

الحمد لله
الذي هدانا لهذا
الذي كنا لنهتدي لولا
أن هدانا الله



دانشکده علوم ریاضی

رشته ریاضی کاربردی، گرایش تحقیق در عملیات

پایان نامه کارشناسی ارشد

یک کاربرد از روش‌های بهینه‌سازی بردار پشتیبان کمترین مربعات برای حل رده‌ای از معادلات دیفرانسیل

نگارنده: فاطمه برزمینی

استاد راهنما

دکتر علیرضا ناظمی

بهمن ماه ۱۳۹۷

ماحصل آموخته‌هایم را تقدیم می‌کنم به آنان که مهر آسمانی‌شان
آرام‌بخش آلام زمینی‌ام است.
به استوارترین تکیه‌گاهم، روح پر مهر پدرم به سرسبزترین نگاه زندگیم،
چشمان زیبای مادرم
ره‌آوردی گران سنگ‌تر ازین ارزان نداشتم تا به خاک پایتان نثار کنم.
بوسه بر دستان پر مه‌رت مادر عزیزم.

سپاس‌گزاری...

اینک که به فضل خداوند این پروژه به سرانجام رسیده، وظیفه خود می‌دانم از زحمات بی‌دریغ استاد گرامی جناب آقای دکتر علیرضا ناظمی که در اجرای این پروژه مرا یاری فرموده‌اند، کمال تشکر و قدردانی را داشته باشم. همواره از خداوند متعال سلامتی و توفیق روز افزون شما را خواهانم.

فاطمه برزمینی
بهمن ماه ۱۳۹۷

تعهد نامه

اینجانب فاطمه برزمینی دانشجوی کارشناسی ارشد رشته ریاضی کاربردی دانشکده علوم ریاضی دانشگاه صنعتی شاهرود، نویسنده پایان نامه با عنوان یک کاربرد از روش های بهینه سازی بردار پشتیبان کمترین مربعات برای حل ردهای از معادلات دیفرانسیل، تحت راهنمایی علیرضا ناظمی متعهد می شوم:

- تحقیقات در این پایان نامه توسط اینجانب انجام شده است و از صحت و اصالت برخوردار است.
- در استفاده از نتایج پژوهش های دیگر پژوهش گران، به مرجع مورد استفاده استناد شده است.
- مطالب این پایان نامه، تا کنون توسط خود، یا فرد دیگری برای دریافت هیچ نوع مدرک یا امتیازی در هیچ جا ارایه نشده است.
- حقوق معنوی این اثر، به دانشگاه صنعتی شاهرود تعلق دارد، و مقالات مستخرج با نام “دانشگاه صنعتی شاهرود” یا “Shahrood University of Technology” به چاپ خواهد رسید.
- حقوق معنوی تمام افرادی که در به دست آوردن نتایج اصلی پایان نامه تاثیرگذار بوده اند، در مقالات مستخرج از پایان نامه رعایت می گردد.
- در تمام مراحل انجام این پایان نامه، در مواردی که از موجود زنده (یا بافت های آنها) استفاده شده است، ضوابط و اصول اخلاقی رعایت شده است.
- در تمام مراحل انجام این پایان نامه، در مواردی که به حوزه اطلاعات شخصی افراد دسترسی یافته (یا استفاده شده است)، اصل رازداری و اصول اخلاق انسانی رعایت شده است.

فاطمه برزمینی

بهمن ماه ۱۳۹۷

مالکیت نتایج و حق نشر

- تمام حقوق معنوی این اثر و محصولات آن (مقالات مستخرج، کتاب، برنامه های رایانه ای، نرم افزارها و تجهیزات ساخته شده) متعلق به دانشگاه صنعتی شاهرود می باشد. این مطلب باید به نحو مقتضی، در تولیدات علمی مربوطه ذکر شود.
- استفاده از اطلاعات و نتایج موجود در این پایان نامه بدون ذکر منبع مجاز نمی باشد.

چکیده

در فصل دوم از این پایان نامه، یک روش جدید مبتنی بر روش ماشین‌های بردار پشتیبان کمترین مربعات برای حل معادلات دیفرانسیل معمولی خطی و غیرخطی پیشنهاد شده است. جواب تقریبی با استفاده از LS-SVM ارائه شده است که پارامترهای آن‌ها برای به حداقل رساندن یک تابع خطا تنظیم می‌شود. این پارامترها با حل یک سیستم معادلات خطی و غیرخطی به دست می‌آیند. این روش برای حل معادلات دیفرانسیل معمولی با شرایط اولیه و مرزی مناسب است. نتایج عددی نشان‌دهنده کارایی روش پیشنهادی در مقایسه با روش‌های موجود است. ماشین‌های بردار پشتیبانی در تشخیص الگو و مسأله‌های تخمین تابع بسیار موفق بوده‌اند. در فصل سوم از این پایان نامه استفاده از ماشین‌های بردار پشتیبانی کمترین مربعات را برای کنترل بهینه سیستم‌های غیرخطی با کنترل بازخورد معرفی می‌کنیم. جواب با مجموعه‌ای از معادلات غیرخطی مشخص می‌شود. نتایج برای ماشین‌های بردار پشتیبان با تابع هسته پایه شعاعی مورد بحث قرار گرفته است. مزایای استفاده از کنترل LS-SVM این است که هیچ تعداد لایه پنهان برای کنترل کننده تعیین نمی‌شود و در هنگام استفاده از شرایط مرسر، هیچ مرکزی نباید برای هسته‌های گاوسی مشخص شود. این در حالی است که مسیر، متغیرهای حالت را به عنوان مجهولات اضافی در مسأله بهینه سازی می‌گیرد، در حالی که روش‌های شبکه‌های عصبی کلاسیک به طور معمول، منجر به مسائل بهینه سازی پارامتری می‌شود. در روش SVM تعداد مجهولات برابر با تعداد داده‌های آموزش است.

کلمات کلیدی: ماشین‌های بردار پشتیبان کمترین مربعات، معادلات دیفرانسیل معمولی، فرم بسته جواب‌های تقریبی، روش هم‌محلی، کنترل بهینه عصبی، ماشین بردار پشتیبان، توابع پایه شعاعی.

فهرست مطالب

س فهرست تصاویر

۱	۱ شناسایی الگو و مفاهیم مقدماتی
۲	۱.۱ نظریه یادگیری به وسیله سرپرست
۲	۱.۱.۱ مدل سازی مسأله آموزش
۳	۲.۱ مسأله بهینه سازی
۶	۳.۱ روش لاگرانژ
۸	۴.۱ شرایط کان-تاکر
۱۰	۵.۱ ماشین بردار پشتیبان
۱۱	۶.۱ رگرسیون با استفاده از ماشین بردار پشتیبان
۱۱	۱.۶.۱ رگرسیون چیست؟
۱۴	۲.۶.۱ استفاده از ماشین بردار پشتیبان در رگرسیون

۲ حل تقریبی معادلات دیفرانسیل معمولی با ماشین بردار پشتیبان کمترین مربعات

۱۹	۱.۲ مقدمه
۱۹	۲.۲ بیان مسأله
۲۱	۳.۲ ساخت روش برای ODE خطی
۲۳	۴.۲ حالت BVP
۲۹	۵.۲ اجرای عملیات و انتخاب مدل
۳۷	۶.۲ نتایج عددی

۳ حل مسأله کنترل بهینه با ماشین بردار پشتیبان کمترین مربعات

۴۵	۱.۳ مقدمه
۴۵	۲.۳ مسأله کنترل مطلوب N - مرحله
۴۷	۳.۳ روش بردار پشتیبان برای تخمین تابع

۴.۳	کنترل بهینه با ماشین‌های بردار پشتیبان	۵۰
۵.۳	نمونه‌های شبیه‌سازی	۵۳
۱.۵.۳	مثال ۱:	۵۳
۶.۳	نتیجه‌گیری	۵۴

۵۷	مراجع
----	-------

فهرست تصاویر

۵	۱.۱	نمونه‌ای از یک تابع محدب.
۱۲	۲.۱	نمودار پراکندگی داده‌ها.
۱۳	۳.۱	برآورد خط رگرسیونی.
۱۶	۴.۱	برآورد خط رگرسیونی با استفاده از <i>SVM</i>
	۱.۲	نتایج عددی برای مسأله ۱. (a) ۱۰ نقطه با فاصله مساوی در بازه $[0, 10]$ برای آموزش استفاده شده‌است. (b) ۲۵ نقطه با فاصله مساوی در بازه $[0, 10]$ برای آموزش استفاده شده‌است. (c) نقاط غیریکنواخت در بازه $[0, 10]$ ، ۱۰ نقطه برای آموزش استفاده شده‌است. (d) خطاهای مطلق به‌دست آمده در بازه $[0, 12]$ زمانی که $[0, 10]$ به ۹ قسمت مساوی تقسیم می‌شوند. (e) خطاهای مطلق به‌دست آمده در بازه $[0, 12]$ زمانی که $[0, 10]$ به ۲۴ قسمت مساوی تقسیم می‌شوند. (f) خطاهای مطلق به‌دست آمده در بازه $[0, 12]$ زمانی که $[0, 10]$ به ۹ قسمت غیریکنواخت تقسیم می‌شوند.
۴۲	۲.۲	(a) نتایج عددی برای مسأله ۲، ۲۰ نقطه با فاصله مساوی در بازه $[0, 10]$ برای آموزش استفاده شده‌است. (b) نتایج عددی برای مسأله ۳، ۲۰ نقطه با فاصله مساوی در بازه $[0, 5]$ برای آموزش استفاده شده‌است. (d) نتایج عددی برای مسأله ۴، ۱۰ نقطه با فاصله مساوی در بازه $[0, 1]$ برای آموزش استفاده شده‌است.
۴۲	۵.۲	a. باقیمانده $y(t) - \hat{y}(t)$ وقتی مسأله ۱ روی بازه $[0, 10^5]$ حل می‌شود. با استفاده از ۵۰۰۰ زیربازه و ۵۰ نقطه هم‌محلی. b. نتایج به‌دست آمده $y(t) - \hat{y}(t)$ برای برخی مسائل با استفاده از ۵۰۰ زیربازه و ۵۰۰۰ نقطه هم‌محلی.
۴۳	۳.۲	(a) نتایج عددی برای مسأله ۵، ۱۰ نقطه با فاصله مساوی در بازه $[0, 1]$ برای آموزش استفاده شده‌است. (b) نتایج عددی برای مسأله ۶، ۱۰ نقطه با فاصله مساوی در بازه $[0, 2]$ برای آموزش استفاده شده‌است. (c) نتایج عددی برای مسأله ۷، ۱۰ نقطه با فاصله مساوی در بازه $[0, 1]$ برای آموزش استفاده شده‌است.
۴۳		

- ۴.۲ میزان حساسیت نتایج به دست آمده پارامتر مدل σ . $\log_{10}(\sigma)$ برای مسأله ۲ و ۶ رسم شده است. ۴۳
- ۱.۳ کنترل بهینه با ماشین های بردار پشتیبان کمترین مربعات با هسته RBF و استفاده از شرایط مرسر را نشان می دهد. متغیر حالت اول برای ردیابی [خط بریده بریده] متغیر حالت واقعی توسط شبیه سازی سیستم حلقه بسته است. داده های $k = 1, \dots, 20$ برای کنترل با مبدأ به عنوان حالت اولیه مورد استفاده قرار گرفتند. ۵۵

فصل ۱

شناسایی الگو و مفاهیم مقدماتی

اگر تعدادی سیب و گلابی در اختیار شما قرار بدهند و از شما بخواهند که تشخیص دهید هر یک به کدام دسته از دو نوع میوه تعلق دارند، چه می‌کنید؟ این مسأله ظاهراً زیاد پیچیده نیست، هرکسی می‌تواند بر اساس مشاهده به راحتی آن‌ها را از هم جدا کند. به علاوه، خیلی سخت نیست که تکه‌های فاسد، کثیف و بقیه اشیائی که نه سیب هستند و نه گلابی را تشخیص داد.

اگر چه این مسأله تشخیص سیب‌ها و گلابی‌ها از یکدیگر به نظر مشکلی نیست، اما ماشینی کردن این فرآیند کم و بیش پیچیده است. مثلاً چه چیزی باید مبنای تصمیم‌گیری باشد که بگوییم یک شیء «سیب» است یا «گلابی»؟ آیا وزن آن شیء است، اندازه آن یا رنگ آن؟ شاید شکل آن، بو یا طعم آن باشد؟ احتمالاً ترکیبی از همه این ویژگی‌هاست. لحظه‌ای فکر کنید ما اندازه‌هایی از رنگ و بوی یک شیء را داریم، چگونه یک سیب را که روی آن کمی گلی است رده‌بندی می‌شود؟ آیا آن یک سیب است یا یک شیء گلی؟

این همان مسأله رده‌بندی است: تخصیص یک شیء جدید به یک مجموعه از کلاس‌ها (در این جا سیب‌ها یا گلابی‌ها) که از پیش معلوم هستند. رده‌بندی کننده‌ای که باید عمل جداسازی را انجام دهد (یا به هر شیء ورودی یک برچسب خروجی تخصیص دهد)، بر پایه مجموعه‌ای از مثال‌ها است. در این جا واژه‌ی «شیء» کاملاً عمومی است، می‌تواند از «سیب» و «گلابی» تا «ارقام دست نویس»، از «سیگنال‌های گفتاری» تا «صدای موزیک» متغیر باشد [۱۵].

۱.۱ نظریه یادگیری به وسیله سرپرست

آیا ماشین می تواند فقط دستوراتی را که به آن می دهیم انجام دهد؟ در موارد بسیاری از ماشین می خواهیم که وظایفی را انجام دهد که نمی توان آن ها را با الگوریتم شرح داد. این مسأله هنگامی که مدل ریاضی مشخص و قابل دسترسی وجود ندارد یا موقعی که محاسبه راه حل نشدنی یا خیلی گران باشد پیش می آید. حل کردن این مسأله با استفاده از روش های برنامه نویسی کلاسیک غیرممکن است. به عنوان مثال شناسایی اعداد را در دست نوشته های مختلف در نظر بگیرید (این موضوع یکی از مسأله های کلاسیک در زمینه ماشین آموزش است). برای حل این مسأله نیاز داریم مسأله را از یک دیدگاه دیگری بررسی کنیم. احتمالاً می توانیم ماشین را آموزش دهیم به همان شیوه ای که کودکان را آموزش می دهیم با این تفاوت که به آن ها خلاصه مفاهیم و تعاریف و مسائل تئوری را نمی دهیم. اما توجه آن ها را به تابع هایی از ورودی و خروجی جلب می کنیم. چگونه بچه ها خواندن حروف الفبا را یاد می گیرند معلم به آن ها تعریف دقیق و کاربرد حروف را به طور مجزا نمی گوید اما به آن ها مثال هایی را نشان می دهد که این حروف در آن ها به کار رفته اند. شاگرد با دقت و تمرین کردن زیاد این مثال ها یک استنباط کلی از مشخصات آن حرف را پیدا می کند در پایان اگر شاگردان به نوع حروف فکر کنند می توانند لغات را در متن بخوانند و تشخیص دهند. از مقصودهای آشکار و اصلی آموزش ^۱ دسته بندی کردن صحیح مشاهدات آینده است. در این جا به یک سری روش های دسته بندی یا تاثیرهای روشنی از تاثیرشان بر روی داده نیاز داریم. چگونه آموزش به عنوان مثال در مسأله ریاضی فرموله می شود؟ حل این مسأله کار خیلی بزرگی است و کلی محاسبات دارد، زیرا شامل یک سری از مسائل پرزحمت و سخت برنامه نویسی می شود.

۱.۱.۱ مدل سازی مسأله آموزش

مسأله آموزش یک مدل کلی را شرح می دهد که شامل سه قسمت اصلی است، این سه قسمت عبارتند از: تولیدکننده داده ^۲، یک سرپرست ^۳ که یک برچسب d را به داده مربوط می کند و ماشین آموزش که خروجی d را تحت عنوان پاسخ سرپرست ^۴ برمی گرداند. هدف تقلید از سرپرست می باشد، تلاش برای ساختن یک عملکرد که بهترین پیشگویی را از خروجی های سرپرست به دست می آورد که بر پایه داده های تولید شده است. این موضوع با اینکه تلاش می کنیم تا عملکرد سرپرست با پاسخ سرپرست را مشخص کنیم فرق دارد. تولید کننده، محیطی که سرپرست ماشین آموزش باید در آن فعالیت کند را مشخص می کند. در حقیقت بردارهای $x \in \mathbb{R}^m$ را تولید می کند که مستقل از هم هستند و توزیع آن ها

^۱ Learning

^۲ Generator

^۳ Supervisor

^۴ Supervisor response

را با توجه به یک توزیع احتمال $P(x)$ مشخص می‌کند. سرپرست مقدار واقعی یا درست را با توجه به توزیع احتمال شرطی $P(y|x)$ منصوب می‌کند این فرض شامل $y = F(x)$ است که سرپرست یک d ثابتی را به تمام x مربوط می‌کند. در این موضوع با مسأله دسته‌بندی دوتایی ^۵ سروکار داریم. بنابراین خروجی شامل عدد ۱ است که برای مقادیر مثبت و منفی ۱+ و ۱- می‌گیرد. ماشین آموزش با مجموعه‌ای ممکن از نگاشت $x \rightarrow F(x, a)$ تعریف می‌شود به طوری که a عضوی از فضای پارامتری A است که $A \equiv (w, b) \in \mathbb{R}^{m+1}$. به‌عنوان مثال یک ماشین آموزش که با ابر صفحه $\{x : \langle w, x \rangle + b = 0\}$ تعریف می‌شود، موقعیت یک نقطه در $\mathbb{R}^m$ را مشخص می‌کند که $\langle w, x \rangle$ ضرب داخلی دو بردار $w, x \in \mathbb{R}^m$ را نشان می‌دهد. این نتایج در ماشین آموزش به‌صورت زیر است:

$$f(x, (w, b)) = \text{sign}(\langle w, x \rangle + b), (w, b) \in \mathbb{R}^{m+1}.$$

تابع $f : \mathbb{R}^M \rightarrow \{-1, +1\}$ که x را به توی ۱+ و ۱- نگاشت می‌کند تحت عنوان تابع تصمیم یاد می‌شود.

۲.۱ مسأله بهینه‌سازی

می‌توان مسأله آموزش را به‌صورت یک مسأله بهینه‌سازی در نظر گرفت که فرض در این مسأله مینیم کردن یک تابع بحرانی است که این تابع، تابع مخاطره ^۶ می‌باشد [۱]. معمولاً مسأله بهینه‌سازی با یک سری قیدهایی مطرح می‌شود که در تئوری بردار پشتیبان فقط به تابع‌هایی توجه داریم که محدب و درجه دوم هستند و همچنین قیود آن‌ها خطی هستند. با فرض این که تابع هدف محدب است، این مسأله را حل می‌کنیم.

در این بخش روش لاگرانژ را بررسی می‌کنیم که شرط لازم و کافی برای تابعی که مسأله را حل کند فراهم می‌کند. علاوه بر این مسأله دوگانی ^۷ را مورد بررسی قرار می‌دهیم که یک نقش اساسی در مسأله ماشین بردار پشتیبان ایفا می‌کند. پیدایش مسأله لاگرانژ به سال ۱۷۹۷ میلادی برمی‌گردد.

در سال ۱۹۵۱ میلادی این تئوری توسط کان-تاکر ^۸ در مورد قیدها و شرط‌هایی که به فرم نامساوی هستند تعمیم داده شد که این تعمیم تحت عنوان تئوری کان-تاکر مطرح می‌باشد.

تعریف ۱.۲.۱ (مسأله بهینه‌سازی). توابع $f, g_i, i = 1, \dots, k$ و $h_i, i = 1, \dots, m$ که در بازه‌ی

^۵ Binary classification theory

^۶ Risk function

^۷ Duality

^۸ Kahn-Tucker

$\Omega \subseteq R^n$ تعریف شده‌اند در نظر بگیرید:

$$\begin{aligned} & \text{minimize} && f(w) \\ & \text{subject to} && g_i(w) \leq 0, \quad i = 1, \dots, k, \\ & && h_i(w) = 0, \quad i = 1, \dots, m, \\ & && w \in \Omega, \end{aligned} \quad (1.1)$$

که در آن $f(w)$ تابع هدف یا تابع ارزش نامیده می‌شوند و قیدهای g_i قیدهای نامساوی و h_i ها قیدهای تساوی نامیده می‌شوند. برای سادگی در کار این تابع‌ها را برای تمام i ها به صورت برداری زیر می‌نویسیم:

$$h(w) = 0,$$

$$g(w) \leq 0,$$

که در آن $h = (h_1, h_2, \dots, h_m)$ ، $g = (g_1, g_2, \dots, g_k)$. مقدار بهینه f تحت عنوان «مقدار بهینه مسأله» یاد می‌شود. به ناحیه $F = \{w \in \Omega : g(w) \leq 0, h(w) = 0\}$ ناحیه امکان‌پذیر یا ناحیه شدنی می‌گوییم. این ناحیه‌ای است که f روی آن تعریف شده و علاوه بر آن تمام شرایط و قیدها هم در آن صدق می‌کنند.

به جواب $w^* \in F$ که مقدار بهینه مسأله می‌باشد مینیمم سراسری^۹ می‌گوییم اگر و فقط اگر هیچ $w \in F$ وجود نداشته باشد به طوری که $f(w) < f(w^*)$ باشد و به نقطه w^* مینیمم محلی^{۱۰} برای $f(w)$ می‌گوییم اگر وجود داشته باشد $\epsilon > 0$ به طوری که برای هر $w \in \Omega$ که $\|w - w^*\| < \epsilon$ داشته باشیم:

$$f(w) \geq f(w^*).$$

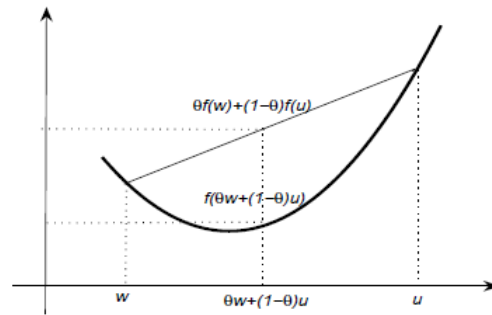
مسأله بهینه‌سازی (۱.۱) را برنامه‌ریزی خطی^{۱۱} می‌نامیم اگر تابع هدف و تمام قیدها و شرایط مسأله خطی باشند و اگر یکی از موارد فوق غیرخطی باشد آن را مسأله برنامه‌ریزی غیرخطی می‌گویند. همچنین به این مسأله برنامه‌ریزی درجه دوم^{۱۲} می‌گوییم اگر تابع هدف به صورت فرم درجه دوم باشد و تمام قیدهای مسأله خطی باشند.

^۹Global minimum

^{۱۰}Local minimum

^{۱۱}Linear programming

^{۱۲}Quadratic programming



شکل ۱.۱: نمونه‌ای از یک تابع محدب.

یک مفهوم مهم در این مسأله قیدهای فعال^{۱۳} هستند. قید g_i را در w فعال گوییم هرگاه $g_i(w) = 0$ باشد. بنابراین اگر w در مرز ناحیه‌شدنی باشد تمام قیدها فعال هستند.

ابتدا تابع محدب و یک سری از خواص آن را روی مجموعه‌ها توضیح می‌دهیم.

تعریف ۲.۲.۱ (محدب بودن). مجموعه $\Omega \subseteq \mathbb{R}^n$ را محدب گوییم هرگاه برای هر $w, u \in \Omega$ و برای هر $\theta \in (0, 1)$ ، داشته باشیم: $(\theta w + (1 - \theta)u) \in \Omega$. به عبارت دیگر، $\Omega \subseteq \mathbb{R}^n$ محدب است اگر برای هر دو عضو دلخواه $w, u \in \Omega$ پاره‌خطی که آن دو را به یکدیگر متصل می‌کند در Ω باشد.

تعریف ۳.۲.۱. تابع $f(w)$ را برای $w \in \mathbb{R}^n$ محدب گوییم هرگاه برای تمام $\theta \in (0, 1)$ رابطه زیر برقرار باشد:

$$f(\theta w + (1 - \theta)u) \leq \theta f(w) + (1 - \theta)f(u).$$

اگر در نامساوی بالا تساوی اتفاق نیفتد تابع مورد نظر اکیداً محدب گوییم.

تعریف ۴.۲.۱. تابعی که برای بعضی ماتریس‌های A و بردارهای b به صورت $f(w) = Aw + b$ تعریف شود تابع آفین^{۱۴} نامیده می‌شود. توابع آفین هم محدب و هم مقعر هستند. مسأله مینیمم کردن توابع محدب بر روی مجموعه‌های محدب تحت عنوان مسأله برنامه‌ریزی محدب^{۱۵} بیان می‌شود. این مسائل خیلی جالب هستند چون در حالت محدب تمام جواب‌های محلی، جواب سراسری مسأله هستند و هم‌چنین جواب سراسری یکتاست در صورتی که تابع هدف و یا یکی از توابع قیود اکیداً محدب باشند.

^{۱۳} Active constraints

^{۱۴} Affin

^{۱۵} Convex programming problem

مسأله ماشین بردار پشتیبان می‌تواند به قیدهای خطی محدود شود و تابع هدف هم به فرم محدب و درجه دوم باشد. به همین دلیل در قسمت بعد فقط به مسائل برنامه‌ریزی درجه دوم می‌پردازیم.

به‌طور کلی، مسأله بهینه‌سازی می‌تواند تحت عنوان یک مسأله مینیمم‌سازی مطرح شود. در واقع مسأله ماکزیمم‌سازی را با ضرب تابع هدف آن در ۱- می‌توان به یک مسأله مینیمم‌سازی تبدیل کرد.

۳.۱ روش لاگرانژ

با یک سری حالت‌های ساده شروع می‌کنیم. در مسائل بهینه‌سازی نامقید که به‌صورت

$$\begin{aligned} & \text{minimize} \quad f(x) \\ & \text{subject to} \quad h_i(x) = 0, \quad i = 1, \dots, m, \end{aligned} \quad (2.1)$$

بیان می‌شوند، مقدار بهینه محلی یا موضعی تابع هدف به سادگی ممکن است با مساوی صفر قرار دادن مشتق تابع $f(x)$ نسبت به متغیرهای اولیه به‌دست می‌آید.

قضیه ۱.۳.۱ ([۵۱]). شرط لازم برای اینکه w^* جواب مینیم $f(w)$ باشد این است که $\frac{\partial f(w^*)}{\partial w} = 0$ باشد. این شرط برای تابع‌های محدب تبدیل به شرط کافی نیز می‌شود.

در مسائل بهینه‌سازی مقید، تابعی تعریف می‌کنیم تحت عنوان تابع لاگرانژ که یک سری اطلاعات واحدی درمورد تابع هدف و قیدهای موجود در مسأله می‌دهد.

تعریف ۱.۳.۱ (تابع لاگرانژ). تابع لاگرانژ متناظر با مسأله (۲.۱) تابعی است که تابع هدف را به همراه ترکیبی خطی از قیدهای کمی $h_i(w) = 0, i = 1, \dots, m$ ، تعریف می‌کند.

$$L(w, \alpha) = f(w) + \sum_{i=1}^m \alpha_i h_i(w),$$

به‌طوری‌که ضرایب α_i ، ضرایب لاگرانژ نامیده می‌شوند.

قضیه ۲.۳.۱. فرض کنیم با مسأله بهینه‌سازی مواجه باشیم که $f(w)$ تابع هدف آن باشد و $h_i(w), i = 1, \dots, m$ ، قیدهای مسأله باشند به‌طوری‌که $f, h_i \in C$ ، آن‌گاه شرط لازم برای این که w^* جواب مسأله بهینه‌سازی باشد به‌صورت زیر است:

$$\begin{aligned} \frac{\partial L}{\partial w}(w^*, \alpha^*) &= 0, \\ \frac{\partial L}{\partial \alpha}(w^*, \alpha^*) &= 0. \end{aligned}$$

این شرط کافی هم هست در صورتی‌که $L(w, \alpha)$ تابعی محدب است. حل این مسأله تبدیل به حل دستگاهی با $n + m$ معادله می‌شود به‌طوری‌که m تعداد شرط‌ها می‌باشد. با حل این

دستگاه معادلات جواب مسأله به دست می آید. توجه داریم که جواب بهینه تمام شرطها را تبدیل به صفر می کند پس مقدار تابع لاگرانژ در جواب بهینه برابر مقدار $f(w^*)$ است، یعنی

$$L(w^*, \alpha^*) = f(w^*).$$

تعریف فوق برای حالتی است که شرایط مسأله بهینه سازی فقط به صورت قیود تساوی هستند. حالا حالتی را در نظر می گیریم که این شرطها به جای این که به فرم تساوی هایی باشند به فرم یک سری نامساوی باشند. این موضوع ما را مجبور می کند تا حالت تعمیم یافته لاگرانژ را تعریف کنیم.

تعریف ۲.۳.۱ (تابع تعمیم یافته لاگرانژ). مسأله کلی بهینه سازی زیر را در نظر بگیرید:

$$\begin{aligned} & \text{minimize} && f(w) \\ & \text{subject to} && g_i(w) \leq 0, \quad i = 1, \dots, k, \\ & && h_j(w) = 0, \quad j = 1, \dots, m, \\ & && w \in \Omega. \end{aligned} \quad (3.1)$$

تابع تعمیم یافته لاگرانژ را به صورت زیر تعریف می کنیم:

$$L(w, \alpha, \beta) = f(w) + \sum_{i=1}^k \alpha_i g_i(w) + \sum_{j=1}^m \beta_j h_j(w) = f(w) + \alpha^T g(w) + \beta^T h(w),$$

که در آن $\alpha^T = (\alpha_1, \dots, \alpha_k)$ آزاد هستند، $\beta_j, j = 1, \dots, m$ و $\alpha_i \geq 0, i = 1, \dots, k$ و $\beta^T = (\beta_1, \dots, \beta_m)$.

تابع لاگرانژ تابعی بر حسب متغیرهای اولیه و دوگان است.

تعریف ۳.۳.۱ (مسأله لاگرانژ دوگان). فرض کنید مسأله اولیه که در تعریف (۱.۳.۱) داشتیم برقرار است. می دانیم که دوگان یک مسأله مینیمم سازی، یک مسأله ماکزیمم سازی است. با قرار دادن شرایط بهینگی در تابع لاگرانژ، تابع هدف مسأله دوگان به صورت تابعی بر حسب متغیرهای دوگان و شرایط مسأله نیز بر حسب متغیرهای دوگان بیان می شوند. مسأله دوگان را به صورت زیر تعریف می کنیم:

$$\max_{\alpha \geq 0} \theta(\alpha, \beta) = \inf_w L(w, \alpha, \beta).$$

در این مسأله مقداری که تابع هدف مسأله دوگان را ماکزیمم کند جواب مسأله است.

قضیه ۳.۳.۱ (قضیه دوگان ضعیف). بین مسأله بهینه سازی اولیه و مسأله دوگان رابطه زیر برقرار است:

$$f(w) \geq \theta(\alpha, \beta).$$

تعریف ۴.۳.۱ (شکاف دوگان). نتیجه مستقیمی که این قضیه می‌دهد این است که مقدار تابع هدف دوگان توسط مقدار تابع هدف اولیه محدود می‌شود یا به عبارت دیگر مقدار تابع هدف اولیه حد بالایی برای مقدار تابع هدف دوگان می‌باشد. تفاوت این دو مقدار تحت عنوان شکاف دوگان شناخته می‌شود. اگر $f(w^*) = \theta(\alpha^*, \beta^*)$ باشد وقتی که قیدها برقرار هستند آن‌گاه w^* و (α^*, β^*) به ترتیب جواب‌های بهینه مسأله اولیه و دوگان هستند. هنگامی که مقادیر توابع هدف مسأله اولیه و دوگان یکسان باشند نامساوی در قضیه قبل به تساوی تبدیل می‌شود و این موضوع نتیجه زیر را حاصل می‌کند:

$$\forall i, \alpha_i^* g_i(w^*) = 0.$$

بنابراین اگر مقدار اولیه را با مقدار مسأله دوگان مقایسه کنیم آن وقت قادر خواهیم بود مسأله بهینه‌سازی را تست کنیم. اگر شکاف دوگان به سمت صفر میل کند می‌توانیم جواب بهینه را پیدا کنیم.

تعریف ۵.۳.۱ (نقطه زینی). بدون توجه به این دو مقدار فوق با تعریف کردن نقطه زینی می‌توانیم یک‌بار دیگر شکاف و مشکل دوگانگی را در نظر بگیریم. نقطه زینی به صورت سه تایی (w^*, α^*, β^*) تعریف می‌شود به طوری که رابطه زیر برقرار کند:

$$L(w^*, \alpha, \beta) \leq L(w^*, \alpha^*, \beta^*) \leq L(w, \alpha^*, \beta^*),$$

که این رابطه برای تمام $w \in \Omega$ برقرار می‌باشد. تابع لاگرانژ تحت w مینیمم می‌شود و تحت (α, β) ماکزیمم می‌شود.

قضیه ۴.۳.۱ ([۲۲]). اگر سه تایی (w^*, α^*, β^*) نقطه زینی برای تابع لاگرانژ اولیه باشد w^* و (α^*, β^*) به ترتیب جواب بهینه مسأله اولیه و دوگان هستند و شکاف دوگان نسبی هم وجود ندارد، یعنی

$$f(w^*) = \theta(\alpha^*, \beta^*).$$

قضیه بعد ضامن این نکته است که مسائل اولیه و دوگان در حالت بهینه دارای مقدار یکسان هستند، که در ماشین بردار پشتیبان از این قضیه برای حل مسائل استفاده می‌کنیم.

قضیه ۵.۳.۱ (قضیه دوگان قوی [۵۷]). مقدار بهینه تابع هدف اولیه با مقدار بهینه تابع هدف دوگان برابر است. یعنی در مسأله بهینه‌سازی (۳.۱) فرض کنید دامنه Ω محدب باشد، در صورتی که مسأله بهینه‌سازی (۳.۱) دارای جواب بهینه باشد آن‌گاه شکاف دوگان برابر صفر است.

۴.۱ شرایط کان-تاکر

قضیه ۱.۴.۱ (کان-تاکر). مسأله بهینه‌سازی اولیه (۳.۱) را در نظر بگیرید. اگر $f \in C^1$ محدب باشد و مجموعه بردارهای g_i که در w^* فعال هستند مستقل خطی باشند. شرط لازم و کافی

برای اینکه w^* جواب بهینه باشد این است که وجود داشته باشد α^* و β^* به طوری که:

$$\begin{aligned}\frac{\partial L}{\partial w}(w^*, \alpha^*, \beta^*) &= 0, \\ \frac{\partial L}{\partial \beta}(w^*, \alpha^*, \beta^*) &= 0, \\ \alpha_i^* g_i(w^*) &= 0, \quad i = 1, \dots, k, \\ g_i(w^*) &\leq 0, \quad i = 1, \dots, k, \\ \alpha_i^* &\geq 0, \quad i = 1, \dots, k.\end{aligned}$$

رابطه $\alpha_i^* g_i(w^*) = 0$ تحت عنوان «شرط مکمل کان-تاکر^{۱۶}» یاد می‌شود. این گزاره بیان می‌کند که جفت α_i^* و $g_i(w^*)$ نمی‌توانند غیر صفر باشند.

برای حل مسأله بهینه‌سازی مجبوریم که لاگرانژ اولیه را مینیمم کنیم، جایی که $\alpha_i g_i$ مجموع مقدار مثبتی دارند، اگر $g_i < 0$ باشد مقدار کل مجموع به وسیله $\alpha_i = 0$ کاهش می‌یابد. یا به عبارت دیگر اگر $g_i = 0$ محدودیتی که باقی می‌ماند $\alpha_i > 0$ است.

اگر هیچ i وجود نداشته باشد به طوری که $\alpha_i^* = g_i(w^*) = 0$ آنگاه گوییم مکمل اکید^{۱۷} برقرار است.

معرفی متغیرهای دوگان ابزار مهمی در اختیار قرار می‌دهند چون حل مسأله بهینه‌سازی دوگان در بسیاری از مواقع آسان‌تر از حل مسأله اولیه است. مسأله دوگان بستگی به متغیرهای اولیه ندارد و تحت قیدهای آسان‌تر باید ماکزیمم گردد.

دوگان را چگونه باید به دست آوریم؟

دوگان یک مسأله در دو مرحله به دست می‌آید ابتدا مشتق لاگرانژ اولیه را نسبت به متغیرهای اولیه صفر قرار می‌دهیم و سپس روابط به دست آمده را در تابع لاگرانژ جایگذاری می‌کنیم. این کار وابستگی به متغیرهای اولیه را از بین می‌برد و جواب وابسته به متغیرهای دوگان به صورت زیر است:

$$\max_{\alpha \geq 0} \theta(\alpha, \beta) = \inf_w L(w, \alpha, \beta).$$

این یک تکنیک استاندارد در تئوری بردار ماشین پشتیبان می‌باشد.

در ماشین بردار پشتیبان روش دوگان با استفاده از روش‌های هسته اجازه کار در فضاهایی با ابعاد بالا در اختیار قرار می‌دهند و شرایط کان-تاکر باعث کاهش مقدار داده‌هایی که در مرحله آموزش با آن‌ها درگیر هستیم، می‌شود. تنها قیدهای فعال دارای متغیرهای دوگان غیر صفر می‌باشند. در نتیجه تعداد متغیرهایی که با آن‌ها در این مرحله درگیر هستیم به میزان چشمگیری کاهش پیدا می‌کند.

^{۱۶}KT complementarity condition

^{۱۷}Strict complementarity

۵.۱ ماشین بردار پشتیبان

مسئله و مفاهیم شناسایی الگو^{۱۸} به عنوان یک فاکتور مهم در طراحی سیستم‌های اطلاعاتی مورد توجه زیادی قرار گرفته است. تئوری شناسایی مبتنی بر روش‌های تحلیل قدرتمندی است که کاربردهای وسیعی در سیستم‌های کنترل و مخابرات دارد. میزان علاقه در این رشته (مخصوصاً در دهه ۶۰ و ۷۰) در حال گسترش بوده و می‌باشد. به طوری که یک موضوع مشترک تحقیقاتی برای رشته‌های آکادمیکی مختلفی از قبیل مهندسی و علوم پایه مانند فیزیک، آمار، شیمی، زبان‌شناسی، روان‌شناسی، بیولوژی، تئوری اطلاعات و علم کامپیوتر و غیره گشته است. هر یک از حوزه‌های تحقیقاتی بر ویژگی‌های خاصی از مسئله شناسایی الگو از قبیل مدل‌سازی فرایندهای فیزیولوژیکی، توسعه تکنیک‌های تحلیلی برای تصمیم‌گیری‌های اتوماتیک تاکید دارند. مسئله شناسایی الگو، عملاً چیزی جز جداسازی داده‌ها یا الگوهای ورودی بین دستجات مختلف نیست. در طراحی یک سیستم شناسایی الگو چند کار باید انجام گیرد: نخست اشیاء مورد شناسایی را باید به الگوها و یا بردارهای ورودی که به سیستم شناسا اعمال می‌گردند تبدیل نمود. مولفه‌های بردار ورودی از طریق اندازه‌گیری به دست می‌آیند. هر کمیت اندازه‌گیری شده، یک ویژگی از شیء مورد نظر را بیان می‌کند. از نظر هندسی هر شیء یا موضوع شناسایی را می‌توان به عنوان یک نقطه در فضای چند بعدی اقلیدسی در نظر گرفت. کار دوم استخراج مشخصه‌های مهم از روی داده‌های ورودی اندازه‌گیری شده و کاهش ابعاد بردار الگوها می‌باشد، به این کار اصطلاحاً استخراج شاخص گویند. مشخصه‌های یک طبقه آن‌هایی هستند که در تمامی الگوهای متعلق به آن طبقه خاص مشترک می‌باشند. عناصری از بردار مشخصه‌ها که در تمامی طبقه‌ها مشترک هستند حذف می‌گردند. سومین کار در شناسایی الگو، این است که روندی جهت تصمیم‌گیری بهینه اتوماتیک، برای طبقه‌بندی الگوها به دست آوریم. به بیان ریاضی مسئله شناسایی الگو به شکل ساده، یعنی تخصیص الگوهای ورودی به یکی از طبقاتی که فضای چندبعدی اقلیدسی برای تصمیم‌گیری به تعداد متناهی از آن‌ها تقسیم شده است. یکی از ایده‌های جدید در شناسایی و دسته‌بندی الگوها، ماشین بردار پشتیبان یا (*SVM*)^{۱۹} است. ماشین بردار پشتیبان دارای خواص بسیار ارزشمندی است که آن را برای شناسایی الگو مناسب می‌سازد از جمله اینکه *SVM* در آموزش خود مشکل بهینه‌های محلی را ندارد، دسته‌بندی کننده‌ها را با حداکثر تعمیم بنا می‌کند. ساختار و توپولوژی خود را به صورت بهینه تعیین می‌کند و توابع تمایز غیرخطی را به راحتی و با محاسبات کم با استفاده از مفهوم حاصل ضرب در فضای هیلبرت تشکیل می‌دهد. اولین الگوریتم برای طبقه‌بندی الگوها در سال ۱۹۳۶ میلادی توسط فیشر^{۲۰} ارائه شد و معیار آن برای بهینه بودن، کم کردن خطای طبقه‌بندی الگوهای آموزشی بود. بسیاری از الگوریتم‌ها و

^{۱۸}Pattern recognition

^{۱۹}Support Vector machine

^{۲۰}Fisher

روش‌هایی نیز که تا کنون برای طبقه‌بندی کننده‌های الگو ارائه شده است از همین استراتژی پیروی می‌کند. در هیچ یک از این روش‌ها خاصیت و تعمیم طبقه‌بندی کننده به طور مستقیم در تابع هزینه‌ی روش دخالت داده نشده است و طبقه‌بندی کننده‌ی طراحی شده نیز دارای خاصیت تعمیم دهنده‌گی کمی می‌باشد. اگر طراحی دسته‌بندی کننده‌ی الگو را به‌عنوان یک مسأله بهینه‌سازی در نظر بگیریم بسیاری از این روش‌ها با مشکل بهینه‌های محلی در تابع هزینه مواجه هستند و در دام بهینه‌های محلی گرفتار می‌آیند. شکل دیگری نیز وجود دارد و آن تعیین ساختار و توپولوژی دسته‌بندی کننده قبل از طراحی است که به‌عنوان مثال تعیین تعداد بهینه‌های گره‌های لایه‌ی مخفی در شبکه‌های عصبی MLP ^{۲۱}، تعداد توابع گاوسی در شبکه‌های عصبی RBF ^{۲۲} و یا تعداد بهینه‌های حالت‌ها و توابع گاوسی در مدل پنهان مارکف می‌باشد. همه این عوامل باعث می‌شوند که در عمل با روش‌های پیشنهاد شده‌ی قبلی نتوانیم به یک دسته‌بندی کننده‌ی بهینه برسیم. محقق روسی به نام ولادمیر وپنیک^{۲۳} در سال ۱۹۶۵ میلادی گامی بسیار مهم در طراحی دسته‌بندی کننده‌ها برداشت و نظریه آماری یادگیری را به‌صورت مستحکم‌تری بنا نهاد و ماشین‌های بردار پشتیبان را بر این اساس ارائه داد. ماشین‌های بردار پشتیبان دارای خواص زیر هستند:

۱. طراحی دسته‌بندی کننده با حداکثر تعمیم
۲. رسیدن به بهینه سراسری تابع هزینه
۳. تعیین خودکار ساختار و توپولوژی بهینه برای طبقه‌بندی کننده
۴. مدل کردن توابع تمایز غیرخطی با استفاده از هسته‌های غیرخطی و مفهوم حاصل ضرب داخلی در فضا‌های هیلبرت.

۶.۱ رگرسیون با استفاده از ماشین بردار پشتیبان

۱.۶.۱ رگرسیون چیست؟

تاریخچه: واژه رگرسیون در فرهنگ لغت به معنی «بازگشت» است و اغلب جهت رساندن مفهوم «بازگشت به یک مقدار متوسط یا میانگین» به کار می‌رود. بدین معنی که برخی پدیده‌ها به مرور زمان از نظر کمی به طرف یک مقدار متوسط میل می‌کنند. بیش از ۱۰۰ سال پیش در سال ۱۸۷۷ میلادی فرانسیس گالتون^{۲۴} در مقاله‌ای که در همین زمینه منتشر کرد اظهار داشت که متوسط قد پسران دارای پدران قدبلند، کمتر از قد پدرانشان

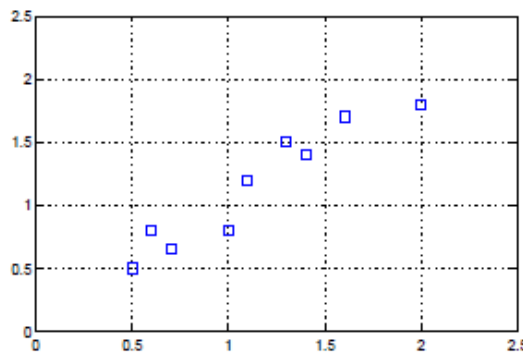
^{۲۱} Multi-layer perceptron

^{۲۲} Radial-basis function

^{۲۳} Veladimir Vapnik

^{۲۴} Francis Galton

می‌باشد. به نحو مشابه متوسط قد پسران دارای پدران قد کوتاه نیز بیشتر از قد پدرانشان گزارش شده است. به این ترتیب گالتون پدیده بازگشت به طرف میانگین را در داده‌هایش مورد تاکید قرار داد. برای گالتون رگرسیون مفهومی زیست شناختی داشت اما کارهای او توسط کارل پیرسون^{۲۵} برای مفاهیم آماری توسعه داده شد. گرچه گالتون برای تاکید بر پدیده «بازگشت به سمت مقدار متوسط» از تحلیل رگرسیون استفاده کرد، اما به هر حال امروزه واژه تحلیل رگرسیون جهت اشاره به مطالعات مربوط به روابط بین متغیرها به کار برده می‌شود.



شکل ۲.۱: نمودار پراکندگی داده‌ها.

نمودار پراکندگی

در حقیقت تحلیل رگرسیونی فن و تکنیکی آماری برای بررسی و مدل‌سازی ارتباط بین متغیرهاست. رگرسیون تقریباً در هر زمینه‌ای از جمله مهندسی، فیزیک، اقتصاد، مدیریت، علوم زیستی، بیولوژی و علوم اجتماعی برای برآورد و پیش‌بینی مورد نیاز است. می‌توان گفت تحلیل رگرسیونی، پرکاربردترین روش در بین تکنیک‌های آماری است. خلاصه‌ای از تحلیل رگرسیونی به صورت زیر است:

در ابتدا تحلیل گر حدس می‌زند که یک رابطه به شکل یک خط بین دو متغیر وجود دارد و سپس به جمع‌آوری اطلاعات کمی از دو متغیر می‌پردازد و این داده‌ها را به صورت نقاطی در یک نمودار دو بعدی رسم می‌کند، (۲.۱).

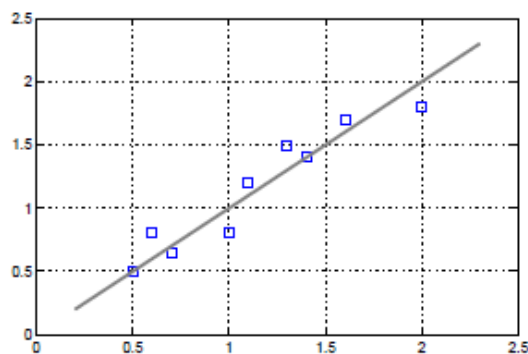
این نمودار که به آن نمودار پراکندگی^{۲۶} گفته می‌شود نقش بسیار مهمی در تحلیل‌های رگرسیونی و نمایش ارتباط بین متغیرها ایفا می‌کند.

در صورتی که، نمودار نشان‌دهنده این باشد که داده‌ها تقریباً (نه لزوماً دقیق) در امتداد یک خط مستقیم پراکنده شده‌اند، حدس تحلیل گر تاییده شده (۳.۱) و این ارتباط خطی به صورت زیر نمایش داده می‌شود:

$$y = ax + b,$$

^{۲۵}Karl Pearson

^{۲۶}Scatter plot



شکل ۳.۱: برآورد خط رگرسیونی.

که در آن b عرض از مبدا و a شیب این خط است.

متغیرها و خطا

بین برخی از نقاط و تصویر آن‌ها بر روی خط رگرسیونی کمی تفاوت به چشم می‌خورد که از آن به عنوان خطای برآورد یاد می‌کنیم. بنابراین معادله اولیه را به صورت زیر اصلاح می‌کنیم:

$$y = ax + b + \varepsilon.$$

معادله فوق یک مدل رگرسیونی خطی نامیده می‌شود.

روش‌های رگرسیونی

تا این مرحله مدل رگرسیونی معرفی شد و کافی است پارامترهای مجهول a و b برآورده شوند. برآورد پارامترها در مدل‌سازی با استفاده از روش‌های مختلف انجام می‌شود از جمله روش کمترین مربعات خطا. روش کمترین مربعات خطا که یکی از روش‌های مورد استفاده در تحلیل رگرسیونی است اولین بار توسط لژاندر^{۲۷} ریاضیدان فرانسوی در سال ۱۸۰۵ میلادی و گاوس^{۲۸} ریاضیدان آلمانی در سال ۱۸۰۹ میلادی معرفی و در مطالعات نجومی به کار برده شد. روش‌های ماشین بردار پشتیبان نیز در رگرسیون بسیار موفقیت‌آمیز بوده است که در بخش بعد به آن اشاره می‌کنیم. روش SVM در رگرسیون اولین بار توسط وپنیک و همکارانش^{۲۹} در سال ۱۹۹۶ میلادی معرفی شد [۲۳]. این روش را رگرسیون بردار پشتیبان (SVR)^{۳۰} می‌نامند.

^{۲۷}Legendre

^{۲۸}Gauss

^{۲۹}Vladimir Vapnik, Harris Drucker, Chris Burges, Linda Kaufman and Alex Smola

^{۳۰}Support Vector Regression

۲.۶.۱ استفاده از ماشین بردار پشتیبان در رگرسیون

برای نمونه‌های آموزشی داده شده $(x_1, y_1), (x_2, y_2), \dots, (x_n, y_n) \in \mathbb{R}^n \times \mathbb{R}$ که $x_i \in \mathbb{R}^n$ نقطه‌ای در فضای ورودی است و y_i متناظر مقدار تابع $f(x_i)$ است. هدف پیدا کردن تابع خطی $f(x)$ که برای همه نمونه‌های آموزشی x_i ، حداکثر فاصله آن از مقدار هدف y_i ، ε باشد. در رگرسیون ε -بردار پشتیبان^{۳۱} [وپنیک، ۱۹۹۵]، هر انحراف که کمتر از ε است، مجاز می‌باشد. تابع $f(x) = \langle w, x \rangle + b$ را در نظر بگیرید که در آن $w \in \mathbb{R}^n, b \in \mathbb{R}$. هدف SVR مینیمم‌سازی نرم اقلیدسی w است به‌قسمی که انحراف $f(x)$ از مقادیر خروجی y_i کمتر از ε باشد. این مسأله بهینه‌سازی محدب به‌صورت زیر بیان می‌شود:

$$\begin{aligned} \text{minimize} \quad & \frac{1}{2} \|w\|^2 + C \sum_{i=1}^n (\xi_i + \xi'_i) \\ \text{subject to} \quad & y_i - \langle w, x_i \rangle - b \leq \varepsilon + \xi_i, \quad i = 1, \dots, n, \\ & \langle w, x_i \rangle + b - y_i \leq \varepsilon + \xi'_i, \quad i = 1, \dots, n, \\ & \xi_i, \xi'_i \geq 0, \quad i = 1, \dots, n. \end{aligned} \quad (۴.۱)$$

توجه کنید که دو پارامتر انحراف ε و C توسط کاربر مشخص می‌شوند. برای ساختن دوگان مسأله بهینه‌سازی (۴.۱) متغیرهای کمکی α_i, α'_i را معرفی می‌کنیم. α_i متناظر محدودیت‌های شامل ξ_i و α'_i متناظر محدودیت کمکی ξ'_i است. این متغیرها در حقیقت ضرایب لاگرانژ برای مسأله اولیه هستند. مشابه روش‌های به‌کار رفته در طراحی رده‌بندی کننده‌های SV ، مسأله بهینه‌سازی مقید فوق را با تشکیل لاگرانژین متغیرهای اولیه حل می‌کنیم که تابع لاگرانژین به‌صورت زیر است:

$$\begin{aligned} L_P(w, b, \xi_i, \xi'_i, \alpha_i, \alpha'_i, \beta_i, \beta'_i) = & \frac{1}{2} w^T w + C \sum_{i=1}^n (\xi_i + \xi'_i) - \sum_{i=1}^n (\beta_i \xi_i + \beta'_i \xi'_i) \\ & - \sum_{i=1}^n \alpha_i (w^T x_i + b - y_i + \varepsilon + \xi_i) - \sum_{i=1}^n \alpha'_i (y_i - w^T x_i - b + \varepsilon + \xi'_i). \end{aligned}$$

یک تابع لاگرانژین $L_P(w, b, \xi_i, \xi'_i, \alpha_i, \alpha'_i, \beta_i, \beta'_i)$ باید نسبت به متغیرهای اولیه w, b, ξ_i, ξ'_i مینیمم شود و نسبت به ضرایب لاگرانژ نامنفی $\alpha_i, \alpha'_i, \beta_i, \beta'_i$ ماکزیمم گردد. بنابراین تابع در نقطه جواب بهینه $(w^*, b^*, \xi_i^*, \xi'_i^*)$ دارای نقطه زینی است.

^{۳۱} ε -SV Regression

در جواب بهینه مشتقات جزئی L_P نسبت به متغیرهای اولیه صفر می‌شود، یعنی:

$$\frac{\partial L_P}{\partial w} = w - \sum_{i=1}^n (\alpha_i - \alpha'_i) x_i = 0, \quad (5.1)$$

$$\frac{\partial L_P}{\partial b} = \sum_{i=1}^n (\alpha_i - \alpha'_i) = 0, \quad (6.1)$$

$$\frac{\partial L_P}{\partial \xi_i} = C - \alpha_i - \beta_i = 0, \quad (7.1)$$

$$\frac{\partial L_P}{\partial \xi'_i} = C - \alpha'_i - \beta'_i = 0. \quad (8.1)$$

با استفاده از شرایط KKT و قضیه مرسر، دوگان مسأله بهینه‌سازی (۴.۱) برای رگرسیون خطی با استفاده از SVM به صورت زیر است:

$$\begin{aligned} \text{maximum} \quad L_d(\alpha, \alpha') &= \sum_{i=1}^n y_i (\alpha - \alpha'_i) - \frac{1}{2} \sum_{i=1}^n \sum_{j=1}^n (\alpha_i - \alpha'_i) (\alpha_j - \alpha'_j) x_i^T x_j - \varepsilon \sum_{i=1}^n (\alpha_i + \alpha'_i) \\ \text{subject to} \quad &\sum_{i=1}^n (\alpha_i - \alpha'_i) = 0 \\ &0 \leq \alpha_i \leq C, \quad i = 1, \dots, n, \\ &0 \leq \alpha'_i \leq C, \quad i = 1, \dots, n. \end{aligned} \quad (9.1)$$

توجه کنید که لاگرانژین $L_d(\alpha, \alpha')$ تنها به صورت عباراتی از ضرایب لاگرانژ α و α' بیان شده است. اگرچه، اندازه مسأله فوق نسبت به اندازه رده‌بندی کننده SVM دو برابر است. در اینجا $2n$ متغیر مجهول برای رگرسیون خطی داریم (n تا α_i و n تا α'_i) و ماتریس هسین H برای مسأله بهینه‌سازی رگرسیون یک ماتریس $2n \times 2n$ است. مسأله بهینه‌سازی درجه دوم استاندارد فوق می‌تواند به صورت شکل ماتریسی زیر فرمول‌بندی شود:

$$\text{minimize} \quad L_d(\alpha) = \frac{1}{2} \alpha^T H \alpha + f^T \alpha,$$

به قسمی که قیود مسأله (۹.۱) برقرار باشند، که در آن $\alpha = (\alpha_1, \alpha_2, \dots, \alpha_n, \alpha'_1, \alpha'_2, \dots, \alpha'_n)$ ، $H = \begin{pmatrix} G & -G \\ -G & G \end{pmatrix}$ که G ماتریسی با مولفه‌های $G_{ij} = [x_i^T x_j]$ (در یک رگرسیون خطی) است و

$$f = [\varepsilon - y_1, \varepsilon - y_2, \dots, \varepsilon - y_n, \varepsilon + y_1, \varepsilon + y_2, \dots, \varepsilon + y_n].$$

(توجه کنید که G_{ij} گفته شده در بالا، یک ماتریس بدوضع^{۳۲} بود و معمولاً به جای آن از $G_{ij} = [x_i^T x_j + 1]$ استفاده می‌شود).

مرحله آموزش n جفت ضریب لاگرانژ (α_i, α'_i) را نتیجه می‌دهد. بعد از مرحله آموزش، تعداد پارامترهای ناصفر α_i یا α'_i مساوی تعداد بردارهای پشتیبان است. از آنجا که حداقل یک

^{۳۲} Badly conditioned matrix

مولفه از هر جفت (α_i, α'_i) ($i = 1, \dots, n$) صفر است، لذا حاصل ضرب α_i و α'_i همیشه صفر است، یعنی $\alpha_i \alpha'_i = 0$.

در بهینگی شرایط مکمل KKT زیر برقرار هستند:

$$\alpha_i(w^T x_i + b - y_i + \varepsilon + \xi_i) = 0, \quad (10.1)$$

$$\alpha'_i(-w^T x_i - b + y_i + \varepsilon + \xi'_i) = 0, \quad (11.1)$$

$$\beta_i \xi_i = (C - \alpha_i) \xi_i = 0, \quad (12.1)$$

$$\beta'_i \xi'_i = (C - \alpha'_i) \xi'_i = 0, \quad (13.1)$$

شرایط فوق نشان می‌دهند که برای $0 < \alpha_i < C$ ، شرط $\xi_i = 0$ برقرار است. به‌طور مشابه اگر $0 < \alpha'_i < C$ ، آنگاه $\xi'_i = 0$ و برای هر $\alpha_i < C$ و $\alpha'_i < C$ داریم:

$$(w^T x_i + b - y_i + \varepsilon) = 0,$$

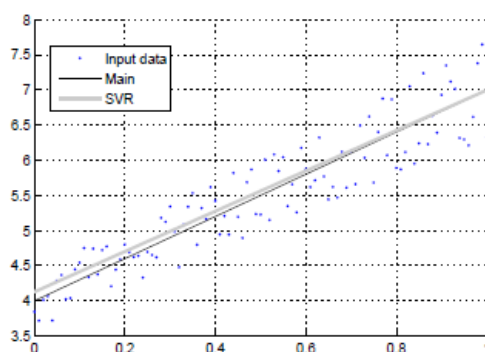
$$(-w^T x_i - b + y_i + \varepsilon) = 0.$$

b به‌صورت زیر محاسبه می‌شود:

$$b = y_i - \langle w, x_i \rangle - \varepsilon, \quad 0 < \alpha_i \leq C, \quad i = 1, \dots, n.$$

$$b = y_i - \langle w, x_i \rangle + \varepsilon, \quad 0 < \alpha_i \leq C \quad i = 1, \dots, n.$$

شرایط (۱۲.۱) و (۱۳.۱) در مورد داده‌هایی هستند که دارای انحراف بیشتر از ε نسبت به خط رگرسیون می‌باشند، یعنی اگر $\xi_i > 0$ ، آنگاه $\alpha_i = C$ و اگر $\xi'_i > 0$ ، آنگاه $\alpha'_i = C$. یا به عبارتی برای داده‌هایی که بالای محدوده با خطای مجاز هستند، $\alpha_i = C$ و برای داده‌هایی که پایین محدوده با خطای مجاز هستند، $\alpha'_i = C$ ، این نقاط را بردارهای پشتیبان مقید می‌نامند. هم‌چنین برای همه نقاط تمرینی داخل ناحیه مجاز، یا جایی که $|y - f(x)| < \varepsilon$ ، هر دوی α_i و α'_i مساوی صفر هستند لذا هیچ‌یک از آن‌ها SV (بردار پشتیبان) نیستند و تاثیری در صفحه تصمیم ندارند.



شکل ۴.۱: برآورد خط رگرسیونی با استفاده از SVM .

اما پس از حل مسأله بهینه‌سازی (۹.۱) و با داشتن ضرایب بهینه α_i و α'_i ، با استفاده از شرط بهینگی (۵.۱) بردار وزنی بهینه برای ابرصفحه رگرسیون به صورت زیر به دست می‌آید:

$$w_{\circ} = \sum_{i=1}^n (\alpha_i - \alpha'_i) x_i. \quad (۱۴.۱)$$

و ابر صفحه بهینه رگرسیون به صورت زیر به دست می‌آید:

$$w_{\circ}^T x + b = \sum_{i=1}^n (\alpha_i - \alpha'_i) x_i^T x + b. \quad (۱۵.۱)$$

در شکل (۴.۱) یک نمونه پراکندگی داده‌ها را ملاحظه می‌کنید. داده‌ها را در امتداد خط $y = 3x + 4$ تولید کرده‌ایم که این خط به رنگ سیاه رسم شده است. با قرار دادن نمونه‌های داده شده در مسأله (۹.۱) به جواب بهینه $(w, b) = (2/87, 4/12)$ می‌رسیم یعنی برآورد خط رگرسیونی با استفاده از SVM به صورت $y = 2/87x + 4/12$ می‌باشد که تقریب بسیار خوبی است.

آنچه بیشتر مورد توجه است، حل مسائل رگرسیون غیرخطی می‌باشد. یک تعمیم به حالت غیرخطی مشابه روش توسعه رده‌بندی کننده غیرخطی با استفاده از رده‌بندی کننده خطی است، یعنی به وسیله یک نگاشت به فضای ویژگی (که معمولاً دارای بعد خیلی بالایی است) داده‌ها را به فضای با قابلیت رگرسیون خطی می‌بریم و سپس تابع رگرسیون غیرخطی را متناظر معادله رگرسیون خطی در فضای ویژگی به دست می‌آوریم. با استفاده از شرایط KKT و قضیه مرسر، دوگان مسأله بهینه‌سازی برای رگرسیون غیرخطی با استفاده از SVM به صورت زیر است:

$$\begin{aligned} \text{maximum} \quad & F(\alpha, \alpha') = \sum_{i=1}^n y_i (\alpha_i - \alpha'_i) \\ & - \frac{1}{2} \sum_{i=1}^n \sum_{j=1}^n (\alpha_i - \alpha'_i) (\alpha_j - \alpha'_j) K(x_i, x_j) - \varepsilon \sum_{i=1}^n (\alpha_i + \alpha'_i) \\ \text{subject to} \quad & \sum_{i=1}^n (\alpha_i - \alpha'_i) = 0 \\ & 0 \leq \alpha_i \leq C, \quad i = 1, \dots, n, \\ & 0 \leq \alpha'_i \leq C, \quad i = 1, \dots, n. \end{aligned} \quad (۱۶.۱)$$

از شرایط بهینگی w به صورت زیر به دست می‌آید:

$$w = \sum_{i=1}^n (\alpha_i - \alpha'_i) \Phi(x_i). \quad (۱۷.۱)$$

بنابراین تابع تخمین $f(x) = \langle w, x \rangle + b$ به صورت زیر بیان می‌شود:

$$f(x) = \sum_{i=1}^n (\alpha_i - \alpha'_i) K(x_i, x) + b. \quad (۱۸.۱)$$

فصل ۲

حل تقریبی معادلات دیفرانسیل معمولی با ماشین بردار پشتیبان کمترین مربعات

۱.۲ مقدمه

معادلات دیفرانسیل در فرمول بندی ریاضی پدیده های فیزیکی و در طیف گسترده ای از برنامه های کاربردی، به ویژه در علوم و مهندسی به کار می رود. بسته به نوع شرایط مرزی که جواب باید در آن برآورده شود، مسائل مربوط به معادلات دیفرانسیل معمولی^۱ را می توان به دو دسته اصلی تقسیم کرد: مسائل مقدار اولیه^۲ و مسائل مقدار مرزی^۳. در حالت کلی، روش های تحلیلی برای حل این مسائل در دسترس نیست و بنابراین روش های عددی باید اعمال شود. روش های متعددی برای حل معادلات دیفرانسیل از قبیل رانگ کوتا^۴، تفاضلات متناهی^۵،

^۱ODE

^۲IVP

^۳BVP

^۴Runge-Kutta

^۵Finite difference

۲۰ حل تقریبی معادلات دیفرانسیل معمولی با ماشین بردار پشتیبان کمترین مربعات

پیش‌بینی کننده اصلاح‌گر^۶ و روش‌های هم‌محلی^۷ توسعه یافته‌اند [۳۷، ۵۶]. به‌طور کلی، روش‌های عددی برای تقریب جواب مسائل مقدار مرزی به دو دسته تقسیم می‌شوند: روش‌های تفاضلی مانند روش پرتابی و روش‌های باقی‌مانده وزنی.

در روش‌های پرتابی فرد تلاش می‌کند تا مسأله را به یک مسأله مقدار اولیه تبدیل کند. این کار به‌وسیله‌ی فراهم کردن یک تقریب تا حد امکان خوب از مقادیر مشتق در نقاط اولیه انجام می‌شود. در مورد معادلات دیفرانسیل مرتبه بالاتر شایع‌ترین رویکرد کاهش مسأله به یک دستگاه از معادلات دیفرانسیل مرتبه اول و سپس حل دستگاه به‌دست آمده به‌وسیله‌ی یکی از روش‌های موجود است، که در مراجع [۳۸، ۱۸، ۱۴] مورد مطالعه قرار گرفته است.

هر چند برخی نویسندگان ادعا کرده‌اند که این روش نیاز به زمان زیادی برای اجرا شدن روی کامپیوتر دارد و دارای پیچیدگی محاسباتی بالایی است [۶۲، ۱۷].

اکثر روش‌های عددی سنتی جواب را در قالب یک آرایه در نقاط خاص از پیش تعیین شده ارائه می‌دهند و به یک روش درونیابی نیز نیاز دارند تا جواب را برای کل دامنه ارائه دهند. برای این‌که یک روش دقیق داشته باشیم، شخص مجبور است یا مرتبه روش را افزایش داده یا اندازه مش بندی را کاهش دهد. این‌ها هر دو هزینه محاسبات را افزایش می‌دهند.

برای غلبه بر این نقص تلاش شده است که روش‌های جدیدی را به‌طور مستقیم و بدون کاهش مسأله به سیستم معادلات دیفرانسیل مرتبه اول ایجاد کنیم. این منجر به ارائه یک جواب تقریبی به‌فرم بسته می‌شود و از این رو از تکرار اضافی اجتناب می‌شود. یکی از این رده‌ها، روش مبتنی بر استفاده از مدل‌های شبکه عصبی است [۲۸، ۳]. لی و کانگ^۸ در [۲۴] از مدل‌های شبکه عصبی هاپفیلد برای حل معادلات دیفرانسیل درجه اول استفاده کردند.

نویسندگان در [۲۹] یک روش مبتنی بر شبکه‌های عصبی پیشرو^۹ برای حل معادلات دیفرانسیل معمولی و جزئی ارائه دادند. در این روش تابع جواب به‌صورت مجموع دو عبارت بیان می‌شود: اولین جمله که حاوی پارامترهای قابل تنظیم نیست، شرایط ابتدایی / مرزی را تضمین می‌کند و جمله دوم شامل یک شبکه عصبی پیش‌بینی شده برای آموزش است.

علی‌رغم این‌که شبکه‌های عصبی کلاسیک ویژگی‌های خوبی مانند تقریب‌کنندگی عمومی دارند، آن‌ها هنوز از دو معضل رنج می‌برند. اولین مشکل، وجود بسیاری از روش‌های یافتن مینیمم محلی است. مشکل دوم این است که چگونه تعداد لایه‌های پنهان را انتخاب کنیم. ماشین‌های بردار پشتیبان (SVM) یک روش قدرتمند برای حل تشخیص الگو و مسائل تخمین عملکرد است [۶۶، ۹].

در این روش، شخص داده‌ها را به یک فضا با بعد بالاتر می‌نگارد و سپس یک مسأله رگرسیون خطی را حل می‌کند. این منجر به حل مسأله برنامه‌ریزی درجه دوم می‌شود. روش حداقل

^۶Predictor-Corrector

^۷Colloncation methods

^۸Lee and Kong

^۹Feed-Forward

مربعات LS-SVM برای تخمین تابع، طبقه‌بندی داده‌ها و مسائلی در یادگیری مورد بررسی قرار گرفته است. [۳۳، ۳۲، ۳۱] در این حالت، فرمول‌بندی مسأله شامل قیود مساوی بجای قیود نامساوی است.

آموزش برای رگرسیون و مسأله طبقه‌بندی داده‌ها با حل مجموعه‌ای از معادلات خطی انجام می‌شود. هدف از این پایان‌نامه معرفی یک رویکرد جدید مبتنی بر LS-SVM برای حل معادلات دیفرانسیل معمولی است.

۲.۲ بیان مسأله

این بخش به بیان مسأله می‌پردازد. در قسمت ”آ” یک مقدمه کوتاه برای معرفی LS-SVM برای رگرسیون داده شده است تا تفاوت را برای مسأله مورد نظر در این فصل برجسته کند. در نهایت، برخی از عملگرها که در بخش‌های بعدی استفاده می‌شوند تعریف می‌شوند. به‌طور کلی، ODE خطی مرتبه m -ام با ضرایب وابسته به زمان را به‌صورت زیر در نظر بگیرید:

$$L[y] = \sum_{l=0}^m f_l(t)y^{(l)}(t) = r(t), \quad t \in [a, c]. \quad (1.2)$$

که در آن L نشان‌دهنده عملگر معادله دیفرانسیل خطی مرتبه m -ام، $[a, c]$ دامنه مسأله است و $r(t)$ سیگنال ورودی است. $f_L(t)$ یک تابع معلوم است. $y^{(l)}(t)$ نشان دهنده مشتق l -ام از y نسبت به t است. یک $m-1$ شرایط اولیه یا مرزی برای حل معادلات دیفرانسیل بالا به‌صورت زیر داده می‌شود.

IVP

$$Ie_\mu[y(t)] = p_\mu, \quad \mu = 0, \dots, m-1;$$

BVP

$$Be_\mu[y(t)] = q_\mu, \quad \mu = 0, \dots, m-1;$$

که در آن Ie_μ شرایط اولیه است و Be_μ شرایط مرزی هستند. p_μ و q_μ اسکالرهایی داده شده‌اند. روش‌های داده شده در [۲۹] و [۲۵] یک جواب آزمایشی را به‌صورت زیر در نظر می‌گیرد:

$$y(t) = H(t) + F(t, N(t, P)),$$

که در آن $H(t)$ شرایط اولیه / مرزی را برآورد می‌کند و دومین عبارت $F(t, N(t, P))$ خروجی یک شبکه عصبی با ورودی t و پارامتر P است. در مقایسه با روش‌های ارائه شده در [۲۹] و

۲۲ حل تقریبی معادلات دیفرانسیل معمولی با ماشین بردار پشتیبان کمترین مربعات

[۲۵] مدل را با ترکیب شرایط اولیه / مرزی به عنوان محدودیت‌های یک مسأله، بهینه‌سازی می‌کنیم. این به‌طور قابل توجهی باعث کاهش پیچیدگی محاسبات می‌شود.

آ: رگرسیون LS-SVM

اجازه دهید، یک مجموعه آموزشی داده شده را $\{x_i, y_i\}_{i=1}^N$ با داده‌های ورودی $x_i \in \mathbb{R}$ و داده‌های خروجی $y_i \in \mathbb{R}$ را در نظر بگیریم. در اینجا، فقط از یک فضای ورودی یک بعدی استفاده می‌کنیم. هدف در یک مسأله رگرسیون برآورد یک مدل به صورت $y(x) = w^T \varphi(x) + b$ است. مدل اولیه LS-SVM برای رگرسیون می‌تواند به صورت زیر نوشته شود [۳۳]:

$$\begin{aligned} \underset{w, b, e}{\text{minimize}} \quad & \frac{1}{2} w^T w + \frac{\gamma}{2} e^T e \\ \text{subject to} \quad & y_i = w^T \varphi(x_i) + b + e_i, \quad i = 1, \dots, N. \end{aligned} \quad (2.2)$$

که در آن $w \in \mathbb{R}^h$ و $b \in \mathbb{R}$ ، $\gamma \in \mathbb{R}^+$ است. سپس جواب دوگان از مسأله (۲.۲) از حل دستگاه زیر به دست می‌آید:

$$\left[\begin{array}{c|c} \Omega + I_{N/\gamma} & \mathbf{1}_N \\ \hline \mathbf{1}_N^T & 0 \end{array} \right] \cdot \left[\begin{array}{c} \alpha \\ b \end{array} \right] = \left[\begin{array}{c} y \\ 0 \end{array} \right] \quad (3.2)$$

که در آن $\Omega_{ij} = K(x_i, x_j) = \varphi(x_i)^T \varphi(x_j)$ ، عنصر i - j ام ماتریس هسته معین مثبت k است. همچنین:

$$\mathbf{1}_N = [1, \dots, 1]^T \in \mathbb{R}^N, \quad \alpha = [\alpha_1, \dots, \alpha_N]^T, \quad y = [y_1, \dots, y_N]^T.$$

و I_N ماتریس همانی است. مدل در فرم دوگان به صورت زیر است:

$$y(x) = \sum_{i=1}^N \alpha_i K(x, x_i) + b.$$

لازم به ذکر است که اگر $b = 0$ ، مسأله با حذف e و در نظر گرفتن w به عنوان مجهول به عنوان رگرسیون ریج^{۱۰} شناخته می‌شود. اما در رویکرد LS-SVM، نگاشت φ در حالت کلی، به صورت صریح شناخته نشده است و می‌تواند دارای بعد بی‌نهایت باشد. بنابراین، ترفند هسته مورد استفاده قرار می‌گیرد و مسأله در حالت دوگان حل می‌شود [۳۲].

هنگامی که با معادلات دیفرانسیل برخورد می‌کنیم، مقادیر هدف y_i به‌طور مستقیم در دسترس نیستند، بنابراین رویکرد رگرسیون به‌طور مستقیم اعمال نمی‌شود. با این وجود، می‌توانیم معادلات دیفرانسیل را در فرآیند یادگیری برای یافتن جواب آن با مسأله رگرسیون تلفیق کنیم.

$$\hat{y}(t) = w^T \varphi(t) + b.$$

^{۱۰} Ridge regression

اکنون مدل صریح را به عنوان تقریبی برای جواب معادله دیفرانسیل در نظر می گیریم. از آن جایی که به منظور یادگیری از معادله دیفرانسیل داده ای وجود ندارد، باید مدل را با معادله دیفرانسیل داده شده جایگزین کنیم. بنابراین، باید مشتق تابع هسته را تعریف کنیم. با استفاده از قضیه مرسر [۶۶]، مشتقات نگاشت طبیعی φ می توانند برحسب مشتقات تابع هسته نوشته شوند [۴۹]. اکنون عملگر دیفرانسیل زیر را که در بخش های بعدی استفاده می شود تعریف می کنیم:

$$\nabla_n^m = \frac{\partial^{n+m}}{\partial u^n \partial v^m}. \quad (۴.۲)$$

اگر $\varphi(u)^T \varphi(v) = k(u, v)$ می توان نشان داد که:

$$[\varphi^{(n)}(u)]^T \varphi^{(m)}(v) = \nabla_n^m [\varphi(u)^T \varphi(v)] = \nabla_n^m [K(u, v)] = \frac{\partial^{n+m} K(u, v)}{\partial u^n \partial v^m}. \quad (۵.۲)$$

با استفاده از (۵.۲)، می توان تمام مشتقات نگاشت طبیعی را از لحاظ عملکرد تابع هسته (با توجه به این که تابع هسته به اندازه کافی مشتق پذیر است) بیان کرد. به عنوان مثال، روابط زیر برقرارند:

$$\begin{aligned} \nabla_1^\circ [k(u, v)] &= \frac{\partial(\varphi(u)^T \varphi(v))}{\partial u} = \varphi^{(1)}(u)^T \varphi(v), \\ \nabla_0^1 [k(u, v)] &= \frac{\partial(\varphi(u)^T \varphi(v))}{\partial v} = \varphi(u)^T \varphi^{(1)}(v), \\ \nabla_1^1 [k(u, v)] &= \frac{\partial^2(\varphi(u)^T \varphi(v))}{\partial u \partial v} = \varphi^{(1)}(u)^T \varphi^{(1)}(v). \end{aligned}$$

۳.۲ ساخت روش برای ODE خطی

فرض می کنیم که یک جواب تقریبی کلی برای (۱.۲) به صورت $\hat{y}(t) = w^T \varphi(t) + b$ است. که در آن w و b نامعین هستند که باید مشخص شوند. برای به دست آوردن مقدار مطلوب این پارامترها، روش های هم محلی را می توان مورد استفاده قرار داد [۱۶]. فرض کنید نقاط هم محلی از بازه $[a, c]$ به صورت $\gamma = \{a = t_1 < t_2 < \dots < t_N = c\}$ انتخاب شوند. بنابراین، w و b باید با حل مسأله بهینه سازی زیر پیدا شوند:

برای مسأله مقدار اولیه

$$\underset{\hat{y}}{\text{minimize}} \quad \frac{1}{N} \sum_{i=1}^N [(L[\hat{y}] - r)(t_i)]^2 \quad (۶.۲)$$

$$\text{subject to} \quad Ie_\mu[\hat{y}(t)] = p_\mu, \quad \mu = 0, \dots, m-1,$$

برای مسأله مقدار نهایی

$$\underset{\hat{y}}{\text{minimize}} \quad \frac{1}{N} \sum_{i=1}^N [(L[\hat{y}] - r)(t_i)]^2 \quad (۷.۲)$$

$$\text{subject to} \quad Be_\mu[\hat{y}(t)] = q_\mu, \quad \mu = 0, \dots, m-1,$$

۲۴ حل تقریبی معادلات دیفرانسیل معمولی با ماشین بردار پشتیبان کمترین مربعات

که در آن N تعداد نقاط هم محل است که به طور کلی برای فرآیند یادگیری استفاده می شود. در ادامه، مسأله بهینه سازی در چارچوب LS-SVM را برای حل ODE های خطی تشکیل می دهیم. برای سادگی نمادگذاری، علامت های زیر را که در بخش های بعدی استفاده می شوند، در نظر می گیریم:

$$[\nabla_n^m K](t, s) = [\nabla_n^m K(u, v)] \Big|_{u=t, v=s},$$

$$[\Omega_n^m]_{i,j} = \nabla_n^m [K(u, v)] \Big|_{u=t_i, v=t_j} = \frac{\partial^{n+m} K(u, v)}{\partial u^n \partial v^m} \Big|_{u=t_i, v=t_j},$$

$$[\Omega^\circ]_{i,j} = \nabla^\circ [K(u, v)] \Big|_{u=t_i, v=t_j} = K(t_i, t_j),$$

که در آن $[\Omega_n^m]_{i,j}$ ، (i,j) -امین عضو ماتریس Ω_n^m است. نشانه $M_{k:l,m:n}$ برای انتخاب یک زیر ماتریس از ماتریس M ، k تا L سطر و m تا n ستون است. $M_{i,:}$ ، i -امین سطر M و $M_{:,j}$ ، j -امین ستون M است.

آ: مسأله مقدار اولیه مرتبه اول

به عنوان اولین مثال، IVP زیر را در نظر بگیرید:

$$y'(t) - f_1(t)y(t) = r(t), \quad y(a) = p_1, \quad a \leq t \leq c. \quad (۸.۲)$$

در چارچوب LS-SVM، جواب تقریبی مسأله را می توان با جواب مسأله بهینه سازی زیر به دست آورد:

$$\underset{w,b,e}{\text{minimize}} \quad \frac{1}{2} w^T w + \frac{\gamma}{2} e^T e$$

$$\text{subject to} \quad w^T \varphi'(t_i) = f_1(t_i)[w^T \varphi(t_i) + b] + \quad (۹.۲)$$

$$r(t_i) + e_i, \quad i = 2, \dots, N,$$

$$w^T \varphi(t_1) + b = p_1.$$

این مسأله به وسیله ترکیب تابع هزینه LS-SVM با محدودیت هایی با در نظر گرفتن جواب $\hat{y}(t) = w^T \varphi(t) + b$ ، داده شده توسط مدل LS-SVM به دست می آید مسأله (۹.۲) برای برآورده کردن معادلات دیفرانسیل داده شده با مقادیر اولیه در نقاط هم محل $\{t_i\}_{i=1}^N$ به کار می رود. مسأله (۹.۲) یک مسأله برنامه ریزی درجه دوم تحت محدودیت های خطی است که پیدا کردن جواب کارا را امکان پذیر می کند.

لم ۱.۳.۲. یک تابع هسته‌ای معین مثبت $K : \mathbb{R} \times \mathbb{R} \rightarrow \mathbb{R}$ با $K(t, s) = \varphi(t)^T \varphi(s)$ و $\gamma \in \mathbb{R}^+$ داده شده است.

جواب مسأله (۹.۲) با حل مسأله دوگان زیر به دست می‌آید:

$$\begin{bmatrix} \mathcal{K} + I_{N-1/\gamma} & h_{p1} & -f_1 \\ h_{p1}^T & 1 & 1 \\ -f_1^T & 1 & 0 \end{bmatrix} \cdot \begin{bmatrix} \alpha \\ \beta \\ b \end{bmatrix} = \begin{bmatrix} r \\ p_1 \\ 0 \end{bmatrix} \quad (10.2)$$

$$\begin{aligned} \alpha &= [\alpha_2, \dots, \alpha_N]^T, f_1 = [f_1(t_2), \dots, f_1(t_N)]^T \in \mathbb{R}^{N-1}, \\ r &= [r(t_2), \dots, r(t_N)]^T \in \mathbb{R}^{N-1}, \\ \mathcal{K} &= \tilde{\Omega}_1^1 - D_1 \tilde{\Omega}_1^0 - \tilde{\Omega}_0^1 D_1 + D_1 \tilde{\Omega}_0^0 D_1, \\ hp_1 &= [\Omega_0^1]_{1,2:N}^T - D_1 [\Omega_0^0]_{1,2:N}^T. \end{aligned} \quad (11.2)$$

D_1 یک ماتریس قطری با عناصر f_1 روی قطر اصلی است.

$$\begin{aligned} [\Omega_n^m]_{1,2:N} &= [[\Omega_n^m]_{1,2}, \dots, [\Omega_n^m]_{1,N}], \\ \tilde{\Omega}_n^m &= [\Omega_n^m]_{2:N,2:N}, \end{aligned}$$

برای $n, m = 0, 1$,

$$\begin{aligned} \mathcal{K} &\in R^{(N-1) \times (N-1)}, \\ hp_1 &\in \mathbb{R}^{N-1}. \end{aligned}$$

اثبات. تابع لاگرانژ از مسأله بهینه‌سازی مقید (۹.۲) به صورت زیر است:

$$\begin{aligned} L(w, b, e_i, \alpha_i, \beta) &= \frac{1}{\gamma} w^T w + \frac{\gamma}{\gamma} e^T e - \sum_{i=2}^N \alpha_i [w^T (\varphi'(t_i) - f_1(t_i) \varphi(t_i)) - f_1(t_i) b - r_i - e_i] \\ &\quad - \beta (w^T \varphi(t_1) + b - p_1) \end{aligned}$$

که در آن $\{\alpha_i\}_{i=2}^N$ و β ضرایب لاگرانژ هستند و برای $i = 2, \dots, N$ داریم $r_i = r(t_i)$ سپس شرایط بهینه (KKT) به شرح زیر است:

$$\left\{ \begin{aligned} \frac{\partial L}{\partial w} = 0 &\rightarrow w = \sum_{i=2}^N \alpha_i [\varphi'(t_i) - f_1(t_i) \varphi(t_i)] + \beta \varphi(t_1), \\ \frac{\partial L}{\partial b} = 0 &\rightarrow \sum_{i=2}^N \alpha_i f_1(t_i) - \beta = 0, \\ \frac{\partial L}{\partial e_i} = 0 &\rightarrow e_i = -\frac{\alpha_i}{\gamma} \quad i = 2, \dots, N, \\ \frac{\partial L}{\partial \alpha_i} = 0 &\rightarrow r_i = w^T (\varphi'(t_i) - f_1(t_i) \varphi(t_i)) - f_1(t_i) b - e_i, \quad i = 2, \dots, N, \\ \frac{\partial L}{\partial \beta} = 0 &\rightarrow p_1 = w^T \varphi(t_1) + b. \end{aligned} \right.$$

۲۶ حل تقریبی معادلات دیفرانسیل معمولی با ماشین بردار پشتیبان کمترین مربعات

سپس از حذف متغیر اولیه w و $\{e_i\}_{i=2}^N$ استفاده از قضیه مرسر، جواب در فرم دوگان به صورت زیر داده می شود:

$$\begin{cases} [f_1(t_i)[w^T \varphi(t) + b] + r(t_i)] = w^T \varphi'(t_i) + e_i \\ r_i = \sum_{j=2}^N \alpha_j [\varphi'(t_j) - f_1(t_j)\varphi(t_j) + \beta\varphi(t_1)] [\varphi'(t_i) - f_1(t_i)\varphi(t_i)] - f_1(t_i)b - e_i \\ r_i = \sum_{j=2}^N \alpha_j [\varphi'(t_j)\varphi'(t_i) - f_1(t_i)\varphi'(t_j)\varphi(t_i) - f_1(t_j)\varphi(t_j)\varphi'(t_i) + f_1(t_j)f_1(t_i)\varphi(t_j)\varphi(t_i) \\ + \beta\varphi(t_1)\varphi'(t_i) - \beta f_1(t_i)\varphi(t_1)\varphi(t_i)] - f_1(t_i)b - e_i \\ r_i = \sum_{j=2}^N \alpha_j [\Omega_1^1(j, i) - f_1(t_i)[\Omega_1^0(j, i) - f_1(t_j)\Omega_0^0(j, i)] - f_1(t_j)\Omega_0^1(j, i)] \\ + \beta[\Omega_0^1(1, i) - f_1(t_i)\Omega_0^0(1, i)] + \frac{\alpha_i}{\gamma} - f_1(t_i)b \\ p_1 = \sum_{j=2}^N \alpha_j [\Omega_1^0(j, 1) - f_1(t_j)\Omega_0^0(j, 1)] + \beta\Omega_0^0(1, 1) + b \\ o = \sum_{j=2}^N \alpha_j f_1(t_j) - \beta \end{cases}$$

وقتی $K(x, y) = \varphi(x) \cdot \varphi(y)$ یک تابع معین مثبت است، از عملگر دیفرانسیل به کار گرفته شده در زیر استفاده می کنیم:

$$\nabla_n^m = \frac{\partial^{n+m}}{\partial^n \partial^m},$$

$$[\varphi^{(n)}(x)] \cdot \varphi^{(m)}(y) = \nabla_n^m [\varphi(x)\varphi(y)] = \nabla_n^m [K(x, y)],$$

$$\nabla_1^0 [K(x, y)] = \varphi^{(1)}(x) \cdot \varphi(y), \quad \nabla^0 [K(x, y)] = \varphi(x) \cdot \varphi(y) = K(x, y),$$

$$\nabla_1^1 [K(x, y)] = \varphi(x) \cdot \varphi^{(1)}(y), \quad \nabla_1^1 [K(x, y)] = \varphi^{(1)}(x) \cdot \varphi^{(1)}(y).$$

بنابراین داریم:

$$\hat{y}(t) = \sum_{i=2}^N \alpha_i [\nabla_1^0 K(t_i, t) - f_1(t_i) \nabla^0 K(t_i, t)] + \beta \nabla^0 K(t_1, t) + b.$$

□

مدل در فرم دوگان به صورت زیر می شود:

$$\hat{y} = \sum_{i=2}^N \alpha_i ([\nabla_1^0 K](t_i, t) - f_1(t_i) [\nabla^0 K](t_i, t)) + \beta [\nabla^0 K](t_1, t) + b.$$

جایی که K تابع هسته است.

ب: مسائل مقدار اولیه و مقدار مرزی مرتبه دوم

IVP مرتبه دوم به صورت زیر در نظر می گیریم:

$$y'' = f_1(t)y'(t) + f_2(t)y(t) + r(t), \quad t \in [a, c]$$

$$y(a) = p_1, \quad y'(a) = p_2.$$

جواب تقریبی، $\hat{y}(t) = w^T \varphi(t) + b$ با حل مسأله بهینه‌سازی زیر حاصل می‌شود:

$$\underset{w, b, e}{\text{minimize}} \quad \frac{1}{2} w^T w + \frac{\gamma}{2} e^T e$$

$$\text{subject to } w^T \varphi''(t_i) = f_1(t_i) w^T \varphi'(t_i) \quad (12.2)$$

$$+ f_2(t_i) [w^T \varphi(t_i) + b] + r(t_i) + e_i, \quad i = 2, \dots, N,$$

$$w^T \varphi(t_1) + b = p_1,$$

$$w^T \varphi'(t_1) = p_2.$$

لم ۲.۳.۲. فرض کنید یک تابع هسته‌ای معین مثبت $K : \mathbb{R} \times \mathbb{R} \rightarrow \mathbb{R}$ ،
 $K(t, s) = \varphi(t)^T \varphi(s)$ و $\gamma \in \mathbb{R}^+$ داده شده است. جواب (۱۲.۲) با حل مسأله دوگان زیر به دست می‌آید:

$$\begin{bmatrix} \mathcal{K} + I_{N-1}/\gamma & h_{p1} & h_{p2} & -f_2 \\ h_{p1}^T & 1 & 0 & 1 \\ h_{p2}^T & 0 & [\Omega_1^1]_{1,1} & 0 \\ -f_2^T & 1 & 0 & 0 \end{bmatrix} \cdot \begin{bmatrix} \alpha \\ \beta_1 \\ \beta_2 \\ b \end{bmatrix} = \begin{bmatrix} r \\ p_1 \\ p_2 \\ 0 \end{bmatrix} \quad (13.2)$$

$$\alpha = [\alpha_2, \dots, \alpha_N]^T, f_1 = [f_1(t_2), \dots, f_1(t_N)]^T \in \mathbb{R}^{N-1},$$

$$f_2 = [f_2(t_2), \dots, f_2(t_N)]^T \in \mathbb{R}^{N-1},$$

$$r = [r(t_2), \dots, r(t_N)]^T \in \mathbb{R}^{N-1}, \quad (14.2)$$

$$\mathcal{K} = \tilde{\Omega}_2^2 - D_1 \tilde{\Omega}_2^1 - D_2 \tilde{\Omega}_2^0 - \tilde{\Omega}_1^2 D_1 - \tilde{\Omega}_1^2 D_2 + D_1 \tilde{\Omega}_1^1 D_1 + D_1 \tilde{\Omega}_1^0 D_2 + D_2 \tilde{\Omega}_1^1 D_1 + D_2 \tilde{\Omega}_1^0 D_2,$$

$$h_{p1} = [\Omega_2^0]_{1,2:N}^T - D_1 [\Omega_1^1]_{1,2:N}^T - D_2 [\Omega_1^0]_{1,2:N}^T,$$

$$h_{p2} = [\Omega_2^1]_{1,2:N}^T - D_1 [\Omega_1^1]_{1,2:N}^T - D_2 [\Omega_1^0]_{1,2:N}^T.$$

D_1 و D_2 ماتریس‌های قطری با عناصر f_1, f_2 روی قطری اصلی هستند. و

$$\mathcal{K} \in \mathbb{R}^{(N-1) \times (N-1)},$$

و

$$h_{p1}, h_{p2} \in \mathbb{R}^{N-1}.$$

برای $m = 0, 1$ و $n = 0, 1$ داریم:

$$[\Omega_n^m]_{1,2:N} = [[\Omega_n^m]_{1,2}, \dots, [\Omega_n^m]_{1,N}].$$

و برای $n, m = 0, 1, 2$ داریم:

$$\Omega_n^m = [\Omega_n^m]_{2:N, 2:N}.$$

اثبات. تابع لاگرانژ زیر را در نظر بگیرید:

$$L(w, b, e_i, \alpha_i, \beta_1, \beta_2) = \frac{1}{\gamma} w^T w + \frac{\gamma}{\gamma} e^T e - \sum_{i=2}^N \alpha_i [w^T [\varphi''(t_i) - f_1(t_i) \varphi'(t_i) - f_2(t_i) \varphi(t_i)] - f_2(t_i) b - r_i - e_i] - \beta_1 [w^T \varphi(t_1) + b - p_1] - \beta_2 [w^T \varphi'(t_1) - p_2] \quad (15.2)$$

که در آن $\{\alpha_i\}_{i=2}^N$ و β_1 و β_2 ضرایب لاگرانژ هستند. شرایط بهینگی K.K.T به شرح زیر است:

$$\left\{ \begin{array}{l} \frac{\partial L}{\partial w} = 0 \rightarrow w = \sum_{i=2}^N \alpha_i [\varphi''_i - f_1(t_i) \varphi'_i - f_2(t_i) \varphi_i] + \beta_1 \varphi_1 + \beta_2 \varphi'_1, \\ \frac{\partial L}{\partial b} = 0 \rightarrow \sum_{i=2}^N \alpha_i f_2(t_i) - \beta_1 = 0, \\ \frac{\partial L}{\partial e_i} = 0 \rightarrow e_i = \frac{-\alpha_i}{\gamma}, \quad i = 2, \dots, N, \\ \frac{\partial L}{\partial \alpha_i} = 0 \rightarrow r_i = w^T [\varphi''_i - f_1(t_i) \varphi'_i - f_2(t_i) \varphi_i] - f_2(t_i) b - e_i, \\ \frac{\partial L}{\partial \beta_1} = 0 \rightarrow w^T \varphi_1 + b = p_1, \\ \frac{\partial L}{\partial \beta_2} = 0 \rightarrow w^T \varphi'_1 = p_2, \\ \varphi_i = \varphi(t_i), \quad \varphi'_i = \varphi'(t_i), \quad \varphi''_i = \varphi''(t_i), \quad i = 2, \dots, N. \end{array} \right.$$

$$\left\{ \begin{array}{l} r_i = \left[\sum_{j=2}^N \alpha_j [\varphi''_j - f_1(t_j) \varphi'_j - f_2(t_j) \varphi_j] + \beta_1 \varphi_1 + \beta_2 \varphi'_1 \right] [\varphi''_i - f_1(t_i) \varphi'_i - f_2(t_i) \varphi_i] \\ - f_2(t_i) b - e_i \\ r_i = \sum_{j=2}^N \alpha_j \left[\varphi''_j \varphi''_i - f_1(t_i) \varphi''_j \varphi'_i - f_2(t_i) \varphi''_j \varphi_i - f_1(t_j) \varphi'_j \varphi''_i + f_1(t_j) f_1(t_i) \varphi'_j \varphi'_i + \right. \\ \left. f_1(t_j) f_2(t_i) \varphi'_j \varphi_i - f_2(t_j) \varphi_j \varphi''_i \right. \\ \left. + f_2(t_j) f_1(t_i) \varphi_j \varphi'_i + f_2(t_j) f_2(t_i) \varphi_j \varphi_i \right] + \beta_1 \varphi_1 \varphi''_i \\ - \beta_1 \varphi_1 f_1(t_i) \varphi'_i - \beta_1 \varphi_1 f_2(t_i) \varphi_i + \beta_2 \varphi'_1 \varphi''_i \\ - \beta_2 \varphi'_1 f_1(t_i) \varphi'_i - \beta_2 \varphi'_1 f_2(t_i) \varphi_i - f_2(t_i) b - e_i \\ r_i = \sum_{j=2}^N \alpha_j [\Omega^\nabla(j, i) - \Omega^\nabla(j, i) f_1(t_i) - \Omega^\nabla(j, i) f_2(t_i) - \Omega^\nabla(j, i) f_1(t_j) + \Omega^\nabla(j, i) f_1(t_j) f_1(t_i) \\ + \Omega^\nabla(j, i) f_1(t_j) f_2(t_i) - \Omega^\nabla(j, i) f_2(t_j) + \Omega^\nabla(j, i) f_2(t_j) f_1(t_i) + \Omega^\nabla(j, i) f_2(t_j) f_2(t_i)] \\ + \beta_1 \Omega^\nabla(1, i) - \beta_1 f_1(t_i) \Omega^\nabla(1, i) - \beta_1 f_2(t_i) \Omega^\nabla(1, i) + \beta_2 \Omega^\nabla(1, i) - \beta_2 f_1(t_i) \Omega^\nabla(1, i) \\ - \beta_2 f_2(t_i) \Omega^\nabla(1, i) - f_2(t_i) b - e_i \\ r_i = \sum_{j=2}^N \alpha_j [\Omega^\nabla(j, i) - f_1(t_i) [\Omega^\nabla(j, i) - f_1(t_j) \Omega^\nabla(j, i) - f_2(t_j) \Omega^\nabla(j, i)] \\ - f_2(t_i) [\Omega^\nabla(j, i) - f_1(t_j) \Omega^\nabla(j, i) - f_2(t_j) \Omega^\nabla(j, i)] - f_1(t_j) \Omega^\nabla(j, i) \\ - f_2(t_j) \Omega^\nabla(j, i)] + \beta_1 [\Omega^\nabla(1, i) - f_1(t_i) \Omega^\nabla(1, i) - f_2(t_i) \Omega^\nabla(1, i)] \\ + \beta_2 [\Omega^\nabla(1, i) - f_1(t_i) \Omega^\nabla(1, i) - f_2(t_i) \Omega^\nabla(1, i)] + \frac{\alpha_i}{\gamma} - f_2(t_i) b \\ p_1 = w^T \varphi(t_1) + b \quad p_2 = w^T \varphi'(t_1) \end{array} \right.$$

$$\left\{ \begin{array}{l} p_1 = \sum_{j=2}^N \alpha_j [\Omega_2^\circ(j, 1) - f_1(t_j) \Omega_1^\circ(j, 1) - f_2(t_j) \Omega_0^\circ(j, 1)] + \beta_1 \Omega_0^\circ(1, 1) + \beta_2 \Omega_1^\circ(1, 1) + b, \\ p_2 = \sum_{j=2}^N \alpha_j [\Omega_2^1(j, 1) - f_1(t_j) \Omega_1^1(j, 1) - f_2(t_j) \Omega_0^1(j, 1)] + \beta_1 \Omega_0^1(1, 1) + \beta_2 \Omega_1^1(1, 1), \\ \sum_{j=2}^N \alpha_j f_2(t_j) - \beta_1 = 0. \\ \rightarrow y(t) = \sum_{i=2}^N \alpha_i [\nabla_2^\circ k(t_i, t) - f_1(t_i) \nabla_1^\circ k(t_i, t) - f_2(t_i) \nabla_0^\circ k(t_i, t)] + \beta_1 \nabla_0^\circ k(t_1, t) \\ + \beta_2 \nabla_1^\circ k(t_1, t) + b, \\ \rightarrow y'(t) = \sum_{i=2}^N \alpha_i [\nabla_2^1 k(t_i, t) - f_1(t_i) \nabla_1^1 k(t_i, t) - f_2(t_i) \nabla_0^1 k(t_i, t)] + \beta_1 \nabla_0^1 k(t_1, t) \\ + \beta_2 \nabla_1^1 k(t_1, t) \end{array} \right.$$

□

مدل LS-SVM برای جواب و مشتق آن در فرم دوگان به صورت زیر بیان می شوند:

$$\begin{aligned} \hat{y}(t) &= \sum_{i=2}^N \alpha_i \left([\nabla_2^\circ K](t_i, t) - f_1(t_i) [\nabla_1^\circ K](t_i, t) \right. \\ &\quad \left. - f_2(t_i) [\nabla_0^\circ K](t_i, t) \right) + \beta_1 [\nabla_0^\circ k](t_1, t) + \beta_2 [\nabla_1^\circ K](t_1, t) + b, \\ \frac{d\hat{y}(t)}{dt} &= \sum_{i=2}^N \alpha_i \left([\nabla_2^1 K](t_i, t) - f_1(t_i) [\nabla_1^1 K](t_i, t) \right. \\ &\quad \left. - f_2(t_i) [\nabla_0^1 K](t_i, t) \right) + \beta_1 [\nabla_0^1 K](t_1, t) \\ &\quad + \beta_2 [\nabla_1^1 K](t_1, t). \end{aligned}$$

۴.۲ حالت BVP

یک BVP مرتبه دوم از ODE را به صورت زیر در نظر بگیرید:

$$y''(t) = f_1(t)y'(t) + f_2(t)y(t) + r(t), \quad t \in [a, c].$$

$$y(a) = p_1, \quad y(c) = q_1.$$

سپس پارامترهای فرم بسته جواب را می توان با حل مسأله بهینه سازی زیر به دست آورد:

$$\underset{w, b, e}{\text{minimize}} \quad \frac{1}{2} w^T w + \frac{\gamma}{2} e^T e$$

۳۰ حل تقریبی معادلات دیفرانسیل معمولی با ماشین بردار پشتیبان کمترین مربعات

$$\begin{aligned}
 \text{subject to } w^T \varphi''(t_i) &= f_1(t_i) w^T \varphi'(t_i) \\
 &+ f_2(t_i) [w^T \varphi(t_i) + b] + r(t_i) + e_i, \\
 i &= 2, \dots, N-1, \\
 w^T \varphi(t_1) + b &= p_1, \\
 w^T \varphi(t_N) + b &= q_1.
 \end{aligned} \tag{۱۶.۲}$$

بر مبنای روش K.K.T همان روش را می‌توان برای استخراج لاگرانژ و جواب مسأله (۱۶.۲) با حل دستگاه خطی زیر به‌دست آورد:

$$\begin{bmatrix} \mathcal{K} + I_{N-2/\gamma} & h_{p1} & h_{q1} & -f_2 \\ h_{p1}^T & 1 & [\Omega^\circ]_{N,1} & 1 \\ h_{q1}^T & [\Omega^\circ]_{N,1} & 1 & 1 \\ -f_2^T & 1 & 1 & 0 \end{bmatrix} \cdot \begin{bmatrix} \alpha \\ \beta_1 \\ \beta_2 \\ b \end{bmatrix} = \begin{bmatrix} r \\ p_1 \\ q_1 \\ 0 \end{bmatrix} \tag{۱۷.۲}$$

$$\begin{aligned}
 \alpha &= [\alpha_2, \dots, \alpha_{N-1}]^T, f_1 = [f_1(t_2), \dots, f_1(t_{N-1})]^T \in \mathbb{R}^{N-2}, \\
 f_2 &= [f_2(t_2), \dots, f_2(t_{N-1})]^T \in \mathbb{R}^{N-2}, \\
 r &= [r(t_2), \dots, r(t_{N-1})]^T \in \mathbb{R}^{N-2}, \\
 \mathcal{K} &= \tilde{\Omega}_2^\top - D_1 \tilde{\Omega}_2^1 - D_2 \tilde{\Omega}_2^\circ - \tilde{\Omega}_2^\top D_1 - \tilde{\Omega}_2^\top D_2 + D_1 \tilde{\Omega}_2^1 D_1 + D_1 \tilde{\Omega}_2^\circ D_2 + D_2 \tilde{\Omega}_2^1 D_1 + D_2 \tilde{\Omega}_2^\circ D_2, \\
 hp_1 &= [\Omega_2^\top]_{1,2:N-1}^T - D_1 [\Omega_2^\circ]_{1,2:N-1}^T - D_2 [\Omega_2^\circ]_{1,2:N-1}^T, \\
 hq_1 &= [\Omega_2^\top]_{N,2:N-1}^T - D_1 [\Omega_2^1]_{N,2:N-1}^T - D_2 [\Omega_2^\circ]_{N,2:N-1}^T.
 \end{aligned}$$

D_1 و D_2 ماتریس‌های قطری با عناصر f_1, f_2 روی قطری اصلی هستند.

$$\mathcal{K} \in \mathbb{R}^{(N-2) \times (N-2)},$$

9

$$hp_1, hq_1 \in \mathbb{R}^{N-2}.$$

$$[\Omega_n^m]_{1,2:N-1} = [[\Omega_n^m]_{1,2}, \dots, [\Omega_n^m]_{1,N-1}].$$

برای $n = 0, 1$ و $m = 0, 1$ داریم:

$$[\Omega_n^m]_{N,2:N-1} = [[\Omega_n^m]_{N,2}, \dots, [\Omega_n^m]_{N,N-1}].$$

و برای $n, m = 0, 1, 2$ داریم:

$$\Omega_n^m = [\Omega_n^m]_{2:N-1,2:N-1}.$$

اثبات. تابع لاگرانژ از مسأله بهینه سازی (۱۶.۲) را در نظر بگیرید:

$$L(w, b, e_i, \alpha_i, \beta_1, \beta_2) = \frac{1}{\gamma} w^T w + \frac{\gamma}{\gamma} e^T e - \sum_{i=2}^{N-1} \alpha_i [w^T [\varphi''(t_i) - f_1(t_i) \varphi'(t_i)] - f_2(t_i) \varphi(t_i)] \\ - f_2(t_i) - r_i - e_i] - \beta_1 [w^T \varphi(t_1) + b - p_1] - \beta_2 [w^T \varphi(t_N) + b - q_1]$$

که در آن $\{\alpha_i\}_{i=1}^{N-1}$ و β_1 و β_2 ضرایب لاگرانژ هستند.

شرایط بهینگی K.K.T به شرح زیر است:

$$\frac{\partial L}{\partial w} = 0 \rightarrow w = \sum_{i=2}^{N-1} \alpha_i [\varphi''(t_i) - f_1(t_i) \varphi'(t_i) - f_2(t_i) \varphi(t_i)] + \beta_1 \varphi(t_1) + \beta_2 \varphi(t_N),$$

$$\frac{\partial L}{\partial b} = 0 \rightarrow \sum_{i=2}^{N-1} \alpha_i f_2(t_i) - \beta = 0,$$

$$\frac{\partial L}{\partial e_i} = 0 \rightarrow e_i = \frac{-\alpha_i}{\gamma}, \quad i = 2, \dots, N-1,$$

$$\frac{\partial L}{\partial \alpha_i} = 0 \rightarrow r_i = w^T [\varphi''(t_i) - f_1(t_i) \varphi'(t_i) - f_2(t_i) \varphi(t_i)] - f_2(t_i) b - e_i$$

$$\frac{\partial L}{\partial \beta_1} = 0 \rightarrow w^T \varphi(t_1) + b = p_1$$

$$\frac{\partial L}{\partial \beta_2} = 0 \rightarrow w^T \varphi(t_N) + b = q_1$$

$$r_i = \sum_{j=2}^{N-1} \alpha_j [\varphi_j'' - f_1(t_j) \varphi_j' - f_2(t_j) \varphi_j + \beta_1 \varphi_1 + \beta_2 \varphi_N] [\varphi_i'' - f_1(t_i) \varphi_i' - f_2(t_i) \varphi_i] - f_2(t_i) b - e_i$$

$$r_i = \sum_{j=2}^{N-1} \alpha_j \left[\varphi_j'' \varphi_i'' - f_1(t_i) \varphi_j'' \varphi_i' - f_2(t_i) \varphi_j'' \varphi_i - f_1(t_j) \varphi_j' \varphi_i'' + f_1(t_j) f_1(t_i) \varphi_j' \varphi_i' + f_1(t_j) f_2(t_i) \varphi_j' \varphi_i \right. \\ \left. - f_2(t_j) \varphi_j \varphi_i'' + f_2(t_j) f_1(t_i) \varphi_j \varphi_i' + f_2(t_j) f_2(t_i) \varphi_j \varphi_i \right] + \beta_1 \varphi_1 \varphi_i'' - f_1(t_i) \beta_1 \varphi_1 \varphi_i'$$

$$- f_2(t_i) \beta_1 \varphi_1 \varphi_i + \beta_2 \varphi_N \varphi_i'' - f_1(t_i) \beta_2 \varphi_N \varphi_i' - f_2(t_i) \beta_2 \varphi_N \varphi_i - f_2(t_i) b - e_i$$

$$r_i = \sum_{j=2}^{N-1} \alpha_j [\Omega_2^{\vee}(j, i) - f_1(t_i) [\Omega_2^{\vee}(j, i) - f_1(t_j) \Omega_1^{\vee}(j, i) - f_2(t_j) \Omega_0^{\vee}(j, i)]]$$

$$- f_2(t_i) [\Omega_2^{\circ}(j, i) - f_1(t_j) \Omega_1^{\circ}(j, i) - f_2(t_j) \Omega_0^{\circ}(j, i)] - f_1(t_j) \Omega_1^{\vee}(j, i)$$

$$- f_2(t_j) \Omega_0^{\vee}(j, i) + \beta_1 [\Omega_0^{\vee}(1, i) - f_1(t_i) \Omega_0^{\vee}(1, i) - f_2(t_i) \Omega_0^{\circ}(1, i)]$$

$$+ \beta_2 [\Omega_0^{\vee}(1, i) - f_1(t_i) \Omega_0^{\vee}(1, i) - f_2(t_i) \Omega_0^{\circ}(1, i)] - f_2(t_i) b + \frac{\alpha_i}{\gamma}$$

$$p_1 = w^T \varphi(t_1) + b, \quad q_1 = w^T \varphi(t_N) + b,$$

$$\begin{cases} p_1 = \sum_{j=2}^{N-1} \alpha_j [\Omega_2^{\circ}(j, 1) - f_1(t_j) \Omega_1^{\circ}(j, 1) - f_2(t_j) \Omega_0^{\circ}(j, 1)] + \beta_1 \Omega_0^{\circ}(1, 1) + \beta_2 \Omega_0^{\circ}(1, 1) + b, \\ q_1 = \sum_{j=2}^{N-1} \alpha_j [\Omega_2^{\vee}(j, i) - f_1(t_j) \Omega_1^{\vee}(j, i) - f_2(t_j) \Omega_0^{\circ}(j, 1)] + \beta_1 \Omega_0^{\circ}(1, 1) + \beta_2 \Omega_0^{\vee}(1, 1) + b, \\ \circ = \sum_{j=2}^{N-1} \alpha_j f_2(t_j) - \beta_1. \end{cases}$$

$$\begin{aligned}
 y(t) &= \sum_{i=2}^{N-1} \alpha_i [\nabla_2^\circ k(t_i, t) - f_1(t_i) \nabla_1^\circ k(t_i, t) - f_2(t_i) \nabla^\circ k(t_i, t)] \\
 &\quad + \beta_1 \nabla^\circ k(t_1, t) + \beta_2 \nabla^\circ k(t_N, t) + b, \\
 y'(t) &= \sum_{i=2}^{N-1} \alpha_i [\nabla_2^1 k(t_i, t) - f_1(t_i) \nabla_1^1 k(t_i, t) - f_2(t_i) \nabla^1 k(t_i, t)] \\
 &\quad + \beta_1 \nabla^1 k(t_1, t) + \beta_2 \nabla^1 k(t_N, t).
 \end{aligned}$$

مدل LS-SVM برای جواب و مشتق آن به فرم دوگان به صورت زیر بیان می شود:

$$\begin{aligned}
 \hat{y}(t) &= \sum_{i=2}^{N-1} \alpha_i \left([\nabla_2^\circ K](t_i, t) - f_1(t_i) [\nabla_1^\circ K](t_i, t) \right. \\
 &\quad \left. - f_2(t_i) [\nabla^\circ K](t_i, t) \right) + \beta_1 [\nabla^\circ K](t_1, t) \\
 &\quad + \beta_2 [\nabla^\circ K](t_N, t) + b, \\
 \frac{d\hat{y}(t)}{dt} &= \sum_{i=2}^{N-1} \alpha_i \left([\nabla_2^1 K](t_i, t) - f_1(t_i) [\nabla_1^1 K](t_i, t) \right. \\
 &\quad \left. - f_2(t_i) [\nabla^1 K](t_i, t) \right) + \beta_1 [\nabla^1 K](t_1, t) \\
 &\quad + \beta_2 [\nabla^1 K](t_N, t).
 \end{aligned}$$

□

پ : ODE خطی مرتبه m -ام

اکنون IVP های معمولی m -ام را به فرم زیر در نظر بگیرید:

$$\begin{aligned}
 y^{(m)}(t) - \sum_{i=1}^m f_i(t) y^{(m-i)}(t) &= r(t), \quad t \in [a, c]. \\
 \begin{cases} y(a) = p_1, \\ y^{(i-1)}(a) = p_i, \quad i = 2, \dots, m. \end{cases} & \quad (18.2)
 \end{aligned}$$

جواب تقریبی می تواند با حل مسأله بهینه سازی زیر به دست آید:

$$\begin{aligned}
 &\underset{w, b, e}{\text{minimize}} \quad \frac{1}{2} w^T w + \frac{\gamma}{2} e^T e \\
 &\text{subject to} \quad w^T \varphi^{(m)}(t_i) = w^T \left[\sum_{k=1}^m f_k(t_i) \varphi_i^{(m-k)} \right] + f_m(t_i) b + r(t_i) + e_{i,i} = 2, \dots, m, \\
 &\quad w^T \varphi(t_1) + b = p_1, \\
 &\quad w^T \varphi^{(i-1)}(t_1) = p_i, \quad i = 2, \dots, m.
 \end{aligned} \quad (19.2)$$

لم ۱.۴.۲. فرض کنید یک تابع هسته‌ای معین مثبت $K : \mathbb{R} \times \mathbb{R} \rightarrow \mathbb{R}$ با $K(t, s) = \varphi(t)^T \varphi(s)$ با پارامترهای منظم‌سازی ثابت و $\gamma \in \mathbb{R}^+$ داده شده است. جواب برای (۱۹.۲) با حل مسأله دوگان زیر به دست می‌آید:

$$\begin{bmatrix} \mathcal{K} + I_{N-1}/\gamma & \mathcal{K}_{in} & -f_m \\ \mathcal{K}_{in}^T & \Delta & C \\ -f_m^T & C^T & \circ \end{bmatrix} \cdot \begin{bmatrix} \alpha \\ \beta \\ b \end{bmatrix} = \begin{bmatrix} r \\ p \\ \circ \end{bmatrix} \quad (۲۰.۲)$$

$$f_k = [f_k(t_1), \dots, f_k(t_N)]^T \in \mathbb{R}^{N-1}, \quad k = 1, \dots, m,$$

$$h_{pl}[\Omega_{\ell-1}^m]^T_{1:N} - \sum_{k=1}^m D_k[\Omega_{\ell-1}^{m-k}]^T_{1:N}, \ell = 1, \dots, m,$$

$$\mathcal{K}_{in} = [h_{p1}, \dots, h_{pm}] \in \mathbb{R}^{(N-1) \times m}, \quad h_{pl} \in \mathbb{R}^{N-1},$$

$$p = [p_1, \dots, p_m]^T \in \mathbb{R}^m, \quad C = [1, \circ, \dots, \circ]^T \in \mathbb{R}^m,$$

$$r = [r(t_1), \dots, r(t_N)]^T \in \mathbb{R}^{N-1}, \quad \alpha = [\alpha_1, \dots, \alpha_N]^T,$$

$$\beta = [\beta_1, \dots, \beta_m]^T, \quad \tilde{\Omega}_1 = [\tilde{\Omega}_m^\circ, \dots, \tilde{\Omega}_m^{m-1}]^T,$$

$$\bar{D} = [D_m, \dots, D_1], \quad \bar{\Omega} = \begin{bmatrix} \tilde{\Omega}_m^\circ & \dots & \tilde{\Omega}_{m-1}^\circ \\ \vdots & \ddots & \vdots \\ \tilde{\Omega}_m^{m-1} & \dots & \tilde{\Omega}_{m-1}^{m-1} \end{bmatrix}$$

$$\mathcal{K} = \tilde{\Omega}_m^m - \bar{D}\bar{\Omega}_1 - \bar{\Omega}_1^T \bar{D}^T + \bar{D}\bar{\Omega}\bar{D}^T,$$

$$\Delta = \begin{bmatrix} [\Omega_m^\circ]_{1,1} & \dots & [\Omega_{m-1}^\circ]_{1,1} \\ \vdots & \ddots & \vdots \\ [\Omega_m^{m-1}]_{1,1} & \dots & [\Omega_{m-1}^{m-1}]_{1,1} \end{bmatrix}_{m \times m}$$

$\{D_i\}_{i=1}^m$ یک ماتریس قطری با عناصر $\{f_i\}_{i=1}^m$ روی قطر اصلی است. همچنین $\bar{\Omega}_1$ و $\bar{D}$ ماتریس مسدود کننده هستند و $\tilde{\Omega}_n^m = [\Omega_n^m]_{1:N, 1:N}$ و همچنین داریم:

$$\mathcal{K} \in \mathbb{R}^{(N-1) \times (N-1)}.$$

اثبات. تابع لاگرانژ زیر را در نظر بگیرید:

$$L(w, b, e_i, \alpha_i, \beta_i) = \frac{1}{\gamma} w^T w + \frac{\gamma}{\gamma} e^T e - \sum_{i=1}^N \alpha_i [w^T (\varphi^m(t_i) - \sum_{k=1}^N f_k(t_i)$$

$$\varphi^{(m-k)}(t_i)) - f_m(t_i)b - r_i - e_i] - \beta_1 (w^T \varphi(t_1) + b - p_1)$$

$$- \beta_2 (w^T \dot{\varphi}(t_1) - p_2) - \dots - \beta_m (w^T \varphi^{(m-1)}(t_1) - p_m).$$

که در آن $\{\alpha_i\}_{i=1}^{N-1}$ و β_i ضرایب لاگرانژ هستند.
شرایط بهینگی K.K.T به صورت زیر است:

$$\frac{\partial L}{\partial w} = 0 \rightarrow w = \sum_{i=1}^N \alpha_i [\varphi^{(m)}(t_i) - \sum_{k=1}^N f_k(t_i) \varphi^{(m-k)}(t_i)] + \beta_1 \varphi(t_1) + \beta_2 \varphi'(t_1) + \dots + \beta_m \varphi^{(m-1)}(t_1)$$

$$\frac{\partial L}{\partial b} = 0 \rightarrow \sum_{i=1}^N \alpha_i f_m(t_i) - \beta_1 = 0$$

$$\frac{\partial L}{\partial \alpha_i} = 0 \rightarrow r_i = w^T [\varphi^{(m)}(t_i) - \sum_{k=1}^N f_k(t_i) \varphi^{(m-k)}(t_i)] - f_m(t_i) b - e_i$$

$$\frac{\partial L}{\partial e_i} = 0 \rightarrow \gamma e_i = -\alpha_i, \quad i = 1, \dots, N.$$

$$\frac{\partial L}{\partial \beta_i} = 0 \rightarrow [w^T \varphi(t_1) + b - p_1] - [w^T \varphi'(t_1) - p_2] - \dots - [w^T \varphi^{(m-1)}(t_1) - p_m]$$

$$r_i = \sum_{j=1}^N \alpha_j [\varphi^{(m)}(t_j) - \sum_{k=1}^m f_k(t_j) \varphi^{(m-k)}(t_j)] + [\beta_1 \varphi(t_1) + \beta_2 \varphi'(t_1) + \dots + \beta_m \varphi^{(m-1)}(t_1)] [\varphi^{(m)}(t_i) - \sum_{L=1}^m f_L(t_i) \varphi^{(m-L)}(t_i)] - f_m(t_i) b - e_i$$

$$r_i = \sum_{j=1}^N \alpha_j [\varphi^{(m)}(t_j) \varphi^{(m)}(t_i) - \varphi^{(m)}(t_j) \sum_{L=1}^m f_L(t_i) \varphi^{(m-L)}(t_i) - \sum_{k=1}^m f_k(t_j) \varphi^{(m-k)}(t_j) \varphi^{(m)}(t_i) + \sum_{k=1}^m \sum_{L=1}^m f_k(t_j) \varphi^{(m-k)}(t_j) f_L(t_i) \varphi^{(m-L)}(t_i)] + \beta_1 \varphi(t_1) \varphi^{(m)}(t_i) - \beta_1 \varphi(t_1) \sum_{L=1}^m f_L(t_i) \varphi^{(m-L)}(t_i) + \dots + \beta_m \varphi^{(m-1)}(t_1) \varphi^{(m)}(t_i) - \beta_m \varphi^{(m-1)}(t_i) \sum_{L=1}^m f_L(t_i) \varphi^{(m-1)}(t_i) - f_m(t_i) b - e_i$$

با حذف w و $\{e_i\}_{i=1}^N$ از شرایط مطلوب KKT، مجموعه معادلات زیر را ارائه می‌دهد:

$$\left\{ \begin{array}{l} r_i = \sum_{j=1}^N \alpha_j [[\Omega_m^m]_{j,i} - \sum_{l=1}^m f_l(t_i) ([\Omega_m^{m-l}]_{j,i} - \sum_{k=1}^m f_k(t_j) [\Omega_{m-k}^{m-l}]_{j,i}) - \sum_{k=1}^m f_k(t_j) [\Omega_{m-k}^m]_{j,i} + \sum_{l=1}^m \beta_l ([\Omega_{l-1}^m]_{1,i} - \sum_{k=1}^m f_k(t_i) [\Omega_{l-1}^{m-k}]_{1,i}) + \frac{\alpha_i}{\gamma} - f_m(t_i) b, \quad i = 1, \dots, N. \end{array} \right.$$

$$\begin{cases} p_1 = \sum_{j=2}^N \alpha_j ([\Omega_m^\circ]_{j,1} - \sum_{k=1}^m f_k(t_j) [\Omega_{m-k}^\circ]_{j,1}) \\ \quad + \sum_{k=1}^m \beta_k [\Omega_{k-1}^\circ]_{1,1} + b, \\ \vdots \\ p_m = \sum_{j=2}^N \alpha_j ([\Omega_m^{m-1}]_{j,1} - \sum_{k=1}^m f_k(t_j) [\Omega_{m-k}^{m-1}]_{j,1}) + \sum_{k=1}^m \beta_k [\Omega_{k-1}^{m-1}]_{1,1}, \\ \circ = \sum_{j=2}^N \alpha_j f_m(t_j) - \beta_1. \end{cases}$$

□

مدل LS-SVM برای جواب و مشتقات آن می‌تواند در فرم دوگان به صورت زیر نوشته شود:

$$\begin{aligned} \hat{y}(t) &= \sum_{i=2}^N \alpha_i \left([\nabla_m^\circ K](t_i, t) - \sum_{k=1}^m f_k(t_i) [\nabla_{m-k}^\circ K](t_i, t) \right) \\ &\quad + \sum_{k=1}^m \beta_k [\nabla_{k-1}^\circ K](t_1, t) + b, \\ \frac{d^p \hat{y}(t)}{dt^p} &= \sum_{i=2}^N \alpha_i \left([\nabla_m^p K](t_i, t) - \sum_{k=1}^m f_k(t_i) [\nabla_{m-k}^p K](t_i, t) \right) \\ &\quad + \sum_{k=1}^m \beta_k [\nabla_{k-1}^p K](t_1, t), \quad p = 1, \dots, m-1. \end{aligned}$$

ت: فرمول‌بندی برای مسأله ODE غیرخطی

در این بخش، یک مسأله بهینه‌سازی را بر اساس LS-SVM برای حل ODE‌های مرتبه اول غیرخطی به صورت زیر می‌سازیم:

$$y' = f(t, y), \quad y(a) = p_1, \quad a \leq t \leq c. \quad (21.2)$$

با فرض این که جواب تقریبی به صورت $\hat{y}(t) = w^T \varphi(t) + b$ است شروع می‌کنیم. مجهولات اضافی y_i برای حفظ قیود خطی در w معرفی می‌شوند. مسأله بهینه‌سازی غیرخطی زیر را ارائه می‌دهیم:

$$\begin{aligned} &\underset{w, b, e, \epsilon, y_i}{\text{minimize}} \quad \frac{1}{2} w^T w + \frac{\gamma}{2} e^T e + \frac{\gamma}{2} \xi^T \xi \\ &\text{subject to} \quad w^T \varphi'(t_i) = f(t_i, y_i) + e_i, \quad i = 2, \dots, N. \\ &\quad w^T \varphi(t_1) + b = p_1, \\ &\quad y_i = w^T \varphi(t_i) + b + \xi_i, \quad i = 2, \dots, N. \end{aligned}$$

۳۶ حل تقریبی معادلات دیفرانسیل معمولی با ماشین بردار پشتیبان کمترین مربعات

تابع لاگرانژ برای مسأله بهینه‌سازی مقید بالا، به صورت زیر است:

$$\begin{aligned} L(w, b, e_i, \varepsilon_i, y_i, \alpha_i, \eta_i, \beta) = & \frac{1}{\gamma} w^T w + \frac{\gamma}{\gamma} e^T e + \frac{\gamma}{\gamma} \xi^T \xi \\ & - \sum_{i=2}^N \alpha_i (w^T \varphi(t_i) - f(t_i, y_i) - e_i) \\ & - \beta (w^T \varphi(t_1) + b - p_1) \\ & - \sum_{i=2}^N \eta_i (y_i - w^T \varphi(t_i) - b - \xi_i). \end{aligned}$$

پس از به دست آوردن شرایط مطلوب KKT و حذف متغیرهای اولیه w و $\{e_i\}_{i=2}^N, \{\xi_i\}_{i=2}^N$ ، و با استفاده از قضیه مرسر، جواب در فرم دوگان با حل دستگاه غیرخطی زیر به دست می‌آید:

$$\begin{bmatrix} \hat{\Omega}_1^\circ & \tilde{\Omega}_1^\circ & h_1^T & \circ_{N-1} & \circ_{(N-1) \times (N-1)} \\ (\tilde{\Omega}_1^\circ)^T & \hat{\Omega}_1^\circ & h_{\circ}^T & \mathbf{1}_{N-1} & -I_{N-1} \\ h_1 & h_{\circ} & [\Omega_1^\circ]_{1,1} & \mathbf{1} & \circ_{N-1}^T \\ \circ_{N-1}^T & I_{N-1}^T & \mathbf{1} & \circ & \circ_{N-1}^T \\ D(y) & I_{N-1} & \circ_{N-1} & \circ_{N-1} & \circ_{(N-1) \times (N-1)} \end{bmatrix} \cdot \begin{bmatrix} \alpha \\ \eta \\ \beta \\ b \\ y \end{bmatrix} = \begin{bmatrix} f(y) \\ \circ_{N-1} \\ p_1 \\ \circ \\ \circ_{N-1} \end{bmatrix} \quad (22.2)$$

که در آن:

$$\begin{aligned} \hat{\Omega}_1^\circ &= \tilde{\Omega}_1^\circ + I_{N-1}/\gamma, \quad \hat{\Omega}_\circ^\circ = \tilde{\Omega}_\circ^\circ + I_{N-1}/\gamma, \\ D(y) &= \text{diag}(f'(y)), \\ f(y) &= [f(t_2, y_2), \dots, f(t_N, y_N)]^T, \\ f'(y) &= \left[\frac{\partial f(t, y)}{\partial y} \Big|_{t=t_2, y=y_2}, \dots, \frac{\partial f(t, y)}{\partial y} \Big|_{t=t_N, y=y_N} \right], \\ \alpha &= [\alpha_2, \dots, \alpha_N]^T, \quad \eta = [\eta_2, \dots, \eta_N]^T, \\ y &= [y_2, \dots, y_N]^T, \quad \tilde{\Omega}_\circ^\circ = [\Omega_\circ^\circ]_{2:N, 2:N}, \\ \tilde{\Omega}_1^\circ &= [\Omega_1^\circ]_{2:N, 2:N}, \quad \tilde{\Omega}_\circ^\circ = [\Omega_\circ^\circ]_{2:N, 2:N}, \\ h_{\circ} &= [[\Omega_\circ^\circ]_{1,2}, \dots, [\Omega_\circ^\circ]_{1,N}], \\ h_1 &= [[\Omega_1^\circ]_{1,2}, \dots, [\Omega_1^\circ]_{1,N}], \\ O_{N-1} &= [\circ, \dots, \circ]^T \in \mathbb{R}^{N-1}. \end{aligned} \quad (23.2)$$

سیستم غیرخطی (۲۳.۲) که شامل معادلات $3N - 1$ با $3N - 1$ مجهول $(y, b, \beta, \eta, \alpha)$ است، با روش نیوتن حل می‌شود. مدل در فرم زیر دوگانه می‌شود:

$$\hat{y}(t) = \sum_{i=2}^N \alpha_i [\nabla_1^\circ K](t_i, t) + \sum_{i=2}^N \eta_i [\nabla_\circ^\circ K](t_i, t) + \beta [\nabla_\circ^\circ K](t_1, t) + b. \quad (24.2)$$

۵.۲ اجرای عملیات و انتخاب مدل

آ: جواب در فاصله زمانی طولانی

در حال حاضر شرایطی را که در آن یک معادله دیفرانسیل داده شده برای یک فاصله زمانی بزرگ $[a, c]$ باید حل شود بررسی می‌کنیم. لازم به ذکر است که برای بهبود دقت، پس از آن باید تعداد نقاط هم افزایش دهیم. این رویکرد منجر به یک سیستم معادلات با بعد بالاتر می‌شود. برای پیاده‌سازی روش پیشنهادی برای حل مسائل مربوط به فواصل زمان بندی بزرگ موثر، روش تجزیه دامنه کاربرد دارد [۷].

ابتدا دامنه $\Xi = [a, c]$ ، به بخش‌های S ، برد دامنه به صورت $\Xi = \cup_{k=1}^S \Xi_k$ ، تجزیه می‌شود. فرض کنیم که جواب تقریبی در k - امین زیربخش به صورت $\hat{y}_k(t) = w_k^T \varphi(t) + b_k$ است. سپس مسأله در هر زیر دامنه k با استفاده از روش توصیف شده در بخش‌های قبلی حل می‌شود. جواب تقریبی محاسبه شده در نقطه نهایی در زیر دامنه Ξ به عنوان نقطه شروع (مقدار اولیه) برای جواب در زیر دامنه Ξ_k و Ξ_{k+1} استفاده می‌شود.

استفاده از این رویکرد منجر به حل S سیستم‌های کوچک‌تر معادلات خواهد شد، که از لحاظ محاسباتی کارآمدتر از حل یک سیستم بسیار بزرگ معادلات حاصل شده با در نظر گرفتن کل دامنه (با همان تعداد کل نقاط هم‌پوشانی) است. این روش در الگوریتم ۱ مشخص شده است.

الگوریتم ۱: تقریب جواب در یک فاصله بزرگ

-
- 1: Decompose the domain $\Xi = [a, c]$ into S sub-domains.
 - 2: set $\Gamma = (c - a)/S$, $t_{in} := a$, $y_{in} := p_1$, $t_f := t_{in} + \Gamma$.
 - 3: for $k = 1$ to S do
 - 4: Obtain a LS-SVM model for the k -th segment $[t_{in}, t_f]$
i.e., $\hat{y}_k(t) = w_k^T \varphi_k(t) + b_k$.
 - 5: set $t_{in} := t_f$, $y_{in} := \hat{y}(t_f)$, $t_f := t_{in} + \Gamma$
 - 6: end for
 - 7: For a given test point t :
 - 1) check to which segment it belongs;
 - 2) use the corresponding model to compute the approximate solution at the given point.
-

ب : پارامتر تنظیم

عملکرد مدل LS-SVM به انتخاب پارامترهای تنظیم بستگی دارد. در این‌جا، برای تمام آزمایش‌ها هسته RBF گاوس استفاده می‌شود. بنابراین، یک مدل توسط پارامتر تنظیم γ و پهنای باند هسته σ تعیین می‌شود. یک مقدار کاملاً بزرگ باید برای ثابت γ در نظر گرفته

۳۸ حل تقریبی معادلات دیفرانسیل معمولی با ماشین بردار پشتیبان کمترین مربعات

شود، به طوری که خطای e به شدت به حداقل می‌رسد و یا معادل آن، محدودیت‌ها به خوبی برآورده می‌شود.

در تمام آزمایشات، مقدار انتخاب شده برای γ مقدار 10^{-1} بود، به جز برای مسأله با بازه بزرگ که برای جلوگیری از بدوضعی 10^{-5} انتخاب می‌شود. بنابراین، تنها پارامتر سمت چپ که باید تنظیم شود، پهنای باند هسته است. در این فصل، مقادیر بهینه σ با استفاده از محدوده معنی‌دار امکان σ به دست می‌آید.

مجموعه اعتبارسنجی مجموعه‌ای از نقاط میانه‌ای $V = \{v_i = \frac{(t_i + t_{i+1})}{4}, i = 1, \dots, N-1\}$ است که در آن $\{t_i\}_{i=1}^N$ نقاط آموزش هستند. سپس مقادیری که خطای متوسط مربع (MSE) را به حداقل برساند در این مجموعه اعتبارسنجی انتخاب شده است.

ملاحظه ۱.۵.۲. در بعضی موارد، یک مقدار فوق‌العاده بزرگ برای γ به طور معمول بیش از 10^7 می‌تواند ماتریس را در موارد گفته‌شده، نزدیک به معکوس ناپذیر بودن کند.

۶.۲ نتایج عددی

در این بخش، عملکرد روش پیشنهادی را در مورد هفت مسأله، چهار مرتبه اول و سه مرتبه دوم بررسی می‌کنیم. برای سه مسأله اول و مسأله ۵، مقایسه‌ای بین جواب‌های به دست آمده در [۲۵] و جواب‌های روش پیشنهادی انجام شده است. نتایج عددی مسأله ۴ و ۶ با آنچه [۲۹] ارائه شده مقایسه شده است. مسأله ۷، که هیچ جواب تحلیلی ندارد و یک مسأله منفرد است، حل شده است و جواب محاسبه شده با آنچه که در [۲۷] گزارش شده است مقایسه می‌شود. برای نشان دادن قابلیت تقریبی و قابلیت تعمیم روش پیشنهادی، جواب دقیق را با جواب محاسبه شده درون و بیرون دامنه مورد بررسی مقایسه می‌کنیم. علاوه بر این، روش پیشنهاد شده با موفقیت برای حل مسأله ۱ برای یک فاصله زمانی بسیار زیاد مورد استفاده قرار می‌گیرد. برای تمام آزمایش‌ها هسته RBF زیر استفاده می‌شود. بنابراین داریم:

$$\begin{aligned}\nabla_1^\circ[K(u, v)] &= -\frac{2(u-v)}{\sigma^2}K(u, v) \\ \nabla_1^1[K(u, v)] &= \frac{2(u-v)}{\sigma^2}K(u, v) \\ \nabla_2^\circ[K(u, v)] &= \left[\frac{4(u-v)^2}{\sigma^4} - \frac{2}{\sigma^2} \right] K(u, v).\end{aligned}$$

آ: معادلات دیفرانسیل معمولی مرتبه اول

مسأله ۱: معادلات دیفرانسیل معمولی مرتبه اول زیر را در نظر بگیرید:

$$\frac{d}{dt}y(t) + 2y(t) = \sin(t) \quad y(0) = 1, \quad t \in [0, 10].$$

جواب تقریبی به دست آمده از روش پیشنهادی با جواب دقیق مقایسه شده است و نتایج در شکل (۱.۲) نشان داده شده است. از نتایج به دست آمده، واضح است که روش پیشنهادی از نظر دقت بهتر از روش گفته شده در [۲۵] است. علاوه بر این، همچنین نقاط خارج از دامنه آموزش را مورد توجه قرار دادیم، و شکل (۱.۲)، (d) و (e) نشان می‌دهند که خطای مطلق برای نقاط در نزدیکی دامنه معادله کم است. همان‌طور که انتظار می‌رود، با افزایش تعداد نقاط آموزش، خطا در داخل و خارج از دامنه آموزش کاهش می‌یابد. شکل (۱.۲)، (c) و (f) عملکرد روش را نشان می‌دهد زمانی که گسسته‌سازی غیریکنواخت برای ایجاد نقاط آموزش استفاده می‌شود.

مسئله ۲: معادله دیفرانسیل مرتبه اول با تحریک سینوسی غیرخطی را به صورت زیر در نظر بگیرید:

$$\frac{d}{dt}y(t) + 2y(t) = t^3 \sin\left(\frac{t}{\pi}\right) \quad y(0) = 1, \quad t \in [0, 10].$$

فاصله $[0, 10]$ به 20 نقطه‌ی $t_1 = 0, \dots, t_{20} = 10$ با استفاده از $t_i = (i-1)h, i = 1, \dots, N$ زمانی که $h = \frac{10}{N-1}$ تقسیم شده است. در شکل (۲.۲)، (a) جواب دقیق را با جواب محاسبه شده در نقاط شبکه و همچنین نقاط دیگر داخل و خارج از دامنه معادله مقایسه می‌کنیم. خطاهای مطلق به دست آمده برای نقاط داخل و خارج از حوزه $[0, 10]$ در جدول (۱.۲) بیان شده است. واضح است که جواب به دست آمده دقت بیشتری نسبت به جواب به دست آمده [۲۵] دارد. با وجود اینکه تعداد نقاط آموزشی کمتر استفاده شده است.

100 نقطه با فاصله مساوی برای آموزش مورد استفاده قرار می‌گیرد و حداکثر خطای مطلق نشان داده شده در روابط گفته شده در [۲۵] حدود $10^{-2} \times 25$ است.

مسئله ۳: معادله دیفرانسیل معمولی مرتبه اول غیرخطی زیر را در نظر بگیرید که هیچ جواب تحلیلی ندارد:

$$\frac{d}{dt}y(t) = y(t)^2 + t^2, \quad y(0) = 1, \quad t \in [0, 0.5].$$

بیست نقطه با فاصله مساوی در فاصله زمانی داده شده برای فرآیند آموزش استفاده می‌شود. جواب تقریبی به دست آمده از روش پیشنهادی و جواب به دست آمده توسط MATLAB ساخته شده در حل کننده ODE45 در شکل (۲.۲)، (b) نمایش داده می‌شود. خطاهای مطلق به دست آمده برای نقاط داخل و خارج از دامنه $[0, 0.5]$ در جدول (۱.۲) خلاصه شده‌اند. روش پیشنهادی عملکرد بهتر را نسبت به روش توصیف شده از لحاظ دقت نشان می‌دهد، با وجود این که تعداد نقاط آموزشی کمتر مورد استفاده قرار می‌گیرد.

۴۰ حل تقریبی معادلات دیفرانسیل معمولی با ماشین بردار پشتیبان کمترین مربعات

مسأله	دامنه	$\ y - \hat{y}\ _\infty$	MSE	STD
۲	داخلی	$4/56 \times 10^{-3}$	$1/47 \times 10^{-6}$	$1/16 \times 10^{-3}$
	خارجی	$4/62 \times 10^{-1}$	$3/85 \times 10^{-2}$	$1/56 \times 10^{-1}$
۳	داخلی	$5/43 \times 10^{-3}$	$8/94 \times 10^{-6}$	$1/60 \times 10^{-3}$
	خارجی	$8/46 \times 10^{-2}$	$1/49 \times 10^{-3}$	$2/27 \times 10^{-2}$
۴	داخلی	$1/46 \times 10^{-4}$	$8/15 \times 10^{-9}$	$3/90 \times 10^{-5}$
	خارجی	$6/76 \times 10^{-2}$	$5/53 \times 10^{-4}$	$2/20 \times 10^{-2}$

جدول ۱.۲: نتایج عددی روش پیشنهادی برای حل مسائل ۲-۴

نکته: MSE میانگین خطای کمترین مربعات است. و STD انحراف استاندارد است.
مسأله ۴: معادله دیفرانسیل مرتبه اول را با ضریب متغیر با زمان در نظر بگیرید:

$$\frac{d}{dt}y(t) + \left(t + \frac{1+3t^2}{1+t+t^3}\right)y(t) = t^3 + 2t + t^2 \frac{1+3t^2}{1+t+t^3},$$

$$y(0) = 1, \quad t \in [0, 1].$$

به منظور مقایسه با نتایج گزارش شده در [۲۹] ده نقطه با فاصله مساوی در بازه داده شده برای روند آموزش استفاده می‌شود. جواب تحلیلی و جواب به‌دست آمده از طریق روش پیشنهادی در شکل (۲.۲)، (c) نمایش داده می‌شوند. خطاهای مطلق به‌دست آمده برای نقاط در داخل و خارج از دامنه $[0, 1]$ در جدول (۱.۲) بیان شده‌است، که بهتر بودن روش پیشنهادی را بر روش توصیف شده در روابط گفته شده در [۲۹]، نشان می‌دهد.

ب: معادله دیفرانسیل معمولی مرتبه دوم

مسأله ۵: معادله دیفرانسیل معمولی مرتبه دوم زیر را با سیگنال ورودی متفاوت متغیر در نظر بگیرید:

$$\frac{d^2}{dt^2}y(t) + y(t) = 2 + 2 \sin(4t) \cos(3t),$$

$$y(0) = 1, \quad y(1) = 0.$$

ده نقطه با فاصله مساوی در بازه زمانی داده شده برای فرآیند آموزش استفاده می‌شود. جواب تحلیلی و جواب به‌دست آمده از طریق روش پیشنهادی در شکل (۳.۲)، (a) نمایش داده می‌شود. خطاهای مطلق به‌دست آمده برای نقاط داخل و خارج از بازه $[0, 1]$ در جدول (۲.۲) بیان شده‌است، که برتری روش پیشنهادی را بر روش توصیف شده در [۲۵] نشان می‌دهد.

مسأله ۶: معادله دیفرانسیل معمولی مرتبه دوم با سیگنال ورودی متغیر با زمان را در نظر

بگیرید:

$$\frac{d^2}{dt^2}y(t) + \frac{1}{5} \frac{d}{dt}y(t) + y(t) = -\frac{1}{5} e^{\frac{-t}{5}} \cos(t),$$

$$y(0) = 1, \quad \dot{y}(0) = 1.$$

ده نقطه با فاصله مساوی در بازه $[0, 2]$ برای فرآیند آموزش استفاده می‌شود. جواب تحلیلی و جواب به‌دست آمده توسط روش پیشنهادی در شکل (۳.۲)، (b) نشان داده شده است. خطاهای مطلق به‌دست آمده برای نقاط داخل و خارج از بازه $[0, 2]$ در جدول (۲.۲) بیان شده‌اند که دوباره بهبود روش پیشنهادی را در مقایسه با روش توصیف شده در [۲۹] نشان می‌دهد. حداکثر خطای مطلق خارج از دامنه $[0, 2]$ ، 8×10^{-4} است.

مسأله ۷: معادله دیفرانسیل معمولی مرتبه دوم زیر را در نظر بگیرید که هیچ روش تحلیلی

برای یافتن جواب آن وجود ندارد:

$$\frac{d^2}{dt^2}y(t) + \frac{1}{t} \frac{d}{dt}y(t) - \frac{1}{t} \cos(t) = 0, \quad y(0) = 0, \quad y'(0) = 1,$$

$$y(t) = \int_0^t \frac{\sin(x)}{x} dx.$$

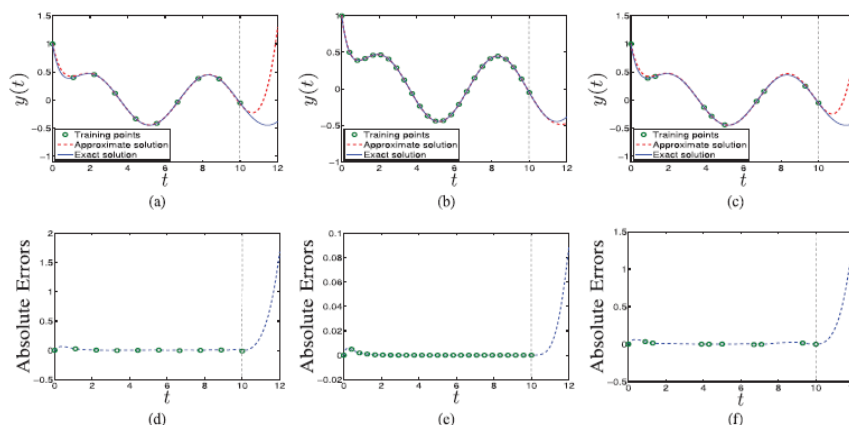
ده نقطه با فاصله مساوی در بازه $[0, 1]$ به‌عنوان نقاط آموزشی استفاده می‌شوند و نتایج به‌دست آمده در شکل (۳.۲)، (c) نشان داده شده و در جدول (۲.۲) بیان شده‌اند. حداکثر خطای مطلق خارج از دامنه $[0, 1]$ برابر $6/51 \times 10^{-2}$ است که کوچکتر از 14×10^{-1} در [۲۷] است.

مسأله	دامنه	متغیر	$\ y - \hat{y}\ _\infty$	MSE	STD
۵	داخلی	y	$1/14 \times 10^{-6}$	$4/16 \times 10^{-13}$	$6/43 \times 10^{-7}$
		y'	$4/81 \times 10^{-5}$	$6/78 \times 10^{-11}$	$8/21 \times 10^{-6}$
	خارجی	y	$4/20 \times 10^{-2}$	$2/64 \times 10^{-4}$	$1/26 \times 10^{-2}$
		y'	$1/00 \times 10^{-1}$	$3/27 \times 10^{-3}$	$3/87 \times 10^{-2}$
۶	داخلی	y	$5/88 \times 10^{-6}$	$1/49 \times 10^{-11}$	$1/63 \times 10^{-6}$
		y'	$7/34 \times 10^{-6}$	$2/18 \times 10^{-11}$	$3/28 \times 10^{-6}$
	خارجی	Y	$3/96 \times 10^{-4}$	$2/39 \times 10^{-8}$	$1/19 \times 10^{-4}$
		y'	$5/15 \times 10^{-4}$	$7/11 \times 10^{-8}$	$1/74 \times 10^{-4}$
۷	داخلی	y	$6/64 \times 10^{-9}$	$2/01 \times 10^{-16}$	$4/07 \times 10^{-9}$
		y'	$8/41 \times 10^{-8}$	$1/30 \times 10^{-15}$	$3/59 \times 10^{-8}$
	خارجی	y	$6/51 \times 10^{-2}$	$3/90 \times 10^{-4}$	$1/65 \times 10^{-2}$
		y'	$7/80 \times 10^{-2}$	$7/31 \times 10^{-4}$	$2/15 \times 10^{-2}$

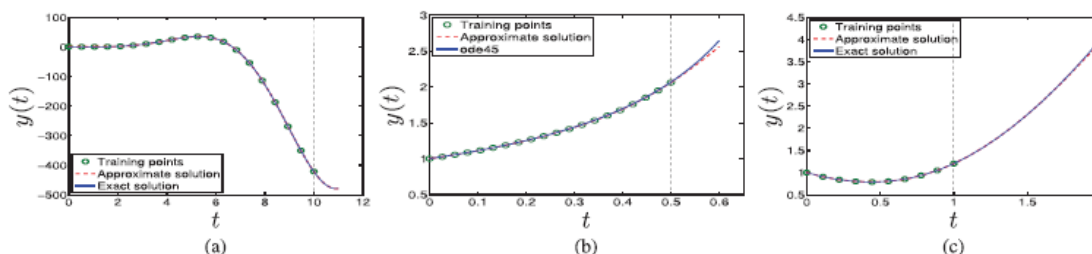
جدول ۲.۲: نتایج عددی روش پیشنهادی برای حل مسائل ۵-۷

۴۲ حل تقریبی معادلات دیفرانسیل معمولی با ماشین بردار پشتیبان کمترین مربعات

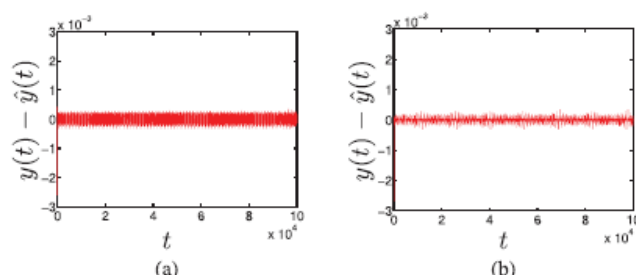
نکته: MSE میانگین خطای کمترین مربعات است. و STD انحراف استاندارد است.



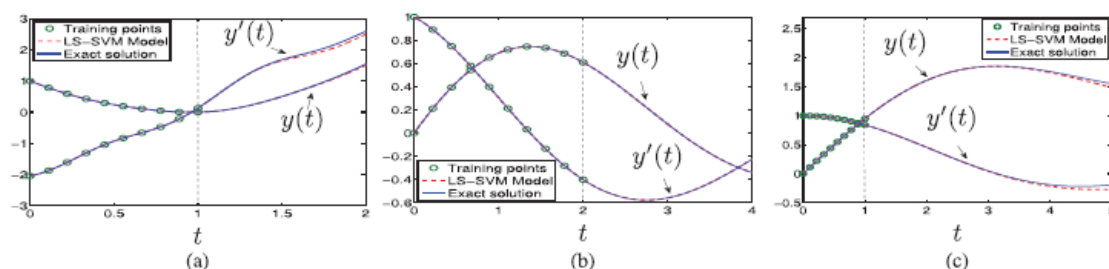
شکل ۱.۲: نتایج عددی برای مسأله ۱. (a) ۱۰ نقطه با فاصله مساوی در بازه $[0, 10]$ برای آموزش استفاده شده است. (b) ۲۵ نقطه با فاصله مساوی در بازه $[0, 10]$ برای آموزش استفاده شده است. (c) نقاط غیریکنواخت در بازه $[0, 10]$ ، ۱۰ نقطه برای آموزش استفاده شده است. (d) خطاهای مطلق به دست آمده در بازه $[0, 12]$ زمانی که آموزش استفاده شده است. (e) خطاهای مطلق به دست آمده در بازه $[0, 10]$ به ۹ قسمت مساوی تقسیم می‌شوند. (f) خطاهای مطلق به دست آمده در بازه $[0, 12]$ زمانی که به ۲۴ قسمت مساوی تقسیم می‌شوند. (g) خطاهای مطلق به دست آمده در بازه $[0, 10]$ زمانی که به ۹ قسمت غیریکنواخت تقسیم می‌شوند.



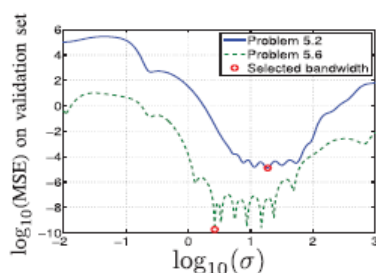
شکل ۲.۲: (a) نتایج عددی برای مسأله ۲، ۲۰ نقطه با فاصله مساوی در بازه $[0, 10]$ برای آموزش استفاده شده است. (b) نتایج عددی برای مسأله ۳، ۲۰ نقطه با فاصله مساوی در بازه $[0, 5]$ برای آموزش استفاده شده است. (c) نتایج عددی برای مسأله ۴، ۱۰ نقطه با فاصله مساوی در بازه $[0, 1]$ برای آموزش استفاده شده است.



شکل ۵.۲: a. باقیمانده $y(t) - \hat{y}(t)$ وقتی مسأله ۱ روی بازه $[0, 10^5]$ حل می‌شود. با استفاده از ۵۰۰۰ زیربازه و ۵۰ نقطه هم‌محلی. b. نتایج به‌دست آمده برای برخی مسائل با استفاده از ۵۰۰ زیربازه و ۵۰۰۰ نقطه هم‌محلی.



شکل ۳.۲: (a). نتایج عددی برای مسأله ۵، ۱۰ نقطه با فاصله مساوی در بازه $[0, 1]$ برای آموزش استفاده شده‌است. (b). نتایج عددی برای مسأله ۶، ۱۰ نقطه با فاصله مساوی در بازه $[0, 2]$ برای آموزش استفاده شده‌است. (c). نتایج عددی برای مسأله ۷، ۱۰ نقطه با فاصله مساوی در بازه $[0, 1]$ برای آموزش استفاده شده‌است.



شکل ۴.۲: میزان حساسیت نتایج به‌دست آمده پارامتر مدل σ . $\log_{10}(\sigma)$ برای مسأله ۲ و ۶ رسم شده‌است.

فصل ۳

حل مسأله کنترل بهینه با ماشین بردار پشتیبان کمترین مربعات

۱.۳ مقدمه

از آن جا که معرفی بازپرداخت از طریق زمان و بازپرداخت دینامیکی، با شیوه‌های مبتنی بر گرادیان یا جواب‌های عقب‌گردانی برای سیستم‌های دینامیکی با استفاده از شبکه‌های عصبی تمرکز کرده است، در میان بسیاری از رویکردهای مبتنی بر مدل، مسأله کنترل مرحله، در ترکیب با چندین ساختار برای کنترل عصبی مورد مطالعه قرار گرفته است. به‌عنوان مثال، نگوین و ویدرو^۱ (۱۹۹۰) اعمال بازگشتی به مسأله پشتیبان‌گیری کامیون تریلر بر اساس یک شبیه‌ساز شبکه عصبی برای این کارخانه را اعمال کردند.

در بازخورد کامل یک کنترل کننده در حالت استاتیکی در زمینه کنترل بهینه مورد مطالعه قرار گرفته است. مطالعه موردی بهینه زمان واقعی در سویکنز^۲ و وندوال^۳ و دی‌مور^۴ مسأله چرخش یک آونگ معکوس و آونگ چرخشی معکوس با ثبات در نقطه پایانی به‌عنوان

^۱Nguyen and Widrow

^۲Suykens

^۳Vandewalle

^۴De Moor, 1996

یک مسأله بهینه‌سازی پارامتری در وزن مجهول کنترل کننده‌های عصبی فوری یا معمولی می‌باشد. در این فصل در مورد حل مسأله کنترل بهینه N -مرحله‌ای روش کمترین مربعات پشتیبان^۵ بحث می‌کنیم. ماشین‌های بردار پشتیبان اخیراً برای حل مسأله تشخیص الگو و مسائل برآورد عملکرد تخمین تابع معرفی شده‌اند.

در این روش تشخیص یک داده را به یک فضای ورودی بزرگتر می‌نگارد و یک ابرصفحه جداکننده در این فضا می‌سازد. از این طریق، قضیه مرسر را مورد استفاده قرار می‌دهد. سپس جواب به صورت یک مجموع وزن دار از نقاط داده نوشته می‌شود. در فرمول بندی اصلی از مسأله، یک مسأله برنامه‌ریزی درجه دوم حل می‌شود که مقدار زیادی وزن صفر دارد. نقاط داده مربوط به وزن‌های غیر صفر، بردارهای پشتیبان نامیده می‌شوند. پارامترهای تابع هسته را می‌توان طوری انتخاب کرد که در خطای کلی به حداقل برسد.

یکی از این‌ها امکان استفاده از اسپلاین‌ها^۶، شبکه‌های رادیال (RBF) یا چندجمله‌ای پرسپترون‌ها^۷ به عنوان هسته است. با در نظر گرفتن اصل کاهش ریسک ساختار و مفهوم ظرفیت با ترکیبات ترکیبی خالص، کیفیت و پیچیدگی جواب با کمترین مربعات پشتیبان SVM به طور مستقیم به ابعاد فضای ورودی بستگی ندارد.

در روش کنترل SVM از ماشین‌های بردار پشتیبان استفاده می‌کنیم. در اصل، از تابع ϵ -حساسیت^۸ و پنیک^۹ برای مسأله تقویت تابع استفاده شده است.

از طرف دیگر، مسأله کمترین مربعات توسط ساندرز^{۱۰} و گمرمن^{۱۱} و ووک^{۱۲} و سویکنز^{۱۳} و وندروال^{۱۴} برای مسأله‌های تخمین تابع، در روش کنترل بهینه با LS-SVM، مسأله کنترل مطلوب N -مرحله‌ای و مسأله بهینه‌سازی مربوط به کنترل کننده LS-SVM در یک فرمول بندی مسأله گنجانده شده است. در اولی تشخیص یک مسأله بهینه‌سازی پارامتری با وزن‌های مجهول حل می‌شود که در آن بردار وضعیت تحقق شدن وزن‌ها تعیین می‌شود. هر چند یک نقص این روش انتخاب تعداد لایه‌های پنهان مورد نیاز برای تعیین تابع کنترل است. هم‌چنین در شبکه‌های RBF زمانی که شخص نقاط منظمی برای مراکز انتخاب می‌کند از معضل بعد بالا رنج می‌برد. در روش کنترل SVM مراکز از مسیر بهینه‌ای که شخص به دست می‌آورد متابعت می‌کند. به علاوه روش‌های استاندارد شبکه عصبی فقط در فضای اولیه کار می‌کند در حالی که، روش‌های SVM در فضای دوگان بررسی می‌شوند به طوری که

^۵(LS-SVM)

^۶Splines

^۷Perceptrons

^۸ ϵ -insensitive

^۹Vapnik

^{۱۰}Saunders

^{۱۱}Gamerman

^{۱۲}Vovk

^{۱۳}Suykens

^{۱۴}Vandewalle

تعداد مجهولات مساوی تعداد نقاط آموزش داده شده است، نه تعداد وزن‌های موجود در فضای اولیه که می‌تواند بی‌نهایت باشد.

مطالب این فصل به شرح زیر است: در بخش ۲ مسأله کنترل مطلوب N - مرحله را بیان می‌کنیم. در بخش ۳ کار بر روی ماشین‌های بردار پشتیبان را بررسی می‌کنیم. در بخش ۴، کنترل بهینه توسط مسأله کمترین مربعات ماشین بردار پشتیبان را توصیف می‌کنیم.

۲.۳ مسأله کنترل مطلوب N - مرحله

مسأله کنترل بهینه N - مرحله‌ای زیر را در نظر بگیرید:

$$\min \mathcal{J}_N(x_k, u_k) = \rho(x_{N+1}) + \sum_{k=1}^N h(x_k, u_k). \quad (1.3)$$

تحت سیستم‌های دینامیکی

$$x_{k+1} = f(x_k, u_k), \quad k = 1, \dots, N \quad (x_1 \text{ معلوم})$$

که $\rho(\cdot) \in \mathbb{R}^n$ و $h(\cdot, \cdot)$ توابع مثبتی هستند. یک نمونه انتخاب تابع درجه دوم $R = R^T > 0$ و $Q = Q^T > 0$ با $\rho(x_{N+1}) = x_{N+1}^T Q x_{N+1}$ و $h(x_k, u_k) = x_k^T Q x_k + u_k^T R u_k$ است. فرض می‌شود توابع $\rho(\cdot)$ ، $h(\cdot, \cdot)$ و $f(\cdot, \cdot)$ دوبار بطور پیوسته مشتق‌پذیر هستند. $x_k \in \mathbb{R}^n$ نشان‌دهنده‌ی بردار حالت است، $u_k \in \mathbb{R}$ ورودی سیستم است. روش‌های مورد بحث در این فصل تنها به سیستم‌های تک ورودی محدود نمی‌شوند. برای به‌دست‌آوردن قانون کنترل بهینه، یک لاگرانژ را به‌صورت زیر در نظر می‌گیریم:

$$\mathcal{L}_N(x_k, u_k; \lambda_k) = \mathcal{J}_N(x_k, u_k) + \sum_{k=1}^N \lambda_k^T [x_{k+1} - f(x_k, u_k)], \quad (2.3)$$

که λ_k ها ضرایب لاگرانژ هستند. شرایط به‌صورت زیر است:

$$\begin{cases} \frac{\partial \mathcal{L}_N}{\partial x_k} = \frac{\partial h}{\partial x_k} + \lambda_{k-1} - \left(\frac{\partial f}{\partial x_k} \right)^T \lambda_k = 0, & k = 2, \dots, N \quad (\text{معادله الحاقی}) \\ \frac{\partial \mathcal{L}_N}{\partial x_{N+1}} = \frac{\partial \rho}{\partial x_{N+1}} + \lambda_N = 0, & (\text{شرط‌های الحاقی}) \\ \frac{\partial \mathcal{L}_N}{\partial u_k} = \frac{\partial h}{\partial u_k} - \lambda_k^T \frac{\partial f}{\partial u_k} = 0, & k = 1, \dots, N \quad (\text{شرط تغییراتی}) \\ \frac{\partial \mathcal{L}_N}{\partial \lambda_k} = x_{k+1} - f(x_k, u_k) = 0, & k = 1, \dots, N \quad (\text{دستگاه دینامیکی}) \end{cases} \quad (3.3)$$

برای حالت یک تابع درجه دوم تحت شرایط سیستم دینامیکی خطی با افق زمانی نامحدود، قانون کنترل بهینه می‌تواند با کنترل بازخورد کامل نمایش داده شود. با این حال، به‌طور کلی قانون کنترل مطلوب نمی‌تواند توسط بازخورد حالت ارائه شود، قانون کنترل بهینه ممکن

است به متغیرهای الحاقی هم بستگی داشته باشد. با این وجود، اغلب علاقه‌مند به پیدا کردن یک راهبرد برای یافتن کنترل بهینه زیر هستیم:

$$u_k = g(x_k), \quad (۴.۳)$$

که $g(\cdot)$ توسط یک معماری شبکه عصبی پارامتری شده است در این فصل، یک قانون کنترل در معادله (۴.۳) را برای ماشین‌های بردار پشتیبان بررسی می‌کنیم. از آنجایی که روش SVM یک رویکرد مدل‌سازی پارامتری نیست، این رویکرد در مقایسه با شبکه‌های عصبی استاندارد مثل MLP و RBF بسیار سرراست است.

۳.۳ روش بردار پشتیبان برای تخمین تابع

مجموعه‌ای از توابع رگرسیون زیر را در نظر بگیرید:

$$\mathcal{F}(\mathcal{X}) = \mathcal{W}^T \varphi(\mathcal{X}) + \mathcal{B}, \quad (۵.۳)$$

با داده‌های آموزشی $\{\mathcal{X}_i, \mathcal{Y}_i\}_{i=1}^M$ که M تعداد داده‌ها را نشان می‌دهد. $\mathcal{X}_i \in \mathbb{R}^m$ داده‌های ورودی و $\mathcal{Y}_i \in \mathbb{R}$ داده‌های خروجی هستند. نگاشت غیرخطی $\varphi: \mathbb{R}^m \rightarrow \mathbb{R}^{n_h}$ داده‌های ورودی را به یک فضای بعد بالا (که می‌تواند ابعاد بی‌نهایت باشد) می‌نگارد که در آن $\mathcal{B} \in \mathbb{R}$ و $\mathcal{W} \in \mathbb{R}^{n_h}$ در روش بردار پشتیبان، هدف حداقل کردن خطر تجربی^{۱۵} است.

$$\mathcal{R}_{emp}(\mathcal{W}, \mathcal{B}) = \frac{1}{M} \sum_{i=1}^M |Y_i - \mathcal{W}^T \varphi(\mathcal{X}_i) - \mathcal{B}|_{\varepsilon}. \quad (۶.۳)$$

تحت عناصری از φ_n ، $\mathcal{W}^T \mathcal{W} \leq c_n$ تعریف می‌شود. با به کار بردن تابع ε -حساسیت^{۱۶} وپنیک به عنوان تابع زیان به صورت:

$$|\mathcal{Y} - \mathcal{F}(\mathcal{X})|_{\varepsilon} = \begin{cases} 0, & \text{اگر } |\mathcal{Y} - \mathcal{F}(\mathcal{X})| \leq \varepsilon \\ |\mathcal{Y} - \mathcal{F}(\mathcal{X})| - \varepsilon, & \text{سایر نقاط.} \end{cases} \quad (۷.۳)$$

مسأله تخمین به صورت مسأله بهینه‌سازی زیر ارائه می‌شود:

$$\min_{\mathcal{W}, \mathcal{B}, \xi^*, \xi} \mathcal{J}_{\varepsilon}(\mathcal{W}, \xi^*, \xi) = \frac{1}{2} \mathcal{W}^T \mathcal{W} + \gamma \left\{ \sum_{i=1}^M \xi_i^* + \sum_{i=1}^M \xi_i \right\} \quad (۸.۳)$$

^{۱۵} Empirical risk

^{۱۶} ε -insensitive

با توجه به محدودیت‌ها

$$\begin{cases} \mathcal{Y}_i - \mathcal{W}^T \varphi(\mathcal{X}_i) - \mathcal{B} \leq \varepsilon + \xi_i^*, & i = 1, \dots, M \\ -\mathcal{Y}_i + \mathcal{W}^T \varphi(\mathcal{X}_i) + \mathcal{B} \leq \varepsilon + \xi_i, & i = 1, \dots, M \\ \xi_i^* \geq 0, & i = 1, \dots, M \\ \xi_i \geq 0, & i = 1, \dots, M \end{cases}$$

که در آن ξ_i^* و ξ_i متغیرهای کمبود^{۱۷} و γ ثابت مثبت است. در این صورت داریم:

$$\mathcal{W} = \sum_{i=1}^M (\alpha_i^* - \alpha_i) \varphi(\mathcal{X}_i),$$

که در آن α_i^* و α_i با حل یک مسأله برنامه‌ریزی درجه دوم به‌دست می‌آیند و ضرایب لاگرانژ متناظر با محدودیت‌های اول و دوم هستند. نقاط داده مربوط به مقادیر غیرصفر برای $(\alpha_i^* - \alpha_i)$ بردارهای پشتیبان نامیده می‌شوند. به‌طور معمول، بسیاری از این مقادیر برابر با صفر هستند. در نهایت، مدل زیر در فضای دوگان به‌دست می‌آید:

$$\mathcal{F}(\mathcal{X}) = \sum_{i=1}^M (\alpha_i^* - \alpha_i) K(\mathcal{X}_i, \mathcal{X}), \quad (9.3)$$

که تابع هسته K بر مبنای شرایط مرسر^{۱۸} به‌صورت زیر است:

$$K(\mathcal{X}_i, \mathcal{X}) = \varphi(\mathcal{X}_i)^T \varphi(\mathcal{X}), \quad (10.3)$$

گزینه‌های متعددی برای انتخاب این تابع هسته شامل حالت خطی، چندجمله‌ای‌ها^{۱۹}، اسپلاین‌ها،^{۲۰} RBF وجود دارند. در ادامه این فصل بر هسته RBF تمرکز خواهیم کرد. نفوذ تعداد بردارهای پشتیبان بر تابع تعمیم در نظرسنجی در وپنیک (۱۹۹۵)، مورد بررسی قرار گرفته‌است. یک توسعه برای انتخاب تابع زیان ε - حساسیت به‌صورت زیر است:

$$\mathcal{J}_{\varepsilon,p}(\mathcal{W}, \xi^*, \xi) = \frac{1}{\gamma} \mathcal{W}^T \mathcal{W} + \gamma \left\{ \sum_{i=1}^M (\xi_i^*)^p + \sum_{i=1}^M (\xi_i)^p \right\}, \quad (11.3)$$

که با $p = 1$ معادله‌ی (۸.۳) به‌دست می‌آید. برای $p = 2$ تابع رگرسیون زیر به‌دست می‌آید:

$$\min_{\mathcal{W}, \beta, \xi} \mathcal{J}_{LS}(\mathcal{W}, \beta, \xi) = \frac{1}{\gamma} \mathcal{W}^T \mathcal{W} + \gamma \frac{1}{\gamma} \sum_{i=1}^M \xi_i^2 \quad (12.3)$$

^{۱۷} slack

^{۱۸} Mercer's condition

^{۱۹} Polynomail

^{۲۰} Spliens

تحت قیود تساوی

$$\mathcal{Y}_i = \mathcal{W}^T \varphi(\mathcal{X}_i) + \beta + \xi_i, \quad i = 1, \dots, M.$$

تابع لاگرانژین را به صورت زیر تعریف می کنیم:

$$\mathcal{L}_{LS}(\mathcal{W}, \beta, \xi; \alpha) = \mathcal{J}_{LS}(\mathcal{W}, \mathcal{B}, \xi) - \sum_{i=1}^M \alpha_i (\mathcal{W}^T \varphi(\mathcal{X}_i) + \mathcal{B} + \xi_i - \mathcal{Y}_i), \quad (13.3)$$

که در آن α_i ضرایب لاگرانژ هستند.

شرایط بهینگی کاروش-کان-تاکر به صورت زیر است:

$$\begin{cases} \frac{\partial \mathcal{L}_{LS}}{\partial \mathcal{W}} = 0 \rightarrow \mathcal{W} = \sum_{i=1}^M \alpha_i \varphi(\mathcal{X}_i), \\ \frac{\partial \mathcal{L}_{LS}}{\partial \mathcal{B}} = 0 \rightarrow \sum_{i=1}^M \alpha_i = 0, \\ \frac{\partial \mathcal{L}_{LS}}{\partial \xi_i} = 0 \rightarrow \alpha_i = \gamma \xi_i, & i = 1, \dots, M \\ \frac{\partial \mathcal{L}_{LS}}{\partial \alpha_i} = 0 \rightarrow \mathcal{W}^T \varphi(\mathcal{X}_i) + \mathcal{B} + \xi_i - \mathcal{Y}_i = 0, & i = 1, \dots, M \end{cases} \quad (14.3)$$

پس از حذف $\mathcal{W}$ و ξ_i ، دستگاه (۱۴.۳) را می توان به صورت زیر نوشت:

$$\begin{bmatrix} 0 & \vec{1}^T \\ \vec{1} & \Omega + \gamma^{-1} I \end{bmatrix} \begin{bmatrix} \mathcal{B} \\ \alpha \end{bmatrix} = \begin{bmatrix} 0 \\ \mathcal{Y} \end{bmatrix} \quad (15.3)$$

با

$$\mathcal{X} = [\mathcal{X}_1, \dots, \mathcal{X}_M]$$

$$\mathcal{Y} = [\mathcal{Y}_1, \dots, \mathcal{Y}_M]$$

$$\vec{1} = [1, \dots, 1]$$

$$\alpha = [\alpha_1, \dots, \alpha_M]$$

و

$$\Omega_{ij} = K(\mathcal{X}_i, \mathcal{X}_j) = \varphi(\mathcal{X}_i)^T \varphi(\mathcal{X}_j), \quad i, j = 1, \dots, N. \quad (16.3)$$

۴.۳ کنترل بهینه با ماشین های بردار پشتیبان

فرض کنید داده های آموزشی متناظر با $\{\mathcal{X}_i, \mathcal{Y}_i\}_{i=1}^M$ به صورت $\{x_k, u_k\}_{k=1}^N$ در نظر گرفته شوند.

مسأله کنترل بهینه N -مرحله ای را به صورت زیر در نظر می گیریم:

$$\min \mathcal{J}_N(x_k, u_k, w, e_k) = \mathcal{J}_N(x_k, u_k) + \frac{1}{\gamma} w^T w + \frac{1}{\gamma} \sum_{k=1}^N e_k^2 \quad (17.3)$$

$$x_{k+1} = f(x_k, u_k), \quad k = 1, \dots, N \quad (x_1 \text{ معلوم})$$

و قانون کنترل

$$u_k = w^T \varphi(x_k) + e_k, \quad k = 1, \dots, N \quad (18.3)$$

که در آن $\mathcal{J}_N$ معادله‌ی (۱.۳) تعریف می‌شود، $\varphi(\cdot) : \mathbb{R}^n \rightarrow \mathbb{R}^{n_h}$ با n_h تعداد لایه‌های پنهان (که می‌تواند ابعاد بی‌نهایت داشته باشد)، از $\varphi(\cdot)$ تعریف می‌شود. سیگنال کنترل حقیقی که برای دستگاه به کار می‌رود $w^T \varphi(x_k)$ است. در حالت کنترل خطی به صورت، $\varphi(x_k) = x_k$ با $n_h = n$ داریم. روش بردار پشتیبان همان‌طور که در ابتدا معرفی شده‌است، از شرایط مرسر^{۲۱} بهره می‌گیرد، به این معنی که لازم نیست که $\varphi(\cdot)$ را بسازیم. برای حالت RBF معادله‌ی (۱۸.۳) به صورت زیر است:

$$u_k = \sum_{i=1}^{n_h} w_i \exp\left(-\frac{1}{\sigma^2} \|x_k - c_i\|^2\right) + e_k, \quad (19.3)$$

که $c_i \in \mathbb{R}^n$ مراکز انتخاب شده و σ یک ثابت انتخاب شده‌است. برای مثال می‌توان مجموعه $\{c_i\}_{i=1}^{n_h}$ ، را برابر $\{x_k\}_{k=1}^N$ انتخاب کرد. برای پیدا کردن قانون کنترل مطلوب تابع لاگرانژ زیر را در نظر می‌گیریم:

$$\begin{aligned} \mathcal{L}(x_k, u_k, e_k, w, \lambda_k, \alpha_k) = & \mathcal{J}_N(x_k, u_k) + \frac{1}{\gamma} w^T w + \gamma \frac{1}{\gamma} \sum_{k=1}^N e_k^2 \\ & + \sum_{k=1}^N \lambda_k^T [x_{k+1} - f(x_k, u_k)] + \sum_{k=1}^N \alpha_k [u_k - w^T \varphi(x_k) - e_k]. \end{aligned} \quad (20.3)$$

شرایط بهینگی به صورت زیر داده می‌شود:

$$\left\{ \begin{aligned} \frac{\partial \mathcal{L}}{\partial x_k} = \frac{\partial h}{\partial x_k} + \lambda_{k-1} - \left(\frac{\partial f}{\partial x_k} \right)^T \lambda_k - \alpha_k \frac{\partial}{\partial x_k} [w^T \varphi(x_k)] &= 0, \quad k = 2, \dots, N \quad (\text{معادله الحاقی}) \\ \frac{\partial \mathcal{L}}{\partial x_{N+1}} = \frac{\partial \rho}{\partial x_{N+1}} + \lambda_N &= 0, \quad (\text{شرط الحاقی}) \\ \frac{\partial \mathcal{L}}{\partial u_k} = \frac{\partial h}{\partial u_k} - \lambda_k^T \frac{\partial f}{\partial u_k} + \alpha_k &= 0, \quad k = 1, \dots, N \quad (\text{شرط تغییراتی}) \\ \frac{\partial \mathcal{L}}{\partial w} = w - \sum_{k=1}^N \alpha_k \varphi(x_k) &= 0, \quad (\text{بردار پشتیبان}) \\ \frac{\partial \mathcal{L}}{\partial e_k} = \gamma e_k - \alpha_k &= 0, \quad k = 1, \dots, N \quad (\text{ارزش پشتیبان}) \\ \frac{\partial \mathcal{L}}{\partial \lambda_k} = x_{k+1} - f(x_k, u_k) &= 0, \quad k = 1, \dots, N \quad (\text{دستگاه دینامیکی}) \\ \frac{\partial \mathcal{L}}{\partial \alpha_k} = u_k - w^T \varphi(x_k) - e_k &= 0, \quad k = 1, \dots, N \quad (\text{کنترل SVM}) \end{aligned} \right. \quad (21.3)$$

^{۲۱} Mercer's condition

برای هسته‌های RBF داریم:

$$\frac{\partial}{\partial x_k} [w^T \varphi(x_k)] = -2 \sum_i w_i \exp\left(-\frac{1}{\sigma^2} \|x_k - c_i\|_2^2\right) \times \frac{(x_k - c_i)}{\sigma^2}. \quad (22.3)$$

مجموعه‌ای از معادلات غیرخطی (۲۱.۳) به صورت زیر است:

$$F_1(x_k, x_{N+1}, u_k, w, e_k, \lambda_k, \alpha_k) = 0, \quad (23.3)$$

برای $k = 1, \dots, N$ که در آن x_1 داده شده و معادله (۲۱.۳) می‌تواند به صورت عددی برحسب متغیرهای مجهول حل شود.

$$w = \sum_k \alpha_k \varphi(x_k),$$

در نظریه SVM، شرایط مرسر می‌تواند در معادله (۲۱.۳) استفاده شود. برای هسته‌هایی که شرایط مرسر را برآورده می‌کنند می‌توان شرط زیر را به معادله (۲۳.۳) اعمال کرد.

$$K(x_k, x_l) = \varphi(x_k)^T \varphi(x_l). \quad (24.3)$$

برای هسته‌های RBF داریم:

$$K(x_k, x_l) = \exp(-\eta \|x_k - x_l\|_2^2), \quad (25.3)$$

که η ثابت مثبت است. واضح است که در معادله (۲۵.۳) مراکز c_i معادله (۱۹.۳) وجود ندارد. پس از حذف w ، مجموعه‌ای از معادلات غیرخطی به صورت زیر به دست می‌آید:

$$F_2(x_k, x_{N+1}, u_k, \lambda_k, \alpha_k) = 0, \quad (26.3)$$

برای $k = 1, \dots, N$ که در آن x_1 داده شده است. به طور خاص، پس از اعمال شرط مرسر، داریم:

$$\begin{cases} \frac{\partial h}{\partial x_k} + \lambda_{k-1} - \left(\frac{\partial f}{\partial x_k}\right)^T \lambda_k - \alpha_k \sum_{l=1}^N \alpha_l \frac{\partial K(x_k, x_l)}{\partial x_k} = 0, & k = 2, \dots, N \\ \frac{\partial \rho}{\partial x_{N+1}} + \lambda_N = 0, \\ \frac{\partial h}{\partial u_k} - \lambda_k^T \frac{\partial f}{\partial u_k} + \alpha_k = 0, & k = 1, \dots, N \\ x_{k+1} - f(x_k, u_k) = 0, & k = 1, \dots, N \\ u_k - \frac{\sum_{l=1}^N \alpha_l K(x_l, x_k) - \alpha_k}{\gamma} = 0, & k = 1, \dots, N \end{cases} \quad (27.3)$$

برای هسته‌های RBF داریم:

$$\frac{\partial K(x_k, x_l)}{\partial x_k} = -2\eta(x_k - x_l) \exp(-\eta \|x_k - x_l\|_2^2). \quad (28.3)$$

بنابراین سیگنال کنترل به صورت زیر اعمال می‌شود:

$$u_k = \sum_{l=1}^N \alpha_l K(x_l, x_k), \quad (29.3)$$

که $\{\alpha_l\}_{l=1}^N$ و $\{x_l\}_{l=1}^N$ از حل معادله غیرخطی (۲۷.۳) حاصل شده‌است و x_k بردار حالت واقعی در زمان k است. داده‌های $\{x_l\}_{l=1}^N$ به‌عنوان داده‌های بردار پشتیبان برای سیگنال کنترل استفاده می‌شود. علاوه براین، توجه داشته باشید که e_k در معادله‌ی (۲۹.۳) برابر با صفر است.

۵.۳ نمونه‌های شبیه‌سازی

۱.۵.۳ مثال ۱:

در این جا روش کنترل بهینه LS-SVM بخش ۴ را بر روی مثال ارائه شده در نارندا ۲۲ به کار می‌بریم. سیستم غیرخطی زیر را در نظر می‌گیریم:

$$\begin{cases} x_{1,k+1} = \phi_1 x_{1,k} + \gamma \frac{u_k + x_{2,k}}{\gamma(u_k + x_{2,k})^2} \\ x_{2,k+1} = \phi_2 x_{2,k} + u_k \left(\gamma + \frac{u_k}{\gamma + x_{1,k} + x_{2,k}} \right) \end{cases} \quad (30.3)$$

ما یک مسأله ردیابی ۲۳ را با

$$h(x_k, u_k) = (x_k - x_k^r)^T Q (x_k - x_k^r) + u_k^T R u_k,$$

و

$$\rho(x_{N+1}) = (x_{N+1} - x_{N+1}^r)^T Q (x_{N+1} - x_{N+1}^r),$$

که x_k^r مسیر مرجع ردیابی است، در نظر می‌گیریم. هدف در ردیابی متغیر حالت اول و انتخاب وضعیت $R = 1$ و $Q = \{1, 0, 0, 1\}$ برای $x_k^r = [\sin(\frac{\gamma \pi k}{\gamma_0}); \cos(\frac{\gamma \pi k}{\gamma_0})]$ با $k = 1, N$ و $N = 20$ است. وضعیت اولیه $x_1 = [0; 0]$ داده شده‌است. به‌عنوان قانون کنترل $w^T \varphi([x_k; x_k^r])$ در نظر گرفته شده‌است. در شکل (۱.۳) نتایج شبیه‌سازی برای مدل (۲۷.۳) نشان داده شده‌است که از شرایط مرسر استفاده می‌کند. مجموعه معادلات غیرخطی با استفاده از جعبه ابزار بهینه‌سازی Matlab با مجهولات $x_k, u_k, \lambda_k, \alpha_k, \sigma$ (متغیر e_k حذف شده) برای $\gamma = 100$ حل می‌شود.

شکل (۱.۳) نتایج شبیه‌سازی را برای سیستم حلقه بسته $\hat{x}_{k+1} = f(\hat{x}_k, \sum_{l=1}^N \alpha_l K(x_l, \hat{x}_k))$ با هسته RBF نشان می‌دهد. کنترل کننده با توجه به شرایط اولیه به غیر از مبدأ و با افق زمانی $N = 20$ داده می‌شود. مدل (۲۲.۳) بدون استفاده از شرایط مرسر نتایج مشابهی را با در نظر گرفتن مراکز $\{c_i\}$ برابر با $\{x_i\}$ ارائه می‌دهد.

^{۲۲}Narendra

^{۲۳}Tracking

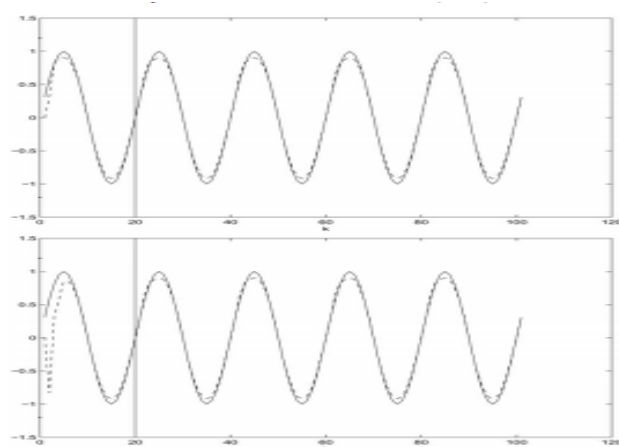
۶.۳ نتیجه‌گیری

در این پایان‌نامه، یک روش جدید مبتنی بر LS-SVM برای حل معادلات دیفرانسیل خطی از مرتبه m و هم‌چنین معادلات دیفرانسیل غیرخطی مرتبه اول توسعه داده شد. در مسائل حل شده، روش ارائه شده در این پایان‌نامه نسبت به روش‌های توصیف شده در [۲۵، ۲۹] کارایی بیشتری دارد. هم‌چنین روش پیشنهادی قادر به حل معادلات دیفرانسیل برای یک بازه طولانی برای پیش‌بینی جواب با دقت بالا است. روش پیشنهادی را می‌توان برای حل یک سیستم از معادلات دیفرانسیل معمولی و معادلات دیفرانسیل جزئی توسعه داد. در این پایان‌نامه استفاده از ماشین‌های بردار پشتیبان را برای حل مسائل کنترل بهینه معرفی کردیم. مسأله کنترل مطلوب N - مرحله‌ای برای یک کنترل‌کننده تابع حالت غیرخطی متشکل از ماشین‌های بردار پشتیبان فرمول‌بندی شده است. بنابراین، کمترین مربعات در نظر گرفته شده‌است، که منجر به محدودیت‌های مساوی به جای محدودیت‌های نامساوی می‌شود که به‌عنوان مثال از تابع زیان ε - حساسیت و پنیکی ناشی می‌شود.

روش کمترین مربعات برای داده‌های پراکنده مناسب نیست، اما دیگر ویژگی‌های SVM هنوز هم قابل اجرا هستند، مانند شرایط مرسر و این واقعیت که هیچ مرکزی برای هسته RBF تعیین نمی‌شود. گرچه به منظور مناسب بودن مسأله کنترل بهینه یک قید مساوی بر مبنای ماشین‌های بردار پشتیبان، انگیزه‌ای برای استفاده از SVM است.

تابع هدف برای مسأله کنترل و برای SVM کمترین مربعات به یک شکل طراحی شده است. سپس جواب با مجموعه‌ای از معادلات غیرخطی مشخص می‌شود. یک نقص این رویکرد این است که مسأله شامل بسیاری از مجهولات، حتی در محدودیت مساوی LS-SVM است. بنابراین یک فرمول جایگزین کنترل بهینه LS-SVM با تعداد مجهولات کمتر داده شده‌است. همگرایی محلی در کنترل مورد بحث قرار گرفته‌است. یک تفاوت اصلی کنترل LS-SVM با دیگر کنترل‌کننده‌های عصبی و به‌طور خاص RBF این است که در مورد اول دنباله بردار حالت در نظر گرفته شده از سیگنال کنترل پشتیبان می‌کند و به‌عنوان جواب برای مسأله بهینه‌سازی غیرخطی مقید استفاده می‌شود، درحالی‌که برای دومی اغلب مسأله بهینه‌سازی پارامتری را برای وزن‌های اتصال مجهول حل می‌کند.

تعداد لایه‌های پنهان LS-SVM با هسته RBF در قانون کنترل برابر با تعداد نقاط مورد نظر در چرخه زمان گسسته است. درحالی‌که، در فضای اولیه تعداد پارامترها می‌تواند بی‌نهایت باشد.



شکل ۱.۳: کنترل بهینه با ماشین‌های بردار پشتیبان کمترین مربعات با هسته RBF و استفاده از شرایط مرسر را نشان می‌دهد. متغیر حالت اول برای ردیابی [خط بریده بریده] متغیر حالت واقعی توسط شبیه‌سازی سیستم حلقه بسته است. داده‌های $k = 1, \dots, 20$ برای کنترل با مبدأ به‌عنوان حالت اولیه مورد استفاده قرار گرفتند.

مراجع

- [۱] محمد جواد ساده؛ رده بندی در داده های ریز آرایه ها؛ پایان نامه کارشناسی ارشد، مشهد، بهمن ۱۳۸۴.
- [2] A. E. Bryson. and Y. C. Ho, Applied optimal control, Waltham, MA: Blaisdel, 1969
- [3] A. J. Meade, Jr. and A. A. Fernadez.(1994), The numerical solution of Linear Ordinary Differential Equations by Feedforward Neural networks, Mathematical and Computer Modelling, 19, 1-25.
- [4] A. Smola, Learning with Kernels. PhD thesis, published by Birlinghoven: GMD, 1999.
- [5] A. Smola.and B. Scholkopf. (1998), On a kernel-based method for pattern recognition, regression, approximation and operator inversion. Algorithmica, 22, 211-231.
- [6] A. Smola, B. Scholkopf, K. -R. Muller. (1998), The connection between regularization operators and support vector kernels. Neural Networks, 11 (4), 637-649.
- [7] A. Toselli. and O. B. Widlund, Domain Decomposition Methods- Algorithms and Theory. Springer-Verlag, Berlin and Heidelberg, 2005.
- [8] B. Ph. van Milligen, V. Tribaldos, and J. A. Jimenez. (1995), Neural Network Differential Equation and Plasma Equilibrium Solver, Physical Review Letters, 75, 3594-3597.
- [9] B. Scholkopf, A. Smola, Learning with Kernels, MIT Press, Cambridge, MA, 2002.
- [10] B. Scholkopf, C. J. C. Burges, A. J. Smola, Advances in kernel methods D Support vector learning, Cambridge, MA: MIT Press, 1999.
- [11] B. Scholkopf, K. -K. Sung, C. Burges, F. Girosi, P. Niyogi, T. Poggio, V. Vapnik. (1997), Comparing support vector machines with Gaussian kernels to radial basis function classifiers. IEEE Transactions on Signal Processing, 45 (11), 2758-2765.

- [12] C. M. Bisho, Neural networks for pattern recognition. Oxford: Oxford University Press, 1995.
- [13] C. Saunders, A. Gammerman, V. Vovk, (1998), Ridge regression learning algorithm in dual variables. Proceedings of the 15th International Conference on Machine Learning ICML-98, Madison, WI, 515-521.
- [14] C. W. Gear. (1965), Hybrid methods for initial value problems in ordinary differential equations, SIAM Journal on Numerical Analysis, 2, 69-86.
- [15] David M. J. TAX, One-class classification, PHD thesis, ter verkrijging van de graad van doctor aan de Technische Universiteit Delf, 2001.
- [16] D. R. Kincaid. and E. W. Cheney, Numerical Analysis: Mathematics of Scientific Computing, third ed., Brooks/Cole, Pacific Grove, CA. 2002.
- [17] D. O. Awoyemi. (2001), A new sixth-order algorithm for general second order ordinary differential equation, International Journal of Computer Mathematics, 71, 117-124.
- [18] D. Sarafyan. (1990), New algorithm for the continuous approximate solution of ordinary differential equations and the upgrading of the order of the processes, Computers & Mathematics with Applications 20(1), 71-100.
- [19] E. S. Plumer. (1996), Optimal control of terminal processes using neural networks. IEEE Transactions on Neural Networks, 7 (2), 408-418.
- [20] G. F. Franklin, J. D. Powell, M. L. Workman, Digital control of dynamic systems, Reading, MA: Addison-Wesley, 1990.
- [21] G. H. Golub. and C. F. Van Loan, Matrix computations, Baltimore, MD: Johns Hopkins University Press, 1989.
- [22] G. Walsh, Methods of Optimization, John Wiley and Sons, N. Y. 1987.
- [23] H. Drucker, C. Burges, L. Kaufman, A. Smola and V. Vapnik. (1996), Support Vector Regression Machines, Advances in Neural Information Processing Systems 9, 155-161, MIT Press.
- [24] H. Lee. and I. Kang. (1990), Neural algorithms for solving differential equations, Journal of Computational Physics, 91, 110-117.

- [25] H. S. Yazdi, M. Pakdaman, H. Modaghegh. (2011), Unsupervised kernel least mean square algorithm for solving ordinary differential equations, *Neurocomputing*, 74, 2062-2071.
- [26] H. Verrelst, K. Van Acker, J. Suykens, B. Motmans, B. De Moor, J. Vandewalle. (1998), Application of NL_q neural control theory to a ball and beam system. *European Journal of Control*, 4, 148-157.
- [27] I. G. Tsoulos, I. E. Lagaris. (2006), Solving differential equations with genetic programming, *Genetic Programming and Evolvable Machines*, 7, 33- 54.
- [28] I. G. Tsoulos, D. Gavrilis, E. Glavas. (2009), Solving differential equations with constructed neural networks, *Neurocomputing*, 72, 2385-2391.
- [29] I. Lagaris, A. Likas, and D. I. Fotiadis. (1998), Artificial neural networks for solving ordinary and partial differential equations, *IEEE Transactions on Neural Networks*, 9, 987-1000.
- [30] J. A. K. Suykens, B. De Moor, J. Vandewalle. (1994), Static and dynamic stabilizing neural controllers, applicable to transition between equilibrium points. *Neural Networks*, 7 (5), 819-831.
- [31] J. A. K. Suykens, B. De Moor, J. Vandewalle. (1997), NL_q theory: a neural control framework with global asymptotic stability criteria. *Neural Networks*, 10 (4), 615-637
- [32] J. A. K. Suykens, J. Vandewalle. (1999), Least squares support vector machine classifiers, *Neural Processing Letters*, 9 (3), 293-300.
- [33] J. A. K. Suykens, T. Van Gestel, J. De Brabanter, B. De Moor, J. Vandewalle *Least Squares Support Vector Machines*, World Scientific Pub. Co., Singapore, 2002.
- [34] J. A. K. Suykens, L. Lukas, J. Vandewalle. (2000), Sparse approximation using least squares support vector machines. *IEEE International Symposium on Circuits and Systems ISCAS 2000*, Geneva, Switzerland, 31, 757-760.
- [35] J. A. K. Suykens, L. Lukas, J. Vandewalle. (2000), Sparse Least Squares Support Vector Machine Classifiers. *8th European Symposium on Artificial Neural Networks ESANN 2000*, Bruges Belgium, 37-42.
- [36] J. A. K. Suykens, J. Vandewalle, B. De Moor, *Artificial neural networks for modelling and control of non-linear systems*, Boston: Kluwer Academic, 1996.

-
- [37] J. C. Butcher, Numerical Methods for Ordinary Differential Equations, 2nd Edition, John Wiley & Sons, Chichester, 2008.
- [38] J. D. Lambert, Numerical Methods for Ordinary Differential Systems, Wiley, New York, 1991.
- [39] J. Hauser, S. Sastry, P. Kokotovic. (1992), Nonlinear control via approximate input-output linearization: the ball and beam example. IEEE Transactions on Automatic Control, 37 (3), 392-398.
- [40] J. A. K. Suykens. and J. Vandewalle. (1999a), Training multilayer perceptron classifiers based on a modified support vector method. IEEE Transactions on Neural Networks, 10 (4), 907-912. bibitem27 J. A. K. Suykens, J. Vandewalle, B. De Moor. (2001), Optimal control by least squares support vector machines, Neural Networks, 14, 23-35.
- [41] J. A. K. Suykens. and J. Vandewalle. (1999), Least squares support vector machine classifiers. Neural Processing Letters, 9 (3), 293-300.
- [42] K. R. Muller, A. J. Smola, G. Ratsch, B. Scholkopf, J. Kohlmorgen, V. Vapnik. (1997), Predicting time series with support vector machines. In W. Gerstner, A. Germond, M. Hasler, J. -D. Nicoud, Proceedings of the International Conference on Artificial Neural Networks ICANN'97 LNCS 1327 (999-1004). Berlin: Springer.
- [43] K. S. McFall. and J. R. Mahan. (2009), Artificial Neural Network Method for Solution of Boundary Value Problems With Exact Satisfaction of Arbitrary Boundary Conditions, IEEE Transactions on Neural Networks, 20, 1221-1233.
- [44] K. S. Narendra. and K. Parthasarathy. (1991), Gradient methods for the optimization of dynamical systems containing neural networks. IEEE Transactions on Neural Networks, 2 (2), 252-262.
- [45] K. S. Narendra. and S. Mukhopadhyay. (1997), Adaptive control using neural networks and approximate models. IEEE Transactions on Neural Networks, 8 (3), 475-485.
- [46] K. Vapnik, Statistical learning theory, New York: Wiley, 1998a.
- [47] L. Ljung, System Identification: Theory for the User, New York: Prentice-Hall, 1987.
- [48] M. E. Sezer. and D. D. Siljak. (1988), Robust stability of discrete systems. International Journal of Control, 48 (5), 2055-2063.

- [49] M. Lazaro, I. Santamaria, F. Perez-Cruz, A. Artes-Rodriguez. (2005), Support vector regression for the simultaneous learning of a multivariate function and its derivative,"Neurocomputing, 69 (1-3), 42-61.
- [50] M. M. Chawla, C. P. Katti. (1979), Finite difference methods for two-point boundary value problems involving high order differential equations, BIT Numerical Mathematics, 19, 27-33.
- [51] M. S. Bazaraa, H. D. Sherali and C. M. Shetty, Nonlinear Programming: Theory and Algorithms, second ed. New York, John Wiley and sons, 1993.
- [52] M. Saerens, J. -M. Renders, H. Bersini, Neural controllers based on backpropagation algorithm. In M. M. Gupta and N. K. Sinha, IEEE Press book on intelligent control: theory and practice New York: IEEE Press, 1993.
- [53] P. E. Gill, w. Murray, M. H. Wright, Practical optimization, London: Academic Press, 1981.
- [54] P. Ramuhalli, L. Udpa, S. S. Udpa. (2005), Finite-element neural networks for solving differential equations, IEEE Transactions on Neural Networks, 16, 1381-1392.
- [55] P. Werbos. (1990), Backpropagation through time: what it does and how to do it. Proceedings of the IEEE, 78 (10), 1150-1560.
- [56] R. D. Russell, L. F. Shampine. (1972), A collocation method for boundary value problems, Numer. Math, 19, 1-28.
- [57] R. Fletcher, Practical Methods of Optimization, John Wiley and Sons, N. Y. 1975.
- [58] R. Fletcher, Practical methods of optimization, New York: Wiley, 1987.
- [59] R. M. Sanner, J. -J. E. Slotine. (1992), Gaussian networks for direct adaptive control. IEEE Transactions on Neural Networks, 3 (6), 837- 863.
- [60] S. Boyd. and C. Barratt, Linear controller design, limits of performance. Englewood Cliffs, NJ: Prentice-Hall, 1991.
- [61] S. Haykin, Neural Networks: a Comprehensive Foundation, (2nd ed.). Englewood Cliffs: Macmillan, 1994.
- [62] S. N. Jator. and J. Li. (2009), A self-starting linear multistep method for a direct solution of the general second-order initial value problem, International Journal of Computer Mathematics, 86(5), 827-836.

-
- [63] T. Parisini. and R. Zoppoli. (1994), Neural networks for feedback feedforward nonlinear control systems. IEEE Transactions on Neural Networks, 5 (3), 436-449.
 - [64] T. Poggio. and F. Girosi. (1990), Networks for approximation and learning. Proceedings of the IEEE, 78 (9), 1481-1497.
 - [65] V. Cherkassky. and f. Mulier, Learning from data: concepts, theory and methods, New York: Wiley, 1998.
 - [66] V. Vapnik, Statistical learning theory, New York: Wiley, 1998.
 - [67] V. Vapnik, The nature of statistical learning theory, New York: Springer, 1995.
 - [68] V. Vapnik. (1998b), The support vector method of function estimation. In J. A. K. Suykens. and J. Vandewalle, Nonlinear modeling: advanced blackbox techniques, 55-85. Boston: Kluwer Academic.
 - [69] V. Vapnik, S. Golowich, A. Smola. (1997), Support vector method for function approximation, regression estimation and signal processing. Advances in Neural Information Processing Systems, (9). Cambridge, MA: MIT Press.

Abstract

In this thesis a new approach based on Least Squares Support Vector Machines (LS-SVMs) is proposed for solving linear and nonlinear ordinary differential equations (ODEs). The approximate solution is presented in closed form by means of LSSVMs, whose parameters are adjusted to minimize an appropriate error function. For the linear and nonlinear cases, these parameters are obtained by solving a system of linear and nonlinear equations respectively. The method is well suited for solving mildly stiff, nonstiff and singular ordinary differential equations with initial and boundary conditions. Numerical results demonstrate the efficiency of the proposed method over existing methods. Support vector machines have been very successful in pattern recognition and function estimation problems. In this paper we introduce the use of least squares support vector machines (LS-SVM's) for the optimal control of nonlinear systems. Linear and neural full static state feedback controllers are considered. The problem is formulated in such a way that it incorporates the N-stage optimal control problem as well as a least squares support vector machine approach for mapping the state space into the action space. The solution is characterized by a set of nonlinear equations. An alternative formulation as a constrained nonlinear optimization problem in less unknowns is given, together with a method for imposing local stability in the LS-SVM control scheme. The results are discussed for support vector machines with radial basis function kernel. Advantages of LS-SVM control are that no number of hidden units has to be determined for the controller and that no centers have to be specified for the Gaussian kernels when applying Mercer's condition. The curse of dimensionality is avoided in comparison with defining a regular grid for the centers in classical radial basis function networks. This is at the expense of taking the trajectory of state variables as additional unknowns in the optimization problem, while classical neural network approaches typically lead to parametric optimization problems. In the SVM methodology the number of unknowns equals the number of training data.

keywords: Least squares support vector machines, Ordinary differential equations, Closed form approximate solution, Collocation method, neural optimal control, Support vector machines, Radial basis functions.



Shahrood University of Technology

Faculty Of Mathematical Sciences

MSc Thesis in: Optimization

**An application of least square support vector
optimization methods for solving a class of
differential equations**

By: Fatemeh Barzamini

Supervisor

Dr. Alireza Nazemi

January 2019