CHAPTER 8: TWO-LEVEL EXPERIMENTS WITH SEVERAL VARIABLES

1. Introduction

In earlier chapters we have seen that causal inferences are less ambiguous when the data are generated by designed experiments, especially when randomization is used in establishing the sequence by which trials are conducted. We studied examples of two very simple experimental designs, the simple randomized experiment and the paired sample design.

Both designs can be applied either to time-series data or to cross-sectional data. For example, in Section 4 of Chapter 5, Randomized Pairs, the blood-pressure experiment was run for one person sequentially across a whole day to evaluate the effect of hyperventilation on blood pressure readings; in Section 4 of Chapter 7, the boys shoes experiment entailed a comparison of wear of different sole materials for a sample of 10 boys.

In the current chapter, all the experiments will be based on time-series data. They were carried out in a definite time sequence -- a randomized sequence, in fact. All studied more than one experimental variable. Each variable was set at two distinct levels only.

These examples only begin to introduce the potentialities of designed experimentation; but they offer a simple and important tool for finding out which, of a number of possible independent variables, actually affect the dependent variable: process output, yield, quality, or cycle time. These designs are called **two-level factorial and fractional-factorial designs.** They can be used either for quantitative or qualitative independent variables.[1]

As usual, we shall develop the key ideas in the context of specific applications. Before doing so, however, we deal with a philosophical point that has important practical consequences both for scientific method and for management practice. There is an obsolete proposition that is still widely taught in colleges and universities:

> "To learn about causation by experimentation, it is essential to vary one variable at a time and **to hold all other variables constant.**"

By the examples of this chapter, and the principles conveyed by them, we shall see that:

- It is possible, and often desirable, to vary several variables at once.

[1] If we already know which independent variables are important and are interested in finding the best settings for quantitative independent variables, another class of designs, called **response surface designs**, is applicable. In this chapter, we shall discuss only the factorial and fractional-factorial designs.

- The one-variable-at-a-time approach leads to unnecessarily large, expensive, and time-consuming experiments.

- The one-variable-at-a-time approach fails to permit learning about **interaction effects -- effects that are either greater or less when the variables are combined than would be inferred from their separate effects.**

For a quick survey of the key ideas of the chapter, look at the examples of Section 2 and Section 5.

2. The Pilot Plant Experiment: Full Factorial Design with Replication

We begin with an application to chemical engineering, where the objective is to increase the yield of a pilot plant. The following data are contained in a file called PILOTPLN.sav:

PILOTPLN.sav - SPSS Data Editor

	t	c	k	txc	txk	cxk	txkxc	y	order	index
1	-1	-1	-1	1	1	1	-1	59.0	6	1
2	1	-1	-1	-1	-1	1	1	74.0	2	2
3	-1	1	-1	-1	1	-1	1	50.0	1	3
4	1	1	-1	1	-1	-1	-1	69.0	5	4
5	-1	-1	1	1	-1	-1	1	50.0	8	5
6	1	-1	1	-1	1	-1	-1	81.0	9	6
7	-1	1	1	-1	-1	1	-1	46.0	3	7
8	1	1	1	1	1	1	1	79.0	7	8
9	-1	-1	-1	1	1	1	-1	61.0	13	9
10	1	-1	-1	-1	-1	1	1	70.0	4	10
11	-1	1	-1	-1	1	-1	1	58.0	16	11
12	1	1	-1	1	-1	-1	-1	67.0	10	12
13	-1	-1	1	1	-1	-1	1	54.0	12	13
14	1	-1	1	-1	1	-1	-1	85.0	14	14
15	-1	1	1	-1	-1	1	-1	44.0	11	15
16	1	1	1	1	1	1	1	81.0	15	16

Pilot plant experiment to increase yield (Y) of a chemical plant
by choosing advantageous combinations of low and high levels of
three variables, temperature, concentration, and catalyst.

Variables:
t: temperature (-1 is low level, +1 is high level)
c: concentration (-1 is low level, +1 is high)
k: catalyst (qualitative variable: A is -1, B is +1)
txc: interaction of temperature and concentration
txk: interaction of temperature and catalyst
cxk: interaction of concentration and catalyst
txcxk: interaction of temperature, concentration, and catalyst.
y: yield
order: order in which the trials were actually made
index: order in which the trials are listed above for convenience, the "standard sequence" or "design matrix"
From Box, Hunter, and Hunter, *Statistics for Experimenters*, Wiley, 1978, Table 10.3, page 320.

Each of the two quantitative variables, temperature -- **t** --and concentration -- **c** -- is set at either a low or a high level, For temperature, the low level (coded -1 for convenience) is 160 degrees celsius, and the high level (coded 1 for convenience) is 180 degrees. Similarly, the low level for concentration is 20 percent, coded -1, and the high level is 40 percent, coded 1. In both instances, the low and high levels are modest departures from current standard practice. For example, the then standard temperature for temperature was about 170 degrees and for concentration, 30 percent.

The third variable -- **k** -- is qualitative, there being two catalysts, A and B, that are arbitrarily coded -1 and 1. One was at that time the standard catalyst, the other a promising alternative.

The interaction variables are simple transformations of two or three of the three primary variables that will be explained shortly.[2]

This experiment may not answer all questions about the effects on yield of **t,c**, and **k**, and further experimentation may be indicated by its results. But the experiment is designed to learn about all three variables at once and to lead to improvement over the current process. To see how this happens, follow carefully the steps in the following analysis.

We now study the pattern of minuses "-" and pluses "+" to show how it was arrived at and what it means for the analysis. Consider first the columns in **Data Editor** for the first three variables, **t,c,** and **k**. To distinguish these from the interaction variables, we use the term **"main effect"**. Thus our potential independent variables include, along with the **interaction effects** to be explained in a moment, three main effects—**t, c**, and **k**:

- The first column, **t**, starts with a minus sign and alternates minuses and pluses.
- The second column, **c**, starts with two minus signs and alternates two minuses and two pluses.
- The third column, **k**, starts with four minus signs and is followed by four plus signs.
- The sequence of eight combinations of **t, c**, and **k** is repeated once more.

There are three important consequences of this layout:

- The column sum, and therefore the mean, of each of the three main-effect variables is zero.
- For each of the three variables, the mean of the squared deviations about the mean of zero is one. Essentially, the variance (squared standard deviation) and the standard deviation of each main effect are one.[3]

---

[2]As we shall show in later examples, it would also have been possible to "block" the 16 observations into two consecutive groups of 8, create a block indicator variable, and to have randomized within each block rather than across all 16 observations.

[3]This statement needs a technical qualification. It would be precisely correct if the sum of squared deviations were divided by the sample size n in the process of computing the standard deviation. However, it is conventional to divide by n-1 rather than n, and this gives rise to a numerical discrepancy. The computed standard deviation, based on the divisor ***n-1***, will therefore be slightly greater than one: in this application it will be 1.033, as we shall show below.

- The sum of cross-products of deviations from zero for any pair of the three variables is zero. For example, for **t** and **c**, we have:

  (-1)(-1)+1(-1)+(-1)1+1(1)+(-1)(-1)+1(-1)+(-1)1+1(1) = 1-1-1+1+1-1-1+1 = 0

  This implies that **t** and **c** are uncorrelated; also **t** and **k** and **c** and **k** are uncorrelated. Thus we have zero multicollinearity (zero correlation between independent variables), and by design!

  Next, we bring the fourth variable -- **txc** -- into the picture:

txc is the **two-factor interaction** between **t** and **c**. Numerically, it is obtained by multiplying **t** by **c**. On the first row, for example, **t=-1** and **c=-1**, so **t** times **c = (-1)(-1) = 1**. If you were setting this up in ***SPSS*** you could create **txc** with **Transform/Compute…**:

txc = t*c

txc equals 1 when **t** and **c** are both high (+1) or both low (-1); **txc** equals **-1** when **t** and **c** are of opposite signs. We shall see how this definition is useful in practice when we come to the actual data analysis.

The remaining interactions -- **txc**, **cxk**, and **txcxk** -- are computed in the same way. The first 8 rows of the design constitute a "2 to the 3rd" factorial design -- written "**2^3 factorial design":** there are 3 **factors** -- **t, c**, and **k** -- each at two **levels,** -1 and +1.

Rows 9 to 16 are simply a replication of the first eight rows, so we speak of the full design as a "**2^3 factorial design with two replications**".

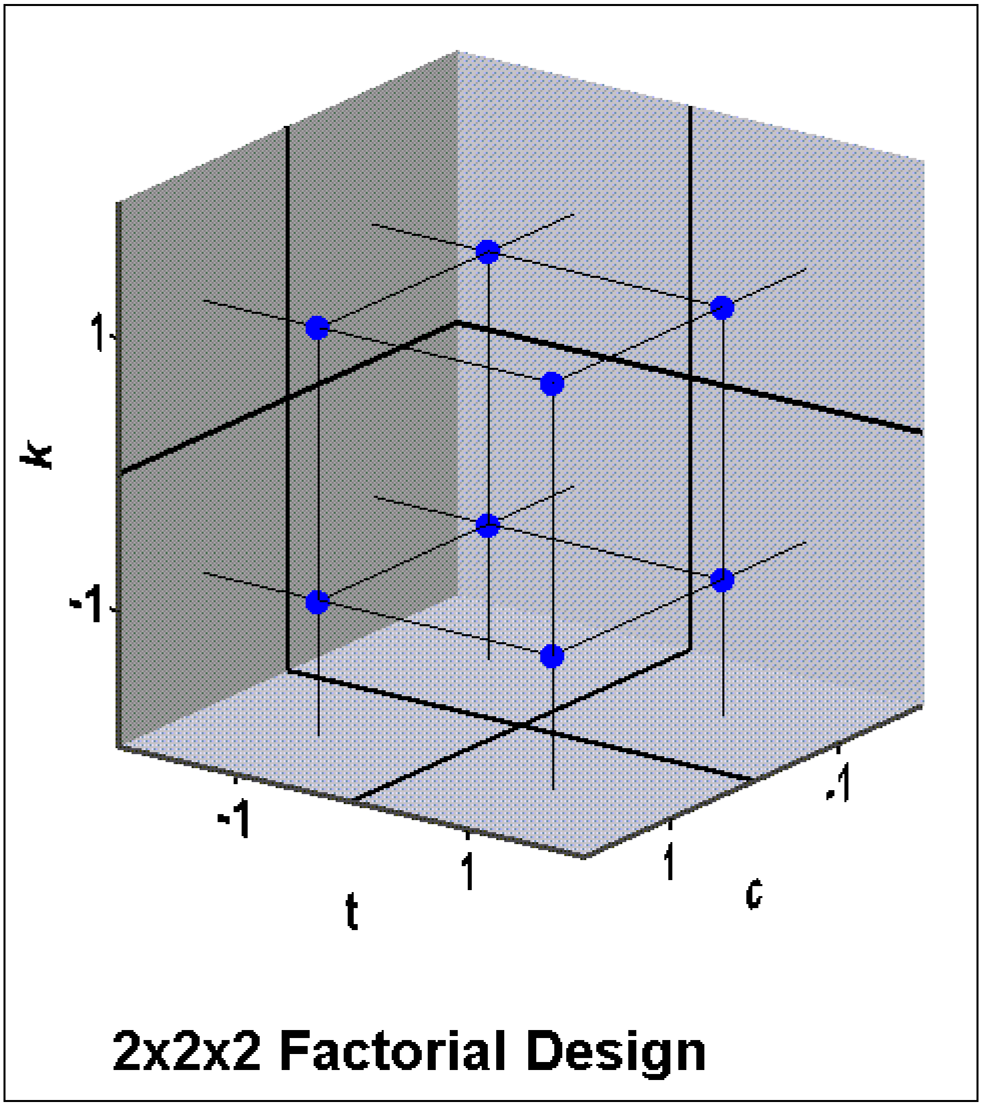
The following table is a convenient way to display the layout of this factorial design:

Pilot Plant Factorial Design

t	c	k		N
-1	-1	-1	y	2
		1	y	2
	1	-1	y	2
		1	y	2
1	-1	-1	y	2
		1	y	2
	1	-1	y	2
		1	y	2

This table was produced by the ***SPSS*** sequence **Analyze/Reports/OLAP Cubes…**where **OLAP** stands for "Online Analytical Processing." The procedure is rather involved and we will not discuss it further here except to say that it also requires **table pivoting.** (Your instructor should demonstrate this procedure in class, and it is also helpful to read Chapter 6, "Working with Output" in your *SPSS 12.0 Brief Guide.*)The table shows that for each of the eight factor combinations, there are two replicates in each cell.

Finally, here is another way to visualize the layout of this design. Imagine taking the layers of the table above and stacking them to form the following 2x2x2 cube shown below:



The eight cells formed by the cube represent the different combinations of factors. This is an example of a 2^n factorial design where $n = 3$.

The next step is to obtain descriptive statistics for the factors and interactions:

Descriptive Statistics

	N	Minimum	Maximum	Mean	Std. Deviation
t	16	-1	1	.00	1.033
c	16	-1	1	.00	1.033
k	16	-1	1	.00	1.033
txc	16	-1	1	.00	1.033
txk	16	-1	1	.00	1.033
cxk	16	-1	1	.00	1.033
txkxc	16	-1	1	.00	1.033
y	16	44.0	85.0	64.250	13.4139
Valid N (listwise)	16				

Note first that the means of **all** potential independent variables (not just the main effects **t, c**, and **k** but also the interactions **txc, txk, cxk,** and **txcxk**) are equal to zero. Second, all the standard deviations are printed out as 1.033. The mean of the squared deviations for each of these variables is 1, so the sum is 16. As explained above in a footnote, the conventional way of computing the standard deviation of a sample is to divide, not by ***n***, but ***n-1***, and then to take the square root. So we have $\frac{\sqrt{16}}{\sqrt{15}} = 1.033$.

Next we look at the correlations:[4]

Correlations

		t	c	k	txc	txk	cxk	txkxc	y
t	Pearson Correlation	1	.000	.000	.000	.000	.000	.000	.885**
c	Pearson Correlation	.000	1	.000	.000	.000	.000	.000	-.192
k	Pearson Correlation	.000	.000	1	.000	.000	.000	.000	.058
txc	Pearson Correlation	.000	.000	.000	1	.000	.000	.000	.058
txk	Pearson Correlation	.000	.000	.000	.000	1	.000	.000	.385
cxk	Pearson Correlation	.000	.000	.000	.000	.000	1	.000	.000
txkxc	Pearson Correlation	.000	.000	.000	.000	.000	.000	1	.019
y	Pearson Correlation	.885**	-.192	.058	.058	.385	.000	.019	1

**. Correlation is significant at the 0.01 level (2-tailed).

The correlation between **all** pairs of the seven potential independent variables is zero! This is a desirable consequence of the design of the study. It means, for example, that when we do regressions of yield **y** on any combination of these variables, we will get the same coefficient for any given variable -- say, **c** -- regardless of the other variables in the regression.

[4] In the **Correlations** table we have hidden the rows for significance level and sample size to make it easier to read. Your instructor can show you how to do this.

We are almost ready to do the analysis. However, in columns 1 through 10 the data are listed in the standard sequence, not in the order in which they were actually run. The variable **order** -- gives the actual order, so we next rearrange the data in the actual order, using **Data/Sort Cases….**

The original data and sorted data are next compared. We would get the same regression equation from either layout, but the sorted data enable us to check as to whether or not the residuals appear to be in control.

Before the sort:

	t	c	k	txc	txk	cxk	txkxc	y	order	index
1	-1	-1	-1	1	1	1	-1	59.0	6	1
2	1	-1	-1	-1	-1	1	1	74.0	2	2
3	-1	1	-1	-1	1	-1	1	50.0	1	3
4	1	1	-1	1	-1	-1	-1	69.0	5	4
5	-1	-1	1	1	-1	-1	1	50.0	8	5
6	1	-1	1	-1	1	-1	-1	81.0	9	6
7	-1	1	1	-1	-1	1	-1	46.0	3	7
8	1	1	1	1	1	1	1	79.0	7	8
9	-1	-1	-1	1	1	1	-1	61.0	13	9
10	1	-1	-1	-1	-1	1	1	70.0	4	10
11	-1	1	-1	-1	1	-1	1	58.0	16	11
12	1	1	-1	1	-1	-1	-1	67.0	10	12
13	-1	-1	1	1	-1	-1	1	54.0	12	13
14	1	-1	1	-1	1	-1	-1	85.0	14	14
15	-1	1	1	-1	-1	1	-1	44.0	11	15
16	1	1	1	1	1	1	1	81.0	15	16
17										

After:

	t	c	k	txc	txk	cxk	txkxc	y	order	index
1	-1	1	-1	-1	1	-1	1	50.0	1	3
2	1	-1	-1	-1	-1	1	1	74.0	2	2
3	-1	1	1	-1	-1	1	-1	46.0	3	7
4	1	-1	-1	-1	-1	1	1	70.0	4	10
5	1	1	-1	1	-1	-1	-1	69.0	5	4
6	-1	-1	-1	1	1	1	-1	59.0	6	1
7	1	1	1	1	1	1	1	79.0	7	8
8	-1	-1	1	1	-1	-1	1	50.0	8	5
9	1	-1	1	-1	1	-1	-1	81.0	9	6
10	1	1	-1	1	-1	-1	-1	67.0	10	12
11	-1	1	1	-1	-1	1	-1	44.0	11	15
12	-1	-1	1	1	-1	-1	1	54.0	12	13
13	-1	-1	-1	1	1	1	-1	61.0	13	9
14	1	-1	1	-1	1	-1	-1	85.0	14	14
15	1	1	1	1	1	1	1	81.0	15	16
16	-1	1	-1	-1	1	-1	1	58.0	16	11
17										

The next step is to apply our standard approach for selection of a tentative regression model. A **Stepwise Regression** will show that the coefficients do not change as new variables are entered into the model:

Model Summary[d]

Model	R	R Square	Adjusted R Square	Std. Error of the Estimate
1	.885[a]	.784	.769	6.4531
2	.966[b]	.932	.922	3.7519
3	.985[c]	.969	.962	2.6300

a. Predictors: (Constant), t

b. Predictors: (Constant), t, txk

c. Predictors: (Constant), t, txk, c

d. Dependent Variable: y

Coefficients[a]

Model		Unstandardized Coefficients		Standardized Coefficients	t	Sig.
		B	Std. Error	Beta		
1	(Constant)	64.250	1.613		39.826	.000
	t	11.500	1.613	.885	7.128	.000
2	(Constant)	64.250	.938		68.498	.000
	t	11.500	.938	.885	12.260	.000
	txk	5.000	.938	.385	5.331	.000
3	(Constant)	64.250	.657		97.720	.000
	t	11.500	.657	.885	17.491	.000
	txk	5.000	.657	.385	7.605	.000
	c	-2.500	.657	-.192	-3.802	.003

a. Dependent Variable: y

Note first that as we move from step 1 to step 2 to step 3 of the stepwise process, the regression coefficient of any one variable is unchanged. (Remember, every pair of independent variables is uncorrelated.) For example, the coefficient of **t** is **11.50** at steps 1, 2, and 3. The t-ratio for **t** increases from 7.13 to 12.26 to 17.49 as the **Std. Error of the Estimate** decreases from 6.4531 to 3.7519 to 2.6300. This decrease of the standard error of the residuals, in turn, is made possible by the additional significant variables that are added to the model at step 2 and step 3. We have not mentioned it before, but the Std. Error of the Estimate is an essential ingredient of the denominator of the t-ratio. (When it decreases, the t-ratio increases.)

Three variables -- two main effects, **t** and **c** -- and one two-factor interaction --**txk** -- appear to be significant. Before going ahead to diagnostic checks, we first examine and interpret the fitted regression equation:

predicted y = 64.25 + 11.5 t - 2.50 c + 5.0 txk

This says that, for given values of the other variables:

- Fitted **y** increases by 11.5x2= 23.0 percent as **t** goes from its low level -1 to its high level +1.

- Fitted **y** decreases by 2.50x2 = 5.00 percent as **c** goes from its low level -1 to its high level +1.

But how do we interpret **txk**? If **k** is at its low level -1 (catalyst A), the fitted equation above becomes:

predicted y = 64.2 + 6.5 t - 2.50 c

Hence the effect of **t** is only 6.5x2 = 13 percent when **k** is at its low level.

However, when **k** is at its high level +1, we have

predicted y = 64.2 + 16.5 t - 2.50 c

Therefore the significant interaction **txk** means that the catalyst appears to affect the result, not directly, but through its effect on the **t** variable. The temperature effect is more pronounced for catalyst B (**k=1**) than for catalyst A (**k=-1**). This illustrates the idea of an interaction variable.

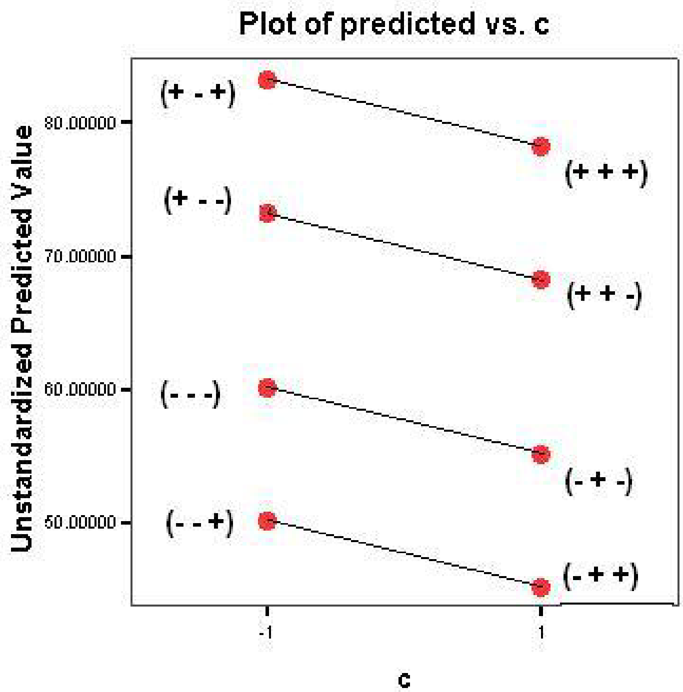
Thus, when asked to provide an equation for the predicted value, it is better to report two equations:

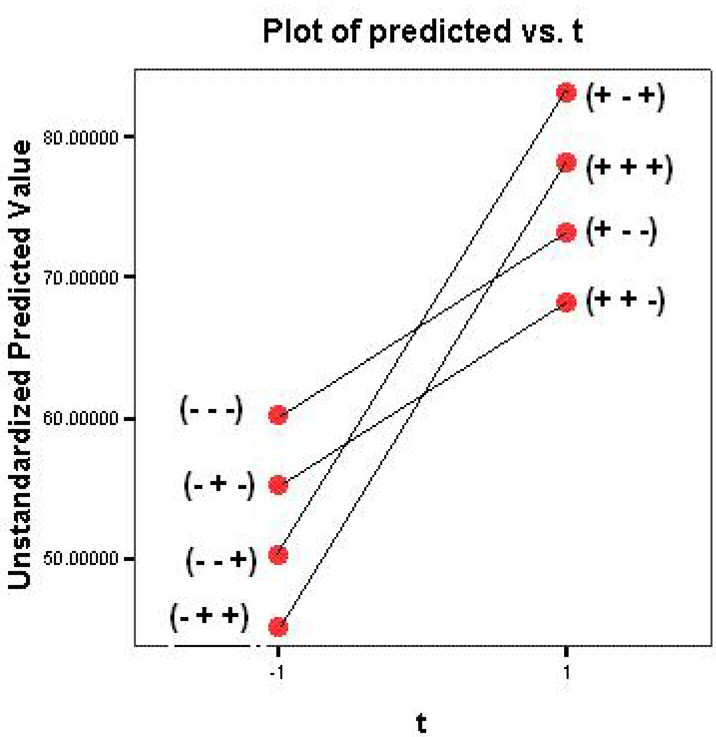
predicted y = 64.2 + 6.5 t - 2.50 c if **k = -1** (Catalyst A)

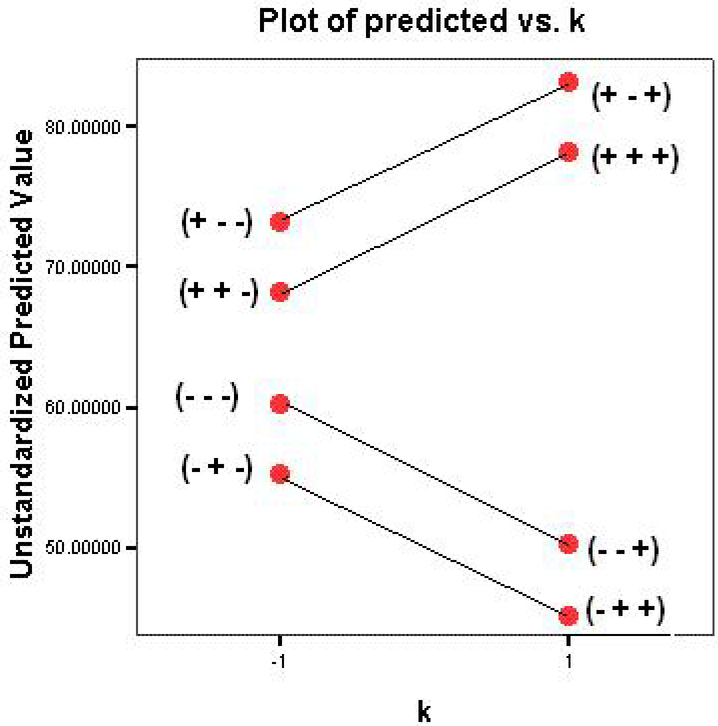
predicted y = 64.2 + 16.5 t - 2.50 c if **k = 1** (Catalyst B)

A Visual Explanation of Interaction

The graphs below explain the effect of interaction from a visual approach:







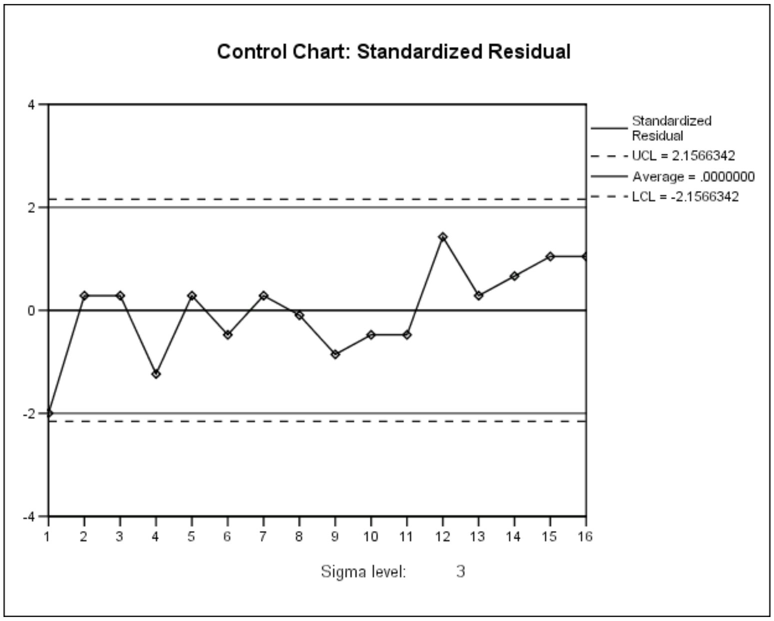
It should be noted that these three plots of **predicted** vs each of the main effects were initially made with ***SPSS*** but that they were embellished by importation into *Microsoft Paint*.

Let's begin our explanation by examining the scatter of **predicted** vs. **c**. At each point on the graph we have indicated the settings for the triplet (**t,c,k**), always in the same order. Thus, for example, you can read that when all three factors are at their low settings, the predicted value is 60; when the settings are all high, the predicted value is 78; and so on, for the other combinations. These predicted values for each combination of factor settings are exactly the same in each of the three graphs, but their position with respect to the horizontal axis can change. The first graph also shows that when **t** is high the fitted value is higher than when **t** is at the low setting. This relation also must hold in each of the three graphs. The straight lines connect the combinations that differ only by the setting of **c**; thus they show what happens to **predicted** when **c** is changed from low to high while keeping the settings on the other factors unchanged. The most important feature about the scatter of **predicted** on **c**, however, is that the lines are parallel. This shows that whatever the settings of **t** and **k**, when **c** increases from low to high the fitted value **decreases by a constant amount**. The fact of this constant change, unaffected by **t** or **k**, shows that **c does not interact with either of those variables**.

Now you've got the idea! The second plot, **predicted** vs. **t**, demonstrates visually that when **t** is increased from low to high, the fitted value of **y** increases, but the **amount of increase depends on the setting of k**. We see that when **k** is low (Catalyst A) the effect of an increase in **t** is not as great as when **k** is high (catalyst B). The crossing of the lines shows the **interaction between t and k**. Note, however, that the two lines that start at the top left are parallel to each other. The same is true for the two lines that begin at the bottom left. The end points of the parallel lines only differ by the setting of **c**, thus showing from another angle that **t and c do not interact**.

Finally, the third graph, the plot of **predicted** vs. **k**, again demonstrates visually the **interaction between t and k**, and the **absence of interaction between c and k**, but from a slightly different approach. See if you can talk yourself through the argument.

Next we turn to diagnostic checking of the residuals, which will lead to a small refinement of the model:



This first visual check gives a strong hint of an upward trend of residuals, suggesting that there was some unintended drift of background conditions during the execution of the 16 trials. (In a broad sense this is good news, because it suggests that even as the experiment was being run, something else was being done to improve mean yields, but it does muddy the interpretation of the data.)

We can capture this discovery into the regression model by inclusion of a time trend into the regression. The variable **order** is already listed in time order 1, 2, ... , 16 because of the sort that we performed, so it will serve as the time sequence variable (we have previously used the name **time**) for modeling trend.

Model Summary[b]

Model	R	R Square	Adjusted R Square	Std. Error of the Estimate
1	.991[a]	.982	.976	2.0725

a. Predictors: (Constant), txk, c, t, order

b. Dependent Variable: y

Coefficients[a]

Model		Unstandardized Coefficients		Standardized Coefficients	t	Sig.
		B	Std. Error	Beta		
1	(Constant)	61.300	1.146		53.471	.000
	order	.347	.120	.123	2.885	.015
	t	11.587	.519	.892	22.325	.000
	c	-2.500	.518	-.192	-4.825	.001
	txk	4.436	.554	.342	8.010	.000

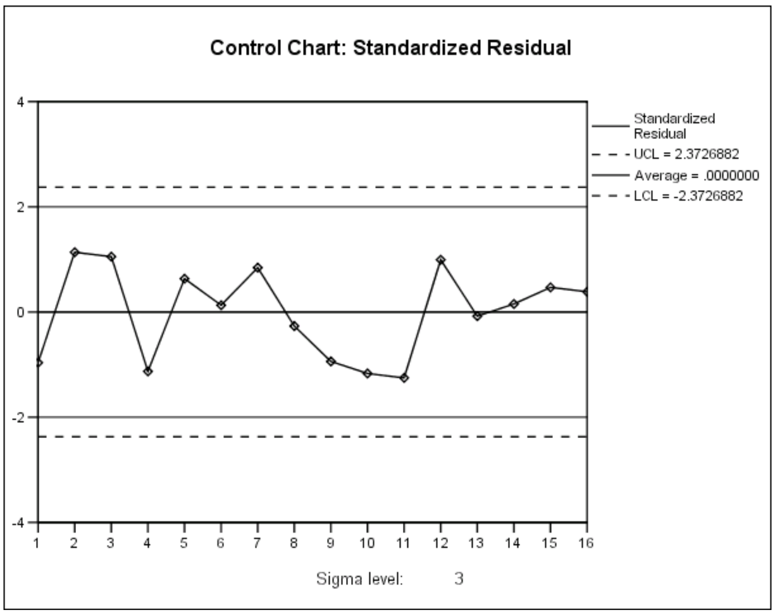
a. Dependent Variable: y

The fit is improved, the trend is significant. The introduction of the trend variable **order** has slightly changed the regression coefficients of the other variables because of their slight correlation with **order**. (The effect of **c** is unchanged because it happened not be correlated at all with **order**.)

If this model holds up under another round of diagnostics, our final regression equation will be:

predicted y = 61.300 + 0.347 order + 11.587 t - 2.500 c + 4.436 txk

The diagnostics are now satisfactory:

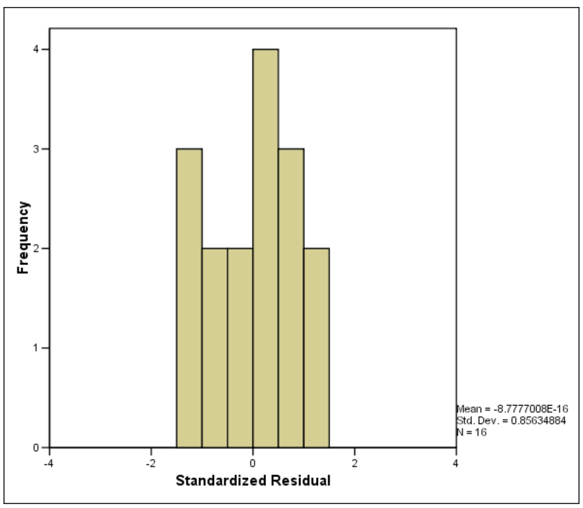


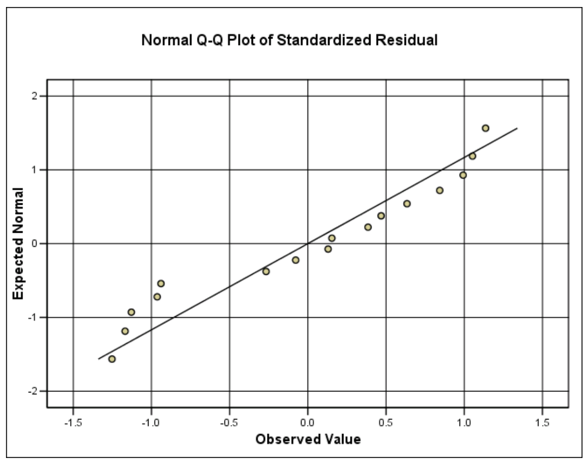
Runs Test

	Standardized Residual
Test Value[a]	.0000000
Cases < Test Value	7
Cases >= Test Value	9
Total Cases	16
Number of Runs	8
Z	-.197
Asymp. Sig. (2-tailed)	.844

a. Mean

```
Autocorrelations:   ZRE_2   Standardized Residual

     Auto- Stand.
Lag  Corr.    Err. -1  -.75  -.5 -.25   0   .25  .5   .75   1   Box-Ljung  Prob.
                    ├────┼────┼────┼────┼────┼────┼────┼────┤
  1  -.010   .228              .        *        .                 .002   .967
  2  -.133   .220              .     ***|        .                 .367   .832
  3  -.050   .212               .      *|       .                  .423   .936
  4  -.150   .204               .    ***|       .                  .965   .915
  5   .036   .195               .       |*      .                  .998   .963
  6  -.222   .186                .  ****|      .                  2.424   .877
  7  -.033   .177                .     *|      .                  2.459   .930
  8  -.236   .167                . *****|      .                  4.469   .813
  9   .120   .156                 .     |**   .                   5.059   .829
 10   .221   .144                 .     |**** .                   7.394   .688
 11  -.106   .132                  .  **|    .                    8.043   .709
 12   .028   .118                  .    |*   .                    8.100   .777

Plot Symbols:      Autocorrelations *     Two Standard Error Limits .

Total cases:  16     Computable first lags:  15
```





Note that in the table at right we have used the table editor to hide the results of the Kolmogorov-Smirnov test, information that we have never used in this course since the Shapiro-Wilk test suffices.

Tests of Normality

	Shapiro-Wilk		
	Statistic	df	Sig.
Standardized Residual	.904	16	.092

A word is in order about the rather boxy shape and the relatively narrow range of variation of the above histogram of residuals, as well as the near-significance of the Shapiro-Wilk statistic.

- The narrow range is partly an artifact. It arises because the formula for the standard deviation of residuals entails division of the sum of squared residuals not by ***n*** or even by ***n-1***, but by ***n-k-1***, where ***k*** is the number of independent variables in the regression, here

k=3. Thus ***n-k-1*** = 16-3-1 = 12. This "squeezes" down the range of the standardized residuals.

- The boxy shape is more puzzling, but it is a common occurrence in residuals from regressions with small sample sizes and a relatively high ratio between number of independent variables and the sample size. We do not believe, however, that this is a serious blemish on the diagnostic checks.[5] The more serious problem, which does not arise here, is raised by outliers, which always suggest the possibility of special causes or important omitted variables.

To gain perspective on what we have learned, it will be helpful to review the results in tabular form, again using the ***SPSS*** procedure **Analyze/Reports/OLAP cubes….** This time we classify summaries of the data in tables for **t** vs. **c,** separately for each level of **k.** The values in the individual cells are the mean values of the process yield, **y.**

Pilot Plant Factorial Design

k: -1

		Mean		
		c		
t		-1	1	Total
-1	y	60.000	54.000	57.000
1	y	72.000	68.000	70.000
Total	y	66.000	61.000	63.500

Pilot Plant Factorial Design

k: 1

		Mean		
		c		
t		-1	1	Total
-1	y	52.000	45.000	48.500
1	y	83.000	80.000	81.500
Total	y	67.500	62.500	65.000

The tables above enables us to see the mean **y** for each combination of the three factors. We can then immediately identify the "winning combination" with the maximum **mean y = 83** as **t = 1**, **c = -1**, and **k = 1**; in other words, high temperature, low concentration, and Catalyst B.

A similar application of **OLAP Cubes…,** but this time with **predicted value** as the dependent variable would show that the maximum fitted value is 83.814, again at high temperature, low concentration, and Catalyst B.[6]

Summarizing our findings, the highest actual and highest fitted yields -- about 83 percent -- are achieved for the cell for which **t** = 1, **c** = -1, and **k** = 1: high temperature, low concentration, and Catalyst B: that cell is the "winner". The regression analysis assures us that the winner's success is statistically significant. Of the alternative input variables tried in this experiment, this winning combination promises highest mean yield in the future. (Recall also, however, that there was a significant uptrend of percentage yield during the course of the experiment; if that continues, we should do somewhat better in the future than would be expected from the best choice of input variables alone.)

---

[5] Notice also that the normality probability plot can hardly be called a straight line.

[6] You can either take our word for this or teach yourself how to use OLAP Cubes. It is fun to do, but rather time consuming.

The winning treatment combination thus identified is now a base from which we may experiment further with temperature and concentration. This can be done in one of two ways:

Using Catalyst B, experiment in the same way with further two-level variations of temperature and concentration about their new best levels. This is the basic idea of a strategy of continuing improvement by a succession of two-level experiments, which is known as **evolutionary operation.**

Using Catalyst B, experiment with a whole grid of quantitative levels (not just a high and a low) about the new best levels of temperature and concentration with the aim of finding an optimum level. This is the basic idea of a strategy known as **response surface analysis.**

Suppose that we had carried out a simpler, less informative tabular analysis in which we used only the two quantitative variables **t** and **c** as classifiers. You will see below that the cell defined by **t** = 1 and **c** = -1 has the highest fitted yield, but the fitted yield is less, **77.903**.

OLAP Cubes

		Mean		
		c		
t		-1	1	Total
-1	Unstandardized Predicted Value	55.597	49.903	52.750
1	Unstandardized Predicted Value	77.903	73.597	75.750
Total	Unstandardized Predicted Value	66.750	61.750	64.250

In other words, if we leave the catalyst variable **k** out of the picture, we miss an important piece of information: Even though **k** is not directly significant as a main effect, it contributes to the overall fit because of its interaction with **t**. **Moreover, the predicted yield is higher if k = 1, that is, if catalyst B is used.** An experimental design in which each factor was varied one-at-a-time, rather than using the present factorial approach, would never have uncovered this fact.

3. Two-level Factorial without Replication: Running Experiment

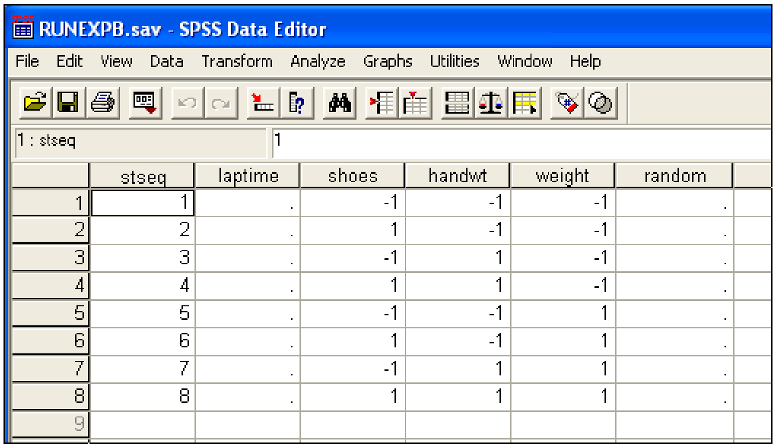
The experiment reported in Section 2 used 16 observations in two replications of the basic 2^3 design, and produced some useful findings. Is it possible to get useful results from a single replication of the design, using 8 observations? We now look at a "lightning data set" in which precisely this was attempted.

Optional Computer Preliminaries

We begin with computer preliminaries; it is necessary only to study the background shown for the file RUNEXPB.sav, immediately below. These preliminaries show how you can set up

SPSS to advantage in conducting such experiments, including the randomization to determine the order in which trials are to be run and the analysis after you collect the data. We illustrate by the set up of a 2^3 factorial experiment, with possible replication in a second block, which is used in the specific running experiment described in the below. The procedure is easily generalized.

The data analysis and results will follow immediately after these computer preliminaries. The contents of RUNEXPB.sav are explained in the first person by the runner who designed the experiment:



	stseq	laptime	shoes	handwt	weight	random
1	1	.	-1	-1	-1	.
2	2	.	1	-1	-1	.
3	3	.	-1	1	-1	.
4	4	.	1	1	-1	.
5	5	.	-1	-1	1	.
6	6	.	1	-1	1	.
7	7	.	-1	1	1	.
8	8	.	1	1	1	.
9						

This is a blank data file for a 23 factorial design.**
It will be filled in with data and random numbers.

stseq: = Standard sequence for experimental trials
laptime:= Time to run around a block that is about 3/8 mile
random:= Sample of random normal numbers generated by ***SPSS***

The aim is to keep effort constant and moderate while varying the following variables:

shoes: = -1 for my old Tiger running shoes
= +1 for my favorite Reebok running shoes
handwt: = -1 for no hand weights
= +1 for one pound gloves for each hand
weight: = -1 for no added weight
= +1 for backpack weighing approximately five pounds

Timing is done with a Casio Lap Memory watch, which records the times for later retrieval.

At the end of each lap, I make necessary changes of shoes, backpack, and weighted gloves.

We shall show in a moment how to generate the random numbers, **random**, that will determine the sequence of the eight trials. The actual **laptime** values can be entered later by hand when the data have been collected.

First we must use **Transform/Compute…** to set up the three **two-factor** and the one **three-factor** interactions:

sxh = shoes*handwt
sxw = shoes*weight
hxw = handwt*weight
sxhxw = shoes*handwt*weight

After adjusting the width and number of decimal places, the data now look like this:

RUNEXPB.sav - SPSS Data Editor

File Edit View Data Transform Analyze Graphs Utilities Window Help

1 : stseq 1

	stseq	laptime	shoes	handwt	weight	random	sxh	sxw	hxw	sxhxw
1	1	.	-1	-1	-1	.	1	1	1	-1
2	2	.	1	-1	-1	.	-1	-1	1	1
3	3	.	-1	1	-1	.	-1	1	-1	1
4	4	.	1	1	-1	.	1	-1	-1	-1
5	5	.	-1	-1	1	.	1	-1	-1	1
6	6	.	1	-1	1	.	-1	1	-1	-1
7	7	.	-1	1	1	.	-1	-1	1	-1
8	8	.	1	1	1	.	1	1	1	1
9										

Before we proceed we must save our spreadsheet as the file RUNEXPB.sav so that it can be easily retrieved and used again for future experiments.

Descriptive Statistics

	N	Minimum	Maximum	Mean	Std. Deviation
shoes	8	-1	1	.00	1.069
handwt	8	-1	1	.00	1.069
weight	8	-1	1	.00	1.069
sxh	8	-1	1	.00	1.069
sxw	8	-1	1	.00	1.069
hxw	8	-1	1	.00	1.069
sxhxw	8	-1	1	.00	1.069
Valid N (listwise)	8				

The means of all variables are zero, and the standard deviations are 1.069. (The standard deviations would be exactly 1 if the sum of squares were divided by n = 8, but the convention is to divide by $n-1$ = 7: $\sqrt{\frac{8}{7}} = 1.069$.)

Correlations

		shoes	handwt	weight	sxh	sxw	hxw	sxhxw
shoes	Pearson Correlation	1	.000	.000	.000	.000	.000	.000
handwt	Pearson Correlation	.000	1	.000	.000	.000	.000	.000
weight	Pearson Correlation	.000	.000	1	.000	.000	.000	.000
sxh	Pearson Correlation	.000	.000	.000	1	.000	.000	.000
sxw	Pearson Correlation	.000	.000	.000	.000	1	.000	.000
hxw	Pearson Correlation	.000	.000	.000	.000	.000	1	.000
sxhxw	Pearson Correlation	.000	.000	.000	.000	.000	.000	1

As we have seen before, by the arrangement of the experimental design, the independent variables are mutually uncorrelated. There is no problem of multicollinearity here!

Next, we prepare for randomization of the sequence by putting random numbers in the column of the spreadsheet for **random** We simply use **Transform/Compute…** to create

random = RV.UNIFORM(0,1)

The spreadsheet now looks like this:[7]

RUNEXPB.sav - SPSS Data Editor

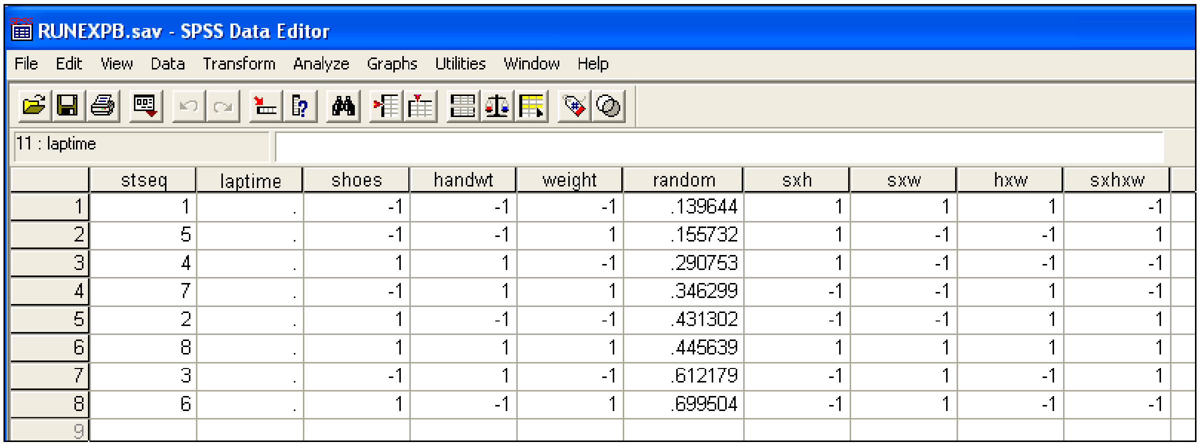
File Edit View Data Transform Analyze Graphs Utilities Window Help

1 : stseq 1

	stseq	laptime	shoes	handwt	weight	random	sxh	sxw	hxw	sxhxw
1	1	.	-1	-1	-1	.139644	1	1	1	-1
2	2	.	1	-1	-1	.431302	-1	-1	1	1
3	3	.	-1	1	-1	.612179	-1	1	-1	1
4	4	.	1	1	-1	.290753	1	-1	-1	-1
5	5	.	-1	-1	1	.155732	1	-1	-1	1
6	6	.	1	-1	1	.699504	-1	1	-1	-1
7	7	.	-1	1	1	.346299	-1	-1	1	-1
8	8	.	1	1	1	.445639	1	1	1	1
9										

Next, to randomize the sequence of the trials, we apply **Data/Sort Cases…**, using **random** as the sort key. The new arrangement of the spreadsheet is

[7]The ***SPSS*** function **RV.UNIFORM(0,1)** generates random numbers from the uniform distribution between 0 and 1. This means roughly that each number in that range has the same chance of occurring. **Of course, when you try this you will get random numbers that are different from ours.**


RUNEXPB.sav - SPSS Data Editor

File Edit View Data Transform Analyze Graphs Utilities Window Help

11 : laptime

	stseq	laptime	shoes	handwt	weight	random	sxh	sxw	hxw	sxhxw
1	1	.	-1	-1	-1	.139644	1	1	1	-1
2	5	.	-1	-1	1	.155732	1	-1	-1	1
3	4	.	1	1	-1	.290753	1	-1	-1	-1
4	7	.	-1	1	1	.346299	-1	-1	1	-1
5	2	.	1	-1	-1	.431302	-1	-1	1	1
6	8	.	1	1	1	.445639	1	1	1	1
7	3	.	-1	1	-1	.612179	-1	1	-1	1
8	6	.	1	-1	1	.699504	-1	1	-1	-1
9										

The order shown immediately above is that in which the eight combinations of factors will be applied.

We note this order by creating one more variable with **Transform**. It is **blseq = $casenum**. The name **blseq** stands for "block sequence" and it will be useful in the analysis.

Save the present spreadsheet as RUNEXP1.sav.

Next, in preparation for setting up a second block of trials. We repeat the process of generating random numbers with **random = RV.UNIFORM(0,1)** and we apply **Data/Sort Cases…** to put the trials in random order. Finally we note the randomized order by resetting **blseq = $casenum**.

Save this file as RUNEXP2.sav .

Now we are ready to record **laptime** for two blocks, each consisting of eight randomized trials.

Actual Data Analysis

The runner recorded the following laptimes on December 5, 1993 while varying the experimental settings according to the randomization scheme in RUNEXP1.sav. They were entered into RUNEXP1.sav in the column for **laptime**:

187.06 188.65, 189.77, 191.82, 186.21, 188.41, 183.65, 193.16

Just in case there might be a nonlinear trend, we use **Transform** to create

blseqsq = blseq*blseq .

Now we are ready to launch **Stepwise Regression** with **laptime** as the dependent variable and **shoes, handwt, weight** and **blseq** as candidates for inclusion in the model. (We do not want to use "the sledgehammer approach" with so few data points.) After initiating the regression routine, this is all that we get for output:

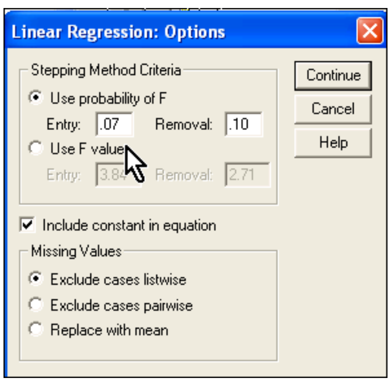
Regression

Variables Entered/Removed[a]

a. Dependent Variable: laptime

This means that **none** of the potential independent variables could enter the model with a p-value low enough to meet the acceptance criteria.

Unwilling to abandon the analysis, we go back to the regression setup, press the **Options…** button, and change the entry value of p just a hair, from 0.05 to 0.07.



We run the stepwise routine again:

Variables Entered/Removed[a]

Model	Variables Entered	Variables Removed	Method
1	weight	.	Stepwise (Criteria: Probability-of-F-to-enter <= .070, Probability-of-F-to-remove >= .100).

a. Dependent Variable: laptime

Model Summary[b]

Model	R	R Square	Adjusted R Square	Std. Error of the Estimate
1	.672[a]	.452	.361	2.43934

a. Predictors: (Constant), weight

b. Dependent Variable: laptime

Coefficients[a]

Model		Unstandardized Coefficients		Standardized Coefficients	t	Sig.
		B	Std. Error	Beta		
1	(Constant)	188.591	.862		218.673	.000
	weight	1.919	.862	.672	2.225	.068

a. Dependent Variable: laptime

The variable **weight** (5-pound backpack) is of borderline significance in slowing down running time. The t-ratio is 2.225, but since the sample is very small, the p-value of 0.068 is slightly above the standard cutoff point of 0.05. If, however, we take the regression coefficient at face value, the following reasoning is interesting:

- The constant 188.6 is the estimated time to run one lap when **weight** = 0, which would, in effect, mean a 2.5 pound backpack.

- 1.92 is the estimated slowing effect of a one-unit increase in **weight**. Since **weight** = -1 means no backpack and **weight** = 1 means the 5-pound backpack, the estimated slowing effect of the backpack is 2x1.92 = 3.84, which is about 2 percent of 188.6. Hence the backpack, which is so light as to be almost unnoticeable, may have a substantial effect.

- Another way to look at it is that 5 pounds, about 3 percent of the runner's body weight of 150 pounds seems to be associated with a 2 percent degradation of running performance.

With so small a sample, diagnostics will not be very informative.

You may recall that back in Section 4 of Chapter 6 when we were discussing **R Square** we said that with a given data set, if you put enough variables into the regression equation you can attain a perfect fit-- i.e., the sum of squared residuals will be zero, and **R Square** = 1. We emphasized the point, with the example of the temperature in Yakutsk, Siberia, that the variables could be utterly nonsensical.

The same is true in our present experiment. When you count the factors, the two-way and the three-way interactions, plus the constant term, there are eight potential elements for the right-hand-side of the equation. With only eight trials, if we force them all in we would have a perfect regression fit, but would be meaningless. We can, however, force in all but the three-way interaction:

Model Summary[b]

Model	R	R Square	Adjusted R Square	Std. Error of the Estimate
1	.758[a]	.575	-1.977	5.26441

a. Predictors: (Constant), hxw, sxw, sxh, weight, handwt, shoes

b. Dependent Variable: laptime

Coefficients[a]

Model		Unstandardized Coefficients		Standardized Coefficients	t	Sig.
		B	Std. Error	Beta		
1	(Constant)	188.591	1.861		101.325	.006
	shoes	.796	1.861	.279	.428	.743
	handwt	-.179	1.861	-.063	-.096	.939
	weight	1.919	1.861	.672	1.031	.490
	sxh	-.119	1.861	-.042	-.064	.959
	sxw	-.521	1.861	-.183	-.280	.826
	hxw	-.216	1.861	-.076	-.116	.926

a. Dependent Variable: laptime

It is useful to compare these results with those from the previous regression on **weight.** First of all, we see that none of independent variables, **weight** included, has a significant t-ratio. This is not because of the effects of multicollinearity (there is none), but rather because the standard error of the residuals, which determines the standard errors of the coefficients, has increased considerably-- **from 2.44 to 5.26-- more than two-fold!** As we know, R Square cannot decrease, and it has increased from 0.452 to 0.575, but look at Adjusted R Square. It is negative and **almost -2!** That is what it should be. It is telling you, in effect, **"Don't do this."** There are not sufficient observations to be trying to fit a six-variable multiple regression, including the constant term, with only eight data points.

> This is what we mean when we use the term "sledgehammer approach." Overfitting can only lead to unreliable and nonsensical analyses.

Analysis of Both Blocks: All 16 Observations

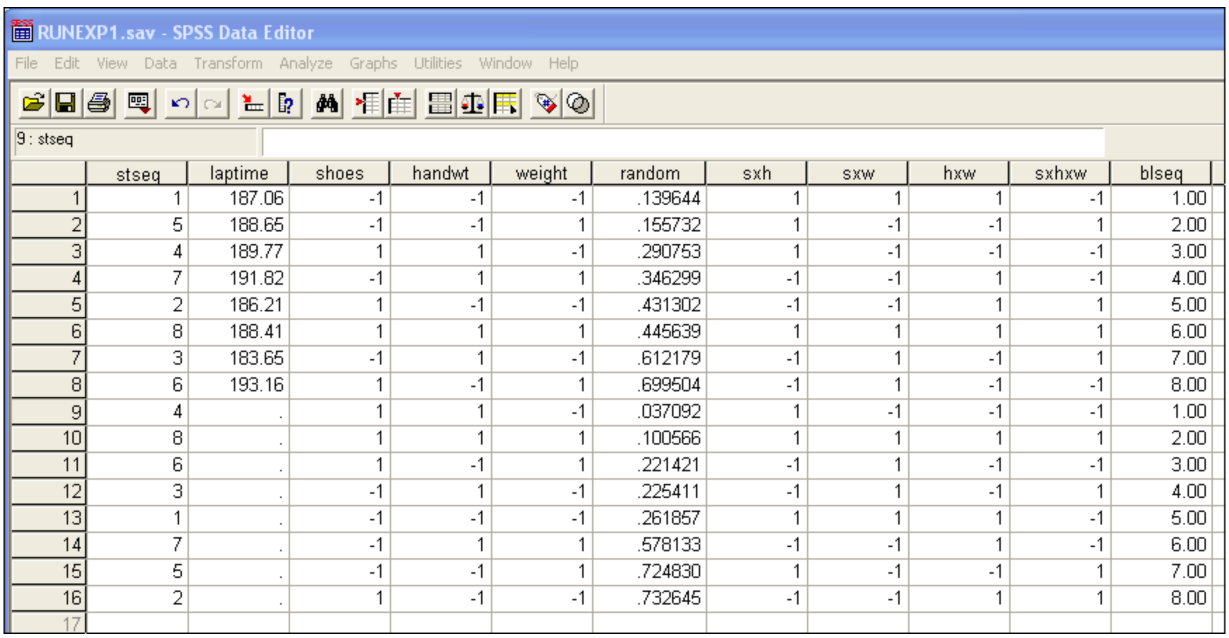
The first block of eight trials, reported above, was run on December 5 in the afternoon. The next morning, December 6, the experimenter ran the second block. He could have analyzed that one separately, but he chose to combine the two blocks in the analysis shown below.

We begin by clearing all of the columns in our present worksheet after **blseqsq**. These columns, if there are any, contain predicted values and residuals from the various regressions that we have performed. We then save our file under the name RUNEXP1,sav. Next it is necessary to open RUNEXP2.sav where we stored the sorted random numbers for the second block of data. Then follow these steps:

- With click-shift-click highlight all of the data cells, even those with missing indicators, but not the variable names.

- Reopen RUNEXP1.sav, and placing the mouse pointer in the cell directly under the number 193.16, paste the cells just copied from RUNEXP2.sav.

The **Data Editor** now looks like this:


RUNEXP1.sav - SPSS Data Editor
File Edit View Data Transform Analyze Graphs Utilities Window Help
9 : stseq

	stseq	laptime	shoes	handwt	weight	random	sxh	sxw	hxw	sxhxw	blseq
1	1	187.06	-1	-1	-1	.139644	1	1	1	-1	1.00
2	5	188.65	-1	-1	1	.155732	1	-1	-1	1	2.00
3	4	189.77	1	1	-1	.290753	1	-1	-1	-1	3.00
4	7	191.82	-1	1	1	.346299	-1	-1	1	-1	4.00
5	2	186.21	1	-1	-1	.431302	-1	-1	1	1	5.00
6	8	188.41	1	1	1	.445639	1	1	1	1	6.00
7	3	183.65	-1	1	-1	.612179	-1	1	-1	1	7.00
8	6	193.16	1	-1	1	.699504	-1	1	-1	-1	8.00
9	4	.	1	1	-1	.037092	1	-1	-1	-1	1.00
10	8	.	1	1	1	.100566	1	1	1	1	2.00
11	6	.	1	-1	1	.221421	-1	1	-1	-1	3.00
12	3	.	-1	1	-1	.225411	-1	1	-1	1	4.00
13	1	.	-1	-1	-1	.261857	1	1	1	-1	5.00
14	7	.	-1	1	1	.578133	-1	-1	1	-1	6.00
15	5	.	-1	-1	1	.724830	1	-1	-1	1	7.00
16	2	.	1	-1	-1	.732645	-1	-1	1	1	8.00
17											

The **laptime** figures for the new block are

196.66, 201.53, 201.38, 197.91, 193.21, 201.04, 203.10, and 193.62.

We enter these into the spreadsheet and then we add the variable **blseqsq** by repeating **blseqsq=blseq*blseq** with **Transform/Compute…**

One more step before running a new regression : We need to create an indicator variable that shows which block the data are from. Calling the new variable **block,** we would like it to equal 0 for the first eight cases and 1 for the last eight. There are a couple of ways to do this using **$casenum** and **Transform/ Compute…** or via **Transform/Recode**, but since we are dealing with only sixteen cases, the easiest thing to do is to insert a new column with the name **block** and then type in eight zeros followed by eight ones.

Finally,

> Save the merged file as RUNEXP3.SX .

We are now ready to run **Stepwise Regression**. We use all of the factors and interactions, plus **blseq, blseqsq**, and the indicator variable, **block**, as candidates for the right-hand side:

Model Summary[c]

Model	R	R Square	Adjusted R Square	Std. Error of the Estimate
1	.840[a]	.706	.685	3.44081
2	.945[b]	.892	.876	2.16060

a. Predictors: (Constant), block

b. Predictors: (Constant), block, weight

c. Dependent Variable: laptime

Coefficients[a]

Model		Unstandardized Coefficients B	Unstandardized Coefficients Std. Error	Standardized Coefficients Beta	t	Sig.
1	(Constant)	188.591	1.217		155.026	.000
	block	9.965	1.720	.840	5.792	.000
2	(Constant)	188.591	.764		246.884	.000
	block	9.965	1.080	.840	9.224	.000
	weight	2.562	.540	.432	4.744	.000

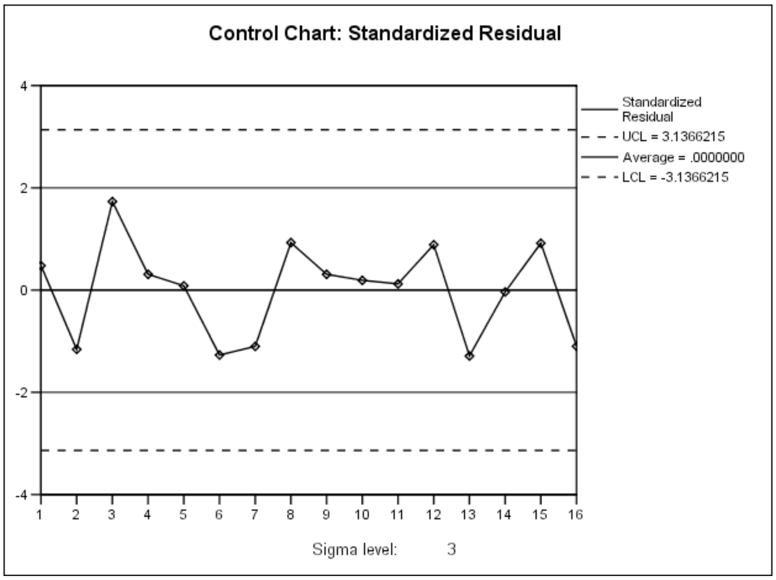
a. Dependent Variable: laptime

We see that the only two variables to enter the model are **block** and **weight.** Note that the **block** coefficient, 9.965, is highly significant. The positive sign for the **block** coefficient means that the next morning the runner was slower by 10 seconds on average than on the previous afternoon. The value of blocking is this: the overall slowing down from one day to the next does not impair our ability to find out about the effect of **weight** (and of possible other variables whose effects might have become visible with the increase of sample size). If, in fact, **block** had not been included, the effect of **weight** would **not** have been significant.

A point to remember in experimental design: Blocking often brings out factor effects that would otherwise be obscured by large standard errors. By removing the block-to-block variability from the sum of squared residuals it makes the standard error of the estimate smaller. Like chicken soup, it can't hurt.

The effect of **weight** is now estimated at 2x2.56 = 5.12. Since 5.12/188.6 = 0.027 or nearly 3 percent, it would appear that a weight handicap of about 3 percent relative to body weight translates to a performance handicap of about 3 percent.[8]

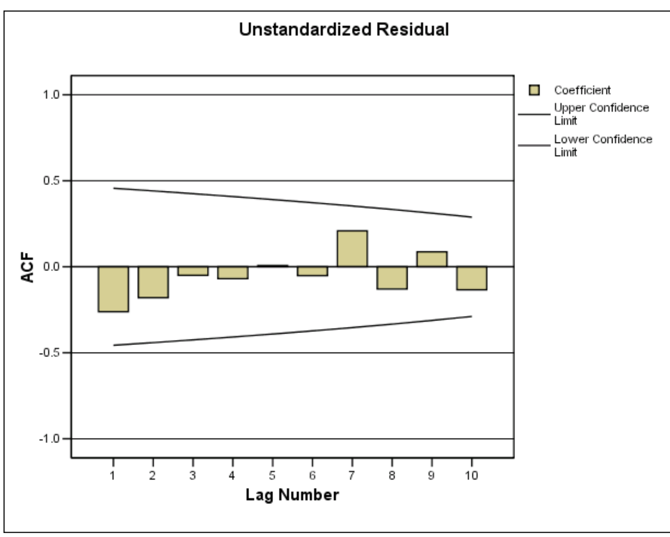
Here are the diagnostic checks:



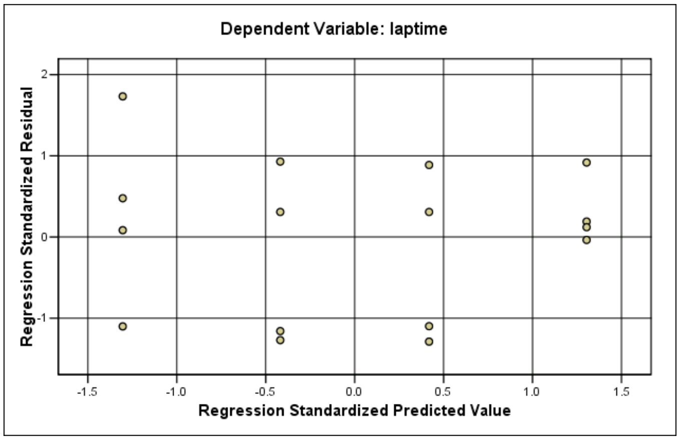
Runs Test

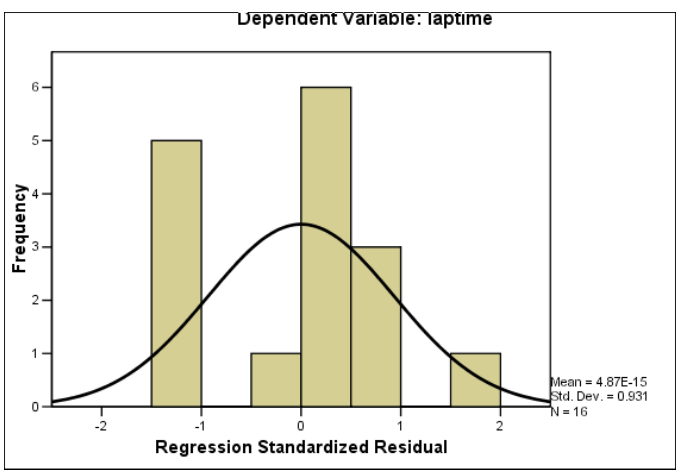
	Unstandardized Residual
Test Value[a]	.0000000
Cases < Test Value	6
Cases >= Test Value	10
Total Cases	16
Number of Runs	8
Z	.000
Asymp. Sig. (2-tailed)	1.000

a. Mean



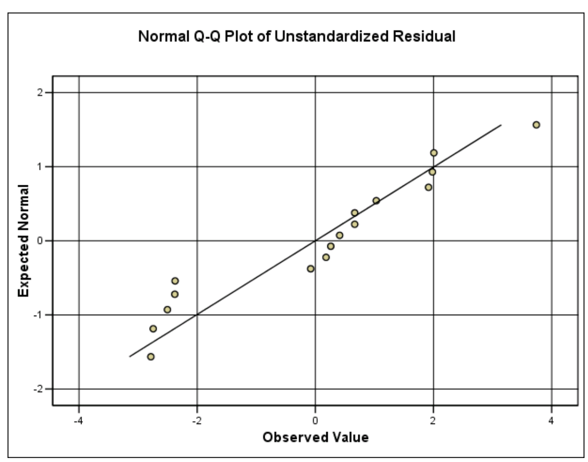
ng the weight-handicap variable in the experiment was
dicap in endurance sports. In age-group running
for many years; then he lost a few pounds and began to





Tests of Normality

	Shapiro-Wilk		
	Statistic	df	Sig.
Unstandardized Residual	.908	16	.109



The histogram does not look very normal, and the S-W statistic is only 0.908. The p-value, however, is still not less than 0.05. This just shows that it is very difficult to make inferences about the shape of an underlying distribution from small samples.[9]

4. A Fractional Factorial Experiment: Top Spinning

We have seen two examples of 2^3 factorial experiments, in which three variables, each at two levels, were simultaneously evaluated. The same idea can be applied to larger numbers of variables. For example, 2^4 = 16, 2^5= 32, etc. But as the number of variables increases, the required minimum sample size goes up rapidly; with 10 variables, for example, we have 2^{10} = 1024.

There is, however, a design approach for dealing with larger numbers of variables, learning a good deal about them, yet using samples of more modest size. We can illustrate the basic approach on a small scale by showing how we can learn about the effects of 4 variables in just 8 trials. The basic design is called a two-level **fractional** factorial design.

Our illustration will be a lightning data set based on the spinning of small toy tops, which are spun by gripping a vertical shaft between two fingers and "snapping" the two fingers. The objective is to find out how to get a long-duration spin. There are four variables:

x1: Two individual tops, one yellow (-1) and one green (1)

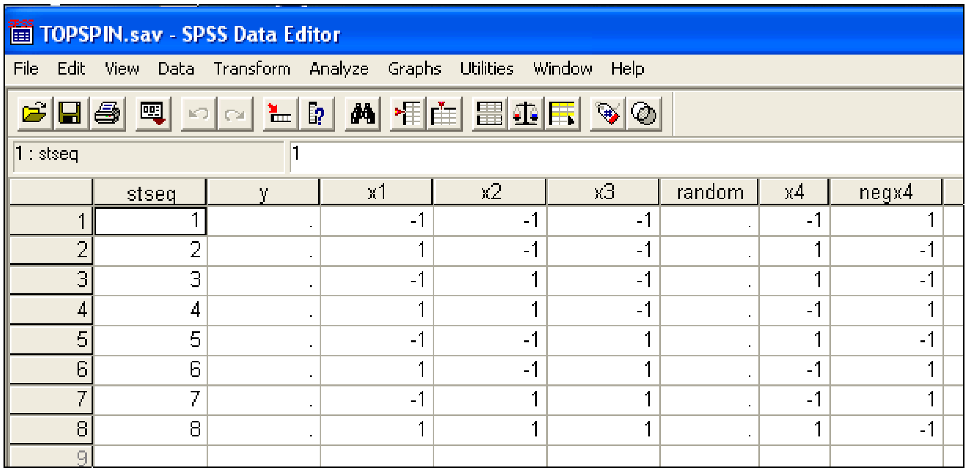
x2: Two ways of gripping the top by thumb and forefinger, "short" (-1) and "full" (1)

x3: Two surfaces on which the tops are spun, "counter" (-1) and "table" (1)

x4: Forearm position: "high elbow" (-1) and "horizontal" (1)

The layout of the experiment is shown in the data file TOPSPIN.sav:

[9]In a footnote in Section 1 of Chapter 4, we have already mentioned the reason for wanting the residuals to conform to normality. It is that we need to make probability statements about unknown quantities-- e.g., regression coefficients, predicted values, etc.-- and the normal distribution is especially easy to work with. We must remark, however, that if the sample is large enough, the distribution of the residuals can be quite non-normal and yet it is safe to base our inferences about the quantities of interest on the normal distribution. This is because many of our estimates are computed as weighted sums of the residuals and it can be shown through a marvelous mathematical result called “the Central Limit Theorem” that weighted sums of non-normal random variables can themselves be approximately normally distributed.



	stseq	y	x1	x2	x3	random	x4	negx4
1	1	.	-1	-1	-1	.	-1	1
2	2	.	1	-1	-1	.	1	-1
3	3	.	-1	1	-1	.	1	-1
4	4	.	1	1	-1	.	-1	1
5	5	.	-1	-1	1	.	1	-1
6	6	.	1	-1	1	.	-1	1
7	7	.	-1	1	1	.	-1	1
8	8	.	1	1	1	.	1	-1
9								

y is the spinning time of the top, left blank to be filled in during the experiment.
random is a random number used for sorting and determining the sequence of trials
x4 is the product of **x1, x2,** and **x3.** It is used to determine forearm position.
negx4 is **x4** multiplied by -1. It can be used for a second $2^{(4-1)}$ fractional factorial experiment which, when separately randomized and pooled with the first, gives a full $2^4 = 16$ experiment, blocked by the two fractional factorial experiments.

Observe that **x1, x2,** and **x3** are laid out as a $2^3 = 8$ full factorial design in three variables. We note that although **x4** has the correct values to complete certain combinations of the four factors, only 8 of the 16 different four-factor combinations are represented. For example, you cannot find the quadruple pattern (- - - +). This is why the design is called a **fractional factorial**. We notice also, however, that the values of **x4** as they are placed in the layout correspond exactly to the values of the product of **x1, x2**, and **x3**, which we would ordinarily call "**x123**", the three-factor interaction of **x1, x2**, and **x3**.

There are often good reasons to believe on a priori grounds that such multi-factor interactions will have little or no effect. **So we let x4 determine the pattern of application for the fourth variable, forearm position**. If, in fact, the three-factor interaction has little or no effect, we can attribute causation to **forearm position** if **x4** turns out significant.

From the data themselves, we cannot distinguish **x4** from the three-factor interaction of **x1, x2**, and **x3**, and the two are said to be **confounded**. But often it will be reasonable to assume that the variable **x4** -- the orientation of the forearm in the spin -- and not the three-factor interaction, will cause any effect that may be found.

(The variable **x5** is the negative of **x4**. It could be used for a second fractional factorial experiment that later could be pooled with the first, using a block indicator variable, in a $2^4 = 16$ full factorial experiment.)

Now we show how the experiment is conducted and analyzed:

These are the random numbers that were generated to fill in the column **random.** We have typed them into the data file and saved the result as TOPSPIN1.sav.

0.529819 0.270049 0.343056 0.121551 0.837784 0.526422 0.408554 0.675279[10]

As before, we sort on **random** and obtain the time ordering that we will use in actually performing the experiment. After some preliminary practicing with the grip and forearm orientation, the experimenter followed the sequence of factor combinations in the layout above and obtained the following spin times, which were entered into the worksheet for **y**:

18.12 13.99 31.50 4.61 10.52 11.83 36.03 30.06

We also add the variable **time=$casenum.** The complete file is then saved as TOPSPIN2.sav.

Next we apply **Stepwise Regression** with **y** as the dependent variable, and **x1, x2, x3**, **x4** and **time** as candidates. For a change, we set the entry probability of F at 0.98 so that all the variables will be forced into the model.

Variables Entered/Removed[a]

Model	Variables Entered	Variables Removed	Method
1	x4	.	Stepwise (Criteria: Probability-of-F-to-enter <= .980, Probability-of-F-to-remove >= .990).
2	x2	.	Stepwise (Criteria: Probability-of-F-to-enter <= .980, Probability-of-F-to-remove >= .990).
3	time	.	Stepwise (Criteria: Probability-of-F-to-enter <= .980, Probability-of-F-to-remove >= .990).
4	x3	.	Stepwise (Criteria: Probability-of-F-to-enter <= .980, Probability-of-F-to-remove >= .990).
5	x1	.	Stepwise (Criteria: Probability-of-F-to-enter <= .980, Probability-of-F-to-remove >= .990).

a. Dependent Variable: y

[10] If you are following along on your PC and you want to obtain the same results as we do, you must copy these random numbers into your spreadsheet. Otherwise if you use your own random numbers you may end up with a different time sequence.

Model Summary[f]

Model	R	R Square	Adjusted R Square	Std. Error of the Estimate
1	.774[a]	.599	.533	7.84607
2	.823[b]	.677	.547	7.72272
3	.877[c]	.769	.596	7.29790
4	.902[d]	.814	.566	7.55977
5	.930[e]	.864	.524	7.91896

a. Predictors: (Constant), x4

b. Predictors: (Constant), x4, x2

c. Predictors: (Constant), x4, x2, time

d. Predictors: (Constant), x4, x2, time, x3

e. Predictors: (Constant), x4, x2, time, x3, x1

f. Dependent Variable: y

Coefficients[a]

Model		Unstandardized Coefficients B	Unstandardized Coefficients Std. Error	Standardized Coefficients Beta	t	Sig.
1	(Constant)	19.583	2.774		7.059	.000
	x4	8.312	2.774	.774	2.997	.024
2	(Constant)	19.583	2.730		7.172	.001
	x4	8.312	2.730	.774	3.044	.029
	x2	2.983	2.730	.278	1.092	.325
3	(Constant)	12.613	6.086		2.072	.107
	x4	7.538	2.652	.702	2.843	.047
	x2	4.144	2.739	.386	1.513	.205
	time	1.549	1.225	.331	1.265	.275
4	(Constant)	7.673	8.560		.896	.436
	x4	6.989	2.821	.651	2.477	.089
	x2	4.967	2.997	.463	1.658	.196
	time	2.647	1.807	.565	1.465	.239
	x3	-3.247	3.807	-.302	-.853	.456
5	(Constant)	3.379	10.272		.329	.773
	x4	6.512	3.007	.607	2.165	.163
	x2	5.683	3.248	.529	1.749	.222
	time	3.601	2.196	.768	1.639	.243
	x3	-4.679	4.323	-.436	-1.082	.392
	x1	2.783	3.248	.259	.857	.482

a. Dependent Variable: y

It looks as if the position of the forearm does matter and that +1, the horizontal orientation is best, with a positive effect of about 8 seconds on spinning time. None of the other variables, when given the opportunity, entered the equation with t-ratios as great as 2 in absolute value.[11] We repeat the simple linear regression on **x4** alone to obtain the residuals:

Model Summary[b]

Model	R	R Square	Adjusted R Square	Std. Error of the Estimate
1	.774[a]	.599	.533	7.84607

a. Predictors: (Constant), x4

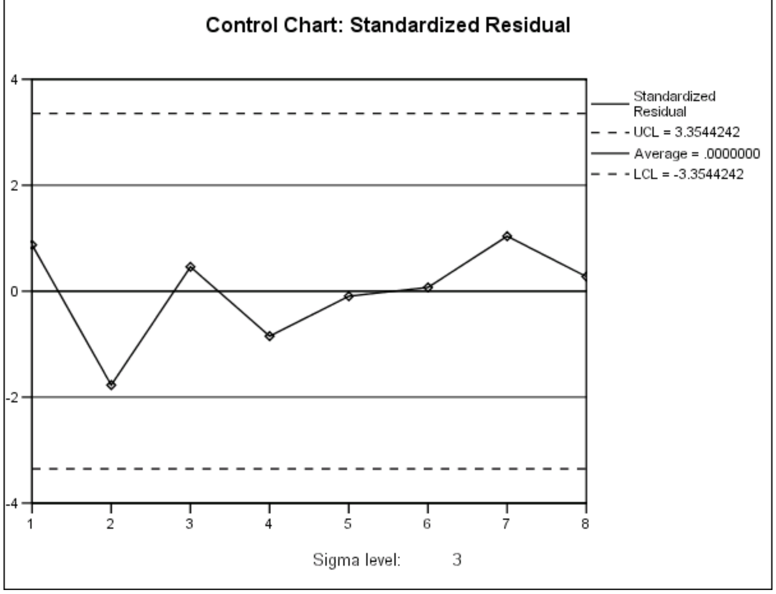
b. Dependent Variable: y

[11] Note, however, that the **Std. Error of the Estimate (Residual Standard Error)** continues to decrease until Step 3, where the model contains the grip variable and time as well as forearm position. It can be proved that if a variable's t-ratio is equal to 1 or greater its inclusion in the model contributes to a reduction in **Adjusted R Square**. Sometimes, if we believe that a variable is important, we will retain it even if its t-ratio is less than 2, but we choose not to do so in the present example.

Coefficients[a]

Model		Unstandardized Coefficients B	Unstandardized Coefficients Std. Error	Standardized Coefficients Beta	t	Sig.
1	(Constant)	19.583	2.774		7.059	.000
	x4	8.312	2.774	.774	2.997	.024

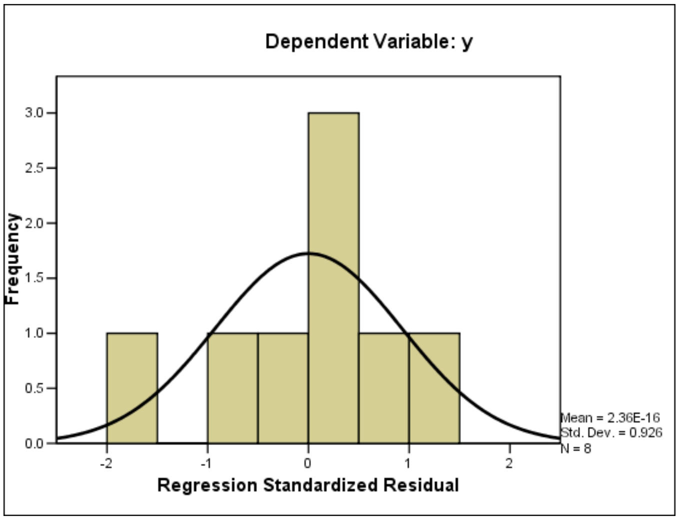
a. Dependent Variable: y



Runs Test

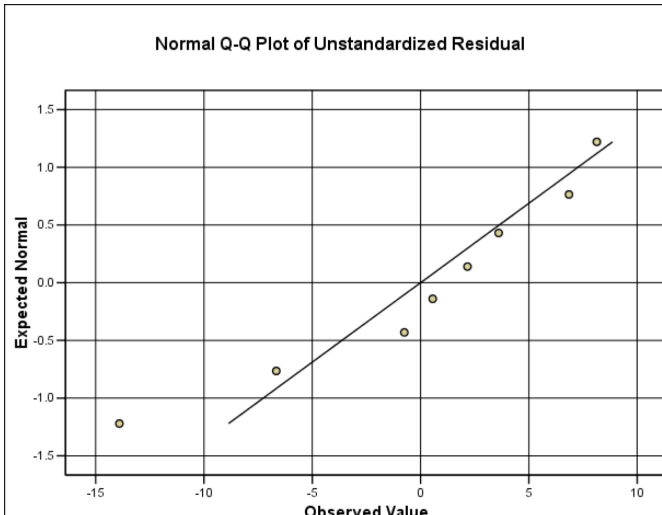
	Unstandardized Residual
Test Value[a]	.0000000
Cases < Test Value	3
Cases >= Test Value	5
Total Cases	8
Number of Runs	5
Z	.000
Asymp. Sig. (2-tailed)	1.000

a. Mean



Tests of Normality

	Shapiro-Wilk Statistic	Shapiro-Wilk df	Shapiro-Wilk Sig.
Unstandardized Residual	.923	8	.453



5. A Pool Experiment: 2^2 Factorial with Blocking

A student had been using an unconventional technique for playing pool: an unorthodox upside-down V bridge and the eye focused on object ball. The standard technique called for a closed bridge and the eye focused on cue ball. Thus there are four variations of technique:

1. Upside-down bridge, object ball (student's technique)

2. Upside-down bridge, cue ball

3. Closed bridge, object ball

4. Closed bridge, cue ball (standard technique)

The design used is built around a 2^2 factorial arrangement of **bridge** and **eye**. The data are contained in the file POOL.sav, for which we show a portion below:

POOL.sav - SPSS Data Editor

File Edit View Data Transform Analyze Graphs Utilities Window Help

40 : block 10

	sess	numshots	bridge	eye	block	order
1	1	50	1	-1	1	1
2	1	39	1	1	1	2
3	1	43	-1	-1	1	3
4	1	78	-1	1	1	4
5	1	62	1	-1	2	5
6	1	40	-1	-1	2	6
7	1	62	-1	1	2	7
8	1	62	1	1	2	8
9	2	32	1	1	3	1
10	2	36	-1	1	3	2
11	2	48	-1	-1	3	3
12	2	45	1	-1	3	4
13	2	54	-1	1	4	5
14	2	46	1	-1	4	6
15	2	55	1	1	4	7
16	2	50	-1	-1	4	8
17	3	39	1	1	5	1
18	3	58	1	-1	5	2
19	3	44	1	1	5	3

If you look at the complete data file you will see that the 40 cases cover five sessions and that each session is blocked into two sets of four games each. The possible combinations of the two factors are randomized within each block. Hence the listing is in the sequence in which the games were played.

Note that the variable **numshots** is the score of each game (the lower the better), **block** numbers the blocks from 1 to 10, and **order** incdicates the sequence of play within each session.

To facilitate the interpretation of later results we shall change the codes of the variables **bridge** and **eye** from (-1,+1) to (0,1). The easiest way to do that is through **Transform**:

bridge = (bridge + 1)/2

eye= (eye + 1)/2

Next, we must perform two other transformations to prepare for the regression analysis:

- **trend=$casenum** to detect any time trend running through the whole data set.

- Using **block** as the source variable, we must create the separate 0-1 block indicators **b1 b2 b3 b4 b5 b6 b7 b8 b9 b10**.

Since setting up the separate block indicator variables is a rather tedious operation, we have prepared a modified data file, POOL1.sav, in which all of the above have been accomplished. Here is a peek at that file:

POOL1.sav - SPSS Data Editor

File Edit View Data Transform Analyze Graphs Utilities Window Help

1 : sess 1

	sess	numshots	bridge	eye	block	order	trend	b1	b2	b3	b4	b5	b6	b7	b8	b9	b10
1	1	50	1	0	1	1	1	1	0	0	0	0	0	0	0	0	0
2	1	39	1	1	1	2	2	1	0	0	0	0	0	0	0	0	0
3	1	43	0	0	1	3	3	1	0	0	0	0	0	0	0	0	0
4	1	78	0	1	1	4	4	1	0	0	0	0	0	0	0	0	0
5	1	62	1	0	2	5	5	0	1	0	0	0	0	0	0	0	0
6	1	40	0	0	2	6	6	0	1	0	0	0	0	0	0	0	0
7	1	62	0	1	2	7	7	0	1	0	0	0	0	0	0	0	0
8	1	62	1	1	2	8	8	0	1	0	0	0	0	0	0	0	0
9	2	32	1	1	3	1	9	0	0	1	0	0	0	0	0	0	0
10	2	36	0	1	3	2	10	0	0	1	0	0	0	0	0	0	0
11	2	48	0	0	3	3	11	0	0	1	0	0	0	0	0	0	0
12	2	45	1	0	3	4	12	0	0	1	0	0	0	0	0	0	0
13	2	54	0	1	4	5	13	0	0	0	1	0	0	0	0	0	0
14	2	46	1	0	4	6	14	0	0	0	1	0	0	0	0	0	0
15	2	55	1	1	4	7	15	0	0	0	1	0	0	0	0	0	0

At this point it will be useful to take a look at the mean values of **numshots** for each of the four **bridge-eye** combinations. We use **Analyze/Reports/OLAP Cubes…**:

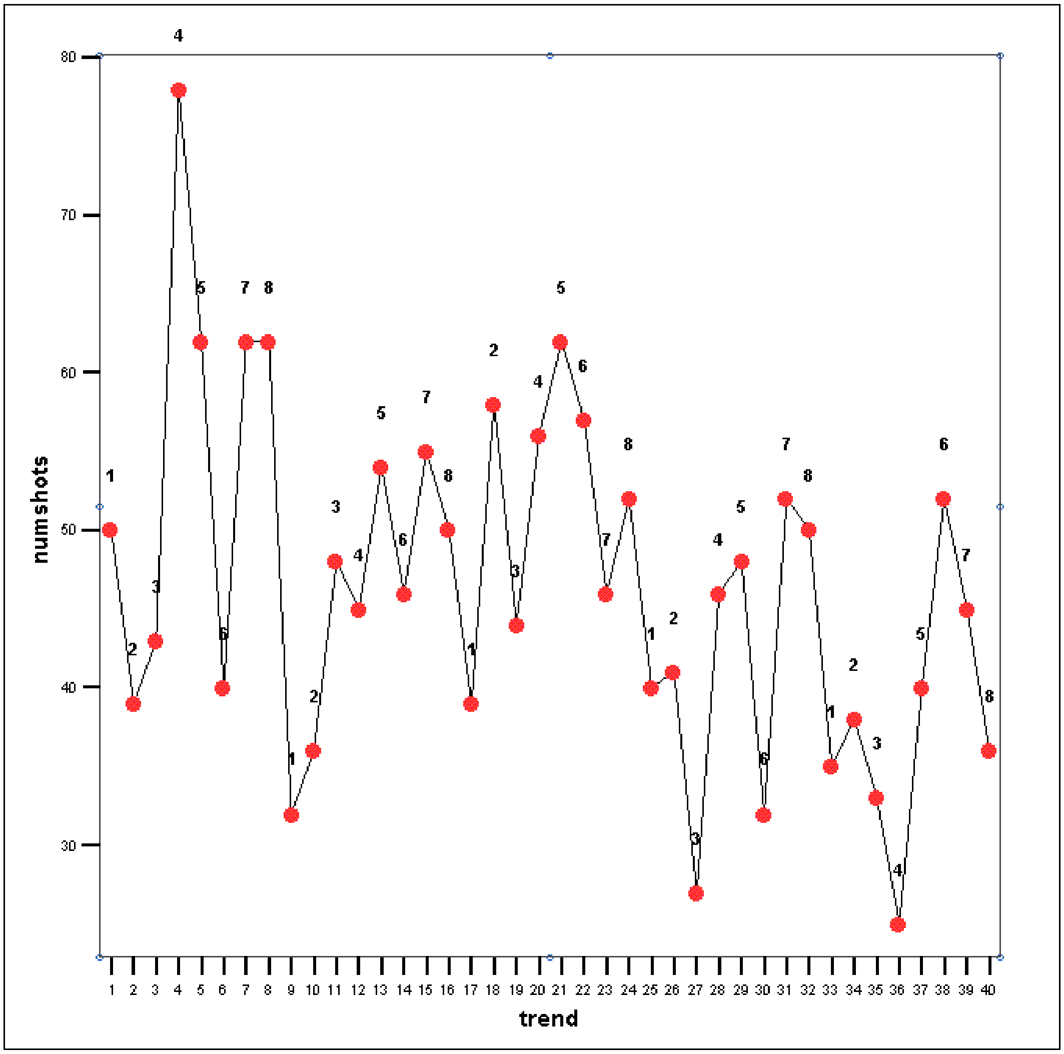
OLAP Cubes

bridge		Mean eye 0	1	Total
0	numshots	46.00	51.50	48.75
1	numshots	48.80	39.30	44.05
Total	numshots	47.40	45.40	46.40

The winning combination, in the lower right of the interior of the table, appears to be **bridge** = 1 and **eye** = 1-- that is, the standard method, not the unorthodox method used by the student. Assuming that this is significant, we are seeing a particular kind of interaction effect that cannot be modeled by (-1,+1) indicator variables. That is, the table suggests that **only when both bridge and eye** are orthodox are good results obtained. Otherwise it doesn't seem to matter.

bridge and **eye** , defined as (0,1) indicator variables, will be multiplied together to obtain **bxe**. **Note that bxe is a (0,1) interaction variable that equals 1 when both bridge and eye are orthodox and equals 0 otherwise**. We will therefore work with **bxe** in our development of the regression model.

The next step is to display **numshots** as a time series:



For more flexibility in the plotting, instead of using **Control Chart,** we have used an **Interactive Scatter Plot**, and after considerable editing we have been able to show the **order** within each daily session for each point in the graph.

There is a clear downward trend overall as well as an indication of nonconstant variance. Within each session, however, there appears to be an uptrend. (Could this be the effect of fatigue?)

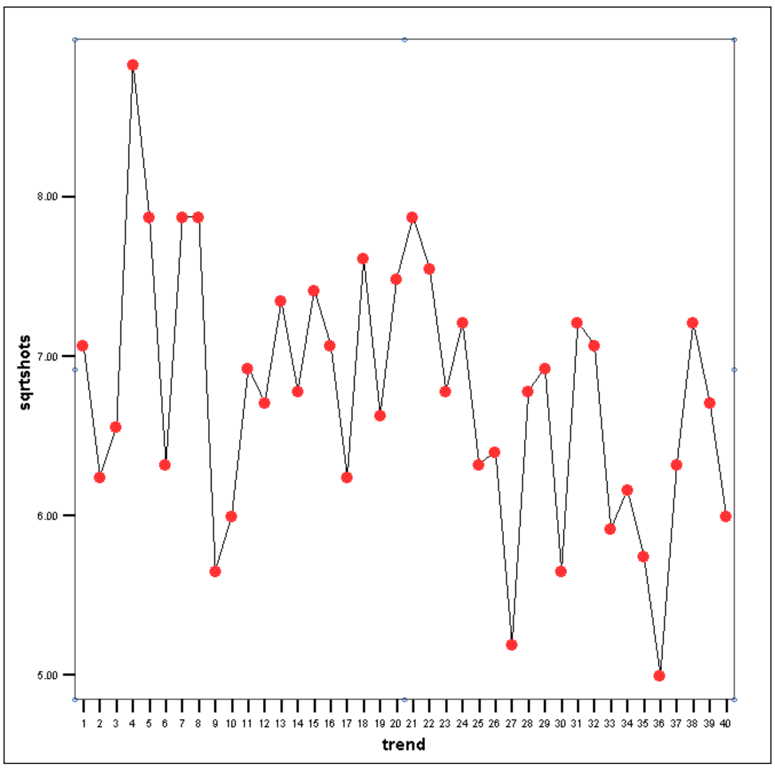
To deal with the problem of nonconstant variance, we will try a square root transformation of **numshots**:

sqrtshots = SQRT(numshots)

Descriptive Statistics

	N	Minimum	Maximum	Mean	Std. Deviation
sqrtshots	40	5.00	8.83	6.7650	.80686
Valid N (listwise)	40				

The standard deviation, 0.8069, is high compared to the 0.5000 that we expect from an in-control process that follows a Poisson distribution, but let's see if the variance is now more nearly constant:



The square root transformation has been quite successful in stabilizing the variance. Remember, however, that these variance-stabilizing transformations cannot turn nonrandom data into random data. We must remove the trends and autocorrelation by other means. We are now ready to apply our regression techniques. We begin with **Stepwise Regression**, using **sqrtshots** for the dependent variable and **bridge, eye, bxe, order, trend**, and all of the **block** indicators as potential independent variables.

Model Summary

Model	R	R Square	Adjusted R Square	Std. Error of the Estimate
1	.443[a]	.196	.175	.73284
2	.600[b]	.360	.325	.66292
3	.669[c]	.448	.402	.62397
4	.724[d]	.523	.469	.58796
5	.762[e]	.580	.519	.55984

a. Predictors: (Constant), b9

b. Predictors: (Constant), b9, bxe

c. Predictors: (Constant), b9, bxe, b7

d. Predictors: (Constant), b9, bxe, b7, b3

e. Predictors: (Constant), b9, bxe, b7, b3, trend

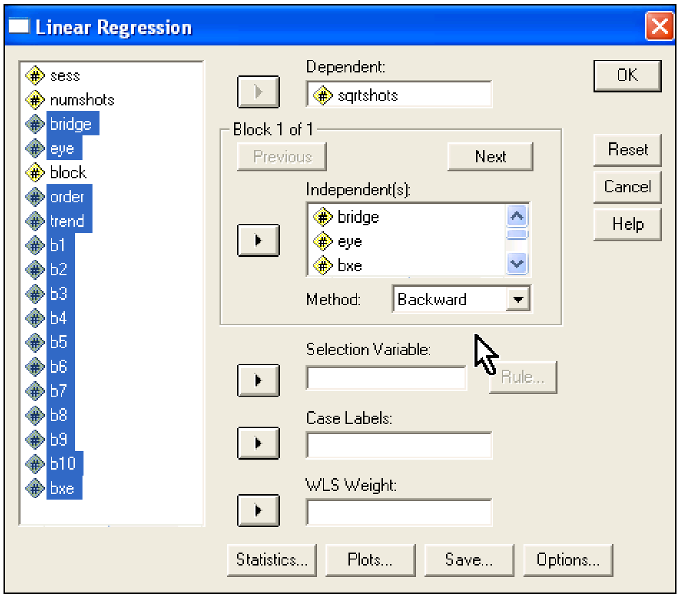
Coefficients[a]

Model		Unstandardized Coefficients		Standardized Coefficients		
		B	Std. Error	Beta	t	Sig.
1	(Constant)	6.883	.122		56.351	.000
	b9	-1.176	.386	-.443	-3.046	.004
2	(Constant)	7.069	.126		56.112	.000
	b9	-1.176	.349	-.443	-3.367	.002
	bxe	-.744	.242	-.404	-3.072	.004
3	(Constant)	7.157	.124		57.650	.000
	b9	-1.265	.331	-.476	-3.822	.001
	bxe	-.744	.228	-.404	-3.264	.002
	b7	-.794	.331	-.299	-2.401	.022
4	(Constant)	7.249	.123		58.748	.000
	b9	-1.357	.314	-.511	-4.318	.000
	bxe	-.744	.215	-.404	-3.464	.001
	b7	-.887	.314	-.334	-2.822	.008
	b3	-.740	.314	-.279	-2.355	.024
5	(Constant)	7.613	.206		36.925	.000
	b9	-1.062	.329	-.400	-3.225	.003
	bxe	-.739	.204	-.401	-3.613	.001
	b7	-.745	.306	-.280	-2.430	.021
	b3	-.904	.309	-.340	-2.927	.006
	trend	-.019	.009	-.277	-2.145	.039

a. Dependent Variable: sqrtshots

We see that the choice of independent variables by the stepwise procedure is **b9, bxe, b7, b3, and trend** with an **R Square** of 0.580 and **Adjusted R Square** equal to 0.519.

There is another way to run **Stepwise Regression** that should be explored. It begins by attempting to enter all of the potential independent variables and then, one-by-one, eliminating a variable at each step until the model can find no more to reject because of too high a significance level. The method is called **backward elimination**. Here is the dialog window setup:



We have highlighted all of the variables except **block**, **sess** and **numshots** and entered them in the **Independent(s):** list. Note also that box for **Method:** is now set to **Backward**.

Because the output table showing the successive sets of coefficients is very long, we show only the last part of it below:

Variables Entered/Removed[b]

Model	Variables Entered	Variables Removed	Method
1	trend, eye, bridge, b6, b5, b7, b4, b8, b3, b9, b2, bxe, order[a]	.	Enter
2	.	b7	Backward (criterion: Probability of F-to-remove >= .100).
3	.	b8	Backward (criterion: Probability of F-to-remove >= .100).
4	.	b9	Backward (criterion: Probability of F-to-remove >= .100).
5	.	b6	Backward (criterion: Probability of F-to-remove >= .100).
6	.	b2	Backward (criterion: Probability of F-to-remove >= .100).
7	.	b4	Backward (criterion: Probability of F-to-remove >= .100).
8	.	b3	Backward (criterion: Probability of F-to-remove >= .100).
9	.	bridge	Backward (criterion: Probability of F-to-remove >= .100).
10	.	eye	Backward (criterion: Probability of F-to-remove >= .100).

a. Tolerance = .000 limits reached.

b. Dependent Variable: sqrtshots

Model Summary

Model	R	R Square	Adjusted R Square	Std. Error of the Estimate
1	.834[a]	.695	.543	.54532
2	.834[b]	.695	.560	.53537
3	.833[c]	.693	.573	.52746
4	.831[d]	.691	.585	.52003
5	.829[e]	.687	.593	.51500
6	.819[f]	.671	.586	.51943
7	.814[g]	.663	.589	.51712
8	.802[h]	.643	.579	.52378
9	.785[i]	.617	.561	.53480
10	.773[j]	.597	.551	.54064

a. Predictors: (Constant), trend, eye, bridge, b6, b5, b7, b4, b8, b3, b9, b2, bxe, order

b. Predictors: (Constant), trend, eye, bridge, b6, b5, b4, b8, b3, b9, b2, bxe, order

c. Predictors: (Constant), trend, eye, bridge, b6, b5, b4, b3, b9, b2, bxe, order

d. Predictors: (Constant), trend, eye, bridge, b6, b5, b4, b3, b2, bxe, order

e. Predictors: (Constant), trend, eye, bridge, b5, b4, b3, b2, bxe, order

f. Predictors: (Constant), trend, eye, bridge, b5, b4, b3, bxe, order

g. Predictors: (Constant), trend, eye, bridge, b5, b3, bxe, order

h. Predictors: (Constant), trend, eye, bridge, b5, bxe, order

i. Predictors: (Constant), trend, eye, b5, bxe, order

j. Predictors: (Constant), trend, b5, bxe, order

Coefficients[a]

	order	.199	.043	.573	4.684	.000
	trend	-.038	.008	-.549	-4.873	.000
7	(Constant)	6.571	.316		20.825	.000
	bridge	.351	.236	.220	1.489	.146
	eye	.450	.233	.283	1.936	.062
	bxe	-1.318	.332	-.717	-3.971	.000
	b3	-.411	.302	-.155	-1.362	.183
	b5	.549	.293	.207	1.875	.070
	order	.189	.041	.544	4.654	.000
	trend	-.036	.007	-.524	-4.842	.000
8	(Constant)	6.381	.287		22.254	.000
	bridge	.372	.238	.234	1.564	.127
	eye	.462	.235	.290	1.963	.058
	bxe	-1.346	.336	-.732	-4.011	.000
	b5	.638	.289	.240	2.209	.034
	order	.206	.039	.593	5.270	.000
	trend	-.034	.007	-.487	-4.590	.000
9	(Constant)	6.617	.249		26.582	.000
	eye	.275	.207	.173	1.330	.192
	bxe	-.970	.239	-.527	-4.053	.000
	b5	.615	.295	.231	2.086	.045
	order	.196	.039	.562	4.970	.000
	trend	-.034	.007	-.487	-4.495	.000
10	(Constant)	6.711	.241		27.815	.000
	bxe	-.787	.198	-.428	-3.980	.000
	b5	.614	.298	.231	2.060	.047
	order	.195	.040	.561	4.905	.000
	trend	-.034	.008	-.487	-4.447	.000

a. Dependent Variable: sqrtshots

We have an interesting result that shows that the various methods of stepwise regression do not always produce the same model. The regular stepwise method ended with the model containing **b9, bxe, b7, b3, and trend** with an **R Square** of 0.580 and **Adjusted R Square** equal to 0.519. The present approach (backward elimination) results in a more parsimonious model with four independent variables—**bxe, b5, order,** and **trend.** Furthermore, **R Square** and **Adjusted R Square** are higher with values 0.597 and 0.551, respectively. Another reason that we might consider this new model to be better is that it contains **order,** which, in the time series plot, appeared to have some importance. If you look back at the **Model Summary** table on the previous page you see that the maximum **Adjusted R Square** (and therefore the minimum residual standard error) is attained at Step 5, at which the model contains **trend, eye, bridge, b5, b4, b3, b2, bxe** and **order**. Although the fit is optimal, all of the included block effects have t-ratios less than 2 and it is difficult to make much sense of them.

As you gain experience in working with linear regression modeling you will see that the choice of the best model is based on a combination of best fit criteria, significance levels, the behavior of residuals, the principle of parsimony, and, perhaps most important of all, what makes sense in the context of what we know about the process under investigation. Accordingly, we, the authors, go one step further by questioning the necessity of the block variable **b5** in our most parsimonious model. There seems to be no rhyme nor reason in its having an effect on the dependent variable. We therefore will drop it also from the model and proceed to analyze the regression based only on **bxe, order**, and **trend**:

Model Summary[b]

Model	R	R Square	Adjusted R Square	Std. Error of the Estimate
1	.740[a]	.548	.511	.56447

a. Predictors: (Constant), trend, bxe, order

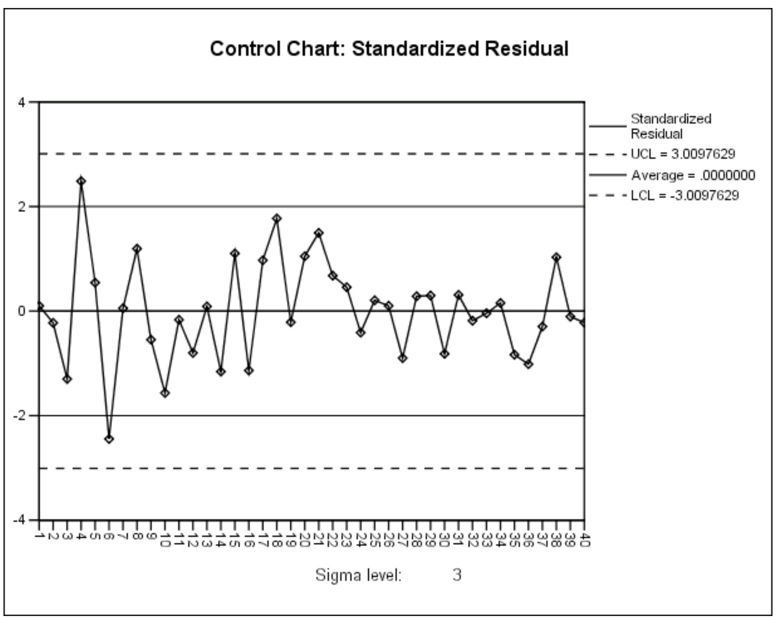
b. Dependent Variable: sqrtshots

Coefficients[a]

Model		Unstandardized Coefficients B	Unstandardized Coefficients Std. Error	Standardized Coefficients Beta	t	Sig.
1	(Constant)	6.877	.238		28.943	.000
	bxe	-.780	.206	-.424	-3.782	.001
	order	.172	.040	.494	4.313	.000
	trend	-.034	.008	-.487	-4.259	.000

a. Dependent Variable: sqrtshots

We see that by dropping **b5** there is a decrease in **Adjusted R Square** from 0.551 to 0.511, but we believe that it is a price worth paying for the additional simplicity and interpretability of the model. You may well disagree, but we shall proceed with the diagnostics:



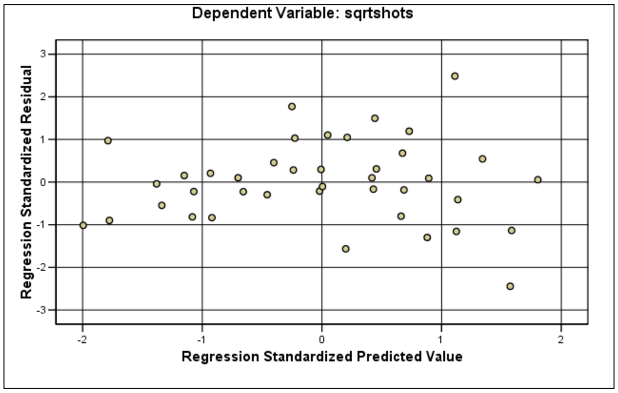
Although we said the transforming to **sqrtshots** alleviated the problem of nonconstant variance there is a wider swing at the beginning of the series. It may be just a start-up effect. Here are the runs test and autocorrelation results:

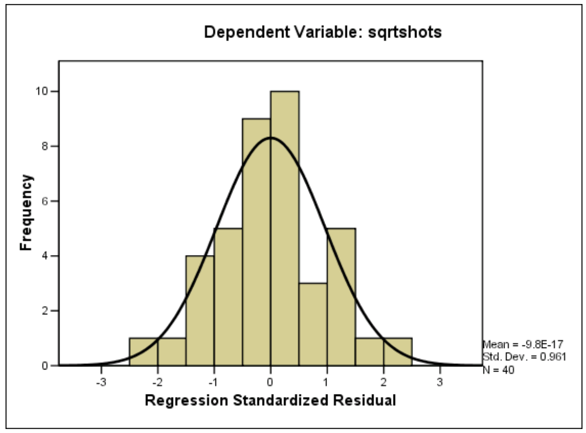
Runs Test

	Unstandardized Residual
Test Value[a]	.0000000
Cases < Test Value	20
Cases >= Test Value	20
Total Cases	40
Number of Runs	24
Z	.801
Asymp. Sig. (2-tailed)	.423

a. Mean

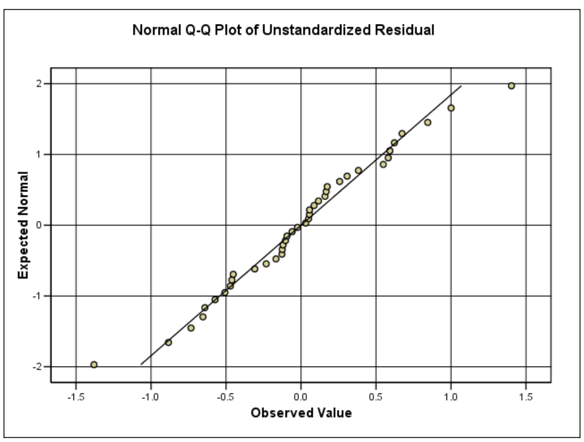
```
Autocorrelations:   RES_1   Unstandardized Residual

     Auto- Stand.
Lag  Corr.   Err. -1  -.75  -.5 -.25   0   .25  .5   .75   1   Box-Ljung  Prob.
                   |----|----|----|----|----|----|----|----|
  1  -.073   .152                .    *|    .                       .231   .631
  2  -.239   .150                .*****|    .                      2.756   .252
  3   .280   .148                .     |******                     6.305   .098
  4   .196   .146                .     |**** .                     8.102   .088
  5  -.096   .144                .   **|    .                      8.541   .129
  6  -.117   .142                .   **|    .                      9.221   .162
  7  -.027   .140                .    *|    .                      9.258   .235
  8  -.088   .138                .   **|    .                      9.662   .290
  9  -.152   .136                 . ***|   .                      10.916   .282
 10   .012   .134                 .    *   .                      10.923   .364
```





Tests of Normality

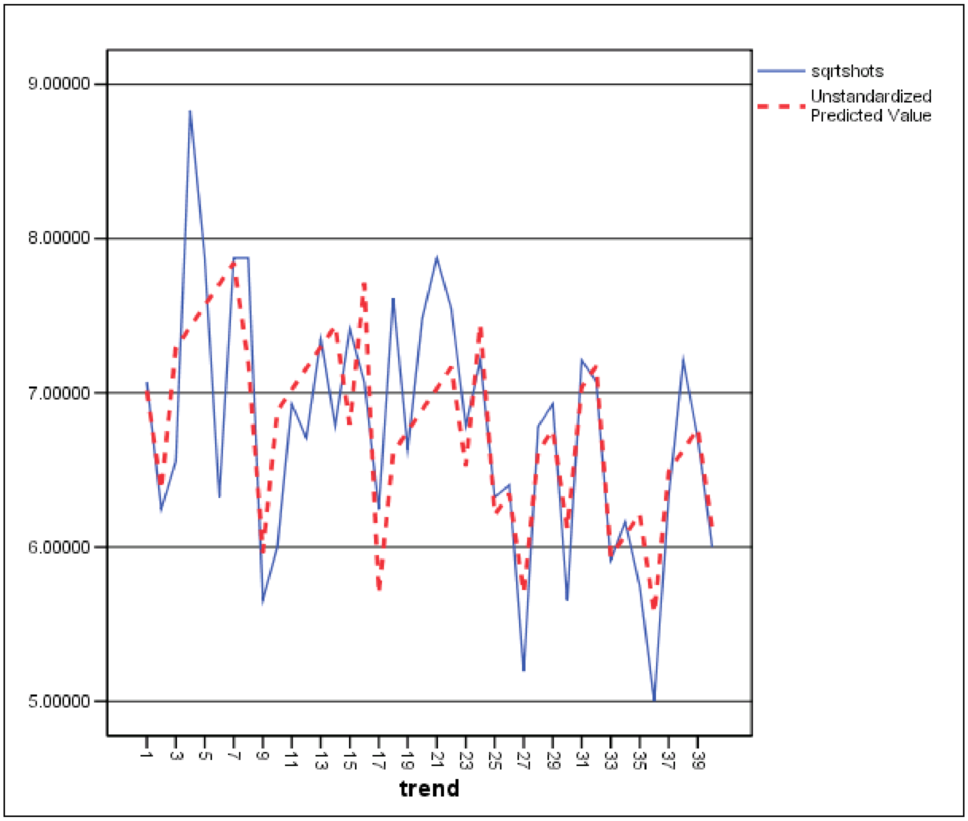
	Shapiro-Wilk		
	Statistic	df	Sig.
Unstandardized Residual	.987	40	.921



All of these diagnostics look very good. In summary, the regression model suggests three important effects:

- **Trend**— An overall downward time trend suggesting improvement with practice.
- **Order--** An upward trend within each daily session, perhaps a fatigue effect.
- **Bridge-eye interaction--** Implies better results with the orthodox method of shooting.

We display below a series plot for both actual and predicted values of **sqrtshots**:



The predicted values track the actual values very well. Next, here is a breakdown table for the predicted value, classified by **bridge and eye**:

OLAP Cubes

		Mean		
		eye		
	bridge	0	1	Total
Unstandardized Predicted Value	0	7.03	6.95	6.99
	1	6.88	6.21	6.54
	Total	6.95	6.58	6.77

The model predicts that using the orthodox method at the present time is the optimal combination of factor settings for the student experimenter. She can expect to have a score of 6.21*6.21= 38.56, well below the average for the data set, which was 46.40. In fact we can go a step further with prediction into the future.

Our model equation is

predicted sqrtshots = 6.877 – 0.780 bxe + 0.172 order – 0.034 trend

We might ask, "What if the player repeats the sequence of eight sessions of pool during the next five days, assuming that the process does not change? What will her score be then?" All that we have to do is to plug the values +1 for **bxe**, 8 for **order**, and 80 for **trend** into the equation and we get a prediction equal to 4.753, implying a score of 22.59!

Do you believe that her improvement will be that great? We certainly do not. It is very unlikely that her downtrend in score would continue in a linear fashion. Real life seldom works that way. Our analysis, however, has pretty well established that the best method of pool shooting for her now is the standard one, i.e., **closed bridge, aim at cue ball**—at least until she gets more practice with the unconventional approaches. If, in the future, she believes that her skill level with other techniques has changed, she can do further experimentation to determine a new, and possibly different, optimal combination of factors. That is what **evolutionary process control** is all about.