

initQFC

Attaches QFC kernel to model struct. Depending on mean options attach zero mean functions and sets all related kernel parameters and dependencies to zero. If mean is polynom fitting, attaches meanPolyQFC as basis function to build polynom matrix H and sets a none zero mean function. Bypasses dataset inputs as they are. Kernel works on matrix data.

Syntax

```
Mdl = initQFC(Mdl)
```

Description

Mdl = initQFC(Mdl) loads quadratic fraction covariance function and basis function depending on mean in **Mdl** struct. Sets input function as bypass.

Input Arguments

Mdl struct with model parameter and training data.

Output Arguments

Mdl struct with attached kernel functionality

Requirements

- Other m-files required: None
- Subfunctions: QFC, meanPolyQFC
- MAT-files required: None

See Also

- [initKernel](#)
- [meanPolyQFC](#)
- [QFC](#)

Created on February 15, 2021 by Tobias Wulf. Copyright Tobias Wulf 2021.

```
function Mdl = initQFC(Mdl)

    % set QFC kernel function
    Mdl.kernelFun = @QFC;

    % set input transformation function to apply adjustments to
    % covariance function, here bypass inputs as they are, no transformation of
    % training data needed
    Mdl.inputFun = @(X) X;

    % set mean function to compute cosine and sine H matrix
    switch Mdl.mean
        % zero mean m(x) = 0
        case 'zero'
            % set polyDegree to -1 for no polynom indication
            Mdl.polyDegree = -1;

            % set basis function
            Mdl.basisFun = @(X) 0;

    % mean by polynom m(x) = H' * beta
```

```
case 'poly'
    % set basis function produces a (polyDeg+1)xN H matrix
    Mdl.basisFun = @(X) meanPolyQFC(X, Mdl.polyDegree);

% end mean select QFC kernel
otherwise
    error('Unknown mean function %.', Mdl.mean);
end
end
```