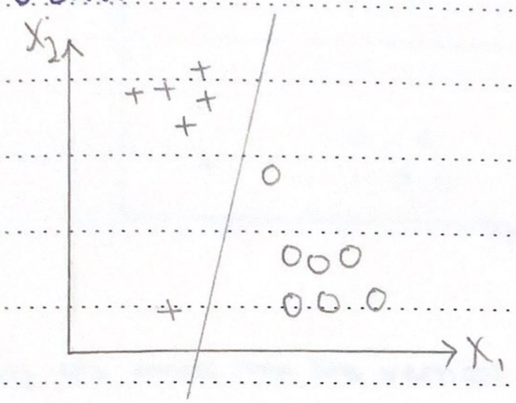
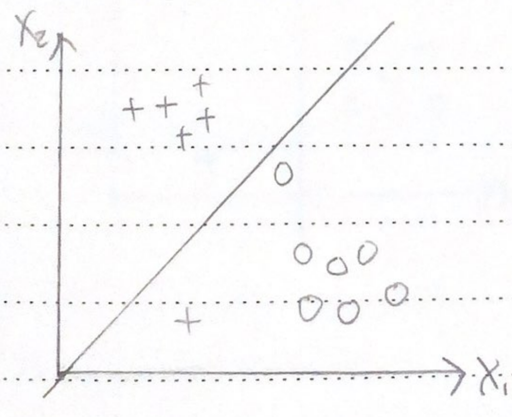
(1.1) As the data are linearly seperable, logistic regression will find a line that fits the data perfectly.

There is a unique ML decision boundary which makes 0 errors, but there are several possible decision boundaries which all makes 0 error.



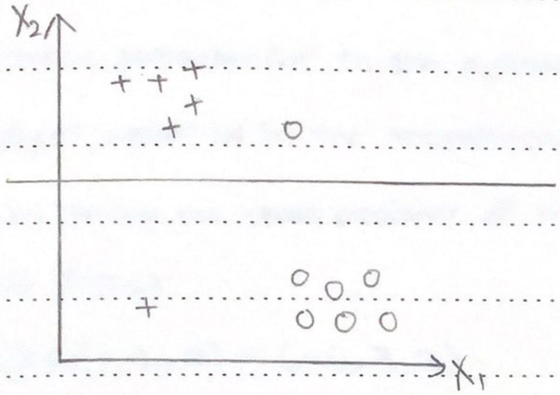
(1.2) Heavily regularizing w_0 sets $w_0 = 0$, this means point (0,0) must be on the decision boundary.

There are several possible line with different slopes, but all will make 1 error.



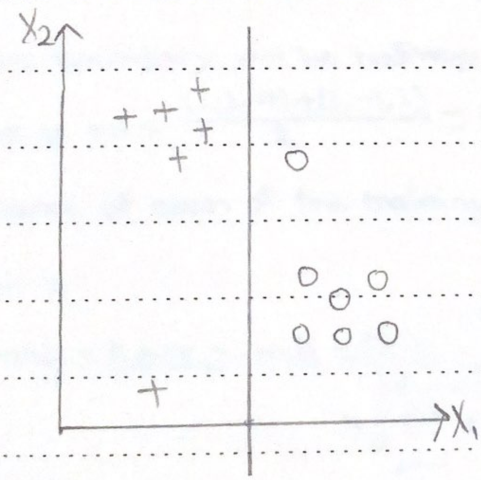
(1.3) Regularizing w_1 makes the line horizontal since x_1 is ignored.

There will be 2 classification errors.



(1.4) Regularizing w_2 makes the line vertical since x_2 is ignored.

There will be 0 classification errors.



(2.1) $\phi(x_1) = (1,-1,1)$ $\phi(x_2) = (1,2,4)$

w is perpendicular to the decision boundary between two points, which is a line through $\phi(x_1)$ and $\phi(x_2)$

A vector that is perpendicular to the optimal orientation of the weight vector w in the transformed 3D space can be found by taking the cross product of the feature vectors $\phi(x_1)$ and $\phi(x_2)$

$$(1,-1,1) \times (1,2,4) = (-6,3,3)$$

or any scalar multiple of this

(2.2) There are 2 support vectors, namely two data points.

The decision boundary will be halfway between them. This midpoint is $m = \frac{(1,2,4)+(1,-1,1)}{2} = (1, \frac{1}{2}, \frac{5}{2})$.

The distance of each of the training points to this midpoint is

$$\|\phi(x_1) - m\| = \|\phi(x_2) - m\| = \|(0,1,1)\|$$
$$= \sqrt{0^2+1^2+1^2}$$
$$= \sqrt{2}$$

Hence, the margin is $\sqrt{2}$

vector that is parallel to optimal vector w:
$(1,2,4)-(1,-1,1) = (0,3,3)$
$(0,1,1)$ is scalar multiple

(2.3) we have $w = (0, \frac{1}{2}, \frac{1}{2})$ which is parallel to $(0,3,3)$ and has $\|w\| = \sqrt{\frac{1}{2}^2 + \frac{1}{2}^2} = \frac{1}{\sqrt{2}}$ as required

(3.1) In this dataset, a linear SVM classifier cannot perfectly seperate the two classes (Class -1 and Class +1) due to the circular arrangement of data points. A linear SVM draws a straight line to seperate data points, but in this case, a single straight line cannot accurately divide the classes with no errors.

In this case, a nonlinear SVM classifier like the Radial Basis Function (RBF) kernel is needed. The RBF kernel allows us to map the data into a higher-dimensional feature space where it might become linearly seperable.

The kernel function is defined as:

$$K(x, x') = \exp(-\gamma \|x - x'\|^2)$$

where x and x' are data points

γ is a hyperparameter controlling kernel shape

By ~~approximating~~ appropriately choosing γ value and using RBF kernel, we can transform the data into a higher-dimensional space where it may be linearly seperable, allowing for effective classification of the circularly arranged data points.

(3.2) The Radial Basis Function (RBF) kernel is a kernelized method used in support vector machines (SVMs) for solving problems that involve non-linearly seperable data. It allows for non-linear seperation in the feature space by mapping data points into a higher-dimensional space.

- The RBF kernel computes the similarity (or distance) between data points in original feature space
- It uses a Gausian function to create a feature space where the data may become linearly seperable

Advantages of the RBF kernel over Linear SVM:

- Non-Linear Seperation: The RBF kernel can handle complex and non-linear decision boundaries. In the specific dataset provided, the circular arrangement of data points cannot be seperated by a straight line, but the RBF kernel can capture the circular pattern by transforming the data into a higher-dimensional space
- Flexibility: The RBF kernel allows for a high degree of flexibility in modelling complex relationships in the data. By adjusting the "γ" parameter, it allows for control of smoothness and complexity of the decision boundary, making it adaptable to different types of data distribution

- Better Fit: The RBF kernel can model intricate patterns and adapt to irregularly shaped clusters. In the provided dataset, it can accurately capture the circular seperation pattern that linear SVM cannot.
- Generalization: The RBF kernel often provides better generalization to unseen data. It can capture the underlying data distribution more accurately, which is particularly useful in cases where the dataset is noisy or when making accurate predictions on new, unseen data.

(4.1) The margin width γ is defined as the distance between the two hyperplanes that are equidistant to the decision boundary. We can express γ in terms of the support vectors, which are data points that lie on the margin.

$$\gamma = \frac{2}{\|w\|}$$

where w is the normal vector to the decision boundary

We can also express w in terms of the support vectors and the kernel functions:

$$w = \sum_i a_i y_i \phi(x_i)$$

The inner product between two transformed feature vectors $\phi(x_i)$ and $\phi(x_j)$ can be expressed as:

$$\phi(x_i) \cdot \phi(x_j) = K(x_i, x_j)$$

Using inner product property, we can rewrite $\|w\|$ as:

Let $n=i, m=j$ $$\|w\| = \sqrt{\sum \sum a_i a_j y_i y_j K(x_i, x_j)}$$

Now substituting $\|w\|$ in the expression for margin width γ:

$$\gamma = \frac{2}{\sqrt{\sum \sum a_i a_j y_i y_j K(x_i, x_j)}}$$

This expression relates the margin width γ to the Lagrange multipliers $\{a_n\}$ through the kernel function K. The kernel function captures the influence of the transformation ϕ on the inner product between feature vectors

(4.2) We can rewrite the expression $\frac{1}{\gamma^2}$ as follows:

$$\frac{1}{\gamma^2} = \frac{1}{\frac{4}{\left(\|\sum_i a_i y_i \phi(x_i)\|^2\right)}}$$

$$= \frac{\|\sum_i a_i y_i \phi(x_i)\|^2}{4}$$

$$= \frac{\|w\|^2}{4}$$

Since $\|w\|^2$ is related to the Lagrange multipliers $\{a_n\}$, the original relationship $\frac{1}{\gamma^2} = \sum_{n=1}^{N} a_n$ holds true even under transformation ϕ.

The introduction of the transformation ϕ in SVMs emphasizes implicit mapping to a higher-dimensional feature space, influencing the decision boundary's geometry. Kernels facilitate operations in this transformed space. The weight vector (w) is impacted by the transformed feature vectors, highlighting SVM's ability to handle non-linear boundaries. The proof underscores the relationship between $\frac{1}{\gamma^2}$ and $\sum_{n=1}^{N} a_n$'s validity under transformations, showcasing SVM robustness