CS 584: Machine Learning

Spring 2020 Assignment 4

In 2014, Allstate provided the data on Kaggle.com for the Allstate Purchase Prediction Challenge which is open. The data contain transaction history for customers that ended up purchasing a policy. For each Customer ID, you are given their quote history and the coverage options they purchased.

The data is available on the Blackboard as Purchase_Likelihood.csv.

1. It contains 665,249 observations on 97,009 unique Customer ID.
2. The nominal target variable is **insurance** which has these categories 0, 1, and 2
3. The nominal features are (categories are inside the parentheses):
   a. **group_size**. How many people will be covered under the policy (1, 2, 3 or 4)?
   b. **homeowner**. Whether the customer owns a home or not (0 = No, 1 = Yes)?
   c. **married_couple**. Does the customer group contain a married couple (0 = No, 1 = Yes)?

Question 1 (35 points)

You will build a multinomial logistic model with the following model specifications.

1. Enter the six effects to the model in this sequence:
   a. group_size
   b. homeowner
   c. married_couple
   d. group_size * homeowner
   e. group_size * married_couple
   f. homeowner * married_couple
2. Include the Intercept term in the model
3. The optimization method is Newton
4. The maximum number of iterations is 100
5. The tolerance level is 1e-8.
6. Use the sympy.Matrix().rref() method to identify the non-aliased parameters

Please answer the following questions based on your model.

a) (5 points) List the aliased columns that you found in your model matrix.

Ans:

```
In [22]: print("List the aliased columns that you found in your model matrix.\n", fullParams_2JM)
List the aliased columns that you found in your model matrix.
                                  1_x       0_y       1_y
const                             0.0  0.469691 -0.886845
group_size_1                      0.0  0.592130  0.546053
group_size_2                      0.0  0.999420  0.723139
group_size_3                      0.0  0.301413  0.503430
group_size_4                      0.0  0.000000  0.000000
homeowner_0                       0.0  0.776052  0.511026
homeowner_1                       0.0  0.000000  0.000000
married_couple_0                  0.0 -0.689248 -0.863883
married_couple_1                  0.0  0.000000  0.000000
group_size_1 * homeowner_0        0.0 -1.395311 -0.880455
group_size_1 * homeowner_1        0.0  0.000000  0.000000
group_size_2 * homeowner_0        0.0 -1.086733 -0.656173
group_size_2 * homeowner_1        0.0  0.000000  0.000000
group_size_3 * homeowner_0        0.0 -0.635960 -0.524617
group_size_3 * homeowner_1        0.0  0.000000  0.000000
group_size_4 * homeowner_0        0.0  0.000000  0.000000
group_size_4 * homeowner_1        0.0  0.000000  0.000000
group_size_1 * married_couple_0   0.0  0.962898  0.902886
group_size_1 * married_couple_1   0.0  0.000000  0.000000
group_size_2 * married_couple_0   0.0  0.094366  0.537978
group_size_2 * married_couple_1   0.0  0.000000  0.000000
group_size_3 * married_couple_0   0.0  0.676821  0.337205
group_size_3 * married_couple_1   0.0  0.000000  0.000000
group_size_4 * married_couple_0   0.0  0.000000  0.000000
group_size_4 * married_couple_1   0.0  0.000000  0.000000
homeowner_0 * married_couple_0    0.0  0.115368  0.135602
homeowner_0 * married_couple_1    0.0  0.000000  0.000000
homeowner_1 * married_couple_0    0.0  0.000000  0.000000
homeowner_1 * married_couple_1    0.0  0.000000  0.000000
```

b) (5 points) How many degrees of freedom does your model have?

Ans: 2

c) (20 points) After entering each model effect, calculate the Deviance test statistic, its degrees of freedom, and its significance value between the current model and the previous model. List your Deviance test results by the model effects in a table.

Step	Effect Entered	# Free Parameter	Log-Likelihood	Deviance	Degrees of Freedom	Significance
0	Intercept	2	-595406.7618844223	Not Applicable		
1	group_size	8	-5.9491e+05	987.5766005259939	6	4.347870389531338e-210
2	homeowner	10	-591979.0828339825	5867.781500353478	2	0.0
3	married_couple	12	-591936.7938327907	84.57800238369964	2	4.3064572185369587e-19
4	group_size * homeowner	18	-591809.754770109	254.07812536344863	6	5.5121059685664295e-52
5	group_size * married_couple	24	-591118.4835882676	1636.6204890462104	12	0.0
6	homeowner * married_couple	26	-591105.4931771926	25.980822149896994	2	2.2821077850015957e-06

d) (5 points) Calculate the Feature Importance Index as the negative base-10 logarithm of the significance value. List your indices by the model effects.

Effect Entered	Importance
Intercept	Not Applicable
group_size	209.36172341075647
homeowner	Not defined
married_couple	18.365879862820417
group_size * homeowner	51.2586824418404
group_size * married_couple	Not defined
homeowner * married_couple	5.641663847505022

Question 2 (25 points)

Please answer the following questions based on your multinomial logistic model in Question 1.

a) (10 points) For each of the sixteen possible value combinations of the three features, calculate the predicted probabilities for insurance = 0, 1, 2 based on your multinomial logistic model. List your answers in a table with proper labeling.

group_size	homeowner	married_couple	Prob(insurance = 0)	Prob(insurance = 1)	Prob(insurance = 2)

```
   ...:
   ...: predictions = thisFit.predict(X_Test)
   ...: pandas.DataFrame.join(pandas.DataFrame(all_combi, columns =
["group_size","homeOwner","Married_couple"]),predictions)
Out[28]:
    group_size  homeOwner  Married_couple         0         1         2
0            1          0               0  0.270442  0.599829  0.129729
1            1          0               1  0.244687  0.607062  0.148251
2            1          1               0  0.189498  0.695656  0.114846
3            1          1               1  0.154197  0.710625  0.135178
4            2          0               0  0.225803  0.642925  0.131272
5            2          0               1  0.203284  0.647446  0.149269
6            2          1               0  0.198085  0.685653  0.116262
7            2          1               1  0.161437  0.701504  0.137059
8            3          0               0  0.216149  0.672952  0.110898
9            3          0               1  0.185277  0.668221  0.146502
10           3          1               0  0.246822  0.616408  0.136770
11           3          1               1  0.202565  0.635071  0.162363
12           4          0               0  0.196873  0.685300  0.117827
13           4          0               1  0.177002  0.689196  0.133802
14           4          1               0  0.364840  0.520785  0.114376
15           4          1               1  0.308125  0.552149  0.139726
```

b) (5 points) Based on your answers in (a), what value combination of group_size, homeowner, and married_couple will maximize the odds value Prob(insurance = 1) / Prob(insurance = 0)? What is that maximum odd value?

Ans:

The maximum odd value is 4.6085394366106724

c) (5 points) Based on your model, what is the odds ratio for group_size = 3 versus group_size = 1, and insurance = 2 versus insurance = 0?
(*Hint*: The odds ratio is this odds (Prob(insurance = 2) / Prob(insurance = 0) | group_size = 3) divided by this odds ((Prob(insurance = 2) / Prob(insurance = 0) | group_size = 1).)

Ans:

Taking insurance=0 as reference target category = Log e((Prob(insurance =2)/Prob(insurance =0) | group_size = 3)) – log

e((Prob(insurance =2)/Prob(insurance =0) | group_size = 1))

= Parameter of (group_size = 3 | insurance =2) – Parameter of (group_size = 1 | insurance =2)

= 0.527471 - 0.801493

= -0.274022

Taking exponent of the previous value: exp(-0.274022) = 0.76031534813

d) (5 points) Based on your model, what is the odds ratio for homeowner = 1 versus homeowner = 0, and insurance = 0 versus insurance = 1?

Ans:

Log (Prob(A=0)/Prob(A=1) | homeowner = 1) - log((Prob(A=0)/Prob(A=1) | homeowner = 0)

= (0.800157 – 1.505554 * g1 – 1.164638 * g2 – 0.654639 * g3 + 0.212483 (1-m)

Exp (Prob(A=0)/Prob(A=1) | homeowner = 1) - log((Prob(A=0)/Prob(A=1) | homeowner = 0)

Question 3 (40 points)

You will build a Naïve Bayes model without any smoothing. In other words, the Laplace/Lidstone alpha is zero. Please answer the following questions based on your model.

a) (5 points) Show in a table the frequency counts and the Class Probabilities of the target variable.

insurance	0	1	2
Frequency Count	143691	426067	95491
Class Probability	0.215996	0.640462	0.143542

```
In [36]:
    ...:
    ...:
    ...: frequency = dataframe.groupby('insurance').size()
    ...: table = pd.DataFrame(columns = ['Count','Class_probability'])
    ...: table.Count = frequency
    ...: table.Class_probability = table.Count/dataframe.shape[0]
    ...: print(table)
            Count  Class_probability
insurance
0          143691           0.215996
1          426067           0.640462
2           95491           0.143542

In [37]:
```

b) (5 points) Show the crosstabulation table of the target variable by the feature group_size. The table contains the frequency counts.

group_size	insurance		
	0	1	2
1	115460	329552	74293
2	25728	91065	19600
3	2282	5069	1505
4	221	381	93

```
In [37]:
    ...:
    ...:
    ...: gs_crosstab = pd.crosstab(dataframe.insurance,dataframe.group_size)
    ...: gs_crosstab
Out[37]:
group_size       1      2     3    4
insurance
0           115460  25728  2282  221
1           329552  91065  5069  381
2            74293  19600  1505   93

In [38]:
```

c) (5 points) Show the crosstabulation table of the target variable by the feature homeowner. The table contains the frequency counts.

Ans:

```
In [38]:
    ...:
    ...:
    ...: ho_crosstab = pd.crosstab(dataframe.insurance,dataframe.homeowner)
    ...: ho_crosstab
Out[38]:
homeowner       0       1
insurance
0           78659   65032
1          183130  242937
2           46734   48757
```

d) (5 points) Show the crosstabulation table of the target variable by the feature married_couple. The table contains the frequency counts.

Ans:

```
In [39]:
    ...:
    ...:
    ...: mc_crosstab = pd.crosstab(dataframe.insurance,dataframe.married_couple)
    ...: mc_crosstab
Out[39]:
married_couple       0      1
insurance
0               117110  26581
1               333272  92795
2                75310  20181
```

e) (5 points) Calculate the Cramer's V statistics for the above three crosstabulations tables. Based on these Cramer's V statistics, which feature has the largest association with the target insurance?

Ans:

```
In [40]:
    ...:
    ...:
    ...: import scipy.stats as ss
    ...: def cramers_v_statistic(confusion_matrix):
    ...:     chi_squared = ss.chi2_contingency(confusion_matrix)[0]
    ...:     n = confusion_matrix.sum().sum()
    ...:     phi_2 = chi_squared/n
    ...:     r,k = confusion_matrix.shape
    ...:     phi2corr = max(0,(phi_2 - ((k-1)*(r-1))/(n-1)))
    ...:     rcorr = r - ((r-1)**2)/(n-1)
    ...:     kcorr = k - ((k-1)**2)/(n-1)
    ...:     print(np.sqrt(phi2corr / min( (kcorr-1), (rcorr-1))))

In [41]:
    ...:
    ...:
    ...:
    ...:
    ...: print("The Cramers V Statistic values for each variable are as follows \n")
    ...: print("For group_size")
    ...: print(cramers_v_statistic(gs_crosstab))
    ...: print()
    ...:
    ...: print("For homeowner")
    ...: print(cramers_v_statistic(ho_crosstab))
    ...: print()
    ...:
    ...: print("For married_couple")
    ...: print(cramers_v_statistic(mc_crosstab))
    ...: print()
The Cramers V Statistic values for each variable are as follows

For group_size
0.027018729877001067
None

For homeowner
0.09707100827090977
None

For married_couple
0.032375272919927714
None
```

f) (10 points) For each of the sixteen possible value combinations of the three features, calculate the predicted probabilities for insurance = 0, 1, 2 based on the Naïve Bayes model. List your answers in a table with proper labeling.

group_size	homeowner	married_couple	Prob(insurance = 0)	Prob(insurance = 1)	Prob(insurance = 2)
1	0	0	0.269722	0.580133	0.150145
1	0	1	0.232789	0.614219	0.152992
1	1	0	0.194038	0.669659	0.136303
1	1	1	0.164935	0.698278	0.136787
2	0	0	0.231143	0.616518	0.152338
2	0	1	0.198016	0.647907	0.154078

group_size	homeowner	married_couple	Prob(insurance = 0)	Prob(insurance = 1)	Prob(insurance = 2)
2	1	0	0.163628	0.700288	0.136085
2	1	1	0.138274	0.725955	0.135771
3	0	0	0.308219	0.515924	0.175856
3	0	1	0.268311	0.550951	0.180738
3	1	0	0.226972	0.609612	0.163416
3	1	1	0.194370	0.640410	0.165221
4	0	0	0.375490	0.487810	0.136700
4	0	1	0.330743	0.527098	0.142158
4	1	0	0.282173	0.588196	0.129631
4	1	1	0.243930	0.623766	0.132304

```
In [70]: Test[['group_size','homeowner','married_couple']]
Out[70]:
    group_size  homeowner  married_couple
0            1          0               0
1            1          0               1
2            1          1               0
3            1          1               1
4            2          0               0
5            2          0               1
6            2          1               0
7            2          1               1
8            3          0               0
9            3          0               1
10           3          1               0
11           3          1               1
12           4          0               0
13           4          0               1
14           4          1               0
15           4          1               1

In [71]: Test[['insurance=0','insurance=1','insurance=2']]
Out[71]:
    insurance=0  insurance=1  insurance=2
0      0.269722     0.580133     0.150145
1      0.232789     0.614219     0.152992
2      0.194038     0.669659     0.136303
3      0.164935     0.698278     0.136787
4      0.231143     0.616518     0.152338
5      0.198016     0.647907     0.154078
6      0.163628     0.700288     0.136085
7      0.138274     0.725955     0.135771
8      0.308219     0.515924     0.175856
9      0.268311     0.550951     0.180738
10     0.226972     0.609612     0.163416
11     0.194370     0.640410     0.165221
12     0.375490     0.487810     0.136700
13     0.330743     0.527098     0.142158
14     0.282173     0.588196     0.129631
15     0.243930     0.623766     0.132304
```

g) (5 points) Based on your model, what value combination of group_size, homeowner, and married_couple will maximize the odds value Prob(insurance = 1) / Prob(insurance = 0)? What is that maximum odd value?

Ans: The maximum value = [group_size , homeowner , married_couple] = [2,1,1]

The maximum odds value for Prob(A=1) / Prob(A = 0) is 5.250113

```
In [72]:
    ...: m=[]
    ...: for i in range(len(nbp)):
    ...:     temp=nbp[i][1]/nbp[i][0]
    ...:     m.append([temp])
    ...: print(numpy.array(m).max())
    ...:
    ...: numpy.array(m).max()
    ...: c[numpy.where(m == numpy.array(m).max())[0][0]]
5.250112589270714
Out[72]: (2, 1, 1)
```