Boosting
Boosting

- Well-established class of techniques applied to data classification
Boosting

- Well-established class of techniques applied to data classification

- **Weak Learner**: imprecise binary function.
Boosting

- Well-established class of techniques applied to data classification
- **Weak Learner:** imprecise binary function.
- Weighted according to classification **accuracy**.
Boosting

- Well-established class of techniques applied to data classification

- **Weak Learner**: imprecise binary function.

- Weighted according to classification **accuracy**.

- Combined into an accurate **strong classifier**.
AdaBoost
AdaBoost

Inputs:

- Training set \( \{(x_i, y_i)\}_{i=1}^{n} \) \( x \in \mathcal{X} \) \( y_i \in [-1, 1] \)
AdaBoost

**Inputs:**

- Training set \( \{(x_i, y_i)\}_{i=1}^{n} \) \( x \in \mathcal{X} \) \( y_i \in [-1, 1] \)
- Weak Learners \( \mathcal{H} \)
**AdaBoost**

**Inputs:**
- Training set \( \{(x_i, y_i)\}_{i=1}^n \) \( x \in \mathcal{X} \) \( y_i \in [-1,1] \)
- Weak Learners \( \mathcal{H} \)

**Initialization:**
- Uniform initial distribution \( D_1(i) = 1/n \)
AdaBoost

Training:

For $t = 1, \ldots, T$: 

![Diagram of training set and weak learners]
AdaBoost

Training:

For $t = 1,\ldots, T$:

- Find the current best weak learner: $h_t = \arg\min_h \sum_{i/y_i \neq h(x_i)} D_t(i)$
AdaBoost

**Training:**

For \( t = 1, \ldots, T \):

- Find the current best weak learner: 
  \[
  h_t = \arg\min_h \sum_{i : y_i \neq h(x_i)} D_t(i)
  \]
AdaBoost

Training:

For $t = 1,\ldots,T$:

- Find the current best weak learner: $h_t = \arg\min_h \sum_{i:y_i \neq h(x_i)} D_t(i)$

- Set $\alpha_t = \frac{1}{2} \log \frac{1 - \epsilon_t}{\epsilon_t}$, $\epsilon_t = \sum_{i:y_i \neq h(x_i)} D_t(i)$
AdaBoost

Training:

For $t = 1, ..., T$:

- Find the current best weak learner: $h_t = \arg\min_h \sum_{i/y_i \neq h(x_i)} D_t(i)$

- Set $\alpha_t = \frac{1}{2} \log \frac{1 - \epsilon_t}{\epsilon_t}$, $\epsilon_t = \sum_{i/y_i \neq h(x_i)} D_t(i)$
AdaBoost

Training:

For $t = 1, \ldots, T$:

• Find the current best weak learner: $h_t = \arg \min_h \sum_{i/y_i \neq h(x_i)} D_t(i)$

• Set $\alpha_t = \frac{1}{2} \log \frac{1 - \epsilon_t}{\epsilon_t}$, $\epsilon_t = \sum_{i/y_i \neq h(x_i)} D_t(i)$

• Update the weights $D_{t+1}(i) = \frac{D_t(i) \exp(-\alpha_t y_i h_t(x_i))}{Z_t}$
AdaBoost

Training:
For $t = 1, \ldots, T$:

- Find the current best weak learner: $h_t = \arg\min_h \sum_{i/y_i \neq h(x_i)} D_t(i)$

- Set $\alpha_t = \frac{1}{2} \log \frac{1 - \epsilon_t}{\epsilon_t}$, $\epsilon_t = \sum_{i/y_i \neq h(x_i)} D_t(i)$

- Update the weights $D_{t+1}(i) = \frac{D_t(i) \exp(-\alpha_t y_i h_t(x_i))}{Z_t}$
Test:

$\text{AdaBoost}$
AdaBoost

Test:

$h_1 \quad \ldots \quad h_t \quad \ldots \quad h_T$
AdaBoost

Test:

For any novel example $x \in \mathcal{X}$:
For any novel example $x \in \mathcal{X}$:

- The strong classifier returns

$$H(x) = \text{sign}\left(\sum_{t=1}^{T} \alpha_t h_t(x)\right)$$
Test:

For any novel example $x \in X$:

- The strong classifier returns

$$H(x) = \text{sign}\left(\sum_{t=1}^{T} \alpha_t h_t(x)\right)$$
Multiple Instance Learning
Multiple Instance Learning

Bags of Instances

- **Images**: collections of features (SURF, SIFT, etc.).
Multiple Instance Learning

Bags of Instances

- **Images**: collections of features (SURF, SIFT, etc.).
Multiple Instance Learning

Bags of Instances

- **Images**: collections of features (SURF, SIFT, etc.).
Multiple Instance Learning

Bags of Instances

- **Images**: collections of features (SURF, SIFT, etc.).
- **Positive instance**: feature extracted from the object.

![Diagram showing bags of instances with red and green points representing negative and positive instances, respectively.]
Multiple Instance Learning

Bags of Instances

- **Images**: collections of features (SURF, SIFT, etc.).
- **Positive instance**: feature extracted from the object.
- **Positive bag**: image with at least one positive feature.
Multiple Instance Learning

Bags of Instances

- **Images**: collections of features (SURF, SIFT, etc.).
- **Positive instance**: feature extracted from the object.
- **Positive bag**: image with at least one positive feature.

Learn to classify bags without knowing the single features labels.
MIL Ball Weak Learners
MIL Ball Weak Learners

MIL Ball: Classification

Feature Space

Instance

Auer and Ortner 2004
MIL Ball Weak Learners

MIL Ball: Classification

- **Center**: a point in the feature space.
- **Radius**: determines the weak learner tolerance.

Auer and Ortner 2004
MIL Ball Weak Learners

**MIL Ball: Classification**

- **Center**: a point in the feature space.
- **Radius**: determines the weak learner tolerance.

**Positive Classification**: non-void intersection.

---

*Auer and Ortner 2004*
Online Boosting
Online Boosting

**Initialization**

- $\mathcal{H} = \{h_1, \ldots, h_N\}$ pre-ordered set of weak learners.
- set $\alpha_n = 0 \quad \forall n \in \{1, \ldots, N\}$. 

![Diagram of Online Boosting Initialization](image-url)
Online Boosting

Training

\( \lambda \)

\( h_1 \quad h_2 \quad \ldots \quad h_N \)

\( \alpha_1 \quad \alpha_2 \quad \ldots \quad \alpha_N \)

\( I \) training sample
Online Boosting

Training

\[ h_1 \quad h_2 \quad \ldots \quad h_N \]

\[ \alpha_1 \quad \alpha_2 \quad \ldots \quad \alpha_N \]

\[ \lambda \]
Online Boosting

Training

- Learning principle $h_n \leftarrow L(h_n, I, \lambda)$. 
Online Boosting

Training

- Learning principle \( h_n \leftarrow L(h_n, I, \lambda) \).
Online Boosting

Training

- Learning principle: $h_n \leftarrow L(h_n, I, \lambda)$.
- correct: $\lambda \leftarrow \lambda/2(1 - \epsilon_n)$.
- incorrect: $\lambda \leftarrow \lambda/2\epsilon_n$. 
Online Boosting

Training

- Learning principle: \( h_n \leftarrow L(h_n, I, \lambda) \).
- correct: \( \lambda \leftarrow \lambda/2(1 - \epsilon_n) \).
- incorrect: \( \lambda \leftarrow \lambda/2\epsilon_n \).

\[ \lambda \]

\[ h_1, h_2, \ldots, h_N \]
Online Boosting

**Training**

- **Learning principle** \( h_n \leftarrow L(h_n, I, \lambda) \).
- **Correct**:
  \( \lambda \leftarrow \lambda / 2(1 - \epsilon_n) \).
- **Incorrect**:
  \( \lambda \leftarrow \lambda / 2\epsilon_n \).
- \( \alpha_n = \log \frac{1 - \epsilon_n}{\epsilon_n} \).
Online Boosting

Training

- Learning principle: $h_n \leftarrow L(h_n, I, \lambda)$.
- Correct: $\lambda \leftarrow \lambda/2(1 - \epsilon_n)$.
- Incorrect: $\lambda \leftarrow \lambda/2\epsilon_n$.
- $\alpha_n = \log \frac{1 - \epsilon_n}{\epsilon_n}$. 

\[
\begin{align*}
\alpha_1 & \quad \alpha_2 & \quad \cdots & \quad \alpha_N \\
h_1 & \quad h_2 & \quad \cdots & \quad h_N \\
\lambda
\end{align*}
\]
MIL Ball Weak Learners

MIL Ball: Learning Principle
MIL Ball Weak Learners

MIL Ball: Learning Principle

- **Accuracy**: keeps track of the MIL Ball error rate.

![Diagram of MIL Ball Learning Principle]

- **negative instance**
- **positive instance**
MIL Ball Weak Learners

MIL Ball: Learning Principle

- **Accuracy**: keeps track of the MIL Ball error rate.
- **Radius**: is updated to keep classification accuracy maximized.

- negative instance
- positive instance
Application: Own Hand Recognition
Application: Own Hand Recognition

<table>
<thead>
<tr>
<th>Why The Hand?</th>
</tr>
</thead>
</table>

Application: Own Hand Recognition

Why The Hand?

- **Humanoids**: main tool of physical exploration.
Application: Own Hand Recognition

Why The Hand?

- **Humanoids**: main tool of physical exploration.
- **Directly controllable**: easier to learn autonomously.
Learning: Data Collection

Labeling

- **MIL**: requires weak supervision.
- **Strategy**: random arm-gaze movements.
- **Positive Label**: co-occurrence of visual and motor activity.

Motors state: Moving  
Motion Detected  
Hand is (probably) in the FoV

- Imprecise but **sufficient** for MIL.
Learning: Localization
Localization

- **Feature Selection:** positive MIL Balls respond to positive features.
Learning: Localization

Localization

- **Feature Selection**: positive MIL Balls respond to positive features.
- **Cluster**: gaussian mixtures.
Learning: Localization
Application: Online Object Recognition
Online Object Recognition