

policy::PolicyEval
<ul style="list-style-type: none">-statespace: statespace.State[][][]-stateactions: java.util.Hashtable-statevalues: java.util.Hashtable-gamma: double-delta: double-theta: double-policy: policy.Policy
<ul style="list-style-type: none">+PolicyEval(double, double, policy.Policy): ctor+PolicyEval(): ctor+main(java.lang.String[]): void+getAction(statespace.State): java.lang.String+multisweep(): int+sweep(): double+updateValue(statespace.State): double+getActionProb(): double+getP(int, statespace.State): double+getReward(statespace.State): double+output(): void+filltable(java.io.File): void+printTable(statespace.Position): void+printList(statespace.Position): void

policy::PolicyIter
<ul style="list-style-type: none">-evaluation_runs: int-improvement_runs: int
<ul style="list-style-type: none">+PolicyIter(double, double): ctor+PolicyIter(): ctor+getAction(statespace.State): java.lang.String+dolteration(): void+doPolicyImprovement(): boolean+argmaxupdateValue(statespace.State): java.lang.String+main(java.lang.String[]): void+doPolicyEvaluationIteration(): int+multisweep_iteration(): int+sweep_iteration(): double+updateValue_iteration(statespace.State): double