



دانشگاه صنعتی شریف

دانشکده‌ی مهندسی کامپیوتر

پایان نامه به عنوان تحقق بخشی از شرایط دریافت درجه‌ی کارشناسی ارشد
گرایش مهندسی نرم افزار

مدل پیش بینی خطا مبتنی بر معیارهای جهش

نگارش

علی محبی

استاد راهنما

دکتر حسن میریان

شهریور ۱۳۹۷

بِسْمِ اللَّهِ الرَّحْمَنِ الرَّحِيمِ

[این صفحه آگاهانه خالی گذاشته شده است.]

تصویب نامه

به نام خدا
دانشگاه صنعتی شریف
دانشکده‌ی مهندسی کامپیوتر

پایان نامه‌ی کارشناسی ارشد

عنوان: مدل پیش بینی خطا مبتنی بر معیارهای جهش
نگارش: علی محبی

کمیته‌ی ممتحنین:

امضاء.....	دکتر حسن میریان	استاد راهنما:
امضاء.....	دکتر <نام استاد مدعو ۱>	استاد مدعو:
امضاء.....	دکتر <نام استاد مدعو ۲>	استاد مدعو:
تاریخ:		



اظهاری نامه (اصالت متن و محتوای رساله‌ی دکتری)

عنوان رساله: _____

نام استاد راهنما: _____ نام استاد راهنمای هم‌کار: _____ نام استاد مشاور: _____
این جانب _____ اظهار می‌دارم:

۱. متن و نتایج علمی ارایه‌شده در این رساله اصیل بوده و منحصرأً توسط این جانب و زیر نظر استادان (راهنما، هم‌کار و مشاور) نام‌برده‌شده در بالا تهیه شده است.

۲. متن رساله به این صورت در هیچ جای دیگری منتشر نشده است.

۳. متن و نتایج مندرج در این رساله، حاصل تحقیقات این جانب به عنوان دانشجوی دکتری دانشگاه صنعتی شریف است.

۴. کلیه‌ی مطالبی که از منابع دیگر در این رساله مورد استفاده قرار گرفته، با ذکر مرجع مشخص شده است.

نام دانشجو: _____

تاریخ:

امضاء:

نتایج تحقیقات مندرج در این رساله و دستاوردهای مادی و معنوی ناشی از آن (شامل فرمول‌ها، نرم‌افزارها، سخت‌افزارها و مواردی که قابلیت ثبت اختراع دارد) متعلق به دانشگاه صنعتی شریف است. هیچ شخصیت حقیقی یا حقوقی بدون کسب اجازه از دانشگاه صنعتی شریف حق فروش و ادعای مالکیت مادی یا معنوی بر آن یا ثبت اختراع از آن را ندارد. همچنین کلیه‌ی حقوق مربوط به چاپ، تکثیر، نسخه‌برداری، ترجمه، اقتباس و نظایر آن در محیط‌های مختلف اعم از الکترونیکی، مجازی یا فیزیکی برای دانشگاه صنعتی شریف محفوظ است. نقل مطالب با ذکر ماخذ بلامانع است.

نام استادان راهنما: _____ نام دانشجو: _____

تاریخ:

امضاء:

تاریخ:

امضاء:

تقديم به ...؛ صفحه‌ی تقديم اختياري است.

[این صفحه آگاهانه خالی گذاشته شده است.]

قدردانی

صفحه‌ی قدردانی. این صفحه اختیاری بوده و می‌توانید آن را حذف کنید. برای این کار کافی است محیط قدردانی در پرونده‌ی تِک را حذف کنید. متداول است که در این صفحه از خانواده، استادها و همکارهای خود قدردانی نمایید.

[این صفحه آگاهانه خالی گذاشته شده است.]

مدل پیش بینی خطا مبتنی بر معیارهای جهش

چکیده

توسعه دهندگان نرم افزار از طریق گزارش خطا در سیستم های رد گیری خطا و یا شکست در آزمون نرم افزار متوجه حضور خطا میشوند و پس از آن به جستجوی محل خطا و درک مشکل نرم افزار می پردازند. کشف زود هنگام خطاها موجب صرفه جویی در زمان و هزینه می شود و فرآیند اشکال زدایی را تسهیل می بخشد. ابزارهای آماری نوین امکان ساخت و بهره برداری از مدل های پیش بینی را فراهم می سازند. اصلی ترین جزء مدل های پیش بینی، معیارهای نرم افزار می باشد که با به کارگیری معیارهای نوین و موثر می توان به مدل های کارا تر دست پیدا کرد. در این پژوهش از معیارهای فرآیند و معیارهایی که بر اساس تحلیل جهش ساخته شده اند استفاده شده عملکرد مدل های حاصل ارزیابی شده اند. علاوه بر بکارگیری معیارهای جهش در کنار معیارهای فرآیند دو دسته معیار جدید به نام های معیارهای فرآیند مبتنی بر جهش و معیارهای ترکیبی جهش-فرآیند نیز جهت به کارگیری در ساخت مدل های پیش بینی معرفی شده اند. نتایج ارزیابی نشان می دهد معیارهای جهش می تواند به قدرت پیش بینی معیارهای فرآیند بیافزاید. معیارهای فرآیند مبتنی بر جهش علاوه بر داشتن قدرت پیش بینی بهتر از معیارهای جهش عمل نمی کنند. همچنین معیارهای ترکیبی جهش-فرآیند بهبود قابل توجهی را در عملکرد مدل های پیش بینی ایجاد می کنند.

کلیدواژه ها: پیش بینی خطا، آزمون نرم افزار، معیارهای جهش، معیارهای فرآیند.

[این صفحه آگاهانه خالی گذاشته شده است.]

سرخ‌ها

۱	۱. سرآغاز
۱	۱.۱. تعاریف مقدماتی
۲	۲.۱. بیان مسئله
۳	۳.۱. ساختار پایان‌نامه
۵	۲. مرور مطالعات پیشین
۵	۱.۲. پیش‌بینی خطا
۵	۱.۱.۲. فرآیند پیش‌بینی خطا
۶	۲.۱.۲. معیارهای ارزیابی
۹	۳.۱.۲. معیارهای پیش‌بینی خطا
۱۲	۴.۱.۲. مدل‌های پیش‌بینی خطا
۱۳	۲.۲. آزمون جهش و کاربردهای آن
۱۵	۱.۲.۲. مکان‌یابی خطا
۱۷	۲.۲.۲. مدل‌های یادگیری و جهش‌یافته‌ها
۱۸	۳.۲. جمع‌بندی مطالعات پیشین
۲۱	۳. معیارهای جهش و فرآیند
۲۱	۱.۳. معیارهای جهش و فرآیند
۲۴	۲.۳. معیارهای جهش مبتنی بر فرآیند
۲۵	۳.۳. معیارهای ترکیبی جهش-فرآیند
۲۹	۴. مورد مطالعاتی
۲۹	۱.۴. طراحی آزمایش

۲۹	۲.۴. آشنایی با ابزارها و مجموعه داده
۲۹	۱.۲.۴. مجموعه داده defect4j
۳۲	۲.۲.۴. ابزار Major
۳۶	۳.۲.۴. کتابخانه Jgit
۳۶	۴.۲.۴. چهارچوب Hibernate
۳۶	۳.۴. نکات پیاده‌سازی پروژه
۳۸	۴.۴. رویکرد اول: معیارهای فرآیند در کنار جهش
۳۸	۱.۴.۴. استخراج اطلاعات مربوط به ثبت‌های حاوی خطا
۴۰	۲.۴.۴. انتخاب پرونده‌های سالم
۴۱	۳.۴.۴. استخراج معیارهای فرآیند
۴۵	۴.۴.۴. استخراج معیارهای جهش
۴۸	۵.۴. رویکرد دوم: معیارهای فرآیند مبتنی بر جهش
۵۰	۶.۴. رویکرد سوم: معیارهای ترکیبی جهش-فرآیند

۵. ارزیابی ۵۱

۵۱	۱.۵. ارزیابی معیارهای فرآیند و جهش
۵۴	۲.۵. ارزیابی معیارهای فرآیند مبتنی بر جهش
۵۴	۱.۲.۵. مرحله اول
۵۶	۲.۲.۵. مرحله دوم
۵۸	۳.۵. ارزیابی معیارهای ترکیبی فرآیند-جهش

۶. نتیجه‌گیری و کارهای آتی ۶۳

پیوست‌ها ۶۶

آ. ساخت مدل‌های پیش‌بینی و ارزیابی ۶۷

ب. معیارهای استخراج شده ۷۱

کتاب‌نامه ۸۵

[این صفحه آگاهانه خالی گذاشته شده است.]

فهرست جدول‌ها

۱.۲	فرمول‌های محاسبه‌ی معیارهای ارزیابی	۷
۲.۲	جدول مشخصات پژوهش‌های مرور شده در حوزه‌ی پیش‌بینی خطا	۱۹
۱.۳	معیارهای فرآیند [۲۰]	۲۲
۲.۳	معیارهای جهش [۲۹]	۲۲
۱.۴	عملیات‌های موجود در defects4j	۳۰
۲.۴	پروژه‌های موجود در defects4j	۳۱
۱.۵	مقایسه‌ی معیارهای فرآیند به تنهایی و به همراه جهش	۵۲
۲.۵	مقادیر زیر نمودار ROC معیارهای فرآیند و به همراه جهش	۵۴
۳.۵	نتایج پیش‌بینی خطای معیارهای فرآیند مبتنی بر جهش – مرحله‌ی اول	۵۵
۴.۵	مقادیر زیر نمودار ROC معیارهای فرآیند مبتنی جهش	۵۶
۵.۵	نتایج پیش‌بینی خطای مدل حاصل از بکارگیری تمامی معیارها	۵۷
۶.۵	مقادیر زیر نمودار ROC تمامی معیارها	۵۸
۷.۵	مقایسه‌ی معیارهای فرآیند و معیارهای ترکیبی جهش-فرآیند	۵۹
۸.۵	مقادیر زیر نمودار ROC معیارهای فرآیند و معیارهای ترکیبی جهش-	۶۱
۱.ب	معیارهای فرآیند	۷۱

[این صفحه آگاهانه خالی گذاشته شده است.]

فهرست شکل‌ها

۱.۲	فرآیند پیش‌بینی خطا [۴]	۶
۲.۲	نمونه‌ای از نمودار ROC [۵]	۸
۳.۲	نمودار موثر بودن از نظر هزینه [۷]	۸
۴.۲	نمونه‌ای از جهش‌یافته‌های یک برنامه [۲۶]	۱۵
۱.۴	اجرای دستور info در defects4j	۳۲
۲.۴	نمونه کد MML در Major	۳۳
۳.۴	اجرای عملیات جهش برای یک پرونده	۳۴
۴.۴	نمونه‌ای از پرونده‌ی mutants.log	۳۵
۵.۴	اجرای تحلیل جهش	۳۵
۶.۴	نتایج خروجی تحلیل جهش	۳۶
۷.۴	نمایی از مخزن نرم‌افزاری	۳۸
۸.۴	نمایی از جدول محتوای انتشارها	۳۹
۹.۴	نمایی از جدول محتوای اطلاعات پرونده‌های حاوی خطا	۴۰
۱۰.۴	نمایی از جدول محتوای اطلاعات پرونده‌های سالم	۴۱
۱۱.۴	نمایی از جدول اطلاعات ثبت‌ها	۴۲
۱۲.۴	نمایی از جدول تغییرات پرونده‌ها در ثبت‌ها	۴۲
۱۳.۴	نمایی از جدول مشارکت‌کنندگان در ویرایش پرونده‌ها	۴۴
۱۴.۴	نمایی از جدول معیارهای فرآیند	۴۵
۱۵.۴	پرونده‌ی mml ساخته شده جهت تولید جهش‌یافته‌ها	۴۷
۱۶.۴	نمایی از جدول نتایج تحلیل جهش	۴۸
۱۷.۴	نمایی از جدول تعداد جهش‌یافته‌های متمایز در انتشارها	۴۹
۱۸.۴	نمایی از جدول نتایج تحلیل جهش در انتشارها	۵۰

- ۱.۵. نمودارهای ROC معیارهای فرآیند و به همراه جهش ۵۳
- ۲.۵. نمودارهای ROC معیارهای فرآیند ، فرآیند و جهش ، فرآیند مبتنی بر جهش ۵۶
- ۳.۵. نمودارهای ROC معیارهای جهش و فرآیند و تمامی معیارها ۵۸
- ۴.۵. نمودارهای ROC معیارهای فرآیند و به همراه جهش ۶۰

فصل ۱

سرآغاز

سامانه‌های نرم‌افزاری بسیار فراگیر شده‌اند و زندگی امروزی را ارتقا داده‌اند. در نتیجه کاربران کیفیت نرم‌افزار بالایی را تقاضا می‌کنند. کشف و برطرف کردن خطاها پرهزینه است و مدل‌های پیش‌بینی خطا از طریق اولویت‌دهی به فعالیت‌های تضمین کیفیت موجب افزایش بازدهی می‌گردند. پیش‌بینی خطا از سال ۱۹۹۲ تا کنون یک زمینه‌ی فعال تحقیقاتی بوده است. محققان همواره به دنبال روش‌هایی بوده‌اند که پیش‌بینی خطا را با کیفیت بهتری انجام دهند و یا دامنه‌ی کاربرد آن را گسترش بخشند.

به منظور افزایش کارایی پیش‌بینی خطا محققان معیارهای نوینی را ارائه داده‌اند [۱]، سعی داشته‌اند محدودیت‌های یادگیری ماشین را تقلیل بخشند [۲] و یا روش‌های بروزتری را به منظور دسته‌بندی^۱ به کار گیرند [۳].

۱.۱. تعاریف مقدماتی

در این قسمت چند اصطلاح رایج در مبحث پیش‌بینی خطا و مورد استفاده در پایانامه نوشته شده است.

- مورد آزمون^۲:

یک مورد آزمون متشکل است از مقادیر مورد آزمون، نتایج مورد انتظار که با اجرای برنامه تحت آزمون یک یا چند عملکرد آنرا ارزیابی می‌کند.

- سامانه‌ی کنترل نسخه^۳:

این سامانه تغییرات اعمال شده بر روی یک یا چندین پرونده^۴ را ذخیره می‌کند تا در آینده بتوان یک نسخه‌ی خاص را بازخوانی کرد.

¹Classification

²Test Case

³Version Control System

⁴File

• ثبت^۵:

ذخیره‌ی تغییرات ایجاد شده بر روی پرونده‌ها در سامانه‌ی کنترل نسخه را ثبت می‌نامند. یک ثبت را می‌تواند معادل یک نسخه از برنامه در نظر گرفت که البته این نسخه می‌تواند ناکامل باشد.

• انتشار^۶:

انتشار به معنی توزیع نسخه‌ی نهایی یک نرم‌افزار است که قابل استفاده برای کاربر می‌باشد. یک انتشار ممکن است نسخه‌ای از یک برنامه‌ی جدید باشد و یا ارتقاء یافته‌ی نرم‌افزار موجود باشد. قبل از یک انتشار معمولاً به ترتیب نسخه‌های *آلفا*^۷ و *بتا*^۸ توزیع می‌شود.

۲.۱. بیان مسئله

آزمون نرم‌افزار اصلی ترین فعالیت تیم تضمین کیفیت می‌باشد. آزمون نرم‌افزار می‌تواند تا ۵۰ درصد هزینه‌ی تولید نرم‌افزار را به خود اختصاص دهد. هدف از پیش‌بینی خطا افزایش بازدهی این فرآیند می‌باشد. حال با بهبود پیش‌بینی خطا می‌توان به دستیابی به این هدف کمک نمود. به منظور پیش‌بینی خطا معیارهایی در سطح مورد نظر استخراج می‌گردد و با استفاده از دسته‌بندی خطا دار بودن یا نبودن قطعه‌ی مورد بررسی پیش‌بینی می‌شود. یک دسته از معیارهای مورد استفاده در این زمینه معیارهای فرآیند است و معیارهای جهش نیز به تازگی در این راستا استفاده شده‌اند. این پژوهش قصد دارد تا بررسی کند که معیارهای جهش در کنار فرآیند موجب چه میزان در پیش‌بینی خطا تاثیر گذار است و همچنین بر اساس مفاهیم تحلیل جهش معیارهای جدیدی ارائه دهد تا پیش‌بینی خطا بهبود یابد.

با توجه به اینکه معیارهای جهش به تازگی در پیش‌بینی خطا مورد استفاده قرار گرفته‌اند لازم است تا تحقیقات بیشتری در مورد آنها صورت گیرد و عملکرد آنها از ابعاد مختلف مورد بررسی قرار گیرد. همچنین با بررسی مطالعات پیشین نقاط ضعف و قوت معیارهایی که تا کنون ارائه شده‌اند مورد بررسی قرار گرفتند. در این پژوهش عملکرد معیارهای جهش مورد بررسی بیشتری قرار گرفتند و با توجه به نقاط ضعف و قدرت معیارهای قبلی، معیارهایی ارائه شده تا بخشی از نقاط ضعف را پوشش دهند و به پیش‌بینی بهتری بیانجامند.

⁵Commit⁶Release⁷Alpha⁸Beta

۳.۱. ساختار پایانامه

این پایانامه در ۶ فصل تهیه گردیده است. در فصل ۲ به مرور مطالعات پیشین پرداخته می‌شود که در قسمت ۱.۲ مباحث مربوط به پیش‌بینی خطا از جمله فرآیند پیش‌بینی، معیارهای ارزیابی، معیارهای پیش‌بینی و مدل‌های پیش‌بینی بررسی می‌شوند. در قسمت ۲.۲ مباحث مربوط به آزمون جهش بررسی شده‌اند. و در قسمت ۳.۲ مطالعات مروری جمع‌بندی شده‌اند. در فصل ۳ معیارهای مورد استفاده و ارائه شده در این پژوهش معرفی می‌شوند. در فصل ۴ پنج پروژه‌ی صنعتی مورد مطالعه قرار گرفته‌اند و در فصل ۵ معیارها مورد ارزیابی قرار گرفته‌اند. در فصل ۶ مباحث مطرح شده در این پایانامه جمع‌بندی شده و کارهای آتی شرح داده شده است.

[این صفحه آگاهانه خالی گذاشته شده است.]

فصل ۲

مرور مطالعات پیشین

۱.۰۲. پیش بینی خطا

۱.۱.۲. فرآیند پیش بینی خطا

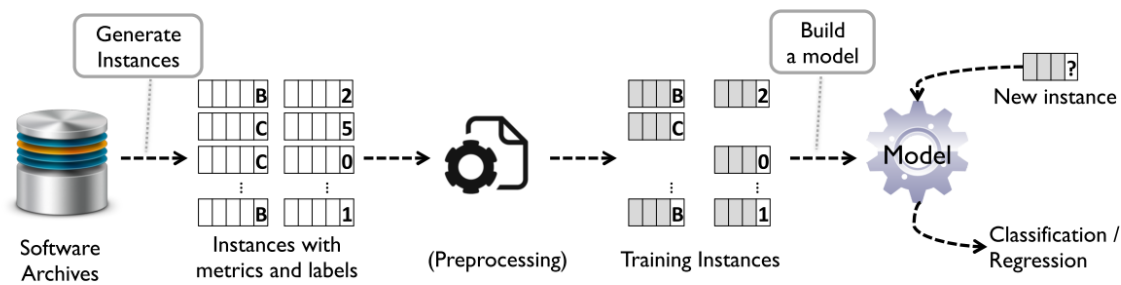
اکثریت پژوهش‌های پیش‌بینی خطا از روش‌های یادگیری ماشین استفاده کرده‌اند. اولین گام در ساخت مدل پیش‌بینی تولید داده‌هایی با استفاده از آرشیوهای نرم‌افزاری همانند سامانه‌های کنترل نسخه مانند گیت^۱، سیستم‌های ردگیری مشکلات مانند جیرا و آرشیو ایمیل‌ها است. هر یک از این داده‌ها بر اساس درشت‌دانی پیش‌بینی می‌توانند نمایانگر یک سیستم، یک قطعه^۲ نرم‌افزاری، بسته^۳، فایل کد منبع، کلاس و یا تابع باشد. مقصود از داده یک بردار ویژگی حاوی چندین معیار (یا ویژگی) می‌باشد که از آرشیوهای نرم‌افزاری استخراج شده و دارای برچسب سالم و خطا/ار و یا تعداد خطاها است. پس از تولید داده‌ها با استفاده از معیارها و برچسب‌ها می‌توان به پیش پردازش داده‌ها پرداخت (مانند انتخاب معیار) که البته این امر اختیاری می‌باشد. پس از بدست آوردن مجموعه‌ی نهایی داده‌ها یک مدل پیش‌بینی را آموزش می‌دهیم که می‌تواند پیش‌بینی کند یک داده‌ی جدید حاوی خطا است یا خیر. تشخیص خطا^۴خیز بودن داده معادل دسته بندی دودویی است و پیش‌بینی تعداد خطاها معادل رگرسیون می‌باشد. در شکل ۱.۲ فرآیند پیش‌بینی خطا نشان داده شده است. داده‌ها نمونه‌هایی هستند که می‌توانند خطا دار و بدون خطا بودن (B = buggy یا C = clean) و یا تعداد خطا را نشان دهند. لازم به ذکر است که در یک مدل پیش‌بینی تنها از یک نوع از این داده‌ها استفاده می‌شود.

¹Git

²Component

³Package

⁴Bug-proneness



شکل ۱.۲: فرآیند پیش‌بینی خطا [۴]

۲.۱.۲. معیارهای ارزیابی

معیارهای ارزیابی را می‌توان به دسته‌ی کلی معیارهای دسته‌بندی و رگرسیون تقسیم کرد. معیارهای دسته‌بندی را می‌توان با استفاده از ماتریس درهم‌ریختگی^۵ محاسبه نمود. در ماتریس درهم‌ریختگی پیش‌بینی خطا، عناصر به صورت زیر تعریف می‌شوند. همچنین نحوه‌ی محاسبه‌ی معیارها در جدول ۱.۲ آمده است.

• TP : تعداد داده‌های حاوی خطا که به درستی تشخیص داده شدند

• FP: تعداد داده‌های سالم که به عنوان خطا دار پیش‌بینی شدند

• TN: تعداد داده‌های سالم که به درستی تشخیص داده شدند

• FN: تعداد داده‌های حاوی خطا که به عنوان داده‌ی سالم پیش‌بینی شدند

⁵Confusion Matrix

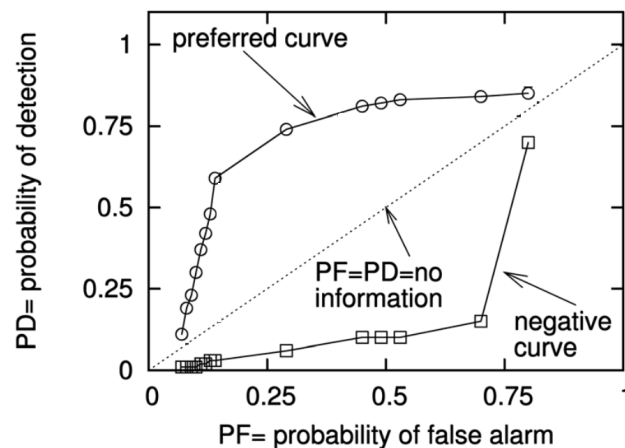
جدول ۱.۲: فرمول‌های محاسبه‌ی معیارهای ارزیابی

نام معیار	نام لاتین	نحوه‌ی محاسبه	توضیح
نرخ مثبت کاذب	False Positive Rate (PF)	$\frac{FP}{TN + FP}$	نسبت تعداد داده‌هایی که به اشتباه خطادار پیش‌بینی شده‌اند به تعداد کل داده‌های بدون خطا
صحت	Accuracy	$\frac{TP + TN}{TP + FP + TN + FN}$	نسبت تعداد پیش‌بینی‌های درست به تعداد کل پیش‌بینی‌ها
دقت	Precision	$\frac{TP}{TP + FP}$	نسبت تعداد داده‌هایی که به درستی خطادار پیش‌بینی شده‌اند به تعداد کل داده‌هایی که خطادار پیش‌بینی شده‌اند
بازخوانی	Recall (PD)	$\frac{TP}{TP + FN}$	نسبت تعداد داده‌هایی که به درستی خطادار پیش‌بینی شده‌اند به تعداد کل داده‌های خطادار
معیار اف	F-Measure	$\frac{2 \times Precision \times Recall}{Precision + Recall}$	از آنجا که در بین معیارهای دقت و بازخوانی مصالحه وجود دارد معیار اف ترکیبی از آن دو را در نظر می‌گیرد

دو معیار دیگر نیز که در پژوهش‌ها کاربرد دارند عبارتند از AUC^۶ و AUCEC^۷ که هر دو به مساحت زیر یک منحنی اشاره می‌کنند. AUC مساحت زیر نمودار ROC^۸ را اندازه‌گیری می‌کند. در نمودار ROC، محورهای عمودی و افقی را به ترتیب بازخوانی و نرخ مثبت کاذب تشکیل می‌دهد. با تغییر آستانه پیش‌بینی برای یک مدل می‌توان میزان بازخوانی و نرخ مثبت کاذب را تغییر داده و بدین ترتیب منحنی ROC را رسم نمود. یک مدل بی‌نقص دارای مساحت زیر نمودار ۱ است. برای یک مدل تصادفی منحنی از مبدا به نقطه‌ی (۱,۱) رسم خواهد شد. یک نمونه از منحنی ROC در شکل ۲.۲ آمده است.

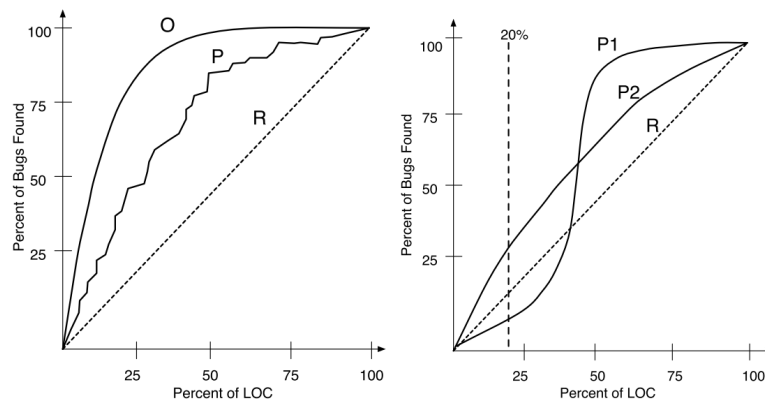
معیار AUCEC معیاری است که تعداد خطوطی از برنامه که توسط تیم تضمین کیفیت و یا توسعه دهندگان نیاز است بررسی و آزموده شود را در نظر می‌گیرد. ایده‌ی موثر بودن از نظر هزینه^۹ برای مدل‌های خطا برای اولین بار توسط آریشلم و همکاران [۶] ارائه گردید. موثر بودن از نظر هزینه به این معنا است که چه تعداد خطا با بررسی و یا تست $n\%$ اول خطوط می‌توان یافت. به عبارت دیگر اگر یک مدل پیش‌بینی خطا بتواند تعداد

^۶Area under curve^۷Area under cost-effectiveness curve^۸Receiver operating characteristic^۹Cost-effectiveness



شکل ۲.۲: نمونه‌ای از نمودار ROC [۵]

خطای بیشتری را با بررسی و تلاش در آزمون کمتر، نسبت به باقی مدل‌ها بیابد می‌توان گفت که تاثیر آن از نظر هزینه بیشتر است. دو منحنی در قسمت راست شکل ۳.۲ برای دو مدل پیش‌بینی مختلف آمده است. هر دو مدل دارای سطح زیر نمودار یکسانی هستند اما زمانی که ۲۰٪ اول محور افقی در نظر گرفته می‌شود مدل P_2 کارایی بهتری دارد. نمودار سمت چپ مدل‌های تصادفی، عملی^{۱۰} و بهینه را نشان می‌دهد.



R = random P = practical O = optimal

شکل ۳.۲: نمودار موثر بودن از نظر هزینه [۷]

معیارهایی که برای ارزیابی نتایج حاصل از روش رگرسیون به کار گرفته می‌شوند بر اساس همبستگی^{۱۱} میان تعداد خطاهای پیش‌بینی شده و خطاهای واقعی محاسبه می‌شوند. نماینده‌ی این معیارها را می‌توان همبستگی اسپیرمن، پیرسون و R^2 دانست [۴].

¹⁰Practical

¹¹Correlation

۳.۱.۲. معیارهای پیش‌بینی خطا

معیارهای پیش‌بینی خطا نقش مهمی را در ساخت مدل پیش‌بینی ایفا می‌کنند. اکثریت معیارهای پیش‌بینی خطا را می‌توان به دو دسته‌ی کلی تقسیم کرد: معیارهای کد و معیارهای فرآیند. معیارهای کد می‌توانند به طور مستقیم از کدهای منبع موجود جمع‌آوری شوند در حالی که معیارهای فرآیند از اطلاعات تاریخی که در مخازن نرم‌افزاری مختلف آرشیو شده‌اند استخراج می‌گردند. نمونه‌ای از این مخازن نرم‌افزاری سیستم‌های کنترل نسخه و سیستم‌های ردگیری خطا است. معیارهای فرآیند از نظر هزینه موثرتر از سایر معیارها هستند [۸]. در برخی از مقالات نیز معیارهای پیش‌بینی خطا به سه دسته‌ی: معیارهای کد منبع سستی، معیارهای شیء‌گرایی و معیارهای فرآیند تقسیم شده‌اند [۹].

معیارهای کد

معیارهای کد تحت عنوان *معیارهای محصول*^{۱۲} نیز شناخته می‌شوند و میزان پیچیدگی کد را می‌سنجند. فرض زمینه‌ی^{۱۳} آنها این است که هرچه کد پیچیده‌تر باشد خطاخیزتر است. برای اندازه‌گیری پیچیدگی کد پژوهش‌گران معیارهای مختلفی را ارائه داده‌اند که در ادامه مهم‌ترین آنها معرفی خواهند شد.

• **معیار اندازه:** معیارهای "اندازه" اندازه‌ی کلی و حجم کد را می‌سنجند. نماینده‌ی این معیارها "تعداد خطوط" می‌باشد و اولین بار توسط *آکیاما*^{۱۴} [۱۰] ارائه شد. *هالستد*^{۱۵} [۱۱] چندین معیار اندازه بر اساس تعداد عملگرها و عملوندها ارائه داده است و در مقاله‌ی [۱۲] مورد بازنگری قرار گرفته است.

• **معیار پیچیدگی حلقوی:** مک‌کیب^{۱۶} معیارهای پیچیدگی حلقوی^{۱۷} را پیشنهاد داد که این معیار با استفاده از تعداد گره‌ها، یالها و قطعات متصل در گراف جریان کنترلی^{۱۸} کد منبع محاسبه می‌گردد [۱۳]. این معیارها نشان می‌دهند که راه‌های کنترلی به چه میزان پیچیده هستند. با وجود اینکه جز اولین معیارها بوده است همچنان در پیش‌بینی خطا کاربرد دارد [۱۴].

• **معیار شیء‌گرایی:** با ظهور زبان‌های شیء‌گرایی و محبوبیت آنها معیارهای کد برای این زبان‌ها ارائه شد

¹²Product Metrics

¹³Ground Assumption

¹⁴Akiyama

¹⁵Halstead

¹⁶McCabe

¹⁷Cyclomatic Complexity

¹⁸Control Flow

تا فرآیند توسعه بهبود یابد. نماینده‌ی معیارهای شیء‌گرایی چد/میر و کمرر^{۱۹} می‌باشند [۱۵]. این معیارها با توجه به خصیصه‌های زبان‌های شیء‌گرا مانند وراثت، زوجیت^{۲۰}، همبستگی^{۲۱} طراحی شده‌اند. بجز معیارهای معیارهای شیء‌گرایی دیگری نیز بر اساس حجم و کمیت کد منبع پیشنهاد داده شده‌اند. مشابه معیارهای اند/زه، معیارهای شیء‌گرایی تعداد نمونه‌های یک کلاس، توابع را می‌شمارند.

معیارهای فرآیند

در ادامه تعدادی از معیارهای فرآیند بررسی می‌شوند که در این دسته شاخص محسوب می‌شوند.

- **تغییر نسبی کد:** ناگاپان و بال^{۲۲} هشت معیار تغییر نسبی کد را ارائه داده‌اند [۱۶]. به عنوان مثال یکی از معیارهای آنها تعداد تجمعی خطوط اضافه و حذف شده بین دو نسخه از برنامه را می‌شمارد و بر تعداد خطوط برنامه تقسیم می‌کند. معیار دیگر تعداد فایل‌های تغییر یافته از یک قطعه برنامه را بر تعداد کل فایل‌ها تقسیم می‌کند.
- **تغییر کد:** این معیارها به عنوان مثال تعداد رفع خطاها، تعداد بازرایی کد^{۲۳} و یا تعداد نویسندگان یک فایل را می‌شمارند. موزر^{۲۴} و همکاران معیارهایی را ارائه داده‌اند که تعداد خطوط اضافه و کم شده را بدون در نظر گرفتن تعداد کل خطوط می‌شمارد. در عوض سن فایل‌ها و تعداد فایل‌هایی که در سامانه‌ی کنترل نسخه ثبت می‌شوند در نظر گرفته می‌شود [۱۷].
- **معیار شهرت:** بکچلی^{۲۵} و همکاران معیارهای شهرت^{۲۶} را بر اساس تحلیل ایمیل‌های آرشیو شده‌ی نویسندگان ارائه داده‌اند. ایده‌ی اصلی این معیارها این است که یک قطعه‌ی نرم‌افزاری که در ایمیل‌ها درباره‌ی آن بیشتر صحبت شده است خطاخیزتر می‌باشد [۱]. برد و همکاران چهار معیار مالکیت بر اساس نویسندگان یک قطعه ارائه داده‌اند. مالکیت یک قطعه بر اساس نسبت تعداد ثبت‌های افراد در سیستم کنترل نسخه برای یک قطعه (مشارکت آنها) تعریف می‌شود.

¹⁹Chidamber and Kemerer (CK)

²⁰Coupling

²¹Cohesion

²²Nagappan and Ball

²³Refactoring

²⁴Moser

²⁵Bacchelli

²⁶Popularity

راجنویک^{۲۷} و همکاران در پژوهش خود به بررسی قاعده‌مند^{۲۸} معیارهای پیش‌بینی خطا در مطالعات پیشین پرداخته‌اند. طبق این پژوهش در ۴۹٪ مطالعات از معیارهای شی‌گرایی، در ۲۷٪ معیارهای سنتی کد و در ۲۶٪ از معیارهای فرآیند استفاده شده است. با توجه به مطالعات بررسی شده دقت پیش‌بینی خطا با انتخاب معیارهای مختلف، تفاوت قابل توجهی پیدا می‌کند. معیارهای شی‌گرایی و فرآیند موفق‌تر از معیارهای سنتی هستند. معیارهای سنتی پیچیدگی کد، قویا با معیارهای اندازه مانند تعداد خطوط کد همبستگی دارند و این دو توانایی پیش‌بینی خطا دارند اما جز بهترین معیارها نیستند. معیارهای شی‌گرایی بهتر از اندازه و پیچیدگی عمل می‌کنند و با این که با معیارهای اندازه همبستگی دارند اما ویژگی‌های بیشتری علاوه بر اندازه را دارند. معیارهای ایستای کد همانند اندازه، پیچیدگی و شی‌گرایی به منظور بررسی یک نسخه از برنامه مفید هستند اما با هر تکرار^{۲۹} در فرآیند توسعه نرم‌افزار دقت پیش‌بینی آنها کاسته می‌شوند و معیارهای فرآیند در چنین شرایطی بهتر عمل می‌کنند. با این وجود که معیارهای فرآیند دارای توانمندی بالقوه‌ای هستند، اما در تعداد کمتری از پژوهش‌ها مورد استفاده قرار گرفته‌اند[۹].

آسترند^{۳۰} و همکاران به بررسی این موضوع پرداخته‌اند که آیا اطلاعاتی درباره‌ی اینکه کدام توسعه‌دهنده یک فایل را اصلاح می‌کند قادر است که پیش‌بینی خطا را بهبود بخشد. در پژوهش قبلی آنها[۱۸] مشخص شده بود که تعداد کلی افراد توسعه‌دهنده در یک فایل می‌تواند در پیش‌بینی خطا تاثیر متوسطی داشته باشد. در مقاله‌ی [۱۹] تعدادی از متغیرهای کد منبع و فرآیند به همراه معیار مرتبط به توسعه‌دهنده در نظر گرفته شده است. در این پژوهش مشخص شد که تعداد خطاهایی که یک توسعه‌دهنده تولید می‌کند ثابت است و با سایر توسعه‌دهندگان فرق دارد. این تفاوت با حجم کدی که یک توسعه‌دهنده اصلاح می‌کند مرتبط است و در نتیجه در نظر گرفتن یک نویسنده خاص نمی‌تواند به بهبود پیش‌بینی خطا کمک کند[۱۹].

رحمان و دونبو^{۳۱} از جنبه‌های مختلف معیارهای فرآیند را با سایر معیارها مقایسه کرده‌اند[۲۰]. نتایج نشان می‌دهد زمانی که مدل پیش‌بینی بر روی یک نسخه آموزش می‌بیند و در نسخه‌ی بعدی آزموده می‌شود معیارهای کد، AUC قابل قبولی دارند اما AUC آنها کمتر از معیارهای فرآیند است و از نظر معیار ۲۰٪ AUCEC بهتر از یک مدل تصادفی عمل نمی‌کنند و به آن معنی است که این معیارها از نظر هزینه چندان موثر نیستند. همچنین

²⁷Radjenovic²⁸Systematic Review²⁹Iteration³⁰Ostrand³¹Rahman and Devanbu

معیارهای کد ایستار هستند، یعنی با تغییرات پروژه و تغییر در توزیع خطاها همچنان معیارها بدون تغییر باقی می‌مانند. معیار ایستا تمایل دارد یک فایل را در انتشارهای متوالی همچنان حاوی خطا معرفی کند. معیارهای ایستا به مدل‌های راکد منجر می‌شوند که این مدل‌ها به سمت فایل‌های بزرگ با تراکم خطای کمتر جهت‌گیری^{۳۲} دارند. به عنوان مثال حالتی را در نظر بگیرید که در یک پروژه فایل‌های بزرگ و پیچیده‌ای وجود دارد که پس از چندین انتشار خطاهای آنها برطرف می‌شود اما مدل‌هایی که بر اساس معیارهای کد ساخته شده‌اند همچنان این فایل‌ها را به عنوان خطاخیز معرفی می‌کنند. از طرف دیگر حالتی را در نظر بگیرید که یک فایل با اندازه و پیچیدگی کم به تازگی به وجود آمده و یا تغییرات فراوان یافته است. مدل‌های مبتنی بر کد به این فایل‌ها توجه چندانی نخواهند کرد در حالیکه که این فایل‌ها مستعد وجود خطا هستند. بدین ترتیب معیارهای فرآیند بهتر از معیارهای کد عمل می‌کنند.

۴.۱.۲. مدل‌های پیش‌بینی خطا

اکثریت مدل‌های پیش‌بینی خطا بر اساس یادگیری ماشین می‌باشند. بر اساس اینکه چه چیزی پیش‌بینی شود (خطاخیز بودن یا تعداد خطا)، مدل‌ها به دو دسته‌ی کلی تقسیم می‌شوند، که عبارتند از دسته‌بندی و رگرسیون. با توسعه‌ی روش‌های جدیدتر یادگیری ماشین تکنیک‌های فعال و نیمه-نظارتی^{۳۳} برای ساخت مدل‌های پیش‌بینی خطای کارا تر به کار گرفته شده است [۲۱]. علاوه بر مدل‌های یادگیری ماشین، مدل‌های غیر آماری مانند باگ‌کش^{۳۴} پیشنهاد داده شده است [۲۲]. در میان روش‌های دسته‌بندی، Logistic Regression، Naive Bayes و Decision Tree بیش از سایرین در پژوهش‌ها مورد استفاده قرار گرفته‌اند. همچنین در میان روش‌های رگرسیون Linear Regression و Negative Binomial Regression به طور گسترده به کار گرفته شده‌اند [۴].

کیم^{۳۵} و همکاران باگ‌کش را ارائه داده‌اند که اولویت موجودیت‌های خطاخیز در کش را نگهداری می‌کند. این روش از اطلاعات محلی خطاها مانند اطلاعات زمانی و مکانی بهره می‌گیرد. به عنوان مثال اگر خطا در یک موجودیت به تازگی به وجود آمده یا همراه با سایر موجودیت‌ها تغییر کرده است، آن موجودیت با احتمال بیشتری حاوی خطا خواهد بود.

اگرچه مدل‌های یادگیری مختلف می‌تواند با توجه به داده‌های ورودی یکسان، متفاوت عمل کنند و کارایی یک روش نسبت به دیگری متفاوت باشد، با این حال پژوهشی که توسط آریشلم و همکاران [۸] انجام شده است نشان

³²Bias

³³Semi-Supervised

³⁴BugCache

³⁵Kim

می‌دهد که تاثیر تکنیک یادگیری در حد متوسطی است و کمتر از انتخاب معیار بر روی کارایی تاثیر گذار است.

مالهوترا^{۳۶} با بکارگیری معیارهای سنتی کد، عملکرد تکنیک‌های یادگیری ماشین و رگرسیون را مقایسه کرده است [۱۴]. وی به منظور پیش پردازش نیز از آماره‌های توصیفی^{۳۷} استفاده کرده است و داده‌های نامناسب را شناسایی نموده است. آماره‌های توصیفی می‌توانند شامل میانگین، کمینه، بیشینه و واریانس باشد. متغیرهای مستقلی که واریانس کمی دارند ماژول‌ها را به خوبی متمایز نمی‌کنند و بعید است که مفید باشند و می‌توانند حذف شوند. یک روش رگرسیون و شش روش دسته‌بندی مورد آزمایش قرار گرفته‌اند که در میان آنها سه روش رایج و سه روش که کمتر مورد استفاده قرار می‌گیرند انتخاب شده‌اند. Logistic Regression به عنوان روش رگرسیون انتخاب شده و نتایج نشان می‌دهد که روش‌های دسته‌بندی بهتر از روش رگرسیون عمل می‌کند. در میان روش‌های دسته‌بندی درخت تصمیم^{۳۸} بهتر از سایرین عمل کرده است.

۱.۴.۱.۲. درشت‌دانگی پیش‌بینی

در پژوهش‌های انجام شده مدل‌های پیش‌بینی در سطوح مختلفی از ریزدانگی ساخته شده‌اند از جمله: زیر سیستم، قطعه یا بسته، فایل یا کلاس، تابع و تغییر. هتا^{۳۹} و همکاران پیش‌بینی در سطح تابع را ارائه داده‌اند و به این نتیجه رسیده‌اند که پیش‌بینی خطا در سطح تابع نسبت به سطوح درشت‌دانه‌تر از نظر هزینه موثرتر است [۲۳]. کیم و همکاران نیز مدل جدیدی ارائه داده‌اند که دسته‌بندی تغییر نام دارد. بر خلاف سایر مدل‌های پیش‌بینی، ”دسته‌بندی تغییر می‌تواند به طور مستقیم به توسعه دهنده کمک کند. این مدل می‌تواند زمانی که توسعه دهنده تغییری در کد منبع ایجاد می‌کند و آنرا در سیستم کنترل نسخه ثبت می‌کند، نتایج آنی را فراهم کند. از آنجا که این مدل بر اساس بیش از ده هزار ویژگی ساخته می‌شود، سنگین‌تر از آن است که در عمل مورد استفاده قرار گیرد [۲۴].

۲.۲. آزمون جهش و کاربردهای آن

توسعه‌دهندگان و پژوهش‌گران حوزه‌ی نرم‌افزار علاقه‌مند به اندازه‌گیری موثر بودن مجموعه‌های آزمون می‌باشند. توسعه‌دهندگان به دنبال آن هستند که بدانند مجموعه آزمون‌های آنها می‌تواند به خوبی خطاها را تشخیص دهد و پژوهشگران به دنبال مقایسه‌ی روش‌های مختلف آزمون و اشکال زدایی^{۴۰} هستند. به طور ایده آل افراد تمایل

³⁶ Malhotra

³⁷ Descriptive Statistics

³⁸ Decision Tree

³⁹ Hata

⁴⁰ Debugging

دارند که بدانند تعداد خطاهایی که یک مجموعه آزمون می‌تواند شناسایی کند چه مقدار است اما از آنجا که خطاها ناشناخته هستند باید از اندازه‌گیری وکالتی^{۴۱} استفاده شود. یکی از اندازه‌گیری‌های شناخته شده / امتیاز جهش^{۴۲} می‌باشد که توانایی مجموعه آزمون در تمیز دادن نسخه‌ی اصلی برنامه از تعداد زیادی نسخه‌های متفاوت را اندازه‌گیری می‌کند. این نسخه‌های متفاوت که تنها یک تفاوت کوچک نحوی نسبت به برنامه‌ی اصلی دارند جهش‌یافته^{۴۳} نامیده می‌شوند. امتیاز جهش درصد جهش‌یافته‌هایی است که توسط مجموعه آزمون از برنامه‌ی اصلی تمیز داده می‌شوند. به این صورت که این جهش‌یافته‌ها باعث شکست یک مورد آزمون می‌شوند در حالی که در نسخه‌ی اصلی مجموعه‌ی آزمون با موفقیت اجرا می‌گردد. جهش‌یافته‌ها با تزریق خطاهای ساختگی به برنامه‌ی تحت آزمون ساخته می‌شوند. نمونه‌ای از جهش‌یافته‌ها برای یک قطعه کد در شکل ۴.۲ آمده است. این خطاهای ساختگی با استفاده از عملگرهای جهش که از پیش تعریف شده‌اند ساخته می‌شود. نمونه‌ی این عملگرها جایگزینی عملگرهای ریاضی یا رابطه‌ای، تغییر شرط شاخه^{۴۴} و یا حذف یک عبارت است [۲۵].

تحلیل آزمون در موارد زیر کاربرد دارد:

- ارزیابی مجموعه آزمون
- انتخاب مجموعه آزمون
- کمینه سازی مجموعه آزمون
- تولید مجموعه آزمون
- مکان‌یابی خطا
- پیش‌بینی خطا

⁴¹Proxy Measurement

⁴²Mutation Score

⁴³Mutant

⁴⁴Branch Condition

Statements	Mutants
s ₁ : max = -x;	m1: max -= x-1; m2: max=x;
s ₂ : if(max < y) {	m3: if(!(max<y)) { m4: if(max==y) {
s ₃ : max = y;	m5: max = -y; m6: max = y+1;
s ₄ : if(x*y<0) {	m7: if(!(x*y<0)) m8: if(x/y<0)
s ₅ : print(`diff.sign`);}	m9: return; m10:;
s ₆ : print(max); }	m11: printf(0);} m12:;

شکل ۴.۲: نمونه‌ای از جهش‌یافته‌های یک برنامه [۲۶]

جاست^{۴۵} و همکاران در پژوهش خود به بررسی این موضوع پرداخته‌اند که آیا جهش‌یافته‌ها می‌توانند جایگزین مناسبی برای خطاهای واقعی باشند یا خیر [۲۵]. در پژوهش‌های گذشته بررسی شده بود که میان جهش‌یافته‌های ساده و پیچیده وابستگی وجود دارد ولی وابستگی میان جهش‌یافته‌های ساده و خطاهای واقعی مشخص نیست. جاست و همکاران دو مجموعه‌ی آزمون برای هر خطا در نظر گرفتند که مجموعه‌ی اول در نسخه‌ی حاوی خطا با موفقیت گذرانده می‌شود. مجموعه‌ی دوم در نسخه‌ی حاوی خطا شکست می‌خورد و در نسخه‌ی رفع خطا با موفقیت اجرا می‌شود. نتایج نشان می‌دهد که مجموعه‌ی آزمون دوم دارای امتیاز جهش بالاتری می‌باشد که نشان می‌دهد هر خطا به یک جهش‌یافته وابستگی دارد. لازم به ذکر است که سعی شده دو مجموعه‌ی آزمون دارای پوشش یکسانی باشند زیرا پوشش بیشتر می‌تواند امتیاز جهش بیشتر بیانجامد. همچنین مشخص شد که ۷۳٪ خطاهای واقعی با جهش‌یافته‌هایی که با عملگرهای متوال تولید شده‌اند وابستگی دارند. در این پژوهش خطاهایی که با جهش‌یافته‌ها وابستگی ندارند در سه دسته قرار می‌گیرند: دسته اول نیازمند عملگرهای قوی‌تری هستند، دسته دوم نیازمند عملگرهای جدیدی هستند و دسته سوم با جهش‌یافته‌ها وابستگی ندارند.

۱.۲.۲. مکان‌یابی خطا

روش‌هایی که از جهش‌یافته‌ها به منظور مکان‌یابی خطا استفاده می‌کنند دارای شباهت‌هایی با روش‌های پیش‌بینی خطا هستند. در هر دوی این روش‌ها از معیارهایی که منبع استفاده می‌شود تا احتمال وجود خطا

⁴⁵Just

محاسبه شود. دو تفاوت عمده‌ی این دو حوزه این است که اولاً در مکان‌یابی خطا از روش‌های یادگیری ماشین استفاده‌ی چندانی نمی‌شود، ثانیاً در مکان‌یابی خطا وجود خطا به وسیله شکست مورد آزمون یا گزارش خطا محرز شده است. با توجه به شباهت‌های موجود میان این دو حوزه در ادامه چند مقاله که با استفاده از آزمون جهش خطا را مکان‌یابی کرده‌اند، بررسی می‌کنیم.

موون^{۴۶} و همکاران در مقاله‌ی خود بر اساس دو فرض روشی به منظور مکان‌یابی خطا ارائه داده‌اند. فرض اول بیان می‌کند که در یک برنامه‌ی حاوی خطا جهش و یا اصلاح یک عبارت خطا دار نسبت به جهش یک عبارت درست می‌تواند موارد آزمون بیشتری را با موفقیت بگذرانند. فرض دوم بیان می‌کند که جهش عبارات صحیح نسبت به جهش یک عبارت غلط موجب می‌شود موارد آزمون بیشتری شکست بخورند. بر اساس این دو فرض معیاری به نام مشکوک بودن^{۴۷} ارائه گردیده است که دو فرض را فرموله می‌کند. این معیار بر اساس تعداد شکست و موفقیت موارد آزمون در نسخه‌ی اصلی و جهش‌یافته عمل می‌کند. سپس با رتبه‌بندی عبارات بر اساس این معیار عبارت حاوی خطا مشخص می‌گردد. در این پژوهش روش جدیدی نیز به منظور ارزیابی روش پیشنهادی ارائه شده است که برخی از مشکلات روش پیشین را بر طرف نموده است. در نهایت روش مکان‌یابی ارائه شده با دو روش ارزیابی شده و نتایج نشان می‌دهد فرضیات پژوهش درست بوده‌اند [۲۶].

پاپاداکیس و تراون^{۴۸} در مقاله‌ی خود به این نکته اشاره کرده‌اند که استفاده از تحلیل جهش در گذشته به دلیل پرهزینه بودن چندان مورد توجه قرار نمی‌گرفته است اما امروزه با وجود ابزارهای مقیاس پذیر، نمونه‌گیری و انتخاب جهش می‌توان به خوبی از تحلیل جهش در انجام پژوهش‌های مختلف استفاده کرد [۲۷]. آنها روشی را برای مکان‌یابی خطا بر اساس دو مشاهده ارائه کرده‌اند. در مشاهده‌ی اول دیده می‌شود که خطای موجود در یک عبارت رفتار مشابهی با جهش در همان عبارت نشان می‌دهد. در مشاهده‌ی دیگر دیده می‌شود که اگر خطا و جهش در دو عبارت متفاوت باشند رفتار متفاوتی خواهند داشت. منظور از رفتار مشابه موفقیت یا شکست در یک آزمون است. بر اساس این دو مشاهده معیاری برای مشکوک بودن عبارات تعیین می‌گردد. این پژوهش بیان می‌کند که مناسب بودن موارد آزمون تاثیر مستقیمی بر عملکرد روش مکان‌یابی خطا دارد. همچنین یک مجموعه‌ی کوچک از جهش‌یافته‌ها می‌تواند به اندازه‌ی مجموعه‌ای کامل تاثیر گذار باشد.

⁴⁶Moon

⁴⁷Suspiciousness

⁴⁸Papadakis and Traon

۲.۲.۲. مدل‌های یادگیری و جهش‌یافته‌ها

ها/و^{۴۹} و همکاران با ارایه‌ی مجموعه‌ای از معیارها و استفاده از یادگیری ماشین مدلی را ارائه داده‌اند که به وسیله‌ی آن بتوان تشخیص داد علت شکست در آزمون رگرسیون وجود خطا است یا منسوخ^{۵۰} شدن یک مورد آزمون [۲۸]. هفت معیار ارائه شده در این پژوهش مرتبط با گراف فراخوانی، تغییر در فایل‌ها و تعداد شکست در آزمون‌ها بوده است. هاو و همکاران به منظور به دست آوردن مجموعه داده‌ی حاوی خطا، به صورت دستی بر اساس استانداردهایی از پیش تعریف شده خطاهایی را در کد قرار داده‌اند. بدین منظور عباراتی به صورت تصادفی که در سراسر کد محصول قرار دارند انتخاب شدند و به وسیله‌ی عملگرهای جهش خطاهایی تولید شده است. به منظور بدست آوردن آزمون‌های منسوخ شده، مجموعه آزمون‌هایی از نسخه‌ی قبلی برنامه بر روی کد نسخه‌ی بعدی به کار گرفته شده است. سپس با استفاده از روش ارزیابی میان دسته‌ی^{۵۱} به آموزش و آزمایش مدل ساخته شده پرداخته می‌شود. نتایج پژوهش نشان می‌دهد که روش پیشنهادی زمانی که بر روی یک نسخه یا نسخه‌های مختلف از یک برنامه اعمال شود نتایج خوبی دارد (۸۰٪ دقت) اما زمانی که بر روی برنامه‌های مختلف اعمال شود (مجموعه آموزش از یک برنامه و آزمون بر روی برنامه‌ای دیگر) موثر نیست. نتایج نشان می‌دهد تکنیک‌ها مکان‌یابی خطا نتیجه‌ی مثبتی بر تشخیص نوع خطا که مربوط به محصول است یا آزمون، ندارد.

بوئر^{۵۲} و همکاران معیارهایی را مبتنی بر جهش معرفی کردند و از ترکیب آنها با معیارهای سنتی و شی‌گرایی، یک مدل پیش‌بینی ساخته شده است [۲۹]. ۸ عملگر جهش در نظر گرفته شده و برای هر یک از آنها یک معیار ایستا (بدون اجرای کد) و چهار معیار پویا ساخته شده و در مجموع ۴۰ معیار جهش ارائه شده است. به این دلیل میان معیار ایستا و پویا تمایز قائل شده‌اند که اگر معیارهای ایستا به تنهایی پیش‌بینی را بهبود بخشند بدون نیاز به موارد آزمون می‌توان از آنها استفاده کرد، در واقع دامنه‌ی کاربرد روش گسترده‌تر می‌گردد. نتایج پژوهش نشان می‌دهد که استفاده از معیارهای جهش بهبود قابل توجهی را در پیش‌بینی خطا به وجود می‌آورد. همچنین معیارهای پویا و ایستا در کنار یکدیگر توانایی پیش‌بینی مناسبی دارند ولی استفاده‌ی جداگانه از آنها تاثیر چندان مثبتی نخواهد داشت. این پژوهش از دو جنبه حائز اهمیت می‌باشد. یکی اینکه اولین پژوهش در زمینه‌ی پیش‌بینی خطاست که از تحلیل جهش استفاده کرده است. دوم آنکه مشابه‌ترین پژوهش به پژوهش کنونی می‌باشد.

⁴⁹Hao

⁵⁰Obsolete

⁵¹Cross-validation

⁵²Bowes

۳.۲. جمع بندی مطالعات پیشین

هدف از پیش‌بینی خطا کمک به توسعه‌دهندگان نرم‌افزار و کاهش هزینه‌های نرم‌افزاری می‌باشد. روند پیش‌بینی خطا به این صورت است که با استفاده از مخازن نرم‌افزاری همانند سیستم کنترل نسخه و سیستم ردگیری خطا، اطلاعات کد منبع، خطا و اطلاعات تاریخی پروژه جمع‌آوری می‌شود. با توجه به معیارهای مختلف داده‌هایی استخراج می‌شود که هر داده دارای برجسته‌ترین "سالم" یا "حاوی خطا" می‌باشد. قسمتی از این داده‌ها با استفاده از روش‌های یادگیری ماشین، مدل‌های پیش‌بینی خطا را تولید می‌کنند و قسمت دیگر جهت آزمایش مدل به کار گرفته می‌شود.

معیارهای متداول در ارزیابی پیش‌بینی دقت و فراخوانی می‌باشند. این معیارها دارای نواقصی هستند. به عنوان مثال مدلی که همه‌ی داده‌ها را خطا دار معرفی می‌کند دارای فراخوانی برابر یک است و مسلماً این مدل کارایی مناسبی ندارد. معیار اف میانگین هارمونیک دو معیار قبلی است و نواقص آنها را بر طرف می‌کند. یکی از معیارهای رایج برای مقایسه‌ی مدل‌های یادگیری ماشین AUC می‌باشد. هرچه این مساحت بیشتر باشد و منحنی مربوطه سریعتر در راستای محور عمودی به یک برسد مدل کارایی بهتری دارد. با استفاده از معیار AUCEC می‌توان موثر بودن مدل از نظر هزینه را سنجید. معمولاً چند درصد اول از منحنی مربوطه در نظر گرفته می‌شود و مساحت آن محاسبه می‌شود.

معیارهای مورد استفاده را می‌توان به سه دسته‌ی معیار سنتی کد، معیار شیء گرایی و معیار فرآیند تقسیم کرد. در برخی از منابع نیز به دو دسته‌ی کلی معیار کد و معیار فرآیند تقسیم شده‌اند. معیارهای اندازه جزء معیارهای ابتدایی و موثر هستند و معیارهای پیچیدگی و شیء گرایی همبستگی فراوانی با معیارهای اندازه دارند. معیارهای شیء گرایی دارای وابستگی فراوانی با معیارهای اندازه هستند. با این حال معیارهای شیء گرایی دارای توانایی بیشتری هستند. معیارهای فرآیند از جنبه‌های مختلفی مانند عدم رکود در تکرارهای چرخه‌ی تولید نرم‌افزار و موثر بودن از نظر هزینه از سایر معیارها برتری دارد. علی‌رغم توانمندی بالقوه‌ی معیارهای فرآیند در پیش‌بینی خطا، این معیارها در پژوهش‌های کمتری مورد تحقیق قرار گرفته‌اند.

در پژوهش‌های مختلف از روش‌های یادگیری ماشین متفاوتی استفاده شده است. در صورتی که هدف پیش‌بینی تعداد خطاها باشد از رگرسیون و در صورتی که هدف پیش‌بینی حاوی خطا بودن باشد از دسته‌بندی

استفاده می‌شود. پژوهش [۸] نشان داده است که روش دسته‌بندی تاثیر متوسطی بر کارایی پیش‌بینی خطا دارد و انتخاب معیار مهم‌تر است.

در ابتدا از امتیاز جهش برای میزان موثر بودن مجموعه آزمون استفاده می‌شد و سپس کاربردهای دیگری همچون انتخاب، رتبه‌بندی و کمینه کردن مجموعه آزمون پیدا کرده است. همچنین در پژوهش‌های اخیر جهت مکان‌یابی خطا و پیش‌بینی خطا مورد استفاده قرار گرفته است. در پژوهش [۲۵] نشان داده شده است که جهش‌یافته‌هایی که با عملگرهای جهش ساده تولید شده‌اند می‌توانند تا ۷۳٪ خطاهای واقعی را شبیه‌سازی کنند و ازین جهت جایگزین مناسبی برای خطاهای واقعی باشند.

جدول ۲.۲: جدول مشخصات پژوهش‌های مرور شده در حوزه‌ی پیش‌بینی خطا

مقاله	معیار	تکنیک یادگیری	ریزدانگی	روش ارزیابی	نوع پروژه‌ها	زبان پروژه‌ها
[۱۹]	فرآیند - سنتی	NBR	فایل	مشابه AUCEC	خصوصی	جاوا
[۲۰]	فرآیند - سنتی - شی‌گرایی	Naive Bayes - Logestic Regression - SMV - J48	فایل	AUC - AUCEC - F-Measure	متن باز	جاوا
[۲۹]	سنتی - شی‌گرایی	Naive Bayes - Logestic Regression - Random Forest - J48	کلاس	غیره	متن باز	جاوا
[۱۴]	سنتی	LR - ANN - DT - SVM - CCN - GMDH - GEP	NA	AUC - Precision	متن باز	سی
[۳۰]	سنتی - فرآیند	Naive Bayes - DT - kNN - RF	سیستم	AUC - Precision - Recall - F-Measure	متن باز	اندروید
[۳۱]	سنتی - شی‌گرایی	LR - ANN - RBFN	کلاس	Accuracy - F-Measure	متن باز	جاوا

[این صفحه آگاهانه خالی گذاشته شده است.]

فصل ۳

معیارهای جهش و فرآیند

با مطالعات مروری انجام شده نقاطی از این حوزه که نیازمند پژوهش بیشتر هستند تا بتوان به وسیله‌ی آن به ارائه‌ی روشی کارا تر در پیش‌بینی خطا پرداخت مشخص شد. مقاله‌ی [۲۹] اولین مقاله‌ای است که یک روش پیش‌بینی خطا با استفاده از تحلیل جهش ارائه نموده است و این موضوع نیازمند تحقیق بیشتر است. از طرف دیگر بر طبق مقاله‌ی [۹] استفاده از معیارهای فرآیند علی‌رغم توانایی بالقوه‌ای که در پیش‌بینی خطا دارند، در پژوهش‌های کمتری مورد بررسی قرار گرفته‌اند. یکی از دلایل آن می‌تواند نو ظهور بودن این معیارها نسبت به سایرین باشد. معیارهای فرآیند از جنبه‌های مختلف نیز از سایر معیارها برتری دارند [۲۰].

این پژوهش قصد دارد سه رویکرد پیشنهادی را به منظور بهبود پیش‌بینی خطا بررسی کند. این رویکردها عبارتند از:

۱. در این رویکرد معیارهای جهش و معیارهای فرآیند در کنار یکدیگر استفاده می‌شوند و به وسیله‌ی آنها پیش‌بینی انجام می‌گیرد. این دو دسته معیار در پژوهش‌های گذشته مطرح شده‌اند اما تاکنون در کنار یکدیگر قرار نگرفته‌اند.

۲. معیارهای جدیدی مطرح می‌شوند که مبتنی بر مفاهیم آزمون جهش و فرآیند توسعه‌ی نرم‌افزار است.

۳. معیارهای جدیدی مطرح می‌شوند که با کمک مفاهیم جهش سعی در بهبود معیارهای فرآیند دارند.

۱.۳. معیارهای جهش و فرآیند

این رویکرد با توجه به مقاله‌ی [۲۹] مطرح شده که در آن بررسی به کارگیری معیارهای جهش و فرآیند را در پژوهش‌های آتی توصیه می‌کند. همچنین معیار جهش یک معیار مرتبط با کد است. مقاله‌ی [۲۰] بیان می‌کند که معیارهای کد ایستا هستند و تمایل دارند که یک موجودیت را در انتشارهای متوالی حاوی خطا معرفی کنند. حال شرایطی را در نظر بگیرید که که امتیاز جهش در یک موجودیت کم باشد و دلیل آن کافی نبودن

مجموعه آزمون باشد چرا که توسعه‌دهندگان از درست بودن کد اطمینان دارند یا اینکه پس از انتشارهای متوالی خطاها بر طرف شده است. چنین موجودیتی حاوی خطا نیست اما با توجه به معیار جهش خطاخیز است. با در نظر گرفتن معیارهای فرآیند در مورد این موجودیت که نشان می‌دهند پایدار و بدون تغییر است از میزان خطاخیز بودن آن کاسته می‌شود و انتظار می‌رود کارایی مدل پیش‌بینی بهبود یابد. برای پاسخ به این پرسش مجموعه معیارهای جهش از پژوهش [۲۹] و معیارهای فرآیند از پژوهش [۲۰] انتخاب می‌شوند. در جداول ۱.۳ و ۲.۳ معیارهای مورد نظر آورده شده است و در ادامه معرفی شده و

جدول ۱.۳: معیارهای فرآیند [۲۰]

نام معیار	توضیح
۱ COMM	تعداد ثبت در سیستم کنترل نسخه
۲ ADEV	تعداد توسعه‌دهندگان فعال
۳ DDEV	تعداد توسعه‌دهندگان متمایز
۴ ADD	مقدار نرمال‌سازی شده‌ی تعداد خطوط اضافه شده
۵ DEL	مقدار نرمال‌سازی شده‌ی تعداد خطوط حذف شده
۶ OWN	درصد خطوطی که مالک فایل مشارکت کرده
۷ MINOR	تعداد مشارکت‌کنندگان جزئی
۸ NCOMM	تعداد ثبت‌های همسایگان
۹ NADEV	تعداد توسعه‌دهندگان فعال همسایگان
۱۰ NDDEV	تعداد توسعه‌دهندگان متمایز همسایگان
۱۱ OEXP	تجربه‌ی مالک فایل
۱۲ AEXP	تجربه‌ی تمام مشارکت‌کنندگان

جدول ۲.۳: معیارهای جهش [۲۹]

نام معیار	توضیح
۱ MuNOM	تعداد جهش‌یافته‌های تولید شده
۲ MuNOC	تعداد جهش‌یافته‌های پوشش داده شده توسط آزمون‌ها
۳ MuNMS	امتیاز جهش‌یافته‌های تولید شده
۴ MuNMSC	امتیاز جهش‌یافته‌های پوشش داده شده توسط آزمون‌ها

از آنجا که در این پژوهش پیش‌بینی‌ها در سطح فایل انجام می‌شود، معیارها برای هر فایل جداگانه محاسبه می‌شوند. در ادامه هر یک از معیارهای فرآیند معرفی و نحوه‌ی محاسبه‌ی آن‌ها بیان می‌شود. معیارهای جهش به طور مستقیم توسط ابزارهای موجود محاسبه می‌گردد.

۱. **تعداد ثبت در سیستم کنترل نسخه:** تعداد ثبت‌هایی که در آن فایل مورد نظر در طول انتشار قبلی تاکنون تغییر کرده است. برای محاسبه‌ی آن لازم است که تمام ثبت‌های پروژه بین ثبت کنونی و انتشار قبلی بررسی شود و ثبت‌هایی که در آن این فایل تغییر کرده‌اند شمرده شوند.
۲. **تعداد توسعه‌دهندگان فعال:** تعداد توسعه‌دهندگانی که در طول انتشار قبلی تا کنون (زمان ثبت) فایل را تغییر داده‌اند. لازم است ثبت‌های موجود در بازه‌ی زمانی خواسته شده بررسی شود و آنها که فایل مورد نظر را تغییر داده‌اند انتخاب شوند. نام کسانی که ثبت را انجام داده‌اند بازایی شود و تعداد نامهای متمایز شمرده شود.
۳. **تعداد توسعه‌دهندگان متمایز:** مشابه معیار قبلی با این تفاوت که در طول انتشار محاسبه نمی‌شود. بلکه از ابتدای پروژه تا زمان ثبت در نظر گرفته می‌شود.
۴. **مقدار نرمال‌سازی شده‌ی تعداد خطوط اضافه شده:** این معیار تعداد خطوط اضافه شده در یک فایل را در طول انتشار قبلی می‌شمارد. سپس جهت نرمال سازی آنرا بر تعداد کل خطوط اضافه شده در پروژه در طول انتشار قبلی تقسیم می‌کند. برای بدست آوردن تعداد خطوط اضافه شده در یک فایل هر ثبت نسبت به ثبت قبلی مقایسه می‌شود و تعداد خطوط اضافه شده جمع زده می‌شود.
۵. **مقدار نرمال‌سازی شده‌ی تعداد خطوط حذف شده:** مشابه معیار قبلی می‌باشد.
۶. **تعداد خطوطی که مالک فایل مشارکت کرده:** درصد خطوطی از فایل، در ثبت مورد نظر که به مالک فایل تعلق دارد. مالک فایل کسی است که در آن لحظه از زمان بیشترین تعداد خطوط موجود در فایل به او تعلق دارد. ابتدا نویسنده‌ی هر خط مشخص می‌شود سپس برای هر نویسنده تعداد خطوطی که به وی تعلق دارد شمرده می‌شود. تعداد خطوط مالک فایل بر تعداد خطوط فایل تقسیم می‌گردد.
۷. **تعداد مشارکت‌کنندگان جزئی:** توسعه‌دهنده‌ی جزئی کسی است که کمتر از ۵٪ خطوط موجود در فایل به او تعلق داشته باشد. بدین منظور نویسنده‌ی هر خط مشخص می‌شود. تعداد خطوط هر نویسنده شمرده می‌شود و بر تعداد خطوط فایل تقسیم می‌شود. سپس تعداد نویسندگانی که کمتر از ۵٪ مشارکت داشته‌اند شمرده می‌شود.
۸. **تعداد ثبت‌های همسایگان میانگین وزن دهی شده:** تعداد ثبت‌های همسایگان فایل از انتشار قبلی تا کنون را اندازه‌گیری می‌کند. همسایگان یک فایل در یک ثبت، فایل‌هایی هستند که در آن نسخه از

برنامه تغییر کرده‌اند. درواقع در هر ثبت از برنامه تعدادی فایل نسبت به ثبت قبلی تغییر کرده‌اند که این فایل‌ها همسایه‌ی یکدیگر محسوب می‌شوند. نحوه‌ی وزن دهی نیز به این صورت است که هرچقدر یک فایل تعداد دفعات بیشتری را در طول انتشار با فایل مورد نظر همسایه شده باشد وزن بیشتری می‌یابد. برای محاسبه ابتدا همسایگان فایل در ثبت و تعداد دفعاتی که در طول انتشار همسایه شده‌اند مشخص می‌شوند. سپس برای هر فایل همسایه، معیار تعداد ثبت در سیستم کنترل نسخه محاسبه می‌شود. هر معیار در تعداد دفعاتی همسایگی ضرب می‌شود و با هم جمع زده می‌شوند. در انتها بر تعداد کل دفعات همسایگی همسایگان تقسیم می‌شود.

۹. **تعداد توسعه‌دهندگان فعال همسایگان:** مشابه معیار قبلی عمل می‌شود با این تفاوت که معیار توسعه‌دهندگان فعال در نظر گرفته خواهد شد.

۱۰. **تعداد توسعه‌دهندگان متمایز همسایگان:** مشابه معیار قبلی عمل می‌شود با این تفاوت که معیار توسعه‌دهندگان متمایز در نظر گرفته خواهد شد.

۱۱. **تجربه‌ی مالک فایل:** ابتدا لازم است که نحوه‌ی محاسبه تجربه را تعریف کنیم. هرچقدر یک فرد تعداد تغییرات بیشتری را در یک پروژه انجام دهد تجربه بیشتری را در آن پروژه دارد و ثبت را می‌توان به ایجاد تغییر تعبیر کرد. برای محاسبه‌ی معیار ابتدا مالک فایل مشخص می‌شود. سپس تعداد ثبت‌هایی که مالک فایل از ابتدای پروژه تا زمان مورد نظر انجام داده، شمرده می‌شود.

۱۲. **تجربه‌ی تمام مشارکت‌کنندگان:** تمام مشارکت‌کنندگان در فایل تا زمان ثبت مورد نظر یافت می‌شوند. برای هر یک مشابه معیار قبلی تجربه، محاسبه می‌شود و از مقدار تجربه‌ها میانگین هندسی گرفته می‌شود.

۲.۳. معیارهای جهش مبتنی بر فرآیند

در رویکرد دوم، چهار معیار جدید در این پژوهش معرفی می‌شوند که با استفاده از مفاهیم آزمون جهش و تاریخچه‌ی توسعه‌ی نرم‌افزار ساخته می‌شوند. از این رو این معیارها معیارهای جهش مبتنی بر فرآیند^۱ نامیده شده‌اند.

۱. **تعداد جهش‌یافته‌های تولید شده‌ی جدید نسبت به انتشار قبلی برنامه:** همانطور که در مقاله‌ی

[۲۵] مطرح شده جهش‌یافته‌ها جایگزین خوبی برای خطاهای واقعی می‌باشند. زمانی که تعداد

^۱Process Based Mutation Metrics (PBMM)

جهش‌یافته‌های جدید زیاد باشد یعنی تغییراتی که خطاخیزتر هستند بیشتر است. به منظور محاسبه‌ی این معیار لازم است خطوط اضافه شده به فایل مورد نظر در ثبت کنونی، نسبت به انتشار قبلی مشخص شود و سپس تعداد جهش یافته‌هایی که این خطوط تولید می‌کنند شمرده شوند.

۲. **تعداد جهش‌یافته‌های متمایز در چند انتشار اخیر:** این معیار نشان می‌دهد موجودیت مورد بررسی به چه میزان سابقه‌ی تغییراتی را دارد که احتمال بروز خطا را افزایش می‌دهد. تعداد انتشارها باید به گونه‌ای باشد که کم یا زیاد نباشد. زیرا تعداد انتشارهای کم سبب می‌شود تفاوت جندانی با معیار قبلی نداشت باشد و سابقه‌ی تغییرات به اندازه‌ی کافی مد نظر قرار نگیرد. از طرف دیگر در نظر گرفتن تعداد زیادی انتشار هم هزینه‌بر است و هم به دلیل تغییرات زیاد فایل در طول توسعه‌ی نرم‌افزار اطلاعات اولیه مفید نخواهد بود. عدد در نظر گرفته شده برای تعداد انتشارها چهار می‌باشد. نحوه‌ی محاسبه به این شکل است که برای هر انتشار تعداد جهش‌یافته‌ها در انتشار جدید، نسبت به قبلی شمرده می‌شود و با یکدیگر جمع زده می‌شوند.

۳. **میزان تغییرات مثبت امتیاز جهش در چند انتشار اخیر:** تغییرات امتیاز جهش نشان از تغییرات در برنامه و آزمون‌های نرم‌افزار است. این معیار نشان می‌دهد این تغییرات به چه میزان در جهت بهبود کیفیت نرم‌افزار بوده. چراکه امتیاز بالاتر جهش نشان از کیفیت بهتر آزمون‌ها و در نتیجه نرم‌افزار است. به منظور محاسبه‌ی این معیار در هر انتشار امتیاز جهش محاسبه می‌شود و در صورتی که نسبت به انتشار قبلی تغییر مثبت بود به مجموع تغییرات افزوده می‌شود.

۴. **میزان تغییرات منفی امتیاز جهش در چند انتشار اخیر:** این معیار مشابه معیار سوم عمل می‌کند با این تفاوت که میزان تغییرات در خلاف جهت بهبود نرم‌افزار را می‌سنجد.

۳.۳. معیارهای ترکیبی جهش-فرآیند

رویکرد سوم با توجه به مطالب گفته شده در مقاله‌ی [۲۰] مطرح شده که بیان می‌کند معیارها هر چقدر هم که پویا باشند (دچار رکود نشوند، مانند معیارهای فرآیند) زمانی در پیش‌بینی خطا مفید هستند که همراه با ایجاد خطا باشند. نکته‌ی قابل توجه این است که همه‌ی تغییرات در یک فایل به یک اندازه موجب بر پیچیدگی فایل نمی‌افزایند و به عبارت دیگر موجب بروز خطا نمی‌شوند. به عنوان مثال در یک فایل به زبان جاوا ممکن است توضیح^۲ و یا مستندجاوا^۳ وجود داشته باشد که بروزرسانی یا اضافه و کم شدن آنها تاثیری بر روند اجرای برنامه

^۲Comment

^۳Javadoc

و میران پیچیدگی ندارند با این حال در محاسبه‌ی معیارهای پیش‌بینی خطا در نظر گرفته می‌شوند. هدف از ارائه‌ی معیارهای ترکیبی جهش-فرآیند^۴ بهبود کاستی‌های معیارهای فرآیند در چنین شرایطی است. در اینجا دو معیار مقدار نرمال شده‌ی خطوط اضافه شده و یا کم شده است. این دو معیار جز شاخص‌ترین معیارهای فرآیند هستند.

در نگاه اول این ایده به ذهن می‌رسد که با توجه به تعداد جهش‌یافته‌هایی که اضافه و یا حذف هر خط ایجاد می‌کند، اضافه یا کم شدن خطوط وزن دهی شود و به منظور اجرای آن از دو فرمول زیر بهره گرفت.

$$M_1 = \text{number of lines added} \times \text{number of mutants derived}$$

$$M_2 = \text{number of lines deleted} \times \text{number of mutants derived}$$

با وجود مناسب بودن ایده‌ی اولیه با بررسی‌های بیشتر دو مشکل در معیارهای فوق مشخص می‌شود. مشکل اول: هدف از ارائه‌ی این معیارها وزن دهی به خطوط اضافه و کم شده است. نکته قابل توجه این است که هر خط باید به صورت جداگانه وزن دهی شود و وزن یک خط بر وزن خط دیگر تأثیری نداشته باشد. مثال زیر را در نظر بگیرید.

```
//this method is important → 0 mutant
// this method get root of → 0 mutant
// sum of a plus b → 0 mutant
b = sqrt(a+b) → 2 mutant
```

فرض کنید ۴ خط بالا به یک فایل اضافه شده است. معیار مقدار نرمال شده‌ی خطوط اضافه شده قبل از نرمال سازی عدد چهار را نمایش می‌دهد در حالی که از این چهار خط ۳ خط توضیح است. حال معیار اولیه پیشنهادی برابر ۸ خواهد بود که بدیهی است، از هدف ارایه‌ی متریک فاصله گرفته است. حال اگر تنها جهش یافته‌های تولید شده در خطوط اضافه شده را در نظر بگیریم این مقدار می‌تواند جایگزین مناسبی باشد. در واقع نگاشتی را ارائه می‌شود که هر خط از برنامه را به یک عدد نگاشت می‌دهد. این عدد میزان پیچیدگی آن خط و یا احتمال بروز خطا را تعریف می‌کند. لازم به یادآوری است که در مقاله‌ی [۲۵] اشاره شده که جهش یافته‌ها جایگزین خوبی برای خطاهای واقعی هستند. این نگاشت برابر است با تعداد جهش یافته‌های تولید شده در آن خط.

مشکل دوم: این معیار برای عمل‌کرد هرچه بهتر مشابه معیار مقدار نرمال شده‌ی خطوط اضافه شده نیاز به نرمال سازی دارد. به جهت نرمال سازی نمی‌توان از همان روش استفاده کنیم چراکه در آن وزن دهی به خطوط

⁴Process-Mutation Hybrid

وجود ندارد و از آن مهم‌تر توضیحات را نیز در نظر می‌گیرد. از طرف دیگر این امکان وجود ندارد که برای تمام خطوط اضافه یا کم شده در کل پروژه در طول یک انتشار جهش یافته تولید شود (به دلیل زمانبر بودن و پیچیدگی‌های فراوان در پیاده‌سازی). در مقالات گذشته اشاره شده که تعداد ثبت‌ها می‌تواند نشانگر میزان تغییرات باشد. بنابراین از تعداد ثبت‌های کل پروژه در طول یک انتشار به منظور نرمال‌سازی استفاده خواهد شد.

[این صفحه آگاهانه خالی گذاشته شده است.]

فصل ۴

مورد مطالعاتی

در این فصل مطالعه‌ی موردی بر روی مجموعه داده‌ی defects4j [۳۲] انجام می‌گیرد. ابتدا نحوه‌ی کلی برپایی آزمایش شرح داده می‌شود و سپس چگونگی استخراج معیارها و پیاده‌سازی آزمایش توضیح داده خواهد شد.

۱.۴. طراحی آزمایش

به منظور ارزیابی رویکردهای گفته شده لازم است که برای مجموعه معیارهای هر رویکرد مدلهای پیش‌بینی ساخته شود و هر عملکرد هر مدل نسبت به پژوهش‌های گذشته مقایسه شود. به این ترتیب ابتدا لازم است از مجموعه داده‌ی فراهم شده معیارهای بیان شده در فصل؟؟ استخراج شوند. مجموعه داده‌ی defects4j که در قسمت‌های آتی معرفی می‌شود شامل اطلاعات خطا در چندین فایل است و به همین تعداد، فایل بدون خطا در ثبت و پروژه‌ی متناظر به طور تصادفی انتخاب می‌گردد. برای فایل‌های حاوی خطا و سالم، معیارها استخراج می‌شود. معیارهای استخراج شده برای هر فایل به عنوان بردار ویژگی در مدلهای دسته‌بندی عمل می‌کند. مدلهای دسته‌بندی به منظور پیش‌بینی حاوی خطا بودن ساخته می‌شود و عملکرد آنها مقایسه می‌گردد. مدلهایی که با هم مقایسه می‌شود در الگوریتم و پیکربندی^۱ یکسان هستند و تنها تفاوت آنها در معیارهای استفاده شده به منظور یادگیری است. بدین ترتیب تاثیر معیارها بر پیش‌بینی خطا سنجیده می‌شود.

۲.۴. آشنایی با ابزارها و مجموعه داده

این قسمت به معرفی ابزارهای استفاده شده در این پژوهش می‌پردازد. آشنایی با این ابزارها به درک هرچه بهتر نحوه‌ی استخراج معیارها و روند آزمایش کمک می‌کند.

۱.۲.۴. مجموعه داده defect4j

مجموعه داده‌ی انتخابی به منظور انجام مورد مطالعاتی لازم است که دارای ویژگی‌های زیر باشد:

¹Configuration

- اطلاعات خطاهای پروژه وجود داشته باشد و این اطلاعات نشان دهد که خطا متعلق به کدام فایل در کدام ثبت است.

- پروژه‌ها متن-باز باشد تا بتوان با استفاده از کد منبع آنها معیارها را استخراج نمود.

- برای پروژه‌ها موارد آزمون مناسب وجود داشته باشد تا بتوان معیارهای جهش را استخراج کرد.

در میان مجموعه‌داده‌های موجود مجموعه‌داده‌ی defects4j تنها موردی است که تمام ویژگی‌ها را دارد.

این مجموعه شامل شش پروژه می‌باشد که این پروژه‌ها متن-باز^۲ هستند و با استفاده از نرم‌افزارهای کنترل نسخه‌ی گیت و svn می‌توان به کدهای آن‌ها در طول فرآیند توسعه‌ی آنها دسترسی پیدا کرد. بجز پروژه‌ی Chart سایرین از سیستم گیت استفاده می‌کنند. همچنین این پروژه به دلیل نداشتن ساختار مناسب کنار گذاشته شد و از پرونده‌های حاوی خطای موجود در آن استفاده نشد. مجموعه‌داده‌ی defects4j به صورت یک چهارچوب^۳ ارائه شده است که کارهایی بیش از نگهداری اطلاعات درباره‌ی پروژه‌ها انجام می‌دهد. مهم‌ترین کارهایی که می‌توان به وسیله‌ی این ابزار انجام داد در جدول ۱.۴ آمده است.

جدول ۱.۴: عملیات‌های موجود در defects4j

نام عملیات	توضیح
info	نمایش پیکربندی یک پروژه‌ی خاص یا خلاصه‌ی یک خطای خاص
checkout	وارسی یک نسخه‌ی حاوی خطا یا تعمیر شده از پروژه
compile	کامپایل کدها و آزمون‌های نوشته شده توسط توسعه‌دهندگان
test	اجرای یک آزمون یا مجموعه‌ی آزمون در یک نسخه‌ی حاوی خطا یا تعمیر شده از پروژه
mutation	اجرای تحلیل جهش در یک نسخه‌ی حاوی خطا یا تعمیر شده از پروژه

این ابزار در اجرای عملیات‌های بالا دارای محدودیت است و تنها آن‌ها را بر روی ثبت‌های از پیش تعیین شده انجام می‌دهد. ثبت‌های از پیش تعیین شده شامل ثبت‌های حاوی خطا و تعمیر آن خطا می‌باشد. در جدول ۲.۴ اطلاعات مربوط به تعداد خطاهای هر پروژه آمده است.

^۲Open-source

^۳Framework

جدول ۲.۴: پروژه‌های موجود در defects4j

نام مختصر	نام کامل	تعداد خطا
Chart	JFreeChart	۲۶
Closure	Closure compiler	۱۳۳
Lang	Apache commons-lang	۶۵
Math	Apache commons-math	۱۰۶
Mockito	Mockito	۳۸
Time	Joda - Time	۲۷
-	کل پروژه‌ها	۳۹۵

به منظور نصب و راه اندازی ابزار defects4j ابتدا از صفحه‌ی گیت‌هاب^۴ آن کدهای مربوطه دریافت می‌شود. سپس باید دستوراتی را اجرا کرد تا سایر متعلقات دریافت شود. این تعلقات شامل مخزن نرم‌افزاری مربوط به شش پروژه‌ی یاد شده است که کدهای پروژه‌ها در آن قرار دارد. نکته‌ی قابل توجه در این ابزار این است که بجز دستور info سایر دستورات عملیات‌های مربوط را بر روی کامپیوتر کاربر انجام می‌دهد و خروجی را نمایش داده می‌شود، نه اینکه از یک پایگاه داده اطلاعات را صرفاً بارخوانی کند.

در نیازمندی‌های این ابزار اشاره شده که باید از جاوا نسخه‌ی ۷ استفاده شود. اما مسأله‌ای که به آن اشاره نشده توزیع کننده‌ی جاوا است. جاوا دو توزیع کننده‌ی عمده دارد. یکی OpenJDK و دیگر با Oracle استفاده از OpenJDK ابزار defects4j و ابزارهایی که به آن وابسته است به خوبی کار نمی‌کنند. به عنوان مثال برخی مجموعه آزمون‌ها که باید بدون خطا اجرا شوند به دلیل نبود وابستگی‌های^۵ لازم با شکست مواجه می‌شوند.

راه ارتباط با این ابزار خط دستور^۶ می‌باشد و یک نمونه از دستورات قابل استفاده در این ابزار در شکل ۱.۴ است که این دستور اطلاعات مربوط به پروژه‌ی Lang خطای شماره‌ی یک را خواهد داد.

defects4j info -p Lang -b 1

^۴Github

^۵Dependency

^۶Command line

```

ali@ali-GL553VD ~ $ defects4j info -p Lang -b 1
Determine revision date..... OK
Summary of configuration for Project: Lang
-----
Script dir: /home/ali/project/defects4j/framework
Base dir: /home/ali/project/defects4j
Major root: /home/ali/project/defects4j/major
Repo dir: /home/ali/project/defects4j/project_repos
-----
Project ID: Lang
Program: commons-lang
Build file: /home/ali/project/defects4j/framework/projects/Lang/Lang.build.xml
-----
Vcs: Vcs::Git
Repository: /home/ali/project/defects4j/project_repos/commons-lang.git
Commit db: /home/ali/project/defects4j/framework/projects/Lang/commit-db
Number of bugs: 65
-----
Summary for Bug: Lang-1
-----
Revision ID (fixed version):
687b2e62b7c6e81cd9d5c872b7fa9cc8fd3f1509
-----
Revision date (fixed version):
2013-07-26 01:03:52 +0000
-----
Root cause in triggering tests:
- org.apache.commons.lang3.math.NumberUtilsTest::TestLang747
--> java.lang.NumberFormatException: For input string: "80000000"
-----
List of modified sources:
- org.apache.commons.lang3.math.NumberUtils
-----

```

شکل ۱.۴: اجرای دستور info در defects4j

۲.۲.۴. ابزار Major

این ابزار جهت تولید جهش یافته و تحلیل جهش استفاده می‌شود. یک ابزار دیگر در این حوزه پیت^۷ می‌باشد اما به دلیل سازگاری ابزار defects4j با Major و نیز قابلیت‌های ویژه‌ی آن از این ابزار استفاده شد. چند مورد از ویژگی‌های مهم ابزار Major عبارتند از:

- راحتی استفاده به دلیل نیاز به دستورات کمتر نسبت به پیت
- امکان اجرای تحلیل جهش در پروژه‌هایی که از گریدل^۸ استفاده می‌کنند
- مجموعه عملگرهای کاملتر
- انعطاف در پیکربندی: امکان انجام تحلیل تنها برای یک کلاس یا تابع، تنظیمات ساده و کامل جهت مشخص کردن مجموعه عملگرها

لازم به ذکر است که این ابزار از کامپایلر مخصوص به خود جهت کامپایل برنامه و ساخت جهش یافته استفاده می‌کند که گسترش یافته‌ی یک کامپایلر جاوا است. استفاده از این ابزار را می‌توان در سه مرحله خلاصه کرد:

^۷PIT - <http://pitest.org/>

^۸Gradle

۱. **پیکربندی تولید جهش یافته به وسیله دستورات MML:** این ابزار برای مشخص نمودن اینکه از چه عملگرهایی استفاده شود و آن‌ها در چه محل‌هایی از برنامه به کار گرفته شوند یک زبان ساده ابداع کرده است به نام MML که یک کامپایلر نیز دارد. ابتدا کد MML نوشته می‌شود سپس با MMLC کامپایل می‌شود و نتیجه به عنوان یکی از پارامترها به در هنگام فراخوانی ابزار ارسال می‌شود. نمونه‌ای از این کد در شکل ۲.۴ آمده است.

```
1 targetOp{
2     // Define the replacements for ROR
3     BIN(>)->{>=,!=,FALSE};
4     BIN(<)->{<=,!=,FALSE};
5     BIN(>=)->{>==,TRUE};
6     BIN(<=)->{<==,TRUE};
7     BIN(==)->{<=,>=,FALSE,LHS,RHS};
8     BIN(!=)->{<,>,TRUE,LHS,RHS};
9     // Define the replacements for COR
10    BIN(&&)->{==,LHS,RHS,FALSE};
11    BIN(||)->{!=,LHS,RHS,TRUE};
12    // Define the type of statement that STD should delete
13    DEL(RETURN);
14
15    // Enable the STD, COR, and ROR mutation operators
16    STD;
17    COR;
18    ROR;
19 }
20 // Call the defined operator group for the target method
21 targetOp<"triangle.Triangle::classify(int,int,int)">;
```

شکل ۲.۴: نمونه کد MML در Major

۲. **تولید جهش یافته‌ها:** همانطور که اشاره شد ابزار Major جهت تولید نسخ جهش یافته نیار به کامپایلر پروژه دارد. امروزه پروژه‌های نرم‌افزاری از جمله پروژه‌های موجود در defects4j از ابزارهایی استفاده می‌کنند که فرآیند ساخت را خودکارسازی می‌کنند. فرآیند ساخت به طور کلی شامل مراحل زیر است:

- پاک سازی پوشه‌های کاری از پرونده‌های ساخت‌های قبلی
- معرفی وابستگی‌ها و کامپایل پروژه
- معرفی وابستگی‌ها و کامپایل موارد آزمون
- اجرای موارد آزمون و ارائه گزارش

سه نوع از مهم ترین ابزارهای خودکارسازی مورد استفاده در صنعت عبارتند از Ant ، Maven و Gradle. در پروژه‌های مورد مطالعه نیز این سه نوع به کار گرفته شده است. هر یک از روشهای

خودکارسازی دارای دستورات مربوط به خود می‌باشد و برای تولید جهش‌یافته باید متناسب با آنها عمل نمود که در زیر خلاصه شده است.

- **Ant** : این دسته از پروژه‌ها دارای یک پرونده به نام `build.xml` است که دستورات لازم جهت پیکربندی و انجام عملیات ساخت در آن قرار دارد. به منظور تولید جهش‌یافته کافیت کامپایلر مورد استفاده در قسمت کامپایل پروژه را کامپایلر توسعه‌یافته‌ی `Major` قرار داد و پارامترهای لازم به آن ارسال شود.

- **Maven** : در این دسته از پروژه‌ها دستورات لازم در پرونده‌ی `pom.xml` قرار دارد. به وسیله‌ی یک افزونه^۹ در ابزار `Maven` این فایل تبدیل به یک فایل `build.xml` می‌شود که قابل استفاده توسط ابزار `Ant` است. پس از این تبدیل مشابه حالت قبل عمل می‌شود.

- **Gragle** : در این دسته از پروژه‌ها دستورات لازم در پرونده‌ی `build.gradle` قرار دارد. به منظور تولید جهش‌یافته کافیت کامپایلر مورد استفاده در قسمت کامپایل پروژه را کامپایلر توسعه‌یافته‌ی `Major` قرار داد و پارامترهای لازم به آن ارسال شود.

نمونه‌ای از حاصل اجرای عملیات جهش برای یک پرونده در شکل ۳.۴ آمده است که نشان می‌دهد ۸۶ جهش یافته تولید شده است. همچنین ابزار یک پرونده به نام `mutants.log` تولید می‌کند که نشان می‌دهد چه جهش‌یافته‌هایی در کجا تولید شده‌اند. نمونه‌ای از محتویات این پرونده در شکل ۴.۴ آمده است.

```
Compiling and mutating project
(ant -DmutOp="$MAJOR_HOME/mml/tutorial.mml.bin" clean compile)
Buildfile: /home/ali/project/defects4j/major/example/ant/build.xml

clean:
  [delete] Deleting directory /home/ali/project/defects4j/major/example/ant/bin

init:
  [mkdir] Created dir: /home/ali/project/defects4j/major/example/ant/bin

compile:
  [javac] Compiling 1 source file to /home/ali/project/defects4j/major/example/ant/bin
  [javac] #Generated Mutants: 86 (66 ms)

BUILD SUCCESSFUL
Total time: 1 second
```

شکل ۳.۴: اجرای عملیات جهش برای یک پرونده

^۹Plugin

```
1 1:ROR:<=(int,int):<(int,int):triangle.Triangle@classify(int,int,int):11:a <= 0 |==> a < 0
2 2:ROR:<=(int,int):==(int,int):triangle.Triangle@classify(int,int,int):11:a <= 0 |==> a == 0
3 3:ROR:<=(int,int):TRUE(int,int):triangle.Triangle@classify(int,int,int):11:a <= 0 |==> true
4 4:ROR:<=(int,int):<(int,int):triangle.Triangle@classify(int,int,int):11:b <= 0 |==> b < 0
5 5:ROR:<=(int,int):==(int,int):triangle.Triangle@classify(int,int,int):11:b <= 0 |==> b == 0
```

شکل ۴.۴: نمونه‌ای از پرونده‌ی mutants.log

۳. اجرای تحلیل جهش: ابتدا پرونده‌های آزمون کامپایل می‌شود و سپس هر مجموعه تست بر روی جهش‌یافته‌هایی که تا کنون کشته نشده‌اند اجرا می‌شود. در پایان نتایج را در خروجی چاپ می‌کند. همچنین نتایج را در فایل‌های با پسوند CSV قرار می‌دهد. نمونه‌ای از اجرای تحلیل جهش در شکل ۵.۴ و پرونده‌ی نتایج خروجی در شکل ۶.۴ آمده است.

```
compile.tests:
[javac] Compiling 3 source files to /home/ali/project/defects4j/major/example/ant/bin

mutation.test:
[echo] Running mutation analysis ...
[junit] MAJOR: Mutation analysis enabled
[junit] MAJOR: -----
[junit] MAJOR: Run 3 ordered tests to verify independence
[junit] MAJOR: -----
[junit] MAJOR: Preprocessing time: 0.06 seconds
[junit] MAJOR: -----
[junit] MAJOR: Mutants generated: 86
[junit] MAJOR: Mutants covered: 86 (100.00%)
[junit] MAJOR: -----
[junit] MAJOR: Export test map to (testMap.csv)
[junit] MAJOR: -----
[junit] MAJOR: Run mutation analysis with 3 individual tests
[junit] MAJOR: -----
[junit] MAJOR: 1/3 - triangle.test.TestSuite (3ms / 86):
[junit] MAJOR: 312 (76 / 86 / 86) -> AVG-RTPM: 3ms
[junit] MAJOR: Mutants killed / live: 76 (76-0-0) / 10
[junit] MAJOR: -----
[junit] MAJOR: 2/3 - triangle.test.TestSuite2 (1ms / 86):
[junit] MAJOR: 545 (0 / 86 / 86) -> AVG-RTPM: 2ms
[junit] MAJOR: Mutants killed / live: 76 (152-0-0) / 10
[junit] MAJOR: -----
[junit] MAJOR: 3/3 - triangle.test.TestSuite3 (1ms / 86):
[junit] MAJOR: 737 (0 / 86 / 86) -> AVG-RTPM: 2ms
[junit] MAJOR: Mutants killed / live: 76 (228-0-0) / 10
[junit] MAJOR: -----
[junit] MAJOR: Summary:
[junit] MAJOR: -----
[junit] MAJOR: Analysis time: 0.7 seconds
[junit] MAJOR: Mutation score: 88.37% (88.37%)
[junit] MAJOR: Mutants killed / live: 76 (228-0-0) / 10
[junit] MAJOR: Mutant executions: 258
[junit] MAJOR: -----
[junit] MAJOR: Export summary of results (to summary.csv)
[junit] MAJOR: Export run-time results (to results.csv)
[junit] MAJOR: Export mutant kill details (to killed.csv)
[junit] MAJOR: Export kill map (to km.csv)!

BUILD SUCCESSFUL
Total time: 1 second
```

شکل ۵.۴: اجرای تحلیل جهش

	A	B	C	D	E	F
1	MutantsGenerated	MutantsCovered	MutantsKilled	MutantsLive	RuntimePreprocSeconds	RuntimeAnalysisSeconds
2	86	86	76	10	0.06	0.7
3						
4						

شکل ۴.۶: نتایج خروجی تحلیل جهش

۳.۲.۴. کتابخانه‌ی Jgit

این کتابخانه جهت کار با مخازن نرم افزاری از نوع گیت هستند به کار گرفته می‌شود و به زبان جاوا است. تمام عملیات‌های مهم و اساسی که در نرم‌افزار اصلی گیت وجود دارد در این کتابخانه نیز قابل انجام است. مشکلی که کار با این کتابخانه دارد نبود منابع آموزشی به اندازه‌ی کافی است. چراکه کاربران زیادی ندارد و آموزش‌های ابتدایی معمولاً نیازهای عموم کاربران را برطرف می‌کند.

۴.۲.۴. چهارچوب Hibernate

به وسیله‌ی این چارچوب می‌توان اشیاء موجود در برنامه‌ی جاوا را به داده‌های موجود در پایگاه داده تبدیل کرد. اصطلاحاً به این ابزارها *ORM*^{۱۰} می‌گویند. در ابتدا تصمیم بر این بود که داده‌های بدست آمده در فایل متنی ذخیره شوند و در هنگام نیاز آن‌ها خوانده شوند یا همه‌ی اشیاء با هر بار اجرا ساخته شوند نکات زیر سبب شد که هزینه‌ی اول کار با پایگاه داده و مزایای بلند مدت آن به سادگی استفاده از فایل متنی ترجیح داده شود.

۱. هر بار ساخت اشیاء با اجرای برنامه بسیار زمانبر است و اتلاف وقت زیادی دارد.
۲. لازم است برای اطمینان از درستی برنامه، داده‌ها در قالب جداولی به صورت چشمی کنترل شوند.
۳. فراخوانی و جستجو در پایگاه داده سریع است و کارایی بالا می‌رود.
۴. نگهداری از برنامه در دراز مدت راحت‌تر خواهد بود و خوانایی کدها بیشتر خواهد بود چرا که کار با پایگاه داده دارای اصول مشخصی است و سایرین از آن اطلاع دارند اما فایل متنی اینگونه نیست.

۳.۴. نکات پیاده‌سازی پروژه

پیاده‌سازی پروژه در زبان جاوا انجام گرفت. یکی از نکات مهم و قابل توجه در پیاده‌سازی این پروژه این است که تمام مراحل انجام کار به طور کاملاً خودکار انجام شود و در هیچ مرحله‌ای نیاز به دخالت عامل خارجی ندارد بجز پیکربندی اولیه مانند آدرس پایگاه داده. همچنین در تمام مراحل سعی شده است که تمام اصول لازم

¹⁰Object Relational Mapping

در طراحی معماری نرم‌افزار به کار گرفته شود و نیازمندی‌های کیفی پروژه نیز مد نظر قرار گیرد. این نیازمندی‌ها شامل موارد زیر است:

۱. کارایی^{۱۱}: جهت پاسخ به این نیازمندی از پایگاه داده استفاده شده است.
۲. قابلیت نگهداری: این قابلیت از سایرین بیشتر حائز اهمیت است. زیرا پروژه‌های پژوهشی خود به صورت مستقیم کاربران عمومی ندارند و از این جهت نیازمند کارایی بالا یا رابط گرافیکی کاربر پسند نیستند. استفاده آن‌ها معمولاً در گسترش آن‌ها توسط سایر محققین است که راه را ادامه خواهند داد.
 - برای پاسخ به این نیازمندی اصول مربوط به کدنویسی در فصل سوم و چهارم کتاب [۳۳] به کار گرفته شده است.
 - از الگوهای نرم‌افزاری پر کاربرد مانند /دپتور^{۱۲}، فکتوری^{۱۳} و سینگلتون^{۱۴} استفاده شده است.
 - به منظور جلوگیری از قطعه کد تکراری از وراثت و توابع عمومی^{۱۵} استفاده شده است. همین‌طور عمق وراثت از عدد ۳ بیشتر نشده است زیرا وراثت عمیق از خوانایی کد می‌کاهد و محل اشتباه خواهد بود.
۳. امنیت: از آنجا که پروژه قرار نیست به استفاده ی عموم برسد و کاربران عمل متخاصمانه‌ای انجام نخواهند داد به نوع خاصی از امنیت نسبت به انواع متداول دارد. باید روند توسعه‌ی پروژه دارای امنیت باشد. از این نظر که کدها مفقود نشوند یا در صورت اشتباه در توسعه بتوان پروژه را به حالت قبل بازگرداند. در این راستا کدهای پروژه در مخزن نرم‌افزاری از نوع گیت نگهداری شده که یک مخزن در کامپیوتر شخصی و دیگری در سایت بیت‌باکت^{۱۶} قرار دارد. مزیت این سایت نسبت به گیت‌هاب این است مخازن خصوصی را به صورت رایگان ارائه می‌دهد. در مخازن خصوصی اجازه‌ی دسترسی تنها به افراد تعیین شده از طرف مالک داده می‌شود و عموم کاربران به آن دسترسی ندارند. از ابتدای شروع پیاده‌سازی کدها در مخازن بروزرسانی شده است. نمایی از ثبت‌های مختلف پروژه در مخزن در شکل ۷.۴ آورده شده است.

¹¹Performance

¹²Adaptor

¹³Factory

¹⁴Singelton

¹⁵Generic

¹⁶Bitbucket - https://bitbucket.org/alimohebbi/bug_predict

Author	Commit	Message	Date	Builds
ali mohebbi	bbbb62c	Extract mutauion matrix for clean files added and some refactorin...	5 days ago	
ali mohebbi	257acdd	change the way random files be selected	5 days ago	
ali mohebbi	2b0da83	Clean file models and tables and process metric calculation added	2018-05-15	
ali mohebbi	2bd3090	mutator for Closure and Time added	2018-05-12	
ali mohebbi	a1d9801	bug fix for lang xml build	2018-05-10	
ali mohebbi	ae25f43	Defect4j class removed and Lang mutator compelted	2018-05-10	
ali mohebbi	f67be44	All methods for extracting mutation metrics implemented and buil...	2018-05-09	
ali mohebbi	d54e900	mutation analysis for Mockito	2018-05-08	
ali mohebbi	5d19521	consider files on original path and working git bug fix	2018-05-04	
ali mohebbi	c608f24	build xml and mml and run major added. un stable	2018-04-29	
ali mohebbi	d48d54f	Process Metrics Completed	2018-04-17	
ali mohebbi	8c90c4e	Neighbor commit count added to process metrics	2018-04-17	
ali mohebbi	2662f1f	Owner experience added	2018-04-16	
ali mohebbi	56028c5	Minor author count added to process metrics	2018-04-16	
ali mohebbi	7a4ab0e	Owner participation added to process metrics	2018-04-16	

شکل ۷.۴: نمایی از مخزن نرم‌افزاری

۴.۴. رویکرد اول: معیارهای فرآیند در کنار جهش

در این قسمت چگونگی استخراج معیارهای رویکرد اول شرح داده می‌شود. ابتدا لازم است اطلاعات مربوط به ثبت‌های حاوی خطا از ابزار defects4j بازیابی شود و سپس این اطلاعات با استفاده از مخزن نرم‌افزاری تکمیل شود. در مراحل بعد ابتدا معیارهای فرآیند و سپس معیارهای جهش استخراج خواهند شد.

۱.۴.۴. استخراج اطلاعات مربوط به ثبت‌های حاوی خطا

اطلاعاتی که درباره‌ی ثبت‌های حاوی خطا قابل بازیابی است در زیر آمده است:

۱. شناسه‌ی ثبت در مخزن

۲. نام فایل حاوی خطا

۳. شماره‌ی خطا در ابزار defects4j

۴. شماره‌ی ثبت تعمیر خطا

۵. نام پروژه

۶. نام انتشار قبلی پروژه

۷. شماره‌ی ثبت انتشار قبلی پروژه

از میان اطلاعات بالا همگی به سادگی با استفاده از ابزار j4defect قابل استخراج است بجز دو مورد آخر. همچنین شماره‌ی ثبت تعمیر مورد استفاده قرار نگیرد ولی نگهداری شد چراکه ممکن بود لازم شود. برای بدست آوردن اطلاعات مربوط به هر انتشار لازم است که مخزن نرم افزاری هر پروژه مورد بررسی قرار گیرد. در مخازن پروژه‌های نرم‌افزاری از نوع گیت برای مشخص کردن یک رویداد مهم از تگ^{۱۷} استفاده می‌شود. هر تگ می‌تواند به یک ثبت از برنامه اشاره کند. تگ می‌تواند نمایانگر رویدادهایی چون انتشار برنامه، انتشار بتا، و یا کاندید انتشار باشد. بنابراین با استفاده از تگ می‌تواند انتشار را پیدا کرد.

تگ‌های مخازن گیت دو نوع سبک‌وزن^{۱۸} و حاشیه‌نویسی شده^{۱۹} که در میان پروژه‌های مورد مطالعه از هر دو نوع جهت مشخص کردن انتشار استفاده شده است. کار کردن با این دو نوع تگ دارای تفاوت‌هایی در پیاده‌سازی است که در اینجا از پرداختن به جزئیات صرف نظر می‌شود. ابتدا همه‌ی تگ‌های موجود در مخازن نرم‌افزاری استخراج می‌شود و در پایگاه داده قرار می‌گیرد. از میان تگ‌های استخراج شده تگ‌های نامرتب با انتشار از پایگاه داده حذف می‌شود. تگ‌های نامرتب با توجه به نام آنها مشخص می‌شود به عنوان مثال تگ‌هایی که حاوی لغات Beta یا Dev هستند نامرتب محسوب می‌شوند. در نهایت جدولی به نام ReleaseProject ساخته می‌شود که در آن اطلاعات انتشارهای مختلف وجود دارد. نمایی از این جدول در شکل ۸.۴ آمده است.

#	ReleaseId	CommitId	Project	SequenceNumbe	TagName
1	1	bd267505764488494ff13ba76ce53...	Lang	1	LANG_1_0
2	2	57be549cd8ffed876aafe0982f039d...	Lang	2	LANG_1_0_1
3	4	2caf1dd699d55338dae167333f676...	Lang	4	LANG_2_0
4	5	0aa8426b3f16d4fd0e6903d269669...	Lang	5	LANG_2_1
5	6	9eb821253181a7e075d7a3ed317f...	Lang	6	LANG_2_2

شکل ۸.۴: نمایی از جدول محتوای انتشارها

در قدم بعدی باید مشخص شود اولین انتشار ما قبل هر ثبت حاوی خطا کدام است. برای این منظور لیست

¹⁷Tag

¹⁸Lightweight

¹⁹Annotated

ثبت‌ها در یک پروژه به ترتیب زمانی بررسی می‌شود. اولین ثبت ماقبل ثبت مورد نظر که مربوط به یک انتشار است یافت می‌شود و به عنوان انتشار ماقبل آن ثبت در نظر گرفته می‌شود.

در نهایت جدولی به نام BugInfo تولید شده که نمایی از آن در شکل ۹.۴ آمده است. این جدول ۴۰۵ سطر دارد که بیشتر از تعداد کل خطاهای ذکر شده در مجموعه داده‌ی defects4j است. علت این است که یک خطا می‌تواند خطا در چندین پرونده به طور همزمان باشد و از آنجا که پیش‌بینی در سطح پرونده انجام می‌شود لازم است اطلاعات برای پرونده‌ها ذخیره شود.

#	ID	BUG_COMMIT_ID	BUG_NUMBER	BUGGY_CLASS_NAME	FIX_COMMIT_ID	TAG_ID	TAG_NAME	PROJECT	TAG_COMMIT
1	1	2c454a4ce3fe771098746...	1	org.apache.commons.lang3...	687b2e62b7c6e81cd9d5c8...	e1fb41239459e1f82...	LANG_3_1	Lang	b1340f422f68be7c237...
2	2	aefc12c38171e1a84a90d...	2	org.apache.commons.lang3...	09d39029b16dee61022dc...	e1fb41239459e1f82...	LANG_3_1	Lang	b1340f422f68be7c237...
3	3	1f001d06a2bde5ee4e32...	3	org.apache.commons.lang3...	2c9c8753165dc7ce5dd1d5...	e1fb41239459e1f82...	LANG_3_1	Lang	b1340f422f68be7c237...
4	4	4ddb99c5805781bd3c2...	4	org.apache.commons.lang3...	fb47b96ab635d7cc6e9edef...	e1fb41239459e1f82...	LANG_3_1	Lang	b1340f422f68be7c237...
5	5	379151bad9c5402c335d...	5	org.apache.commons.lang3...	75944e541d358d5b06ebb...	e1fb41239459e1f82...	LANG_3_1	Lang	b1340f422f68be7c237...
6	6	6823c3742ee16f5b28e5...	6	org.apache.commons.lang3...	cff0f1ae37bb2b7ab2dcd1...	e1fb41239459e1f82...	LANG_3_1	Lang	b1340f422f68be7c237...
7	7	f0c7e60bbaf975b64ab5b...	7	org.apache.commons.lang3...	e71f6dd3f2f70c640ae73d2...	e1fb41239459e1f82...	LANG_3_1	Lang	b1340f422f68be7c237...

شکل ۹.۴: نمایی از جدول محتوای اطلاعات پرونده‌های حاوی خطا

۲.۴.۴. انتخاب پرونده‌های سالم

همانطور که مطرح شد تعداد پرونده‌های حاوی خطا برابر ۴۰۵ عدد است که از تعداد کل پرونده‌ها کمتر است. بنابراین جهت ساخت مدل‌های بدون جهت‌گیری به همین تعداد پرونده‌های بدون خطا به طور تصادفی انتخاب می‌شود. بدین ترتیب یک مجموعه داده‌ی متعادل^{۲۰} حاوی ۸۱۰ پرونده ساخته شده است. این روش طبق مقاله‌ی [۳۴] به کار گرفته شده است. در این انتخاب به تعداد پرونده‌های حاوی خطا، پرونده‌های بدون خطا انتخاب می‌شود.

به ازای هر پرونده‌ی دارای خطا در همان ثبت از پروژه‌ی مربوط یک پرونده‌ی بدون خطا به صورت تصادفی انتخاب می‌شود. برای این کار لیست تمام پرونده‌های داخل پروژه در ثبت فایل حاوی خطا در نظر گرفته می‌شود و یک فایل به صورت تصادفی انتخاب می‌شود. این پرونده نباید جز پرونده‌های حاوی خطا در آن ثبت از پروژه باشد. البته ممکن اسن پرونده در ثبت‌های بعدی یا قبلی خطا داشته باشد و از این نظر محدودیتی ندارد. همانطور که گفته شد یک ثبت ممکن از بیش از یک فایل حاوی خطا داشته باشد. سپس مشخصات این فایل در جدول CleanInfo قرار می‌گیرد. نمایی از این جدول در تصویر ۱۰.۴ آورده شده است.

²⁰Balanced

#	Id	CleanClassName	CommitId	PriorTagName	Project	RelatedBugId	TagCommit
1	1	org.apache.commons.la...	2c454a4ce3fe7710987468...	LANG_3_1	Lang	1	b1340f422f68be...
2	2	org.apache.commons.la...	aefc12c38171e1a84a90dc...	LANG_3_1	Lang	2	b1340f422f68be...
3	3	org.apache.commons.la...	1f001d06a2bde5ee4e3204...	LANG_3_1	Lang	3	b1340f422f68be...
4	4	org.apache.commons.la...	4ddbd99c5805781bd3c22...	LANG_3_1	Lang	4	b1340f422f68be...

شکل ۴.۱۰: نمایشی از جدول محتوای اطلاعات پرونده‌های سالم

۳.۴.۴. استخراج معیارهای فرآیند

در این قسمت نحوه‌ی استخراج هر یک از معیارهای ذکر شده در قسمت ۱.۳ بیان می‌شود.

تعداد ثبت در سیستم کنترل نسخه: اولین راه حلی که به ذهن می‌رسد استفاده‌ی مستقیم از Jgit برای این کار است. به این صورت که تعداد ثبت‌های بین ثبت کنونی و انتشار قبلی بررسی کرده و تعداد ثبت‌هایی که در آن‌ها فایل حاوی خطا تغییر کرده است شمرده شوند. مشکل این راه این است که بسیار پر هزینه خواهد بود زیرا مرتباً باید عملیات ورودی/خروجی^{۲۱} بر روی دیسک انجام پذیرد و همچنین بررسی‌های تکراری بسیاری انجام می‌گیرد. به عنوان مثال دو ثبت حاوی خطا را در نظر بگیرید که دارای انتشار ما قبل یکسانی هستند. تعدادی از بررسی‌های ثبت‌های ما بین آن‌ها تا ثبت مربوط به انتشار دارای همپوشانی خواهد بود. از طرف دیگر می‌توان اطلاعاتی که در بررسی ثبت‌ها بدست می‌آید در محاسبه‌ی معیارهای دیگر نیز مورد استفاده قرار گیرد.

همچنین برای یافتن ثبت‌های بین انتشار و ثبت مورد نظر نمی‌توان از تاریخ ثبت آنها استفاده کرد. زیرا تعداد زیادی از ثبت‌های ابتدای برخی پروژه‌های مورد مطالعه دارای تاریخ یکسانی هستند استفاده از تاریخ غیر ممکن می‌شود. علت داشتن تاریخ یکسان احتمالاً مهاجرت از یک نوع مخزن نرم‌افزاری به نوع گیت بوده است.

بنابراین کل ثبت‌های پروژه‌ها مورد بررسی قرار گرفت و دو جدول تولید شد. جدول اول به نام CommitInfo که اطلاعات کلی ثبت‌ها را در بر می‌گیرد و جدول دوم CommitChangedFile که اطلاعات مربوط به پرونده‌هایی که در یک ثبت از برنامه نسبت به ثبت قبلی تغییر کرده است نگهداری می‌شود. در این جدول برای هر پرونده تعداد خطوط اضافه و کم شده نسبت به ثبت قبلی ذخیره شده است. در جدول اول Sequence_Number نشان می‌دهد که چندمین نسخه از ابتدای پروژه می‌باشد و این عدد در هنگام بررسی‌ها به آن ثبت داده شده زیرا برای یافتن ثبت‌های بین ثبت کنونی و ثبت مربوط به انتشار قبلی لازم است از آنها استفاده شود.

²¹Input/Output (IO)

هر سطر از جدول دوم یک کلید خارجی دارد به سطری از جدول اول. قسمتی از جدول CommitInfo در شکل ۱۱.۴ و جدول CommitChangeFile در شکل ۱۲.۴ زیر آمده است:

#	ID	COMMIT_GIT_ID	COMMITTER_MAIL	COMMITTER_NAME	PROJECT	SEQUENCE_NUMBER
35	9506	1e11bf4cfb934f6bd3788a2d47089...	szczepiq@gmail.c...	Szczepan Faber	Mockito	34
36	9507	f28e6c5ebecee8cb75f6ab79b7ad3...	szczepiq@gmail.c...	Szczepan Faber	Mockito	35
37	9508	01b9ebd8ab76460d8b2b59ec581a...	szczepiq@gmail.c...	Szczepan Faber	Mockito	36
38	9509	99e14ba7fd4ce6a101485d65a4949...	szczepiq@gmail.c...	Szczepan Faber	Mockito	37
39	9510	4d48778d08c11825c9c4f089c1730...	iczechowski@gm...	Igor Czechowski	Mockito	38
40	9511	8af6740c7ec9d65a2a7f68c7ca8ea...	iczechowski@gm...	Igor Czechowski	Mockito	39
41	9512	8e871ae69e946c89bccd8ee8f93fc...	szczepiq@gmail.c...	Szczepan Faber	Mockito	40

شکل ۱۱.۴: نمایی از جدول اطلاعات ثبت‌ها

#	ID	ADDED_LINES	DELETE	FILE_NAME	PATH	COMMIT_INFO_ID
433	433	2	3	org.apache.commons.lang.builder.T...	src/java/org/apache/com...	371
434	434	19	4	org.apache.commons.lang.builder.T...	src/java/org/apache/com...	372
435	435	10	5	org.apache.commons.lang.builder....	src/java/org/apache/com...	373
436	436	10	5	org.apache.commons.lang.builder....	src/java/org/apache/com...	373
437	437	14	11	org.apache.commons.lang.builder....	src/java/org/apache/com...	373
438	438	12	1	org.apache.commons.lang.WordWr...	src/java/org/apache/com...	376

شکل ۱۲.۴: نمایی از جدول تغییرات پرونده‌ها در ثبت‌ها

در نهایت با استفاده از قطعه کد ۱.۴ اطلاعات مربوط به ثبت مورد نظر و ثبت انتشار بازیابی می‌شوند و سپس از شماره‌ی دنباله‌ی آنها در پرسمان موجود در قطعه کد ؟؟ استفاده می‌شود و معیار محاسبه می‌گردد.

```
1 SELECT * from CommitInfo CI where CI.COMMIT_GIT_ID = :gitId AND CI.
PROJECT = :project
```

قطعه کد ۱.۴: بازیابی اطلاعات ثبت

```
1 SELECT count(*) from CommitChangedFile CC where CC.COMMIT_INFO_ID IN
2 (SELECT CI.ID from CommitInfo CI WHERE CI.SEQUENCE_NUMBER BETWEEN
3 :startSeq AND :endSeq AND CI.PROJECT = :project)
4 AND CC.FILE_NAME = :fileName
```

قطعه کد ۲.۴: محاسبه‌ی معیار تعداد ثبت در سیستم کنترل نسخه

تعداد توسعه‌دهندگان فعال: به منظور محاسبه‌ی این معیار تعداد آدرس ایمیل‌های ثبت‌کننده‌های ثبت‌هایی شمرده می‌شود که آن ثبت‌ها شماره‌ی دنباله‌ی آنها بین شماره‌ی دنباله‌ی ثبت پرونده‌ی مورد نظر و ثبت انتشار

قبلی است و همچنین در آن ثبت پرونده‌ی مورد نظر در آن ثبت‌ها تغییر کرده است. به عبارت دیگر ثبت‌هایی که نام پرونده در جدول CommitChangeFile برای آن‌ها وجود دارد.

```
1 SELECT count(DISTINCT CI.COMMITTER_MAIL) from CommitInfo CI WHERE
2 CI.SEQUENCE_NUMBER BETWEEN :startSeq AND :endSeq AND CI.PROJECT =
3 project AND CI.ID IN
4 (SELECT CC.COMMIT_INFO_ID from CommitChangedFile CC where CC.FILE_NAME
   = :fileName)
```

قطعه کد ۳.۴: محاسبه‌ی تعداد توسعه‌دهندگان فعال

تعداد توسعه‌دهندگان متمایز: برای محاسبه‌ی معیار از پرسمان قبلی استفاده می‌شود اما اینبار به جای استفاده Sequence_Number انتشار قبلی، عدد یک قرار داده می‌شود که از ابتدای پروژه توسعه دهندگان شمرده شوند.

مقدار نرمال‌سازی شده‌ی تعداد خطوط اضافه شده: از پرسمان ۴.۴ جهت محاسبه‌ی مجموع تعداد خطوط اضافه شده به پرونده در طول انتشار استفاده می‌شود و از پرسمان ۵.۴ جهت محاسبه‌ی مجموع خطوط اضافه شده به پروژه استفاده می‌شود. سرانجام حاصل پرسمان اول بر دوم تقسیم می‌شود.

```
1 SELECT sum(CC.ADDED_LINES) from CommitChangedFile CC where
2 CC.COMMIT_INFO_ID IN
3 (SELECT CI.ID from CommitInfo CI WHERE CI.SEQUENCE_NUMBER BETWEEN
4 :startSeq AND :endSeq AND CI.PROJECT = :project)
5 AND CC.FILE_NAME = :fileName
6
```

قطعه کد ۴.۴: محاسبه‌ی تعداد خطوط اضافه شده به پرونده

```
1 SELECT sum(CC.ADDED_LINES) from CommitChangedFile CC where
2 CC.COMMIT_INFO_ID IN
3 (SELECT CI.ID from CommitInfo CI WHERE CI.SEQUENCE_NUMBER
4 BETWEEN :startSeq AND :endSeq AND CI.PROJECT = :project)
```

قطعه کد ۵.۴: محاسبه‌ی تعداد خطوط اضافه شده به پروژه

مقدار نرمال‌سازی شده‌ی تعداد خطوط حذف شده: به طور مشابه معیار قبلی محاسبه می‌گردد.

درصد خطوطی که مالک فایل مشارکت کرده: دستور Blame در Jgit نشان می‌دهد که هر خط از پرونده در یک ثبت در کدام یک از ثبت‌های گذشته اضافه شده است. با یافتن ثبت مسئول اضافه کردن آن خط نویسنده‌ی آن خط مشخص می‌شود که همان ثبت‌کننده است. با کمک این دستور به دلایل مشابه ساخت جداول مربوط به ثبت‌ها، جدولی با عنوان Participation ساخته شده که در آن هر سطر نشان می‌دهد که یک نویسنده در یک نسخه از برنامه چند درصد از خطوط به وی اختصاص دارد. در شکل ۱۳.۴ نمایی از این جدول آورده شده

است. از این جدول علاوه بر محاسبه‌ی این معیار برای یافت سایر معیارها نیز استفاده خواهد شد. در نهایت معیاری که در ابتدا بسیار پیچیده به نظر می رسید به کمک پرسمان ساده‌ی ۶.۴ محاسبه خواهد شد.

#	ID	AUTHOR_EMAIL	COMMIT_ID	FILE_NAME	FILE_PATH	PARTICIPATION_PERCENT
1	1	dirkv@apache.org	2c454a4ce3fe771098...	org.apache.commons.lang3.time.FastDateFormat	src/main/java/org/ap...	0.00169492
2	2	oheger@apache.org	2c454a4ce3fe771098...	org.apache.commons.lang3.time.FastDateFormat	src/main/java/org/ap...	0.0745763
3	3	joehni@apache.org	2c454a4ce3fe771098...	org.apache.commons.lang3.time.FastDateFormat	src/main/java/org/ap...	0.0118644
4	4	stevencaswell@apa...	2c454a4ce3fe771098...	org.apache.commons.lang3.time.FastDateFormat	src/main/java/org/ap...	0.00508475

شکل ۱۳.۴: نمایشی از جدول مشارکت‌کنندگان در ویرایش پرونده‌ها

```
1 SELECT max(PARTICIPATION_PERCENT) from Participation P
2 where COMMIT_ID = :commitId AND FILE_NAME = :fileName")
```

قطعه کد ۶.۴: محاسبه‌ی درصد خطوط مالک پرونده

تعداد مشارکت‌کنندگان جزئی: با استفاده از جدول Participation و پرسمان ۷.۴ معیار محاسبه می‌شود. مقدار minorThreshold برابر ۵ درصد قرار می‌گیرد.

```
1 SELECT count(AUTHOR_EMAIL) from Participation P
2 where COMMIT_ID = :commitId AND FILE_NAME = :fileName
3 and PARTICIPATION_PERCENT < :minorThreshold
```

قطعه کد ۷.۴: محاسبه‌ی تعداد مشارکت‌کنندگان جزئی

تعداد ثبت‌های همسایگان: ابتدا لازم است که همسایگان پرونده در یک ثبت و نیز تعداد دفعات همسایگی در طول انتشار مشخص شود. این عمل به وسیله‌ی پرسمان ۸.۴ انجام می‌شود. سپس معیار تعداد ثبت‌ها در سیستم کنترل نسخه مشابه قبل با استفاده از کد ۲.۴ محاسبه می‌گردد و از آنها میانگین وزن‌دهی شده گرفته می‌شود.

```
1 SELECT FILE_NAME as 'name', count(ID) as 'frequency' FROM
2 CommitChangedFile WHERE COMMIT_INFO_ID IN
3 (SELECT COMMIT_INFO_ID FROM CommitChangedFile WHERE FILE_NAME = :
4 fileName)
5 AND COMMIT_INFO_ID IN
6 (SELECT CI.ID from CommitInfo CI WHERE CI.SEQUENCE_NUMBER BETWEEN :
7 startSeq AND :endSeq AND PROJECT = :project)
8 AND FILE_NAME != :fileName GROUP BY FILE_NAME
```

قطعه کد ۸.۴: یافتن همسایگان و تعدد همسایگی

تعداد توسعه‌دهندگان فعال همسایگان: به طور مشابه با معیار قبلی محاسبه می‌شود.
تعداد توسعه‌دهندگان متمایز همسایگان: به طور مشابه با معیار قبلی محاسبه می‌شود.

تجربه‌ی مالک فایل: برای محاسبه‌ی معیار ابتدا با استفاده از پرسمان ۹.۴ مالک پرونده مشخص می‌شود.

سپس تعداد ثبت‌هایی که مالک پرونده از ابتدای پروژه تا آن زمان ثبت کرده است با استفاده از پرسمان ۱۰.۴

شمرده می‌شود. به ترتیب از دو جدول Participation و CommitInfo استفاده می‌شود.

```
1 SELECT AUTHOR_EMAIL FROM Participation P WHERE COMMIT_ID = :commitId
2 AND FILE_NAME = :fileName AND PARTICIPATION_PERCENT =
3 (SELECT max(PARTICIPATION_PERCENT) FROM Participation P2
4 WHERE P2.COMMIT_ID = :commitId AND P2.FILE_NAME = :fileName)
```

قطعه کد ۹.۴: یافتن مالک پرونده

```
1 SELECT count(*) from CommitInfo CI where CI.SEQUENCE_NUMBER BETWEEN
2 :startSeq AND :endSeq AND CI.PROJECT = :project AND CI.COMMITTER_MAIL =
3 :authorEmail
```

قطعه کد ۱۰.۴: شمارش تعداد ثبت‌های یک ثبت کننده در بازه‌ی زمانی داده شده

تجربه‌ی تمام مشارکت کنندگان: ابتدا همه‌ی توسعه‌دهندگان پرونده با استفاده از پرسمان ۱۱.۴ مشخص

می‌شوند. سپس میزان تجربه‌ی هر یک با استفاده از پرسمان ۱۰.۴ جداگانه محاسبه می‌شود و از آن‌ها میانگین

هندسی گرفته می‌شود.

```
1 SELECT AUTHOR_EMAIL FROM Participation P WHERE COMMIT_ID = :commitId
2 AND FILE_NAME = :fileName
```

قطعه کد ۱۱.۴: یافتن مشارکت کنندگان در پرونده

در نهایت جدولی برای معیارهای فرآیند تولید می‌شود که نمایی از آن در شکل ۱۴.۴ آورده شده است.

#	ID	ACTIVE_D	COMM	DEV_COUNT	FILE_INFO_ID	NORMAL_ADC	NORMAL_DE	FILE_I	MINOR	OWNER_PARTI	ALL_AUTH	OWNER_EXPERIEN	NEIGHBORS_COMM	NEIGHBORS_A	NEIGHBORS_TOTAL_DEV
98	98	0	0	1	98	0	0	B	0	1	536	536	0	0	0
99	99	1	3	2	99	0.0716029	0.060241	B	0	0.595238	158	525	4	1	2
100	100	1	1	2	100	0.000898...	0.0019084	B	1	0.990909	148	524	1.5	1.5	2
101	101	3	8	4	101	0.000521...	0.001285...	B	2	0.892635	717	1324	4.39672	1.93115	3.94754
102	102	6	22	11	102	0.00794188	0.0134352	B	6	0.497545	213	450	4.08507	1.85764	4.02778
103	103	4	32	4	103	0.0671348	0.0951634	B	2	0.996774	570	441	5.73	2.25	3.16

شکل ۱۴.۴: نمایی از جدول معیارهای فرآیند

۴.۴.۴. استخراج معیارهای جهش

روند کلی به این صورت است که برای هر سطر از جدول BugInfo یا CleanInfo که معادل یک پرونده در

یک نسخه است ابتدا آن نسخه از برنامه در پوشه‌ی کاری قرار می‌گیرد. منظور از پوشه‌ی کاری محلی است که

پرونده‌های پروژه از مخزن نرم‌افزاری فراخوانی می‌شود و در آن قرار می‌گیرد. سپس به فایل build.xml و یا

build.gradle قطعه کدهایی به منظور اجرای صحیح فرآیند ساخت اضافه می‌شود.

همچنین جهت تولید جهش یافته و تحلیل جهش لازم است برای هر پروژه پیکربندی‌هایی انجام شود که این پیکربندی‌ها با اجرای عملیات مهندسی معکوس در ابزار Defects4j به دست آمد. به منظور انجام مهندسی معکوس کدهای ابزار که به زبان پریل^{۲۲} نوشته شده‌اند مورد بررسی قرار گرفتند و نحوه‌ی عملکرد ابزار با پروژه‌های مختلف و پیکربندی‌ها مشخص شد.

از آنجا که اجرای تحلیل جهش زمان زیادی می‌گیرد برای انجام آن یک رایانه به صورت اختصاصی برای انجام آن در آزمایشگاه کیفیت نرم‌افزار^{۲۳} واقع در دانشگاه صنعتی شریف در نظر گرفته شد. این رایانه به یک سرور لینوکس^{۲۴} تبدیل شد تا امکان نظارت و رفع خطا در استخراج معیارهای جهش همواره امکان پذیر باشد و استخراج معیارها و توسعه‌ی سایر قسمت‌های این پژوهش به صورت موازی انجام گیرد. جزییات تبدیل رایانه به سرور لینوکس در پیوست آمده است.

از آنجا که انجام تحلیل جهش بر روی موارد مطالعاتی صنعتی انجام گرفته است و پروژه‌های انتخاب شده حجم زیادی دارند لازم است تا پیکربندی‌هایی در نظر گرفته شود تا از بروز خطا و توقف محاسبات جلوگیری شود. این پیکربندی‌ها در زیر آمده است.

- **افزایش فضای PermGen^{۲۵}:** این فضا یک هیپ^{۲۶} مخصوص است که از فضای هیپ اصلی جاوا مجزا است و در آن ماشین مجازی جاوا^{۲۷} فراداده‌های^{۲۸} کلاس‌های بارگذاری شده را ردگیری می‌کند. به دلیل حجم زیاد پروژه‌های مورد مطالعه لازم است که این فضا بیشتر از حالت پیش فرض قرار داده شود. برای انجام این پژوهش فضای ۲ گیگابایت در نظر گرفته شده است.
- **افزایش فضای Codecache:** کدهای ترجمه شده به زبان ماشین در این فضا قرار می‌گیرد که به دلیل مشابه پیکربندی قبلی لازم است این فضا از حالت پیش فرض بیشتر باشد. فضای در نظر گرفته شده ۵۱۲ مگابایت می‌باشد.

- **قرار دادن زمان خروج^{۲۹}:** زمانی که یک جهش یافته از کد اصلی ساخته می‌شود ممکن است که جریان کنترلی به نحوی تغییر کند که برنامه در حلقه‌ی بی‌نهایت یا بن‌بست قرار گیرد. برای جلوگیری از چنین حالتی لازم است تا در تنظیمات ابزار JUnit مهلت زمانی در نظر گرفته شود تا در صورت قرارگیری در چنین شرایطی پس از مدت زمان معین اجرای مورد آزمون متوقف شود و مورد آزمون شکست خورده

^{۲۲}Perl

^{۲۳}Software Quality Research Lab - <http://sqrllab.ce.sharif.edu/>

^{۲۴}Linux

^{۲۵}Premanent Generation

^{۲۶}Heap

^{۲۷}Java Virtual Machine

^{۲۸}Metadata

^{۲۹}Timeout

تلقی شود. مدت زمان تعیین شده جهت خروج ۱۳ ثانیه می‌باشد. عملگرهای جهش انتخابی با توجه به هزینه‌ی زمانی تحلیل جهش به کارگیری تمامی عملگرهای موجود در ابزار Major به صرفه نمی‌باشد. برای تولید جهش‌یافته‌ها از مجموعه عملگرهای استفاده شده در مقاله‌ی بوئر و همکاران [۲۹] استفاده شده که مطابق عملگرهای پیش‌فرض در ابزار PIT می‌باشد. پرونده‌ی MML ساخته شده در شکل ۱۵.۴

```
1 targetOp{
2
3 BIN(+) ->{-};
4 BIN(-) ->{+};
5 BIN(*) ->{/};
6 BIN(/) ->{*};
7 BIN(%) ->{*};
8
9 BIN(>>) ->{<<};
10 BIN(<<) ->{>>};
11 BIN(>>>) ->{<<<};
12
13 BIN(&) ->{|};
14 BIN(|) ->{&};
15 BIN(^) ->{&};
16
17 UNR(+) ->{-};
18 UNR(-) ->{+};
19
20 // Use sufficient replacements for ROR
21 BIN(>) ->{>=,<=};
22 BIN(<) ->{<=,>=};
23 BIN(>=) ->{>,<};
24 BIN(<=) ->{<,>};
25 BIN(==) ->{!=};
26 BIN(!=) ->{==};
27
28 // Delete all types of supported statements
29 DEL(CALL);
30
31 // Enable all operators
32 AOR;
33 EVR;
34 LOR;
35 SOR;
36 ROR;
37 ORU;
38 STD;
39 }
```

شکل ۱۵.۴: پرونده‌ی mml ساخته شده جهت تولید جهش‌یافته‌ها

پس از انجام تحلیل جهش برای پرونده‌های حاوی خطا و سالم نتایج در جدول MutationMetrics قرار داده شد که نمایی از این جدول در شکل ۱۶.۴ آمده است.

Result Grid							
Filter Rows: <input type="text"/>							
#	MetricId	Covered	FileInfold	FileType	GeneratedCount	Killed	Lived
1	1	369	1	B	373	310	59
2	2	82	2	B	82	80	2
3	3	364	3	B	368	304	60
4	4	26	4	B	26	20	6

شکل ۱۶.۴: نمایشی از جدول نتایج تحلیل جهش

۵.۴. رویکرد دوم: معیارهای فرآیند مبتنی بر جهش

همانطور که در قسمت ۲.۳ اشاره شده چهار معیار معرفی شدند و مبتنی بر جهش نامیده شدند. این قسمت به نحوه پیاده‌سازی دسته‌ی دوم از معیارها را شرح خواهد داد.

- **تعداد جهش‌یافته‌های تولید شده‌ی جدید نسبت به انتشار قبلی برنامه:** به منظور محاسبه‌ی این معیار ابتدا لازم است که مشخص شود که پرونده‌ی مورد نظر نسبت به انتشار قبلی چه تغییراتی داشته است. این کار با استفاده از ابزار JGit انجام می‌شود. JGit این امکان را فراهم می‌کند که دو پرونده در دو ثبت متفاوت مقایسه شوند و مشخص می‌کند که کدام خطوط حذف شده‌اند و کدام خطوط اضافه شده‌اند. در اینجا لازم است خطوط اضافه شده مشخص شود. سپس با استفاده از ابزار Major جهش‌یافته‌ها تولید می‌شود. در قسمت ۲.۲.۴ توضیح داده شد که پس تولید جهش‌یافته‌ها یک فایل خروجی نیز به نام mutant.log تولید می‌شود که در آن مشخص شده در هر خط از برنامه چه جهش‌یافته‌هایی تولید شده است. حال کافیست تعداد جهش‌یافته‌های تولید شده در خطوطی شمرده شوند که ابزار JGit آن‌ها را به عنوان خطوط جدید نسبت به انتشار قبلی معرفی کرده است. بدین ترتیب این معیار محاسبه خواهد شد. لازم به ذکر است روش یاد شده پایه‌ی محاسبه‌ی معیار بعدی و معیارهای رویکرد سوم است.

• تعداد جهش‌یافته‌های متمایز در چند انتشار اخیر:

به منظور افزایش کارایی ابتدا بررسی می‌شود که فایل مورد نظر در آن انتشار وجود دارد یا خیر در صورت عدم وجود محاسبات برای آن انتشار انجام نمی‌گیرد. برای محاسبه ابتدا چهار انتشار قبلی با استفاده از پرسمان ۱۲.۴ از جدول ProjectRelease بازیابی می‌شود. سپس مشابه معیار قبلی جهش‌یافته‌های جدید نسبت به انتشار قبلی برای هر انتشار محاسبه می‌شود و با هم جمع زده می‌شود. یک جدول برای نتایج تولید جهش‌یافته‌ها به نام DistinctMutantLog در نظر گرفته شده که تعداد جهش‌یافته‌های جدید برای هر انتشار نسبت به انتشار قبلی در آن ذخیره می‌گردد. از مزایای ایجاد این جدول پایداری در

انجام محاسبات است به عنوان مثال در صورت توقف محاسبات امکان از سرگیری محاسبات از محل توقف وجود داشته دارد و همچنین نگهداری به عنوان یک مجموعه داده می‌تواند در پژوهش‌های دیگر بکار گرفته شود. نمایی از جدول در شکل زیر آمده است. به طور مثال سطر اول جدول بیان می‌کند که در انتشاری از برنامه با شماره ثبت 21a.. پرونده‌ی شماره یک شماره یک از فایل‌های حاوی خطا ۴۳۰ جهش‌یافته‌ی جدید نسبت به انتشار قبلی داشته است.

```
1 SELECT * FROM ProjectRelease WHERE Project = :project AND
2 SequenceNumber <
3 (SELECT SequenceNumber FROM ProjectRelease WHERE
4 Project = :project AND CommitId = :releaseCommit)
5 ORDER BY SequenceNumber DESC LIMIT 4
```

قطعه کد ۱۲.۴: بازیابی چهار نسخه‌ی اخیر یک ثبت

#	LogId	CommitId	FileId	FileType	NewMutants
1	1	aad55e0d568d152e7290a18136d247b1abbaa21a	1	B	430
2	2	9ee116a6a54763f0e86567df2a290cf81d8a3437	1	B	44
3	3	b1340f422f68be7c237bbc9127d1f12a92be16a2	1	B	4
4	4	aad55e0d568d152e7290a18136d247b1abbaa21a	2	B	98
5	5	9ee116a6a54763f0e86567df2a290cf81d8a3437	2	B	0
6	6	b1340f422f68be7c237bbc9127d1f12a92be16a2	2	B	0

شکل ۱۷.۴: نمایی از جدول تعداد جهش‌یافته‌های متمایز در انتشارها

- **میزان تغییرات مثبت امتیاز جهش در چند انتشار اخیر:** ابتدا انتشارها مشابه معیار قبلی بازیابی می‌شوند و سپس برای هر یک تحلیل جهش انجام می‌گردد. نتایج جهش در جدولی به نام ReleaseMutation قرار می‌گیرد. نمایی از این جدول در شکل ۱۸.۴ آمده است. سپس هر انتشار با انتشار قبلی مقایسه می‌شود و در صورتی که تغییر امتیاز جهش مثبت باشد با مجموعه تغییرات مثبت جمع می‌گردد.

#	Id	Covered	FileInfold	FileType	GeneratedCount	Killed	Lived	ReleaseId
1	1	426	1	B	430	324	102	17
2	2	427	1	B	431	325	102	18
3	3	429	1	B	433	325	104	19
4	4	98	2	B	98	84	14	17
5	5	98	2	B	98	84	14	18

شکل ۱۸.۴: نمایشی از جدول نتایج تحلیل جهش در انتشارها

- میزان تغییرات منفی امتیاز جهش در چند انتشار اخیر: به طور مشابه با معیار قبلی عمل می‌گردد با این تفاوت که تغییرات منفی در نظر گرفته می‌شود.

۶.۴. رویکرد سوم: معیارهای ترکیبی جهش-فرآیند

نحوه‌ی محاسبه به این صورت خواهد بود که ابتدا ثبت‌هایی از برنامه در طول آخرین انتشار که در آن فایل مورد نظر تغییر کرده است توسط پرسمان ۱۳.۴ بازیابی می‌شود. سپس برای هر ثبت تعداد جهش یافته‌های جدید نسبت به ثبت قبلی محاسبه می‌شود و برای محاسبه‌ی جهش یافته‌های حذف شده تعداد جهش یافته‌ها در ثبت قبلی را یافته و آن‌ها که جز خطوط حذف شده در ثبت بعدی است شمرده می‌شود. تعداد جهش یافته‌های اضافه و حذف شده در ثبت‌ها جمع شده و بر تعداد ثبت‌های کل پروژه در طول انتشار تقسیم می‌گردد.

```

1 SELECT CC.* from CommitChangedFile CC, CommitInfo CI where CC.
   COMMIT_INFO_ID = CI.ID
2 AND CI.SEQUENCE_NUMBER BETWEEN :startSeq AND :endSeq
3 AND CI.PROJECT = :project
4 AND CC.FILE_NAME = :fileName ORDER BY CI.SEQUENCE_NUMBER asc

```

قطعه کد ۱۳.۴: بازیابی اطلاعات ثبت‌هایی که یک فایل در بازه‌ی مشخص در آنها تغییر کرده است

فصل ۵

ارزیابی

در این بخش به تشریح نحوه‌ی ساخت مدل‌های پیش‌بینی و ارزیابی معیارهای شرح داده شده در فصل ۳ پرداخته می‌شود. با استفاده از معیارهای استخراج شده در فصل ۴ مدل‌های مورد نظر ساخته می‌شوند. ساخت مدل‌ها در زبان R انجام می‌گردد به وسیله‌ی بسته‌ی کرت^۱ [۳۵] انجام می‌شود.

در ساخت و ارزیابی مدل‌ها از روش ارزیابی میان دسته‌ای استفاده می‌شود که تعداد دسته‌ها ۱۰ و تعداد تکرار نیز ۱۰ مورد می‌باشد. لازم به ذکر است که دسته‌بندی‌ها به طور تصادفی انجام می‌شود. همچنین در بسته‌ی کرت در هر روش دسته‌بندی پارامترهای مختلفی به طور پیش فرض به کار گرفته می‌شود تا بهترین مدل ممکن ساخته شود. در ابتدا ۱۰ درصد از داده‌ها به عنوان داده‌ی آزمون جدا می‌شود. با استفاده از ۹۰ درصد باقی‌مانده به ساخت مدل پرداخته می‌شود. با استفاده از ارزیابی میان‌دسته‌ای و تنظیم خودکار پارامترهای مختلف مدل نهایی ساخته شده و از این مدل برای پیش‌بینی داده‌های آزمون مورد استفاده قرار گرفته است. در ادامه هر یک از رویکردها به طور جداگانه ارزیابی شده و نتایج در زیر آمده است.

۱۰.۵. ارزیابی معیارهای فرآیند و جهش

همانطور که اشاره شد هدف از این آزمایش این است که مشخص شود قرارگیری معیارهای جهش در کنار معیارهای فرآیند باعث بهبود پیش‌بینی خطا می‌گردد یا خیر و این تاثیر تا چه میزان است. به همین منظور یک با استفاده از ۱۲ معیار فرآیند یک مدل پیش‌بینی ساخته شده و مدل دیگری با استفاده از ۱۲ معیار فرآیند و ۴ معیار جهش ساخته شده است. در نهایت این دو مدل با استفاده از معیارهای ارزیابی مختلف با هم مقایسه شدند. بدیهی است که دو مدلی که با هم مقایسه می‌شوند بجز در معیارهای استفاده شده (بردار ویژگی) به منظور ساخت مدل از هیچ منظری تفاوت ندارند و داده‌های یکسانی در ساخت و ارزیابی آنها استفاده شده. در این ارزیابی از چهار روش دسته‌بندی استفاده شده است. این روش‌های دسته‌بندی بیش از سایرین در مقالات

¹Caret

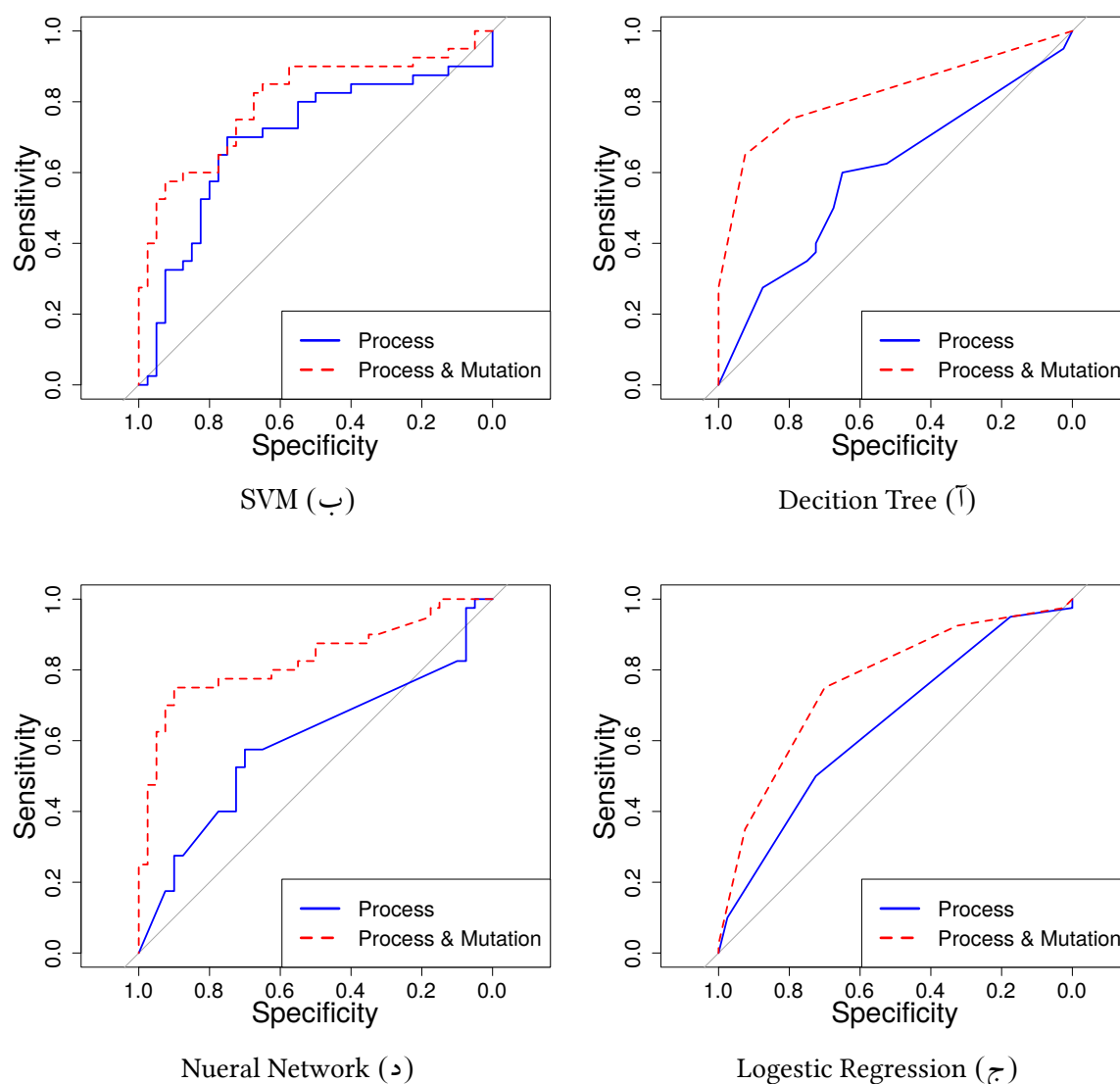
مورد استفاده قرار گرفته‌اند.

در جدول ۱.۵ بخشی از نتایج آمده است. این نتایج نشان می‌دهد که قرار گیری معیارهای جهش در کنار معیارهای فرآیند موجب بهبود پیش‌بینی خطا به مقدار قابل ملاحظه‌ای می‌شود و در تمام روشهای یادگیری موجب بهبود پیش‌بینی می‌گردد. از میان روشهای دسته‌بندی بهترین عملکرد پس از افزودن معیارهای جهش از نظر صحت و دقت را روش Neural Network داشته است. روش Decition Tree نیز بهترین عملکرد از نظر معیار بازخوانی را داشته است. همچنین بیشترین تغییر مثبت در صحت پیش‌بینی پس از افزودن معیارهای جهش را روش Neural Network و Decition Tree با مقدار ۲۰ درصد داشته است. کمترین تاثیر با مقدار ۱۵/۱ درصد در روش SVM بوده است. بیشترین افزایش دقت در روش Decition Tree بوده است که مقدار آن ۱۵/۱ درصد می‌باشد. از نظر معیار بازخوانی بیشترین تغییر مثبت را درخت تصمیم دارد که رشد ۲۵ درصدی داشته و روش Logistic Regression کاهش ۲/۵ درصدی داشته است. به طور کلی می‌توان این نتیجه حاصل شود که بیشترین بهبود در روش Decision Tree و کمترین در SVM روی داده است.

جدول ۱.۵: مقایسه‌ی معیارهای فرآیند به تنهایی و به همراه جهش

معیار	نام روش	صحت	دقت	بازخوانی
فرآیند	Decition Tree	۰/۵۸۷	۰/۵۷۴	۰/۶۷۵
فرآیند و جهش	Decition Tree	۰/۷۸۷	۰/۷۲۵	۰/۹۲۵
فرآیند	SVM	۰/۶۶۲	۰/۶۸۵	۰/۶۰۰
فرآیند و جهش	SVM	۰/۷۳۷	۰/۸۰۶	۰/۶۲۵
فرآیند	Logestic Regression	۰/۶۱۲	۰/۵۹۱	۰/۷۲۵
فرآیند و جهش	Logestic Regression	۰/۷۲۵	۰/۷۳۶	۰/۷۰۰
فرآیند	Nueral Network	۰/۶۱۲	۰/۷۲۵	۰/۵۹۱
فرآیند و جهش	Nueral Network	۰/۸۱۲	۰/۷۷۷	۰/۸۷۵

در شکل ۱.۵ نمودارهای ROC به تفکیک روش دسته‌بندی آمده است. در هر یک از زیر شکل‌ها منحنی ROC مربوط به دو مدل با هم مقایسه شده است. در مدل اول که در ساخت آن از معیارهای فرآیند استفاده شده با خط ممند نمایش داده شده است و مدل دوم از معیارهای فرآیند به همراه معیارهای جهش ساخته شده است و با خط چین نمایش داده شده. همانطور که قابل مشاهده است در تمامی روش‌ها دسته‌بندی مدل حاوی معیار جهش مساحت زیر نمودار بیشتری نسبت به مدل دیگر دارند و نشان از عملکرد بهتر این مدل‌ها می‌باشد.



شکل ۱.۵: نمودارهای ROC معیارهای فرآیند و به همراه جهش

در جدول ۲.۵ مساحت زیر نمودار ROC در هر یک از روش‌های دسته‌بندی آورده شده است. در میان روش‌های یادگیری به کار گرفته شده بیشترین افزایش مساحت زیر نمودار را Neural Network به مقدار ۰/۲۲۶ واحد داشته است و کمترین تغییر را نیز Logistic Regression با مقدار ۰/۰۳۸ واحد داشته است. به طور متوسط ۰/۱۵۱ واحد در مدلها بهبود مشاهده می‌شود. این موضوع نشان از تاثیر قابل توجه معیارهای جهش می‌باشد.

جدول ۲.۵: مقادیر زیر نمودار ROC معیارهای فرآیند و به همراه جهش

Neural Network	Logestic Regression	SVM	Decition Tree	معیار
۵۹۳	۶۴۳	۶۹۷	۵۹۶	فرآیند
۸۲۹	۷۶۱	۸۰۲	۸۲۲	فرآیند و جهش

۲.۵. ارزیابی معیارهای فرآیند مبتنی بر جهش

ارزیابی این معیارها در دو مرحله انجام می‌شود. در مرحله‌ی اول سه مدل ساخته می‌شود. این مدل‌ها به ترتیب با استفاده از معیارهای فرآیند، فرآیند و جهش و مدل آخر با استفاده از معیارهای فرآیند و فرآیند مبتنی بر جهش ساخته می‌شود. در مرحله‌ی دوم دو مدل ساخته می‌شود. در مدل اول معیارهای فرآیند و جهش مدل پیش‌بینی را خواهد ساخت و در مدل دوم معیارهای فرآیند مبتنی بر جهش نیز به مجموعه‌ی معیارها افزوده می‌شود.

۱.۲.۵. مرحله‌ی اول

مقایسه‌ی این مدل‌ها امکان را فراهم می‌کند مشخص شود آیا معیارهای فرآیند مبتنی بر جهش دارای قابلیت پیش‌بینی هستند یا خیر. همچنین در صورت داشتن این قابلیت مشخص شود که این قابلیت از معیارهای جهش کمتر است یا بیشتر.

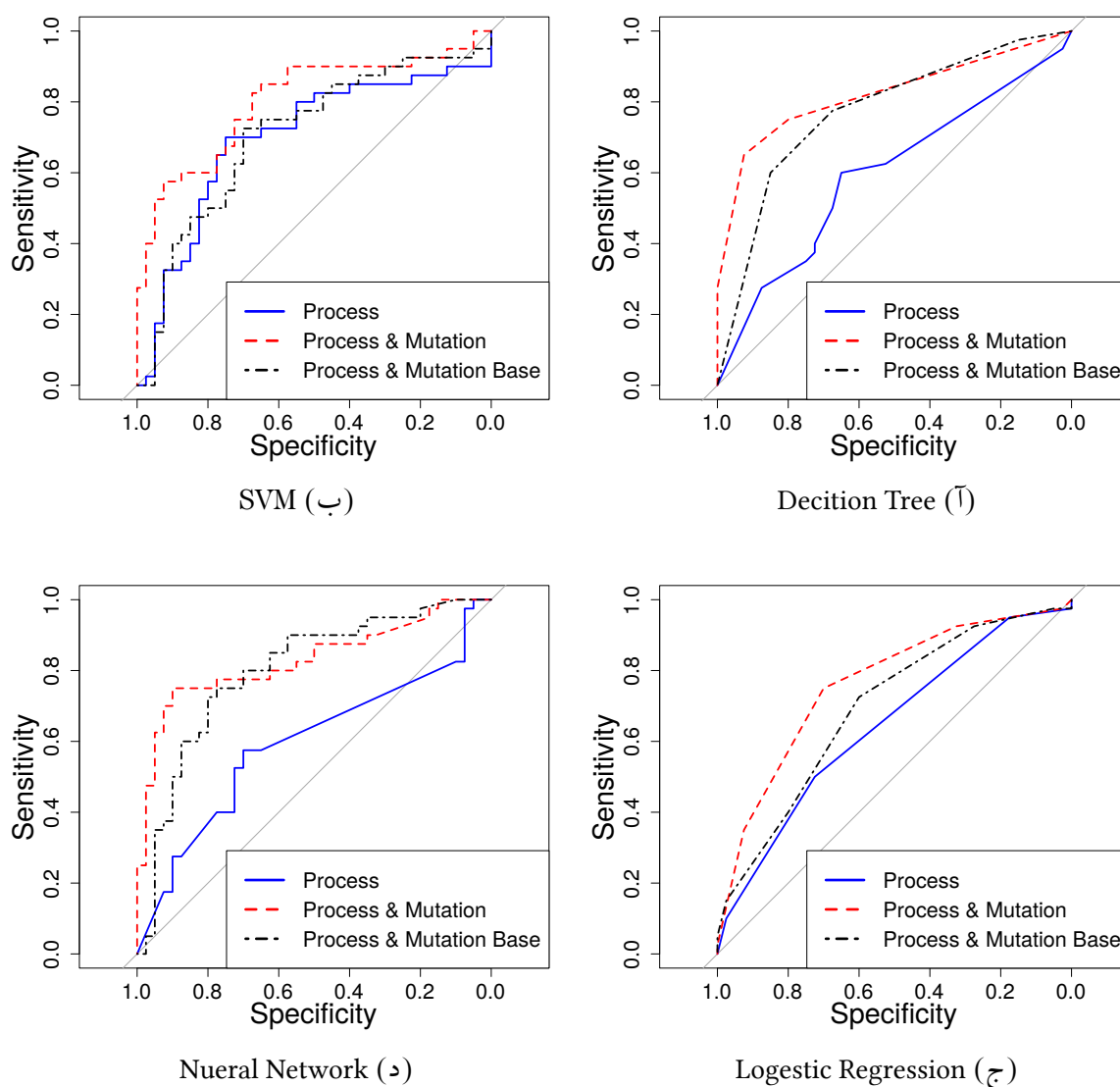
مقایسه‌ی نتایج بدست آمده در جدول ۳.۵ با جدول ۱.۵ نشان می‌دهد که در تمامی روشهای دسته‌بندی بجز SVM معیار صحت در مدل سوم از مدل اول مقدار بیشتری دارد. در مدل ساخته شده توسط SVM نیز اختلاف معیار صحت کم می‌باشد (۳ درصد). این مدل در مقایسه با مدل دوم عملکرد بهتری از نظر معیار صحت و بازخوانی در هیچکدام از روشهای دسته‌بندی نداشته است. از نظر معیار دقت در تمامی روشها مدل سوم از مدل اول عملکرد بهتری داشته و حتی در روش Decition Tree مدل سوم از مدل دوم نیز بهتر عملکرده است. از نظر معیار بازخوانی مدل سوم نسبت به مدل اول تنها در روش Neural Network عملکرد بهتری داشته، در Decision Tree بدون تغییر مانده و در دو روش دیگر کاهش یافته است. می‌توان این نتیجه را برداشت کرد که معیارهای ارائه شده دارای توانایی پیش‌بینی بیشتری نسبت به معیارهای فرآیند به تنهایی هستند.

جدول ۳.۵: نتایج پیش‌بینی خطای معیارهای فرآیند مبتنی بر جهش - مرحله‌ی اول

نام روش	صحت	دقت	بازخوانی
Decition Tree	۰/۷۲۵	۰/۷۵۰	۰/۶۷۵
SVM	۰/۶۳۷	۰/۶۸۹	۰/۵۰۰
Logestic Regression	۰/۶۶۲	۰/۶۸۵	۰/۶۰۰
Neural Network	۰/۷۶۲	۰/۷۵۶	۰/۷۷۵

در شکل ۲.۵ نمودارهای ROC سه مدل ساخته شده نشان داده شده است. در زیرشکل‌های (آ) (ج) (د) به وضوح عملکرد بهتر مدل سوم از مدل اول قابل مشاهده است. در زیرشکل (ب) نیز که متعلق به SVM است با رجوع به جدول ۴.۵ مشخص می‌شود که در این شکل نیز مساحت زیر نمودار ROC در مدل سوم بیشتر از اول است. همچنین مساحت زیر نمودار در مدل سوم در زیرشکل (ج) به مقدار ۱۵٪ واحد از مدل دوم نیز بیشتر است.

این نتایج در راستای نتایج بدست آمده از جدول ۳.۵ می‌باشد. در نهایت می‌توان این نتیجه را گرفت که معیارهای مبتنی بر جهش معرفی شده دارای توانایی پیش‌بینی خطای بیشتری نسبت به معیارهای فرآیند هستند اما این توانایی بیشتر از معیارهای جهش نیست. همچنین از آنجا که هزینه‌ی محاسباتی بیشتری نسبت به معیارهای جهش دارند جایگزینی آنها به جای یکدیگر مزیتی ندارد.



شکل ۲.۵: نمودارهای ROC معیارهای فرآیند، فرآیند و جهش، فرآیند مبتنی بر جهش

جدول ۴.۵: مقادیر زیر نمودار ROC معیارهای فرآیند مبتنی جهش

Neural Network	Logestic Regression	SVM	Decition Tree
۷۹۸	۶۹۳	۷۰۷	۷۷۲

۲.۲.۵. مرحله‌ی دوم

همانطور که اشاره شد دو مدل ساخته می‌شود که مدل اول از معیارهای فرآیند و جهش استفاده می‌کند و مدل دوم همگی معیارها (با افزودن معیارهای فرآیند مبتنی بر جهش) در ساخت مدل استفاده می‌شود. هدف از این

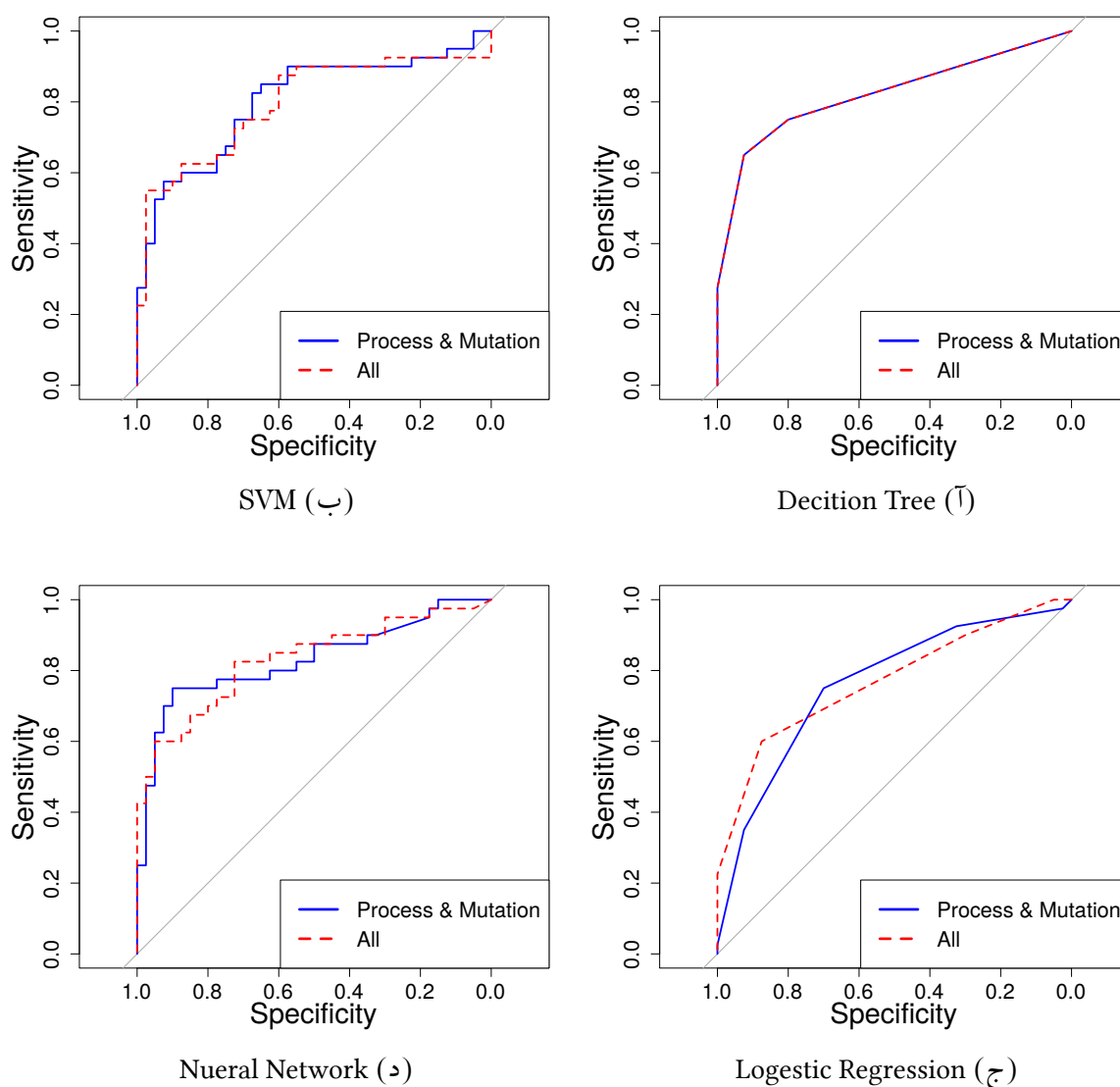
آزمایش این است که مشخص شود در صورتی که معیارهای ارائه شده‌ی جدید در کنار معیارهای قبلی قرار گیرد، در پیش‌بینی بهبودی حاصل می‌گردد یا خیر.

نتایج بدست آمد در جدول ۵.۵ نشان می‌دهد که مدل دوم در هیچ یک از روش‌ها بجز Logistic Regression از نظر معیارهای صحت، دقت و بازخوانی نسبت به مدل اول بهبودی پیدا نکرده است. همچنین در روش Decition Tree نتایج دو مدل یکسان است. در روش Logistic Regression مدل دوم در معیار صحت ۱/۲ درصد افزایش، در معیار دقت ۵ درصد کاهش و ۱۷/۵ درصد، بازخوانی افزایش داشته است.

جدول ۵.۵: نتایج پیش‌بینی خطای مدل حاصل از بکارگیری تمامی معیارها

نام روش	صحت	دقت	بازخوانی
Decition Tree	۰/۷۸۷	۰/۷۲۵	۰/۹۲۵
SVM	۰/۷۱۲	۰/۷۷۴	۰/۶۰۰
Logestic Regression	۰/۷۳۷	۰/۶۸۶	۰/۸۷۵
Nueral Network	۰/۷۵۰	۰/۷۱۷	۰/۸۲۵

نمودارهای ROC هر یک از این دو مدل در روش‌های دسته‌بندی مختلف در شکل ۳.۵ آمده است. در روش‌های مختلف مدل اول با دوم تفاوت چندانی ندارند و طبق جدول ۶.۵ تنها در مدل‌های حاصل از روش Logistic Regression به مقدار ۰/۰۰۹ واحد مساحت زیر نمودار افزایش پیدا کرده است. بنابراین قرار گیری معیارهای فرآیند مبتنی بر جهش نمی‌تواند به بهبود پیش‌بینی بیانجامد.



شکل ۳.۵: نمودارهای ROC معیارهای جهش و فرآیند و تمامی معیارها

جدول ۳.۶: مقادیر زیر نمودار ROC تمامی معیارها

Neural Network	Logestic Regression	SVM	Decition Tree
۸۳۰	۷۷۰	۷۸۶	۸۲۲

۳.۵. ارزیابی معیارهای ترکیبی فرآیند-جهش

در این قسمت به ارزیابی دو معیار مطرح شده پرداخته می‌شود. به منظور ارزیابی آنها دو مدل با استفاده از هر یک از روشهای انتخابی استفاده می‌شود. در مدل اول معیارهای فرآیند استفاده می‌شود و در مدل دوم معیار

مقدار نرمال شده‌ی خطوط اضافه شده با معیار تعداد خطوط اضافی وزن‌دهی شده جایگزین می‌شود و معیار مقدار نرمال شده‌ی خطوط حذف شده به طور مشابه جایگزین می‌شود. سایر معیارهای مدل دوم با مدل اول یکسان خواهد بود.

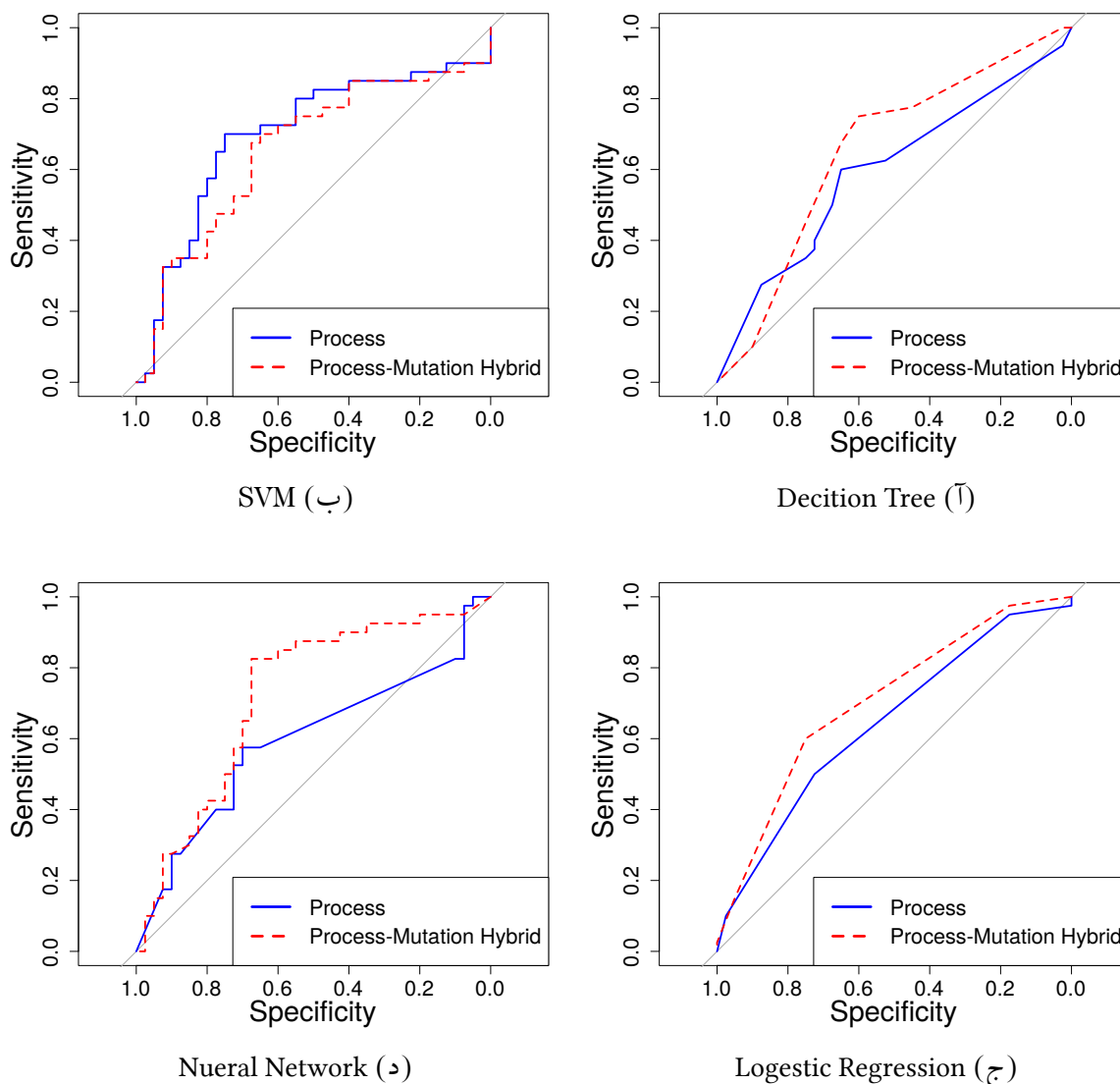
نتایج به دست آمده در جدول ۷.۵ نشان می‌دهد که معیارهای صحت، دقت و بازخوانی برای تمامی مدل‌ها بجز مدل ساخته شده توسط روش SVM افزایش قابل ملاحظه‌ای داشته است. بیشترین افزایش صحت در روش Neural Network به میزان ۱۳/۸ درصد روی داده است. از نظر افزایش دقت بیشترین تغییر مثبت در روش Decition Tree بوده است که ۱۳/۱ درصد رشد داشته است. معیار بازخوانی در دو روش Logistic Regression و Neural Network به ترتیب ۸/۴ و ۲/۵ رشد داشته و در دو روش دیگر کاهش داشته است. به طور میانگین معیار صحت ۶/۶ درصد افزایش، معیار دقت ۶ درصد افزایش و معیار بازخوانی ۰/۴ درصد کاهش داشته است. در نهایت می‌توان این نتیجه را گرفت که معیارهای ترکیبی جهش-فرآیند موجب بهبود در صحت و دقت پیش‌بینی می‌شوند و تاثیر چندانی در بازخوانی ندارند. لازم به ذکر است که تنها دو معیار از ۱۲ معیار مورد استفاده در دو مدل ساخته شده با هم متفاوت هستند که این دو معیار توانسته‌اند حدود ۶ درصد صحت و دقت را بهبود بخشند. این امر نشان از تاثیر قابل ملاحظه‌ای این معیارها می‌باشد.

جدول ۷.۵: مقایسه‌ی معیارهای فرآیند و معیارهای ترکیبی جهش-فرآیند

معیار	نام روش	صحت	دقت	بازخوانی
فرآیند	Decition Tree	۰/۵۸۷	۰/۵۷۴	۰/۶۷۵
ترکیبی جهش-فرآیند	Decition Tree	۰/۶۷۵	۰/۷۰۵	۰/۶۰۰
فرآیند	SVM	۰/۶۶۲	۰/۶۸۵	۰/۶۰۰
ترکیبی جهش-فرآیند	SVM	۰/۶۳۷	۰/۶۶۶	۰/۵۵۰
فرآیند	Logestic Regression	۰/۶۱۲	۰/۵۹۱	۰/۷۲۵
ترکیبی جهش-فرآیند	Logestic Regression	۰/۶۷۵	۰/۶۵۲	۰/۷۵۰
فرآیند	Nueral Network	۰/۶۱۲	۰/۷۲۵	۰/۵۹۱
ترکیبی جهش-فرآیند	Nueral Network	۰/۷۵۰	۰/۷۹۴	۰/۶۷۵

در شکل ۴.۵ نمودارهای ROC به تفکیک روش دسته‌بندی آمده است. در هر یک از زیر شکل‌ها منحنی ROC مربوط به دو مدل با هم مقایسه شده است. در مدل اول که در ساخت آن از معیارهای فرآیند استفاده شده با خط ممند نمایش داده شده است و مدل دوم از جایگزینی دو معیار فرآیند با معیارهای ترکیبی جهش-فرآیند ساخته شده و با خط چین نمایش داده شده. همانطور که قابل مشاهده است در تمامی روش‌ها بجز SVM مدل دوم

مساحت زیر نمودار بیشتری نسبت به مدل اول داشته است.



شکل ۴.۵: نمودارهای ROC معیارهای فرآیند و به همراه جهش

در جدول ۸.۵ مساحت زیر نمودار ROC دو مدل به تفکیک روش دسته‌بندی آورده شده است. در میان روش‌های یادگیری به کار گرفته شده بیشترین افزایش مساحت زیر نمودار را روش Neural Network به مقدار ۰/۱۲۸ واحد داشته است. به طور متوسط ۰/۰۵۱ واحد در مدل‌ها بهبود مشاهده می‌شود. این موضوع نشان می‌دهد که معیارهای ترکیبی جهش-فرآیند از نظر مساحت زیر نمودار ROC نیز موجب تغییر مثبت ایجاد می‌کند.

با توجه به اینکه تنها روش SVM نتایج ضعیفی نسبت به سایرین داشته است این موضوع را می‌توان با توجه

نحوه‌ی عملکرد این روش توجیه کرد. به طور خلاصه این روش سعی می‌کند که فضای ویژگی^۲ را با ایجاد یک /برصفحه^۳ به دسته‌های مختلف تقسیم کند اما توزیع نقاط داده در فضای ویژگی به نحوی نیست که این روش بتواند به خوبی عمل کند.

جدول ۸.۵: مقادیر زیر نمودار ROC معیارهای فرآیند و معیارهای ترکیبی جهش-

معیار	Decition Tree	SVM	Logestic Regression	Neural Network
فرآیند	۵۹۶	۶۹۷	۶۴۳	۵۹۳
جهش-فرآیند	۶۵۴	۶۵۶	۷۰۵	۷۲۱

^۲Feature Space

^۳Hyperplane

[این صفحه آگاهانه خالی گذاشته شده است.]

فصل ۶

نتیجه‌گیری و کارهای آتی

در این پژوهش سعی شد که تاثیر معیارهای جهش پیش‌بینی خطا بر پیش‌خطا در هنگام قرار گیری در کنار معیارهای فرآیند ارزیابی شود و معیارهای جدیدی با استفاده از مفاهیم تحلیل جهش و تاریخچه‌ی توسعه‌ی نرم‌افزار ارائه گردد. در فصل ۱ به بیان مسئله و مفاهیم مقدماتی پرداخته شد. در فصل ۲ پژوهش‌های پیشین در حوزه‌ی پیش‌بینی خطا مورد بررسی قرار گرفت. پژوهشگران به طرق مختلف سعی در دستیابی به نتایج بهتری در پیش‌بینی خطا هستند. در این بررسی مشخص شد که در پژوهش‌های پیشین دو دسته‌ی کلی از معیارها مورد استفاده قرار گرفته است. این دسته‌ها عبارتند از معیارهای کد و معیارهای فرآیند. معیارهای فرآیند دارای مزیت‌ها بیشتری نسبت به معیارهای کد هستند و پژوهش‌های کمتری نیز به بررسی آنها پرداخته است. در یکی از پژوهش‌های اخیر از معیارهای جهش در کنار معیارهای کد به منظور پیش‌بینی خطا استفاده گردیده و موجب بهبود پیش‌بینی شده است.

پس از مشخص شدن بخش‌هایی از این حوزه که نیازمند تحقیق بیشتر هستند و شناسایی پتانسیل‌های موجود در معیارهای فرآیند و جهش در فصل ۳ راهکارهایی ارائه شدند تا با استفاده از معیارهای فرآیند و مفاهیم تحلیل جهش پیش‌بینی خطا بهبود یابد. در رویکرد اول معیارهای فرآیند در کنار معیارهای جهش قرار می‌گیرند و پیش‌بینی خطا با استفاده از آنها انجام می‌پذیرد. در رویکرد دوم، چهار معیار فرآیند مبتنی بر مفاهیم تحلیل جهش ارائه شده‌اند و در رویکرد سوم دو معیار فرآیند با استفاده از مفاهیم جهش اصلاح شدند و معیارهای ترکیبی جهش-فرآیند به وجود آمدند.

در فصل ۴ نحوه‌ی پیاده‌سازی هر یک از سه رویکرد ارائه شده و ابزارهای مورد استفاده شرح داده شد. به منظور انجام مطالعه‌ی موردی ۵ پروژه‌ی صنعتی جاوا مورد استفاده قرار گرفتند و معیارهای مورد بررسی در آنها استخراج شد. این معیارها برای دو گروه از پرونده‌ها که یکی حاوی خطا و دیگر سالم هستند محاسبه شده

است. در این دو گروه تعداد یکسانی پرونده وجود دارد. پرونده‌های حاوی خطا در مجموعه داده‌ی defects4j مشخص شده‌اند و پرونده‌های سالم به طور تصادفی انتخاب شدند.

معیارهای استخراج شده در فصل ۵ ارزیابی شدند. مدل‌های پیش‌بینی با استفاده از چهار روش دسته‌بندی ساخته شدند و عملکرد مدل‌ها با یکدیگر مقایسه گردید. نتایج ارزیابی نشان داد که معیارهای جهش زمانی که در کنار معیارهای فرآیند قرار گیرند می‌توانند تاثیر قابل توجهی در بهبود پیش‌بینی داشته باشند.

معیارهایی که تحت عنوان فرآیند مبتنی بر جهش ارائه شدند، زمانی که در کنار معیارهای فرآیند قرار می‌گیرند موجب بهبود پیش‌بینی خطا می‌شوند اما توانایی آنها بیشتر از معیارهای جهش نیست. از آنجا که این دسته از معیارها هزینه‌ی محاسباتی بیشترین دارند جایگزینی آنها با معیارهای جهش نمی‌تواند مزیتی داشته باشد. همچنین قرارگیری همه‌ی این معیارها در کنار هم نیز تاثیر مثبت چندانی نخواهد داشت.

معیارهای ترکیبی جهش-فرآیند به طور میانگین ۶ درصد در صحت، ۶/۶ درصد در دقت و ۵/۱ در مساحت زیر نمودار ROC تغییر مثبت ایجاد کرده است و از نظر معیار بازخوانی تغییر قابل توجهی ایجاد نشده است. این تغییرات نشان می‌دهد که اصلاح معیارهای فرآیند موفق آمیز بوده است و عرصه‌ی جدیدی را می‌توان به منظور ساخت معیارهای جدید در نظر گرفت و این عرصه ارائه‌ی معیارهای ترکیبی است. همچنین با توجه به این نکته تولید جهش‌یافته نیازمند وجود موارد آزمون نیست می‌توان برای این معیارها دامنه‌ی کاربرد وسیع‌تری در نظر گرفت.

در ادامه به گام‌هایی اشاره می‌شود که می‌توانند موجب جامعیت بخشید به نتایج این پژوهش شود و ابعاد دیگری از بکارگیری این معیارها مورد بررسی قرار گیرد.

• بررسی تاثیر استفاده از عملگرهای متفاوت:

در این پژوهش مجموعه‌ی محدودی از عملگرها جهت ساخت جهش‌یافته استفاده شده است. در پژوهش‌های آتی می‌توان به این موضوع پرداخت که افزایش و یا کاهش مجموعه‌ی عملگرهای جهش‌یافته چه تاثیری بر پیش‌بینی خطا داشته باشد. همچنین اینکه کدام نوع از عملگرهای مورد استفاده در استخراج معیارهای ارائه شده تاثیر بیشتری بر پیش‌بینی خطا دارد.

• ارزیابی معیارهای کد در کنار معیارهای ارائه شده:

همانطور که بیان شد معیارهای جهش می‌توانند به معیارهای فرآیند کمک کنند تا پیش‌بینی دقیق‌تری انجام شود. از طرف دیگر استفاده از معیارهای کد نیز می‌تواند به معیارهای جهش کمک کند و این

معیارها هزینه‌ی محاسباتی کمتری دارند. با توجه به پرهزینه بودن معیارهای جهش لازم است میزان بهبود پیش‌بینی خطا توسط آنها با معیارهای کد مقایسه شود و مشخص شود در هنگام قرارگیری در کنار معیارهای فرآیند مزیتی در مقابل معیارهای کد دارند یا خیر.

• **ساخت چهارچوب پیش‌بینی خطا با استفاده از پژوهش موجود:**

استخراج معیارها و ساخت مدل‌های پیش‌بینی در این پژوهش به صورت خودکار انجام می‌گیرد. با ایجاد تغییرات لازم می‌توان چهارچوبی ارائه داده که برای سایر پروژه‌های نرم‌افزاری نیز این معیارها را استخراج کند. با ایجاد یک چهارچوب هم انجام پژوهش‌های آتی توسط سایرین سهولت می‌یابد و هم ضمیمه‌ی به کارگیری پیش‌بینی خطا در صنعت توسعه می‌یابد.

[این صفحه آگاهانه خالی گذاشته شده است.]

پیوست آ

ساخت مدل‌های پیش‌بینی و ارزیابی

در این قسمت قطعه کدهای ساخت مدل‌های پیش‌بینی و ارزیابی آنها آورده شده است. قطعه کد ۱.آ مجموعه داده‌ها را آماده می‌کند و تنظیمات مربوط به آموزش مدل‌ها را انجام می‌دهد.

```
1 library(RMySQL);
2 library(caret);
3 library(pROC);
4 library(e1071)
5
6 mydb = dbConnect(MySQL(), user='root', password='1', dbname='
  bug_predict', host='127.0.0.1');
7
8 rs_mutation_metric = dbSendQuery(mydb, "select * from
  MutationMetric");
9 mutation_metircs = fetch(rs_mutation_metric, n=-1);
10 rs_process_metric = dbSendQuery(mydb, "select * from
  ProcessMetric ");
11 process_metircs = fetch(rs_process_metric, n=-1);
12
13 ##### clean up data #####
14 source("/home/ali/project/R-scripts/kill-live-to-score.R");
15
16 merged_metrics <- merge(x=clean_mutation_metircs, y=process_
  metircs, by.x="MetricId", by.y="ID")
17 lables<- as.factor(merged_metrics[,names(merged_metrics) %in%
  c("FILE_TYPE")]);
18
19
20 ##### test and train #####
21 b_number<-nrow(merged_metrics[merged_metrics$FILE_TYPE == "B"
  ,])
22 c_number<- nrow(merged_metrics[merged_metrics$FILE_TYPE == "C"
  ,])
23 smp_size_b <- floor(0.9 * b_number);
24 smp_size_c <- floor(0.9 * c_number);
25
```

```

26 ## set the seed to make your partition reproducible
27 set.seed(1423)
28 train_ind_b <- sample(seq_len(b_number), size = smp_size_b)
29 train_ind_c <- sample(seq_len(c_number), size = smp_size_c)
30 train_ind_c <- train_ind_c + b_number
31 train_ind <- c(train_ind_b, train_ind_c)
32
33 #####train control#####
34 MyFolds <- createMultiFolds(merged_metrics[train_ind,4], k =
    10, times=10)
35 train_control <- trainControl(method = "cv", index = MyFolds,
36 savePredictions = TRUE,
37 classProbs = TRUE
38 # ,summaryFunction = twoClassSummary
39 )
40
41

```

قطعه کد آ.۱: آماده‌سازی مجموعه داده

در قطعه کد آ.۲ پاک‌سازی داده‌ها و تبدیل داده‌های جهش به امتیاز جهش انجام می‌شود.

```

1 clean_mutation_metircs<- mutation_metircs[!is.na(mutation_
  metircs$Covered),];
2
3 for(i in 1:dim(clean_mutation_metircs)[1])
4 {
5   if(clean_mutation_metircs[i, 'Lived']==-1)
6   {
7     clean_mutation_metircs[i, 'Lived']<- 0;
8     clean_mutation_metircs[i, 'Killed']<-clean_mutation_metircs[
9       i, 'Killed']-1;
10   }
11 }
12 temp<-clean_mutation_metircs;
13 temp[,6]<-clean_mutation_metircs[,6]/clean_mutation_metircs
  [,5]
14 temp[,7]<-clean_mutation_metircs[,6]/clean_mutation_metircs
  [,2]
15 clean_mutation_metircs <- temp
16
17 for(i in 1:dim(clean_mutation_metircs)[1])
18 {
19   if(is.nan(clean_mutation_metircs[i, 'Lived']))
20     clean_mutation_metircs[i, 'Lived']<- 0;
21   if(is.nan(clean_mutation_metircs[i, 'Killed']))

```



```

22 clean_mutation_metrics[i, 'Killed'] <- 0;
23 }
24

```

قطعه کد ۲.آ: تبدیل داده‌های جهش به امتیاز جهش

در قطعه کد ۳.آ مدل‌های پیش‌بینی ساخته می‌شوند و با استفاده از داده‌های آزمون پیش‌بینی انجام می‌گیرد. تمامی معیارها در متغیر merged_metrics وجود دارند و با انتخاب ستون‌های مورد نظر زیر مجموعه‌ی مناسب انتخاب می‌شود. همچنین در تابع train روش دسته‌بندی انتخاب می‌گردد.

```

1
2 ##### Process Metrics #####
3 p_features <- merged_metrics[, -c(seq(1,13), 17, 20)];
4 model1 <- train(p_features[train_ind,], labels[train_ind],
5   trControl = train_control, method="nnet");
6 predict1_raw <- predict.train(model1, p_features[-train_ind,],
7   type="raw")
8 predict1_prob <- predict.train(model1, p_features[-train_ind,],
9   type="prob")
10
11 ##### Process Metrics with mutation #####
12
13 m_features1 <- merged_metrics[, !names(merged_metrics) %in% c("
14   FILE_TYPE", "MetricId", "FileType", "FileInfoId", "FILE_INFO_
15   ID")];
16 m_features1 <- m_features1[, -c(seq(5,10))];
17 model2 <- train(m_features1[train_ind,], labels[train_ind],
18   trControl=train_control, method="nnet");
19 predict1_raw <- predict.train(model2, m_features1[-train_ind,],
20   type="raw")
21 predict1_prob <- predict.train(model2, m_features1[-train_ind
22   ,], type="prob")
23
24 m_features2 <- merged_metrics[, !names(merged_metrics) %in% c("
25   FILE_TYPE", "MetricId", "FileType", "FileInfoId", "FILE_INFO_
26   ID")];
27 m_features2 <- m_features2[, -c(7,8)];
28 model3 <- train(m_features2[train_ind,], labels[train_ind],
29   trControl=train_control, method="nnet");
30 predict2_raw <- predict.train(model3, m_features2[-train_ind,],
31   type="raw")
32 predict2_prob <- predict.train(model3, m_features2[-train_ind
33   ,], type="prob")
34

```

23

24

قطعه کد آ.۳: ساخت مدل‌های پیش‌بینی

در قطعه کد آ.۴: پیش‌بینی‌های انجام شده ارزیابی می‌شوند.

```

1
2 auc(lables[-train_ind], predict1_prob$B)
3 auc(lables[-train_ind], predict2_prob$B)
4 auc(lables[-train_ind], predict3_prob$B)
5
6 plot(roc(lables[-train_ind], predict1_prob$B), col="blue",
      cex.lab=2, cex.axis=1.5);
7 plot(roc(lables[-train_ind], predict2_prob$B), add=TRUE, col=
  "red", lty=2);
8 plot(roc(lables[-train_ind], predict3_prob$B), add=TRUE, col=
  "black", lty=4);
9 legend(x="bottomright", col=c("blue", "red", "black"), lwd=3,
  legend=c("Process", "Process & Mutation", "Process &
  Mutation Base"), bty="n", lty=c(1,2,4), cex = 1.5)
10
11
12 confusionMatrix(predict1_raw, lables[-train_ind])
13 confusionMatrix(predict2_raw, lables[-train_ind])
14 confusionMatrix(predict3_raw, lables[-train_ind])
15
16

```

قطعه کد آ.۴: ساخت مدل‌های پیش‌بینی

پیوست ب

معیارهای استخراج شده

معیارهای فرآیند: 1. جدول

ID	ADEV	COMM	DDEV	FID	NADD	NDEL	B/C	MNR	OWN	EXP	OEXP	NCOMM	NADEV	NDDEV
	3	15	8	1	0.0128068	0.036284	B	6	0.547302	111	612	6.13351	2.45026	7.00262
2	2	8	8	2	0.00272765	0.00553797	B	3	0.757764	365	612	6.08269	2.44444	6.99483
3	3	14	8	3	0.0127718	0.0358162	B	6	0.549613	111	612	5.97644	2.42932	6.98168
4	3	5	7	4	0.000709747	0.000961693	B	2	0.75	538	1423	5.95026	2.42408	6.99215
5	2	5	8	5	0.00254708	0.00208261	B	3	0.902357	358	612	3.68372	1.76744	6.8
6	1	2	7	6	0.00106213	0	B	2	0.856	567	1410	3.64762	1.74286	6.79048
7	1	4	7	7	0.0121035	0.00429646	B	7	0.650492	141	612	3.58095	1.74286	6.79048
8	3	5	3	8	0.0161281	0.0058548	B	1	0.964602	867	1410	3.51575	1.79921	6.75591
9	3	20	3	9	0.0115356	0.0944777	B	1	0.83391	642	1410	3.10236	1.77953	6.73622
10	3	19	3	10	0.0115356	0.0890228	B	1	0.835616	642	1410	3.10236	1.77953	6.73622
11	1	2	7	11	0.00291261	0.00773395	B	5	0.486405	153	1410	1.87619	1.6381	6.7619
12	1	1	7	12	0.00275077	0	B	4	0.506211	139	1410	1.87619	1.6381	6.7619
13	0	0	8	13	0	0	B	6	0.686347	167	1406	0	0	0
14	1	1	10	14	0.00471698	0.125	B	11	0.424339	68	612	0	0	0
15	0	0	7	15	0	0	B	4	0.956801	308	206	0	0	0
16	0	0	7	16	0	0	B	7	0.665959	131	612	0	0	0
17	3	8	7	17	0.00160561	0.000578072	B	3	0.931818	493	1268	8.0847	3.66667	7.16667
18	5	10	8	18	0.0148416	0.0316134	B	5	0.492399	124	1265	7.70562	3.54157	6.79326
19	2	5	7	19	0.000886518	0.000201993	B	3	0.945205	434	1255	7.54521	3.38356	7.12329
20	7	55	10	20	0.0645949	0.223854	B	11	0.428071	65	609	6.83019	3.32075	6.80713
21	1	2	7	21	0.0167824	0.0043956	B	7	0.48744	122	1148	1.3125	1.01042	5.92708
22	1	2	6	22	0.00860016	0	B	4	0.561128	126	568	1.01042	1	5.9375
23	4	9	6	23	0.00469659	0	B	3	0.873727	336	131	10.2536	4.12738	5.9631
24	4	7	6	24	0.0144015	0.000608643	B	6	0.697821	110	568	7.14674	4.11775	5.97645
25	4	7	6	25	0.00405178	0.00125	B	3	0.989474	363	1101	5.08201	3.26984	5.92328
26	3	4	6	26	0.0187728	0	B	4	0.491747	122	1093	4.87293	3.20442	5.94475
27	2	3	6	27	0.0160076	0	B	6	0.7	108	568	3.56667	2.31111	5.88889
28	2	3	6	28	0.000784635	0	B	2	0.942029	485	1049	3.13333	2.02222	5.84444
29	3	4	6	29	0.0198223	0.976845	B	3	0.818531	95	378	3.08889	2.01111	5.84444
30	7	53	7	30	0.0578556	0.0221107	B	9	0.471508	60	568	26.3559	6.01524	6.01524
31	7	48	7	31	0.0578296	0.0212777	B	9	0.472757	60	568	25.3232	6.02518	6.02518
32	4	17	4	32	0.00917572	0.000662837	B	4	0.771341	152	360	14.5515	4.05618	4.05618
33	4	13	4	33	0.00974401	0.0000552608	B	3	0.275928	122	1001	13.5421	4.05584	4.05584
34	4	11	4	34	0.020985	0.000221496	B	3	0.763389	143	568	12.4844	4.05556	4.05556
35	3	7	3	35	0.00194652	0.00128448	B	4	0.49531	119	568	1.65	1.1	1.1
36	1	1	1	36	0.0000434254	0.000126207	B	5	0.706676	96	568	1.3625	1.0875	1.0875
37	2	3	2	37	0.000195652	0.00044259	B	3	0.503175	185	568	1.25	1.05	1.05
38	1	1	1	38	0.0000435076	0.000126526	B	4	0.491458	112	964	1.225	1.0625	1.0625
39	1	2	1	39	0.00142164	0.0000633794	B	10	0.526812	52	568	1.1	1	1
40	4	25	13	40	0.0414176	0.14308	B	10	0.531646	49	540	13.7593	2.22294	7.01454
41	3	33	10	41	0.0264353	0.0817106	B	3	0.272547	93	4	13.5083	2.28896	7.12405
42	2	11	9	42	0.000942426	0.00529865	B	3	0.39798	82	360	9.53704	1.98148	7.25926
43	0	0	4	43	0	0	B	2	0.974684	130	71	0	0	0
44	0	0	8	44	0	0	B	5	0.566011	86	760	0	0	0
45	0	0	7	45	0	0	B	4	0.73053	101	540	0	0	0
46	0	0	8	46	0	0	B	2	0.353488	73	21	0	0	0

فصل ب: معیارهای استخراج شده

جدول ب.۱ ادامه از صفحه قبل

ID	ADEV	COMM	DDEV	FID	NADD	NDEL	B/C	MNR	OWN	EXP	OEXP	NCOMM	NADEV	NDDEV
47	1	3	7	47	0.0468673	0.0000886977	B	5	0.941003	45	540	4.98429	1.30366	6.10471
48	2	7	7	48	0.0127311	0.015819	B	3	0.715302	101	540	4.54196	1.18881	5.66783
49	2	6	7	49	0.016262	0.0229752	B	3	0.576882	72	540	4.49355	1.16774	5.74194
50	2	7	8	50	0.027339	0.0341798	B	4	0.499714	84	652	4.19011	1.1673	5.89354
51	1	6	8	51	0.0164773	0.0212923	B	4	0.628601	84	540	4.06129	1.13548	5.70323
52	1	3	8	52	0.0115457	0.00004483	B	2	0.442504	79	21	4.10471	1.1623	6.03141
53	1	4	6	53	0.0195429	0.0298941	B	3	0.738384	195	540	3.70609	1.00717	5.60573
54	4	10	4	54	0.00212146	0.000895962	B	3	0.969072	231	540	27.9639	5.06024	5.06024
55	6	20	6	55	0.00254599	0.00238468	B	4	0.494297	93	540	30.7631	5.30485	5.30485
56	7	33	7	56	0.0219584	0.0286736	B	4	0.493656	77	459	28.8216	5.21252	5.21252
57	4	9	4	57	0.00213237	0.000894165	B	3	0.972509	221	527	27.3012	4.98795	4.98795
58	7	29	7	58	0.0116996	0.00442915	B	4	0.696374	77	527	30.3367	5.32465	5.32465
59	7	46	7	59	0.0172662	0.00710118	B	4	0.932741	31	527	27.5	5.0748	5.0748
60	7	45	7	60	0.0172631	0.00707391	B	4	0.933587	31	527	27.3032	5.0748	5.0748
61	7	44	7	61	0.0172511	0.00705857	B	4	0.934405	30	527	27.3032	5.0748	5.0748
62	7	29	7	62	0.00805702	0.00569946	B	5	0.622363	72	21	29.672	5.35763	5.35763
63	5	33	5	63	0.00763639	0.00931936	B	1	0.485225	130	421	28.8138	5.27924	5.27924
64	4	7	4	64	0.00135935	0.0001916	B	3	0.974227	114	317	25.5814	5.30233	5.30233
65	5	45	5	65	0.010678	0.0173784	B	2	0.776008	165	511	25.8019	5.2673	5.2673
66	0	0	2	66	0	0	B	1	0.996997	452	863	0	0	0
67	0	0	4	67	0	0	B	1	0.982036	443	843	0	0	0
68	2	5	6	68	0.0614928	0.361991	B	3	0.84507	628	826	0	0	0
69	0	0	2	69	0	0	B	0	0.922535	1123	810	0	0	0
70	0	0	2	70	0	0	B	1	0.983753	1094	795	0	0	0
71	2	2	2	71	0.00238075	0.00155925	B	1	0.997375	384	792	1.34	1.28	2.04
72	0	0	2	72	0	0	B	0	0.945626	1089	792	0	0	0
73	1	1	1	73	0.0233219	0	B	0	1	1476	1476	1	1	1
74	0	0	2	74	0	0	B	1	0.977778	1074	792	0	0	0
75	0	0	2	75	0	0	B	1	0.981818	501	1443	0	0	0
76	0	0	1	76	0	0	B	0	1	788	788	0	0	0
77	0	0	1	77	0	0	B	0	1	775	775	0	0	0
78	0	0	5	78	0	0	B	2	0.847518	736	770	0	0	0
79	1	2	2	79	0.117739	0.360577	B	0	0.903346	1051	769	1.92308	1	1.92308
80	2	8	2	80	0.0107618	0.0207951	B	1	0.97852	1011	715	9.6	1.6	1.6
81	2	7	2	81	0.00953698	0.0196902	B	1	0.978397	1009	712	9.6	1.6	1.6
82	0	0	1	82	0	0	B	0	1	654	654	0	0	0
83	4	5	4	83	0.00173706	0.000550782	B	1	0.999276	298	630	3.57857	2.52857	3.47857
84	4	4	4	84	0.000747175	0.000449438	B	1	0.999265	294	624	3.52857	2.51429	3.46429
85	3	3	4	85	0.00014018	0.000337117	B	1	0.999258	294	623	3.52857	2.51429	3.46429
86	0	0	1	86	0	0	B	0	1	620	620	0	0	0
87	1	2	1	87	0.0018582	0.00394777	B	0	1	619	619	2.48387	1.41935	1.58065
88	3	3	3	88	0.00303662	0.00365079	B	0	1	617	617	2.97857	2.35714	3.45
89	2	2	3	89	0.00123548	0.0033428	B	0	1	613	613	2.90714	2.35714	3.45
90	2	2	3	90	0.000170445	0.000486027	B	0	1	600	600	2.60714	2.3	3.45
91	1	1	1	91	0.000128899	0.00027027	B	0	1	581	581	0	0	0
92	0	0	1	92	0	0	B	0	1	580	580	0	0	0
93	1	1	2	93	0.00504332	0.00407277	B	0	0.828947	244	580	5	3	4
94	2	2	2	94	0.00198747	0	B	1	0.995128	370	545	1.34572	1.14216	1.14863
95	1	1	1	95	0.000566542	0	B	0	1	544	544	1.30357	1.13149	1.13799
96	1	1	1	96	0.000485492	0	B	0	1	540	540	1.23377	1.11364	1.12013
97	1	1	1	97	0.00114608	0	B	0	1	539	539	1.23052	1.11039	1.11688
98	1	1	1	98	0.00151742	0	B	0	1	536	536	1.19805	1.09091	1.0974
99	2	4	2	99	0.000597061	0.000273561	B	0	0.595238	158	525	1.19163	1.07729	1.08374
100	2	2	2	100	0.000270582	0.00000781953	B	1	0.990909	148	524	1.15645	1.07097	1.07742
101	4	14	7	101	0.000919797	0.00198361	B	2	0.892635	717	1324	8.70529	3.35157	8.31795
102	7	26	15	102	0.0111132	0.022023	B	6	0.497545	213	450	8.30787	3.30443	8.17418
103	4	33	4	103	0.036146	0.0311197	B	2	0.996774	570	441	8.19219	3.1021	7.17117
104	3	14	8	104	0.000421907	0.00098547	B	0	0.593496	1205	1100	8.33802	3.26382	8.21588
105	2	6	2	105	0.00463703	0.000149512	B	1	0.95466	756	1320	7.98129	3.12286	7.39707
106	4	9	12	106	0.000211438	0.000375714	B	5	0.812903	312	1096	7.93123	3.25432	8.12562
107	3	10	7	107	0.000695985	0.00126965	B	3	0.964401	695	1315	7.92343	3.25135	8.08701
108	4	10	12	108	0.000305066	0.000470288	B	4	0.655405	311	1091	7.74331	3.25478	8.1293
109	3	17	9	109	0.00173969	0.00359657	B	3	0.977215	427	1297	7.55917	3.09096	8.0794

جدول ب.۱ ادامه از صفحه قبل

ID	ADEV	COMM	DDEV	FID	NADD	NDEL	B/C	MNR	OWN	EXP	OEXP	NCOMM	NADEV	NDDEV
110	3	7	9	110	0.00116361	0.00254561	B	1	0.801471	420	1295	7.34906	3.09151	8.1412
111	6	18	14	111	0.0134088	0.0283935	B	5	0.52071	209	334	6.53272	2.92841	7.92379
112	6	17	14	112	0.013349	0.0283322	B	5	0.516156	209	333	6.53272	2.92841	7.92379
113	2	12	2	113	0.00311388	0.00074855	B	0	0.749064	642	332	6.4344	2.8106	7.21199
114	3	6	9	114	0.0037814	0.00849174	B	2	0.635676	378	60	4.78545	1.99629	7.89064
115	2	10	2	115	0.00288384	0.00102459	B	0	0.755474	604	296	4.65812	1.91966	6.86496
116	2	9	2	116	0.00272836	0.000863038	B	0	0.744275	600	292	4.64786	1.91624	6.86496
117	1	5	1	117	0.00810906	0.0000841142	B	1	0.717614	713	1033	4.63181	1.88591	6.93258
118	3	5	13	118	0.00107783	0.00180432	B	6	0.652941	201	1013	2.85045	1.13694	6.82883
119	0	0	3	119	0	0	B	1	0.998333	612	1152	0	0	0
120	1	2	10	120	0.000302797	0.000534386	B	2	0.580038	533	973	1.90469	1.01939	7.83522
121	1	2	5	121	0.000489146	0.000863487	B	1	0.930693	435	223	1.90145	1.01616	7.85137
122	2	2	4	122	0.0000352779	0.0000020903	B	1	0.981982	1040	1113	285.586	5.47475	9.23232
123	5	264	5	123	0.0018022	0.00297603	B	0	0.788274	455	190	228.656	5.19663	8.21857
124	5	119	5	124	0.00303879	0.0019702	B	1	0.993417	393	1028	105.176	5.22967	8.28731
125	3	52	12	125	0.00730388	0.0122863	B	3	0.468254	150	894	45.1615	3.34703	6.80644
126	3	5	10	126	0.000284023	0.000128006	B	1	0.623762	196	885	7.44976	2.2488	7.33333
127	3	18	3	127	0.0338367	0	B	0	1	52	52	32.1399	3.77648	5.25267
128	4	14	13	128	0.029784	0.0448739	B	4	0.300147	116	937	4.39914	2.07868	6.9814
129	3	9	6	129	0.00784226	0.0121401	B	2	0.980616	308	935	3.84991	2.09381	7.08255
130	1	2	5	130	0.000850857	0.00243605	B	1	0.96206	902	935	2.71472	1.24233	6.61656
131	1	2	6	131	0.0134288	0.0191809	B	1	0.614754	278	918	1.5	1.5	6
132	1	1	4	132	0.000677966	0.00162955	B	1	0.996865	893	917	1.55696	1.24684	6.82911
133	0	0	5	133	0	0	B	1	0.978622	882	906	0	0	0
134	0	0	4	134	0	0	B	1	0.957295	881	858	0	0	0
135	0	0	5	135	0	0	B	1	0.385321	189	191	0	0	0
136	4	4	5	136	0.00166568	0.00162971	B	0	1	898	898	2.99926	2.85788	6.51147
137	4	4	5	137	0.00243947	0.00198362	B	2	0.485893	242	869	2.74091	2.732	6.43059
138	3	3	5	138	0.00236469	0.00183132	B	2	0.469453	242	868	2.74091	2.732	6.43059
139	4	31	4	139	0.00361271	0.00624287	B	0	0.649254	833	805	22.3234	4.03917	5.57057
140	6	40	12	140	0.00598912	0.00531258	B	5	0.60371	116	803	22.3126	4.13558	5.85222
141	3	24	3	141	0.0028228	0.00591384	B	2	0.995485	327	857	21.352	3.79335	5.14127
142	2	10	2	142	0.00477387	0.00901629	B	2	0.988146	326	855	20.3882	3.6837	4.93198
143	4	28	4	143	0.00343733	0.000632547	B	0	1	840	840	22.1388	4.03829	5.58655
144	4	49	10	144	0.0149779	0.00440546	B	3	0.372219	179	89	21.8111	3.99346	5.72256
145	3	32	3	145	0.013617	0.014184	B	2	0.989148	316	833	21.4336	3.83468	5.27927
146	3	31	3	146	0.0136105	0.0140701	B	2	0.989125	315	832	21.4336	3.83468	5.27927
147	3	10	3	147	0.000777031	0.000225296	B	1	0.994872	780	787	18.8654	3.70656	4.99574
148	2	14	2	148	0.00219024	0.00103681	B	1	0.998165	780	786	18.6094	3.6827	4.92674
149	3	14	3	149	0.000595966	0.00151013	B	0	1	785	785	18.5477	3.65809	4.90373
150	3	12	3	150	0.00134856	0.000644745	B	2	0.580913	228	739	17.7389	3.91878	5.31803
151	1	3	1	151	0.000376971	0.000196367	B	1	0.991501	154	645	16.4474	2.86842	3.21053
152	1	6	1	152	0.000608795	0.000834684	B	0	1	644	644	6.73333	1.53333	1.6
153	1	5	1	153	0.00059108	0.00017199	B	0	1	637	637	6.73333	1.53333	1.6
154	6	26	12	154	0.00654088	0.00414839	B	3	0.718367	104	715	16.7875	4.30351	6.22681
155	6	25	12	155	0.00652192	0.00412208	B	4	0.744186	103	715	16.7875	4.30351	6.22681
156	5	26	8	156	0.00769247	0.00233448	B	1	0.973214	616	539	16.6057	4.23945	6.17252
157	2	17	8	157	0.0152619	0.00183389	B	1	0.631919	205	85	13.0007	3.21426	5.52836
158	2	15	8	158	0.0152315	0.00159763	B	1	0.644528	205	85	12.9973	3.21426	5.52836
159	2	13	8	159	0.0152571	0.00154358	B	1	0.645794	204	85	12.9771	3.21426	5.52836
160	1	7	9	160	0.00298498	0	B	3	0.712042	127	191	9.09375	1.7893	5.19176
161	3	16	9	161	0.0138152	0.00174694	B	5	0.754846	116	667	9.7114	1.95163	5.0804
162	3	10	10	162	0.00445004	0.0153273	B	3	0.468992	122	191	6.65639	1.68871	5.0542
163	4	15	7	163	0.0239796	0.0431446	B	2	0.532742	112	84	6.06077	1.73156	5.21357
164	3	22	9	164	0.0153877	0.00201684	B	1	0.57725	208	85	14.939	4.06551	6.17767
165	0	0	2	165	0	0	B	1	0.996678	381	221	0	0	0
166	0	0	8	166	0	0	B	4	0.909427	97	80	0	0	0
167	2	5	6	167	0.0064838	0.00356528	B	3	0.866324	92	620	4.42157	1.91863	5.01078
168	8	49	8	168	0.00582189	0.00550822	B	3	0.590909	78	601	35.0512	5.89929	5.89929
169	8	58	8	169	0.00763772	0.0047332	B	4	0.480916	72	191	35.2172	5.95529	5.95529
170	4	19	4	170	0.0069222	0.0022238	B	2	0.890873	103	191	25.6804	4.74862	4.74862
171	4	14	4	171	0.0022944	0.000266999	B	2	0.990385	84	65	25.7132	4.74937	4.74937
172	0	0	2	172	0	0	B	1	0.978984	574	863	0	0	0

جدول ب.۱ ادامه از صفحه قبل

ID	ADEV	COMM	DDEV	FID	NADD	NDEL	B/C	MNR	OWN	EXP	OEXP	NCOMM	NADEV	NDDEV
173	0	0	2	173	0	0	B	1	0.990431	1123	810	0	0	0
174	3	3	3	174	0.0025508	0.00831601	B	2	0.995086	605	792	1.52727	1.25455	1.96364
175	2	2	2	175	0.00263583	0.00623701	B	1	0.995455	384	792	1.34	1.28	2.04
176	2	2	3	176	0.00170054	0.000519751	B	2	0.950276	605	792	1.3	1.225	2.175
177	2	2	2	177	0.000935295	0.00519751	B	1	0.99924	384	792	1.34	1.28	2.04
178	1	1	2	178	0.000850268	0.00155925	B	0	1	792	792	1.58333	1.58333	1.75
179	2	2	3	179	0.00178556	0.0010395	B	2	0.991649	605	792	1.3	1.225	2.175
180	1	2	2	180	0.0282575	0.0865385	B	0	0.859155	1051	769	1.92308	1	1.92308
181	1	2	1	181	0.00191626	0.00349226	B	0	1	619	619	2.48387	1.41935	1.58065
182	4	4	5	182	0.00317873	0.00375327	B	0	0.632754	865	835	2.99926	2.85788	6.51147
183	4	20	4	183	0.00452031	0.0142382	B	2	0.779978	200	51	20.2497	3.66905	4.89733
184	4	14	10	184	0.0191731	0.0389961	B	5	0.671477	113	664	6.06903	1.73156	5.18879
185	2	2	7	185	0.00285624	0.419355	B	1	0.642045	101	2095	1.00662	1.00497	3.25993
186	1	2	2	186	0.000405999	0	B	0	0.538462	129	2095	2.01874	1.01363	3.15503
187	2	3	7	187	0.0030521	0.454545	B	1	0.604278	105	2095	1.00662	1.00497	3.25993
188	0	0	17	188	0	0	B	11	0.583333	0	2095	0	0	0
189	2	4	3	189	0.00215506	0.00487439	B	0	0.585185	47	1	3.02901	1.02218	3.13652
190	3	3	9	190	0.0134027	0.00836987	B	2	0.726136	0	2073	1.15049	1.14563	3.26214
191	4	8	9	191	0.00964814	0.0061583	B	4	0.984026	38	287	2.36371	1.21264	3.23153
192	3	4	9	192	0.00943847	0.00495616	B	3	0.985623	61	286	1.18887	1.08432	3.19224
193	1	3	4	193	0.000606003	0	B	0	0.868421	208	1984	3.09149	1.04892	3.16264
194	1	3	12	194	0.00254287	0	B	3	0.828025	26	279	3.02515	1.00387	3.20696
195	1	1	4	195	0.000942829	0	B	1	0.982759	74	3	1	1	2.23092
196	1	1	1	196	0.00333511	0	B	0	1	1372	1372	9	1	1
197	2	25	4	197	0.0529013	0.12664	B	0	0.811475	173	1361	9.16864	1.68047	2.08876
198	2	23	4	198	0.0549564	0.129141	B	0	0.801724	164	1347	8.95166	1.68278	2.09063
199	1	1	1	199	0.0674342	0	B	0	1	2	2	1.4	1.2	1.2
200	3	18	4	200	0.0583255	0.0838059	B	3	0.964309	70	1238	5.82883	1.65766	2.11712
201	2	9	2	201	0.0164766	0.0203562	B	0	0.905882	157	1233	5.65537	1.58192	2.16949
202	0	0	7	202	0	0	B	2	0.772358	91	2095	0	0	0
203	2	6	6	203	0.0018648	0.000790005	B	1	0.9	145	371	4.46492	1.44749	3.31216
204	1	15	1	204	0.0015759	0.125265	B	0	0.854167	0	339	4.33333	1.19048	1.57143
205	2	5	2	205	0.000465475	0.057554	B	0	1	1952	1952	2.0872	1.05118	3.19052
206	2	2	3	206	0.000786318	0.00145243	B	0	1	1624	1624	0	0	0
207	5	11	9	207	0.0295193	0.0338882	B	2	0.736	33	246	5.76471	2.94118	5.57353
208	1	1	4	208	0.00457235	0.00254453	B	1	0.796748	206	1623	0	0	0
209	0	0	6	209	0	0	B	0	0.565217	63	1616	0	0	0
210	2	4	2	210	0.00154642	0.00341459	B	0	0.661765	420	1590	5.44884	1.69767	3.09535
211	3	5	4	211	0.00400827	0.00673212	B	1	0.9	112	1499	2.86447	1.52015	1.96337
212	3	7	3	212	0.0957179	0.202115	B	0	0.578947	132	1461	3.53846	1.38462	1.73077
213	2	4	3	213	0.00976404	0.0194489	B	0	0.902439	179	1457	5.21875	1.875	2.75
214	1	10	3	214	0.0532623	0.10177	B	1	0.998192	201	1449	3.2	1.9	2.1
215	0	0	2	215	0	0	B	0	0.945205	178	1449	0	0	0
216	0	0	1	216	0	0	B	0	1	1423	1423	0	0	0
217	0	0	3	217	0	0	B	1	0.97561	175	1399	0	0	0
218	2	6	3	218	0.0049007	0.0147136	B	1	0.973214	158	1252	6.09783	1.53261	2.11957
219	1	4	3	219	0.0215827	0.0206795	B	1	0.968011	177	1169	3.57143	2	2.28571
220	2	6	3	220	0.0252033	0.0676819	B	0	0.9447	96	1156	2.28866	1.30928	2.10309
221	0	0	1	221	0	0	B	0	1	1153	1153	0	0	0
222	1	8	1	222	0.00141588	0.00323485	B	0	1	1069	1069	6.29167	1	1.04167
223	2	17	3	223	0.0118186	0.0226504	B	0	0.891026	226	1347	10.6652	1.7533	2.1674
224	2	7	3	224	0.0084666	0.0195337	B	0	0.863309	216	1238	6.58242	1.71429	2.23077
225	2	11	2	225	0.0202788	0.0388041	B	0	0.925532	157	1233	6.46226	1.69811	2.22642
226	2	5	4	226	0.000804557	0.00128742	B	0	0.916667	881	371	4.50812	1.48144	3.35228
227	2	5	4	227	0.000293186	0.000409632	B	0	0.777778	881	371	4.50812	1.48144	3.35228
228	2	5	4	228	0.000565917	0.000877783	B	0	0.882353	881	371	4.50812	1.48144	3.35228
229	2	5	4	229	0.000497735	0.000760746	B	0	0.866667	881	371	4.50812	1.48144	3.35228
230	1	1	3	230	0.0146471	0.0221239	B	1	0.822785	31	1449	0	0	0
231	1	2	2	231	0.0163711	0.01373	B	1	0.998006	723	1327	1	1	2
232	1	1	2	232	0.00554017	0.00691244	B	1	0.997996	722	1326	0	0	0
233	0	0	2	233	0	0	B	0	0.813906	722	1325	0	0	0
234	0	0	2	234	0	0	B	1	0.997994	722	1324	0	0	0
235	0	0	2	235	0	0	B	1	0.984137	719	1315	0	0	0

جدول ب.۱ ادامه از صفحه ی قبل

ID	ADEV	COMM	DDEV	FID	NADD	NDEL	B/C	MNR	OWN	EXP	OEXP	NCOMM	NADEV	NDDEV
236	1	2	4	236	0.0259823	0.0346275	B	1	0.648381	159	1285	1.4	1	2
237	2	2	5	237	0.00869565	0.0377358	B	2	0.937895	53	1278	0	0	0
238	1	1	2	238	0.0469799	0.142857	B	1	0.960184	706	1268	2	2	3
239	0	0	2	239	0	0	B	1	0.959199	706	1267	0	0	0
240	0	0	2	240	0	0	B	1	0.997093	705	1264	0	0	0
241	0	0	4	241	0	0	B	1	0.902123	143	1254	0	0	0
242	2	3	3	242	0.0110497	0.0467836	B	1	0.998571	694	1224	2.5	1.66667	2.66667
243	0	0	4	243	0	0	B	0	0.729657	691	1215	0	0	0
244	0	0	2	244	0	0	B	1	0.997222	689	1205	0	0	0
245	0	0	4	245	0	0	B	1	0.699088	141	1203	0	0	0
246	0	0	4	246	0	0	B	1	0.941799	141	1201	0	0	0
247	1	1	2	247	0.00892857	0.0196078	B	1	0.958529	683	1186	0	0	0
248	0	0	4	248	0	0	B	1	0.615032	140	1176	0	0	0
249	0	0	2	249	0	0	B	1	0.958529	679	1173	0	0	0
250	0	0	4	250	0	0	B	1	0.77825	140	1172	0	0	0
251	1	1	2	251	0.00129627	0.00152161	B	1	0.988764	677	1165	1.91026	1.28205	2.11538
252	0	0	2	252	0	0	B	0	0.942244	653	1085	0	0	0
253	1	12	2	253	0.00114952	0.0189017	B	1	0.958496	653	1084	2.01818	1.01818	1.92727
254	1	2	2	254	0.000434994	0.00562516	B	0	0.730693	652	394	2.22642	1.01887	1.92453
255	1	9	2	255	0.000848508	0.0164271	B	1	0.957464	649	1071	1.87037	1.01852	1.92593
256	1	1	2	256	0.0000295212	0.00102907	B	0	0.700997	649	1070	2	1.01923	1.92308
257	1	2	2	257	0.000148211	0.00524659	B	0	0.728296	645	1056	1.75	1.01923	1.92308
258	1	1	2	258	0.00409277	0	B	1	0.992908	723	1327	2	1	2
259	0	0	2	259	0	0	B	1	0.992832	722	1326	0	0	0
260	1	2	2	260	0.031768	0.134503	B	1	0.998279	694	1224	1.5	1.25	2.25
261	0	0	10	261	0	0	B	1	0.331307	131	217	0	0	0
262	2	2	12	262	0.0553746	0.00795756	B	2	0.615692	96	41	1	1	9
263	0	0	9	263	0	0	B	2	0.619318	111	41	0	0	0
264	0	0	7	264	0	0	B	2	0.535	237	41	0	0	0
265	0	0	4	265	0	0	B	1	0.783262	154	83	0	0	0
266	2	4	8	266	0.00379363	0.00130528	B	6	0.702002	66	41	8.03448	3.44828	10.931
267	0	0	2	267	0	0	B	1	0.782303	230	41	0	0	0
268	0	0	9	268	0	0	B	1	0.581967	179	41	0	0	0
269	1	1	3	269	0.00691085	0.0115108	B	1	0.90106	122	6	0	0	0
270	2	2	15	270	0.033805	0.135802	B	8	0.475998	19	41	1.07692	1.07692	6.69231
271	1	3	12	271	0.224949	0.318182	B	4	0.646082	81	41	3	1	9.44444
272	0	0	7	272	0	0	B	3	0.97193	224	41	0	0	0
273	0	0	5	273	0	0	B	3	0.663462	141	442	0	0	0
274	0	0	8	274	0	0	B	3	0.879962	183	41	0	0	0
275	3	5	9	275	0.00446429	0.00627287	B	3	0.674044	106	41	2.90535	1.60082	6.79012
276	2	3	5	276	0.0218997	0.00698549	B	2	0.541841	130	215	1.99351	1.35714	6.41558
277	2	6	8	277	0.00660851	0.0107527	B	5	0.644059	43	632	2.89377	1.62637	6.77656
278	2	3	20	278	0.0076679	0.00645856	B	12	0.514435	18	41	1.99363	1.36306	6.34395
279	1	2	2	279	0.00847009	0	B	1	0.795745	222	41	3.5	1.25	6.5
280	2	2	11	280	0.00362117	0.00717835	B	4	0.45979	69	215	2.04459	1.36306	6.43312
281	2	3	8	281	0.0182584	0.0368899	B	1	0.493506	221	436	2.00645	1.35484	6.50968
282	1	1	8	282	0.000574878	0.00118835	B	2	0.515152	75	435	1.89542	1.34641	6.41176
283	2	2	8	283	0.00191644	0.00363636	B	1	0.514677	261	429	1.89785	1.39247	6.45161
284	1	1	5	284	0.000496278	0.000903342	B	2	0.647059	127	215	1.60131	1.30065	6.42484
285	2	5	9	285	0.0192853	0.0439078	B	2	0.687366	111	41	1.51579	1.33158	6.46842
286	2	2	3	286	0.002849	0.00329308	B	1	0.934066	106	5	1.42484	1.24837	6.43137
287	1	1	2	287	0.00652985	0.00516796	B	1	0.987931	490	409	1.30303	1.06061	6.72727
288	0	0	6	288	0	0	B	1	0.989796	155	41	0	0	0
289	0	0	4	289	0	0	B	2	0.844944	141	83	0	0	0
290	1	1	9	290	0.000334672	0	B	4	0.795402	107	41	1.61905	1	6.09524
291	1	7	20	291	0.0417094	0.0613668	B	12	0.53432	17	41	1.5814	1.02326	6.93023
292	1	1	11	292	0.00278552	0	B	5	0.765032	100	41	1	1	7.66667
293	0	0	7	293	0	0	B	3	0.592401	68	41	0	0	0
294	0	0	9	294	0	0	B	2	0.668852	75	41	0	0	0
295	0	0	9	295	0	0	B	2	0.682236	106	41	0	0	0
296	0	0	6	296	0	0	B	3	0.777778	79	41	0	0	0
297	0	0	11	297	0	0	B	4	0.467375	27	41	0	0	0
298	0	0	7	298	0	0	B	2	0.87291	203	41	0	0	0

فصل ب: معیارهای استخراج شده

جدول ب.۱ ادامه از صفحه قبل

ID	ADEV	COMM	DDEV	FID	NADD	NDEL	B/C	MNR	OWN	EXP	OEXP	NCOMM	NADEV	NDDEV
299	1	1	7	299	0.000736377	0.000982318	B	3	0.620253	66	41	1	1	7.52174
300	0	0	10	300	0	0	B	4	0.730599	59	41	0	0	0
301	1	1	8	301	0.0000304047	0.00115473	B	2	0.572629	61	41	3.08696	1.31884	6.97101
302	3	7	11	302	0.00185502	0.0592992	B	4	0.470588	27	41	1.71605	1.12346	3.11111
303	2	4	8	303	0.00136952	0.0454545	B	5	0.630928	39	487	2.44444	1.18519	6.97531
304	0	0	7	304	0	0	B	2	0.893836	196	41	0	0	0
305	1	2	10	305	0.00235849	0.00328587	B	1	0.341237	114	214	2.22973	1.11486	6.87162
306	0	0	7	306	0	0	B	0	0.747541	92	41	0	0	0
307	0	0	8	307	0	0	B	3	0.756345	66	337	0	0	0
308	3	10	8	308	0.0361398	0.016132	B	5	0.616976	37	446	2.84536	1.30928	7.11856
309	0	0	4	309	0	0	B	2	0.641476	185	41	0	0	0
310	1	1	2	310	0.0317228	0.00145349	B	0	0.898596	24	75	0	0	0
311	0	0	6	311	0	0	B	1	0.909722	154	41	0	0	0
312	0	0	9	312	0	0	B	2	0.730579	69	41	0	0	0
313	0	0	4	313	0	0	B	2	0.885845	106	74	0	0	0
314	1	5	7	314	0.0568769	0.102339	B	4	0.603624	31	372	3.8875	1.0375	7.85
315	0	0	8	315	0	0	B	4	0.986891	71	41	0	0	0
316	0	0	5	316	0	0	B	1	0.719078	173	41	0	0	0
317	3	8	10	317	0.000892638	0.013738	B	2	0.6875	78	41	5.64516	3.19355	8.29032
318	0	0	6	318	0	0	B	2	0.989899	145	41	0	0	0
319	6	18	16	319	0.00091932	0.00727748	B	10	0.64786	27	41	7.0122	2.5122	6.53659
320	4	15	13	320	0.000717892	0.0117107	B	7	0.574221	19	41	3.63014	2.0274	7.0274
321	3	12	12	321	0.000620041	0.0121825	B	6	0.576271	26	41	3.44	2.04	7.28
322	1	2	7	322	0.0000313959	0.0010433	B	2	0.88	88	41	0	0	0
323	0	0	6	323	0	0	B	1	0.885714	138	41	0	0	0
324	3	10	15	324	0.00438394	0.00598314	B	9	0.66451	26	41	6.67568	2.2973	6.21622
325	3	5	8	325	0.00620484	0.00818833	B	1	0.730579	102	41	3.15	1.7	8.2
326	3	7	9	326	0.0184012	0.0340502	B	3	0.798232	99	41	3.5625	1.52083	7.60417
327	2	2	8	327	0.000803859	0.00524194	B	3	0.816092	81	41	2.95652	1.78261	7.91304
328	2	5	10	328	0.00841314	0.0149858	B	4	0.794271	68	41	3.85294	1.55882	7.23529
329	2	6	9	329	0.0177007	0.04046	B	3	0.804929	97	41	3.39583	1.47917	7.58333
330	1	2	6	330	0.000202737	0.00260281	B	3	0.609056	45	275	2.4375	1.5	8.125
331	1	2	6	331	0.00130634	0.00118064	B	4	0.860681	57	41	6.11111	2.55556	8.66667
332	0	0	3	332	0	0	B	2	0.946636	129	41	0	0	0
333	1	2	7	333	0.0208599	0.0205066	B	1	0.743504	144	41	1.7	1.15	8
334	1	1	7	334	0.0626609	0	B	1	0.508524	146	207	4	1	11
335	0	0	11	335	0	0	B	5	0.58952	26	41	0	0	0
336	0	0	7	336	0	0	B	0	0.759887	134	41	0	0	0
337	1	4	7	337	0.00259703	0.00475561	B	1	0.764655	134	41	5.91667	2.5	7.5
338	2	9	7	338	0.0413371	0.0303797	B	1	0.53366	129	206	6	2.35714	8.28571
339	0	0	7	339	0	0	B	0	0.685921	119	41	0	0	0
340	2	5	11	340	0.0563596	0.0212977	B	5	0.592929	23	41	5.4	2.3	7.6
341	1	1	8	341	0.0014758	0	B	2	0.74493	54	41	2	1.66667	6.77778
342	2	4	8	342	0.0447046	0.0284784	B	1	0.747082	35	41	2.59259	1.55556	6.51852
343	3	27	3	343	0.00453093	0.010667	B	1	0.725522	63	205	12.3185	5.0716	5.0716
344	7	28	7	344	0.00983201	0.0133947	B	1	0.788129	110	41	11.7421	5.91998	5.91998
345	8	13	8	345	0.00193304	0.00259492	B	1	0.476987	51	41	10.9408	5.89518	5.89518
346	10	58	10	346	0.0183851	0.0142857	B	4	0.701186	20	41	11.9927	5.80731	5.80731
347	7	15	7	347	0.0077011	0.00538134	B	1	0.820298	71	205	11.1359	5.94548	5.94548
348	6	15	6	348	0.00261068	0.00255873	B	0	0.802899	103	41	10.8128	5.81157	5.81157
349	6	12	6	349	0.00714672	0.00133296	B	2	0.968401	99	41	10.3545	5.68724	5.68724
350	7	25	7	350	0.0074947	0.00885263	B	1	0.733191	82	41	10.0559	5.20791	5.20791
351	6	12	6	351	0.00144094	0.00117894	B	1	0.734375	89	41	9.43024	5.26077	5.26077
352	7	23	7	352	0.0087684	0.0182218	B	1	0.599345	87	41	9.71542	5.19261	5.19261
353	6	21	6	353	0.00876302	0.018149	B	1	0.599345	86	41	9.66957	5.18591	5.18591
354	10	43	10	354	0.0174243	0.0139631	B	4	0.779741	16	41	10.0978	5.10489	5.10489
355	4	25	4	355	0.0209944	0.0695834	B	1	0.486361	30	41	11.3136	4.71319	4.71319
356	4	19	4	356	0.0129337	0.00669583	B	0	0.945189	91	41	8.62996	4.01943	4.01943
357	1	5	1	357	0.00967045	0.00578187	B	0	1	185	185	7.4	2	2
358	2	9	2	358	0.0048784	0.00461873	B	0	0.930481	78	41	6.1087	1.79842	1.79842
359	2	4	2	359	0.0015625	0.00090629	B	0	0.795699	78	41	4.45533	1.71182	1.71182
360	2	2	2	360	0.00127983	0.000097924	B	1	0.993671	76	41	4.11562	1.71875	1.71875
361	2	10	2	361	0.00458	0.0415205	B	1	0.713402	26	76	4.39706	1.85294	1.85294

جدول ب.۱ ادامه از صفحه ی قبل														
ID	ADEV	COMM	DDEV	FID	NADD	NDEL	B/C	MNR	OWN	EXP	OEXP	NCOMM	NADEV	NDDEV
362	1	7	1	362	0.00561849	0.0235383	B	0	1	41	41	3.30208	1.35156	1.35156
363	2	7	2	363	0.00919168	0.00918133	B	1	0.99285	41	41	3.299	1.3593	1.3593
364	2	4	2	364	0.00485397	0.00944118	B	0	0.883768	41	41	2.65455	1.32121	1.32121
365	2	6	2	365	0.0187539	0.00251186	B	1	0.998517	34	41	2.94269	1.27221	1.27221
366	1	2	1	366	0.00768762	0	B	0	1	6	6	1.24063	1	1
367	0	0	11	367	0	0	B	7	0.356713	38	218	0	0	0
368	0	0	7	368	0	0	B	3	0.385502	132	218	0	0	0
369	1	1	17	369	0.00639024	0.142857	B	5	0.488137	109	41	0	0	0
370	0	0	11	370	0	0	B	3	0.54616	84	218	0	0	0
371	0	0	2	371	0	0	B	1	0.619403	242	41	0	0	0
372	1	5	15	372	0.0250896	0.0322581	B	7	0.537789	54	41	3.5	1	2.5
373	1	1	15	373	0.00241838	0.0046729	B	6	0.363341	78	41	0	0	0
374	0	0	11	374	0	0	B	3	0.659936	83	41	0	0	0
375	1	1	7	375	0.0182768	0	B	4	0.754737	187	41	0	0	0
376	0	0	7	376	0	0	B	4	0.766026	186	41	0	0	0
377	0	0	11	377	0	0	B	8	0.665485	64	41	0	0	0
378	0	0	12	378	0	0	B	7	0.789056	75	41	0	0	0
379	1	2	9	379	0.0218228	0.00564972	B	3	0.630611	213	41	0	0	0
380	1	2	7	380	0.0618812	0.0868056	B	2	0.633978	132	41	2	1	7
381	0	0	7	381	0	0	B	4	0.748971	106	41	0	0	0
382	1	2	15	382	0.0202224	0.0157233	B	7	0.397117	27	41	1.5	1	10
383	1	1	11	383	0.0136494	0.00635593	B	4	0.593103	84	41	2	1	15
384	0	0	2	384	0	0	B	0	0.946721	208	84	0	0	0
385	1	4	16	385	0.0270068	0.00579151	B	3	0.563945	85	41	1.5	1	7.5
386	0	0	5	386	0	0	B	4	0.922078	158	41	0	0	0
387	1	1	11	387	0.00717781	0.0204662	B	2	0.514056	122	84	1.38462	1	8.46154
388	0	0	10	388	0	0	B	3	0.604507	72	41	0	0	0
389	0	0	4	389	0	0	B	0	0.598131	438	218	0	0	0
390	1	1	7	390	0.000476644	0.00109051	B	1	0.723426	242	41	2.58333	1.66667	8.75
391	0	0	7	391	0	0	B	1	0.495098	242	520	0	0	0
392	1	1	11	392	0.000491884	0.000741015	B	4	0.455652	73	218	2.5	1	8
393	5	7	16	393	0.0215264	0.0119115	B	10	0.703355	40	41	3.33333	2.16667	10
394	0	0	9	394	0	0	B	3	0.693878	101	41	0	0	0
395	0	0	9	395	0	0	B	2	0.638003	91	41	0	0	0
396	0	0	7	396	0	0	B	3	0.878655	203	41	0	0	0
397	0	0	6	397	0	0	B	4	0.852006	8	337	0	0	0
398	2	7	9	398	0.0760083	0.114035	B	2	0.520985	88	41	3.70874	1.01942	7.66019
399	0	0	3	399	0	0	B	2	0.988327	129	41	0	0	0
400	0	0	4	400	0	0	B	1	0.498715	127	206	0	0	0
401	6	16	6	401	0.00591586	0.00266591	B	2	0.956573	99	41	10.5956	5.74142	5.74142
402	6	18	6	402	0.00549786	0.00481687	B	1	0.735537	29	41	9.7781	5.16824	5.16824
403	1	3	1	403	0.00909206	0.00229533	B	0	1	41	41	2.95751	1.31445	1.31445
404	1	1	1	404	0.0105867	0	B	0	1	6	6	1.15203	1	1
405	0	0	6	405	0	0	B	2	0.413428	101	218	0	0	0
732	0	0	7	327	0	0	C	3	0.917355	70	41	0	0	0
710	0	0	1	305	0	0	C	0	1	463	463	0	0	0
575	4	15	4	170	0.00503949	0	C	0	1	553	553	23.6093	4.60479	4.60479
533	2	8	6	128	0.0142938	0.0331092	C	1	0.682081	325	42	4.54475	2.06615	6.28016
431	3	4	4	26	0.00076929	0	C	0	1	19	19	4.87293	3.20442	5.96685
679	0	0	7	274	0	0	C	1	0.792453	223	41	0	0	0
768	2	3	2	363	0.00322388	0.000510074	C	1	0.983051	41	41	2.90169	1.33146	1.33146
738	0	0	9	333	0	0	C	4	0.890274	71	41	0	0	0
627	1	6	1	222	0.000871308	0.00107828	C	0	1	1069	1069	9.62791	1.13953	1.4186
711	0	0	4	306	0	0	C	3	0.924051	193	41	0	0	0
795	0	0	7	390	0	0	C	1	0.854033	242	41	0	0	0
534	0	0	5	129	0	0	C	2	0.942308	355	935	0	0	0
750	2	3	2	345	0.000913	0.0000763213	C	1	0.979021	91	41	10.3583	5.10556	5.10556
801	0	0	6	396	0	0	C	1	0.940426	347	214	0	0	0
529	1	1	5	124	0.0000010795	0.00000184476	C	0	0.692308	989	953	96.4871	4.65517	8.23707
438	4	13	4	33	0.00194188	0.00364721	C	3	0.962766	305	86	13.5336	4.05601	4.05601
439	5	26	5	34	0.0569146	0.0044853	C	9	0.503768	58	568	12.3068	4.0454	4.0454
440	1	1	1	35	0.0000213904	0.0000611658	C	5	0.523077	91	994	1.725	1.125	1.125
441	1	1	1	36	0.0000217127	0.0000631034	C	0	0.647059	742	568	1.3625	1.0875	1.0875

فصل ب: معیارهای استخراج شده

جدول ب.۱ ادامه از صفحه قبل

ID	ADEV	COMM	DDEV	FID	NADD	NDEL	B/C	MNR	OWN	EXP	OEXP	NCOMM	NADEV	NDDEV
442	2	2	2	37	0.00026087	0.000569044	C	0	0.842424	419	568	1.2625	1.05	1.05
443	1	1	1	38	0.0000217538	0.0000632631	C	4	0.772589	145	360	1.225	1.0625	1.0625
444	1	1	1	39	0.0000218713	0.0000633794	C	1	0.608541	119	568	1.1125	1	1
445	2	8	7	40	0.00989219	0.0285863	C	2	0.597938	281	540	12.9632	2.20588	7.03676
446	2	13	8	41	0.00966557	0.0268874	C	3	0.451444	131	360	13.4186	2.21336	7.01954
447	1	6	1	42	0.0906443	0.00144509	C	0	1	72	72	8.8125	1.9375	7.34375
448	0	0	5	43	0	0	C	3	0.941176	145	360	0	0	0
449	0	0	9	44	0	0	C	3	0.582755	58	540	0	0	0
450	0	0	5	45	0	0	C	3	0.959184	259	540	0	0	0
451	0	0	7	46	0	0	C	3	0.814709	99	540	0	0	0
452	1	2	5	47	0.00506235	0	C	0	0.605769	107	540	4.725	1.275	5.9875
453	2	4	2	48	0.00149103	0.000311485	C	1	0.994048	137	27	3.33333	1.875	2.04167
454	1	2	7	49	0.00431677	0	C	3	0.751908	100	540	4.3625	1.2	5.9375
455	1	5	8	50	0.0211143	0.0114979	C	3	0.39798	79	347	4.2132	1.18782	6.06091
456	1	5	5	51	0.00202501	0.00540153	C	2	0.777273	138	651	4.0273	1.11945	5.71331
457	1	2	4	52	0.00457124	0	C	0	0.75	495	540	3.9125	1.15	5.95
458	1	2	5	53	0.00176973	0	C	2	0.87619	244	640	3.65	1.0125	5.9125
459	7	69	7	54	0.00638861	0.00640841	C	3	0.731013	74	540	36.4835	5.35816	5.35816
460	5	14	5	55	0.000957779	0.00142777	C	2	0.46875	96	540	27.9641	5.15011	5.15011
461	5	17	5	56	0.00252332	0.00181908	C	1	0.823729	123	527	28.2697	5.20866	5.20866
462	6	67	6	57	0.0793313	0.0900177	C	2	0.542676	195	527	32.2259	5.11293	5.11293
463	4	11	4	58	0.000904668	0.000524708	C	1	0.692982	123	527	27.4434	5.1682	5.1682
464	6	26	6	59	0.00475064	0.00297941	C	4	0.803082	90	527	31.8592	5.3835	5.3835
465	5	32	5	60	0.00230623	0.00279558	C	1	0.668367	122	527	31.7725	5.1455	5.1455
466	5	22	5	61	0.00807497	0.00585382	C	2	0.580234	84	527	27.9739	5.16387	5.16387
467	4	21	4	62	0.00180472	0.00332082	C	1	0.55	121	430	31.275	5.06977	5.06977
468	8	43	8	63	0.00526458	0.00703202	C	4	0.405941	60	421	31.2086	5.11734	5.11734
469	6	17	6	64	0.00275201	0.002874	C	3	0.498099	86	526	27.7255	5.33152	5.33152
470	5	38	5	65	0.0141387	0.0270203	C	2	0.768678	113	511	29.0348	5.0443	5.0443
471	0	0	1	66	0	0	C	0	1	863	863	0	0	0
472	0	0	2	67	0	0	C	0	0.933333	443	843	0	0	0
473	0	0	1	68	0	0	C	0	1	826	826	0	0	0
474	0	0	1	69	0	0	C	0	1	810	810	0	0	0
475	0	0	3	70	0	0	C	1	0.507692	1118	795	0	0	0
476	0	0	1	71	0	0	C	0	1	792	792	0	0	0
477	0	0	2	72	0	0	C	0	0.604343	550	792	0	0	0
478	0	0	2	73	0	0	C	0	0.907975	376	792	0	0	0
479	0	0	1	74	0	0	C	0	1	792	792	0	0	0
480	0	0	1	75	0	0	C	0	1	792	792	0	0	0
481	0	0	1	76	0	0	C	0	1	788	788	0	0	0
482	0	0	1	77	0	0	C	0	1	775	775	0	0	0
483	0	0	1	78	0	0	C	0	1	770	770	0	0	0
484	0	0	2	79	0	0	C	1	0.982143	942	769	0	0	0
485	0	0	1	80	0	0	C	0	1	715	715	0	0	0
486	0	0	1	81	0	0	C	0	1	712	712	0	0	0
487	1	2	2	82	0.00449953	0.00625496	C	0	0.934045	475	654	4.5	1	2
488	1	1	2	83	0.0000457122	0.000110156	C	1	0.970588	298	630	2.5	1.81818	2.68182
489	2	4	3	84	0.0119548	0.0286517	C	0	0.886148	940	624	3.31737	2.34132	3.2994
490	1	1	2	85	0.000607448	0.000898977	C	0	0.939535	835	623	1.28	1.12	2.28
491	2	2	3	86	0.000106525	0.00026476	C	0	1	620	620	3.2	2.39286	3.46429
492	0	0	1	87	0	0	C	0	1	619	619	0	0	0
493	2	2	3	88	0.0151831	0	C	1	0.996078	408	1392	2.87097	2.22581	3.09677
494	2	2	3	89	0.00111193	0.00150426	C	1	0.666667	464	613	1.94521	1.94521	2.79452
495	0	0	1	90	0	0	C	0	1	600	600	0	0	0
496	0	0	1	91	0	0	C	0	1	581	581	0	0	0
497	0	0	1	92	0	0	C	0	1	580	580	0	0	0
421	1	1	7	16	0.001554	0	C	3	0.673077	174	612	1	1	7
630	3	17	4	225	0.0564005	0.0801527	C	3	0.964286	70	1233	5.77477	1.65766	2.11712
719	0	0	5	314	0	0	C	1	0.986072	280	212	0	0	0
720	0	0	1	315	0	0	C	0	1	363	363	0	0	0
721	0	0	6	316	0	0	C	2	0.989933	146	41	0	0	0
722	2	4	8	317	0.00125713	0.0207668	C	2	0.638191	54	41	7.65116	3.04651	7.83721
723	1	5	9	318	0.000663051	0.00691986	C	2	0.731475	92	41	4.84211	2.26316	7.31579

جدول ب.۱. ادامه از صفحه ی قبل															
ID	ADEV	COMM	DDEV	FID	NADD	NDEL	B/C	MNR	OWN	EXP	OEXP	NCOMM	NADEV	NDDEV	
805	0	0	5	400	0	0	C	0	0.68	91	41	0	0	0	
797	0	0	7	392	0	0	C	1	0.875	251	41	0	0	0	
791	0	0	8	386	0	0	C	1	0.445483	106	41	0	0	0	
788	0	0	1	383	0	0	C	0	1	260	260	0	0	0	
786	0	0	6	381	0	0	C	5	0.944	157	41	0	0	0	
677	0	0	8	272	0	0	C	2	0.868421	64	41	0	0	0	
781	1	1	15	376	0.00276243	0.00356506	C	10	0.261168	22	744	2.33333	1	9.66667	
606	0	0	1	201	0	0	C	0	1	1233	1233	0	0	0	
798	0	0	3	393	0	0	C	1	0.910569	194	19	0	0	0	
543	3	3	4	138	0.000599692	0.000915658	C	1	0.985714	837	808	2.74091	2.732	6.43281	
527	6	280	8	122	0.0043037	0.00740908	C	1	0.559172	111	1113	238.255	5.18359	8.201	
521	1	4	5	116	0.00400932	0.000162837	C	1	0.886029	221	30	4.77117	1.91393	7.42547	
659	1	1	2	254	0.0000294911	0.00102276	C	1	0.996344	652	1081	2.13462	1.01923	1.92308	
423	5	8	8	18	0.0229765	0.00828889	C	6	0.886236	93	611	7.97925	3.57054	6.84855	
412	2	3	8	7	0.00246491	0.00322234	C	3	0.549645	223	220	3.59048	1.73333	6.78095	
667	0	0	2	262	0	0	C	1	0.986842	601	727	0	0	0	
612	0	0	3	207	0	0	C	0	0.851852	188	1623	0	0	0	
562	4	15	7	157	0.00825972	0.00470303	C	2	0.687395	135	191	13.4733	3.37213	5.51742	
570	1	1	6	165	0.000162188	0.0117188	C	3	0.685606	106	80	1.2	1	6.4	
668	0	0	7	263	0	0	C	2	0.782609	237	41	0	0	0	
634	2	5	6	229	0.00344664	0.00582263	C	1	0.980296	881	371	4.50812	1.48144	3.34841	
635	1	1	3	230	0.00266312	0.00884956	C	1	0.991416	178	1449	0	0	0	
636	0	0	2	231	0	0	C	0	0.797753	723	1327	0	0	0	
599	1	3	2	194	0.000680257	0	C	0	1	1889	1889	3.02515	1.00387	3.22631	
426	1	1	4	21	0.0011469	0	C	0	1	28	28	1.32292	1.01042	5.95833	
745	0	0	1	340	0	0	C	0	1	197	197	0	0	0	
409	2	4	8	4	0.00929768	0.0100978	C	4	0.451941	134	612	5.96073	2.43456	6.98168	
712	0	0	8	307	0	0	C	2	0.515707	222	337	0	0	0	
545	2	6	2	140	0.000166582	0.0000561188	C	1	0.984127	830	803	20.2163	3.62982	4.89185	
546	5	41	5	141	0.000705537	0.00129131	C	0	0.927273	267	89	22.173	4.17943	5.8932	
547	5	28	6	142	0.00219186	0.00405733	C	0	0.697143	824	855	22.3005	4.15967	5.89665	
548	3	10	3	143	0.000934597	0.000115009	C	1	0.991903	190	840	20.2605	3.70871	4.94928	
549	3	10	3	144	0.000653973	0.000904177	C	1	0.973822	814	835	20.1526	3.68063	4.92621	
550	3	14	3	145	0.000908673	0.000423403	C	1	0.919118	316	833	21.0858	3.82817	5.19638	
551	3	9	3	146	0.000379052	0.000192478	C	1	0.988764	810	832	20.0951	3.68017	4.93027	
552	2	5	2	147	0.000285131	0.0000225296	C	1	0.990741	780	775	18.4006	3.63051	4.89435	
553	3	11	3	148	0.000766518	0.000135236	C	1	0.952756	780	786	19.6689	3.8418	5.22924	
554	1	5	1	149	0.000506406	0.0000450785	C	0	1	785	785	18.455	3.56635	4.88863	
555	4	22	10	150	0.00251631	0.000575665	C	2	0.57561	148	191	18.8375	4.12773	5.96874	
556	6	29	9	151	0.00274322	0.00230731	C	5	0.584475	101	726	17.5689	4.35255	6.28658	
557	4	12	4	152	0.000661007	0.00112928	C	0	0.88	683	726	17.0778	4.25876	6.03238	
558	4	13	4	153	0.00316426	0.00014742	C	0	1	637	637	17.2161	4.21285	5.94508	
559	5	21	6	154	0.000684104	0.000211383	C	1	0.959184	633	562	16.7869	4.30509	6.23909	
530	3	18	3	125	0.00572897	0.000031755	C	1	0.963801	350	120	42.6531	3.28083	6.2118	
637	0	0	1	232	0	0	C	0	1	1326	1326	0	0	0	
623	0	0	1	218	0	0	C	0	1	1252	1252	0	0	0	
614	0	0	2	209	0	0	C	0	1	1616	1616	0	0	0	
615	2	3	2	210	0.00101112	0.000379399	C	0	1	1590	1590	7.86598	2.21134	3.39691	
616	1	1	1	211	0.000129299	0	C	0	1	1499	1499	2.44882	1.43307	1.8622	
560	3	13	3	155	0.00198553	0	C	1	0.951389	633	562	16.1856	4.11056	6.1899	
769	1	1	1	364	0.00039846	0	C	0	1	41	41	2.40203	1.25338	1.25338	
747	0	0	7	342	0	0	C	1	0.752396	120	41	0	0	0	
758	5	5	5	353	0.000263946	0.000176204	C	0	1	41	41	9.0098	5.18791	5.18791	
619	0	0	2	214	0	0	C	0	0.93617	178	1449	0	0	0	
794	0	0	10	389	0	0	C	4	0.822086	114	41	0	0	0	
756	2	5	2	351	0.00359248	0.000654965	C	0	0.932203	91	41	9.10644	4.38366	4.38366	
749	6	9	6	344	0.00459246	0.000456638	C	1	0.998609	78	41	10.9054	5.90992	5.90992	
726	3	3	9	321	0.000321793	0.00181441	C	2	0.436831	65	260	1.8	1.6	6.6	
539	0	0	5	134	0	0	C	1	0.988764	881	905	0	0	0	
418	0	0	7	13	0	0	C	1	0.971429	311	69	0	0	0	
620	0	0	2	215	0	0	C	0	0.8	178	1449	0	0	0	
757	2	3	2	352	0.000553794	0.000572183	C	1	0.640693	87	41	33	6.66667	6.66667	
655	0	0	2	250	0	0	C	0	0.634021	679	394	0	0	0	

جدول ب.۱ ادامه از صفحه قبل

ID	ADEV	COMM	DDEV	FID	NADD	NDEL	B/C	MNR	OWN	EXP	OEXP	NCOMM	NADEV	NDDEV
656	0	0	2	251	0	0	C	0	1	1165	1165	0	0	0
419	0	0	7	14	0	0	C	2	0.528497	497	503	0	0	0
608	1	4	5	203	0.000927286	0	C	0	0.632353	0	0	4.45412	1.43383	3.29767
695	0	0	8	290	0	0	C	4	0.913249	72	41	0	0	0
688	1	1	9	283	0.00804906	0.0152727	C	5	0.324574	23	41	1.7451	1.3268	6.40523
803	0	0	1	398	0	0	C	1	0.693878	165	41	0	0	0
680	1	1	3	275	0.00052521	0.00104548	C	1	0.569672	177	20	2.03268	1.36601	6.44444
672	0	0	1	267	0	0	C	0	0.930233	168	41	0	0	0
414	1	2	7	9	0.00366827	0	C	2	0.887059	293	1410	3.14286	1.69524	6.77143
583	0	0	2	178	0	0	C	0	0.928241	1090	792	0	0	0
669	0	0	2	264	0	0	C	0	0.892857	65	6	0	0	0
762	1	1	1	357	0.00462368	0	C	0	1	41	41	4.7973	1.85473	1.85473
661	1	1	2	256	0.0000590423	0.00205814	C	1	0.996951	649	1070	2	1.01923	1.92308
751	1	2	1	346	0.000215475	0.000038506	C	0	1	138	138	9.51013	4.97973	4.97973
718	0	0	2	313	0	0	C	1	0.636364	149	41	0	0	0
689	1	1	2	284	0.000496278	0.000903342	C	0	0.934066	498	598	1.60131	1.30065	6.44444
673	0	0	7	268	0	0	C	1	0.661692	228	41	0	0	0
785	1	1	15	380	0.0123762	0.00694444	C	10	0.264605	22	744	0	0	0
415	2	3	7	10	0.00339638	0	C	2	0.976982	373	69	3.23887	1.76113	6.75708
713	0	0	2	308	0	0	C	1	0.983607	135	41	0	0	0
564	1	2	1	159	0.000301022	0.000121065	C	0	1	460	460	15.8077	2.84615	5.03846
541	3	3	6	136	0.0013206	0.00483975	C	1	0.584158	381	898	2.99852	2.85672	6.51373
504	1	1	1	99	0.0001779	0	C	0	1	525	525	1.16558	1.07143	1.07792
434	2	3	6	29	0.0199815	0	C	4	0.491747	120	1046	3.1	2.02222	5.84444
657	1	1	2	252	0.0000147361	0.000509424	C	0	0.75	653	1085	2.19231	1.01923	1.92308
649	0	0	2	244	0	0	C	0	0.823529	689	1205	0	0	0
643	0	0	4	238	0	0	C	2	0.940828	158	1268	0	0	0
644	0	0	2	239	0	0	C	0	0.821429	706	394	0	0	0
645	0	0	2	240	0	0	C	1	0.996344	705	1264	0	0	0
610	1	2	2	205	0.000112356	0	C	0	1	0	0	2.07744	1.04589	3.17017
601	1	4	2	196	0.00106724	0.00152486	C	0	0.857143	173	1372	6.15484	1.36774	2.12903
596	1	2	4	191	0.00106535	0	C	0	1	2067	2067	2.16551	1.0993	3.11237
584	0	0	1	179	0	0	C	0	1	792	792	0	0	0
585	0	0	4	180	0	0	C	2	0.674757	536	169	0	0	0
586	1	1	2	181	0.0000580686	0.000151837	C	1	0.970588	272	619	1.81818	1.81818	2.68182
572	2	3	3	167	0.00104166	0	C	0	0.938776	270	620	4.24797	1.89455	4.79143
565	1	1	1	160	0.000870713	0	C	0	1	336	336	0	0	0
566	2	5	2	161	0.000855474	0.005428	C	0	0.681529	463	667	9.30571	1.94286	4.70952
542	3	3	7	137	0.000511063	0.000559483	C	0	0.890625	837	808	2.74313	2.73422	6.42613
536	2	2	4	131	0.00298418	0.0124417	C	1	0.84874	278	918	1.67857	1.27381	6.57143
775	0	0	15	370	0	0	C	6	0.388932	52	41	0	0	0
772	0	0	1	367	0	0	C	0	1	502	502	0	0	0
505	1	1	1	100	0.000558228	0	C	0	1	524	524	1.14935	1.06656	1.07305
506	2	5	5	101	0.000299211	0.0000865862	C	1	0.970588	789	471	8.66784	3.25377	7.81156
507	3	6	9	102	0.0000225877	0.0000482785	C	1	0.660714	869	1323	8.27262	3.30343	8.13924
508	2	4	5	103	0.00122387	0.00000814438	C	2	0.962617	259	30	8.27388	3.17787	7.74634
509	3	14	8	104	0.000456116	0.00100176	C	3	0.980535	442	1321	8.32833	3.26517	8.21551
510	2	4	2	105	0.00199442	0.000257492	C	0	1	433	433	7.87961	3.14675	7.442
511	4	10	6	106	0.00150314	0.000116889	C	1	0.784173	471	426	8.04499	3.13159	7.64683
512	3	7	7	107	0.000196106	0.000384236	C	1	0.527094	472	1096	7.90197	3.24161	8.06638
513	2	4	5	108	0.000305066	0.000075582	C	1	0.970588	743	422	7.8095	3.1622	7.73067
514	3	8	9	109	0.000138385	0.000315339	C	2	0.964286	822	1297	7.51664	3.10797	8.15474
515	3	6	7	110	0.0000405439	0.0000443486	C	2	0.951807	788	1295	7.21734	3.08593	7.98678
516	3	5	14	111	0.0000233197	0.0000526	C	2	0.457143	173	1049	6.53729	2.92971	7.88247
517	2	6	5	112	0.00167912	0.000031562	C	0	0.932773	643	333	6.58963	2.82981	7.5136
518	3	10	9	113	0.000662926	0.00126516	C	2	0.690141	175	1242	6.61221	2.9302	8.05196
519	2	4	12	114	0.000137112	0.000226879	C	5	0.62381	236	1038	4.66159	1.99556	7.63365
520	1	5	1	115	0.00126484	0.00268345	C	1	0.953333	604	1234	4.6068	1.87446	6.85092
416	2	2	8	11	0.00222076	0	C	3	0.671937	316	23	2.03521	1.77465	6.72535
735	0	0	5	330	0	0	C	1	0.974359	92	41	0	0	0
631	2	5	5	226	0.000187503	0.000321854	C	0	1	371	371	4.50812	1.48144	3.35035
617	0	0	1	212	0	0	C	0	1	1461	1461	0	0	0
714	0	0	2	309	0	0	C	1	0.996795	62	433	0	0	0

جدول ب.۱ ادامه از صفحه ی قبل														
ID	ADEV	COMM	DDEV	FID	NADD	NDEL	B/C	MNR	OWN	EXP	OEXP	NCOMM	NADEV	NDDEV
602	1	4	2	197	0.00110788	0.00164042	C	0	0.851852	173	1361	6.11613	1.36774	2.12903
810	1	1	1	405	0.00276221	0	C	0	1	428	428	1	1	1
573	2	7	2	168	0.000926111	0.000400169	C	1	0.994253	178	601	26.7508	4.73406	4.73406
537	0	0	7	132	0	0	C	2	0.776699	252	917	0	0	0
670	0	0	1	265	0	0	C	0	1	7	7	0	0	0
763	1	1	1	358	0.00100438	0	C	0	1	41	41	4.17905	1.69595	1.69595
429	4	6	6	24	0.0109467	0	C	4	0.680851	163	382	7.15761	4.11775	5.97645
654	0	0	2	249	0	0	C	1	0.977064	679	1173	0	0	0
589	1	5	10	184	0.00565181	0	C	3	0.601173	104	191	5.66622	1.60268	5.08562
502	1	1	1	97	0.000361791	0	C	0	1	539	539	1.23052	1.11039	1.11688
621	0	0	1	216	0	0	C	0	1	1423	1423	0	0	0
796	0	0	3	391	0	0	C	1	0.833333	70	33	0	0	0
676	0	0	6	271	0	0	C	1	0.938298	398	215	0	0	0
535	1	2	4	130	0.000364653	0.000812018	C	1	0.97561	902	935	2.71472	1.24233	6.6227
433	2	3	7	28	0.0115307	0	C	4	0.463511	195	12	3.13333	2.02222	5.83333
420	0	0	6	15	0	0	C	2	0.977528	560	1396	0	0	0
595	1	1	5	190	0.000988487	0	C	0	0.632353	0	0	1.06186	1.06014	3.13058
733	1	1	1	328	0.000268504	0.000405022	C	0	1	54	54	0	0	0
808	1	3	1	403	0.00267147	0.00153022	C	0	1	41	41	2.64537	1.28435	1.28435
746	0	0	5	341	0	0	C	0	1	41	41	0	0	0
740	0	0	8	335	0	0	C	3	0.790074	90	41	0	0	0
728	1	1	1	323	0.00332813	0	C	0	1	7	7	2	1.5	1.5
605	1	1	3	200	0.00125431	0.00063012	C	0	0.944444	157	1238	4.78947	1.38596	2.14035
706	0	0	6	301	0	0	C	2	0.988095	162	41	0	0	0
704	0	0	2	299	0	0	C	0	0.933333	84	19	0	0	0
809	1	1	1	404	0.00105967	0	C	0	1	6	6	1.15203	1	1
581	0	0	1	176	0	0	C	0	1	792	792	0	0	0
674	0	0	2	269	0	0	C	0	0.934066	552	676	0	0	0
526	1	2	14	121	0.000163049	0.000287829	C	1	0.477477	89	191	1.90145	1.01616	7.82229
435	6	24	6	30	0.000763322	0.0000534074	C	2	0.950617	479	1016	26.7179	6.03375	6.03375
436	6	23	6	31	0.000876161	0.0000537317	C	2	0.956989	479	1016	25.6177	6.03897	6.03897
804	0	0	2	399	0	0	C	0	0.815126	39	6	0	0	0
780	0	0	4	375	0	0	C	2	0.819672	224	41	0	0	0
498	2	2	3	93	0.00181042	0.00380125	C	0	1	580	580	2.42857	2.14286	3.33571
638	0	0	2	233	0	0	C	0	0.68693	722	394	0	0	0
708	0	0	4	303	0	0	C	2	0.773723	199	41	0	0	0
531	2	3	5	126	0.000133136	0.0000320015	C	1	0.981132	231	976	7.41935	2.28629	7.33669
588	3	22	3	183	0.00097634	0.00127731	C	2	0.954198	326	855	21.4801	3.84243	5.30657
569	3	12	4	164	0.00058156	0	C	1	0.951219	591	705	14.8498	4.02762	6.17044
501	1	2	1	96	0.000427233	0.0000624064	C	0	1	540	540	1.23871	1.1129	1.11935
425	2	5	7	20	0.00267546	0.00242588	C	3	0.65625	133	609	7.46403	3.36331	7.1223
408	2	3	7	3	0.000746863	0.000158479	C	3	0.966292	575	1425	5.78095	2.39048	6.94286
578	0	0	1	173	0	0	C	0	1	223	223	0	0	0
766	1	1	1	361	0.000637834	0	C	0	1	41	41	2.77365	1.42905	1.42905
428	4	9	6	23	0.00186439	0	C	3	0.536458	416	382	10.2536	4.12738	5.9631
639	0	0	2	234	0	0	C	0	0.509434	722	1324	0	0	0
730	0	0	6	325	0	0	C	2	0.971591	136	41	0	0	0
724	1	1	8	319	0.0000227931	0.000373204	C	2	0.486641	96	41	4.77778	2.22222	6.77778
592	1	1	18	187	0.0125559	0	C	12	0.554753	0	2095	1.00993	1.00662	3.24172
792	1	1	4	387	0.000326264	0.000568505	C	3	0.976351	80	84	1.36923	1.08462	7.51538
633	2	5	6	228	0.00229094	0.00052667	C	1	0.748466	257	2095	4.4606	1.43713	3.31224
641	0	0	1	236	0	0	C	0	1	1285	1285	0	0	0
807	2	11	2	402	0.00441405	0.00256032	C	0	0.840131	91	41	11.0354	4.60904	4.60904
563	3	10	3	158	0.00336283	0.00840237	C	0	0.52766	569	689	13.3557	3.32672	5.24224
540	0	0	4	135	0	0	C	1	0.990741	878	853	0	0	0
777	1	3	14	372	0.00955794	0.00614439	C	3	0.469204	92	41	4	1	7.75
778	0	0	6	373	0	0	C	1	0.858065	188	41	0	0	0
779	0	0	4	374	0	0	C	3	0.990491	41	8	0	0	0
664	0	0	1	259	0	0	C	0	0.748299	722	394	0	0	0
739	0	0	7	334	0	0	C	2	0.885417	140	41	0	0	0
590	1	1	4	185	0.00051382	0	C	0	1	382	382	1.00828	1.00662	3.2649
790	1	1	9	385	0.00425106	0	C	4	0.678153	196	41	1	1	8
765	1	1	1	360	0.000209281	0	C	0	1	41	41	4.0777	1.69595	1.69595

جدول ب.۱ ادامه از صفحه قبل

ID	ADEV	COMM	DDEV	FID	NADD	NDEL	B/C	MNR	OWN	EXP	OEXP	NCOMM	NADEV	NDDEV
571	0	0	5	166	0	0	C	2	0.482759	103	191	0	0	0
427	1	1	7	22	0.00517627	0	C	5	0.677083	80	568	1.02083	1	5.92708
629	0	0	1	224	0	0	C	0	1	1238	1238	0	0	0
681	0	0	1	276	0	0	C	2	0.840555	222	41	0	0	0
734	0	0	5	329	0	0	C	0	1	41	41	0	0	0
690	0	0	6	285	0	0	C	1	0.973684	155	41	0	0	0
742	1	1	2	337	0.000144279	0.000264201	C	1	0.973333	210	206	1	1	3.58621
683	0	0	1	278	0	0	C	0	1	5	5	0	0	0
773	0	0	6	368	0	0	C	1	0.966667	174	41	0	0	0
783	0	0	11	378	0	0	C	3	0.529661	123	41	0	0	0
647	0	0	2	242	0	0	C	0	0.947368	694	1224	0	0	0
594	1	3	4	189	0.000247471	0	C	0	1	2095	2095	3.03072	1.02389	3.13481
685	2	2	2	280	0.00167131	0.00552181	C	1	0.952775	522	627	2.10215	1.41398	6.51613
642	0	0	2	237	0	0	C	1	0.99115	709	1278	0	0	0
600	1	1	3	195	0.000715249	0	C	0	0.477273	224	1845	1	1	2.23288
693	0	0	7	288	0	0	C	2	0.729059	214	41	0	0	0
591	1	2	4	186	0.000156154	0	C	0	0.7	214	2095	2.01874	1.01363	3.15162
686	1	2	6	281	0.00337079	0.00227015	C	3	0.759184	83	41	1.96154	1.35897	6.42308
682	2	3	8	277	0.00713719	0.0241935	C	1	0.784483	222	41	0	0	0
499	1	1	1	94	0.000104094	0	C	0	1	545	545	1.31494	1.13312	1.13961
665	0	0	2	260	0	0	C	0	0.915663	694	1224	0	0	0
662	0	0	2	257	0	0	C	1	0.984615	645	1056	0	0	0
658	1	7	2	253	0.00045686	0.00280971	C	1	0.996833	653	1084	3.24444	1.01111	1.81111
624	2	5	3	219	0.00850229	0.0384047	C	1	0.972222	96	1169	3.02174	1.34783	2.07609
579	0	0	5	174	0	0	C	3	0.556962	749	383	0	0	0
567	1	6	3	162	0.00930064	0	C	1	0.989381	215	664	6.35552	1.60823	5.01028
716	0	0	2	311	0	0	C	0	0.75	301	214	0	0	0
598	1	3	4	193	0.000462476	0	C	0	0.862069	751	1984	3.09149	1.04892	3.16264
523	1	2	5	118	0.00133913	0.0000340437	C	1	0.987654	568	265	3.02002	1.1437	6.95524
717	0	0	6	312	0	0	C	3	0.934783	43	41	0	0	0
761	4	5	4	356	0.000928355	0.00037799	C	1	0.96	91	41	7.82336	4.09814	4.09814
727	1	1	2	322	0.000125584	0.000260824	C	0	0.902299	254	209	0	0	0
692	0	0	2	287	0	0	C	0	0.896552	355	215	0	0	0
525	1	2	15	120	0.00461184	0.0081391	C	4	0.257576	138	191	1.90469	1.01939	7.81906
753	7	8	7	348	0.000774948	0.00031015	C	3	0.912281	32	41	10.6054	5.98802	5.98802
411	1	2	7	6	0.00505574	0	C	5	0.902521	324	220	3.64762	1.74286	6.79048
806	5	5	5	401	0.00396324	0.00016157	C	0	1	41	41	10.2606	5.9134	5.9134
802	0	0	3	397	0	0	C	1	0.995475	47	448	0	0	0
800	1	1	2	395	0.000389257	0.00175131	C	0	0.860544	340	215	1	1	1
799	1	1	1	394	0.0110442	0	C	0	1	561	561	3.25	1	11
793	0	0	7	388	0	0	C	1	0.769759	121	41	0	0	0
789	0	0	7	384	0	0	C	1	0.869565	242	41	0	0	0
787	0	0	1	382	0	0	C	0	1	275	275	0	0	0
782	0	0	11	377	0	0	C	3	0.331897	86	520	0	0	0
776	0	0	5	371	0	0	C	2	0.777778	229	41	0	0	0
774	1	1	1	369	0.00181541	0	C	0	1	455	455	1	1	1
771	1	2	1	366	0.000989693	0.0030722	C	0	1	6	6	1.18212	1	1
767	2	2	2	362	0.00119235	0.002531	C	0	0.852459	42	41	2.49333	1.26667	1.26667
764	2	3	2	359	0.00133131	0.000271887	C	1	0.981707	78	41	4.60665	1.72853	1.72853
759	2	4	2	354	0.0011105	0.00030931	C	1	0.968944	91	41	8.89894	4.31915	4.31915
755	5	8	5	350	0.00203625	0.00264711	C	0	0.722892	91	41	9.31205	5.28537	5.28537
748	4	19	4	343	0.00198659	0.00132863	C	1	0.680851	112	41	12.2975	5.34815	5.34815
744	0	0	3	339	0	0	C	2	0.989775	119	41	0	0	0
737	0	0	6	332	0	0	C	1	0.985915	102	41	0	0	0
576	4	31	4	171	0.022675	0.0395455	C	1	0.703338	84	65	26.5569	4.833	4.833
604	0	0	2	199	0	0	C	0	0.851852	178	1449	0	0	0
410	2	4	2	5	0.00189333	0.00416522	C	1	0.985849	752	1410	3.85581	1.79535	6.77674
577	0	0	2	172	0	0	C	1	0.972477	1165	863	0	0	0
752	5	7	5	347	0.000603136	0.000774293	C	0	0.68	91	41	10.7327	5.97997	5.97997
741	0	0	6	336	0	0	C	1	0.978723	94	41	0	0	0
729	0	0	5	324	0	0	C	0	1	41	41	0	0	0
770	1	3	1	365	0.00272263	0.00167457	C	0	1	41	41	2.50479	1.23323	1.23323
736	0	0	5	331	0	0	C	1	0.979381	92	41	0	0	0

جدول ب.۱. ادامه از صفحه ی قبل														
ID	ADEV	COMM	DDEV	FID	NADD	NDEL	B/C	MNR	OWN	EXP	OEXP	NCOMM	NADEV	NDDEV
731	1	3	9	326	0.00529468	0.009319	C	3	0.741864	62	41	1.66667	1.66667	5.33333
725	1	1	6	320	0.000124851	0.000229621	C	0	0.87218	114	41	4.77778	2	5.44444
715	1	1	9	310	0.0136652	0.00290698	C	4	0.817757	41	41	0	0	0
709	1	1	7	304	0.000627156	0.00158061	C	1	0.792453	196	41	2.43478	1.14493	6.91304
707	0	0	6	302	0	0	C	1	0.898734	162	41	0	0	0
705	0	0	2	300	0	0	C	1	0.993151	327	214	0	0	0
694	0	0	2	289	0	0	C	0	0.94702	45	5	0	0	0
691	0	0	6	286	0	0	C	1	0.674242	173	41	0	0	0
687	1	1	6	282	0.00114976	0.00237671	C	1	0.640884	107	41	1.89542	1.34641	6.42484
684	1	1	7	279	0.00026469	0.000538213	C	0	0.681481	222	41	1.99346	1.35948	6.4183
678	0	0	9	273	0	0	C	4	0.675127	114	41	0	0	0
675	1	1	3	270	0.000786163	0.00205761	C	1	0.773381	133	8	1.15385	1.15385	7.61538
671	4	6	10	266	0.00847243	0.0045301	C	4	0.493976	33	41	6.6	2.98182	9.76364
666	0	0	6	261	0	0	C	4	0.850746	92	217	0	0	0
663	0	0	1	258	0	0	C	0	1	1327	1327	0	0	0
660	0	0	2	255	0	0	C	1	0.977064	649	1071	0	0	0
651	0	0	2	246	0	0	C	0	0.640625	687	1201	0	0	0
652	0	0	2	247	0	0	C	1	0.997093	683	1186	0	0	0
653	0	0	4	248	0	0	C	0	0.729657	680	1176	0	0	0
626	0	0	1	221	0	0	C	0	1	1153	1153	0	0	0
760	5	5	5	355	0.000844483	0.000181917	C	0	1	41	41	8.82026	5.12663	5.12663
784	1	1	11	379	0.00256739	0	C	7	0.362895	37	218	0	0	0
432	2	3	7	27	0.00287461	0	C	2	0.992157	218	17	3.56667	2.31111	5.87778
524	1	1	6	119	0.00200541	0	C	0	1	239	239	2.00303	1.04545	7.4303
628	1	6	1	223	0.0196484	0.0030426	C	0	1	1347	1347	7.23529	1.26471	1.64706
613	0	0	4	208	0	0	C	0	0.941176	630	245	0	0	0
503	1	1	1	98	0.000109605	0	C	0	1	536	536	1.19805	1.09091	1.0974
648	0	0	2	243	0	0	C	1	0.984436	691	1215	0	0	0
609	1	5	10	204	0.00292005	0	C	1	0.825397	37	2095	5.05051	1.03959	3.13652
696	0	0	7	291	0	0	C	1	0.678261	209	41	0	0	0
697	0	0	1	292	0	0	C	0	1	549	549	0	0	0
698	0	0	1	293	0	0	C	0	1	392	392	0	0	0
699	0	0	3	294	0	0	C	1	0.713333	356	215	0	0	0
700	1	1	9	295	0.0423654	0.0158996	C	2	0.637119	86	41	5	3	18
701	0	0	9	296	0	0	C	3	0.860696	96	41	0	0	0
702	0	0	1	297	0	0	C	0	1	514	514	0	0	0
703	0	0	10	298	0	0	C	3	0.428818	100	41	0	0	0
622	0	0	2	217	0	0	C	0	1	1399	1399	0	0	0
422	3	5	6	17	0.00106909	0.00012846	C	2	0.977528	518	1268	8.32105	3.69737	7.21053
754	3	5	3	349	0.00213306	0.000363534	C	2	0.975155	99	41	10.1041	5.05572	5.05572
743	0	0	6	338	0	0	C	2	0.961538	122	41	0	0	0
650	0	0	2	245	0	0	C	1	0.961039	688	1203	0	0	0
646	0	0	2	241	0	0	C	0	0.640625	702	1254	0	0	0
640	0	0	1	235	0	0	C	0	1	1315	1315	0	0	0
632	2	5	4	227	0.00035796	0.000497411	C	0	0.772727	881	371	4.50812	1.48144	3.35228
625	0	0	1	220	0	0	C	0	1	1156	1156	0	0	0
618	0	0	1	213	0	0	C	0	1	1457	1457	0	0	0
611	0	0	2	206	0	0	C	0	0.944444	654	1624	0	0	0
607	0	0	8	202	0	0	C	5	0.905797	24	2095	0	0	0
603	0	0	1	198	0	0	C	0	1	1347	1347	0	0	0
597	1	1	3	192	0.000347428	0	C	1	0.958333	213	2064	1.08481	1.04417	3.09541
593	0	0	2	188	0	0	C	0	1	2095	2095	0	0	0
587	3	3	4	182	0.0118987	0.00622253	C	3	0.994001	230	898	2.99852	2.85672	6.51819
582	0	0	4	177	0	0	C	0	1	792	792	0	0	0
580	0	0	1	175	0	0	C	0	1	792	792	0	0	0
574	6	28	6	169	0.000761946	0.000117741	C	1	0.735294	62	594	33.2394	5.64783	5.64783
568	2	8	6	163	0.00169389	0.00124456	C	2	0.445545	122	664	5.98801	1.69724	5.14089
561	4	12	4	156	0.00105608	0.00244306	C	0	0.681416	616	705	16.1634	4.14825	5.92157
544	3	16	3	139	0.0014605	0.00168221	C	0	1	863	863	21.9149	3.96852	5.46736
538	0	0	7	133	0	0	C	0	0.672727	882	906	0	0	0
532	2	3	5	127	0.0000890908	0.000129038	C	1	0.924419	355	884	7.38105	2.28629	7.33669
528	1	1	5	123	0.00000645489	0.000000368823	C	2	0.972973	692	1093	170.087	4.1746	8.70635
522	1	2	1	117	0.00105215	0	C	0	1	1232	1232	4.5742	1.89929	6.9523

جدول ب.۱ ادامه از صفحه قبل

ID	ADEV	COMM	DDEV	FID	NADD	NDEL	B/C	MNR	OWN	EXP	OEXP	NCOMM	NADEV	NDDEV
500	1	1	1	95	0.000927289	0	C	0	1	544	544	1.30357	1.13149	1.13799
437	4	14	4	32	0.00178173	0.000165709	C	2	0.549738	149	5	14.6071	4.05613	4.05613
430	3	4	6	25	0.00300685	0	C	2	0.606383	144	568	5.06354	3.28453	5.95856
424	2	5	5	19	0.000858814	0.000067331	C	1	0.986111	294	69	7.51444	3.35696	7.12598
417	2	2	8	12	0.00186232	0.000779423	C	3	0.423963	193	1410	1.90909	1.64545	6.79091
413	1	3	8	8	0.0197083	0	C	4	0.763596	185	612	3.19048	1.70476	6.77143
406	3	9	8	1	0.00536776	0.162183	C	5	0.413559	121	612	6.36916	2.46028	6.90421
407	2	3	7	2	0.00475462	0.00316456	C	1	0.8	229	612	5.94426	2.41639	6.95738

کتاب نامه

- [1] A. Bacchelli, M. D'Ambros, and M. Lanza, "Are popular classes more defect prone?" In *International Conference on Fundamental Approaches to Software Engineering*, Springer, 2010, pp. 59–73.
- [2] N. Limsettho, K. E. Bennin, J. W. Keung, H. Hata, and K. Matsumoto, "Cross project defect prediction using class distribution estimation and oversampling," *Information and Software Technology*, vol. 100, pp. 87–102, 2018.
- [3] L. Chen, B. Fang, and Z. Shang, "Software fault prediction based on one-class svm," in *Machine Learning and Cybernetics (ICMLC), 2016 International Conference on*, IEEE, vol. 2, 2016, pp. 1003–1008.
- [4] J. Nam, "Survey on software defect prediction," *Department of Computer Science and Engineering, The Hong Kong University of Science and Technology, Tech. Rep*, 2014.
- [5] T. Menzies, J. Greenwald, and A. Frank, "Data mining static code attributes to learn defect predictors," *IEEE transactions on software engineering*, vol. 33, no. 1, pp. 2–13, 2007.
- [6] E. Arisholm, L. C. Briand, and M. Fuglerud, "Data mining techniques for building fault-proneness models in telecom java software," in *Software Reliability, 2007. IS-SRE'07. The 18th IEEE International Symposium on*, IEEE, 2007, pp. 215–224.
- [7] F. Rahman, D. Posnett, A. Hindle, E. Barr, and P. Devanbu, "Bugcache for inspections: Hit or miss?" In *Proceedings of the 19th ACM SIGSOFT symposium and the 13th European conference on Foundations of software engineering*, ACM, 2011, pp. 322–331.
- [8] E. Arisholm, L. C. Briand, and E. B. Johannessen, "A systematic and comprehensive investigation of methods to build and evaluate fault prediction models," *Journal of Systems and Software*, vol. 83, no. 1, pp. 2–17, 2010.
- [9] D. Radjenović, M. Heričko, R. Torkar, and A. Živković, "Software fault prediction metrics: A systematic literature review," *Information and Software Technology*, vol. 55, no. 8, pp. 1397–1418, 2013.
- [10] F. Akiyama, "An example of software system debugging.," in *IFIP Congress (1)*, vol. 71, 1971, pp. 353–359.
- [11] M. H. Halstead, *Elements of software science*. Elsevier New York, 1977, vol. 7.
- [12] D. Pawade, D. J. Dave, and A. Kamath, "Exploring software complexity metric from procedure oriented to object oriented," in *Cloud System and Big Data Engineering (Confluence), 2016 6th International Conference*, IEEE, 2016, pp. 630–634.

- [13] T. J. McCabe, "A complexity measure," *IEEE Transactions on software Engineering*, no. 4, pp. 308–320, 1976.
- [14] R. Malhotra, "Comparative analysis of statistical and machine learning methods for predicting faulty modules," *Applied Soft Computing*, vol. 21, pp. 286–297, 2014.
- [15] S. R. Chidamber and C. F. Kemerer, "A metrics suite for object oriented design," *IEEE Transactions on software engineering*, vol. 20, no. 6, pp. 476–493, 1994.
- [16] N. Nagappan and T. Ball, "Use of relative code churn measures to predict system defect density," in *Software Engineering, 2005. ICSE 2005. Proceedings. 27th International Conference on*, IEEE, 2005, pp. 284–292.
- [17] R. Moser, W. Pedrycz, and G. Succi, "A comparative analysis of the efficiency of change metrics and static code attributes for defect prediction," in *Proceedings of the 30th international conference on Software engineering*, ACM, 2008, pp. 181–190.
- [18] E. J. Weyuker, T. J. Ostrand, and R. M. Bell, "Do too many cooks spoil the broth? using the number of developers to enhance defect prediction models," *Empirical Software Engineering*, vol. 13, no. 5, pp. 539–559, 2008.
- [19] T. J. Ostrand, E. J. Weyuker, and R. M. Bell, "Programmer-based fault prediction," in *Proceedings of the 6th International Conference on Predictive Models in Software Engineering*, ACM, 2010, p. 19.
- [20] F. Rahman and P. Devanbu, "How, and why, process metrics are better," in *Proceedings of the 2013 International Conference on Software Engineering*, IEEE Press, 2013, pp. 432–441.
- [21] M. Li, H. Zhang, R. Wu, and Z.-H. Zhou, "Sample-based software defect prediction with active and semi-supervised learning," *Automated Software Engineering*, vol. 19, no. 2, pp. 201–230, 2012.
- [22] S. Kim, T. Zimmermann, E. J. Whitehead Jr, and A. Zeller, "Predicting faults from cached history," in *Proceedings of the 29th international conference on Software Engineering*, IEEE Computer Society, 2007, pp. 489–498.
- [23] H. Hata, O. Mizuno, and T. Kikuno, "Bug prediction based on fine-grained module histories," in *Proceedings of the 34th International Conference on Software Engineering*, IEEE Press, 2012, pp. 200–210.
- [24] S. Kim, E. J. Whitehead Jr, and Y. Zhang, "Classifying software changes: Clean or buggy?" *IEEE Transactions on Software Engineering*, vol. 34, no. 2, pp. 181–196, 2008.
- [25] R. Just, D. Jalali, L. Inozemtseva, M. D. Ernst, R. Holmes, and G. Fraser, "Are mutants a valid substitute for real faults in software testing?" In *Proceedings of the 22nd ACM SIGSOFT International Symposium on Foundations of Software Engineering*, ACM, 2014, pp. 654–665.
- [26] S. Moon, Y. Kim, M. Kim, and S. Yoo, "Ask the mutants: Mutating faulty programs for fault localization," in *Software Testing, Verification and Validation (ICST), 2014 IEEE Seventh International Conference on*, IEEE, 2014, pp. 153–162.
- [27] M. Papadakis and Y. Le Traon, "Metallaxis-fl: Mutation-based fault localization," *Software Testing, Verification and Reliability*, vol. 25, no. 5-7, pp. 605–628, 2015.

- [28] D. Hao, T. Lan, H. Zhang, C. Guo, and L. Zhang, “Is this a bug or an obsolete test?” In *European Conference on Object-Oriented Programming*, Springer, 2013, pp. 602–628.
- [29] D. Bowes, T. Hall, M. Harman, Y. Jia, F. Sarro, and F. Wu, “Mutation-aware fault prediction,” in *Proceedings of the 25th International Symposium on Software Testing and Analysis*, ACM, 2016, pp. 330–341.
- [30] X. Xia, E. Shihab, Y. Kamei, D. Lo, and X. Wang, “Predicting crashing releases of mobile applications,” in *Proceedings of the 10th ACM/IEEE International Symposium on Empirical Software Engineering and Measurement*, ACM, 2016, p. 29.
- [31] L. Kumar, S. Rath, and A. Sureka, “An empirical analysis on effective fault prediction model developed using ensemble methods,” in *2017 IEEE 41st Annual Computer Software and Applications Conference (COMPSAC)*, vol. 1, Jul. 2017, pp. 244–249. DOI: [10.1109/COMPSAC.2017.53](https://doi.org/10.1109/COMPSAC.2017.53).
- [32] R. Just, D. Jalali, and M. D. Ernst, “Defects4j: A database of existing faults to enable controlled testing studies for java programs,” in *Proceedings of the 2014 International Symposium on Software Testing and Analysis*, ser. ISSTA 2014, San Jose, CA, USA: ACM, 2014, pp. 437–440, ISBN: 978-1-4503-2645-2. DOI: [10.1145/2610384.2628055](https://doi.org/10.1145/2610384.2628055). [Online]. Available: <http://doi.acm.org/10.1145/2610384.2628055>.
- [33] R. C. Martin, *Clean code: a handbook of agile software craftsmanship*. Pearson Education, 2009.
- [34] E. B. Johannessen, “Data mining techniques, candidate measures and evaluation methods for building practically useful fault-proneness prediction models,” Master’s thesis, University of Oslo, 2008.
- [35] M. Kuhn *et al.*, “Caret package,” *Journal of statistical software*, vol. 28, no. 5, pp. 1–26, 2008.

[این صفحه آگahانه خالی گذاشته شده است.]

واژه‌نامه انگلیسی به فارسی

A Ground Assumption فرض زمینه‌ای

H

Hyperplane ابرصفحه

I

Input/Output (IO) ورودی/خروجی

Iteration تکرار

J

Java Virtual Machine ماشین مجازی جاوا

Javadoc مستند جاوا

L

Lightweight سبک وزن

M

Metadata فراداده

Mutant جهش یافته

Mutation Score امتیاز جهش

O

Obsolete منسوخ

Open-source متن باز

P

Package بسته

Performance کارایی

Plugin افزونه

Popularity شهرت

Product Metrics معیارهای محصول

Proxy Measurement اندازه‌گیری وکالتی

R

Refactoring بازآرایی کد

A

Annotated حاشیه‌نویسی شده

B

Balanced متعادل

Bias جهت‌گیری

Branch Condition شرط شاخه

Bug-proneness خطا‌پذیری

C

Classification دسته‌بندی

Cohesion همبستگی

Command line خط دستور

Comment توضیح

Commit ثبت

Component قطعه

Configuration پیکربندی

Confusion Matrix ماتریس درهم‌ریختگی

Control Flow جریان کنترلی

Coupling زوجیت

Cross-validation ارزیابی میان دسته‌ای

Cyclomatic Complexity پیچیدگی حلقوی

D

Debugging اشکال زدایی

Decision Tree درخت تصمیم

Dependency وابستگی

Descriptive Statistics آمارهای توصیفی

F

Feature Space فضای ویژگی

Framework چهارچوب

G

Generic عمومی

انتشار. Release

S

نیمه-نظارتی. Semi-Supervised

مشکوک بودن. Suspiciousness

بررسی قاعده‌مند. Systematic Review

T

زمان خروج. Timeout

V

.....

واژه‌نامه فارسی به انگلیسی

ا		چ	
آماره‌های توصیفی	Descriptive Statistics	چارچوب	Framework
اب‌صفحه	Hyperplane		
ارزیابی میان دسته‌ای	Cross-validation		
اشکال زدایی	Debugging		
افزونه	Plugin		
امتیاز جهش	Mutation Score		
انتشار	Release	حاشیه‌نویسی شده	Annotated
اندازه‌گیری وکالتی	Proxy Measurement		
ب		خ	
بازآرایی کد	Refactoring	خط دستور	Command line
بررسی قاعده‌مند	Systematic Review	خطا‌خیزی	Bug-proneness
پسته	Package		
پ		د	
پیچیدگی حلقوی	Cyclomatic Complexity	درخت تصمیم	Decision Tree
پیکربندی	Configuration	دسته‌بندی	Classification
ت		ز	
تکرار	Iteration	زمان خروج	Timeout
توضیح	Comment	زوجیت	Coupling
ث		س	
ثبت	Commit	سبک‌وزن	Lightweight
ج		ش	
جریان کنترلی	Control Flow	شرط شاخه	Branch Condition
جهت‌گیری	Bias	شهرت	Popularity
جهش‌یافته	Mutant		

ع

Generic عمومی

ف

Metadata فراداده

Ground Assumption فرض زمینه‌ای

Feature Space فضای ویژگی

ق

Component قطعه

ک

Performance کارایی

م

Confusion Matrix ماتریس درهم‌ریختگی

Java Virtual Machine ماشین مجازی جاوا

Balanced متعادل

Open-source متن‌باز

Javadoc مستند جاوا

Suspiciousness مشکوک بودن

Product Metrics معیارهای محصول

Obsolete منسوخ

ن

Semi-Supervised نیمه‌نظارتی

و

Dependency وابستگی

Input/Output (IO) ورودی/خروجی

ه

Cohesion همبستگی

The Bug Prediction Model Based On Mutation Metrics

Abstract

Software developers notice existence of faults by report of a fault issue tracking systems or failure in software test. Then they try to locate bug and understanding the problem. Early detection of fault results in saving time and money and facilitates debugging process. Prediction models can be built and used easily by modern statistical tools. Software metrics are the most important part of prediction models. Therefore higher performance in model can be achieved using new and effective metrics. In this study, process metrics and metrics that built base on mutation analysis used and resulting models evaluated. In addition to using process metrics with mutation metrics, two group of metrics named *mutation base process* metrics and *mutation-process hybrid* introduced for building prediction models. Results showed that can mutation metrics can improve prediction performance of process metrics. Although mutation base process metrics have a predictive value, they can not perform better than mutation metrics. Also mutation-process hybrid metrics can improve performance in prediction models significantly.

Keywords: Bug Prediction, Software Testing, Mutation Metrics, Process Metrics.

[این صفحه آگاهانه خالی گذاشته شده است.]



Sharif University of Technology
Computer Engineering Department

A thesis submitted in partial fulfillment of the requirements
for the M.Sc. degree
Software Engineering

The Bug Prediction Model Based On Mutation Metrics

By:

Ali Mohebbi

Supervisor:

Dr. Hassan Mirian

August 2018

