drawNLPDplots.m

Table of Contents

Summary: Script to assess predictive power of a model. Computes NLPD and RelDiff of model predictions along the trajectory generated by the policy trained on it.

High-level steps

1. Initialize empty matrices
2. Loop over the trajectory gathering statistics
3. Draw plots

Code

```
try
    LP = zeros(last_size, 1);
    sampleLP = zeros(last_size, 1);
    sampleLPstd = zeros(last_size, 1);
    relDiff = zeros(last_size, 1);
    sampleSize = 500;
    fprintf(1, '\nCalculating NLPD and relative differences of predictions and exp
    for k=1:last_size
        [mu, sigma] = dynmodel.fcn(dynmodel, newdata(k,1:16)', 0.0001*eye(16));
        mu(difi) = mu(difi) + newdata(k, difi)';
        LP(k) = NLPD(newdata(k+1,1:12)', mu, sigma, 12);
        sample = mvnrnd(mu, sigma, sampleSize);
        aux = cellfun(@(v) NLPD(v', mu, sigma, 12), num2cell(sample, 2));
        sampleLP(k) = mean(aux);
        sampleLPstd(k) = std(aux);
        relDiff(k) = 100*mean((mu - newdata(k+1,1:12)')./newdata(k+1,1:12)');
        fprintf(1, '\b\b\b\b%4i', k);
    end
    aux = 1:last_size;
    figure(10)
    plot(aux, LP, 'k-',  ...
      aux, sampleLP, 'g-', ...
      aux, sampleLP + sampleLPstd, 'r:', ...
      aux, sampleLP - sampleLPstd, 'r:'  ); drawnow;
      xlabel('Steps');
      ylabel('NLPD');
   legend('Real', 'Optimal', '1-sigma belt');

    figure(11)
    plot(relDiff); drawnow;
    xlabel('Steps');
    ylabel('Relative Difference (%)');

catch ME
    disp('Error computing NLPD');
    disp('Exception:');
    disp(ME);
    disp('Covariance matrix:');
    disp(sigma);
end
```

Published with MATLAB® R2014a