

بسم الله الرحمن الرحيم



دانشگاه حکیم بسزوری

دانشکده ریاضی و علوم کامپیوتر

پایان نامه برای دریافت درجه کارشناسی ارشد در رشته ریاضی کاربردی  
گرایش تحقیق در عملیات

## روش جستجوی زیرفضا در بهینه سازی تنک

استاد راهنما

دکتر محمود امین طوسی

استاد مشاور

دکتر امین رفیعی

پژوهشگر:

محبوبه بخش علی

شهریور ۹۷



دانشگاه آزاد اسلامی

شماره:

باسمه تعالی

تاریخ:

صورتجلسه دفاع از پایان نامه کارشناسی ارشد

با تلاوت آیاتی چند از کلام الله مجید جلسه دفاع از پایان نامه آقای / خانم محبوبه بخش علی دانشجوی رشته ریاضی کاربردی به شماره دانشجویی ۹۴۱۳۱۳۳۰۹۳ با عنوان:

### روش جستجوی زیرفضا در بهینه سازی تنک

در ساعت مورخه در محل دانشکده ریاضی و علوم کامپیوتر تشکیل گردید . پس از استماع گزارش ارائه شده توسط دانشجو و استاد راهنما، هیات داوران و حاضران سئوالاتی را مطرح و آقای / خانم محبوبه بخش علی به دفاع از موضوع پرداخت و به سئوالات آنها پاسخ گفت. سپس پایان نامه توسط هیات داوران مورد ارزشیابی قرار گرفت و نمره برابر درجه برای آن تعیین گردید .  
به این ترتیب ضمن تصویب پایان نامه مزبور از این تاریخ آقای / خانم محبوبه بخش علی به عنوان کارشناس ارشد در رشته ریاضی کاربردی شناخته می شود .

ردیف	نام و نام خانوادگی	سمت	امضاء
۱	دکتر محمود امین طوسی	استاد راهنما	
۲	دکتر امین رفیعی	استاد مشاور	
۳	دکتر مهدی زعفرانیه	استاد داور	
۴	دکتر اعظم پورمیرزایی	نماینده تحصیلات تکمیلی	

مدیر گروه: دکتر محمدعلی پرتانیان

رونوشت:

۱. معاون محترم آموزشی و تحصیلات تکمیلی دانشگاه جهت اطلاع
۲. معاون محترم پژوهشی دانشگاه جهت اطلاع
۳. آموزش دانشکده ریاضی و علوم کامپیوتر جهت درج در پرونده دانشجو



## سوگند نامه دانش آموختگان دانشگاه حکیم سبزواری

به نام خداوند جان و خرد      کزین برتر اندیشه بر نگذرد

اینک که به خواست آفریدگار پاک، کوشش خویش و بهره گیری از دانش استادان و سرمایه‌های مادی و معنوی این مرز و بوم، توشه‌ای از دانش و خرد گردآورده‌ام، در پیشگاه خداوند بزرگ سوگند یاد می‌کنم که در به کارگیری دانش خویش، همواره بر راه راست و درست گام بردارم. خداوند بزرگ، شما شاهدان، دانشجویان و دیگر حاضران را به عنوان داورانی امین گواه می‌گیرم که از همه دانش و توان خود برای گسترش مرزهای دانش بهره‌گیرم و از هیچ کوششی برای تبدیل جهان به جایی بهتر برای زیستن، دریغ نورزم. پیمان می‌بندم که همواره کرامت انسانی را در نظر داشته باشم و هموعان خود را در هر زمان و مکان تا سر حد امکان یاری دهم. سوگند می‌خورم که در به کارگیری دانش خویش به کاری که باراه و رسم انسانی، آیین پرهیزگاری، شرافت و اصول اخلاقی برخاسته از ادیان بزرگ الهی، به ویژه دین مبین اسلام، مابینت دارد دست نیازم. همچنین در سایه اصول جهان شمول انسانی و اسلامی، پیمان می‌بندم از هیچ کوششی برای آبادانی و سرافرازی میهن و هم میهنانم فروگذاری نکنم و خداوند بزرگ را به یاری طلبم تا همواره در پیشگاه او و در برابر وجدان بیدار خویش و ملت سرافراز، بر این پیمان تا ابد استوار بمانم.

نام و نام خانوادگی:      محبوبه بخش علی

تاریخ و امضا:

## تأییدی صحت و اصالت نتایج

باسمه تعالی

اینجانب محبوبه بخش علی به شماره دانشجویی ۹۴۱۳۱۳۳۰۹۳ دانشجوی رشته ریاضی کاربردی مقطع تحصیلی کارشناسی ارشد تأیید می‌نمایم که کلیه نتایج این پایان‌نامه حاصل کار اینجانب و بدون هرگونه دخل و تصرف است و موارد نسخه برداری شده از آثار دیگران را با ذکر کامل مشخصات منبع ذکر کرده‌ام. در صورت اثبات خلاف مندرجات فوق، به تشخیص دانشگاه مطابق با ضوابط و مقررات حاکم (قانون حمایت از حقوق مؤلفان و مصنفان و قانون ترجمه و تکثیر کتب و نشریات و آثار صوتی، ضوابط و مقررات آموزشی، پژوهشی و انضباطی ...) با اینجانب رفتار خواهد شد و حق هرگونه اعتراض در خصوص احقاق حقوق مکتسب و تشخیص و تعیین تخلف و مجازات را از خویش سلب می‌نمایم. در ضمن، مسئولیت هرگونه پاسخگویی به اشخاص اعم از حقیقی و حقوقی و مراجع ذی صلاح (اعم از اداری و قضایی) به عهده ی اینجانب خواهد بود و دانشگاه هیچ‌گونه مسئولیتی در این خصوص نخواهد داشت.

نام و نام خانوادگی: محبوبه بخش علی

تاریخ و امضا:

## مجوز بهره برداری از پایان نامه

بهره برداری از این پایان نامه در چهارچوب مقررات کتابخانه و با توجه به محدودیتی که توسط استاد راهنما

به شرح زیر تعیین می شود، بلامانع است:

بهره برداری از این پایان نامه برای همگان بلامانع است.

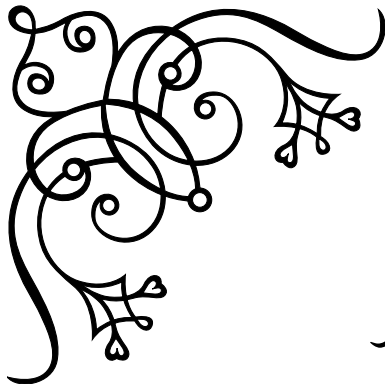
بهره برداری از این پایان نامه با اخذ مجوز از استاد راهنما، بلامانع است.

بهره برداری از این پایان نامه تا تاریخ ..... ممنوع است.

استاد راهنما: دکتر محمود امین طوسی

تاریخ و امضا:

تقدیم به:



پدر و مادر عزیزم و

همسر فداکارم



سپاس خداوندگار حکیم را که با لطف بی کران خود، آدمی را زیور عقل آراست. در آغاز وظیفه خود می دانم از زحمات بی دریغ استاد راهنمای خود، جناب آقای دکتر محمود امین طوسی، صمیمانه تشکر و قدردانی کنم که قطعاً بدون راهنمایی های ارزنده ایشان، این مجموعه به انجام نمی رسید. از جناب آقای دکتر رفیعی و آقای دکتر زعفرانی که زحمت مطالعه و مشاوره این رساله را تقبل فرمودند و در آماده سازی این رساله، به نحو احسن اینجانب را مورد راهنمایی قرار دادند، کمال امتنان را دارم. همچنین لازم می دانم از گروه پارسی لانتک در پاسخگویی به مشکلات کاربران کمال قدردانی را داشته باشم. در پایان، بوسه می زنم بر دستان خداوندگاران مهر و مهربانی، پدر و مادر عزیزم و بعد از خدا، ستایش می کنم وجود مقدس شان را و تشکر می کنم از خانواده عزیزم به پاس عاطفه سرشار و گرمای امیدبخش وجودشان، که بهترین پشتیبان من بودند.

محبوبه بخش علی

شهریور ۹۷

# فهرست مطالب

ج	فهرست جداول
د	فهرست تصاویر
۱	چکیده
۲	پیش‌گفتار
۴	فصل ۱: مقدمه
۴	۱-۱ بیان مسئله و ضرورت انجام پروژه
۶	۲-۱ راهکارهای پیشنهادی در حل مسئله
۹	فصل ۲: روش‌های تعیین نمایش تنک سیگنال
۹	۱-۲ معرفی برخی اصطلاحات و قراردادهای
۱۰	۲-۲ تعیین نمایش تنک سیگنال
۱۲	۳-۲ یکتایی تنک‌ترین جواب
۱۴	۴-۲ پایداری تقریب تنک سیگنال
۱۶	۵-۲ روش‌های موجود
۱۷	۱-۵-۲ کمینه‌سازی نرم $l_0$
۱۹	۲-۵-۲ کمینه‌سازی نرم $l_1$
۲۲	۳-۵-۲ روش جستجوی تطابقی
۲۷	۴-۵-۲ روش جستجوی تطابقی متعامد
۲۸	۵-۵-۲ الگوریتم CoSaMP
۳۱	فصل ۳: روش جستجوی زیرفضا

۳۱	مقدمه	۱-۳
۳۲	تفسیر هندسی روش جست و جوی زیرفضا	۲-۳
۳۴	فرمول بندی در قالب یک مسئله بهینه سازی	۳-۳
۳۶	جواب مینیمم فاصله تا یک زیرفضا	۴-۳
۴۱	یافتن جواب LS دستگاه معادلات خطی فرامعین	۵-۳
۴۴	یافتن جواب دارای کمترین نرم $l_2$ دستگاه معادلات خطی فرامعین	۶-۳
۴۸	یافتن جواب مینیمم فاصله تا زیرفضا به روشی دیگر	۷-۳

#### فصل ۴: روش آشکارسازی-تخمین پیاپی

۵۱	مقدمه	۱-۴
۵۲	آزمون فرضیه (یا آشکارسازی) باینری	۲-۴
۵۴	دیدگاه های بیز و نیمن-پیرسون	۳-۴
۵۵	آزمون فرضیه مرکب و تست UMP	۴-۴
۵۶	آماره بسنده	۵-۴
۵۷	گام آشکارسازی	۶-۴
۶۱	روش دیگر از گام آشکارسازی	۱-۶-۴
۶۲	گام تقریب/تخمین	۷-۴
۶۲	نگاه در فضای $x$	۱-۷-۴
۶۳	نگاه در فضای $s$	۲-۷-۴
۶۴	جواب اولیه بردار منبع	۸-۴
۶۵	نکات اصلی انتخاب سطح آستانه	۹-۴
۶۵	کران های ضمنی	۱-۹-۴
۶۷	کران های صریح	۲-۹-۴

#### فهرست منابع

پیوست آ: ۷۷

واژه نامه فارسی به انگلیسی ۸۶

واژه نامه انگلیسی به فارسی ۸۷



# فهرست جداول

# فهرست تصاویر

- ۱-۲ نرم های مختلف . . . . . ۱۰
- ۲-۲ تقاطع بین توپ های  $l_p$  و سیستم  $x = As$  با استفاده از حل  $p_p$  در فضای سه بعدی . . . . . ۱۱
- ۳-۲ ایده ی اصلی روش MP: انتخاب اتمی ( $a_i$  ای) که نزدیک ترین فاصله (عمودی) را تا بردار  $x$  دارد که در اینجا بردار  $a_1$  است. در مرحله بعد این عمل برای بردار باقیمانده  $r^{(1)}$  تکرار می شود و به همین ترتیب . . . . . ۲۳
- ۱-۳ تعبیر هندسی روش جست و جوی زیرفضا در حالت  $m = 3$  و  $n = 2$ . (آروش SP جوابی از دستگاه  $x = As$  (که در اینجا یک خط است) را به عنوان تنگ ترین جواب انتخاب می کند که نزدیک ترین فاصله را تا توزیع  $s$  (در اینجا محورهای مختصات) را داشته باشد. (ب) همان شکل از نمایی دیگر است و همچنین جواب LP نیز با یک مثلث مشخص شده است. . . . . ۳۳
- ۱-۴ کران های صریح برای سطح آستانه (در ابعاد پایین): نمودارهای پراکندگی ماکزیمم اندازه منابع ( $\|s\|_\infty$ ) بر حسب هر یک از سه تخمین گر پیشنهادی برای  $N=5000$  تحقق تصادفی مسئله . . . . . ۶۹
- ۲-۴ کران های صریح برای سطح آستانه (در ابعاد بالا): نمودارهای پراکندگی ماکزیمم اندازه منابع ( $\|s\|_\infty$ ) بر حسب سه تخمین گر پیشنهادی در ابعاد  $m = 100, 200, 500, 1000$  . . . . . ۷۰
- ۳-۴ کران های صریح برای سطح آستانه (رفتار حدی): نسبت ماکزیمم اندازه منابع ( $\|s\|_\infty$ ) به تخمین آن، بر حسب  $n$  به عنوان معیاری از بعد مسئله، برای سه تخمین گر پیشنهادی . . . . . ۷۱



دانشگاه گیلان

## فرم چکیده ی پایان نامه ی دوره ی تحصیلات تکمیلی

مدیریت تحصیلات تکمیلی

نام خانوادگی دانشجو: بخش علی	نام: محبوبه	ش. دانشجویی: ۹۳۰۹۳۱۳۱۳۳۹۴
استاد راهنما: دکتر محمود امین طوسی		
استاد مشاور: دکتر امین رفیعی		
دانشکده ریاضی و علوم کامپیوتر	رشته: ریاضی کاربردی	گرایش: تحقیق در عملیات
مقطع: کارشناسی ارشد	تاریخ دفاع: شهریور ۹۷	تعداد صفحات: ۸۸
عنوان پایان نامه: روش جستجوی زیرفضا در بهینه سازی تنک		
کلید واژه ها: بهینه سازی تنک، جداسازی منابع تنک، تجزیه تنک، نمایش تنک سیگنال		
<p>چکیده: یافتن جواب های تنک دستگاه های معادلات خطی "فرومعین" یعنی دستگاههایی که تعداد معادلات آنها کمتر از تعداد مجهول هاست، از جمله مسائل اساسی در پردازش سیگنال است. در حالت کلی تعداد جوابهای این دستگاه ها بی شمار است ولی می توان با اضافه کردن شرط تنک بودن جواب، یکتایی جواب را تضمین کرد. از این ویژگی میتوان برای یافتن نمایش تنک سیگنال روی یک دیکشنری "فوق کامل" استفاده کرد.</p> <p>دو مسئله مهم در پردازش تنک وجود دارد. یکی از این مسائل آموزش دیکشنری یعنی یافتن دیکشنری فوق کامل مناسب برای یک کلاس مشخص از داده هاست؛ یا به عبارتی، دیکشنری ای که بتواند برای همه سیگنالهای آن کلاس، یک نمایش به اندازه کافی تنک ارائه دهد. مسئله دوم داشتن یک الگوریتم کارا برای به دست آوردن تنک ترین نمایش سیگنال (یا کدینگ تنک سیگنال) است.</p> <p>در سالهای اخیر استفاده از نمایش تنک سیگنالها در کاربردهایی مانند فشرده سازی، نویز زدایی و طبقه بندی مورد توجه قرار گرفته است. موفقیت نمایش تنک در این کاربردها به این دلیل است که اکثر سیگنال های طبیعی مانند تصویر یا گفتار، با در نظر گرفتن پایه های مشخص نمایشی تنک دارند. هدف ما در این پایان نامه معرفی خانواده ای از روش ها برای کدینگ تنک سیگنال ها، تحت عنوان "آشکارسازی-تخمین پیاپی (IDE)" است که اساس کار این روش آشکارسازی مولفه های فعال بردار جواب و سپس تخمین آنها به صورت مرحله به مرحله است که به سمت جواب نهایی همگرا می شوند. در گام تخمین می توان از روش های موجود برای تخمین بردار منبع از جمله روش جست و جوی زیرفضا بهره جست.</p>		

## پیش‌گفتار

یافتن جواب‌های تنک دستگاه‌های معادلات خطی "فرومعیّن" یعنی دستگاه‌هایی که تعداد معادلات آنها کمتر از تعداد مجهول‌هاست، از جمله مسائل اساسی در پردازش سیگنال است. در حالت کلی تعداد جواب‌های این دستگاه‌ها بی‌شمار است ولی می‌توان با اضافه کردن شرط تنک بودن جواب، یکتایی جواب را تضمین کرد. از این ویژگی می‌توان برای یافتن نمایش تنک سیگنال روی یک دیکشنری "فوق‌کامل" استفاده کرد.

دو مسئله مهم در پردازش تنک وجود دارد. یکی از این مسائل آموزش دیکشنری یعنی یافتن دیکشنری فوق‌کامل مناسب برای یک کلاس مشخص از داده‌هاست؛ یا به عبارتی، دیکشنری‌ای که بتواند برای همه سیگنال‌های آن کلاس، یک نمایش به اندازه کافی تنک ارائه دهد. مسئله دوم داشتن یک الگوریتم کارا برای به دست آوردن تنک‌ترین نمایش سیگنال (یا کدینگ تنک سیگنال) است.

در سال‌های اخیر استفاده از نمایش تنک سیگنال‌ها در کاربردهایی مانند فشرده‌سازی، نويز زدایی و طبقه‌بندی مورد توجه قرار گرفته است. موفقیت نمایش تنک در این کاربردها به این دلیل است که اکثر سیگنال‌های طبیعی مانند تصویر یا گفتار، با در نظر گرفتن پایه‌های مشخص نمایشی تنک دارند. هدف ما در این پایان‌نامه معرفی خانواده‌ای از روش‌ها برای کدینگ تنک سیگنال‌ها، تحت عنوان "آشکارسازی-تخمین پیاپی (IDE)" است که اساس کار این روش آشکارسازی مولفه‌های فعال بردار جواب و سپس تخمین آنها به صورت مرحله به مرحله است که به سمت جواب نهایی همگرا می‌شوند.

این پایان‌نامه در چهار فصل تنظیم شده است که در آن سعی می‌شود ضمن بیان پشتوانه‌های تئوریک نمایش تنک و الگوریتم‌های مطرح شده، نحوه عملکرد الگوریتم‌ها به طور مبسوط ارائه و شرح داده شود.

در فصل اول به عنوان مقدمه، به بیان مسئله و ضرورت انجام پروژه پرداخته و راهکارهای موجود برای حل مسئله به طور خلاصه مطرح می‌شوند.

در فصل دوم، پیش از ورود به بخش تئوریک مسئله، برخی اصطلاحات و قراردادهای مورد استفاده در این پایان‌نامه معرفی می‌شود. پس از آن، مباحث پایه‌ای بحث نمایش تنک سیگنال را به همراه تعدادی از الگوریتم‌هایی که تاکنون برای بازیابی نمایش تنک سیگنال ارائه شده است، مرور می‌کنیم. در این فصل علاوه بر روش

”مینیمم سازی نرم  $l_1$ “ دو روش تجزیه اتمی ”جستجوی تطابق (MP)“ و ”جستجوی تطابق متعامد (OMP)“ را بررسی می کنیم.

در فصل سوم با استفاده از یک تعبیر هندسی از تنک بودن ، روش ”جستجوی زیرفضا (SP)“ شرح داده می شود و در فصل چهارم بر اساس ایده های تئوری آشکارسازی ، خانواده الگوریتم های ”آشکارسازی-تخمین پیاپی (IDE)“ مطرح می شوند.

مراجع اصلی این پایان نامه عبارتند از:

- A. A. Amini, M. Babie-Zadeh, and C. Jutten, “A new approach for sparse decomposition and sparse source separation,” accepted for EUSIPCO’06, 2006.
- A. A. Amini, M. Babie-Zadeh, and C. Jutten, “A fast method for sparse component analysis based on iterative detection-projection,” accepted for MaxEnt’06, 2006.
- M. Elad, Sparse and Redundant Representations, Springer, 2010.
- S. S. Chen, D. L. Donoho, and M. A. Saunders, “Atomic decomposition by basis pursuit,” SIAM Journal on Scientific Computing, vol. 20, no. 1, pp. 33–61, 1999.
- S. Mallat and Z. Zhang, “Matching pursuits with time-frequency dictionaries,” IEEE Trans. on Signal Proc., vol. 41, no. 12, pp. 3397–3415, 1993.
- Y. C. Pati, R. Rezaifar, and P. S. Krishnaprasad, “Orthogonal matching pursuit: recursive function approximation with applications to wavelet decomposition,” in In Proc. Asilomar Conf. Signal Syst. Comput., 1993.

# فصل ۱

## مقدمه

### ۱-۱ بیان مسئله و ضرورت انجام پروژه

سیستم‌ها و تبدیلات خطی از جایگاه ویژه‌ای در شاخه‌های علوم و مهندسی برخوردارند. چنین تبدیلی‌هایی را (در فضاهای با بعد محدود) می‌توان با یک دستگاه معادلات خطی نمایش داد. دستگاه معادلات خطی

$$Ax = b \quad (1-1)$$

از منظرهای مختلف، به طور خاص از نظر "تعداد معلومات و مجهولات" قابل دسته‌بندی اند. فرض کنید  $A$  یک ماتریس  $n \times m$  و  $x$  و  $b$  به ترتیب  $m \times 1$  و  $n \times 1$  ( $A$  و  $b$  معلوم و  $x$  مجهول) باشند. در حالت اول، اگر  $n = m$  و  $A$  معکوس پذیر باشد،  $x = A^{-1}b$  جواب یکتای دستگاه  $Ax = b$  خواهد بود. در حالت دوم، اگر  $n > m$  باشد، دستگاه  $Ax = b$ ، که یک دستگاه "فرامعین" نامیده می‌شود، جواب مشخصی ندارد؛ اما با حل مسئله بهینه‌سازی زیر می‌توان یک تقریب LS<sup>۱</sup> (به مفهوم کمترین مربعات) از دستگاه را به دست آورد:

$$\arg \min_x \|Ax - b\|_2 \quad (2-1)$$

اما حالت مورد علاقه ما در این پایان‌نامه دستگاه  $Ax = b$  به گونه‌ای است که  $n < m$  باشد؛ که دستگاه در این حالت "فرومعین"<sup>۲</sup> نامیده می‌شود. یکی از مسائل اساسی در پردازش سیگنال که در سالهای اخیر مورد توجه محققین قرار گرفته است، یافتن جواب‌های تنگ دستگاه معادلات خطی "فرومعین" است [۱، ۲، ۳]. به

<sup>۱</sup>Least Squares

<sup>۲</sup>Underdetermined

عنوان نمونه، می توان به مسئله یافتن نمایش تنک برای یک سیگنال روی یک دیکشنری "فوق کامل"<sup>۱</sup> از سیگنال های از پیش تعیین شده اشاره کرد [۱]. (اصطلاحاً "تجزیه اتمی"<sup>۲</sup>). نمایش تنک سیگنالها را می توان برای فشرده سازی<sup>۳</sup> به کار برد که این کاربرد غالباً تحت عنوان "کدینگ تنک"<sup>۴</sup> مطرح می شود و به خصوص در حوزه پردازش تصویر مورد توجه است [۴]. نمایش تنک<sup>۵</sup>، یک سیگنال را به صورت یک ترکیب خطی از تعداد کمی از پایه های مشخص توصیف می کند. می دانیم دستگاه معادلات خطی در حالت فرومعین یک جواب یکتا ندارد. اما می توان با اضافه کردن شرایطی، جواب هایی از دستگاه های فرومعین را هم به طور یکتا مشخص کرد. یکی از این شرایط، شرط تنک بودن جواب است. در فصل بعد خواهیم دید که با فرض های ساده ای می توان نشان داد که جواب های "به اندازه کافی تنک" دستگاه های معادلات خطی "فرومعین" یکتا هستند. به این ترتیب، با محدود کردن دامنه های تبدیل مربوطه به جواب های به اندازه کافی تنک، یک تبدیل وارون پذیر به دست می آوریم. این جواب های تنک در کاربردهای پردازش سیگنال از جمله فشرده سازی، نویززدایی تصویر، فراتفکیک پذیری<sup>۶</sup>، طبقه بندی تصاویر<sup>۷</sup>، تشخیص چهره<sup>۸</sup> و...، نمایش های بسیار کارا<sup>۹</sup> و فشرده<sup>۱۰</sup> از سیگنالها به دست می دهد که عملاً برای تحلیل و پردازش مفیدند [۵، ۶، ۷، ۸، ۹]. قبل از طرح مسئله اصلی، در ابتدا تعاریف مورد نیاز در مبحث پردازش تنک را مطرح می کنیم:

**تعریف ۱-۱-۱** (ماتریس رتبه کامل). یک ماتریس رتبه کامل<sup>۱۱</sup> نامیده می شود اگر تمام سطرها و ستون های آن مستقل خطی باشد [۱۰].

**تعریف ۱-۱-۲** (اتم). به بردارهای پایه در رویکرد نمایش تنک، اتم<sup>۱۲</sup> می گویند.

**تعریف ۱-۱-۳** (دیکشنری). به ماتریسی که ستون های آن شامل پایه ها یا اتم ها در تعریف قبل است، دیکشنری<sup>۱۳</sup> می گویند.

**تعریف ۱-۱-۴** (یادگیری دیکشنری). عمل استخراج بردارهای پایه، یادگیری دیکشنری<sup>۱۴</sup> نامیده می شود.

**تعریف ۱-۱-۵** (کدینگ تنک). کدینگ تنک، بردارهای داده را بصورت ترکیب خطی مینیمم تعداد از اتم های دیکشنری بیان می کند.

دو مسئله مهم در نمایش تنک، اولاً پیدا کردن دیکشنری فوق کامل مناسب برای یک کلاس مشخص از داده ها و دوماً یافتن یک الگوریتم کارا برای به دست آوردن تنک ترین نمایش سیگنال است که مسئله مورد نظر ما در این پایان نامه نیز می باشد. حال می خواهیم مسئله را در قالب دیدگاه "تجزیه اتمی" بیان کنیم.

---

<sup>۱</sup>Overcomplete    <sup>۲</sup>Atomic decomposition    <sup>۳</sup>compression    <sup>۴</sup>Sparse Coding    <sup>۵</sup>Sparse Representation  
<sup>۶</sup>Super-Resolution    <sup>۷</sup>Image classification    <sup>۸</sup>Image Recognition    <sup>۹</sup>efficient  
<sup>۱۰</sup>compact    <sup>۱۱</sup>Full rank    <sup>۱۲</sup>Atom    <sup>۱۳</sup>Dictionary    <sup>۱۴</sup>Dictionary Learning

در دیدگاه "تجزیه اتمی" یک بردار سیگنال (یا بردار مخلوط)  $1 \times n$  مانند  $x$  داده شده است و هدف یافتن نمایشی از آن به شکل یک ترکیب خطی از  $m$  بردار سیگنال از پیش تعیین شده مانند  $\{a_i\}_{i=1}^m$  است. بردارهای  $a_i$  را اصطلاحاً اتم و مجموعه‌ی آن‌ها را دیکشنری می‌گویند. به بیان ریاضی داریم:

$$x = \sum_{i=1}^n s_i a_i = As \quad (3-1)$$

که در آن  $A$  ماتریس  $n \times m$  دیکشنری است (که ستون‌های آن اتمهای  $a_i$  هستند) و  $s$  بردار  $1 \times m$  ضرایب (یا بردار منبع) است. برای نشان دادن هر بردار  $1 \times n$ ، تنها به مجموعه‌ای از  $n$  اتم مستقل نیاز داریم. ضرایب چنین بسطی در حالت کلی همگی می‌توانند مقادیر قابل ملاحظه‌ای داشته باشند. اما اگر تعداد اتم‌ها را (خیلی) بیشتر از حد مورد نیاز بگیریم ( $m \gg n$ ) احتمال اینکه بتوان برای یک بردار سیگنال، نمایشی تنها بر حسب تعداد محدودی از اتم‌ها (یعنی کمتر از  $n$ ) به دست آورد، به شدت افزایش می‌یابد؛ در چنین حالتی، بیشتر ضرایب بسط ناچیز و یا به عبارتی بردار ضرایب بسط (یعنی بردار  $s$ ) تنک خواهد بود. در واقع با انتخاب مناسب دیکشنری ممکن است بتوان برای تمام سیگنال‌های موجود در یک فضای سیگنال مورد نظر، نمایش تنک پیدا کرد. یک نمایش تنک بر حسب تعداد محدودی از اتم‌های شناخته شده، ساختار سیگنال را بهتر آشکار می‌کند و برای نگهداری، پردازش و انتقال سیگنال کاراتر است. (به جای  $n$  مولفه اولیه  $x$  با مولفه‌های غیر صفر  $s$  کار خواهیم کرد که تعدادشان بسیار کمتر است.) مسئله مورد بحث همچنین تحت عنوان "نمایش تنک"، "تجزیه تنک" <sup>۱</sup> و "تجزیه اتمی" نیز شناخته می‌شود.

## ۲-۱ راهکارهای پیشنهادی در حل مسئله

هدف اصلی ما، یافتن تنک‌ترین جواب دستگاه  $x = As$  است. تاکنون معیارهای بسیاری برای تنک بودن معرفی شده‌اند. یکی از معیارهای ساده برای سنجش تنک بودن یک بردار، نرم  $l_0$  است که به طور شهودی عبارت است از تعداد عناصر غیر صفر آن بردار. به این ترتیب عملیات جستجوی تنک‌ترین جواب را می‌توان به صورت یک مسئله بهینه‌سازی مطرح کرد:

$$P_0 : \min \|s\|_0 \quad \text{s.t.} \quad x = As \quad (4-1)$$

<sup>۱</sup>Sparse Decomposition



اما مشکل اساسی حل این مسئله و جوب نیاز به جست و جوی ترکیبیاتی<sup>۱</sup> است. علاوه بر این در بسیاری از کاربردهای واقعی سیستم  $x = As$  با نویز همراه است لذا با تساوی  $x = As + v$  روبرو هستیم. برای این منظور به جای حل مسئله ۳-۸ باید مسئله زیر را که نسبت به نویز پایدار است [۱۱]، حل شود:

$$P^\epsilon : \min \|s\|_0 \quad \text{s.t.} \quad \|x - As\|_2 < \epsilon \quad (5-1)$$

برای حل مسئله ۲-۲ تاکنون روش های متعددی پیشنهاد شده است که از جمله این روش ها میتوان الگوریتم جست و جوی تطابقی<sup>۲</sup> و الگوریتم جست و جوی تطابقی متعامد<sup>۳</sup> را نام برد که از جمله الگوریتم های حرصانه<sup>۴</sup> هستند [۱۲، ۱۳].

معیار دیگری که برای سنجش تنگ بودن مورد استفاده قرار گرفته است نرم  $l_1$  است؛ یعنی جوابی از دستگاه  $x = As$  را جستجو کرد که کمترین نرم  $l_1$  را داراست:

$$P_1 : \min \|s\|_1 \quad \text{s.t.} \quad x = As \quad (6-1)$$

تعریف ۱-۲-۱ (مجموعه محدب). فرض کنید  $S$  یک زیر مجموعه از  $\mathbb{R}^n$  باشد. آنگاه گوئیم  $S$  محدب<sup>۵</sup> است اگر برای هر  $x, y \in S$  و هر  $\alpha \in [0, 1]$   $\alpha x + (1 - \alpha)y$  متعلق به مجموعه  $S$  باشد.

تعریف ۲-۲-۱ (تابع محدب). فرض کنید  $S$  یک زیر مجموعه محدب از  $\mathbb{R}^n$  باشد. یک تابع  $f : S \rightarrow \mathbb{R}$  تابع محدب<sup>۶</sup> نامیده می شود هر گاه برای هر  $x, y \in S$  و هر  $\alpha \in [0, 1]$  داشته باشیم

$$f(\alpha x + (1 - \alpha)y) \leq \alpha f(x) + (1 - \alpha)f(y)$$

تعریف ۳-۲-۱ (بهینه سازی محدب). به یافتن مقدار کمینه یک تابع محدب (یا بیشینه یک تابع مقعر) از بین مجموعه ای محدب، بهینه سازی محدب<sup>۷</sup> یا بهینه سازی کوژ گفته می شود. به عبارت دیگر، مسئله بهینه سازی که تابع هدف و توابع شرایط آن محدب باشند محدب گفته می شود. مهم ترین مزیت این نوع مسائل بهینه سازی در این است که هر نقطه بهینه محلی یک نقطه بهینه سراسری نیز هست و هر الگوریتم بهینه سازی که یک نقطه بهینه محلی را بیابد، در حقیقت یک نقطه بهینه سراسری را یافته است.

نرم  $l_1$  حساسیت بسیار کمتری نسبت به نویز دارد و همچنین مسئله ۱-۶ یک مسئله بهینه سازی محدب<sup>۸</sup>

---

<sup>۱</sup>Combinatorial Search Algorithms      <sup>۲</sup>Matching Pursuit      <sup>۳</sup>Orthogonal Matching Pursuit      <sup>۴</sup>Greedy Algorithms  
<sup>۵</sup>convex set      <sup>۶</sup>convex function      <sup>۷</sup>convex optimization      <sup>۸</sup>convex optimization

است و همین اساس موفق ترین روش تجزیه تنک ، تحت عنوان ” جستجوی پایه “<sup>۱</sup> است . مسئله ۱-۶ معادل یک مسئله برنامه ریزی خطی (LP)<sup>۲</sup> است و لذا می توان از روش های نقطه درونی<sup>۳</sup> برای حل مسئله استفاده کرد .

علاوه بر این دو معیار ، تعبیر دیگری از تنک بودن که در روش ” جستجوی زیرفضا “ (SP)<sup>۴</sup> مطرح شده و به تعبیری منجر به کمینه سازی نرم  $l_1$  خواهد شد [۱۴] را ، بیان کرده و از ایده های این روش برای پیاده سازی نسخه ای از خانواده روش های ” آشکار سازی - تخمین پیایی “ (IDE)<sup>۵</sup> استفاده خواهد شد . ایده ی اصلی روش IDE آشکار سازی ( تقریبی ) منابع ” فعال “ ( مولفه هایی از بردار  $s$  که دارای مقداری قابل ملاحظه اند ) است که با این ایده می توان نیاز به جست و جوی ترکیبیاتی را از بین برد و معیارهای بهینگی برای یافتن تنک ترین جواب را عملاً پیاده کرد [۱۵] .

---

<sup>۱</sup>Basis Pursuit    <sup>۲</sup>Linear Programing    <sup>۳</sup>interior-point methods    <sup>۴</sup>Subspace Pursuit    <sup>۵</sup>Iterative  
Detection-Estimation

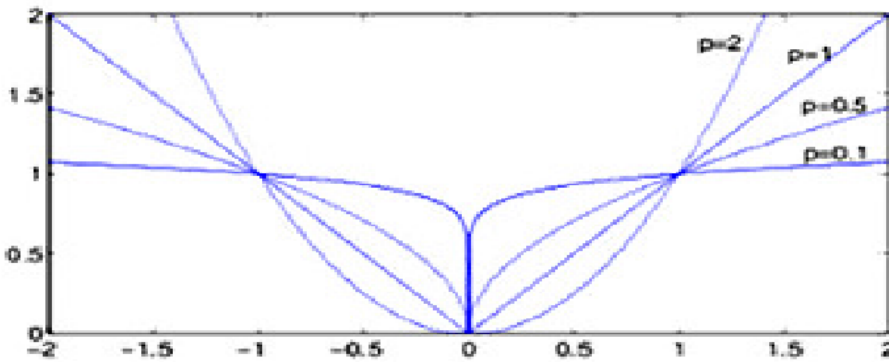
## فصل ۲

# روش های تعیین نمایش تنک سیگنال

### ۱-۲ معرفی برخی اصطلاحات و قراردادهای

در این بخش برخی اصطلاحات و مفاهیم را که در این فصل و فصل های آتی مورد استفاده قرار خواهد گرفت ، معرفی می شود . منظور از فضای  $x$ - همان فضای  $\mathbb{R}^n$  است که بردار  $x$  به آن تعلق دارد و همچنین فضای  $s$ - در واقع همان فضای  $\mathbb{R}^m$  است . مفهوم دیگری که زیاد از آن استفاده می شود مفهوم ” فعال بودن “ یک منبع است . در فصل قبل ، با یک دید ساده تنک بودن بردار منبع ( $s$ ) را به این معنا گرفتیم که بیشتر مولفه های آن غیر صفرند ؛ یا به عبارت دیگر نرم  $l_2$  آن خیلی کمتر از  $m$  است . با این تعریف با قرار گرفتن نویز کوچکی روی مولفه های صفر بردار منبع ( $s$ ) ، بردار مذکور عملاً از دایره ی تعریف تنک بودن خارج می شود ؛ حال آن که چنین برداری برای بسیاری از مقاصد عملی ، از نظر ما ، همچنان تنک است . مولفه هایی از بردار را که ” مقدار قابل ملاحظه ای “ دارند ، مولفه های ” فعال “ و آنها که مقدار ناچیزی دارند را مولفه های ” غیر فعال “ می نامیم . به این ترتیب ، با توسعه ی مفهوم تنک بودن ، برداری تنک است که مولفه های غیر فعال زیادی داشته باشد . گاهی اصطلاح ” وضعیت فعالیت “ یک مولفه یا یک منبع ( $s_i$ ) را به کار می بریم . متذکر می شویم مفهوم ” فعال بودن “ یک معیار کیفی است و هر جا که بخواهیم از این اصطلاح به شکل کمی استفاده کنیم آن را به طور دقیق تر تعریف می کنیم.

همچنین در پایان قرارداد می کنیم که ستون های ماتریس دیکشنری  $A$  ، همواره به نرم  $l_2$  واحد نرمالیزه شده اند و همچنین در آزمایش ها این ماتریس را ” به طور تصادفی “ انتخاب می کنیم ؛ منظور این است که  $m$  ستون آن را از یک توزیع یکنواخت روی کره ی واحد در  $\mathbb{R}^n$  به دست می آوریم .



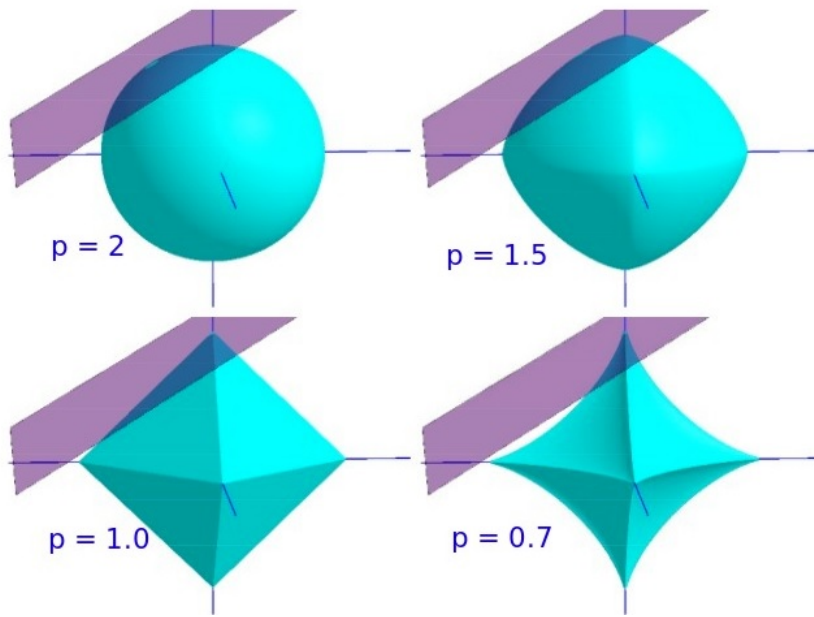
شکل ۲-۱: نرم های مختلف

## ۲-۲ تعیین نمایش تنگ سیگنال

همان گونه که در فصل قبل گفتیم، برای یافتن تنگ ترین جواب دستگاه  $x = As$  باید قیدی به مسئله اضافه شود که تعداد مولفه های غیر صفر نمایش حاصل را محدود کند. از آن جا که نرم صفر یک بردار، معرف تعداد مولفه های غیر صفر آن می باشد، لذا کمینه سازی نرم صفر منجر به تحمیل شرط تنگی به بردار خواهد شد. برای به دست آوردن تنگ ترین جواب باید مسئله

$$p_0 : \min \|s\|_0 \quad s.t. \quad x = As \quad (1-2)$$

حل شود، که در آن  $\|s\|_0 \triangleq |supp(s)| = \{i : s_i \neq 0\}$  نشان دهنده تعداد مولفه های غیر صفر بردار  $s$  است. از آن جا که نرم صفر محدب نیست و از آن مهم تر، مشتق پذیر نیست، مسئله فوق NP-Hard تلقی می شود و حل آن منوط به جستجوی کامل است که به وضوح در ابعاد بالا غیر عملی است. به عبارت دیگر، برای پیدا کردن جوابی با  $k$  درایه ی غیر صفر، لازم است در بدترین حالت تعداد  $N = \binom{m}{k}$  زیر ماتریس از  $A$  که هر کدام نماینده ی یک زیر فضا است، مورد بررسی قرار گیرد. یکی از راهکارهای حل این نوع مسئله، استفاده از الگوریتم های حریصانه است که در فصل بعد دو الگوریتم جستجوی تطابقی و جستجوی تطابقی متعامد را به تفصیل شرح خواهیم داد. یکی دیگر از راهکارهای مهم در مواجهه با سیستم های خطی با فرض تنگ بودن جواب، تقریب زدن نرم صفر با نرمی از مرتبه بالاتر است که قابلیت کمینه کردن آن ساده تر باشد. طبیعی است که هر چه از نرم مرتبه پایین تری استفاده شود، جواب حاصل از کمینه سازی به جواب حاصل از کمینه سازی نرم صفر نزدیک تر خواهد بود. همان طور که در شکل ۲-۱ مشاهده می شود، تابع نرم به ازای  $0 \leq p < 1$  محدب نیست و هر چه به سمت یک نزدیک می شود، تابع هموارتر شده و به ازای  $p = 1$  تابع محدب خواهد بود.



شکل ۲-۲: تقاطع بین توپ های  $l_p$  و سیستم  $x = As$  با استفاده از حل  $p_p$  در فضای سه بعدی

به طور کلی دسته ای از توابع مسئله ی ما را به سمت تنگی میل می دهد ، به صورت  $J(s) = \|s\|_p^p$  با  $0 \leq p \leq 1$  است [۱۰]. از میان این توابع تنها  $\|s\|_1$  تمام خواص نرم را داراست و بقیه این توابع شبه نرم خوانده می شوند . با این انتخاب مسئله متناظر به فرم زیر است :

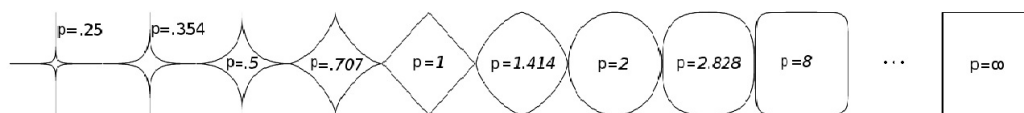
$$P_p : \min \|s\|_p^p \quad \text{s.t.} \quad x = As \quad (2-2)$$

برای درک بهتر ، به تعبیر هندسی نرم  $l_p$  از مرجع [۱۰] توجه کنید . برای به دست آوردن جواب مسئله ۲-۲ از دید هندسی ، باید اصطلاحاً در توپ های  $l_p$  که به صورت  $\|s\|_p = cte$  تعریف می شوند ، دمید تا جایی که اولین برخورد با زیر فضای آفین ذکر شده حاصل شود .

تعریف ۲-۲-۱ (فضای آفین). در ریاضیات ، فضای آفین<sup>۱</sup> ساختار هندسی ای است که ویژگی های فضای اقلیدسی را تعمیم می دهد ؛ که مستقل از مفاهیم فاصله و اندازه گیری زاویه هستند ، و تنها خواص مربوط به موازی بودن را نگه می دارند . در یک تبدیل آفین خطوط موازی ، موازی باقی می مانند .

در شکل ۲-۲ فصل مشترک زیر فضای آفین متناظر با  $x = As$  ، با توپ های  $l_p$  برای چند مقدار مختلف  $p$  نشان داده شده است .

<sup>۱</sup>Affine space



با توجه به شکل دو نکته مهم را می توان دریافت ؛ اول این که هر چه  $p$  به سمت صفر میل می کند به پاسخ های تنک تری می رسیم و دوم این که نرم یک به عنوان بهترین تقریب محدب شبه نرم صفر ، به جواب های تنکی می انجامد . از آن جا که نرم یک ( $l_1$ ) نزدیک ترین نرم محدب به نرم صفر است ، جایگزینی نرم صفر با نرم یک ، منطقی ترین تقریب به نظر می رسد :

$$p_1 : \min \|s\|_1 \quad s.t. \quad x = As \quad (3-2)$$

برای مسئله اخیر ، راه حل هایی به کمک برنامه ریزی خطی معرفی شده اند که به خانواده ” جستجوی پایه “ معروفند و پیچیدگی محاسباتی آنها از مرتبه  $n^3$  است [۱۶] .  
 اما پیش از آن که راه حل های ارائه شده برای مسئله شده برای مسئله یافتن تنک ترین جواب را شرح دهیم ، لازم است قضایای مربوط به یکتایی این جواب را بیان کنیم .

## ۳-۲ یکتایی تنک ترین جواب

در این قسمت قضایای مربوط به یکتایی تنک ترین نمایش سیگنال از مرجع [۱۰] بیان می شوند ( تمامی تعاریف و قضایای این بخش در خصوص یکتایی تنک ترین جواب از مرجع [۱۰] استخراج شده اند) . ابتدا قضیه یکتایی با استفاده از مفهوم اسپارک<sup>۱</sup> دیکشنری بیان می شود . اسپارک یک ماتریس روشی برای توصیف فضای پوچ<sup>۲</sup> آن ماتریس است .

**تعریف ۳-۲-۱** (اسپارک یک ماتریس) . اسپارک یک ماتریس عبارت است از کم ترین تعداد ستون هایی از آن که وابسته خطی اند .

بنا به این تعریف ، هر بردار  $y$  که در رابطه  $Ay = 0$  صدق کند (یعنی بردار  $y$  متعلق به فضای پوچ  $A$  باشد) ، لزوما شرط

$$\|y\|_0 \geq \text{spark}(A) \quad (4-2)$$

<sup>۱</sup>Spark      <sup>۲</sup>Nall-space

برقرار خواهد بود. چون بنا به تعریف اسپارک ماتریس، تعداد ستون هایی از  $A$  که با یک ترکیب خطی منجر به بردار  $0$  می شوند، باید از  $\text{spark}(A)$  بیشتر یا مساوی باشند. با این تعریف، قضیه یکتایی به صورت زیر بیان می شود:

**قضیه ۲-۳-۲.** اگر جوابی از دستگاه فرامعین  $x = As$  در شرط  $\|s\|_0 < \text{spark}(A)/2$  صدق کند، آنگاه لزوماً تنگ ترین جواب این دستگاه معادلات خواهد بود.

**برهان.** فرض کنید  $s'$  جواب دلخواه دیگری از دستگاه معادلات خطی مفروض باشد؛ یعنی  $As' = x$ . بنابراین، بردار  $s - s'$  متعلق به فضای پوچ ماتریس  $A$  است؛ یعنی  $A(s - s') = 0$ . لذا بنا به تعریف نرم  $l_0$  و رابطه ۲-۴ داریم:

$$\|s\|_0 + \|s'\|_0 \geq \|s - s'\|_0 \geq \text{spark}(A) \quad (5-2)$$

نامساوی سمت چپ به این معناست که تعداد عناصر غیرصفر بردار  $s - s'$  نمی تواند بیشتر از مجموع عناصر غیرصفر بردار  $s$  و  $s'$  باشد. از طرفی بنا به فرض  $\|s\|_0 < \text{spark}(A)/2$ ، بنابراین خواهیم داشت:

$$\|s'\|_0 \geq \text{spark}(A)/2 \quad (6-2)$$

□ (برای هر  $s'$  که  $As' = x$ ) بنابراین  $s$  تنگ ترین جواب دستگاه  $As = x$  خواهد بود.

همان گونه که از تعریف ۲-۳-۱ بر می آید، یافتن اسپارک یک ماتریس خود نیازمند جستجوی ترکیباتی است. بنابراین، به روش ساده تری برای تضمین یکتایی تنگ ترین جواب نیاز است. یک راه بسیار ساده استفاده از همبستگی متقابل بین اتم های دیکشنری است.

**تعریف ۲-۳-۳ (همبستگی متقابل).** <sup>۱</sup> همبستگی متقابل عبارت است از ماکزیمم قدر مطلق ضریب همبستگی بین ستون های متمایز آن. به بیان ریاضی، همبستگی متقابل برای ماتریس  $A$  به صورت زیر تعریف می شود:

$$\mu(A) = \max_{i \neq j} \frac{|a_i^T a_j|}{\|a_i\|_2 \|a_j\|_2}$$

به این ترتیب، همبستگی متقابل کمیتی برای ارزیابی وابستگی خطی بین ستون های یک ماتریس است. هرچه ضریب همبستگی کوچک تر باشد، ماتریس به متعامد بودن نزدیک تر است.

<sup>۱</sup>Mutual-coherence

لم ۲-۳-۴. برای هر ماتریس  $A \in \mathbb{R}^{n \times m}$  رابطه زیر برقرار است:

$$\text{spark}(A) \geq 1 + 1/\mu(A) \quad (7-2)$$

□ برهان. به مرجع [۱۰] مراجعه شود.

قضیه یکتایی با استفاده از همبستگی متقابل دیکشنری به صورت زیر بیان می شود:

قضیه ۲-۳-۵. اگر جوابی از دستگاه فرومعی  $x = As$  در شرط  $\|s\|_0 < \frac{1}{\mu(A)}(1 + 1/\mu(A))$  صدق کند، آنگاه لزوماً تنگ ترین جواب این دستگاه معادلات خواهد بود.

□ برهان. با استفاده از قضیه ۲-۳-۲ و لم ۲-۳-۴ مستقیماً نتیجه می شود.

اگر جوابی از  $x = As$  در شرط قضیه ۲-۳-۵ صدق کند، آنگاه این جواب هم جواب مسئله:

$$p_0 : \min \|s\|_0 \quad \text{s.t.} \quad x = As \quad (8-2)$$

وهم جواب مسئله:

$$p_1 : \min \|s\|_1 \quad \text{s.t.} \quad x = As \quad (9-2)$$

خواهد بود. به عبارت دیگر، در این وضعیت دو مسئله معادل خواهند بود [۳]. لازم به ذکر است که معمولاً  $\text{spark}(A) \gg 1 + 1/\mu(A)$ ؛ در نتیجه، شرطی که قضیه ۲-۳-۵ روی تنگ بودن، نمایش یکتای سیگنال می گذارد، بسیار محدود کننده تر از شرط قضیه ۲-۳-۲ است.

## ۴-۲ پایداری تقریب تنگ سیگنال

همان گونه که در فصل قبل متذکر شدیم، تساوی  $x = As$  عموماً برقرار نیست. در بسیاری از کاربردها، با تساوی  $x = As + v$  مواجه هستیم که در آن بردار  $v$ ، برداری با انرژی محدود است که بسته به کاربرد، تعبیر مختلفی دارد. به عنوان مثال در فشرده سازی، این بردار نشان دهنده نویزی است که سیگنال تمیز را آلوده کرده



است. در چنین حالتی، به منظور پایداری جواب، باید تنگ ترین پاسخ را در ابر کره  $\|x - As\|_2 \leq \epsilon$  یافت  
 . بنا بر این به جای دو مسئله  $p_0$  و  $p_1$ ، باید از نسخه های پایدار نسبت به نویز آنها استفاده کرد [۱۱]:

$$p_0^\epsilon : \min_s \|s\|_0 \quad s.t. \quad \|x - As\|_2 \leq \epsilon \quad (10-2)$$

و

$$p_1^\epsilon : \min_s \|s\|_1 \quad s.t. \quad \|x - As\|_2 \leq \epsilon \quad (11-2)$$

به این ترتیب با حل هریک از مسائل فوق، می توان تخمینی از سیگنال بدون نویز  $x_0$  به صورت  $\hat{x}_0 = A\hat{s}$  به دست  
 آورد که در آن،  $\hat{s}$  جواب یکی از دو مسئله فوق است. شرط استفاده شده در رابطه های بالا ( $\|x - As\|_2 \leq \epsilon$ )  
 به شرط LASSO شهرت دارد. در [۱۱] نشان داده شده است که هر دو مسئله فوق نسبت به نویز پایدارند. دو  
 قضیه زیر به نقل از این مرجع بیان می شوند:

**قضیه ۲-۴-۱.** سیگنال تمیز  $x_0 = As_0$  را در نظر بگیرید. در این صورت اگر

$$\|s_0\|_0 = k < \frac{1}{4}(1 + 1/\mu(A))$$

آنگاه:

$$\|s_0^\epsilon - s_0\|_2 \leq \frac{\epsilon + \delta}{\sqrt{1 - \mu(A)(4k - 1)}}, \forall \epsilon \geq \delta > 0$$

**قضیه ۲-۴-۲.** سیگنال تمیز  $x_0 = As_0$  را در نظر بگیرید. در این صورت اگر

$$\|s_0\|_0 = k < \frac{1}{4}(1 + 1/\mu(A))$$

آنگاه:

$$\|s_0^\epsilon - s_0\|_2 \leq \frac{\epsilon + \delta}{\sqrt{1 - \mu(A)(4k - 1)}}, \forall \epsilon \geq \delta > 0$$

پس از بیان پشتوانه های تئوری و قضایایی که یکتایی و پایداری نمایش تنگ سیگنال را اثبات می کنند، در  
 ادامه، تعدادی از الگوریتم های رایج برای به دست آوردن نمایش تنگ سیگنال ها معرفی می شوند.

## ۵-۲ روش های موجود

طی چند دهه ی اخیر الگوریتم های زیادی برای به دست آوردن تنک ترین نمایش یک سیگنال معرفی شده است [۱۷]. این الگوریتم ها عمدتاً به دو دسته کلی الگوریتم های مبتنی بر حل مسئله بهینه سازی والگوریتمهای حریصانه تقسیم می شوند .

روش های دسته اول ، تابع هزینه ای <sup>۱</sup> از بردار  $s$  تعریف می کنند که نشان دهنده ی میزان دوری  $s$  از تنک بودن است ( به عبارت دیگر ، هرچه تابع هزینه ی مذکور کوچک تر باشد ، بردار  $s$  تنک تر است ) ؛ سپس با مینیمم کردن این تابع هزینه روی مجموعه جواب  $x = As$  ، تنک ترین جواب را می یابند . طبیعی است که تابع هزینه ی انتخابی هم باید معیار بهتری از تنک بودن باشد تا دقت جواب نهایی بیشتر باشد و هم باید به گونه ای باشد که حل مسئله بهینه سازی مربوط به آن ساده تر باشد . همان طور که انتظار می رود ، این دو ویژگی غالباً در تضاد اند و تلاش برای یافتن معیارهای مناسب برای تنک بودن چالش اصلی روش های مبتنی بر حل مسئله بهینه سازی است . البته نمونه موفق از تابع هزینه ای که هر دو ویژگی را داراست مینیمم سازی نرم  $l_1$  است که در ادامه شرح خواهیم داد .

دسته دوم یعنی الگوریتم های حریص روش هایی هستند که به جای حل صریح یک مسئله بهینه سازی ، الگوریتمی برای انتخاب یک جواب از میان مجموعه ی جواب دستگاه ارائه می دهند . الگوریتم های حریص به صورت قدم به قدم یک یا چند اتم را که بیشترین همبستگی <sup>۲</sup> ( یا شباهت ) با باقی مانده ی <sup>۳</sup> مربوط به نمایش سیگنال را دارند ، انتخاب کرده و با استفاده از آن ها این باقی مانده را به روز می کنند . اساس این روش ها مبتنی بر تخمین مرحله به مرحله ی سیگنال با استفاده از اتم های دیکشنری است .

الگوریتمهای حریصانه به دو دسته کلی تقسیم می شوند : دسته اول الگوریتم هایی هستند که در هر مرحله یک اتم که بیشترین همبستگی را به باقی مانده ی نمایش سیگنال دارد را انتخاب می کند و دسته دوم الگوریتم هایی هستند که در هر مرحله بیش از یک اتم را انتخاب و در طی پیشروی الگوریتم تعدادی از این اتم ها را حذف کرده و یا اتم های جدیدی به این مجموعه اضافه می کند . دو روش جستجوی تطابقی ( MP ) و جستجوی تطابقی متعامد ( OMP ) از جمله روش های دسته اول و الگوریتم ( CoSaMP ) از جمله روش های دسته دوم هستند که از جمله الگوریتم های پرکاربرد در حوزه پردازش تنک می باشند .

---

<sup>۱</sup>cost function      <sup>۲</sup>correlation      <sup>۳</sup>residual

## روش های مبتنی بر حل مسئله بهینه سازی

### ۱-۵-۲ کمینه سازی نرم $l_1$

همان گونه که قبلا گفته شد، نرم  $l_1$  ساده ترین معیار برای تنک بودن است. یاد آوری می کنیم نرم  $l_1$  یک بردار برابر است با تعداد عناصر غیر صفر آن. اگر این کمیت را به عنوان معیاری از تنک بودن بردار  $s$  بپذیریم، می توان تنک ترین جواب را با حل مسئله بهینه سازی زیر بدست آورد:

$$(P_*) : \min \|s\|_1 \quad s.t. \quad x = As \quad (12-2)$$

تابع  $\|s\|_1$  از  $s$  به شدت غیر خطی و به علاوه ناپیوسته است؛ به این معنا که اگر یکی از مولفه های  $s$  اندکی از مقدار صفر جا به جا شود تابع  $\|s\|_1$  به اندازه یک واحد افزایش می یابد و همین ویژگی نشان می دهد که تابع  $\|s\|_1$  نسبت به نویز حساس است. به عبارت دیگر تابع  $\|s\|_1$  تنها برای یافتن جواب های تنک مسائلی کاربرد دارد که مولفه های غیر فعال بردار منبع اولیه "دقیقا" برابر صفر باشند؛ یا به بیان دیگر، سیگنال مورد نظر بسط تنک دقیقی بر حسب اتم های دیکشنری داشته باشد؛ هر گونه مقدار دیگری، هر چند ناچیز، تابع هزینه بردار مربوطه را به شدت افزایش داده و احتمال یافتن آن را کاهش می دهد. همچنین به دلیل رفتار (شدیدا) غیر هموار تابع  $\|s\|_1$  نمی توان از روش های کارایی که برای حل مسائل غیر خطی وجود دارند، برای حل مسئله  $P_*$  استفاده کرد و تنها راه موجود برای حل صریح این مسئله استفاده از جست و جوی ترکیبیاتی است که در ادامه اشاره ای به آن خواهیم داشت.

در حالت کلی دستگاه  $x = As$  (با  $n < m$ ) دارای جواب های با حداقل  $m - n$  مولفه صفر است. برای یافتن این جواب ها باید  $m - n$  مولفه را به دلخواه انتخاب کرده و برابر با صفر قرار دهیم، سپس با صفر قرار دادن  $m - n$  مولفه، دستگاه  $x = As$  به یک دستگاه  $n \times n$  (معادله و  $n$  مجهول) تبدیل خواهد شد. به طور مثال اگر  $m - n$  مولفه آخر بردار  $s$  را برابر صفر فرض کنیم، جواب دستگاه به صورت  $(s_{1:n} = A_{1:n}^{-1}x, s_{n+1:m} = 0)$  خواهد بود، که در آن  $A_{1:n}$  ماتریسی شامل اتم های  $a_1$  تا  $a_n$  است. در اصطلاح "برنامه ریزی خطی" به این جواب های به دست آمده "جواب های پایه ای" گفته می شود. هر کدام از این جواب ها متناظر با انتخاب یک پایه  $n$ -عضوی از  $m$  ستون ماتریس  $A$  است. در مثال گفته شده پایه متشکل از بردار های  $\{a_i\}_{i=1}^n$  است و همچنین آن بخش از جواب که متناظر با بردارهای پایه است، یعنی  $s_{1:n}$ ، بخش پایه ای جواب می باشد. تعداد جواب های پایه ای با فرض وارون پذیر بودن تمام دستگاه های  $n \times n$ ، دقیقا برابر با  $\binom{m}{n}$  است. اگر

<sup>1</sup>basic solutions

دستگاه  $x = As$  جواب هایی داشته باشد که بیش از  $m - n$  مولفه صفر داشته باشد، این جواب ها لزوما بین  $\binom{m}{n}$  جواب پایه ای خواهند بود. در این موارد علاوه بر بخش غیر پایه ای جواب تعدادی مولفه صفر در بخش پایه ای جواب وجود خواهد داشت که این جواب ها "تباهیده" خوانده می شوند. به عنوان نمونه، در مثال اخیر اگر پس از حل دستگاه  $x = A_{1:n} s_{1:n}$  تعدادی از مولفه های  $s_{1:n}$  برابر صفر به دست بیایند، جواب پایه ای مربوط تباهیده نامیده می شود.

با توجه به توضیحات گفته شده، تنک ترین جواب دستگاه  $x = As$  لزوما در میان جواب های پایه ای دستگاه  $x = As$  قرار دارد و غالبا تباهیده است. بنابراین برای یافتن جواب (دقیق) دارای کمترین نرم  $l_0$ ، باید آن را در میان  $\binom{m}{n}$  جواب پایه ای جستجو کرد، که الگوریتم کارایی برای اینکار وجود ندارد. لذا ناچاراً باید تمام  $\binom{m}{n}$  جواب را تک تک بررسی کرد یا به عبارت دیگر، نیاز به جستجوی ترکیباتی (جستجوی فرد به فرد حالات ممکن) داریم. البته روش های MP و OMP از جمله الگوریتمهای حریص برای فرار از این نوع جستجو در یافتن تقریب تنک ترین جواب است که در ادامه در مورد آن ها شرح خواهیم داد.

**مثال ۲-۵-۱.** مساله بهینه سازی زیر را در نظر بگیرید:

$$\min z = \|s\|. \quad (۲-۱۳)$$

$$\text{s.t.} \quad s_1 + s_2 + 4s_3 - s_4 + 3s_5 = 1 \quad (۲-۱۴)$$

$$s_1 - s_2 - 2s_3 + 3s_4 + s_5 = 1 \quad (۲-۱۵)$$

برای این مسئله  $10 = \binom{5}{2}$  جواب پایه ای وجود دارد :

$$(nonbasic\ variables : s_2 = s_4 = s_5 = 0 \Rightarrow basic : s_1 = 1, s_3 = 0, z = 1) \quad (16-2)$$

$$(nonbasic\ variables : s_2 = s_4 = s_5 = 0 \Rightarrow basic : s_1 = 1, s_3 = 0, z = 1) \quad (17-2)$$

$$(nonbasic\ variables : s_2 = s_3 = s_5 = 0 \Rightarrow basic : s_1 = 1, s_4 = 0, z = 1) \quad (18-2)$$

$$(nonbasic\ variables : s_2 = s_3 = s_4 = 0 \Rightarrow basic : s_1 = 1, s_5 = 0, z = 1) \quad (19-2)$$

$$(nonbasic\ variables : s_1 = s_4 = s_5 = 0 \Rightarrow basic : s_2 = -3, s_3 = 1, z = 2) \quad (20-2)$$

$$(nonbasic\ variables : s_1 = s_3 = s_5 = 0 \Rightarrow basic : s_2 = 2, s_4 = 1, z = 2) \quad (21-2)$$

$$(nonbasic\ variables : s_1 = s_3 = s_4 = 0 \Rightarrow basic : s_2 = \frac{-1}{2}, s_5 = \frac{1}{2}, z = 2) \quad (22-2)$$

$$(nonbasic\ variables : s_1 = s_2 = s_5 = 0 \Rightarrow basic : s_3 = \frac{2}{5}, s_4 = \frac{3}{5}, z = 2) \quad (23-2)$$

$$(nonbasic\ variables : s_1 = s_2 = s_4 = 0 \Rightarrow basic : s_3 = \frac{-1}{5}, s_5 = \frac{3}{5}, z = 2) \quad (24-2)$$

$$(nonbasic\ variables : s_1 = s_2 = s_3 = 0 \Rightarrow basic : s_4 = \frac{1}{5}, s_5 = \frac{2}{5}, z = 2) \quad (25-2)$$

چنان که ملاحظه می شود جواب های ۲-۱۶ تا ۲-۱۹ جواب های بهینه این مسئله هستند ( جواب بهینه چندگانه ) که همگی جواب های تباهیده اند .

## ۲-۵-۲ کمینه سازی نرم $l_1$

” برنامه ریزی خطی “ مسئله بهینه سازی ای است که تابع هدف آن یک تابع خطی از متغیرهاست و محدودیت ها شامل معادلات ( و نامعادلات ) خطی است . فرم استاندارد برنامه ریزی خطی به صورت زیر است [۱۸] :

$$\min z = c^T x \quad (26-2)$$

$$s.t. \quad Ax = b, \quad x \geq 0$$

جواب هایی از دستگاه  $Ax = b$  که در شرط  $x \geq 0$  صدق کند ، جواب های شدنی<sup>۱</sup> نام دارند ؛ به عبارتی آن ها را می توان به عنوان گزینه های شدنی برای جواب بهینه در نظر گرفت.

**قضیه ۲-۵-۲** (قضیه اساسی برنامه ریزی خطی). با توجه به یک برنامه ریزی خطی در فرم استاندارد ۲-۲۶ که

<sup>۱</sup>feasible solutions

$A$  یک ماتریس  $n \times m$  از رتبه  $n$  است :

(i) اگر یک جواب شدنی وجود داشته باشد ، یک جواب پایه ای شدنی وجود دارد .

(ii) اگر یک جواب بهینه شدنی وجود داشته باشد ، یک جواب پایه ای شدنی بهینه وجود دارد .

□

برهان. برای برهان به مرجع [۱۸] مراجعه کنید.

بر این اساس یافتن جواب بهینه را به یافتن جواب بهینه در میان جواب های پایه ای شدنی دستگاه  $Ax = b$  محدود کرد . با این اعتبار ، LP را می توان یک روش " جستجوی پایه " (BP) دانست که سعی می کند یک پایه ی بهینه را از میان  $\binom{m}{n}$  پایه ی ممکن انتخاب کند . ایده ی اصلی این الگوریتم کمینه کردن نرم  $l_1$  است . پایه ی انتخاب شده به گونه ای است که نمایش بردار  $b$  بر حسب آن ( یعنی  $x$  ) تابع هزینه  $c^T x$  را مینیمم می کند . با توجه به ذات این تابع می توان انتظار داشت که  $x$  بهینه تا حد امکان تنک باشد . همان گونه که قبلا اشاره کردیم ، یکی از راه های یافتن جواب مسئله

$$(P_0) : \min \|s\|_1. \quad (27-2)$$

$$\text{s.t. } x = As$$

تقریب زدن نرم  $l_0$  با نرم  $l_1$  است :

$$(P_1) : \min \|s\|_1 = \sum_{i=1}^m |s_i| \quad (28-2)$$

$$\text{s.t. } x = As$$

با این خصوصیت که نرم  $l_1$  تابعی محدب و خطی است ؛ مهم تر از آن ، می توان نشان داد که مسئله  $P_1$  معادل است با یک مسئله برنامه ریزی خطی و لذا می توان از الگوریتمهای تعبیه شده برای مسائل برنامه ریزی خطی بهره جست . توجه کنید اگر پاسخ سیستم خطی فرومعین به اندازه کافی تنک باشد ( در نتیجه یکتاست و ) جواب بهینه محلی مسئله  $P_1$  همان جواب بهینه سراسری است . علاوه بر این ، مسئله  $P_1$  نسبت به مسئله  $P_0$  حساسیت کمتری نسبت به نویز دارد . همان طور که در بخش قبل گفته شد ، با تغییر ناچیز هر یک از مولفه های بردار  $s$  مقدار هدف مسئله  $P_0$  به شدت تغییر می کند و همین نشان دهنده حساسیت مسئله  $P_0$  نسبت به نویز است ، اما در مسئله  $P_1$  چنین نیست .

برای جستجوی جواب بهینه شدنی از میان جواب های پایه ای مسائل خطی روش کارایی وجود دارد که همان

روش معروف سیمپلکس<sup>۱</sup> است [۱۸]. در این روش، با داشتن یک جواب پایه ای شدنی؛ یا به عبارتی یک پایه از  $\{a_i\}$  ها؛ در هر مرحله یکی از این بردارها با یکی از ستون های  $A$  که در پایه حضور ندارد به نحوی تعویض می شود که  $s$  به دست آمده مقدار هدف را کاهش دهد. قابل ذکر است، با کمک قواعد "ضد دوری"<sup>۲</sup> می توان همگرایی به سمت جواب بهینه را تضمین کرد [۱۸].

برای درک تعبیر هندسی روش سیمپلکس، مجموعه جواب های شدنی  $\{x \mid Ax = b, x \geq 0\}$  مسئله ۲-۲۶ را که یک چند وجهی محدب است، در نظر بگیرید. روش سیمپلکس با شروع از یک جواب پایه ای شدنی، که یک نقطه گوشه ای از این چند وجهی است، شروع کرده و در هر مرحله از یک راس به راس مجاور به نحوی جابه جا می شود که مقدار هدف کاهش یابد.

خانواده روش های دیگری که برای حل مسائل برنامه ریزی خطی وجود دارد، "روش های نقطه درونی" هستند. این روش ها با شروع از یک نقطه درون چندوجهی مذکور در نهایت به نقطه ی بهینه روی مرز این چند وجهی می رسند. این روش ها در مسائل با ابعاد بالا بسیار کاراتر از روش سیمپلکس هستند. به علاوه خانواده روش های نقطه درونی را می توان علاوه بر مسائل LP برای هر مسئله برنامه ریزی محدب (خطی یا غیر خطی) در زمانی که به شکل یک چندجمله ای با ابعاد مسئله افزایش می یابد، به کار برد.

اشاره می کنیم که مسئله  $P_1$  (۲-۲۸) را با تغییر متغیر  $s = u - v$  می توان به مسئله LP به فرم استاندارد در آورد [۱۹]:

$$\min (1^T \quad -1^T) \begin{pmatrix} u \\ v \end{pmatrix} = \sum_{i=1}^m (u_i + v_i) \quad (2-29)$$

$$s.t \quad (A \quad -A) \begin{pmatrix} u \\ v \end{pmatrix} = x \quad (2-30)$$

$$\begin{pmatrix} u \\ v \end{pmatrix} \geq 0 \quad (2-31)$$

(توجه داشته باشید با این تغییر متغیر ابعاد مسئله دوبرابر می شود.)

---

<sup>۱</sup>Simplex      <sup>۲</sup>anti-cycling

مثال ۲-۵-۳. مثال ۲-۵-۱ را با تابع هدف کمینه سازی نرم  $l_1$  در نظر بگیرید :

$$\min z = \|s\|_1 = \sum_{i=1}^5 |s_i| \quad (32-2)$$

$$s.t \quad s_1 + s_2 + 4s_3 - s_4 + 3s_5 = 1 \quad (33-2)$$

$$s_1 - s_2 - 2s_3 + 3s_4 + s_5 = 1 \quad (34-2)$$

پس از اعمال تغییر متغیر و حل مسئله به روش سیمپلکس جواب مسئله عبارت است از :

$$(nonbasicvariables : s_1 = s_2 = s_3 = 0 \Rightarrow basic : s_4 = \frac{1}{5}, s_5 = \frac{2}{5}, z = \frac{3}{5}) \quad (35-2)$$

که همان جواب پایه ای ۲-۲۵ از مسئله ۲-۱۳ است .

اگر چه جواب مسئله فوق با مسئله ۲-۱۳ (که در ابعاد پایین مطرح شده) یکسان نیست ، اما در [۳] نشان داده شده است هنگامی که ابعاد مسئله بزرگ می شود ، برای بیشتر دستگاه های فرو معین جواب مسئله  $P_1$  و  $P_0$  برهم منطبق می شود ؛ به عبارتی مینیمم سازی نرم  $l_1$  منجر به مینیمم سازی نرم  $l_0$  می شود .

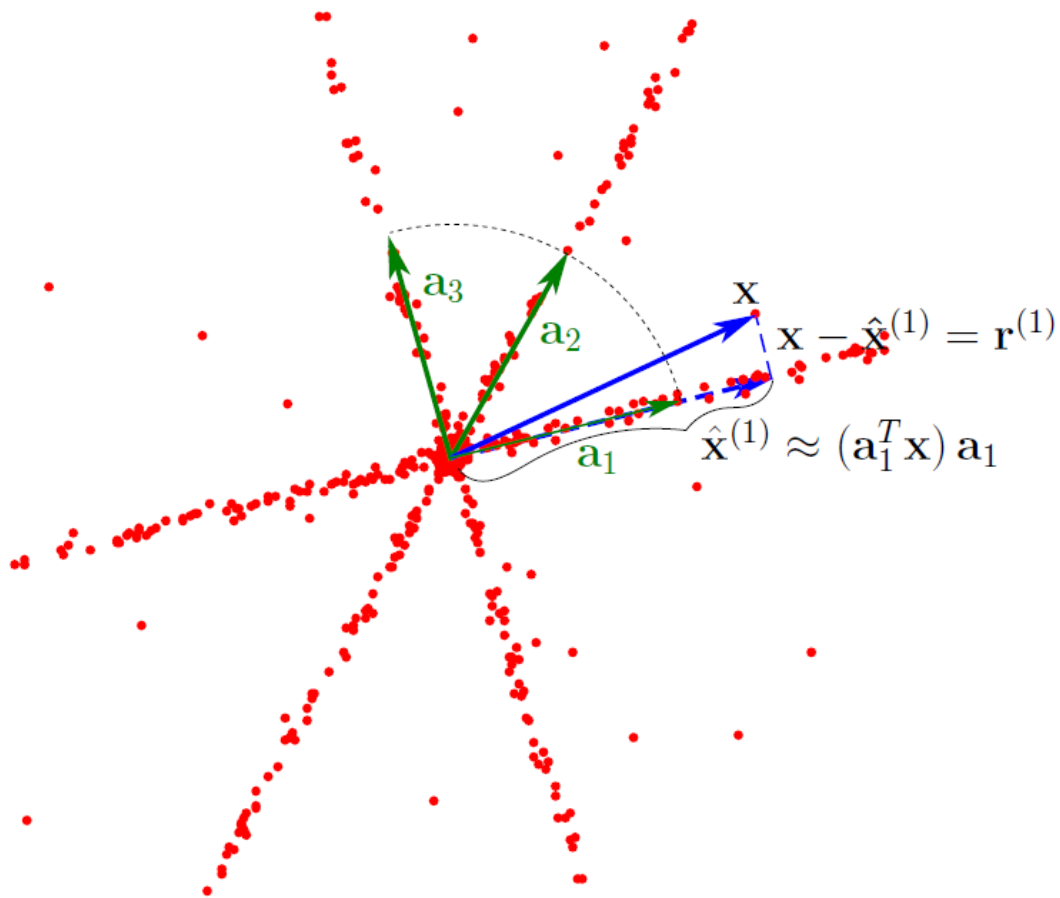
## روش های حریصانه

### ۲-۵-۳ روش جستجوی تطابقی

روش MP ابتدا توسط Zhang و Mallat به عنوان یک روش تجزیه اتمی مطرح شد [۱۲] . ایده ی اصلی این الگوریتم مبتنی بر تقریب مرحله به مرحله ی بردار مخلوط با اتم های  $\{a_i\}$  است . برای درک بهتر ایده ی اصلی این روش ، حالت خاص  $m = 3$  و  $n = 2$  را در نظر بگیرید .

شکل ۲-۳ یک نمونه توزیع بردار مخلوط را که از بردارهای منبع تنک ناشی شده است ، نشان می دهد [۱۹] . توجه کنید که بردار های مخلوط و اتم های  $\{a_i\}$  در مسئله حاضر دوبعدی هستند و فضای  $x$  همان  $R^2$  است و اتم ها بردار های یکه هستند . همچنین توزیع بردارهای مخلوط ( $x$  ها) حول بردارهای یکه  $\{a_i\}$  متمرکز شده اند . ایده ی اصلی این است که در مرحله اول بردار  $x$  را با نزدیک ترین بردار  $\{a_i\}$  تقریب بزینم و این هدف به این گونه حاصل می شود که اگر فاصله (عمودی) بردار  $x$  تا خطی که توسط  $a_i$  مشخص می شود ، کمترین مقدار را داشته باشد ( از میان تمام  $\{a_i\}$  ها ) .





شکل ۲-۳: ایده ی اصلی روش MP: انتخاب اتمی ( $a_i$ ) که نزدیک ترین فاصله (عمودی) را تا بردار  $x$  دارد که در اینجا بردار  $a_1$  است. در مرحله بعد این عمل برای بردار باقیمانده  $r^{(1)}$  تکرار می شود و به همین ترتیب.

به بیان دقیق تر، بردار مخلوط  $x$  دارای بسط دقیق  $x = \sum_{i=1}^3 s_i a_i$  بر حسب بردارهای  $\{a_i\}$  است. حال هدف ما، به دست آوردن تقریبی از  $x$  بر حسب یکی از بردارهای  $\{a_i\}$  است  $(x \approx s_i a_i)$ ؛ به گونه ای که خطا، یعنی  $\|x - s_i a_i\|_2$ ، روی تمام  $i$ ها و  $s_i$ ها مینیمم شود.

در این صورت  $s_i a_i$  تصویر بردار  $x$  روی زیرفضای تولید شده توسط  $a_i$  (که در این شکل یک خط است) خواهد بود و مقدار خطای تقریب برابر با  $\|x - s_i a_i\|_2$  است (و  $s_i = \frac{a_i^T x}{\|a_i\|_2^2}$  که با توجه به این که اتم ها را بردارهای یکه در نظر گرفتیم، می توانیم از مقدار مخرج صرف نظر کنیم). حال برای یافتن بهترین تقریب کافی است که خطای مذکور را برای تمام  $i$ ها مینیمم، و یا به طور معادل، قدر مطلق تصویر  $|a_i^T x|$  را ماکزیمم کنیم. کمیت  $|a_i^T x|$  تحت عنوان همبستگی<sup>۱</sup> میان دو بردار  $a_i$  و  $x$  شناخته می شود. با این تعریف، روش MP در مرحله ی اول اتمی را انتخاب می کند که بیشترین همبستگی را با بردار  $x$  داشته باشد. اگر اندیس این اتم را با  $i_0$  نشان دهیم، تقریب مرحله ی اول  $\hat{x}^{(1)} = \frac{(a_{i_0}^T x)}{\|a_{i_0}\|_2^2} a_{i_0}$  و باقیمانده ی<sup>۲</sup> مرحله اول  $r^{(1)} = x - \hat{x}^{(1)}$  هستند و همچنین  $\hat{s}_{i_0} = \frac{(a_{i_0}^T x)}{\|a_{i_0}\|_2^2}$  نیز تخمینی از مولفه  $i_0$ -ام منبع  $s$  است. حال اگر باقی مانده ناچیز باشد، الگوریتم پایان می یابد؛ در غیر این صورت فرآیند بالا برای باقی مانده تکرار می شود. به طور مثال در شکل ۲-۳ به نظر می رسد که تقریب مرحله ی اول بر حسب  $a_1$  کافی نیست و لازم است الگوریتم یک تکرار دیگر اجرا شود.

---

<sup>۱</sup>correlation      <sup>۲</sup>residual

الگوریتم ۱-۲ جست و جوی تطابقی با ورودی‌های ماتریس  $A$  و بردار  $x$  و حد آستانه  $\varepsilon$ .

**Require:** We are given the matrix  $A$ , the vector  $x$ , and the error threshold  $\varepsilon_0$  for approx-

imating the solution of  $(P0) : \min_s \|s\|_0$  subject to  $x = As$

**Ensure:** The proposed solution is  $s^k$  obtained after  $k$  iterations

// Initialize

- 1:  $k = 0$ ,
- 2: The initial solution  $s^0 = 0$ ,
- 3: The initial residual  $r^0 = x - As^0 = x$
- 4: The initial solution support  $S^0 = \text{Support}\{s^0\} = 0$

// Main Iteration

- 5: Increment  $k$  by 1 and perform the following steps:

//Sweep

- 6: Compute the errors  $\varepsilon(j) = \min_{z_j} \|a_j z_j - r^{k-1}\|_2^2$  for all  $j$  using the optimal choice

$$z_j^* = a_j^T r^{k-1} / \|a_j\|_2^2$$

//Update Support

- 7: Find a minimizer,  $j_0$  of  $\varepsilon(j) : \forall 1 \leq j \leq m, \varepsilon(j_0) \leq \varepsilon(j)$ , and update  $S^k = S^{k-1} \cup \{j_0\}$ .

//Update Provisional Solution

- 8: Set  $s^k = s^{k-1}$ , and update the entry  $s^k(j_0) = s^{k-1}(j_0) + z_j^*$

//Update Residual

- 9: Compute  $r^k = x - As^k = r^{k-1} - z_{j_0}^* a_{j_0}$

//Stopping Rule

- 10: If  $\|r^k\|_2 < \varepsilon_0$ , stop. Otherwise, apply another iteration.

مثال ۲-۵-۴. دستگاه  $x = As$  مثال ۲-۵-۱ را در نظر بگیرید که در آن

$$A = \begin{pmatrix} 1 & 1 & 4 & -1 & 3 \\ 1 & -1 & -2 & 3 & 1 \end{pmatrix}, \quad x = \begin{pmatrix} 1 \\ 1 \end{pmatrix}$$

ابتدا قرار می دهیم  $k = 0$  و همچنین :

$$r^{(0)} = x = \begin{pmatrix} 1 \\ 1 \end{pmatrix}, s^{(0)} = \begin{pmatrix} 0 & 0 & 0 & 0 & 0 \end{pmatrix}^T, S^0 = \{\}$$

(به عبارتی مجموعه  $S$  مجموعه اندیس منابع فعال است ؛ یعنی اندیس مولفه هایی از بردار  $s$  که در مرحله نهایی مقدار غیرصفر دارند.)

• تکرار اول:

برای هر  $1 \leq j \leq 5$  مقدار  $z_j = \frac{a_j^T x}{\|a_j\|_2}$  را به دست می آوریم

$$\begin{pmatrix} z_1 & z_2 & z_3 & z_4 & z_5 \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0.1 & 0.2 & 0.4 \end{pmatrix}$$

و از آن ها برای محاسبه بردار های (باقیمانده)  $r^{(0)} - z_j a_j$  استفاده می کنیم :

$$r^{(0)} - z_1 a_1 = \begin{pmatrix} 1 \\ 1 \end{pmatrix} - 1 \begin{pmatrix} 1 \\ 1 \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \end{pmatrix}$$

$$r^{(0)} - z_2 a_2 = \begin{pmatrix} 1 \\ 1 \end{pmatrix}$$

$$r^{(0)} - z_3 a_3 = \begin{pmatrix} 0.6 \\ 1.2 \end{pmatrix}$$

$$r^{(0)} - z_4 a_4 = \begin{pmatrix} 1.2 \\ 0.4 \end{pmatrix}$$

$$r^{(0)} - z_5 a_5 = \begin{pmatrix} -0.2 \\ 0.6 \end{pmatrix}$$

با توجه به روابط بالا واضح است که  $\{1\} = \arg \min_j \|r^{(0)} - a_j z_j\|_2 = j$  بنابراین داریم

$s_1^{(1)} = s_1^{(0)} + z_1 = 1$  و همین طور :

$$r^{(1)} = r^{(0)} - z_{j_0} a_{j_0} = \begin{pmatrix} \circ \\ \circ \\ \circ \end{pmatrix}, s^{(1)} = \begin{pmatrix} 1 & \circ & \circ & \circ & \circ \end{pmatrix}^T \quad S^0 = \{1\}$$

با توجه به این که  $\|r^{(1)}\|_2 = 0$ ، لذا الگوریتم در همین جا متوقف می شود. ملاحظه می شود که جواب بهینه به دست آمده، همان جواب پایه ای ۲-۱۶ در مثال ۲-۵-۱ است.

الگوریتم جستجوی تطابقی یک الگوریتم حریصانه است، با این ویژگی که در هر مرحله بهترین انتخاب ممکن را (با توجه به وضعیت موجود) انجام می دهد. لذا می توان این الگوریتم را به طور محلی بهینه دانست<sup>۱</sup>. در این نوع الگوریتم ها انتخاب فعلی وابسته به انتخاب قبلی است، اما قابلیت تغییر وضعیت انتخاب های قبلی را ندارند. به عنوان مثال، اگر یک اتم به خصوص در یک مرحله به عنوان بهترین اتم انتخاب شود، هیچ یک از اتم های بعدی نمی تواند اتم مذکور را از مجموعه اتم هایی که برای تخمین بردار  $x$  به کار می روند، کنار بگذارد؛ لذا می توان مثال هایی یافت که در آن ها الگوریتم بسیار بد عمل می کند. به عنوان نمونه، Temlyakov و Devove در [۲۰] مسئله ای را مطرح کرده که در آن "جستجوی تطابقی" هیچ گاه متوقف نمی شود. اما تعمیمی<sup>۲</sup> از الگوریتم جستجوی تطابقی تحت عنوان "جستجوی تطابقی متعامد" وجود دارد که در مثال مذکور به مشکل بر نمی خورد.

## ۲-۵-۴ روش جستجوی تطابقی متعامد

در روش OMP (مطرح شده توسط Rezaiifar, Pati و Krishnaprasad در سال ۱۹۹۳) [۱۳]، که تعمیم یافته روش MP است، در مرحله  $k$ -ام پس از مشخص شدن بهترین اتم ( $a_{i_k}$ ) یک بار  $x$  را روی زیرفضای متشکل از تمام اتم های انتخاب شده تا کنون تصویر می کنیم؛ ضرایب این تصویر (یا بسط) را به عنوان تخمین  $k$ -ام بردار  $s$  در نظر گرفته و خطای تخمین را به عنوان باقی مانده برای مرحله  $i$  بعد مورد استفاده قرار می دهیم؛ یعنی در هر مرحله بعد از مشخص شدن اتم جدید، از تمام ضرایب قبلی صرف نظر شده و ترکیب خطی (جدید) از تمام اتم های انتخاب شده تا این مرحله، محاسبه می شود که کمترین خطا را برای بازسازی سیگنال آزمون داشته باشد.

مقدار باقی مانده در هر مرحله بر تمام اتم های انتخاب شده تا آن مرحله عمود خواهد بود و لذا الگوریتم جستجوی تطابقی "متعامد" شناخته می شود.

<sup>۱</sup>locally optimum

<sup>۲</sup>refinement

---

الگوریتم ۲-۲ جست و جوی تطابقی متعامد با ورودی‌های ماتریس  $A$  و بردار  $x$  و حد آستانه  $\varepsilon_0$ .

---

**Require:** We are given the matrix  $A$ , the vector  $x$ , and the error threshold  $\varepsilon_0$  for approx-

imating the solution of  $(P0) : \min_s \|s\|_0$  subject to  $x = As$

**Ensure:** The proposed solution is  $s^k$  obtained after  $k$  iterations

// Initialize

1:  $k = 0$ , and set

2: The initial solution  $s^0 = 0$ ,

3: The initial residual  $r^0 = x - As^0 = x$

4: The initial solution support  $S^0 = \text{Support}\{s^0\} = 0$

//Main Iteration

5: Increment  $k$  by 1 and perform the following steps:

//Sweep

6: Compute the errors  $\varepsilon(j) = \min_{z_j} \|a_j z_j - r^{k-1}\|_2^2$  for all  $j$  using the optimal choice

$$z_j^* = a_j^T r^{k-1} / \|a_j\|_2^2$$

//Update Support

7: Find a minimizer,  $j_0$  of  $\varepsilon(j) : \forall j \notin S^{k-1}, \varepsilon(j_0) \leq \varepsilon(j)$ , and update  $S^k = S^{k-1} \cup \{j_0\}$ .

//Update Provisional Solution

8: Compute  $s^k$ , the minimizer of  $\|x - As\|_2^2$  subject to  $\text{Support}\{s\} = S^k$

//Update Residual:

9: Compute  $r^k = x - As^k$

//Stopping Rule

10: If  $\|r^k\|_2 < \varepsilon_0$ , stop. Otherwise, apply another iteration.

---

## ۵-۵-۲ الگوریتم CoSaMP

در الگوریتم‌های حریمانه مبتنی بر انتخاب یک اتم، مانند دو الگوریتم MP و OMP، اتمی که در هر گام انتخاب می‌شود، به طور قطع تا پایان الگوریتم در نمایش تنک حضور دارد. به عبارت دیگر، اگر این الگوریتم‌ها در انتخاب یک اتم اشتباه کنند، اثر آن تا پایان الگوریتم وجود داشته و در نتیجه منجر به جواب‌های نادرست

می شود. احتمال بروز این اشتباه به خصوص زمانی که همبستگی (شبهات) بین اتم های دیکشنری زیاد است ، بالا می رود. یک راه حل برای غلبه بر این مشکل ، انتخاب چندین اتم در گام انتخاب بهترین اتم ها ، به عنوان کاندید حضور در نمایش تنک سیگنال ، و سپس استفاده از تعدادی از این اتم ها برای به روز کردن نمایش سیگنال است [۲۱]. این اساس کار الگوریتم CoSaMP<sup>۱</sup> است که پشتوانه تئوری خوبی هم دارد [۲۲]. این الگوریتم یک نمایش  $K$ -تنک ، یعنی حداکثر با  $K$  مولفه غیرصفر ، از سیگنال به دست می دهد. به عبارت دیگر  $K$  باید معلوم باشد ؛ هر چند ، همانطور که در [۲۲] پیشنهاد شده ، می توان از رابطه  $K \leq \frac{n}{\sqrt{\log(1+m/K)}}$  به عنوان تخمینی از  $K$  استفاده کرد.

---

<sup>۱</sup>compressive sampling matching pursuit

---

الگوریتم ۲-۳ جست و جوی با CoSaMP ورودی‌های ماتریس  $A$  و بردار  $x$  و اسکالر  $K$  و حد آستانه  $\varepsilon_0$ .

---

**Require:** We are given the matrix  $A$ , the vector  $x$ , the scalar  $K$ , and the error threshold  $\varepsilon_0$

for approximating the  $K$ -sparse solution of  $(P0) : \min_s \|s\|_0$  subject to  $x = As$

**Ensure:** The proposed solution is  $s^k$  obtained after  $k$  iterations

// Initialize

1:  $k = 0$ , and set

2: The initial solution  $s^0 = 0$ ,

3: The initial residual  $r^0 = x - As^0 = x$

4: The initial solution support  $S^0 = \text{Support}\{s^0\} = 0$

//Main Iteration

5: Increment  $k$  by 1 and perform the following steps:

//Sweep

6: Compute the projection vector  $z = A^T r^{k-1}$

//Identify Large Components

7: Compute the best  $2K$  support set of the projection vector  $\Omega = \text{support}\{z_{2K}\}$

//Update Support

8: Merge the strongest support sets,  $T = S^k = S^{k-1} \cup \Omega$

//Estimate Of Solution

9: Perform a Least-Squares estimation of solution,  $b_{|T} = A_{|T}^\dagger x$ , and  $b_{|T^c} = 0$

//Update Provisional Solution

10: Compute  $s^k = b_K$

//Update Residual:

11: Compute  $r^k = x - As^k$

//Stopping Rule

12: If  $\|r^k\|_2 < \varepsilon_0$ , stop. Otherwise, apply another iteration.

---



## فصل ۳

# روش جستجوی زیر فضا

### ۱-۳ مقدمه

هدف ما در این بخش توضیح روش جستجوی زیرفضاست که یکی از روش های یافتن تنک ترین جواب دستگاه معادلات خطی فرومعیین است. برای این منظور، با ارائه یک تعبیر هندسی از تنک بودن، فرم کلی این روش شرح داده خواهد شد. اساس کار این روش یافتن نزدیک ترین زیر فضا از میان تمام زیرفضاهای  $p$ -بعدی مختصات (از فضای  $R^m$ ) به مجموعه دستگاه  $x = As$  است که به صورت جست و جوی ترکیبیاتی یا کامل صورت می گیرد و سپس تصویر این مجموعه جواب روی زیرفضای مذکور را به عنوان تخمینی از جواب های تنک مسئله به دست می دهد.

مسئله یافتن فاصله دستگاه  $x = As$  از یک زیرفضای  $p$ -بعدی مختصات را می توان به صورت یک مسئله بهینه سازی مطرح کرد و جواب صریحی برای آن یافت. اما یافتن بهترین زیرفضا، همان گونه که گفته شد، نیاز به جست و جوی ترکیبیاتی در میان  $\binom{m}{p}$  حالت ممکن دارد. توجه کنید که بعد  $p$  مورد نیاز، بستگی به تعداد منابع فعال بردار منبع اصلی دارد. با توجه به ویژگی های گفته شده (نیاز به جست و جوی ترکیبیاتی) انتظار می رود پیاده سازی این الگوریتم برای مسائل با ابعاد بالا مقرون به صرفه نباشد. در ... با استفاده از نتایج تجربی عملکرد الگوریتم مورد بررسی قرار گرفته و نشان داده شده است که جز برای مقادیر ناچیز  $p$ ، پیاده سازی الگوریتم مقرون به صرفه نیست. به بیان دیگر، الگوریتم با ساختار گفته شده تنها برای حل مسائل بسیار تنک و یا در ابعاد پایین مناسب است.

## ۲-۳ تفسیر هندسی روش جست و جوی زیرفضا

اساس کار روش جست و جوی زیرفضا بر پایه یک تعبیر هندسی از تنک بودن است. برای توضیح این تعبیر از مدل (ساده) مخلوط گوسی برای منابع تنک استفاده خواهد شد. در این مدل، مولفه های بردار منبع با توزیع یکسان مخلوط گوسی فرض می شوند. به طور خلاصه می توان نوشت:

$$s_i \sim \pi_0 \mathcal{N}(0, \sigma_0^2) + \pi_1 \mathcal{N}(0, \sigma_1^2) \quad 1 \leq i \leq m$$

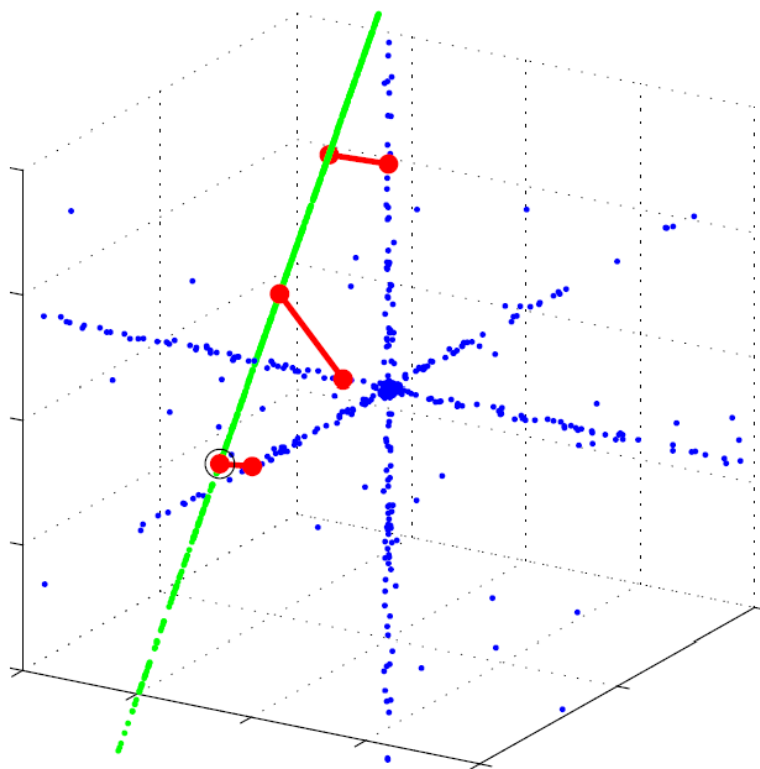
توضیح این مدل، با معرفی متغیرهای تصادفی برنولی  $z_i$ ، متناظر با هر منبع  $s_i$ ، بسیار ساده تر است. این متغیرها را در اصطلاح متغیرهای پنهان<sup>۱</sup> می نامند. هر  $z_i$  یکی از دو مقدار ۰ یا ۱ را، به ترتیب با احتمال های  $\pi_0$  و  $\pi_1 = 1 - \pi_0$  می پذیرد. وابسته به مقدار انتخاب شده توسط  $z_i$  وضعیت "فعال بودن" منبع مربوطه مشخص می شود. اگر  $z_i = 1$  باشد، منبع مربوطه "فعال" است و با یک توزیع گوسی با میانگین صفر و واریانس  $\sigma_1$  مدل می شود؛ در غیر این صورت (اگر  $z_i = 0$ )، منبع مذکور "غیر فعال" بوده و با یک توزیع گوسی با میانگین صفر و واریانس  $\sigma_0 \ll \sigma_1$  مدل می شود. شرط  $\sigma_0 \ll \sigma_1$  باعث می شود که دامنه یک منبع غیرفعال (غالباً) بسیار کمتر از یک منبع فعال باشد. حال با اضافه کردن شرط  $\pi_0 \gg \pi_1$ ، می توان تنک بودن بردار منبع حاصل را تضمین کرد. این شرط به این معناست که احتمال فعال بودن یک منبع، بسیار کمتر از احتمال غیر فعال بودن آن است. به طور خلاصه، در این مدل یک مولفه بردار منبع (یعنی بردار  $s$ ) با احتمال  $\lesssim 1$  غیرفعال، یعنی دارای مقدار ناچیز، و با احتمال  $\ll 1$  فعال است. با این شرایط خواهیم دید که توزیع نمونه ای بردار منبعی که از چنین مدلی به دست آمده، به این گونه خواهد بود که تجمع بیشتر نقاط حول مبدا مختصات، پس از آن بیشترین تمرکز حول محورهای مختصات و سپس صفحات مختصات و به همین ترتیب زیرفضاهای با ابعاد بالاتر می رسد.

این مطلب را می توان چنین توجیه نمود؛ احتمال قرار داشتن بردار منبع  $s$  حول مبدا مختصات برابر است با  $\pi_0^m$ . همینطور احتمال قرار داشتن بردار منبع  $s$  حول یک زیرفضای  $p$ -بعدی مختصات تقریباً برابر با  $\mathcal{B}(\pi_0; p, m) \triangleq \binom{m}{p} \pi_0^{m-p} (1 - \pi_0)^p$  خواهد بود. با توجه به این که  $\pi_0 \lesssim 1$  است، این احتمال ها غالباً به صورت زیر مرتب می شوند:

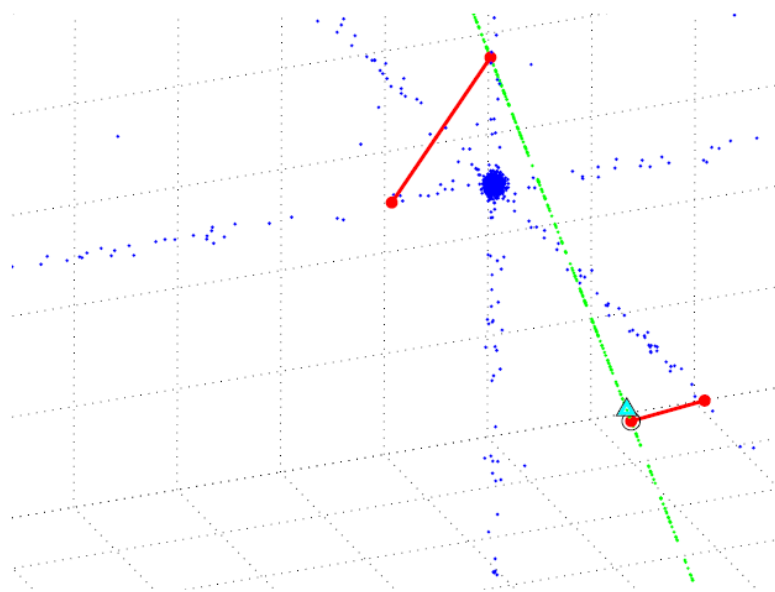
$$\pi_0^m > \binom{m}{1} \pi_0^{m-1} (1 - \pi_0)^1 > \binom{m}{2} \pi_0^{m-2} (1 - \pi_0)^2 > \dots > (1 - \pi_0)^m$$

با این تفاسیر تجمع نقاط به شکلی که گفتیم خواهد بود.

<sup>۱</sup>hidden variables



(ا)



(ب)

شکل ۱-۳: تعبیر هندسی روش جست و جوی زیرفضا در حالت  $m = 3$  و  $n = 2$ . آروش SP جوابی از دستگاه  $x = As$  (که در اینجا یک خط است) را به عنوان تنگ ترین جواب انتخاب می کند که نزدیک ترین فاصله را تا توزیع  $s$  (در اینجا محورهای مختصات) را داشته باشد. (ب) همان شکل از نمایی دیگر است و همچنین جواب LP نیز با یک مثلث مشخص شده است.

برای درک بهتر این موضوع، حالت خاص  $m = 3$  و  $n = 1$  و  $\pi_0 = 0/9$  را در نظر بگیرید. شکل ۱-۳ توزیع نمونه ای بردار منبع را در فضای  $s$  (یعنی  $R^3$ ) نشان می دهد. همان طور که مشاهده می کنید تجمع بیشتر نقاط حول مبدا مختصات و سپس حول محورهاست؛ به علاوه تجمع حول صفحات مختصات به مقداری ناچیز است که قابل مشاهده نیست.

همچنین علاوه بر توزیع  $s$ ، دستگاه  $x = As$  نیز در شکل با خط نشان داده شده است. توجه کنید که دستگاه  $x = As$  برای  $m = 3$  و  $n = 2$  (با فرض رتبه کامل بودن ماتریس  $A$ ) یک خط را در فضای سه بعدی نشان می دهد.

می توان گفت مسئله یافتن تنک ترین جواب دستگاه، یافتن نقطه ای روی این خط است که کمترین فاصله را تا توزیع نمونه ای بردار  $s$  دارد. به عبارت دیگر؛ می توان به طور شهودی برای این مثال این طور گفت که یافتن نقطه ای از دستگاه  $x = As$  که کمترین فاصله را تا محورهای مختصات دارد منجر به یافتن تنک ترین جواب با دقت خوبی خواهد شد.

البته توجه داشته باشید که مقدار دقیق نقاط حول مبدا اهمیت چندانی ندارد و شکل اصلی توزیع توسط نقاطی تعیین می شود که حول محورها تمرکز یافته اند. همان طور که در شکل ۱-۳ (آ) مشاهده می کنید، سه نقطه ای که هر کدام دارای کمترین فاصله تا یکی از محورهای مختصات دارند، مشخص شده اند. بین این سه نقطه، نقطه ای که دارای کمترین فاصله است تخمین مدنظر روش جست و جوی زیرفضا برای بردار منبع خواهد بود. این نقطه در شکل توسط یک دایره علامت گذاری شده است. به علاوه در شکل ۱-۳ (ب) این نقطه از نمایی دیگر نشان داده شده است. همچنین جواب حاصل از برنامه ریزی خطی با یک مثلث نشان داده شده است. همان طور که ملاحظه می شود، این دو جواب بسیار به یکدیگر نزدیک هستند. توجه کنید با تغییر بردار  $x$  خطی که نشانگر دستگاه  $x = As$  است به موازات (خود) جابجا می شود؛ لذا نقطه دارای مینیمم فاصله از محور ها تغییر می کند، در نتیجه تخمین ما از بردار منبع هم تغییر خواهد کرد.

### ۳-۳ فرمول بندی در قالب یک مسئله بهینه سازی

در این بخش مسئله یافتن تنک ترین جواب دستگاه  $x = As$  با تعبیر هندسی فوق در قالب یک مسئله بهینه سازی مطرح خواهد شد. برای شروع، مثال مذکور (۱-۳) را در نظر بگیرید. برای یافتن مجذور فاصله یک نقطه در فضای  $s$  تا محور مختصات  $i$ -ام کافی است مجذور مولفه های آن را، غیر از مولفه  $i$ -ام، با یکدیگر جمع کنیم. به عنوان مثال، مجذور فاصله از محور  $s_1$  را در فضای  $R^3$  می توان با  $s_2^2 + s_3^2$  نشان داد. بنابراین مسئله

یافتن جوابی از دستگاه که در مینیمم فاصله از محور  $s_1$  باشد، به شکل زیر قابل بیان است:

$$\min s_1^{\downarrow} + s_1^{\uparrow} \quad s.t. \quad x = As \quad (1-3)$$

تعریف می کنیم:

$$s_{23} \triangleq [s_2 \ s_3] \quad A_{23} \triangleq [a_2 \ a_3]$$

با توجه به این که  $x = s_1 a_1 + A_{23} s_{23}$  و  $\|s_{23}\|_2^2 = s_2^2 + s_3^2$  می توان مسئله ۱-۳ را به صورت مسئله بدون محدودیت<sup>۱</sup> زیر نوشت:

$$\min_{s_1} \|A_{23}^{-1}(x - s_1 a_1)\|_2^2 \quad (2-3)$$

اگر در بردار منبع اصلی تنها مولفه  $s_1$  فعال باشد، انتظار می رود کمینه سازی  $\|s_{23}\|_2^2$  جواب را با دقت خوبی به دست دهد. تابع هزینه فوق را می توان با آنچه روش "جست و جوی تطابق" در چنین حالتی (در گام ششم الگوریتم ۱-۲) مینیمم می کند، یعنی  $\|x - s_1 a_1\|$  مقایسه کرد. توجه کنید که در روش حاضر به جای تصویر کردن  $x$  در امتداد  $a_1$ ، بردار  $A_{23}^{-1}x$  را در امتداد  $A_{23}^{-1}a_1$  تصویر می کند. برای یافتن جوابی از  $x = As$  که کمترین فاصله را از محورها دارد؛ به عبارت دیگر تنگ ترین جواب دستگاه  $x = As$ ؛ کافی است مشابه مسئله بهینه سازی ۲-۳ را برای دو محور دیگر حل کرده و از میان جواب های حاصل، جوابی که دارای کمترین فاصله از محور مربوطه است، انتخاب شود. به طور خلاصه، برای حالت  $m = 3$  می توان نوشت:

$$\min_{i \neq j} \min_{x=As} (s_i^{\downarrow} + s_j^{\uparrow}) \quad (3-3)$$

با تعمیم این رابطه می توان روش "جست و جوی زیرفضا" را در حالت کلی به شکل زیر بیان کرد:

$$\arg \min_k \min_{\mathcal{I} \in \mathcal{I}_k} \min_{x=As} \sum_{i \in \mathcal{I}} s_i^{\downarrow} \quad (4-3)$$

در عبارت فوق  $\mathcal{I}_k$  خانواده تمام زیرمجموعه های  $k$ -عضوی مجموعه  $\mathbb{Z}_m^+ = \{1, 2, \dots, m\}$  است. به این ترتیب، به ازای هر عضو این خانواده مانند  $\mathcal{I} \in \mathcal{I}_k$ ، عبارت  $\sum_{i \in \mathcal{I}} s_i^{\downarrow}$  فاصله تائیکی از زیرفضاهای  $p \triangleq m - k$ -بعدی مختصات را در فضای  $R^m$  نشان می دهد (توجه کنید  $k$  تعداد مولفه های غیرفعال و  $p$  تعداد مولفه های

<sup>۱</sup>unconstrained

فعال بردار منبع خواهد بود). به طور مثال، برای  $m = 3$  و  $k = 2$  داریم  $\mathcal{I}_2 = \{\{1, 2\}, \{1, 3\}, \{2, 3\}\}$  و به ازای هر  $\mathcal{I} \in \mathcal{I}_2$ ، تابع هزینه  $3-4$  فاصله تا یکی از محورها (زیر فضاهای  $(= p)$  یک-بعدی) را به دست می دهد.

برای بخش اول مسئله بهینه سازی  $3-4$  یعنی مینیمم سازی تابع هزینه برای یک  $\mathcal{I}$  مشخص روی  $x = As$ ، یا به عبارت دیگر مینیمم کردن فاصله تا یک زیر فضای  $p$ -بعدی به خصوص؛ می توان پاسخ صریحی یافت. اما بخش دوم مسئله که نام روش نیز مربوط به آن است، یعنی جستجو روی تمام زیر فضاهای  $p$ -بعدی برای یافتن زیر فضای بهینه، به شکل جستجوی ترکیبیاتی انجام می شود؛ به این معنی که تمام  $\binom{m}{p}$  حالت ممکن را بررسی می کنیم؛ و مشکل اصلی این روش به همین بخش بازمی گردد.

فرم کلی روش جست و جوی زیر فضا را به صورت زیر نیز می توان نوشت:

$$\arg \min_k \min_{W \in \mathcal{W}_k} \min_{x=As} s^T W s \quad (5-3)$$

که در آن  $\mathcal{W}_k$  خانواده تمام ماتریس های وزن قطری است که دارای  $k$  عنصر یک و  $m - k$  عنصر صفر روی قطری اصلی اند.

## 4-3 جواب مینیمم فاصله تا یک زیر فضا

هدف ما در این بخش یافتن پاسخ مسئله بهینه سازی زیر است:

$$\min \sum_{i=1}^k s_i \quad s.t. \quad x = As \quad (6-3)$$

تابع هزینه فوق همان تابع هزینه به کار رفته در  $3-4$  به ازای  $\mathcal{I} = \{1, 2, \dots, k\}$  است. توجه کنید این انتخاب برای راحتی در محاسبات انجام شده و چیزی از کلیت مسئله کم نمی کند. همچنین از جواب به دست آمده در این بخش برای گام تخمین روش "آشکارسازی-تخمین پیاپی" در فصل بعد مورد استفاده قرار خواهد گرفت.

برای به دست آوردن جواب مسئله  $3-6$  از دو نتیجه معروف در جبر خطی استفاده خواهیم کرد. دستگاه  $x = As$  را در نظر بگیرید که در آن ماتریس  $A$ ،  $n \times m$  (ورته کامل) است. در حالت کلی، برای یک دستگاه "فرامعین" ( $n > m$ ) جواب دقیقی وجود ندارد. در این حالت جواب LS، یعنی بردار  $x$  ای که  $\|b - Ax\|_2$  را

مینیمم می کند ، عبارت است از

$$x^\dagger = A^\dagger b = (A^T A)^{-1} A^T b \quad (7-3)$$

از طرف دیگر ، یک دستگاه "فرومعیّن" ( $n < m$ ) در حالت کلی بی شمار جواب دارد. جواب دارای مینیمم نرم  $l_2$  ی آن را می توان به شکل زیر نوشت:

$$x^\dagger = A^\dagger b = A^T (A A^T)^{-1} b \quad (8-3)$$

ماتریس  $A^\dagger$  ، شبه وارون ماتریس  $A$  است که با توجه به این که  $n$  بزرگتر  $m$  است یا کوچکتر به دو صورت متفاوت بر حسب  $A$  نوشته می شود. برای  $n > m$  ، ماتریس  $A$  دارای معکوس چپ<sup>۱</sup> به صورت  $(A^T A)^{-1} A^T$  (به شرط آن که ماتریس  $A$  رتبه کامل ستونی باشد) و برای  $n < m$  ماتریس  $A$  دارای معکوس راست<sup>۲</sup> به صورت  $A^T (A A^T)^{-1}$  می باشد (به شرط رتبه کامل سطری بودن  $A$ ). صحیح بودن روابط ۷-۳ و ۸-۳ در دو بخش بعدی با استفاده از حل مسئله بهینه سازی نشان داده می شود.

حال مسئله ۶-۳ را در نظر بگیرید. بردار  $s$  و ماتریس  $A$  را متناظر با تابع هدف مسئله ۶-۳ افراز می کنیم.

داریم:

$$s = [s_l \ s_\alpha]^T \quad A = [A_l \ A_\alpha]$$

که در آن  $A_l$  ،  $A_\alpha$  ،  $s_l$  و  $s_\alpha$  به ترتیب  $n \times k$  ،  $n \times (m - k)$  ،  $k \times 1$  و  $(m - k) \times 1$  هستند. اندیس های  $\alpha$  و  $l$  به ترتیب بخش فعال و غیر فعال بردار (یا ماتریس) را نشان می دهند. لذا با این افراز مسئله ۶-۳ را می توان بر حسب متغیرهای جدید به صورت زیر بازنویسی کرد:

$$\min \|s_l\|_2^2 \quad A_l s_l = x - A_\alpha s_\alpha \quad (9-3)$$

فرض می کنیم  $n \leq k$  که این فرض برای آن است که دستگاه  $A_l s_l = x - A_\alpha s_\alpha$  بر حسب  $s_l$  دارای جواب باشد؛ در این صورت دستگاه  $A_l s_l = x - A_\alpha s_\alpha$  بر حسب  $s_l$  ، یک دستگاه فرومعیّن خواهد بود و لذا با توجه به رابطه ۸-۳ و وابسته نبودن تابع هزینه ۹-۳ به  $s_\alpha$  جواب مسئله ۹-۳ به صورت زیر خواهد بود:

$$s_l^*(s_\alpha) = A_l^T (A_l A_l^T)^{-1} (x - A_\alpha s_\alpha) \quad (10-3)$$

---

<sup>۱</sup>left inverse      <sup>۲</sup>right inverse

علامت \* نشان دهنده جواب بهینه است. جواب بهینه ۳-۹ به شکل پارامتری بر حسب  $s_\alpha$  بیان شده است. تعریف می کنیم  $P \triangleq (A_l A_l^T)^{-1}$ . عبارت ۳-۱۱ را می توان به صورت زیر نوشت:

$$s_l^*(s_\alpha) = A_l^T P x - (A_l P A_\alpha) s_\alpha \triangleq c - B s_\alpha \quad (۱۱-۳)$$

با استفاده از نمادگذاری های جدید مسئله بهینه سازی اصلی را می توان به صورت مسئله بدون قید زیر نوشت:

$$\min \|s_l^*(s_\alpha)\|_2^2 = \|c - B s_\alpha\|_2^2 \quad (۱۲-۳)$$

که در آن  $c$  و  $B$  به ترتیب  $1 \times k$  و  $(m-k) \times k$  هستند. برای یکتایی جواب بهینه فرض می کنیم  $k > m - n$  تا دستگاه  $B s_\alpha = c$  یک دستگاه فرامعین باشد. لذا می توان با استفاده از رابطه ۳-۷ در مورد جواب LS یک دستگاه فرامعین، جواب ۳-۱۲ را به شکل زیر نوشت:

$$s_\alpha^* = (B^T B)^{-1} B^T c \quad (۱۳-۳)$$

$$= (A_\alpha^T P A_\alpha)^{-1} A_\alpha^T P x \quad (۱۴-۳)$$

به طور خلاصه برای حالت  $k > \max\{n, m - n\}$  جواب مسئله بهینه سازی ۳-۶ را با توجه به ۳-۱۱ و ۳-۱۴ می توان به صورت زیر بیان کرد:

$$\begin{cases} s_\alpha^* = (A_\alpha^T P A_\alpha)^{-1} A_\alpha^T P x \\ s_l^* = A_l^T P (x - A_\alpha s_\alpha^*) \end{cases} \quad (۱۵-۳)$$

به عنوان نکته آخر توجه کنید مطابق آنچه در فصل های قبل نیز گفتیم، ماتریس  $A$  تصادفی است؛ یعنی ستون هایش از یک توزیع یکنواخت روی کره  $1$  واحد در فضای  $R^n$  به دست آمده اند. در این حالت می توان فرض کرد ماتریس های وارون به کار رفته در جواب به دست آمده ۳-۱۵ تقریباً به طور حتم <sup>۱</sup> وجود دارند. با تمام فرضیاتی که برای این مسئله در نظر گرفتیم جواب ۳-۱۵ جواب یکتای مسئله ۳-۶ است. به سادگی می توان نشان داد شرط  $k > \max\{n, m - n\}$  در حالت تساوی نیز یکتایی جواب را حفظ می کند. قابل ذکر است شرط  $k \geq \max\{n, m - n\}$  شرط اصلی برای یکتایی جواب است و اگر این شرط برقرار نباشد، جواب در حالت کلی یکتا نخواهد بود.

<sup>۱</sup>almost surely



مثال ۳-۴-۱. مسئله زیر را در نظر بگیرید:

$$\min z = s_1^2 + s_2^2 + s_5^2 \quad (۱۶-۳)$$

$$s.t \quad s_1 + s_2 + 4s_3 - s_4 + 3s_5 = 1 \quad (۱۷-۳)$$

$$s_1 - s_2 - 2s_3 + 3s_4 + s_5 = 1 \quad (۱۸-۳)$$

محدودیت های ۱۷-۳ و ۱۸-۳ همان دستگاه  $x = As$  به کار رفته در مثال ۲-۵-۳ است. هدف مسئله فوق یافتن فاصله دستگاه  $x = As$  (محدودیت های ۱۷-۳ و ۱۸-۳) تا زیرفضای دوبعدی تولید شده توسط  $a_1$  و  $a_2$  از فضای  $R^5$  است. توجه کنید شرط  $k \geq \max\{n, m\} (= \{2, 3\})$  نیز برقرار است. دو افراز ماتریس  $A$ ، بردار  $s$  را به صورت زیر در نظر بگیرید:

$$s_\alpha^T = (s_1, s_2) \quad s_l^T = (s_3, s_4, s_5)$$

و

$$A_\alpha = \begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix} \quad A_l = \begin{pmatrix} 4 & -1 & 3 \\ -2 & 3 & 1 \end{pmatrix}$$

دستگاه  $x = As$  را به صورت زیر بازنویسی می کنیم:

$$\begin{pmatrix} 4 & -1 & 3 \\ -2 & 3 & 1 \end{pmatrix} \begin{pmatrix} s_3 \\ s_4 \\ s_5 \end{pmatrix} = \begin{pmatrix} 1 \\ 1 \end{pmatrix} - \begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix} \begin{pmatrix} s_1 \\ s_2 \end{pmatrix} \quad (۱۹-۳)$$

از طرفی داریم:

$$\begin{pmatrix} 4 & -1 & 3 \\ -2 & 3 & 1 \end{pmatrix}^\dagger = \begin{pmatrix} 4 & -2 \\ -1 & 3 \\ 3 & 1 \end{pmatrix} \left( \begin{pmatrix} 4 & -1 & 3 \\ -2 & 3 & 1 \end{pmatrix} \begin{pmatrix} 4 & -2 \\ -1 & 3 \\ 3 & 1 \end{pmatrix} \right)^{-1} \quad (20-3)$$

$$= \begin{pmatrix} \frac{2}{15} & \frac{-1}{15} \\ \frac{1}{30} & \frac{7}{30} \\ \frac{1}{6} & \frac{1}{6} \end{pmatrix} \quad (21-3)$$

لذا با توجه به روابط ۱۹-۳ و ۲۱-۳ داریم:

$$\begin{pmatrix} s_3 \\ s_4 \\ s_5 \end{pmatrix} = \begin{pmatrix} \frac{2}{15} & \frac{-1}{15} \\ \frac{1}{30} & \frac{7}{30} \\ \frac{1}{6} & \frac{1}{6} \end{pmatrix} \begin{pmatrix} 1 \\ 1 \end{pmatrix} - \begin{pmatrix} \frac{2}{15} & \frac{-1}{15} \\ \frac{1}{30} & \frac{7}{30} \\ \frac{1}{6} & \frac{1}{6} \end{pmatrix} \begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix} \begin{pmatrix} s_1 \\ s_2 \end{pmatrix} \quad (22-3)$$

$$= \begin{pmatrix} \frac{1}{15} \\ \frac{4}{15} \\ \frac{1}{3} \end{pmatrix} - \begin{pmatrix} \frac{1}{15} & \frac{1}{5} \\ \frac{4}{15} & \frac{-1}{5} \\ \frac{1}{3} & 0 \end{pmatrix} \begin{pmatrix} s_1 \\ s_2 \end{pmatrix} \quad (23-3)$$

اکنون با توجه به رابطه بدست آمده فوق، می خواهیم مسئله بدون قید زیر را حل کنیم:

$$\min \left\| \begin{pmatrix} \frac{1}{15} \\ \frac{4}{15} \\ \frac{1}{3} \end{pmatrix} - \begin{pmatrix} \frac{1}{15} & \frac{1}{5} \\ \frac{4}{15} & \frac{-1}{5} \\ \frac{1}{3} & 0 \end{pmatrix} \begin{pmatrix} s_1 \\ s_2 \end{pmatrix} \right\|_2 \quad (24-3)$$

و جواب این مسئله با توجه به رابطه ۷-۳ برابر است با:

$$\begin{pmatrix} s_1 \\ s_2 \end{pmatrix} = \begin{pmatrix} \frac{1}{15} & \frac{1}{5} \\ \frac{4}{15} & \frac{-1}{15} \\ \frac{1}{3} & 0 \end{pmatrix}^\dagger \begin{pmatrix} \frac{1}{15} \\ \frac{4}{15} \\ \frac{1}{3} \end{pmatrix} \quad (25-3)$$

که در آن

$$\begin{aligned} \begin{pmatrix} \frac{1}{15} & \frac{1}{5} \\ \frac{1}{15} & \frac{1}{5} \\ \frac{1}{3} & 0 \end{pmatrix}^\dagger &= \left( \begin{pmatrix} \frac{1}{15} & \frac{1}{15} & \frac{1}{3} \\ \frac{1}{5} & \frac{1}{5} & 0 \end{pmatrix} \begin{pmatrix} \frac{1}{15} & \frac{1}{5} \\ \frac{1}{15} & \frac{1}{5} \\ \frac{1}{3} & 0 \end{pmatrix} \right)^{-1} \begin{pmatrix} \frac{1}{15} & \frac{1}{15} & \frac{1}{3} \\ \frac{1}{5} & \frac{1}{5} & 0 \end{pmatrix} \\ &= \begin{pmatrix} 1 & 1 & 2 \\ 3 & -2 & 1 \end{pmatrix} \end{aligned}$$

لذا با توجه به روابط ۲۳-۳ و ۲۵-۳:

$$\begin{pmatrix} s_1 \\ s_2 \end{pmatrix}^* = \begin{pmatrix} 1 \\ 0 \end{pmatrix}, \quad \begin{pmatrix} s_3 \\ s_4 \\ s_5 \end{pmatrix}^* = \begin{pmatrix} 0 \\ 0 \\ 0 \end{pmatrix} \quad (26-3)$$

توجه کنید جواب فوق همان جواب پایه ای به دست آمده ۱۶-۲ در مثال ۱-۵-۲ است.

### ۵-۳ یافتن جواب LS دستگاه معادلات خطی فرامعین

برای یافتن جواب LS (به مفهوم کمترین مربعات) معادلات خطی فرامعین، باید مسئله بهینه سازی نامقید زیر حل شود:

$$\min \|b - Ax\|_2; \quad x \in R^m \quad (27-3)$$

که در آن  $A \in R^{n \times m}$  با  $n < m$  و  $b \in R^n$  هستند. برای حل، ابتدا به یادآوری فرم مسائل بهینه سازی نامقید و قضایای مربوط به شرایط لازم و کافی برای بهینگی این دسته از مسائل نیاز است. مسائل بهینه سازی نامقید در حالت کلی به فرم زیر است [۱۸]:

$$\min f(x) \quad \text{s.t.} \quad x \in \Omega \quad (28-3)$$

که در آن  $f$  یک تابع حقیقی مقدار روی  $\Omega$  است. (تمامی تعاریف و قضایای مربوط به این بخش از [۱۸] استخراج شده اند.)

**تعریف ۳-۵-۱.** نقطه  $x^* \in \Omega$  مینیم موضعی یا مینیم محلی تابع  $f$  روی  $\Omega$  نامیده می شود اگر یک  $\epsilon > 0$  وجود داشته باشد به طوری که برای هر  $x \neq x^*$  و  $|x - x^*| < \epsilon$ ،  $f(x)$  بزرگتر مساوی  $f(x^*)$  باشد  
 $(f(x) \geq f(x^*))$ .

**تعریف ۳-۵-۲.** نقطه  $x^* \in \Omega$  مینیم سراسری  $f$  روی  $\Omega$  نامیده می شود اگر  $f(x) \geq f(x^*)$  برای هر  $x \in \Omega$ .

**قضیه ۳-۵-۳.** فرض کنید  $f : \Omega \rightarrow R$  یک تابع محدب روی مجموعه محدب  $\Omega$  باشد. در این صورت:

۱. هر مینیم محلی  $f$  روی  $\Omega$  یک مینیم سراسری روی  $\Omega$  است.

۲. اگر  $f$  محدب و  $\Omega$  باز باشد، در این صورت شرط  $\nabla f(x^*) = 0$  شرط لازم و کافی است برای آن که بردار  $x^*$  مینیم سراسری روی  $f$  باشد.

□

برهان. در مرجع [۱۸] ببینید.

**تعریف ۳-۵-۴.** ماتریس  $A \in R^{n \times n}$  نیمه معین مثبت<sup>۱</sup> است اگر و تنها اگر برای هر  $x \in R^n$  و  $x \neq 0$  داشته باشیم  $x^T A x \geq 0$ .

**قضیه ۳-۵-۵.** فرض کنید  $f$  تابعی دوبار پیوسته و مشتق پذیر باشد ( $f \in C^2$ ). در این صورت  $f$  یک تابع محدب روی مجموعه محدب  $\Omega$  است، اگر و تنها اگر، ماتریس هسین<sup>۲</sup> تابع  $f$  ( $F$ ) برای هر  $x \in \Omega$  نیمه معین مثبت باشد.

□

برهان. در مرجع [۱۸] ببینید.

تابع هدف مسئله ۳-۲۷ با تابع هدف  $\frac{1}{2} \|b - Ax\|_2^2$  دارای مینیم یکسان اند. لذا مسئله ۳-۲۷ با مسئله زیر معادل است:

$$\min \frac{1}{2} \|b - Ax\|_2^2 ; x \in R^m \quad (۳-۲۹)$$

<sup>۱</sup>positive semidefinite

<sup>۲</sup>Hessian

تابع هدف مسئله فوق را در نظر بگیرید:

$$\begin{aligned} f(x) &= \frac{1}{2} \|b - Ax\|_2^2 \\ &= \frac{1}{2} (b - Ax)^T (b - Ax) \\ &= \frac{1}{2} x^T A^T A x - x^T A^T b + \frac{1}{2} b^T b \end{aligned}$$

لذا بردار گرادیان<sup>۱</sup> تابع  $f$  برابر است با:

$$\nabla f(x) = A^T A x - A^T b \quad (30-3)$$

و هسین تابع  $f$  برابر است با:

$$F(x) = \nabla^2 f(x) = A^T A \quad (31-3)$$

و برای هر  $y \in R^m$  خواهیم داشت:

$$\begin{aligned} y^T \nabla^2 f(x) y &= y^T A^T A y \\ &= \|Ay\|_2^2 \geq 0 \end{aligned}$$

لذا ماتریس هسین تابع  $f$  (روی  $R^m$ ) نیمه معین مثبت است. بنابراین بنا به قضیه ۳-۵-۵ تابع  $f$  یک تابع محدب روی  $R^m$  است. اکنون جواب معادله  $\nabla f(x) = 0$  را به دست می آوریم:

$$\nabla f(x) = A^T A x - A^T b = 0 \quad (32-3)$$

خواهیم داشت:

$$x^* = (A^T A)^{-1} A^T b \quad (33-3)$$

(تنها با فرض رتبه کامل بودن  $A$  وارون ماتریس  $A^T A$  وجود دارد.) بنا به قضیه ۳-۵-۳،  $x^*$  به دست آمده شرط

---

<sup>۱</sup>gradient vector

لازم و کافی را برای مینیمم سراسری تابع هدف مسئله ۳-۲۹ داراست. به طور معادل  $x^* = (A^T A)^{-1} A^T b$  جواب مسئله ۳-۲۷ است.

### ۳-۶ یافتن جواب دارای کمترین نرم $l_2$ دستگاه معادلات خطی فرومعین

برای یافتن جوابی از دستگاه معادلات خطی فرومعین که دارای کمترین نرم  $l_2$  است، باید مسئله زیر حل شود:

$$\min \|x\|_2 \text{ s.t. } Ax = b \quad (3-34)$$

که در آن  $A \in R^{n \times m}$  و  $b \in R^n$  با  $n < m$  هستند. برای یافتن جواب بهینه مسئله فوق به یادآوری فرم کلی مسائل بهینه سازی مقید با محدودیت تساوی و قضایای مربوط به حل این نوع مسائل نیاز است. فرم کلی مسائل بهینه سازی مقید با محدودیت تساوی به صورت زیر است [۱۸]:

$$\min f(x) \text{ s.t. } h(x) = 0 \quad (3-35)$$

که در آن  $x \in R^m$  و  $f: R^m \rightarrow R$  و  $h: R^m \rightarrow R^n$  با  $n \leq m$  هستند. همچنین فرض می شود تابع  $h$  پیوسته و مشتق پذیر باشد ( $h \in C^1$ ).

**تعریف ۳-۶-۱.** نقطه  $x^*$  که در شرایط  $h_i(x^*) = 0, i = 1, \dots, n$  صدق می کند، یک نقطه منظم نامیده می شود اگر بردارهای گرادیان  $\nabla h_i(x^*), i = 1, \dots, n$  مستقل خطی باشند [۱۸]. اگر  $Dh(x^*)$  ماتریس ژاکوبین تابع  $h = (h_1, h_2, \dots, h_n)^T$  در  $x^*$  باشد که به صورت

$$Dh(x^*) = (\nabla h_1(x^*) \nabla h_2(x^*) \dots \nabla h_n(x^*))^T$$

است؛ در این صورت  $x^*$  یک نقطه منظم است اگر و تنها اگر  $Dh(x^*)$  رتبه کامل باشد.

**قضیه ۳-۶-۲ (قضیه لاگرانژ).** فرض کنید  $x^*$  یک نقطه مینیمم (یا ماکزیمم) تابع  $f: R^m \rightarrow R$  باشد به طوری که  $h(x^*) = 0$  برای  $h: R^m \rightarrow R^n$  با  $n \leq m$ . اگر  $x^*$  یک نقطه منظم باشد، در این صورت یک  $\lambda^* \in R^n$  وجود دارد به طوری که:

$$Df(x^*) + \lambda^{*T} Dh(x^*) = 0 \quad (3-36)$$

برهان. در [۱۸] ببینید.

□

به عبارت دیگر قضیه لاگرانژ<sup>۱</sup> بیان می کند اگر  $x^*$  یک نقطه اکسترمم تابع هدف  $f$  باشد، آنگاه بردار گرادیان تابع  $f$  را می توان به صورت یک ترکیب خطی از بردار گرادیان تابع شرایط نوشت. در رابطه ۳-۳۶ بردار  $\lambda^*$  بردار ضرایب لاگرانژ<sup>۲</sup> و مولفه های آن ضرایب لاگرانژ نامیده می شوند [۱۸]. متناسب با رابطه ۳-۳۶ می توان تابع لاگرانژی  $l : R^m \times R^n \rightarrow R$  را به صورت زیر معرفی کرد:

$$l(x, \lambda) = f(x) + \lambda^T h(x) \quad (3-37)$$

بنابراین شرایط لاگرانژی برای مینیمم موضعی  $x^*$  با استفاده از تابع لاگرانژی را می توان به صورت زیر نمایش داد:

$$Dl(x^*, \lambda^*) = 0 \quad (3-38)$$

به عبارت دیگر، شرایط لازم در قضیه لاگرانژ برای مینیمم موضعی را می توان به صورت زیر نوشت:

$$\begin{cases} D_x l(x, \lambda) = 0 \\ D_\lambda l(x, \lambda) = 0 \end{cases} \quad (3-39)$$

به عبارتی:

$$\begin{cases} Df(x) + \lambda^T Dh(x) = 0 \\ h(x) = 0 \end{cases} \quad (3-40)$$

یادآوری می شود شرایط لاگرانژی تنها شرایط لازم هستند؛ اما کافی نیستند. به این معنا که اگر  $x^*$  در معادلات بالا صدق کند، لزوماً یک نقطه اکسترمم نیست.

در نظر بگیرید توابع  $f : R^m \rightarrow R$  و  $h : R^m \rightarrow R^n$  توابعی دوبار پیوسته و مشتق پذیر باشند  $(f, h \in C^2)$  و همچنین

$$l(x, \lambda) = f(x) + \lambda^T h(x) = f(x) + \lambda_1 h_1(x) + \dots + \lambda_n h_n(x) \quad (3-41)$$

<sup>۱</sup>Lagrange's Theorem

<sup>۲</sup>Lagrange multiplier vector

<sup>۳</sup>Lagrangian function

تابع لاگرانژی باشد. در این صورت  $L(x, \lambda)$  ماتریس هسین تابع  $l(x, \lambda)$  به صورت زیر خواهد بود:

$$L(x, \lambda) = F(x) + \lambda_1 H_1(x) + \dots + \lambda_n H_n(x) \quad (42-3)$$

که در آن ماتریس هسین تابع  $f$  در  $x$  و  $H_k(x)$  ماتریس هسین تابع محدودیت  $h_k$  ( $k = 1, \dots, n$ ) در  $x$  است.

تعریف ۳-۶-۳ (فضای تانژانتی). فضای تانژانتی<sup>۱</sup> در نقطه  $x^*$  روی رویه  $S = \{x \in R^m; h(x) = 0\}$  مجموعه ای به صورت زیر است [۱۸]:

$$T(x^*) = \{y; Dh(x^*)y = 0\} \quad (43-3)$$

قضیه ۳-۶-۴. فرض کنید  $f$  و  $h$  توابعی دوبار پیوسته و مشتق پذیر باشند و نقطه  $x^* \in R^m$  و  $\lambda^* \in R^n$  وجود داشته باشند به طوری که

$$Df(x^*) + \lambda^* Dh(x^*) = 0 \quad ۱.$$

$$۲. \text{ برای هر } y \in T(x^*), y \neq 0, \text{ داشته باشیم } y^T L(x^*, \lambda^*)y > 0.$$

در این صورت  $x^*$  مینیمم موضعی اکید تابع  $f$  است به طوری که  $h(x) = 0$ .

□

برهان. در [۱۸] ببینید.

مثال ۳-۶-۵. مسئله زیر را در نظر بگیرید:

$$\min \frac{1}{2} x^T Q x \quad \text{s.t.} \quad Ax = b \quad (44-3)$$

که در آن  $Q^T = Q > 0$  (ماتریس معین مثبت متقارن) و  $A \in R^{n \times m}$  باشد به طوری که  $n < m$  و  $\text{rank}(A) = n$ . ابتدا فرم تابع لاگرانژی مسئله فوق را در نظر بگیرید:

$$l(x, \lambda) = \frac{1}{2} x^T Q x + \lambda^T (b - Ax) \quad (45-3)$$

<sup>۱</sup>tangent space



اول- شرط لازم لاگرانژی به صورت زیر خواهد بود:

$$D_x l(x, \lambda) = x^T Q - \lambda^T A = 0 \quad (46-3)$$

بنابراین داریم:

$$x = Q^{-1} A^T \lambda \quad (47-3)$$

با ضرب طرفین معادله در ماتریس  $A$  داریم:

$$Ax = A Q^{-1} A^T \lambda \quad (48-3)$$

از طرفی داریم  $Ax = b$  بنابراین

$$b = A Q^{-1} A^T \lambda \quad (49-3)$$

لذا

$$\lambda = (A Q^{-1} A^T)^{-1} b \quad (50-3)$$

بنابراین

$$x^* = Q^{-1} A^T (A Q^{-1} A^T)^{-1} b \quad (51-3)$$

برای نشان دادن آن که  $x^*$  به دست آمده در فوق جواب بهینه است، کافی است دوم- شرط کافی در رابطه؟؟ بررسی شود. برای این منظور، ماتریس هسین تابع لاگرانژی در  $(x, \lambda)$  را در نظر بگیرید:

$$L(x, \lambda) = Q \quad (52-3)$$

که بنا به فرض  $Q$  یک ماتریس معین مثبت است؛ بنابراین  $x^*$  جواب بهینه مدنظر است.

در مثال قبل به طور خاص  $Q = I_m$  را در نظر بگیرید:

$$\min \frac{1}{2} \|x\|_2^2 \quad \text{s.t.} \quad Ax = b \quad (53-3)$$

که معادل با مسئله ۳-۳۴ است. لذا جواب این مسئله با توجه به جواب به دست آمده ۳-۵۱ به صورت زیر خواهد بود:

$$x = A^T(AA^T)^{-1}b \quad (54-3)$$

### ۷-۳ یافتن جواب مینیمم فاصله تا زیرفضا به روشی دیگر

همان طور که در بخش های قبل گفته شد مسئله مینیمم فاصله تا یک زیرفضای بخصوص را می توان به صورت زیر نوشت:

$$\min s^T W s \quad \text{s.t.} \quad As = x \quad (55-3)$$

که در آن  $W$  یک ماتریس قطری با  $k$  درایه ۱ روی قطر اصلی و بقیه درایه ها صفر هستند. لذا می توان ماتریس  $W$  را به صورت  $\begin{pmatrix} \circ & \circ \\ \circ & I_k \end{pmatrix}$  در نظر گرفت که  $k$  تعداد منابع غیر فعال است. برای یافتن جواب مسئله فوق با استفاده از شرایط لاگرانژی ابتدا مسئله زیر برای  $\delta > 0$  حل می شود:

$$\min s^T W s = (s_\alpha^T \ s_l^T) \begin{pmatrix} \delta I_{m-k} & \circ \\ \circ & I_k \end{pmatrix} \begin{pmatrix} s_\alpha \\ s_l \end{pmatrix} \quad (56-3)$$

$$\text{s.t.} \quad As = x$$

سپس  $\delta$  را به سمت صفر میل می دهیم [۱۹]. با توجه به این که  $W$  یک ماتریس معین مثبت متقارن است؛ با استفاده از جواب به دست آمده ۳-۵۱ در مثال ۳-۶-۵ جواب مسئله فوق به صورت زیر خواهد بود:

$$s^* = W^{-1}A^T(AW^{-1}A^T)^{-1}x \quad (۵۷-۳)$$

حال سعی می شود جواب به دست آمده را به فرم ساده تری نوشت.

لم ۳-۷-۱ (لم وارون ماتریس).<sup>۱</sup> برای ماتریس های  $A, B, C$  و  $D$  با ابعاد متناسب رابطه زیر برقرار است:

$$(A - BD^{-1}C)^{-1} = A^{-1} + A^{-1}B(D - CA^{-1}B)^{-1}CA^{-1} \quad (۵۸-۳)$$

(با فرض موجود بودن وارون های به کار رفته در این رابطه. ) فرمول بالا به فرمول *Woodbury*<sup>۲</sup> معروف است.

۳

□

برهان. در [۲۳] ببینید.

تعریف می کنیم  $\omega = \delta^{-1}$ ، آنگاه داریم:

$$(AW^{-1}A^T) = ((A_\alpha \ A_l) \begin{pmatrix} \omega I & \circ \\ \circ & I \end{pmatrix} \begin{pmatrix} A_\alpha^T \\ A_l^T \end{pmatrix})^{-1} \quad (۵۹-۳)$$

$$= \underbrace{(A_l A_l^T + A_\alpha (\omega I) A_\alpha^T)^{-1}}_{P^{-1}} \quad (۶۰-۳)$$

$$= P - PA_\alpha (\delta I + \underbrace{A_\alpha^T P A_\alpha}_B)^{-1} A_\alpha^T P \quad (۶۱-۳)$$

در نوشتن تساوی آخر از لم وارون ماتریس استفاده شده است. از طرفی

$$\begin{pmatrix} s_\alpha^* \\ s_l^* \end{pmatrix} = \begin{pmatrix} \omega I & \circ \\ \circ & I \end{pmatrix} \begin{pmatrix} A_\alpha^T \\ A_l^T \end{pmatrix} (AW^{-1}A^T)^{-1}x \quad (۶۲-۳)$$

$$= \begin{pmatrix} \omega A_\alpha^T \\ A_l^T \end{pmatrix} (AW^{-1}A^T)^{-1}x \quad (۶۳-۳)$$

<sup>۱</sup>Matrix inversion lemma

<sup>۲</sup>Woodbury formula

<sup>۳</sup>Sherman-Morrison-Woodbury

بنابراین

$$s_{\alpha}^* = \omega A_{\alpha}^T (AW^{-1}A^T)^{-1} x \quad (۶۴-۳)$$

$$= \omega (A_{\alpha}^T Px - B(\delta I + B)^{-1} A_{\alpha}^T Px) \quad (۶۵-۳)$$

$$= \omega ((\delta I + B) - B)(\delta I + B)^{-1} A_{\alpha}^T Px \quad (۶۶-۳)$$

$$= (\delta I + B)^{-1} A_{\alpha}^T Px \quad (۶۷-۳)$$

و

$$s_l^* = A_l^T Px - A_l^T P A_{\alpha} (\delta I + B)^{-1} A_{\alpha}^T Px \quad (۶۸-۳)$$

$$= A_l^T P (x - A_{\alpha} (\delta I + B)^{-1} A_{\alpha}^T Px) \quad (۶۹-۳)$$

$$= A_l^T P (x - A_{\alpha} s_{\alpha}^*) \quad (۷۰-۳)$$

به طور خلاصه جواب مسئله بهینه سازی ۵۵-۳ عبارت است از

$$\begin{cases} s_{\alpha}^* = (\delta I + B)^{-1} A_{\alpha}^T Px \\ s_l^* = A_l^T P (x - A_{\alpha} s_{\alpha}^*) \end{cases} \quad (۷۱-۳)$$

با میل دادن  $\delta$  به سمت صفر، عبارت فوق به همان جواب ۱۵-۳ منجر می شود. توجه کنید برای تضمین موجود بودن وارون های به کار رفته در عبارت اخیر به شرایط مشابه آنچه در به دست آوردن ۱۵-۳ فرض شد، نیاز است؛ یعنی باید  $k \geq \max\{n, m - n\}$  باشد.

## فصل ۴

# روش آشکارسازی-تخمین پایایی

### ۱-۴ مقدمه

در فصل های قبل اشاره کردیم برخی از روش های یافتن جواب های تنک دستگاه معادلات خطی فرو معین (، اغلب روش های مبتنی بر حل مسئله بهینه سازی)، از جمله روش ”جست و جوی زیر فضا“، نیاز به جست و جوی ترکیبیاتی دارند. در این فصل روشی برای ”آشکار سازی منابع فعال“ شرح داده می شود که با استفاده از آن، نیاز به این جست و جو برطرف خواهد شد. در این روش وضعیت ”فعالیت“ هر یک از منابع (هر یک از مولفه های بردار منبع  $s$ ) به طور جداگانه با مقایسه تابعی ساده با یک سطح آستانه، مشخص می شود. بنابراین مشخص کردن منابع فعال تنها به  $m$  مقایسه ساده نیاز دارد که بسیار ساده تر از جست و جوی ترکیبیاتی است. سپس با مشخص شدن منابع فعال (مولفه های فعال بردار منبع  $s$ )، معیارهای بهینگی متفاوتی که قبلا اشاره شدند، برای یافتن جواب اعمال می شوند.

آشکارسازی بهینه ”فعالیت“ یک منبع، با روابطی که به دست آورده می شود، نیاز به مقادیر سایر منابع، که در دسترس نیستند، دارد. لذا راهکار موجود یک آشکارسازی زیر بهینه بر اساس یک تخمین اولیه از بردار منبع است. سپس طی یک الگوریتم تکراری<sup>۱</sup> هم کیفیت تخمین و هم کیفیت آشکارسازی بهبود می یابد. به عبارت دیگر، در این روش با شروع از یک تخمین اولیه، ابتدا با آشکارسازی منابع فعال مشخص می شود. سپس بر اساس اطلاعات حاصل، تخمین بهتری از جواب به دست آورده می شود. در مرحله بعد، با استفاده از این تخمین، آشکارسازی را با دقت بیشتری انجام داده و به همین ترتیب، با تکرار این فرایند تخمین های دقیق تری به دست می آوریم. در هر مرحله بخش اول الگوریتم را ”گام آشکارسازی“ و بخش دوم را ”گام تقریب یا تخمین“ می نامند [۱۹]. به این ترتیب خانواده روش های حاصل ”آشکارسازی-تخمین پایایی“ یا به اختصار

<sup>۱</sup>Iterative

IDE<sup>۱</sup> نامیده می شوند. قابل ذکر است "آشکارسازی-تخمین پیاپی" یک متدولوژی برای حل مسئله است و لذا می توان هر یک از گام های آشکارسازی و تخمین را با روش هایی جز روش های پیاده شده در این فصل پیاده کرد و روش های کاملاً متفاوتی به دست آورد.

برای گام آشکارسازی از آزمون فرضیه و برای گام تقریب/تخمین می توان از معیارهای متفاوت بهینگی تنک استفاده کرد. در این فصل برای گام تقریب/تخمین از دو معیار که یکی مشابه تابع هزینه روش جست و جوی زیرفضا و دیگری مشابه تابع هزینه به کار رفته در روش جست و جوی تطابقی متعامد (گام ۸ الگوریتم ۲-۲) است، استفاده خواهد شد. این دو معیار را می توان به ترتیب متناظر با نگاه به مسئله در فضای  $s$ - و فضای  $x$ - دانست. لذا روش های حاصل به شکل IDE-s و IDE-x مشخص می شوند.

در این فصل ابتدا مباحث پیش نیاز روش IDE در مبحث آزمون فرضیه (یا آشکارسازی) باینری مطرح شده و سپس هر یک از دو گام آشکارسازی و تخمین به طور جداگانه شرح داده خواهد شد. به عنوان نکته آخر، متذکر می شویم عبارت "منابع فعال" گاه به معنای "واقعا فعال" و گاهی به معنای "فعال تشخیص داده شده" توسط گام آشکارسازی است که غالباً منظور حالت دوم خواهد بود. در صورتی که منابع فعال واقعی مدنظر باشند پسوند "واقعا" برای تاکید استفاده خواهد شد. همچنین از نماد های  $k_i$  و  $k_\alpha$  به ترتیب برای نمایش تعداد منابع فعال و غیر فعال استفاده خواهد شد که در فصل قبل این دو کمیت به ترتیب با  $p$  و  $k$  نشان داده می شد. در این فصل،  $k$  اندیس مرحله<sup>۲</sup> (یا تکرار) را مشخص می کند.

## ۲-۴ آزمون فرضیه (یا آشکارسازی) باینری

این بخش را با تعریف آزمون فرضیه شروع می کنیم:

تعریف ۱-۲-۴ (آزمون فرضیه).<sup>۳</sup> روشی است در علم آمار، برای بررسی ادعاها یا فرض هایی درباره پارامترهای توزیع در جوامع آماری. به بیان دیگر، آزمون فرضیه بخشی از علم آمار است که به دنبال تایید درستی یا نادرستی تعدادی فرضیه از پیش مشخص شده، بر اساس مشاهدات است.

موضوع مورد کاربرد ما در این فصل "آزمون فرضیه باینری"، یعنی حالتی که تنها دو فرضیه وجود دارد؛ است. به بیان دقیق تر، فرض کنید دو فرضیه  $H_0$  و  $H_1$  موجودند و ما می خواهیم بر اساس مشاهده بردار تصادفی  $y$  تصمیم بگیریم کدام یک از دو فرضیه برقرار است. مدل کردن ریاضی این رابطه به گونه ای است که توزیع بردار مشاهدات  $y$  را تحت دو فرضیه متفاوت می گیرند. به عبارت دیگر، فرضیه ها توزیع بردار مشاهدات  $y$  را تغییر

<sup>۱</sup>Iterative Detection-Estimation

<sup>۲</sup>iteration

<sup>۳</sup>Hypothesis Testing

می دهند؛ لذا با توجه به این توزیع می توان میان دو فرضیه تمایز قائل شد. مسئله آزمون فرضیه را در بررسی سیستم های رادار (و به طور کلی در مخابرات) و همچنین در حوزه "پردازش آماری سیگنال" غالباً تحت عنوان "آشکارسازی" می شناسند. در این دیدگاه، فرضیه  $H_1$  غالباً متناظر با وجود هدف (یا سیگنال) است و فرضیه  $H_0$  را متناظر با عدم وجود هدف (یا وجود نویز) است.

به بیان ریاضی، یک مسئله "آزمون فرضیه باینری" به صورت زیر نشان داده می شود [۲۴، ۲۵]:

$$\begin{aligned} H_0 &: y \sim P_0 \\ H_1 &: y \sim P_1 \end{aligned} \quad (1-4)$$

که در آن  $P_0$  نشان دهنده توزیع بردار  $y$  تحت فرضیه  $H_0$  و  $P_1$  نشان دهنده توزیع بردار  $y$  تحت فرضیه  $H_1$  است. فرض کنید مجموعه مقادیر ممکن بردار  $y$  را با  $\Gamma$  نشان دهیم. مسئله هنگامی حل می شود که مشخص شود برای هر  $y \in \Gamma$  کدام یک از دو فرضیه باید انتخاب شود. به عبارت دیگر، به دنبال یک "قاعده تصمیم گیری" هستیم که هر  $y \in \Gamma$  را به ۱ یا ۰ (متناظر با دو فرضیه  $H_0$  و  $H_1$ ) بنگارد. قاعده تصمیم گیری  $\delta$  را می توان به صورت یک تابع روی  $\Gamma$  به صورت زیر در نظر گرفت:

$$\delta(y) = \begin{cases} 1 & \text{if } y \in \Gamma_1 \\ 0 & \text{if } y \in \Gamma_0 (= \Gamma_1^c) \end{cases} \quad (2-4)$$

زیرمجموعه ای از  $\Gamma$  که به ۱ نگاشته می شود (یا فرضیه  $H_1$  را می پذیرد) را "ناحیه رد" <sup>۲</sup> و زیرمجموعه ای از  $\Gamma$  که به ۰ نگاشته می شود "ناحیه قبول" <sup>۳</sup> می نامند. این دو مجموعه را به ترتیب با  $\Gamma_1$  و  $\Gamma_0$  نشان می دهیم. لذا با این تفاسیر، هر قاعده تصمیم گیری متناظر با یک افراز مجموعه  $\Gamma$  به صورت  $\{\Gamma_0, \Gamma_1\}$  است. برای هر قاعده تصمیم گیری دو احتمال خطا وجود دارد:

۱. خطای نوع اول <sup>۴</sup>: رد فرضیه  $H_0$  وقتی آن درست باشد.

۲. خطای نوع دوم <sup>۵</sup>: قبول فرضیه  $H_0$  وقتی آن نادرست باشد.

در اصطلاح آشکارسازی خطای نوع اول را "هشدار بی مورد" <sup>۶</sup> و خطای نوع دوم را "آشکارسازی از دست رفته" <sup>۷</sup> می نامند. فرض کنید احتمال انتخاب فرضیه  $H_j$  تحت فرضیه  $H_i$  باشد؛ در این صورت می توان احتمال وقوع دو نوع خطای فوق را به صورت  $P_{Fa} = P_0(\Gamma_1)$  و  $P_{Mi} = P_1(\Gamma_0)$  نوشت. همچنین احتمال

---

¹decision rule	²rejection region	³acceptance region	⁴type I error	⁵type II error
⁶false alarm	⁷missed detection			

آشکارسازی صحیح<sup>۱</sup> عبارت است از  $P_{De} = 1 - P_{Mi} = P_1(\Gamma_1)$ . حال هدف ما یافتن قاعده تصمیم گیری به گونه ای است که احتمال هر دو نوع خطا تا حد ممکن کوچک باشد؛ یا به عبارتی احتمال آشکارسازی صحیح ( $P_{De}$ ) تا حد امکان بزرگ و احتمال هشدار بی مورد ( $P_{Fa}$ ) تا حد امکان کوچک باشد. روشن است که امکان مینیمم کردن هم زمان هر دو نوع خطا وجود ندارد؛ دو دیدگاه<sup>۲</sup> "بیز<sup>۱</sup>" و "نیمن-پیرسون<sup>۳</sup>" (NP) از جمله دیدگاه های موجود برای رفع این مشکل با انتخاب معیارهای بهینگی متفاوت هستند که در ادامه به آن ها پرداخته خواهد شد.

### ۳-۴ دیدگاه های بیز و نیمن-پیرسون

در دیدگاه بیز برای مینیمم کردن خطای نوع اول و خطای نوع دوم، به هر یک از دو فرضیه یک احتمال پیشین نسبت داده می شود؛ سپس یک ترکیب خطی معنی دار از دو نوع خطا تحت عنوان "ریسک بیز<sup>۳</sup>" مینیمم سازی می شود. اگر احتمال پیشین فرضیه  $H_i$  با  $\pi_i$  نشان داده شود، ریسک بیز نیز همان احتمال خطای کل خواهد بود. یعنی:

$$P_e = \pi_1 P_{Mi} + \pi_0 P_{Fa} \quad (۳-۴)$$

فرض کنید  $f_i(y)$  تابع چگالی احتمال بردار مشاهده  $y$  تحت فرضیه  $H_i$  باشد؛ در این صورت می توان نشان داد "ناحیه بحرانی" (یا ناحیه رد) آزمون بهینه، یعنی آزمونی که ریسک بیز را مینیمم می کند، به صورت زیر است [۲۴، ۲۵]:

$$L(y) \triangleq \frac{f_1(y)}{f_0(y)} > \tau \quad (۴-۴)$$

تابع  $L(y)$  را "نسبت درستمایی<sup>۴</sup>" و آزمون فوق را که با مقایسه نسبت درستمایی با یک سطح آستانه میان دو فرضیه تمایز قائل می شود "آزمون نسبت درستمایی" یا به اختصار آزمون LR می نامند. مقدار سطح آستانه توسط احتمال های پیشین دو فرضیه، که ضرایب ریسک نامیده می شوند، تعیین می شود. به عنوان نمونه برای ریسک ساده ای که در نظر گرفته شده، سطح آستانه آزمون بهینه عبارت است از  $\tau = \frac{\pi_0}{\pi_1}$ .

در دیدگاه نیمن-پیرسون (یا به اختصار NP) رویکرد متفاوتی وجود دارد و برخلاف دیدگاه بیز، احتمال پیشینی به دو فرضیه نسبت داده نمی شود. در دیدگاه نیمن-پیرسون سعی می شود با نگه داشتن  $P_{Fa}$  زیر حد

<sup>۱</sup>Bayes      <sup>۲</sup>Neyman-Pearson      <sup>۳</sup>Bayes risk      <sup>۴</sup>Likelihood Ratio



معینی، احتمال آشکارسازی صحیح را ماکزیمم کند. به بیان ریاضی مسئله بهینه سازی زیر در نظر گرفته می شود:

$$\max P_{De}(\delta) \quad s.t. \quad P_{Fa}(\delta) \leq \alpha \quad (5-4)$$

بهینه سازی روی تمام قاعده های تصمیم گیری ( $\delta$ ) ممکن انجام می شود و  $\alpha$  مقدار مجاز  $P_{Fa}$  است. به کمک لم نیمن-پیرسون می توان نشان داد که آزمون بهینه با این دیدگاه همان آزمون LR است؛ با این تفاوت که مقدار سطح آستانه ( $\tau$ ) به گونه ای تعیین می شود که  $P_{Fa}$  کمتر مساوی  $\alpha$  باشد. (اشاره می کنیم لم نیمن-پیرسون قاعده تصمیم گیری بهینه یکتای مدنظر مسئله بهینه سازی 4-5 را به دست می دهد.)

## 4-4 آزمون فرضیه مرکب و تست UMP

مسئله آزمون فرضیه ای که تاکنون بررسی کردیم، آزمون فرضیه ساده نام دارد که در آن، بردار مشاهدات تحت هر یک از دو فرضیه دارای توزیع مشخصی است. اما نوع دیگری از آزمون فرضیه که در آن بردار مشاهدات  $y$  تحت هر یک از دو فرضیه می تواند خانواده متفاوتی از توزیع ها را داشته باشد، آزمون فرضیه "مرکب" <sup>۱</sup> نامیده می شود. آزمون فرضیه مرکب برای مدل سازی مسائلی که در آن ها پارامترهای نامعلومی وجود دارند مناسب است. به عبارت دیگر فرض کنید  $\{P_\theta | \theta \in \Lambda\}$  یک خانواده از توزیع ها برای بردار  $y$  داده شده است ( $y \sim P_\theta$ ) که مولفه های بردار  $\theta$  در واقع همان پارامترهای نامعلوم و  $\Lambda$  فضای مقادیر ممکن برای  $\theta$  است. همچنین فرض کنید  $\{f_\theta(y) | \theta \in \Lambda\}$  توابع چگالی متناظر با خانواده توزیع های  $\{P_\theta\}$  باشند. یک مسئله آزمون فرضیه مرکب را می توان به صورت زیر بیان کرد:

$$\begin{aligned} H_0 &: \theta \in \Lambda_0 \\ H_1 &: \theta \in \Lambda_1 \end{aligned} \quad (6-4)$$

که در آن  $\Lambda_0 \cap \Lambda_1 = \emptyset$  و  $\Lambda_0 \cup \Lambda_1 = \Lambda$ . این امکان وجود دارد که برخی از مولفه های بردار پارامتر مقادیر کاملاً یکسانی را تحت دو فرضیه اختیار کنند؛ چنین پارامترهایی که به تمایز میان دو فرضیه کمکی نمی کنند، "پارامترهای مزاحم" <sup>۲</sup> می نامند.

نحوه مدل کردن پارامترها نیز در دو دیدگاه بیز و نیمن-پیرسون با هم متفاوت است. در دیدگاه بیز یک توزیع

<sup>۱</sup>composite      <sup>۲</sup>nuisance parameters

پیشین به بردار  $\theta$  نسبت داده می شود ( که در این صورت می توان توزیع نهایی  $y$  را تحت هر دو فرضیه محاسبه کرد )؛ در حالی که در دیدگاه نیمن-پیرسون پارامترها را کمیت های نامعلوم فرض می کنند . حالت دوم را در نظر بگیرید ؛ به طور کلی برای یک آزمون فرضیه مرکب ، احتمال های  $P_{Fa}$  و  $P_{De}$  علاوه بر قاعده تصمیم گیری تابعی از مقادیر پارامترهای نامعلوم نیز می باشند . در این صورت یک آزمون ایده آل به مفهوم NP ، به صورت زیر خواهد بود:

$$\max_{\theta \in \Lambda_1} P_{De}(\delta; \theta) \quad s.t. \quad P_{Fa}(\delta; \theta) \leq \alpha, \theta \in \Lambda. \quad (7-4)$$

چنین آزمونی را در صورت وجود ، UMP<sup>۱</sup> ( به طور یکنواخت قدرتمندترین آزمون ) می نامند . اما آزمون های UMP تحت شرایط بسیار خاصی وجود دارند ؛ به عبارت دیگر به ندرت می توان مسئله ای را یافت که برای آن آزمون UMP موجود باشد [۲۴، ۲۵] . به عنوان مثال ، مسئله ای را در نظر بگیرید که علاوه بر پارامتر اصلی تفکیک کننده دو فرضیه مثل  $\theta_1 \in \{0, 1\}$  یک پارامتر مزاحم مانند  $\theta_2$  داشته باشد که مجموعه مقادیر یکسانی را تحت دو فرضیه اختیار می کند . حال می توان آزمون بهینه یعنی آزمون LR را برای هر یک از مقادیر  $\theta_2$  به دست آورد . آزمون UMP در صورتی وجود دارد که ناحیه بحرانی برای تمام این آزمون های بهینه یکسان باشد ؛ یعنی ناحیه بحرانی آزمون LR به  $\theta_2$  بستگی نداشته باشد .

## ۵-۴ آماره بسنده

خانواده پارامتری توزیع های  $\{P_\theta\}$  با توابع چگالی متناظر  $\{f_\theta(y)\}$  معرفی شده در قبل را در نظر بگیرید . آماره بسنده<sup>۲</sup> برای یک پارامتر آماره ای است از داده ها که تمام اطلاعات مورد نیاز برای تخمین آن پارامتر را در بر دارد . به عبارت دیگر ، آماره بسنده برای  $\theta$  تابعی از  $y$  است که تمام اطلاعات موجود در  $y$  را برای تخمین  $\theta$  یا تمایز میان دو فرضیه وابسته به  $\theta$  را داراست . به بیان ریاضی ، آماره  $t = h(y)$  یک آماره بسنده برای  $\theta$  است اگر توزیع نمونه  $y$  به شرط  $t$  به  $\theta$  بستگی نداشته باشد . در این صورت می توان نوشت :

$$f_\theta(y) = f(y|t) f_\theta(t) \quad (8-4)$$

<sup>۱</sup>Uniformly Most Powerful

<sup>۲</sup>sufficient statistic

(برای هر  $t = h(y)$ ) با توجه به رابطه فوق، "نسبت درستمایی" برای  $t$  و  $y$  یکسان است. یعنی:

$$\frac{f_1(t)}{f_0(t)} = \frac{f_1(y)}{f_0(y)} \quad (9-4)$$

به این ترتیب آزمون LR را می توان به جای  $y$  بر حسب آماره بسنده  $t$  بیان کرد. لذا اطلاعات موجود در  $y$  و  $t$  برای تمایز میان دو فرضیه یکسان خواهد بود. از آن جا که بعد آماره بسنده غالباً کمتر از بعد بردار مشاهدات است، مسئله آزمون فرضیه معادل بر حسب آن غالباً ساده تر حل می شود [۲۴].

## ۶-۴ گام آشکارسازی

هدف این بخش، تشریح روشی برای آشکارسازی منابع فعال با استفاده از ایده های "آزمون فرضیه" (از مرجع [۱۹]) است. برای این منظور، از توزیع مخلوط گوسی برای مدل سازی آماری منابع استفاده خواهد شد. در این روش، فعالیت هر منبع (یعنی هر مولفه از بردار  $s$ ) به طور جداگانه و مستقل از سایر منابع بررسی خواهد شد. برای شروع فرض کنید می خواهیم فعال بودن  $s_1$  را مورد آزمون قرار دهیم. برای این منظور رابطه  $x = As$  را به شکل:

$$x = s_1 a_1 + \sum_{i=2}^m s_i a_i \quad (10-4)$$

نوشته و مسئله آزمون فرضیه زیر را مطرح می کنیم:

$$\begin{aligned} H_0 : s_1 &\sim \mathcal{N}(0, \sigma_1^2) \\ H_1 : s_1 &\sim \mathcal{N}(0, \sigma_1^2) \end{aligned} \quad (11-4)$$

فرض بر این است که  $\sigma_1^2 \ll \sigma_0^2$ ؛ لذا فرضیه  $H_0$  متناظر با ناچیز بودن مقدار  $s$ ، یعنی غیر فعال بودن آن است و به همین ترتیب،  $H_1$  نشان دهنده قابل ملاحظه بودن احتمال مقدار  $s_1$ ، یا به عبارت دیگر فعال بودن آن می باشد. بردار مخلوط  $x$  نیز همان بردار مشاهدات است که بر مبنای آن باید میان  $H_0$  و  $H_1$  قضاوت کنیم.

می توان نشان داد کمیت اسکالر  $a_1^T x$  از جهت تمایز گذاشتن میان دو فرضیه، دارای همان اطلاعاتی است که در بردار مخلوط  $x$  وجود دارد. یعنی  $t$  آماره بسنده در مسئله است. برای نشان دادن این مطلب از یک استدلال شهودی استفاده می شود [۱۹]. متناظر با هر بردار غیرصفر  $a_1$  همواره می توان  $n-1$  بردار  $\{a_i'\}_{i=2}^n$

یافت، به طوری که مجموعه  $\{a_1, a'_2, \dots, a'_n\}$  یک پایه متعامد یکه<sup>۱</sup> برای فضای  $\mathbb{R}^n$  بسازند. مولفه های بردار  $x$  در این پایه ی جدید عبارتند از  $\{a_1^T x, a_2'^T x, \dots, a_n'^T x\}$ . چون تغییر پایه یک تبدیل وارون پذیر است و در نتیجه باعث از بین رفتن اطلاعات آماری موجود در  $x$  نمی شود، لذا می توان از این مولفه ها به جای مولفه های اولیه بردار  $x$  برای آزمون دو فرضیه به کار برد. از طرفی برای  $2 \leq j \leq m$  داریم:

$$\begin{aligned} a_j'^T x &= a_j'^T (s_1 a_1 + \sum_{i=2}^m s_i a_i) \\ &= s_1 a_j'^T a_1 + \sum_{i=2}^m s_i a_j'^T a_i \\ &= \sum_{i=2}^m s_i a_j'^T a_i \end{aligned}$$

پس از میان مولفه های جدید تنها  $t$  است که وابسته به  $s_1$  می باشد. از آن جا که مولفه های  $\{a_i'^T x\}_{i=2}^m$  به  $s_1$  وابسته نیستند؛ لذا تحت دو فرضیه یکسان می باشند. در نتیجه اطلاعاتی در رابطه با جداسازی دو فرضیه در اختیارمان قرار نمی دهند و می توان آن ها را کنار گذاشت. این امر یعنی مولفه  $t$  تمام اطلاعات موجود در  $x$  مربوط به تمایز دو فرضیه را داراست و لذا آماره ی بسنده است.

حال می توان مسئله آزمون فرضیه ۴-۱۱ را برحسب آماره بسنده  $t$  بیان کرد. برای بررسی مسئله از چارچوب نیمن-پیرسون تبعیت خواهد شد. یعنی به دو فرضیه احتمال پیشین نسبت داده نخواهد شد. به علاوه، سایر منابع یعنی  $\{s_i\}_{i=2}^m$  به عنوان پارامترهای مزاحم مدل می شوند، نه متغیرهای تصادفی. تعریف می کنیم:

$$\mu = \sum_{i=2}^m s_i a_1^T a_i \quad (۱۲-۴)$$

لذا می توان نوشت:

$$\begin{aligned} t &= a_1^T x \\ &= a_1^T (s_1 a_1 + \sum_{i=2}^m s_i a_i) \\ &= s_1 (a_1^T a_1) + \sum_{i=2}^m s_i a_1^T a_i \\ &= s_1 + \mu \end{aligned}$$

---

<sup>۱</sup>orthonormal

که در آن  $\mu$  یک پارامتر نامعلوم است. بنابراین این آزمون معادل بر حسب آماره بسنده عبارت است از:

$$H_0: t \sim \mathcal{N}(\mu, \sigma_0^2) \quad (13-4)$$

$$H_1: t \sim \mathcal{N}(\mu, \sigma_1^2)$$

از طرفی می‌دانیم آزمون بهینه به مفهوم NP "آزمون نسبت درست‌نمایی" است. فرض کنید  $f_i(t)$  تابع چگالی احتمال آماره  $t$  تحت فرضیه  $i$ -ام باشد. در این صورت آزمون 13-4 را در صورتی می‌پذیرد که:

$$\log\left(\frac{f_1(t)}{f_0(t)}\right) > \tau \quad (14-4)$$

همچنین سطح آستانه  $\tau$  توسط میزان مجاز احتمال "هشدار بی‌مورد" یا اندازه آزمون تعیین می‌شود. برای مسئله 13-4 داریم:

$$f_i(t) = \frac{1}{\sigma_i \sqrt{2\pi}} \exp\left(-\frac{(t-\mu)^2}{2\sigma_i^2}\right) \quad (15-4)$$

پس ناحیه بحرانی برای آزمون بهینه به صورت زیر است:

$$\log\left(\frac{\sigma_0}{\sigma_1}\right) + \left(\frac{1}{2\sigma_0^2} - \frac{1}{2\sigma_1^2}\right)(t-\mu)^2 > \tau \quad (16-4)$$

با جذب ثابت‌های معلوم  $\sigma_0$  و  $\sigma_1$  به درون سطح آستانه "ناحیه بحرانی" را می‌توان به فرم ساده زیر نوشت:

$$|t - \mu| > \tau' \quad (17-4)$$

به طوری که:

$$\tau' = \left(\frac{2\sigma_0^2\sigma_1^2(\tau - \log(\sigma_0/\sigma_1))}{\sigma_1^2 - \sigma_0^2}\right)^{\frac{1}{2}} \quad (18-4)$$

از آن جا که تعیین ناحیه بحرانی وابسته به پارامتر نامعلوم  $\mu$  است، می‌توان نتیجه گرفت که آزمون UMP وجود ندارد. همچنین، ناحیه بحرانی بازه‌ای است که مرکز آن را  $\mu$  و شعاع آن را نیز میزان مجاز  $P_{Fa}$  از طریق کنترل سطح آستانه معین می‌کند.

یادآوری می کنیم  $\mu$  تابعی از مقادیر  $\{s_i\}_{i=2}^m$  است. بنابراین برای پیاده سازی آزمون بهینه، برای آزمون فعالیت منبع  $s_1$  نیاز به دانستن مقادیر  $\{s_i\}_{i=2}^m$  داریم. از آن جا که مقادیر واقعی این مولفه ها در دسترس نیستند؛ لذا تخمینی از این مقادیر را به جای مقادیر واقعی  $\{s_i\}_{i=2}^m$  به کار می بریم. بدین ترتیب آزمون زیر بهینه با " ناحیه بحرانی " زیر را خواهیم داشت:

$$|a_1^T x - \sum_{i=2}^m \hat{s}_i a_i^T a_i| > \epsilon \quad (19-4)$$

که در آن  $\{\hat{s}_i\}_{i=2}^m$  تخمین های موجود برای منابع مربوطه هستند که می توانند توسط یک حدس اولیه و یا از نتایج تکرار های قبلی ( در صورت پیاده سازی تکرارهای الگوریتم ) به دست آمده باشند. همچنین،  $\epsilon$  میزان سطح آستانه است.

آزمون فوق را می توان برای سایر منابع به طور مشابه تکرار کرد و وضعیت فعالیت هر یک از را مشخص کرد. این اساس کار " گام آشکارسازی " است. نتایج حاصل در این گام، به عنوان راهنما برای گام تقریب/تخمین مورد استفاده قرار می گیرد. بنابراین با شروع از یک جواب اولیه، با به کار بردن گام آشکارسازی و به دنبال آن ( یک تکرار ) گام تقریب/تخمین به تخمین جدیدی از بردار منبع رسید.

( با توجه به رابطه ۴-۱۹ ) برای هر  $1 \leq i \leq m$  تابعی به شکل زیر تعریف می کنیم:

$$g_i(x, \hat{s}) \triangleq |a_i^T(x - \sum_{j \neq i} \hat{s}_j a_j)| \quad (20-4)$$

این تابع را می توان " تابع فعالیت متناظر با منبع  $i$ -ام با داشتن تخمین  $\hat{s}$  " نامید. اگر  $\hat{s}^{(k)}$  تخمین موجود برای بردار منبع در پایان مرحله  $k$ ام باشد، گام آشکارسازی مربوط به مرحله  $k+1$ ام را می توان به صورت زیر نوشت:

○ گام آشکارسازی: اندیس های منابع فعال را به صورت زیر بیابید

$$\mathcal{I}_\alpha = \{1 \leq i \leq m : g_i(x, \hat{s}^{(k)}) > \epsilon^{(k+1)}\}$$

قابل ذکر است در عبارت بالا، با نوشتن  $\epsilon^{(k+1)}$  این امکان فراهم است که سطوح آستانه مرحله به مرحله تغییر کند. دنباله سطوح آستانه معمولاً به شکل یک دنباله نزولی انتخاب می شود [۱۹]. در عبارت فوق مجموعه

اندیس های متناظر با منابع "فعال تشخیص داده شده" با  $\mathcal{I}_\alpha$  نشان داده شده است. همچنین مجموعه اندیس های منابع غیرفعال با  $\mathcal{I}_l = \mathbb{Z} - \mathcal{I}_\alpha$  نشان داده می شود که تابع فعالیت ۴-۲۰ را می توان به صورت زیر نوشت:

$$g_i(x, \hat{s}) = |a_i^T(x - A\hat{s}) + \hat{s}_i| \quad (21-4)$$

با جمع آوری مولفه های  $g_i(x, \hat{s})$  در یک "تابع برداری فعالیت" مانند  $g(x, \hat{s})$ ، می توان فرم برداری - ماتریسی گام آشکارسازی را به شکل زیر بیان کرد:

$$g(x, \hat{s}) \triangleq |A^T(x - A\hat{s}) + \hat{s}| > \epsilon \quad (22-4)$$

توجه کنید که در عبارت فوق، منظور از اعمال عملگرهای قدر مطلق ( $|\cdot|$ ) و ترتیبی ( $>$ ) به یک بردار، اعمال آنها به صورت مولفه به مولفه<sup>۱</sup> است.

#### ۱-۶-۴ روشی دیگر از گام آشکارسازی

روش دیگری از گام آشکارسازی وجود دارد که در آن به جای مقایسه ی توابع فعالیت با یک سطح آستانه، درصدی از منابع که تابع فعالیت آن ها بزرگتر است، به عنوان منابع فعال انتخاب می شود [۱۹]. اگر اندیس گذاری منابع را به گونه ای تغییر دهیم که توابع فعالیت به ترتیب نزولی مرتب شوند، می توان این نسخه ی جدید گام آشکارسازی را به صورت زیر نشان داد:

$$\underbrace{g_1(x, \hat{s}^{(k)}) \geq g_2(x, \hat{s}^{(k)}) \geq \dots \geq g_{\rho m}(x, \hat{s}^{(k)})}_{\text{متناظر با منابع فعال}} \geq \underbrace{g_{\rho m+1}(x, \hat{s}^{(k)}) \geq \dots \geq g_m(x, \hat{s}^{(k)})}_{\text{متناظر با منابع غیرفعال}}$$

که در آن  $0 < \rho < 1$  درصدی از منابع است که می خواهیم توسط گام آشکارسازی فعال تشخیص داده شوند. توابع فعالیت  $g_1$  تا  $g_{\rho m}$  متناظر با منابع فعال و توابع فعالیت  $g_{\rho m+1}$  تا  $g_m$  متناظر با منابع غیرفعالند.

<sup>۱</sup>component wise

## ۷-۴ گام تقریب/تخمین

با مشخص شدن این که کدام یک از منابع فعال هستند، تخمین بردار منبع ساده تر خواهد شد. در این بخش دو روش برای تخمین جواب مسئله مطرح می شود که یکی با ملاحظه به دستگاه  $x = As$  در فضای  $x$  و دیگری با ملاحظه به دستگاه در فضای  $s$  است.

### ۱-۷-۴ نگاه در فضای $x$

در این روش برای یافتن تقریب  $s$ ، تابع هزینه ای مینیمم می شود که در بحث رویکرد بهینه مرتبط با " جستجوی تطابقی متعامد " مطرح شد. یعنی مسئله بدون قید زیر حل می شود:

$$\min_{s_\alpha} \|x - A_\alpha s_\alpha\|_2 \quad (23-4)$$

که در آن ستون های ماتریس  $A_\alpha$ ، اتم های فعال تشخیص داده شده اند. بنابراین مسئله فوق معادل یافتن بسط LS بردار  $x$  بر حسب اتم های فعال است. به بیان دیگر، این مسئله معادل یافتن تصویر بردار  $x$  روی زیرفضای تولید شده توسط اتم های  $\{a_i\}_{i \in \mathcal{I}_\alpha}$  است. این زیرفضا " زیرفضای فعالیت<sup>۱</sup> در فضای  $x$  " نامیده می شود [۱۹]. در مسئله حاضر، با فرض رتبه کامل بودن ماتریس  $A$  شرط یکتایی جواب، برقرار بودن رابطه  $k_\alpha \leq n$  خواهد بود؛ با توجه به این که ماتریس  $A_\alpha$ ،  $n \times k_\alpha$  است. در این صورت جواب مسئله را می توان به صورت

$$s_\alpha^* = (A_\alpha^T A_\alpha)^{-1} A_\alpha^T x \quad (24-4)$$

نوشت، که در این صورت تصویر مربوطه عبارت است از  $\hat{x} = A_\alpha s_\alpha^*$ . توجه کنید که تصویر، یعنی  $\hat{x}$  همواره وجود دارد و یکتاست؛ مستقل از این که اتم های  $\{a_i\}_{i \in \mathcal{I}_\alpha}$  مستقل خطی هستند یا خیر؛ اما نمایش این تصویر بر حسب اتم های مذکور، یعنی  $s_\alpha^*$ ، تنها در صورتی منحصر به فرد است که این اتم ها مستقل خطی باشند.

برای تکمیل جواب، بخش غیرفعال جواب را برابر صفر تخمین می زنیم که با توجه به فرض تنگ بودن کاملاً قابل توجیه است. پس تقریب بردار منبع در این روش به صورت زیر است:

$$\begin{cases} s_\alpha^* = (A_\alpha^T A_\alpha)^{-1} A_\alpha^T x \\ s_l^* = 0 \end{cases} \quad (25-4)$$

<sup>۱</sup>activity subspace



الگوریتم IDE که در گام تقریب از ۴-۲۳ (یا ۴-۲۵) استفاده می کند، با پسوند  $x$ - مشخص می شود.

#### ۲-۷-۴ نگاه در فضای $s$

در این روش برای یافتن تقریبی از  $s$ ، تابع هزینه ای مشابه آن چه در روش جستجوی زیرفضا به کار می رود، مینیمم سازی می شود. به بیان ریاضی، تقریب مورد نظر، جواب مسئله بهینه سازی زیر است:

$$\min \sum_{i \in \mathcal{I}_i} s_i^* \quad \text{s.t.} \quad x = As \quad (26-4)$$

که در آن،  $\mathcal{I}_i$  مجموعه اندیس های "غیرفعال تشخیص داده شده" توسط گام آشکار سازی است. فرض کنید پایه استاندارد  $\mathbb{R}_m$  با  $\{e - i\}_{i=1}^m$  نشان داده شود. در این صورت، در نتیجه ی اعمال گام آشکار سازی بردار منبع باید در زیرفضای تولید شده توسط  $\{e_i\}_{i \in \mathcal{I}_\alpha}$  قرار داشته باشد. این زیرفضا "زیرفضای فعالیت در فضای  $s$ " نامیده می شود [۱۹].

در این جا، برخلاف روش جستجوی زیرفضا، دیگر نیازی به جستجوی زیرفضای بهینه نیست؛ چون زیرفضای مذکور توسط گام آشکار سازی پیشنهاد می شود. همچنین مسئله بهینه سازی فوق، جوابی از  $x = As$  را انتخاب می کند که به زیرفضای فعالیت در فضای  $s$  "نزدیک تر" (با متر  $l_2$ ) است.

در فصل قبل روابط ۳-۱۵ جواب صریح مسئله ۴-۲۶ را برای حالتی که تعداد منابع غیرفعال یعنی  $k_i$ ، بیش از  $\max\{n, m - n\}$  است، به دست آورده شد:

$$\begin{cases} s_\alpha^* = (A_\alpha^T P A_\alpha)^{-1} A_\alpha^T P x \\ s_i^* = A_i^T P (x - A_\alpha s_\alpha^*) \end{cases} \quad (27-4)$$

که در آن  $P \triangleq (A_i A_i^T)^{-1}$ ؛ همچنین شرط  $k_i \geq \max\{n, m - n\}$  برای تضمین یکتایی جواب قرار داده شد. اگر این شرط برقرار نباشد، در حالت کلی جواب یکتا نیست؛ در این صورت، دیگر مسئله ۴-۲۶ برای به دست آوردن تقریبی از بردار منبع ( $s$ ) کافی نخواهد بود.

اکنون سعی می شود فرم ساده تری برای معادلات ۴-۲۷ یافت شود. ماتریس های  $A_i$  و  $A_\alpha$  به ترتیب  $n \times k_\alpha$  و  $n \times k_i$  هستند. تنگ بودن بردار منبع باعث می شود که  $k_\alpha$  غالباً بسیار کوچکتر از  $k_i$  باشد، یعنی  $k_\alpha \ll m$  و  $k_i = m - k_\alpha \lesssim m$  بنابراین برای به دست آوردن بخش فعال جواب ( $s_\alpha^*$ ) نیاز به حل یک دستگاه خطی کوچک از مرتبه  $k_\alpha \times k_\alpha$  است. در حالی که، یافتن بخش غیرفعال جواب ( $s_i^*$ ) نیازمند محاسباتی روی بردارها

و ماتریس های بزرگ از مرتبه  $k_i \times k_i$  (یعنی تقریباً  $m \times m$ ) است. یعنی بخش عمده ای از پیچیدگی محاسباتی معادلات ۴-۲۷ مربوط به محاسبه بخش غیرفعال جواب می شود. از آن جاکه انتظار می رود برای منابع تنک، بخش غیرفعال جواب با تقریب خیلی خوبی نزدیک به صفر باشد، می توان از محاسبه ی آن صرف نظر کرد و آن را برابر صفر تخمین زد؛ یعنی به جای معادلات ۴-۲۷، روابط زیر را جایگزین کرد:

$$\begin{cases} s_{\alpha}^* = (A_{\alpha}^T P A_{\alpha})^{-1} A_{\alpha}^T P x \\ s_l^* = 0 \end{cases} \quad (28-4)$$

به این ترتیب، بدون از دست دادن دقت زیاد محاسبات مربوطه ساده تر می شود [۱۹]. با مقایسه جواب روش اول یعنی روابط ۴-۲۵، با فرم ساده شده جواب اخیر یعنی روابط ۴-۲۸، ملاحظه می شود که این جواب یک ماتریس وزن  $P$  بیشتر دارد.

الگوریتم IDE که در گام تقریب/تخمین از روابط ۴-۲۸ استفاده می کند، با پسوند  $s$ -مشخص می شود. در پایان این بخش دو روشی که برای گام تقریب/تخمین مطرح شد، به صورت زیر خلاصه می شود:

○ گام تقریب/تخمین: به کمک نتایج گام آشکار سازی، بردار منبع ( $s$ ) را به یکی از دو روش زیر بیابید

$$(i) \min \sum_{i \in \mathcal{I}_l} s_i^{\dagger} \quad \text{s.t. } x = As \quad (\text{IDE-}s)$$

$$(ii) \min_{s_{\alpha}} \|x - A_{\alpha} s_{\alpha}\|_2 \quad (\text{IDE-}x)$$

## ۸-۴ جواب اولیه بردار منبع

زمانی که اطلاعات پیشین خاصی در مورد بردار منبع ( $s$ ) وجود ندارد، می توان از جواب اولیه  $\hat{s}^{(0)} = 0$  استفاده کرد؛ زیرا محتمل ترین جواب برای یک منبع تنک، برداری نزدیک به بردار صفر است. لذا این تخمین اولیه برای بسیاری از مولفه های بردار منبع ( $k_i$ ها)، نزدیک به مقدار واقعی خواهد بود. البته جواب اولیه صفر، جوابی از دستگاه  $x = As$  نیست؛ اما در طی اجرای الگوریتم به عملکرد خوبی می انجامد و مزیت عمده ی آن سادگی بیش از حد آن است [۱۹].

علاوه بر جواب اولیه  $\hat{s}^{(0)} = 0$  می توان از جواب حاصل از برخی روش های ساده ی موجود که به خودی خود کیفیت مطلوب ندارند نیز، استفاده کرد. یک نمونه از چنین جوابی ، جواب حاصل از کمینه سازی نرم  $l_2$  بردار منبع است، یعنی  $\hat{s}_M^{(0)} = A^T(AA^T)^{-1}x$ .

فرض کنید مقدار قدر مطلق تابع  $g(x, \hat{s})$  را با  $g'(x, \hat{s})$  نشان دهیم. در این صورت،

$$g'(x, \hat{s}_M^{(0)}) = A^T(x - AA^T(AA^T)^{-1}x) + A^T(AA^T)^{-1}x = \hat{s}_M^{(0)}$$

یعنی  $\hat{s}_M^{(0)}$  نقطه ثابت تابع  $g'(x, \hat{s})$  است.

## ۹-۴ نکات اصلی انتخاب سطح آستانه

در این بخش دو نکته برای انتخاب سطح آستانه شرح داده خواهد شد. انتخاب سطح آستانه بر مبنای تقریب جوابی از  $x = As$  است که مجموع مربعات منابع غیر فعال آن مینیمم باشد.

### ۱-۹-۴ کران های ضمنی

در این زیربخش سعی می شود با یک استدلال شهودی (که از مرجع [۱۹] استخراج شده) شرایطی که لازم (و کافی) است برقرار باشند تا جواب حاصل از اعمال گام آشکارسازی و گام تقریب/تخمین نزدیک به جواب اصلی شود ، بیان شوند. برای این هدف ، حالتی را در نظر بگیرید که در بردار منبع اصلی  $(s^*)$  ، منابع غیرفعال دقیقا برابر صفرند. همچنین فرض کنید که گام آشکارسازی حداقل منابع فعال را درست تشخیص دهد ؛ یعنی منابع واقعا فعال در میان منابع فعال تشخیص داده شده ، باشند؛ به عبارت دیگر  $\mathcal{I}_\alpha^* \subseteq \mathcal{I}_\alpha$ . یا به طور معادل ، فرض کنید منابع واقعا غیر فعال شامل منابع غیرفعال تشخیص داده شده اند؛ به عبارتی  $\mathcal{I}_\alpha \subseteq \mathcal{I}_\alpha^*$ . در این صورت بردار منبع اصلی  $(s^*)$  یک جواب مسئله بهینه سازی ۲۶-۴ (IDE-s) خواهد بود چون

$$\sum_{i \in \mathcal{I}_\alpha} (s_i^*)^2 \leq \sum_{i \in \mathcal{I}_\alpha^*} (s_i^*)^2 = 0$$

حال اگر جواب مسئله بهینه سازی مذکور یکتا باشد، می توان مطمئن بود که گام تقریب/تخمین واقعا جواب اصلی را می دهد. لذا این استدلال بر فرض ”آشکارسازی صحیح“ و ”یکتایی جواب“ گام تقریب/تخمین استوار است.

به طور شهودی با تعمیم این استدلال انتظار می رود الگوریتم در حالت هایی که منابع غیرفعال اصلی دقیقا صفر نیستند نیز همچنان خوب عمل کند.

همان طور که قبلا گفته شد، منظور از آشکارسازی صحیح تنها یافتن صحیح منابع فعال است؛ یعنی مشخص کردن صحیح منابع غیرفعال، تا جایی که به فرض "یکتایی جواب" لطمه نزنند، چندان اهمیتی ندارد. به عبارت دیگر، هدف آن است که احتمال آشکارسازی صحیح ( $P_{De}$ ) تا حد امکان نزدیک به یک باشد، اما احتمال هشدار بی مورد ( $P_{Fa}$ ) لزومی ندارد خیلی به صفر نزدیک شود. همین قدر که  $P_{Fa}$  از حدی کمتر باشد که فرض یکتایی جواب برقرار شود، کافی است. میزان  $P_{Fa}$  باید به اندازه ای باشد که فرض یکتایی جواب مسئله ۴-۲۶ یعنی  $k_i \geq \max\{n, m - n\}$  برقرار باشد. به این ترتیب، تعداد منابع غیرفعال تشخیص داده شده نباید از حدی کمتر باشد؛ به بیان دیگر، منابعی که اشتباها فعال تشخیص داده می شوند (یا همان "هشدار بی مورد") نباید خیلی زیاد باشند.

با توجه به مطالب گفته شده، ملاحظاتی برای کران های بالا و پایین ضمنی روی مقدار آستانه وجود دارد. سطح آستانه باید به اندازه کافی پایین باشد که منابع واقعا فعال درست تشخیص داده شوند (آشکارسازی صحیح)؛ از طرف دیگر باید به اندازه کافی بالا باشد که یکتایی جواب تضمین شود (حداقل  $\max\{n, m - n\}$  تا از منابع، غیرفعال تشخیص داده شوند). دو پارامتر مشخصا روی گستره ی مجاز سطح آستانه تاثیر دارند. یکی میزان تنک بودن است که می توان آن را با نسبت  $k_i^*/m$  (و در مورد منابع دارای توزیع مخلوط گوسی با پارامتر  $\pi$ ) سنجید؛ و دیگری نسبت ابعاد مسئله یعنی  $n/m$  است. هرچه تنک بودن و بیشتر و  $n/m$  به یک نزدیک تر باشد، مسئله ساده تر و محدوده مجاز برای سطح آستانه بزرگتر خواهد بود. از طرف دیگر، با کاهش تنک بودن و (یا) کاهش  $n/m$  مسئله دشوارتر و انتخاب محدوده مجاز سطح آستانه دشوارتر و با ملاحظه بیشتری خواهد بود. در ادامه با بررسی نتایج تجربی (صورت گرفته در مرجع [۱۹]) سطوح آستانه مناسب برای خانواده هایی از مسائل (برای مقادیر مشخصی از  $n/m$  و  $\pi$ ) معرفی خواهد شد.

به طور خلاصه دو فرض "آشکارسازی صحیح" و "یکتایی جواب" منجر به طرح دو کران ضمنی برای سطح آستانه می شوند. تا زمانی که این دو کران رعایت شوند، یک آشکارسازی تقریبی برای یافتن جواب کافی است. این، اساس همگرایی روش "آشکارسازی-تخمین پیایی" است. نکته دیگری که در مورد این الگوریتم وجود دارد وضعیت منابع فعالی است که به درستی تشخیص داده نمی شوند. در چنین حالتی انتظار می رود که جواب گام تقریب/تخمین از جواب اصلی دور شود؛ چون منابع فعال مربوطه به اشتباه جز، تابع هزینه  $\sum_{i \in I_i} s_i$  قرار گرفته و مینیمم می شوند. البته مقدار واقعی چنین منابعی طبیعتا کوچک است که اشتباها غیرفعال تشخیص داده می شوند. با این حال آزمایشاتی که در [۱۹] صورت گرفته است، به طور تجربی نشان می دهد الگوریتم (در غالب موارد) این توانایی را دارد که این اشتباه را در مراحل بعد اصلاح کند و در نهایت تمام منابع فعال را به

درستی به دست آورد.

## ۲-۹-۴ کران های صریح

در گام آشکارسازی نشان داده شد وضعیت فعالیت یک منبع را می توان با تابع فعالیت متناظر آن منبع، یعنی  $g_i(x, \hat{s})$ ، توسط یک سطح آستانه سنجید. تابع  $g_i(x, \hat{s})$  بر اساس تخمین موجود برای بردار منبع  $(\hat{s})$  و بردار مخلوط  $(x)$  تخمینی جدید از اندازه بردار منبع به دست می دهد. در صورتی که این تخمین به اندازه ی کافی بزرگ باشد (بزرگتر از سطح آستانه) منبع مربوطه فعال تشخیص داده می شود. هنگامی که تخمین موجود برای بردار منبع، همان بردار منبع اصلی باشد  $(\hat{s} = s^*)$  با توجه به تعریف تابع فعالیت (رابطه ۴-۲۰) داریم:

$$\begin{aligned} g_i(x, s^*) &= |a_i^T (\sum_{j=1}^m s_j^* a_j - \sum_{j \neq i} s_j^* a_j)| \\ &= |s_i^*| \end{aligned}$$

این رابطه نشان می دهد که تابع فعالیت در حالت ایده آل، برابر اندازه منبع مربوطه است. بنابراین استفاده از تابع فعالیت برای سنجش فعالیت یک منبع، بار دیگر توجیه می شود. همچنین این رابطه نشان می دهد اگر کران هایی روی اندازه منابع وجود داشته باشد، می توان آن کران ها را برای سطوح آستانه نیز به کار برد. به عبارت دیگر:

$$\circ \approx \min_i |s_i^*| \leq \epsilon \leq \max_i |s_i^*| \quad (۲۹-۴)$$

برای منابع تک مینیمم اندازه منابع تقریباً برابر صفر است. لذا کران پایین رابطه فوق اطلاعات چندانی را برای سطح آستانه به دست نمی دهد. از طرفی ماکزیمم اندازه منابع (غالبا) از پیش مشخص نیست، و همین کمیت است که نقش اساسی را در تعیین محدوده مجاز سطح آستانه دارد. لذا در غیاب بردار منبع سعی می شود این کمیت بر اساس  $x$  و  $A$  تخمین زده شود. بر این اساس، سه تخمین گر پیشنهاد شده در [۱۹] برای این کمیت را معرفی کرده و عملکرد آنها مورد بررسی قرار می گیرد. لازم به ذکر است  $\max_i |s_i^*|$ ، نرم بردار منبع اصلی است که به صورت  $\|s^*\|_\infty$  نیز نشان داده می شود.

رویکرد طبیعی برای یافتن تخمین گر کمیت  $\|s^*\|_\infty$ ، استفاده از جواب اولیه بردار منبعی است که در تکرار اول الگوریتم "آشکارسازی-تخمین پیاپی" مورد استفاده قرار می گیرد. بدین ترتیب، با توجه به جواب اولیه انتخاب شده برای الگوریتم، می توان تخمین های متفاوتی به دست آورد. اگر جواب اولیه را برابر صفر بگیریم

( $\hat{s}^{(0)} = 0$ ) تابع برداری فعالیت عبارت خواهد بود از

$$g(x, 0) = |A^T x| \quad (30-4)$$

که یک تخمین مناسب برای نرم  $l_\infty$  بردار منبع، نرم  $l_\infty$  همین بردار است. بنابراین تخمین گر اولی که مورد بررسی قرار می گیرد مقدار زیر است:

$$\|A^T x\|_\infty = \max_i |a_i^T x| \quad (31-4)$$

حال اگر جواب اولیه  $\hat{s}^{(0)} = A^T(AA^T)^{-1}x$  انتخاب شود، مقدار

$$\|A^T(AA^T)^{-1}x\|_\infty = \max_i |a_i^T(AA^T)^{-1}x| \quad (32-4)$$

تخمین گر دوم برای نرم  $l_\infty$  بردار منبع خواهد بود (جواب  $\hat{s}^{(0)} = A^T(AA^T)^{-1}x$  نقطه ثابت تابع فعالیت است). قابل ذکر است، بخش عمده محاسبات این دو تخمین گر در مرحله ابتدایی صورت می گیرد و تنها محاسبه اضافی، یافتن ماکزیمم بردار تخمین گر است. علاوه بر این دو تخمین گر از  $\|x\|_2$  نیز می توان به عنوان یک تخمین ساده از ماکزیمم اندازه بردار منبع استفاده کرد. برای حفظ تطابق این تخمین ( $\|x\|_2$ ) با مقدار واقعی لازم است با تابعی از  $n$  (یا  $m$ ) scale شود (مطالعه بیشتر در [۱۹]).

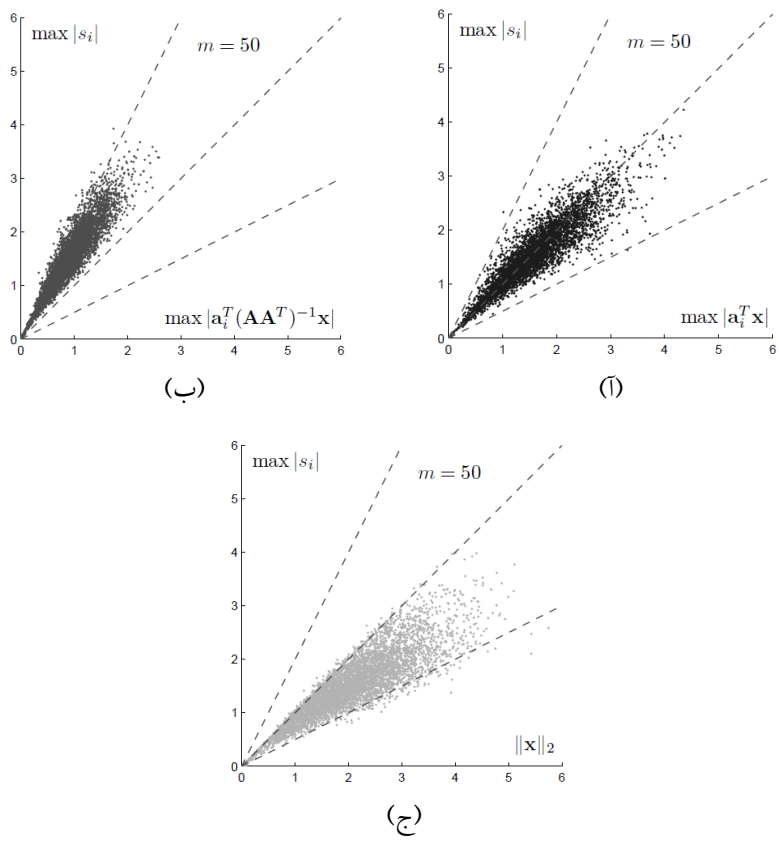
با توجه به مطالب گفته شده، تخمین گره های معرفی شده برای ماکزیمم اندازه بردار منبع، کران هایی برای کمیت مذکور هستند که به طور خلاصه می توان به صورت زیر نوشت:

$$\|s^*\|_\infty \lesssim K_1 \|A^T x\|_\infty \quad (33-4)$$

$$\|s^*\|_\infty \lesssim K_2 \|A^T(AA^T)^{-1}x\|_\infty \quad (34-4)$$

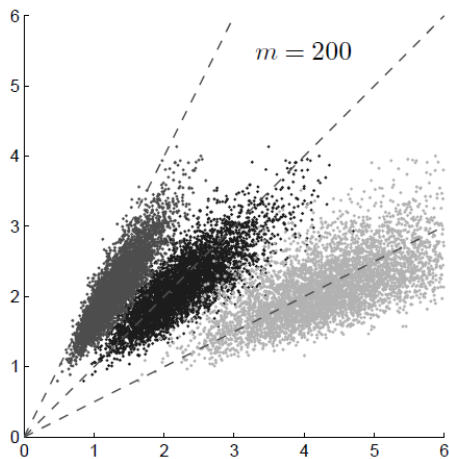
$$\|s^*\|_\infty \lesssim K_3(n) \|x\|_2 \quad (35-4)$$

فرض می کنیم که  $K_3(n) \propto n^{-\alpha}$  و  $\alpha$  از طریق آزمایش مشخص خواهد شد (متذکر می شویم تمامی آزمایشاتی که در ادامه در مورد کران های صریح برای سطح آستانه، مطرح می شوند از مرجع [۱۹] استخراج شده اند). به کمک نتایج تجربی می توان مفید بودن تخمین های فوق را در عمل بررسی کرد. برای این منظور آزمایشی بدین ترتیب صورت داده شده است: تعداد  $N=5000$  تحقق تصادفی دستگاه  $x = As$  تولید شده است؛ به این

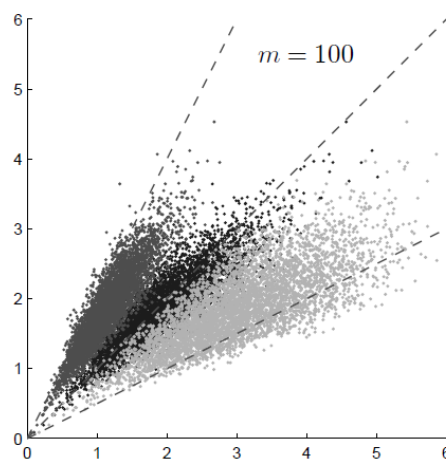


شکل ۴-۱: کران های صریح برای سطح آستانه (در ابعاد پایین): نمودارهای پراکنندگی ماکزیمم اندازه منابع  $(\|s\|_\infty)$  بر حسب هر یک از سه تخمین گر پیشنهادی برای  $N=5000$  تحقق تصادفی مسئله

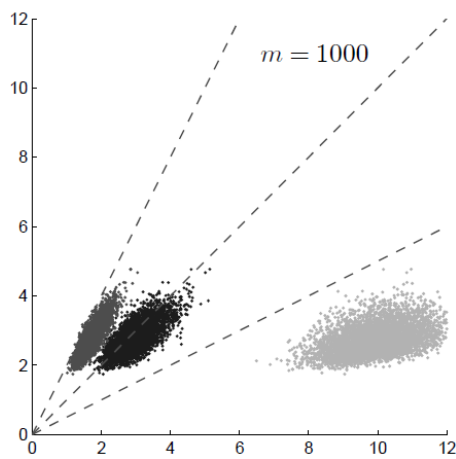
ترتیب که در هر یک از  $N$  تحقق، ماتریس  $A$  به شکل تصادفی با ستون هایی دارای توزیع یکنواخت روی کره ی واحد در  $\mathbb{R}^n$  انتخاب می شود. به همین ترتیب، بردار  $s$  هر بار به تصادف از یک توزیع مخلوط گوسی با پارامترهای  $(\pi_0 = 0.9, \sigma_0 = 0.1, \sigma_1 = 1)$  به دست می آید. سپس برای هر تحقق مقدار واقعی کمیت  $\|s\|_\infty$  (که در آن  $s$  بردار منبع اصلی است) و تخمین آن را به شکل نقطه ای در یک صفحه مشخص می شود. به عنوان مثال، برای تخمین اول، صفحه ی مورد نظر صفحه ی  $\|s\|_\infty$  بر حسب  $\|A^T x\|_\infty$  است و هر یک از  $N$  تحقق با توجه به مقادیر  $s$ ،  $x$  و  $A$  متناظر با نقطه ای در این صفحه خواهد بود. نمودار حاصل، برای حالت  $m = 50$  و  $n/m = 0.6$  در شکل ۴-۱ (آ) نشان داده شده است. در شکل خطوط  $y = x$  و  $y = 2x$  و  $y = \frac{1}{2}x$  نیز مشخص شده اند. همان طور که ملاحظه می شود، نقاط حول  $y = x$  تجمع کرده اند؛ به عبارت دیگر اگر خطی به نمودار نقاط مذکور تطبیق داده شود، شیب خط نزدیک به یک خواهد بود. همچنین ملاحظه می شود نقاط زیر خط  $y = 2x$  قرار دارند. به این ترتیب انتخاب  $2 \approx K_1$  در رابطه ۴-۳۳ انتخاب مناسبی است. شکل های ۴-۱ (ب و ج) نمودارهای مربوط به دو تخمین دیگر را در وضعیتی مشابه  $(n/m = 0.6, m = 50)$  نشان می دهند. همان طور که ملاحظه می شود، نمودار تخمین  $\|A^T (AA^T)^{-1} x\|_\infty$  تقریباً زیر خط  $y = 2x$  قرار می گیرد؛



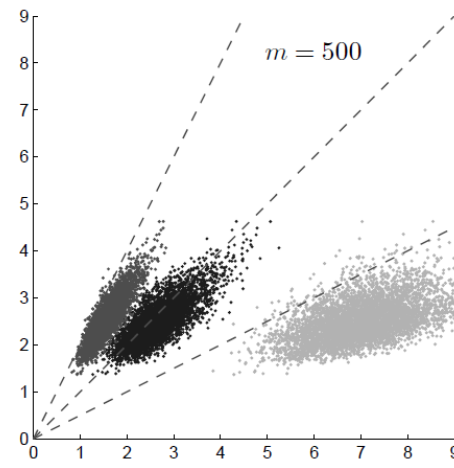
(ب)



(ا)



(د)

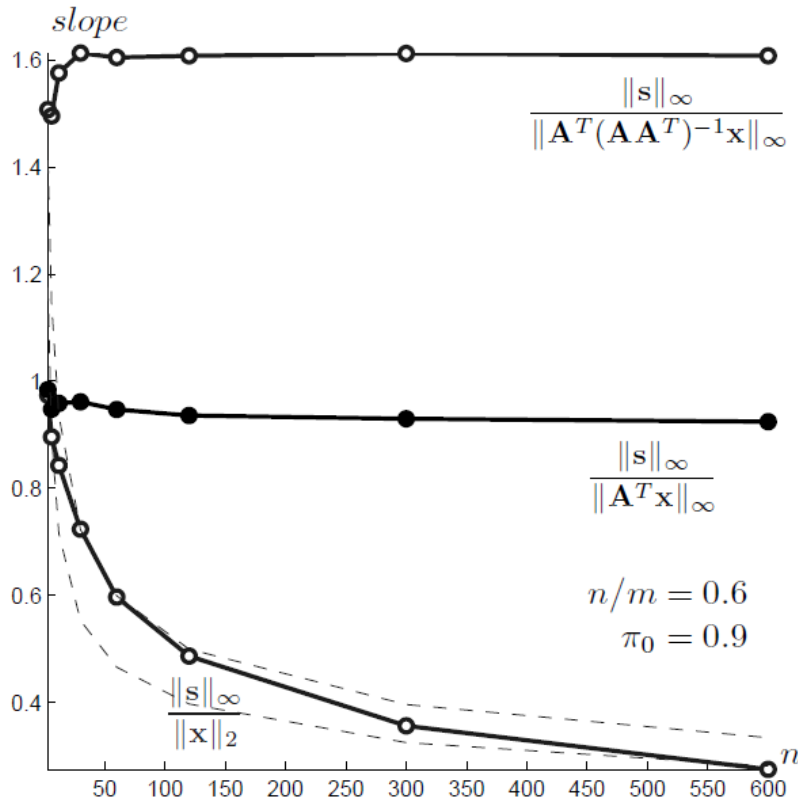


(ج)

شکل ۲-۴: کران های صریح برای سطح آستانه ( در ابعاد بالا) : نمودارهای پراکندگی ماکزیمم اندازه منابع  $(\|s\|_\infty)$  بر حسب سه تخمین گر پیشنهادی در ابعاد  $m = 100, 200, 500, 1000$

البته شیب خطی که به نمودار تطبیق می شود از یک بیشتر است. بنابراین می توان  $K_2$  را نیز تقریباً برابر ۲ گرفت ( $K_1 \approx 2$ ). وضعیت برای تخمین  $\|x\|_2$  متفاوت است. در واقع، برای  $m$  (یا  $n$ ) کوچک، تجمع نقاط مربوط به این تخمین کمابیش مشابه دو تخمین دیگر است. لذا در چنین حالت هایی می توان  $\|s\|_\infty$  با ضریب ثابتی از  $\|x\|_2$ ، مستقل از مقدار  $n$ ، تقریب زد؛ یعنی  $K_2(n) \approx 2$  (for small  $n$ ). با افزایش  $m$ ، نمودار مربوط به  $\|x\|_2$  از دئ نمودار دیگر فاصله می گیرد. حالت  $m = 50$  که در شکل ۴-۱ (ج) نشان داده شده است، یک وضعیت مرزی (نسبی) را نشان می دهد که در آن، این جدایی آغاز شده است. برای بررسی رفتار تخمین گرها در ابعاد بالا، در شکل های ۲-۴ ( (ا) تا (د) ) نمودارهای سه تخمین گر برای ابعاد  $m = 100, 200, 500, 1000$  نشان داده شده اند. در تمام حالت ها  $n/m = 0.6$  فرض شده است. در هر شکل، نمودارهای سه تخمین





شکل ۳-۴: کران های صریح برای سطح آستانه (رفتار حدی): نسبت ماکزیمم اندازه منابع  $(\|s\|_\infty)$  به تخمین آن، بر حسب  $n$  به عنوان معیاری از بعد مسئله، برای سه تخمین گر پیشنهادی

گر برای یک مقدار مشخص  $m$  در کنار یکدیگر آورده شده و برای تمایز آن ها از رنگ های متفاوتی استفاده شده است. رنگ ها مشابه رنگ های به کار رفته در شکل ۴-۱ هستند. ملاحظه می شود نمودارهای دو تخمین گر  $\|A^Tx\|_\infty$  و  $\|A^T(AA^T)^{-1}x\|_\infty$  وضعیت مشابهی را در تمام ابعاد دارند. به خصوص، شیب خطی که به نمودار آن ها تطبیق می یابد، بسیار به کندی تغییر می کند و چنین به نظر می رسد که به مقدار ثابت میل می کند. در مقابل، شیب خط تطبیق یافته به نمودار تخمین گر  $\|x\|_2$ ، با افزایش بعد به سرعت کاهش می یابد و نمودار مربوطه به وضوح از دو نمودار دیگر فاصله می گیرد.

در شکل ۳-۴ شیب خط تطبیق یافته به سه نمودار بر حسب  $n$  رسم شده است. منحنی های حاصل، مشاهدات قبل را تایید می کند. شیب مربوطه به دو تخمین اول به ترتیب به ۰.۹۵ و ۱.۰۶ میل می کند. در مورد تخمین سوم، یعنی  $\|x\|_2$ ، شیب با افزایش  $n$  به سرعت کاهش می یابد. در شکل منحنی های  $n^{-0.2} + n^{-0.45}$  و  $n^{-0.2} + n^{-0.9}$  نیز به شکل خط چین رسم شده اند. ملاحظه می شود که منحنی شیب مربوط به  $\|x\|_2$ ، حداقل برای  $n < 600$  میان این دو منحنی قرار می گیرد. این پیشنهاد می کند که  $(K_3(n) \approx 2n^{-0.2})$  حداقل روی گستره  $n < 600$  (یا  $m < 1000$ ) انتخاب مناسبی است. با وجود این که آزمایش فوق برای حالت  $(n/m = 0.6, \pi_0 = 0.9)$  انجام شده است، نتایج کمابیش برای مقادیر دیگر  $\pi_0$  و  $n/m$  نیز صادق اند

[۱۹]. تفاوت در این خواهد بود که اعداد و ارقام به دست آمده ، مثلا برای مقادیر حدی شیب نمودارها، ممکن است تغییر کند. در این میان دو تخمین گر اول تا حدود زیادی نسبت به تغییرات پارامترهای مسئله مقاوم اند؛ برخلاف تخمین گر  $\|x\|_2$  که ضریب مورد نیاز آن ، یعنی  $K_3(n)$ ، کاملا وابسته به پارامترها مسئله ( مخصوصا ابعاد مسئله) است.

در مجموع بر اساس آزمایشات فوق می توان گفت که دو تخمین  $\|A^T x\|_\infty$  و  $\|A^T(AA^T)^{-1}x\|_\infty$  هر دو ، تخمین های مناسبی برای ماکزیمم اندازه منابع ( و در نتیجه برای کران صریح سطح آستانه ) هستند.

مراجع

## فهرست منابع

- [1] Chen, Scott Shaobing, Donoho, David L, and Saunders, Michael A. Atomic decomposition by basis pursuit. *SIAM review*, 43(1):129–159, 2001.
- [2] Elad, Michael and Bruckstein, Alfred M. A generalized uncertainty principle and sparse representation in pairs of bases. *IEEE Transactions on Information Theory*, 48(9):2558–2567, 2002.
- [3] Donoho, David L and Elad, Michael. Optimally sparse representation in general (nonorthogonal) dictionaries via  $\ell_1$  minimization. *Proceedings of the National Academy of Sciences*, 100(5):2197–2202, 2003.
- [4] Donoho, David Leigh. Sparse components of images and optimal atomic decompositions. *Constructive Approximation*, 17(3):353–382, 2001.
- [5] Aharon, Michal, Elad, Michael, Bruckstein, Alfred, et al. K-svd: An algorithm for designing overcomplete dictionaries for sparse representation. *IEEE Transactions on signal processing*, 54(11):4311, 2006.
- [6] Dabov, Kostadin, Foi, Alessandro, Katkovnik, Vladimir, and Egiazarian, Karen. Image denoising by sparse 3-d transform-domain collaborative filtering. *IEEE Transactions on image processing*, 16(8):2080–2095, 2007.
- [7] Capel, David. Image mosaicing. in *Image Mosaicing and super-resolution*, pp. 47–79. Springer, 2004.
- [8] Chan, Tak Ming and Zhang, Junping. An improved super-resolution with manifold learning and histogram matching. in *International Conference on Biometrics*, pp. 756–762. Springer, 2006.
- [9] Wright, John, Yang, Allen Y, Ganesh, Arvind, Sastry, S Shankar, and Ma, Yi. Robust face recognition via sparse representation. *IEEE transactions on pattern analysis and machine intelligence*, 31(2):210–227, 2009.

- [10] Elad, Michael. *Sparse and Redundant Representations: From Theory to Applications in Signal and Image Processing*. Springer Publishing Company, Incorporated, 1st ed. , 2010.
- [11] Donoho, David L, Elad, Michael, and Temlyakov, Vladimir N. Stable recovery of sparse overcomplete representations in the presence of noise. *IEEE Transactions on information theory*, 52(1):6–18, 2006.
- [12] Mallat, Stéphane and Zhang, Zhifeng. Matching pursuit with time-frequency dictionaries. tech. rep., Courant Institute of Mathematical Sciences New York United States, 1993.
- [13] Pati, Yagyensh Chandra, Rezaifar, Ramin, and Krishnaprasad, Perinkulam Sambamurthy. Orthogonal matching pursuit: Recursive function approximation with applications to wavelet decomposition. in *Signals, Systems and Computers, 1993. 1993 Conference Record of The Twenty-Seventh Asilomar Conference on*, pp. 40–44. IEEE, 1993.
- [14] Amini, Arash Ali, Babaie-Zadeh, Massoud, and Jutten, Christian. A new approach for sparse decomposition and sparse source separation. in *Signal Processing Conference, 2006 14th European*, pp. 1–5. IEEE, 2006.
- [15] Amini, Arash Ali, Babaie-Zadeh, Massoud, and Jutten, Christian. A fast method for sparse component analysis based on iterative detection-estimation. in *Bayesian Inference and Maximum Entropy Methods in Science and Engineering(AIP Conference Proceedings Volume 872)*, vol. 872, pp. 123–130. Citeseer, 2006.
- [16] Candes, Emmanuel and Romberg, Justin. Sparsity and incoherence in compressive sampling. *Inverse problems*, 23(3):969, 2007.
- [17] Tropp, Joel A and Wright, Stephen J. Computational methods for sparse solution of linear inverse problems. *Proceedings of the IEEE*, 98(6):948–958, 2010.
- [18] Luenberger, DG and Ye, Y. *Linear and nonlinear programming.*, 2008.
- [۱۹] امینی، آرش علی. جداسازی منابع تنک. پایان نامه کارشناسی ارشد، دانشکده برق، دانشگاه صنعتی شریف، ۱۳۸۵.
- [20] DeVore, Ronald A and Temlyakov, Vladimir N. Some remarks on greedy algorithms. *Advances in computational Mathematics*, 5(1):173–187, 1996.
- [۲۱] صادقی، مصطفی. نمایش تنک و کاربرد آن در نویز زدایی تصاویر. پایان نامه کارشناسی ارشد، دانشکده برق، دانشگاه صنعتی شریف، ۱۳۹۱.

- [22] Needell, Deanna and Tropp, Joel A. Cosamp: Iterative signal recovery from incomplete and inaccurate samples. *Applied and computational harmonic analysis*, 26(3):301–321, 2009.
- [23] Woodbury, Max A. Inverting modified matrices. *Memorandum report*, 42(106):336, 1950.
- [24] Scharf, Louis L and Demeure, Cédric. *Statistical signal processing: detection, estimation, and time series analysis*, vol. 63. Addison-Wesley Reading, MA, 1991.
- [25] Poor, H Vincent. *An introduction to signal detection and estimation*. Springer Science & Business Media, 2013.

# پیوست آ

برنامه آ-۱: کد متلب الگوریتم جست و جوی تطابقی MP

```

% define a continuous function                                ۱
function [xMP,POZ] = MP(A,b,thrMP)                          ۲
m = size(A,2);                                            ۳
r=b;                                                       ۴
xMP=zeros(m,1);                                          ۵
POZ = [];                                                ۶
iter =1;                                                 ۷
%while r'*r>thrMP,                                        ۸
while norm(r,1)>thrMP                                     ۹
    Z=abs(A'*r);                                         ۱۰
    posZ=find(Z==max(Z),1);                             ۱۱
    xMP(posZ)=xMP(posZ)+A(:,posZ)'*r;                   ۱۲
    % [A(:,posZ)'*r norm(A*xMP-b)]                       ۱۳
    r=r-A(:,posZ)*A(:,posZ)'*r;                         ۱۴
    POZ(iter) = posZ;                                    ۱۵
    iter = iter+1;                                       ۱۶
end                                                       ۱۷
[sum(xMP>.001),norm(xMP,1)]                              ۱۸

```

برنامه آ-۲: کد متلب الگوریتم مینیمم سازی نرم  $l_1$

```

% define a continuous function                                ۱
function xf = norm1MZ(A,b,thrMP)                          ۲
% Mehdi Zaferanieh                                        ۳
% min ||x||1 s.t. ||Ax-b||1<eps                          ۴
% m=3; n=2; eps=1; %initial values                      ۵
%A=[1 2 3 ; 1 4 5]; b=[0;10]; %initial matrix and right hand side ۶
    values;
m = size(A,1);                                           ۷
n = size(A,2);                                           ۸
Bn=[eye(n),-eye(n)]; %extreme directions for ||x||1;   ۹
Bm=[eye(m),-eye(m)]; %extreme directions for ||Ax-b||1; ۱۰
Aeqx=[eye(n),-Bn,zeros(n,2*m)]; %Aeq corresponding to equality $x=\sum_{i=1}^n \lambda_i b_i$; ۱۱
beqx=zeros(n,1); %beq corresponding to equaltiy $x=\sum_{i=1}^n \lambda_i b_i$; ۱۲

```

```

Aeqc=[A,zeros(m,2*n),-Bm]; %Aeq corresponding to equality $Ax-b=\sum_{i=1}^m \lambda'_i b_i$; ۱۳
beqc=b; %beq corresponding to equality $Ax-b=\sum_{i=1}^m \lambda'_i b_i$; ۱۴
Aeq=[Aeqx;Aeqc]; %The main Aeq corresponding to all constraints; ۱۵
beq=[beqx;beqc]; % The main righ-hand-side values; ۱۶
cx=zeros(1,n); % The ceofficient of variables x; ۱۷
cli=ones(1,2*n); % The ceofficient of extreme directions for $||\Delta x||_1$; ۱۸
clip=zeros(1,2*m); % The ceofficient of extreme directions for $||\Delta x-b||_1$; ۱۹
lb=zeros(n+2*m+2*n,1); % The lower bound for all variables; ۲۰
f=[cx,cli,clip]; %The main cost ceofficient; ۲۱
Ainq=[zeros(1,n),zeros(1,2*n),ones(1,2*m)]; %The constraint $||AX-b||_1 \le \epsilon$; ۲۲
eps;
binq=[thrMP]; %The given threshold ۲۳
[xllp,fval] = linprog(f,Ainq,binq,Aeq,beq,lb,[]); %The initial ۲۴
obtained solution
for i=1:n ۲۵
    xf(i)=xllp(i); ۲۶
end ۲۷
xf = xf'; ۲۸
[sum(xf>.001),norm(xf,1)] ۲۹

```

### برنامه آ-۳: کد متلب الگوریتم آشکارسازی تخمین بیای IDE

```

% define a continuous function ۱
function [xide,POZ] = IDE(A,b,thrMP) ۲
m = size(A,2); ۳
r=b; ۴
xide=zeros(m,1); ۵
POZ = []; ۶
k = 0; ۷
while norm((A*xide-b),1)>thrMP ۸
    g = abs(A'*(b-A*xide)+xide); ۹
    POZ = (g>.6*2^(-k)); ۱۰
    xide(POZ)= pinv(A(:,POZ))*b; ۱۱
    xide(~POZ)= 0; ۱۲
    k = k+1; ۱۳
end ۱۴
[sum(xide>.001),norm(xide,1)] ۱۵

```

### برنامه آ-۴: کد متلب برنامه مقایسه سه روش MP,LP,IDE

```

% define a continuous function ۱
xMP=[];xMZ=[];xdcM=[];xide=[]; ۲
n = 480; ۳
m = 1200; ۴
A = randn(n,m); ۵
%b=[1;2]; ۶
W = sum(A.*A).^-.5; ۷
for k = 1:m ۸

```



```

    A(:,k) = A(:,k)/W(k);
end
%
S = round(m/10) ;
% Generate a test signal of cardinality S
x = zeros(m,1);
pos = randperm(m);
x(pos(1:S)) = sign(randn(S,1)).*(1+rand(S,1));%b ra khodeman tolid
    kardim
b = A*x;

%%
thrMP =1e-3;

tic
for i=1:5
    xMP = MP(A,b,thrMP);
end
mpTime = toc/5;
tic
for i=1:5
    xMZ = norm1MZ(A,b,thrMP);
end
bnTime = toc/5;
tic
for i=1:5
    xide = IDE(A,b,thrMP);
end
ideTime = toc/5;

xdcm=xdcm';
s1=0;s2=0;s3=0;
s1=norm(xMP,1);s2=norm(xMZ,1);s3=norm(xdcm,1);
% for i=1:size(xMP,1)
% s1=abs(xMP(i))+s1;% norm1 of the solution by greedy
% s2=abs(xMZ(i))+s2;% norm1 of the solution by block norm
% s3=abs(xdcm(i))+s3;% norm1 of the solution by columngeneration
% end
disp('Errors of different approaches:')
disp('      MP      BlockNorm      IDE')
disp([norm(A*xMP-b,1) norm(A*xMZ-b,1) norm(A*xide-b,1)])% norm(A*xdcm-
,1])

disp('      MP runTime      BN runTime      IDE runTime')% dcm runTime
disp([mpTime bnTime ideTime]);%dcmTime])

disp('Time: MP/MP      BN/MP      IDE/mp ')
disp([mpTime bnTime ideTime ]./mpTime)
%norm(xmp,1)

```

برنامه آ-۵: کد متلب الگوریتم جست و جوی تطابقی برای یک مثال خاص OMP

% define a continuous function

<i>% MP with noise and bordare amood</i>	۲
	۳
n=30; m=200;	۴
A=randn(n,m);	۵
W=sum(A.*A).^0.5;	۶
for k=1:1:m,	۷
A(:,k)=A(:,k)/W(k);	۸
end;	۹
%%	۱۰
S =5;	۱۱
<i>% Generate a test signal of cardinality S</i>	۱۲
x=zeros(m,1);	۱۳
pos=randperm(m);	۱۴
x(pos(1:S))=sign(randn(S,1)).*(1+rand(S,1));	۱۵
noise = rand(n,1)/1000;	۱۶
b=A*x ;%+ noise;	۱۷
thrOMP=1e-1;	۱۸
%%	۱۹
r=b;	۲۰
xOMP=zeros(m,1);	۲۱
SS = [];	۲۲
iter = 0;	۲۳
while r'*r>thrOMP,	۲۴
Z=abs(A'*r);	۲۵
posZ=find(Z==max(Z));	۲۶
SS=sort([SS,posZ(1)]);	۲۷
xOMP(SS)=pinv(A(:,SS))*b;	۲۸
r=b-A(:,SS)*xOMP(SS);%b-Ax_k	۲۹
iter = iter+1;	۳۰
end;	۳۱
iter	۳۲
sum(xOMP~=0)	۳۳
norm(A*xOMP-b)	۳۴

### برنامه آ-۶: کد متلب الگوریتم جست و جوی تطابقی نمونه برداری فشرده CoSaMP

<i>% define a continuous function</i>	۱
function [x,r,normR,residHist, errHist] = CoSaMP( A, b, k, errFcn, opts	
)	
<i>% x = CoSaMP( A, b, k )</i>	۳
<i>% uses the Compressive Sampling Matched Pursuit (CoSaMP) algorithm</i>	۴
<i>% (see Needell and Tropp's 2008 paper <a href="http://arxiv.org/abs/0803.2392">http://arxiv.org/abs/0803.2392</a>)</i>	
)	
<i>% to estimate the solution to the equation</i>	۶
<i>%     b = A*x     (or b = A*x + noise )</i>	۷
<i>% where there is prior information that x is sparse.</i>	۸
if nargin < 5, opts = []; end	۹
if ~isempty(opts) && ~isstruct(opts)	۱۰
error('"opts" must be a structure');	۱۱
end	۱۲
	۱۳
function out = setOpts( field, default )	۱۴
if ~isfield( opts, field )	۱۵
opts.(field) = default;	۱۶
end	۱۷

```

    out = opts.(field);
end
printEvery = setOpts( 'printEvery', 1000 );
maxiter    = setOpts( 'maxiter', 1000 );
normTol    = setOpts( 'normTol', 1e-10 );
cg_tol     = setOpts( 'cg_tol', 1e-6 );
cg_maxit   = setOpts( 'cg_maxit', 20 );
% Allow some synonyms
cg_tol     = setOpts( 'LSQR_tol', cg_tol );
cg_maxit   = setOpts( 'LSQR_maxit', cg_maxit );
HSS        = setOpts( 'HSS', false );
TWO_SOLVES = setOpts( 'two_solves', false );
addK       = round(setOpts( 'addK', 2*k ));
support_tol = setOpts( 'support_tol', 1e-10 );

if nargin < 5 || isempty(printEvery)
    printEvery = round(k,maxiter);
end

if nargin < 4
    errFcn = [];
elseif ~isempty(errFcn) && ~isa(errFcn,'function_handle')
    error('errFcn input must be a function handle (or leave the input
        empty)');
end

if iscell(A)
    LARGESCALE = true;
    Af = A{1};
    At = A{2};    % we don't really need this...
else
    LARGESCALE = false;
    Af = @(x) A*x;
    At = @(x) A'*x;
end

% -- Intitialize --
% start at x = 0, so r = b - A*x = b
r = b;
Ar = At(r);
N = size(Ar,1);    % number of atoms
M = size(r,1);    % size of atoms
if k > M/3
    error('K cannot be larger than the dimension of the atoms');
end
x = zeros(N,1);
ind_k = [];

% indx_set = zeros(k,1); % created on-the-fly
% A_T = zeros(M,k);
residHist = zeros(k,1);
errHist = zeros(k,1);

```

```

fprintf('Iter, |T|, Resid');
if ~isempty(errFcn)
    fprintf(' Error');
end
if LARGESCALE
    fprintf(' LSQR iterations and norm(residual)');
end
fprintf('\n');

if LARGESCALE

% if exist( 'lsqr_wrapper','file')
% LSQR_ALG = @lsqr_wrapper;
% % This is Stephen's wrapper (designed to imitate Matlab's
% % syntax ) to the version of LSQR that you can get
% % at http://www.stanford.edu/group/SOL/software/lsqr/matlab
lsqr.m
% % This version of LSQR is better than Matlab's implementation
.
% elseif exist( 'lsqr','file' )
% LSQR_ALG = @lsqr; % Matlab's version
% else
% disp('You need to install LSQR! Download it from:');
% disp('http://www.stanford.edu/group/SOL/software/lsqr/matlab
lsqr.m');
% error('Need to have working copy of LSQR');
% end

% Unfortunately, the Stanford group's version doesn't make it easy
% to provide a starting value, so we will stick with Matlab's
version.

LSQR_ALG = @(RHS,Afcn,x0) lsqr(Afcn,RHS,cg_tol,cg_maxit,[],[],x0
);
end

for kk = 1:maxiter

% -- Step 1: find new index and atom to add
y_sort = sort( abs(Ar),'descend');
cutoff = y_sort(addK); % addK is typically 2*k
cutoff = max( cutoff, support_tol );
ind_new = find( abs(Ar) >= cutoff );

% -- Merge:
T = union( ind_new, ind_k );

% -- Step 2: update residual
if HSS
    RHS = r; % where r = b - A*x, so we'll need to add in "x"
later
x_warmstart = zeros(length(T),1);

```

```

else
    RHS      = b;
    x_warmstart  = x(T);
end

% -- solve for x on the support set "T"
if LARGESCALE
    % Could do CG on the normal equations...
    % Or, use LSQR:
    x_T = LSQR_ALG(Afcn,RHS,cg_tol,cg_maxit);

    % use an initial guess to warm-start the solver
    Afcn = @(x,mode) partialA( N, T, Af, At, x, mode );
    [x_T,flag,relres,CGiter] = LSQR_ALG(RHS,Afcn,x_warmstart);
else
    x_T = A(:,T)\RHS; % more efficient; equivalent to pinv when
    % underdetermined.
    x_T = pinv( A(:,T) )*RHS;
end

if HSS
    % HSS variation of CoSaMP
    x_new      = zeros(N,1);
    x_new(T)   = x_T;
    x          = x + x_new; % this is the key extra step in HSS
    cutoff     = findCutoff( x, k );
    x          = x.*( abs(x) >= cutoff );
    ind_k      = find(x);

    if TWO_SOLVES
        if LARGESCALE
            Afcn = @(x,mode) partialA( N, ind_k, Af, At, x, mode );
            [x_T2,flag,relres,CGiter] = LSQR_ALG(b,Afcn,x(ind_k));
            % not using "r", just using "b"
        else
            x_T2      = A(:,ind_k)\b;
        end
        x( ind_k ) = x_T2;
    end

    % update r
    r_old = r;
    r      = b - Af(x);
else
    % Standard CoSaMP
    % Note: this is implemented *slightly* more efficiently
    % that the HSS variation

    % Prune x to keep only "k" entries:
    cutoff = findCutoff(x_T, k);
    Tk     = find( abs(x_T) >= cutoff );
    % This is assuming there are no ties. If there are,
    % from a practical standpoint, it probably doesn't

```

```

%   matter much what you do. So out of laziness, we don't   187
    worry about it.                                          188
ind_k   = T(Tk);                                           189
x       = 0*x;                                             190
x( ind_k ) = x_T( Tk );                                    191
                                                         192
if TWO_SOLVES                                             193
    if LARGESCALE                                         194
        Afcn = @(x,mode) partialA( N, ind_k, Af, At, x, mode ); 195
        [x_T2,flag,relres,CGiter] = LSQR_ALG(b,Afcn,x(ind_k)); 196
    else                                                  197
        x_T2   = A(:,ind_k)\b;                             198
    end                                                  199
    x( ind_k ) = x_T2;                                    200
end                                                    201
                                                         202
% Update x and r                                          203
r_old   = r;                                             204
if LARGESCALE                                           205
    r    = b - Af( x );                                   206
else                                              207
    % don't do a full matrix-vector multiply, just use
      necessary columns
    r    = b - A(:,ind_k)*x_T( Tk );                     208
end                                                    209
end                                                    210
                                                         211
% -- Print some info --                                  212
PRINT   = ( ~mod( kk, printEvery ) || kk == maxiter );  213
normR   = norm(r);                                       214
STOP    = false;                                         215
if normR < normTol || norm( r - r_old ) < normTol      216
    STOP    = true;                                       217
    PRINT   = true;                                       218
end                                                    219
                                                         220
if ~isempty(errFcn)                                       221
    er = errFcn(x);                                       222
    errHist(kk) = er;                                     223
end                                                    224
if PRINT                                                  225
    fprintf('%4d, %4d, %.2e', kk, length(T), normR);     226
    if ~isempty(errFcn)                                    227
        fprintf(' , %.2e',er);                          228
    end                                                  229
    if LARGESCALE                                         230
        fprintf(' , %d, %.2e', CGiter,relres);          231
    end                                                  232
    fprintf('\n');                                        233
end                                                    234
                                                         235
residHist(kk) = normR;                                    236
                                                         237

```

```

if STOP
    disp('Reached stopping criteria');
    break;
end

if kk < maxiter
    Ar = At(r); % prepare for next round
end

end

end % -- end of main function

function tau = findCutoff( x, k )
% finds the appropriate cutoff such that after hard-thresholding,
% "x" will be k-sparse
x = sort( abs(x), 'descend' );
if k > length(x)
    tau = x(end)*.999;
else
    tau = x(k);
end
end

% for LSQR, PCG, etc.
function z = partialA( N, T, Af, At, x, mode )
% Multiplies A_T(x) or A_T'(x) (the transpose)
% where _T denotes restriction to the set "T"

switch lower(mode)
    case 'notransp'
        % zero-pad:
        y = zeros(N,1);
        y(T) = x;
        z = Af(y);

    case 'transp'
        y = At(x);
        % truncate:
        z = y(T);
end

end % -- end of subfunction

```

# واژه‌نامه فارسی به انگلیسی

Atom	اتم
Sparse Optimization	بهینه سازی تنک
Convex Optimization	بهینه سازی محدب
Atomic Decomposition	تجزیه اتمی
Sparse Decomposition	تجزیه تنک
Combinatorial Search	جست و جوی ترکیبیاتی
Full Rank	رتبه کامل
Underdetermined	فرومعیین
Sparse Coding	کدینگ تنک
Convex Set	مجموعه محدب
Dictionary Learning	یادگیری دیکشنری



# واژه‌نامه انگلیسی به فارسی

Atom . . . . .	اتم
Atomic Decomposition . . . . .	تجزیه اتمی
Combinatorial Search . . . . .	جست و جوی ترکیبیاتی
Convex Optimization . . . . .	بهینه سازی محدب
Convex Set . . . . .	مجموعه محدب
Dictionary Learning . . . . .	یادگیری دیکشنری
Full Rank . . . . .	رتبه کامل
Sparse Coding . . . . .	کدینگ تنک
Sparse Decomposition . . . . .	تجزیه تنک
Sparse Optimization . . . . .	بهینه سازی تنک
Underdetermined . . . . .	فرومعیین

در این فایل، عنوان پایان نامه، مشخصات خود و چکیده پایان نامه را به انگلیسی، وارد کنید.

Hakim Sabzevari University  
An Outline of MSc. Thesis



Surname: Bakhshali	Name: Mahboube	Student No.: 9413133093
Supervisor: Dr. Amintoosi		
Advisor: Dr. Rafiei		
Faculty of Mathematics and Computer Science	Applied Mathematics	Operational Research
Title of thesis: The subspace pursuit method in sparse optimization		
Keywords: sparse optimization , Sparse Decomposition, sparse representation, Blind source separation		

**Abstract:** In this paper, We introduce an approach for sparse decomposition, based on a geometrical interpretation of sparsity. By sparse decomposition we mean finding sufficiently sparse solutions of underdetermined linear systems of equations. The proposed algorithm is based on minimizing a family of quadratic forms, each measuring the distance of the solution set of the system to one of the coordinate subspaces (i.e. coordinate axes, planes, etc.).

Furthermore, we introduce an iterative algorithm for Sparse Component Analysis (SCA). The algorithm, which called Iterative Detection-Estimation (IDE), is essentially a method to find sufficiently sparse solutions of underdetermined linear systems of equations. In the SCA context, this solves the source separation part of the problem. Each iteration of IDE consists of two steps. In the detection step, starting with a previously known estimate of the sparse solution vector, we detect which components of the solution are (possibly) active, i.e., having a considerable value. Then, in the estimation step, we compute the new estimate by finding a solution of the system which is the closest to the subspace specified by the detection step. This is called projection into the activity subspace.



**Hakim Sabzevari University**  
**Faculty of Mathematics and Computer Science**

**A Thesis Submitted in Partial Fulfilment of the Requirement for the  
Degree of Master of Science in Applied Mathematics**

# **The subspace pursuit method in sparse optimization**

**Supervisor:**  
**Dr. Amintoosi**

**Advisor:**  
**Dr. Rafiei**

**By:**  
**Mahboube Bakhshali**

**September 2018**