Linear Models in SAS

Statistics 135

Autumn 2005

Linear Models in SAS

There are a number of ways to fit linear models in **SAS**, though some deal with specific situations. These include

- `PROC ANOVA`: Analysis of Variance for balanced designs
- `PROC REG`: Regression analysis. Does not handle categorical factors. Includes a wide range of diagnostics and model selection approaches.
- `PROC GLM`: General Linear Model. Equivalent of `lm` in **S**.
- `PROC MIXED`: Mixed effects ANOVA which are models which include random effects
- `PROC VARCOMP`: Another approach to random effects models.
- `PROC NESTED`: Nested ANOVA model

- `PROC ORTHOREG`: Alternative to `REG` and `GLM` to handle ill-conditioned (high collinearity) problems

- `PROC ROBUSTREG`: Robust regression approaches.

We will focus on the first three (`ANOVA, REG, GLM`). Generally anything you can do in `ANOVA` or `REG` can be done in `GLM`, but not everything. For example, to use automatic model selection procedures, you must use `PROC REG`. These procedures don't exist in `PROC GLM`.

PROC REG

A general linear regression model procedure. Will only work with continuous predictors, predefined indicator variables, and predefined products or powers of variables. Due to this, you cannot look at interactions on the fly. You need to plan these out ahead of time and create the necessary variables in a DATA step.

The general form of the function looks like

```
PROC REG < options > ;
  < label: > MODEL dependents=<regressors> < / options > ;
  BY variables ;
  FREQ variable ;
  ID variables ;
  VAR variables ;
  WEIGHT variable ;
  ADD variables ;
  DELETE variables ;
```

```
< label: > MTEST <equation, ... ,equation> < / options > ;
OUTPUT < OUT=SAS-data-set > keyword=names
    < ... keyword=names > ;
PAINT <condition | ALLOBS>
    < / options > | < STATUS | UNDO> ;
PLOT <yvariable*xvariable> <=symbol>
    < ...yvariable*xvariable> <=symbol> < / options > ;
PRINT < options > < ANOVA > < MODELDATA > ;
REFIT;
RESTRICT equation, ... ,equation ;
REWEIGHT <condition | ALLOBS>
      < / options > | < STATUS | UNDO> ;
< label: > TEST equation,<, ...,equation> < / option > ;
```

While there are many possible statements, most analyzes will only involve a few of these. The important statements are

- < label: > MODEL dependents=<regressors> < / options >: Describes the model to be fit and the summaries to be displayed.

A simple example showing the default output is

```
PROC REG DATA=shingles;
  simple_ex: MODEL sales = promotion accounts;
```

which gives the default output

```
Roofing Shingle Sales

The REG Procedure
Model: simple_ex
Dependent Variable: sales

Number of Observations Read          49
Number of Observations Used          49
```

Analysis of Variance

Source	DF	Sum of Squares	Mean Square	F Value	Pr > F
Model	2	199131	99565	43.01	<.0001
Error	46	106481	2314.80167		
Corrected Total	48	305612			

Root MSE	48.11239	R-Square	0.6516
Dependent Mean	178.61837	Adj R-Sq	0.6364
Coeff Var	26.93586		

Parameter Estimates

Variable	DF	Parameter Estimate	Standard Error	t Value	Pr > |t|
Intercept	1	-68.11980	34.54693	-1.97	0.0547
promotion	1	1.36726	4.49513	0.30	0.7624
accounts	1	4.53993	0.49460	9.18	<.0001

However we can get much more output. To get more you can use given the `ALL` option to the model statement as with

```
PROC REG DATA=shingles;
  simple_ex: MODEL sales = promotion accounts / ALL;
```

while it won't give all the possible output, it gives alot to stuff that people want. This includes in the default output plus

– `ACOV`:asymptotic covariance matrix assuming heteroscedasticity
– `CLB`: Confidence intervals for βs

- `CLI`: Prediction interval for predicted value
- `CLM`: Confidence interval for mean response
- `CORRB` & `COVB`: Correlation and covariance matrices for $\hat{\beta}$
- `I` & `XPX`: $(X^T X)^{-1}$ and $X^T X$
- `P` & `R`: Predicted values and residuals
- `PCORR1` & `PCORR2`: Squared partial correlation coefficients using Type I and Type II sums of squares
- `SCORR1` & `SCORR2`: Squared semi-partial correlation coefficients using Type I and Type II sums of squares
- `SEQB`: sequence of parameter estimates during selection process
- `SPEC`: Tests that first and second moments of model are correctly specified
- `SS1` & `SS2`: Sequential and partial sums of squares
- `STB`: Standardized parameter estimates.
- `TOL` & `VIF`: Tolerance and Variance Inflation Factors, two measures of the effect of multicollinearity on parameter estimation. Note that $\texttt{VIF} = 1/\texttt{TOL}$.

Here is a reduced version of the output with the `ALL` option

Roofing Shingle Sales
The REG Procedure
Model: simple_all

Model Crossproducts X'X X'Y Y'Y

Variable	Intercept	promotion	accounts	sales
Intercept	49	269.2	2582	8752.3
promotion	269.2	1594.94	14302.2	48773.86
accounts	2582	14302.2	145636	504847
sales	8752.3	48773.86	504847	1868933.41

X'X Inverse, Parameter Estimates, and SSE

Variable	Intercept	promotion	accounts	sales
Intercept	0.5155906285	-0.042338952	-0.004983073	-68.11979754
promotion	-0.042338952	0.0087291181	-0.000106611	1.3672644591
accounts	-0.004983073	-0.000106611	0.0001056818	4.5399312499
sales	-68.11979754	1.3672644591	4.5399312499	106480.87701

Parameter Estimates

Variable	DF	Parameter Estimate	Standard Error	t Value	Pr > |t|	Type I SS
Intercept	1	-68.11980	34.54693	-1.97	0.0547	1563322
promotion	1	1.36726	4.49513	0.30	0.7624	4102.29658
accounts	1	4.53993	0.49460	9.18	<.0001	195029

Parameter Estimates

Variable	DF	Type II SS	Standardized Estimate	Squared Semi-partial Corr Type I	Squared Partial Corr Type I
Intercept	1	8999.98286	0	.	.
promotion	1	214.15819	0.02664	0.01342	0.01342
accounts	1	195029	0.80382	0.63816	0.64684

Parameter Estimates

Variable	DF	Squared Semi-partial Corr Type II	Squared Partial Corr Type II	Tolerance	Variance Inflation
Intercept	1	.	.	.	0
promotion	1	0.00070075	0.00201	0.98768	1.01247
accounts	1	0.63816	0.64684	0.98768	1.01247

Parameter Estimates

Variable	DF	95% Confidence Limits	
Intercept	1	-137.65915	1.41956
promotion	1	-7.68096	10.41549
accounts	1	3.54435	5.53552

Covariance of Estimates

Variable	Intercept	promotion	accounts
Intercept	1193.4900499	-98.00627776	-11.53482603
promotion	-98.00627776	20.206177118	-0.246783606
accounts	-11.53482603	-0.246783606	0.244632309

Correlation of Estimates

Variable	Intercept	promotion	accounts
Intercept	1.0000	-0.6311	-0.6751
promotion	-0.6311	1.0000	-0.1110
accounts	-0.6751	-0.1110	1.0000

Sequential Parameter Estimates

Intercept	promotion	accounts
178.618367	0	0
145.945619	5.947120	0
-68.119798	1.367264	4.539931

Consistent Covariance of Estimates

Variable	Intercept	promotion	accounts
Intercept	869.36449744	-101.8146215	-6.899579295
promotion	-101.8146215	26.112551078	-0.515260596
accounts	-6.899579295	-0.515260596	0.1912422233

Test of First and Second
Moment Specification

DF	Chi-Square	Pr > ChiSq
5	10.13	0.0717

Output Statistics

Obs	Dependent Variable	Predicted Value	Std Error Mean Predict	95% CL Mean	
1	79.3000	80.1380	12.7451	54.4835	105.7925
2	200.1000	184.9946	15.2664	154.2649	215.7243
3	163.2000	246.9937	14.3708	218.0667	275.9207
4	200.1000	162.9786	13.0909	136.6280	189.3291

Output Statistics

Obs	95% CL Predict		Residual	Std Error Residual	Student Residual	-2-1 0 1 2
1	-20.0475	180.3236	-0.8380	46.394	-0.0181	| | |
2	83.3909	286.5983	15.1054	45.626	0.331	| | |
3	145.9206	348.0668	-83.7937	45.916	-1.825	| ***| |
4	62.6125	263.3446	37.1214	46.297	0.802	| |* |

Output Statistics

Obs	Cook's D
1	0.000
2	0.004
3	0.109
4	0.017

```
Sum of Residuals                               0
Sum of Squared Residuals                  106481
Predicted Residual SS (PRESS)             122344
```

Note that any of the options included in ALL can be asked for individually. Other options for output that aren't part of ALL include

- COLLIN: Prints collinearity diagnostics
- DW & DWPROB: Displays Durbin-Watson test for first-order autocorrelation on the residuals. Only appropriate when observations are ordered in time
- INFLUENCE: Prints influence statistics DFFITS and DFBETA which measures the effect of each observation on the fits of the model
- ALPHA=α: Sets significance value for confidence and prediction intervals and tests
- NOPRINT: Suppresses printing of output.

```
PROC REG DATA=shingles;
  simple_select: MODEL sales = promotion accounts /
    COLLIN INFLUENCE;
```

Collinearity Diagnostics

		Condition	-------Proportion of Variation-------		
Number	Eigenvalue	Index	Intercept	promotion	accounts
1	2.91200	1.00000	0.00459	0.00816	0.00740
2	0.06170	6.86979	0.00144	0.62075	0.48911
3	0.02630	10.52309	0.99397	0.37109	0.50349

Output Statistics

Obs	Residual	RStudent	Hat Diag H	Cov Ratio	DFFITS
1	-0.8380	-0.0179	0.0702	1.1487	-0.0049
2	15.1054	0.3278	0.1007	1.1793	0.1097
3	-83.7937	-1.8741	0.0892	0.9361	-0.5866
4	37.1214	0.7986	0.0740	1.1059	0.2258

Output Statistics

	--------------	DFBETAS	-------------
Obs	Intercept	promotion	accounts
1	-0.0033	-0.0005	0.0041
2	0.0653	-0.0976	0.0189
3	0.4293	-0.4277	-0.2378
4	0.1611	-0.1908	-0.0015

- ADD variables & DELETE variables:

  It is possible to build models sequentially by either adding or deleting variables from the current model. For example

```
PROC REG DATA=shingles;
  VAR promotion accounts brands potential;
  simple_ex: MODEL sales = promotion accounts;
  RUN;
  ADD potential;
  PRINT;
  RUN;
```

  This adds the variable potential to the model with promotion and accounts.

  Any variable listed in an ADD statement must be listed either in the original MODEL statement or in the VAR statement. An ADD statement will not print any output. That must be requested with a PRINT statement, which takes the same output options as MODEL.

The REG Procedure
Model: simple_ex.1
Dependent Variable: sales

Number of Observations Read	49
Number of Observations Used	49

Analysis of Variance

Source	DF	Sum of Squares	Mean Square	F Value	Pr > F
Model	3	200761	66920	28.72	<.0001
Error	45	104851	2330.01589		
Corrected Total	48	305612			

Root MSE	48.27024	R-Square	0.6569
Dependent Mean	178.61837	Adj R-Sq	0.6340
Coeff Var	27.02423		

Parameter Estimates

Variable	DF	Parameter Estimate	Standard Error	t Value	Pr > |t|
Intercept	1	-71.17003	34.85158	-2.04	0.0470
promotion	1	1.52963	4.51405	0.34	0.7363
accounts	1	4.31724	0.56313	7.67	<.0001
potential	1	1.38925	1.66090	0.84	0.4073

PROC ANOVA

This PROC is used for balanced ANOVA designs, where each possible treatment combination has the same number of observations. The balance assumption is not needed for 1-way designs.

Today these analyzes are usually done in PROC GLM which does everything that ANOVA does, plus much more. However, for completeness lets take a quick look at this PROC.

The form of the function is

```
PROC ANOVA < options > ;
  CLASS variables < / option > ;
  MODEL dependents=effects < / options > ;
  ABSORB variables ;
  BY variables ;
  FREQ variable ;
  MANOVA < test-options >< / detail-options > ;
  MEANS effects < / options > ;
```

```
REPEATED factor-specification < / options > ;
TEST < H=effects > E=effect ;
```

The important options are

- `CLASS`: Defines variables to be predictive factors and must occur before the `MODEL` statement.

- `MODEL dependents=effects < / options > ;`: Describes the model to be fit.

  The structure of describing models in **SAS** is similar to **S**, but there are significant differences.

  – To indicate an interaction, use a `*`. For example, `A*B` is the A*B interaction.
  – To indicate an interaction and all lower order effects, use `|`. For example `A | B` is equivalent to `A B A*B`.

- `MEANS effects < / options >`: Post hoc analysis of means. Examines difference of means based on different assumptions of mean structure. Common choices include `TUKEY` (all pairwise comparisons), `DUNNETT` (each vs control), `BON` (Bonferroni), `SCHEFFE` (all contrasts), plus many more.

- `TEST`: Tests of specific effects. Useful for models with random effects or more complicated design structure, such as a split plot design, when MSE is not the correct denominator for F test.

- `REPEATED`: Analysis of repeated measures designs based on multiple dependent variables.

An example of a 1-way ANOVA with this `PROC` is

```
PROC ANOVA;
  CLASS brand;    /* declare brand to be a factor */
  MODEL invtime = brand;
  MEANS brand;
  MEANS brand / dunnett('Butter');
```

Margarine Experiment

The ANOVA Procedure

Class Level Information

Class	Levels	Values
brand	4	Brand 1 Brand 2 Brand 3 Butter

Number of Observations Read	40
Number of Observations Used	40

Dependent Variable: invtime

Source	DF	Sum of Squares	Mean Square	F Value	Pr > F
Model	3	0.00001782	0.00000594	208.33	<.0001

Error	36	0.00000103	0.00000003
Corrected Total	39	0.00001884	

R-Square	Coeff Var	Root MSE	invtime Mean
0.945537	3.294115	0.000169	0.005125

Source	DF	Anova SS	Mean Square	F Value	Pr > F
brand	3	0.00001782	0.00000594	208.33	<.0001

Dunnett's t Tests for invtime

NOTE: This test controls the Type I experimentwise error for comparisons of all treatments against a control.

Alpha	0.05
Error Degrees of Freedom	36
Error Mean Square	2.851E-8
Critical Value of Dunnett's t	2.45216
Minimum Significant Difference	0.0002

Comparisons significant at the 0.05 level are indicated by ***.

brand Comparison	Difference Between Means	Simultaneous 95% Confidence Limits		
Brand 3 - Butter	0.00104353	0.00085838	0.00122869	***
Brand 1 - Butter	0.00087997	0.00069482	0.00106513	***
Brand 2 - Butter	-.00059797	-.00078313	-.00041282	***

So this analysis indicates that all 3 margarines are significantly different from butter, though the melting rates appear faster for margarines 1 and 3 and slower for margarine 2.

PROC GLM

This procedure fits normal general linear models. This includes

- Simple regression
- Multiple regression
- Analysis of variance (ANOVA), especially for unbalanced data
- Analysis of covariance
- Response-surface models
- Weighted regression
- Polynomial regression

- Partial correlation
- Multivariate analysis of variance (MANOVA)
- Repeated measures analysis of variance

This is a more general procedure that others available. While often this will be the route you want to go, some of the more specific procedures will be required for some analyzes. For example, many of the regression diagnostics are only available in `PROC REG`.

The general structure of the procedure is

```
PROC GLM < options > ;
  CLASS variables < / option > ;
  MODEL dependents=independents < / options > ;
  ABSORB variables ;
  BY variables ;
  FREQ variable ;
```

```
ID variables ;
WEIGHT variable ;
CONTRAST 'label' effect values < ... effect values >
           < / options > ;
ESTIMATE 'label' effect values < ... effect values >
           < / options > ;
LSMEANS effects < / options > ;
MANOVA < test-options >< / detail-options > ;
MEANS effects < / options > ;
OUTPUT <OUT=SAS-data-set >
    keyword=names < ... keyword=names > < / option > ;
RANDOM effects < / options > ;
REPEATED factor-specification < / options > ;
TEST < H=effects > E=effect < / options > ;
```

The important options are

- `CLASS`: Declares variables to the categorical like in `PROC ANOVA`

- MODEL dependents=independents < / options >: States the model. Unlike PROC REG, there can only be one MODEL statement in the PROC. Works the same way as PROC ANOVA so interactions can be declared on the fly. For example

```
MODEL z = x y x*y ; /* fits beta_0 + beta_1 x + beta_2 y
                              + beta_3 xy + e */
MODEL z = x x*x ;   /* fits beta_0 + beta_1 x
                              + beta_2 x^2 + e */

CLASS A;

MODEL z = x A x*A ; /* fits a different regression line in x
                       for each level of A  */
```

For example, a two way ANOVA can be written in a couple of ways

```
PROC GLM DATA = shingles2;
  CLASS potentcat training;
  MODEL sales = potentcat * training;

RUN;

PROC GLM DATA = shingles2;
  CLASS potentcat training;
  MODEL sales = potentcat training potentcat * training;

RUN;
```

Roofing Shingle Sales

The GLM Procedure

Class Level Information

Class	Levels	Values
potentcat	3	High Low Moderate
training	2	FALSE TRUE

Number of Observations Read	49
Number of Observations Used	49

Roofing Shingle Sales

The GLM Procedure

Dependent Variable: sales

Source	DF	Sum of Squares	Mean Square	F Value	Pr > F
Model	5	57349.7076	11469.9415	1.99	0.0999
Error	43	248262.1658	5773.5387		
Corrected Total	48	305611.8735			

R-Square	Coeff Var	Root MSE	sales Mean
0.187655	42.53975	75.98381	178.6184

Source	DF	Type I SS	Mean Square	F Value	Pr > F
potentcat*training	5	57349.70764	11469.94153	1.99	0.0999

Source	DF	Type III SS	Mean Square	F Value	Pr > F
potentcat*training	5	57349.70764	11469.94153	1.99	0.0999

The second form of the describing the model gives

```
Roofing Shingle Sales

The GLM Procedure

          Class Level Information

Class          Levels    Values

potentcat           3    High Low Moderate

training            2    FALSE TRUE


Number of Observations Read          49
Number of Observations Used          49
```

Roofing Shingle Sales

The GLM Procedure

Dependent Variable: sales

Source	DF	Sum of Squares	Mean Square	F Value	Pr > F
Model	5	57349.7076	11469.9415	1.99	0.0999
Error	43	248262.1658	5773.5387		
Corrected Total	48	305611.8735			

R-Square	Coeff Var	Root MSE	sales Mean
0.187655	42.53975	75.98381	178.6184

Source	DF	Type I SS	Mean Square	F Value	Pr > F
potentcat	2	51036.96071	25518.48035	4.42	0.0180
training	1	4130.54415	4130.54415	0.72	0.4023
potentcat*training	2	2182.20278	1091.10139	0.19	0.8285

Source	DF	Type III SS	Mean Square	F Value	Pr > F
potentcat	2	50873.93345	25436.96672	4.41	0.0182
training	1	6278.49158	6278.49158	1.09	0.3029
potentcat*training	2	2182.20278	1091.10139	0.19	0.8285

In the default output, if there is a class variable, the parameter estimates are not automatically printed. To display this you must add the `SOLUTIONS` options to the model statement. For example compare the output from

```
PROC GLM DATA = shingles2;
  CLASS potentcat;
  MODEL sales = potentcat accounts brands;

PROC GLM DATA = shingles2;
  CLASS potentcat;
  MODEL sales = potentcat accounts brands / SOLUTION;
```

The first version gives:

The GLM Procedure

Dependent Variable: sales

Source	DF	Sum of Squares	Mean Square	F Value	Pr > F
Model	4	301713.0691	75428.2673	851.25	<.0001
Error	44	3898.8044	88.6092		
Corrected Total	48	305611.8735			

R-Square	Coeff Var	Root MSE	sales Mean
0.987243	5.270032	9.413245	178.6184

Source	DF	Type I SS	Mean Square	F Value	Pr > F
potentcat	2	51036.9607	25518.4804	287.99	<.0001
accounts	1	151731.9730	151731.9730	1712.37	<.0001
brands	1	98944.1354	98944.1354	1116.64	<.0001

Source	DF	Type III SS	Mean Square	F Value	Pr > F
potentcat	2	527.45092	263.72546	2.98	0.0613
accounts	1	87909.24275	87909.24275	992.10	<.0001
brands	1	98944.13544	98944.13544	1116.64	<.0001

The second version has all this output plus the following section

```
                                                   Standard
Parameter                    Estimate                 Error    t Value    Pr > |t|

Intercept                 181.1492619 B          8.96814304      20.20      <.0001
potentcat High             -0.2614770 B          3.89458046      -0.07      0.9468
potentcat Low              -8.7367963 B          3.61246666      -2.42      0.0198
potentcat Moderate          0.0000000 B           .                .         .
accounts                    3.4886595            0.11075942      31.50      <.0001
brands                    -20.6638084            0.61837898     -33.42      <.0001

NOTE: The X'X matrix has been found to be singular, and a generalized
      inverse was used to solve the normal equations.  Terms whose
      estimates are followed by the letter 'B' are not uniquely estimable.
```

The reason that **SAS** doesn't always give the parameter estimates is given in the note. As discussed earlier, this is an overparameterized model. There are really are only 3 parameters involving intercepts for different levels for potentcat. **SAS** picks one possible solution by setting the β for one level to zero.

One other way to handle this would be to fit a no intercept model in this case, which can be done by

```
PROC GLM DATA = shingles2;
  CLASS potentcat;
  MODEL sales = potentcat accounts brands / NOINT SOLUTION;
```

which gives the output

```
Dependent Variable: sales

                                    Sum of
Source                  DF         Squares    Mean Square   F Value   Pr > F

Model                    5     1865034.606     373006.921   4209.57   <.0001

Error                   44        3898.804         88.609

Uncorrected Total       49     1868933.410
```

R-Square	Coeff Var	Root MSE	sales Mean
0.987243	5.270032	9.413245	178.6184

Source	DF	Type I SS	Mean Square	F Value	Pr > F
potentcat	3	1614358.497	538119.499	6072.95	<.0001
accounts	1	151731.973	151731.973	1712.37	<.0001
brands	1	98944.135	98944.135	1116.64	<.0001

Source	DF	Type III SS	Mean Square	F Value	Pr > F
potentcat	3	37185.82841	12395.27614	139.89	<.0001
accounts	1	87909.24275	87909.24275	992.10	<.0001
brands	1	98944.13544	98944.13544	1116.64	<.0001

Parameter	Estimate	Standard Error	t Value	Pr > |t|
potentcat High	180.8877849	10.33514583	17.50	<.0001
potentcat Low	172.4124657	9.46295192	18.22	<.0001
potentcat Moderate	181.1492619	8.96814304	20.20	<.0001
accounts	3.4886595	0.11075942	31.50	<.0001
brands	-20.6638084	0.61837898	-33.42	<.0001

Note that you need that you need to be careful in this situation as it changes some of the standard tests. For example the F test for `potentcat` examines the null hypothesis

$$H_0 : \beta_1 = \beta_2 = \beta_3 = 0$$

not

$$H_0 : \beta_1 = \beta_2 = \beta_3 = \text{arbitrary value}$$

Other useful options to the `MODEL` statement are

- `ALPHA =` α: Sets the alpha level for confidence intervals
- `CLM`: Confidence intervals for mean response
- `CLI`: Predictions intervals. Ignored if `CLM` option is given
- `CLPARM`: Confidence intervals for β.
- `P`: Prints predicted and residual values

- `CONTRAST 'label' effect values < ... effect values > < / options >`:

- `ESTIMATE 'label' effect values < ... effect values > < / options >`:

  These two statements examine $L\beta$, linear combinations of the parameters, which is estimated by $L\hat{\beta}$. For example this can be used to predict the response variable for a given combination of the predictor variables. For example

```
PROC GLM DATA = shingles2;
  MODEL sales = accounts brands / CLPARM ;
  ESTIMATE 'Accounts = 10, Brands = 7'
            intercept 1 accounts 10 brands 7;
```

which gives

```
                                                    Standard
Parameter                          Estimate            Error    t Value    Pr > |t|

Accounts = 10, Brands = 7        68.6247175       5.19582917      13.21      <.0001

Parameter                        95% Confidence Limits

Accounts = 10, Brands = 7        58.1660558     79.0833792
```

Confidence intervals are only included if the CLPARM option is given in the MODEL statement.

To state the linear combination of interest, you need to list the variable followed by level you wish that variable to take. To included the intercept in the model you need to add `intercept 1` to the statement.

When you have class variable in the model it will generate one parameter for each level of the variable. For example, `potentcat` leads to three αs say. So when including this variable in an `ESTIMATE` or `CONTRAST` statement, you need to give three values. For example

```
ESTIMATE 'Moderate - 15 - 10'
  intercept 1 potentcat 0 0 1 accounts 15 brands 10;
```

The order for the 3 levels matches with the the levels given by output of the `CLASS` statement. In this case the order is `High`, `Low`, and `Moderate`.

Note that all levels are not required to be given. For example, suppose I want to estimate the difference in response between `Low` and `High` level regions while keeping the levels of the other variables fixed, I can do

```
PROC GLM DATA = shingles2;
  CLASS potentcat;
  MODEL sales = potentcat accounts brands / SOLUTION;
  ESTIMATE 'High vs Low' potentcat 1 -1 0;
```

which gives

```
                                       Standard
Parameter               Estimate          Error    t Value    Pr > |t|

High vs Low            8.4753192     4.87276856       1.74      0.0890
```

You need to be careful sometimes when describing the vector L, as it must correspond to an estimable function of the parameters.

For example, consider the two examples

```
ESTIMATE 'Moderate - 15 - 10'
  intercept 1 potentcat 0 0 1 accounts 15 brands 10;
ESTIMATE 'Problem' potentcat 0 0 1 accounts 15 brands 10;
```

SAS gives the output

Parameter	Estimate	Standard Error	t Value	Pr > |t|
High vs Low	8.4753192	4.87276856	1.74	0.0890

There is no output for the effect labeled `Problem`. Checking the log file we see

```
NOTE: Problem is not estimable.
```

The again relates to the problem being overparametrized. The estimate $L\hat{\beta}$ in this case depends on how the model is parametrized. However for the other effect, this isn't a problem. $L\hat{\beta}$ is the same under any consistent parameterization.

`CONTRAST` examines hypotheses of the form

$$H_0 : L\beta = 0 \qquad \text{vs} \qquad H_A : L\beta \neq 0$$

via an F test.

For example,

```
CONTRAST 'High vs Low' potentcat 1 -1 0;
CONTRAST 'High vs Moderate' potentcat 1 0 -1;
CONTRAST 'Moderate vs Low' potentcat 0 -1 1;
```

gives

Contrast	DF	Contrast SS	Mean Square	F Value	Pr > F
High vs Low	1	268.0645158	268.0645158	3.03	0.0890
High vs Moderate	1	0.3994151	0.3994151	0.00	0.9468
Moderate vs Low	1	518.2931717	518.2931717	5.85	0.0198

Note that this gives equivalent information to ESTIMATE, except as an F test instead of a t test.

Contrast	DF	Contrast SS	Mean Square	F Value	Pr > F
High vs Low	1	268.0645158	268.0645158	3.03	0.0890

Parameter	Estimate	Standard Error	t Value	Pr > |t|
High vs Low	8.4753192	4.87276856	1.74	0.0890

Note that output from ESTIMATE gives the same effective test statistic for this example, as it must.

Again you must be careful in that $L\beta$ is estimable. As long as you are dealing with contrasts of the class variables, this shouldn't be a problem.