```
function K = MTGP_covPeriodicisoU(hyp, x, z, i)

% Stationary covariance function for a smooth periodic function, with period p:
%
% Based on the covPeriodicisoU.m function of the GPML Toolbox -
%   with the following changes:
%       - only elements of x(:,1:end-1)/z(:,1:end-1) will be analyzed,
%       - x(:,end)/z(:,end) will be ignored, as it contains only the label information%       - distance measure is fixed to 1
%       - independent of the label all x values will have the same hyp
%       - feature scaling hyperparameter is fixed to 1
%
% k(x,y) = sf2 * exp( -2*sin^2( pi*||x-y||/p ) )
%
% where the hyperparameters are:
%
% hyp = [ log(p)
%         log(sqrt(sf2)) ]
%
% modified by Robert Duerichen
% 04/02/2014

if nargin<2, K = '2'; return; end                   % report number of parameters
if nargin<3, z = []; end                                    % make sure, z exists
xeqz = numel(z)==0; dg = strcmp(z,'diag') && numel(z)>0;         % determine mode

n = size(x,1);

p   = exp(hyp(1));
sf2 = exp(2*hyp(2));

% precompute distances
if dg                                                                % vector kxx
  K = zeros(size(x(:,1),1),1);
else
  if xeqz                                                  % symmetric matrix Kxx
    K = sqrt(sq_dist(x(:,1)'));
  else                                                    % cross covariances Kxz
    K = sqrt(sq_dist(x(:,1)',z(:,1)'));
  end
end

K = pi*K/p;
if nargin<4                                                         % covariances
    K = sin(K); K = K.*K; K =   sf2*exp(-2*K);
else                                                                % derivatives
  if i==1
    R = sin(K); K = 4*sf2*exp(-2*R.*R).*R.*cos(K).*K;
  elseif i==2
    K = sin(K); K = K.*K; K = 2*sf2*exp(-2*K);
  else
    error('Unknown hyperparameter')
  end
end
```