二〇一九年六月四日星期二

下午1时55分

基于今年1月份被《计研发》退稿的意见，重新梳理自己的创新点与应补充的实验数据，为了争取在本周末（假期周日投稿）赶上此次11日的专刊，特进行以下工作：

1. 明确本文的创新点与解决问题的逻辑思路；
2. 基于梳理的思路，补充实验与数据分析图例；

本文的基本贡献：

* 内部威胁异常行为检测与威胁检测并不能等同；从反生产工作行为CPB的角度来刻画内部用户攻击倾向强弱；
* 心理测评问卷的不足：用户CPB静态反映，不能反映实时变化；攻击者可以降低测评可信度；
* 用户行为审计数据驱动的CPB模型，关键从常规审计数据中提取CPB特征；
* 目标是用低CPB训练OCSVM以检测高CPB攻击者，然而无法进行定量人工标记（依据心理测评分数不可信、CPB特征显式数值计算不可行）；
* 故提出了低CPB定性标记算法，核心思想是将CPB所有特征看作重要性相同的正负相关因素，然后通过聚类后中心对应CPB水平的相对高低选择出目标用户集合中的低CPB特征集合作为训练集；
* 通过模型分析，对于分析的两类典型内部攻击，可以将离职用户作为测试集检测；

补充的分析一

对于低CPB定性标记算法，有专家质疑其合理性，可以用CERT5.2的在职用户自动KMeans（K=2）来进行说明：

ATF数据：CERT5.2\_Leave\_Static\_CPB\_ATF-02.csv

Auto Choose K: 2 SC\_value is 0.4267696251405651

第二大轮廓系数为K=4：K值为： 4 轮廓系数为 0.1740251308160634

K=2时聚类结果：

**Cluster 0 has : 1564**

**Cluster 1 has : 197**

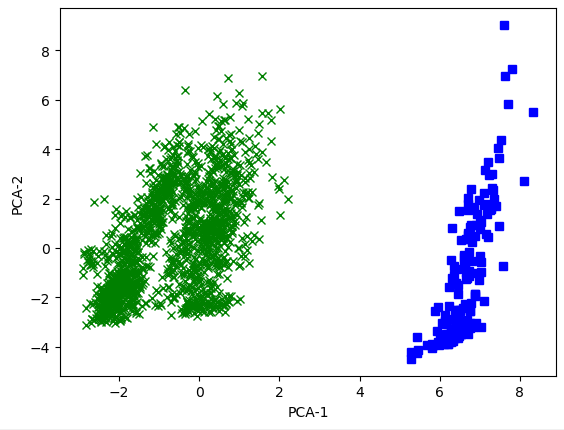
Leave Users has : 238

且计算Cluster0与Cluster1的中心CPB分数为：可见Cluster0更低

0 ['3.354237081011262', '3.2577632039058733'] 均值计算与中位数计算

1 ['3.269116358494986', '3.219709313711901']

对于Cluster0与Cluster1分别PCA=2后绘图，发现两个类别位于泾渭分明的群簇中，且分隔区域明显：左侧绿色x表示Cluster0，右侧蓝色方块表示Cluster1，经过计算，得到Cluster0中心代表的CPB要低于Cluster1，故选择Cluster0为低CPB特征集合训练OCSVM。



补充的分析二：

为了验证CPB\_FEATSlow训练OCSVM检测CPB\_FEATSleave的合理性，设置第一组实验：

1. 采用低CPB定性标记算法自动选择CPB\_FEATSlow训练OCSVM，并检测CPB\_FEATSleave，参数选择采取遍历的形式，采取最优Recall策略；
2. 采用同样的CPB\_FEATSlow训练OCSVM，但是将CPB\_FEATShigh+CPB\_FEATSleave一起作为测试集，依旧采取最优Recall策略，发现此时Recall未提升，但是FPR升高；
3. 采用传统人工标记方法，依据正常用户与攻击者进行人工标记，选择所有的CPB\_FEATSnormal训练OCSVM（从CPB\_FEATSall中去除leave与未离职attacker后），然后检测CPB\_FEATSleave，同样采用最优Recall训练策略，发现Recall降低且FPR升高，说明了定性标记算法比传统人工比较更加合理；（训练集中包含了部分K图中蓝色样本，而蓝色样本代表的CPB也较高，容易与攻击者的高CPB发生混淆）
4. 采用（1）的同样检测方法，但是采用K=4时聚类的结果（说明最优结果对应的轮廓系数即可），提供一个稍逊的检测结果，证明采用轮廓系数选择的合理性；

补充的分析三：静态检测模式

实际检测中添加了一个验证集，即：

* 采用CPB\_FEATSlow训练OCSVM；
* 从CPB\_FEATSleave中按比例随机选择20%的样本作为验证集，帮助OCSVM确定一个最优Recall参数；确定的标准为：
* 使用上述方法确定的最佳参数决定的OCSVM，检测剩余80%的CPB\_FEATSleave，输出Recall与FPRleave；
* 上述结果重复五次，取平均值；
* 上述实验，分别取MIX=1/0.8/0.5三种情况，表示不同选参数策略下对于最终结果的影响；

补充的分析四：动态检测模式

静态检测模式与动态检测模式的最大区别在于：

计算个人社交环境因子时，即考虑通讯特征时，是否考虑了同等的时间跨度。如一个工作2个月后离职的用户与一直在职的用户，其与离职用户建立通信机会是不同的，导致后者可能在通信流特征上高于前者。

* 依据用户大五人格计算其自身的三个CPB-O/P+JS
* 依据用户最常工作期限共事环境的并集计算CPB环境特征
* 动态：每次逐月检测，使得每次提取用户CPB的REL特征均依据于相同时间跨度；
* 每次检测T月离职用户时，采用[1,T-1]月的用户集合进行静态模式检测，且以[1,T-1]静态模式的最优[Recall,FPR]对应的参数作为检测T月的参数，使用T月用户的静态检测特征提取模式，使用[1,T-1]训练得到的OCSVM检测T月离职用户，输出结果；
* 检测T+1月时重复上述过程；
* 上述过程有几个基本假设：

1. 每月离职人数属于小概率事件，约为5%左右，故可以假设相邻月份的CPB特征分布基本不变；故采用上月训练的OCSVM检测下个月；
2. 假设每个月结束时已知当月的真正离职攻击者，即已获得GroundTruth；
3. 存在攻击者离职的第一个月份（2010-06），直接采用最优参数遍历方法得到当月最优Recall与FPR；
4. 本实验时为了证明动态检测可疑缓解静态检测中的CPB计算不同步问题，并没有设置验证集，每一步都使用了自动获知GroundTruth的形式；
5. 当然也可以采用上述描述的T-1月OCSVM检测T月OCSVM，也算是一种动态检测模式，而非单纯的理论验证；（此时又需要分MIX=1/0.8/0,5来进行讨论了，但是结果不易展示，故作罢。）

好了，基本补充分析说明完毕，开始实验。

下午3时28分

首先确认ATF特征格式是否29维度？

user\_id,

CPB-I,CPB-O, 2

JS\_Score, 1

Team\_CPB-I-mean,Team\_CPB-O-mean,Users-less-mean-A,Users-less-mean-A and C,Users-less-mean-C,Users-High-mean-N,Team\_CPB-I-median,Team\_CPB-O-median,

leader-CPB-I,leader-CPB-O, 10

dis\_ocean,avg\_dis\_ocean,dis\_os,avg\_dis\_os, 4

email\_ratio, 1

cnt\_send/recv,cnt\_s/r\_size,cnt\_s/r\_attach,cnt\_s/r\_days,~~cnt\_email\_days（比论文多的一个）~~, 9

cnt\_late\_days,cnt\_early\_days,month\_work\_days, 3

MMK1532,

-6.56,-9.51,

6.76,

-13.794,-20.291933333333333,16,7,12,16,-12.584,-22.037599999999998,-8.712,-14.6

168,12.3121009712,0.286327929563,0.245534276386,

0.00571009945084,

-0.627906976744,0.186046511628,0.860465116279,5859.51162791,208139.372093,0.0,0.186046511628,0.186046511628,0.813953488372,0.953488372093,

169.0,161.0,356.0,

以后可以使用CERT5.2\_Leave\_Static\_CPB\_ATF-03.csv了，其中去掉了email\_days特征，与论文中的29维度特征一致。

新的检测文件为：V09\_KMeans\_OCSVM\_JS\_Predictor\_Class\_Basic\_20190604.py

通过修改实验，发现如果以Recall为单指标，会造成分类器将所有离职-1样本全部判断为+1，得到FPRleave=1，故训练时还是需要进行某种制约，同时考虑到FPRleave

比如：

Nu,Mix,Score,Recall,FPRtotal,Cnt\_FP,FPRleave

[**0.004, 1.0, 0.9855072463768116, 0.9855072463768116, 0.0889942074776198, 169.0, 1.0**, OneClassSVM(cache\_size=200, coef0=0.0, degree=3, gamma='auto', kernel='rbf',

max\_iter=-1, nu=0.004, random\_state=10, shrinking=True, tol=0.01,

verbose=False), 0.9655172413793104, 1.0, 1.0,

故：

可以设定遍历策略：

Mix = range(5,10,1)

遍历公式为：



**以下nu=range(1,1000,1) 每次以0.001变化**

如果仅仅以**Recall-FPRleave**作为选择指标，那么最优：

[0.697, 0.01, 0.08361204013377926, 0.391304347826087, 0.02738283307003686, **52.0**, **0.3076923076923077**, OneClassSVM(cache\_size=200, coef0=0.0, degree=3, gamma='auto', kernel='rbf',

max\_iter=-1, nu=0.697, random\_state=10, shrinking=True, tol=0.01,

verbose=False), 0.3103448275862069, 0.26666666666666666, 1.0,

设置if recall<0.9: continue

[0.014, 0.01, 0.013034902667009685, 0.9420289855072463, 0.08267509215376515, 157.0, 0.9289940828402367, OneClassSVM(cache\_size=200, coef0=0.0, degree=3, gamma='auto', kernel='rbf',

max\_iter=-1, nu=0.014, random\_state=10, shrinking=True, tol=0.01,

verbose=False), **0.9655172413793104, 0.9333333333333333, 0.9**,

if recall < 0.85 or recall > 0.9:

[0.146, 0.01, 0.05239687848383501, **0.8985507246376812**, 0.07530279094260137, **143.0, 0.8461538461538461**, OneClassSVM(cache\_size=200, coef0=0.0, degree=3, gamma='auto', kernel='rbf',

max\_iter=-1, nu=0.146, random\_state=10, shrinking=True, tol=0.01,

verbose=False), 0.896551724137931, 0.8666666666666667, 1.0,

如果考虑fpr<0.8

[0.21, 0.01, 0.8115942028985508, 0.8115942028985508, 0.07109004739336493, 135.0, 0.7988165680473372, OneClassSVM(cache\_size=200, coef0=0.0, degree=3, gamma='auto', kernel='rbf',

max\_iter=-1, nu=0.21, random\_state=10, shrinking=True, tol=0.01,

verbose=False), 0.7586206896551724, 0.8, 1.0,

若Nu调整为10000：

[0.2082, 0.01, 0.8115942028985508, 0.8115942028985508, 0.07109004739336493, 135.0, 0.7988165680473372, OneClassSVM(cache\_size=200, coef0=0.0, degree=3, gamma='auto', kernel='rbf',

max\_iter=-1, nu=0.2082, random\_state=10, shrinking=True, tol=0.01,

verbose=False), 0.7586206896551724, 0.8, 1.0,

若继续考虑mix\*recall - (1-mix)\*fprleave，mix从0.5---0.9，同时令fpr<0.8

[0.221, 0.9, 0.6517365577566248, 0.8115942028985508, 0.07003686150605581, 133.0, 0.7869822485207101, OneClassSVM(cache\_size=200, coef0=0.0, degree=3, gamma='auto', kernel='rbf',

max\_iter=-1, nu=0.221, random\_state=10, shrinking=True, tol=0.01,

verbose=False), 0.7586206896551724, 0.8, 1.0,

**若使用 mscore = recall / exp(fpr)**

去掉了fpr<0.8的条件：

最优：

[0.14, 0.05, 0.3832594417716617, **0.8985507246376812**, 0.07582938388625593, 144.0（也需要修正）, **0.8520710059171598（修正为0.682）**, OneClassSVM(cache\_size=200, coef0=0.0, degree=3, gamma='auto', kernel='rbf',

max\_iter=-1, nu=0.14, random\_state=10, shrinking=True, tol=0.01,

verbose=False), **0.896551724137931, 0.8666666666666667, 1.0**,

[0.146, 0.05, 0.38553397186192373, **0.8985507246376812**, 0.07530279094260137, 143.0, **0.8461538461538461**, OneClassSVM(cache\_size=200, coef0=0.0, degree=3, gamma='auto', kernel='rbf',

max\_iter=-1, nu=0.146, random\_state=10, shrinking=True, tol=0.01,

verbose=False), 0.896551724137931, 0.8666666666666667, 1.0,

二〇一九年六月五日星期三

上午10时7分

leave中有238个用户

其中69个insiders,剩余169个用户为Users

先温习下昨天的实验，看看遍历参数策略时最佳结果的返回条件。

nu\_lst = range(1, 1000, 1)

若：**m\_score = recall / math.exp(fpr\_leave)**为选择条件，则有：

[0.146, 0.05, 0.38553397186192373, **0.8985507246376812(Recall)**, 0.07530279094260137(FPRt), 143.0, **0.8461538461538461(FPRl)**, OneClassSVM(cache\_size=200, coef0=0.0, degree=3, gamma='auto', kernel='rbf',

max\_iter=-1, nu=0.146, random\_state=10, shrinking=True, tol=0.01,

verbose=False), 0.896551724137931, 0.8666666666666667, 1.0,

m\_score = recall / (1 + math.exp(fpr\_leave))结果一样。

若：m\_score = math.exp(recall) / math.log(math.e + fpr\_leave, math.e)

则：筛选条件过度强调了分子Recall的作用，而使得结果十分偏向全部判定为1

[0.004, 0.05, 2.040088860292206, 0.9855072463768116, 0.0889942074776198, 169.0, 1.0, OneClassSVM(cache\_size=200, coef0=0.0, degree=3, gamma='auto', kernel='rbf',

max\_iter=-1, nu=0.004, random\_state=10, shrinking=True, tol=0.01,

verbose=False), 0.9655172413793104, 1.0, 1.0,

若：m\_score = math.log(math.e + recall, math.e) / math.exp(fpr\_leave)

则：筛选结果偏向FPR，最终选择了变化率更大的那个数值

[0.996, 0.05, 0.9941003121481687, 0.0, 0.000526592943654555, 1.0, 0.005917159763313609, OneClassSVM(cache\_size=200, coef0=0.0, degree=3, gamma='auto', kernel='rbf',

max\_iter=-1, nu=0.996, random\_state=10, shrinking=True, tol=0.01,

verbose=False), 0.0, 0.0, 0.0,

若：m\_score = recall / math.log(math.e + fpr\_leave, math.e)

则：

[0.004, 0.05, 0.7504271659970563, 0.9855072463768116, 0.0889942074776198, 169.0, 1.0, OneClassSVM(cache\_size=200, coef0=0.0, degree=3, gamma='auto', kernel='rbf',

max\_iter=-1, nu=0.004, random\_state=10, shrinking=True, tol=0.01,

verbose=False), 0.9655172413793104, 1.0, 1.0,

目前来看，尚未找到除了公式：



之外更合适的遍历条件，暂时以该公式为基准。

上午10时54分

依据上述基准公式，

基础实验1的最佳结果为：

若：**m\_score = recall / math.exp(fpr\_leave)**为选择条件，则有：

[0.146, 0.05, 0.38553397186192373, **0.8985507246376812(Recall)**, 0.07530279094260137(FPRt), 143.0, **0.8461538461538461(FPRl)**, OneClassSVM(cache\_size=200, coef0=0.0, degree=3, gamma='auto', kernel='rbf',

max\_iter=-1, nu=0.146, random\_state=10, shrinking=True, tol=0.01,

verbose=False), 0.896551724137931, 0.8666666666666667, 1.0,

基础实验2的最佳结果为：

即测试集变更为：CPB\_FEATSh+CPB\_FEATSleave：

[0.176, 0.05, 0.38377873122427925, 0.8695652173913043, **0.15376513954713006**, **292.0**, 0.8179271708683473, OneClassSVM(cache\_size=200, coef0=0.0, degree=3, gamma='auto', kernel='rbf',

max\_iter=-1, nu=0.176, random\_state=10, shrinking=True, tol=0.01,

verbose=False), 0.8620689655172413, 0.8333333333333334, 1.0,

表现为：

1. 整体召回率下降： 0.8695652173913043
2. FPRtotal迅速升高：**0.15376513954713006**
3. FPRleave略有下降：0.8179271708683473

基础实验3的最佳结果为：

即训练集变为CPB\_FEATSjob=CPB\_FEATSl+CPB\_FEATSh

此时每次从CPB\_FEATSjob中随机挑出80%训练OCSVM ,20%与CPB\_FEATSleave一起作为测试集

执行五次

五次结果为：**召回率结果一致，大大低于我们的方法（修正误报率差不多）**

0.015, 1.0, 0.3796899288144356, 0.8405797101449275, 0.1590310689836756, 302.0, 0.7947368421052632, OneClassSVM(cache\_size=200, coef0=0.0, degree=3, gamma='auto', kernel='rbf',

max\_iter=-1, nu=0.015, random\_state=10, shrinking=True, tol=0.01,

verbose=False), 0.4482758620689655, 0.6666666666666666, 0.6, 0.0,

[0.026, 1.0, 0.3796899288144356, 0.8405797101449275, 0.1590310689836756, 302.0, 0.7947368421052632, OneClassSVM(cache\_size=200, coef0=0.0, degree=3, gamma='auto', kernel='rbf',

max\_iter=-1, nu=0.026, random\_state=10, shrinking=True, tol=0.01,

verbose=False), 0.4482758620689655, 0.6666666666666666, 0.7, 0.0,

[0.014, 1.0, 0.388274488496169, 0.855072463768116, 0.1579778830963665, 300.0, 0.7894736842105263, OneClassSVM(cache\_size=200, coef0=0.0, degree=3, gamma='auto', kernel='rbf',

max\_iter=-1, nu=0.014, random\_state=10, shrinking=True, tol=0.01,

verbose=False), **0.4482758620689655, 0.6333333333333333, 0.7**, 0.0

[0.027, 1.0, 0.3781983557909577, **0.7536231884057971, 0.13796735123749343, 262.0, 0.6894736842105263**, OneClassSVM(cache\_size=200, coef0=0.0, degree=3, gamma='auto', kernel='rbf',

max\_iter=-1, nu=0.027, random\_state=10, shrinking=True, tol=0.01,

verbose=False), **0.4482758620689655, 0.5333333333333333, 0.6**, 0.0,

由于以攻击者与正常用户划分测试集与训练集，故而只能采取上述方式

下午2时5分

接下来需要static检测模式，即引入了验证集来自动化训练集，然后测试剩余部分的结果：

即只需要提供极少部分的已知用户数据作为验证即可。

依旧采用CPB\_FEATSlow训练OCSVM，但是从CPB\_FEATSleave中选出20%的比例用户用于训练OCSVM得到最佳参数，然后检测剩余80%。

开始依次记录五次结果：

第一次

[0.001, 0.005, 0.3831479923485084, 0.7692307692307693, 0.012111637704054766, 23.0, 0.696969696969697, OneClassSVM(cache\_size=200, coef0=0.0, degree=3, gamma='auto', kernel='rbf',

max\_iter=-1, nu=0.001, random\_state=10, shrinking=True, tol=0.01,

verbose=False), 0.5, 0.8333333333333334, 1.0,

FPR total is 0.0494997367035

Count FPR is 94.0

FPR leave is 0.691176470588

Recall is 0.785714285714

Recall for 1 is 0.84

Recall for 2 is 0.875

Recall for 3 is 0.285714285714

第二次

0.178, 0.005, 0.4344298699506548, 0.8461538461538461, 0.01158504476040021, 22.0, 0.6666666666666666, OneClassSVM(cache\_size=200, coef0=0.0, degree=3, gamma='auto', kernel='rbf',

max\_iter=-1, nu=0.178, random\_state=10, shrinking=True, tol=0.01,

verbose=False), 0.75, 0.875, 1.0,

FPR total is 0.0610847814639

Count FPR is 116.0

FPR leave is 0.852941176471

Recall is 0.875

Recall for 1 is 0.84

Recall for 2 is 0.863636363636

Recall for 3 is 1.0

第三次

[0.272, 0.005, 0.4344298699506548, 0.8461538461538461, 0.01158504476040021, 22.0, 0.6666666666666666, OneClassSVM(cache\_size=200, coef0=0.0, degree=3, gamma='auto', kernel='rbf',

max\_iter=-1, nu=0.272, random\_state=10, shrinking=True, tol=0.01,

verbose=False), 0.7142857142857143, 1.0, 1.0

FPR total is 0.0521327014218

Count FPR is 99.0

FPR leave is 0.727941176471

Recall is 0.75

Recall for 1 is 0.681818181818

Recall for 2 is 0.730769230769

Recall for 3 is 1.0

第四次

0.008, 0.005, 0.44123316775998395, 1.0, 0.014218009478672985, 27.0, 0.8181818181818182, OneClassSVM(cache\_size=200, coef0=0.0, degree=3, gamma='auto', kernel='rbf',

max\_iter=-1, nu=0.008, random\_state=10, shrinking=True, tol=0.01,

verbose=False), 1.0, 1.0, 1.0,

FPR total is 0.0631911532385

Count FPR is 120.0

FPR leave is 0.882352941176

Recall is 0.839285714286

Recall for 1 is 0.95652173913

Recall for 2 is 0.88

Recall for 3 is 0.375

第五次

0.004, 0.005, 0.3791979291581653, 1.0, 0.01685097419694576, 32.0, 0.9696969696969697, OneClassSVM(cache\_size=200, coef0=0.0, degree=3, gamma='auto', kernel='rbf',

max\_iter=-1, nu=0.004, random\_state=10, shrinking=True, tol=0.01,

verbose=False), 1.0, 1.0, 1.0,

FPR total is 0.071616640337

Count FPR is 136.0

FPR leave is 1.0

Recall is 1.0

Recall for 1 is 1.0

Recall for 2 is 1.0

Recall for 3 is 1.0

第六次

[0.178, 0.005, 0.40729215485536985, 0.9230769230769231, 0.014218009478672985, 27.0, 0.8181818181818182, OneClassSVM(cache\_size=200, coef0=0.0, degree=3, gamma='auto', kernel='rbf',

max\_iter=-1, nu=0.178, random\_state=10, shrinking=True, tol=0.01,

verbose=False), 1.0, 0.8333333333333334, 1.0,

FPR total is 0.059505002633

Count FPR is 113.0

FPR leave is 0.830882352941

Recall is 0.821428571429

Recall for 1 is 0.782608695652

Recall for 2 is 0.791666666667

Recall for 3 is 1.0

以上，可以选择：5折交叉验证

Recall = 87.5%

Recall\_1 = 82.6%

Recall\_2 = 88.3%

Recall\_3 = 91.5%

FPRtotal = 5.9%

FPRleave = 65.3%

下午3时39分

再来考虑K=4时的聚类结果

针对JS\_lst的最佳自动K值为[2:10] 4 : 0.2018479216755096

由于当时写的代码太乱，放弃，转而直接使用一个较差结果即可，从之前的结果中选一个差的即可。

1-29

2-30

3-10

User-169

根据基础实验结果选择一个较差的即可；说明应当用轮廓系数选择K。

最后一个动态实验，更新一下当初的结果数值即可：

还是选择用该图即可。

新的结果，综合考虑：

Recall 1: 29个用户检测出26个（0.89655），遗漏3个

Recall 2: 30个用户检测出29个（0.96666），遗漏1个

Recall 3: 10个用户检测出9个（0.9），遗漏1个

新结果

2011-05

[0.265, 1.0, 0.5292133415000503, **1.0, 0.003686150605581885, 7.0, 0.6363636363636364**, OneClassSVM(cache\_size=200, coef0=0.0, degree=3, gamma='auto', kernel='rbf',

max\_iter=-1, nu=0.265, random\_state=10, shrinking=True, tol=0.01,

verbose=False), **1.0, 1.0, 1.0**,

Insiders\_1 in Risk\_Users\_Sort:...

['VAH1292', 1, 1, '0.4064558760385353'] : 10

Insiders\_2 in Risk\_Users\_Sort:...

['TMT0851', 1, 1, '5.256599351130717'] : 3

Insiders\_3 in Risk\_Users\_Sort:...

['CWW1120', 1, 1, '5.650844732527958'] : 2

['LAH0463', 1, 1, '2.656260815585597'] : 7

2011-05 分析完毕..

2011-04

[0.465, 1.0, 0.6549846024623855, **0.8, 0.00105318588730911, 2.0, 0.2**, OneClassSVM(cache\_size=200, coef0=0.0, degree=3, gamma='auto', kernel='rbf',

max\_iter=-1, nu=0.465, random\_state=10, shrinking=True, tol=0.01,

verbose=False), **0.5, 1.0, 1.0**,

Insiders\_1 in Risk\_Users\_Sort:...（场景1掉了一个）

['ETW0002', -1, 1, '-39.867981928225504'] : 11

['WWW0701', 1, 1, '6.028385366072598'] : 2

Insiders\_2 in Risk\_Users\_Sort:...

['HSN0675', 1, 1, '6.834776439249822'] : 1

['OKM1092', 1, 1, '3.0245989027899896'] : 3

Insiders\_3 in Risk\_Users\_Sort:...

['ZEH0685', 1, 1, '0.21907977521209432'] : 5

2011-04 分析完毕..

2011-03

[0.072, 1.0, 0.4168620196785084, 1.0, 0.003686150605581885, 7.0, 0.875, OneClassSVM(cache\_size=200, coef0=0.0, degree=3, gamma='auto', kernel='rbf',

max\_iter=-1, nu=0.072, random\_state=10, shrinking=True, tol=0.01,

verbose=False), **1.0, None, None**,

Insiders\_1 in Risk\_Users\_Sort:...

['ELT1370', 1, 1, '1.9491496004806521'] : 6

['JUP1472', 1, 1, '1.344983390955889'] : 8

['NIV1608', 1, 1, '0.05397364454000808'] : 12

['TMC0934', 1, 1, '3.223887298393649'] : 2

['WHB1247', 1, 1, '0.3783781731743687'] : 11

['WSK1857', 1, 1, '2.6605892806460076'] : 3

Insiders\_2 in Risk\_Users\_Sort:...

Insiders\_3 in Risk\_Users\_Sort:...

2011-03 分析完毕..

2011-02

[0.001, 1.0, 0.36787944117144233, **1.0, 0.004739336492890996, 9.0, 1.0**, OneClassSVM(cache\_size=200, coef0=0.0, degree=3, gamma='auto', kernel='rbf',

max\_iter=-1, nu=0.001, random\_state=10, shrinking=True, tol=0.01,

verbose=False), **1.0, 1.0, 1.0**,

Insiders\_1 in Risk\_Users\_Sort:...

['ALW0764', 1, 1, '0.038312752083368595'] : 8

['PTH0005', 1, 1, '0.013967129289790492'] : 15

['ZKP0542', 1, 1, '0.062018110477127575'] : 2

Insiders\_2 in Risk\_Users\_Sort:...

['ITA0159', 1, 1, '0.017180273397010337'] : 14

['JAL0811', 1, 1, '0.06426459251844863'] : 1

['SNK1280', 1, 1, '0.02870857810739036'] : 11

Insiders\_3 in Risk\_Users\_Sort:...

['KCM0466', 1, 1, '0.03459402680703372'] : 9

2011-02 分析完毕..

2011-01

[0.238, 1.0, 0.513417119032592, **1.0, 0.00315955766192733, 6.0, 0.6666666666666666**, OneClassSVM(cache\_size=200, coef0=0.0, degree=3, gamma='auto', kernel='rbf',

max\_iter=-1, nu=0.238, random\_state=10, shrinking=True, tol=0.01,

verbose=False), **None, 1.0, None**,

Insiders\_1 in Risk\_Users\_Sort:...

Insiders\_2 in Risk\_Users\_Sort:..

['DCC1119', 1, 1, '3.111390853226794'] : 5

['LVF1626', 1, 1, '7.0975298125068775'] : 2

['MGB1235', 1, 1, '1.3984629699386915'] : 8

Insiders\_3 in Risk\_Users\_Sort:...

2011-01 分析完毕..

2011-02

[0.037, 1.0, 0.42437284567695, **1.0, 0.00631911532385466, 12.0, 0.8571428571428571**, OneClassSVM(cache\_size=200, coef0=0.0, degree=3, gamma='auto', kernel='rbf',

max\_iter=-1, nu=0.037, random\_state=10, shrinking=True, tol=0.01,

verbose=False), **1.0, 1.0, 1.0**,

Insiders\_1 in Risk\_Users\_Sort:...

['FZG0389', 1, 1, '0.11733160496140016'] : 19

Insiders\_2 in Risk\_Users\_Sort:...

['BYO1846', 1, 1, '0.9396961398710459'] : 11

['HIS1394', 1, 1, '1.5853352057085104'] : 7

['HMS1658', 1, 1, '0.42599880821220637'] : 17

['HXP0976', 1, 1, '0.8313873548823238'] : 12

['ICB1354', 1, 1, '0.46590084165994483'] : 14

['RRS0056', 1, 1, '0.35879298185343345'] : 18

Insiders\_3 in Risk\_Users\_Sort:...

['ACA1126', 1, 1, '0.44641882814553213'] : 15

2010-12 分析完毕..

2010-11

[0.055, 1.0, 0.4345982085070782, **1.0, 0.0052659294365455505, 10.0, 0.8333333333333334**, OneClassSVM(cache\_size=200, coef0=0.0, degree=3, gamma='auto', kernel='rbf',

max\_iter=-1, nu=0.055, random\_state=10, shrinking=True, tol=0.01,

verbose=False), **1.0, 1.0, 1.0**

Insiders\_1 in Risk\_Users\_Sort:...

['AYG1697', 1, 1, '0.9901433131844044'] : 11

['ISW0738', 1, 1, '2.020710723482196'] : 3

['MIB0203', 1, 1, '1.3806767879031696'] : 9

['REF1924', 1, 1, '2.0046902720480517'] : 4

Insiders\_2 in Risk\_Users\_Sort:...

['NAH1366', 1, 1, '0.06519223911951144'] : 15

Insiders\_3 in Risk\_Users\_Sort:...

['GKW0043', 1, 1, '1.8086316971060512'] : 6

2010-11 分析完毕..

2010-10

[0.303, 1.0, 0.4840441045780793, **0.8571428571428571, 0.00210637177461822, 4.0, 0.5714285714285714**, OneClassSVM(cache\_size=200, coef0=0.0, degree=3, gamma='auto', kernel='rbf',

max\_iter=-1, nu=0.303, random\_state=10, shrinking=True, tol=0.01,

verbose=False), **0.5, 1.0, 1.0**

Insiders\_1 in Risk\_Users\_Sort:...（场景1遗漏一个）

['DNJ0740', 1, 1, '9.930318253950531'] : 1

['JKB0287', -1, 1, '-13.777967924549273'] : 13

Insiders\_2 in Risk\_Users\_Sort:...

['CHP1711', 1, 1, '1.2095931573350072'] : 8

['GWG0497', 1, 1, '6.665083620719884'] : 6

['KSS1005', 1, 1, '8.208070548927367'] : 4

['MCP0611', 1, 1, '0.48814367260956715'] : 9

Insiders\_3 in Risk\_Users\_Sort:...

['ELM1123', 1, 1, '10.720305029036126'] : 0

2010-10 分析完毕..

2010-09

[0.194, 1.0, 0.4251298974669132, **0.9, 0.004739336492890996, 9.0, 0.75**, OneClassSVM(cache\_size=200, coef0=0.0, degree=3, gamma='auto', kernel='rbf',

max\_iter=-1, nu=0.194, random\_state=10, shrinking=True, tol=0.01,

verbose=False), **1.0, 0.8571428571428571, None**

Insiders\_1 in Risk\_Users\_Sort:...

['ALT1465', 1, 1, '1.1596158795592828'] : 17

['IHC0561', 1, 1, '10.099990859301101'] : 0

['SLL0193', 1, 1, '3.6154916986787953'] : 11

Insiders\_2 in Risk\_Users\_Sort:... (7个检测出6个，丢失1个)

['CIF1430', 1, 1, '3.039358333693258'] : 13

['MDS0680', 1, 1, '8.389999037341227'] : 1

['OSS1463', 1, 1, '4.858557180648347'] : 5

['SIS0042', 1, 1, '2.2837726901650512'] : 14

['TNB1616', 1, 1, '2.011865917736884'] : 15

['TRC1838', 1, 1, '1.402773092271957'] : 16

['WDT1634', -1, 1, '-1.9504318011479995'] : 20

Insiders\_3 in Risk\_Users\_Sort:...

2010-09 分析完毕..

2010-08

[0.267, 1.0, 0.4303150655530746, **0.75, 0.0026329647182727752, 5.0, 0.5555555555555556**, OneClassSVM(cache\_size=200, coef0=0.0, degree=3, gamma='auto', kernel='rbf',

max\_iter=-1, nu=0.267, random\_state=10, shrinking=True, tol=0.01,

verbose=False), **0.75, 1.0, 0.0**,

Insiders\_1 in Risk\_Users\_Sort:...（丢失1个）

['EPG1196', 1, 1, '7.921625520903035'] : 3

['KBC1390', 1, 1, '2.417004314180417'] : 6

['PBC0077', -1, 1, '-4.588311445040631'] : 16

['SAF1942', 1, 1, '2.10492454210862'] : 7

Insiders\_2 in Risk\_Users\_Sort:...

['CKP0630', 1, 1, '1.9994161576054807'] : 8

['VCF1602', 1, 1, '0.19880200455844488'] : 10

['ZIE0741', 1, 1, '8.557962594129094'] : 2

Insiders\_3 in Risk\_Users\_Sort:...（丢失1个）

['VRP0267', -1, 1, '-3.954934559490539'] : 15

2010-08 分析完毕..

2010-07

[0.283, 1.0, 0.5737534207374327, **1.0, 0.0026329647182727752, 5.0**, **0.5555555555555556**, OneClassSVM(cache\_size=200, coef0=0.0, degree=3, gamma='auto', kernel='rbf',

max\_iter=-1, nu=0.283, random\_state=10, shrinking=True, tol=0.01,

verbose=False), **1.0, None, 1.0**,

Insiders\_1 in Risk\_Users\_Sort:...

['DAS1320', 1, 1, '0.759575027116739'] : 7

['GFM1815', 1, 1, '1.23724729338727'] : 6

['KEW0198', 1, 1, '2.959452071931537'] : 4

Insiders\_2 in Risk\_Users\_Sort:...

Insiders\_3 in Risk\_Users\_Sort:...

['CRD0272', 1, 1, '0.09370708774338254'] : 8

2010-07 分析完毕..

2010-06

[0.528, 1.0, 0.5889513097505534, **1.0, 0.004739336492890996, 9.0, 0.5294117647058824**, OneClassSVM(cache\_size=200, coef0=0.0, degree=3, gamma='auto', kernel='rbf',

max\_iter=-1, nu=0.528, random\_state=10, shrinking=True, tol=0.01,

verbose=False), **None, None, 1.0**, array([ 1, -1, 1, 1, 1, -1, -1, 1, -1, 1, 1, -1, 1, -1, -1, -1, 1,

1], dtype=int64)]

Insiders\_1 in Risk\_Users\_Sort:...

Insiders\_2 in Risk\_Users\_Sort:...

Insiders\_3 in Risk\_Users\_Sort:...

['MPF0690', 1, 1, '0.13523417716680797'] : 9

2010-06 分析完毕..

补充结束，开始修改论文。

加油！