Online Automatic Modulation Classification Based on Distributional Signal Representation

Xinpeng **Li**[1,2,⋆], **Z**ile **Jiang**[1,⋆], **K**ai **M**ing **Ting**[1,2], and **Y**e **Zhu**[3]

[1]State Key Laboratory for Novel Software Technology, Nanjing University, Nanjing 210023, China

[2]School of Artificial Intelligence, Nanjing University, Nanjing 210023, China

[3]School of Information Technology, Deakin University, Geelong 3125, VIC, Australia

⋆ These authors contributed equally to this work.

lixp@lamda.nju.edu.cn, zilerion99@gmail.com, tingkm@nju.edu.cn, ye.zhu@ieee.org

Background & Motivation

Automatic Modulation Classification (AMC):

- Automatically recognize the modulation format of an incoming signal so that the receiver can correctly demodulate the data.

The Deep Learning Bottleneck:

- Highly complex architectures.
- Computationally heavy, require extensive hyperparameter tuning.
- Restricted to offline batch-mode processing: fails to adapt to realistic, time-varying channel conditions (mismatched environments).

Background & Motivation

Current indirect signal representations:

- Engineered Features: Lacks flexibility, requires manual extraction.
- Image Representation: Grid conversion introduces noise and loses fine-grained information.
- Sequence Representation: Hard to extract robust temporal features.
- Combined Representation: Whether there is a synergistic effect remains poorly understood.

The problem:

- Data conversion inherently compromises classification performance.

Distributional Signal Representation

In general, a baseband signal $r(t)$ received from a wireless channel can be expressed as:

$$r(t) = s(t) * c(t) + w(t)$$

where $s(t)$ denotes the transmitted baseband signal, $c(t)$ is the channel impulse response, $w(t)$ represents the additive noise, and $*$ is a convolution operator.

The transmitted amplitude-phase modulated baseband signal $s(t)$ with N symbols can be expressed as:

$$s(t) = \sum_{n=1}^{N} a_n \exp(j\varphi_n) g(t - nT_s)$$

and

$$a_n \exp(j\varphi_n) = a_n \cos\varphi_n + j \cdot a_n \sin\varphi_n = I_n + j \cdot Q_n$$

where a_n and φ_n are the amplitude and phase of the n-th symbol; $g(t)$ signifies the pulse shaping filter; T_s is the symbol duration; and I_n and Q_n denote the in-phase (I) and quadrature (Q) components of the n-th symbol, respectively.

Distributional Signal Representation

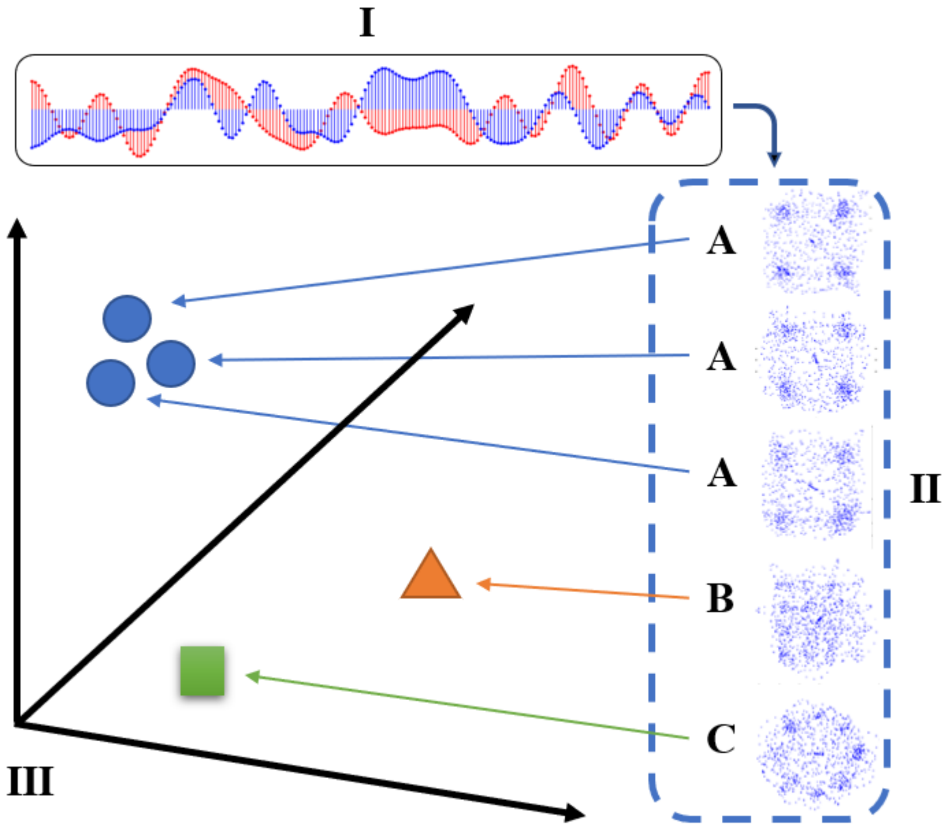
Insight:

A signal S in the I/Q space (or constellation diagram) can be treated as a set of independent and identically distributed (i.i.d.) points x drawn from an unknown probability distribution P_S, i.e., $x \sim P_S$, without loss of information.

$$AMC(S) = \operatorname{argmax}_{\alpha \in \{M_1, M_2, \cdots, M_m\}} K(P_S, P_{S_\alpha})$$

where K is a distributional kernel which measures the similarity between two distributions.

Distributional Signal Representation



Isolation Distributional Kernel (IDK):

- Injective mapping guarantee:

  Two distributions are mapped to the same point in the feature space only when they are exactly identical.

$$\| \widehat{\Phi}(P_S) - \widehat{\Phi}(P_T) \|_H = 0 \text{ if and only if } P_S = P_T$$

Proposed Method: IDK-OGD

Isolation Distributional Kernel (IDK):

- Map the signal's distribution into a high-dimensional feature space.

Online Gradient Descent (OGD):

- Perform modulation classification and online update.

Algorithm 1 IDK-OGD

Input:

m - The number of modulation formats
$\widehat{\Phi}_j$ - IDK feature mapping of modulation format j
η - Learning rate

Output:

Predicted modulation format $\widehat{c}_i$ for each signal S_i

```
Initialize weight vector w_j to 0 for j = 1, ..., m
while a new batch of signals B arrives do
    for i = 1 : |B| do
        for j = 1 : m do
            g_{i,j} = ⟨ w_j, Φ̂_j(P_{S_i}) ⟩
        end for
        ĉ_i = arg max_j g_{i,j}
    end for                                              ▷ Classification stage ends
    if ground-truth labels c_i for S_i ∈ B are available then
        for i = 1 : |B| do
            k_i = { 1,   if c_i = ĉ_i
                  { −1,  if c_i ≠ ĉ_i
            for j = 1 : m do
                w_j = w_j − η∇L(g_{i,j}; k_i) k_i Φ̂_j(P_{S_i})
                          ▷ ∇L is the gradient of the hinge loss function L
            end for
        end for
    end if                                               ▷ Online model update stage ends
end while
```

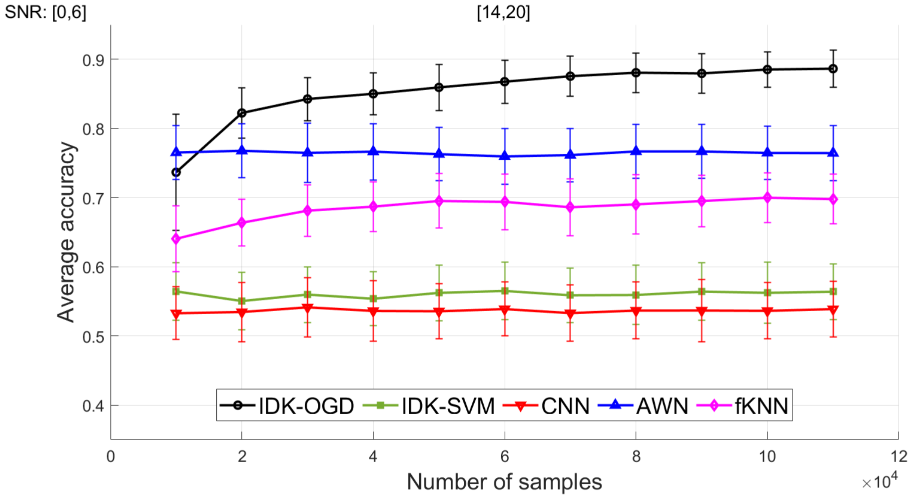
Experiments

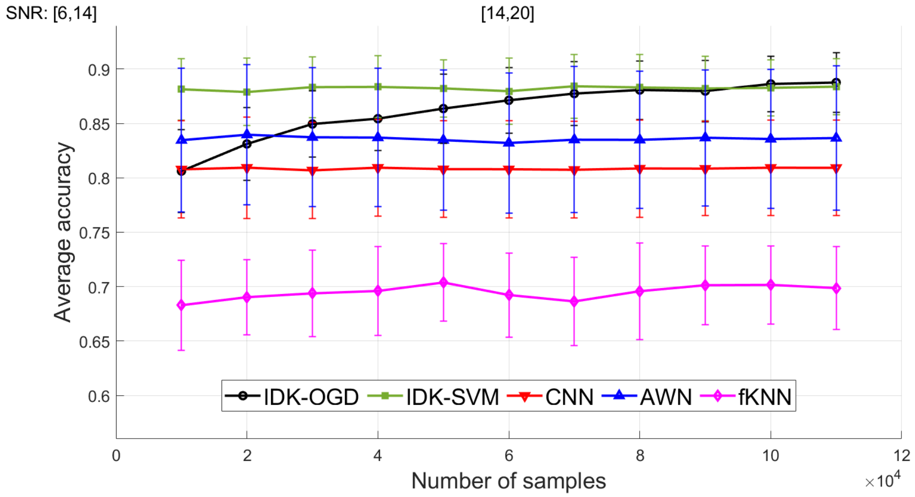
- **Datasets**:
  - Open-source RadioML2018.01A dataset and synthetic datasets.
  - 10 diverse (amplitude-phase modulated) modulation formats.
  - All classifiers are initialized on a training set and evaluated on a streaming testing set to simulate real-time data arrival.

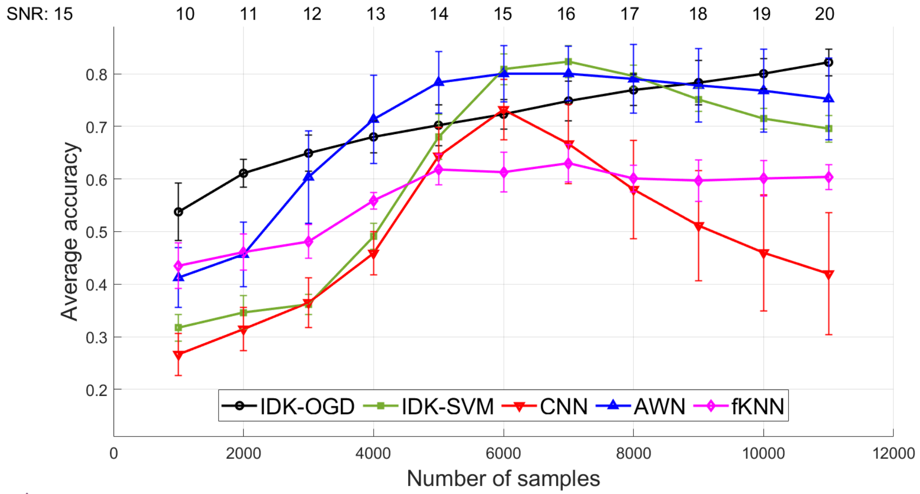
- **Time-varying channel conditions:**
  - Signal-to-Noise Ratio (SNR), phase noise, I/Q amplitude imbalance.
  - The channel conditions of the testing set deliberately and dynamically deviate from the training data.

- **Methods**:
  - Incremental online update: IDK-OGD.
  - Retraining on cumulative data: fKNN.
  - Static batch-learning: CNN, AWN, IDK-SVM.

Experiments



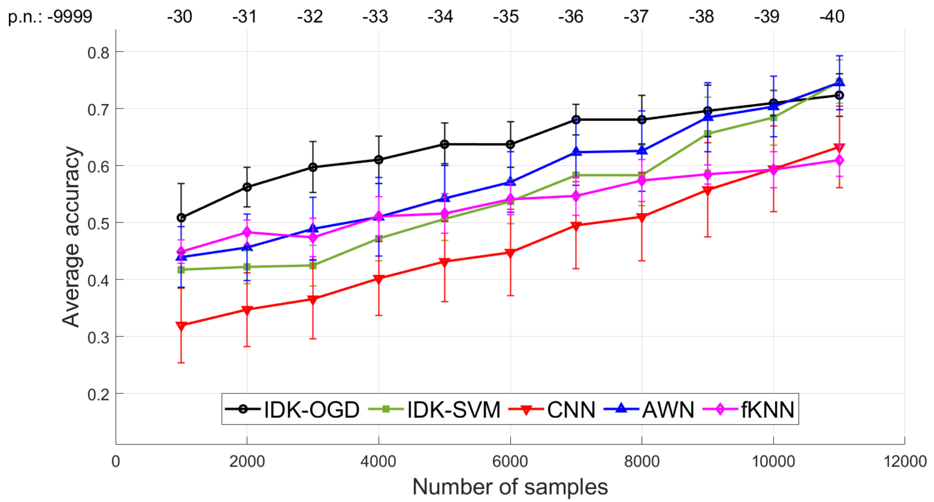


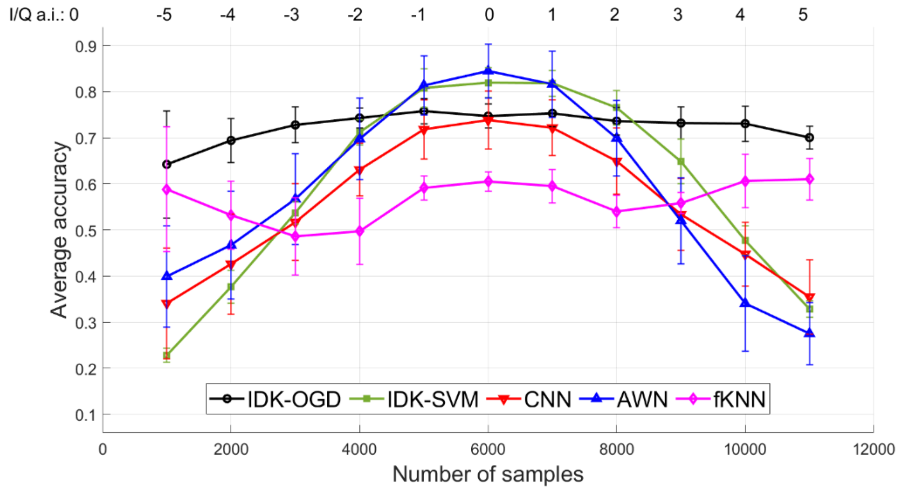


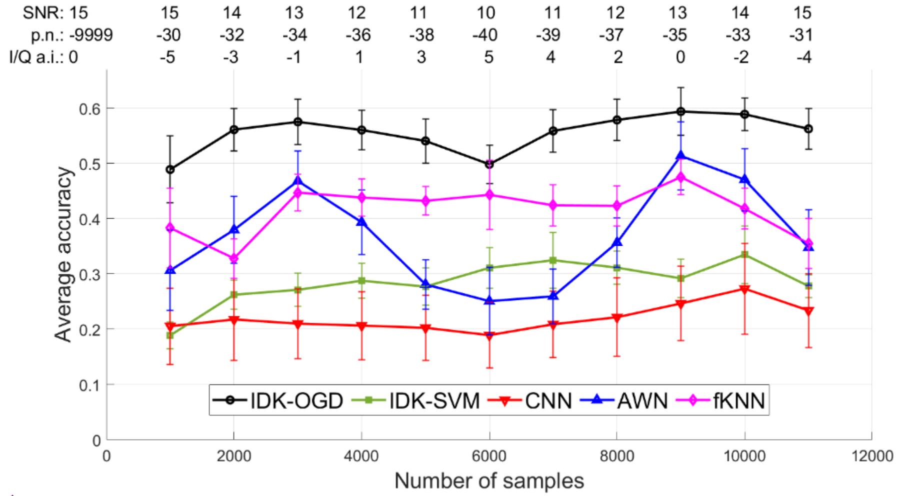
SNR: Signal-to-Noise Ratio

In each plot, the parameters for the training and testing channel conditions are specified at the top.

Experiments





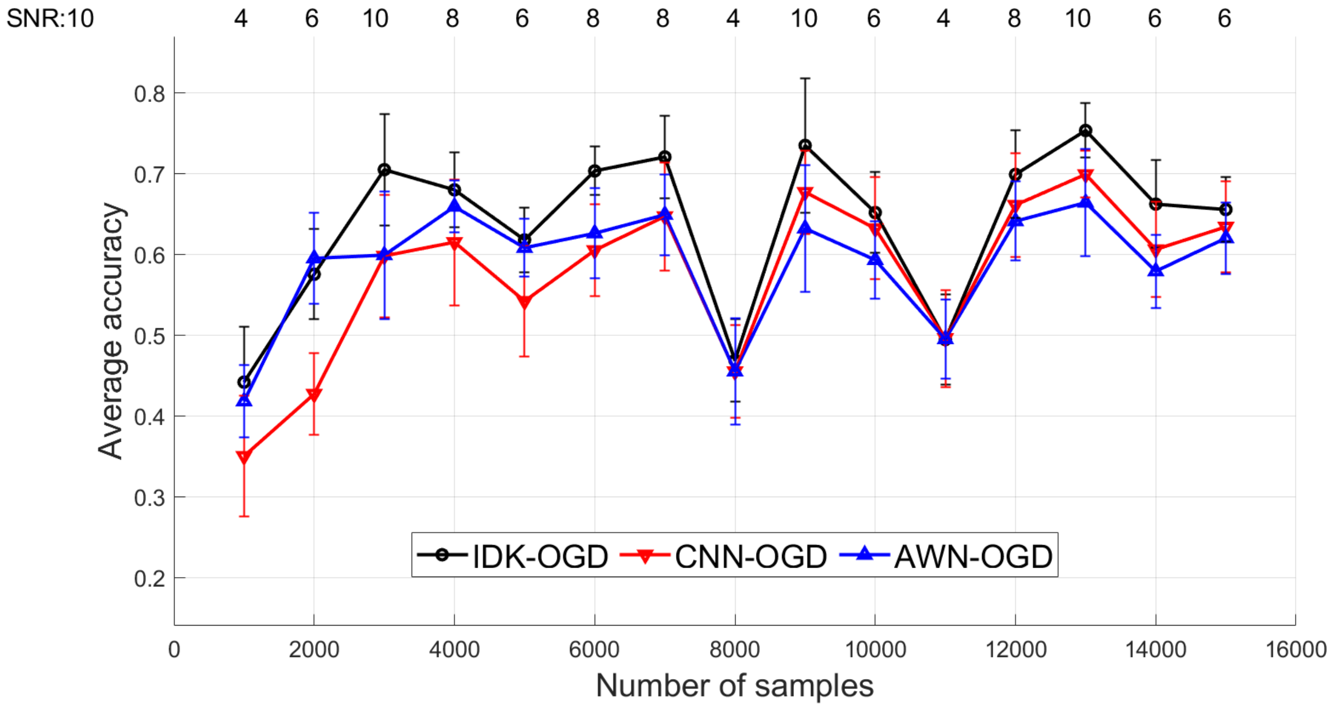


p.n.: phase noise.

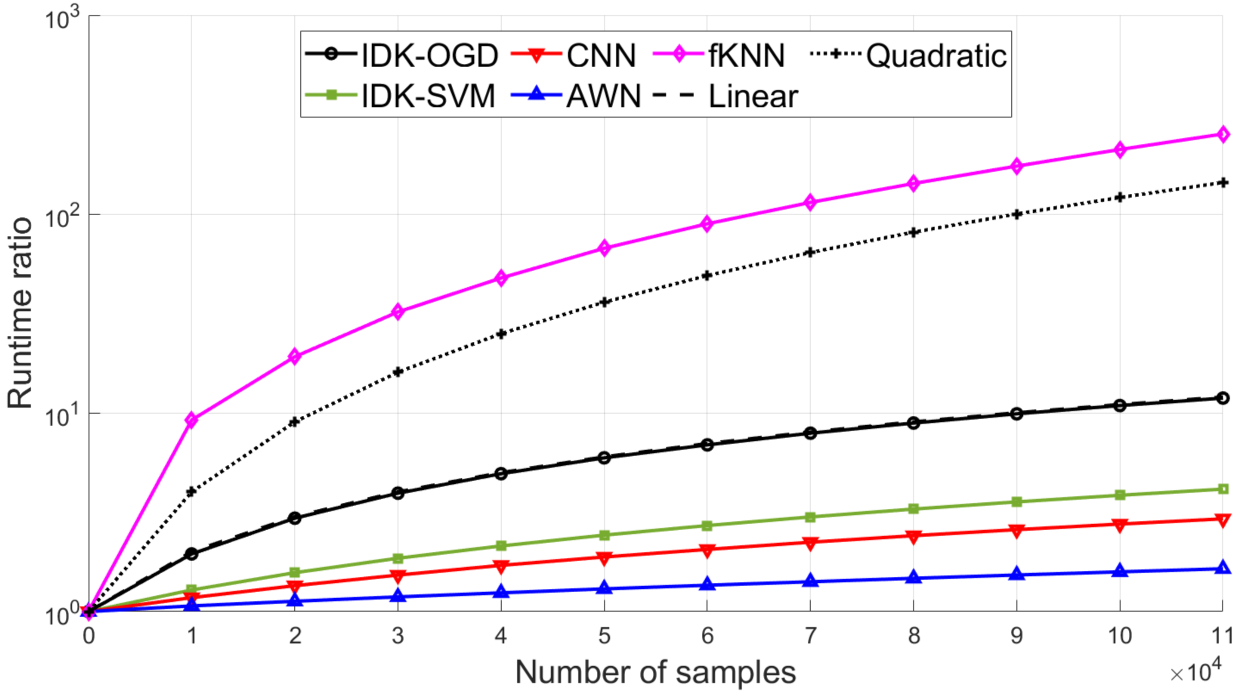
I/Q a.i.: I/Q amplitude imbalance.

In each plot, the parameters for the training and testing channel conditions are specified at the top.

Ablation Study



Runtime Comparison



Conclusions

- Addressing the fundamental flaws in current AMC methods by introducing a novel, loseless distributional signal representation.
- By proposing IDK-OGD, we successfully integrated the power of distributional kernels with efficient online learning.
- We demonstrated that this approach achieves superior performance under realistic, time-varying, and mismatched channel conditions with linear time complexity.

Thanks!

Full paper

Contact me

Presented by **Xinpeng Li**