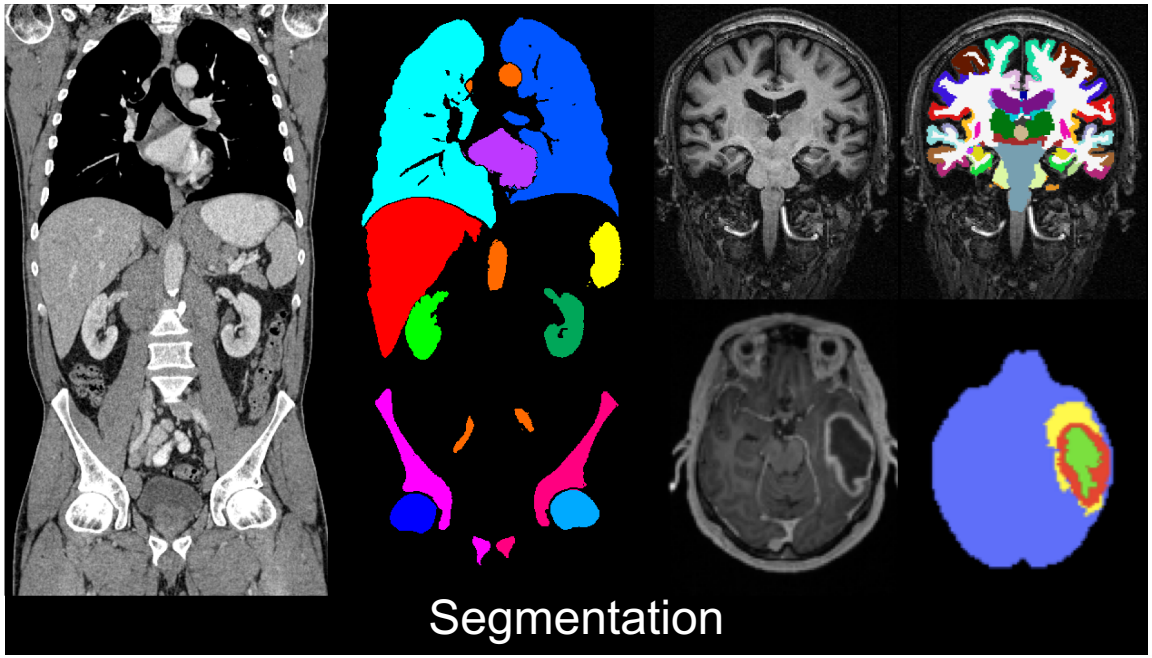
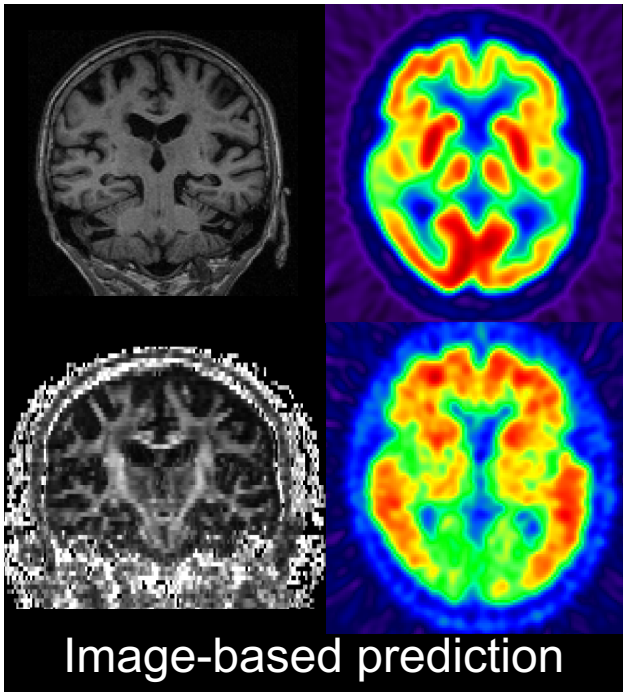
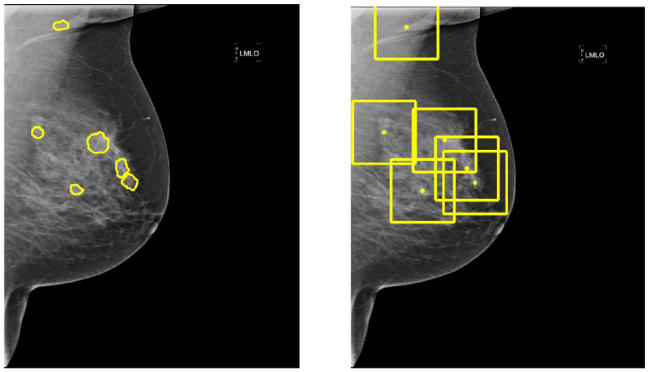
From random forests to neural networks

Ender Konukoglu

Computer Vision Laboratory, ETH Zürich







Detection

[Taken from Kooi et al. MedIA 2017]

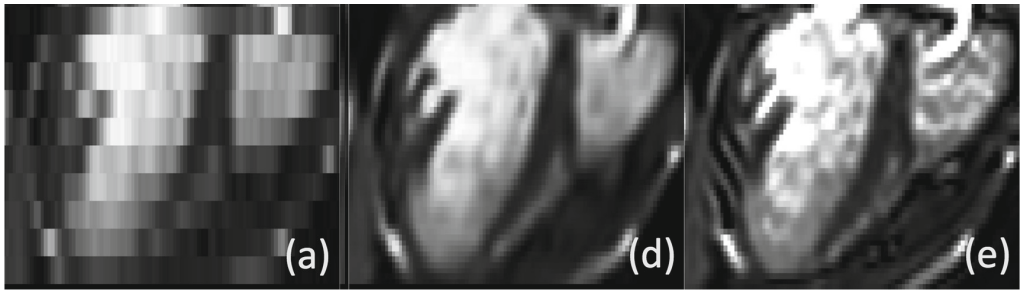


Image Enhancement

[Taken from Oktay et al. MICCAI 2016]

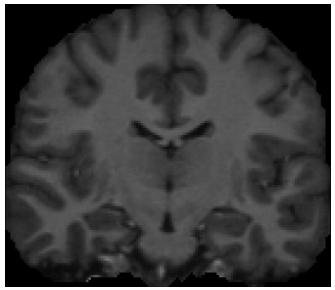
CVL Computer Vision Lab

Outline

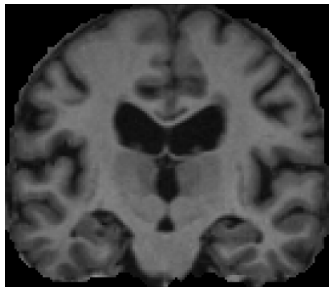
- Elementary stuff
- Historical overview (supervised learning)
- Closer look at Random Forests
- Comparison to DL
- Cross-breeds

Elementary stuff – Patterns

- Patterns may exist in the data (correlations, relationships, dependencies, …)
- Within images, e.g. liver is always right below the lungs,…
- Between images and other info, e.g. brains age similarly and organs look alike
- Patterns may arise due to nature or causal relationships

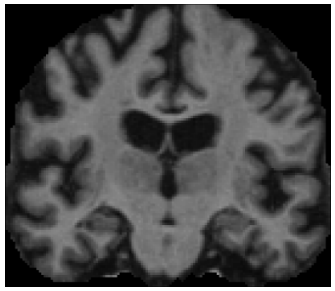


17 years old

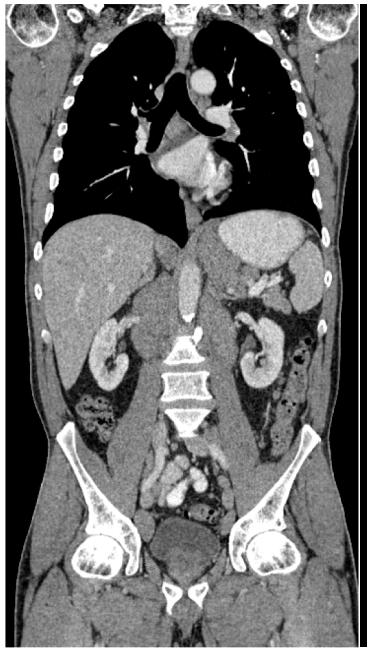


81 years old

30 years old



74 years old



CVL Computer Vision Lab

Elementary stuff – Notation

- Data representation:

$$\{x_1, \ldots, x_N, y_1, \ldots, y_M\} = \{\mathbf{x}, \mathbf{y}\} = \{\text{features}, \text{labels}\}$$

- Continuous , discrete, categorical
- Supervised learning – mapping between features and labels

$$\mathbf{y} = f\left(\mathbf{x}|\theta\right), \ \theta : \text{Model parameters}$$

- Unsupervised learning – learning distribution of features

$$p(\mathbf{x}|\theta), \ \theta : \text{Model parameters}$$

Elementary stuff – Learning and prediction

- Determining the parameters based on examples - **training**

$$\{\mathbf{x}_s, \mathbf{y}_s\}_{s=1,\ldots,S}$$

$$\theta^* = \arg_\theta \max \sum_{s=1}^{S} d\left(f\left(\mathbf{x}_s|\theta\right), \mathbf{y}_s\right)$$

- Prediction with learnt parameters - **testing** $\hat{\mathbf{y}} = f\left(\mathbf{x}|\theta^*\right)$
- Unsupervised learning – maximizing and predicting likelihood

$$\theta^* = \arg_\theta \max \prod_{s=1}^{S} p\left(\mathbf{x}_s|\theta\right)$$

$$p\left(\mathbf{x}|\theta^*\right)$$

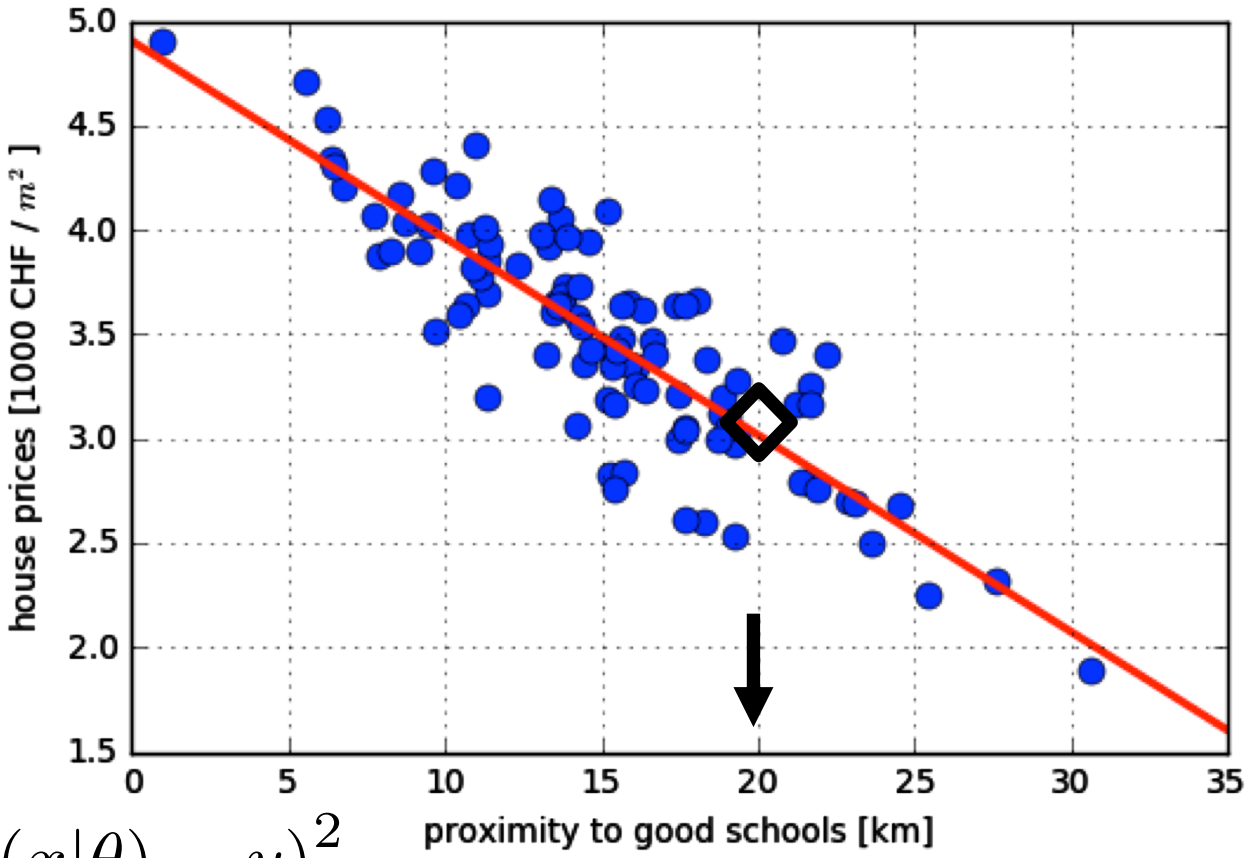
Elementary stuff – Regression example

x: proximity
y: house price

$$\{x_s, y_s\}_{s=1,\ldots,S}$$

$$y = f(x) = ax + b$$

$$\theta = \{a, b\}$$



$$d\left(f\left(x|\theta\right), y\right) = \left(f(x|\theta) - y\right)^2$$

Elementary stuff – Classification example

x: weight, height
y: gender

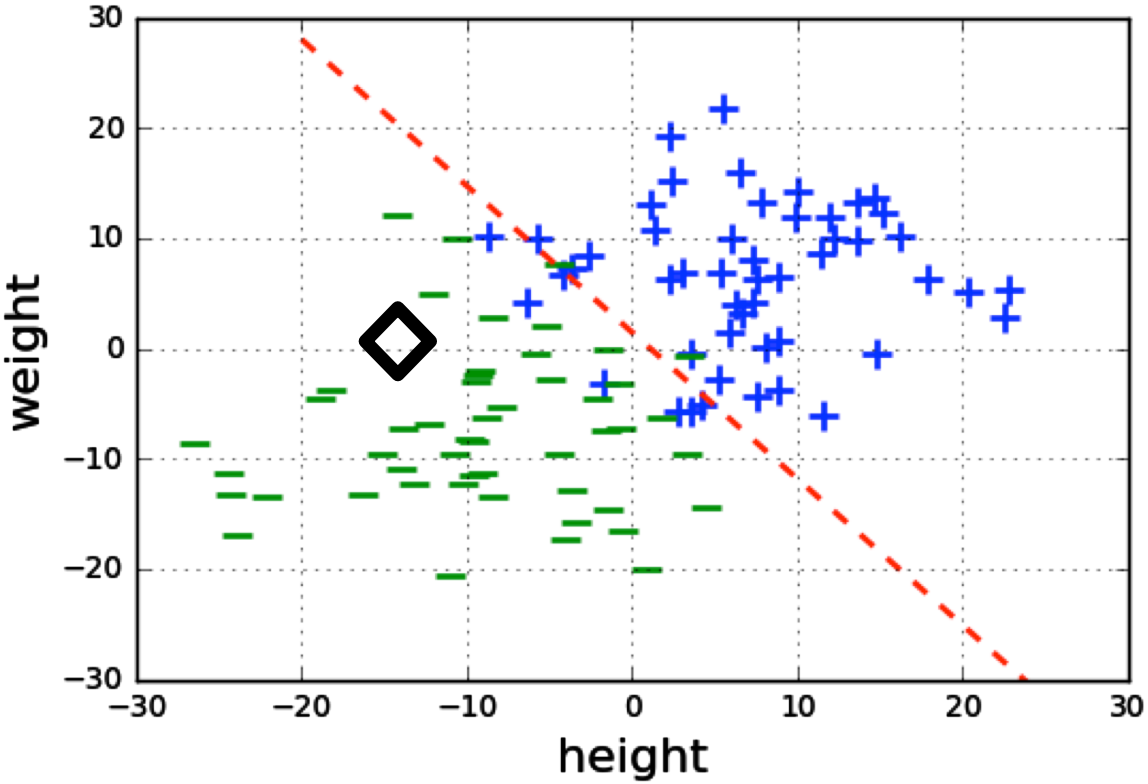
$$\{\mathbf{x}_s, y_s\}_{s=1,\ldots,S}$$

$$\begin{aligned} y &= f(x) = p(\text{gender=F}) \\ &= \sigma(a_1 x_1 + a_2 x_2 + b) \\ \sigma(z) &= \frac{1}{1+e^{-z}} \end{aligned}$$

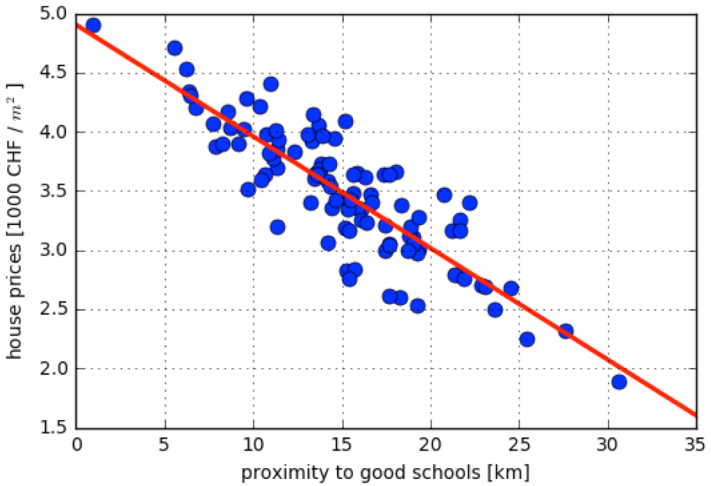
$$\theta : \{a_1, a_2, b\}$$

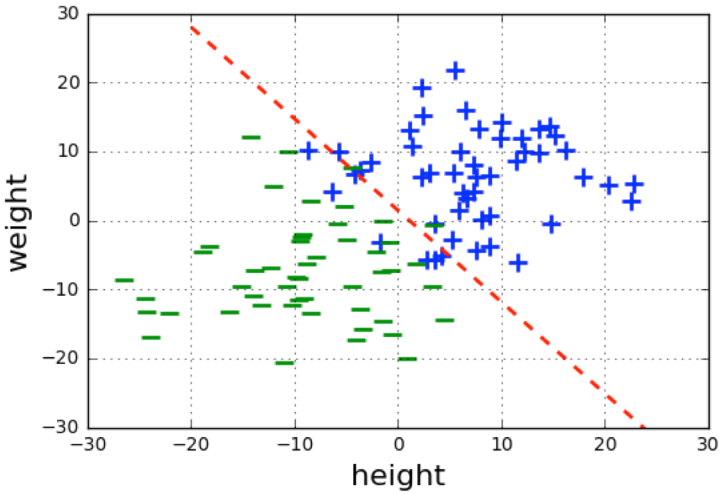
$$d\left(f\left(x|\theta\right), y\right) = \begin{cases} \log\{f(x|\theta)\}, & y = + \\ \log\{1 - f(x|\theta)\}, & y = - \end{cases}$$

Line is the point where y = 0.5



Note 1: Predictors

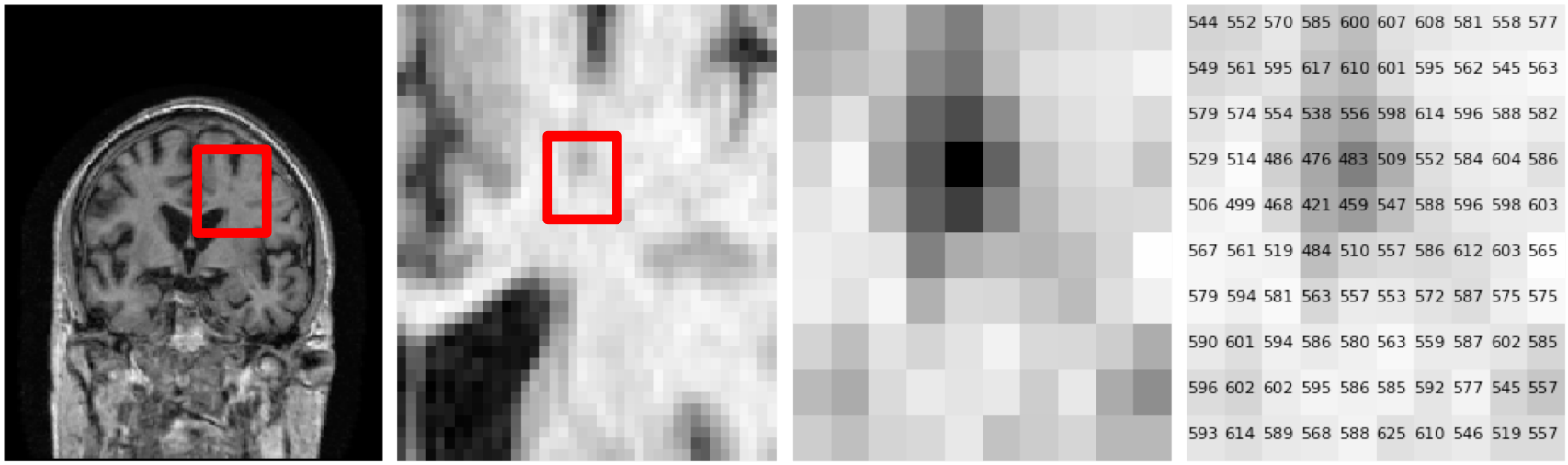




Proximity to good schools
Number of rooms
Surface area
View
Garden
…

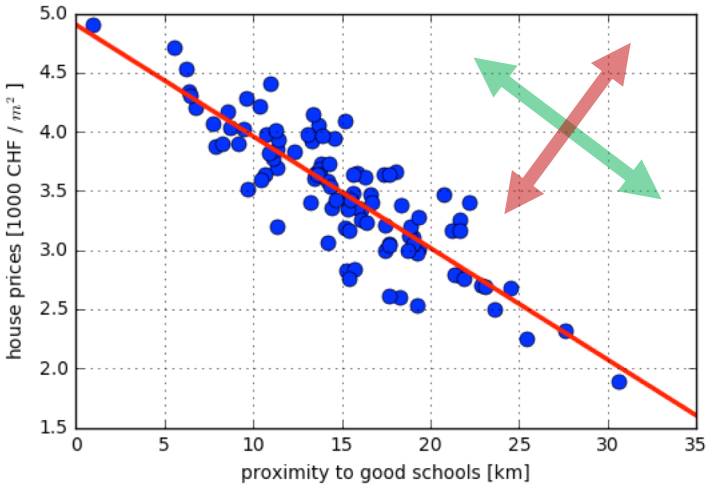
Weight
Height
Age
Shoe size
….

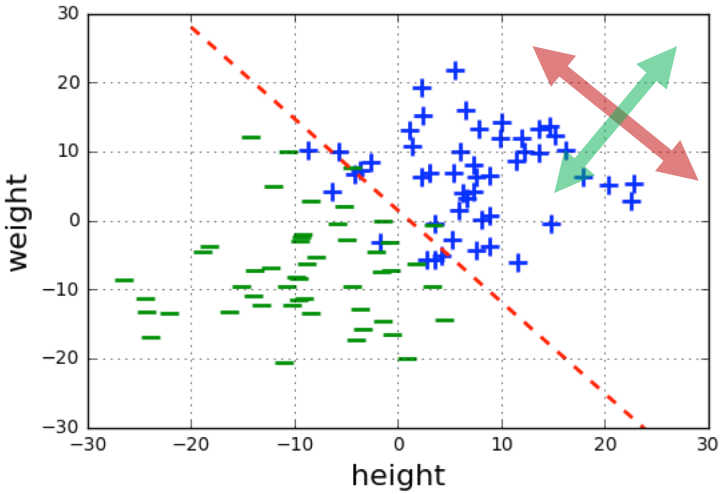
Predictors in images



- Multiple predictors
- Complex spatial correlations between predictors
- Two challenges to address
  - x=?
  - $F(x|\boldsymbol{\theta}) = ?$

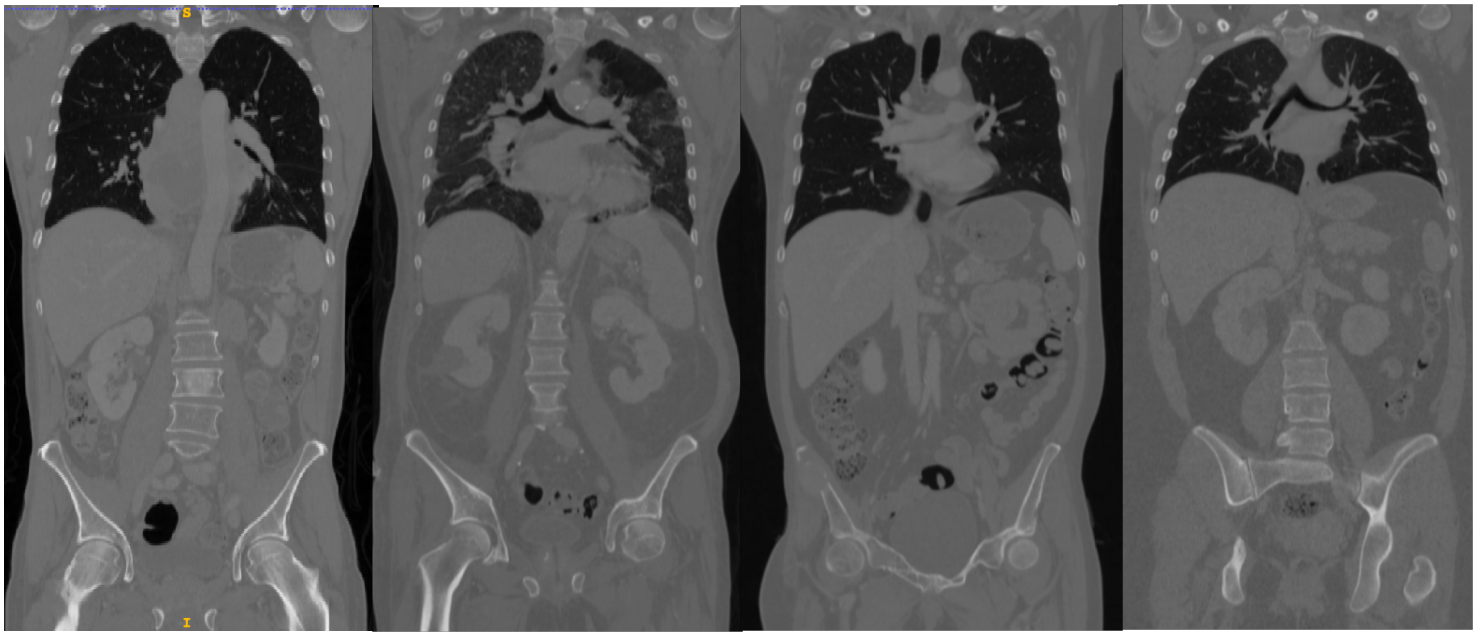
Note 2: Variability

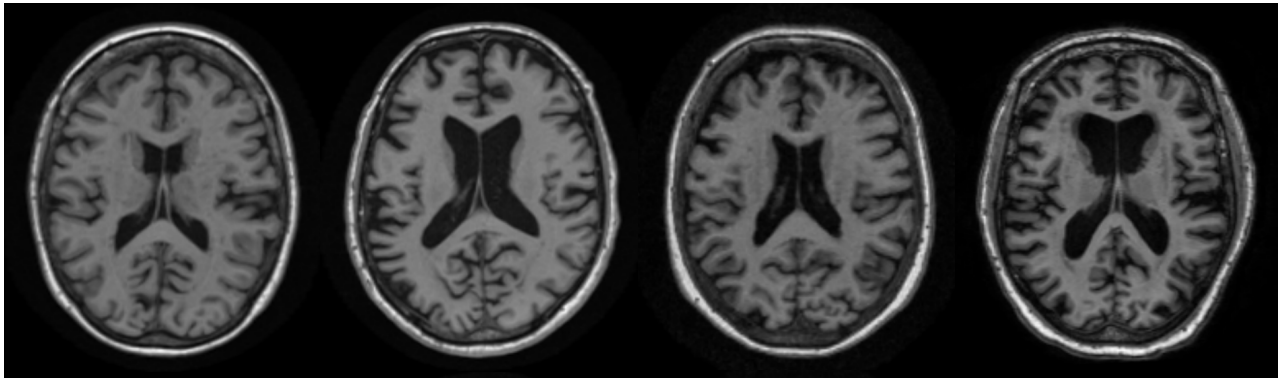




- Two types of variation in predictors
- Variation useful for prediction, e.g. inter-group variation
- Variation not useful for prediction, e.g, intra-group variation
- Models need to learn to use one and ignore the other

Variation in high dimensions is a bigger issue

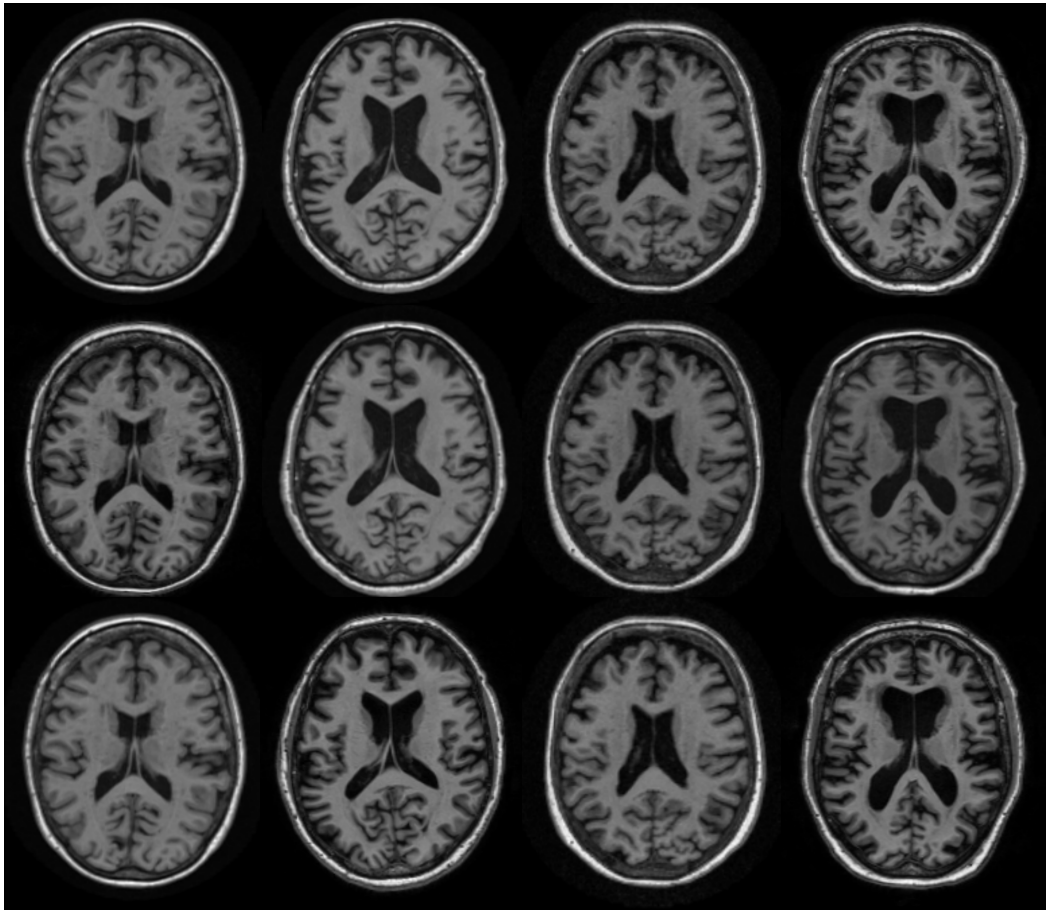




Large amount of images to capture variability

Different Individuals

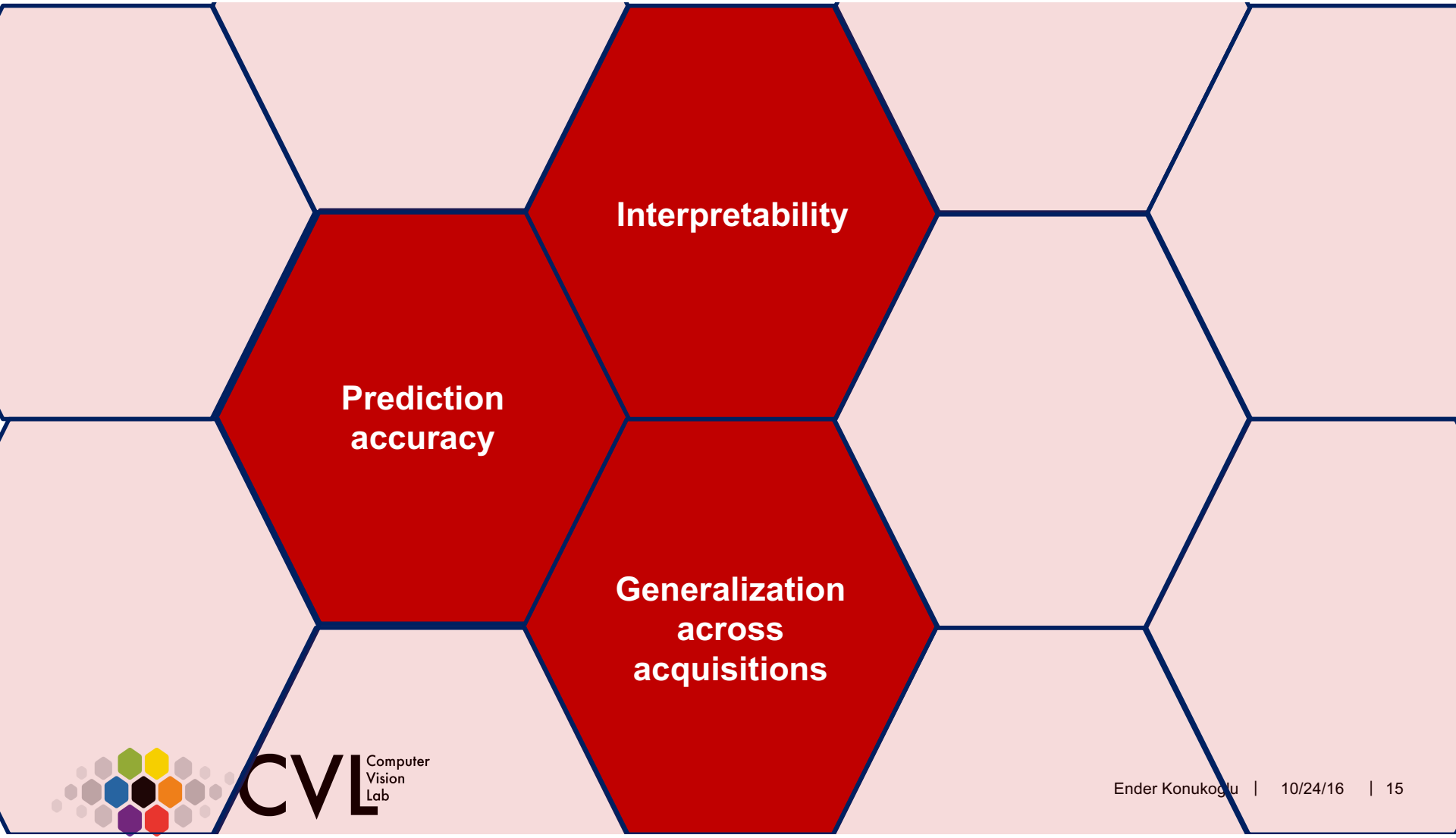
Different Time Points



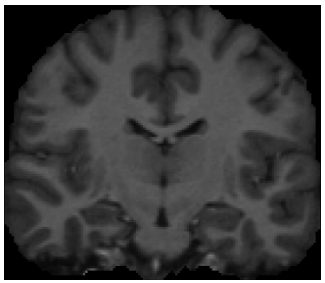
Outline

- Elementary stuff
- Historical overview (supervised learning)
  - Univariate regression analysis
  - Multivariate analysis with linear models
  - Explicit feature selection methods – Random forests, Boosting
  - Deep learning
- Closer look at Random Forests
- Comparison to DL
- Cross-breeds

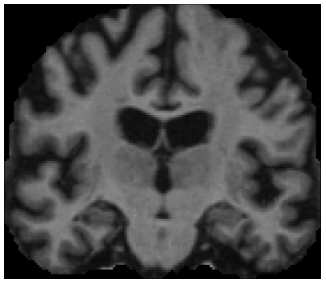
Goals for supervised learning in MIC

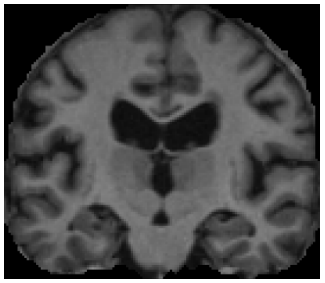


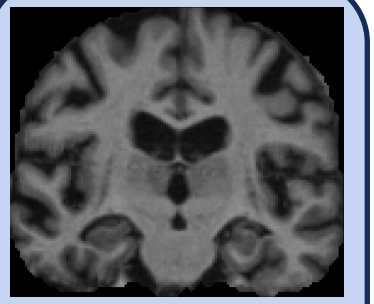
Different algorithms on a model problem

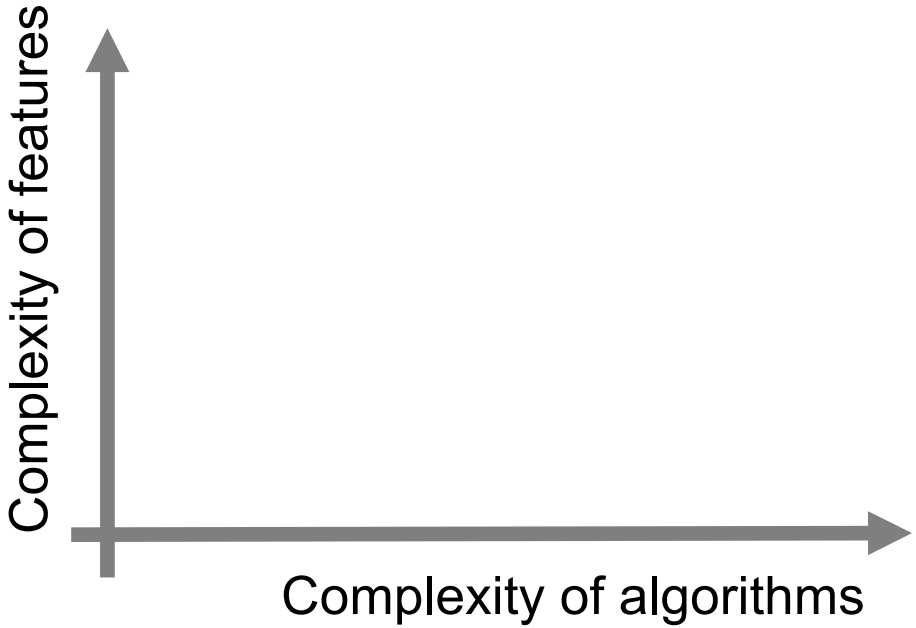

17 years old

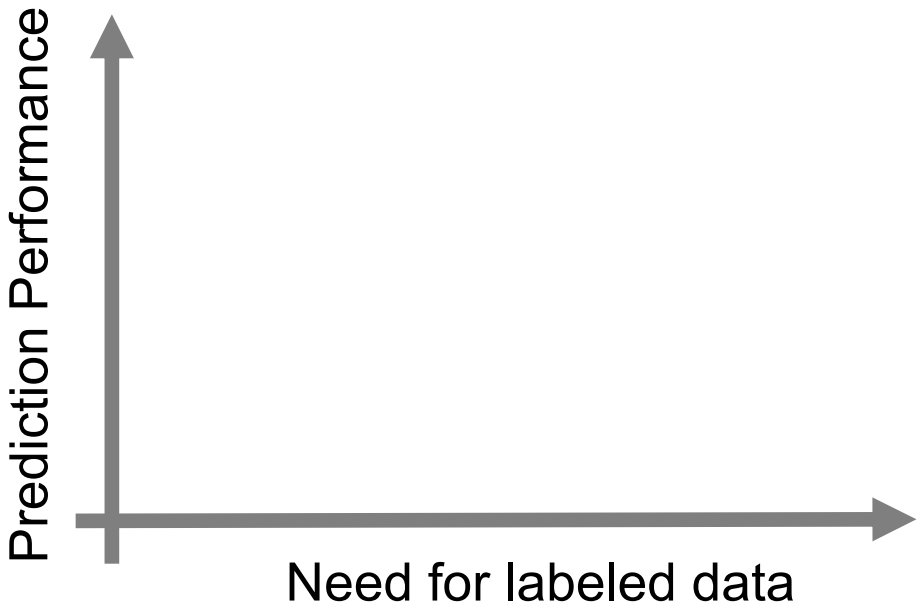

30 years old


74 years old


81 years old

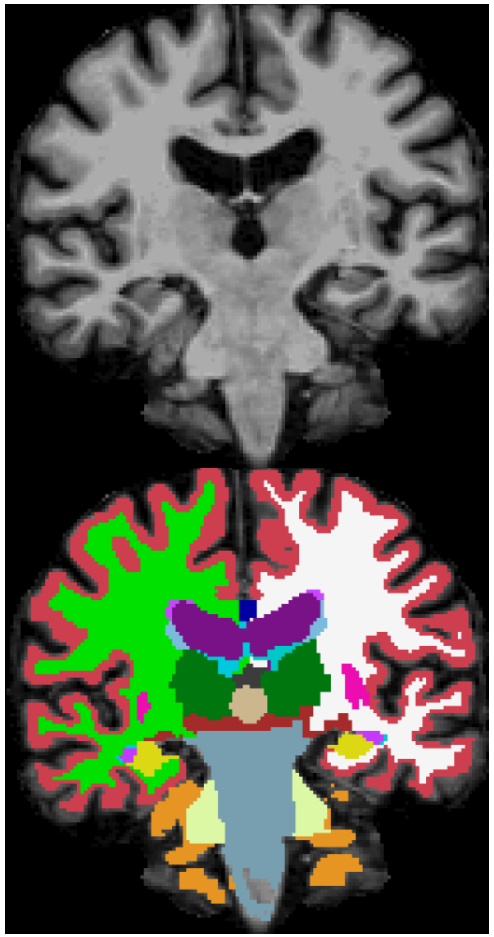

How old is she?





CVL Computer Vision Lab

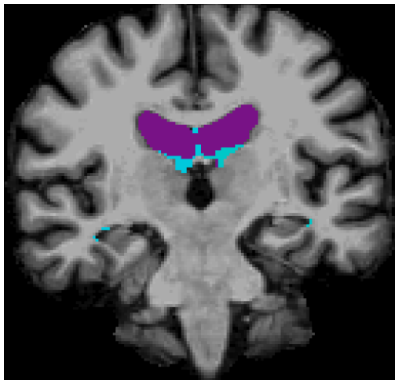
Measurements and univariate analysis



x: volume of one anatomical structures

$$f(x|\theta) = ax + b$$

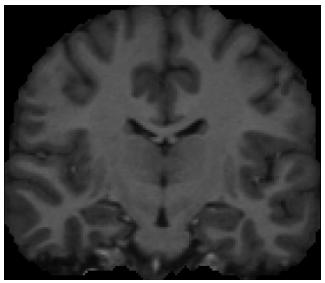
- Hand-crafted measurements
- Correlation / regression analysis
- Discovering statistical relations
- Interpretation on specific structures
- Statistical significance tests
- Univariate and does not combine info
- Difficult to extend to other problems, e.g. segmentation



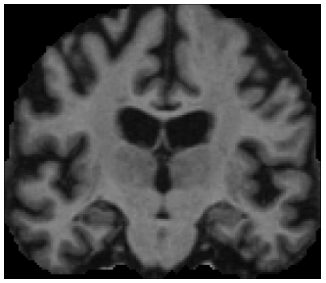
Pearson's Correlation Coef.
Left Lateral Ventricle – 0.72
Right Lateral Ventricle – 0.72
Left Choroid Plexus – 0.71
Right Choroid Plexus – 0.73

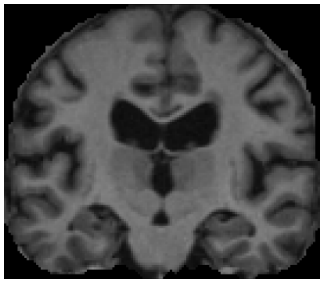
CVL Computer Vision Lab

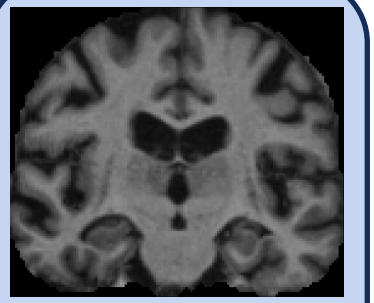
Different algorithms on a model problem

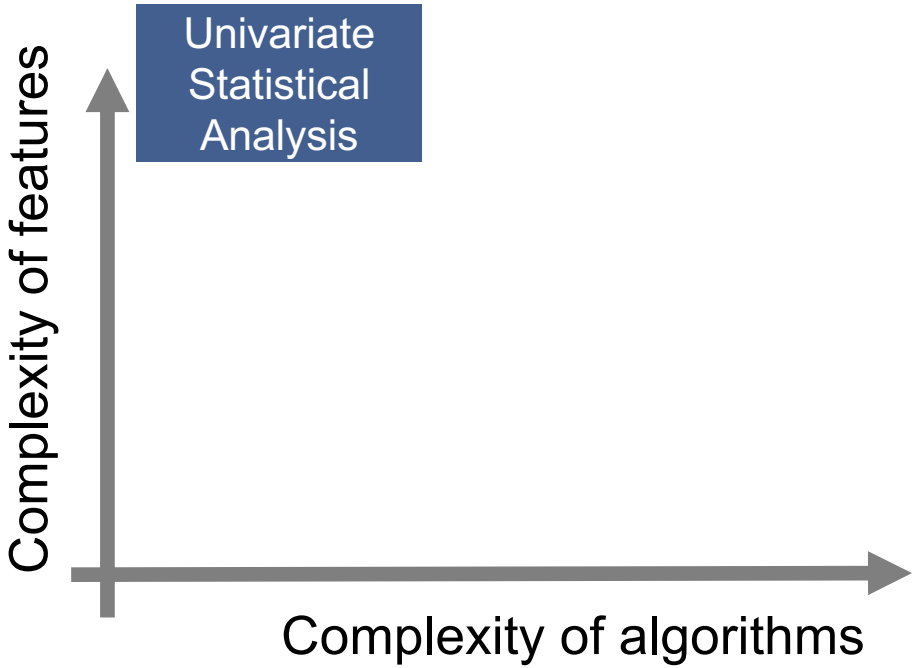

17 years old

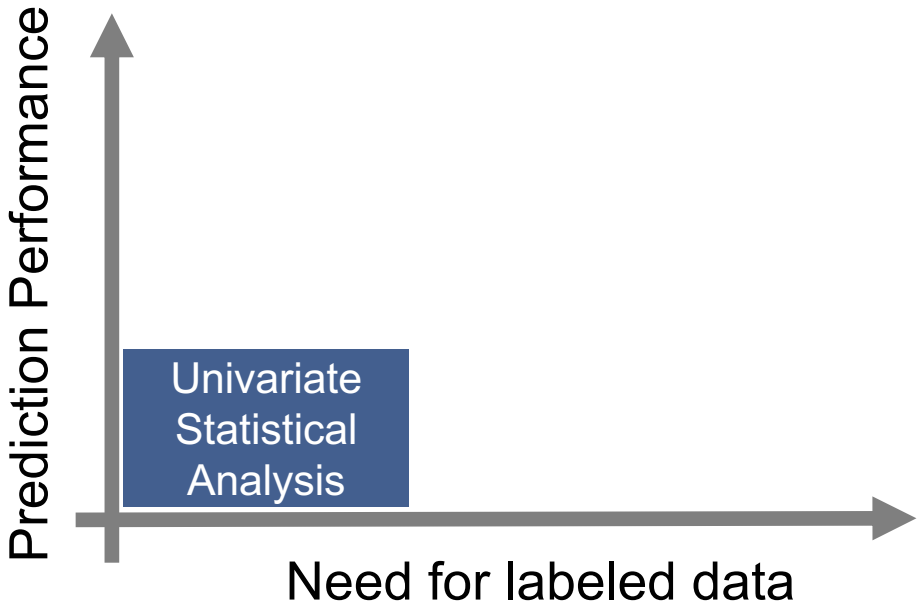

30 years old


74 years old


81 years old

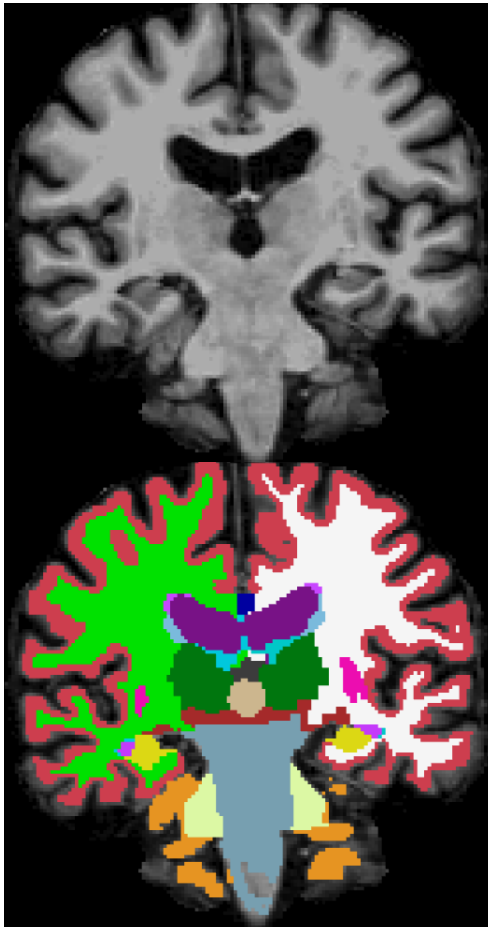

How old is she?





CVL Computer Vision Lab

Few measurements and predictive analysis



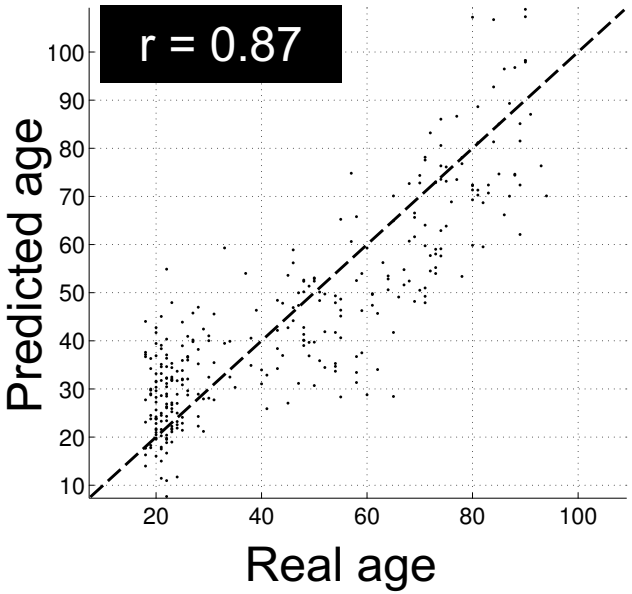
[Sabuncu and Konukoglu, 2015]

x: volume of all anatomical structures

SVM, K-nearest neighbors, decision trees, LDA, …

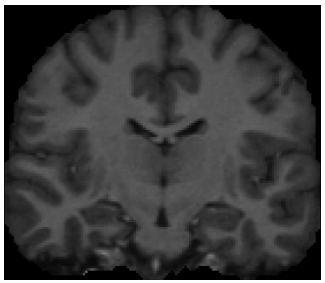
Example: $f(\mathbf{x}|\theta) = \theta^T \mathbf{x} + b$

- Hand-crafted measurements
- Multivariate
- Need to extract meaningful measurements
- Not trivial to extend to other problems

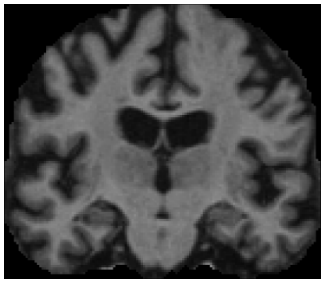


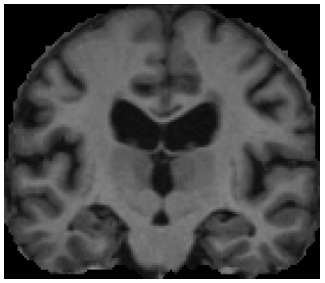
CVL Computer Vision Lab

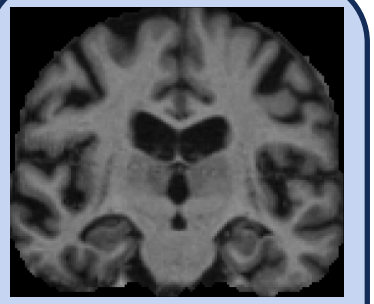
Different algorithms on a model problem

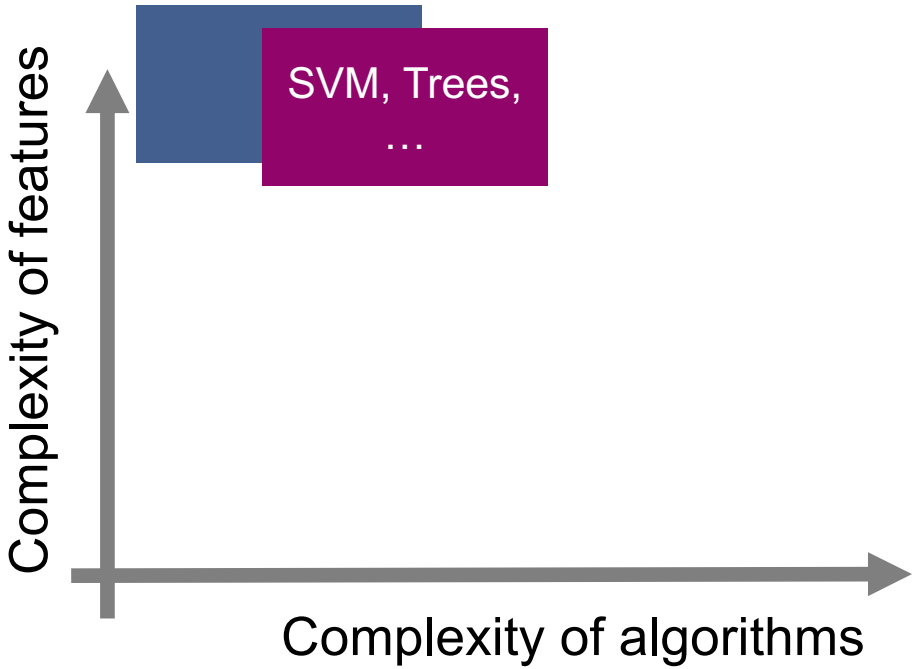

17 years old

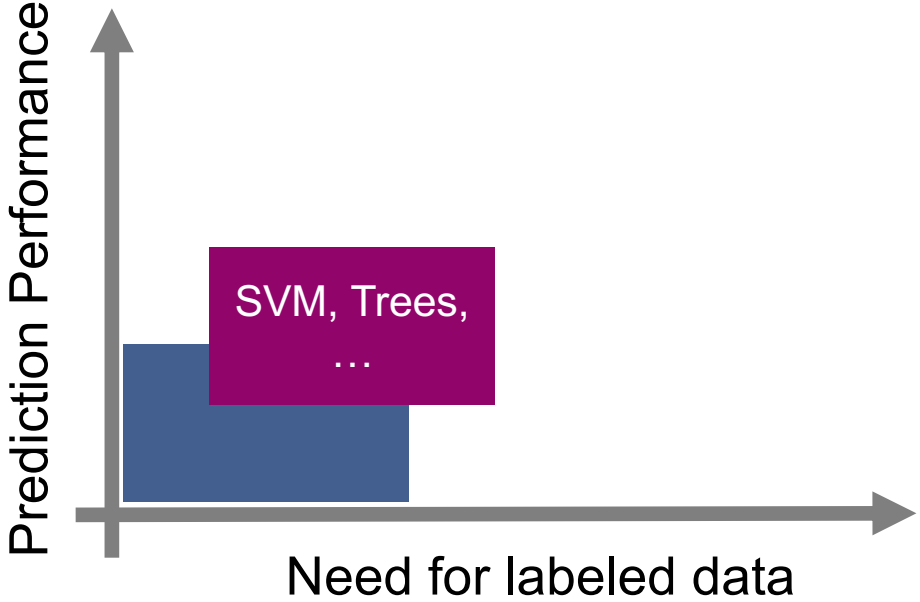

30 years old


74 years old


81 years old


How old is she?



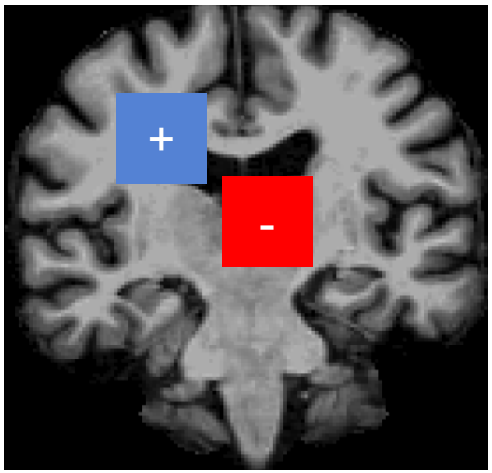


CVL Computer Vision Lab

Lots of measurements and feature selection

x: intensity differences between regions

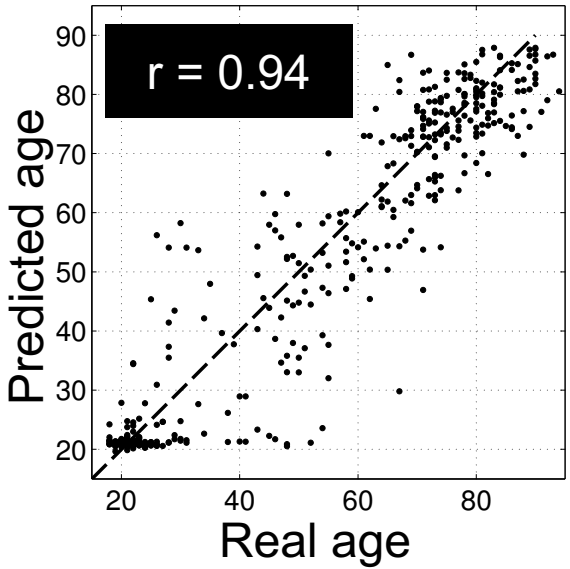
$$f(\mathbf{x}|\theta)$$ Random Forest, Boosting Trees, …



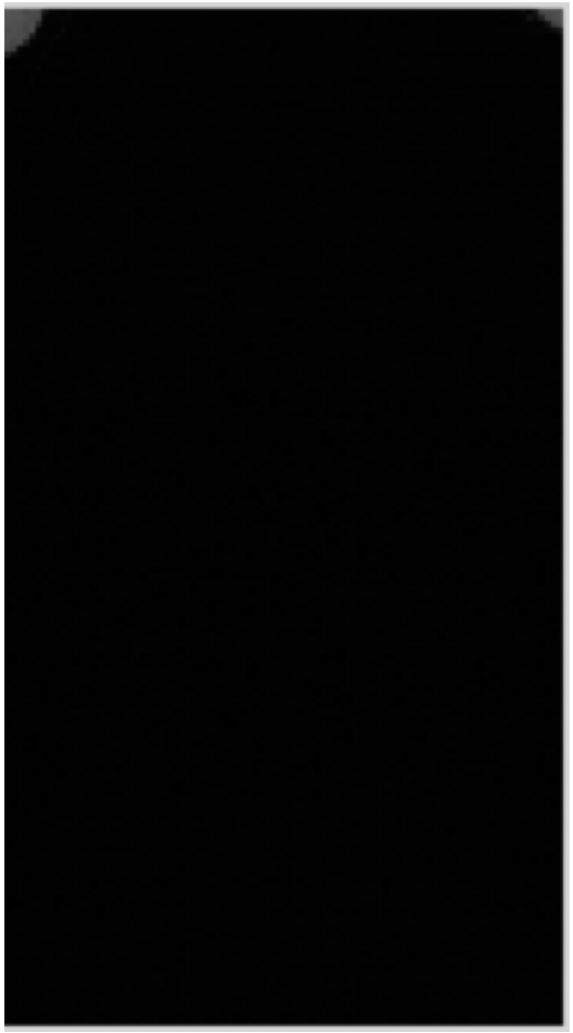
- Less hand-crafted and lots of measurements
- Automatic selection of **relevant** features
- Still hand-crafted to some extent
- Needs more labeled data than before

Easily extendable to other problems

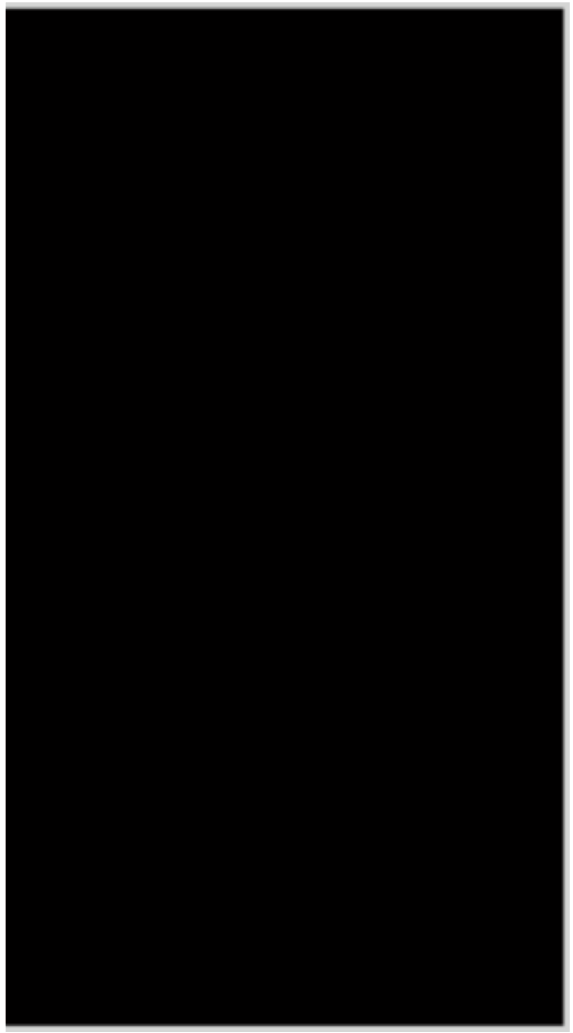
[Konukoglu et al. , 2013]



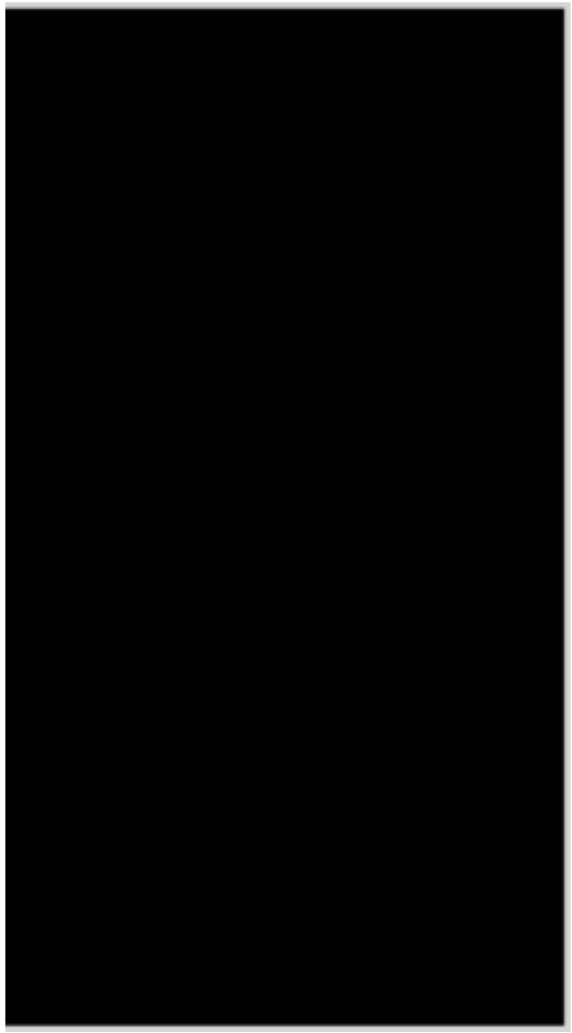
CVL Computer Vision Lab



CT Image



Manual labeling

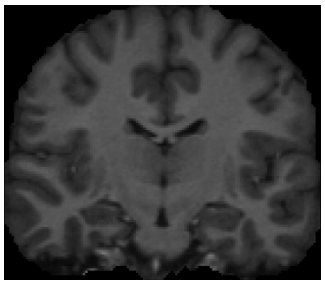


Algorithmic result

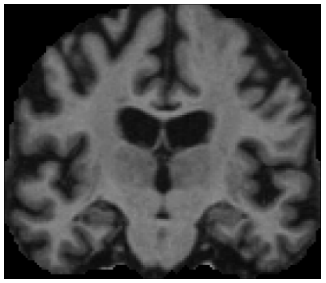
CVL Computer Vision Lab

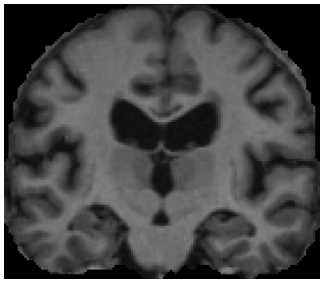
[Konukoglu et al. , 2013]

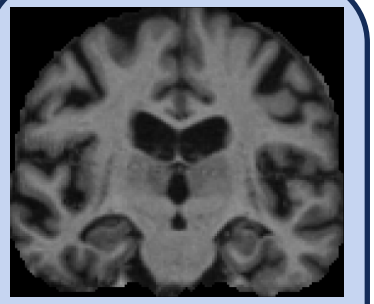
Different algorithms on a model problem

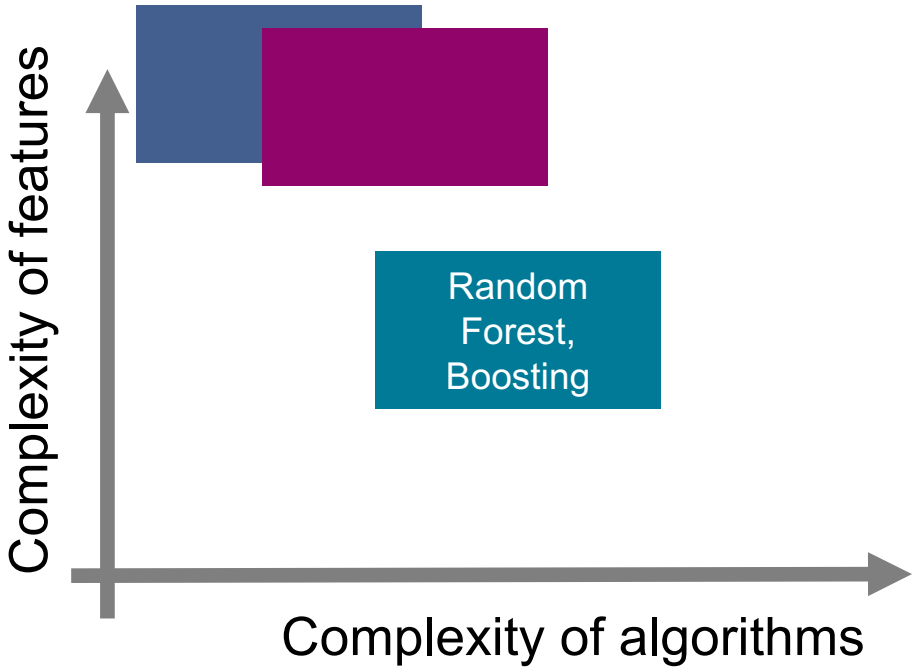

17 years old

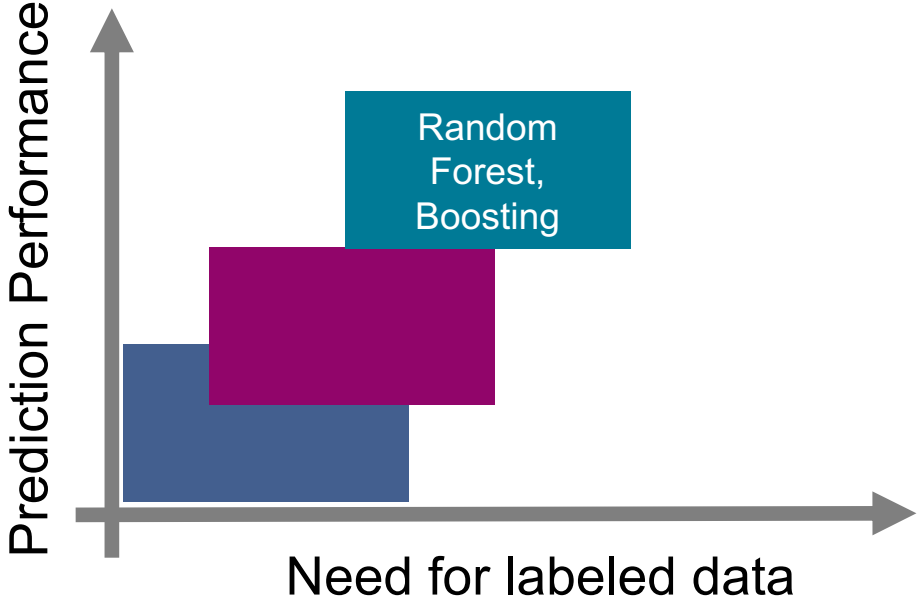

30 years old


74 years old


81 years old


How old is she?

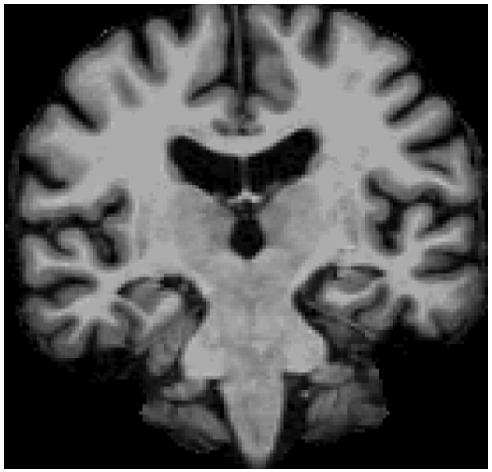




Automatic Feature Construction – Deep neural networks

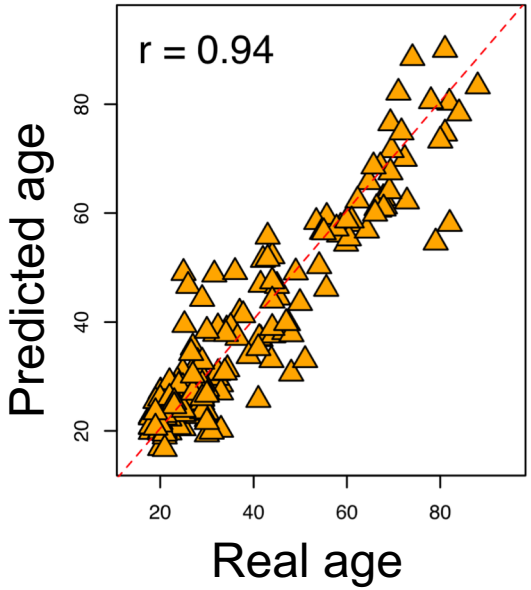
x: raw intensities / normalized intensities

$f(\mathbf{x}|\theta)$ Neural Networks
Convolutional Neural Networks



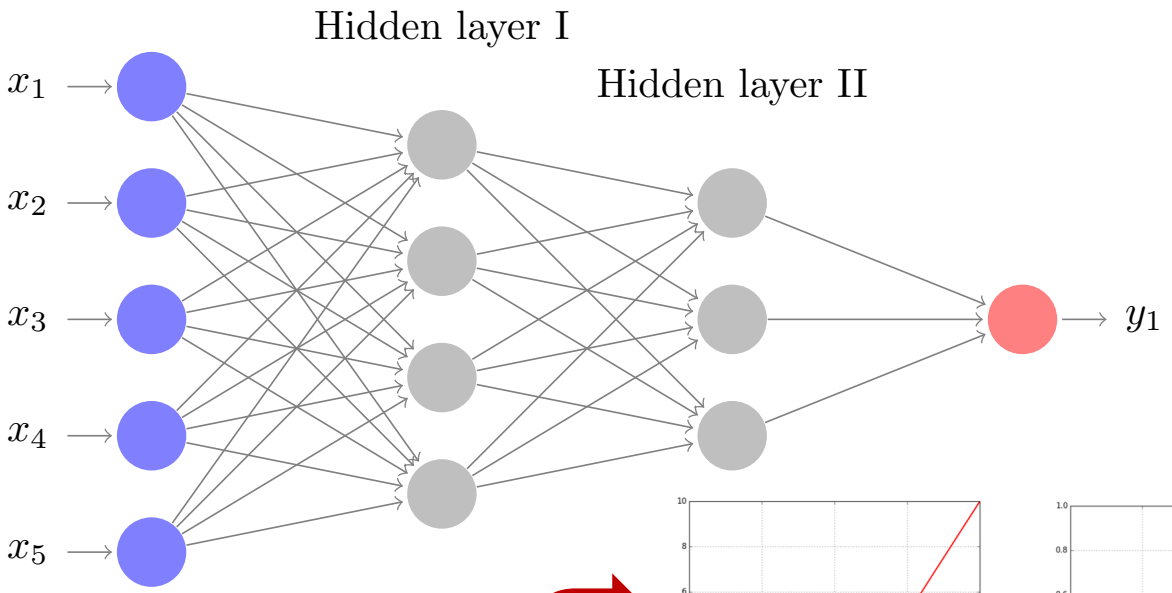
[Cole et al. arXiv, December 2016]

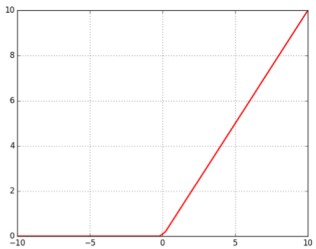
- Completely data-driven
- Automatic extraction of task specific features
- Not the first time this idea has been proposed
- Needs a LOT more labeled data

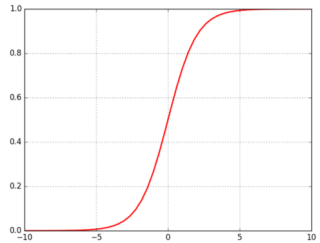


CVL Computer Vision Lab

The mapping and feature extractors







$$y = f(x|\theta)$$

$$= \sigma\left(W_3\sigma\left(W_2\sigma\left(W_1x + b_1\right) + b_2\right) + b_3\right)$$

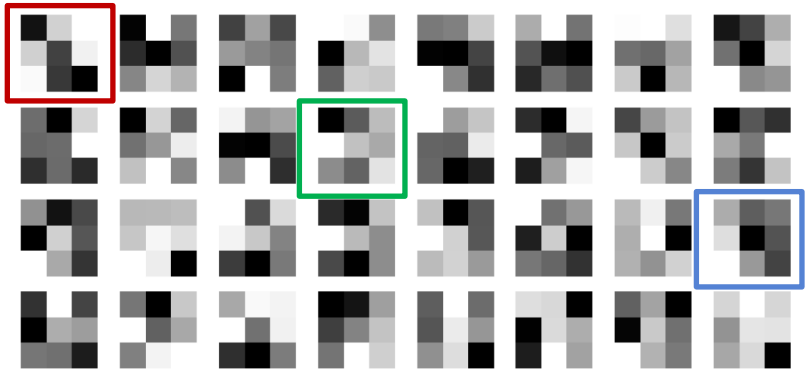
Task specific feature extractors

Segmentation with convolutional networks

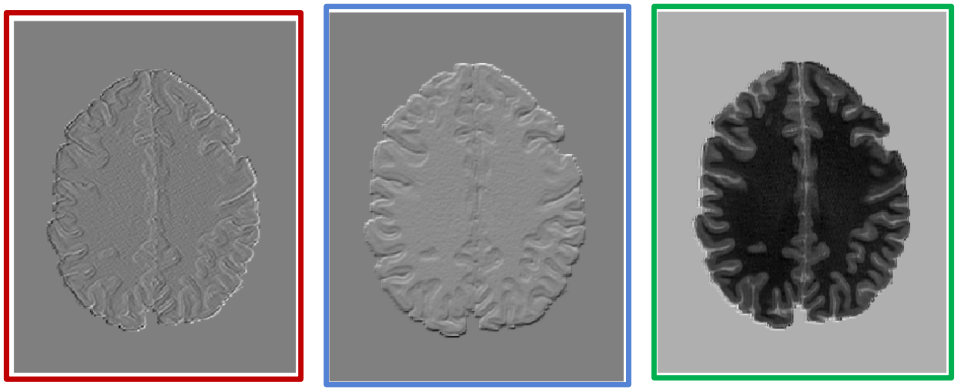
Image

Ground truth

Automatic Segmentation

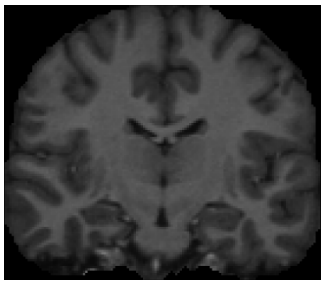


Task specific convolutional filters at layer 1

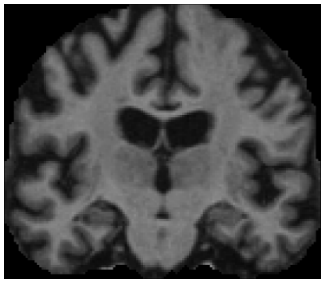


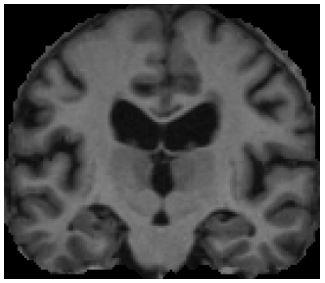
[K. Tezcan, K. Chaitanya]

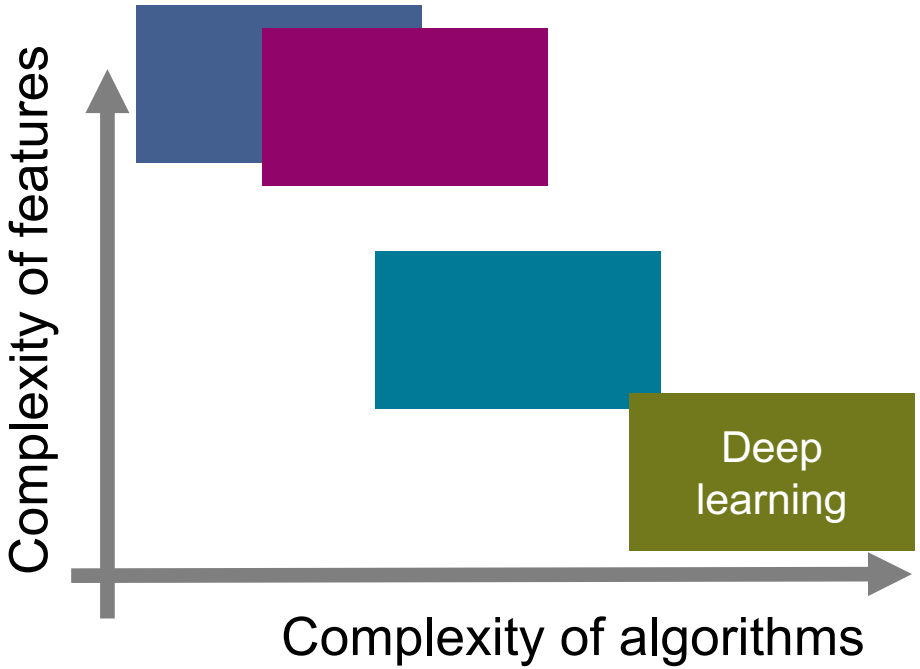
Different algorithms on a model problem

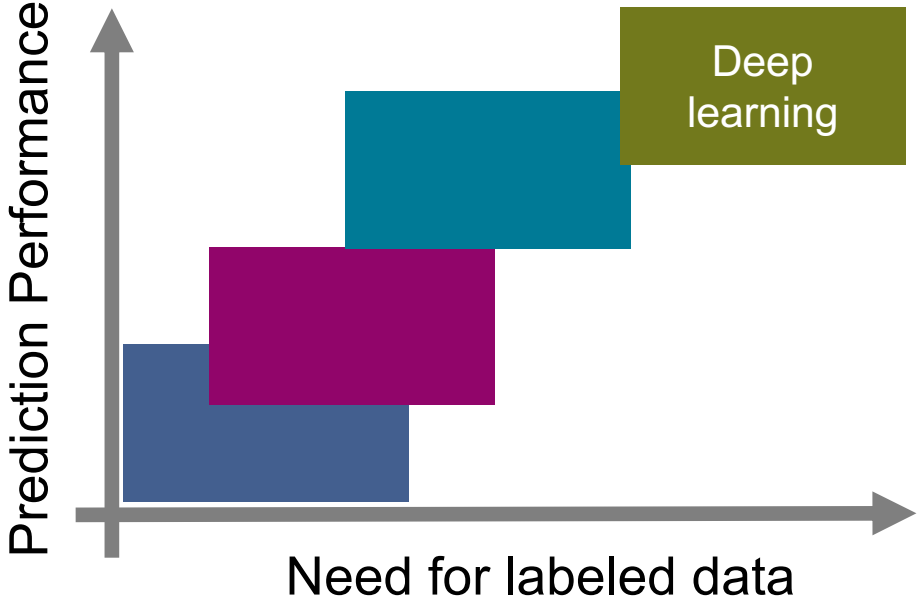

17 years old


30 years old


74 years old


81 years old





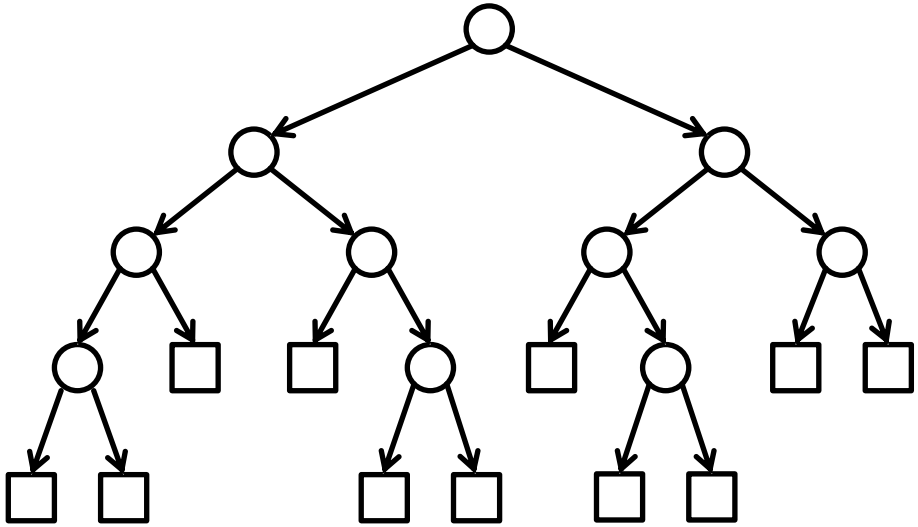
CVL Computer Vision Lab

Outline

- Elementary stuff
- Historical overview (supervised learning)
- Closer look at Random Forests
  - Basic algorithm
  - Same algorithm various tasks
  - Integrating context
  - Interpretability
  - [Amit & Geman 1997, Breiman 2001]
    [Criminisi, Shotton & Konukoglu 2011]
- Comparison to DL
- Cross-breeds

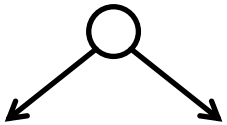
Binary decision trees

Hierarchically nested binary tests represented as a tree

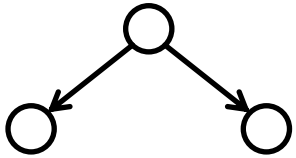


○ : Node – n, internal or root

□ : Leaf node – l

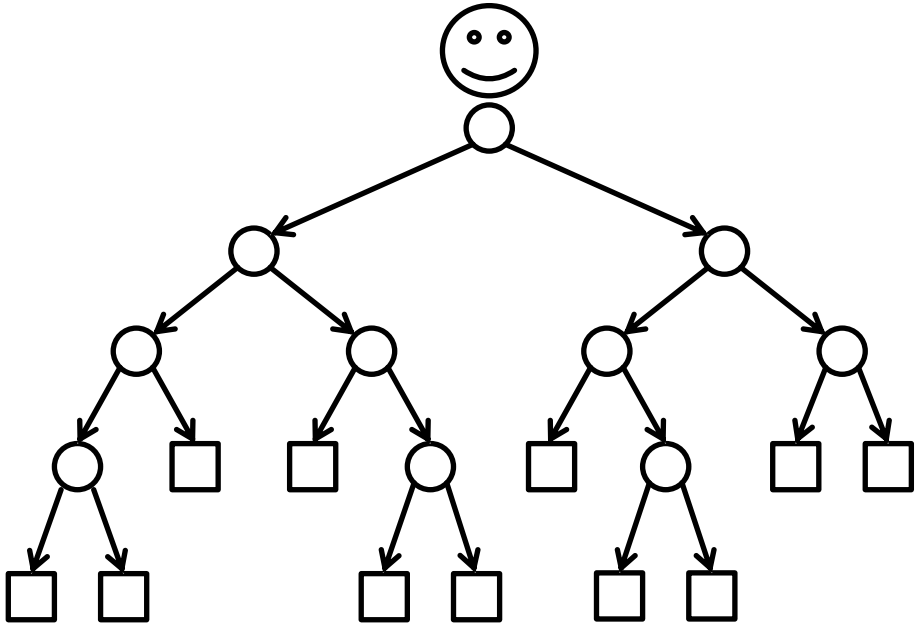


: Split – Binary test



Inference

A sample goes down the tree



A binary test at the root node and each internal node

$$\phi_n(\mathbf{x}) = \begin{cases} 0, & \text{go left} \\ 1, & \text{go right} \end{cases}$$

Leaf nodes hold predictions
Prediction depends on which leaf node sample lands on

$$f_t(\mathbf{x}|\theta) = y_t^l$$

Parameterization examples

$$f_t(\mathbf{x}|\theta) = y_t^l$$

Splits

Binary stump

$$\phi_n(\mathbf{x}) = \begin{cases} 0, & x_{j(n)} < \tau_n \\ 1, & x_{j(n)} \geq \tau_n \end{cases}$$

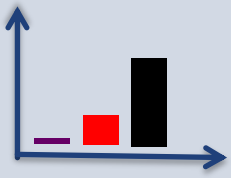
Oblique split

[Heath et al. 1993, Menze et al. 2011]

$$\phi_n(\mathbf{x}) = \begin{cases} 0, & \beta_n^T \mathbf{x} < \tau_n \\ 1, & \beta_n^T \mathbf{x} \geq \tau_n \end{cases}$$

Leaf node predictions

Histogram for classification



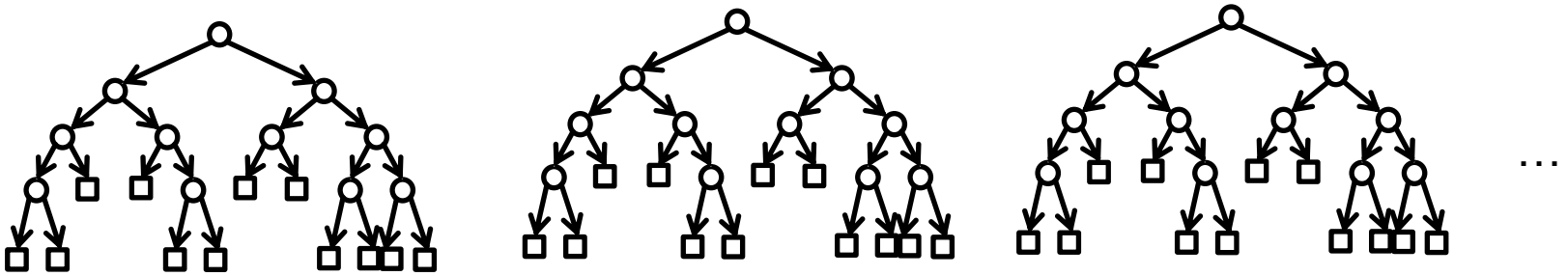
$$y_t^l = \arg\max_y p(y|t,l)$$

Normal distribution for regression

$$\mathcal{N}\left(\mu_t^l, \sigma_t^l\right)$$

$$y_t^l = \mu_t^l$$

Inference at the forest level – ensemble of weak learners



$$y_1^l = f_1(\mathbf{x}|\theta) \qquad y_2^l = f_2(\mathbf{x}|\theta) \qquad y_3^l = f_3(\mathbf{x}|\theta) \qquad \dots$$

Aggregate individual tree predictions

$$y_F = \mathcal{A}\left(\{f_t(\mathbf{x}|\theta)\}_{t=1}^{T}\right)$$

Examples

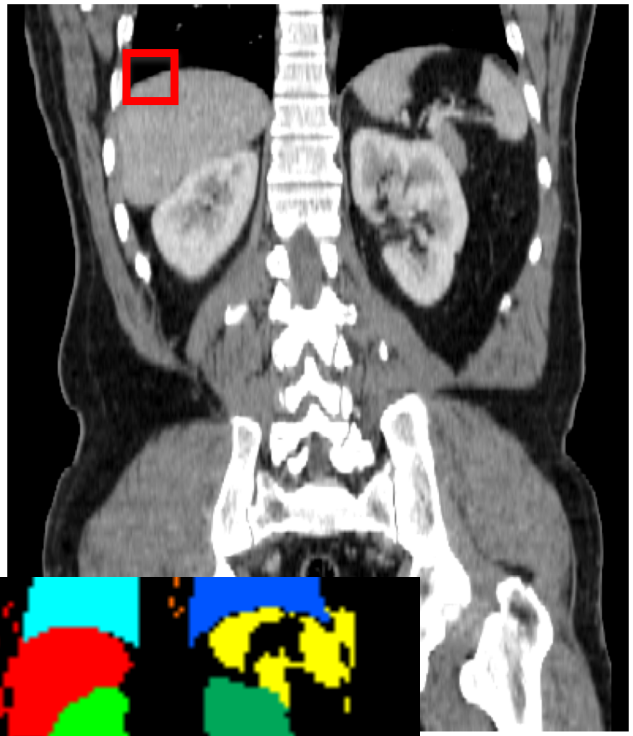
Classification

$$p(y_F = y) = \frac{1}{T}\sum_{t=1}^{T}\delta\left(f_t(\mathbf{x}|\theta) = y\right)$$

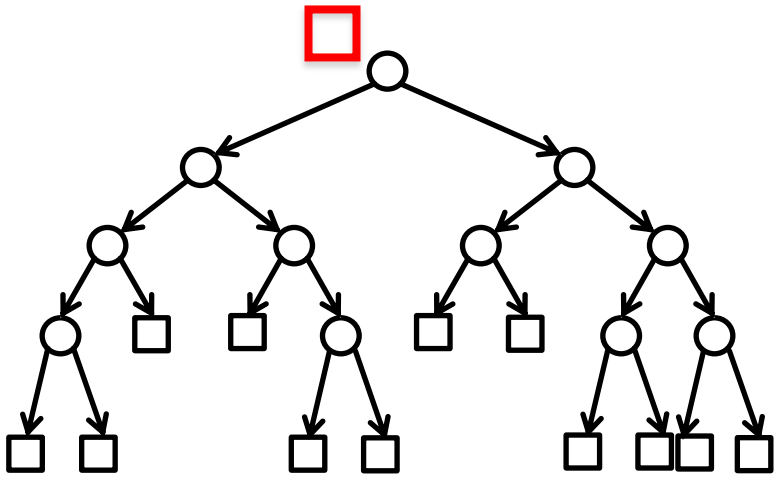
Regression

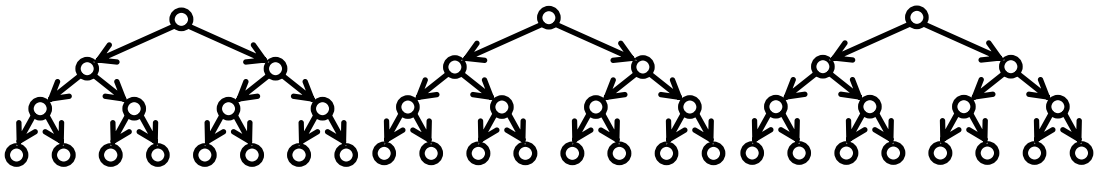
$$p(y_F = y) = \frac{1}{T}\sum_{t=1}^{T} f_t(\mathbf{x}|\theta)$$

Example: segmentation

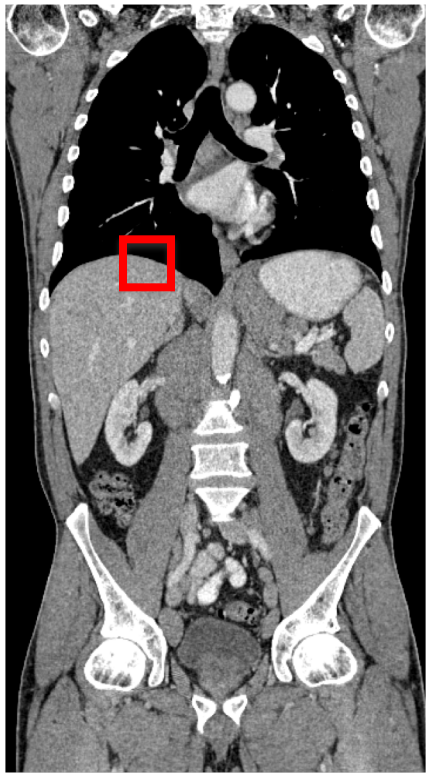


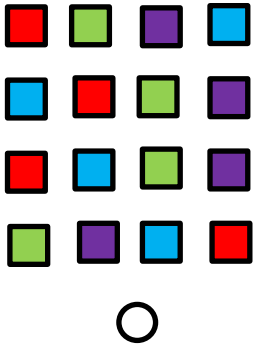
$$\square = \mathbf{x} = \{x_1, \ldots, x_D\}$$





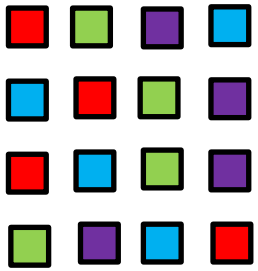
Training

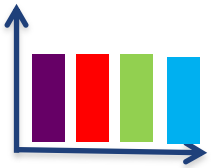


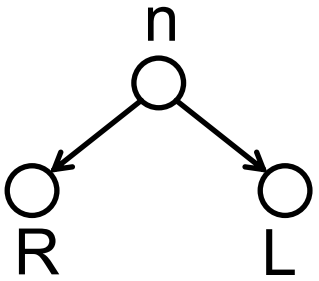


$= \mathbf{x}_1$ $= \mathbf{x}_2$ $= \mathbf{x}_S$

Node optimization







$$\phi_n(\mathbf{x}) = \begin{cases} 0, & x_{j(n)} < \tau_n \\ 1, & x_{j(n)} \geq \tau_n \end{cases}$$

$$\mathcal{D}_n = \left\{ (\mathbf{x}_s, y_s)_{s=1}^{S_n} \in \text{node n} \right\}$$

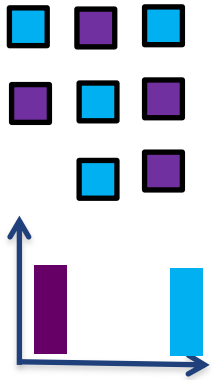
$$j(n), \tau_n = \arg\max_{j,\tau} \mathcal{L}(j, \tau)$$

$$\mathcal{L}(j,\tau) = H(\mathcal{D}_n) - \frac{|\mathcal{D}_R|}{|\mathcal{D}|} H(\mathcal{D}_R) - \frac{|\mathcal{D}_L|}{|\mathcal{D}|} H(\mathcal{D}_L)$$

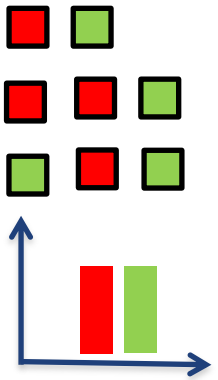
$$H(\mathcal{D}) = -\sum_{k=1}^{K} p(y = y_k|\mathcal{D}) \ln p(y = y_k|\mathcal{D})$$

$$p(y = y_k|\mathcal{D}) = \frac{|\mathbf{x} : \mathbf{x} \in \mathcal{D} \text{ and } y(\mathbf{x}) = y_k|}{\mathcal{D}}$$

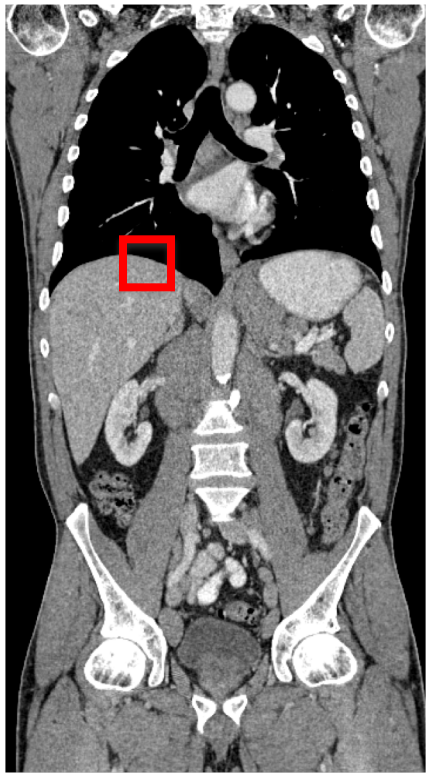
Left child

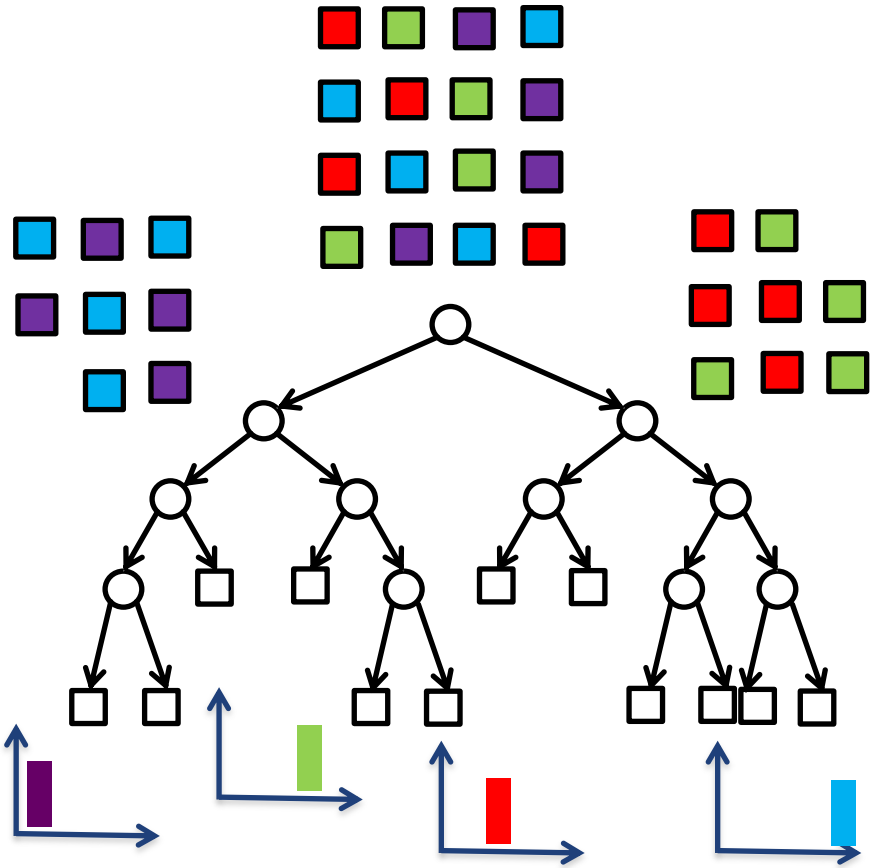


Right child



Training

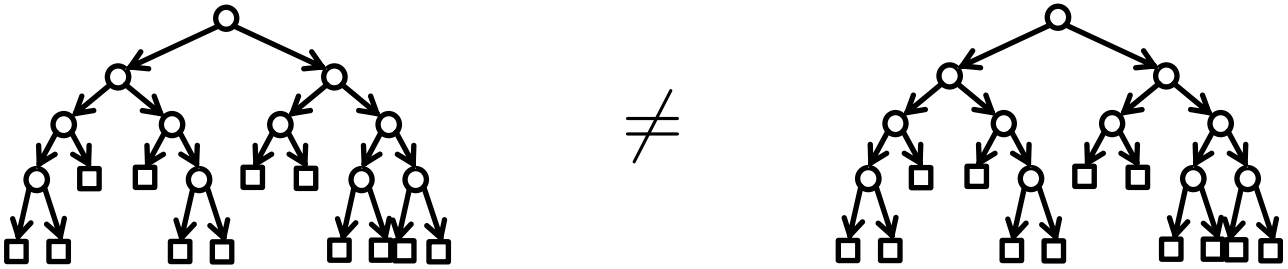




$\blacksquare = \mathbf{x}_1 \quad \blacksquare = \mathbf{x}_2 \quad \blacksquare = \mathbf{x}_S$

$$y_t^l = \arg\max_y p(y|t,l)$$

The “Random” in the forest



Feature subsampling

Entire feature space

$$\mathbf{x} = \{x_1, \ldots, x_D\} \in \mathbb{R}^D$$

Each tree sees a random subspace

$$\mathbf{x}_t = \{x_{t(1)}, \ldots, x_{t(d)}\} \in \mathbb{R}^d \subset \mathbb{R}^D$$

Bagging

Entire training dataset

$$\mathcal{D} = \{(\mathbf{x}_s, y_s)\}_{s=1}^{S}$$

Each tree sees a random subset

$$\mathcal{D}_t = \left\{(\mathbf{x}_{t(s)}, y_{t(s)})\right\}_{s=1}^{S_t}, \; S_t < S$$

Empirical finding: Randomization helps avoid overfitting and improve generalization

Different costs – different tasks

$$\mathcal{L}(j,\tau) = H(\mathcal{D}_n) - \frac{|\mathcal{D}_R|}{|\mathcal{D}|} H(\mathcal{D}_R) - \frac{|\mathcal{D}_L|}{|\mathcal{D}|} H(\mathcal{D}_L)$$

Classification

$$H(\mathcal{D}) = -\sum_{k=1}^{K} p(y = y_k|\mathcal{D}) \ln p(y = y_k|\mathcal{D})$$

Semisupervised
[Criminisi et al. 2011]

Multivariate regression

$$H(\mathcal{D}) = \frac{1}{|\mathcal{D}|} \sum_{\mathcal{D}} (\mathbf{y}_s - \mu)(\mathbf{y}_s - \mu)^T$$

Joint regression and Classification
[Glocker et al. 2012]

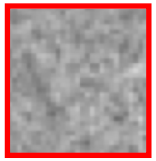
Unsupervised clustering

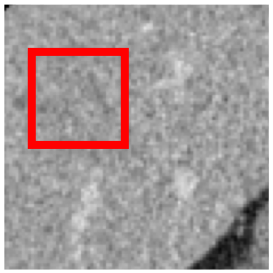
$$H(\mathcal{D}) = \frac{1}{|\mathcal{D}|} \sum_{\mathcal{D}} (\mathbf{x}_s - \mu)(\mathbf{x}_s - \mu)^T$$

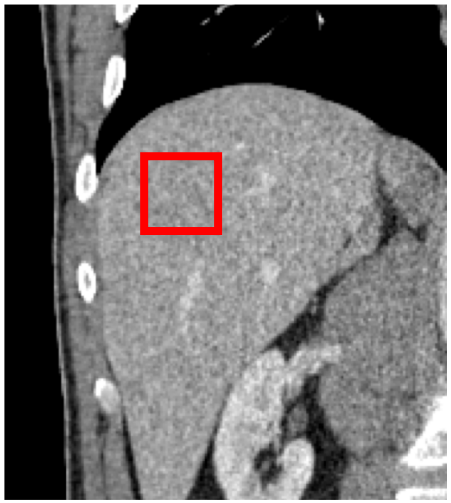
Supervised clustering [EK et al. 2013]

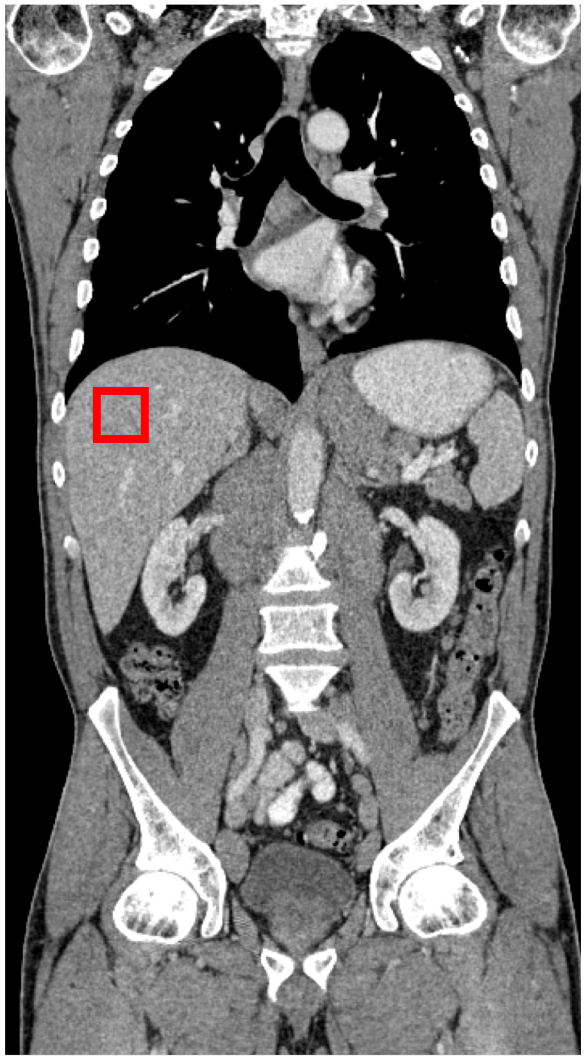
$$H(\mathcal{D}) = \frac{1}{|\mathcal{D}|} \sum_{s \in \mathcal{D}} \sum_{r \in \mathcal{D}} \rho(s,r)$$

Context

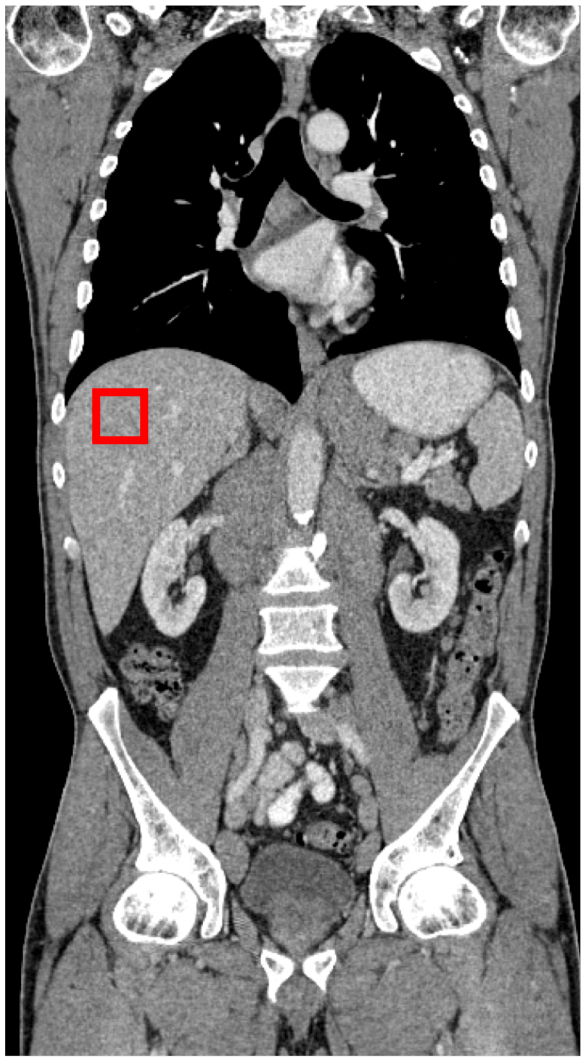


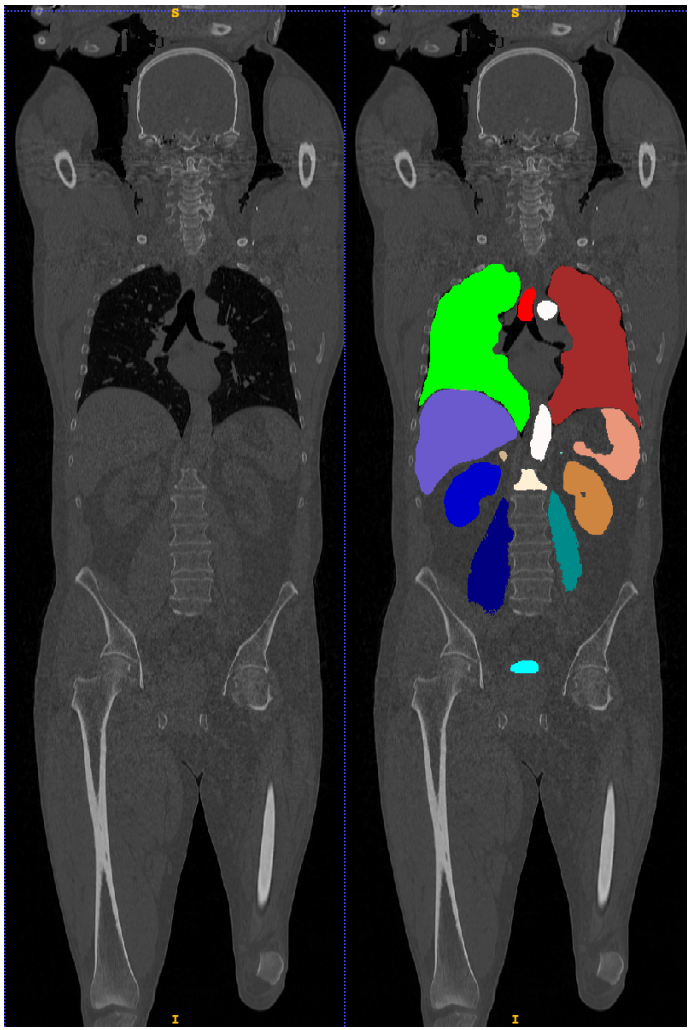




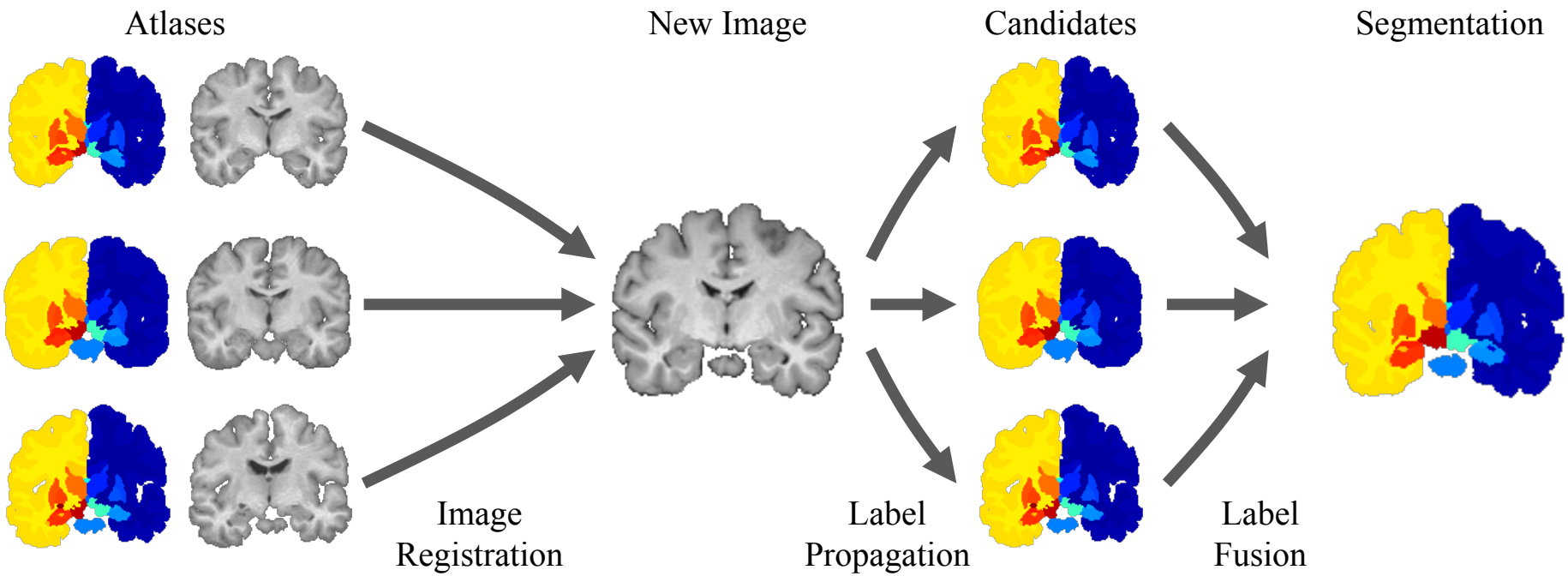


Context – atlas-based methods

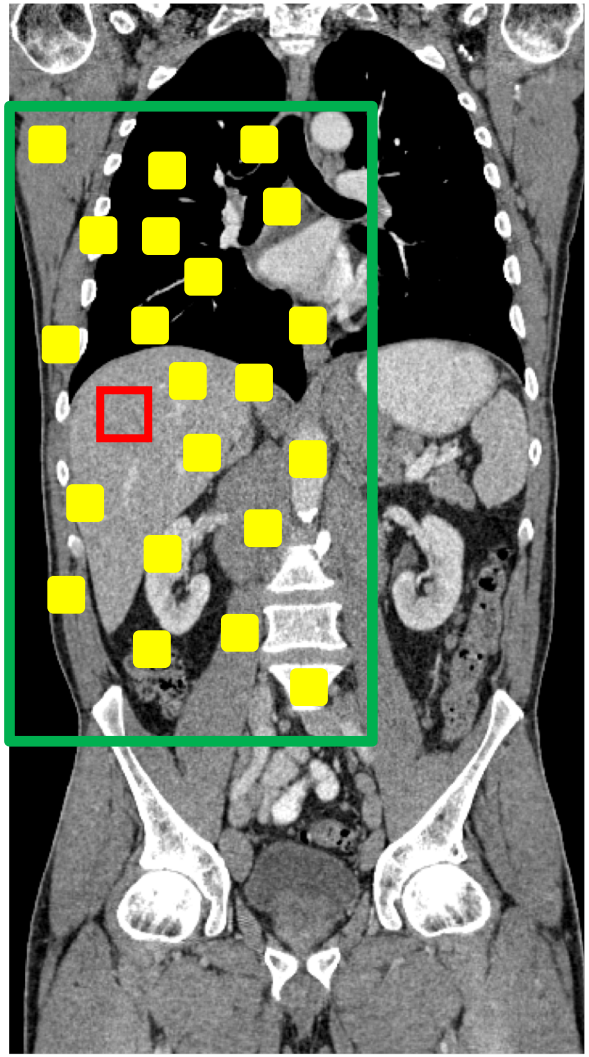




Segmentation through registration



Context – Feature selection



- Many simple features [Viola & Jones 2001]

  Haar-like intensity differences

  Simple averages

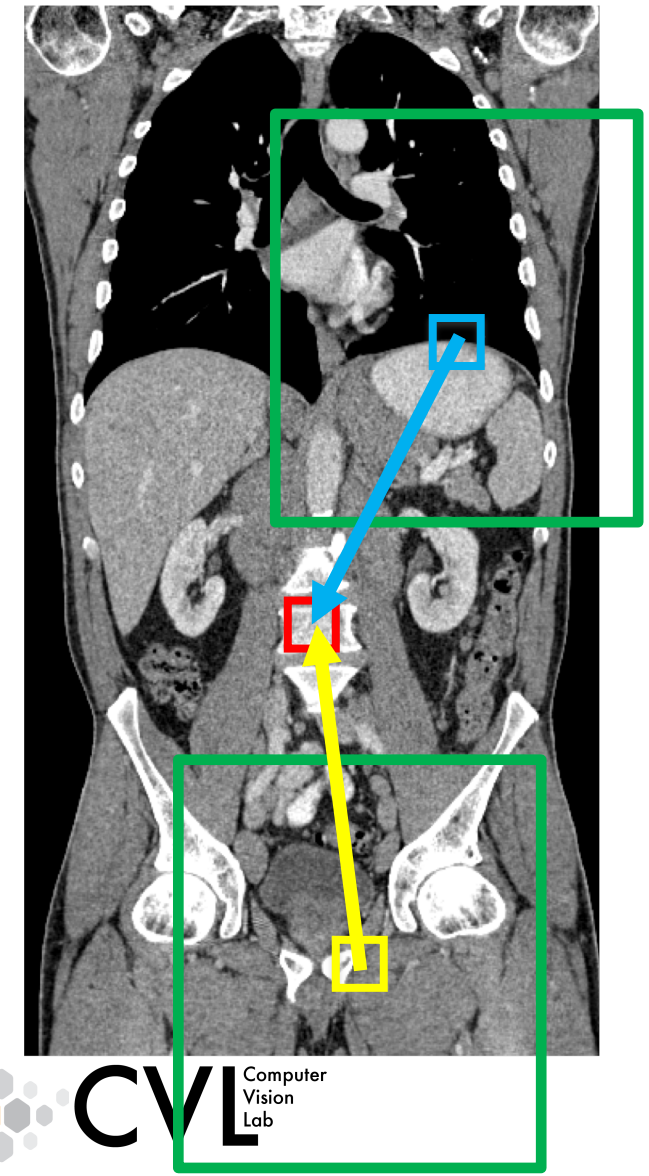
  Local binary patterns [Pauly et al. 2011]
- Node optimization in RF chooses what feature to use – no need to define good features

$$j(n), \tau_n = \arg\max_{j,\tau} \mathcal{L}(j, \tau)$$

- RF is not the only algorithm that solves it this way, see Boosting.

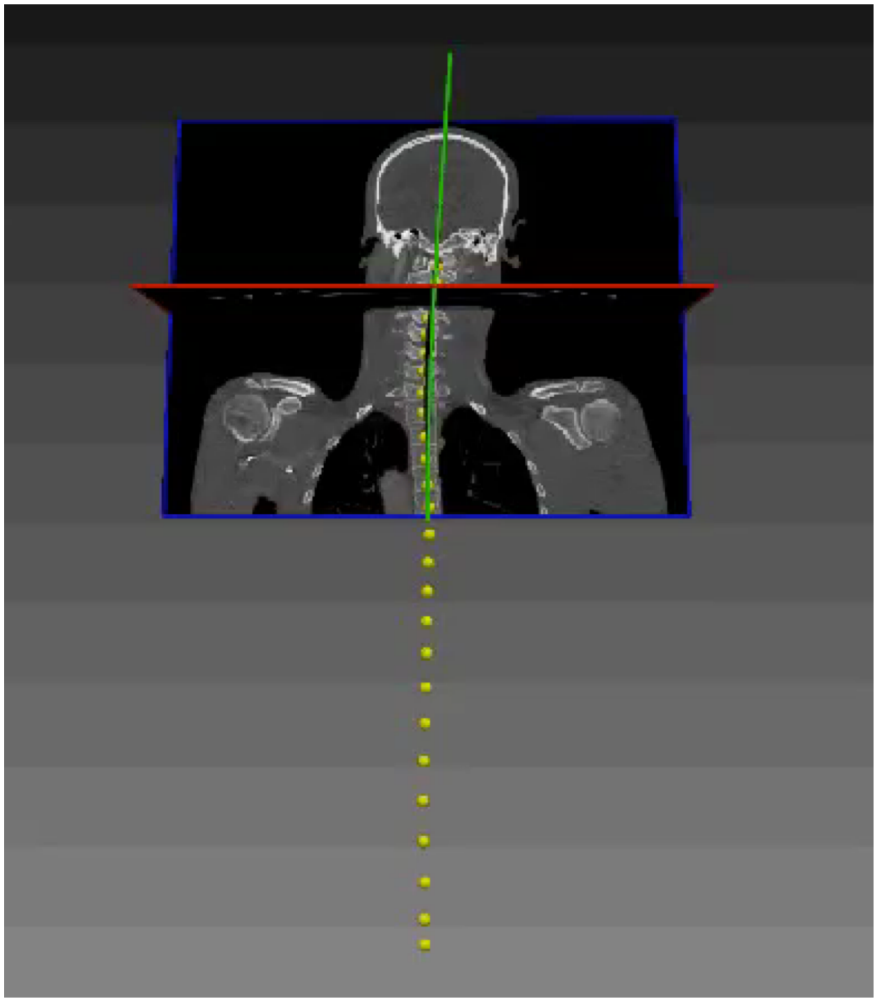
CVL Computer Vision Lab

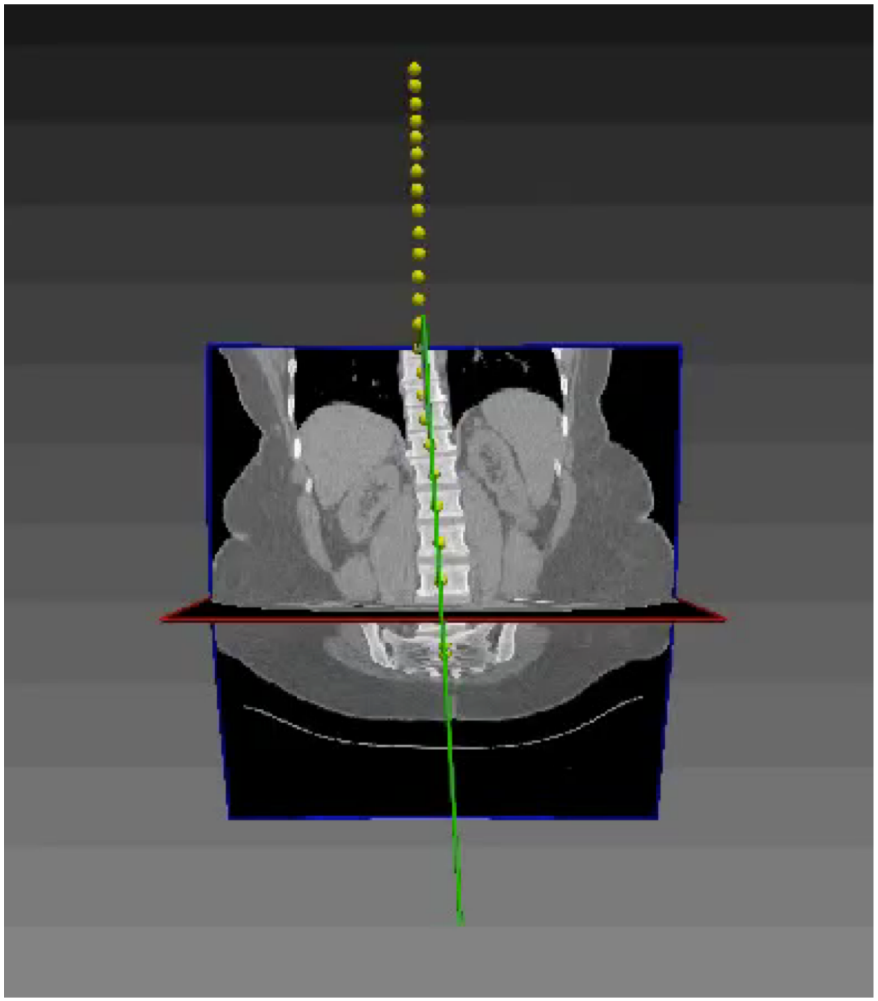
Context – Hough transform type voting



- Every voxel makes a prediction [Gall et al. TPAMI 2011]
- Useful for localization and detection of normal anatomical structures
- Voxels use their context for the prediction

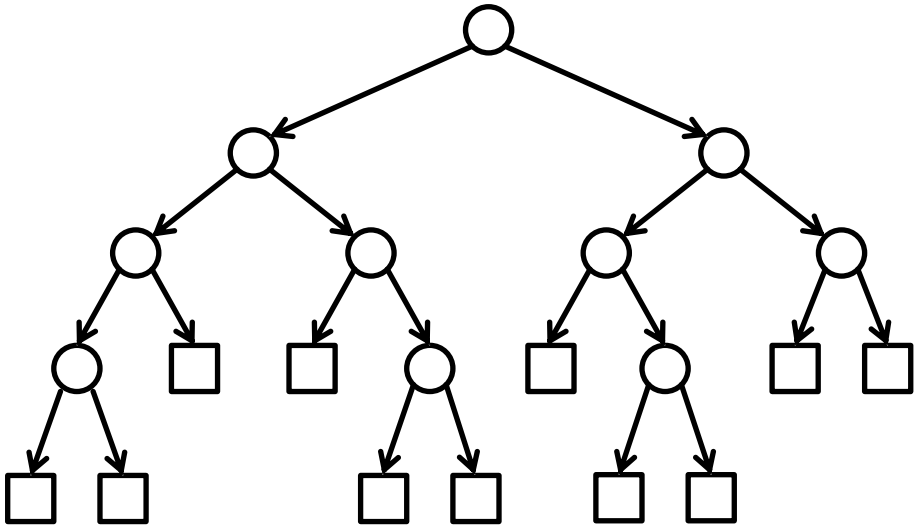
Detecting and predicting anatomical structures

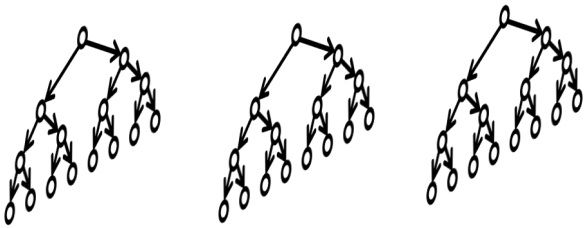




[Glocker et al. , 2013]

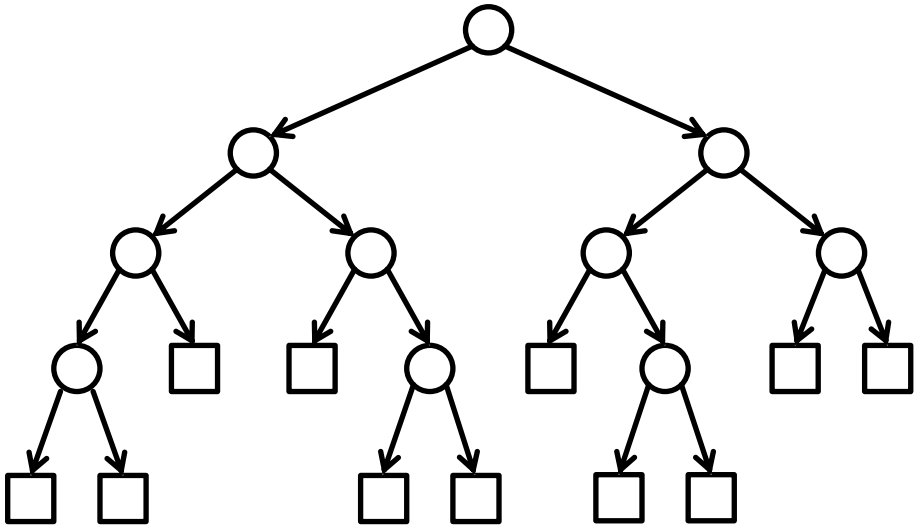
Interpretability





- Interpretation = is the feature useful for prediction?
- Decision trees are interpretable
- At each level, we can visualize the test being used
- Random forests are less interpretable due to multiple trees
- Feature selection is essential in identifying relevant features among many

Importance measures



- Various alternative measures that “quantify” feature selection

Selection Frequency

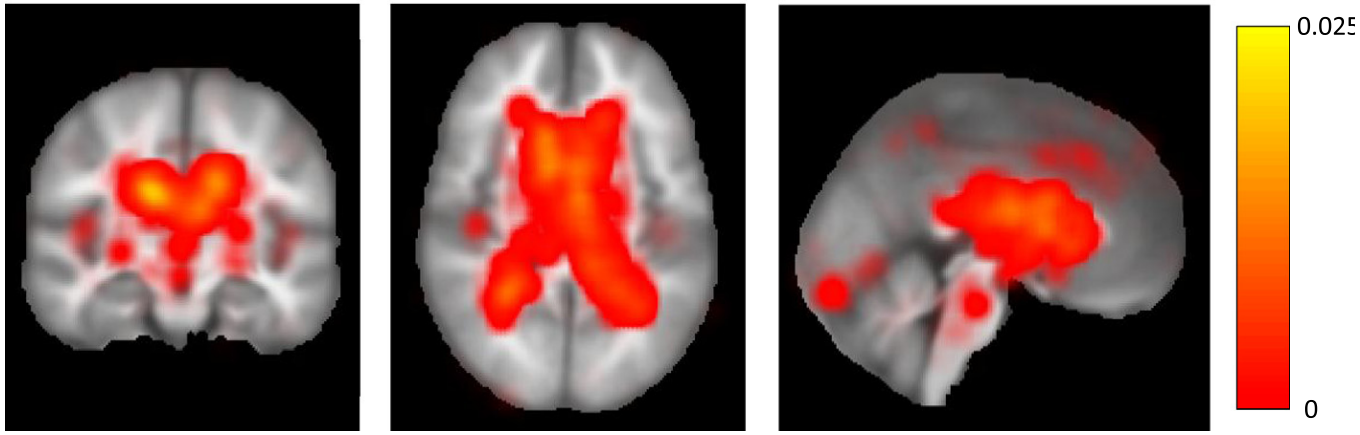
$$I_{SF}(x) = \sum_T \sum_n \mathbf{1}(x = x_n)$$

Cost-based, e.g. Gini importance

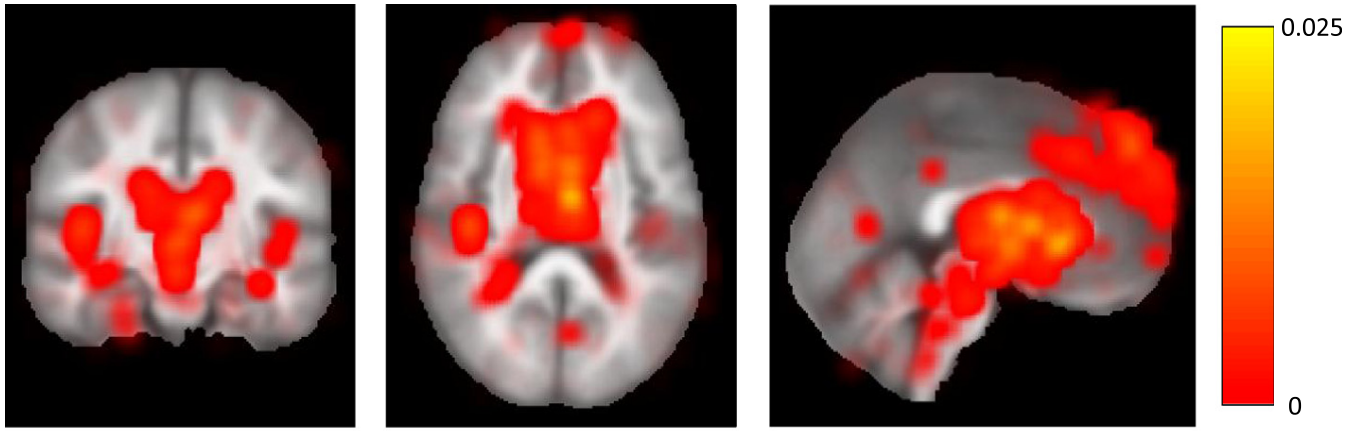
$$I_{GI}(x) = \sum_T \sum_n \mathbf{1}(x = x_n)\mathcal{L}(x_n, \tau_n^*)$$

Permutation testing:
Permute the feature of interest across the dataset
Retrain the Random Forest
Evaluate drop in prediction accuracy

Example: Age and brain



Intensity clustering selected features



Age regression selected features

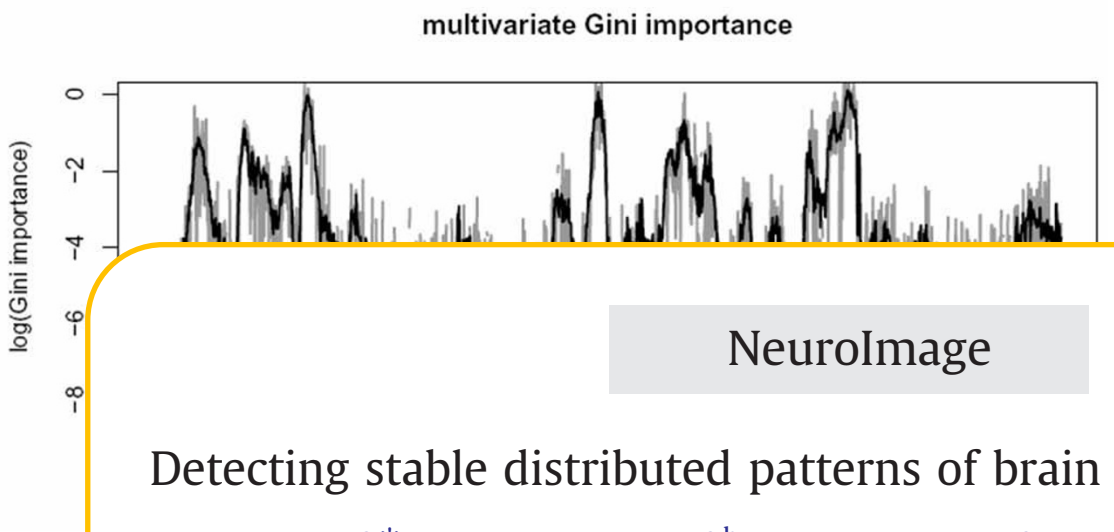
[Konukoglu et al. MedIA 2013]

Applications

BMC Bioinformatics

A comparison of random forest and its Gini importance with standard chemometric methods for the feature selection and classification of spectral data

Bjoern H Menze[1,2], B Michael Kelm[1], Ralf Masuch[3], Uwe Himmelreich[4], Peter Bachert[5], Wolfgang Petrich[6,7] and Fred A Hamprecht*[1,6]



NeuroImage

Detecting stable distributed patterns of brain activation using Gini contrast

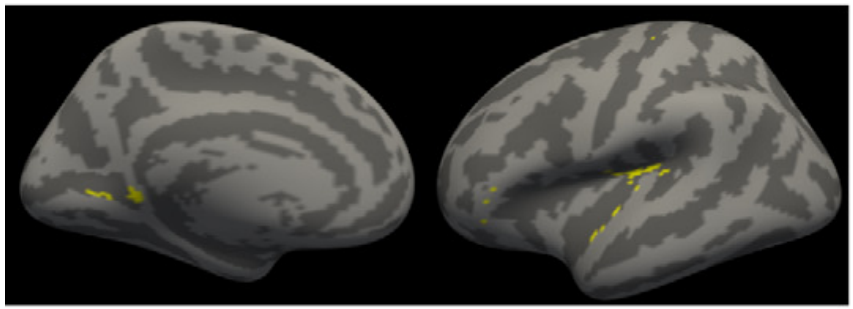
Georg Langs [a,*], Bjoern H. Menze [a,b], Danial Lashkari [a], Polina Golland [a]

[a] *Computer Science and Artificial Intelligence Laboratory, Massachusetts Institute of Technology, Cambridge, MA, USA*
[b] *Asclepios Research Project, INRIA Sophia-Antipolis, France*

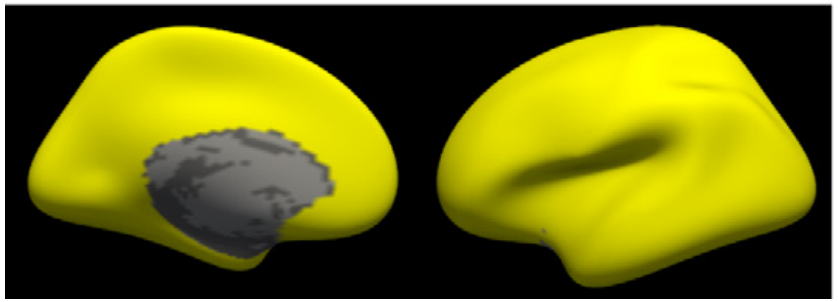
Extra

- It is possible to compute false positive rates for feature selection frequency
  [Konukoglu and Ganz 2014, arXiv:1410.2838]
- Selected set is not complete!
  Knock-out procedure similar to gene studies
  [Ganz et al. Neuroimage 2015]

Features predictive of aging



Set determined by RF Gini Importance
top 85% of selected features



Set determined by Knock-out

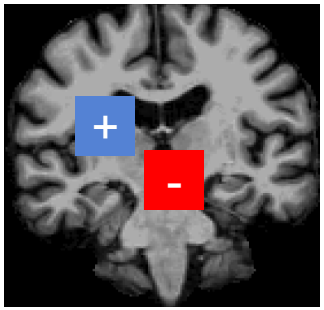
Outline

- Elementary stuff
- Historical overview (supervised learning)
- Closer look at Random Forests
- Comparison to DL
  - Feature selection vs. representation learning
  - Hierarchical partitioning vs combining different attributes
  - Context integration vs. receptive field
  - Ensembles vs. Drop-out
  - Interpretability
- Cross-breeds

Feature selection vs. representation learning

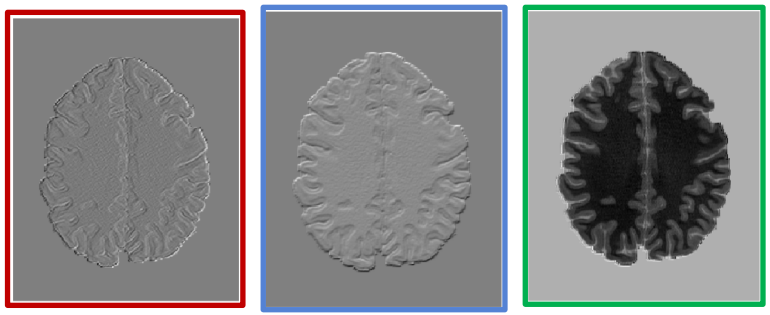
Random forest

- Forest use hand-crafted features
- Very simple but lots
- Select important features at nodes
- Greedy selection – suboptimal
- Fast to extract even in CPU



Deep learning

- Learn task-specific better features
- Simple features in one layer
- Complicated features when combined through layers
- Global optimization – less suboptimal features
- Often requires GPU

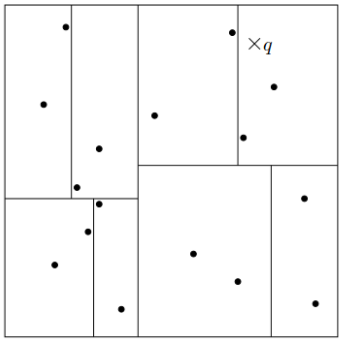


Partitioning vs combining attributes

[Balestriero arXiv 2017]

Random forest

- Hierarchical partitioning of the feature space
- Divide a complex problem into successive binary problems
- Many regions for each class different parts of a tree separates different regions



[Image taken from Balestriero 2017]

Deep learning

- Extracting information so that classes are linearly separable (think about the final layer)
- Extract and combine many different attributes in non-linear functions
- Solves one big problem
- In final separation, one region per class, all samples in the same region

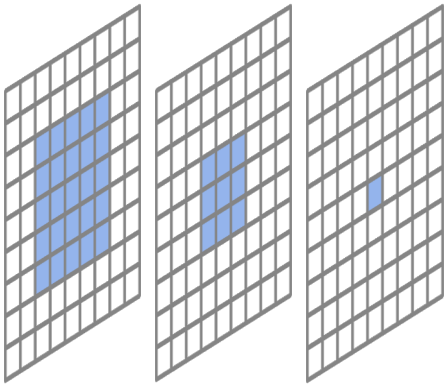
Context integration vs. receptive field

Random forest

- Contextual features
- Representing each point with random features at different offsets
- Selects relevant features to take into account part of the context that is useful

Deep learning

- Receptive fields
- Neurons at deeper layers “sees” a larger portion of an image
- Pooling, dilated convolution,…
- Extracts relevant features to take into account context for the task



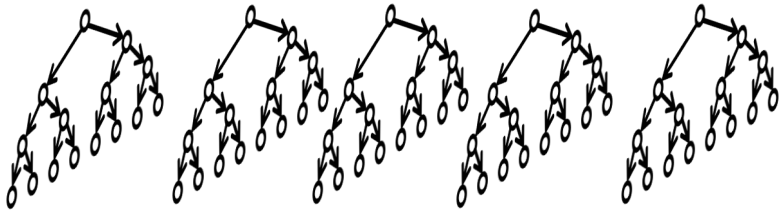
Ensemble vs. drop-out

Random forest

- By construction
- Many independent trees
- Random feature subsampling
- Bagging
- Aggregation
- Better generalization

Deep learning

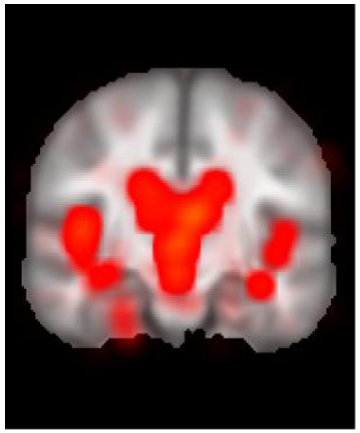
- Drop-out
- At test-time it emulates different networks
- Aggregation either at each node or at the final round
- Is there bagging? Is there use for it?
- Explicit feature randomization?



Interpretability

Random forest

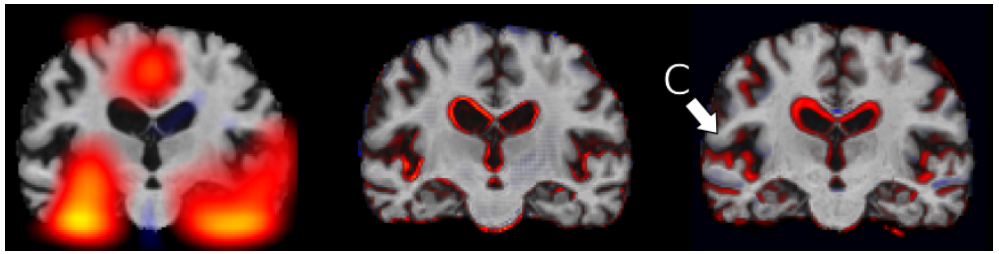
- Feature selection based importance measures
- False positive rate limits [Konukoglu and Ganz 2014]
- Various strategies to improve [Ganz et al. 2015]



[Image taken from Konukoglu et al. 2013]

Deep learning

- Less interpretable in general
- Saliency maps
- Class activation mapping
- Visual attribution [Baumgartner et al. 2018]



CAM VAGAN Observed

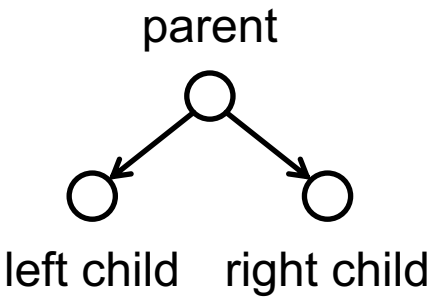
[Image taken from Baumgartner et al. 2018]

Outline

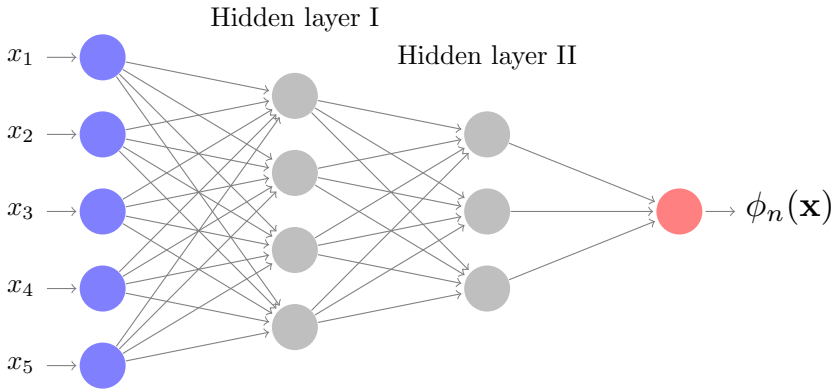
- Elementary stuff
- Historical overview (supervised learning)
- Closer look at Random Forests
- Comparison to DL
- Cross-breeds
  - Neural decision forests for semantic image labeling [Bulo and Kontschieder, CVPR 2014]
  - Deep neural decision forest, Neural decision trees [Kontschieder et al. ICCV 2015], [Balestriero arXiv 2017]

Changing the node split

[Bulo & Kontschieder CVPR 2014]



Using MLP as a non-linear split function



Traditional deterministic split function

e.g.

$$\phi_n(\mathbf{x}) = \begin{cases} 0, & \beta_n^T \mathbf{x} < \tau_n \\ 1, & \beta_n^T \mathbf{x} \geq \tau_n \end{cases}$$

$$\theta^* = \arg\max_{\theta} \mathcal{L}(\theta)$$

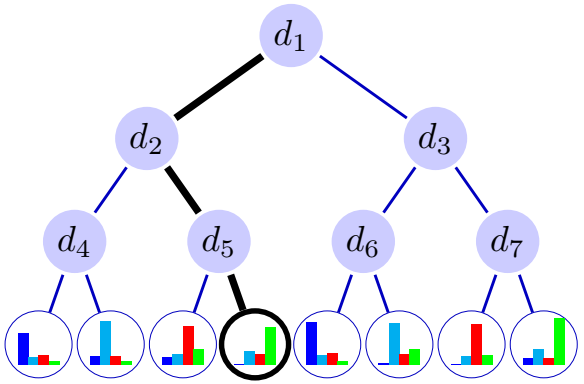
- Soft-split function instead of deterministic
- Appropriate cost function

$$\mathcal{L}(\theta) = \max_{(\pi_R, \pi_L)} \prod_{s=1}^{S} p\left(y_s | \mathbf{x}_s, \underbrace{(\pi_R, \pi_L)}, \theta\right)$$

$$p\left(y_s|\pi_R\right) \phi_n(\mathbf{x}_s|\theta) + p\left(y_s|\pi_L\right)\left(1 - \phi_n(\mathbf{x}_s|\theta)\right)$$

Deep neural decision forests – differentiable trees [Kontschieder et al. ICCV 2015]

- Global optimization not greedy
- Differentiable trees



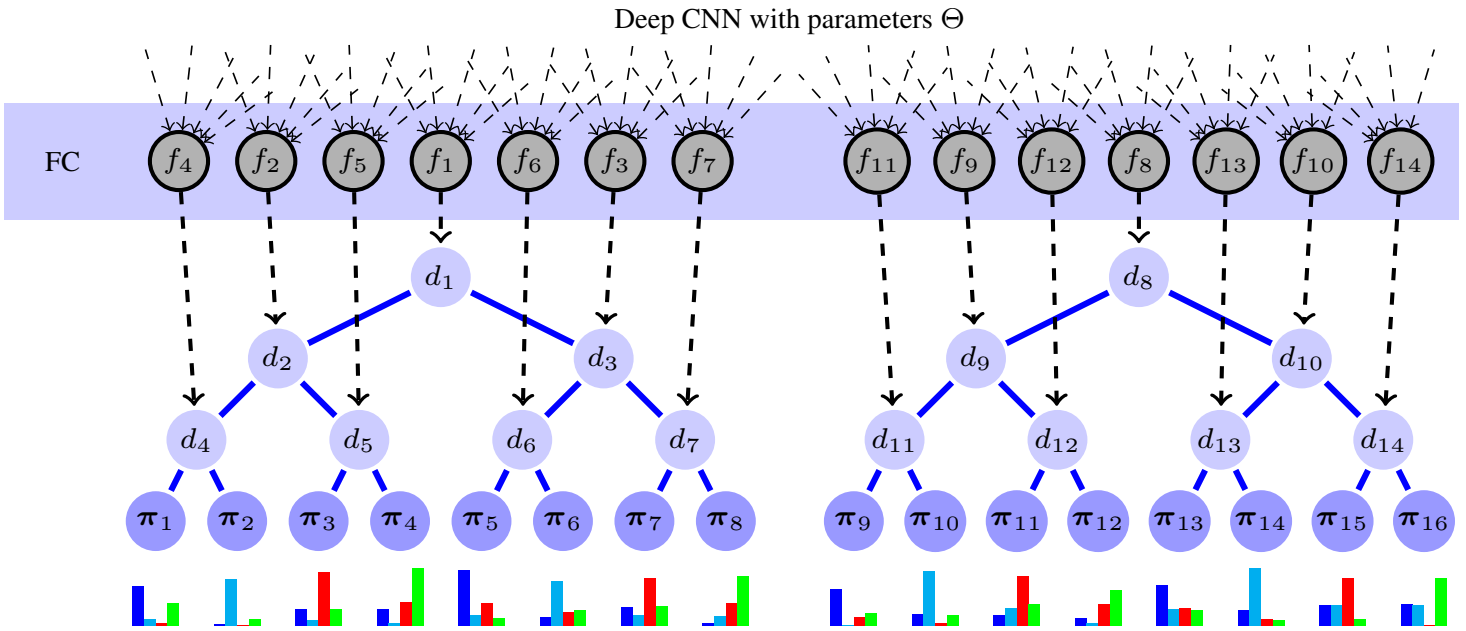
Predictive distribution for a fixed tree

$$p_t(y|\mathbf{x},\pi,\theta) = \sum_{l\in t} p(y|\pi_l)\mu_l(\mathbf{x}|\theta)$$

- Stochastic routing

$$\mu_l(\mathbf{x}|\theta) = \prod_n \phi_n(\mathbf{x}|\theta)^{\delta(l\swarrow n)}(1-\phi_n(\mathbf{x}|\theta))^{\delta(n\searrow l)}$$

Representation learning and hierarchy



- Representation learning in trees
- For each split, a CNN extracts the task-specific feature
- Hierarchy: Each output node of the CNN focuses on a subproblem defined by the tree structure
- [Balestriero 2017] uses a very similar idea and extends to unsupervised learning

ETH zürich

Thank you

Questions?

ender.konukoglu@vision.ee.ethz.ch
www.vision.ee.ethz.ch/~kender