Multi-Modal Ensembles of Regressor Chains for Multi-Output Prediction

Symposium on Intelligent Data Analysis 2022
April 20-22, 2022, Rennes, France

Ekaterina Antonenko and Jesse Read

April 21, 2022

Laboratoire d'informatique, École Polytechnique, IP Paris
DigitalentLab, MIA, Moteurs d'Intelligence Artificielle

Digitalent

Definition of a multi-output problem

Given: Dataset $D = \{(x^{(i)}, y^{(i)})\}_{i=1}^{N}$ of N samples:

- features $x^{(i)} = [x_1^{(i)}, \ldots, x_M^{(i)}]$
- outputs $y^{(i)} = [y_1^{(i)}, \ldots, y_K^{(i)}]$

Goal: Model $h(x) = y$ which outputs predictions $\hat{y}^{(i)} = [\hat{y}_1^{(i)}, \ldots, \hat{y}_K^{(i)}]$ having D observed.

Definition of a multi-output problem

Given: Dataset $D = f(\text{’}^{\text{i}}; \text{‘}^{\text{i}})g_{\text{i}=1}^{\acute{Y}}$ of $\acute{Y}$ samples:

- features $\text{’}^{\text{i}} = [\text{’}^{\text{i}}_{1}; :::; \text{’}^{\text{i}}_{\grave{U}}]$
- outputs $\text{‘}^{\text{i}} = [\text{‘}^{\text{i}}_{1}; :::; \text{‘}^{\text{i}}_{\acute{I}}]$

Goal: Model $\grave{\imath}(\text{ŧ}) = \text{‘}$ which outputs predictions $\text{ʔ}^{\text{i}} = [\text{ʔ}^{\text{i}}_{1}; :::; \text{ʔ}^{\text{i}}_{\acute{I}}]$ having D observed.

Example:

	x	Age	Height	Body length	Weight
A		12,44	127,4	151	294,5
B		9,44	137,6	156	328
C		10,44	128,6	157	377
D		6,13	125,6	150	305,5
E		6,15	139	156	325
F		?	?	?	?

Definition of a multi-output problem

Given: Dataset $D = f(\text{’}^{(i)}; \text{‘}^{(i)})g_{i=1}^{N}$ of N samples:

- features $\text{’}^{(i)} = [\text{’}_1^{(i)}; \ldots; \text{’}_M^{(i)}]$
- outputs $\text{‘}^{(i)} = [\text{‘}_1^{(i)}; \ldots; \text{‘}_K^{(i)}]$

Goal: Model $h(x) = \text{‘}$ which outputs predictions $\hat{\text{‘}}^{(i)} = [\hat{\text{‘}}_1^{(i)}; \ldots; \hat{\text{‘}}_K^{(i)}]$ having D observed.

Example:

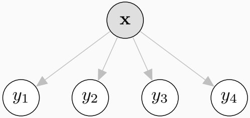
	x	Age	Height	Body length	Weight
A		12,44	127,4	151	294,5
B		9,44	137,6	156	328
C		10,44	128,6	157	377
D		6,13	125,6	150	305,5
E		6,15	139	156	325
F		?	?	?	?

Idea: to model these labels together in order to get better prediction performance

Some approaches to multi-output problems

Independent models (= [illegible] for classification):

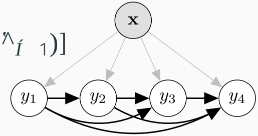
$$\hat{y} = [\hat{y}_1, \ldots, \hat{y}_m] = [h_1(x), \ldots, h_m(x)]$$



Fully-cascaded chain[1]:

$$\hat{y} = [\hat{y}_1, \ldots, \hat{y}_m] = [h_1(x), h_2(x, \hat{y}_1), \ldots, h_m(x, \hat{y}_1, \ldots, \hat{y}_{m-1})]$$

$h_1, h_2, \ldots, h_m$ = [illegible] (i.e. any single-output models)

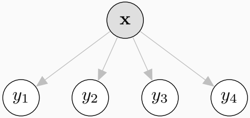


[1] [illegible], Read et al., 2011, [illegible], Spyromitros-Xioufis et al., 2016.

Some approaches to multi-output problems

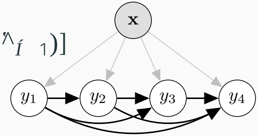
Independent models (= [illegible] for classification):

$$\hat{y} = [\hat{y}_1, \ldots, \hat{y}_m] = [h_1(x), \ldots, h_m(x)]$$



Fully-cascaded chain[1]:

$$\hat{y} = [\hat{y}_1, \ldots, \hat{y}_m] = [h_1(x), h_2(x, \hat{y}_1), \ldots, h_m(x, \hat{y}_1, \ldots, \hat{y}_{m-1})]$$



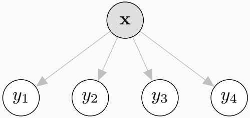
$h_1, h_2, \ldots, h_m$ = [illegible] (i.e. any single-output models)

	x	Age	Height	Body length	Weight
A		?	?	?	?
B		?	?	?	?
C		?	?	?	?
D		?	?	?	?
E		?	?	?	?

[1] [illegible], Read et al., 2011, [illegible], Spyromitros-Xioufis et al., 2016.

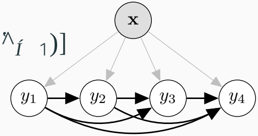
Some approaches to multi-output problems

Independent models (= *binary relevance* for classification):



$$\hat{\mathbf{y}} = [\hat{y}_1, \ldots, \hat{y}_m] = [h_1(\mathbf{x}), \ldots, h_m(\mathbf{x})]$$

Fully-cascaded chain[1]:



$$\hat{\mathbf{y}} = [\hat{y}_1, \ldots, \hat{y}_m] = [h_1(\mathbf{x}), h_2(\mathbf{x}, \hat{y}_1), \ldots, h_m(\mathbf{x}, \hat{y}_1, \ldots, \hat{y}_{m-1})]$$

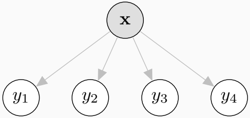
$h_1, h_2, \ldots, h_m$ = *base estimators* (i.e. any single-output models)

	X	Age	Height	Body length	Weight
A		12,44	?	?	?
B		9,44	?	?	?
C		10,44	?	?	?
D		6,13	?	?	?
E		6,15	?	?	?

[1] *Classifier chains for multi-label classification*, Read et al., 2011, *Multi-Target Classification Methods for Multi-Target Regression*, Spyromitros-Xioufis et al., 2016.

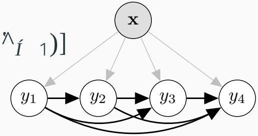
Some approaches to multi-output problems

Independent models (= [illegible] for classification):



$$\hat{\mathbf{y}} = [\hat{y}_1, \ldots, \hat{y}_m] = [h_1(\mathbf{x}), \ldots, h_m(\mathbf{x})]$$

Fully-cascaded chain[1]:



$$\hat{\mathbf{y}} = [\hat{y}_1, \ldots, \hat{y}_m] = [h_1(\mathbf{x}), h_2(\mathbf{x}, \hat{y}_1), \ldots, h_m(\mathbf{x}, \hat{y}_1, \ldots, \hat{y}_{m-1})]$$

$h_1, h_2, \ldots, h_m$ = [illegible] (i.e. any single-output models)

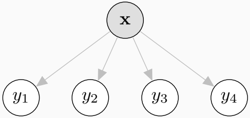
	x	Age	Height	Body length	Weight
A		12,44	127,4	?	?
B		9,44	137,6	?	?
C		10,44	128,6	?	?
D		6,13	125,6	?	?
E		6,15	139	?	?

[1] [illegible], Read et al., 2011, [illegible], Spyromitros-Xioufis et al., 2016.

Some approaches to multi-output problems

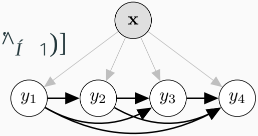
Independent models (= binary relevance for classification):

$$\hat{\mathbf{y}} = [\hat{y}_1, \ldots, \hat{y}_m] = [f_1(\mathbf{x}), \ldots, f_m(\mathbf{x})]$$



Fully-cascaded chain[1]:

$$\hat{\mathbf{y}} = [\hat{y}_1, \ldots, \hat{y}_m] = [f_1(\mathbf{x}), f_2(\mathbf{x}, \hat{y}_1), \ldots, f_m(\mathbf{x}, \hat{y}_1, \ldots, \hat{y}_{m-1})]$$



$f_1, f_2, \ldots, f_m$ = *base learners* (i.e. any single-output models)

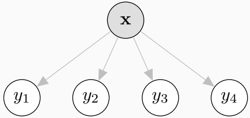
	X	Age	Height	Body length	Weight
A		12,44	127,4	151	?
B		9,44	137,6	156	?
C		10,44	128,6	157	?
D		6,13	125,6	150	?
E		6,15	139	156	?

[1] *Classifier chains for multi-label classification*, Read et al., 2011, *Multi-target regression via input space expansion: treating targets as inputs*, Spyromitros-Xioufis et al., 2016.

Some approaches to multi-output problems

Independent models (= binary relevance for classification):

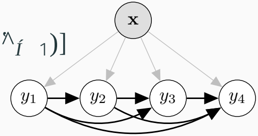
$$\hat{\mathbf{y}} = [\hat{y}_1, \ldots, \hat{y}_m] = [h_1(\mathbf{x}), \ldots, h_m(\mathbf{x})]$$



Fully-cascaded chain[1]:

$$\hat{\mathbf{y}} = [\hat{y}_1, \ldots, \hat{y}_m] = [h_1(\mathbf{x}), h_2(\mathbf{x}, \hat{y}_1), \ldots, h_m(\mathbf{x}, \hat{y}_1, \ldots, \hat{y}_{m-1})]$$

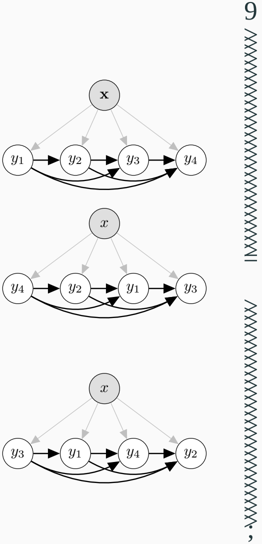
$h_1, h_2, \ldots, h_m$ = base learners (i.e. any single-output models)



	x	Age	Height	Body length	Weight
A		12,44	127,4	151	294,5
B		9,44	137,6	156	328
C		10,44	128,6	157	377
D		6,13	125,6	150	305,5
E		6,15	139	156	325

[1] *Classifier chains for multi-label classification*, Read et al., 2011, *Multi-target regression via input space expansion: treating targets as inputs*, Spyromitros-Xioufis et al., 2016.

Ensembles of chains[2]



Mean average

[2] Ù˵ȳ¯̊0ˉ7Íhʋɛā̄ȳ Gȳhʋ’’0̊´ !hʋ¯0̊ωɧ Ùā̄¯ȡωă’ ȉω𝅶 Ù˵ȳ¯̊0ˉ7İhʋ𝅶Ôā̄¯ Ĝā̄Ô𝅶ā̄’’0̊ωɧ, Spyromitros-Xioufis et al., 2016.

Does the chaining approach work?

Classification

Classifier Chains have proved to be **flexible and effective** and have achieved **state-of-the-art** empirical performance

- *[illegible]*, Read et al., 2021

Regression

Regressor Chains show **relatively few advantages** compared to individual regression models. State-of-the-art methods:

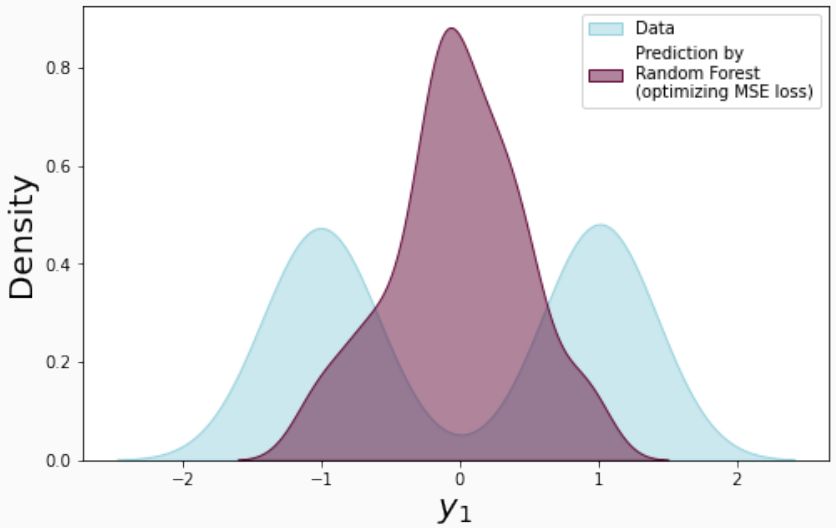
- Multi-output Decision Trees (DT)
- Multi-output Random Forests (RF)
- Independent Regressors (IR)

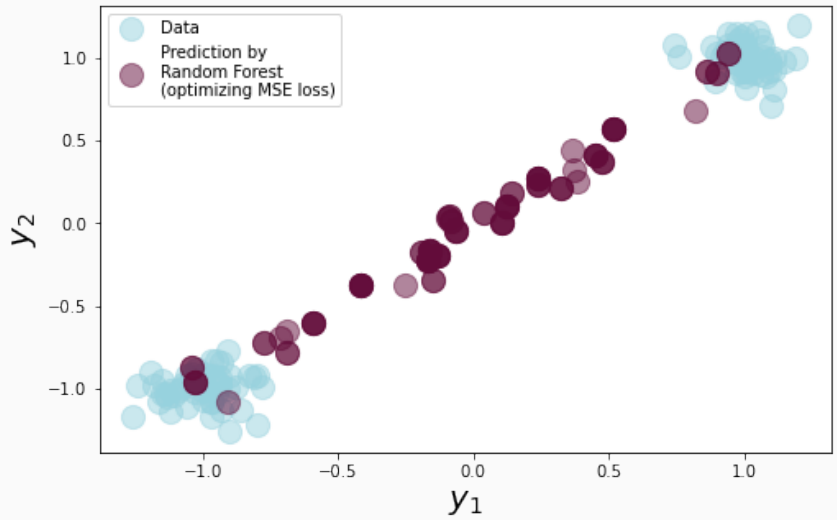
Regressor chains: why don't they work?

One possible reason: inadequate choice of the loss function

Most models optimize MSE $= \frac{1}{\tilde{Y}} \sum_{\text{?}=1}^{\acute{Y}} (\text{?} - \text{?})^2$:

Example: Multi-modal distribution $=)$ standard models may be inappropriate.





Optimizing MSE does not help to exploit the dependencies between the targets.

Multi-Modal Ensembles of Regressor Chains (mmERC)

Uniform Cost Function (UCF)[3] is an analogue of 0/1 loss for regression.

$$\mathrm{UCF}(\ ) = \frac{1}{\acute{Y}} \sum_{\ddot{\imath}=1}^{\acute{Y}} \begin{cases} 0 & \text{if } k_{\ddot{\imath}} - \hat{k}_{\ddot{\imath}} k_2 < \frac{1}{2}; \\ 1 & \text{otherwise} \end{cases}$$

Goal = challenge: optimize UCF.

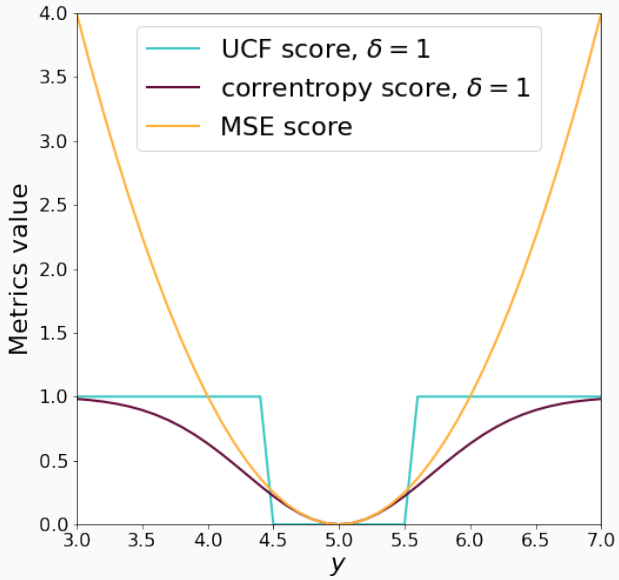
[3] *Ùhu' ï g ˇ g hu £ω'¯ā₩ï ω₩ï ā'¯ï g hu¯ā' ïƒ ƿïƒāhu₩ ïƒ· ā₩'ā £₩ωƐƿā g ' ˝ ï¯ɖ ƿωÔ7!ωƒ!hu· ā £₩ï ω₩' hu₩ā £₩ω£ā₩ Ɛhu· ā' ā'¯ï g hu¯ω₩'*, Burger and Lucka, 2014.

Multi-Modal Ensembles of Regressor Chains (mmERC)

Uniform Cost Function (UCF)[3] is an analogue of 0/1 loss for regression.

$$\mathrm{UCF}(\quad) = \frac{1}{\acute{Y}} \sum_{\mathring{0}=1}^{\acute{\mathrm{X}}} \left(0 \_ \mathrm{T}\ k \text{‘}^{\mathring{0}} \quad \text{‘}^{\mathring{0}} k_2 < \frac{}{2}; \right.$$
$$\left. 1\ \mathrm{n}\S \backslash 0\} \circ \_\ 0\mathrm{u} \right)$$

Goal = challenge: optimize UCF.



[3] *Ùhu’ ⁰g ˗ g hu £ω’¯ā₩⁰ω₩⁰ ā’¯⁰g hu¯ā’ ⁰ɦ ¥⁰ɦāhu₩ ⁰ɦ˙ā₩’ā £₩ωƐ¥āg ’ ˝ ⁰¯ɖ ¥ωÔ7!ωɦ!hu˙ā £₩⁰ω₩’ hu₩ā £₩ω£ā₩ Ɛhu˙ ā’ ā’¯⁰g hu¯ω₩’*, Burger and Lucka, 2014.

Multi-Modal Ensembles of Regressor Chains (mmERC)

Base Estimator level

ERC

- Single round of training

mmERC

- Train on all dataset
- Choose portion of data giving best predictions
- Retrain on this part

Algorithm 1 mmERC: Training h_j for target y_j (done for $j = 1, \ldots, L$)

procedure FIT(h_j, $\{\boldsymbol{x}, y_j\}$) ▷ Train b.e. h_j for target y_j on $\{\boldsymbol{x}, y_j\}$
  $\widetilde{h_j} \leftarrow$ clone of h_j
  fit $\widetilde{h_j}$ on $\{(\boldsymbol{x}, y_j)\}$ ▷ First training phase (full training set)
  $y_{pred} \leftarrow \widetilde{h_j}(\boldsymbol{x})$ ▷ Prediction of $\widetilde{h_j}$ on $\boldsymbol{x}$
  $corr \leftarrow 1 - e^{-(y_j - y_{pred})^2}$ ▷ Correntropy
  $\{\boldsymbol{x}', y'_j\} \subset \{\boldsymbol{x}, y_j\}$ ▷ Top s-instances wrt (lowest) $corr$, $0 < s < 1$
  fit h_j on $\{(\boldsymbol{x}', y'_j)\}$ ▷ Second training phase
  return h_j ▷ Return the trained model

Multi-Modal Ensembles of Regressor Chains (mmERC)

Base Estimator level

ERC

- Single round of training

mmERC

- Train on all dataset
- Choose portion of data giving best predictions
- Retrain on this part

Algorithm 1 mmERC: Training h_j for target y_j (done for $j = 1, \ldots, L$)

procedure FIT(h_j, $\{\boldsymbol{x}, y_j\}$) ▷ Train b.e. h_j for target y_j on $\{\boldsymbol{x}, y_j\}$
  $\widetilde{h_j} \leftarrow$ clone of h_j
  fit $\widetilde{h_j}$ on $\{(\boldsymbol{x}, y_j)\}$ ▷ First training phase (full training set)
  $y_{pred} \leftarrow \widetilde{h_j}(\boldsymbol{x})$ ▷ Prediction of $\widetilde{h_j}$ on $\boldsymbol{x}$
  $corr \leftarrow 1 - e^{-(y_j - y_{pred})^2}$ ▷ Correntropy
  $\{\boldsymbol{x}', y_j'\} \subset \{\boldsymbol{x}, y_j\}$ ▷ Top s-instances wrt (lowest) $corr$, $0 < s < 1$
  fit h_j on $\{(\boldsymbol{x}', y_j')\}$ ▷ Second training phase
  return h_j ▷ Return the trained model

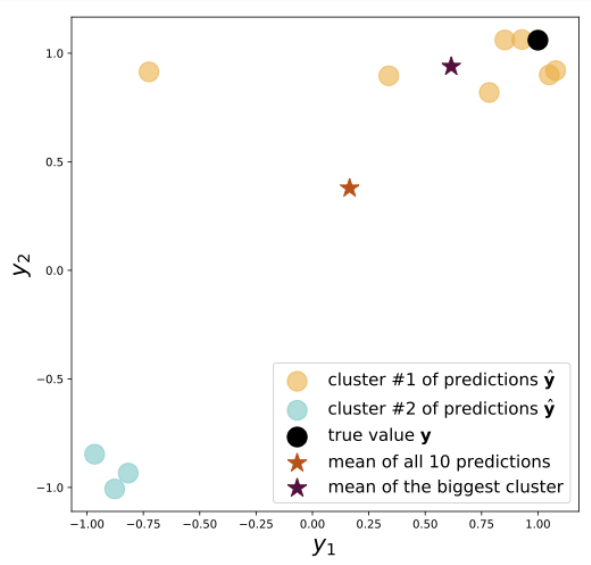
Ensemble level

ERC

- Mean for all predictions

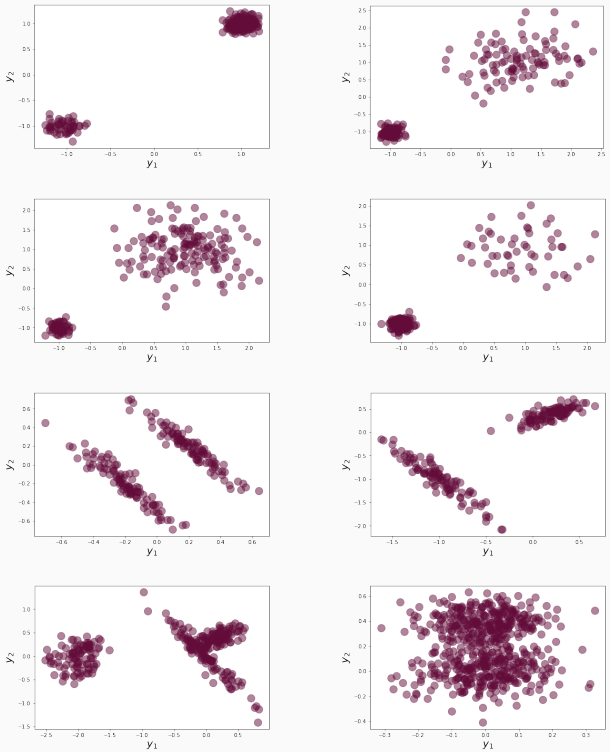
mmERC

- Choose the largest cluster of predictions
- Take mean for this cluster only



mmERC: results on 40 synthetic datasets

Datasets (’ ; ‘ 1; ‘ 2):

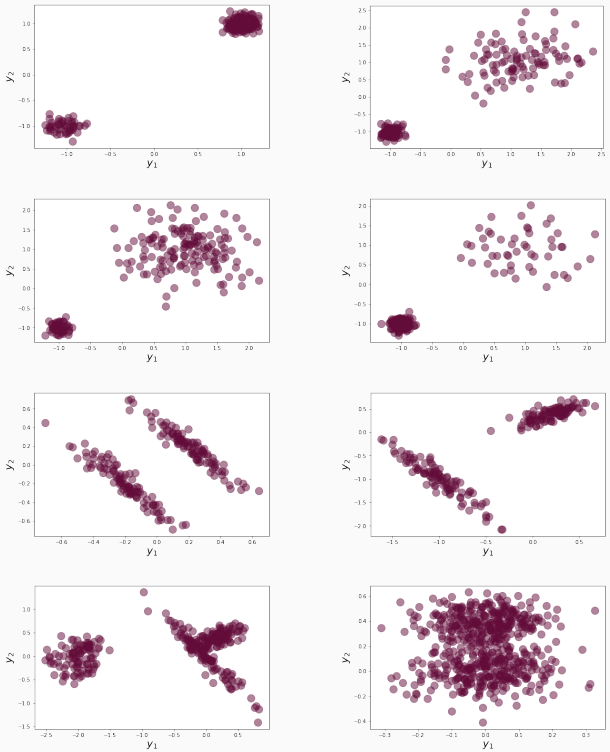


For each ‘ distribution, there are five ’ distributions

A: $\mathcal{L}(0; 1)$ where $\mathcal{L}$ stands for uniform distribution
B: $f\,0; 1g$ (according to the cluster)
C: $f \quad \mathcal{L}(0; 1); \quad \mathcal{L}(1; 2)\,g$ (according to the cluster)
D: $f \quad N\ (0; 1); \quad N\ (1; 1)\,g$ (according to the cluster)
E: $N\ (0; 1)$

mmERC: results on 40 synthetic datasets

Datasets (’ ; ‘ 1; ‘ 2):

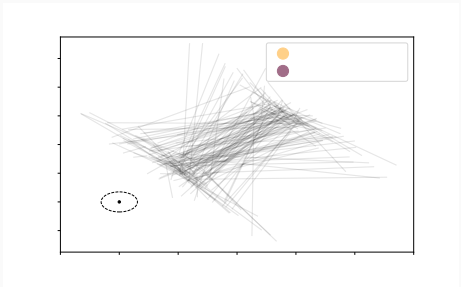


For each ‘ distribution, there are five ’ distributions

A:	$L(0; 1)$ where L stands for uniform distribution
B:	$f\,0; 1g$ (according to the cluster)
C:	$f \quad L(0; 1); \quad L(1; 2)g$ (according to the cluster)
D:	$f \quad N\,(0; 1); \quad N\,(1; 1)g$ (according to the cluster)
E:	$N\,(0; 1)$

Results (UCF):

(grouped by ’ distribution)



Random Forest

mmERC (RF)

mmERC: results on ˋ ℎ!"ℎ dataset

$\tilde{U}$ℎ!"ℎ dataset:

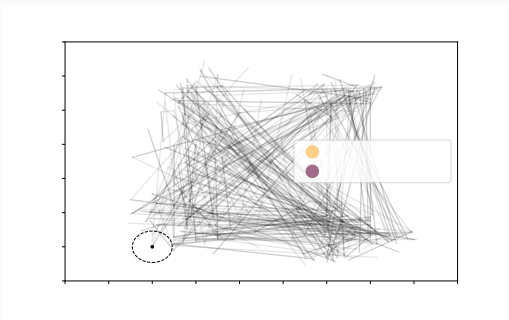
mmERC: results on ˋ ℎ!"ℎ dataset

$\tilde{U}$ℎ!"ℎ dataset:

mmERC: results on ˋ $h\omega\!f\!\!f$ dataset

$\tilde{U}h\omega\!f\!\!f$ dataset:

Results:



mmERC (rf) DT mmERC (rf) vs. DT

Discussion

- **mmERC** on average **outperforms the independent regressors** and standard Regressor Chains
- **mmERC improves ERC** for all base estimators in most of the scenarios
- Decision Trees: recognize cluster shape, but may assign clusters randomly, not smooth
  Random Forests: "smooth", but put the predictions between the real clusters
  mmERC: improves the performance of Random Forests and outputs a **smooth function** at the same time

Thank you!