Introduction to
Machine Learning Applications

Spring 2023

Exam 1 review

Minor Gordon

gordom6@rpi.edu

Rensselaer

Data preprocessing

Main steps of data preprocessing

- Aggregation
- Sampling
- Dimensionality reduction (future lecture)
- Feature subset selection
- Feature creation
- Discretization and binarization
- Attribute transformation

Aggregation

- Combining two or more attributes (or objects) into a single attribute (or object)
- Purpose
    - Data reduction
        - Reduce the number of attributes or objects
    - Change of scale
        - Cities aggregated into regions, states, countries, etc.
        - Days aggregated into weeks, months, or years
    - More “stable” data
        - Aggregated data tends to have less variability

Aggregation example

Date	Value
01/10/2020	10
01/27/2020	2
02/10/2020	4
02/19/2020	13
03/05/2020	19
03/21/2020	11
04/10/2020	15
04/16/2020	19
05/03/2020	8
05/18/2020	10
05/31/2020	7

Aggregate using sum (or any other metric that fits the problem)

Month	Value
January 2020	12
February 2020	17
March 2020	30
April 2020	34
May 2020	25

Sampling

- Sampling is the main technique employed for data reduction.
    - It is often used for both the preliminary investigation of the data and the final data analysis.
- Statisticians often sample because obtaining the entire set of data of interest is too expensive or time consuming.
- Sampling is typically used because processing the entire set of data of interest is too expensive or time consuming.

Sampling

- The key principle for effective sampling is the following:
  - Using a sample will work almost as well as using the entire data set, if the sample is representative
  - A sample is representative if it has approximately the same properties (of interest) as the original set of data

Types of sampling

- Simple random sampling
  - There is an equal probability of selecting any particular item
  - Sampling without replacement
    - As each item is selected, it is removed from the population
  - Sampling with replacement
    - Objects are not removed from the population as they are selected for the sample.
    - In sampling with replacement, the same object can be picked up more than once
- Stratified sampling
  - Split the data into several partitions; then draw random samples from each partition

Random sampling example

Date	Value
01/10/2020	10
01/27/2020	2
02/10/2020	4
02/19/2020	13
03/05/2020	19
03/21/2020	11
04/10/2020	15
04/16/2020	19
05/03/2020	8
05/18/2020	10
05/31/2020	7

Random sampling (n=3) →

Date	Value
02/10/2020	4
05/18/2020	10
01/10/2020	10
04/16/2020	19
05/03/2020	8

Stratified sampling example

Date	Value
01/10/2020	10
01/27/2020	2
02/10/2020	4
02/19/2020	13
03/05/2020	19
03/21/2020	11
04/10/2020	15
04/16/2020	19
05/03/2020	8
05/18/2020	10
05/31/2020	7

Bin-based sampling →

Date	Value
01/10/2020	10
02/19/2020	13
03/21/2020	11
04/16/2020	19
05/03/2020	8

Feature subset selection

- Another way to reduce dimensionality of data
- Redundant features
    - Duplicate much or all of the information contained in one or more other attributes
    - Example: purchase price of a product and the amount of sales tax paid
- Irrelevant features
    - Contain no information that is useful for the task at hand
    - Example: students' ID is often irrelevant to the task of predicting students' GPA
- Many techniques developed, especially for classification

Feature creation

- Create new attributes that can capture the important information in a data set much more efficiently than the original attributes

- Three general methodologies:
  - Feature extraction
    - Example: extracting edges from images
  - Feature construction
    - Example: dividing mass by volume to get density
  - Mapping data to new space
    - Example: Fourier and wavelet analysis

Discretization

- Discretization is the process of converting a continuous attribute into an ordinal attribute
  - A potentially infinite number of values are mapped into a small number of categories
  - Discretization is commonly used in classification
  - Many classification algorithms work best if both the independent and dependent variables have only a few values

Discretization example

Date	Value
01/10/2020	1.354
01/27/2020	1.83
02/10/2020	2.63
02/19/2020	9.242
03/05/2020	6.43
03/21/2020	9.23
04/10/2020	1.32
04/16/2020	1.756
05/03/2020	0.344
05/18/2020	3.33
05/31/2020	5.014

Assuming the range of value is [0,10) continuous

Assume [0,6): label1
[6,10): label2

Date	Value
01/10/2020	Label1
01/27/2020	Label1
02/10/2020	Label1
02/19/2020	Label2
03/05/2020	Label2
03/21/2020	Label2
04/10/2020	Label1
04/16/2020	Label1
05/03/2020	Label1
05/18/2020	Label1
05/31/2020	Label2

Binarization

- Binarization maps a continuous or categorical attribute into one or more binary variables

- Often convert a continuous attribute to a categorical attribute and then convert a categorical attribute to a set of binary attributes
    - Association analysis needs asymmetric binary attributes
    - Examples: eye color and height measured as {low, medium, high}

Binarization example

Date	Value
01/10/2020	Label1
01/27/2020	Label1
02/10/2020	Label3
02/19/2020	Label2
03/05/2020	Label2
03/21/2020	Label2
04/10/2020	Label1
04/16/2020	Label3
05/03/2020	Label1
05/18/2020	Label3
05/31/2020	Label2

Assuming 0 – {label1, label2}; 1– {label3}

Date	Value
01/10/2020	0
01/27/2020	0
02/10/2020	1
02/19/2020	0
03/05/2020	0
03/21/2020	0
04/10/2020	0
04/16/2020	1
05/03/2020	0
05/18/2020	1
05/31/2020	0

Attribute transformation

- An attribute transform is a function that maps the entire set of values of a given attribute to a new set of replacement values such that each old value can be identified with one of the new values
  - Simple functions: x^k, $\log(x)$, e^x, $|x|$
  - Normalization
    - Refers to various techniques to adjust to differences among attributes in terms of frequency of occurrence, mean, variance, range
    - Take out unwanted, common signal, e.g., seasonality
  - In statistics, standardization refers to subtracting off the means and dividing by the standard deviation

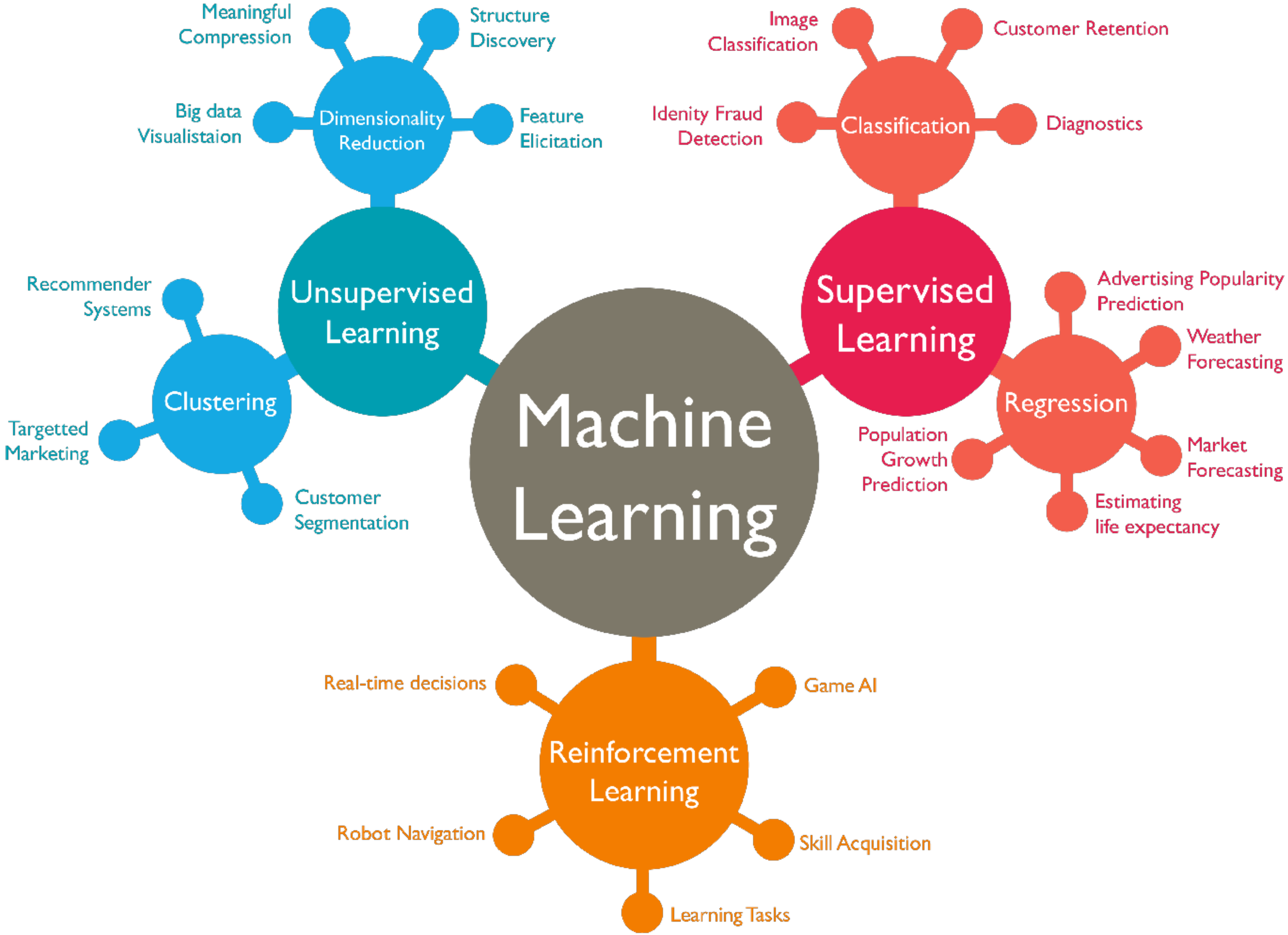
Attribute transformation using normalization

Original data = [0.5, 1.0, 0.5]

Computation = [0.5/(0.5+1.0+0.5),1.0/(0.5+1.0+0.5),0.5/(0.5+1.0+0.5)]
= [0.5/2.0, 1.0/2.0, 0.5/2.0]

Normalized data = [0.25, 0.5, 0.25] – sum of the list is 1.

Machine Learning overview


Machine Learning
Unsupervised Learning
Dimensionality Reduction
Meaningful Compression
Structure Discovery
Big data Visualistaion
Feature Elicitation
Clustering
Recommender Systems
Targetted Marketing
Customer Segmentation
Supervised Learning
Classification
Image Classification
Customer Retention
Idenity Fraud Detection
Diagnostics
Regression
Advertising Popularity Prediction
Weather Forecasting
Population Growth Prediction
Market Forecasting
Estimating life expectancy
Reinforcement Learning
Real-time decisions
Game AI
Robot Navigation
Skill Acquisition
Learning Tasks

Supervised Learning

- Prediction with focused target variable
- Training data provided
- Example:
  - Iris Example
  - Titanic Example
  - Housing prices
  - Nearly every Kaggle (there are some exploratory visualization tasks that wouldn’t be supervised)

Supervised Learning

- Must have a defined problem, dataset, ideal solution

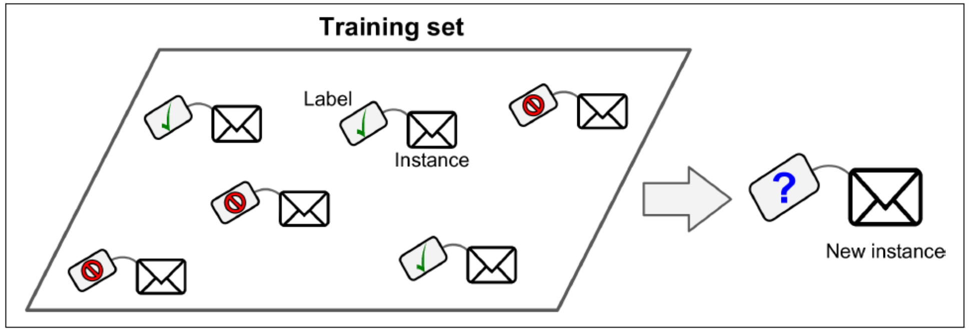


Figure 1-5. A labeled training set for supervised learning (e.g., spam classification)

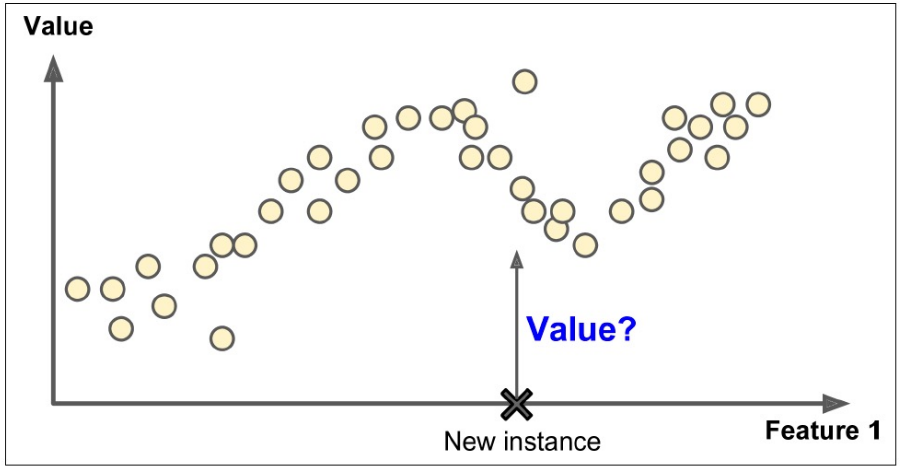


Figure 1-6. Regression

Can you predict species from characteristics given a training set?

Iris setosa

Iris versicolor

Iris virginica

https://en.wikipedia.org/wiki/Iris_flower_data_set

“**Unsupervised machine learning** is the machine learning task of inferring a function to describe hidden structure from "unlabeled" data (a classification or categorization is not included in the observations). Since the examples given to the learner are unlabeled, there is no evaluation of the accuracy of the structure that is output by the relevant.”

- Wikipedia

How can we do an analysis if we don’t know the dependent variables?

Unsupervised Learning

- Finding hidden structures in unlabeled data
- No target dependent variable is provided
- Example: Clustering
  - K-Means
  - DBSCAN
  - Hierarchical Cluster Analysis
- Anomaly detection
  - Once class SVM
  - Isolation Forest

Unsupervised Learning

- Visualization and dimensionality reduction
  - Principal Component Analysis (PCA)
  - t-distributed stochastic neighbor embedding (t-SNE)

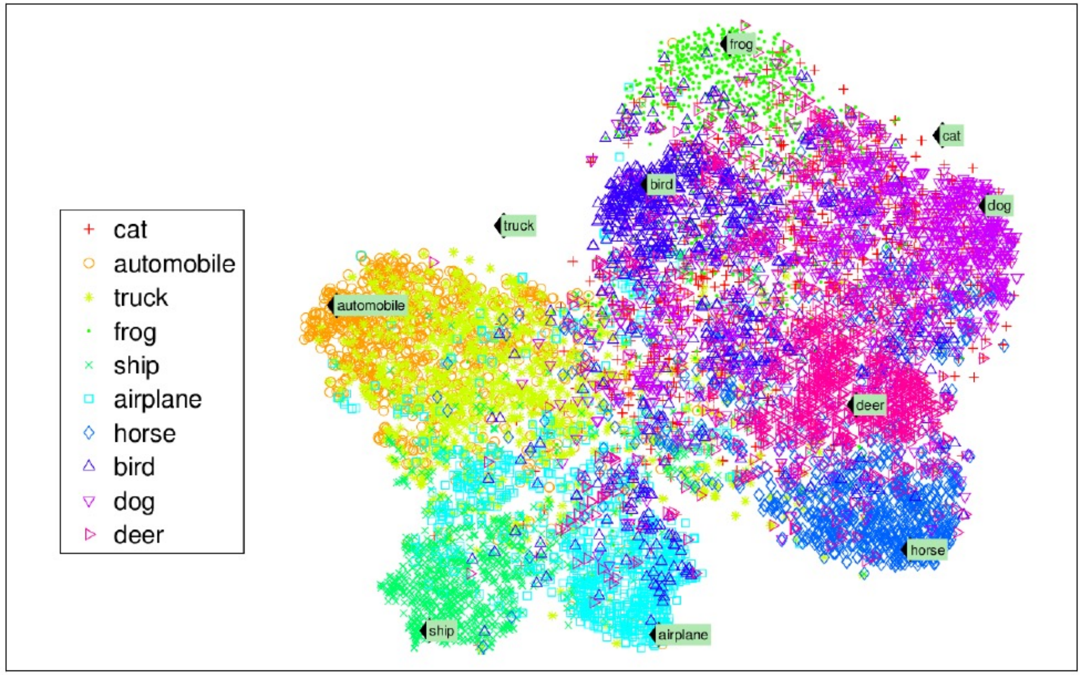
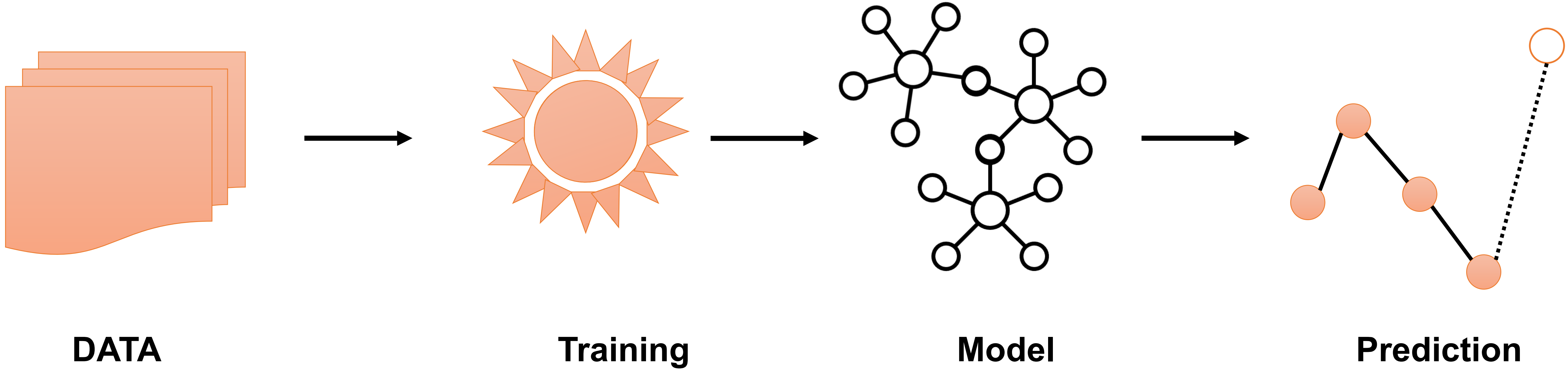


Figure 1-9. Example of a t-SNE visualization highlighting semantic clusters[3]

Model overview

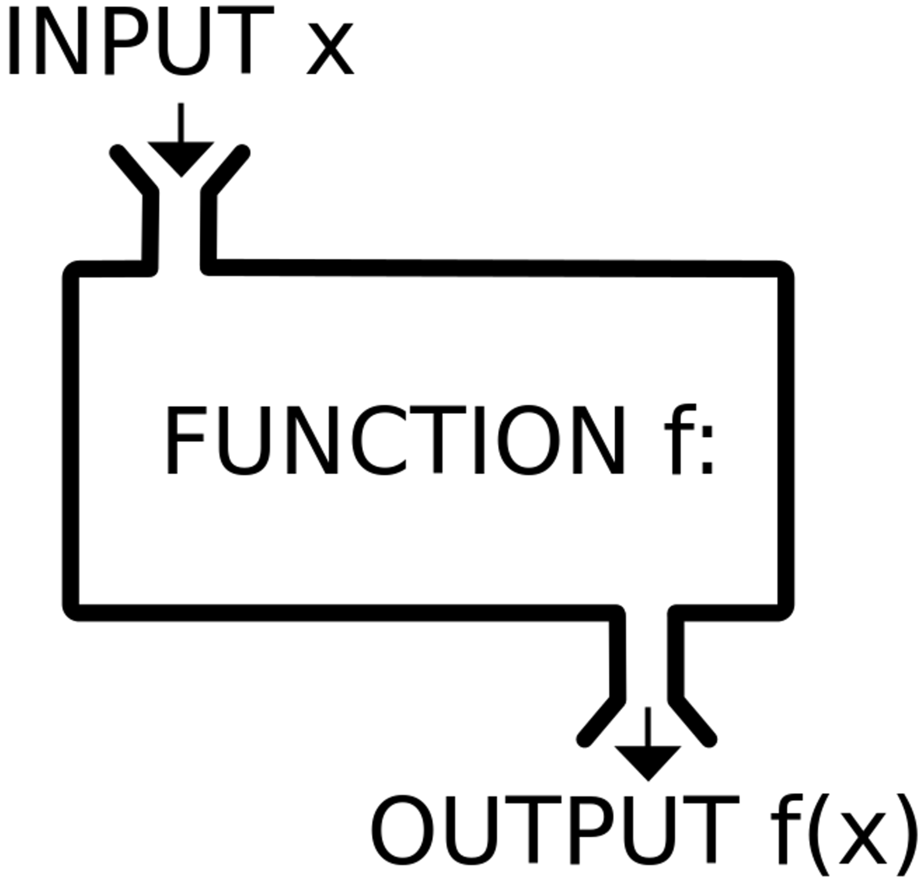
“A model is a simplified representation of reality created to serve a purpose.” - Provost & Fawcett

A standard learning pipeline



DATA → Training → Model → Prediction

Independent or Explanatory Variables



INPUT x

FUNCTION f:

OUTPUT f(x)

Target or Dependent Variable

Model evaluation

Evaluating classification

CONFUSION MATRIX

“confusion matrix, also known as an error matrix, is a specific table layout that allows visualization of the performance of an algorithm, typically a supervised learning one.”

https://en.wikipedia.org/wiki/Confusion_matrix

	Predicted Class		
		True	False
Actual Class	True	True positive (tp)	False Negative (fn)
	False	False Positive (fp)	True Negative (tn)

Confusion Matrix

- A table that is often used to describe the performance of a classification model on a set of test data.
- This allows the visualization of the algorithm’s performance.

		Actual Class	
		Class = 1	**Class = 0**
Predicted Class	**Class = 1**	f_{11}	f_{10}
	Class = 0	f_{01}	f_{00}

Accuracy

Accuracy = (tp + tn) / (P+N)
= (tp + tn)/(tp + fp + tn + fn)

Both classes interesting and not severely unbalanced (why?).

	Predicted Class		
		True	False
Actual Class	True	True positive (tp)	False Negative (fn)
	False	False Positive (fp)	True Negative (tn)

Sensitivity/Recall/Hit Rate/True Positive Rate (TPR)

Sensitivity/Recall = tp/P
= (tp)/(tp + fn)

(Positive class more interesting.)

	Predicted Class		
		True	False
Actual Class	True	True positive (tp)	False Negative (fn)
	False	False Positive (fp)	True Negative (tn)

Recall

How many relevant items are selected?

$$Recall = \frac{|\{relevant\ documents\} \cap \{retrieved\ documents\}|}{|\{relevant\ documents\}|}$$

$$Recall = \frac{f_{11}}{(f_{01} + f_{11})}$$

Precision

Precision = tp/(tp + fp)
(Actual positive class more interesting and higher costs of false positives)

	Predicted Class		
		True	False
Actual Class	True	True positive (tp)	False Negative (fn)
	False	False Positive (fp)	True Negative (tn)

Precision

How many selected items are relevant?

$$Precision = \frac{|\{relevant\ documents\} \cap \{retrieved\ documents\}|}{|\{retrieved\ documents\}|}$$

$$Precision = \frac{f_{11}}{(f_{10} + f_{11})}$$

		Actual Class	
		Class = 1	Class = 0
Predicted Class	Class = 1	f_{11}	f_{10}
	Class = 0	f_{01}	f_{00}

- f_{11} – True Positive
- f_{10} – False Positive – Type I error
- f_{01} – False Negative – Type II error
- f_{00} – True Negative

Precision: How many selected items are relevant?

Recall: How many relevant items are selected?

F-measure

Better measure that considers the harmonic mean of *precision* and *recall*

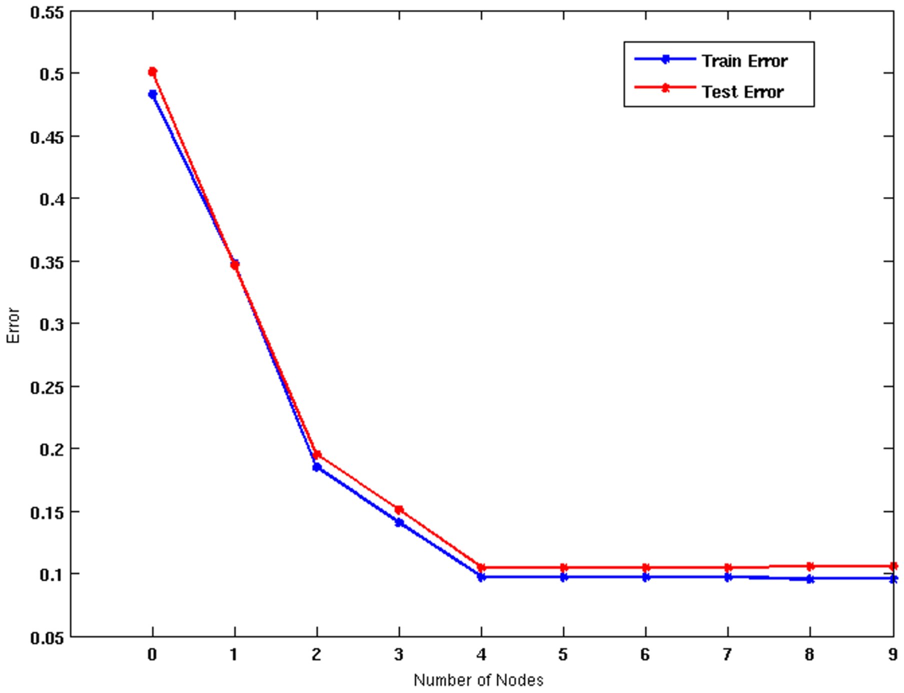
$$f - measure = \frac{2*(precision*recall)}{(precision+recall)}$$

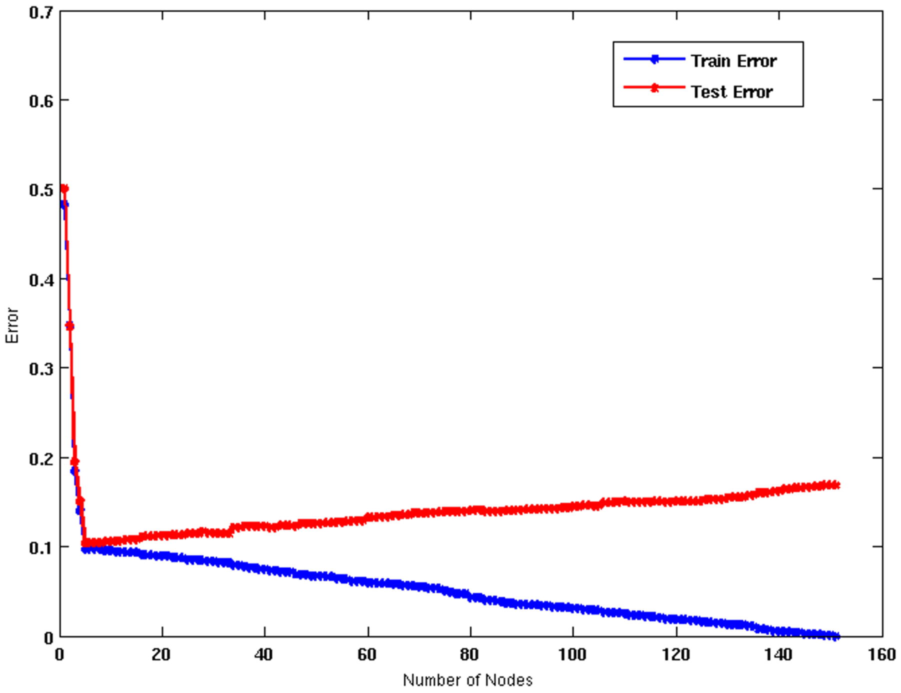
$$f1score = \frac{2 * precision * recall}{(precision + recall)}$$

Model training

Model training should result in a model that can adequately generalize to new data.

Model Overfitting & Underfitting



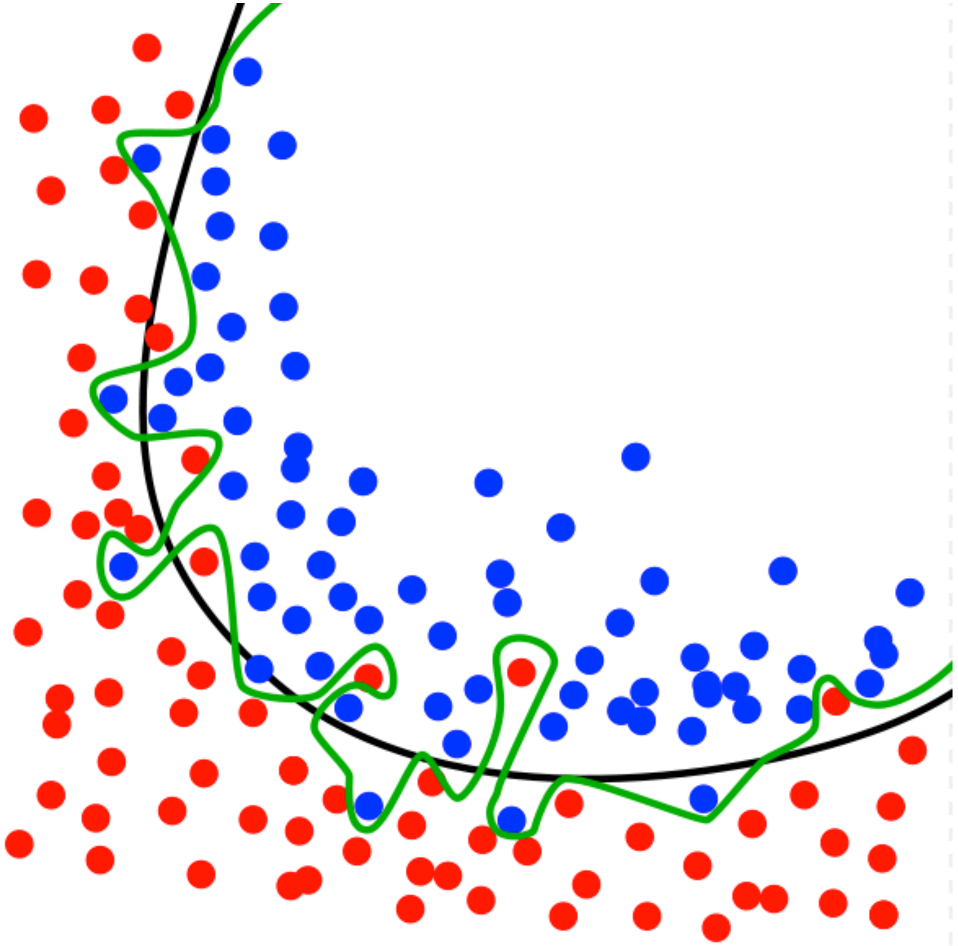


Underfitting: when model is too simple, both training and test errors are large

Overfitting: when model is too complex, training error is small but test error is large

Overfitting Training Set

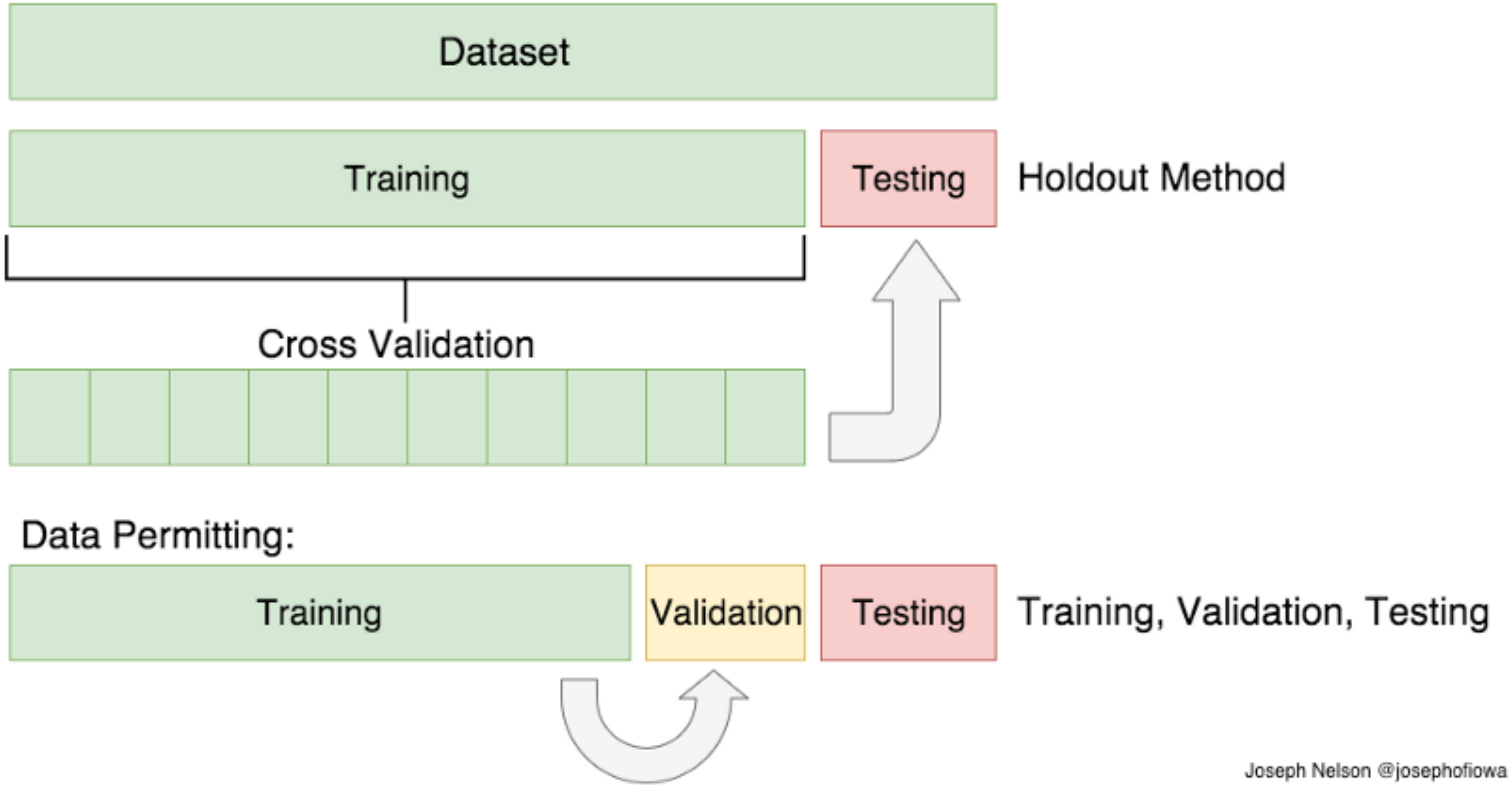
- This model with have a high level of accuracy but will not generalize to other data



By Chabacano (Own work) [GFDL (http://www.gnu.org/copyleft/fdl.html) or CC BY-SA 4.0-3.0-2.5-2.0-1.0 (https://creativecommons.org/licenses/by-sa/4.0-3.0-2.5

How do we prepare data to train a model and prevent over fitting?

Separation Into Train/Test



https://medium.com/towards-data-science/train-test-split-and-cross-validation-in-python-80b61beca4b6

Cross Validation

- Used to prevent overfitting of model and/or improving fit

k-fold Cross-validation

> Shuffle the dataset (better)

> Split the dataset into *k* disjoint groups

> For each unique group:

> Take the group as a hold out or test (validation) data set

> Take the remaining groups as a training data set

> Fit a model on the training set and evaluate it on the test set

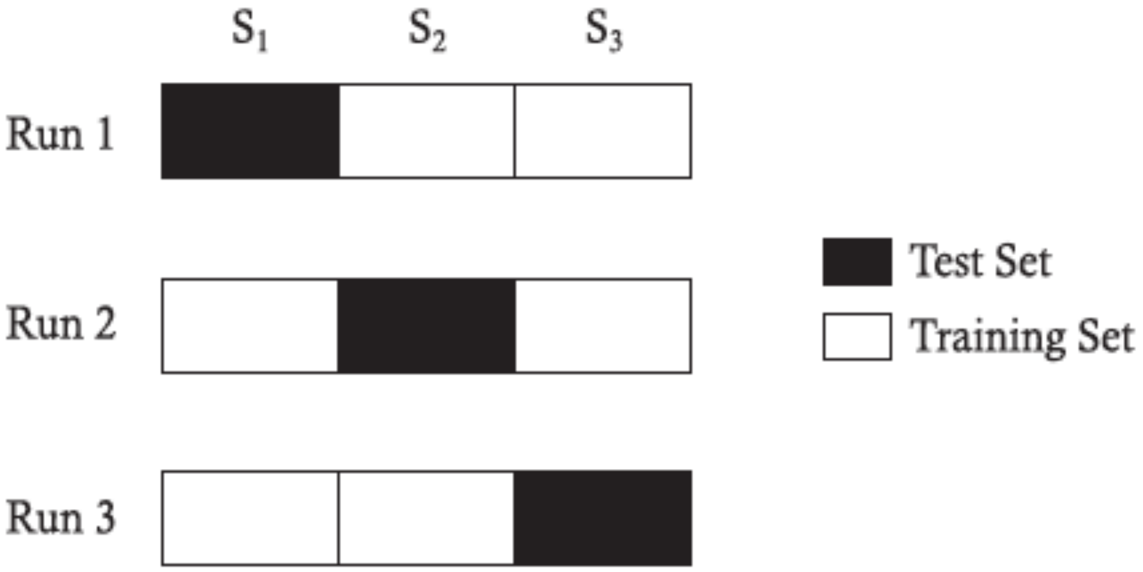
> Record the evaluation score

> Find the mean of all the sample of model evaluation scores

k-fold Cross-validation

[1, 2, 3, 4, 5, 6]

Fold1: [5, 3]
Fold2: [1, 6]
Fold3: [2,4]

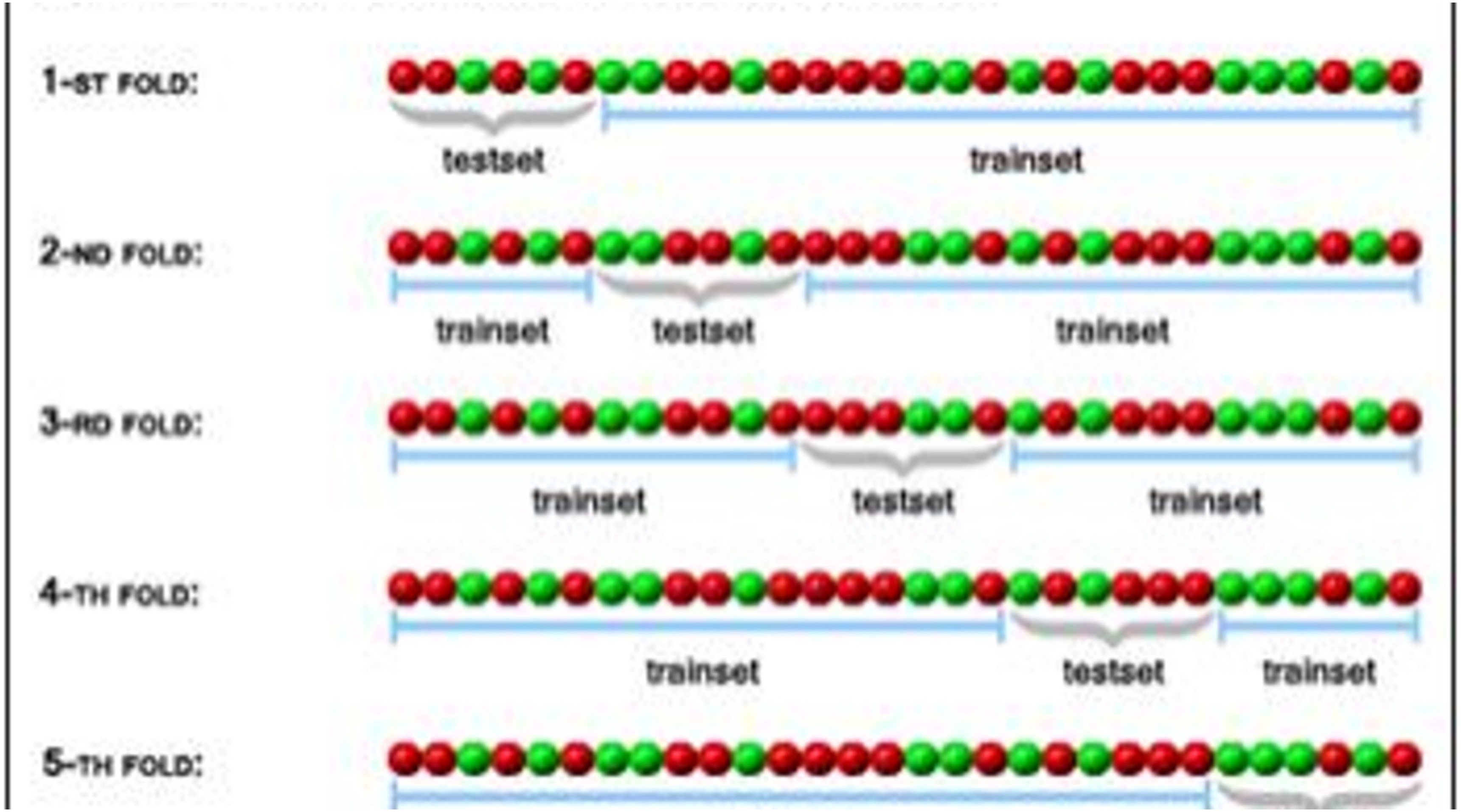


Model1: Trained on Fold2 + Fold3, Tested on Fold1
Model2: Trained on Fold1 + Fold3, Tested on Fold2
Model3: Trained on Fold1 + Fold2, Tested on Fold3

Example

- Given a set of data points – {a, b, c, d, e, f, g, h}
  - Perform 4-fold cross validation
  - Explain it in your own terms – what are the folds and how do you use them?

5 Fold Cross Validation



Regression

Linear Regression

The technique is used to **predict** the value of one variable (the dependent variable - y) **based on** the value of other variables (independent variables x_1, x_2,…x_k) where $\mathcal{E}$ is the error.

Logistic Regression

- Special case of linear regression where the target variable is categorical in nature