2018年11月20日星期二

下午4时22分

在今天早上的实验中，自己尝试在使用传统的RLV人际关联影响分析前，首先引入了从当月用户出勤表现中筛选出低满意度用户，其假设来源于：工作满意度高的用户一定工作认真，其CPB-O低，从而其上下班更守时。

我们的第一个实验依据现有的上下班统计特征：

JBI1134,6.5,19.0,6.0,0.0,25,

上述特征分别对应于字段：user\_id, workon-time, workoff-time, late-days, early-days, all-work-days

第一个实验计算ELV（迟到早退指数）时使用的是当月该用户迟到比例+早退比例，按照该方法分析，对于30个Insiders\_2有结果：



2010-8离职用户，需要分析2010-07出勤数据

Last\_ELV\_mean: JKB1843,0.5072463768115942(1082）

*Feat: JKB1843,8.5,16.0,10.0,0.0,23,*

均值命中2/3

Last\_ELV\_meadian: IDB0843,0.6049382716049383(936)

中位数命中2/3

VCF1602,2010-08-20,

VCF1602,0.6481481481481483,(877)

CKP0630,2010-08-26,

CKP0630,0.4365079365079365,(1187)

*Feat: CKP0630,8.0,16.0,7.0,1.0,21,*

分析可以发现，JKB23天中迟到天数为10天，而CKP21天中迟到7天

ZIE0741,2010-08-27,

ZIE0741,0.8253968253968254,(405)

2010-8月分析，预测9月

Last\_ELV\_Mean:ILJ0929,0.5(1076) 命中4/7

Last\_ELV\_Median: PMB1928,0.5852534562211981(926) 命中3/7

SIS0042,2010-09-02,

SIS0042,0.3636363636363636,(1222)

分析该用户在2010-08的表现，

SIS0042,6.5,19.0,7.0,0.0,22, 仅有7天迟到，没有早退

而最后一名Last\_ELV\_Mean

ILJ0929,9.0,16.0,0.0,**11.0**,22,

虽然没有迟到，但是有11天早退；

而对于另一个攻击者

TRC1838,8.5,16.0,0.0,0.0,22,

该月竟然无早退 无迟到，因此看来单独分析一个月并不足够反应该用户的情况；

验证是否可以计算累积迟到早退比例，先来看看2010-7

ILJ0929,9.0,16.0,0.0,**21.0**,21,

SIS0042,6.5,19.0,6.0,0.0,21, SIS0042,6.5,17.0,6.0,0.0,21,

新的SIS0042数据为：

SIS0042,WorkOn:6.5,WorkOff:16.5

TRC1838,8.5,16.0,0.0,0.0,21,

请注意，ILJ0929的下班时间16：00计算错误，其大部分下班时间都在16：00之前

再看2010-06

TRC1838,8.5,16.0,0.0,0.0,22,

ILJ0929,9.0,16.0,0.0,6.0,22,

SIS0042,6.5,19.0,11.0,0.0,22,

我们

TNB1616,2010-09-10,

TNB1616,0.6753246753246753,(731)

TRC1838,2010-09-15,

TRC1838,0.0,(1654)

MDS0680,2010-09-17,

MDS0680,0.0,(1590)

WDT1634,2010-09-20,

WDT1634,1.0064935064935063,(88)

OSS1463,2010-09-21,

OSS1463,0.5714285714285714,(966)

CIF1430,2010-09-23,

CIF1430,0.6103896103896104,(892)

MCP0611,2010-10-06,

CHP1711,2010-10-13,

GWG0497,2010-10-15,

KSS1005,2010-10-16,

NAH1366,2010-11-17,

RRS0056,2010-12-10,

ICB1354,2010-12-15,

BYO1846,2010-12-15,

HXP0976,2010-12-20,

HMS1658,2010-12-30,

HIS1394,2010-12-30,

LVF1626,2011-01-14,

MGB1235,2011-01-21,

DCC1119,2011-01-26,

SNK1280,2011-02-11,

ITA0159,2011-02-17,

JAL0811,2011-02-25,

OKM1092,2011-04-29,

HSN0675,2011-04-29,

TMT0851,2011-05-11,

2018年11月20日星期二

下午6时1分

继续下午的实验。

自己分析数据时，突然发现最初确定用户上下班时间的算法有错误，比如2010-01月份

ILJ0929,9.0,16.0,0.0,20.0,20,

而这个月该用户的下班时间明显都在16：00之前，其上班时间确定为9：00没问题，而下班时间应该为15：30

Line 2630: 01/04/2010 08:40:00,ILJ0929,PC-8384,Logon,

Line 3622: 01/04/2010 15:48:00,ILJ0929,PC-8384,Logoff,

Line 7929: 01/05/2010 08:53:00,ILJ0929,PC-8384,Logon,

Line 8879: 01/05/2010 15:34:00,ILJ0929,PC-8384,Logoff,

Line 13202: 01/06/2010 08:50:00,ILJ0929,PC-8384,Logon,

Line 13995: 01/06/2010 15:46:00,ILJ0929,PC-8384,Logoff,

Line 18332: 01/07/2010 08:46:00,ILJ0929,PC-8384,Logon,

Line 19462: 01/07/2010 15:42:00,ILJ0929,PC-8384,Logoff,

Line 23511: 01/08/2010 08:42:00,ILJ0929,PC-8384,Logon,

Line 24538: 01/08/2010 15:30:00,ILJ0929,PC-8384,Logoff,

通过将下班时间提前的方法，可以修正上述错误，即统计下班时间时，不再从16：00开始，而是提前两个小时到14：00，从而充分考虑灵活上下班的影响。

ILJ0929,WorkOn:9.0,WorkOff:15.5

关于SIS0042的几点说明：

1. 其职业为ITAdmin，因此经常在上班前、下班后登录其他用户机器
2. 其自己的登录时间十分规范，8：30-16：30是其自己的上下班时间，其他时间登录都算职业决定的加班，不作为考勤数据；
3. 我们通过在划分0.5小时长度时间窗口的方法，可以很好地区分上述用户的工作时间规律，正确识别出了其正确的上下班时间SIS0042,WorkOn:6.5,WorkOff:16.5

2018年11月20日星期二

下午9时37分

重新计算的出勤特征还在运行，先投机运行分析了2010-07的数据，先来看看Insiders\_2的表现；

2010-07

ELV\_On\_Mean: LSN1462,0.4827586206896552 (1035)

ELV\_On\_Median: RJW1406,0.5416666666666666 (970)

VCF1602,2010-08-20,

VCF1602,0.6481481481481483,(829)

CKP0630,2010-08-26,

CKP0630,0.3888888888888889,(1115)

ZIE0741,2010-08-27,

ZIE0741,0.8968253968253969,(285)

由于CKP用户在均值与中位数之外，故进行分析

2010-07月份相关用户的出勤特征：

CKP0630,8.0,15.5,**7.0,**0.0,21, （2010-07月份中，21天工作，**7天迟到**，无早退）

LSN1462,9.0,17.5,12.0,0.0,29, （2010-07月份中，29天工作，12天迟到，无早退）

RJW1406,8.0,18.0,9.0,1.0,24, （2010-07月份中，24天工作，9天迟到，1天早退）

IBR0131,0.4827586206896552

IBR0131,8.0,17.0,12.0,0.0,29, （2010-07月份中，29天工作，12天迟到，无早退）

貌似单独看当月的出勤记录，无法进一步提升Insiders的排序，考虑下联合2010-06

2010-06：

CKP0630,8.0,15.5,7.0,0.0,22, （2010-06月份中，22天工作，7天迟到，无早退）

VCF1602,9.0,19.0,11.0,0.0,27, （2010-06月份中，27天工作，11天迟到，无早退）

ZIE0741,8.0,19.0,8.0,2.0,22, （2010-06月份中， 22天工作，8天迟到，2天早退）

LSN1462,9.0,17.5,18.0,0.0,30, （2010-06月份中，30天工作，18天迟到）

RJW1406,8.0,18.0,6.0,1.0,24, （2010-06月份中，24天工作，6天迟到，1天早退）

IBR0131,8.0,17.0,13.0,1.0,30, （2010-06月份中，30天工作，13天迟到，1天早退）

ERD（迟到早退指数）：

CKP：0.31818 + 0.33

VCF：0.407

ZIE：0.454545

LSN：0.6

RJW：0.291 + 0.416

IBR：0.46666

问题依旧不明显，如果进一步考虑前三个月，即包括2010-05

CKP0630,8.0,15.5,0.0,0.0,20, （2010-05月份中，20天工作，无早退无迟到）

VCF1602,9.0,19.0,9.0,3.0,29, （2010-05月份中，29天工作，9天迟到，3天早退）

LSN1462,9.0,17.5,15.0,0.0,29, （2010-05月份中， 29天工作，15天迟到，无早退）

ZIE0741,8.0,19.0,7.0,0.0,20, （2010-05月份中，20天工作，7天迟到，无早退）

RJW1406,8.0,18.0,5.0,0.0,25,

IBR0131,8.0,17.0,18.0,0.0,31,

小结：

现在的情况综合来看，也许采用出勤率进行第一轮过滤不能作为很好的区分器，自己开始所想的可以过滤掉一部分用户，剩余的用户的出勤率表现十分差。

还是以LSN用户为例，该用户的上班时间是否应该由其同组的用户来确定呢？

即所有用户共享一个共同的上下班时间？

看看与LSN的同组用户ACB1485

通过查阅资料，美国上班的时间一般是早上9：00-下午17：00，因此，凡是晚于9：00登录或者早于17：00登出的都认为是迟到或者早退。

SIS0042,2010-09-02,

TNB1616,2010-09-10,

TRC1838,2010-09-15,

MDS0680,2010-09-17,

WDT1634,2010-09-20,

OSS1463,2010-09-21,

CIF1430,2010-09-23,

怀疑为何LSN用户每月迟到天数那么多？

是否最初的统计有问题？

分析LSN用户在2010-01月份的记录：

9点以前上班：14

9点以后上班：14

01/02/2010 08:52:00,LSN1462,PC-2754,Logon,

01/02/2010 18:08:00,LSN1462,PC-2754,Logoff,

01/03/2010 09:12:00,LSN1462,PC-2754,Logon,

01/03/2010 17:52:00,LSN1462,PC-2754,Logoff,

01/04/2010 08:58:00,LSN1462,PC-2754,Logon,

01/04/2010 17:50:00,LSN1462,PC-2754,Logoff,

01/05/2010 09:02:00,LSN1462,PC-2754,Logon,

01/05/2010 17:57:00,LSN1462,PC-2754,Logoff,

01/06/2010 08:46:00,LSN1462,PC-2754,Logon,

01/06/2010 17:50:00,LSN1462,PC-2754,Logoff,

01/07/2010 08:46:00,LSN1462,PC-2754,Logon,

01/07/2010 18:02:00,LSN1462,PC-2754,Logoff,

01/08/2010 08:52:00,LSN1462,PC-2754,Logon,

01/08/2010 18:13:00,LSN1462,PC-2754,Logoff,

01/10/2010 09:07:00,LSN1462,PC-2754,Logon,

01/10/2010 17:53:00,LSN1462,PC-2754,Logoff,

01/11/2010 08:55:00,LSN1462,PC-2754,Logon,

01/11/2010 18:13:00,LSN1462,PC-2754,Logoff,

01/12/2010 09:02:00,LSN1462,PC-2754,Logon,

01/12/2010 17:52:00,LSN1462,PC-2754,Logoff,

01/13/2010 08:58:00,LSN1462,PC-2754,Logon,

01/13/2010 17:56:00,LSN1462,PC-2754,Logoff,

01/14/2010 08:58:00,LSN1462,PC-2754,Logon,

01/14/2010 18:09:00,LSN1462,PC-2754,Logoff,

01/15/2010 08:50:00,LSN1462,PC-2754,Logon,

01/15/2010 18:05:00,LSN1462,PC-2754,Logoff,

01/16/2010 09:09:00,LSN1462,PC-2754,Logon,

01/16/2010 18:00:00,LSN1462,PC-2754,Logoff,

01/17/2010 08:52:00,LSN1462,PC-2754,Logon,

01/17/2010 18:06:00,LSN1462,PC-2754,Logoff,

01/18/2010 08:54:00,LSN1462,PC-2754,Logon,

01/18/2010 18:00:00,LSN1462,PC-2754,Logoff,

01/19/2010 08:49:00,LSN1462,PC-2754,Logon,

01/19/2010 17:58:00,LSN1462,PC-2754,Logoff,

01/20/2010 08:49:00,LSN1462,PC-2754,Logon,

01/20/2010 18:12:00,LSN1462,PC-2754,Logoff,

01/21/2010 09:00:00,LSN1462,PC-2754,Logon,

01/21/2010 18:15:00,LSN1462,PC-2754,Logoff,

01/22/2010 09:04:00,LSN1462,PC-2754,Logon,

01/22/2010 17:57:00,LSN1462,PC-2754,Logoff,

01/23/2010 09:11:00,LSN1462,PC-2754,Logon,

01/23/2010 18:10:00,LSN1462,PC-2754,Logoff,

01/24/2010 09:01:00,LSN1462,PC-2754,Logon,

01/24/2010 17:53:00,LSN1462,PC-2754,Logoff,

01/25/2010 09:09:00,LSN1462,PC-2754,Logon,

01/25/2010 17:59:00,LSN1462,PC-2754,Logoff,

01/26/2010 09:00:00,LSN1462,PC-2754,Logon,

01/26/2010 17:57:00,LSN1462,PC-2754,Logoff,

01/27/2010 09:02:00,LSN1462,PC-2754,Logon,

01/27/2010 17:50:00,LSN1462,PC-2754,Logoff,

01/28/2010 08:55:00,LSN1462,PC-2754,Logon,

01/28/2010 17:58:00,LSN1462,PC-2754,Logoff,

01/29/2010 09:11:00,LSN1462,PC-2754,Logon,

01/29/2010 17:50:00,LSN1462,PC-2754,Logoff,

01/31/2010 09:12:00,LSN1462,PC-2754,Logon,

01/31/2010 17:53:00,LSN1462,PC-2754,Logoff,

2018年11月21日星期三

下午3时46分

上午发现了现在统计出勤率的问题：

1. 出勤率取决于上下班时间，而一个企业的上下班时间通常是固定的，因而不太可能因人而异，就美国而言，通用上下班时间是9：00-17：00
2. 在统计某个用户2010-01月份登入登出时间时，由于假设用户大多情况是遵守上班时间的，因此选取的是“基于时间段上下班频率的投票算法”，由此选择的是登入/登出频率最高的时间段，并将该时间段的终点/起点作为上下班时间。一般而言该算法没有问题，但是遇到了奇葩用户LSN，该用户8.5前与9.0后的上班频率几乎一样，此时凭借微笑差距选择8.5而非9.0导致大量工作日被判定为上班迟到；

基于上述考虑，决定使用美国固定的上下班时间（早9：00，下午17：00）作为标准对于CERT5.2中每个月份用户上班迟到与下班早退的天数进行统计。

* 突然发现之前计算迟到早退特征时，计算早退时+=错写成了=，可能对用户的迟到天数统计有影响，在所机器上重新跑机器

重新使用标准上下班时间统计2010-01

Standard: JBI1134,9.0,17.0,0.0,10.0,0.0,0.4,25,

Old: JBI1134,6.5,16.0,6.0,0.0,25,

下午5时8分

开始基于标准上下班时间重新针对CERT5.2用户月度数据分析

笔记上分析的7月份--2011-06

所机器上跑的2010-01：2011-06

下午8时45分

一个半小时后，2010-07分析完毕，我们来初步看看结果。

Over\_Mean: XNC1338,0.4230769230769231【883】

Over\_Median: MKG0606,0.125【970】

**VCF1602,2010-08-20,**

**Feat: VCF1602,9.0,17.0,15.0,0.0,0.555555555556,0.0,27,**

**VCF1602,0.5555555555555556,【818】**

**CKP0630,2010-08-26, （验证通过）**

**Feat: CKP0630,9.0,17.0,0.0,21.0,0.0,1.0,21,**

**CKP0630,1.0,【164】**

ZIE0741,2010-08-27,

Feat: ZIE0741,9.0,17.0,0.0,0.0,0.0,0.0,21, (按照标准上班时间，ZIE用户确实在2010-07月份没有迟到早退)

ZIE0741,0.0,(1394)

* 我们通过场景与数据分析发现了端倪：
* ZIE用户在2010-07月准备跳槽，因此开始提前到单位，晚下班开始访问竞聘HTTP，发邮件以及考取文件到USB，因此这段时间该用户的上下班时间是正常的；
* 需要下一步分析其之前的月份，比如8月份之前的5月-6月-7月，发现其是否存在出勤率突然下降的情况

我们先来看看2010-09月离职的用户在2010-08月份的表现：7个用户5个完全中位数命中，1个用户当月缺勤0，1个用户差一点进入该用户范围

SIS0042,2010-09-02,

2010-07： SIS0042,9.0,17.0,0.0,8.0,0.0,0.380952380952,21,

2010-07：ERD（迟到早退指数）【**SIS0042,0.38095238095238093,**】【893】在2010-07月份的中位数之上

Over\_Mean: NJB0011,0.4230769230769231[858]

Over\_Median: DAR0139,0.13636363636363635[963]

TNB1616,2010-09-10,

Feat: TNB1616,9.0,17.0,13.0,0.0,0.590909090909,0.0,22,

**TNB1616,0.5909090909090909,[742]**

TRC1838,2010-09-15,

Feat：TRC1838,9.0,17.0,0.0,22.0,0.0,1.0,22,

**TRC1838,1.0,【434】**

MDS0680,2010-09-17,

Feat: MDS0680,9.0,17.0,0.0,0.0,0.0,0.0,22,

MDS0680,0.0,[1295]

WDT1634,2010-09-20,

Feat: WDT1634,9.0,17.0,2.0,0.0,0.0909090909091,0.0,22,

WDT1634,0.09090909090909091,[970]

2010-07:

对于该用户而言，当月的ELD指标：可以看到，970位置的WDT1634距离DAR0139只有几个数

DAR0139,0.13636363636363635,

VRP0267,0.125,

EBA0758,0.09090909090909091,

DDW0757,0.09090909090909091,

MMR1502,0.09090909090909091,

HGG0535,0.09090909090909091,

HFM0465,0.09090909090909091,

WDT1634,0.09090909090909091,

OSS1463,2010-09-21,

Feat: OSS1463,9.0,17.0,13.0,0.0,0.5,0.0,26,

**OSS1463,0.5, 【824】**

CIF1430,2010-09-23,

Feat: CIF1430,9.0,17.0,0.0,22.0,0.0,1.0,22,

**CIF1430,1.0,【359】**

下午10时44分

睡前小结一下：

1. 使用标准时间后，检测Insiders2的效果有所上升；
2. 检测的目标私以为应当确定为检测出“场景二的离职攻击者”，在此基础上，如果检测的结果针对其他内部攻击者或者普通离职用户也有效果的话最好，作为附加效果说明即可；
3. 如同测血糖，需要当月以及糖化蛋白两种，因此要想全面反映一个用户的出勤特征，当月特征是一部分，截止到当月的ERD的均值也很重要，一个初步的想法是，判断一个用户某月的ERD，其结果应该有由：当月ERD与之前月份的均值ERD组合而成，希望上述公式可以解决某些用户在最后几个月因窃取数据需要早到晚走造成的出勤完美假象（Work Perfect Dream）



1. 明天需要首先分析、探查所有30个Insiders\_2在离职前一个月的ERD排序；紧接着实现上述符合的ERD计算方法，查看对于结果是否有改进；（小部分样本实验推进，一旦结果不好，即可检查修改）

2018年11月23日星期五

下午6时12分

完成今天早上的实验，首先，考虑原始的，基于标准时间计算迟到早退行为，并仅仅使用离职前一个月的数据进行预测的效果。

..<<统计结果>>..

单月预测，Over\_Mean命中： 16 0.533333333333

单月预测，Over\_Median命中： 20 0.666666666667

Over\_Mean失败用户：

0 ['ZIE0741', '2010-08-27']

1 ['SIS0042', '2010-09-02']

2 ['MDS0680', '2010-09-17']

3 ['WDT1634', '2010-09-20']

4 ['CHP1711', '2010-10-13']

5 ['GWG0497', '2010-10-15']

6 ['KSS1005', '2010-10-16']

7 ['HMS1658', '2010-12-30']

8 ['HIS1394', '2010-12-30']

9 ['MGB1235', '2011-01-21']

10 ['DCC1119', '2011-01-26']

11 ['OKM1092', '2011-04-29']

12 ['HSN0675', '2011-04-29']

13 ['TMT0851', '2011-05-11']

Over\_Median失败用户：

0 ['ZIE0741', '2010-08-27']

1 ['MDS0680', '2010-09-17']

2 ['*WDT1634*', '2010-09-20']

3 ['CHP1711', '2010-10-13']

4 ['HMS1658', '2010-12-30']

5 ['HIS1394', '2010-12-30']

6 ['MGB1235', '2011-01-21']

7 ['OKM1092', '2011-04-29']

8 ['HSN0675', '2011-04-29']

9 ['TMT0851', '2011-05-11']

我们发现，Over\_Median要好于Over\_Mean，这说明数据分布明显不均匀，偏向上部分。

针对一个具体用户分析：**我们发现，Over\_Median中10个用户中有9个用户是因为离职前一个月早到晚回导致的出勤完美，因而无法无法直接体现在直接的缺勤中。**

**但是，一旦JS低了早晚要缺勤的，只不过不在实施窃取的当月，肯定在之前有所体现；**

**因此，需要实验综合的ERD计算实验。**

2010-07

Over\_ELV\_Mean:,XNC1338,0.4230769230769231,**883**

Over\_ELV\_Median:,MKG0606,0.125,**970**

***ZIE0741***,2010-08-27,ELV:,0.0,1394（无迟到早退）

Over\_Mean:,False

Over\_Median:,False

2010-08

Over\_ELV\_Mean:,NJB0011,0.4230769230769231,858

Over\_ELV\_Median:,DAR0139,0.13636363636363635,963

,2010-09-17,ELV:,0.0,1295

Over\_Mean:,False

Over\_Median:,False

2010-09

Over\_ELV\_Mean:,ALT1465,0.4,764

Over\_ELV\_Median:,LDF0644,0.14285714285714285,949

CHP1711,2010-10-13,ELV:,0.0,**1186**

Over\_Mean:,False

Over\_Median:,False

2010-11

Over\_ELV\_Mean:,XNC1338,0.4074074074074074,742

Over\_ELV\_Median:,AYG1697,0.16666666666666666,931

HMS1658,2010-12-30,ELV:,0.0,**1752**

Over\_Mean:,False

Over\_Median:,False

2010-11

Over\_ELV\_Mean:,XNC1338,0.4074074074074074,742

Over\_ELV\_Median:,AYG1697,0.16666666666666666,931

HIS1394,2010-12-30,ELV:,0.0,**1308**

Over\_Mean:,False

Over\_Median:,False

2010-12

Over\_ELV\_Mean:,SKC0074,0.4166666666666667,801

Over\_ELV\_Median:,RCH1216,0.20833333333333334,927

MGB1235,2011-01-21,ELV:,0.0,1186

Over\_Mean:,False

Over\_Median:,False

2011-03

Over\_ELV\_Mean:,NJB0011,0.4230769230769231,797

Over\_ELV\_Median:,LDM0399,0.23076923076923078,895

OKM1092,2011-04-29,ELV:,0.0,1349

Over\_Mean:,False

Over\_Median:,False

2011-03

Over\_ELV\_Mean:,NJB0011,0.4230769230769231,797

Over\_ELV\_Median:,LDM0399,0.23076923076923078,895

HSN0675,2011-04-29,ELV:,0.0,1127

Over\_Mean:,False

Over\_Median:,False

2011-04

Over\_ELV\_Mean:,HST0084,0.42857142857142855,765

Over\_ELV\_Median:,CGS0647,0.23809523809523808,886

TMT0851,2011-05-11,ELV:,0.0,1545

Over\_Mean:,False

Over\_Median:,False

------------------------------------------------------------------------------------------------------------------------------

2010-08

Over\_ELV\_Mean:,NJB0011,0.4230769230769231,858

Over\_ELV\_Median:,DAR0139,0.13636363636363635,963

***WDT1634***,2010-09-20,ELV:,0.09090909090909091,**970**

Over\_Mean:,False

Over\_Median:,False

2018年11月25日星期日

下午12时19分

上次实验中，我们发现基于中位数单月LED检测，我们发现有9个用户在离职前一个月的出勤率完美，因此需要分析综合出勤表现。为了验证综合分析的合理性，及是否该类用户的LJS可以在出勤表现上有所反映，我们先挑选一个用户进行手动分析。

以用户***ZIE0741为例***

2010-01： ZIE0741,0.0,

2010-02： ZIE0741,0.0,

2010-03： ZIE0741,0.0,

2010-04： ZIE0741,0.0,

2010-05： ZIE0741,0.0,

2010-06：ZIE0741,0.0,

2010-07: ZIE0741,0.0,

2010-08: ZIE0741,0.0,

奇怪！ZIE用户竟然从来没有过缺勤！是都这样还是特殊情况？

继续考察用户***MDS0680***

2010-01： MDS0680,0.0,

2010-02：MDS0680,0.0,

2010-03：MDS0680,0.0,

2010-04： MDS0680,0.0,

2010-05：MDS0680,0.0,

2010-06：MDS0680,0.0,

2010-07：MDS0680,0.0,

2010-08：MDS0680,0.0,

2010-09：MDS0680,0.0,

由此看出，依据标准上下班时间9：00-17：00而言，有9个用户完全没有缺勤的表现。

为了与之前的情况有所对照，我们发现：

ZIE用户一直工作上班时间都在8：00左右，每天都比9：00早近一个小时；

而其下班时间在19：00左右。

根据最初基于用户2010-01个人工作开始结束时间的统计分析，可以得到

ZIE0741,WorkOn:8.0,WorkOff:19.0

MDS0680,WorkOn:8.0,WorkOff:17.5

如果以8点<9点为所有用户上班开始的时间，那么必然有某些用户有大量迟到记录，因为他们日常开始的时间晚于8：00但是早于9：00；那么，是否同一个Team的人有着独特的工作开始时间呢？

以ZIE0741为例：

其工作单位团队为'2 - Executive', '2 - ResearchAndEngineering', '3 - SoftwareManagement', '4 - WebSoftware'，则同一个团队下，其他人自有工作开始结束时间为：

ZIE0741,WorkOn:8.0,WorkOff:19.0

BAG0315,WorkOn:8.0,WorkOff:19.0

CKH0316,WorkOn:8.5,WorkOff:19.0

DNJ0740,WorkOn:8.0,WorkOff:19.0

DRN0735,WorkOn:8.0,WorkOff:19.0

GHM0736,WorkOn:8.0,WorkOff:19.0

GIB0314,WorkOn:8.0,WorkOff:19.0

HDW0743,WorkOn:8.0,WorkOff:18.5

ISW0738,WorkOn:8.0,WorkOff:19.0

JLC0742,WorkOn:8.0,WorkOff:19.0

KHT0739,WorkOn:8.0,WorkOff:19.0

LKB0734,WorkOn:8.0,WorkOff:19.0

LNS0317,WorkOn:8.5,WorkOff:18.5

MTH0737,WorkOn:8.0,WorkOff:19.0

**TAH0300**,WorkOn:8.0,WorkOff:19.0 （Team Leader）

WZB0733,WorkOn:8.0,WorkOff:19.0

ZIE0741,WorkOn:8.0,WorkOff:19.0

为了避免偶然性，我们随机选择一个Team继续验证是否该团队内部遵循一致的上下班时间；

['1 - Executive', '1 - Adminstration', '6 - Security', '1 - BuildingSecurity'],ASH0458,CTK0037,FMP0459,GCW0456,GEM0462,JAC0457,JMW0038,NCC0454,ODB0034,OHS0036,OKH0453,ORB0455,QRF0039,TTG0460,VCH0461,XKB0040,

ASH0458,WorkOn:8.0,WorkOff:14.5

CTK0037,WorkOn:8.0,WorkOff:14.5

FMP0459,WorkOn:8.0,WorkOff:14.5

GCW0456,WorkOn:8.0,WorkOff:14.5

GEM0462,WorkOn:8.0,WorkOff:14.5

JAC0457,WorkOn:8.0,WorkOff:14.5

JMW0038,WorkOn:8.0,WorkOff:14.5

NCC0454,WorkOn:8.0,WorkOff:14.5

ODB0034,WorkOn:8.0,WorkOff:14.5

OHS0036,WorkOn:8.0,WorkOff:14.5

OKH0453,WorkOn:8.0,WorkOff:14.5

ORB0455,WorkOn:8.0,WorkOff:14.5

QRF0039,WorkOn:8.0,WorkOff:14.5

TTG0460,WorkOn:8.0,WorkOff:14.5

VCH0461,WorkOn:8.0,WorkOff:14.5

XKB0040,WorkOn:8.0,WorkOff:14.5

* 这里规律非常明显，同一个Team应当由自己独特的上下班时间，直接选取Team中出现次数最多的上下班时间即可

因此，我们可以对用户个人开始工作的时间与标准时间重新开始定义

2018年11月27日星期二

上午9时14分

通过之前对于用户上下班时间的分析，发现既不能完全依靠用户的自身行为时间，也不能单纯依靠标准的朝九晚五时间，而是要使用Team所体现的时间特征。然而，如此对于领导者而言，其时间是否也一致呢？让我们先来探查一番：

比如对于部门领导：

['1 - Executive', '2 - ResearchAndEngineering', '', ''],

GMC0051,VBK0007,

GMC0051,WorkOn:7.5,WorkOff:16.5

VBK0007,WorkOn:7.5,WorkOff:16.5

['1 - Executive', '2 - ResearchAndEngineering', '1 - ProjectManagement', ''],

FYC0060,HBW0057,KTS0058,RAH0052,RRS0056,TTM0059,

FYC0060,WorkOn:8.5,WorkOff:15.0

HBW0057,WorkOn:8.5,WorkOff:15.0

KTS0058,WorkOn:8.5,WorkOff:15.0

RAH0052,WorkOn:8.5,WorkOff:15.0

RRS0056,WorkOn:8.5,WorkOff:15.0

TTM0059,WorkOn:8.5,WorkOff:15.0

['1 - Executive', '2 - ResearchAndEngineering', '2 - Research', '1 - Lab'],

BBF0480,CPD0474,DRJ0070,FKS0069,HSK0479,ITM0068,KJH0475,KMG0062,NRR0476,PTV0067,RKM0478,ROA0482,ROB0477,SKW0066,SMB0473,VDB0481,

VDB0481,WorkOn:8.0,WorkOff:18.0

SMB0473,WorkOn:8.0,WorkOff:18.0

SKW0066,WorkOn:8.0,WorkOff:18.0

ROB0477,WorkOn:8.0,WorkOff:18.0

ROA0482,WorkOn:8.0,WorkOff:18.0

RKM0478,WorkOn:8.0,WorkOff:18.0

PTV0067,WorkOn:8.0,WorkOff:18.0

NRR0476,WorkOn:8.0,WorkOff:17.5

KMG0062,WorkOn:8.0,WorkOff:17.5

KJH0475,WorkOn:8.0,WorkOff:18.0

BBF0480,WorkOn:8.0,WorkOff:18.0

CPD0474,WorkOn:8.0,WorkOff:18.0

DRJ0070,WorkOn:8.0,WorkOff:18.0

FKS0069,WorkOn:8.0,WorkOff:18.0

HSK0479,WorkOn:8.0,WorkOff:18.0

ITM0068,WorkOn:8.0,WorkOff:18.0

综合上述案例分析，我们可以得到结论：

1. 领导管理层依照抽象Team分组后，其上下班时间保持一致；
2. 具体Team分组内部，除去极个别用户出现浮动，绝大多数用户遵循统一的上下班时间；
3. 结论：依据LDAP的四元结构确定的抽线TEAM分组，依据统计优势确定用户上下班时间推定可行；
4. 具体假设分析前提：1）用户上班期间，守时频率要高于不守时，否则早就被解雇了，因此可以通过分析用户在正常月份的上下班时间窗口的统计特性（多少决定）来推定用户的上下班时间；2）虽然有朝九晚五的标准时间，然而不同的企业乃至不同部门之间都会有工作绩效的不同，因此其上下班时间不能采用标准时间笼统概括，而应在抽象TEAM的基础上，同样依据好人多的假设，选取最频繁出现的上下班时间作为该TEAM的上下班时间；在此基础上，再去分析用户的单月LED与符合LED；

我们先写程序统计了Team的上下班时间，进行个简单的验证：

对于['': '': '': ''],ETW0002,

['': '': '': ''],8.5,17.0,

ETW0002,WorkOn:8.5,WorkOff:17.0

对于['1 - Executive': '': '': ''],NJC0003,PTH0005,

NJC0003,WorkOn:8.0,WorkOff:15.5

PTH0005,WorkOn:8.0,WorkOff:16.0

['1 - Executive': '': '': ''],8.0,15.5,

对于2个成员，而下班时间不一致的情况，由于此时投票无法区分，默认选择的往往是第一个，因而我们可以：1）选择时间较小的一个，则后一个不会判定早退；2）如果该Team仅有2个用户，则二者要么一样，要么不一样，此时，每个用户各自使用自己的下班时间，不再统一处理

**['1 - Executive': '': '': ''],8.0,15.5, （同样依据最好人假设，选取最常出现的最晚上班时间与最早下班时间）**

对于：['1 - Executive': '1 - Adminstration': '': ''],BBB0012,CWC0006,

BBB0012,WorkOn:8.0,WorkOff:20.0

CWC0006,WorkOn:8.0,WorkOff:20.0

['1 - Executive': '1 - Adminstration': '': ''],8.0,20.0,

上午11时0分

程序终于编写完毕，整体结果使用ZIE验证：

ZIE0741,WorkOn:8.0,WorkOff:19.0，不同于标准时间

而

KMG0062,WorkOn:8.0,WorkOff:18.0的Team时间不同于

KMG0062,WorkOn:8.0,WorkOff:17.5

基本验证通过！

开始重新统计所有用户的出勤表现！

**V07\_CERT5.2\_Users\_WorkOn-Off\_Time\_Team.csv**

**2018年11月27日星期二**

**下午2时4分**

中午初步得到了2010-07月份的Team数据，我们来一起看看对于2010-08月份的单月预测结果。

2010-07

Over\_Mean\_Team:RBC1624,0.7142857142857142 (1099)

Over\_Median\_Team:LSN1462,0.8620689655172413 (967)

VCF1602,2010-08-20,

VCF1602,1.1481481481481481,(567)

CKP0630,2010-08-26,

CKP0630,0.5238095238095237,(1187)

ZIE0741,2010-08-27,

ZIE0741,1.0952380952380951,(735)

从单月来看，ZIE用户的结果不再为全部全勤，而是出现了迟到与早退：

ZIE0741,8.0,19.0,14.0,9.0,21,

CKP0630,8.0,15.5,7.0,4.0,21,

VCF1602,9.0,19.0,15.0,16.0,27,

RBC1624,9.0,19.0,5.0,10.0,21,

RBC1624,WorkOn:9.0,WorkOff:19.0

我们来看看2010-07月份到底出现缺勤的有多少天（去掉重复的天）

LSN1462,9.0,18.0,12.0,13.0,29,

2010-08

Over\_Mean\_Team: SKW0066,0.6451612903225807（1075）

Over\_Median\_Team:IBR0131,0.7931034482758621（963）

SIS0042,2010-09-02,

SIS0042,0.40909090909090906,（1228）

TNB1616,2010-09-10,

TNB1616,0.9090909090909092, （834）

TRC1838,2010-09-15,

TRC1838,0.0,（1642）

TRC1838,8.5,16.0,0.0,0.0,22,

MDS0680,2010-09-17,

MDS0680,0.0,

MDS0680,8.0,17.5,0.0,0.0,22,

WDT1634,2010-09-20,

WDT1634,1.3181818181818181,（60）

WDT1634,8.5,17.5,15.0,14.0,22,

OSS1463,2010-09-21,

OSS1463,1.0,（728）

CIF1430,2010-09-23,

CIF1430,0.9090909090909091,（847）

CIF1430,8.0,16.0,10.0,10.0,22,

RBC1624,9.0,19.0,5.0,10.0,21,

RBC1624,WorkOn:9.0,WorkOff:19.0

我们来看看2010-07月份到底出现缺勤的有多少天（去掉重复的天）21天中一共有13天出勤不规范，同时迟到早退天数为0

其上下班时间认定为：**RBC1624,WorkOn:9.0,WorkOff:19.0**

07/01/2010 08:58:00,RBC1624,PC-0127,Logon,

07/01/2010 19:04:00,RBC1624,PC-0127,Logoff,

07/02/2010 08:54:00,RBC1624,PC-0127,Logon,

07/02/2010 19:09:00,RBC1624,PC-0127,Logoff,

07/06/2010 09:00:00,RBC1624,PC-0127,Logon,

07/06/2010 18:58:00,RBC1624,PC-0127,Logoff, （1）

07/07/2010 09:00:00,RBC1624,PC-0127,Logon,

07/07/2010 18:58:00,RBC1624,PC-0127,Logoff, （2）

07/08/2010 09:01:00,RBC1624,PC-0127,Logon, （3）

07/08/2010 19:14:00,RBC1624,PC-0127,Logoff,

07/09/2010 09:00:00,RBC1624,PC-0127,Logon,

07/09/2010 18:58:00,RBC1624,PC-0127,Logoff, （4）

07/12/2010 08:58:00,RBC1624,PC-0127,Logon,

07/12/2010 19:00:00,RBC1624,PC-0127,Logoff,

07/13/2010 09:00:00,RBC1624,PC-0127,Logon,

07/13/2010 18:59:00,RBC1624,PC-0127,Logoff, （5）

07/14/2010 09:00:00,RBC1624,PC-0127,Logon,

07/14/2010 19:00:00,RBC1624,PC-0127,Logoff,

07/15/2010 08:53:00,RBC1624,PC-0127,Logon,

07/15/2010 19:06:00,RBC1624,PC-0127,Logoff,

07/16/2010 09:00:00,RBC1624,PC-0127,Logon,

07/16/2010 19:00:00,RBC1624,PC-0127,Logoff,

07/19/2010 09:00:00,RBC1624,PC-0127,Logon,

07/19/2010 19:00:00,RBC1624,PC-0127,Logoff,

07/20/2010 09:00:00,RBC1624,PC-0127,Logon,

07/20/2010 18:58:00,RBC1624,PC-0127,Logoff, （6）

07/21/2010 09:00:00,RBC1624,PC-0127,Logon,

07/21/2010 18:59:00,RBC1624,PC-0127,Logoff, （7）

07/22/2010 08:52:00,RBC1624,PC-0127,Logon,

07/22/2010 19:05:00,RBC1624,PC-0127,Logoff,

07/23/2010 09:00:00,RBC1624,PC-0127,Logon,

07/23/2010 18:59:00,RBC1624,PC-0127,Logoff, （8）

07/26/2010 09:02:00,RBC1624,PC-0127,Logon, （9）

07/26/2010 19:01:00,RBC1624,PC-0127,Logoff,

07/27/2010 09:01:00,RBC1624,PC-0127,Logon, （10）

07/27/2010 18:59:00,RBC1624,PC-0127,Logoff,

07/28/2010 09:01:00,RBC1624,PC-0127,Logon, （11）

07/28/2010 18:59:00,RBC1624,PC-0127,Logoff,

07/29/2010 09:01:00,RBC1624,PC-0127,Logon, （12）

07/29/2010 19:05:00,RBC1624,PC-0127,Logoff,

07/30/2010 09:00:00,RBC1624,PC-0127,Logon,

07/30/2010 18:59:00,RBC1624,PC-0127,Logoff, （13）

接下来我们来看看CKP0630， 21天中9天不规范，3天同时迟到早退

CKP0630,WorkOn:8.0,WorkOff:15.5（15：30分）

Line 567: 07/01/2010 07:31:00,CKP0630,PC-4612,Logon,

Line 3022: 07/01/2010 15:51:00,CKP0630,PC-4612,Logoff,

Line 5828: 07/02/2010 07:37:00,CKP0630,PC-4612,Logon,

Line 7976: 07/02/2010 15:48:00,CKP0630,PC-4612,Logoff,

Line 12125: 07/06/2010 07:50:00,CKP0630,PC-4612,Logon,

Line 14018: 07/06/2010 15:39:00,CKP0630,PC-4612,Logoff,

Line 17183: 07/07/2010 08:00:00,CKP0630,PC-4612,Logon,

Line 19273: 07/07/2010 15:36:00,CKP0630,PC-4612,Logoff,

Line 22590: 07/08/2010 08:02:00,CKP0630,PC-4612,Logon, （1）

Line 24286: 07/08/2010 15:40:00,CKP0630,PC-4612,Logoff,

Line 27267: 07/09/2010 07:57:00,CKP0630,PC-4612,Logon,

Line 29384: 07/09/2010 15:32:00,CKP0630,PC-4612,Logoff,

Line 32891: 07/12/2010 07:49:00,CKP0630,PC-4612,Logon,

Line 35046: 07/12/2010 15:32:00,CKP0630,PC-4612,Logoff,

Line 38320: 07/13/2010 08:08:00,CKP0630,PC-4612,Logon, （2）

Line 39949: 07/13/2010 15:24:00,CKP0630,PC-4612,Logoff,

Line 42805: 07/14/2010 07:54:00,CKP0630,PC-4612,Logon,

Line 44933: 07/14/2010 15:39:00,CKP0630,PC-4612,Logoff,

Line 47418: 07/15/2010 07:31:00,CKP0630,PC-4612,Logon,

Line 49806: 07/15/2010 15:20:00,CKP0630,PC-4612,Logoff, （4）

Line 52877: 07/16/2010 07:54:00,CKP0630,PC-4612,Logon,

Line 54998: 07/16/2010 15:35:00,CKP0630,PC-4612,Logoff,

Line 59017: 07/19/2010 08:09:00,CKP0630,PC-4612,Logon, （5）

Line 60644: 07/19/2010 15:18:00,CKP0630,PC-4612,Logoff,

Line 63997: 07/20/2010 08:09:00,CKP0630,PC-4612,Logon, （6）

Line 65614: 07/20/2010 15:39:00,CKP0630,PC-4612,Logoff,

Line 68428: 07/21/2010 07:41:00,CKP0630,PC-4612,Logon,

Line 70788: 07/21/2010 15:39:00,CKP0630,PC-4612,Logoff,

Line 73615: 07/22/2010 07:43:00,CKP0630,PC-4612,Logon,

Line 75713: 07/22/2010 15:53:00,CKP0630,PC-4612,Logoff,

Line 79369: 07/23/2010 08:11:00,CKP0630,PC-4612,Logon, （7）

Line 81045: 07/23/2010 15:49:00,CKP0630,PC-4612,Logoff,

Line 84238: 07/26/2010 07:41:00,CKP0630,PC-4612,Logon,

Line 86525: 07/26/2010 15:37:00,CKP0630,PC-4612,Logoff,

Line 89762: 07/27/2010 08:06:00,CKP0630,PC-4612,Logon, （8）

Line 91524: 07/27/2010 15:41:00,CKP0630,PC-4612,Logoff,

Line 94603: 07/28/2010 08:04:00,CKP0630,PC-4612,Logon, （9）

Line 96367: 07/28/2010 15:23:00,CKP0630,PC-4612,Logoff,

Line 99101: 07/29/2010 07:36:00,CKP0630,PC-4612,Logon,

Line 101385: 07/29/2010 16:00:00,CKP0630,PC-4612,Logoff,

Line 104323: 07/30/2010 07:55:00,CKP0630,PC-4612,Logon,

Line 106421: 07/30/2010 15:31:00,CKP0630,PC-4612,Logoff,

再来看看ZIE0741

ZIE0741,WorkOn:8.0,WorkOff:19.0， 16天不规范，9天同一天迟到早退

Line 1117: 07/01/2010 07:54:00,ZIE0741,PC-9132,Logon,

Line 4547: 07/01/2010 19:08:00,ZIE0741,PC-9132,Logoff,

Line 6116: 07/02/2010 07:48:00,ZIE0741,PC-9132,Logon,

Line 9662: 07/02/2010 19:10:00,ZIE0741,PC-9132,Logoff,

Line 11997: 07/06/2010 07:45:00,ZIE0741,PC-9132,Logon,

Line 15657: 07/06/2010 19:04:00,ZIE0741,PC-9132,Logoff,

Line 17460: 07/07/2010 08:03:00,ZIE0741,PC-9132,Logon, （1）

Line 20797: 07/07/2010 18:57:00,ZIE0741,PC-9132,Logoff,

Line 22164: 07/08/2010 07:46:00,ZIE0741,PC-9132,Logon,

Line 25898: 07/08/2010 19:00:00,ZIE0741,PC-9132,Logoff,

Line 27531: 07/09/2010 08:02:00,ZIE0741,PC-9132,Logon, （2）

Line 30970: 07/09/2010 18:58:00,ZIE0741,PC-9132,Logoff,

Line 32886: 07/12/2010 07:48:00,ZIE0741,PC-9132,Logon,

Line 36657: 07/12/2010 19:10:00,ZIE0741,PC-9132,Logoff,

Line 38102: 07/13/2010 08:01:00,ZIE0741,PC-9132,Logon, （3）

Line 41488: 07/13/2010 18:56:00,ZIE0741,PC-9132,Logoff,

Line 43180: 07/14/2010 08:04:00,ZIE0741,PC-9132,Logon, （5）

Line 46455: 07/14/2010 18:58:00,ZIE0741,PC-9132,Logoff,

Line 48227: 07/15/2010 08:04:00,ZIE0741,PC-9132,Logon, （6）

Line 51502: 07/15/2010 19:10:00,ZIE0741,PC-9132,Logoff,

Line 53143: 07/16/2010 08:02:00,ZIE0741,PC-9132,Logon, （7）

Line 56607: 07/16/2010 19:00:00,ZIE0741,PC-9132,Logoff,

Line 58749: 07/19/2010 08:01:00,ZIE0741,PC-9132,Logon, （8）

Line 62219: 07/19/2010 18:59:00,ZIE0741,PC-9132,Logoff,

Line 63882: 07/20/2010 08:03:00,ZIE0741,PC-9132,Logon, （9）

Line 67262: 07/20/2010 19:07:00,ZIE0741,PC-9132,Logoff,

Line 68964: 07/21/2010 08:02:00,ZIE0741,PC-9132,Logon, （10）

Line 72295: 07/21/2010 18:57:00,ZIE0741,PC-9132,Logoff,

Line 73793: 07/22/2010 07:48:00,ZIE0741,PC-9132,Logon,

Line 77339: 07/22/2010 19:12:00,ZIE0741,PC-9132,Logoff,

Line 79242: 07/23/2010 08:04:00,ZIE0741,PC-9132,Logon, （11）

Line 82522: 07/23/2010 19:05:00,ZIE0741,PC-9132,Logoff,

Line 84893: 07/26/2010 08:04:00,ZIE0741,PC-9132,Logon, （12）

Line 88034: 07/26/2010 18:58:00,ZIE0741,PC-9132,Logoff,

Line 89616: 07/27/2010 08:02:00,ZIE0741,PC-9132,Logon, （13）

Line 93072: 07/27/2010 19:00:00,ZIE0741,PC-9132,Logoff,

Line 94397: 07/28/2010 08:00:00,ZIE0741,PC-9132,Logon,

Line 97924: 07/28/2010 18:58:00,ZIE0741,PC-9132,Logoff, （14）

Line 99832: 07/29/2010 08:03:00,ZIE0741,PC-9132,Logon, （15）

Line 102887: 07/29/2010 19:09:00,ZIE0741,PC-9132,Logoff,

Line 104578: 07/30/2010 08:02:00,ZIE0741,PC-9132,Logon, （16）

Line 107966: 07/30/2010 18:56:00,ZIE0741,PC-9132,Logoff,

我们看看2010-09离职的

TRC1838,8.5,16.0,0.0,0.0,22,

Line 1673: 08/02/2010 08:06:00,TRC1838,PC-7903,Logon,

Line 3407: 08/02/2010 16:11:00,TRC1838,PC-7903,Logoff,

Line 6888: 08/03/2010 08:19:00,TRC1838,PC-7903,Logon,

Line 8574: 08/03/2010 16:09:00,TRC1838,PC-7903,Logoff,

Line 11811: 08/04/2010 08:05:00,TRC1838,PC-7903,Logon,

Line 13553: 08/04/2010 16:27:00,TRC1838,PC-7903,Logoff,

Line 17013: 08/05/2010 08:18:00,TRC1838,PC-7903,Logon,

Line 18831: 08/05/2010 16:17:00,TRC1838,PC-7903,Logoff,

Line 21851: 08/06/2010 08:16:00,TRC1838,PC-7903,Logon,

Line 23731: 08/06/2010 16:12:00,TRC1838,PC-7903,Logoff,

Line 27592: 08/09/2010 08:23:00,TRC1838,PC-7903,Logon,

Line 29195: 08/09/2010 16:14:00,TRC1838,PC-7903,Logoff,

Line 32533: 08/10/2010 08:10:00,TRC1838,PC-7903,Logon,

Line 34473: 08/10/2010 16:30:00,TRC1838,PC-7903,Logoff,

Line 37684: 08/11/2010 08:22:00,TRC1838,PC-7903,Logon,

Line 39369: 08/11/2010 16:22:00,TRC1838,PC-7903,Logoff,

Line 42663: 08/12/2010 08:12:00,TRC1838,PC-7903,Logon,

Line 44258: 08/12/2010 16:24:00,TRC1838,PC-7903,Logoff,

Line 47630: 08/13/2010 08:18:00,TRC1838,PC-7903,Logon,

Line 49384: 08/13/2010 16:11:00,TRC1838,PC-7903,Logoff,

Line 53106: 08/16/2010 08:05:00,TRC1838,PC-7903,Logon,

Line 54910: 08/16/2010 16:11:00,TRC1838,PC-7903,Logoff,

Line 58282: 08/17/2010 08:17:00,TRC1838,PC-7903,Logon,

Line 60048: 08/17/2010 16:08:00,TRC1838,PC-7903,Logoff,

Line 63468: 08/18/2010 08:15:00,TRC1838,PC-7903,Logon,

Line 65020: 08/18/2010 16:28:00,TRC1838,PC-7903,Logoff,

Line 68476: 08/19/2010 08:06:00,TRC1838,PC-7903,Logon,

Line 70195: 08/19/2010 16:19:00,TRC1838,PC-7903,Logoff,

Line 73531: 08/20/2010 08:22:00,TRC1838,PC-7903,Logon,

Line 75199: 08/20/2010 16:13:00,TRC1838,PC-7903,Logoff,

Line 79200: 08/23/2010 08:14:00,TRC1838,PC-7903,Logon,

Line 80799: 08/23/2010 16:28:00,TRC1838,PC-7903,Logoff,

Line 84197: 08/24/2010 08:23:00,TRC1838,PC-7903,Logon,

Line 85891: 08/24/2010 16:09:00,TRC1838,PC-7903,Logoff,

Line 89114: 08/25/2010 08:19:00,TRC1838,PC-7903,Logon,

Line 90864: 08/25/2010 16:10:00,TRC1838,PC-7903,Logoff,

Line 94029: 08/26/2010 08:20:00,TRC1838,PC-7903,Logon,

Line 95793: 08/26/2010 16:10:00,TRC1838,PC-7903,Logoff,

Line 98897: 08/27/2010 08:16:00,TRC1838,PC-7903,Logon,

Line 100659: 08/27/2010 16:07:00,TRC1838,PC-7903,Logoff,

Line 104508: 08/30/2010 08:22:00,TRC1838,PC-7903,Logon,

Line 106247: 08/30/2010 16:13:00,TRC1838,PC-7903,Logoff,

Line 109326: 08/31/2010 08:18:00,TRC1838,PC-7903,Logon,

Line 111134: 08/31/2010 16:12:00,TRC1838,PC-7903,Logoff,

检查下TRC的Team情况

重点分析TRC1838是否真的一直全勤？

Oneself：TRC1838,WorkOn:8.5,WorkOff:16.0

Team: TRC1838,WorkOn:8.5,WorkOff:16.0

['1 - Executive': '5 - SalesAndMarketing': '2 - Sales': '5 - RegionalSales'],

ACH1831,WorkOn:8.5,WorkOff:16.0

ACH1840,WorkOn:8.5,WorkOff:16.0

BGF1845,WorkOn:8.5,WorkOff:16.0

BYO1846,WorkOn:8.5,WorkOff:16.0

CBW1826,WorkOn:8.5,WorkOff:16.0

CCB1836,WorkOn:8.5,WorkOff:16.0

DEB1767,WorkOn:8.5,WorkOff:16.0

ESM1828,WorkOn:8.5,WorkOff:16.0

FDS1841,WorkOn:8.5,WorkOff:16.0

GFV1837,WorkOn:8.5,WorkOff:16.0

GRB1842,WorkOn:8.5,WorkOff:16.0

HDA1824,WorkOn:8.5,WorkOff:16.0

HMC1847,WorkOn:8.5,WorkOff:16.0

JBP1832,WorkOn:8.5,WorkOff:16.0

JKB1843,WorkOn:8.5,WorkOff:16.0

MPK1844,WorkOn:8.5,WorkOff:16.0

MUP1819,WorkOn:8.5,WorkOff:16.0

NJV1818,WorkOn:8.5,WorkOff:16.0

NRR1835,WorkOn:8.5,WorkOff:16.0

ACH1831,ACH1840,BGF1845,BYO1846,CBW1826,CCB1836,DEB1767,ESM1828,FDS1841,GFV1837,GRB1842,HDA1824,HMC1847,JBP1832,JKB1843,MPK1844,MUP1819,NJV1818,NRR1835,NZH1839,ONB1833,RMB1821,RMV1820,RSS1825,TRC1838,WAS1823,WCR1830,WLH1827,WOS1834,XKB1829,YDM1822,

由上述数据验证，可知TRC所在Team的上下班时间真的为8.5-16.0

我们开始依次检查TRC各个月份的出勤情况

2010-01： 2010-09

2010-01：TRC1838,0.0,

2010-02：TRC1838,8.5,16.0,0.0,0.0,20,

2010-03：TRC1838,0.0,

2010-04：TRC1838,0.0,

2010-05：TRC1838,0.0,

说明，TRC这家伙真的按时上下班啊~

CHP1711,WorkOn:8.0,WorkOff:17.5

2018年11月29日星期四

上午9时10分

终于跑完了2010-01：2011：04的基于Team上下班时间的用户出勤分析，并且将该月同一天迟到早退的天数也顺带进行了统计，得到了新的特征文件与新的特征形式：

* *Month*\_early\_late\_team\_feats.csv (斜体表示可以替换的文件名称)
* BGR0917,6.5,18.0,15.0,1.0,29,-1 （-1字段表示该月该用户无同时迟到早退天数）
* MYB0686,6.5,18.5,7.0,7.0,21,2,2010-07-19,2010-07-27,（若该月该用户存在同时迟到早退，则传统出勤特征后追加了同时迟到早退的天数以及日期特征）

上次自己感觉上同一天迟到与早退可以有效避免偶然性的迟到早退，而突出主观倾向更强的缺勤表现。为此，我们先来统计CERT5.2中全部用户的同时迟到与早退的天数（2010-01：2011：04）

对于Insiders\_2而言，30个用户中还是那9个没有缺勤的用户没有记录，其余均有记录

1 ['BYO1846', [0.0, 222.0]]

2 ['CHP1711', [0.0, 199.0]]

3 ['CIF1430', [44.0, 165.0]]

4 ['CKP0630', [8.0, 166.0]]

5 ['DCC1119', [4.0, 269.0]]

6 ['GWG0497', [60.0, 181.0]]

7 ['HIS1394', [52.0, 231.0]]

8 ['HMS1658', [75.0, 231.0]]

9 ['HSN0675', [1.0, 335.0]]

10 ['HXP0976', [70.0, 245.0]]

11 ['ICB1354', [0.0, 242.0]]

12 ['ITA0159', [0.0, 285.0]]

13 ['JAL0811', [1.0, 291.0]]

14 ['KSS1005', [2.0, 253.0]]

15 ['LVF1626', [81.0, 241.0]]

16 ['MCP0611', [0.0, 196.0]]

17 ['MDS0680', [0.0, 181.0]]

18 ['MGB1235', [1.0, 266.0]]

19 ['NAH1366', [0.0, 224.0]]

20 ['OKM1092', [104.0, 335.0]]

21 ['OSS1463', [65.0, 182.0]]

22 ['RRS0056', [0.0, 219.0]]

23 ['SIS0042', [2.0, 171.0]]

24 ['SNK1280', [97.0, 261.0]]

25 ['TMT0851', [75.0, 325.0]]

26 ['TNB1616', [42.0, 159.0]]

27 ['TRC1838', [0.0, 159.0]]

28 ['VCF1602', [58.0, 211.0]]

29 ['WDT1634', [64.0, 162.0]]

30 ['ZIE0741', [35.0, 147.0]]

Cnt\_Null is : 9

若直接使用全部月份总的同时迟到与早退的比例进行分析，Insiders\_2的排序并不靠前：

同样是9个用户分析不到，不如直接用中位数等还可以筛选出一半多的用户；

36 ['WDT1634', '64.0', '162.0', '0.395061728395']

125 ['SNK1280', '97.0', '261.0', '0.371647509579']

224 ['OSS1463', '65.0', '182.0', '0.357142857143']

412 ['LVF1626', '81.0', '241.0', '0.336099585062']

450 ['GWG0497', '60.0', '181.0', '0.331491712707']

504 ['HMS1658', '75.0', '231.0', '0.324675324675']

597 ['OKM1092', '104.0', '335.0', '0.310447761194']

719 ['HXP0976', '70.0', '245.0', '0.285714285714']

775 ['VCF1602', '58.0', '211.0', '0.274881516588']

816 ['CIF1430', '44.0', '165.0', '0.266666666667']

825 ['TNB1616', '42.0', '159.0', '0.264150943396']

943 ['ZIE0741', '35.0', '147.0', '0.238095238095']

967 ['TMT0851', '75.0', '325.0', '0.230769230769']

980 ['HIS1394', '52.0', '231.0', '0.225108225108']

1229 ['CKP0630', '8.0', '166.0', '0.0481927710843']

1394 ['DCC1119', '4.0', '269.0', '0.0148698884758']

1428 ['SIS0042', '2.0', '171.0', '0.0116959064327']

1465 ['KSS1005', '2.0', '253.0', '0.00790513833992']

1487 ['MGB1235', '1.0', '266.0', '0.00375939849624']

1488 ['JAL0811', '1.0', '291.0', '0.00343642611684']

1490 ['HSN0675', '1.0', '335.0', '0.00298507462687']

针对后续的使用基于Team的上下班时间来重新计算Insiders\_2，发现单月预测结果同之前一样，依旧是10个用户无法区分（记住中位数划分）