

1 Learning rate

The learning rate w governs the strength of the vector updates. Considering the base x of a vector from a short read aligned to a base y in the reference sequence, at iteration i , updates are performed using the following equation

$$y_{i+1} = w * y_i + (1 - w) * x$$

1.1 Selecting learning rate

The average coverage is calculated to choose the value of the learning rate as it will determine the expected number of iteration. Considering a value of $y_1 = 0$, we want the rate strong enough to bring it to 1 if the reads values $x = 1$ so that substitution are fully defined at the end of the learning. Using the above equation in this special case,

$$y_2 = w * y_1 + (1 - w) * x$$

$$y_2 = (1 - w)$$

At the next iterations

$$y_3 = w * y_2 + (1 - w) * x$$

$$y_3 = w * (1 - w) + (1 - w) = 1 - w^2$$

$$y_4 = w * (1 - w^2) + (1 - w) = 1 - w^3$$

Therefore, it can be seen that

$$y_i = 1 - w^{i-1}$$

when $y_1 = 0$ and $x = 1$.

If we want $y = 0.99$, y cannot reach 1 exactly, at iteration $i = 100$

$$y_{100} = 0.99$$

$$1 - w^{99} = 0.99$$

$$w = \sqrt[0.99]{0.01}$$

Similarly, we want the weight to be strong enough to reach 0 when starting at 1.

$$y_i = w^{i-1}$$

when $y_1 = 1$ and $x = 0$. We want

$$w^{99} = 0.01$$

$$w = \sqrt[99]{0.01}$$

In other words, to find the weight value, one can use the following equation

$$w = \sqrt[coverage-1]{precision}$$

wherer the *precision* is the required approximation to 0 and *coverage* is the expected mean coverage.

In practice, we use half the average coverage to calculate the learning rate so that variation in the coverage across the sequeunce is taken into account. To set the threshold for INDELS, half of the learning rate is used.

1.2 Issue

The order with which the reads are chosen will have an impact on the update of the vector values. For example, when $y_1 = 0$, half of the reads are $x = 0$ and half $x = 1$, the final value y_{10} is

```
>> y=0;for n=[0 0 0 0 0 1 1 1 1 1],y=y*0.9+n*0.1;end;disp(y);  
0.4095  
>> y=0;for n=[1 1 1 1 1 0 0 0 0 0],y=y*0.9+n*0.1;end;disp(y);  
0.2418  
>> y=0;for n=[1 0 1 0 1 0 1 0 1 0],y=y*0.9+n*0.1;end;disp(y);  
0.3085
```

Therefore, alternative calculation for the update, e.g. weighted iterative mean, could be utilised.