

Solution to the exercises of Session 4

Aaron Ponti

- **Section 3.1 / 1**

```
>> [ h, p, ci, stats ] = ttest2( ...
    iris.SL( iris.species == 'setosa' ), ...
    iris.SL( iris.species == 'versicolor' ), ...
    0.05, 'both', 'unequal' );

>> stats.sd

ans =
    0.3525
    0.5162
```

These are the estimated *standard deviations* for the two samples. To get the variances just square them:

```
>> stats.sd .^ 2

ans =
    0.1242
    0.2664
```

- **Section 5 / 1**

We test for instance whether $\text{median}(\text{sampleX}) \geq \text{median}(\text{sampleY})$, instead of $\text{median}(\text{sampleX}) \neq \text{median}(\text{sampleY})$. This boils down to test whether $\text{median}(\text{sampleX}) - \text{median}(\text{sampleY}) \geq \text{realDiff}$ instead of $\text{abs}(\text{median}(\text{sampleX}) - \text{median}(\text{sampleY})) \geq \text{realDiff}$.

```
function p = testDiffMedianOneSided( x, y, N )

% Calculate the real difference (no absolute value!)
realDiff = median( x ) - median( y );

% Pool all measurements from both samples
allMeas = [ x y ];

% Initialize counter and lengths
count = 0;
nX    = numel( x );
n     = numel( allMeas );
```

```

% Now do the Monte Carlo stuff
for i = 1 : N

    % Randomize the measurements
    randData = allMeas( randperm( n ) );
    sampleX = randData( 1 : nX );
    sampleY = randData( nX + 1 : end );

    % Calculate the difference of the randomized samples
    % Here we calculate median( sampleX ) - median( sampleY ) instead of
    % abs( median( sampleX ) - median( sampleY ) )
    randDiff = median( sampleX ) - median( sampleY );

    % Is it larger than the real difference?
    if randDiff >= realDiff
        count = count + 1;
    end

end

% Calculate the p-value (add one pseudo-count)
p = ( count + 1 ) / ( N + 1 );

```

- **Section 5 / 2**

Here we compare the variances of samples from x and y instead of their medians.

```

function p = testDiffVar( x, y, N )

% Calculate the real difference between variances
realDiff = abs( var( x ) - var( y ) );

% Pool all measurements from both samples
allMeas = [ x y ];

% Initialize counter and lengths
count = 0;
nX     = numel( x );
n      = numel( allMeas );

% Now do the Monte Carlo stuff
for i = 1 : N

    % Randomize the measurements
    randData = allMeas( randperm( n ) );
    sampleX = randData( 1 : nX );
    sampleY = randData( nX + 1 : end );

    % Calculate the difference in variance of the

```

```
% randomized samples
randDiff = abs( var( sampleX ) - var( sampleY ) );

% Is it larger than the real difference?
if randDiff >= realDiff
    count = count + 1;
end

end

% Calculate the p-value (add one pseudo-count)
p = ( count + 1 ) / ( N + 1 );
```