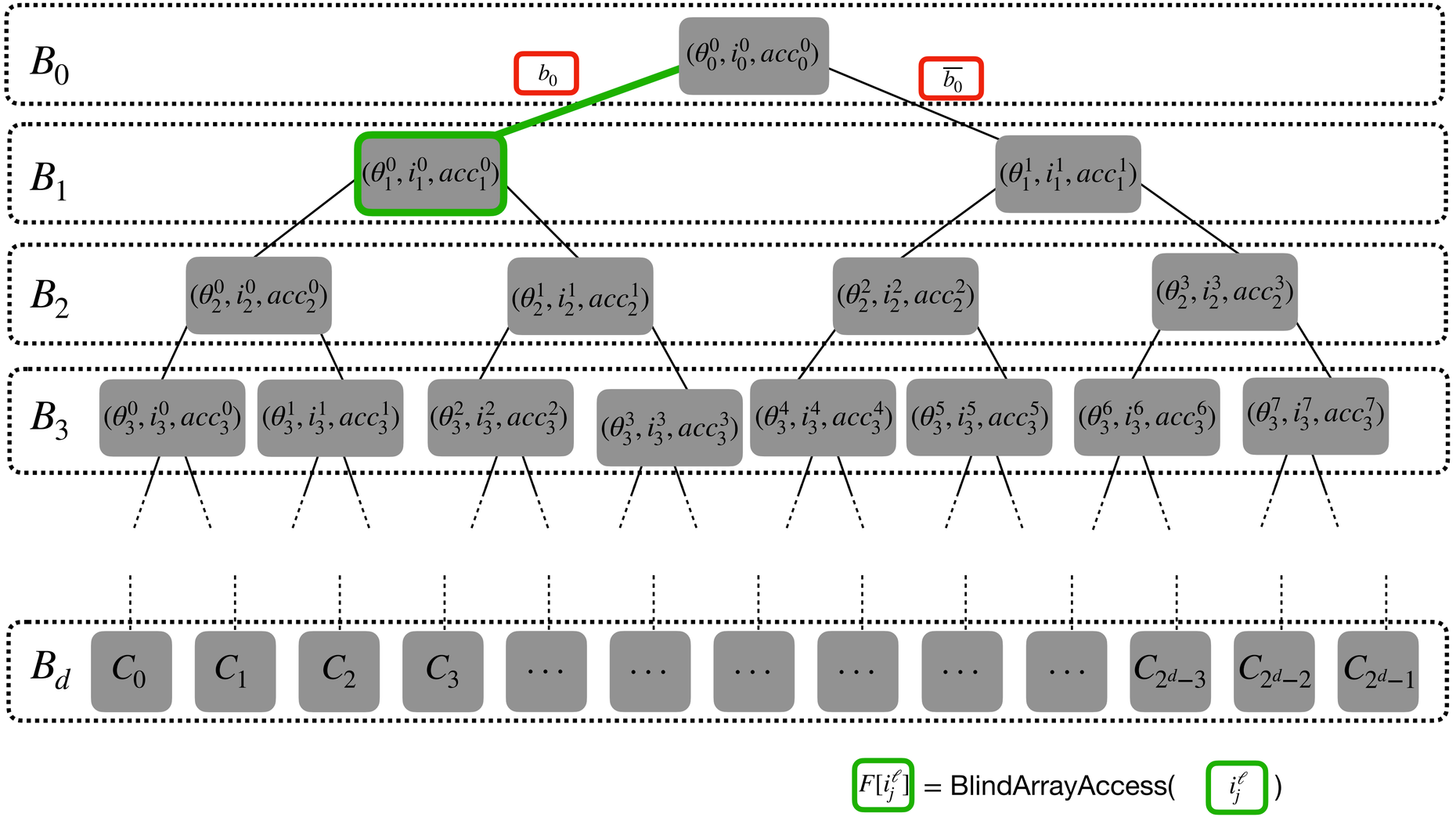
Server's Decision Tree



Client's Features

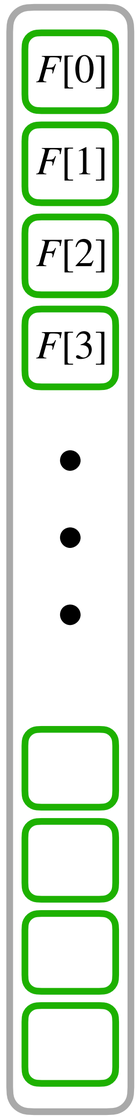


Server's Decision Tree

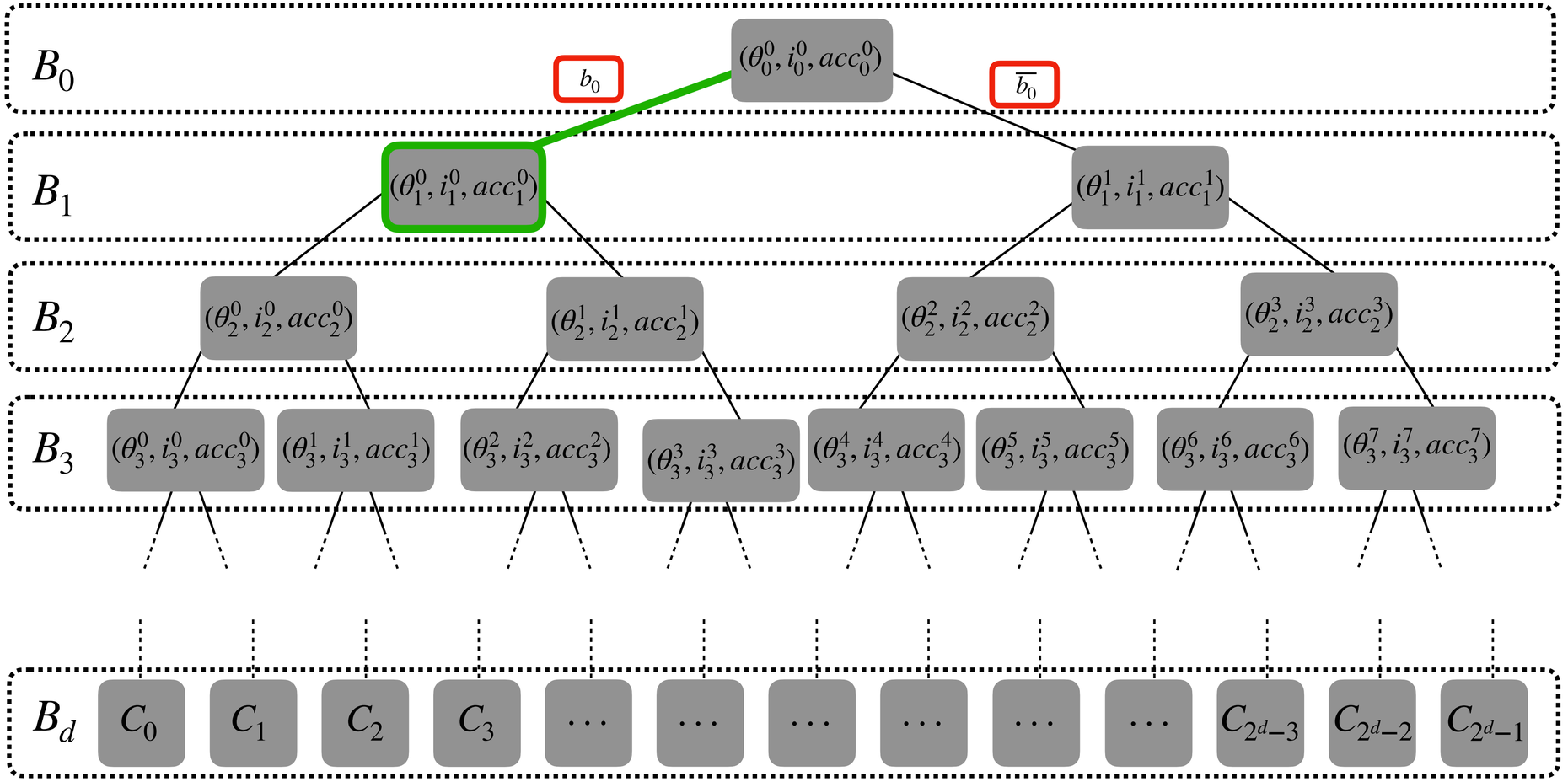


$F[i_j^\ell]$ = BlindArrayAccess(i_j^ℓ)

Client's Features



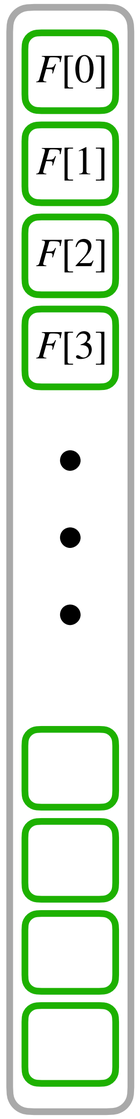
Server's Decision Tree



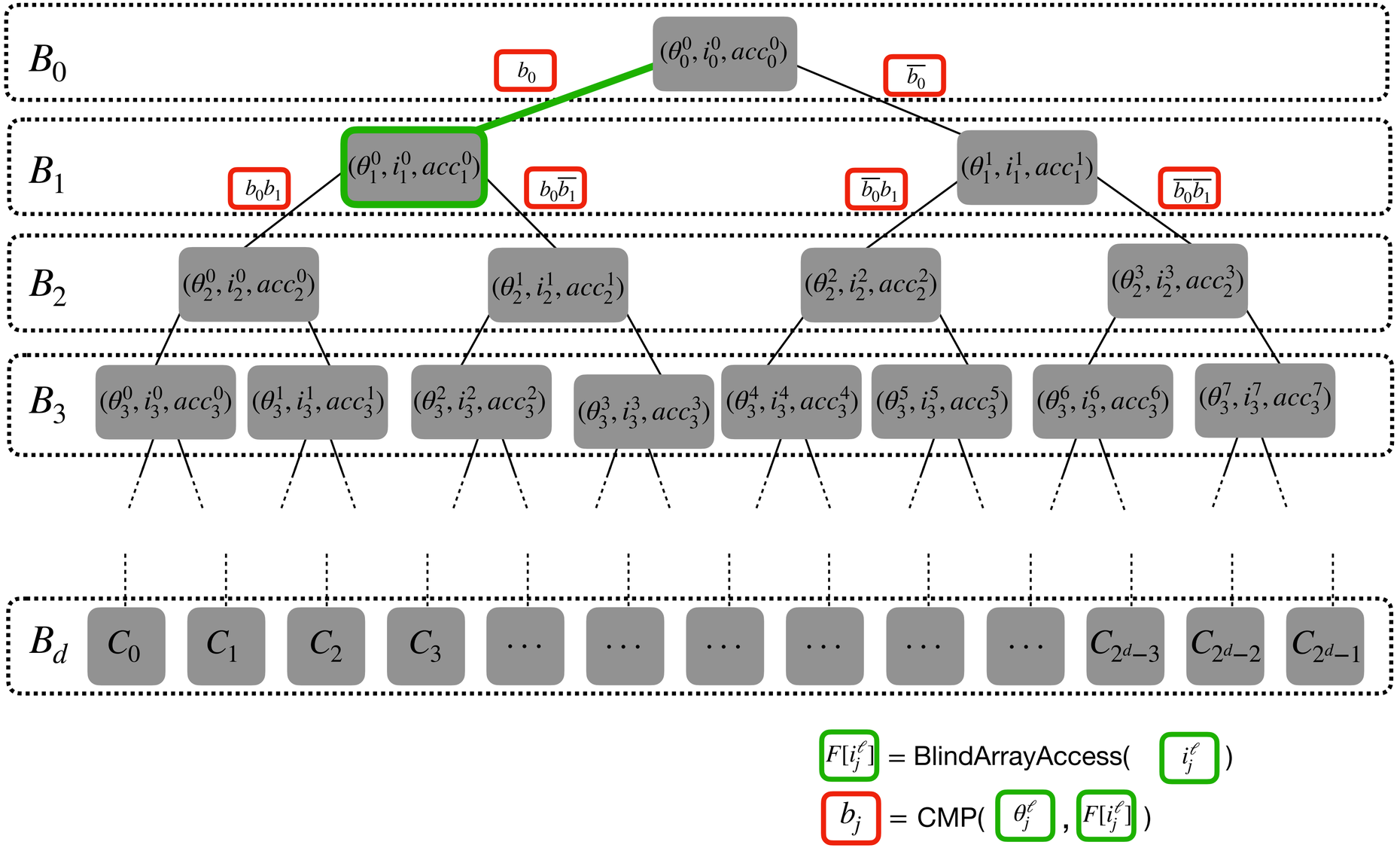
$F[i_j^\ell]$ = BlindArrayAccess(i_j^ℓ)

b_j = CMP(θ_j^ℓ , $F[i_j^\ell]$)

Client's Features



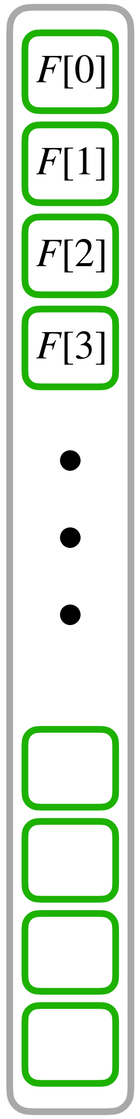
Server's Decision Tree



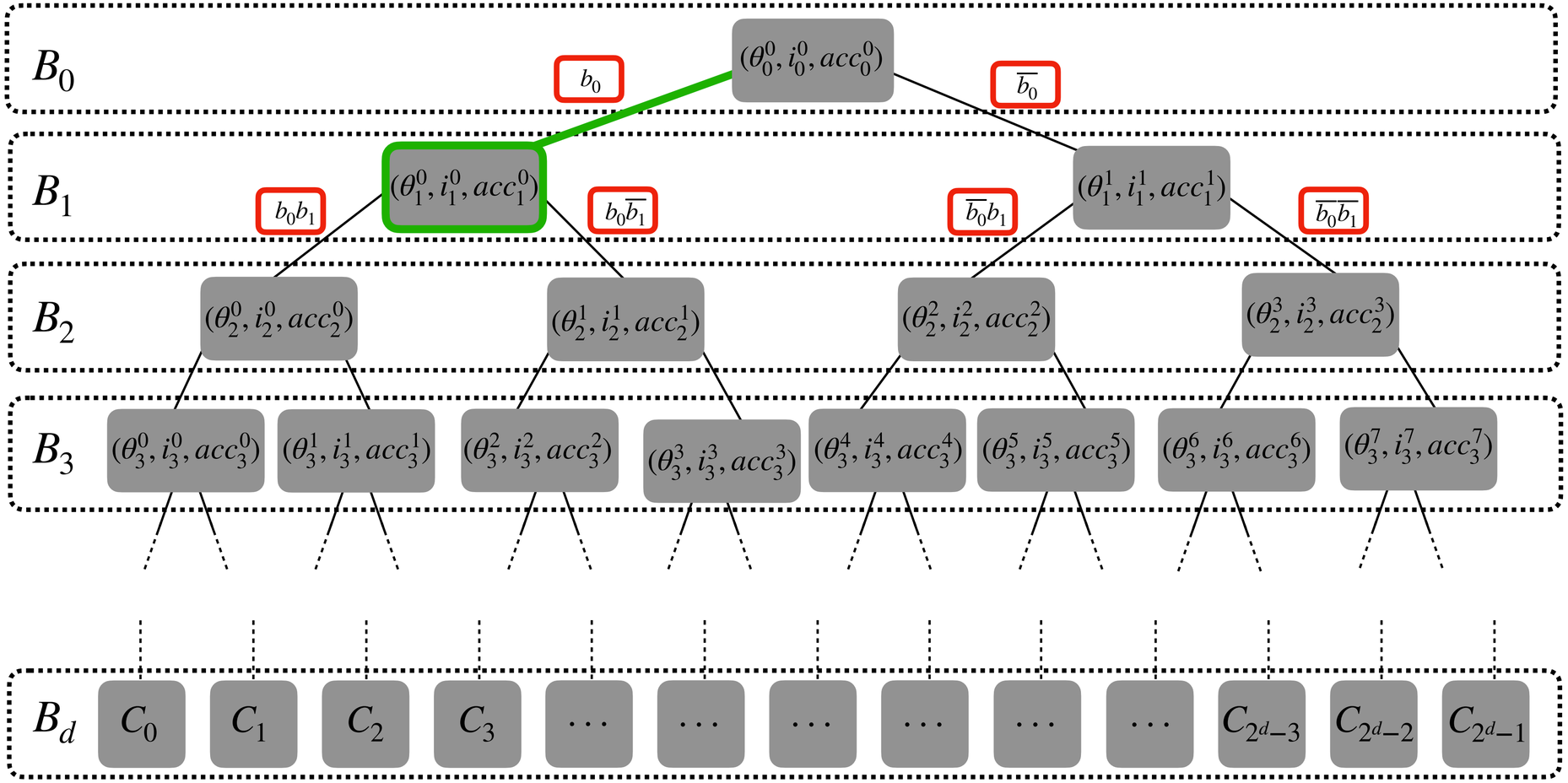
$F[i_j^\ell]$ = BlindArrayAccess(i_j^ℓ)

b_j = CMP(θ_j^ℓ , $F[i_j^\ell]$)

Client's Features



Server's Decision Tree

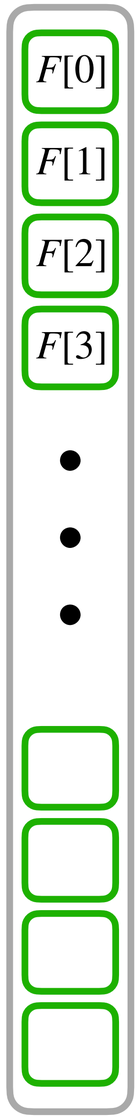


$F[i_j^\ell]$ = BlindArrayAccess(i_j^ℓ)

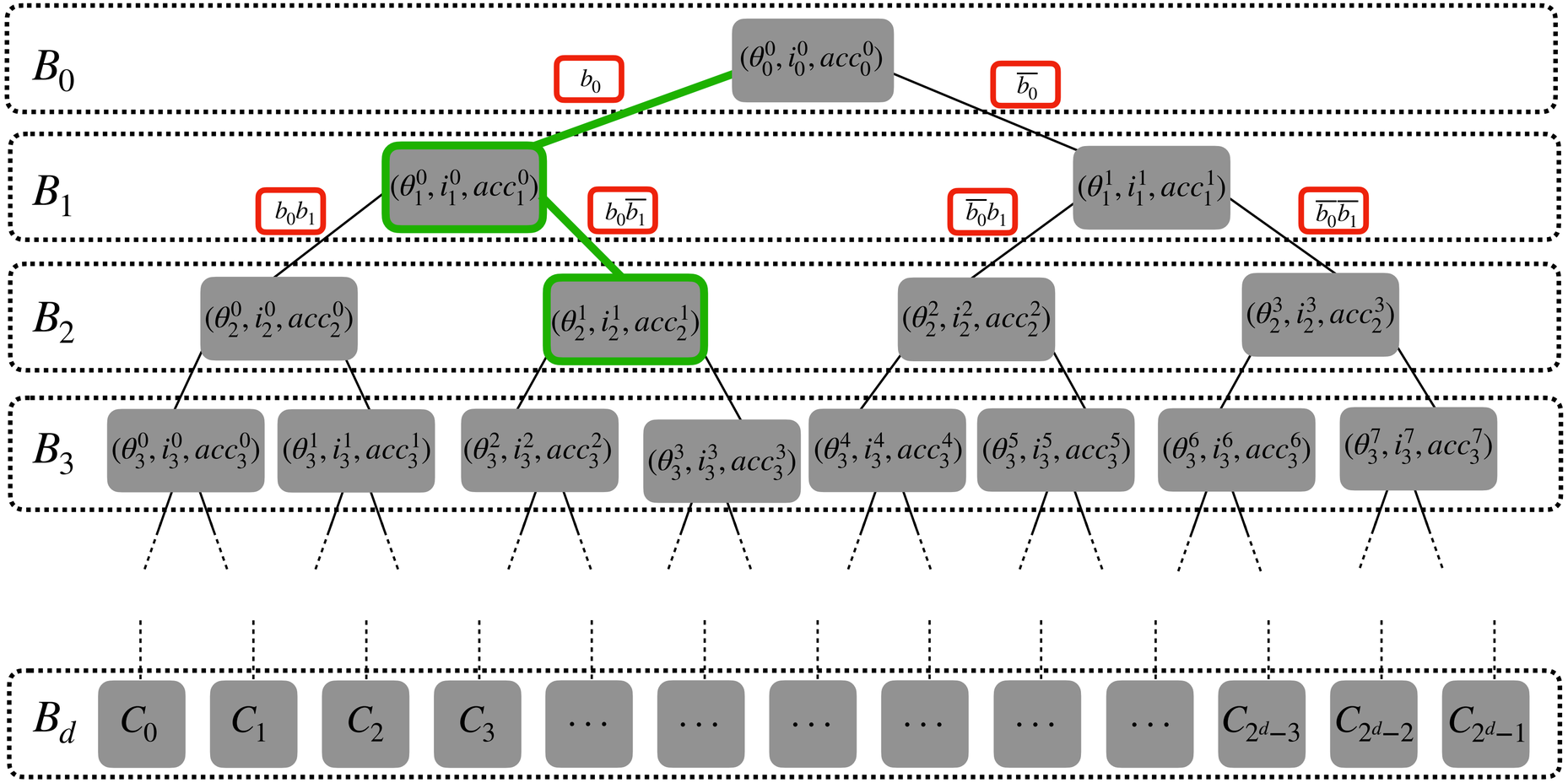
b_j = CMP(θ_j^ℓ , $F[i_j^\ell]$)

$B_{j+1}^{\ell'}$ = BlindNodeSelection(b_j , B_j , B_{j+1})

Client's Features



Server's Decision Tree

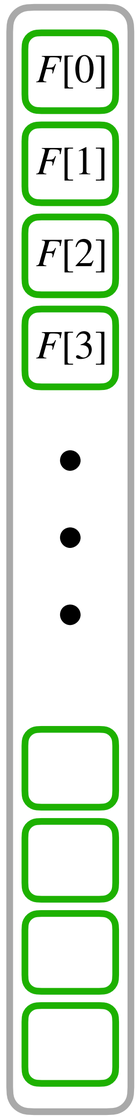


$F[i_j^\ell]$ = BlindArrayAccess(i_j^ℓ)

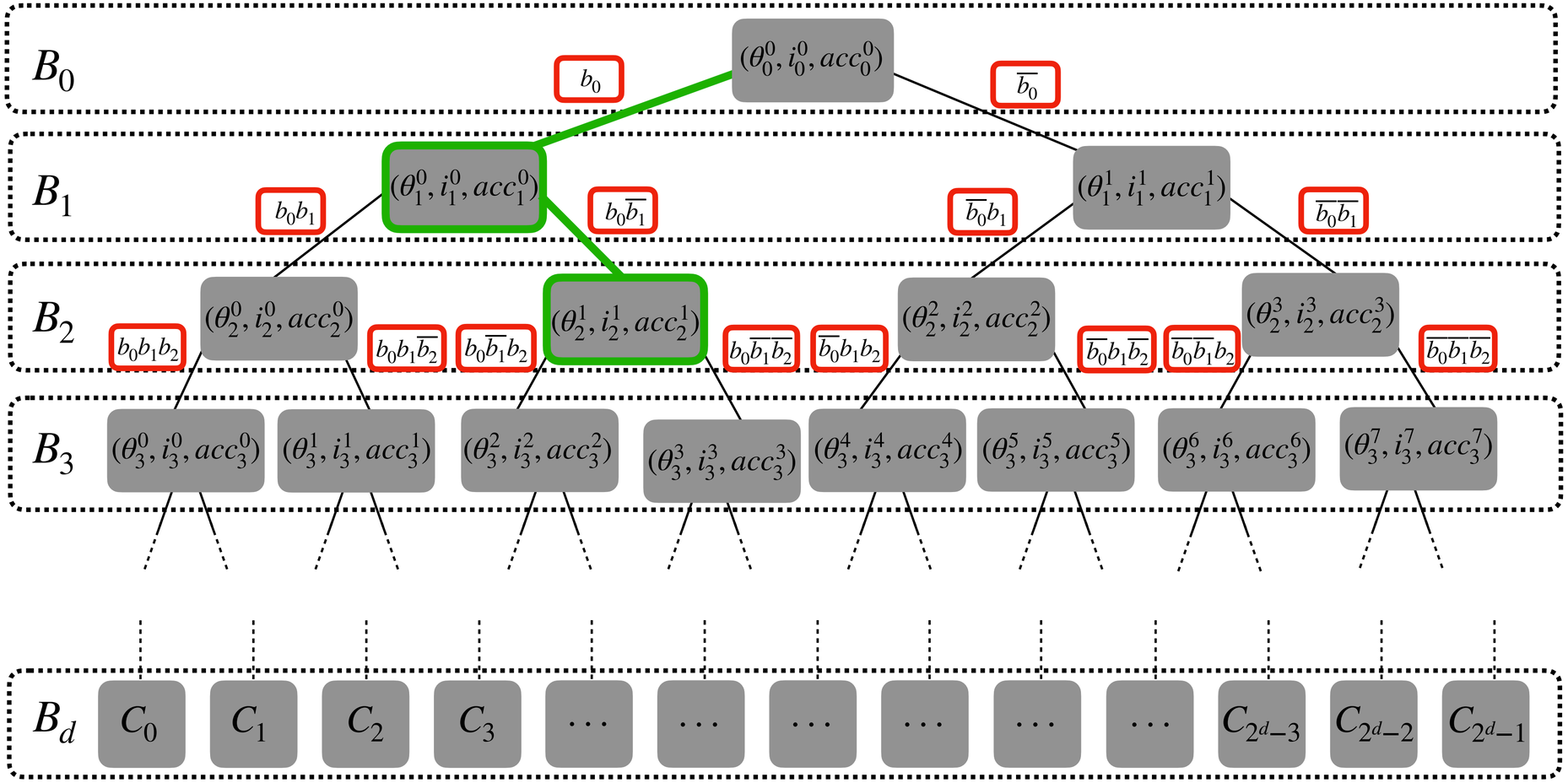
b_j = CMP(θ_j^ℓ , $F[i_j^\ell]$)

$B_{j+1}^{\ell'}$ = BlindNodeSelection(b_j , B_j , B_{j+1})

Client's Features



Server's Decision Tree

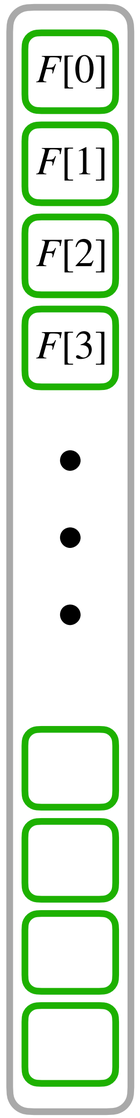


$F[i_j^\ell]$ = BlindArrayAccess(i_j^ℓ)

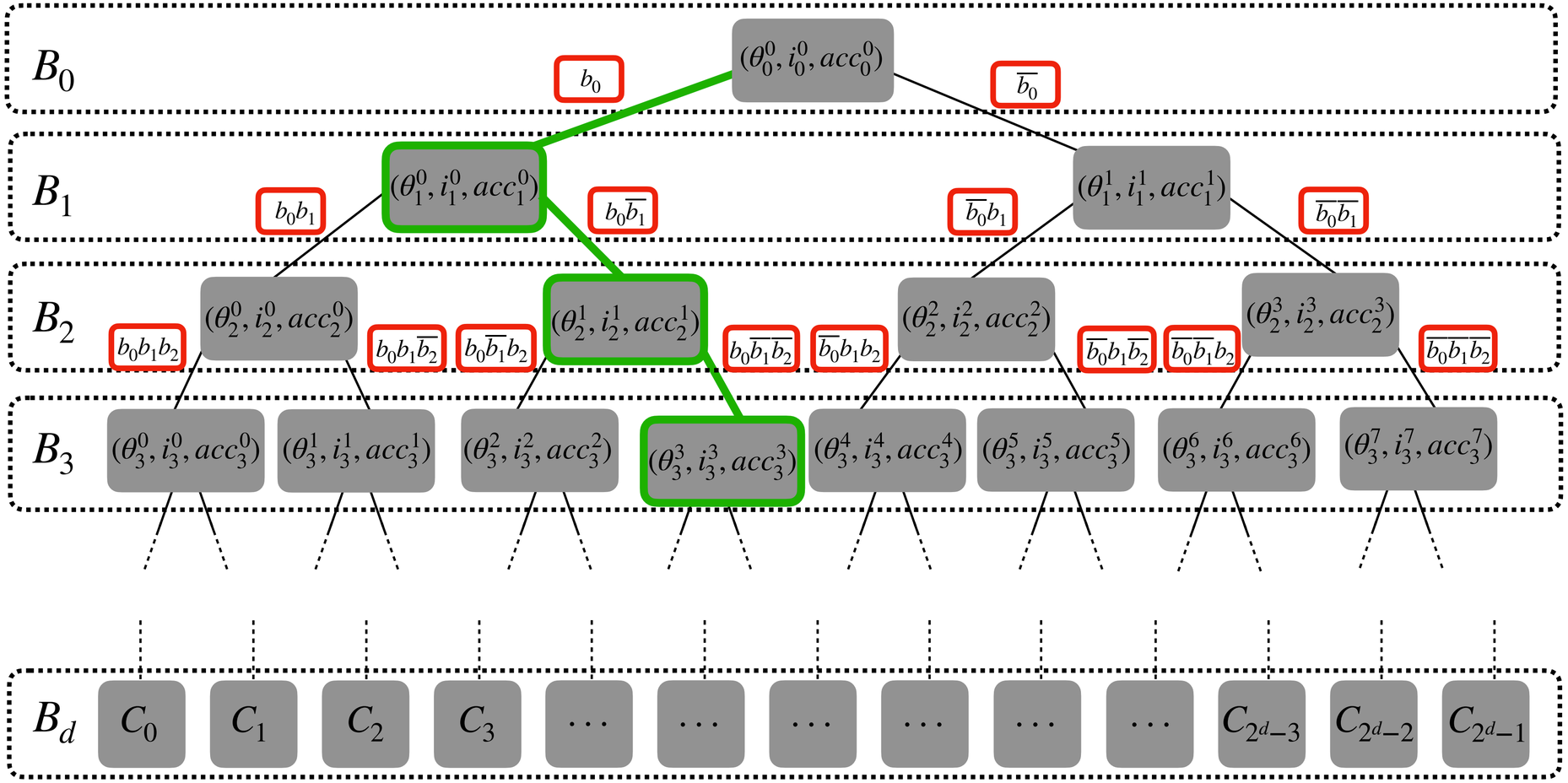
b_j = CMP(θ_j^ℓ , $F[i_j^\ell]$)

$B_{j+1}^{\ell'}$ = BlindNodeSelection(b_j , B_j , B_{j+1})

Client's Features



Server's Decision Tree

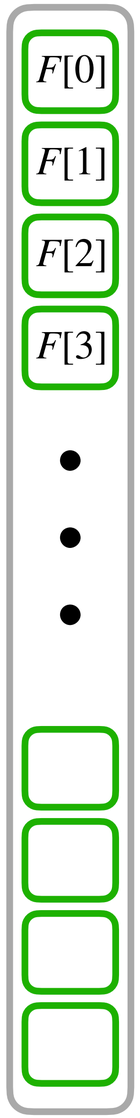


$F[i_j^\ell]$ = BlindArrayAccess(i_j^ℓ)

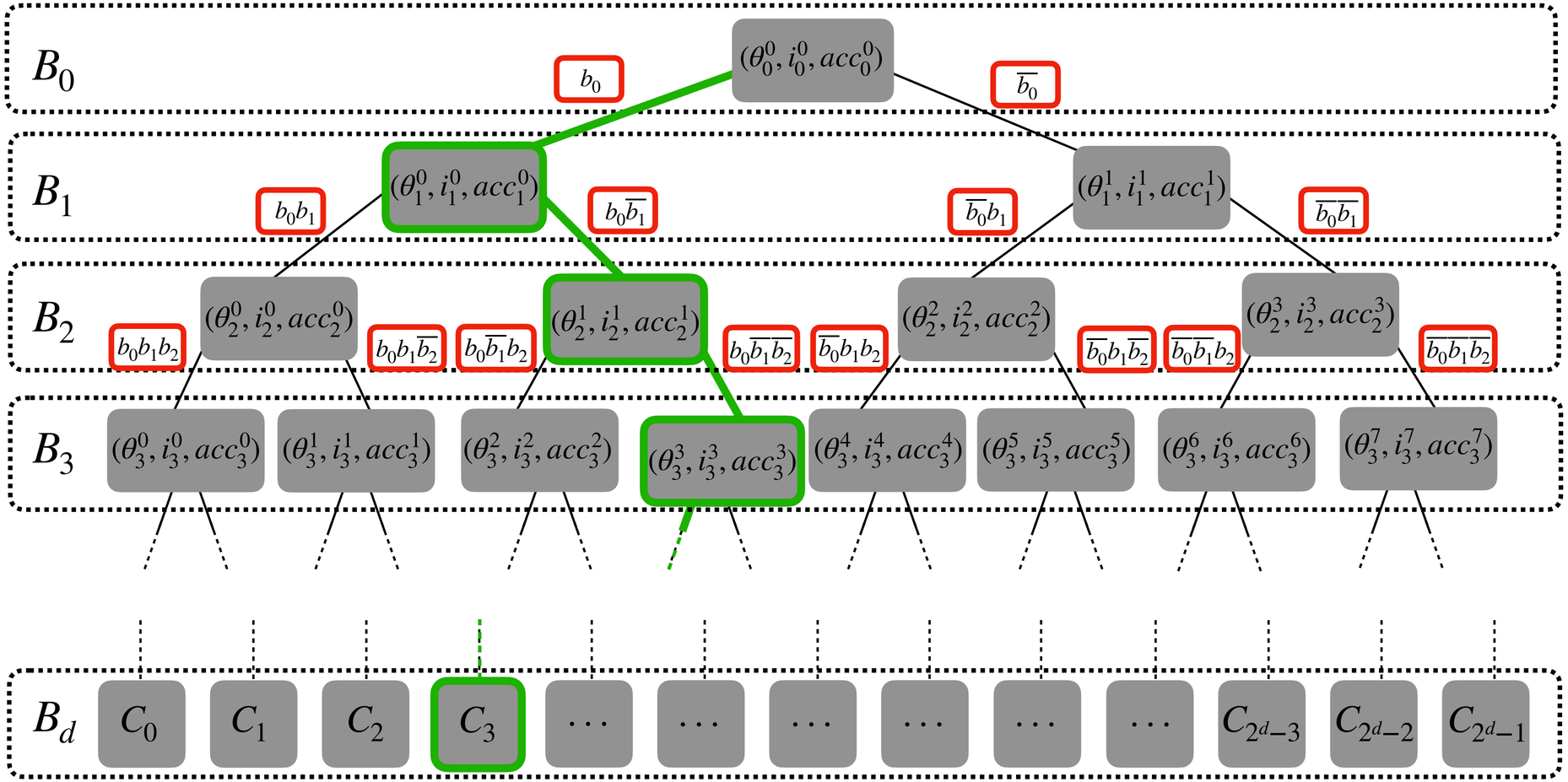
b_j = CMP(θ_j^ℓ , $F[i_j^\ell]$)

$B_{j+1}^{\ell'}$ = BlindNodeSelection(b_j , B_j , B_{j+1})

Client's Features



Server's Decision Tree



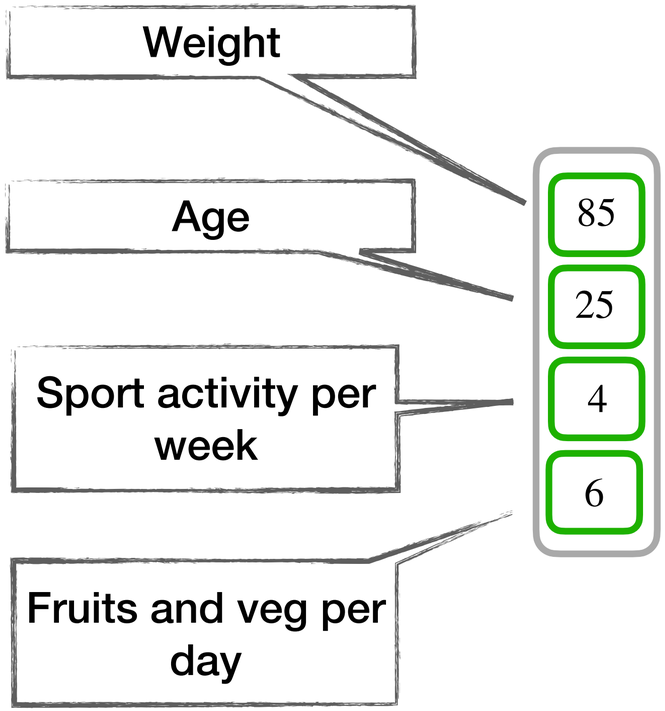
$F[i_j^\ell]$ = BlindArrayAccess(i_j^ℓ)

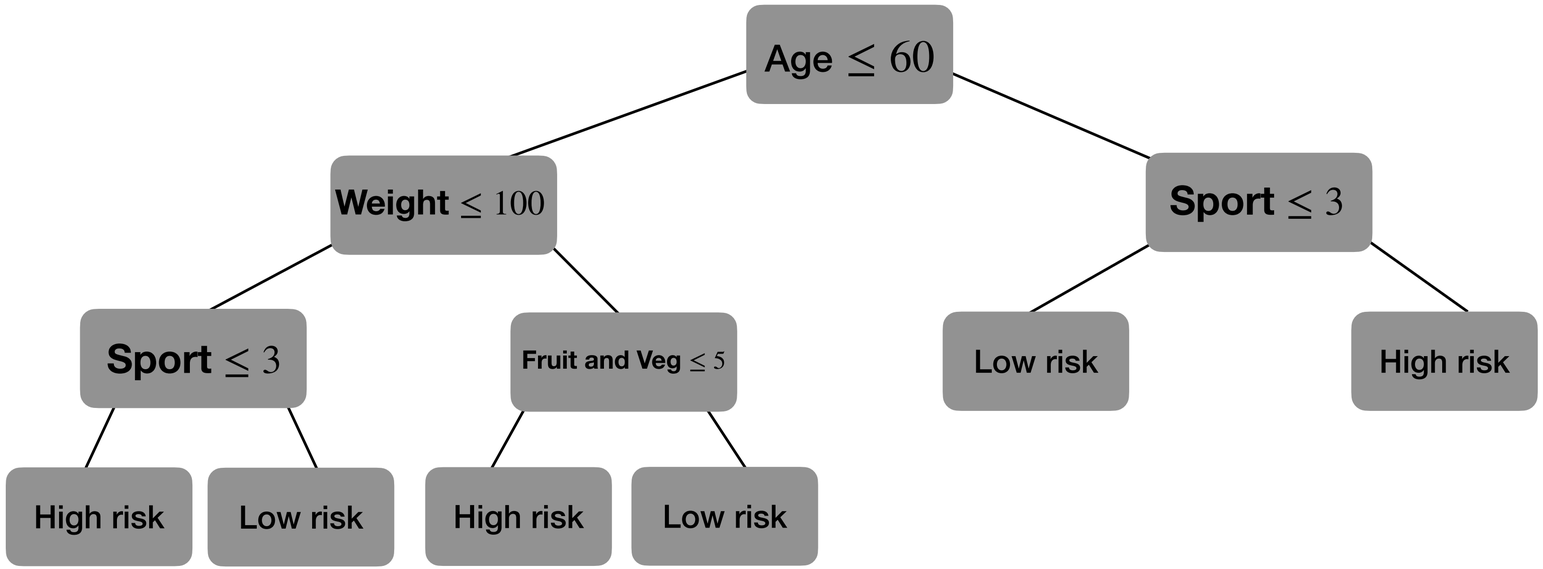
b_j = CMP(θ_j^ℓ , $F[i_j^\ell]$)

$B_{j+1}^{\ell'}$ = BlindNodeSelection(b_j , B_j , B_{j+1})

Our proposal

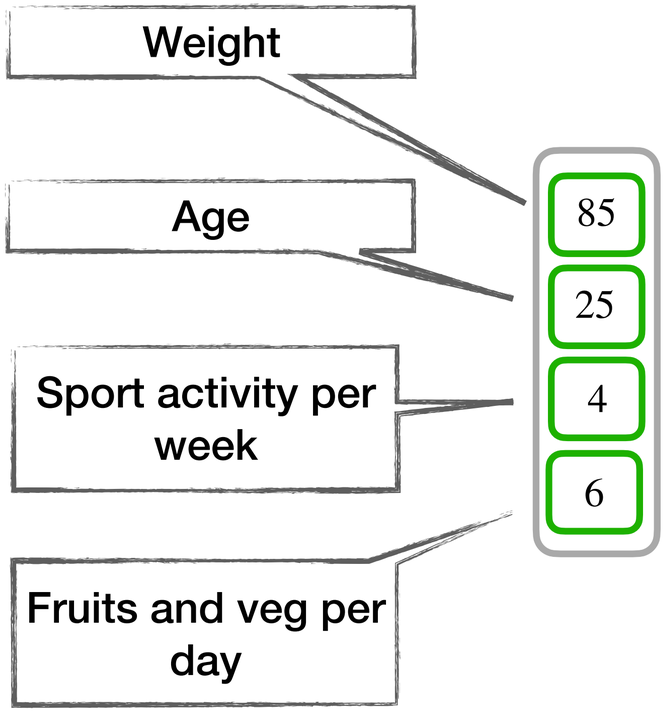
Toy example

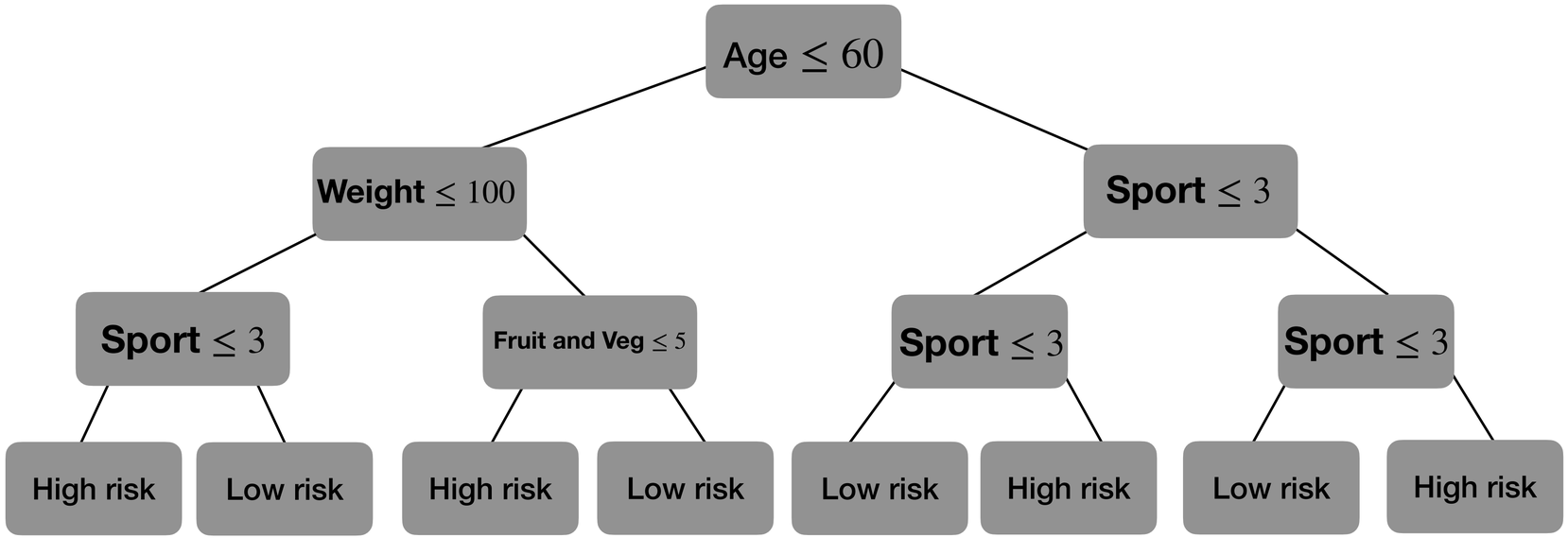




Our proposal

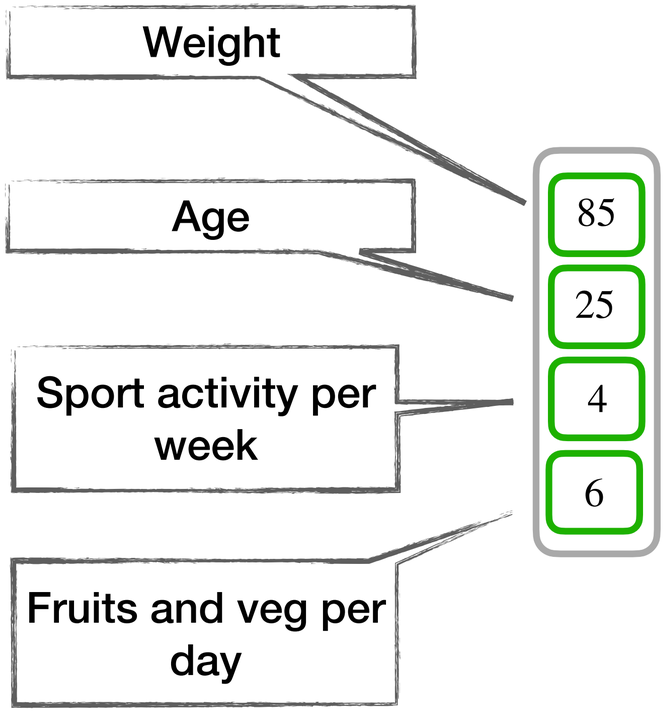
Toy example

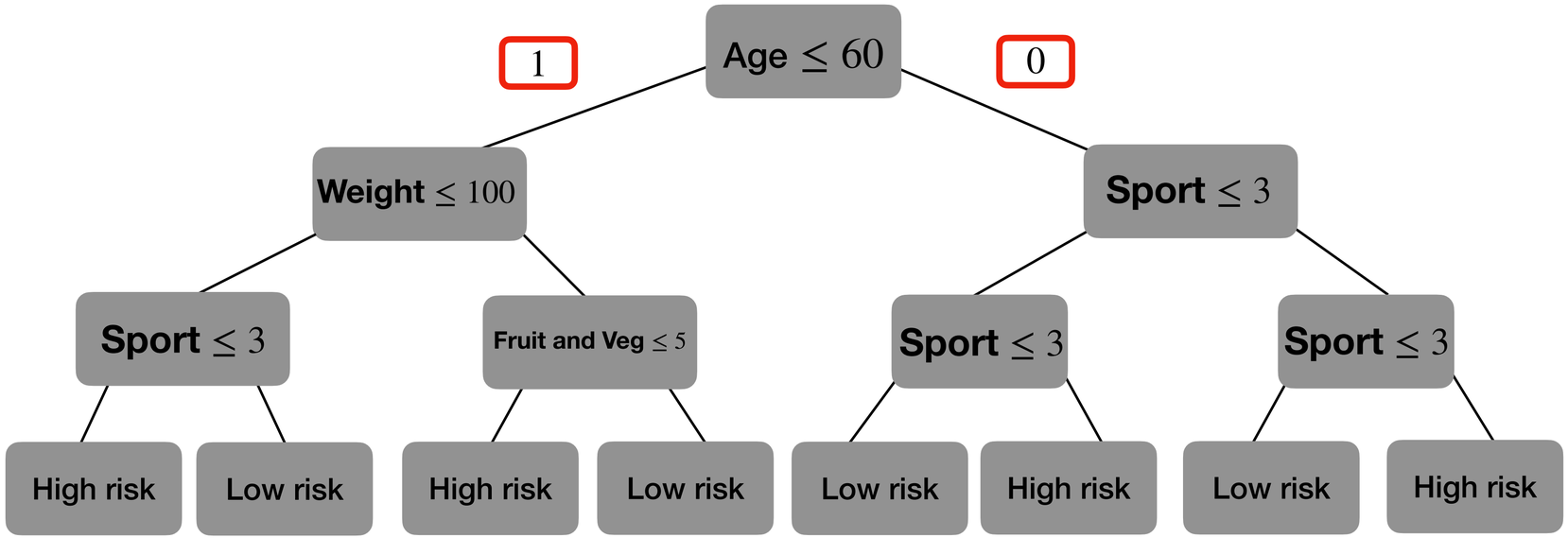




Our proposal

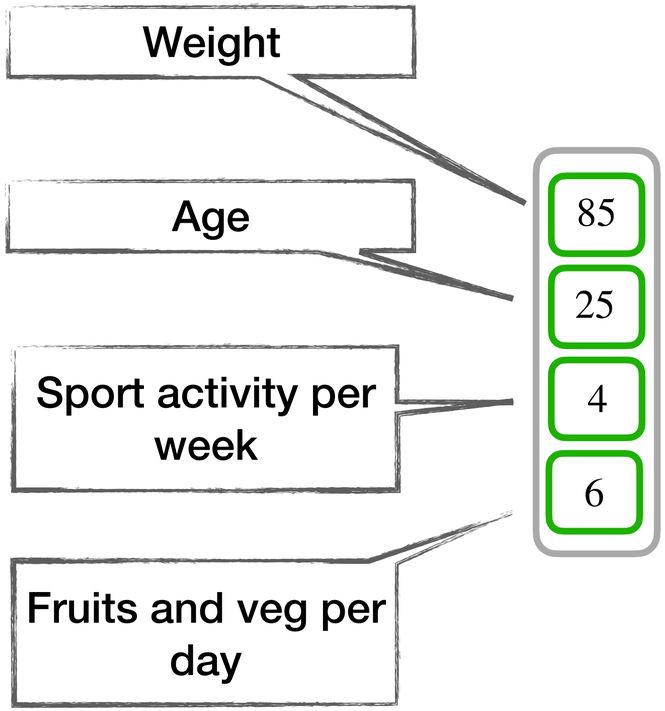
Toy example

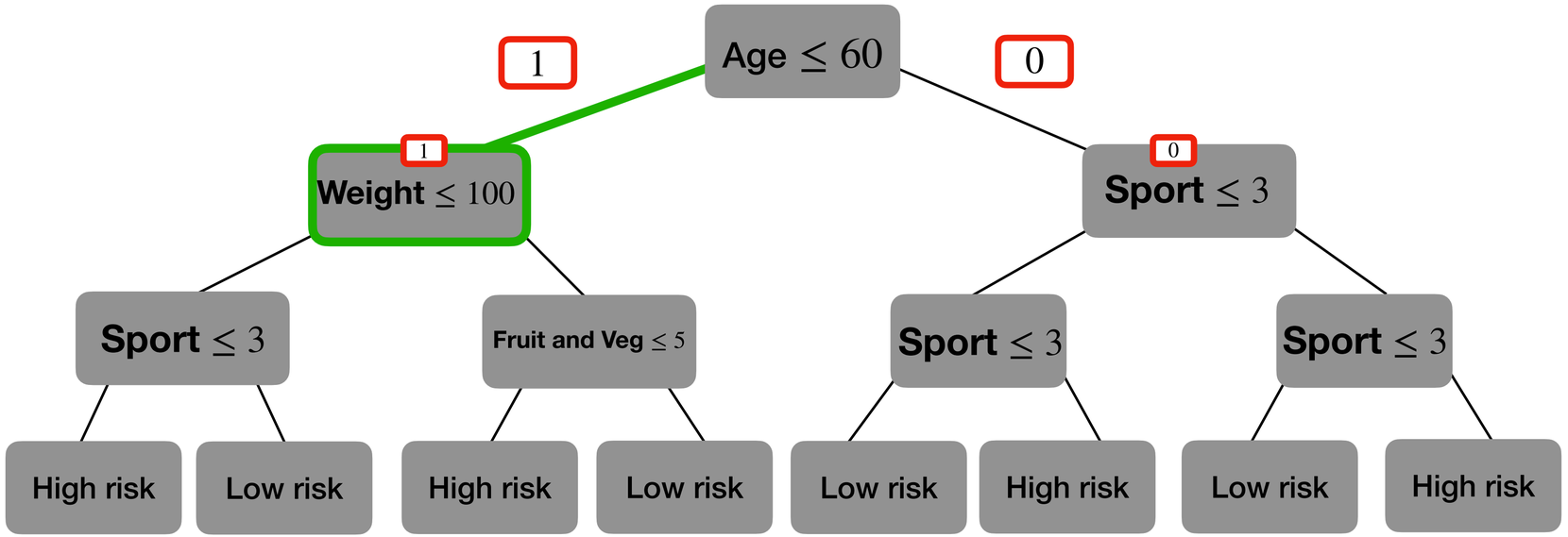




Our proposal

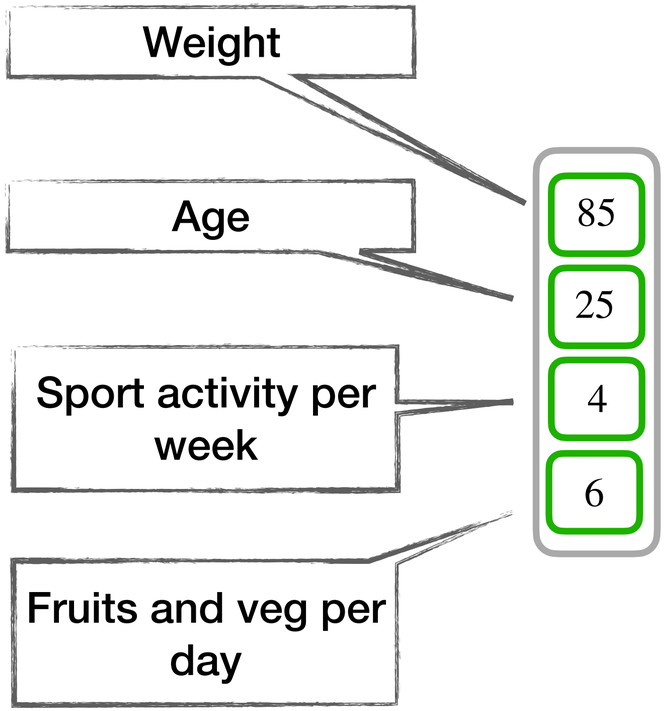
Toy example

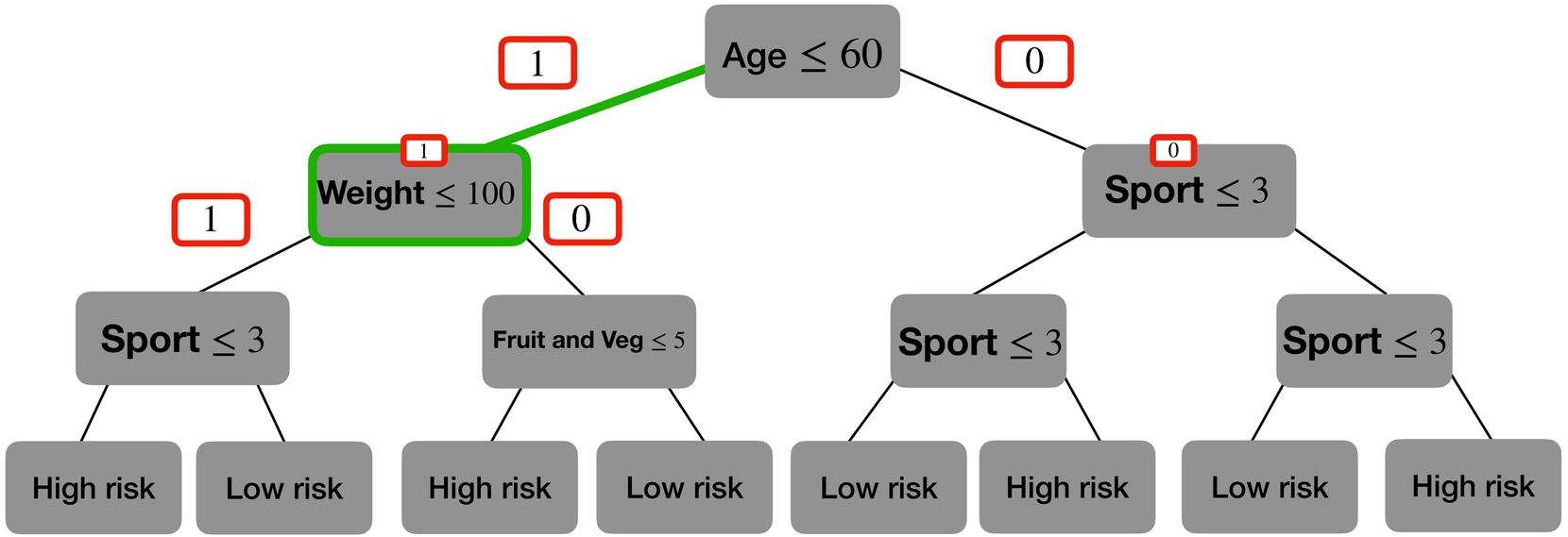




Our proposal

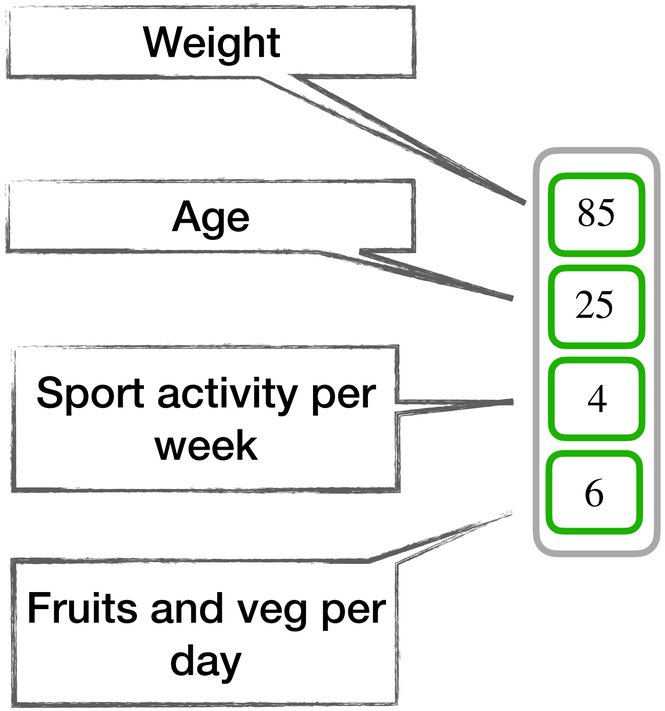
Toy example

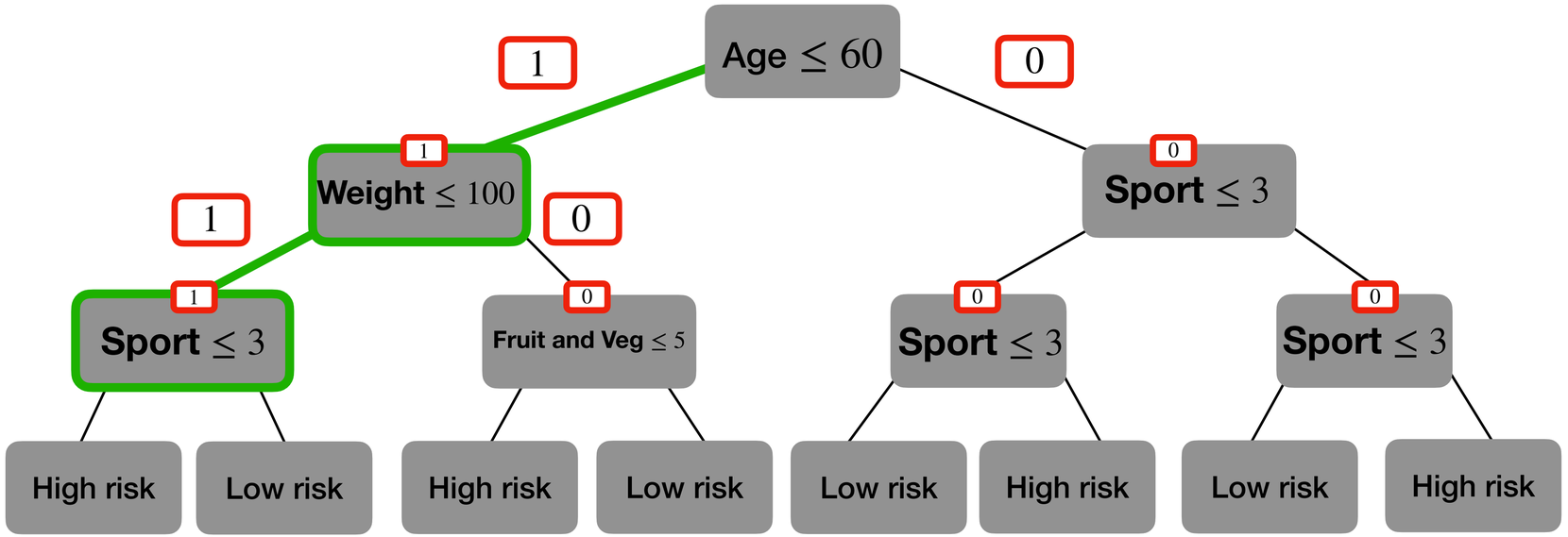




Our proposal

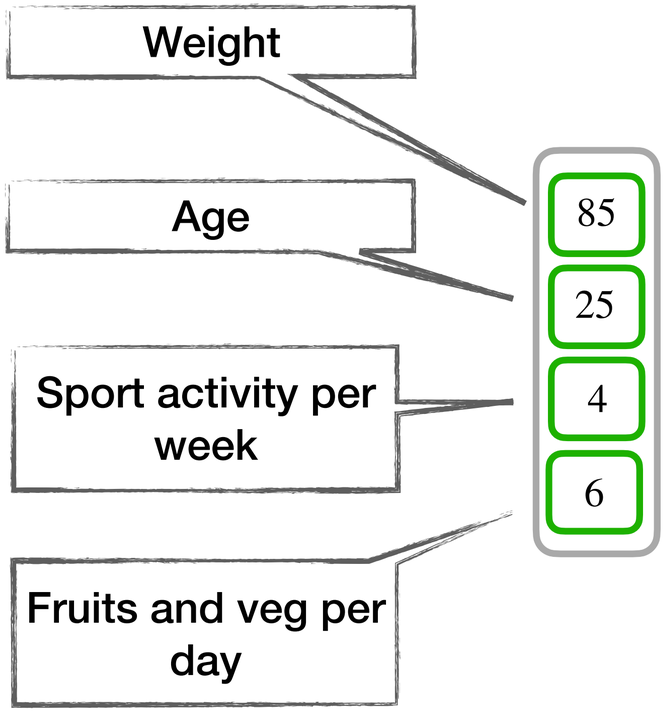
Toy example

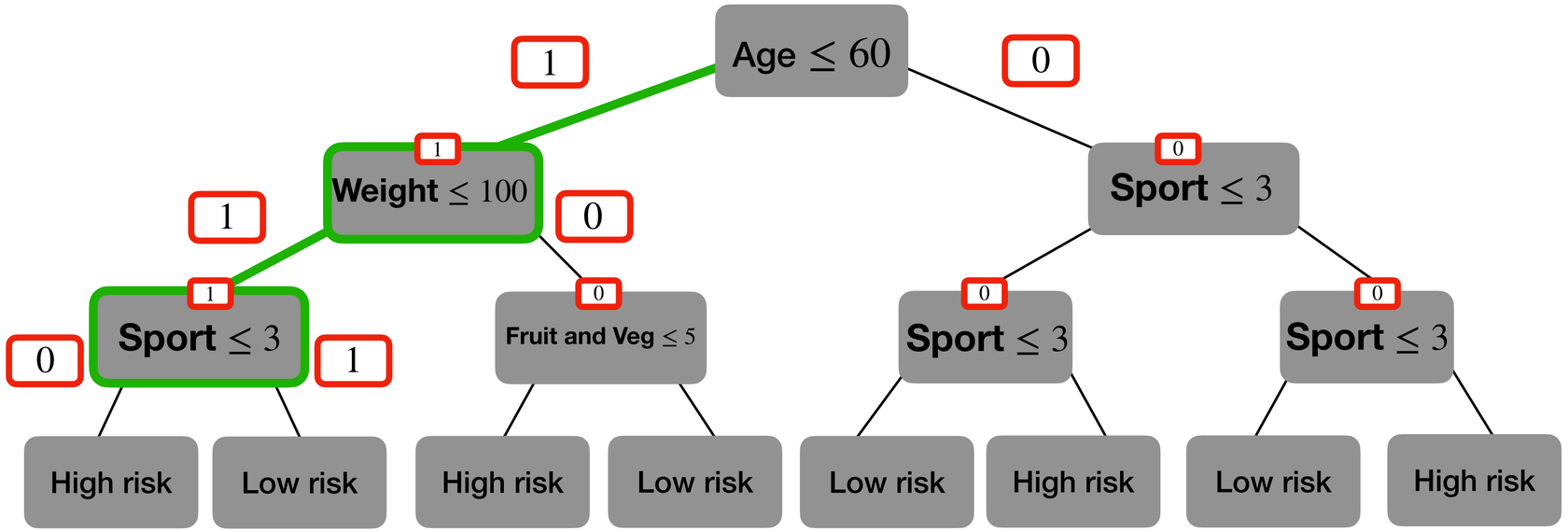




Our proposal

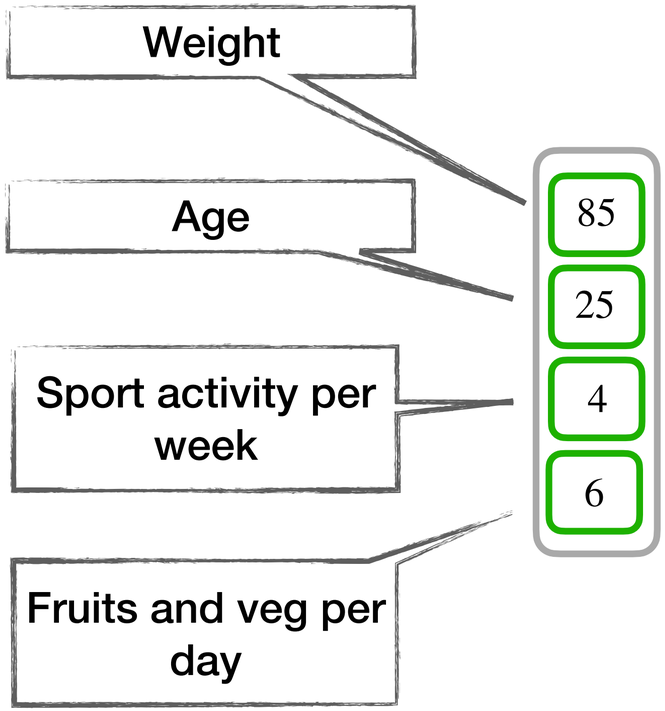
Toy example

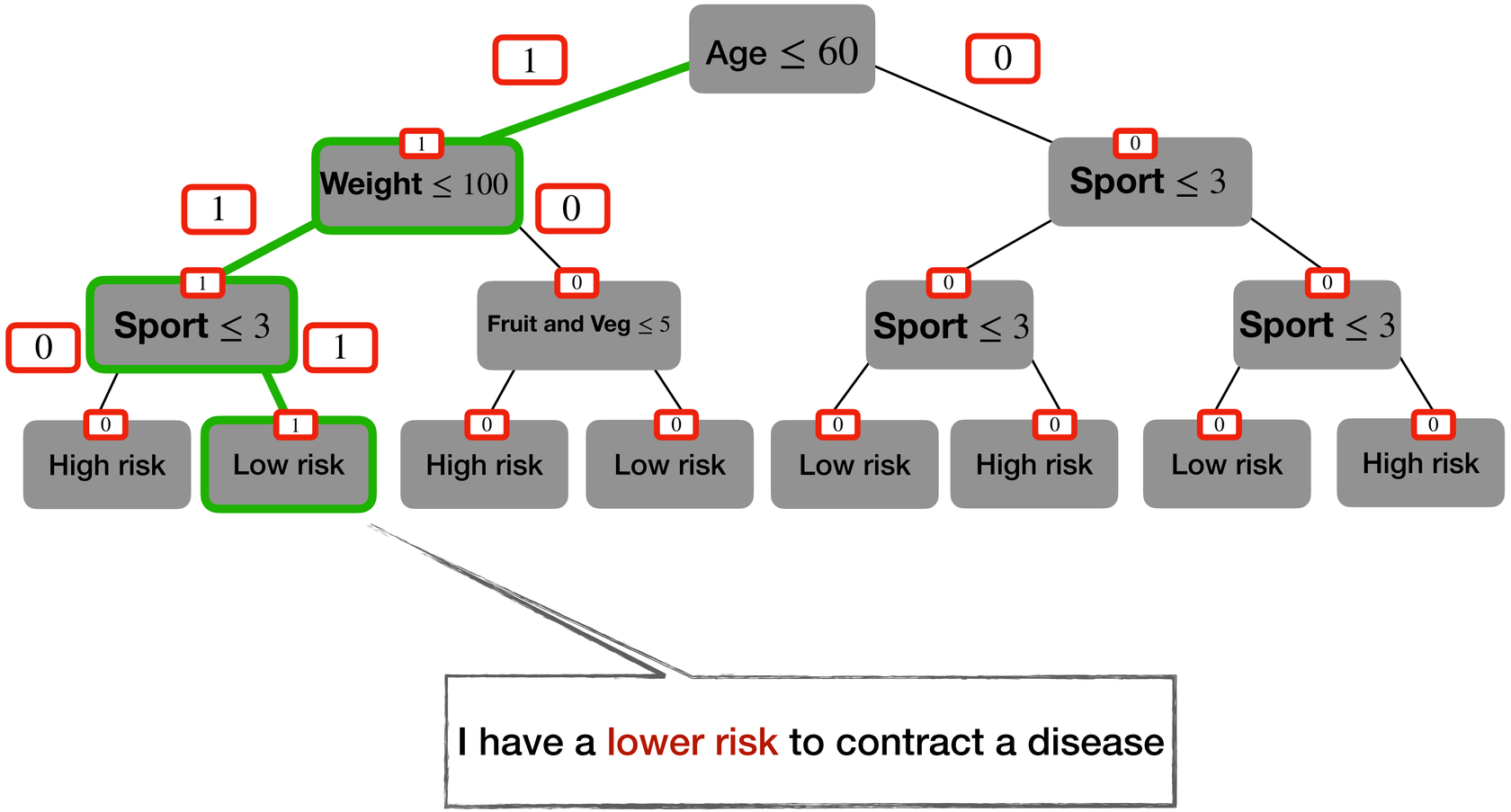




Our proposal

Toy example





Conclusions and Perspectives

PROBONITE :

- A simple yet effective protocol for decision tree evaluation
- Based on homomorphic encryption and non-interactive
- Reduces the number of comparisons to its bare minimum
- Two new primitives : Blind Array Access and Blind Node Selection

Conclusions and Perspectives

Perspectives :

- Implementation with a FHE library
- Packing techniques
- Use efficient private comparison to improve the protocol

Thanks ! 😄

Any Questions ?

azogagh.sofiane@courrier.uqam.ca