Survival Support Vector Machines

Dario Comanducci

***Sommario*—Le Support Vector Machines (SVMs) applicate alla Survival Analysis consentono di superare alcune delle limitazioni del classico modello di Cox sviluppato in Statistica. In questo report verrà descritta la modellazione matematica dei problemi di Survival Analysis come una specializzazione delle *Ranking SVMs*, mostrandone poi l'applicazione su un dataset ad uso didattico.**

I. Introduzione

Gli studi di Survival Analysis (analisi di sopravvivenza) nascono in diversi ambiti: sebbene siano più noti in ambito medico per valutare l'aspettativa di vita, vengono impiegati anche anche in economia (previsione del tempo al fallimento di un'azienda), in meccanica ed elettronica (tempo di rottura di un componente), scienze sociali (stima del tempo dal matrimonio al divorzio). A seconda della domanda dello studio, si è interessati ai gruppi di rischio (quale gruppo di persone/componenti ha maggiori probabilità di sperimentare l'evento?) o alle previsioni temporali (entro quando il motore dovrebbe essere sostituito per ridurre il rischio di guasto?) [18].

Nell'accezione più comune l'analisi di sopravvivenza è un campo della statistica in cui l'obiettivo è modellare *dati di tempo all'evento* (time-to-event data), ossia dati in cui il risultato è il tempo fino al verificarsi di un evento di interesse (*tempo di sopravvivenza*): cercando di stabilire una connessione con le altre variabili dei dati, tale tempo è trattato come variabile dipendente [19].

A. Dati censurati

Una delle maggiori sfide in questo contesto è la presenza di istanze i cui eventi non vengono osservati a causa di limitazioni temporali o per la perdita del tracciamento durante il periodo di osservazione, cioè i dati sono *censurati*.

In generale la censura è classificabile in tre gruppi in base al motivo per cui si è verificata [19]:

- *censura a destra*, in cui il tempo di sopravvivenza osservato è inferiore o uguale al tempo di sopravvivenza reale (dati del paziente assenti dopo un certo periodo di tempo);
- *censura a sinistra*, in cui il tempo di sopravvivenza osservato è superiore o uguale al tempo di sopravvivenza reale (dati del paziente assenti prima di un certo momento);
- *censura a intervalli*, in cui sappiamo solo che l'evento si verifica durante un dato intervallo di tempo.

Si noti che il tempo di occorrenza dell'evento reale è sconosciuto in tutti e tre i casi. In questo report verranno considerati i dati con censura a destra, essendo lo scenario più comune.

In un problema di sopravvivenza il tempo all'evento di interesse T è noto con precisione solo per quei casi in cui l'evento si verifica durante il periodo di studio; per i casi rimanenti, poiché potremmo perderne traccia durante il tempo di osservazione o il loro tempo all'evento è maggiore del tempo di osservazione, possiamo avere solo il tempo censurato C, che può essere il momento del ritiro, della perdita o della fine dell'osservazione: per ogni istanza i possiamo osservare quindi solo il tempo di sopravvivenza (T_i) o il tempo censurato (C_i) ma non entrambi. Se e solo se il tempo $y_i = \min(T_i, C_i)$ può essere osservato durante lo studio, si dice che il *dataset è censurato a destra*. In un problema di analisi di sopravvivenza con casi censurati a destra, anche il tempo di censura è una variabile aleatoria poiché i soggetti entrano nello studio in modo casuale e anche il momento del ritiro o della perdita del tracciamento è casuale [19].

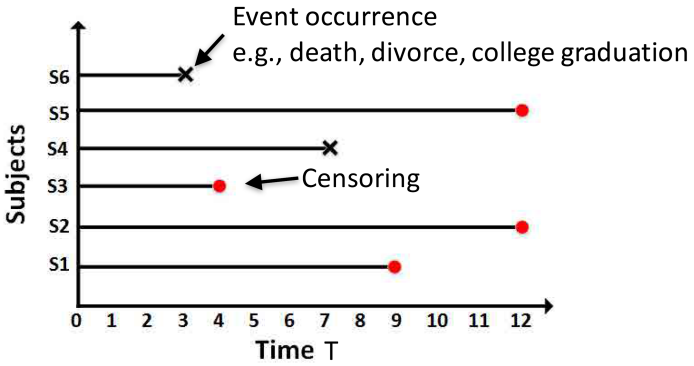


Figura 1. Ogni riga rappresenta un'istanza nello studio: le croci nere rappresentano il verificarsi di un evento e i punti rossi indicano i punti temporali censurati; nel caso di un punto temporale censurato sappiamo solo che l'evento d'interesse si è verificato dopo il punto temporale censurato [17].

Fig. 1 mostra uno studio in cui vengono osservati sei casi in un periodo di 12 mesi e vengono registrate le informazioni sull'accadimento dell'evento durante tale periodo di tempo, durante il quale [19]:

- S4 e S6 hanno effettivamente sperimentato l'evento;
- per i pazienti S1 e S3 è stato perso il monitoraggio;
- nei soggetti S2 e S5 non si è verificato alcun evento durante il periodo di studio.

Poiché l'evento non si è verificato entro il periodo di monitoraggio di 12 mesi per i soggetti S1, S2, S3 e S5, questi sono considerati censurati.

Una gestione naïve del problema, escludendo dal dataset le istanze censurate oppure trattando la censura come priva di eventi, non è risolutiva in quanto tali strategie possono funzionare solo in caso di un numero di campioni sufficientemente grande con pochi casi censurati; il rischio è che il modello tenda a non stimare correttamente i valori predetti, in quanto ignora le informazioni utili dei dati censurati [19].

B. Formulazione del problema

Una generica istanza i nel dataset è rappresentata da una tripletta $(\mathbf{x}_i, y_i, \delta_i)$, dove [19]

- $\mathbf{x}_i \in \mathbb{R}^p$ è il vettore delle *features* che caratterizza l'istanza i;
- δ_i è l'indicatore di evento binario (ossia $\delta_i = 1$ per un'istanza non censurata, $\delta_i = 0$ per un'istanza censurata);
- y_i denota il tempo osservato ed è uguale al tempo di sopravvivenza T_i per un'istanza non censurata e C_i per un'istanza censurata, ossia

$$y_i = \min(T_i, C_i) = \begin{cases} T_i & \delta_i = 1 \\ C_i & \delta_i = 0 \end{cases}$$

L'obiettivo è la stima del tempo T_j all'evento per una nuova istanza j con features $\mathbf{x}_j$ ($T_j > 0$, $T_j \in \mathbb{R}$) .

1) Modellazione statistica: La *funzione di sopravvivenza* $S(t)$ rappresenta la probabilità che il tempo T all'evento non sia anteriore ad un tempo t ("sopravvivenza a t") [19]:

$$S(t) = \mathsf{P}(T \geqslant t) \tag{1}$$

La funzione di sopravvivenza decresce monotonicamente con t e il valore iniziale è 1 per $t = 0$, ossia all'inizio dell'osservazione non si è verificato nessun evento d'interesse (in altre parole, a $t = 0$ il 100% dei soggetti osservati sopravvive).

La *funzione di densità della mortalità*[1] $f(t)$ è invece

$$f(t) = \frac{\mathrm{d}F(t)}{\mathrm{d}t} \tag{2a}$$

$$F(t) = \mathsf{P}(T < t) = 1 - S(t) \tag{2b}$$

essendo $F(t)$ la *funzione cumulativa della distribuzione di mortalità* (o *incidenza cumulativa*), ossia la probabilità che l'evento di interesse si verifichi prima di t [19]. Poiché è rilevante come le caratteristiche $\mathbf{x}$ influenzino la funzione di sopravvivenza, spesso si pone $f(t) = f(t|\mathbf{x})$ [17].

Fig. 2 riassume i legati tra le funzioni appena introdotte.

Altra funzione rilevante nell'analisi di sopravvivenza è la *funzione di hazard* $h(t)$ (o *funzione di "rischio"*) [19]:

$$\begin{aligned} h(t) &= \lim_{\Delta t \to 0} \frac{\mathsf{P}(t \leqslant T < t + \Delta t | T \geqslant t)}{\Delta t} \\ &= \lim_{\Delta t \to 0} \frac{\mathsf{P}(t \leqslant T < t + \Delta t)}{\Delta t \cdot \mathbb{P}(T \geqslant t)} \\ &= \lim_{\Delta t \to 0} \frac{F(t + \Delta t) - F(t)}{\Delta t \cdot S(t)} \\ &= \frac{f(t)}{S(t)} \end{aligned} \tag{3}$$

Intuitivamente la funzione di hazard $h(t)$ rappresenta la probabilità di morire (o più in generale che l'evento si manifesti) esattamente all'istante t.

Poiché da Eq. (2) si ha che $f(t) = -\mathrm{d}S(t)/\mathrm{d}t$, vale anche

$$h(t) = \frac{f(t)}{S(t)} = -\frac{\mathrm{d}S(t)}{\mathrm{d}t} \cdot \frac{1}{S(t)} = -\frac{\mathrm{d}\ln S(t)}{\mathrm{d}t}$$

da cui segue che

$$S(t) = \exp(-H(t)) \iff H(t) = -\log S(t) \tag{4}$$

con $H(t) = \int_0^t h(u)\mathrm{d}u$ la *funzione cumulativa di hazard* [19].

[1] Per consuetudine il lessico adottato è associato ad eventi di morte.

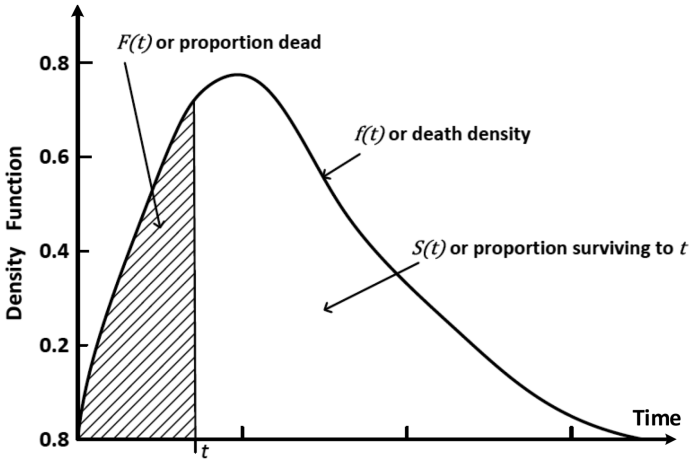


Figura 2. Legami tra le diverse entità $f(t)$, $F(t)$ e $S(t)$.

II. Metodi statistici per la survival analysis

I metodi statistici nella survival analysis si concentrano prevalentemente sulla caratterizzazione sia delle distribuzioni dei tempi all'evento, sia delle proprietà statistiche della stima dei parametri stimando le curve di sopravvivenza.

A grandi linee esistono tre diversi tipi di approcci comunemente utilizzati per stimare le funzioni di sopravvivenza e di rischio: metodi non parametrici, semi-parametrici e parametrici.

L'esempio più noto tra i modelli non paramedici è il modello di Kaplan-Meier[2] (§ II-1); i metodi non parametrici hanno il vantaggio di non fare ipotesi sulla natura della distribuzione dei tempi per l'evento d'interesse, ma non sono di facile interpretazione né consentono di modellare dati multivariati.

All'estremo opposto i metodi parametrici assumono una parametrizzazione $\boldsymbol{\theta}$ da stimare per $f(t)$ o $h(t)$ (Tab. I), con i parametri in $\boldsymbol{\theta}$ che possono essere espressi in funzione delle features $\mathbf{x}$ impiegando anche tecniche di machine learning.[3] Tuttavia la distribuzione assunta potrebbe essere errata, casuando un bias nel modello.

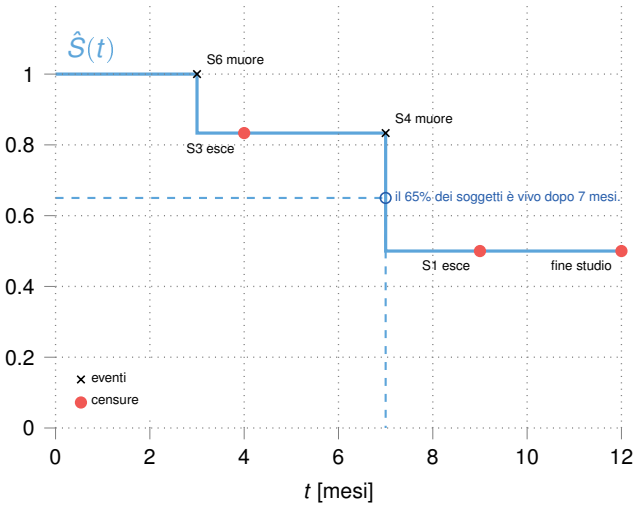
I metodi semi-parametrici cercano di combinare il meglio dei due precedenti approcci, come il modello di Cox (§ II-2).

Tabella I. Tipiche distribuzioni impiegate nei metodi parametrici per la survival analysis: si tratta prevalentemente di distribuzioni con lunghe code, trattandosi di tempi di attesa.

distribuzione	$S(t) = \int_t^\infty f(u)\mathrm{d}u$	$h(t)$
Esponenziale ($\lambda > 0$)	$\exp(-\lambda t)$	$\exp(\lambda)$
Weibull ($\lambda, \phi > 0$)	$\exp(-\lambda t^\phi)$	$\phi t^{\phi-1} \exp(\lambda)$
Gompertz ($\lambda, \phi > 0$)	$\exp\left(\lambda(1 - e^{\phi t})/\phi\right)$	$\exp(\lambda)\exp(\phi t)$

[2] Altro stimatore non parametrico è il Nelson-Aalen per l'hazard cumulato: $\hat{H}(t) = \sum_{i:T_i<t} = d_i/r_i$ (§ II-1 per d_i, r_i; d_i/r_i è una stima dell'hazard).

[3] La stima dei parametri è a massima verosimiglianza: poiché la probabilità che un evento non censurato si verifichi al tempo T_i è $\mathsf{P}_{\boldsymbol{\theta}}(T_i|\mathbf{x}_i, \delta_i = 1) = f_{\boldsymbol{\theta}}(T_i)$, mentre nel caso delle censure abbiamo $\mathsf{P}_{\boldsymbol{\theta}}(T_i|\mathbf{x}_i, \delta_i = 0) = \mathsf{P}_{\boldsymbol{\theta}}(t > T_i) = S_{\boldsymbol{\theta}}(t)$, assumendo eventi indipendenti la verosimiglianza è espressa da $L(\boldsymbol{\theta}) = \prod_{\delta_i=1} f_{\boldsymbol{\theta}}(T_i) \prod_{\delta_i=0} S_{\boldsymbol{\theta}}(T_i)$; la stima per $\boldsymbol{\theta}$ è quindi ottenuta massimizzando la log-verosimiglianza $\sum_i (1-\delta_i) \log S_{\boldsymbol{\theta}}(T_i) + \delta_i \log f_{\boldsymbol{\theta}}(T_i)$.



$N = 6,\ K = 2$

evento/cesura	d_j	c_{j-1}	r_j	$\mathbb{P}(t > T_j \mid \text{vivo})$	$\hat{S}(t)$
$T_6 = 3$	1	0	6	$1 - \frac{1}{6} = \frac{5}{6}$	$\frac{5}{6}$
$C_3 = 4$	1	1	6		
$T_4 = 7$	2	1	5	$1 - \frac{2}{5} = \frac{3}{5}$	$\frac{5}{6} \cdot \frac{3}{5} = \frac{1}{2}$
$C_1 = 9$	2	2	5		
$C_2, C_5 = 12$ (fine studio)	–	–	–		

Figura 3. Metodo di Kaplan-Meier sui dati di Fig. 1

1) Modello Kaplan-Meier: il modello Kaplan-Meier è il metodo non parametrico più impiegato [19]: solitamente si ricorre ai metodi non parametrici quando la distribuzione sottostante per T è ignota, oppure quando l'ipotesi di hazard proporzionale non è valida (cfr. § II-2).

Sia $\mathbb{T} = \{T_1 \dots T_K : T1 < T_2 < \dots T_K\}$ un insieme ordinato di tempi all'evento, con ciascun tempo osservato per ognuna di K istanze sulle N presenti nello studio ($K \leqslant N$); oltre a questi tempi all'evento, ci sono anche i tempi censurati per le $N - K$ istanze i cui eventi non sono osservati.

Per uno specifico tempo all'evento $T_j \in \mathbb{T}$ ($j = 1, 2 \dots K$), sia $d_j \geqslant 1$ il numero di eventi osservati; poniamo inoltre

$$r_j = r_{j-1} - d_j - c_{j-1}$$

essendo c_{j-1} il numero di istanze censurate nell'intervallo di tempo (T_{j-1}, T_j): r_j è cioè il numero di soggetti vivi e non già censurati, potenzialmente ancora a rischio di morte (osservata) oltre il tempo T_j. Attraverso d_j e r_j la probabilità di sopravvivere oltre il tempo T_j può ora essere espressa come

$$\mathsf{P}(t > T_j \mid \text{ancora vivo in } t) = \frac{r_j - d_j}{r_j}$$

da cui segue che la funzione di sopravvivenza $S(t)$ è data da

$$\hat{S}(t) = \prod_{j:T_j<t} \mathsf{P}(t > T_j \mid \text{ancora vivo in } t) = \prod_{j:T_j<t} \left(1 - \frac{d_j}{r_j}\right) \quad (5)$$

Fig. 3 illustra la funzione di sopravvivenza $\hat{S}(t)$ calcolata sui dati di Fig. 1.

2) Modello di Cox: Nella categoria semi-parametrica (ed in generale tra i metodi statistici), il modello di Cox è l'approccio più comunemente utilizzato per i dati di sopravvivenza [19].

L'ipotesi alla base del modello di Cox è che le funzioni di hazard di due individui siano proporzionali nel tempo (ipotesi di *hazard proporzionali*), assumendo per $h(t)$ la forma

$$h(t|\mathbf{x}) = h_0(t) \exp\left(\boldsymbol{\beta}^\top \mathbf{x}\right) \quad (6)$$

avendo modellato nella regressione $\boldsymbol{\beta}^\top \mathbf{x}$ la dipendenza della funzione di hazard con le caratteristiche $\mathbf{x}$ del soggetto; la funzione $h_0(t)$ è la funzione di hazard di base, e non necessita di essere determinata [6] (viene chiesto solo che $h_0(t) \geqslant 0$).

Da Eq. (6), per due istanze i e j, lo *hazard ratio* è indipendente da $h_0(t)$:

$$\frac{h(t|\mathbf{x}_1)}{h(t|\mathbf{x}_2)} = \frac{h_0(t) \exp(\boldsymbol{\beta}^\top \mathbf{x}_1)}{h_0(t) \exp(\boldsymbol{\beta}^\top \mathbf{x}_2)} = \exp\left(\boldsymbol{\beta}^\top (\mathbf{x}_1 - \mathbf{x}_2)\right) \quad (7)$$

Da Eq. (4) abbiamo infine che

$$S(t) = \exp\left(-H_0(t) \exp\left(\boldsymbol{\beta}^\top \mathbf{x}\right)\right) = S_0(t)^{\exp\left(\boldsymbol{\beta}^\top \mathbf{x}\right)} \quad (8)$$

essendo $S_0(t) = \exp(-H_0(t))$ la funzione di sopravvivenza di base (non specificata).

a) Stima del modello: Poiché la funzione di hazard di base $h_0(t|\mathbf{x})$ nel modello di Cox di base non è specificata, non è possibile adattare questo modello utilizzando la funzione di verosimiglianza standard: La funzione di rischio $h_0(t|\mathbf{x})$ è effettivamente una funzione di disturbo, mentre i coefficienti in $\boldsymbol{\beta}$ sono i parametri di interesse nel modello.

L'assunzione di proporzionalità degli hazard permette di decomporre la verosimiglianza in due parti: una è prevalentemente influenzata dall'hazard di base mentre l'altra, detta *verosimiglianza parziale*, è indipendente da $h_0(t|\mathbf{x})$ [5]. Si procede quindi a stimare $\boldsymbol{\beta}$ massimizzando la verosimiglianza parziale $L(\boldsymbol{\beta})$, dato che permette di non considerare $h_0(t|\mathbf{x})$:

$$\hat{\boldsymbol{\beta}} = \arg\max_{\boldsymbol{\beta}} \underbrace{\prod_i \left(\frac{\exp(\boldsymbol{\beta}^\top \mathbf{x}_i)}{\sum_{j \in \mathcal{R}_i} \exp(\boldsymbol{\beta}^\top \mathbf{x}_j)} \right)}_{L(\boldsymbol{\beta})}$$

con $\mathcal{R}_i$ l'insieme dei soggetti a rischio in T_i (incluso i).

Tramite $\hat{\boldsymbol{\beta}}$ una stima *non parametrica* di $H_0(t)$ (ma non essenziale) è ottenibile con lo stimatore di Aalen-Breslow:[4]

$$\hat{H}_0(t) = \sum_{i:T_i<t} \frac{d_i}{\sum_{j \in \mathcal{R}_i} \exp(\hat{\boldsymbol{\beta}}^\top \mathbf{x}_i)}$$

III. SVM PER LA SURVIVAL ANALYSIS

In alternativa ai metodi statistici, i metodi di machine learning applicati alla survival analysis si concentrano principalmente sulla previsione del verificarsi dell'evento in un dato momento, combinando la potenza dei metodi tradizionali di analisi della sopravvivenza con varie tecniche di apprendimento automatico tra cui Survival Random Forest, reti neurali e Support Vector Machines (SVMs) [19].

In questo report verranno analizzate delle formulazioni in termini di SVMs (§ III-B).

[4] d_i è sempre il numero di eventi osservati fino al tempo T_i. Il denominatore rappresenta la somma dei punteggi di rischio per tutti gli individui a rischio al tempo T_i. Il punteggio di rischio per ogni individuo j è stimato in base a $\hat{\boldsymbol{\beta}}$ e $\mathbf{x}_j$.

A. SVM standard

1) Classificazione binaria: Le SVM sono state inizialmente proposte per problemi di classificazione binaria supervisionata: pertanto la formulazione di una SVM presuppone una variabile target $z \in \{-1, 1\}$ e features $\mathbf{x} \in R^d$, dato un dataset $\{(\mathbf{x}_n, z_n) : n = 1 \dots N\}$.

Supponendo che le due classi target siano linearmente separabili[5], esiste una funzione lineare $f(\mathbf{x}) = \mathtt{w}^\top \mathbf{x} + b$ tale che $z_n f(\mathbf{x}_n) > 0$, $\forall\, n$. Il compito della SVM è trovare l'iperpiano di separazione $\Pi_f = \{\mathbf{x} : f(\mathbf{x}) = 0\}$ con il massimo *margine* di separazione per le due classi, essendo il margine la distanza più piccola tra qualsiasi punto $\mathbf{x}_n$ e l'iperpiano di separazione (Fig. 4a): i punti che si trovano a questa distanza dall'iperpiano di separazione sono chiamati *vettori di supporto*, in quanto determinano il margine e verificando esattamente $f(\mathbf{x}) = \pm 1$ [6]. Per un'approfondimento su come si ricavi l'iperpiano $\hat{\Pi}_f$ a massimo margine, si veda l'appendice in § A-A.

Nel caso di due classi non linearmente separabili, possono essere consentite classificazioni errate. Ciò si ottiene introducendo un vettore $\boldsymbol{\xi} = [\xi_1 \dots \xi_N]^\top$ di *variabili di slack* $\xi_n \geqslant 0$, consentendo ma penalizzando le classificazioni errate (Fig. 4b) [2, p. 331-332]:

- $\xi_n = 0$ per le classificazioni "sicure" (sopra o all'interno del margine di confine),
- altrimenti, $\xi_n = |z_n - f(\mathbf{x}_n)|$ per cui per un punto $\mathbf{x}_n$
  - se $f(\mathbf{x}_n) = 0$ allora $\xi_n = 1$;
  - se $\xi_n > 1$, il punto è classificato erroneamente;
  - se $0 < \xi_n < 1$, il punto è nel lato corretto ma all'interno del margine.

I parametri dell'iperpiano sono determinati risolvendo il seguente problema di ottimizzazione vincolata [13, p. 888]

$$\min_{\mathbf{w}, b, \boldsymbol{\xi}} \frac{1}{2}\|\mathbf{w}\|^2 + \gamma \sum_n \xi_n \tag{9a}$$

$$\text{t.c. } z_n f(\mathbf{x}_n) \geqslant 1 - \xi_n \tag{9b}$$

$$\text{con } \xi_n \geqslant 0 \quad (n = 1 \dots N) \tag{9c}$$

con γ parametro di regolarizzazione da determinare tramite cross-validation. Si veda l'appendice § A-B per ulteriori informazioni nella determinazione dei parametri $\hat{\mathbf{w}}$ e $\hat{b}$ per l'iperpiano di separazione $\hat{\Pi}_f$.

a) Il "kernel trick": Nel caso più generale, le SVM ricorrono a delle funzioni $\phi(\mathbf{x}_n)$ che pre-proiettano le features $\mathbf{x}_n$ in uno spazio vettoriale a dimensioni $D > d$.

La trattazione è in gran parte analoga a quanto già visto, lavorando con $\phi(\mathbf{x}_n) \in \mathbb{R}^D$ invece che con $\mathbf{x}_n \in \mathbb{R}^d$. La novità risiede nel calcolo della matrice $\mathtt{G}$ di Eq. (28) in appendice § A, i cui elementi adesso sono dati da

$$G_{ij} = \phi(\mathbf{x}_i)^\top \phi(\mathbf{x}_j) \tag{10}$$

Il "trucco" risiede nel fatto che non è necessario esplicitare $\phi(\mathbf{x})$, ma è sufficiente rimpiazzare $\mathtt{G} \in R^{N \times N}$ con una matrice

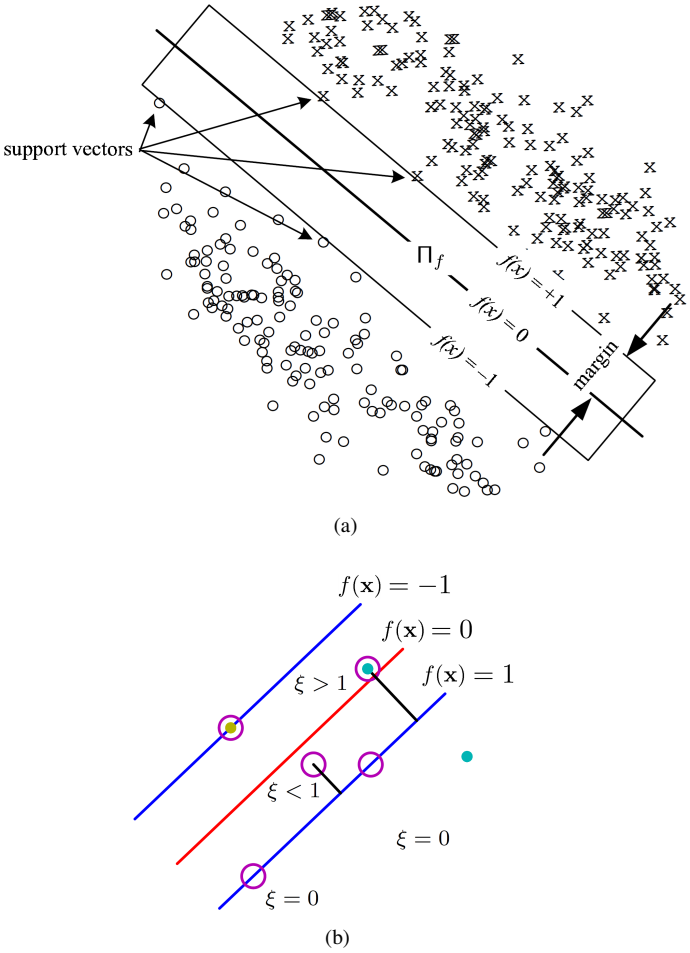


Figura 4. Iperpiano separatore Π_f con massimo margine tra le due classi di dati: in (a) il caso linearmente separabile [13, p. 885]; in (b) impiegando le variabili di slack $\xi_n \geqslant 0$ (i punti cerchiati sono vettori di supporto) [2, p. 332].

$\mathtt{K} \in R^{N \times N}$ che sia definita positiva[6], i cui elementi K_{ij} siano funzione dei vettori $\mathbf{x}_i$ e $\mathbf{x}_j$ [15, p. 751]: $K_{ij} = \kappa(\mathbf{x}_i, \mathbf{x}_j)$, dove la funzione $\kappa(\mathbf{x}_i, \mathbf{x}_j)$ costituisce un *kernel*.

Possibili kernel proposti in letteratura sono:

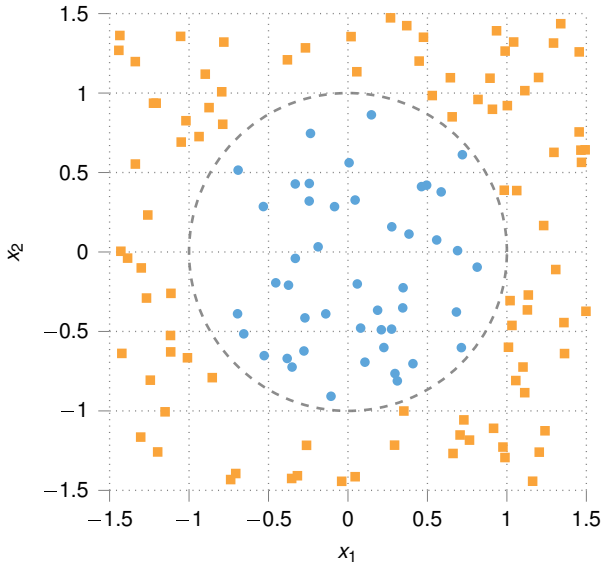
- $\kappa(\mathbf{x}_i, \mathbf{x}_j) = \mathbf{x}_i^\top \mathbf{x}_j$ (kernel lineare)
- $\kappa(\mathbf{x}_i, \mathbf{x}_j) = (\mathbf{x}_i^\top \mathbf{x}_j)^d$ (kernel a potenza)
- $\kappa(\mathbf{x}_i, \mathbf{x}_j) = (\alpha \mathbf{x}_i^\top \mathbf{x}_j + \beta)^d$ (kernel polinomiale)
- $\kappa(\mathbf{x}_i, \mathbf{x}_j) = \tanh(\alpha \mathbf{x}_i^\top \mathbf{x}_j + \beta)$ (kernel sigmoideo)
- $\kappa(\mathbf{x}_i, \mathbf{x}_j) = \exp(-\frac{1}{2}|\mathbf{x}_i - \mathbf{x}_j|^2/\sigma^2)$ (RBF gaussiana)

Anche gli eventuali iper-parametri che governano i vari kernel devono essere determinati tramite cross-validation. Fig. 5 illustra il comportamento di una SVM con kernel a potenza su un dataset non separabile linearmente [15, p. 750].

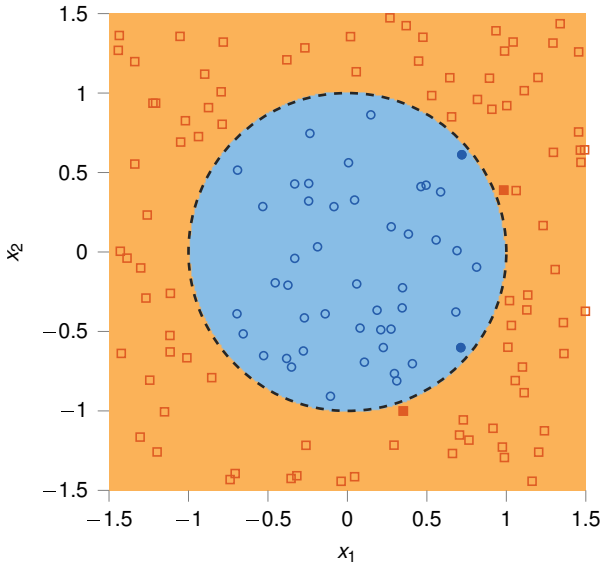
2) Support Vector Regression (SVR): Le SVM sono state successivamente estese a problemi di regressione, minimizzando una funzione di costo regolarizzata. Mentre nella regressione lineare semplice la funzione di costo regolarizzata è

[5] Due insiemi $\mathbb{A}, \mathbb{B} \subset \mathbb{R}^d$ si dicono *linearmente separabili* se esistono $w_0 \dots w_d \in \mathbb{R}$ tali che $\sum_{i=1}^d w_i x_i \geqslant w_0 \forall\, \mathbf{x} \in \mathbb{A}$ e $\sum_{i=1}^d w_i x_i < w_0 \forall\, \mathbf{x} \in \mathbb{B}$ [14, p. 63]

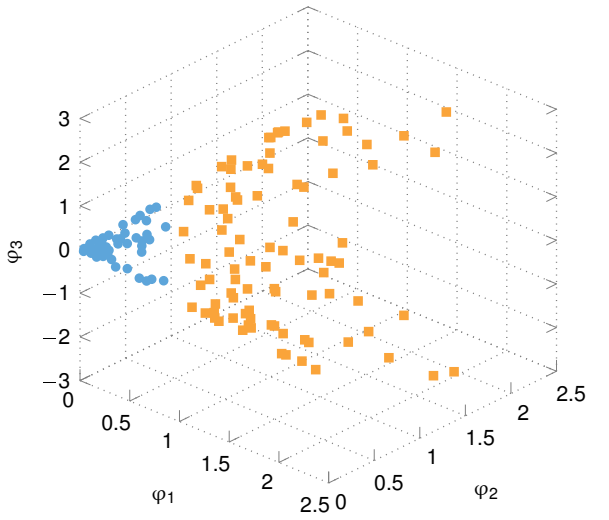
[6] Una matrice $\mathtt{M} \in \mathbb{R}^{d \times d}$ è *definita positiva* se $\mathbf{v}^\top \mathtt{M} \mathbf{v} > 0$, $\forall\, \mathbf{v} \in \mathbb{R}^d - \mathbf{0}$ [1, p. 323].



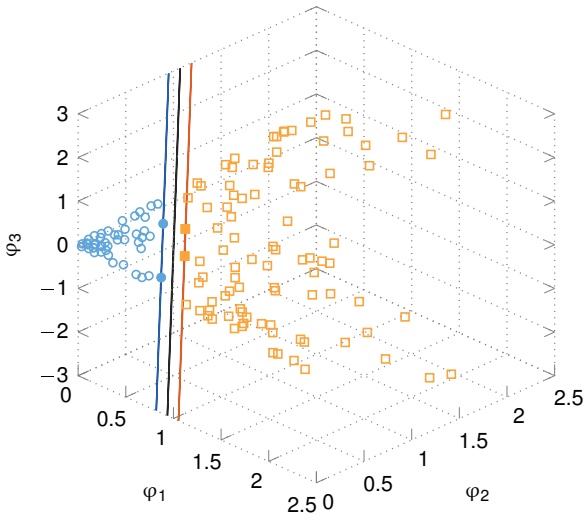
(a) dati in input, con bordo di separazione $x_1^1 + x_2^2 = 1$ (cerchio a tratti)



(b) classificazione con SVM ($\gamma = 1000$)



(c) mappatura da $\mathbb{R}^2$ a $\mathbb{R}^3$ tramite $\phi(\mathbf{x})$



(d) piano separatore $\hat{\Pi}_f$ in $\mathbb{R}^3$

Figura 5. La mappatura tramite $\phi(\mathbf{x}) = [x_1^2 \; x_2^2 \; \sqrt{2}x_1x_2]^\top = [\varphi_1 \; \varphi_2 \; \varphi_3]^\top$, corrispondente al kernel a potenza $(\mathbf{x}_i^\top \mathbf{x}_j)^2$, consente la corretta classificazione dei dati in (a). I punti marcati come pieni in (b), (d) sono i vettori di supporto.

tipicamente data da

$$\frac{1}{2}\sum_n (z_n - f(\mathbf{x}_n))^2 + \frac{\rho}{2}\|\mathbf{w}\|^2$$

con $z_n \in \mathbb{R}$ e $f(\mathbf{x}) = \mathbf{w}^\top \mathbf{x} + b$, nel caso della regressione tramite SVM si ricorre ad una funzione di errore ϵ-*insensitive*, con $\epsilon \geqslant 0$ (Fig. 6a) [2, 340]:

$$\hat{\mathbf{w}}, \hat{b} = \arg\min_{\mathbf{w},b} \frac{1}{2}\|\mathbf{w}\|^2 + \gamma \sum_n E_\epsilon(|f(\mathbf{x}_n) - z_n|) \tag{11a}$$

$$E_\epsilon(r) = \begin{cases} 0 & |r| < \epsilon \\ r - \epsilon & \text{altrimenti} \end{cases} \tag{11b}$$

Introducendo dele variabili di slack ξ_n, ξ_n^* possiamo tenere conto di quando $z_n - f(\mathbf{x}) > \epsilon$ e $z_n - f(\mathbf{x}) < -\epsilon$, per modellare quando $f(\mathbf{x})$ esce dal "tubo" di raggio ϵ attorno a z_n (Fig. 6b):

$$\begin{aligned} z_n - f(\mathbf{x}_n) &\leqslant \epsilon + \xi_n \\ z_n - f(\mathbf{x}_n) &\geqslant -\epsilon - \xi_n^* \\ \xi_n, \xi_n^* &\geqslant 0 \end{aligned}$$

giungendo quindi al problema

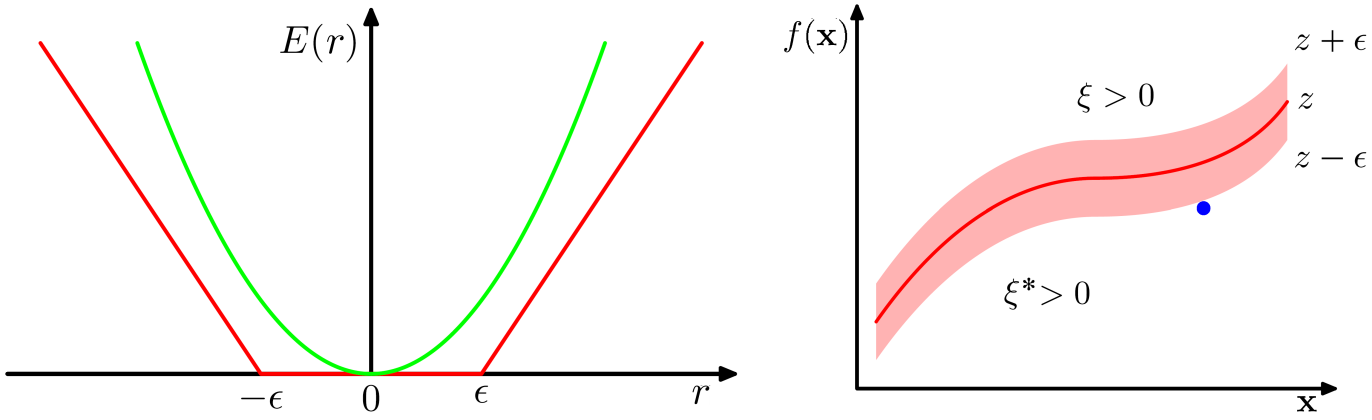
$$\hat{\mathbf{w}}, \hat{b} = \arg\min_{\mathbf{w},b} \frac{1}{2}\|\mathbf{w}\|^2 + \gamma \sum_n (\xi_n + \xi_n^*)^2 \tag{12a}$$

$$z_n - f(\mathbf{x}_n) \leqslant \epsilon + \xi_n \tag{12b}$$

$$z_n - f(\mathbf{x}_n) \geqslant -\epsilon - \xi_n^* \tag{12c}$$

$$\xi_n, \xi_n^* \geqslant 0 \tag{12d}$$

il cui problema duale, introducendo per completezza anche il kernel $\kappa(\mathbf{x}_i, \mathbf{x}_j) = \phi(\mathbf{x}_i^\top \mathbf{x}_j)$, richiede di massimizzare la



(a) (b)

Figura 6. In (a), la funzione d'errore ϵ-insensitive $E_\epsilon(x)$ (in rosso), a confronto con la funzione quadratica (in verde). In (b), la curva di regressione insieme al "tubo" ϵ-insensitive, con alcuni esempi delle variabili di slack ξ, ξ^*: i punti sopra il "tubo" hanno $\xi_> 0$ e $\xi^* = 0$, i punti al di sotto hanno $\xi = 0$ e $\xi^* > 0$ mentre per i punti all'interno $\xi = \xi^* = 0$ [2, pp. 340-341].

funzione [2, 342]

$$\mathcal{L}(\lambda, \lambda^*) = \sum_n (\lambda_n - \lambda_n^*) z_n - \epsilon \sum_n (\lambda_n + \lambda_n^*) - \frac{1}{2} \sum_i \sum_j (\lambda_i, \lambda_i^*)(\lambda_j - \lambda_j^*) \kappa(\mathbf{x}_i, \mathbf{x}_j)$$

vincolata a $\lambda_n, \lambda_n^* \in [0, \gamma]$, con $\sum_n \lambda_n - \lambda_n^* = 0$.

Ricavati i valori ottimi $\hat{\lambda}_n, \hat{\lambda}_n^*$, la regressione è infine ottenuta come [2, 342]

$$f(\mathbf{x}) = \sum_n (\hat{\lambda}_n - \hat{\lambda}_n^*) \kappa(\mathbf{x}_n, \mathbf{x}) + \hat{b} \tag{13}$$

Per determinare $\hat{b}$ basta prendere un qualsiasi punto $\mathbf{x}_m$ per cui $\hat{\lambda}_m \in (0, \gamma)$ oppure $\hat{\lambda}_m^* \in (0, \gamma)$, da cui [2, 343]

$$\hat{b} = z_n - \epsilon - \sum_n (\hat{\lambda}_n - \hat{\lambda}_n^*) \kappa(\mathbf{x}_n, \mathbf{x}_m) \tag{14}$$

3) Ranking SVM: Differentemente dai problemi di classificazione o regressione, nei problemi di *ranking* il training set $\mathcal{D} = \{(\mathbf{x}_n, z_n) : n = 1 \dots N\}$ è ordinato: z_n è il ranking di $\mathbf{x}_n$, ovvero $z_i < z_j$ denota che $\mathbf{x}_i$ è *preferibile* a $\mathbf{x}_j$ ($\mathbf{x}_i \succ \mathbf{x}_j$): poiché non è detto che tutte le istanze siano confrontabili, viene definito l'insieme delle *preferenze* $\mathbb{P} = \{(i, j) : \mathbf{x}_i \succ \mathbf{x}_j\}$, formato delle coppie confrontabili.[7]

Pertanto una funzione di ranking $f(\mathbf{x})$ restituisce un *punteggio* per ciascuna istanza $\mathbf{x}$ tale che [20]

$$f(\mathbf{x}_i) > f(\mathbf{x}_j) \text{ per qualsiasi } (\mathbf{x}_i, \mathbf{x}_j) \in \mathbb{P}$$

Nel caso delle SVM di ranking $f(\mathbf{x}) = \mathbf{w}^\top \mathbf{x}$, ovvero si cerca il vettore $\mathbf{w}$ lungo il quale la proiezione di $\mathbf{x}_i$ è maggiore di quella per $\mathbf{x}_j$ (Fig. 7):

$$\mathbf{w}^\top \mathbf{x}_i > \mathbf{w}^\top \mathbf{x}_j \iff \mathbf{w}^\top (\mathbf{x}_i - \mathbf{x}_j) > 0 \qquad \text{se } \mathbf{x}_i \succ \mathbf{x}_j \tag{15}$$

[7] Il problema nasce nell'ambito dell'information retrieval, per ottimizzare il ranking delle pagine web nei motori di ricerca [8], dove non tutti i risultati sono confrontabili con le preferenze dell'utente.

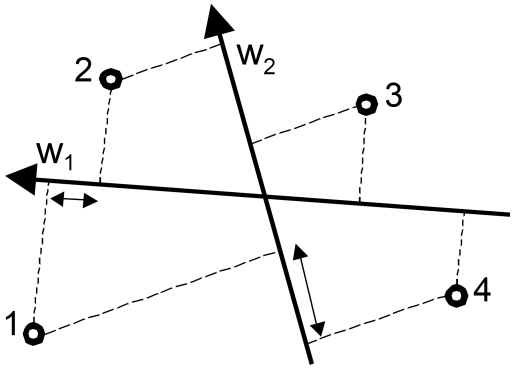


Figura 7. Esempio di come due vettori $\mathbf{w}_1$ e $\mathbf{w}_2$ ordinano quattro punti: per $\mathbf{w}_1$ i punti sono ordinati (1, 2, 3, 4), mentre $\mathbf{w}_2$ implica l'ordinamento (2, 3, 1, 4) [8].

Introducendo delle variabili di slack $\xi_{ij} \geqslant 0$, il problema è esprimibile come [7][20]

$$\min_{\mathbf{w}, \xi_n} \frac{1}{2} \|\mathbf{w}\|^2 + \gamma \sum_{(i,j) \in \mathbb{P}} \xi_{ij} \tag{16a}$$

$$\text{t. c. } \mathbf{w}^\top (\mathbf{x}_i - \mathbf{x}_j) \geqslant 1 - \xi_{ij} \quad \forall (i, j) \in \mathbb{P} \tag{16b}$$

$$\xi_{ij} \geqslant 0 \quad \forall (i, j) \in \mathbb{P} \tag{16c}$$

Possiamo quindi estendere a tale caso i metodi adottati nelle SVM [20] (si confronti Eq. (16) con Eq. (9), dove $z_n f(\mathbf{x}_n)$ e ξ_n sono stati sostituiti rispettivamente da $\mathbf{w}^\top (\mathbf{x}_i - \mathbf{x}_j)$ e ξ_{ij}).

Tuttavia, piuttosto che lavorare sul problema duale come è consuetudine procedere con le SVM, in questo caso è preferibile riformulare Eq. (16) come[8] [3]

$$\min_{\mathbf{w}} \frac{1}{2} \|\mathbf{w}\|^2 + \frac{\gamma}{2} \sum_{(i,j) \in \mathbb{P}} \max(0, 1 - \mathbf{w}^\top (\mathbf{x}_i - \mathbf{x}_j))^2 \tag{17}$$

che può essere espressa in forma matriciale come

$$\min_{\mathbf{w}} \frac{1}{2} \mathbf{w}^\top \mathbf{w} + \frac{\gamma}{2} (\mathbf{1} - \mathtt{AX}\mathbf{w})^\top \mathtt{D}_\mathbf{w} (\mathbf{1} - \mathtt{AX}\mathbf{w}) \tag{18}$$

[8] Da Eq. (16), si può vedere che $\xi_{ij} \geqslant \max(0, 1 - \mathbf{w}^\top (\mathbf{x}_i - \mathbf{x}_j)) \geqslant 0$.

dove

- $\mathbf{1} = [1 \dots 1]^\top \in \mathbb{R}^p$ con $p = |\mathbb{P}|$;
- $\mathtt{X} = [\mathbf{x}_1 \dots \mathbf{x}_N]^\top$;
- $\mathtt{A} \in \mathbb{R}^{p \times N}$ è una matrice sparsa la cui k^{a} riga corrisponde alla k^{a} coppia $(i, j) \in \mathbb{P}$, con $A_{ki} = 1$ e $A_{kj} = -1$ (0 altrimenti);
- $\mathtt{D}_\mathbf{w}$ è una matrice diagonale $p \times p$ tale che in corrispondenza della k^{a} coppia $(i, j) \in \mathbb{P}$

$$(\mathtt{D}_\mathbf{w})_{kk} = I(\mathbf{w}^\top(\mathbf{x}_i - \mathbf{x}_j) < 1)$$

La soluzione è quindi determinata sul problema primale, impiegando un metodo di Newton troncato: rispetto al metodo di Newton standard, che aggiorna ad ogni iterazione $\hat{\mathbf{w}}_{k+1} \leftarrow \hat{\mathbf{w}}_k - \mathtt{H}^{-1}\mathbf{g}$ (essendo rispettivamente $\mathtt{H}$ e $\mathbf{g}$ la matrice Hessiana ed il gradiente della funzione obbiettivo), in questo caso l'Hessiana $\mathtt{H}$ non viene mai calcolata esplicitamente ed il calcolo di $\mathtt{H}^{-1}\mathbf{g}$ è ottenuto tramite gradiente coniugato in maniera approssimata, richiedendo solo il calcolo di $\mathtt{H}\mathbf{s}$ per qualche $\mathbf{s}$; inoltre, poiché la funzione obbiettivo non è differenziabile due volte, al posto dell'Hessiana viene impiegata la sua versione generalizzata $\mathtt{H} = \mathtt{I} + \gamma\mathtt{X}^\top\mathtt{A}^\top\mathtt{D}_\mathbf{w}\mathtt{A}\mathtt{X}$.

Il costo computazionale di un passo del gradiente coniugato ($\mathtt{H}\mathbf{s} = \mathbf{s} + \gamma\mathtt{X}^\top\mathtt{A}^\top\mathtt{D}_\mathbf{w}\mathtt{A}\mathtt{X}\mathbf{s}$) è $O(Nd + p)$ procedendo da destra, per la sparsità di $\mathtt{A}$ e $\mathtt{D}_\mathbf{w}$; in [10] la tecnica appena descritta è ulteriormente velocizzata ricorrendo a degli speciali alberi binari bilanciati (gli *order statistic tree*).[9]

In [9] inoltre un approccio simile a [10] è applicato al caso con kernel, partendo dalla funzione di costo primale

$$\frac{1}{2}\boldsymbol{\beta}^\top\mathtt{K}\boldsymbol{\beta} + \frac{\gamma}{2}(\mathbf{1} - \mathtt{A}\mathtt{K}\boldsymbol{\beta})^\top\mathtt{D}_{\boldsymbol{\beta}}(\mathbf{1} - \mathtt{A}\mathtt{K}\boldsymbol{\beta})$$

dove $\mathtt{K}$ è la matrice di kernel ($K_{ij} = \kappa(\mathbf{x}_i, \mathbf{x}_j)$), ed avendo espresso $\mathbf{w}$ come combinazione lineare delle features mappate da $\phi(\mathbf{x})$ ossia $\mathbf{w} = \boldsymbol{\beta}^\top[\phi(\mathbf{x}_1) \dots \phi(\mathbf{x}_N)]$ (cfr. Eq. (33)).

B. Survival SVM

Il modello di Cox (§ II-2) è lo standard per l'analisi dei dati time-to-event. Tuttavia la sua funzione di decisione è lineare in $\mathbf{x}$, il che può portare a scarse prestazioni predittive se le non linearità e le interazioni non vengono modellate in modo esplicito, con il peso di dover provare molte formulazioni di modelli. Il successo dei metodi kernel nell'apprendimento automatico ha motivato i ricercatori a proporre modelli di sopravvivenza basati su kernel, che semplificano l'analisi in presenza di non linearità [12].

In particolare, gli approcci di survival analysis basati su SVM sono riconducibili a formulazioni in termini di SVR o di Ranking SVM. A tal proposito, dati due vettori $\mathbf{u} = [u_1 \dots u_d]^\top$, $\mathbf{v} = [v_1 \dots v_d]^\top$, un kernel pensato proprio per dati clinici (data la loro eterogeneità con variabili continue, ordinali e nominali) è [4]

$$\bar{\kappa}(\mathbf{u}, \mathbf{v}) = \frac{1}{d}\sum_{p=1}^{d}\bar{\kappa}_p(u_p, v_p) \tag{19a}$$

$$\bar{\kappa}_p(u_p, v_p) = \begin{cases} \frac{\Delta_p - |u_p - v_p|}{\Delta_p} & (u_p, v_p \text{ continue o ordinali}) \\ I(u_p = v_p) & (u_p, v_p \text{ categor. o binarie}) \end{cases} \tag{19b}$$

avendo definito $I(s) = 1$ se s è vera e $I(s) = 0$ altrimenti, ed essendo Δ_p lo scarto massimo nel dataset per la p^{a} variabile.

1) Survival Support Vector Regression: Nel caso delle SVR, l'idea è di penalizzare diversamente i dati a seconda che siano censurati o meno. Con riferimento a [16],

- per le osservazioni censurate, il tempo all'evento dopo la censura è sconosciuto e quindi le previsioni maggiori del tempo di censura non devono essere penalizzate;
- tuttavia, tutte le previsioni di sopravvivenza inferiori al tempo di censura vengono penalizzate come al solito.

Per i dati non censurati, i tempi di sopravvivenza esatti sono noti e, come nell'SVR standard, tutte le previsioni di sopravvivenza inferiori o superiori al tempo di sopravvivenza osservato vengono penalizzate [6]. Il problema di ottimizzazione vincolato è così espresso:

$$\min_{\mathbf{w}, b, \boldsymbol{\xi}} \frac{1}{2}\|\mathbf{w}\|^2 + \gamma\sum_n(\xi_n + \xi_n^*) \tag{20a}$$

$$\text{t. c. } z_n - (\mathbf{w}^\top\phi(\mathbf{x}) + b) \leqslant \xi_n, \tag{20b}$$

$$\delta_i(\mathbf{w}^\top\phi(\mathbf{x}) + b - z_n) \leqslant \xi_n^* \tag{20c}$$

$$\text{con } \xi_n, \xi_n^* \geqslant 0 \quad (n = 1 \dots N) \tag{20d}$$

avendo introdotto in Eq. (20c) l'indicatore di censura δ_n.

Indicando con $\hat{\lambda}_n, \hat{\lambda}_n^*$ le soluzioni del problema duale, la previsione per una nuova osservazione con features $\mathbf{x}$ è infine ottenuta come

$$f(\mathbf{x}) = \sum_n(\hat{\lambda}_n - \delta_n\hat{\lambda}_n^*)\bar{\kappa}(\mathbf{x}_n, \mathbf{x}) + \hat{b} \tag{21}$$

Si noti che in Eq. 20 è stato posto $\epsilon = 0$: questa scelta riduce sostanzialmente la complessità delle formulazioni, mentre non è stata osservata alcuna conseguente perdita di prestazioni negli esperimenti [18].

2) Survival Ranking SVM: Nel ranking l'obiettivo è recuperare l'ordine corretto dei campioni in base alla loro pertinenza; nella survival analysis, la pertinenza corrisponde al tempo di sopravvivenza.

Quando si addestra un modello di sopravvivenza, si deve considerare che una coppia (i, j) con dati $(\mathbf{x}_i, y_i, \delta_i)$ e $(\mathbf{x}_j, yj, \imath j)$ è confrontabile solo ogni volta che è noto l'ordine dei tempi dei loro eventi (ad esempio due eventi o un evento e un punto censurato a destra per i quali il tempo di censura di quest'ultimo è successivo al tempo di errore del primo) [18]; in altre parole, la coppia (i, j) si dice confrontabile quando

- entrambe le istanze hanno sperimentato un evento (ossia $\delta_i, \delta_j = 1$),
- oppure solo una di loro ha sperimentato un evento e il momento dell'evento si è verificato prima della censura (formalmente $(y_i < y_j \wedge \delta_i = 1) \vee (y_i > y_j \wedge \delta_j = 1)$).

[9] Se fosse stata usato il metodo di Newton standard, il calcolo dell'Hessiana avrebbe richiesto $O(Nd^2)$, con $O(d^3)$ per risolvere poi $\mathtt{H}^{-1}\mathbf{g}$ (d è il numero di feature in un generico $\mathbf{x}_n \in \mathbb{R}^d$).

Se la coppia (i, j) non soddisfa questa condizione, è incomparabile e la relazione dei loro dati non può essere utilizzata per dedurre un modello di sopravvivenza [11].

Pertanto, un confronto tra le istanze i e j è valido solo se l'istanza con il tempo osservato minore non è censurata: l'insieme

$$\mathbb{P} = \{(i, j) : y_i > y_j \wedge \delta_j = 1, \;\; i, j = 1 \dots N\}$$

definisce quindi le coppie di campioni comparabili (assumendo tempi univoci) che possono essere utilizzate per l'addestramento.

Avendo definito l'insieme $\mathbb{P}$ delle coppie confrontabili (e ordinate), il problema è trattabile in maniera simile ad Eq. (17): al fine di realizzare un'implementazione efficiente, in [11] è stato applicato il metodo proposto in [10] per le SVM lineari e quello in [9] per le SVM con kernel.

3) C-index: La metrica primaria utilizzata per valutare i modelli di sopravvivenza è *l'indice di concordanza*, altrimenti noto come *C-index*: è il rapporto tra il numero di coppie concordanti e il numero di coppie comparabili, ossia

$$c = \frac{1}{|\mathbb{P}|} \sum_{i:\delta_i=0} \sum_{j=T_i<T_j} I(\hat{T}_i < \hat{T}_j) \tag{22}$$

L'idea dell'indice di concordanza è che un buon modello dovrebbe classificare le istanze per le quali l'evento si manifesta più tardi in una posizione più alta rispetto alle istanze per le quali si verifica prima.

Appendice A
Ottimizzazione per le SVM

A. Caso linearmente separabile

Data la funzione lineare $f(\mathbf{x}|\mathbf{w}, b) = \mathbf{w}^\top \mathbf{x} + b$ e partendo dal caso di un insieme di dati $(\mathbf{x}_n, z_n)$ linearmente separabile, con $\mathbf{x} \in \mathbb{R}^d$ e $y_n \in \{\pm 1\}$, la distanza (con segno) tra un punto $\mathbf{x}_n$ e l'iperpiano $\Pi_f = \{\mathbf{x} : f(\mathbf{x}) = 0\}$ è data da [2, pp. 327-328]

$$d_\perp(\mathbf{x}_n, \Pi_f) = \frac{\mathbf{w}^\top \mathbf{x}_n + b}{\|\mathbf{w}\|}$$

L'indice di concordanza è il rapporto tra il numero di coppie concordanti e il numero di coppie comparabili Tra tutti gli iperpiani Π_f tali che $z_n f(\mathbf{x}) > 0 \, \forall \, n$, quello con il massimo margine è dato da

$$\begin{aligned} \hat{\Pi}_f &= \arg\max_{\Pi_f} \min_n (z_n \cdot d_\perp(\mathbf{x}_n, \Pi_f)) \\ &= \arg\max_{\mathbf{w},b} \min_n \frac{z_n(\mathbf{w}^\top \mathbf{x}_n + b)}{\|\mathbf{x}\|} \\ &= \arg\max_{\mathbf{w},b} \left(\frac{1}{\|\mathbf{w}\|} \min_n z_n(\mathbf{w}^\top \mathbf{x}_n + b) \right) \end{aligned} \tag{23}$$

Poiché i parametri $\mathbf{w}, b$ di un iperpiano Π_f sono definiti a meno di una costante moltiplicativa, possiamo rimuovere tale grado di libertà in modo che per il punto più vicino a Π_f valga esattamente $z_n(\mathbf{w}^\top \mathbf{x}_n + b) = 1$, per cui in generale avremo

$$z_n(\mathbf{w}^\top \mathbf{x}_n + b) = z_n f(\mathbf{x}_n) \geqslant 1 \quad \forall \, n \tag{24}$$

Il problema in Eq. (23) si riduce quindi a massimizzare $1/\|\mathbf{w}\|$ con il vincolo di Eq. (24). Per consuetudine, il problema viene riformulato come segue[10]

$$\hat{\Pi}_f = \arg\min_{\mathbf{w}} \overbrace{\frac{1}{2}\|\mathbf{w}\|^2}^{c(\mathbf{w})} \tag{25a}$$

$$\text{t. c. } \underbrace{1 - z_n f(\mathbf{x}_n)}_{g_n(\mathbf{w},b)} \leqslant 0 \quad \forall \, n \tag{25b}$$

ossia in termini di un problema di ottimizzazione vincolata, esprimibile come caso particolare delle *condizioni di Karush-Kuhn-Tucker* (KKT) per cui sappiamo che esistono dei moltiplicatori di Lagrange λ_n tali che per il punto di minimo in $\hat{\mathbf{w}}$ e $\hat{b}$ per Eq. (25) deve valere

$$\nabla_{\mathbf{w},b} \overbrace{\left(c(\hat{\mathbf{w}}) + \boldsymbol{\lambda}^\top \mathbf{g}(\hat{\mathbf{w}}, \hat{b})\right)}^{L(\mathbf{w},b,\boldsymbol{\lambda})} = \mathbf{0} \tag{26a}$$

$$g_n(\hat{\mathbf{w}}, \hat{b}) \leqslant 0 \qquad \forall \, n \tag{26b}$$

$$\lambda_n g_n(\hat{\mathbf{w}}, \hat{b}) = 0 \qquad \forall \, n \tag{26c}$$

$$\lambda_n \geqslant 0 \qquad \forall \, n \tag{26d}$$

posti $\boldsymbol{\lambda} = [\lambda_1 \dots \lambda_N]^\top$ e $\mathbf{g}(\mathbf{w}, b) = [g_1(\mathbf{w}, b) \dots g_N(\mathbf{w}, b)]^\top$. Si noti che per ciascun punto $\mathbf{x}_n$ deve valere o che $\lambda_n = 0$ e $g_n(\mathbf{w}, b) < 0$, oppure $\lambda_n > 0$ e $g_n(\mathbf{w}, b) = 0$: i punti $\mathbf{x}_n$ per i quali $\lambda_n > 0$ sono proprio i *vettori di suppurto* [2, p. 330].

Da Eq. (26a), posto a $\mathbf{0}$ il gradiente di $\nabla_{\mathbf{w},b} L(\mathbf{w}, b, \boldsymbol{\lambda})$, abbiamo che [13, pp. 886-887]

$$\frac{\partial L}{\partial \mathbf{w}} = \mathbf{w} - \sum_n \lambda_n z_n \mathbf{x}_n = \mathbf{0} \tag{27a}$$

$$\frac{\partial L}{\partial b} = \sum_n \lambda_n z_n = 0 \tag{27b}$$

che riportate in $L(\mathbf{w}, b, \boldsymbol{\lambda})$ producono la funzione Lagrangiana duale

$$\begin{aligned} \mathcal{L}(\boldsymbol{\lambda}) &= \sum_{i=1}^N \lambda_i - \frac{1}{2} \sum_{j=1}^N \sum_{k=1}^N \lambda_j z_j (\mathbf{x}_j^\top \mathbf{x}_k) z_k \lambda_k \\ &= \mathbf{1}^\top \boldsymbol{\lambda} - \frac{1}{2} \boldsymbol{\lambda}^\top \operatorname{diag}(\mathbf{z}) \, \mathtt{G} \operatorname{diag}(\mathbf{z}) \, \boldsymbol{\lambda} \end{aligned} \tag{28}$$

(dove $\mathbf{z} = [z_1 \dots z_N]^\top$ e $\mathtt{G}$ è la *matrice di Gram* con elementi $G_{ij} = \mathbf{x}_i^\top \mathbf{x}_j$), arrivando al *problema duale* [2, p. 329]

$$\hat{\boldsymbol{\lambda}} = \arg\max_{\boldsymbol{\lambda}} \mathcal{L}(\boldsymbol{\lambda}) \tag{29a}$$

$$\text{t. c. } \lambda_n \geqslant 0 \tag{29b}$$

$$\boldsymbol{\lambda}^\top \mathbf{z} = 0 \tag{29c}$$

che dipende solo dai moltiplicatori di Lagrange in $\boldsymbol{\lambda}$ (per una motivazione su Eq. (29) si veda l'appendice § B).

La soluzione $\hat{\boldsymbol{\lambda}}$ per Eq. (29) permette di ricavare $\hat{\mathbf{w}}$ da Eq. (27a), cioè

$$\hat{\mathbf{w}} = \sum_n \hat{\lambda}_n z_n \mathbf{x}_n \tag{30}$$

[10] Anche se in Eq. (25a) i parametri d'interesse sono solo quelli in $\mathbf{w}$, il termine b è implicitamente definiti dai vincoli in Eq. (25b).

per poi ottenere $\hat{b}$ come [13, p. 887]

$$\hat{b} = \frac{\sum_n \hat{\lambda}_n (z_n - \hat{\mathbf{w}}^\top \mathbf{x}_n)}{\sum_n \hat{\lambda}_n}$$

Poiché $\lambda_n > 0$ solo per i vettori di supporto, solo tali punti concorrono a definire $\hat{\mathbf{w}}$ e $\hat{b}$.

B. Caso linearmente non separabile

Il problema *duale* per per Eq. (9) diventa [13, p. 889]

$$\hat{\boldsymbol{\lambda}} = \arg\max_{\boldsymbol{\lambda}} \mathcal{L}(\boldsymbol{\lambda}) \tag{31a}$$

$$\text{t.c. } 0 \leqslant \lambda_n \leqslant \gamma \quad \forall n \tag{31b}$$

$$\boldsymbol{\lambda}^\top \mathbf{z} = 0 \tag{31c}$$

sempre con $\mathcal{L}(\boldsymbol{\lambda}) = \mathbf{1}^\top \boldsymbol{\lambda} - \frac{1}{2} \boldsymbol{\lambda}^\top \operatorname{diag}(\mathbf{z})\, \mathtt{G}\, \operatorname{diag}(\mathbf{z})\, \boldsymbol{\lambda}$ di Eq (28).

Una volta ricavato $\hat{\boldsymbol{\lambda}}$, la soluzione per $\hat{\mathbf{w}}$ è sempre data da Eq. (30) mentre $\hat{b}$ è calcolabile come

$$\hat{b} = \frac{\sum_n \lambda_n (\gamma - \lambda_n)\, (z_i - \hat{\mathbf{w}}^\top \mathbf{x}_n)}{\sum_n \lambda_n (\gamma - \lambda_n)} \tag{32}$$

1) Uso dei kernel: Nella formulazione più generale, applicando $\phi(\mathbf{x})$, notiamo che $\hat{\mathbf{w}}$ è sempre dato da Eq. (30) con $\phi(\mathbf{x})$ al posto di $\mathbf{x}$

$$\hat{\mathbf{w}} = \sum_n \hat{\lambda}_n z_n \phi(\mathbf{x}_n) \tag{33}$$

per cui [13, p. 891]

$$\begin{aligned} f(\mathbf{x}) &= \hat{\mathbf{w}}^\top \phi(\mathbf{x}) + \hat{b} = \sum_n \left(\hat{\lambda}_n z_n \phi(\mathbf{x}_n)^\top \phi(\mathbf{x})\right) + \hat{b} \\ &= \sum_n \left(\hat{\lambda}_n z_n \kappa(\mathbf{x}_n, \mathbf{x})\right) + \hat{b} \end{aligned} \tag{34}$$

$$\begin{aligned} \hat{b} &= \frac{\sum_n \lambda_n (\gamma - \lambda_n)\, (z_n - \hat{\mathbf{w}}^\top \phi(\mathbf{x}_n))}{\sum_n \lambda_n (\gamma - \lambda_n)} \\ &= \frac{\sum_n \lambda_n (\gamma - \lambda_n)\, (z_n - \hat{\lambda}_n \sum_m z_m \kappa(\mathbf{x}_m, \mathbf{x}_n))}{\sum_n \lambda_n (\gamma - \lambda_n)} \end{aligned} \tag{35}$$

Appendice B
La coppia primale-duale

Dato il seguente problema di ottimizzazione vincolata, con $f(\mathbf{x})$, $g_i(\mathbf{x})$, $h_j(\mathbf{x})$ continue con derivata continua (ossia $f(\mathbf{x}), g_i(\mathbf{x}), h_j(\mathbf{x}) \in C^1(\mathbb{R}^N)$),

$$\min_{\mathbf{x} \in \mathbb{R}^N} f(\mathbf{x}) \tag{36a}$$

$$\text{t.c. } g_i(\mathbf{x}) \leqslant 0, \qquad i = 1 \dots m \tag{36b}$$

$$h_j(\mathbf{x}) = 0, \qquad j = 1 \dots p \tag{36c}$$

sia $I_0(\mathbf{x}) = \{i : g_i(\mathbf{x}) = 0\}$ l'insieme dei *vincoli attivi* $g_i(\mathbf{x})$.

Se $\hat{\mathbf{x}}$ è un punto di minimo ammissibile per Eq. (36) e se vale che $\nabla g_i(\hat{\mathbf{x}}) = 0$ (con $i \in I_0(\hat{\mathbf{x}})$) e $\nabla h_j(\hat{\mathbf{x}}) = 0$ sono linearmente indipendenti, allora esistono $\hat{\lambda}_i \geqslant 0$ e $\hat{\mu}_j \in \mathbb{R}$ $(i = 1 \dots m, j = 1 \dots p)$ tali che

$$\hat{\lambda}_i g_i(\hat{\mathbf{x}}) = 0, \qquad i = 1 \dots m \tag{37a}$$

$$\nabla_{\mathbf{x}} \underbrace{\left(f(\hat{\mathbf{x}}) + \sum_{i=1}^{m} \hat{\lambda}_i g_i(\hat{\mathbf{x}}) + \sum_{j=1}^{p} \hat{\mu}_j h_j(\hat{\mathbf{x}})\right)}_{L(\hat{\mathbf{x}}, \hat{\boldsymbol{\lambda}}, \hat{\boldsymbol{\mu}})} = \mathbf{0} \tag{37b}$$

$L(\mathbf{x}, \boldsymbol{\lambda}, \boldsymbol{\mu})$ prende il nome di *funzione Lagrangiana*.

Definendo la *funzione Lagrangiana duale*

$$\mathcal{L}(\boldsymbol{\lambda}, \boldsymbol{\mu}) = \inf_{\mathbf{x}} L(\mathbf{x}, \boldsymbol{\lambda}, \boldsymbol{\mu})$$

possiamo inoltre scrivere

$$\begin{aligned} \mathcal{L}(\boldsymbol{\lambda}, \boldsymbol{\mu}) \quad \leqslant \quad & \mathcal{L}(\boldsymbol{\lambda}, \boldsymbol{\mu}) \text{ t.c. } \overset{(\lambda_i \geqslant 0)}{\leqslant} \quad \inf_{\mathbf{x}} f(\mathbf{x}) \text{ t.c.} \\ & g_i(\mathbf{x}) \leqslant 0 \qquad\quad g_i(\mathbf{x}) \leqslant 0 \\ & h_j(\mathbf{x}) = 0 \qquad\quad h_j(\mathbf{x}) = 0 \end{aligned}$$

e poiché λ_i, μ_j sono arbitrari (purché $\lambda_i \geqslant 0$) allora vale anche

$$\begin{aligned} \sup_{\lambda_i \geqslant 0, \mu} \mathcal{L}(\boldsymbol{\lambda}, \boldsymbol{\mu}) \quad \leqslant \quad & \sup_{\lambda_i \geqslant 0, \mu} \mathcal{L}(\boldsymbol{\lambda}, \boldsymbol{\mu}) \leqslant \quad \inf_{\mathbf{x}} f(\mathbf{x}) \text{ t.c.} \\ & g_i(\mathbf{x}) \leqslant 0 \\ & h_j(\mathbf{x}) = 0 \end{aligned}$$

Riassumendo, per la coppia primale-duale abbiamo che

$$\begin{aligned} \sup_{\lambda_i, \mu_j} \mathcal{L}(\boldsymbol{\lambda}, \boldsymbol{\mu}) \text{ t.c.} \quad & \leqslant \inf_{\mathbf{x}} f(\mathbf{x}) \text{ t.c.} \\ \lambda_i \geqslant 0 \qquad & g_i(\mathbf{x}) \leqslant 0 \\ & h_j(\mathbf{x}) = 0 \end{aligned}$$

Quando $\sup \mathcal{L}(\hat{\boldsymbol{\lambda}}, \hat{\boldsymbol{\mu}}) = \inf f(\hat{\mathbf{x}})$ (con $\hat{\lambda}_i \geqslant 0$, $g_i(\hat{\mathbf{x}}) \leqslant 0$ e $h_j(\hat{\mathbf{x}}) = 0$), allora $\hat{\mathbf{x}}$, $\hat{\boldsymbol{\lambda}}$ e $\hat{\boldsymbol{\mu}}$ individuano i punti di ottimo per i rispettivi problemi; ciò accade se $f(\mathbf{x})$ e le $g_i(\mathbf{x})$ sono convessse, e le $h_j(\mathbf{x})$ sono affini (come nei casi incontrati per le varie tipologie di SVM).

Riferimenti bibliografici

[1] Silvana Abeasis. *Algebra lineare e geometria*. Zanichelli editore, Bologna, 1990.

[2] Christopher M. Bishop. *Pattern Recognition and Machine Learning (Information Science and Statistics)*. Springer, 1 edition, 2007.

[3] Olivier Chapelle and Surendra Keerthi. Efficient algorithms for ranking with svms. *Information Retrieval*, 13:201–215, 06 2010.

[4] Anneleen Daemen and Bart De Moor. Development of a kernel function for clinical data. In *2009 Annual International Conference of the IEEE Engineering in Medicine and Biology Society*, 2009.

[5] Ludger Evers and Claudia-Martina Messow. Sparse kernel methods for high-dimensional survival data. *Bioinformatics*, 24(14):1632–1638, 2008.

[6] Césaire JK Fouodo, Inke R König, Claus Weihs, Andreas Ziegler, and Marvin N Wright. Support vector machines for survival analysis with r. *R Journal*, 10(1), 2018.

[7] Ralf Herbrich, Thore Graepel, and Klaus Obermayer. Large margin rank boundaries for ordinal regression. In P. J. Bartlett, B. Schölkopf, D. Schuurmans, and A. J. Smola, editors, *Advances in Large Margin Classifiers*, pages 115–132. MIT Press, 2000.

[8] Thorsten Joachims. Optimizing search engines using clickthrough data. In *Proceedings of the Eighth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '02, pages 133–142, New York, NY, USA, 2002. ACM.

[9] Tzu-Ming Kuo, Ching-Pei Lee, and Chih-Jen Lin. Large-scale kernel rankSVM. In *Proceedings of SIAM International Conference on Data Mining (SDM)*, 2014.

[10] Ching-Pei Lee and Chih-Jen Lin. Large-scale linear rankSVM. *Neural Computation*, 26(4):781–817, 2014.

[11] Sebastian Pölsterl, Nassir Navab, and Amin Katouzian. Fast training of support vector machines for survival analysis. In *Machine Learning and Knowledge Discovery in Databases*, Lecture Notes in Computer Science, 2015.

[12] Sebastian Pölsterl, Nassir Navab, and Amin Katouzian. An Efficient Training Algorithm for Kernel Survival Support Vector Machines. In *3rd Workshop on Machine Learning in Life Sciences*, September 2016.

[13] William H. Press, Saul A. Teukolsky, William T. Vetterling, and Brian P. Flannery. *Numerical Recipes 3rd Edition: The Art of Scientific Computing*. Cambridge University Press, 3 edition, 2007.

[14] R. Rojas. *Neural Networks - A Systematic Introduction*. Springer-Verlag, 1996. https://page.mi.fu-berlin.de/rojas/neural/neuron.pdf.

[15] Stuart J. Russell and Peter Norvig. *Artificial Intelligence: A Modern Approach (2nd Edition)*. Prentice Hall, 2002.

[16] Pannagadatta K. Shivaswamy, Wei Chu, and Martin Jansche. A support vector approach to censored targets. In *Seventh IEEE International Conference on Data Mining*. IEEE Computer Society, 2007.

[17] David Sontag. Lecture 6: Physiological time-series. In *6.S897/HST.956 Machine Learning for Healthcare*. MIT, Cambridge MA, 2019. MIT OpenCourseWare (license: Creative Commons BY-NC-SA 4.0.). https://ocw.mit.edu/courses/6-s897-machine-learning-for-healthcare-spring-2019/resources/mit6_s897s19_lec6/.

[18] Vanya Van Belle, Kristiaan Pelckmans, Sabine Van Huffel, and Johan A. K. Suykens. Support vector methods for survival analysis: a comparison between ranking and regression approaches. *Artificial Intelligence in Medicine*, 53(2):107–118, 2011.

[19] Ping Wang, Yan Li, and Chandan K. Reddy. Machine learning for survival analysis: A survey. *ACM Computing Surveys*, 51(6), 2019.

[20] Hwanjo Yu and Sungchul Kim. Svm tutorial — classification, regression and ranking. In Grzegorz Rozenberg, Thomas Bäck, and Joost N. Kok, editors, *Handbook of Natural Computing*, pages 479–506. Springer Berlin Heidelberg, Berlin, Heidelberg, 2012.