BUAN 6337.001 - Predictive Analytics Using SAS

Homework – 5

Submitted by

Hemasrila Sampath Kumar
Aditi Maharudra Kulkarni
Chitresh Kumar
Aman Pandey
Rishabh Wagh
Michal Zamulinski
(Group -9)

Homework dataset (Crackers): This dataset contains store sales data of crackers at a supermarket that carries four brands of crackers. Each observation corresponds to one purchase occasion and provides data on the price, display and feature of each brand as well as which brand was chosen.

1. OBS : = Observation number
2. Private, Keebler, Sunshine, Nabisco: Indicator variables for which brand was chosen. Value of 1
3. indicates the brand that was chosen. Other 3 brands will be 0 in that observation.
4. PricePrivate, PriceNabisco, PriceKeebler and PriceSunshine: Prices that were offered by each brand
5. for that purchase occasion.
6. DisplPrivate : = 1 if Private had a store display, =0 if Private did not have a store display
7. DisplKeebler : = 1 if Keebler had a store display, =0 if Keebler did not have a store display
8. DisplSunshin:= 1 if Sunshin had a store display, =0 if Sunshin did not have a store display
9. DisplNabisco:= 1 if Nabiscohad a store display, =0 if Nabisco did not have a store display
10. FeatPrivate: = 1 if Privatehad a store feature, =0 if Private did not have a store feature
11. FeatKeebler: = 1 if Keebler had a store feature, =0 if Keebler did not have a store feature
12. FeatSunshin: = 1 if Sunshinhad a store feature, =0 if Sunshin did not have a store feature
13. FeatNabisco: = 1 if Nabiscohad a store feature, =0 if Nabisco did not have a store feature

Question – 1:

Use PROC SURVEYSELECT to sample the original data into training and testing data sets. Use 75% for training and 25% for testing. Use the seed= option to set random seed to a value of 23.

Answer :

Firstly, the dataset was loaded into SAS and we used PROC SURVEYSELECT to sample the original data such that 75% was used for the training dataset and 25% was used for the testing dataset. A seed of 23 was set. The datasets and code are shown below,

Code *:*

```
LIBNAME HW5 'E:\Users\hxs190042\Desktop\Predictive\HomeWork 5';
ODS GRAPHICS ON;
/*
Question -1 :
) Use PROC SURVEYSELECT to sample the original data into training and testing data sets. Use 75% for
training and 25% for testing. Use the seed= option to set random seed to a value of 23.
*/

data crackers;
set HW5.crackers;
run;


/* Question 1 */
proc surveyselect data=HW5.crackers out=crackers_sampled outall samprate=0.75 seed=23;
run;
* Split data into training vs. test set;
data crackers_training crackers_test;
 set crackers_sampled;
 if selected then output crackers_training;
 else output crackers_test;
run;

proc print data = crackers_test;
title 'Testing Dataset';
run;
```

Output :

The SURVEYSELECT Procedure

Selection Method	Simple Random Sampling

Input Data Set	CRACKERS
Random Number Seed	23
Sampling Rate	0.75
Sample Size	2469
Selection Probability	0.750228
Sampling Weight	0
Output Data Set	CRACKERS_SAMPLED

Training Dataset:

Training Dataset

Obs	Selected	OBS	PRIVATE	SUNSHINE	KEEBLER	NABISCO	PRICEPRIVATE	PRICESUNSHINE	PRICEKEEBLER	PRICENABISCO	FeatPrivate	FeatSunshine	FeatKeet
1	1	1	0	0	0	1	0.709999979	0.99000001	1.089999914	0.99000001	0	0	
2	1	2	0	1	0	0	0.779999912	0.49000001	1.089999914	1.089999914	0	0	
3	1	3	0	0	0	1	0.779999912	1.029999971	1.089999914	0.889999986	0	0	
4	1	4	0	0	0	1	0.639999986	1.089999914	1.089999914	1.190000057	0	0	
5	1	5	0	0	0	1	0.839999914	0.889999986	1.089999914	1.190000057	0	0	
6	1	6	0	1	0	0	0.779999912	1.089999914	1.089999914	1.289999962	0	0	
7	1	7	0	0	0	1	0.779999912	1.089999914	1.190000057	1.289999962	0	0	
8	1	8	0	0	0	1	0.779999912	1.089999914	1.209999919	1.089999914	0	0	
9	1	9	0	0	0	1	0.779999912	0.789999962	1.209999919	1.089999914	0	0	
10	1	10	0	0	0	1	0.959999979	1.089999914	1.129999995	1.089999914	0	0	
11	1	11	0	0	0	1	0.859999955	1.089999914	1.209999919	0.99000001	0	0	
12	1	12	0	0	0	1	0.859999955	0.889999986	1.209999919	0.99000001	0	0	
13	1	15	0	0	0	1	0.959999979	1.289999962	1.039999962	1.289999962	0	0	
14	1	16	0	0	0	1	0.689999998	0.789999962	0.99000001	0.689999998	0	0	
15	1	18	0	1	0	0	0.689999998	0.789999962	1.25	1.059999824	0	1	
16	1	19	0	1	0	0	0.689999998	0.789999962	1.25	1.059999824	0	0	
17	1	22	0	1	0	0	0.649999976	0.789999962	1.349999905	1.089999914	0	0	
18	1	23	0	0	1	0	0.599999964	0.970000029	0.99000001	1.25	0	0	
19	1	24	0	1	0	0	0.730000019	1.089999914	1.349999905	1.289999962	0	0	
20	1	26	0	0	0	1	0.589999914	1.049999952	1.25	0.99000001	0	0	
21	1	27	0	0	1	0	0.589999914	1.049999952	0.99000001	1.289999962	0	0	
22	1	28	0	1	0	0	0.889999986	0.889999986	1.349999905	1.039999962	0	0	
23	1	29	0	0	0	1	0.589999914	1.049999952	1.289999962	0.99000001	0	0	
24	1	31	0	1	0	0	0.649999976	1.049999952	1.289999962	1.289999962	0	0	
25	1	32	0	0	0	1	0.589999914	0.680000007	0.879999995	0.879999995	0	0	

Testing Dataset :

Testing Dataset

Obs	Selected	OBS	PRIVATE	SUNSHINE	KEEBLER	NABISCO	PRICEPRIVATE	PRICESUNSHINE	PRICEKEEBLER	PRICENABISCO	FeatPrivate	FeatSunshine	FeatKeebl
1	0	13	0	0	0	1	0.959999979	1.089999914	1.089999914	1.289999962	0	0	
2	0	14	0	0	0	1	0.789999962	1.089999914	1.089999914	1.289999962	0	0	
3	0	17	0	1	0	0	0.649999976	0.689999998	1.049999952	0.889999986	0	1	
4	0	20	0	1	0	0	0.689999998	0.789999962	1.25	0.99000001	0	0	
5	0	21	0	1	0	0	0.569999933	0.970000029	1.149999976	1.089999914	1	0	
6	0	25	0	1	0	0	0.759999931	1.089999914	1.349999905	1.149999976	0	0	
7	0	30	0	0	0	1	0.649999976	1.049999952	1.289999962	0.99000001	0	0	
8	0	35	1	0	0	0	0.5	0.889999986	1.039999962	0.99000001	1	0	
9	0	37	1	0	0	0	0.889999986	1.089999914	1.089999914	1.190000057	0	0	
10	0	41	1	0	0	0	0.889999986	0.789999962	1.209999919	1.089999914	0	0	
11	0	45	1	0	0	0	0.680000007	0.930000007	1.190000057	0.980000019	1	0	
12	0	46	1	0	0	0	0.549999952	0.980000019	0.99000001	1.089999914	0	0	
13	0	59	1	0	0	0	0.549999952	0.980000019	1.209999919	1.190000057	0	0	
14	0	62	1	0	0	0	0.549999952	0.980000019	1.190000057	1.289999962	0	0	
15	0	68	1	0	0	0	0.589999914	0.889999986	1.209999919	0.99000001	0	0	
16	0	70	1	0	0	0	0.889999986	1.190000057	1.289999962	1.289999962	0	0	
17	0	72	1	0	0	0	0.589999914	0.949999988	1.190000057	1.289999962	0	0	
18	0	76	0	0	0	1	0.680000007	0.889999986	1.079999924	0.889999986	0	0	
19	0	83	0	0	1	0	0.680000007	1.039999962	1.089999914	1.089999914	0	0	
20	0	88	0	0	1	0	0.649999976	1.049999952	0.99000001	0.99000001	0	0	
21	0	90	0	0	0	1	0.680000007	0.849999964	0.980000019	0.889999986	0	0	
22	0	93	0	0	0	1	0.779999912	1.029999971	1.089999914	0.769999921	0	0	
23	0	94	1	0	0	0	0.549999952	0.789999962	1.049999952	1.089999914	0	0	
24	0	95	1	0	0	0	0.549999952	0.980000019	1.089999914	0.889999986	0	0	
25	0	98	1	0	0	0	0.549999952	1.089999914	1.089999914	1.190000057	0	0	
26	0	100	0	0	0	1	0.599999964	0.889999986	1.149999976	0.99000001	0	0	

Question -2 :

The store manager would like to predict the choice probabilities for each brand of crackers depending on the price, display and promotion for all brands. What type of multinomial logit model would you estimate – a model with alternative-specific characteristics or with individual-specific characteristics? Write the general utility model to estimate this logit model.

Answer:

To predict the choice probabilities for each brand of crackers depending on the price, display and promotion for all brands, ***we would need to estimate a model with alternative-specific characteristics****. The variables such as PricePrivate, PriceNabisco, PriceKeebler, and PriceSunshine are alternative specific variable that change across alternatives. The data does not have any individual specific variables Let* $I_{Keebler}$, $I_{Sunshine}$, $I_{Nabisco}$, *be indicators for Keeber, Sunshine and Nabisco.* ***Since there are no individual-specific noticeable effects*** *and only alternative characteristics such as price, feature and display exist for each choice occasion at the time of purchase****, we use a model with alternative-specific characteristics****.*

The general utility model to estimate this logit model is as follows:

$$V_j = \beta\ Price_j + \alpha_{0,\ Private}\ I_{Private} + \alpha_{1,\ Private}\ feat_{Private} * I_{Private} + \alpha_{2,\ Private}\ disp_{Private} * I_{Private}$$
$$+ \alpha_{0,\ Sunshine}\ I_{Sunshine} + \alpha_{1,\ Sunshine}\ feat_{Sunshine} * I_{Sunshine} + \alpha_{2,\ Sunshine}\ disp_{Sunshine} * I_{Sunshine}$$
$$+ \alpha_{0,\ Keebler}\ I_{Keebler} + \alpha_{1,\ Keebler}\ feat_{Keebler} * I_{Keebler} + \alpha_{2,\ Keebler}\ disp_{Keebler} * I_{Keebler}$$
$$+ \alpha_{1,\ Nabisco}\ feat_{Nabisco} * I_{Nabisco} + \alpha_{2,\ Nabisco}\ disp_{Nabisco} * I_{Nabisco}$$

Question -3 :

Is the data formatted as needed to estimate the above multinomial logit model using PROC LOGISTIC? If not, how should the data be formatted? Reformat the data as necessary.

Answer :

Since we use a model with alternative-specific characteristics here, the training data is not in a format we need to estimate the multinomial logit model using PROC LOGISTIC. The training dataset should be formatted in such a way that each observation has substitutes for all the 4 brands of crackers with their corresponding prices, feature and display along with a choice variable which is 1 for a certain observation's choice of brand while the other brand choices are 0.

Code :

```
/* Question 3

Is the data formatted as needed to estimate the above multinomial logit model using PROC LOGISTIC?
If not, how should the data be formatted? Reformat the data as necessary.*/


data Cracker_train_format (keep = selected obs brand_feature brand_display brand_name brand_price brand_choice);
set crackers_training;
array name[4] $ ('Private' 'Sunshine' 'Keebler' 'Nabisco');
array price[4] priceprivate pricesunshine pricekeebler pricenabisco;
array feature[4] featprivate featsunshine featkeebler featnabisco;
array display[4] displprivate displsunshine displkeebler displnabisco;
array choice[4] private sunshine keebler nabisco;
do i = 1 to 4;
brand_name = name[i];
brand_price = price[i];
brand_feature = feature[i];
brand_display = display[i];
brand_choice = choice[i];
output;
end;
run;

PROC print data=Cracker_train_format;
title 'Formatted training data';
run;
```

The formatted training data is as below.

Formatted training data

Obs	Selected	OBS	brand_name	brand_price	brand_feature	brand_display	brand_choice
1	1	1	Private	0.71000	0	0	0
2	1	1	Sunshine	0.99000	0	0	0
3	1	1	Keebler	1.09000	0	0	0
4	1	1	Nabisco	0.99000	0	0	1
5	1	2	Private	0.78000	0	0	0
6	1	2	Sunshine	0.49000	0	1	1
7	1	2	Keebler	1.09000	0	0	0
8	1	2	Nabisco	1.09000	0	0	0
9	1	3	Private	0.78000	0	0	0
10	1	3	Sunshine	1.03000	0	0	0
11	1	3	Keebler	1.09000	0	0	0
12	1	3	Nabisco	0.89000	0	0	1
13	1	4	Private	0.64000	0	0	0
14	1	4	Sunshine	1.09000	0	0	0
15	1	4	Keebler	1.09000	0	0	0
16	1	4	Nabisco	1.19000	0	0	1
17	1	5	Private	0.84000	0	0	0
18	1	5	Sunshine	0.89000	0	0	0
19	1	5	Keebler	1.09000	0	0	0
20	1	5	Nabisco	1.19000	0	0	1
21	1	6	Private	0.78000	0	0	0
22	1	6	Sunshine	1.09000	0	0	1
23	1	6	Keebler	1.09000	0	0	0
24	1	6	Nabisco	1.29000	0	1	0
25	1	7	Private	0.78000	0	0	0
26	1	7	Sunshine	1.09000	0	0	0

Question -4 :

Estimate the logit model on the training sample using PROC LOGISTIC and report the estimation results (model parameters, significance).

Answer :

Using PROC LOGISTIC, the logit model was estimated using the training sample. The results are as follows,

Code:

```
/* Question 4
Estimate the logit model on the training sample using PROC LOGISTIC and report the estimation results (model parameters, significance).*/

proc logistic data = Cracker_train_format;
strata obs ;
class brand_name (ref = 'Private') / param=glm;
model brand_choice (event='1') = brand_name brand_price brand_name*brand_feature brand_name*brand_display / clodds=wald orpvalue;
run;
```

The LOGISTIC Procedure

Conditional Analysis

Model Information	
Data Set	WORK.CRACKER_TRAIN_FORMAT
Response Variable	brand_choice
Number of Response Levels	2
Number of Strata	2469
Model	binary logit
Optimization Technique	Newton-Raphson ridge

Number of Observations Read	9876
Number of Observations Used	9876
Number of Observations Informative	9876

Response Profile		
Ordered Value	brand_choice	Total Frequency
1	0	7407
2	1	2469

Probability modeled is brand_choice=1.

Class Level Information					
Class	Value	Design Variables			
brand_name	Keebler	1	0	0	0
	Nabisco	0	1	0	0
	Sunshine	0	0	1	0
	Private	0	0	0	1

Strata Summary				
Response Pattern	brand_choice 0	brand_choice 1	Number of Strata	Frequency
1	3	1	2469	9876

Note: The following parameters have been set to 0, since the variables are a linear combination of other variables as shown.

brand_namePrivate =	1 - brand_nameKeebler - brand_nameNabisco - brand_nameSunshine

Newton-Raphson Ridge Optimization

Without Parameter Scaling

Convergence criterion (GCONV=1E-8) satisfied.

Model Fit Statistics		
Criterion	Without Covariates	With Covariates
AIC	6845.522	5089.535
SC	6845.522	5175.909
-2 Log L	6845.522	5065.535

Testing Global Null Hypothesis: BETA=0			
Test	Chi-Square	DF	Pr > ChiSq
Likelihood Ratio	1779.9869	12	<.0001
Score	1696.6539	12	<.0001
Wald	1214.3634	12	<.0001

Type 3 Analysis of Effects			
		Wald	

Wald	1214.3634	12	<.0001

Type 3 Analysis of Effects

Effect	DF	Wald Chi-Square	Pr > ChiSq
brand_name	3	874.6607	<.0001
brand_price	1	156.0034	<.0001
brand_fea*brand_name	4	20.6969	0.0004
brand_dis*brand_name	4	11.4293	0.0221

Analysis of Conditional Maximum Likelihood Estimates

Parameter		DF	Estimate	Standard Error	Wald Chi-Square	Pr > ChiSq
brand_name	Keebler	1	-0.2664	0.1409	3.5763	0.0586
brand_name	Nabisco	1	1.6731	0.1178	201.8435	<.0001
brand_name	Sunshine	1	-0.8451	0.1193	50.2187	<.0001
brand_name	Private	0	0	.	.	.
brand_price		1	-3.0324	0.2428	156.0034	<.0001
brand_fea*brand_name	Keebler	1	0.5723	0.2931	3.8128	0.0509
brand_fea*brand_name	Nabisco	1	0.6033	0.1651	13.3554	0.0003
brand_fea*brand_name	Sunshine	1	0.5448	0.2846	3.6644	0.0556
brand_fea*brand_name	Private	1	0.1090	0.2392	0.2076	0.6487
brand_dis*brand_name	Keebler	1	0.2892	0.2318	1.5557	0.2123
brand_dis*brand_name	Nabisco	1	0.0665	0.0893	0.5557	0.4560
brand_dis*brand_name	Sunshine	1	0.4867	0.1921	6.4184	0.0113
brand_dis*brand_name	Private	1	-0.2838	0.1689	2.8245	0.0928

Odds Ratio Estimates and Wald Confidence Intervals

Effect	Unit	Estimate	95% Confidence Limits		p-Value
brand_price	1.0000	0.048	0.030	0.078	<.0001

The odds ratio coefficient of 0.048 shows that 1 unit increase in price in crackers declines the odds of buying crackers.

Question -5:

Reproduce your results using multinomial discrete choice command PROC MDC

Answer:

Using PROC MDC, the results as using PROC LOGISTIC are reproduced as shown below. However here we manually restrict the brand 'PRIVATE' here.

Code:

```
/* Question 5
Reproduce your results using multinomial discrete choice command PROC MDC */

proc mdc data = Cracker_train_format;
title 'Multinomial Discrete choice command MDC';
id OBS;
class brand_name;
model brand_choice = brand_name brand_price brand_name*brand_feature brand_name*brand_display / type = clogit nchoice = 4;
restrict brand_nameprivate =0;
run;
```

Output:

Multinomial Discrete choice command MDC

The MDC Procedure

Conditional Logit Estimates

Algorithm converged.

Model Fit Summary	
Dependent Variable	brand_choice
Number of Observations	2469
Number of Cases	9876
Log Likelihood	-2533
Log Likelihood Null (LogL(0))	-3423
Maximum Absolute Gradient	1.6434E-10
Number of Iterations	5
Optimization Method	Newton-Raphson
AIC	5090
Schwarz Criterion	5159

Discrete Response Profile			
Index	CHOICE	Frequency	Percent
0	1	783	31.71
1	2	180	7.29
2	3	179	7.25
3	4	1327	53.75

Goodness-of-Fit Measures		
Measure	Value	Formula
Likelihood Ratio (R)	1780	2 * (LogL - LogL0)
Upper Bound of R (U)	6845.5	- 2 * LogL0

Multinomial Discrete choice command MDC

The MDC Procedure

Conditional Logit Estimates

Parameter Estimates

Parameter	DF	Estimate	Standard Error	t Value	Approx Pr > |t|	Parameter Label
BRAND_NAMEPrivate	0	0	0			
BRAND_NAMESunshine	1	-0.8451	0.1193	-7.09	<.0001	
BRAND_NAMEKeebler	1	-0.2664	0.1409	-1.89	0.0586	
BRAND_NAMENabisco	1	1.6731	0.1178	14.21	<.0001	
brand_price	1	-3.0324	0.2428	-12.49	<.0001	
BRAND_NAMEPrivateBRAND_FEATURE	1	0.1090	0.2392	0.46	0.6487	
BRAND_NAMESunshine2	1	0.5448	0.2846	1.91	0.0556	
BRAND_NAMEKeeblerBRAND_FEATURE	1	0.5723	0.2931	1.95	0.0509	
BRAND_NAMENabiscoBRAND_FEATURE	1	0.6033	0.1651	3.65	0.0003	
BRAND_NAMEPrivateBRAND_DISPLAY	1	-0.2838	0.1689	-1.68	0.0928	
BRAND_NAMESunshine3	1	0.4867	0.1921	2.53	0.0113	
BRAND_NAMEKeeblerBRAND_DISPLAY	1	0.2892	0.2318	1.25	0.2123	
BRAND_NAMENabiscoBRAND_DISPLAY	1	0.0665	0.0893	0.75	0.4560	
Restrict1	1	-5.76E-11	12.2086	-0.00	1.0000*	Linear EC [1]

* Probability computed using beta distribution.

Linearly Independent Active Linear Constraints

1	0	=	0	+	1.0000	*	BRAND_NAMEPrivate

Question -6 :

Use PROC MDC to predict the choice probabilities for the test sample using the estimated model

Answer:

Before proceeding, we reformatted one dataset that contains both the training and test dataset as shown below,

Following this, we set the choice data (i.e. test data) to be missing as shown in the code.

```
/* Question 6
Use PROC MDC to predict the choice probabilities for the test sample using the estimated model.
*/

/* Formatting test data*/
data crackers_format(keep = selected obs brand_feature brand_display brand_name brand_price brand_choice);
set crackers_sampled;
array name[4] $ ('Private' 'Sunshine' 'Keebler' 'Nabisco');
array price[4] priceprivate pricesunshine pricekeebler pricenabisco;
array feature[4] featprivate featsunshine featkeebler featnabisco;
array display[4] displprivate displsunshine displkeebler displnabisco;
array choice[4] private sunshine keebler nabisco;
do i = 1 to 4;
brand_name = name[i];
brand_price = price[i];
brand_feature = feature[i];
brand_display = display[i];
brand_choice = choice[i];
output;
end;
run;
data crackers_prob;
set crackers_format;
if selected=0 then brand_choice=.;
run;
proc mdc data = crackers_prob;
id obs;
class brand_name;
model brand_choice = brand_name brand_price brand_name*brand_feature brand_name*brand_display / type = mprobit nchoice = 4;
restrict brand_nameprivate=0;
output out = probdata pred = prob;
run;
```

Finally, using the OUTPUT AND PRED statements, the choice probabilities for the entire dataset using the estimated model is calculated as shown below.

Using the predicted probabilities calculated in the previous step, we create a dataset with 1 row per observation associated with the maximum predicted probability as the 'predicted choice' for the test data and merge this data with the 'actual choice' for the test data using PROC SQL as shown below,

```
NOTE: The data set WORK.PROBDATA has 13164 observations and 20 variables.
NOTE: PROCEDURE MDC used (Total process time):
      real time           1:41.74
      cpu time            1:37.89
```

```
NOTE: There were 13164 observations read from the data set WORK.CRACKERS_FORMAT.
NOTE: The data set WORK.CRACKERS_PROB has 13164 observations and 7 variables.
NOTE: DATA statement used (Total process time):
      real time           0.04 seconds
      cpu time            0.03 seconds
```

```
NOTE: There were 3291 observations read from the data set WORK.CRACKERS_SAMPLED.
NOTE: The data set WORK.CRACKERS_FORMAT has 13164 observations and 7 variables.
NOTE: DATA statement used (Total process time):
      real time           0.08 seconds
      cpu time            0.07 seconds
```

```
proc print data = crackers_pred;
title'predicted';
run;
```

predicted

Obs	OBS	Selected	predicted_choice	prob	actual_choice
1	1	1	Nabisco	0.61003	Nabisco
2	2	1	Nabisco	0.41385	Sunshine
3	3	1	Nabisco	0.70627	Nabisco
4	4	1	Nabisco	0.44602	Nabisco
5	5	1	Nabisco	0.50637	Nabisco
6	6	1	Nabisco	0.45255	Sunshine
7	7	1	Nabisco	0.46458	Nabisco
8	8	1	Nabisco	0.60899	Nabisco
9	9	1	Nabisco	0.53375	Nabisco
10	10	1	Nabisco	0.66636	Nabisco
11	11	1	Nabisco	0.69774	Nabisco
12	12	1	Nabisco	0.64831	Nabisco
13	13	0	Nabisco	0.50264	Nabisco
14	14	0	Nabisco	0.44214	Nabisco
15	15	1	Nabisco	0.51857	Nabisco
16	16	1	Nabisco	0.75265	Nabisco
17	17	0	Nabisco	0.55305	Sunshine
18	18	1	Nabisco	0.47369	Sunshine
19	19	1	Nabisco	0.51035	Sunshine
20	20	0	Nabisco	0.59569	Sunshine
21	21	0	Nabisco	0.62282	Sunshine
22	22	1	Nabisco	0.48753	Sunshine

*Following this, we use PROC FREQ to create a 4*4 classification table with the actual and predicted choices are shown.*

```
proc freq data = crackers_pred;
tables actual_choice * predicted_choice;
run;
```

Multinomial Discrete choice command MDC

The FREQ Procedure

Table of actual_choice by predicted_choice

Frequency Percent Row Pct Col Pct	predicted_choice			
actual_choice	Nabisco	Private	Sunshine	Total
Keebler	192 5.83 84.96 6.87	34 1.03 15.04 7.01	0 0.00 0.00 0.00	226 6.87
Nabisco	1626 49.41 90.79 58.15	162 4.92 9.05 33.40	3 0.09 0.17 30.00	1791 54.42
Private	780 23.70 75.36 27.90	252 7.66 24.35 51.96	3 0.09 0.29 30.00	1035 31.45
Sunshine	198 6.02 82.85 7.08	37 1.12 15.48 7.63	4 0.12 1.67 40.00	239 7.26
Total	2796 84.96	485 14.74	10 0.30	3291 100.00

*Since the predicted choice data probabilities only evaluated to the choices 'Private', 'Sunshine' and 'Nabisco', the table reduces to a 4*3 table as shown above.*