Novelty Detection in Convolutional Neural Networks Using Density Forests

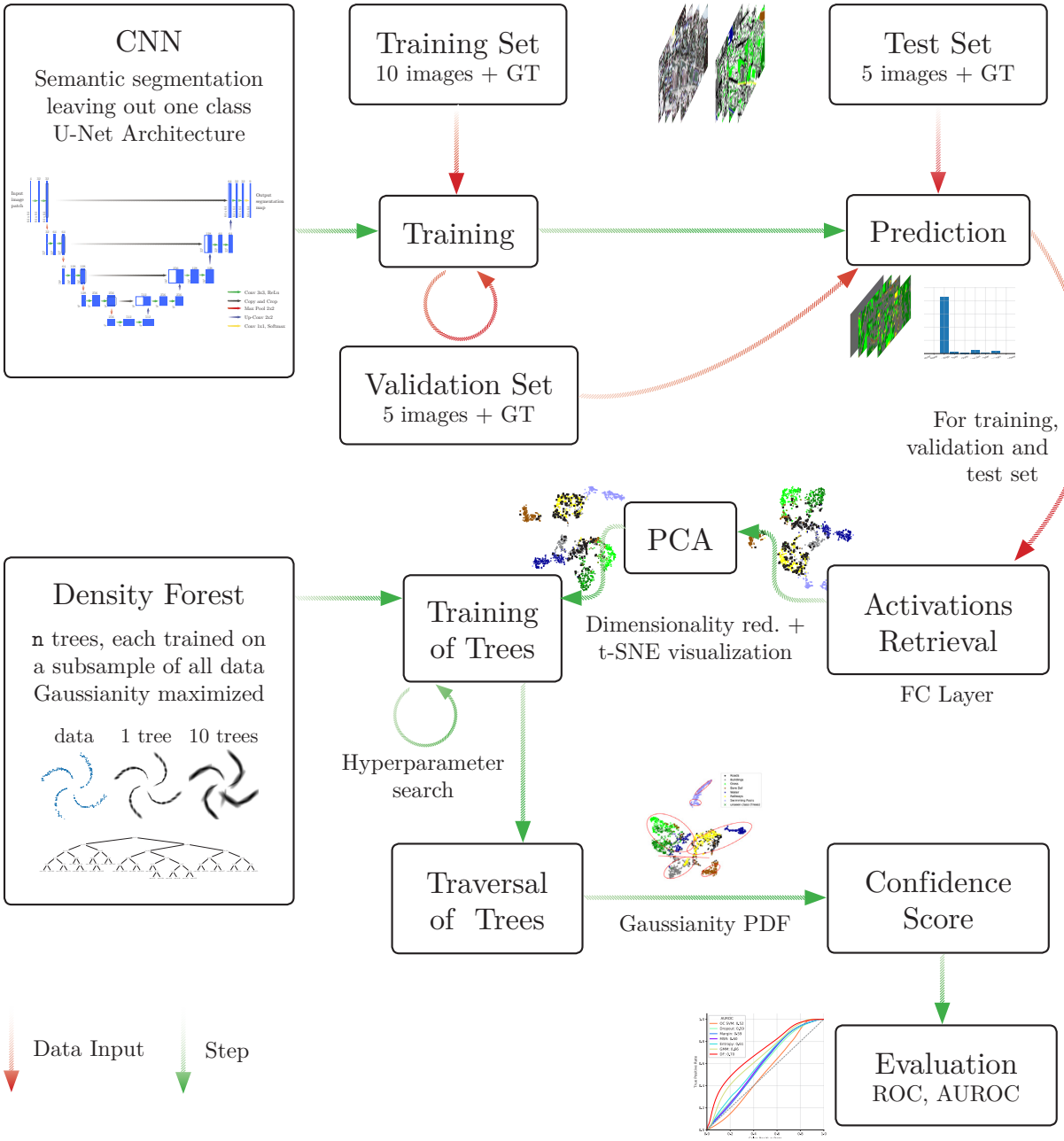
Master Thesis

Cyril Wendl

Environmental Sciences and Engineering
EPFL

07/09/2018

Outline


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
Density Forest
n trees, each trained on
a subsample of all data
Gaussianity maximized
data
1 tree
10 trees
PCA
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

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: $\mathcal{L} = \{c_1, c_2\}$

$p(c_1) = ?$

$p(c_2) = ?$

What is Uncertainty?

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

$\Rightarrow$ Model M $\Rightarrow$

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

$p(c_1) = 1$

$p(c_2) = 0$

What is Uncertainty?

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

$\Rightarrow$ 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?

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

What is Uncertainty?

Uncertainty

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

Types of Uncertainty Measures

- Based on new network architectures
- **Using standard network architectures**

What is Uncertainty?

Uncertainty

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

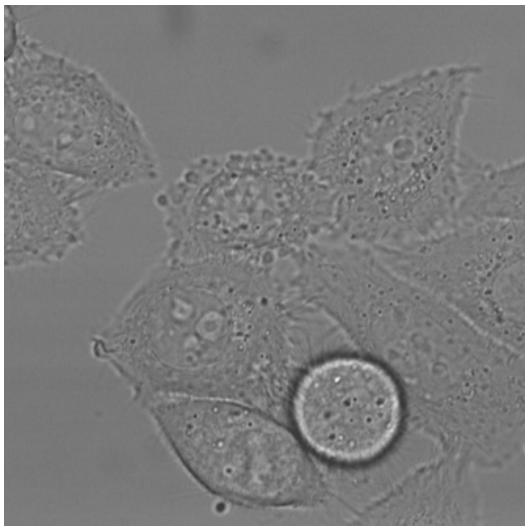
Types of Uncertainty Measures

- Based on new network architectures
- **Using standard network architectures**

Evaluation heuristics

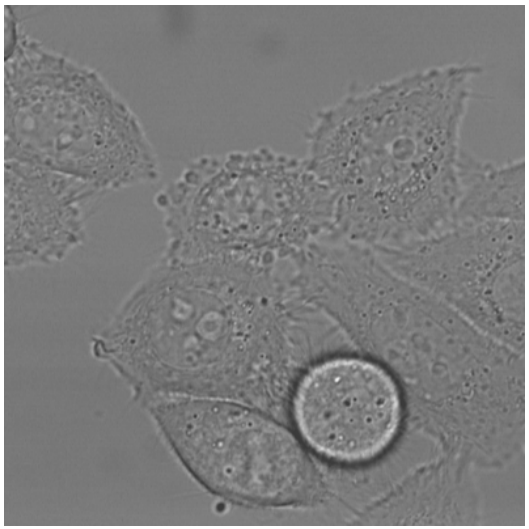
- Error detection: wrong prediction $\Rightarrow$ low confidence
- **Novelty detection: unseen class $\Rightarrow$ low confidence**

Relevant Applications of Uncertainty



Medical imaging

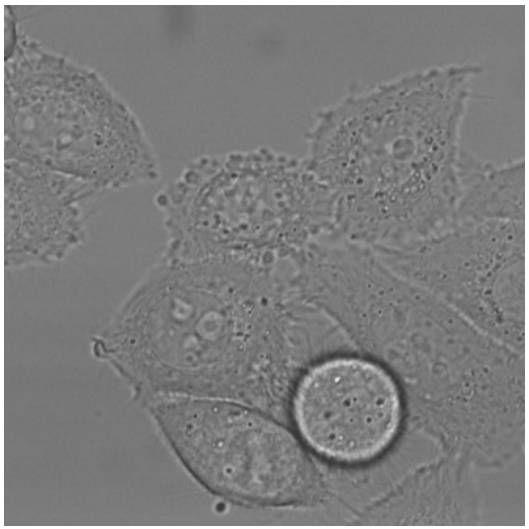
Relevant Applications of Uncertainty



Medical imaging

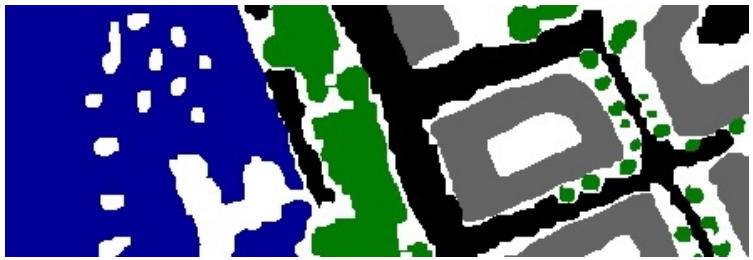
Autonomous cars

Relevant Applications of Uncertainty



Medical imaging

Autonomous cars



Land Cover Classification

Research Objectives

- Determine uncertainty of standard Convolutional Neural Network (CNN) architectures using
  - softmax-based methods
  - methods based on pre-softmax activations
- Implement Density Forests and compare them to baseline methods
- Evaluate these uncertainty measures with respect to their performance for **novelty detection**

Main Contributions

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

Main Contributions

1. Implementation of a Python library for creating Density Forests
   - Installation:
     `pip install density_forest`
   - Documentation:
     `http://github.com/CyrilWendl/SIE-Master`

Main Contributions

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

Main Contributions

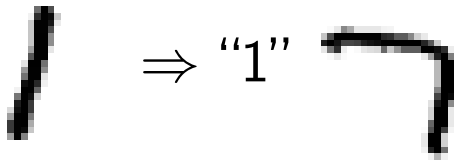
1. Implementation of a Python library for creating Density Forests
   - Installation:
     `pip install density_forest`
   - Documentation:
     `http://github.com/CyrilWendl/SIE-Master`
   - Syntax:
     `model.fit(X_train)`, `model.predict(X_test)`
2. Comparison of standard novelty detection baselines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

Main Contributions

1. Implementation of a Python library for creating Density Forests
   - Installation:
     `pip install density_forest`
   - Documentation:
     `http://github.com/CyrilWendl/SIE-Master`
   - Syntax:
     `model.fit(X_train)`, `model.predict(X_test)`
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
3. **Demonstration of novelty detection methods using pre-softmax activations in a complex, real-world dataset**

Terminology

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

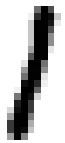


Terminology

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

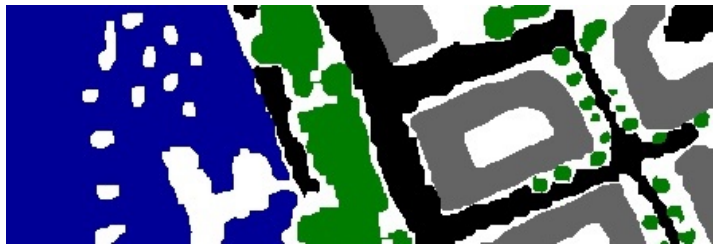

⇒ cat

⇒ dog


⇒ “1”

⇒ “7”

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

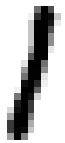

⇒


Terminology

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

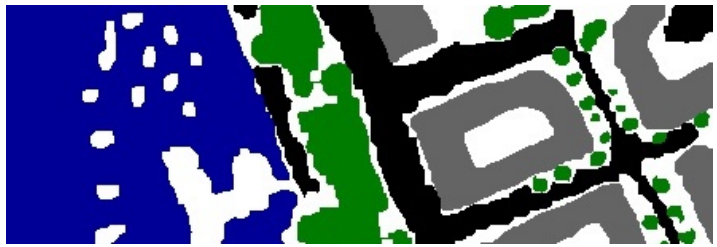

⇒ cat

⇒ dog


⇒ “1”

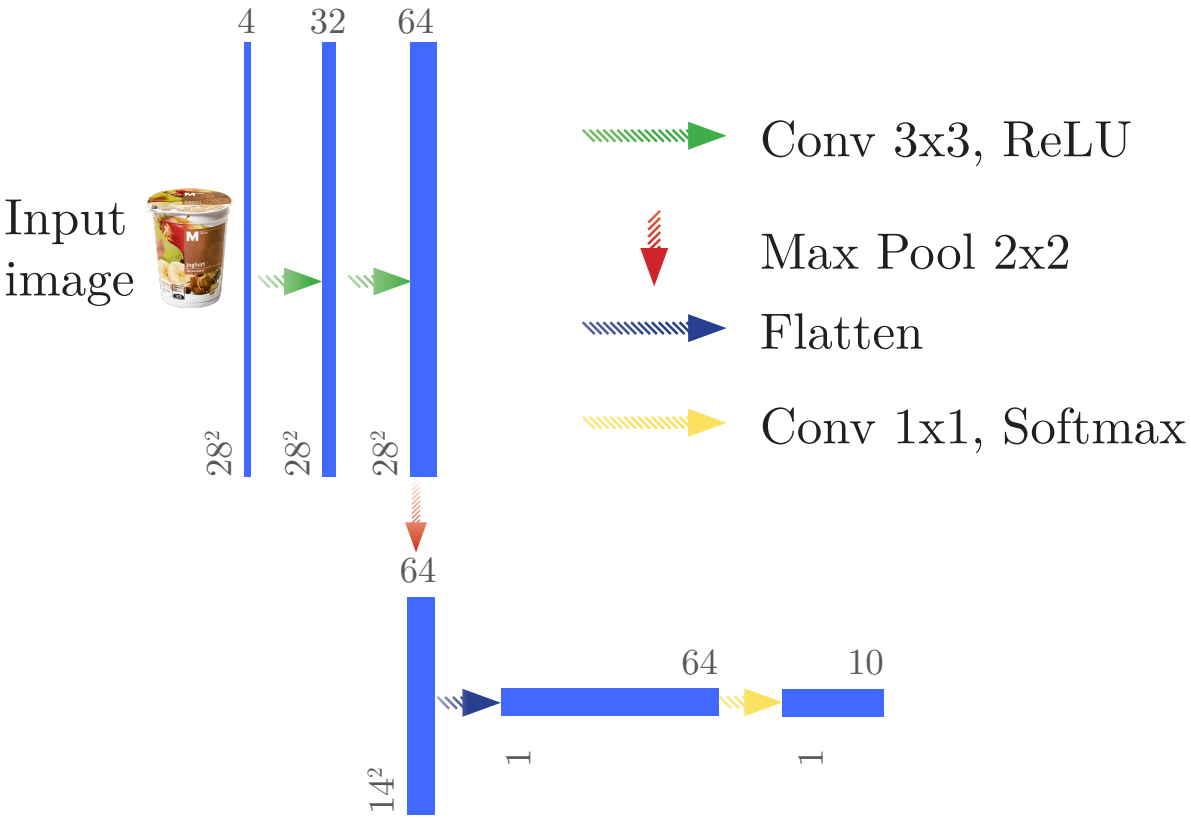
⇒ “7”

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

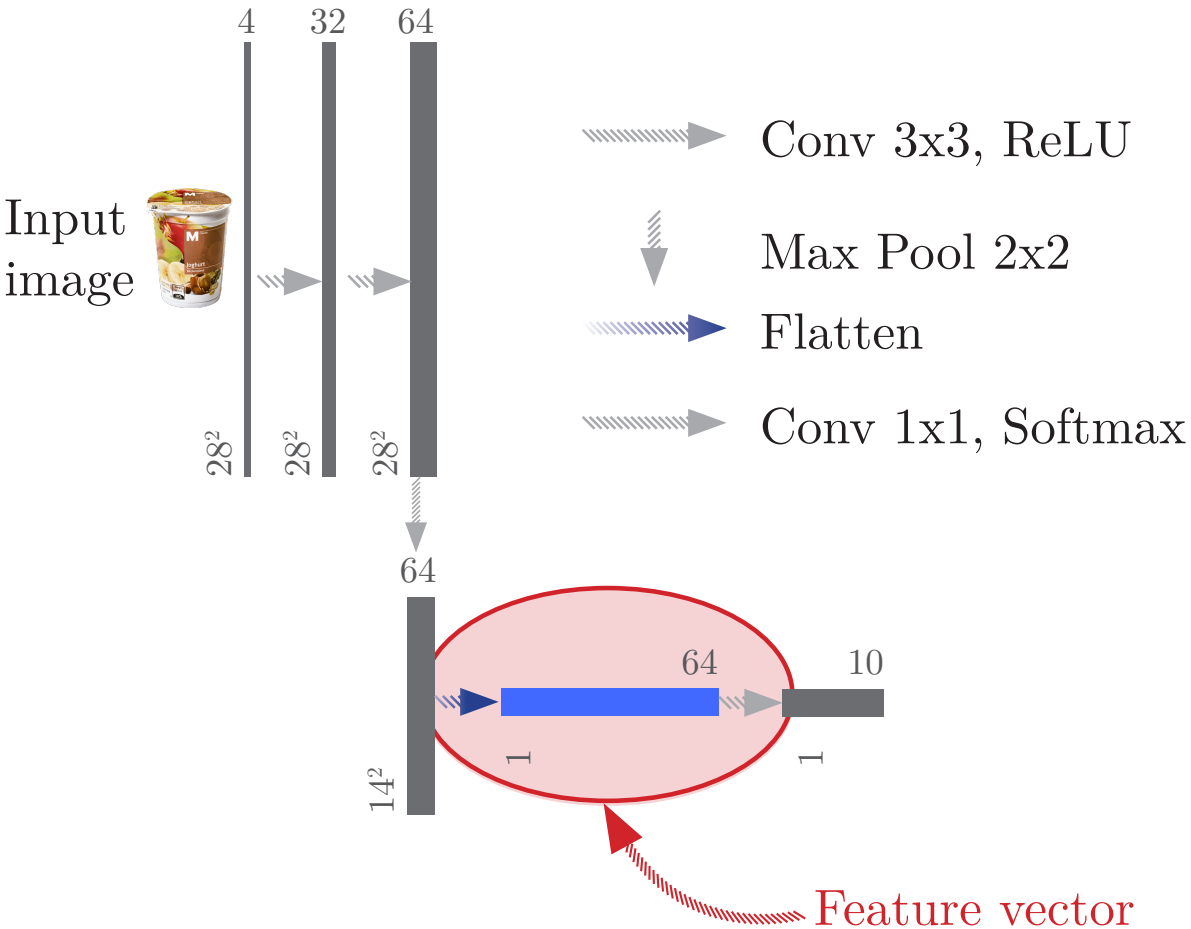

⇒


Classification or regression

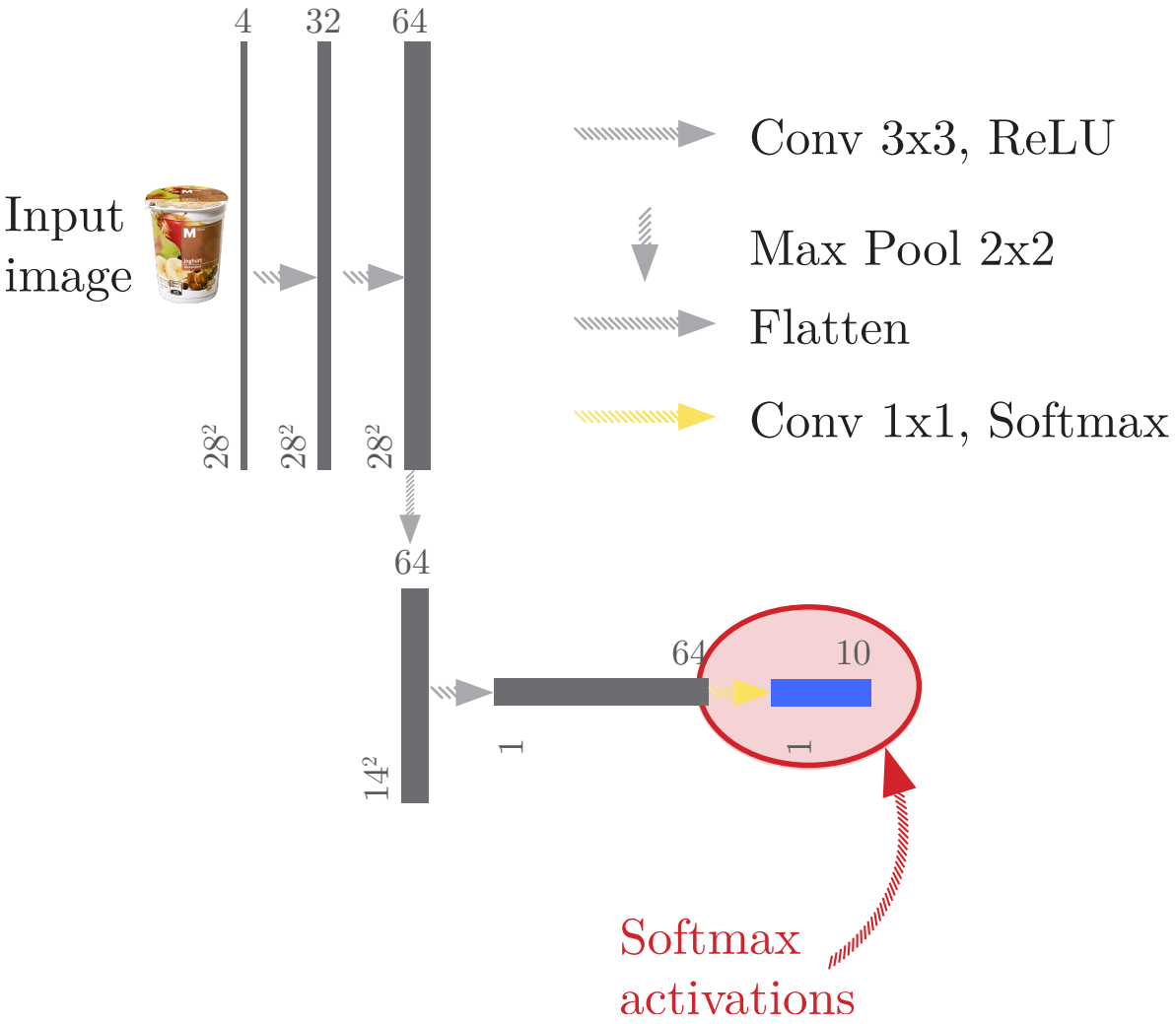
CNNs


4
32
64
Input image
28^2
28^2
28^2
Conv 3x3, ReLU
Max Pool 2x2
Flatten
Conv 1x1, Softmax
64
64
10
14^2
1
1

CNNs


4
32
64
Input image
28^2
28^2
28^2
Conv 3x3, ReLU
Max Pool 2x2
Flatten
Conv 1x1, Softmax
64
64
10
1
1
14^2
Feature vector

CNNs



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

No real probability!

Problems with Softmax Activations

- Can be easily fooled

 $\Rightarrow$ cat  $\Rightarrow$ dog

Problems with Softmax Activations

- Can be easily fooled

 ⇒ cat  ⇒ dog

- Not robust to transformations

 ⇒ cat  ⇒ dog

Problems with Softmax Activations

- Can be easily fooled

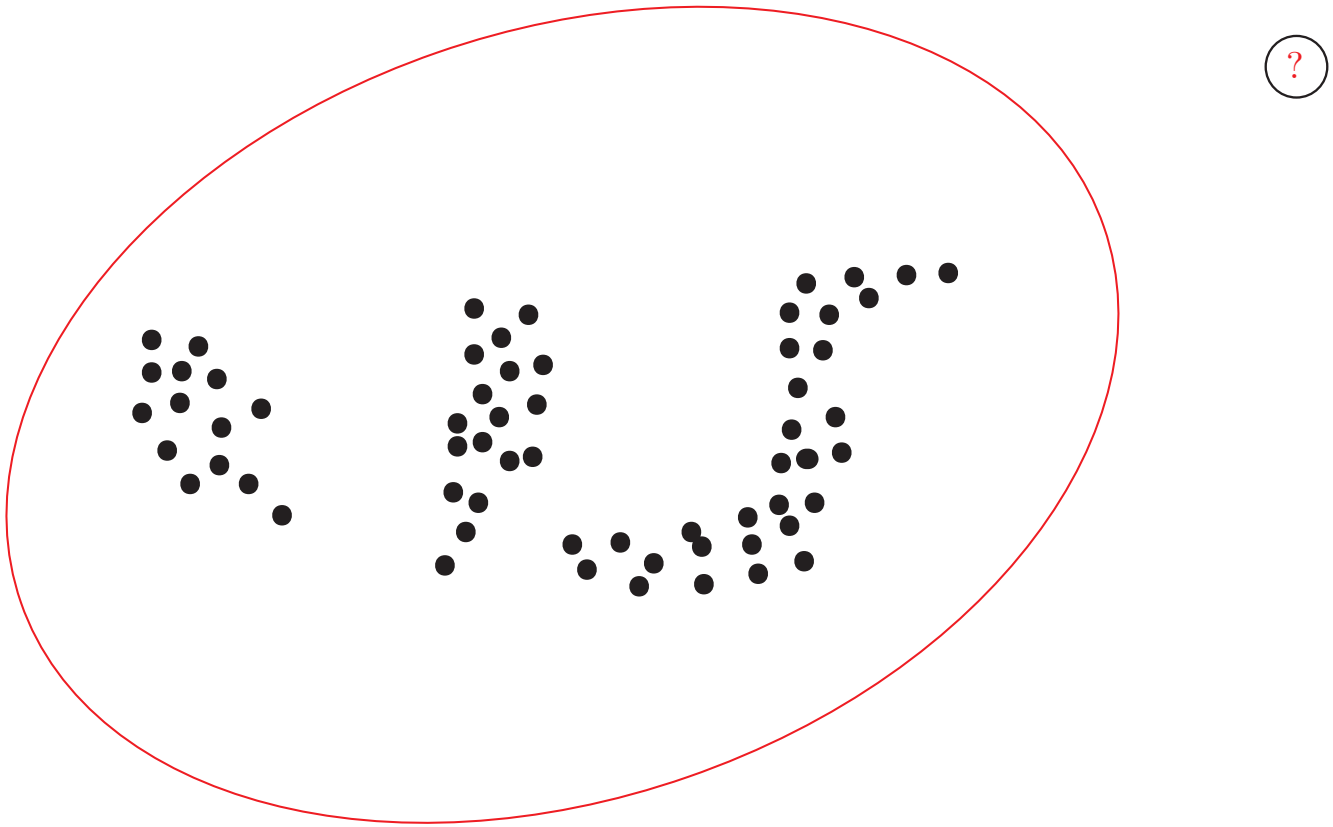
 $\Rightarrow$ cat  $\Rightarrow$ dog

- Not robust to transformations

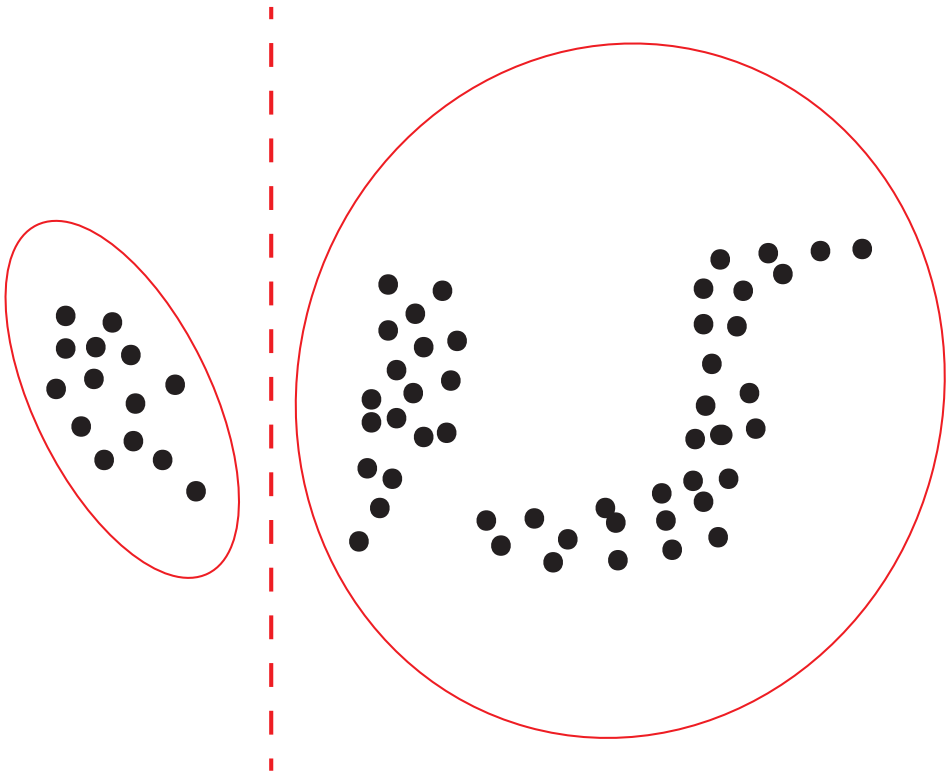
 $\Rightarrow$ cat  $\Rightarrow$ dog

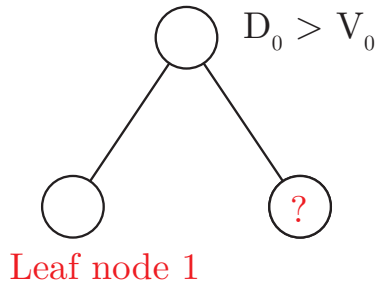
- Can yield high scores despite being wrong

Density Trees

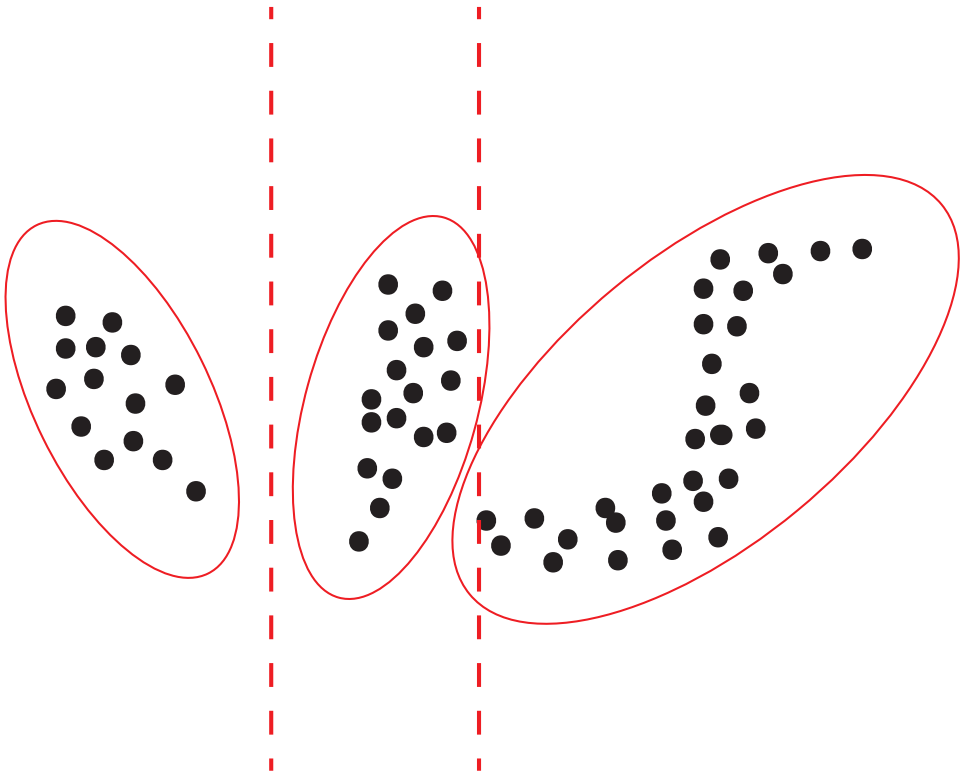


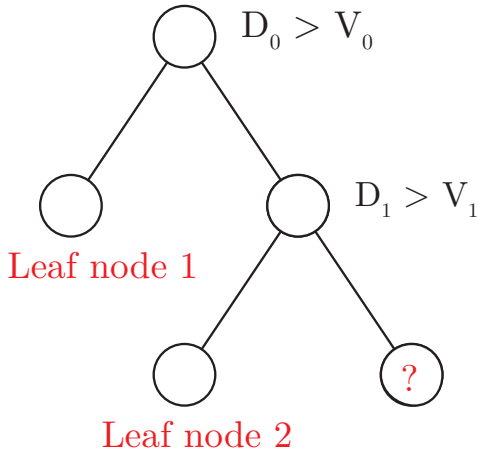
Density Trees



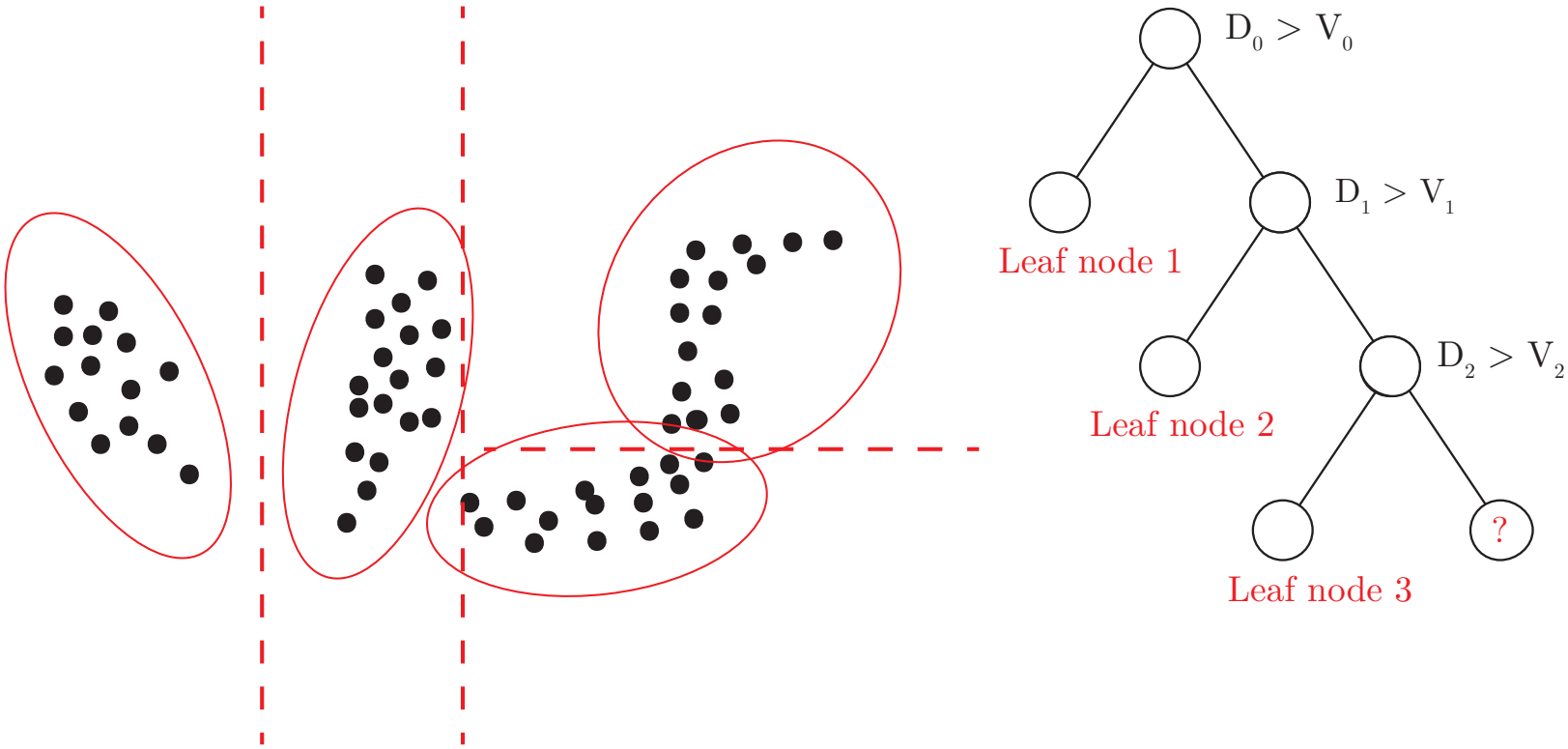


Density Trees

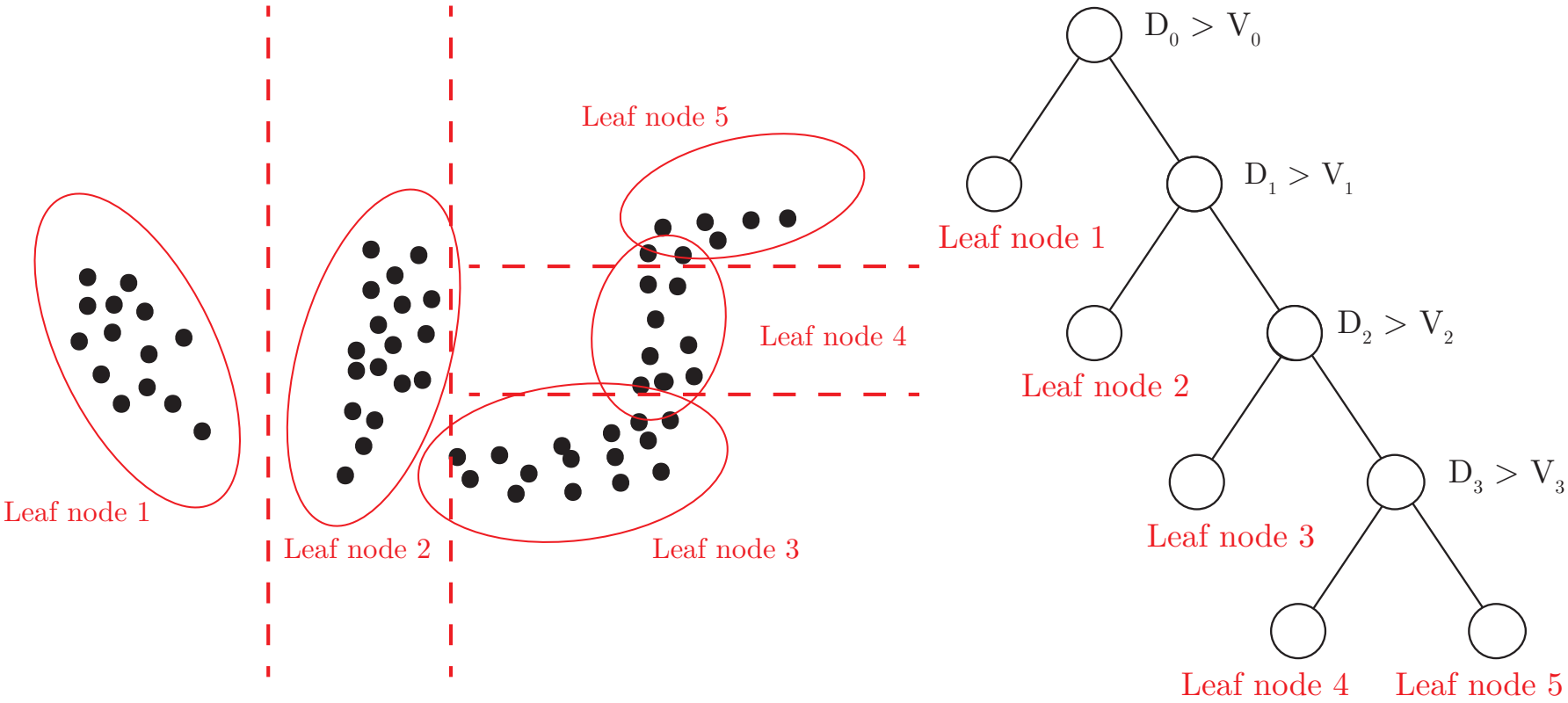




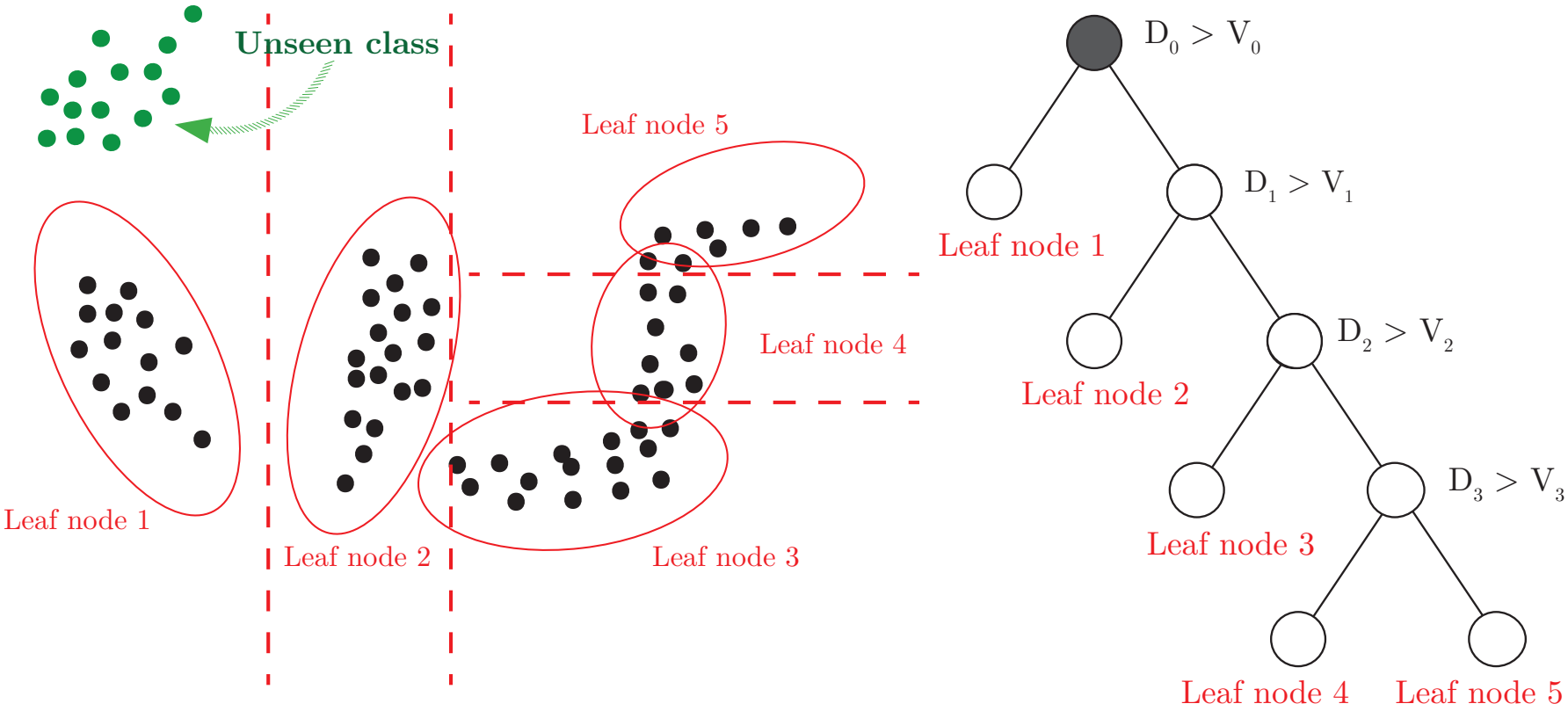
Density Trees



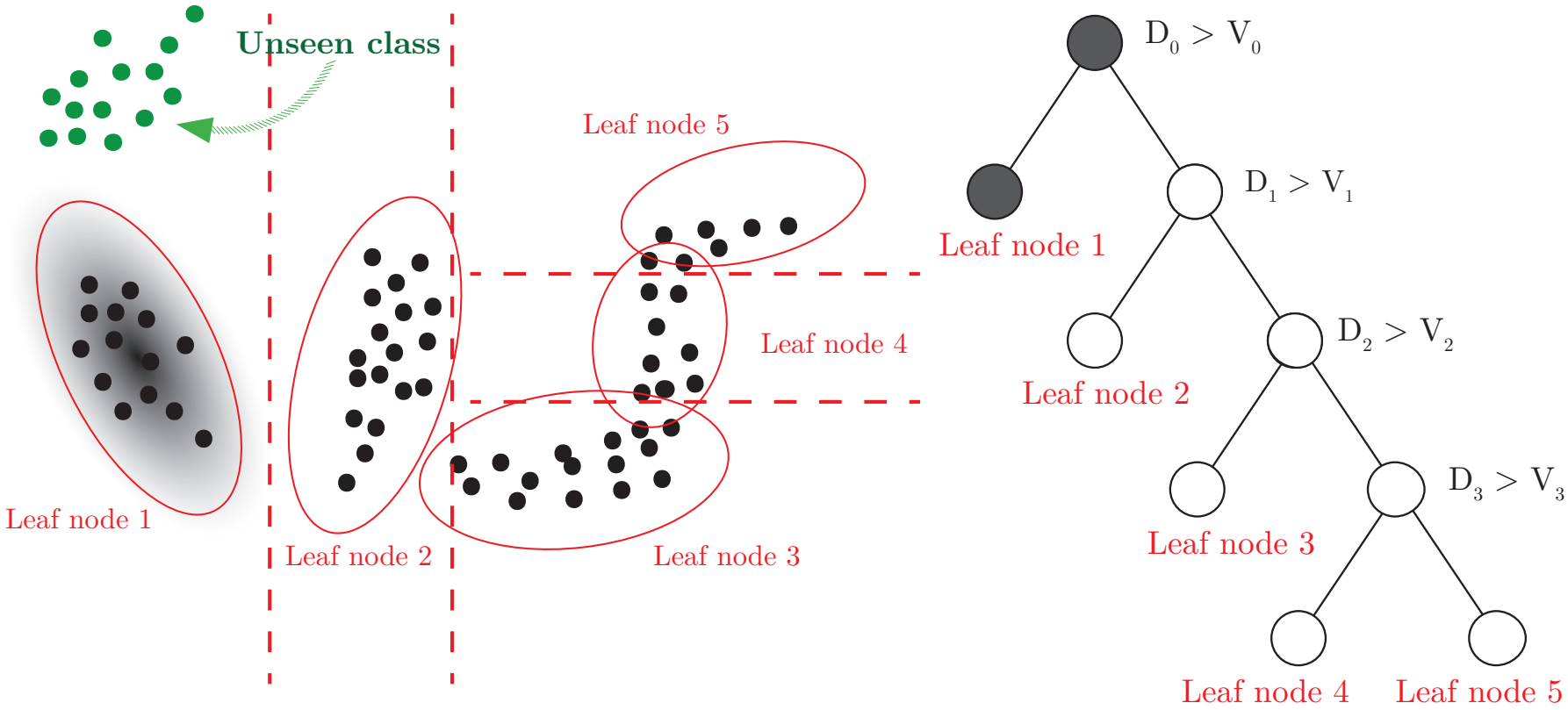
Density Trees



Density Trees



Density Trees



Density Forests

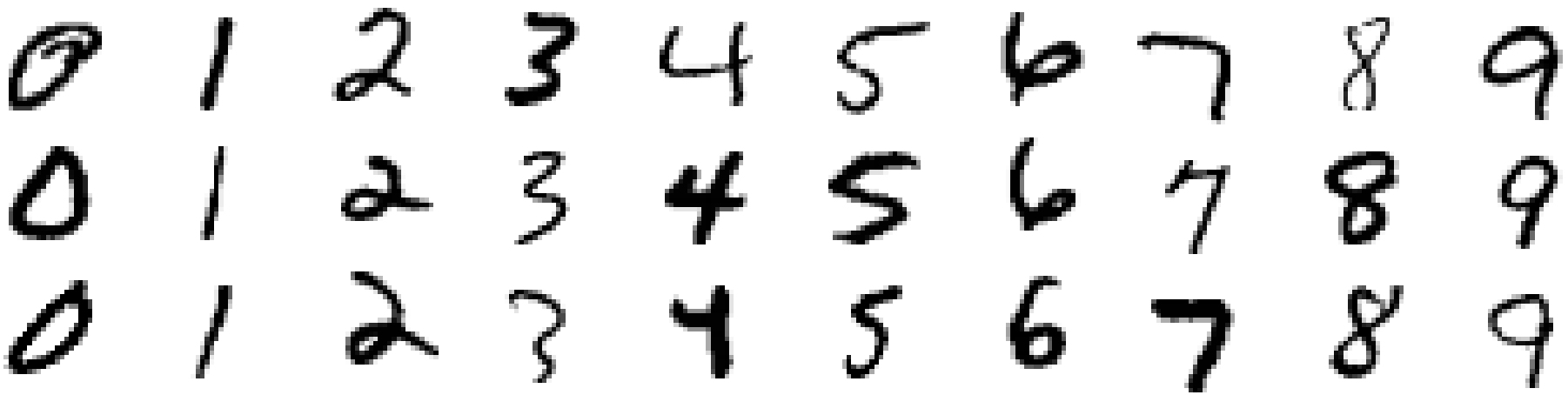
- Finding best split by maximizing **Gaussianity** on each side
- Combination of multiple weak learners

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

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

Datasets

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

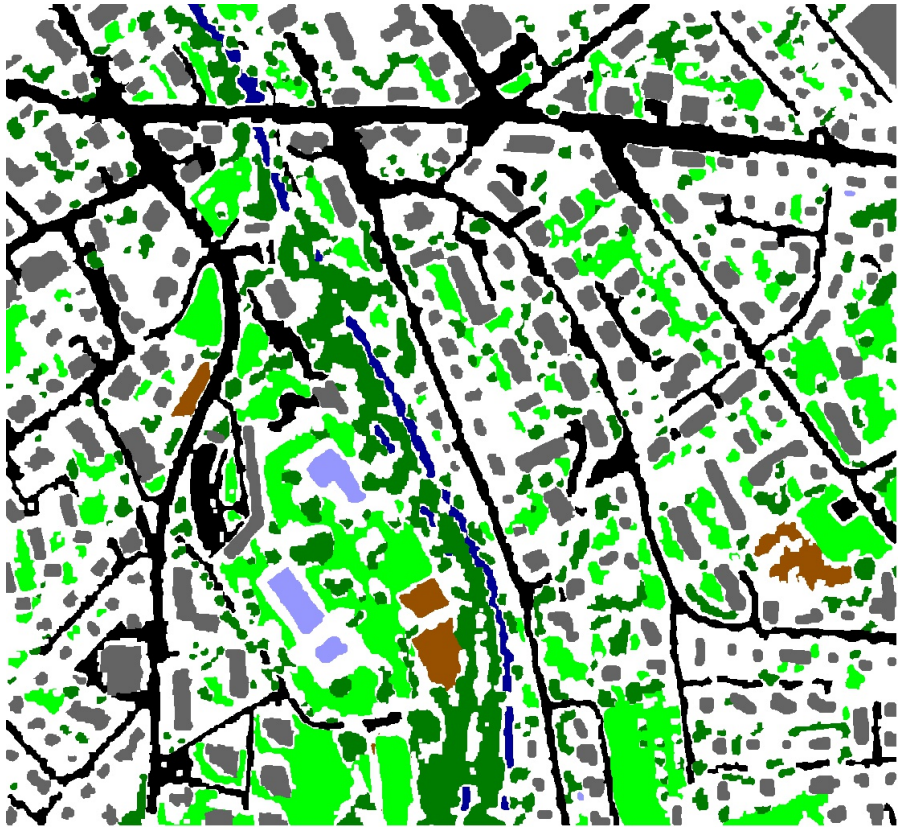


- 60'000 training images
- 10'000 test images

Datasets
Zurich Dataset [2]

RGB (+ IR) Image

Ground Truth



Background Roads Buildings Trees Grass

Bare Soil Water Railways Swimming Pools

Training Setup

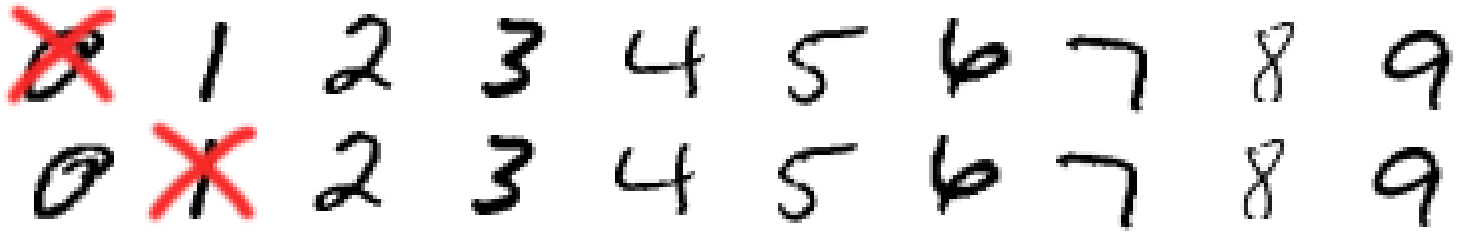
CNNs

- Standard network architectures
- Leaving out one class during training

Training Setup

CNNs

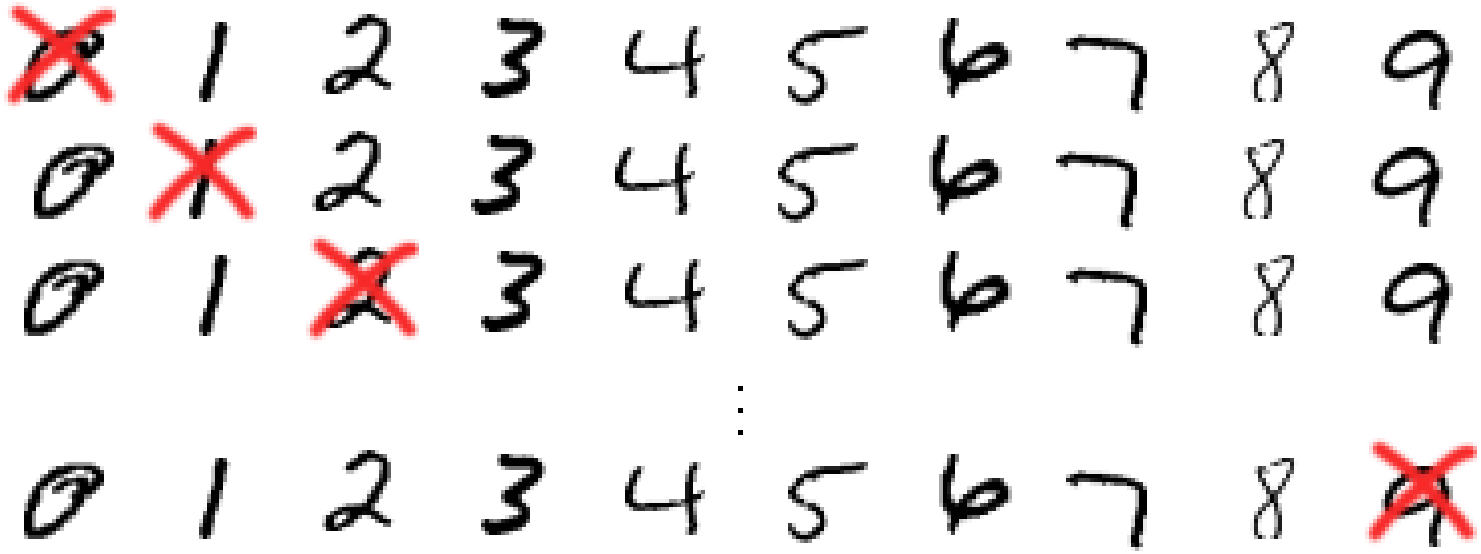
- Standard network architectures
- Leaving out one class during training



Training Setup

CNNs

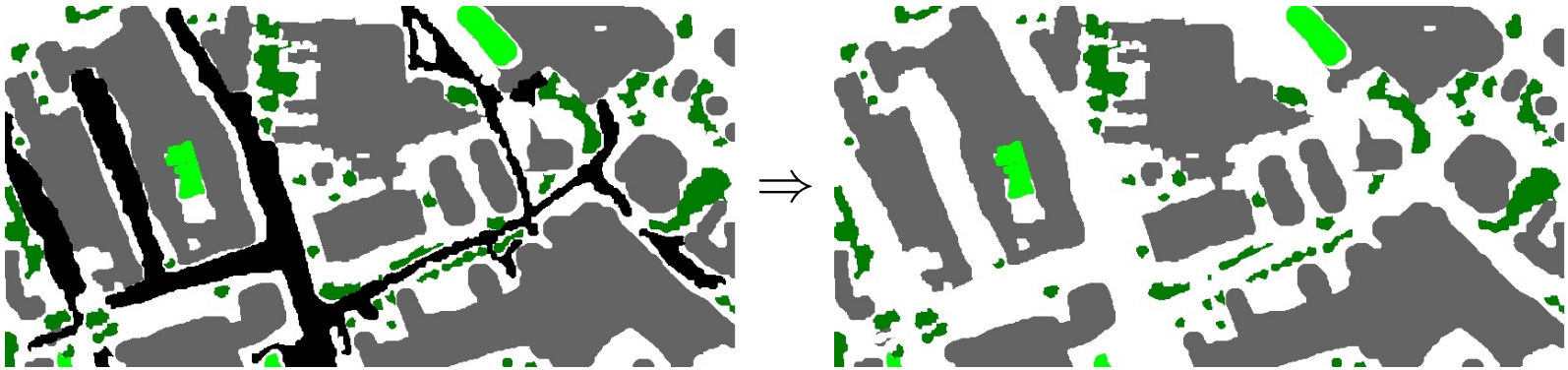
- Standard network architectures
- Leaving out one class during training



Training Setup

CNNs

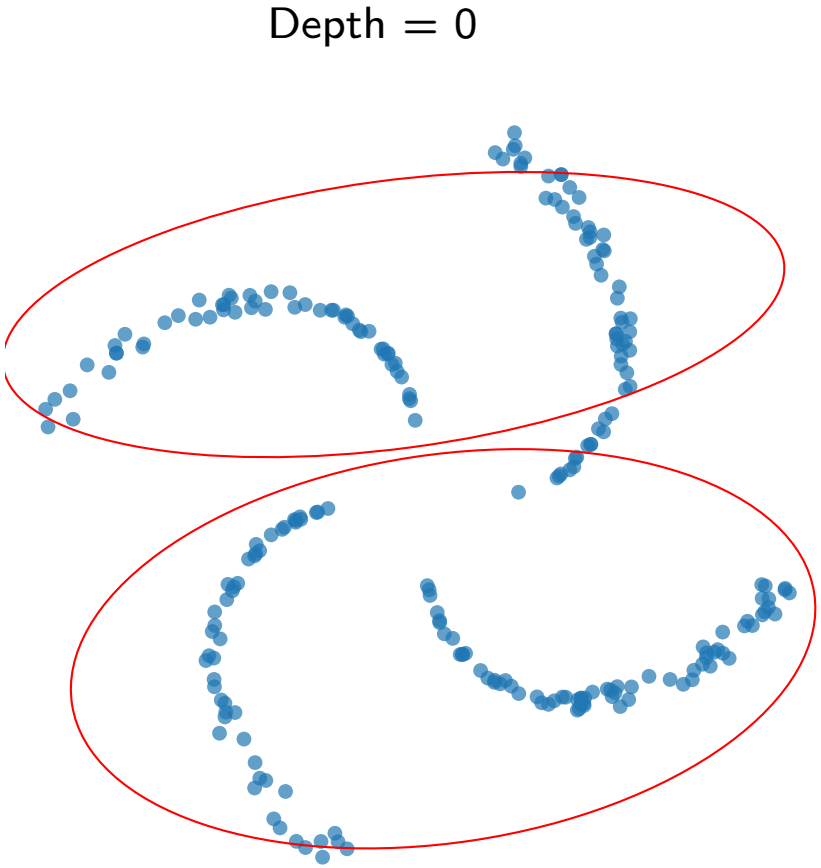
- Standard network architectures
- Leaving out one class during training



Evaluation

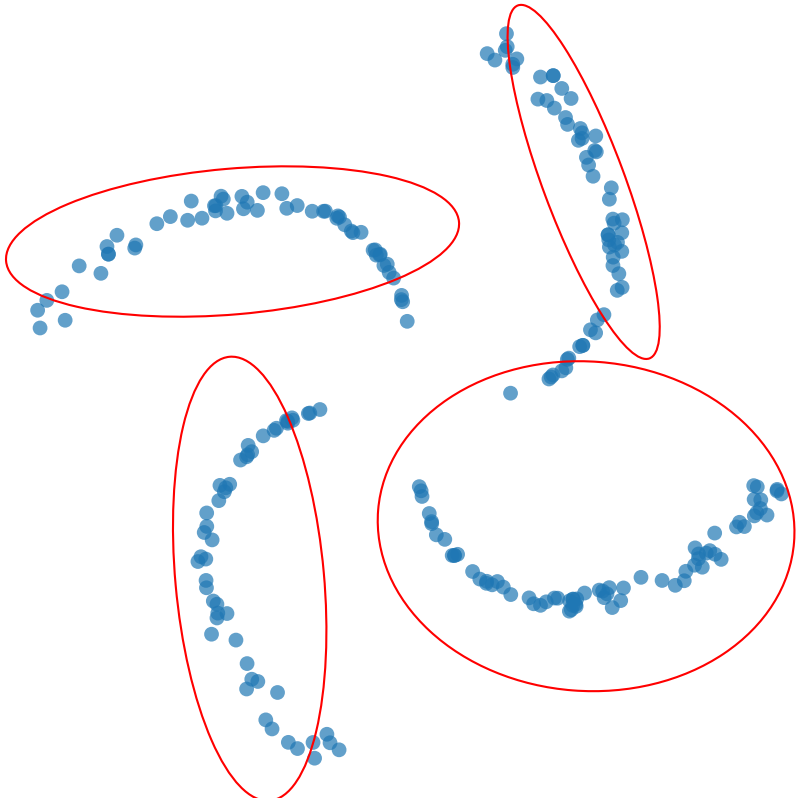
- CNN models with left-out class:
  - Overall Accuracy (OA)
  - Average Accuracy (AA)
- Novelty Detection:
  - Area Under the curve of the Receiver Operating Characteristic (AUROC)
  - Visual quality of results

Results
One Tree

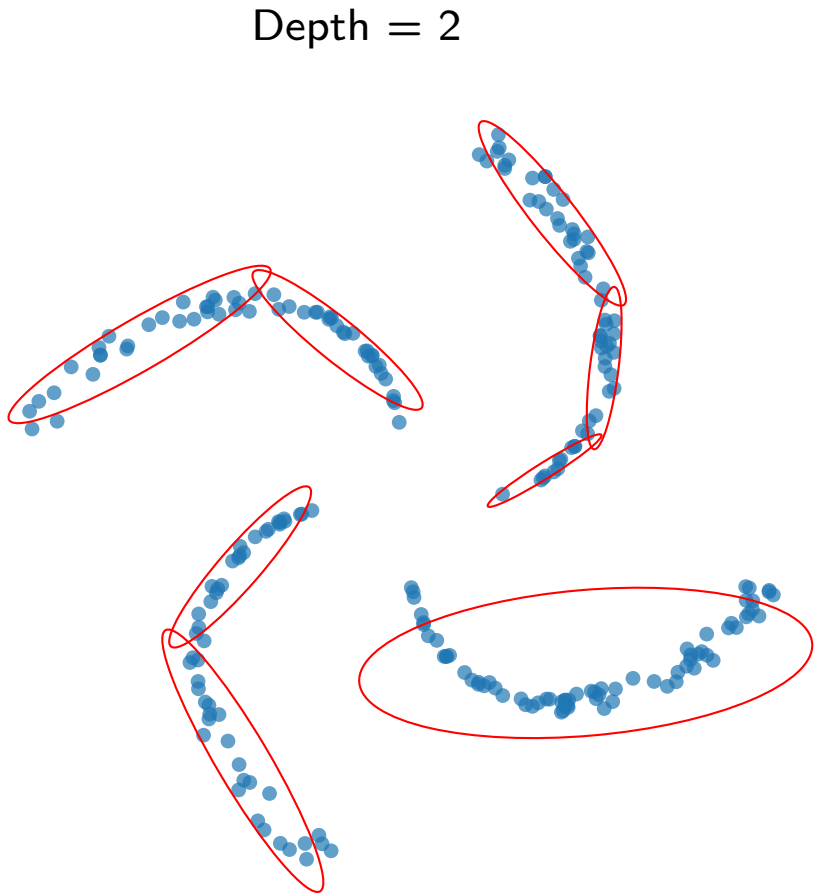


Results
One Tree

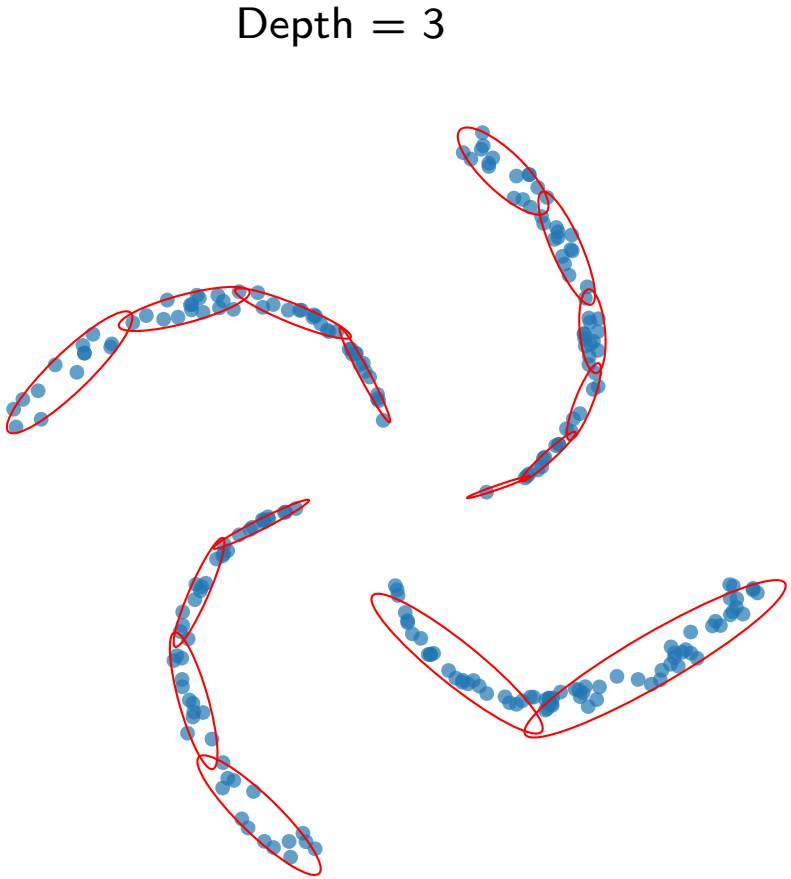
Depth = 1



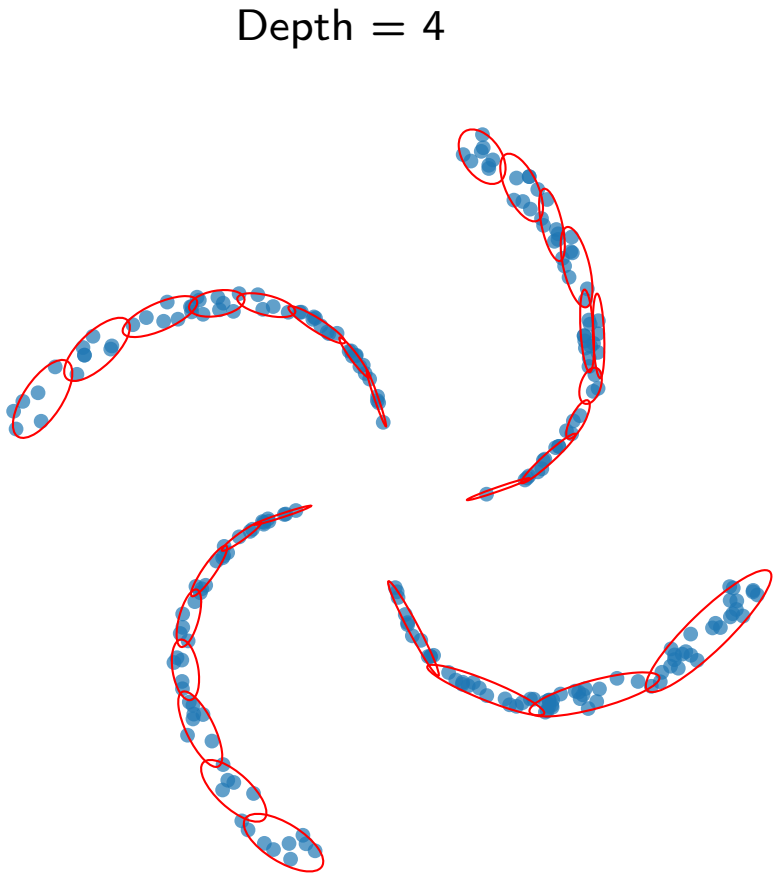
Results
One Tree



Results
One Tree



Results
One Tree



Results
One Tree

Gaussian Probability Density Function (PDF)

Several Density Trees

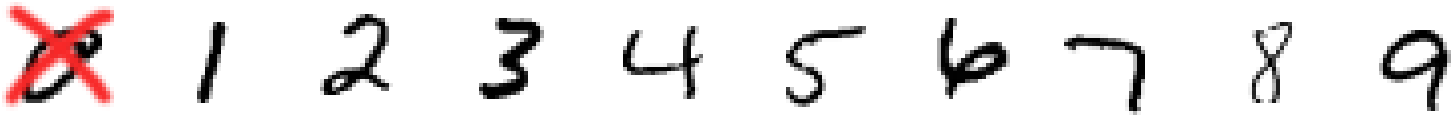
Several Density Trees

CNNs

CNNs trained leaving out one class

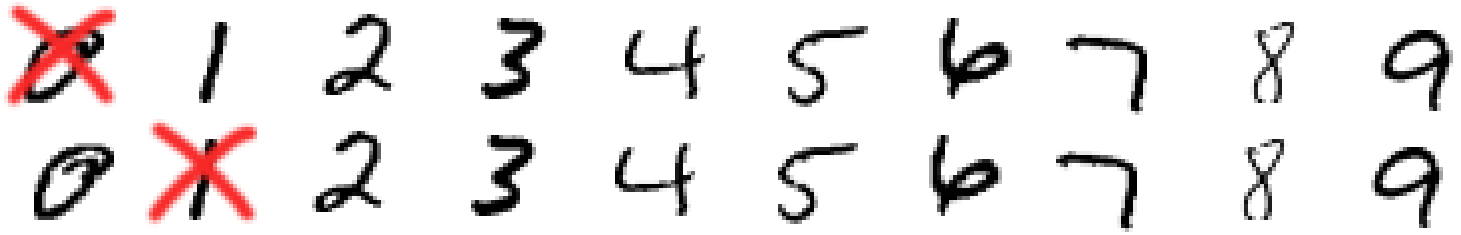
CNNs

CNNs trained leaving out one class



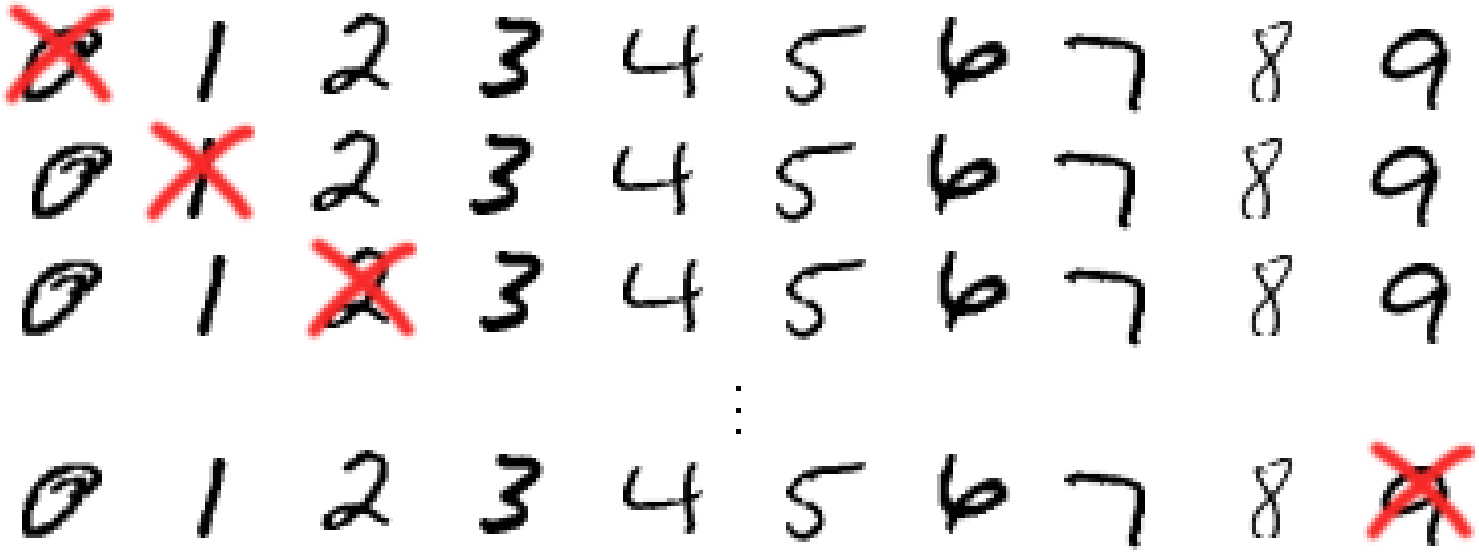
CNNs

CNNs trained leaving out one class



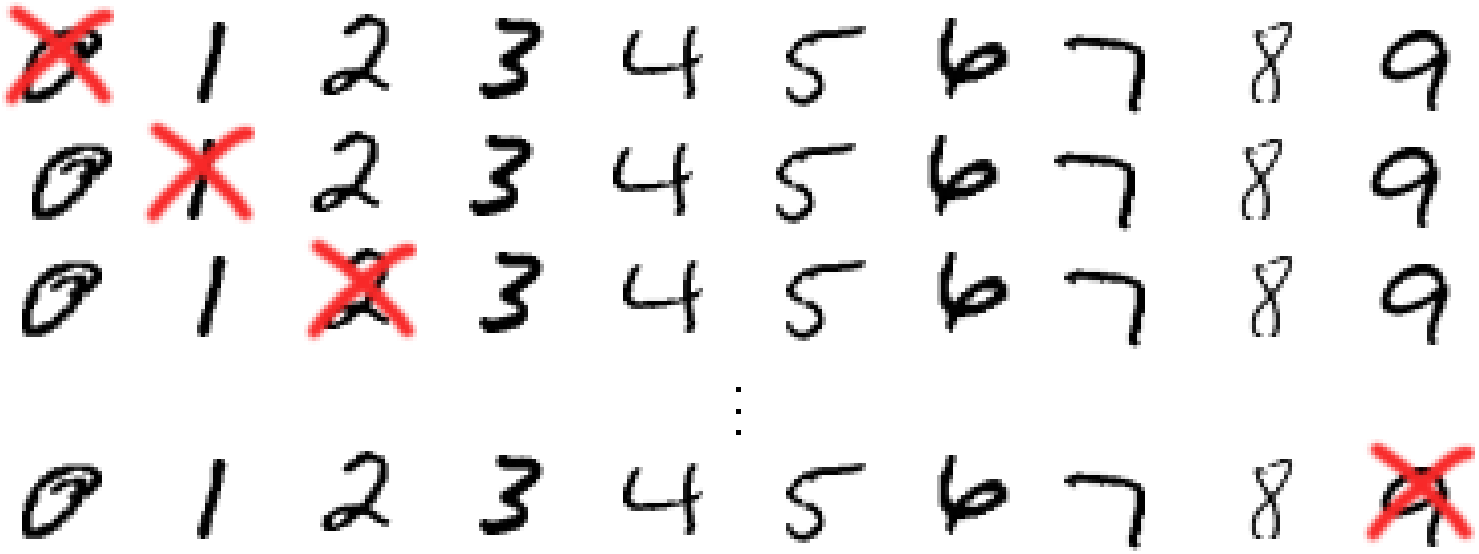
CNNs

CNNs trained leaving out one class



CNNs

CNNs trained leaving out one class



Accuracy:

Training set	Test set
99.34	99.03

Novelty Detection

Softmax-based				Pre-softmax-based		
MSR	Margin	Entropy	MC-Dropout	GMM	OC-SVM	DF
0.97	**0.97**	**0.97**	0.96	0.67	0.75	0.75

Novelty Detection

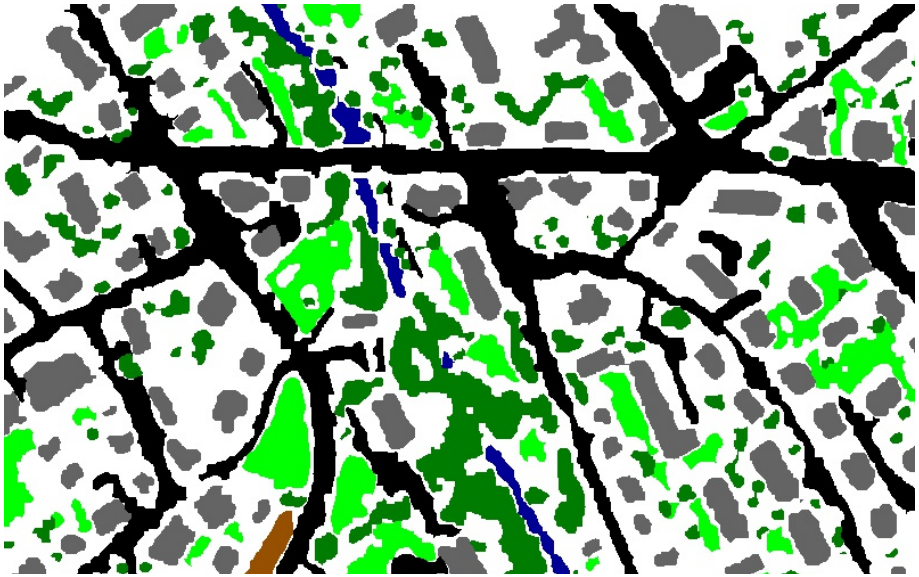
Softmax-based				Pre-softmax-based		
MSR	Margin	Entropy	MC-Dropout	GMM	OC-SVM	DF
0.97	**0.97**	**0.97**	0.96	0.67	0.75	0.75

:/

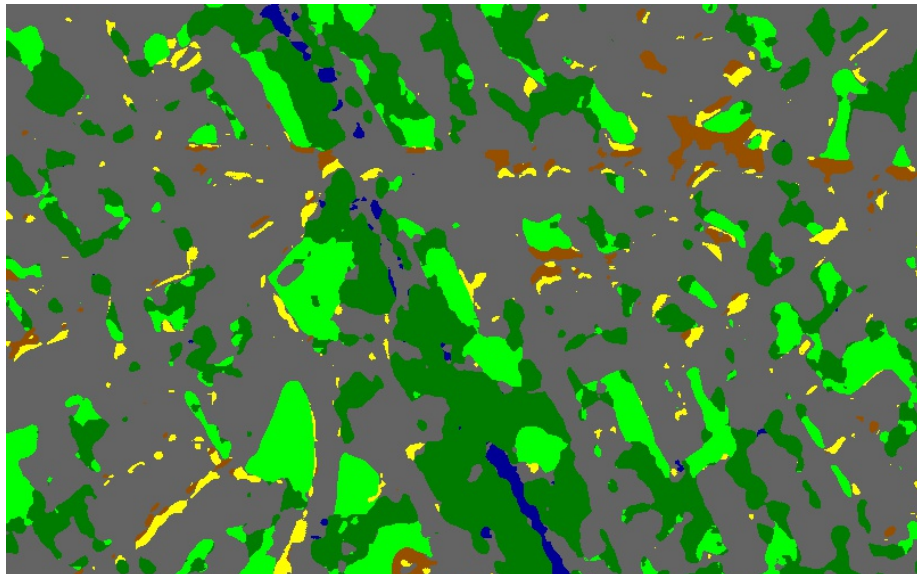
We'll come back to this...

CNN

Ground Truth



Prediction (unseen class Roads)

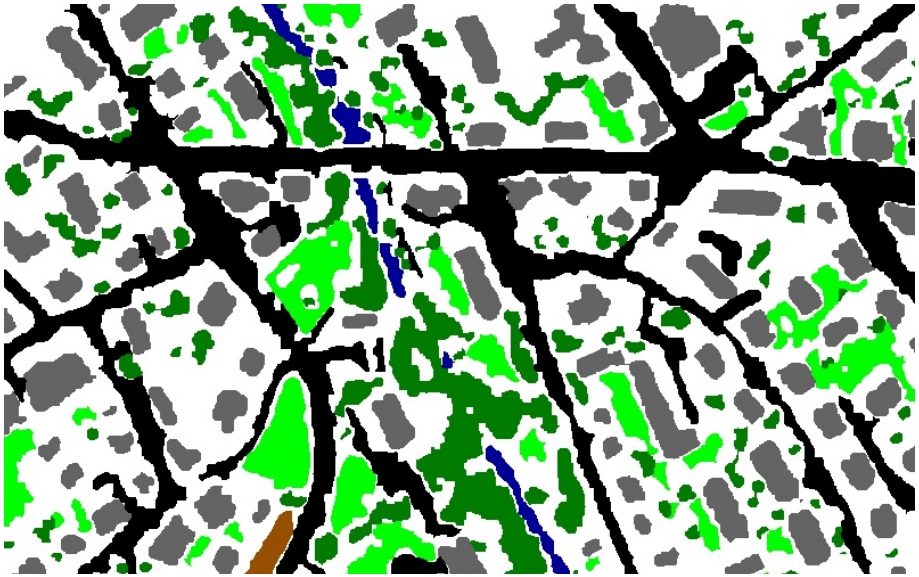


Background Roads Buildings Trees Grass

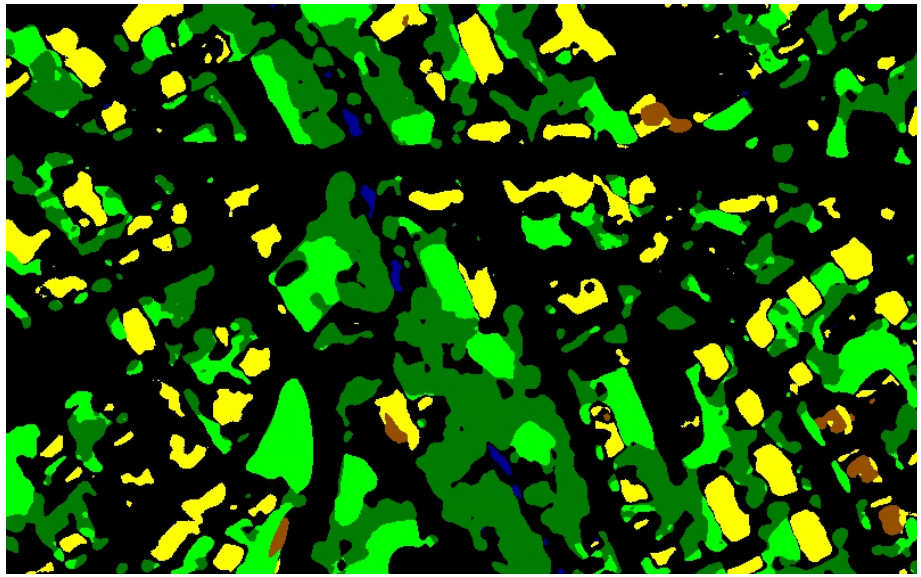
Bare Soil Water Railways Swimming Pools

CNN

Ground Truth



Prediction (unseen class Buildings)

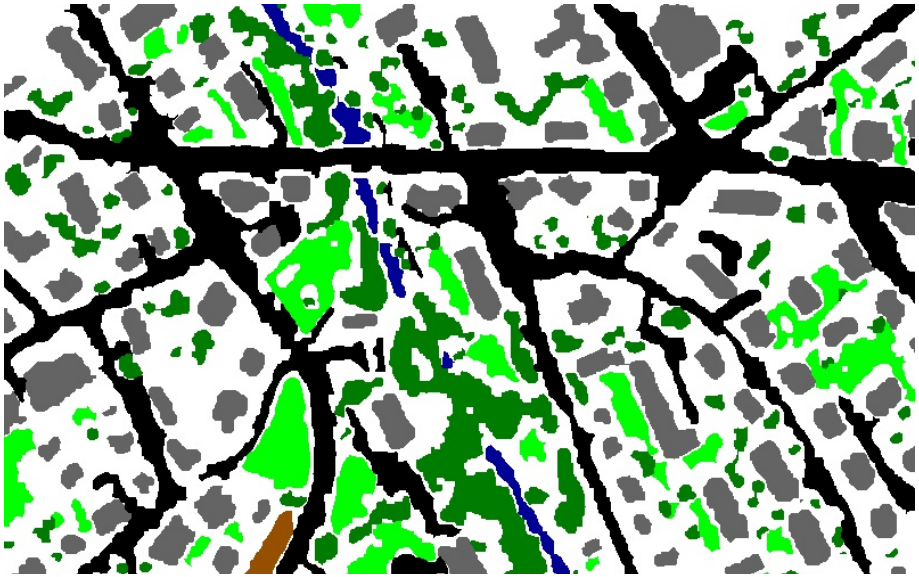


Background Roads Buildings Trees Grass

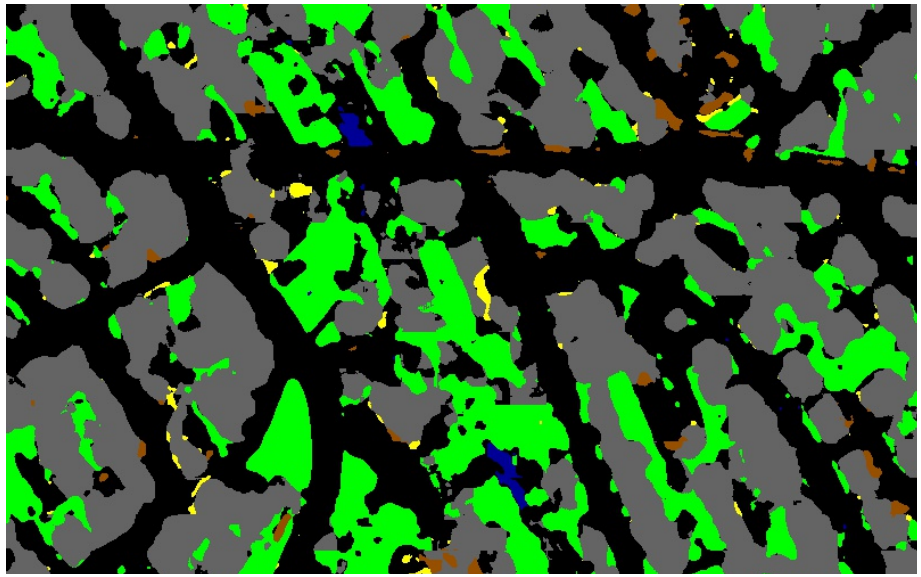
Bare Soil Water Railways Swimming Pools

CNN

Ground Truth



Prediction (unseen class Trees)

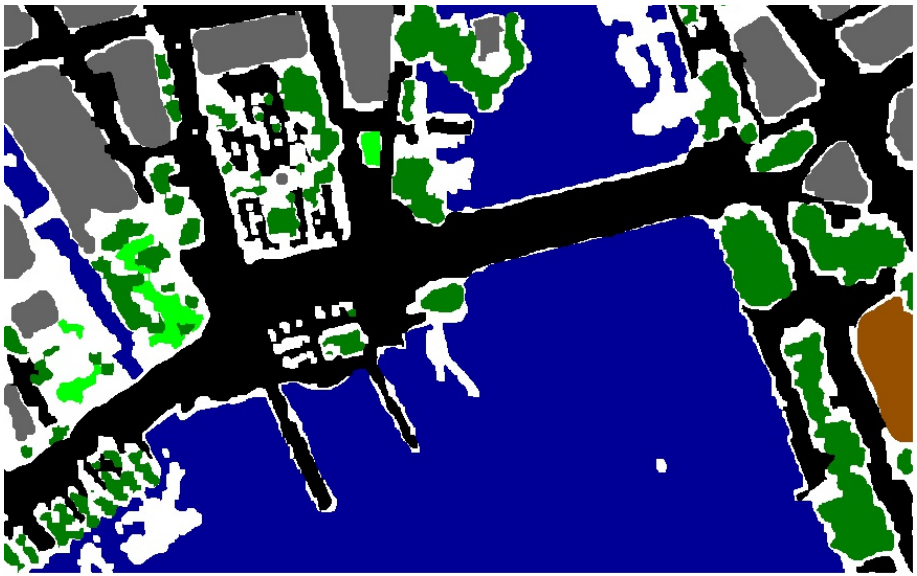


Background Roads Buildings Trees Grass

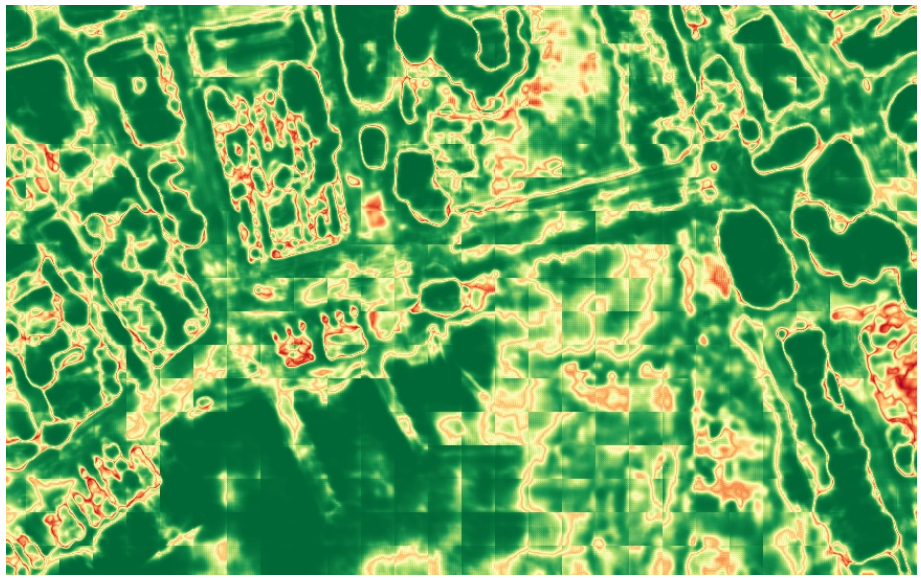
Bare Soil Water Railways Swimming Pools

Novelty Detection
Unseen Class **Water**

Ground Truth



MSR (Unseen Class Water)



Ground Truth

Background Roads Buildings Trees Grass

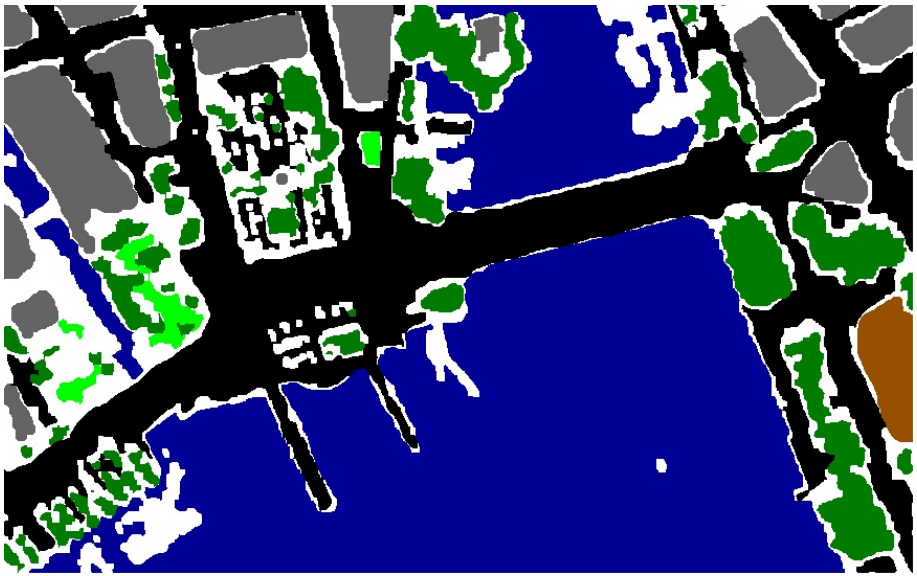
Bare Soil Water Railways Swimming Pools

Confidence

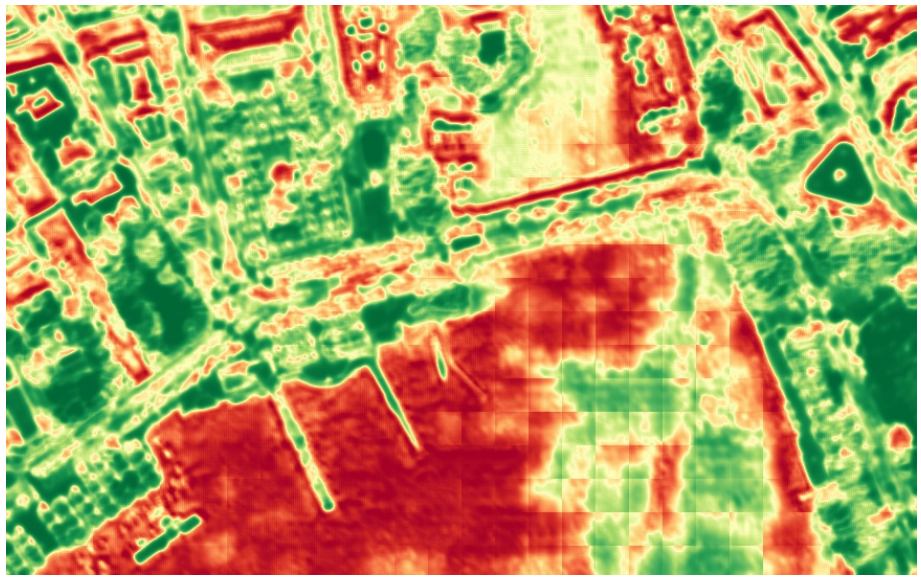
Low High

Novelty Detection
Unseen Class **Water**

Ground Truth



OC-SVM (Unseen Class Water)



Ground Truth

Background Roads Buildings Trees Grass

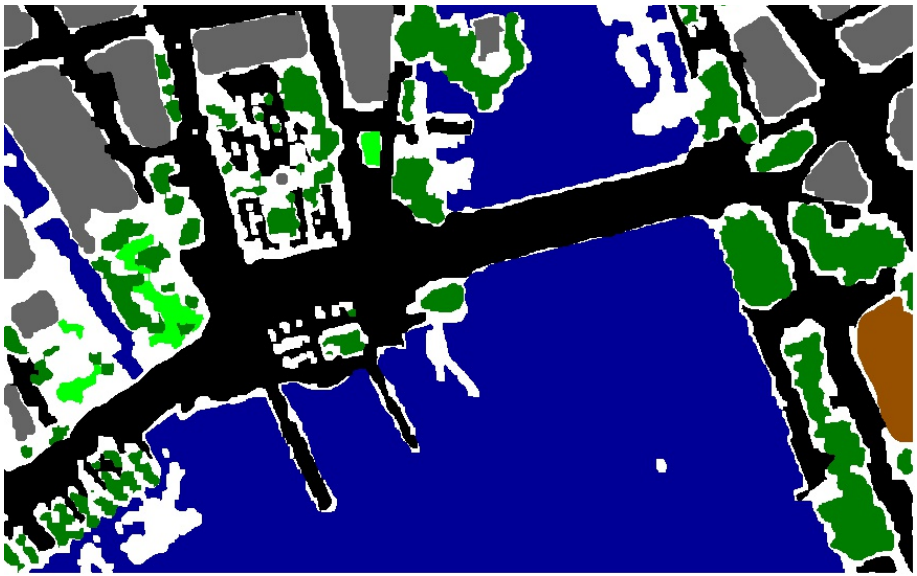
Bare Soil Water Railways Swimming Pools

Confidence

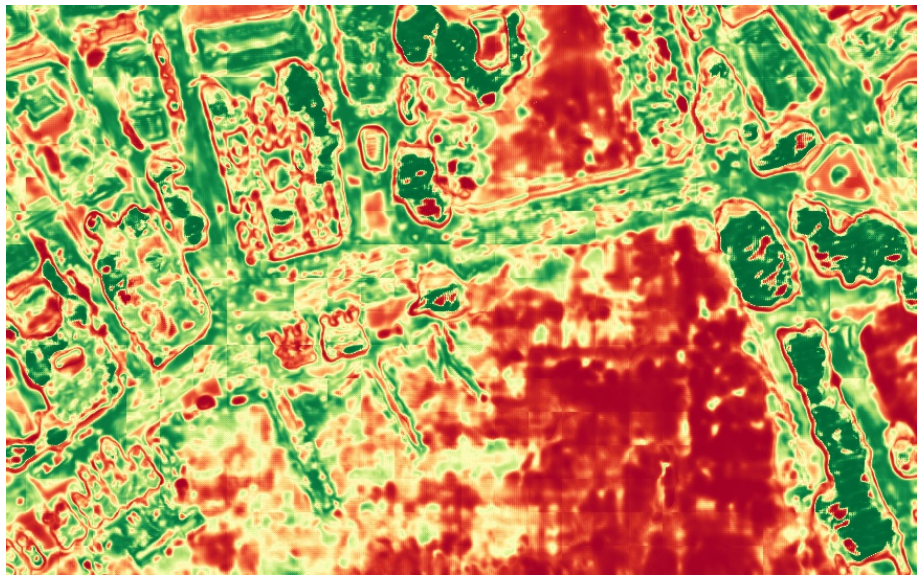
Low High

Novelty Detection
Unseen Class **Water**

Ground Truth



DF (Unseen Class Water)



Ground Truth

Background Roads Buildings Trees Grass

Bare Soil Water Railways Swimming Pools

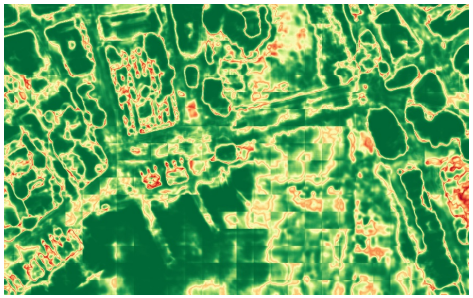
Confidence

Low High

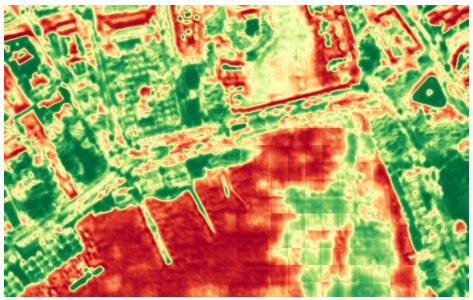
Novelty Detection

Unseen Class **Water**

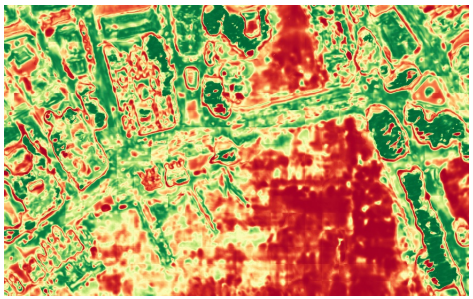
MSR

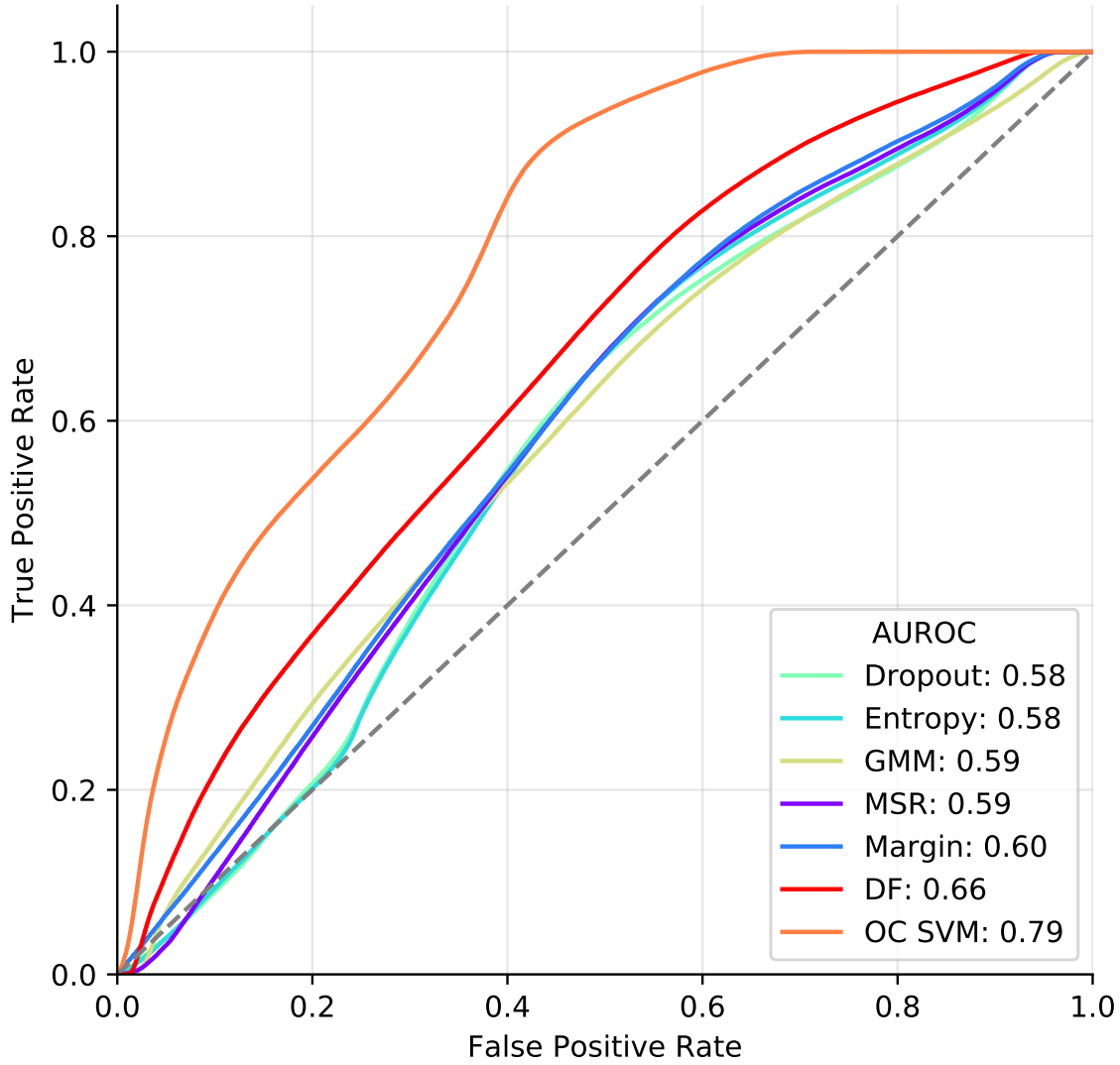


OC-SVM



DF

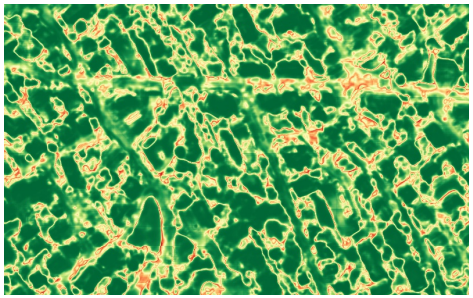




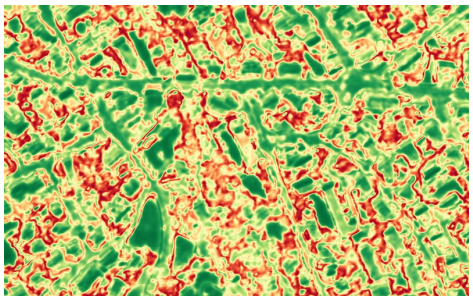
Novelty Detection

Unseen Class **Roads**

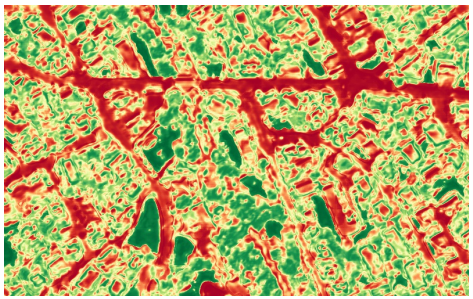
MSR

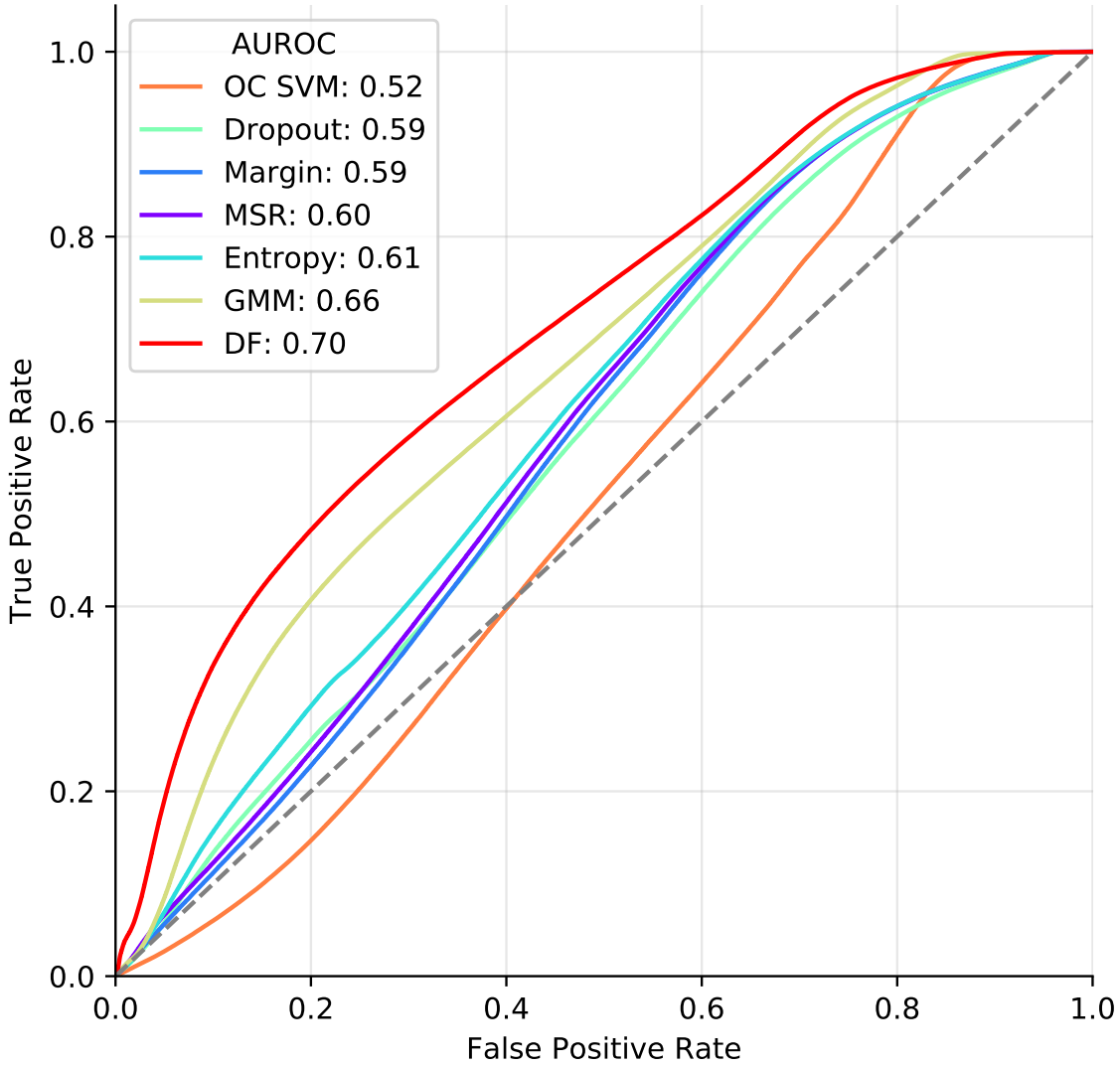


OC-SVM



DF





Novelty Detection
Best Uncertainty Measures

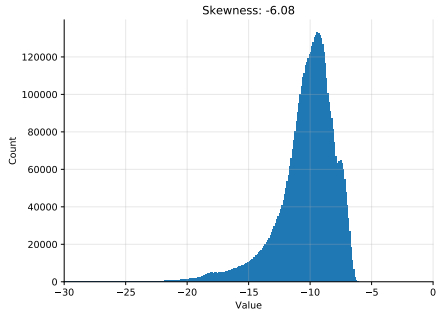
	Softmax-based				Pre-softmax-based		
Left-Out Class	MSR	Margin	Entropy	MC-Dropout	GMM	OC-SVM	DF
Roads							✓
Buildings	✓						
Trees						✓	
Grass							✓
Bare Soil		✓					
Water						✓	
Railways						✓	
Swimming Pools					✓		✓
Average						✓	✓

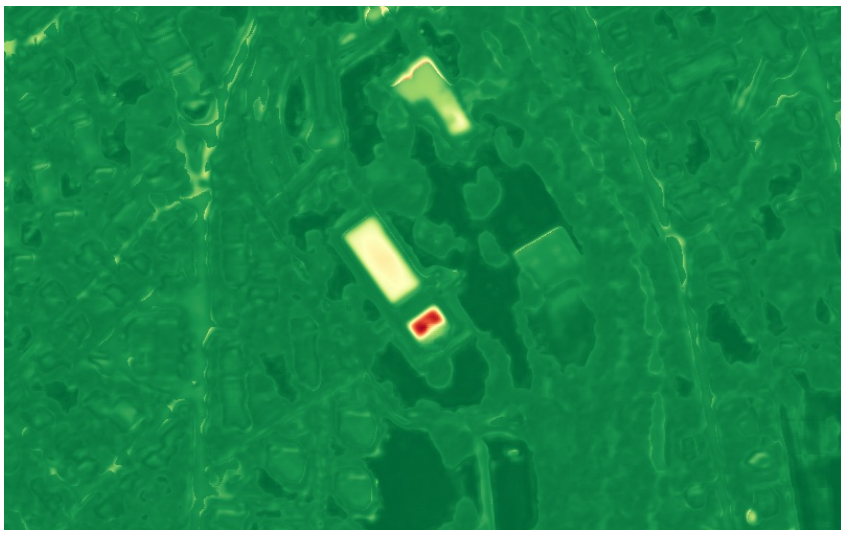
Novelty Detection
Best Uncertainty Measures

	Softmax-based				Pre-softmax-based		
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



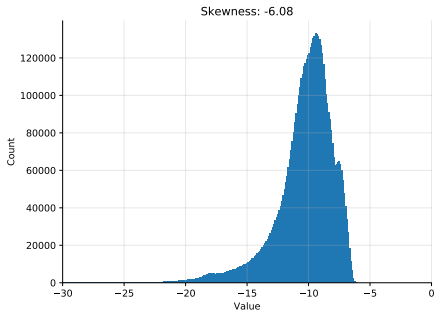


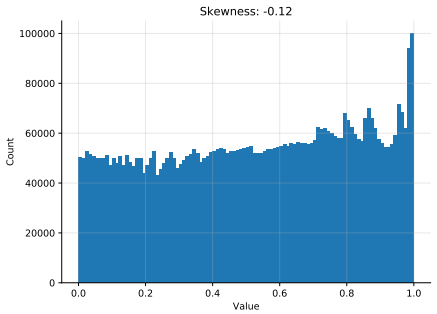
Original

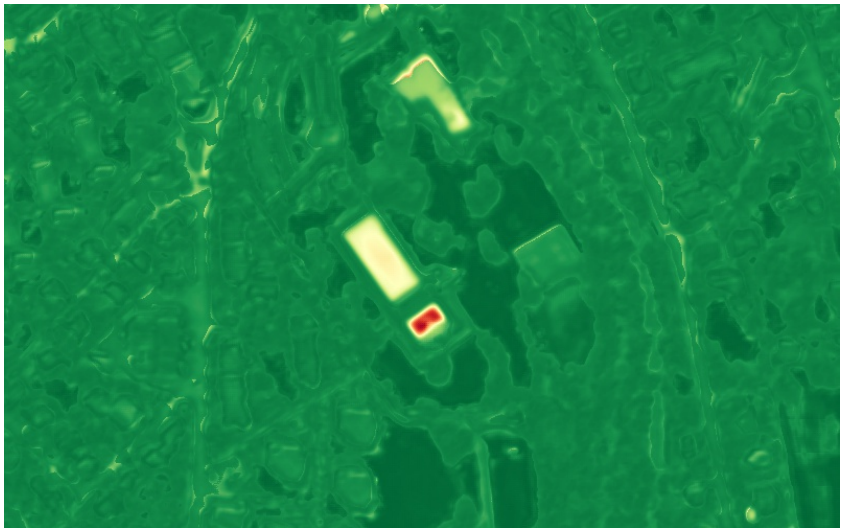
Certainty (DF, left-out class Roads)

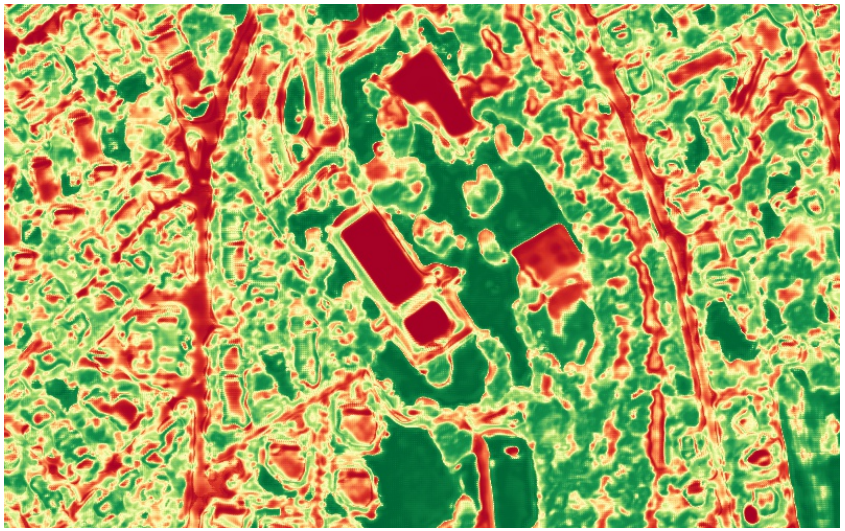
Low  High

Particular Objects
Histogram Stretching









Original

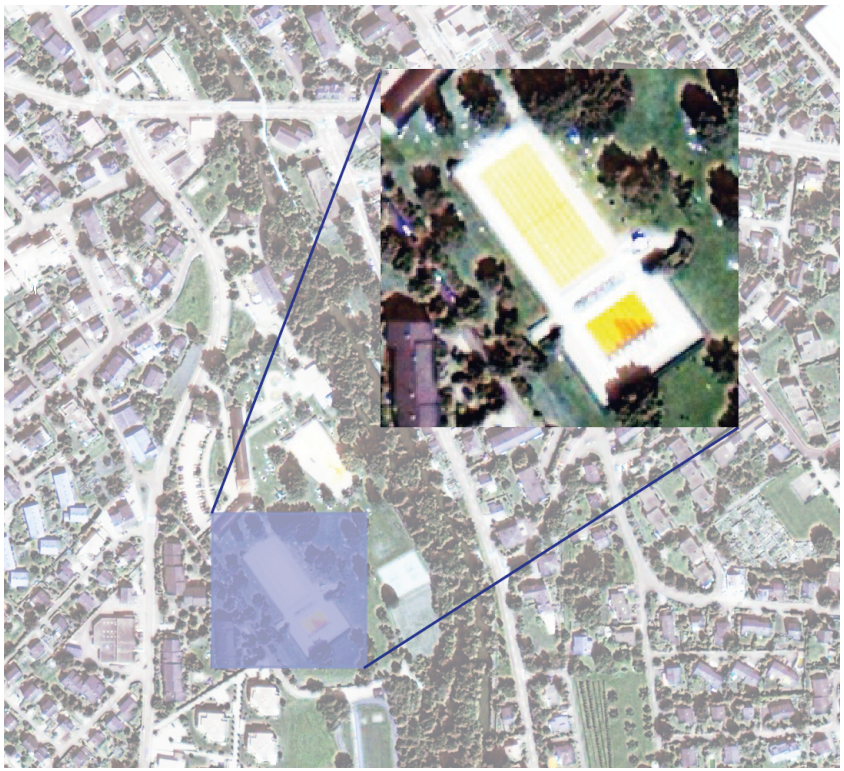
Equalized

Certainty (DF, left-out class Roads)

Low  High

Zurich Dataset
Particular Objects

Image with Region of Interest (ROI)



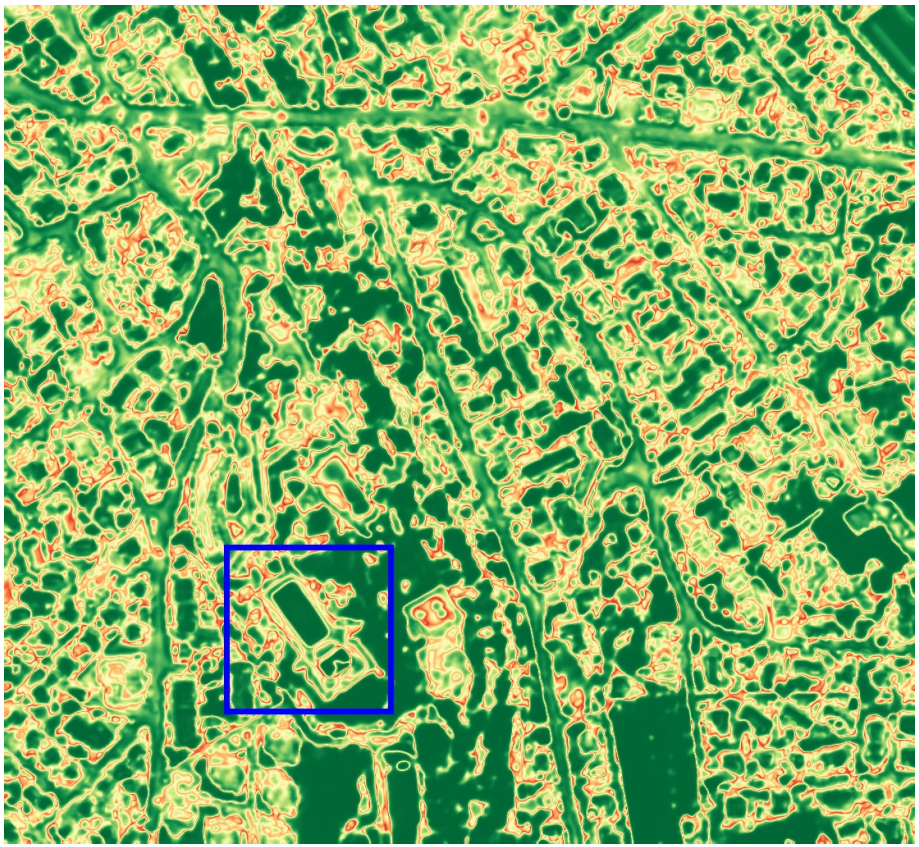
Ground Truth

Background | Roads | Buildings | Trees | Grass

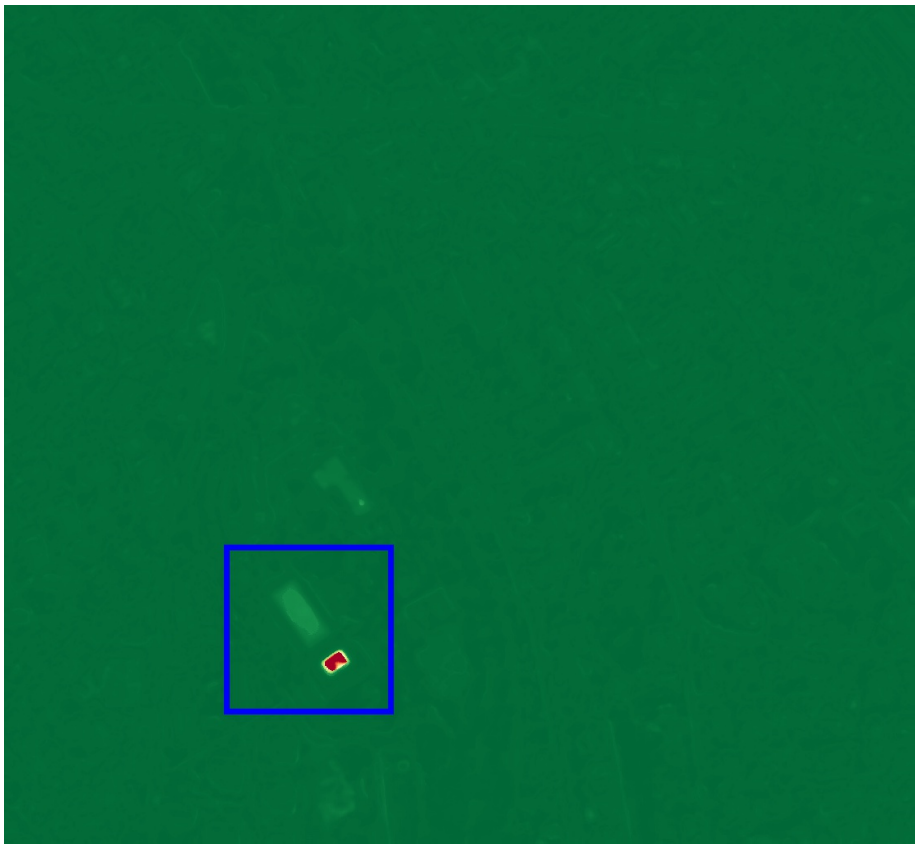
Bare Soil | Water | Railways | Swimming Pools

Zurich Dataset
Particular Objects

MSR



Density Forest (non-equalized)

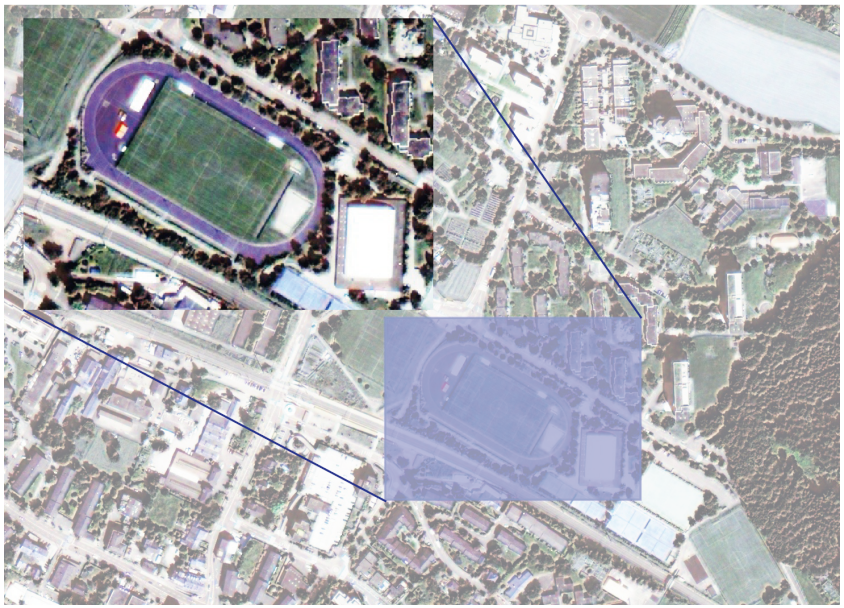


Confidence

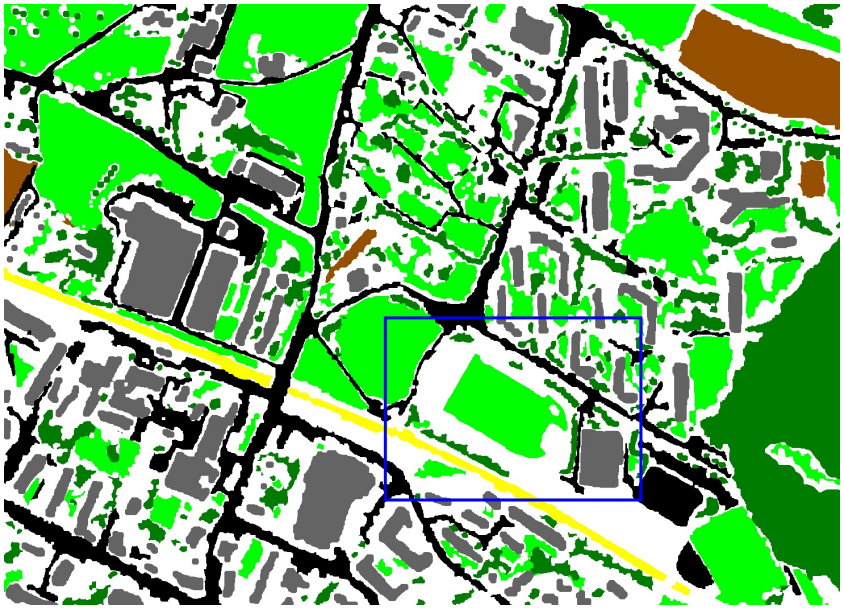
Low  High

Zurich Dataset
Particular Objects

Image with ROI



Ground Truth

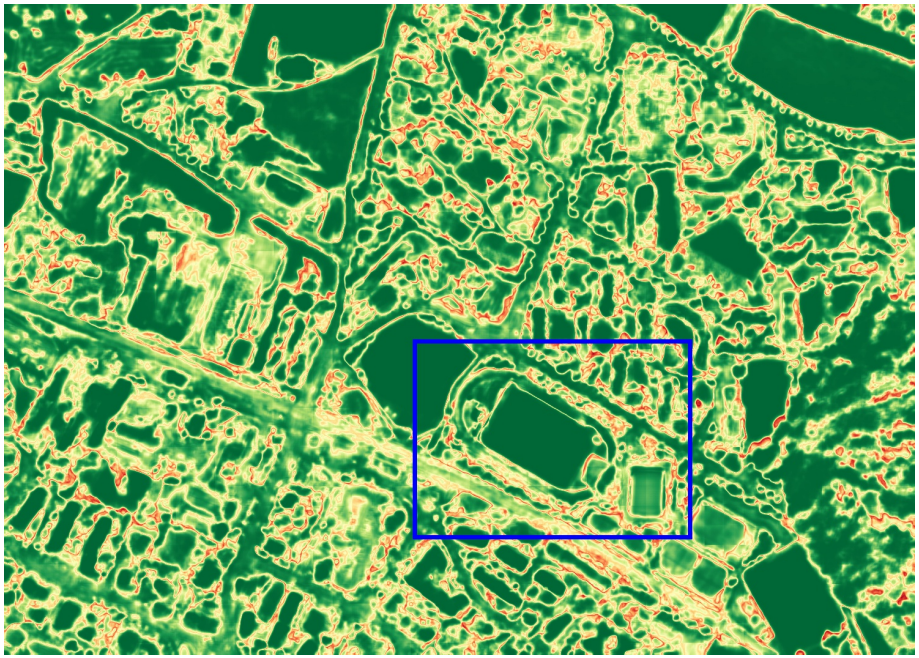


Background | Roads | Buildings | Trees | Grass

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
  - One-Class Support Vector Machine (OC-SVM) and Density Forest work particularly well

Discussion

Winning methods

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

Varying performance

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

Conclusion
Open Questions

Influence of...

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

Conclusion
Open Questions

Influence of...

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

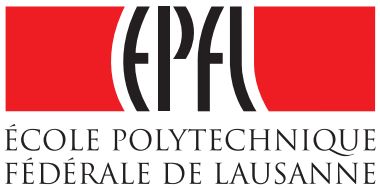
Further applications:

- Change Detection
- Active Learning
- ...

Acknowledgments

Devis Tuia, WUR

Diego Marcos, WUR



François Golay, EPFL

Thank you for your attention!

Thank you for your attention!

Questions?

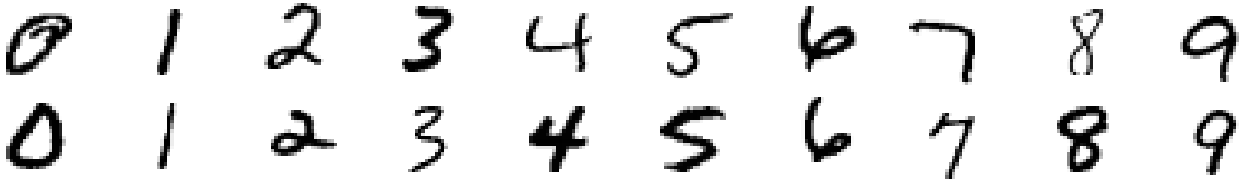
- Y. LeCun, C. Cortes, and C. Burge. *The MNIST Database of Handwritten Digits*. url: `http://yann.lecun.com/exdb/mnist/`.
- M. Volpi and V. Ferrari. "Semantic segmentation of urban scenes by learning local class interactions". In: *2015 IEEE Conference on Computer Vision and Pattern Recognition Workshops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

Appendix

How Does Machine Learning Work?

Training

Dataset:



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

⇓

Training

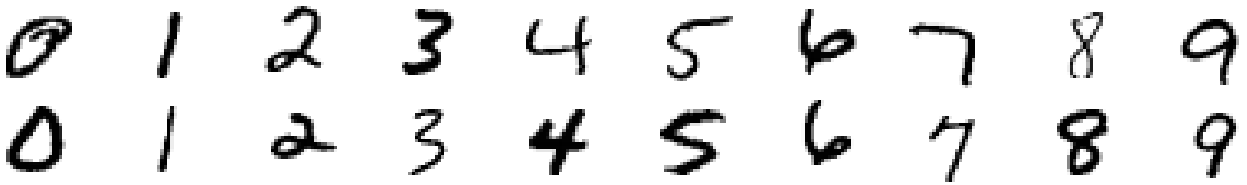
⇓

Model M

How Does Machine Learning Work?

Training

Dataset:



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

⇓

Training

⇓

Model M

Prediction

New image:

⇓

Prediction using model M

⇓

Label: 1

Uncertainty Based on Network Output

Maximum Softmax Response (MSR)

$$C_1(\mathbf{x}) = P^{(c_1)}(\mathbf{x})$$

where $\mathbf{x}$ is a data point and $c_1 = \text{argmax}_{c \in \mathcal{L}} P^{(c)}(\mathbf{x})$.

Margin

$$C(\mathbf{x}) = P^{(c_1)}(\mathbf{x}) - P^{(c_2)}(\mathbf{x})$$

where $c_1 = \text{argmax}_{c \in \mathcal{L}} P^{(c)}(\mathbf{x})$ and $c_2 = \text{argmax}_{c \in \mathcal{L} \setminus c_1} P^{(c)}(\mathbf{x})$.

Entropy

$$C_2(\mathbf{x}) = -H(\mathbf{P}(\mathbf{x})) = -\sum_{c \in \mathcal{L}} P^{(c)}(\mathbf{x}) \log P^{(c)}(\mathbf{x})$$

Uncertainty Based on Network Output

Based on softmax activations

- 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 model

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?

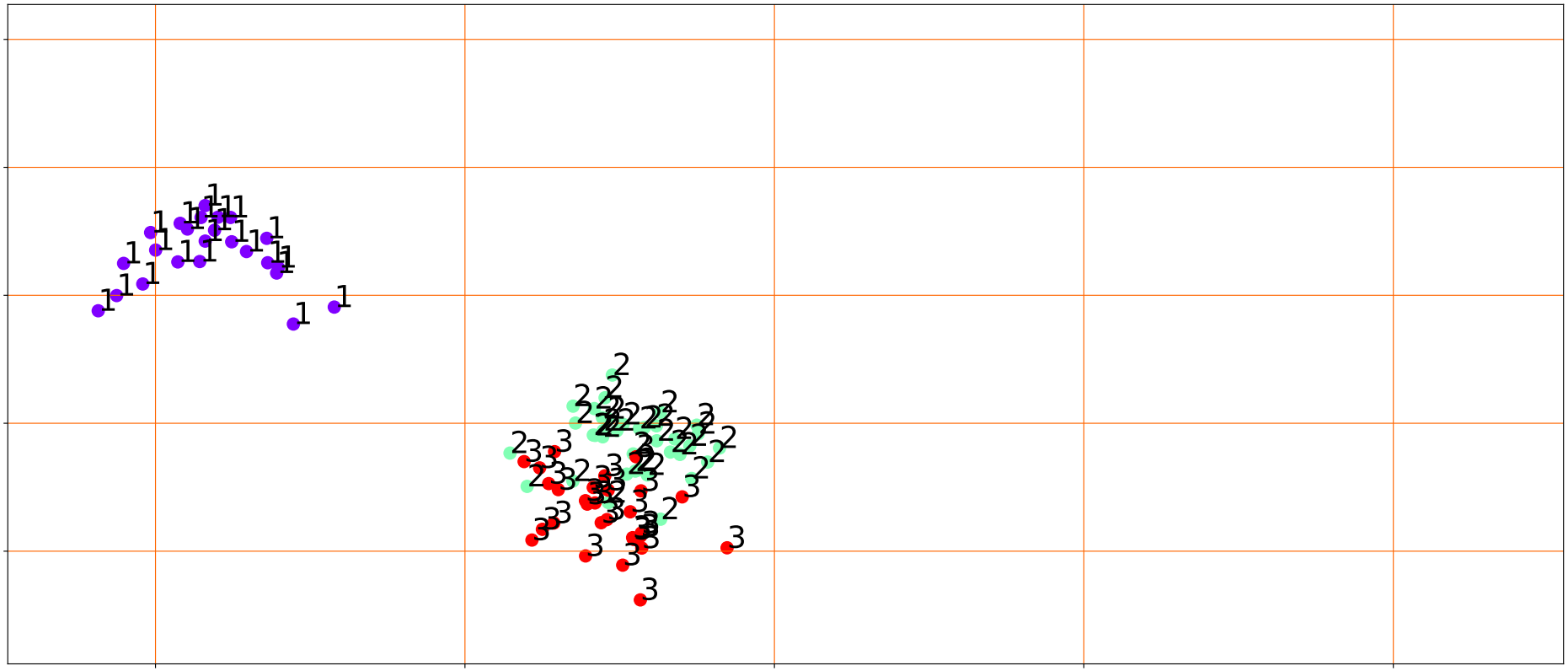
- Train many imperfect models (in parallel)
- Average them

Random Forests

A yoghurt example

Random Forests

A yoghurt example



Random Forests

A yoghurt example

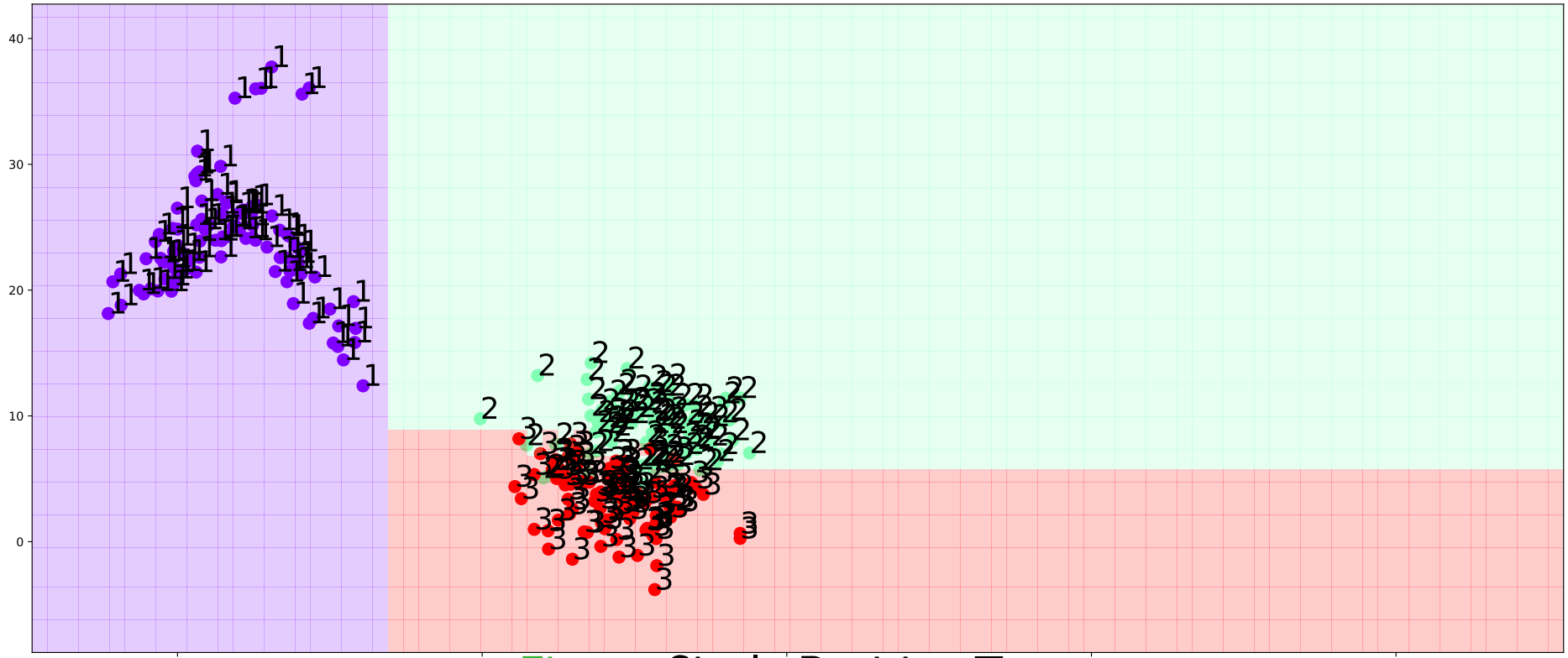


Figure: Single Decision Tree

Random Forests
A yoghurt example

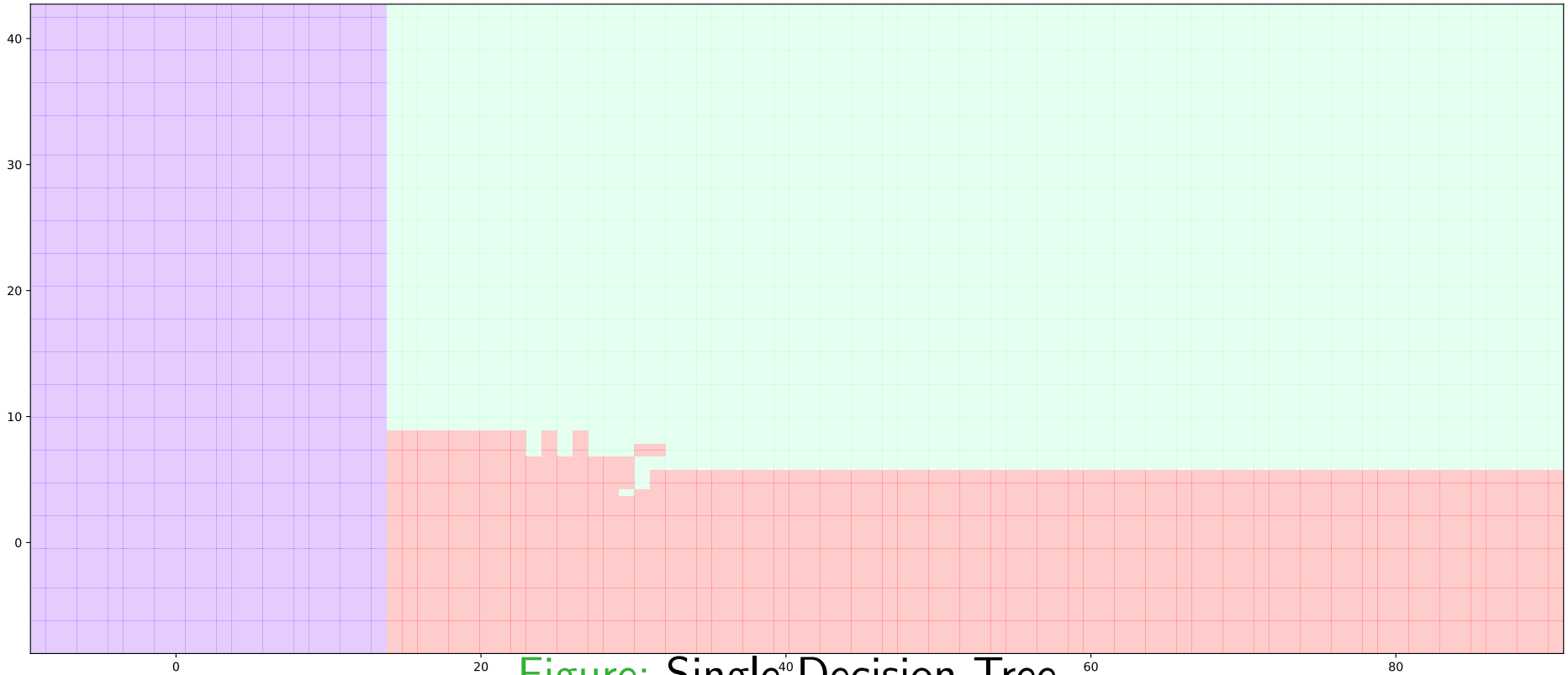


Figure: Single Decision Tree

Random Forests

A yoghurt example

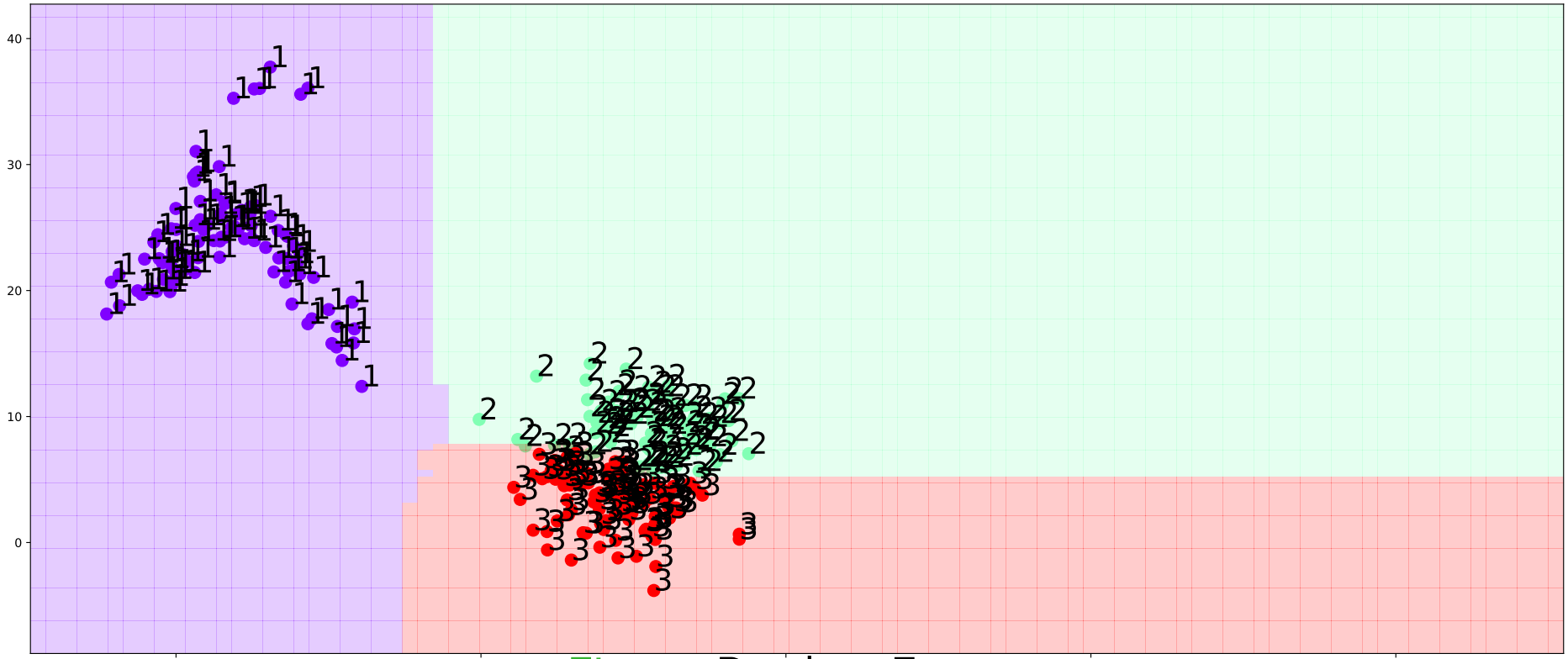


Figure: Random Forest

Random Forests
A yoghurt example

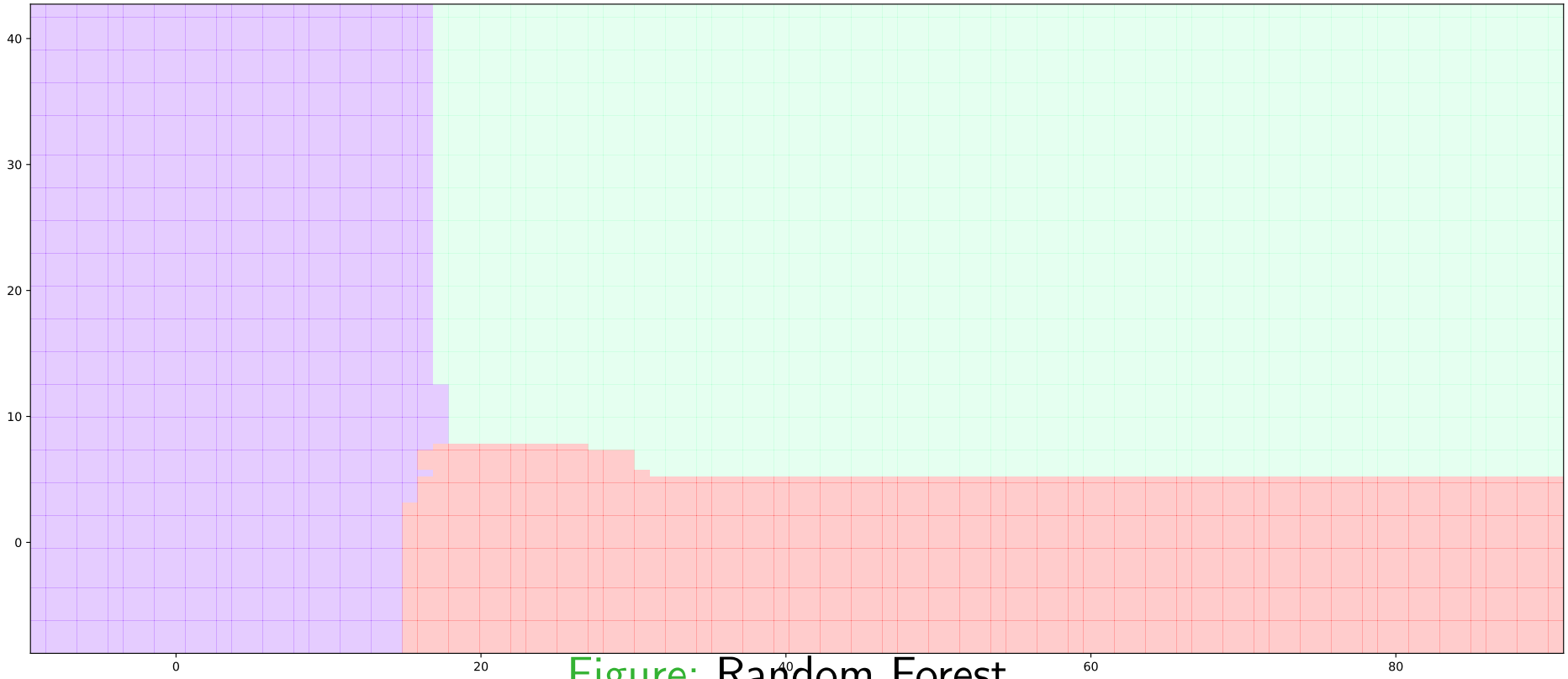
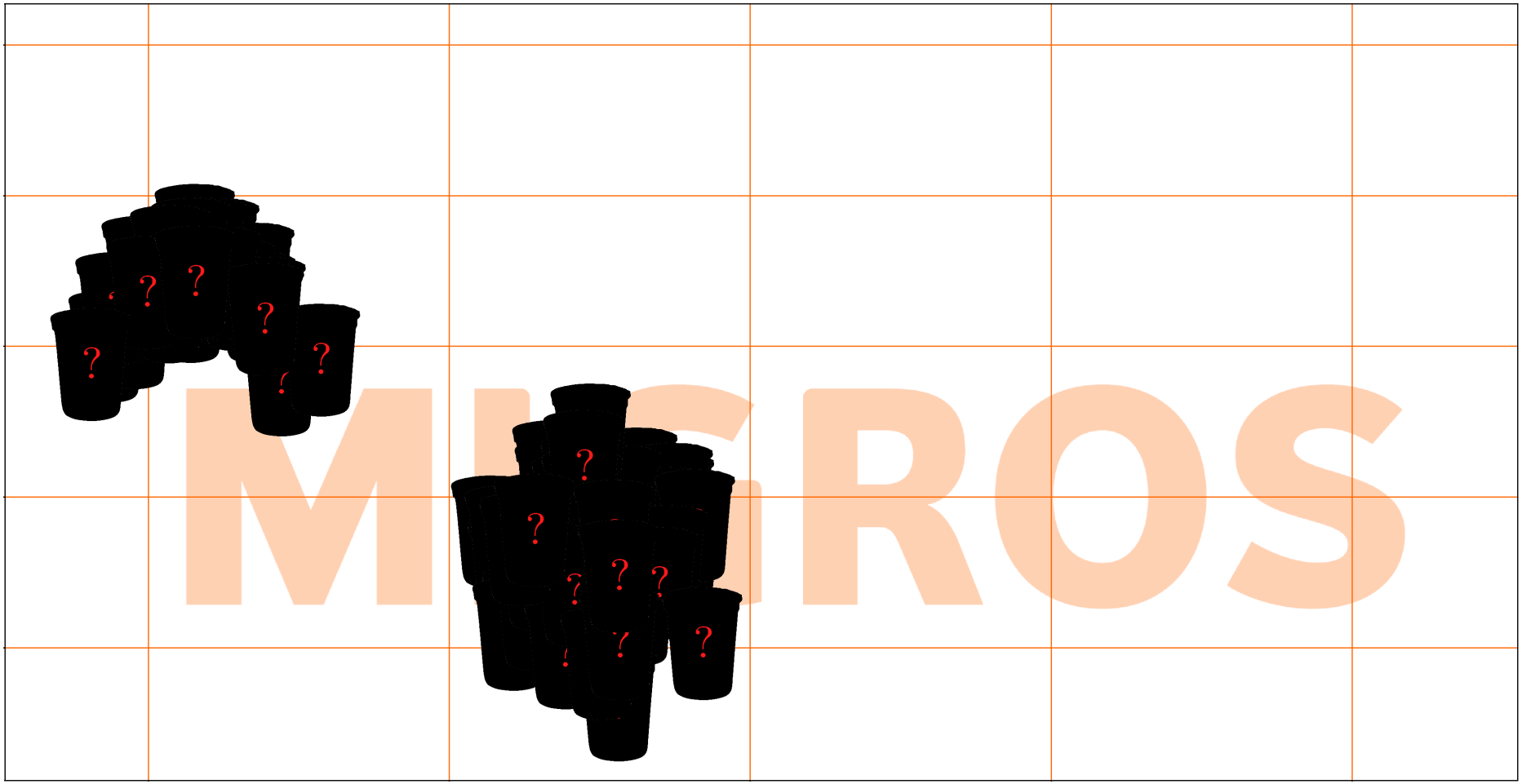


Figure: Random Forest

Density Forests

Question: How to define subspaces of unlabelled data?



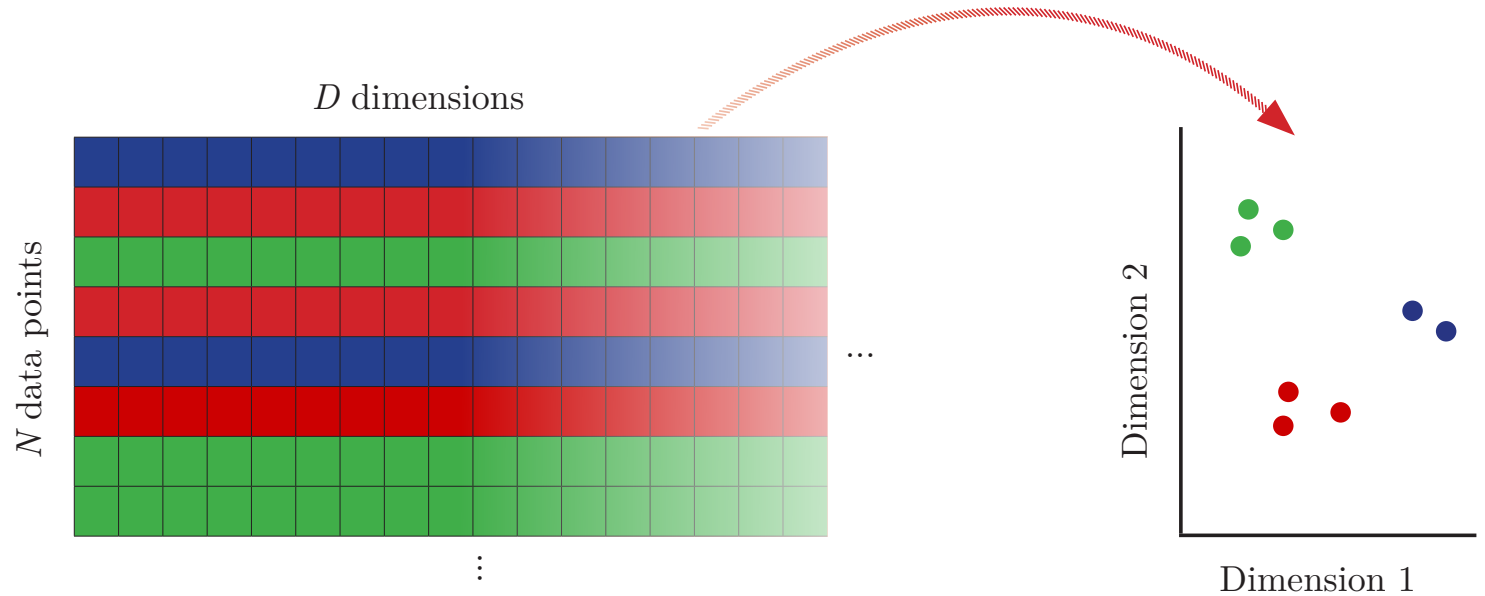
Experimental Setup
CNN

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

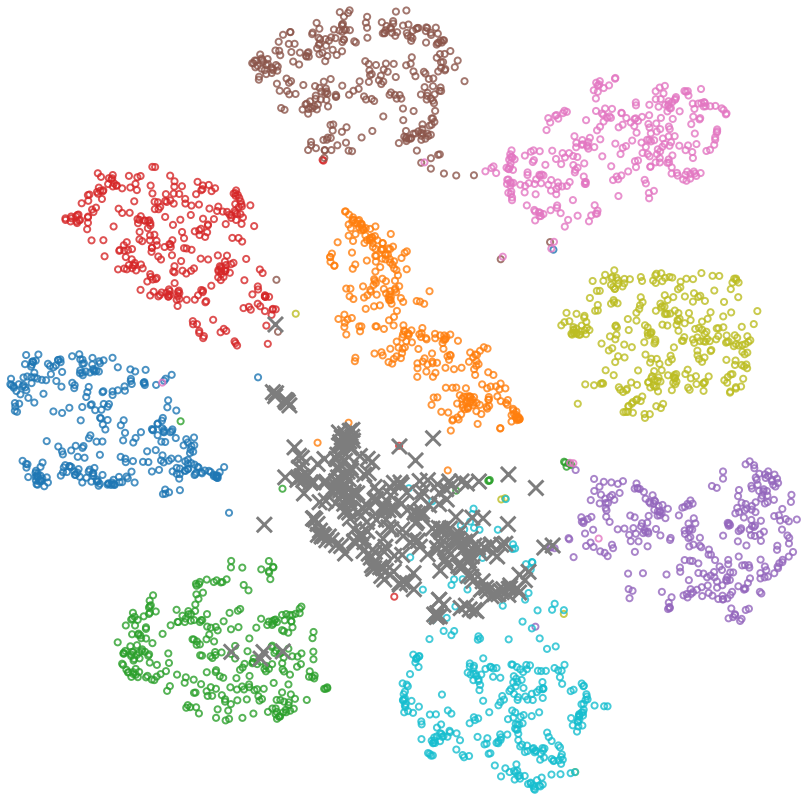
- Model training set activations of the *seen classes*
- Predicting confidence for test set activations, including the *unseen class*

Experimental Setup
Dimensionality Reduction

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

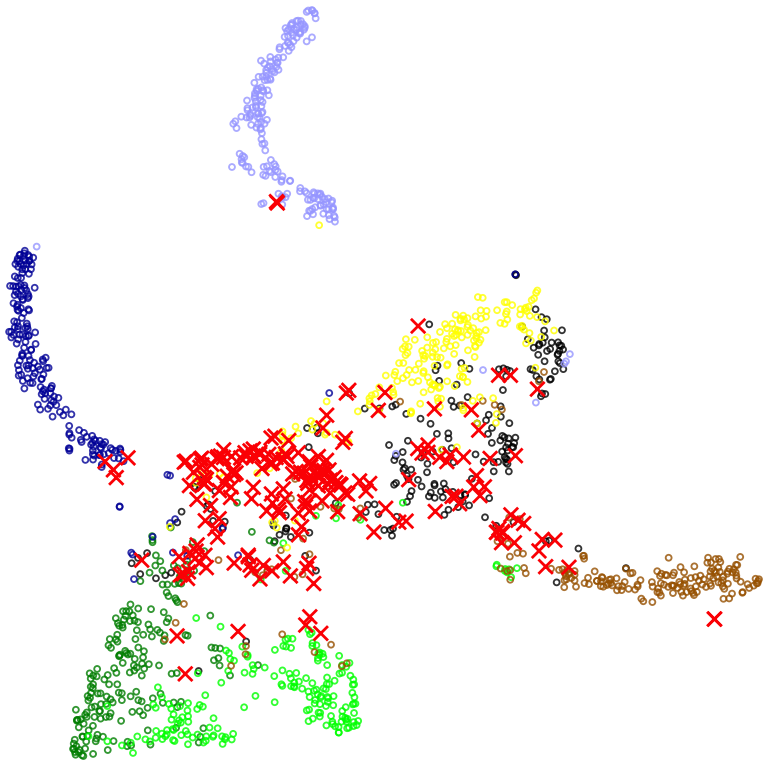


t-SNE

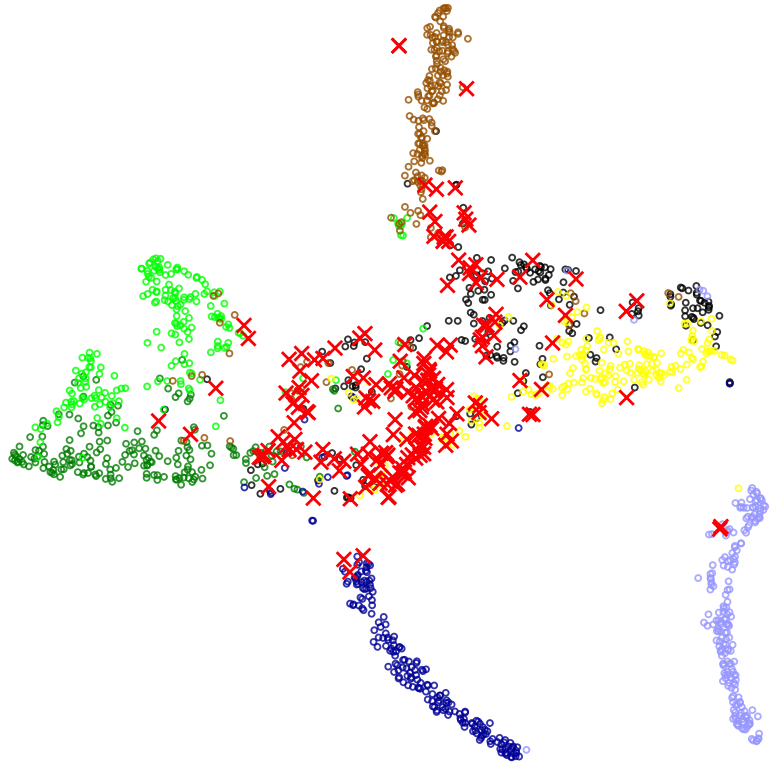


Novelty Detection
Are the Activations Separable?

t-SNE before PCA



t-SNE after PCA



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