Sassafras Manual

Contents

1 Introduction

Sassafras is a shell mode program for statistical analysis.

To build and run:

```
make
./sassafras infile
```

Infile is a text file that tells the program what to do. The syntax is a subset of SAS-Language. There are "data steps" and "procedure steps." Data steps get data into the program and procedure steps compute the results. A data step begins with the keyword *data* and a procedure step begins with the keyword *proc*.

Example

A die, which may be loaded, is tossed six times. The observed point values are one to six. Compute a 95% confidence interval for the true mean μ given the observed data.

```
data
input y
datalines
1
2
3
4
5
6
;
proc means clm
```

The following result is displayed.

```
            Variable     95% CLM MIN     95% CLM MAX
            Y                  1.537           5.463
```

Here is the same result using R.

```
> y = c(1,2,3,4,5,6)
> t.test(y)

        One Sample t-test

data:  y
t = 4.5826, df = 5, p-value = 0.005934
alternative hypothesis: true mean is not equal to 0
95 percent confidence interval:
 1.536686 5.463314
```

2 Data Step

A data step is used to get data into the program.

> data *name*
>
> infile "*filename*" dlm="*delims*" firstobs=*n*
>
> input *list*
>
> *var* = *expression*
>
> datalines

Notes

1. *name* is optional.

2. The `dlm` and `firstobs` settings are optional.

3. *delims* is a sequence of delimiter characters. The default is tab, comma, and space.

4. n is the starting input record number. Use `firstobs=2` to skip a header in the data file.

5. *list* is a list of variable names separated by spaces. For each categorical variable place a `$` after the variable name.

6. Optional *var = expression* statements create new vectors in the data set.

7. The `datalines` statement is followed by observational data. At the end of the data a blank line or a semicolon terminates the statement.

Example 1

The following example is a minimalist data step with in-line data.

```
data
input y
datalines
1
2
3
4
5
6
```

Example 2

Use `@@` at the end of an input statement to allow multiple values on an input line.

```
data
input y @@
datalines
1 2 3
4 5 6
```

Example 3

A dollar sign after an input variable indicates that the variable is categorical instead of numerical.

```
data
input trt $ y @@
datalines
A 6     A 0     A 2     A 8     A 11
A 4     A 13    A 1     A 8     A 0
B 0     B 2     B 3     B 1     B 18
B 4     B 14    B 9     B 1     B 9
C 13    C 10    C 18    C 5     C 23
C 12    C 5     C 16    C 1     C 20
```

Example 4

An infile statement is used to read data from a file.

```
data
input color $ x1 x2 x3 x4 x5 x6 x7 x8 x9 x10 x11 y
infile "wine.txt"
```

Example 5

Expressions in a data step create new data vectors. The following example creates Y2 which is the input vector Y squared element-wise.

```
data
input color $ x1 x2 x3 x4 x5 x6 x7 x8 x9 x10 x11 y
infile "wine.txt"
y2 = y ** 2
```

3   Anova Procedure

The anova procedure fits a classification model to data using ordinary least squares. The response variable must be numeric and the explanatory variables must be categorical.

```
proc anova data=name

model y = list

means list

means list / lsd ttest alpha=value
```

Notes

1. `data=`*name* is optional. The default is data from the most recent data step.

2. y is the response variable which must be numeric.

3. *list* is one or more explanatory variables separated by spaces. The explanatory variables must be categorical. Interaction terms are specified using the syntax `A*B`.

4. The means statement can include one or more of the following options.

`lsd`	Compare treatment means using least significance difference
`ttest`	Compare treatment means using two sample t-test
`alpha`	Set the level of significance. Default is 0.05.

Example

```
data
input trt $ y @@
datalines
A 6     A 0     A 2     A 8     A 11
A 4     A 13    A 1     A 8     A 0
B 0     B 2     B 3     B 1     B 18
B 4     B 14    B 9     B 1     B 9
C 13    C 10    C 18    C 5     C 23
C 12    C 5     C 16    C 1     C 20

proc anova
model y = trt
means trt / lsd ttest
```

The following result is displayed.

```
                    Analysis of Variance

   Source      DF     Sum of Squares     Mean Square     F Value     Pr > F
   Model       2        293.60000000     146.80000000       3.98     0.0305
```

```
   Error       27        995.10000000        36.85555556
   Total       29       1288.70000000
```

```
              R-Square      Coeff Var      Root MSE      Y Mean
              0.227826      76.846553      6.070878      7.900000
```

```
   Source      DF          Anova SS       Mean Square    F Value    Pr > F
   TRT          2        293.60000000     146.80000000      3.98    0.0305
```

Mean Response

```
       TRT      N       Mean Y      95% CI MIN      95% CI MAX
       A        10      5.300000     1.360938        9.239062
       B        10      6.100000     2.160938       10.039062
       C        10     12.300000     8.360938       16.239062
```

Least Significant Difference Test

TRT	TRT	Delta Y	95% CI MIN	95% CI MAX	t Value	Pr > |t|	
A	B	-0.800000	-6.370676	4.770676	-0.29	0.7705	
A	C	-7.000000	-12.570676	-1.429324	-2.58	0.0157	*
B	A	0.800000	-4.770676	6.370676	0.29	0.7705	
B	C	-6.200000	-11.770676	-0.629324	-2.28	0.0305	*
C	A	7.000000	1.429324	12.570676	2.58	0.0157	*
C	B	6.200000	0.629324	11.770676	2.28	0.0305	*

Two Sample t-Test

TRT	TRT	Delta Y	95% CI MIN	95% CI MAX	t Value	Pr > |t|	
A	B	-0.800000	-5.922306	4.322306	-0.33	0.7466	
A	C	-7.000000	-12.664270	-1.335730	-2.60	0.0182	*
B	A	0.800000	-4.322306	5.922306	0.33	0.7466	
B	C	-6.200000	-12.467653	0.067653	-2.08	0.0523	
C	A	7.000000	1.335730	12.664270	2.60	0.0182	*
C	B	6.200000	-0.067653	12.467653	2.08	0.0523	

Mean response table

The confidence interval for a treatment mean is computed as follows.

$$\bar{y}_i \pm t(1 - \alpha/2, dfe) \cdot \sqrt{\frac{MSE}{n_i}}$$

Recall that MSE is an estimate of model variance. From the anova table

```
   Error       27        995.10000000        36.85555556
```

we obtain

$$MSE = 36.85555556$$
$$dfe = 27$$

Using R, the confidence interval for the mean of treatment A can be checked as follows.

```
> MSE = 36.8556
> dfe = 27
> t = qt(0.975,dfe)
> 5.3 - t * sqrt(MSE/10)
[1] 1.360934
> 5.3 + t * sqrt(MSE/10)
[1] 9.239066
```

Least significant difference test

The least significant difference of two means $\bar{y}_i$ and $\bar{y}_j$ is

$$LSD_{ij} = t(1 - \alpha/2, dfe) \cdot \sqrt{MSE \cdot \left(\frac{1}{n_i} + \frac{1}{n_j} \right)}$$

The corresponding confidence interval is

$$\bar{y}_i - \bar{y}_j \pm LSD_{ij}$$

Two sample t-test

The two sample t-test is computed as follows.

$$SSE = \widehat{Var}_i \cdot (n_i - 1) + \widehat{Var}_j \cdot (n_j - 1)$$
$$dfe = n_i + n_j - 2$$
$$MSE = \frac{SSE}{dfe}$$
$$SE = \sqrt{MSE \cdot \left(\frac{1}{n_i} + \frac{1}{n_j} \right)}$$
$$t^* = \frac{\bar{y}_i - \bar{y}_j}{SE}$$

SSE is the sum of squares error recovered from variance estimates, dfe is the degrees of freedom error, MSE is mean square error, SE is the standard error, and t^* is the test statistic. The confidence interval is

$$\bar{y}_i - \bar{y}_j \pm t(1 - \alpha/2, dfe) \cdot SE$$

The null hypothesis is that the two treatment means are equal.

$$H_0 : \bar{y}_i = \bar{y}_j$$

If $|t^*|$ is greater than the critical value $t(1 - \alpha/2, dfe)$, or equivalently, if the confidence interval does not cross zero, then reject H_0 and conclude that the treatment means are not equal. The following R session uses the above equations to duplicate the Sassafras result for treatments A and B.

```
> YA = c(6,0,2,8,11,4,13,1,8,0)
> YB = c(0,2,3,1,18,4,14,9,1,9)
> sse = var(YA) * (length(YA) - 1) + var(YB) * (length(YB) - 1)
> dfe = length(YA) + length(YB) - 2
> mse = sse / dfe
> se = sqrt(mse * (1 / length(YA) + 1 / length(YB)))
> t = (mean(YA) - mean(YB)) / se
> mean(YA) - mean(YB) - qt(0.975,dfe) * se
[1] -5.922307
> mean(YA) - mean(YB) + qt(0.975,dfe) * se
[1] 4.322307
> 2 * (1 - pt(abs(t),dfe))
[1] 0.746606
```

The same result is obtained with the t-test function.

```
> t.test(YA,YB,var.equal=TRUE)

        Two Sample t-test

data:  YA and YB
t = -0.3281, df = 18, p-value = 0.7466
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 -5.922307  4.322307
sample estimates:
mean of x mean of y
      5.3       6.1
```

4 Means Procedure

The means procedure prints statistics about a data set.

```
proc means data=name alpha=value maxdec=n  stats

var  list

class  list
```

Notes

1. The settings that follow the `means` keyword are optional. The settings can appear in any order.

2. If `data` is not specified then the default is data from the most recent data step.

3. `alpha` sets the level of significance. The default is 0.05.

4. `maxdec` sets the decimal precision in the output. n ranges from 0 to 8. The default is 3.

5. *stats* is a list of statistics keywords from the following table.

`clm`	Confidence limits of the mean
`max`	Maximum value
`mean`	Mean value
`min`	Minimum value
`n`	Number of observations
`range`	$\max - \min$
`std`	Standard deviation s
`stddev`	Another keyword for s
`stderr`	Standard error $s/\sqrt{n}$
`var`	Variance s^2

   If *stats* is not specified then the default list is `n mean std min max`.

6. The optional `var` statement specifies which variables to print. The default is all variables. Variable names in *list* are separated by spaces.

7. The optional `class` statement prints statistics for each level of the categorical variables in *list*. Variable names in *list* are separated by spaces.

Example 1

The following example reads in the wine[1] data set and shows the default action of proc means.

```
data wine
input color $ x1 x2 x3 x4 x5 x6 x7 x8 x9 x10 x11 y
```

---

[1]P. Cortez, A. Cerdeira, F. Almeida, T. Matos and J. Reis. *Modeling wine preferences by data mining from physicochemical properties.* In Decision Support Systems, Elsevier, 47(4):547-553, 2009.

```
infile "wine.txt"

proc means
```

The following result is displayed.

Variable	N	Mean	Std Dev	Minimum	Maximum
X1	6497	7.215	1.296	3.800	15.900
X2	6497	0.340	0.165	0.080	1.580
X3	6497	0.319	0.145	0.000	1.660
X4	6497	5.443	4.758	0.600	65.800
X5	6497	0.056	0.035	0.009	0.611
X6	6497	30.525	17.749	1.000	289.000
X7	6497	115.745	56.522	6.000	440.000
X8	6497	0.995	0.003	0.987	1.039
X9	6497	3.219	0.161	2.720	4.010
X10	6497	0.531	0.149	0.220	2.000
X11	6497	10.492	1.193	8.000	14.900
Y	6497	5.818	0.873	3.000	9.000

Example 2

The following example adds a var statement to show Y by itself. Also, the desired statistics are specified.

```
data wine
input color $ x1 x2 x3 x4 x5 x6 x7 x8 x9 x10 x11 y
infile "wine.txt"

proc means n mean clm
var y
```

The following result is displayed.

Variable	N	Mean	95% CLM MIN	95% CLM MAX
Y	6497	5.818	5.797	5.840

Example 3

The following example adds a class statement to show statistics for each wine color.

```
data wine
input color $ x1 x2 x3 x4 x5 x6 x7 x8 x9 x10 x11 y
infile "wine.txt"

proc means n mean clm
var y
class color
```

The following result is displayed.

```
COLOR      Variable         N      Mean     95% CLM MIN     95% CLM MAX
red        Y             1599     5.636           5.596           5.676
white      Y             4898     5.878           5.853           5.903
```

5　Print Procedure

The print procedure prints data in a data set.

　　proc print data=*name*

　　var *list*

Notes

1. **data=***name* is optional. The default is data from the most recent data step.

2. The optional **var** statement specifies which variables to print. The default is all variables. Variable names in *list* are separated by spaces.

Example

The following example reads a data set and prints it.

```
data
input trt $ y @@
datalines
A 6     A 0     A 2     A 8     A 11
A 4     A 13    A 1     A 8     A 0
B 0     B 2     B 3     B 1     B 18
B 4     B 14    B 9     B 1     B 9

proc print
```

The following result is displayed.

Obs	TRT	Y
1	A	6
2	A	0
3	A	2
4	A	8
5	A	11
6	A	4
7	A	13
8	A	1
9	A	8
10	A	0
11	B	0
12	B	2
13	B	3
14	B	1
15	B	18
16	B	4

17	B	14
18	B	9
19	B	1
20	B	9

6 Reg Procedure

The reg procedure fits a linear model to data using ordinary least squares. The response variable must be numeric. For models with no intercept, anova results will differ from R. This is because R switches to uncorrected sums of squares for models with no intercept.

proc reg data=*name*

model *y* = *list*

model *y* = *list* / noint

Notes

1. data=*name* is optional. The default is data from the most recent data step.

2. *y* is the response variable which must be numeric.

3. *list* is a list of explanatory variables separated by spaces. If functions of explanatory variables are required then they must be defined in the data step.

4. The noint option fits a linear model with no intercept term.

Example 1

The following example reads in the wine data set and fits a linear model with no intercept term.

```
data
input color $ x1 x2 x3 x4 x5 x6 x7 x8 x9 x10 x11 y
infile "wine.txt"

proc reg
model y = color x1 / noint
```

The following result is displayed.

```
                        Analysis of Variance

                DF     Sum of Squares     Mean Square     F Value     Pr > F
    Model        2           72.79210        36.39605       48.42     0.0000
    Error     6494         4880.89360         0.75160
    Total     6496         4953.68570


            Root MSE                 0.86695     R-Square     0.0147
            Dependent Mean           5.81838     Adj R-Sq     0.0144
            Coeff Var               14.90018


                        Parameter Estimates
```

	Estimate	Std Err	t Value	Pr > |t|
COLOR red	5.77309	0.08194	70.45	0.0000
COLOR white	5.99084	0.06628	90.39	0.0000
X1	-0.01647	0.00950	-1.73	0.0829

Example 2

The following exercise is from *Econometrics*[2]. Using data from a 1963 paper by Marc Nerlove, estimate parameters for the model

$$\log(\text{COST}) = \beta_0 + \beta_1 \log(\text{KWH}) + \beta_2 \log(\text{PL}) + \beta_3 \log(\text{PF}) + \beta_4 \log(\text{PK}) + \varepsilon$$

where COST is production cost, KWH is kilowatt hours, PL is price of labor, PF is price of fuel, and PK is price of capital.

```
data
infile "nerlove.txt"
input COST KWH PL PF PK
LCOST = log(COST)
LKWH = log(KWH)
LPL = log(PL)
LPF = log(PF)
LPK = log(PK)

proc reg
model LCOST = LKWH LPL LPF LPK
```

The following result is displayed.

Analysis of Variance

	DF	Sum of Squares	Mean Square	F Value	Pr > F
Model	4	269.51482	67.37870	437.69	0.0000
Error	140	21.55201	0.15394		
Total	144	291.06683			

Root MSE	0.39236	R-Square	0.9260
Dependent Mean	1.72466	Adj R-Sq	0.9238
Coeff Var	22.74969		

Parameter Estimates

	Estimate	Std Err	t Value	Pr > |t|
Intercept	-3.52650	1.77437	-1.99	0.0488
LKWH	0.72039	0.01747	41.24	0.0000
LPL	0.43634	0.29105	1.50	0.1361

[2]Hansen, Bruce E. *Econometrics.* www.ssc.wisc.edu/~bhansen

```
           LPF          0.42652      0.10037      4.25      0.0000
           LPK         -0.21989      0.33943     -0.65      0.5182
```

The following code can be pasted into R to obtain a similar result.

```
d = read.table("nerlove.txt")
lcost = log(d[,1])
lkwh = log(d[,2])
lpl = log(d[,3])
lpf = log(d[,4])
lpk = log(d[,5])
m = lm(lcost ~ lkwh + lpl + lpf + lpk)
summary(m)
```

The following result is displayed in R.

```
Call:
lm(formula = lcost ~ lkwh + lpl + lpf + lpk)

Residuals:
    Min       1Q   Median       3Q      Max
-0.97784 -0.23817 -0.01372  0.16031  1.81751

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept) -3.52650    1.77437  -1.987   0.0488 *
lkwh         0.72039    0.01747  41.244  < 2e-16 ***
lpl          0.43634    0.29105   1.499   0.1361
lpf          0.42652    0.10037   4.249 3.89e-05 ***
lpk         -0.21989    0.33943  -0.648   0.5182
---
Signif. codes:  0 *** 0.001 ** 0.01 * 0.05 . 0.1   1

Residual standard error: 0.3924 on 140 degrees of freedom
Multiple R-squared:  0.926,        Adjusted R-squared:  0.9238
F-statistic: 437.7 on 4 and 140 DF,  p-value: < 2.2e-16
```

Example 3

The following model uses the "trees" data set from R.

```
data
input Girth Height Volume
LG = log(Girth)
LH = log(Height)
LV = log(Volume)
datalines
  8.3     70    10.3
```

```
 8.6     65    10.3
 8.8     63    10.2
10.5     72    16.4
10.7     81    18.8
10.8     83    19.7
11.0     66    15.6
11.0     75    18.2
11.1     80    22.6
11.2     75    19.9
11.3     79    24.2
11.4     76    21.0
11.4     76    21.4
11.7     69    21.3
12.0     75    19.1
12.9     74    22.2
12.9     85    33.8
13.3     86    27.4
13.7     71    25.7
13.8     64    24.9
14.0     78    34.5
14.2     80    31.7
14.5     74    36.3
16.0     72    38.3
16.3     77    42.6
17.3     81    55.4
17.5     82    55.7
17.9     80    58.3
18.0     80    51.5
18.0     80    51.0
20.6     87    77.0
```

```
proc reg
model LV = LG LH
```

The following result is displayed.

```
                         Analysis of Variance

   Source      DF    Sum of Squares    Mean Square    F Value    Pr > F
   Model        2       1.53213547      0.76606773     613.19    0.0000
   Error       28       0.03498056      0.00124931
   Total       30       1.56711603


           Root MSE             0.03535    R-Square    0.9777
           Dependent Mean       1.42133    Adj R-Sq    0.9761
           Coeff Var            2.48679


                        Parameter Estimates
```

```
        Parameter      Estimate     Std Err     t Value     Pr > |t|
        (Intercept)    -2.88007     0.34734      -8.29       0.0000
        log(Girth)      1.98265     0.07501      26.43       0.0000
        log(Height)     1.11712     0.20444       5.46       0.0000
```

Let us see if the above parameters correspond to the volume of a cone given by

$$V = \frac{\pi}{12} d^2 h$$

where d is the diameter (girth) and h is the height of the cone. The model from the regression is

$$\log V = -2.88 + 1.98 \log d + 1.12 \log h$$

Take the antilog of both sides and obtain

$$V = 0.00132 \times d^{1.98} \times h^{1.12}$$

The exponents resemble the volume formula but the overall coefficient 0.00132 is two orders of magnitude smaller than $\pi/12 \approx 0.262$. It turns out the discrepancy is due to the units of measure. Girth is measured in inches while height and volume are measured in feet. To convert girth from inches to feet requires a factor of $1/12$. Hence the leading coefficient should be

$$\frac{\pi}{12} \times \frac{1}{144} \approx 0.00182$$

which is in the ballpark of 0.00132 from the regression model.

Let us compare the Reg results to R. The following block of code can be pasted directly into the R shell prompt.

```
d=log10(trees[,1])
h=log10(trees[,2])
V=log10(trees[,3])
m=lm(V~d+h)
summary(m)
```

This is the R result, which matches Reg.

```
Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept) -2.88007    0.34734  -8.292 5.06e-09 ***
d            1.98265    0.07501  26.432  < 2e-16 ***
h            1.11712    0.20444   5.464 7.81e-06 ***
---
Signif. codes:  0 *** 0.001 ** 0.01 * 0.05 . 0.1   1

Residual standard error: 0.03535 on 28 degrees of freedom
Multiple R-squared:  0.9777,      Adjusted R-squared:  0.9761
F-statistic: 613.2 on 2 and 28 DF,  p-value: < 2.2e-16
```

7 Review

Analysis of Variance

The components of an analysis of variance table are computed as follows.

	DF	SS	Mean Square	F-value	p-value
Model	$p-1$	SSR	$MSR = SSR/(p-1)$	$F^* = MSR/MSE$	$1 - F(F^*, p-1, n-p)$
Error	$n-p$	SSE	$MSE = SSE/(n-p)$		
Total	$n-1$	SST			

In the table, n is the number of observations and p is the number of model parameters including the intercept term if there is one. The sums of squares are computed as follows.

$$SSR = \sum (\hat{y}_i - \bar{y})^2$$
$$SSE = \sum (y_i - \hat{y}_i)^2$$
$$SST = \sum (y_i - \bar{y})^2$$

Recall that MSE is an estimate of model variance.

$$MSE = \hat{\sigma}^2$$

A simple way to model the response variable is to use the average $\bar{y}$. The p-value above indicates whether or not the regression model is better than $\bar{y}$. The null hypothesis is that the regression model is no better than the average, that is

$$H_0 : SST = SSE$$

The test for H_0 is known as an omnibus test because an equivalent hypothesis is

$$H_0 : \beta_1 = \beta_2 = \cdots = \beta_{p-1} = 0$$

Under H_0 we have $SSR = 0$ hence another equivalent hypothesis is

$$H_0 : F^* = 0$$

The test statistic F^* is used because it has a well-known distribution. Recall that the p-value is (loosely) the probability that H_0 is true. Hence for small p-values, reject H_0 and conclude that the regression model is better than $\bar{y}$.

Confidence interval of the mean

The confidence interval of the mean is

$$\bar{x} \pm t_{1-\alpha/2, n-1} \frac{s}{\sqrt{n}}$$

where $\bar{x}$ is the observed mean, s is the observed standard deviation, n is the number of observations, and $t_{1-\alpha/2, n-1}$ is the quantile function. In R, the confidence interval of the mean of `1:10` can be computed as follows.

```
> x = 1:10
> n = length(x)
> alpha = 0.05
> mean(x) - qt(1-alpha/2,n-1) * sd(x)/sqrt(n)
[1] 3.334149
> mean(x) + qt(1-alpha/2,n-1) * sd(x)/sqrt(n)
[1] 7.665851
```

Alternatively, the `t.test` function can be used.

```
> t.test(1:10)

        One Sample t-test

data:  1:10
t = 5.7446, df = 9, p-value = 0.0002782
alternative hypothesis: true mean is not equal to 0
95 percent confidence interval:
 3.334149 7.665851
sample estimates:
mean of x
      5.5
```

Recall that the quantile function is the inverse of the cumulative distribution function. Let F be the cumulative distribution function. Then

$$F(t_{1-\alpha/2,n-1}) = 1 - \alpha/2$$

For example, in R we have

```
> t = qt(0.975,8)
> t
[1] 2.306004
> pt(t,8)
[1] 0.975
```

8 Example infiles

8.1 alfalfa-demo.txt

```
* Compare yields of four types of alfalfa.
* This is a randomized complete block design.

* References
* 1. "Statistical Design" by G. Casella, p. 97.
* 2. http://www.stat.ufl.edu/~casella/StatDesign/WebDataSets/Alfalfa.txt

data
input Variety $ Block $ Rep $ Yield
datalines
Ladak     1    1     3.1
Ladak     2    1     4.06
Ladak     3    1     4.73
Ladak     4    1     3.1
Ladak     1    2     3.25
Ladak     2    2     4.26
Ladak     3    2     4.71
Ladak     4    2     4.21
Ladak     1    3     3.86
Ladak     2    3     4.53
Ladak     3    3     5.26
Ladak     4    3     3.84
Narrag    1    1     4.65
Narrag    2    1     5.64
Narrag    3    1     4.94
Narrag    4    1     5.38
Narrag    1    2     5.46
Narrag    2    2     5.48
Narrag    3    2     5.26
Narrag    4    2     5.68
Narrag    1    3     4.21
Narrag    2    3     5.09
Narrag    3    3     5.8
Narrag    4    3     5.82
DuPuits   1    1     5.47
DuPuits   2    1     5.62
DuPuits   3    1     6.71
DuPuits   4    1     6.87
DuPuits   1    2     6.41
DuPuits   2    2     6.3
DuPuits   3    2     6.96
DuPuits   4    2     6.28
DuPuits   1    3     5.57
DuPuits   2    3     6.46
DuPuits   3    3     5.92
DuPuits   4    3     6.46
Flamand   1    1     6.85
Flamand   2    1     6.33
Flamand   3    1     6.88
Flamand   4    1     6.23
Flamand   1    2     6.34
```

```
Flamand  2    2    5.83
Flamand  3    2    6.59
Flamand  4    2    6.52
Flamand  1    3    5.45
Flamand  2    3    4.33
Flamand  3    3    6.06
Flamand  4    3    6.81


proc anova
model Yield = Block Variety Block*Variety
means Variety / lsd
```

8.2   corrosion-demo.txt

```
* Reference
* "Split-Plot Designs: What, Why, and How" by Bradley Jones, Christopher L. Nachtsheim
* jmp.info/software/pdf/30612.pdf

* W - Whole plot (furnace)
* T - Temperature
* C - Coating
* Y - Corrosion resistance of steel bars

data
input W $ T $ C $ Y
datalines
1 360 C2 73
1 360 C3 83
1 360 C1 67
1 360 C4 89
2 370 C1 65
2 370 C3 87
2 370 C4 86
2 370 C2 91
3 380 C3 147
3 380 C1 155
3 380 C2 127
3 380 C4 212
4 380 C4 153
4 380 C3 90
4 380 C2 100
4 380 C1 108
5 370 C4 150
5 370 C1 140
5 370 C3 121
5 370 C2 142
6 360 C1 33
6 360 C4 54
6 360 C2 8
6 360 C3 46

proc anova
model Y = T C T*C W
means C / lsd

* Proc anova computes F(T) = MS(T)/MSE = 106.47
* Correct value is F(T) = MS(T)/MS(W) = 2.75 and
* corresponding p value is 0.209
```

8.3 diet-demo.txt

* This is a split plot design.

* References
* 1. "Statistical Design" by G. Casella, p. 171.
* 2. http://www.stat.ufl.edu/~casella/StatDesign/WebDataSets/Diet.txt

```
data
input Diet $ Subject $ Time $ BP
datalines
1    1    AM    123
1    1    PM    135
1    2    AM    120
1    2    PM    136
1    3    AM    122
1    3    PM    129
2    4    AM    117
2    4    PM    139
2    5    AM    125
2    5    PM    136
2    6    AM    122
2    6    PM    142
3    7    AM    114
3    7    PM    123
3    8    AM    109
3    8    PM    132
3    9    AM    115
3    9    PM    132
4    10   AM    140
4    10   PM    150
4    11   AM    141
4    11   PM    147
4    12   AM    138
4    12   PM    154


proc anova
model BP = Diet Subject*Diet Time Time*Diet
means Diet / lsd
```

* Subject is nested within Diet.

* Replacing Subject*Diet with just Subject yields
* the same result because each subject is uniquely
* identified.

* Anova in the book includes Time*Subject*Diet.
* It turns out that
* MS(Time*Subject*Diet) = MSE = 16.17

* Proc anova computes F(*) = MS(*)/MSE
* Correct F(Diet) = MS(Diet)/MS(Subject*Diet)
* Hence F(Diet) = 624.49/7.33 = 85.20

8.4 dry-weight-demo.txt

* Compare dry weights of geraniums grown with three types of fertilizer.

* References
* 1. "Statistical Design" by G. Casella, p. 2.
* 2. http://www.stat.ufl.edu/~casella/StatDesign/WebDataSets/DryWeight.txt

```
data
input Fert $ DryWeight
datalines
A       1.02
A       0.79
A       1
A       0.59
A       0.97
B       1
B       1.21
B       1.22
B       0.96
B       0.79
C       0.99
C       1.36
C       1.22
C       1.12
C       1.17

proc anova
model DryWeight = Fert
means Fert / lsd
```

8.5 fish-tank-demo.txt

* Compare fish weight gains for three types of diets.
* Tanks are nested within diets.

* References
* 1. "Statistical Design" by G. Casella, p. 6.
* 2. http://www.stat.ufl.edu/~casella/StatDesign/WebDataSets/FishTank.txt

```
data
input Diet $ Tank $ WtGain
datalines
```

Diet	Tank	WtGain
1	1	9.759
1	1	7.399
1	1	−0.209
1	1	2.204
1	1	0.267
1	1	3.002
1	2	8.031
1	2	6.25
1	2	0.134
1	2	4.594
1	2	4.414
1	2	9.816
1	3	−2.23
1	3	9.69
1	3	−2.239
1	3	9.499
1	3	6.927
1	3	3.449
1	4	4.274
1	4	7.52
1	4	12.141
1	4	4.828
1	4	3.391
1	4	3.621
2	5	21.819
2	5	6.503
2	5	31.596
2	5	24.633
2	5	15.73
2	5	22.231
2	6	11.672
2	6	26.479
2	6	19.784
2	6	20.884
2	6	21.811
2	6	26.344
2	7	22.161
2	7	16.429
2	7	23.311
2	7	21.983
2	7	12.181
2	7	18.252
2	8	30.865
2	8	17.875

2	8	24.562
2	8	20.442
2	8	20.791
2	8	19.44
3	9	47.588
3	9	38.219
3	9	44.445
3	9	47.115
3	9	30.83
3	9	58.708
3	10	62.477
3	10	29.834
3	10	49.902
3	10	59.946
3	10	46.587
3	10	38.567
3	11	54.545
3	11	38.846
3	11	52.163
3	11	43.141
3	11	27.822
3	11	58.025
3	12	57.749
3	12	47.737
3	12	40.653
3	12	45.344
3	12	46.702
3	12	50.091

```
proc anova
model WtGain = Diet Tank*Diet
means Diet / lsd
```

8.6   ivd-demo.txt

* Analyze in vitro digestibility (IVD) of dried alfalfa grown at different temperatures.
* This is a one−way completely randomized design.

* References:
* 1. "Statistical Design" by G. Casella, p. 44.
* 2. http://www.stat.ufl.edu/~casella/StatDesign/WebDataSets/IVD.txt

```
data
input Temp $ IVD
datalines
17    94.2
17    94.5
17    95
17    94.7
22    94.5
22    94
22    94.6
22    94
27    95.1
27    95.7
27    95.5
27    96.1
32    95.2
32    96
32    96.1
32    95.3

proc anova
model IVD = Temp
means Temp / lsd
```

8.7   lmc-demo.txt

* Distance to the Large Magellanic Cloud

* Dataset from http://astrostatistics.psu.edu/datasets/LMC_distance.dat

* TYPE is population type
* DM is distance modulum
* PM is +/−

```
data
input TYPE $ DM PM
datalines
I 18.70      0.16     Feast & Catchpole 1997
I 18.55      0.06     Laney & Stobie 1994
I 18.55      0.10     Gieren et al. 1998, Di Benedetto 1997
I 18.575     0.2      Groenewegen 2000
I 18.4       0.1      Ribas et al. 2002
I 18.42      0.07     This paper see \S 6.2
I 18.45      0.07     This paper see \S 6.2
I 18.59      0.09     Romaniello et al. 2000
I 18.471     0.12     Pietrzynsky \& Gieren 2002
I 18.54      0.10     Sarajedini et al. 2002
I 18.54      0.18     Van Leeuwen 1997
I 18.64      0.14     Feast 2000
I 18.58      0.05     Panagia 1998
II 18.45      0.09     Gratton et al. 2002 and this paper
II 18.45      0.13     Carretta et al. 2000b and this paper
II 18.44      0.13     Carretta et al. 2000b and this paper
II 18.30      0.14     This paper see \S 7
II 18.38      0.16     This paper see \S 7
II 18.50      0.16     Cacciari et al. 2000 and this paper
II 18.55      0.19     A97 and A00
II 18.52      0.18     Kovacs 2000 and this paper
II 18.40      0.15     Gratton et al. 2002 and this paper
II 18.45      0.16     Benedict et al. 2002 and this paper
II 18.55      0.09     Cioni et al. 2000
II 18.69      0.26     Romaniello et al. 2000.
```

* Does DM depend on TYPE? (Answer: No)
```
proc anova
model DM = TYPE
means TYPE / lsd
```

* Does PM depend on TYPE? (Answer: Yes)
```
proc anova
model PM = TYPE
means TYPE / lsd
```

8.8 peanut-demo.txt

```
* Compare peanut yields for four treatments.
* This is a latin square design.

* References
* 1. "Statistical Design" by G. Casella, p. 118.
* 2. http://www.stat.ufl.edu/~casella/StatDesign/WebDataSets/Peanut.txt

data
input Row $ Column $ Treatment $ Yield
datalines
1    1    3     26.7
2    1    1     23.1
3    1    2     28.3
4    1    4     25.1
1    2    1     19.7
2    2    2     20.7
3    2    4     20.1
4    2    3     17.4
1    3    2     28
2    3    4     24.9
3    3    3     29
4    3    1     28.7
1    4    4     29.8
2    4    3     29
3    4    1     27.3
4    4    2     34.1


proc anova
model Yield = Row Column Treatment
means Treatment / lsd
```

8.9 prof-hunter-demo.txt

* Comparision of two experimental designs.

* Reference
* "Design of Experiments" learning series by Professor J. Stuart Hunter, Lesson 12.
* http://www.youtube.com/watch?v=k3n9iSB6Cns
* http://www.youtube.com/watch?v=3fwoU16MHJM
*
* A list of all videos in the series:
* http://www.youtube.com/playlist?list=PLzaIpDs2EIQgXzVAPS9rviC9J7Cpyet5y
*
* Below are the anova tables seen in the video.
*
* **CSS** – corrected sum of squares
* TSS – treatment sum of squares
* BSS – block sum of squares
* RSS – residual sum of squares
*
* 1st anova table
*   DF   MS
* **CSS**   112.0055   19
* TSS   0.8405   1
* RSS   111.1650   18   6.1758
*
* 2nd anova table
*   DF   MS
* **CSS**   112.0055   19
* TSS   0.8405   1
* BSS   110.4905   9
* RSS   0.6745   9   0.0750
*
* 2nd design reduces RSS.
*
*   I   Gasoline with additive
* II   Gasoline without additive
*   y   Miles per 5 gallons

title "Completely Randomized Design"

data
input trt $ y
datalines
I 74.0
I 68.8
I 71.2
I 74.2
I 71.8
I 66.4
I 69.8
I 71.3
I 69.3
I 73.6
II 73.2
II 68.2
II 70.9

```
II  74.3
II  70.7
II  66.6
II  69.5
II  70.8
II  68.8
II  73.3

proc anova
model y = trt
means trt / lsd

title "Randomized Block Design"

data
input car $ trt $ y
datalines
A I  74.0
B I  68.8
C I  71.2
D I  74.2
E I  71.8
F I  66.4
G I  69.8
H I  71.3
I I  69.3
J I  73.6
A II  73.2
B II  68.2
C II  70.9
D II  74.3
E II  70.7
F II  66.6
G II  69.5
H II  70.8
I II  68.8
J II  73.3

proc anova
model y = car trt
means trt / lsd
```

8.10 red-clover-demo.txt

* Compare red clover yields for two levels of nitrogen and four levels of sulphur.
* This is a two–way completely randomized design.

* References
* 1. "Statistical Design" by G. Casella, p. 46.
* 2. http://www.stat.ufl.edu/~casella/StatDesign/WebDataSets/RedClover.txt

```
data
input Nitrogen $ Sulphur $ Yield
datalines
0      0    4.48
0      0    4.52
0      0    4.63
0      3    4.7
0      3    4.65
0      3    4.57
0      6    5.21
0      6    5.23
0      6    5.28
0      9    5.88
0      9    5.98
0      9    5.88
20     0    5.76
20     0    5.72
20     0    5.78
20     3    7.01
20     3    7.11
20     3    7.02
20     6    5.88
20     6    5.82
20     6    5.73
20     9    6.26
20     9    6.26
20     9    6.37

proc anova
model Yield = Sulphur Nitrogen Sulphur*Nitrogen
means Sulphur*Nitrogen
```

8.11 seedling-demo.txt

* Compare heights of seedlings grown from seeds from five forests.
* This is a nested design. Source tree is nested within forest.

* Reference
* http://www.ohio.edu/plantbio/staff/mccarthy/quantmet/lectures/ANOVA–III.pdf

```
data
input Forest $ Tree $ Height
datalines
A T1  15.8
A T1  15.6
A T1  16.0
A T2  13.9
A T2  14.2
A T2  13.5
B T3  18.5
B T3  18.0
B T3  18.4
B T4  17.9
B T4  18.1
B T4  17.4
C T5  12.3
C T5  13.0
C T5  12.7
C T6  14.0
C T6  13.1
C T6  13.5
D T7  19.5
D T7  17.5
D T7  19.1
D T8  18.7
D T8  19.0
D T8  18.8
E T9  16.0
E T9  15.7
E T9  16.1
E T0  15.8
E T0  15.6
E T0  16.3

proc anova
model Height = Forest Tree*Forest
means Forest / lsd
```

8.12   strawberry-demo.txt

```
* Compare yields for three types of strawberries.
* This is a randomized complete block design.

* References
* 1. "Statistical Design" by G. Casella, p. 9 and 94.
* 2. http://www.stat.ufl.edu/~casella/StatDesign/WebDataSets/Strawberry.txt

data
input Block $ Treatment $ Yield
datalines
1    A    10.1
1    B    6.3
1    C    8.4
2    A    10.8
2    B    6.9
2    C    9.4
3    A    9.8
3    B    5.3
3    C    9
4    A    10.5
4    B    6.2
4    C    9.2


proc anova
model Yield = Block Treatment
means Treatment / lsd
```

8.13   trees-demo.txt

```
data
input Girth Height Volume
LG = log(Girth)
LH = log(Height)
LV = log(Volume)
datalines
  8.3     70    10.3
  8.6     65    10.3
  8.8     63    10.2
 10.5     72    16.4
 10.7     81    18.8
 10.8     83    19.7
 11.0     66    15.6
 11.0     75    18.2
 11.1     80    22.6
 11.2     75    19.9
 11.3     79    24.2
 11.4     76    21.0
 11.4     76    21.4
 11.7     69    21.3
 12.0     75    19.1
 12.9     74    22.2
 12.9     85    33.8
 13.3     86    27.4
 13.7     71    25.7
 13.8     64    24.9
 14.0     78    34.5
 14.2     80    31.7
 14.5     74    36.3
 16.0     72    38.3
 16.3     77    42.6
 17.3     81    55.4
 17.5     82    55.7
 17.9     80    58.3
 18.0     80    51.5
 18.0     80    51.0
 20.6     87    77.0

proc reg
model LV = LG LH
```