Neural Network I: Fundamental Theory and Applications

Project V

Project member:

① Atsushi Takamiya (s1260013)
② Tomonori Suzuki (s1260032)
③ Taiki Watanabe (m5251152)
④ Kaito Suzuki (m5251144)
⑤ Kaito Iwamatsu (m5251115)

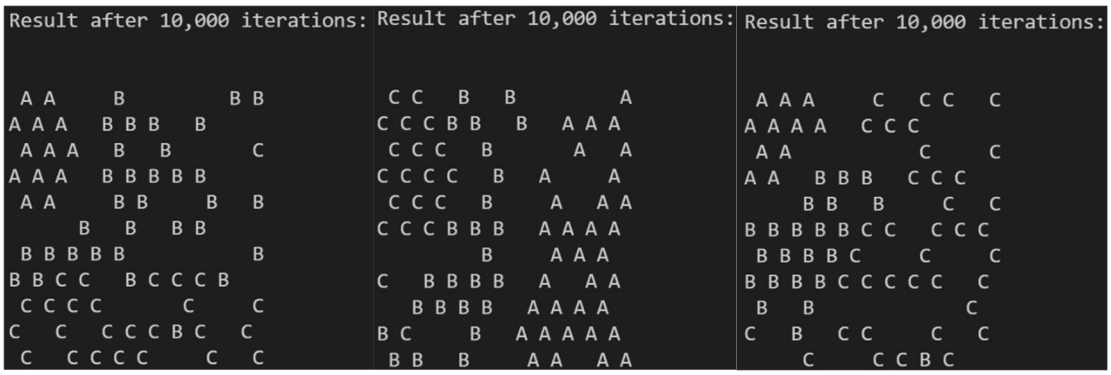
• Project V:

a,b,c)

a) Modify the program, and apply our program to IRIS dataset.

b) SOFM (self-organizing feature map)

c)



Appropriate parameters are changed depending on the number of data, the amount of information, and the number of classes to be classified.

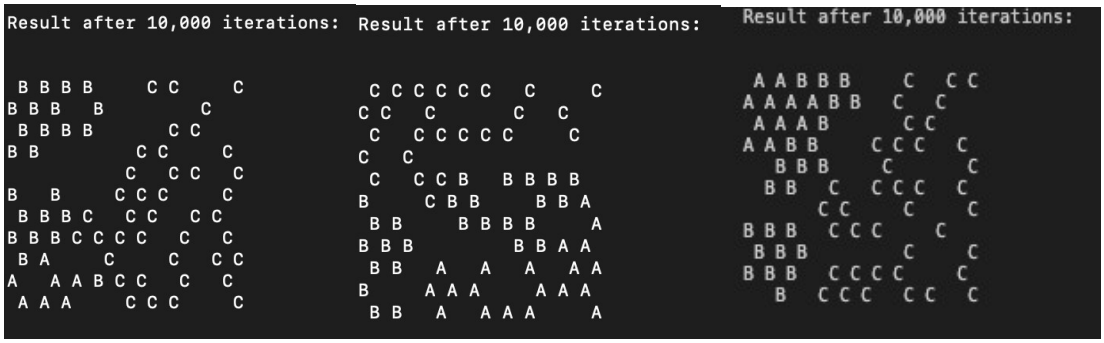
Also, in this time, since similar data is included between the two classes, if the range for updating the weights is too wide, it may not be possible to create a beautiful feature map.

d,e,f)

d) How can the feature map be more beautiful?

e) Feature scaling

f)



Class A, B, C were classified without mixing by feature scaling.

Project Program (Major changes)

a,b,c)

Change input data to IRIS datasets.

Change arguments of SOFM function

```
SOFM(2000, 0.3, 0.1, 3, 2);
Calibration();
printf("¥n¥nResult after the first 2,000 iterations:¥n");
PrintResult();

SOFM(8000, 0.1, 0.01, 1, 1);
Calibration();
printf("¥n¥nResult after 10,000 iterations:¥n");
PrintResult();
```

d,e,f)

Standardize IRIS datasets.

```
double sum[I];
double ave[I];
double b[I];

for(i=0; i<I; i++){
 sum[i]=0;
 ave[i]=0;
 for(int j=0; j<P; j++){
  sum[i]+=x[j][i];
 }
}

for(i=0; i<I; i++){
 ave[i]=sum[i]/P;
}

for(i=0; i<I; i++){
 b[i]=0;
 for(int j=0; j<P; j++){
  b[i]+=pow(x[j][i]-ave[i],2.0);
  //b[i]+=x[j][i]-ave[i];
 }
 b[i] = b[i]/=P;
}

for(i=0; i<I; i++){
 for(int j=0; j<P; j++){
  x[j][i] = (x[j][i] - ave[i]) / b[i];
 }
}
```