

Tensor Ensemble Learning

Authors:

Ilija Kisil, Ahmad Moniri, Danilo P. Mandic

Electrical and Electronic Engineering Department
Imperial College London, SW7 2AZ, UK

E-mails: {i.kisil15, ahmad.moniri13, d.mandic}@imperial.ac.uk

November 28, 2018

- Wisdom of the crowd
- Ensemble learning and existing algorithms
- Multidimensional representaiton of data
- Basics of tensor decompositions
- Tensor Ensemble Learning (TEL)
- Simulations and results

Ensemble learning: The wisdom of the crowd

"I'LL ASK THE AUDIENCE!"



"CAN I PHONE A FRIEND?"



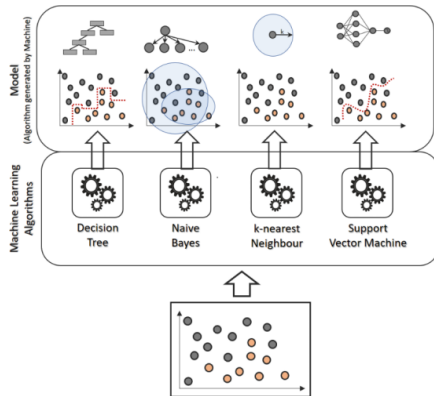
A game show segment featuring a host with a beard and a blue shirt. A bar chart is overlaid on the screen, showing the results of a poll. The chart has four bars labeled A, B, C, and D, with percentages 7%, 24%, 55%, and 14% respectively. A small icon of three people is above the chart. Below the chart, a question is displayed in a blue box, followed by four answer options in yellow boxes.

Option	Percentage
A	7%
B	24%
C	55%
D	14%

What's the most delicious thing to put in a sandwich?

- A: Money
- B: A smaller sandwich
- C: Dead animals
- D: Your grandmother's hair

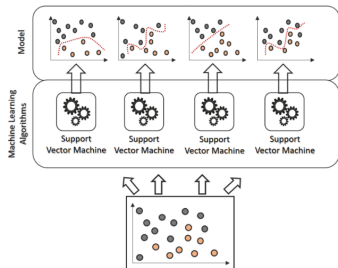
Ensemble learning: Motivation and limitations



- Every model has its own weaknesses \rightarrow Combining different models can find a better hypothesis
- Every model explores its own hypothesis space \rightarrow Robust to outliers
- Strong assumption that our individual errors are uncorrelated

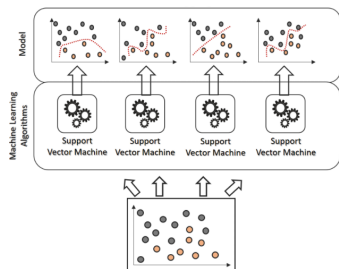
Ensemble learning: Existing approaches

Bagging

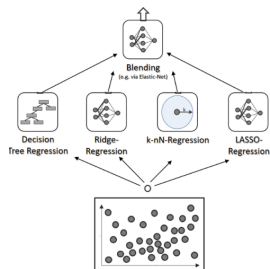


Ensemble learning: Existing approaches

Bagging

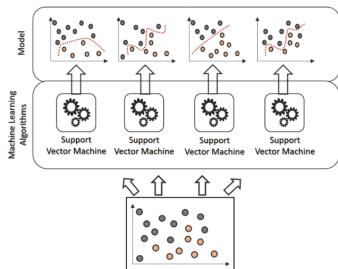


Stacking

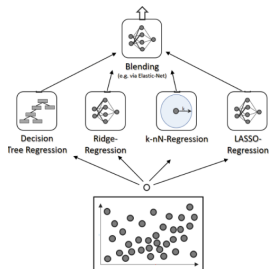


Ensemble learning: Existing approaches

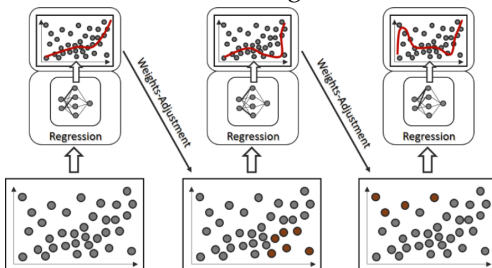
Bagging



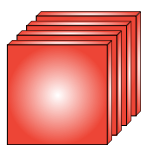
Stacking



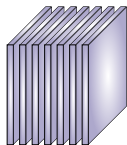
Boosting



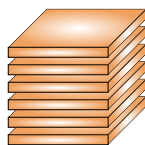
Tensors and basic sub-structures



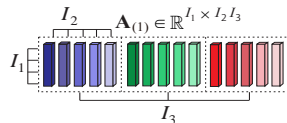
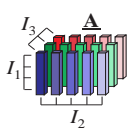
$\underline{\mathbf{A}}_{(:, :, k)}$



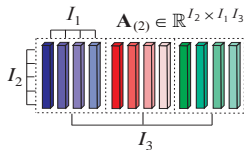
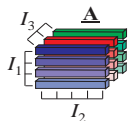
$\underline{\mathbf{A}}_{(:, j, :)}$



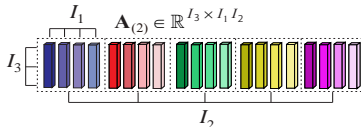
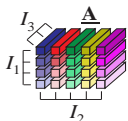
$\underline{\mathbf{A}}_{(i, :, :)}$



mode-1 unfolding

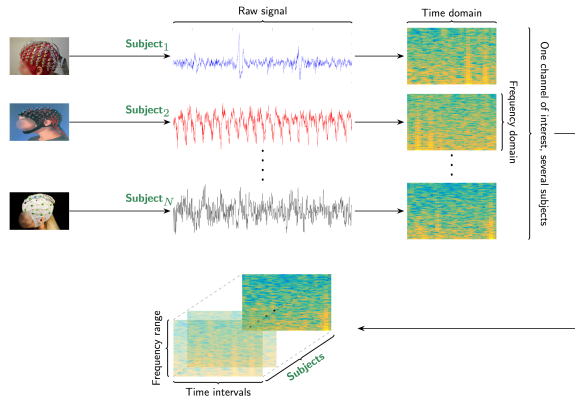
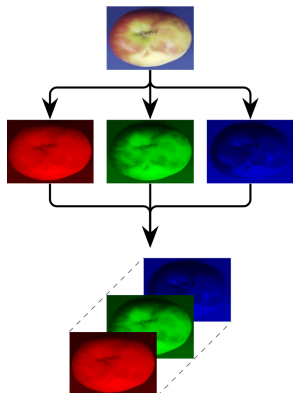
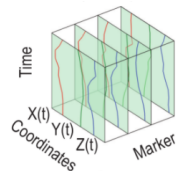
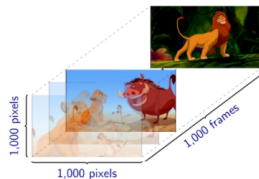
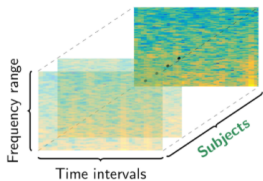


mode-2 unfolding

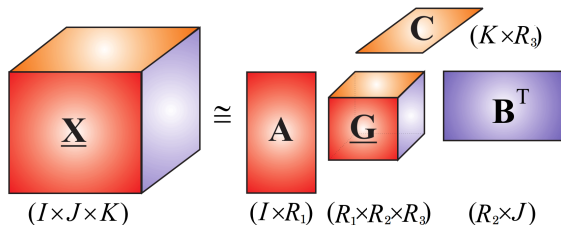


mode-3 unfolding

Multidimensional data and tensor construction



Tucker decomposition \leftrightarrow HOSVD

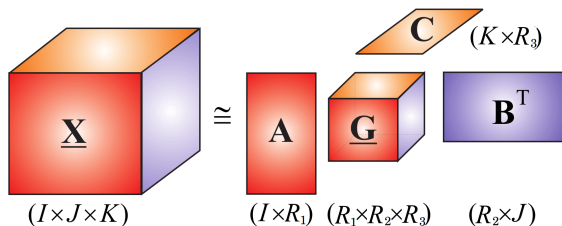


- Each vector of $\underline{\mathbf{A}}$ is associated with every vector of $\underline{\mathbf{B}}$ and $\underline{\mathbf{C}}$ through the

$$\text{core tensor } \underline{\mathbf{G}} \leftrightarrow \underline{\mathbf{X}} \approx \sum_{r_1=1}^{R_1} \sum_{r_2=1}^{R_2} \sum_{r_3=1}^{R_3} \underline{\mathbf{G}}_{r_1 r_2 r_3} \cdot \mathbf{a}_{r_1} \circ \mathbf{b}_{r_2} \circ \mathbf{c}_{r_3}$$

- In general, the Tucker decomposition is not unique
- But the subspaces spanned by vectors of $\underline{\mathbf{A}}$, $\underline{\mathbf{B}}$, $\underline{\mathbf{C}}$ are unique
- By imposing orthogonality constraints on each factor matrix, we arrive at the natural generalisation of the matrix SVD, the higher-order SVD (HOSVD)

Computation of the HOSVD



- 1 Compute the factor matrices first:

$$\begin{aligned}\underline{\mathbf{X}}_{(1)} &= \mathbf{A}\boldsymbol{\Sigma}_{(1)}(\mathbf{V}^{(1)})^T \\ \underline{\mathbf{X}}_{(2)} &= \mathbf{B}\boldsymbol{\Sigma}_{(2)}(\mathbf{V}^{(2)})^T \\ \underline{\mathbf{X}}_{(3)} &= \mathbf{C}\boldsymbol{\Sigma}_{(3)}(\mathbf{V}^{(3)})^T\end{aligned}\tag{1}$$

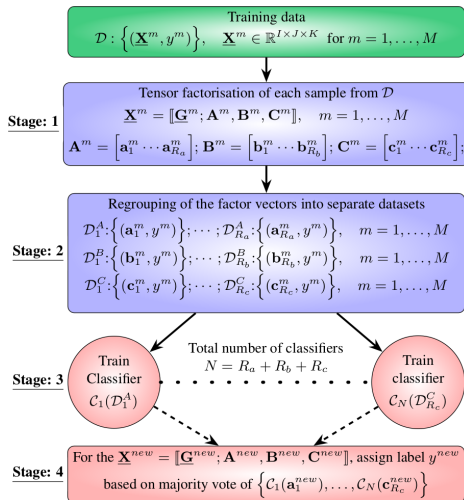
- 2 Compute the core tensor

$$\underline{\mathbf{G}} = \underline{\mathbf{X}} \times_1 \mathbf{A}^T \times_2 \mathbf{B}^T \times_3 \mathbf{C}^T\tag{2}$$

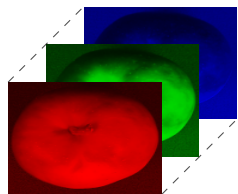
Where $\underline{\mathbf{G}} = \underline{\mathbf{X}} \times_1 \mathbf{A}^T \Leftrightarrow \mathbf{G}_{(1)} = \mathbf{A}^T \mathbf{X}_{(1)}$

Tensor ensemble learning (TEL): General concept

- 1 Apply tensor decomposition to each multidimensional sample to extract hidden information
- 2 Perform reorganisation of the obtained latent components
- 3 Use them to train an ensemble of base learners
- 4 For new sample, aggregate the knowledge about extracted latent components based on trained models in stage 3

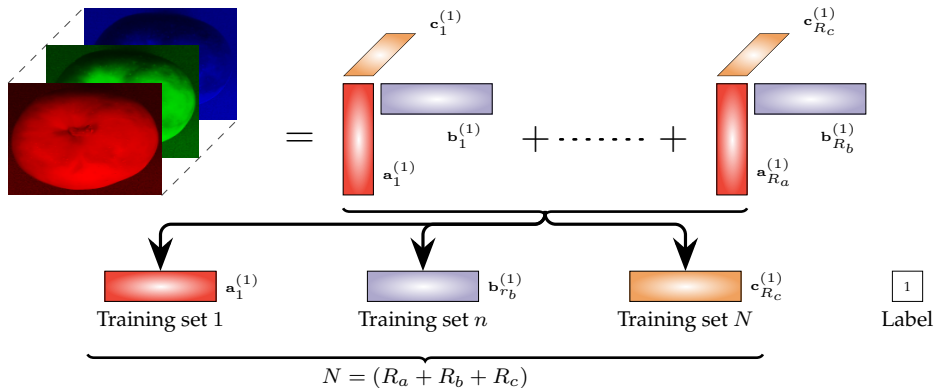


TEL: Formation of training set

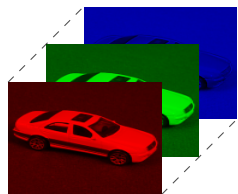


$$= \begin{array}{c} \text{orange trapezoid} \\ \text{red vertical bar} \\ \mathbf{a}_1^{(1)} \end{array} \begin{array}{c} \mathbf{c}_1^{(1)} \\ \text{grey horizontal bar} \\ \mathbf{b}_1^{(1)} \end{array} + \dots + \begin{array}{c} \text{orange trapezoid} \\ \text{red vertical bar} \\ \mathbf{a}_{R_a}^{(1)} \end{array} \begin{array}{c} \mathbf{c}_{R_c}^{(1)} \\ \text{grey horizontal bar} \\ \mathbf{b}_{R_b}^{(1)} \end{array}$$


TEL: Formation of training set





TEL: Formation of training set



$$= \begin{array}{c} \text{orange trapezoid} \\ \text{red vertical bar} \\ \text{purple horizontal bar} \\ \mathbf{a}_1^{(m)} \end{array} \begin{array}{c} \mathbf{c}_1^{(m)} \\ \mathbf{b}_1^{(m)} \end{array} + \dots + \begin{array}{c} \text{orange trapezoid} \\ \text{red vertical bar} \\ \text{purple horizontal bar} \\ \mathbf{a}_{R_a}^{(m)} \end{array} \begin{array}{c} \mathbf{c}_{R_c}^{(m)} \\ \mathbf{b}_{R_b}^{(m)} \end{array}$$

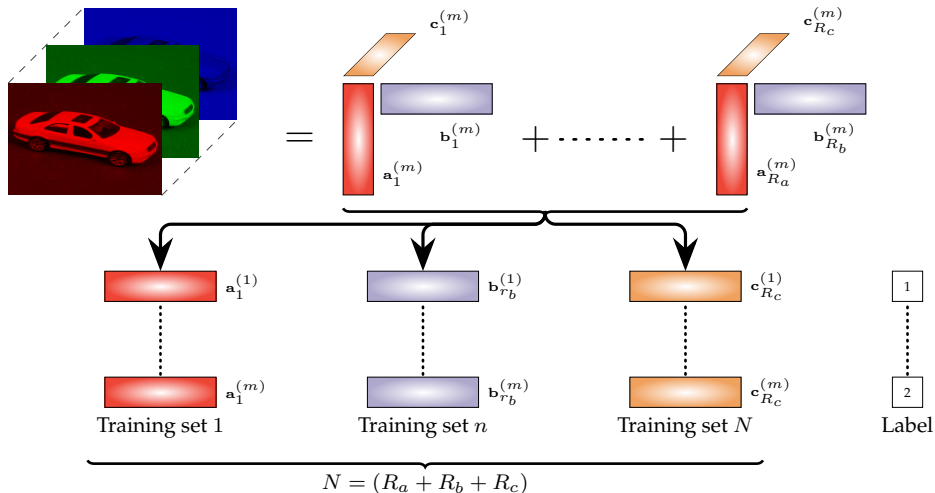
 $\mathbf{a}_1^{(1)}$
Training set 1

 $\mathbf{b}_{r_b}^{(1)}$
Training set n

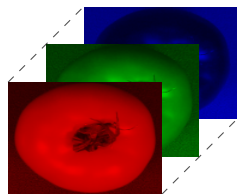
 $\mathbf{c}_{R_c}^{(1)}$
Training set N

 1
Label

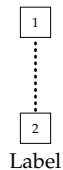
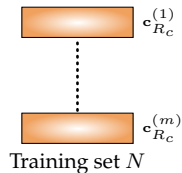
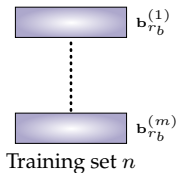
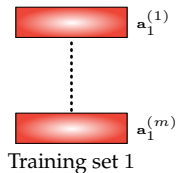
TEL: Formation of training set



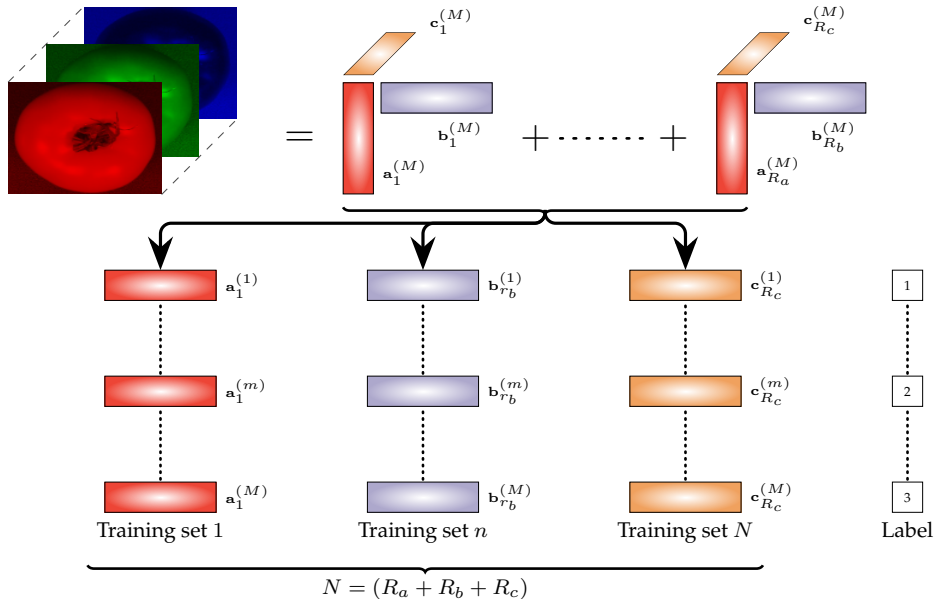
TEL: Formation of training set



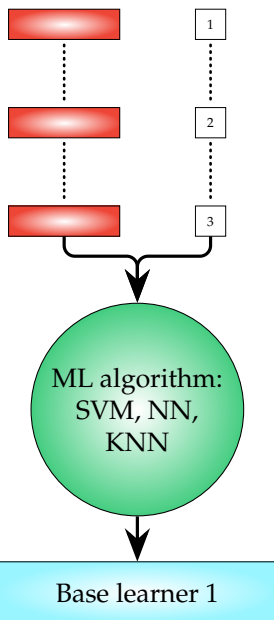
$$= \begin{array}{c} \text{orange trapezoid} \\ \mathbf{c}_1^{(M)} \\ \text{red vertical bar} \\ \mathbf{a}_1^{(M)} \end{array} \begin{array}{c} \text{purple rectangle} \\ \mathbf{b}_1^{(M)} \end{array} + \dots + \begin{array}{c} \text{orange trapezoid} \\ \mathbf{c}_{R_c}^{(M)} \\ \text{red vertical bar} \\ \mathbf{a}_{R_a}^{(M)} \end{array} \begin{array}{c} \text{purple rectangle} \\ \mathbf{b}_{R_b}^{(M)} \end{array}$$



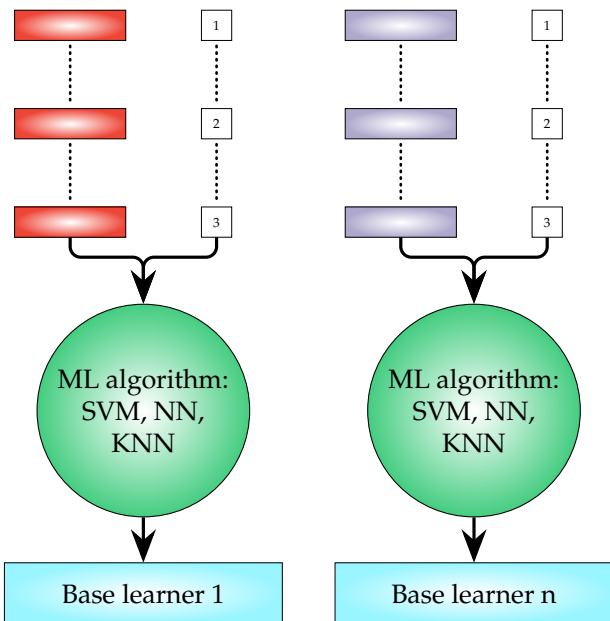
TEL: Formation of training set



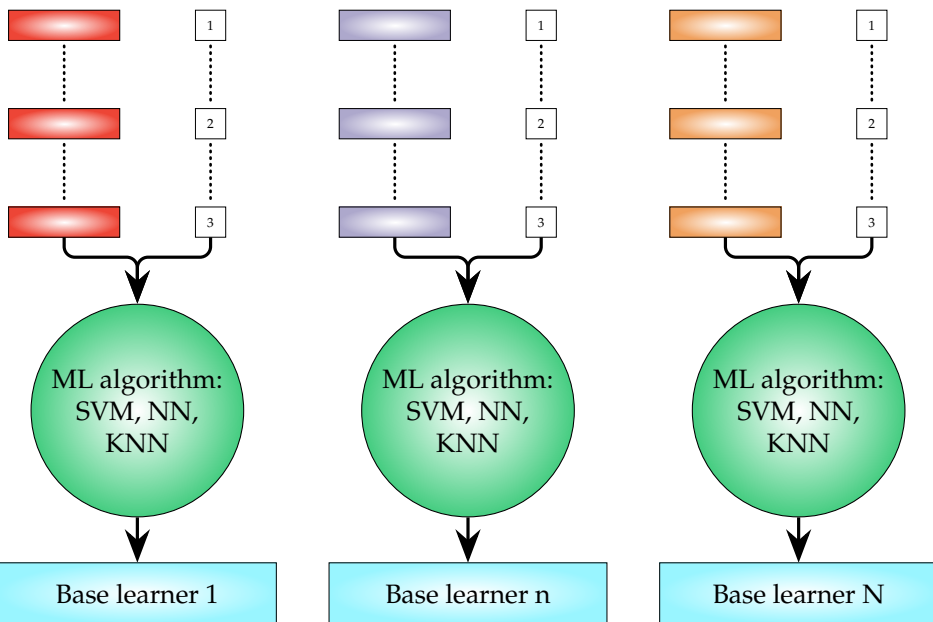
TEL: Training stage



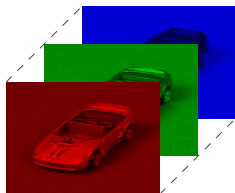
TEL: Training stage



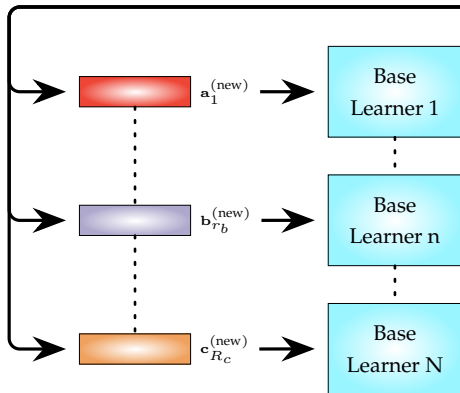
TEL: Training stage



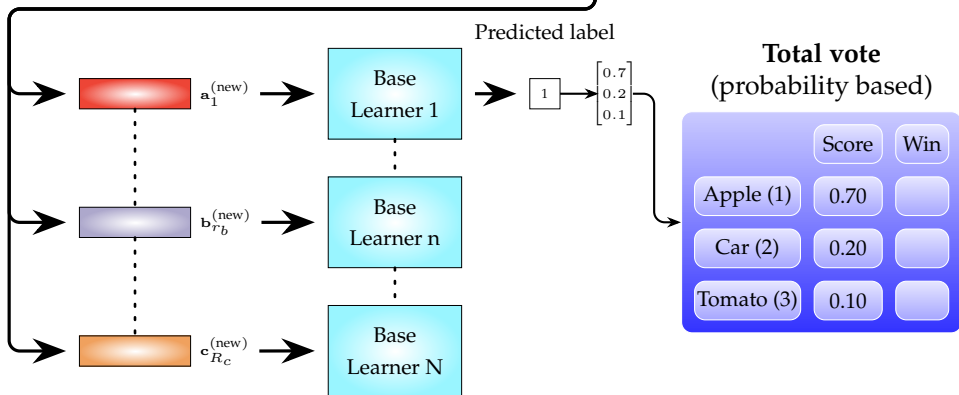
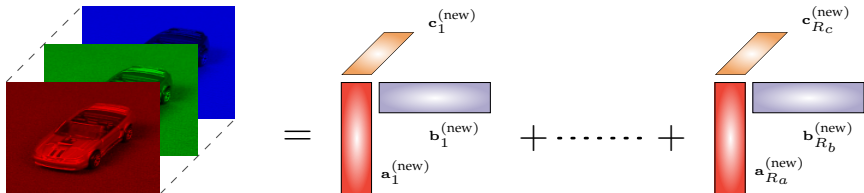
TEL: Testing stage



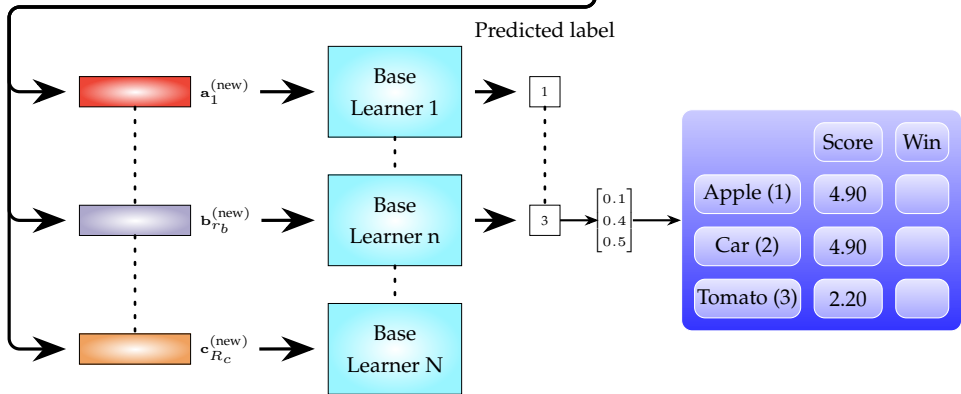
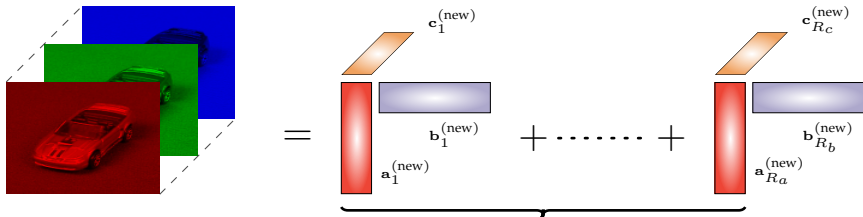
$$= \underbrace{\begin{matrix} \text{orange trapezoid} & c_1^{(new)} \\ \text{red vertical bar} & \text{purple horizontal bar} & b_1^{(new)} \\ \text{red vertical bar} & a_1^{(new)} \end{matrix}} + \dots + \begin{matrix} \text{orange trapezoid} & c_{R_c}^{(new)} \\ \text{red vertical bar} & \text{purple horizontal bar} & b_{R_b}^{(new)} \\ \text{red vertical bar} & a_{R_a}^{(new)} \end{matrix}$$



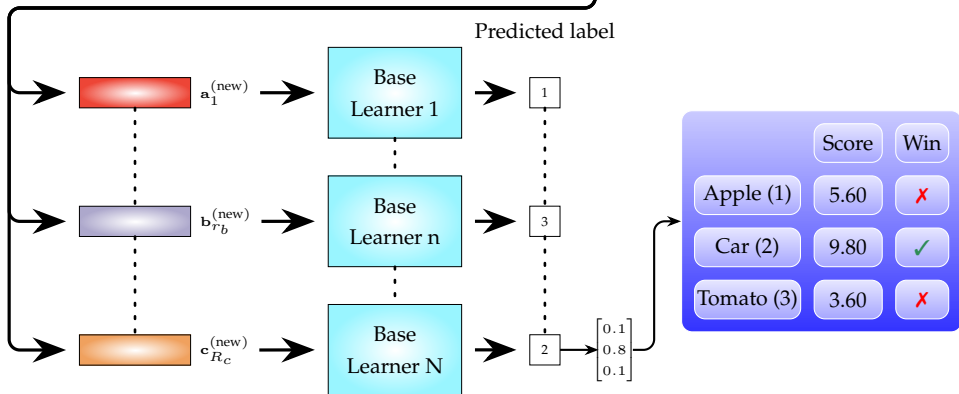
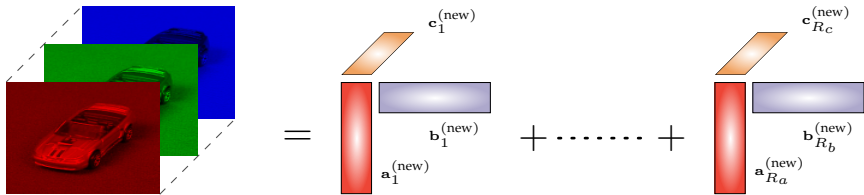
TEL: Testing stage



TEL: Testing stage



TEL: Testing stage

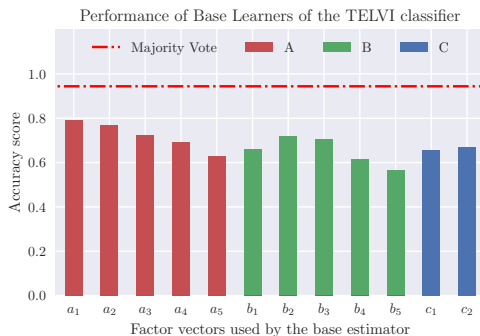


ETH-80 dataset



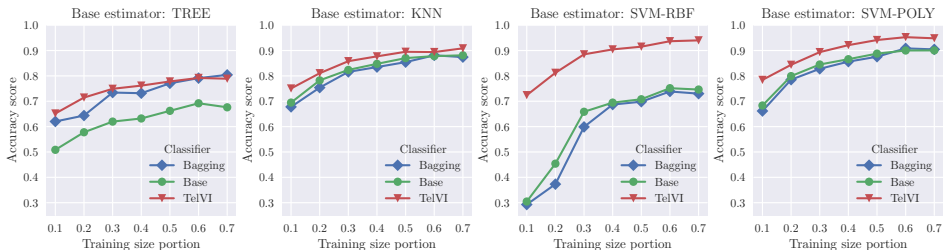
- 8 different categories
- 10 objects per category
- Each object is captured under 41 different viewpoints
- Total: 3280 samples

Performance of base estimators



- We employed HOSVD with multilinear rank $(5, 5, 2)$
- Utilised 12 base estimators (train/test split 50%)
- None of the base classifiers exhibited strong performance on the training set
- Combinational behaviour is similar to classic ensemble learning

Comparison of the overall test performance



- Random split into training and test data varied in range from 10% 70%
- Hyperparameter tuning: grid search with the 5-fold CV of the training data
- For fair comparison, the Bagging classifier also utilised 12 base learners

Conclusions: Key points to take home

- 1 TEL is a novel framework for generating ensembles of base estimators for multidimensional data
- 2 TEL highly parallelisable and suitable for large-scale problems
- 3 Enhanced performance is due to ability to obtain uncorrelated surrogate datasets that are generated by HOSVD

Conclusions: Key points to take home

- 1 TEL is a novel framework for generating ensembles of base estimators for multidimensional data
- 2 TEL highly parallelisable and suitable for large-scale problems
- 3 Enhanced performance is due to ability to obtain uncorrelated surrogate datasets that are generated by HOSVD

New Software: Higher Order Tensors ToolBOX (HOTTBOX)



Our python package for multilinear algebra:

`github.com/hottbox/hottbox`



Documentation: `hottbox.github.io`



Tutorials: `github.com/hottbox/hottbox-tutorials`

Thank you for your attention 

Questions?

Scalar



Vector



Matrix



Tensor

