学习如下的python包的内容，并调试成功代码。

https://github.com/amphibian-dev/toad

用上述链接中的数据，调试如下的代码包，并实现调用。

def check\_empty\_bins(dtm,bins):

# check empty bins

bin\_list = np.unique(dtm[bins].astype(str)).tolist()

if 'nan' in bin\_list:

bin\_list.remove('nan')

binleft = set([re.match(r'\((.+),(.+)\]', i).group(1) for i in bin\_list]).difference(set(['-inf', 'inf']))

binright = set([re.match(r'\((.+),(.+)\]', i).group(2) for i in bin\_list]).difference(set(['-inf', 'inf']))

if binleft != binright:

bstbrks = sorted(list(map(float, ['-inf'] + list(binright) + ['inf'])))

bstbrks.pop(-2)

labels = ['[{},{})'.format(bstbrks[i], bstbrks[i + 1]) for i in range(len(bstbrks) - 1)]

# print("The break points are modified into '[{}]'. There are empty bins based on the provided break points.".format(','.join(binright)))

# binning

# dtm['bin'] = dtm['bin'].astype(str)

# return

return bstbrks

#字符型或者唯一值较少的变量

def psi1(data,psi\_data,var):

a = data[var].value\_counts().reset\_index(drop=False)#.astype(str)

a.rename(columns={'index':'name',var:'开发'+var},inplace=True)

b=psi\_data[var].value\_counts().reset\_index(drop=False)#.astype(str)

b.rename(columns={'index':'name',var:'验证'+var},inplace=True)

m=pd.merge(a,b,on='name',how='inner')

m[var+'开发频率']=m['开发'+var]/sum(m['开发'+var])

m[var+'验证频率']=m['验证'+var]/sum(m['验证'+var])

m['psi']=(m[var+'开发频率']-m[var+'验证频率'])\*np.log(m[var+'开发频率']/m[var+'验证频率'])

psi\_sum=sum(m['psi'])

return psi\_sum

def psi2(data,psi\_data,var,brk):

a = pd.cut(data[var], brk, right=False).value\_counts().reset\_index(drop=False)#.astype(str)

a.rename(columns={'index':'name',var:'开发'+var},inplace=True)

b=pd.cut(psi\_data[var], brk,right=False).value\_counts().reset\_index(drop=False)#.astype(str)

b.rename(columns={'index':'name',var:'验证'+var},inplace=True)

m=pd.merge(a,b,on='name',how='inner')

m[var+'开发频率']=m['开发'+var]/sum(m['开发'+var])

m[var+'验证频率']=m['验证'+var]/sum(m['验证'+var])

m['psi']=(m[var+'开发频率']-m[var+'验证频率'])\*np.log(m[var+'开发频率']/m[var+'验证频率'])

psi\_sum=sum(m['psi'])

return psi\_sum

def psi\_hui(df,psi\_data,target,n=5):

chat=list(df.columns[df.dtypes == 'object'])

name=df.columns.drop(target)

psis=[]

for i in name:

X=df[i]

Y=df[target]

nuniq=X.nunique()

if nuniq<=n:

chat.append(i)

if i in chat:

psi=psi1(df,psi\_data,i)

psis.append(psi)

else:

d1=pd.DataFrame({"X":X,"Y":Y,"bin":pd.qcut(X,n,duplicates='drop')})

# d1['bin']=d1['bin'].astype(str)

brk=check\_empty\_bins(d1, 'bin')

psi=psi2(df,psi\_data,i,brk)

psis.append(psi)

print (i)

d=pd.DataFrame({"name":name,"psi":psis})

return d