

Paper 5: Entropy Regularization Module

Formulation

Entropy regularization penalizes overly concentrated weight allocations to encourage diversification. The entropy term is given by:

$$H(w) = - \sum_{i=1}^N w_i \log w_i$$

We add a regularization term to the objective function:

$$\text{Objective} = \text{Original Objective} + \gamma H(w)$$

Where:

- w_i : Portfolio weight for asset i .
- γ : Entropy regularization strength (controls spread).

Why this Works

Entropy maximization spreads weights across assets, mitigating concentration risk and enhancing robustness to shocks or sudden drawdowns.

Algorithm Steps

1. Calculate entropy of current weights.
2. Compute gradient or adjust objective to include entropy term.
3. Update weights while balancing entropy and performance.

Code Equivalent

```
def entropy_regularize(weights, gamma=0.01):  
    entropy = -np.sum(weights * np.log(weights + 1e-8))  
    adjusted_weights = weights + gamma * entropy
```

```
adjusted_weights = np.maximum(adjusted_weights, 1e-5)
adjusted_weights /= np.sum(adjusted_weights)
return adjusted_weights

# Example usage:
weights = entropy_regularize(weights, gamma=0.01)
```

Summary

- Promotes diversified exposure.
- Reduces vulnerability to individual asset risk.
- Balances performance with robustness via controlled spread.