

# Random Forest Proximities and Their Applications in Data Science



J A K E   R H O D E S

B R I G H A M   Y O U N G  
U N I V E R S I T Y



# Outline

- Classification and Regression Trees
  - What are they?
  - How to optimize?
  - Benefits and Limitations
- Random Forests
  - Randomizations
  - Out-of-Bag Points
  - Predictions
  - Proximities



# Decision Making

---

Suppose you would like to buy a car. What are some aspects you need to consider? What are the most important variables for your decision?



# CART

## Classification and Regression Trees

- Predictive models
- Sequences of binary questions (yes/no)
- Make predictions by following paths and voting/aggregating responses.



# CART Terminology

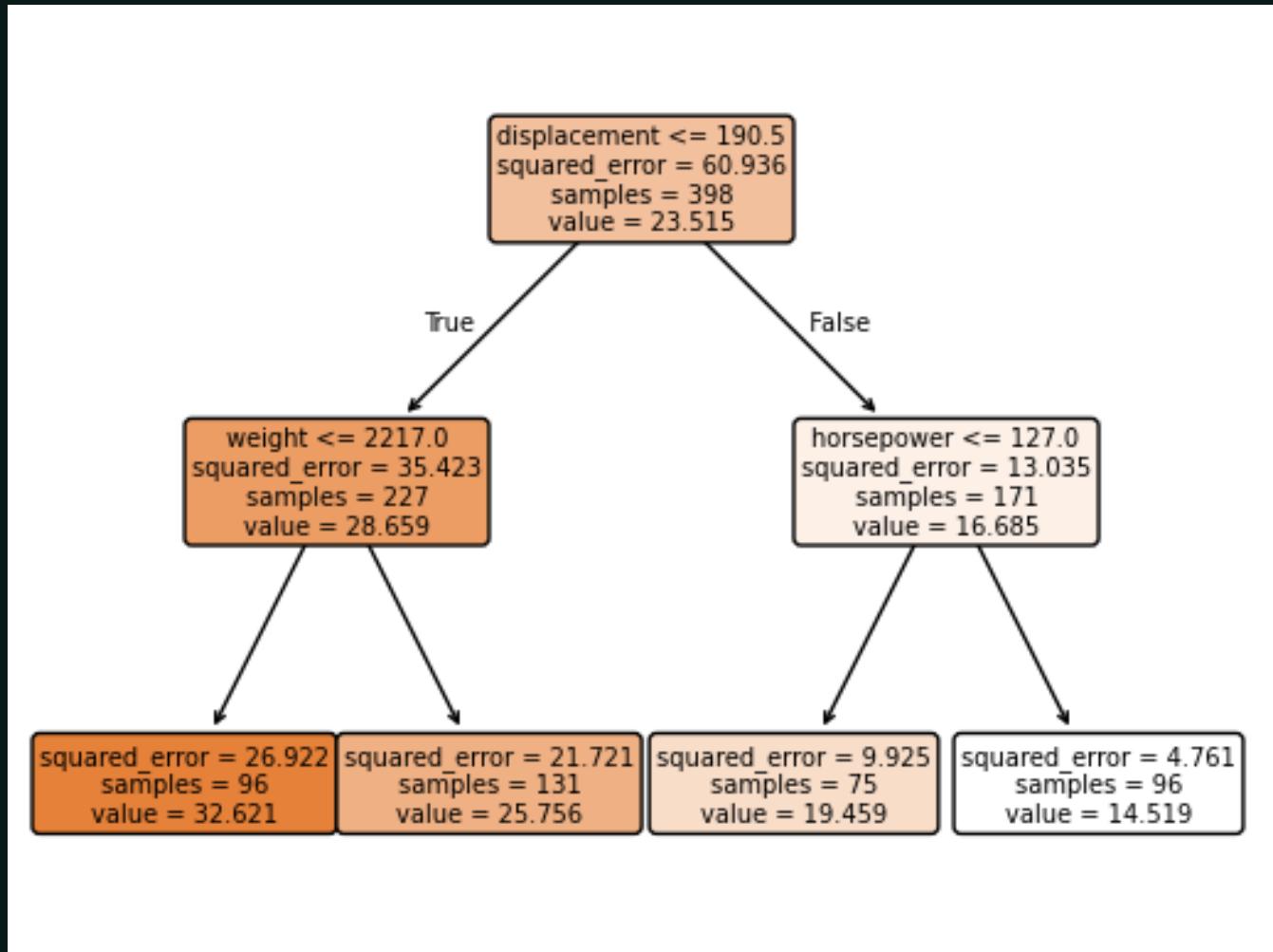
Nodes:

- Root (Beginning)
- Terminal or Leaf (End)
- Daughter / (Internal)

Splits:

- Splitting Variables (What)
- Split Points (Where)
- Splitting Criteria (Why / How)

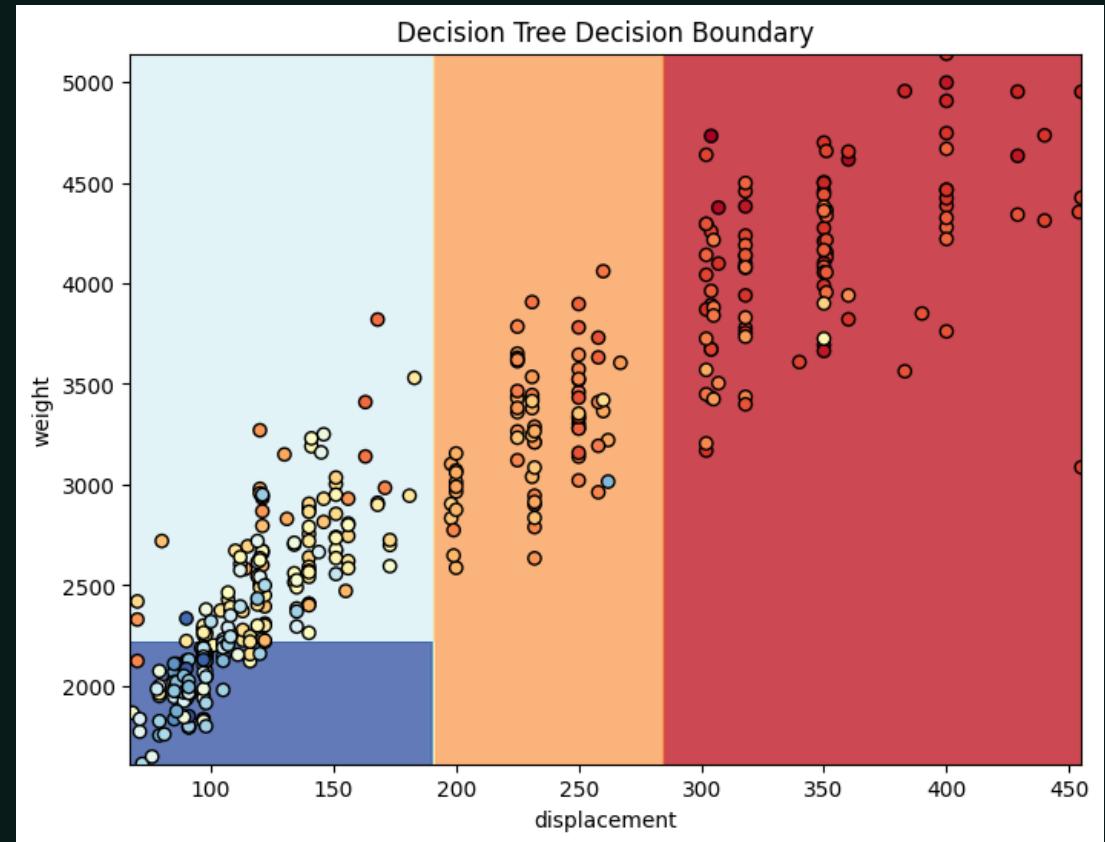
- Decision Tree: Estimate MGP



# CART Decision Boundaries

---

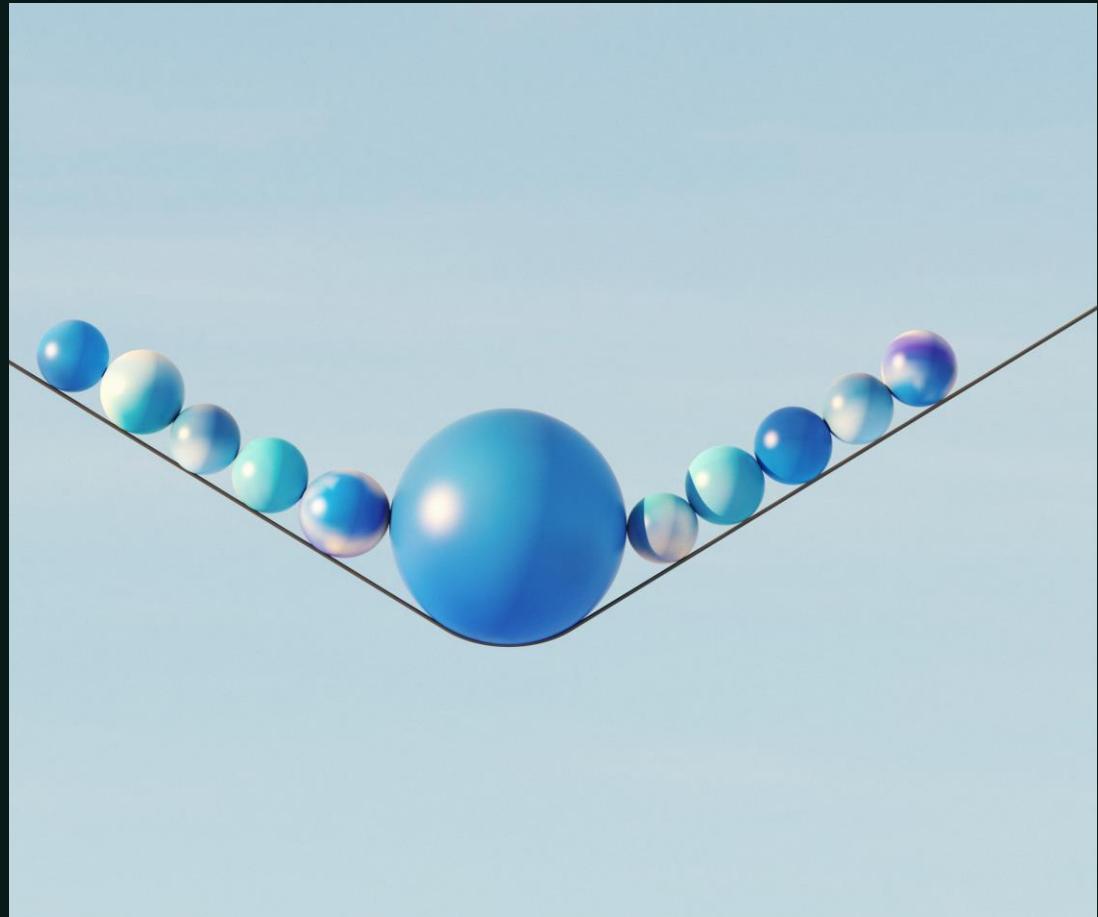
- Splits form orthogonal decision regions.
- Splits spanning the space come first.
- Early variables → important



# Mechanics of CART

---

- How to determine which variables?
- Which questions to ask first?
- How to determine split points?
- What process optimizes the decision-making?



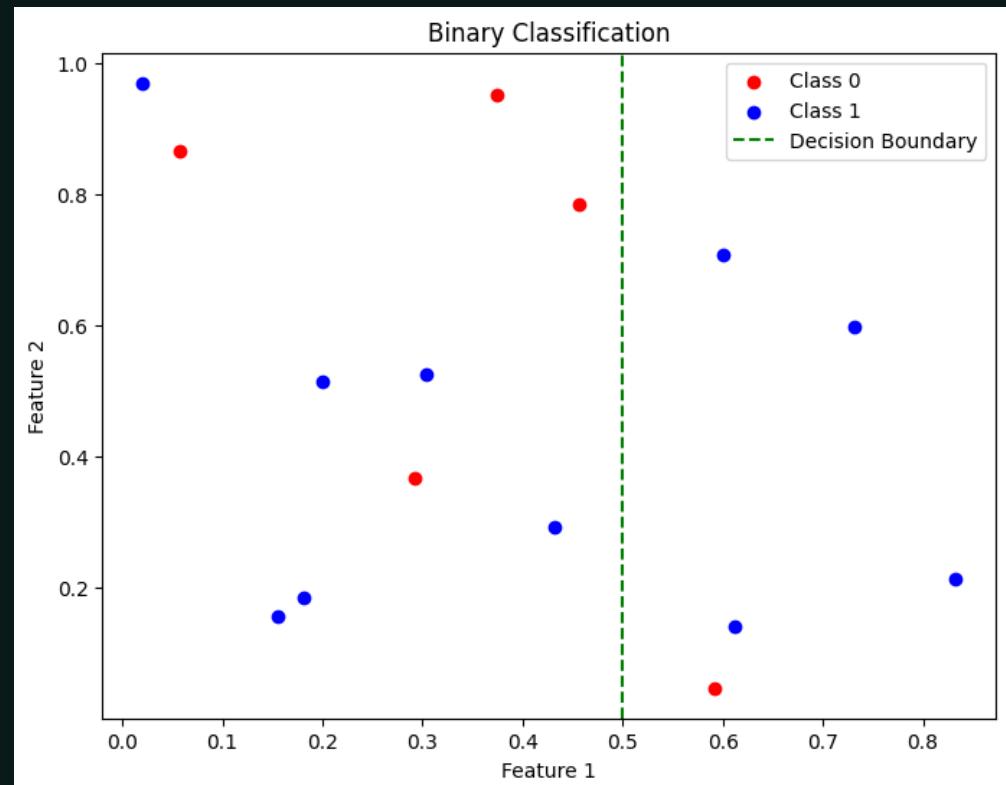
# CART: Optimizing Splits (Classification)

## G I N I   I M P U R I T Y

- Quantifies the probability of misclassifying a randomly chosen data point if it's labeled according to the class distribution.
- A lower Gini Impurity (closer to 0) indicates a purer node (e.g., all one class)

$$G = 1 - \sum_{i=1}^C p_i^2$$

## I M P U R I T Y   E X A M P L E



# CART: Optimization Steps

$$G = 1 - \sum_{i=1}^C p_i^2$$

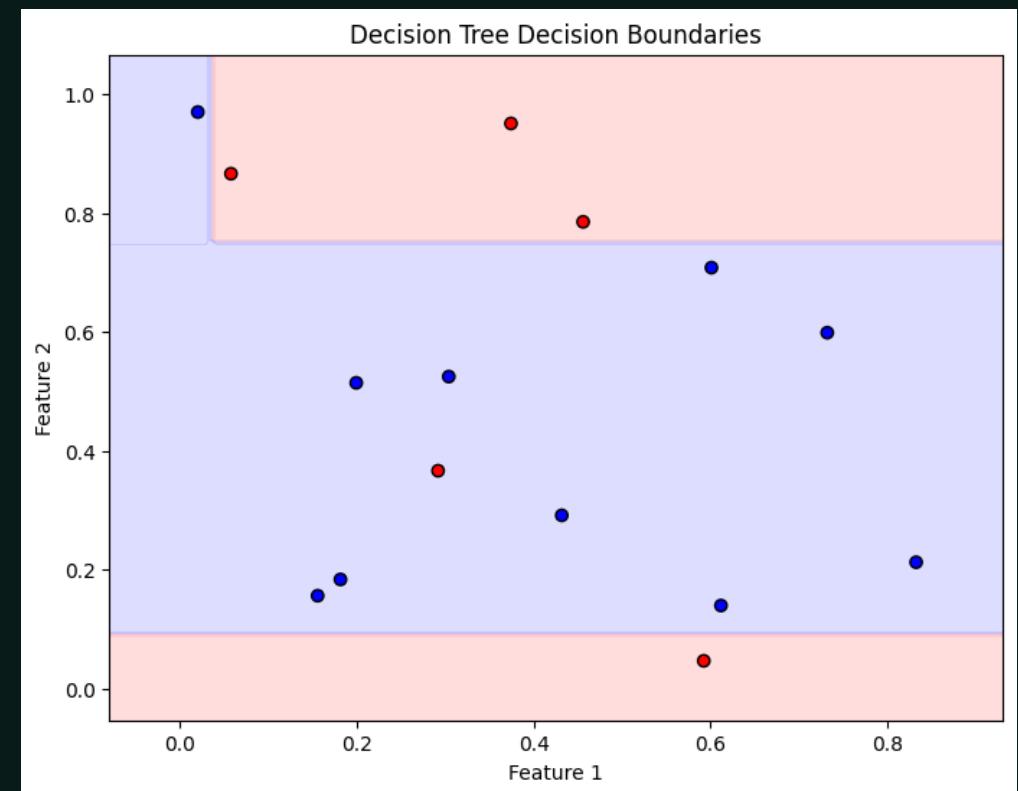
## GINI OPTIMIZATION

1. Evaluate all possible splits
  - a) Median between continuous values
  - b) Subsets/groupings of categorical
2. Partition the data by split
3. Compute Gini impurity by branch
4. Calculate weighted impurity

$$G_{\text{split}} = \frac{n_L}{n} \cdot G_L + \frac{n_R}{n} \cdot G_R$$

5. Choose split by min. impurity

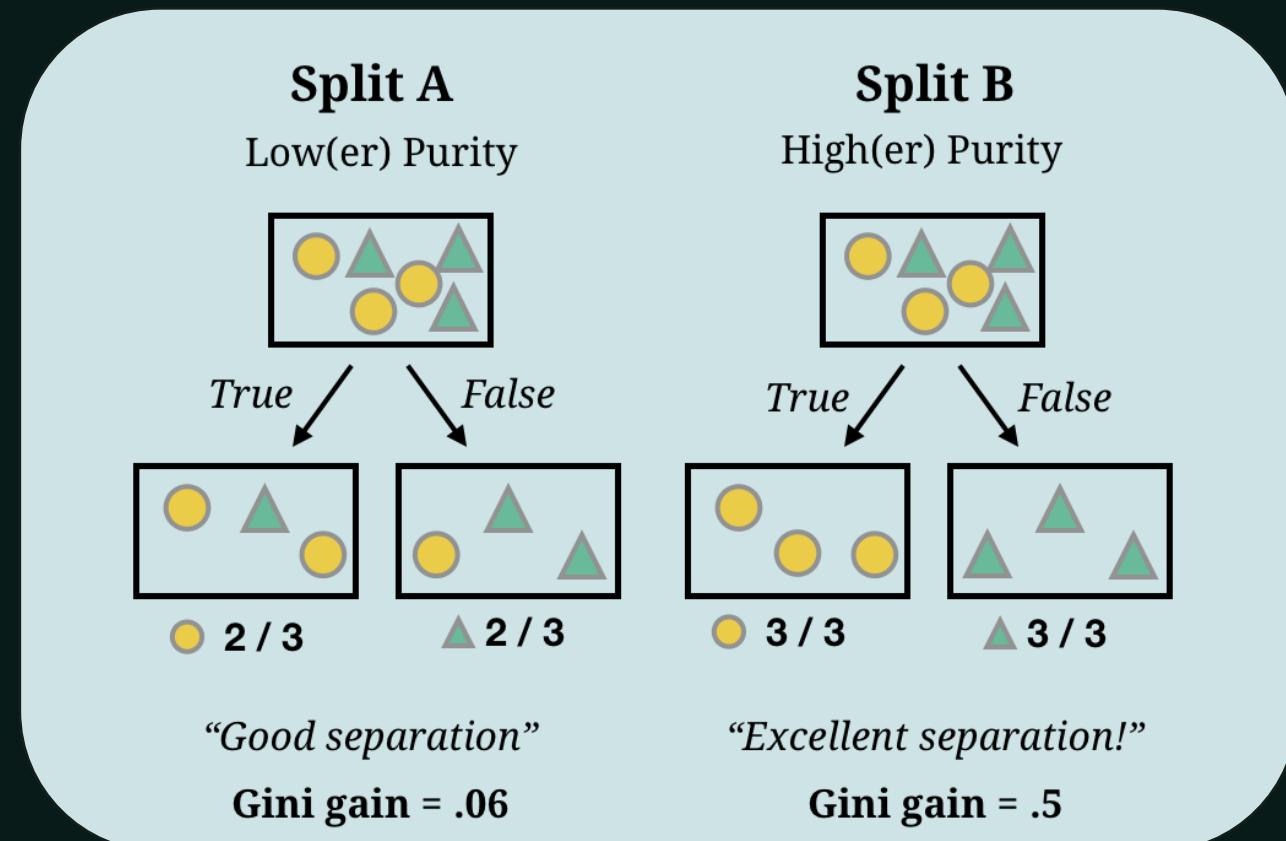
## BINARY CLASSIFICATION



# Other Considerations

## PARAMETERS ET AL

- When to stop splitting?
- Depth of the tree?
- Control for complexity.
- Data encoding.
- Data preparation.
- Feature selection.



# CART: Benefits and Limitations

---

## B E N E F I T S

- Both Classification and Regression
- Scale Invariant (rank-based)
- Learns Non-Linear Relationships
- Interpretable
- Mixed Variable Types

## L I M I T A T I O N S

- Low Performing
- High Variance
- Easily Overfit



# Random Forests

Overcoming CART's  
Limitations



But first.. a little game!

# Ensemble Methods

---

- Combine predictions from multiple models to produce more accurate / more stable results.
- Typically, a collection of “weak” learners





# Random Forest

- An ensemble of randomized CART decision trees.
- Combining trees beneficial if uncorrelated.
- Two-part randomization.



# Recall: CART Limitations

---

## C A R T

- High variance
- Easy to overfit.



## R A N D O M   F O R E S T S

- Decreased variance (ensembling)
- Less sensitive to outliers



# Random Forest Randomizations

---

## BOOTSTRAPPING

- For each tree, data is sampled with replacement
- Sampled data: In-bag
- Otherwise: Out-of-bag
- P(In-Bag) -->

## RANDOM FEATURE SELECTION

- At each split,  $m$  variables considered
- For classification:  $m = \sqrt{d}$
- For regression:  $m = (1/3)d$

$$1 - (1 - \frac{1}{n})^n \rightarrow 1 - \frac{1}{e} \approx 0.632$$

# Out-of-Bag Points

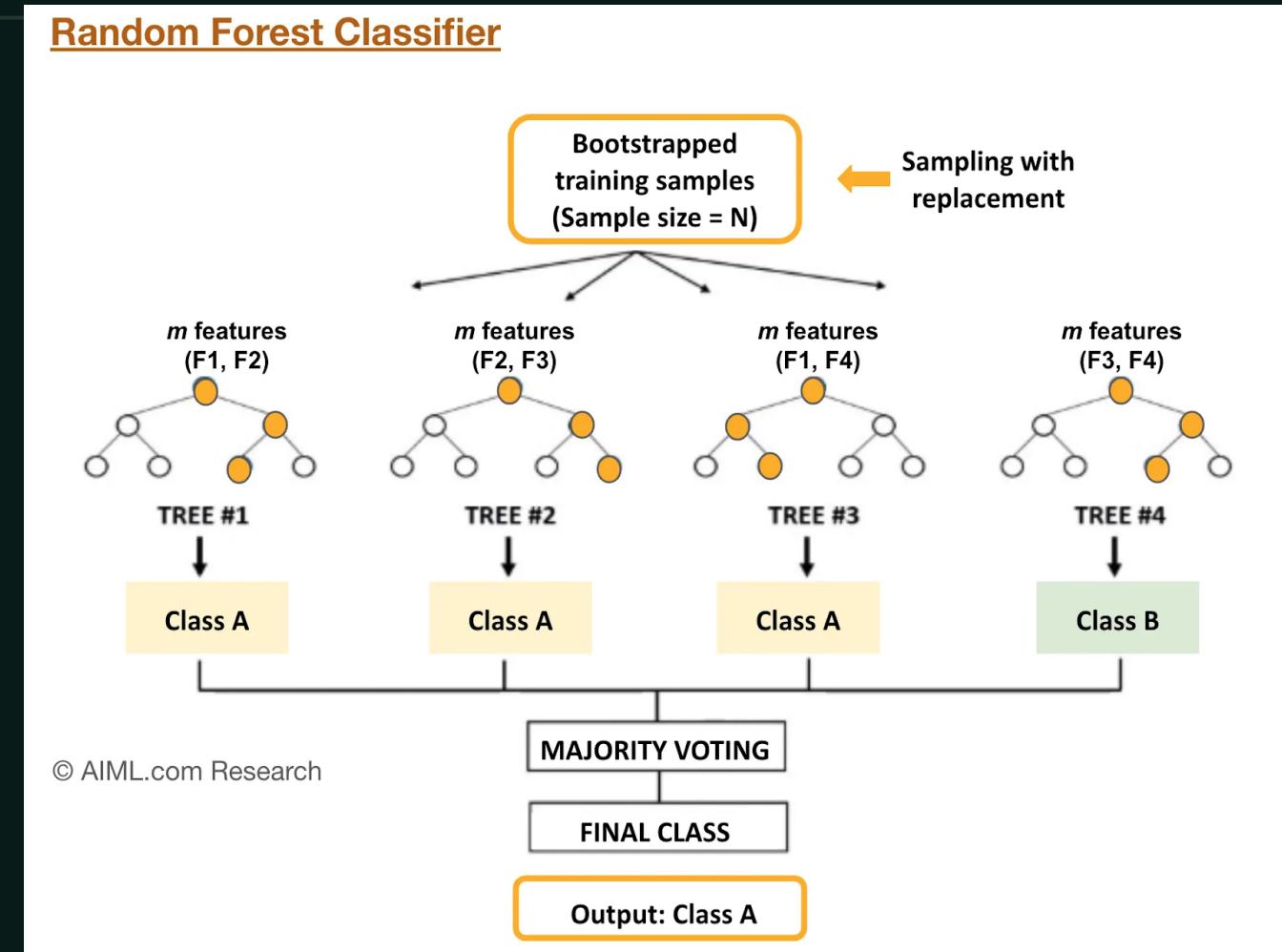
---

- OOB pts. unseen by about 37% of trees
- Served as an internal validation set.
- Used to estimate generalization error.
- Can serve as cross-validation for hyperparameter tuning



# Random Forest Predictions

- Each tree provides one vote
- Unweighted majority voting
  - Weights implicitly defined in trees
- Averaging for regression



# Benefits of Random Forests



- Classification and regression
- Mixed variable types
- Estimate of generalization error
- Trivially parallelizable
- Scale invariant
- Insensitive to outliers in predictor space
- Model non-linear relationships
- Natural produce a measure of similarity

# Random Forest Proximities

---

“[Proximities] are one of the most useful tools in random forests.”

--Leo Breiman

Original description:

“The proximities originally formed a NxN matrix. After a tree is grown, put all of the data, both training and oob, down the tree. If cases k and n are in the same terminal node increase their proximity by one. At the end, normalize the proximities by dividing by the number of trees.”

[https://www.stat.berkeley.edu/~breiman/RandomForests/cc\\_home.htm#:~:text=few%20data%20sets.-,Proximities,by%20the%20number%20of%20trees.](https://www.stat.berkeley.edu/~breiman/RandomForests/cc_home.htm#:~:text=few%20data%20sets.-,Proximities,by%20the%20number%20of%20trees.)

# Proximities in a Nutshell

---

“Proximities don’t just measure the similarity of the variables---they also take into account the importance of the variables.”

“Two cases that have quite different predictor variables might have large proximity if they differ only on variables that are not important.”

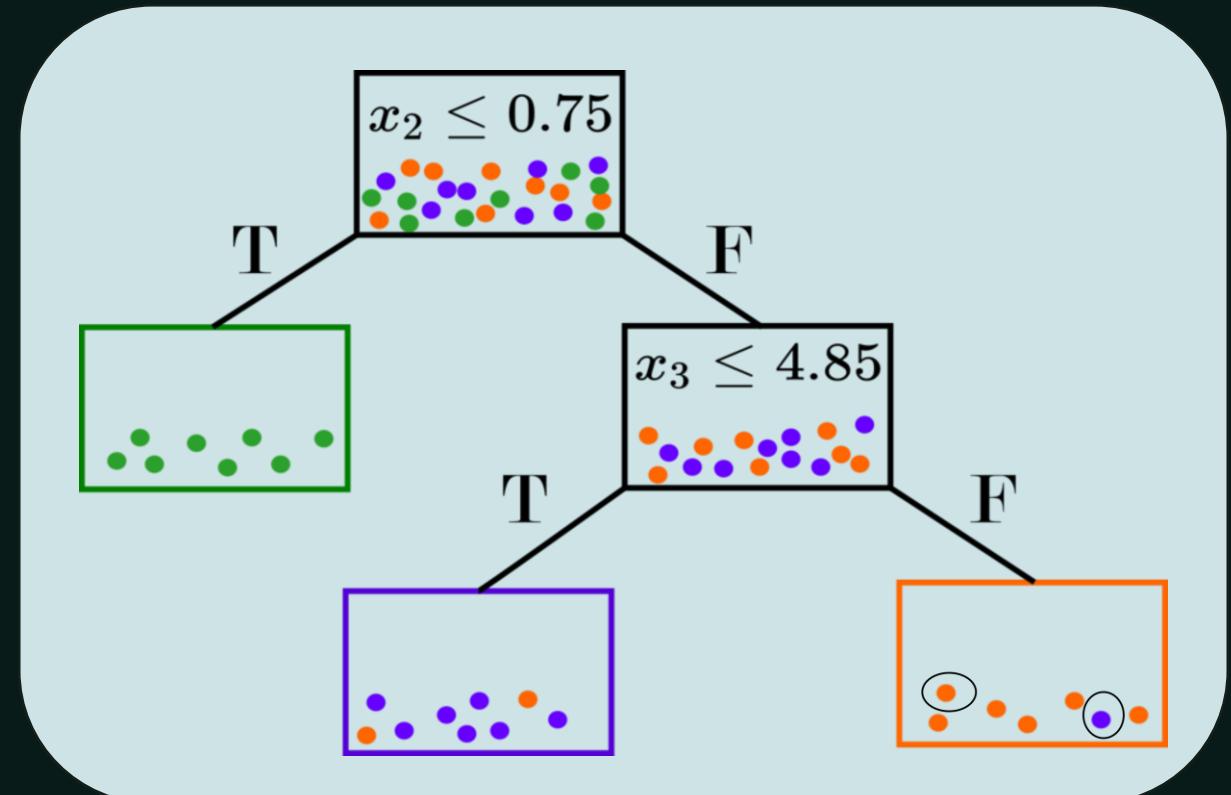
“Two cases that have quite similar values of the predictor variables might have small proximity if they differ on inputs that are important.”

--Adele Cutler

# What are Random Forest Proximities

## PROXIMITIES AS A METRIC

- A measure of “closeness”
- Supervised, but not class-conditional
- Based on splits (optimized over responses)
- Natural incorporation of variable importance



# Applications of Proximities

Visualization

Imputation

Outlier Detection

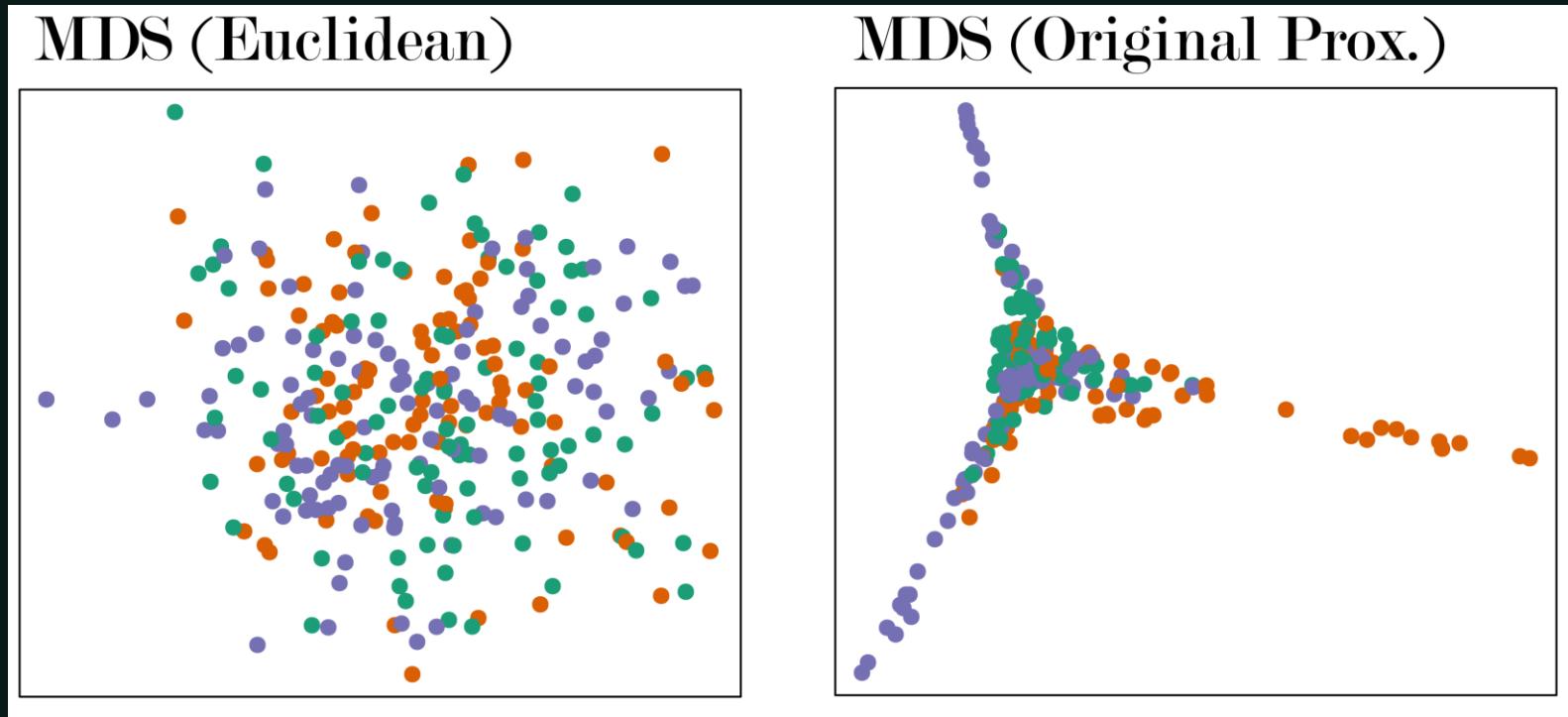
Uncertainty Quantification

Manifold Learning/ Alignment



# Other Proximity Considerations

- Overfit if use all training data
  - Trees grown until pure (typically)
  - In-bag points of different classes always in different nodes
  - Exaggerate class separation



A random sample of 300 points generated from a bivariate normal distribution was randomly assigned one of three classes. MDS was applied to  $1 - \text{proximities}$ .

# Proximities as Neighbor-based Predictor

---

What do we want?

- The similarity measure to encapsulate the supervised learning from the random forest!
- E.g., we want a proximity-weighted predictor to match the RF OOB predictions.
- This is **not the case** with the original definition, even when limiting it to OOB points only.

Why does this not work?

- The voting mechanism in random forests considers all training instances, not just pairwise relationships.

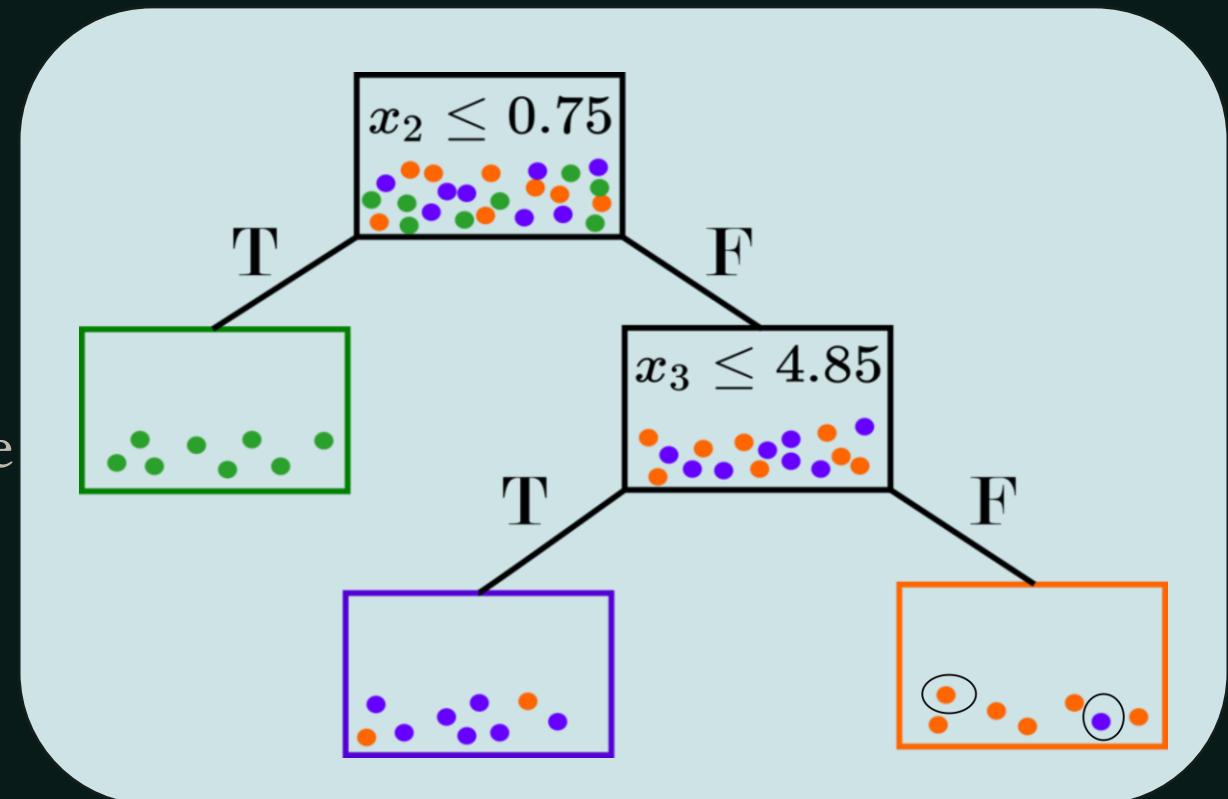
Similar to SVM Predictions:

$$\hat{y}_i(p) = \sum_{j=1}^n p(i, j) y_j$$

# Accuracy Preserving Proximities

## ACCOUNT FOR PREDICTION

- Proximities to be defined between an OOB point and all “similar” training points.
- Tree-wise proximities proportionate to size of terminal node.
- Proximity weighted predictor matches OOB or test predictions.



# Random Forest Geometry- and Accuracy-Preserving Proximities



$T$ : Set of trees in the forest ( $|T| = T$ )



$S_i$ : Trees where obs.  $i$  is OOB



$J_i(t)$ : In-bag indices in node with  $i$  in tree  $t$



$M_i(t)$ : Multiset of in-bag indices in  $i$ 's terminal node

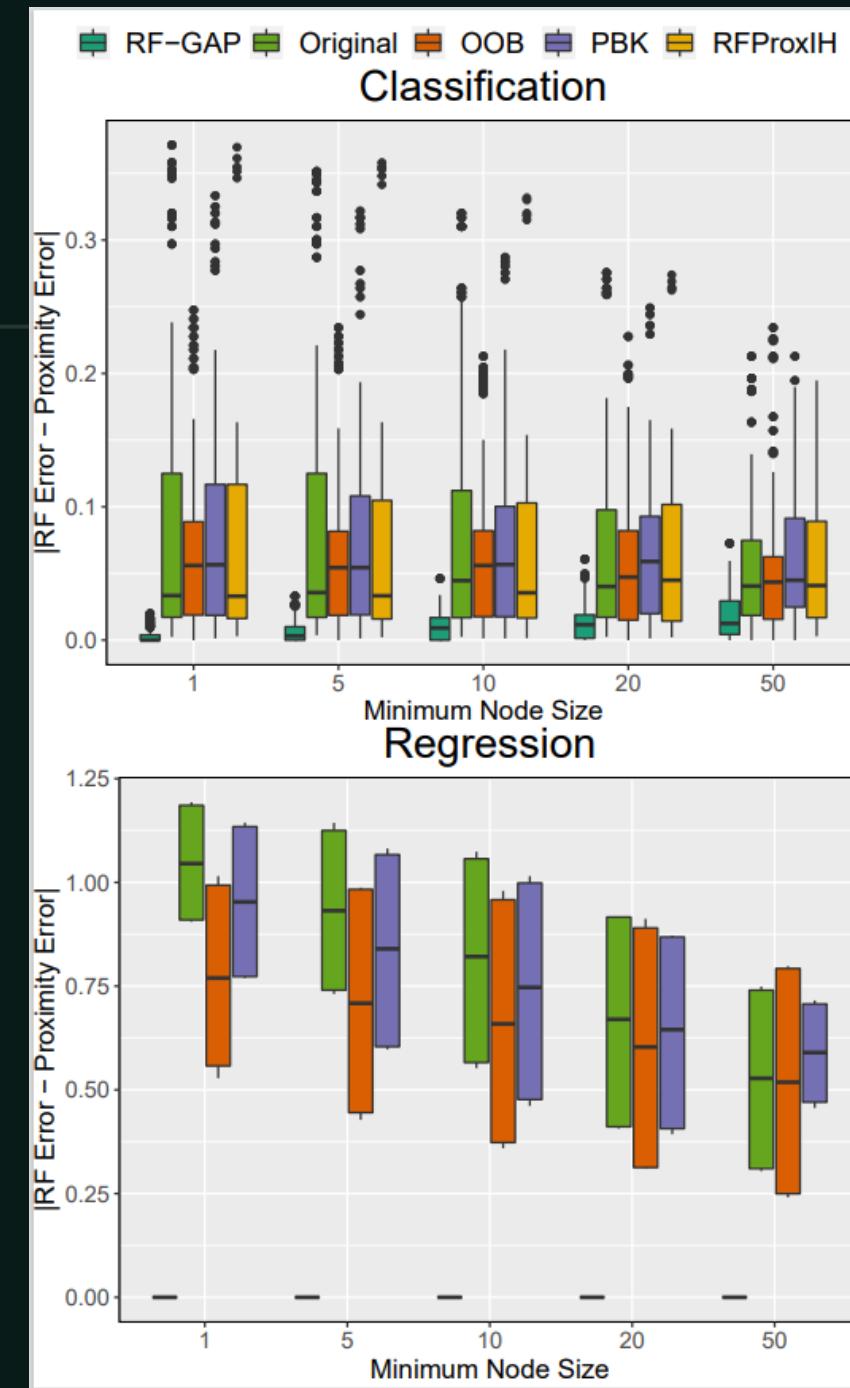


$c_j(t)$ : In-bag count of obs.  $j$  in tree  $t$ .

$$p_{GAP}(i, j) = \frac{1}{|S_i|} \sum_{t \in S_i} \frac{c_j(t) \cdot I(j \in J_i(t))}{|M_i(t)|}.$$

# Proximity-Weighted Errors

- As desired, RF predictions match proximity predictions
- (Results aggregated across 24 datasets / 10 repetitions)



# RF-GAP Proximities Notes

---

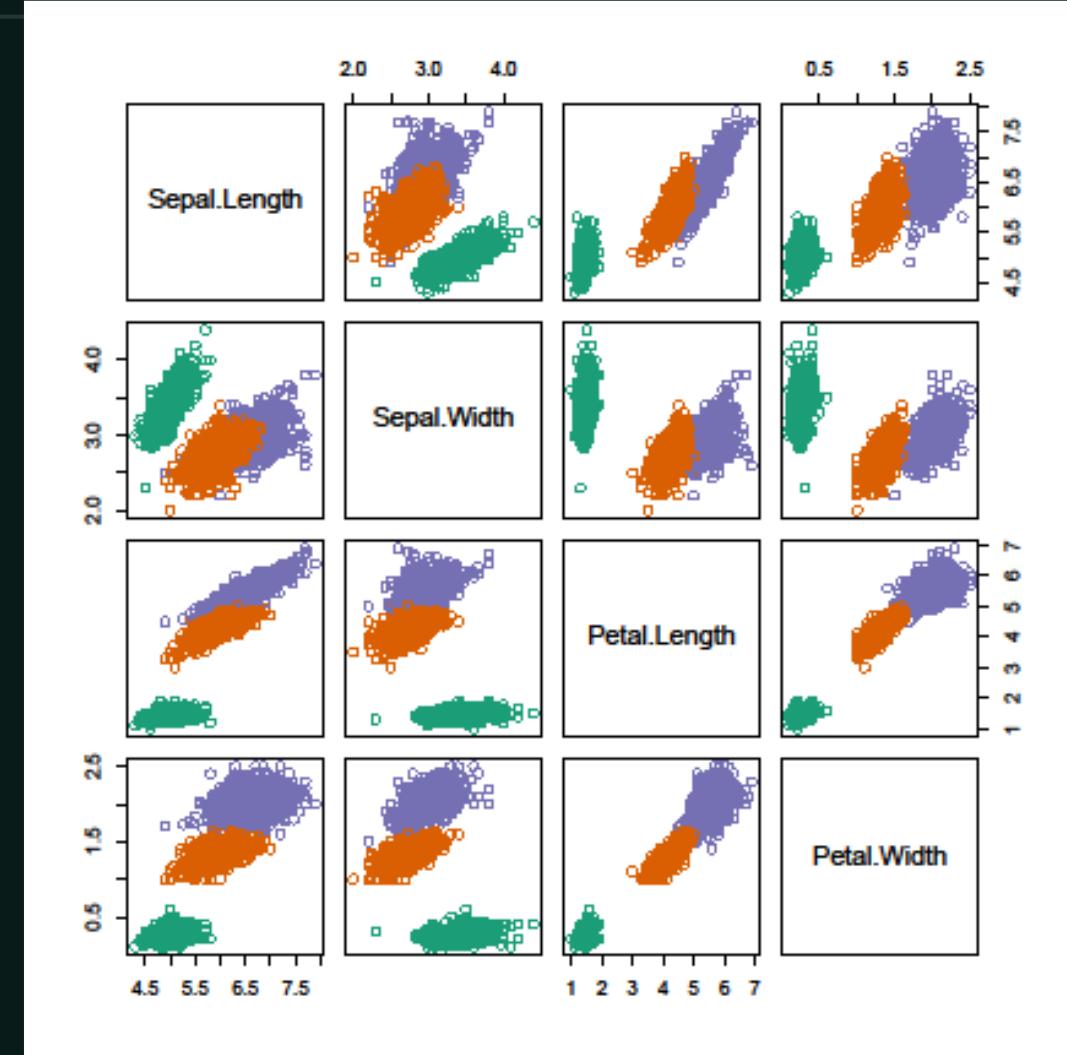
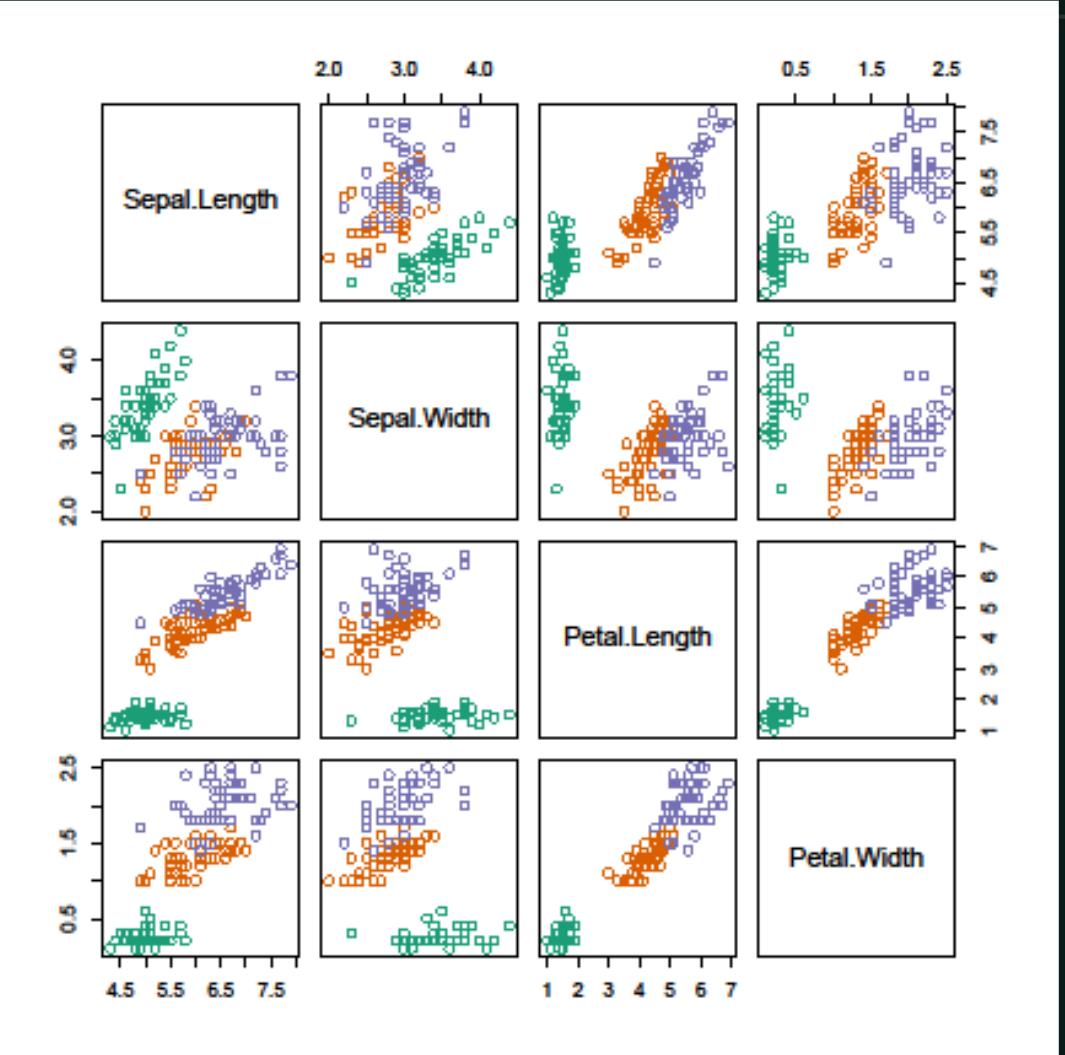
- RF-GAP proximities are not symmetric!
- By design,  $p(i, i) = 0$  to serve as proper weights for RF predictions.
- For some applications, we must reassign self-similarity and symmetrize.

$$\begin{cases} \frac{1}{|\bar{S}_i|} \sum_{t \in \bar{S}_i} \frac{c_i(t)}{|M_i(t)|}, & j = i, \\ \frac{1}{|S_i|} \sum_{t \in S_i} \frac{c_j(t) I(j \in J_i(t))}{|M_i(t)|}, & j \neq i, \end{cases}$$

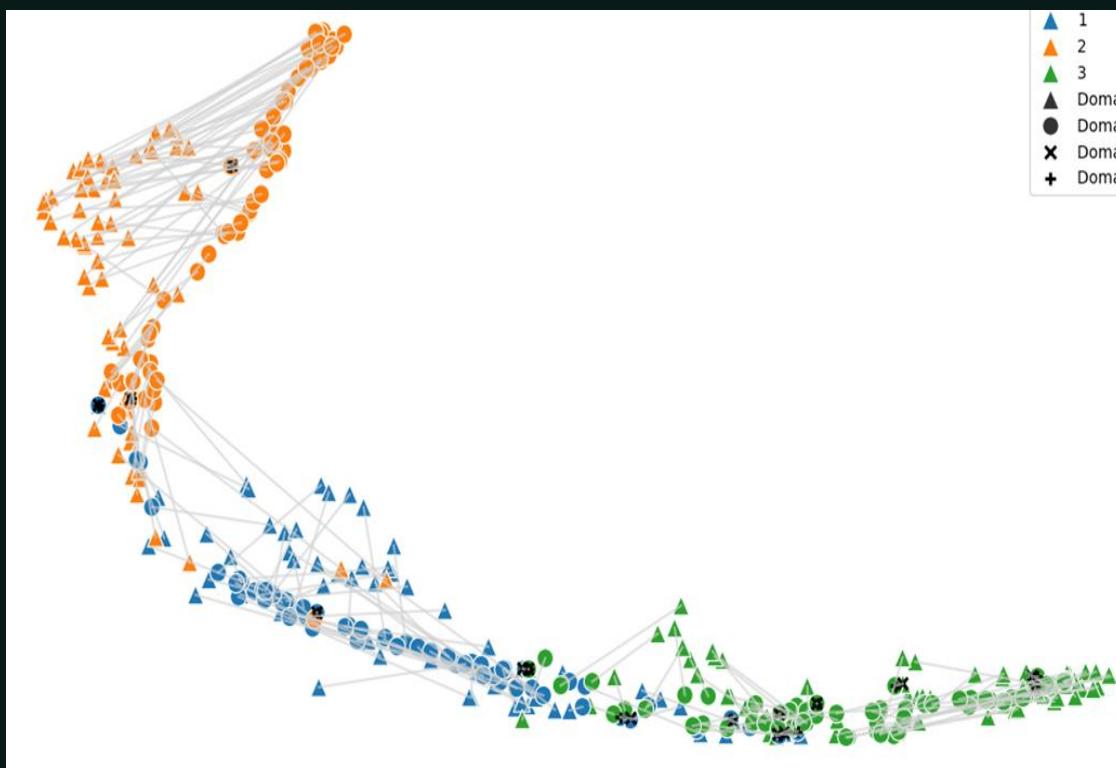
- This guarantees that self-similarity is maximal but on the same scale.

# Bonus Applications

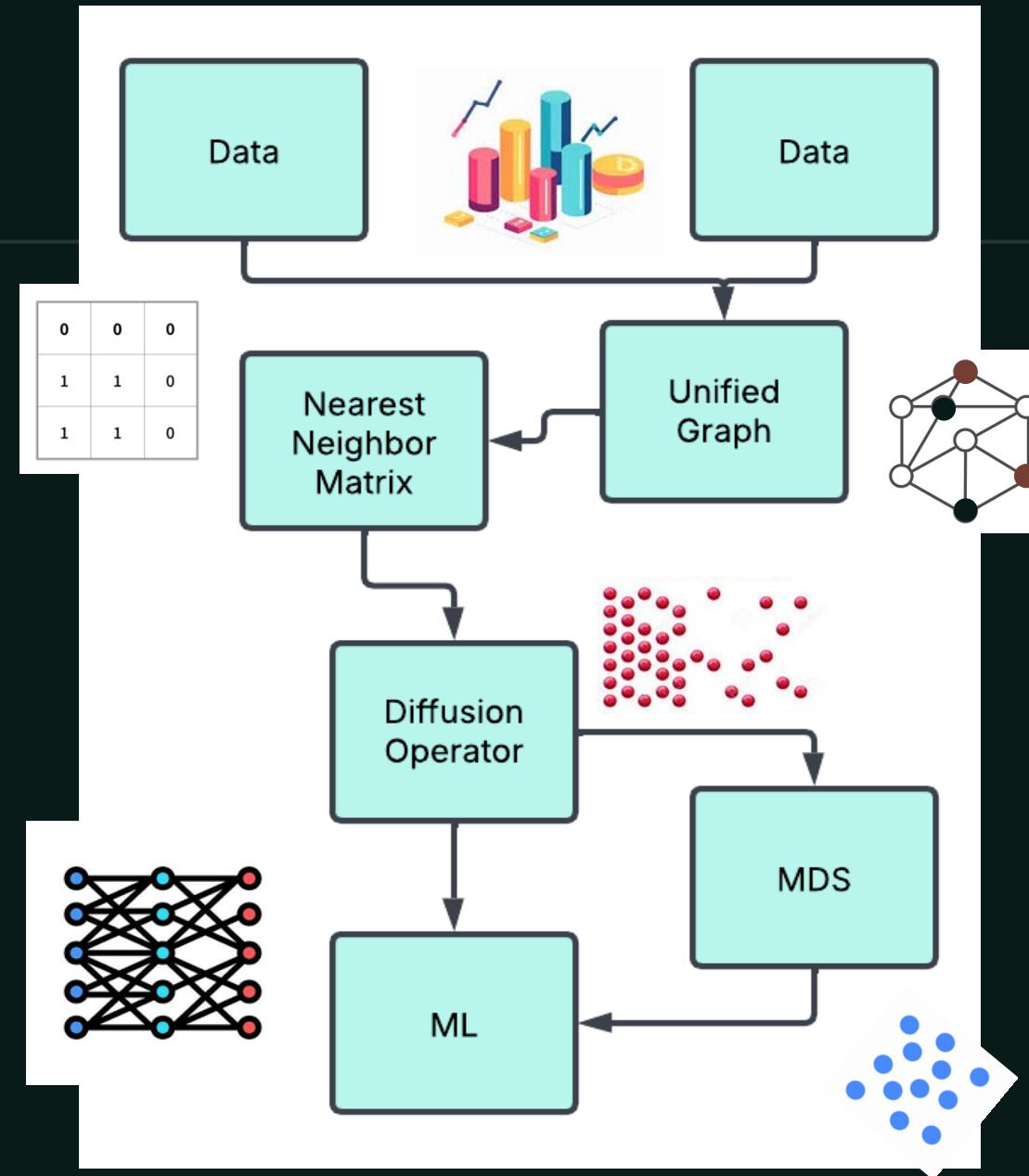
# Oversampling



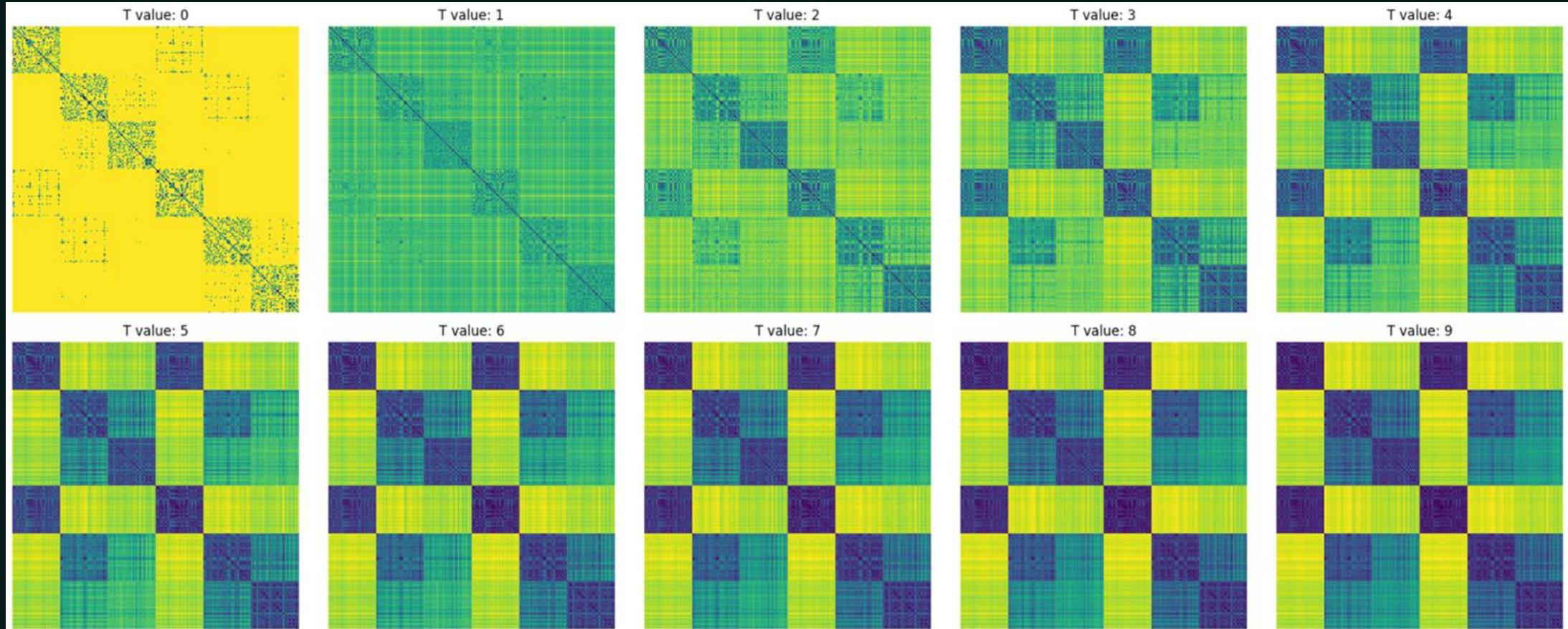
# RF-Manifold Alignment



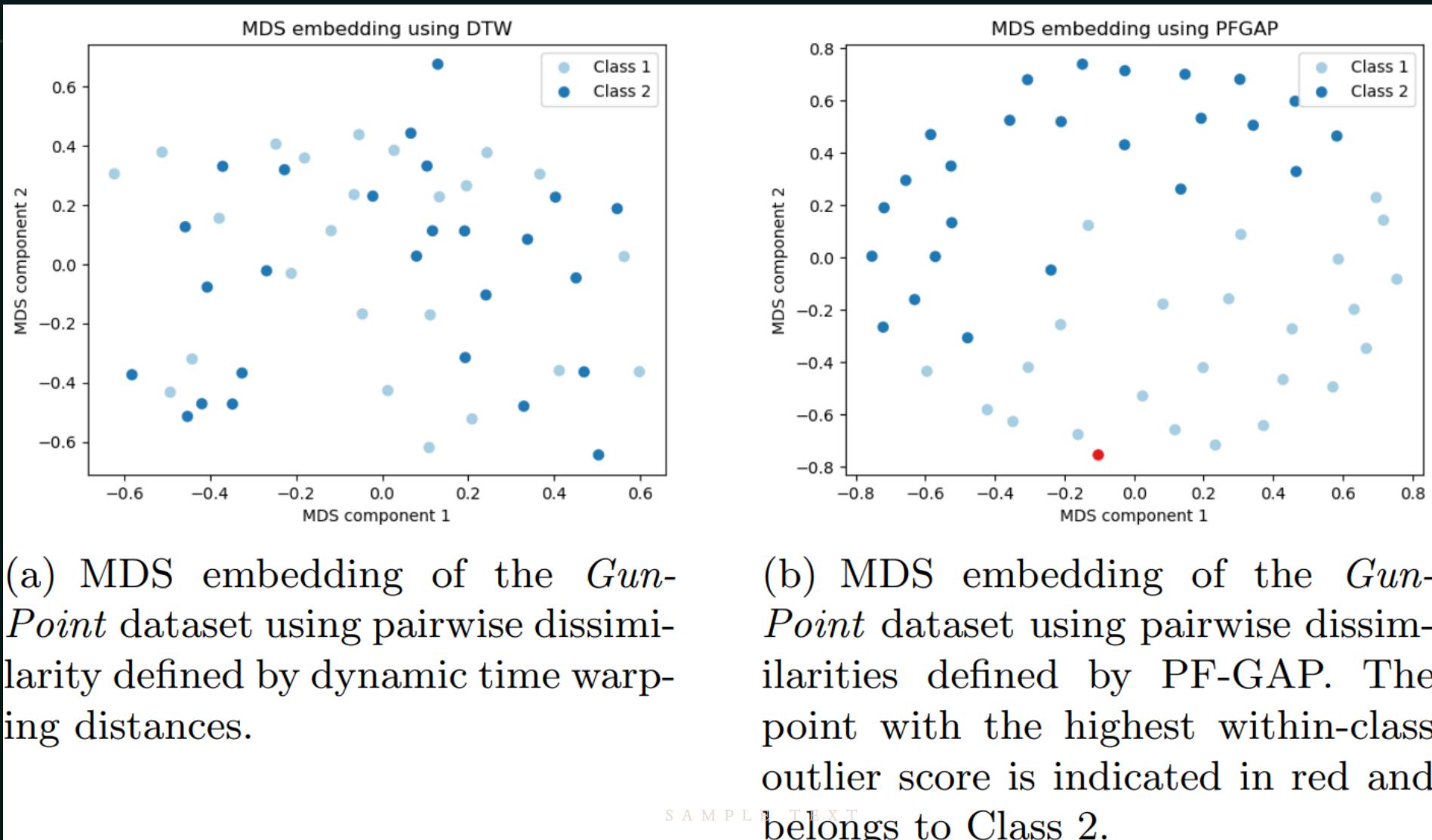
RF-MASH's resulting manifold on the  
seeds data set



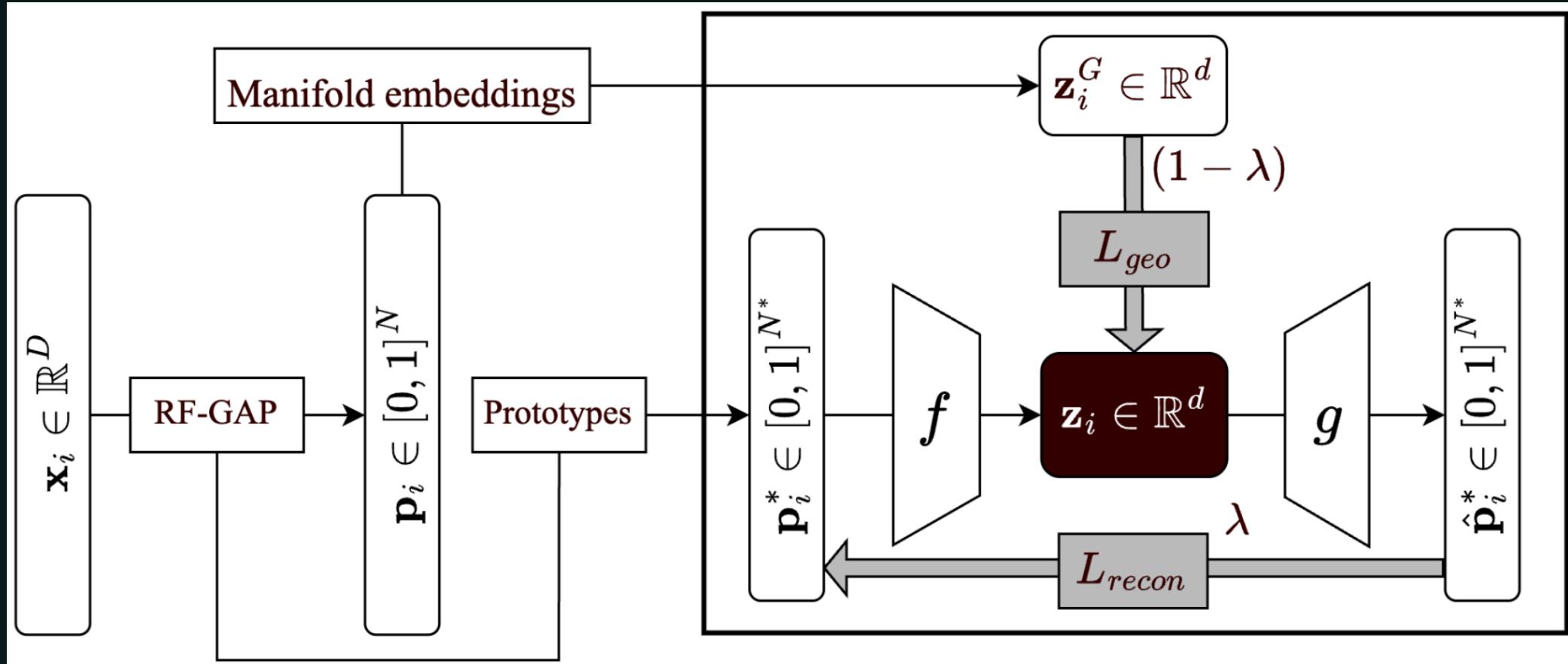
# Diffusion over Combined Proximities



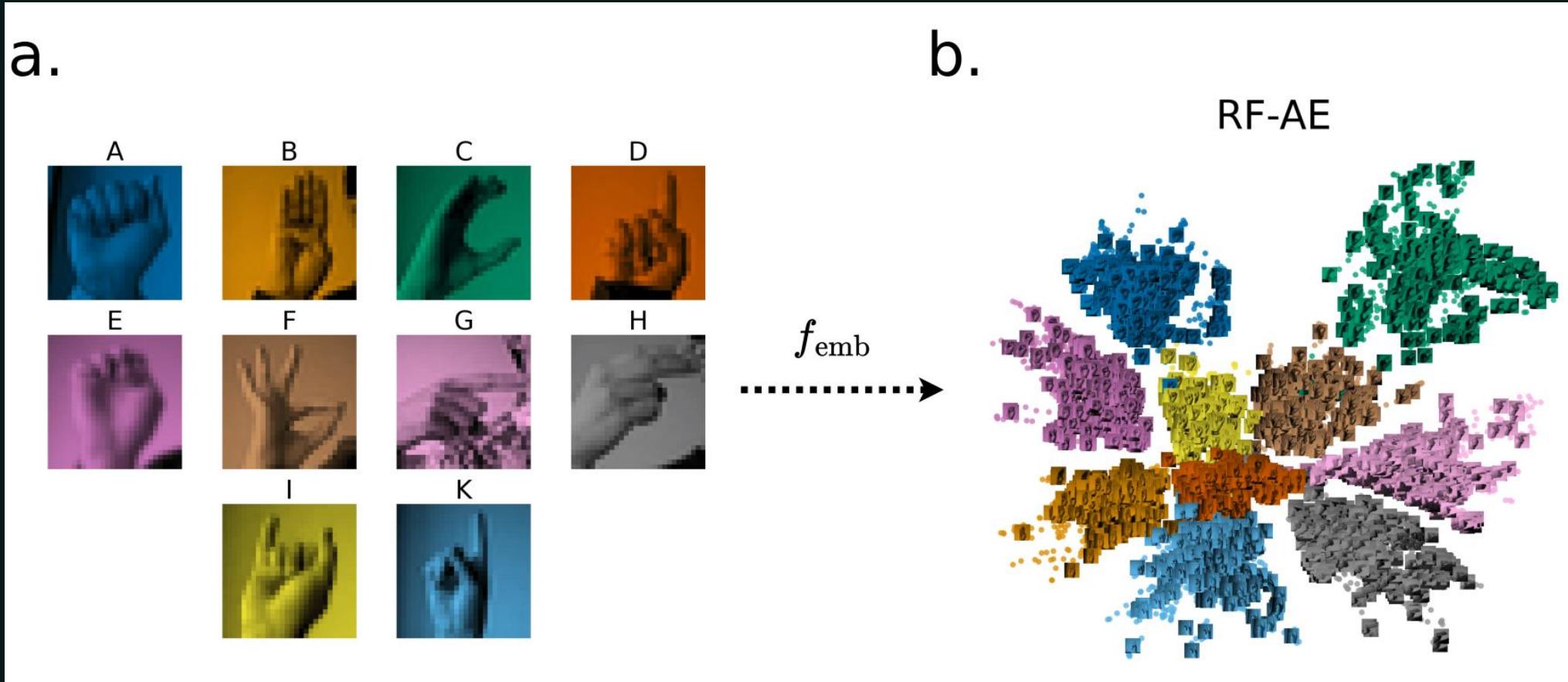
# Time Series Classification – Proximity Forests



# Random Forest Autoencoders



# RF-AE Example





The way to get started is to quit talking and begin doing.

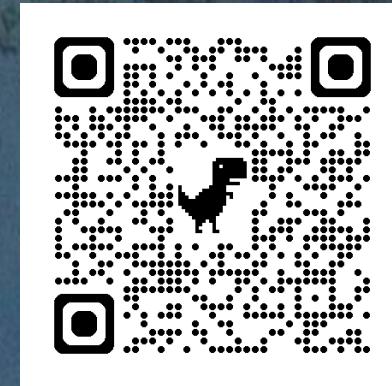
Walt Disney



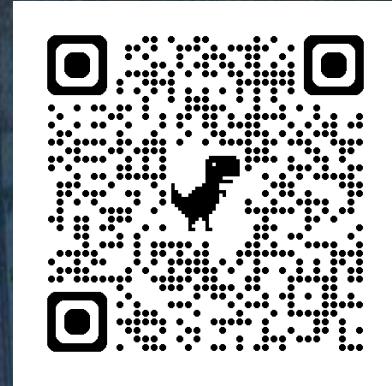
# Thank you



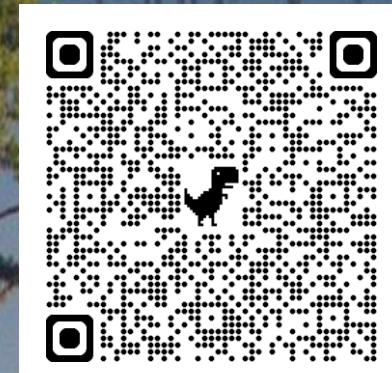
RF-GAP



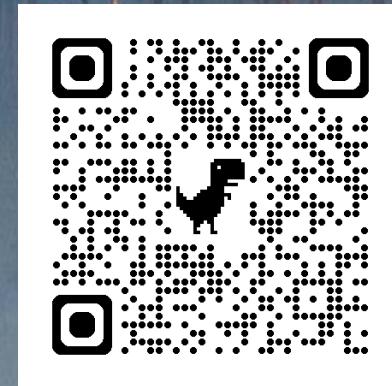
RF-PHATE



Time Seres (PF-GAP)



RF-MA



RF-AE