دانشگاه صنعتی امیرکبیر
(پلی‌تکنیک تهران)
دانشکده مهندسی کامپیوتر

گزارش پروژه‌ی پایانی درس یادگیری ماشین

عنوان‌مقاله:

زیرنمونه‌برداری برای روش‌های ترکیبی و بدون نظارت جهت کشف داده‌های پرت به صورت کارآمد و مؤثر

نام دانشجو: سید احمد نقوی نوزاد

ش-د: ۹٤۱۳۱۰٦۰

استاد درس:

دکتر ناظرفرد

بهار ۱۳۹۵

بسمه تعالی

۱) مقدمه:

در ابتدا تعریف مختصری از داده‌ی پرت ارائه داده و سپس به لزوم کشف داده‌های پرت پرداخته و در ادامه به دو دسته‌بندی کلی از انواع روش‌های کشف داده‌های پرت اشاره خواهیم نمود. در نهایت به بررسی مختصری در مورد رویکرد پیشنهادشده در این مقاله خواهیم پرداخت.

یک داده‌ی پرت به یک داده‌ی ضبط‌شده و یا مجموعه‌ای از داده‌ها اطلاق می‌گردد که بنا به ظاهر نسبت به سایر مجموعه داده ناسازگار بوده و رفتار غیر نرمالی از خود بروز می‌دهند. حال این رفتار می‌تواند نسبت به کلیت مجموعه داده مورد بررسی قرار گیرد (داده‌ی پرت سراسری یا **Global Outlier**) و یا هم در حیطه‌ی یک همسایگی خاص از داده‌ی مورد نظر (مانند **K** نزدیک‌ترین همسایه) مورد توجه واقع گردد (داده‌ی پرت محلی یا **Local Outlier**). در این مقاله ما به طور خاص به دنبال کشف داده‌های پرت محلی با استفاده از روش‌های اصطلاحاً ترکیبی (**ensemble**) خواهیم بود.

کشف داده‌های پرت از اهمیت ویژه‌ای در بسیاری از کاربردهای عملی، نظیر کشف خطاهای اندازه‌گیری توسط حسگرهای مختلف (که به نوعی با داده‌ی پرت به عنوان نویز رفتار می‌کنند)، کشف سوء استفاده از کارت‌های اعتباری و یا هم یافتن اندازه‌گیری‌های غیرمتعارف در داده‌های علمی دارد؛ چرا که رفتار غیرمتعارفی که داده‌های پرت نسبت به باقی داده‌ها از خود بروز می‌دهند، می‌تواند موجب بروز مشکلات عدیده گشته و در نتیجه نیازمند توجهات خاصی بوده و البته می‌توانند سبب به وجود آمدن بینش‌های جدیدی نیز نسبت به مجموعه‌داده‌ی مورد بررسی گردند.

روش‌ها و رویکردهای متعددی جهت کشف داده‌های پرت تاکنون ارائه شده‌اند، که البته هر کدام با توجه به نوع داده‌ی پرتی که بر روی آن متمرکز شده‌اند و یا هم کاربرد و زمینه‌ی خاصی که نیازمند کشف داده‌ی پرت می‌باشد، با دیگران تفاوت دارند. دسته‌ی اول، رویکردهای آماری پارامتریک[1] می‌باشند، که سعی دارند تا با استفاده از تخمین پارامترهای یک تخمین فرضی، آن تخمین را به مجموعه‌داده‌ی مورد بررسی نسبت دهند. اما مشکل موجود در مورد این نوع رویکردهای مبتنی بر یک توزیع فرضیِ خاص، این است که پارامترهای توزیع مدنظر نسبت به حضور داده‌های پرت حساس بوده و ممکن است در محاسبه‌ی آن‌ها دچار خطا شویم. از جمله اثرات مضری که داده‌های پرت بر روی تخمین پارامترهای توزیع مد نظر می‌گذارند، می‌توان به پدیده‌هایی تحت عنوان «پوشش»[2] و نیز «غرقانیدن»[3] اشاره نمود. و اما دسته‌ی دوم از رویکردهای مورد استفاده جهت کشف داده‌های پرت، با نام رویکردهای غیرپارامتریک[4] شناخته می‌شوند، که بر عکس مورد قبلی، توزیع خاصی را به داده‌ها نسبت نداده بلکه سعی در آن دارند تا به صورت آشکارا و یا هم ضمنی، وجهه‌های خاصی از تابع توزیع چگالی احتمالاتی را تخمین بزنند. از جمله روش‌های مبتنی بر این نوع رویکرد، روش‌های «مبتنی بر فاصله» و روش‌های «مبتنی بر چگالی» می‌باشند که البته این روش‌ها نیز می‌توانند مانند مورد قبل، از اثرات مضری مشابه پوشش و غرقانیدن که در نتیجه‌ی عمل تخمین‌زدن فاصله و چگالی صورت می‌گیرد، آسیب ببینند.

و اما در مورد رویکرد مورد استفاده در این مقاله باید گفت که با توجه به این مطلب که تکنیک‌های ترکیبی در مورد کشف داده‌های پرت کمتر مورد مطالعه قرار گرفته‌اند، لذا قصد داریم تا به «زیرنمونه‌برداری» تحت عنوان یک تکنیک خاص بپردازیم، که سبب می‌شود تا تنوع میان انواع روش‌های کشف داده‌ی پرت آشکار گردد. هم‌چنین به صورت آماری و مبتنی بر آزمایشات نشان خواهیم داد که یک روش کشف‌کننده‌ی داده‌ی پرت، بر روی یک

---

[1] Parametric Statistical Approaches
[2] Masking
[3] Swamping
[4] Non-parametric Approaches

زیرنمونه به خودی خود، علاوه بر اینکه سبب استنتاج تنوع میان انواع روش‌ها می‌گردد، تحت شرایط خاصی از عملکرد همان روشِ کشف داده‌ی پرت بر روی کل مجموعه‌داده پیشی می‌گیرد. البته که ساخت یک روش ترکیبی که بر روی زیرنمونه‌های متعدد کار می‌کند، می‌تواند سبب بهبود چشمگیر نتایج گردد؛ و نیز ذکر این نکته ضروری می‌نماید که روش‌های ترکیبی مطرح‌شده تاکنون در مقالات متعدد، روش‌های بانظارت[5] بوده و تلاش در دسته‌بندی[6] داده‌ها دارند، ولی در این مقاله تلاش خواهیم داشت تا به صورت آماری، صحت عملکرد روش‌های ترکیبی بی‌نظارت[7] و مبتنی بر خوشه‌بندی[8] در کشف داده‌های پرت را به اثبات برسانیم. در پایان باید گفت که استفاده از ترکیبی از روش‌های کشف داده‌های پرت بر روی زیرنمونه‌های متعدد از مجموعه‌داده، با توجه به اندازه‌ی زیرنمونه‌ها[9] و نیز حجم کلی روش‌های ترکیب‌شده[10]، به طور کلی نسبت به استفاده از یک روش منفرد کشف داده‌ی پرت بر روی کل مجموعه‌داده عملکرد بهتری خواهد داشت.

۲) شرح روش و پارامترها:

در این مقاله این‌گونه نیست که تنها از مجموعه‌داده نمونه‌برداری کرده و سپس الگوریتم کشف داده‌ی پرت را بر روی زیرنمونه‌های مربوطه و با احتساب حضور سایر داده‌های خارج از زیرنمونه اجرا نمائیم، چرا که با این کار اطلاعات زیادی در مورد ماهیت پرت‌بودن بسیاری از داده‌ها از بین رفته و نیز بسیاری از داده‌ها نیز تنها امتیازی حسب پرت‌بودن از تنها برخی از زیرنمونه‌ها خواهند برد. بلکه در عوض ما در این مقاله، به ازای هر عضو **ensemble**، یک زیرنمونه از مجموعه داده انتخاب کرده و همسایگی مربوطه را تنها به ازای داده‌های موجود در زیرنمونه برای هر کدام از الگوریتم‌های کشف داده‌ی پرت محاسبه می‌نمائیم، که این روش سبب افزایش سرعت قابل توجهی در مقایسه با سایر روش‌های **ensemble** گشته و نیز نتایج نهائی بهتری را حاصل می‌نماید.

در این مقاله یک روش پایه را با نام **Feature Bagging** ، به عنوان رقیب روش **ensemble** معرفی‌شده در این مقاله مورد استفاده قرار می‌دهیم و برای هر دوی روش‌های ترکیبی (**ensemble** و **Feature Bagging**) یک مقدار معین برای تعداد اعضای **ensemble** یعنی ۲۵ را انتخاب خواهیم نمود. در مورد روش رقیب یعنی **FeatureBagging** نیز جهت ترکیب نتایج نهائی، از عمل میانگین‌گیری استفاده خواهیم نمود. در مورد روش **ensemble** نیز از زیرنمونه‌های با اندازه‌های متفاوت جهت ارزیابی نتایج نهائی استفاده نموده و در مورد روش‌های پایه نیز از اندازه‌های مختلف **k** برای یافتن نزدیک‌ترین همسایه‌ها استفاده می‌کنیم. در نهایت نتایج نهائی را در قالب‌های زیر نشان خواهیم داد: **i)** مقدار ثابت **k** و اندازه‌های مختلف زیرنمونه؛ **ii)** مقدار ثابت زیرنمونه و مقادیر مختلف **k**؛ **iii)** مقادیر ثابت برای هر دوی اندازه‌ی زیرنمونه و مقدار **k**. در ضمن مقداری را برای **k** انتخاب خواهیم نمود که در مورد روش پایه به نتایج مطلوبی منجر گردد.

هم‌چنین برای ارزیابی نتایج نهائی از **ROC AUC**[11] استفاده می‌نمائیم که مقادیر مختلف **True Positive Rate** را بر حسب مقادیر **False Positive Rate** رسم نموده و نیز یک معیار عمومی و پرکاربرد جهت ارزیابی روش‌های مختلف کشف داده‌های پرت می‌باشد.

---

[5] Supervised
[6] Classification
[7] Unsupervised
[8] Clustering
[9] Subsample size
[10] Ensemble size
[11] Area under the receiver operating characteristic curve (ROC AUC)

۳)مجموعه داده‌های مورد استفاده

در این‌جا دو مجموعه داده‌ی مستقل دست‌ساز و مصنوعی[۱۲] را تولید می‌نمائیم (batch1 و batch2) که هر کدام از متشکل از ۳۰ مجموعه‌داده‌ی مختلف با ابعاد، تعداد خوشه‌ها و نیز اندازه‌ی متفاوت خوشه‌ها می‌باشند. مجموعه‌ی داده‌های مربوطه را با استفاده از توزیع گاوسین مخلوط[۱۳] تولید می‌نمائیم و برای این کار از مقادیر مختلف میانگین و انحراف از معیار که از هر کدام در بازه‌های مشخصی می‌باشند استفاده می‌نمائیم. در نهایت می‌بایست مجموعه داده‌های مربوطه را در فضای چندبعدی چرخش[۱۴] دهیم که برای این‌کار نیز ابتدا یک ماتریس A را با داده‌های رندوم نرمال تولید نموده و سپس با ضرب ترانهاده‌ی این ماتریس در خود آن، یک ماتریس واریانس-کوواریانس به دست می‌آوریم که خاصیت لازم به اصلاح positive-semiDefinite در مورد آن برقرار می‌باشد. بعد ازین با استفاده از تابع mvnrnd()، یک توزیع گاوسین مخلوط را در مورد داده‌های مربوط به هر کدام از خوشه‌های یک مجموعه‌داده‌ی مشخص به دست می‌آوریم. جهت این‌که داده‌های پرت محلی را در مورد این مجموعه‌داده‌ها برچسب‌گذاری نمائیم، در ابتدا فاصله‌ی ماهالانوبیس[۱۵] را در مورد داده‌های موجود درون هر خوشه از مرکز خوشه محاسبه نموده و سپس آن داده‌هایی را که این فاصله برای آن‌ها بیشتر از چارک ۰,۹۷۵ توزیع مربوط به فاصله‌های داده‌های درون خوشه باشد، به عنوان داده‌ی پرت برچسب‌گذاری می‌نمائیم.

در مورد مجموعه‌داده‌های واقعی مورد استفاده نیز، در این‌جا از مجموعه‌داده‌های معروف Satimage، Segment (که به سه مجموعه‌داده‌ی مجزا با داده‌های پرت متفاوت تبدیل خواهد شد)، Wisconsin Breast Cancer (WBC) و نیز مجموعه‌داده‌ی Waveform Database Generator (waveform) استفاده خواهیم نمود. در مورد هر کدام از این مجموعه‌داده‌ها در ابتدا آن کلاسی را که تعداد داده‌های کمتری را داراست به عنوان طعمه انتخاب نموده و سپس مانند مجموعه‌داده‌های مصنوعی که پیش ازین قید گردید، آن داده‌هایی را که فاصله‌ی ماهالانوبیس آن‌ها از مرکز خوشه، بیشتر از چارک ۰,۹ توزیع مربوط به فاصله‌های داده‌های درون خوشه باشد، به عنوان داده‌ی پرت برچسب‌گذاری می‌نمائیم.

در نهایت از الگوریتم‌های کشف داده‌های پرت محلی جهت شناسائی داده‌های پرت در مجموعه‌داده‌های نامبرده استفاده خواهیم نمود.

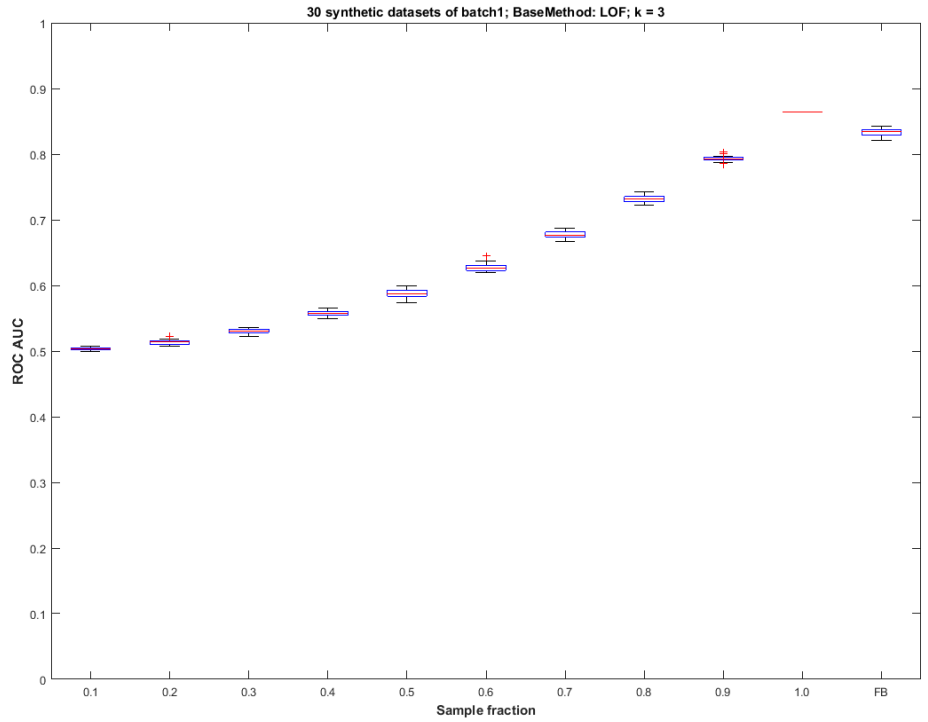
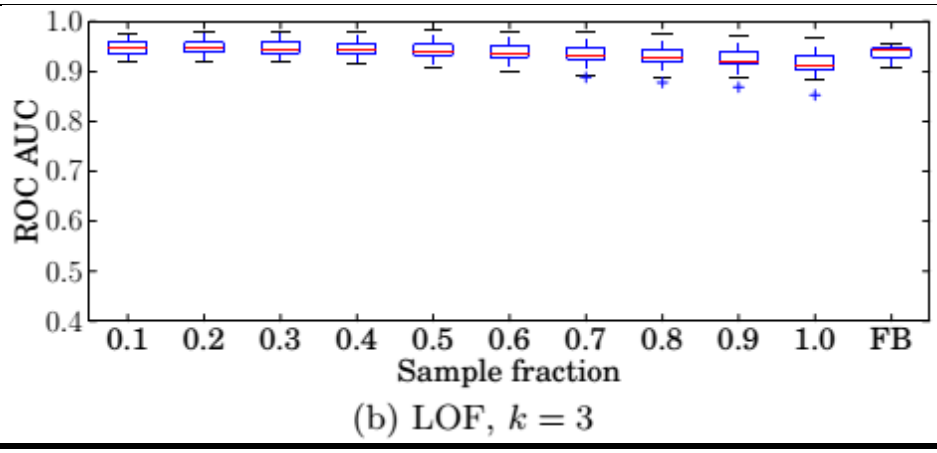
۴)نتایج مربوط به پیاده‌سازی

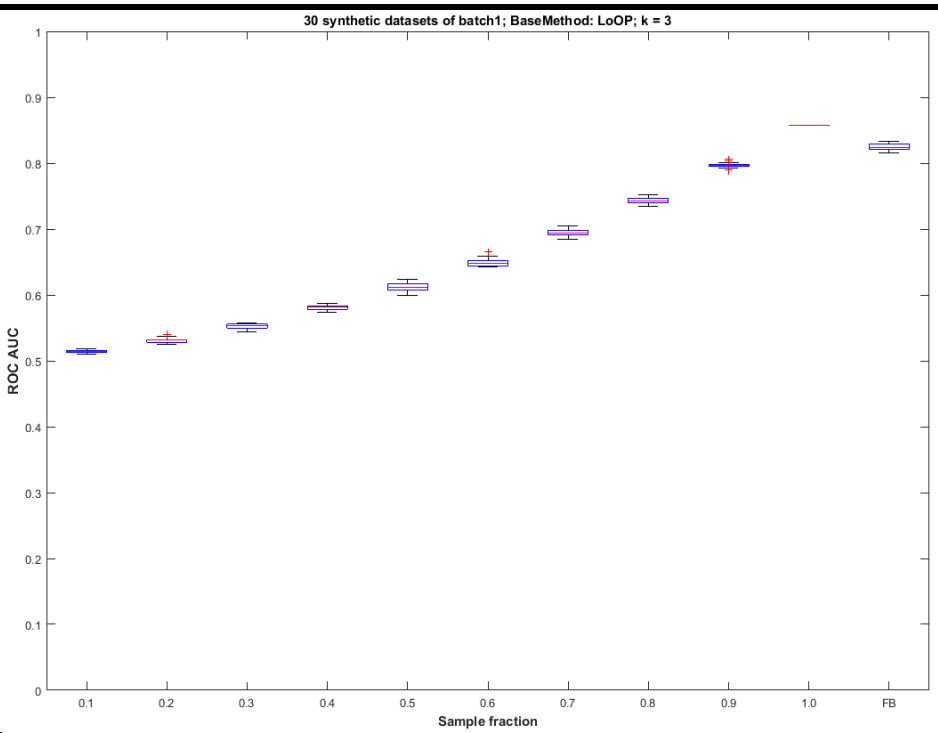
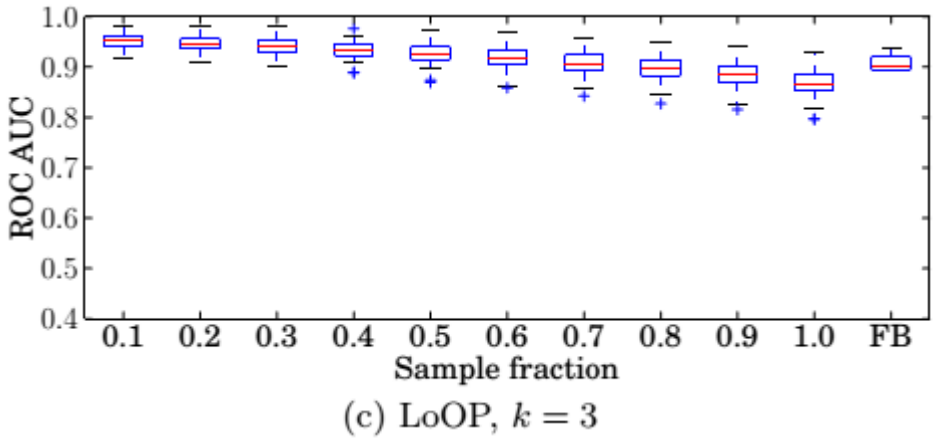
۱) مجموعه‌داده‌های مصنوعی

در این‌جا مجموعه‌داده‌ی مصنوعی batch1 متشکل از ۳۰ مجموعه‌داده‌ی مجزا را به عنوان کاندید برگزیده و دو روش پایه جهت کشف داده‌های پرت محلی با نام‌های LOF[۱۶] و LoOP[۱۷] را بر روی آن‌ها پیاده نموده و روش‌های ترکیبی قیدشده پیش ازین، با نام‌های FeatureBagging و ensemble را نیز با پارامترهای مشخص (اندازه‌ی زیرنمونه‌های مختلف و مقدار مشخص k=3) بر روی آن‌ها پیاده می‌نمائیم. در این‌جا مقادیر ROC AUC مختلف به ازای زیرنمونه‌های مختلف را به دست آورده و برای رسم آن‌ها از تابع boxplot() استفاده می‌نمائیم. نتایج بیان‌شده در مقاله و نیز نتایج حاصله از پیاده‌سازی در ادامه می‌آید:

---

12 Synthetic
13 Gaussian Mixture Distributions
14 Rotation
15 Mahalanobis
16 Local Outlier Factor (LOF)
17 Local Outlier Probability (LoOP)

	نتایج حاصل از پیاده‌سازی بر روی مجموعه‌داده‌ی مصنوعی batch1 و روش پایه‌ی LOF و k=3
 (b) LOF, $k = 3$	نتایج بیان‌شده در مقاله در مورد مجموعه‌داده‌ی مصنوعی batch1 و روش پایه‌ی LOF و k=3

	نتایج حاصل از پیاده‌سازی بر روی مجموعه‌داده‌ی مصنوعی batch1 و روش پایه‌ی LoOP و k=3
	نتایج بیان‌شده در مقاله در مورد مجموعه‌داده‌ی مصنوعی batch1 و روش پایه‌ی LoOP و k=3

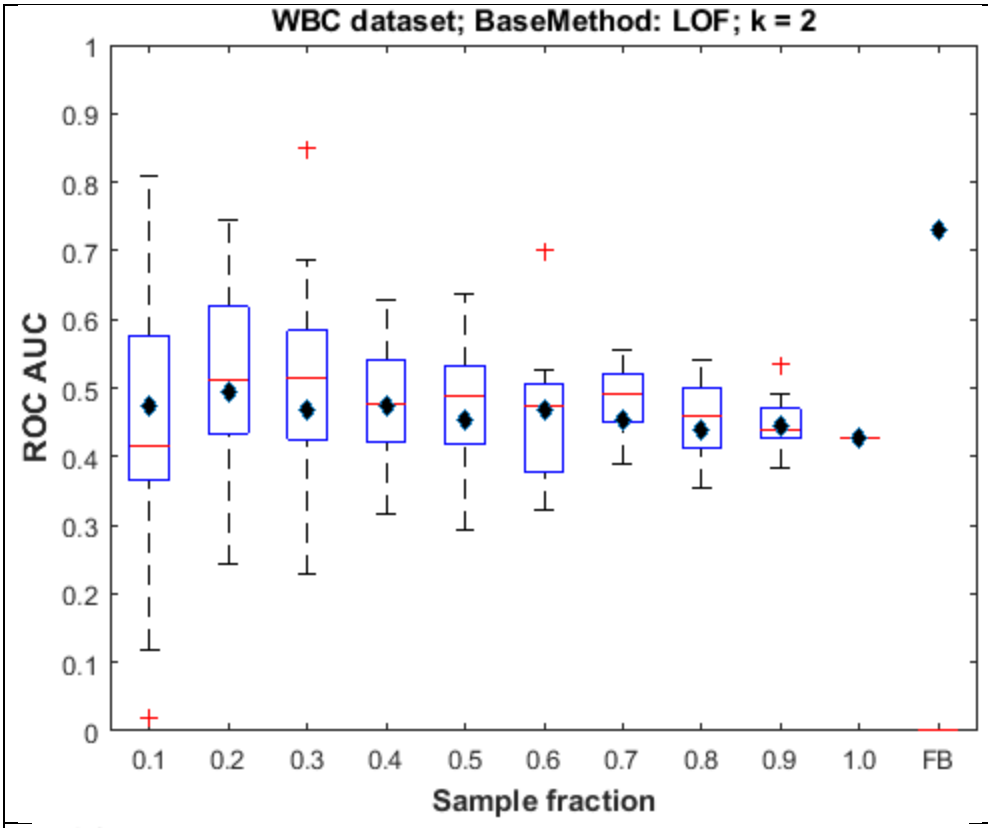
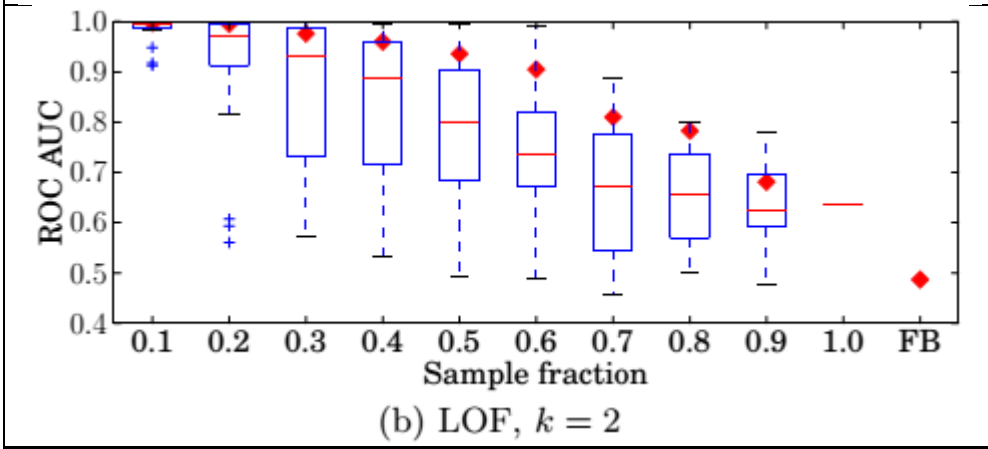
شکل ۱. نتایج حاصل از پیاده‌سازی بر روی مجموعه‌داده‌ی مصنوعی batch1 و روش‌های پایه‌ی مختلف

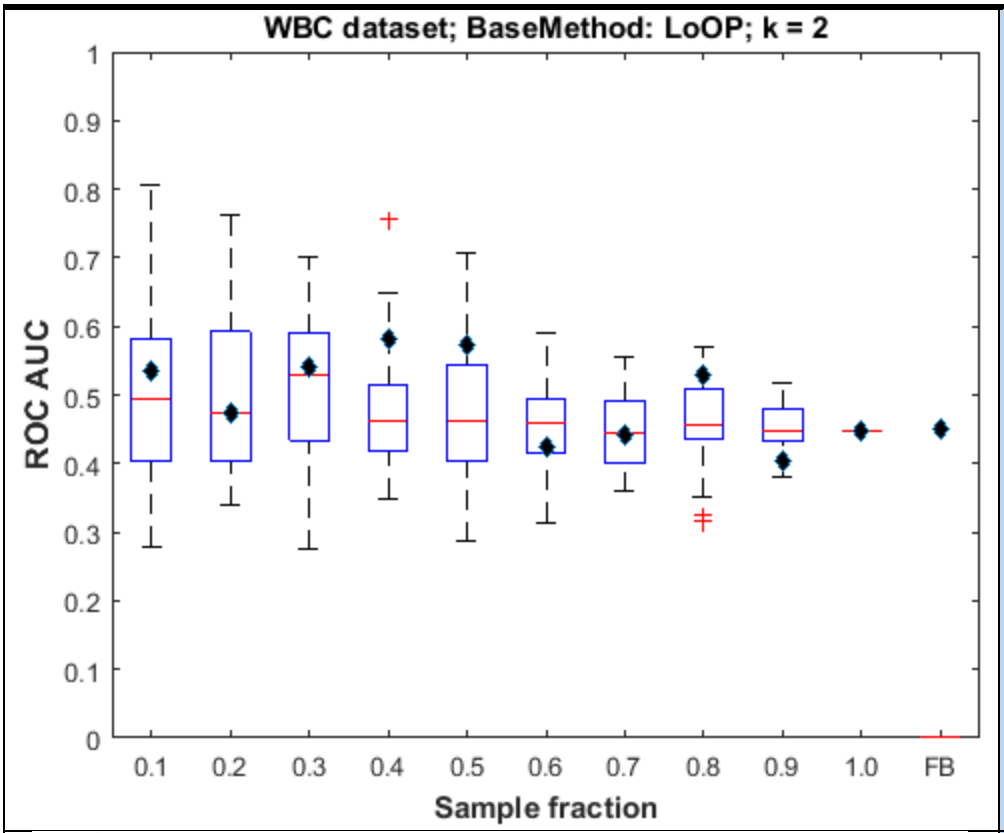
متأسفانه همانطور که قابل مشاهده است، نتایج حاصل از پیاده‌سازی با نتایج قیدشده در مقاله یکسان نمی‌باشد و علت این مسئله را نیز می‌توان در تصادفی‌بودن رویه‌ی تولید داده‌های مصنوعی جستجو نمود.

با توجه به نکات قیدشده درمقاله انتظار آن بود که با افزایش اندازه‌ی زیرنمونه، شاهد کاهش محدوده‌ی توزیع مقادیر ROC AUC باشیم که در این‌جا این‌گونه نشد، اما در مورد برخی مجموعه‌داده‌های واقعی که در ادامه خواهد آمد، نتایج خوبی را مطابق مندرجات موجود در مقاله شاهد خواهیم بود.

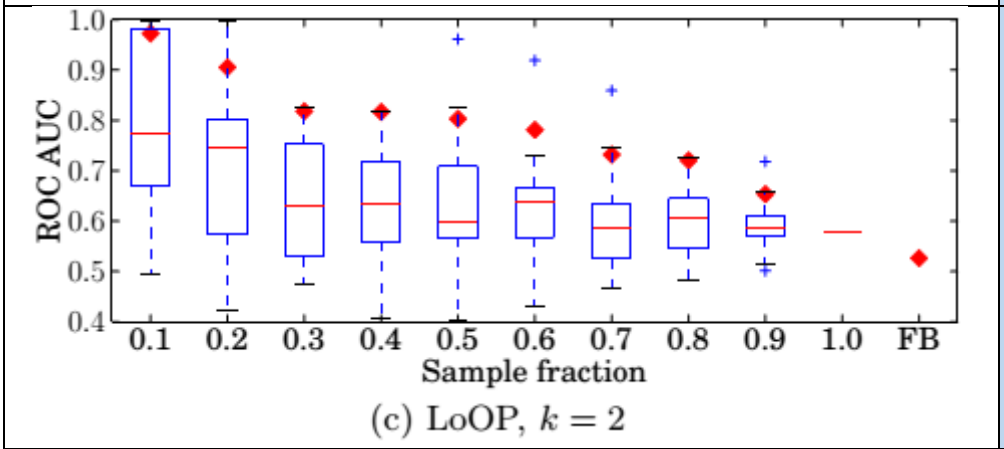
۲) مجموعه‌داده‌های واقعی

در ابتدا به مجموعه‌داده‌های WBC و Satimage خواهیم پرداخت که تنها در مورد دومی نتایج مطلوبی را شاهد خواهیم بود. نتایج مربوط به مجموعه‌داده‌ی WBC به شرح ذیل می‌باشند:

	نتایج حاصل از پیاده‌سازی بر روی مجموعه‌داده‌ی WBC و روش پایه‌ی LOF و k=2
 (b) LOF, $k = 2$	نتایج بیان‌شده در مقاله در مورد مجموعه‌داده‌ی WBC و روش پایه‌ی LOF و k=2



نتایج حاصل از پیاده‌سازی بر روی مجموعه‌داده‌ی WBC و روش پایه‌ی LoOP و k=2

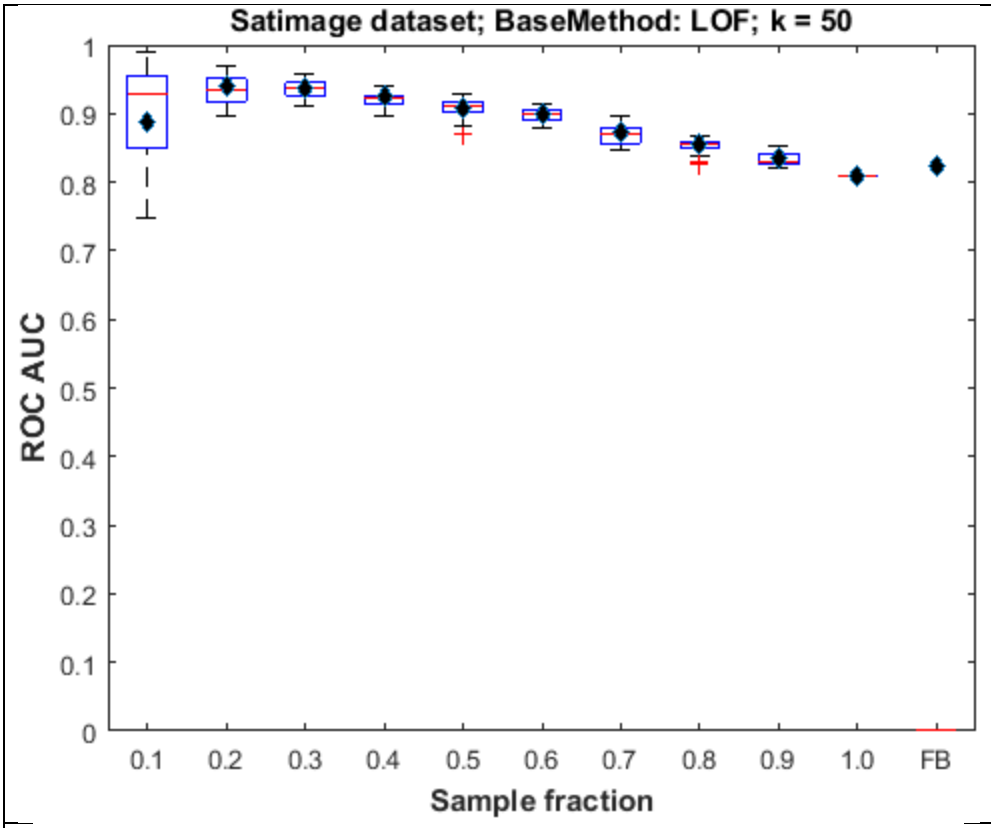
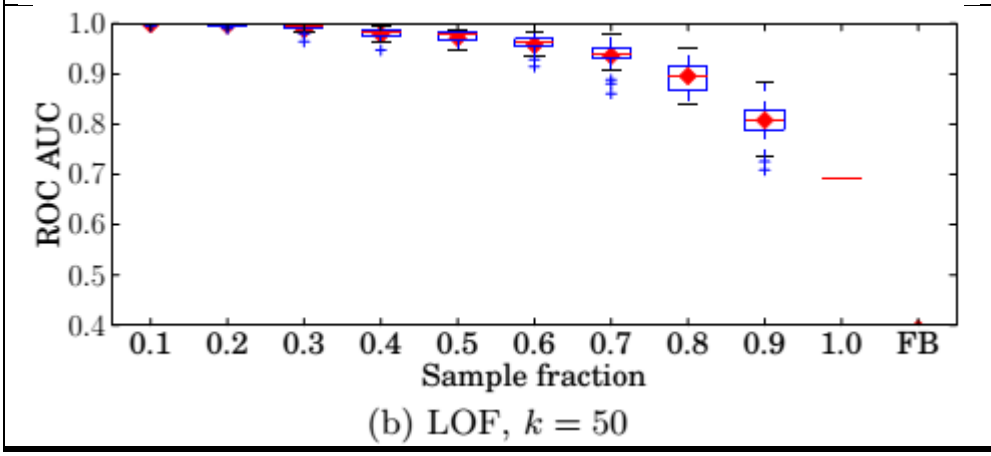


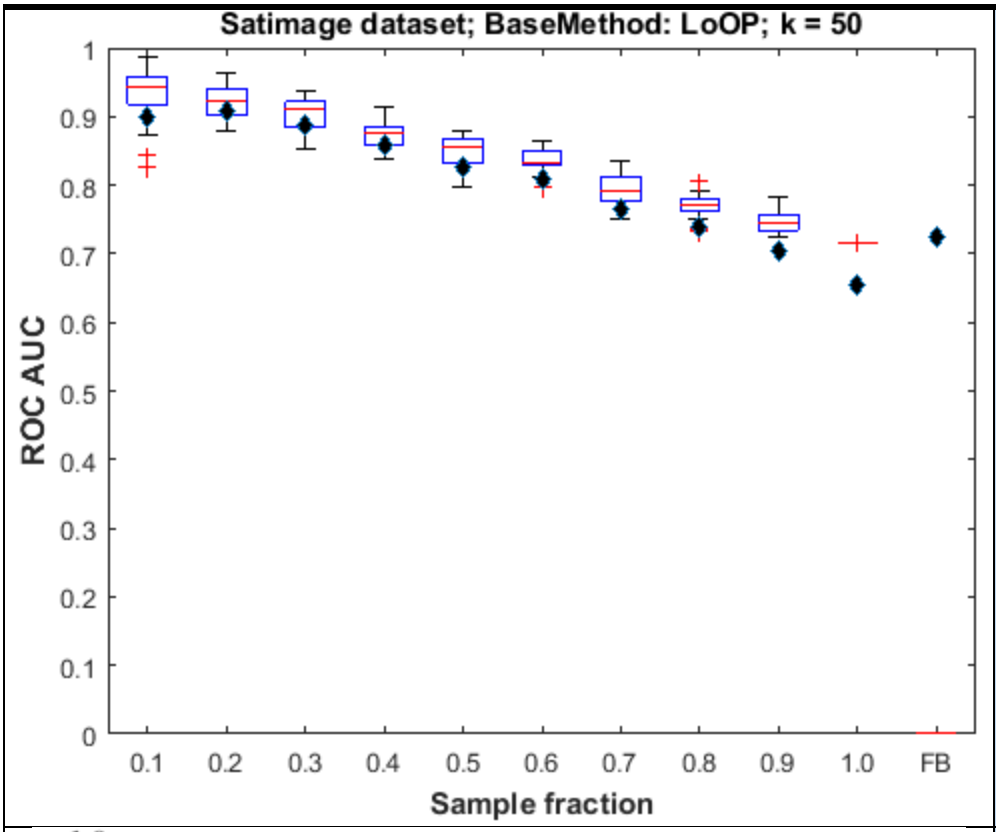
(c) LoOP, $k = 2$

نتایج بیان‌شده در مقاله در مورد مجموعه‌داده‌ی WBC و روش پایه‌ی LoOP و k=2

شکل ۲. نتایج حاصل از پیاده‌سازی بر روی مجموعه‌داده‌ی WBC و روش‌های پایه‌ی مختلف

در ادامه نتایج مربوط به مجموعه‌داده‌ی Satimage ذکر می‌گردد. لازم به ذکر است که با توجه به بزرگ‌تر بودن مجموعه‌داده‌ی Satimage ، مقدار k را نیز بزرگ‌تر و برابر ۵۰ انتخاب می‌نمائیم. در ادامه داریم:

	نتایج حاصل از پیاده‌سازی بر روی مجموعه‌داده‌ی Satimage و روش پایه‌ی LOF و k=50
 (b) LOF, $k = 50$	نتایج بیان‌شده در مقاله در مورد مجموعه‌داده‌ی Satimage و روش پایه‌ی LOF و k=50

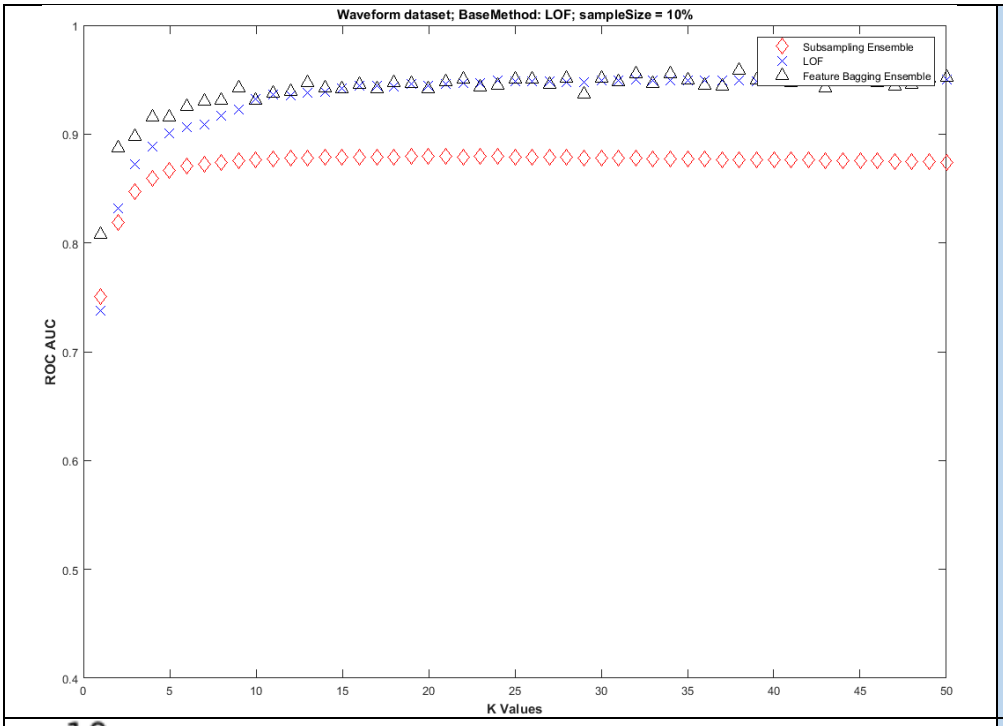
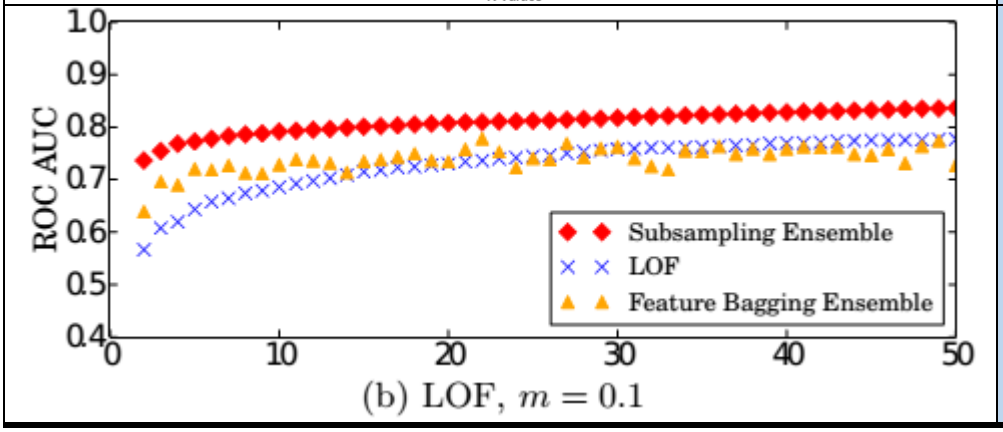


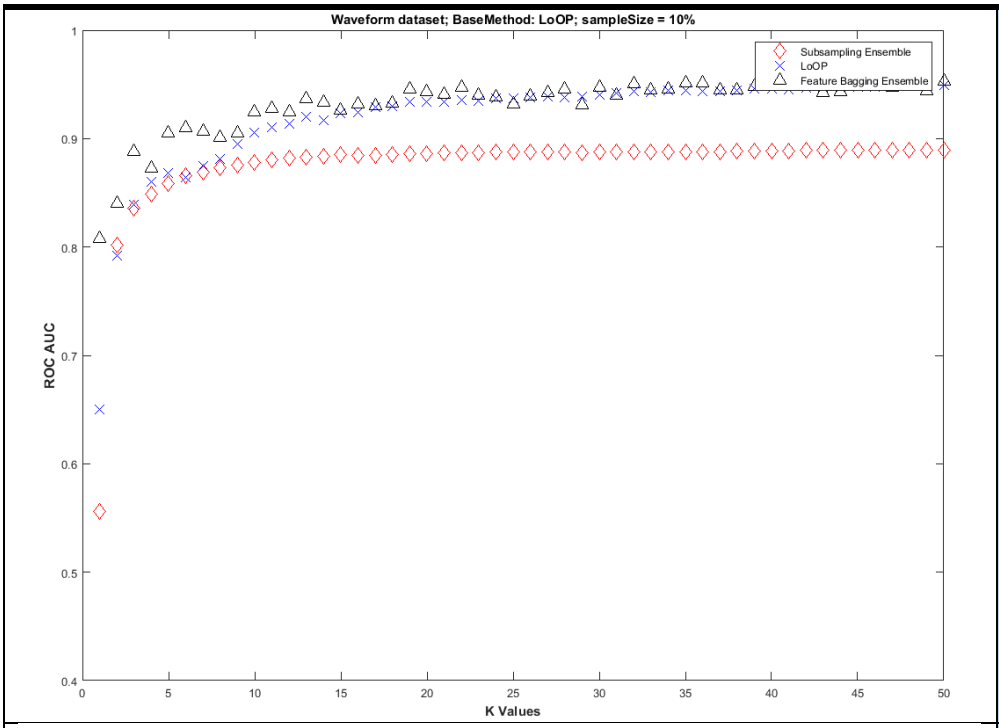
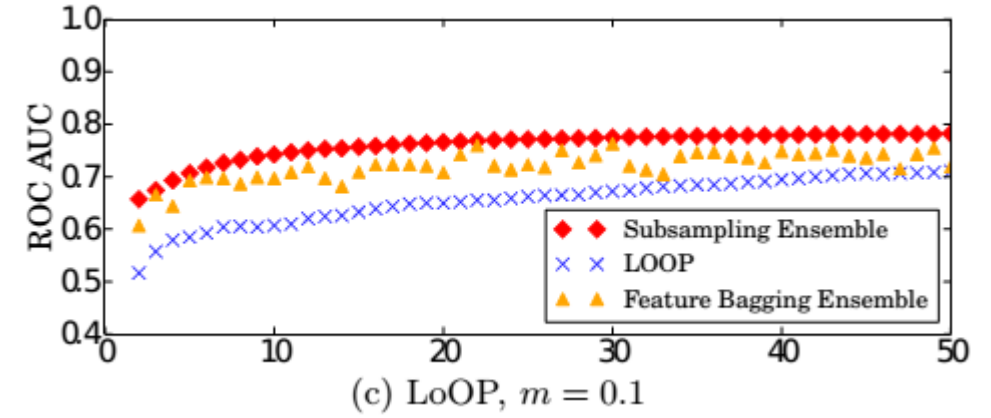
نتایج حاصل از پیاده‌سازی بر روی مجموعه‌داده‌ی Satimage و روش پایه‌ی LoOP و k=50	
نتایج بیان‌شده در مقاله در مورد مجموعه‌داده‌ی Satimage و روش پایه‌ی LoOP و k=50	

شکل ۳. نتایج حاصل از پیاده‌سازی بر روی مجموعه‌داده‌ی Satimage و روش‌های پایه‌ی مختلف

در این قسمت، اندازه‌ی زیرنمونه را ثابت و برابر ۰٫۱ در نظر گرفته و نتایج را بر حسب مقادیر متفاوت k بر روی مجموعه داده‌ی Waveform و برای روش‌های پایه، ترکیب زیرنمونه‌ها[18] و ترکیب چینش‌های مختلفی از ویژگی‌ها[19] نمایش می‌دهیم. در ادامه داریم:

[18] Subsampling Ensemble
[19] Feature Bagging Ensemble

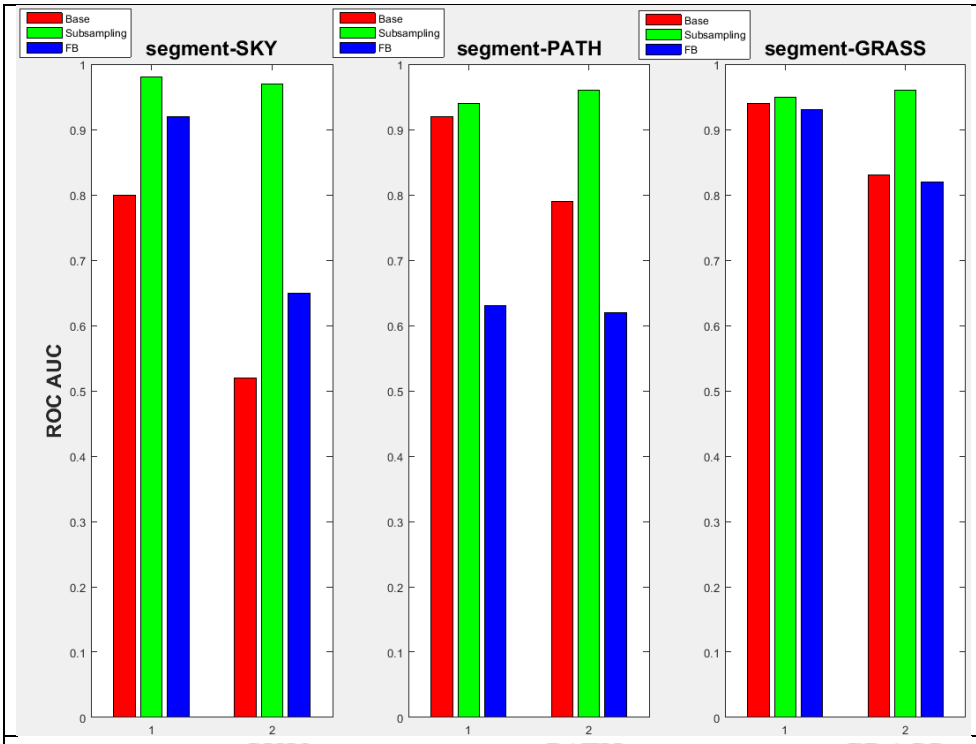
	نتایج حاصل از پیاده‌سازی بر روی مجموعه‌داده‌ی Waveform و روش پایه‌ی LOF و sampleSize=0.1
 (b) LOF, $m = 0.1$	نتایج بیان‌شده در مقاله در مورد مجموعه‌داده‌ی Satimage و روش پایه‌ی LOF و sampleSize =0.1

	نتایج حاصل از پیاده‌سازی بر روی مجموعه‌داده‌ی Waveform و روش پایه‌ی LoOP و sampleSize=0.1
	نتایج بیان‌شده در مقاله در مورد مجموعه‌داده‌ی Waveform و روش پایه‌ی LoOP و sampleSize=0.1

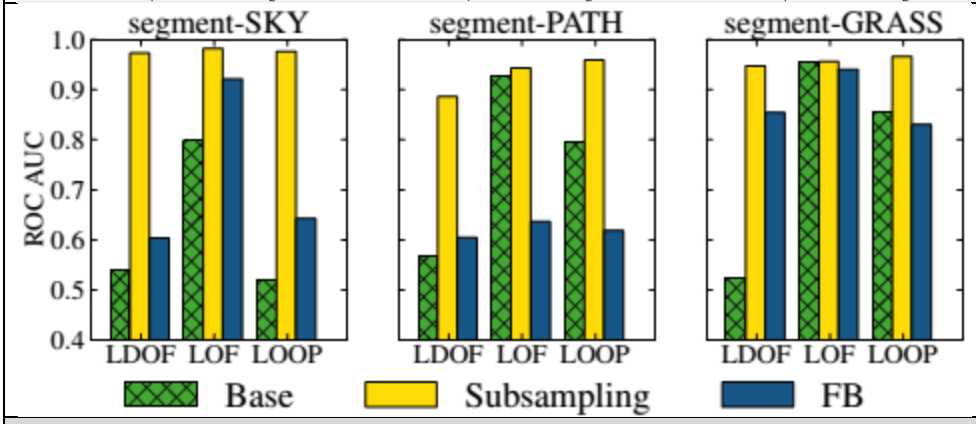
شکل ٤. نتایج حاصل از پیاده‌سازی بر روی مجموعه‌داده‌ی Waveform و روش‌های پایه‌ی مختلف

علاوه بر اینکه نتایج حاصل از پیاده‌سازی خوشبختانه با نتایج مندرج در مقاله همخوانی بالائی دارند! باید گفت که با افزایش مقدار k شاهد افزایش اندک اما پایدار در مقدار ROC AUC برای روش‌های پایه (LOF و LOoP) و Subsampling Ensemble خواهیم بود و به عبارتی هر دوی این روش‌ها از یک الگوی یکسان و پایدار پیروی می‌نمایند، در حالی که برای روش FeatureBagging شاهد نوسان و واریانس بالاتری در مورد مقادیر ROC AUC هستیم.

در ادامه نتایج حاصله در مورد مجموعه‌داده‌ی Segment را با مقادیر مشخص برای پارامترها (k=20 که در مورد این مجموعه‌داده و البته روش‌های پایه‌ی مختلف نتایج خوبی را حاصل نموده است؛ و subsampleSize=0.1) به صورتی که در ادامه می‌آید نشان می‌دهیم:



نتایج حاصل از پیاده‌سازی بر روی مجموعه‌داده‌های Segment و روش‌های پایه‌ی LOF(1) و LoOP(2) و k=20 و sampleSize=0.1



نتایج مندرج در مقاله در مورد مجموعه‌داده‌های Segment و روش‌های پایه‌ی LOF ،LDOF و LoOP و k=20 و sampleSize=0.1

شکل ۵. نتایج حاصل از پیاده‌سازی بر روی مجموعه‌داده‌های Segment و روش‌های پایه‌ی مختلف

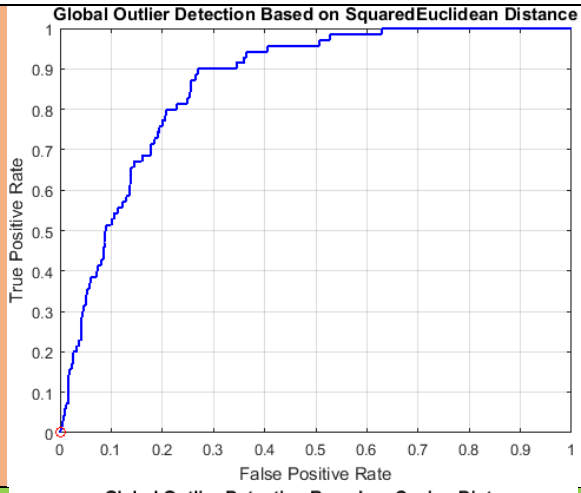
همانطور که قابل مشاهده است، روش زیرنمونه‌برداری معرفی‌شده در این مقاله به مراتب نتایج بهتری را نسبت به روش‌های پایه و البته روش رقیب FeatureBagging حاصل نموده است، که این خود گواه بر صحت ادعاهای قیدشده در مقاله می‌باشد.

۵) معرفی ایده‌ی پیشنهادی و نتایج پیاده‌سازی

تا این‌جا ما از روش‌های ترکیبی، جهت کشف داده‌های پرت محلی استفاده نمودیم؛ حال قصد داریم تا روشی را جهت کشف داده‌های پرت سراسری معرفی نمائیم. به این ترتیب که ابتدا یک مجموعه‌داده‌ی مصنوعی را مانند آن‌چه پیش از این قید گردید تهیه می‌نمائیم، تنها با این تفاوت که تعدادی داده‌ی جدید را با مقادیر بسیار دورتر از میانگین هر بعد به مجموعه داده اضافه نموده و آن‌ها را تحت عنوان داده‌ی پرت سراسری برچسب می‌زنیم. حال در عمل پیاده‌سازی ابتدا مجموعه‌داده را با استفاده از تابع kmeans() و مقادیر متفاوت clustNo (که همان

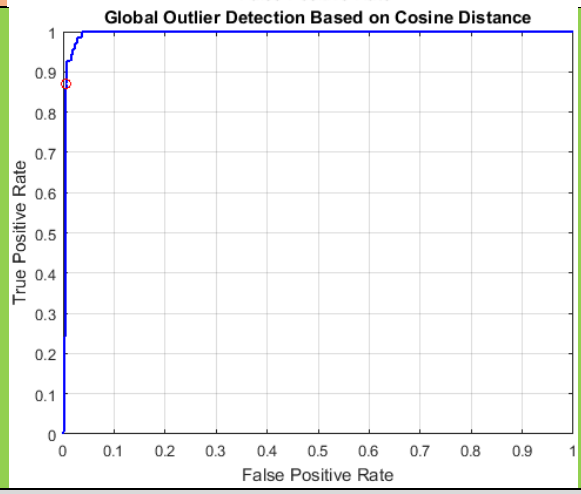
تعداد خوشه‌هاست) خوشه‌بندی نموده و سپس برای هر داده یک مقدار احتمالاتی بین ۰ و ۱ را به عنوان ضریب داده‌ی پرت سراسری معرفی می‌نمائیم، که این مقدار از تقسیم فاصله‌ی هر داده از مرکز خوشه‌ای که در آن قرار دارد بر بیشینه‌ی فاصله‌های داده‌های موجود درون خوشه از مرکز به دست می‌آید و البته عددی مابین ۰ و ۱ می‌باشد. در نهایت به ازای خوشه‌بندی‌های مختلف، مقادیر متفاوتی برای این ضرایب خواهیم داشت. سپس جهت به دست‌آوردن مقدار نهائی برای ضریب مربوطه به ازای هر داده، از روش اول‌عمق استفاده می‌نمائیم، که در آن ابتدا ماتریس متشکل از ضرایب (سطرها بیانگر داده‌ها و ستون‌ها بیانگر تعداد خوشه‌ها) را به صورت ستونی و هر ستون مجزا از دیگری، در قالب ترتیب نزولی مرتب می‌نمائیم. سپس از سطر اول شروع نموده و سطر به سطر جلو می‌رویم و اولین مقدار ضریب که به ازای هر داده مشاهده می‌کنیم را به عنوان ضریب مربوط به آن داده گزارش می‌نمائیم تا زمانی که به تمامی داده‌ها مقدار ضریب مربوطه نسبت داده شود. به عبارتی در این روش ضریبی برای هر داده از ماتریس مرتب‌شده صورت نزولی انتخاب می‌شود، که در میان خوشه‌بندی‌های مختلف، در بالاترین ردیف ممکن قرار دارد.

در نهایت، بردار ضرایب داده‌ی پرت سراسری را به همراه بردار مربوط به برچسب‌ها به تابع perfcurve() می‌دهیم تا منحنی ROC مربوط به روش پیشنهادی را برای ما رسم نموده و مساحت زیر نمودار (AUC) را نیز برای ما فراهم نماید. هر چه این مساحت بیشتر باشد، عملکرد الگوریتم پیشنهادی بهتر بوده است. لازم به ذکر است که در این‌جا ما در عمل خوشه‌بندی با استفاده از kmeans()، از دو معیار فاصله‌ی SquaredEuclidean Distance و نیز Cosine Distance استفاده می‌نمائیم که البته در مورد دومی به نتایج بهتری دست پیدا می‌کنیم. در ادامه داریم:



منحنی ROC برای روش پیشنهادی، با معیار فاصله‌ی SquaredEuqlidean Distance برای الگوریتم kmeans

AUC = 0.8648 (x)



منحنی ROC برای روش پیشنهادی، با معیار فاصله‌ی Cosine Distance برای الگوریتم kmeans

AUC = 0.9941 (√)

شکل ٦. نتایج حاصل از پیاده‌سازی روش پیشنهادی جهت کشف داده‌ی پرت سراسری

٦) مراجع

[1] **Zimek, Arthur, et al. "Subsampling for efficient and effective unsupervised outlier detection ensembles."** ***Proceedings of the 19th ACM SIGKDD international conference on Knowledge discovery and data mining*****. ACM, 2013.**

[2] **Breunig, Markus M., et al. "LOF: identifying density-based local outliers."** ***ACM sigmod record*****. Vol. 29. No. 2. ACM, 2000.**

[3] **Kriegel, Hans-Peter, et al. "LoOP: local outlier probabilities."** ***Proceedings of the 18th ACM conference on Information and knowledge management*****. ACM, 2009.**