

In financial markets, traders seek to identify trends and directional strength to optimize entry and exit points. Traditional technical indicators such as **ADX (Average Directional Index)** and **DI+/DI- (Directional Indicators)** measure trend strength but often fail to capture **hidden relationships** between assets or sectors.

To address this, we define a **financial market as a graph** $G = (V, E)$, where:

- Each **node** $i \in V$ represents a financial asset.
- An **edge** $(i, j) \in E$ exists if asset i and asset j share a significant correlation or dependency.

Each asset i is associated with a **feature vector** X_i consisting of ADX, DI+, and DI- values:

$$X_i = [x_i^{\text{ADX}}, x_i^{\text{DI+}}, x_i^{\text{DI-}}]$$

Problem Statement:

Given a financial graph $G = (V, E)$ and a set of node features X_i , **how can we use information from neighboring assets to improve the predictive power of trend-following indicators and optimize trading decisions?**

To solve this, we apply a **Graph-Based Feature Aggregation**:

$$\tilde{x}_i^{\text{feature}} = \frac{1}{|\mathcal{N}(i)|} \sum_{j \in \mathcal{N}(i)} x_j^{\text{feature}}$$

where $\mathcal{N}(i)$ represents the set of **neighboring assets** of node i .

The goal is to use the aggregated signals:

$$S = \bigcup_{i=1}^N X_i$$

as input for a **Perceptron-based trading model** that makes optimal trading decisions based on learned patterns in asset relationships.

Thus, we seek to:

1. **Enhance trend-following indicators** via graph-based aggregation.
2. **Improve trading signal reliability** by incorporating cross-asset dependencies.
3. **Optimize trade execution** using a supervised learning model.

This leads to the **Graph-Enhanced Directional Trading (GEDT) Strategy**, which dynamically updates market trend strength across a network of assets for improved trade decision-making.