Sudden change detect

We use the test function (piecewise constant function for sudden change) to transform the data space to coefficient space. In coefficient space, the behavior of the sudden change is the outlier. By transforming the data space to coefficient space, we transform the sudden change detect to the outlier detect.

Outlier detect

For the outlier detect, there are several standard methods. We introduce the Z-Score method (the mean and standard deviation) and the IQR (InterQuartile Range, use the quartile).

Z-Score Method

Z-score is a parametric outlier detection method in a one or low dimensional feature space. This technique assumes a Gaussian distribution of the data. The outliers are the data points that are in the tails of the distribution and therefore far from the mean. How far depends on a set threshold $z_{thr}$ for the normalized data points $z(i)$ calculated with the formula: $z(i) = \frac{x(i) - \mu}{\sigma}$, where $x(i)$ is a data point, $\mu$ is the mean of all $x(i)$ and $\sigma$ is the standard deviation of all $x(i)$. An outlier is then a normalized data point which has an absolute value greater than $z_{thr}$. That is: $|z(i)| > z_{thr}$ $\Rightarrow$ $z(i)$ $\text{Commonly used } z_{thr}$ values are $2.5, 3.0$ and $3.5$.

- The advantage of the Z-score method: the mean and standard deviation can be calculated online. Thus, they can be calculated very efficiently. $\bar{x}_n$ denotes the sample mean of the first $n$ samples ($x_1, ..., x_n$), $s_n^2$ denote the sample variance, then $\bar{x}_n = \frac{n}{n-1}\bar{x}_{\{n\}} + x_n/[n] = \bar{x}_{\{n-1\}} + s_n^2/[n] = \frac{n}{n-2}\bar{x}_{\{n-1\}} + s_{n-1}^2 + (x_n - \bar{x}_{\{n-1\}})^2/[n]$, $n \geq 2$.

- The disadvantage of the Z-score method: the mean and standard deviation is sensitive to the outlier. Thus, the result of the detect is not very accurate and sometimes strange.

IQR method
This is the simplest, nonparametric outlier detection method in a one dimensional feature space. Here outliers are calculated by means of the $IQR$ (InterQuartile Range).

A quantile is the element at a certain rank in the dataset after sort. The first $25\%$ and the third $75\%$ quartile $Q_{1}, Q_{3}$ are calculated. An outlier is then a data point $x_{i}$ that lies outside the interquartile range. That is: $x_{i} \gt Q_{3} + k(IQR)$ or $x_{i} \lt Q_{1} - k(IQR)$ where $IQR = Q_{3} - Q_{1}$ (and $k \geq 0$).

Using the interquartile multiplier value $k = 1.5$, the range limits are the typical upper and lower whiskers of a box plot.

- The advantage of IQR method is that: Compare with means and standard deviations, the quantile are less sensitive to outlier, so the result is more accurate.
- The disadvantage of the IQR method:
  - the computation of the exact quantile is expensive. It is difficult for the large data file or streaming data. However, there are several fast algorithms to calculate the approximate quantile.

I am reading this paper now and I think I need several days to finish it.
- the IQR method is only useful for one dimension.

create sudden change data

The length of the data is 100000, first change happens at 30000, second change happens at 70000. In this file, we use the IQR method to detect the outlier.

```python
In [2]:
data = mq.sudden_drift_data(initial=0.8, final=0.4,
                              changel = 30000, change2 = 70000, total_time=100000)
```

Singlscale online detect

We summarize the singlscale online detect algorithm as the following:

- $x_{1}, x_{2}, \ldots, x_{t}$ is the stream data set we detect; Given the scale parameter $m$; threshold parameter $\sigma$; $f$ is the test function define on $[0, 1]$; $Ws$ denote an empty stream data set; $A$ denote an empty coefficient set.
- For $k = 1, 2, \ldots$:
  - $W := W \cup \{t \}$; we use $s_{0}, s_{1}, \ldots, s_{2^{m-1}}$ to denote the new $2^{m}$ elements,
  - Calculate the coefficient for the new $2^{m}$ elements: $a = \sum_i (2^{m-1})f(t_i)s_i$ where $t_i = (i+0.5)/2^{m}$.
  - Add the new coefficient to the coefficient set $A := A \cup \{a\}$
  - Calculate the $Q_{1}, Q_{2}$, where $25\%, 75\%$ quantile number for the coefficient set let $IQR = Q_{3} - Q_{1}$.
  - if $a < Q_{1} - \sigma IQR$ or $a > Q_{3} + \sigma IQR$:
    - Conclude the change happens between the new $2^{m}$ elements
    - Set the new coefficient set $A = \{\}$

Computation for calculating the coefficients: If we assume the length of the stream data is $N$, then, for a fixed scale $m$, we need do $\frac{N}{2^{m}}$ times $2^{m}$ summation.
In [6]:

```python
reload(sq)
test = sq.contSent
threshold = 4
scale = 1
start = time.time()
a, lower, upper, anomalyInterval = sq.singlescale_online(data, test, threshold, scale)
end = time.time()
fig1, fig2 = sq.plot_amplitudes(data, a, lower, upper, scale)
print('{} time: {}'.format(end - start))
print('anomalyInterval: {}'.format(anomalyInterval))

ANOMALY: scale: 1, partition: [30000, 30C02], amplitude: 0.1623245239645973, lower: -0.04239554646770544, upper: 0.04230377653439058
ANOMALY: scale: 1, partition: [30000, 30C02], amplitude: -0.1978847288082387, lower: -0.04279938960694715, upper: 0.04282035052478644

time: 0.55216858175488
IQRTime : 18.069312572479248
addTime : 1.6530795097351074
appendTime : 2.0463735673095703
appendIQRTime : 2.1502236677978516
findchangeTime : 0.07546782493591309

time : 24.880094242095947
anomalyInterval : [{'level 1': [[30000, 30C02], [70000, 70C02]]
```

![Image of data and amplitude scale 1]
<table>
<thead>
<tr>
<th>scale</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
<th>9</th>
<th>10</th>
</tr>
</thead>
<tbody>
<tr>
<td>coefficient time</td>
<td>0.4666</td>
<td>0.2230</td>
<td>0.1097</td>
<td>0.0529</td>
<td>0.0259</td>
<td>0.0131</td>
<td>0.0088</td>
<td>0.0043</td>
<td>0.0027</td>
<td>0.0017</td>
</tr>
<tr>
<td>IQR time</td>
<td>15.8939</td>
<td>5.0263</td>
<td>2.5172</td>
<td>1.1220</td>
<td>0.5221</td>
<td>0.2508</td>
<td>0.1407</td>
<td>0.0627</td>
<td>0.0349</td>
<td>0.0195</td>
</tr>
<tr>
<td>total time</td>
<td>21.8146</td>
<td>8.4379</td>
<td>3.5943</td>
<td>1.6517</td>
<td>0.7810</td>
<td>0.3800</td>
<td>0.2210</td>
<td>0.1047</td>
<td>0.0583</td>
<td>0.0365</td>
</tr>
</tbody>
</table>

**single scale shift online detect**

We summarize the single scale shift online detect algorithm as following:

- $x_{-1}, x_{2}, \cdots, x_{t}, \cdots$ is the data set we detect; Given the scale parameter $m$; threshold parameter $\sigma$; $f$ is the test function define on $[0, 1]$; $W$ denote an empty stream data set; $A$ denote an empty coefficient set.
- $W := \cup \{\text{add } 2^m \text{ new elements}\}$
- For $k = 1, 2, \cdots$:
  - $W := W \cup \{\text{add } 2^m \text{ new elements}\}$; we use $s_{0, 1}, \cdots, s_{(2^m - 1)}$ to denote the new $2^m$ elements, and the $s_{(2^m - m)}, s_{(2^m + 1)}, \cdots, s_{(2^m - 1)}$ to denote the last $2^m$ elements in old $W$.
  - $C$ an empty set to store the coefficients for the new $2^m$ element.
  - For $j = 1, 2, \cdots, 2^m$:
    - Calculate the coefficient for the element $s_{(2^m + j)}, s_{(2^m + j + 1)}$, \ldots, $s_{(2^m - 1)}$.
    - Add the new coefficient to the coefficient set $A := A \cup \{C[idx] \}$.
    - Calculate the $Q_2$, where $25\%$, $75\%$ quantile number for the coefficient set.
    - Let $Q_2 = Q_2 - Q_1$.
- If $C[idx] < Q_2 - \sigma$ or $C[idx] > Q_2 + \sigma$:
  - Conclude the change happens between $s_{(2^m + idx)}$, $s_{(2^m + idx + 1)}$, \ldots
  - Set the coefficient set $A = \emptyset$ to be an empty set.

Computation for calculating coefficient: If we assume the length of the stream data if $N$, then, for a fixed scale $m$, we need do $N - 2^m + 1$ times $2^m$ summation.

```
In [31]:
reload(ssq)
test = ssq.conttest
threshold = 4
scale = 2
start = time.time()
da, lower, upper, anomalyIntegral = ssq.singlescale_shift_online(data, test, threshold, end = time.time())
fig1, fig2 = ssq.plot_amplitudes(data, a, lower, upper, scale)
print("time : ", format(end - start))
print("anomalyIntegral : ", format(anomalyIntegral))
```

ANOMALY: scale: 2, partition: [29999, 30003], amplitude: 0.2032994617906707, lower: -0.06027930605085042, upper: 0.06031726952340303 ANOMALY: scale: 2, partition: [69999, 70003], amplitude: -0.1988290157088446, lower: -0.05867812201385368, upper: 0.059804537105101765
ccoeffTime : 1.69778332173352395
IQRTIme : 6.093077182769775
addTime : 0.8020226955413818
appendTime : 0.7342238426268496
appendIQRTime : 0.6694283485412598
findchangeTime : 0.03143906593322754
time : 10,152550220489502
anomalyInterval : ('level 2': [[29999, 3C003], [69999, 70003]])

Here the coefficient time is the total time for calculating the coefficient, the IQR time is the total time for calculating the $Q_1$ and $Q_3$.

**singlescale online method**

<table>
<thead>
<tr>
<th>scale</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
<th>9</th>
<th>10</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>coefficient time</strong></td>
<td>2.0310</td>
<td>1.7261</td>
<td>1.5069</td>
<td>1.3741</td>
<td>1.3479</td>
<td>1.3285</td>
<td>1.5479</td>
<td>1.8042</td>
<td>1.8243</td>
<td>2.1602</td>
</tr>
<tr>
<td><strong>IQR time</strong></td>
<td>15.9050</td>
<td>6.1988</td>
<td>2.5479</td>
<td>1.1418</td>
<td>0.5348</td>
<td>0.2548</td>
<td>0.1302</td>
<td>0.0710</td>
<td>0.0329</td>
<td>0.0166</td>
</tr>
<tr>
<td><strong>total time</strong></td>
<td>23.5356</td>
<td>10.2852</td>
<td>5.1084</td>
<td>3.0205</td>
<td>2.1326</td>
<td>1.7079</td>
<td>1.7461</td>
<td>1.9115</td>
<td>1.8756</td>
<td>2.1867</td>
</tr>
<tr>
<td>scale</td>
<td>1</td>
<td>2</td>
<td>3</td>
<td>4</td>
<td>5</td>
<td>6</td>
<td>7</td>
<td>8</td>
<td>9</td>
<td>10</td>
</tr>
<tr>
<td>-------</td>
<td>-------</td>
<td>-------</td>
<td>-------</td>
<td>-------</td>
<td>-------</td>
<td>-------</td>
<td>-------</td>
<td>-------</td>
<td>-------</td>
<td>-------</td>
</tr>
<tr>
<td>total time</td>
<td>21.8146</td>
<td>9.4379</td>
<td>3.5943</td>
<td>1.6517</td>
<td>0.7810</td>
<td>0.3800</td>
<td>0.2210</td>
<td>0.1047</td>
<td>0.0583</td>
<td>0.0365</td>
</tr>
</tbody>
</table>

**singlescale shift online method**

<table>
<thead>
<tr>
<th>scale</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
<th>9</th>
<th>10</th>
</tr>
</thead>
<tbody>
<tr>
<td>coefficient time</td>
<td>2.0310</td>
<td>1.7261</td>
<td>1.5069</td>
<td>1.3741</td>
<td>1.3479</td>
<td>1.3285</td>
<td>1.5479</td>
<td>1.8042</td>
<td>1.8243</td>
<td>2.1602</td>
</tr>
<tr>
<td>IQR time</td>
<td>15.9050</td>
<td>5.1988</td>
<td>2.5479</td>
<td>1.1418</td>
<td>0.5348</td>
<td>0.2548</td>
<td>0.1302</td>
<td>0.0710</td>
<td>0.0329</td>
<td>0.0166</td>
</tr>
<tr>
<td>total time</td>
<td>23.5356</td>
<td>10.2852</td>
<td>5.1084</td>
<td>3.0205</td>
<td>2.1326</td>
<td>1.7079</td>
<td>1.7461</td>
<td>1.9115</td>
<td>1.8756</td>
<td>2.1867</td>
</tr>
</tbody>
</table>

- From the table, we can observe the most time calculate is spend on calculate the IQR. Thus, using a fast algorithm to calculate the IQR for the streaming data is necessary. This is what I am doing now.
- Compare the coefficient time for both method: singlescale shift online method is approximate $2^m$ times for the singlescale online method.
- For the singlescale online method, the IQR time spends the most times for all scales. However, for the singlescale shift online method, the coefficient time spends the most time in for the large scale. Thus, if we use fast algorithm for calculate IQR time, the singlescale online method will improve significantly; the singlescale shift online method for the large scale just improve a little bit.
- The phenomenon I am not clear is that: the coefficient time for singlescale shift online method is decrease from scale 1 to scale 6 and then increse. If we assume the length of the stream data if $N$, then, for a fixed scale $m$, we need do $N - 2^m + 1$ times $2^m$ summation. So if $m$ increase, I think the coefficient time will also increase. I do not how to explain this phenomenon at this moment.