

```

1  function graph_matrix = generate_ancestry_graph(dataset, N,
graph_matrix, Q, W, E)
2
3  %copyright Charles Davi 2022
4  %BLACK TREE AUTOML
5  %HUMANITY'S FASTEST DEEP LEARNING SOFTWARE
6
7  %=====
8  %=====
9  %VECTORIZED COMPUTATIONAL GENOMICS - mtDNA Dataset Command Line
10 %=====
11 %=====
12
13 %=====
14 %INPUT VARIABLES
15 %=====
16
17 best_match = 0; %default value for maximum matching percentage
18
19 temp_vector = [Q W E];
20
21 %uses simple modulo to iterate over all combinations
22 for K = 0 : 2
23
24     %cycles through each combination using modular arithmetic
25     tempA = mod(K,3) + 1;
26     tempB = mod(K + 1,3) + 1;
27     tempC = mod(K + 2,3) + 1;
28
29     A = temp_vector(tempA);
30     B = temp_vector(tempB);
31     C = temp_vector(tempC);
32
33     x = find(dataset(:,N+1) == A); %finds all rows in root class
34     y = find(dataset(:,N+1) == B); %finds all rows in comparison
class 1
35     z = find(dataset(:,N+1) == C); %finds all rows in comparison
class 2
36
37     %the number of genomes per class
38     a = size(x,1);

```

```

39     b = size(y,1);
40     c = size(z,1);
41
42     match_count = 0; %the number of successful ancestry tests
43
44     counter = 0;
45
46     for i = 1 : a
47
48         root = dataset(x(i),:);
49
50         for j = 1 : b
51
52             top = dataset(y(j),:);
53
54             for k = 1 : c
55
56                 bot = dataset(z(k),:);
57
58                 L1 = sum(root(1:N) == top(1:N)); %root match to top
59                 L2 = sum(root(1:N) == bot(1:N)); %root match to bot
60                 R = sum(top(1:N) == bot(1:N)); %top match to bot
61
62                 counter = counter + 1;
63
64                 %if true, the root is the ancestor of the top and bot
65                 if(L1 > R && L2 > R)
66
67                     match_count = match_count + 1;
68
69                 endif
70
71             endfor
72
73         endfor
74
75     endfor
76
77     graph_matrix(A,B) = graph_matrix(A,B) + match_count/counter;
78     graph_matrix(A,C) = graph_matrix(A,C) + match_count/counter;
79
80 endfor
81

```