HAIMLC701 AI & ML in Healthcare

2.0		**AI, ML, Deep Learning and Data Mining Methods for Healthcare**	10
	2.1	Knowledge discovery and Data Mining, ML, Multi classifier Decision Fusion, Ensemble Learning, Meta-Learning and other Abstract Methods.	
	2.2	Evolutionary Algorithms, Illustrative Medical Application-Multiagent Infectious Disease Propagation and Outbreak Prediction, Automated Amblyopia Screening System etc.	
	2.3	Computational Intelligence Techniques, Deep Learning, Unsupervised learning, dimensionality reduction algorithms.	

CO Mapped: CO2-Apply advanced AI and Computational Intelligence techniques for Healthcare Problems (L3)

Textbook to refer:

1. Arjun Panesar, "Machine Learning and AI for Healthcare", A Press.
2. Arvin Agah, "Medical applications of Artificial Systems ", CRC Press

Ensemble Learning

- A composite model, combines a series of low performing classifiers with the aim of creating an improved classifier
- Here, individual classifier vote and final prediction label returned that performs majority voting
- Offer more accuracy than individual or base classifier
- Can be parallelized by allocating each base learner to different-different machines
- Decrease variance using bagging
    - bias using a boosting and improve predictions using stacking

Ensemble Learning

- Use multiple learning algorithms (classifiers)
- Combine the decisions
- Can be more accurate than the individual classifiers
- Generate a group of base-learners
- Different learners use different
  - Algorithms
  - Hyperparameters
  - Representations (Modalities)
  - Training sets

Advantages

- Performance:
  - better predictions and achieve better performance than any single contributing model
- Robustness:
  - reduces the spread or dispersion of the predictions and model performance

Model Error

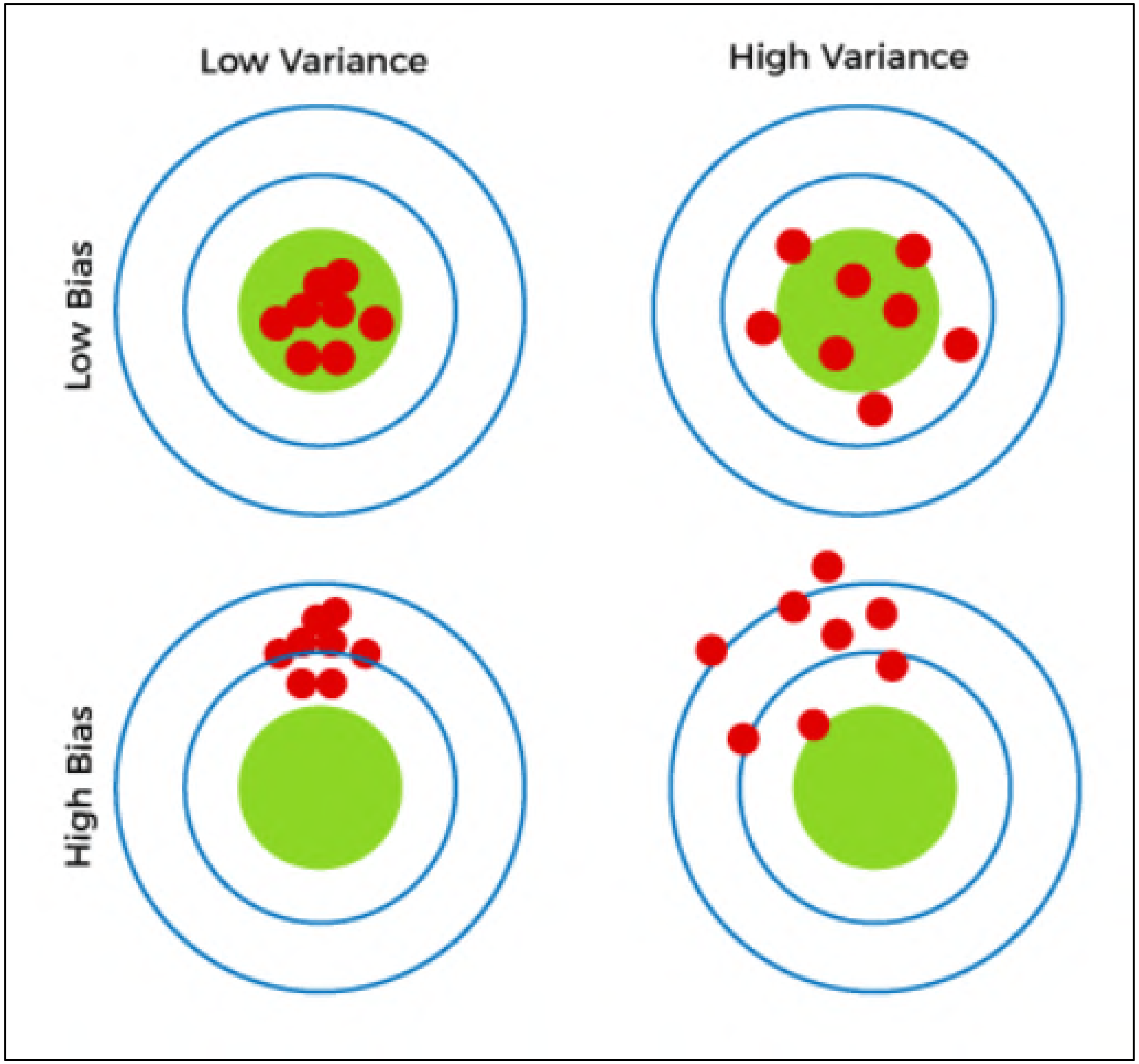
- The error emerging from any machine model

$$Err(x) = \left(E[\hat{f}(x)] - f(x)\right)^2 + E\left[\hat{f}(x) - E[\hat{f}(x)]\right]^2 + \sigma_e^2$$

$$Err(x) = \mathrm{Bias}^2 + \mathrm{Variance} + \mathrm{Irreducible\ Error}$$

- Bias:
  - Assumptions made by a model to make a function easier to learn/ error rate of the training data
  - When the error rate has a high value- High Bias(underfit)
- Variance:
  - The difference between the error rate of training data and testing data
  - If the difference is high- high variance(overfit)

Bias and Variance



Ensemble Learning

- Linear algorithm -high bias, as it makes them learn fast
  - **Linear Regression, Linear Discriminant Analysis and Logistic Regression.**
- Low bias **are Decision Trees, k-Nearest Neighbours and Support Vector Machines**

Ways to reduce High Bias:

- Increase the input features as the model is underfitted
- Decrease the regularization term
- Use more complex models, such as including some polynomial features(feature engineering)

- Ways to Reduce High Variance:

- Reduce the input features or number of parameters as a model is overfitted
- Do not use a much complex model
- Increase the training data
- Increase the Regularization term

Regularization

- Form of regression, that regularizes or shrinks the coefficient estimates towards zero
- ***Discourages learning a more complex or flexible model, so as to avoid the risk of overfitting***
- The fitting procedure involves a loss function, known as residual sum of squares or RSS
- The coefficients are chosen, such that they minimize this loss function
- y= β0+β1x1+β2x2+β3x3+⋯+βnxn +b
- X1, X2, …Xn are the features for Y.
- β0,β1,…..βn are the weights or magnitude attached to the features, respectively. Here β0 represents the bias of the model, and b represents the intercept

$$\sum_{i=1}^{M}\left(y_i - y'_i\right)^2 = \sum_{i=1}^{M}\left(y_i - \sum_{j=0}^{n}\beta_j * Xij\right)^2$$

Ensemble Learning Methods

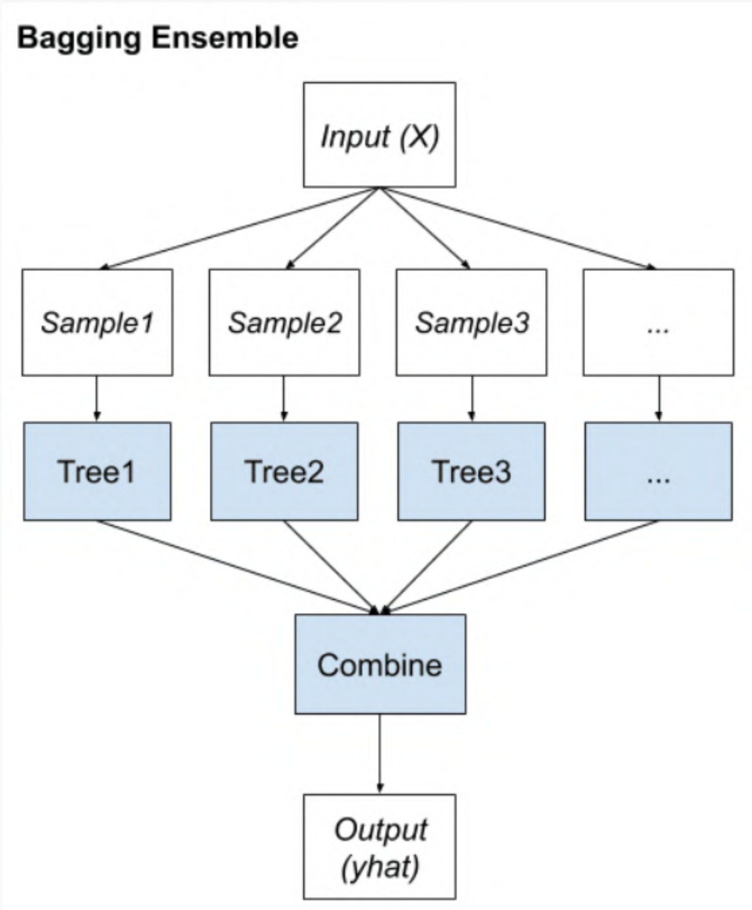
- **Bagging**

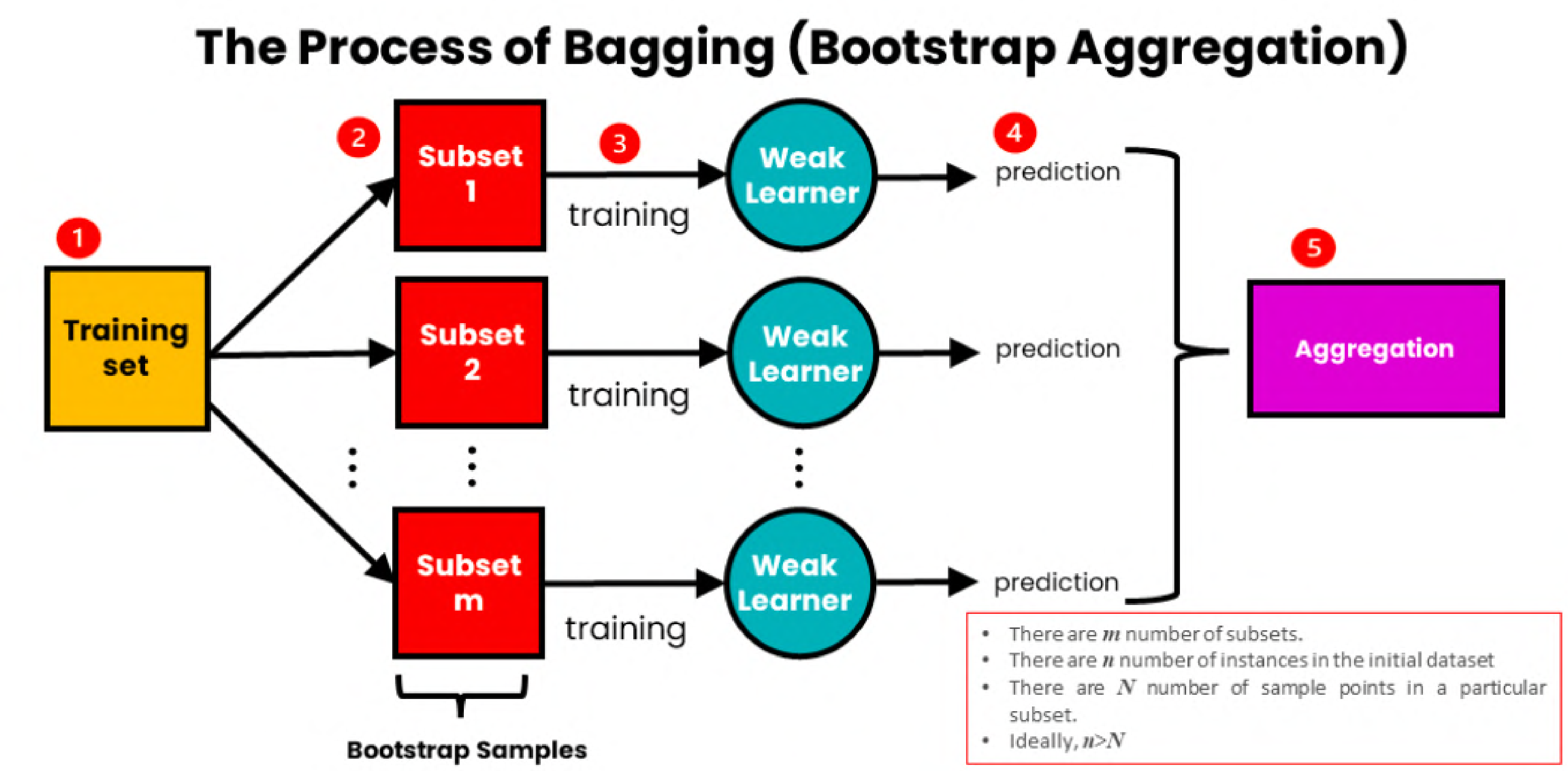
1. **Data Sampling**: Creating multiple subsets of the training dataset using bootstrap sampling (random sampling with replacement).
2. **Model Training**: raining a separate model on each subset of the data.
3. **Aggregation**: Combining the predictions from all individual models (averaged for regression or majority voting for classification) to produce the final output

- **Key Benefits**:

1. **Reduces Variance**: By averaging multiple predictions, bagging reduces the variance of the model and helps prevent overfitting.
2. **Improves Accuracy**: Combining multiple models usually leads to better performance than individual models.

- Examples:
  - Bagged Decision Trees (canonical bagging)
  - Random Forest
  - Extra Trees

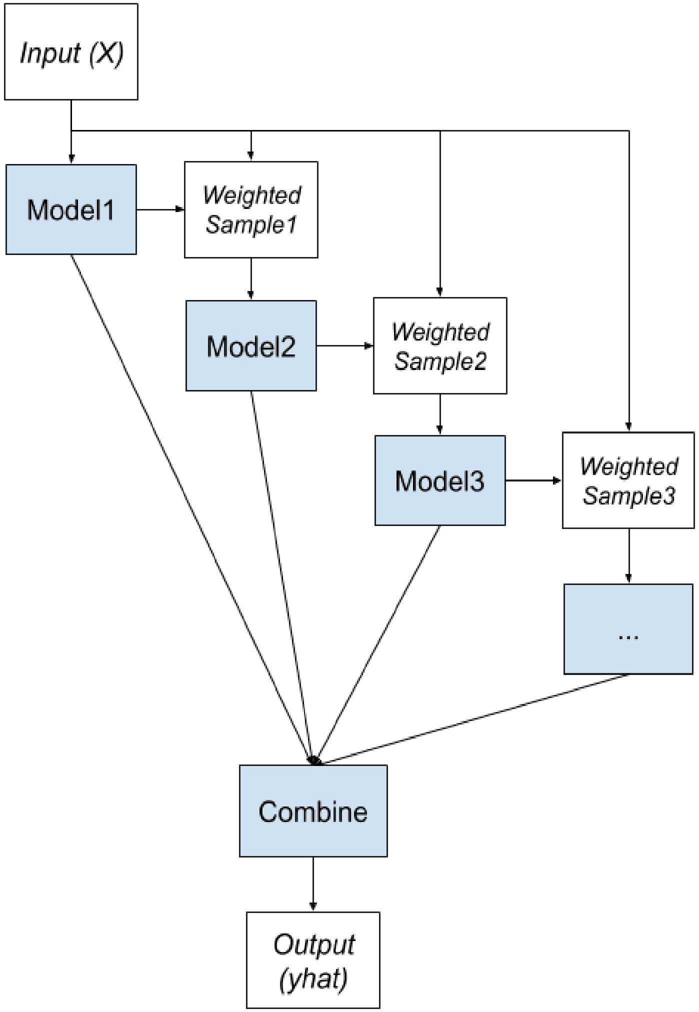



The Process of Bagging (Bootstrap Aggregation)
1
Training set
2
Subset 1
Subset 2
Subset m
Bootstrap Samples
3
training
training
training
Weak Learner
Weak Learner
Weak Learner
4
prediction
prediction
prediction
5
Aggregation
There are m number of subsets.
There are n number of instances in the initial dataset
There are N number of sample points in a particular subset.
Ideally, $n>N$

Ensemble Learning Methods

- Boosting
- another ensemble learning technique that focuses on creating a strong model by combining several weak models. It involves the following steps:

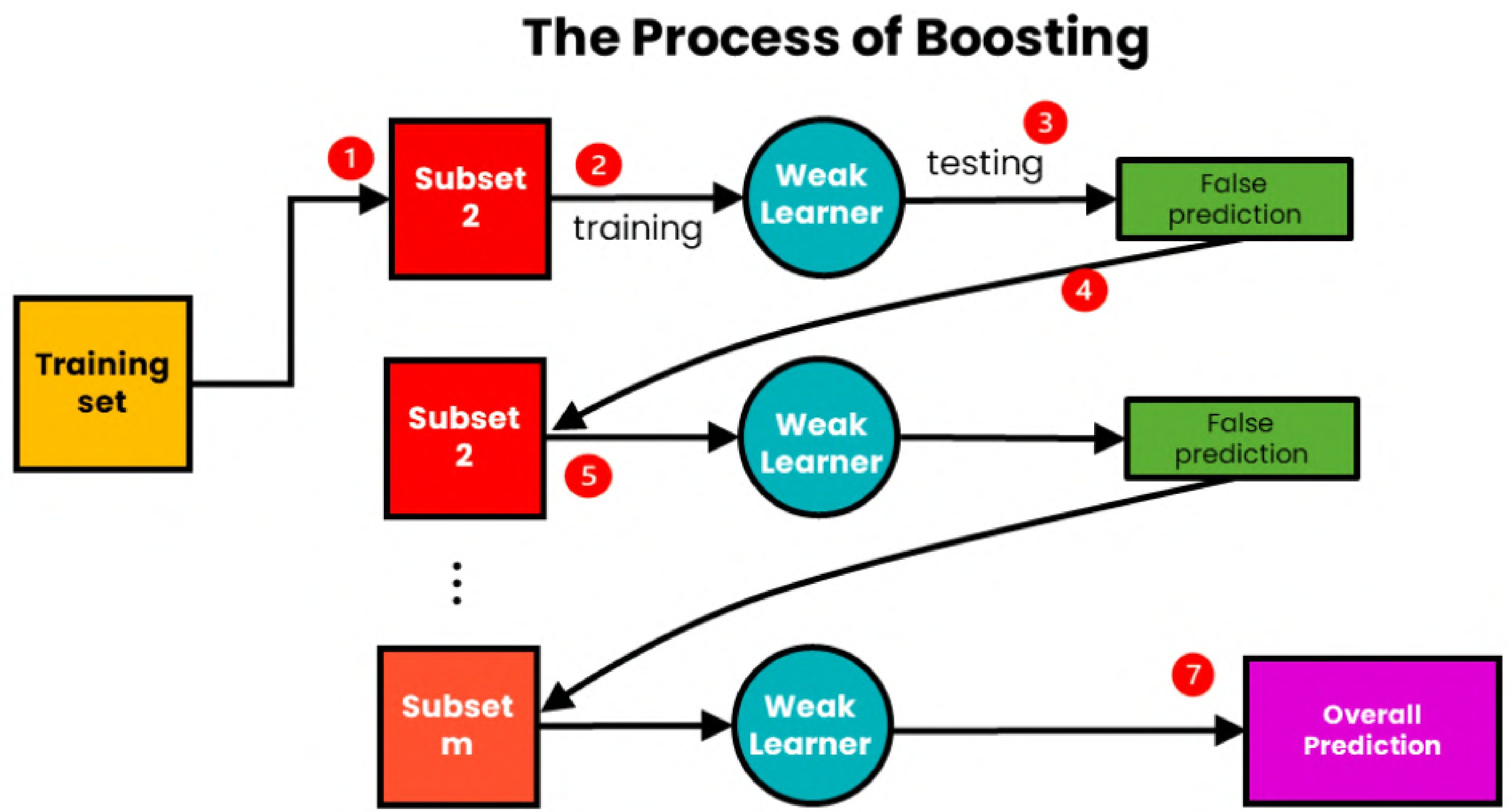
1. **Sequential Training**: Training models sequentially, each one trying to correct the errors made by the previous models.
2. **Weight Adjustment**: Each instance in the training set is weighted. Initially, all instances have equal weights. After each model is trained, the weights of misclassified instances are increased so that the next model focuses more on difficult cases.
3. **Model Combination**: Combining the predictions from all models to produce the final output, typically by weighted voting or weighted averaging.



AdaBoost (canonical boosting)
Gradient Boosting Machines
Stochastic Gradient Boosting
(XGBoost and similar)

Boosting

- **Key Benefits**:
- **Reduces Bias**: By focusing on hard-to-classify instances, boosting reduces bias and improves the overall model accuracy.
- **Produces Strong Predictors**: Combining weak learners leads to a strong predictive model.

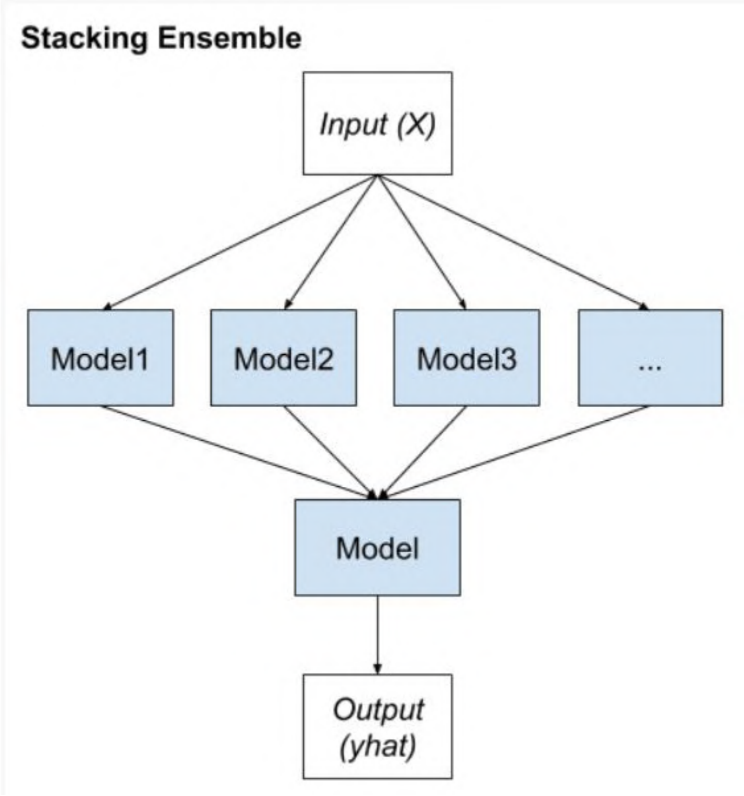

The Process of Boosting
Training set
1
Subset 2
2
training
Weak Learner
3
testing
False prediction
4
Subset 2
5
Weak Learner
False prediction
Subset m
Weak Learner
7
Overall Prediction

Ensemble Learning Methods

- Stacking

- **Stacking** (Stacked Generalization) is an ensemble learning technique that aims to combine multiple models to improve predictive performance. It involves the following steps:

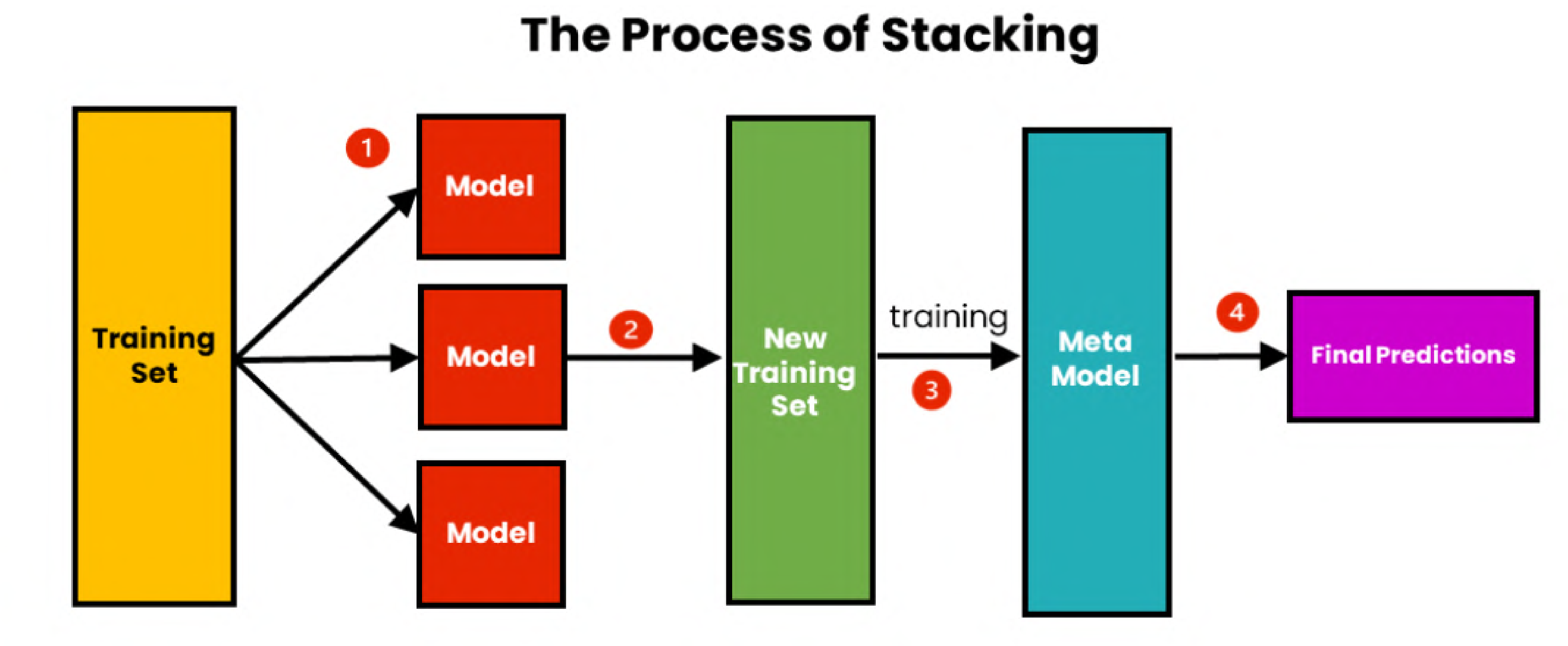
1. **Base Models**: Training multiple models (level-0 models) on the same dataset.
2. **Meta-Model**: Training a new model (level-1 or meta-model) to combine the predictions of the base models. Using the predictions of the base models as input features for the meta-model.



Key Benefits:

•**Leverages Model Diversity**: By combining different types of models, stacking can capture a wide range of patterns in the data.

•**Improves Performance**: The meta-model learns how to best combine the predictions from the base models, often leading to improved performance over individual models.



Meta Learning

- Subset of machine learning described as “learning to learn”
- Metalearning differs from base-learning in the scope of the level of adaptation
- Synonyms:
  - Selection of Algorithms, Ranking learning methods; Hyper-parameter Optimization; Adaptive learning; Self-adaptive systems; Dynamic selection of bias; Learning to learn
- Learning at the base-level is focused on accumulating experience on a specific task
  - e.g., credit rating, medical diagnosis, mine-rock discrimination, fraud detection, etc.
- Learning at the metalevel is concerned with accumulating experience on the performance of multiple applications of a learning system

Meta Learning

- Predictions of the base model, along with the expected outputs, provide the input and output pairs of the training dataset used to fit the meta-model
- Focuses on the design and application of learning algorithms to acquire and use metaknowledge to assist machine learning users with the process of model selection
- It studies how to choose the right bias dynamically, and thus differs from base-level learning, where the bias is fixed or user-parameterized
- Important feature of self-adaptive systems- increase in efficiency through experience

Importance of Meta Learning

- Machine learning algorithms have some problems, such as
  - The need for large datasets for training
  - High operating costs due to many trials/experiments during the training phase
  - Experiments/trials take a long time to find the best model that performs best for a given data set

Structure of the Meta-learning System

- composed of two parts
  - Acquisition of metaknowledge from ML systems
  - Application of metaknowledge to new problems with the objective of identifying an optimal learning algorithm or technique
  - Example
    - if we are dealing with a classification task, metaknowledge can be used to select a suitable classifier for the new problem

Meta-Learning

- Two primary phases are involved in the typical meta-learning workflow:
- **Meta – Learning**
  - **Tasks:** Exposure to a range of tasks, each with its own set of parameters or characteristics, is part of the meta-training phase
  - **Model Training**: Many tasks are used to train a base model, also known as a learner. The purpose of this model is to represent shared knowledge or common patterns among various tasks
  - **Adaption**: With few examples, the model is trained to quickly adjust its parameters to new tasks.
- **Meta – Testing(Adaption**)
  - **New Task**: The model is given a brand-new task during the meta-testing stage that it was not exposed to during training
  - **Few Shots:** With only a small amount of data, the model is modified for the new task (few-shot learning). In order to make this adaptation, the model's parameters are frequently updated using the examples from the new task
  - **Generalization**: Meta-learning efficacy is evaluated by looking at how well the model quickly generalizes to the new task

Learning the meta-parameters

- Throughout the whole training process, backpropagation is used in meta-learning to back-propagate the meta-loss gradient, all the way back to the original model weights
- It is highly computational, uses second derivatives, and is made easier by frameworks such as Tensorflow and PyTorch
- By contrasting model predictions with ground truth labels, the meta-loss—a measure of the meta-learner's efficacy—is obtained
- Parameters are updated during training by meta-optimizers such as SGD, RMSProp, and Adam
- Three main steps subsumed in meta-learning are as follows:

1. **Inclusion of a learning sub-model**
2. **A dynamic inductive bias:** Altering the inductive bias of a learning algorithm to match the given problem. This is done by altering key aspects of the learning algorithm, such as the hypothesis representation, heuristic formulae, or parameters. Many different approaches exist.
3. **Extracting useful knowledge and experience from the metadata of the model:** Metadata consists of knowledge about previous learning episodes and is used to efficiently develop an effective hypothesis for a new task. This is also a form of **Inductive transfer.**

Employing Metaknowledge to Select Machine Learning Algorithms

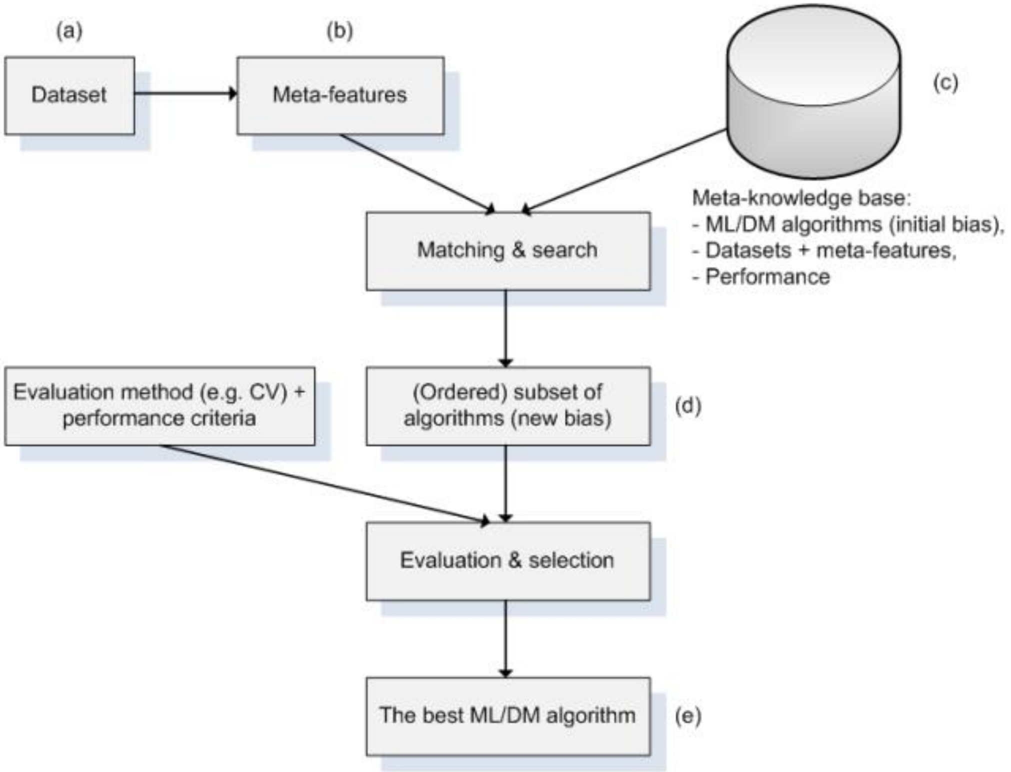


Fig. 1. Selection of machine learning algorithms: Determining the reduced space and selecting the best alternative.

Input/Output of Meta-learning System

- Depends on dataset characteristics or meta features that provide some information to differentiate performance among a given set of learning algorithms
- Measures:
  - number of classes, the number of features, the ratio of examples to features, the degree of correlation between features and target concept, the average class entropy, etc.
- disadvantage:
  - limit on how much information these meta-features can capture, given that all these measures are uni- or bi-lateral measures only

Acquisition of Metaknowledge

- Meta learning in the context of selecting algorithms for machine learning (there are a number of other areas, such as regression, time series forecasting, and optimization)
- rely on expert knowledge
  - representing metaknowledge in the form of rules that match domain (dataset) characteristics with machine learning algorithms
  - Disadvantages:
    - the resulting rule set is likely to be incomplete
    - timely and accurate maintenance of the rule set as new machine learning algorithms become available is problematic
- automatic procedure
  - we need a pool of problems (datasets) and a set of machine learning algorithms that we wish to consider

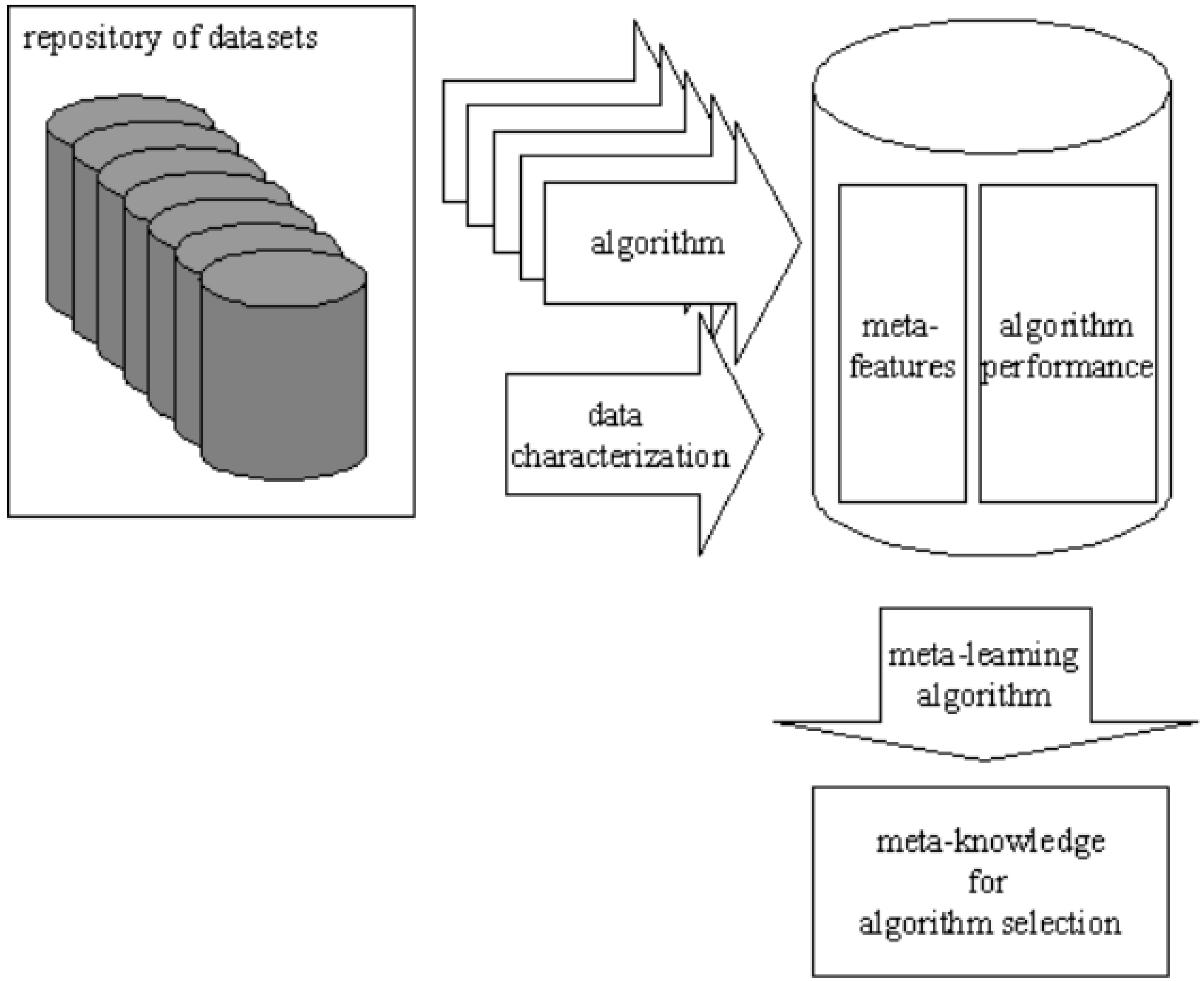
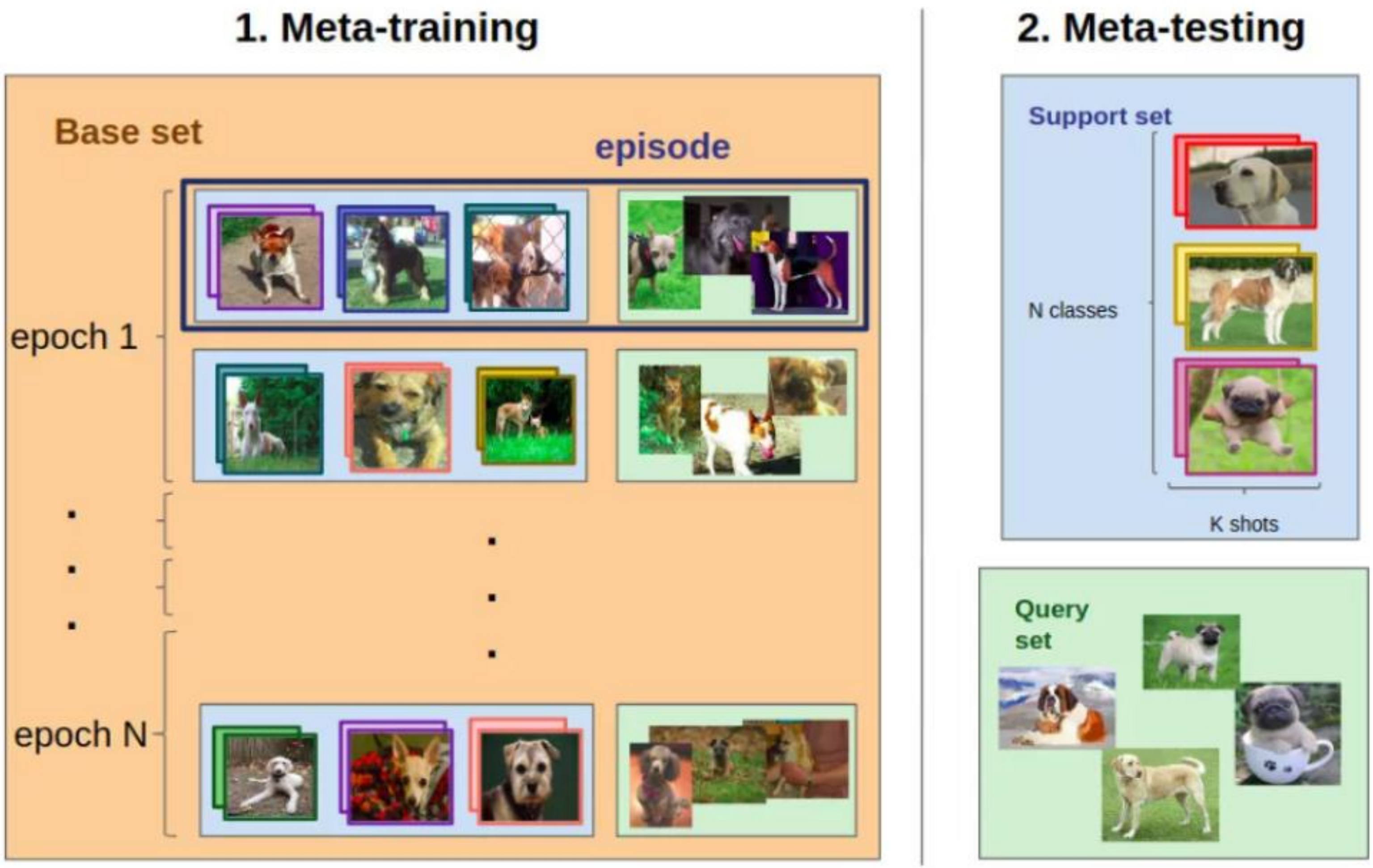
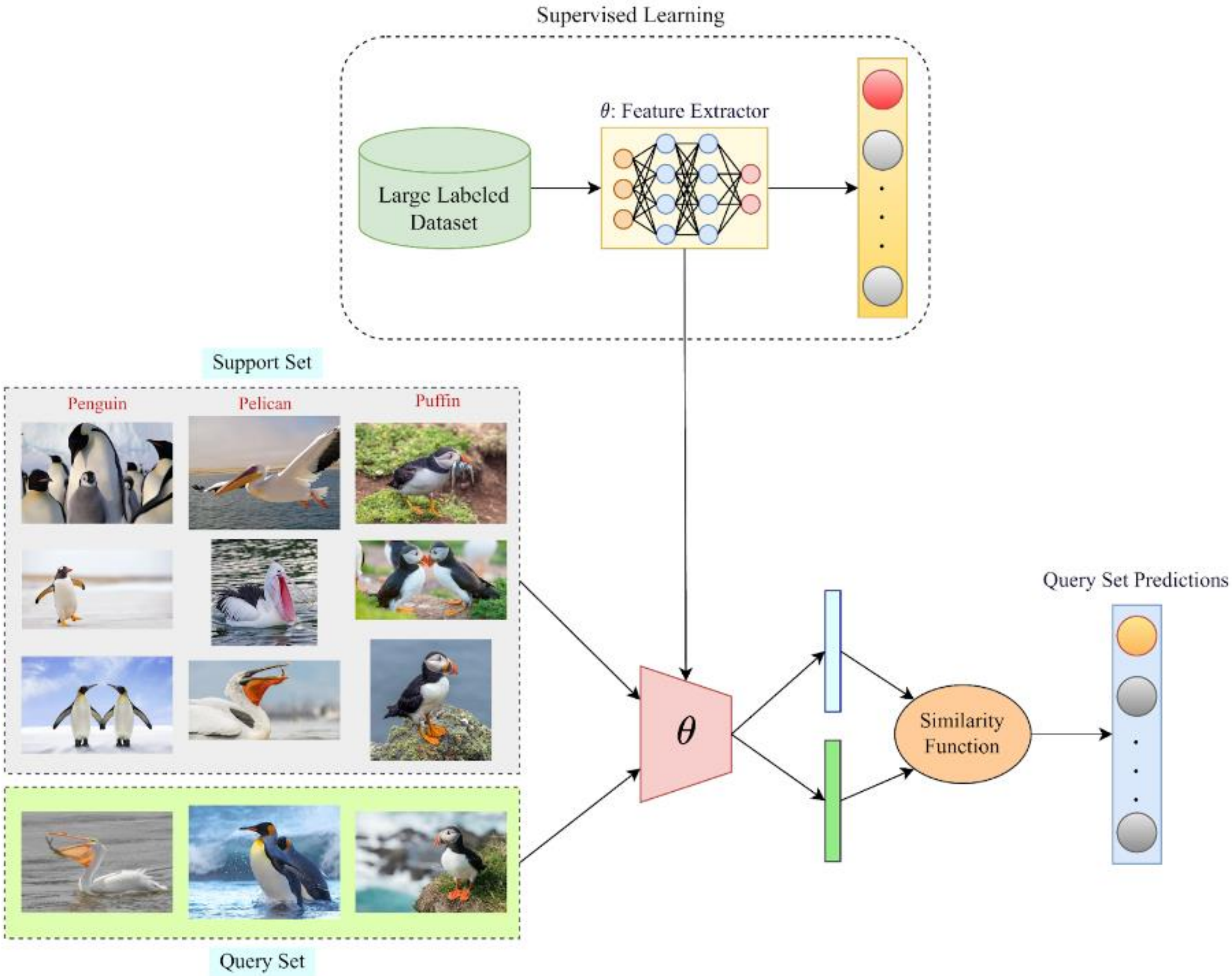


Fig. 2 Acquisition of Metadata for the Metaknowledge base

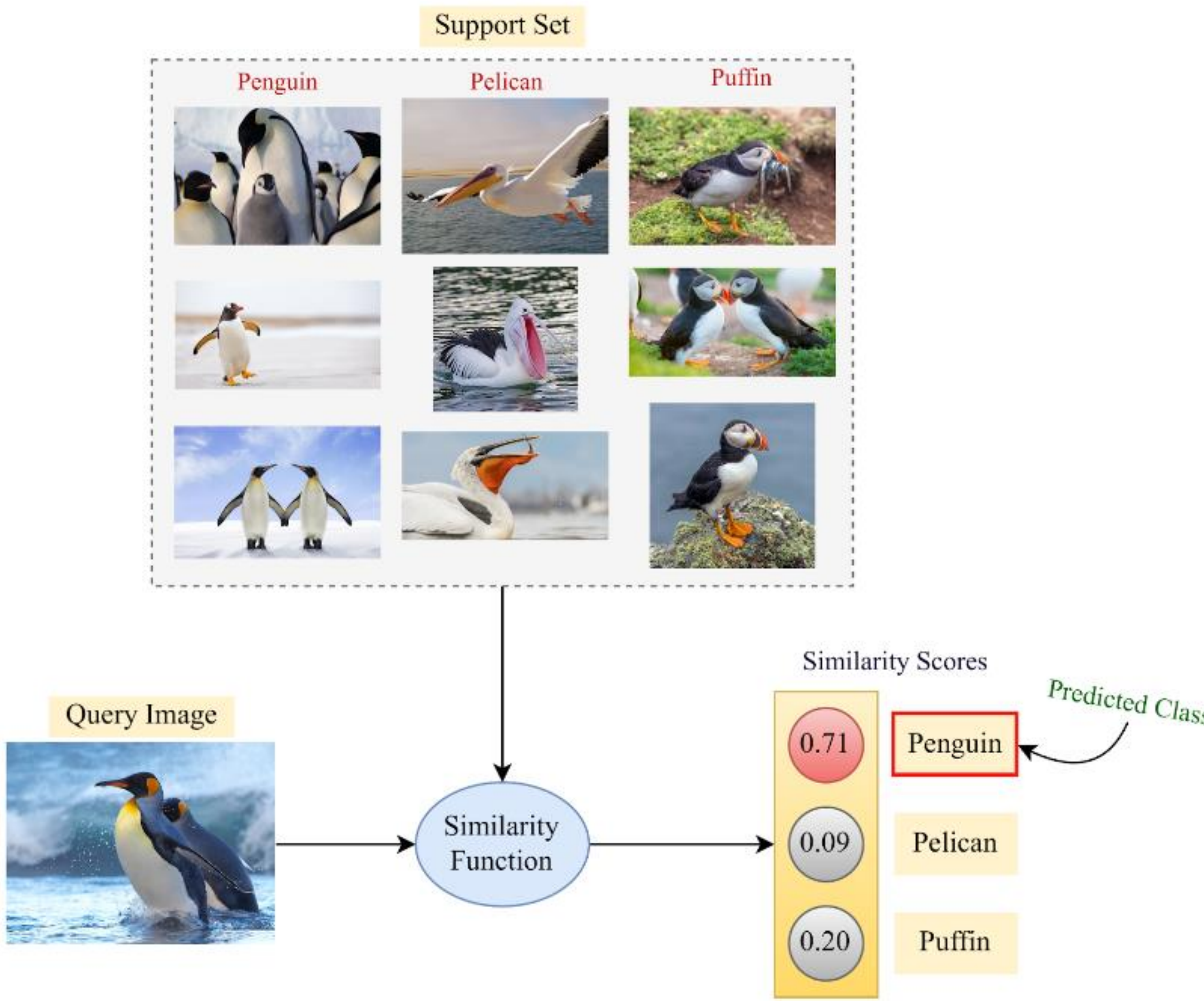

1. Meta-training
2. Meta-testing
Base set
episode
epoch 1
epoch N
Support set
N classes
K shots
Query set



Supervised Learning
Large Labeled Dataset
θ: Feature Extractor
Support Set
Penguin
Pelican
Puffin
θ
Similarity Function
Query Set Predictions
Query Set

Meta-learning approaches and its applications

- Metric Learning
  - learning the metric space for predictions
  - similar to the nearest neighbor algorithm which measures the similarity or distance to learn the given examples
  - Applications include few shot classification, where the goal is to classify new classes with very few examples
- Model-Agnostic Meta-Learning (MAML)
  - It is an optimization-based framework enables a model to quickly adapt to new tasks with only a few examples by learning generalizable features that can be used in different tasks
  - In MAML, the model is trained on a set of meta-training tasks, which are similar to the target tasks but have a different distribution of data
  - The model learns a set of generalizable parameters that can be quickly adapted to new tasks with only a few examples by performing a few gradient descent steps
- Model-based Meta-Learning:
  - It could be a neural network with a certain architecture that is designed for fast updates, or it could be a more general optimization algorithm that can quickly adapt to new tasks
  - The parameters of a model are trained such that even a few iterations of applying gradient descent with relatively few data samples from a new task (new domain) can lead to good generalization on that task



Overview of how a Few-Shot model makes a prediction. Image by the author

Advantages of Meta-Learning

- **Higher model prediction accuracy**
  - Optimization of learning algorithms: For example, optimization of hyperparameters to achieve the best results
  - Helps to learn algorithms better adapt to changing conditions
  - Identifying clues for designing better learning algorithms
- **The faster and cheaper training process**
  - Supporting learning from fewer examples
  - Increase the speed of learning processes by limiting the necessary experiments
- **Building more generalized models**:
  - learning to solve many tasks, not just one task: meta-learning does not focus on training one model on one specific data set