Unit – 4

Machine learning

INTRODUCTION

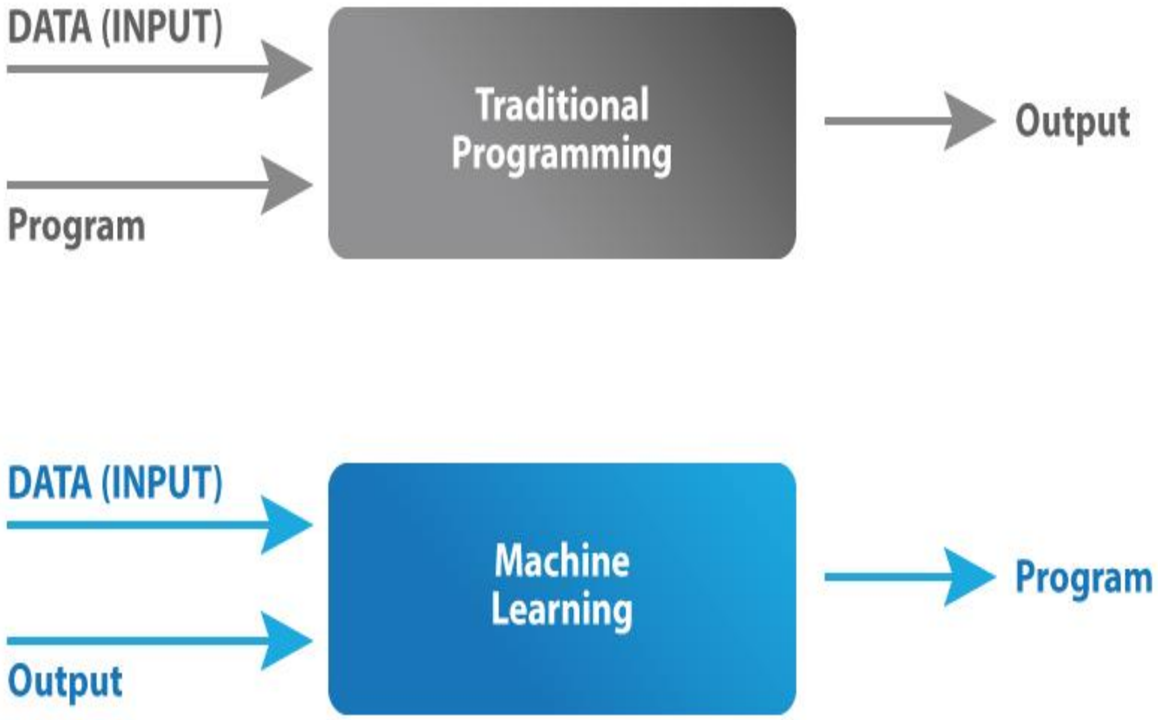
What is the Learning Problem?

Learning = Improving with experience at some task

- Improve over task T,
- with respect to performance measure P,
- based on experience E.

E.g., Learn to play checkers

- T: Play checkers
- P: % of games won in world tournament
- E: opportunity to play against self



Definition

A computer program is said to learn from experience E with respect to some task T and some performance measure P, if its performance on T, as measured by P, improves with experience E.

—Tom Mitchell, 1997

E * T = P

Experience	*	Task	=	Performance
Input Data: • Housing prices • Customer transactions • Clickstream data • Images		**Task:** • Predict prices • Segment customers • Optimize user flows • Categorize images		**Performance:** • Accurate prices • Coherent groupings • KPI lifts • Correctly sorted images

Example

three features: the class of tasks, the measure of performance to be improved, and the source of experience.

A checkers learning problem:

- Task T: playing checkers
- Performance measure P: percent of games won against opponents
- Training experience E: playing practice games against itself

Practice

- **Autonomous Driving Car**
- **Hand writing Recognition**

A robot driving learning problem:

- Task T: driving on public four-lane highways using vision sensors
- Performance measure P: average distance traveled before an error (as judged by human overseer)
- Training experience E: a sequence of images and steering commands recorded while observing a human driver

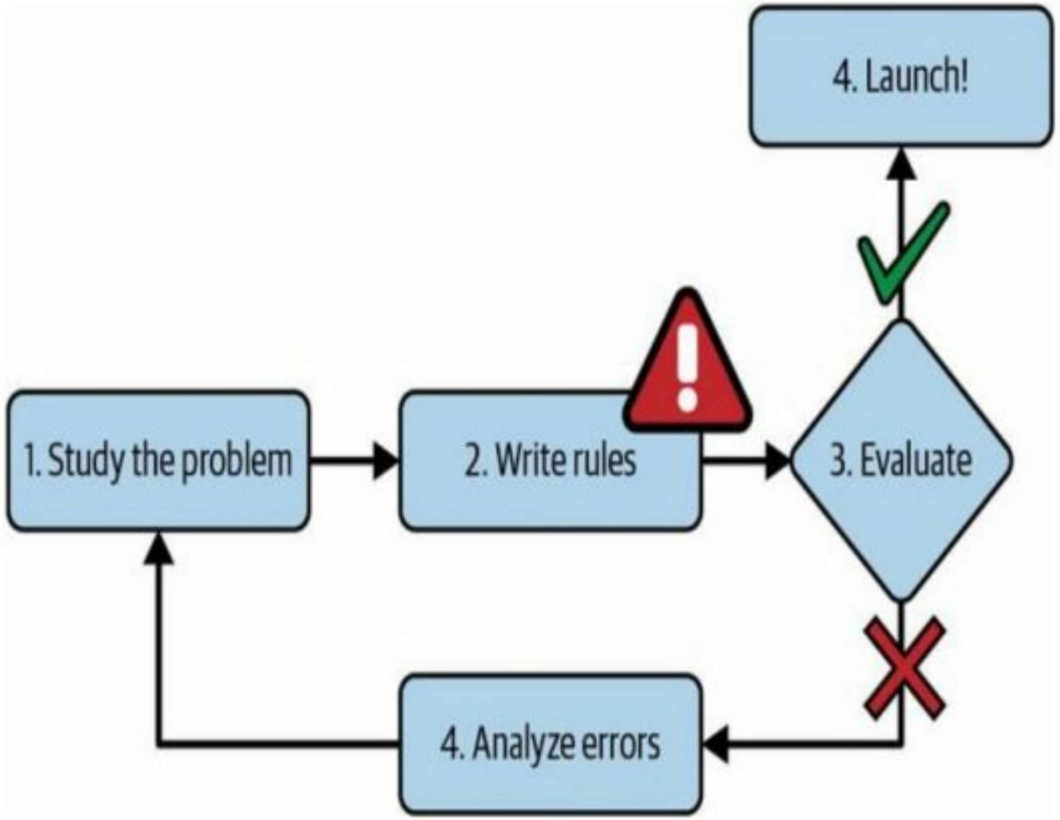


Figure 1-1. The traditional approach

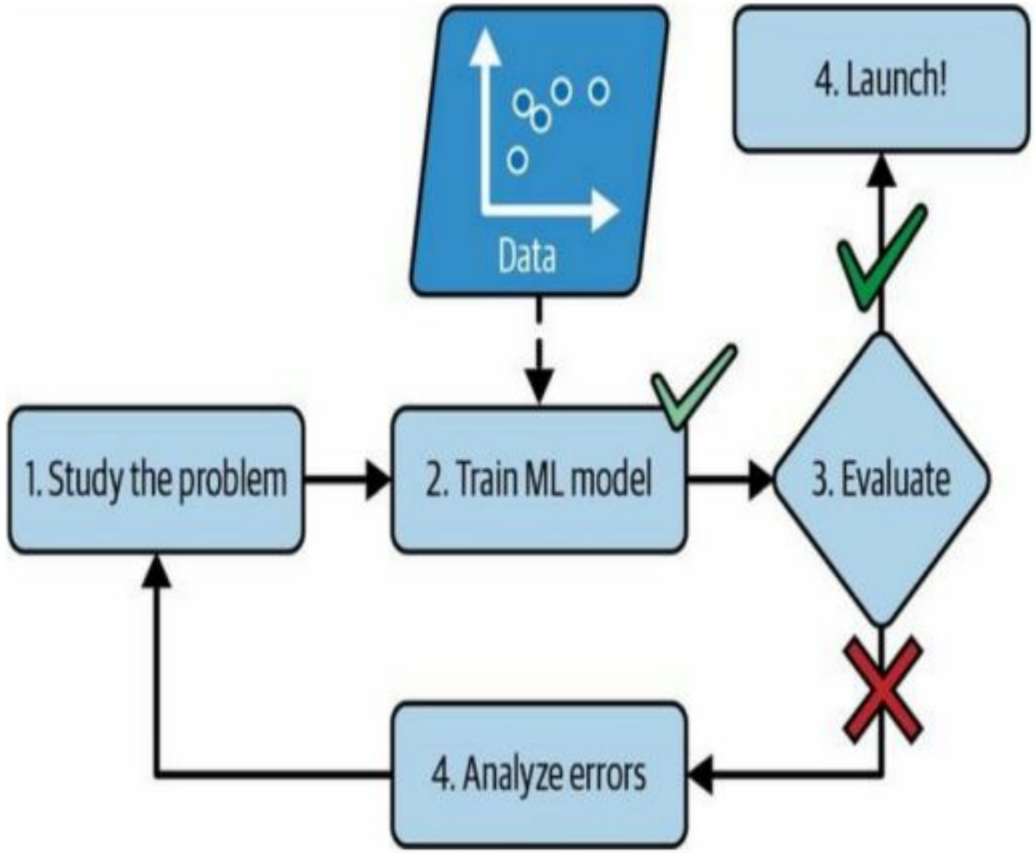
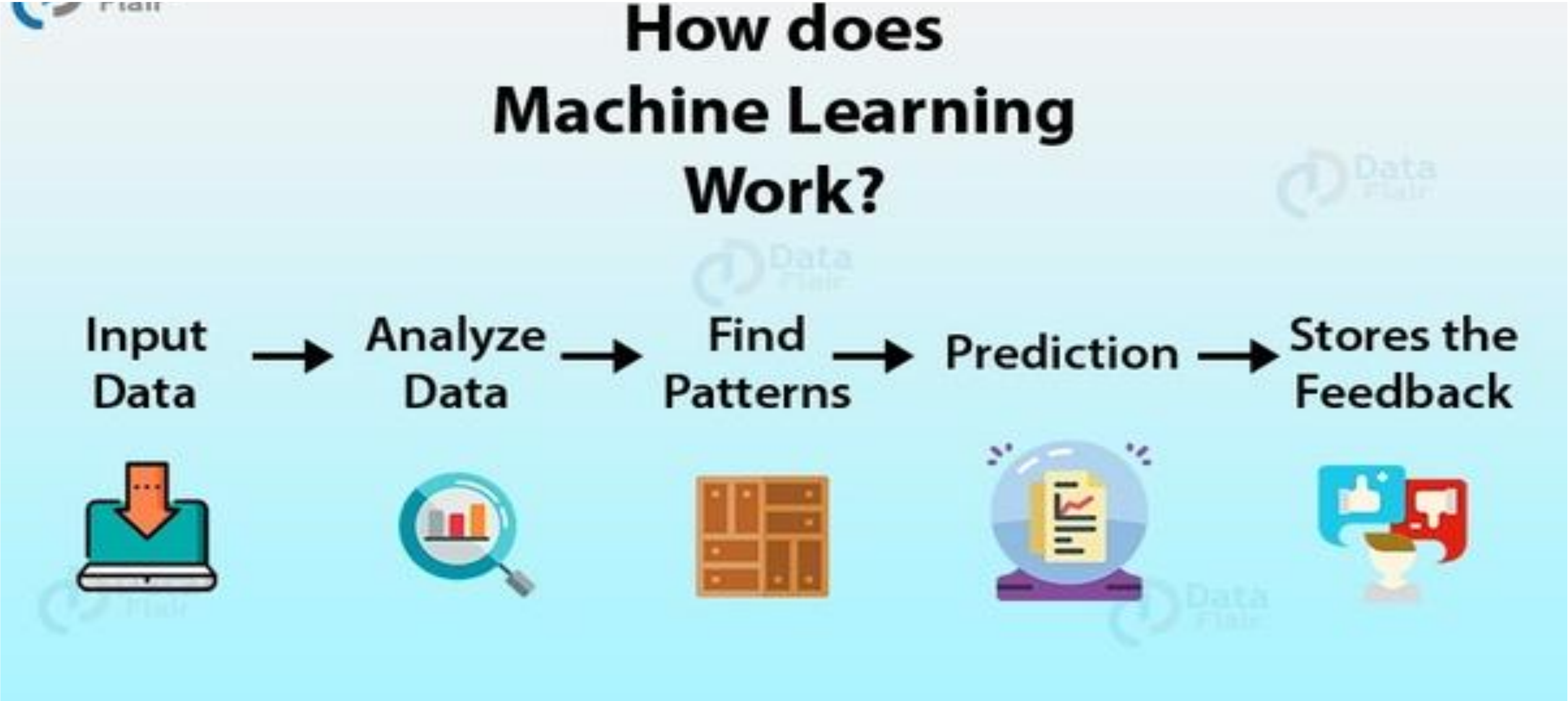
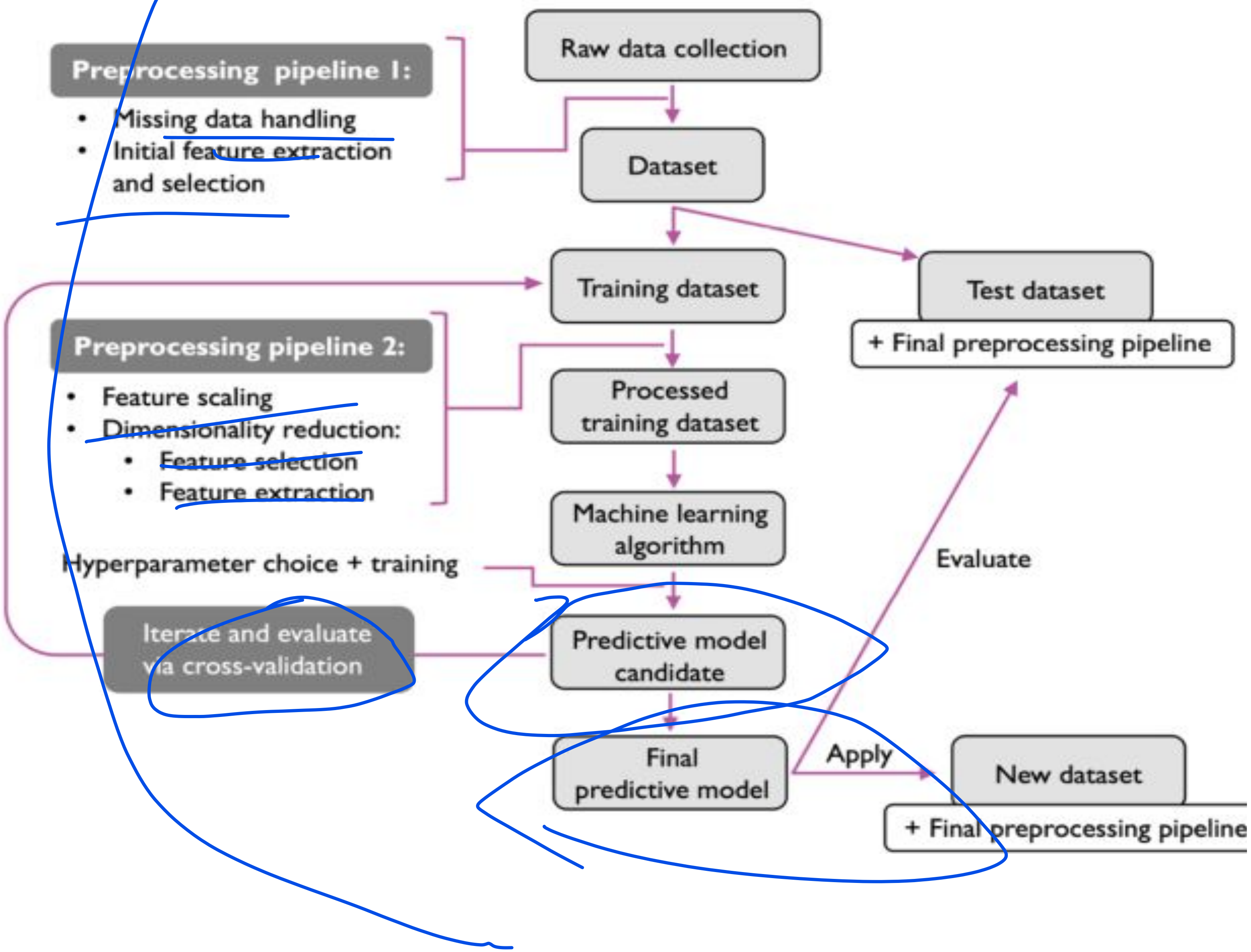
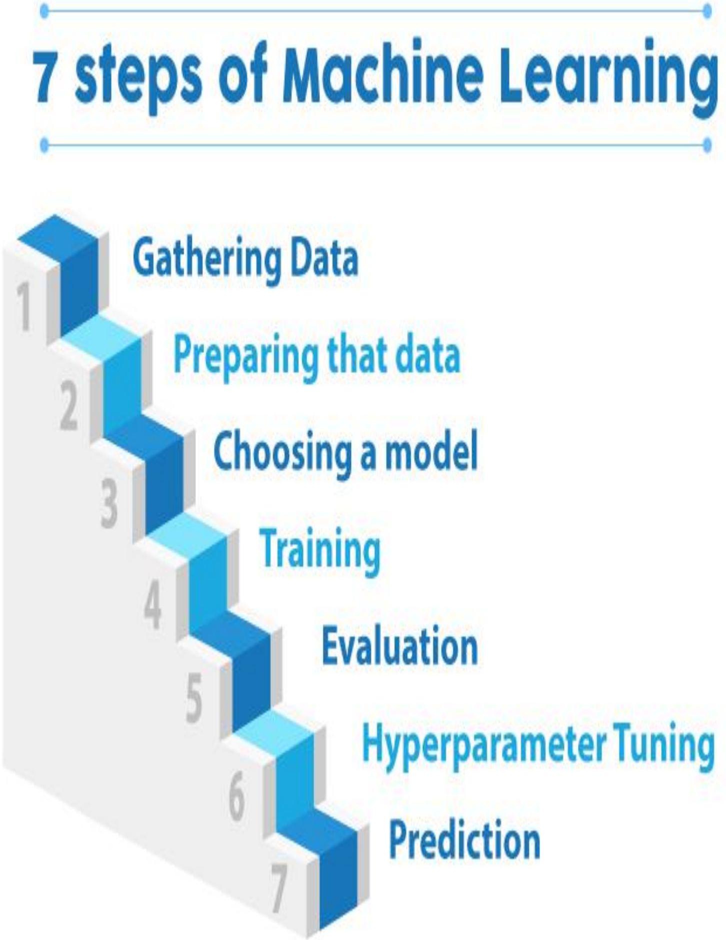


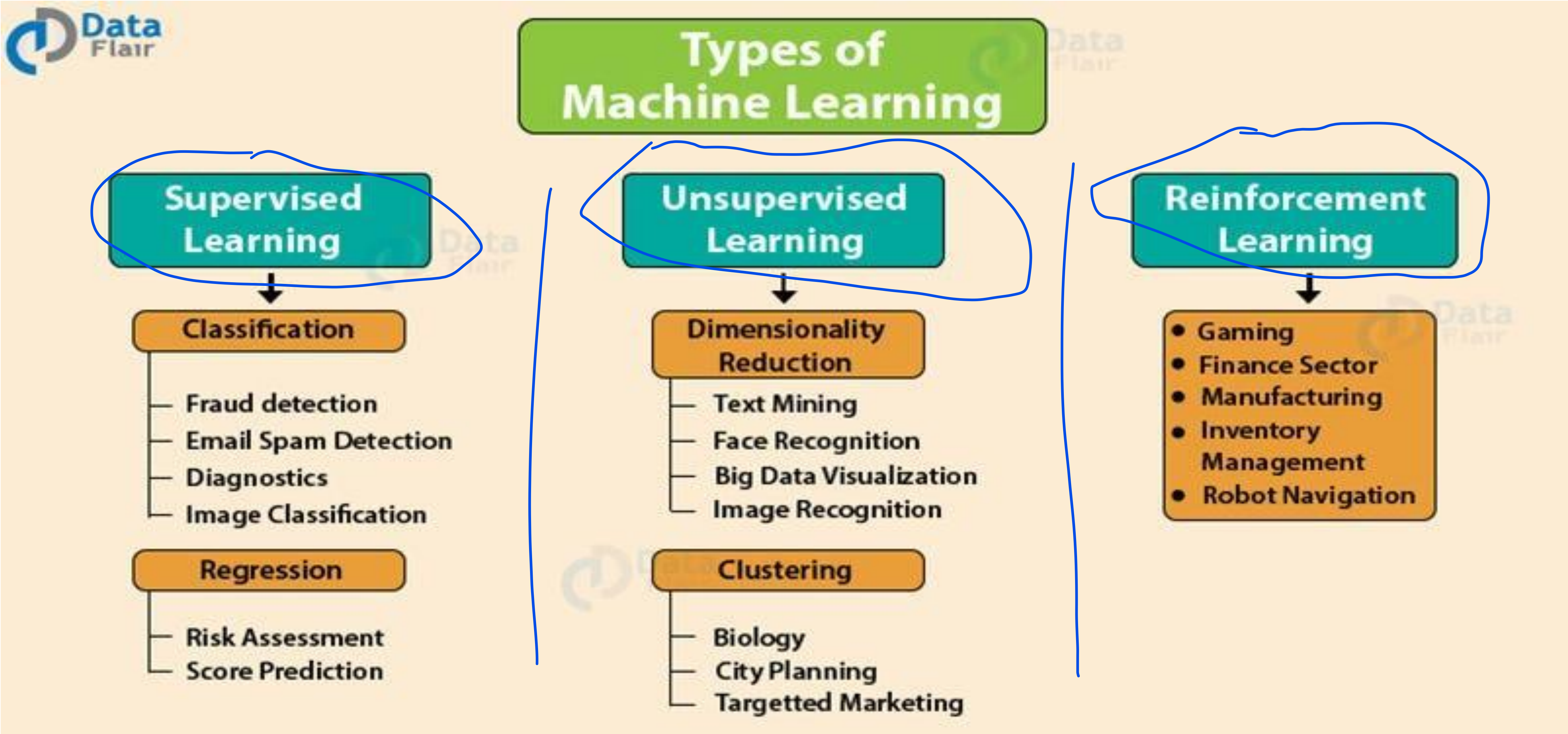
Figure 1-2. The machine learning approach


How does
Machine Learning
Work?
Input
Data
Analyze
Data
Find
Patterns
Prediction
Stores the
Feedback

Workflow of ML






Data Flair
Types of Machine Learning
Supervised Learning
Classification
Fraud detection
Email Spam Detection
Diagnostics
Image Classification
Regression
Risk Assessment
Score Prediction
Unsupervised Learning
Dimensionality Reduction
Text Mining
Face Recognition
Big Data Visualization
Image Recognition
Clustering
Biology
City Planning
Targetted Marketing
Reinforcement Learning
Gaming
Finance Sector
Manufacturing
Inventory Management
Robot Navigation

Supervised learning

- In supervised learning, the training set you feed to the algorithm includes the desired solutions, called labels (Figure 1-5).

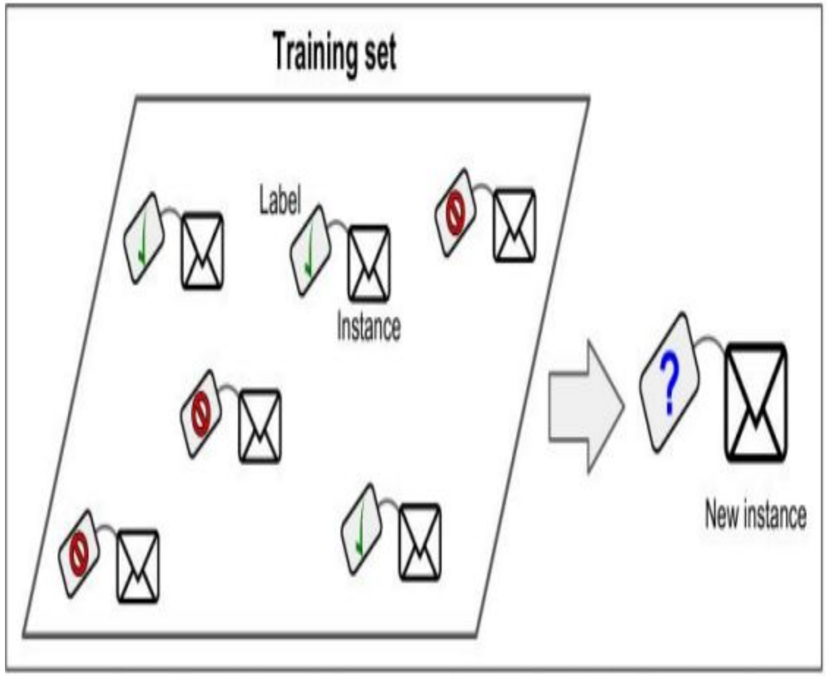


Figure 1-5. A labeled training set for supervised learning (e.g., spam classification)

Label or Target: In supervised learning, the output or the variable that the model is trying to predict based on the **input features.**

Medical Diagnosis:

- Input Features (X): Patient's medical history, symptoms, and test results.
- Label or Target (Y): The diagnosis or disease classification (e.g., "diabetes" or "no diabetes").

Stock Price Prediction:

- Input Features (X): Historical stock prices, trading volume, and economic indicators.
- Label or Target (Y): The future stock price.

Image Classification:

- **Input Features (X): Pixel values of an image.**
- **Label or Target (Y): The object or category present in the image (e.g., "cat," "dog," or "car").**

Sentiment Analysis:

- **Input Features (X): Text or reviews.**
- **Label or Target (Y): Sentiment label (e.g., "positive," "negative," or "neutral").**

Credit Risk Assessment:

- **Input Features (X): Financial data of loan applicants.**
- **Label or Target (Y): Whether the applicant is likely to default on the loan (e.g., "high risk" or "low risk").**

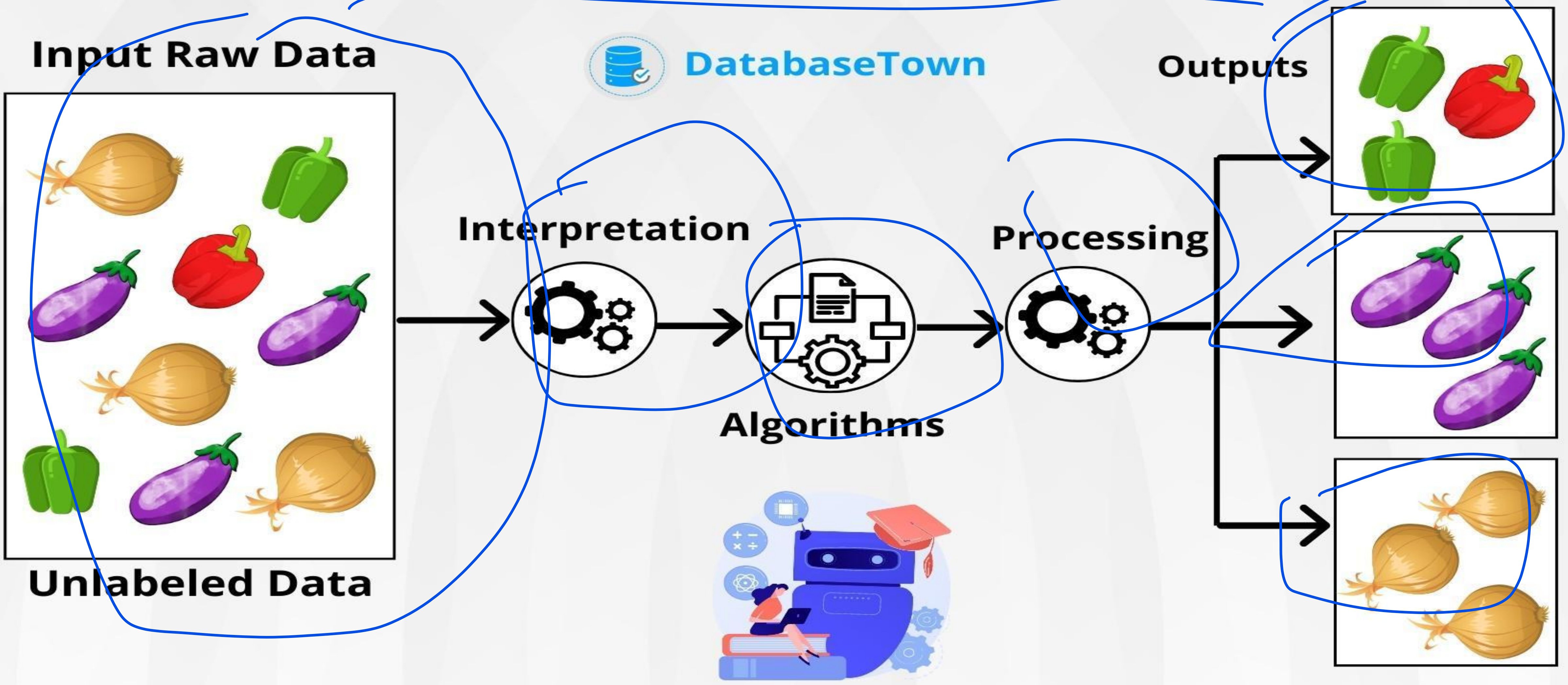
Unsupervised learning

- Unsupervised learning is a machine learning paradigm where the algorithm is trained on data without explicit supervision or labeled outcomes. In other words, it involves finding patterns, structures, or relationships within the data without any predefined target variable to predict.

Clustering
— K-Means
— DBSCAN
— Hierarchical Cluster Analysis (HCA)
• Anomaly detection and novelty detection
— One-class SVM
— Isolation Forest

UNSUPERVISED LEARNING

Unsupervised learning is a type of machine learning where the algorithm learns from unlabeled data without any predefined outputs or target variables.



Semi-supervised learning (SSL)

semi-supervised learning (SSL) is a machine learning technique that uses a small portion of labeled data and lots of unlabeled data to train a predictive model.

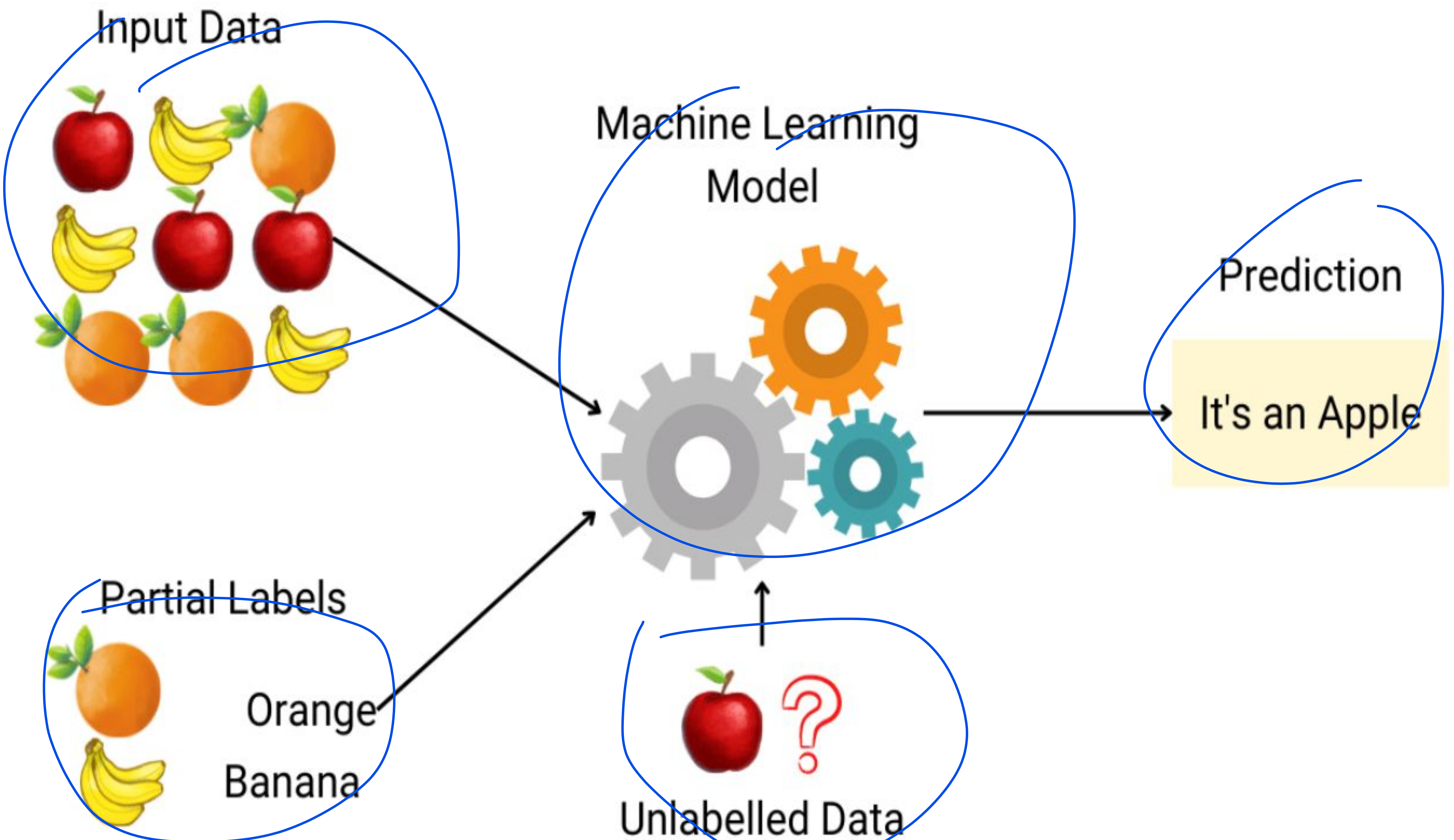
Some semi-supervised learning techniques, such as those based on generative models or adversarial training, can introduce additional complexity to the model architecture and training process.

Example:

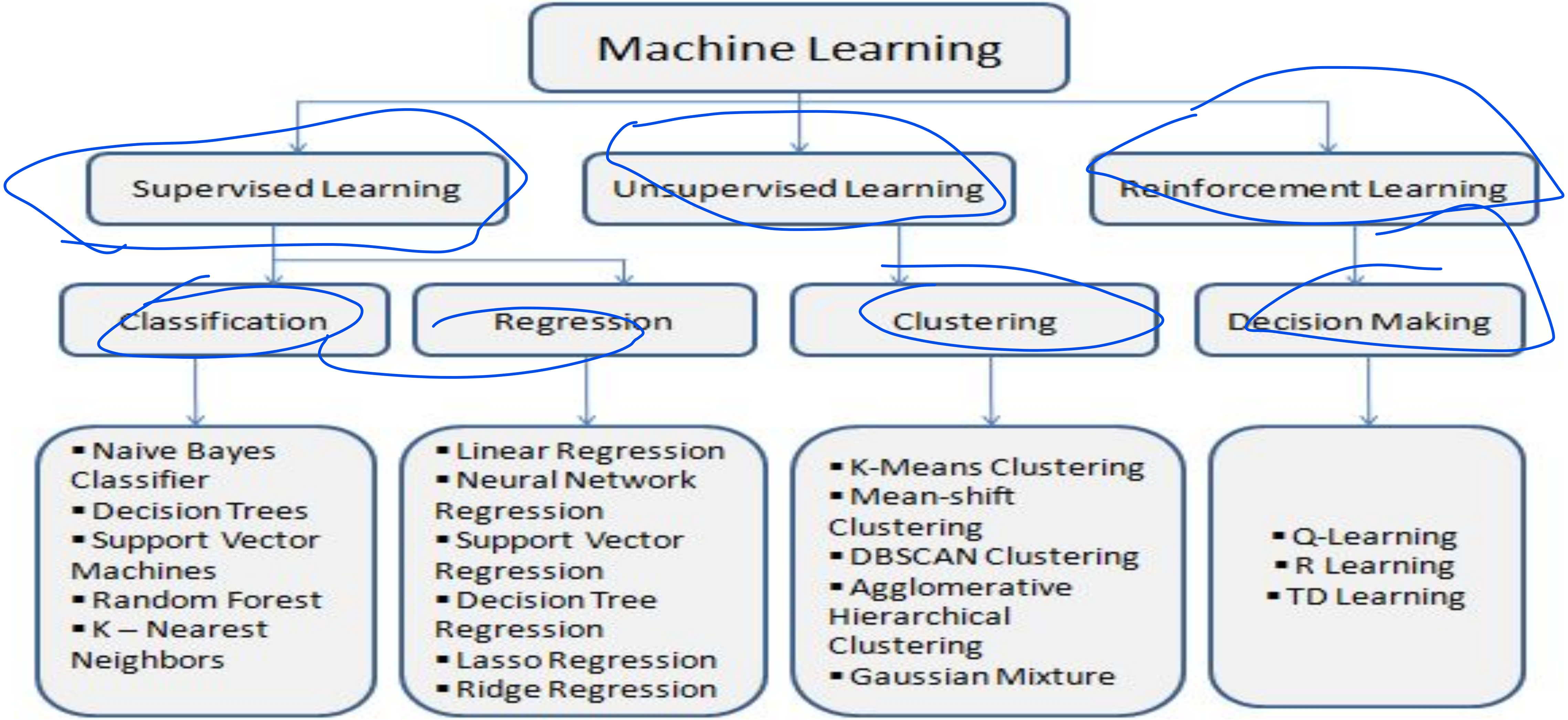
Speech recognition - Facebook has successfully applied semi-supervised learning to its speech recognition models

Web content classification - With SSL, Google Search finds content that is most relevant to a particular user query

Text document classification - LSTM


Input Data
Machine Learning
Model
Prediction
It's an Apple
Partial Labels
Orange
Banana
Unlabelled Data

Algorithms in ML



DESIGNING A LEARNING SYSTEM

the basic design issues and approaches to machine learning

Learning to Play Checkers

- T: Play checkers
- P: Percent of games won in world tournament
- What experience?
- What exactly should be learned?
- How shall it be represented?
- What specific algorithm to learn it?

1. Choosing the Training Experience

Type of Training Experience

- Direct or indirect?
- Teacher or not?

A problem: is training experience representative of performance goal?

Choose the Target Function

- $ChooseMove : Board \rightarrow Move$??
- $V : Board \rightarrow \Re$??
- ...

Possible Definition for Target Function V

- if b is a final board state that is won, then $V(b) = 100$
- if b is a final board state that is lost, then $V(b) = -100$
- if b is a final board state that is drawn, then $V(b) = 0$
- if b is a not a final state in the game, then $V(b) = V(b')$, where b' is the best final board state that can be achieved starting from b and playing optimally until the end of the game.

This gives correct values, but is not operational

Choose Representation for Target Function

- collection of rules?
- neural network ?
- polynomial function of board features?
- ...

- x_1: the number of black pieces on the board
- x_2: the number of red pieces on the board
- x_3: the number of black kings on the board
- x_4: the number of red kings on the board

- x_5: the number of black pieces threatened by red (i.e., which can be captured on red's next turn)
- x_6: the number of red pieces threatened by black

Thus, our learning program will represent $\hat{V}(b)$ as a linear function of the form

$$\hat{V}(b) = w_0 + w_1x_1 + w_2x_2 + w_3x_3 + w_4x_4 + w_5x_5 + w_6x_6$$

Partial design of a checkers learning program:

- Task T: playing checkers
- Performance measure P: percent of games won in the world tournament
- Training experience E: games played against itself
- *Target function:* $V: Board \rightarrow \Re$
- *Target function representation*

$$\hat{V}(b) = w_0 + w_1x_1 + w_2x_2 + w_3x_3 + w_4x_4 + w_5x_5 + w_6x_6$$

Estimating Training Values

Obtaining Training Examples

- $V(b)$: the true target function
- $\hat{V}(b)$: the learned function
- $V_{train}(b)$: the training value

One rule for estimating training values:

- $V_{train}(b) \leftarrow \hat{V}(Successor(b))$

Adjusting The Weights

Using least mean squares Rule

$$E \equiv \sum_{\langle b, V_{train}(b)\rangle \in \ training\ examples} (V_{train}(b) - \hat{V}(b))^2$$

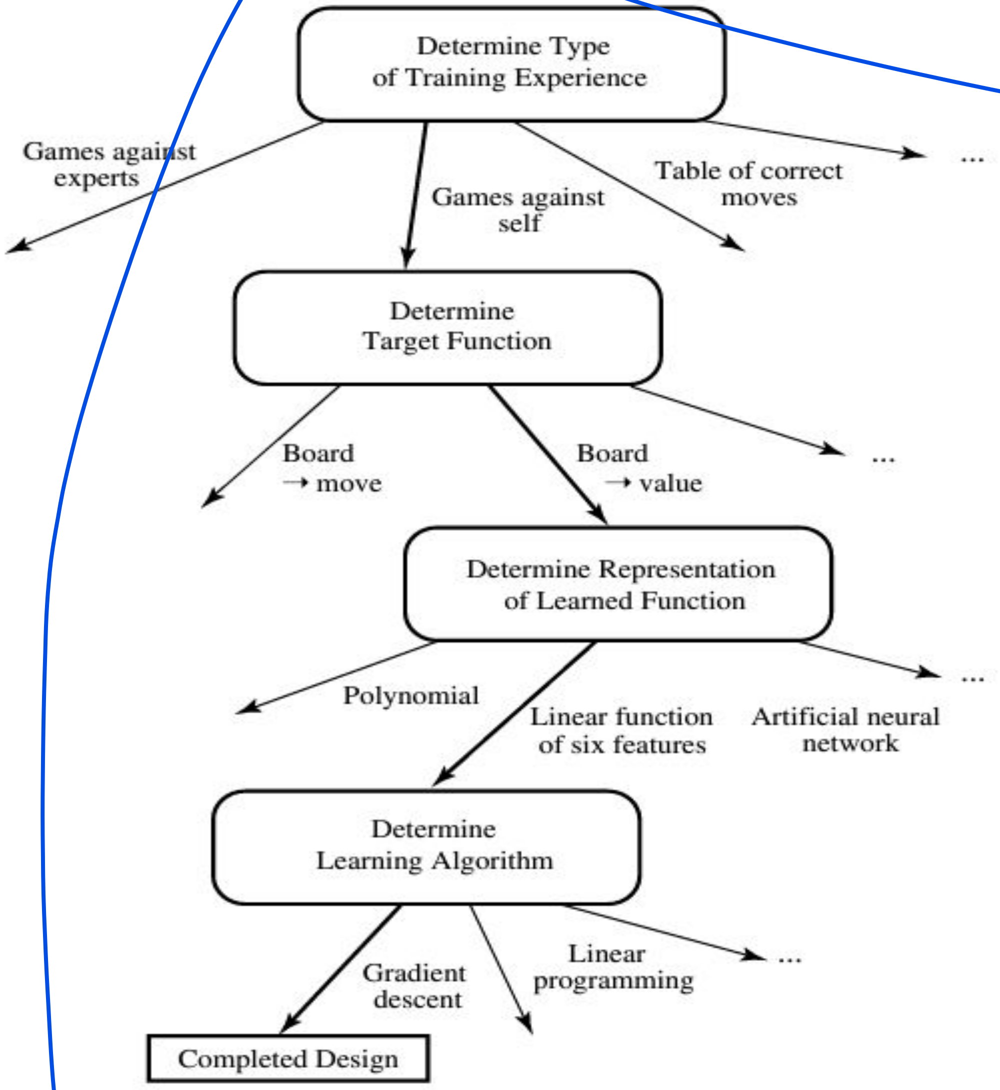
LMS weight update rule.

For each training example $\langle b, V_{train}(b)\rangle$

- Use the current weights to calculate $\hat{V}(b)$
- For each weight w_i, update it as

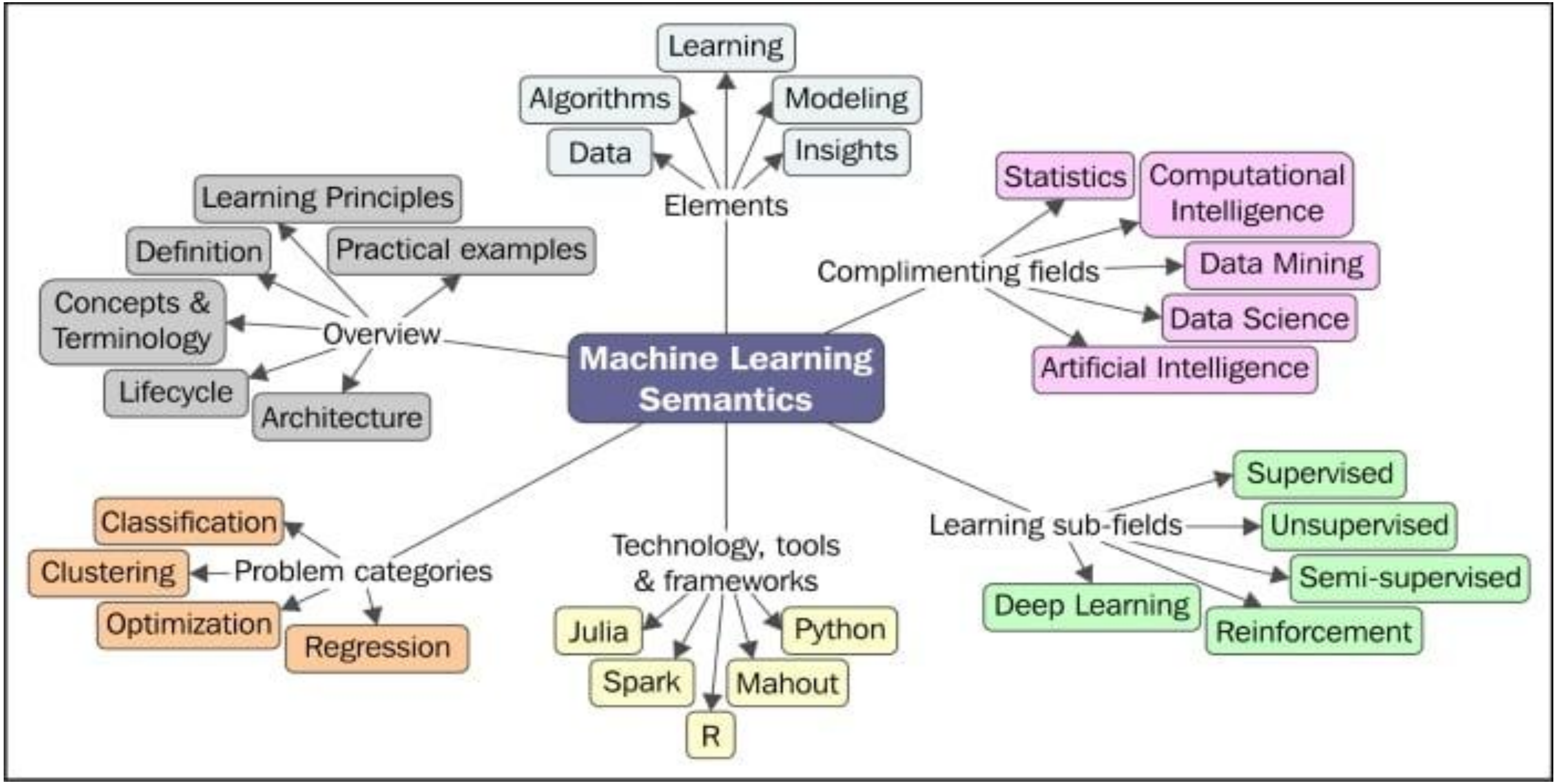
$$w_i \leftarrow w_i + \eta\ (V_{train}(b) - \hat{V}(b))\ x_i$$

Final Design

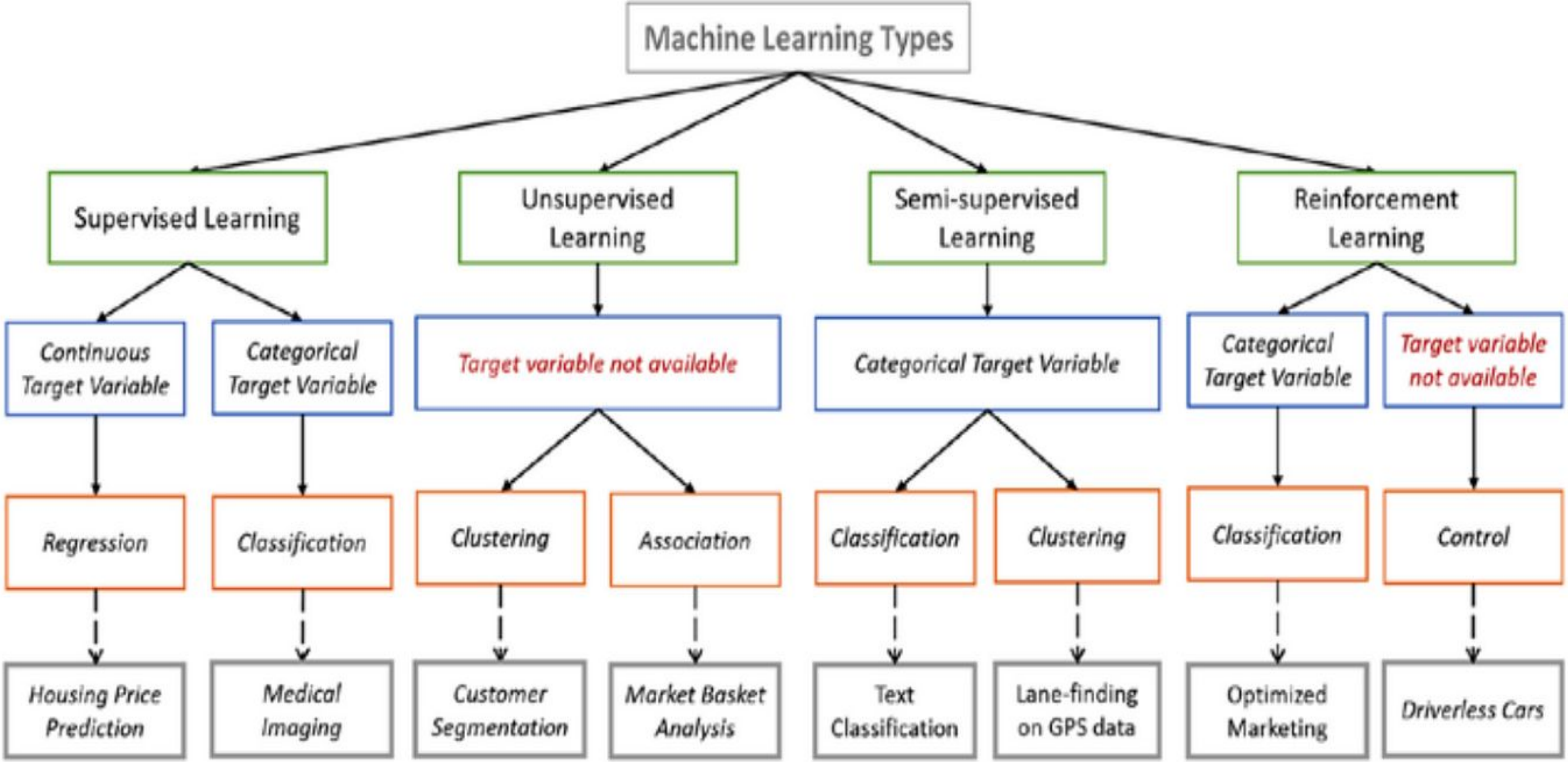


Some Issues in Machine Learning

- What algorithms can approximate functions well (and when)?
- How does number of training examples influence accuracy?
- How does complexity of hypothesis representation impact it?
- How does noisy data influence accuracy?
- What are the theoretical limits of learnability?
- How can prior knowledge of learner help?
- What clues can we get from biological learning systems?
- How can systems alter their own representations?



Machine Learning Semantics
Elements
Learning
Algorithms
Modeling
Data
Insights
Overview
Learning Principles
Definition
Practical examples
Concepts & Terminology
Lifecycle
Architecture
Complimenting fields
Statistics
Computational Intelligence
Data Mining
Data Science
Artificial Intelligence
Problem categories
Classification
Clustering
Optimization
Regression
Technology, tools & frameworks
Julia
Python
Spark
Mahout
R
Learning sub-fields
Supervised
Unsupervised
Semi-supervised
Deep Learning
Reinforcement


Machine Learning Types
Supervised Learning
Unsupervised Learning
Semi-supervised Learning
Reinforcement Learning
Continuous Target Variable
Categorical Target Variable
Target variable not available
Categorical Target Variable
Categorical Target Variable
Target variable not available
Regression
Classification
Clustering
Association
Classification
Clustering
Classification
Control
Housing Price Prediction
Medical Imaging
Customer Segmentation
Market Basket Analysis
Text Classification
Lane-finding on GPS data
Optimized Marketing
Driverless Cars

Example	*Sky*	*AirTemp*	*Humidity*	*Wind*	*Water*	*Forecast*	*EnjoySport*
1	Sunny	Warm	Normal	Strong	Warm	Same	Yes
2	Sunny	Warm	High	Strong	Warm	Same	Yes
3	Rainy	Cold	High	Strong	Warm	Change	No
4	Sunny	Warm	High	Strong	Cool	Change	Yes

Instance	*Sky*	*AirTemp*	*Humidity*	*Wind*	*Water*	*Forecast*	*EnjoySport*
A	Sunny	Warm	Normal	Strong	Cool	Change	?
B	Rainy	Cold	Normal	Light	Warm	Same	?
C	Sunny	Warm	Normal	Light	Warm	Same	?
D	Sunny	Cold	Normal	Strong	Warm	Same	?