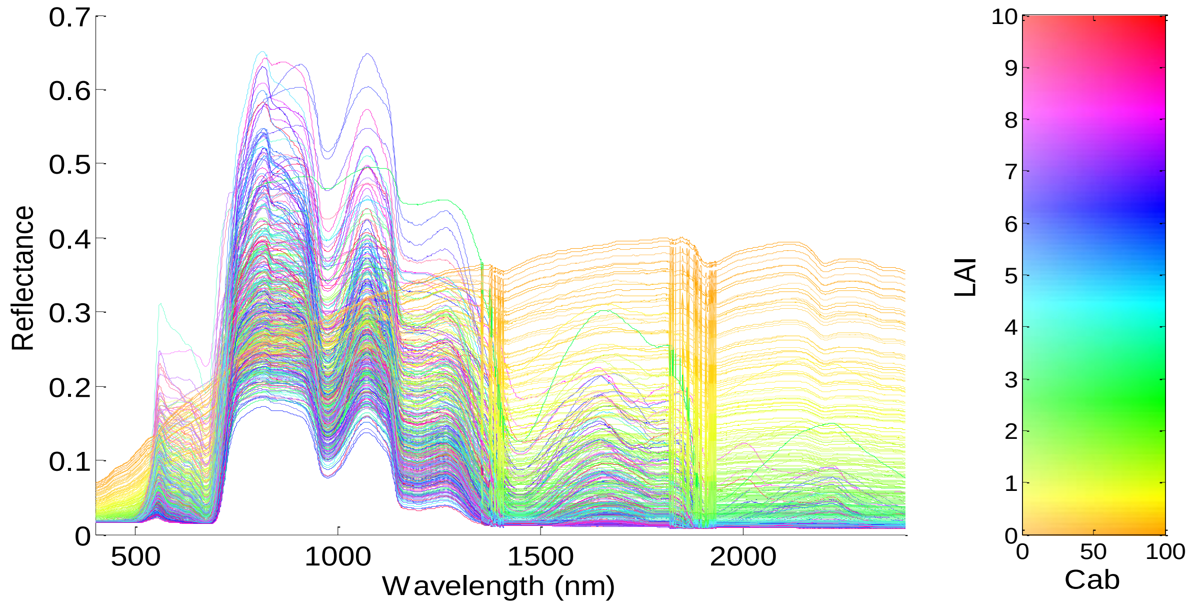
Emulation of radiative transfer models

Jochem Verrelst, Juan Pablo Rivera & Jose Moreno
Image Processing Laboratory, Univ. of Valencia (Spain)

Annual OPTIMIZE Workshop and MC Meeting
22 February 2017

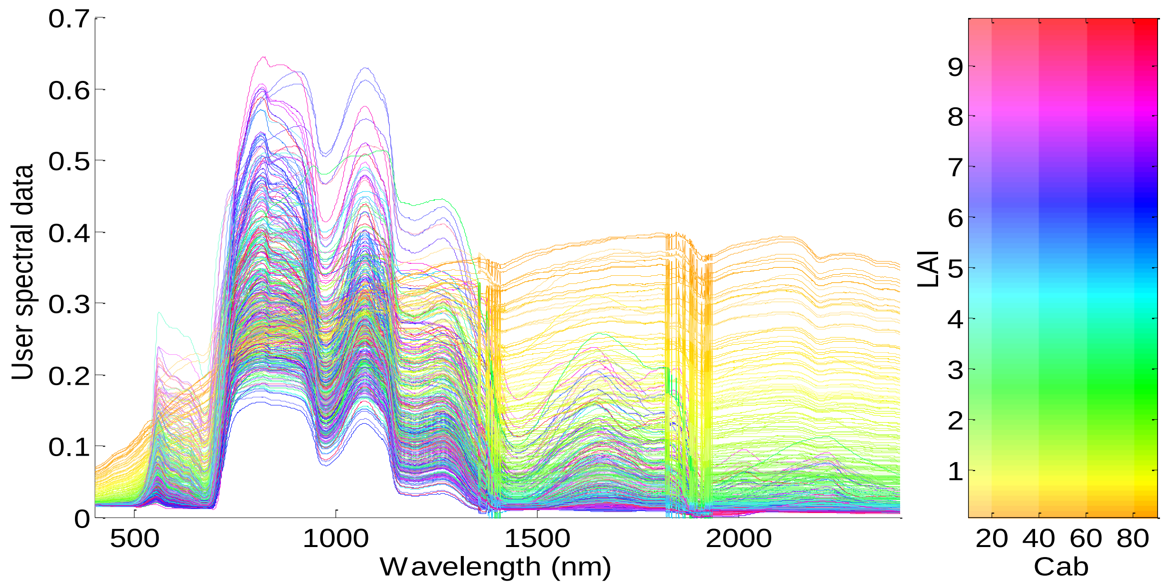
Any difference? Which model would you choose?

37 min



SCOPE

11 s



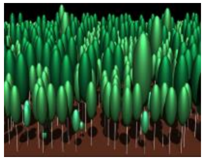
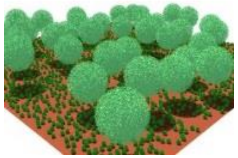
Emulator

(emulated SCOPE)

MOTIVATION

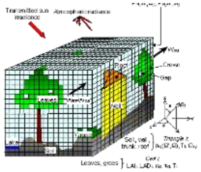
Advanced RTMs: *more realistic but slow*

- **Radiative transfer models (RTMs)** are widely used in remote sensing science, e.g. for development of new missions and retrieval (**inversion**).
- When choosing an **RTM**, **a trade-of between invertibility and realism** has to be made: **simpler models** are easier to invert but **less realistic**, while **advanced models** more realistic but require a large amount of variables to be configured.

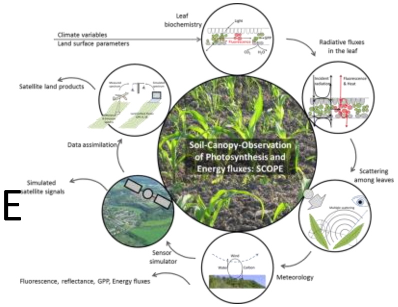




<u>Examples of advanced models:</u>

- **Ray tracing models** (e.g. FLIGHT, RAYTRAN, DRAT)



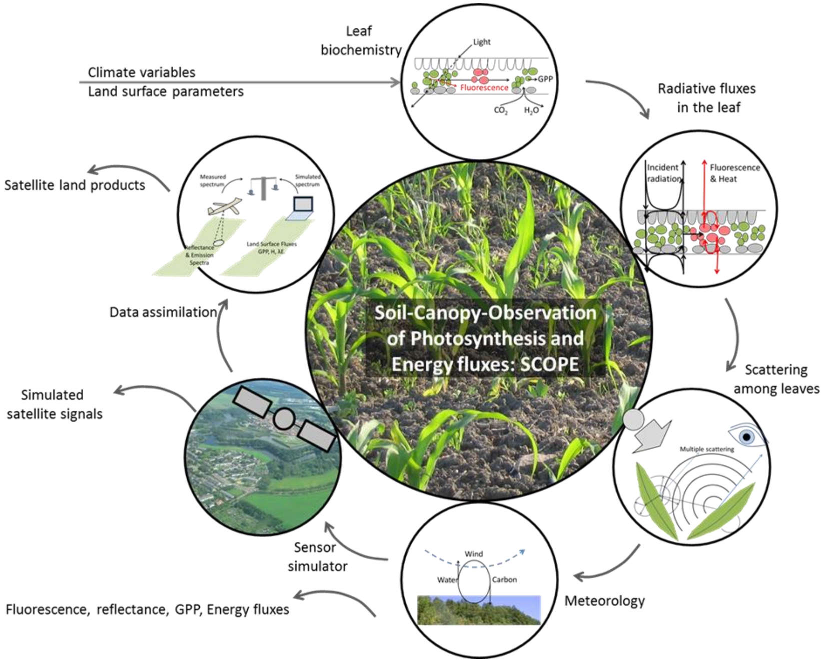
- **Voxel models**: DART

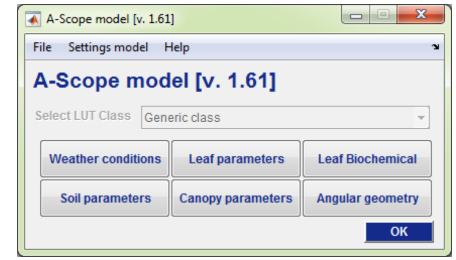


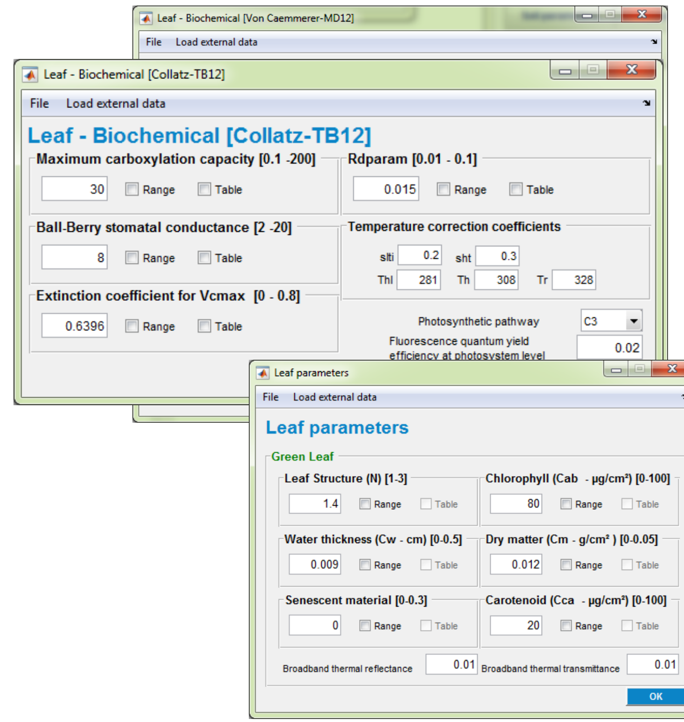
- **Soil-Vegetation-Atmosphere-Transfer (SVAT) models:** e.g. SCOPE

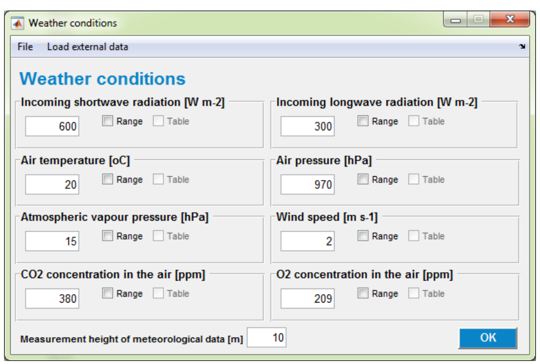
- **Main drawback of complex models involves their long processing speed**: ***the more computationally expensive, the longer it takes to generate output.***
- Long processing time makes that **advanced RTMs are of little use** for operational tasks, e.g., pixel-by-pixel retrieval schemes.

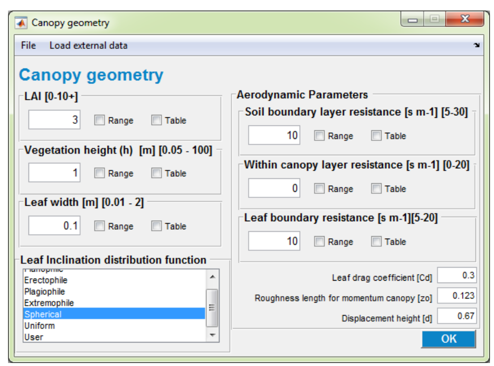
SCOPE (C. Van der Tol)











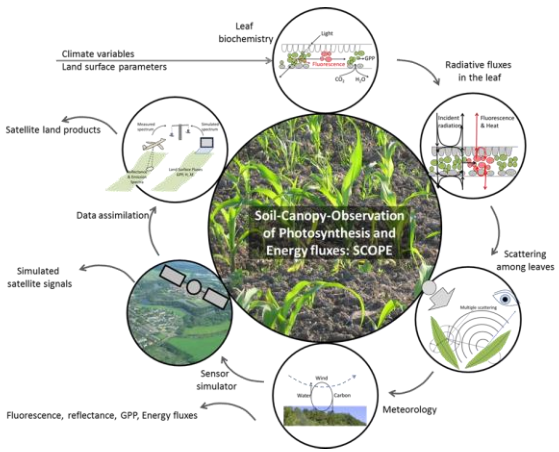
- **SCOPE generates multiple outputs, including directional SIF & reflectance. However, for operational use it is rather slow (>7 min for 100#).**
- Recently, it has been proposed **to approximate RTMs through machine learning** (Rivera et al., 2015; Gomez-Dans and Lewis, 2016).

EMULATION

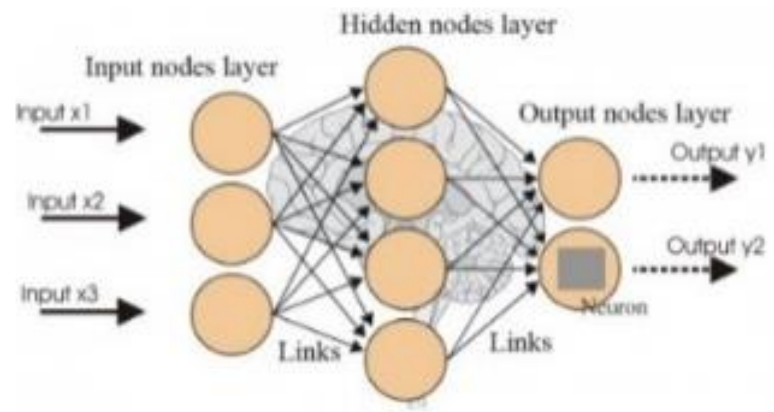
<u>Emulators</u> are surrogate statistical models that are able to approximate the processing of an RTM - <u>at a fraction of the computational cost</u>:

making a statistical model of a physical model

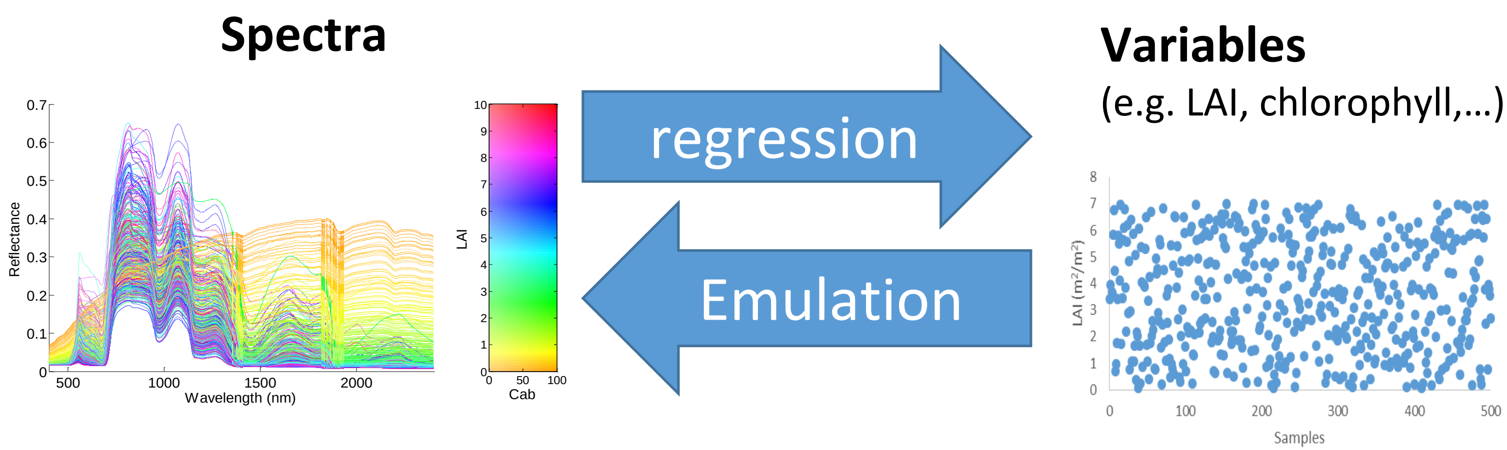
RTM



Machine learning



Emulation applied to RTMs:



- In principle any **nonlinear, adaptive machine learning regression algorithm (MLRAs) can serve as emulator**.
- However**, to emulate RTM spectral output,** the MLRA should have the capability to reconstruct **multiple outputs**, i.e. the complete spectrum: **resolved with dimensionality reduction techniques** (e.g. PCA).

"Curse of dimensionality" is a blessing

Processing steps *(Rivera et al, 2015; Verrelst et al., 2016)*

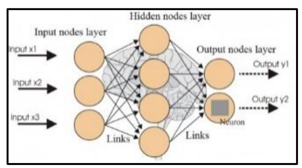
Input (variables + spectra) → **Splitting into training/validation** → **PCA on spectra** → **MLRA training looping over components** → **Prediction of components** → **Reconstruction of full spectrum** → **Validation** → **Emulator**

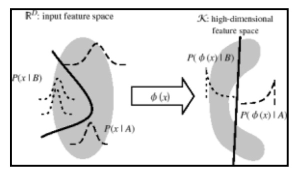
With this processing chain any RTM can be converted into an accurate emulator.

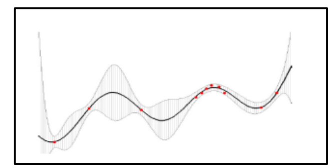
Emulators great idea… what about accuracy?

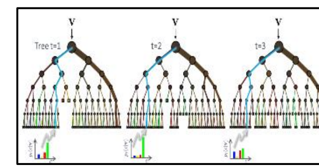
Various open questions:

1) *Role of machine learning regression algorithm (MLRA)*

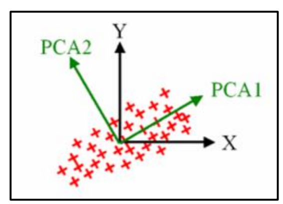




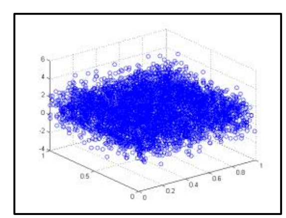




2) *Role of dimensionality reduction (DR) method?*
   1) *DR method*
   2) *# components*



3) *Role of LUT size training?*



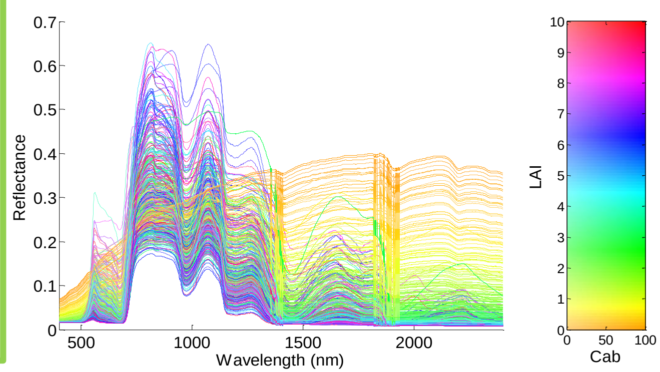
Experimental setup: emulating SCOPE

Experimental setup:

- SCOPE TB12-D: LUT 500# @ 1 nm; **8 variables**
- 4 machine learning methods tested: **RF, KRR, NN, GPR**
- **PCA, PLS and PPLS** dimensionality reduction methods
- **# 10, 20, 30, 40** components tested; **70/30%** training/validation
- **LUT of # 500, 1000** samples

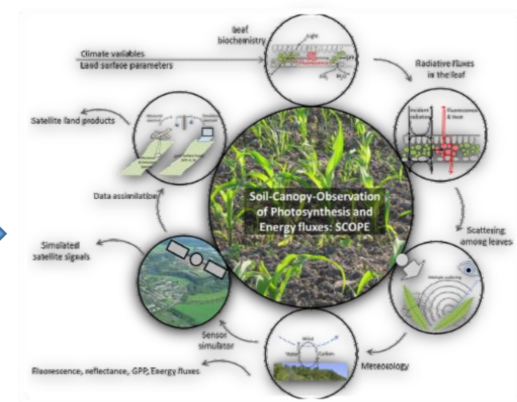
- N (1-3)
- LCC (0-100)
- Cw (0-0.5)
- Cdm (0-0.5)
- Cs (0-0.3
- LAI (0-10)
- Hc (0-2)
- SMC (0.-0.7)

SCOPE: ⏰ 37 min (500#)



Emulator

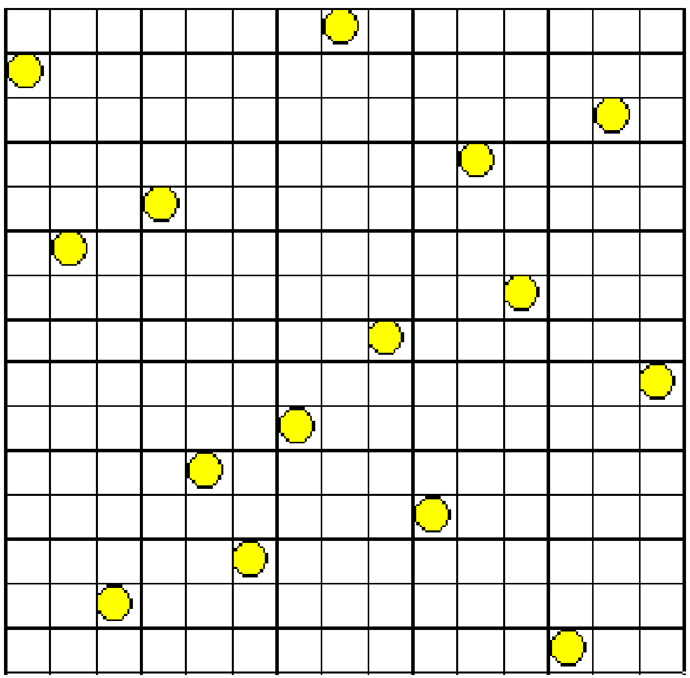
Emulated SCOPE

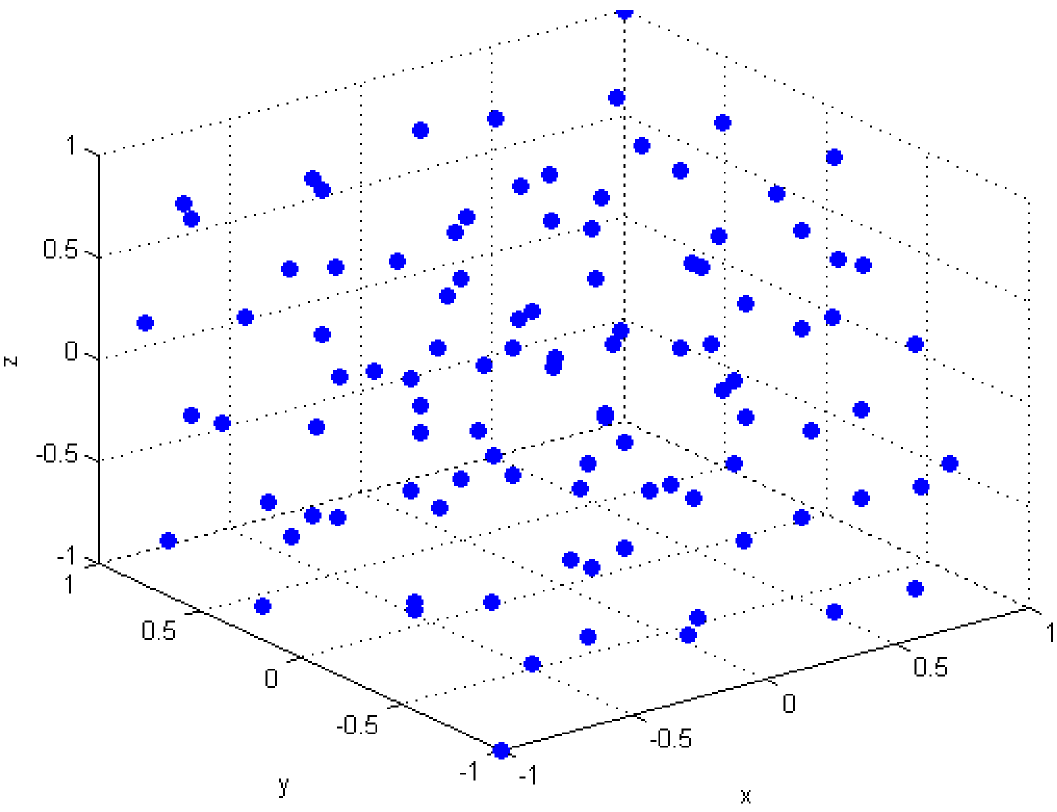


- Random Forests (RF)
- Kernel Ridge Regression (KRR)
- Neural Networks (NN)
- Gaussian Processes Regression (GPR)

Is a small LUT of #500 samples sufficiently covering the parameter space?

Latin Hypercube Sampling (LHS)

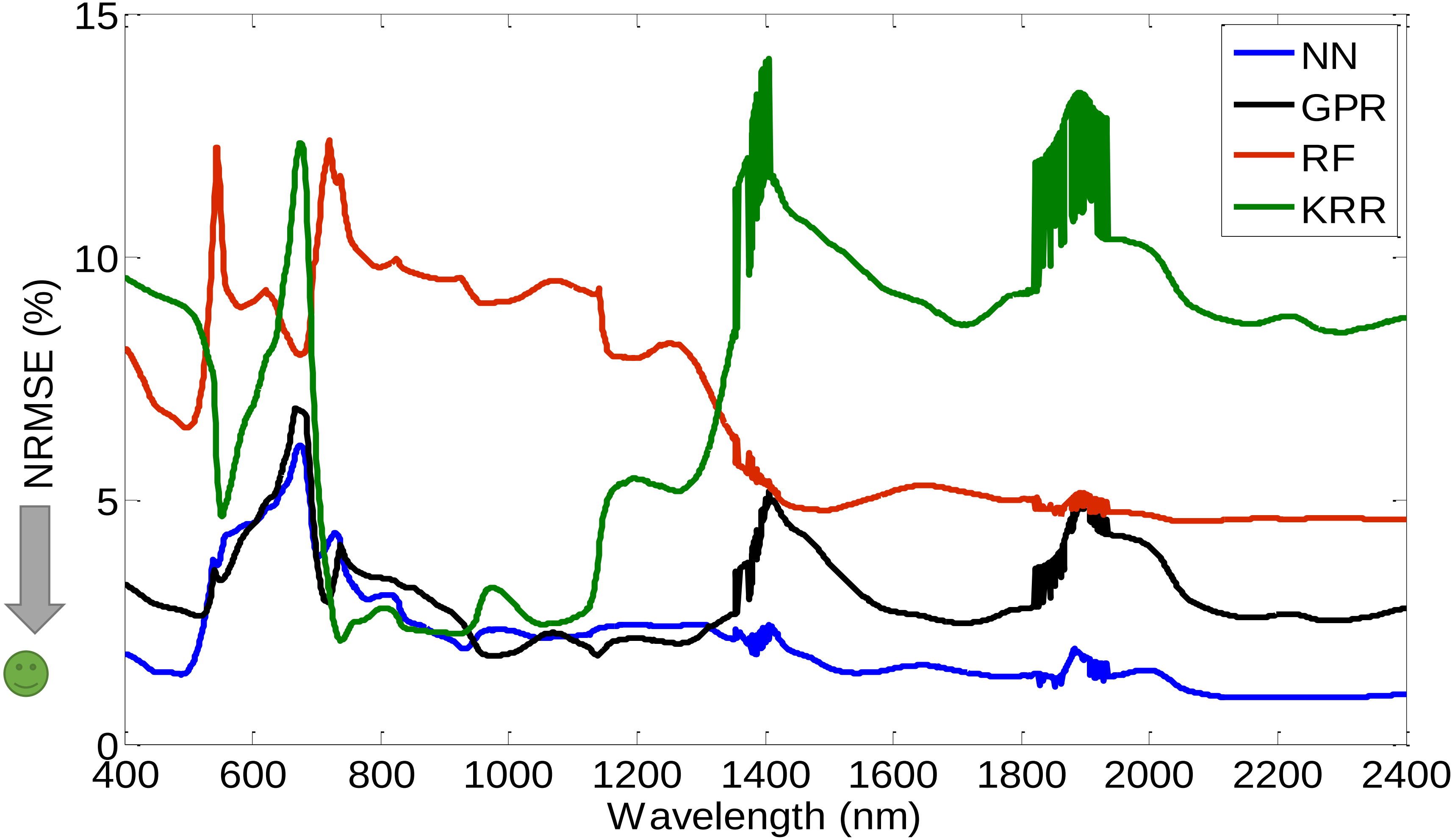




RESULTS

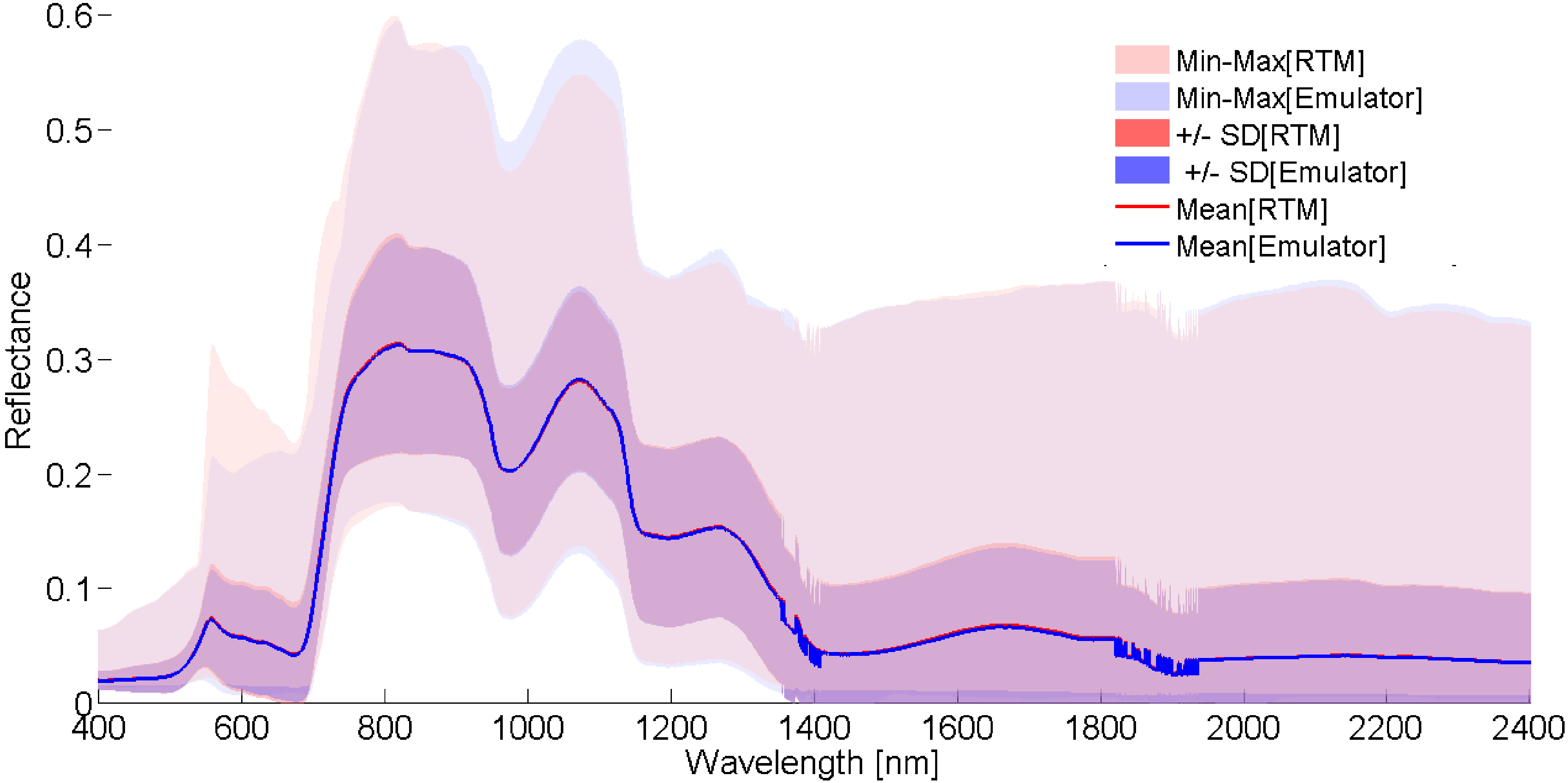
1) Role of machine learning algorithms

Relative errors 30% validation data MLRAs with 20PCA



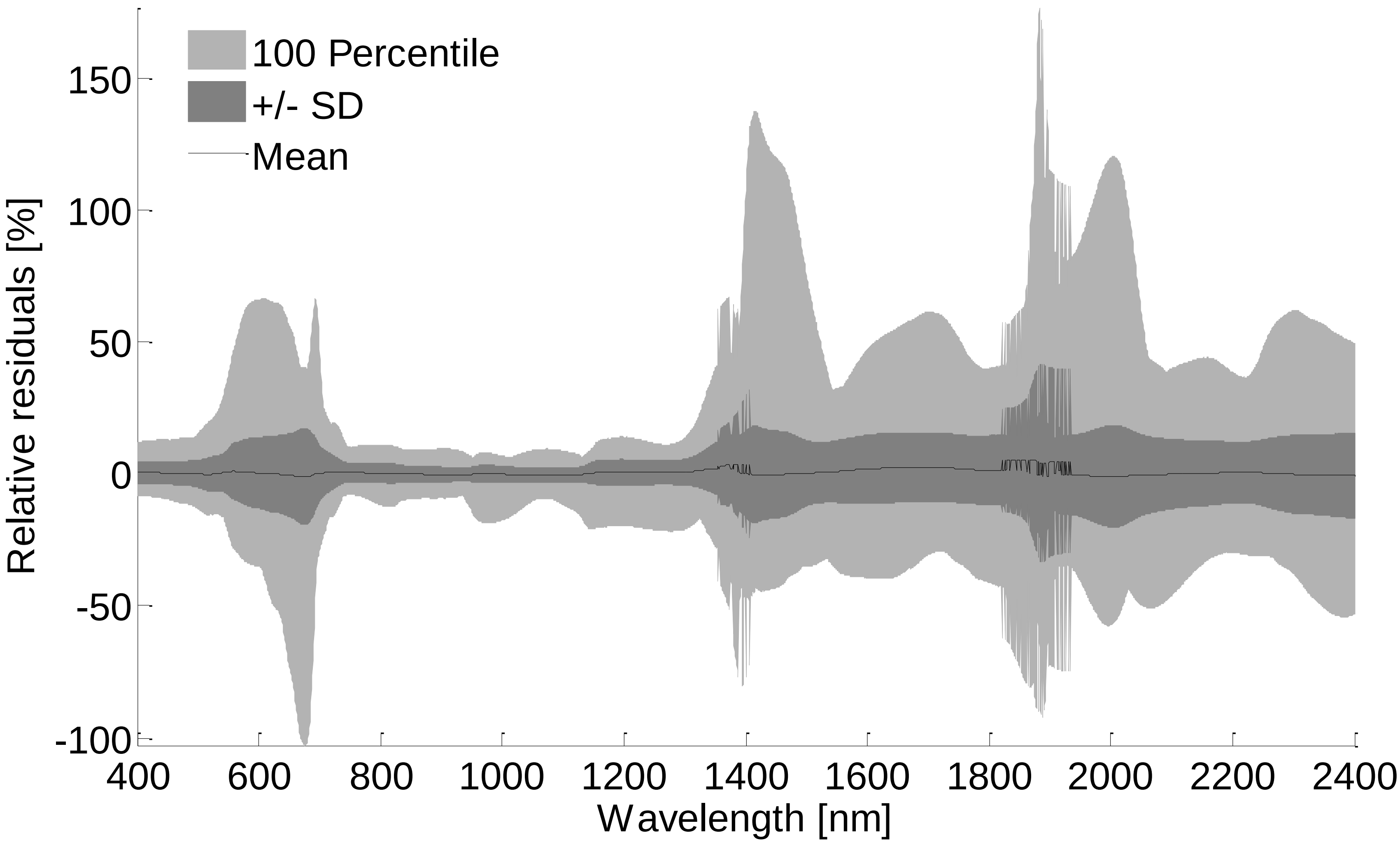
Here, **NN** best performing (<3%).

NN emulator vs RTM validation data: overview stats



The mean and SD closely matching, however min-max boundaries not.

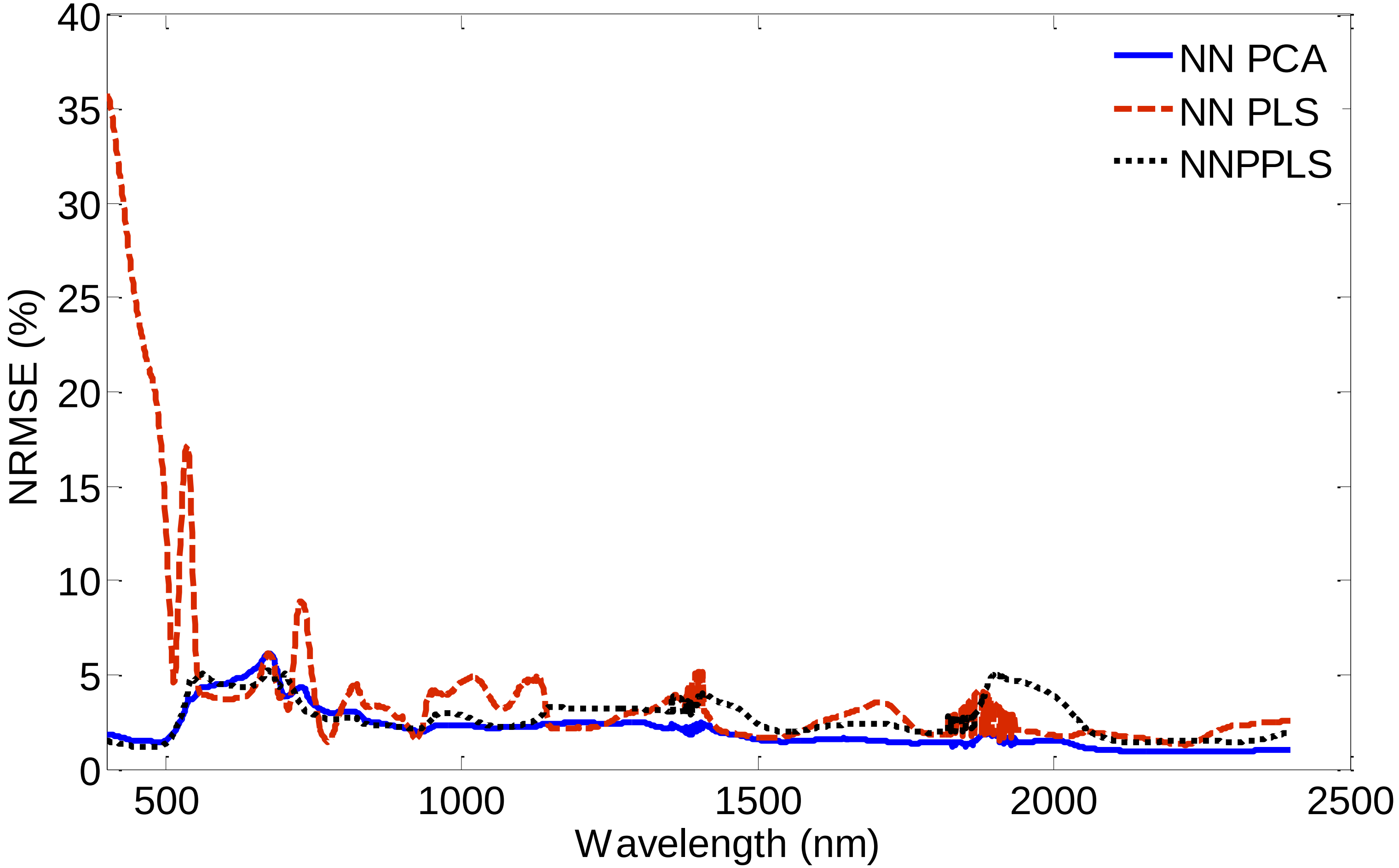
NN emulator relative residuals



Residuals show outliers…. Need for improvements NN emulator.

2a) Role of dimensionality reduction methods:

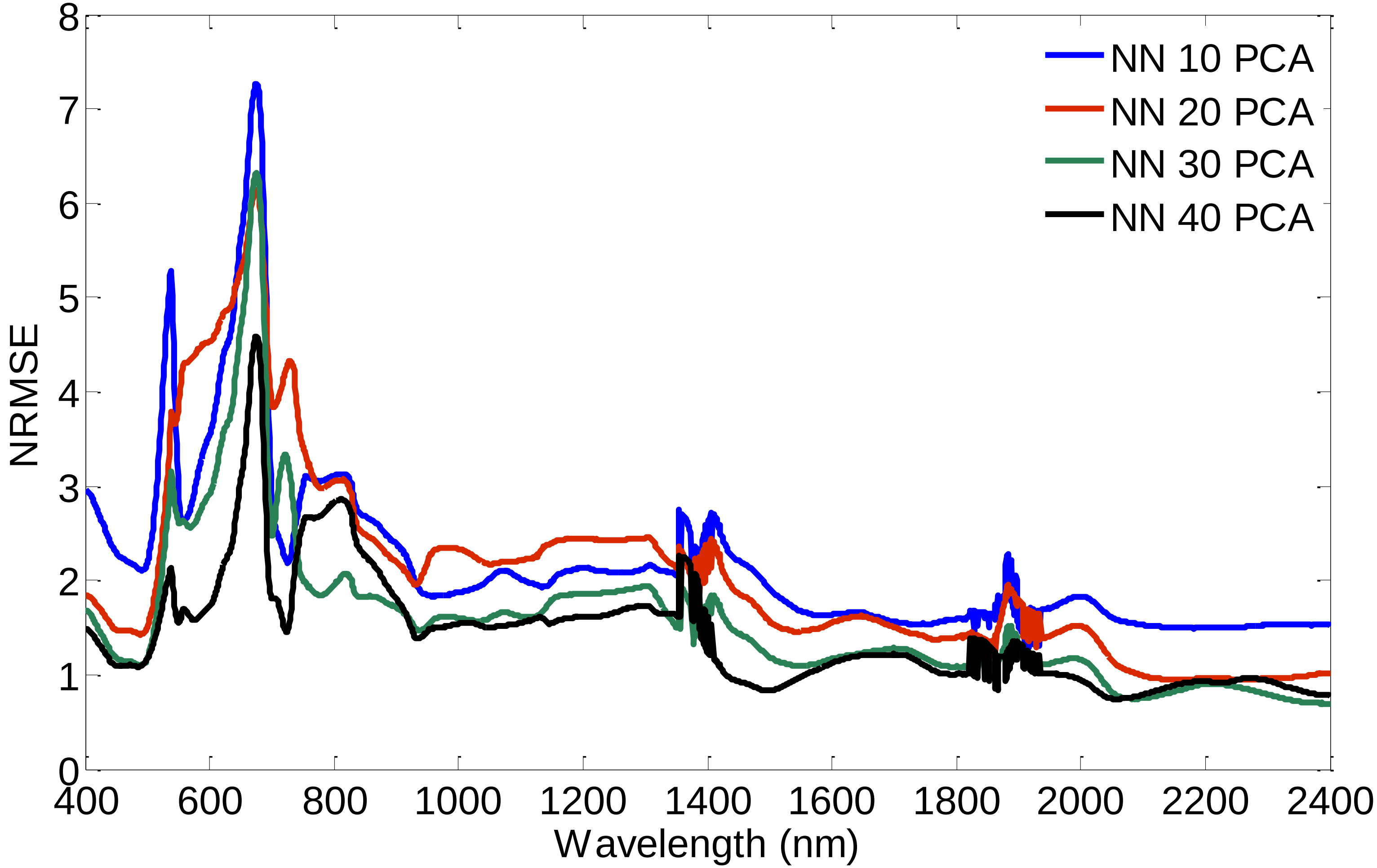
- Principal component analysis (PCA)
- Partial least squares (PLS)
- Penalized PLS (PPLS)



PCA seems to be best suited.

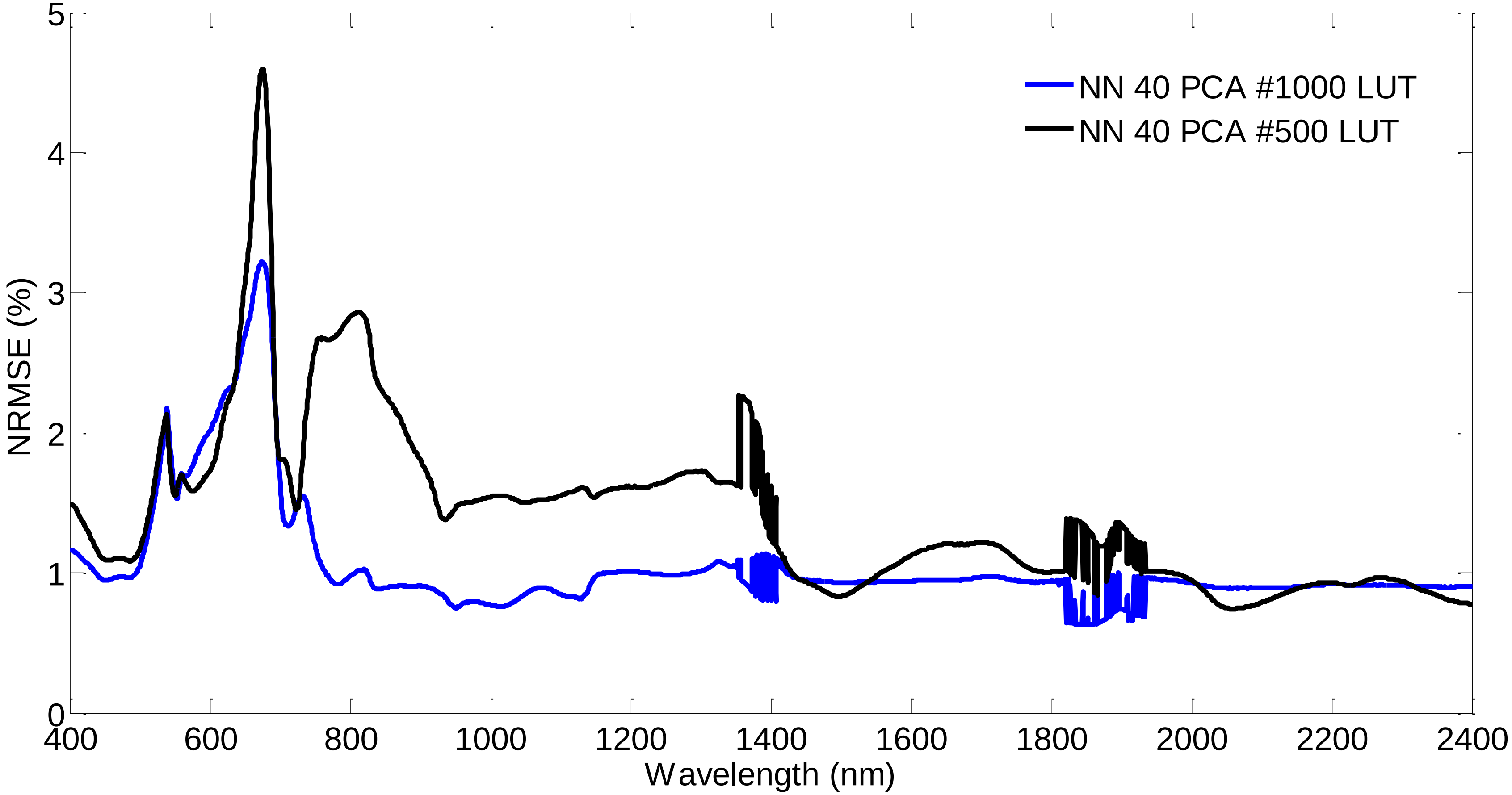
2b) Role of #PCA components:

10, 20, 30, 40 PCA



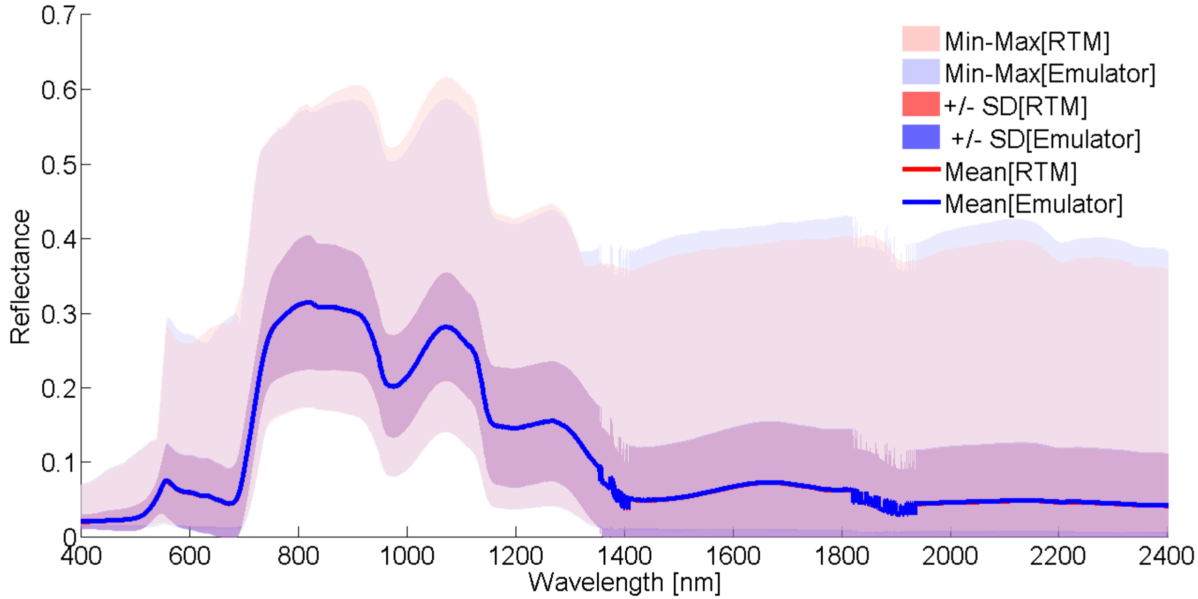
The # of components has a considerable impact on the accuracy: **40 components bring errors down to < 2%** (but slows down processing a bit).

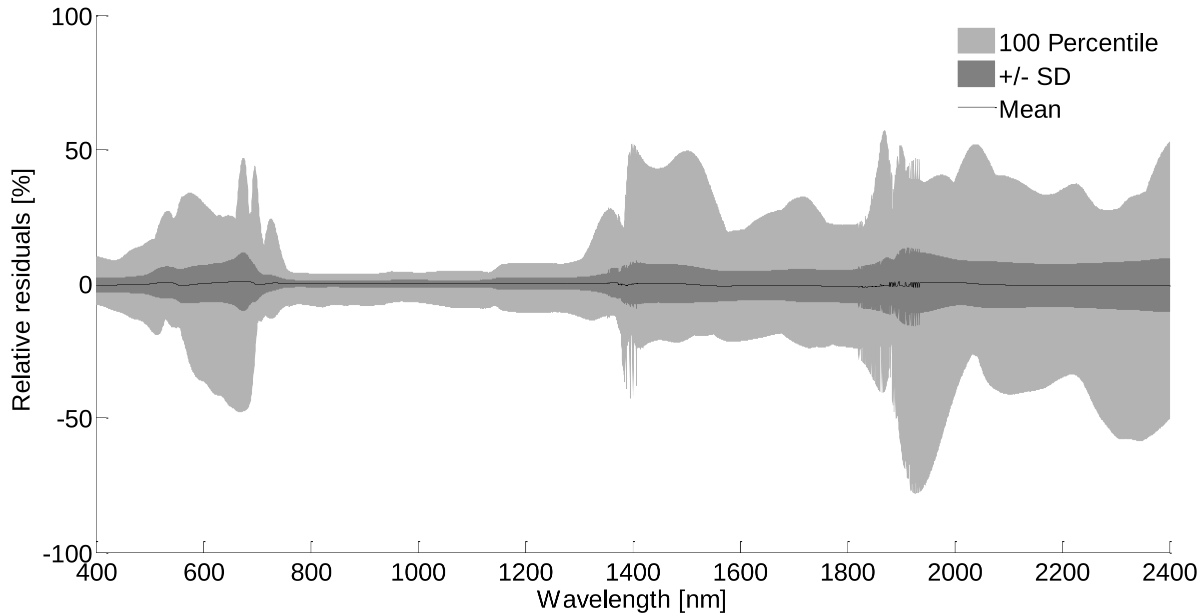
3) Role of LUT size:
#500, #1000



Larger LUT improves accuracy (but takes longer (69 min) and slows down training: ~42 min)

Best performing emulator (NN, #1000, 40PCA)

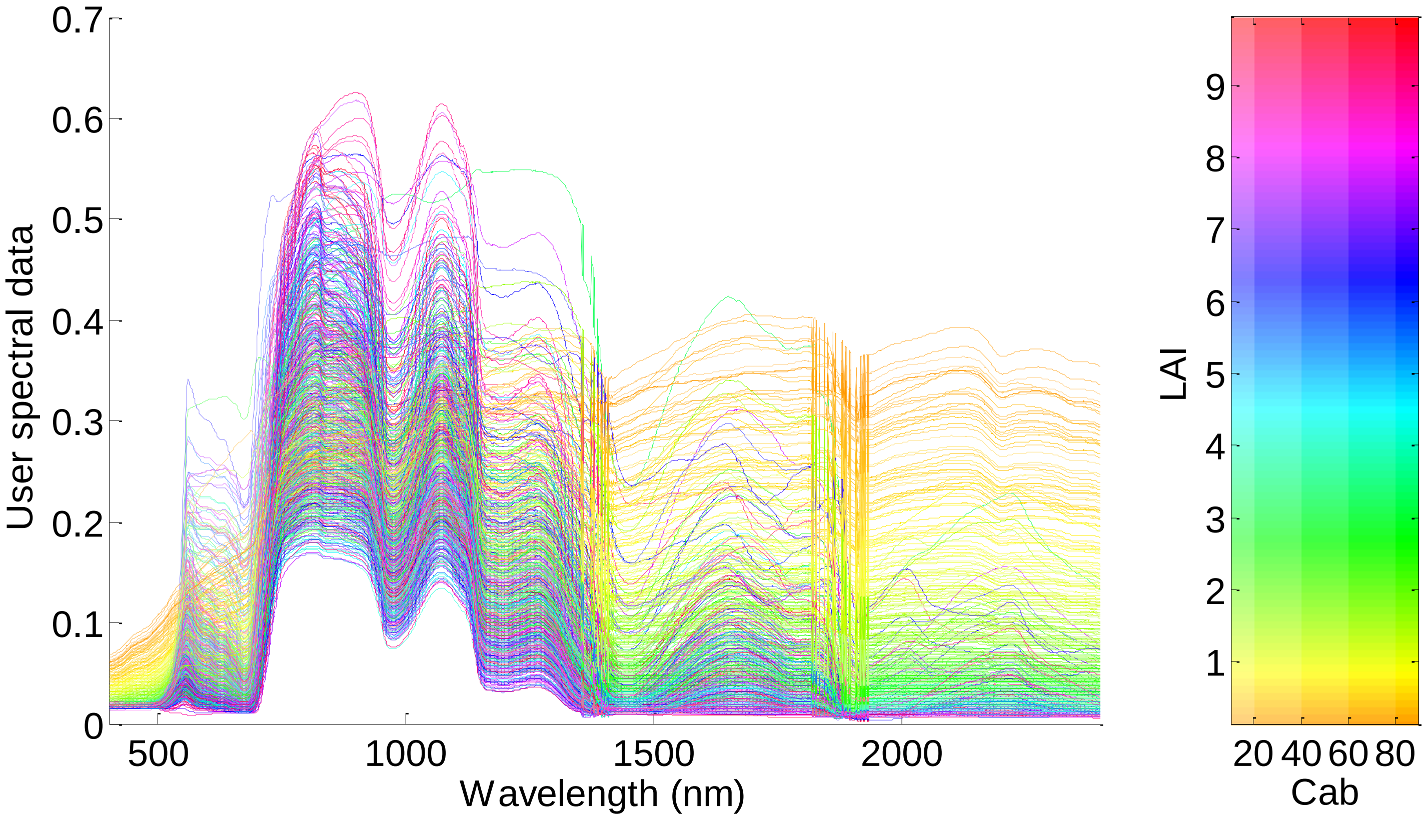




Generation of **1000# random spectra in full parameter space** by best emulator

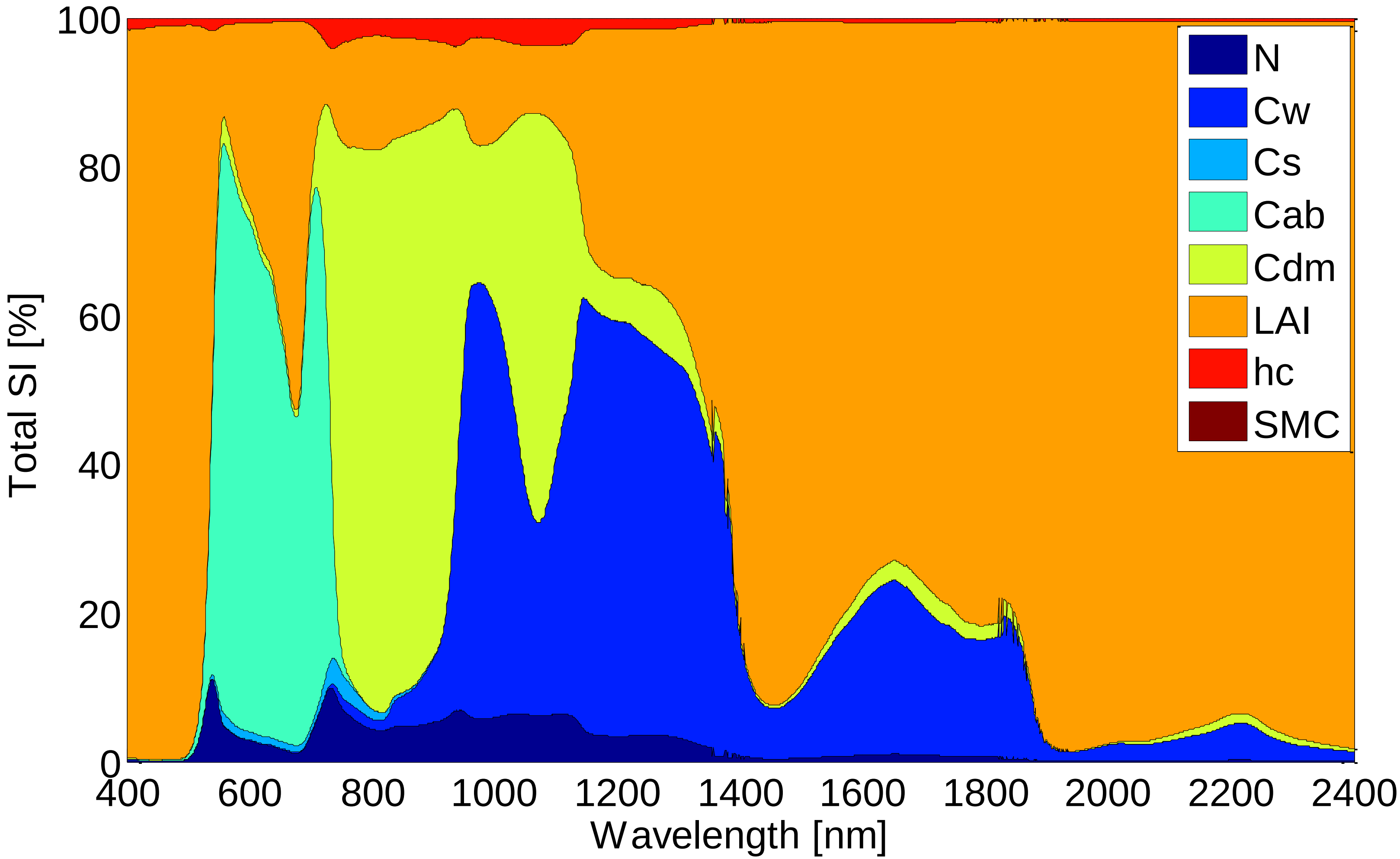
1 min **(SCOPE: 69min)**

- N (1-3)
- LCC (0-100)
- Cw (0-0.5)
- Cdm (0-0.5)
- Cs (0-0.3
- LAI (0-10)
- Hc (0-2)
- SMC (0.-0.7)



GSA SCOPE (NN, #1000, 40PCA)

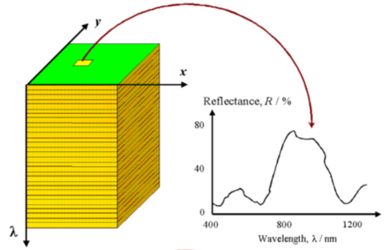
20 sec (SCOPE: hours)

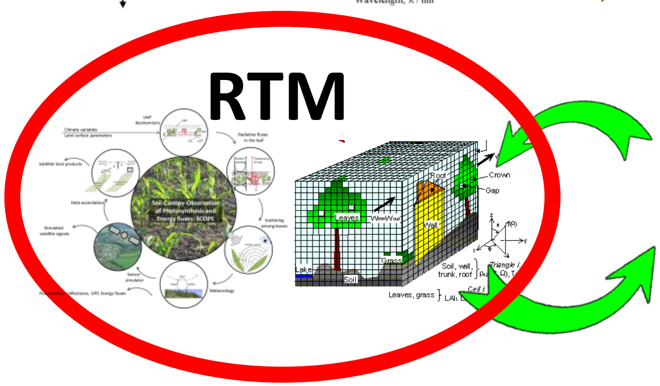


Driving variables can be rapidly identified.

Emulators into numerical inversion

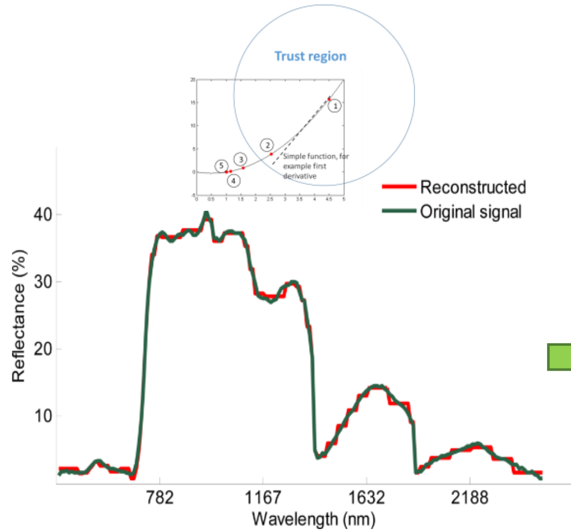
Image



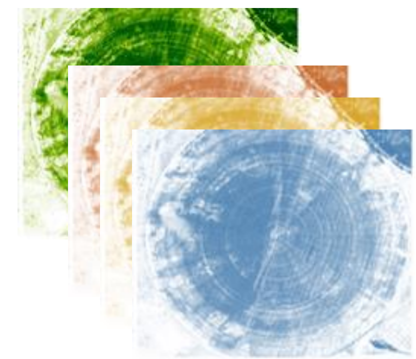


Per-pixel RTM iterations: very slow method, inapplicable to computationally expensive RTMs.

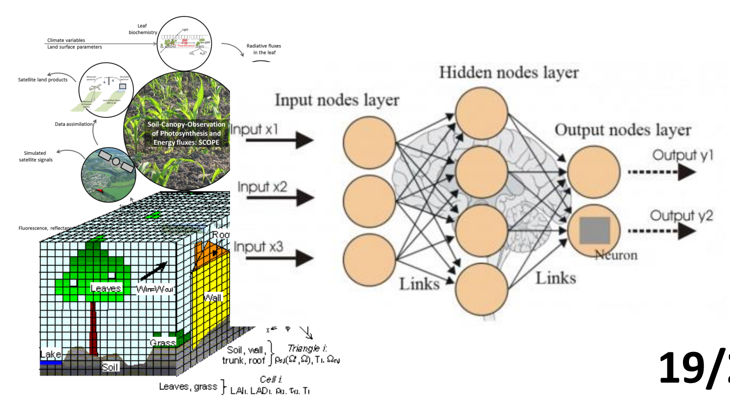
Minimization algorithm: lsqnonlin



Output maps of RTM variables



Emulation of an RTM



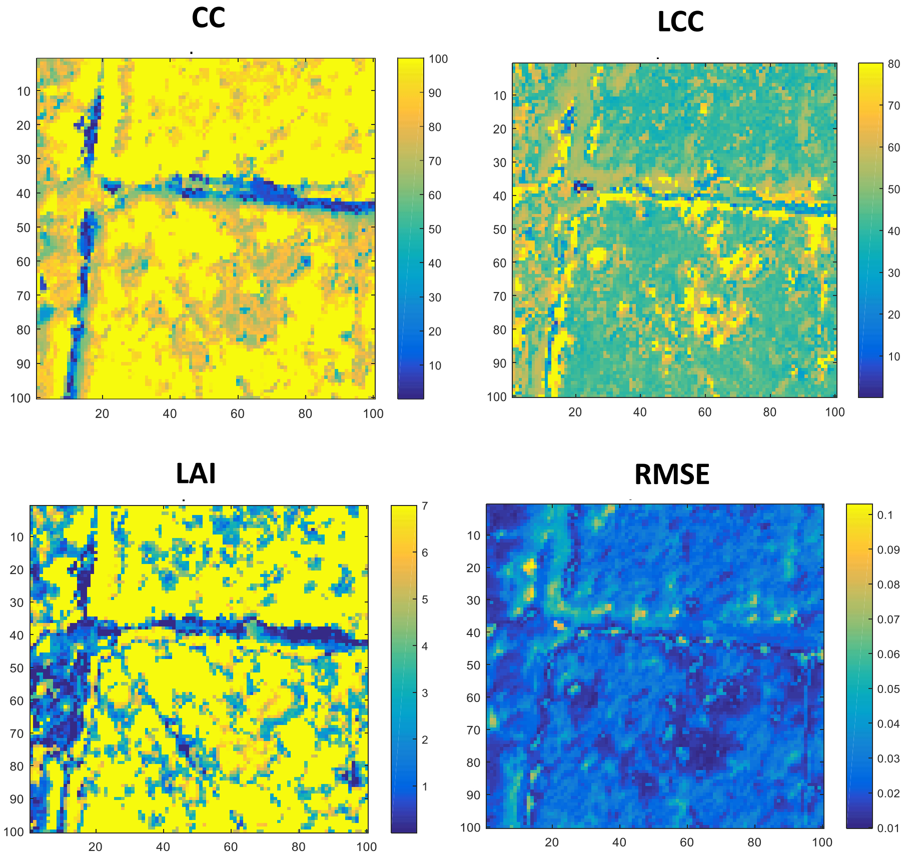
Forest

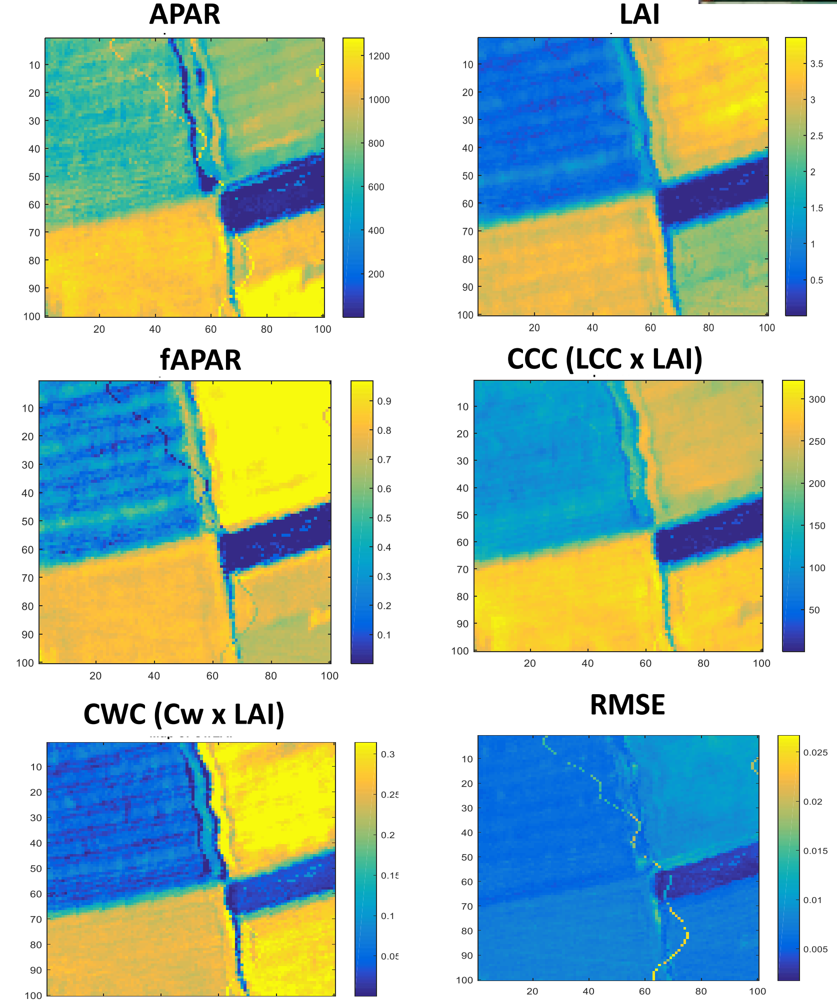
DART KRR emulator applied to HyPlant DUAL (450-2500 nm)

< 1 h

Agriculture

SCOPE KRR emulator applied HyPlant DUAL (bare soil spectra added)





Retrieval quality depends on : (1) emulator, (2) number and type of included variables.

Conclusions

Findings:

❖Thanks to machine learning regression algorithms, **RTMs can be emulated**.

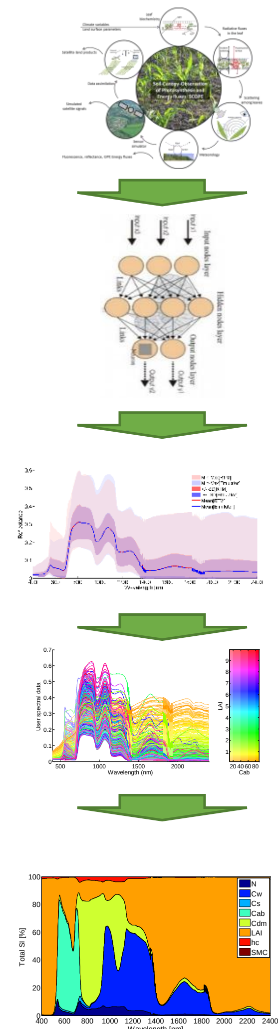
❖**Trade-off: tremendous gain in processing speed** at expense of some loss in accuracy.

❖The following factors determine accuracy:

1. **Nature of RTM data**
2. **Applied MLRA method**
3. **# of components**
4. **LUT size**

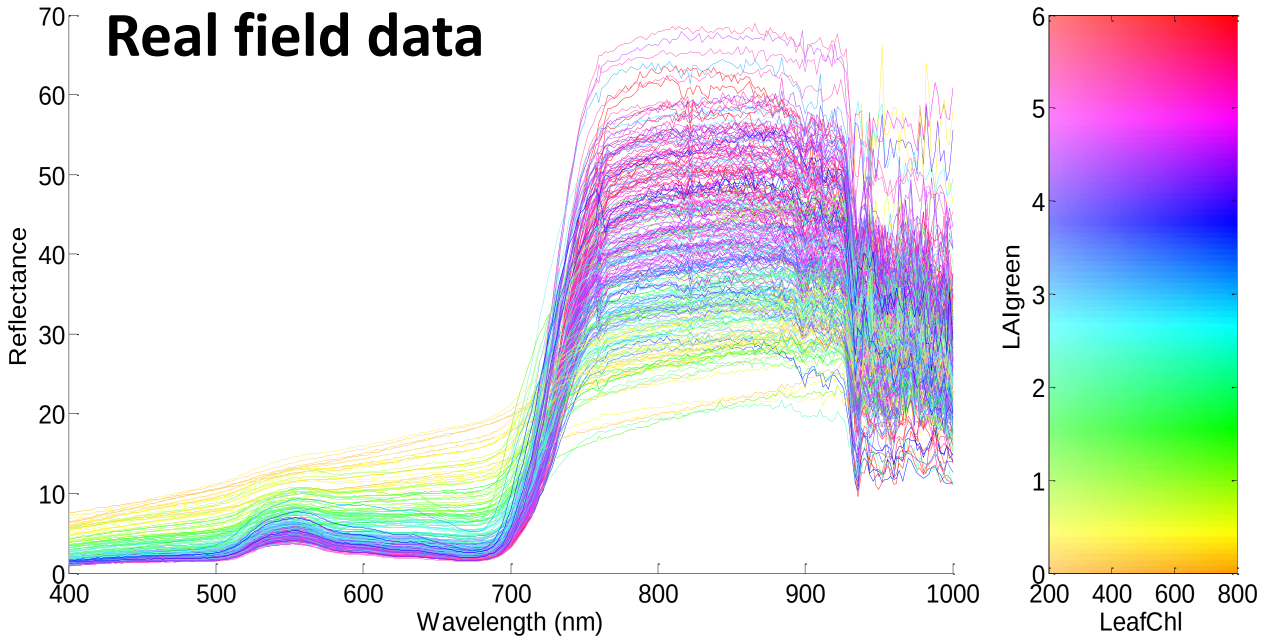
❖**Emulation allows applying advanced RTMs into tedious, operational processing chains**:

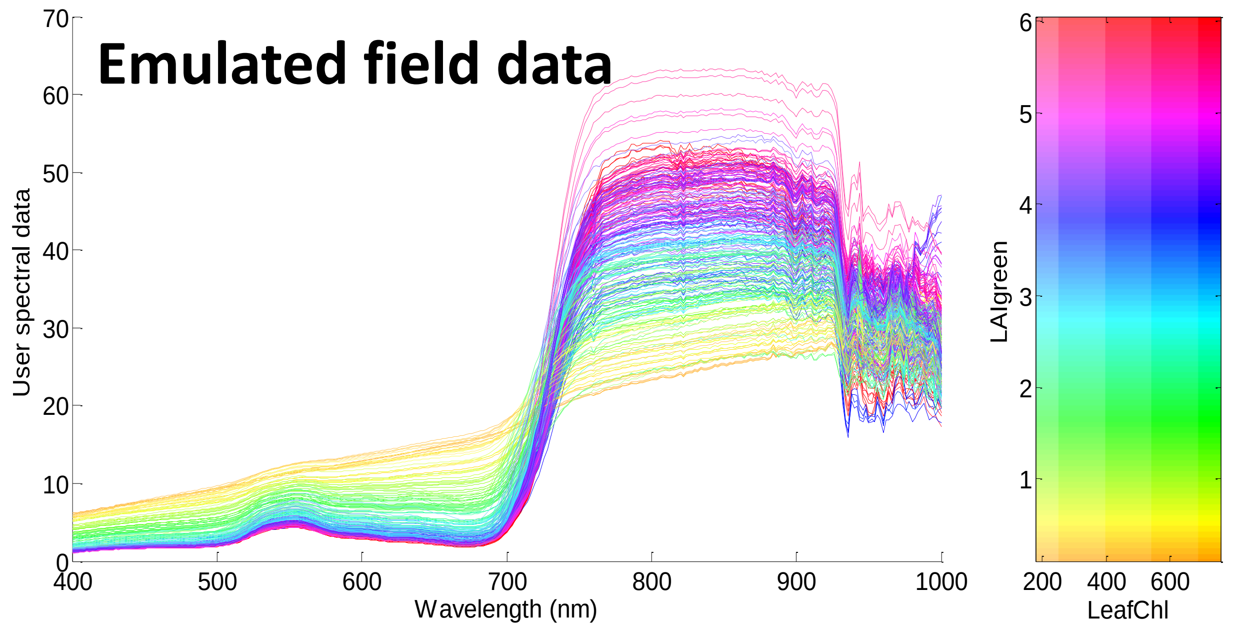
1. **Global sensitivity analysis**
2. **Scene generation, E2E**
3. **Retrieval (numerical inversion)**



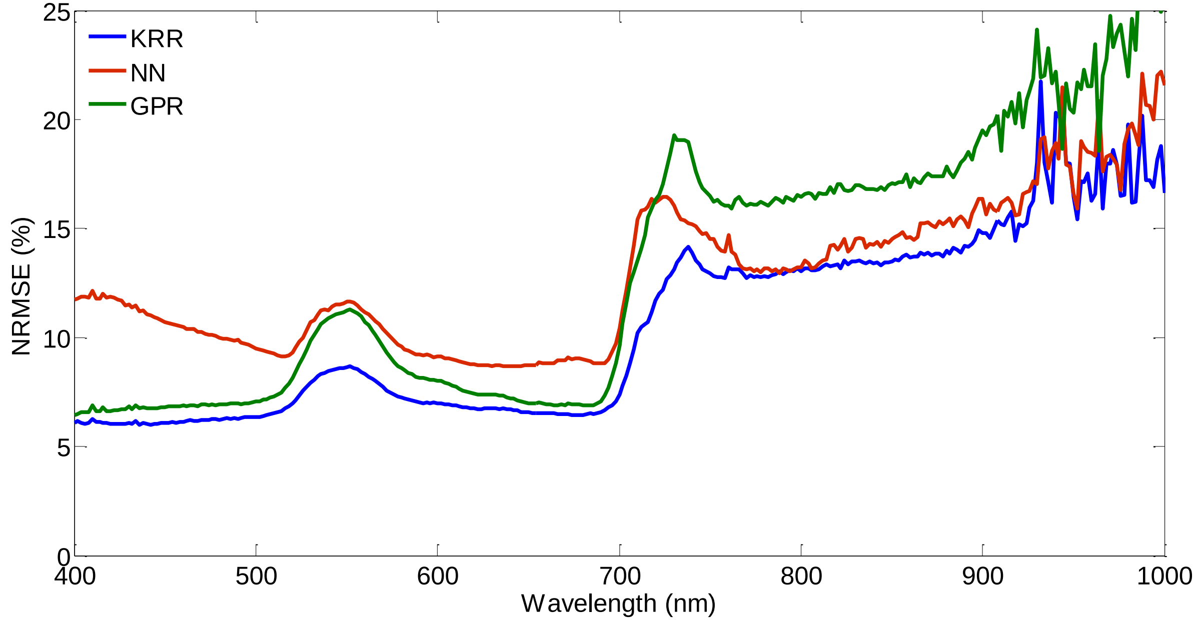
Extra, emulators beyond RTMs

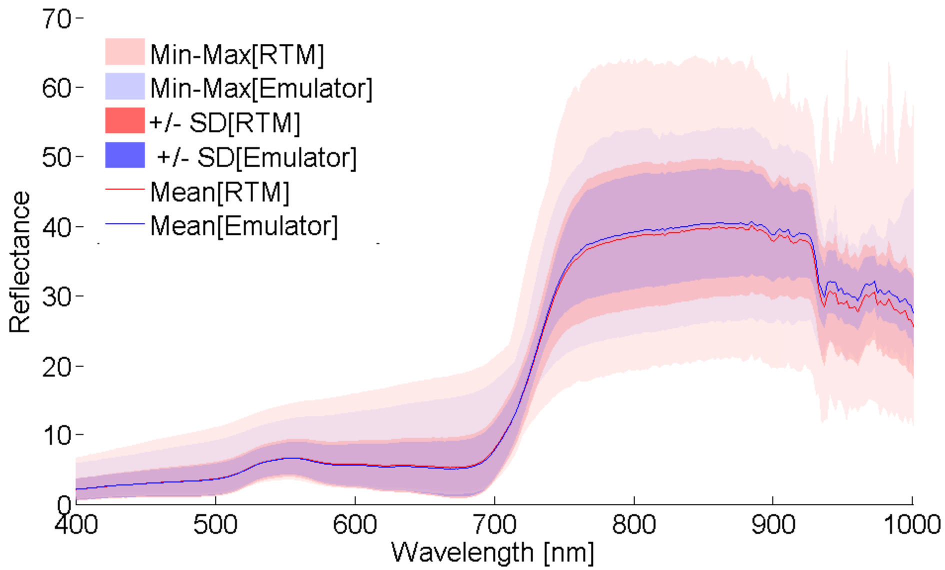
Thanks to Anatoly ☺

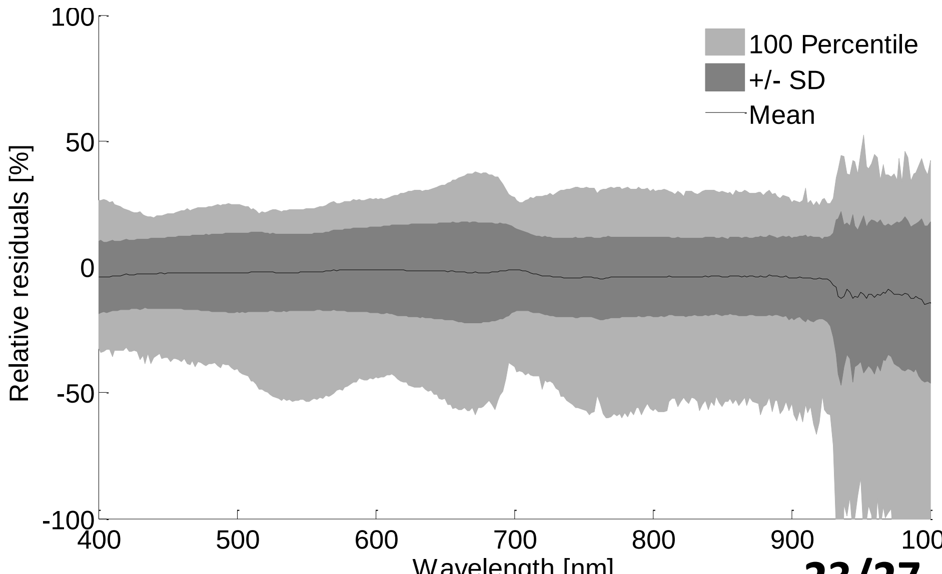




KRR emulator (50PCA, 80/20%)

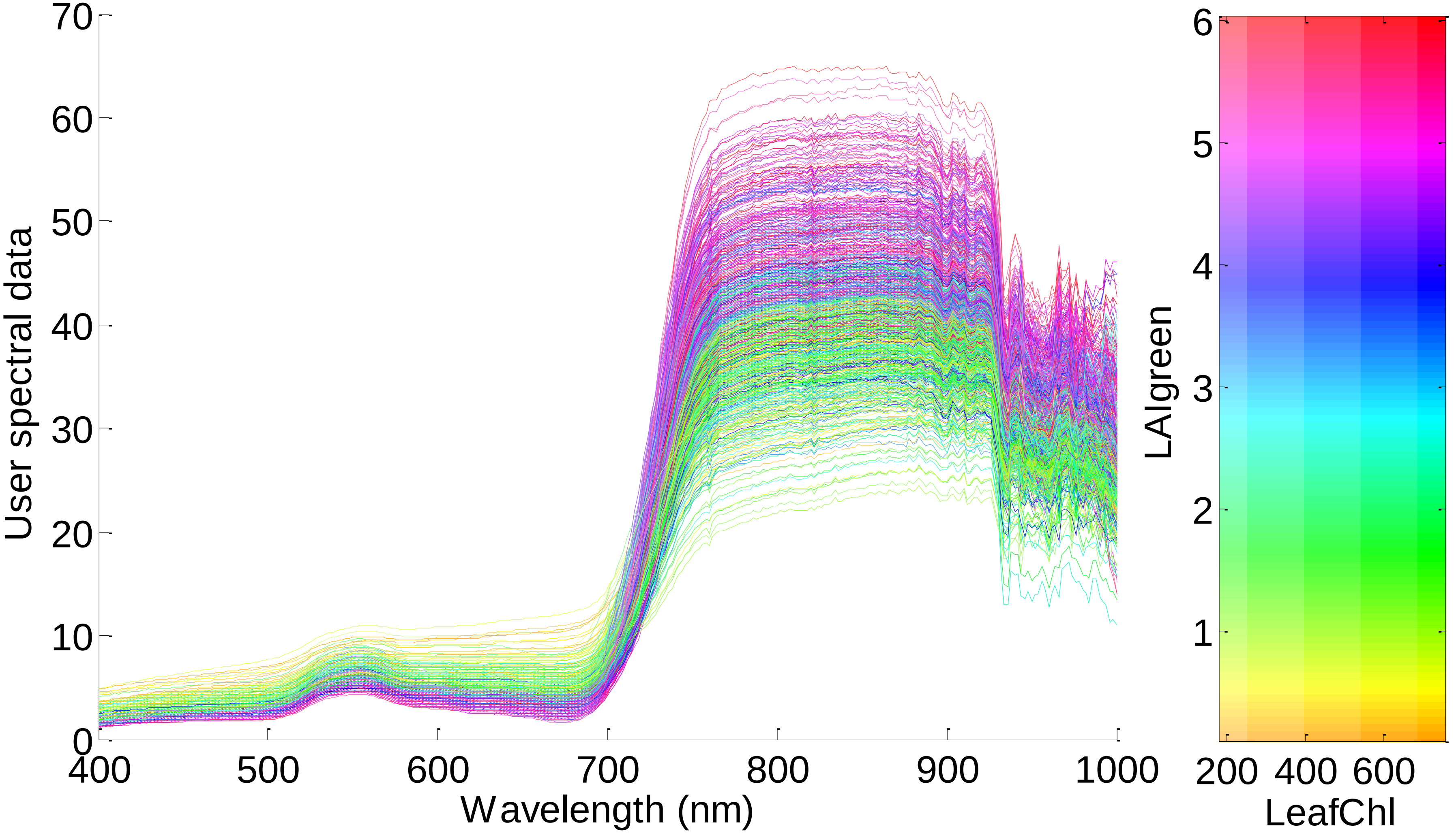






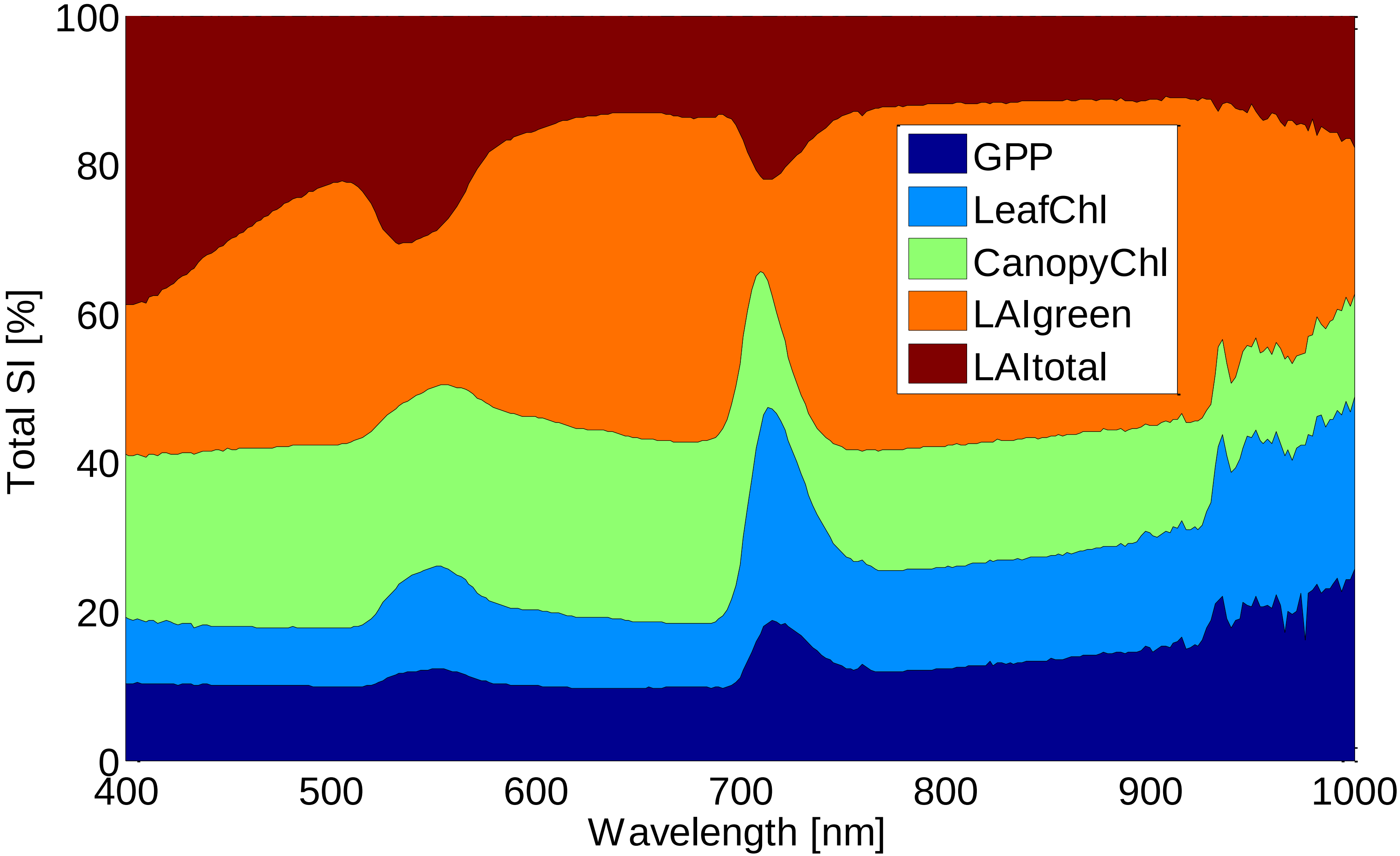
Emulation #1000 spectra

25 s

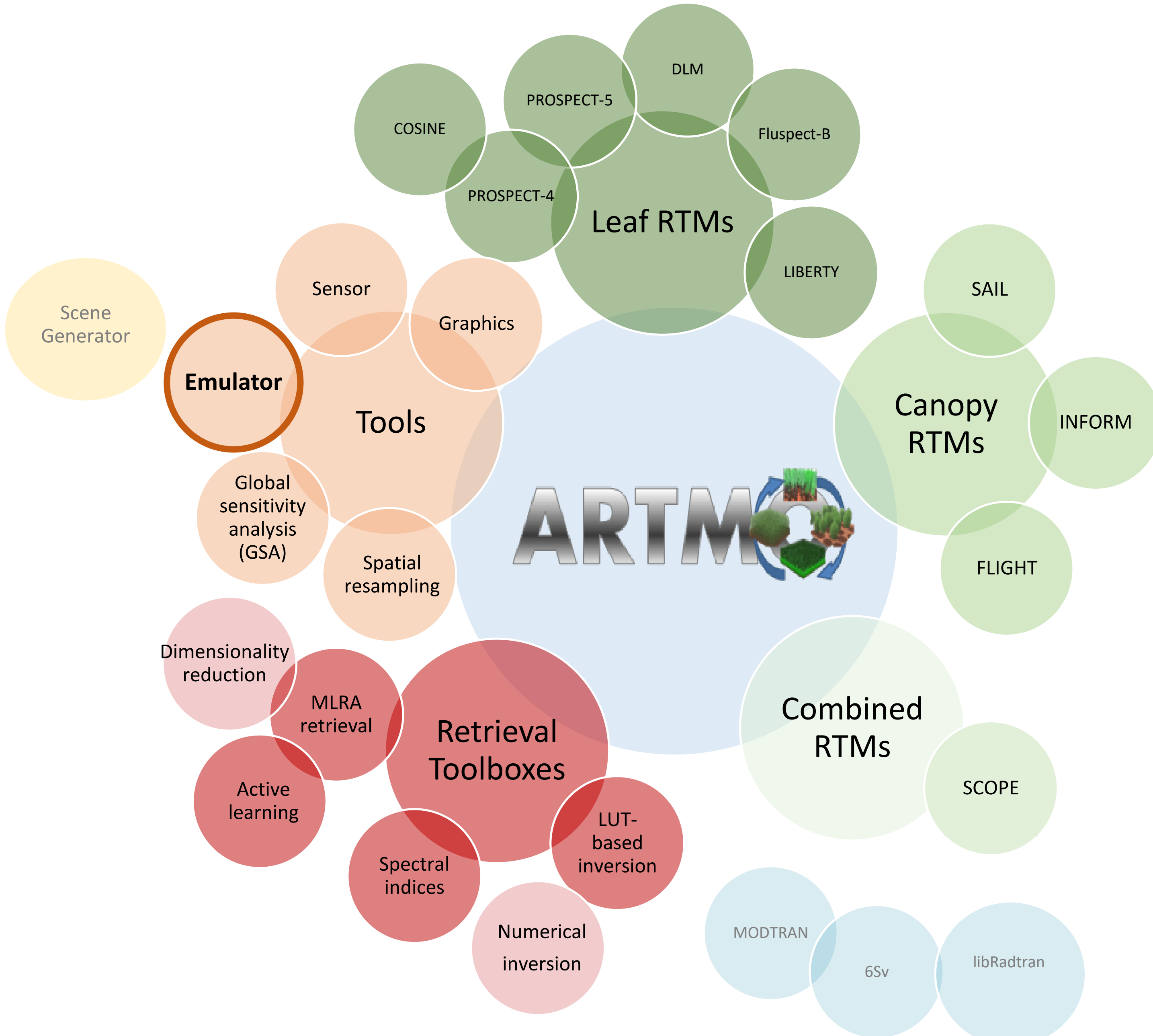


GSA Anatoly dataset

11 s



- **Peak in red edge driven by GPP, LAIgreen and LAItotal**
- **Smaller peak in PRI region**



Thanks

http://ipl.uv.es/artmo/