

# Video Game Publisher Factor Model

## Project Overview

A machine learning factor model analyzing video game publisher performance and its relationship to stock returns. The system aggregates game performance data to generate predictive features for publisher stock prediction.

## Architecture

### System Components

#### 1. Data Collection Layer

- Interfaces with Gamalytic API
- Retrieves detailed game performance metrics
- Handles API rate limiting and error recovery

#### 2. Feature Engineering Pipeline

- Processes raw game data into predictive features
- Generates metrics at both game and publisher levels
- Implements sophisticated aggregation strategies

#### 3. Parallel Processing Framework

- Multi-level parallelization architecture:
  - Publisher-level: Concurrent game processing
  - Game-level: Parallel similar game loading
  - Feature-level: Concurrent aggregate calculations
- Optimized for large-scale data processing

#### 4. Triple Barrier Labeling System

- Implements financial machine learning labeling approach
- Creates classification labels for stock returns
- Supports dynamic barrier calculation

#### 5. Persistence Infrastructure

- Manages efficient data storage and retrieval
- Implements versioning through timestamps
- Enables rapid development iteration

### Core Classes

#### Publisher Layer

- `ParallelPublisher` : Orchestrates publisher-level operations
- Manages game portfolio
- Handles feature aggregation

#### Game Layer

- `Game` : Encapsulates individual game data and operations
- Processes historical performance metrics
- Manages relationship with similar games

#### Similarity Analysis

- `Genre/ParallelGenre` : Processes genre-based similarity
- `AudienceOverlap/ParallelAudienceOverlap` : Handles player base similarity

## Feature Framework

### Key Categories

1. Game Engagement Metrics
  - Player growth rates
  - Quality-adjusted engagement
  - Player retention metrics

## 2. Sentiment Metrics

- Review score momentum
- Sentiment divergence analysis
- Weighted sentiment indicators

## 3. Monetization Metrics

- Revenue efficiency
- Sales conversion rates
- Price-adjusted metrics

## 4. Lifecycle-Adjusted Metrics

- Age-normalized performance
- Lifecycle stage indicators
- Temporal adjustment factors

## 5. Stability Metrics

- Revenue consistency measures
- Player base stability
- Performance volatility indicators

## 6. Leading Indicator Metrics

- Social momentum signals
- Wishlist trends
- Forward-looking indicators

## 7. Value Metrics

- Quality-price ratios
- Engagement efficiency
- Market position indicators

# Recent Implementation: Persistence Layer

---

## Motivation

The system's computational intensity and data volume necessitated an efficient persistence strategy to optimize development workflow and enable rapid iteration.

## Key Features

### 1. Automatic Persistence

- Seamless data saving post-calculation
- Integrated with existing workflows
- Transparent to system operations

### 2. Version Control

- Timestamp-based file naming
- Historical calculation preservation
- Clear data lineage

### 3. Efficient Recovery

- Automatic latest version loading
- Fast data retrieval
- Robust error handling

### 4. Clean Integration

- Maintains existing functionality
- Non-intrusive implementation
- Extensible design

## Implementation Status

- ✓ Core persistence functionality
- ✓ Parallel processing integration
- ✓ Automatic data recovery
- ✓ Version control system

This overview represents the current state of the factor model system, with a focus on recent persistence layer improvements that enable more efficient development and iteration.