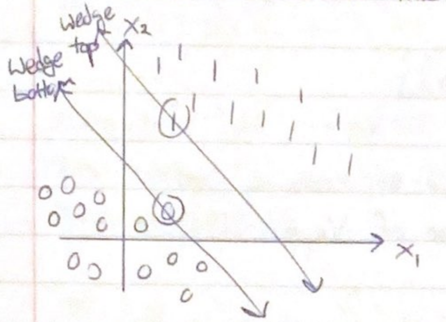
M342W 2/10/21

$\mathcal{Y} = \{0, 1\}$, $p+1 = 3$, $\mathcal{H} = \{ \mathbb{1}_{\vec{w} \cdot \vec{x} \geq 0} : \vec{w} \in \mathbb{R}^3 \}$

Assume the data is linearly separable so it looks like:



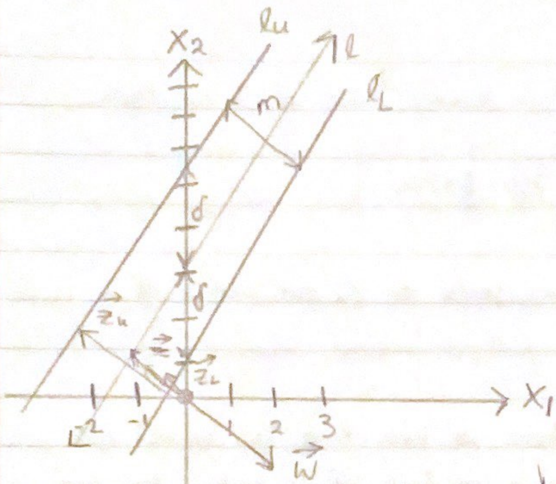
We need an algorithm that locates the middle of that wedge.
Let the top of the wedge be the linearly separable model "closest" to the $y=1$'s and the bottom of the wedge be the linearly separable model "closest" to the $y=0$'s. The "max margin hyperplane" is the parallel line in the center of the top and bottom.

Note: there are two critical observations (the circled points). Since observations are x-vectors, these critical observations are called "support vectors" and hence the final model is called a "support vector machine" (SVM). "Machine" is a fancy word meaning "complex model". So, "machine learning" just means "learning complex models."
To find the SVM...

first rewrite $\mathcal{H} = \{ \mathbb{1}_{\vec{w} \cdot \vec{x} - b \geq 0} : \vec{w} \in \mathbb{R}^p, b \in \mathbb{R} \}$

Note $\underbrace{\vec{w} \cdot \vec{x} - b = 0}_{\text{Hesse Normal Form}}$ defines a line/hyperplane.

$$\ell : x_2 = 2x_1 + 3 \Rightarrow \ell : 2x_1 - x_2 + 3 = 0 \Rightarrow \ell : \overbrace{\begin{bmatrix} 2 \\ -1 \end{bmatrix}}^{\vec{w}} \cdot \vec{x} - \overbrace{(-3)}^{b} = 0$$



The $\vec{w}$ is perpendicular to l (line) and called the "normal vector."

Let $\vec{w}_0 := \frac{\vec{w}}{\|\vec{w}\|}$,

The direction of $\vec{w}$ with unit length.

$\vec{z} = \alpha \vec{w}_0$, $\vec{z} \in l$

$\vec{w} \cdot \vec{z} - b = 0$

$\Downarrow$

$\vec{w} \cdot (\alpha \vec{w}_0) - b = 0 \Rightarrow \frac{\alpha}{\|\vec{w}\|} \|\vec{w}\|^2 - b = 0$

$\Rightarrow \alpha = \frac{b}{\|\vec{w}\|} \Rightarrow \vec{z} = \frac{b}{\|\vec{w}\|} \vec{w}_0$

~~Th~~ Let $m > 0$ be the $\perp$ distance between l_u and l_L and let $\delta > 0$ be the distance between l_u and l (and l_L and l) on the x_2 axis.

l_u: $\vec{w} \cdot \vec{x} - (b + \delta) = 0$, $\vec{z}_u = \frac{b+\delta}{\|\vec{w}\|} \vec{w}_0$

l_L: $\vec{w} \cdot \vec{x} - (b - \delta) = 0$, $\vec{z}_L = \frac{b-\delta}{\|\vec{w}\|} \vec{w}_0$

$\Downarrow$

$$m = \|\vec{z}_u - \vec{z}_L\| = \left\| \frac{b+\delta}{\|\vec{w}\|} \vec{w}_0 - \frac{b-\delta}{\|\vec{w}\|} \right\| = \frac{1}{\|\vec{w}\|} 2\delta \|\vec{w}_0\| = \frac{2\delta}{\|\vec{w}\|}$$

Goal is to make m as large as possible (maximum margin) $\Leftrightarrow$ (making the $\vec{w}$ as small as possible).

Hesse Normal form is not unique There are infinite equivalent specification of a line:

$\forall c \neq 0 \quad c(\vec{w} \cdot \vec{x} - b) = 0$. Let $c = \frac{1}{\delta}$

$\Downarrow$

$$m = \frac{2}{\|\vec{w}\|}$$

Now we need two conditions.

1) All $y = 1$'s are above or equal to l_u. $\forall_i$ s.t. $y_i = 1$ $\vec{w} \cdot \vec{x}_i - (b+1) \geq 0$

$\Rightarrow \vec{w} \cdot \vec{x}_i - b \geq 1 \Rightarrow \frac{1}{2}(\vec{w} \cdot \vec{x}_i - b) \geq 1/2 \Rightarrow (y_i - 1/2)(\vec{w} \cdot \vec{x}_i - b) \geq 1/2$

2) All $y = 0$'s are below or equal to ℓ_L:

$\forall_i$ s.t. $y_i = 0$ $\quad \vec{w} \cdot \vec{x}_i - (b-1) \leq 0 \Rightarrow \vec{w} \cdot \vec{x}_i - b \leq -1 \Rightarrow \frac{1}{2}(\vec{w} \cdot \vec{x}_i - b) \leq -1/2$

$\Rightarrow -1/2(\vec{w} \cdot \vec{x}_i - b) \geq 1/2 \quad \Rightarrow (y_i - 1/2)(\vec{w} \cdot \vec{x}_i - b) \geq 1/2$

Note how both inequalities are the same for both $\overset{(1)}{\text{I}}$ and $\overset{(2)}{\text{II}}$. Thus this inequality satisfies both constraints. So all observations will be in their right places.

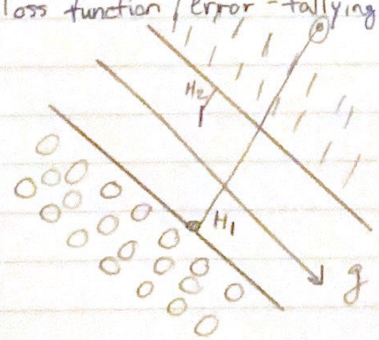
$$\underbrace{\forall_i \; (y_i - 1/2)(\vec{w} \cdot \vec{x}_i - b) \geq 1/2}\ \Rightarrow \text{line is linearly separable.}$$

You compute the SVM by optimizing the following problem.
min $\|\vec{w}\|$ s.t. $\rightarrow$ is true.

and return the ~~solution~~ resulting $\vec{w}$ and b. There is no analytical solution. You need optimization algorithms. It can be solved with quadratic programming and other procedures as well.

Note: everything we did above ~~generalization~~ generalizes to $p > 2$. Note: most txtbooks have 1's in the place of our 1/2's that's because they assumed $y = \{-1, 1\}$ but we assumed binary.

What is the data is not linearly separable? You can never satisfy that constraint. So this whole thing doesn't work. We will use a new objective function/ loss function/ error-tallying function called "hinge loss," H:



$$H_i := \max\left\{0, 1/2 - \overbrace{(y_i - \tfrac{1}{2})(\vec{w} \cdot \vec{x}_i - b)}^{\text{Should be } \geq 1/2}\right\}$$

Let's say a point is d away from where it should be.

$(y_i - 1/2)(\vec{w} \cdot \vec{x}_i - b) = \frac{1}{2} - d$

$H_i = \max\{0, 1/2 - (1/2 - d)\} = \max\{0, d\} = d$

With this loss function, it is clear we wish to minimize the sum of the hinge errors:

$$SHE := \sum_{i=1}^{n} \max\left\{0, \tfrac{1}{2} - (y_i - \tfrac{1}{2})(\vec{w} \cdot \vec{x}_i - b)\right\}$$

But we also want to maximize the margin. So we combine both considerations together into the objective function of Vapnik (1963):

$$\underset{\vec{w}, b}{\text{argmin}} \left\{ \underbrace{\frac{1}{n} SHE}_{\text{minimizing distance errors}} + \underbrace{\lambda}_{} \underbrace{\|\vec{w}\|^2}_{} \right\}$$

Once λ is set, the computer can do the optimization to find the resulting SVM.

Maximizing the width of the wedge.

What is λ?

It is a "hyperparameter", "tuning parameter". It is set by you! It controls the tradeoff between these two considerations.

$g = \mathcal{A}(\mathbb{D}, \mathcal{H}, \lambda)$

What if you have the modelling setting where $= \{1, 2, \ldots, L\}$, a nominal categorical response with $L > 2$ levels. The model will still be a "classification model" but not a "binary classification model" and it's sometimes called a "multinomial classification model". What is the null model g_0? Again, $g_0 =$ Sample Mode $[y]$.

Consider a model that predicts on a new x_* by looking through the training data and finding the "closest" $\vec{x}_i$ and returning its y_i as the predicted response value. This is called a "nearest neighbor" model. Further, you may also want to find the K closest observations and return the mode of these K observations as the predicted response value (randomize ties). That's called "K nearest neighbors" (KNN) model where K is a natural number hyperparameter. There is another hyperparameter that must be specified, the "distance function" $d : \mathcal{X}^2 \to \mathbb{R}_{\geq 0}$. The typical distance function is Euclidean distance squared: $d(\vec{x}_*, \vec{x}_i) := \sum_{j=1}^{p} (x_{i,j} - x_{*,j})^2$

What is $\mathcal{H}$? $\mathcal{A}$?