交易系统的泛属性状态编号机制与参数反馈优化闭环结构（代码）

- 作者：GaoZheng

- 日期：2025-03-19
- 版本：v1.0.0

```
(*---清空环境---*)
ClearAll[S, P, SamplePaths, ObservedValues, T, MicroDifferential,
  PathIntegralLogic, params, DeriOptimize, InferTopology,
  InferAlgebra, GcpolaaOptimizeDynamic, PredictEvolution];

(*---定义状态集合 S（中文）---*)
S = {
  "账户健康", "账户压力", "账户极端", "价格上涨",
  "价格下跌", "价格剧烈波动", "策略盈利","策略小亏",
  "策略大亏", "趋势确认", "趋势反转", "趋势震荡",
  "持仓增加", "持仓减少", "持仓锁仓", "波动率上升",
  "波动率下降", "波动率稳定"};

(*---定义属性集合 P（包括波动参数）---*)
P = <|"账户健康" -> <|"净值" -> 1.0, "敞口" -> 0.2, "收益" -> 0.5, "波动" -> 0.1|>,
     "账户压力" -> <|"净值" -> -0.7, "敞口" -> 0.7, "收益" -> -0.3, "波动" -> 0.4|>,
     "账户极端" -> <|"净值" -> -1.0, "敞口" -> 1.0, "收益" -> -0.8, "波动" -> 0.6|>,
     "价格上涨" -> <|"净值" -> 0.3, "敞口" -> 0.1, "收益" -> 0.8, "波动" -> 0.2|>,
    "价格下跌" -> <|"净值" -> -0.3, "敞口" -> 0.1, "收益" -> -0.8, "波动" -> 0.2|>,
     "价格剧烈波动" -> <|"净值" -> 0.0, "敞口" -> 0.2, "收益" -> 0.0, "波动" -> 0.9|>,
     "策略盈利" -> <|"净值" -> 0.7, "敞口" -> 0.3, "收益" -> 0.9, "波动" -> 0.1|>,
    "策略小亏" -> <|"净值" -> -0.2, "敞口" -> 0.4, "收益" -> -0.3, "波动" -> 0.2|>,
     "策略大亏" -> <|"净值" -> -0.6, "敞口" -> 0.7, "收益" -> -0.7, "波动" -> 0.4|>,
     "趋势确认" -> <|"净值" -> 0.5, "敞口" -> 0.2, "收益" -> 0.7, "波动" -> 0.2|>,
    "趋势反转" -> <|"净值" -> -0.4, "敞口" -> 0.3, "收益" -> -0.6, "波动" -> 0.5|>,
     "趋势震荡" -> <|"净值" -> 0.1, "敞口" -> 0.1, "收益" -> 0.1, "波动" -> 0.6|>,
    "持仓增加" -> <|"净值" -> 0.2, "敞口" -> 0.9, "收益" -> 0.3, "波动" -> 0.3|>,
    "持仓减少" -> <|"净值" -> 0.1, "敞口" -> -0.8, "收益" -> 0.2, "波动" -> 0.2|>,
    "持仓锁仓" -> <|"净值" -> 0.0, "敞口" -> 0.0, "收益" -> 0.0, "波动" -> 0.1|>,
    "波动率上升" -> <|"净值" -> 0.0, "敞口" -> 0.4, "收益" -> 0.0, "波动" -> 1.0|>,
    "波动率下降" -> <|"净值" -> 0.0, "敞口" -> 0.2, "收益" -> 0.0, "波动" -> -1.0|>,
     "波动率稳定" -> <|"净值" -> 0.0, "敞口" -> 0.1, "收益" -> 0.0, "波动" -> 0.0|>|>;

(*---定义样本路径---*)
SamplePaths = {
  {"账户健康", "价格上涨", "策略盈利", "趋势确认", "持仓增加"},
  {"账户健康", "价格下跌", "策略小亏", "趋势震荡", "持仓减少"},
  {"账户压力", "价格下跌", "策略大亏", "趋势反转", "持仓减少"},
  {"账户极端", "价格剧烈波动", "策略大亏", "趋势反转", "波动率上升"},
  {"账户压力", "价格上涨", "策略小亏", "趋势震荡", "波动率稳定"},
```

```
  {"账户健康", "价格上涨", "策略盈利", "趋势确认", "波动率下降"}};

(*---定义观测得分---*)
ObservedValues = {3.5, 1.0, -3.0, -4.0, 0.5, 4.0};

(*---定义微分动力量子（包括波动参数）---*)
MicroDifferential[s1_, s2_, {wM_, wC_, wE_, wV_}] :=
  Module[{dM, dC, dE, dV}, dM = P[s2]["净值"] - P[s1]["净值"];
   dC = P[s2]["敞口"] - P[s1]["敞口"];
   dE = P[s2]["收益"] - P[s1]["收益"];
   dV = P[s2]["波动"] - P[s1]["波动"];
   wM dM + wC dC + wE dE + wV dV];

(*---路径积分逻辑性度量---*)
PathIntegralLogic[path_, {wM_, wC_, wE_, wV_}] :=
  Total[Table[
    Tanh[MicroDifferential[path[[i]],
      path[[i + 1]], {wM, wC, wE, wV}]], {i, Length[path] - 1}]];

(*---参数优化 DeriOptimize---*)
DeriOptimize[paths_, obsVals_] :=
  Module[{loss, res},
   loss[{wM_, wC_, wE_, wV_}] :=
    Total[(Table[
         PathIntegralLogic[path, {wM, wC, wE, wV}], {path, paths}] -
        obsVals)^2];
   res =
    NMinimize[{loss[{wM, wC, wE, wV}], -2 <= wM <= 2 && -2 <= wC <=
        2 && -2 <= wE <= 2 && -2 <= wV <= 2}, {wM, wC, wE, wV}];
   {wM, wC, wE, wV} /. Last[res]];

(*---推导局部代数规则 InferAlgebra---*)
InferAlgebra[paths_, params_] :=
  Flatten[Table[
    MicroDifferential[path[[i]], path[[i + 1]], params] == 0, {path,
     paths}, {i, Length[path] - 1}]];

(*---推导拓扑结构 InferTopology---*)
InferTopology[paths_, params_] :=
  Module[{T0}, T0 = Association[Table[state -> {}, {state, S}]];
```

```
    Do[Do[
      If[! MemberQ[T0[path[[i]]],
          path[[i + 1]]] && (MicroDifferential[path[[i]],
           path[[i + 1]], params] >= -0.5),
       AppendTo[T0[path[[i]]], path[[i + 1]]]], {i,
       Length[path] - 1}], {path, paths}];
    Association[
     KeyValueMap[#1 ->
        Select[#2, (Abs[MicroDifferential[#1, #, params]] <= 1.5) &] &,
       T0]]];

(*---动态路径优化 GcpolaaOptimizeDynamic---*)
GcpolaaOptimizeDynamic[init_, learningRate_ : 0.05] :=
  Module[{current = init, path = {init}, totalScore = 0,
    initLocalParams = params, localParams = params, diff, step = 1},
   Print["初始状态: ", current];
   Print["初始参数 (params) : ", localParams];
   While[T[current] =!= {},
    current =
     First@MaximalBy[T[current],
       Tanh[MicroDifferential[path[[-1]], #, localParams]] &];
    diff = MicroDifferential[path[[-1]], current, localParams];
    Print["第 ", step, " 步: 从 ", path[[-1]], " $$RightArrow] ", current,
      ", 局部微分压强 = ", N[diff], ", 当前参数 = ", N[localParams]];
    localParams =
     localParams + learningRate*Sign[{diff, diff, diff, diff}];
    totalScore +=
     Tanh[MicroDifferential[path[[-1]], current, localParams]];
    AppendTo[path, current];
    step++;];
   <|"Path" -> path, "FinalParams" -> localParams,
    "InitLocalParams" -> initLocalParams, "Score" -> totalScore|>];

(*---预测未来路径 PredictEvolution---*)
PredictEvolution[init_, stepsMax_ : 10, learningRate_ : 0.05,
   threshold_ : 0.3] :=
  Module[{current = init, path = {init}, totalScore = 0,
    initLocalParams = params, localParams = params, diff, step = 1,
    stop = False}, Print["初始状态: ", current];
   Print["初始参数 (params) : ", localParams];
```

```
    While[! stop && step <= stepsMax && T[current] =!= {},
     current =
      First@MaximalBy[T[current],
        Tanh[MicroDifferential[path[[-1]], #, localParams]] &];
     diff = MicroDifferential[path[[-1]], current, localParams];
     Print["第 ", step, " 步：从 ", path[[-1]], " $$RightArrow] ", current,
       "，局部微分压强 = ", N[diff], "，当前参数 = ", N[localParams]];
     localParams =
      localParams + learningRate*Sign[{diff, diff, diff, diff}];
     totalScore +=
      Tanh[MicroDifferential[path[[-1]], current, localParams]];
     AppendTo[path, current];
     If[Abs[diff] < threshold, Print["逻辑性塌缩或路径分岔检测：局部微分压强太小，停止演化"];
      stop = True;];
     step++;];
    <|"PredictedPath" -> path, "FinalParams" -> localParams,
     "InitLocalParams" -> initLocalParams, "TotalScore" -> totalScore|>];

(*---执行完整流程---*)
params = DeriOptimize[SamplePaths, ObservedValues];
algebraConstraints = InferAlgebra[SamplePaths, params];
T = InferTopology[SamplePaths, params];
optimizedResult = GcpolaaOptimizeDynamic["账户健康"];
predictedResult = PredictEvolution["账户压力"];

(*---最终输出---*)
Print@Dataset[<|"优化后的参数params" -> params,
    "推导的局部代数规则" -> Dataset@algebraConstraints, "推导的拓扑结构" -> Dataset@T,
     "最优路径与得分" -> optimizedResult, "预测路径参数与得分" -> predictedResult|>];
```

初始状态：账户健康

初始参数（params）：{-2.,2.,0.189653,-2.}

第 1 步：从 账户健康 $$RightArrow] 价格下跌，
局部微分压强 = 1.95345，当前参数 = {-2.,2.,0.189653,-2.}

第 2 步：从 价格下跌 $$RightArrow] 策略大亏，
局部微分压强 = 1.44897，当前参数 = {-1.95,2.05,0.239653,-1.95}

初始状态：账户压力

初始参数（params）：{-2.,2.,0.189653,-2.}

<|优化后的参数params->{-2.,2.,0.189653,-2.},
推导的局部代数规则->
{False,False,False,False,False,False,False,False,False,
False,False,False,False,False,False,False,False,False,
False,False,False,False,False,False},
推导的拓扑结构->
<|账户健康->{价格上涨,价格下跌},
账户压力->{},账户极端->{},价格上涨->{策略盈利,策略小亏},
价格下跌->{策略小亏,策略大亏},价格剧烈波动->{策略大亏},
策略盈利->{趋势确认},策略小亏->{},策略大亏->{},
趋势确认->{持仓增加,波动率下降},趋势反转->{},
趋势震荡->{波动率稳定},持仓增加->{},持仓减少->{},
持仓锁仓->{},波动率上升->{},波动率下降->{},波动率稳定->{}|>,
最优路径与得分->|
Path->{账户健康,价格下跌,策略大亏},
FinalParams->{-1.9,2.1,0.289653,-1.9},
InitLocalParams->{-2.,2.,0.189653,-2.},
Score->1.85046|>,
预测路径参数与得分->|
PredictedPath->{账户压力},
FinalParams->{-2.,2.,0.189653,-2.},
InitLocalParams->{-2.,2.,0.189653,-2.},
TotalScore->0|>|>

许可声明 (License)