```
!python kics_real.py

!python kics_surrogate.py

=== Phase 2-2: Training AI Surrogate Model ===
[-] Training MLP...

[-] Validating Scalability...
[-] Scalability Test (Asset: 10B KRW): MAPE = 0.0127%
[SUCCESS] AI Brain is Robust & Scalable!
Figure(800x600)

!python regime.py

!python system.py

=== Dynamic Shield System Test (Real Data) ===
[-] HMM 모델 학습 시작 (Data shape: (5292, 4))...
[-] 모델 학습 완료. 상태 매핑: {2: 'Normal', 0: 'Transition', 1: 'Panic'}
Start Date: 2004-07-26 00:00:00
Initial State: [13.22813     0.          -0.01771999  0.3679      1.5
0.        ]
Step 1: 2004-07-27 | Regime: Normal | K-ICS: 90.9% |
CDS_Proxy(YieldSpread): 0.37
Step 2: 2004-07-28 | Regime: Normal | K-ICS: 97.7% |
CDS_Proxy(YieldSpread): 0.37
Step 3: 2004-07-29 | Regime: Normal | K-ICS: 99.2% |
CDS_Proxy(YieldSpread): 0.37
Step 4: 2004-07-30 | Regime: Normal | K-ICS: 93.5% |
CDS_Proxy(YieldSpread): 0.37
Step 5: 2004-08-02 | Regime: Normal | K-ICS: 90.7% |
CDS_Proxy(YieldSpread): 0.37

!python ppo_trainer.py

python: can't open file 'c:\\Users\\PC\\Desktop\\Quant\\한화\\src\\
ppo_trainer.py': [Errno 2] No such file or directory

!python proof_risk_paradox.py

============================================================
Phase 5.1: Risk Paradox Proof
============================================================

[Correlation: -0.6]
  Optimal Hedge Ratio: 0.0%
  SCR Ratio at Optimal: 0.1116
  SCR Ratio at 100% Hedge: 0.1000
  Capital Savings: 10.38%
  Paradox Proven: YES ✓
```

```
[Correlation: -0.4]
  Optimal Hedge Ratio: 0.0%
  SCR Ratio at Optimal: 0.1064
  SCR Ratio at 100% Hedge: 0.1000
  Capital Savings: 5.98%
  Paradox Proven: YES ✓

[Correlation: -0.2]
  Optimal Hedge Ratio: 10.0%
  SCR Ratio at Optimal: 0.1019
  SCR Ratio at 100% Hedge: 0.1000
  Capital Savings: 1.82%
  Paradox Proven: YES ✓

[Correlation: 0.0]
  Optimal Hedge Ratio: 100.0%
  SCR Ratio at Optimal: 0.1000
  SCR Ratio at 100% Hedge: 0.1000
  Capital Savings: 0.00%
  Paradox Proven: NO

[Correlation: 0.2]
  Optimal Hedge Ratio: 100.0%
  SCR Ratio at Optimal: 0.1000
  SCR Ratio at 100% Hedge: 0.1000
  Capital Savings: 0.00%
  Paradox Proven: NO

============================================================
[SUCCESS] Risk Paradox Proven!
  3/5 scenarios show the paradox
Figure(1200x600)

[Saved] risk_paradox_proof.png

!python solvency_visualizer.py

============================================================
Phase 5.2: Solvency Analysis (COVID-19 Scenario)
============================================================

[100% Hedge]
  Min K-ICS: 1449.6%
  Final K-ICS: 1449.6%

[80% Fixed]
  Min K-ICS: 1159.5%
  Final K-ICS: 1212.9%

[Dynamic Shield]
```

```
  Min K-ICS: 1437.0%
  Final K-ICS: 1547.1%
Figure(1400x1200)

[Saved] kics_defense_result.png

!python stress_safety.py

============================================================
Phase 5.3: Safety Layer Stress Test
============================================================

[Test 1] VIX > 40 Injection Test
----------------------------------------
Step  1 | VIX: 15 | Hedge: 0.60 | Action: 3 | CRITICAL: K-ICS < 100%,
FORCE HEDGE 100%
Step  2 | VIX: 20 | Hedge: 0.70 | Action: 3 | CRITICAL: K-ICS < 100%,
FORCE HEDGE 100%
Step  3 | VIX: 25 | Hedge: 0.80 | Action: 3 | CRITICAL: K-ICS < 100%,
FORCE HEDGE 100%
Step  4 | VIX: 30 | Hedge: 0.90 | Action: 3 | CRITICAL: K-ICS < 100%,
FORCE HEDGE 100%
Step  5 | VIX: 35 | Hedge: 1.00 | Action: 3 | CRITICAL: K-ICS < 100%,
FORCE HEDGE 100%
Step  6 | VIX: 40 | Hedge: 1.00 | Action: 3 | CRITICAL: K-ICS < 100%,
FORCE HEDGE 100%
  >>> Emergency: Gradual De-risking Triggered <<<
Step  7 | VIX: 45 | Hedge: 1.00 | Action: 3 | CRITICAL: K-ICS < 100%,
FORCE HEDGE 100%
  >>> Emergency: Gradual De-risking Triggered <<<
Step  8 | VIX: 50 | Hedge: 1.00 | Action: 3 | CRITICAL: K-ICS < 100%,
FORCE HEDGE 100%
  >>> Emergency: Gradual De-risking Triggered <<<
Step  9 | VIX: 55 | Hedge: 1.00 | Action: 3 | CRITICAL: K-ICS < 100%,
FORCE HEDGE 100%
  >>> Emergency: Gradual De-risking Triggered <<<
Step 10 | VIX: 50 | Hedge: 1.00 | Action: 3 | CRITICAL: K-ICS < 100%,
FORCE HEDGE 100%
  >>> Emergency: Gradual De-risking Triggered <<<
Step 11 | VIX: 45 | Hedge: 1.00 | Action: 3 | CRITICAL: K-ICS < 100%,
FORCE HEDGE 100%
  >>> Emergency: Gradual De-risking Triggered <<<
Step 12 | VIX: 40 | Hedge: 1.00 | Action: 3 | CRITICAL: K-ICS < 100%,
FORCE HEDGE 100%
  >>> Emergency: Gradual De-risking Triggered <<<
Step 13 | VIX: 35 | Hedge: 1.00 | Action: 3 | CRITICAL: K-ICS < 100%,
FORCE HEDGE 100%
Step 14 | VIX: 30 | Hedge: 1.00 | Action: 3 | CRITICAL: K-ICS < 100%,
FORCE HEDGE 100%
Step 15 | VIX: 25 | Hedge: 1.00 | Action: 3 | CRITICAL: K-ICS < 100%,
```

```
FORCE HEDGE 100%
Step 16 | VIX: 20 | Hedge: 1.00 | Action: 3 | CRITICAL: K-ICS < 100%,
FORCE HEDGE 100%

[Test 2] Gradual Increase Verification
----------------------------------------
Max single-step hedge change: 0.10
[PASS] Hedge ratio changed gradually (max step <= 0.15)

[Test 3] K-ICS < 100% Penalty Test
----------------------------------------
Forced K-ICS Ratio: 3.0%
Agent Response: CRITICAL: K-ICS < 100%, FORCE HEDGE 100%
[PASS] Agent correctly responded to K-ICS < 100% with maximum hedge
increase

============================================================
Stress Test Results
============================================================
✓ Emergency De-risking: TRIGGERED
✓ Gradual Increase: CONFIRMED

[SUCCESS] Safety Layer passed all stress tests!

!python backtest.py

============================================================
Phase 5.4: Backtesting & Performance Analysis (With Real AI)
[v4.0] Anti-Overfitting: 실제 데이터 사용, Train/Test 분리
============================================================

[Info] □ Real AI Model loaded successfully from: ppo_kics.zip

[Scenario: NORMAL]
[실제 데이터 로드] 총 5292 일
  -> 학습용: 3704 일 (70%)
  -> 테스트용: 1588 일 (30%)

[Scenario: 2008_CRISIS]

[Scenario: 2020_PANDEMIC]

============================================================
Performance Summary (All Scenarios)
============================================================
                  CAGR   Sharpe     MDD     RCR  Avg_SCR  Net_Benefit
Strategy
100% Hedge     -0.0040   0.0000 -0.0079  0.0000   0.1000      -0.7937
80% Fixed      -0.0026 -10.3745 -0.0080  0.1322   0.1008      -0.5510
Dynamic Shield -0.0040 -10.9376 -0.0080  0.0024   0.1000      -0.7857
```

```
Rule-based      0.0010  -4.5708 -0.0100  0.5554   0.1025      -0.2032
Figure(1400x1000)

[Saved] backtest_result_ai.png

!python advanced_viz.py

============================================================
Phase 5.5: Advanced Visualization (XAI)
============================================================

[Plot 1] Counterfactual Dashboard (Decision Boundary)
--------------------------------------------------
Figure(1200x600)
[Saved] counterfactual_dashboard.png

[Plot 2] Efficient Frontier (Risk vs Cost)
--------------------------------------------------

[Info] □ Real AI Model loaded successfully from: ppo_kics.zip
[실제 데이터 로드] 총 5292 일
  -> 학습용: 3704 일 (70%)
  -> 테스트용: 1588 일 (30%)
Figure(1000x800)
[Saved] efficient_frontier.png

[Efficient Frontier Summary]
  100% Hedge     : Risk=10.00%, Cost=60.00%
  80% Fixed      : Risk=10.10%, Cost=48.00%
  Rule-based     : Risk=10.28%, Cost=33.69%
  Dynamic Shield : Risk=10.00%, Cost=59.21%

============================================================
[COMPLETE] All advanced visualizations generated!
  1. counterfactual_dashboard.png
  2. efficient_frontier.png
============================================================

!python shap_analysis.py

[WARNING] SHAP not installed. Run: pip install shap
============================================================
Phase 6.2.1: SHAP - Why Not 100% Hedge?
============================================================

[Feature Importance Analysis]
--------------------------------------------------

Normal (Natural Hedge):
  Correlation: [-0.6, -0.2)
  Optimal Hedge Ratio: 0.4%
```

```
  Average SCR: 0.1036

Transition:
  Correlation: [-0.2, 0.5)
  Optimal Hedge Ratio: 0.3%
  Average SCR: 0.0979

Panic:
  Correlation: [0.5, 0.9)
  Optimal Hedge Ratio: 0.9%
  Average SCR: 0.0934

============================================================
WHY NOT 100% HEDGE?
============================================================

[Normal Regime: Correlation = -0.4]
  100% Hedge: SCR=0.1000, Annual Cost=50.40%
   80% Hedge: SCR=0.1015, Annual Cost=40.32%
  SCR Difference: -0.15%p (80% is BETTER)
  Cost Savings: 10.08%p

[CONCLUSION]
  1. Natural Hedge 효과: 주식-환율 음의 상관관계로 분산 효과
  2. 헤지 비용 절감: 불필요한 오버헤지 비용 제거
  3. Risk Paradox: 적정 헤지가 완전 헤지보다 위험이 낮음
Figure(1400x1000)

[Saved] shap_why_not_analysis.png

!python phase6_final_review.py


======================================================================
FINAL REVIEW SUMMARY
======================================================================
======================================================================
Phase 6.1: Logic Consistency Check
======================================================================

[Check 1] Risk Paradox Proof
--------------------------------------------------
  SCR Ratio at 80% Hedge: 0.1015
  SCR Ratio at 100% Hedge: 0.1000
  [PASS] 80% 헤지가 100% 헤지보다 높은 지급여력비율!

[Check 2] Safety Layer Operation
--------------------------------------------------
  VIX=45 상황에서 에이전트 반응: CRITICAL: K-ICS < 100%, FORCE HEDGE 100%
  [PASS] Emergency De-risking Triggered!
```

```
[Check 3] Surrogate Model Accuracy
--------------------------------------------------
=== Phase 2-2: Training AI Surrogate Model ===
[-] Training MLP...

[-] Validating Scalability...
[-] Scalability Test (Asset: 10B KRW): MAPE = 0.0137%
[SUCCESS] AI Brain is Robust & Scalable!
Figure(800x600)
  Real SCR: 0.0932
  Pred SCR: 0.0932
  Error Rate: 0.00%
  [PASS] Surrogate 오차율 5% 미만!

======================================================================
Phase 6.2: Award-Winning Action Items
======================================================================

[Item 1] 'Why Not' Analysis (SHAP)
--------------------------------------------------
  [PENDING] 시각화 파일 생성 필요
            Run: python src/validation/shap_analysis.py

[Item 2] Efficient Frontier
--------------------------------------------------
  [PENDING] 시각화 파일 생성 필요

[Item 3] RCR (Risk-Cost Ratio) Metric
--------------------------------------------------
  [PASS] RCR 계산 로직 구현 완료

[Item 4] Code Philosophy Annotation
--------------------------------------------------
  [PASS] 'Capital Optimization, not Prediction' 철학 명시됨

======================================================================
OVERALL STATUS
======================================================================

[Logic Consistency]
  □ risk_paradox
  □ safety_layer
  □ surrogate_error

[Award-Winning Items]
  □ why_not_analysis
  □ efficient_frontier
  □ rcr_metric
  □ code_philosophy
```

```
[ACTION REQUIRED]
  - Award-Winning 항목 완성 필요
```