یادگیری ماشین

دکتر امیر نجفی

AI CONTEST
rayan

مباحث این جلسه

- ماشین بردار پشتیبان (Support Vector Machine)
- اهمیت حاشیه (Margin) در طبقه بندی
- Hard Margin SVM
- Soft Margin SVM
- مفهوم Kernel و SVM های غیر خطی

داده ، ویژگی و
فضای طبقه بندی

مفهوم داده و فضای ویژگی (Feature Space)

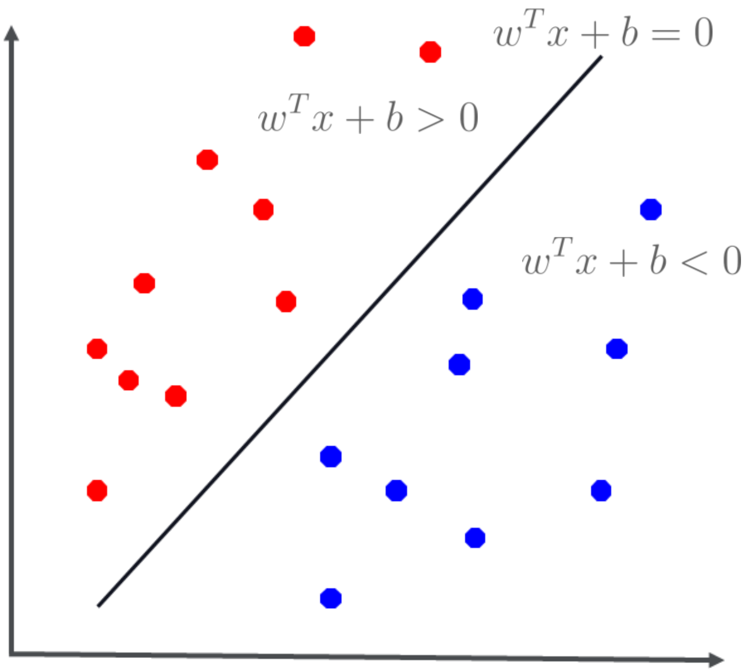
- هر یک از داده‌های آموزش (اعم از تصویر، صوت و ...) را می‌توان به صورت یک نقطه در یک فضای معمولاً بعد بالا در نظر گرفت

- این فضا را معمولاً فضای ویژگی (feature space) می‌نامیم

مفهوم داده و فضای ویژگی (Feature Space)

- برای مثال، ابعاد مختلف فضای ویژگی برای دسته‌ای از تصاویر می‌توانند مقادیر پیکسل‌های تصاویر باشند
- معمولا از نمایش‌های مناسب‌تر (و فشرده‌تری) برای نمایش داده‌ها استفاده می‌شود (مهندسی ویژگی یا Feature Engineering)

طبقه بندی (Classification)

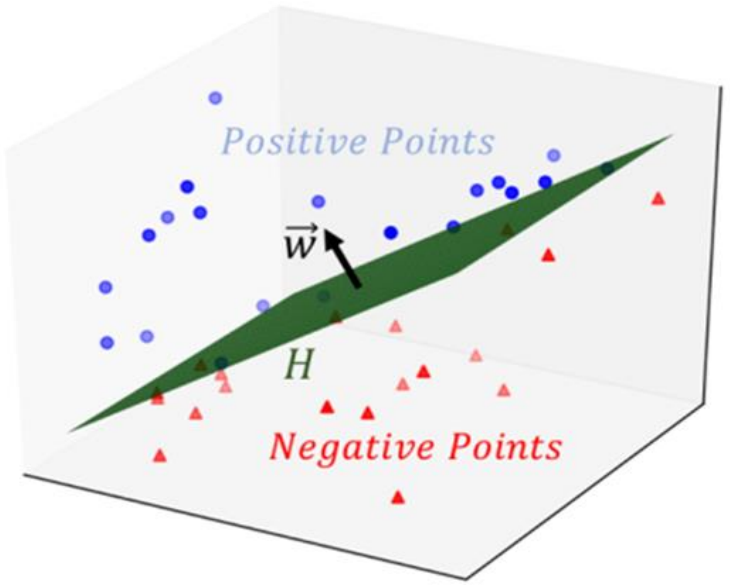


- **تعریف** : طبقه‌بندی، به یافتن طریقی برای تقسیم فضای ویژگی به دو یا چند ناحیه اطلاق می‌گردد. قرار است هر ناحیه نماینده یکی از کلاس ها باشد

روش : این کار می‌تواند به طرق فراوانی انجام پذیرد. استفاده از طبقه‌بندی‌های خطی یکی از اولین روش‌ها بوده است که هنوز هم پرطرفدار است!

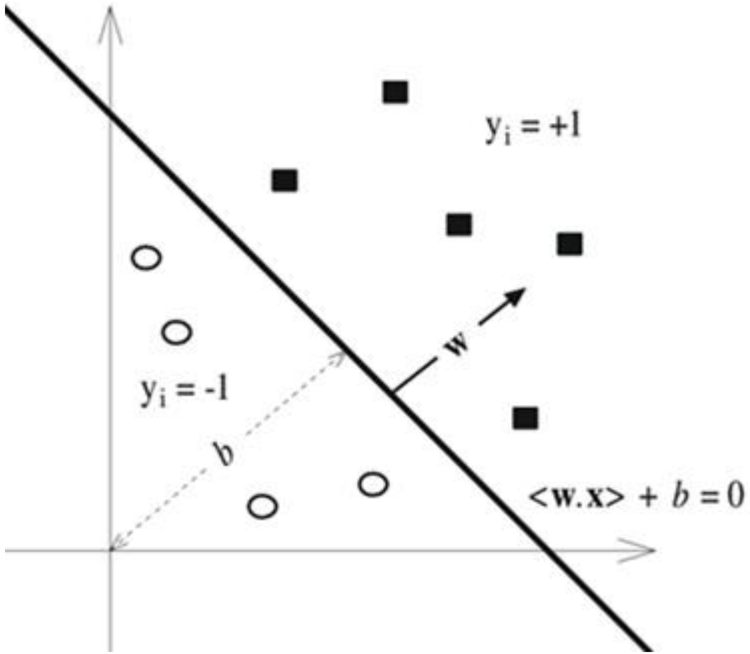
$$y = sign(w^T x + b)$$

طبقه بندی (Classification)



در اینجا مقصود از w بردار عمود بر ابرصفحه جداکننده است. b نیز نشان‌دهنده مقدار بایاس صفحه می‌باشد

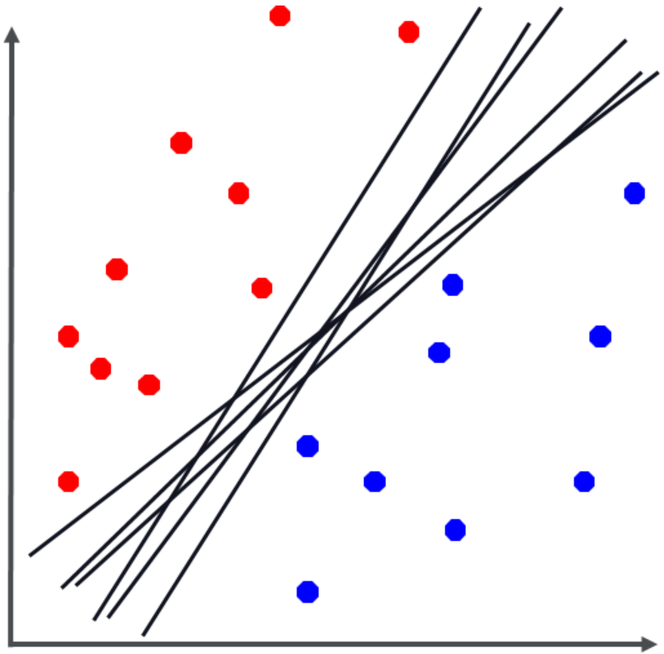
طبقه بندی (Classification)



در حالت ساده دوبعدی، این کار مشابه با جدا کردن صفحه مختصات به واسطه یک خط ساده است

$$w^T x + b = w_1 x_1 + w_2 x_2 + b$$

طبقه بندی (Classification)

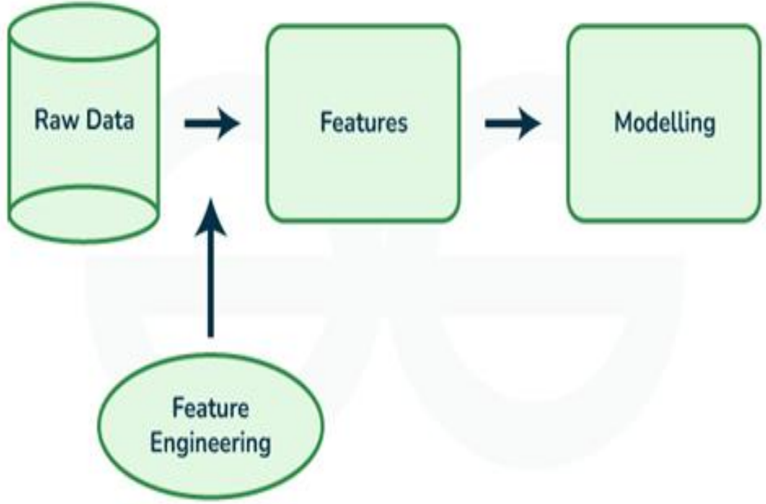


آیا طبقه بندی خطی ممکن یا یکتا است؟

- ممکن است جداسازی خطی اصلا امکان پذیر نباشد!
(به این مورد دوباره برمی‌گردیم)

- اما یک مسئله دیگر نیز اینجاست که در برخی موارد بیش از یک خط می‌توانند مسئله را به شکل ایده‌آل حل کنند (**حدس می‌زنید چرا؟**)

مهندسی ویژگی (Feature Engineering)



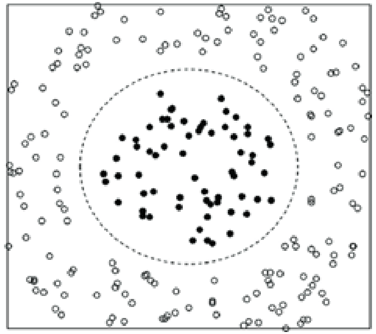
- **تعریف** : مهندسی ویژگی به مجموعه تکنیک‌های ریاضی و شهودی برای استخراج اعداد معنادار از داده‌ها (تصاویر، صوت و ...) اطلاق می‌گردد

- **مثال** : کمی کردن جنس بافت‌های بصری، شدت تغییرات یا وجود لبه در تصاویر، تند/کندی نرخ بیان کلمات در صوت و امثالهم همگی «ویژگی‌های مفید» هستند

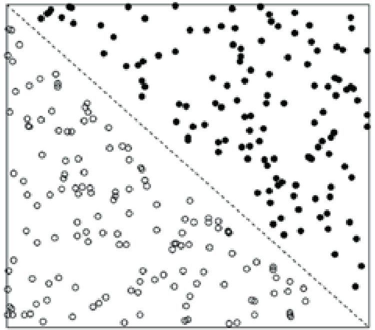
مهندسی ویژگی (Feature Engineering)

- می توان به جای اطلاعات خام داده ها، از ویژگی های "مفید و جدا کننده" آنان استفاده نمود

- در صورتی که از ویژگی های "خوب" استفاده کنیم، داده های کلاس های متفاوت فضای ویژگی نسبت به هم حاشیه پیدا می کنند



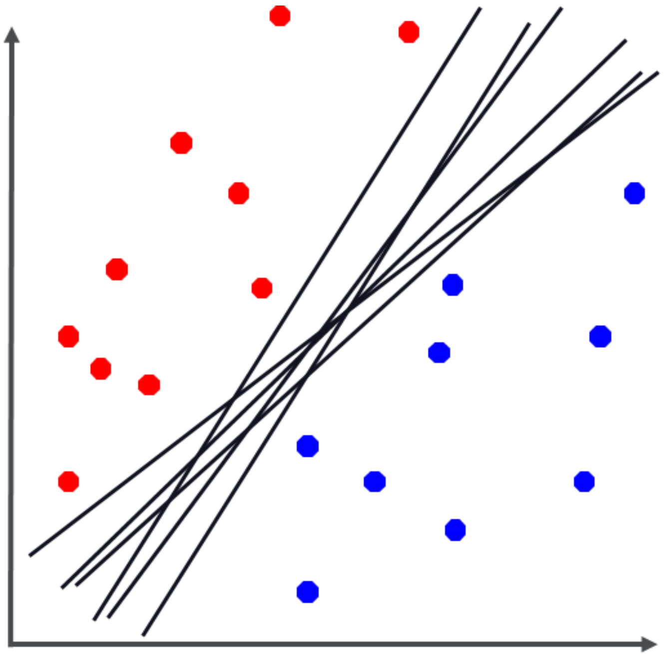
قبل از مهندسی ویژگی



بعد از مهندسی ویژگی

Support Vector Machine

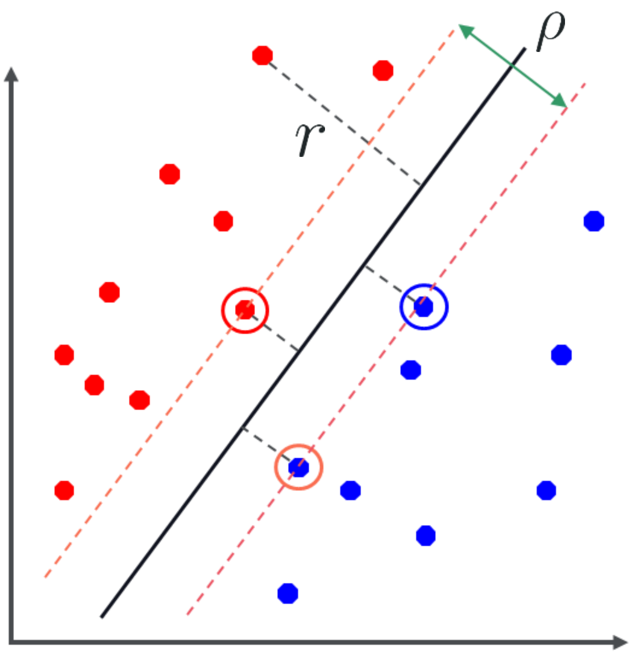
طبقه بندی (Classification)



طبقه بندی "خوب" کدام یکی است؟

- حال فرض کنید که طبقه بندی خطی داده های دو کلاس از اساس ممکن باشد. یعنی داده ها به صورت خطی قابل جداسازی باشند
- در این صورت احتمالا بی نهایت گزینه در اختیار خواهیم داشت
- کدام طبقه بند خطی را انتخاب کنیم؟

فرمولاسیون ریاضیاتی "حاشیه"

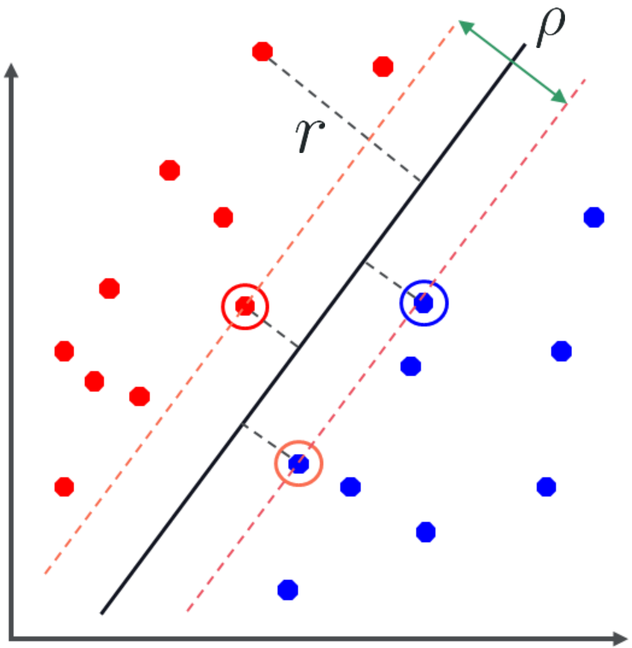


- فاصله هر نقطه x_i تا خط جدا کننده :

$$r = \frac{w^T x_i + b}{||w||}$$

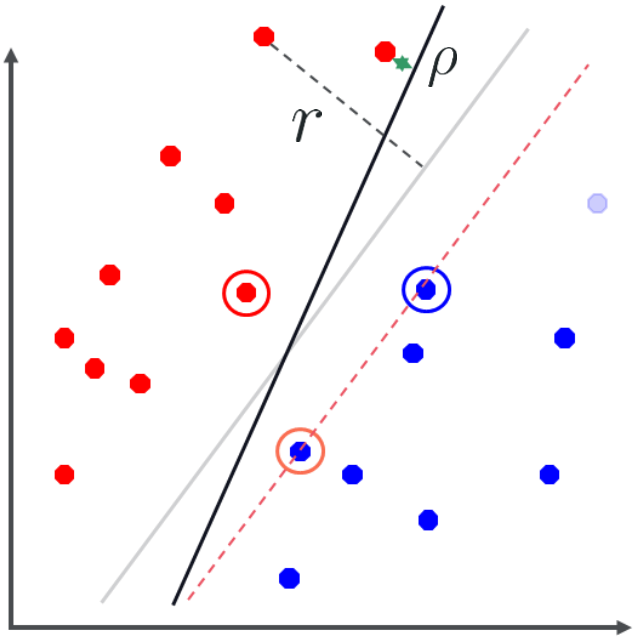
- به نزدیک نمونه ها به ابرصفحه جداکننده بردار های پشتیبان (Support Vector) گفته می شود

فرمولاسیون ریاضیاتی "حاشیه"



- به فاصله بین نقاط پشتیبان از ابر صفحه جدا کننده حاشیه یا Margin ρ اطلاق می گردد
- هدف یافتن صفحه ای است که بیشترین حاشیه را داشته باشد
- آیا تمام نقاط در یافتن صفحه جداکننده با بیشترین حاشیه نقش دارند؟

فرمولاسیون ریاضیاتی "حاشیه"



- شکل مقابل نمونه یک جداکننده با حاشیه بسیار کم است

SVM as a Constrained Optimization Problem

- بدون کاستن از کلیت مسئله، و با فرض جدایی‌پذیر بودن داده‌ها به صورت خطی، می‌توان یک جدا کننده خطی مانند (w, b) را به صورت ابرصفحه‌ای یافت که قیدهای روبرو را ارضا نماید:

- حال لازم است یک مسئله بهینه‌سازی را طوری طراحی کنیم تا آن (w, b) برنده شوند که علاوه بر برقرار نمودن قیود فوق، «حاشیه بیشینه» نیز داشته باشند.

- فرمول حاشیه به صورت روبرو خواهد بود:

$$\begin{cases} w^T x_i + b \leqslant -1 & if \quad y_i = -1 \\ w^T x_i + b \geqslant 1 & if \quad y_i = 1 \end{cases}$$

$$y_i(w^T x_i + b) \geqslant 1 \quad \forall i = 1, \cdots, n$$

$$\rho = \frac{min_i |w^T x_i + b|}{||w||} = \frac{1}{||w||}$$

SVM as a Constrained Optimization Problem

$$\begin{cases} w^T x_i + b \leqslant -1 \quad if \quad y_i = -1 \\ w^T x_i + b \geqslant 1 \quad if \quad y_i = 1 \end{cases}$$

فرض کنید داده های برچسب دار
به صورت (x_i, y_i) نوشته شوند

$$y_i(w^T x_i + b) \geqslant 1 \quad \forall i = 1, \cdots, n$$

قیود به ازای بردار های پشتیبان
یا SV ها "فعال" می شوند!

$$\rho = \frac{min_i |w^T x_i + b|}{||w||} = \frac{1}{||w||}$$

در واقع فاصله یکی از بردارهای
پشتیبان از صفحه جداکننده است

SVM as a Constrained Optimization Problem

$$\min_{\mathbf{w},\mathbf{b}} \frac{1}{2}||\mathbf{w}||_2^2$$

$$s.t. \;\; y_i(\mathbf{w}^T\mathbf{x}_i + \mathbf{b}) \geqslant 1$$

$$\forall i = 1, \cdots, n$$

- بیشینه کردن حاشیه، معادل با کمینه کردن نرم l_2 بردار وزن‌ها، یعنی $||\mathbf{w}||$ یا معادلا $||\mathbf{w}||^2$ است

- در این صورت، به فرم کلی روبرو برای یک «ماشین بردار پشتیبان» یا SVM در حالت جدایی‌پذیری خطی می‌رسیم:

SVM as a Constrained Optimization Problem

$$\min_{\mathbf{w},\mathbf{b}} \frac{1}{2}||\mathbf{w}||_2^2$$

$$s.t. \quad y_i(\mathbf{w}^T\mathbf{x}_i + \mathbf{b}) \geqslant 1$$

$$\forall i = 1, \cdots, n$$

- دقت کنید که مسئله فوق محدب (convex) است. یعنی جزو آن دسته از مسائل بهینه‌سازی مقید قرار می‌گیرد که به صورت عددی، در زمان کم قابل حل می‌باشند
- همچنین دقت کنید که به دلیل فرض جدایی‌پذیری خطی، مسئله فوق حتماً جواب دارد

SVM as a Constrained Optimization Problem

$$\min_{\mathbf{w},\mathbf{b}} \frac{1}{2}||\mathbf{w}||_2^2$$

$$s.t. \;\; y_i(\mathbf{w}^T\mathbf{x}_i + \mathbf{b}) \geqslant 1$$

$$\forall i = 1, \cdots, n$$

- همانطور که پیش‌تر گفته شد مسئله روبرو حل سریع یا به اصطلاح tneicffie دارد
- اما به دلایلی، گاهی اوقات علاقه داریم حالت «دوگان یا dual» آن را حل کنیم. اتفاقاً حالت دوگان (که جواب آن با حالت اصلی یکی است) نیز محدب بوده و به راحتی حل می‌شود

SVM as a Constrained Optimization Problem

نکته جالب در حالت دوگان مسئله SVM این است که مقادیر خود بردارهای ویژگی (از جمله بُعد آنان) اهمیتی ندارند. تنها مقادیر مهمی که نیاز است آنان را بدانیم، ضرب داخلی بردارهای ویژگی در یکدیگر هستند، یعنی:

$$\mathbf{x}_i^T \mathbf{x}_j$$

$$\forall i, j = 1, \cdots, n$$

مباحث تکمیلی SVM

در ادامه دو سناریو تکمیلی برای SVM ها را به اختصار بررسی خواهیم کرد :

سناریو اول : فرض کنید که داده‌ها ذاتا به صورت خطی جدایی‌پذیر نباشند. لذا تعداد کمی از داده‌ها همواره به اشتباه طبقه‌بندی می‌شوند و مسئله بهینه‌سازی چند صفحه پیش اصلا جواب نداشته باشد! هدف این است که اجازه دهیم برخی داده‌ها «حاشیه منفی» پیدا کنند...

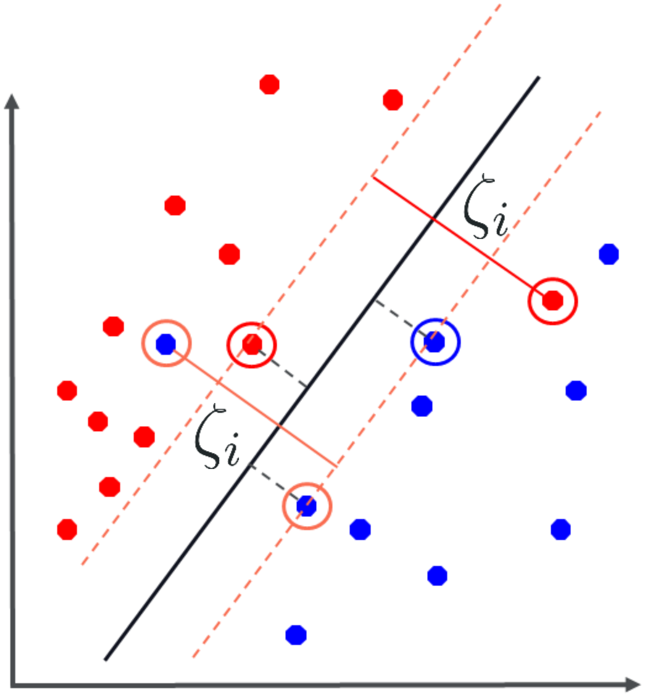
مباحث تکمیلی SVM

در ادامه دو سناریو تکمیلی برای SVM ها را به اختصار بررسی خواهیم کرد :

سناریو دوم : دوباره فرض کنید داده‌ها به صورت خطی جدایی‌پذیر نباشند، اما جداکننده‌های غیرخطی و پیچیده‌تر قابلیت جدا کردن آنان را داشته باشند. قصد داریم یک «نگاشت» یا «تبدیل» به داده‌ها اعمال کنیم تا در فضای جدید به صورت خطی طبقه‌بندی شوند

Soft Margin SVM

Soft Margin SVM

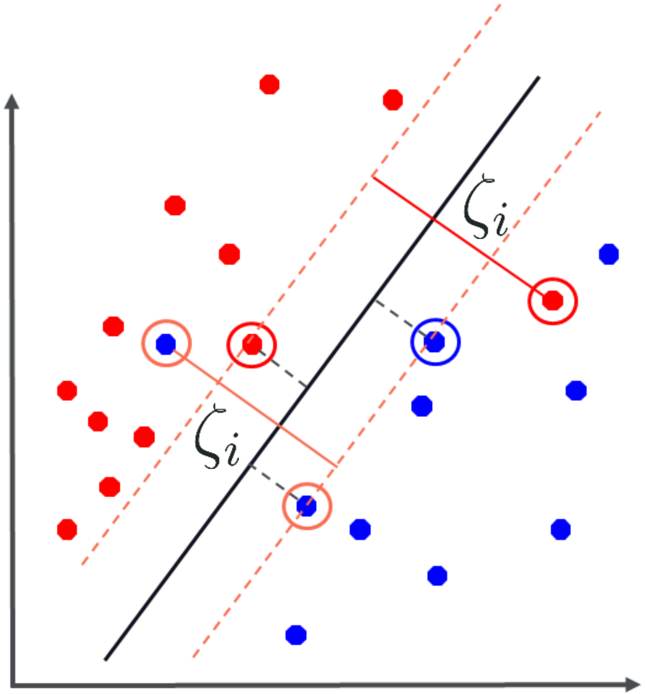


- در صورتیکه تعداد اندکی از داده‌ها به صورت خطی جدایی‌پذیر نباشند، چه باید کرد؟

- می توان تعدادی متغیر لقی یا Slack Variable به مسئله اضافه نمود (معمولا با ζ_i نمایش داده می شوند) که اجازه دهند برای بعضی داده ها حاشیه منفی شود

- اما باید کنترل کرد که ماشین زیاد از آنها استفاده نکند!

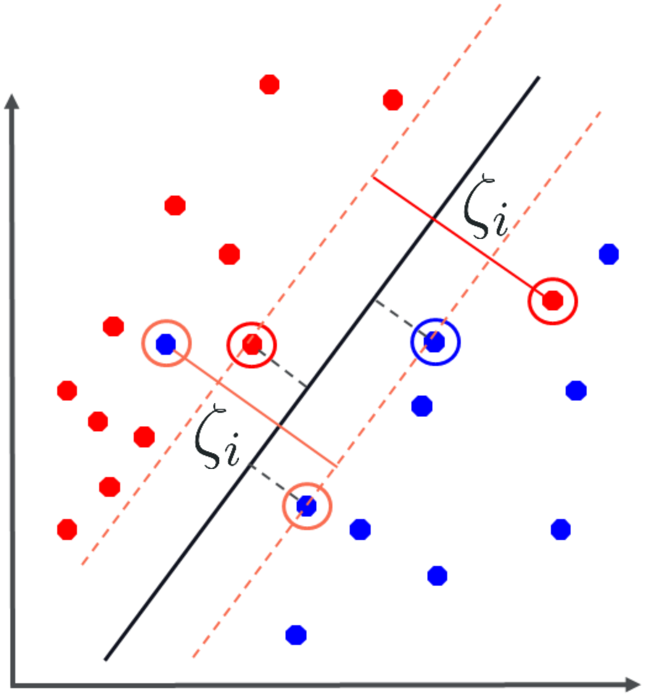
Soft Margin SVM



- شروط مسئله به صورت زیر تغییر می یابند :

$$\begin{cases} \mathbf{w}^T\mathbf{x}_i + \mathbf{b} \leqslant -1 + \zeta_i \quad if \quad y_i = -1 \\ \mathbf{w}^T\mathbf{x}_i + \mathbf{b} \geqslant 1 - \zeta_i \quad if \quad y_i = 1 \end{cases}$$

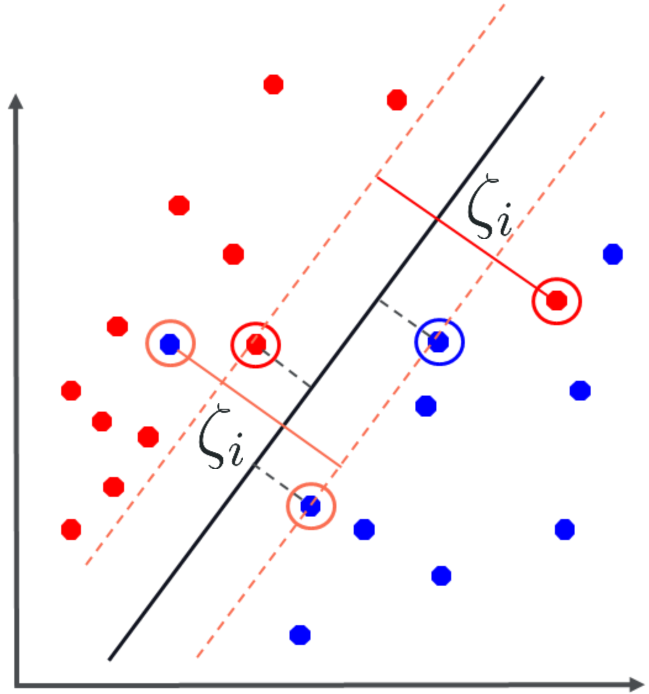
Soft Margin SVM



- در این صورت مسئله به صورت زیر بازنویسی می شود :

$$\begin{aligned} \underset{\mathbf{w},\, b,\, \zeta}{\text{minimize}} \quad & \|\mathbf{w}\|_2^2 + C\sum_{i=1}^{n} \zeta_i \\ \text{subject to} \quad & y_i(\mathbf{w}^\top \mathbf{x}_i - b) \geq 1 - \zeta_i, \quad \zeta_i \geq 0 \end{aligned}$$

Soft Margin SVM

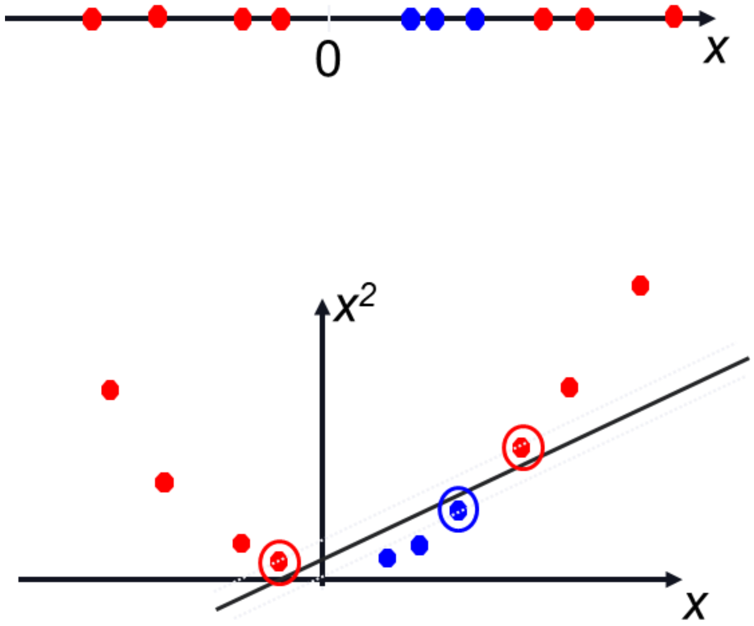


- پارامتر $C > 0$ توسط کاربر و معمولاً طی یک پروسه Cross-Validation تعیین می‌شود

- باید این پارامتر را به گونه‌ای تعیین کرد که مسئله حاشیه قابل‌توجهی داشته باشد، و از طرف overfitting نیز رخ ندهد!

Kernel SVM

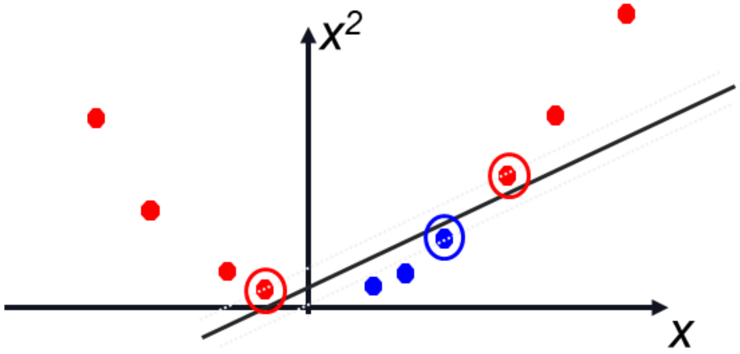
Kernel SVM



- گاهی اوقات داده‌ها به صورت خطی جدایی‌پذیر نیستند. از طرفی، مشکل عدم جدایی‌پذیری نیز ناشی از تعداد کمی داده نویزی نیست، بلکه ویژگی هندسی داده‌هاست.

- در این صورت، می‌توان از طریق یک تبدیل عموماً غیرخطی داده‌ها را به فضایی با بعد بالاتر برد به طوریکه در فضای جدید به صورت خطی جدا شوند

Kernel SVM



- در این مثال تبدیل صورت گرفته به شکل زیر است :

$$x \in \mathbb{R}$$

$$\varphi : \mathbb{R} \longrightarrow \mathbb{R}^2$$

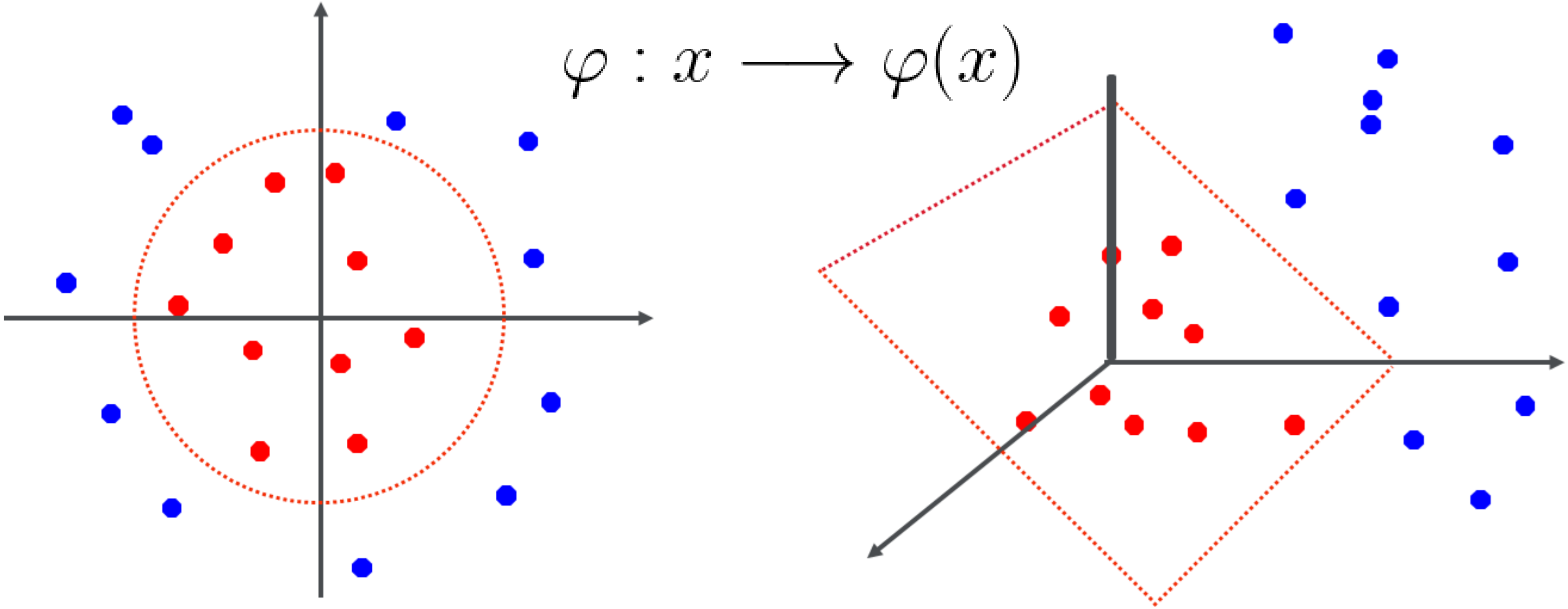
$$\varphi(x) = \begin{bmatrix} x \\ x^2 \end{bmatrix}$$

Kernel SVM

نکته جالب : برای هر مجموعه‌ای از داده‌ها، همواره می‌توان یک تبدیل مخصوص مانند φ یافت که داده‌ها بعد از اعمال نگاشت به صورت خطی جدایی‌پذیر شوند

- در این صورت خطای آموزش (Training Error) می‌تواند همواره صفر شود
- اما این کار لزوماً اثرات مثبتی به همراه ندارد. می‌توانید حدس بزنید چرا؟!

Kernel SVM



$$\varphi : x \longrightarrow \varphi(x)$$

Kernel SVM

- به یاد بیاورید که برای یک SVM، در حالت بهینه‌سازی دوگان، تنها «ضرب داخلی / Inner Product» بردارهای ویژگی $\mathbf{x}_i$ با یکدیگر اهمیت داشت و نه خود آن‌ها!
- همین موضوع در مورد نگاشت‌یافته $\mathbf{x}_i$ ها (به عبارتی $\varphi(\mathbf{x}_i)$ ها) نیز برقرار است. لذا لازم نیست خود مقدار $\varphi(\mathbf{x}_i)$ ها را تعیین کنیم. کافی است مقادیر زیر تعیین شوند :

$$\varphi(\mathbf{x}_i)^T \varphi(\mathbf{x}_j) \quad \forall i, j = 1, \cdots, n$$

Kernel

تعریف : به توابعی که ضرب داخلی تبدیل یافته بردار های ویژگی را تعیین می کنند اصطلاحاً Kernel اطلاق می‌گردد. در صورت تعیین κ دیگر نیازی به تعیین φ نیست (خود به خود تعیین می‌شود)

$$\kappa : \mathbb{R}^d \times \mathbb{R}^d \longrightarrow \mathbb{R}$$

$$\kappa(\mathbf{x}_i, \mathbf{x}_j) = \varphi(\mathbf{x}_i)^T \varphi(\mathbf{x}_j)$$

Kernel

- دقت کنید که هر تابع دلخواه κ نمی‌تواند نماینده یک Kernel واقعی و درست باشد توابعی که واقعاً معرف ضرب داخلی در یک فضایی با بعد بالاتر هستند باید ویژگی‌های مشخصی را داشته باشند

- یک مثال تعاملی با SVM ها در اینجا قابل مشاهده است

کرنل های پر کاربرد

برخی کرنل های درست و پر کاربرد :

Polynomial Kernel:

$$\kappa(\mathbf{x}_i, \mathbf{x}_j) = (1 + \mathbf{x}_i^T \mathbf{x}_j)^p \quad p \in \mathbb{N}_{\geqslant 2}$$

Radial Basis Function (RBF) Kernel:

$$\kappa(\mathbf{x}_i, \mathbf{x}_j) = e^{-\gamma||\mathbf{x}_i - \mathbf{x}_j||^2}$$

Thank You!