DATA1002 Week 10 Tutorial

Monday 13/10/25

Tutorial Outline

- Content revision (Machine Learning Intro) [15 min]
- Content revision (Clustering) [30 min]
- Content revision (Supervised Learning) [30 min]
- Python Task & Research Task (for Assignment 2) [15 min]
- Revision for Coding Test [30 min]

THE UNIVERSITY OF SYDNEY

Tutor: *Tommy Lu*

Housekeeping

Group Project Stage 2 (Presentation)

Due: 11:59 PM on Sunday at the end of week 12 (Nov 2nd)

Value: 8% of Total Mark

Note: Get started your project ASAP. Discuss with your tutors and make use of Ed to ask questions.

- Aggregate summaries
- Charts and visual representations
- Machine learning predictions
- Presentations

Data analysis	**Project stage 2** in-class group presentation on predictive model and evaluation of its success.	8%	Week 12

Content

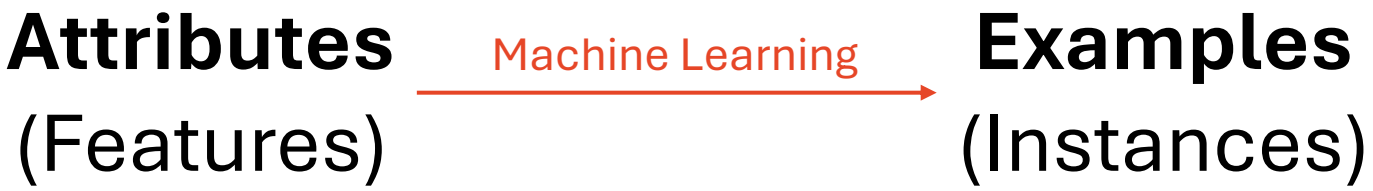
Introduction to Machine Learning

Data Science Process

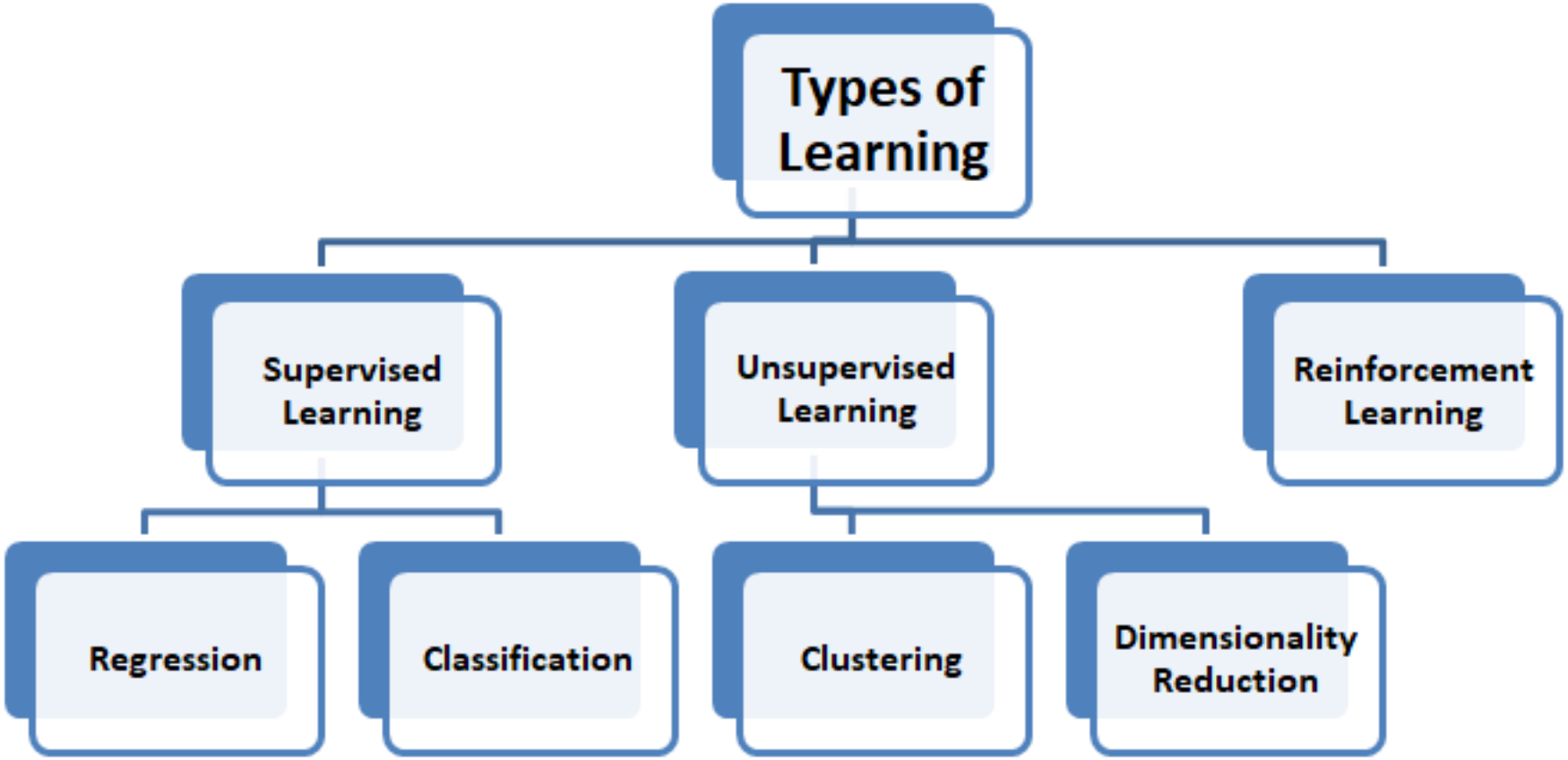
Attributes
(Features)

Examples
(Instances)

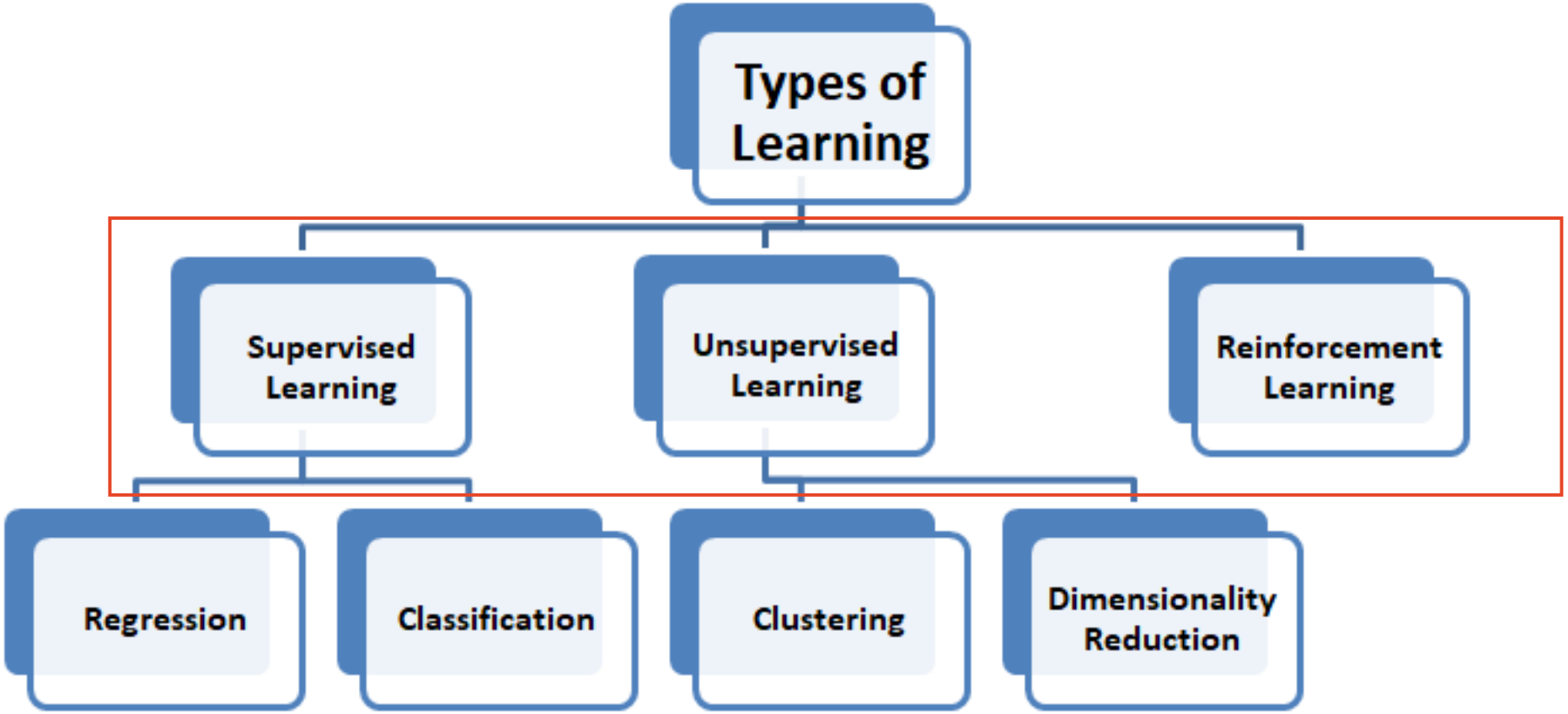
Data Science Process



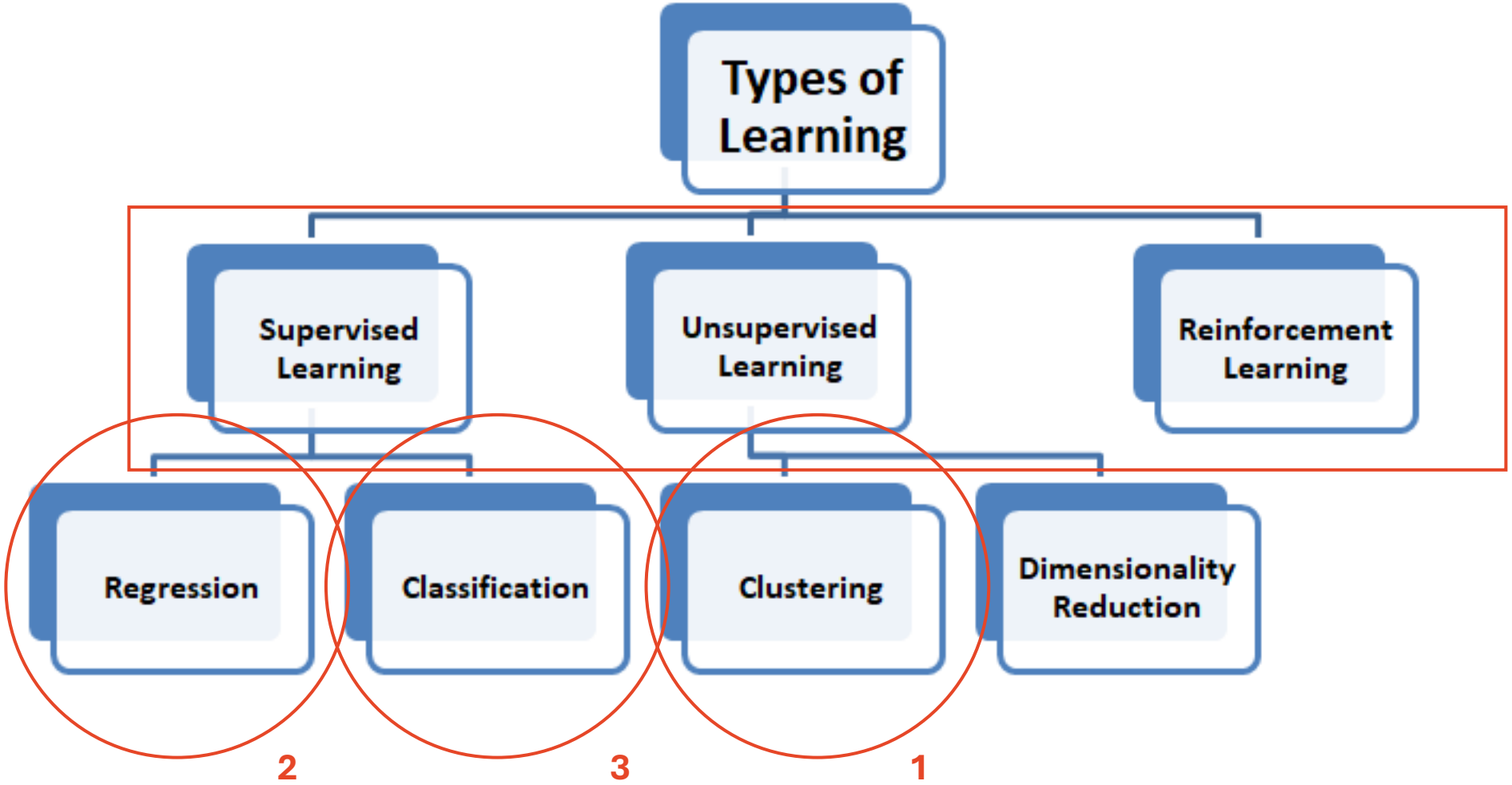
Machine Learning



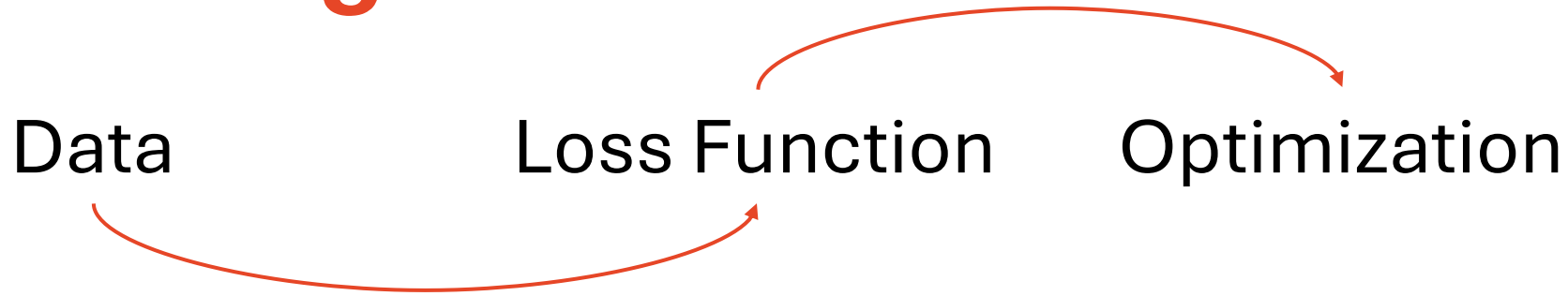
Machine Learning



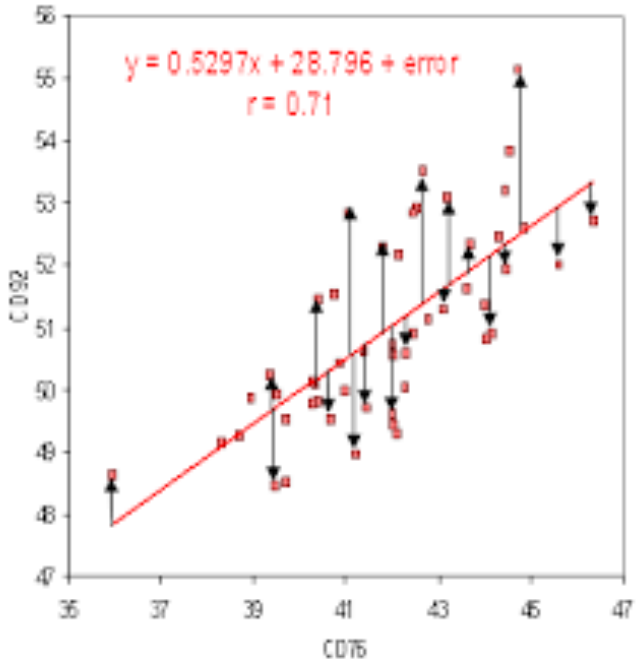
Machine Learning



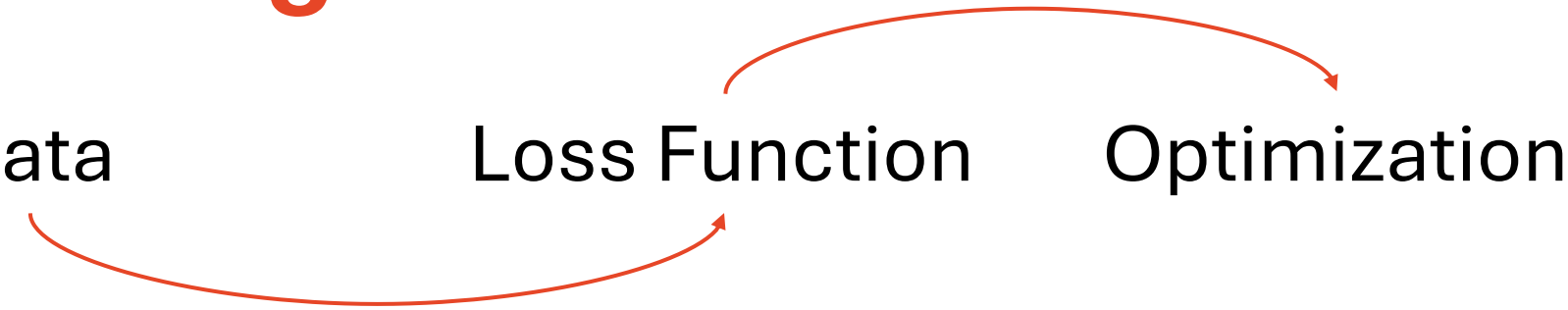
Model Training



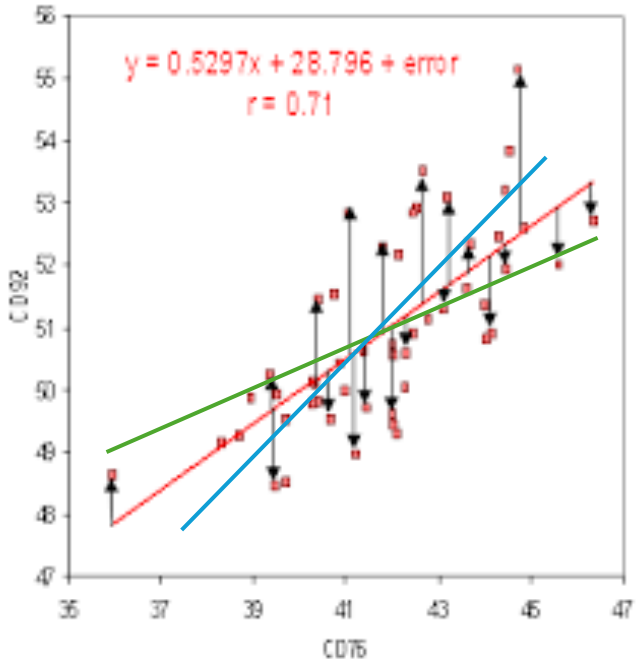
Model training uses data, a loss function, and an optimization method (e.g., gradient descent)



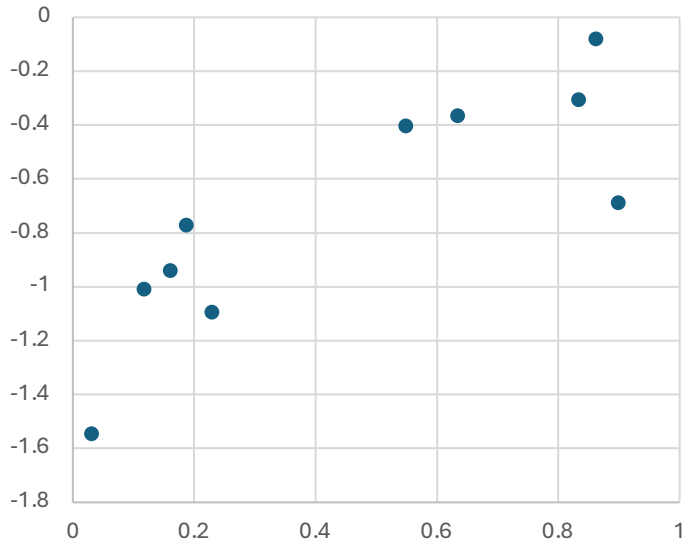
Model Training



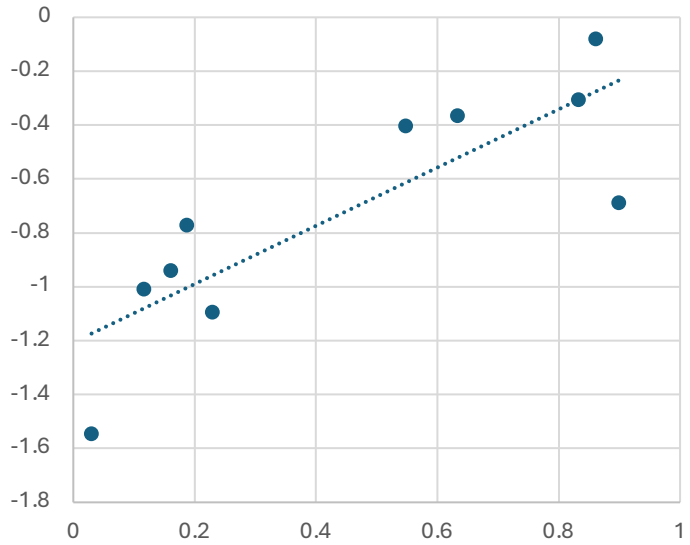
Model training uses data, a loss function, and an optimization method (e.g., gradient descent)



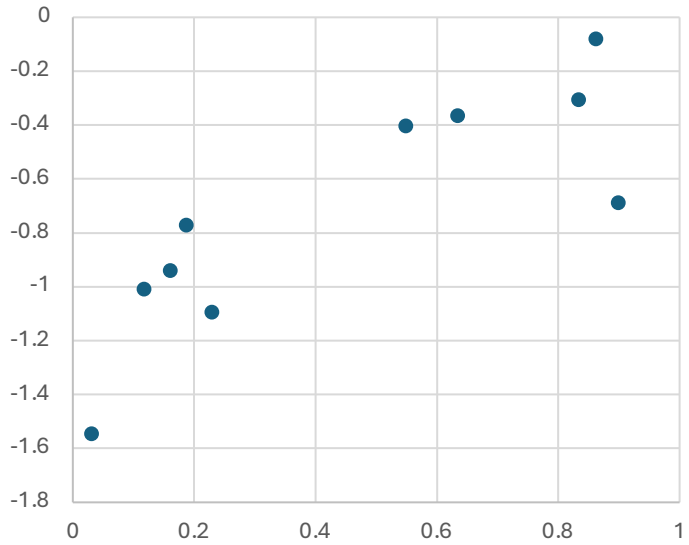
Model Training

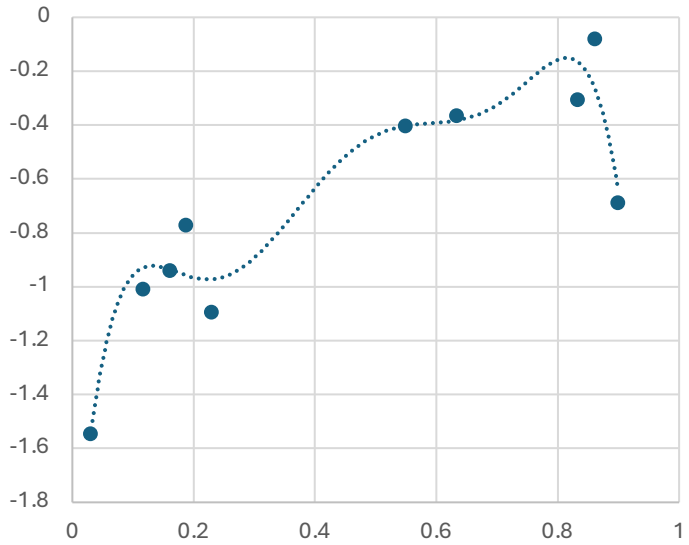


Model Training



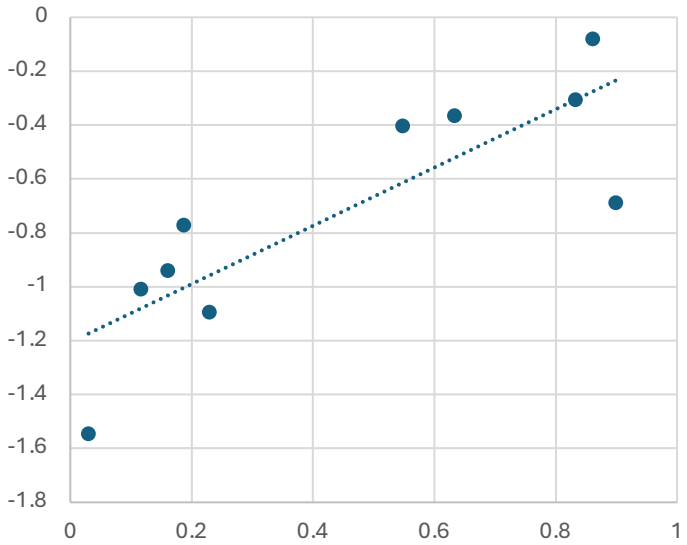
Underfitting



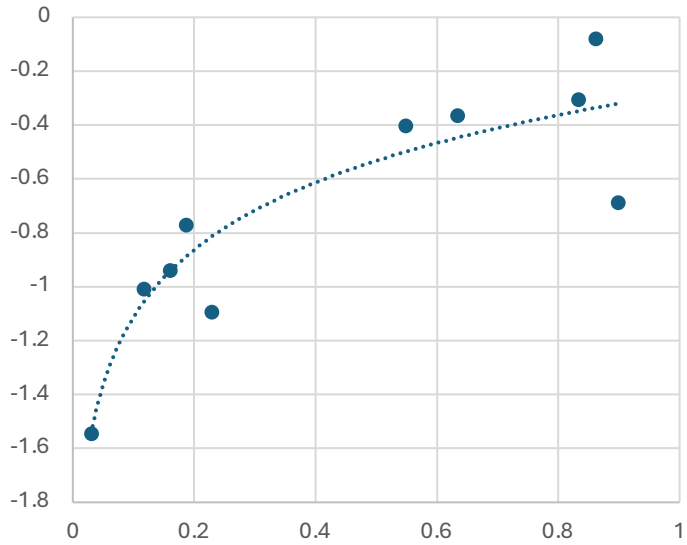


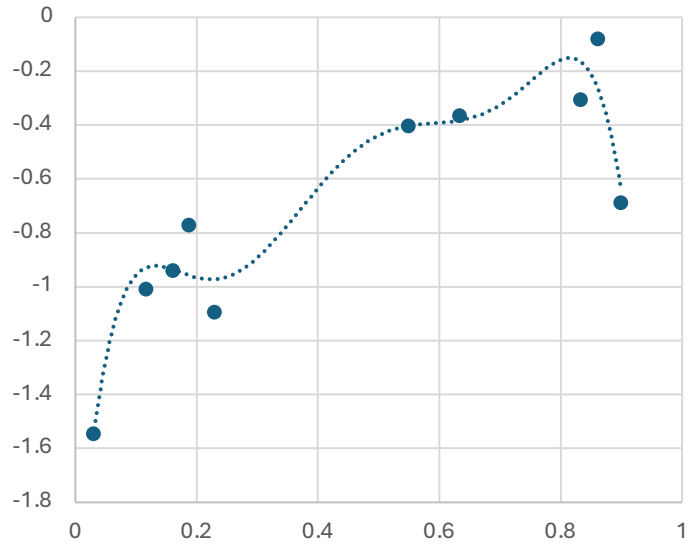
Overfitting

Model Training



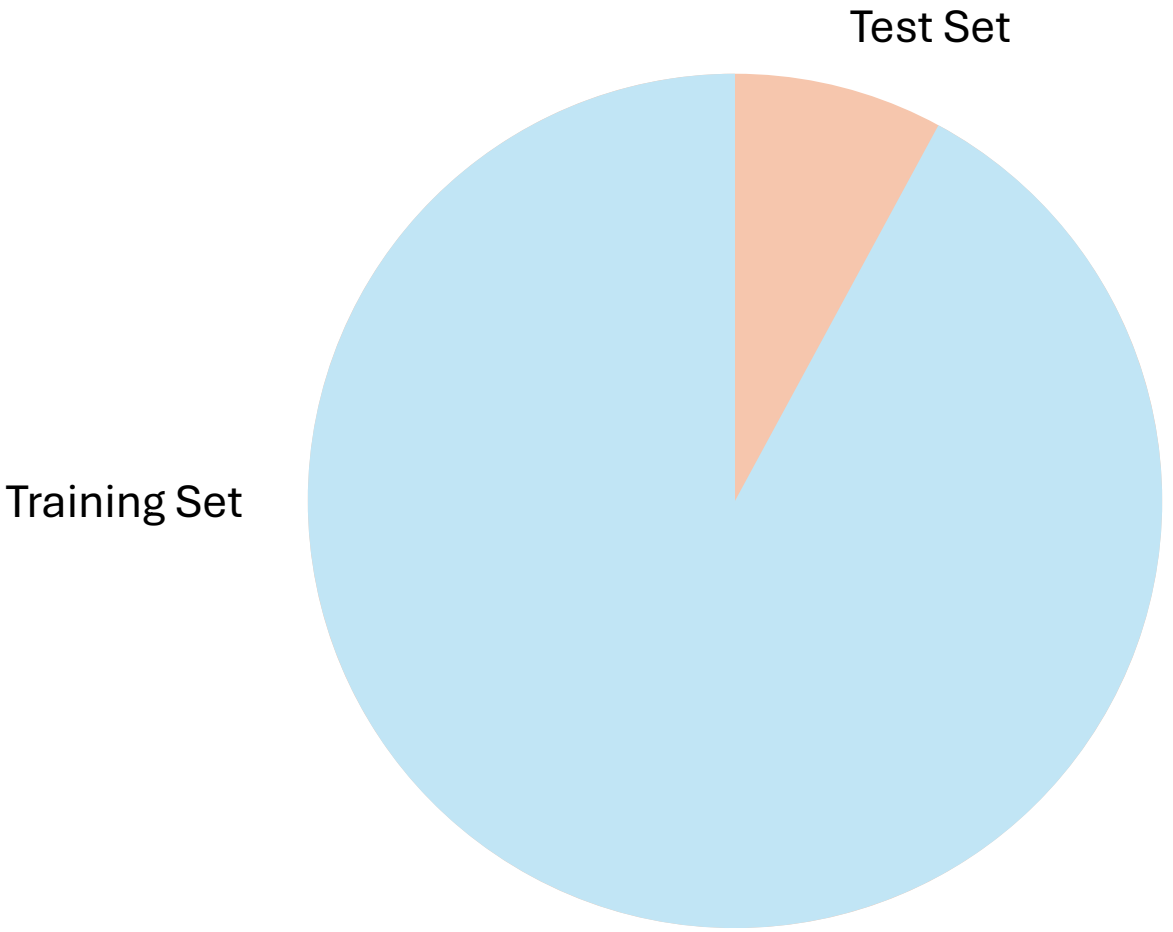
Underfitting



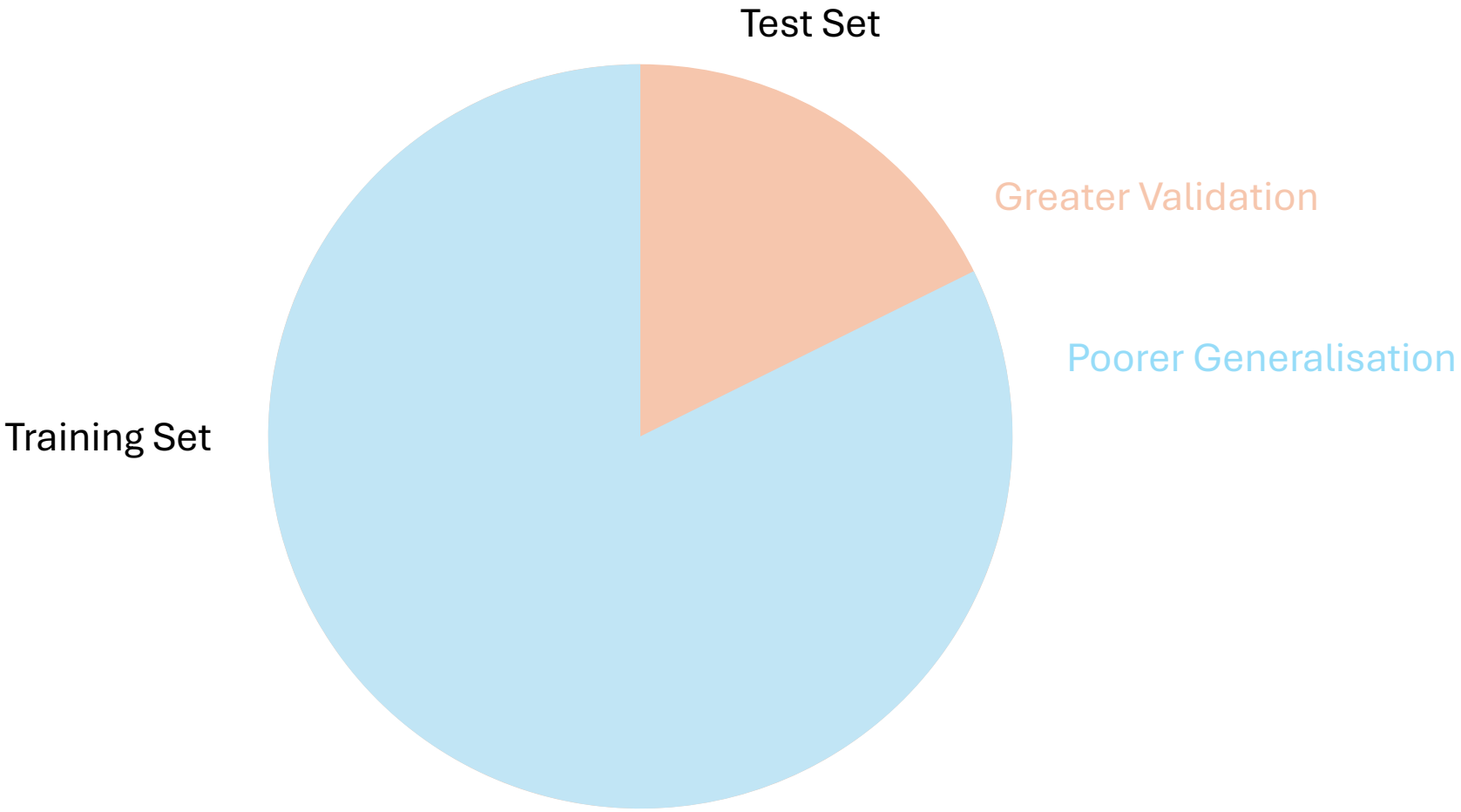


Overfitting

Model Training



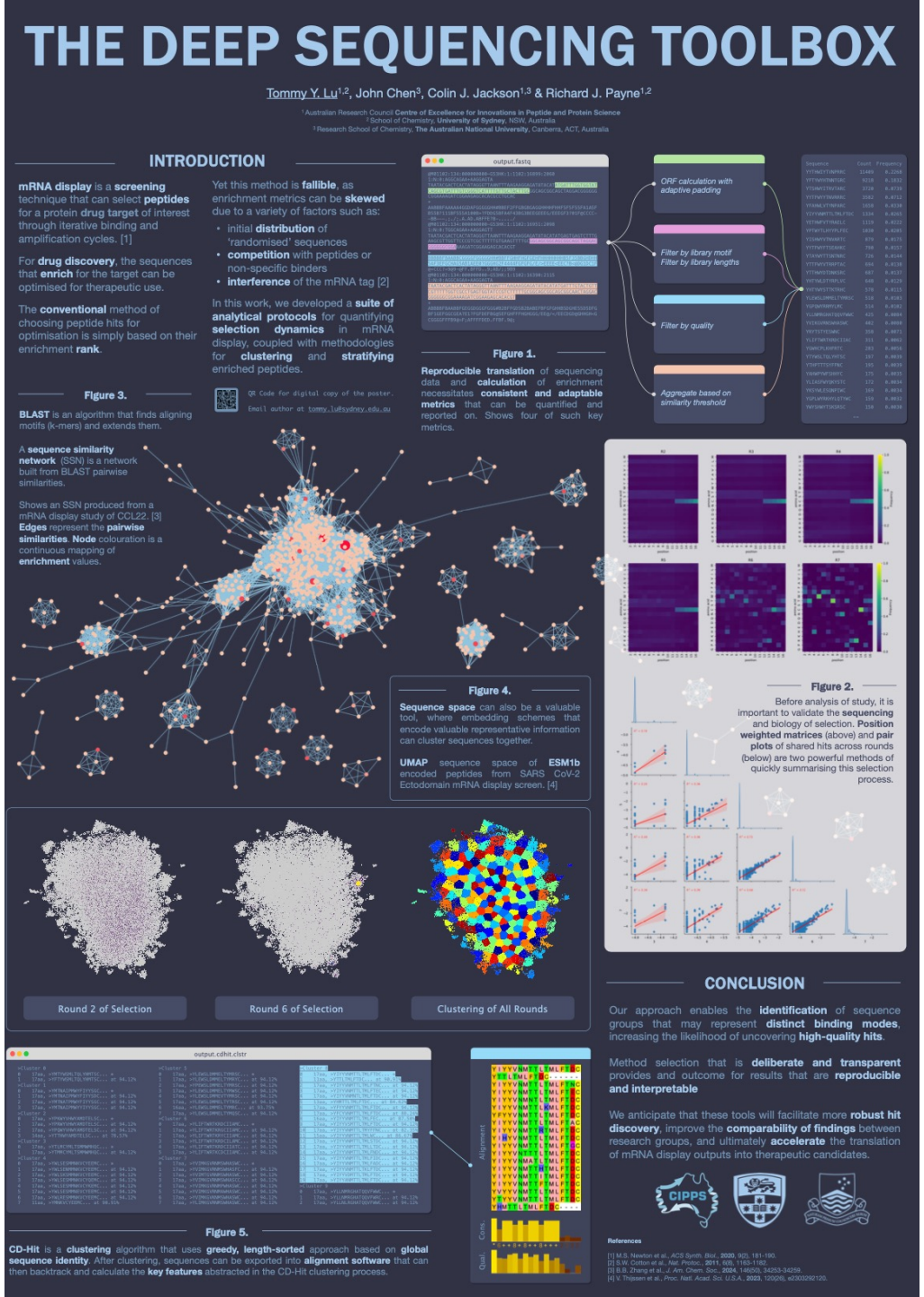
Model Training



Content

Clustering

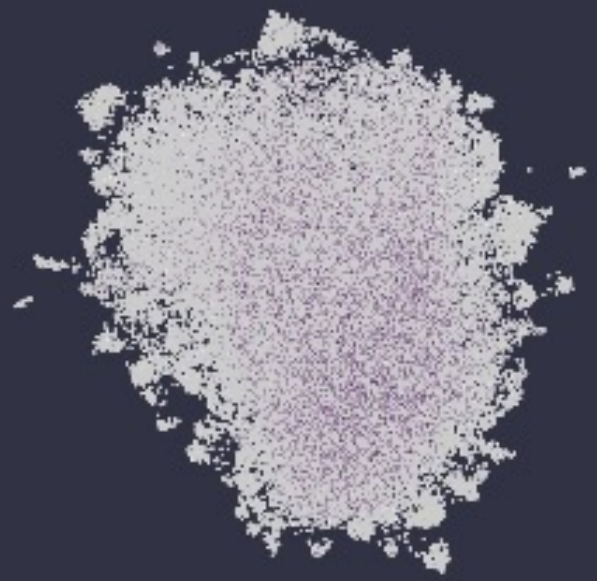
Clustering


THE DEEP SEQUENCING TOOLBOX
Tommy Y. Lu[1,2], John Chen[3], Colin J. Jackson[1,3] & Richard J. Payne[1,2]
[1] Australian Research Council Centre of Excellence for Innovations in Peptide and Protein Science
[2] School of Chemistry, University of Sydney, NSW, Australia
[3] Research School of Chemistry, The Australian National University, Canberra, ACT, Australia
INTRODUCTION
mRNA display is a screening technique that can select peptides for a protein drug target of interest through iterative binding and amplification cycles. [1]
For drug discovery, the sequences that enrich for the target can be optimised for therapeutic use.
The conventional method of choosing peptide hits for optimisation is simply based on their enrichment rank.
Yet this method is fallible, as enrichment metrics can be skewed due to a variety of factors such as:
initial distribution of 'randomised' sequences
competition with peptides or non-specific binders
interference of the mRNA tag [2]
In this work, we developed a suite of analytical protocols for quantifying selection dynamics in mRNA display, coupled with methodologies for clustering and stratifying enriched peptides.
Figure 1.
Reproducible translation of sequencing data and calculation of enrichment necessitates consistent and adaptable metrics that can be quantified and reported on. Shows four of such key metrics.
ORF calculation with adaptive padding
Filter by library motif
Filter by library lengths
Filter by quality
Aggregate based on similarity threshold
Figure 3.
BLAST is an algorithm that finds aligning motifs (k-mers) and extends them.
A sequence similarity network (SSN) is a network built from BLAST pairwise similarities.
Shows an SSN produced from a mRNA display study of CCL22. [3] Edges represent the pairwise similarities. Node colouration is a continuous mapping of enrichment values.
QR Code for digital copy of the poster.
Email author at tommy.lu@sydney.edu.au
Figure 4.
Sequence space can also be a valuable tool, where embedding schemes that encode valuable representative information can cluster sequences together.
UMAP sequence space of ESM1b encoded peptides from SARS CoV-2 Ectodomain mRNA display screen. [4]
Figure 2.
Before analysis of study, it is important to validate the sequencing and biology of selection. Position weighted matrices (above) and pair plots of shared hits across rounds (below) are two powerful methods of quickly summarising this selection process.
Round 2 of Selection
Round 6 of Selection
Clustering of All Rounds
CONCLUSION
Our approach enables the identification of sequence groups that may represent distinct binding modes, increasing the likelihood of uncovering high-quality hits.
Method selection that is deliberate and transparent provides and outcome for results that are reproducible and interpretable
We anticipate that these tools will facilitate more robust hit discovery, improve the comparability of findings between research groups, and ultimately accelerate the translation of mRNA display outputs into therapeutic candidates.
CIPPS
Figure 5.
CD-Hit is a clustering algorithm that uses greedy, length-sorted approach based on global sequence identity. After clustering, sequences can be exported into alignment software that can then backtrack and calculate the key features abstracted in the CD-Hit clustering process.
References

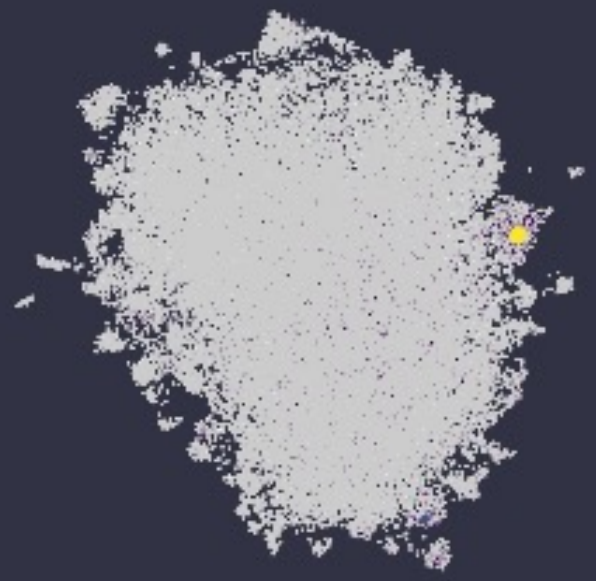
Clustering

can cluster sequences together.

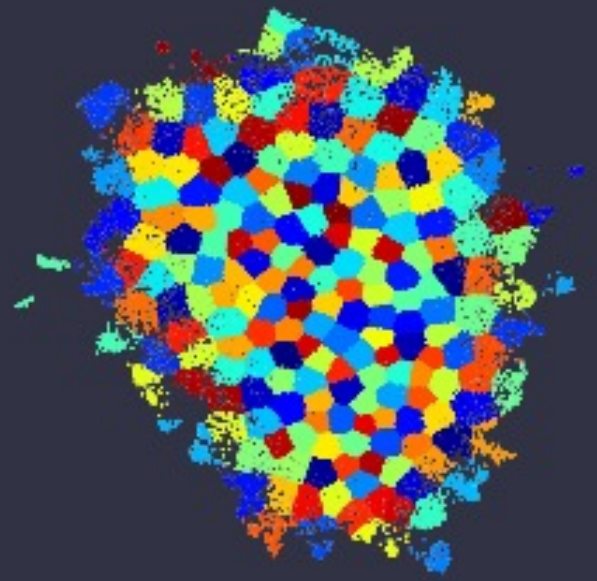
UMAP sequence space of **ESM1b** encoded peptides from SARS CoV-2 Ectodomain mRNA display screen. [4]



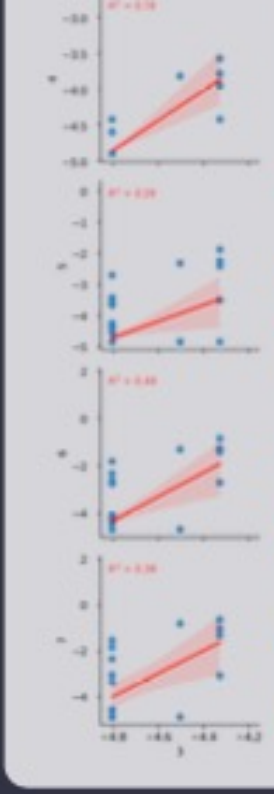
Round 2 of Selection



Round 6 of Selection



Clustering of All Rounds



Our approach
groups that
increasing the

Method selec
provides and
and interpreta

We anticipate
discovery, imp

THE

Clustering

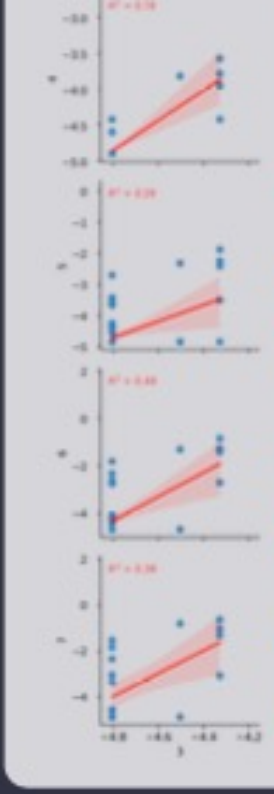
can cluster sequences together.

UMAP sequence space of **ESM1b** encoded peptides from SARS CoV-2 Ectodomain mRNA display screen. [4]

Round 2 of Selection

Round 6 of Selection

Clustering of All Rounds

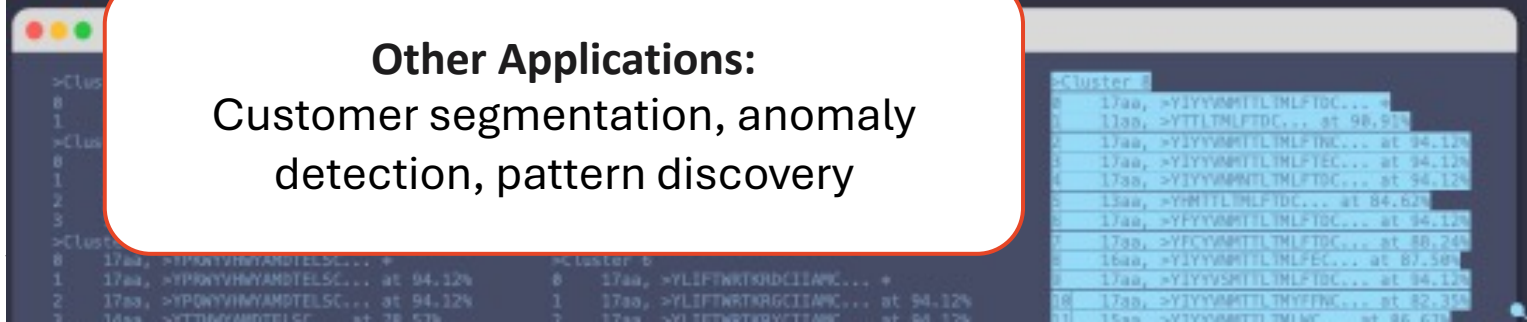


Our approach
groups that
increasing the

Method selec
provides and
end interpreta

We anticipate
discovery, imp

Other Applications:
Customer segmentation, anomaly detection, pattern discovery

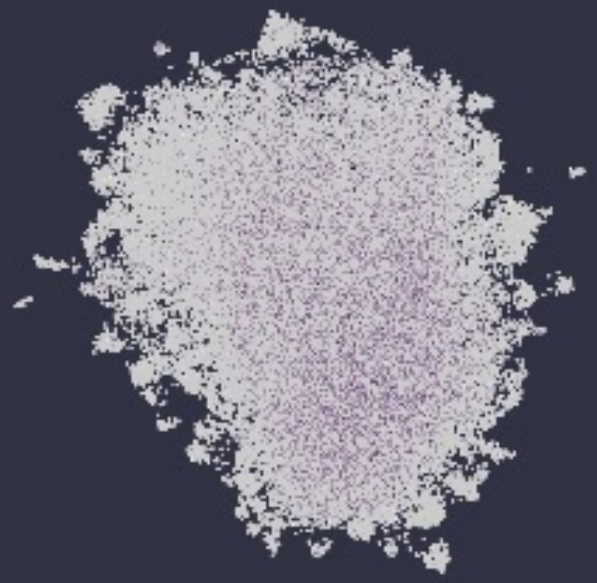


Similarity metrics:
Euclidean, Manhattan, Cosine

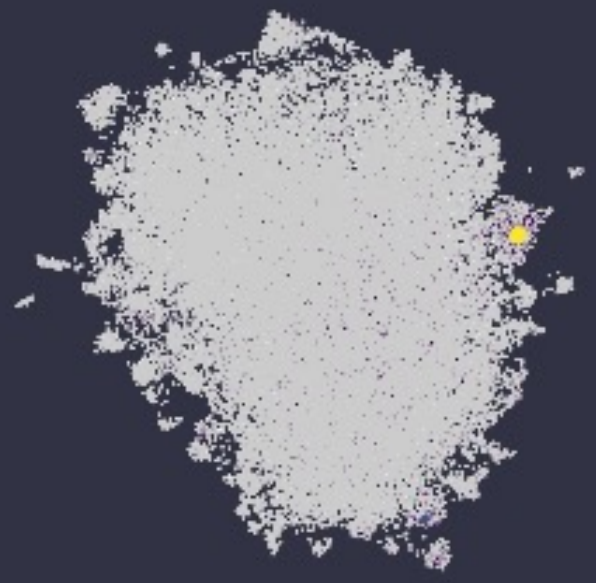
Clustering

can cluster sequences together.

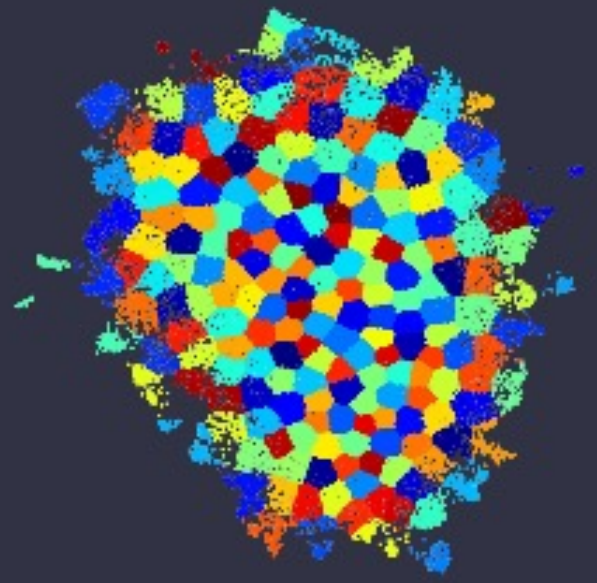
UMAP sequence space of **ESM1b** encoded peptides from SARS CoV-2 Ectodomain mRNA display screen. [4]



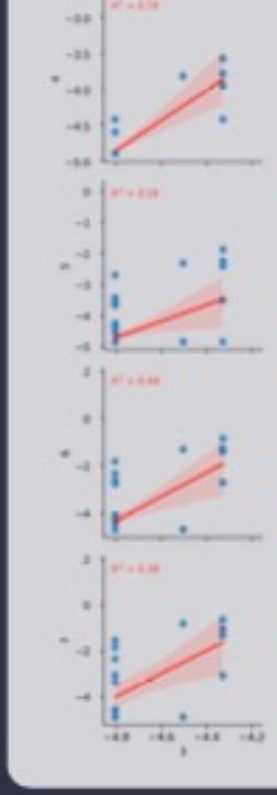
Round 2 of Selection



Round 6 of Selection



Clustering of All Rounds

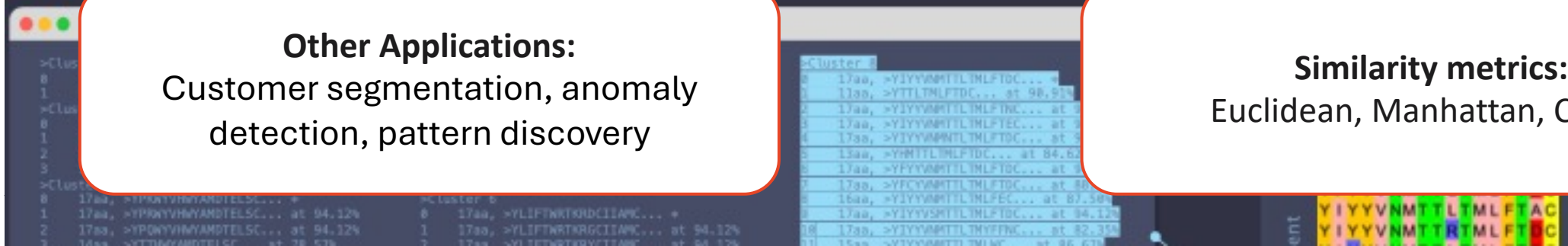


Our approach
groups that
increasing the

discovery, imp

Other Applications:
Customer segmentation, anomaly detection, pattern discovery

Similarity metrics:
Euclidean, Manhattan, Cosine

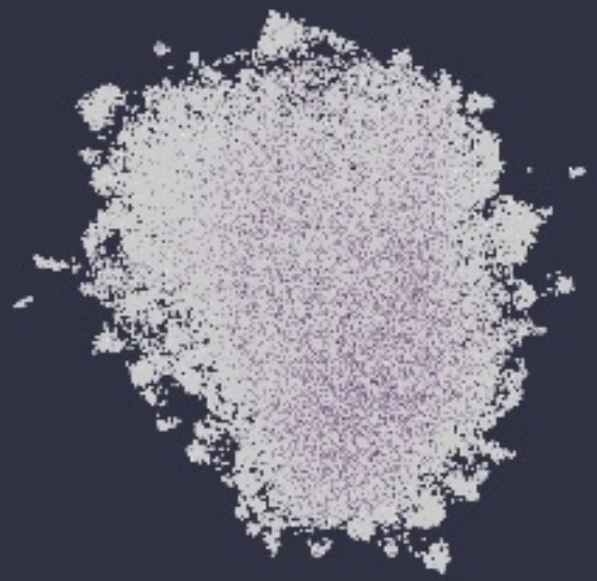


THE

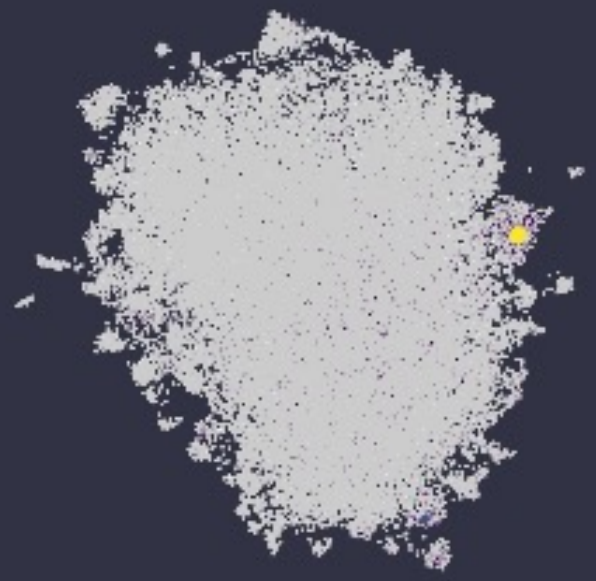
Clustering

can cluster sequences together.

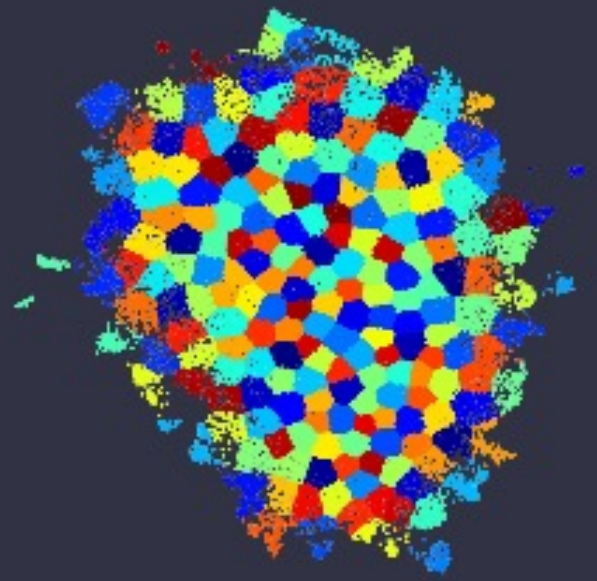
UMAP sequence space of **ESM1b** encoded peptides from SARS CoV-2 Ectodomain mRNA display screen. [4]



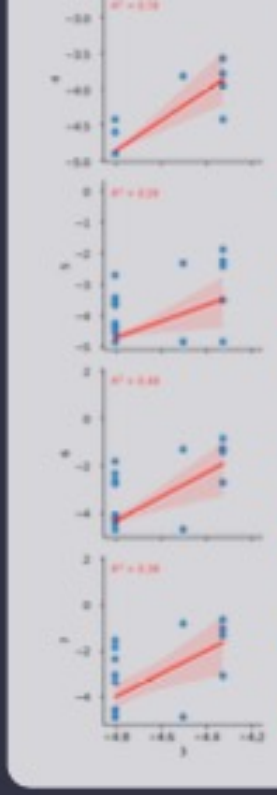
Round 2 of Selection



Round 6 of Selection



Clustering of All Rounds



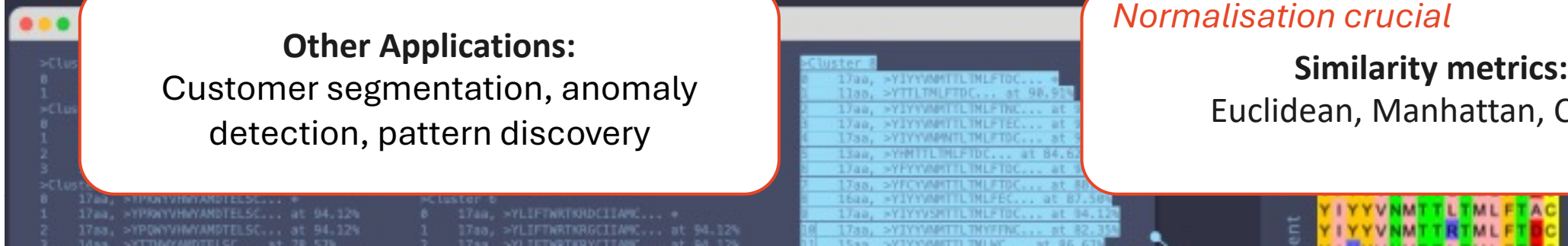
Our approach
groups that
increasing the

discovery, imp

Other Applications:
Customer segmentation, anomaly detection, pattern discovery

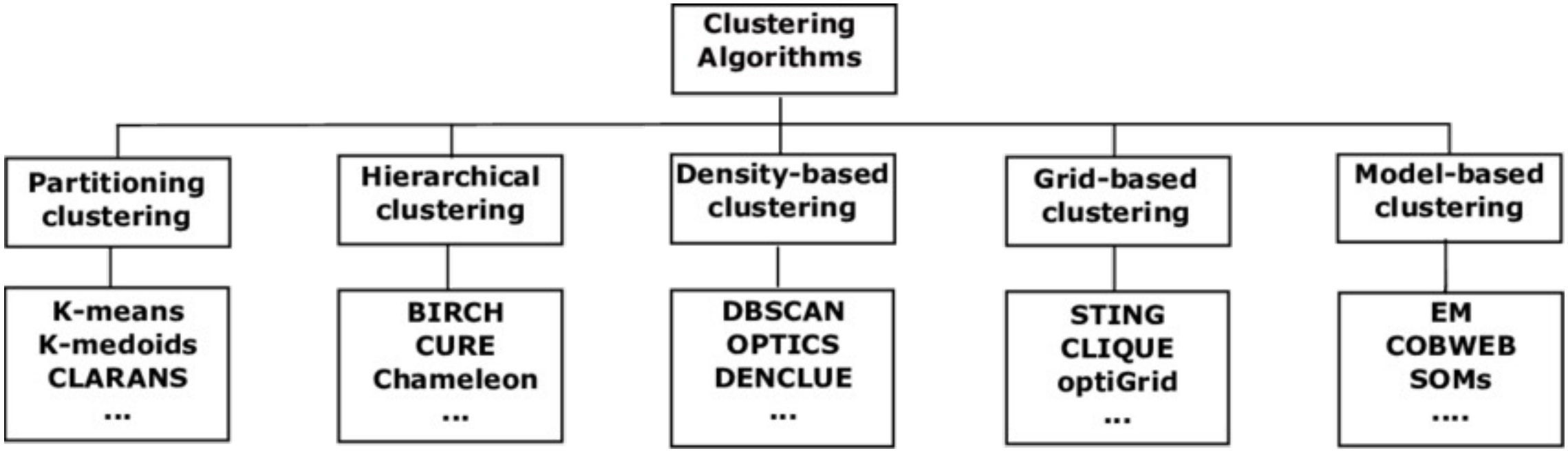
Normalisation crucial

Similarity metrics:
Euclidean, Manhattan, Cosine

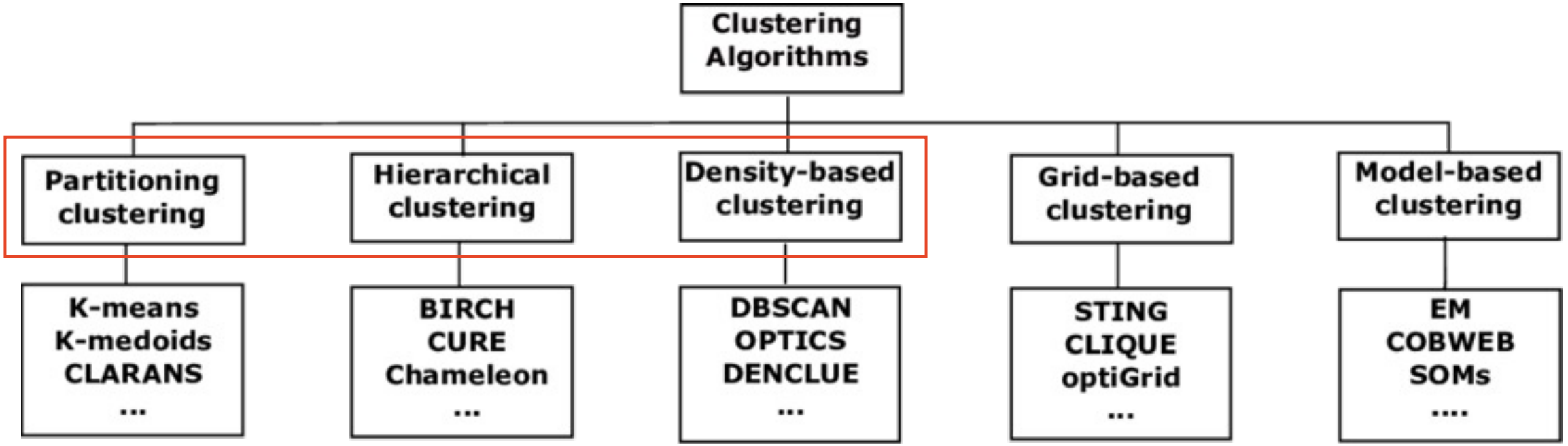


THE

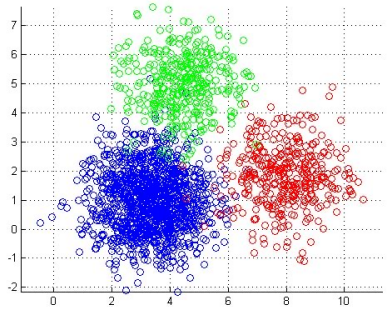
Types of Clustering

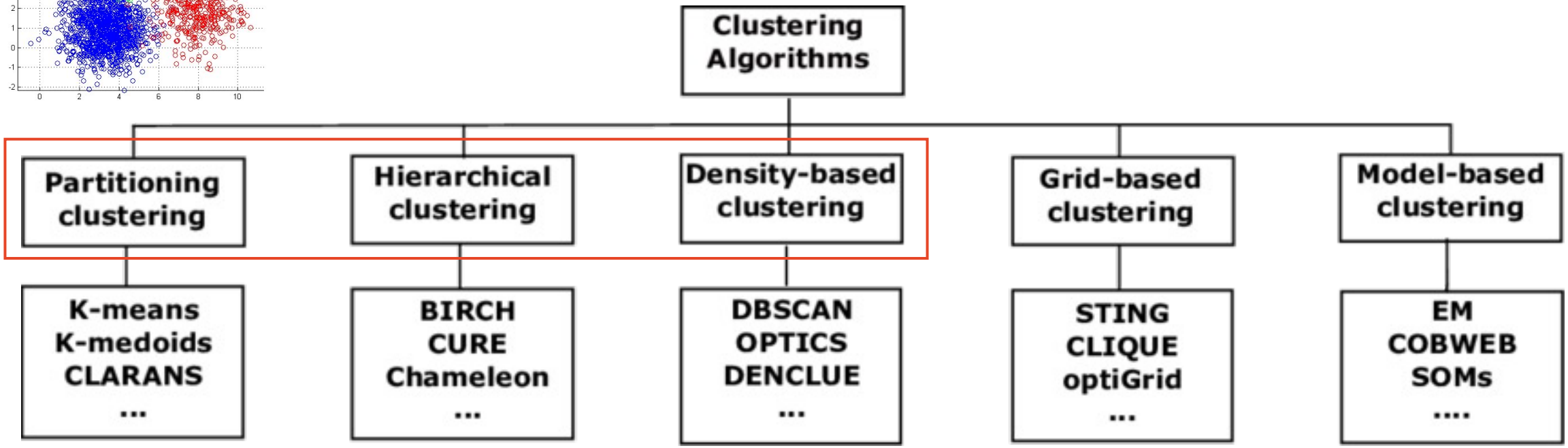


Types of Clustering



Types of Clustering



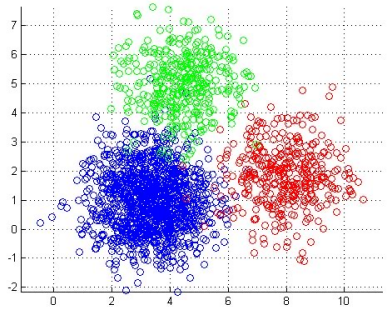


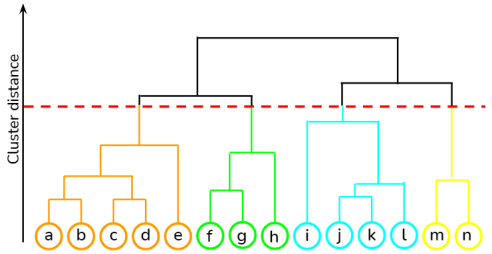
fix k clusters, iteratively update centroids until convergence

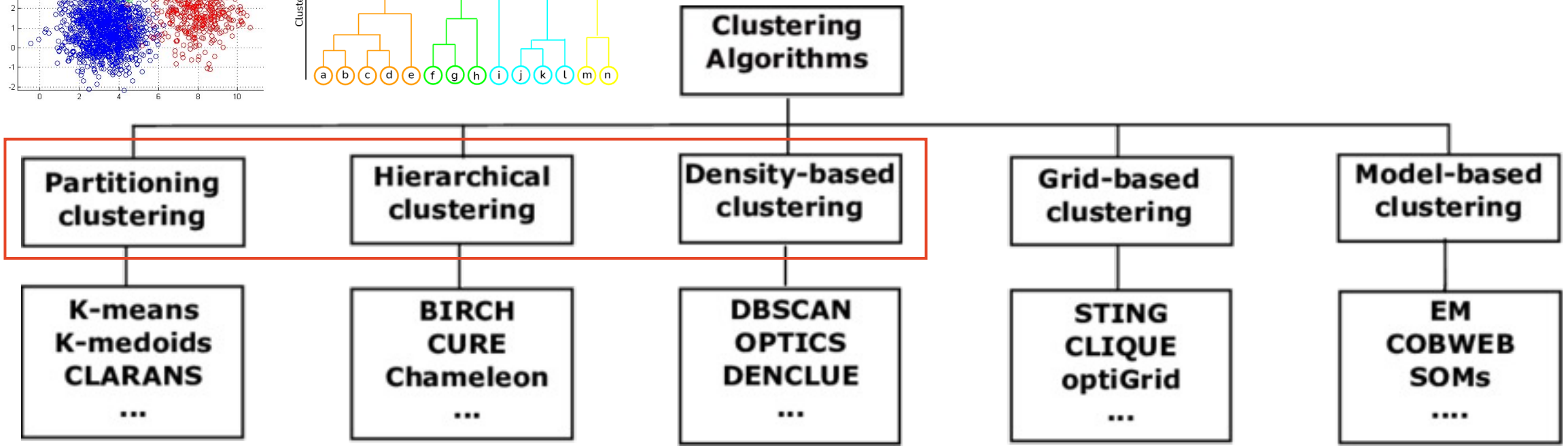
merge closest clusters step by step → tree structure

detect dense regions separated by sparse areas

Types of Clustering





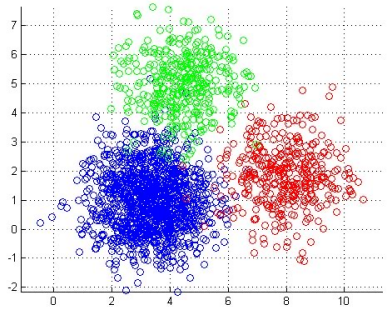


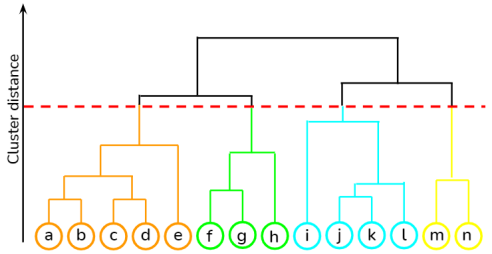
fix k clusters, iteratively update centroids until convergence

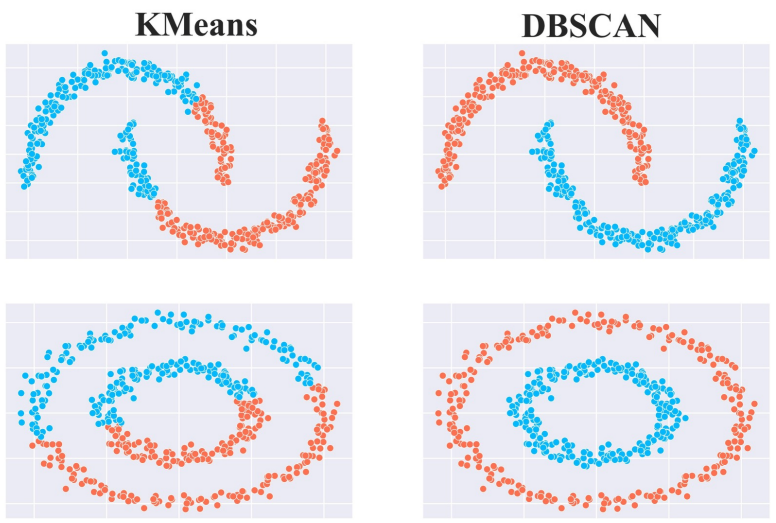
merge closest clusters step by step → tree structure

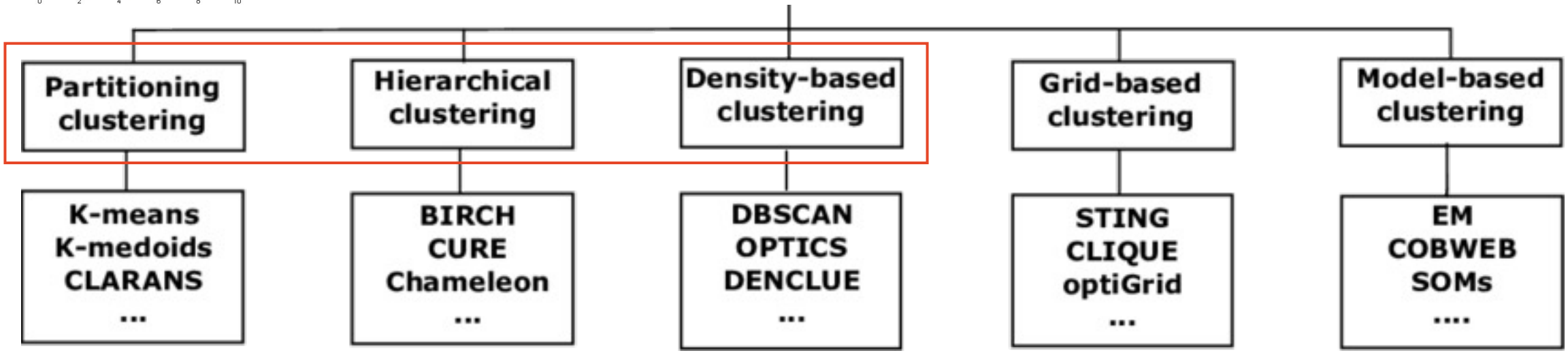
detect dense regions separated by sparse areas

Types of Clustering







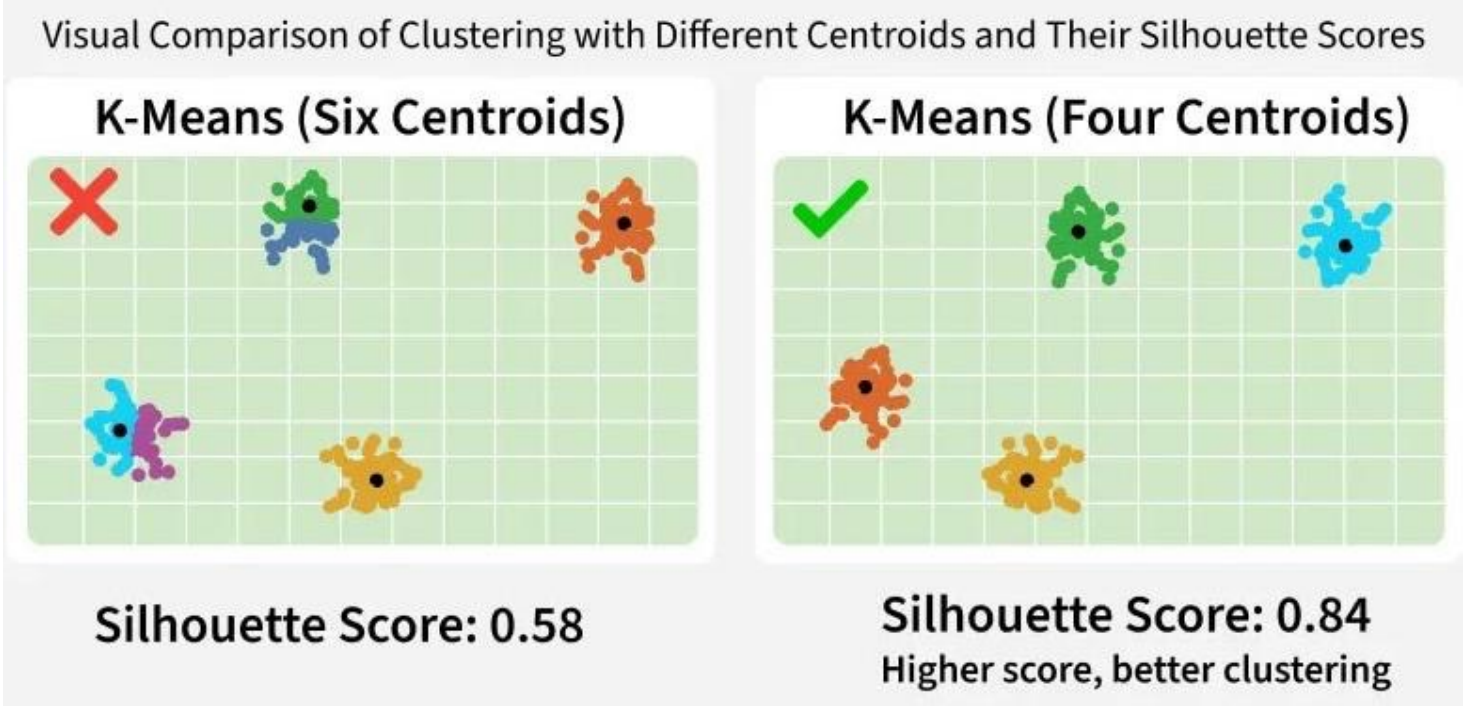


fix k clusters, iteratively update centroids until convergence

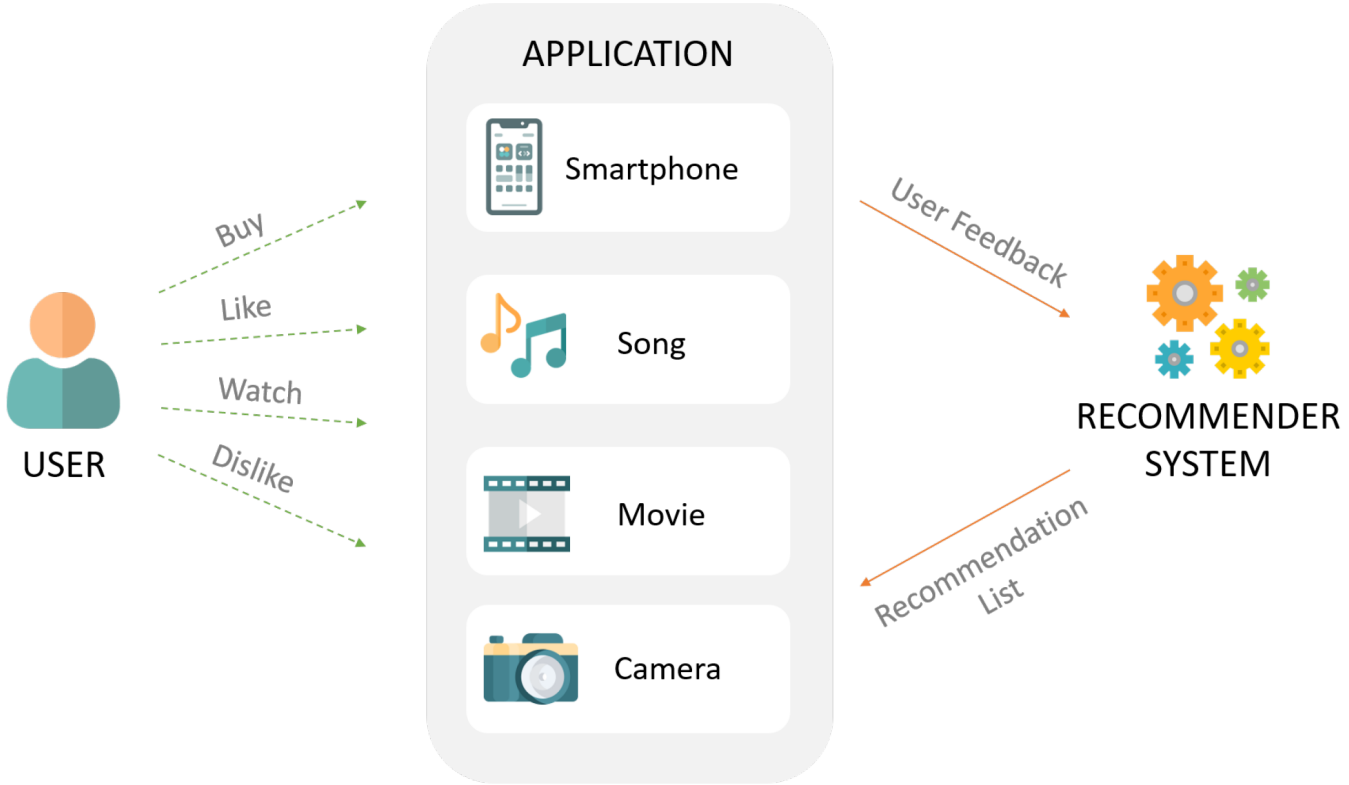
merge closest clusters step by step → tree structure

detect dense regions separated by sparse areas

Evaluation


Visual Comparison of Clustering with Different Centroids and Their Silhouette Scores
K-Means (Six Centroids)
K-Means (Four Centroids)
Silhouette Score: 0.58
Silhouette Score: 0.84
Higher score, better clustering

Recommender



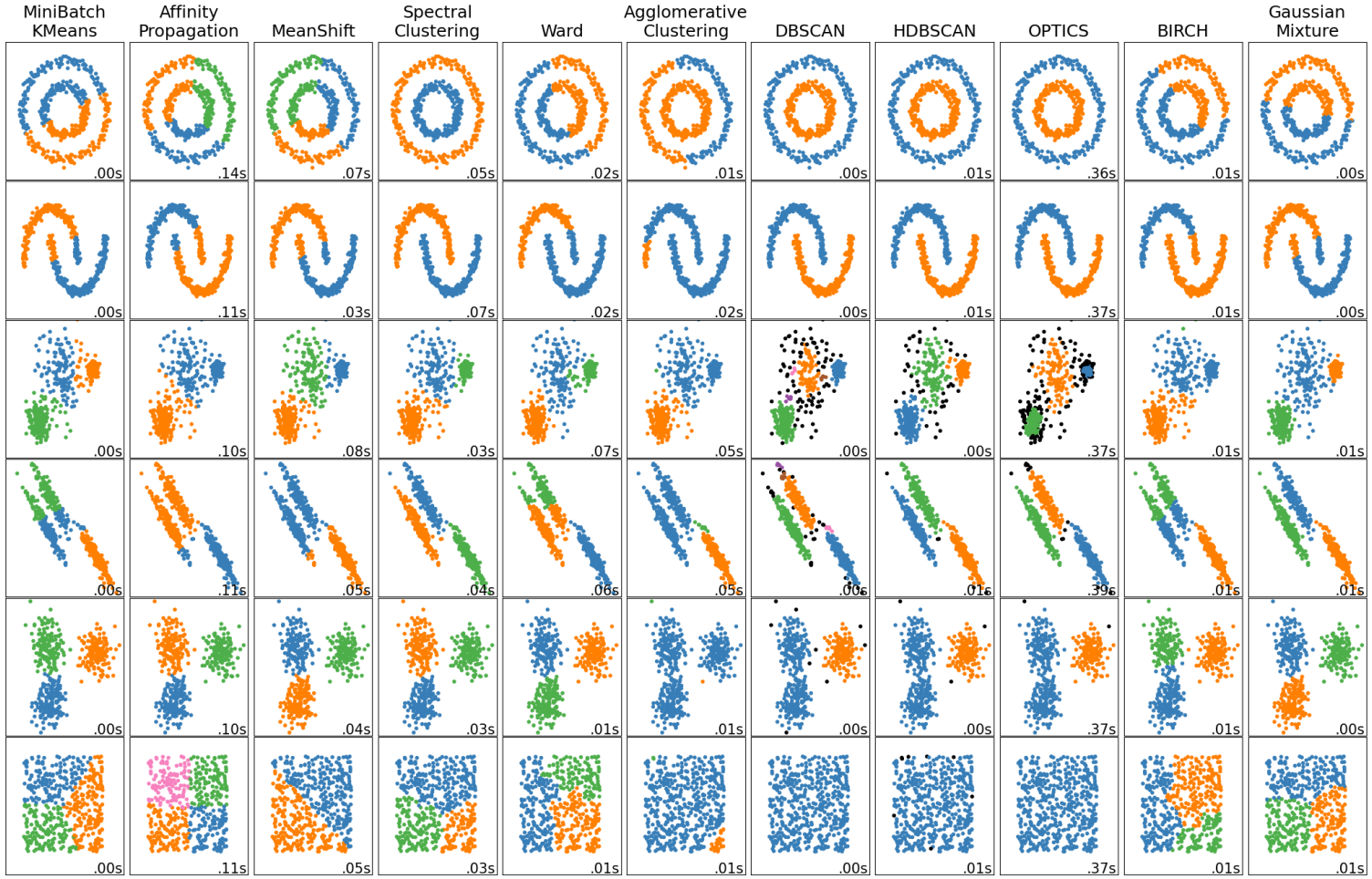
Python Demonstration

Clustering

Try running clustering algorithm

https://scikit-learn.org/stable/modules/clustering.html

See **cluster.py** on the Ed Workspace



Content

Regression

Regression Model

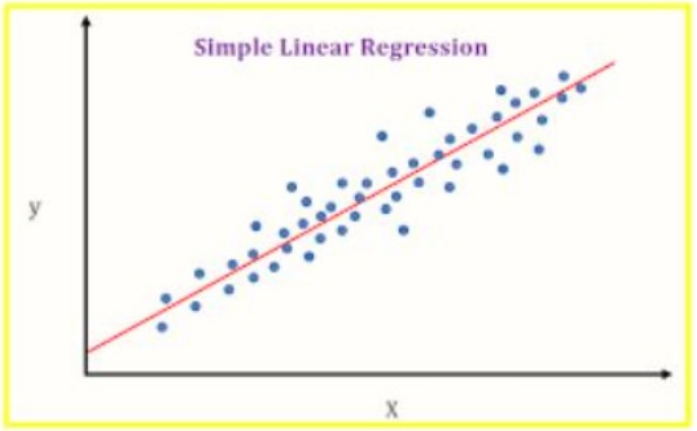
- It is a statistical technique used in machine learning, statistics, and data analysis to quantify the relationship between a dependent variable (also called the target) and one or more independent variables (predictors or features)
- Its primary purpose is to make predictions or estimate the value of the dependent variable based on the values of the independent variables

Types of Regression Model

- Linear Regression Model
- Polynomial Regression Model
- Multiple Regression Model
- Logistic Regression Model

Linear Regression Model

- Simple and effective way to model linear relationships between two variables
- Increase or decrease of dependent variable is proportional to the increase of the predictor – i.e., independent variable

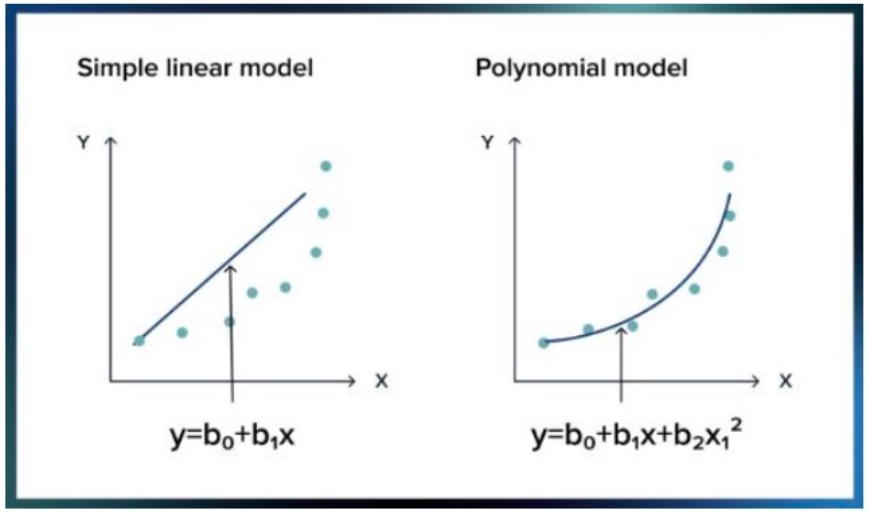


Linear Regression Model

- **Linearity:** It assumes that a linear relationship between predictors and the target variable
- **Least Squares Method:** The coefficient is estimated by minimising the sum of squared differences between predicted and actual values
- **Assumptions:** Linear regression assumes that residuals (differences between predicted and actual values) are normally distributed. The residuals have constant variance, and there is no correlation between predictors (multicollinearity)
- **Inference:** This regression model allows hypothesis testing on coefficients to determine their significance and assess the model's explanatory power

Polynomial Regression Model

- Models the relationship between a dependent variable and one or more independent variables using a polynomial function
  - Polynomial functions: functions in which the independent variable is raised to a power



Polynomial Regression Model

- It is an extension of the linear regression model by allowing for nonlinear relationships between predictors and the target variable
- It uses polynomial terms (e.g.,x^2, x^3) to capture curved patterns in the data.
- It aims to fit a polynomial equation to the data using the method of least squares, minimising the sum of squared differences between predicted and actual values.
- **Warning:** only do regression with powers of x_i, when you have good reason to expect a polynomial impact of the degree you are including
  - It is very easy to overfit training data with polynomial model

Which Model To Use?

- It depends on the specific problem you are trying to solve
  - If you are trying to model a linear relationship between variables, then linear regression is a good choice
  - If you are trying to model a non-linear relationship between variables, then polynomial regression or multiple regression may be a better choice
  - If you are trying to classify data, then logistic regression is a good choice
- All regression models are statistical models, and they have limitations
  - It is important to consider the assumptions of the model and the quality of the data before using the model to make predictions

Research Task

Regression

Regression Models

Pretend you're doing your assignment and look up all the different types of regression and try and get at least one running in Python

https://scikit-learn.org/stable/supervised_learning.html

Content

Regression vs Classification

What is classification?

Classification is a supervised learning task where the goal is to predict a discrete class label for a given input. The model learns from labeled training data to make predictions on new, unseen data.

Key aspects of classification:

Types of classification:

- Binary classification: Two possible classes (e.g., spam or not spam)
- Multi-class classification: More than two classes (e.g., classifying animals into species)
- Multi-label classification: Each instance can belong to multiple classes simultaneously

Examples of common algorithms:

- Logistic Regression: Despite its name, it's used for binary classification
- Decision Trees: Tree-like model of decisions
- Random Forests: Ensemble of decision trees

Output of these algorithms are **categorical**

Evaluation Metrics

Accuracy: Proportion of correct predictions

Precision: Proportion of true positive predictions among all positive predictions

- True Positives / (True Positives + False Positives)

Recall: Proportion of true positive predictions among all actual positive instances

- Recall = True Positives / (True Positives + False Negatives)

F1-score: Harmonic mean of precision and recall

- F1 = 2 * (Precision * Recall) / (Precision + Recall)

Regression vs. Classification

Output type:
- Classification produces categorical outputs
- Regression produces numerical outputs.

Evaluation metrics:
- Classification uses metrics like accuracy, precision, recall, and F1-score.
- Regression uses metrics like mean squared error (MSE), root mean squared error (RMSE), and R-squared.

Decision boundaries:
- Classification algorithms try to find decision boundaries between classes
- Regression algorithms aim to fit a line or curve to the data.

Lab Activities

Working on Assignment 2

Activity

The first main goal is to agree on the dataset and which attribute you plan to predict (each person using a different approach for producing a predictive model). First though, make sure that you have several ways to communicate with one another between labs.

In many cases, the dataset will be carried over from previous Stage, but even so, you need to consider which attribute you will be predicting. Notice that if you want to predict a nominal attribute, then you will all be using classification techniques; on the other hand, if you predict a quantitative attribute then you all must use various regression approaches.

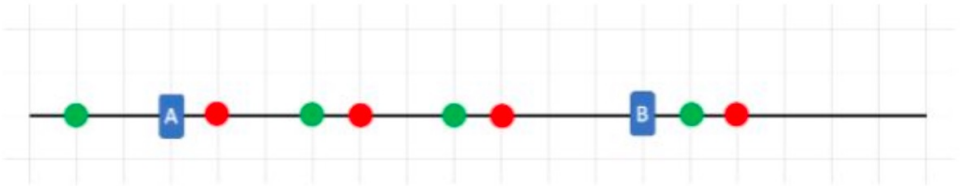
You should at this time divide the dataset into a training set, a validation set and a test set.

Now start choosing the ways you will each produce a predictive model. In the slides, we presented a few approaches for regression and classification (namely linear regression and kNN regression, and decision tree classifier and "logistic regression" classifier) but there are plenty of variants possible within these (for example, which distance function to use in kNN, how to encode any categorical attributes for linear regression, whether to transform any quantitative attributes by re- scaling them, which attributes to include in making the prediction, whether to apply regularization, how to tune any hyper-parameters, etc). You may also want to do some reading to find more ideas to try. But at least pick a few things to try initially.

Please make sure that you document your choice of dataset (including indication of where the original dataset, and the training and test subsets, are available), and which attribute to predict, by a post in the Canvas page of your group. Also post about the tasks people are given, and a due date for each task, to keep the group's progress going.

Exam-Style Questions

- What would A and B be classified as (green or red) under different values of k in a k-NN algorithm?
- What will be your default colour if there is a tie?



k	A	B
1		
2		
3		
4		

Exam-Style Questions

- Build a decision tree from the following data using Gini Impurity Index.

outlook	temp.	humidity	windy	play
sunny	hot	high	false	no
sunny	hot	high	true	no
overcast	hot	high	false	yes
rainy	mild	high	false	yes
rainy	cool	normal	false	yes
rainy	cool	normal	true	no
overcast	cool	normal	true	yes
sunny	mild	high	false	no
sunny	cool	normal	false	yes
rainy	mild	normal	false	yes
sunny	mild	normal	true	yes
overcast	mild	high	true	yes
overcast	hot	normal	false	yes
rainy	mild	high	true	no

That’s it folks!

Remaining Ed Lessons, Questions, Assignment etc.