Personalised Cancer Diagnosis

Business Problem:

The workflow is as follows

1. A molecular pathologist selects a list of genetic variations of interest that he/she want to analyze
2. The molecular pathologist searches for evidence in the medical literature that somehow are relevant to the genetic variations of interest
3. Finally this molecular pathologist spends a huge amount of time analyzing the evidence related to each of the variations to classify them

   Our goal here is to replace step 3 by a machine learning model.The molecular pathologist will still have to decide which variations are of interest, and also collect the relevant evidence for them. But the last step, which is also the most time consuming, will be fully automated.

Business objectives and Constraints

1. No latency required
2. Probability of occurence is reuired,since the doctor had to interpret the cause of occurence
3. Errors can be very costly.

Data

- Training Variants consists of 4 columns: ID:the id of the row used to link the mutation to the clinical evidence Gene:the gene where this genetic mutation is located Variants:the aminoacid change for this mutations Class:1-9 the class this genetic mutation has been classified on
- Training Text consists of 2 columns:ID,Text
- Depending on the text related to each ID class label will be given.This mapping is done through an inner join between Training Variants and Training Text.
- Goal of machine learning is to detect class label depending on the text,gene,variation.Since class label is between [1 -9] it is multi-class classification.Metric used is multi-class log-loss and confusion matrix.

Load Data

In [1]:

```
import pandas as pd

data = pd.read_csv(r'C:\Users\Friend\AI\AI_datasets\Cancer\training_variants')
print(data.shape)
```

```
(3321, 4)
```

In [2]:

```
data_text = pd.read_csv(r'C:\Users\Friend\AI\AI_datasets\Cancer\training_text',sep = '\|\|',names=["ID"
,"TEXT"],skiprows=1)
print(data_text.shape)
```

```
C:\Users\Friend\Anaconda3\lib\site-packages\ipykernel_launcher.py:1: ParserWarning: Falling back to the
'python' engine because the 'c' engine does not support regex separators (separators > 1 char and diffe
rent from '\s+' are interpreted as regex); you can avoid this warning by specifying engine='python'.
  """Entry point for launching an IPython kernel.
```

```
(3321, 2)
```

Preprocess data

In [6]:

```
from nltk.corpus import stopwords
import re

stop_words = set(stopwords.words('english'))

def nlp_preprocessing(total_text, index, column):
    if type(total_text) is not int:
        string = ""
        total_text = re.sub('[^a-zA-Z0-9\n]', ' ', total_text)
        total_text = re.sub('\s+',' ', total_text)
        total_text = total_text.lower()
        for word in total_text.split():
            if not word in stop_words:
                string += word + " "
        data_text[column][index] = string

for index, row in data_text.iterrows():
    if type(row['TEXT']) is str:
        nlp_preprocessing(row['TEXT'], index, 'TEXT')
```

```
C:\Users\Friend\Anaconda3\lib\site-packages\ipykernel_launcher.py:15: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame

See the caveats in the documentation: http://pandas.pydata.org/pandas-docs/stable/indexing.html#indexing-view-versus-copy
  from ipykernel import kernelapp as app
```

In [7]:

```
result = pd.merge(data, data_text,on='ID', how='left')
result.loc[result['TEXT'].isnull(),'TEXT'] = result['Gene'] +' '+result['Variation']
```

In [14]:

```
result.head(2)
```

Out[14]:

	ID	Gene	Variation	Class	TEXT
0	0	FAM58A	Truncating Mutations	1	cyclin dependent kinases cdks regulate variety...
1	1	CBL	W802*	2	abstract background non small cell lung cancer...

In [ ]:

```
y_true = result['Class'].values
```

Split Data

In [12]:

```
from sklearn.model_selection import train_test_split

X_train, test_df, y_train, y_test = train_test_split(result, y_true, test_size=0.2, random_state=42)
train_df, cv_df, y_train, y_cv = train_test_split(X_train, y_train,test_size=0.2)
print(train_df.shape,cv_df.shape,test_df.shape)
```

```
(2124, 5) (532, 5) (665, 5)
```

In [12]:

```
train_df.to_csv(r'C:\Users\Friend\AI\AI_datasets\Cancer\train_df.csv', index=False)
cv_df.to_csv(r'C:\Users\Friend\AI\AI_datasets\Cancer\cv_df.csv', index=False)
test_df.to_csv(r'C:\Users\Friend\AI\AI_datasets\Cancer\test_df.csv', index=False)
```

In [14]:

```
np.save(r'C:\Users\Friend\AI\AI_datasets\Cancer\y_train.npy', y_train)
np.save(r'C:\Users\Friend\AI\AI_datasets\Cancer\y_cv.npy',y_cv)
np.save(r'C:\Users\Friend\AI\AI_datasets\Cancer\y_test.npy', y_test)
```

In [16]:

```
train_df = pd.read_csv(r'C:\Users\Friend\AI\AI_datasets\Cancer\train_df.csv')
cv_df = pd.read_csv(r'C:\Users\Friend\AI\AI_datasets\Cancer\cv_df.csv')
test_df = pd.read_csv(r'C:\Users\Friend\AI\AI_datasets\Cancer\test_df.csv')
```

In [17]:

```
y_train = np.load(r'C:\Users\Friend\AI\AI_datasets\Cancer\y_train.npy')
y_cv = np.load(r'C:\Users\Friend\AI\AI_datasets\Cancer\y_cv.npy')
y_test = np.load(r'C:\Users\Friend\AI\AI_datasets\Cancer\y_test.npy')
```

In [18]:

```
print(train_df.shape,cv_df.shape,test_df.shape)
```

```
(2124, 5) (532, 5) (665, 5)
```

Featurizations

Response Encoding

In [19]:

```
import numpy as np

def get_gv_fea_dict(alpha, feature, df):
    value_count = train_df[feature].value_counts()
    gv_dict = dict()
    for i, denominator in value_count.items():
        vec = []
        for k in range(1,10):
            cls_cnt = train_df.loc[(train_df['Class']==k) & (train_df[feature]==i)]
            vec.append((cls_cnt.shape[0] + alpha*10)/ (denominator + 90*alpha))
        gv_dict[i]=vec
    return gv_dict


def get_gv_feature(alpha, feature, df):
    gv_dict = get_gv_fea_dict(alpha, feature, df)
    value_count = train_df[feature].value_counts()
    gv_fea = []
    for index, row in df.iterrows():
        if row[feature] in dict(value_count).keys():
            gv_fea.append(gv_dict[row[feature]])
        else:
            gv_fea.append([1/9,1/9,1/9,1/9,1/9,1/9,1/9,1/9,1/9])
    return gv_fea
```

In [20]:

```
alpha = 1

train_gene_feature_responseCoding = np.array(get_gv_feature(alpha, "Gene", train_df))
```

```
test_gene_feature_responseCoding = np.array(get_gv_feature(alpha, "Gene", test_df))
cv_gene_feature_responseCoding = np.array(get_gv_feature(alpha, "Gene", cv_df))
```

In [26]:

```
train_gene_feature_responseCoding.shape
```

Out[26]:

```
(2124, 9)
```

In [22]:

```
alpha = 1

train_variation_feature_responseCoding = np.array(get_gv_feature(alpha, "Variation", train_df))
test_variation_feature_responseCoding = np.array(get_gv_feature(alpha, "Variation", test_df))
cv_variation_feature_responseCoding = np.array(get_gv_feature(alpha, "Variation", cv_df))
```

In [25]:

```
train_variation_feature_responseCoding.shape
```

Out[25]:

```
(2124, 9)
```

In [ ]:

```
from collections import defaultdict
import math

def extract_dictionary_paddle(cls_text):
    dictionary = defaultdict(int)
    for index, row in cls_text.iterrows():
        for word in row['TEXT'].split():
            dictionary[word] +=1
    return dictionary

dict_list = []
for i in range(1,10):
    cls_text = train_df[train_df['Class']==i]
    dict_list.append(extract_dictionary_paddle(cls_text))

total_dict = extract_dictionary_paddle(train_df)


confuse_array = []
train_text_features= text_vectorizer.get_feature_names()
for i in train_text_features:
    ratios = []
    max_val = -1
    for j in range(0,9):
        ratios.append((dict_list[j][i]+10 )/(total_dict[i]+90))
    confuse_array.append(ratios)
confuse_array = np.array(confuse_array)

def get_text_responsecoding(df):
    text_feature_responseCoding = np.zeros((df.shape[0],9))
    for i in range(0,9):
        row_index = 0
        for index, row in df.iterrows():
            sum_prob = 0
            for word in row['TEXT'].split():
                sum_prob += math.log(((dict_list[i].get(word,0)+10 )/(total_dict.get(word,0)+90)))
            text_feature_responseCoding[row_index][i] = math.exp(sum_prob/len(row['TEXT'].split()))
            row_index += 1
    return text_feature_responseCoding
```

In [55]:

```
train_text_feature_responseCoding  = get_text_responsecoding(train_df)
test_text_feature_responseCoding  = get_text_responsecoding(test_df)
cv_text_feature_responseCoding  = get_text_responsecoding(cv_df)
```

In [56]:

```
train_text_feature_responseCoding = (train_text_feature_responseCoding.T/train_text_feature_responseCod
ing.sum(axis=1)).T
test_text_feature_responseCoding = (test_text_feature_responseCoding.T/test_text_feature_responseCoding
.sum(axis=1)).T
cv_text_feature_responseCoding = (cv_text_feature_responseCoding.T/cv_text_feature_responseCoding.sum(a
xis=1)).T
```

In [57]:

```
train_text_feature_responseCoding.shape
```

Out[57]:

```
(2124, 9)
```

In [60]:

```
train_gene_var_responseCoding = np.hstack((train_gene_feature_responseCoding,train_variation_feature_re
sponseCoding))
test_gene_var_responseCoding = np.hstack((test_gene_feature_responseCoding,test_variation_feature_respo
nseCoding))
cv_gene_var_responseCoding = np.hstack((cv_gene_feature_responseCoding,cv_variation_feature_responseCod
ing))

train_x_responseCoding = np.hstack((train_gene_var_responseCoding, train_text_feature_responseCoding))
test_x_responseCoding = np.hstack((test_gene_var_responseCoding, test_text_feature_responseCoding))
cv_x_responseCoding = np.hstack((cv_gene_var_responseCoding, cv_text_feature_responseCoding))
```

In [65]:

```
print(train_x_responseCoding.shape,test_x_responseCoding.shape,cv_x_responseCoding.shape)
```

```
(2124, 27) (665, 27) (532, 27)
```

One-Hot Encoding

td-idf

In [28]:

```
from sklearn.feature_extraction.text import TfidfVectorizer

gene_vectorizer = TfidfVectorizer(ngram_range=(1,1))

train_gene_feature_onehotCoding = gene_vectorizer.fit_transform(train_df['Gene'])
test_gene_feature_onehotCoding = gene_vectorizer.transform(test_df['Gene'])
cv_gene_feature_onehotCoding = gene_vectorizer.transform(cv_df['Gene'])
```

In [29]:

```
train_gene_feature_onehotCoding.shape
```

Out[29]:

```
(2124, 221)
```

In [30]:

```
variation_vectorizer = TfidfVectorizer(ngram_range=(1,1))
```

```
train_variation_feature_onehotCoding = variation_vectorizer.fit_transform(train_df['Variation'])
test_variation_feature_onehotCoding = variation_vectorizer.transform(test_df['Variation'])
cv_variation_feature_onehotCoding = variation_vectorizer.transform(cv_df['Variation'])
```

In [31]:

```
train_variation_feature_onehotCoding.shape
```

Out[31]:

```
(2124, 1974)
```

Tf-idf Uni-gram

In [76]:

```
from sklearn.preprocessing import normalize

text_vectorizer = TfidfVectorizer(ngram_range=(1,1),max_features=1000)

tfidf_train_text_feature_onehotCoding = text_vectorizer.fit_transform(train_df['TEXT'])
tfidf_test_text_feature_onehotCoding = text_vectorizer.transform(test_df['TEXT'])
tfidf_cv_text_feature_onehotCoding = text_vectorizer.transform(cv_df['TEXT'])

tfidf_train_text_feature_onehotCoding = normalize(tfidf_train_text_feature_onehotCoding, axis=0)
tfidf_test_text_feature_onehotCoding = normalize(tfidf_test_text_feature_onehotCoding, axis=0)
tfidf_cv_text_feature_onehotCoding = normalize(tfidf_cv_text_feature_onehotCoding, axis=0)
```

In [77]:

```
tfidf_train_text_feature_onehotCoding.shape
```

Out[77]:

```
(2124, 1000)
```

Tf-idf Bi-gram

In [74]:

```
from sklearn.preprocessing import normalize

text_vectorizer = TfidfVectorizer(ngram_range=(1,2),max_features=1000)

bi_tfidf_train_text_feature_onehotCoding = text_vectorizer.fit_transform(train_df['TEXT'])
bi_tfidf_test_text_feature_onehotCoding = text_vectorizer.transform(test_df['TEXT'])
bi_tfidf_cv_text_feature_onehotCoding = text_vectorizer.transform(cv_df['TEXT'])

bi_tfidf_train_text_feature_onehotCoding = normalize(bi_tfidf_train_text_feature_onehotCoding, axis=0)
bi_tfidf_test_text_feature_onehotCoding = normalize(bi_tfidf_test_text_feature_onehotCoding, axis=0)
bi_tfidf_cv_text_feature_onehotCoding = normalize(bi_tfidf_cv_text_feature_onehotCoding, axis=0)
```

In [ ]:

```
bi_tfidf_train_text_feature_onehotCoding.shape
```

Count-Vectorizer Uni-Gram

In [51]:

```
from sklearn.feature_extraction.text import CountVectorizer

text_vectorizer = CountVectorizer(min_df =3,ngram_range=(1,1))

count_train_text_feature_onehotCoding = text_vectorizer.fit_transform(train_df['TEXT'])
count_test_text_feature_onehotCoding = text_vectorizer.transform(test_df['TEXT'])
count_cv_text_feature_onehotCoding = text_vectorizer.transform(cv_df['TEXT'])
```

```
count_train_text_feature_onehotCoding = normalize(count_train_text_feature_onehotCoding, axis=0)
count_test_text_feature_onehotCoding = normalize(count_test_text_feature_onehotCoding, axis=0)
count_cv_text_feature_onehotCoding = normalize(count_cv_text_feature_onehotCoding, axis=0)
```

In [52]:

```
count_train_text_feature_onehotCoding.shape
```

Out[52]:

```
(2124, 56607)
```

Count-Vectorizer Bi-Gram

In [53]:

```
from sklearn.feature_extraction.text import CountVectorizer

text_vectorizer = CountVectorizer(min_df =3,ngram_range=(1,2))

count_bi_train_text_feature_onehotCoding = text_vectorizer.fit_transform(train_df['TEXT'])
count_bi_test_text_feature_onehotCoding = text_vectorizer.transform(test_df['TEXT'])
count_bi_cv_text_feature_onehotCoding = text_vectorizer.transform(cv_df['TEXT'])

count_bi_train_text_feature_onehotCoding = normalize(count_bi_train_text_feature_onehotCoding, axis=0)
count_bi_test_text_feature_onehotCoding = normalize(count_bi_test_text_feature_onehotCoding, axis=0)
count_bi_cv_text_feature_onehotCoding = normalize(count_bi_cv_text_feature_onehotCoding, axis=0)
```

In [54]:

```
count_bi_train_text_feature_onehotCoding.shape
```

Out[54]:

```
(2124, 682245)
```

In [44]:

```
np.save(r'C:\Users\Friend\AI\AI_datasets\Cancer\count_train_text_feature_onehotCoding.npy',count_train_
text_feature_onehotCoding)
np.save(r'C:\Users\Friend\AI\AI_datasets\Cancer\count_test_text_feature_onehotCoding.npy',count_test_te
xt_feature_onehotCoding)
np.save(r'C:\Users\Friend\AI\AI_datasets\Cancer\count_cv_text_feature_onehotCoding.npy',count_cv_text_f
eature_onehotCoding)
np.save(r'C:\Users\Friend\AI\AI_datasets\Cancer\count_bi_train_text_feature_onehotCoding.npy',count_bi_
train_text_feature_onehotCoding)
np.save(r'C:\Users\Friend\AI\AI_datasets\Cancer\count_bi_test_text_feature_onehotCoding.npy',count_bi_t
est_text_feature_onehotCoding)
np.save(r'C:\Users\Friend\AI\AI_datasets\Cancer\count_bi_cv_text_feature_onehotCoding.npy',count_bi_cv_
text_feature_onehotCoding)
```

In [64]:

```
from scipy.sparse import hstack

train_gene_var_onehotCoding = hstack((train_gene_feature_onehotCoding,train_variation_feature_onehotCod
ing))
test_gene_var_onehotCoding = hstack((test_gene_feature_onehotCoding,test_variation_feature_onehotCoding
))
cv_gene_var_onehotCoding = hstack((cv_gene_feature_onehotCoding,cv_variation_feature_onehotCoding))

train_x_onehotCoding = hstack((train_gene_var_onehotCoding, train_text_feature_onehotCoding)).tocsr()
train_y = np.array(list(train_df['Class']))

test_x_onehotCoding = hstack((test_gene_var_onehotCoding, test_text_feature_onehotCoding)).tocsr()
test_y = np.array(list(test_df['Class']))

cv_x_onehotCoding = hstack((cv_gene_var_onehotCoding, cv_text_feature_onehotCoding)).tocsr()
cv_y = np.array(list(cv_df['Class']))
```

In [66]:

```
print(train_x_onehotCoding.shape,test_x_onehotCoding.shape,cv_x_onehotCoding.shape)
```

```
(2124, 3191) (665, 3191) (532, 3191)
```

Machine Learning Models

In [60]:

```
import seaborn as sns
from sklearn.metrics import confusion_matrix

def plot_confusion_matrix(test_y, predict_y):
    C = confusion_matrix(test_y, predict_y)
    A =(((C.T)/(C.sum(axis=1))).T)
    B =(C/C.sum(axis=0))
    labels = [1,2,3,4,5,6,7,8,9]

    print("-"*20, "Confusion matrix", "-"*20)
    plt.figure(figsize=(20,7))
    sns.heatmap(C, annot=True, cmap="YlGnBu", fmt=".3f", xticklabels=labels, yticklabels=labels)
    plt.xlabel('Predicted Class')
    plt.ylabel('Original Class')
    plt.show()

    print("-"*20, "Precision matrix (Columm Sum=1)", "-"*20)
    plt.figure(figsize=(20,7))
    sns.heatmap(B, annot=True, cmap="YlGnBu", fmt=".3f", xticklabels=labels, yticklabels=labels)
    plt.xlabel('Predicted Class')
    plt.ylabel('Original Class')
    plt.show()

    print("-"*20, "Recall matrix (Row sum=1)", "-"*20)
    plt.figure(figsize=(20,7))
    sns.heatmap(A, annot=True, cmap="YlGnBu", fmt=".3f", xticklabels=labels, yticklabels=labels)
    plt.xlabel('Predicted Class')
    plt.ylabel('Original Class')
    plt.show()
```

In [61]:

```
def predict_and_plot_confusion_matrix(train_x, train_y,test_x, test_y, clf):
    clf.fit(train_x, train_y)
    sig_clf = CalibratedClassifierCV(clf, method="sigmoid")
    sig_clf.fit(train_x, train_y)
    pred_y = sig_clf.predict(test_x)
    print("Log loss :",log_loss(test_y, sig_clf.predict_proba(test_x)))
    print("Number of mis-classified points :", np.count_nonzero((pred_y- test_y))/test_y.shape[0])
    plot_confusion_matrix(test_y, pred_y)
```

K-Nearest Neighbour

In [32]:

```
from sklearn.neighbors import KNeighborsClassifier
from sklearn.calibration import CalibratedClassifierCV
from sklearn.metrics.classification import accuracy_score, log_loss
```

In [67]:

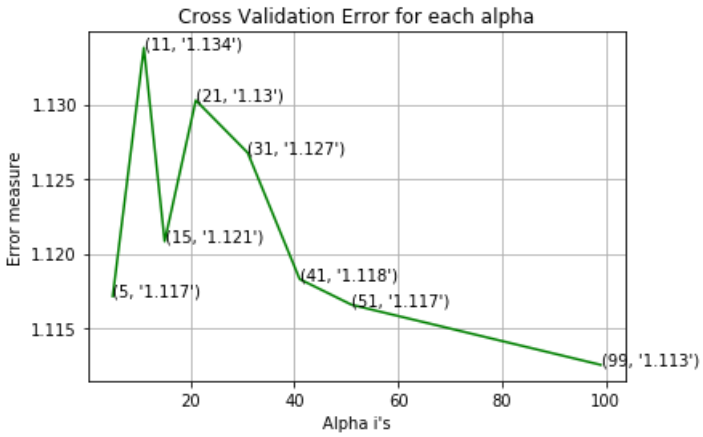
```
alpha = [5, 11, 15, 21, 31, 41, 51, 99]
cv_log_error_array = []
for i in alpha:
    print("for alpha =", i)
    clf = KNeighborsClassifier(n_neighbors=i)
    clf.fit(train_x_responseCoding, train_y)
```

```
    sig_clf = CalibratedClassifierCV(clf, method="sigmoid")
    sig_clf.fit(train_x_responseCoding, train_y)
    sig_clf_probs = sig_clf.predict_proba(cv_x_responseCoding)
    cv_log_error_array.append(log_loss(cv_y, sig_clf_probs, labels=clf.classes_, eps=1e-15))
    print("Log Loss :",log_loss(cv_y, sig_clf_probs))
```

```
for alpha = 5
Log Loss : 1.1171556430124965
for alpha = 11
Log Loss : 1.1337873802476925
for alpha = 15
Log Loss : 1.1208349360744225
for alpha = 21
Log Loss : 1.1302718880265141
for alpha = 31
Log Loss : 1.126756649994753
for alpha = 41
Log Loss : 1.1183014532791826
for alpha = 51
Log Loss : 1.1165855683945538
for alpha = 99
Log Loss : 1.1125636787444793
```

In [71]:

```
from matplotlib import pyplot as plt
import seaborn as sns

fig, ax = plt.subplots()
ax.plot(alpha, cv_log_error_array,c='g')
for i, txt in enumerate(np.round(cv_log_error_array,3)):
    ax.annotate((alpha[i],str(txt)), (alpha[i],cv_log_error_array[i]))
plt.grid()
plt.title("Cross Validation Error for each alpha")
plt.xlabel("Alpha i's")
plt.ylabel("Error measure")
plt.show()
```



In [80]:

```
best_alpha = np.argmin(cv_log_error_array)

clf = KNeighborsClassifier(n_neighbors=alpha[best_alpha])
clf.fit(train_x_responseCoding, train_y)
sig_clf = CalibratedClassifierCV(clf, method="sigmoid")
sig_clf.fit(train_x_responseCoding, train_y)

predict_y = sig_clf.predict_proba(train_x_responseCoding)
k_train_log_loss = log_loss(y_train, predict_y, labels=clf.classes_, eps=1e-15)
print('For values of best alpha = ', alpha[best_alpha], "The train log loss is:",k_train_log_loss)
predict_y = sig_clf.predict_proba(cv_x_responseCoding)
k_cv_log_loss = log_loss(y_cv, predict_y, labels=clf.classes_, eps=1e-15)
print('For values of best alpha = ', alpha[best_alpha], "The cross validation log loss is:",k_cv_log_lo
ss)
predict_y = sig_clf.predict_proba(test_x_responseCoding)
k_test_log_loss = log_loss(y_test, predict_y, labels=clf.classes_, eps=1e-15)
print('For values of best alpha = ', alpha[best_alpha], "The test log loss is:",k_test_log_loss)
```

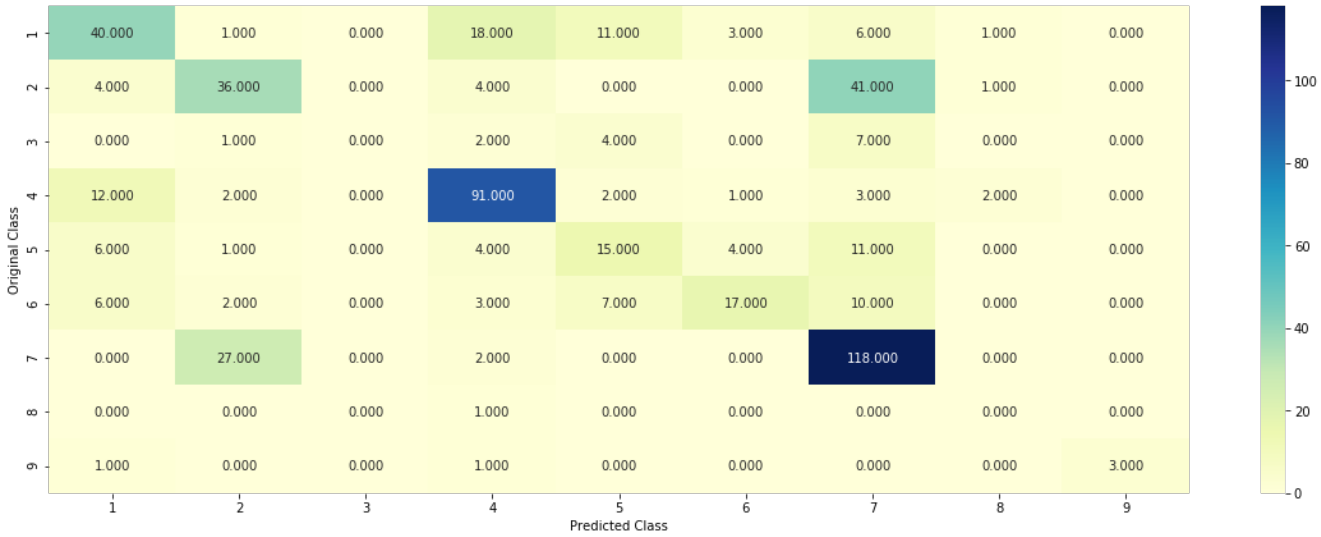
```
For values of best alpha =  99 The train log loss is: 0.9582284186052225
For values of best alpha =  99 The cross validation log loss is: 1.1125636787444793
For values of best alpha =  99 The test log loss is: 1.08911155067031
```

In [79]:

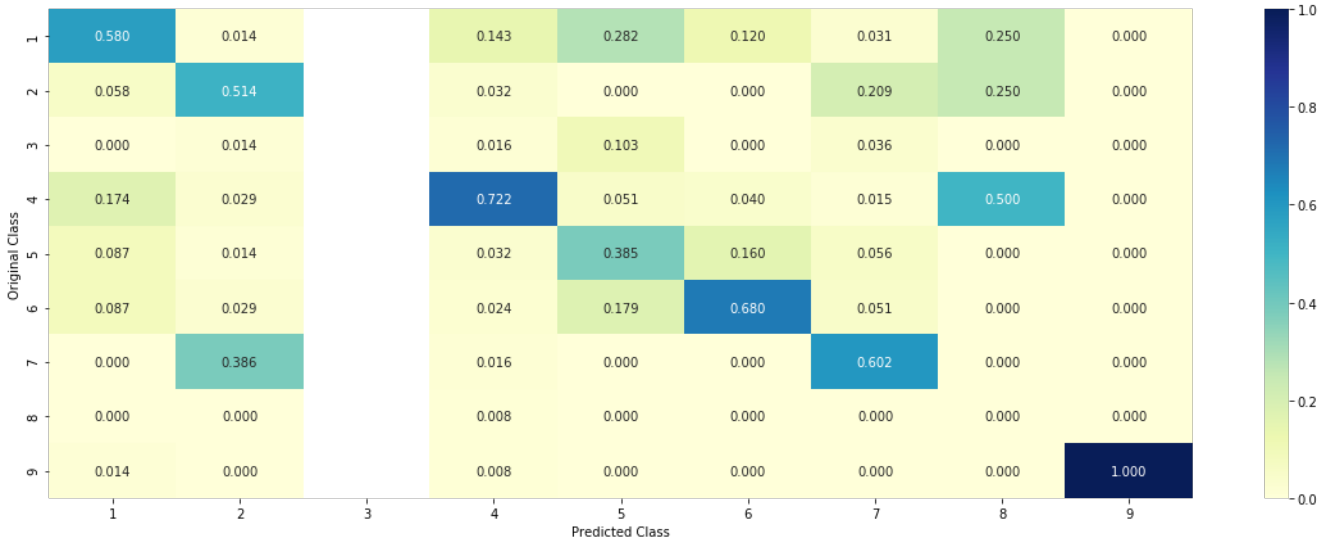
```
clf = KNeighborsClassifier(n_neighbors=alpha[best_alpha])
predict_and_plot_confusion_matrix(train_x_responseCoding, train_y, cv_x_responseCoding, cv_y, clf)
```

```
Log loss : 1.1125636787444793
Number of mis-classified points : 0.39849624060150374
-------------------- Confusion matrix --------------------
```

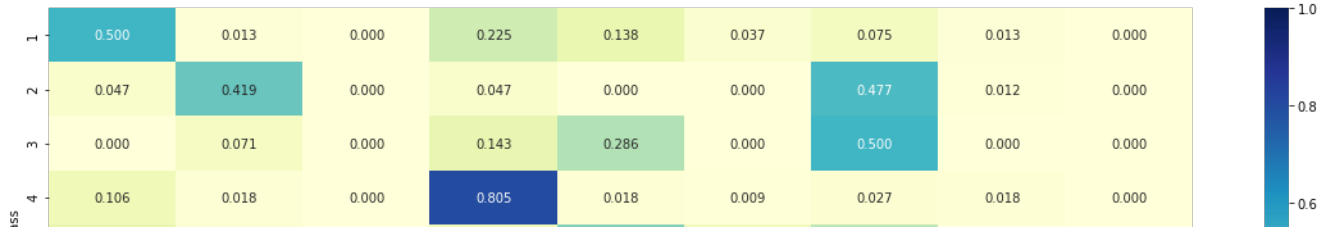
```
C:\Users\Friend\Anaconda3\lib\site-packages\ipykernel_launcher.py:7: RuntimeWarning: invalid value enco
untered in true_divide
  import sys
```

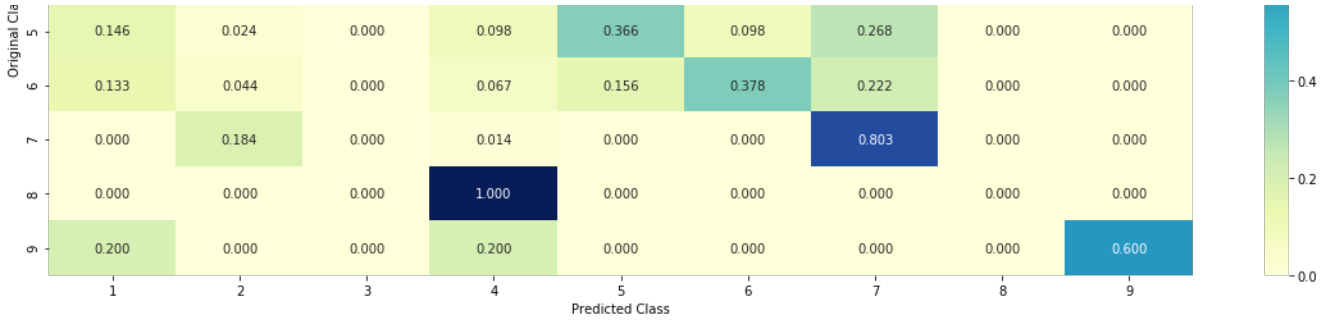



```
-------------------- Precision matrix (Columm Sum=1) --------------------
```




```
-------------------- Recall matrix (Row sum=1) --------------------
```





Naive Bayes

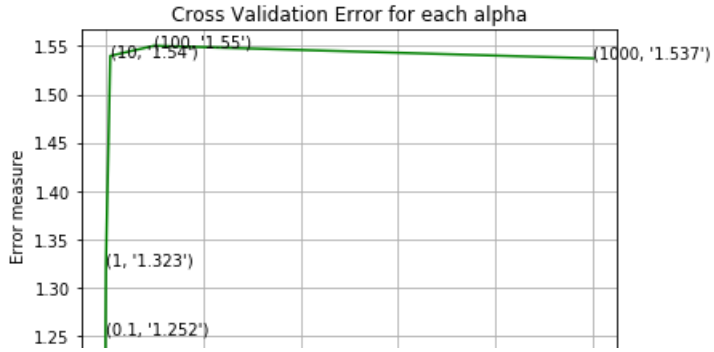
In [83]:

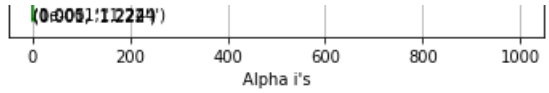
```
from sklearn.naive_bayes import MultinomialNB

alpha = [0.00001, 0.0001, 0.001, 0.1, 1, 10, 100,1000]
cv_log_error_array = []
for i in alpha:
    print("for alpha =", i)
    clf = MultinomialNB(alpha=i)
    clf.fit(train_x_onehotCoding, train_y)
    sig_clf = CalibratedClassifierCV(clf, method="sigmoid")
    sig_clf.fit(train_x_onehotCoding, train_y)
    sig_clf_probs = sig_clf.predict_proba(cv_x_onehotCoding)
    cv_log_error_array.append(log_loss(cv_y, sig_clf_probs, labels=clf.classes_, eps=1e-15))
    print("Log Loss :",log_loss(cv_y, sig_clf_probs))
```

```
for alpha = 1e-05
Log Loss : 1.223873779543032
for alpha = 0.0001
Log Loss : 1.223856243413156
for alpha = 0.001
Log Loss : 1.2219297058574758
for alpha = 0.1
Log Loss : 1.2518988804793472
for alpha = 1
Log Loss : 1.3231352458576149
for alpha = 10
Log Loss : 1.5398828958736168
for alpha = 100
Log Loss : 1.5503765228631956
for alpha = 1000
Log Loss : 1.5372633043412547
```

In [84]:

```
fig, ax = plt.subplots()
ax.plot(alpha, cv_log_error_array,c='g')
for i, txt in enumerate(np.round(cv_log_error_array,3)):
    ax.annotate((alpha[i],str(txt)), (alpha[i],cv_log_error_array[i]))
plt.grid()
plt.title("Cross Validation Error for each alpha")
plt.xlabel("Alpha i's")
plt.ylabel("Error measure")
plt.show()
```





In [85]:

```
best_alpha = np.argmin(cv_log_error_array)

clf = MultinomialNB(alpha=alpha[best_alpha])
clf.fit(train_x_onehotCoding, train_y)
sig_clf = CalibratedClassifierCV(clf, method="sigmoid")
sig_clf.fit(train_x_onehotCoding, train_y)

predict_y = sig_clf.predict_proba(train_x_onehotCoding)
naive_train_log_loss = log_loss(y_train, predict_y, labels=clf.classes_, eps=1e-15)
print('For values of best alpha = ', alpha[best_alpha], "The train log loss is:",naive_train_log_loss)
predict_y = sig_clf.predict_proba(cv_x_onehotCoding)
naive_cv_log_loss = log_loss(y_cv, predict_y, labels=clf.classes_, eps=1e-15)
print('For values of best alpha = ', alpha[best_alpha], "The cross validation log loss is:",naive_cv_lo
g_loss)
predict_y = sig_clf.predict_proba(test_x_onehotCoding)
naive_test_log_loss = log_loss(y_test, predict_y, labels=clf.classes_, eps=1e-15)
print('For values of best alpha = ', alpha[best_alpha], "The test log loss is:",naive_test_log_loss)
```

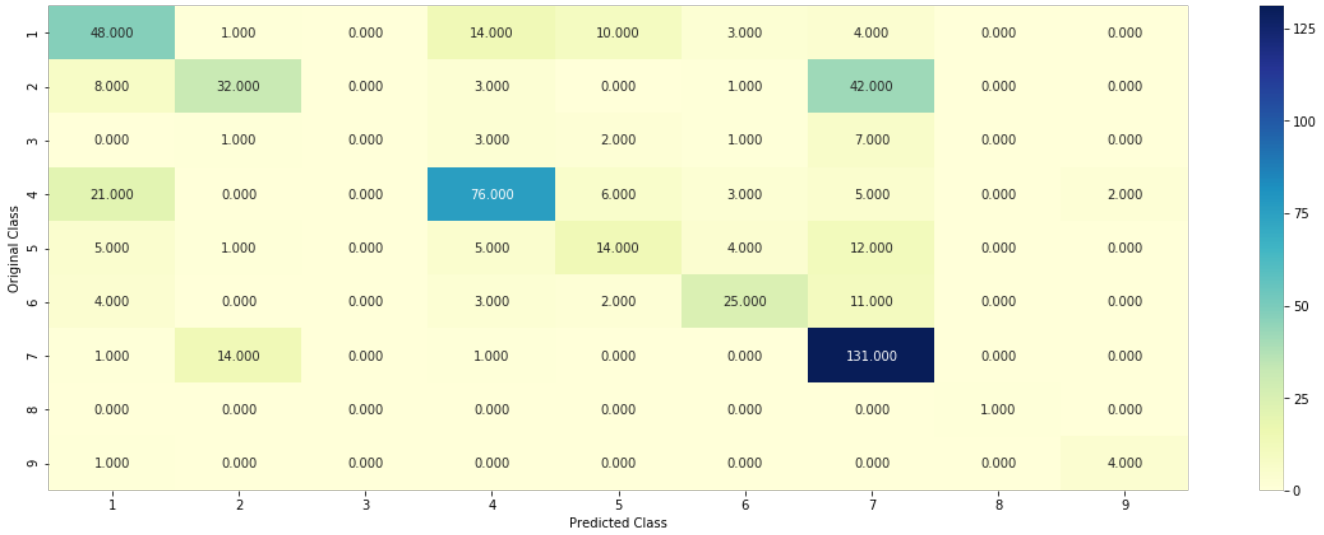
```
For values of best alpha =  0.001 The train log loss is: 0.5176799623398368
For values of best alpha =  0.001 The cross validation log loss is: 1.2219297058574758
For values of best alpha =  0.001 The test log loss is: 1.1679082869734096
```

In [86]:

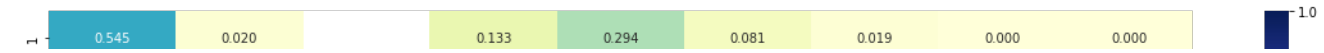
```
clf = MultinomialNB(alpha=alpha[best_alpha])
clf.fit(train_x_onehotCoding, train_y)
sig_clf = CalibratedClassifierCV(clf, method="sigmoid")
sig_clf.fit(train_x_onehotCoding, train_y)
sig_clf_probs = sig_clf.predict_proba(cv_x_onehotCoding)
print("Log Loss :",log_loss(cv_y, sig_clf_probs))
print("Number of missclassified point :", np.count_nonzero((sig_clf.predict(cv_x_onehotCoding)- cv_y))/
cv_y.shape[0])
plot_confusion_matrix(cv_y, sig_clf.predict(cv_x_onehotCoding.toarray()))
```

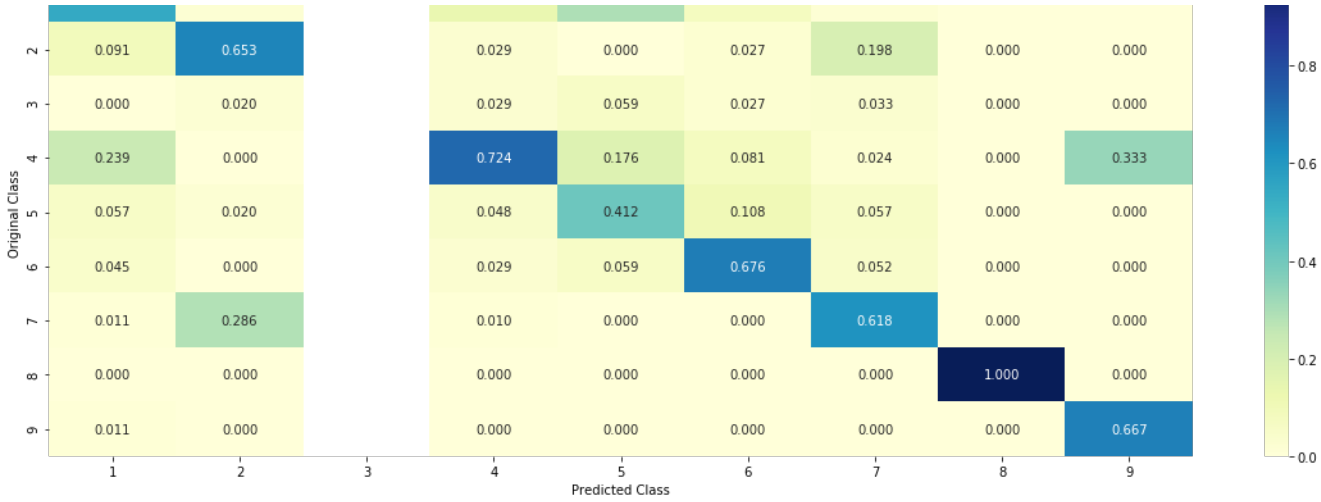
```
Log Loss : 1.2219297058574758
Number of missclassified point : 0.37781954887218044
-------------------- Confusion matrix --------------------
```

```
C:\Users\Friend\Anaconda3\lib\site-packages\ipykernel_launcher.py:7: RuntimeWarning: invalid value enco
untered in true_divide
  import sys
```

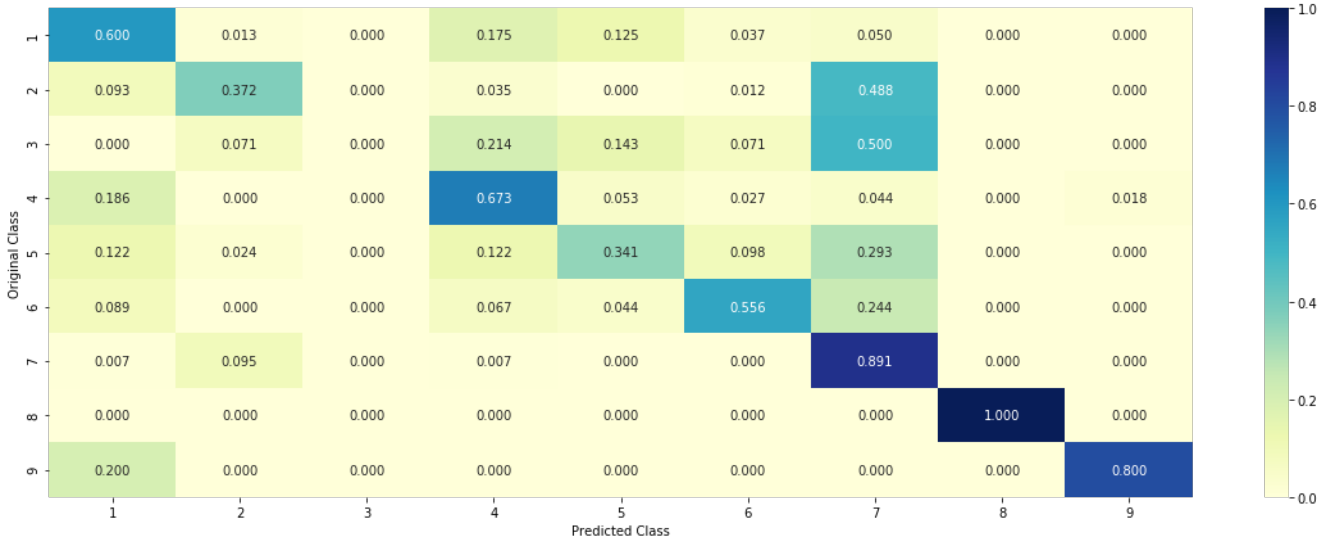



```
-------------------- Precision matrix (Column Sum=1) --------------------
```





---------- Recall matrix (Row sum=1) ----------



Logistic Regression(tfidf)

In [87]:

```
from sklearn.linear_model import SGDClassifier

alpha = [10 ** x for x in range(-6, 3)]
cv_log_error_array = []
for i in alpha:
    print("for alpha =", i)
    clf = SGDClassifier(class_weight='balanced', alpha=i, penalty='l2', loss='log', random_state=42)
    clf.fit(train_x_onehotCoding, train_y)
    sig_clf = CalibratedClassifierCV(clf, method="sigmoid")
    sig_clf.fit(train_x_onehotCoding, train_y)
    sig_clf_probs = sig_clf.predict_proba(cv_x_onehotCoding)
    cv_log_error_array.append(log_loss(cv_y, sig_clf_probs, labels=clf.classes_, eps=1e-15))
    print("Log Loss :",log_loss(cv_y, sig_clf_probs))
```

for alpha = 1e-06

```
C:\Users\Friend\Anaconda3\lib\site-packages\sklearn\linear_model\stochastic_gradient.py:128: FutureWarn
ing: max_iter and tol parameters have been added in <class 'sklearn.linear_model.stochastic_gradient.SG
DClassifier'> in 0.19. If both are left unset, they default to max_iter=5 and tol=None. If tol is not N
one, max_iter defaults to max_iter=1000. From 0.21, default max_iter will be 1000, and default tol will
be 1e-3.
  "and default tol will be 1e-3." % type(self), FutureWarning)
C:\Users\Friend\Anaconda3\lib\site-packages\sklearn\linear_model\stochastic_gradient.py:128: FutureWarn
ing: max_iter and tol parameters have been added in <class 'sklearn.linear_model.stochastic_gradient.SG
DClassifier'> in 0.19. If both are left unset, they default to max_iter=5 and tol=None. If tol is not N
one, max_iter defaults to max_iter=1000. From 0.21, default max_iter will be 1000, and default tol will
be 1e-3.
```

```
  "and default tol will be 1e-3." % type(self), FutureWarning)
C:\Users\Friend\Anaconda3\lib\site-packages\sklearn\linear_model\stochastic_gradient.py:128: FutureWarn
ing: max_iter and tol parameters have been added in <class 'sklearn.linear_model.stochastic_gradient.SG
DClassifier'> in 0.19. If both are left unset, they default to max_iter=5 and tol=None. If tol is not N
one, max_iter defaults to max_iter=1000. From 0.21, default max_iter will be 1000, and default tol will
be 1e-3.
  "and default tol will be 1e-3." % type(self), FutureWarning)
C:\Users\Friend\Anaconda3\lib\site-packages\sklearn\linear_model\stochastic_gradient.py:128: FutureWarn
ing: max_iter and tol parameters have been added in <class 'sklearn.linear_model.stochastic_gradient.SG
DClassifier'> in 0.19. If both are left unset, they default to max_iter=5 and tol=None. If tol is not N
one, max_iter defaults to max_iter=1000. From 0.21, default max_iter will be 1000, and default tol will
be 1e-3.
  "and default tol will be 1e-3." % type(self), FutureWarning)
```

```
Log Loss : 1.1502935401107754
for alpha = 1e-05
```

```
C:\Users\Friend\Anaconda3\lib\site-packages\sklearn\linear_model\stochastic_gradient.py:128: FutureWarn
ing: max_iter and tol parameters have been added in <class 'sklearn.linear_model.stochastic_gradient.SG
DClassifier'> in 0.19. If both are left unset, they default to max_iter=5 and tol=None. If tol is not N
one, max_iter defaults to max_iter=1000. From 0.21, default max_iter will be 1000, and default tol will
be 1e-3.
  "and default tol will be 1e-3." % type(self), FutureWarning)
C:\Users\Friend\Anaconda3\lib\site-packages\sklearn\linear_model\stochastic_gradient.py:128: FutureWarn
ing: max_iter and tol parameters have been added in <class 'sklearn.linear_model.stochastic_gradient.SG
DClassifier'> in 0.19. If both are left unset, they default to max_iter=5 and tol=None. If tol is not N
one, max_iter defaults to max_iter=1000. From 0.21, default max_iter will be 1000, and default tol will
be 1e-3.
  "and default tol will be 1e-3." % type(self), FutureWarning)
C:\Users\Friend\Anaconda3\lib\site-packages\sklearn\linear_model\stochastic_gradient.py:128: FutureWarn
ing: max_iter and tol parameters have been added in <class 'sklearn.linear_model.stochastic_gradient.SG
DClassifier'> in 0.19. If both are left unset, they default to max_iter=5 and tol=None. If tol is not N
one, max_iter defaults to max_iter=1000. From 0.21, default max_iter will be 1000, and default tol will
be 1e-3.
  "and default tol will be 1e-3." % type(self), FutureWarning)
```

```
Log Loss : 1.1112593225877703
for alpha = 0.0001
```

```
C:\Users\Friend\Anaconda3\lib\site-packages\sklearn\linear_model\stochastic_gradient.py:128: FutureWarn
ing: max_iter and tol parameters have been added in <class 'sklearn.linear_model.stochastic_gradient.SG
DClassifier'> in 0.19. If both are left unset, they default to max_iter=5 and tol=None. If tol is not N
one, max_iter defaults to max_iter=1000. From 0.21, default max_iter will be 1000, and default tol will
be 1e-3.
  "and default tol will be 1e-3." % type(self), FutureWarning)
C:\Users\Friend\Anaconda3\lib\site-packages\sklearn\linear_model\stochastic_gradient.py:128: FutureWarn
ing: max_iter and tol parameters have been added in <class 'sklearn.linear_model.stochastic_gradient.SG
DClassifier'> in 0.19. If both are left unset, they default to max_iter=5 and tol=None. If tol is not N
one, max_iter defaults to max_iter=1000. From 0.21, default max_iter will be 1000, and default tol will
be 1e-3.
  "and default tol will be 1e-3." % type(self), FutureWarning)
C:\Users\Friend\Anaconda3\lib\site-packages\sklearn\linear_model\stochastic_gradient.py:128: FutureWarn
ing: max_iter and tol parameters have been added in <class 'sklearn.linear_model.stochastic_gradient.SG
DClassifier'> in 0.19. If both are left unset, they default to max_iter=5 and tol=None. If tol is not N
one, max_iter defaults to max_iter=1000. From 0.21, default max_iter will be 1000, and default tol will
be 1e-3.
  "and default tol will be 1e-3." % type(self), FutureWarning)
```

```
Log Loss : 1.072988566235074
for alpha = 0.001
```

```
C:\Users\Friend\Anaconda3\lib\site-packages\sklearn\linear_model\stochastic_gradient.py:128: FutureWarn
ing: max_iter and tol parameters have been added in <class 'sklearn.linear_model.stochastic_gradient.SG
DClassifier'> in 0.19. If both are left unset, they default to max_iter=5 and tol=None. If tol is not N
one, max_iter defaults to max_iter=1000. From 0.21, default max_iter will be 1000, and default tol will
be 1e-3.
  "and default tol will be 1e-3." % type(self), FutureWarning)
C:\Users\Friend\Anaconda3\lib\site-packages\sklearn\linear_model\stochastic_gradient.py:128: FutureWarn
ing: max_iter and tol parameters have been added in <class 'sklearn.linear_model.stochastic_gradient.SG
DClassifier'> in 0.19. If both are left unset, they default to max_iter=5 and tol=None. If tol is not N
one, max_iter defaults to max_iter=1000. From 0.21, default max_iter will be 1000, and default tol will
be 1e-3.
  "and default tol will be 1e-3." % type(self), FutureWarning)
C:\Users\Friend\Anaconda3\lib\site-packages\sklearn\linear_model\stochastic_gradient.py:128: FutureWarn
```

```
C:\Users\Friend\Anaconda3\lib\site-packages\sklearn\linear_model\stochastic_gradient.py:128: FutureWarn
ing: max_iter and tol parameters have been added in <class 'sklearn.linear_model.stochastic_gradient.SG
DClassifier'> in 0.19. If both are left unset, they default to max_iter=5 and tol=None. If tol is not N
one, max_iter defaults to max_iter=1000. From 0.21, default max_iter will be 1000, and default tol will
be 1e-3.
  "and default tol will be 1e-3." % type(self), FutureWarning)
```

```
Log Loss : 1.0723810107015828
for alpha = 0.01
```

```
C:\Users\Friend\Anaconda3\lib\site-packages\sklearn\linear_model\stochastic_gradient.py:128: FutureWarn
ing: max_iter and tol parameters have been added in <class 'sklearn.linear_model.stochastic_gradient.SG
DClassifier'> in 0.19. If both are left unset, they default to max_iter=5 and tol=None. If tol is not N
one, max_iter defaults to max_iter=1000. From 0.21, default max_iter will be 1000, and default tol will
be 1e-3.
  "and default tol will be 1e-3." % type(self), FutureWarning)
C:\Users\Friend\Anaconda3\lib\site-packages\sklearn\linear_model\stochastic_gradient.py:128: FutureWarn
ing: max_iter and tol parameters have been added in <class 'sklearn.linear_model.stochastic_gradient.SG
DClassifier'> in 0.19. If both are left unset, they default to max_iter=5 and tol=None. If tol is not N
one, max_iter defaults to max_iter=1000. From 0.21, default max_iter will be 1000, and default tol will
be 1e-3.
  "and default tol will be 1e-3." % type(self), FutureWarning)
C:\Users\Friend\Anaconda3\lib\site-packages\sklearn\linear_model\stochastic_gradient.py:128: FutureWarn
ing: max_iter and tol parameters have been added in <class 'sklearn.linear_model.stochastic_gradient.SG
DClassifier'> in 0.19. If both are left unset, they default to max_iter=5 and tol=None. If tol is not N
one, max_iter defaults to max_iter=1000. From 0.21, default max_iter will be 1000, and default tol will
be 1e-3.
  "and default tol will be 1e-3." % type(self), FutureWarning)
```

```
Log Loss : 1.228330990818837
for alpha = 0.1
```

```
C:\Users\Friend\Anaconda3\lib\site-packages\sklearn\linear_model\stochastic_gradient.py:128: FutureWarn
ing: max_iter and tol parameters have been added in <class 'sklearn.linear_model.stochastic_gradient.SG
DClassifier'> in 0.19. If both are left unset, they default to max_iter=5 and tol=None. If tol is not N
one, max_iter defaults to max_iter=1000. From 0.21, default max_iter will be 1000, and default tol will
be 1e-3.
  "and default tol will be 1e-3." % type(self), FutureWarning)
C:\Users\Friend\Anaconda3\lib\site-packages\sklearn\linear_model\stochastic_gradient.py:128: FutureWarn
ing: max_iter and tol parameters have been added in <class 'sklearn.linear_model.stochastic_gradient.SG
DClassifier'> in 0.19. If both are left unset, they default to max_iter=5 and tol=None. If tol is not N
one, max_iter defaults to max_iter=1000. From 0.21, default max_iter will be 1000, and default tol will
be 1e-3.
  "and default tol will be 1e-3." % type(self), FutureWarning)
C:\Users\Friend\Anaconda3\lib\site-packages\sklearn\linear_model\stochastic_gradient.py:128: FutureWarn
ing: max_iter and tol parameters have been added in <class 'sklearn.linear_model.stochastic_gradient.SG
DClassifier'> in 0.19. If both are left unset, they default to max_iter=5 and tol=None. If tol is not N
one, max_iter defaults to max_iter=1000. From 0.21, default max_iter will be 1000, and default tol will
be 1e-3.
  "and default tol will be 1e-3." % type(self), FutureWarning)
```

```
Log Loss : 1.671242183622479
for alpha = 1
```

```
C:\Users\Friend\Anaconda3\lib\site-packages\sklearn\linear_model\stochastic_gradient.py:128: FutureWarn
ing: max_iter and tol parameters have been added in <class 'sklearn.linear_model.stochastic_gradient.SG
DClassifier'> in 0.19. If both are left unset, they default to max_iter=5 and tol=None. If tol is not N
one, max_iter defaults to max_iter=1000. From 0.21, default max_iter will be 1000, and default tol will
be 1e-3.
  "and default tol will be 1e-3." % type(self), FutureWarning)
C:\Users\Friend\Anaconda3\lib\site-packages\sklearn\linear_model\stochastic_gradient.py:128: FutureWarn
ing: max_iter and tol parameters have been added in <class 'sklearn.linear_model.stochastic_gradient.SG
DClassifier'> in 0.19. If both are left unset, they default to max_iter=5 and tol=None. If tol is not N
one, max_iter defaults to max_iter=1000. From 0.21, default max_iter will be 1000, and default tol will
be 1e-3.
  "and default tol will be 1e-3." % type(self), FutureWarning)
C:\Users\Friend\Anaconda3\lib\site-packages\sklearn\linear_model\stochastic_gradient.py:128: FutureWarn
ing: max_iter and tol parameters have been added in <class 'sklearn.linear_model.stochastic_gradient.SG
DClassifier'> in 0.19. If both are left unset, they default to max_iter=5 and tol=None. If tol is not N
one, max_iter defaults to max_iter=1000. From 0.21, default max_iter will be 1000, and default tol will
be 1e-3.
  "and default tol will be 1e-3." % type(self), FutureWarning)
```

```
Log Loss : 1.8241937700264823
for alpha = 10
```

```
C:\Users\Friend\Anaconda3\lib\site-packages\sklearn\linear_model\stochastic_gradient.py:128: FutureWarning: max_iter and tol parameters have been added in <class 'sklearn.linear_model.stochastic_gradient.SGDClassifier'> in 0.19. If both are left unset, they default to max_iter=5 and tol=None. If tol is not None, max_iter defaults to max_iter=1000. From 0.21, default max_iter will be 1000, and default tol will be 1e-3.
  "and default tol will be 1e-3." % type(self), FutureWarning)
C:\Users\Friend\Anaconda3\lib\site-packages\sklearn\linear_model\stochastic_gradient.py:128: FutureWarning: max_iter and tol parameters have been added in <class 'sklearn.linear_model.stochastic_gradient.SGDClassifier'> in 0.19. If both are left unset, they default to max_iter=5 and tol=None. If tol is not None, max_iter defaults to max_iter=1000. From 0.21, default max_iter will be 1000, and default tol will be 1e-3.
  "and default tol will be 1e-3." % type(self), FutureWarning)
C:\Users\Friend\Anaconda3\lib\site-packages\sklearn\linear_model\stochastic_gradient.py:128: FutureWarning: max_iter and tol parameters have been added in <class 'sklearn.linear_model.stochastic_gradient.SGDClassifier'> in 0.19. If both are left unset, they default to max_iter=5 and tol=None. If tol is not None, max_iter defaults to max_iter=1000. From 0.21, default max_iter will be 1000, and default tol will be 1e-3.
  "and default tol will be 1e-3." % type(self), FutureWarning)
```

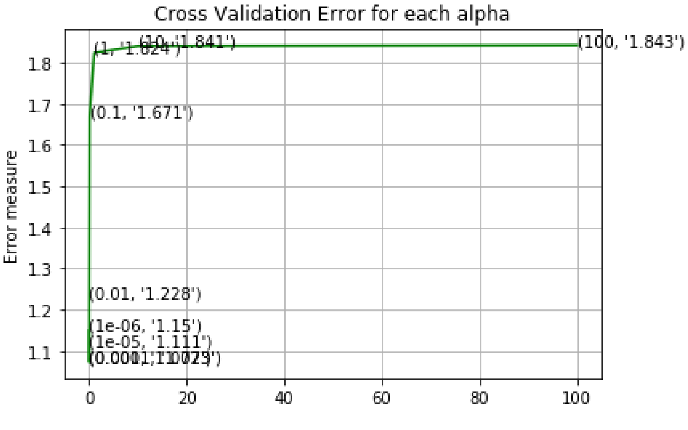
```
Log Loss : 1.840587627777384
for alpha = 100
```

```
C:\Users\Friend\Anaconda3\lib\site-packages\sklearn\linear_model\stochastic_gradient.py:128: FutureWarning: max_iter and tol parameters have been added in <class 'sklearn.linear_model.stochastic_gradient.SGDClassifier'> in 0.19. If both are left unset, they default to max_iter=5 and tol=None. If tol is not None, max_iter defaults to max_iter=1000. From 0.21, default max_iter will be 1000, and default tol will be 1e-3.
  "and default tol will be 1e-3." % type(self), FutureWarning)
C:\Users\Friend\Anaconda3\lib\site-packages\sklearn\linear_model\stochastic_gradient.py:128: FutureWarning: max_iter and tol parameters have been added in <class 'sklearn.linear_model.stochastic_gradient.SGDClassifier'> in 0.19. If both are left unset, they default to max_iter=5 and tol=None. If tol is not None, max_iter defaults to max_iter=1000. From 0.21, default max_iter will be 1000, and default tol will be 1e-3.
  "and default tol will be 1e-3." % type(self), FutureWarning)
C:\Users\Friend\Anaconda3\lib\site-packages\sklearn\linear_model\stochastic_gradient.py:128: FutureWarning: max_iter and tol parameters have been added in <class 'sklearn.linear_model.stochastic_gradient.SGDClassifier'> in 0.19. If both are left unset, they default to max_iter=5 and tol=None. If tol is not None, max_iter defaults to max_iter=1000. From 0.21, default max_iter will be 1000, and default tol will be 1e-3.
  "and default tol will be 1e-3." % type(self), FutureWarning)
```

```
Log Loss : 1.8425492407491664
```

In [88]:

```python
fig, ax = plt.subplots()
ax.plot(alpha, cv_log_error_array,c='g')
for i, txt in enumerate(np.round(cv_log_error_array,3)):
    ax.annotate((alpha[i],str(txt)), (alpha[i],cv_log_error_array[i]))
plt.grid()
plt.title("Cross Validation Error for each alpha")
plt.xlabel("Alpha i's")
plt.ylabel("Error measure")
plt.show()
```



Alpha i's

In [89]:

```
best_alpha = np.argmin(cv_log_error_array)
clf = SGDClassifier(class_weight='balanced', alpha=alpha[best_alpha], penalty='l2', loss='log', random_
state=42)
clf.fit(train_x_onehotCoding, train_y)
sig_clf = CalibratedClassifierCV(clf, method="sigmoid")
sig_clf.fit(train_x_onehotCoding, train_y)

predict_y = sig_clf.predict_proba(train_x_onehotCoding)
log_train_log_loss = log_loss(y_train, predict_y, labels=clf.classes_, eps=1e-15)
print('For values of best alpha = ', alpha[best_alpha], "The train log loss is:",log_train_log_loss)
predict_y = sig_clf.predict_proba(cv_x_onehotCoding)
log_cv_log_loss = log_loss(y_cv, predict_y, labels=clf.classes_, eps=1e-15)
print('For values of best alpha = ', alpha[best_alpha], "The cross validation log loss is:",log_cv_log_
loss)
predict_y = sig_clf.predict_proba(test_x_onehotCoding)
log_test_log_loss = log_loss(y_test, predict_y, labels=clf.classes_, eps=1e-15)
print('For values of best alpha = ', alpha[best_alpha], "The test log loss is:",log_test_log_loss)
```

```
C:\Users\Friend\Anaconda3\lib\site-packages\sklearn\linear_model\stochastic_gradient.py:128: FutureWarn
ing: max_iter and tol parameters have been added in <class 'sklearn.linear_model.stochastic_gradient.SG
DClassifier'> in 0.19. If both are left unset, they default to max_iter=5 and tol=None. If tol is not N
one, max_iter defaults to max_iter=1000. From 0.21, default max_iter will be 1000, and default tol will
be 1e-3.
  "and default tol will be 1e-3." % type(self), FutureWarning)
C:\Users\Friend\Anaconda3\lib\site-packages\sklearn\linear_model\stochastic_gradient.py:128: FutureWarn
ing: max_iter and tol parameters have been added in <class 'sklearn.linear_model.stochastic_gradient.SG
DClassifier'> in 0.19. If both are left unset, they default to max_iter=5 and tol=None. If tol is not N
one, max_iter defaults to max_iter=1000. From 0.21, default max_iter will be 1000, and default tol will
be 1e-3.
  "and default tol will be 1e-3." % type(self), FutureWarning)
C:\Users\Friend\Anaconda3\lib\site-packages\sklearn\linear_model\stochastic_gradient.py:128: FutureWarn
ing: max_iter and tol parameters have been added in <class 'sklearn.linear_model.stochastic_gradient.SG
DClassifier'> in 0.19. If both are left unset, they default to max_iter=5 and tol=None. If tol is not N
one, max_iter defaults to max_iter=1000. From 0.21, default max_iter will be 1000, and default tol will
be 1e-3.
  "and default tol will be 1e-3." % type(self), FutureWarning)
C:\Users\Friend\Anaconda3\lib\site-packages\sklearn\linear_model\stochastic_gradient.py:128: FutureWarn
ing: max_iter and tol parameters have been added in <class 'sklearn.linear_model.stochastic_gradient.SG
DClassifier'> in 0.19. If both are left unset, they default to max_iter=5 and tol=None. If tol is not N
one, max_iter defaults to max_iter=1000. From 0.21, default max_iter will be 1000, and default tol will
be 1e-3.
  "and default tol will be 1e-3." % type(self), FutureWarning)
```

```
For values of best alpha =  0.001 The train log loss is: 0.7088692795978287
For values of best alpha =  0.001 The cross validation log loss is: 1.0723810107015828
For values of best alpha =  0.001 The test log loss is: 1.0041908224441571
```

In [90]:

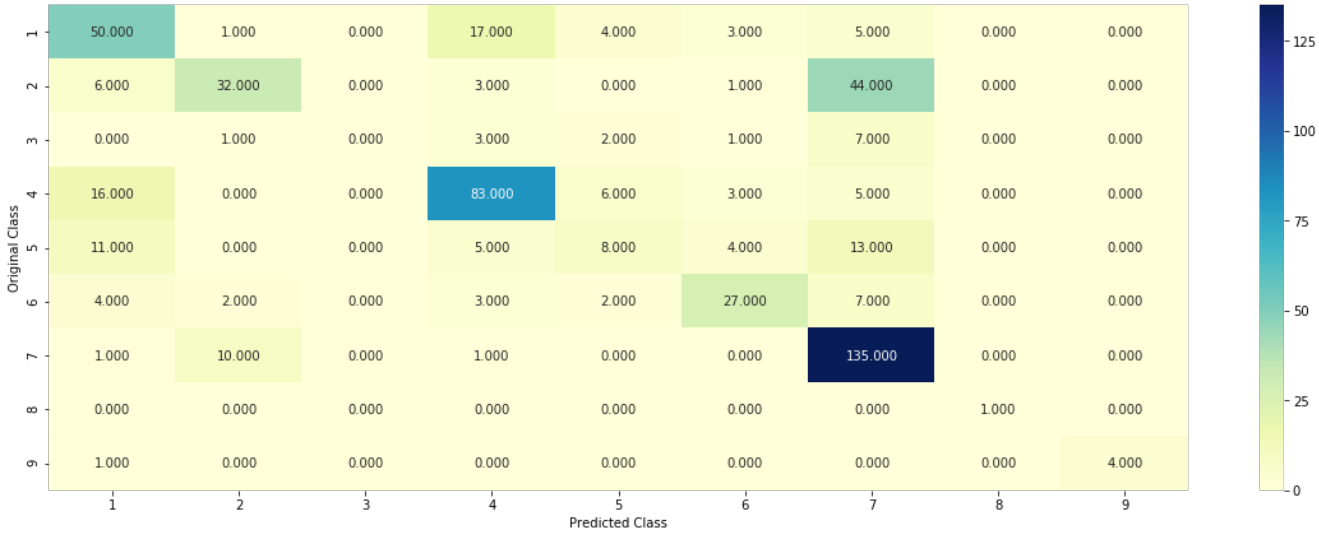
```
clf = SGDClassifier(class_weight='balanced', alpha=alpha[best_alpha], penalty='l2', loss='log', random_
state=42)
predict_and_plot_confusion_matrix(train_x_onehotCoding, train_y, cv_x_onehotCoding, cv_y, clf)
```

```
C:\Users\Friend\Anaconda3\lib\site-packages\sklearn\linear_model\stochastic_gradient.py:128: FutureWarn
ing: max_iter and tol parameters have been added in <class 'sklearn.linear_model.stochastic_gradient.SG
DClassifier'> in 0.19. If both are left unset, they default to max_iter=5 and tol=None. If tol is not N
one, max_iter defaults to max_iter=1000. From 0.21, default max_iter will be 1000, and default tol will
be 1e-3.
  "and default tol will be 1e-3." % type(self), FutureWarning)
C:\Users\Friend\Anaconda3\lib\site-packages\sklearn\linear_model\stochastic_gradient.py:128: FutureWarn
ing: max_iter and tol parameters have been added in <class 'sklearn.linear_model.stochastic_gradient.SG
DClassifier'> in 0.19. If both are left unset, they default to max_iter=5 and tol=None. If tol is not N
one, max_iter defaults to max_iter=1000. From 0.21, default max_iter will be 1000, and default tol will
be 1e-3.
  "and default tol will be 1e-3." % type(self), FutureWarning)
C:\Users\Friend\Anaconda3\lib\site-packages\sklearn\linear_model\stochastic_gradient.py:128: FutureWarn
ing: max_iter and tol parameters have been added in <class 'sklearn.linear_model.stochastic_gradient.SG
DClassifier'> in 0.19. If both are left unset, they default to max_iter=5 and tol=None. If tol is not N
one, max_iter defaults to max_iter=1000. From 0.21, default max_iter will be 1000, and default tol will
be 1e-3.
```

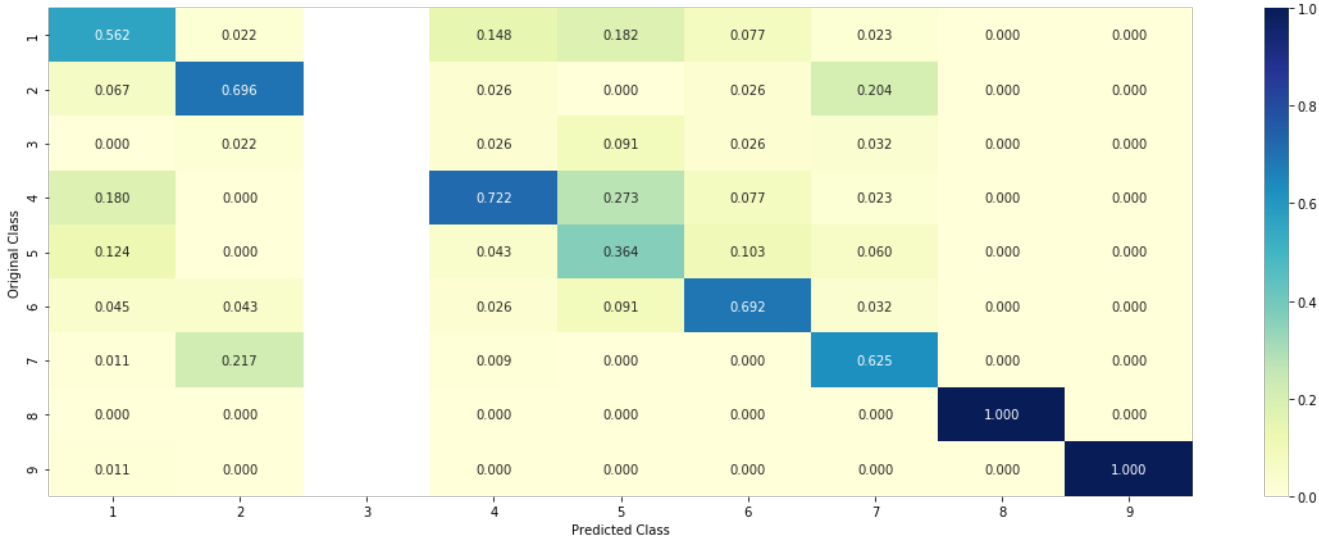
```
  "and default tol will be 1e-3." % type(self), FutureWarning)
C:\Users\Friend\Anaconda3\lib\site-packages\sklearn\linear_model\stochastic_gradient.py:128: FutureWarn
ing: max_iter and tol parameters have been added in <class 'sklearn.linear_model.stochastic_gradient.SG
DClassifier'> in 0.19. If both are left unset, they default to max_iter=5 and tol=None. If tol is not N
one, max_iter defaults to max_iter=1000. From 0.21, default max_iter will be 1000, and default tol will
be 1e-3.
  "and default tol will be 1e-3." % type(self), FutureWarning)
```

```
Log loss : 1.0723810107015828
Number of mis-classified points : 0.3609022556390977
-------------------- Confusion matrix --------------------
```

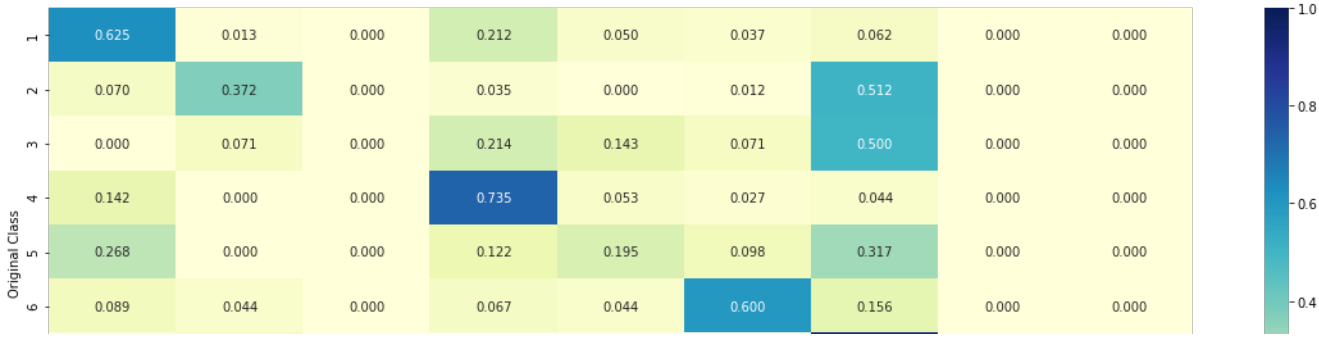
```
C:\Users\Friend\Anaconda3\lib\site-packages\ipykernel_launcher.py:7: RuntimeWarning: invalid value enco
untered in true_divide
  import sys
```

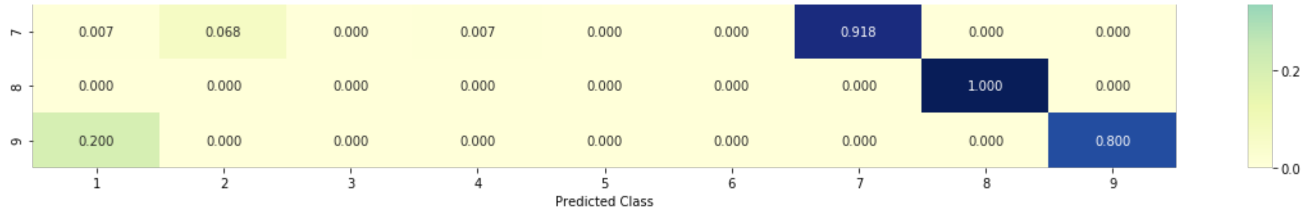



```
-------------------- Precision matrix (Column Sum=1) --------------------
```




```
-------------------- Recall matrix (Row sum=1) --------------------
```





SVM

In [91]:

```python
alpha = [10 ** x for x in range(-5, 3)]
cv_log_error_array = []
for i in alpha:
    print("for C =", i)
    clf = SGDClassifier( class_weight='balanced', alpha=i, penalty='l2', loss='hinge', random_state=42)
    clf.fit(train_x_onehotCoding, train_y)
    sig_clf = CalibratedClassifierCV(clf, method="sigmoid")
    sig_clf.fit(train_x_onehotCoding, train_y)
    sig_clf_probs = sig_clf.predict_proba(cv_x_onehotCoding)
    cv_log_error_array.append(log_loss(cv_y, sig_clf_probs, labels=clf.classes_, eps=1e-15))
    print("Log Loss :",log_loss(cv_y, sig_clf_probs))
```

```
for C = 1e-05
```

```
C:\Users\Friend\Anaconda3\lib\site-packages\sklearn\linear_model\stochastic_gradient.py:128: FutureWarn
ing: max_iter and tol parameters have been added in <class 'sklearn.linear_model.stochastic_gradient.SG
DClassifier'> in 0.19. If both are left unset, they default to max_iter=5 and tol=None. If tol is not N
one, max_iter defaults to max_iter=1000. From 0.21, default max_iter will be 1000, and default tol will
be 1e-3.
  "and default tol will be 1e-3." % type(self), FutureWarning)
C:\Users\Friend\Anaconda3\lib\site-packages\sklearn\linear_model\stochastic_gradient.py:128: FutureWarn
ing: max_iter and tol parameters have been added in <class 'sklearn.linear_model.stochastic_gradient.SG
DClassifier'> in 0.19. If both are left unset, they default to max_iter=5 and tol=None. If tol is not N
one, max_iter defaults to max_iter=1000. From 0.21, default max_iter will be 1000, and default tol will
be 1e-3.
  "and default tol will be 1e-3." % type(self), FutureWarning)
C:\Users\Friend\Anaconda3\lib\site-packages\sklearn\linear_model\stochastic_gradient.py:128: FutureWarn
ing: max_iter and tol parameters have been added in <class 'sklearn.linear_model.stochastic_gradient.SG
DClassifier'> in 0.19. If both are left unset, they default to max_iter=5 and tol=None. If tol is not N
one, max_iter defaults to max_iter=1000. From 0.21, default max_iter will be 1000, and default tol will
be 1e-3.
  "and default tol will be 1e-3." % type(self), FutureWarning)
C:\Users\Friend\Anaconda3\lib\site-packages\sklearn\linear_model\stochastic_gradient.py:128: FutureWarn
ing: max_iter and tol parameters have been added in <class 'sklearn.linear_model.stochastic_gradient.SG
DClassifier'> in 0.19. If both are left unset, they default to max_iter=5 and tol=None. If tol is not N
one, max_iter defaults to max_iter=1000. From 0.21, default max_iter will be 1000, and default tol will
be 1e-3.
  "and default tol will be 1e-3." % type(self), FutureWarning)
```

```
Log Loss : 1.13676971957375
for C = 0.0001
```

```
C:\Users\Friend\Anaconda3\lib\site-packages\sklearn\linear_model\stochastic_gradient.py:128: FutureWarn
ing: max_iter and tol parameters have been added in <class 'sklearn.linear_model.stochastic_gradient.SG
DClassifier'> in 0.19. If both are left unset, they default to max_iter=5 and tol=None. If tol is not N
one, max_iter defaults to max_iter=1000. From 0.21, default max_iter will be 1000, and default tol will
be 1e-3.
  "and default tol will be 1e-3." % type(self), FutureWarning)
C:\Users\Friend\Anaconda3\lib\site-packages\sklearn\linear_model\stochastic_gradient.py:128: FutureWarn
ing: max_iter and tol parameters have been added in <class 'sklearn.linear_model.stochastic_gradient.SG
DClassifier'> in 0.19. If both are left unset, they default to max_iter=5 and tol=None. If tol is not N
one, max_iter defaults to max_iter=1000. From 0.21, default max_iter will be 1000, and default tol will
be 1e-3.
  "and default tol will be 1e-3." % type(self), FutureWarning)
C:\Users\Friend\Anaconda3\lib\site-packages\sklearn\linear_model\stochastic_gradient.py:128: FutureWarn
ing: max_iter and tol parameters have been added in <class 'sklearn.linear_model.stochastic_gradient.SG
DClassifier'> in 0.19. If both are left unset, they default to max_iter=5 and tol=None. If tol is not N
one, max_iter defaults to max_iter=1000. From 0.21, default max_iter will be 1000, and default tol will
be 1e-3.
  "and default tol will be 1e-3." % type(self), FutureWarning)
```

```
Log Loss : 1.0799975073717587
for C = 0.001
```

```
C:\Users\Friend\Anaconda3\lib\site-packages\sklearn\linear_model\stochastic_gradient.py:128: FutureWarn
ing: max_iter and tol parameters have been added in <class 'sklearn.linear_model.stochastic_gradient.SG
DClassifier'> in 0.19. If both are left unset, they default to max_iter=5 and tol=None. If tol is not N
one, max_iter defaults to max_iter=1000. From 0.21, default max_iter will be 1000, and default tol will
be 1e-3.
  "and default tol will be 1e-3." % type(self), FutureWarning)
C:\Users\Friend\Anaconda3\lib\site-packages\sklearn\linear_model\stochastic_gradient.py:128: FutureWarn
ing: max_iter and tol parameters have been added in <class 'sklearn.linear_model.stochastic_gradient.SG
DClassifier'> in 0.19. If both are left unset, they default to max_iter=5 and tol=None. If tol is not N
one, max_iter defaults to max_iter=1000. From 0.21, default max_iter will be 1000, and default tol will
be 1e-3.
  "and default tol will be 1e-3." % type(self), FutureWarning)
C:\Users\Friend\Anaconda3\lib\site-packages\sklearn\linear_model\stochastic_gradient.py:128: FutureWarn
ing: max_iter and tol parameters have been added in <class 'sklearn.linear_model.stochastic_gradient.SG
DClassifier'> in 0.19. If both are left unset, they default to max_iter=5 and tol=None. If tol is not N
one, max_iter defaults to max_iter=1000. From 0.21, default max_iter will be 1000, and default tol will
be 1e-3.
  "and default tol will be 1e-3." % type(self), FutureWarning)
```

```
Log Loss : 1.0972792118081969
for C = 0.01
```

```
C:\Users\Friend\Anaconda3\lib\site-packages\sklearn\linear_model\stochastic_gradient.py:128: FutureWarn
ing: max_iter and tol parameters have been added in <class 'sklearn.linear_model.stochastic_gradient.SG
DClassifier'> in 0.19. If both are left unset, they default to max_iter=5 and tol=None. If tol is not N
one, max_iter defaults to max_iter=1000. From 0.21, default max_iter will be 1000, and default tol will
be 1e-3.
  "and default tol will be 1e-3." % type(self), FutureWarning)
C:\Users\Friend\Anaconda3\lib\site-packages\sklearn\linear_model\stochastic_gradient.py:128: FutureWarn
ing: max_iter and tol parameters have been added in <class 'sklearn.linear_model.stochastic_gradient.SG
DClassifier'> in 0.19. If both are left unset, they default to max_iter=5 and tol=None. If tol is not N
one, max_iter defaults to max_iter=1000. From 0.21, default max_iter will be 1000, and default tol will
be 1e-3.
  "and default tol will be 1e-3." % type(self), FutureWarning)
C:\Users\Friend\Anaconda3\lib\site-packages\sklearn\linear_model\stochastic_gradient.py:128: FutureWarn
ing: max_iter and tol parameters have been added in <class 'sklearn.linear_model.stochastic_gradient.SG
DClassifier'> in 0.19. If both are left unset, they default to max_iter=5 and tol=None. If tol is not N
one, max_iter defaults to max_iter=1000. From 0.21, default max_iter will be 1000, and default tol will
be 1e-3.
  "and default tol will be 1e-3." % type(self), FutureWarning)
```

```
Log Loss : 1.3703162735588812
for C = 0.1
```

```
C:\Users\Friend\Anaconda3\lib\site-packages\sklearn\linear_model\stochastic_gradient.py:128: FutureWarn
ing: max_iter and tol parameters have been added in <class 'sklearn.linear_model.stochastic_gradient.SG
DClassifier'> in 0.19. If both are left unset, they default to max_iter=5 and tol=None. If tol is not N
one, max_iter defaults to max_iter=1000. From 0.21, default max_iter will be 1000, and default tol will
be 1e-3.
  "and default tol will be 1e-3." % type(self), FutureWarning)
C:\Users\Friend\Anaconda3\lib\site-packages\sklearn\linear_model\stochastic_gradient.py:128: FutureWarn
ing: max_iter and tol parameters have been added in <class 'sklearn.linear_model.stochastic_gradient.SG
DClassifier'> in 0.19. If both are left unset, they default to max_iter=5 and tol=None. If tol is not N
one, max_iter defaults to max_iter=1000. From 0.21, default max_iter will be 1000, and default tol will
be 1e-3.
  "and default tol will be 1e-3." % type(self), FutureWarning)
C:\Users\Friend\Anaconda3\lib\site-packages\sklearn\linear_model\stochastic_gradient.py:128: FutureWarn
ing: max_iter and tol parameters have been added in <class 'sklearn.linear_model.stochastic_gradient.SG
DClassifier'> in 0.19. If both are left unset, they default to max_iter=5 and tol=None. If tol is not N
one, max_iter defaults to max_iter=1000. From 0.21, default max_iter will be 1000, and default tol will
be 1e-3.
  "and default tol will be 1e-3." % type(self), FutureWarning)
```

```
Log Loss : 1.657609695450692
for C = 1
```

```
C:\Users\Friend\Anaconda3\lib\site-packages\sklearn\linear_model\stochastic_gradient.py:128: FutureWarn
ing: max_iter and tol parameters have been added in <class 'sklearn.linear_model.stochastic_gradient.SG
```

```
DClassifier'> in 0.19. If both are left unset, they default to max_iter=5 and tol=None. If tol is not N
one, max_iter defaults to max_iter=1000. From 0.21, default max_iter will be 1000, and default tol will
be 1e-3.
  "and default tol will be 1e-3." % type(self), FutureWarning)
C:\Users\Friend\Anaconda3\lib\site-packages\sklearn\linear_model\stochastic_gradient.py:128: FutureWarn
ing: max_iter and tol parameters have been added in <class 'sklearn.linear_model.stochastic_gradient.SG
DClassifier'> in 0.19. If both are left unset, they default to max_iter=5 and tol=None. If tol is not N
one, max_iter defaults to max_iter=1000. From 0.21, default max_iter will be 1000, and default tol will
be 1e-3.
  "and default tol will be 1e-3." % type(self), FutureWarning)
C:\Users\Friend\Anaconda3\lib\site-packages\sklearn\linear_model\stochastic_gradient.py:128: FutureWarn
ing: max_iter and tol parameters have been added in <class 'sklearn.linear_model.stochastic_gradient.SG
DClassifier'> in 0.19. If both are left unset, they default to max_iter=5 and tol=None. If tol is not N
one, max_iter defaults to max_iter=1000. From 0.21, default max_iter will be 1000, and default tol will
be 1e-3.
  "and default tol will be 1e-3." % type(self), FutureWarning)
```

```
Log Loss : 1.8431465651070733
for C = 10
```

```
C:\Users\Friend\Anaconda3\lib\site-packages\sklearn\linear_model\stochastic_gradient.py:128: FutureWarn
ing: max_iter and tol parameters have been added in <class 'sklearn.linear_model.stochastic_gradient.SG
DClassifier'> in 0.19. If both are left unset, they default to max_iter=5 and tol=None. If tol is not N
one, max_iter defaults to max_iter=1000. From 0.21, default max_iter will be 1000, and default tol will
be 1e-3.
  "and default tol will be 1e-3." % type(self), FutureWarning)
C:\Users\Friend\Anaconda3\lib\site-packages\sklearn\linear_model\stochastic_gradient.py:128: FutureWarn
ing: max_iter and tol parameters have been added in <class 'sklearn.linear_model.stochastic_gradient.SG
DClassifier'> in 0.19. If both are left unset, they default to max_iter=5 and tol=None. If tol is not N
one, max_iter defaults to max_iter=1000. From 0.21, default max_iter will be 1000, and default tol will
be 1e-3.
  "and default tol will be 1e-3." % type(self), FutureWarning)
C:\Users\Friend\Anaconda3\lib\site-packages\sklearn\linear_model\stochastic_gradient.py:128: FutureWarn
ing: max_iter and tol parameters have been added in <class 'sklearn.linear_model.stochastic_gradient.SG
DClassifier'> in 0.19. If both are left unset, they default to max_iter=5 and tol=None. If tol is not N
one, max_iter defaults to max_iter=1000. From 0.21, default max_iter will be 1000, and default tol will
be 1e-3.
  "and default tol will be 1e-3." % type(self), FutureWarning)
```

```
Log Loss : 1.8431465355315242
for C = 100
```

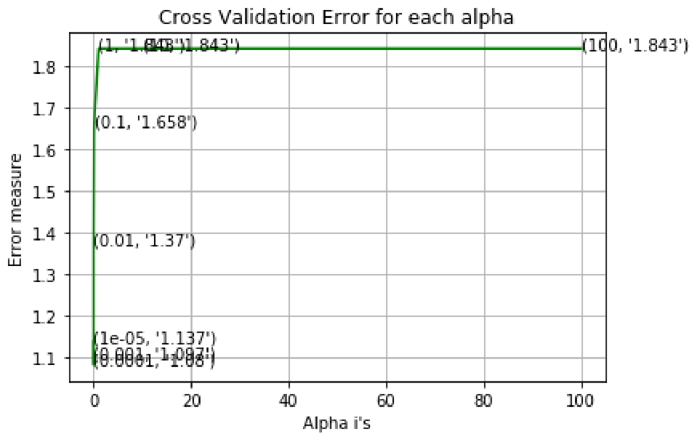
```
C:\Users\Friend\Anaconda3\lib\site-packages\sklearn\linear_model\stochastic_gradient.py:128: FutureWarn
ing: max_iter and tol parameters have been added in <class 'sklearn.linear_model.stochastic_gradient.SG
DClassifier'> in 0.19. If both are left unset, they default to max_iter=5 and tol=None. If tol is not N
one, max_iter defaults to max_iter=1000. From 0.21, default max_iter will be 1000, and default tol will
be 1e-3.
  "and default tol will be 1e-3." % type(self), FutureWarning)
C:\Users\Friend\Anaconda3\lib\site-packages\sklearn\linear_model\stochastic_gradient.py:128: FutureWarn
ing: max_iter and tol parameters have been added in <class 'sklearn.linear_model.stochastic_gradient.SG
DClassifier'> in 0.19. If both are left unset, they default to max_iter=5 and tol=None. If tol is not N
one, max_iter defaults to max_iter=1000. From 0.21, default max_iter will be 1000, and default tol will
be 1e-3.
  "and default tol will be 1e-3." % type(self), FutureWarning)
C:\Users\Friend\Anaconda3\lib\site-packages\sklearn\linear_model\stochastic_gradient.py:128: FutureWarn
ing: max_iter and tol parameters have been added in <class 'sklearn.linear_model.stochastic_gradient.SG
DClassifier'> in 0.19. If both are left unset, they default to max_iter=5 and tol=None. If tol is not N
one, max_iter defaults to max_iter=1000. From 0.21, default max_iter will be 1000, and default tol will
be 1e-3.
  "and default tol will be 1e-3." % type(self), FutureWarning)
```

```
Log Loss : 1.8431465715711939
```

In [92]:

```
fig, ax = plt.subplots()
ax.plot(alpha, cv_log_error_array,c='g')
for i, txt in enumerate(np.round(cv_log_error_array,3)):
    ax.annotate((alpha[i],str(txt)), (alpha[i],cv_log_error_array[i]))
plt.grid()
plt.title("Cross Validation Error for each alpha")
plt.xlabel("Alpha i's")
plt.ylabel("Error measure")
```

```
plt.show()
```



In [93]:

```
best_alpha = np.argmin(cv_log_error_array)

clf = SGDClassifier(class_weight='balanced', alpha=alpha[best_alpha], penalty='l2', loss='hinge', rando
m_state=42)
clf.fit(train_x_onehotCoding, train_y)
sig_clf = CalibratedClassifierCV(clf, method="sigmoid")
sig_clf.fit(train_x_onehotCoding, train_y)

predict_y = sig_clf.predict_proba(train_x_onehotCoding)
svm_train_log_loss = log_loss(y_train, predict_y, labels=clf.classes_, eps=1e-15)
print('For values of best alpha = ', alpha[best_alpha], "The train log loss is:",svm_train_log_loss)
predict_y = sig_clf.predict_proba(cv_x_onehotCoding)
svm_cv_log_loss = log_loss(y_cv, predict_y, labels=clf.classes_, eps=1e-15)
print('For values of best alpha = ', alpha[best_alpha], "The cross validation log loss is:",svm_cv_log_
loss)
predict_y = sig_clf.predict_proba(test_x_onehotCoding)
svm_test_log_loss = log_loss(y_test, predict_y, labels=clf.classes_, eps=1e-15)
print('For values of best alpha = ', alpha[best_alpha], "The test log loss is:",svm_test_log_loss)
```

```
C:\Users\Friend\Anaconda3\lib\site-packages\sklearn\linear_model\stochastic_gradient.py:128: FutureWarn
ing: max_iter and tol parameters have been added in <class 'sklearn.linear_model.stochastic_gradient.SG
DClassifier'> in 0.19. If both are left unset, they default to max_iter=5 and tol=None. If tol is not N
one, max_iter defaults to max_iter=1000. From 0.21, default max_iter will be 1000, and default tol will
be 1e-3.
  "and default tol will be 1e-3." % type(self), FutureWarning)
C:\Users\Friend\Anaconda3\lib\site-packages\sklearn\linear_model\stochastic_gradient.py:128: FutureWarn
ing: max_iter and tol parameters have been added in <class 'sklearn.linear_model.stochastic_gradient.SG
DClassifier'> in 0.19. If both are left unset, they default to max_iter=5 and tol=None. If tol is not N
one, max_iter defaults to max_iter=1000. From 0.21, default max_iter will be 1000, and default tol will
be 1e-3.
  "and default tol will be 1e-3." % type(self), FutureWarning)
C:\Users\Friend\Anaconda3\lib\site-packages\sklearn\linear_model\stochastic_gradient.py:128: FutureWarn
ing: max_iter and tol parameters have been added in <class 'sklearn.linear_model.stochastic_gradient.SG
DClassifier'> in 0.19. If both are left unset, they default to max_iter=5 and tol=None. If tol is not N
one, max_iter defaults to max_iter=1000. From 0.21, default max_iter will be 1000, and default tol will
be 1e-3.
  "and default tol will be 1e-3." % type(self), FutureWarning)
C:\Users\Friend\Anaconda3\lib\site-packages\sklearn\linear_model\stochastic_gradient.py:128: FutureWarn
ing: max_iter and tol parameters have been added in <class 'sklearn.linear_model.stochastic_gradient.SG
DClassifier'> in 0.19. If both are left unset, they default to max_iter=5 and tol=None. If tol is not N
one, max_iter defaults to max_iter=1000. From 0.21, default max_iter will be 1000, and default tol will
be 1e-3.
  "and default tol will be 1e-3." % type(self), FutureWarning)
```

```
For values of best alpha =  0.0001 The train log loss is: 0.4926686683306196
For values of best alpha =  0.0001 The cross validation log loss is: 1.0799975073717587
For values of best alpha =  0.0001 The test log loss is: 1.0345586207144635
```

In [94]:

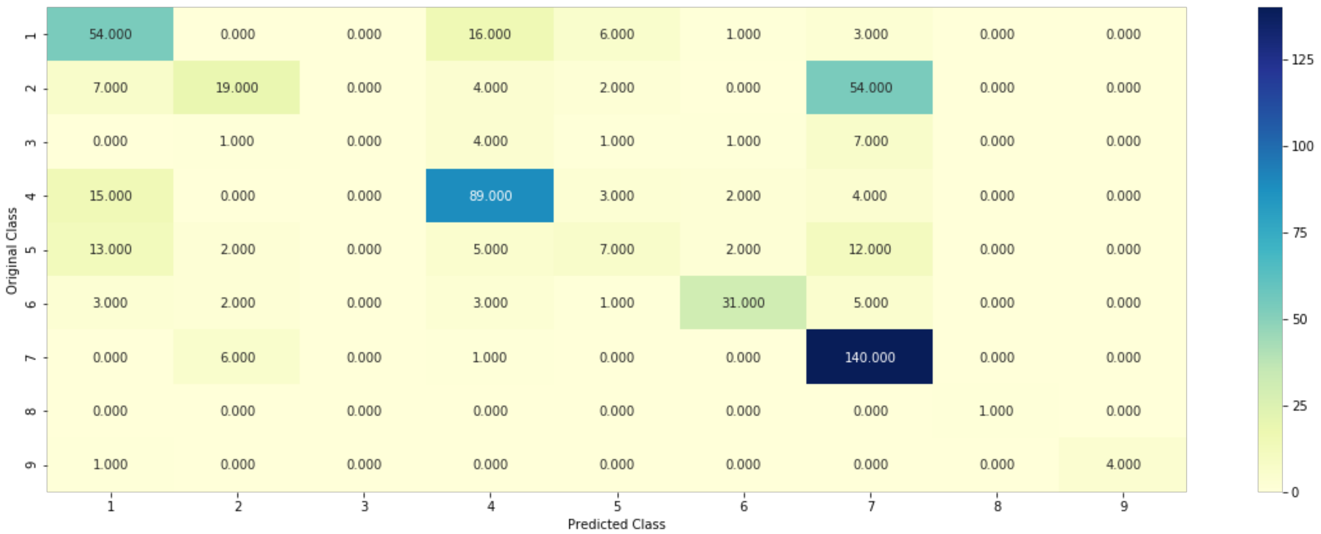
```
clf = SGDClassifier(alpha=alpha[best_alpha], penalty='l2', loss='hinge', random_state=42,class_weight='
balanced')
predict_and_plot_confusion_matrix(train_x_onehotCoding, train_y,cv_x_onehotCoding,cv_y, clf)
```

```
predict_and_plot_confusion_matrix(train_x_onehotCoding, train_y,cv_x_onehotCoding,cv_y, clf)
```

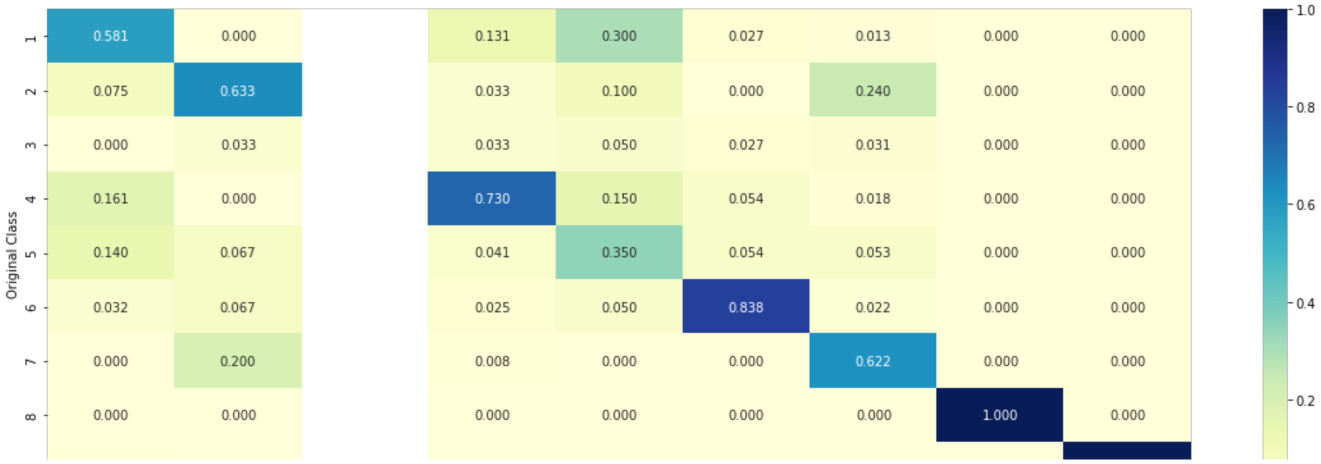
```
C:\Users\Friend\Anaconda3\lib\site-packages\sklearn\linear_model\stochastic_gradient.py:128: FutureWarning: max_iter and tol parameters have been added in <class 'sklearn.linear_model.stochastic_gradient.SGDClassifier'> in 0.19. If both are left unset, they default to max_iter=5 and tol=None. If tol is not None, max_iter defaults to max_iter=1000. From 0.21, default max_iter will be 1000, and default tol will be 1e-3.
  "and default tol will be 1e-3." % type(self), FutureWarning)
C:\Users\Friend\Anaconda3\lib\site-packages\sklearn\linear_model\stochastic_gradient.py:128: FutureWarning: max_iter and tol parameters have been added in <class 'sklearn.linear_model.stochastic_gradient.SGDClassifier'> in 0.19. If both are left unset, they default to max_iter=5 and tol=None. If tol is not None, max_iter defaults to max_iter=1000. From 0.21, default max_iter will be 1000, and default tol will be 1e-3.
  "and default tol will be 1e-3." % type(self), FutureWarning)
C:\Users\Friend\Anaconda3\lib\site-packages\sklearn\linear_model\stochastic_gradient.py:128: FutureWarning: max_iter and tol parameters have been added in <class 'sklearn.linear_model.stochastic_gradient.SGDClassifier'> in 0.19. If both are left unset, they default to max_iter=5 and tol=None. If tol is not None, max_iter defaults to max_iter=1000. From 0.21, default max_iter will be 1000, and default tol will be 1e-3.
  "and default tol will be 1e-3." % type(self), FutureWarning)
C:\Users\Friend\Anaconda3\lib\site-packages\sklearn\linear_model\stochastic_gradient.py:128: FutureWarning: max_iter and tol parameters have been added in <class 'sklearn.linear_model.stochastic_gradient.SGDClassifier'> in 0.19. If both are left unset, they default to max_iter=5 and tol=None. If tol is not None, max_iter defaults to max_iter=1000. From 0.21, default max_iter will be 1000, and default tol will be 1e-3.
  "and default tol will be 1e-3." % type(self), FutureWarning)
```

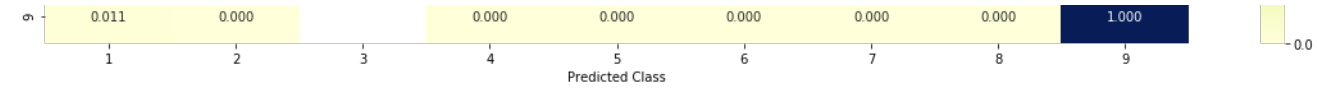
```
Log loss : 1.0799975073717587
Number of mis-classified points : 0.35150375939849626
-------------------- Confusion matrix --------------------
```

```
C:\Users\Friend\Anaconda3\lib\site-packages\ipykernel_launcher.py:7: RuntimeWarning: invalid value encountered in true_divide
  import sys
```

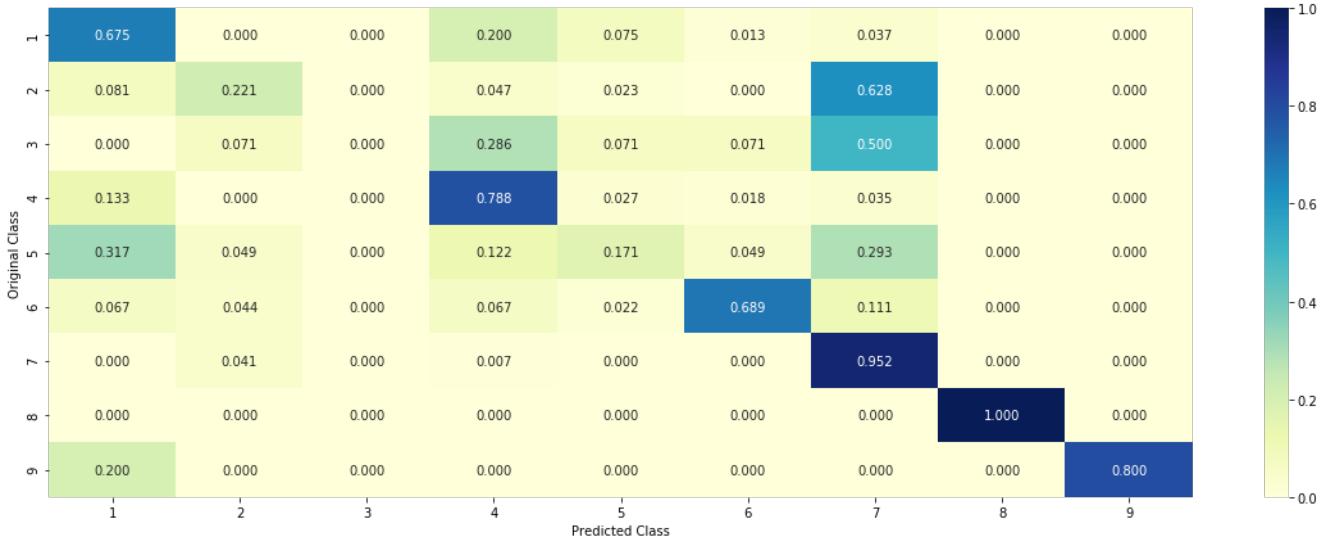



```
-------------------- Precision matrix (Columm Sum=1) --------------------
```





---------- Recall matrix (Row sum=1) ----------



Random Forest

In [95]:

```
from sklearn.ensemble import RandomForestClassifier

alpha = [100,200,500,1000,2000]
max_depth = [5, 10]
cv_log_error_array = []
for i in alpha:
    for j in max_depth:
        print("for n_estimators =", i,"and max depth = ", j)
        clf = RandomForestClassifier(n_estimators=i, criterion='gini', max_depth=j, random_state=42, n_
jobs=-1)
        clf.fit(train_x_onehotCoding, train_y)
        sig_clf = CalibratedClassifierCV(clf, method="sigmoid")
        sig_clf.fit(train_x_onehotCoding, train_y)
        sig_clf_probs = sig_clf.predict_proba(cv_x_onehotCoding)
        cv_log_error_array.append(log_loss(cv_y, sig_clf_probs, labels=clf.classes_, eps=1e-15))
        print("Log Loss :",log_loss(cv_y, sig_clf_probs))
```

```
for n_estimators = 100 and max depth =  5
Log Loss : 1.2539856101506972
for n_estimators = 100 and max depth =  10
Log Loss : 1.2833631183999452
for n_estimators = 200 and max depth =  5
Log Loss : 1.2393182790123813
for n_estimators = 200 and max depth =  10
Log Loss : 1.264915098906191
for n_estimators = 500 and max depth =  5
Log Loss : 1.2312803718325231
for n_estimators = 500 and max depth =  10
Log Loss : 1.2512042015828562
for n_estimators = 1000 and max depth =  5
Log Loss : 1.2328866870037252
for n_estimators = 1000 and max depth =  10
Log Loss : 1.246881969239195
for n_estimators = 2000 and max depth =  5
Log Loss : 1.229275015767204
for n_estimators = 2000 and max depth =  10
Log Loss : 1.2463955670222122
```

In [99]:

```
best_alpha = np.argmin(cv_log_error_array)
```

```
clf = RandomForestClassifier(n_estimators=alpha[int(best_alpha/2)], criterion='gini', max_depth=max_dep
th[int(best_alpha%2)], random_state=42, n_jobs=-1)
clf.fit(train_x_onehotCoding, train_y)
sig_clf = CalibratedClassifierCV(clf, method="sigmoid")
sig_clf.fit(train_x_onehotCoding, train_y)

predict_y = sig_clf.predict_proba(train_x_onehotCoding)
forest_train_log_loss = log_loss(y_train, predict_y, labels=clf.classes_, eps=1e-15)
print('For values of best estimator = ', alpha[int(best_alpha/2)], "The train log loss is:",)
predict_y = sig_clf.predict_proba(cv_x_onehotCoding)
forest_cv_log_loss = log_loss(y_cv, predict_y, labels=clf.classes_, eps=1e-15)
print('For values of best estimator = ', alpha[int(best_alpha/2)], "The cross validation log loss is:",
)
predict_y = sig_clf.predict_proba(test_x_onehotCoding)
forest_test_log_loss = log_loss(y_test, predict_y, labels=clf.classes_, eps=1e-15)
print('For values of best estimator = ', alpha[int(best_alpha/2)], "The test log loss is:",)
```

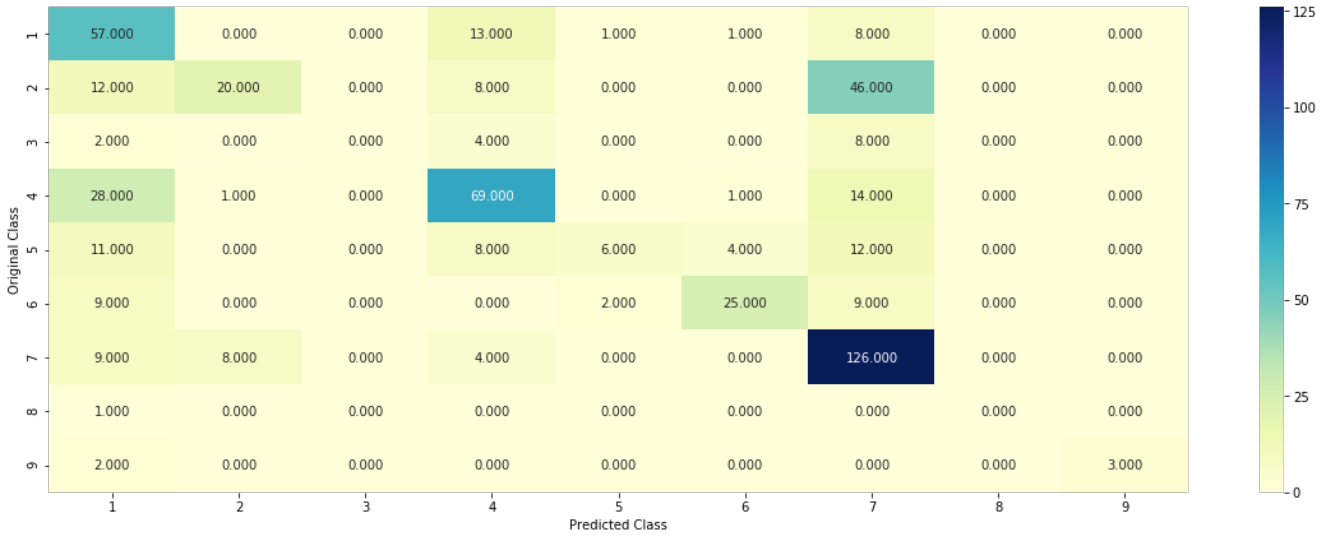
```
For values of best estimator =  2000 The train log loss is:
For values of best estimator =  2000 The cross validation log loss is:
For values of best estimator =  2000 The test log loss is:
```

In [101]:

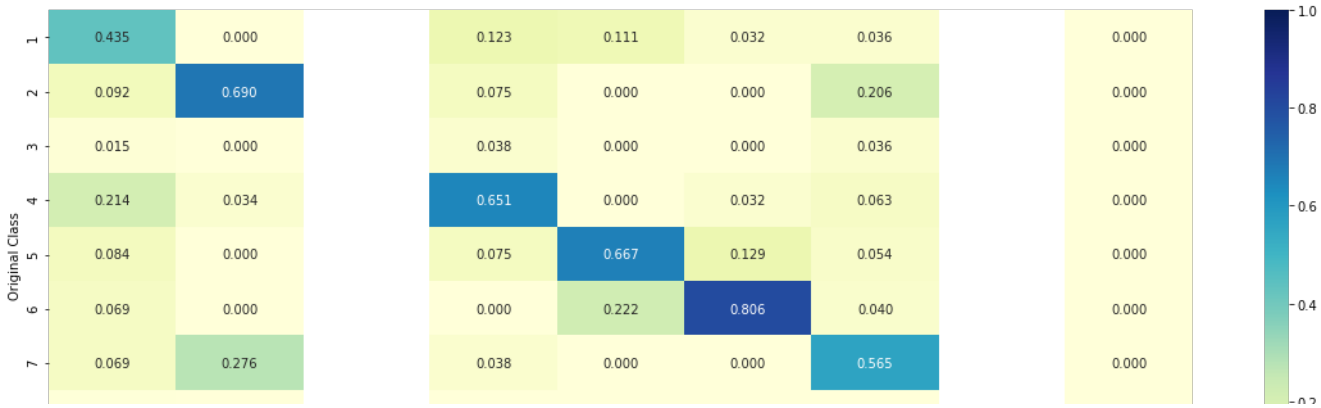
```
clf = RandomForestClassifier(n_estimators=alpha[int(best_alpha/2)], criterion='gini', max_depth=max_dep
th[int(best_alpha%2)], random_state=42, n_jobs=-1)
predict_and_plot_confusion_matrix(train_x_onehotCoding, train_y,cv_x_onehotCoding,cv_y, clf)
```

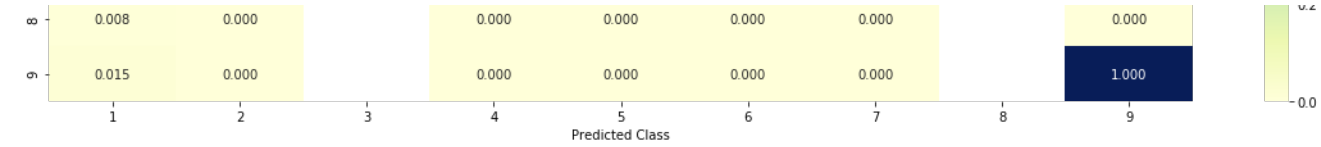
```
Log loss : 1.229275015767204
Number of mis-classified points : 0.424812030075188
-------------------- Confusion matrix --------------------
```

```
C:\Users\Friend\Anaconda3\lib\site-packages\ipykernel_launcher.py:7: RuntimeWarning: invalid value enco
untered in true_divide
  import sys
```

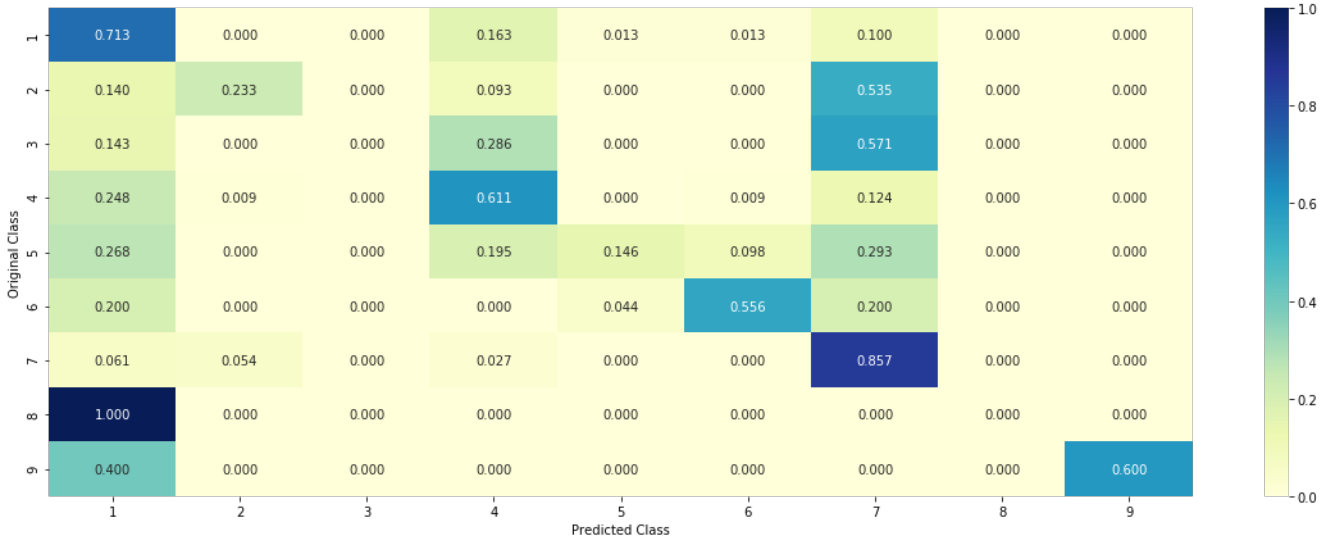



```
-------------------- Precision matrix (Column Sum=1) --------------------
```





---------- Recall matrix (Row sum=1) ----------



Conclusion

In [100]:

```
from prettytable import PrettyTable

Table = PrettyTable()

Table.field_names  = ["Model", "Train Loss", "Cross Validation Loss", "Test Loss"]

Table.add_row(["K-NN", k_train_log_loss, k_cv_log_loss,k_test_log_loss])
Table.add_row(["Naive Bayes", naive_train_log_loss, naive_cv_log_loss,naive_test_log_loss])
Table.add_row(["Logistic Regression", log_train_log_loss, log_cv_log_loss,log_test_log_loss])
Table.add_row(["Support Vector Machines", svm_train_log_loss, svm_cv_log_loss,svm_test_log_loss])
Table.add_row(["Random Forest", forest_train_log_loss, forest_cv_log_loss,forest_test_log_loss])

print(Table)
```

```
+-------------------------+--------------------+-----------------------+--------------------+
|          Model          |     Train Loss     | Cross Validation Loss |     Test Loss      |
+-------------------------+--------------------+-----------------------+--------------------+
|           K-NN          | 0.9582284186052225 |   1.1125636787444793  |  1.08911155067031  |
|       Naive Bayes       | 0.5176799623398368 |   1.2219297058574758  | 1.1679082869734096 |
|   Logistic Regression   | 0.7088692795978287 |   1.0723810107015828  | 1.0041908224441571 |
| Support Vector Machines | 0.4926686683306196 |   1.0799975073717587  | 1.0345586207144635 |
|      Random Forest      | 0.8476740604468344 |   1.229275015767204   | 1.144385287527102  |
+-------------------------+--------------------+-----------------------+--------------------+
```

Logistic Regression(Count vectorizer-Unigram + Bigram)

In [55]:

```
train_x_onehotCoding = hstack((train_gene_feature_onehotCoding, train_variation_feature_onehotCoding,co
unt_train_text_feature_onehotCoding,count_bi_train_text_feature_onehotCoding))
cv_x_onehotCoding = hstack((cv_gene_feature_onehotCoding, cv_variation_feature_onehotCoding,count_cv_te
xt_feature_onehotCoding, count_bi_cv_text_feature_onehotCoding))
test_x_onehotCoding = hstack((test_gene_feature_onehotCoding, test_variation_feature_onehotCoding,count
_test_text_feature_onehotCoding, count_bi_test_text_feature_onehotCoding))
```

In [67]:

```
print(train_x_onehotCoding.shape,cv_x_onehotCoding.shape,test_x_onehotCoding.shape)
```

```
(2124, 741047) (532, 741047) (665, 741047)
```

Logistic Regression(Count vectorizer-bigram)

In [65]:

```
from sklearn.linear_model import SGDClassifier

alpha = [10 ** x for x in range(-6, 3)]
cv_log_error_array = []
for i in alpha:
    print("for alpha =", i)
    clf = SGDClassifier(class_weight='balanced', alpha=i, penalty='l2', loss='log', random_state=42)
    clf.fit(train_x_onehotCoding, train_y)
    sig_clf = CalibratedClassifierCV(clf, method="sigmoid")
    sig_clf.fit(train_x_onehotCoding, train_y)
    sig_clf_probs = sig_clf.predict_proba(cv_x_onehotCoding)
    cv_log_error_array.append(log_loss(cv_y, sig_clf_probs, labels=clf.classes_, eps=1e-15))
    print("Log Loss :",log_loss(cv_y, sig_clf_probs))
```

```
for alpha = 1e-06
```

```
C:\Users\Friend\Anaconda3\lib\site-packages\sklearn\linear_model\stochastic_gradient.py:128: FutureWarn
ing: max_iter and tol parameters have been added in <class 'sklearn.linear_model.stochastic_gradient.SG
DClassifier'> in 0.19. If both are left unset, they default to max_iter=5 and tol=None. If tol is not N
one, max_iter defaults to max_iter=1000. From 0.21, default max_iter will be 1000, and default tol will
be 1e-3.
  "and default tol will be 1e-3." % type(self), FutureWarning)
C:\Users\Friend\Anaconda3\lib\site-packages\sklearn\linear_model\stochastic_gradient.py:128: FutureWarn
ing: max_iter and tol parameters have been added in <class 'sklearn.linear_model.stochastic_gradient.SG
DClassifier'> in 0.19. If both are left unset, they default to max_iter=5 and tol=None. If tol is not N
one, max_iter defaults to max_iter=1000. From 0.21, default max_iter will be 1000, and default tol will
be 1e-3.
  "and default tol will be 1e-3." % type(self), FutureWarning)
C:\Users\Friend\Anaconda3\lib\site-packages\sklearn\calibration.py:447: RuntimeWarning: invalid value e
ncountered in multiply
  TEP_minus_T1P = P * (T * E - T1)
C:\Users\Friend\Anaconda3\lib\site-packages\sklearn\calibration.py:447: RuntimeWarning: invalid value e
ncountered in multiply
  TEP_minus_T1P = P * (T * E - T1)
C:\Users\Friend\Anaconda3\lib\site-packages\sklearn\calibration.py:447: RuntimeWarning: invalid value e
ncountered in multiply
  TEP_minus_T1P = P * (T * E - T1)
C:\Users\Friend\Anaconda3\lib\site-packages\sklearn\calibration.py:447: RuntimeWarning: invalid value e
ncountered in multiply
  TEP_minus_T1P = P * (T * E - T1)
C:\Users\Friend\Anaconda3\lib\site-packages\sklearn\calibration.py:447: RuntimeWarning: invalid value e
ncountered in multiply
  TEP_minus_T1P = P * (T * E - T1)
C:\Users\Friend\Anaconda3\lib\site-packages\sklearn\calibration.py:447: RuntimeWarning: invalid value e
ncountered in multiply
  TEP_minus_T1P = P * (T * E - T1)
C:\Users\Friend\Anaconda3\lib\site-packages\sklearn\calibration.py:447: RuntimeWarning: invalid value e
ncountered in multiply
  TEP_minus_T1P = P * (T * E - T1)
C:\Users\Friend\Anaconda3\lib\site-packages\sklearn\calibration.py:447: RuntimeWarning: invalid value e
ncountered in multiply
  TEP_minus_T1P = P * (T * E - T1)
C:\Users\Friend\Anaconda3\lib\site-packages\sklearn\calibration.py:447: RuntimeWarning: invalid value e
ncountered in multiply
  TEP_minus_T1P = P * (T * E - T1)
C:\Users\Friend\Anaconda3\lib\site-packages\sklearn\linear_model\stochastic_gradient.py:128: FutureWarn
ing: max_iter and tol parameters have been added in <class 'sklearn.linear_model.stochastic_gradient.SG
DClassifier'> in 0.19. If both are left unset, they default to max_iter=5 and tol=None. If tol is not N
one, max_iter defaults to max_iter=1000. From 0.21, default max_iter will be 1000, and default tol will
be 1e-3.
  "and default tol will be 1e-3." % type(self), FutureWarning)
C:\Users\Friend\Anaconda3\lib\site-packages\sklearn\calibration.py:447: RuntimeWarning: invalid value e
ncountered in multiply
  TEP_minus_T1P = P * (T * E - T1)
```

```
C:\Users\Friend\Anaconda3\lib\site-packages\sklearn\calibration.py:447: RuntimeWarning: invalid value e
ncountered in multiply
  TEP_minus_T1P = P * (T * E - T1)
C:\Users\Friend\Anaconda3\lib\site-packages\sklearn\calibration.py:447: RuntimeWarning: invalid value e
ncountered in multiply
  TEP_minus_T1P = P * (T * E - T1)
C:\Users\Friend\Anaconda3\lib\site-packages\sklearn\calibration.py:447: RuntimeWarning: invalid value e
ncountered in multiply
  TEP_minus_T1P = P * (T * E - T1)
C:\Users\Friend\Anaconda3\lib\site-packages\sklearn\calibration.py:447: RuntimeWarning: invalid value e
ncountered in multiply
  TEP_minus_T1P = P * (T * E - T1)
C:\Users\Friend\Anaconda3\lib\site-packages\sklearn\calibration.py:447: RuntimeWarning: invalid value e
ncountered in multiply
  TEP_minus_T1P = P * (T * E - T1)
C:\Users\Friend\Anaconda3\lib\site-packages\sklearn\calibration.py:447: RuntimeWarning: invalid value e
ncountered in multiply
  TEP_minus_T1P = P * (T * E - T1)
C:\Users\Friend\Anaconda3\lib\site-packages\sklearn\calibration.py:447: RuntimeWarning: invalid value e
ncountered in multiply
  TEP_minus_T1P = P * (T * E - T1)
C:\Users\Friend\Anaconda3\lib\site-packages\sklearn\calibration.py:447: RuntimeWarning: invalid value e
ncountered in multiply
  TEP_minus_T1P = P * (T * E - T1)
C:\Users\Friend\Anaconda3\lib\site-packages\sklearn\linear_model\stochastic_gradient.py:128: FutureWarn
ing: max_iter and tol parameters have been added in <class 'sklearn.linear_model.stochastic_gradient.SG
DClassifier'> in 0.19. If both are left unset, they default to max_iter=5 and tol=None. If tol is not N
one, max_iter defaults to max_iter=1000. From 0.21, default max_iter will be 1000, and default tol will
be 1e-3.
  "and default tol will be 1e-3." % type(self), FutureWarning)
C:\Users\Friend\Anaconda3\lib\site-packages\sklearn\calibration.py:447: RuntimeWarning: invalid value e
ncountered in multiply
  TEP_minus_T1P = P * (T * E - T1)
C:\Users\Friend\Anaconda3\lib\site-packages\sklearn\calibration.py:447: RuntimeWarning: invalid value e
ncountered in multiply
  TEP_minus_T1P = P * (T * E - T1)
C:\Users\Friend\Anaconda3\lib\site-packages\sklearn\calibration.py:447: RuntimeWarning: invalid value e
ncountered in multiply
  TEP_minus_T1P = P * (T * E - T1)
C:\Users\Friend\Anaconda3\lib\site-packages\sklearn\calibration.py:447: RuntimeWarning: invalid value e
ncountered in multiply
  TEP_minus_T1P = P * (T * E - T1)
C:\Users\Friend\Anaconda3\lib\site-packages\sklearn\calibration.py:447: RuntimeWarning: invalid value e
ncountered in multiply
  TEP_minus_T1P = P * (T * E - T1)
C:\Users\Friend\Anaconda3\lib\site-packages\sklearn\calibration.py:447: RuntimeWarning: invalid value e
ncountered in multiply
  TEP_minus_T1P = P * (T * E - T1)
C:\Users\Friend\Anaconda3\lib\site-packages\sklearn\calibration.py:447: RuntimeWarning: invalid value e
ncountered in multiply
  TEP_minus_T1P = P * (T * E - T1)
C:\Users\Friend\Anaconda3\lib\site-packages\sklearn\calibration.py:447: RuntimeWarning: invalid value e
ncountered in multiply
  TEP_minus_T1P = P * (T * E - T1)
C:\Users\Friend\Anaconda3\lib\site-packages\sklearn\calibration.py:447: RuntimeWarning: invalid value e
ncountered in multiply
  TEP_minus_T1P = P * (T * E - T1)
```

```
Log Loss : 1.5552440345515521
for alpha = 1e-05
```

```
C:\Users\Friend\Anaconda3\lib\site-packages\sklearn\linear_model\stochastic_gradient.py:128: FutureWarn
ing: max_iter and tol parameters have been added in <class 'sklearn.linear_model.stochastic_gradient.SG
DClassifier'> in 0.19. If both are left unset, they default to max_iter=5 and tol=None. If tol is not N
one, max_iter defaults to max_iter=1000. From 0.21, default max_iter will be 1000, and default tol will
be 1e-3.
  "and default tol will be 1e-3." % type(self), FutureWarning)
C:\Users\Friend\Anaconda3\lib\site-packages\sklearn\linear_model\stochastic_gradient.py:128: FutureWarn
ing: max_iter and tol parameters have been added in <class 'sklearn.linear_model.stochastic_gradient.SG
DClassifier'> in 0.19. If both are left unset, they default to max_iter=5 and tol=None. If tol is not N
one, max_iter defaults to max_iter=1000. From 0.21, default max_iter will be 1000, and default tol will
be 1e-3.
  "and default tol will be 1e-3." % type(self), FutureWarning)
C:\Users\Friend\Anaconda3\lib\site-packages\sklearn\calibration.py:447: RuntimeWarning: invalid value e
ncountered in multiply
  TEP_minus_T1P = P * (T * E - T1)
```

```
C:\Users\Friend\Anaconda3\lib\site-packages\sklearn\calibration.py:447: RuntimeWarning: invalid value e
ncountered in multiply
  TEP_minus_T1P = P * (T * E - T1)
C:\Users\Friend\Anaconda3\lib\site-packages\sklearn\calibration.py:447: RuntimeWarning: invalid value e
ncountered in multiply
  TEP_minus_T1P = P * (T * E - T1)
C:\Users\Friend\Anaconda3\lib\site-packages\sklearn\calibration.py:447: RuntimeWarning: invalid value e
ncountered in multiply
  TEP_minus_T1P = P * (T * E - T1)
C:\Users\Friend\Anaconda3\lib\site-packages\sklearn\calibration.py:447: RuntimeWarning: invalid value e
ncountered in multiply
  TEP_minus_T1P = P * (T * E - T1)
C:\Users\Friend\Anaconda3\lib\site-packages\sklearn\calibration.py:447: RuntimeWarning: invalid value e
ncountered in multiply
  TEP_minus_T1P = P * (T * E - T1)
C:\Users\Friend\Anaconda3\lib\site-packages\sklearn\calibration.py:447: RuntimeWarning: invalid value e
ncountered in multiply
  TEP_minus_T1P = P * (T * E - T1)
C:\Users\Friend\Anaconda3\lib\site-packages\sklearn\calibration.py:447: RuntimeWarning: invalid value e
ncountered in multiply
  TEP_minus_T1P = P * (T * E - T1)
C:\Users\Friend\Anaconda3\lib\site-packages\sklearn\calibration.py:447: RuntimeWarning: invalid value e
ncountered in multiply
  TEP_minus_T1P = P * (T * E - T1)
C:\Users\Friend\Anaconda3\lib\site-packages\sklearn\linear_model\stochastic_gradient.py:128: FutureWarn
ing: max_iter and tol parameters have been added in <class 'sklearn.linear_model.stochastic_gradient.SG
DClassifier'> in 0.19. If both are left unset, they default to max_iter=5 and tol=None. If tol is not N
one, max_iter defaults to max_iter=1000. From 0.21, default max_iter will be 1000, and default tol will
be 1e-3.
  "and default tol will be 1e-3." % type(self), FutureWarning)
C:\Users\Friend\Anaconda3\lib\site-packages\sklearn\calibration.py:447: RuntimeWarning: invalid value e
ncountered in multiply
  TEP_minus_T1P = P * (T * E - T1)
C:\Users\Friend\Anaconda3\lib\site-packages\sklearn\calibration.py:447: RuntimeWarning: invalid value e
ncountered in multiply
  TEP_minus_T1P = P * (T * E - T1)
C:\Users\Friend\Anaconda3\lib\site-packages\sklearn\calibration.py:447: RuntimeWarning: invalid value e
ncountered in multiply
  TEP_minus_T1P = P * (T * E - T1)
C:\Users\Friend\Anaconda3\lib\site-packages\sklearn\calibration.py:447: RuntimeWarning: invalid value e
ncountered in multiply
  TEP_minus_T1P = P * (T * E - T1)
C:\Users\Friend\Anaconda3\lib\site-packages\sklearn\calibration.py:447: RuntimeWarning: invalid value e
ncountered in multiply
  TEP_minus_T1P = P * (T * E - T1)
C:\Users\Friend\Anaconda3\lib\site-packages\sklearn\calibration.py:447: RuntimeWarning: invalid value e
ncountered in multiply
  TEP_minus_T1P = P * (T * E - T1)
C:\Users\Friend\Anaconda3\lib\site-packages\sklearn\calibration.py:447: RuntimeWarning: invalid value e
ncountered in multiply
  TEP_minus_T1P = P * (T * E - T1)
C:\Users\Friend\Anaconda3\lib\site-packages\sklearn\calibration.py:447: RuntimeWarning: invalid value e
ncountered in multiply
  TEP_minus_T1P = P * (T * E - T1)
C:\Users\Friend\Anaconda3\lib\site-packages\sklearn\calibration.py:447: RuntimeWarning: invalid value e
ncountered in multiply
  TEP_minus_T1P = P * (T * E - T1)
C:\Users\Friend\Anaconda3\lib\site-packages\sklearn\linear_model\stochastic_gradient.py:128: FutureWarn
ing: max_iter and tol parameters have been added in <class 'sklearn.linear_model.stochastic_gradient.SG
DClassifier'> in 0.19. If both are left unset, they default to max_iter=5 and tol=None. If tol is not N
one, max_iter defaults to max_iter=1000. From 0.21, default max_iter will be 1000, and default tol will
be 1e-3.
  "and default tol will be 1e-3." % type(self), FutureWarning)
C:\Users\Friend\Anaconda3\lib\site-packages\sklearn\calibration.py:447: RuntimeWarning: invalid value e
ncountered in multiply
  TEP_minus_T1P = P * (T * E - T1)
C:\Users\Friend\Anaconda3\lib\site-packages\sklearn\calibration.py:447: RuntimeWarning: invalid value e
ncountered in multiply
  TEP_minus_T1P = P * (T * E - T1)
C:\Users\Friend\Anaconda3\lib\site-packages\sklearn\calibration.py:447: RuntimeWarning: invalid value e
ncountered in multiply
  TEP_minus_T1P = P * (T * E - T1)
C:\Users\Friend\Anaconda3\lib\site-packages\sklearn\calibration.py:447: RuntimeWarning: invalid value e
ncountered in multiply
  TEP_minus_T1P = P * (T * E - T1)
C:\Users\Friend\Anaconda3\lib\site-packages\sklearn\calibration.py:447: RuntimeWarning: invalid value e
ncountered in multiply
```

```
  TEP_minus_T1P = P * (T * E - T1)
C:\Users\Friend\Anaconda3\lib\site-packages\sklearn\calibration.py:447: RuntimeWarning: invalid value e
ncountered in multiply
  TEP_minus_T1P = P * (T * E - T1)
C:\Users\Friend\Anaconda3\lib\site-packages\sklearn\calibration.py:447: RuntimeWarning: invalid value e
ncountered in multiply
  TEP_minus_T1P = P * (T * E - T1)
C:\Users\Friend\Anaconda3\lib\site-packages\sklearn\calibration.py:447: RuntimeWarning: invalid value e
ncountered in multiply
  TEP_minus_T1P = P * (T * E - T1)
C:\Users\Friend\Anaconda3\lib\site-packages\sklearn\calibration.py:447: RuntimeWarning: invalid value e
ncountered in multiply
  TEP_minus_T1P = P * (T * E - T1)
```

```
Log Loss : 1.5622770746623058
for alpha = 0.0001
```

```
C:\Users\Friend\Anaconda3\lib\site-packages\sklearn\linear_model\stochastic_gradient.py:128: FutureWarn
ing: max_iter and tol parameters have been added in <class 'sklearn.linear_model.stochastic_gradient.SG
DClassifier'> in 0.19. If both are left unset, they default to max_iter=5 and tol=None. If tol is not N
one, max_iter defaults to max_iter=1000. From 0.21, default max_iter will be 1000, and default tol will
be 1e-3.
  "and default tol will be 1e-3." % type(self), FutureWarning)
C:\Users\Friend\Anaconda3\lib\site-packages\sklearn\linear_model\stochastic_gradient.py:128: FutureWarn
ing: max_iter and tol parameters have been added in <class 'sklearn.linear_model.stochastic_gradient.SG
DClassifier'> in 0.19. If both are left unset, they default to max_iter=5 and tol=None. If tol is not N
one, max_iter defaults to max_iter=1000. From 0.21, default max_iter will be 1000, and default tol will
be 1e-3.
  "and default tol will be 1e-3." % type(self), FutureWarning)
C:\Users\Friend\Anaconda3\lib\site-packages\sklearn\calibration.py:447: RuntimeWarning: invalid value e
ncountered in multiply
  TEP_minus_T1P = P * (T * E - T1)
C:\Users\Friend\Anaconda3\lib\site-packages\sklearn\calibration.py:447: RuntimeWarning: invalid value e
ncountered in multiply
  TEP_minus_T1P = P * (T * E - T1)
C:\Users\Friend\Anaconda3\lib\site-packages\sklearn\calibration.py:447: RuntimeWarning: invalid value e
ncountered in multiply
  TEP_minus_T1P = P * (T * E - T1)
C:\Users\Friend\Anaconda3\lib\site-packages\sklearn\calibration.py:447: RuntimeWarning: invalid value e
ncountered in multiply
  TEP_minus_T1P = P * (T * E - T1)
C:\Users\Friend\Anaconda3\lib\site-packages\sklearn\calibration.py:447: RuntimeWarning: invalid value e
ncountered in multiply
  TEP_minus_T1P = P * (T * E - T1)
C:\Users\Friend\Anaconda3\lib\site-packages\sklearn\calibration.py:447: RuntimeWarning: invalid value e
ncountered in multiply
  TEP_minus_T1P = P * (T * E - T1)
C:\Users\Friend\Anaconda3\lib\site-packages\sklearn\calibration.py:447: RuntimeWarning: invalid value e
ncountered in multiply
  TEP_minus_T1P = P * (T * E - T1)
C:\Users\Friend\Anaconda3\lib\site-packages\sklearn\calibration.py:447: RuntimeWarning: invalid value e
ncountered in multiply
  TEP_minus_T1P = P * (T * E - T1)
C:\Users\Friend\Anaconda3\lib\site-packages\sklearn\calibration.py:447: RuntimeWarning: invalid value e
ncountered in multiply
  TEP_minus_T1P = P * (T * E - T1)
C:\Users\Friend\Anaconda3\lib\site-packages\sklearn\linear_model\stochastic_gradient.py:128: FutureWarn
ing: max_iter and tol parameters have been added in <class 'sklearn.linear_model.stochastic_gradient.SG
DClassifier'> in 0.19. If both are left unset, they default to max_iter=5 and tol=None. If tol is not N
one, max_iter defaults to max_iter=1000. From 0.21, default max_iter will be 1000, and default tol will
be 1e-3.
  "and default tol will be 1e-3." % type(self), FutureWarning)
C:\Users\Friend\Anaconda3\lib\site-packages\sklearn\calibration.py:447: RuntimeWarning: invalid value e
ncountered in multiply
  TEP_minus_T1P = P * (T * E - T1)
C:\Users\Friend\Anaconda3\lib\site-packages\sklearn\calibration.py:447: RuntimeWarning: invalid value e
ncountered in multiply
  TEP_minus_T1P = P * (T * E - T1)
C:\Users\Friend\Anaconda3\lib\site-packages\sklearn\calibration.py:447: RuntimeWarning: invalid value e
ncountered in multiply
  TEP_minus_T1P = P * (T * E - T1)
C:\Users\Friend\Anaconda3\lib\site-packages\sklearn\calibration.py:447: RuntimeWarning: invalid value e
ncountered in multiply
  TEP_minus_T1P = P * (T * E - T1)
C:\Users\Friend\Anaconda3\lib\site-packages\sklearn\calibration.py:447: RuntimeWarning: invalid value e
ncountered in multiply
```

```
ncountered in multiply
  TEP_minus_T1P = P * (T * E - T1)
C:\Users\Friend\Anaconda3\lib\site-packages\sklearn\calibration.py:447: RuntimeWarning: invalid value e
ncountered in multiply
  TEP_minus_T1P = P * (T * E - T1)
C:\Users\Friend\Anaconda3\lib\site-packages\sklearn\calibration.py:447: RuntimeWarning: invalid value e
ncountered in multiply
  TEP_minus_T1P = P * (T * E - T1)
C:\Users\Friend\Anaconda3\lib\site-packages\sklearn\calibration.py:447: RuntimeWarning: invalid value e
ncountered in multiply
  TEP_minus_T1P = P * (T * E - T1)
C:\Users\Friend\Anaconda3\lib\site-packages\sklearn\calibration.py:447: RuntimeWarning: invalid value e
ncountered in multiply
  TEP_minus_T1P = P * (T * E - T1)
C:\Users\Friend\Anaconda3\lib\site-packages\sklearn\linear_model\stochastic_gradient.py:128: FutureWarn
ing: max_iter and tol parameters have been added in <class 'sklearn.linear_model.stochastic_gradient.SG
DClassifier'> in 0.19. If both are left unset, they default to max_iter=5 and tol=None. If tol is not N
one, max_iter defaults to max_iter=1000. From 0.21, default max_iter will be 1000, and default tol will
be 1e-3.
  "and default tol will be 1e-3." % type(self), FutureWarning)
C:\Users\Friend\Anaconda3\lib\site-packages\sklearn\calibration.py:447: RuntimeWarning: invalid value e
ncountered in multiply
  TEP_minus_T1P = P * (T * E - T1)
C:\Users\Friend\Anaconda3\lib\site-packages\sklearn\calibration.py:447: RuntimeWarning: invalid value e
ncountered in multiply
  TEP_minus_T1P = P * (T * E - T1)
C:\Users\Friend\Anaconda3\lib\site-packages\sklearn\calibration.py:447: RuntimeWarning: invalid value e
ncountered in multiply
  TEP_minus_T1P = P * (T * E - T1)
C:\Users\Friend\Anaconda3\lib\site-packages\sklearn\calibration.py:447: RuntimeWarning: invalid value e
ncountered in multiply
  TEP_minus_T1P = P * (T * E - T1)
C:\Users\Friend\Anaconda3\lib\site-packages\sklearn\calibration.py:447: RuntimeWarning: invalid value e
ncountered in multiply
  TEP_minus_T1P = P * (T * E - T1)
C:\Users\Friend\Anaconda3\lib\site-packages\sklearn\calibration.py:447: RuntimeWarning: invalid value e
ncountered in multiply
  TEP_minus_T1P = P * (T * E - T1)
C:\Users\Friend\Anaconda3\lib\site-packages\sklearn\calibration.py:447: RuntimeWarning: invalid value e
ncountered in multiply
  TEP_minus_T1P = P * (T * E - T1)
C:\Users\Friend\Anaconda3\lib\site-packages\sklearn\calibration.py:447: RuntimeWarning: invalid value e
ncountered in multiply
  TEP_minus_T1P = P * (T * E - T1)
C:\Users\Friend\Anaconda3\lib\site-packages\sklearn\calibration.py:447: RuntimeWarning: invalid value e
ncountered in multiply
  TEP_minus_T1P = P * (T * E - T1)
```

```
Log Loss : 1.5662121879607596
for alpha = 0.001
```

```
C:\Users\Friend\Anaconda3\lib\site-packages\sklearn\linear_model\stochastic_gradient.py:128: FutureWarn
ing: max_iter and tol parameters have been added in <class 'sklearn.linear_model.stochastic_gradient.SG
DClassifier'> in 0.19. If both are left unset, they default to max_iter=5 and tol=None. If tol is not N
one, max_iter defaults to max_iter=1000. From 0.21, default max_iter will be 1000, and default tol will
be 1e-3.
  "and default tol will be 1e-3." % type(self), FutureWarning)
C:\Users\Friend\Anaconda3\lib\site-packages\sklearn\linear_model\stochastic_gradient.py:128: FutureWarn
ing: max_iter and tol parameters have been added in <class 'sklearn.linear_model.stochastic_gradient.SG
DClassifier'> in 0.19. If both are left unset, they default to max_iter=5 and tol=None. If tol is not N
one, max_iter defaults to max_iter=1000. From 0.21, default max_iter will be 1000, and default tol will
be 1e-3.
  "and default tol will be 1e-3." % type(self), FutureWarning)
C:\Users\Friend\Anaconda3\lib\site-packages\sklearn\linear_model\stochastic_gradient.py:128: FutureWarn
ing: max_iter and tol parameters have been added in <class 'sklearn.linear_model.stochastic_gradient.SG
DClassifier'> in 0.19. If both are left unset, they default to max_iter=5 and tol=None. If tol is not N
one, max_iter defaults to max_iter=1000. From 0.21, default max_iter will be 1000, and default tol will
be 1e-3.
  "and default tol will be 1e-3." % type(self), FutureWarning)
C:\Users\Friend\Anaconda3\lib\site-packages\sklearn\linear_model\stochastic_gradient.py:128: FutureWarn
ing: max_iter and tol parameters have been added in <class 'sklearn.linear_model.stochastic_gradient.SG
DClassifier'> in 0.19. If both are left unset, they default to max_iter=5 and tol=None. If tol is not N
one, max_iter defaults to max_iter=1000. From 0.21, default max_iter will be 1000, and default tol will
be 1e-3.
  "and default tol will be 1e-3." % type(self), FutureWarning)
```

```
Log Loss : 1.4959101630570983
for alpha = 0.01
```

```
C:\Users\Friend\Anaconda3\lib\site-packages\sklearn\linear_model\stochastic_gradient.py:128: FutureWarn
ing: max_iter and tol parameters have been added in <class 'sklearn.linear_model.stochastic_gradient.SG
DClassifier'> in 0.19. If both are left unset, they default to max_iter=5 and tol=None. If tol is not N
one, max_iter defaults to max_iter=1000. From 0.21, default max_iter will be 1000, and default tol will
be 1e-3.
  "and default tol will be 1e-3." % type(self), FutureWarning)
C:\Users\Friend\Anaconda3\lib\site-packages\sklearn\linear_model\stochastic_gradient.py:128: FutureWarn
ing: max_iter and tol parameters have been added in <class 'sklearn.linear_model.stochastic_gradient.SG
DClassifier'> in 0.19. If both are left unset, they default to max_iter=5 and tol=None. If tol is not N
one, max_iter defaults to max_iter=1000. From 0.21, default max_iter will be 1000, and default tol will
be 1e-3.
  "and default tol will be 1e-3." % type(self), FutureWarning)
C:\Users\Friend\Anaconda3\lib\site-packages\sklearn\linear_model\stochastic_gradient.py:128: FutureWarn
ing: max_iter and tol parameters have been added in <class 'sklearn.linear_model.stochastic_gradient.SG
DClassifier'> in 0.19. If both are left unset, they default to max_iter=5 and tol=None. If tol is not N
one, max_iter defaults to max_iter=1000. From 0.21, default max_iter will be 1000, and default tol will
be 1e-3.
  "and default tol will be 1e-3." % type(self), FutureWarning)
```

```
Log Loss : 1.2185638735846782
for alpha = 0.1
```

```
C:\Users\Friend\Anaconda3\lib\site-packages\sklearn\linear_model\stochastic_gradient.py:128: FutureWarn
ing: max_iter and tol parameters have been added in <class 'sklearn.linear_model.stochastic_gradient.SG
DClassifier'> in 0.19. If both are left unset, they default to max_iter=5 and tol=None. If tol is not N
one, max_iter defaults to max_iter=1000. From 0.21, default max_iter will be 1000, and default tol will
be 1e-3.
  "and default tol will be 1e-3." % type(self), FutureWarning)
C:\Users\Friend\Anaconda3\lib\site-packages\sklearn\linear_model\stochastic_gradient.py:128: FutureWarn
ing: max_iter and tol parameters have been added in <class 'sklearn.linear_model.stochastic_gradient.SG
DClassifier'> in 0.19. If both are left unset, they default to max_iter=5 and tol=None. If tol is not N
one, max_iter defaults to max_iter=1000. From 0.21, default max_iter will be 1000, and default tol will
be 1e-3.
  "and default tol will be 1e-3." % type(self), FutureWarning)
C:\Users\Friend\Anaconda3\lib\site-packages\sklearn\linear_model\stochastic_gradient.py:128: FutureWarn
ing: max_iter and tol parameters have been added in <class 'sklearn.linear_model.stochastic_gradient.SG
DClassifier'> in 0.19. If both are left unset, they default to max_iter=5 and tol=None. If tol is not N
one, max_iter defaults to max_iter=1000. From 0.21, default max_iter will be 1000, and default tol will
be 1e-3.
  "and default tol will be 1e-3." % type(self), FutureWarning)
```

```
Log Loss : 1.2247297297101698
for alpha = 1
```

```
C:\Users\Friend\Anaconda3\lib\site-packages\sklearn\linear_model\stochastic_gradient.py:128: FutureWarn
ing: max_iter and tol parameters have been added in <class 'sklearn.linear_model.stochastic_gradient.SG
DClassifier'> in 0.19. If both are left unset, they default to max_iter=5 and tol=None. If tol is not N
one, max_iter defaults to max_iter=1000. From 0.21, default max_iter will be 1000, and default tol will
be 1e-3.
  "and default tol will be 1e-3." % type(self), FutureWarning)
C:\Users\Friend\Anaconda3\lib\site-packages\sklearn\linear_model\stochastic_gradient.py:128: FutureWarn
ing: max_iter and tol parameters have been added in <class 'sklearn.linear_model.stochastic_gradient.SG
DClassifier'> in 0.19. If both are left unset, they default to max_iter=5 and tol=None. If tol is not N
one, max_iter defaults to max_iter=1000. From 0.21, default max_iter will be 1000, and default tol will
be 1e-3.
  "and default tol will be 1e-3." % type(self), FutureWarning)
C:\Users\Friend\Anaconda3\lib\site-packages\sklearn\linear_model\stochastic_gradient.py:128: FutureWarn
ing: max_iter and tol parameters have been added in <class 'sklearn.linear_model.stochastic_gradient.SG
DClassifier'> in 0.19. If both are left unset, they default to max_iter=5 and tol=None. If tol is not N
one, max_iter defaults to max_iter=1000. From 0.21, default max_iter will be 1000, and default tol will
be 1e-3.
  "and default tol will be 1e-3." % type(self), FutureWarning)
```

```
Log Loss : 1.3990201740433128
for alpha = 10
```

```
C:\Users\Friend\Anaconda3\lib\site-packages\sklearn\linear_model\stochastic_gradient.py:128: FutureWarn
ing: max_iter and tol parameters have been added in <class 'sklearn.linear_model.stochastic_gradient.SG
DClassifier'> in 0.19. If both are left unset, they default to max_iter=5 and tol=None. If tol is not N
one, max_iter defaults to max_iter=1000. From 0.21, default max_iter will be 1000, and default tol will
```

```
one, max_iter defaults to max_iter=1000. From 0.21, default max_iter will be 1000, and default tol will
be 1e-3.
  "and default tol will be 1e-3." % type(self), FutureWarning)
C:\Users\Friend\Anaconda3\lib\site-packages\sklearn\linear_model\stochastic_gradient.py:128: FutureWarn
ing: max_iter and tol parameters have been added in <class 'sklearn.linear_model.stochastic_gradient.SG
DClassifier'> in 0.19. If both are left unset, they default to max_iter=5 and tol=None. If tol is not N
one, max_iter defaults to max_iter=1000. From 0.21, default max_iter will be 1000, and default tol will
be 1e-3.
  "and default tol will be 1e-3." % type(self), FutureWarning)
C:\Users\Friend\Anaconda3\lib\site-packages\sklearn\linear_model\stochastic_gradient.py:128: FutureWarn
ing: max_iter and tol parameters have been added in <class 'sklearn.linear_model.stochastic_gradient.SG
DClassifier'> in 0.19. If both are left unset, they default to max_iter=5 and tol=None. If tol is not N
one, max_iter defaults to max_iter=1000. From 0.21, default max_iter will be 1000, and default tol will
be 1e-3.
  "and default tol will be 1e-3." % type(self), FutureWarning)
```

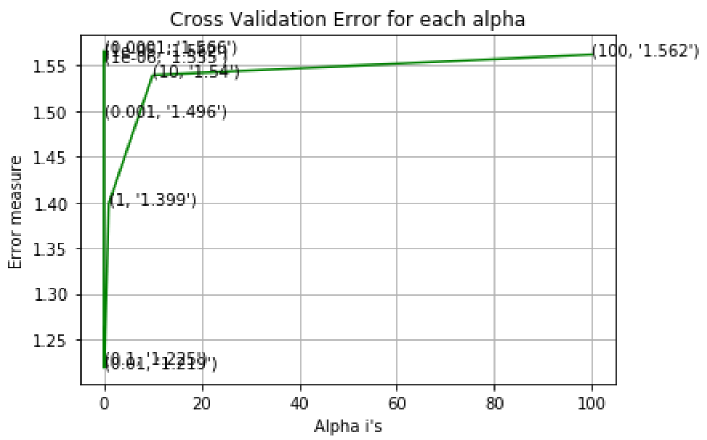
```
Log Loss : 1.5395037163602163
for alpha = 100
```

```
C:\Users\Friend\Anaconda3\lib\site-packages\sklearn\linear_model\stochastic_gradient.py:128: FutureWarn
ing: max_iter and tol parameters have been added in <class 'sklearn.linear_model.stochastic_gradient.SG
DClassifier'> in 0.19. If both are left unset, they default to max_iter=5 and tol=None. If tol is not N
one, max_iter defaults to max_iter=1000. From 0.21, default max_iter will be 1000, and default tol will
be 1e-3.
  "and default tol will be 1e-3." % type(self), FutureWarning)
C:\Users\Friend\Anaconda3\lib\site-packages\sklearn\linear_model\stochastic_gradient.py:128: FutureWarn
ing: max_iter and tol parameters have been added in <class 'sklearn.linear_model.stochastic_gradient.SG
DClassifier'> in 0.19. If both are left unset, they default to max_iter=5 and tol=None. If tol is not N
one, max_iter defaults to max_iter=1000. From 0.21, default max_iter will be 1000, and default tol will
be 1e-3.
  "and default tol will be 1e-3." % type(self), FutureWarning)
C:\Users\Friend\Anaconda3\lib\site-packages\sklearn\linear_model\stochastic_gradient.py:128: FutureWarn
ing: max_iter and tol parameters have been added in <class 'sklearn.linear_model.stochastic_gradient.SG
DClassifier'> in 0.19. If both are left unset, they default to max_iter=5 and tol=None. If tol is not N
one, max_iter defaults to max_iter=1000. From 0.21, default max_iter will be 1000, and default tol will
be 1e-3.
  "and default tol will be 1e-3." % type(self), FutureWarning)
```

```
Log Loss : 1.562023204363665
```

In [70]:

```python
fig, ax = plt.subplots()
ax.plot(alpha, cv_log_error_array,c='g')
for i, txt in enumerate(np.round(cv_log_error_array,3)):
    ax.annotate((alpha[i],str(txt)), (alpha[i],cv_log_error_array[i]))
plt.grid()
plt.title("Cross Validation Error for each alpha")
plt.xlabel("Alpha i's")
plt.ylabel("Error measure")
plt.show()
```



In [71]:

```python
best_alpha = np.argmin(cv_log_error_array)
clf = SGDClassifier(class_weight='balanced', alpha=alpha[best_alpha], penalty='l2', loss='log', random_
state=42)
```

```
state=42)
clf.fit(train_x_onehotCoding, train_y)
sig_clf = CalibratedClassifierCV(clf, method="sigmoid")
sig_clf.fit(train_x_onehotCoding, train_y)

predict_y = sig_clf.predict_proba(train_x_onehotCoding)
log_train_log_loss = log_loss(y_train, predict_y, labels=clf.classes_, eps=1e-15)
print('For values of best alpha = ', alpha[best_alpha], "The train log loss is:",log_train_log_loss)
predict_y = sig_clf.predict_proba(cv_x_onehotCoding)
log_cv_log_loss = log_loss(y_cv, predict_y, labels=clf.classes_, eps=1e-15)
print('For values of best alpha = ', alpha[best_alpha], "The cross validation log loss is:",log_cv_log_
loss)
predict_y = sig_clf.predict_proba(test_x_onehotCoding)
log_test_log_loss = log_loss(y_test, predict_y, labels=clf.classes_, eps=1e-15)
print('For values of best alpha = ', alpha[best_alpha], "The test log loss is:",log_test_log_loss)
```

```
C:\Users\Friend\Anaconda3\lib\site-packages\sklearn\linear_model\stochastic_gradient.py:128: FutureWarn
ing: max_iter and tol parameters have been added in <class 'sklearn.linear_model.stochastic_gradient.SG
DClassifier'> in 0.19. If both are left unset, they default to max_iter=5 and tol=None. If tol is not N
one, max_iter defaults to max_iter=1000. From 0.21, default max_iter will be 1000, and default tol will
be 1e-3.
  "and default tol will be 1e-3." % type(self), FutureWarning)
C:\Users\Friend\Anaconda3\lib\site-packages\sklearn\linear_model\stochastic_gradient.py:128: FutureWarn
ing: max_iter and tol parameters have been added in <class 'sklearn.linear_model.stochastic_gradient.SG
DClassifier'> in 0.19. If both are left unset, they default to max_iter=5 and tol=None. If tol is not N
one, max_iter defaults to max_iter=1000. From 0.21, default max_iter will be 1000, and default tol will
be 1e-3.
  "and default tol will be 1e-3." % type(self), FutureWarning)
C:\Users\Friend\Anaconda3\lib\site-packages\sklearn\linear_model\stochastic_gradient.py:128: FutureWarn
ing: max_iter and tol parameters have been added in <class 'sklearn.linear_model.stochastic_gradient.SG
DClassifier'> in 0.19. If both are left unset, they default to max_iter=5 and tol=None. If tol is not N
one, max_iter defaults to max_iter=1000. From 0.21, default max_iter will be 1000, and default tol will
be 1e-3.
  "and default tol will be 1e-3." % type(self), FutureWarning)
C:\Users\Friend\Anaconda3\lib\site-packages\sklearn\linear_model\stochastic_gradient.py:128: FutureWarn
ing: max_iter and tol parameters have been added in <class 'sklearn.linear_model.stochastic_gradient.SG
DClassifier'> in 0.19. If both are left unset, they default to max_iter=5 and tol=None. If tol is not N
one, max_iter defaults to max_iter=1000. From 0.21, default max_iter will be 1000, and default tol will
be 1e-3.
  "and default tol will be 1e-3." % type(self), FutureWarning)
```

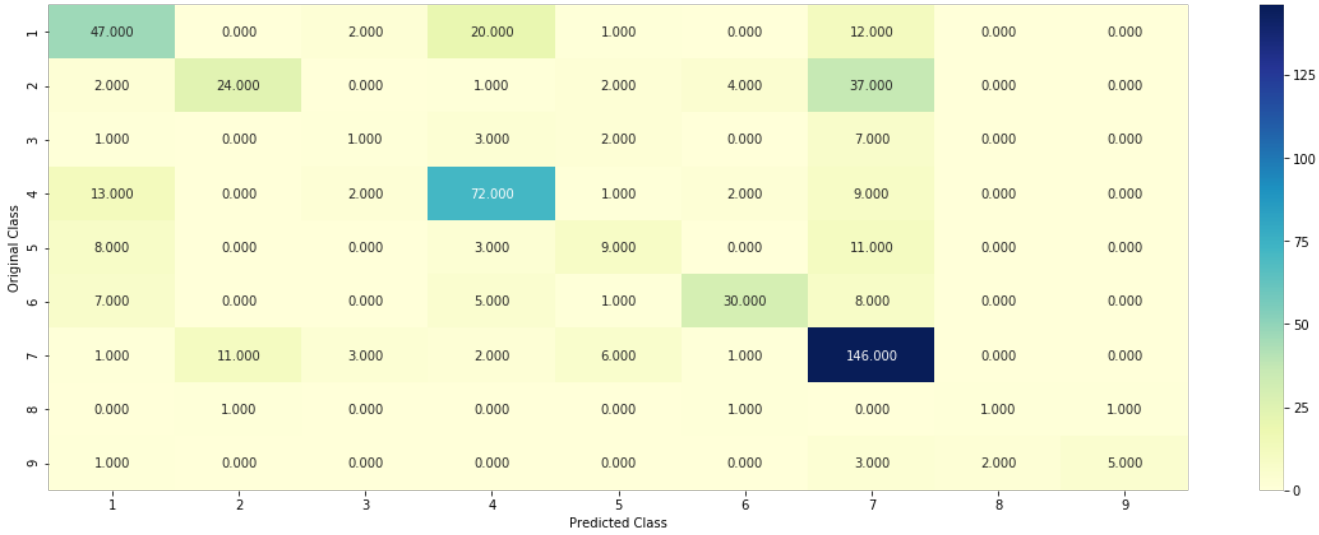
```
For values of best alpha =  0.01 The train log loss is: 0.8522449793595871
For values of best alpha =  0.01 The cross validation log loss is: 1.2185638735846782
For values of best alpha =  0.01 The test log loss is: 1.1701141141449427
```

In [72]:

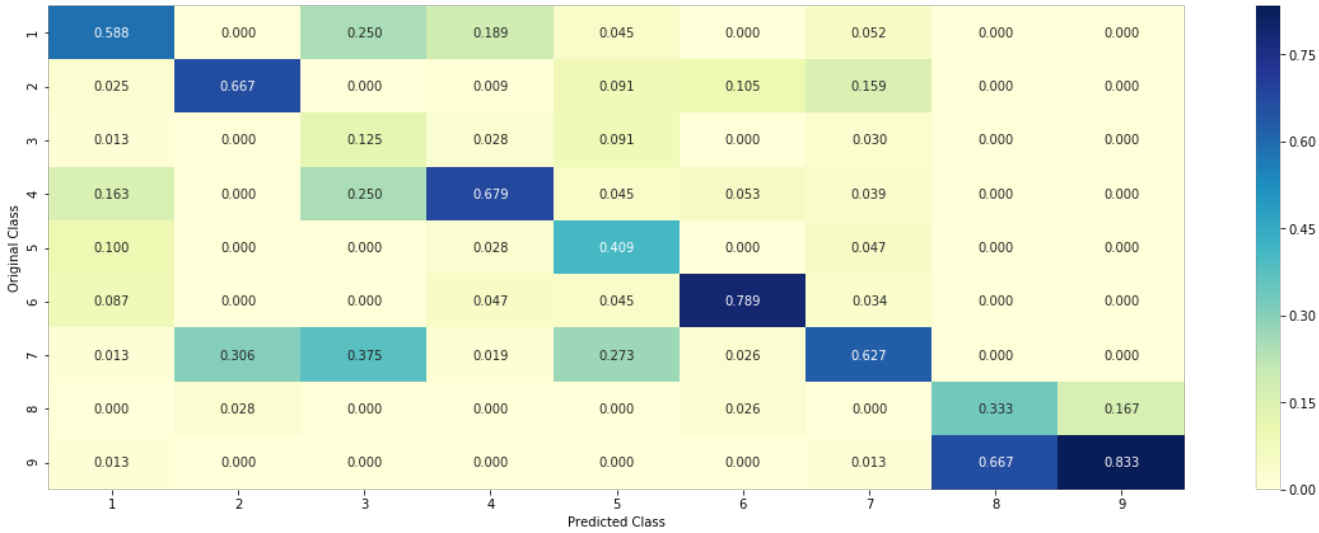
```
clf = SGDClassifier(class_weight='balanced', alpha=alpha[best_alpha], penalty='l2', loss='log', random_
state=42)
predict_and_plot_confusion_matrix(train_x_onehotCoding, train_y, cv_x_onehotCoding, cv_y, clf)
```

```
C:\Users\Friend\Anaconda3\lib\site-packages\sklearn\linear_model\stochastic_gradient.py:128: FutureWarn
ing: max_iter and tol parameters have been added in <class 'sklearn.linear_model.stochastic_gradient.SG
DClassifier'> in 0.19. If both are left unset, they default to max_iter=5 and tol=None. If tol is not N
one, max_iter defaults to max_iter=1000. From 0.21, default max_iter will be 1000, and default tol will
be 1e-3.
  "and default tol will be 1e-3." % type(self), FutureWarning)
C:\Users\Friend\Anaconda3\lib\site-packages\sklearn\linear_model\stochastic_gradient.py:128: FutureWarn
ing: max_iter and tol parameters have been added in <class 'sklearn.linear_model.stochastic_gradient.SG
DClassifier'> in 0.19. If both are left unset, they default to max_iter=5 and tol=None. If tol is not N
one, max_iter defaults to max_iter=1000. From 0.21, default max_iter will be 1000, and default tol will
be 1e-3.
  "and default tol will be 1e-3." % type(self), FutureWarning)
C:\Users\Friend\Anaconda3\lib\site-packages\sklearn\linear_model\stochastic_gradient.py:128: FutureWarn
ing: max_iter and tol parameters have been added in <class 'sklearn.linear_model.stochastic_gradient.SG
DClassifier'> in 0.19. If both are left unset, they default to max_iter=5 and tol=None. If tol is not N
one, max_iter defaults to max_iter=1000. From 0.21, default max_iter will be 1000, and default tol will
be 1e-3.
  "and default tol will be 1e-3." % type(self), FutureWarning)
C:\Users\Friend\Anaconda3\lib\site-packages\sklearn\linear_model\stochastic_gradient.py:128: FutureWarn
ing: max_iter and tol parameters have been added in <class 'sklearn.linear_model.stochastic_gradient.SG
DClassifier'> in 0.19. If both are left unset, they default to max_iter=5 and tol=None. If tol is not N
one, max_iter defaults to max_iter=1000. From 0.21, default max_iter will be 1000, and default tol will
be 1e-3.
  "and default tol will be 1e-3." % type(self), FutureWarning)
```

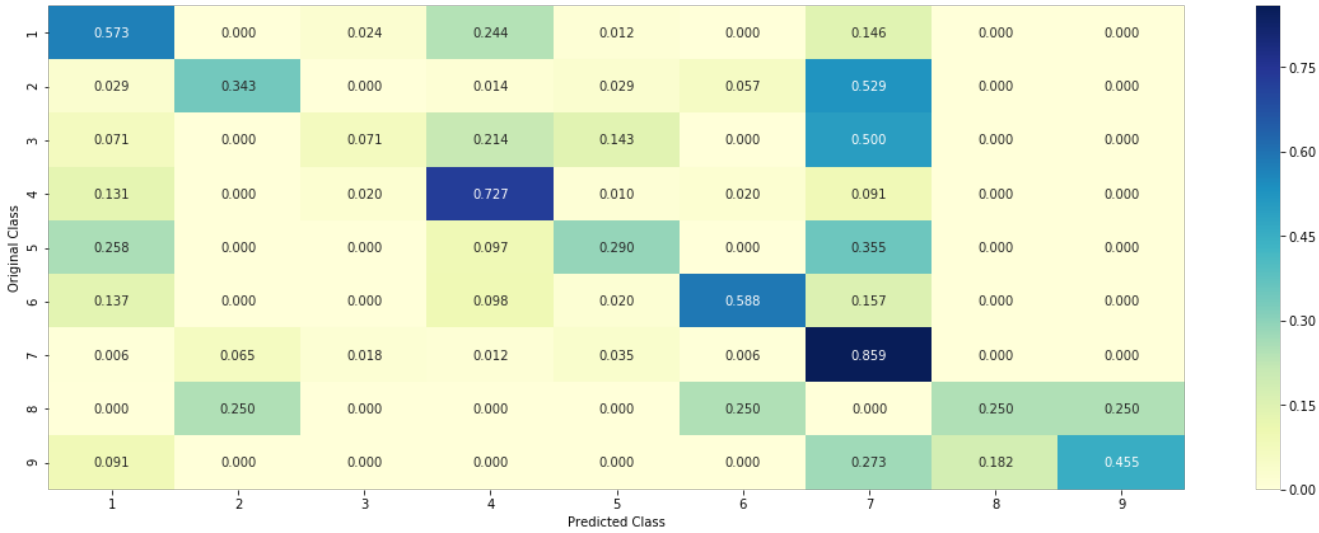
Log loss : 1.2185638735846782
Number of mis-classified points : 0.37030075187969924
-------------------- Confusion matrix --------------------



-------------------- Precision matrix (Columm Sum=1) --------------------



-------------------- Recall matrix (Row sum=1) --------------------



Conclusion

In [73]:

```
from prettytable import PrettyTable

Table = PrettyTable()
Table.field_names  = ["Model", "Train Loss", "Cross Validation Loss", "Test Loss"]
Table.add_row(["Logistic Regression", log_train_log_loss, log_cv_log_loss,log_test_log_loss])
print(Table)
```

```
+---------------------+--------------------+-----------------------+--------------------+
|        Model        |     Train Loss     | Cross Validation Loss |     Test Loss      |
+---------------------+--------------------+-----------------------+--------------------+
| Logistic Regression | 0.8522449793595871 |   1.2185638735846782  | 1.1701141141449427 |
+---------------------+--------------------+-----------------------+--------------------+
```

Logistic Regression(Tf-idf:Uni-gram and Bi-gram)

In [78]:

```
train_x_onehotCoding = hstack((train_gene_feature_onehotCoding, train_variation_feature_onehotCoding,tf
idf_train_text_feature_onehotCoding,bi_tfidf_train_text_feature_onehotCoding))
cv_x_onehotCoding = hstack((cv_gene_feature_onehotCoding, cv_variation_feature_onehotCoding,tfidf_cv_te
xt_feature_onehotCoding, bi_tfidf_cv_text_feature_onehotCoding))
test_x_onehotCoding = hstack((test_gene_feature_onehotCoding, test_variation_feature_onehotCoding,tfidf
_test_text_feature_onehotCoding, bi_tfidf_test_text_feature_onehotCoding))
```

In [80]:

```
print(train_x_onehotCoding.shape,cv_x_onehotCoding.shape,test_x_onehotCoding.shape)
```

```
(2124, 4195) (532, 4195) (665, 4195)
```

In [81]:

```
from sklearn.linear_model import SGDClassifier

alpha = [10 ** x for x in range(-6, 3)]
cv_log_error_array = []
for i in alpha:
    print("for alpha =", i)
    clf = SGDClassifier(class_weight='balanced', alpha=i, penalty='l2', loss='log', random_state=42)
    clf.fit(train_x_onehotCoding, train_y)
    sig_clf = CalibratedClassifierCV(clf, method="sigmoid")
    sig_clf.fit(train_x_onehotCoding, train_y)
    sig_clf_probs = sig_clf.predict_proba(cv_x_onehotCoding)
    cv_log_error_array.append(log_loss(cv_y, sig_clf_probs, labels=clf.classes_, eps=1e-15))
    print("Log Loss :",log_loss(cv_y, sig_clf_probs))
```

```
for alpha = 1e-06
```

```
C:\Users\Friend\Anaconda3\lib\site-packages\sklearn\linear_model\stochastic_gradient.py:128: FutureWarn
ing: max_iter and tol parameters have been added in <class 'sklearn.linear_model.stochastic_gradient.SG
DClassifier'> in 0.19. If both are left unset, they default to max_iter=5 and tol=None. If tol is not N
one, max_iter defaults to max_iter=1000. From 0.21, default max_iter will be 1000, and default tol will
be 1e-3.
  "and default tol will be 1e-3." % type(self), FutureWarning)
C:\Users\Friend\Anaconda3\lib\site-packages\sklearn\linear_model\stochastic_gradient.py:128: FutureWarn
ing: max_iter and tol parameters have been added in <class 'sklearn.linear_model.stochastic_gradient.SG
DClassifier'> in 0.19. If both are left unset, they default to max_iter=5 and tol=None. If tol is not N
one, max_iter defaults to max_iter=1000. From 0.21, default max_iter will be 1000, and default tol will
be 1e-3.
  "and default tol will be 1e-3." % type(self), FutureWarning)
C:\Users\Friend\Anaconda3\lib\site-packages\sklearn\linear_model\stochastic_gradient.py:128: FutureWarn
ing: max_iter and tol parameters have been added in <class 'sklearn.linear_model.stochastic_gradient.SG
DClassifier'> in 0.19. If both are left unset, they default to max_iter=5 and tol=None. If tol is not N
one, max_iter defaults to max_iter=1000. From 0.21, default max_iter will be 1000, and default tol will
be 1e-3.
  "and default tol will be 1e-3." % type(self), FutureWarning)
C:\Users\Friend\Anaconda3\lib\site-packages\sklearn\linear_model\stochastic_gradient.py:128: FutureWarn
```

```
C:\Users\Friend\Anaconda3\lib\site-packages\sklearn\linear_model\stochastic_gradient.py:128: FutureWarn
ing: max_iter and tol parameters have been added in <class 'sklearn.linear_model.stochastic_gradient.SG
DClassifier'> in 0.19. If both are left unset, they default to max_iter=5 and tol=None. If tol is not N
one, max_iter defaults to max_iter=1000. From 0.21, default max_iter will be 1000, and default tol will
be 1e-3.
  "and default tol will be 1e-3." % type(self), FutureWarning)
```

```
Log Loss : 1.1188118321806382
for alpha = 1e-05
```

```
C:\Users\Friend\Anaconda3\lib\site-packages\sklearn\linear_model\stochastic_gradient.py:128: FutureWarn
ing: max_iter and tol parameters have been added in <class 'sklearn.linear_model.stochastic_gradient.SG
DClassifier'> in 0.19. If both are left unset, they default to max_iter=5 and tol=None. If tol is not N
one, max_iter defaults to max_iter=1000. From 0.21, default max_iter will be 1000, and default tol will
be 1e-3.
  "and default tol will be 1e-3." % type(self), FutureWarning)
C:\Users\Friend\Anaconda3\lib\site-packages\sklearn\linear_model\stochastic_gradient.py:128: FutureWarn
ing: max_iter and tol parameters have been added in <class 'sklearn.linear_model.stochastic_gradient.SG
DClassifier'> in 0.19. If both are left unset, they default to max_iter=5 and tol=None. If tol is not N
one, max_iter defaults to max_iter=1000. From 0.21, default max_iter will be 1000, and default tol will
be 1e-3.
  "and default tol will be 1e-3." % type(self), FutureWarning)
C:\Users\Friend\Anaconda3\lib\site-packages\sklearn\linear_model\stochastic_gradient.py:128: FutureWarn
ing: max_iter and tol parameters have been added in <class 'sklearn.linear_model.stochastic_gradient.SG
DClassifier'> in 0.19. If both are left unset, they default to max_iter=5 and tol=None. If tol is not N
one, max_iter defaults to max_iter=1000. From 0.21, default max_iter will be 1000, and default tol will
be 1e-3.
  "and default tol will be 1e-3." % type(self), FutureWarning)
```

```
Log Loss : 1.0847341296981239
for alpha = 0.0001
```

```
C:\Users\Friend\Anaconda3\lib\site-packages\sklearn\linear_model\stochastic_gradient.py:128: FutureWarn
ing: max_iter and tol parameters have been added in <class 'sklearn.linear_model.stochastic_gradient.SG
DClassifier'> in 0.19. If both are left unset, they default to max_iter=5 and tol=None. If tol is not N
one, max_iter defaults to max_iter=1000. From 0.21, default max_iter will be 1000, and default tol will
be 1e-3.
  "and default tol will be 1e-3." % type(self), FutureWarning)
C:\Users\Friend\Anaconda3\lib\site-packages\sklearn\linear_model\stochastic_gradient.py:128: FutureWarn
ing: max_iter and tol parameters have been added in <class 'sklearn.linear_model.stochastic_gradient.SG
DClassifier'> in 0.19. If both are left unset, they default to max_iter=5 and tol=None. If tol is not N
one, max_iter defaults to max_iter=1000. From 0.21, default max_iter will be 1000, and default tol will
be 1e-3.
  "and default tol will be 1e-3." % type(self), FutureWarning)
C:\Users\Friend\Anaconda3\lib\site-packages\sklearn\linear_model\stochastic_gradient.py:128: FutureWarn
ing: max_iter and tol parameters have been added in <class 'sklearn.linear_model.stochastic_gradient.SG
DClassifier'> in 0.19. If both are left unset, they default to max_iter=5 and tol=None. If tol is not N
one, max_iter defaults to max_iter=1000. From 0.21, default max_iter will be 1000, and default tol will
be 1e-3.
  "and default tol will be 1e-3." % type(self), FutureWarning)
```

```
Log Loss : 1.024864596109853
for alpha = 0.001
```

```
C:\Users\Friend\Anaconda3\lib\site-packages\sklearn\linear_model\stochastic_gradient.py:128: FutureWarn
ing: max_iter and tol parameters have been added in <class 'sklearn.linear_model.stochastic_gradient.SG
DClassifier'> in 0.19. If both are left unset, they default to max_iter=5 and tol=None. If tol is not N
one, max_iter defaults to max_iter=1000. From 0.21, default max_iter will be 1000, and default tol will
be 1e-3.
  "and default tol will be 1e-3." % type(self), FutureWarning)
C:\Users\Friend\Anaconda3\lib\site-packages\sklearn\linear_model\stochastic_gradient.py:128: FutureWarn
ing: max_iter and tol parameters have been added in <class 'sklearn.linear_model.stochastic_gradient.SG
DClassifier'> in 0.19. If both are left unset, they default to max_iter=5 and tol=None. If tol is not N
one, max_iter defaults to max_iter=1000. From 0.21, default max_iter will be 1000, and default tol will
be 1e-3.
  "and default tol will be 1e-3." % type(self), FutureWarning)
C:\Users\Friend\Anaconda3\lib\site-packages\sklearn\linear_model\stochastic_gradient.py:128: FutureWarn
ing: max_iter and tol parameters have been added in <class 'sklearn.linear_model.stochastic_gradient.SG
DClassifier'> in 0.19. If both are left unset, they default to max_iter=5 and tol=None. If tol is not N
one, max_iter defaults to max_iter=1000. From 0.21, default max_iter will be 1000, and default tol will
be 1e-3.
  "and default tol will be 1e-3." % type(self), FutureWarning)
```

```
Log Loss : 1.0246178765053338
```

```
Log Loss : 1.0346178765053338
for alpha = 0.01
```

```
C:\Users\Friend\Anaconda3\lib\site-packages\sklearn\linear_model\stochastic_gradient.py:128: FutureWarn
ing: max_iter and tol parameters have been added in <class 'sklearn.linear_model.stochastic_gradient.SG
DClassifier'> in 0.19. If both are left unset, they default to max_iter=5 and tol=None. If tol is not N
one, max_iter defaults to max_iter=1000. From 0.21, default max_iter will be 1000, and default tol will
be 1e-3.
  "and default tol will be 1e-3." % type(self), FutureWarning)
C:\Users\Friend\Anaconda3\lib\site-packages\sklearn\linear_model\stochastic_gradient.py:128: FutureWarn
ing: max_iter and tol parameters have been added in <class 'sklearn.linear_model.stochastic_gradient.SG
DClassifier'> in 0.19. If both are left unset, they default to max_iter=5 and tol=None. If tol is not N
one, max_iter defaults to max_iter=1000. From 0.21, default max_iter will be 1000, and default tol will
be 1e-3.
  "and default tol will be 1e-3." % type(self), FutureWarning)
C:\Users\Friend\Anaconda3\lib\site-packages\sklearn\linear_model\stochastic_gradient.py:128: FutureWarn
ing: max_iter and tol parameters have been added in <class 'sklearn.linear_model.stochastic_gradient.SG
DClassifier'> in 0.19. If both are left unset, they default to max_iter=5 and tol=None. If tol is not N
one, max_iter defaults to max_iter=1000. From 0.21, default max_iter will be 1000, and default tol will
be 1e-3.
  "and default tol will be 1e-3." % type(self), FutureWarning)
```

```
Log Loss : 1.1667313983332197
for alpha = 0.1
```

```
C:\Users\Friend\Anaconda3\lib\site-packages\sklearn\linear_model\stochastic_gradient.py:128: FutureWarn
ing: max_iter and tol parameters have been added in <class 'sklearn.linear_model.stochastic_gradient.SG
DClassifier'> in 0.19. If both are left unset, they default to max_iter=5 and tol=None. If tol is not N
one, max_iter defaults to max_iter=1000. From 0.21, default max_iter will be 1000, and default tol will
be 1e-3.
  "and default tol will be 1e-3." % type(self), FutureWarning)
C:\Users\Friend\Anaconda3\lib\site-packages\sklearn\linear_model\stochastic_gradient.py:128: FutureWarn
ing: max_iter and tol parameters have been added in <class 'sklearn.linear_model.stochastic_gradient.SG
DClassifier'> in 0.19. If both are left unset, they default to max_iter=5 and tol=None. If tol is not N
one, max_iter defaults to max_iter=1000. From 0.21, default max_iter will be 1000, and default tol will
be 1e-3.
  "and default tol will be 1e-3." % type(self), FutureWarning)
C:\Users\Friend\Anaconda3\lib\site-packages\sklearn\linear_model\stochastic_gradient.py:128: FutureWarn
ing: max_iter and tol parameters have been added in <class 'sklearn.linear_model.stochastic_gradient.SG
DClassifier'> in 0.19. If both are left unset, they default to max_iter=5 and tol=None. If tol is not N
one, max_iter defaults to max_iter=1000. From 0.21, default max_iter will be 1000, and default tol will
be 1e-3.
  "and default tol will be 1e-3." % type(self), FutureWarning)
```

```
Log Loss : 1.7974071463759016
for alpha = 1
```

```
C:\Users\Friend\Anaconda3\lib\site-packages\sklearn\linear_model\stochastic_gradient.py:128: FutureWarn
ing: max_iter and tol parameters have been added in <class 'sklearn.linear_model.stochastic_gradient.SG
DClassifier'> in 0.19. If both are left unset, they default to max_iter=5 and tol=None. If tol is not N
one, max_iter defaults to max_iter=1000. From 0.21, default max_iter will be 1000, and default tol will
be 1e-3.
  "and default tol will be 1e-3." % type(self), FutureWarning)
C:\Users\Friend\Anaconda3\lib\site-packages\sklearn\linear_model\stochastic_gradient.py:128: FutureWarn
ing: max_iter and tol parameters have been added in <class 'sklearn.linear_model.stochastic_gradient.SG
DClassifier'> in 0.19. If both are left unset, they default to max_iter=5 and tol=None. If tol is not N
one, max_iter defaults to max_iter=1000. From 0.21, default max_iter will be 1000, and default tol will
be 1e-3.
  "and default tol will be 1e-3." % type(self), FutureWarning)
C:\Users\Friend\Anaconda3\lib\site-packages\sklearn\linear_model\stochastic_gradient.py:128: FutureWarn
ing: max_iter and tol parameters have been added in <class 'sklearn.linear_model.stochastic_gradient.SG
DClassifier'> in 0.19. If both are left unset, they default to max_iter=5 and tol=None. If tol is not N
one, max_iter defaults to max_iter=1000. From 0.21, default max_iter will be 1000, and default tol will
be 1e-3.
  "and default tol will be 1e-3." % type(self), FutureWarning)
```

```
Log Loss : 2.260549495688129
for alpha = 10
```

```
C:\Users\Friend\Anaconda3\lib\site-packages\sklearn\linear_model\stochastic_gradient.py:128: FutureWarn
ing: max_iter and tol parameters have been added in <class 'sklearn.linear_model.stochastic_gradient.SG
DClassifier'> in 0.19. If both are left unset, they default to max_iter=5 and tol=None. If tol is not N
one, max_iter defaults to max_iter=1000. From 0.21, default max_iter will be 1000, and default tol will
```

```
be 1e-3.
  "and default tol will be 1e-3." % type(self), FutureWarning)
C:\Users\Friend\Anaconda3\lib\site-packages\sklearn\linear_model\stochastic_gradient.py:128: FutureWarn
ing: max_iter and tol parameters have been added in <class 'sklearn.linear_model.stochastic_gradient.SG
DClassifier'> in 0.19. If both are left unset, they default to max_iter=5 and tol=None. If tol is not N
one, max_iter defaults to max_iter=1000. From 0.21, default max_iter will be 1000, and default tol will
be 1e-3.
  "and default tol will be 1e-3." % type(self), FutureWarning)
C:\Users\Friend\Anaconda3\lib\site-packages\sklearn\linear_model\stochastic_gradient.py:128: FutureWarn
ing: max_iter and tol parameters have been added in <class 'sklearn.linear_model.stochastic_gradient.SG
DClassifier'> in 0.19. If both are left unset, they default to max_iter=5 and tol=None. If tol is not N
one, max_iter defaults to max_iter=1000. From 0.21, default max_iter will be 1000, and default tol will
be 1e-3.
  "and default tol will be 1e-3." % type(self), FutureWarning)
```

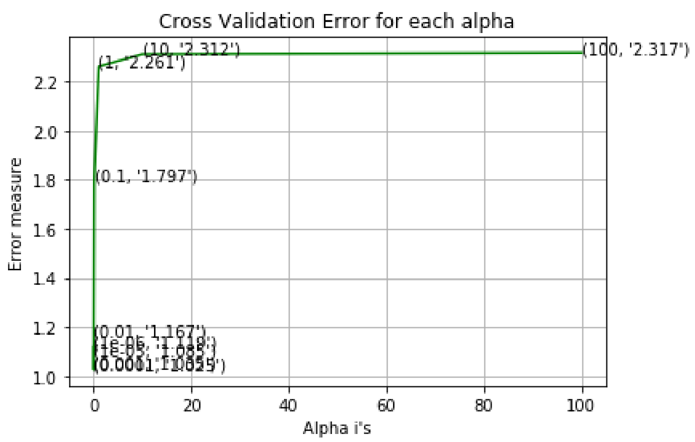
```
Log Loss : 2.3119916569705015
for alpha = 100
```

```
C:\Users\Friend\Anaconda3\lib\site-packages\sklearn\linear_model\stochastic_gradient.py:128: FutureWarn
ing: max_iter and tol parameters have been added in <class 'sklearn.linear_model.stochastic_gradient.SG
DClassifier'> in 0.19. If both are left unset, they default to max_iter=5 and tol=None. If tol is not N
one, max_iter defaults to max_iter=1000. From 0.21, default max_iter will be 1000, and default tol will
be 1e-3.
  "and default tol will be 1e-3." % type(self), FutureWarning)
C:\Users\Friend\Anaconda3\lib\site-packages\sklearn\linear_model\stochastic_gradient.py:128: FutureWarn
ing: max_iter and tol parameters have been added in <class 'sklearn.linear_model.stochastic_gradient.SG
DClassifier'> in 0.19. If both are left unset, they default to max_iter=5 and tol=None. If tol is not N
one, max_iter defaults to max_iter=1000. From 0.21, default max_iter will be 1000, and default tol will
be 1e-3.
  "and default tol will be 1e-3." % type(self), FutureWarning)
C:\Users\Friend\Anaconda3\lib\site-packages\sklearn\linear_model\stochastic_gradient.py:128: FutureWarn
ing: max_iter and tol parameters have been added in <class 'sklearn.linear_model.stochastic_gradient.SG
DClassifier'> in 0.19. If both are left unset, they default to max_iter=5 and tol=None. If tol is not N
one, max_iter defaults to max_iter=1000. From 0.21, default max_iter will be 1000, and default tol will
be 1e-3.
  "and default tol will be 1e-3." % type(self), FutureWarning)
```

```
Log Loss : 2.3172774070981212
```

In [82]:

```python
fig, ax = plt.subplots()
ax.plot(alpha, cv_log_error_array,c='g')
for i, txt in enumerate(np.round(cv_log_error_array,3)):
    ax.annotate((alpha[i],str(txt)), (alpha[i],cv_log_error_array[i]))
plt.grid()
plt.title("Cross Validation Error for each alpha")
plt.xlabel("Alpha i's")
plt.ylabel("Error measure")
plt.show()
```



In [83]:

```python
best_alpha = np.argmin(cv_log_error_array)
clf = SGDClassifier(class_weight='balanced', alpha=alpha[best_alpha], penalty='l2', loss='log', random_
state=42)
```

```
clf.fit(train_x_onehotCoding, train_y)
sig_clf = CalibratedClassifierCV(clf, method="sigmoid")
sig_clf.fit(train_x_onehotCoding, train_y)

predict_y = sig_clf.predict_proba(train_x_onehotCoding)
log_train_log_loss = log_loss(y_train, predict_y, labels=clf.classes_, eps=1e-15)
print('For values of best alpha = ', alpha[best_alpha], "The train log loss is:",log_train_log_loss)
predict_y = sig_clf.predict_proba(cv_x_onehotCoding)
log_cv_log_loss = log_loss(y_cv, predict_y, labels=clf.classes_, eps=1e-15)
print('For values of best alpha = ', alpha[best_alpha], "The cross validation log loss is:",log_cv_log_
loss)
predict_y = sig_clf.predict_proba(test_x_onehotCoding)
log_test_log_loss = log_loss(y_test, predict_y, labels=clf.classes_, eps=1e-15)
print('For values of best alpha = ', alpha[best_alpha], "The test log loss is:",log_test_log_loss)
```

```
C:\Users\Friend\Anaconda3\lib\site-packages\sklearn\linear_model\stochastic_gradient.py:128: FutureWarn
ing: max_iter and tol parameters have been added in <class 'sklearn.linear_model.stochastic_gradient.SG
DClassifier'> in 0.19. If both are left unset, they default to max_iter=5 and tol=None. If tol is not N
one, max_iter defaults to max_iter=1000. From 0.21, default max_iter will be 1000, and default tol will
be 1e-3.
  "and default tol will be 1e-3." % type(self), FutureWarning)
C:\Users\Friend\Anaconda3\lib\site-packages\sklearn\linear_model\stochastic_gradient.py:128: FutureWarn
ing: max_iter and tol parameters have been added in <class 'sklearn.linear_model.stochastic_gradient.SG
DClassifier'> in 0.19. If both are left unset, they default to max_iter=5 and tol=None. If tol is not N
one, max_iter defaults to max_iter=1000. From 0.21, default max_iter will be 1000, and default tol will
be 1e-3.
  "and default tol will be 1e-3." % type(self), FutureWarning)
C:\Users\Friend\Anaconda3\lib\site-packages\sklearn\linear_model\stochastic_gradient.py:128: FutureWarn
ing: max_iter and tol parameters have been added in <class 'sklearn.linear_model.stochastic_gradient.SG
DClassifier'> in 0.19. If both are left unset, they default to max_iter=5 and tol=None. If tol is not N
one, max_iter defaults to max_iter=1000. From 0.21, default max_iter will be 1000, and default tol will
be 1e-3.
  "and default tol will be 1e-3." % type(self), FutureWarning)
C:\Users\Friend\Anaconda3\lib\site-packages\sklearn\linear_model\stochastic_gradient.py:128: FutureWarn
ing: max_iter and tol parameters have been added in <class 'sklearn.linear_model.stochastic_gradient.SG
DClassifier'> in 0.19. If both are left unset, they default to max_iter=5 and tol=None. If tol is not N
one, max_iter defaults to max_iter=1000. From 0.21, default max_iter will be 1000, and default tol will
be 1e-3.
  "and default tol will be 1e-3." % type(self), FutureWarning)
```

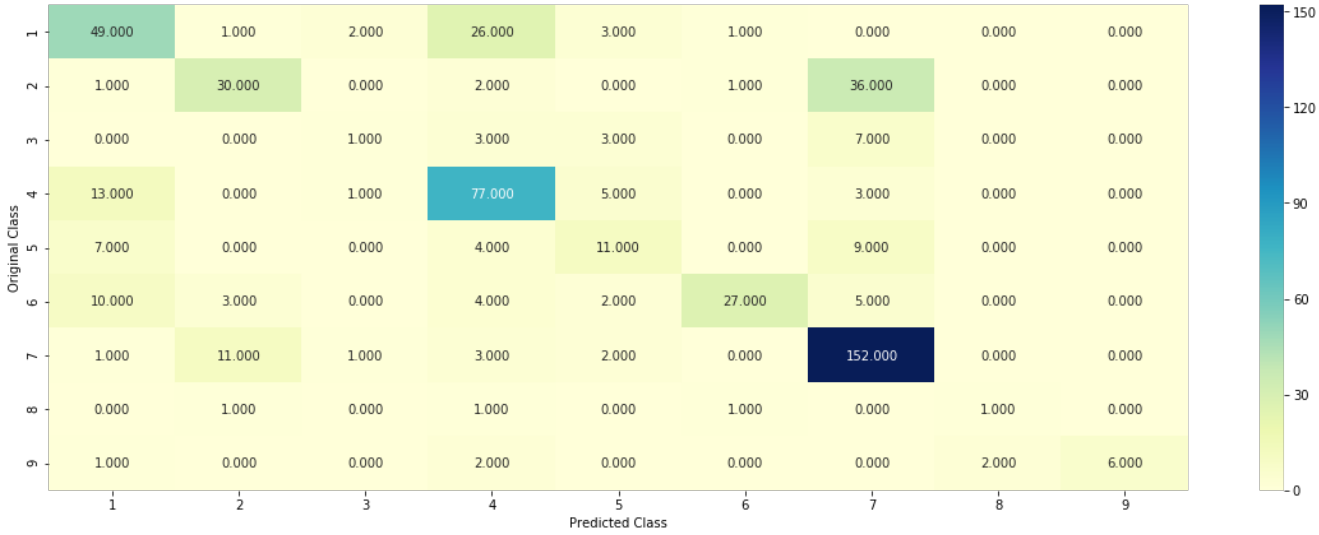
```
For values of best alpha =  0.0001 The train log loss is: 0.44009517292552414
For values of best alpha =  0.0001 The cross validation log loss is: 1.024864596109853
For values of best alpha =  0.0001 The test log loss is: 0.9603419948632671
```

In [84]:

```
clf = SGDClassifier(class_weight='balanced', alpha=alpha[best_alpha], penalty='l2', loss='log', random_
state=42)
predict_and_plot_confusion_matrix(train_x_onehotCoding, train_y, cv_x_onehotCoding, cv_y, clf)
```

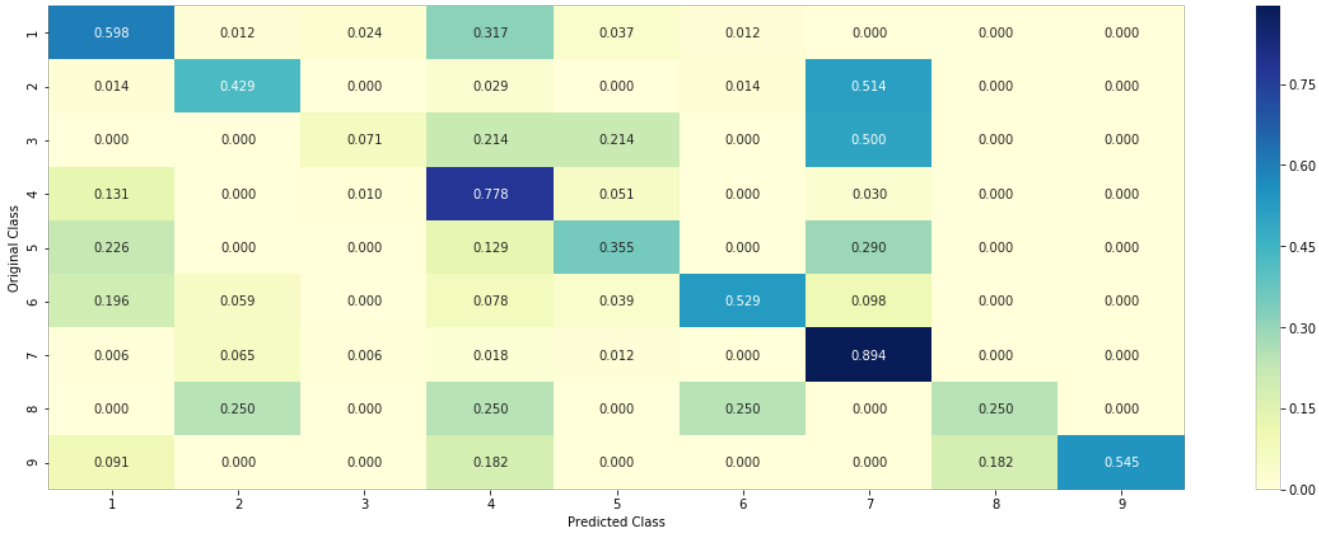
```
C:\Users\Friend\Anaconda3\lib\site-packages\sklearn\linear_model\stochastic_gradient.py:128: FutureWarn
ing: max_iter and tol parameters have been added in <class 'sklearn.linear_model.stochastic_gradient.SG
DClassifier'> in 0.19. If both are left unset, they default to max_iter=5 and tol=None. If tol is not N
one, max_iter defaults to max_iter=1000. From 0.21, default max_iter will be 1000, and default tol will
be 1e-3.
  "and default tol will be 1e-3." % type(self), FutureWarning)
C:\Users\Friend\Anaconda3\lib\site-packages\sklearn\linear_model\stochastic_gradient.py:128: FutureWarn
ing: max_iter and tol parameters have been added in <class 'sklearn.linear_model.stochastic_gradient.SG
DClassifier'> in 0.19. If both are left unset, they default to max_iter=5 and tol=None. If tol is not N
one, max_iter defaults to max_iter=1000. From 0.21, default max_iter will be 1000, and default tol will
be 1e-3.
  "and default tol will be 1e-3." % type(self), FutureWarning)
C:\Users\Friend\Anaconda3\lib\site-packages\sklearn\linear_model\stochastic_gradient.py:128: FutureWarn
ing: max_iter and tol parameters have been added in <class 'sklearn.linear_model.stochastic_gradient.SG
DClassifier'> in 0.19. If both are left unset, they default to max_iter=5 and tol=None. If tol is not N
one, max_iter defaults to max_iter=1000. From 0.21, default max_iter will be 1000, and default tol will
be 1e-3.
  "and default tol will be 1e-3." % type(self), FutureWarning)
C:\Users\Friend\Anaconda3\lib\site-packages\sklearn\linear_model\stochastic_gradient.py:128: FutureWarn
ing: max_iter and tol parameters have been added in <class 'sklearn.linear_model.stochastic_gradient.SG
DClassifier'> in 0.19. If both are left unset, they default to max_iter=5 and tol=None. If tol is not N
one, max_iter defaults to max_iter=1000. From 0.21, default max_iter will be 1000, and default tol will
be 1e-3.
  "and default tol will be 1e-3." % type(self), FutureWarning)
```

```
Log loss : 1.024864596109853
Number of mis-classified points : 0.33458646616541354
-------------------- Confusion matrix --------------------
```




```
-------------------- Precision matrix (Column Sum=1) --------------------
```


```
-------------------- Recall matrix (Row sum=1) --------------------
```



Conclusion

In [85]:

```
from prettytable import PrettyTable

Table = PrettyTable()
Table.field_names  = ["Model", "Train Loss", "Cross Validation Loss", "Test Loss"]
Table.add_row(["Logistic Regression", log_train_log_loss, log_cv_log_loss,log_test_log_loss])
print(Table)
```

```
+---------------------+---------------------+-----------------------+---------------------+
|        Model        |      Train Loss     | Cross Validation Loss |     Test Loss       |
+---------------------+---------------------+-----------------------+---------------------+
| Logistic Regression | 0.44009517292552414 |   1.024864596109853   | 0.9603419948632671 |
+---------------------+---------------------+-----------------------+---------------------+
```