Neural Networks and Deep Learning

Dr. Jerome J. Braun

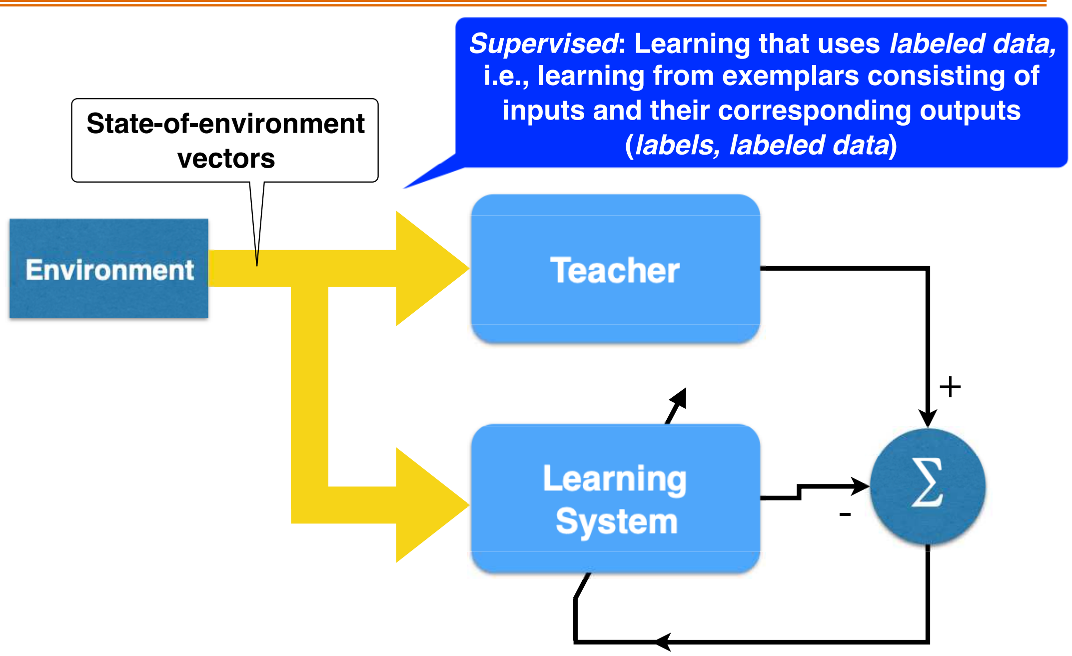
This Lecture:
Introduction — Part 3

Course: Neural Networks and Deep Learning
IE 7615

This Lecture

- Recap
- Learning — example
- Feature-space issues
- Decision boundary
- Multiple classes

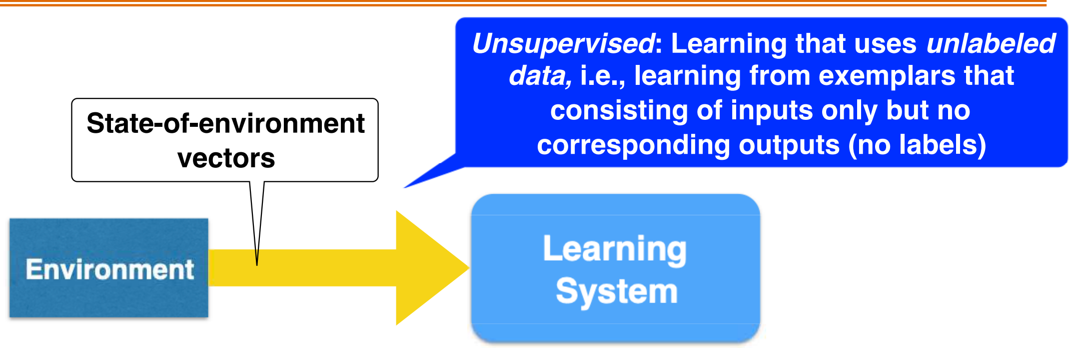
Supervised Learning (Recap)



Supervised Learning (Recap)

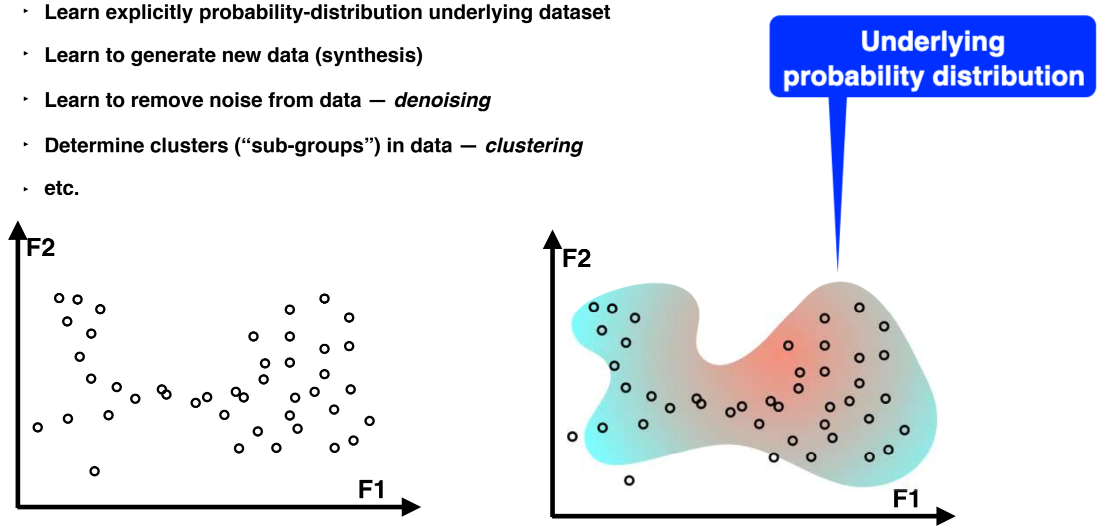
- **Training data consist of inputs and outputs**
  - **Training set**
- **Previously unseen (novel) input to be categorized**
  - **Test set**
  - **Categorize/classify = determine output given input**
- **Output types**
  - **Discrete (categorical) — classification task**
  - **Continuous — regression task**

Unsupervised Learning (Recap)

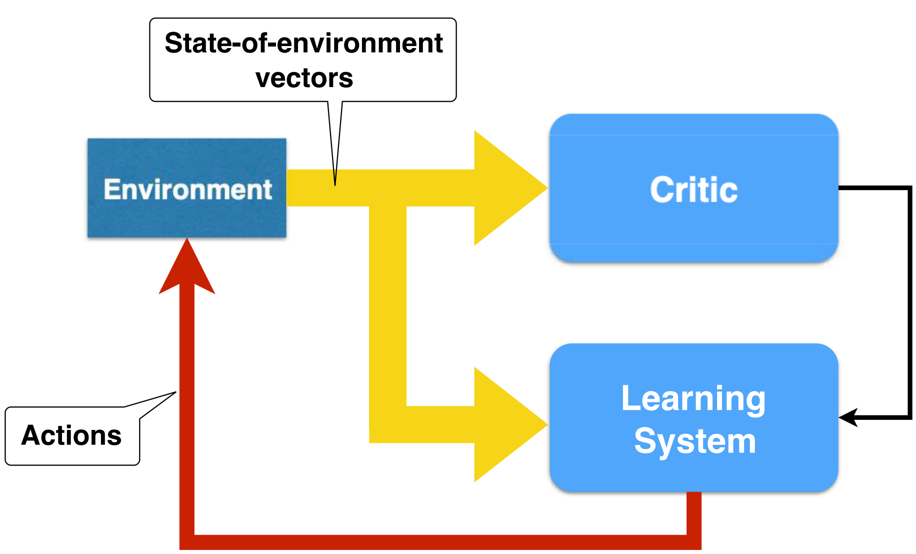


Unsupervised Learning Tasks (Recap)

- Training set contains data without labels
- Learning — from unlabeled data, learn some useful characteristics of data or process that underlies unlabeled data
  - Learn explicitly probability-distribution underlying dataset
  - Learn to generate new data (synthesis)
  - Learn to remove noise from data — *denoising*
  - Determine clusters ("sub-groups") in data — *clustering*
  - etc.



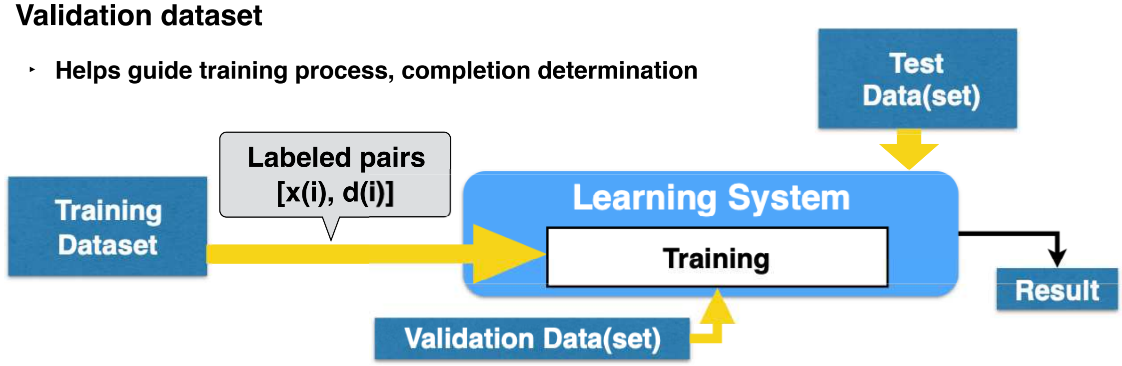
Reinforcement Learning (Recap)



Training, Validation, Testing (Recap)

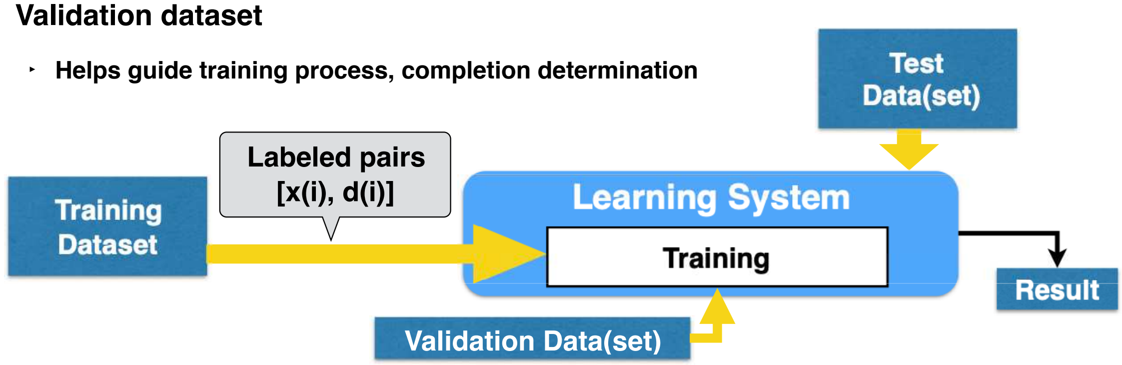
In supervised learning

- Training — learning phase
  - Training dataset — collection of labeled (annotated) exemplars
- Testing — operation of trained system
  - Test dataset — unannotated data to categorize (classify, recognize)
- Validation dataset
  - Helps guide training process, completion determination



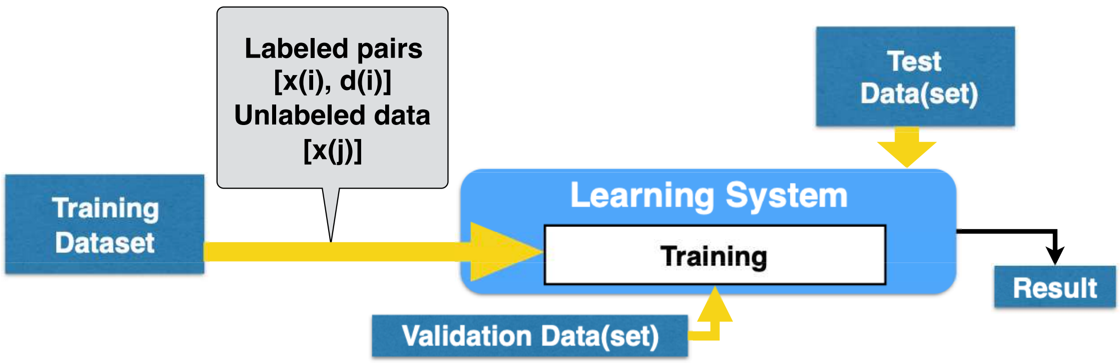
Supervised Learning (Recap)

- Training — learning phase
  - Training dataset — collection of labeled (annotated) exemplars
- Testing — operation of trained system
  - Test dataset — data to classify (categorize, recognize)
    - Labels used ONLY for performance assessment (not for classification)
- Validation dataset
  - Helps guide training process, completion determination

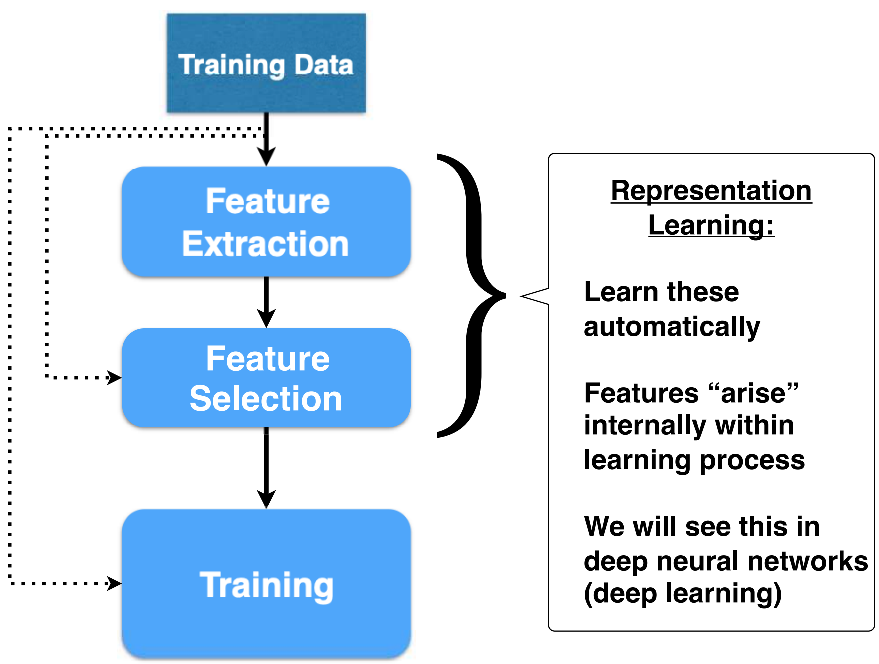


Semisupervised Learning (Recap)

- Some training data annotated
  - Have category labels
- More training data available — but no labels
  - E.g., large collection of images, only some annotated



Features (Recap)



This Lecture

- Recap
- Learning — example
- Feature-space issues
- Decision boundary
- Multiple classes

Hypothetical Example

- Automated preserves-making facility
  - Cherry preserves and grape preserves
- Video camera monitoring conveyor belt
  - Split fruits into two separate follow-up production lines
-

Cherries

Grapes

[Images credit: Wikipedia, Wikipedia Commons]

Distinguishable Characteristics

Cherries

Grapes

[Images credit: Wikipedia, Wikipedia Commons]

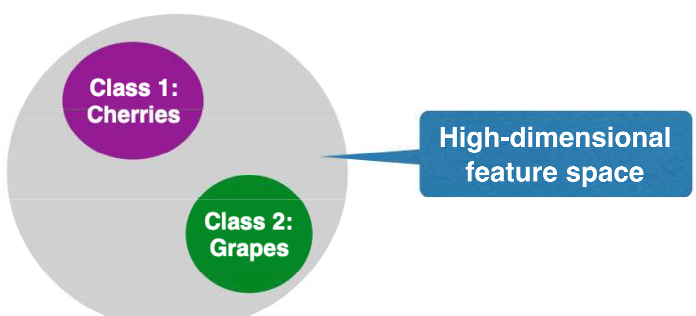
- **Many distinguishable characteristics not available to camera, e.g., physical and chemical characteristics**
- **Potential image characteristics for “cherry/grape problem”**
  - **Brightness (or color)**
  - **Size**
  - **Shape (e.g., circular or not)**
  - **…**

Deep neural networks often operate on raw inputs (e.g., images) directly
- **Without human-engineered features**
- **Features arise (emerge) within network (*representation learning*)**

(As we will see when we study deep networks)

Ideal vs. Realistic Feature Space

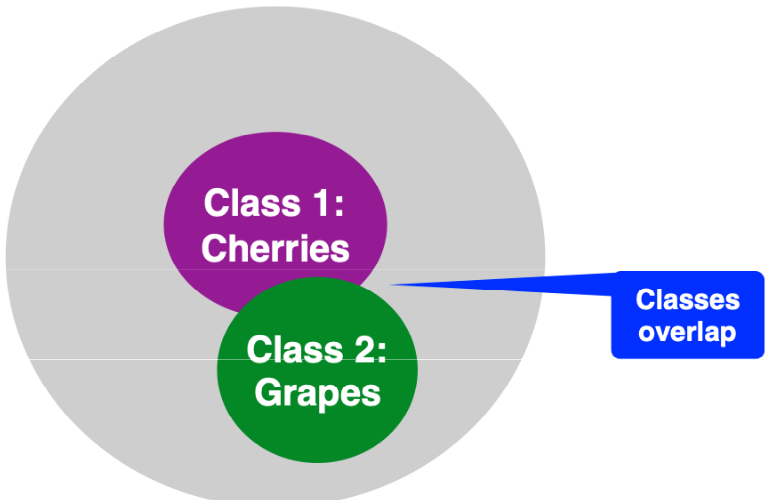
- **If features represented entire characteristics, feature space would be separable**
  - **High-dimensional feature space**



- **But many features not accessible to available sensing modality such as video camera (image domain)**

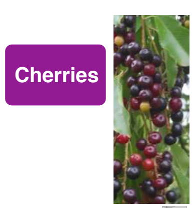
Ideal vs. Realistic Feature Space

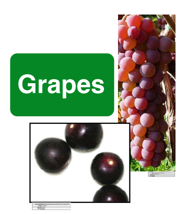
- In many subspace projections — e.g., image domain
  - Classes not separable



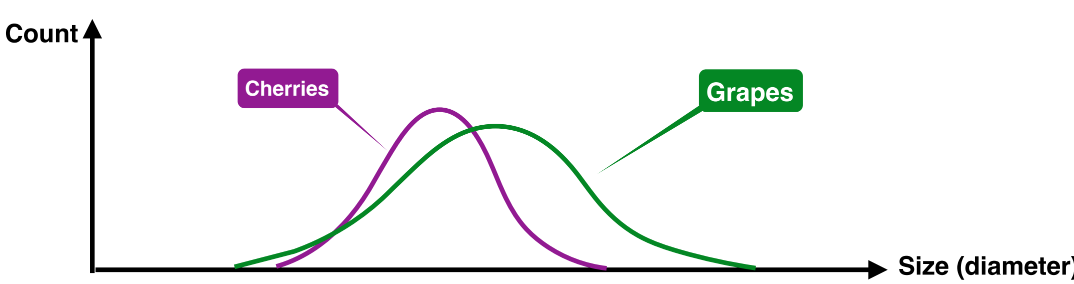
Training Set — Size Feature

- Collect many examples (exemplars) of each type
- Generate marginal distribution of size feature





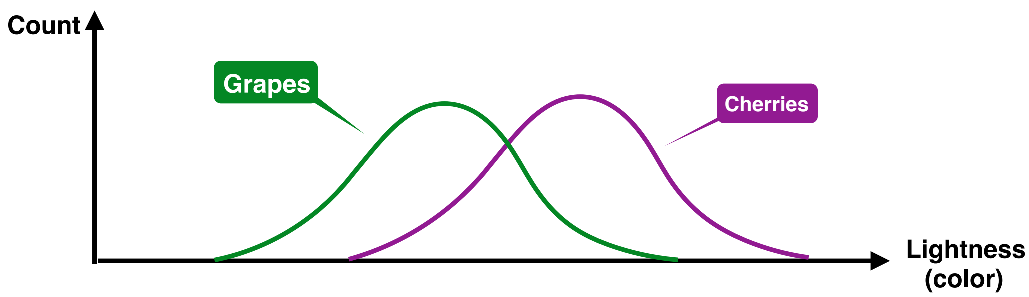
[Images credit: Wikipedia, Wikipedia Commons]



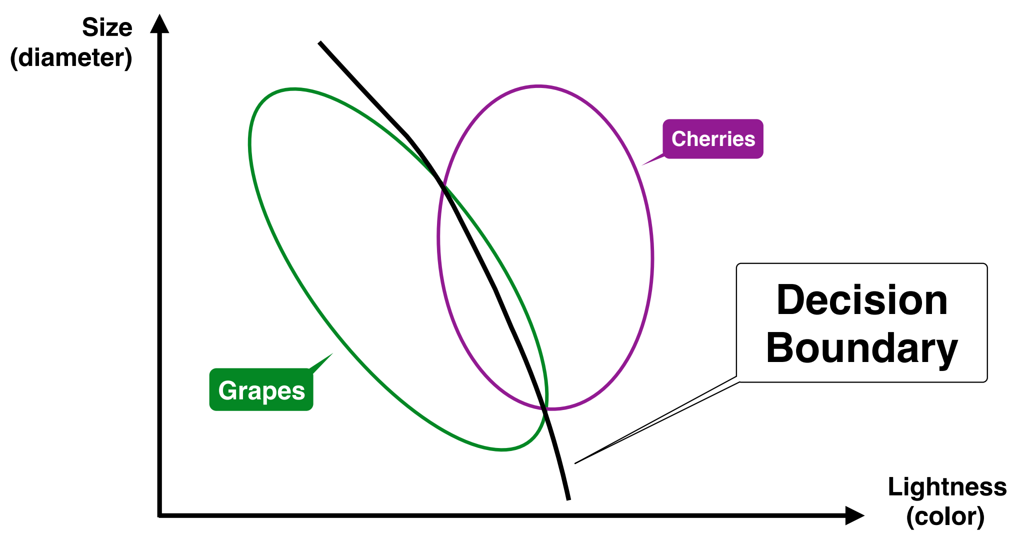
Training Set — Lightness Feature

- Collect many examples (exemplars) of each type
- Generate marginal distribution of brightness feature

[Images credit: Wikipedia, Wikipedia Commons]



Multidimensional Feature Space and Decision Boundary



- Joint distribution $[\text{color, size}]^T$
- Two features may not be sufficient for sufficiently high accuracy
- Some points (fruits) misclassified

This Lecture

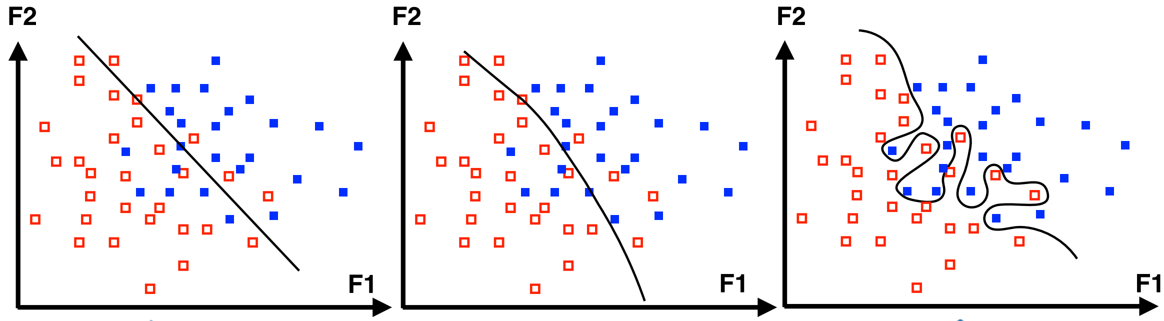
- Recap
- Learning — example
- Feature-space issues
- Decision boundary
- Multiple classes

Feature-Space Issues

- Issues associated with feature-design by humans ("feature engineering")
  - What features, and how many (curse of dimensionality)
    - Feature extraction
  - Which features will work best
    - Feature selection
  - How mitigate feature redundancy and/or correlations
    - Dimensionality reduction
- Deep neural networks often do not require human-designed features
    - Representation learning in deep-learning constructs
      - Network operates with raw input data
      - Features are learned by network
        - Features or feature-equivalents arise (emerge) within network
  - More robust than human-designed features

We will see this when we study deep networks in detail in this course

Decision Boundary Complexity

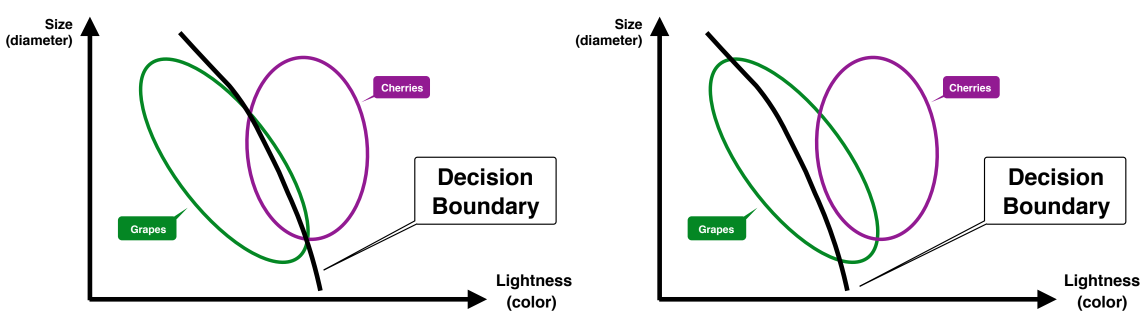


- Simple boundary — may be too simplistic
- May be inadequate for true data behavior representation
- Bias

- Balance
- Generalization — good performance on unseen (not in training set) data

- Complex boundary
- May place excessive emphasis on data idiosyncrasies (noise)
- Variance

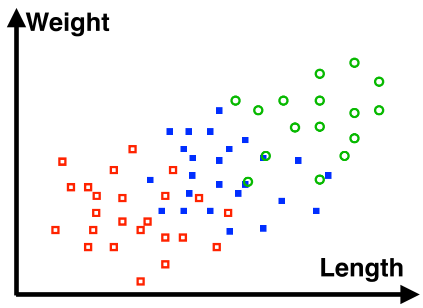
Misclassifications

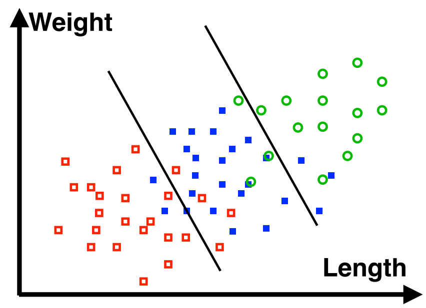


- Does it matter points of which class are misclassified ?
  - YES!
- Cost of errors may be different for one class than for another
  - E.g., misclassifying cherry as grape may be much more costly than vice-versa
    - Consider unexpected cherry-seeds in jars of grape jam (vs. a bit of grape-like taste of cherry jam)
- Decision boundary must be adjusted to account for different costs of errors — decision theory

Three-Class Supervised Learning (Toy Example)

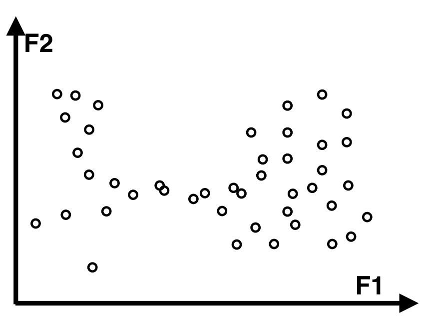
- **Example: classify metal items, based on weight and size**
  - **Wrenches — generally longer and heavy**
  - **Bolts — generally longer or medium-length and light-weight or medium-weight**
  - **Nuts — short (small) and light-weight**
- **Training set — weight and length data of labeled items**
  - **Label for each item — e.g., W or B or N ("Wrench", "Bolt", "Nut")**
- **Train classifier — learn to recognize Wrench vs. Bolt vs. Nut**

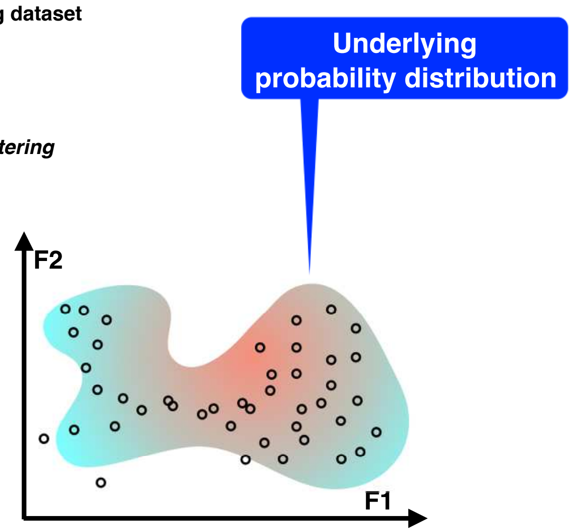




Unsupervised Learning Tasks (Recap)

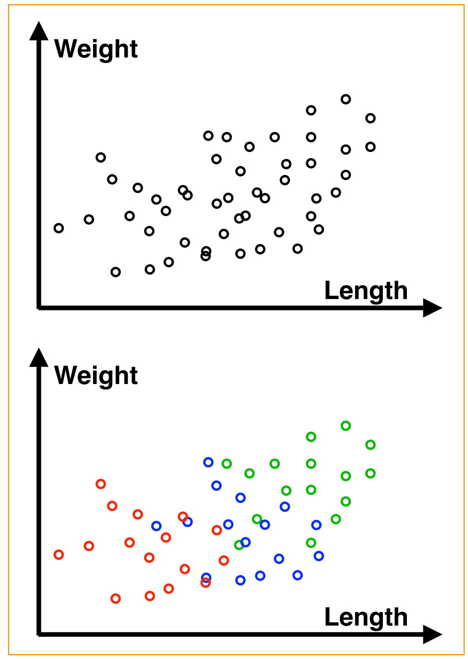
- **Training set contains data without labels**
- **Learning — from unlabeled data, learn some useful characteristics of data or process that underlies unlabeled data**
  - **Learn explicitly probability-distribution underlying dataset**
  - **Learn to generate new data (synthesis)**
  - **Learn to remove noise from data — *denoising***
  - **Determine clusters ("sub-groups") in data — *clustering***
  - **etc.**





Unsupervised Learning — Clustering Task

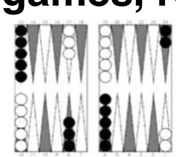
- **Training set contains data without labels**
- **Learning**
  - **Discover internal "similarities" in training data**
- **At test-time**
  - **Decide based on "similarity" of test-exemplar to learned types (categories)**
- **Example: metal-parts dataset, consisting of weight and size data (features)**

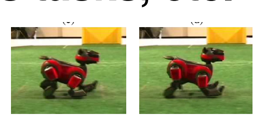


Modern unsupervised deep-learning techniques go beyond "classical" clustering

Reinforcement Learning

- **Given input or state**
- **Respond, predict output (outcome)**
- **Receive score on response**
  - **"reward/punishment"**
- **Applicable e.g., to games, robotics tasks, etc.**





[Kohl, 2004]

Determination of Learning Type — Exercise

- Learn to determine longevity (time-to-failure) of mechanical components, based on their physical features, such as thickness, material composition, etc.
- Learn to determine moves in card game
- Learn to determine driver competency/state, based on vehicle behavior in traffic
- What, and how, can be learned from large number of images showing several kinds of shirts

Questions?