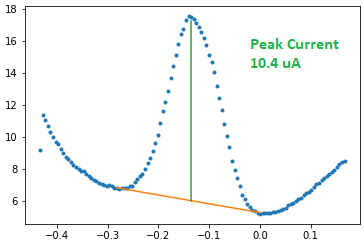
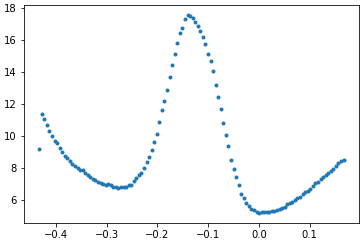
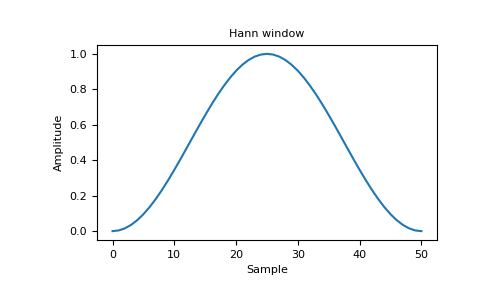
Step 1. Peak fitting for each potentiostat scan.







From the raw signal, we smooth the signal with hanning window convolve – the smoothing window is 11 datapoints.

Then we find the peak of the curve with scipy.signal.find\_peaks. Height and prominence thresholds are set to the 3 times the 80% quantile of the neighboring signal point difference. Width limit is 1/30 of the data length. The peak height evaluation point is 0.5 peak width.

If multiple peaks are found in the signal, we use a set of rules to determine which peak to pick.

1. In general, we favor the peaks that are higher, wider and more prominent.
2. We don’t want the peak that have the left intersection point too early (before 1/10th of the total points).
3. We calculate a score from these criteria.
4. We pick a peak that have the highest score.

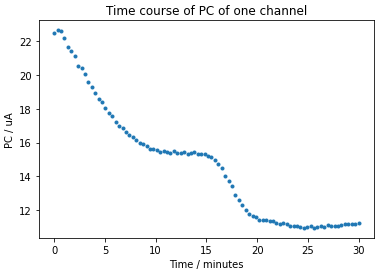
After the peak position is found, we need to determine the peak height. (The peak height is the green line in the figure above, it’s different from the peak height given by the scipy.signal.find\_peaks)

We do this by find the tangential line (the orange line in the figure above) first, then find the delta between the signal value and the y value on the tangential line at peak position.

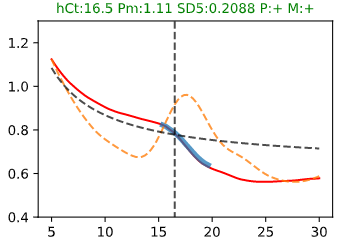
Then the peak height, or peak current is determined.

Step 2. Calculate the Ct (Cycle threshold), Pm (Prominence) and SD (Signal Drop) from the time – series data. The 3 values are then used to call whether a channel is positive or negative.

Because the test last for 30minutes, the data similar as shown above is collected every 20 seconds on each channel. For a given channel, after 30minutes, there will be a total of 90 data points. As shown below. At each time point, the peak current we got from the step1 is used for plotting.



From the Peak Current time – series data, we need to calculate Ct, Pm and SD. As shown in the figure below. The x-coordinate of the vertical dashed line is the Ct.



We first reject data outliers that are 2 stdev away, then smooth the curve with another hanning window convolve (window = 11). Then we normalize the data to the average data value between 5-10 minutes.

The data from 0 – 5 minutes is discarded. The following process only use 5-30minutes data.

The smooth-normalized data (red curve) is then used to calculate the first derivative, the first derivative is then flipped sign (the orange curve). Peak is found on the first derivative curve with scipy.signal.find\_peak. If multiple peaks are found, the most prominent peak is chosen.

The left and right half peak height intersection points (left and right IPS) of the found first derivative peak is determined (the blue overlay segment indicates the start and end of the two points).

The Pm (prominence) is the prominence of this peak. The SD (signal drop) is determined by the signal drop from left IPS to the data point 5 minutes after the left IPS.

We then use the data from 5minutes to the left IPS to perform a hyperbolic fit (the segment on the red curve from 5minutes to the start of the blue segment (around 14minutes)).

The hyperbolic function is a 3 parameter function: Y = P0 / (X+P1) + P2

The fit hyperbolic function is then shift down by 5% of the y value of the last data point used in fitting.

(This curve is the black dashed curve in the figure above)

The intersection point of the shifted hyperbolic fit with the red curve is determined. (The intersection point is indicated by the vertical black dashed line in the figure above)

After we determine the SD, Pm and Ct, we then call the data is positive or negative based on the certain prediction rule, based on the specific target gene and reaction we are considering. For example, if the data is from RP4 channel, we call the reaction positive if Ct<25minutes and SD > 0.05 and Pm>0.22.