Support Vector Machine

prof. Edson Cilos Vargas Júnior
Universidade Federal de Santa Catarina

Introdução

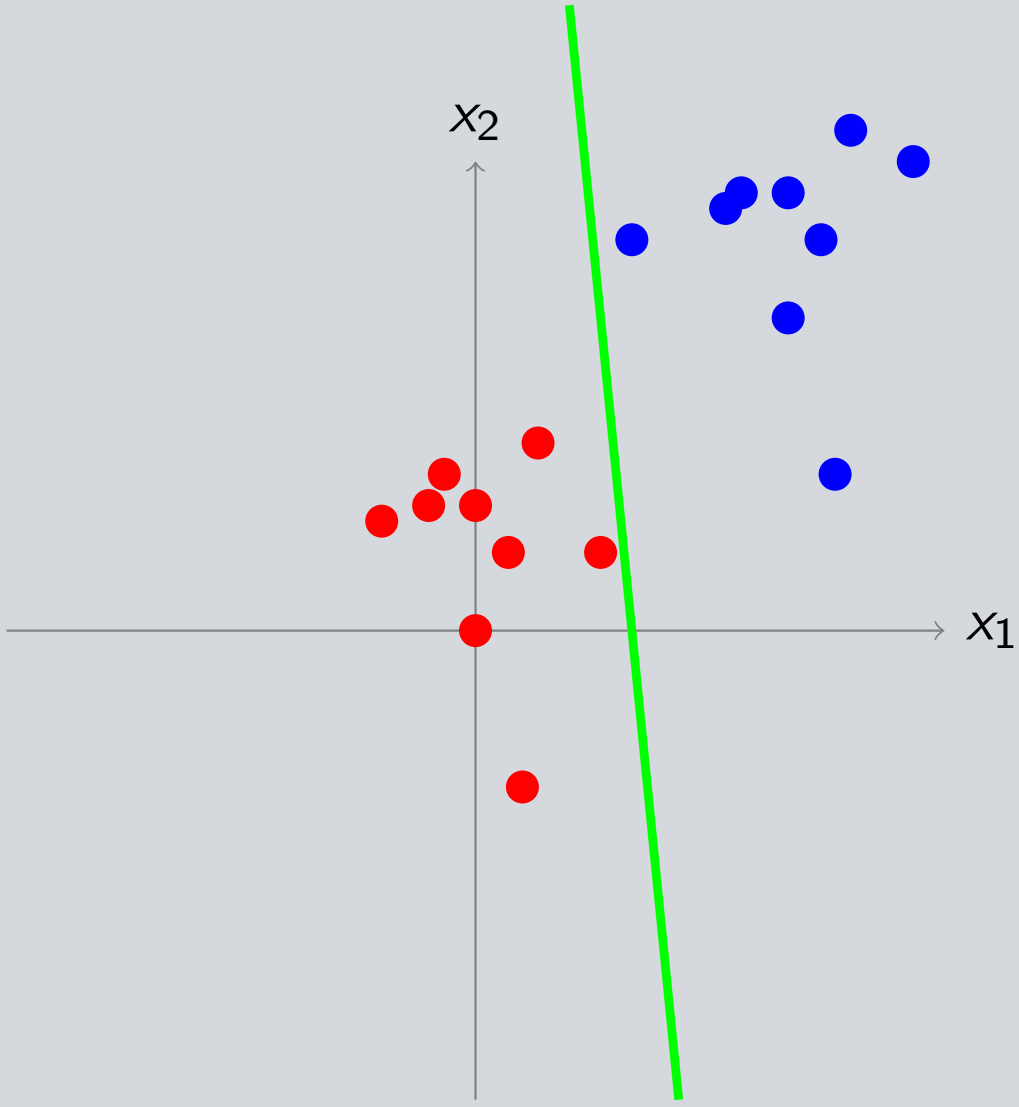
Máquina de vetores de suporte (SVM em inglês) é um conjunto de métodos de aprendizado supervisionado usado para classificação, regressão e detecção de outliers.

Support Vector Machine

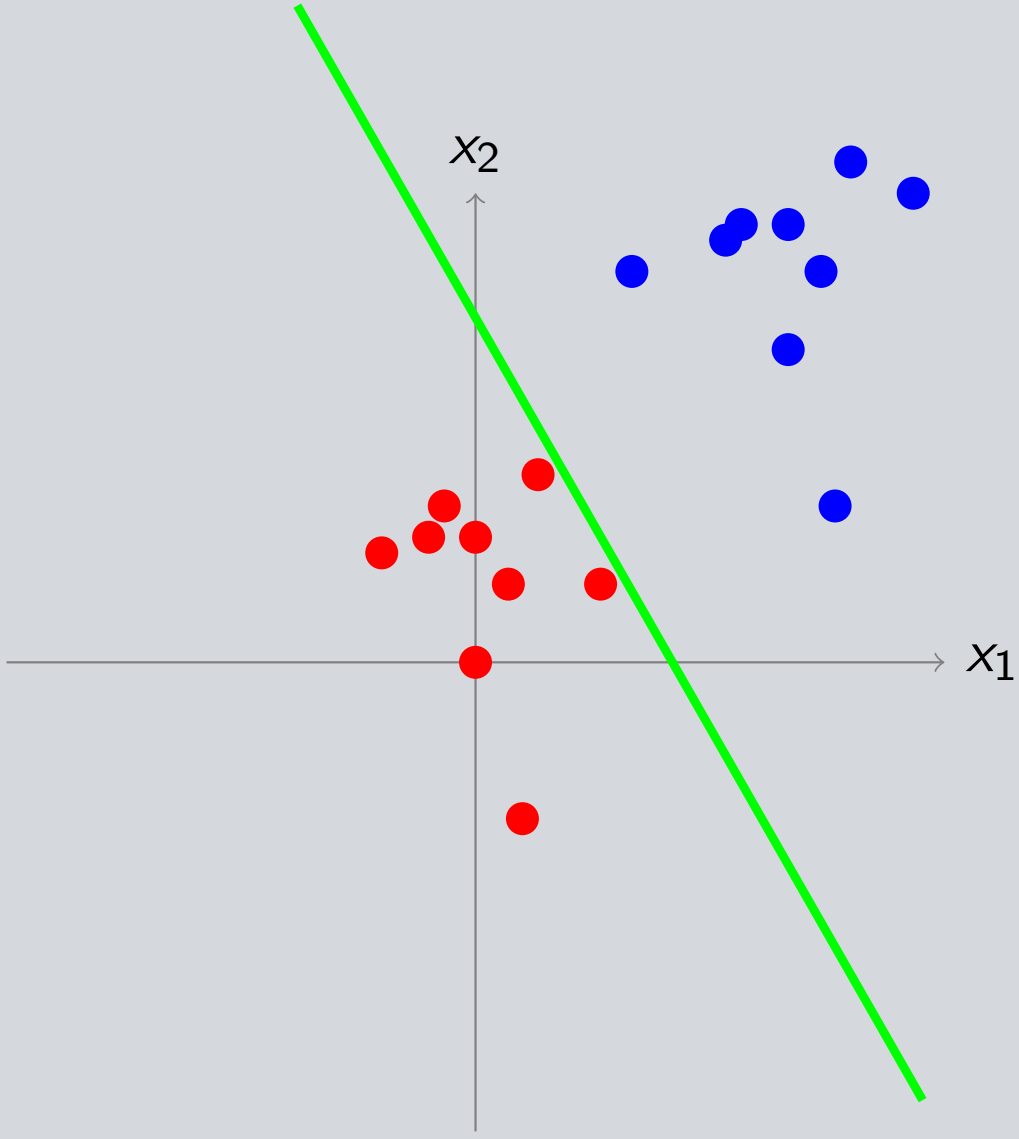
Até então temos ilustrado, de maneira vaga, o SVM.

- Foi dito que o SVM tenta encontrar hiperplanos que separam os dados;
- Mas o quê significa na prática e como funciona o algoritmo?

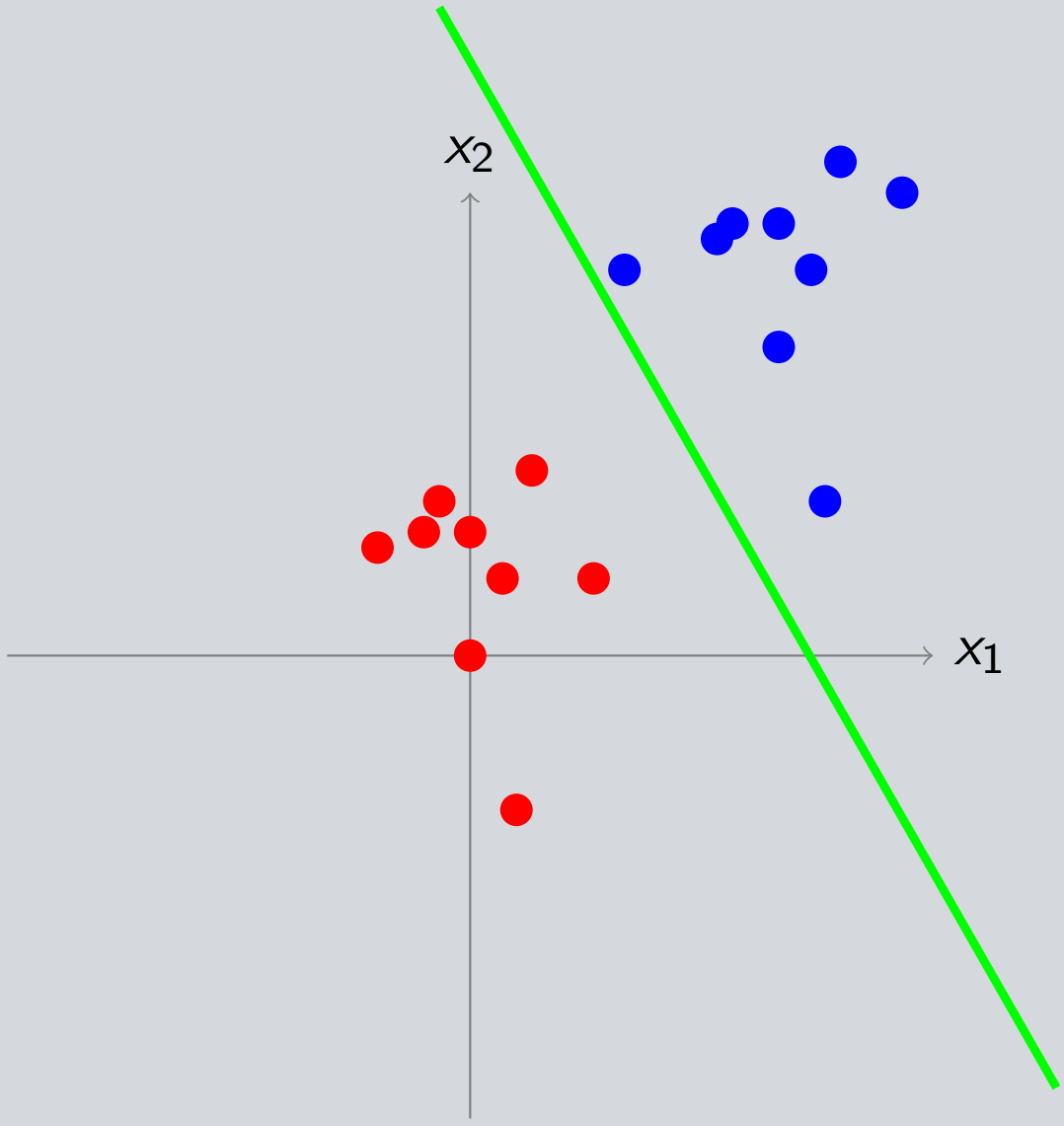
Encontrando hiperplano



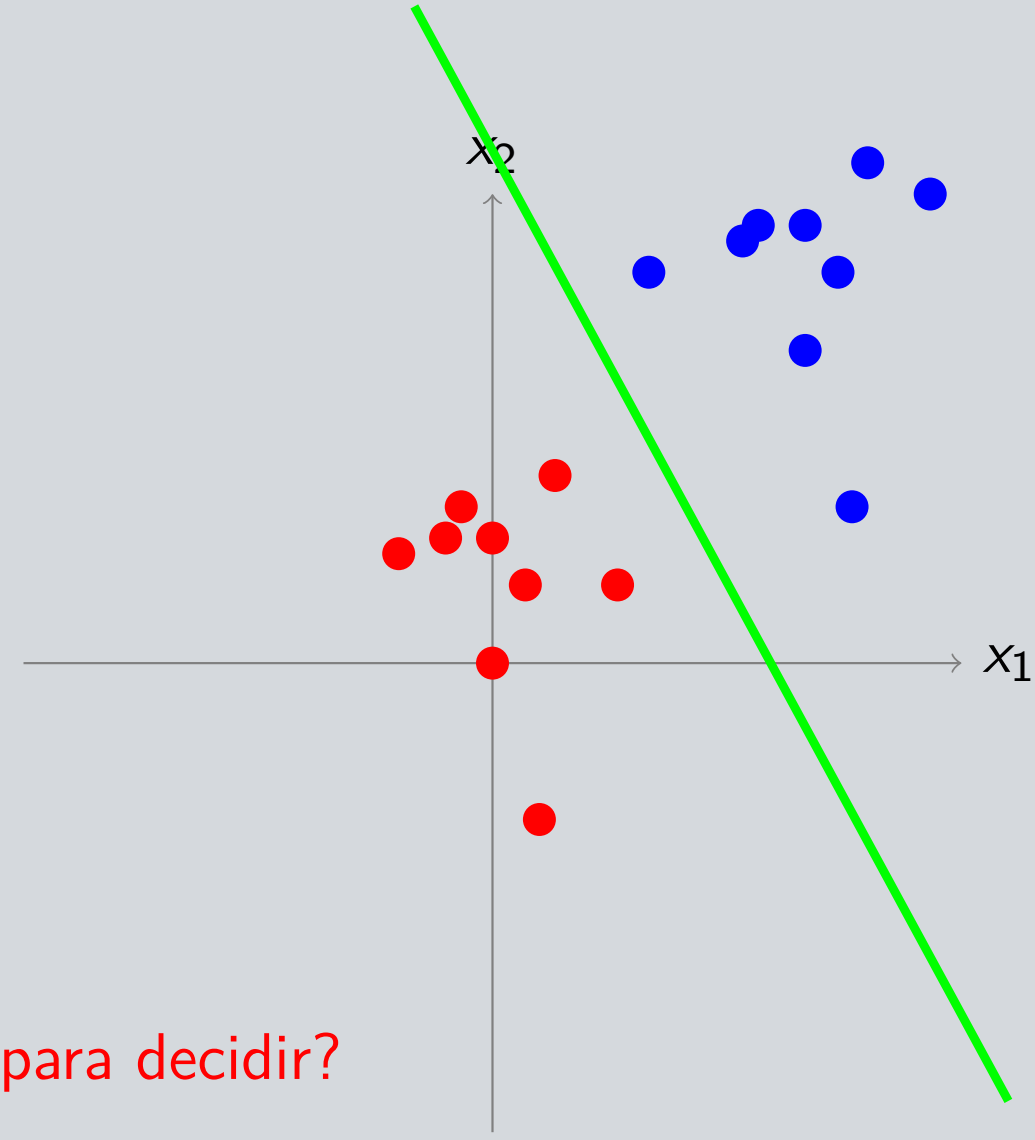
Essa seria uma opção melhor?



Ou esta?

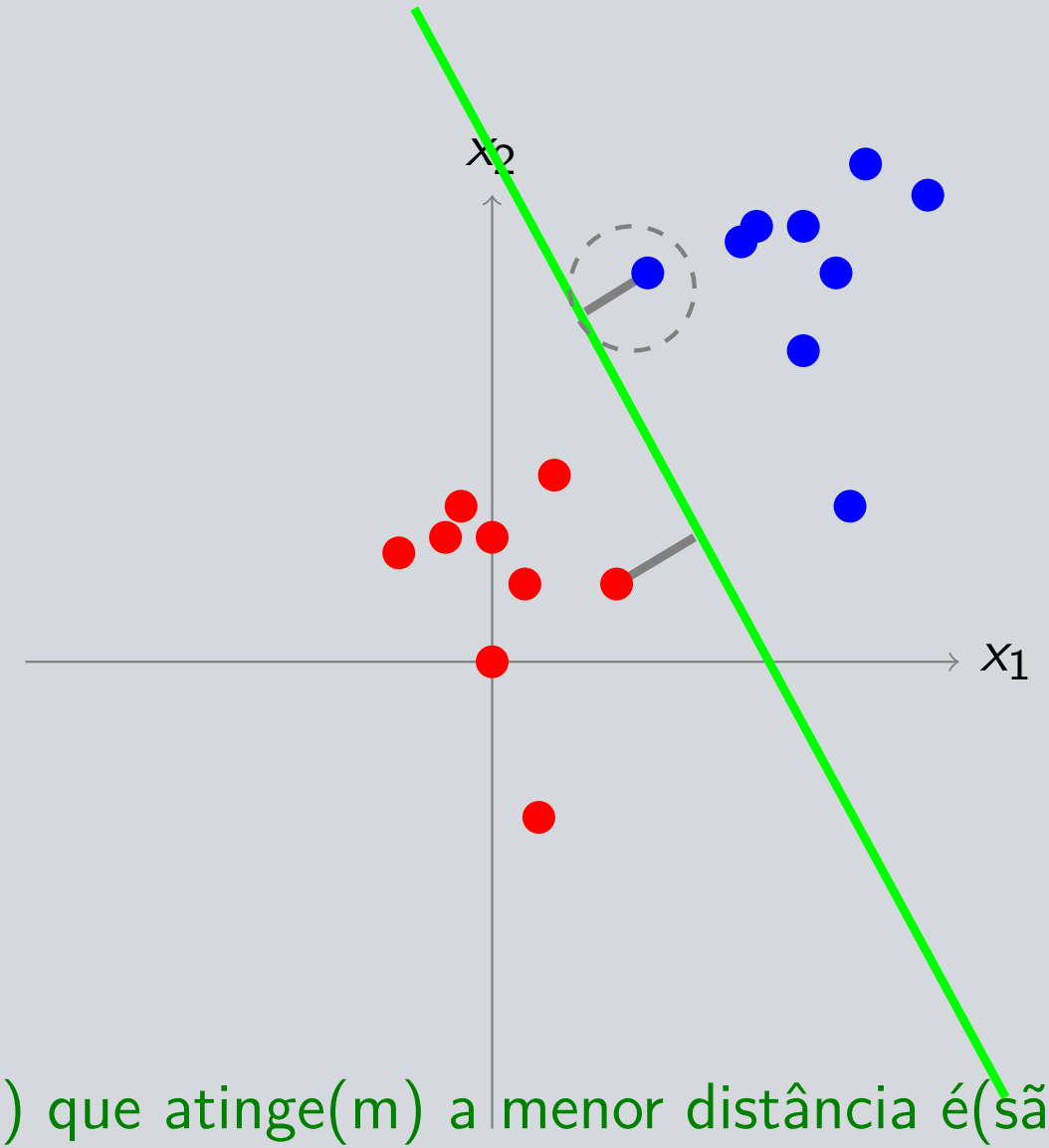


Ou seria essa?



Qual critério para decidir?

Minimizar distâncias!



O(s) vetor(es) que atinge(m) a menor distância é(são) chamado(s) de vetor(es) de suporte.

Sensibilidade à escala do SVM

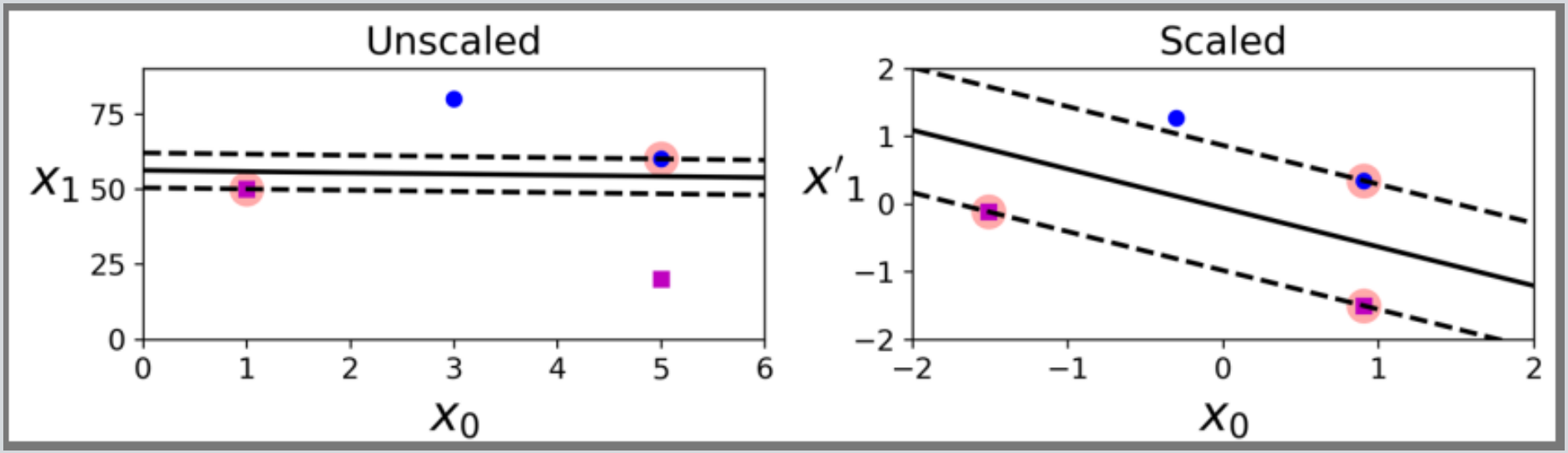


Figura 1: Extraído de Gron, A. Hands-On Machine Learning with Scikit-Learn and TensorFlow: Concepts, Tools, and Techniques to Build Intelligent Systems. Sebastopol: O'Reilly Media, Inc, 2017.

Separação de hiperplanos

Considere uma função:

$$f_{w,b}: \mathbb{R}^m \to \mathbb{R}$$
$$x \mapsto < x, w > + b.$$

parametretrizada por $w \in \mathbb{R}^m$ e $b \in \mathbb{R}$.

Separação de hiperplanos

Considere uma função:

$$f_{w,b}: \mathbb{R}^m \to \mathbb{R}$$
$$x \mapsto < x, w > + b.$$

parametretrizada por $w \in \mathbb{R}^m$ e $b \in \mathbb{R}$.

- Hyperplanos são subespaços afins;
- Um hiperplano pode ser escrito na forma:

$$H_{b,w} := \{x \in \mathbb{R}^m : f_{b,w}(x) = 0\};$$

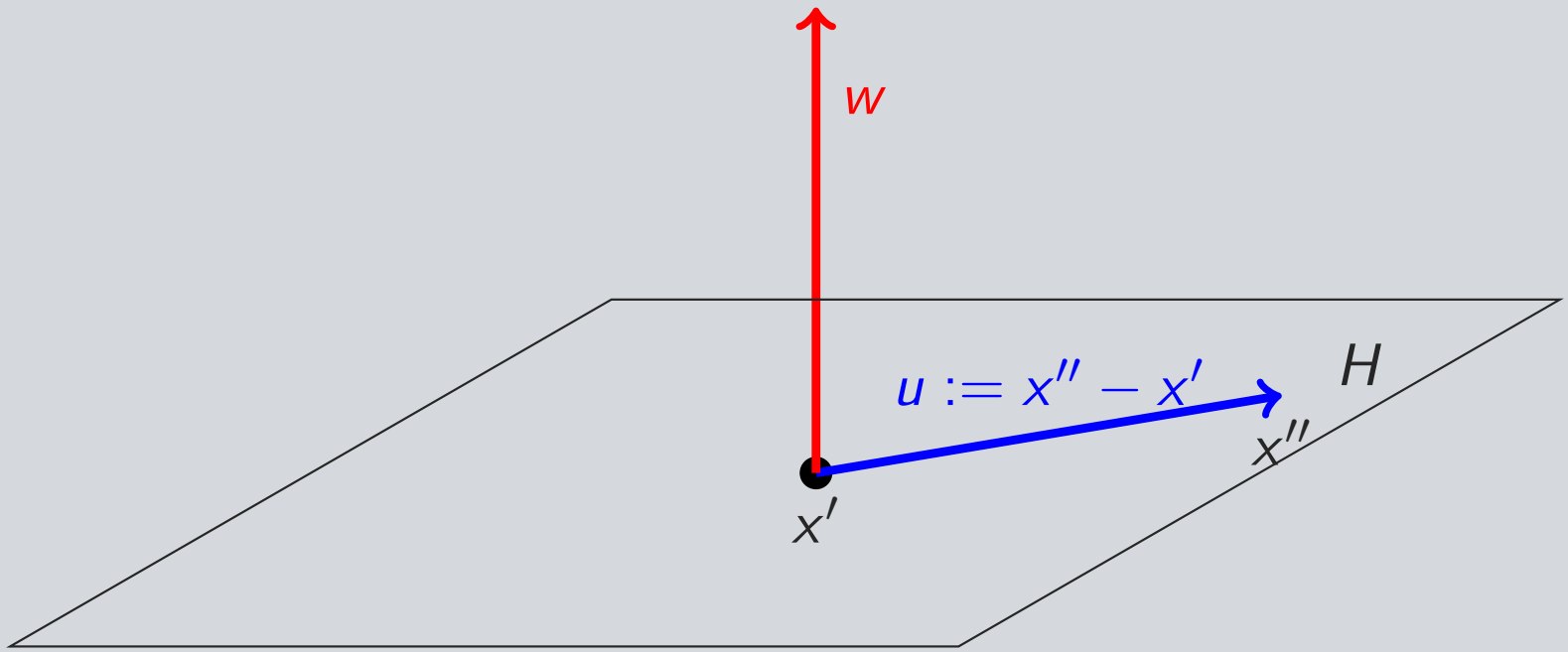
- $\mathcal{H} := \{f_{w,b} : w \in \mathbb{R}^m \text{ e } b \in \mathbb{R}\}$ é um conjunto de hipóteses que representa todos os hiperplanos que separam $\mathbb{R}^m$ em dois conjuntos (daí a aplicação para um problema de classificação binária).

Vejamos que o vetor w é sempre ortogonal ao hiperplano H.

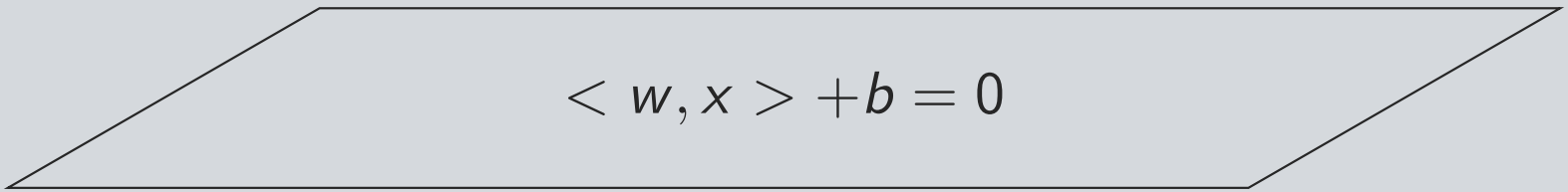
- Considere x' e x'' dois pontos do hiperplano H;
- $f(x') = f(x'') = 0$ e além disso:

$$< x'' - x', w >= f(x'') - f(x') = 0;$$

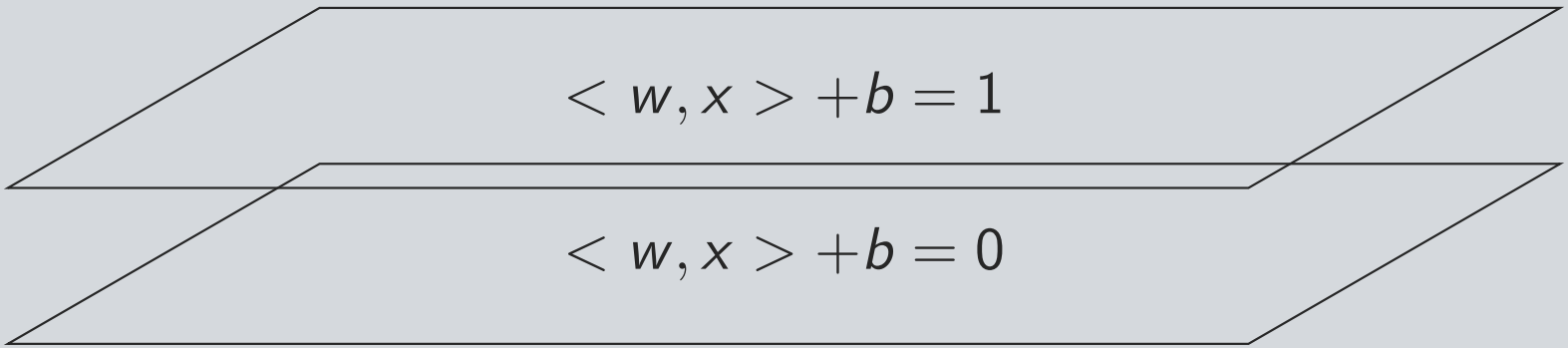
- Ou seja, w é ortogonal a qualquer vetor de H.



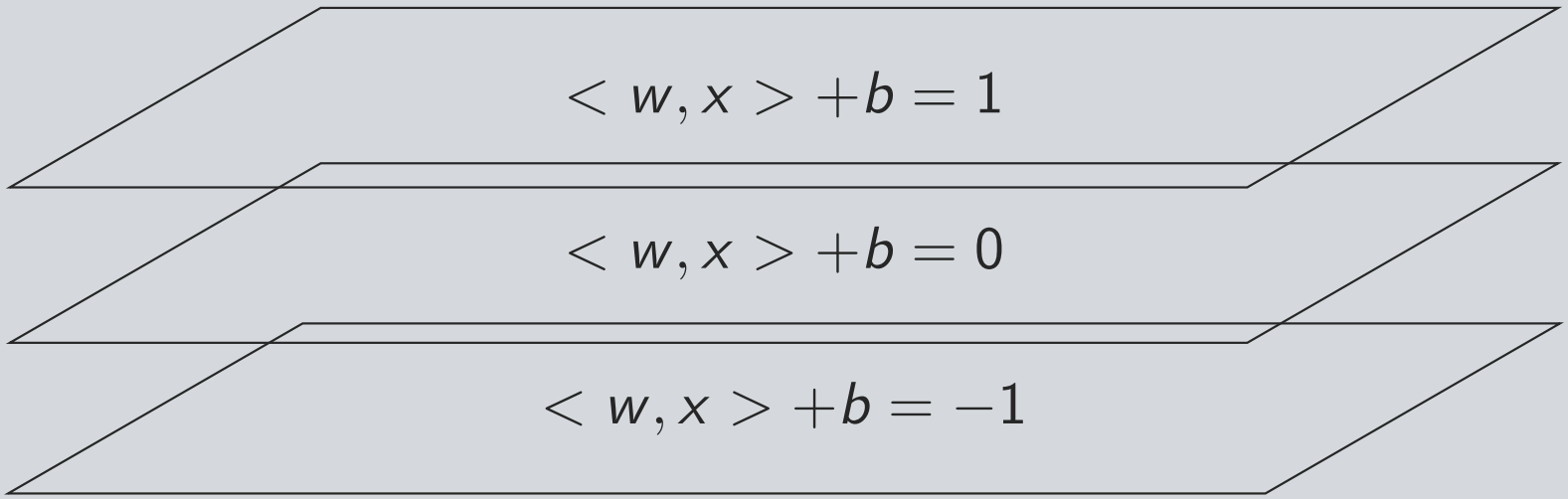
Um pouco da geometria...



Um pouco da geometria...



Um pouco da geometria...



Visualizando em $2D$

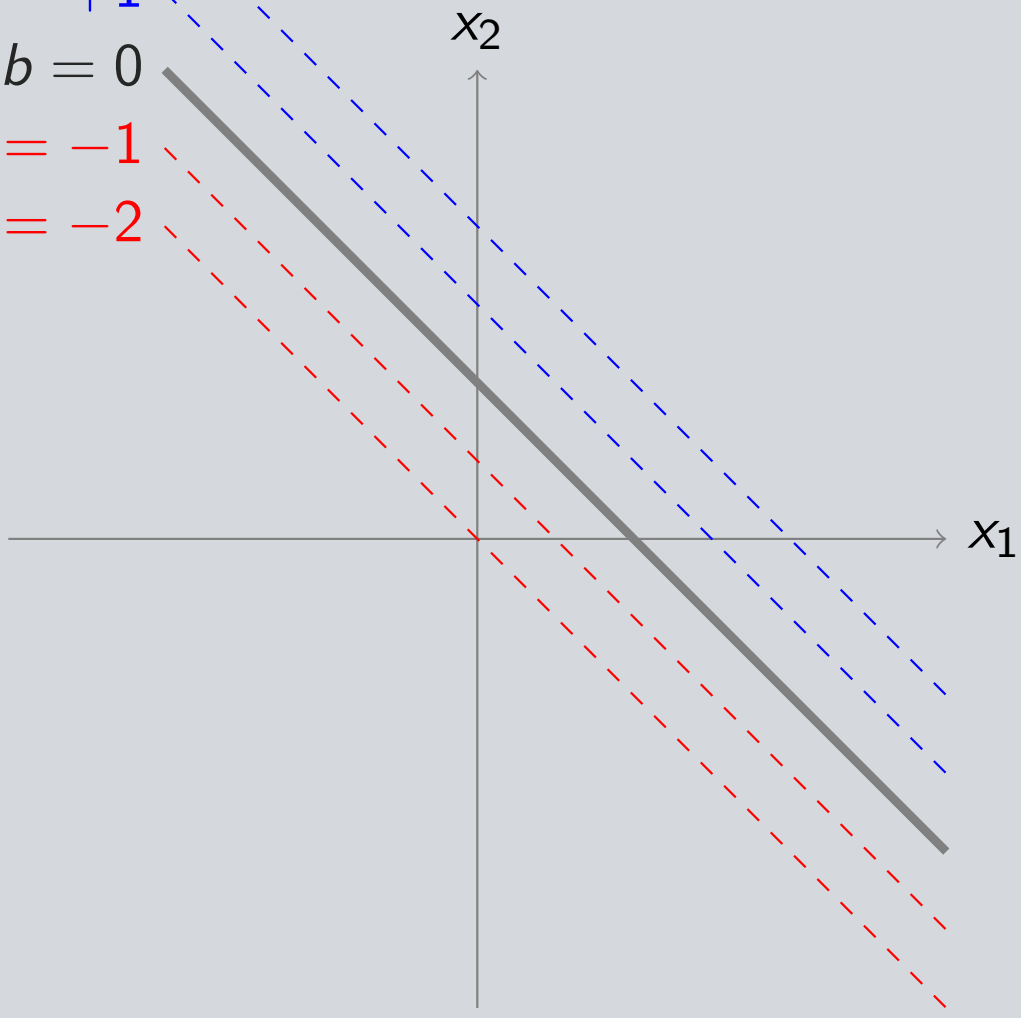
$< w, x > + b = +2$

$< w, x > + b = +1$

$< w, x > + b = 0$

$< w, x > + b = -1$

$< w, x > + b = -2$



Considerando uma amostra rótulada $\mathcal{D} = \{(x^{(1)}, y^{(1)}), \cdots, (x^{(N)}, y^{(N)})\}$ na qual $y^{(i)} \in \{-1, +1\}$, decidimos o rótulo da seguinte maneira:

$$< x^{(n)}, w > + b \geq 0 \iff y^{(n)} = +1$$

$$< x^{(n)}, w > + b < 0 \iff y^{(n)} = -1$$

Considerando uma amostra rótulada $\mathcal{D} = \{(x^{(1)}, y^{(1)}), \cdots, (x^{(N)}, y^{(N)})\}$ na qual $y^{(i)} \in \{-1, +1\}$, decidimos o rótulo da seguinte maneira:

$$< x^{(n)}, w > + b \geq 0 \iff y^{(n)} = +1$$

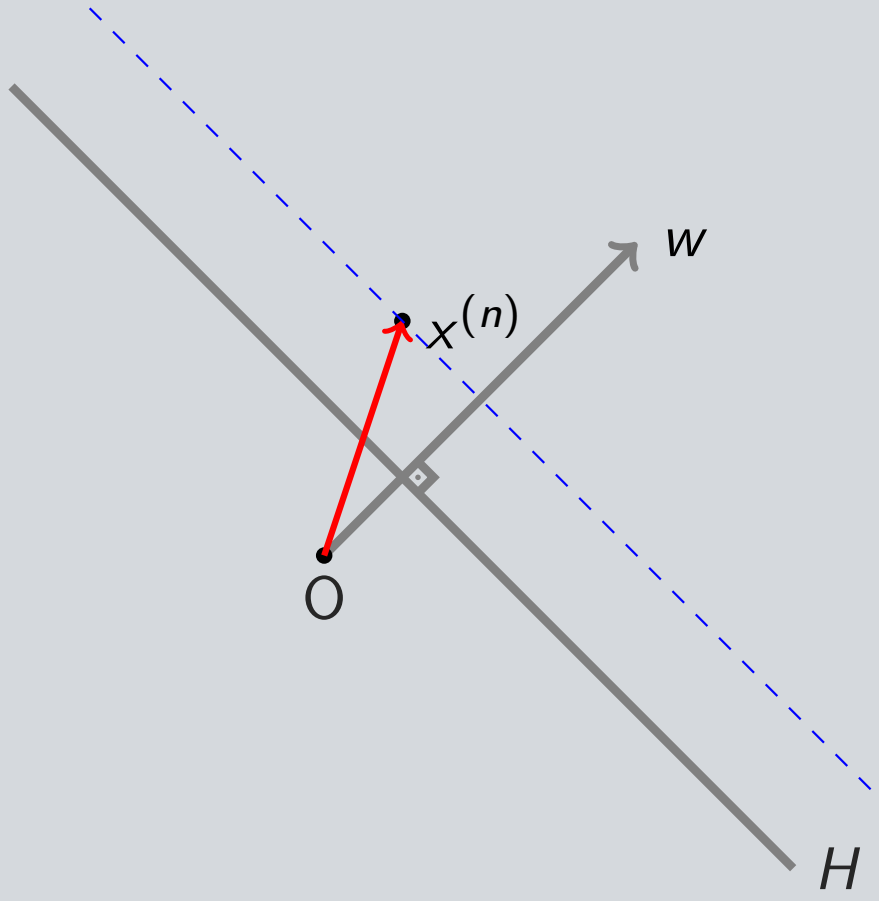
$$< x^{(n)}, w > + b < 0 \iff y^{(n)} = -1$$

Podemos considerar uma única desigualdade:

- $y^{(n)} \left(< x^{(n)}, w > + b \right) \geq 0$

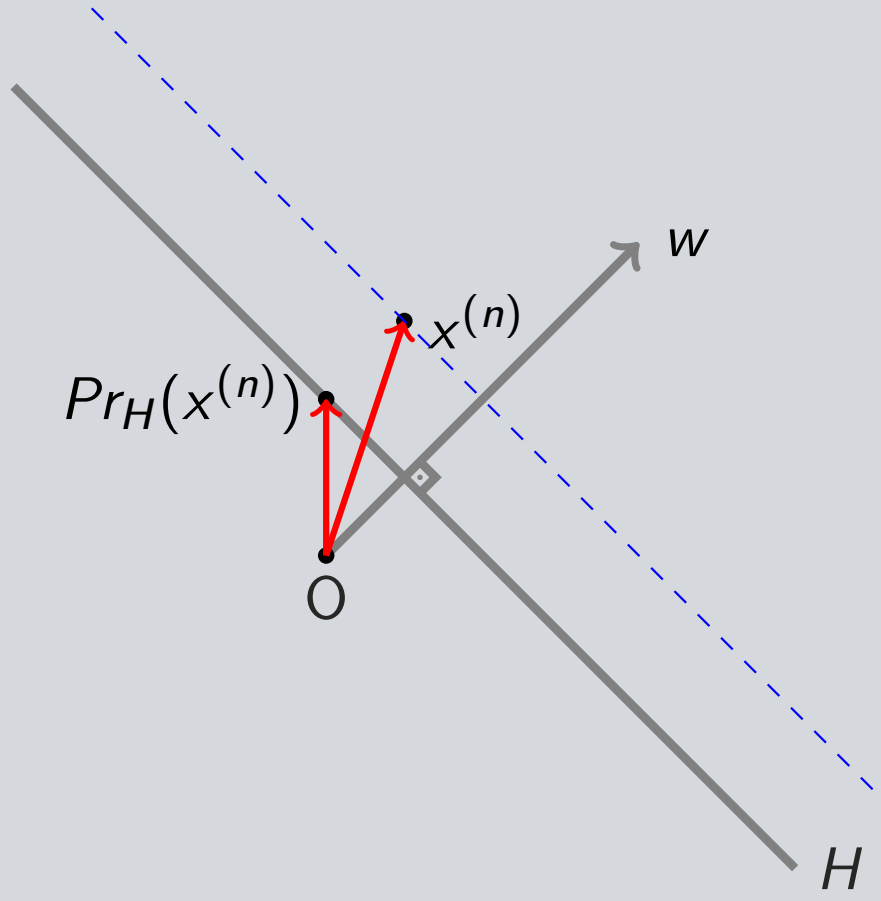
Conceito de Margem

Considere $x^{(n)}$ o ponto mais próximo ao hiperplano H



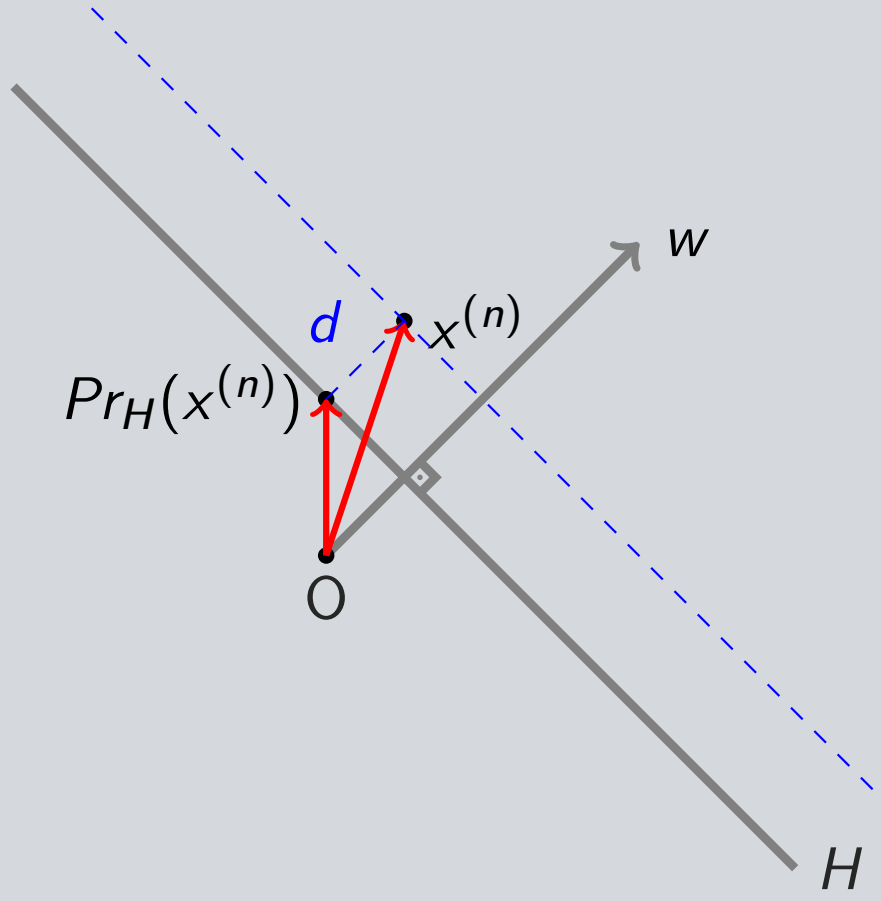
Conceito de Margem

Considere $x^{(n)}$ o ponto mais próximo ao hiperplano H



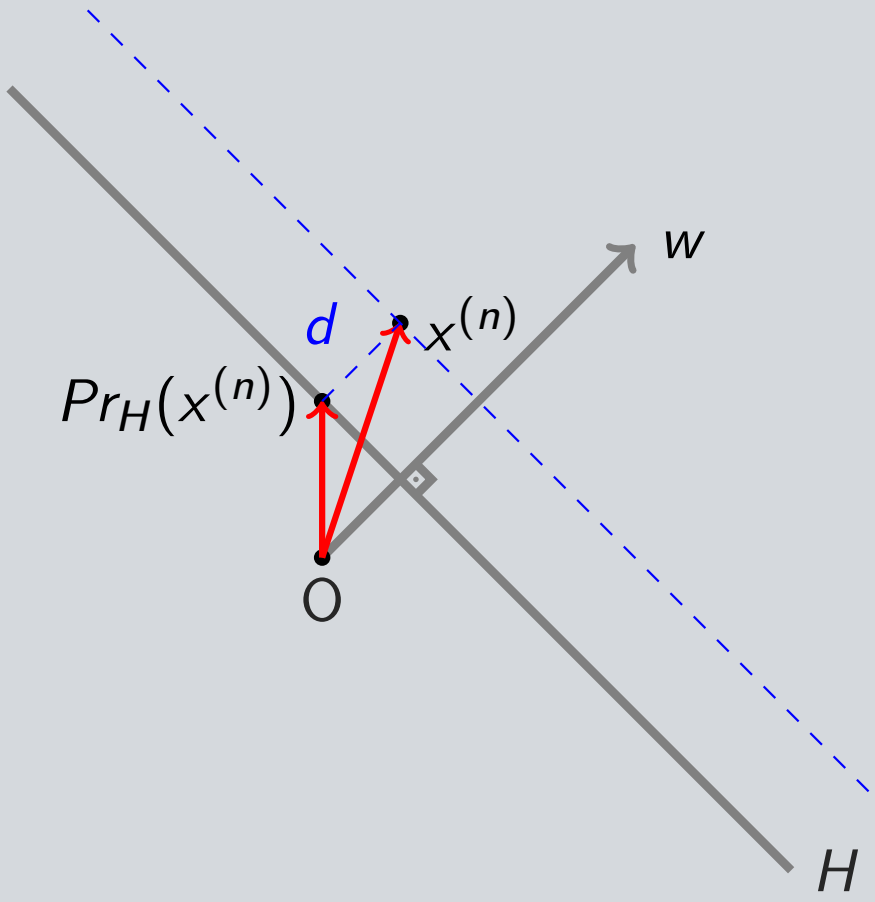
Conceito de Margem

Considere $x^{(n)}$ o ponto mais próximo ao hiperplano H

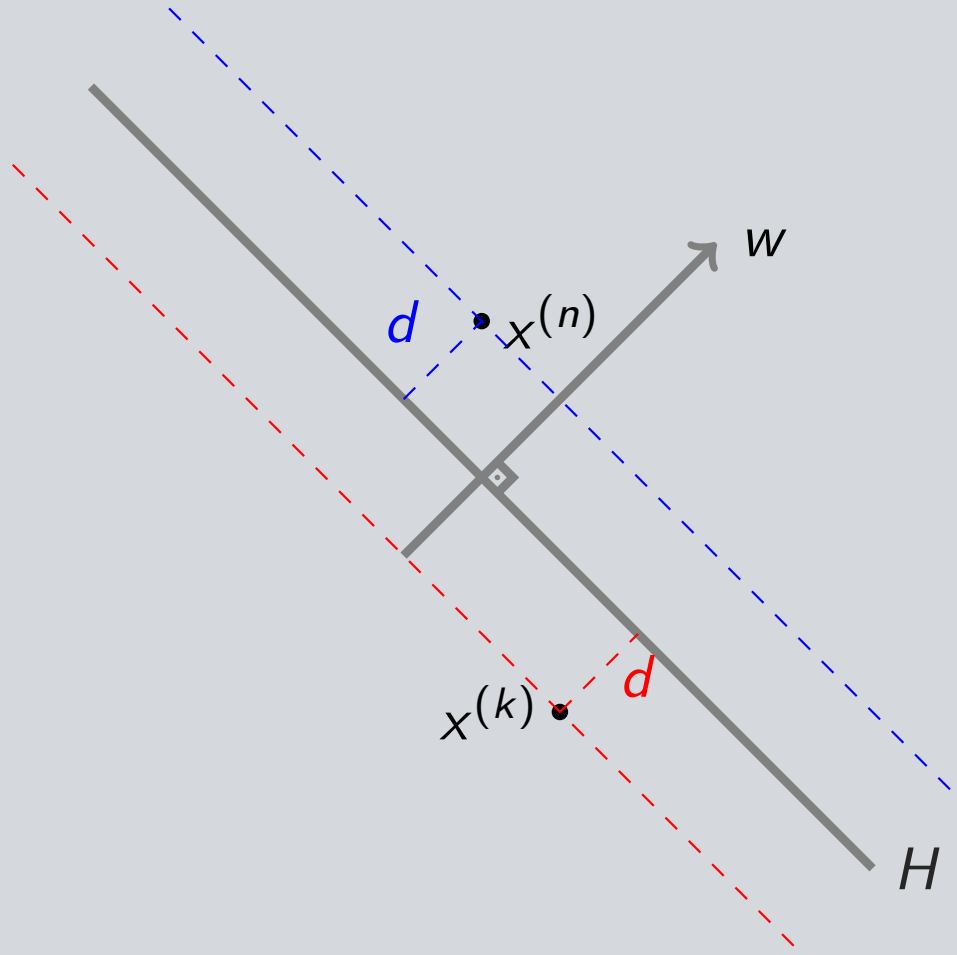


Conceito de Margem

Considere $x^{(n)}$ o ponto mais próximo ao hiperplano H

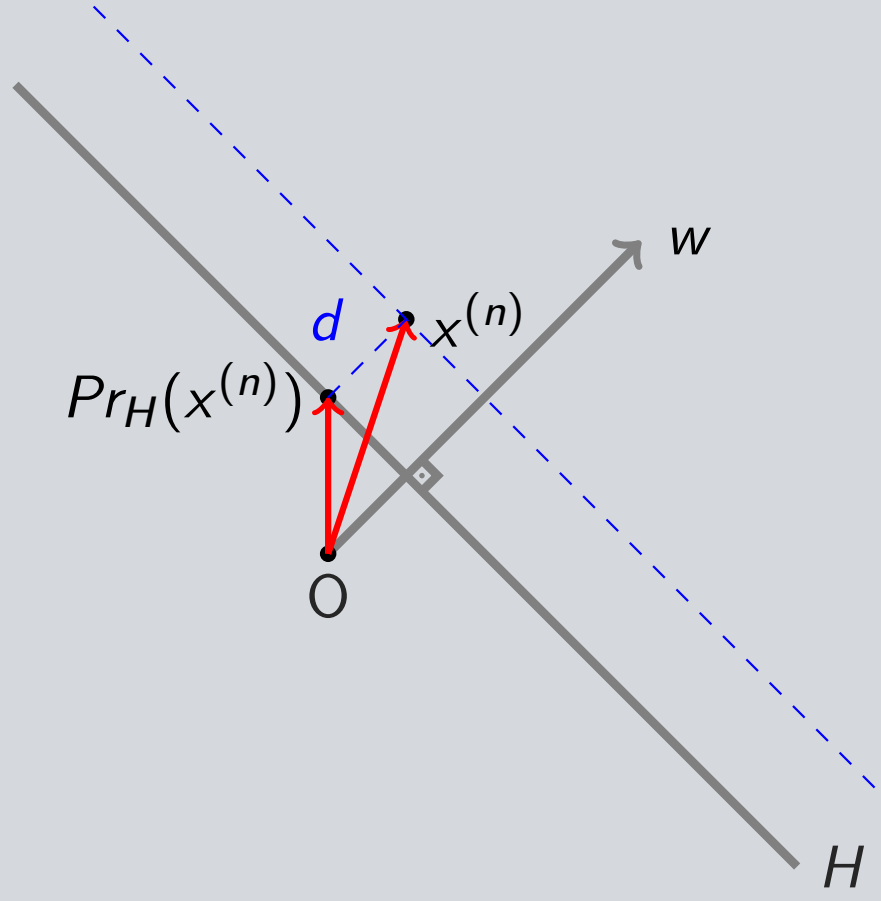


- Margem é a distância do vetor de suporte até o hiperplano!



- Eventualmente o vetor de suporte não é único!

Voltando a questão da margem...



- A distância $x^{(n)} - Pr_H(x^{(n)})$ está na mesma direção de w!
- $x^{(n)} = Pr_H(x^{(n)}) + d\frac{w}{||w||}$.

Dedução do problema - Ridge SVM

Considere $\mathcal{D} = \{(x^{(1)}, y^{(1)}), \cdots, (x^{(N)}, y^{(N)})\}$ a nossa amostra rotulada.

- Assuma que exista $w \in \mathbb{R}^m$ e $b \in \mathbb{R}$ tais que:

$$y^{(k)} \left(\langle w, x \rangle + b \right) > 0,$$

para todo $k \in \{1, \cdots, N\}$.

Na prática, essa hipótese afirma que os nossos dados são (perfeitamente) linearmente separáveis.

Podemos considerar $w \in \mathbb{R}^m$ e $b \in \mathbb{R}$ tais que

$$y^{(k)} \left(\langle w, x \rangle + b \right) \geq 1,$$

para todo $k \in \{1, \cdots, N\}$. Além disso, podemos assumir $\min_{k \in \{1, \cdots N\}} y^{(k)} \left(\langle w, x \rangle + b \right) = 1$.

- De fato, seja $\alpha > 0$ tal que

$$\alpha = \min_{k \in \{1, \cdots N\}} y^{(k)} \left(\langle \tilde{w}, x \rangle + \tilde{b} \right) > 0.$$

Tomando $w = \frac{1}{\alpha} \tilde{w}$ e $b = \frac{1}{\alpha} b$ a afirmação segue.

Podemos considerar $w \in \mathbb{R}^m$ e $b \in \mathbb{R}$ tais que

$$y^{(k)} \left(\langle w, x \rangle + b \right) \geq 1,$$

para todo $k \in \{1, \cdots, N\}$. Além disso, podemos assumir $\min_{k \in \{1, \cdots N\}} y^{(k)} \left(\langle w, x \rangle + b \right) = 1$.

- De fato, seja $\alpha > 0$ tal que

$$\alpha = \min_{k \in \{1, \cdots N\}} y^{(k)} \left(\langle \tilde{w}, x \rangle + \tilde{b} \right) > 0.$$

Tomando $w = \frac{1}{\alpha} \tilde{w}$ e $b = \frac{1}{\alpha} b$ a afirmação segue.

Observe que na verdade ambas as formulações são equivalentes!

Considere então H o hiperplano determinado por w e b, e considere que $x^{(n)}$ seja o vetor de suporte

- Sem perda de generalidade, assuma que $x^{(n)}$ está na parte positiva.

Projetando o vetor $x^{(n)}$ em H, temos $Pr_H(x^{(n)}) \in H$. Logo,

$$0 = \left\langle Pr_H(x^{(n)}), w \right\rangle + b \quad = \quad \left\langle x^{(n)} - d\frac{w}{||w||}, w \right\rangle + b$$

Considere então H o hiperplano determinado por w e b, e considere que $x^{(n)}$ seja o vetor de suporte

- Sem perda de generalidade, assuma que $x^{(n)}$ está na parte positiva.

Projetando o vetor $x^{(n)}$ em H, temos $Pr_H(x^{(n)}) \in H$. Logo,

$$\begin{aligned} 0 = \left\langle Pr_H(x^{(n)}), w \right\rangle + b &= \left\langle x^{(n)} - d\frac{w}{||w||}, w \right\rangle + b \\ &= \left\langle x^{(n)}, w \right\rangle + \left\langle -d\frac{w}{||w||}, w \right\rangle + b \end{aligned}$$

Considere então H o hiperplano determinado por w e b, e considere que $x^{(n)}$ seja o vetor de suporte

- Sem perda de generalidade, assuma que $x^{(n)}$ está na parte positiva.

Projetando o vetor $x^{(n)}$ em H, temos $Pr_H(x^{(n)}) \in H$. Logo,

$$
\begin{aligned}
0 = \left\langle Pr_H(x^{(n)}), w \right\rangle + b &= \left\langle x^{(n)} - d\frac{w}{||w||}, w \right\rangle + b \\
&= \left\langle x^{(n)}, w \right\rangle + \left\langle -d\frac{w}{||w||}, w \right\rangle + b \\
&= \left\langle x^{(n)}, w \right\rangle - \frac{d}{||w||}\langle w, w \rangle + b
\end{aligned}
$$

Considere então H o hiperplano determinado por w e b, e considere que $x^{(n)}$ seja o vetor de suporte

- Sem perda de generalidade, assuma que $x^{(n)}$ está na parte positiva.

Projetando o vetor $x^{(n)}$ em H, temos $Pr_H(x^{(n)}) \in H$. Logo,

$$
\begin{aligned}
0 = \left\langle Pr_H(x^{(n)}), w \right\rangle + b &= \left\langle x^{(n)} - d\frac{w}{||w||}, w \right\rangle + b \\
&= \left\langle x^{(n)}, w \right\rangle + \left\langle -d\frac{w}{||w||}, w \right\rangle + b \\
&= \left\langle x^{(n)}, w \right\rangle - \frac{d}{||w||}\langle w, w \rangle + b \\
&= \left\langle x^{(n)}, w \right\rangle - \frac{d}{||w||}||w||^2 + b
\end{aligned}
$$

Considere então H o hiperplano determinado por w e b, e considere que $x^{(n)}$ seja o vetor de suporte

- Sem perda de generalidade, assuma que $x^{(n)}$ está na parte positiva.

Projetando o vetor $x^{(n)}$ em H, temos $Pr_H(x^{(n)}) \in H$. Logo,

$$
\begin{aligned}
0 = \left\langle Pr_H(x^{(n)}), w \right\rangle + b &= \left\langle x^{(n)} - d\frac{w}{||w||}, w \right\rangle + b \\
&= \left\langle x^{(n)}, w \right\rangle + \left\langle -d\frac{w}{||w||}, w \right\rangle + b \\
&= \left\langle x^{(n)}, w \right\rangle - \frac{d}{||w||}\langle w, w \rangle + b \\
&= \left\langle x^{(n)}, w \right\rangle - \frac{d}{||w||}||w||^2 + b \\
&= \left\langle x^{(n)}, w \right\rangle + b - d||w||
\end{aligned}
$$

Considere então H o hiperplano determinado por w e b, e considere que $x^{(n)}$ seja o vetor de suporte

- Sem perda de generalidade, assuma que $x^{(n)}$ está na parte positiva.

Projetando o vetor $x^{(n)}$ em H, temos $Pr_H(x^{(n)}) \in H$. Logo,

$$
\begin{aligned}
0 = \left\langle Pr_H(x^{(n)}), w \right\rangle + b &= \left\langle x^{(n)} - d\frac{w}{||w||}, w \right\rangle + b \\
&= \left\langle x^{(n)}, w \right\rangle + \left\langle -d\frac{w}{||w||}, w \right\rangle + b \\
&= \left\langle x^{(n)}, w \right\rangle - \frac{d}{||w||}\langle w, w \rangle + b \\
&= \left\langle x^{(n)}, w \right\rangle - \frac{d}{||w||}||w||^2 + b \\
&= \left\langle x^{(n)}, w \right\rangle + b - d||w|| \\
&= 1 - d||w||.
\end{aligned}
$$

Resolvendo para d, segue que $d = \frac{1}{||w||}$ é o comprimento da margem. [1]

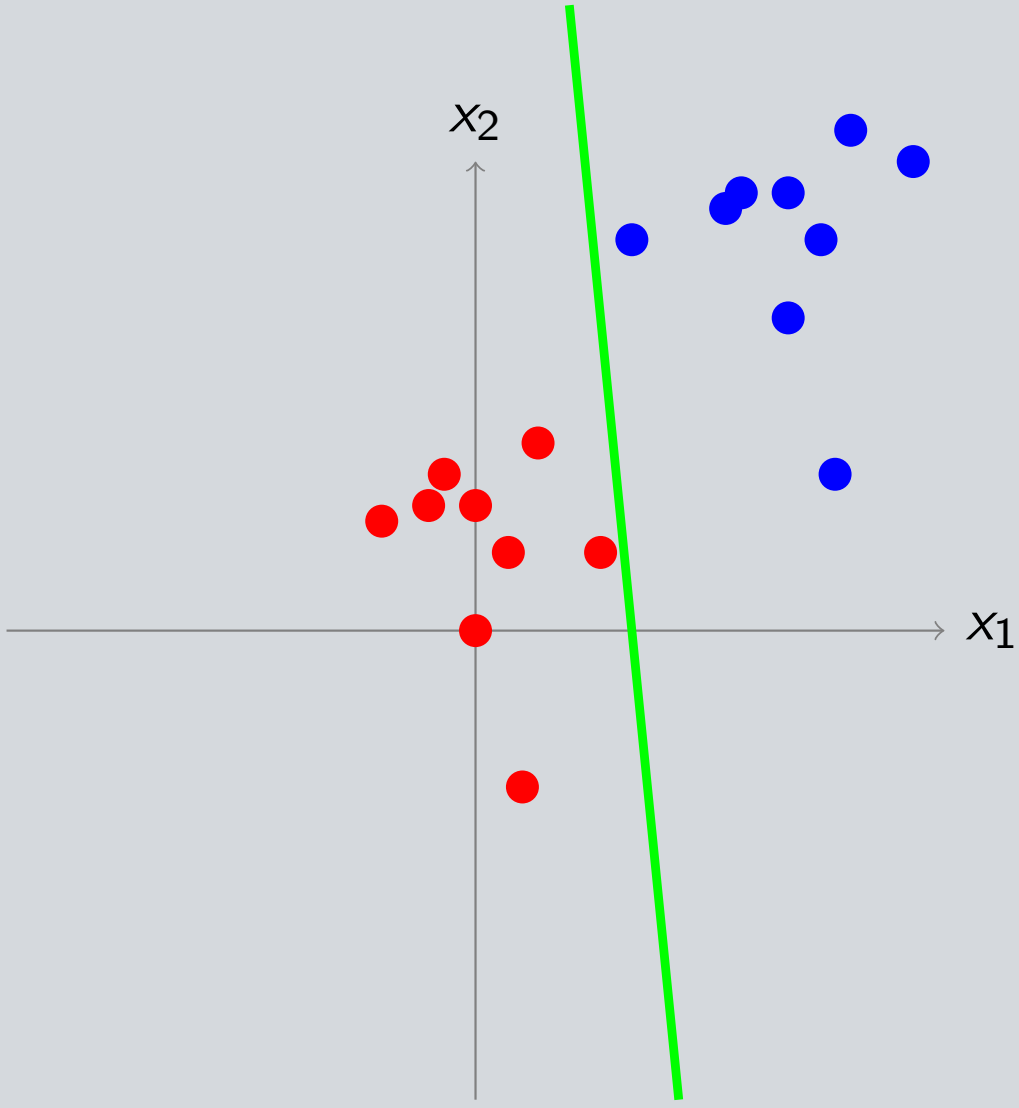
[1] Ou meia margem, dependendo da bibliografia.

Resolvendo para d, segue que $d = \frac{1}{||w||}$ é o comprimento da margem. [1]

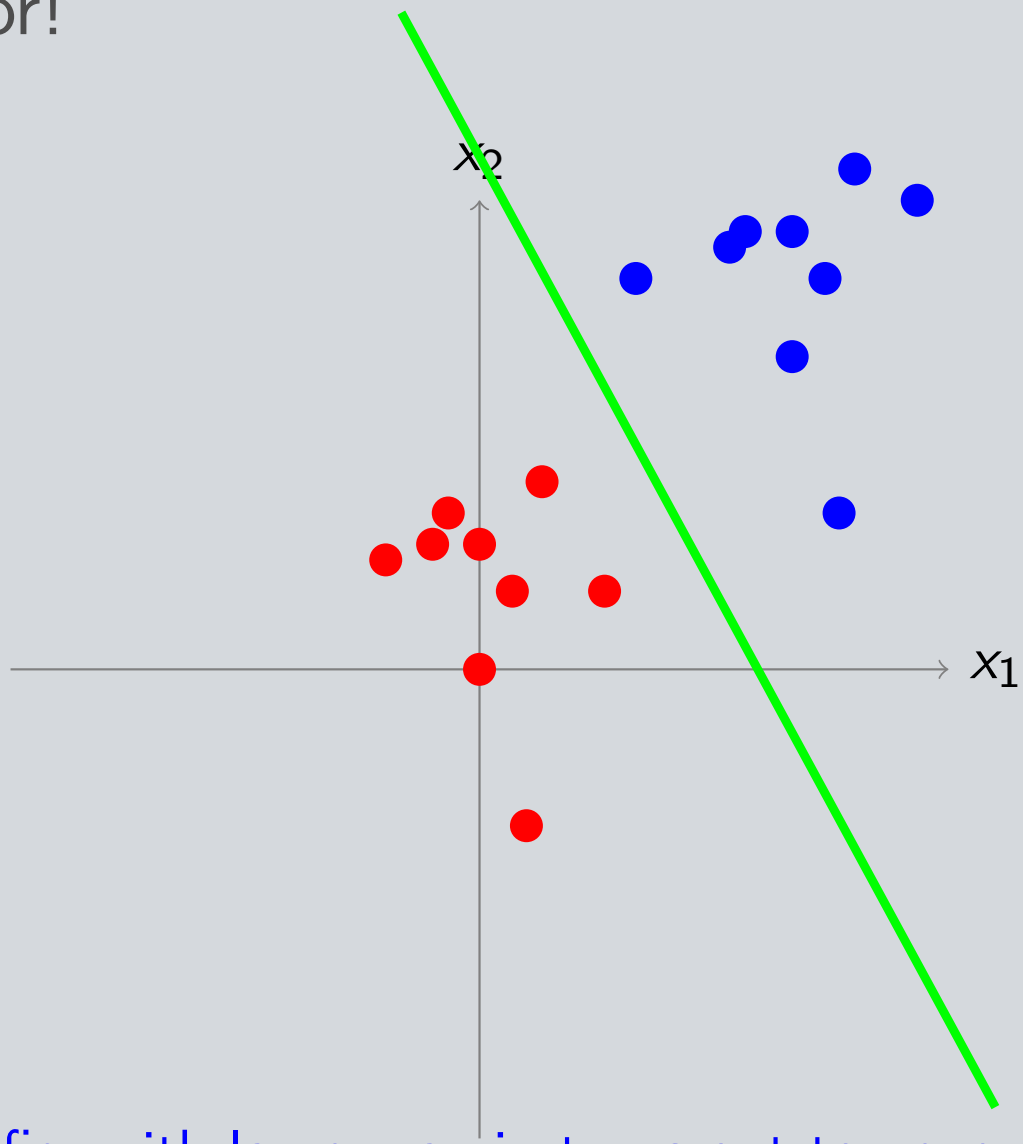
- Queremos sempre uma margem maior possível!
- Por quê?

[1] Ou meia margem, dependendo da bibliografia.

Margem pequena



Margem maior!



"A classifier with large margin turns out to generalize well"
(Steinwart and Christmann, 2008).

Formulação do problema

Então o nosso problema é dado por:

$$\max_{w,b} \frac{1}{||w||}$$

$$\text{sujeito a } y^{(k)}(< w, x^{(k)} > + b) \geq 1, \text{ para } k \in \{1, \cdots, N\}.$$

Formulação do problema

Então o nosso problema é dado por:

$$\max_{w,b} \frac{1}{||w||}$$
$$\text{sujeito a } y^{(k)}(< w, x^{(k)} > + b) \geq 1, \text{ para } k \in \{1, \cdots, N\}.$$

que é equivalente a

$$\min_{w,b} \frac{1}{2}||w||^2$$
$$\text{sujeito a } y^{(k)}(< w, x^{(k)} > + b) \geq 1, \text{ para } k \in \{1, \cdots, N\}.$$

Problema de otimização convexa - forma quadrática com restrições por desigualdades lineares.

Sensibilidade à outliers da margem rígida!

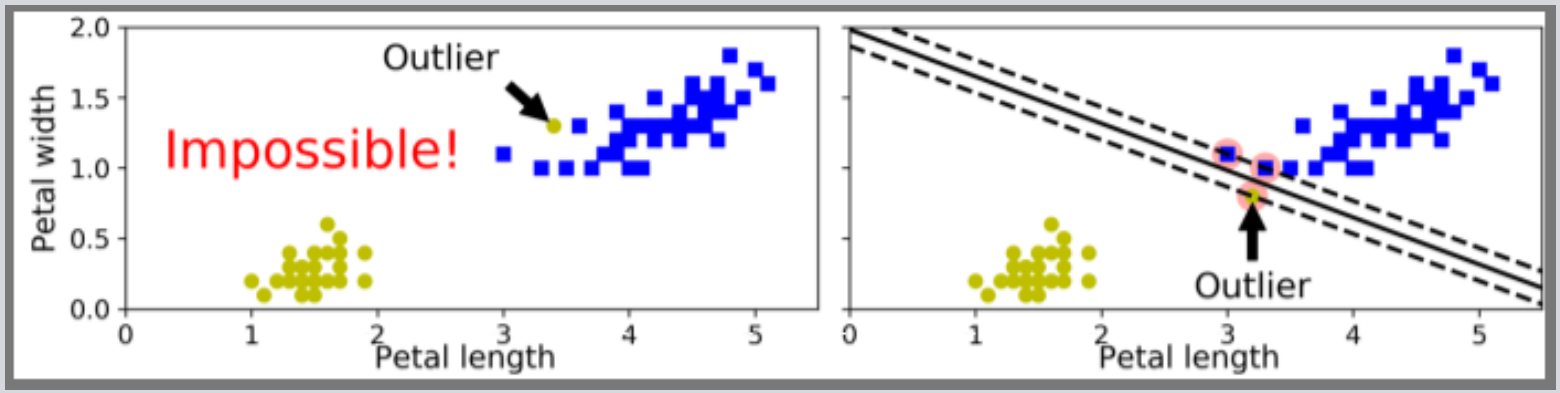


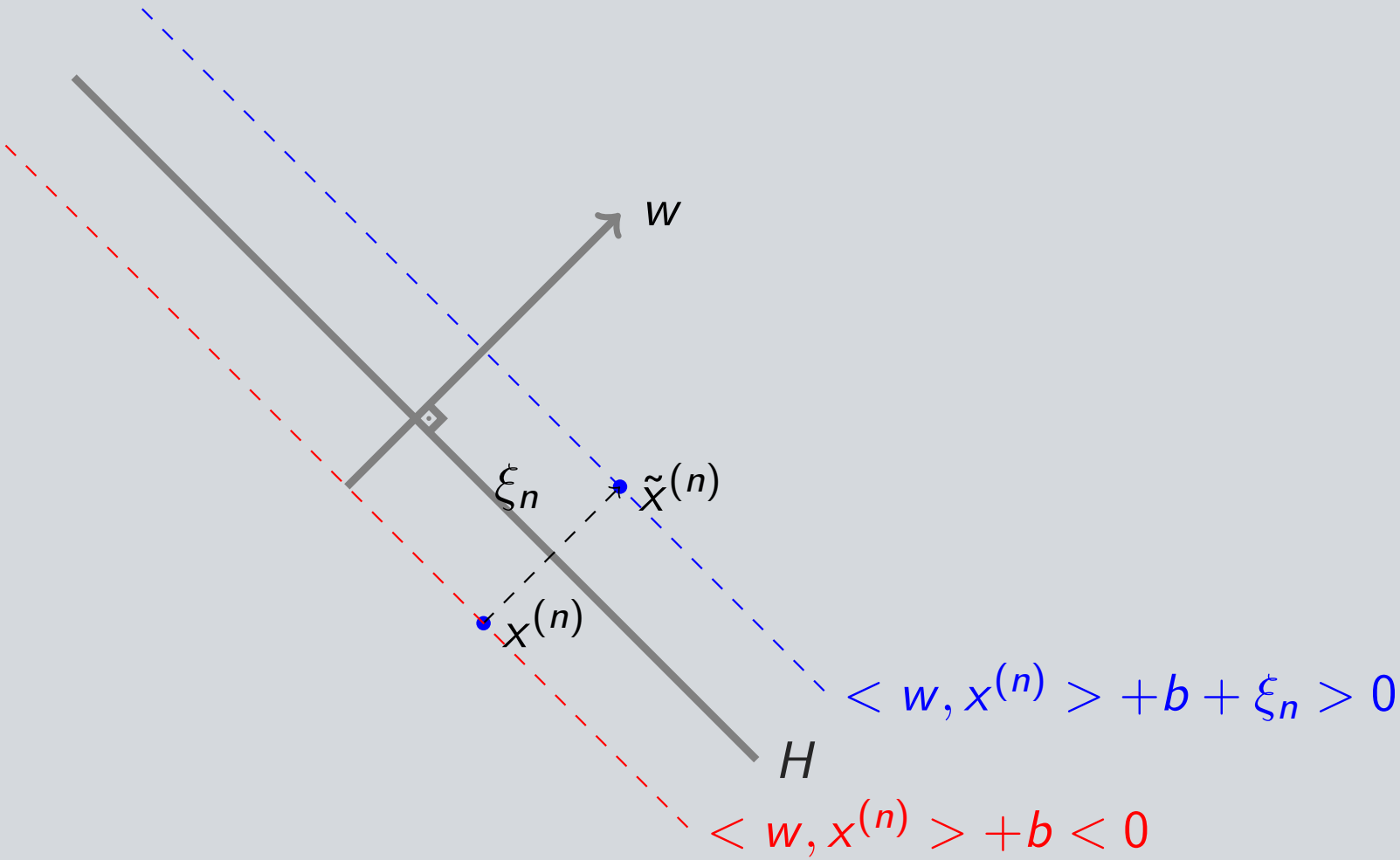
Figura 2: Extraído de Gron, A. Hands-On Machine Learning with Scikit-Learn and TensorFlow: Concepts, Tools, and Techniques to Build Intelligent Systems. Sebastopol: O'Reilly Media, Inc, 2017.

Soft Margin SVM

Quando os nossos dados não são linearmente separáveis, gostaríamos de permitir alguns exemplos ficarem dentro da margem ou até no lado errado do hiperplano.

- O modelo que permite tais propriedades é chamado de Máquina de Vetores de Suporte de Margem Suave.

Intuição



Ideia

Para cada $n \in \{1, \cdots, N\}$ introduzimos variáveis de relaxamento $\xi_n \geq 0$ associado a um par $(x^{(n)}, y^{(n)})$.

Ideia

Para cada $n \in \{1, \cdots, N\}$ introduzimos variáveis de relaxamento $\xi_n \geq 0$ associado a um par $(x^{(n)}, y^{(n)})$.

Fixe H um hiperplano (com paramêtros $w \in \mathbb{R}^n$ e $n \in \mathbb{R}$)

- Suponha que exista $n \in \{1, \cdots, N\}$ de forma que

$$y^{(n)}(< w, x^{(n)} > + b) < 0.$$

Afirmamos que existe $\xi_n > 0$ de modo que:

$$y^{(n)} \left(\left\langle w, x^{(n)} + y^{(n)} \xi_n \frac{w}{||w||^2} \right\rangle + b \right) > 0.$$

Afirmamos que existe $\xi_n \geq 0$ de modo que:

$$y^{(n)} \left(\left\langle w, x^{(n)} + y^{(n)} \xi_n \frac{w}{||w||^2} \right\rangle + b \right) > 0. \tag{1}$$

Afirmamos que existe $\xi_n \geq 0$ de modo que:

$$y^{(n)} \left(\left\langle w, x^{(n)} + y^{(n)} \xi_n \frac{w}{||w||^2} \right\rangle + b \right) > 0. \tag{1}$$

Note que

$$\begin{aligned} & y^{(n)} \left(\left\langle w, x^{(n)} + y^{(n)} \xi_n \frac{w}{||w||^2} \right\rangle + b \right) > 0 \\ \iff & y^{(n)} \left(< w, x^{(n)} > + b \right) + \xi_n > 0 \\ \iff & \xi_n > -y^{(n)} \left(< w, x^{(n)} > + b \right). \end{aligned}$$

Afirmamos que existe $\xi_n \geq 0$ de modo que:

$$y^{(n)}\left(\left\langle w, x^{(n)} + y^{(n)}\xi_n \frac{w}{||w||^2}\right\rangle + b\right) > 0. \tag{1}$$

Note que

$$\begin{aligned} & y^{(n)}\left(\left\langle w, x^{(n)} + y^{(n)}\xi_n \frac{w}{||w||^2}\right\rangle + b\right) > 0 \\ \iff & y^{(n)}\left(< w, x^{(n)} > + b\right) + \xi_n > 0 \\ \iff & \xi_n > -y^{(n)}\left(< w, x^{(n)} > + b\right). \end{aligned}$$

Para cada n tal que $y^{(n)}(< w, x^{(n)} > + b) < 0$, escolhendo ξ_n como acima a desigualdade (1) é verdadeira.

SVM Soft Margin

$$\min_{w,b} \frac{1}{2}||w||^2 + C\sum_{n=1}^{N}\xi_n$$

$$\text{sujeito a } y^{(k)}(< w, x^{(k)} > + b) \geq 1 - \xi_k,$$

$$\xi_k \geq 0, \text{ para } k \in \{1, \cdots, N\}.$$

- Classificações erradas ocorrem quando $\xi_k > 1$;
- o problema acima poderia ser reescrito como (cte é n° max de rótulos errados):

$$\min_{w,b} \frac{1}{2}||w||^2$$

$$\text{sujeito a } y^{(k)}(< w, x^{(k)} > + b) \geq 1 - \xi_k,$$

$$\xi_k \geq 0, \sum_{n=1}^{N}\xi_n \leq \text{Cte}, \text{ para } k \in \{1, \cdots, N\}.$$

Equilíbrio:

- Manter a via mais larga, ou seja, minizar $\frac{1}{2}||w||^2$;
- Limitar a violação de margens, ou seja, manter $\sum_{n=1}^{N} \xi_n$ pequeno.

$$\min_{w,b} \frac{1}{2}||w||^2 + C\sum_{n=1}^{N} \xi_n$$

$$\text{sujeito a } y^{(k)}(< w, x^{(k)} > + b) \geq 1 - \xi_k,$$

$$\xi_k \geq 0, \text{ para } k \in \{1, \cdots, N\}.$$

Hiperparâmetro C

Observe que o (hiper)parâmetro C não é minimizado no problema

- No Sklearn, você pode ajustar C;
- Quanto C menor, temos margem mais larga, mas com maiores chances de violação de margem!

$$\min_{w,b} \frac{1}{2}||w||^2 + C\sum_{n=1}^{N} \xi_n$$

$$\text{sujeito a } y^{(k)}(< w, x^{(k)} > + b) \geq 1 - \xi_k,$$

$$\xi_k \geq 0, \text{ para } k \in \{1, \cdots, N\}.$$

Tamanho margem v.s. violação de margem

- Quanto maior C maior a penalização no erro de classificação e portanto teremos uma margem mais estreita;
- Menor C teremos uma menor penalização (...).

$$\min_{w,b} \frac{1}{2}||w||^2 + C\sum_{n=1}^{N}\xi_n$$

$$\text{sujeito a } y^{(k)}(< w, x^{(k)} > + b) \geq 1 - \xi_k,$$

$$\xi_k \geq 0, \text{ para } k \in \{1, \cdots, N\}.$$

violações margem v.s. grandes margens

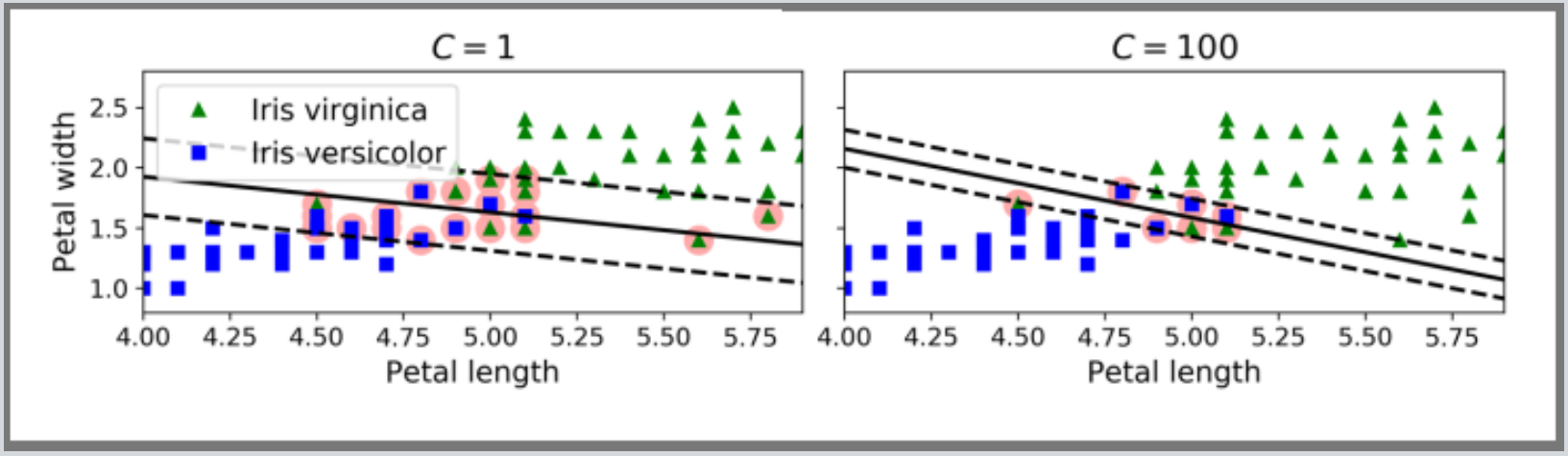


Figura 3: Extraído de Gron, A. Hands-On Machine Learning with Scikit-Learn and TensorFlow: Concepts, Tools, and Techniques to Build Intelligent Systems. Sebastopol: O'Reilly Media, Inc, 2017.

Dicas e comentários

- Como encontrar o melhor C?

Dicas e comentários

- Como encontrar o melhor C? GridSearch!
- Ao contrário da Regressão logística, os classidores SVM não fornecem probabilidade para cada classe;
- Podemos treinar um modelo SVM linear usando a classe LinearSVC com algum C (digamos, $C = 1$) com loss = "hinge".

Hinge-loss

Outra maneira de formular o problema do SVM é minizar a função custo:

$$J(w,b) := \frac{1}{2}||w||^2 + C\sum_{n=1}^{N} \max\left\{0, 1 - y^{(n)}\left(\left\langle w, x^{(n)}\right\rangle + b\right)\right\}.$$

SGDClassifier v.s. LinearSVC

- Uma abordagem é usar o gradiente descendente na classe SGDClassifier do Sklearn:

$$\text{clf} = \textit{SGDClassifier}(\text{loss} = \text{“hinge”}, \text{alpha} = 1/\text{m*C}).$$

- Não converge tão rápido quanto a classe LinearSVC (usa Programação Quadrática);
- Útil para trabalhar com grande conjuntos de dados que não cabem na memória (out-of-core);
- Útil para tarefas de classificação online.

Problema Dual

Outra maneira de formular o problema é o seguinte:

$$\min_{\alpha} \frac{1}{2} \sum_{i=1}^{N} \sum_{j=1}^{N} y^{(i)} y^{(j)} \alpha_i \alpha_j \left\langle x^{(i)}, x^{(j)} \right\rangle - \sum_{n=1}^{N} \alpha_n$$

$$\text{sujeito a } \sum_{i=1}^{N} y^{(i)} \alpha_i = 0$$

$$0 \leq \alpha_k \leq C, \text{ para } k \in \{1, \cdots, N\}.$$

- Por padrão o parâmetro "dual" é "True" na classe LinerSVC;
- Devemos usar a formulação dual quando tivermos mais características do que instâncias, do contrário devemos usar dual=False!

SVM não linear

Embora os classificadores lineares SVM sejam eficientes e funcionem surpreendentemente bem em muitos casos, muitos conjuntos de dados estão longe de serem linearmente separáveis!

Ideia: Adicionar características não lineares!

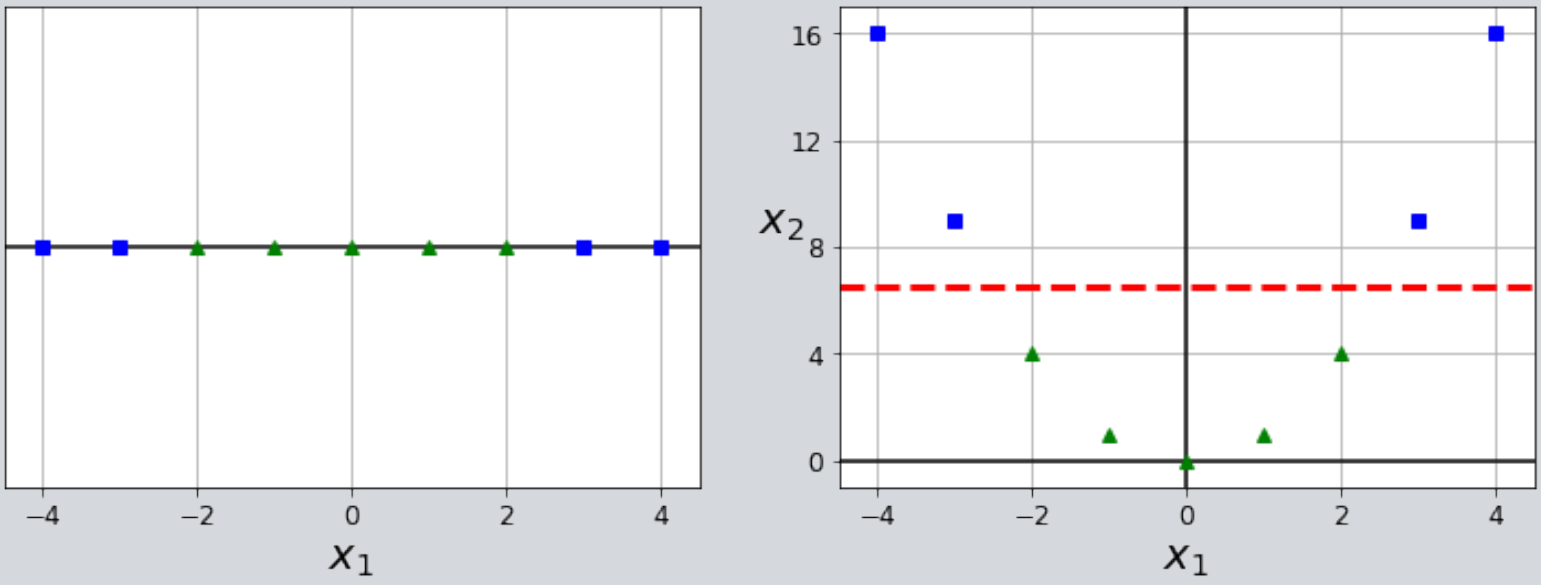


Figura 4: Extraído de Gron, A. Hands-On Machine Learning with Scikit-Learn and TensorFlow: Concepts, Tools, and Techniques to Build Intelligent Systems. Sebastopol: O'Reilly Media, Inc, 2017.

Dados com características polinomiais

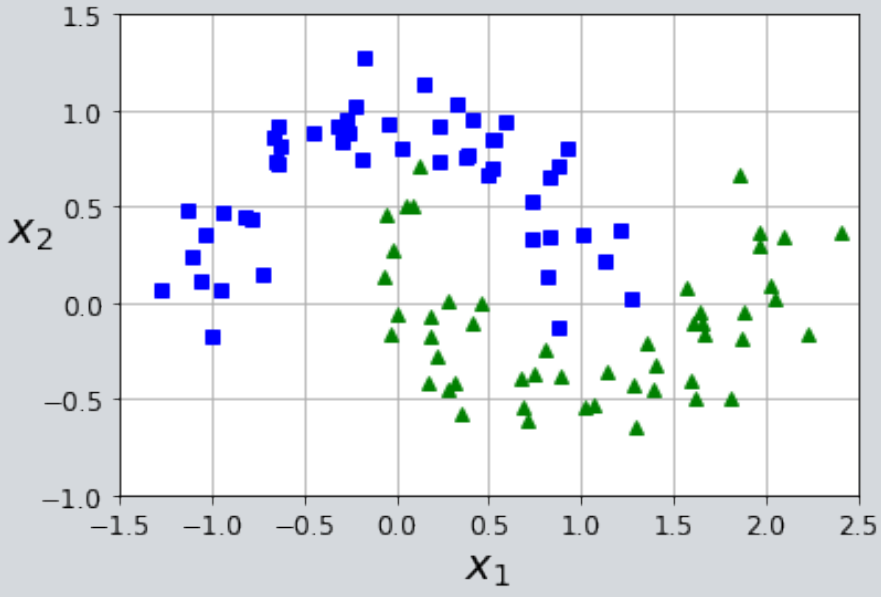


Figura 5: Extraído de Gron, A. Hands-On Machine Learning with Scikit-Learn and TensorFlow: Concepts, Tools, and Techniques to Build Intelligent Systems. Sebastopol: O'Reilly Media, Inc, 2017.

Características polinomiais

Prós:

- Simples de implementar;
- Pode funcionar bem em qualquer outro tipo de algoritmo de ML.

Contras:

- Polinômios de grau baixo podem não captar padrões complexos;
- Polinômios de grau muito alto podem tornar o algoritmo lento!

Núcleos

Técnica “milagrosa”, chamada de truque do núcleo (kernal trick)

Núcleos

Técnica “milagrosa”, chamada de truque do núcleo (kernal trick)

- É possível obter o mesmo resultado de adicionar características polinômiais, mas sem acrescentá-las de fato;
- Evitamos a explosão combinatória de características!
- No Sklearn o hiperparâmetro “coef0” controla o quanto o modelo é influenciado por polinômios de alto grau versus polinômios de baixo grau.

Truqe do núcleo

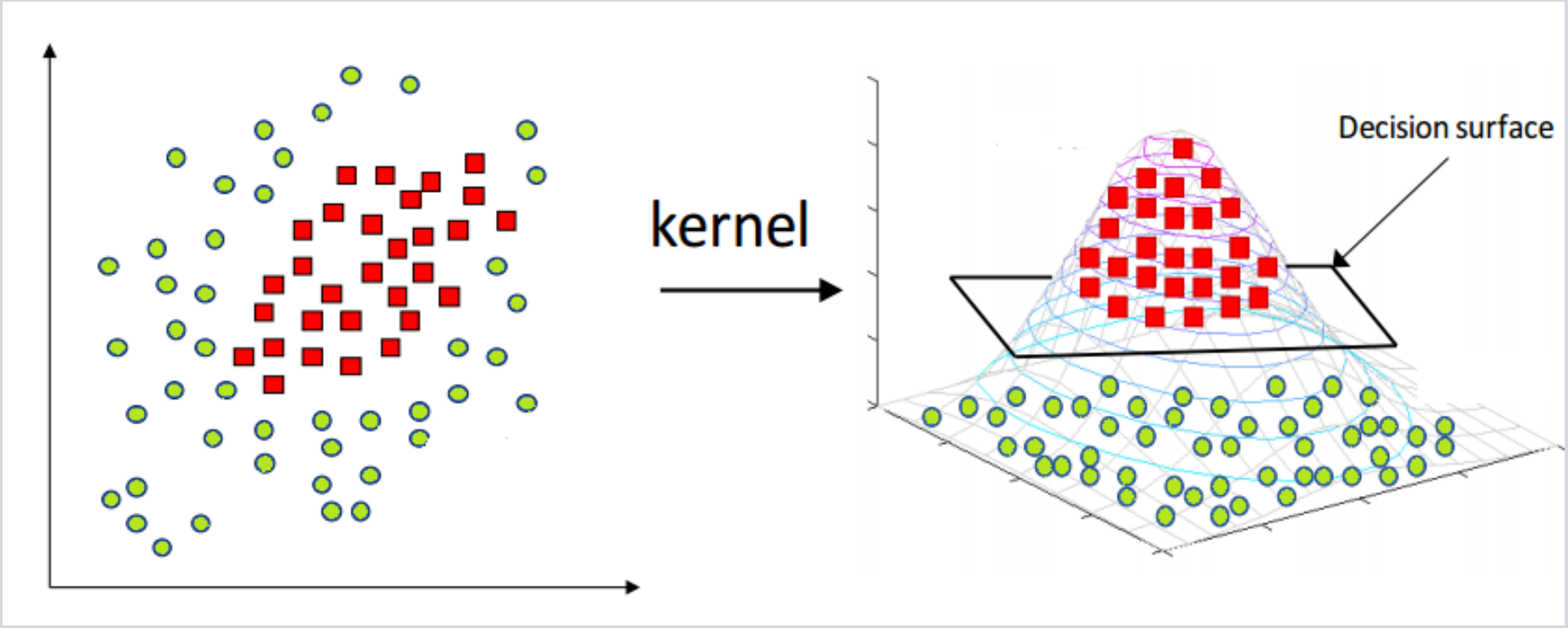


Figura 6: Extraído do blog Medium, autoria: Analytics Vidhya.

Exemplos

- Linear: $K(x, y) =< x, y >$;
- Polinômio de grau d: $K(x, y) = (k_1 + k_2 < x, y >)^d$;
- Base radial: $K(x, y) = exp\left(-\gamma||x - y||^2\right)$;
- Rede neural(sigmóide): $K(x, y) = tanh(k_1 < x, y > +k_2)$;

SVM Kernalizado

Suponha que precisamos aplicar uma transformação polinômial de grau 2 para um conjunto de treino bidimensional (moon dataset por ex)

- Depois queremos usar o SVM no conjunto transformado;
- Considere a função:

$$\phi(x) = \phi(x_1, x_2) = (x_1^2, \sqrt{2}x_1x_2, x_2^2);$$

- Note que os dados agora são 3d ao invés de 2d.

Considere agora a e b dois vetores no $\mathbb{R}^2$, vejamos o que acontece:

$$< \phi(a), \phi(b) > \quad = \quad (a_1^2, \sqrt{2}a_1 a_2, a_2^2)^T (b_1^2, \sqrt{2}b_1 b_2, b_2^2)$$

Considere agora a e b dois vetores no $\mathbb{R}^2$, vejamos o que acontece:

$$\begin{aligned} < \phi(a), \phi(b) > &= (a_1^2, \sqrt{2}a_1 a_2, a_2^2)^T (b_1^2, \sqrt{2}b_1 b_2, b_2^2) \\ &= a_1^2 b_1^2 + 2a_1 b_1 a_2 b_2 + a_2^2 b_2^2 \end{aligned}$$

Considere agora a e b dois vetores no $\mathbb{R}^2$, vejamos o que acontece:

$$\begin{aligned} < \phi(a), \phi(b) > &= (a_1^2, \sqrt{2}a_1 a_2, a_2^2)^T (b_1^2, \sqrt{2}b_1 b_2, b_2^2) \\ &= a_1^2 b_1^2 + 2a_1 b_1 a_2 b_2 + a_2^2 b_2^2 \\ &= (a_1 b_1 + a_2 b_2)^2 \end{aligned}$$

Considere agora a e b dois vetores no $\mathbb{R}^2$, vejamos o que acontece:

$$\begin{aligned} < \phi(a), \phi(b) > &= (a_1^2, \sqrt{2}a_1 a_2, a_2^2)^T (b_1^2, \sqrt{2}b_1 b_2, b_2^2) \\ &= a_1^2 b_1^2 + 2a_1 b_1 a_2 b_2 + a_2^2 b_2^2 \\ &= (a_1 b_1 + a_2 b_2)^2 \\ &= < a, b >^2 \end{aligned}$$

Voltando ao problema Dual

$$\min_{\alpha} \frac{1}{2} \sum_{i=1}^{N} \sum_{j=1}^{N} y^{(i)} y^{(j)} \alpha_i \alpha_j \left\langle x^{(i)}, x^{(j)} \right\rangle - \sum_{n=1}^{N} \alpha_n$$

$$\text{sujeito a } \sum_{i=1}^{N} y^{(i)} \alpha_i = 0$$

$$0 \leq \alpha_k \leq C, \text{ para } k \in \{1, \cdots, N\}.$$

Aplicando a transformação do núcleo nas instância $x^{(i)}$, temos então o novo problema:

Aplicando a transformação do núcleo nas instância $x^{(i)}$, temos então o novo problema:

$$\min_{\alpha} \frac{1}{2} \sum_{i=1}^{N} \sum_{j=1}^{N} y^{(i)} y^{(j)} \alpha_i \alpha_j \left\langle \phi(x^{(i)}), \phi(x^{(j)}) \right\rangle - \sum_{n=1}^{N} \alpha_n$$

$$\text{sujeito a } \sum_{i=1}^{N} y^{(i)} \alpha_i = 0$$

$$0 \leq \alpha_k \leq C, \text{ para } k \in \{1, \cdots, N\}.$$

Mas $< \phi(x^{(i)}), \phi(x^{(j)} >=< x^{(i)}, x^{(j)} >^2$!

Finalmente

$$\min_{\alpha} \frac{1}{2} \sum_{i=1}^{N} \sum_{j=1}^{N} y^{(i)} y^{(j)} \alpha_i \alpha_j \left\langle x^{(i)}, x^{(j)} \right\rangle^2 - \sum_{n=1}^{N} \alpha_n$$

$$\text{sujeito a } \sum_{i=1}^{N} y^{(i)} \alpha_i = 0$$

$$0 \leq \alpha_k \leq C, \text{ para } k \in \{1, \cdots, N\}.$$

Truque: No final das contas, você não precisou na verdade transformar o conjunto de treino, apenas trocou o produto interno pelo quadrado!

Teorema de Mercer

Se $K(\cdot, \cdot)$ for continua, simétria e não negativa, existe ϕ (de dimensão bem alta, possivelmente infinita) tal que:

$$K(a, b) =< \phi(a), \phi(b) >$$

Teorema de Mercer

Se $K(\cdot, \cdot)$ for continua, simétria e não negativa, existe ϕ (de dimensão bem alta, possivelmente infinita) tal que:

$$K(a, b) =< \phi(a), \phi(b) >$$

- Você pode usar o truqe do Kernel pois sabe que ϕ existe, mesmo não sabendo ϕ;
- No caso do núcleo Gaussiano RBF, é possível mostrar que ϕ leva o conjunto de treino em um espaço de dimensão infinita!
- Alguns núcleos não satisfazem as condições do Teorema de Mercer, mas funcionam bem na prática!

Na prática...

$$\min_{\alpha} \frac{1}{2} \sum_{i=1}^{N} \sum_{j=1}^{N} y^{(i)} y^{(j)} \alpha_i \alpha_j K(x^{(i)}, x^{(j)}) - \sum_{n=1}^{N} \alpha_n$$

$$\text{sujeito a } \sum_{i=1}^{N} y^{(i)} \alpha_i = 0$$

$$0 \leq \alpha_k \leq C, \text{ para } k \in \{1, \cdots, N\}.$$

Revisão da aula

1. Ideia fundamental do SVM

Revisão da aula

1. Ideia fundamental do SVM
   - Adequar a "margem" mais larga possível entre as classes;

Revisão da aula

1. Ideia fundamental do SVM
   - Adequar a "margem" mais larga possível entre as classes;
   - Maior margem possível entre a fronteira de decisão, que separa as duas classes, e as instâncias de treinamento;

Revisão da aula

1. Ideia fundamental do SVM

- Adequar a "margem" mais larga possível entre as classes;
- Maior margem possível entre a fronteira de decisão, que separa as duas classes, e as instâncias de treinamento;
- Ao realizar a classificação de margem suave, a SVM procura um compromisso entre a separação perfeita das duas classes e a existência da margem mais ampla possível;

Revisão da aula

1. Ideia fundamental do SVM

- Adequar a "margem" mais larga possível entre as classes;
- Maior margem possível entre a fronteira de decisão, que separa as duas classes, e as instâncias de treinamento;
- Ao realizar a classificação de margem suave, a SVM procura um compromisso entre a separação perfeita das duas classes e a existência da margem mais ampla possível;
- Pode-se usar o truque do Kernel.

2. Vetor de suporte

2. Vetor de suporte

- Um vetor de suporte é qualquer instância localizada na "margem" após o treinamento de uma SVM (veja a resposta anterior), incluindo sua borda;

2. Vetor de suporte

- Um vetor de suporte é qualquer instância localizada na "margem" após o treinamento de uma SVM (veja a resposta anterior), incluindo sua borda;
- A fronteira de decisão é inteiramente determinada pelos vetores de suporte;

2. Vetor de suporte

- Um vetor de suporte é qualquer instância localizada na "margem" após o treinamento de uma SVM (veja a resposta anterior), incluindo sua borda;
- A fronteira de decisão é inteiramente determinada pelos vetores de suporte;
- Qualquer instância que não seja um vetor de suporte (isto é, fora da margem) não terá influência alguma; você pode removê-los, adicionar mais instâncias ou movê-los e, desde que permaneçam fora da margem, eles não afetarão a fronteira de decisão;

2. Vetor de suporte

- Um vetor de suporte é qualquer instância localizada na "margem" após o treinamento de uma SVM (veja a resposta anterior), incluindo sua borda;
- A fronteira de decisão é inteiramente determinada pelos vetores de suporte;
- Qualquer instância que não seja um vetor de suporte (isto é, fora da margem) não terá influência alguma; você pode removê-los, adicionar mais instâncias ou movê-los e, desde que permaneçam fora da margem, eles não afetarão a fronteira de decisão;
- O cálculo das previsões envolve apenas os vetores de suporte, não todo o conjunto de treinamento.

3. Por que é importante dimensionar as entradas ao utilizar SVM?

3. Por que é importante dimensionar as entradas ao utilizar SVM?

- As SVM tentam ajustar a maior "margem" possível entre as classes (veja a primeira resposta), portanto, se o conjunto de treinamento não for escalonado, a SVM tenderá a negligenciar pequenas características;

3. Por que é importante dimensionar as entradas ao utilizar SVM?

- As SVM tentam ajustar a maior "margem" possível entre as classes (veja a primeira resposta), portanto, se o conjunto de treinamento não for escalonado, a SVM tenderá a negligenciar pequenas características;
- Lembrar da aula de Feature Scaling!

4. Um classificador SVM pode produzir uma pontuação de confiança quando classifica uma instância? E quanto a uma probabilidade?

4. Um classificador SVM pode produzir uma pontuação de confiança quando classifica uma instância? E quanto a uma probabilidade?

- Um classificador SVM pode gerar a distância entre a distância de teste e o limite de decisão, e você pode utilizar isso como uma pontuação de confiança. Entretanto, essa pontuação não poderá ser convertida diretamente em uma estimativa da probabilidade da classe;

4. Um classificador SVM pode produzir uma pontuação de confiança quando classifica uma instância? E quanto a uma probabilidade?

- Um classificador SVM pode gerar a distância entre a distância de teste e o limite de decisão, e você pode utilizar isso como uma pontuação de confiança. Entretanto, essa pontuação não poderá ser convertida diretamente em uma estimativa da probabilidade da classe;
- Se você configurar probability=True quando criar um SVM no Scikit-Learn, então, depois do treinamento, ele calibrará as probabilidades usando Regressão Logística nas pontuações da SVM (treinadas por uma validação cruzada de cinco dobras nos dados de treinamento), o que adicionará os métodos predict_proba() e predict_log_proba() à SVM.

5. Você deve utilizar a forma primal ou dual do problema SVM no treinamento de um modelo em um conjunto de treinamento com milhões de instâncias e centenas de características?

5. Você deve utilizar a forma primal ou dual do problema SVM no treinamento de um modelo em um conjunto de treinamento com milhões de instâncias e centenas de características?

- Esta questão se aplica apenas às SVM lineares, já que a kernelizada só pode utilizar a forma dual;

5. Você deve utilizar a forma primal ou dual do problema SVM no treinamento de um modelo em um conjunto de treinamento com milhões de instâncias e centenas de características?

- Esta questão se aplica apenas às SVM lineares, já que a kernelizada só pode utilizar a forma dual;
- A complexidade dos cálculos do problema SVM da forma primal é proporcional ao número de instâncias de treinamento m, enquanto a complexidade dos cálculos da forma dual é proporcional a um número entre m^2 e m^2;

5. Você deve utilizar a forma primal ou dual do problema SVM no treinamento de um modelo em um conjunto de treinamento com milhões de instâncias e centenas de características?

- Esta questão se aplica apenas às SVM lineares, já que a kernelizada só pode utilizar a forma dual;
- A complexidade dos cálculos do problema SVM da forma primal é proporcional ao número de instâncias de treinamento m, enquanto a complexidade dos cálculos da forma dual é proporcional a um número entre m^2 e m^2;
- Caso existam milhões de instâncias, você definitivamente deveria utilizar a forma primal, porque a forma dual será muito lenta.

6. Digamos que você treinou um classificador SVM com o kernel RBF. Parece que ele se subajusta ao conjunto de treinamento: você deve aumentar ou diminuir γ? E quanto ao C?

6. Digamos que você treinou um classificador SVM com o kernel RBF. Parece que ele se subajusta ao conjunto de treinamento: você deve aumentar ou diminuir γ? E quanto ao C?

- Caso um classificador SVM treinado com um kernel RBF se subajuste ao conjunto de treinamento, poderá haver muita regularização. Para diminuí-la, você precisará aumentar γ ou C (ou ambos).

Obrigado!

Contato:
edson.junior@ufsc.br

UNIVERSIDADE FEDERAL
DE SANTA CATARINA