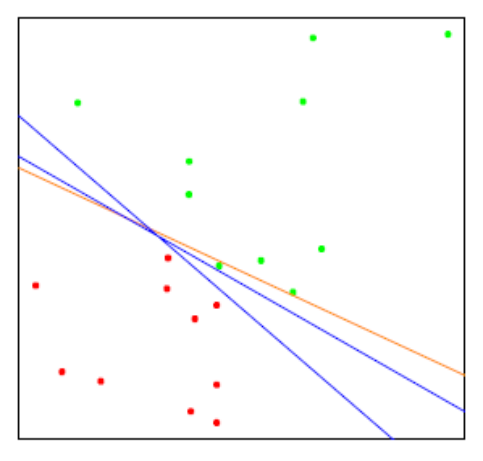


Parallel & Scalable Machine Learning

Introduction to Machine Learning Algorithms

Dr. Gabriele Cavallaro

Postdoctoral Researcher

High Productivity Data Processing Group

Juelich Supercomputing Centre

LECTURE 8

Data Preparation and Performance Evaluation

February 26th, 2019

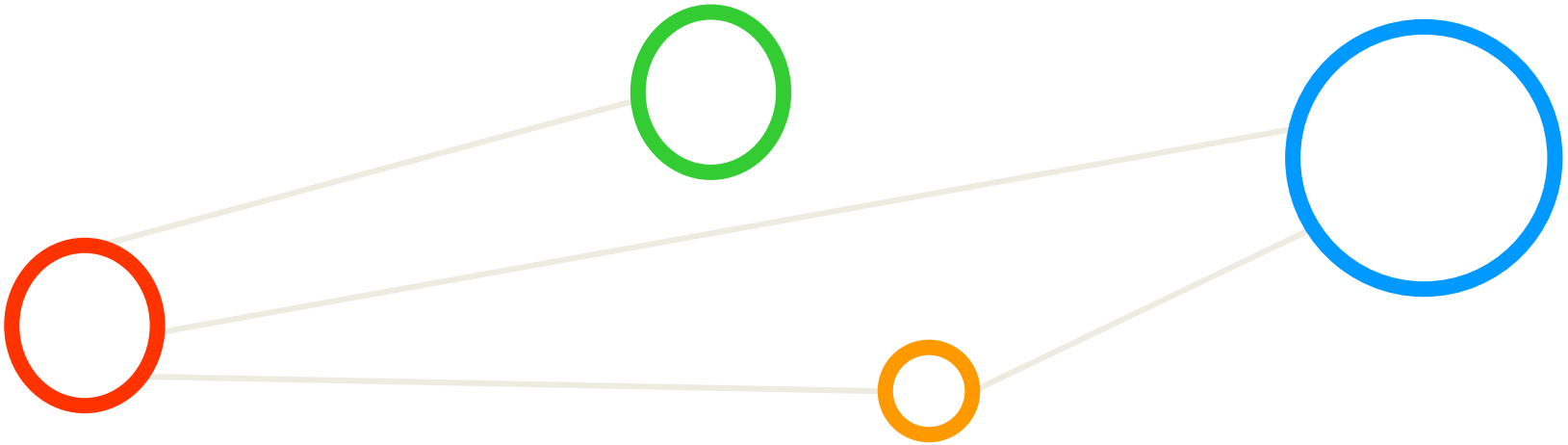
JSC, Germany

UNIVERSITY OF ICELAND

SCHOOL OF ENGINEERING AND NATURAL SCIENCES

FACULTY OF INDUSTRIAL ENGINEERING, MECHANICAL ENGINEERING AND COMPUTER SCIENCE

Outline



Outline of the Course

1. Parallel & Scalable Machine Learning driven by HPC
2. Introduction to Machine Learning Fundamentals
3. Introduction to Machine Learning Fundamentals
4. Feed Forward Neural Networks
5. Feed Forward Neural Networks
6. Validation and Regularization
7. Validation and Regularization
8. Data Preparation and Performance Evaluation
9. Data Preparation and Performance Evaluation
10. Theory of Generalization
11. Unsupervised Clustering and Applications
12. Unsupervised Clustering and Applications
13. Deep Learning Introduction

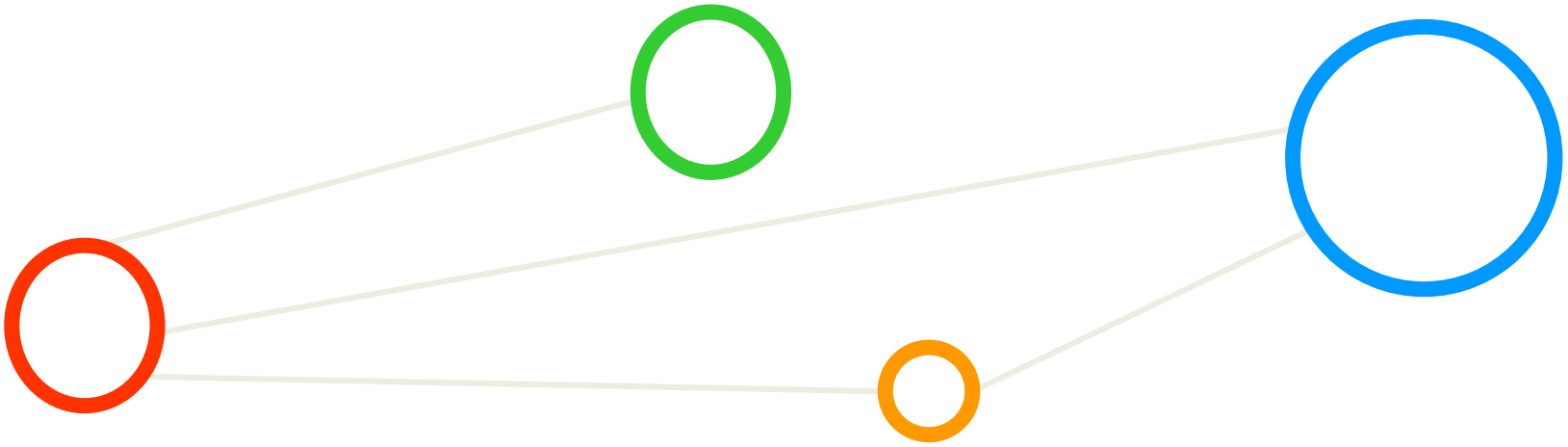
Theoretical Lectures

Practical Lectures

Outline

- Introduction
  - Classical Design Loop for Machine Learning
  - Data Collection, Features, Models, Training and Performances
- Data Pre-Processing
  - Mean Subtraction and Normalization
  - Feature Extraction and Hughes Phenomenon
  - Principal Component Analysis
- Remote Sensing
  - Classification of Satellite Images
  - Hyperspectral images
  - Spectral and Spatial Information
- Learning with Limited Training Data
  - Challenges of Generalization
  - Data Augmentation
- Performance Evaluation
  - Training, Validation and Test Sets
  - Confusion Matrix and Metrics

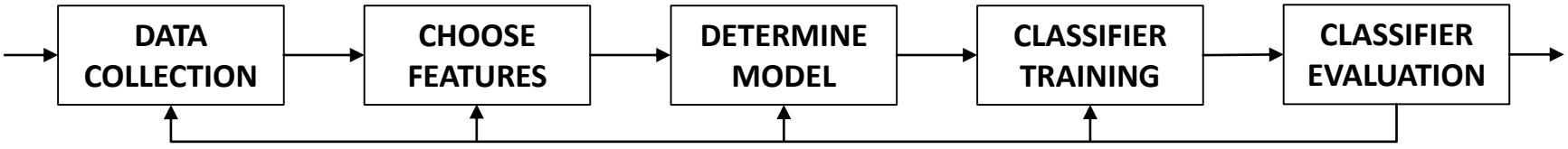
Introduction



Classical Design Loop for Machine Learning

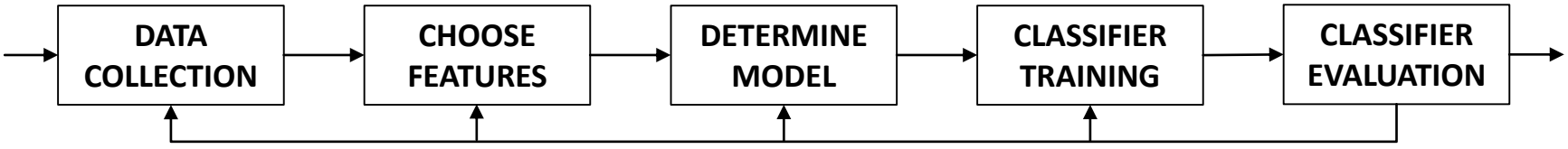
- **Data collection**
  - How to estimate when the amount of data collected is adequately large and representative (i.e., set of samples to train and test the classifier?

- **Features choice**
  - Depends on the properties of the problem domain
  - Features should be:
    - Simple to extract
    - Insensitive to noise
    - Discriminative of patterns within different classes



Classical Design Loop for Machine Learning

- **Model choice**
  - Objective: map between low level features and high level information
  - Many different approaches for modeling/parametrizing this mapping
    - The choice of the method is not always rational
  - Need to take into account the memory required, scalability of training data, ease of implementation and hyper parameter tuning
  - A good approach is to test several models

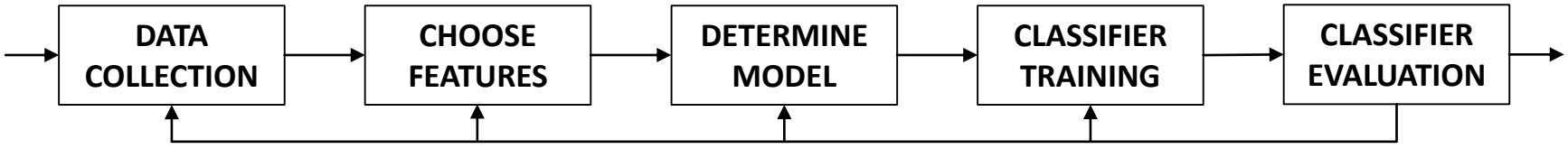


Classical Design Loop for Machine Learning

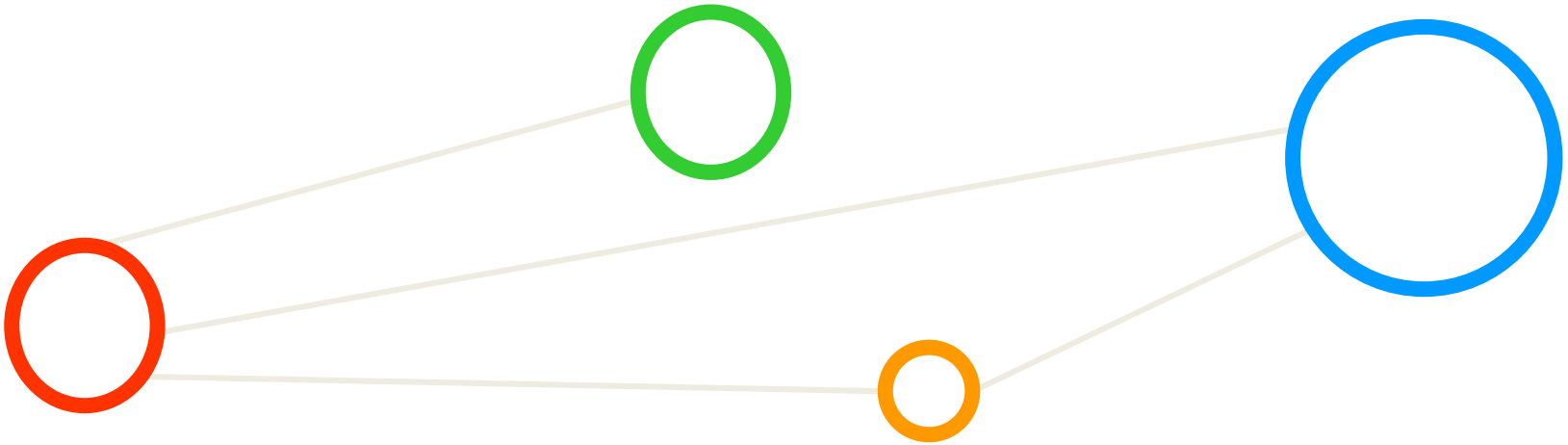
- **Training**
  - Training set must be representative
  - Importance of cross validation / separate datasets

- **Evaluation**
  - Measure the error rate (or performance)
  - Test different set of features/models to compare performance

- **Computational Complexity**
  - What is the trade off between computational ease and performance?
  - How an algorithm scales as a function of the number of features or categories?



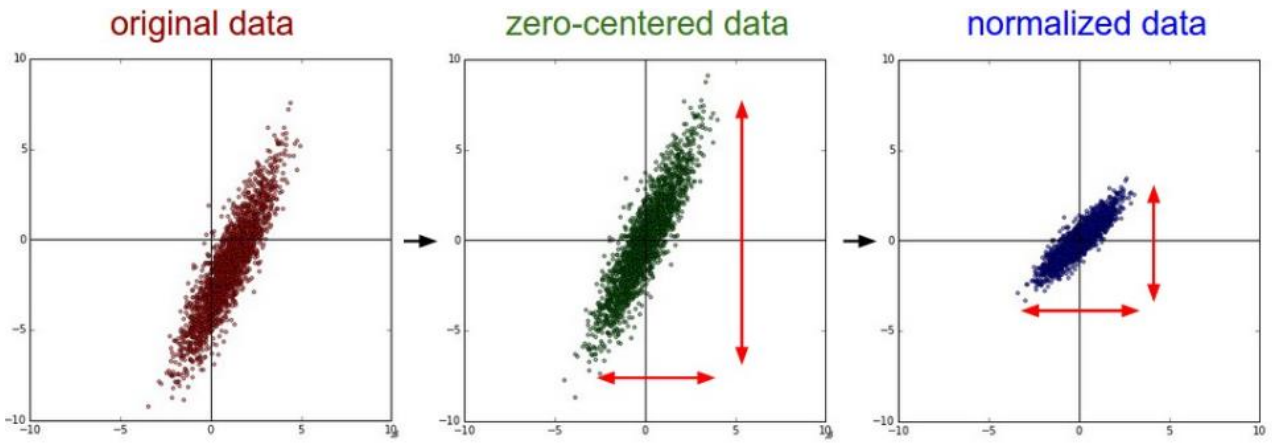
Pre-Processing



Setting up the Data

- **Mean Subtraction**
  - Subtract the mean across every individual feature in the data
  - Geometric interpretation: center the data cloud around the origin of every dimension
  - *With images:* subtract a single value from all pixels (separately across the channels)

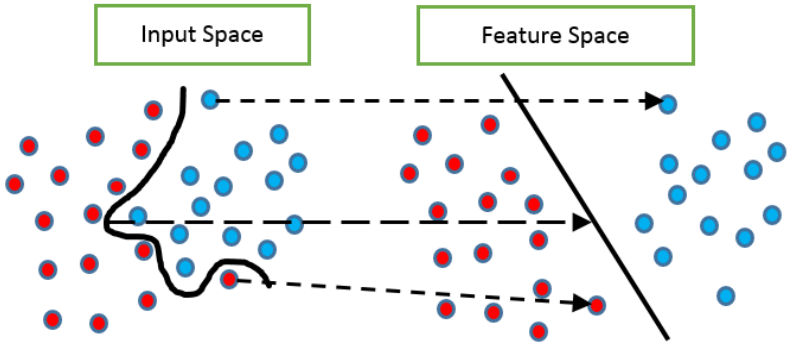
- **Normalization**
  - Normalize the data dimensions so that they are of approximately the same scale
  - One way: divide each dimension by its standard deviation once it has been zero-centered
  - Another way: scale the min and max along [-1,+1]
    - Apply this if the different input features have different scales (or units)
    - *With images*: relative scales of pixels are already approximately equal (8bits)



[1] Common data reprocessing

Feature Extraction

- **Main motivation:** get out most of the data
- For classification task:
  find a space where samples from different classes are well separable



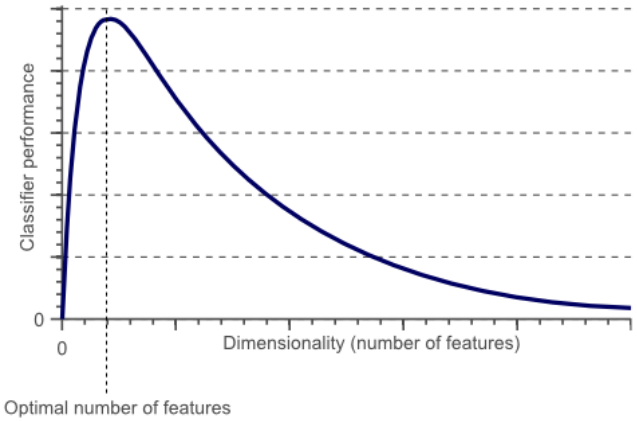
[2] Aju D. and Rajkumar Rajasekaran

Objectives:

- Reduce computational load of the classifier
- Increase data consistency
- Incorporate different sources of information into a feature vector: spectral, spatial, multisource, ...

Why Dimensionality Reduction

- High number of correlated features leads to
  - **Collinearity:** some of the independent variables are highly correlated
  - **Overfitting:** model too close to a particular training set (poor generalization)
  - **Hughes phenomenon:** increasing the dimensionality without increasing the number of training samples results in a decrease in classifier performance

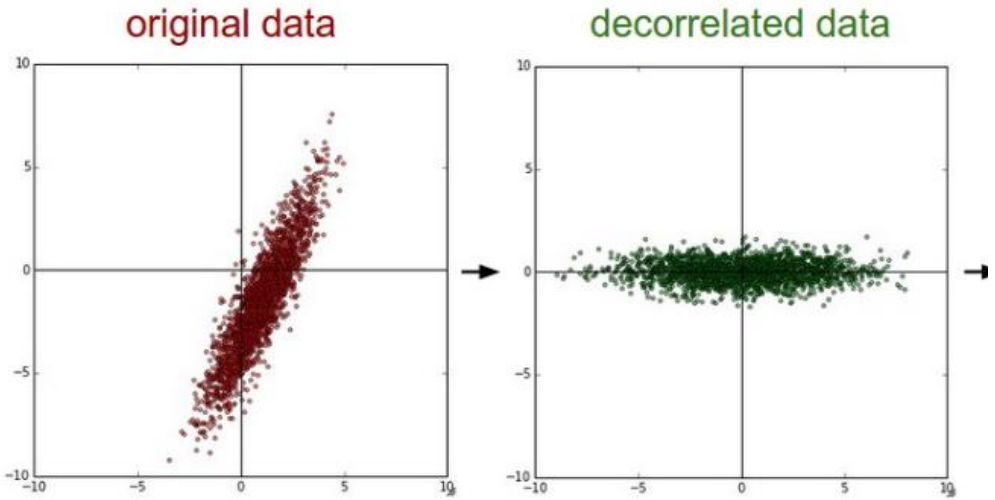


[3] Curse of dimensionality

- Most of the spectral feature extractors are based on multivariate analysis:
  - **"project data onto a subspace that maximize explained variance, minimize correlation, minimize error, etc."**
  - Linear methods are simple and intuitive, yet often not appropriate
  - Nonlinear methods give improved expressive power

Principal Component Analysis (PCA)

- Objective: identify patterns in data by detecting the correlation between variables
- Find the directions of maximum variance in high-dimensional data



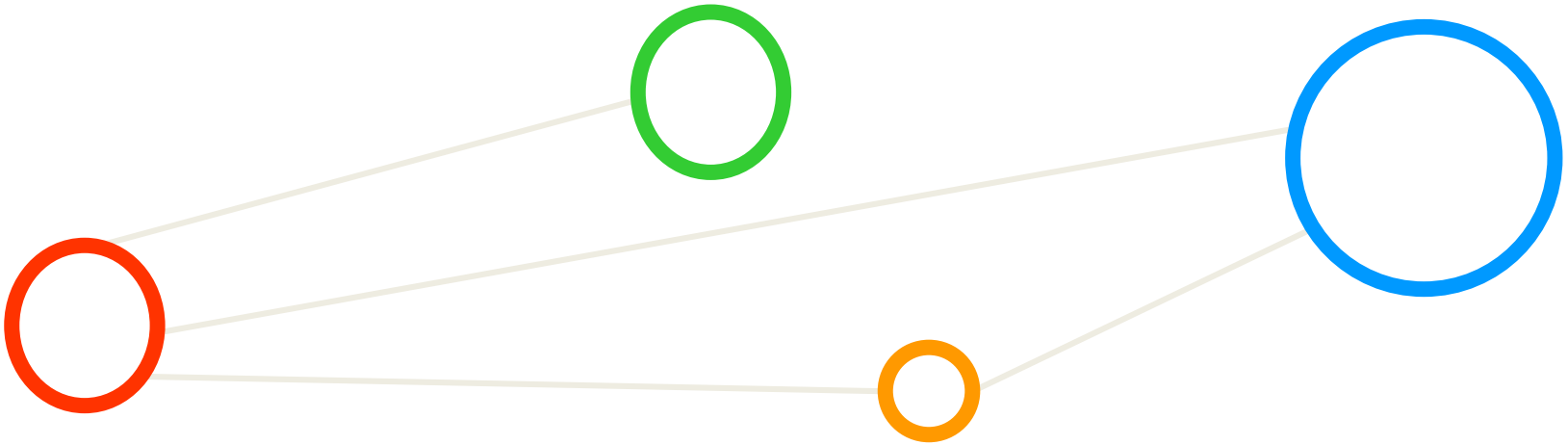
[1] Common data reprocessing

[4] Python Data Science Handbook

- If a strong correlation between variables exists -> **reduce the dimensionality**

- Project onto a smaller dimensional subspace while retaining most of the information
  - Projections that maximize the variance of the data

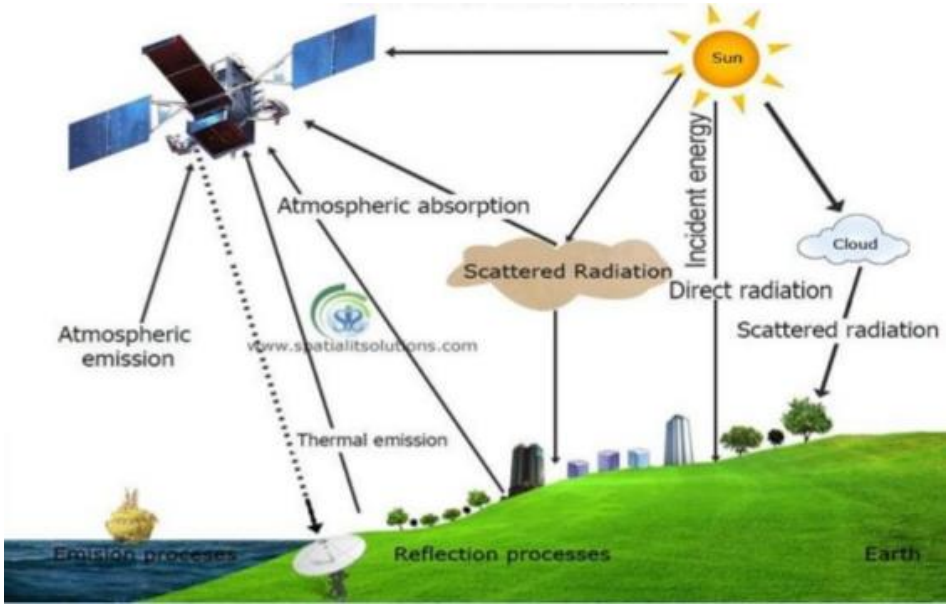
Remote Sensing



Overview of Remote Sensing

- Materials reflect, absorb, and emit electromagnetic radiation in a different way depending of their molecular composition and shape

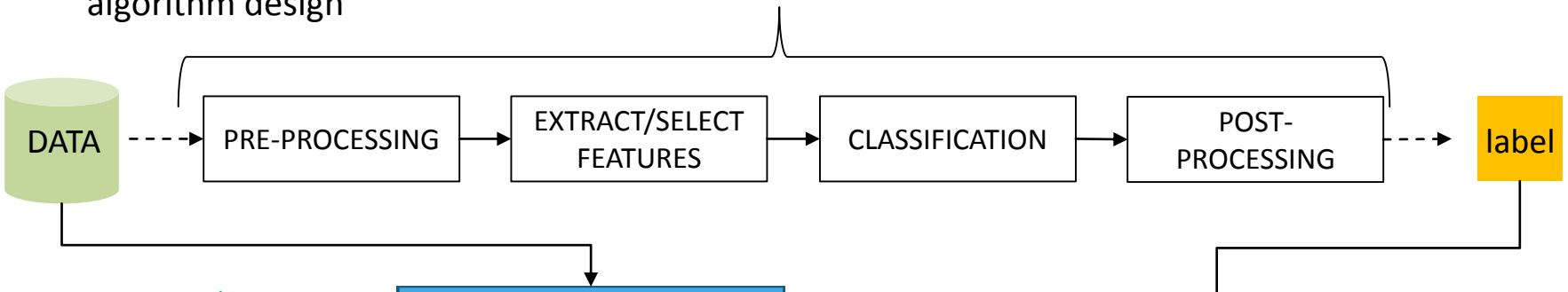
- Remote sensing exploits this physical fact and deals with the acquisition of information about a scene



[5] Remote Sensing

Exploitation of Pattern Recognition Systems

- **Pattern recognition** is the science of making inferences from perceptual data, using tools from statistics, probability, computational geometry**, machine (deep) learning**, signal processing, and algorithm design

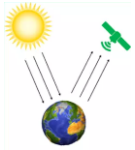


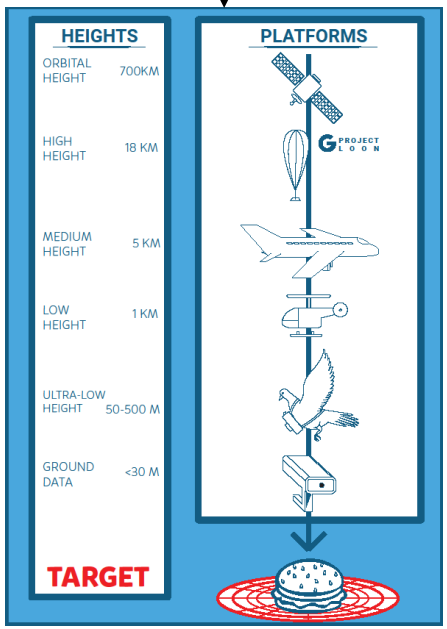
Active Sensor:

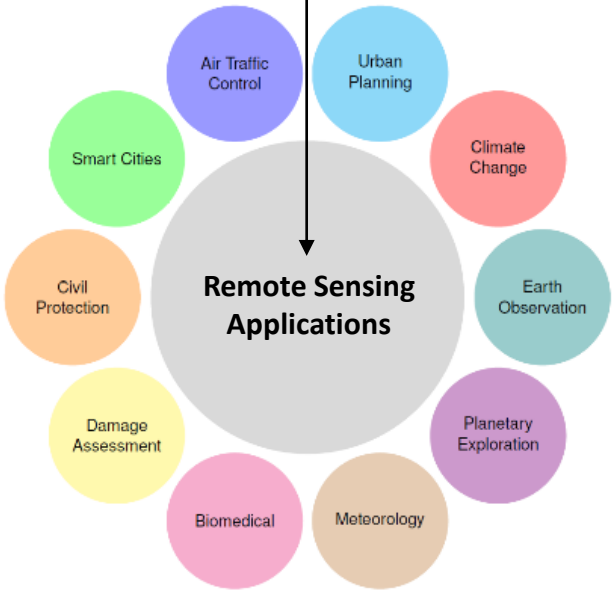
- Own source of illumination
- Capture image in day and night
- Any weather or cloud conditions

Passive Sensor:

- Natural light available
- Great quality satellite imagery
- E.g., Hyperspectral technology



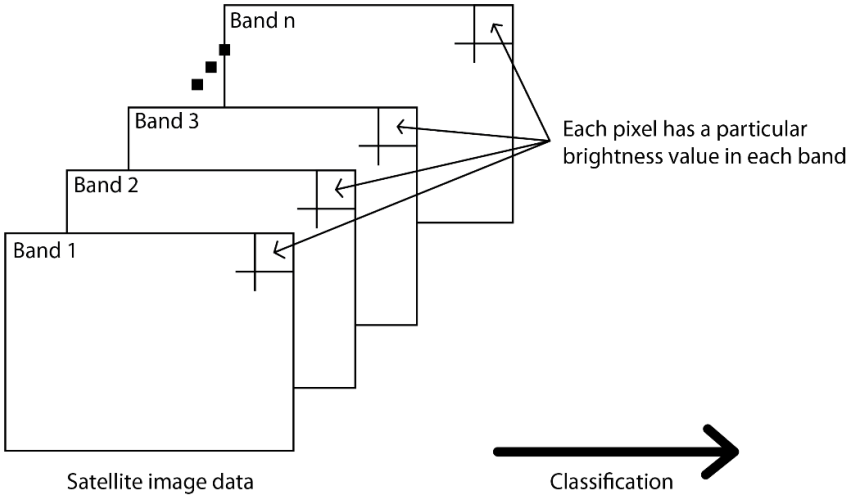




Platform selected according to the application

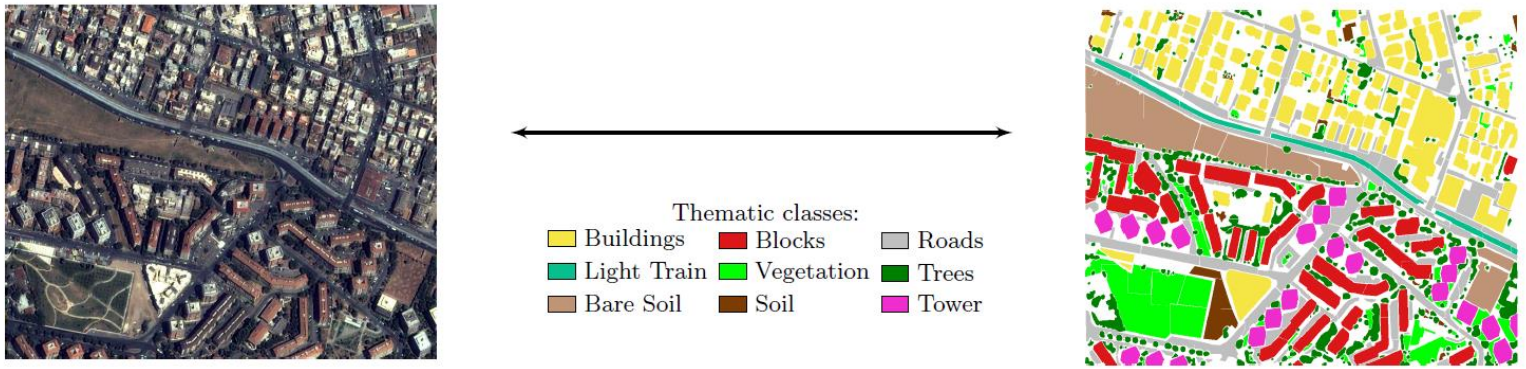
Classification of Satellite Images

- Acquired with ***Remote Sensing*** systems



Generation of thematic maps

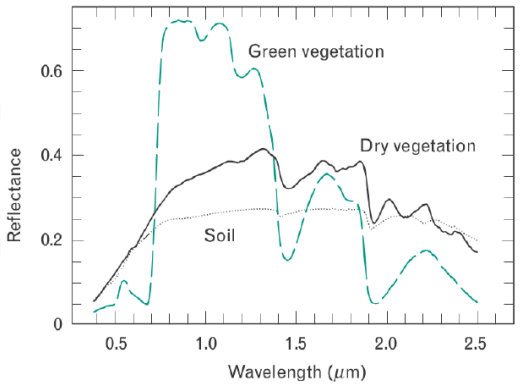
- E.g., Classification of urban areas

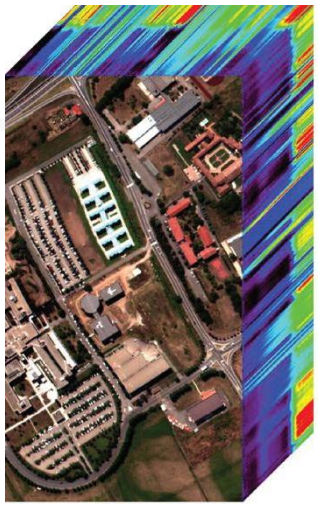


Extracting Features from Remote Sensing Images

- It is essential for:
  - Compress information for storage/transmission
  - Reduce (**spatial and spectral)** redundancy
  - Make image processing algorithms more robust (noise, dimension)
  - Visualize data characteristics
  - Understand the underlying physical relations

- **Hyperspectral images**
  - Allow finer material characterization
  - Different materials produce distinct electromagnetic radiation spectra
  - The spectral information contained in a hyperspectral image pixel can indicate the various materials present in a scene



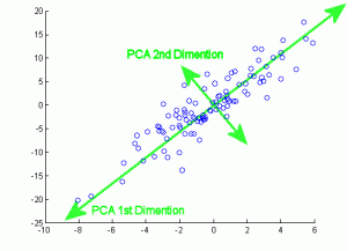


[6] Hyperspectral images

Example of PCA Applied on a Hyperspectral Image

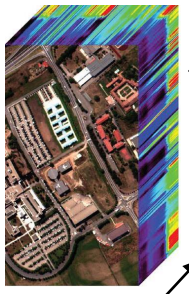
- Nearby bands tend to be correlated (correlation means redundancy - images “look alike”)
- Theoretically n bands = n dimensional data
- The “actual” dimension required to represent data with negligible information loss is lower

PCA finds the linear subspace that shows the largest variances (i.e., eigenvalue decomposition of the covariance matrix)



[6] Hyperspectral images

[7] H. Hotelling

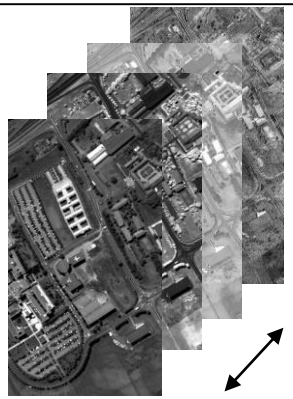


n = 103 bands

Affine transformation to decorrelate the axis

103 components

Pick K dimensions with the highest amount of variance



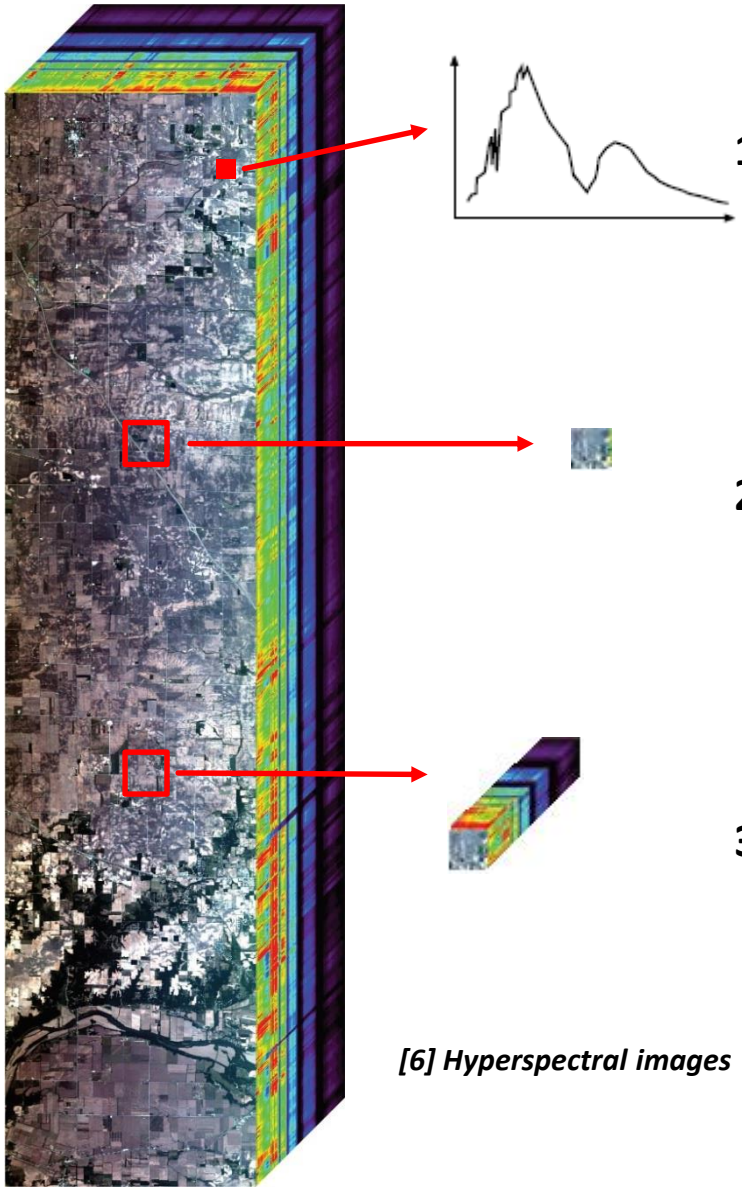
K = 4 Principal Components

Original ***PCA***

OA 71.82% ***70.55%***

Practicals with PCA and Hyperspectral images in Lecture 9

Extract Spectral and Spatial Features



[6] Hyperspectral images

1. **Spectral:**
   - Physically-based spectral features
   - Statistical multivariate methods: linear (**PCA**) and nonlinear

2. **Spatial/contextual**
   - Standard image processing descriptors
   - Advanced computer vision descriptors

3. **Spatio-spectral**
   - Extract features from spectral patches or regions

Complexity of Spatial Information

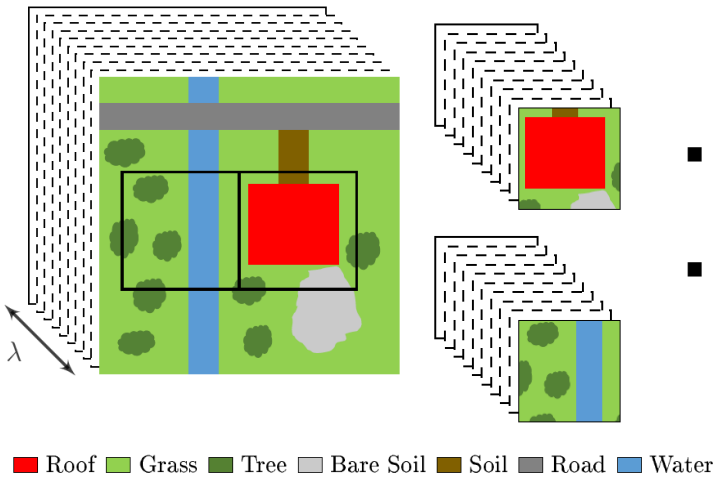
- Very High Spatial Resolution images: huge amount of details

WorldView-2 Panchromatic - Resolution 0.46 [m]

- Sub-metric resolution
- Allows for accurate analysis
- Objects with different scales and shapes

Spatial Information (1)

- As long as the pixel sizes remained similar to the objects of interest
  - Most classifier employed the pixel as the basic unit

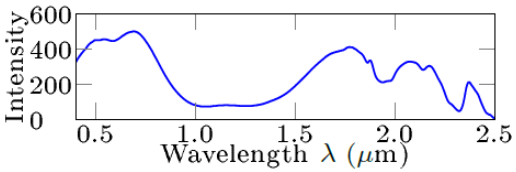


- Homogeneous structures can have similar sizes of pixels
- Two neighboring pixels may have low correlation

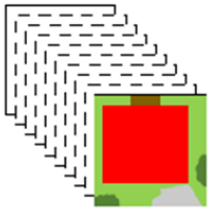
- During the 1980s and 1990s, pixel-wise classification methods:
  - assumed that each pixel is pure and typically labeled as a single land cover class

Spatial Information (2)

- The scene complexity and the spatial resolution determines the number of mixed pixels
- The spectral unmixing problem:
  - identify the pure materials (endmembers)
  - estimate their corresponding proportions (abundances)

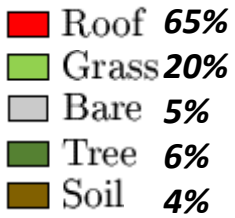


mixed spectral signature

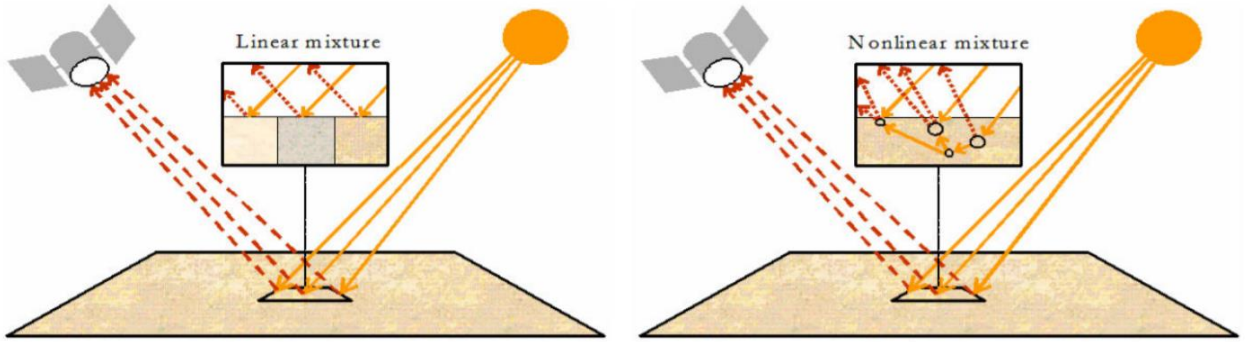


mixed pixel

soft classification



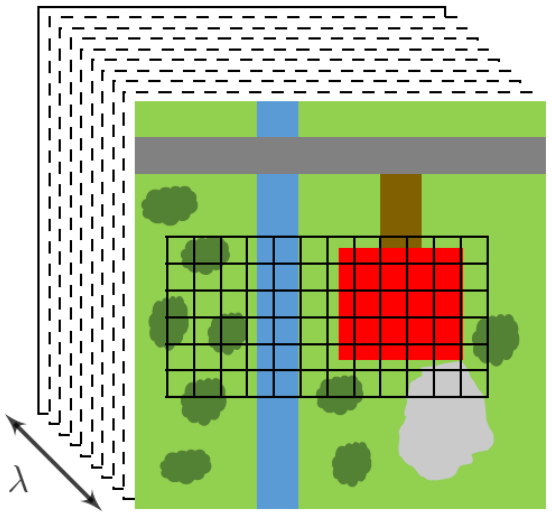
- Two models to analyze the mixed pixel

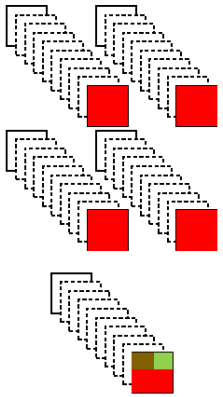


[8] A. Plaza, et al.

Spatial Information (3)

- When spatial resolution increases, structures are larger than the pixel size
- The correlation between neighboring pixels increases
  - adjacent pixels of a roof pixel belong to the same class with a high probability
  - structures can be represented as regions of spatially connected pixels





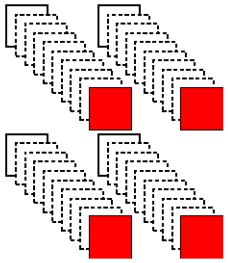
The presence of mixed pixels can not be avoided

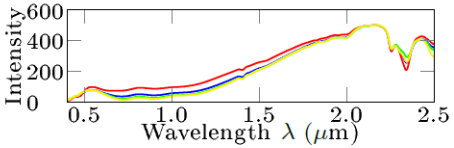
Spatial Information (4)

- The separability on the different classes in the spectral domain is reduced
  - spectral variability ***within-class*** increases (caused by shadows, sun angle, etc.)
  - spectral variability ***between*** different classes decreases

- Limited spectral resolution (technological constraints)

A pixel is a small part roof

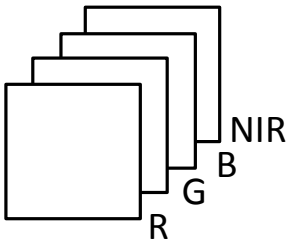




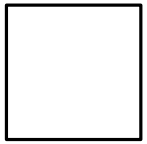
- Spatial contextual classifiers can exploit the correlation of pixels within a subset domain

Normalized Difference Vegetation Index (NDVI)

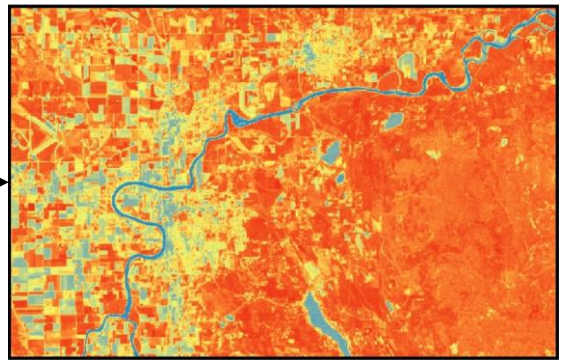
- Create additional relevant features from the existing raw features in the data
- Increase the predictive power of the classifier

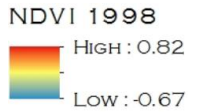


NDVI = (NIR – R) / (NIR + R)

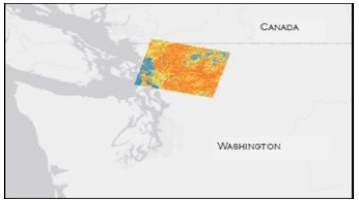


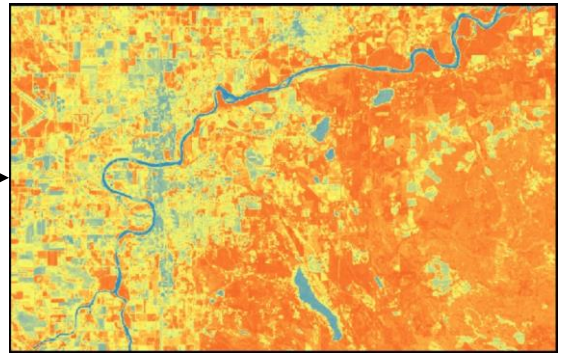
PREPROCESSING

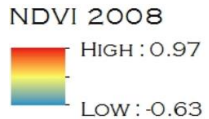


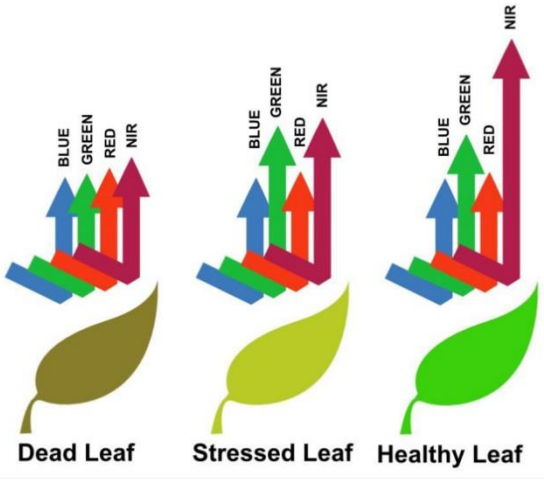


Bellingham, WA , US





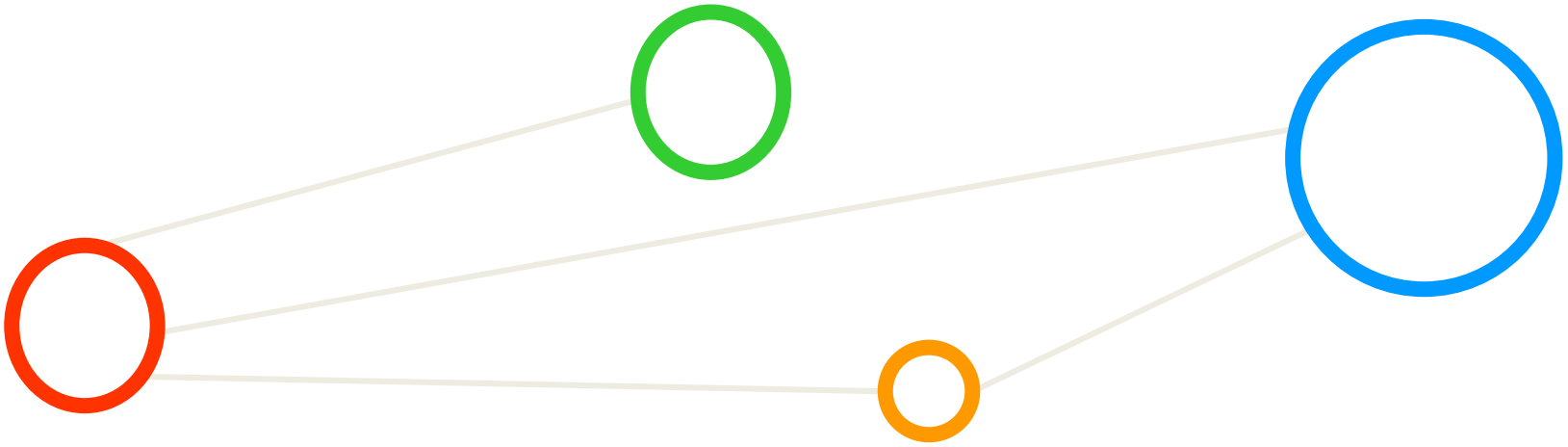




[9] NDVI

[10] NDVI & Classification

Learning with Limited Training Data

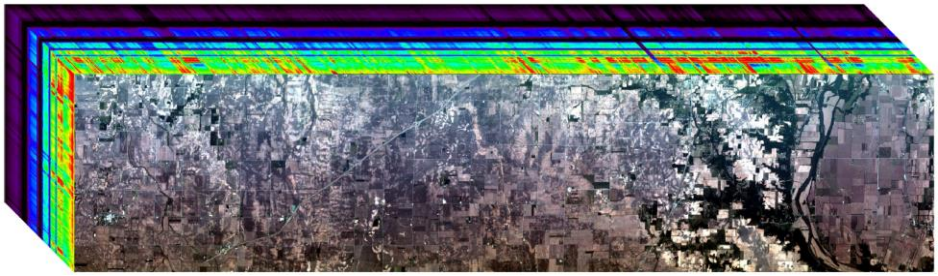


Limited Remote Sensing Training Data

- RS applications have massive amounts of temporal and spatial data (e.g., Sentinel 2)

- But not enough labeled training samples, which usually don't fully represent:
  - Seasonal variations
  - Object variation (e.g., plants, crops, etc.)

- Most online hyperspectral data sets have little-to-no variety



[6] Hyperspectral images

- DL systems with many parameters require large amounts of training data
  - Else they can easily overtrain and not generalize well

- DL systems in CV use very large training sets
  e.g., millions or billions of faces in different illuminations, poses, inner class variations, etc.

Possible Solutions

- Possible approaches to mitigate small training samples:

1. **Data augmentation**
   - Affine transformations, rotations, small patch removal, etc.

2. **Transfer learning**
   - Train on other imagery to obtain low-level to mid-level features

3. **Use ancillary data**
   - Other sensor modalities (e.g., LiDAR, SAR, etc.)

4. **Unsupervised training**
   - Training labels not required

Data Augmentation

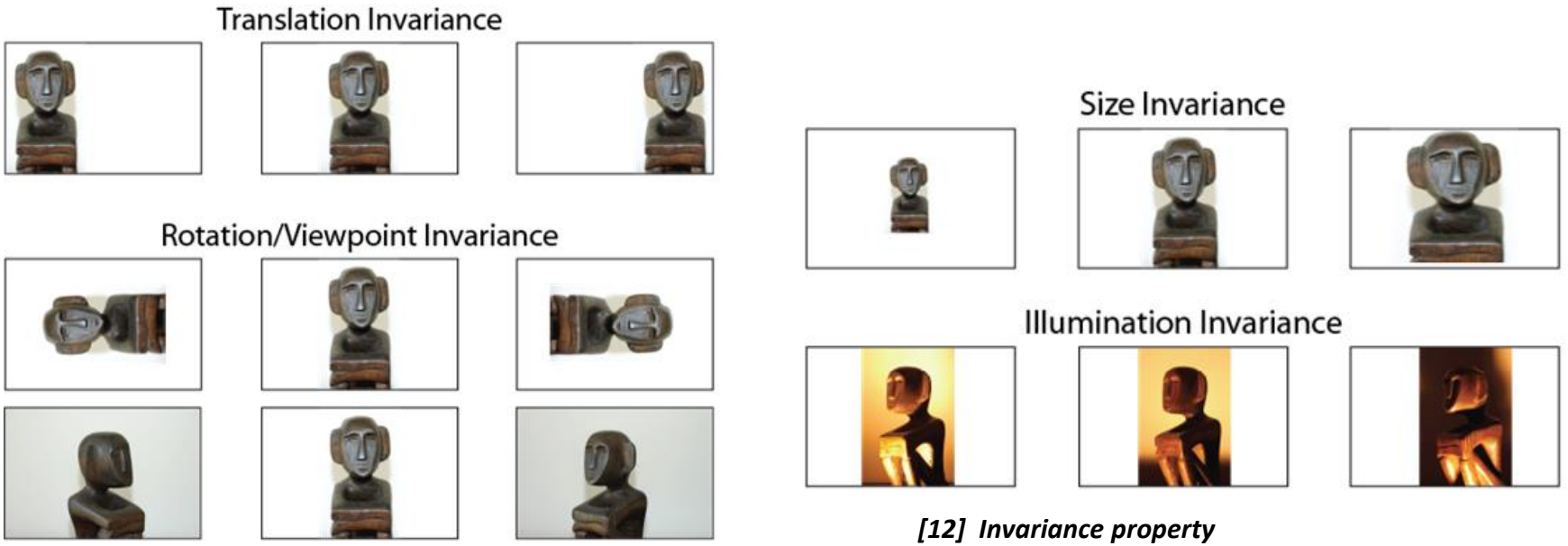
- Train with additional synthetically modified data
- Techniques to artificially increase the size of the training set
- Make minor changes such as flips, translations and rotations to the existing dataset
- Employed to counteract overfitting

[11] Data Augmentation

"A poorly trained neural network would think that these three tennis balls, are distinct, unique images"

Invariance

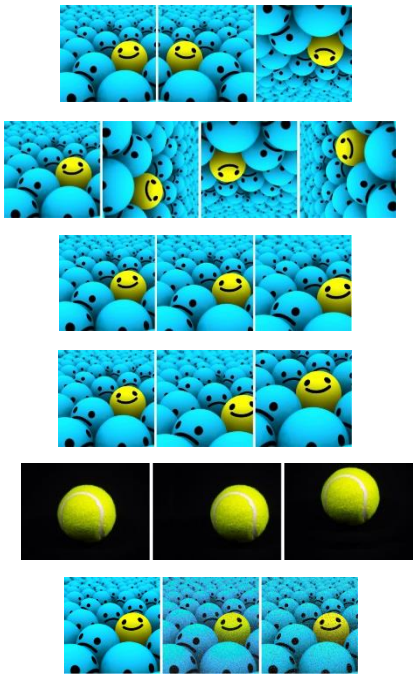
- Ability to recognize an object as an object, even when its appearance varies in some way
- It allows to abstract an object's identity from the specifics of the visual input
  - E.g., relative positions of the viewer/camera and the object.
- Well-trained classifiers can be invariant to translation, viewpoint, size or illumination



[12] Invariance property

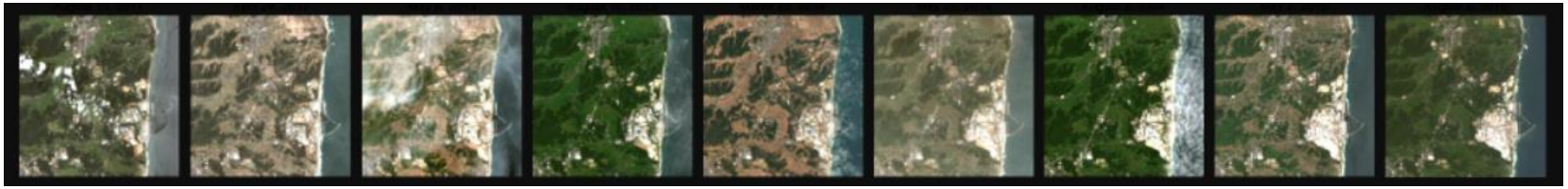
Popular Augmentation Techniques

- Flip horizontally and vertically
- Rotate
- Scaled outward or inward
- Crop: random sample a section
- Translate: moving the image along the X or Y direction
- Add noise



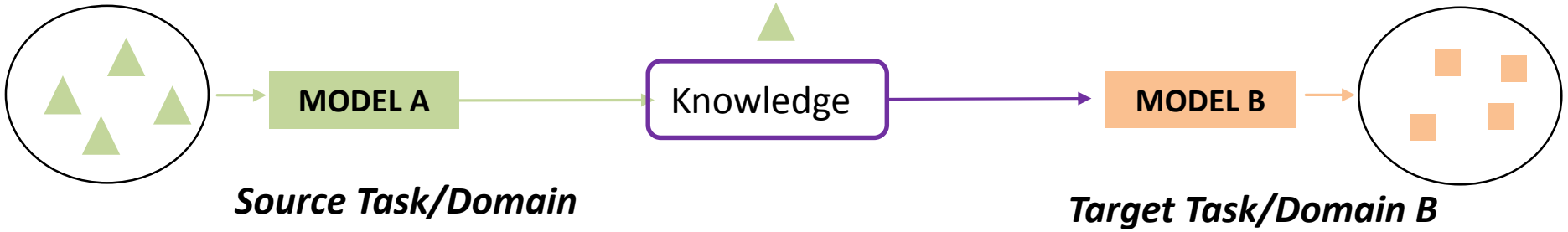
- Data augmentation is more challenging for remote sensing
  - Images exist in a variety of conditions (e.g., different seasons)
  - They cannot be accounted for by the above simple methods

[11] Data Augmentation



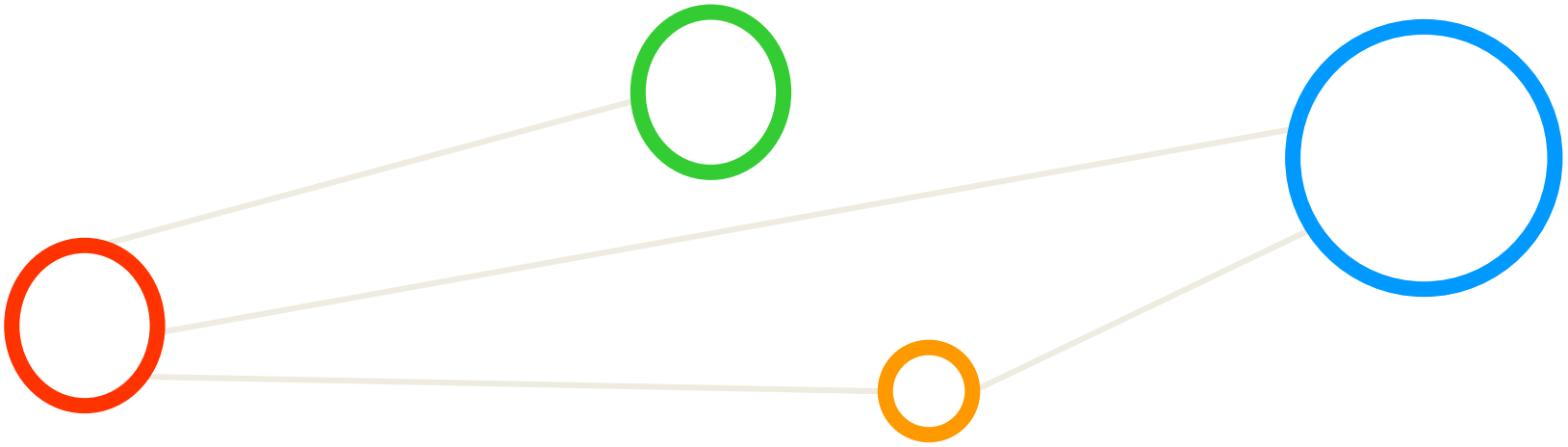
Transfer the Knowledge

- *Direct solution*: rebuild from scratch the predictive model using new training samples
- However it is preferable to reduce the need for and effort in recollecting new samples
- *Other solutions*: **transfer learning**, **domain adaptation** and **active learning** approaches
- Exploit the knowledge acquired by the available reference samples for classifying new images acquired over different geographical locations at diverse times with different sensors



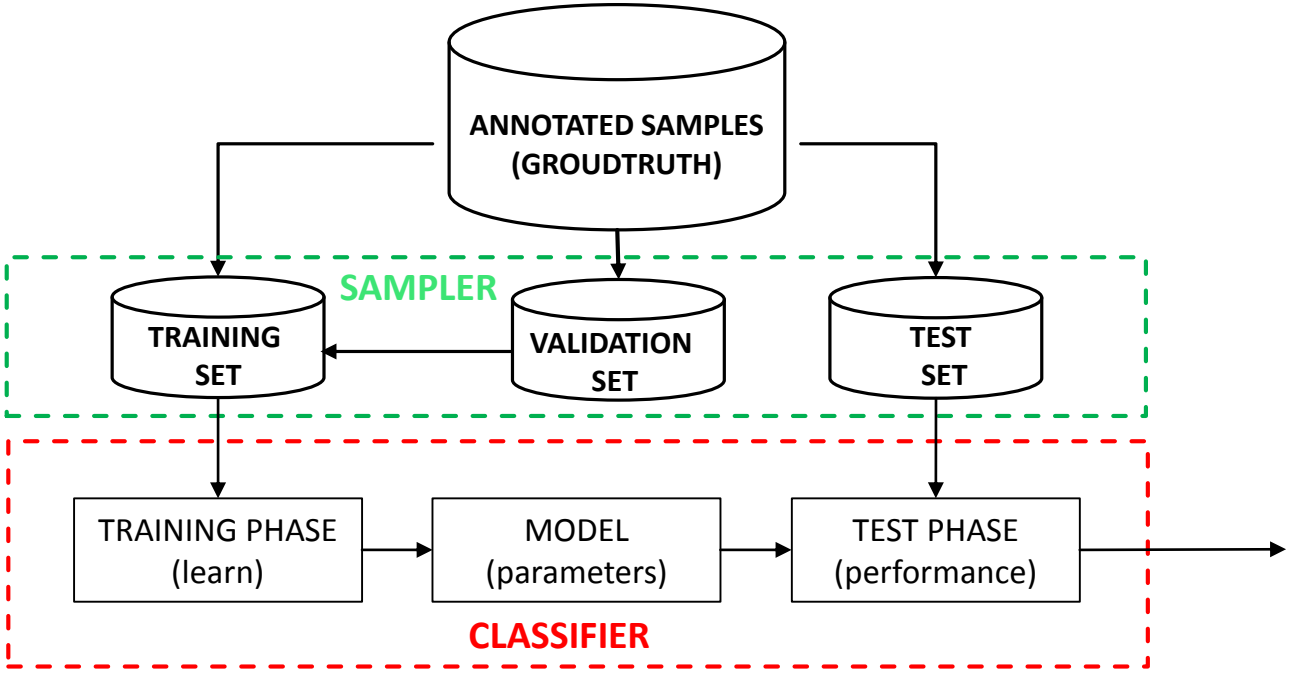
[13] Domain Adaptation

Performance Evaluation



How to Assess the Classifier Performance?

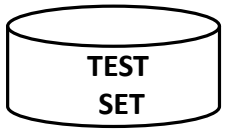
- If annotated samples are available, the classifier parameters are learned in a supervised way
- How to estimate the generalization error: split the groundtruth into three disjoint sets

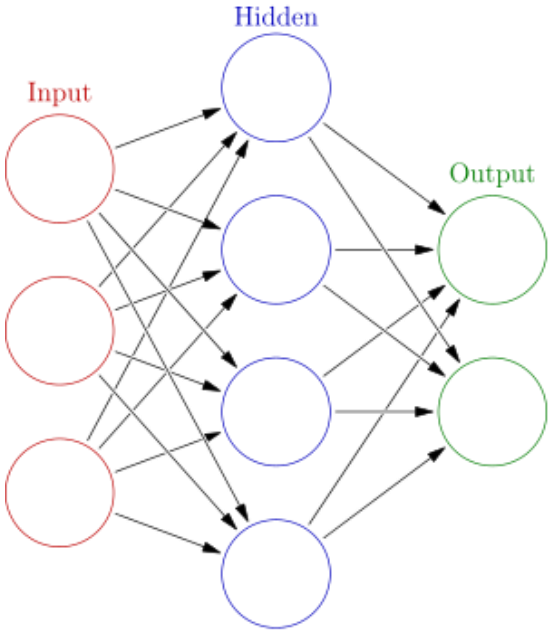


Performances: usually more influenced by the amount and quality of the training samples (i.e., sampling design) rather than the classifier/model complexity

Three Disjoint Sets

- **Training set**: used to train the model
  - How do we ensure that the model is not overfitting to the data in the training set?
- **Validation set**: used to validate the model during training
  - Its classification is based only on the model that is learnt from the training set
  - The model weights are updated based on this set
  - Help to adjust the hyperparameters (e.g., number of hidden layers, learning rate, etc..)
- **Test set:** used to test the model after it has been trained






```
Epoch 7/10
2152/2152 [==============================] - 34s 16ms/step - loss: 0.2539 - acc: 0.9021 - v
al_loss: 0.4802 - val_acc: 0.8374
Epoch 8/10
2152/2152 [==============================] - 34s 16ms/step - loss: 0.2087 - acc: 0.9198 - v
al_loss: 0.4097 - val_acc: 0.8589
Epoch 9/10
2152/2152 [==============================] - 34s 16ms/step - loss: 0.2465 - acc: 0.9080 - v
al_loss: 0.8998 - val_acc: 0.7405
Epoch 10/10
2152/2152 [==============================] - 34s 16ms/step - loss: 0.3857 - acc: 0.8568 - v
al_loss: 0.4116 - val_acc: 0.8549
```

Choose Parameters for a Classifier?

- **Validation:** if a lot of training data, we can use X% for training and Y% for validation
  - Test on different parameters
  - Retain parameters that give the highest accuracy on validation set

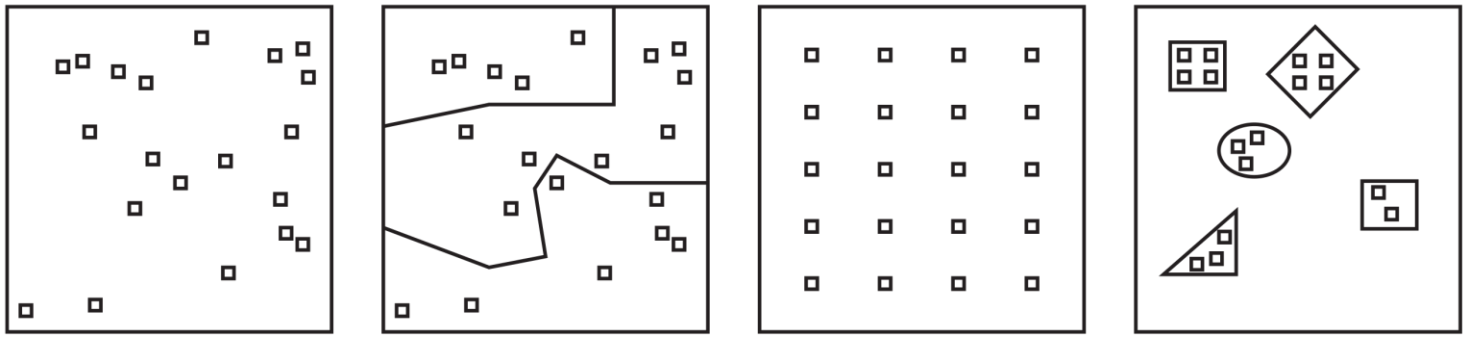
- If limited training data ⇒ **cross-validation**
  - Different types
  - Popular choice: k-fold cross-validation
    - Randomly partition the training data into k equal sized subsets
    - k times: one of k subsets is used for validation, and the rest of data are used for training

Sampling Strategies

- **Random sampling**:
  - Randomly select training samples within the area of each class
  - Often used, but bad idea if generalization required

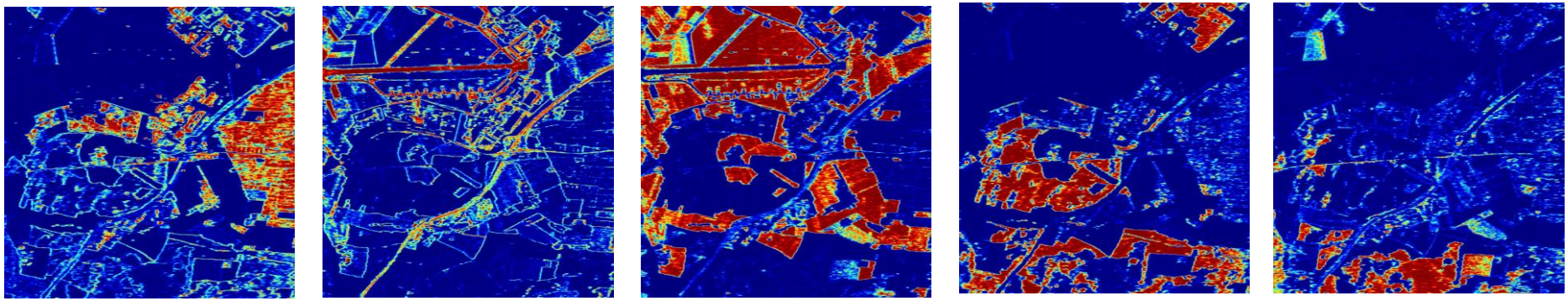
- **Patch sampling:**
  - Image is divided into blocks, test samples are from blocks that haven't been used for training

- **Cluster sampling:**
  - Train on one area, test on another area



How to Compare the Obtained Results with the Test Data?

- **Photo interpretation**
  - E.g., visual comparison of classification maps (it can idenfifies weaknesses of the classifier)



Estimated class posterior
Blue = low probability; Red = high probability

[14] R. Hänsch, et al.

- **Metrics**

  Accuracy measures: overall, class-specific, average, kappa coefficient, etc.

Confusion Matrix

- **Confusion matrix:**
  - It shows where the system mislabels one class as another
  - Each column represents the instances in a predicted class
  - Each row represents the instances in an actual class

Percentage	Classification data				
Reference data	C_1	C_2	C_3	Row total	Class-specific accuracy
C_1	C_{11}	C_{12}	C_{13}	$\sum_i^K C_{1i}$	$\frac{C_{11}}{\sum_i^K C_{1i}}$
C_2	C_{21}	C_{22}	C_{23}	$\sum_i^K C_{2i}$	$\frac{C_{22}}{\sum_i^K C_{2i}}$
C_3	C_{31}	C_{22}	C_{33}	$\sum_i^K C_{3i}$	$\frac{C_{33}}{\sum_i^K C_{3i}}$
Column total	$\sum_i^K C_{i1}$	$\sum_i^K C_{i2}$	$\sum_i^K C_{i3}$	N	
User's accuracy	$\frac{C_{11}}{\sum_i^K C_{i1}}$	$\frac{C_{11}}{\sum_i^K C_{i2}}$	$\frac{C_{33}}{\sum_i^K C_{i3}}$		

C_i : the class i

C_{ij}: number of pixels classified to the class j and referenced as the class i

Accuracies Measures (1)

- **Overall Accuracy (OA):** percentage of correctly classified pixels (K is the number of classes)

$$OA = \frac{\sum_{i}^{K} C_{ii}}{\sum_{ij}^{K} C_{ij}}$$

- **Class Accuracy (CA):** percentage of correctly classified pixels for a given class

$$CA_i = \frac{C_{ii}}{\sum_{j}^{K} C_{ij}}$$

- **Average Accuracy (AA):** mean of class-specific accuracies for all the classes:

$$AA = \frac{\sum_{i}^{K} CA_i}{K}$$

Percentage	Classification data				
Reference data	C_1	C_2	C_3	Row total	Class-specific accuracy
C_1	C_{11}	C_{12}	C_{13}	$\sum_{i}^{K} C_{1i}$	$\frac{C_{11}}{\sum_{i}^{K} C_{1i}}$
C_2	C_{21}	C_{22}	C_{23}	$\sum_{i}^{K} C_{2i}$	$\frac{C_{22}}{\sum_{i}^{K} C_{2i}}$
C_3	C_{31}	C_{22}	C_{33}	$\sum_{i}^{K} C_{3i}$	$\frac{C_{33}}{\sum_{i}^{K} C_{3i}}$
Column total	$\sum_{i}^{K} C_{i1}$	$\sum_{i}^{K} C_{i2}$	$\sum_{i}^{K} C_{i3}$	N	
User's accuracy	$\frac{C_{11}}{\sum_{i}^{K} C_{i1}}$	$\frac{C_{11}}{\sum_{i}^{K} C_{i2}}$	$\frac{C_{33}}{\sum_{i}^{K} C_{i3}}$		

Accuracies Measures (2)

- **Kappa Coefficient (*k*):** percentage of agreement
  - Correctly classified pixels
  - Corrected by the number of agreements that would be expected purely by chance

$$\kappa = \frac{P_o - P_e}{1 - P_e}$$

$$P_o = OA$$

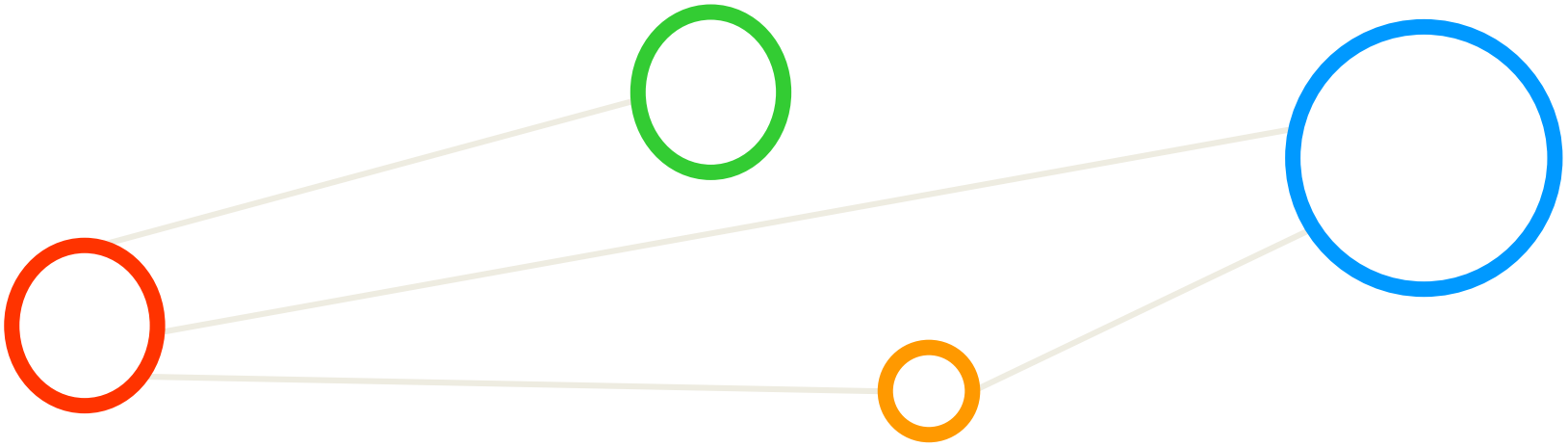
$$P_e = \frac{1}{N^2} \sum_i^K C_{i\cdot} C_{\cdot i}$$

$$C_{i\cdot} = \sum_j^K C_{ij}, \quad C_{\cdot i} = \sum_j^K C_{ji}$$

Percentage	Classification data				
Reference data	C_1	C_2	C_3	Row total	Class-specific accuracy
C_1	C_{11}	C_{12}	C_{13}	$\sum_i^K C_{1i}$	$\frac{C_{11}}{\sum_i^K C_{1i}}$
C_2	C_{21}	C_{22}	C_{23}	$\sum_i^K C_{2i}$	$\frac{C_{22}}{\sum_i^K C_{2i}}$
C_3	C_{31}	C_{22}	C_{33}	$\sum_i^K C_{3i}$	$\frac{C_{33}}{\sum_i^K C_{3i}}$
Column total	$\sum_i^K C_{i1}$	$\sum_i^K C_{i2}$	$\sum_i^K C_{i3}$	N	
User's accuracy	$\frac{C_{11}}{\sum_i^K C_{i1}}$	$\frac{C_{11}}{\sum_i^K C_{i2}}$	$\frac{C_{33}}{\sum_i^K C_{i3}}$		

N: number of referenced pixels

Lecture Bibliography



Lecture Bibliography (1)

- [1] Common Data Processing Pipeline
  Online: http://cs231n.github.io/neural-networks-2/
- [2] Aju D. and Rajkumar Rajasekaran , “A Multi-Stage Hybrid CAD Approach for MRI Brain Tumor Recognition and Classification” Online: https://www.researchgate.net/publication/296373267_A_MultiStage_Hybrid_CAD_Approach_for_MRI_Brain_Tumor_Recognition_and_Classification
- [3] Curse of Dimensionality
  Online: https://towardsdatascience.com/curse-of-dimensionality-2092410f3d27
- [4] Jake VanderPlas, "Python Data Science Handbook, Essential Tools for Working with Data", O'Reilly Media, 2016
  Online: http://shop.oreilly.com/product/0636920034919.do
- [5] What is Remote Sensing?
  Online: https://www.baltic-transcoast.uni-rostock.de/en/news/news-2018-2017/remote-sensing-class-2017/
- [6] Hyperspectral Images
  Online: https://engineering.purdue.edu/~biehl/MultiSpec/hyperspectral.html
- [7] H. Hotelling, “Analysis of a complex of statistical variables into principal components", in Journal of Educational Psychology, 24, 417–441, and 498–520, 1933
  Online: https://www.scribd.com/document/59617538/Analysis-of-a-Complex-of-Statistical-Variables-Into-Principal-Components
- [8] A. Plaza, G. Martín, J. Plaza, M. Zortea and S. Sánchez, “Recent Developments in Endmember Extraction and Spectral Unmixing“ , in Optical Remote Sensing, vol 3. Springer, Berlin, Heidelberg, 2011.
- [9] Normalized Difference Vegetation Index (NDVI)
  Online: http://www.agasyst.com/portals/NDVI.html
- [10] NDVI & Classification
  Online: https://lholmesmaps.wordpress.com/my-work-2/environmental-studies-421-gis-iv-advanced-gis-applications/2-2/

Lecture Bibliography (2)

- [11] Data Augmentation - How to use Deep Learning when you have Limited Data
  Online: https://medium.com/nanonets/how-to-use-deep-learning-when-you-have-limited-data-part-2-data-augmentation-c26971dc8ced
- [12] Invariance property
  Online: https://i.stack.imgur.com/iY5n5.png
- [13] D. Tuia, C. Persello and L. Bruzzone, "Domain Adaptation for the Classification of Remote Sensing Data: An Overview of Recent Advances," in IEEE Geoscience and Remote Sensing Magazine, vol. 4, no. 2, pp. 41-57, June 2016.
- [14] R. Hänsch, A. Ley and O. Hellwich, "Correct and still wrong: The relationship between sampling strategies and the estimation of the generalization error," 2017 IEEE International Geoscience and Remote Sensing Symposium (IGARSS), Fort Worth, TX, 2017, pp. 3672-3675


Data
Science
Big
Research
science
analysis
methods
simulations
processing
big
technologies
computing
European
TB
Computational
computational
Scientific
Supercomputing
datasets
HPC
Juelich
Climate
modelling
structures
research
databases
services
measurement
within
fusion
new
funding
Policy-based
concepts
device
Analysis
forms
cross-disciplinary
resources
Centre
climate
Computer
expertise
year
like
Test
E.g
Computing
disciplines
results
Approaches
construction
Fusion
Enable
Cross-Disciplinary
High
nodes
PB
community
Environment
Modelling
different
EU
centers
Key
scientific
important
files
model
international
SMAQ
engineering
References
etc
DKRZ
Use
storage
basis
Infrastructure
ENES
formats
brain
increasing
via
EUDAT
physical
IPCC
Common
Project
Methods
MapReduce
create
Using
GB
Work
images
codes
stored
Resources
analyze
Large
Summary
data-intensive
Service
levels
often
DLCL
Level
external
use
JSC
Matters
Concrete
performance
e.g
system
acquisition
Landscape
System
AR5
Hardware
compute
access
hundreds
Nutshell
Services
large-scale
cores
network
Understanding
Technologies
Earth
distributed
Structure
management
DLCLs
directory
project
General
current
Germany
Simulation
Provide
NASA
Energy
systems
day
Processing
Health