Agencio Predict

PRODUCT GUIDE · V1.0 · MAY 2026

Prediction Market Integration

The complete technical and user guide to trading on real-world event probabilities inside Agencio Predict — covering the UI, the DSL primitives, the data pipeline, and the AI learning loop that ties them together.

Polymarket | Kalshi | Metaculus | PredictIt

FOR

Macro · Crypto · Volatility · Algo Builders

SURFACE AREA

/admin/algorithms · /predict · /derivatives

CONTENTS

What's inside

election_uncertainty()

Best Practices

1. Combine with Technical Analysis
2. Watch Divergence
3. Volume Matters
4. Lead Time Awareness
5. Event Specificity

Troubleshooting

"prediction_market_prob() not available"

Low Market Correlation

Stale Predictions

No Predictions Found for Asset

Related Documentation

SECTION

Overview

Agencio Predict integrates real-time prediction market data from Polymarket, Kalshi, Metaculus, and PredictIt directly into the AI trading system. This creates a unique edge:

What makes this different:

- Trade on real-world event probabilities (Fed rate cuts, recession odds, elections)
- AI learns which prediction sources are most accurate per domain
- System correlates prediction accuracy with market moves
- Generates actionable trading signals from crowd intelligence

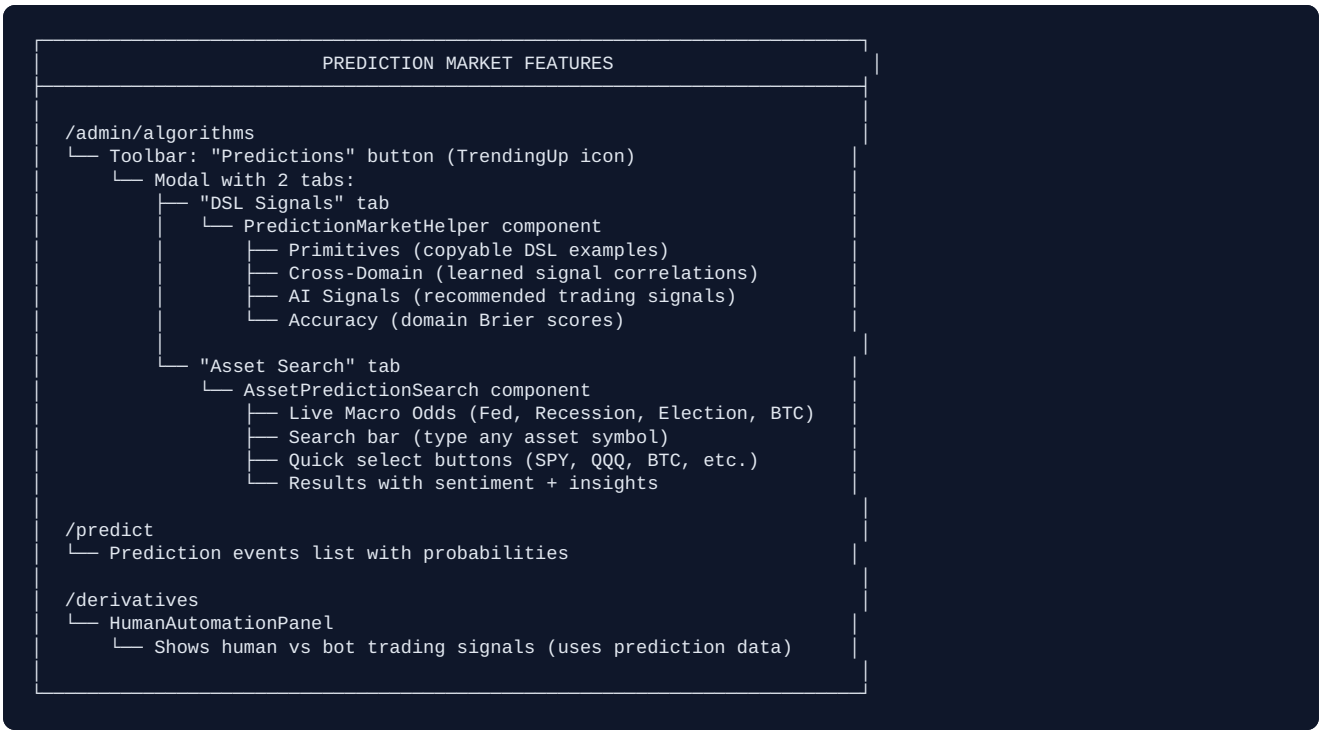
Who benefits:

- **Macro traders**: Position ahead of Fed decisions, elections, geopolitical events
- **Crypto traders**: Trade on Bitcoin ETF approvals, regulatory decisions
- **Volatility traders**: Position for election uncertainty, crisis events
- **Algorithm builders**: Add prediction market signals to any trading strategy

SECTION

Where to Find These Features

UI Navigation Map



Step-by-Step: Finding the Prediction Markets Panel

1. **Log in** to Agencio Predict
2. Navigate to **Admin → Algorithms** (`/admin/algorithms`)
3. Look at the **toolbar** at the top of the page
4. Click the **"Predictions"** button (has a trending chart icon)
5. A modal opens with two tabs:
    - **DSL Signals**: View all prediction market DSL primitives
    - **Asset Search**: Search for predictions by asset

Where Each Feature Lives

Feature	URL	Component	What You'll Find
Algorithm Builder	`/admin/algorithms`	AlgorithmPage	Build trading strategies with prediction signals
Predictions Modal	`/admin/algorithms` → Predictions button	PredictionHelperModal	DSL primitives + Asset search
DSL Primitives	Predictions → DSL Signals tab	PredictionMarketHelper	Copyable primitive examples
Asset Search	Predictions → Asset Search tab	AssetPredictionSearch	Find predictions for any asset
Live Macro Odds	Predictions → Asset Search tab	OddsCard	Fed, recession, election, BTC odds
Cross-Domain Signals	Predictions → DSL Signals → Cross-Domain	SignalCard	Learned correlations
AI Recommendations	Predictions → DSL Signals → AI Signals	RecommendationCard	AI-suggested trading signals
Domain Accuracy	Predictions → DSL Signals → Accuracy	AccuracyCard	Brier scores by domain
Prediction Events	`/predict`	EventsList	All active prediction events
Human vs Bot	`/derivatives`	HumanAutomationPanel	Human/bot trading classification

Quick Access Tips

1. **Keyboard shortcut**: Press `P` while on `/admin/algorithms` to open Predictions panel (if supported)
2. **Direct URL**: Bookmark `/admin/algorithms` for quick access to the algorithm builder
3. **API Testing**: Use browser dev tools to hit APIs directly:
    - `/api/predict/v1/insights/live-odds`
    - `/api/predict/v1/insights/asset-predictions/SPY`

SECTION

User Guide

Quick Start

1. **Navigate to the Algorithm Builder**: Go to `/admin/algorithms`
2. **Open Prediction Markets Panel**: Click the "Predictions" button in the toolbar
3. **Explore Two Views**:
    - **DSL Signals**: See all available prediction market primitives with examples
    - **Asset Search**: Find predictions related to any asset (SPY, BTC, GLD, etc.)
4. **Create a Strategy**: Use prediction market signals in your trading algorithm

Example entry condition:

```
fed_rate_cut_prob() > 0.65 && rsi(14) < 40
```

Asset Search

The Asset Search feature lets you find prediction market signals relevant to any asset you trade.

How to use:

1. Open the Predictions panel and click "Asset Search" tab
2. Type an asset symbol (SPY, BTC-USD, QQQ, GLD, etc.) or click a quick-select button
3. View related predictions with:
    - Probability (current odds)
    - Volume (market conviction)
    - Trading implications (what it means for your asset)
    - Overall sentiment (bullish/bearish/neutral aggregate)
    - Key insights (AI-generated trading recommendations)

Example: Searching "SPY"

Returns predictions about:

- Recession probability → Bearish for SPY
- Fed rate cuts → Bullish for SPY
- GDP forecasts → Affects earnings expectations
- Unemployment claims → Leading indicator

The system automatically calculates:

- **Overall Sentiment**: Weighted average of all relevant predictions
- **Key Insights**: Actionable trading recommendations

Live Odds Dashboard

The Live Odds panel shows real-time probabilities for key macro events:

Event	What It Means	Trading Application
Fed Rate Cut	Probability of rate cut at next FOMC	Rate cuts bullish for risk assets, bearish for USD
Recession	12-month recession probability	High odds → flight to bonds, gold; bearish equities
Election Uncertainty	How contested the outcome is	High uncertainty → expect VIX spikes
BTC $100k	Bitcoin reaching $100,000	Crypto market sentiment indicator

Color coding:

- Green: Favorable for risk assets
- Yellow: Neutral/uncertain
- Red: Unfavorable for risk assets (or elevated risk)

Using DSL Primitives

DSL primitives are the building blocks for prediction market signals in your algorithms.

Basic Usage:

```
entry:
  when: |
    fed_rate_cut_prob() > 0.65
```

Combining with Technical Analysis:

```
entry:
  when: |
    fed_rate_cut_prob() > 0.65 &&
    prediction_market_delta("fed-rate-cut", 24) > 0.05 &&
    rsi(14) < 40 &&
    price > sma(50)
```

Available Primitives:

Primitive	Returns	Use Case
`prediction_market_prob("keyword")`	0-1	Current probability
`prediction_market_delta("keyword", hours)`	-1 to 1	Momentum in sentiment
`prediction_market_volume("keyword")`	USD	Conviction/liquidity
`prediction_category_sentiment("category")`	-1 to 1	Broad category sentiment
`prediction_market_consensus("keyword")`	0-1	Cross-platform average
`prediction_market_divergence("keyword")`	0-1	Platform disagreement
`fed_rate_cut_prob()`	0-1	Fed rate cut shortcut
`recession_prob()`	0-1	Recession shortcut
`election_uncertainty()`	0-1	Election uncertainty

Building Trading Strategies

Strategy 1: Fed-Aware Crypto Trading

Buy Bitcoin when rate cut probability is high and technicals support entry:

```
name: Fed Rate Cut BTC Long
universe: [BTC-USD]

entry:
  when: |
    fed_rate_cut_prob() > 0.65 &&
    prediction_market_delta("fed-rate-cut", 24) > 0.02 &&
    rsi(14) < 40
  size: kelly(0.6, 1.5, 5000)

exit:
  when: |
    fed_rate_cut_prob() < 0.40 ||
    recession_prob() > 0.6 ||
    position_pnl_pct() > 15 ||
    position_pnl_pct() < -5
```

Strategy 2: Election Volatility Play

Go long volatility when election outcome is uncertain:

```
name: Election Volatility Long
universe: [UVXY]

entry:
  when: |
    election_uncertainty() > 0.75 &&
    prediction_market_divergence("election") > 0.2 &&
    vix() < 25
  size: fixed_usd(2000)

exit:
  when: |
    election_uncertainty() < 0.5 ||
    position_age_hours() > 72 ||
    position_pnl_pct() < -10
```

Strategy 3: Recession Hedge

Allocate to safe havens when recession odds rise:

```
name: Recession Bond Allocation
universe: [TLT, GLD]

entry:
  when: |
    recession_prob() > 0.5 &&
    prediction_market_delta("recession", 168) > 0.1
  size: risk_pct(2)

exit:
  when: |
    recession_prob() < 0.3 ||
    position_age_days() > 60
```

SECTION

Technical Architecture

Data Sources

The system aggregates data from multiple prediction market platforms:

Source	Coverage	Update Frequency	API Type
Polymarket	Crypto, elections, macro, geopolitical	Real-time	GraphQL
Kalshi	Macro events, Fed decisions, economic data	Real-time	REST
Metaculus	Long-term forecasts, science, tech	Daily	REST
PredictIt	US elections, political events	Hourly	REST

Internal Database:

- `predict.prediction_events` — Active prediction events
- `predict.prediction_snapshots` — Historical probability snapshots
- `predict.prediction_accuracy` — Brier scores and market correlation

Data Flow

Database Schema

Core Tables:

```
-- Active prediction events
predict.prediction_events (
  id UUID PRIMARY KEY,
  title TEXT,
  slug TEXT UNIQUE,
  category_id UUID REFERENCES prediction_categories,
  current_probability NUMERIC(5,4),  -- Weighted composite
  status TEXT CHECK (status IN ('ACTIVE', 'RESOLVED', 'CANCELLED')),
  resolution TEXT,  -- 'YES', 'NO', or NULL
  resolved_at TIMESTAMPTZ,
  metadata JSONB,  -- source, volume, etc.
  created_at TIMESTAMPTZ,
  updated_at TIMESTAMPTZ
)

-- Historical probability snapshots
predict.prediction_snapshots (
  id UUID PRIMARY KEY,
  event_id UUID REFERENCES prediction_events,
  probability NUMERIC(5,4),
  source TEXT,
  volume NUMERIC,
  captured_at TIMESTAMPTZ
)

-- Accuracy tracking
predict.prediction_accuracy (
  id UUID PRIMARY KEY,
  event_id UUID REFERENCES prediction_events,
  predicted_prob NUMERIC(5,4),
  actual_outcome INT CHECK (actual_outcome IN (0, 1)),
  brier_score NUMERIC(10,6),
  market_symbol TEXT,  -- e.g., 'BTC-USD'
  market_move_percent NUMERIC(10,4),
  market_move_correct BOOLEAN,
  resolved_at TIMESTAMPTZ
)

-- Learned cross-domain signals
predict.cross_domain_signals (
  id UUID PRIMARY KEY,
  source_domain TEXT,
  target_market TEXT,
  correlation NUMERIC(5,4),
  sample_size INT,
  signal_strength TEXT,  -- 'strong', 'moderate', 'weak', 'none'
  last_updated TIMESTAMPTZ
)
```

API Endpoints

Endpoint	Method	Description
`/api/predict/v1/insights/live-odds`	GET	Real-time macro event odds
`/api/predict/v1/insights/asset-predictions/:asset`	GET	Predictions for specific asset
`/api/predict/v1/insights/search-predictions?q=keyword`	GET	Search predictions by keyword
`/api/predict/v1/insights/prediction-accuracy`	GET	Domain accuracy statistics
`/api/predict/v1/insights/cross-domain-signals`	GET	Learned signal correlations
`/api/predict/v1/insights/recommended-signals`	GET	AI-recommended trading signals
`/api/predict/v1/insights/prediction-tooltips`	GET	DSL primitive documentation
`/api/predict/v1/insights/explain-prediction`	POST	AI explanation of a signal

Example Responses:

GET /api/predict/v1/insights/live-odds

```
{
  "fedRateCut": 0.72,
  "recession": 0.35,
  "electionUncertainty": 0.81,
  "btc100k": 0.45,
  "lastUpdated": "2026-05-04T12:00:00Z"
}
```

GET /api/predict/v1/insights/asset-predictions/SPY

```
{
  "asset": "SPY",
  "assetClass": "index",
  "overallSentiment": -0.15,
  "keyInsights": [
    "3 high-conviction predictions affecting SPY",
    "Mixed prediction market signals for SPY",
    "Elevated recession risk (42%) → Bearish SPY"
  ],
  "predictions": [
    {
      "id": "abc123",
      "source": "polymarket",
      "title": "Will the US enter a recession in 2026?",
      "probability": 0.42,
      "volume": 2500000,
      "tradingImplication": "Elevated recession risk (42%) → Bearish SPY",
      "assetCorrelation": -0.6
    }
  ]
}
```

SECTION

AI Learning System

Accuracy Tracking

When a prediction event resolves, the system calculates:

Brier Score (lower is better):

```
Brier = (predicted_probability - actual_outcome)²
```

Example:

- Predicted 70% probability of YES
- Event resolved to YES (1)
- Brier = (0.70 - 1)² = 0.09 (good prediction)

Perfect Brier = 0 (100% confident and correct) **Worst Brier = 1** (100% confident and wrong)

Domain Weight Learning

Each prediction source has a weight per domain:

```
Crypto weights:   Polymarket=0.40, Kalshi=0.25, Metaculus=0.20, Users=0.15
Elections weights: Polymarket=0.35, Kalshi=0.25, PredictIt=0.25, Users=0.15
Economics weights: Polymarket=0.30, Kalshi=0.35, Fed models=0.25, Users=0.10
```

Daily weight updates:

1. Calculate average Brier score per source for resolved events in last 90 days
2. Sources with below-average Brier get weight increase (max +5%)
3. Sources with above-average Brier get weight decrease (max -5%)
4. Weights normalized to sum to 1.0

Cross-Domain Signal Analysis

The system learns correlations between prediction domains and market moves:

Process:

1. Track market price 24 hours after each prediction snapshot
2. Calculate correlation between prediction probability and price change

3. Classify signal strength:
    - **Strong**: correlation > 0.6, sample_size > 30
    - **Moderate**: correlation > 0.3, sample_size > 20
    - **Weak**: correlation > 0.1, sample_size > 10
    - **None**: correlation < 0.1

Example learned correlations:

- Fed rate cut probability → +0.65 correlation with BTC-USD (strong)
- Recession probability → -0.72 correlation with SPY (strong)
- Election uncertainty → +0.58 correlation with VIX (moderate)

SECTION

DSL Primitive Reference

prediction_market_prob(keyword)

Returns the current weighted probability (0-1) for events matching the keyword.

Parameters:

- `keyword`: String to search in event titles (e.g., "fed-rate-cut", "recession", "bitcoin")

Returns: Number between 0 and 1

Example:

```
when: prediction_market_prob("fed-rate-cut") > 0.7
```

> **Trading tip —** High probability events (>0.7) are often already priced in. Watch for probability *changes* rather than absolute levels.

prediction_market_delta(keyword, hours)

Returns the probability change over the specified time window.

Parameters:

- `keyword`: String to search in event titles
- `hours`: Number of hours to look back

Returns: Number between -1 and 1

Example:

```
when: prediction_market_delta("recession", 24) > 0.05
```

> **Trading tip —** Rapid changes (>5% in 24h) often precede market moves. Delta is a momentum signal.

prediction_market_volume(keyword)

Returns the total trading volume (in USD) on prediction markets for matching events.

Parameters:

- `keyword` : String to search in event titles

Returns: Number (USD volume)

Example:

```
when: prediction_market_volume("btc-100k") > 1000000
```

> **Trading tip —** Higher volume = higher conviction. Low volume (<$100k) predictions may be noise.

prediction_category_sentiment(category)

Returns aggregate sentiment for an entire prediction category.

Parameters:

- `category` : Category slug (e.g., "crypto", "economics", "politics", "technology")

Returns: Number between -1 and 1

Example:

```
when: prediction_category_sentiment("crypto") > 0.5
```

> **Trading tip —** Extreme values (>0.8 or <-0.8) often signal crowded trades. Consider fading extremes.

prediction_market_consensus(keyword)

Returns the volume-weighted average probability across all prediction platforms.

Parameters:

- `keyword` : String to search in event titles

Returns: Number between 0 and 1

Example:

```
when: prediction_market_consensus("election-2024") > 0.6
```

> **Trading tip —** Consensus is more reliable than single-platform odds. Divergence from consensus may signal arbitrage opportunities.

prediction_market_divergence(keyword)

Returns the disagreement level between prediction platforms.

Parameters:

- `keyword`: String to search in event titles

Returns: Number between 0 (full agreement) and 1 (maximum divergence)

Example:

```
when: prediction_market_divergence("war") > 0.3
```

> **Trading tip —** High divergence (>0.3) indicates uncertainty. Consider reducing position size or waiting for convergence.

fed_rate_cut_prob()

Shortcut for Fed rate cut probability at the next FOMC meeting.

Parameters: None

Returns: Number between 0 and 1

Example:

```
when: fed_rate_cut_prob() > 0.65
```

> **Trading tip —** Rate cuts are bullish for crypto/stocks, bearish for USD. Watch for probability spikes after economic data releases.

recession_prob()

Shortcut for 12-month recession probability.

Parameters: None

Returns: Number between 0 and 1

Example:

```
when: recession_prob() < 0.3
```

> **Trading tip —** Rising recession odds favor defensive plays (bonds, gold, utilities). Falling odds favor risk-on positioning.

election_uncertainty()

Returns election uncertainty index (how contested the outcome is).

Parameters: None

Returns: Number between 0 (clear winner) and 1 (highly contested)

Example:

```
when: election_uncertainty() > 0.7
```

> **Trading tip —** High uncertainty = high VIX. Consider volatility plays (UVXY, options) ahead of elections.

SECTION

Best Practices

1. Combine with Technical Analysis

Prediction market signals work best when combined with price action:

```
# Good: Combine macro + technicals
when: |
  fed_rate_cut_prob() > 0.65 &&
  rsi(14) < 40 &&
  price > sma(200)

# Bad: Macro signal alone
when: fed_rate_cut_prob() > 0.65
```

2. Watch Divergence

High platform divergence means low confidence:

```
# Reduce size when uncertain
size: |
  if prediction_market_divergence("event") > 0.3
    then fixed_usd(1000)
    else fixed_usd(3000)
```

3. Volume Matters

Ignore low-volume predictions:

```
when: |
  prediction_market_prob("event") > 0.7 &&
  prediction_market_volume("event") > 500000
```

4. Lead Time Awareness

Prediction markets often lead price by hours/days. Use for positioning, not precise timing.

5. Event Specificity

Prediction markets excel at discrete events (Fed meetings, elections, ETF approvals) more than continuous moves.

SECTION

Troubleshooting

"prediction_market_prob() not available"

Cause: The prediction market data provider isn't configured in the evaluator context.

Fix:

1. Ensure the algorithm is running in paper/live mode (not just backtest)
2. Check that prediction_events table has active events
3. Verify scheduler jobs are running (`event-probability-composite`)

Low Market Correlation

Cause: Predictions are accurate but don't correlate with market moves.

What happens: The system automatically reduces the weight of non-actionable signals.

What to do: The system self-corrects. For immediate use, combine prediction signals with other indicators.

Stale Predictions

Cause: Scheduler job not running.

Fix:

1. Check scheduler logs: `docker logs predict-web-dev | grep scheduler`
2. Verify `event-probability-composite` job is active
3. Manually trigger: Hit `/api/predict/v1/admin/scheduler/trigger?job=event-probability-composite`

No Predictions Found for Asset

Cause: Asset not in the mapping or no matching predictions.

What happens: Returns empty predictions array with neutral sentiment.

What to do:

1. Use `search-predictions` endpoint with different keywords
2. Check if Polymarket/Kalshi have events for this asset type
3. Consider adding the asset to `ASSET_PREDICTION_MAP` in `live-prediction-market.ts`

SECTION

Related Documentation

- `docs/12-algorithm-builder.md` — Full algorithm builder documentation
- `docs/PREDICTION-MARKET-TRADING.md` — Quick reference for trading with prediction markets
- `docs/40-ai-trader-compliance-and-audit.md` — Compliance and audit documentation
- `packages/be/src/algorithms/dsl/types.ts` — Full primitive registry