数据科学基础Ⅰ（Matlab）

（理工类）结课作业

姓　　名	：	孔庆芃
学　　号	：	20201043
班　　级	：	软件学院 数字 2001

一、 实验环境

1. 操作系统：macOS Big Sur Version 11.6
2. 调试软件：MATLAB R2019b

二、 问题概述

一家提供数据科学培训的公司正需要为自己的公司寻找数据科学家。然而，他们在招聘过程中面临困难。在处理了一些候选人后，候选人往往在招聘过程结束时跳出。这种情况使人力资源部门不得不重新开始寻找合适的候选人。

到了年底，人力资源部门决定在招聘过程中对报名参加培训的候选人进行新的改进。这样做的好处是，公司将很容易知道这些候选人中哪些是真正想在培训后为公司工作或寻找新工作的，因为它有助于减少成本和时间，以及培训或计划课程和候选人分类的质量。与人口统计学、教育、经验有关的信息都在候选人的注册和登记中。

因此我的实验思路就是对以往候选人的各项指标及他们最后是否选择更换工作（即来到当前的公司）之间的关系进行**训练**，用训练的模型对一批新候选人各项指标进行分析，**预测**他们是否会选择来到本公司的工作（即更换工作）。

三、 期待解决的问题

1. 机器学习方面：如何预测候选人寻找新工作或为公司工作的概率，以及解释影响员工决定的因素？
2. 探索性数据分析方面：每个数据特征对目标列的影响如何？

四、 数据集指标介绍

- enrollee_id : Unique ID for candidate 候选人的唯一 ID
- city: City code 城市代码
- city_ development _index : Developement index of the city (scaled) 城市的发展指数（按比例）
- gender: Gender of candidate 候选人的性别
- relevent_experience: Relevant experience of candidate 候选人的相关经验
- enrolled_university: Type of University course enrolled if any 大学课程的类型（如果有的话）
- education_level: Education level of candidate 候选人的教育水平
- major_discipline :Education major discipline of candidate 候选人的教育专业学科
- experience: Candidate total experience in years 候选人的总工作经验（年）
- company_size: No of employees in current employer's company 目前雇主公司的员工数量
- company_type : Type of current employer 当前雇主的类型
- last new job: Difference in years between previous job and current job 上一份工作与当前工作的年限差值
- training_hours: training hours completed 完成的培训时间

- target: 0 – Not looking for job change 不寻求工作变化, 1 – Looking for a job change 寻求工作变化

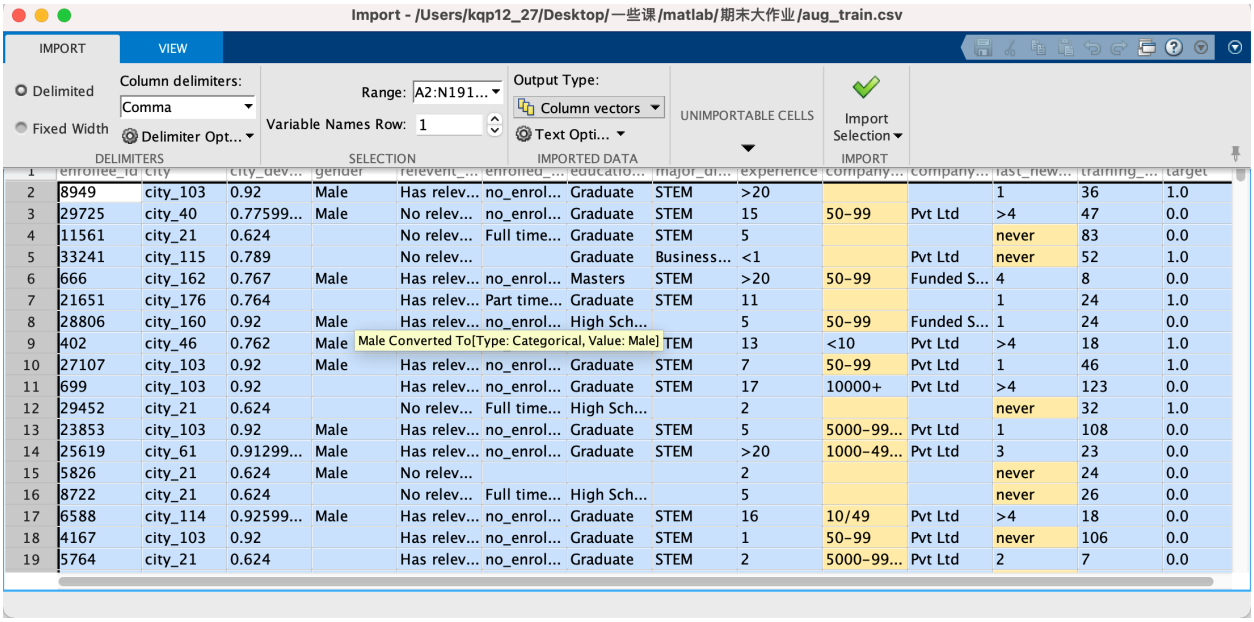
aug_train.csv 数据：19158 rows * 14 columns

aug_test.csv 数据：2129 rows * 13 columns

五、 数据描述

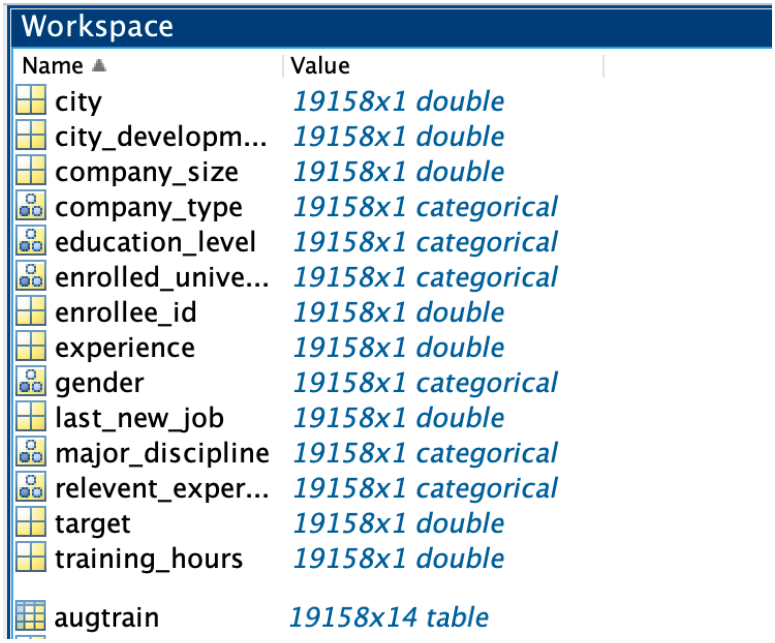
1. 准备工作——数据导入

将 aug_train.csv 数据分别按照“导入 table”和“导入列向量”两种模式导入 Matlab。



	enrollee_id	city	city_dev...	gender	relevent_...	enrolled_...	educatio...	major_di...	experience	company...	company...	last_new...	training_...	target
2	8949	city_103	0.92	Male	Has relev...	no_enrol...	Graduate	STEM	>20			1	36	1.0
3	29725	city_40	0.77599...	Male	No relev...	no_enrol...	Graduate	STEM	15	50-99	Pvt Ltd	>4	47	0.0
4	11561	city_21	0.624		No relev...	Full time...	Graduate	STEM	5			never	83	0.0
5	33241	city_115	0.789		No relev...		Graduate	Business...	<1		Pvt Ltd	never	52	1.0
6	666	city_162	0.767	Male	Has relev...	no_enrol...	Masters	STEM	>20	50-99	Funded S...	4	8	0.0
7	21651	city_176	0.764		Has relev...	Part time...	Graduate	STEM	11			1	24	1.0
8	28806	city_160	0.92	Male	Has relev...	no_enrol...	High Sch...		5	50-99	Funded S...	1	24	0.0
9	402	city_46	0.762	Male	Has relev...	no_enrol...	Graduate	STEM	13	<10	Pvt Ltd	>4	18	1.0
10	27107	city_103	0.92	Male	Has relev...	no_enrol...	Graduate	STEM	7	50-99	Pvt Ltd	1	46	1.0
11	699	city_103	0.92		Has relev...	no_enrol...	Graduate	STEM	17	10000+	Pvt Ltd	>4	123	0.0
12	29452	city_21	0.624		No relev...	Full time...	High Sch...		2			never	32	1.0
13	23853	city_103	0.92	Male	Has relev...	no_enrol...	Graduate	STEM	5	5000-99...	Pvt Ltd	1	108	0.0
14	25619	city_61	0.91299...	Male	Has relev...	no_enrol...	Graduate	STEM	>20	1000-49...	Pvt Ltd	3	23	0.0
15	5826	city_21	0.624	Male	No relev...				2			never	24	0.0
16	8722	city_21	0.624		No relev...	Full time...	High Sch...		5			never	26	0.0
17	6588	city_114	0.92599...	Male	Has relev...	no_enrol...	Graduate	STEM	16	10/49	Pvt Ltd	>4	18	0.0
18	4167	city_103	0.92		Has relev...	no_enrol...	Graduate	STEM	1	50-99	Pvt Ltd	never	106	0.0
19	5764	city_21	0.624		Has relev...	no_enrol...	Graduate	STEM	2	5000-99...	Pvt Ltd	2	7	0.0

导入结果：



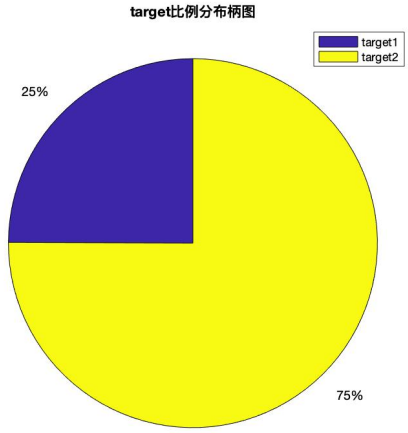
2. 检查 target 数据比例

代码：

```
tar1_per = sum(target(target==1))/length(target);
tar0_per = 1-tar1_per;

fprintf("target=1 情况的占比为:")
fprintf(num2str(tar1_per*100))
disp("%")

fprintf("target=0 情况的占比为:")
fprintf(num2str(tar0_per*100))
disp("%")

pie([tar1_per,tar0_per])
legend("target1","target2")
```

输出：

```
target=1 情况的占比为:24.9348%
target=0 情况的占比为:75.0652%
```



3. 检查数据信息和统计数据

此处由于用 Matlab 不太方便实现，因此使用一点 Python 进行统计，后续会继续使用 Matlab。

Python 代码：

```
# For Dataset handling
import numpy as np
import pandas as pd

# For Checking Distribution Data
from scipy.stats import chisquare, kstest, normaltest
```

```
df = pd.read_csv('aug_train.csv')
df.info()
```

输出：

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 19158 entries, 0 to 19157
Data columns (total 14 columns):
 #   Column                  Non-Null Count  Dtype  
---  ------                  --------------  -----  
 0   enrollee_id             19158 non-null  int64  
 1   city                    19158 non-null  object 
 2   city_development_index  19158 non-null  float64
 3   gender                  14650 non-null  object 
 4   relevent_experience     19158 non-null  object 
 5   enrolled_university     18772 non-null  object 
 6   education_level         18698 non-null  object 
 7   major_discipline        16345 non-null  object 
 8   experience              19093 non-null  object 
 9   company_size            13220 non-null  object 
 10  company_type            13018 non-null  object 
 11  last_new_job            18735 non-null  object 
 12  training_hours          19158 non-null  int64  
 13  target                  19158 non-null  float64
dtypes: float64(2), int64(2), object(10)
memory usage: 2.0+ MB
```

代码：

```
df.describe().T
```

输出：

	count	mean	std	min	25%	50%	75%	max
enrollee_id	19158	16875.36	9616.293	1	8554.25	16982.5	25169.75	33380
city_development_index	19158	0.828848	0.123362	0.448	0.74	0.903	0.92	0.949
training_hours	19158	65.3669	60.05846	1	23	47	88	336
target	19158	0.249348	0.432647	0	0	0	0	1

代码：

```
df.describe(include = 'object').T
```

输出：

	count	unique	top	freq
city	19158	123	city_103	4355
gender	14650	3	Male	13221

relevent_experience	19158	2	Has relevent experience	13792
enrolled_university	18772	3	no_enrollment	13817
education_level	18698	5	Graduate	11598
major_discipline	16345	6	STEM	14492
experience	19093	22	>20	3286
company_size	13220	8	50-99	3083
company_type	13018	6	Pvt Ltd	9817
last_new_job	18735	6	1	8040

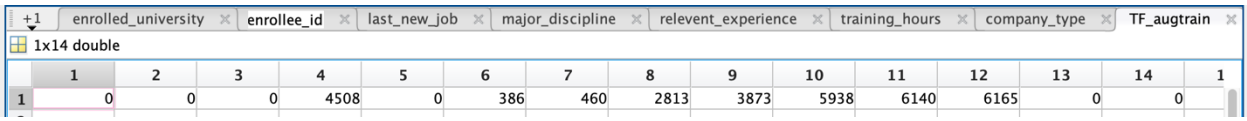
六、 数据清洗

1. 替换缺省值

代码：

```
TF_augtrain = sum(ismissing(augtrain))
```

输出：



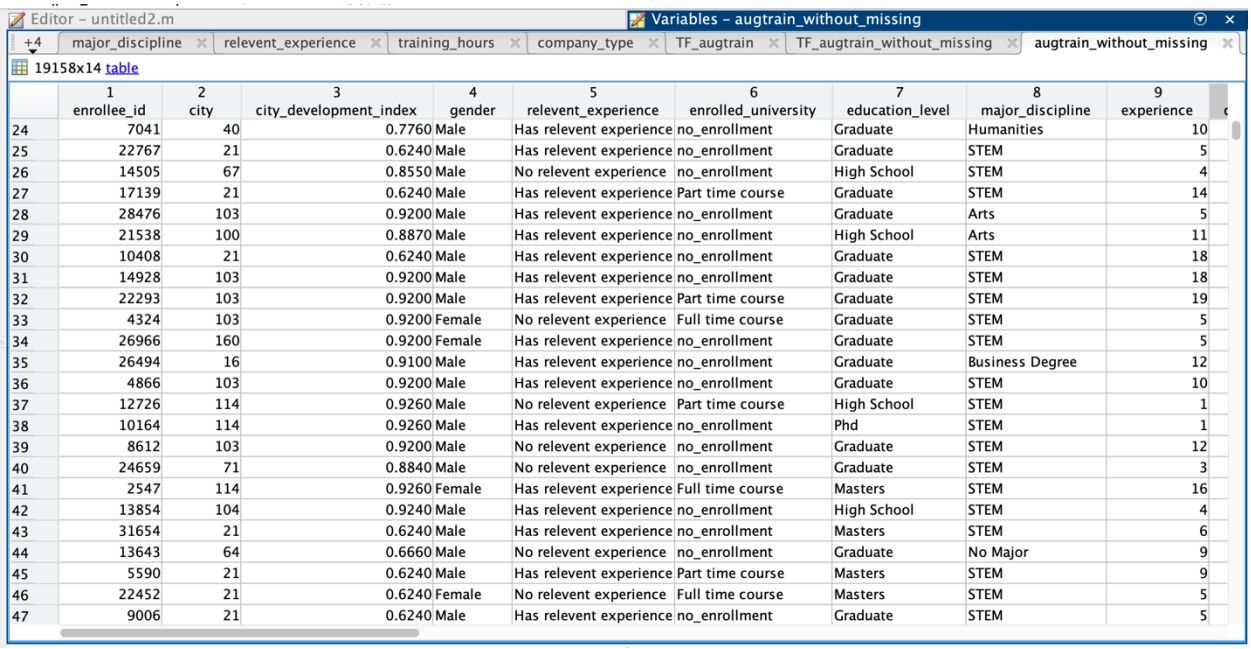
2. 替换缺省值

a) 用上一个条目的值替换所有缺失的条目

代码：

```
augtrain_without_missing = fillmissing(augtrain,'previous');
```

输出：



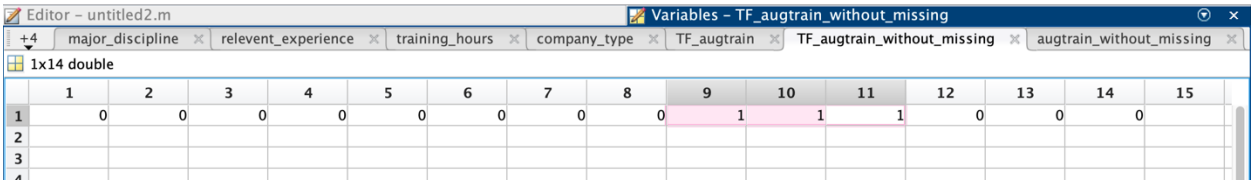
	1 enrollee_id	2 city	3 city_development_index	4 gender	5 relevent_experience	6 enrolled_university	7 education_level	8 major_discipline	9 experience
24	7041	40	0.7760	Male	Has relevent experience	no_enrollment	Graduate	Humanities	10
25	22767	21	0.6240	Male	Has relevent experience	no_enrollment	Graduate	STEM	5
26	14505	67	0.8550	Male	No relevent experience	no_enrollment	High School	STEM	4
27	17139	21	0.6240	Male	Has relevent experience	Part time course	Graduate	STEM	14
28	28476	103	0.9200	Male	Has relevent experience	no_enrollment	Graduate	Arts	5
29	21538	100	0.8870	Male	Has relevent experience	no_enrollment	High School	Arts	11
30	10408	21	0.6240	Male	Has relevent experience	no_enrollment	Graduate	STEM	18
31	14928	103	0.9200	Male	Has relevent experience	no_enrollment	Graduate	STEM	18
32	22293	103	0.9200	Male	Has relevent experience	Part time course	Graduate	STEM	19
33	4324	103	0.9200	Female	No relevent experience	Full time course	Graduate	STEM	5
34	26966	160	0.9200	Female	Has relevent experience	no_enrollment	Graduate	STEM	5
35	26494	16	0.9100	Male	Has relevent experience	no_enrollment	Graduate	Business Degree	12
36	4866	103	0.9200	Male	Has relevent experience	no_enrollment	Graduate	STEM	10
37	12726	114	0.9260	Male	No relevent experience	Part time course	High School	STEM	1
38	10164	114	0.9260	Male	Has relevent experience	no_enrollment	Phd	STEM	1
39	8612	103	0.9200	Male	No relevent experience	no_enrollment	Graduate	STEM	12
40	24659	71	0.8840	Male	No relevent experience	no_enrollment	Graduate	STEM	3
41	2547	114	0.9260	Female	Has relevent experience	Full time course	Masters	STEM	16
42	13854	104	0.9240	Male	Has relevent experience	no_enrollment	High School	STEM	4
43	31654	21	0.6240	Male	Has relevent experience	no_enrollment	Masters	STEM	6
44	13643	64	0.6660	Male	No relevent experience	no_enrollment	Graduate	No Major	9
45	5590	21	0.6240	Male	Has relevent experience	Part time course	Masters	STEM	9
46	22452	21	0.6240	Female	No relevent experience	Full time course	Masters	STEM	5
47	9006	21	0.6240	Male	Has relevent experience	no_enrollment	Graduate	STEM	5

b) 检验是否将缺省值都去除

代码：

```
TF_augtrain_without_missing = sum(ismissing(augtrain_without_missing));
```

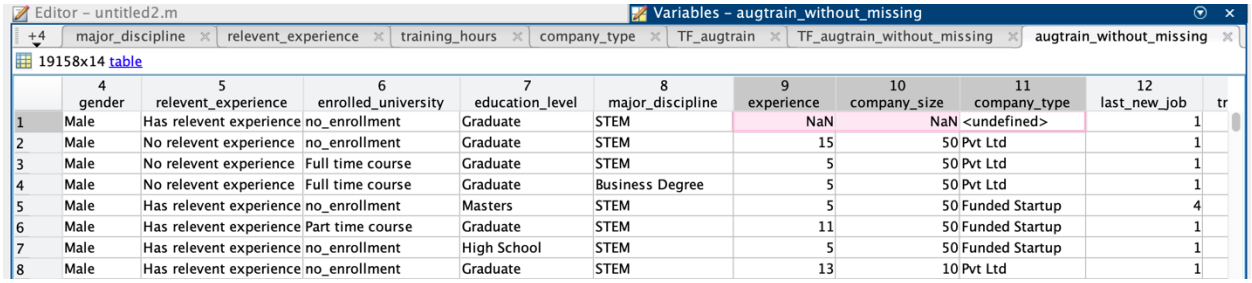
输出：



	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
1	0	0	0	0	0	0	0	0	1	1	1	0	0	0	
2															
3															

结果：9-11 列仍然有缺省值。

原因：



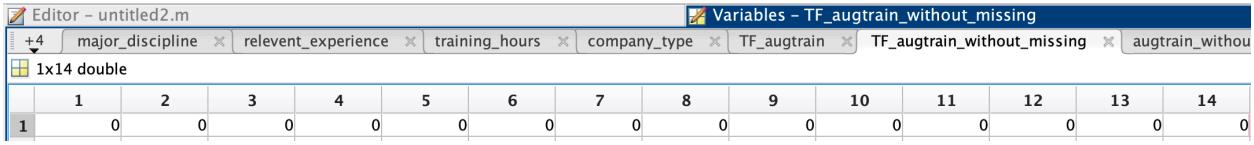
	4 gender	5 relevent_experience	6 enrolled_university	7 education_level	8 major_discipline	9 experience	10 company_size	11 company_type	12 last_new_job	tr
1	Male	Has relevent experience	no_enrollment	Graduate	STEM	NaN	NaN	<undefined>	1	
2	Male	No relevent experience	no_enrollment	Graduate	STEM	15	50	Pvt Ltd	1	
3	Male	No relevent experience	Full time course	Graduate	STEM	5	50	Pvt Ltd	1	
4	Male	No relevent experience	Full time course	Graduate	Business Degree	5	50	Pvt Ltd	1	
5	Male	Has relevent experience	no_enrollment	Masters	STEM	5	50	Funded Startup	4	
6	Male	Has relevent experience	Part time course	Graduate	STEM	11	50	Funded Startup	1	
7	Male	Has relevent experience	no_enrollment	High School	STEM	5	50	Funded Startup	1	
8	Male	Has relevent experience	no_enrollment	Graduate	STEM	13	10	Pvt Ltd	1	

由于 9-11 列的缺省值位于头部，因此利用“上一个条目的值”无法完全替换。

c) 进一步替换缺省值

由于数据较多，1/19158 的地方做微小改动产生的误差可以忽略，因此**手动**将此处的三个值用“下一个条目的值”替换。

利用先前的检验代码进行检验，所有缺省值均被替换，检验结果如下：



	1	2	3	4	5	6	7	8	9	10	11	12	13	14
1	0	0	0	0	0	0	0	0	0	0	0	0	0	0

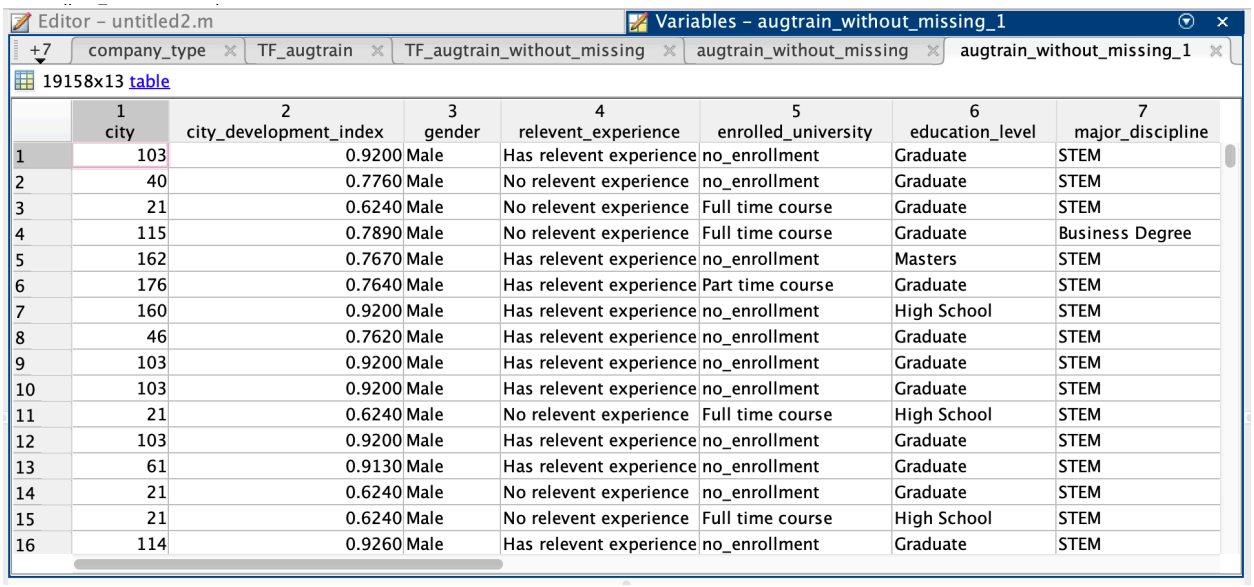
3. 去除无关数据

由于本次分析与候选人个体关系不密切，因此直接去除 enrollee_id 数据。

代码：

```
augtrain_without_missing_1 = augtrain_without_missing;
augtrain_without_missing_1(:,[1])=[];
```

输出：



	1 city	2 city_development_index	3 gender	4 relevent_experience	5 enrolled_university	6 education_level	7 major_discipline
1	103	0.9200	Male	Has relevent experience	no_enrollment	Graduate	STEM
2	40	0.7760	Male	No relevent experience	no_enrollment	Graduate	STEM
3	21	0.6240	Male	No relevent experience	Full time course	Graduate	STEM
4	115	0.7890	Male	No relevent experience	Full time course	Graduate	Business Degree
5	162	0.7670	Male	Has relevent experience	no_enrollment	Masters	STEM
6	176	0.7640	Male	Has relevent experience	Part time course	Graduate	STEM
7	160	0.9200	Male	Has relevent experience	no_enrollment	High School	STEM
8	46	0.7620	Male	Has relevent experience	no_enrollment	Graduate	STEM
9	103	0.9200	Male	Has relevent experience	no_enrollment	Graduate	STEM
10	103	0.9200	Male	Has relevent experience	no_enrollment	Graduate	STEM
11	21	0.6240	Male	No relevent experience	Full time course	High School	STEM
12	103	0.9200	Male	Has relevent experience	no_enrollment	Graduate	STEM
13	61	0.9130	Male	Has relevent experience	no_enrollment	Graduate	STEM
14	21	0.6240	Male	No relevent experience	no_enrollment	Graduate	STEM
15	21	0.6240	Male	No relevent experience	Full time course	High School	STEM
16	114	0.9260	Male	Has relevent experience	no_enrollment	Graduate	STEM

七、 计算各个 feature 和 target 的相关性

1. 将所有文本数据用 double 类型数据表示（便于计算相关系数）

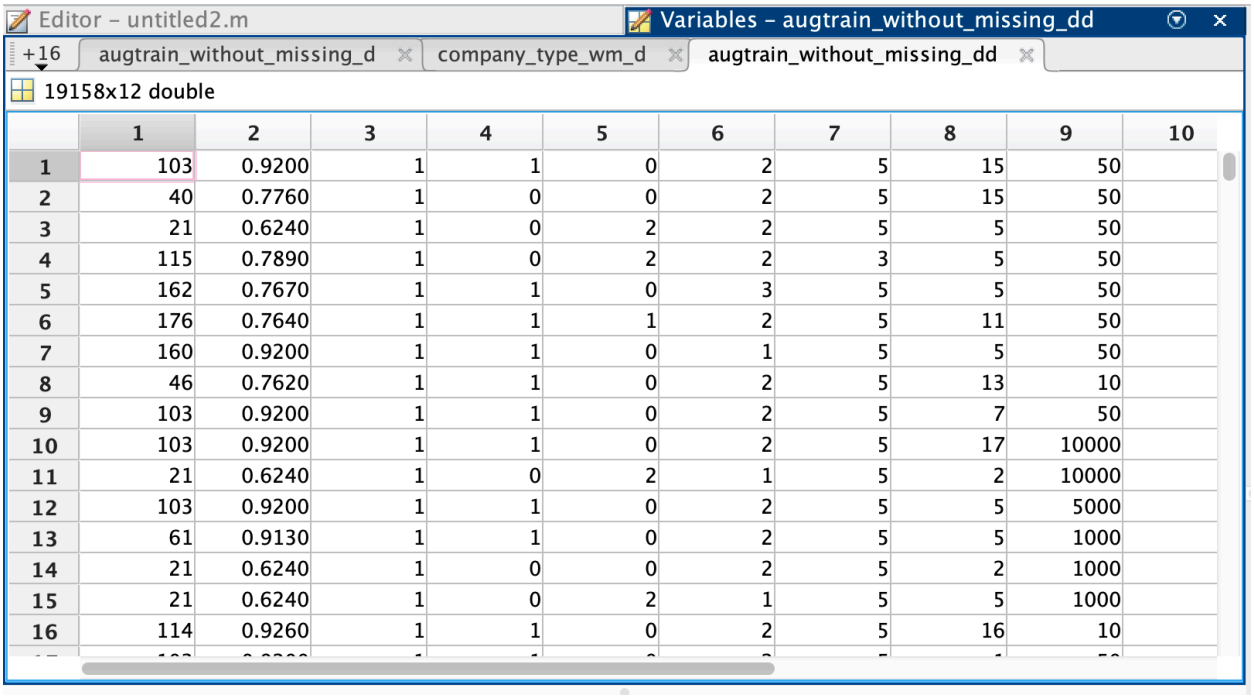
代码：

```
% 将categorical类型数据转换为用于区分的数值
gender_wm = table2array(augtrain_without_missing_1(:,[3]));
relevent_experience_wm = table2array(augtrain_without_missing_1(:,[4]));
enrolled_university_wm = table2array(augtrain_without_missing_1(:,[5]));
education_level_vm = table2array(augtrain_without_missing_1(:,[6]));
major_discipline_wm = table2array(augtrain_without_missing_1(:,[7]));
company_type_wm = table2array(augtrain_without_missing_1(:,[10]));

for i=1:19158
    if gender_wm(i)=="Male"
        gender_wm_d(i) = 1;
    else
        gender_wm_d(i) = 0;
    end
end
gender_wm_d = gender_wm_d';

for i=1:19158
    if relevent_experience_wm(i)=="Has relevent experience"
        relevent_experience_wm_d(i) = 1;
    else
        relevent_experience_wm_d(i) = 0;
    end
end
relevent_experience_wm_d = relevent_experience_wm_d';

for i=1:19158
    if enrolled_university_wm(i)=="Full time course"
        enrolled_university_wm_d(i) = 2;
    elseif enrolled_university_wm(i)=="Part time course"
        enrolled_university_wm_d(i) = 1;
```

```
    else
        enrolled_university_wm_d(i) = 0;
    end
end
enrolled_university_wm_d = enrolled_university_wm_d';

for i=1:19158
    if education_level_vm(i)=="Phd"
        education_level_vm_d(i) = 4;
    elseif education_level_vm(i)=="Masters"
        education_level_vm_d(i) = 3;
    elseif education_level_vm(i)=="Graduate"
        education_level_vm_d(i) = 2;
    elseif education_level_vm(i)=="High School"
        education_level_vm_d(i) = 1;
    else
        education_level_vm_d(i) = 0;
    end
end
education_level_vm_d = education_level_vm_d';

for i=1:19158
    if major_discipline_wm(i)=="STEM"
        major_discipline_wm_d(i) = 5;
    elseif major_discipline_wm(i)=="Arts"
        major_discipline_wm_d(i) = 4;
    elseif major_discipline_wm(i)=="Business Degree"
        major_discipline_wm_d(i) = 3;
    elseif major_discipline_wm(i)=="Humanities"
        major_discipline_wm_d(i) = 2;
    elseif major_discipline_wm(i)=="Other"
        major_discipline_wm_d(i) = 1;
    else
        major_discipline_wm_d(i) = 0;
    end
end
major_discipline_wm_d = major_discipline_wm_d';

for i=1:19158
    if company_type_wm(i)=="Pvt Ltd"
        company_type_wm_d(i) = 5;
    elseif company_type_wm(i)=="Early Stage Startup"
        company_type_wm_d(i) = 4;
    elseif company_type_wm(i)=="Public Sector"
        company_type_wm_d(i) = 3;
    elseif company_type_wm(i)=="NGO"
        company_type_wm_d(i) = 2;
    elseif company_type_wm(i)=="Funded Startup"
        company_type_wm_d(i) = 1;
    else
        company_type_wm_d(i) = 0;
    end
end
company_type_wm_d = company_type_wm_d';

%%
city_wm = table2array(augtrain_without_missing_1(:,[1]));
```

```
city_development_index_wm = table2array(augtrain_without_missing_1(:,[2]));
experience_wm = table2array(augtrain_without_missing_1(:,[8]));
company_size_wm = table2array(augtrain_without_missing_1(:,[9]));
last_new_job_wm = table2array(augtrain_without_missing_1(:,[11]));
training_hours_wm = table2array(augtrain_without_missing_1(:,[12]));

%%
augtrain_without_missing_dd                                                    =
[city_wm,city_development_index_wm,gender_wm_d,relevent_experience_wm_d,enrol
led_university_wm_d,education_level_vm_d,major_discipline_wm_d,experience_wm,
company_size_wm,company_type_wm_d,last_new_job_wm,training_hours_wm];
```

输出：



	1	2	3	4	5	6	7	8	9	10
1	103	0.9200	1	1	0	2	5	15	50	
2	40	0.7760	1	0	0	2	5	15	50	
3	21	0.6240	1	0	2	2	5	5	50	
4	115	0.7890	1	0	2	2	3	5	50	
5	162	0.7670	1	1	0	3	5	5	50	
6	176	0.7640	1	1	1	2	5	11	50	
7	160	0.9200	1	1	0	1	5	5	50	
8	46	0.7620	1	1	0	2	5	13	10	
9	103	0.9200	1	1	0	2	5	7	50	
10	103	0.9200	1	1	0	2	5	17	10000	
11	21	0.6240	1	0	2	1	5	2	10000	
12	103	0.9200	1	1	0	2	5	5	5000	
13	61	0.9130	1	1	0	2	5	5	1000	
14	21	0.6240	1	0	0	2	5	2	1000	
15	21	0.6240	1	0	2	1	5	5	1000	
16	114	0.9260	1	1	0	2	5	16	10	

2. 分别计算每个 feature 和 target 的相关性

代码：

```
for i=1:12
    r_all(i)                                                                   =
abs(corr(augtrain_without_missing_dd(:,[i]),target,'type','Spearman'));
end
r_all = r_all'
```

输出：

```
r_all =

    0.1308
    0.2792
    0.0245
    0.1284
    0.1436
    0.0190
```

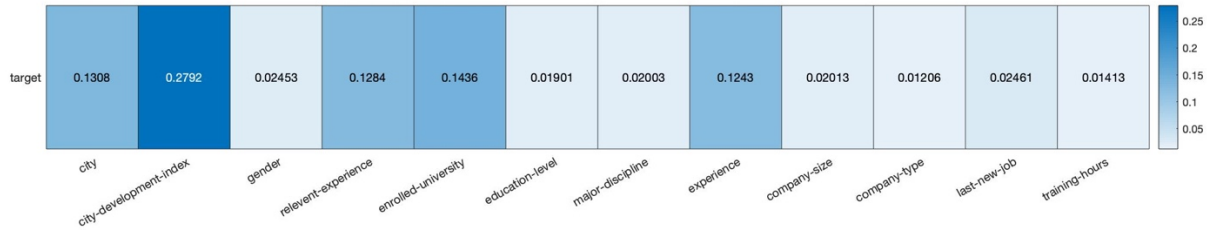
```
    0.0200
    0.1243
    0.0201
    0.0121
    0.0246
    0.0141
```

3. 绘制相关性的热力图

代码：

```
yvalues = {'target'};
xvalues          =          {'city','city-development-index','gender','relevent-
experience','enrolled-university','education-level','major-
discipline','experience','company-size','company-type','last-new-
job','training-hours'};
h = heatmap(xvalues,yvalues,r_all);
```

输出：



结论：最终寻求工作变化的结果（target）与按比例的城市的发展指数（city_development_index）的相关性最高，与城市（city）、候选人相关经验（relevent_experience）、大学课程的类型（enrolled-university）和以往总工作经验时间（experience）有较高的相关性，与其余的指标相关性较低。

八、 数据预处理和探索性数据分析

1. 城市（City）

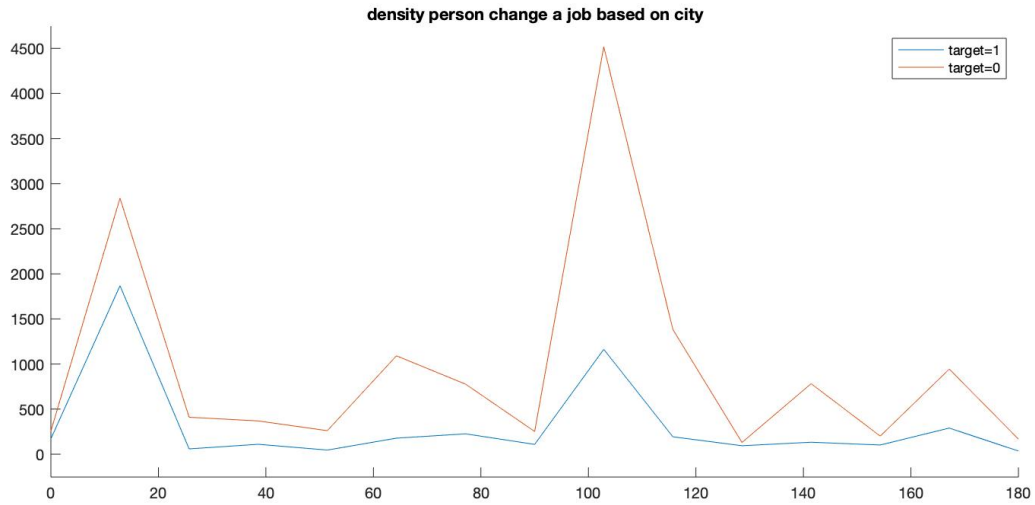
通过分析数据，用图表直观反映“是否更换工作（target）”与“城市（city）”的关系。

代码：

```
% city
city_wm_1 = [];
city_wm_0 = [];
for i=1:19158
    if target(i)==1
        city_wm_1 = [city_wm_1,city_wm(i)];
    else
        city_wm_0 = [city_wm_0,city_wm(i)];
    end
end
hold on;
x = 0:180/14:180
a = hist(city_wm_1',15);
plot(x,a)
```

```
b = hist(city_wm_0',15);
plot(x,b)
legend("target=1","target=0")
title("density person change a job based on city")
```

输出：



结论：数量在 20-30 左右的城市具有峰值密度，该处的候选人有较大几率更换工作。

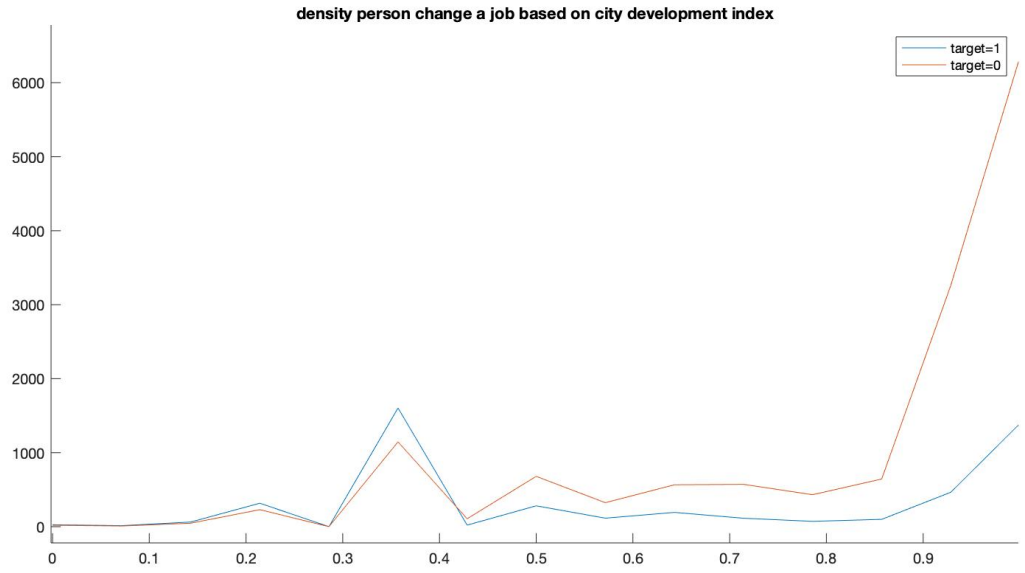
2. 按比例的城市的发展指数（City Development Index）

通过分析数据，用图表直观反映“是否更换工作（target）”与“按比例的城市的发展指数（City Development Index）”的关系。

代码：

```
% city development index
city_development_index_1 = [];
city_development_index_0 = [];
for i=1:19158
    if target(i)==1
        city_development_index_1 =
[city_development_index_1,city_development_index_wm(i)];
    else
        city_development_index_0 =
[city_development_index_0,city_development_index_wm(i)];
    end
end
hold on;
x = 0:1/14:1
a = hist(city_development_index_1',15);
plot(x,a)
b = hist(city_development_index_0',15);
plot(x,b)
legend("target=1","target=0")
title("density person change a job based on city development index")
```

输出：



结论：发展指数较高的城市不太可能有想换工作的人（target=0）。

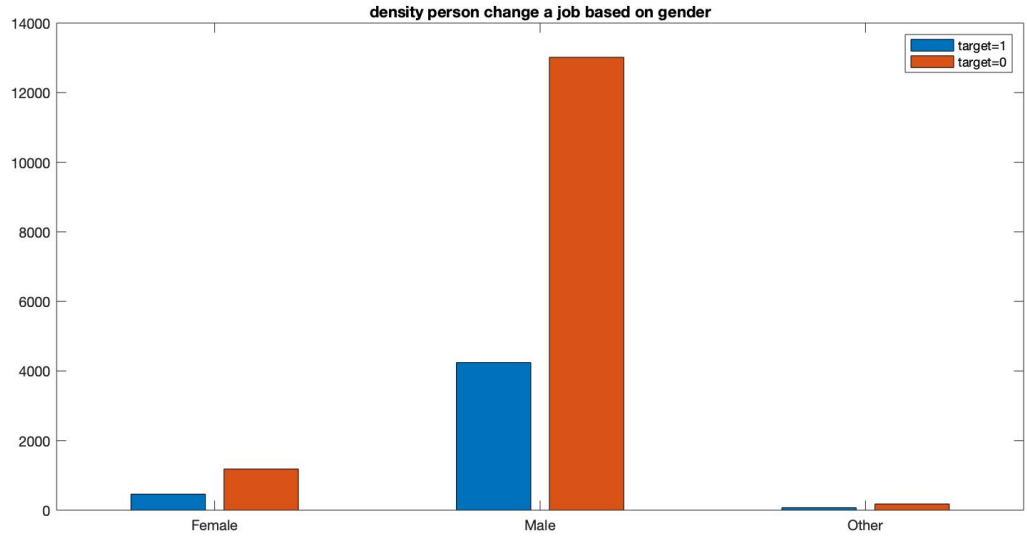
3. 性别（gender）

通过分析数据，用图表直观反映“是否更换工作（target）”与“性别（gender）”的关系。

代码：

```
% gender
gender_1 = [];
gender_0 = [];
for i=1:19158
    if target(i)==1
        gender_1 = [gender_1,gender_wm(i)];
    else
        gender_0 = [gender_0,gender_wm(i)];
    end
end
Y1 = hist(gender_1);
Y2 = hist(gender_0);
bar([Y1;Y2]');
set(gca,'XTickLabel',{'Female','Male','Other'})
legend("target=1","target=0")
title("density person change a job based on gender")
```

输出：



结论：男性的候选人较多，而女性更换工作的比例较高。

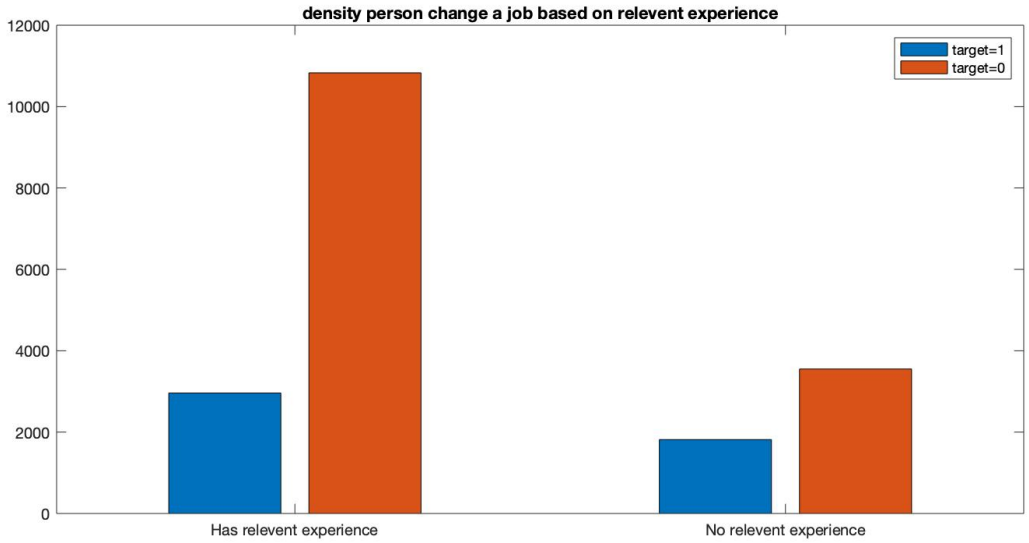
4. 相关经验（Relevant Experience）

通过分析数据，用图表直观反映“是否更换工作（target）”与“相关经验（Relevant Experience）”的关系。

代码：

```
% relevent_experience
relevent_experience_1 = [];
relevent_experience_0 = [];
for i=1:19158
    if target(i)==1
        relevent_experience_1 =
[relevent_experience_1,relevent_experience_wm(i)];
    else
        relevent_experience_0 =
[relevent_experience_0,relevent_experience_wm(i)];
    end
end
relevent_experience_Y1 = hist(relevent_experience_1);
relevent_experience_Y2 = hist(relevent_experience_0);
bar([relevent_experience_Y1;relevent_experience_Y2]');
set(gca,'XTickLabel',{'Has relevent experience','No relevent experience'})
legend("target=1","target=0")
title("density person change a job based on relevent experience")
```

输出：



target / relevent experience	0	1
0	66.19	33.81
1	78.52	21.48

结论：没有数据科学经验的人更有可能转为数据科学家，在所有参加培训的人中百分比为 33%。

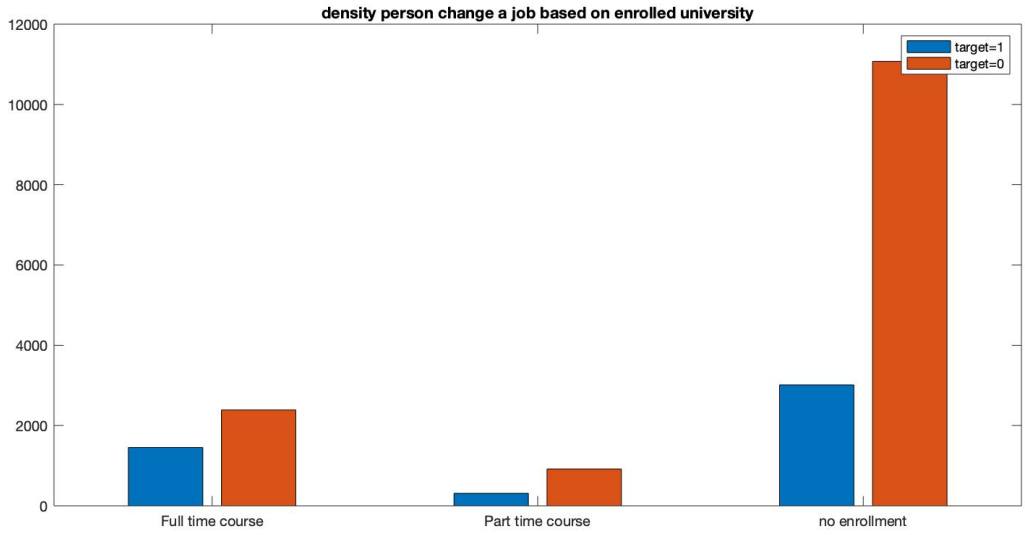
5. 大学课程的类型（enrolled university）

通过分析数据，用图表直观反映“是否更换工作（target）”与“大学课程的类型（enrolled university）”的关系。

代码：

```
% enrolled university
enrolled_university_1 = [];
enrolled_university_0 = [];
for i=1:19158
    if target(i)==1
        enrolled_university_1 =
[enrolled_university_1,enrolled_university_wm(i)];
    else
        enrolled_university_0 =
[enrolled_university_0,enrolled_university_wm(i)];
    end
end
enrolled_university_Y1 = hist(enrolled_university_1);
enrolled_university_Y2 = hist(enrolled_university_0);
bar([enrolled_university_Y1;enrolled_university_Y2]');
set(gca,'XTickLabel',{'Full time course','Part time course','no enrollment'})
legend("target=1","target=0")
title("density person change a job based on enrolled university")
```

输出：



target	0	1
enrolled_university		
Full time course	62.06	37.94
Part time course	74.79	25.21
no_enrollment	78.53	21.47

结论：参加培训完整课程的人是有意向改变其工作的人，百分比为 37.94%。

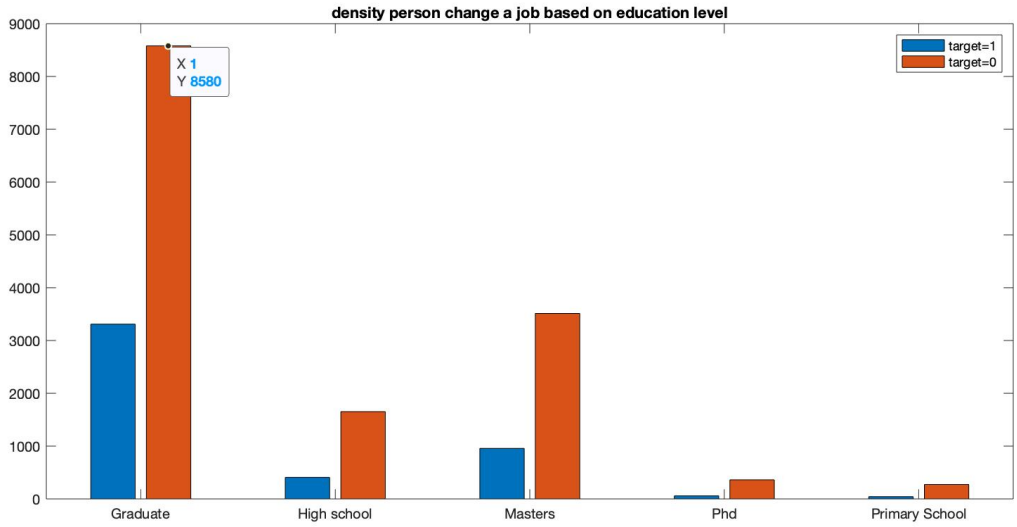
6. 候选人的教育水平（education level）

通过分析数据，用图表直观反映“是否更换工作（target）”与“候选人的教育水平（education level）”的关系。

代码：

```
% education level
education_level_1 = [];
education_level_0 = [];
for i=1:19158
    if target(i)==1
        education_level_1 = [education_level_1,education_level_vm(i)];
    else
        education_level_0 = [education_level_0,education_level_vm(i)];
    end
end
education_level_Y1 = hist(education_level_1);
education_level_Y2 = hist(education_level_0);
bar([education_level_Y1;education_level_Y2]');
set(gca,'XTickLabel',{'Graduate','High school','Masters','Phd','Primary
School'})
legend("target=1","target=0")
title("density person change a job based on education level")
```

输出：



target	0	1
education_level		
Graduate	72.21	27.79
Masters	78.54	21.46
High School	80.49	19.51
Phd	85.99	14.01
Primary School	86.56	13.44

结论：毕业生更容易改变工作成为数据科学家。

7. 候选人的教育专业学科（major discipline）

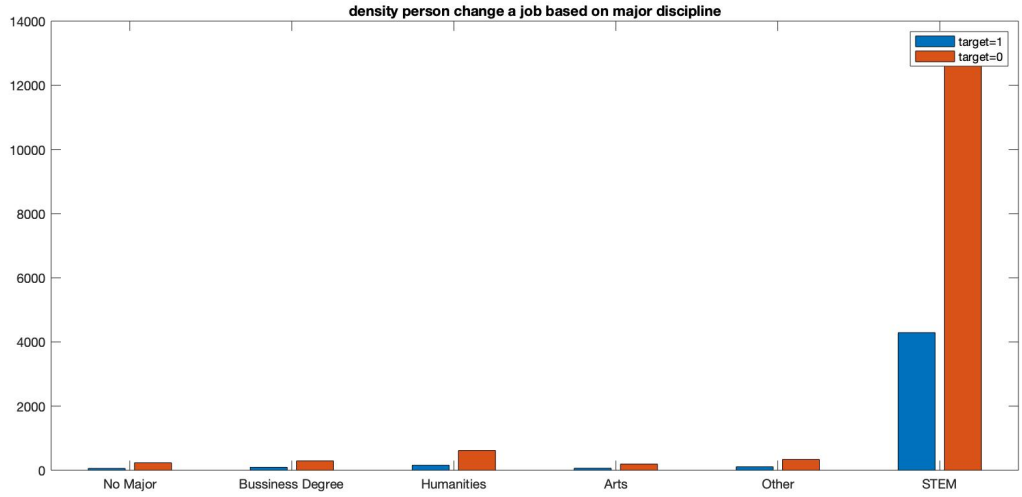
通过分析数据，用图表直观反映“是否更换工作（target）”与“候选人的教育专业学科（major discipline）”的关系。

代码：

```
% major discipline wm
major_discipline_1 = [];
major_discipline_0 = [];
for i=1:19158
    if target(i)==1
        major_discipline_1 = [major_discipline_1,major_discipline_wm(i)];
    else
        major_discipline_0 = [major_discipline_0,major_discipline_wm(i)];
    end
end
major_discipline_Y1 = hist(major_discipline_1);
major_discipline_Y2 = hist(major_discipline_0);
bar([major_discipline_Y1;major_discipline_Y2]');
set(gca,'XTickLabel',{'No Major','Bussiness
Degree','Humanities','Arts','Other','STEM'})
legend("target=1","target=0")
```

```
title("density person change a job based on major discipline")
```

输出：



target	0	1
major_discipline		
Other	73.23	26.77
Business Degree	73.7	26.3
STEM	74.91	25.09
No Major	75.34	24.66
Humanities	78.92	21.08
Arts	79.05	20.95

结论：是否想换职业与候选人的教育专业学科关系不大。

8. 候选人的总工作经验（experience）

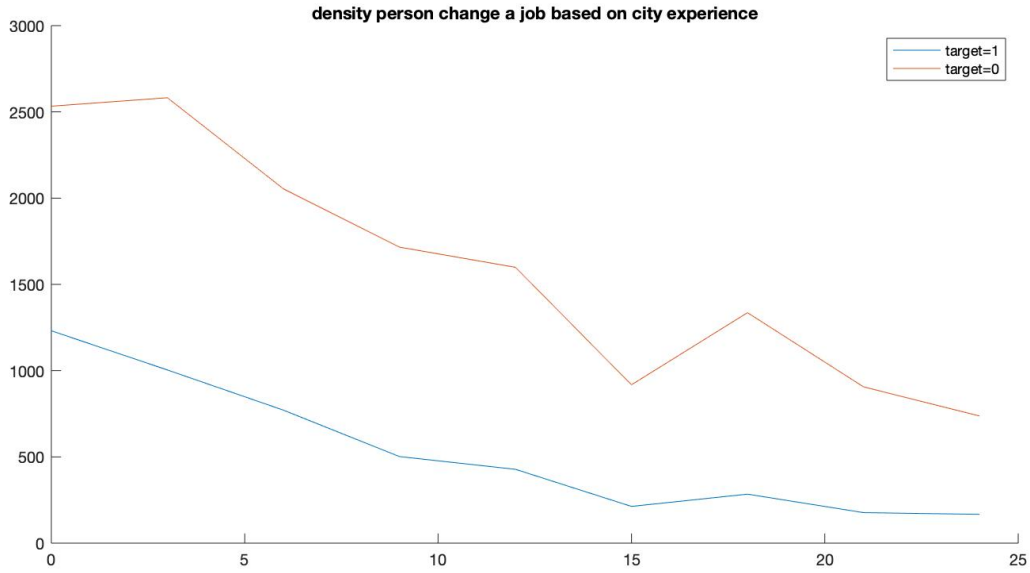
通过分析数据，用图表直观反映“是否更换工作（target）”与“候选人的总工作经验（experience）”的关系。

代码：

```
% experience
experience_1 = [];
experience_0 = [];
for i=1:19158
    if target(i)==1
        experience_1 = [experience_1,experience_wm(i)];
    else
        experience_0 = [experience_0,experience_wm(i)];
    end
end
hold on;
x = 0:24/8:24
a = hist(experience_1',9);
```

```
plot(x,a)
b = hist(experience_0',9);
plot(x,b)
legend("target=1","target=0")
title("density person change a job based on city experience")
```

输出：



结论：有 1-5 年工作经验的人更有可能更换工作，之后的趋势是下降。

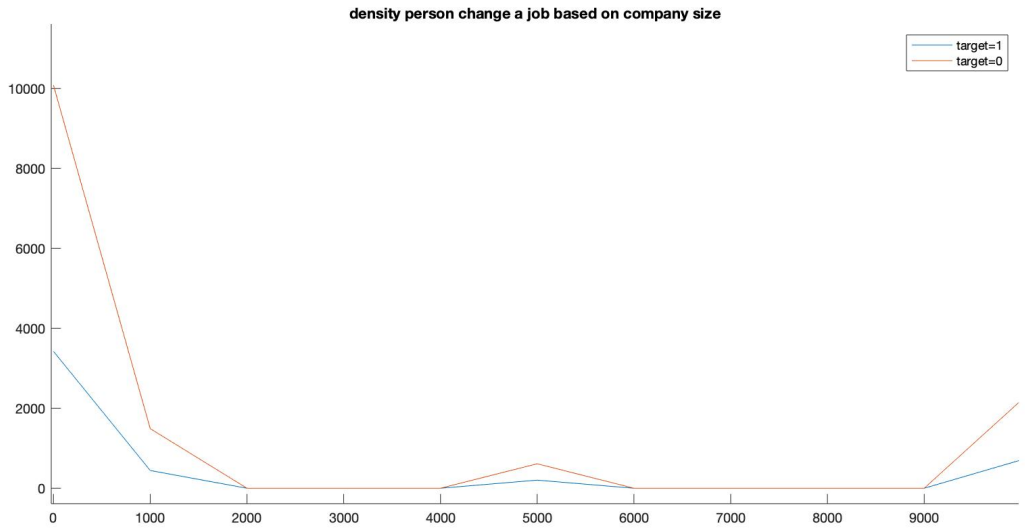
9. 目前雇主公司的员工数量（company_size）

通过分析数据，用图表直观反映“是否更换工作（target）”与“目前雇主公司的员工数量（company_size）”的关系。

代码：

```
% company size
company_size_1 = [];
company_size_0 = [];
for i=1:19158
    if target(i)==1
        company_size_1 = [company_size_1,company_size_wm(i)];
    else
        company_size_0 = [company_size_0,company_size_wm(i)];
    end
end
hold on;
x = 0:10000/10:10000
a = hist(company_size_1',11);
plot(x,a)
b = hist(company_size_0',11);
plot(x,b)
legend("target=1","target=0")
title("density person change a job based on company size")
```

输出：



结论：在公司规模为 3 级，即包含 50-99 人的公司工作的人，改变工作的密度最高。

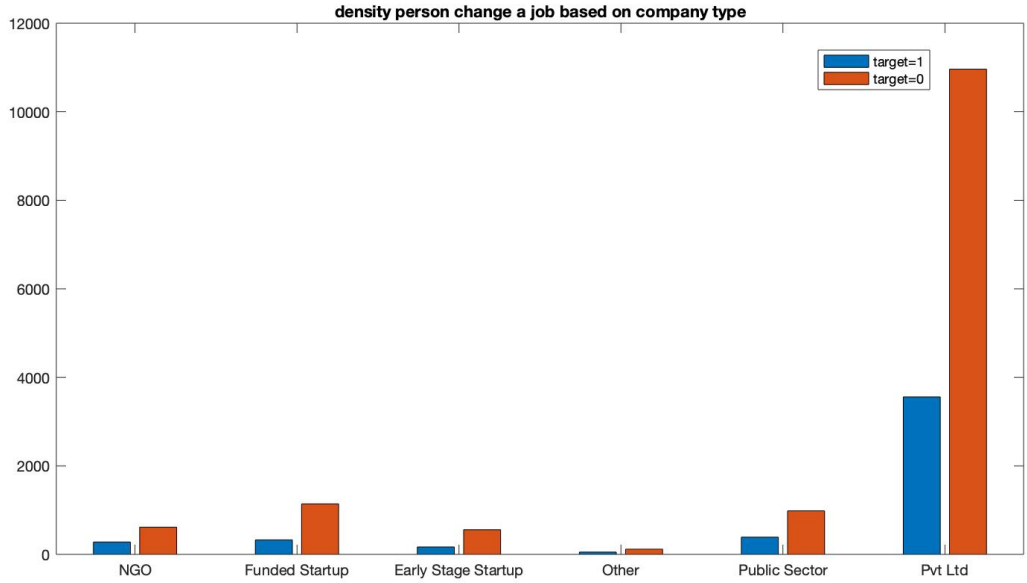
10. 当前雇主的类型（company_type）

通过分析数据，用图表直观反映“是否更换工作（target）”与“当前雇主的类型（company_type）”的关系。

代码：

```
% company type
company_type_1 = [];
company_type_0 = [];
for i=1:19158
    if target(i)==1
        company_type_1 = [company_type_1,company_type_wm(i)];
    else
        company_type_0 = [company_type_0,company_type_wm(i)];
    end
end
company_type_Y1 = hist(company_type_1);
company_type_Y2 = hist(company_type_0);
bar([company_type_Y1;company_type_Y2]');
set(gca,'XTickLabel',{'NGO','Funded Startup','Early Stage
Startup','Other','Public Sector','Pvt Ltd'})
legend("target=1","target=0")
title("density person change a job based on company type")
```

输出：



结论：在私人公司工作的人，更换工作的比例最高。

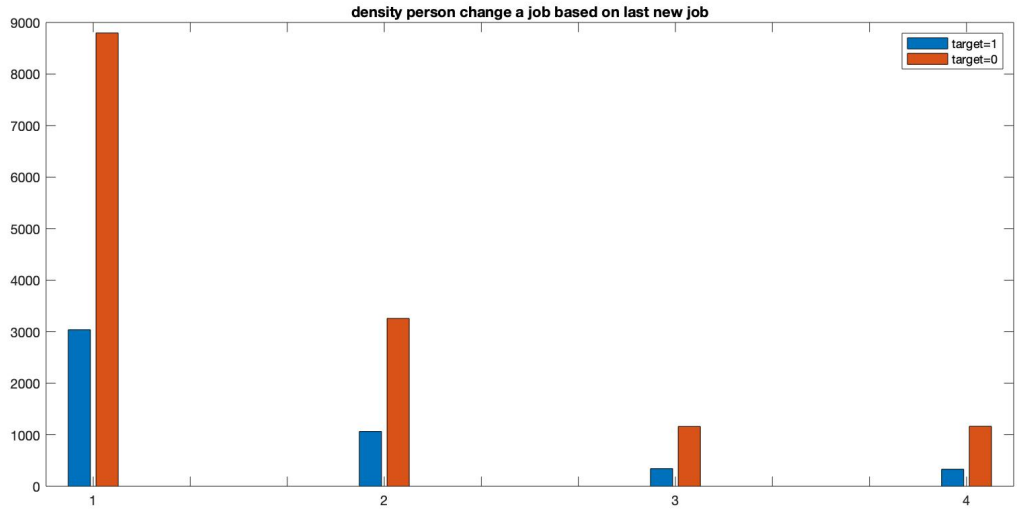
11. 上一份工作与当前工作的年限差值（last new job）

通过分析数据，用图表直观反映“是否更换工作（target）”与“上一份工作与当前工作的年限差值（last new job）”的关系。

代码：

```
% last new job
last_new_job_1 = [];
last_new_job_0 = [];
for i=1:19158
    if target(i)==1
        last_new_job_1 = [last_new_job_1,last_new_job_wm(i)];
    else
        last_new_job_0 = [last_new_job_0,last_new_job_wm(i)];
    end
end
last_new_job_Y1 = hist(last_new_job_1);
last_new_job_Y2 = hist(last_new_job_0);
bar([last_new_job_Y1;last_new_job_Y2]');
set(gca,'XTickLabel',{'1','','','2','','','3','','','4','',''})
legend("target=1","target=0")
title("density person change a job based on last new job")
```

输出：



结论：从来没有工作的人/刚毕业的人在参加课程后倾向于成为数据科学家。其次是在上一份工作中工作了一年的人。

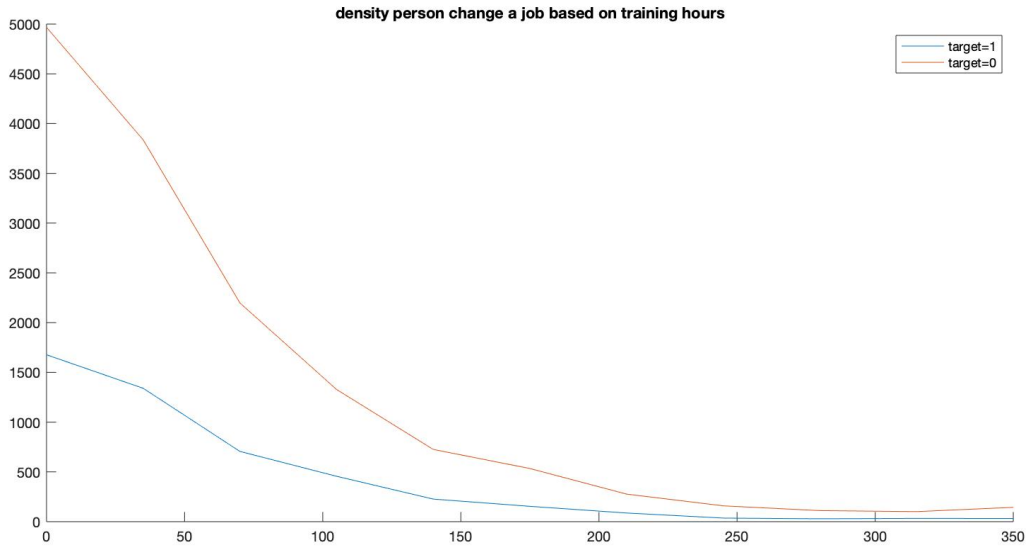
12. 完成的培训时间（training_hours）

通过分析数据，用图表直观反映“是否更换工作（target）”与“完成的培训时间（training_hours）”的关系。

代码：

```
% training hours
training_hours_1 = [];
training_hours_0 = [];
for i=1:19158
    if target(i)==1
        training_hours_1 = [training_hours_1,training_hours_wm(i)];
    else
        training_hours_0 = [training_hours_0,training_hours_wm(i)];
    end
end
hold on;
x = 0:350/10:350
a = hist(training_hours_1',11);
plot(x,a)
b = hist(training_hours_0',11);
plot(x,b)
legend("target=1","target=0")
title("density person change a job based on training hours")
```

输出：



结论：参加培训达 25 小时左右的人，倾向于将他们的工作改为数据科学家。

九、 SVM 二分类算法分析

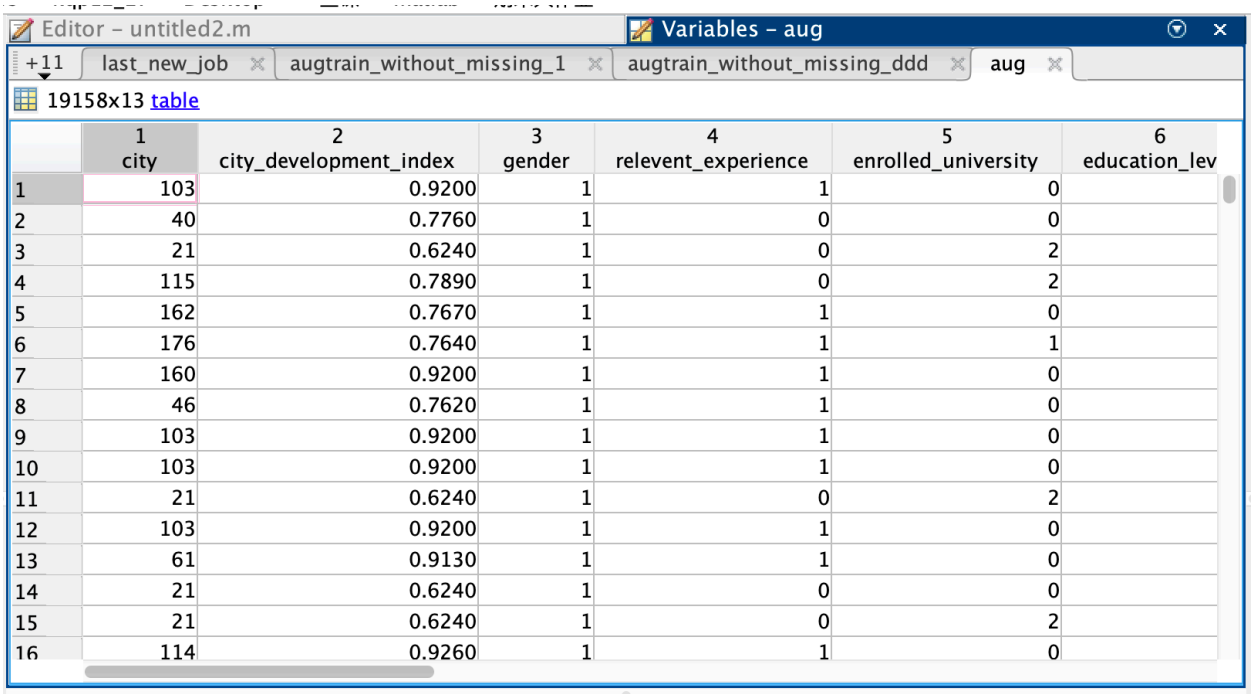
采用 SVM（支持向量机）二分类算法对数据进行分类。其中 Kernal 核函数分别采用“linear”类型和“rbf”类型进行训练。

首先将之前将文本字符都转换为数字的 double 类型矩阵转换为 table 类型，方便后续的分类分析。

代码：

```
% double转换为table类型
aug =
array2table(augtrain_without_missing_ddd,'VariableNames',{'city','city_develo
pment_index','gender','relevent_experience','enrolled_university','education_
level','major_discipline','experience','company_size','company_type','last_ne
w_job','training_hours','target'});
```

输出：



	1 city	2 city_development_index	3 gender	4 relevent_experience	5 enrolled_university	6 education_lev
1	103	0.9200	1	1	0	
2	40	0.7760	1	0	0	
3	21	0.6240	1	0	2	
4	115	0.7890	1	0	2	
5	162	0.7670	1	1	0	
6	176	0.7640	1	1	1	
7	160	0.9200	1	1	0	
8	46	0.7620	1	1	0	
9	103	0.9200	1	1	0	
10	103	0.9200	1	1	0	
11	21	0.6240	1	0	2	
12	103	0.9200	1	1	0	
13	61	0.9130	1	1	0	
14	21	0.6240	1	0	0	
15	21	0.6240	1	0	2	
16	114	0.9260	1	1	0	

其次进行 SVM 分类分析。

代码：（核函数为 rbf）

```
% SVM
inputTable = aug;
predictorNames = {'city', 'city_development_index', 'gender',
'relevent_experience', 'enrolled_university', 'education_level',
'major_discipline', 'experience', 'company_size', 'company_type',
'last_new_job', 'training_hours'};
predictors = inputTable(:, predictorNames);
response = inputTable.target;
isCategoricalPredictor = [false, false, true, true, true, true, true, false,
false, true, false, false];

classificationSVM = fitcsvm(...
    predictors, ...
    response, ...
    'KernelFunction', 'rbf', ...
    'PolynomialOrder', [], ...
    'KernelScale', 'auto', ...
    'BoxConstraint', 1, ...
    'Standardize', true, ...
    'ClassNames', [0; 1]);

predictorExtractionFcn = @(t) t(:, predictorNames);
svmPredictFcn = @(x) predict(classificationSVM, x);
trainedClassifier.predictFcn = @(x) svmPredictFcn(predictorExtractionFcn(x));

trainedClassifier.RequiredVariables = {'city', 'city_development_index',
'company_size', 'company_type', 'education_level', 'enrolled_university',
'experience', 'gender', 'last_new_job', 'major_discipline',
'relevent_experience', 'training_hours'};
```

```
trainedClassifier.ClassificationSVM = classificationSVM;

inputTable = aug;
predictorNames = {'city', 'city_development_index', 'gender',
'relevent_experience', 'enrolled_university', 'education_level',
'major_discipline', 'experience', 'company_size', 'company_type',
'last_new_job', 'training_hours'};
predictors = inputTable(:, predictorNames);
response = inputTable.target;
isCategoricalPredictor = [false, false, true, true, true, true, true, false,
false, true, false, false];

% cross-validation
partitionedModel = crossval(trainedClassifier.ClassificationSVM, 'KFold', 4)

% Compute validation predictions
[validationPredictions, validationScores] = kfoldPredict(partitionedModel);

% Compute validation accuracy
validationAccuracy = 1 - kfoldLoss(partitionedModel, 'LossFun',
'ClassifError')
```

输出：（核函数为 rbf）

```
partitionedModel =

  classreg.learning.partition.ClassificationPartitionedModel
    CrossValidatedModel: 'SVM'
         PredictorNames: {1×12 cell}
           ResponseName: 'Y'
        NumObservations: 19158
                  KFold: 4
              Partition: [1×1 cvpartition]
             ClassNames: [0 1]
         ScoreTransform: 'none'


validationAccuracy =

    0.7587
```

之后将核函数变为“linear”方法（代码中标红处）

输出：（核函数为 linear）

```
partitionedModel =

  classreg.learning.partition.ClassificationPartitionedModel
    CrossValidatedModel: 'SVM'
         PredictorNames: {1×12 cell}
           ResponseName: 'Y'
        NumObservations: 19158
                  KFold: 4
              Partition: [1×1 cvpartition]
             ClassNames: [0 1]
         ScoreTransform: 'none'
```

```
validationAccuracy =

    0.7507
```

结论：核函数为 rbf 的 SVM 分类效果更好，准确率达到 75.87%。

十、 KNN 分类算法分析

1. Linear KNN

代码：

```
% Linear KNN
inputTable = aug;
predictorNames = {'city', 'city_development_index', 'gender',
'relevent_experience', 'enrolled_university', 'education_level',
'major_discipline', 'experience', 'company_size', 'company_type',
'last_new_job', 'training_hours'};
predictors = inputTable(:, predictorNames);
response = inputTable.target;
isCategoricalPredictor = [false, false, false, false, false, false, false,
false, false, false, false, false];

classificationKNN = fitcknn(...
    predictors, ...
    response, ...
    'Distance', 'Euclidean', ...
    'Exponent', [], ...
    'NumNeighbors', 1, ...
    'DistanceWeight', 'Equal', ...
    'Standardize', true, ...
    'ClassNames', [0; 1]);

predictorExtractionFcn = @(t) t(:, predictorNames);
knnPredictFcn = @(x) predict(classificationKNN, x);
trainedClassifier.predictFcn = @(x) knnPredictFcn(predictorExtractionFcn(x));

trainedClassifier.RequiredVariables = {'city', 'city_development_index',
'company_size', 'company_type', 'education_level', 'enrolled_university',
'experience', 'gender', 'last_new_job', 'major_discipline',
'relevent_experience', 'training_hours'};
trainedClassifier.ClassificationKNN = classificationKNN;

inputTable = aug;
predictorNames = {'city', 'city_development_index', 'gender',
'relevent_experience', 'enrolled_university', 'education_level',
'major_discipline', 'experience', 'company_size', 'company_type',
'last_new_job', 'training_hours'};
predictors = inputTable(:, predictorNames);
response = inputTable.target;
isCategoricalPredictor = [false, false, false, false, false, false, false,
false, false, false, false, false];

% Perform cross-validation
partitionedModel = crossval(trainedClassifier.ClassificationKNN, 'KFold', 5)

% Compute validation predictions
[validationPredictions, validationScores] = kfoldPredict(partitionedModel);
```

```
% Compute validation accuracy
validationAccuracy = 1 - kfoldLoss(partitionedModel, 'LossFun',
'ClassifError')
```

输出：

```
partitionedModel =

  classreg.learning.partition.ClassificationPartitionedModel
    CrossValidatedModel: 'KNN'
         PredictorNames: {1×12 cell}
           ResponseName: 'Y'
        NumObservations: 19158
                  KFold: 5
              Partition: [1×1 cvpartition]
             ClassNames: [0 1]
         ScoreTransform: 'none'


  Properties, Methods


validationAccuracy =

    0.6922
```

2. Medium KNN

代码：

```
% Medium KNN
inputTable = aug;
predictorNames = {'city', 'city_development_index', 'gender',
'relevent_experience', 'enrolled_university', 'education_level',
'major_discipline', 'experience', 'company_size', 'company_type',
'last_new_job', 'training_hours'};
predictors = inputTable(:, predictorNames);
response = inputTable.target;
isCategoricalPredictor = [false, false, false, false, false, false, false,
false, false, false, false, false];

classificationKNN = fitcknn(...
    predictors, ...
    response, ...
    'Distance', 'Euclidean', ...
    'Exponent', [], ...
    'NumNeighbors', 10, ...
    'DistanceWeight', 'Equal', ...
    'Standardize', true, ...
    'ClassNames', [0; 1]);

predictorExtractionFcn = @(t) t(:, predictorNames);
knnPredictFcn = @(x) predict(classificationKNN, x);
trainedClassifier.predictFcn = @(x) knnPredictFcn(predictorExtractionFcn(x));

trainedClassifier.RequiredVariables = {'city', 'city_development_index',
'company_size', 'company_type', 'education_level', 'enrolled_university',
```

```
'experience', 'gender', 'last_new_job', 'major_discipline',
'relevent_experience', 'training_hours'};
trainedClassifier.ClassificationKNN = classificationKNN;

inputTable = aug;
predictorNames = {'city', 'city_development_index', 'gender',
'relevent_experience', 'enrolled_university', 'education_level',
'major_discipline', 'experience', 'company_size', 'company_type',
'last_new_job', 'training_hours'};
predictors = inputTable(:, predictorNames);
response = inputTable.target;
isCategoricalPredictor = [false, false, false, false, false, false, false,
false, false, false, false, false];

% Perform cross-validation
partitionedModel = crossval(trainedClassifier.ClassificationKNN, 'KFold', 5)

% Compute validation predictions
[validationPredictions, validationScores] = kfoldPredict(partitionedModel);

% Compute validation accuracy
validationAccuracy = 1 - kfoldLoss(partitionedModel, 'LossFun',
'ClassifError')
```

输出：

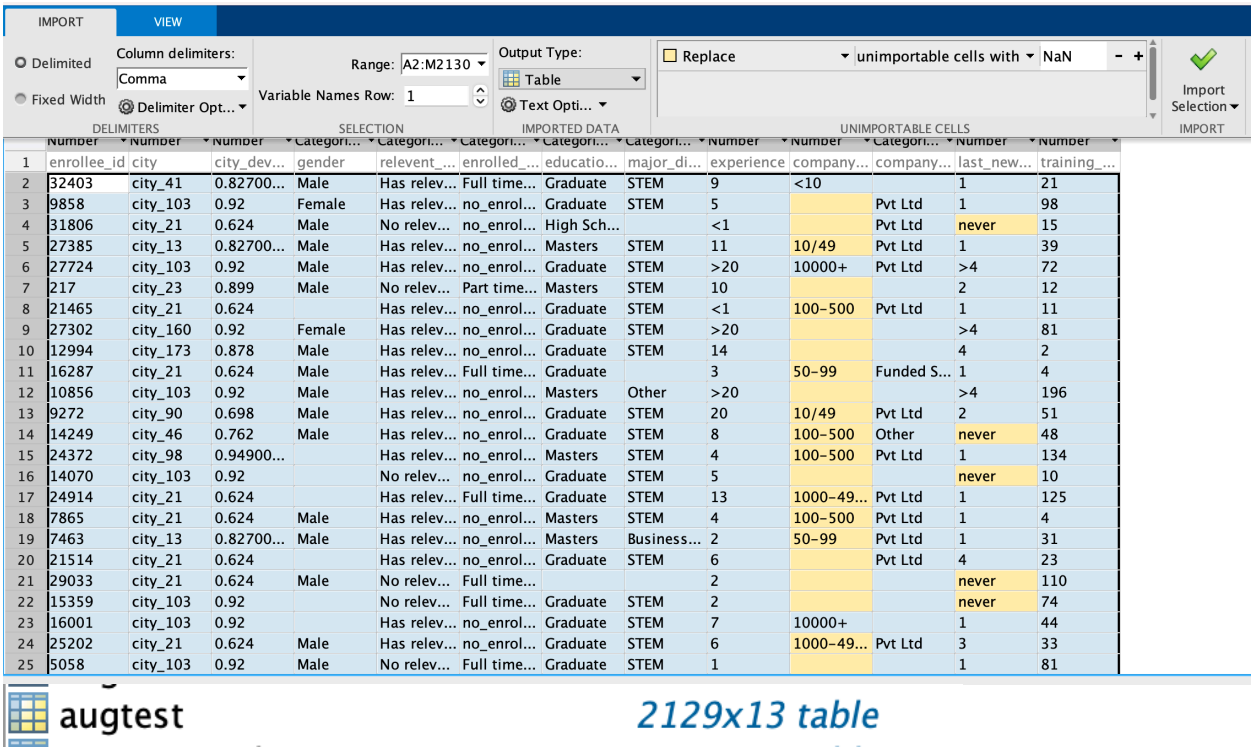
```
partitionedModel =

  classreg.learning.partition.ClassificationPartitionedModel
    CrossValidatedModel: 'KNN'
         PredictorNames: {1×12 cell}
           ResponseName: 'Y'
        NumObservations: 19158
                  KFold: 5
              Partition: [1×1 cvpartition]
             ClassNames: [0 1]
         ScoreTransform: 'none'


  Properties, Methods


validationAccuracy =

    0.7620
```

结论：Medium KNN 分类效果较 Linear KNN 分类效果好，准确率达到 76.2%。

十一、 运用训练好的模型对 aug_test 中的数据进行预测

1. 导入 aug_test.csv 的数据

以 table 类型导入数据为 aug_test。（由于此数据集中没有最后 target 列的数据，因此 train 数据有 14 列，这里的 test 数据只有 13 列）。



	enrollee_id	city	city_dev...	gender	relevent_...	enrolled_...	educatio...	major_di...	experience	company...	company...	last_new...	training_...
2	32403	city_41	0.82700...	Male	Has relev...	Full time...	Graduate	STEM	9	<10		1	21
3	9858	city_103	0.92	Female	Has relev...	no_enrol...	Graduate	STEM	5		Pvt Ltd	1	98
4	31806	city_21	0.624	Male	No relev...	no_enrol...	High Sch...		<1		Pvt Ltd	never	15
5	27385	city_13	0.82700...	Male	Has relev...	no_enrol...	Masters	STEM	11	10/49	Pvt Ltd	1	39
6	27724	city_103	0.92	Male	Has relev...	no_enrol...	Graduate	STEM	>20	10000+	Pvt Ltd	>4	72
7	217	city_23	0.899	Male	No relev...	Part time...	Masters	STEM	10			2	12
8	21465	city_21	0.624		Has relev...	no_enrol...	Graduate	STEM	<1	100-500	Pvt Ltd	1	11
9	27302	city_160	0.92	Female	Has relev...	no_enrol...	Graduate	STEM	>20			>4	81
10	12994	city_173	0.878	Male	Has relev...	no_enrol...	Graduate	STEM	14			4	2
11	16287	city_21	0.624	Male	Has relev...	Full time...	Graduate		3	50-99	Funded S...	1	4
12	10856	city_103	0.92	Male	Has relev...	no_enrol...	Masters	Other	>20			>4	196
13	9272	city_90	0.698	Male	Has relev...	no_enrol...	Graduate	STEM	20	10/49	Pvt Ltd	2	51
14	14249	city_46	0.762	Male	Has relev...	no_enrol...	Graduate	STEM	8	100-500	Other	never	48
15	24372	city_98	0.94900...		Has relev...	no_enrol...	Masters	STEM	4	100-500	Pvt Ltd	1	134
16	14070	city_103	0.92		No relev...	no_enrol...	Graduate	STEM	5			never	10
17	24914	city_21	0.624		Has relev...	Full time...	Graduate	STEM	13	1000-49...	Pvt Ltd	1	125
18	7865	city_21	0.624	Male	Has relev...	no_enrol...	Masters	STEM	4	100-500	Pvt Ltd	1	4
19	7463	city_13	0.82700...	Male	Has relev...	no_enrol...	Masters	Business...	2	50-99	Pvt Ltd	1	31
20	21514	city_21	0.624		Has relev...	no_enrol...	Graduate	STEM	6		Pvt Ltd	4	23
21	29033	city_21	0.624	Male	No relev...	Full time...			2			never	110
22	15359	city_103	0.92		No relev...	Full time...	Graduate	STEM	2			never	74
23	16001	city_103	0.92		Has relev...	no_enrol...	Graduate	STEM	7	10000+		1	44
24	25202	city_21	0.624	Male	Has relev...	no_enrol...	Graduate	STEM	6	1000-49...	Pvt Ltd	3	33
25	5058	city_103	0.92	Male	No relev...	Full time...	Graduate	STEM	1			1	81

2. 处理 aug_test.csv 的数据

与前文同理，进行数据清洗、去掉无影响的 feature、将文本类型数据转化为数值类型数据等步骤，得到可以进行测试的数据。

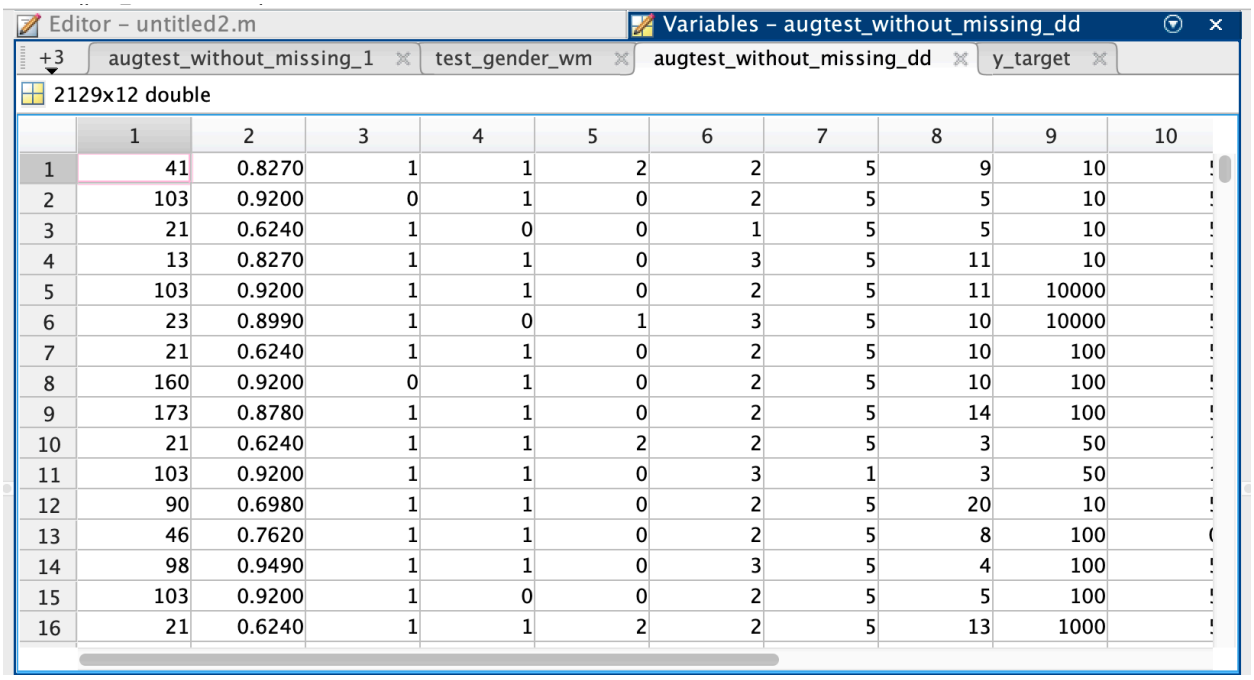
代码：

```
%%
TF_augtest = sum(ismissing(augtest));
augtest_without_missing = fillmissing(augtest,'previous');

TF_augtest_without_missing = sum(ismissing(augtest_without_missing))

%%
% Âé³Èô§enrollee_idÊï∞ÊçÆ
augtest_without_missing_1 = augtest_without_missing;
augtest_without_missing_1(:,[1])=[];

%%
% Â∞ÜcategoricalÁ±ªÂûãÊï∞ÊçÆËΩ¨Êç¢‰∏∫ Áî®‰∫éÂå ÂàÜÁöÑÊï∞ÊçÆ
test_gender_wm = table2array(augtest_without_missing_1(:,[3]));
test_relevent_experience_wm = table2array(augtest_without_missing_1(:,[4]));
test_enrolled_university_wm = table2array(augtest_without_missing_1(:,[5]));
test_education_level_vm = table2array(augtest_without_missing_1(:,[6]));
test_major_discipline_wm = table2array(augtest_without_missing_1(:,[7]));
test_company_type_wm = table2array(augtest_without_missing_1(:,[10]));

%%
for i=1:2129
    if test_gender_wm(i)=="Male"
        test_gender_wm_d(i) = 1;
```

```
    else
        test_gender_wm_d(i) = 0;
    end
end
test_gender_wm_d = test_gender_wm_d';

for i=1:2129
    if test_relevent_experience_wm(i)=="Has relevent experience"
        test_relevent_experience_wm_d(i) = 1;
    else
        test_relevent_experience_wm_d(i) = 0;
    end
end
test_relevent_experience_wm_d = test_relevent_experience_wm_d';

for i=1:2129
    if test_enrolled_university_wm(i)=="Full time course"
        test_enrolled_university_wm_d(i) = 2;
    elseif test_enrolled_university_wm(i)=="Part time course"
        test_enrolled_university_wm_d(i) = 1;
    else
        test_enrolled_university_wm_d(i) = 0;
    end
end
test_enrolled_university_wm_d = test_enrolled_university_wm_d';

for i=1:2129
    if test_education_level_vm(i)=="Phd"
        test_education_level_vm_d(i) = 4;
    elseif test_education_level_vm(i)=="Masters"
        test_education_level_vm_d(i) = 3;
    elseif test_education_level_vm(i)=="Graduate"
        test_education_level_vm_d(i) = 2;
    elseif test_education_level_vm(i)=="High School"
        test_education_level_vm_d(i) = 1;
    else
        test_education_level_vm_d(i) = 0;
    end
end
test_education_level_vm_d = test_education_level_vm_d';

for i=1:2129
    if test_major_discipline_wm(i)=="STEM"
        test_major_discipline_wm_d(i) = 5;
    elseif test_major_discipline_wm(i)=="Arts"
        test_major_discipline_wm_d(i) = 4;
    elseif test_major_discipline_wm(i)=="Business Degree"
        test_major_discipline_wm_d(i) = 3;
    elseif test_major_discipline_wm(i)=="Humanities"
        test_major_discipline_wm_d(i) = 2;
    elseif test_major_discipline_wm(i)=="Other"
        test_major_discipline_wm_d(i) = 1;
    else
        test_major_discipline_wm_d(i) = 0;
    end
end
test_major_discipline_wm_d = test_major_discipline_wm_d';
```

```
for i=1:2129
    if test_company_type_wm(i)=="Pvt Ltd"
        test_company_type_wm_d(i) = 5;
    elseif test_company_type_wm(i)=="Early Stage Startup"
        test_company_type_wm_d(i) = 4;
    elseif test_company_type_wm(i)=="Public Sector"
        test_company_type_wm_d(i) = 3;
    elseif test_company_type_wm(i)=="NGO"
        test_company_type_wm_d(i) = 2;
    elseif test_company_type_wm(i)=="Funded Startup"
        test_company_type_wm_d(i) = 1;
    else
        test_company_type_wm_d(i) = 0;
    end
end
test_company_type_wm_d = test_company_type_wm_d';

%%
test_city_wm = table2array(augtest_without_missing_1(:,[1]));
test_city_development_index_wm =
table2array(augtest_without_missing_1(:,[2]));
test_experience_wm = table2array(augtest_without_missing_1(:,[8]));
test_company_size_wm = table2array(augtest_without_missing_1(:,[9]));
test_last_new_job_wm = table2array(augtest_without_missing_1(:,[11]));
test_training_hours_wm = table2array(augtest_without_missing_1(:,[12]));

%%
augtest_without_missing_dd =
[test_city_wm,test_city_development_index_wm,test_gender_wm_d,test_relevent_e
xperience_wm_d,test_enrolled_university_wm_d,test_education_level_vm_d,test_m
ajor_discipline_wm_d,test_experience_wm,test_company_size_wm,test_company_typ
e_wm_d,test_last_new_job_wm,test_training_hours_wm];
```

输出：



	1	2	3	4	5	6	7	8	9	10
1	41	0.8270	1	1	2	2	5	9	10	
2	103	0.9200	0	1	0	2	5	5	10	
3	21	0.6240	1	0	0	1	5	5	10	
4	13	0.8270	1	1	0	3	5	11	10	
5	103	0.9200	1	1	0	2	5	11	10000	
6	23	0.8990	1	0	1	3	5	10	10000	
7	21	0.6240	1	1	0	2	5	10	100	
8	160	0.9200	0	1	0	2	5	10	100	
9	173	0.8780	1	1	0	2	5	14	100	
10	21	0.6240	1	1	2	2	5	3	50	
11	103	0.9200	1	1	0	3	1	3	50	
12	90	0.6980	1	1	0	2	5	20	10	
13	46	0.7620	1	1	0	2	5	8	100	
14	98	0.9490	1	1	0	3	5	4	100	
15	103	0.9200	1	0	0	2	5	5	100	
16	21	0.6240	1	1	2	2	5	13	1000	

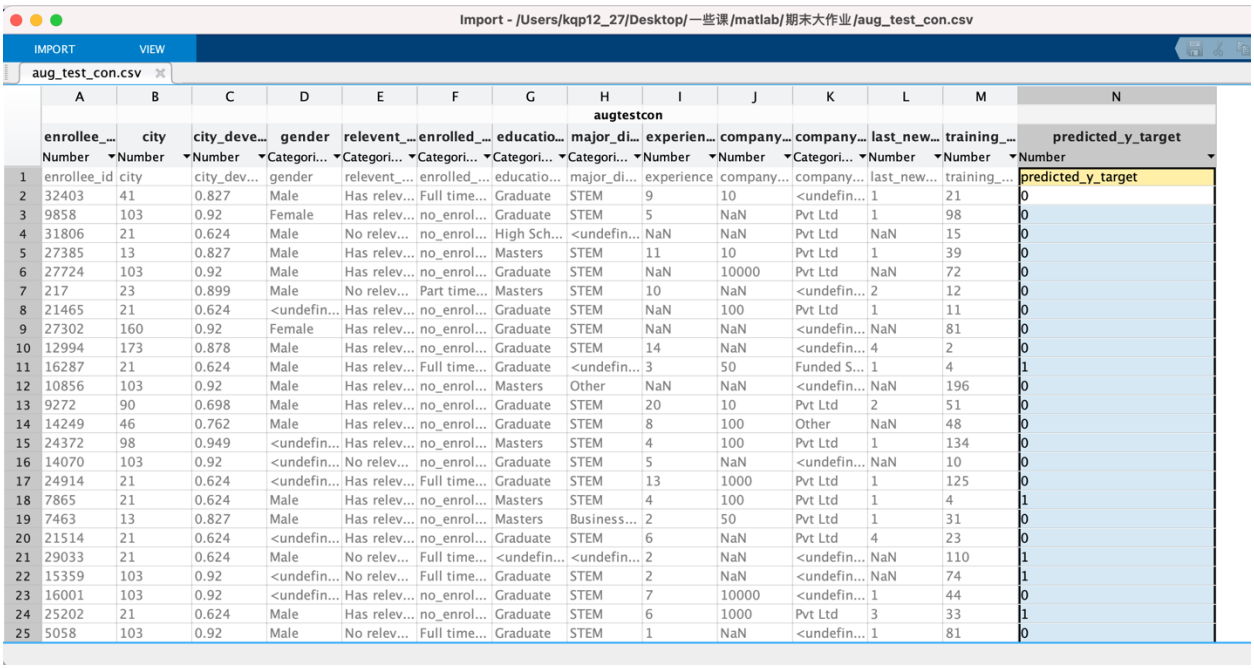
3. 运用训练好的模型对 aug_test 中的数据进行预测

由于前文中训练的四个模型中 Medium KNN 分类效果最好，准确率达到 76.2%，因此此处采用该模型对 test 数据进行预测。

代码：

```
% 调用训练好的 Medium KNN模型
y_target = predict(classificationKNN,augtest_without_missing_dd)

% 写入csv文件
y_target_table =
array2table(y_target,'VariableNames',{'predicted_y_target'});
aug_test_con = [augtest, y_target_table];
writetable(aug_test_con, "aug_test_con.csv")
```

输出：



augtestcon

	A	B	C	D	E	F	G	H	I	J	K	L	M	N
	enrollee_...	city	city_deve...	gender	relevent_...	enrolled_...	educatio...	major_di...	experien...	company...	company...	last_new...	training_...	predicted_y_target
	Number	Number	Number	Categori...	Categori...	Categori...	Categori...	Categori...	Number	Number	Categori...	Number	Number	Number
1	enrollee_id	city	city_dev...	gender	relevent_...	enrolled_...	educatio...	major_di...	experience	company...	company...	last_new...	training_...	predicted_y_target
2	32403	41	0.827	Male	Has relev...	Full time...	Graduate	STEM	9	10	<undefin...	1	21	0
3	9858	103	0.92	Female	Has relev...	no_enrol...	Graduate	STEM	5	NaN	Pvt Ltd	1	98	0
4	31806	21	0.624	Male	No relev...	no_enrol...	High Sch...	<undefin...	NaN	NaN	Pvt Ltd	NaN	15	0
5	27385	13	0.827	Male	Has relev...	no_enrol...	Masters	STEM	11	10	Pvt Ltd	1	39	0
6	27724	103	0.92	Male	Has relev...	no_enrol...	Graduate	STEM	NaN	10000	Pvt Ltd	NaN	72	0
7	217	23	0.899	Male	No relev...	Part time...	Masters	STEM	10	NaN	<undefin...	2	12	0
8	21465	21	0.624	<undefin...	Has relev...	no_enrol...	Graduate	STEM	NaN	100	Pvt Ltd	1	11	0
9	27302	160	0.92	Female	Has relev...	no_enrol...	Graduate	STEM	NaN	NaN	<undefin...	NaN	81	0
10	12994	173	0.878	Male	Has relev...	no_enrol...	Graduate	STEM	14	NaN	<undefin...	4	2	0
11	16287	21	0.624	Male	Has relev...	Full time...	Graduate	<undefin...	3	50	Funded S...	1	4	1
12	10856	103	0.92	Male	Has relev...	no_enrol...	Masters	Other	NaN	NaN	<undefin...	NaN	196	0
13	9272	90	0.698	Male	Has relev...	no_enrol...	Graduate	STEM	20	10	Pvt Ltd	2	51	0
14	14249	46	0.762	Male	Has relev...	no_enrol...	Graduate	STEM	8	100	Other	NaN	48	0
15	24372	98	0.949	<undefin...	Has relev...	no_enrol...	Masters	STEM	4	100	Pvt Ltd	1	134	0
16	14070	103	0.92	<undefin...	No relev...	no_enrol...	Graduate	STEM	5	NaN	<undefin...	NaN	10	0
17	24914	21	0.624	<undefin...	Has relev...	Full time...	Graduate	STEM	13	1000	Pvt Ltd	1	125	0
18	7865	21	0.624	Male	Has relev...	no_enrol...	Masters	STEM	4	100	Pvt Ltd	1	4	1
19	7463	13	0.827	Male	Has relev...	no_enrol...	Masters	Business...	2	50	Pvt Ltd	1	31	0
20	21514	21	0.624	<undefin...	Has relev...	no_enrol...	Graduate	STEM	6	NaN	Pvt Ltd	4	23	0
21	29033	21	0.624	Male	No relev...	Full time...	<undefin...	<undefin...	2	NaN	<undefin...	NaN	110	1
22	15359	103	0.92	<undefin...	No relev...	Full time...	Graduate	STEM	2	NaN	<undefin...	NaN	74	1
23	16001	103	0.92	<undefin...	Has relev...	no_enrol...	Graduate	STEM	7	10000	<undefin...	1	44	0
24	25202	21	0.624	Male	Has relev...	no_enrol...	Graduate	STEM	6	1000	Pvt Ltd	3	33	1
25	5058	103	0.92	Male	No relev...	Full time...	Graduate	STEM	1	NaN	<undefin...	1	81	0

十二、 实验过程总结与结论

1. 实验过程总结

本实验先对 aug_train.csv 的数据进行数据清洗，之后用图表展示的形式进行探索性分析，然后采用了核函数分别为“linear”和“rbf”的两种 SVM 算法、linear KNN 算法、Medium KNN 算法对数据进行训练，得到模型。

之后运用 Medium KNN 算法训练的模型准确率最高，达到 76.2%，进而调用该模型对数据清洗过后的 aug_test.csv 数据进行预测，输出预测的 target 的值（即预测中候选人最终的选择），并输入 aug_test_con.csv 文件。

2. 结论

aug_test_con.csv 文件中 predicted_y_target 为预测结果，公司可以着重对此列元素为 1 的候选人进行关注、选拔、录取。