SVM

In[ ]:=
```
ClearAll[X]; ClearAll[Y]; ClearAll[BigMatrix];
```

In[21]:=
```
X = {{1.0, 1}, {2, 1}, {2, 2}, {4, 1}, {4, 4}};
```

In[22]:=
```
Y = {{1}, {1}, {-1}, {-1}, {-1}};
```

Calculus

In[23]:=
```
BigMatrix = Table[Y[[i]][[1]] * Y[[j]][[1]] * (X[[i]].X[[j]]), {i, 1, Length[X]}, {j, 1, Length[X]}];
```

In[24]:=
```
BigMatrix // TableForm
```

Out[24]//TableForm=

2.	3.	-4.	-5.	-8.
3.	5	-6	-9	-12
-4.	-6	8	10	16
-5.	-9	10	17	20
-8.	-12	16	20	32

In[25]:=
```
ClearAll[alphas];
ClearAll[W];
ClearAll[TmpW];
ClearAll[rules];
ClearAll[W0];
ClearAll[svPos]
```

In[26]:=
```
alphas = Table[Symbol["alpha" ~~ ToString[i]], {i, 1, Length[X]}];
```

In[27]:=
```
rules = Minimize[{1 / 2 * alphas.BigMatrix.alphas +
      Table[-1, {i, 1, Length[X]}].alphas, Transpose[Y].alphas == 0 &&
      (Table[alph ≥ 0, {alph, alphas}] /. (List → And))}, alphas][[2]]
```

Out[27]=
$$\{\text{alpha1} \to 2.62365 \times 10^{-9},\ \text{alpha2} \to 2.5,\ \text{alpha3} \to 2.,\ \text{alpha4} \to 0.5,\ \text{alpha5} \to 4.36441 \times 10^{-10}\}$$

In[28]:=
```
TmpW = Sum[(alphas[[n]] /. rules) * Y[[n]][[1]] * X[[n]], {n, 1, Length[X]}]
```

Out[28]=
```
{-1., -2.}
```

In[29]:=
```
svPos = FirstPosition[rules, alpha_Rule /; alpha[[2]] > 0.000001][[1]];
```

In[30]:=
```
W0 = 1 / Y[[svPos]][[1]] - Sum[TmpW[[i]] * X[[svPos]][[i]], {i, 1, Length[X[[svPos]]]}]
```

Out[30]=
```
5.
```

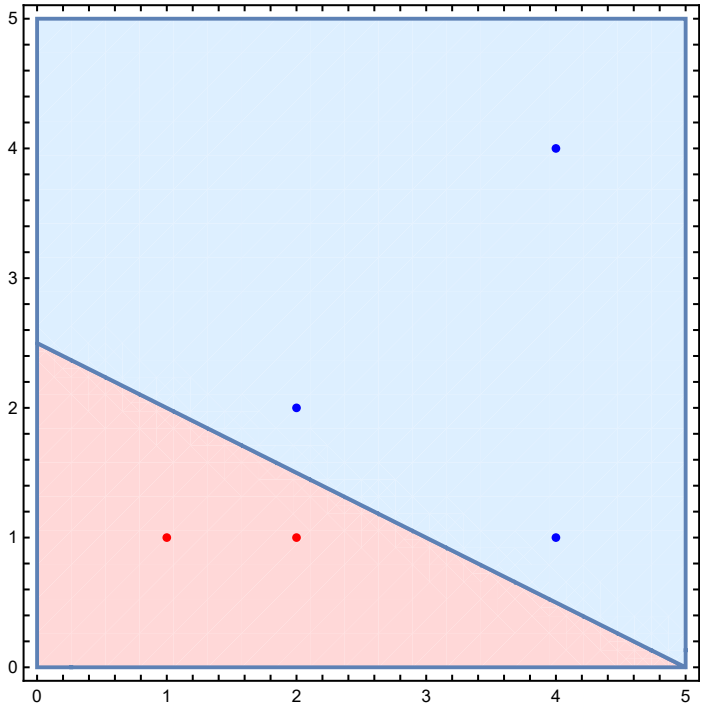
In[31]:=
```
W = Prepend[TmpW, W0]
```

Out[31]=
```
{5., -1., -2.}
```

Result

In[32]:=
```
Show[RegionPlot[Sign[W.{1, x, y}] > 0, {x, 0, 5}, {y, 0, 5}, PlotStyle → LightRed],
 RegionPlot[Sign[W.{1, x, y}] ≤ 0, {x, 0, 5}, {y, 0, 5}, PlotStyle → LightBlue],
 ListPlot[Select[X, Y[[FirstPosition[X, #][[1]]]][[1]] == 1 &], PlotStyle → Red],
 ListPlot[Select[X, Y[[FirstPosition[X, #][[1]]]][[1]] == -1 &], PlotStyle → Blue]]
```

Out[32]=



Kernel Trick

In[ ]:=
```
ClearAll[X]; ClearAll[Y]; ClearAll[BigMatrix]; ClearAll[radialKernel]
```

In[ ]:=
```
X = {{1, 1}, {2, 1}, {2, 2}, {2, 0.5}, {4, 4}, {0, 0}};
```

In[ ]:=
$$Y = \begin{pmatrix} 1 \\ 1 \\ 1 \\ -1 \\ -1 \\ -1 \end{pmatrix};$$

Calculus

```
In[ ]:= radialKernel[xn_, xm_] := Exp[-xn.xn] * Exp[-xm.xm] * Exp[2 xn.xm] // N

In[ ]:= BigMatrix = Table[Y[[i]][[1]] * Y[[j]][[1]] * radialKernel[X[[i]], X[[j]]],
          {i, 1, Length[X]}, {j, 1, Length[X]}];

In[ ]:= BigMatrix // TableForm
```

Out[ ]//TableForm=

1.	0.367879	0.135335	−0.286505	-1.523×10^{-8}	−0.13
0.367879	1.	0.367879	−0.778801	-2.26033×10^{-6}	−0.00
0.135335	0.367879	1.	−0.105399	−0.000335463	−0.00
−0.286505	−0.778801	−0.105399	1.	8.76425×10^{-8}	0.014
-1.523×10^{-8}	-2.26033×10^{-6}	−0.000335463	8.76425×10^{-8}	1.	1.266
−0.135335	−0.00673795	−0.000335463	0.0142642	1.26642×10^{-14}	1.

```
In[ ]:= ClearAll[alphas]; ClearAll[rules]; ClearAll[W0]; ClearAll[svPos]; ClearAll[model];

In[ ]:= alphas = Table[Symbol["alpha" ~~ ToString[i]], {i, 1, Length[X]}];

In[ ]:= rules = Minimize[{1 / 2 * alphas.BigMatrix.alphas +
              Table[-1, {i, 1, Length[X]}].alphas, Transpose[Y].alphas == 0 &&
              (Table[alph ≥ 0, {alph, alphas}] /. (List → And))}, alphas][[2]]
```

Out[ ]=

```
{alpha1 → 1.14086, alpha2 → 4.21583, alpha3 → 0.149222,
 alpha4 → 4.20481, alpha5 → 0.589142, alpha6 → 0.711958}
```

```
In[ ]:= svPos = FirstPosition[rules, alpha_Rule /; alpha[[2]] > 0.000001][[1]];

In[ ]:= W0 = 1 / Y[[svPos]][[1]] -
            Sum[(alphas[[k]] /. rules) * Y[[k]][[1]] * radialKernel[X[[k]], X[[svPos]]], {k, 1, Length[X]}]
```

Out[ ]=

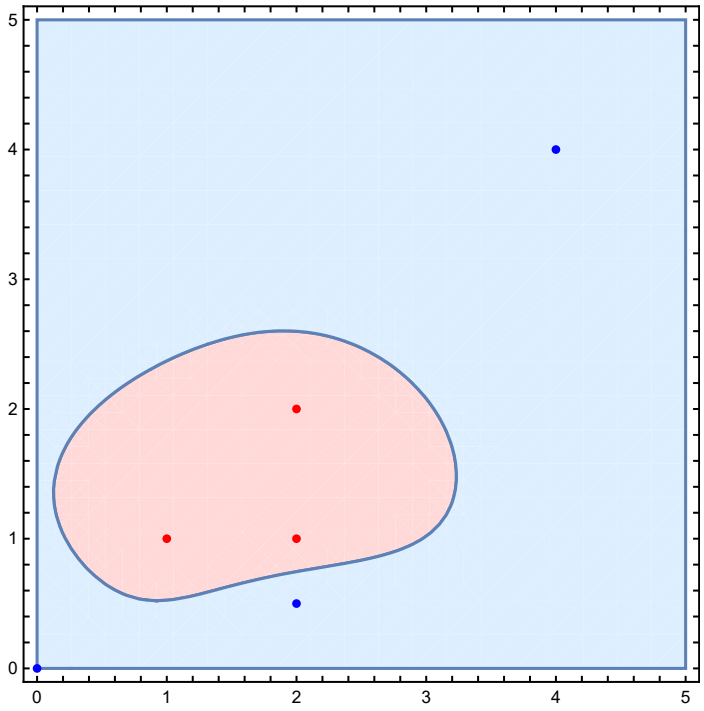
```
-0.410917
```

```
In[ ]:= model[input_] :=
          Sum[(alphas[[k]] /. rules) * Y[[k]][[1]] * radialKernel[X[[k]], input], {k, 1, Length[X]}] + W0
```

Result

```
In[•]:= Show[RegionPlot[model[{x, y}] > 0, {x, 0, 5}, {y, 0, 5}, PlotStyle → LightRed],
 RegionPlot[model[{x, y}] < 0, {x, 0, 5}, {y, 0, 5}, PlotStyle → LightBlue],
 ListPlot[Select[X, Y[[FirstPosition[X, #][[1]]]][[1]] == 1 &], PlotStyle → Red],
 ListPlot[Select[X, Y[[FirstPosition[X, #][[1]]]][[1]] == -1 &], PlotStyle → Blue]]
```

Out[•]=



Kernel Trick With Lots of samples

```
In[•]:= ClearAll[X]; ClearAll[Y]; ClearAll[BigMatrix]; ClearAll[radialKernel]
```

```
In[•]:= X = Table[RandomReal[{0.5, 4.5}], {i, 1, 20}, {j, 1, 2}];
```

```
In[•]:= Y = Table[RandomChoice[{-1, 1}], {i, 1, 20}, {j, 1, 1}];
```

Calculus

```
In[•]:= radialKernel[xn_, xm_] := Exp[-xn.xn] * Exp[-xm.xm] * Exp[2 xn.xm] // N
```

```
In[•]:= BigMatrix = Table[Y[[i]][[1]] * Y[[j]][[1]] * radialKernel[X[[i]], X[[j]]],
    {i, 1, Length[X]}, {j, 1, Length[X]}];
```

In[ ]:=
```
BigMatrix // TableForm
```

Out[ ]//TableForm=

1.	−0.129367	0.250647	−0.00197724	−0.109743	0.94
−0.129367	1.	−0.0175152	0.0624299	0.00597073	−0.1
0.250647	−0.0175152	1.	−0.00858707	−0.907575	0.13
−0.00197724	0.0624299	−0.00858707	1.	0.00695176	−0.0
−0.109743	0.00597073	−0.907575	0.00695176	1.	−0.0
0.946505	−0.155876	0.137931	−0.00129047	−0.0522123	1.
-6.55965×10^{-7}	0.00157998	-1.36675×10^{-6}	0.100468	8.30922×10^{-7}	−5.0
−0.819643	0.0988668	−0.0741276	0.000357619	0.0249782	−0.9
0.00875344	−0.498264	0.000304541	−0.0144171	−0.000069892	0.01
-6.69307×10^{-6}	0.0117004	-4.7169×10^{-8}	0.000248523	6.63883×10^{-9}	−0.0
−0.00490606	0.219066	−0.0060428	0.750164	0.00350508	−0.0
0.000491543	−0.0519502	1.5458×10^{-6}	−0.000077429	-1.88006×10^{-7}	0.00
0.0453244	−0.28785	0.0830941	−0.581458	−0.0557882	0.03
−0.000363103	0.0811418	-2.53972×10^{-6}	0.000548864	3.73489×10^{-7}	−0.0
−0.00247816	0.0023395	-2.01291×10^{-6}	4.01885×10^{-8}	1.91675×10^{-7}	−0.0
9.74367×10^{-7}	−0.00172175	2.71074×10^{-6}	−0.137385	-1.78815×10^{-6}	7.82
0.00187849	−0.126998	0.0000109149	−0.000344341	-1.57132×10^{-6}	0.00
-5.85153×10^{-6}	0.0133575	-8.38814×10^{-8}	0.000873046	1.40997×10^{-8}	−0.0
−0.720194	0.0501544	−0.0581055	0.000114539	0.0192442	−0.8
-1.30502×10^{-6}	0.00228418	-2.03428×10^{-9}	6.09065×10^{-6}	1.93863×10^{-10}	−4.3

In[ ]:=
```
ClearAll[alphas]; ClearAll[rules]; ClearAll[W0]; ClearAll[svPos]; ClearAll[model];
```

In[ ]:=
```
alphas = Table[Symbol["alpha" ~~ ToString[i]], {i, 1, Length[X]}];
```

In[ ]:=
```
rules = Minimize[{1 / 2 * alphas.BigMatrix.alphas +
      Table[-1, {i, 1, Length[X]}].alphas, Transpose[Y].alphas == 0 &&
      (Table[alph ≥ 0, {alph, alphas}] /. (List → And))}, alphas][[2]]
```

Out[ ]=
$$\{\text{alpha1}\to 1.12858\times10^{-9},\ \text{alpha2}\to 5.55214,\ \text{alpha3}\to 6.78379,\ \text{alpha4}\to 4.35518,$$
$$\text{alpha5}\to 7.22967,\ \text{alpha6}\to 16.2192,\ \text{alpha7}\to 47.5887,\ \text{alpha8}\to 15.5059,\ \text{alpha9}\to 6.92404,$$
$$\text{alpha10}\to 3.1236\times10^{-9},\ \text{alpha11}\to 1.19368,\ \text{alpha12}\to 6.85907,\ \text{alpha13}\to 5.46551,$$
$$\text{alpha14}\to 9.22159,\ \text{alpha15}\to 0.418223,\ \text{alpha16}\to 48.8134,\ \text{alpha17}\to 5.38197\times10^{-9},$$
$$\text{alpha18}\to 1.50079\times10^{-8},\ \text{alpha19}\to 7.5261\times10^{-9},\ \text{alpha20}\to 8.4002\times10^{-9}\}$$

In[ ]:=
```
svPos = FirstPosition[rules, alpha_Rule /; alpha[[2]] > 0.000001][[1]];
```

In[ ]:=
```
W0 = 1 / Y[[svPos]][[1]] -
  Sum[(alphas[[k]] /. rules) * Y[[k]][[1]] * radialKernel[X[[k]], X[[svPos]]], {k, 1, Length[X]}]
```

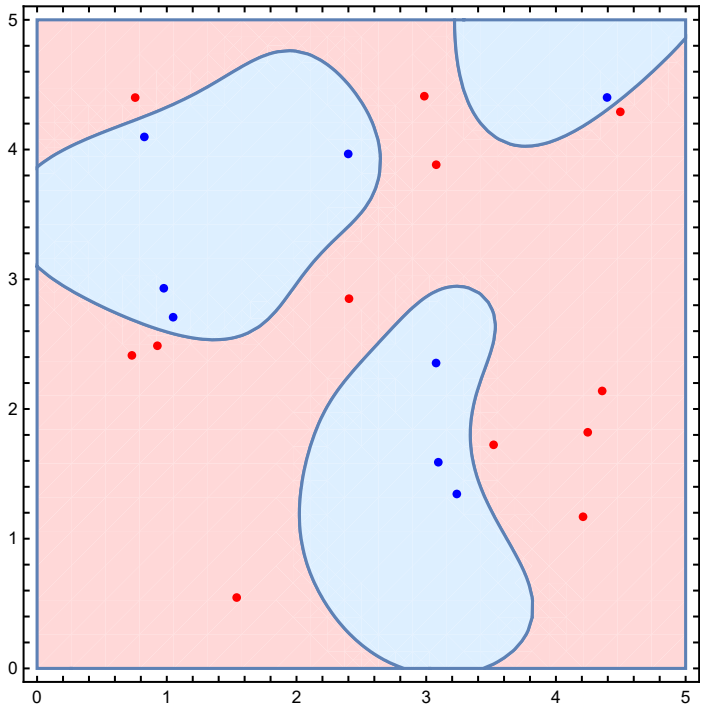
Out[ ]=
```
0.624492
```

In[ ]:=
```
model[input_] :=
 Sum[(alphas[[k]] /. rules) * Y[[k]][[1]] * radialKernel[X[[k]], input], {k, 1, Length[X]}] + W0
```

Result

In[•]:=
```
Show[RegionPlot[model[{x, y}] > 0, {x, 0, 5}, {y, 0, 5}, PlotStyle → LightRed],
 RegionPlot[model[{x, y}] < 0, {x, 0, 5}, {y, 0, 5}, PlotStyle → LightBlue],
 ListPlot[Select[X, Y[[FirstPosition[X, #][[1]]]][[1]] == 1 &], PlotStyle → Red],
 ListPlot[Select[X, Y[[FirstPosition[X, #][[1]]]][[1]] == -1 &], PlotStyle → Blue]]
```

Out[•]=



SVM With Soft Margin

```
ClearAll[X]; ClearAll[Y]; ClearAll[BigMatrix];
```

$$X = \begin{pmatrix} 2 & 1 \\ 1.0 & 3 \\ 2 & 0 \\ 4 & 1 \\ 4 & 4 \\ 4 & 3 \\ 1 & 2 \end{pmatrix};$$

$$Y = \begin{pmatrix} 1 \\ 1 \\ 1 \\ -1 \\ -1 \\ -1 \\ -1 \end{pmatrix};$$

Calculus

Result

```
Show[RegionPlot[Sign[W.{1, x, y}] > 0, {x, 0, 5}, {y, 0, 5}, PlotStyle → LightRed],
 RegionPlot[Sign[W.{1, x, y}] ≤ 0, {x, 0, 5}, {y, 0, 5}, PlotStyle → LightBlue],
 ListPlot[Select[X, Y[[FirstPosition[X, #][[1]]]][[1]] == 1 &], PlotStyle → Red],
 ListPlot[Select[X, Y[[FirstPosition[X, #][[1]]]][[1]] == -1 &], PlotStyle → Blue]]
```