Supplementary Materials

1. The calculation process of the operators

For the Addition and Subtraction operators, the fore of (-1)(+1) or (+1)(-1) is a carry shown in Table 1. The carry will join the calculation of its fore digit pair. Note that the carry of the first digit pair $\{C_L[1], C_R[1]\}$ would be ignored; otherwise, the string may overflow.

In the example shown in Figure 3, $O[3]$ accepts the carry (-1) produced by $\{C_L[4], C_R[4]\}$, so that $O[3]= C_L[3]+C_R[3] +(-1)=-1$. In this way, b_1 generates a new ternary string [-1, -1, -1, +1, +1], containing a column (last three digits [-1, +1, +1]) in the feature subspace with index [-1, -1].

The calculation process of other ternary arithmetical operators works in the similar way.

2. The analysis of the outputs using various ternary operators

Table S1 investigates the results of diverse operators with the same input digits $C_L[i]=C_R[i]$. Here T and F are the judgments whether the output of an operator equals the input or not. It is interesting to find that for an input digit pair {0, 0}, all operators would output 0. While for the other two input pairs {+1, +1} and {-1, -1}, about 3/4 operators produce the same as inputs. In short, in the case that the two input digits are the same, various operators tend to produce the same digit.

For the 6 cases of $C_L[i]\neq C_R[i]$ ({+1, -1}, {-1, +1}, {0, +1}, {+1, 0}, {0, -1}, {-1, 0}), different operators perform in various ways. `Table S2 summarizes different results that operators preserve the left or right input digit. It is found that **AD**, **SU** and **R** would give 2/6, 2/6, and 6/6 chance to output a third digit other than the inputs. While **MU**, **LA** and **LO** operators give 50% probability to allow one of the inputs to survive after calculation.

As there are 6 cases to receive two different input digits, for eight operators, there are 6×8=48 cases in all. If each operator is selected at an equal chance, then in the case of $C_L[i] \neq C_R[i]$, the outputs are of 10/48 probability to produce a completely different digit. In detail, both **AD** and **SU** contribute 2 chances, and **R** produces 6 chances.

As for the 3 cases of receiving $C_L[i]=C_R[i]$, 3×8=24 cases should be taken into consideration. As both **AD** and **SU** produce 2 chances to generate the third digit, and **MU** maintains 1 case, the results are of 5/24 probability to be different from inputs.

Overall, our operators generate columns with a high probability to include a part of information inheriting from inputs, and a small proportion of digits completely different from inputs. In this way, operators can guide our GP to evolve stably.

Table S1: The probability of digits to be preserved after calculation

Operator	$\{C_L[i], C_R[i]\}$		
	{+1,+1}	{-1,-1}	{0,0}
AD	F	F	T
SU	F	F	T
MU	T	F	T
LA	T	T	T
LO	T	T	T
R	T	T	T
O-E	T	T	T
H-H	T	T	T

`Table S2: The probability of symbols in outputs

Operators	$C_L[i] \neq C_R[i]$(6 cases)	
	$O[i]=C_L[i]$	$O[i]=C_R[i]$
AD	2	2
SU	2	2
MU	3	3
LA	3	3
LO	3	3
R	0	0
O-E	3	3
H-H	3	3

`Table S3: The appearance times of different digits in outputs

Operators	**Probability**		
	+1	-1	0
AD	3	3	3
SU	3	3	3
MU	2	2	5
LA	1	5	3
LO	5	1	3
R	3	3	3

In the ECOC ternary coding system, classes relabeled as 0 are ignored in the training process. If 0 digit takes part in a column at a high probability, the associated dichotomizer may face the undertraining problem, and its classification accuracy is unreliable because it has no information about the ignored classes. So it is important to limit the probability of 0 in outputs to avoid the generate of a highly sparse codematrix.

In general, there are 9 different pairs to serve as inputs. Since each ternary number is randomly picked to form a column in the column pool initialization, each pair are picked up as inputs at the same probability. So the probability of diverse operators to output a digit simply depends on its working principle, as summarized in `Table S3. As the **O-E** and **H-H** operators keep half of the original inputs from $C_L$ and $C_R$ respectively, their outputs are connected with the input sequences and do not affect the output distribution directly. Therefore, they are ignored in `Table S3.

It is found that **AD**, **SU** and **R** operators produce each digit with an equal chance. Logic operators **LA** and **LO** tend to focus on -1 and +1 respectively, but by treating them as a group, the probability of each digit appears in their outputs is equal. While the **MU** operator prefers 0 by 2/9 higher probability than the other two digits. For 6 operators and 9 inputs, there are $6 \times 9 = 54$ possible combinations. As 0 appears in 5 cases of the outputs of MU operator and 15 cases of other 5 operators, the total appearance probability of 0 is 20/54 in outputs, and 17/54 for both +1 and -1. Because 0 appears at a lower rate than 0.5, the newly produced columns are denser than the inputs. In another word, a parent node tends to contain less 0 digits than its child nodes, avoiding the generation of a sparse codematrix.

In short, the difference among operators offers our GP higher search ability. The contributions of different operators are discussed in Section 4.7.