Novelty Detection in Convolutional Neural Networks Using Density Forests

Master Thesis

Cyril Wendl

Environmental Sciences and Engineering
EPFL

07/09/2018

Outline

What is Uncertainty?

Input image
$I \in \mathbb{R}^{h,w,n_c}$

$\Rightarrow$ Model M $\Rightarrow$

What is Uncertainty?

Input image
$I \in \mathbb{R}^{h,w,n_c}$

$\Rightarrow$ Model M $\Rightarrow$

Set of classes: $\mathcal{L} = \{c_i\}_{1 \leq i \leq n_c}$

c_1

$c_2 \ldots$

What is Uncertainty?

What happens to these cases?

What is Uncertainty?

What happens to these cases?

:)

What is Uncertainty?

What happens to these cases?

:)

What is Uncertainty?

What happens to these cases?

:)

:)

What is Uncertainty?

What happens to these cases?

:)

:)

What is Uncertainty?

What happens to these cases?

:)

:)

:(

What is Uncertainty?

What happens to this case?

Input image of
unseen class
"birchermüesli"

$\Rightarrow$ Model M $\Rightarrow$

What is Uncertainty?

What happens to this case?

Input image of
unseen class
"birchermüesli"

$\Rightarrow$ Model M $\Rightarrow$

Set of classes: $\mathcal{L} = \{c_1, c_2\}$

$p(c_1) = 1$

$p(c_2) = 0$

What is Uncertainty?

What happens to this case?

Input image of
unseen class
"birchermüesli"

$\Rightarrow$ Model M $\Rightarrow$

Set of classes: $\mathcal{L} = \{c_1, c_2\}$

$p(c_1) = 1$

$p(c_2) = 0$

This is not what we want :(

What is Uncertainty?

What happens to this case?

Input image of
unseen class
"birchermüesli"

$\Rightarrow$ Model M $\Rightarrow$

Set of classes: $\mathcal{L} = \{c_1, c_2\}$

$p(c_1) = 0.5$

$p(c_2) = 0.5$

This would be better :)

What is Uncertainty?

Uncertainty

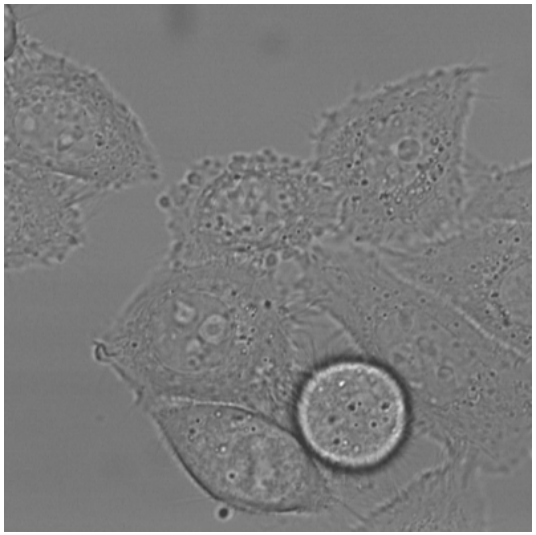
- Information on *confidence* of the model
- Ability to model incomplete information

Evaluation heuristics

- Error detection: wrong prediction $\Rightarrow$ low confidence
- Novelty detection: unseen class $\Rightarrow$ low confidence

Relevant applications of Uncertainty

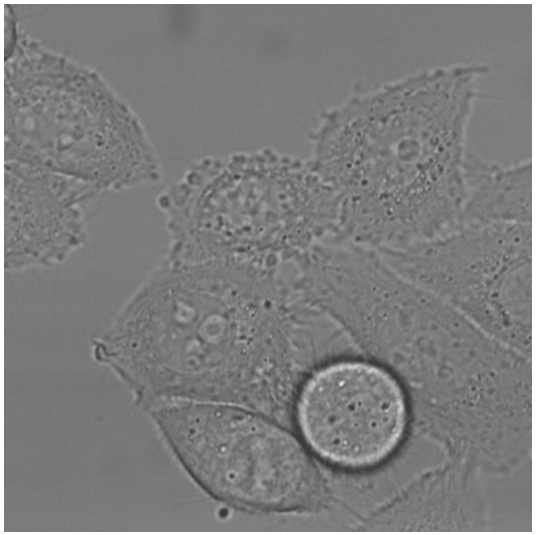
Other than yoghurt classification



Medical imaging

Relevant applications of Uncertainty

Other than yoghurt classification



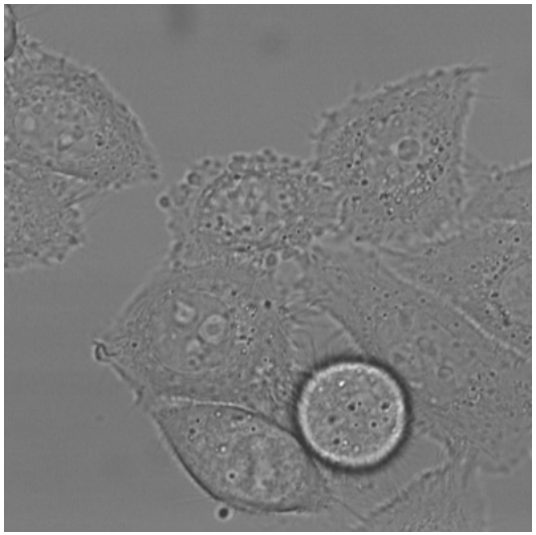
Medical imaging

TESLA

Autonomous cars

Relevant applications of Uncertainty

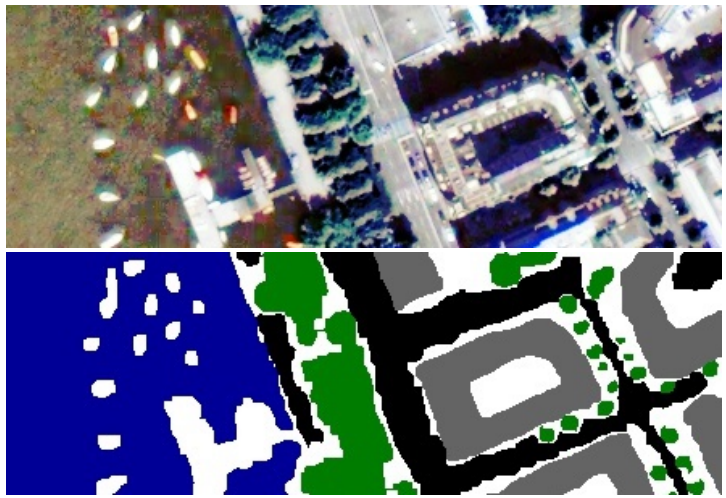
Other than yoghurt classification



Medical imaging

TESLA

Autonomous cars



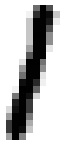
Land Cover Classification

How does Machine Learning work?
Goals

Terminology

- **Image classification**: For a given image attribute one class label, i.e.:

 ⇒ cat  ⇒ dog

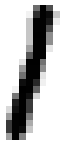
 ⇒ “1”  ⇒ “7”

How does Machine Learning work?
Goals

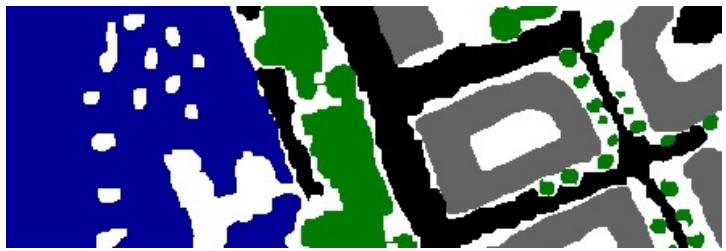
Terminology

- **Image classification**: For a given image attribute one class label, i.e.:

 ⇒ cat  ⇒ dog

 ⇒ “1”  ⇒ “7”

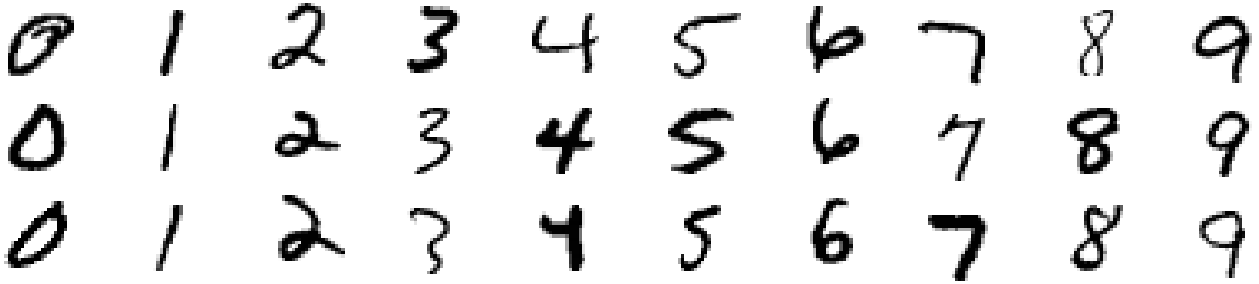
- **Semantic segmentation**: segmentation of an image into class labels

 ⇒ 

How does Machine Learning work?

Training

Dataset:



Labels: 0, 1, 2, 3, 4, 5, 6, 7, 8, 9

$\Downarrow$

Training (+ Validation)

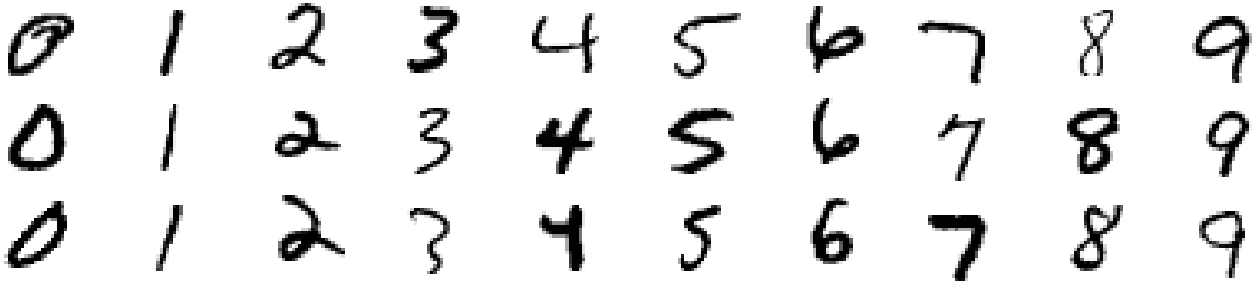
$\Downarrow$

Model M

How does Machine Learning work?

Training

Dataset:



Labels: 0, 1, 2, 3, 4, 5, 6, 7, 8, 9

⇓

Training (+ Validation)

⇓

Model M

Prediction

New image:

⇓

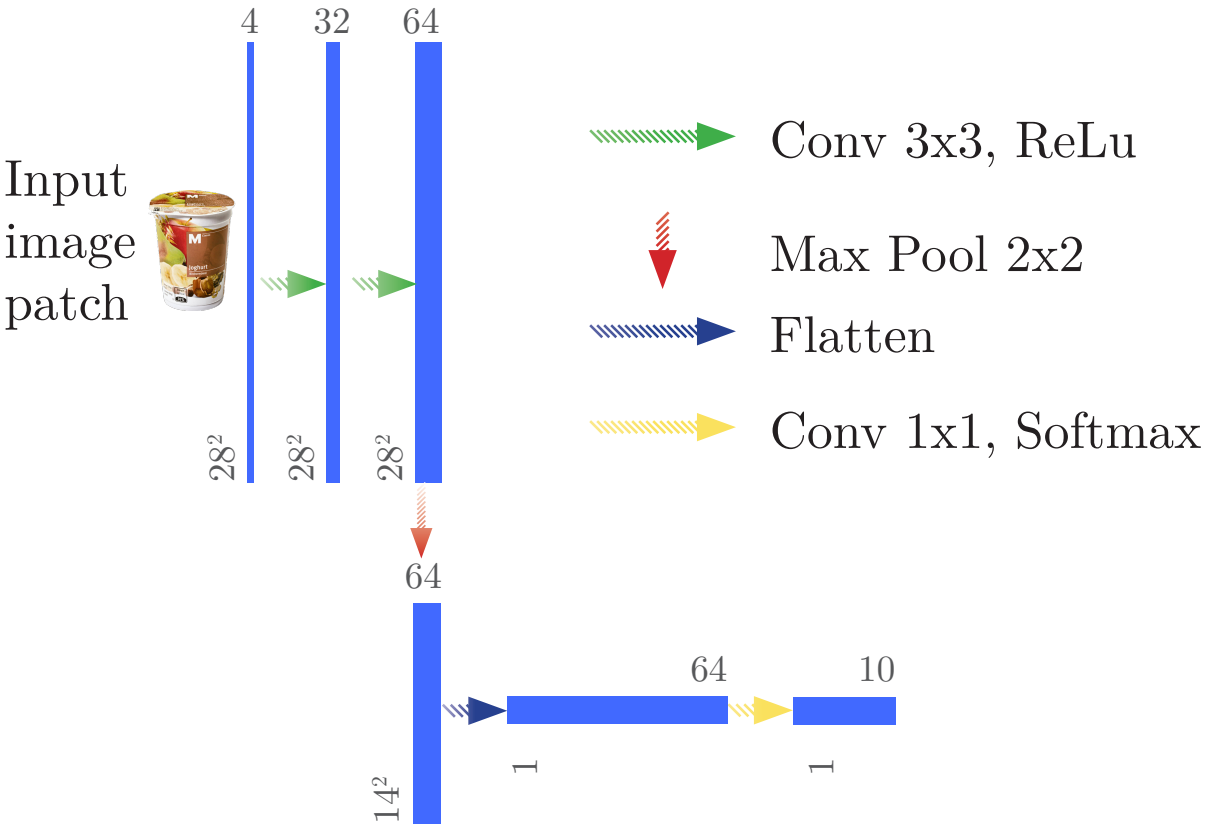
Prediction using model M

⇓

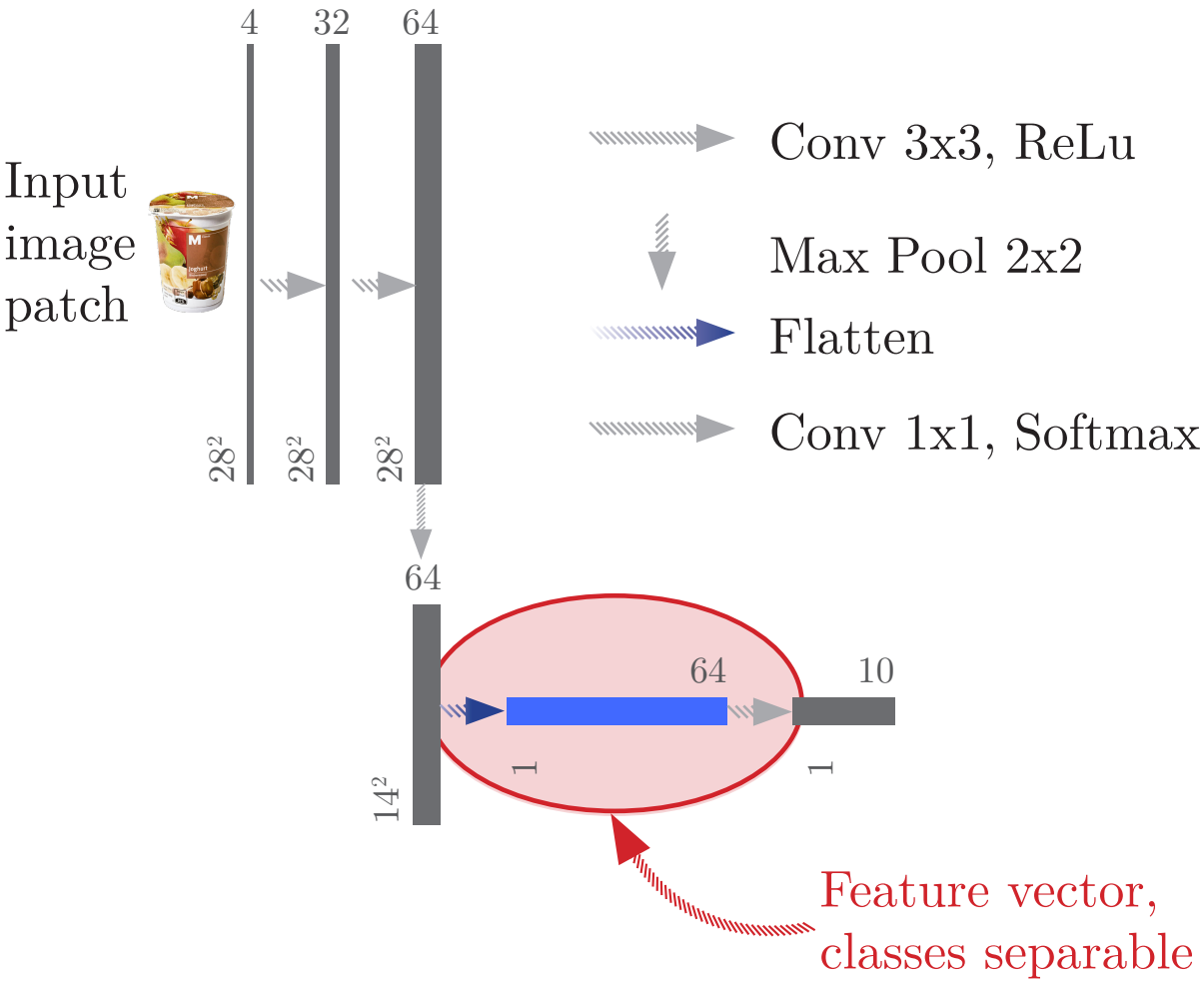
Label: 1

Classification vs. regression

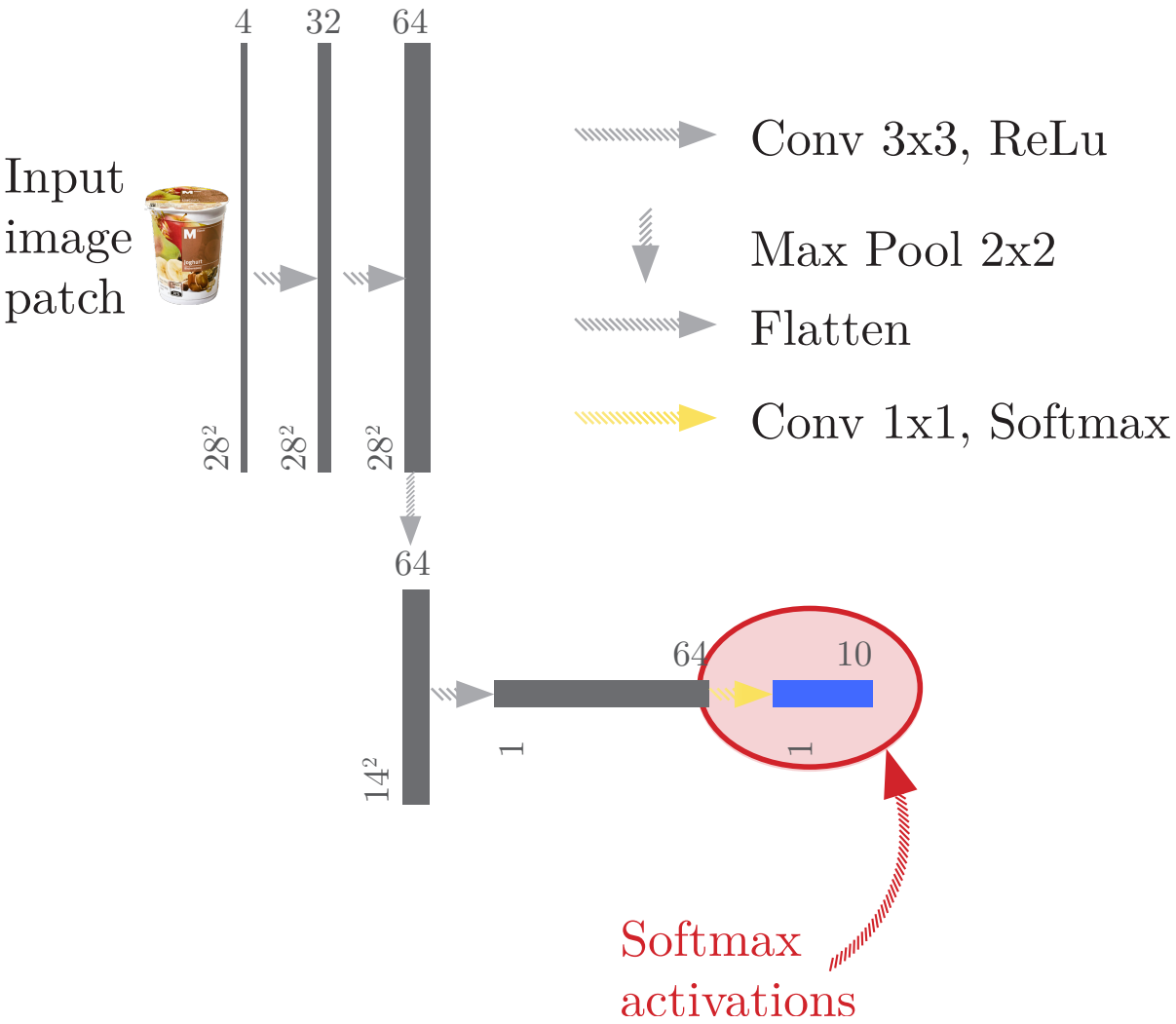
How does Machine Learning work?
Convolutional Neural Networks (CNNs)



How does Machine Learning work?
CNNs



How does Machine Learning work?
CNNs



p( )=	.3
p( )=	.4
p( )=	.3
Sum	1

Research objectives

- Determine uncertainty of standard CNNs using
  - softmax-based methods
  - methods based on pre-softmax activations
- In particular, implement Density Forests and compare them to baseline methods
- Evaluate these uncertainty measures in terms of their performance for **novelty detection**

Types of Uncertainty Measures

1. Based on new network architectures
2. *Using standard network architectures*
   - Based on network output (i.e., softmax activations)
   - Using network activations before the final classification layer

Uncertainty based on Network Output
Problems

Problems with softmax output

- Can be easily fooled

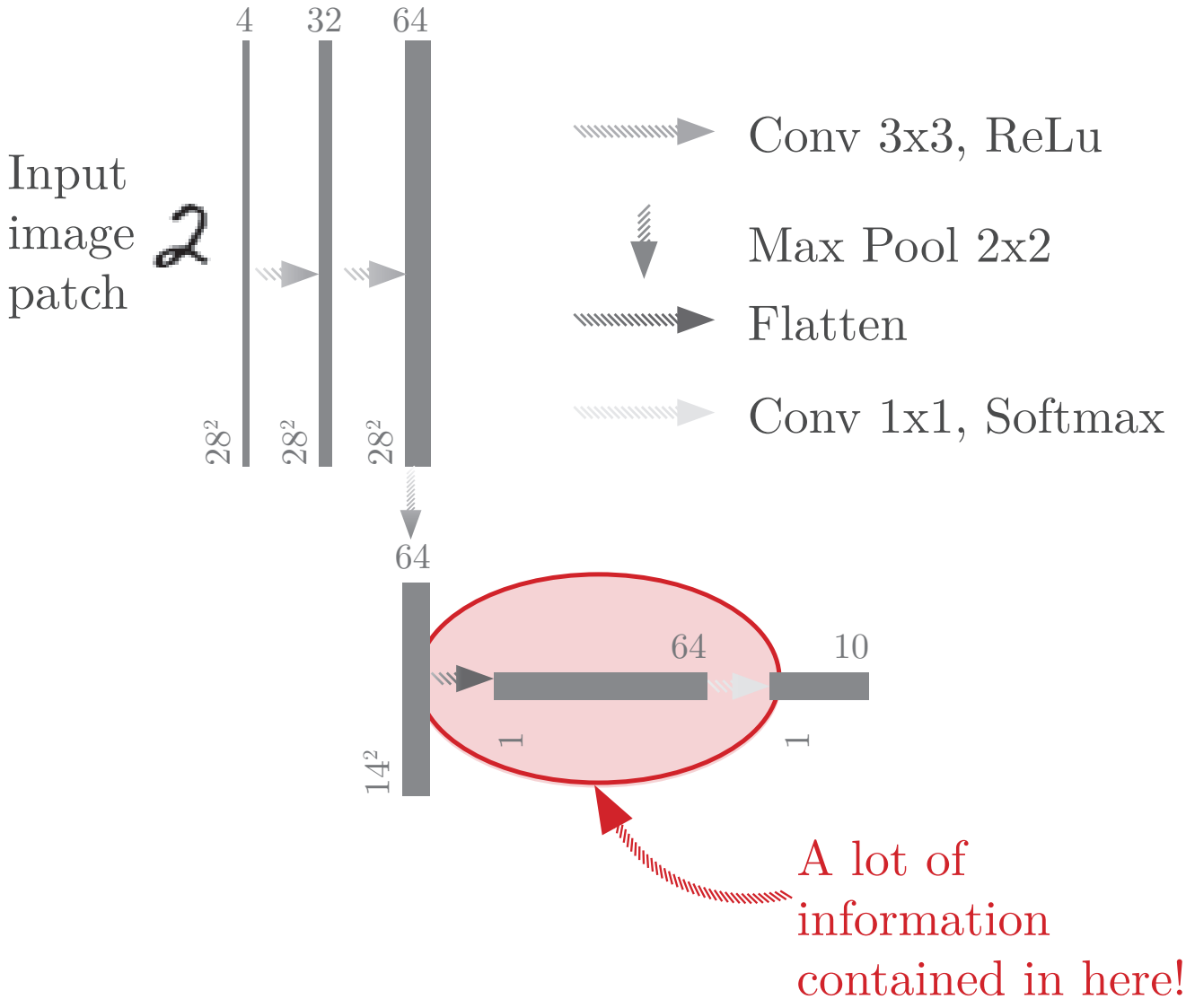
 ⇒ cat  ⇒ dog

- Not robust to minor transformations

 ⇒ cat  ⇒ dog

- Can yield high scores despite being wrong

Uncertainty based on Network Activations



Uncertainty based on Network Output

Based on softmax activations

- Maximum Softmax Response (MSR)
- Margin
- Entropy

Monte-Carlo Dropout (MC-Dropout)

1. Perform prediction *using dropout*
2. Repeat prediction n times
3. Prediction = mean of outputs, Certainty = variance of outputs

$\Rightarrow$ Simplified version used:

- Dropout only in last layer before softmax
- Using entropy of mean output rather than variance

Novelty Detection Methods

Goal: find samples belonging to novel, unseen classes

- Model distribution or support of "normal class"
- Attribute low confidence to samples of "abnormal class"

$\Rightarrow$ Binary classification task!

Gaussian Mixture Models (GMMs)

- ~~Fit n Gaussians to training data using Expectation Maximization (EM)~~
- ~~Predict log-likelihood of test data given the fitted distributions~~

One-Class Support Vector Machines (OC-SVMs)

- ~~Find support of the "normal data" class~~
- ~~Use decision function to decide whether a data point is an inlier or outlier~~

Random Forests
Idea

Why should we need to model the "normal class" perfectly if we can do simpler?

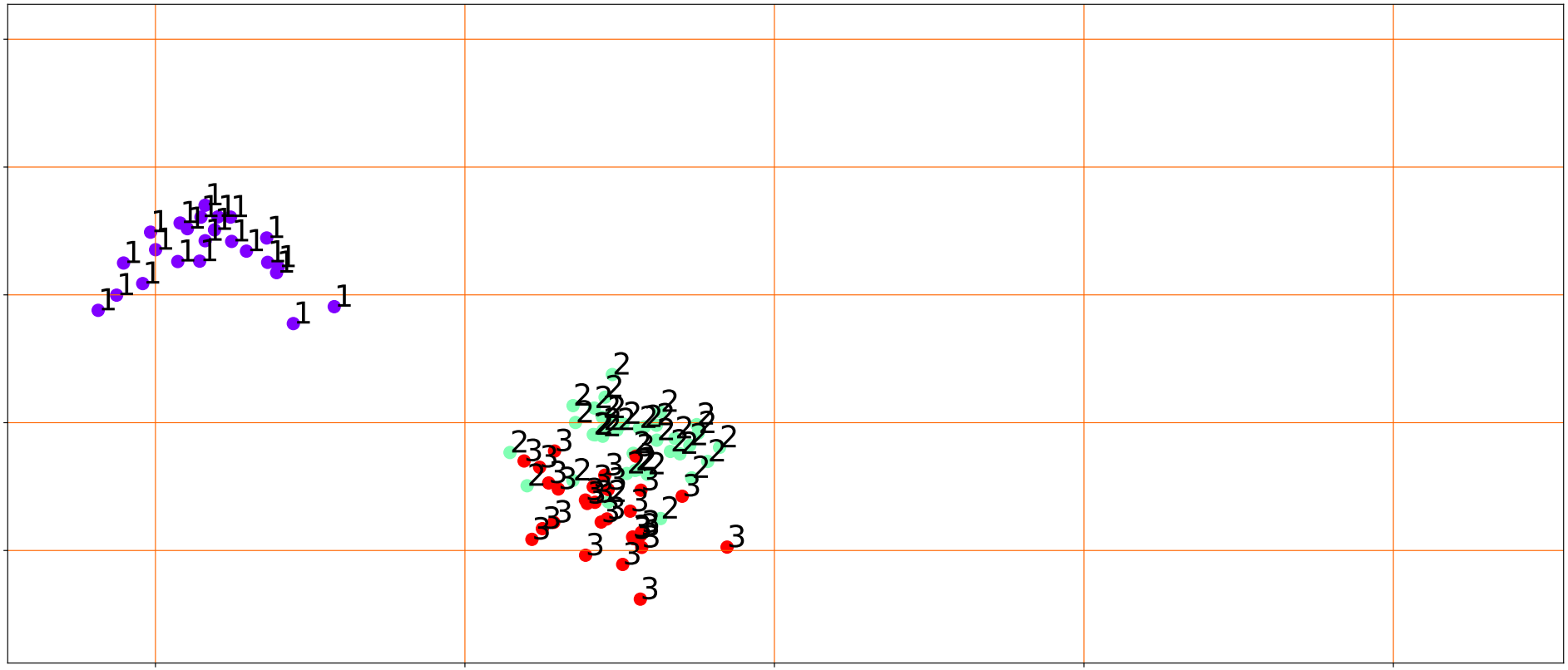
- Train many imperfect models (in parallel)
- Average them

Random Forests

A yoghurt example

Random Forests

A yoghurt example



Random Forests

A yoghurt example

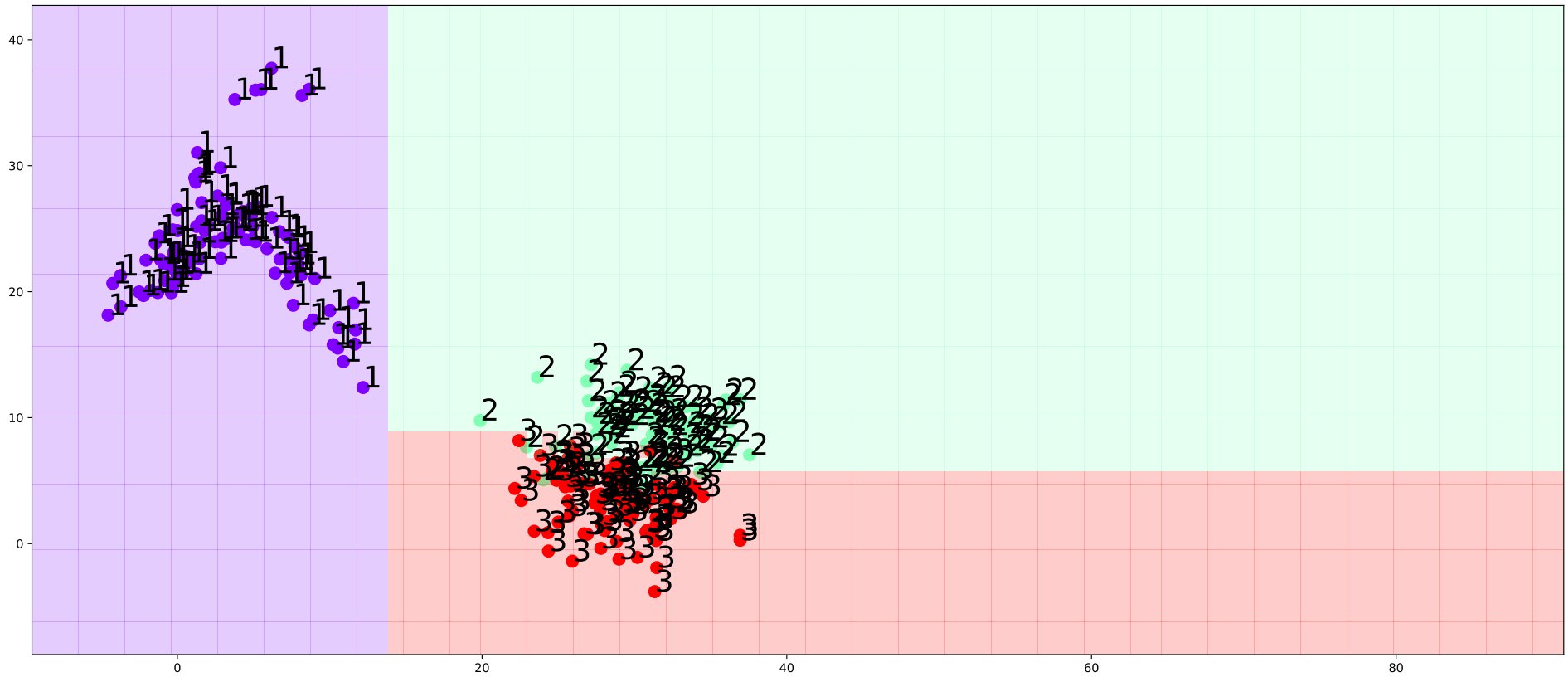


Figure: Single Decision Tree

Random Forests
A yoghurt example

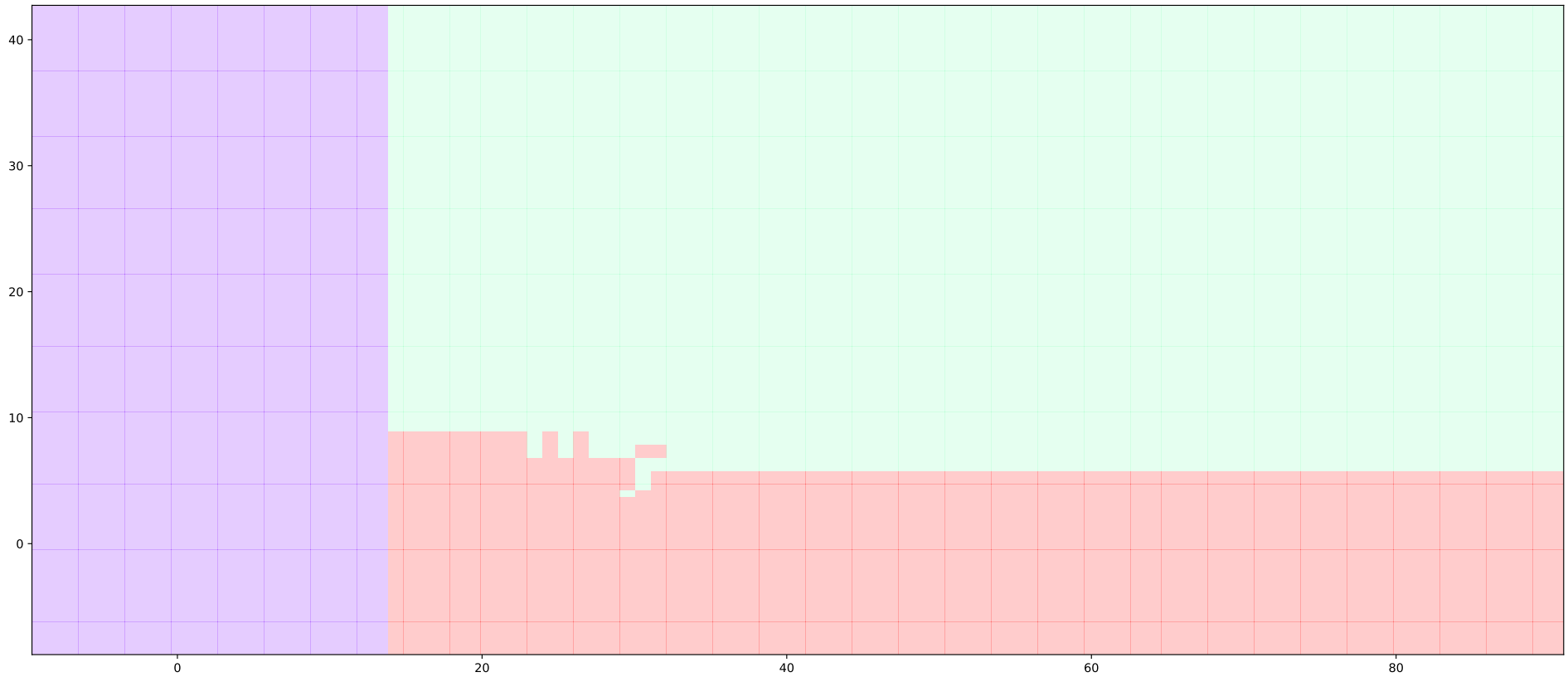


Figure: Single Decision Tree

Random Forests

A yoghurt example

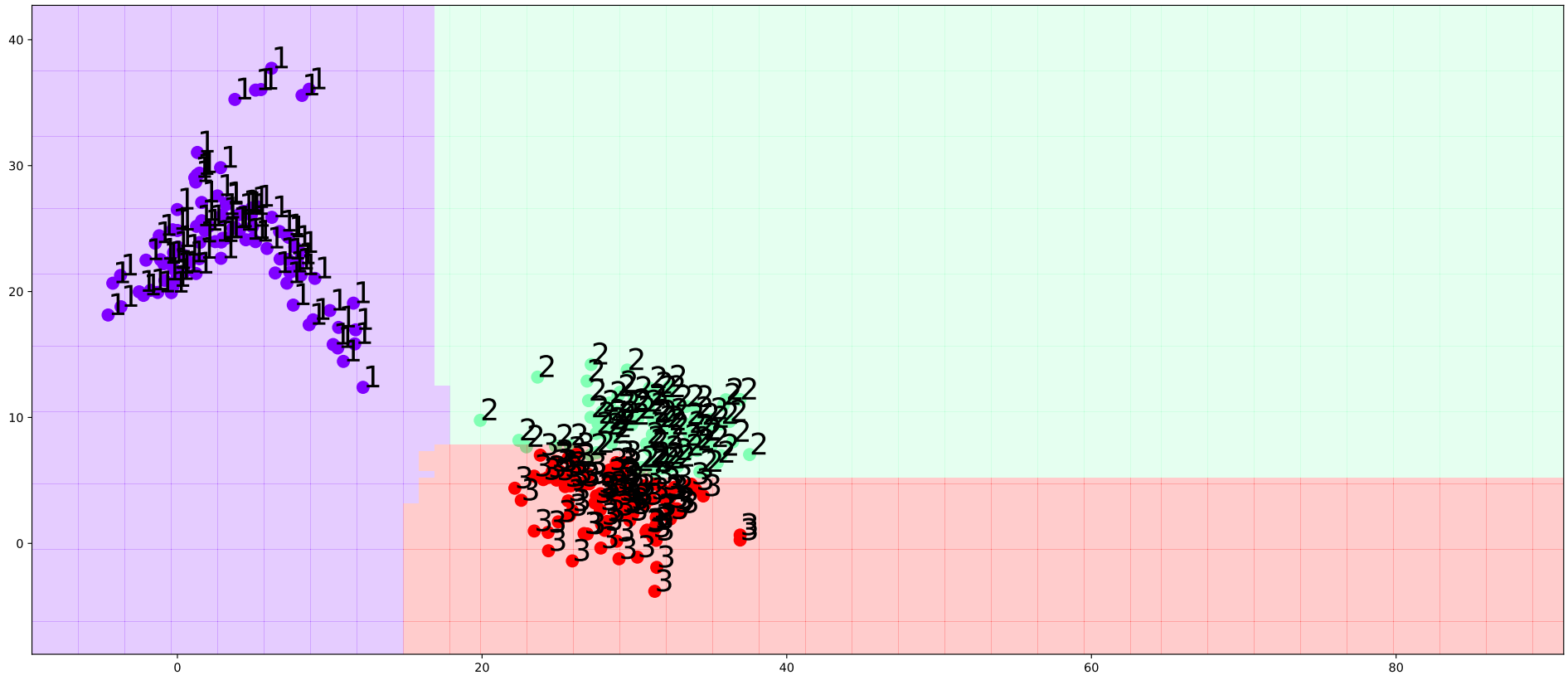


Figure: Random Forest

Random Forests

A yoghurt example

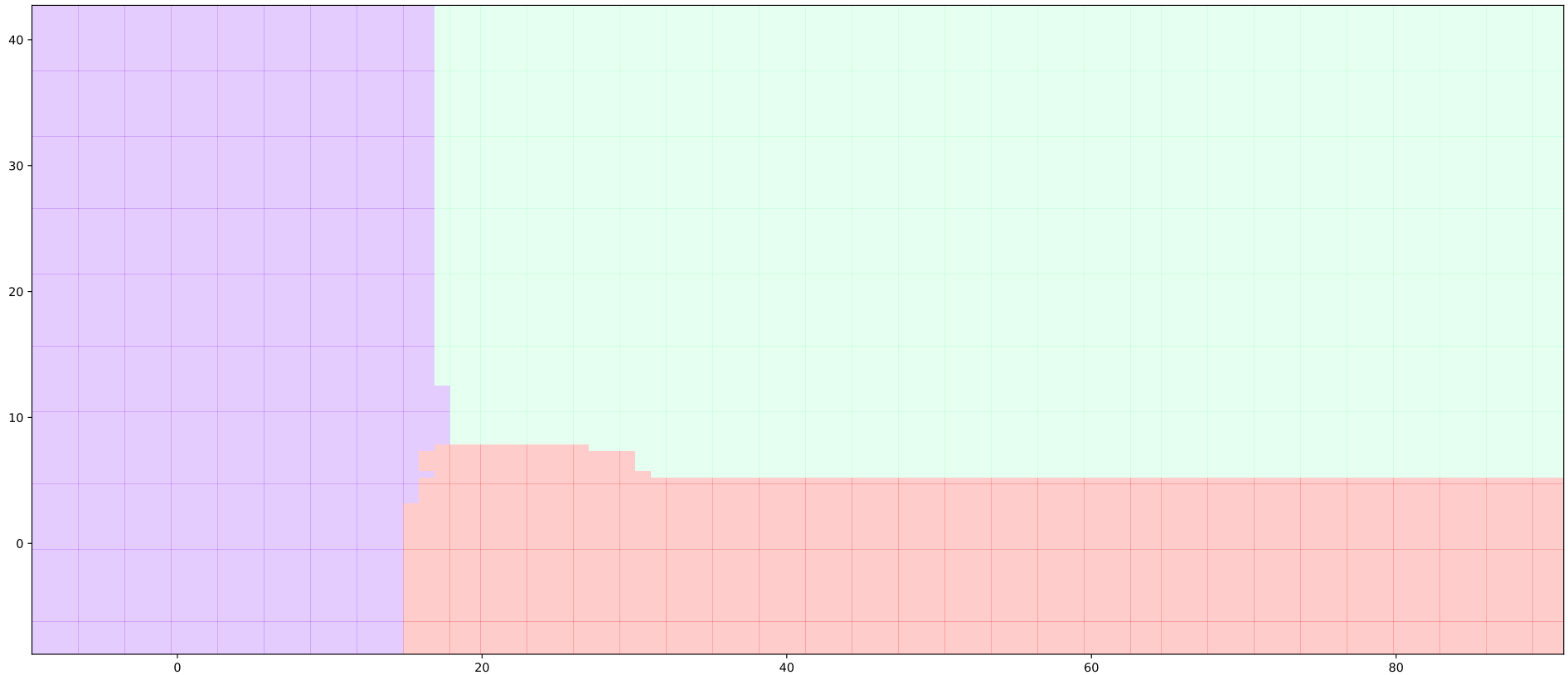
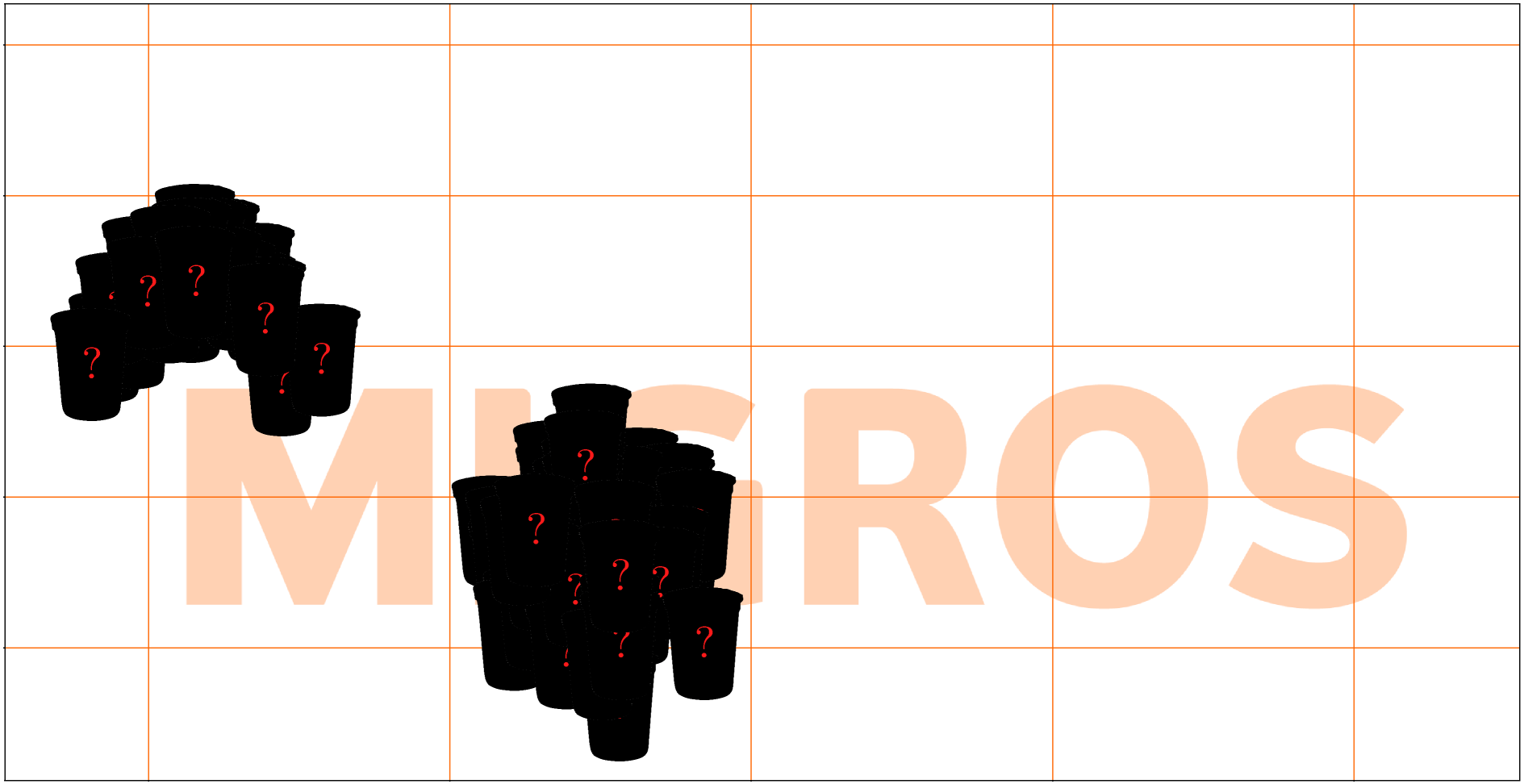


Figure: Random Forest

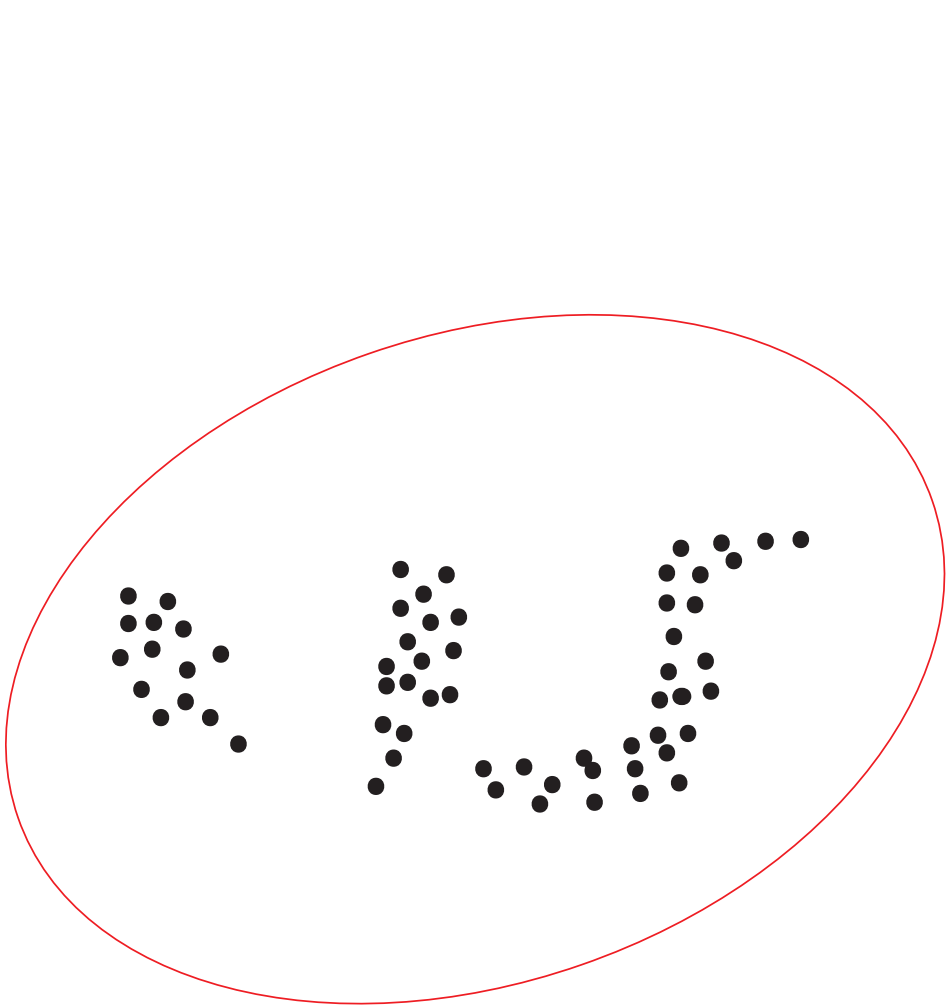
Density Forests
An non-yoghurt example

Question: How to define subspaces of unlabelled data?



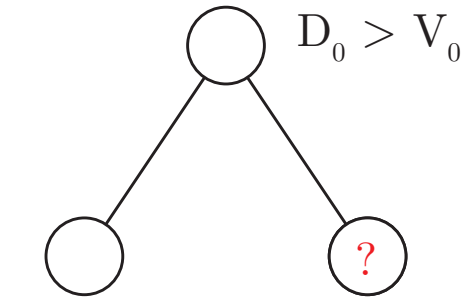
Density Forests

An non-yoghurt example

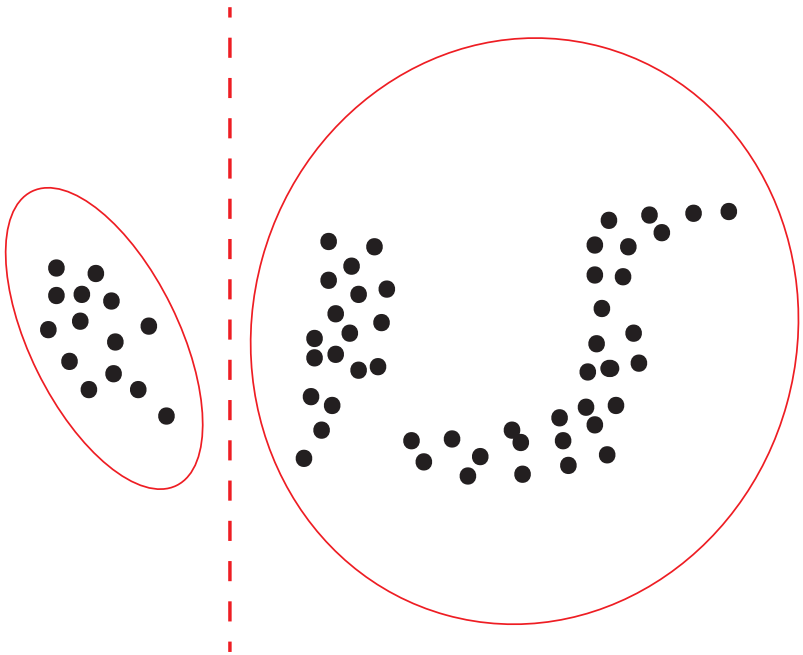


Density Forests

An non-yoghurt example

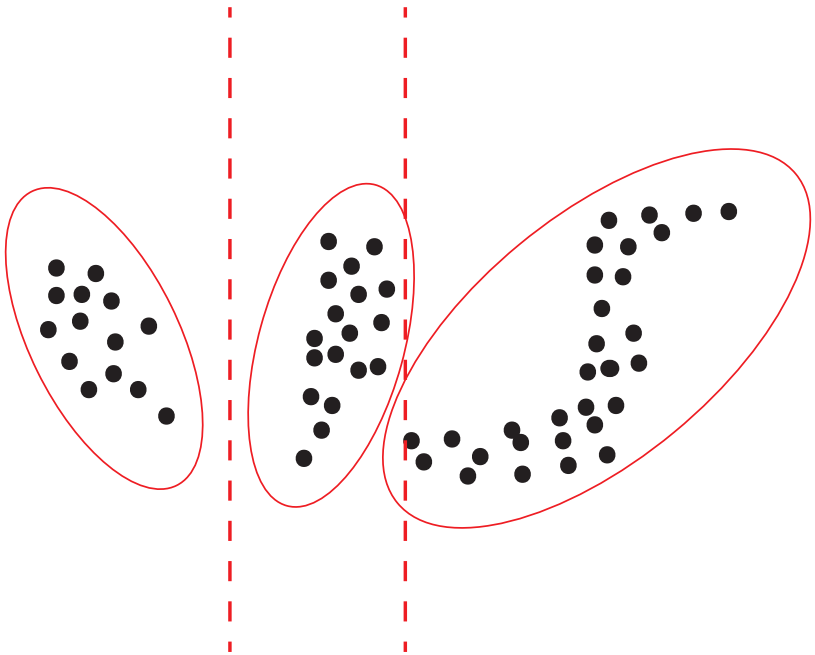


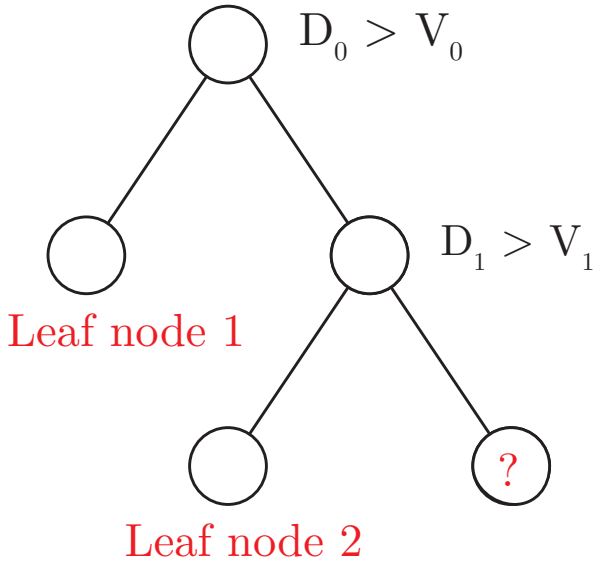
Leaf node 1



Density Forests

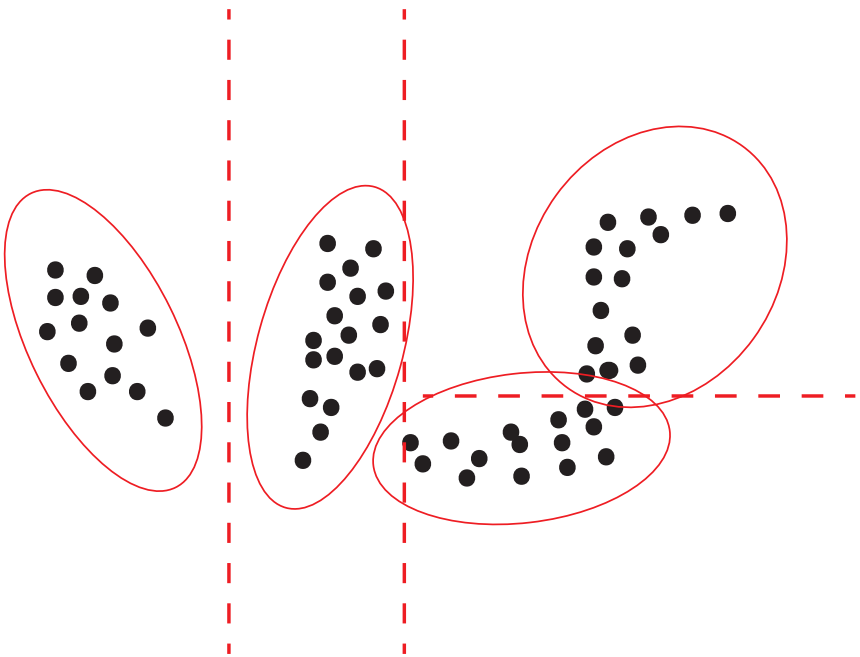
An non-yoghurt example

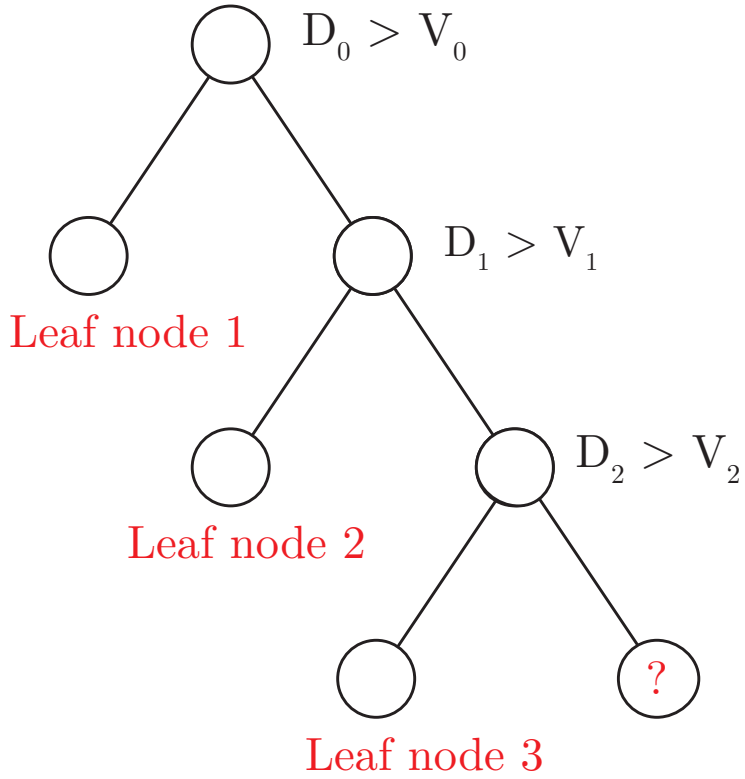




Density Forests

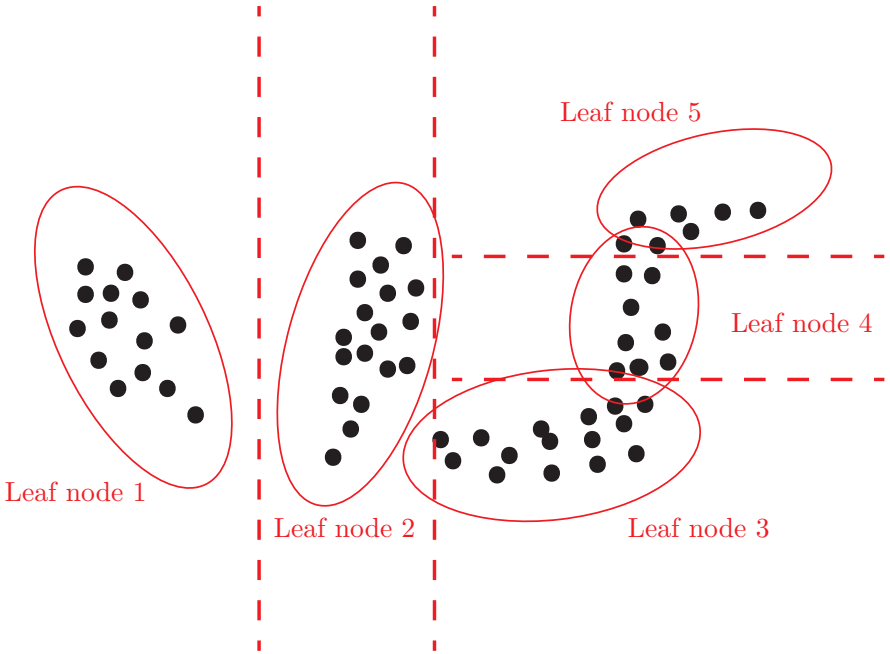
An non-yoghurt example

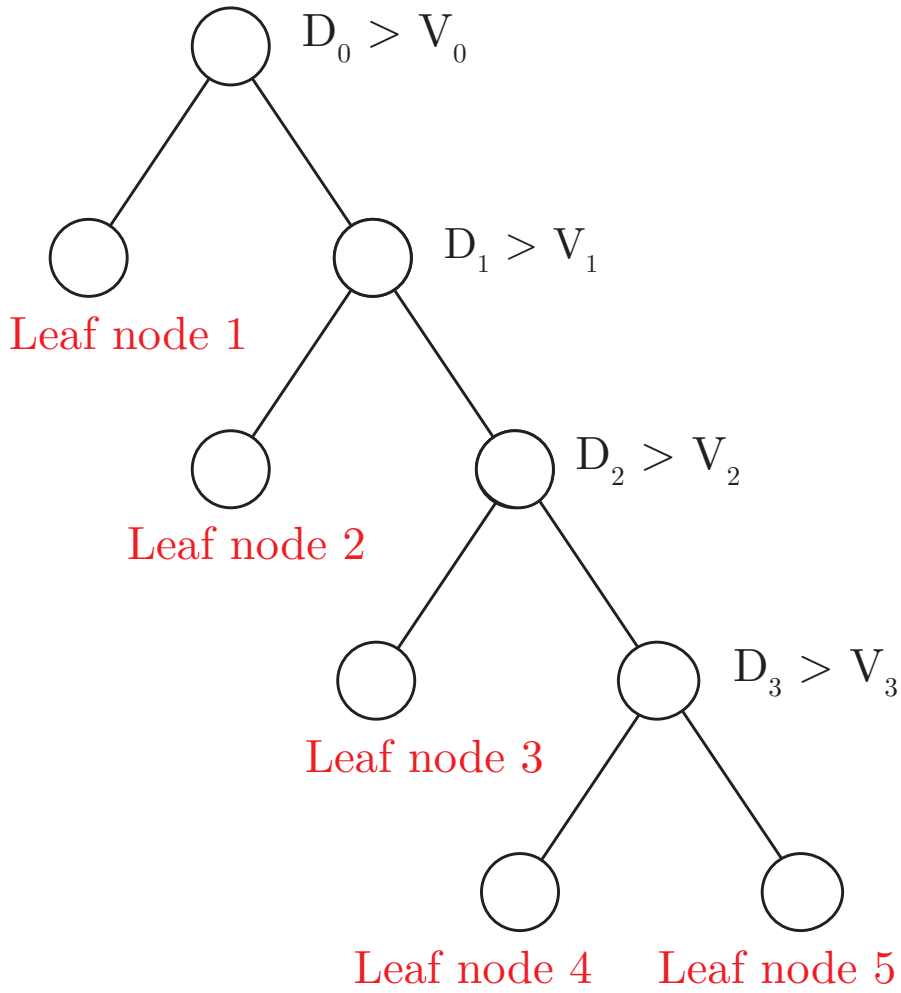




Density Forests

An non-yoghurt example





Density Forests

An non-yoghurt example

- Finding best split by maximizing **Gaussianity** at each side
- **Bagging** (**B**ootstrap **Agg**regat**ing**) of multiple weak learners

$\Rightarrow$ Requires tuning **hyperparameters** to avoid under- and overfitting:

- Number of trees
- Tree depth
- Number of dimensions to consider for splitting

Datasets
Synthetic Datasets

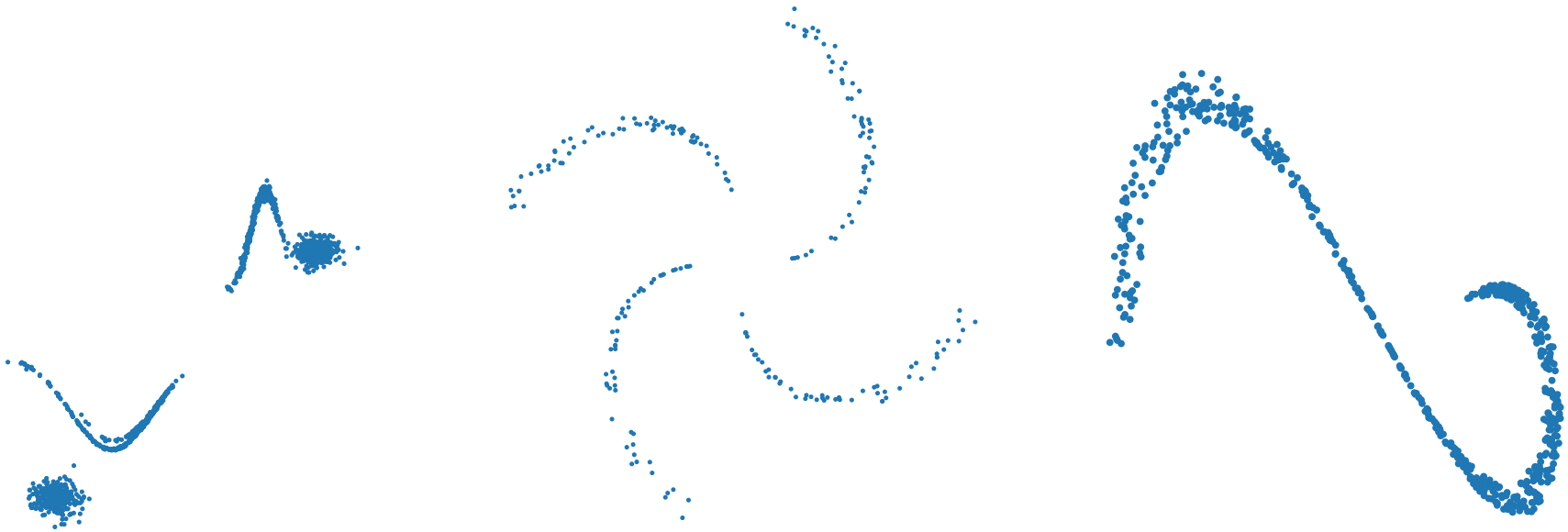
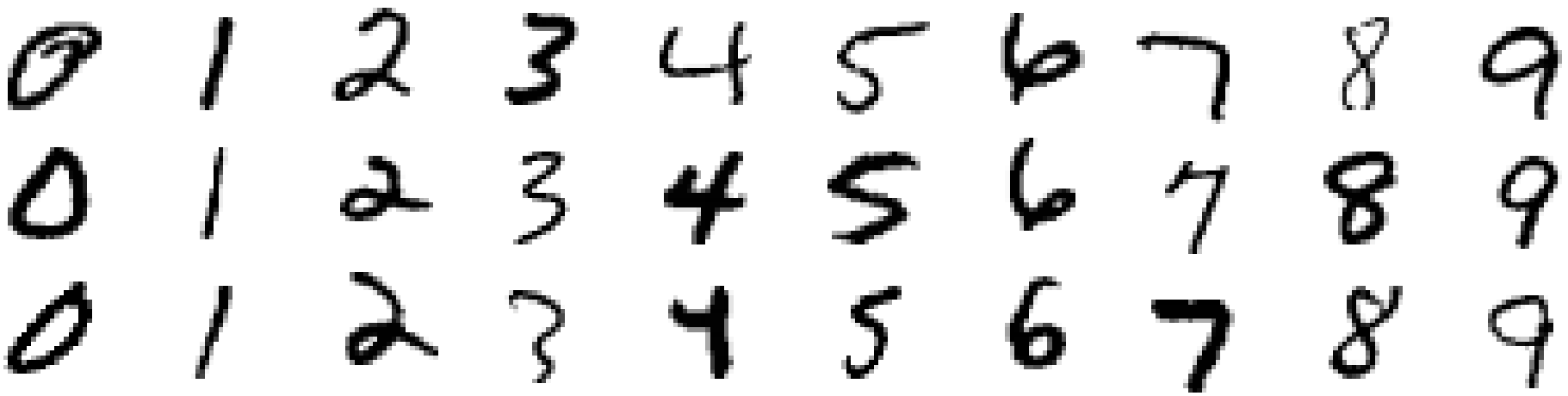


Figure: Synthetic datasets

Just used to illustrate Density Forests.

Datasets

Modified National Institute of Standards and Technology (MNIST) [1]

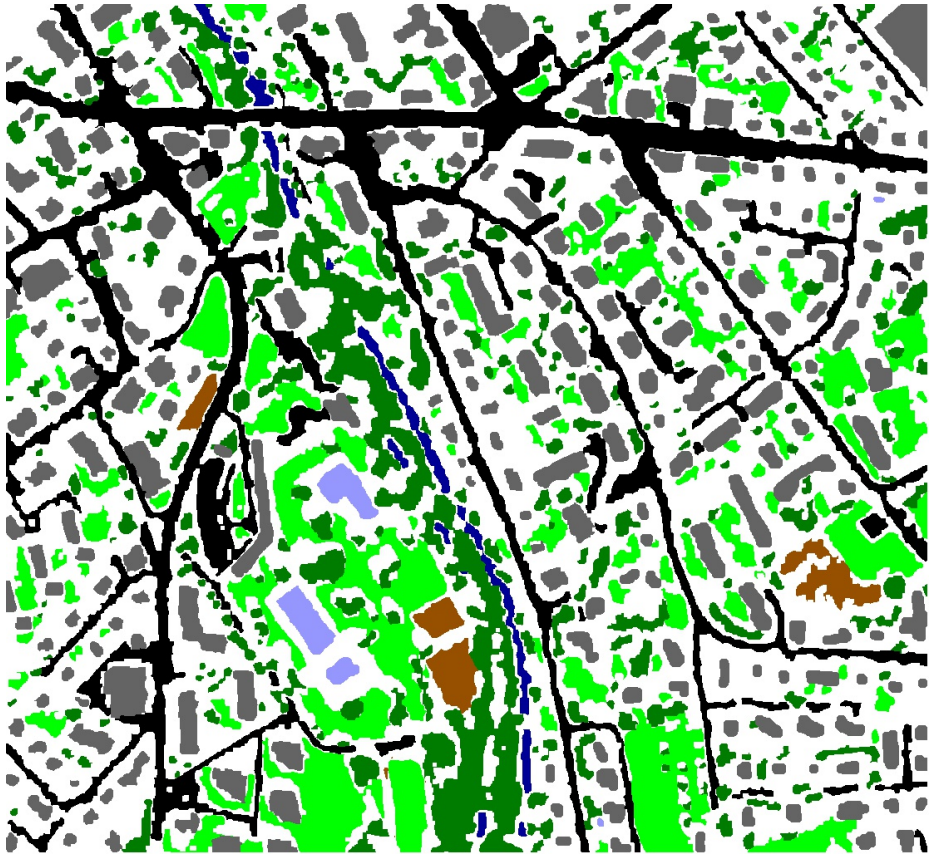


60'000 training images, 10'000 test images, classes balanced

Datasets
Zurich Dataset [2]

~~RGB Image~~

Ground Truth



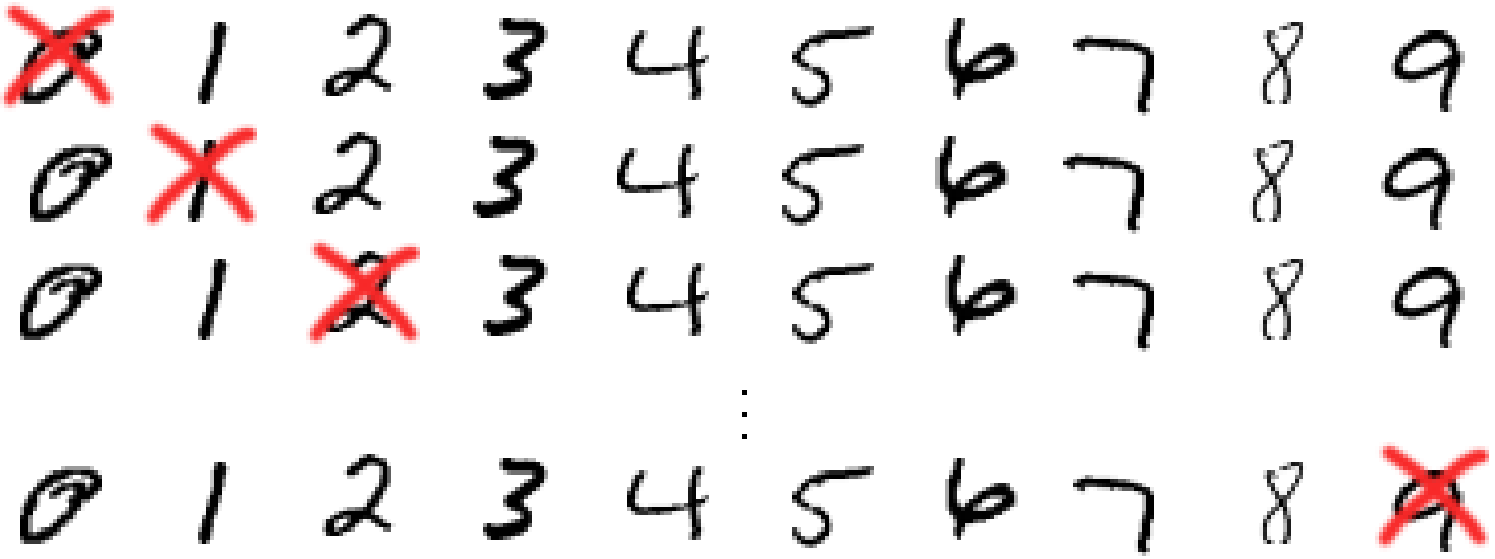
Background Roads Buildings Trees Grass

Bare Soil Water Railways Swimming Pools

Experimental Setup
CNN

CNNs

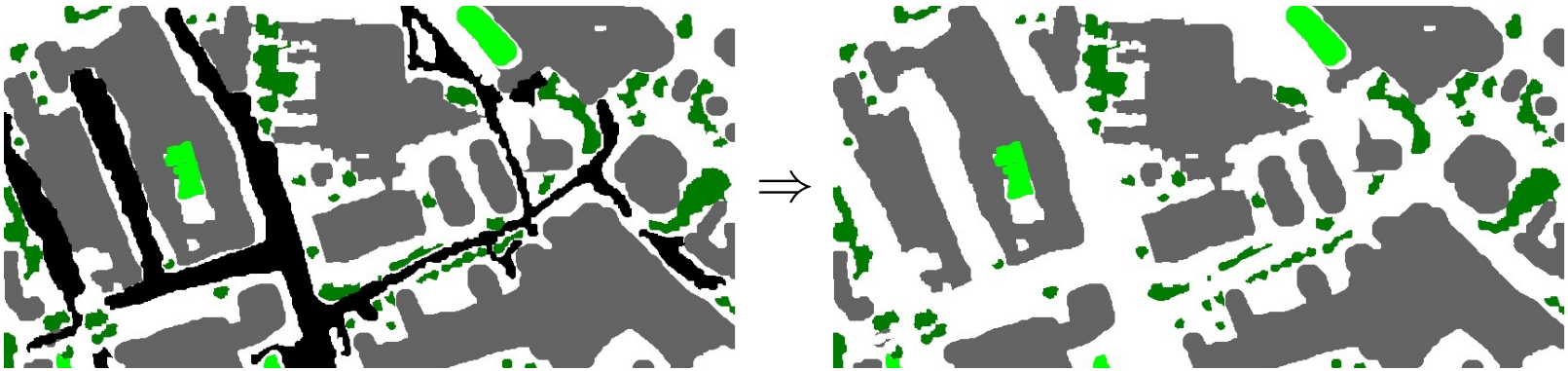
- Standard network architectures
- Leaving one class out during training



Experimental Setup
CNN

CNNs

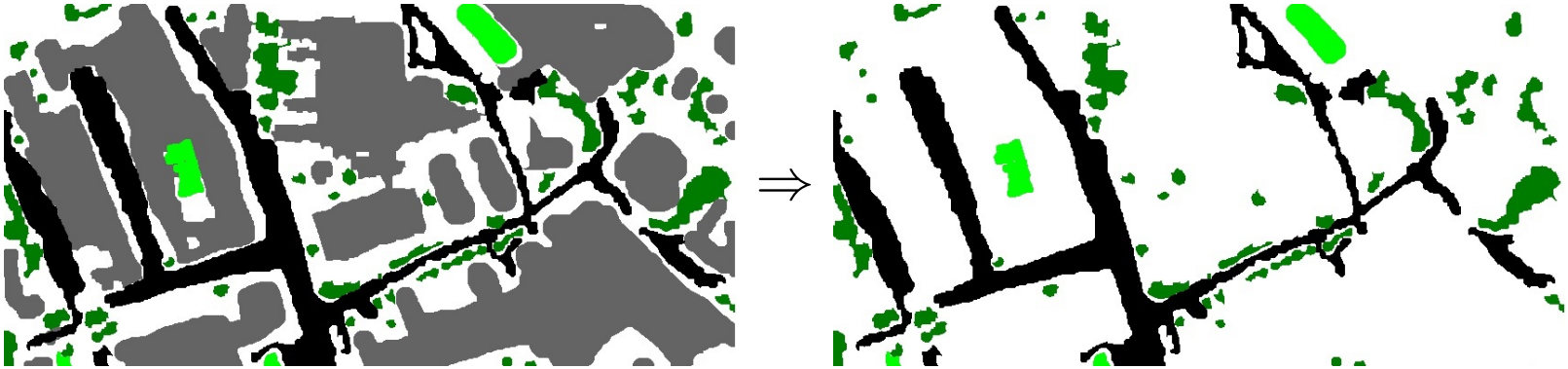
- Standard network architectures
- Leaving one class out during training



Experimental Setup
CNN

CNNs

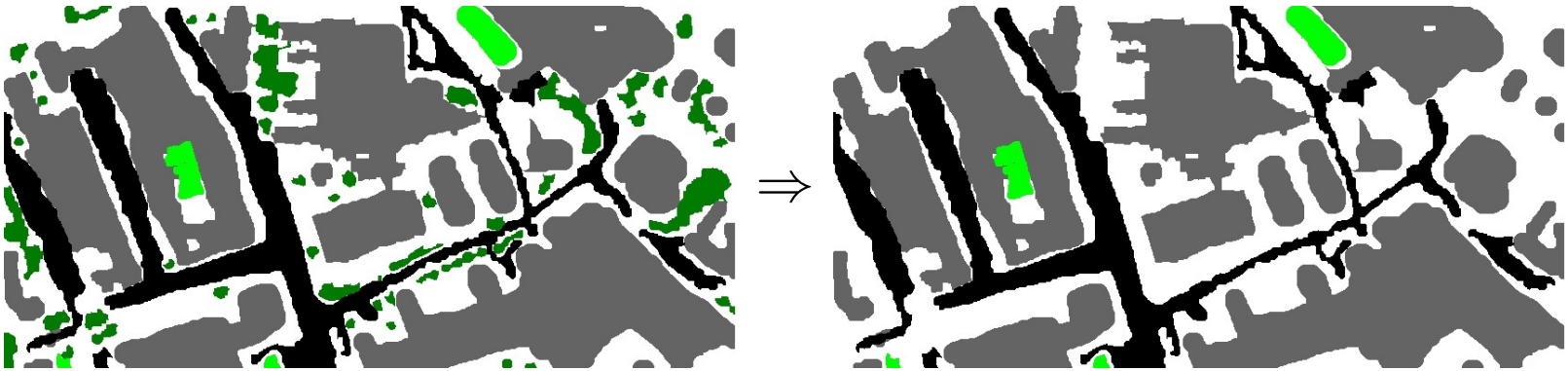
- Standard network architectures
- Leaving one class out during training



Experimental Setup
CNN

CNNs

- Standard network architectures
- Leaving one class out during training



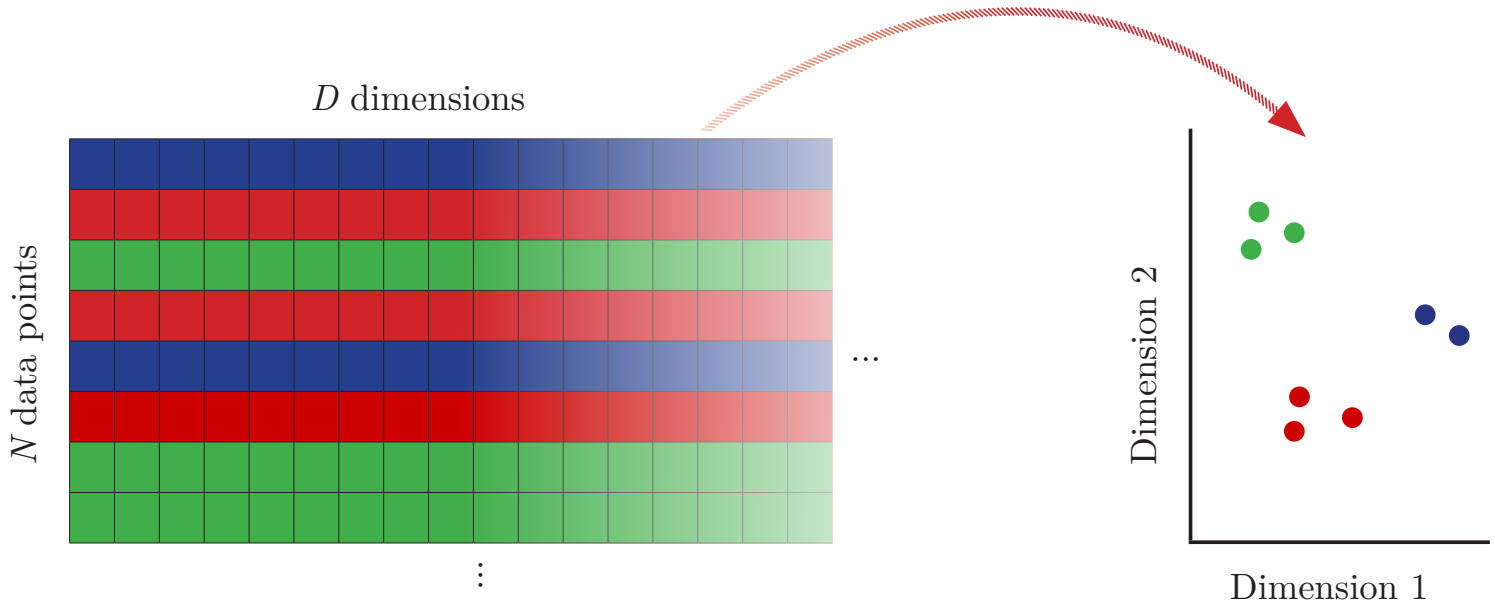
Experimental Setup
CNN

Novelty Detection methods GMM, OC-SVM, Density Forest (DF)

- Model activations of the *seen classes*
- Predicting confidence for entire test set, including the *unseen class*

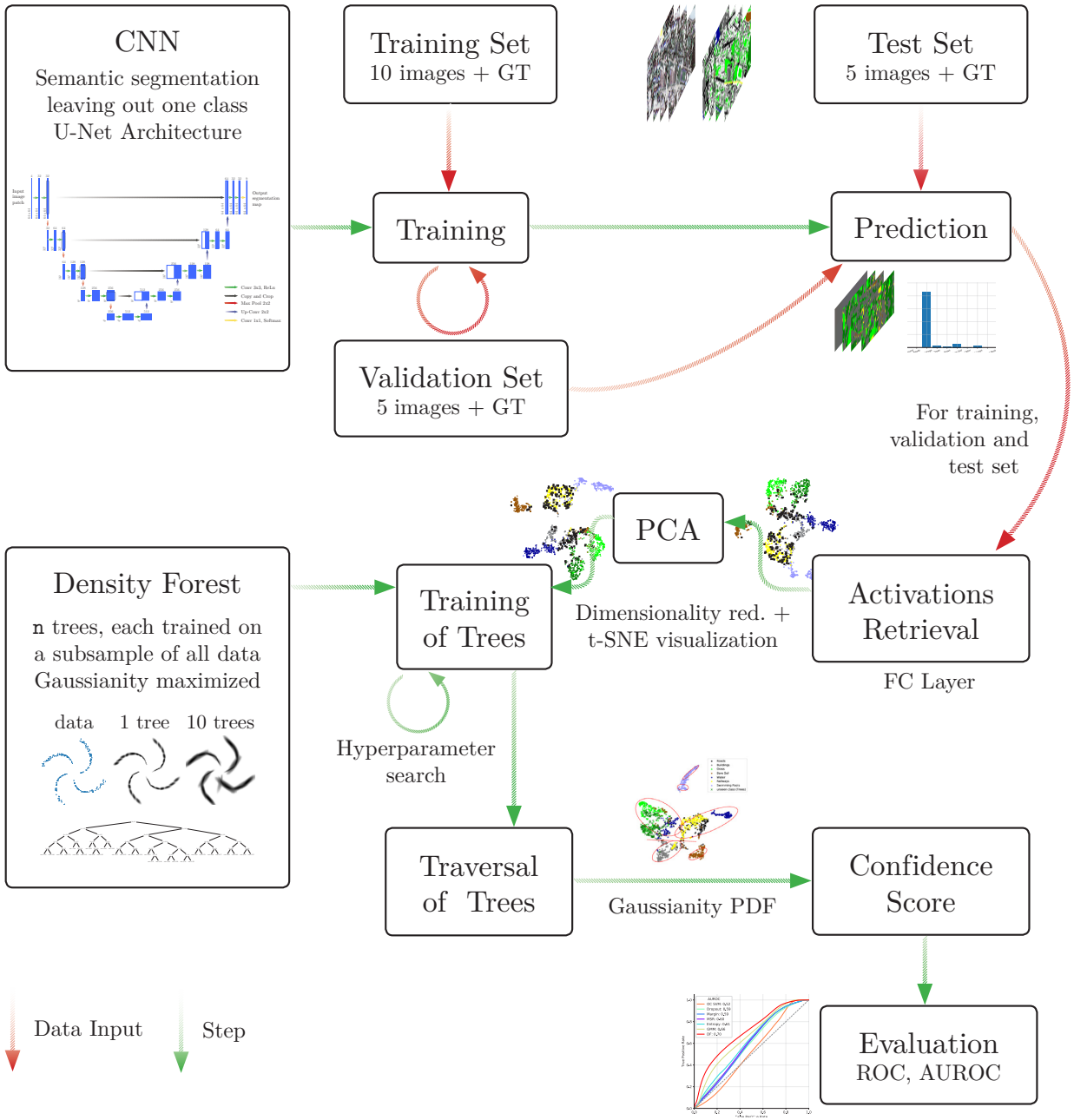
Experimental Setup
Dimensionality Reduction

- Standard CNN for MNIST yields 128 activations
  - Redundancy, collinearity
  - High-dimensional data difficult to handle
- Principal Component Analysis (PCA): preserve data variance
- Visualization: t-distributed Stochastic Neighbor Embedding (t-SNE)



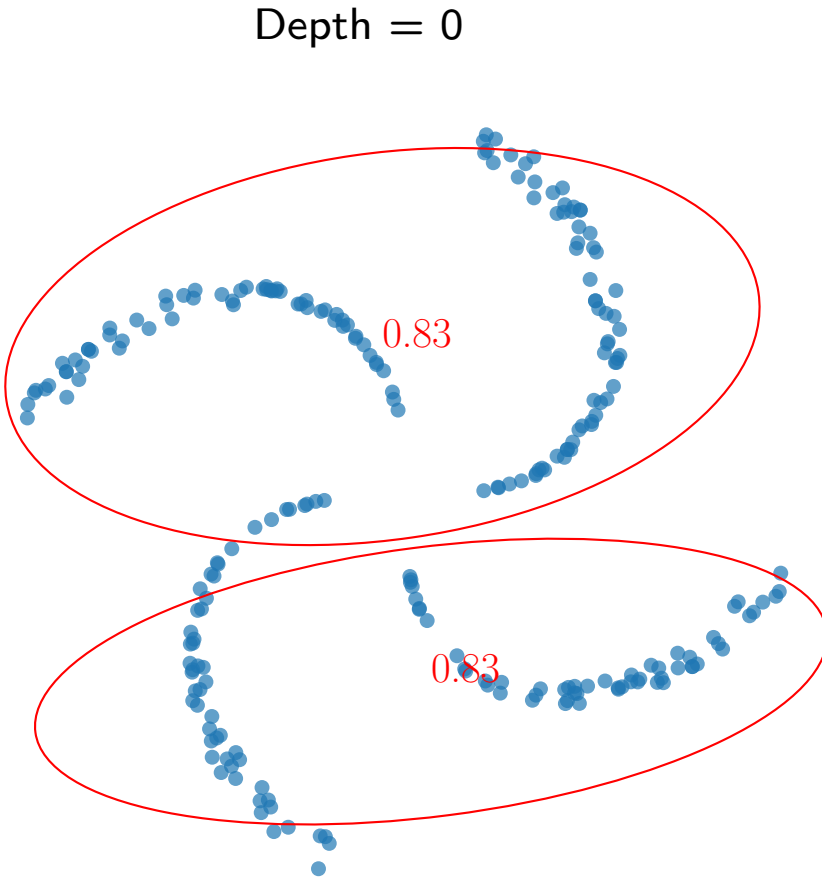
Evaluation

- **U-Net CNN** models with left-out class: Overall Accuracy (OA), Average Accuracy (AA)
- **Novelty Detection**:
  - Area Under the curve of the Receiver Operating Characteristic (AUROC)
  - Visual quality of results

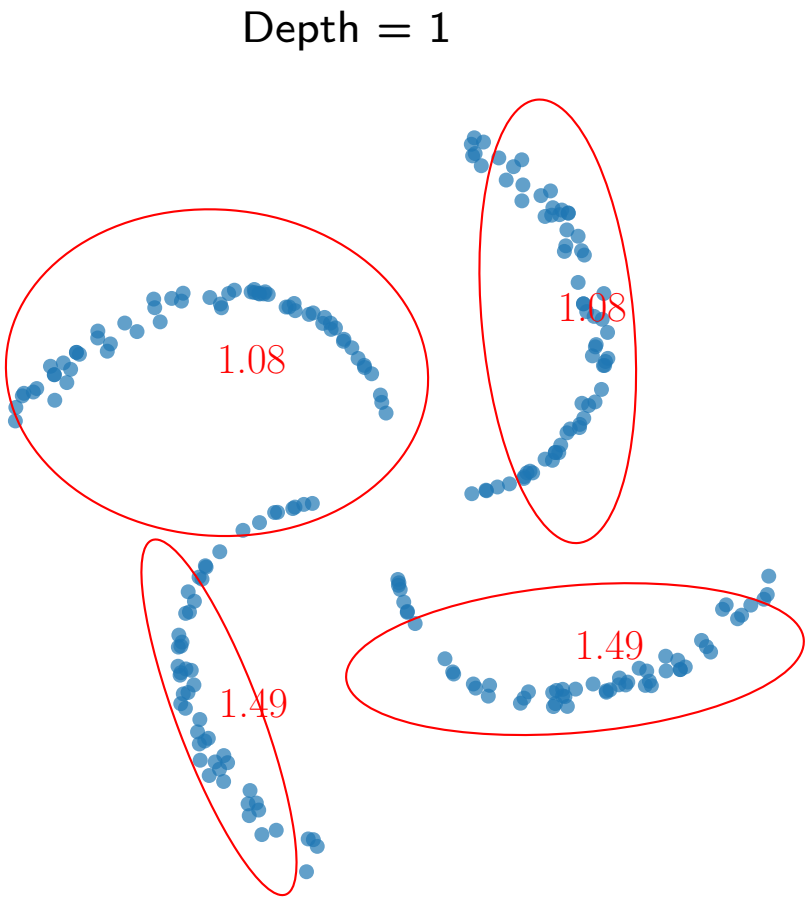

CNN
Semantic segmentation
leaving out one class
U-Net Architecture
Training Set
10 images + GT
Test Set
5 images + GT
Training
Prediction
Validation Set
5 images + GT
For training,
validation and
test set
PCA
Density Forest
n trees, each trained on
a subsample of all data
Gaussianity maximized
data
1 tree
10 trees
Training
of Trees
Dimensionality red. +
t-SNE visualization
Activations
Retrieval
FC Layer
Hyperparameter
search
Traversal
of Trees
Gaussianity PDF
Confidence
Score
Evaluation
ROC, AUROC
Data Input
Step

Results

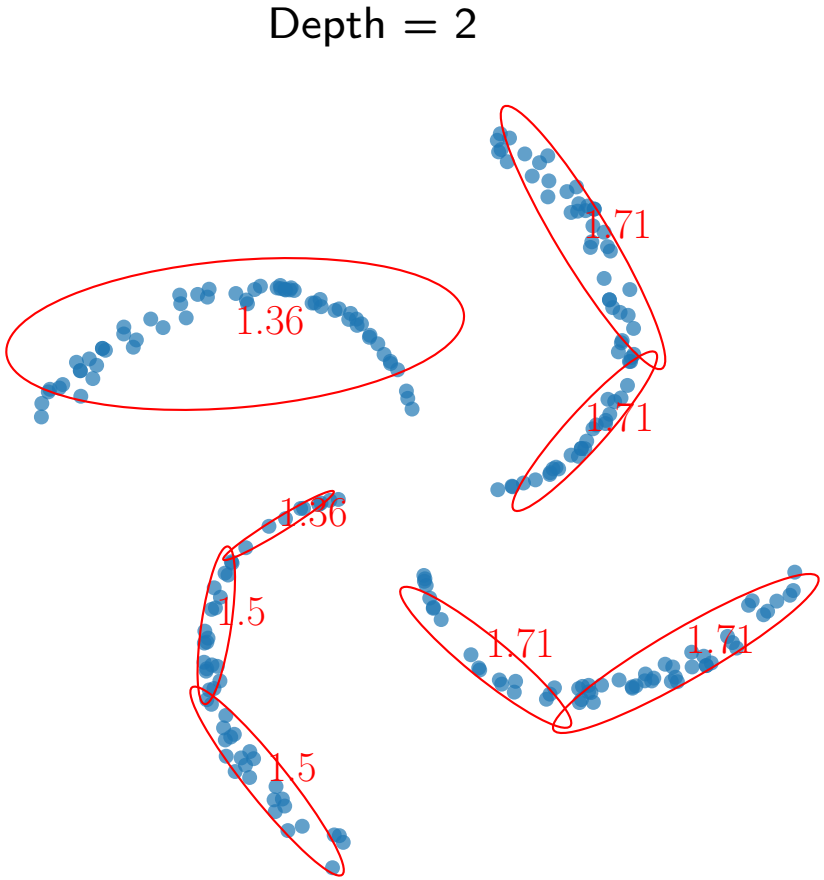
Dummy Dataset: One Tree



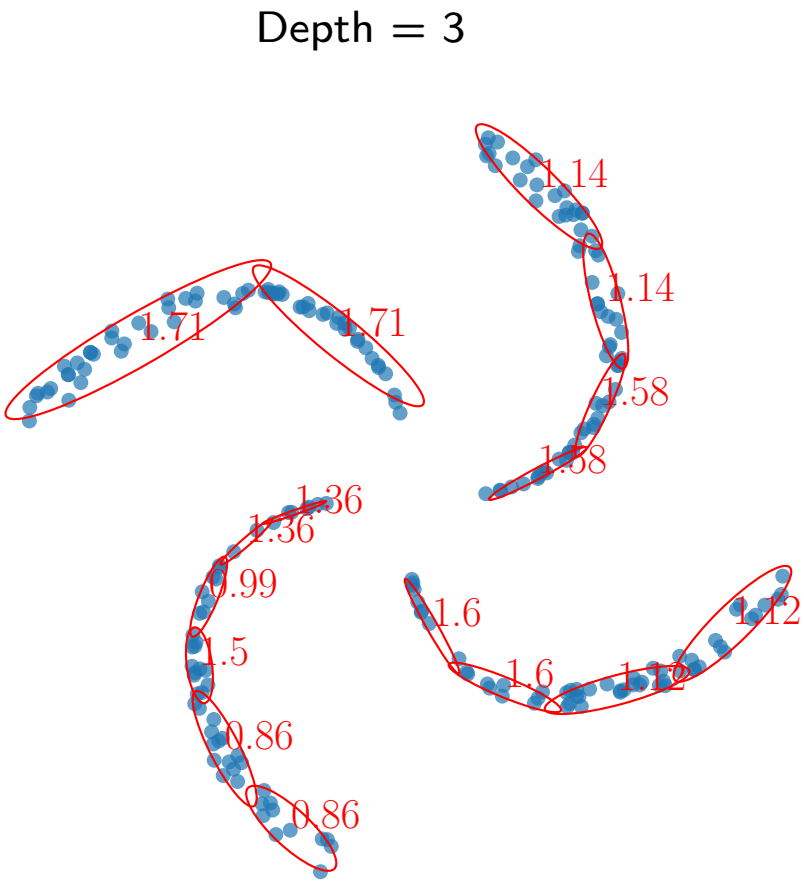
Dummy Dataset: One Tree



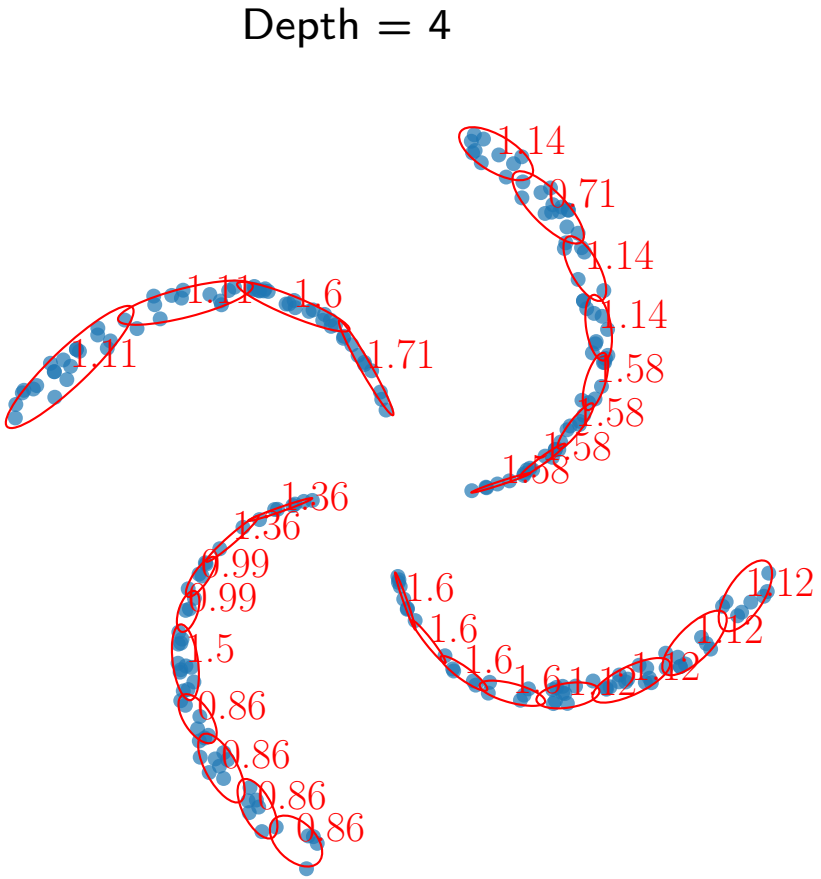
Dummy Dataset: One Tree



Dummy Dataset: One Tree



Dummy Dataset: One Tree



Dummy Dataset: One Tree

Gaussian Probability Density Function (PDF)

Dummy Datasets: Several Trees

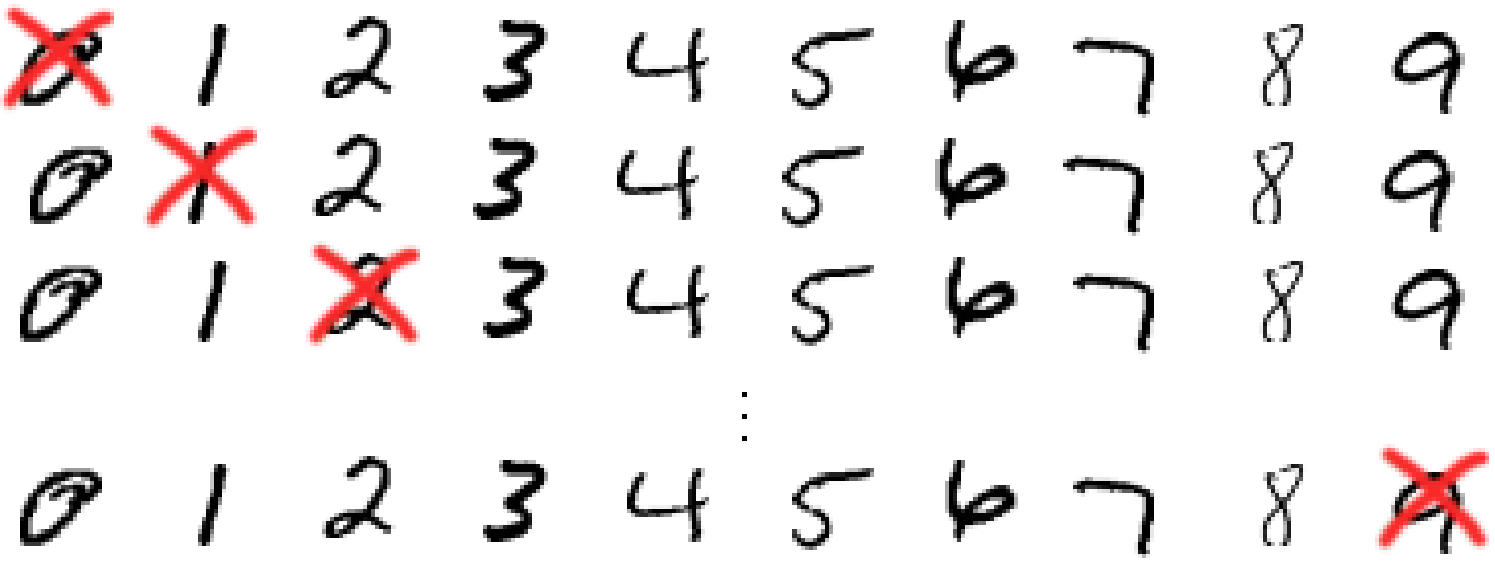
Dummy Datasets: Several Trees

MNIST Dataset
CNNs

CNNs trained leaving out one class

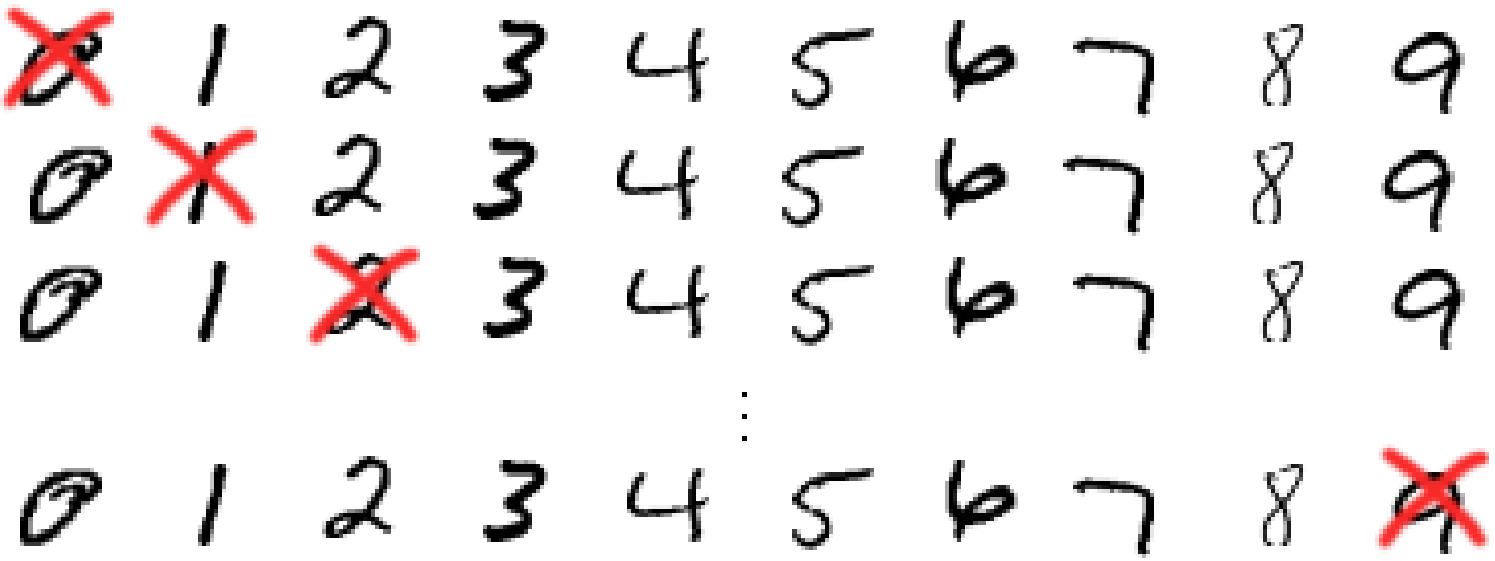
MNIST Dataset
CNNs

CNNs trained leaving out one class



MNIST Dataset
CNNs

CNNs trained leaving out one class

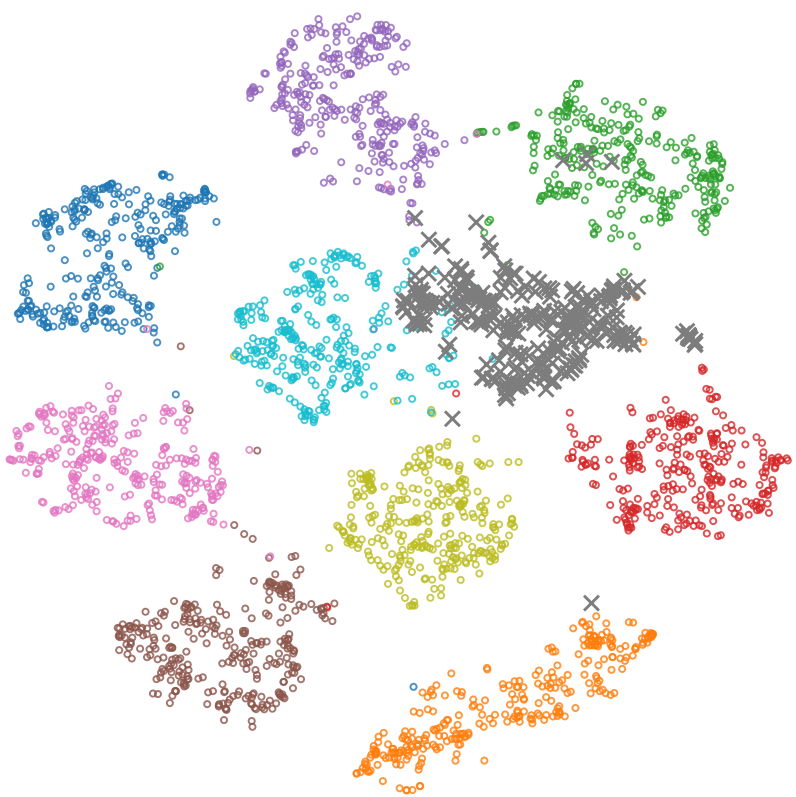


Accuracy:

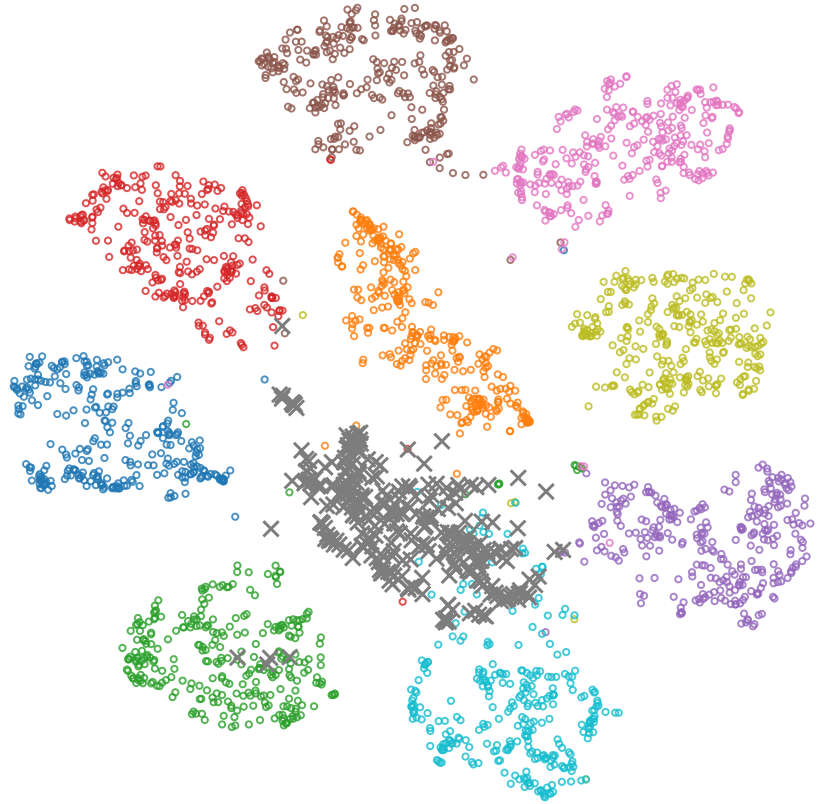
Training set	Test set
99.34	99.03

High accuracy, but...

MNIST
t-SNE



t-SNE before PCA



t-SNE after PCA

Class: 0 1 2 3 4 5 6 7 8 9

× Unseen class (class 7)

MNIST
Novelty Detection

MSR	Margin	Entropy	MC-Dropout	GMM	OC-SVM	DF
0.97	**0.97**	**0.97**	0.96	0.67	0.75	0.75

Table: Mean AUROC for each left-out class in the MNIST dataset

MNIST
Novelty Detection

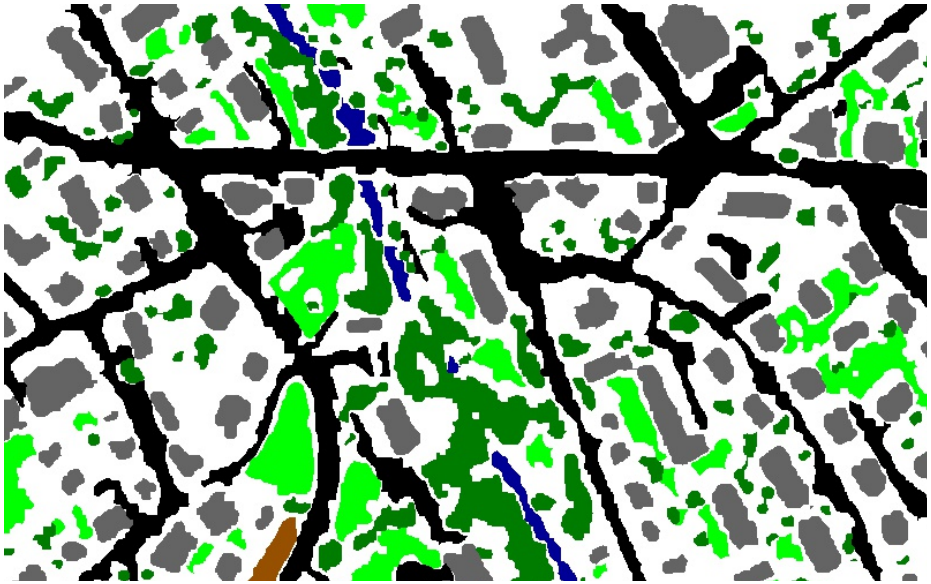
MSR	Margin	Entropy	MC-Dropout	GMM	OC-SVM	DF
0.97	**0.97**	**0.97**	0.96	0.67	0.75	0.75

Table: Mean AUROC for each left-out class in the MNIST dataset

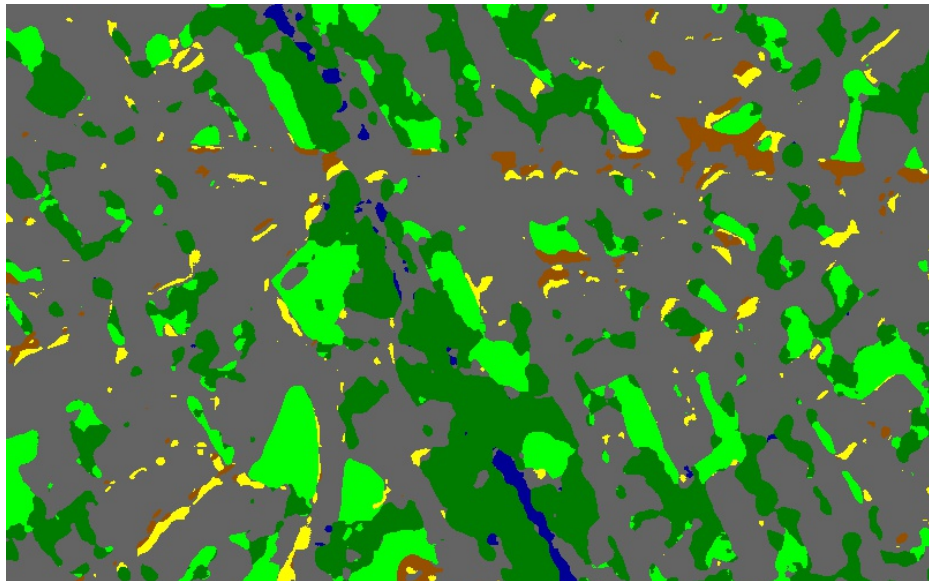
:/
We'll come back to this...

Zurich Dataset
CNNs

Ground Truth



Prediction (wo. Roads)

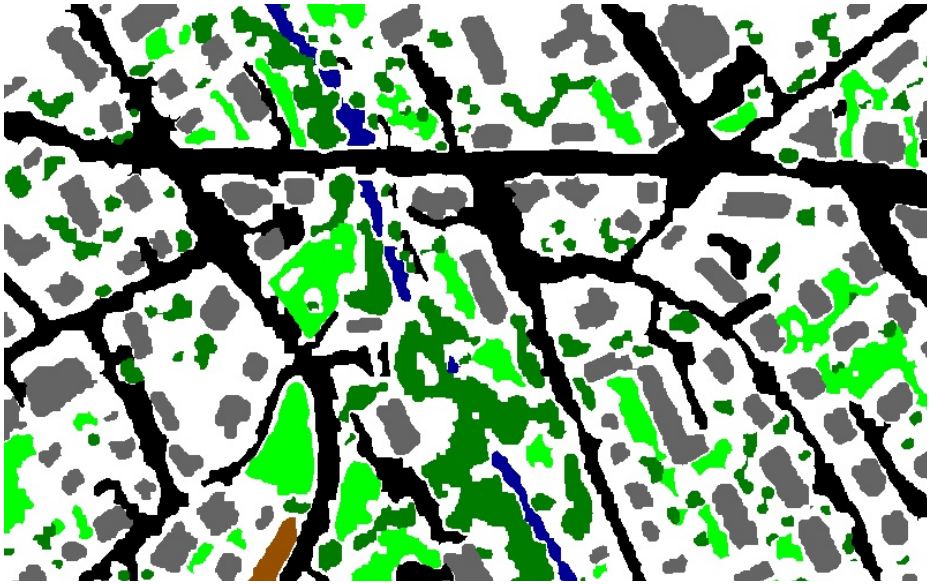


Background Roads Buildings Trees Grass

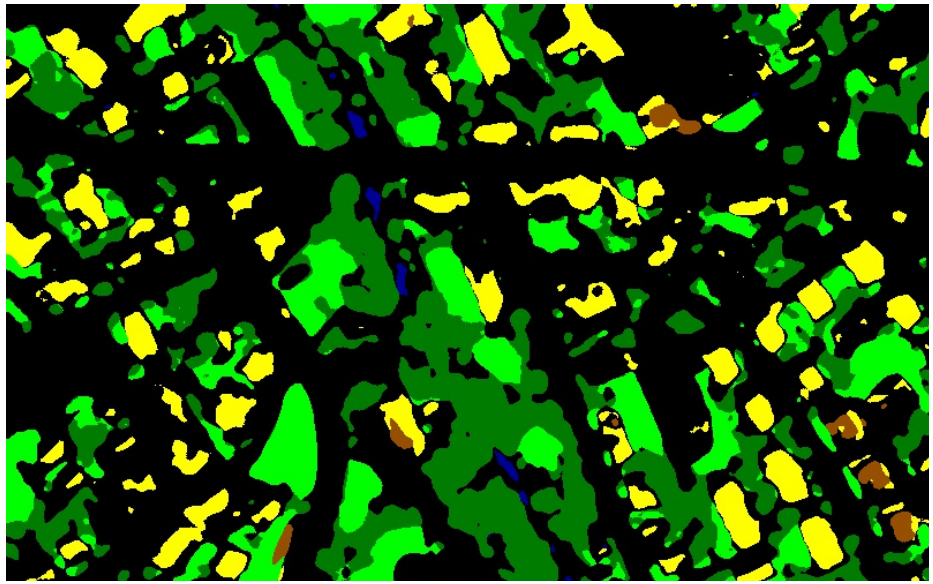
Bare Soil Water Railways Swimming Pools

Zurich Dataset
CNNs

Ground Truth



Prediction (wo. Buildings)

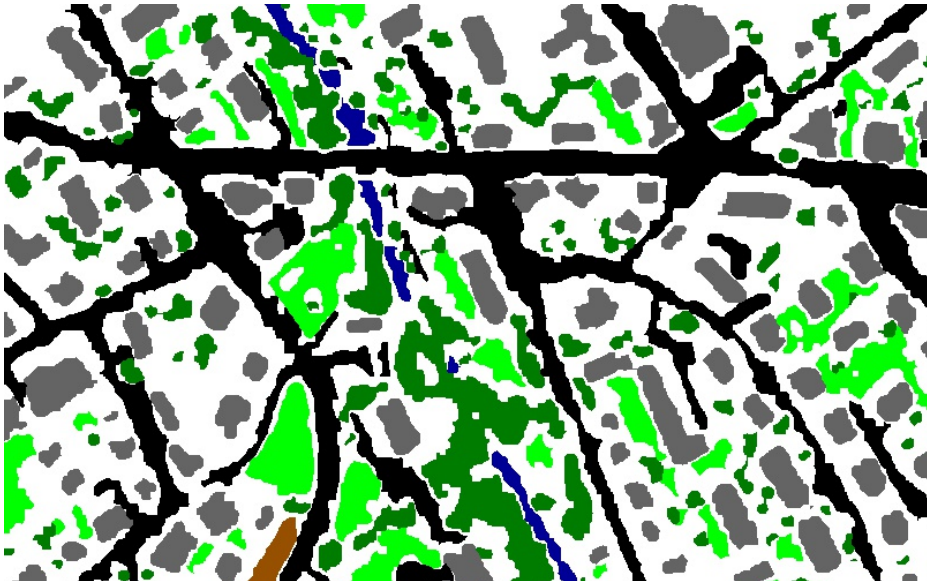


Background Roads Buildings Trees Grass

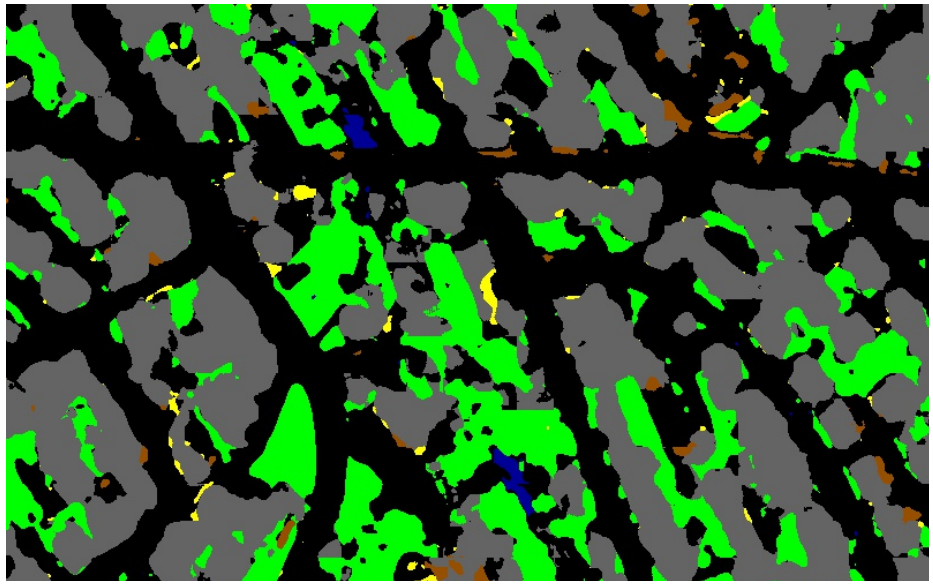
Bare Soil Water Railways Swimming Pools

Zurich Dataset
CNNs

Ground Truth



Prediction (wo. Trees)



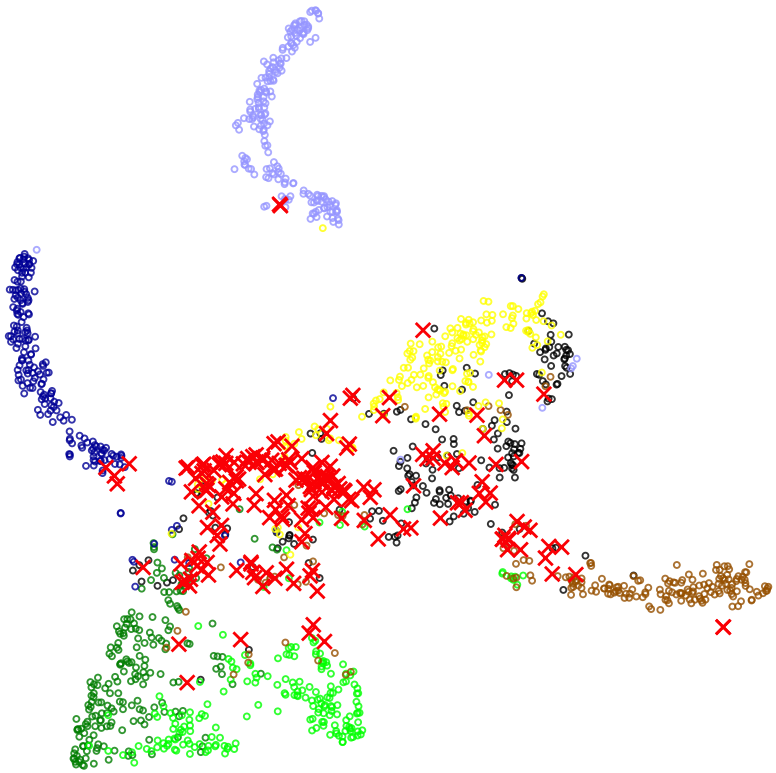
Background Roads Buildings Trees Grass

Bare Soil Water Railways Swimming Pools

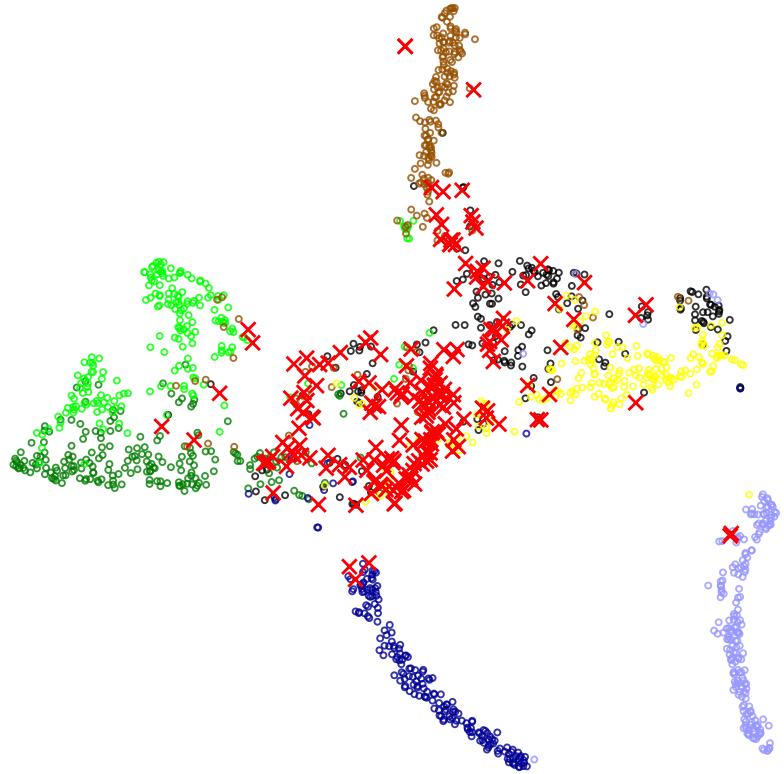
Novelty Detection

Are the activations separable?

t-SNE before PCA



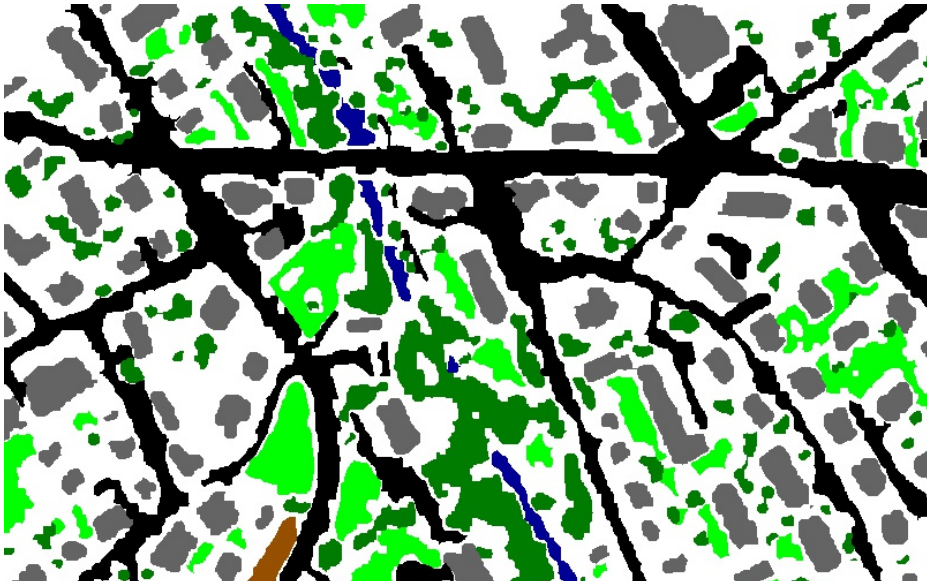
t-SNE after PCA



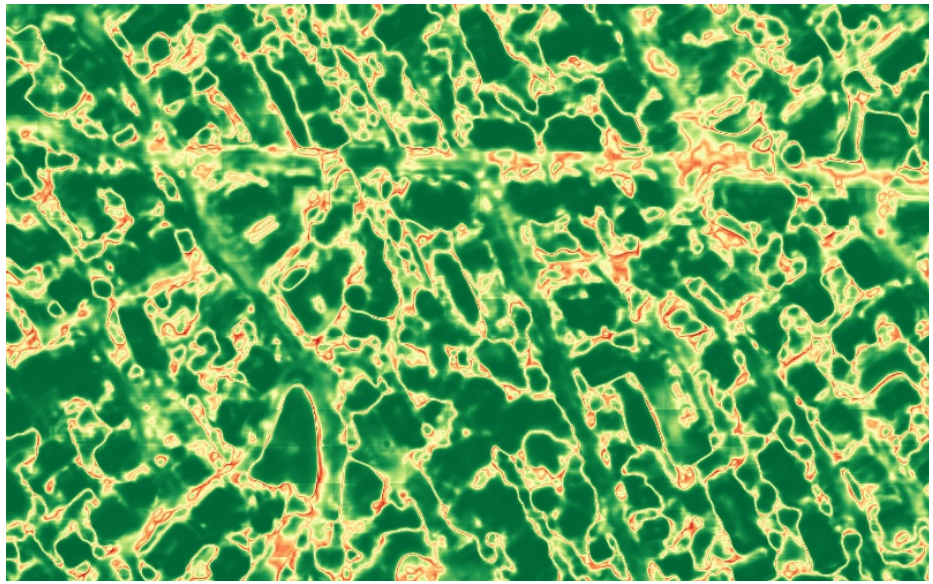
Class: • Roads • Buildings • Trees • Grass
• Bare Soil • Water • Railways • Pools
× Unseen class

Novelty Detection
Unseen Class **Roads**

Ground Truth



MSR



Ground Truth

Background Roads Buildings Trees Grass

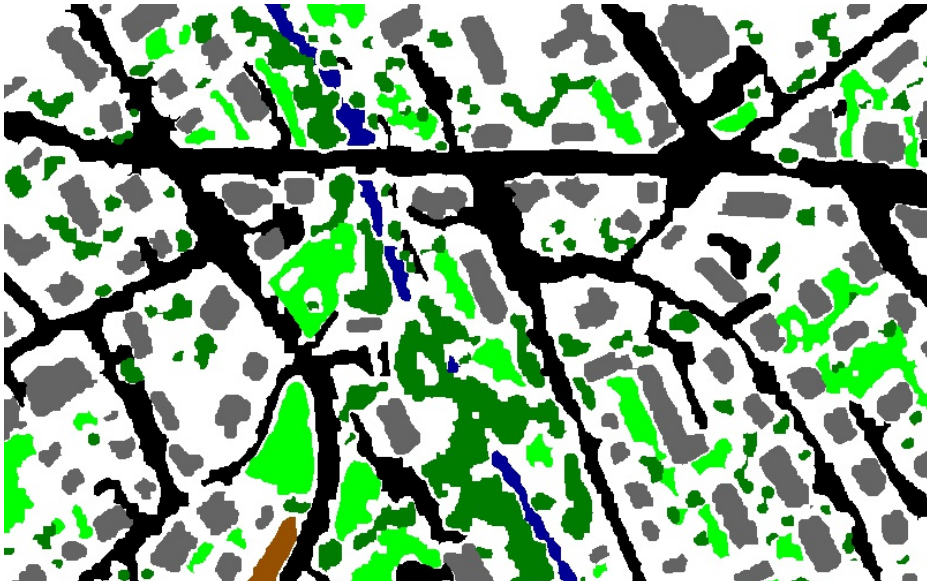
Bare Soil Water Railways Swimming Pools

Confidence

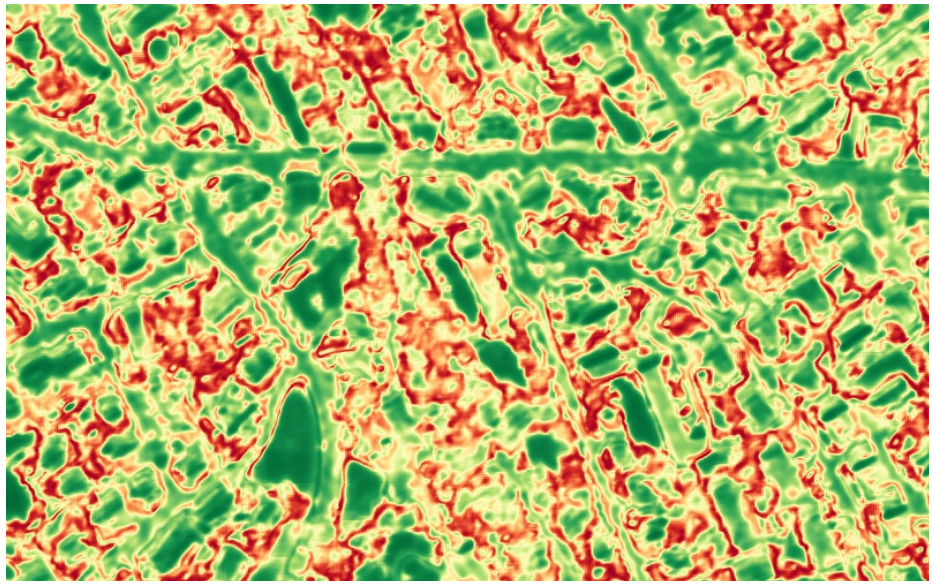
Low High

Novelty Detection
Unseen Class **Roads**

Ground Truth



OC-SVM



Ground Truth

Background Roads Buildings Trees Grass

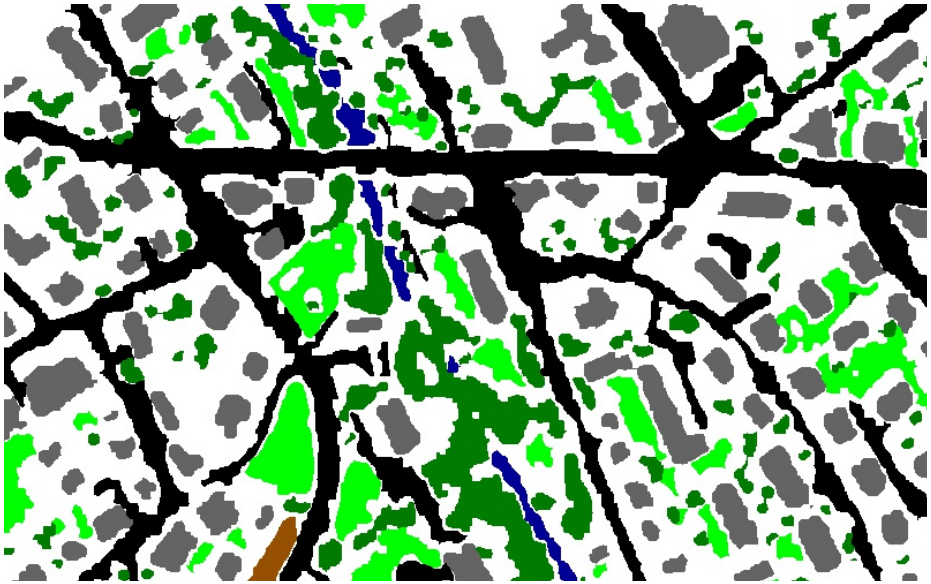
Bare Soil Water Railways Swimming Pools

Confidence

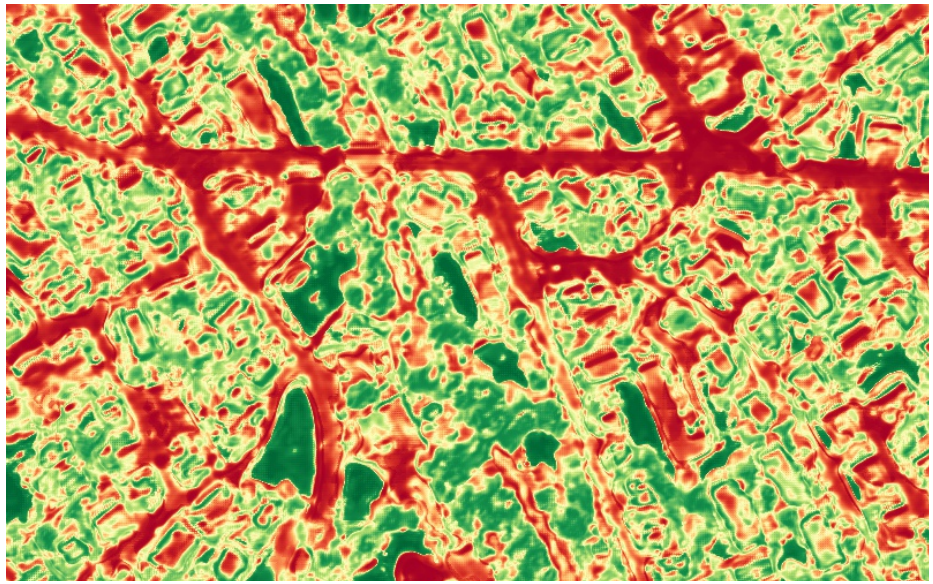
Low High

Novelty Detection
Unseen Class **Roads**

Ground Truth



DF



Ground Truth

Background Roads Buildings Trees Grass

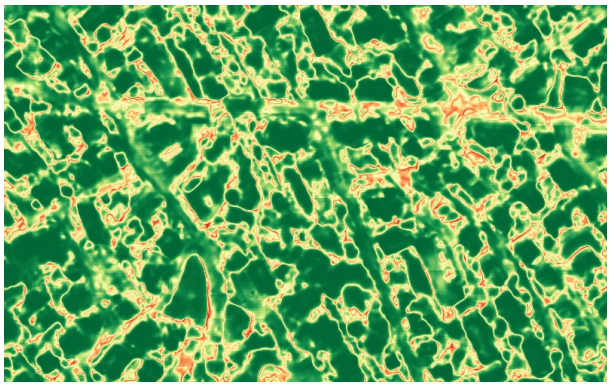
Bare Soil Water Railways Swimming Pools

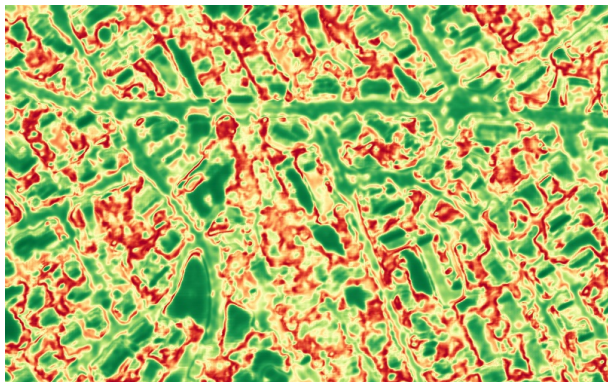
Confidence

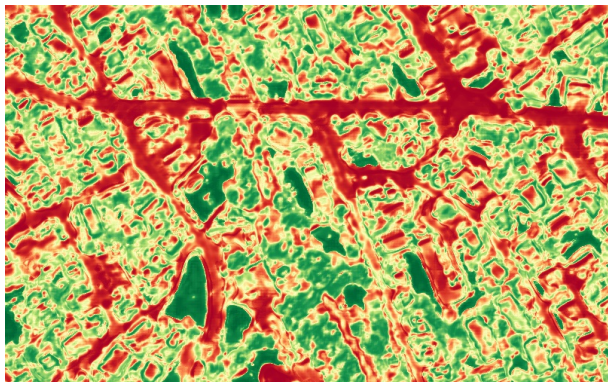
Low High

Novelty Detection

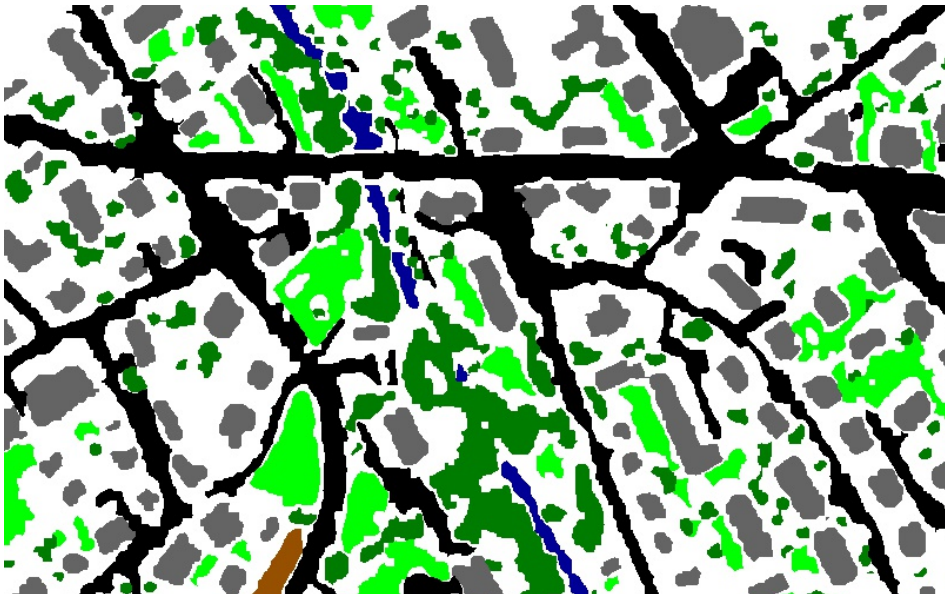
Unseen Class **Roads**

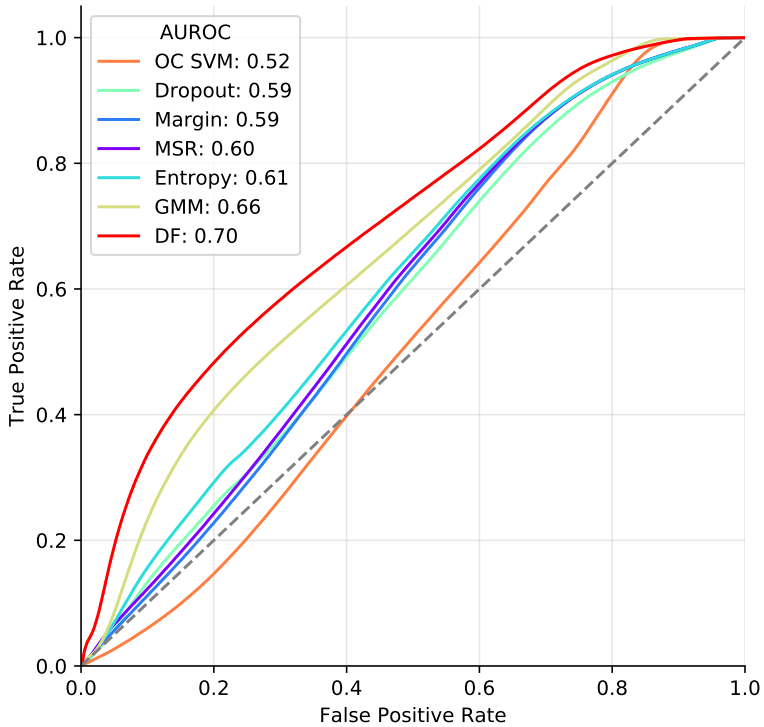






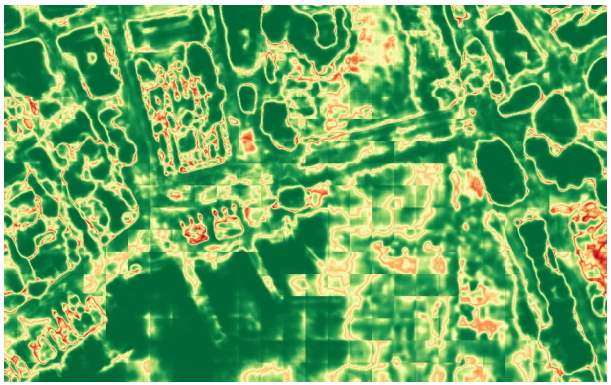
MSR | OC-SVM | DF

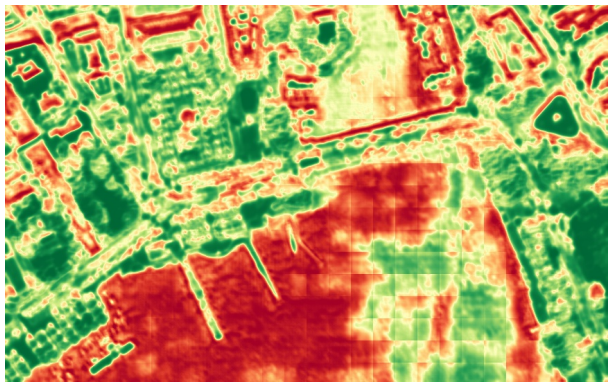


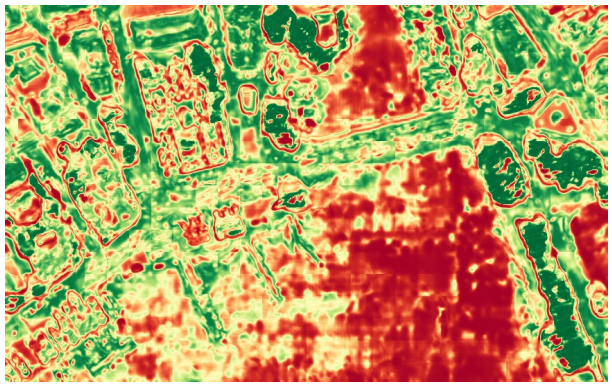


Novelty Detection

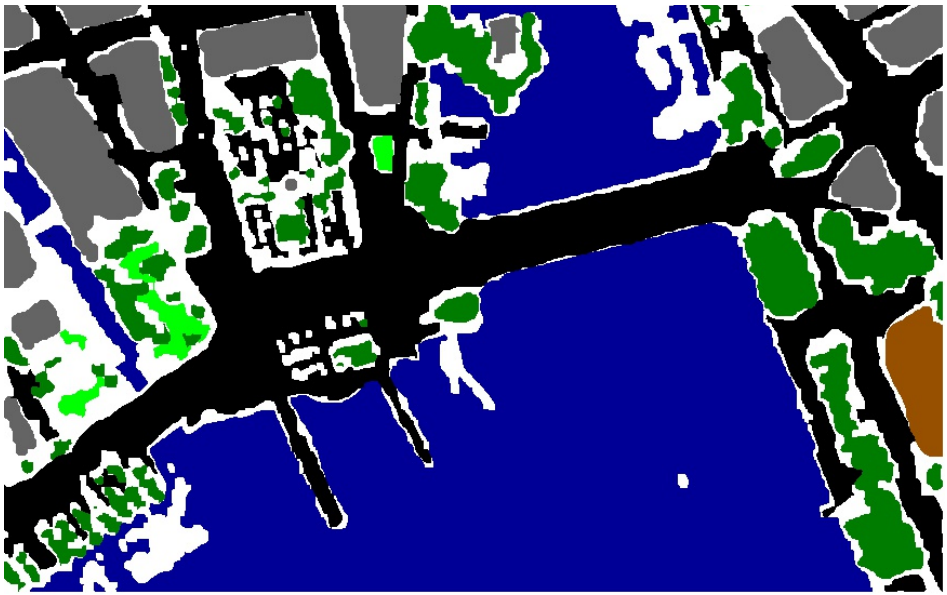
Unseen Class **Water**

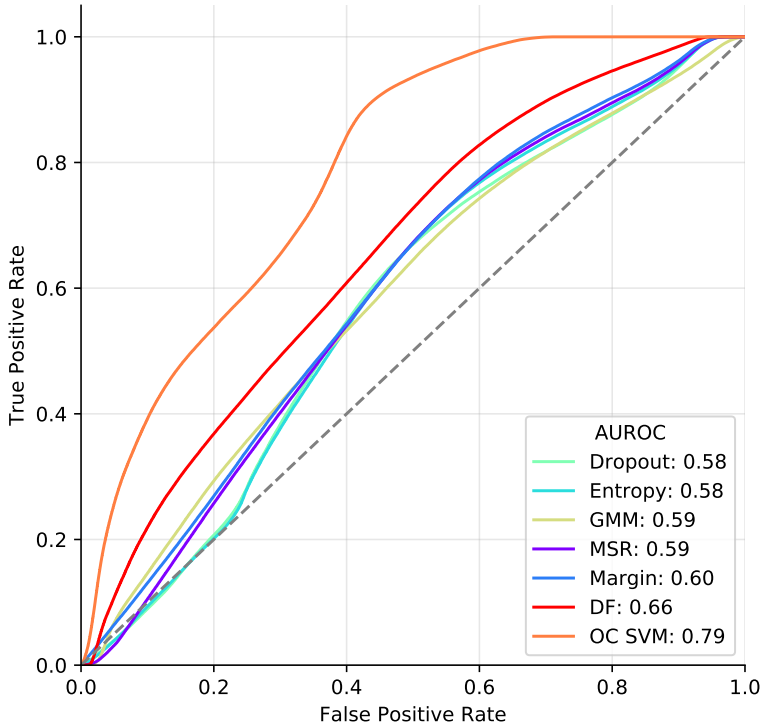






MSR | OC-SVM | DF





Novelty Detection

Best Uncertainty Measures

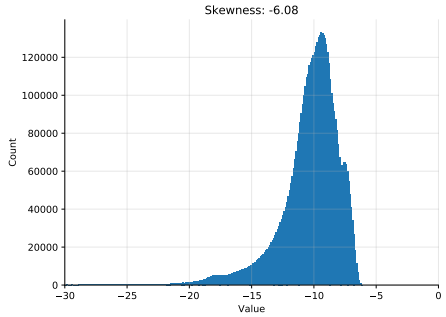
Left-Out Class	MSR	Margin	Entropy	MC-Dropout	GMM	OC-SVM	DF
Roads							✓
Buildings	✓						
Trees						✓	
Grass							✓
Bare Soil		✓					
Water						✓	
Railways						✓	
Swimming Pools					✓		✓
Average						✓	✓

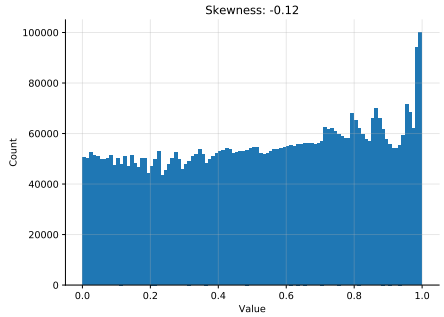
Novelty Detection
Best Uncertainty Measures

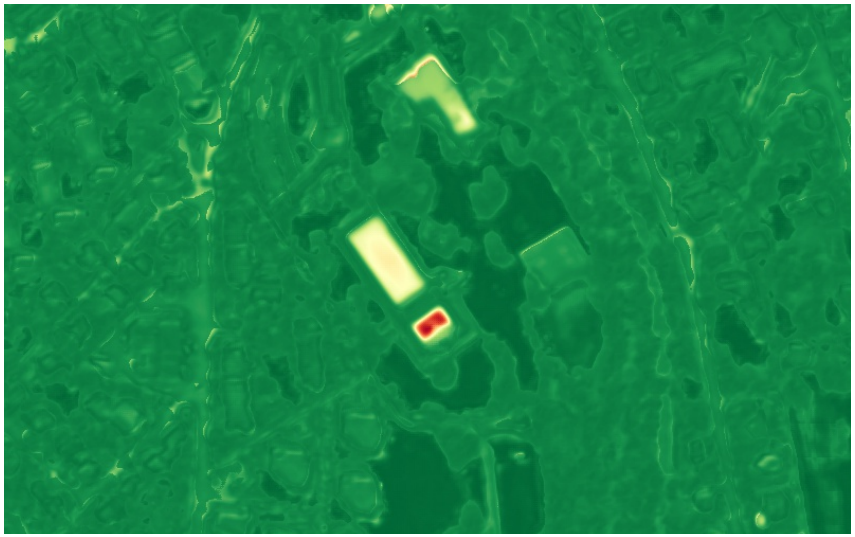
Left-Out Class	MSR	Margin	Entropy	MC-Dropout	GMM	OC-SVM	DF
Roads							✓
Buildings	✓						
Trees						✓	
Grass							✓
Bare Soil		✓					
Water						✓	
Railways						✓	
Swimming Pools					✓		✓
Average						✓	✓

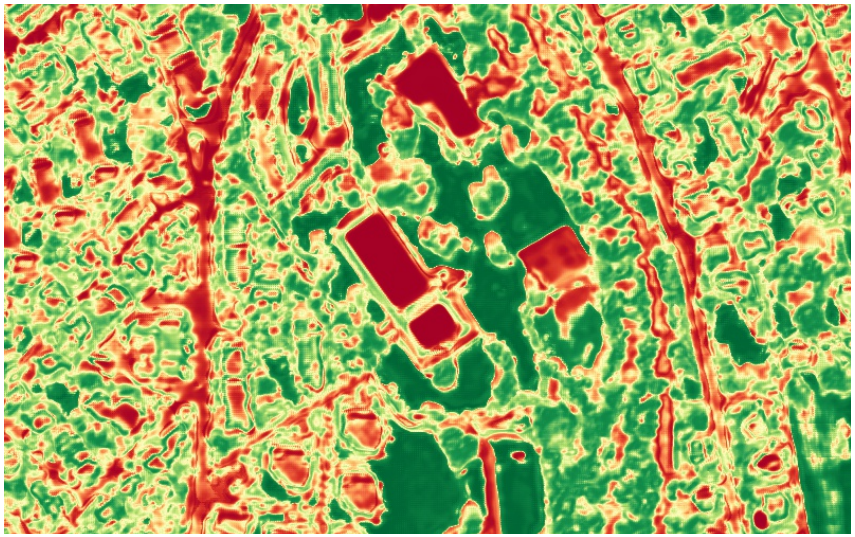
:)

Particular Objects
Histogram Stretching









Original | Equalized

Certainty (DF, left-out class “roads”)

Low  High

Zurich Dataset
Particular Objects

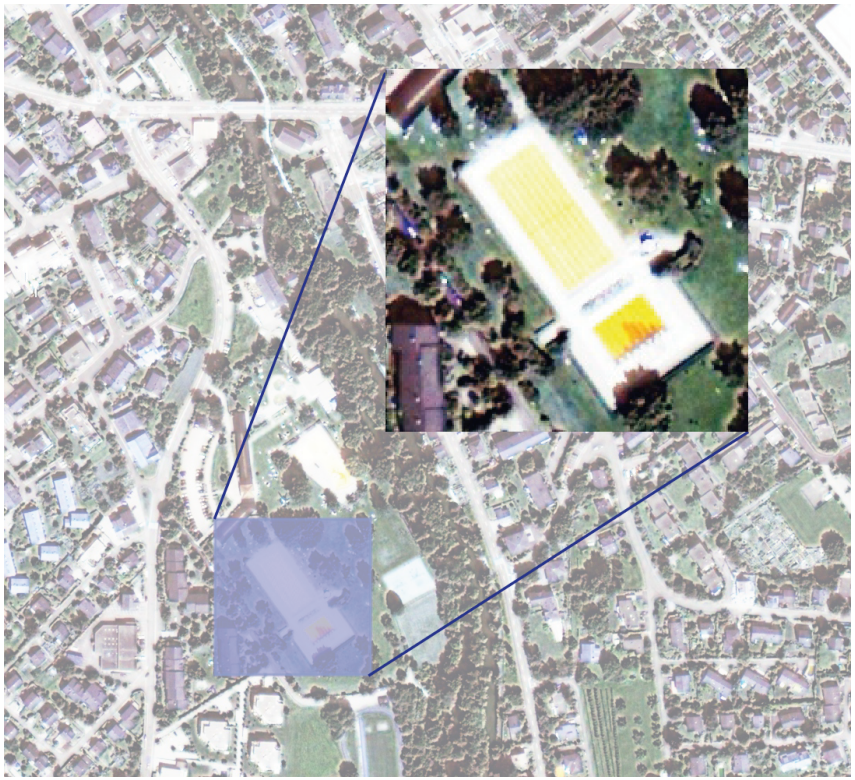
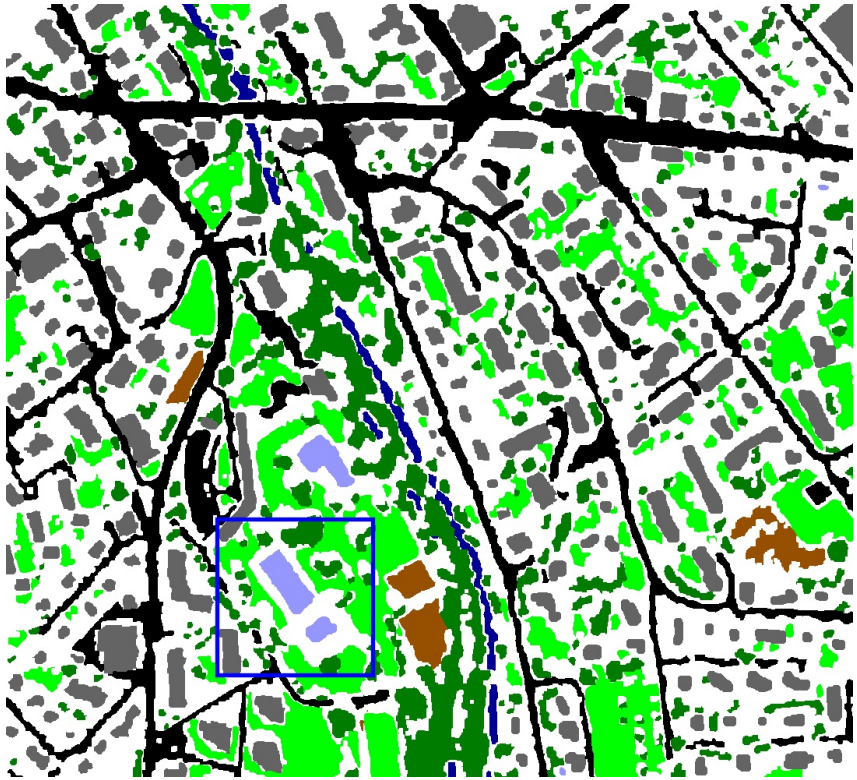


Image with Region of Interest (ROI)

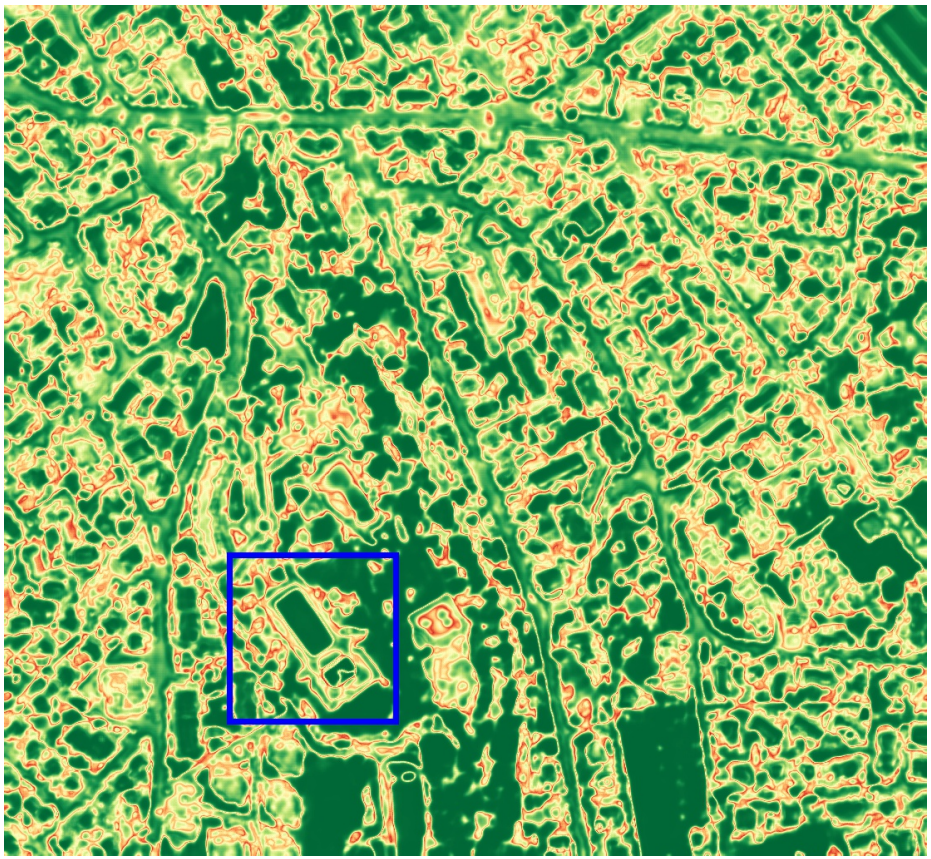


Ground Truth

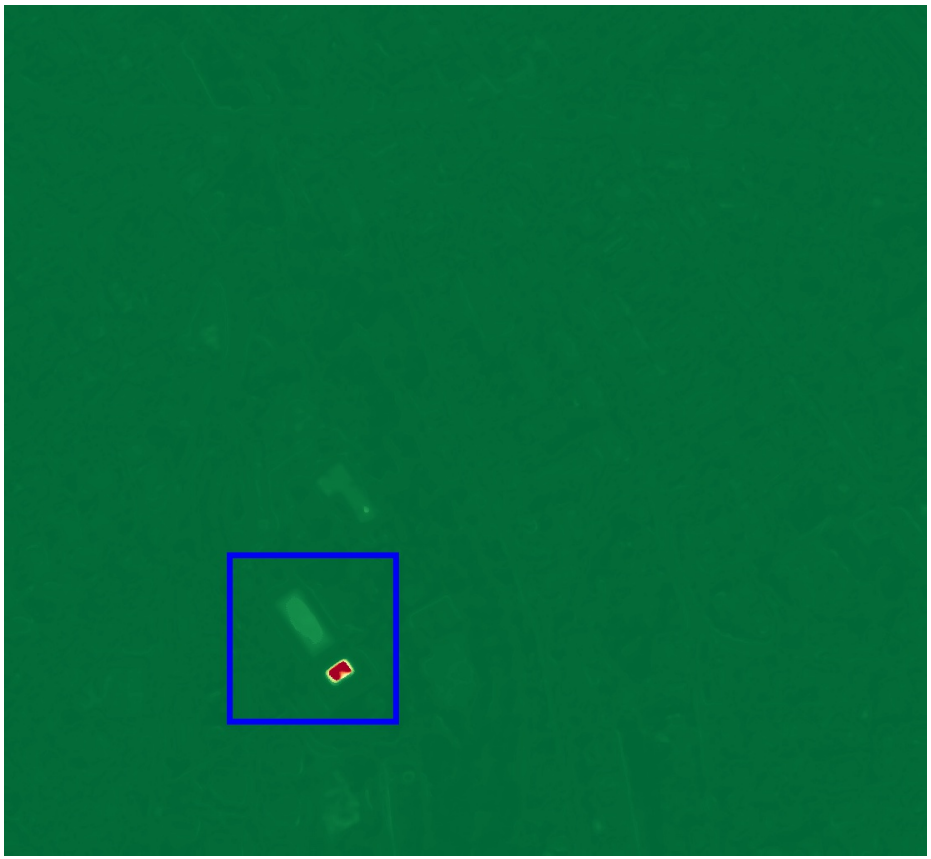
Background | Roads | Buildings | Trees | Grass

Bare Soil | Water | Railways | Swimming Pools

Zurich Dataset
Particular Objects



MSR



Density Forest (non-equalized)

Confidence

Low  High

Zurich Dataset
Particular Objects

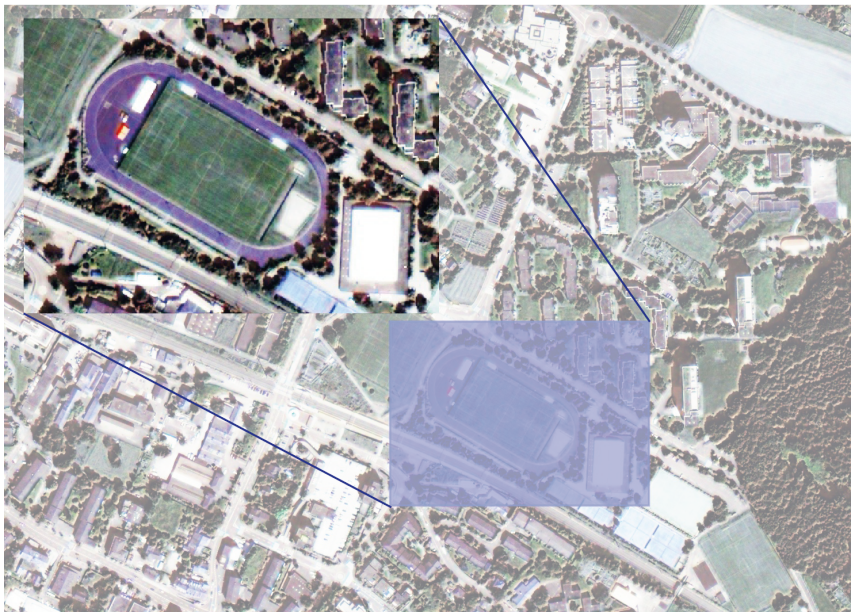
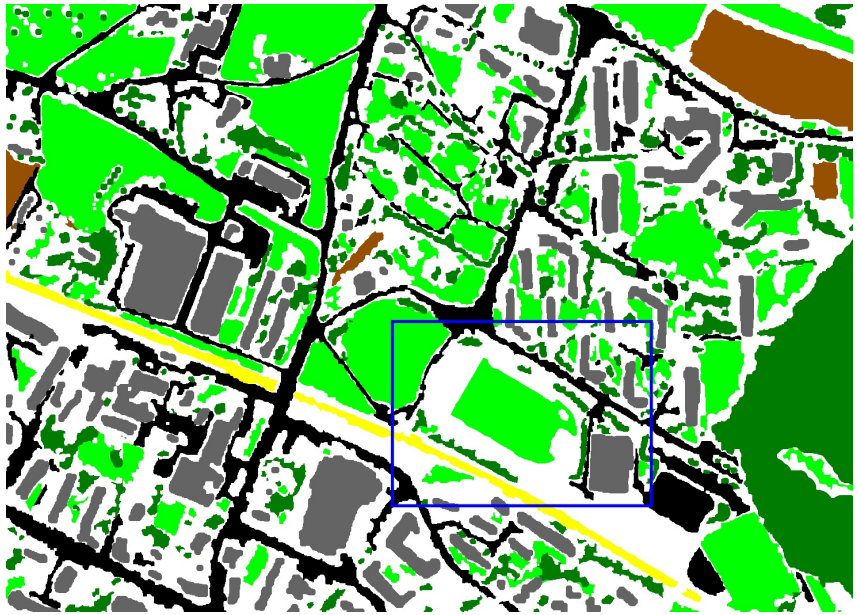


Image with ROI

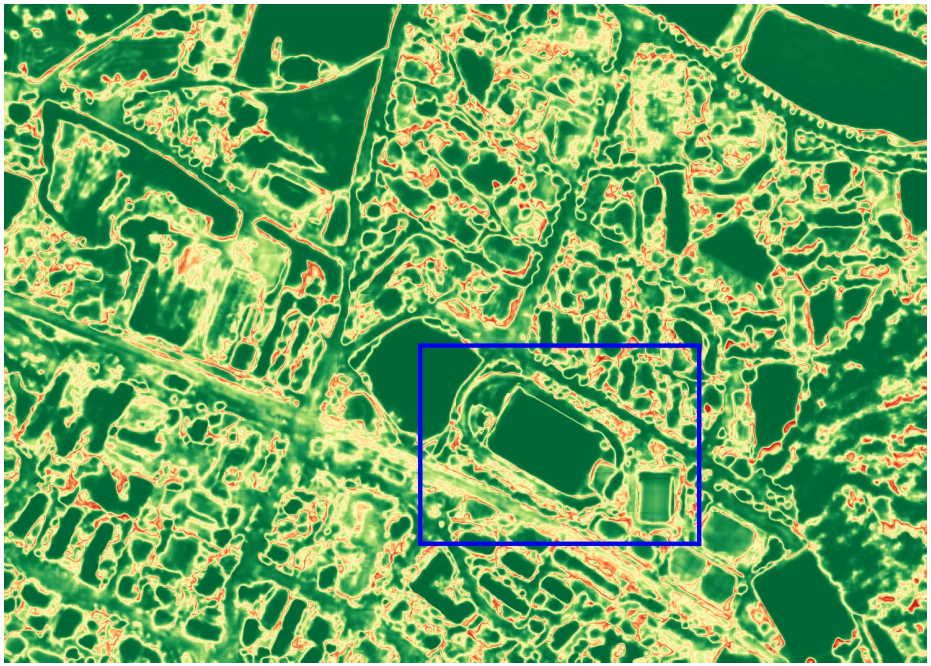


Ground Truth

Background Roads Buildings Trees Grass

Bare Soil Water Railways Swimming Pools

Zurich Dataset
Particular Objects



MSR

Density Forest (non-equalized)

Confidence

Low  High

Discussion

Winning methods

- MNIST dataset: softmax-based methods
- Zurich dataset: pre-softmax-based methods
  - OC-SVM and Density Forest work particularly well

Varying Performance
MNIST and Zurich dataset

Intuitions

- Number of pre-softmax activations
  - MNIST: 128 components, Zurich: 32 components
  - Curse of Dimensionality
- Data complexity

Varying Performance
Zurich Dataset

- Class imbalance and class separability
  - Similar classes: Roads and buildings, trees and grass
  - Distinctive classes: Bare soil, swimming pools

Conclusion
Open Questions

Influence of...

- dimensionality?
- data complexity?
- class separability?
- parameter sensitivity?

Conclusion
Main contributions

1. Implementation of a Python library for creating Density Forests
    - Installation:
      `pip install density_forest`

Conclusion
Main contributions

1. Implementation of a Python library for creating Density Forests
   - Installation:
     `pip install density_forest`
   - Syntax:
     `model.fit(X_train, **kwargs)`
     `model.predict(X_test)`

Conclusion
Main contributions

1. Implementation of a Python library for creating Density Forests
   - Installation:
     `pip install density_forest`
   - Syntax:
     `model.fit(X_train, **kwargs)`
     `model.predict(X_test)`
   - Documentation:
     `http://github.com/CyrilWendl/SIE-Master`

Conclusion
Main contributions

1. Implementation of a Python library for creating Density Forests
   - Installation:
     `pip install density_forest`
   - Syntax:
     `model.fit(X_train, **kwargs)`
     `model.predict(X_test)`
   - Documentation:
     `http://github.com/CyrilWendl/SIE-Master`
2. Comparison of standard novelty detection baselines
   - Based on softmax scores: MSR, margin, entropy, MC-Dropout
   - Based on pre-softmax activations: GMM, OC-SVM, DF

Conclusion
Main contributions

Demonstration of novelty detection methods using pre-softmax activations in a complex, real-world dataset.

Further applications:

- Change Detection
- Active Learning
- ...

Acknowledgements

Devis Tuia, WUR

Diego Marcos, WUR

François Golay, EPFL

Thank you for your attention!

Thank you for your attention!

Questions?

- Y. LeCun, C. Cortes, and C. Burge. *The MNIST Database of Handwritten Digits*. url: `http://yann.lecun.com/exdb/mnist/`.
- M. Volpi and V. Ferrari. "Semantic segmentation of urban scenes by learning local class interactions". In: *2015 IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)* (2015), pp. 1–9.