

Linear modeling

8 solutions submitted (max: Unlimited) | [View my solutions](#)[Open Problem in MATLAB Online](#) ?

Introduction

EMG signal is the electrical signal associated with muscular contraction. Level of activity of the EMG signal increases with force; see Section 1.2.4 of the course book for details on the EMG signal.

In order to investigate the relationship between an independent variable and a dependent variable, it is common practice to fit a mathematical function or a curve to the data samples available, and then to examine the accuracy of the underlying model (goodness of fit). In the present lab assignment, you will investigate the appropriateness of linear fits (straight-line or linear models) to represent the variation of each of the parameters DR, MS, ZCR and TCR as a function of independent variable force.

A brief explanation of the parameters:

- The dynamic range (DR) of a signal is the difference between its maximum and minimum values over a specified duration. This range of values is also referred to as the peak-to-peak swing or range.
- The average power of a signal is provided by the mean-squared (MS) value over a specified duration of time; the average magnitude over the same duration is given by the root-mean-squared (RMS) value. With the mean of the EMG signal being zero, the RMS value is equal to the standard deviation of the signal.
- An approximate indicator of the level of activity in a signal is given by the number of times the signal crosses the zero line (assuming a mean value of zero; if the mean is not zero, it should be subtracted from the signal). A zero crossing is said to occur when the sign of a sample of the signal is different from the sign of the preceding sample. The average number of zero crossings over a certain time period is known as the zero-crossing rate (ZCR). It is expected that increased presence of high-frequency components in a signal will lead to larger ZCR. See Section 5.6 on statistical analysis of EMG signals.
- Turns count (TC) is the number of times the signal amplitude changes direction. In Willison method (used in this assignment) turns greater than 100 μV are counted, with the threshold selected to avoid counting insignificant fluctuations due to noise. TCR (Turns Count Rate) is the turns count divided by the time duration of the segment.

Data

Muscle force and EMG data from the original experiment are shown in Figure 1. The sampling rate is 2000 Hz for all provided signals, and the EMG sample values are in mV. The data segments have been prepared for the assignment and are available in `data.mat` file. The segments were cropped at their 70% of maximal value within the segment. The data variable is a 1x5 struct (for the 5 segments), and has the following fields:

- `t`: the time points for the segment
- `force`: muscle force data readily prepared in percent of maximal voluntary contraction
- `EMG`: the EMG data for the segment with its mean (DC) subtracted
- `length`: the number of samples in the segment

You can reach the data with the following syntax: for example `data(2).force` is the force signal from the second segment, `data(5).EMG` is the EMG signal of the fifth segment, and so on.

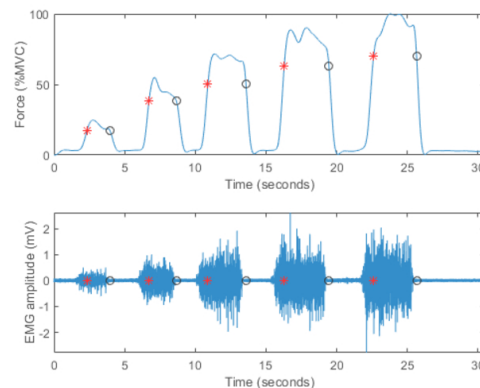


Figure 1. Muscle force and EMG signals of the experiment with segment starting and ending points marked with red stars and black circles, respectively. %MVC: percent of maximal voluntary contraction

Useful MATLAB commands:

`hold`, `max`, `min`, `diff`, `sign`, `abs`, `polyfit`, `polyval`, `find`

Task

Your task is to:

- Study the data by plotting it with respect to time. I.e. plot all the segments of the already normalized force (in %MVC) in a `subplot(2,1,1)`, and the EMG signal (in mV) in a `subplot(2,1,2)` against their time corresponding time axes. Your plot will be like Figure 1 above, with the discarded signal parts (not belonging to any segment) missing from the plot. Hint: you can plot the 5 segments with separate plot signals (using `hold on`), and keep the segments in their default colors.
- Calculate the average force exerted within each segment of the force signal.
- Compute the DR for each segment of the EMG signal.
- For each segment of the EMG signal, compute MS. Ensure that the MS values are computed using the appropriate number of samples for each segment. The MS value of a signal x over its duration of N samples is the following: $MS = \frac{1}{N} \sum_{i=0}^{N-1} x^2(i)$.
- For each segment of the EMG signal, compute ZCR **without** using `dsp.ZeroCrossingDetector`. Ensure that you normalize ZCR to zero crossings per second by dividing the number of zero crossings by the time duration of the corresponding segment. Do not remove the mean but assume the signal is zero mean already.
- For each segment of the EMG signal, compute TCR with the algorithm* given below.
- Using the `polyfit` and `polyval` functions in MATLAB, obtain a straight-line (linear) fit to represent the variation of each EMG parameter as a function of independent variable force. Superimpose the obtained linear models (straight-line fits) on their corresponding subplots of the preceding step.
- Compute the correlation coefficients (r) for each parameter with `corr` function.

*) TCR algorithm

To find the TCR, proceed as follows:

- Compute the derivative of the EMG signal (`diff`)
- Detect the points of change in its sign (defined as the MATLAB `sign` function, taking values in $\{-1, 0, 1\}$)
- A significant turn is found wherever the derivative signal changes sign, and the original signal differs by at least 100 μV between current and previous (significant or non-significant) turn points (not the previous samples).
- TCR is the amount of significant turns divided by the time duration of the corresponding segment

Note 1: A sign change between two of the derivative samples implies that there exists a zero crossing between them (as the derivative is considered also to be a continuous signal). This

vanishing of the derivative happens at an extrema point (min/max) of the original signal.

Note 2: If utilizing the MATLAB `sign` command, please also consider that it evaluates to 0 at 0.

Script

SaveResetMATLAB DocumentationOpen Problem in MATLAB Online

```
1 % The sampling rate is 2000 Hz
2 FS = 2000;
3
4 % Load the signals from data.mat into the struct 'data'
5 % << insert loading code here >>
6 load('data.mat');
7 % Number of segments
8 N = numel(data);
9
10 % Calculate average force of each segment (1xN vector)
11 AF = zeros(1, N);
12
13 % Calculate EMG dynamic range in each segment (1xN vector)
14 DR = zeros(1, N);
15
16 % Calculate EMG mean squared value in each segment (1xN vector)
17 MS = zeros(1, N);
18
19 % Calculate EMG zero crossing rate in each segment (1xN vector)
20 ZCR = zeros(1, N);
21
22 % Calculate EMG turns rate in each segment (1xN vector)
23 TCR = zeros(1, N);
24 for i = 1:N
25     AF(i) = mean(data(i).force);
26     DR(i) = max(data(i).EMG) - min(data(i).EMG);
27
28     signal_length = length(data(i).EMG);
29     MS(i) = sum((data(i).EMG).^2) / signal_length;
30
31     %ZCR calculation
32     time_duration = data(i).t(end) - data(i).t(1);
33     zero_crossings = sum(abs(diff(sign(data(i).EMG)))) / 2;
34     ZCR(i) = zero_crossings / time_duration;
35
36     % Calculate Turns Count Rate (TCR)
37     derivative = diff(data(i).EMG);
38     signs = sign(derivative);
39     turns = signs(1:end-1) .* signs(2:end);
40     turn_indices = find(turns <= 0) + 1;
41     extremes = data(i).EMG(turn_indices);
42     extreme_diff = diff(extremes);
43     valid_turns = find(abs(extreme_diff) > 0.1);
44     turn_count = length(valid_turns);
45     TCR(i) = turn_count / time_duration;
46 end
47 % Calculate the linear model coefficients for each parameter
48 % The models are in the form: parameter(force) = constant + slope * force,
49 % and the coefficients are stored in a 1x2 vectors: p_<param> = [slope constant]
50 % For example, p_DR(1) is the slope and p_DR(2) is the constant of the linear model mapping the average force into the dynamic range
51 % You can use the 'polyfit' (or the 'regress') command(s) to find the model coefficients
52 p_DR = polyfit(AF, DR, 1);
53 p_MS = polyfit(AF, MS, 1);
54 p_ZCR = polyfit(AF, ZCR, 1);
55 p_TCR = polyfit(AF, TCR, 1);
56
57 % Calculate correlation coefficients between the average forces and each of the parameters using 'corr'
58 c_DR = corr(AF', DR');
59 c_MS = corr(AF', MS');
60 c_ZCR = corr(AF', ZCR');
61 c_TCR = corr(AF', TCR');
```

Run Script

Previous Assessment: All Tests Passed

Run PretestSubmit

Correct average forces

Correct dynamic ranges

Correct mean squared values

Correct zero crossing rates

Correct turn count rates

Correct AF-DR correlation

Correct AF-MS correlation

Correct AF-ZCR correlation

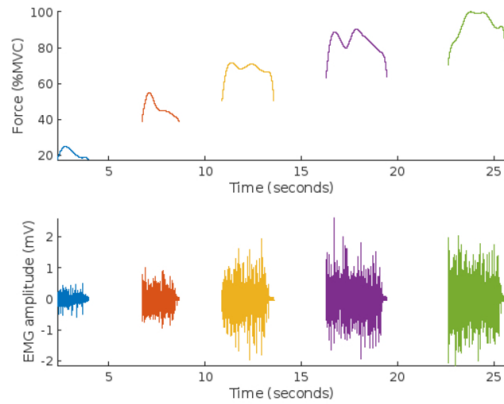
Correct AF-TCR correlation

✓ Plotting 1 - not a real test (Open to view) (Pretest)

Test Code:

```
1 N = numel(data); % number of segments
2
3 figure;
4
5 subplot(211);
6 hold on;
7 for i = 1:N
8     plot(data(i).t, data(i).force);
9 end
10 axis tight;
11 xlabel('Time (seconds)');
12 ylabel('Force (%MVC)');
13
14 subplot(212);
15 hold on;
16 for i = 1:N
17     plot(data(i).t, data(i).EMG);
18 end
19 axis tight;
20 xlabel('Time (seconds)');
21 ylabel('EMG amplitude (mV)');
22
```

Figures:

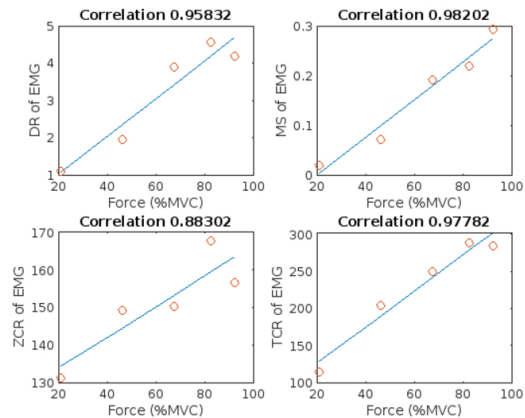


✓ Plotting 2 - not a real test (Open to view) (Pretest)

Test Code:

```
1 % Gathering data into arrays for looping
2 X = [DR; MS; ZCR; TCR];
3 MODELS = {p_DR, p_MS, p_ZCR, p_TCR};
4 LABELS = {'DR', 'MS', 'ZCR', 'TCR'};
5 CORRS = [c_DR, c_MS, c_ZCR, c_TCR];
6
7 % Plotting linear models and raw data
8 figure;
9 for i = 1:4
10     subplot(2,2,i);
11     line = polyval(MODELS{i}, AF); % line estimate
12     plot(AF, line);
13     hold on;
14     plot(AF, X(i,:), 'o');
15     xlabel('Force (%MVC)');
16     ylabel([LABELS{i}, ' of EMG']);
17     title(['Correlation ', num2str(CORRS(i))]);
18 end
19
```

Figures:



Output

Code ran without output.

