

KNN PATTERN HUNTER

User Guide



Figure 1. KNN Pattern Hunter in MetaTrader 5.

Contents

1. What KNN Pattern Hunter does
2. Workflow: from bar to trading signal
3. Dashboard and manual control
4. Installation and first launch
5. EA parameters
6. How the KNN forecast is formed
7. Position management and averaging
8. Model saving and loading
9. Recommended setup and testing workflow
10. Important limitations and implementation details

1. What KNN Pattern Hunter does

KNN Pattern Hunter 2.0 is an MT5 Expert Advisor that analyzes the market on the selected timeframe and makes trading decisions based on historical analogues. Trading decisions are calculated from a closed bar; management of open positions and the Dashboard are updated on every tick.

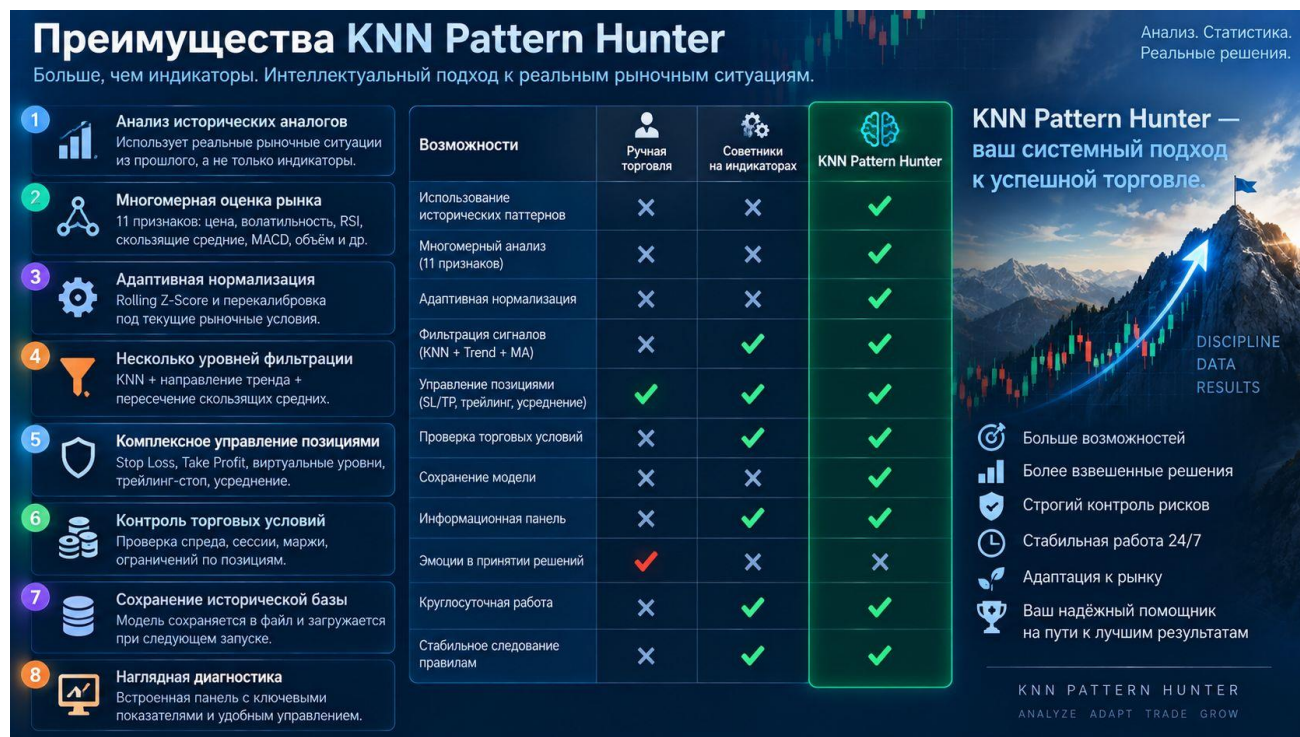


Figure 2. Overview of KNN Pattern Hunter capabilities.

Key capabilities

- 11 market features: three price changes, relative candle body, RSI, relative ATR, price-to-Fast-MA ratio, Fast/Slow MA difference, MACD–Signal, relative volume change, and correlation between price-change and volume-change.
- Adaptive KNN model with rolling Z-score normalization.
- Three neighbor aggregation methods: Mean, Weighted by distance, and Median.
- Combination of KNN, Trend Signal, and Fast/Slow SMA Cross Signal.
- Real and virtual SL/TP, trailing stop, and spread/position/session limits.
- ATR-based adaptive distance between positions.
- Model persistence in a binary file whose name contains the symbol and timeframe.
- Dashboard with trading statistics and BUY / SELL / CLOSE buttons.

Important note about the term “training”

KNN is not trained by a gradient-based method in this EA. The pattern database is built from historical examples; for the current state, nearest neighbors are searched and their target returns are aggregated.

2. Workflow: from bar to trading signal

2.1. Building the 11 features

No	Feature	Scale
1–3	Price change over 1 / 2 / 3 bars	%
4	Relative candle body (Close–Open) / previous Close	%
5	RSI	0...100 before normalization
6	ATR / price	%
7	Price relative to Fast MA	%
8	Fast MA minus Slow MA	%
9	MACD Main – Signal	absolute units
10	Relative volume change	%
11	Correlation of price changes and volume changes	–1...+1

Key point

MACD, volume, and correlation are KNN features, but they are not separate votes in the trading signal. The individual decision sources are KNN, the trend based on the selected PriceMethod, and the two-SMA crossover.

2.2. Rolling Z-score normalization

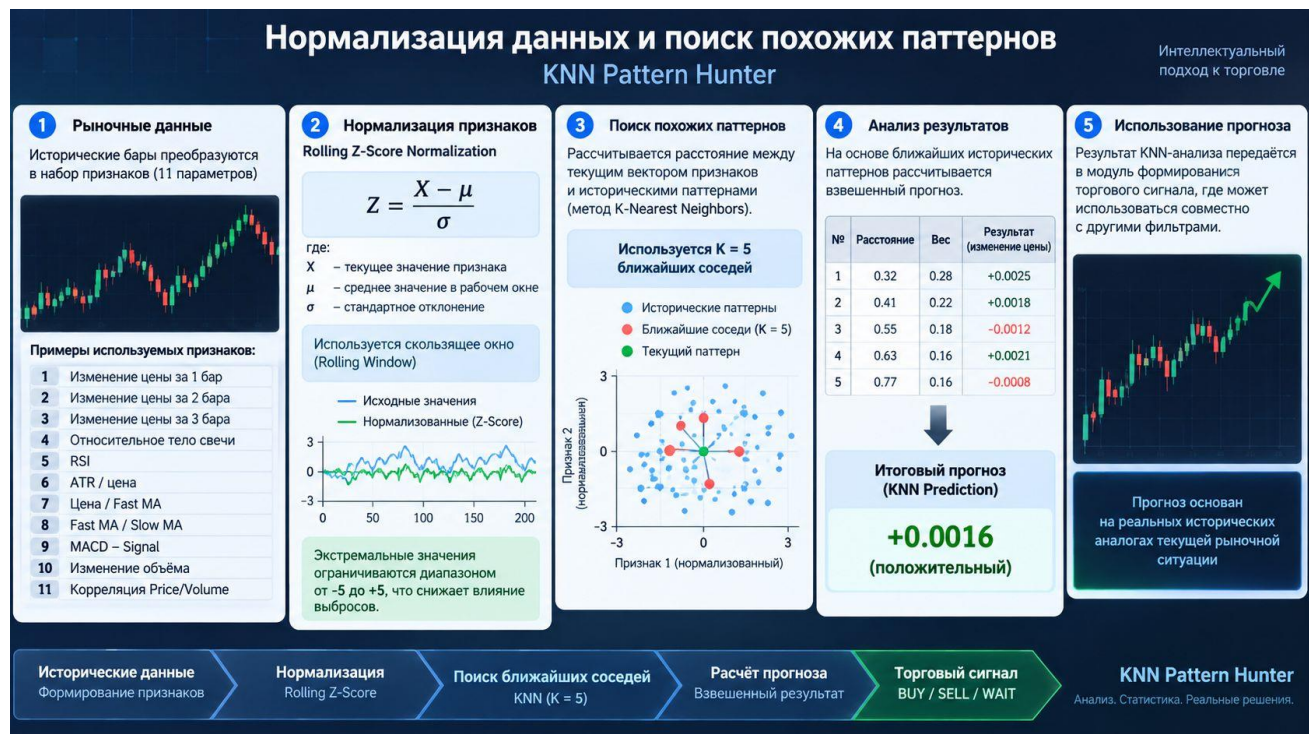


Figure 3. Normalization and similar-pattern search. K=5 is shown as an example; the actual value is set by KNN_Neighbors.

The normalizer brings features to a comparable scale. Its statistics are rebuilt from the available historical database. In live mode, a new pattern is added only after its TargetBars outcome is already known, so the pattern belongs to information from the past relative to the current trading bar.

2.3. Neighbor search and forecast

- Euclidean distance is calculated from the current feature vector to all valid historical patterns.
- The K nearest neighbors are selected.
- Mean: simple average of the neighbors' target returns.
- Weighted: a closer neighbor receives a higher weight; the code uses $1 / (\text{distance} + 1e-9)$.
- Median: median of the neighbors' target returns.

2.4. What KNN_Threshold means

KNN_Threshold compares the predicted return in percent with the effective threshold. The effective threshold is adapted using volatilityRatio: it is first multiplied by the volatility factor, then clamped to 0.3...0.7, and cannot be lower than MinTargetReturn.

Condition	KNN result
Prediction > effective threshold	BUY
Prediction < -effective threshold	SELL
Prediction ≤ effective threshold	WAIT

3. Dashboard and manual control

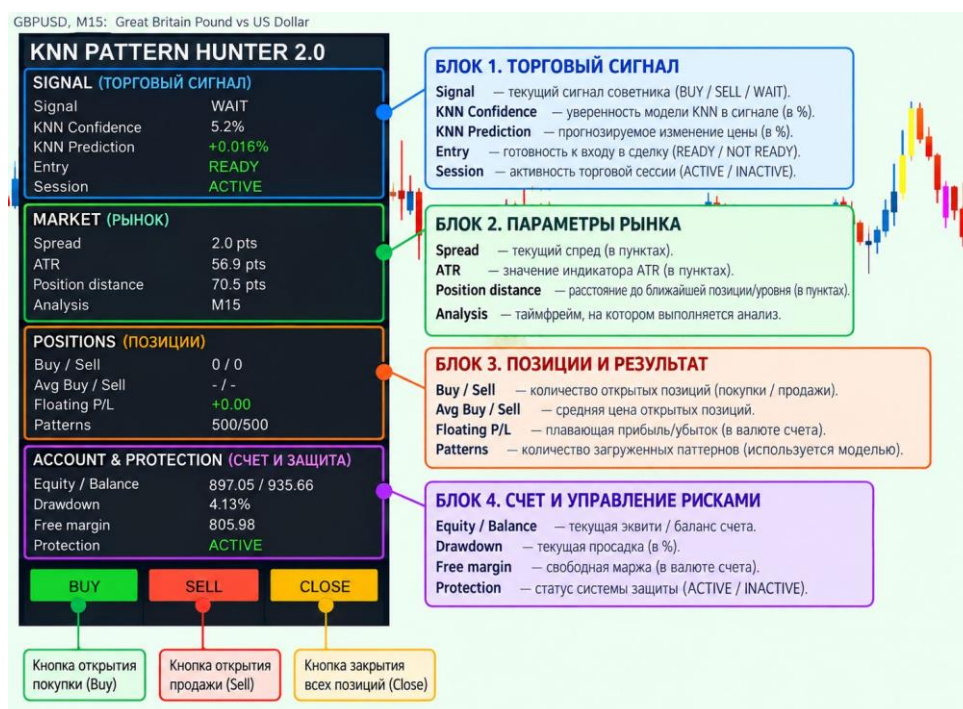


Figure 4. KNN Pattern Hunter 2.0 Dashboard.

3.1. Dashboard blocks

Block	What
SIGNAL	Signal, KNN Confidence, KNN Prediction, Entry, Session
MARKET	Spread, ATR, Position distance, Analysis timeframe
POSITIONS	Buy/Sell, Avg Buy/Sell, Floating P/L, Patterns
ACCOUNT & PROTECTION	Equity/Balance, Drawdown, Free margin, Protection

Buttons	BUY — manual Buy entry; SELL — manual Sell entry; CLOSE — closes EA positions
---------	---

KNN Confidence ≠ probability of a profitable trade

Confidence is calculated as the normalized value $|\text{Prediction}| / \text{effective threshold} \times 100\%$, capped at 0...100%. It indicates KNN strength relative to the threshold, not the statistical probability of a profitable trade.

3.2. CLOSE button

CLOSE closes only positions for the current symbol and current MagicNumber. Positions belonging to other EAs and manual positions with a different MagicNumber are not affected.

3.3. Entry and Session

- Session shows whether the current server time is inside the configured trading interval.
- Entry shows the result of the trading-condition checks: connection, trading permission, session, spread, and, when enabled, correlation.
- If SessionStart = SessionEnd, the trading session is treated as 24/7.

4. Installation and first launch

1. Place the compiled EA in MQL5/Experts, or open the .mq5 source file in MetaEditor and compile it.
2. Open the required chart in MT5 and attach KNN Pattern Hunter 2.0 to it.
3. Enable Algo Trading if the EA is expected to open positions.
4. Select AnalysisPeriod. The EA can analyze a timeframe different from the chart timeframe.
5. Check the MagicNumber and risk parameters.
6. Make sure sufficient history is available for KNN_HistoryBars and indicator-feature calculation.
7. On the first launch, the EA validates the parameters, creates indicator handles and the model, creates the Dashboard, and then attempts to load a saved model. If loading is not possible, it trains the model from historical data.
8. Before live trading, test the EA in Strategy Tester and on a demo account.

Practical tip

For virtual SL/TP and correct position management, the terminal and EA must remain running continuously. A VPS is typically used for 24/7 operation.

5. EA parameters

The parameters below are taken directly from the .mq5 source. Values in the “Default” column are the code's input defaults, not values from any particular screenshot.

KNN Pattern Hunter 2.0 (AUDUSD,M15)

Общие Вводные параметры

Переменная	Значение
Main Settings	
01 EA magic number	7799
01 Trading period start (hours)	9
01 Trading period end (hours)	23
Enable logging	true
Neutral broker-compatible order comment	Trade
Working timeframe	15 Minutes
01 Slippage (points)	3
Risk Management	
Lot calculation mode	Risk percent
Fixed lot size (if Fixed mode)	0.01
Risk per trade (% of free margin)	3.0
Reduce lot on high drawdown	true
Drawdown % to start reducing lot	20.0
01 Max open trades	3
01 Max allowed spread	20
Real SL / TP	
01 Real stop loss (points), 0 = off	0
01 Real take profit (points), 0 = off	0
Virtual SL / TP	
Enable virtual stop loss	false
01 Virtual stop loss (points)	0
Enable virtual take profit	false
01 Virtual take profit (points)	0
Trailing Stop	
Enable trailing stop	true
Average entry price for all positions	true
Activation profit (points)	100.0
Distance from price to SL (points)	60.0
Minimum SL improvement before modify (points)	5.0
Order Distance	
01 Fixed distance if dynamic=false (points)	100
ATR-based inter-order distance	true
01 ATR period for dynamic distance	12

Загрузить Сохранить

OK Отмена Сброс

Figure 5. Example of the Inputs window: Main Settings, Risk Management, SL/TP, Trailing Stop, and Order Distance.

KNN Pattern Hunter 2.0 (AUDUSD,M15)

Общие Вводные параметры

Переменная	Значение
Price method for analysis	(High + Low) / 2
01 Target horizon (bars)	3
Minimum target return (%)	0.07
KNN decision method	Distance-weighted
01 K nearest neighbors	7
Signal strength threshold	0.12
01 Maximum pattern database	500
01 Minimum patterns to start	100
01 History depth for training	1000
Indicator Settings	
01 RSI_Period	14
01 ATR_Period	14
01 FastMA_Period	8
01 SlowMA_Period	26
01 MACD_Fast	12
01 MACD_Slow	24
01 MACD_Signal	8
01 Correlation_Period	24
Adaptive Normalization	
01 RollingWindowSize	200
01 RecalibrationInterval	25
EnableAdaptiveWindow	true
AdaptationRate	0.05
Signal Filters	
UseKNN_Prediction	true
UseTrendSignal	true
UseCrossSignal	true
UseVolatilityFilter	false
MinVolatility	0.3
System Settings	
EnableModelPersistence	true
ModelFileName	KNN_Model
AutoSaveModel	true
01 AutoSaveInterval	3600

Загрузить Сохранить

OK Отмена Сброс

Figure 6. Example of the Inputs window: KNN, indicators, normalization, filters, and model persistence.

Main Settings

Parameter	Default	Purpose
MagicNumber	2356	Identifier for EA positions. Used to select and manage only its own positions.
SessionStart	8	Start of the trading window, in server time.
SessionEnd	22	End of the trading window, in server time. Start/End equality = 24/7.
EnableDebugLogging	false	Detailed diagnostic messages in the log.
InpOrderComment	Trade	Comment attached to market orders.
AnalysisPeriod	H1	Timeframe used to build features and trading decisions.
Slippage	3 points	Allowed deviation for CTrade operations.

Risk Management

Parameter	Default	Purpose
LotCalculationMode	FIXED	LOT_MODE_FIXED or LOT_MODE_FREE_MARGIN.
FixedLotSize	0.01	Fixed volume. Also used as a base value in some lot-normalization fallback scenarios.
RiskPercent	1.0%	In FREE_MARGIN mode, the share of free margin allocated to the margin requirement of a new position. This is not SL-based risk.
UseEquityProtection	true	When the drawdown threshold is reached, halves effective RiskPercent in FREE_MARGIN mode. It does not block trading.
MaxDrawdownPercent	20.0%	Drawdown threshold relative to the highest recorded equity.
MaxConcurrentTrades	3	Maximum number of simultaneously managed positions for the current symbol and MagicNumber.
MaxSpreadPoints	40	Maximum allowed current spread in points; 0 disables the limit.

Real SL / TP

Parameter	Default	Purpose
StopLoss	300 points	Real SL sent to the broker when opening. 0 = no real SL.
TakeProfit	300 points	Real TP sent to the broker when opening. 0 = no real TP.

Virtual SL / TP

Parameter	Default	Purpose
UseVirtualSL	false	Enable virtual SL.
VirtualStopLoss	0 points	Virtual SL distance.
UseVirtualTP	false	Enable virtual TP.
VirtualTakeProfit	0 points	Virtual TP distance.

Trailing Stop

Parameter	Default	Purpose
UseTrailing_Stop	true	Enables extended trailing. When false, the code still runs basic trailing if TrailingDistance > 0.
UseAveragePrice	true	For extended trailing, activation is calculated from the volume-weighted average price of all positions in the direction.
TrailingStart	100 points	Trailing activation threshold.
TrailingDistance	50 points	Distance from the current Bid/Ask to the new SL. 0 completely disables trailing.
TrailingStep	5 points	Minimum SL improvement before modification.

Order Distance

Parameter	Default	Purpose
FixedDistancePoints	100 points	Minimum distance between the current price and an existing position when dynamic distance is disabled.
UseDynamicDistance	true	Use ATR-based distance.
DynamicDistanceAtrPeriod	12	ATR period used for distance.
DynamicDistanceMultiplier	1.2	Distance = ATR in points × multiplier.

KNN Settings

Parameter	Default	Purpose
PriceMethod	Close	Close / HL2 / Typical. Used in the price feature for Trend Signal and in the pattern reference price.
TargetBars	4	Target-return horizon in bars.
MinTargetReturn	0.05%	If the absolute target return is below this threshold, target becomes 0.
KNN_Method	Weighted	Mean / Weighted / Median.
KNN_Neighbors	5	Number of nearest neighbors K.
KNN_Threshold	0.5	Base KNN classification threshold in forecast-percent units.
KNN_MaxPatterns	500	Maximum pattern-database size.
KNN_MinPatterns	50	Minimum number of patterns required to allow a KNN signal.
KNN_HistoryBars	1000	Maximum historical training depth.

Indicator Settings

Parameter	Default	Purpose
RSI_Period	14	RSI period.
ATR_Period	14	ATR period for the feature and Dashboard.
FastMA_Period	10	Fast SMA period.
SlowMA_Period	20	Slow SMA period.
MACD_Fast	12	Fast MACD period.
MACD_Slow	26	Slow MACD period.
MACD_Signal	9	MACD signal line.
Correlation_Period	20	Correlation window for price and volume changes.

Adaptive Normalization

Parameter	Default	Purpose
RollingWindowSize	200	Rolling Z-score normalization window size.
RecalibrationInterval	50	Recalibration interval by pattern count; the code also checks elapsed time and volatility change.
EnableAdaptiveWindow	true	Allows the window size to adapt when volatility changes significantly.
AdaptationRate	0.05	Exponential adaptation rate; the code limits the valid value to (0; 0.5].

Signal Filters

Parameter	Default	Purpose
UseKNN_Prediction	true	Enables KNN classification as one signal source.
UseTrendSignal	true	Enables comparison of PriceMethod on the latest and previous closed bars.
UseCrossSignal	true	Enables the Fast SMA / Slow SMA crossover check between the two latest closed bars.
UseVolatilityFilter	false	Despite its name, this is actually an absolute price/volume correlation filter.
MinVolatility	0.30	Minimum absolute correlation when the filter is enabled.

System Settings

Parameter	Default	Purpose
EnableModelPersistence	true	Allows loading/saving of the binary model.
ModelFileName	KNN_Model	Base file name; the actual name also contains the symbol and timeframe.
AutoSaveModel	true	Allows automatic saving.
AutoSaveInterval	3600 s	Auto-save interval; checked on ticks rather than through OnTimer.

6. How the KNN forecast and trading signal are formed

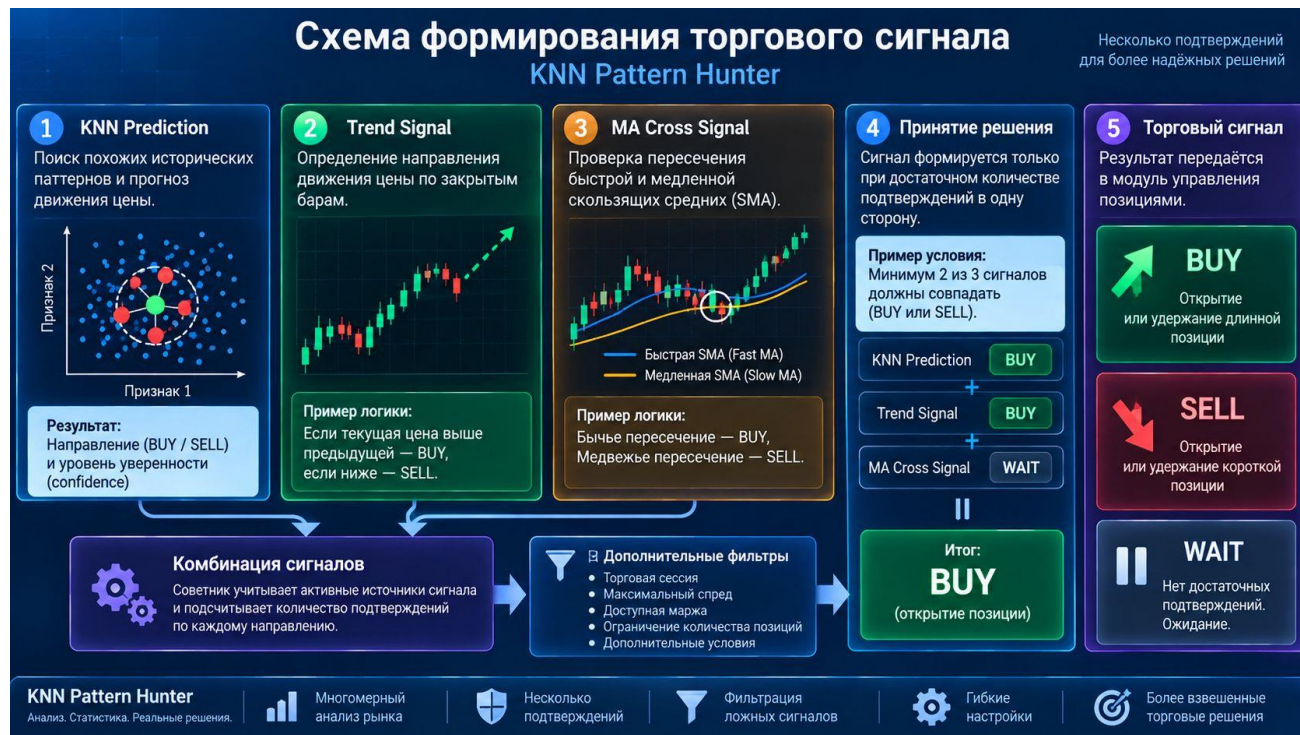


Figure 7. Conceptual BUY / SELL / WAIT signal flow. For exact behavior, use the rules below.

6.1. KNN source

The latest closed bar (shift = 1) is used for the trading decision. Its features are built, KNN Classify is executed, and the result is BUY / SELL / WAIT.

6.2. Trend Signal

Trend Signal compares the selected PriceMethod at shift 1 and shift 2: higher → BUY, lower → SELL, equal → neutral.

6.3. MA Cross Signal

A classic crossover of the two SMAs is checked between shift 2 and shift 1: Fast crosses above Slow → BUY; Fast crosses below Slow → SELL; otherwise neutral.

6.4. Combination rule

Scenario	Result
BUY votes exceed SELL votes and BUY ≥ half of the enabled sources	BUY
SELL votes exceed BUY votes and SELL ≥ half of the enabled sources	SELL
Otherwise	WAIT

Two-source edge case

When only two sources are enabled, one BUY and one neutral source already satisfy the BUY condition: BUY >

SELL and BUY $\geq 50\%$ of the sources. Therefore, the word “majority” in the old guide was too broad; “lead + minimum share” is more precise.

6.5. When a position is opened

- When a new bar appears, the EA runs ProcessNewBar() once.
- Trading conditions are checked first.
- A new historical pattern with a known target is added.
- MaxConcurrentTrades is checked.
- The trading signal is calculated.
- The trade volume is calculated.
- BUY/SELL triggers a market entry; WAIT does not.

7. Position management and averaging



Figure 8. Presentation diagram of position management. Important clarification: in the current code, the averaging lot is not multiplied by coefficient k .

7.1. Real SL/TP

When a market position is opened, the real StopLoss and TakeProfit are sent to the broker. The code accounts for StopLevel and FreezeLevel and increases the minimum distance when required.

7.2. Virtual SL/TP

Virtual levels are not sent to the broker. OnTick() continuously checks the price and closes the position when the level is reached. If the terminal or EA is stopped, this monitoring does not run.

7.3. Trailing

- UseTrailing_Stop = true: extended trailing is used. With UseAveragePrice = true, activation is calculated from the volume-weighted average price of all positions in the direction.
- UseTrailing_Stop = false: basic trailing still runs. Therefore, this is not a complete trailing-stop disable switch.
- TrailingDistance = 0: trailing is actually disabled.
- TrailingStep defines the minimum SL improvement required before another modification.

7.4. Averaging

- Averaging works only when MaxConcurrentTrades > 1 and the trading-condition checks pass.
- For BUY, a new position is added when Ask is at least the adaptive distance below the price of the latest BUY position; for SELL, when Bid is above the price of the latest SELL position by that distance.
- Adaptive distance = $ATR(\text{DynamicDistanceAtrPeriod}) \times \text{DynamicDistanceMultiplier}$ in points. When UseDynamicDistance = false, FixedDistancePoints is used.
- The current full trading signal is checked before adding a position. BUY is averaged only when the current signal is BUY; SELL only when the current signal is SELL.
- Averaging volume: the average volume of existing positions in the direction; in FREE_MARGIN mode it is additionally capped by the current CalculateLotSize().
- The grid is limited by MaxConcurrentTrades.

Averaging risk

The mechanism is not a classic martingale based on a lot-increase formula, but the grid itself increases exposure when price moves against the position. Risk should be assessed by total volume and the maximum possible number of positions.

8. Model saving and loading

8.1. File name

The actual file name follows this pattern: ModelFileName_SYMBOL_TIMEFRAME_adaptive.bin. For example, with base name KNN_Model and EURUSD H1, the file is KNN_Model_EURUSD_H1_adaptive.bin.

8.2. What is saved

- Raw features for each pattern, target, time, and reference price.
- Parameters required to restore the adaptive model state.
- On loading, normalized features are not treated as authoritative: normalization is rebuilt from the raw features.

8.3. Compatibility

Loading is rejected if saved maxPatterns, feature count, K, PriceMethod, TargetBars, or MinTargetReturn do not match the current settings. The stored normalization parameters are also validated.

If the loaded model contains fewer patterns than KNN_MinPatterns

The current code does not simply “continue” the existing database. It calls TrainKNNModel(), which clears the current database and rebuilds it from scratch. This corrects an inaccurate description in the old guide.

8.4. Auto-save

AutoSaveInterval is specified in seconds but checked inside OnTick(). The EA does not use a separate OnTimer() for saving. In addition, during deinitialization the model is automatically saved if EnableModelPersistence and AutoSaveModel are enabled and enough patterns have been accumulated.

9. Recommended setup and testing workflow

Do not treat the ranges as guarantees

The ranges below are practical starting points from the original guide. They do not demonstrate profitability and must be validated for the specific symbol, broker, spread, commission, and account type.

1. Start with Strategy Tester and a demo account.
2. For initial testing, use M15 or higher; M1/M5 may be noisier.
3. For H1, reasonable starting ranges are KNN_HistoryBars 1000–5000, KNN_MaxPatterns 300–1000, KNN_MinPatterns 50–200, and RollingWindowSize 100–500.
4. KNN_Neighbors: a practical starting range is 3–10.
5. KNN_Threshold: a practical starting range is 0.3–0.7, but the effective threshold also depends on volatilityRatio.
6. Set MinTargetReturn according to the instrument's typical price-movement scale.
7. For safer initial testing, use a small FixedLotSize and realistic MaxConcurrentTrades/MaxSpreadPoints limits.
8. If protection must remain active when the terminal is disconnected, use real SL/TP; virtual levels are not server-side protection.
9. Enable averaging only after a separate stress test of the position grid.
10. During optimization, control overfitting: KNN is sensitive to the structure of the historical sample.
11. In the tester, include commissions, swaps, and a spread that is as close to real conditions as possible.

9.1. What to test separately

Scenario	What to check
KNN	Stability when changing K, Threshold, HistoryBars, and MinTargetReturn.
Normalization	Behavior when changing RollingWindowSize, RecalibrationInterval, and AdaptationRate.
Filters	Results with KNN-only, KNN+Trend, KNN+MA, and all three sources.

Risk	Fixed vs FREE_MARGIN, drawdown protection, and MaxConcurrentTrades.
Exits	Real SL/TP, virtual SL/TP, and trailing.
Averaging	Different ATR multipliers, MaxConcurrentTrades, and signal directions.
Execution	StopLevel/FreezeLevel, spread, slippage, and partial/full closures.

10. Important limitations and implementation details

- RiskPercent in LOT_MODE_FREE_MARGIN is the share of free margin allocated to the margin requirement of a new position. It is NOT the percentage of capital that will be lost if the SL is hit.
- When the drawdown threshold is exceeded, UseEquityProtection does not block trading. It only halves effective RiskPercent, and only in FREE_MARGIN mode.
- UseVolatilityFilter is the code actually filters the absolute correlation between price changes and volume changes.
- KNN Confidence is normalized forecast strength relative to the threshold, not a probability.
- UseTrailing_Stop=false does not disable basic trailing when TrailingDistance > 0.
- Virtual SL/TP depend on OnTick(). They do not trigger when the terminal is stopped.
- FastMA_Period >= SlowMA_Period produces a warning rather than a validation error.
- RiskPercent > 0 is validated even when LOT_MODE_FIXED is selected.
- AutoSaveInterval is checked on ticks; no separate OnTimer() is used.
- If the loaded model contains fewer than KNN_MinPatterns, the current database is cleared and a full TrainKNNModel() run is performed.
- When KNN_MaxPatterns is reached, the oldest pattern is removed.
- The trading decision uses only the closed bar at shift=1; open-position management runs on ticks.
- Manual BUY/SELL passes the trading checks and position limit, but does not pass the separate IsNewPositionAllowed() check used by automatic entries. Therefore, a manual button is not an exact copy of an automatic entry.

Important

All trading involves risk. This document describes the technical behavior of the EA and is not investment advice.