# 536

Xinyang Wang

189001002

# 1 HW5

## 1.1

1. let's say the radius equals r,and the point is (a,b).We set sign function equal to the following which can separate the data point linearly:

$$\begin{aligned} f(x_1, x_2) &= sign(r^2 - (x_1 - a)^2 - (x_2 - b)^2) \\ &= sign((r^2 - a^2 - b^2) + 2ax_1 + 2bx_2 - x_1^2 - x_2^2) \end{aligned}$$

So under the feature map $(1, x_1, x_2, x_2x_2, x_1^2, x_1^2)$, there is a linear separator.

2. let's say the separator is the following sign function with center at (a,b), major axis = m; minor axis = n;

$$f(x_1, x_2) = sign(1 - \frac{(x_1 - a)^2}{m^2} - \frac{(x_2 - b)^2}{n^2})$$

We can rewrite the above form as below:

$$\begin{aligned} f(x_1, x_2) &= sign(m^2n^2 - n^2(x_1 - a)^2 - m^2(x_2 - b)^2) \\ &= sign((m^2n^2 - a^2n^2 - b^2m^2) + 2an^2x_1 + 2bm^2x_2 - n^2x_1^2 - m^2x_2^2) \end{aligned}$$

So we can see if an ellipsoidal separator exists, it can be found under the quadratic feature map.

## 1.2

Giving the kernel $K(\underline{x}, \underline{y}) = (1 + (\underline{x} * \underline{y}))^4$, we have the feature map of $(1, x_1, x_2, ..., x_1^4, x_2^4)$.

Consider we have two ellipse e1 and e2:
$m^2n^2 - n^2(x_1 - a)^2 - m^2(x_2 - b)^2 = 0$ and
$k^2l^2 - k^2(x_1 - c)^2 - l^2(x_2 - d)^2 = 0$

Then we set the sign function equaling to the following:

$$\begin{aligned} f(x_1, x_2) &= sign(-(m^2n^2 - n^2(x_1 - a)^2 - m^2(x_2 - b)^2) * (k^2l^2 - k^2(x_1 - c)^2 - l^2(x_2 - d)^2)) \\ &= () * 1 + () * x_1 + () * x_2 + ... + () * x_1^4 + () * x_2^4 \\ &* () \quad represents \quad the \quad corresponding \quad coefficient. \end{aligned}$$

As we can see, when data is in e1 or e2, $f(x_1, x_2) = 1$. And when data is out of e1 and e2,$f(x_1, x_2) = -1$. And we can also observe that the sign function is based on the feature mentioned above

### 1.3

Please pay attention here: As circle is a special form of ellipse, so the in the following part we use ellipse instead of circle.

As described in question 1.2, Giving the kernel $K(\underline{x}, \underline{y}) = (1+(\underline{x}*\underline{y}))^4$, we have the feature map of $(1, x_1, x_2, ..., x_1^4, x_2^4)$.

Then consider we have two ellipse e1 and e2 like what we have in question 1.2:
$m^2n^2 - n^2(x_1 - a)^2 - m^2(x_2 - b)^2 = 0$ and
$k^2l^2 - k^2(x_1 - c)^2 - l^2(x_2 - d)^2 = 0$
Assume e1 is surrounded by e2

Then we set the sign function equaling to the following (here is what makes difference from question 1.2: we do not have minus in the front of sign function):

$$
\begin{aligned}
f(x_1, x_2) &= sign((m^2n^2 - n^2(x_1 - a)^2 - m^2(x_2 - b)^2) * (k^2l^2 - k^2(x_1 - c)^2 - l^2(x_2 - d)^2)) \\
&= () * 1 + () * x_1 + () * x_2 + ... + () * x_1^4 + () * x_2^4 \\
&* () \quad represents \quad the \quad corresponding \quad coefficient.
\end{aligned}
$$

As we can see, when data is in e1, $f(x_1, x_2) = 1$. And when data is out of e1 while in e2, $f(x_1, x_2) = -1$. And when data is out of e2, $f(x_1, x_2) = 1$. We can also observe that the sign function is based on the feature mentioned above

### 1.4

For XOR data, we have (1,1) and (-1,-1) labeled -1, (1,-1) and (-1,1) labeled +1.

1. for Kernel $K(\underline{x}, \underline{y}) = (1 + (\underline{x} * \underline{y}))^2$

   We can express the dual SVM problem as follows:
   $\max\limits_{\alpha} \sum_i \alpha_i - 1/2 \sum_i \sum_j \alpha_i y_i (1 + (\underline{x} * \underline{y}))^2 y_j \alpha_i$

   Under this kernel, we use the svm from sklearn library to generate a separator. And below is the picture that illustrate the separator.

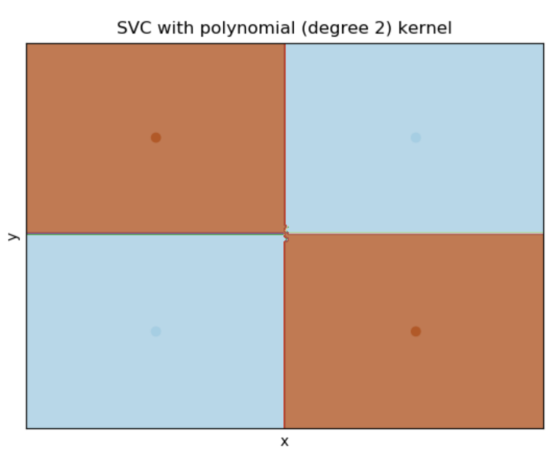


Figure 1: poly-svm

2. for Kernel $K(\underline{x}, \underline{y}) = exp(-|\underline{x} - \underline{y}|^2)$

   We can express the dual SVM problem as follows:
   $\max\limits_{\alpha} \sum_i \alpha_i - 1/2 \sum_i \sum_j \alpha_i y_i exp(-|\underline{x} - \underline{y}|^2) y_j \alpha_i$

   Under this kernel, we also take use of sklearn library to generate a separator. And below is the generator.

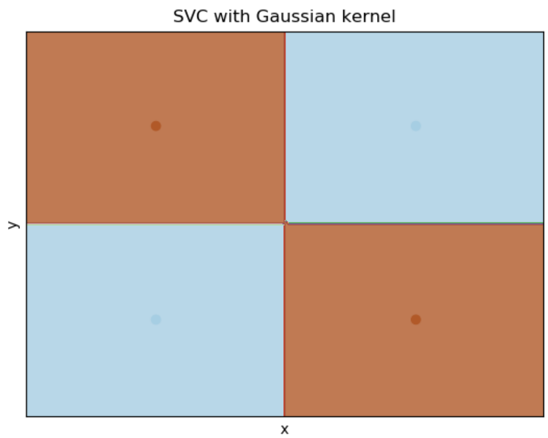


Figure 2: Gaussian-svm

   From above we can see Gaussian kernel generates a little better separator than poly kernel does.(the boundary is more smooth) But why is that? After searching on the Internet, I find that Gaussian kernel can map the feature space into an infinite feature space. And under the infinite feature space, every classification can be done. So this may be the reason why Gaussian kernel has a better effect.