



پیش بینی و ارزیابی تماس های ورودی شرکت گل‌دیران

دانشجو : حمید جلالی

در مراکزی که به صورت (مرکز تماس call center) هستند همواره نیاز است که کاربران با هدف بهبود پاسخگویی مورد ارزیابی و مورد مطالعه قرار گیرند.

ما میخواهیم مدلی بسازیم تا عملکرد کارمندان جدیدی که وارد شرکت می شوند رو بررسی و پیش بینی کنیم که کاربر جدید نسبت به بقیه کاربران چه عملکردی را داشته اشت

تماس های ورودی مشتریان از طریق Voip به هر کاربر تعریف شده در قسمت پاسخگویی متصل می شود



هر کاربر از طریق برنامه xlite تماس های ورودی به خود را دریافت می کند و هر کاربر داخلی اختصاصی به خود را دارد.

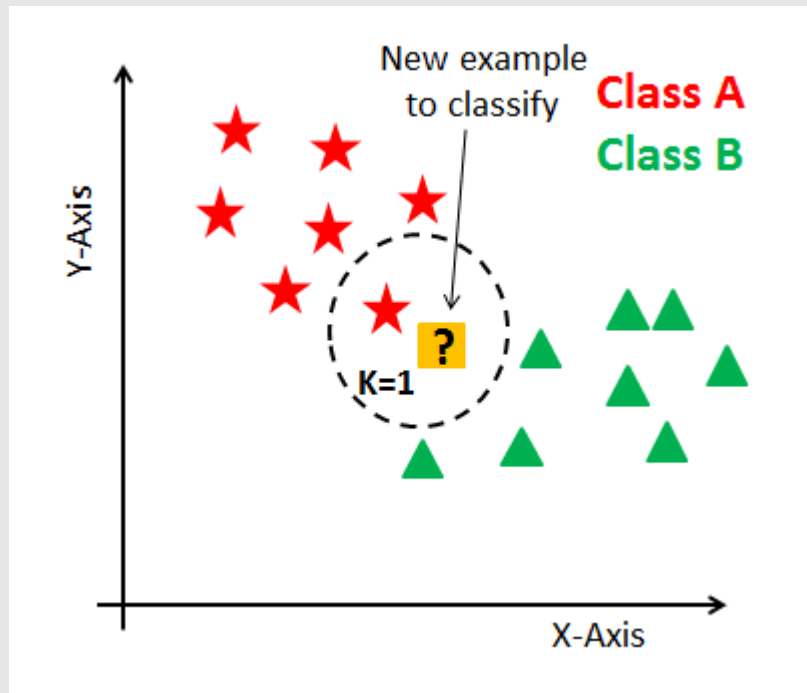
برای هر تماسی از سمت مشتری، اطلاعاتی در جدول تماس ثبت می‌گردد. اطلاعات کامل هر تماس از قبیل تعداد تماس ورودی، نمرات دریافت شده، تعداد تماس از دست رفته و داخلی هر کاربر و میانگین مکالمات در یک جدول ذخیره می‌شود.

که در این دیتاست مهمترین ها را برای پیش بینی و تحلیل جدا کرده ایم.

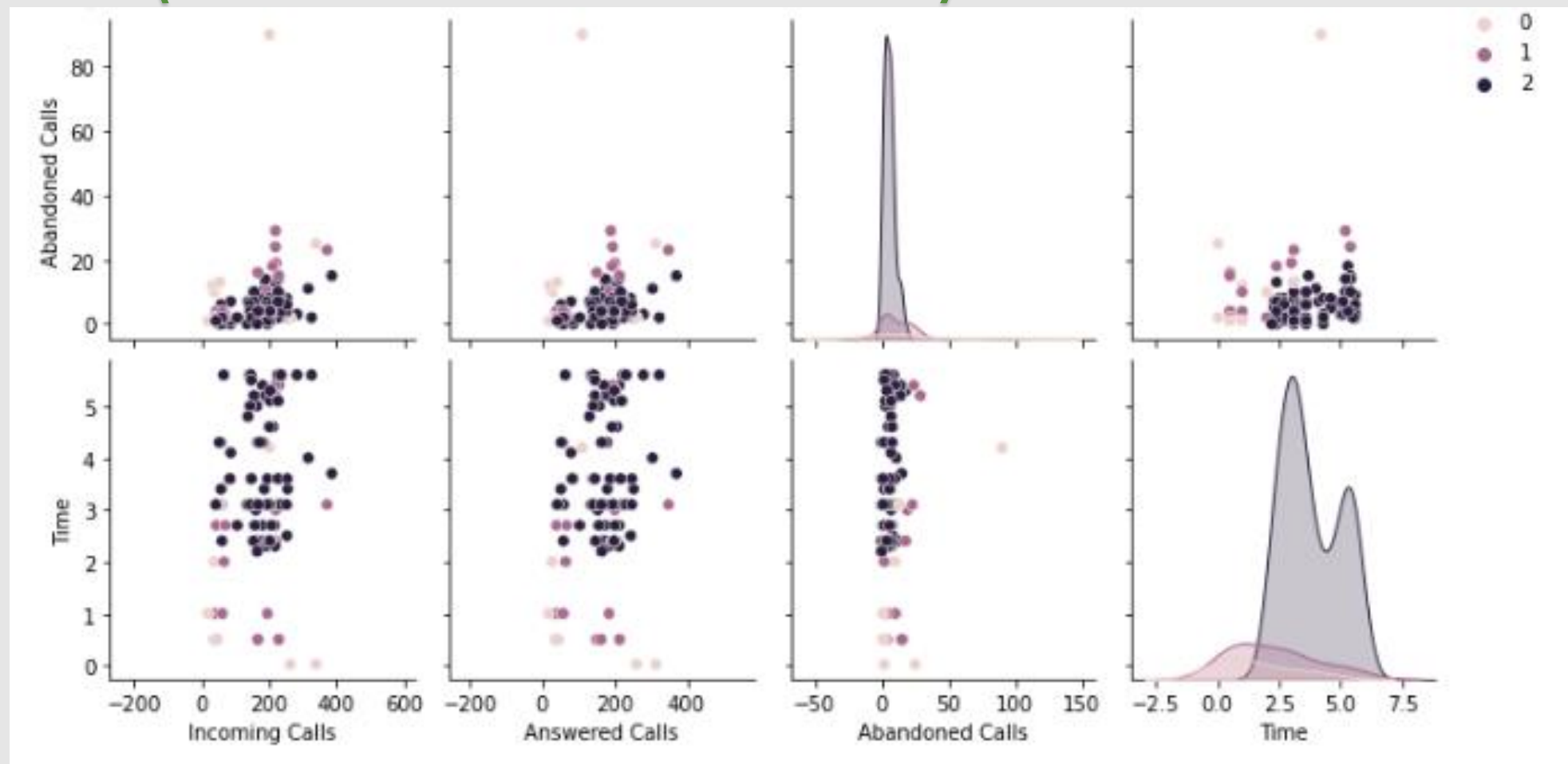
اطلاعاتی که در این دیتاست  
ذخیره شده است از قبیل،  
ایندکس کاربر میانگین زمان  
مکالمات ، تعداد تماس های  
ورودی ، تماس های پاسخ  
داده، تماس های از دست  
رفته، ارزیابی کارمند.

	Incoming Calls	Answered Calls	Abandoned Calls	Time	point
0	217	204	13	2.4	good
1	200	110	90	4.2	weak
2	216	198	18	5.3	good
3	155	145	10	5.2	good
4	37	27	10	2.0	weak
...	...	...	...	...	...
98	202	198	4	5.3	good
99	226	219	7	5.1	good
100	167	163	4	3.1	good
101	164	164	0	2.2	good
102	42	41	1	3.1	good

ما می‌خواهیم با استفاده از Data mining و Machine learning یک سری پیش بینی ها را روی این دیتاست پیاده سازی کنیم که به عنوان نمونه الگوریتم KNN را مورد استفاده قرار دادیم.



## نمودار F2F (feature to feature)



نمودار کامل درون کد پروژه

# ما با استفاده از دستور replace داده های غیر عددی را به داده های عددی تقسیم کردیم

In [90]:

```
cleanup= {"point": {"weak": 0, "normal": 1, "good": 2 }}  
df =df.replace(cleanup)  
df
```

Out[90]:

	Incoming Calls	Answered Calls	Abandoned Calls	Time	point	Unnamed: 5
0	217	204	13	2.4	2	NaN
1	200	110	90	4.2	0	NaN
2	216	198	18	5.3	2	NaN
3	155	145	10	5.2	2	NaN
4	37	27	10	2.0	0	NaN
...	...	...	...	...	...	...
98	202	198	4	5.3	2	NaN
99	226	219	7	5.1	2	NaN
100	167	163	4	3.1	2	NaN
101	164	164	0	2.2	2	NaN
102	42	41	1	3.1	2	NaN

ارزیابی کاربران به ۳ دسته  
خوب (عدد ۲) متوسط (۱)  
و ضعیف (۰) رتبه بندی  
می شود.



ما برای مدل کردن و یادگیری ماشین به یک الگوریتم نیاز داریم و یک متد یادگیری

ابتدا ما تمام داده ها را به دو دسته train و Test تقسیم می کنیم . با دسته Train ماشین را آموزش میدهیم و با دسته test داده ها را آزمایش میکنیم

## دلایل استفاده از متد Train و Test

ما همه ی داده ها با تمام مشخصات را به ماشین نمیدهیم تا بتوانیم با یک قسمت ناشناخته از داده ها از ماشین آزمون بگیریم

```
Knn = KNeighborsClassifier(n_neighbors=4,metric="minkowski",p=2)
```

KNN را فراخوانی کرده و تعداد همسایگی و متد بررسی آن را مشخص می کنیم .  
حال الگوریتم آماده ی دریافت و پردازش ورودی های فراهم شده است

## نتیجه پیش بینی

```
y_predict = Knn.predict(np.array([[150, 70, 80, 2]]))
```

```
y_predict
```

```
array([2], dtype=int64)
```