Statistical methods in global NO_2 mapping

University of Utrecht, The Netherlands

Meng Lu

1. Introduction to NO_2 mapping and data

hands-on 1 (10 mins): “getting ready”
install packages, visualize data, data exploration: paired correlations, spatial dependence and scatterplot

2. Machine learning methods
ensemble trees, deep convolutional neural networks

hands-on 2 + break (45 mins): “machine learning methods.”
R + Python (Kaggle)

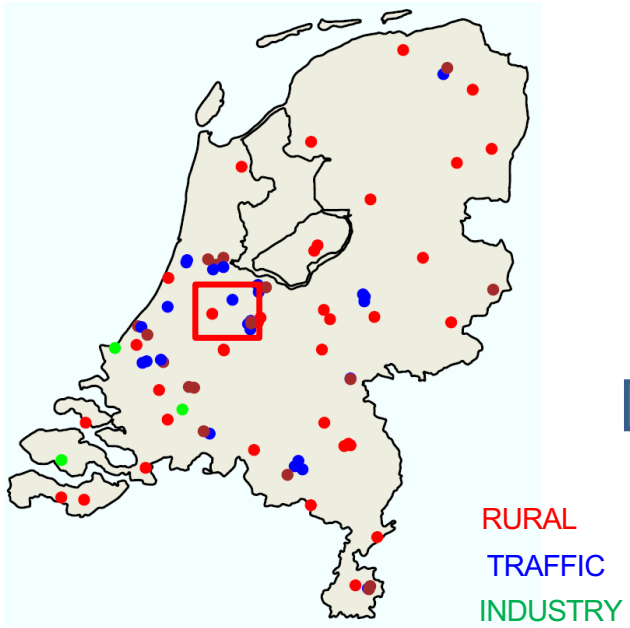
3. Modelling process in practice

hands-on 3 (15 mins): “modelling process”
hyper-parameter optimization, bootstrapping-based cross-validation, mapping and compare spatial patterns of different predictions

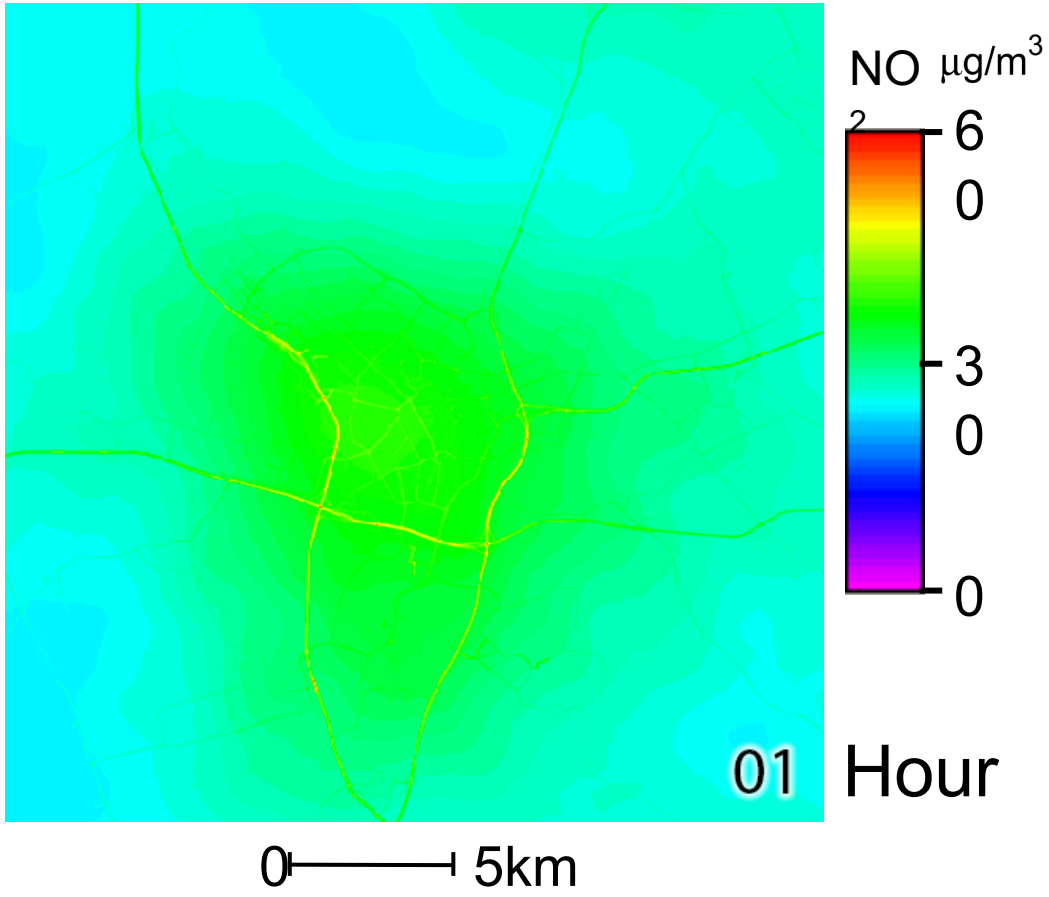
Part 1

Introduction and data

NO_2 mapping



Ground stations for NO_2 monitoring



Land use regression (LUR)

Predicting air pollution and analyzing the sources.

$$y = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \cdots + \beta_n x_n$$

Ground station measurements

Geospatial predictors

Development:

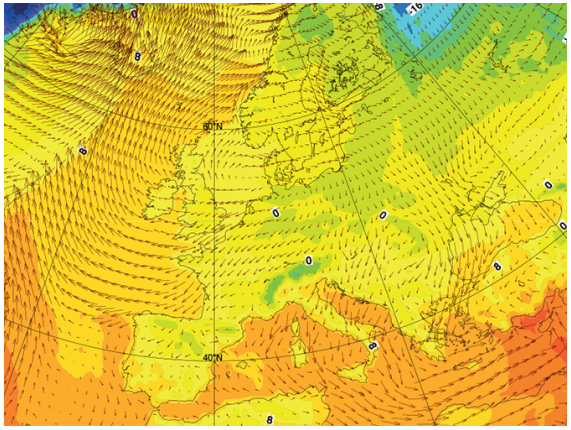
- Integrating spatiotemporal dependency
- General Additive Models
- Machine learning models
- Hierarchical models

Geospatial predictors

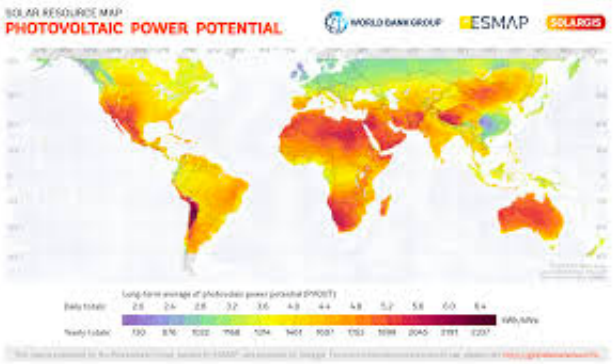
Road and industrial area from OpenStreetMap (vector)

Earth nightlight (500 m), VIIRS (visible infrared imaging radiometer suite)

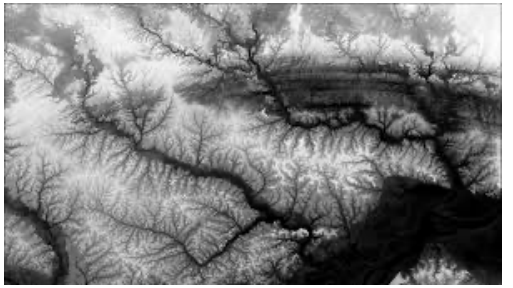
Population (250 m)
Human settlement layers



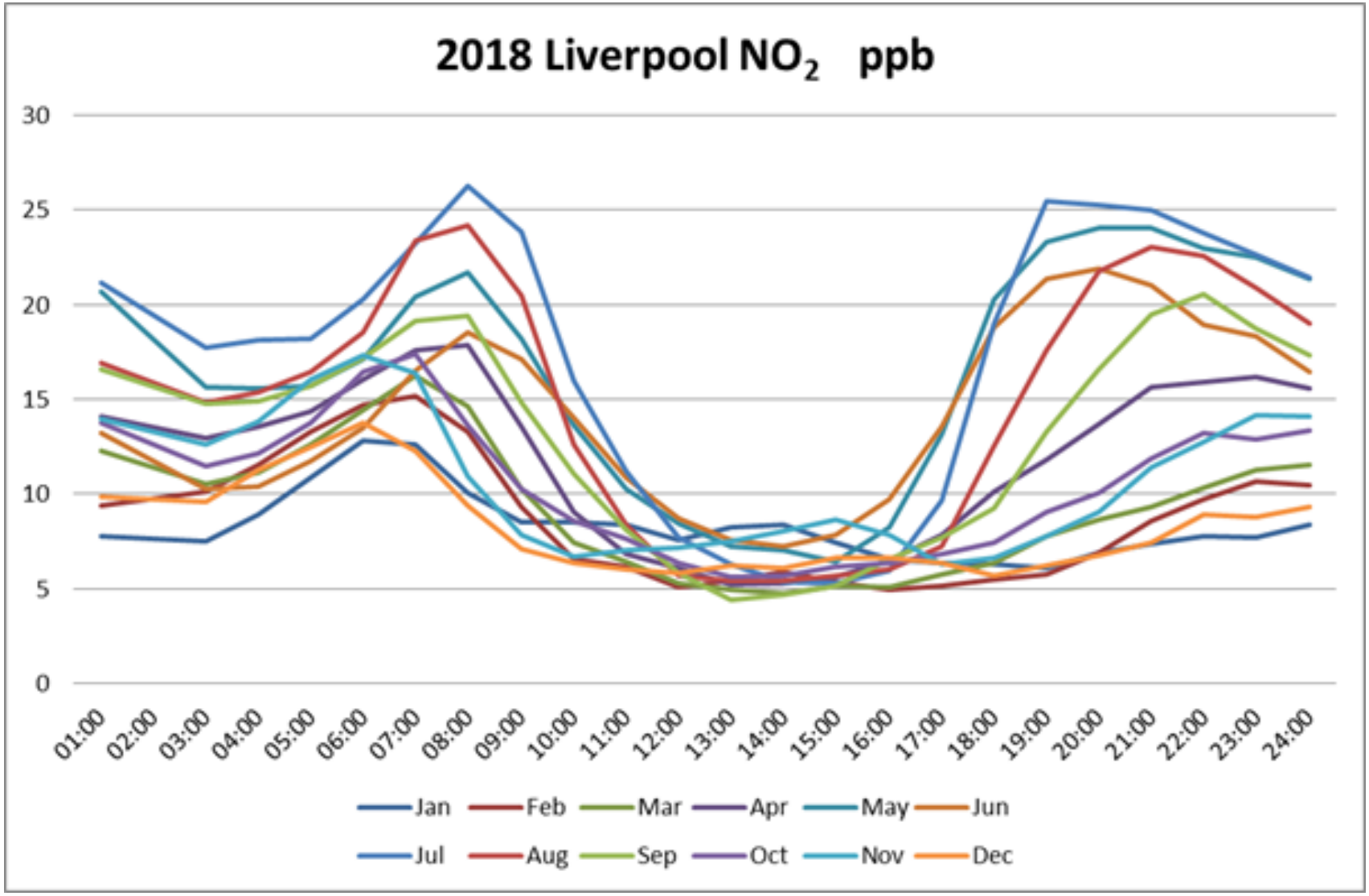
Wind speed and temperature
ECMWF (European Centre for Medium-Range Weather Forecasts) climate reanalysis
ERA5-land



Solar radiation
Numerical model (Solargis)



Elevation: SRTM (90 m)

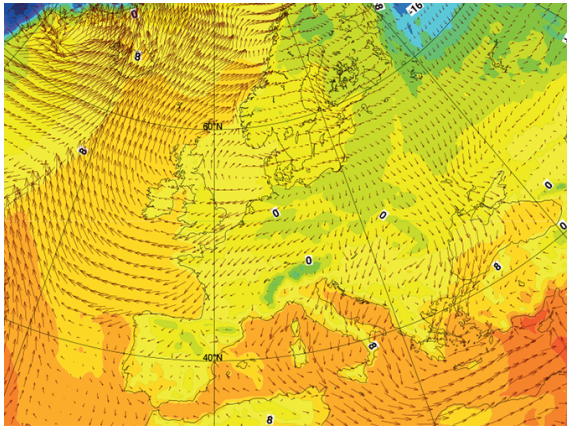

2018 Liverpool NO2 ppb
30
25
20
15
10
5
0
01:00
02:00
03:00
04:00
05:00
06:00
07:00
08:00
09:00
10:00
11:00
12:00
13:00
14:00
15:00
16:00
17:00
18:00
19:00
20:00
21:00
22:00
23:00
24:00
Jan
Feb
Mar
Apr
May
Jun
Jul
Aug
Sep
Oct
Nov
Dec

Geospatial predictors

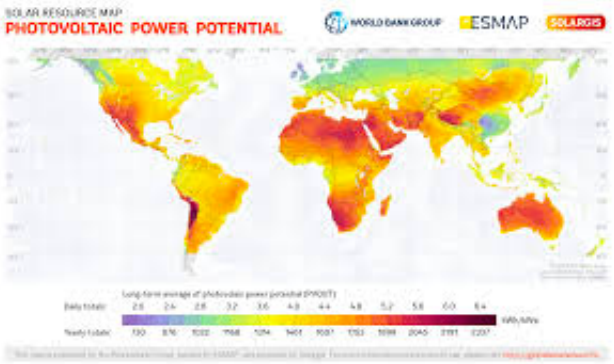
Road and industrial area from OpenStreetMap (vector)

Earth nightlight (500 m), VIIRS (visible infrared imaging radiometer suite)

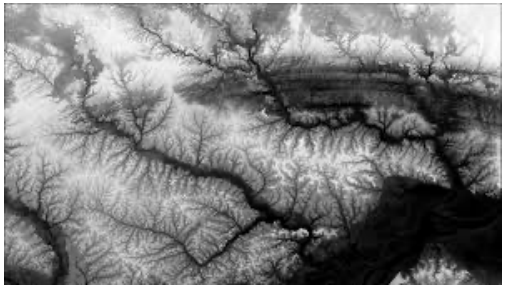
Population (250 m)
Human settlement layers



Wind speed and temperature
ECMWF (European Centre for Medium-Range Weather Forecasts) climate reanalysis
ERA5-land



Solar radiation
Numerical model (Solargis)

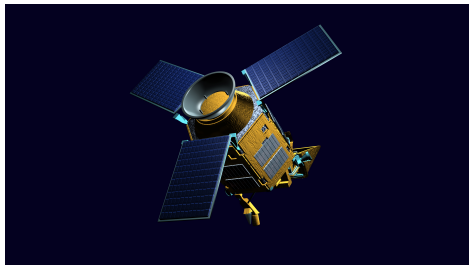


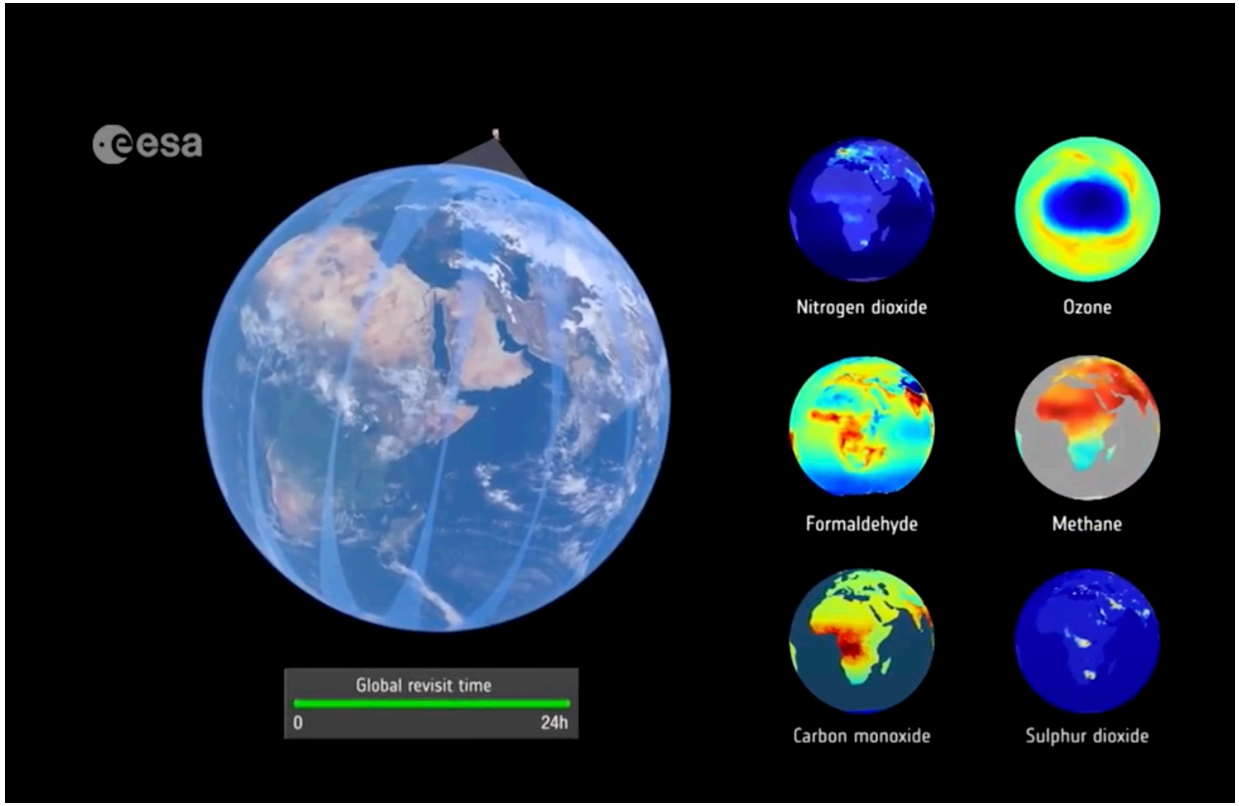
Elevation: SRTM (90 m)

Tropomi: Sentinel 5p
(TROPOspheric Monitoring Instrument)

Launched Oct. 2017,

Resolution (NO2)
2018-2019: 7 km
Since Oct. 2019: 5.5 km





Measurements: NO_2, O_3 (7km × 28km), SO_2, methane and CO

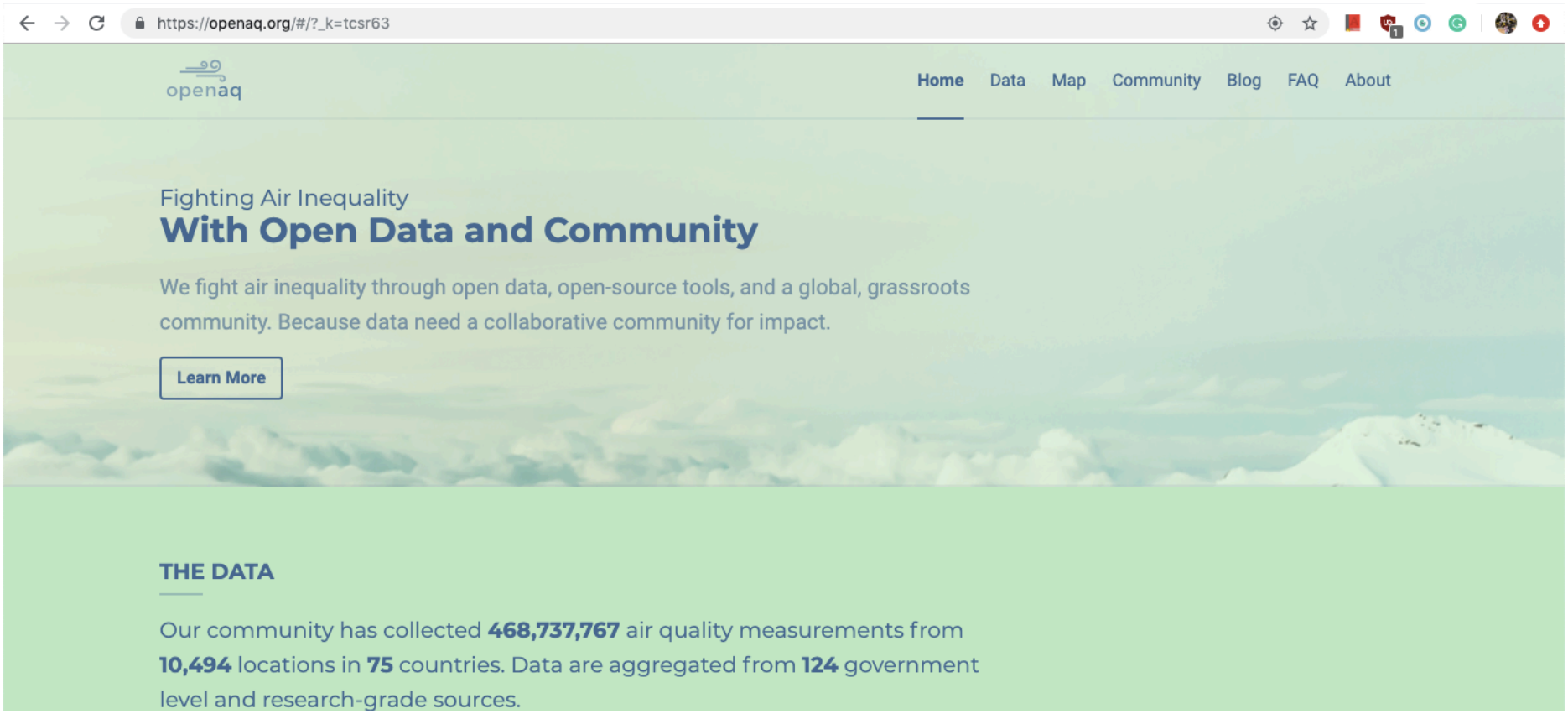
Surface concentration product based on remote sensing and GEOS-CHEM (2012)

- Chemical transportation models: GEOS-CHEM

- Surface NO2 product made using GEOS-CHEM together with GOME, SCIAMACHY, and GOME-2 satellite instruments. http://fizz.phys.dal.ca/~atmos/martin/?page_id=232

Geddes, J. A., R.V. Martin, B. L. Boys, and A. van Donkelaar, **Long-term trends worldwide in ambient NO2 concentrations inferred from satellite observations for exposure assessment**, *Environ. Health Perspec.*, DOI:10.1289/ehp.1409567, 2015

Ground monitor Data: OpenAQ (2017)



Hands-on 1: Setup and data visualization

In the folder R_scripts

- Data_visualize/Openaq.Rmd:
  query station measurements from OpenAQ API

- Data_visualize/ Plot_RS.Rmd:
  visualize TROPOMI, OMI satellite measurements, and the remote sensing-based surface concentration product.

- Introduction/GeoHub2020.Rmd:
  run ”Section 1” for data exploration

Part 2

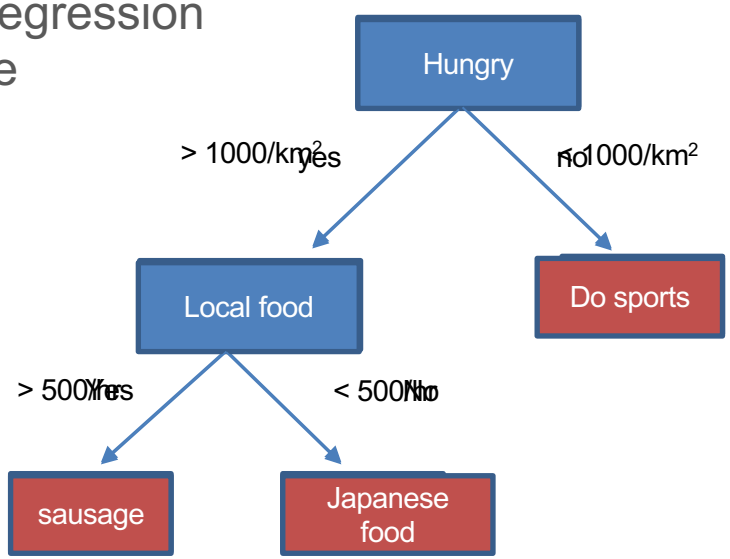
Machine learning

1940s	1950s	1970s – 1980s,	1990s			1997	2006
Walter Pitts	Turing test	AI winter	boosting,	Deep Blue, chess	IBM Jeopardy	LSTM	Deep learning

Statistical learning
--Statistical modelling and uncertainty

Decision and regression trees

A regression tree



Features:

- Non-parametric
- Different kinds of variables
- Redundant variables are ignored
- Handle missing data
- Small trees are easy to interpret

Ensemble trees

- Bagging
- Random forest
- Boosting

Dominance
Boosting > Randomforest > bagging > single tree

Random forest

Variance reduction

Identically distributed variables, each has variance σ^2

An average of B of i.i.d random variables has variance $\frac{1}{B}\sigma^2$

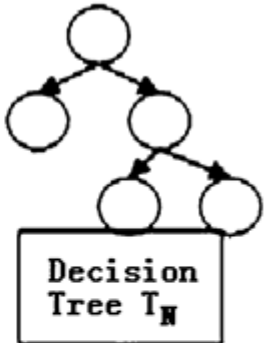
If the variables are not independent (but identically distributed) with positive pairwise correlation ρ, the variance of the average:

$$\rho\sigma^2 + \frac{1-\rho}{B}\sigma^2$$

"the more uncorrelated, the more you bringing down the variance".

(tunning parameter: number of trees, tree depth)

Random forest



For each tree:

1. Bootstraping sample D* from the training data D
2. Draw **m*** variables randomly from all variables m, pick the best split-point (variable), split the node.

Limitation:
Not based on a hypothesis test for spliting point, does not assess uncertainty

Variations:

Recursive partition trees:
Hypothesis testing of dependency between variables and resursively fitting the splitting weight

Baysian based sampling and variable selection:
Baysian framework for 1 and 2

Quantile random forest:
Estimate quantiles (beyond the conditional mean)

Random forest

Hyperparameters:

Number of trees
Number of variables
Minimum observations at the leaf (terminal) node.

R_scripts/Introduction/shiny_randomforest/app.R

Gradient Boosting

-- Reweight based on the previous trees, stage-wise fitting

Each successive tree is built for the prediction residuals of the preceding tree in an adaptive way to reduce bias.

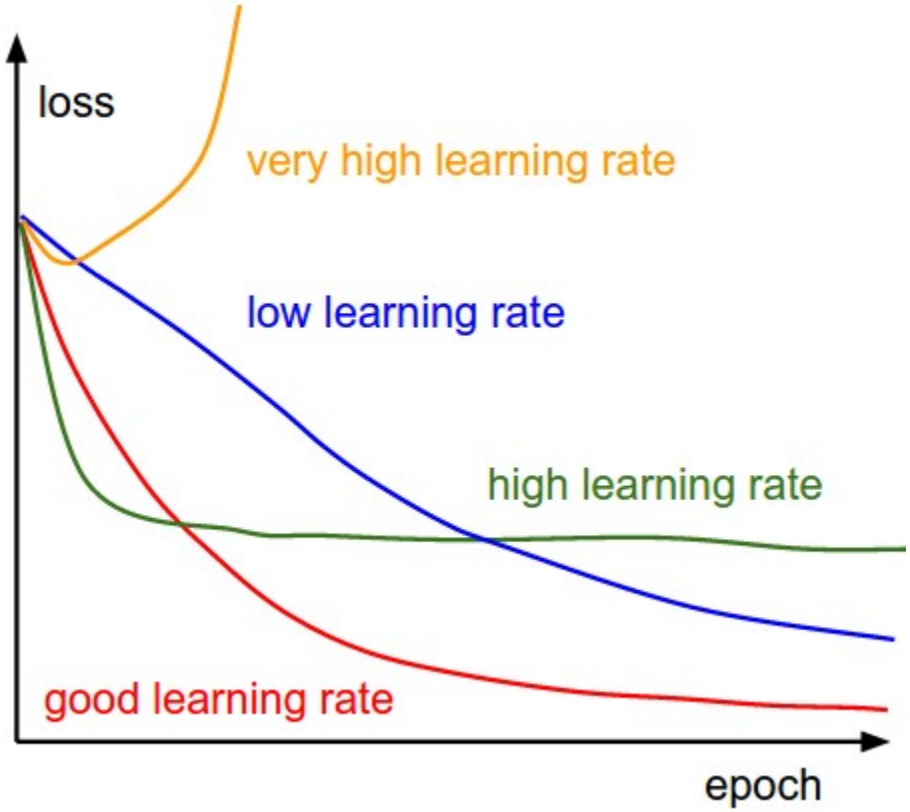
.

	Algorithm 1: Gradient_TreeBoost	
1	$F_0(\mathbf{x}) = \arg\min_\gamma \sum_{i=1}^{N} \Psi\left(y_i, \gamma\right)$	
2	For $m = 1$ to M do:	
3	$\tilde{y}_{im} = -\left[\frac{\partial \Psi(y_i, F(\mathbf{x}_i))}{\partial F(\mathbf{x}_i)}\right]_{F(\mathbf{x})=F_{m-1}(\mathbf{x})}, \; i = 1, N$	Caculate the gradient
4	$\{R_{lm}\}_1^L = L - \text{terminal node } tree(\{\tilde{y}_{im}, \mathbf{x}_i\}_1^N)$	Fit a tree
5	$\gamma_{lm} = \arg\min_\gamma \sum_{\mathbf{x}_i \in R_{lm}} \Psi\left(y_i, F_{m-1}(\mathbf{x}_i) + \gamma\right)$	Calculate residuals
6	$F_m(\mathbf{x}) = F_{m-1}(\mathbf{x}) + \nu \cdot \gamma_{lm} 1(\mathbf{x} \in R_{lm})$	Update the approximation
7	endFor	

approach the gradient of the loss function (e.g. binomial, logistic, poison) by trees.

Each consecutive tree is built for the prediction residuals (from all preceding trees) of an independently drawn random sample

Learning Rate



Stochastic Gradient Boosting

Each consecutive tree is built for the prediction residuals (from all preceding trees) of an independently drawn random sample

XGBoost
Exteme gradient boosting

Idea
Not only impurity, but also model complexity

$$\text{Cost}(\theta) = \text{Loss}(\theta) + \Omega(\theta)$$

Features

- Parallel computation
- Objective function includes model complexity

- Support dense and sparse matrix
- Can costomize objective fucntions

Regularization

Penalizing the number of leaves: only add nodes if the "gain" is larger than a certain value.

Penalizing large values of leaves:

L1 norm regularization (Lasso path) on values of the terminal nodes

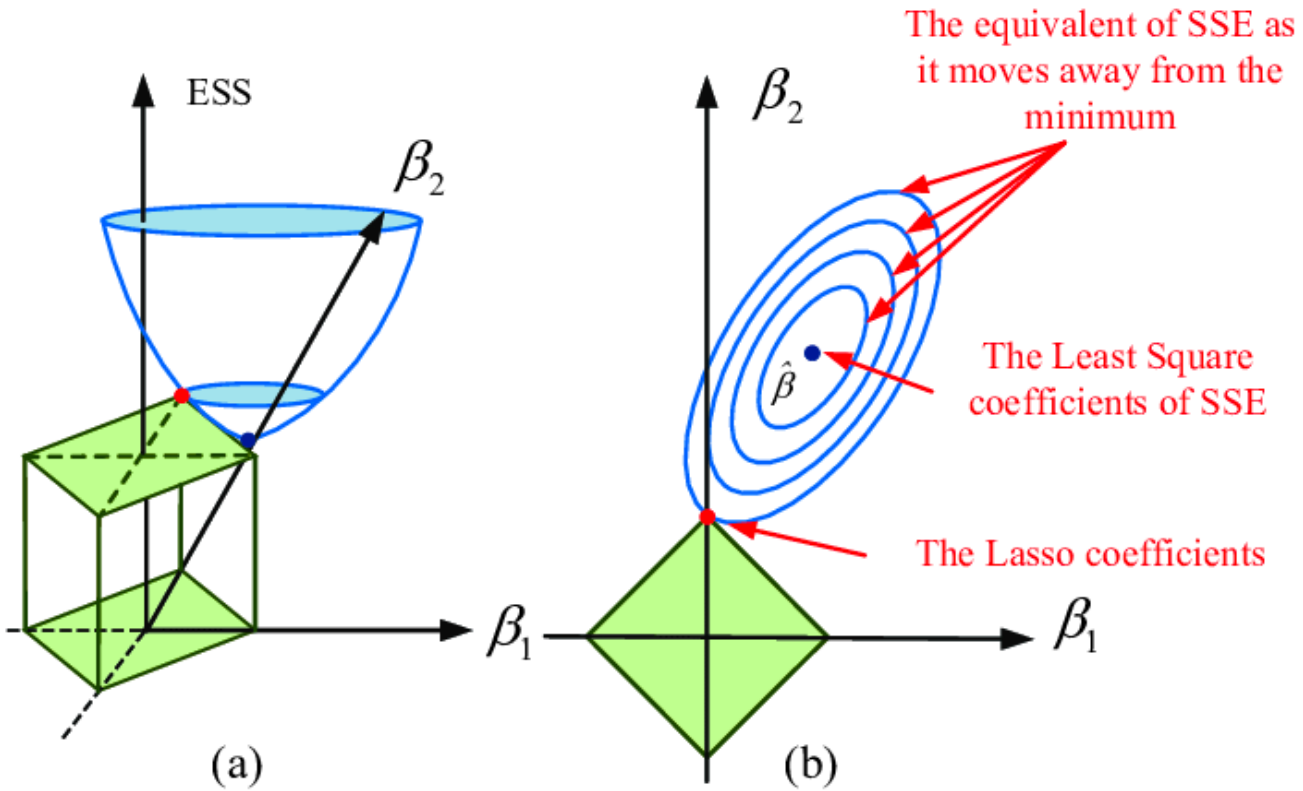
$$\mathrm{C}(F) = \sum_{1}^{N} L(yi, F(xi)) + \alpha \sum_{1}^{N} |F(xi)|$$

L2 norm regularization (Ridge path) on values of the terminal nodes

$$\mathrm{C}(F) = \sum_{1}^{N} L(yi, F(xi)) + \lambda \sum_{1}^{N} \big(F(xi)\big)^2$$

Isosphere view of L1 norm

$$C(F) = \sum_{1}^{N} L(yi, F(xi)) + \alpha \sum_{1}^{N} |F(xi)|$$



	L1 norm (Lasso) $\sum_{1}^{N} \lvert F(xi) \rvert$	*L2 norm (Ridge)* $\sum_{1}^{N} \big(F(xi)\big)^2$	
Robust			Resistance to outliers
Stability			Resistance to data adjustment
Computational difficulty			
Feature selection			

Control over-fitting: Post-processing

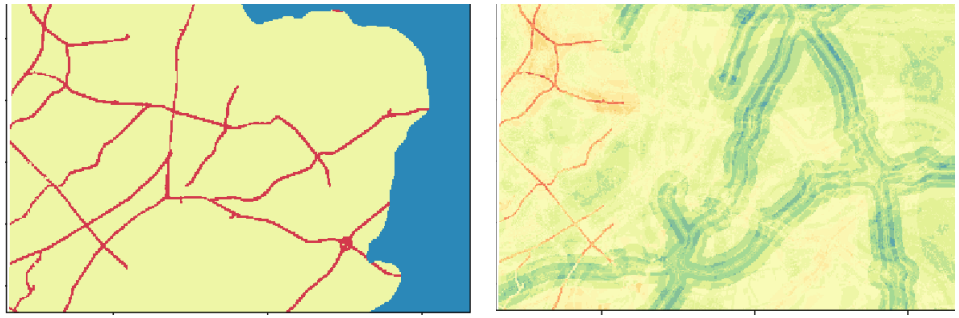
Aggregate trees using Lasso

$$\alpha(\lambda) = \arg\min_{\alpha} \sum_{i=1}^{N} L[y_i, \alpha_0 + \sum_{m=1}^{M} \alpha_m T_m(x_i)] + \lambda \sum_{m=1}^{M} |\alpha_m|$$

Hands-on 2 (1)

R_scripts/Introduction/GeoHub2020.Rmd:

Shiny apps:
R_scripts/Introduction/shiny_randomforest/app.r
R_scripts/Introduction/shiny_xgboost/app.r

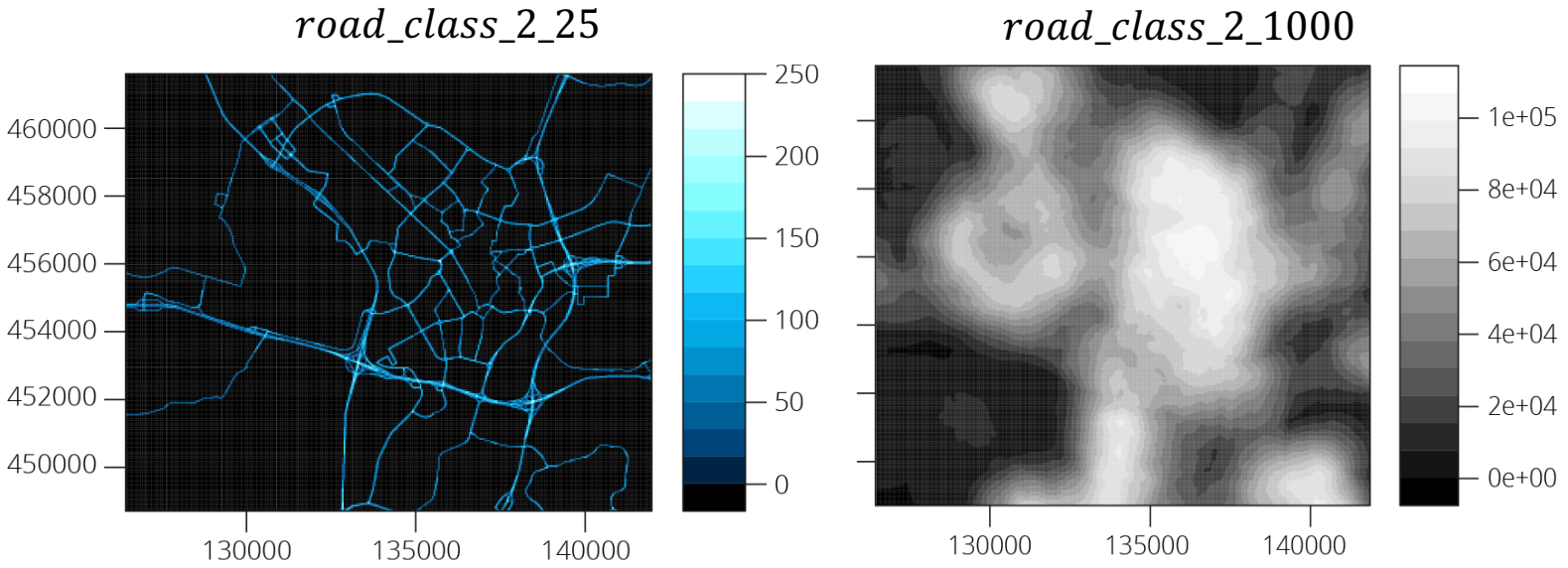


1 tree

100 trees

https://lumeng0312.shinyapps.io/xgboost/?_ga=2.179522658.79817579.1592385947-986486774.1592216474

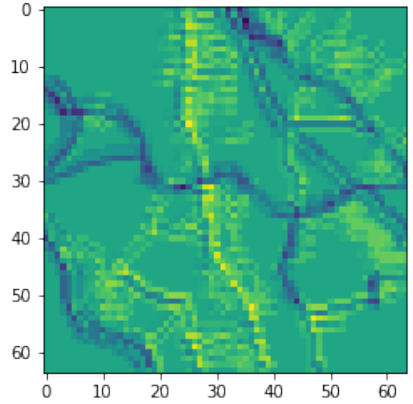
Automatic feature extraction



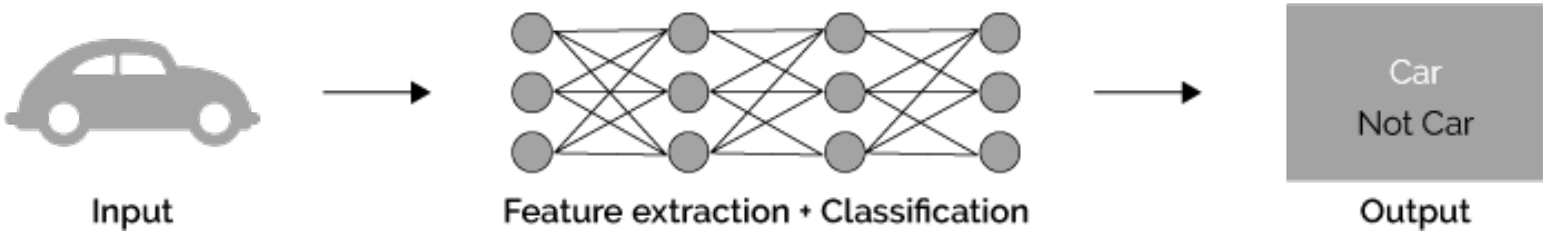
Automatic feature extraction

Road length within a buffer

Transportation networks

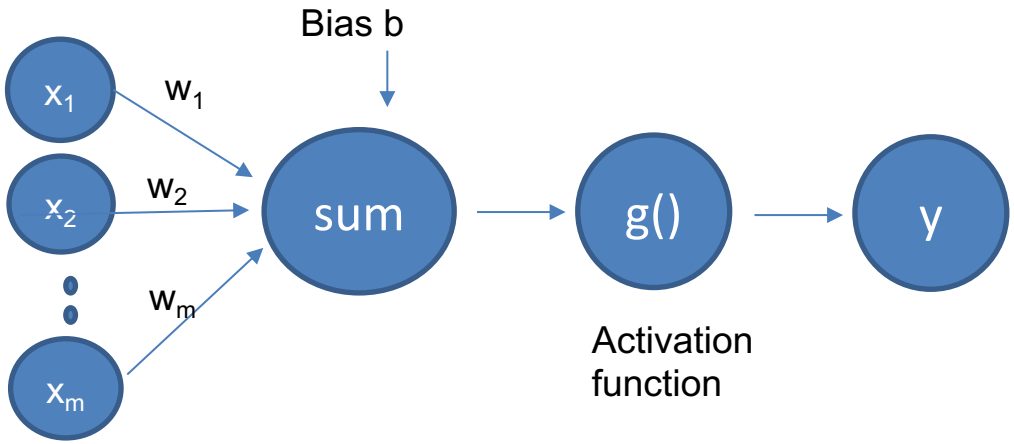


Deep Learning



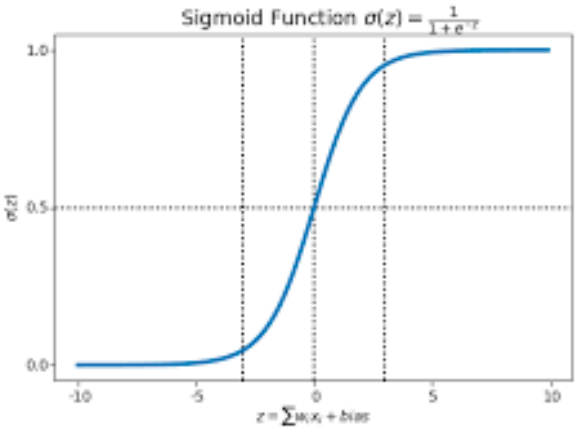
Deep learning: a neuron

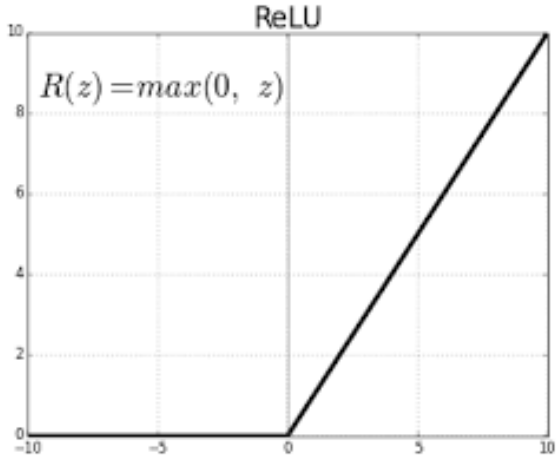
A single neuron: perceptron



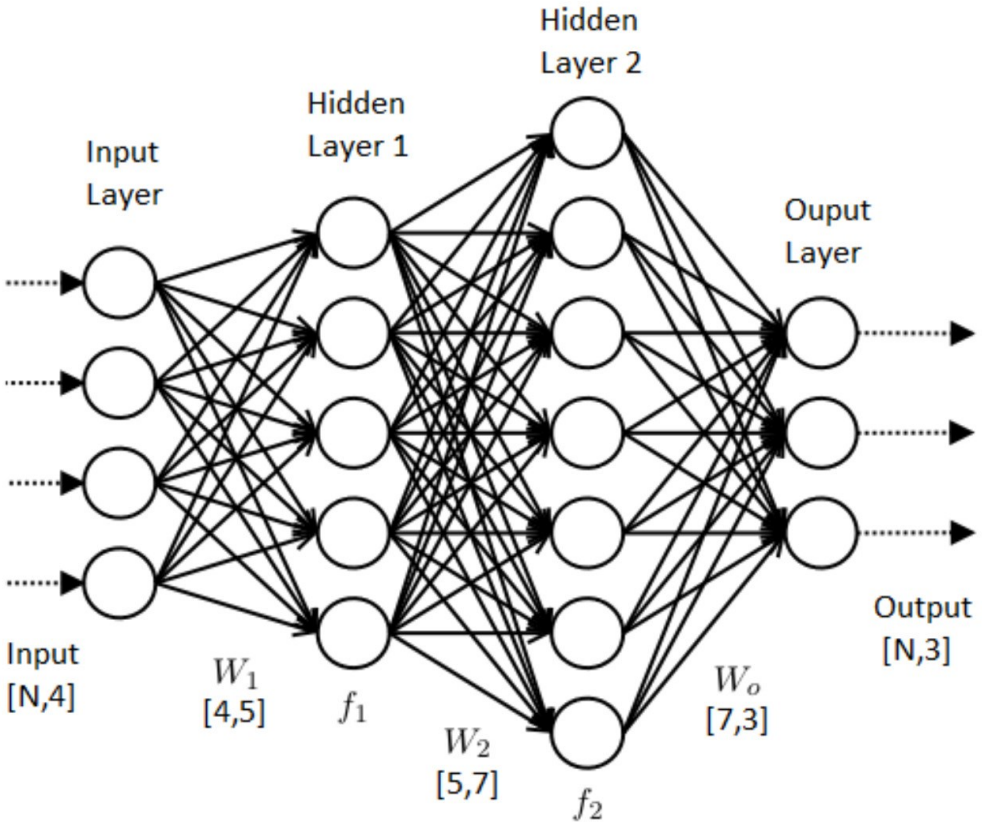
$$y = g(\sum_i^n (w_i X_i) + b)$$

sigmoid





Deep learning: neural networks



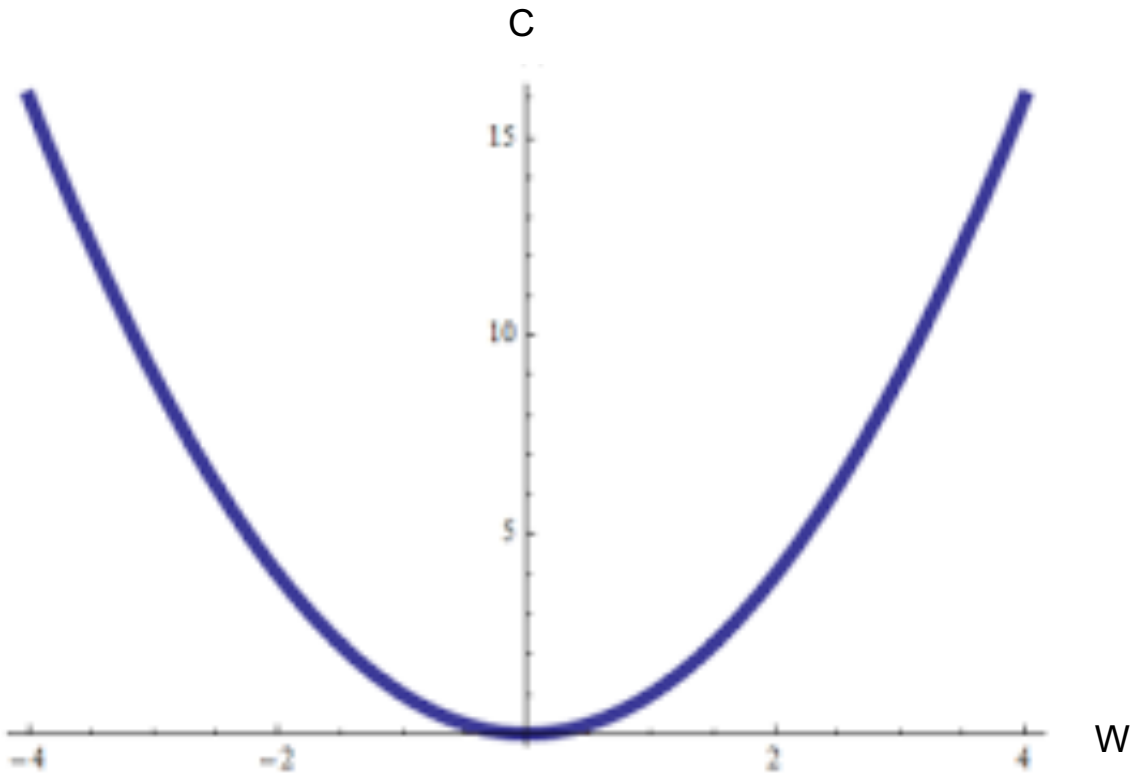
Learning: optimizing the cost

$$Cost\ function\colon\ C(W) = \frac{1}{m}\sum_{1}^{m} L(f(x_i, W), yi)$$

Regression:
Mean Squared Error

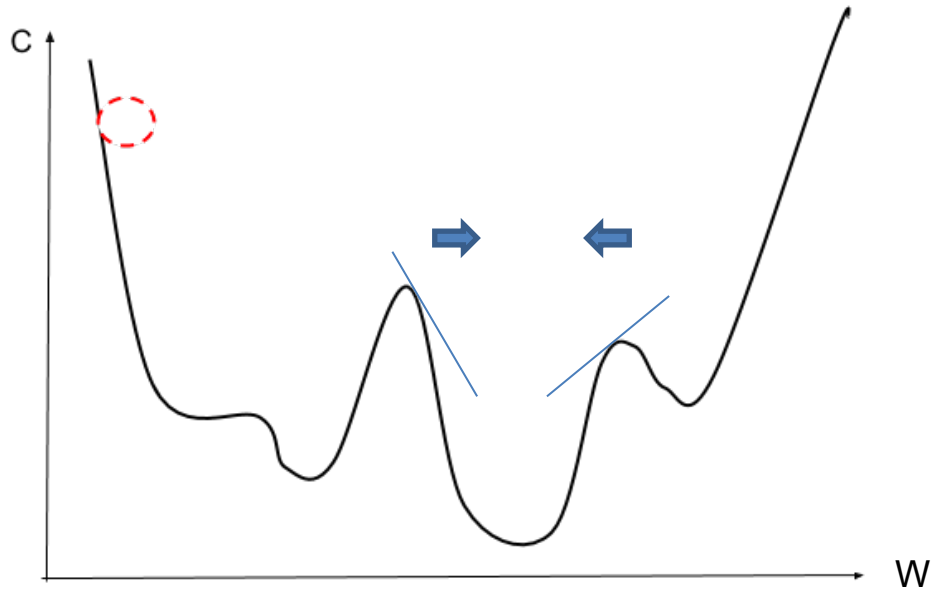
Classification:
Cross-entropy

Gradient Descent



$$\frac{\partial C(W)}{\partial W} = 0$$

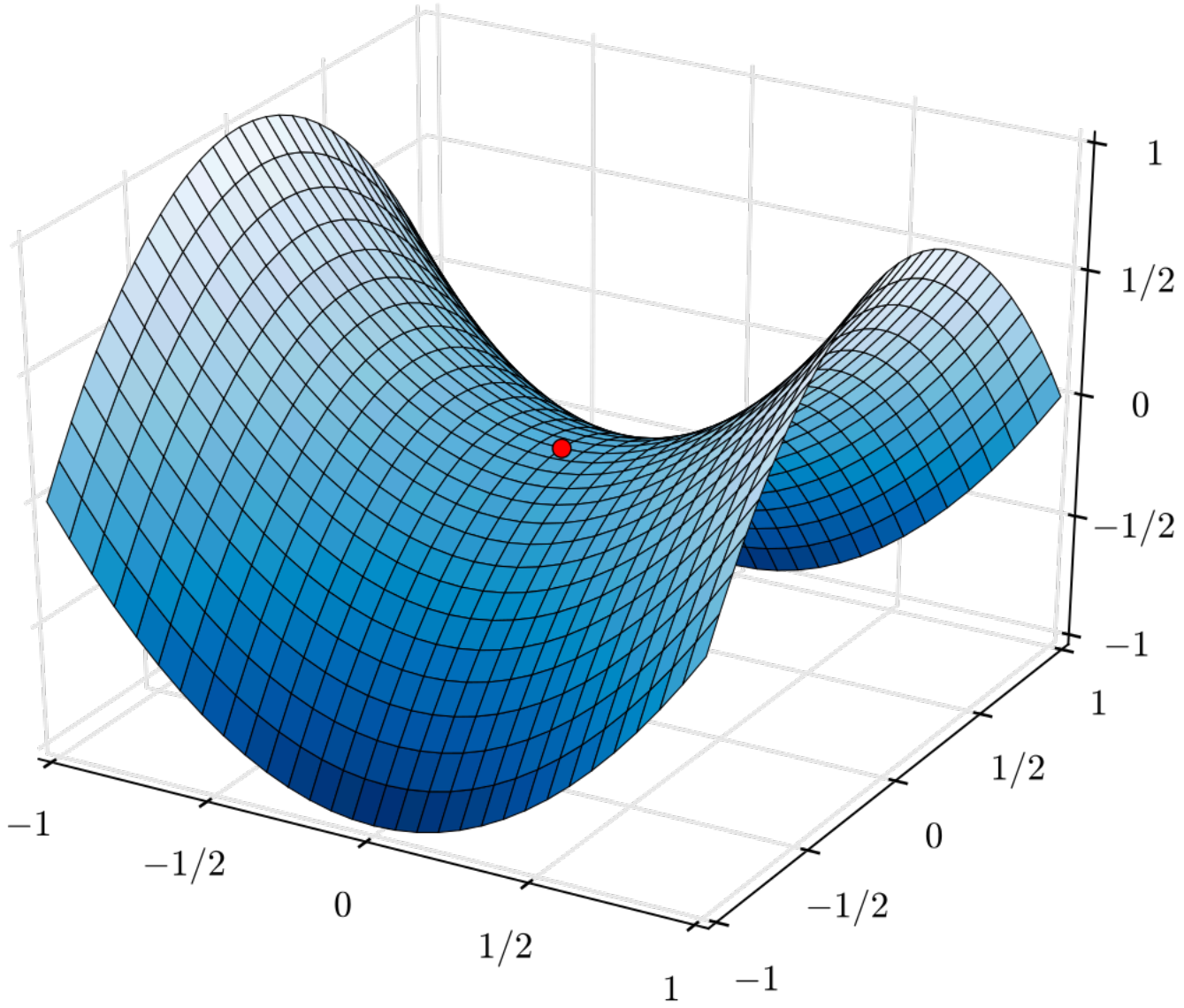
Gradient Descent



Stochastic Gradient descent
Mini-Batch Gradient descent

$$W = W - \Upsilon \frac{\partial C(W)}{\partial W}$$

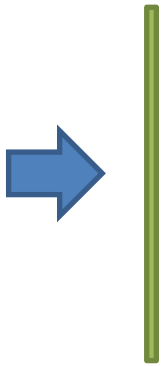
Saddle point

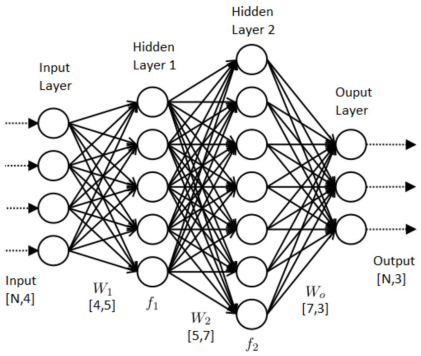


Regularization

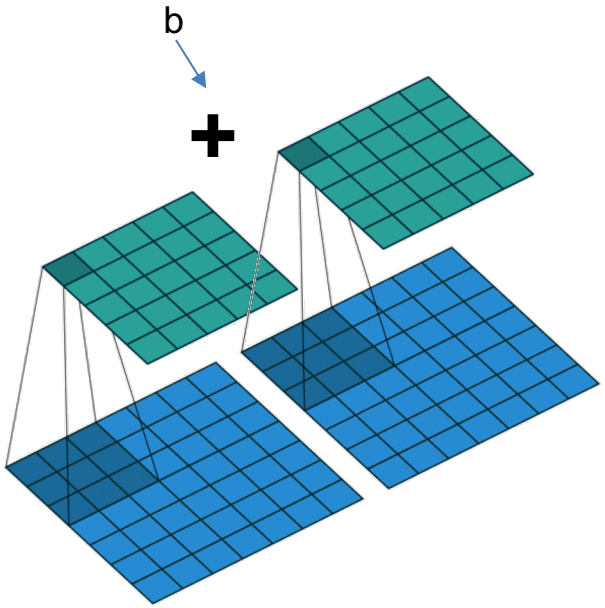
Ridge and Lasso paths (L1 and L2 norms)
Early stopping
Drop-out
Batch normalization

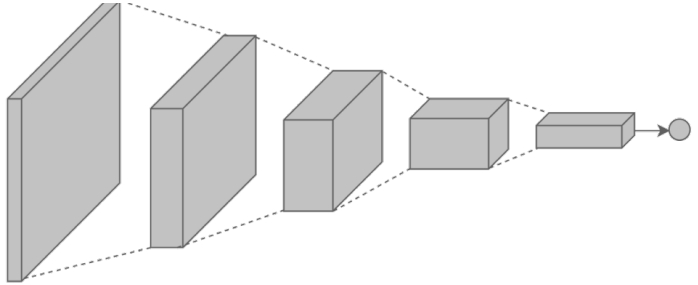
Convolutional neural networks





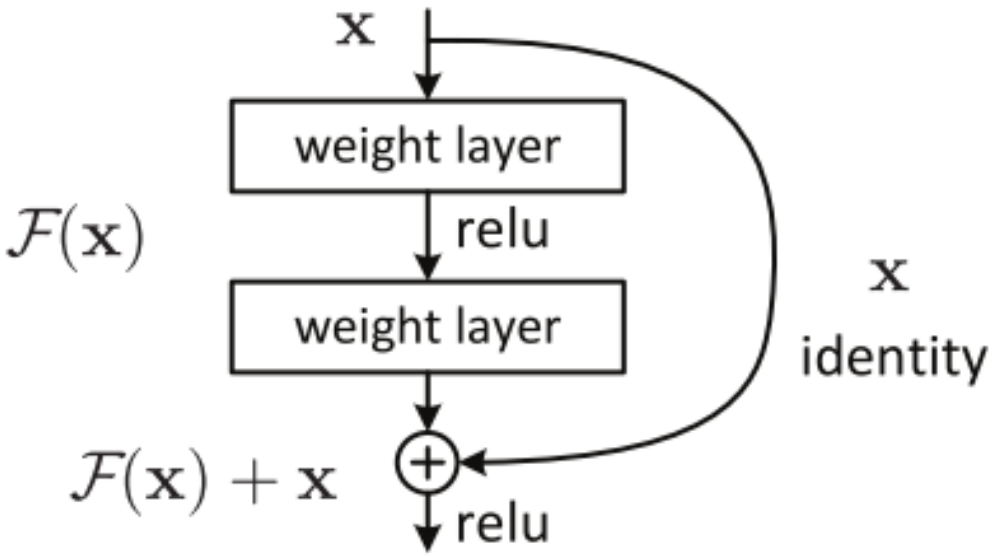
Convolutional filter



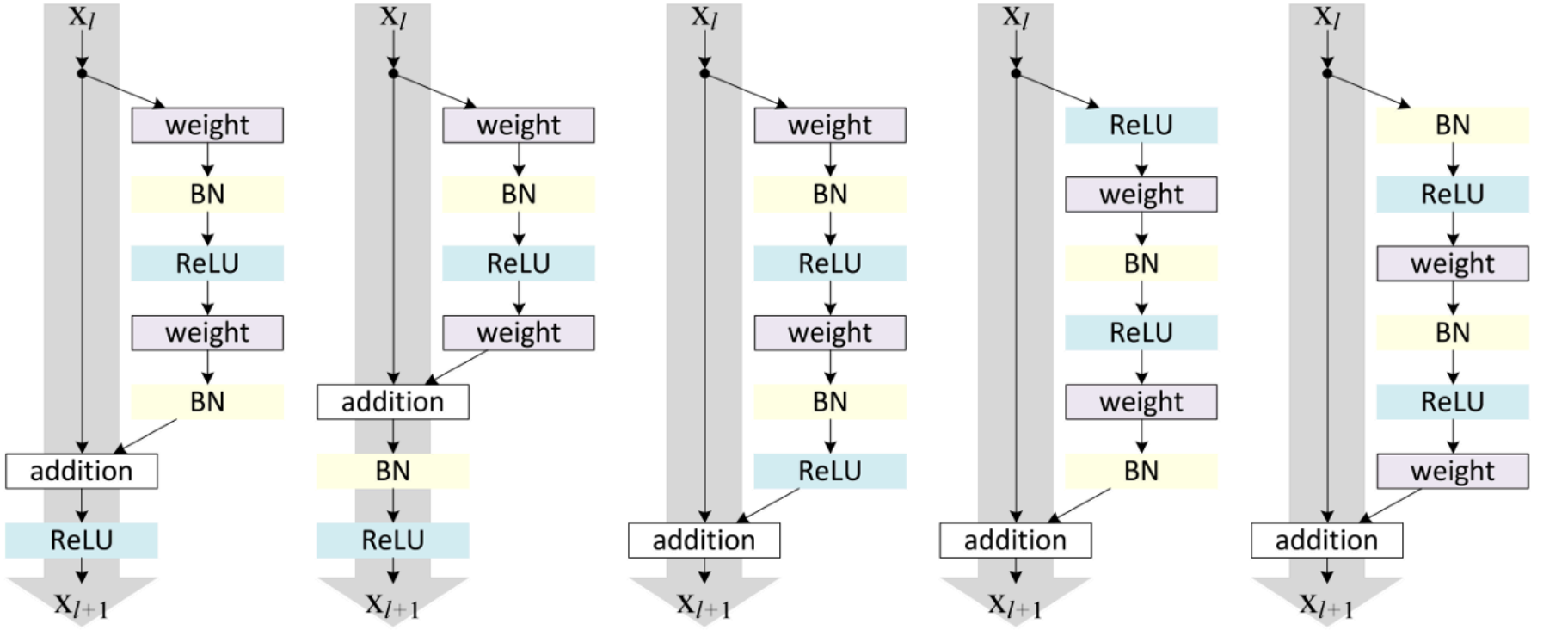


R_script/convolution_filter/convolution_illusrated

ResNet


x
weight layer
relu
weight layer
$\mathcal{F}(\mathbf{x})$
x
identity
$\mathcal{F}(\mathbf{x}) + \mathbf{x}$
relu

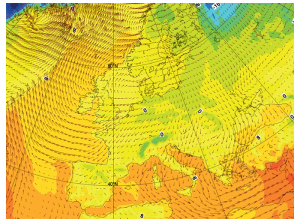
Variants of ResNet

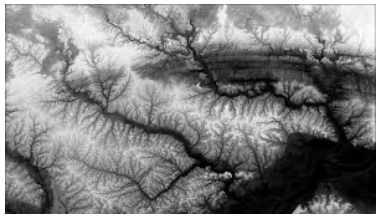


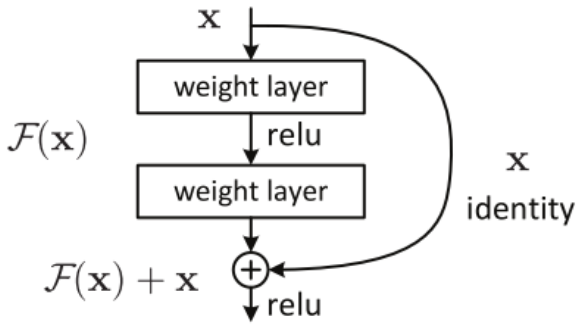
NO_2 mapping using concatenated neural networks

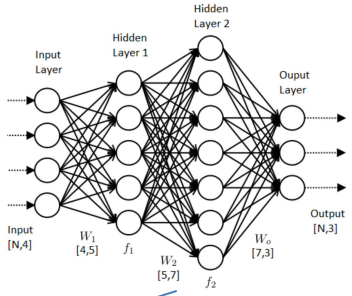
Emission-related

Background









concatenated

Dense

prediction

Hands-on 2 (2)

Kaggle notebook
https://www.kaggle.com/menglugeo/gap-mixed

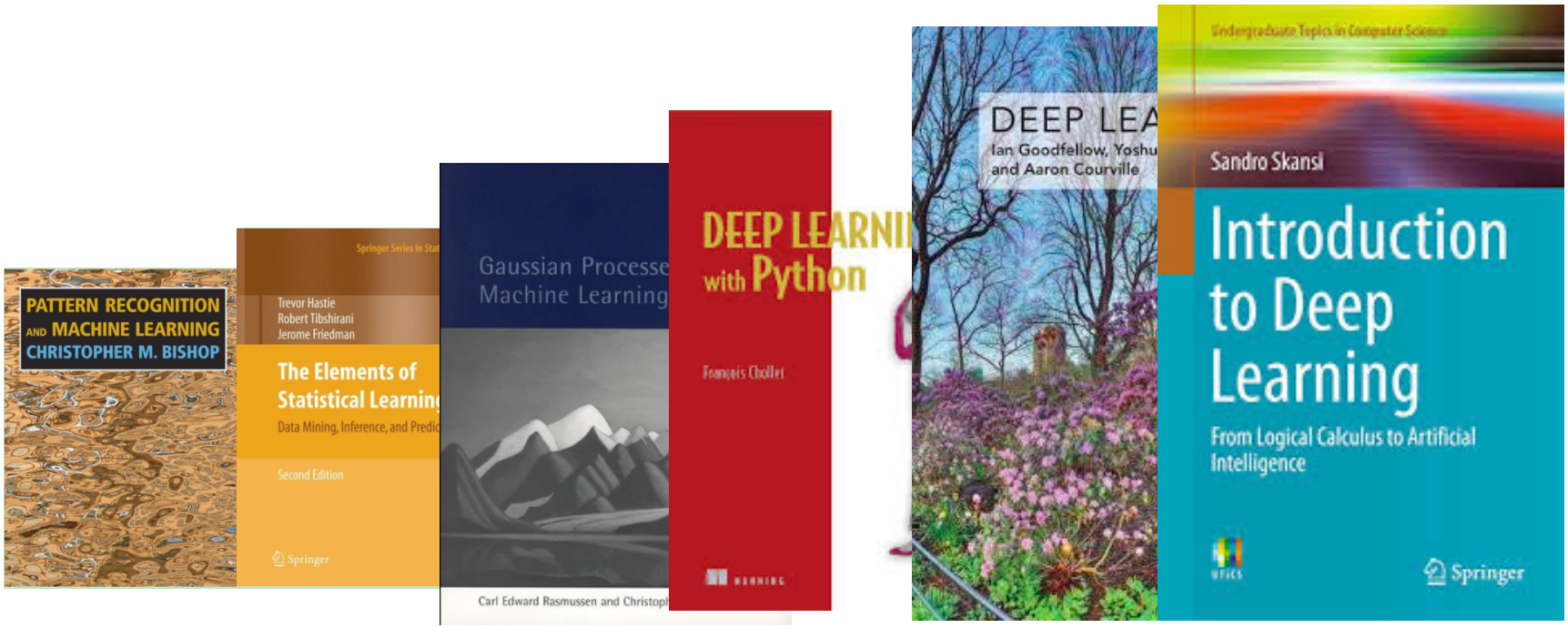
Python/deep_learning/CNN

Hands-on 3: Modeling process

From R_scripts/modeling process

You can find the three R markdown files corresponding to hyper-parameter tunning, cross-validation and mapping.

Congratulations!

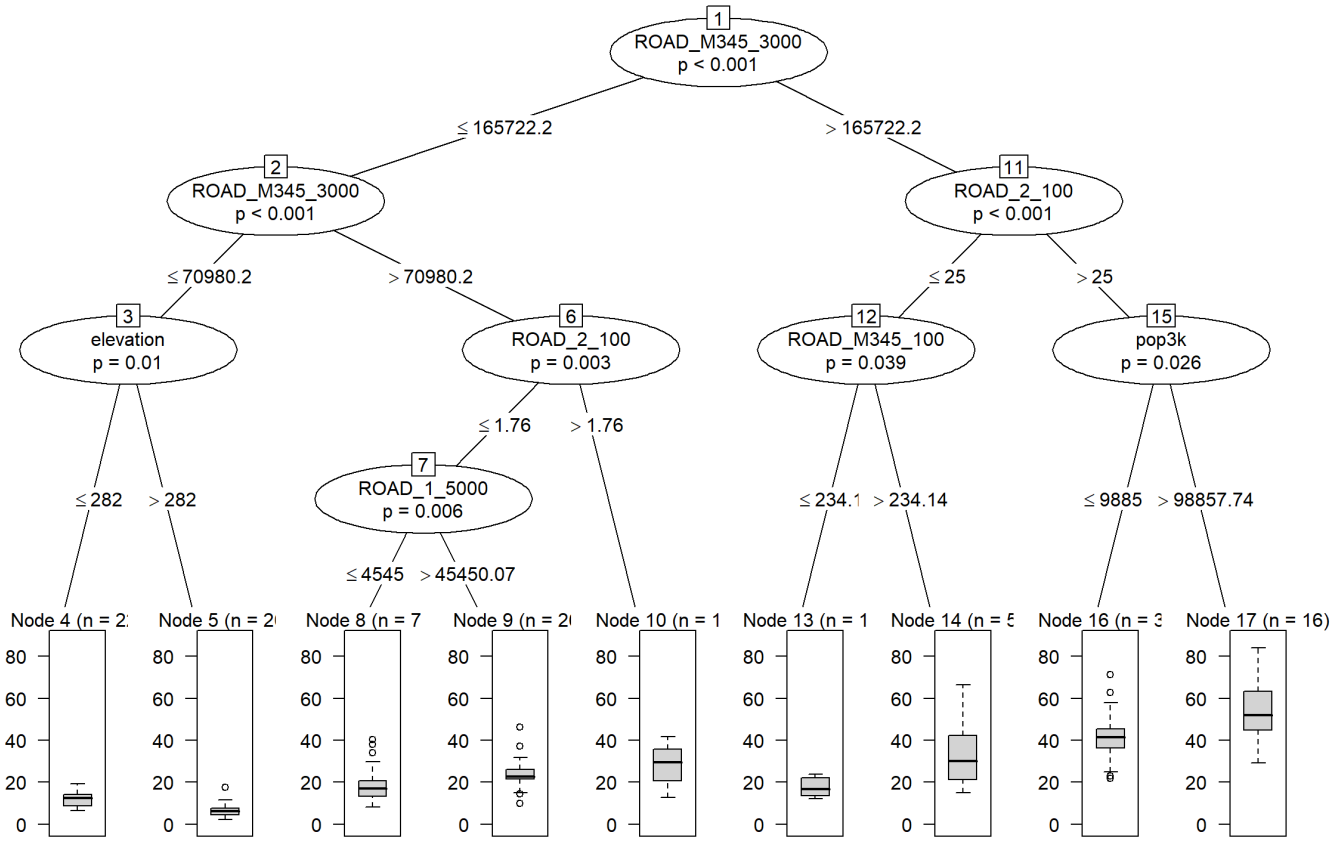


Gaussian processes for machine learning

Online courses:
Standford
MIT
Coursera

A closer look at the model

Visualizing a tree



ROAD_M345: secondry and local roads
Pop_: population
ROAD_2: primary roads
ROAD_1: highway

Partial dependence.

-- Shows the relationship between the target and a feature.

$$\hat{f}_{x_S}(x_S) = E_{x_C}\left[\hat{f}(x_S, x_C)\right] = \int \hat{f}(x_S, x_C) d\mathbb{P}(x_C)$$

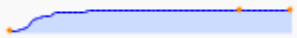
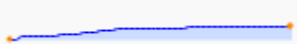
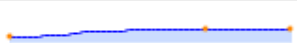
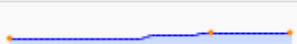
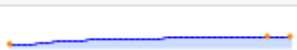
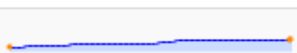
Xs : the features of the partial dependence function
Xc: the other features used in the machine learning model

Marginalizing the model output over the distribution of the features in set C,

Assumption: the features in C are not correlated with the features in S

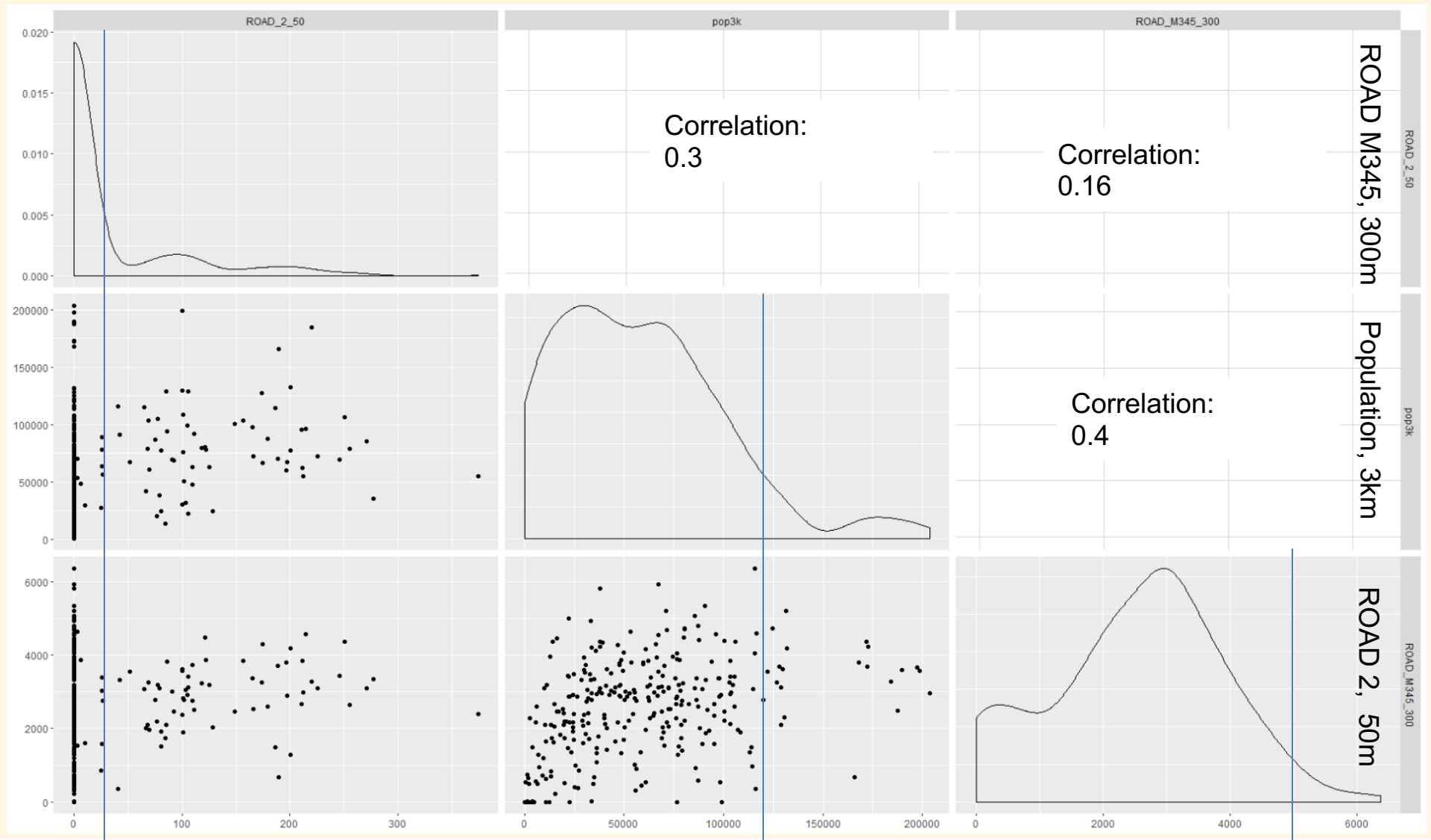
Show 10 entries

Search:

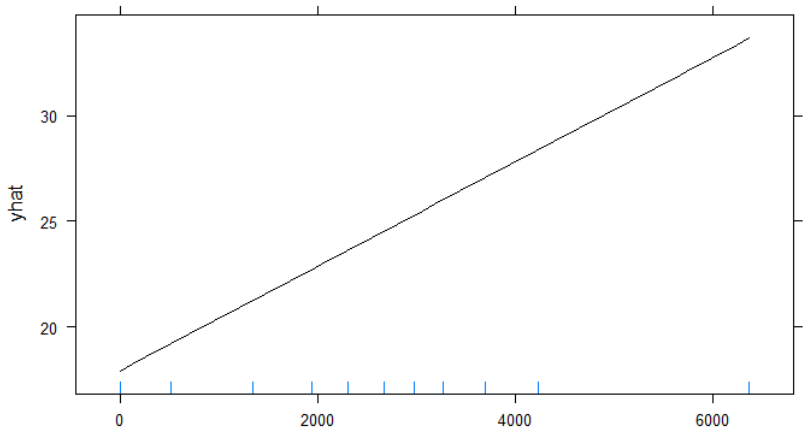
	Variable	Importance	Effect
1	ROAD_2_50	3.032	
2	ROAD_M345_3000	1.542	
3	pop3k	1.379	
4	ROAD_2_100	1.084	
5	ROAD_M345_300	1.058	
6	pop5k	0.840	
7	pop1k	0.756	
8	ROAD_M345_5000	0.674	
9	Tropomi_2018	0.654	
10	ROAD_M345_100	0.578	

Showing 1 to 10 of 65 entries

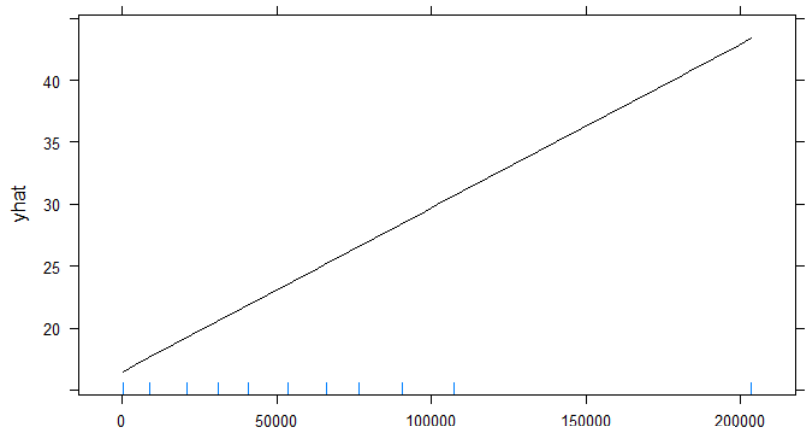
Previous 1 2 3 4 5 6 7 Next


ROAD 2, 50m
Population, 3km
ROAD M345, 300m
ROAD_2_50
pop3k
ROAD_M345_300
Correlation:
0.3
Correlation:
0.16
Correlation:
0.4
ROAD M345, 300m
Population, 3km
ROAD 2, 50m
30m
120000m
5000m

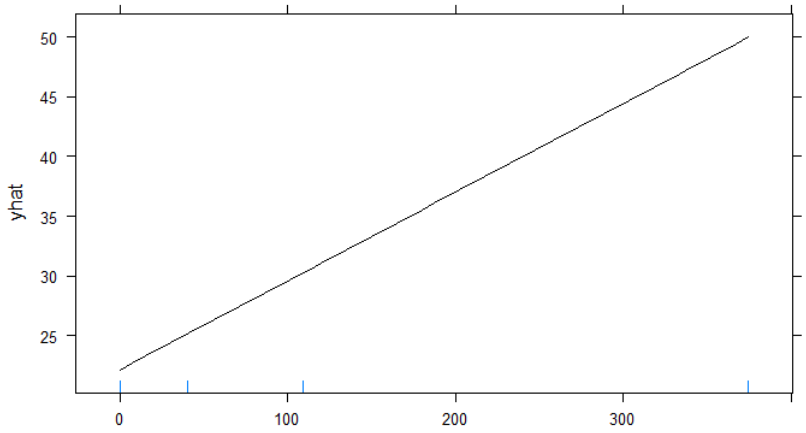
Partial dependent plots: Linear regression



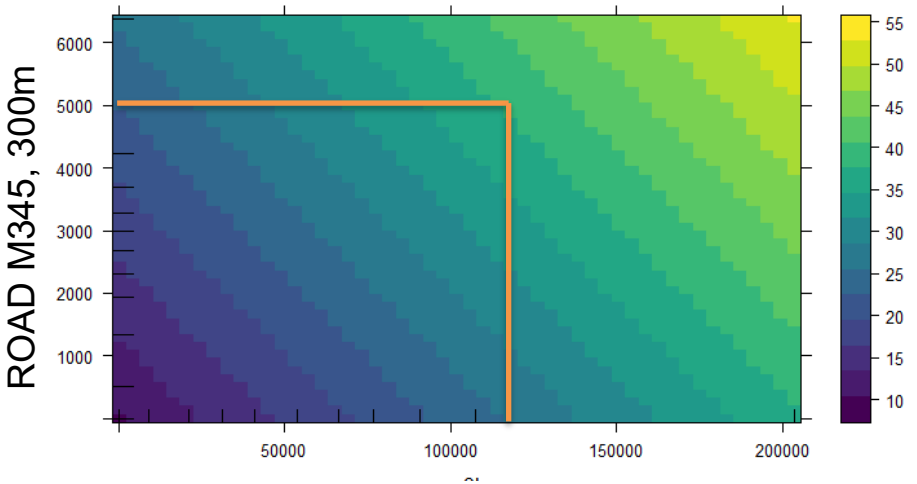
ROAD M345, 300m



Population, 3km

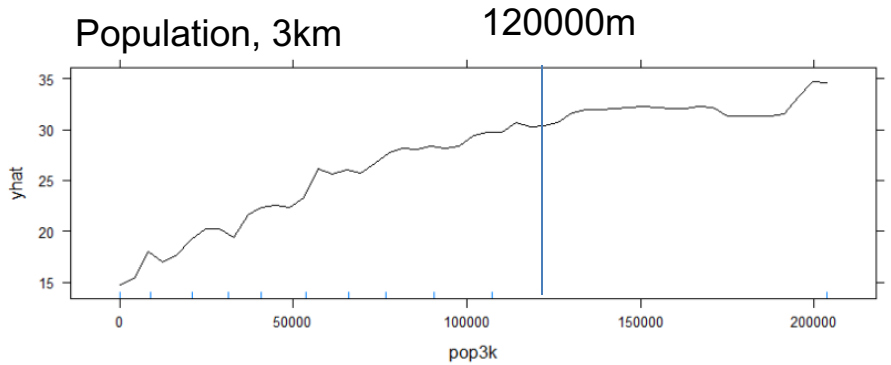


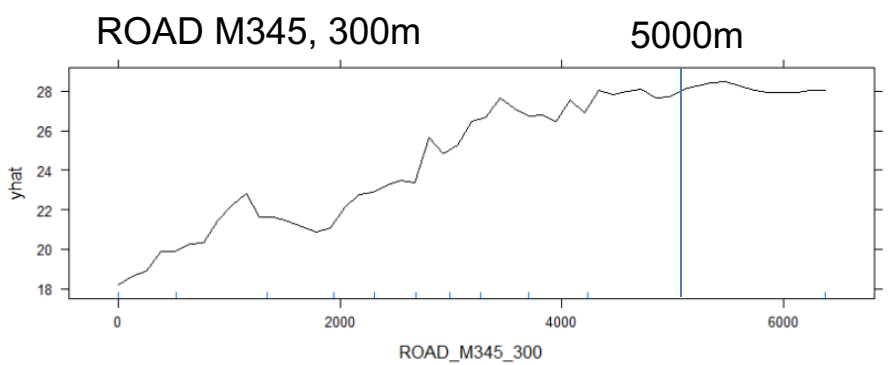
ROAD 2, 50m

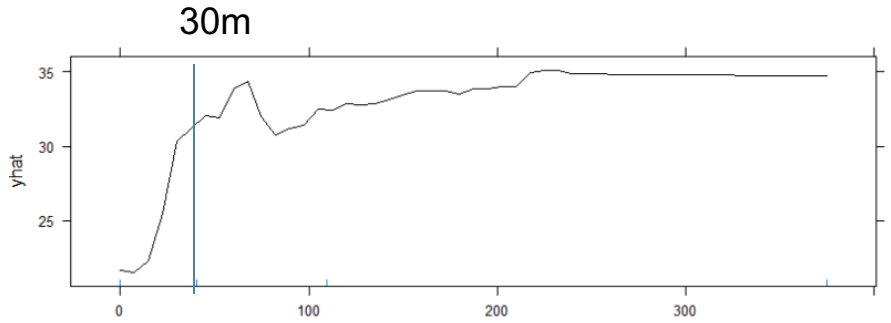


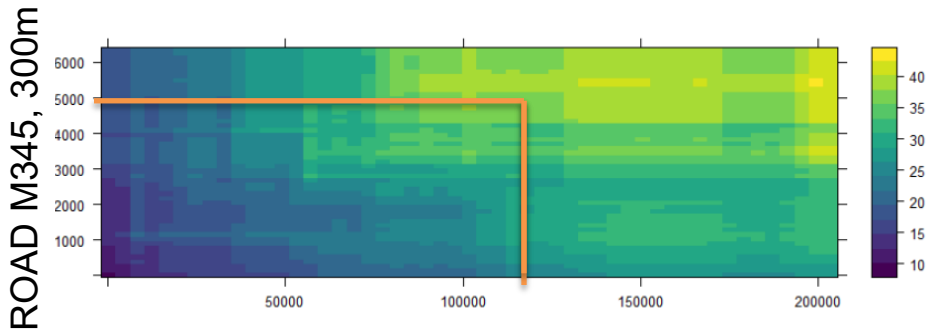
Population, 3km

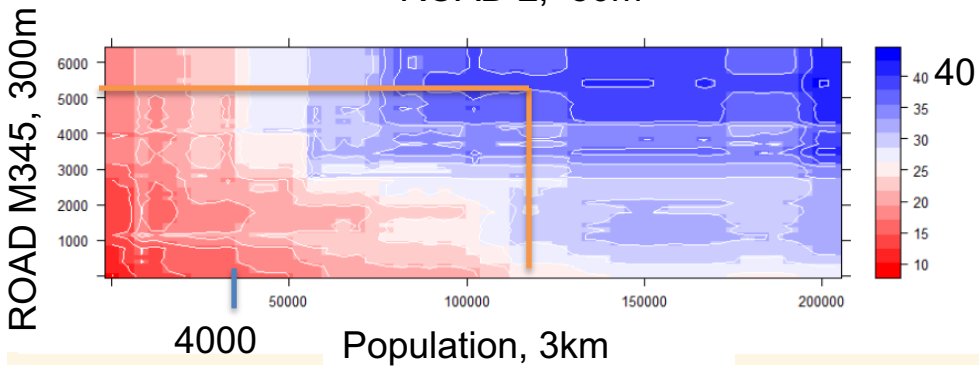
Partial dependent plots: Random forest



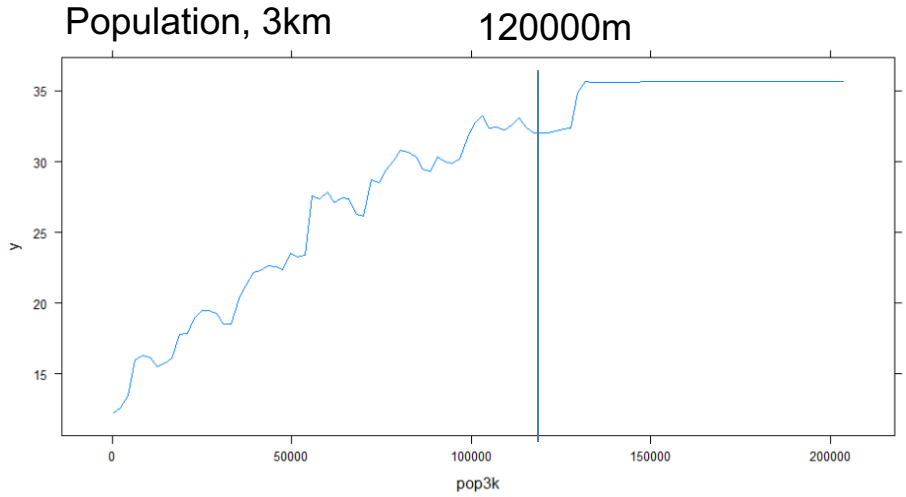


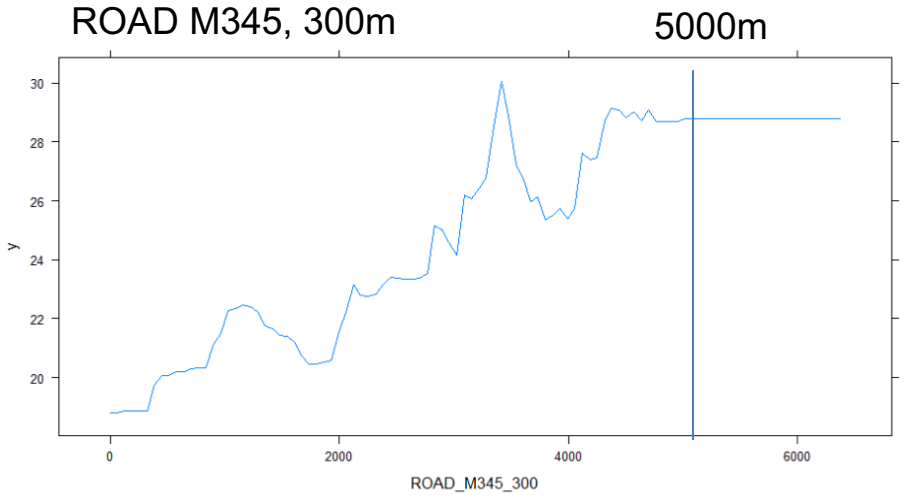


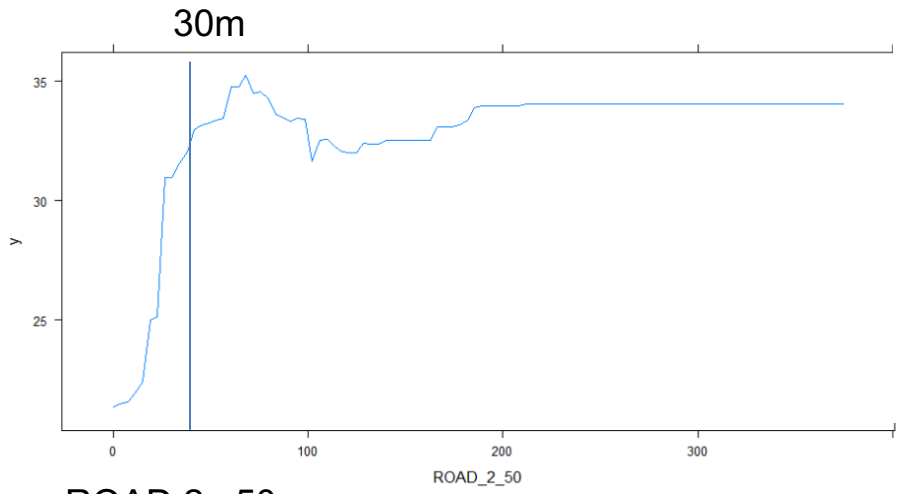


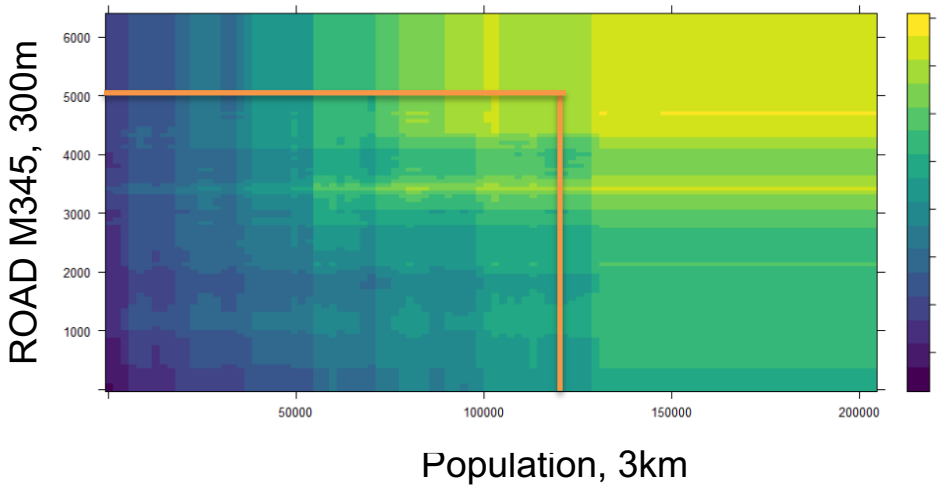


Partial dependent plots: stochastic gradient boosting

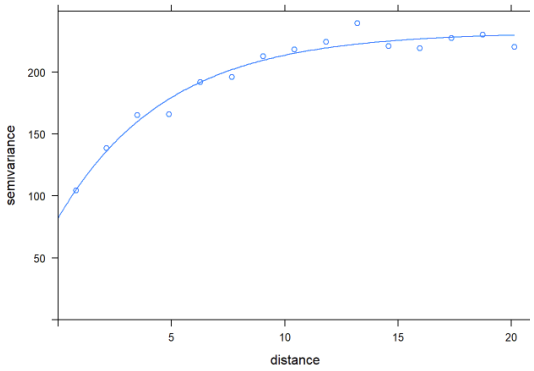


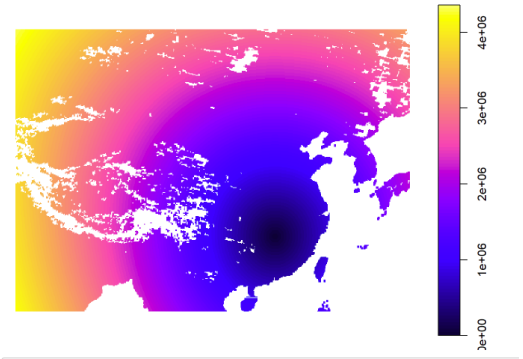


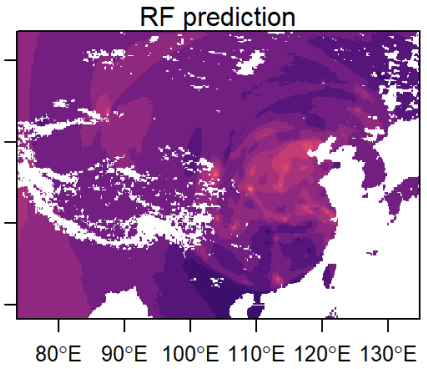




Random Forest only based on distances



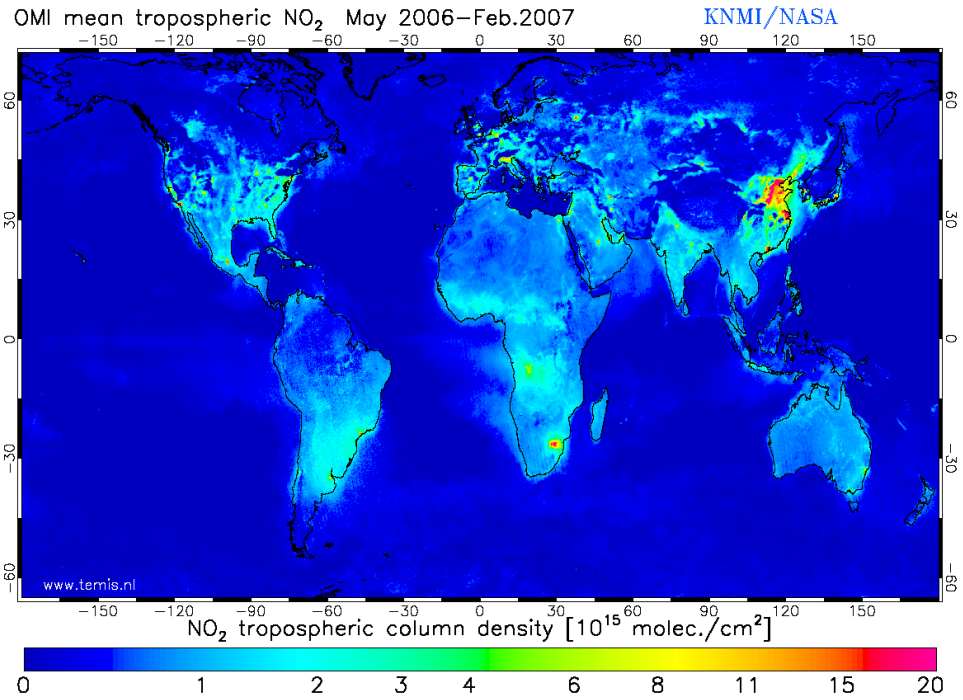




http://rpubs.com/menglu/473973

Road length	https://www.openstreetmap.org/
Industrial area	https://www.openstreetmap.org/
Population	http://www.integrated-assessment.eu/eu/index.html
Elevation	https://www2.jpl.nasa.gov/srtm/
Wind speed	https://confluence.ecmwf.int/pages/viewpage.action?pageId=74764925
Temperature	https://confluence.ecmwf.int/pages/viewpage.action?pageId=74764925
TROPOMI	http://www.tropomi.nl
NO2 Surface product	http://fizz.phys.dal.ca/~atmos/martin/?page_id=232
OMI	https://mirador.gsfc.nasa.gov/cgi-bin/mirador/presentNavigation.pl?tree=project&dataset=OMNO2d.003&project=OMI&dataGroup=L3_V003&version=003&CGISESSID=57642d429543194f1007d9b91bfab0ae

Remote sensing measurements: OMI (Ozone Monitoring Instrument)



Spectral bands: ultraviolet and visible (270 to 500 nm)

Zoom in mode 13 km× 12 km

Daily global coverage

Date of Launch: 15 July 2004

At nadir 13 km × 24 km

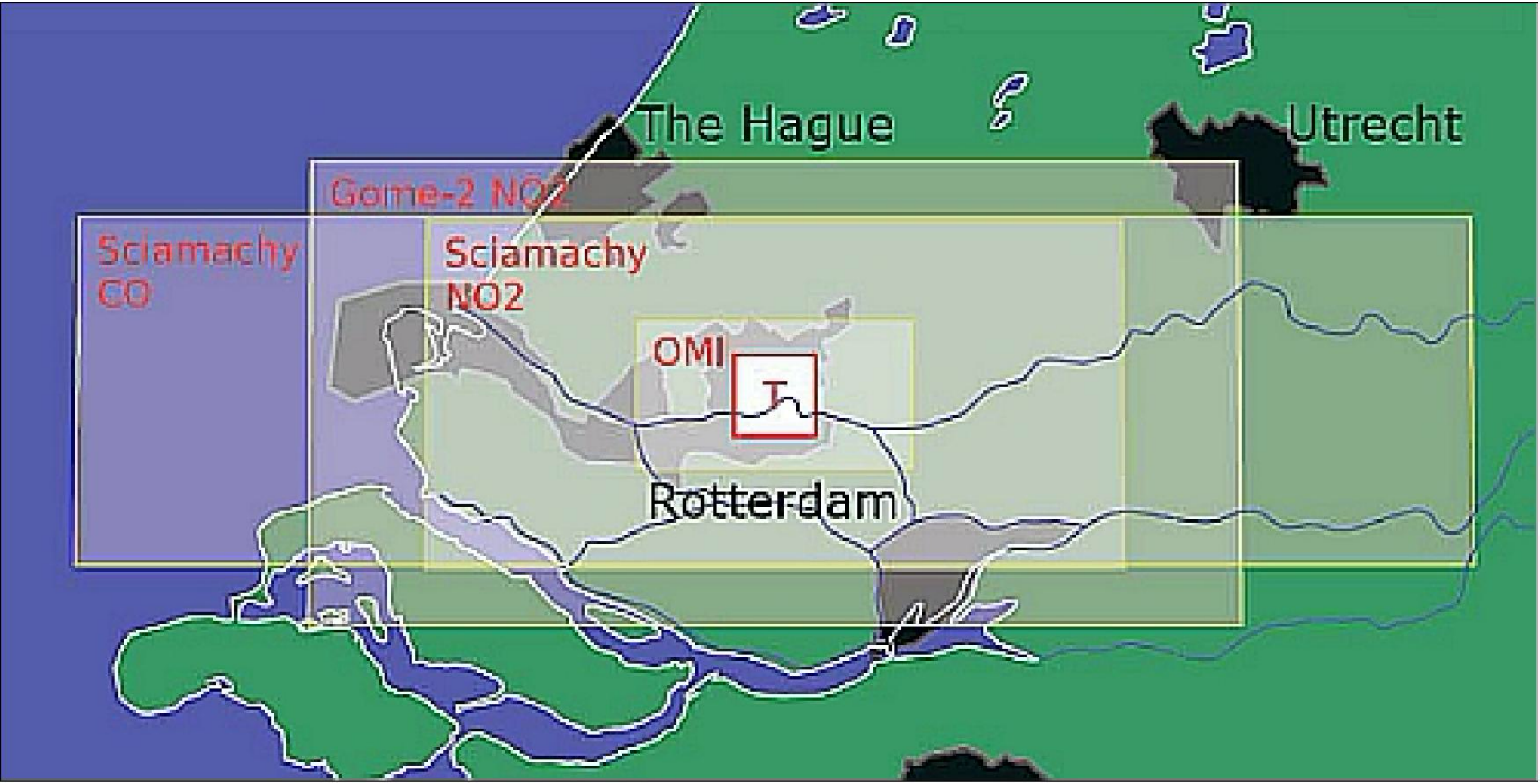
NO2, SO2, BrO, OClO, O3 (36 km × 48 km)

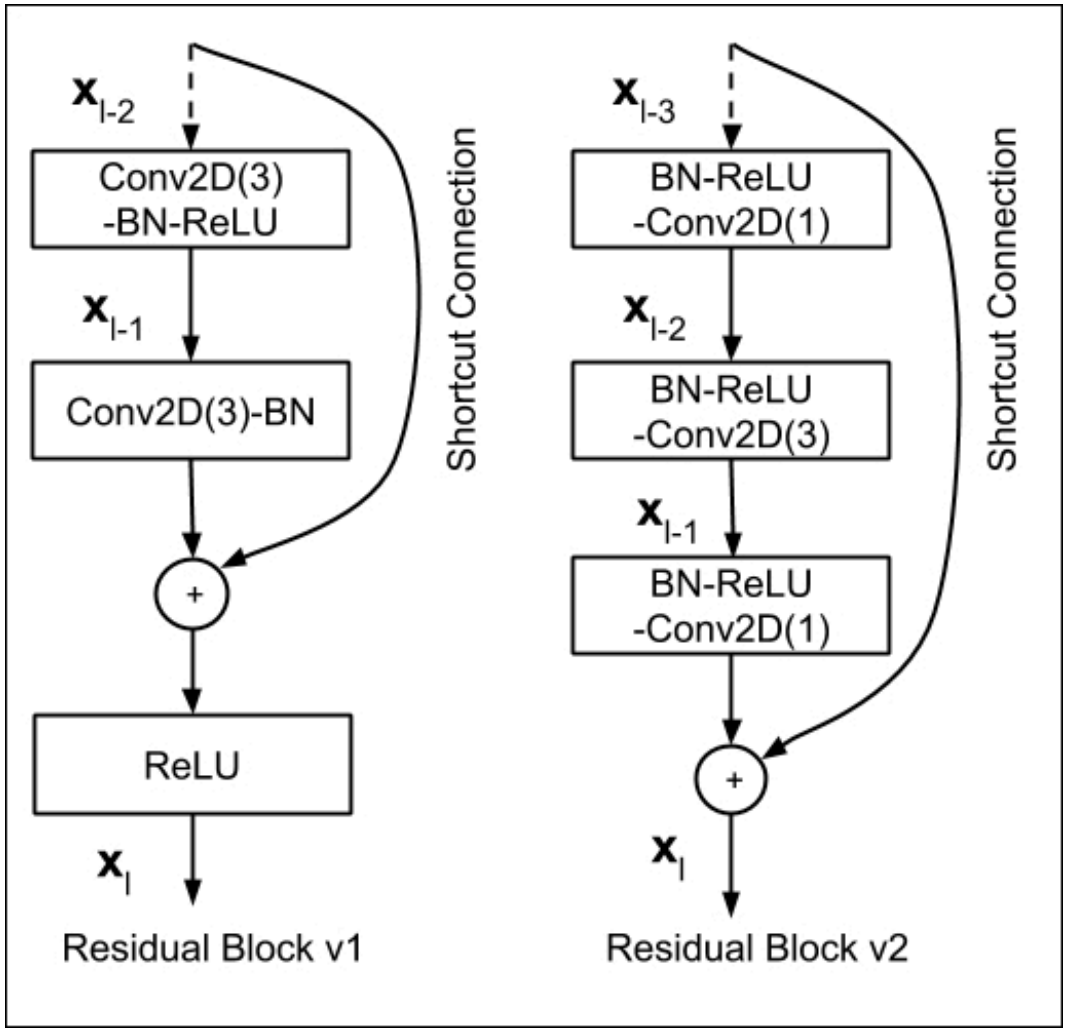
Tropomi: Sentinel 5p

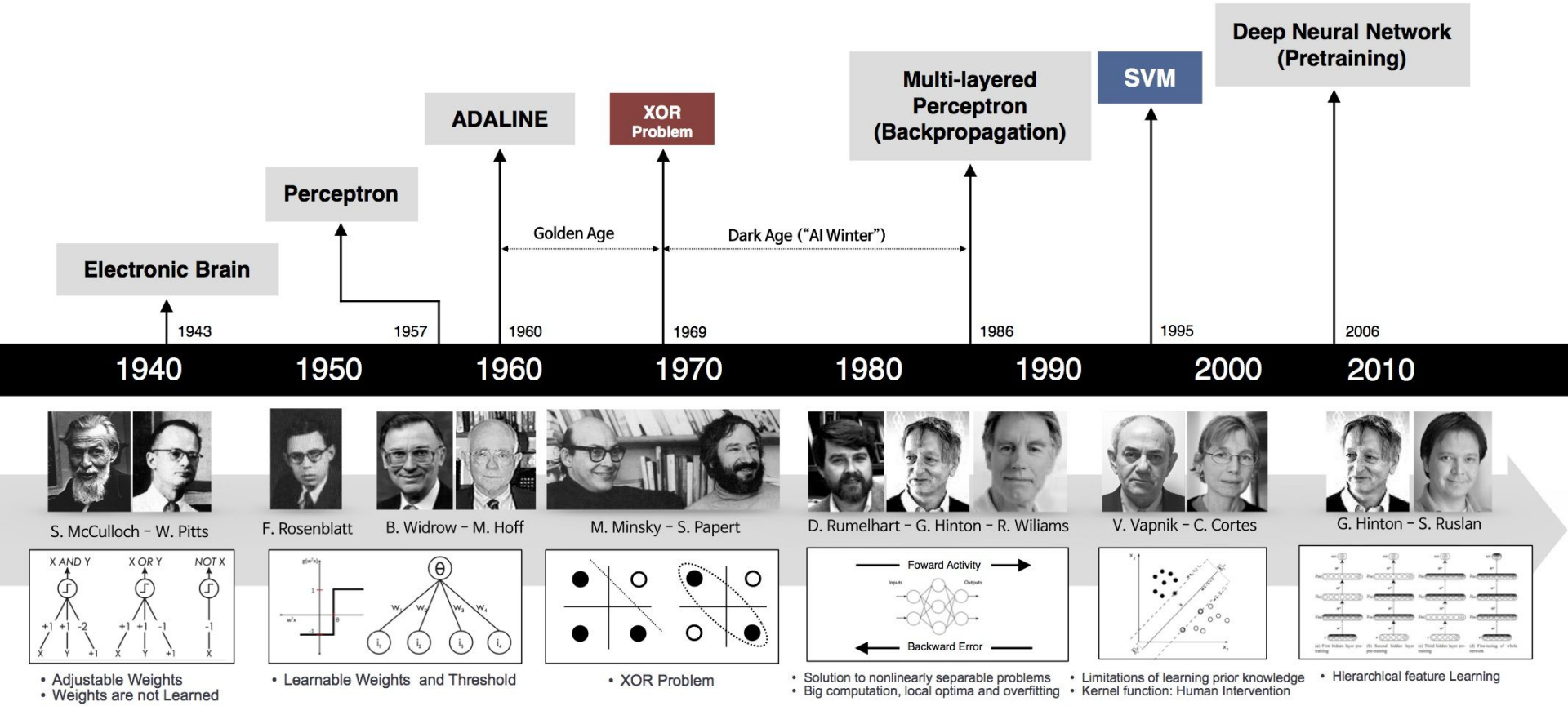
Product	Spectrometer
Ozone	UV, UVIS
NO_2	UVIS
CO	SWIR
CH_2O	UVIS
CH_4	SWIR
SO_2	UVIS
Aerosol	UVIS, NIR
Clouds	UVIS, NIR
UV-Index	UVIS

TROPOMI	UV		UVIS		NIR		SWIR	
Band	1	2	3	4	5	6	7	8
Spectral coverage [nm]	270 – 320		320 – 495		675 - 775		2305 – 2385	
Full spectral coverage [nm]	267 - 332		303 - 499		660 - 784		2299 - 2390	
Spectral resolution [nm]	0.49		0.54		0.38		0.25	
Spectral sampling ratio	6.7		2.5		2.8		2.5	
Spatial sampling [km^2]	7 x 28	7 x 3.5				7 x 3.5	7 x 7	

Feb 2018 to Jan 2019


The Hague
Utrecht
Gome-2 NO2
Sciamachy
CO
Sciamachy
NO2
OMI
Rotterdam


x_{l-2}
Conv2D(3)
-BN-ReLU
x_{l-1}
Conv2D(3)-BN
+
ReLU
x_l
Residual Block v1
Shortcut Connection
x_{l-3}
BN-ReLU
-Conv2D(1)
x_{l-2}
BN-ReLU
-Conv2D(3)
x_{l-1}
BN-ReLU
-Conv2D(1)
+
x_l
Residual Block v2
Shortcut Connection



- Adjustable Weights
- Weights are not Learned

- Learnable Weights and Threshold

- XOR Problem

- Solution to nonlinearly separable problems
- Big computation, local optima and overfitting

- Limitations of learning prior knowledge
- Kernel function: Human Intervention

- Hierarchical feature Learning