B : Real = double
T : Matrix = OpenMPMatrix

CorrelationFunction

+getName(): string
+getAllCorrelation(P1: Point, P2: Point, corrMat: T, diagAdd: B, i: int, j: int, eqMemb: bool): void
+getAllCorrelationPartDer(P1: Point, P2: Point, corrMat: T, corrMatDer: T, diagAdd: B, i: int, j: int, eqMemb: bool, varNr: int): void

B : Real = double
T : Matrix = OpenMPMatrix

CorrelationFunctionGauss

-Thetas: T

+getName(): string
+getAllCorrelation(P1: Point, P2: Point, corrMat: T, diagAdd: B, i: int, j: int, eqMemb: bool): void
+getAllCorrelationPartDer(P1: Point, P2: Point, corrMat: T, corrMatDer: T, diagAdd: B, i: int, j: int, eqMemb: bool, varNr: int): void
-calcCorrelation(Point1: Point, Point2: Point): B
-calcCorrelationPartialDerivative(Point1: Point, Point2: Point, correlation: B, nrVar: size): B

B : Real = double
T : Matrix = OpenMPMatrix

CorrelationFunctionSpline

-Thetas: T

+getName(): string
+getAllCorrelation(P1: Point, P2: Point, corrMat: T, diagAdd: B, i: int, j: int, eqMemb: bool): void
+getAllCorrelationPartDer(P1: Point, P2: Point, corrMat: T, corrMatDer: T, diagAdd: B, i: int, j: int, eqMemb: bool, varNr: int): void
-calcCorrelation(Point1: Point, Point2: Point, p: vector): B
-calcCorrelationPartialDerivative(Point1: Point, Point2: Point, correlation: B, nrVar: size, withoutProduct: bool): B