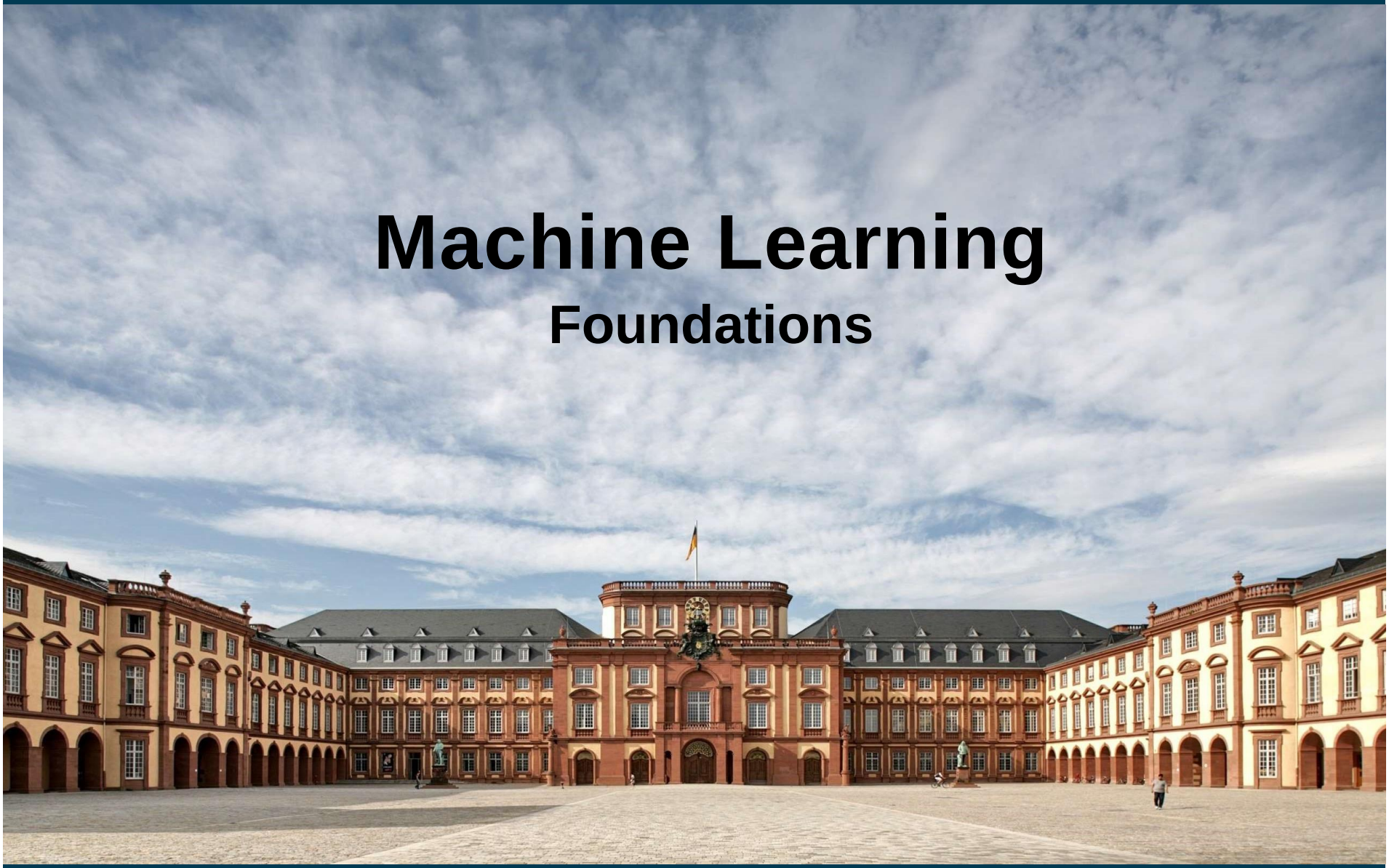
Machine Learning

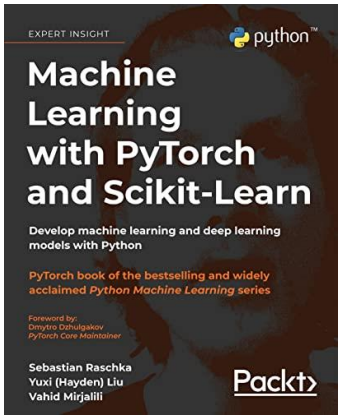
Foundations



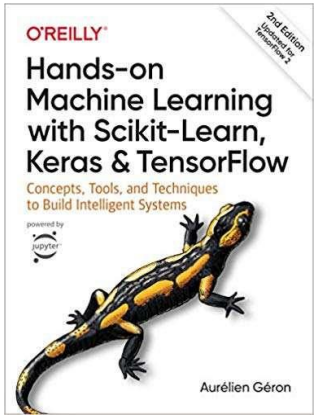
"As to methods, there may be a million and then some, but principles are few. The man who grasps principles can successfully select his own methods."

Ralph Waldo Emerson

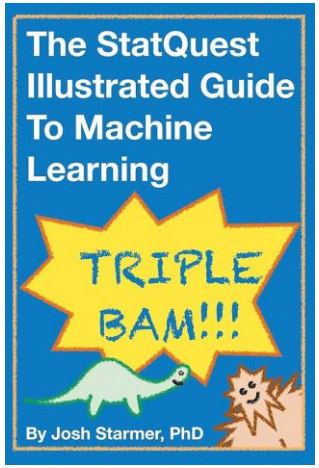
Recommended Books



Sebastian Raschka , Yuxi (Hayden) Liu, Vahid Mirjalili:
Machine Learning with PyTorch and Scikit-Learn: Develop machine learning and deep learning models with Python.
Packt.



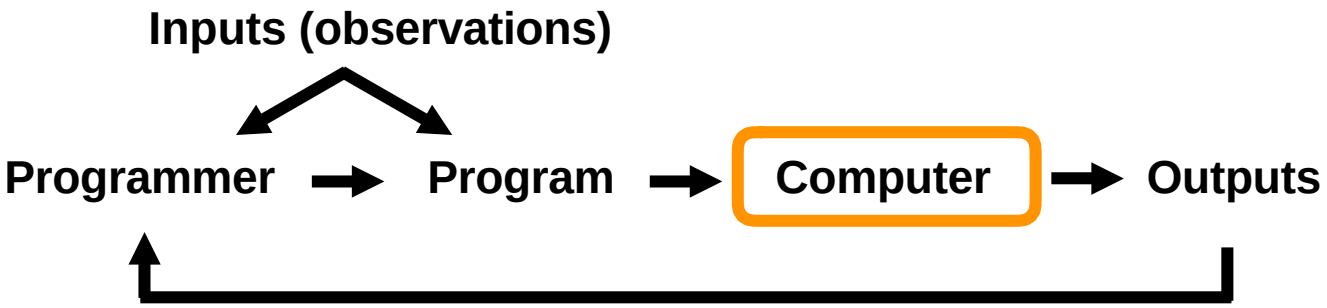
Aurélien Géron:
Hands-on Machine Learning with Scikit-Learn, Keras & TensorFlow.
2nd or 3rd Edition, O'Reilly, 2019 or 2022



Josh Starmer:
The StatQuest Illustrated Guide To Machine Learning.
StatQuest Publications.

https://www.youtube.com/@statquest

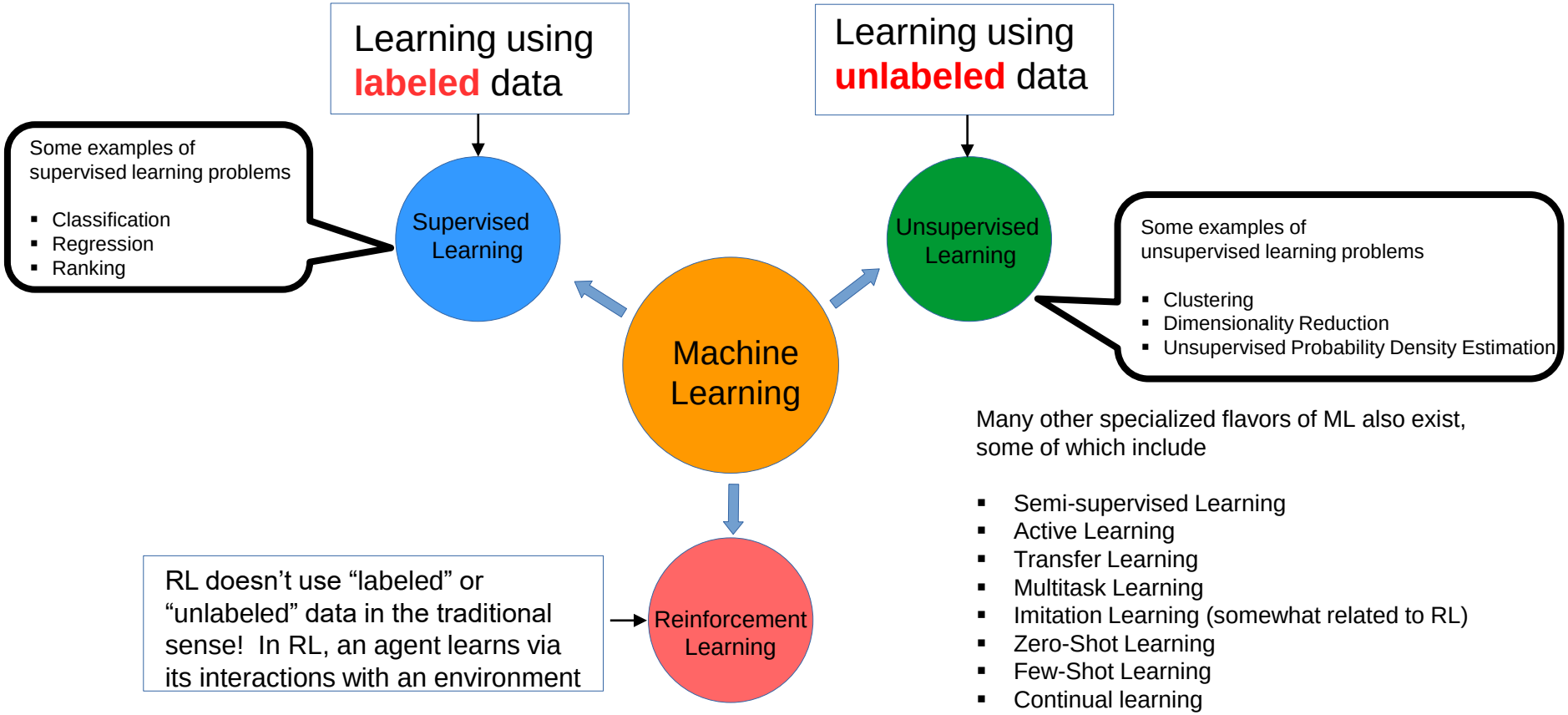
YouTube



Machine learning is the field of study that gives computers the ability to learn without being explicitly programmed

— Arthur Samuel (1959)

A Loose Taxonomy of ML



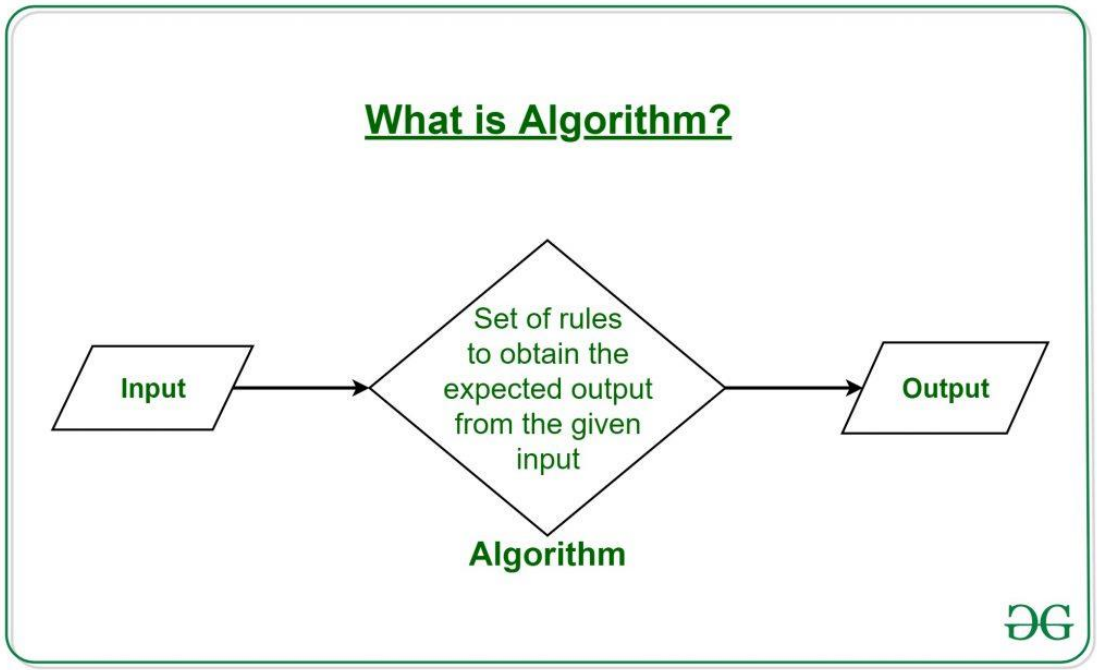
Many other specialized flavors of ML also exist, some of which include

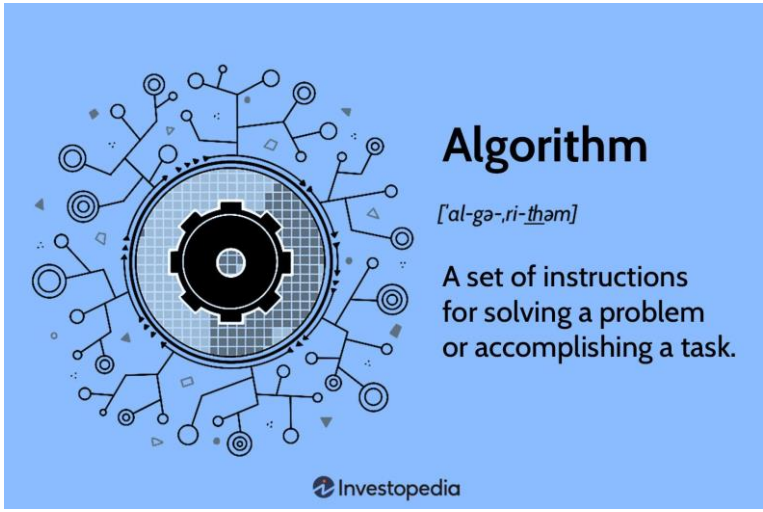
- Semi-supervised Learning
- Active Learning
- Transfer Learning
- Multitask Learning
- Imitation Learning (somewhat related to RL)
- Zero-Shot Learning
- Few-Shot Learning
- Continual learning

So, What is ML?

Machine learning (ML) is a branch of artificial intelligence (AI) and computer science that focuses on the using **data** and **algorithms** to enable AI to imitate the way that humans learn, gradually improving its accuracy.
-IBM

https://www.ibm.com/topics/machine-learning





What is Machine Learning?

Given examples (training data), make a machine learn system behavior or discover patterns

Data

Given to us

Algorithm

We need to design it

Model

$f(\mathbf{x})$

Final output which enables us to make predictions

Algorithms vs Model

- Linear regression algorithm produces a model, that is, a vector of values of the coefficients of the model.

- Decision tree algorithm produces a model comprised of a tree of if-then statements with specific values.

- Neural network along with backpropagation + gradient descent: produces a model comprised of a trained (weights assigned) neural network.

How does machine learning work?

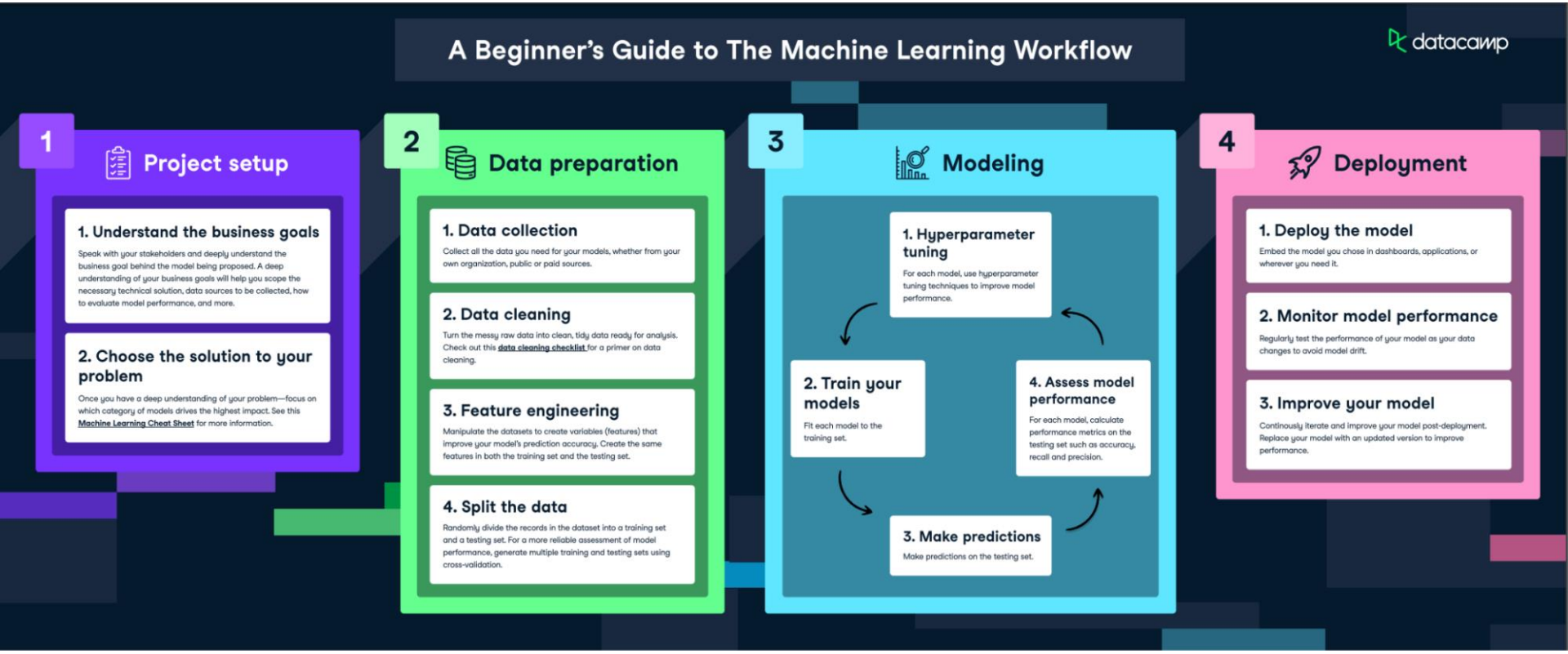
UC Berkeley breaks out the learning system of a machine learning algorithm into three main parts.

1.A Decision Process: In general, machine learning algorithms are used to make a prediction or classification. Based on some input data, which can be labeled or unlabeled, your algorithm will produce an *estimate* about a pattern in the data.

2.An Error Function: An error function evaluates the prediction of the model. If there are known examples, an error function can make a comparison to assess the accuracy of the model.

3.A Model Optimization Process: If the model can fit better to the data points in the training set, then weights are adjusted to *reduce the discrepancy between the known example and the model estimate*. The algorithm will repeat this iterative "evaluate and optimize" process, updating weights autonomously until a threshold of accuracy has been met.

How does machine learning work?



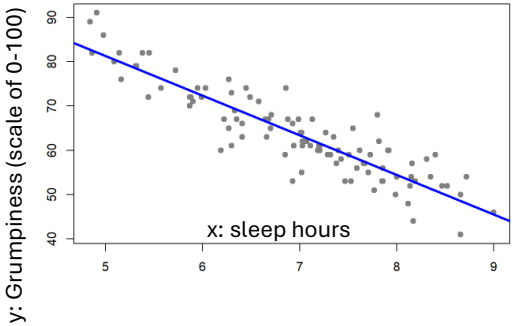
ML: Some Perspectives

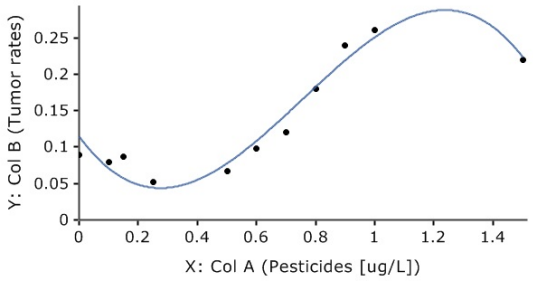
Geometric Perspective

Recall that feature extraction converts inputs into a numeric representation

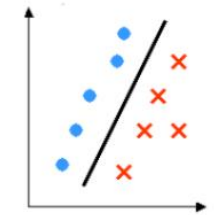
- Basic fact: Inputs in ML problems can often be represented as points or vectors in some vector space
- Doing ML on such data can thus be seen from a geometric view

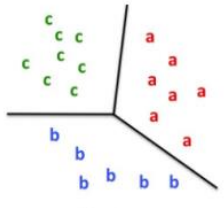
Regression: A supervised learning problem. Goal is to model the relationship between input (x) and real-valued output (y). This is akin to a line or curve fitting problem

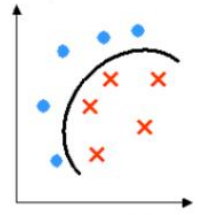


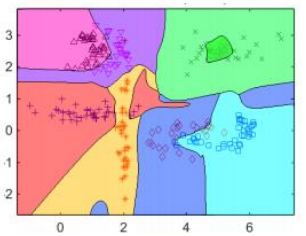


Classification: A supervised learning problem. Goal is to learn a to predict which of the two or more classes an input belongs to. Akin to learning linear/nonlinear separator for the inputs


Two-Class (binary) Linear Classification

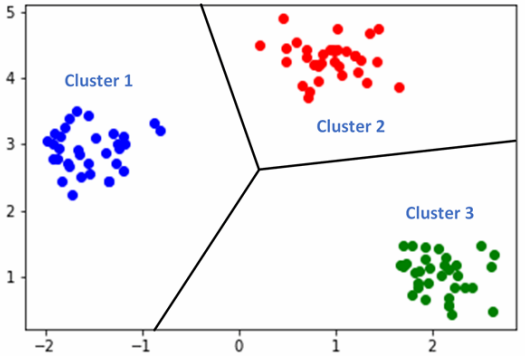

Multi-Class Linear Classification


Two-Class (binary) Nonlinear Classification

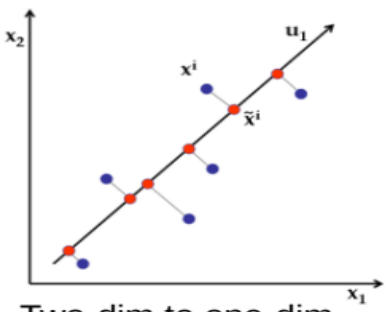

Multi-Class Nonlinear Classification

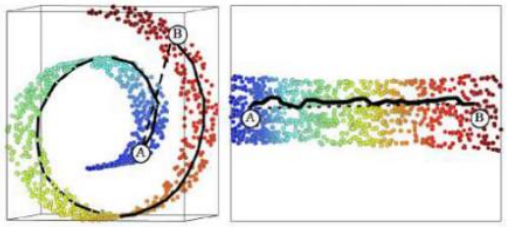
Geometric Perspective

Clustering: An unsupervised learning problem. Goal is to group inputs in a few clusters based on their similarities with each other



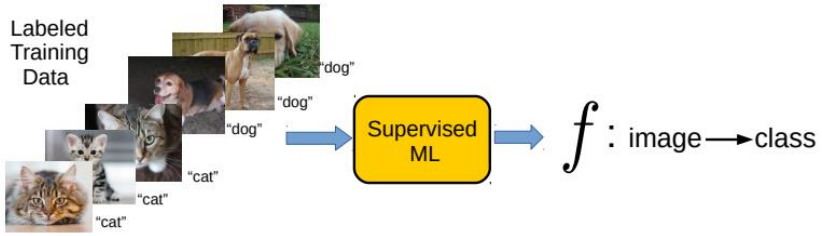
Dimensionality Reduction: An unsupervised learning problem. Goal is to compress the size of each input without losing much information present in the data





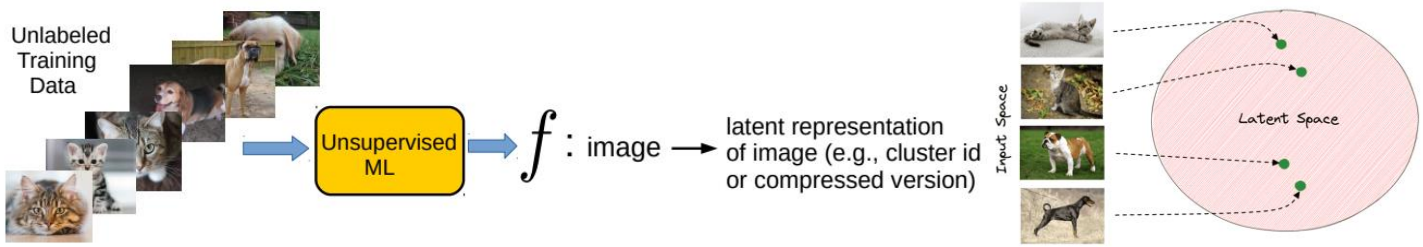
Perspective as function approximation

- Supervised Learning ("predict output given input") can be usually thought of as learning a function *f* that maps each input to the corresponding output



- Unsupervised Learning ("model/compress inputs") can also be usually thought of as learning a function *f* that maps each input to a compact representation

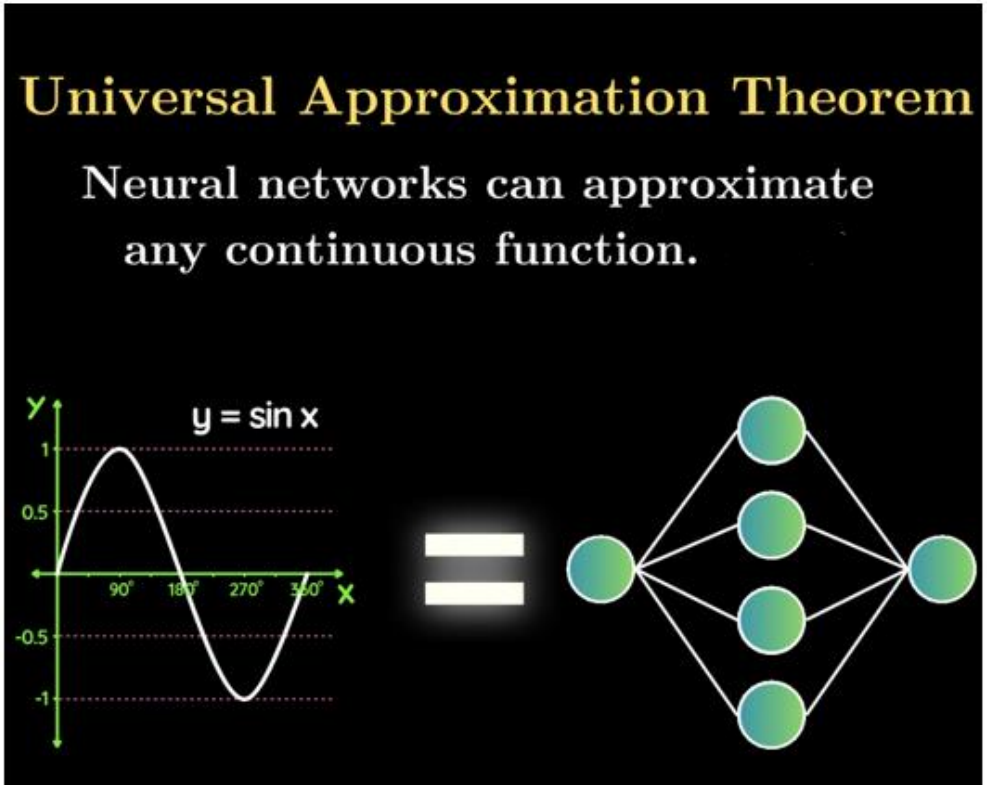
Harder since we don't know the labels in this case

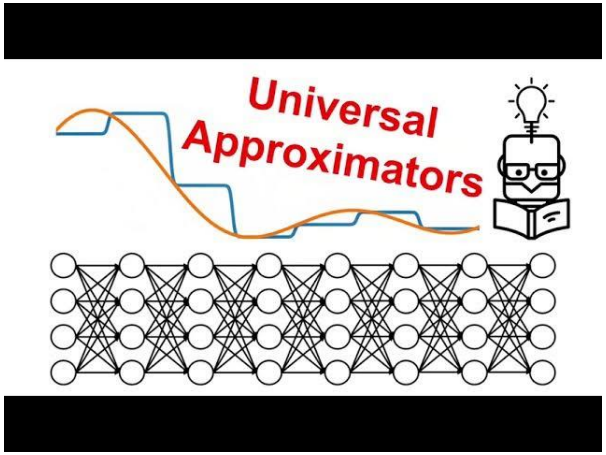


- Reinforcement Learning can also be seen as doing function approximation

Perspective as function approximation

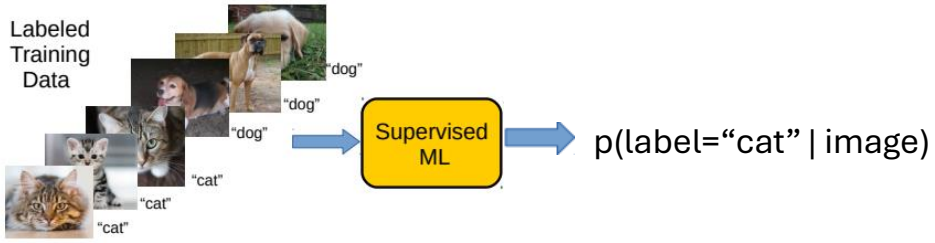
Neural networks can learn any function due to their flexible architecture, the *Universal Approximation Theorem*, the power of backpropagation, and their ability to model complex, non-linear relationships.





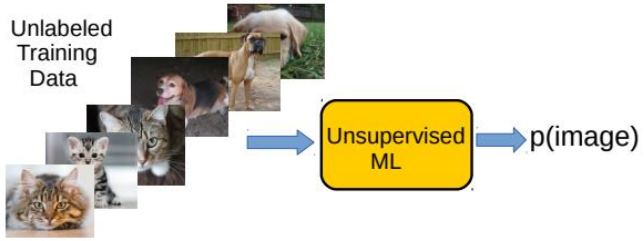
Perspective as probability estimation

- Supervised Learning ("predict output given input") can be thought of as estimating the conditional probability of each possible output given an input



- Unsupervised Learning ("model/compress inputs") can be thought of as estimating the probability density of the inputs

Harder since we don't know the labels in this case



- Reinforcement Learning can also be seen as estimating probability densities

Classes of Machine Learning Algorithms

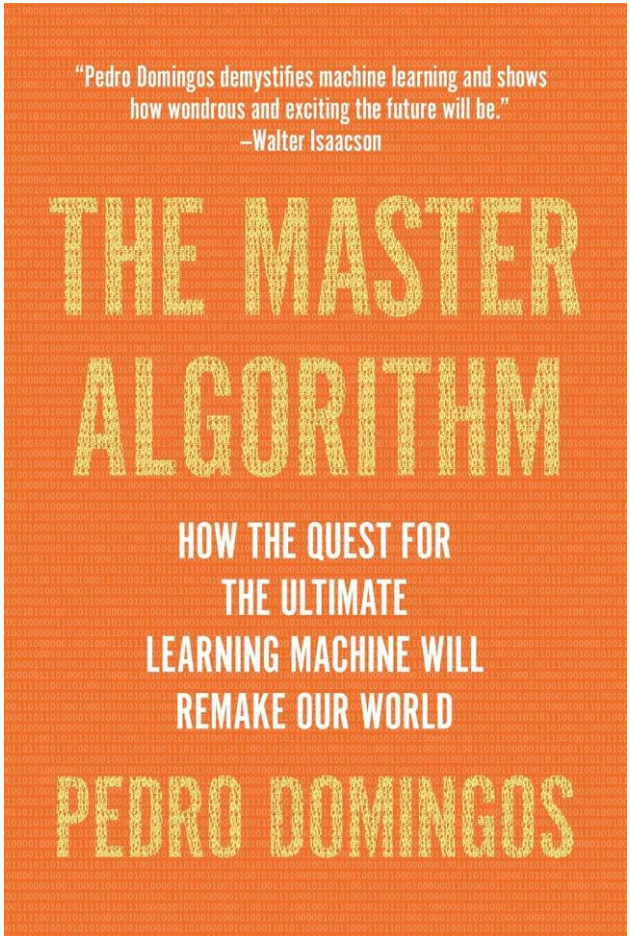
Classes of Machine Learning Algorithms

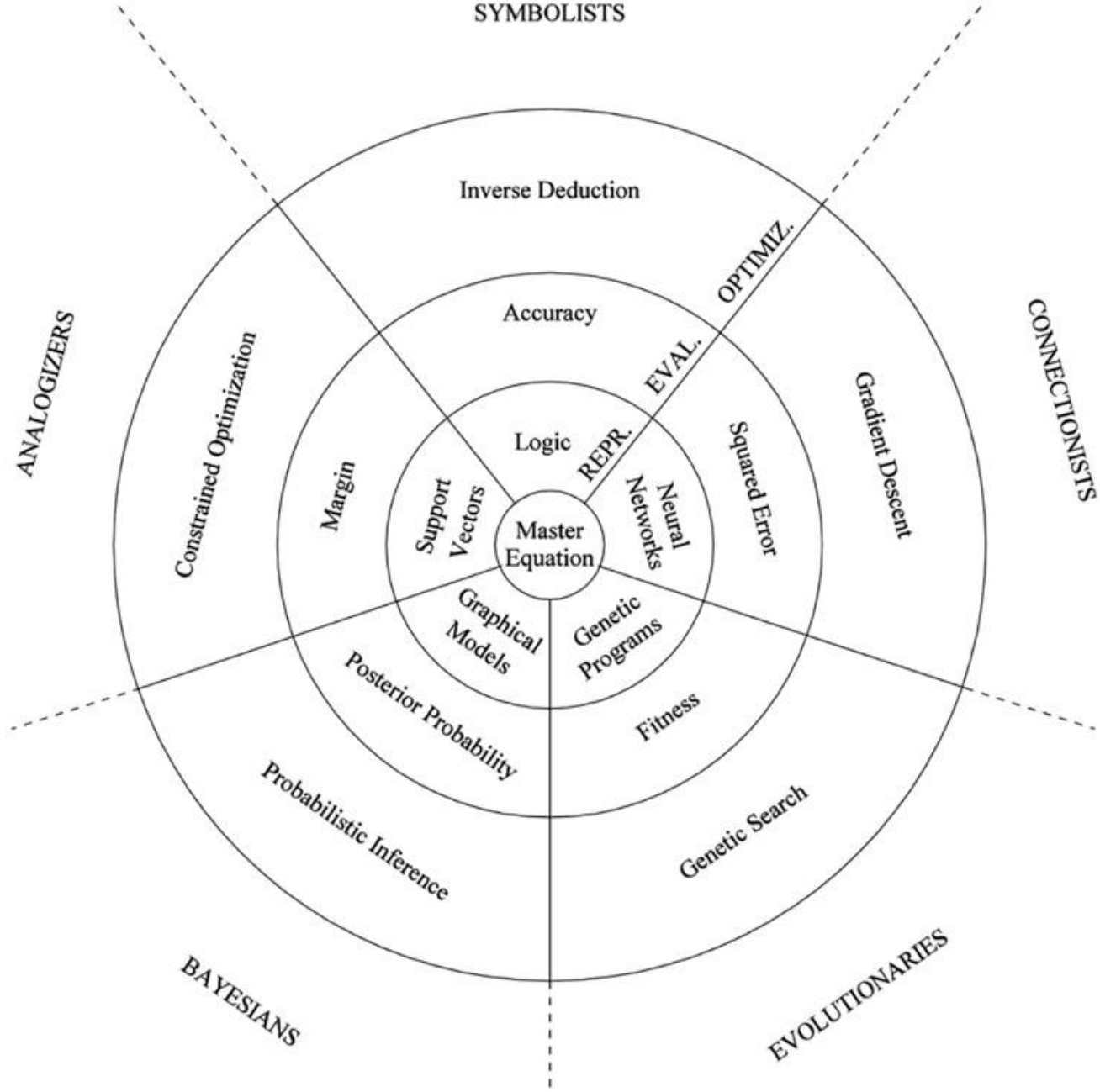
- Generalized linear models (e.g., logistic regression)
- Support vector machines (e.g., linear SVM, RBF-kernel SVM)
- Artificial neural networks (e.g., multi-layer perceptrons)
- Tree- or rule-based models (e.g., decision trees)
- Graphical models (e.g., Bayesian networks)
- Ensembles (e.g., Random Forest)
- Instance-based learners (e.g., K-nearest neighbors)

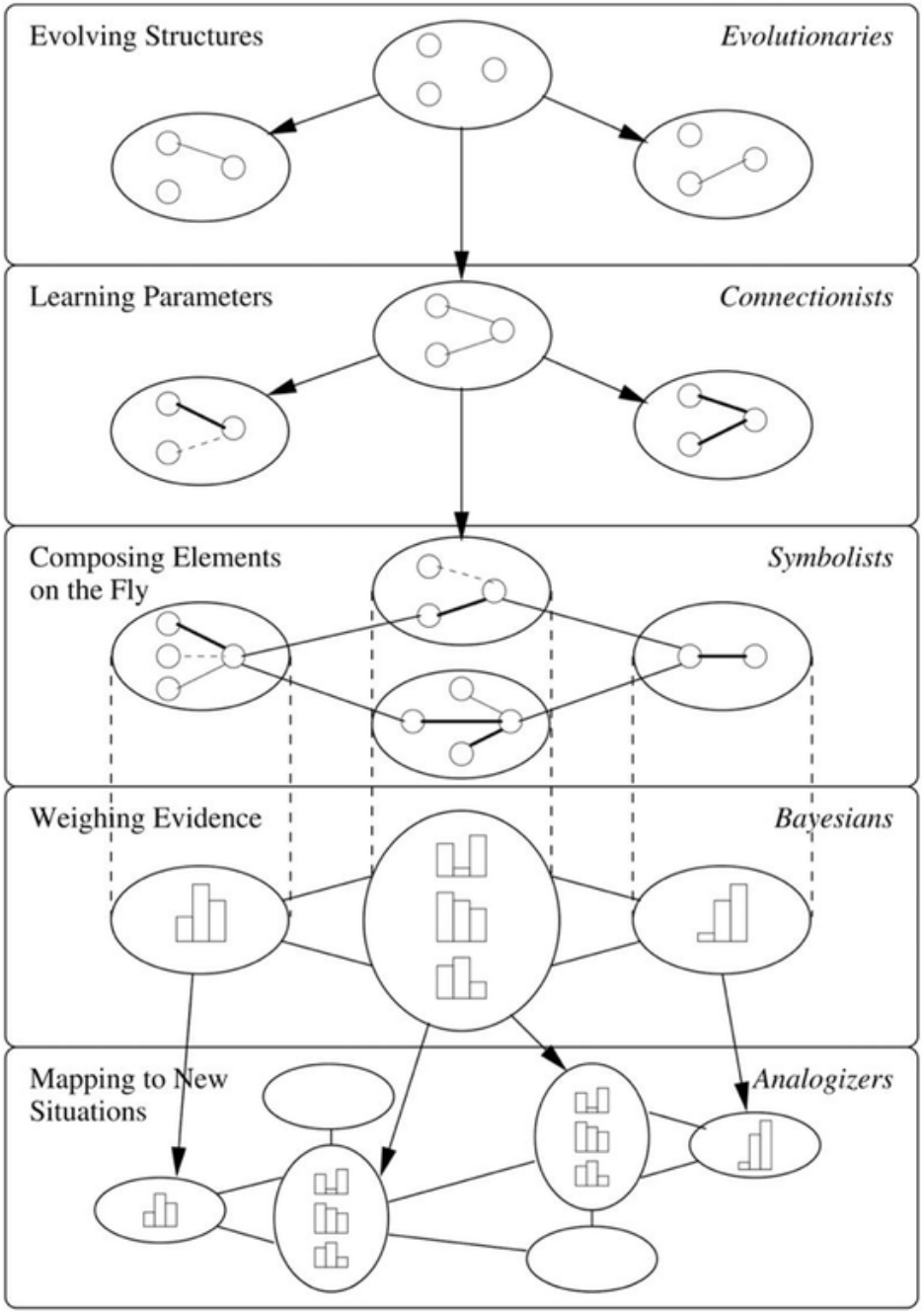
The 5 Tribes of the ML world

The Master Algorithm: The books' central hypothesis is presented:
'All knowledge — past, present, and future — can be deduced from data by a single, universal learning algorithm.'

Tribe	Origins	Master Algorithm
Symbolists	Logic, philosophy	Inverse deduction
Connectionists	Neuroscience	Backpropagation
Evolutionaries	Evolutionary biology	Genetic programming
Bayesians	Statistics	Probabilistic inference
Analogizers	Psychology	Kernel machines




SYMBOLISTS
CONNECTIONISTS
EVOLUTIONARIES
BAYESIANS
ANALOGIZERS
Inverse Deduction
Gradient Descent
Genetic Search
Probabilistic Inference
Constrained Optimization
OPTIMIZ.
Accuracy
EVAL.
Squared Error
Fitness
Posterior Probability
Margin
REPR.
Logic
Neural Networks
Genetic Programs
Graphical Models
Support Vectors
Master Equation


Evolving Structures
Evolutionaries
Learning Parameters
Connectionists
Composing Elements
on the Fly
Symbolists
Weighing Evidence
Bayesians
Mapping to New
Situations
Analogizers

The Five Tribes

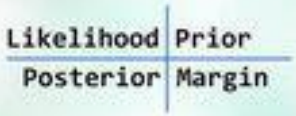
Machine Learning Evolution

1 Symbolists

Use symbols, rules, and logic to represent knowledge and draw logical inference

Favored Algorithm
Rules and Decision Trees

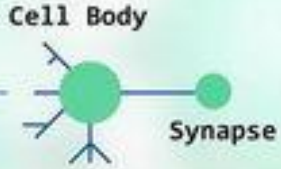
2 Bayesians



Assess the likelihood of occurrence for globalistic inference

Favored Algorithm
Naive Bayes or Markov

3 Connectionists



Recognize and generalize patterns dynamically with matrices of probabilistic, weighted neurons

Favored Algorithm
Neural Networks

4 Evolutionaries

Generate variations and then assess the fitness of each for a given purpose

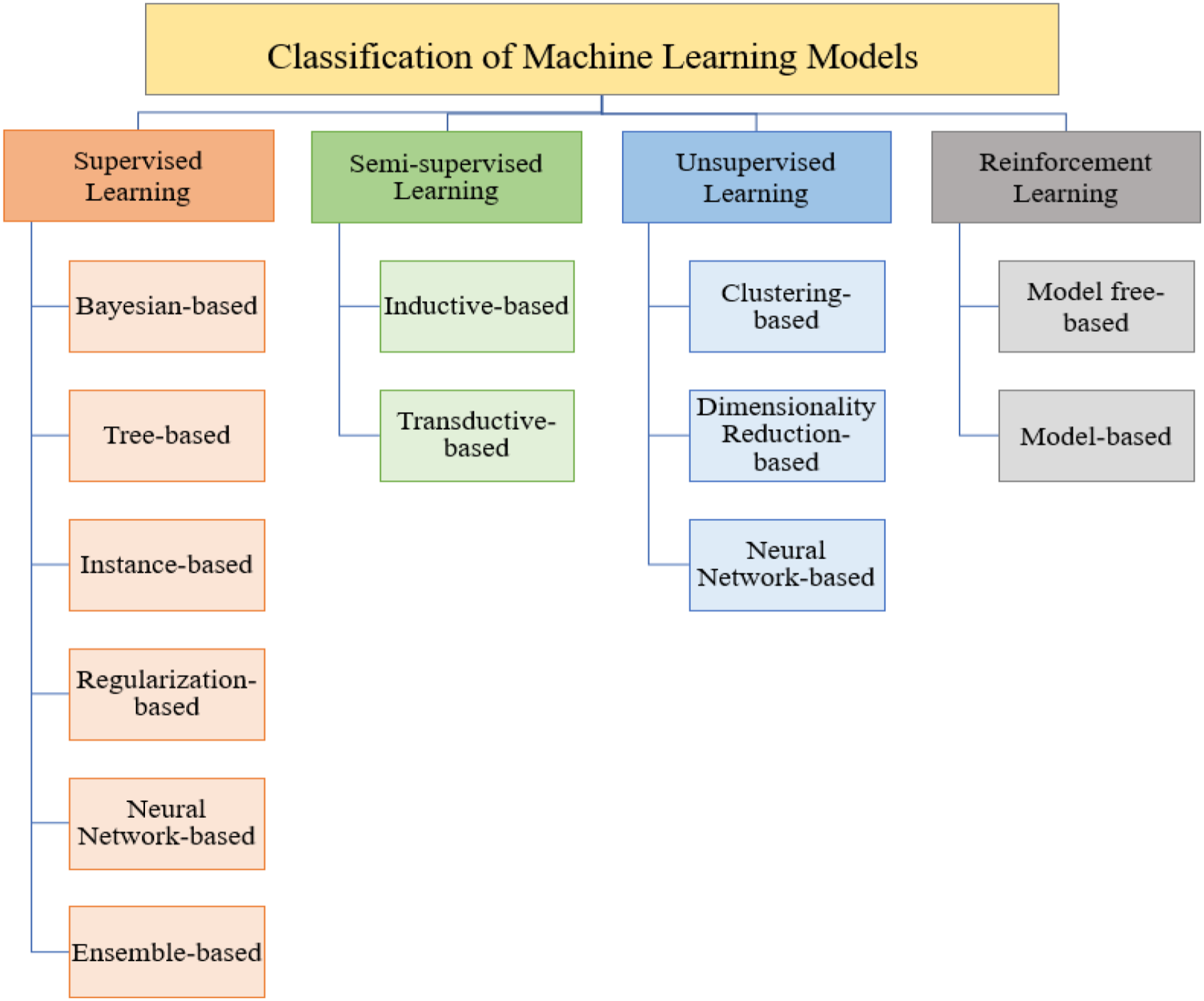
Favored Algorithm
Genetic Programs

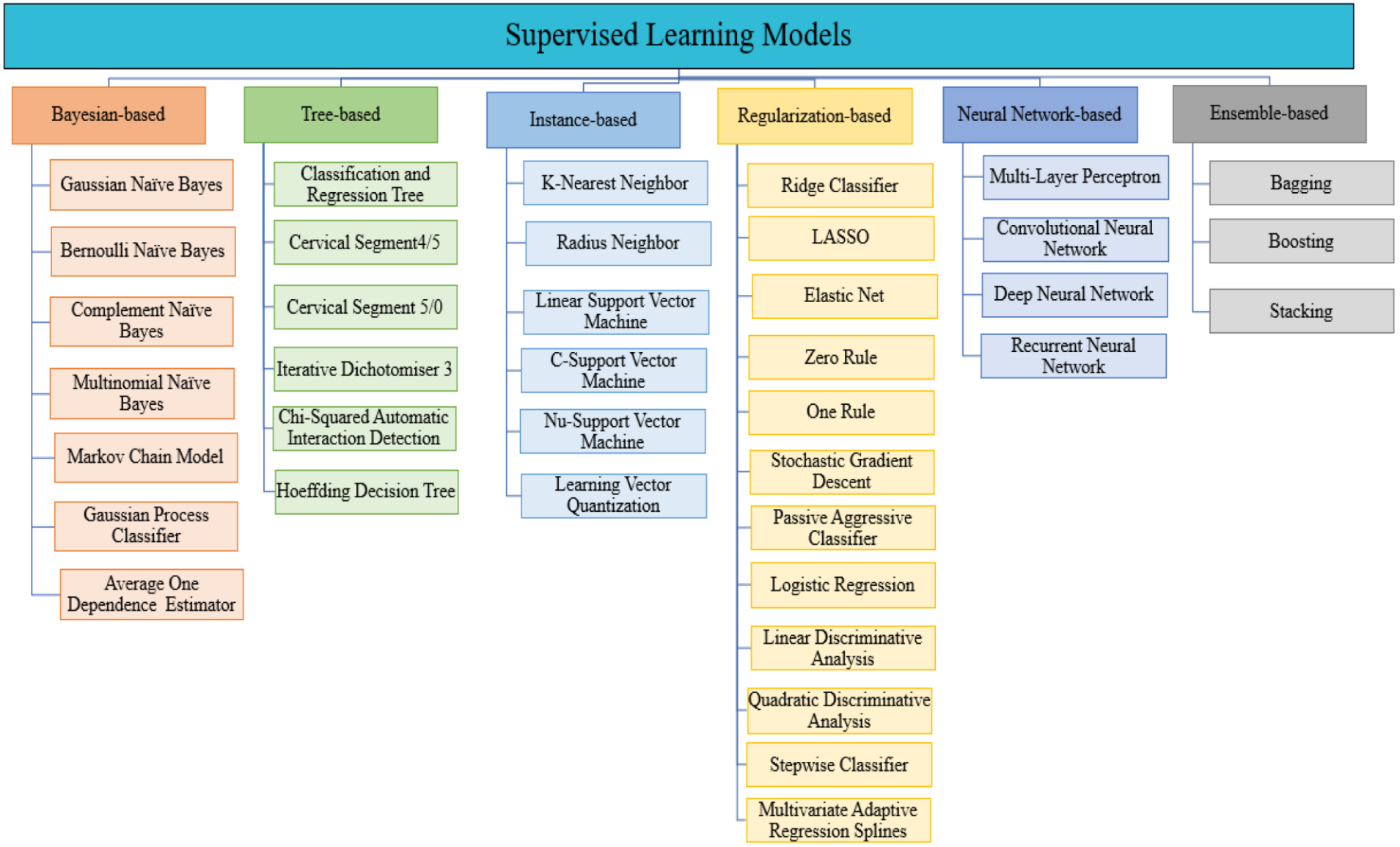
5 Analogizers

Optimize a function in light of constraints ("going as high as you can while staying on the road")

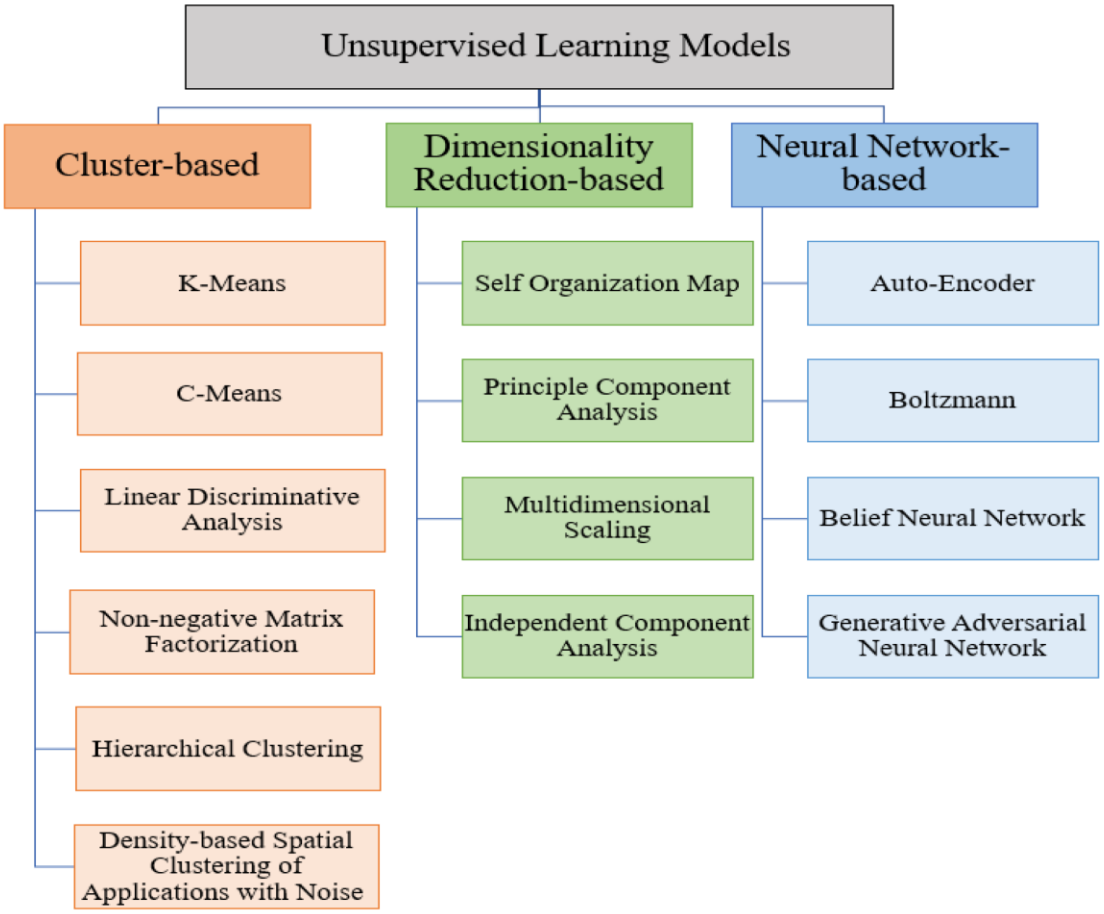
Favored Algorithm
Support Vectors

Talaei Khoei, T.; Kaabouch, N. *Machine Learning: Models, Challenges, and Research Directions*. *Future Internet* **2023**, *15*, 332.
https://doi.org/10.3390/fi15100332
School of Computer Science and Electrical Engineering, University of North Dakota, Grand Forks, ND 58202, USA

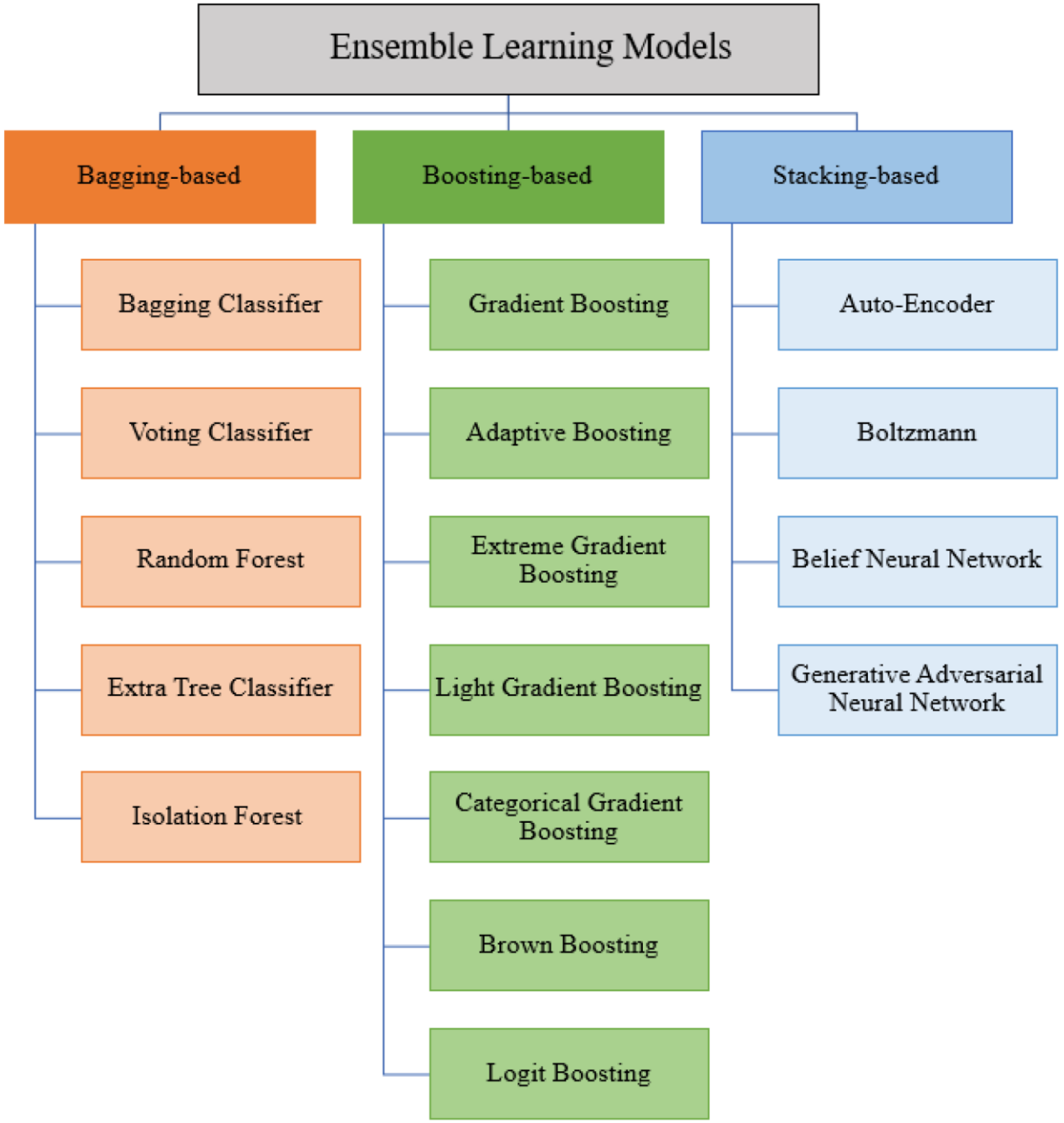

Classification of Machine Learning Models
Supervised Learning
Bayesian-based
Tree-based
Instance-based
Regularization-based
Neural Network-based
Ensemble-based
Semi-supervised Learning
Inductive-based
Transductive-based
Unsupervised Learning
Clustering-based
Dimensionality Reduction-based
Neural Network-based
Reinforcement Learning
Model free-based
Model-based

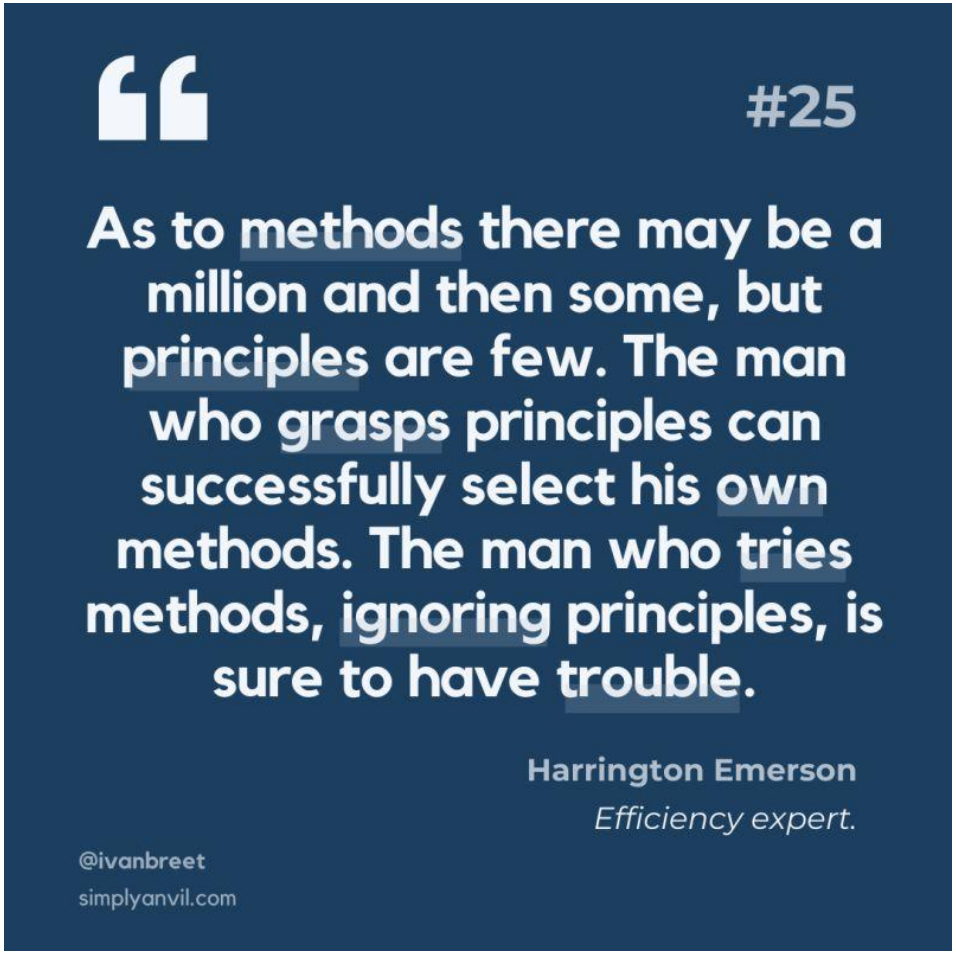

Supervised Learning Models
Bayesian-based
Gaussian Naïve Bayes
Bernoulli Naïve Bayes
Complement Naïve Bayes
Multinomial Naïve Bayes
Markov Chain Model
Gaussian Process Classifier
Average One Dependence Estimator
Tree-based
Classification and Regression Tree
Cervical Segment4/5
Cervical Segment 5/0
Iterative Dichotomiser 3
Chi-Squared Automatic Interaction Detection
Hoeffding Decision Tree
Instance-based
K-Nearest Neighbor
Radius Neighbor
Linear Support Vector Machine
C-Support Vector Machine
Nu-Support Vector Machine
Learning Vector Quantization
Regularization-based
Ridge Classifier
LASSO
Elastic Net
Zero Rule
One Rule
Stochastic Gradient Descent
Passive Aggressive Classifier
Logistic Regression
Linear Discriminative Analysis
Quadratic Discriminative Analysis
Stepwise Classifier
Multivariate Adaptive Regression Splines
Neural Network-based
Multi-Layer Perceptron
Convolutional Neural Network
Deep Neural Network
Recurrent Neural Network
Ensemble-based
Bagging
Boosting
Stacking

Classification Category	Characteristics	Advantages	Disadvantages
Bayesian-Based	• Dealing with uncertain data.	• Ability to quantify the uncertainty; • Capacity for the incorporation of prior knowledge in a principled manner; • Useful for small datasets.	• High computational costs; • Difficulty with selecting priors.
Tree-based	• Data splitting-based; • Non-parametric method.	• High accuracy; • Ease of interpretation; • Handling large datasets.	• Prone to overfitting; • Non-robust.
Instance-based	• Adapting for new and real-time data.	• Capacity to adapt to new and real-time data; • Ability to change the similarity function for each instance at each prediction step; • Fast training process.	• High computational costs; • Expensive memory usage.
Regularization-based	• Regularizing the coefficient estimates towards zero; • Generalization.	• Reduction in model variance with no increase in bias; • Reducing the overfitting issues; • Simplicity; • Computational efficiency.	• Dimensionality reduction; • High bias error.
Neural network-based	• Storing information on an entire network; • Distributed memory; • Ability of parallel processing.	• Ability to detect all possible interactions between predictor variables; • Availability of multiple training processes; • High rate of adaptability.	• Requiring large datasets; • High computational power; • Black box nature; • Prone to overfitting;
Ensemble-based	• Combine multiple learners	• Low variance; • High performance; • Removing noise and biased data.	• High inference time; • Lack of simplicity; • Lack of generalization.



Classification Category	Characteristics	Advantages	Disadvantages
Cluster-based	Divides uncategorized data into similar groups;	• Easy implementation.	• High complexity.
Dimensionality reduction-based	Decreases the number of features in the given dataset;	• Decrease time and storage.	• Data loss.
Neural network-based	Inspiration of human brains.	• Can detect the interactions among predictor variables; • Availability of multiple training processes; • High adaptability.	• Require huge datasets; • High computational power; • Tendency to overfit.


Ensemble Learning Models
Bagging-based
Boosting-based
Stacking-based
Bagging Classifier
Voting Classifier
Random Forest
Extra Tree Classifier
Isolation Forest
Gradient Boosting
Adaptive Boosting
Extreme Gradient Boosting
Light Gradient Boosting
Categorical Gradient Boosting
Brown Boosting
Logit Boosting
Auto-Encoder
Boltzmann
Belief Neural Network
Generative Adversarial Neural Network


#25
As to methods there may be a million and then some, but principles are few. The man who grasps principles can successfully select his own methods. The man who tries methods, ignoring principles, is sure to have trouble.
Harrington Emerson
Efficiency expert.
@ivanbreet
simplyanvil.com

Bias vs Variance

❑ **Bias-Variance Tradeoff:**

•**Bias:** Refers to the error introduced by approximating a real-world problem, which may be complex, by a simplified model. ***High bias leads to underfitting***, where the model is too simple and cannot capture the underlying patterns in the data.

•**Variance:** Refers to the model's sensitivity to fluctuations in the training data. ***High variance leads to overfitting***, where the model captures noise in the training data, leading to poor generalization on new data.

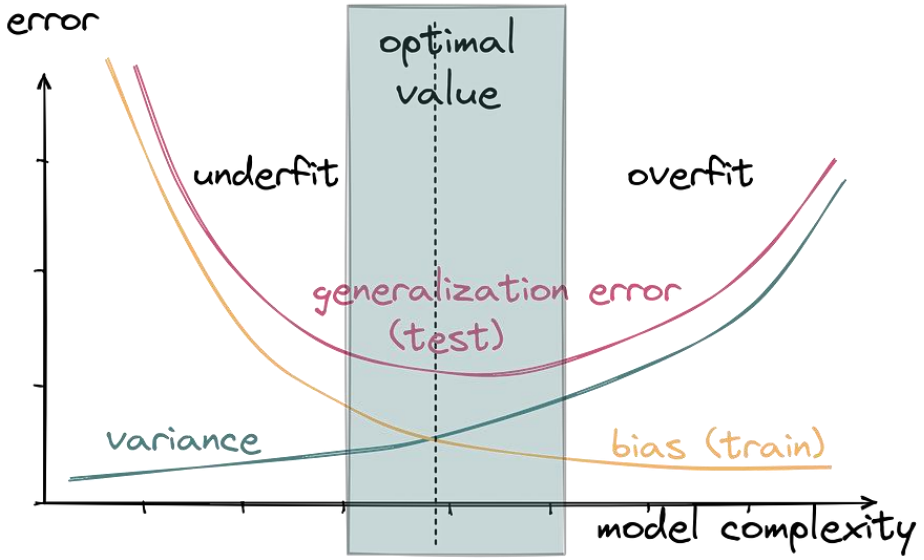
❑ **Generalization Error:**

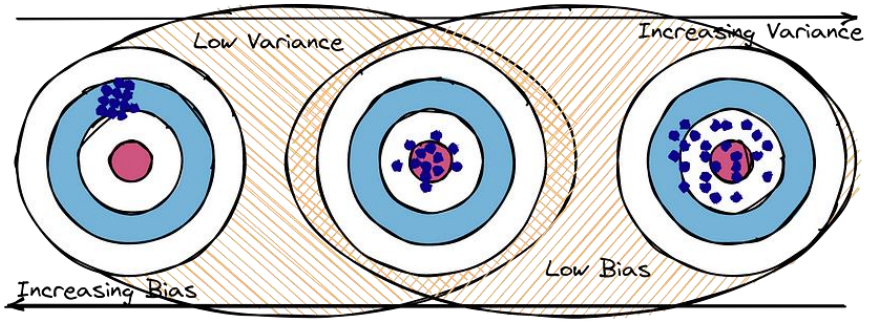
•The graph shows the relationship between bias, variance, and generalization error.

•**Test Error:** As complexity increases, test error initially decreases as the model fits the data better but eventually increases as overfitting occurs.

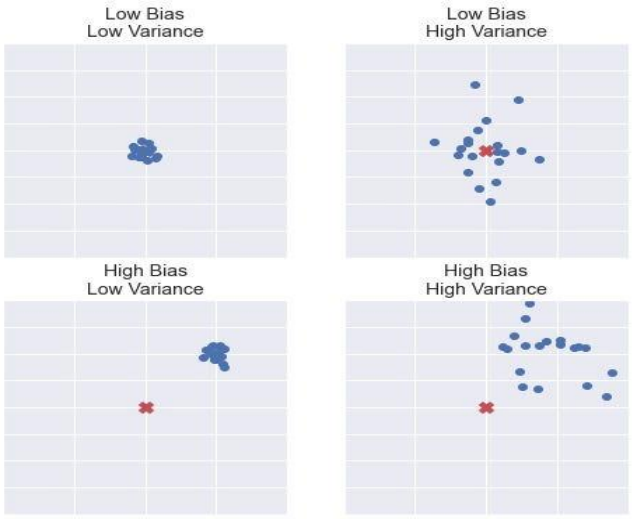
❑ **Optimal Model Complexity:**

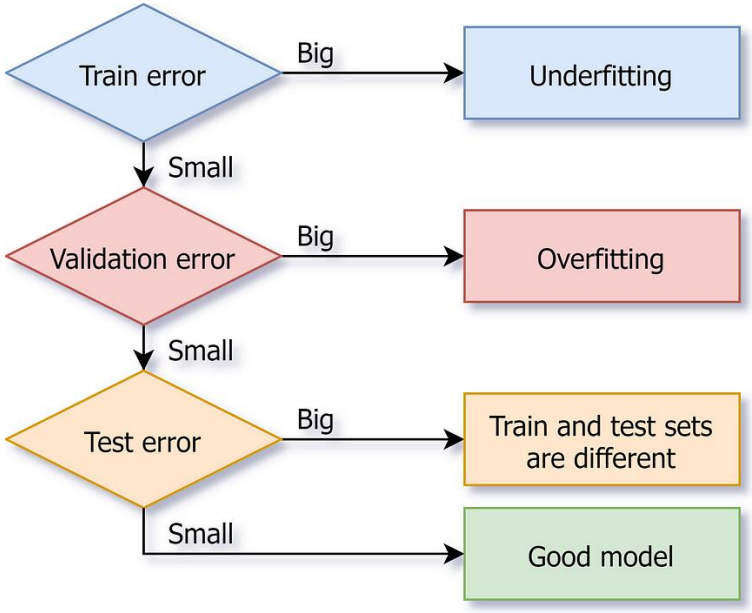
•The region where the test error is minimized represents the optimal complexity for the model. This is where the bias and variance are balanced, leading to the best generalization to new data.

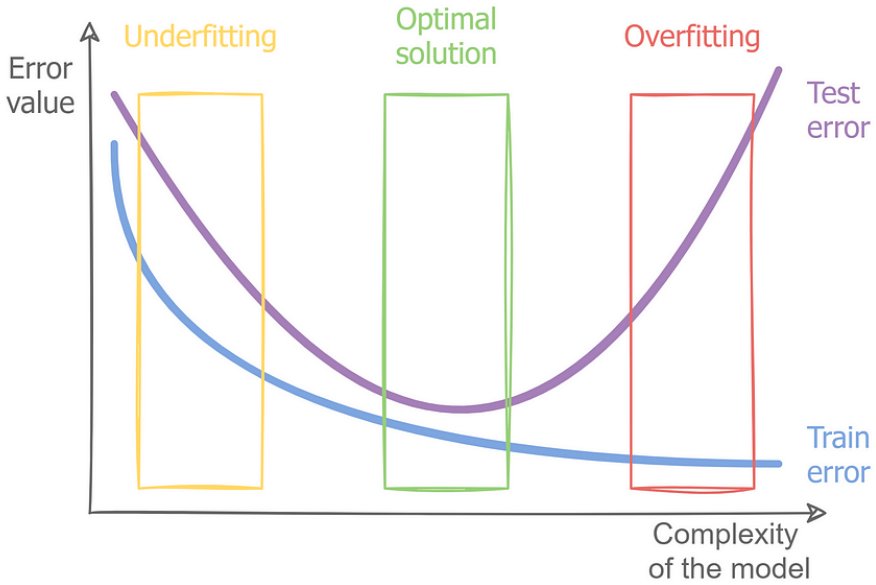




Underfitting and Overfitting







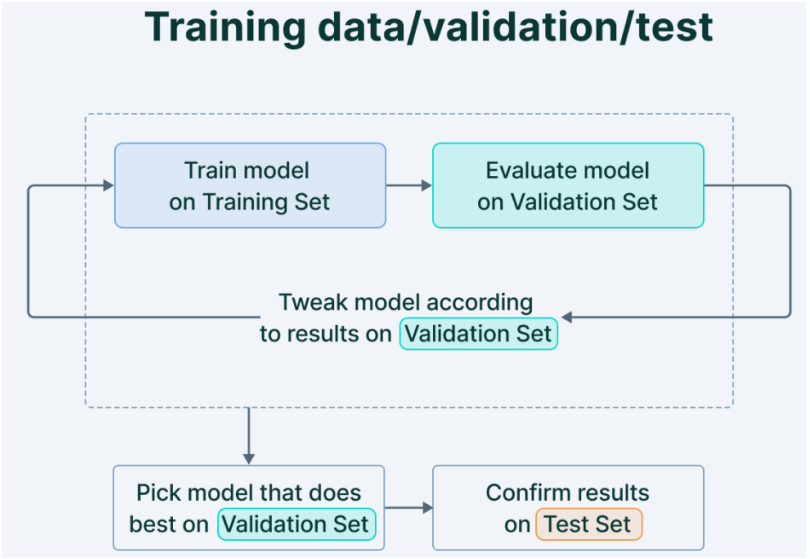
Techniques to fight underfitting and overfitting	
Underfitting	**Overfitting**
More complex model	More simple model
Less regularization	More regularization
Larger quantity of features	Smaller quantity of features
More data can't help	More data can help

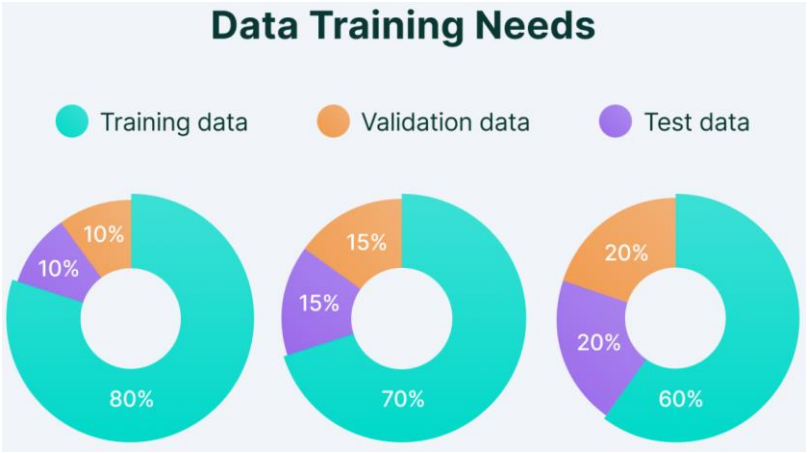
Train Test Validation Split

•**The Training Set:** It is the set of data that is used to train and make the model learn the hidden features/patterns in the data.
•In each *iteration or epoch*, the **same** training data is fed to the ML model repeatedly, and the model continues to learn the features of the data.

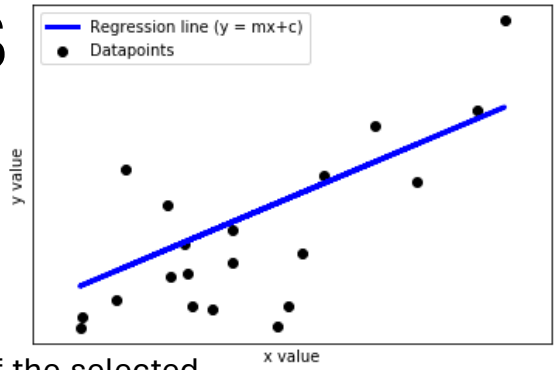
•**The Validation Set:** The validation set is a set of data, separate from the training set, that is used to validate our model performance **during** training.
•This validation process gives information that helps us ***tune*** *the model's hyperparameters* and configurations accordingly.
•The model is trained on the training set and **simultaneously**, the *model evaluation* is performed on the validation set after every epoch.
•The main idea of splitting the dataset into a validation set is to *prevent* our model from **overfitting**

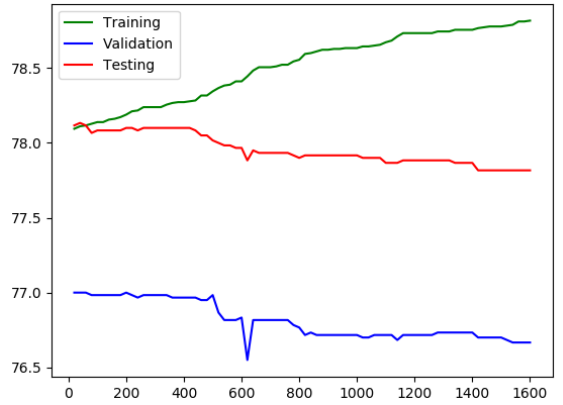
•**The Test Set:** The test set is a separate set of data used to test the model **after** completing the training.
•It provides an unbiased final model **performance metric** in terms of accuracy, precision, etc.





Model Parameters VS HyperParameters

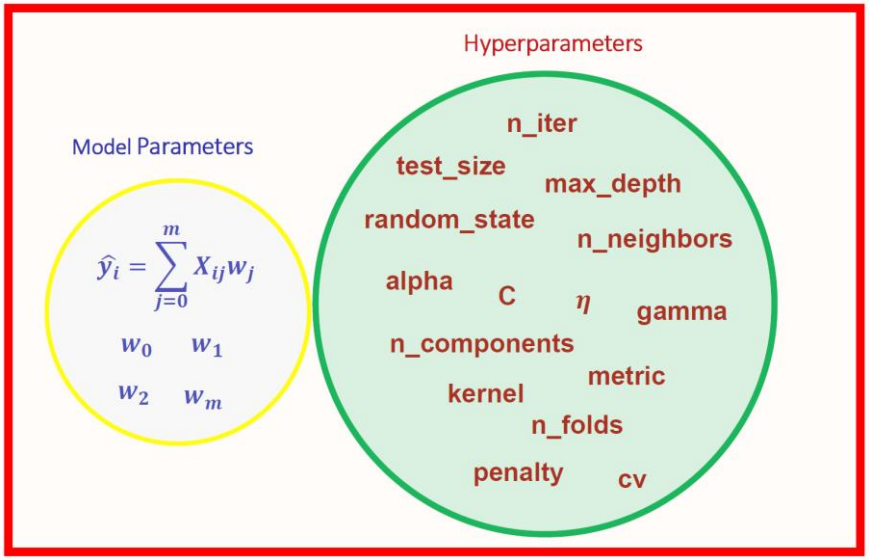




- **Parameter:** A model parameter is a variable of the selected model which can be estimated by fitting the given (training) data to the model. It can not be manually set and is required for making predictions.

- **Model parameters in different models:**
- **m(slope)** and **c(intercept)** in Linear Regression
- **Split points** in Decision Trees
- **weights** and **biases** in Neural Networks

- **Hyperparameter:** A model hyperparameter is the parameter whose value is set before the model start training. They cannot be learned by fitting the model to the data. They are set manually and are required for estimating the model parameters.



- **Model hyperparameters in different models:**
- **Learning rate** in gradient descent
- **Number of iterations** in gradient descent
- **Max-depth** in Decision Trees
- **Number of layers** in a Neural Network
- **Number of neurons** per layer in a Neural Network
- **Number of clusters(k)** in k means clustering