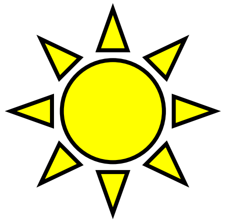
# Approximating Boosted Decision Trees with Differential Privacy

Thorsten Peinemann

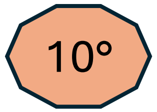
My personal website: tpein160.github.io

UNIVERSITÄT ZU LÜBECK
INSTITUTE FOR IT SECURITY
PRIVACY & SECURITY GROUP

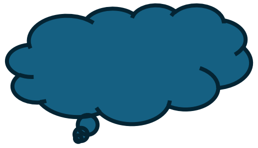
# Temperature prediction



Sun in the last hour, yes/no?



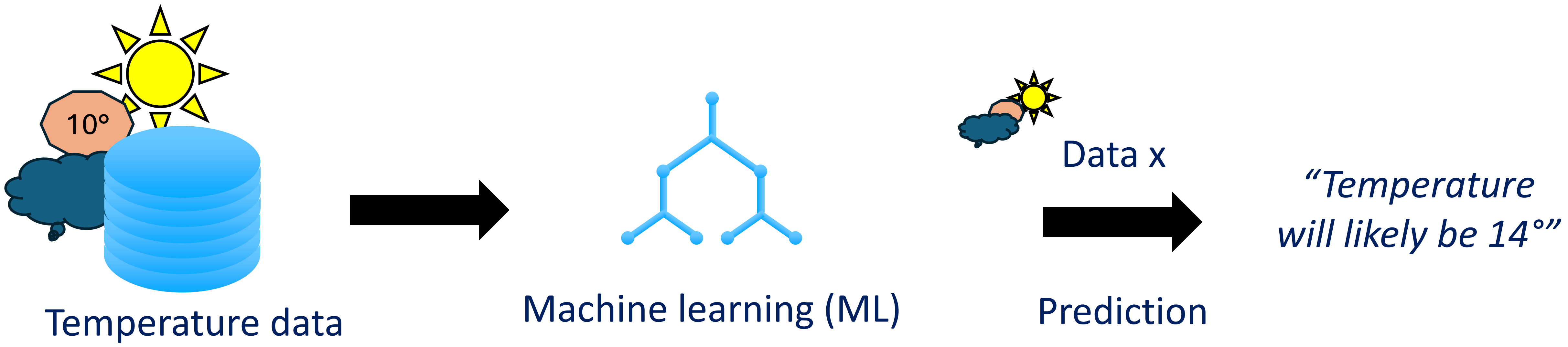

Current temperature in celsius



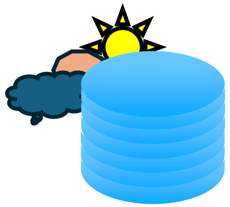
Rain in the last hour, yes/no?

Predict temperature
in 3 hours

# Temperature prediction using boosted decision tree (BDT) model



# **Non-private training** of BDT model

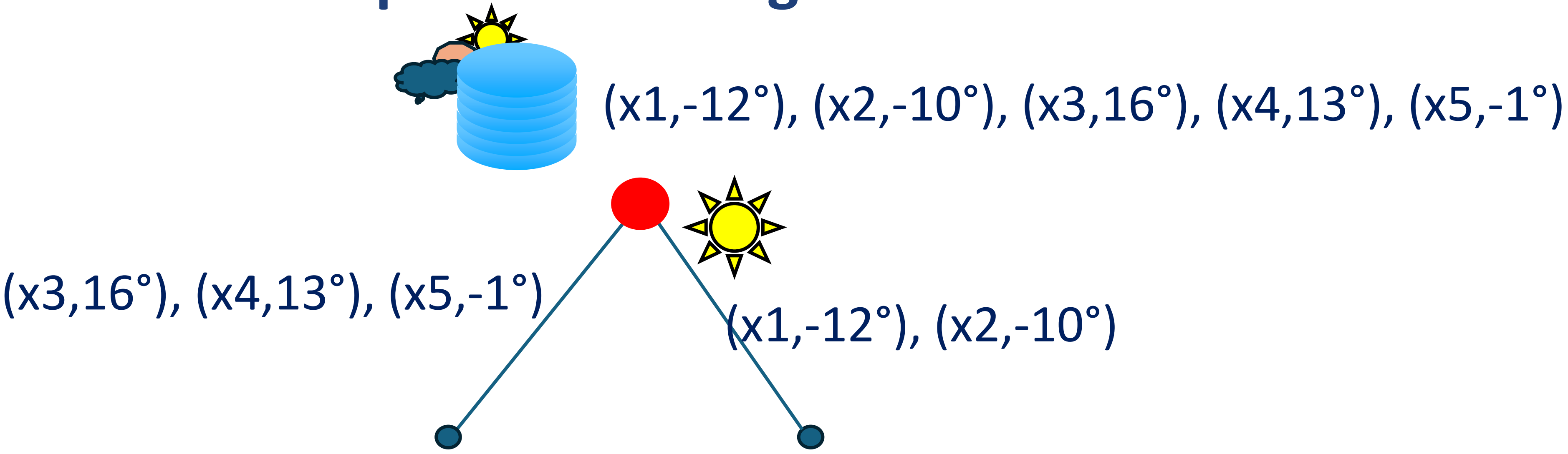


(x1,-12°), (x2,-10°), (x3,16°), (x4,13°), (x5,-1°)

(Data, Label)

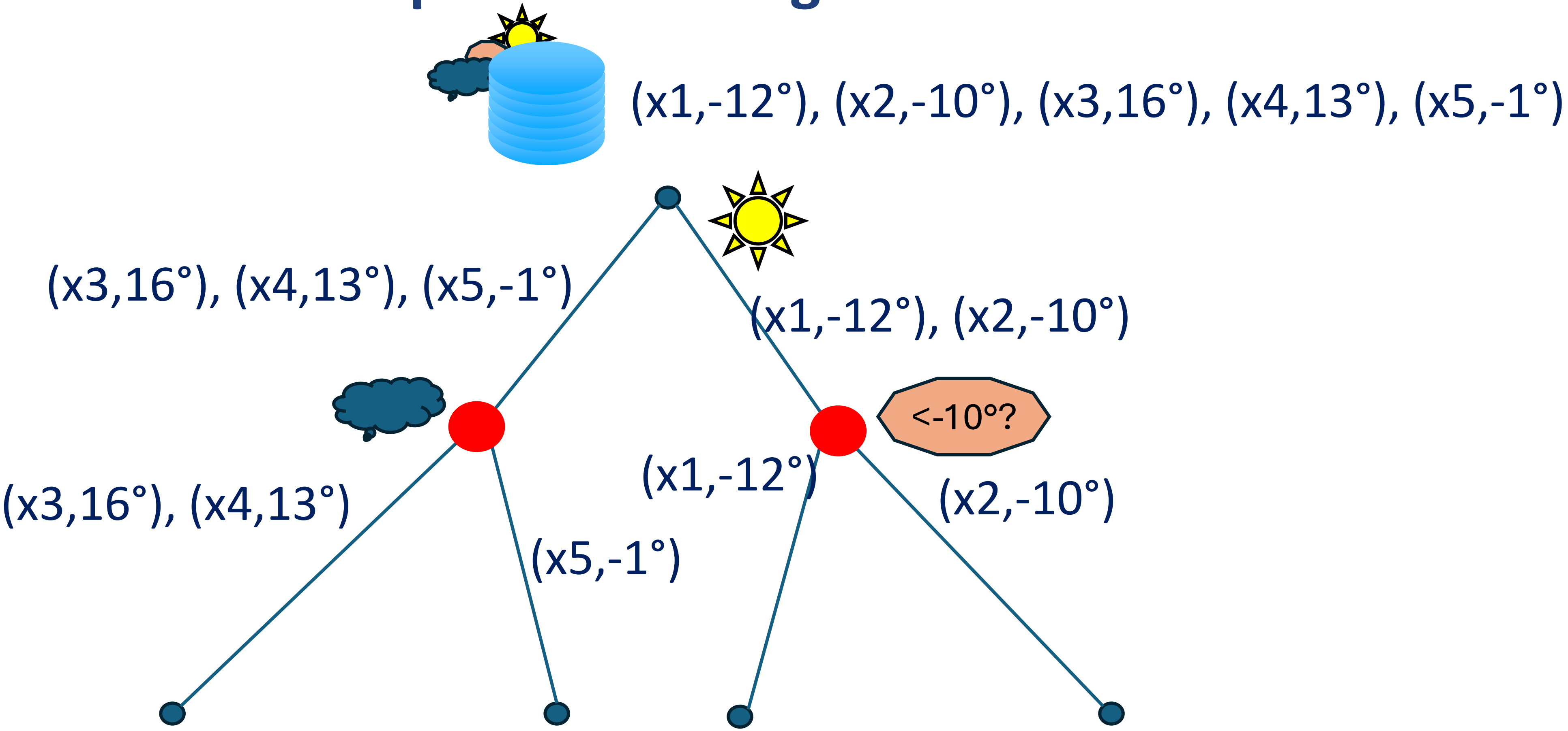
Root

# **Non-private training** of BDT model



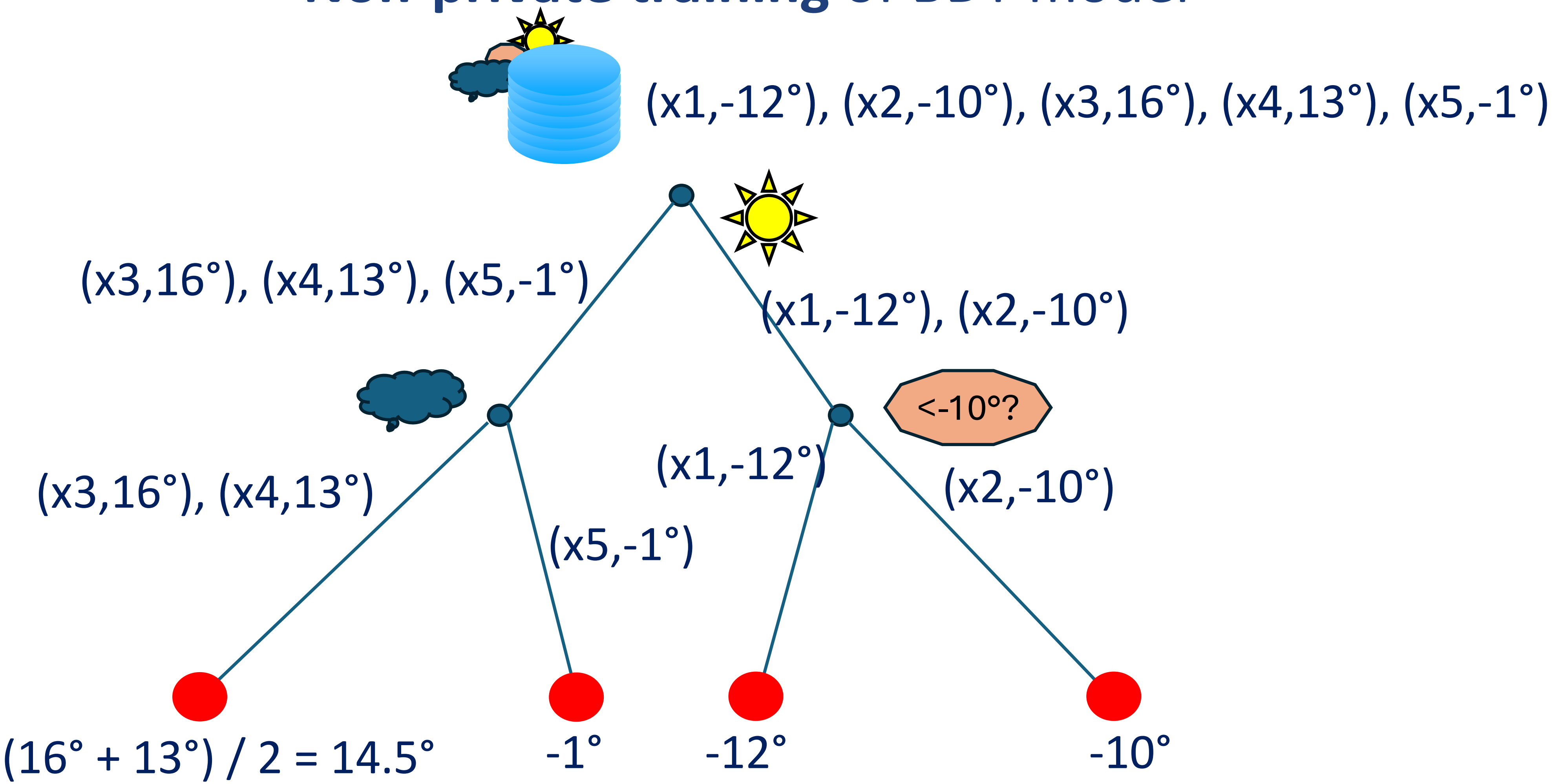

Each node splits the data in two subsets so that each subset groups together alike labels (e.g. gini-coefficient)

# **Non-private training** of BDT model



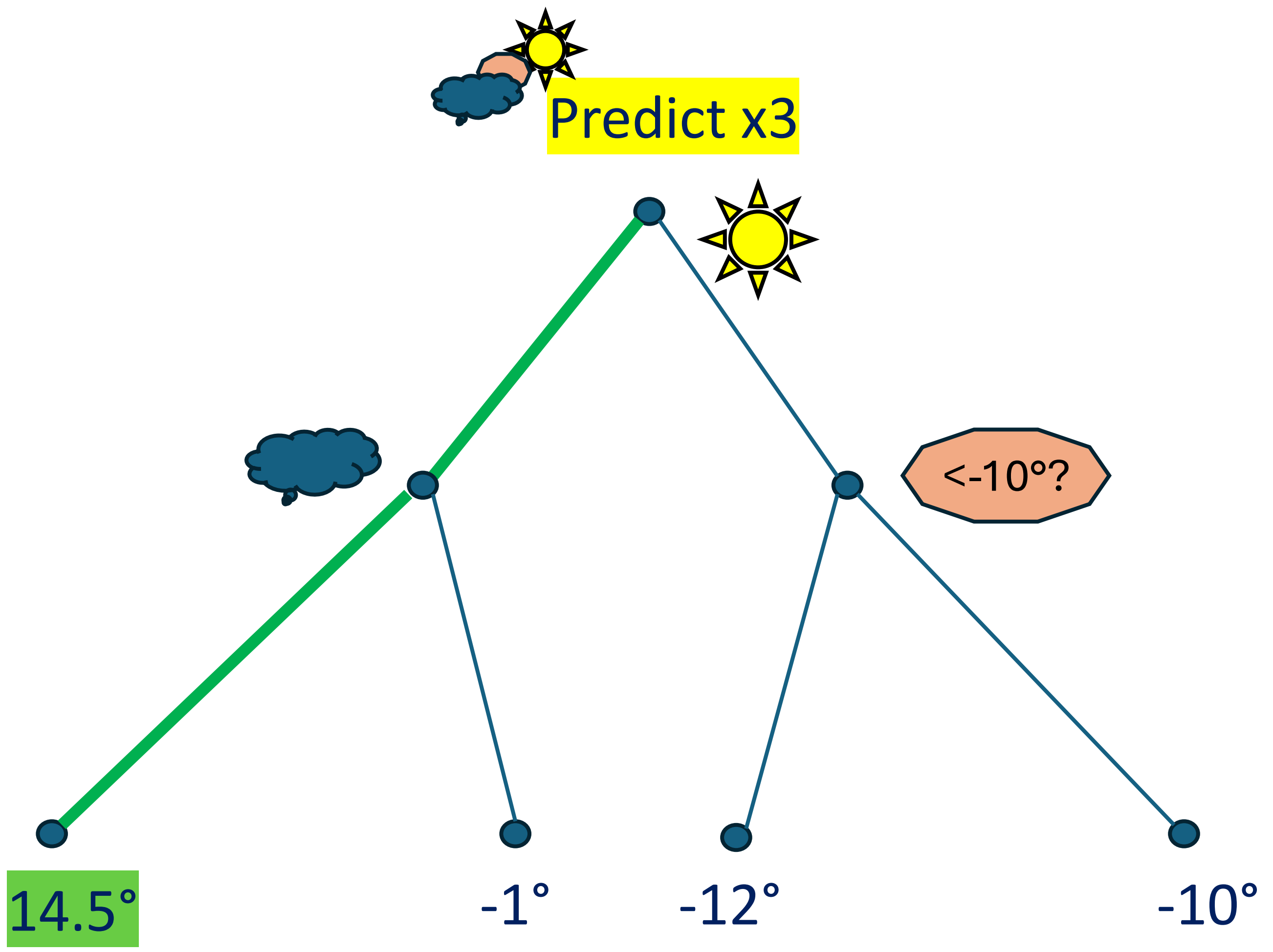

Each node splits the data in two subsets so that each subset groups together alike labels (e.g. gini-coefficient)

# **Non-private training** of BDT model

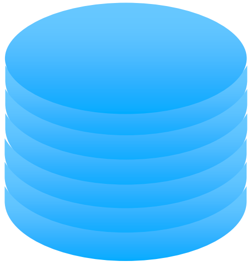



A leaf stores the average label of data points in that leaf

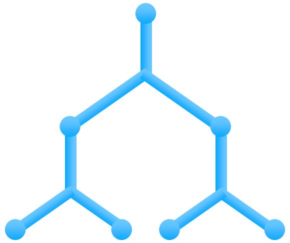
# Prediction of BDT model



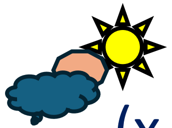
# **Error correction** for iterative BDT training



(x, 16°)



1st round machine learning (ML)



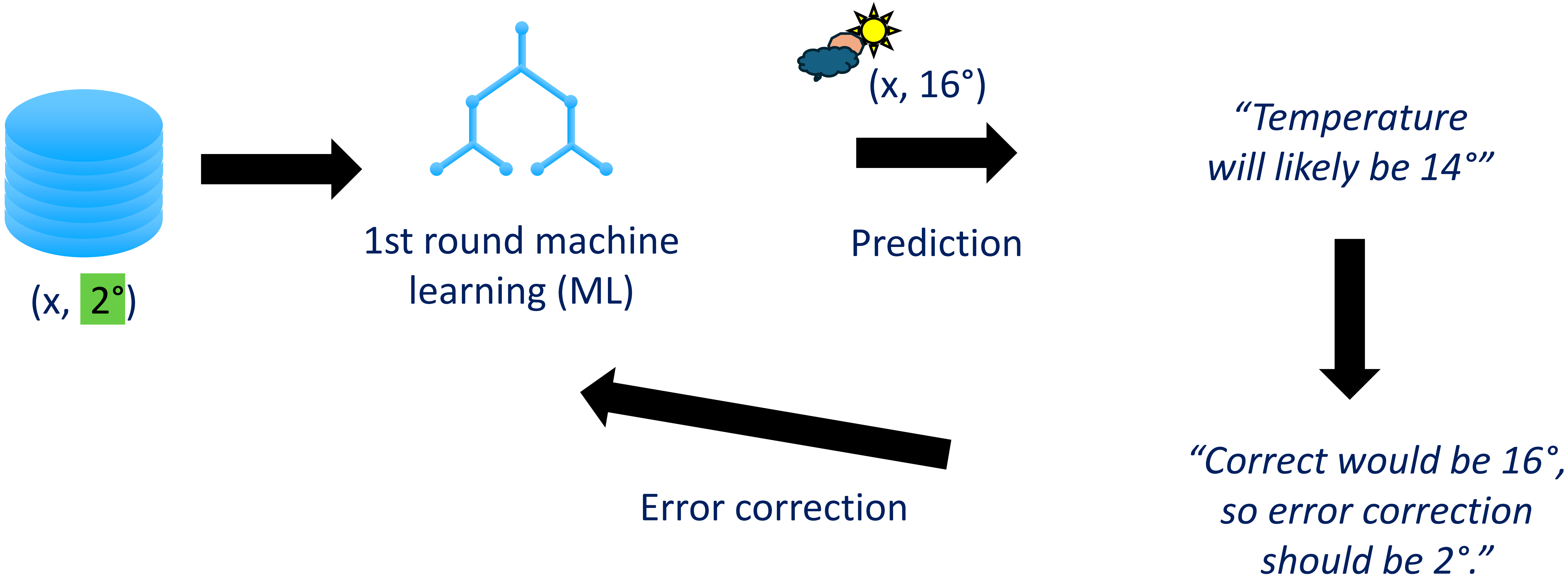
(x, 16°)

Prediction

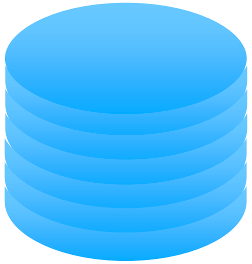
*“Temperature will likely be 14°”*

*“Correct would be 16°, so error correction should be 2°.”*

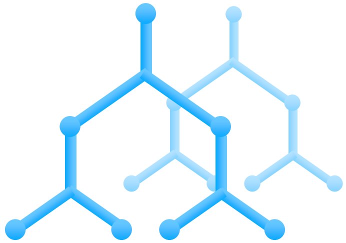
# **Error correction** for iterative BDT training



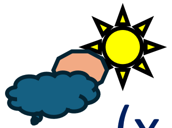
# **Error correction** for iterative BDT training



(x, 2°)



2nd round machine learning (ML)



(x, 16°)

Prediction

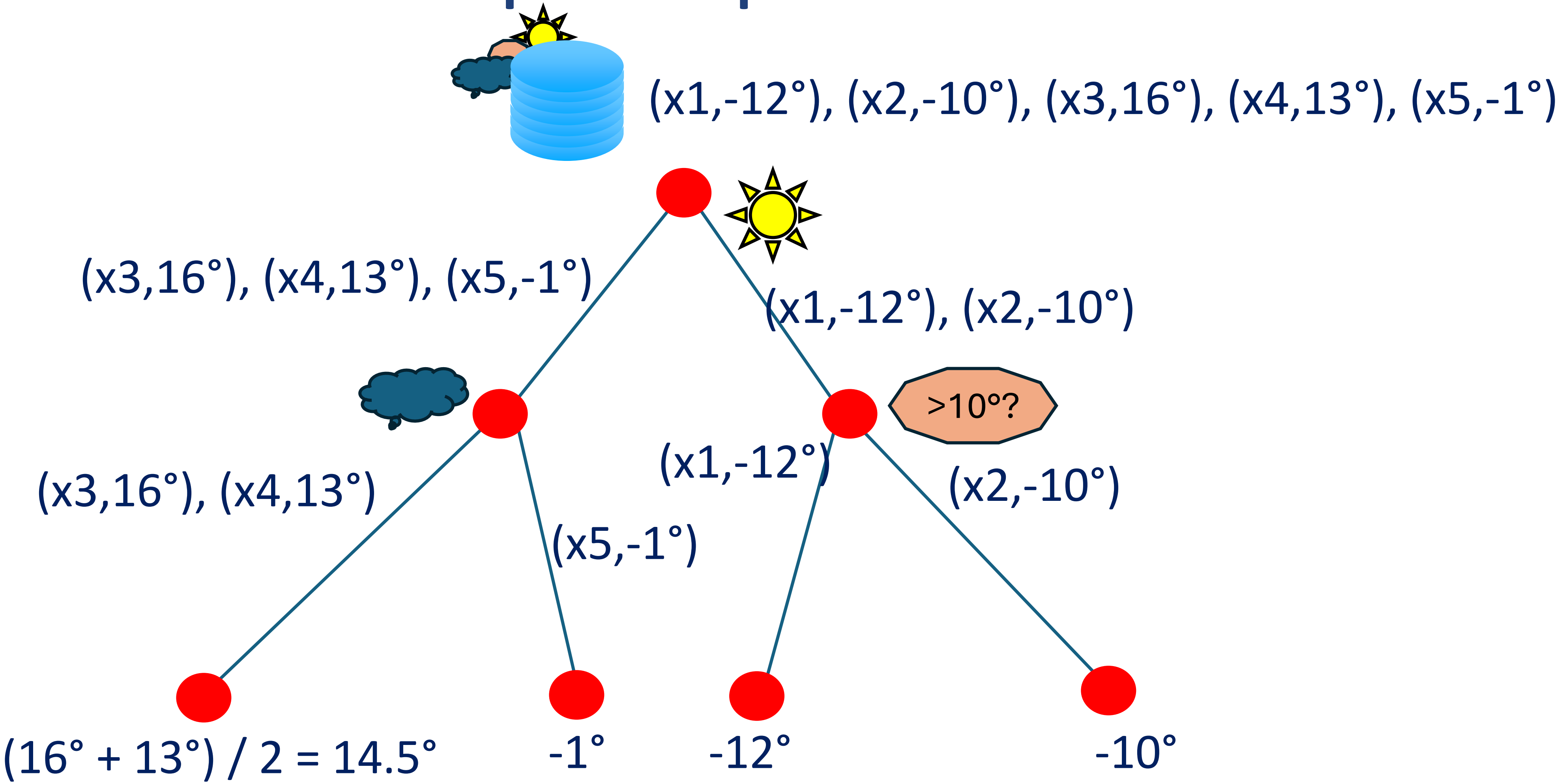
*"Temperature will likely be 15°"*

*"Correct would be 16°, so error correction should be 1°."*

# Differentially private boosted decision trees

*Whence cometh the noise?*

# **Data-dependent part** of BDT model



# **Algorithms** for splitting and leaves

**Splitting**

From top to bottom:

Split each node until maximum depth

Split such that equal
error-corrected labels are grouped

**Leaves**

For each leaf:

Find all data points in this leaf

Sum their error-corrected labels → S

Count number of data points → C

Store S/C

# DP-approximated splitting algorithm

**DP-Splitting**

From top to bottom:
- Split each node until max depth is reached
- Split randomly

**Leaves**

For each leaf:
- Find all data points in this leaf
- Sum their error-corrected labels → S
- Count number of data points → C
- Store S/C

# **DP-approximated** splitting and leaves algorithms

## DP-Splitting

From top to bottom:

Split each node until max depth is reached

Split randomly

## DP-Leaves

For each leaf:

Find all data points in this leaf

Clip their error-corrected labels to length **L**

Sum clipped error-corrected labels $\rightarrow S_c$

Add Gaussian noise: $S_c \rightarrow S_c'$

Count number of data points $\rightarrow C$

Add Gaussian noise: $C \rightarrow C'$

Store $S_c'/C'$

# **DP-Proof** for DP-approximated splitting and leaves algorithms

**DP-Splitting**

Output of randomized function has no leakage

# **DP-Proof** for DP-approximated splitting and leaves algorithms

## DP-Splitting

Output of randomized function has no leakage

## DP-Leaves

(ε, δ)-Differential Privacy (DP):

$Pr[M(D) \in S] <= e^{\varepsilon} Pr[M(D \cup \{x\}) \in S] + \delta$

DP-Leaves:

(1) Leakage for x occurs only in x's leaf $\mathbf{P_x}$

(2) $M(D) = \left(\sum_{(v,l) \in D:\, (v,l) \text{ in } \mathbf{P_x}} \text{clip}(l, (-L,L))\right) + N(0, \sigma^2)$

Gaussian Mechanism:

M satisfies (ε, δ)-DP for any δ > 0, ε in (0,1) when

σ > sqrt(2*ln(1.25/ δ))*L/ε

# Approximating Boosted Decision Trees with Differential Privacy

Thorsten Peinemann

My personal website: tpein160.github.io

UNIVERSITÄT ZU LÜBECK
INSTITUTE FOR IT SECURITY
PRIVACY & SECURITY GROUP