

Chapter 41

UCHART Statement

Chapter Table of Contents

OVERVIEW	1419
GETTING STARTED	1420
Creating u Charts from Defect Count Data	1420
Saving Control Limits	1422
Reading Preestablished Control Limits	1424
Creating u Charts from Nonconformities per Unit	1425
Saving Nonconformities per Unit	1428
SYNTAX	1430
Summary of Options	1432
DETAILS	1440
Constructing Charts for Nonconformities per Unit (u Charts)	1440
Output Data Sets	1442
ODS Tables	1445
Input Data Sets	1445
Axis Labels	1448
Missing Values	1448
EXAMPLES	1449
Example 41.1 Applying Tests for Special Causes	1449
Example 41.2 Specifying a Known Expected Number of Nonconformities	1451
Example 41.3 Creating u Charts for Varying Numbers of Units	1452

Chapter 41

UCHAR Statement

Overview

The UCHAR statement creates u charts for the numbers of nonconformities (defects) per inspection unit in subgroup samples containing arbitrary numbers of units.

You can use options in the UCHAR statement to

- specify the number of inspection units per subgroup
- compute control limits from the data based on a multiple of the standard error of the plotted values or as probability limits
- tabulate subgroup summary statistics and control limits
- save control limits in an output data set
- save subgroup summary statistics in an output data set
- read preestablished control limits from a data set
- apply tests for special causes (also known as runs tests and Western Electric rules)
- specify a known (standard) value for the average number of nonconformities per inspection unit
- display distinct sets of control limits for data from successive time phases
- add block legends and symbol markers to reveal stratification in process data
- superimpose stars at points to represent related multivariate factors
- clip extreme points to make the chart more readable
- display vertical and horizontal reference lines
- control axis values and labels
- control layout and appearance of the chart

Getting Started

This section introduces the UCHART statement with simple examples that illustrate commonly used options. Complete syntax for the UCHART statement is presented in the “Syntax” section on page 1430, and advanced examples are given in the “Examples” section on page 1449.

Creating u Charts from Defect Count Data

See SHWUCHR
in the SAS/QC
Sample Library

A textile company uses a *u* chart to monitor the number of defects per square meter of fabric. The fabric is spooled onto rolls as it is inspected for defects. Each piece of fabric is one meter wide and 30 meters in length. The following statements create a SAS data set named FABRIC, which contains the defect counts for 20 rolls:

```
data fabric;
  input roll defects @@;
datalines;
  1 12    2 11    3 9    4 15
  5 7    6 6    7 5    8 10
  9 8    10 8    11 14    12 5
  13 9    14 13    15 7    16 5
  17 8    18 11    19 7    20 12
;
```

A partial listing of FABRIC is shown in Figure 41.1.

Number of Fabric Defects	
roll	defects
1	12
2	11
3	9
4	15
5	7
.	.
.	.
.	.

Figure 41.1. The Data Set FABRIC

There is a single observation per roll. The variable ROLL identifies the subgroup sample and is referred to as the *subgroup-variable*. The variable DEFECTS contains the number of nonconformities (defect count) for each subgroup sample and is referred to as the *process variable* (or *process* for short).

The following statements create the *u* chart shown in Figure 41.2:

```
title 'u Chart for Fabric Defects';
symbol v=dot;
proc shewhart data=fabric;
  uchart defects*roll / subgroupn=30;
run;
```

This example illustrates the basic form of the UCHART statement. After the keyword UCHART, you specify the *process* to analyze (in this case, DEFECTS), followed by an asterisk and the *subgroup-variable* (ROLL).

The SUBGROUPN= option specifies the number of inspection units in each subgroup sample and is required if the input data set is a DATA= data set. In this example, each square meter of fabric is an inspection unit, and each roll is a subgroup sample. The number of inspection units per subgroup can be thought of as the subgroup sample size.

You can use the SUBGROUPN= option to specify one of the following:

- a constant subgroup sample size (as in this example)
- an input variable name whose values contain the subgroup sample sizes (for an example, see “Saving Nonconformities per Unit” on page 1428)

Options such as SUBGROUPN= are specified after the slash (/) in the UCHART statement. A complete list of options is presented in the “Syntax” section on page 1430.

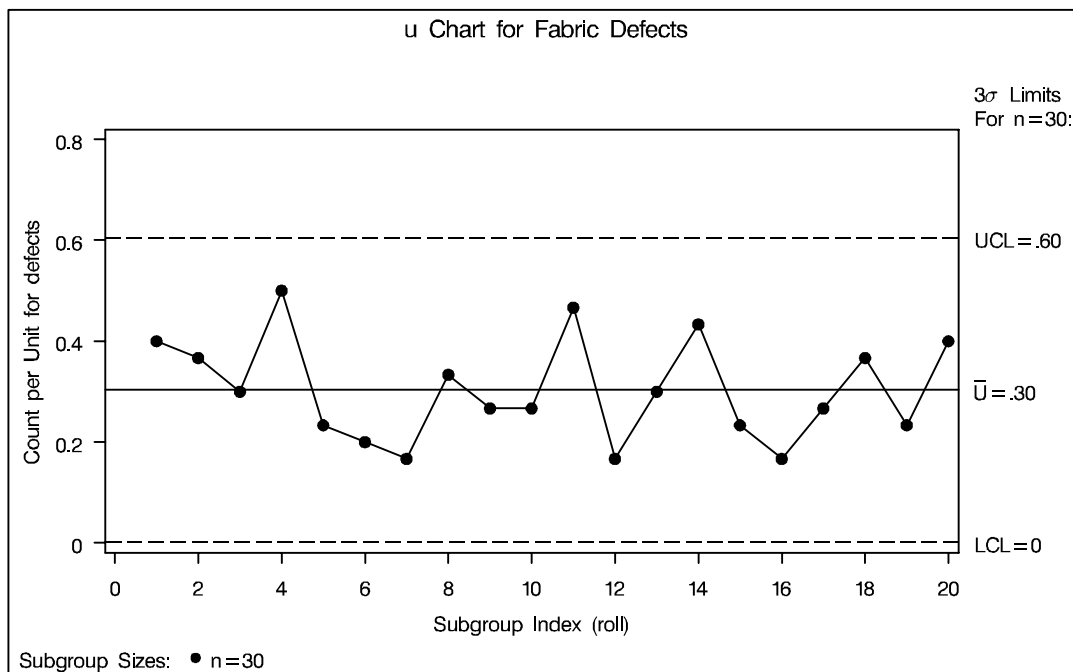


Figure 41.2. *u* Chart Example

The input data set is specified with the DATA= option in the PROC SHEWHART statement.

Each point on the *u* chart represents the number of nonconformities per inspection unit for a particular subgroup. For instance, the value plotted for the first subgroup is $12/30 = 0.4$ (since there are 12 defects on the first roll and this roll contains 30 square meters of fabric). By default, the control limits shown are 3σ limits estimated from the data; the formulas for the limits are given on page 1441. Since none of

the points exceed the 3σ limits, the u chart indicates that the fabric manufacturing process is in statistical control.

See “Constructing Charts for Nonconformities per Unit (u Charts)” on page 1440 for details concerning u charts. For more details on reading defect count data, see “DATA= Data Set” on page 1445.

Saving Control Limits

See SHWUCHR
in the SAS/QC
Sample Library

You can save the control limits for a u chart in a SAS data set; this enables you to apply the control limits to future data (see “Reading Preestablished Control Limits” on page 1424) or modify the limits with a DATA step program.

The following statements read defect counts from the data set FABRIC (see page 1420) and save the control limits displayed in Figure 41.2 in a data set named FABLIM:

```

title 'Control Limits Data Set FABLIM';
proc shewhart data=fabric;
    uchart defects*roll / subgroupn=30
                        outlimits=fabl原因
                        nochart;
run;
    
```

The SUBGROUPN= option specifies the number of inspection units in each subgroup sample. The OUTLIMITS= option names the data set containing the control limits, and the NOCHART option suppresses the display of the chart. The data set FABLIM is listed in Figure 41.3.

Control Limits Data Set FABLIM								
	S		L		S			
	U		I		A	I		
	B	T	M		L	G	L	U
V	G	Y	I		P	M	C	C
A	R	P	T		H	A	L	L
R	P	E	N		A	S	U	U
defects	roll	ESTIMATE	30	.002550178	3	.001671271	0.30333	0.60500

Figure 41.3. The Data Set FABLIM Containing Control Limit Information

The data set FABLIM contains one observation with the limits for *process* DEFECTS. The variables `_LCLU_` and `_UCLU_` contain the lower and upper control limits, and the variable `_U_` contains the central line. The value of `_LIMITN_` is the nominal sample size associated with the control limits, and the value of `_SIGMAS_` is the multiple of σ associated with the control limits. The variables `_VAR_` and `_SUBGRP_` are bookkeeping variables that save the *process* and *subgroup-variable*. The variable `_TYPE_` is a bookkeeping variable that indicates whether the value of `_U_` is an estimate or standard value. For more information, see “OUTLIMITS= Data Set” on page 1442.

Alternatively, you can use the OUTTABLE= option to create an output data set that saves both the control limits and the subgroup statistics, as illustrated by the following statements:

```

title 'Number of Defects Per Square Meter and Control Limits';
proc shewhart data=fabric;
    uchart defects*roll / subgroupn=30
                        outtable=fabtab
                        nochart;
run;

```

The data set FABTAB is listed in Figure 41.4.

Number of Defects Per Square Meter and Control Limits									
		S	L						
		I	I						E
		G	M	S	L	S		U	X
V	r	M	I	U	C	U		C	L
A	o	A	T	B	L	B		L	I
R	l	S	N	N	U	U	U	U	M
	l								
defects	1	3	30	30	.001671271	0.40000	0.30333	0.60500	
defects	2	3	30	30	.001671271	0.36667	0.30333	0.60500	
defects	3	3	30	30	.001671271	0.30000	0.30333	0.60500	
defects	4	3	30	30	.001671271	0.50000	0.30333	0.60500	
defects	5	3	30	30	.001671271	0.23333	0.30333	0.60500	
defects	6	3	30	30	.001671271	0.20000	0.30333	0.60500	
defects	7	3	30	30	.001671271	0.16667	0.30333	0.60500	
defects	8	3	30	30	.001671271	0.33333	0.30333	0.60500	
defects	9	3	30	30	.001671271	0.26667	0.30333	0.60500	
defects	10	3	30	30	.001671271	0.26667	0.30333	0.60500	
defects	11	3	30	30	.001671271	0.46667	0.30333	0.60500	
defects	12	3	30	30	.001671271	0.16667	0.30333	0.60500	
defects	13	3	30	30	.001671271	0.30000	0.30333	0.60500	
defects	14	3	30	30	.001671271	0.43333	0.30333	0.60500	
defects	15	3	30	30	.001671271	0.23333	0.30333	0.60500	
defects	16	3	30	30	.001671271	0.16667	0.30333	0.60500	
defects	17	3	30	30	.001671271	0.26667	0.30333	0.60500	
defects	18	3	30	30	.001671271	0.36667	0.30333	0.60500	
defects	19	3	30	30	.001671271	0.23333	0.30333	0.60500	
defects	20	3	30	30	.001671271	0.40000	0.30333	0.60500	

Figure 41.4. The Data Set FABTAB

This data set contains one observation for each subgroup sample. The variables `_SUBU_` and `_SUBN_` contain the number of nonconformities per unit in each subgroup and the number of inspection units per subgroup. The variables `_LCLU_` and `_UCLU_` contain the lower and upper control limits, and the variable `_U_` contains the central line. The variables `_VAR_` and `ROLL` contain the *process* name and values of the *subgroup-variable*, respectively. For more information, see “OUTTABLE= Data Set” on page 1444.

An OUTTABLE= data set can be read later as a TABLE= data set by the SHEWHART procedure. For example, the following statements read FABTAB and display a *u* chart (not shown here) identical to the chart in Figure 41.2:

```
title 'u Chart for Fabric Defects';
proc shewhart table=fabtab;
    uchart defects*roll / subgroupn=30;
run;
```

Because the SHEWHART procedure simply displays the information in a TABLE= data set, you can use TABLE= data sets to create specialized control charts (see Chapter 49, “Specialized Control Charts”). For more information, see “TABLE= Data Set” on page 1447.

Reading Prestablished Control Limits

See SHWUCHR
in the SAS/QC
Sample Library

In the previous example, control limits were saved in a SAS data set named FABLIM. This example shows how these limits can be applied to defect counts for an additional 20 rolls of fabric, which are provided in the following data set:

```
data fabric2;
    input roll defects @@;
    datalines;
21 9    22 9    23 12    24 7    25 14
26 5    27 4    28 13    29 6    30 7
31 11   32 10   33 7    34 11   35 8
36 8    37 3    38 10   39 9    40 13
;
```

The following statements create a *u* chart for the second group of rolls using the control limits in FABLIM:

```
title 'u Chart for Fabric Defects';
proc shewhart data=fabric2 limits=fablim;
    uchart defects*roll / subgroupn=30;
run;
```

The chart is shown in Figure 41.5 and indicates that the process is in control.

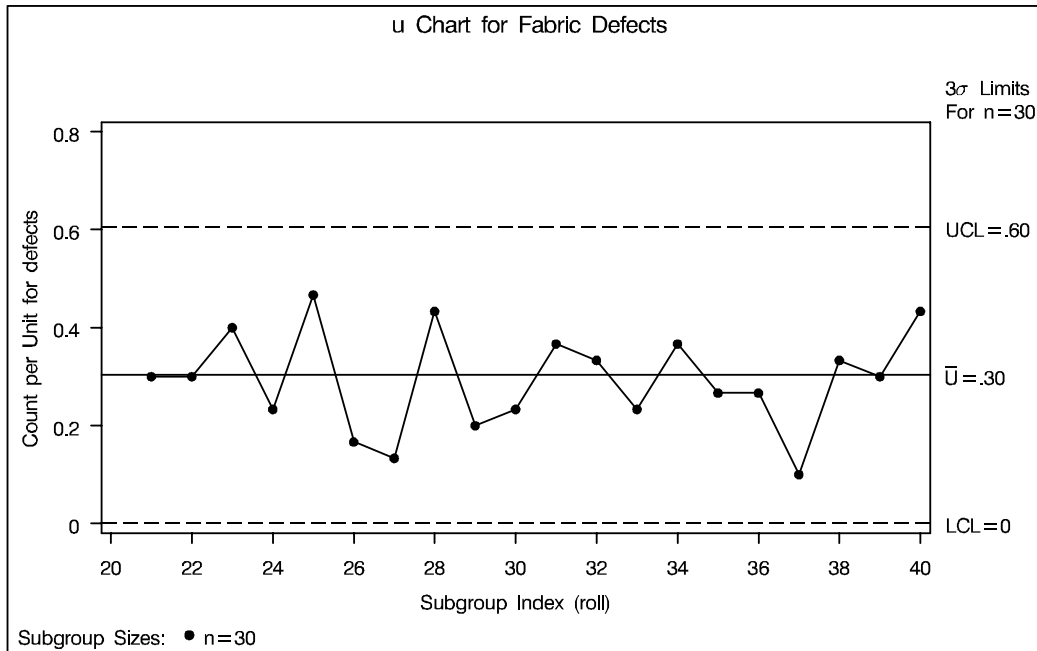


Figure 41.5. A u Chart for Second Set of Fabric Rolls

The LIMITS= option in the PROC SHEWHART statement specifies the data set containing the control limits. By default,* this information is read from the first observation in the LIMITS= data set for which

- the value of `_VAR_` matches the *process* DEFECTS
- the value of `_SUBGRP_` matches the *subgroup-variable* name ROLL

In this example, the LIMITS= data set was created in a previous run of the SHEWHART procedure. You can also create a LIMITS= data set with the DATA step. See “LIMITS= Data Set” on page 1446 for details concerning the variables that you must provide.

Creating u Charts from Nonconformities per Unit

In the previous example, the input data set provided the number of nonconformities for each subgroup sample. However, in some applications, as illustrated here, the data provide the number of nonconformities *per inspection unit* for each subgroup.

See SHWUCHR
in the SAS/QC
Sample Library

A clothing manufacturer ships shirts in boxes of ten. Prior to shipment, each shirt is inspected for flaws. Since the manufacturer is interested in the average number of flaws per shirt, the number of flaws found in each box is divided by ten and then recorded. The following statements create a SAS data set named SHIRTS, which contains the average number of flaws per shirt for 25 boxes:

*In Release 6.09 and in earlier releases, it is also necessary to specify the READLIMITS option to read control limits from a LIMITS= data set.

```

data shirts;
  input box avgdefu @@;
  avgdefn=10;
  datalines;
  1  0.4    2  0.7    3  0.5    4  1.0    5  0.3
  6  0.2    7  0.0    8  0.4    9  0.4   10  0.6
 11  0.2   12  0.7   13  0.3   14  0.1   15  0.3
 16  0.6   17  0.6   18  0.3   19  0.7   20  0.3
 21  0.0   22  0.1   23  0.5   24  0.6   25  0.4
;

```

Note that this is the same data set used in “Getting Started” of Chapter 33, “CCHART Statement.” A partial listing of SHIRTS is shown in Figure 41.6.

Average Number of Shirt Flaws			
	box	avgdefu	avgdefn
	1	0.4	10
	2	0.7	10
	3	0.5	10
	.	.	.
	.	.	.
	.	.	.

Figure 41.6. The Data Set SHIRTS

The data set SHIRTS contains three variables: the box number (BOX), the average number of flaws per shirt (AVGDEFU), and the number of shirts per box (AVGDEFN). Here, a *subgroup* is a box of shirts, and an *inspection unit* is an individual shirt. Note that each subgroup contains ten inspection units.

To create a *u* chart for the average number of flaws per shirt in each box, you can specify SHIRTS as a HISTORY= data set.

```

title 'Total Flaws per Box of Shirts';
symbol v=dot;
proc shewhart history=shirts;
  uchart avgdef*box;
run;

```

Note that AVGDEF is *not* the name of a SAS variable in the data set but is, instead, the common prefix for the names of the SAS variables AVGDEFU and AVGDEFN. The suffix characters *U* and *N* indicate *number of nonconformities per unit* and *sample size*, respectively. This naming convention enables you to specify two variables in the HISTORY= data set with a single name, which is referred to as the *process*. The name BOX, specified after the asterisk, is the name of the *subgroup-variable*. The *u* chart is shown in Figure 41.7.

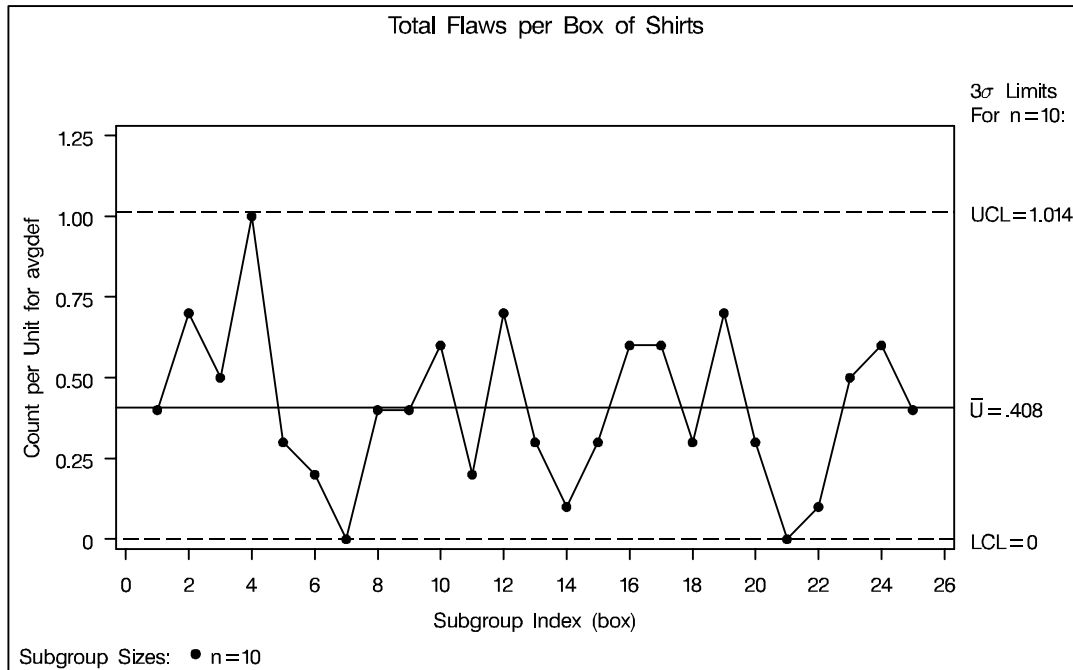


Figure 41.7. A u Chart for Boxes of Shirts

In general, a HISTORY= input data set used with the UCHART statement must contain the following variables:

- subgroup variable
- subgroup number of nonconformities per unit variable
- subgroup sample size variable

Furthermore, the names of the nonconformities per unit and sample size variables must begin with the *process* name specified in the UCHART statement and end with the special suffix characters *U* and *N*, respectively. If the names do not follow this convention, you can use the RENAME option to rename the variables for the duration of the SHEWHART procedure step. Suppose that, instead of the variables AVGDEFU and AVGDEFN, the data set SHIRTS contained the variables SHIRTDEF and SIZES. The following statements temporarily rename SHIRTDEF and SIZES to AVGDEFU and AVGDEFN:

```
proc shewhart
  history=shirts (rename=(shirtdef = avgdefu
                          sizes      = avgdefn ));
  uchart avgdef*box;
run;
```

For more information, see “HISTORY= Data Set” on page 1446.

Saving Nonconformities per Unit

See SHWUCHR
in the SAS/QC
Sample Library

In this example, the UCHART statement is used to create a summary data set containing the number of nonconformities per unit. This data set can be read later by the SHEWHART procedure (as in the preceding example).

A department store receives boxes of shirts containing 10, 25, or 50 shirts. Each box is inspected, and the total number of defects per box is recorded. The following statements create a SAS data set named SHIRTS2, which contains the total defects per box for 20 boxes:

```
data shirts2;
  input box flaws nshirts @@;
  datalines;
  1 3 10 2 8 10 3 15 25 4 20 25
  5 9 25 6 1 10 7 1 10 8 21 50
  9 3 10 10 7 10 11 1 10 12 21 25
  13 9 25 14 3 25 15 12 50 16 18 50
  17 7 10 18 4 10 19 8 10 20 4 10
  ;
```

A partial listing of SHIRTS2 is shown in Figure 41.8.

Number of Shirt Flaws per Box		
box	avgdefu	avgdefn
1	0.4	10
2	0.7	10
3	0.5	10
4	1.0	10
5	0.3	10
.	.	.
.	.	.
.	.	.

Figure 41.8. The Data Set SHIRTS2

The variable BOX contains the box number, the variable FLAWS contains the number of flaws in each box, and the variable NSHIRTS contains the number of shirts in each box. To evaluate the quality of the shirts, you should report the average number of defects per shirt. The following statements create a data set containing the number of flaws per shirt and the number of shirts per box:

```
title 'Average Defects Per Tee Shirt';
proc shewhart data=shirts2;
  uchart flaws*box / subgroupn =nshirts
                  outhistory=shrthist
                  nochart;
run;
```

The SUBGROUPN= option names the variable in the DATA= data set whose values specify the number of inspection units per subgroup. The OUTHISTORY= option names an output data set containing the number of nonconformities per inspection unit and the number of inspection units per subgroup. A partial listing of SHRTHIST is shown in Figure 41.9.

Average Defects Per Tee Shirt		
box	flawsU	flawsN
1	0.30	10
2	0.80	10
3	0.60	25
4	0.80	25
5	0.36	25
.	.	.
.	.	.
.	.	.

Figure 41.9. The Data Set SHRTHIST

There are three variables in the data set SHRTHIST.

- BOX contains the subgroup index.
- FLAWSU contains the numbers of nonconformities per inspection unit.
- FLAWSN contains the subgroup sample sizes.

Note that the variables containing the numbers of nonconformities per inspection unit and subgroup sample sizes are named by adding the suffix characters *U* and *N* to the *process* FLAWS specified in the UCHART statement. In other words, the variable naming convention for OUTHISTORY= data sets is the same as that for HISTORY= data sets.

For more information, see “OUTHISTORY= Data Set” on page 1443.

Syntax

The basic syntax for the UCHART statement is as follows:

```
UCHART process*subgroup-variable ;
```

The general form of this syntax is as follows:

```
UCHART (processes)*subgroup-variable <(block-variables ) >  
      < =symbol-variable | ='character' > < / options >;
```

You can use any number of UCHART statements in the SHEWHART procedure. The components of the UCHART statement are described as follows.

process

processes

identify one or more processes to be analyzed. The specification of *process* depends on the input data set specified in the PROC SHEWHART statement.

- If numbers of nonconformities per subgroup are read from a DATA= data set, *process* must be the name of the variable containing the numbers of nonconformities. For an example, see “Creating u Charts from Defect Count Data” on page 1420.
- If numbers of nonconformities per unit and numbers of inspection units per subgroup are read from a HISTORY= data set, *process* must be the common prefix of the appropriate variables in the HISTORY= data set. For an example, see “Creating u Charts from Nonconformities per Unit” on page 1425.
- If numbers of nonconformities per item, numbers of inspection units per subgroup, and control limits are read from a TABLE= data set, *process* must be the value of the variable `_VAR_` in the TABLE= data set. For an example, see “Saving Control Limits” on page 1422.

A *process* is required. If you specify more than one process, enclose the list in parentheses. For example, the following statements request distinct *u* charts for DEFECTS and FLAWS:

```
proc shewhart data=measures;  
  uchart (defects flaws)*sample / subgroupn=50;  
run;
```

Note that when data are read from a DATA= data set with the UCHART statement, the SUBGROUPN= option (which specifies the number of inspection units per subgroup) is required.

subgroup-variable

is the variable that identifies subgroups in the data. The *subgroup-variable* is required. In the preceding UCHART statement, SAMPLE is the subgroup variable. For details, see “Subgroup Variables” on page 1534.

block-variables

are optional variables that group the data into blocks of consecutive subgroups. The blocks are labeled in a legend, and each *block-variable* provides one level of labels in the legend. See “Displaying Stratification in Blocks of Observations” on page 1684 for an example.

symbol-variable

is an optional variable whose levels (unique values) determine the symbol marker or character used to plot the number of nonconformities per unit.

- If you produce a chart on a line printer, an ‘A’ is displayed for the points corresponding to the first level of the *symbol-variable*, a ‘B’ is displayed for the points corresponding to the second level, and so on.
- If you produce a chart on a graphics device, distinct symbol markers are displayed for points corresponding to the various levels of the *symbol-variable*. You can specify the symbol markers with SYMBOL n statements. See “Displaying Stratification in Levels of a Classification Variable” on page 1683 for an example.

character

specifies a plotting character for charts produced on line printers. For example, the following statements create a u chart using an asterisk (*) to plot the points:

```
proc shewhart data=values;
    uchart defects*sample='*' / subgroupn=100;
run;
```

options

enhance the appearance of the chart, request additional analyses, save results in data sets, and so on. The “Summary of Options” section, which follows, lists all options by function. Chapter 46, “Dictionary of Options,” describes each option in detail.

Summary of Options

The following tables list the UCHART statement options by function. For complete descriptions, see Chapter 46, “Dictionary of Options.”

Table 41.1. Tabulation Options

TABLE	creates a basic table of subgroup sample sizes, subgroup numbers of nonconformities per unit, and control limits
TABLEALL	is equivalent to the options TABLE, TABLECENTRAL, TABLEID, TABLELEGEND, TABLEOUTLIM, and TABLETESTS
TABLECENTRAL	augments basic table with values of central lines
TABLEID	augments basic table with columns for ID variables
TABLELEGEND	augments basic table with legend for tests for special causes
TABLEOUTLIM	augments basic table with columns indicating control limits exceeded
TABLETESTS	augments basic table with a column indicating which tests for special causes are positive

Note that specifying (EXCEPTIONS) after a tabulation option creates a table for exceptional points only.

Table 41.2. Options for Specifying Tests for Special Causes

NO3SIGMACHECK	allows tests to be applied with control limits other than 3σ limits
TESTS= <i>value-list</i> <i>customized-pattern-list</i>	specifies tests for special causes
TEST2RUN= <i>n</i>	specifies length of pattern for Test 2
TEST3RUN= <i>n</i>	specifies length of pattern for Test 3
TESTACROSS	applies tests across <i>phase</i> boundaries
TESTLABEL= <i>'label'</i> <i>(variable)</i> <i>keyword</i>	provides labels for points where test is positive
TESTLABEL <i>n</i> = <i>'label'</i>	specifies label for <i>n</i> th test for special causes
TESTNMETHOD= STANDARDIZE	applies tests to standardized chart statistics
TESTOVERLAP	performs tests on overlapping patterns of points
ZONELABELS	adds labels A, B, and C to zone lines
ZONES	adds lines delineating zones A, B, and C
ZONEVALPOS= <i>n</i>	specifies position of ZONEVALUES labels
ZONEVALUES	labels zone lines with their values

Table 41.3. Graphical Options for Displaying Tests for Special Causes

CTESTS= <i>color</i> <i>test-color-list</i>	specifies color for labels indicating points where test is positive
CZONES= <i>color</i>	specifies color for lines and labels delineating zones A, B, and C
LABELFONT= <i>font</i>	specifies software font for labels at points where test is positive (alias for the TESTFONT= option)
LABELHEIGHT= <i>value</i>	specifies height of labels at points where test is positive (alias for the TESTHEIGHT= option)
LTESTS= <i>linetype</i>	specifies type of line connecting points where test is positive
LZONES= <i>linetype</i>	specifies line type for lines delineating zones A, B, and C
TESTFONT= <i>font</i>	specifies software font for labels at points where test is positive
TESTHEIGHT= <i>value</i>	specifies height of labels at points where test is positive

Table 41.4. Line Printer Options for Displaying Tests for Special Causes

TESTCHAR= <i>'character'</i>	specifies character for line segments that connect any sequence of points for which a test for special causes is positive
ZONECHAR= <i>'character'</i>	specifies character for lines that delineate zones for tests for special causes

Table 41.5. Reference Line Options

CHREF= <i>color</i>	specifies color for lines requested by the HREF= option
CVREF= <i>color</i>	specifies color for lines requested by the VREF= option
HREF= <i>values</i> <i>SAS-data-set</i>	specifies position of reference lines perpendicular to horizontal axis on <i>u</i> chart
HREFCHAR= <i>'character'</i>	specifies line character for HREF= lines
HREFDATA= <i>SAS-data-set</i>	specifies position of reference lines perpendicular to horizontal axis on <i>u</i> chart
HREFLABELS= <i>'label1' ... 'labeln'</i>	specifies labels for HREF= lines
HREFLABPOS= <i>n</i>	specifies position of HREFLABELS= labels
LHREF= <i>linetype</i>	specifies line type for HREF= lines
LVREF= <i>linetype</i>	specifies line type for VREF= lines
NOBYREF	specifies that reference line information in a data set is to be applied uniformly to charts created for all BY groups
VREF= <i>values</i> <i>SAS-data-set</i>	specifies position of reference lines perpendicular to vertical axis
VREFCHAR= <i>'character'</i>	specifies line character for VREF= lines
VREFLABELS= <i>'label1' ... 'labeln'</i>	specifies labels for VREF= lines
VREFLABPOS= <i>n</i>	specifies position of VREFLABELS= labels

Table 41.6. Clipping Options

CCLIP= <i>color</i>	specifies color for plot symbol for clipped points
CLIPCHAR= <i>'character'</i>	specifies plot character for clipped points
CLIPFACTOR= <i>value</i>	determines extent to which extreme points are clipped
CLIPLEGEND= <i>'string'</i>	specifies text for clipping legend
CLIPLEGPOS= <i>keyword</i>	specifies position of clipping legend
CLIPSUBCHAR= <i>'character'</i>	specifies substitution character for CLIPLEGEND= text
CLIPSYMBOL= <i>symbol</i>	specifies plot symbol for clipped points
CLIPSYMBOLHT= <i>value</i>	specifies symbol marker height for clipped points

Table 41.7. Axis and Axis Label Options

CAXIS= <i>color</i>	specifies color for axis lines and tick marks
CFRAME= <i>color</i> <i>(color-list)</i>	specifies fill colors for frame for plot area
CTEXT= <i>color</i>	specifies color for tick mark values and axis labels
HAXIS= <i>values</i> <i>AXISn</i>	specifies major tick mark values for horizontal axis
HEIGHT= <i>value</i>	specifies height of axis label and axis legend text
HMINOR= <i>n</i>	specifies number of minor tick marks between major tick marks on horizontal axis
HOFFSET= <i>value</i>	specifies length of offset at both ends of horizontal axis
INTSTART= <i>value</i>	specifies first major tick mark value for numeric horizontal axis
NOHLABEL	suppresses label for horizontal axis
NOTICKREP	specifies that only the first occurrence of repeated, adjacent sub-group values is to be labeled on horizontal axis
NOTRUNC	suppresses vertical axis truncation at zero applied by default
NOVANGLE	requests vertical axis labels that are strung out vertically
SKIPLABELS= <i>n</i>	specifies thinning factor for tick mark labels on horizontal axis
TURNHLABELS	requests horizontal axis labels that are strung out vertically
VAXIS= <i>values</i> <i>AXISn</i>	specifies major tick mark values for vertical axis
VMINOR= <i>n</i>	specifies number of minor tick marks between major tick marks on vertical axis
VOFFSET= <i>value</i>	specifies length of offset at both ends of vertical axis
VZERO	forces origin to be included in vertical axis for primary chart
VZERO2	forces origin to be included in vertical axis for secondary chart
WAXIS= <i>n</i>	specifies width of axis lines

Table 41.8. Options for Specifying Control Limits

ALPHA= <i>value</i>	requests probability limits for control charts
LIMITN= <i>n</i> VARYING	specifies either nominal sample size for fixed control limits or varying limits
NOREADLIMITS	computes control limits for each <i>process</i> from the data rather than from a LIMITS= data set (Release 6.10 and later releases)
READALPHA	reads the variable <code>_ALPHA_</code> instead of the variable <code>_SIGMAS_</code> from a LIMITS= data set
READINDEXES=ALL <i>'label1'...'labeln'</i>	reads multiple sets of control limits for each <i>process</i> from a LIMITS= data set
READLIMITS	reads single set of control limits for each <i>process</i> from a LIMITS= data set (Release 6.09 and earlier releases)
SIGMAS= <i>k</i>	specifies width of control limits in terms of multiple <i>k</i> of standard error of plotted statistic

Table 41.9. Options for Displaying Control Limits

CINFILL= <i>color</i>	specifies color for area inside control limits
CLIMITS= <i>color</i>	specifies color of control limits, central line, and related labels
LCLLABEL= <i>'label'</i>	specifies label for lower control limit
LIMLABSUBCHAR= <i>'character'</i>	specifies a substitution character for labels provided as quoted strings; the character is replaced with the value of the control limit
LLIMITS= <i>linetype</i>	specifies line type for control limits
NDECIMAL= <i>n</i>	specifies number of digits to right of decimal place in default labels for control limits and central line
NOCTL	suppresses display of central line on <i>u</i> chart
NOLCL	suppresses display of lower control limit
NOLIMITLABEL	suppresses labels for control limits and central line
NOLIMITS	suppresses display of control limits
NOLIMITSFRAME	suppresses default frame around control limit information when multiple sets of control limits are read from a LIMITS= data set
NOLIMITSLEGEND	suppresses legend for control limits
NOLIMIT0	suppresses display of zero lower control limit for <i>u</i> chart
NOUCL	suppresses display of upper control limit
UCLLABEL= <i>'string'</i>	specifies label for upper control limit
USYMBOL= <i>'string'</i> <i>keyword</i>	specifies label for central line
WLIMITS= <i>n</i>	specifies width for control limits and central line

Table 41.10. Grid Options

ENDGRID	adds grid after last plotted point
GRID	adds grid to control chart
LENDGRID= <i>linetype</i>	specifies line type for grid requested with the ENDGRID option
LGRID= <i>linetype</i>	specifies line type for grid requested with the GRID option
WGRID= <i>n</i>	specifies width of grid lines

Table 41.11. Options for Plotting and Labeling Points

ALLLABEL=VALUE (<i>variable</i>)	labels every point
CCONNECT= <i>color</i>	specifies color for line segments that connect points on chart
CFRAMELAB= <i>color</i>	specifies fill color for frame around labeled points
CNEEDLES= <i>color</i>	specifies color for needles that connect points to central line
CONNECTCHAR= ' <i>character</i> '	specifies character used to form line segments that connect points on chart
COUT= <i>color</i>	specifies color for portions of line segments that connect points outside control limits
COUTFILL= <i>color</i>	specifies color for shading areas between the connected points and control limits outside the limits
NEEDLES	connects points to central line with vertical needles
NOCONNECT	suppresses line segments that connect points on chart
OUTLABEL=VALUE (<i>variable</i>)	labels points outside control limits
SYMBOLCHARS= ' <i>characters</i> '	specifies characters indicating <i>symbol-variable</i>
SYMBOLLEGEND= NONE <i>name</i>	specifies LEGEND statement for levels of <i>symbol-variable</i>
SYMBOLORDER= <i>keyword</i>	specifies order in which symbols are assigned for levels of <i>symbol-variable</i>
TURNALL TURNOUT	turns point labels so that they are strung out vertically

Table 41.12. Options for Interactive Control Charts

HTML=(<i>variable</i>)	specifies a variable whose values are URLs to be associated with subgroups
HTML_LEGEND= (<i>variable</i>)	specifies a variable whose values are URLs to be associated with symbols in the symbol legend
TESTURLS= <i>SAS-data-set</i>	associates URLs with tests for special causes
WEBOUT= <i>SAS-data-set</i>	creates an OUTTABLE= data set with additional graphics coordinate data

Table 41.13. Block Variable Legend Options

BLOCKLABELPOS= <i>keyword</i>	specifies position of label for <i>block-variable</i> legend
BLOCKLABTYPE= <i>value keyword</i>	specifies text size of <i>block-variable</i> legend
BLOCKPOS= <i>n</i>	specifies vertical position of <i>block-variable</i> legend
BLOCKREP	repeats identical consecutive labels in <i>block-variable</i> legend
CBLOCKLAB= <i>color</i>	specifies color for filling background in <i>block-variable</i> legend
CBLOCKVAR= <i>variable </i> <i>(variables)</i>	specifies one or more variables whose values are colors for filling background of <i>block-variable</i> legend

Table 41.14. Phase Options

CPHASEBOX= <i>color</i>	specifies color for box enclosing all plotted points for a phase
CPHASEBOX- CONNECT= <i>color</i>	specifies color for line segments connecting adjacent enclosing boxes
CPHASEBOXFILL= <i>color</i>	specifies fill color for box enclosing all plotted points for a phase
CPHASELEG= <i>color</i>	specifies text color for <i>phase</i> legend
CPHASEMEAN- CONNECT= <i>color</i>	specifies color for line segments connecting average value points within a phase
NOPHASEFRAME	suppresses default frame for <i>phase</i> legend
OUTPHASE= <i>'string'</i>	specifies value of the variable <code>_PHASE_</code> in the <code>OUTHISTORY=</code> data set
PHASEBREAK	disconnects last point in a <i>phase</i> from first point in next <i>phase</i>
PHASELABTYPE= <i>value </i> <i>keyword</i>	specifies text size of <i>phase</i> legend
PHASELEGEND	displays <i>phase</i> labels in a legend across top of chart
PHASELIMITS	labels control limits for each phase, provided they are constant within that phase
PHASEMEANSYMBOL= <i>symbol</i>	specifies symbol marker for average of values within a phase
PHASEREF	delineates <i>phases</i> with vertical reference lines
READPHASES= ALL <i>'label1'...'labeln'</i>	specifies <i>phases</i> to be read from an input data set

Table 41.15. Standard Value Options

TYPE= <i>keyword</i>	identifies whether parameters are estimates or standard values and specifies value of the variable <code>_TYPE_</code> in the <code>OUTLIMITS=</code> data set
U0= <i>value</i>	specifies known average number of nonconformities per unit

Table 41.16. Input Data Set Option

MISSBREAK	specifies that observations with missing values are not to be processed
SUBGROUPN= <i>n</i> <i>variable</i>	specifies subgroup sample sizes as constant number <i>n</i> or as values of <i>variable</i> in the DATA= data set

Table 41.17. Output Data Set Options

OUTHISTORY= <i>SAS-data-set</i>	creates output data set containing subgroup numbers of nonconformities per unit and subgroup sample sizes
OUTINDEX='string'	specifies value of the variable <code>_INDEX_</code> in the OUTLIMITS= data set
OUTLIMITS= <i>SAS-data-set</i>	creates output data set containing control limits
OUTTABLE= <i>SAS-data-set</i>	creates output data set containing subgroup numbers of nonconformities per unit, subgroup sample sizes, and control limits

Table 41.18. Plot Layout Options

ALLN	plots numbers of nonconformities per unit for all subgroups
BILEVEL	creates control charts using half-screens and half-pages
EXCHART	creates control charts for a process variable only when exceptions occur
INTERVAL= <i>keyword</i>	specifies natural time interval between consecutive subgroup positions when time, date, or datetime format is associated with a numeric subgroup variable
MAXPANELS= <i>n</i>	specifies maximum number of pages or screens for chart
NMARKERS	requests special markers for points corresponding to sample sizes not equal to nominal sample size for fixed control limits
NOCHART	suppresses creation of chart
NOFRAME	suppresses frame for plot area
NOLEGEND	suppresses legend for subgroup sample sizes
NPANELPOS= <i>n</i>	specifies number of subgroup positions per panel on each chart
REPEAT	repeats last subgroup position on panel as first subgroup position of next panel
TOTPANELS= <i>n</i>	specifies number of pages or screens to be used to display chart
ZEROSTD	displays <i>u</i> chart regardless of whether $\hat{\sigma} = 0$

Table 41.19. Graphical Enhancement Options

ANNOTATE= <i>SAS-data-set</i>	specifies annotate data set that adds features to chart
DESCRIPTION='string'	specifies string that appears in the description field of the PROC GREPLAY master menu
FONT= <i>font</i>	specifies software font for labels and legends on charts
NAME='string'	specifies name that appears in the name field of the PROC GREPLAY master menu
PAGENUM='string'	specifies the form of the label used in pagination
PAGENUMPOS= <i>keyword</i>	specifies the position of the page number requested with the PAGENUM= option

Table 41.20. Star Options

CSTARCIRCLES= <i>color</i>	specifies color for circles specified by the STARCIRCLES= option
CSTARFILL= <i>color</i> <i>(variable)</i>	specifies color for filling stars
CSTAROUT= <i>color</i>	specifies outline color for stars exceeding inner or outer circles
CSTARS= <i>color</i> <i>(variable)</i>	specifies color for outlines of stars
LSTARCIRCLES= <i>linetypes</i>	specifies line types STARCIRCLES= circles
LSTARS= <i>linetype</i> <i>(variable)</i>	specifies line types for outlines of stars requested with the STARVERTICES= option
STARBDRADIUS= <i>value</i>	specifies radius of outer bound circle for vertices of stars
STARCIRCLES= <i>value-list</i>	specifies reference circles for stars
STARINRADIUS= <i>value</i>	specifies inner radius of stars
STARLABEL= <i>keyword</i>	specifies vertices to be labeled
STARLEGEND= <i>keyword</i>	specifies style of legend for star vertices
STARLENDLAB= <i>'label'</i>	specifies label for STARLEGEND= legend
STAROUTRADIUS= <i>value</i>	specifies outer radius of stars
STARSPEC= <i>value</i> <i>SAS-data-set</i>	specifies method used to standardize vertex variables
STARSTART= <i>value</i>	specifies angle for first vertex
STARTYPE= <i>keyword</i>	specifies graphical style of star
STARVERTICES= <i>variable</i> <i>(variables)</i>	superimposes star at each point on chart
WSTARCIRCLES= <i>n</i>	specifies width of circles requested by the STARCIRCLES= option
WSTARS= <i>n</i>	specifies width of stars requested by the STARVERTICES= option

Details

Constructing Charts for Nonconformities per Unit (u Charts)

The following notation is used in this section:

u	expected number of nonconformities per unit produced by process
u_i	number of nonconformities per unit in the i^{th} subgroup. In general, $u_i = c_i/n_i$.
c_i	total number of nonconformities in the i^{th} subgroup
n_i	number of inspection units in the i^{th} subgroup
\bar{u}	average number of nonconformities per unit taken across subgroups. The quantity \bar{u} is computed as a weighted average:
$\bar{u} = \frac{n_1 u_1 + \cdots + n_N u_N}{n_1 + \cdots + n_N} = \frac{c_1 + \cdots + c_N}{n_1 + \cdots + n_N}$	
N	number of subgroups
χ^2_ν	has a central χ^2 distribution with ν degrees of freedom

Plotted Points

Each point on a u chart indicates the number of nonconformities per unit (u_i) in a subgroup. For example, Figure 41.10 displays three sections of pipeline that are inspected for defective welds (indicated by an X). Each section represents a *subgroup* composed of a number of *inspection units*, which are 1000-foot-long sections. The number of units in the i^{th} subgroup is denoted by n_i , which is the subgroup sample size.

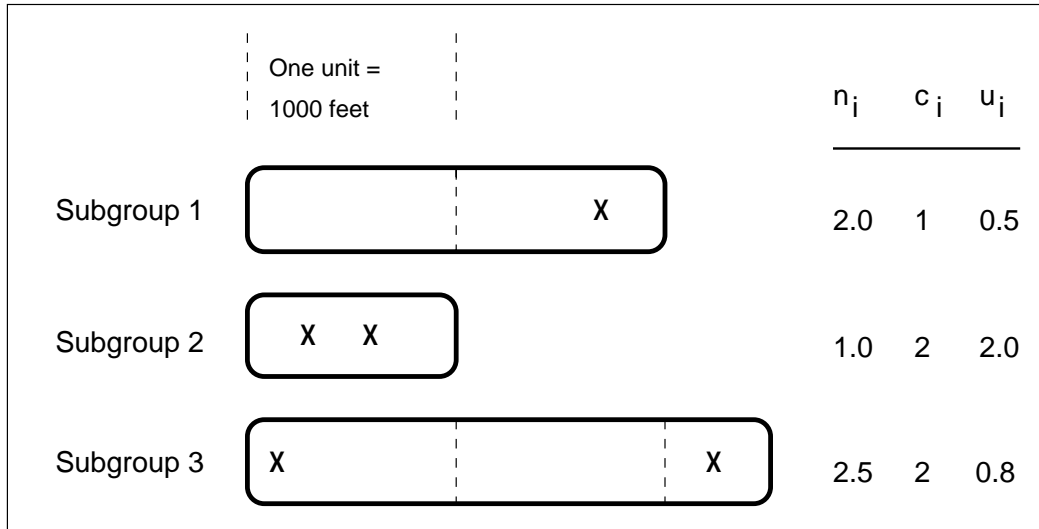


Figure 41.10. Terminology for c Charts and u Charts

The number of nonconformities in the i^{th} subgroup is denoted by c_i . The number of nonconformities per unit in the i^{th} subgroup is denoted by $u_i = c_i/n_i$. In Figure 41.10, the number of defective welds per unit in the third subgroup is $u_3 = 2/2.5 = 0.8$.

A u chart plots the quantity u_i for the i^{th} subgroup. A c chart plots the quantity c_i for the i^{th} subgroup (see Chapter 33, “CCHART Statement”). An advantage of a u chart is that the value of the central line at the i^{th} subgroup does not depend on n_i . This is not the case for a c chart, and consequently, a u chart is often preferred when the number of units n_i is not constant across subgroups.

Central Line

On a u chart, the central line indicates an estimate of u , which is computed as \bar{u} by default. If you specify a known value (u_0) for u , the central line indicates the value of u_0 .

Control Limits

You can compute the limits in the following ways:

- as a specified multiple (k) of the standard error of u_i above and below the central line. The default limits are computed with $k = 3$ (these are referred to as 3σ limits).
- as probability limits defined in terms of α , a specified probability that u_i exceeds the limits

The lower and upper control limits, LCLU and UCLU, respectively, are given by

$$\begin{aligned} \text{LCLU} &= \max\left(\bar{u} - k\sqrt{\bar{u}/n_i}, 0\right) \\ \text{UCLU} &= \bar{u} + k\sqrt{\bar{u}/n_i} \end{aligned}$$

The limits vary with n_i .

The upper probability limit UCLU for u_i can be determined using the fact that

$$\begin{aligned} P\{u_i > \text{UCLU}\} &= 1 - P\{u_i \leq \text{UCLU}\} \\ &= 1 - P\{c_i \leq n_i \text{UCLU}\} \\ &= 1 - P\{\chi_{2(n_i(\text{UCLU}+1))}^2 \geq 2n_i\bar{u}\} \end{aligned}$$

The limit UCLU is then calculated by setting

$$1 - P\{\chi_{2(n_i(\text{UCLU}+1))}^2 \geq 2n_i\bar{u}\} = \alpha/2$$

and solving for UCLU.

Likewise, the lower probability limit LCLC for u_i can be determined using the fact that

$$\begin{aligned} P\{u_i < \text{LCLC}\} &= P\{c_i < n_i \text{LCLC}\} \\ &= P\{\chi_{2(n_i(\text{LCLC}+1))}^2 > 2n_i\bar{u}\} \end{aligned}$$

The limit LCLC is then calculated by setting

$$P\{\chi^2_{2(n_i(LCLC+1))} > 2n_i\bar{u}\} = \alpha/2$$

and solving for LCLC. For more information, refer to Johnson, Kotz, and Kemp (1992). This assumes that the process is in statistical control and that c_i has a Poisson distribution. Note that the probability limits vary with n_i and are asymmetric around the central line. If a standard value u_0 is available for u , replace \bar{u} with u_0 in the formulas for the control limits.

You can specify parameters for the limits as follows:

- Specify k with the SIGMAS= option or with the variable _SIGMAS_ in a LIMITS= data set.
- Specify α with the ALPHA= option or with the variable _ALPHA_ in a LIMITS= data set.
- Specify a constant nominal sample size $n_i \equiv n$ for the control limits with the LIMITN= option or with the variable _LIMITN_ in a LIMITS= data set.
- Specify u_0 with the U0= option or with the variable _U_ in a LIMITS= data set.

Output Data Sets

OUTLIMITS= Data Set

The OUTLIMITS= data set saves control limits and control limit parameters. The following variables can be saved:

Table 41.21. OUTLIMITS= Data Set

Variable	Description
ALPHA	probability (α) of exceeding limits
INDEX	optional identifier for the control limits specified with the OUTINDEX= option
LCLU	lower control limit for number of nonconformities per unit
LIMITN	sample size associated with the control limits
SIGMAS	multiple (k) of standard error of u_i
SUBGRP	<i>subgroup-variable</i> specified in the UCHART statement
TYPE	type (estimate or standard value) of _U_
U	value of central line of u chart (\bar{u} or u_0)
UCLU	upper control limit for number of nonconformities per unit
VAR	<i>process</i> specified in the UCHART statement

Notes:

1. If the control limits vary with subgroup sample size, the special missing value V is assigned to the variables _LCLU_, _UCLU_, and _LIMITN_.

2. If the limits are defined in terms of a multiple k of the standard error of u_i , the value of `_ALPHA_` is computed as $P\{u_i < \text{_LCLU_}\} + P\{u_i > \text{_UCLU_}\}$, provided that n_i is a constant. Otherwise, `_ALPHA_` is assigned the special missing value V .
3. If the limits are probability limits, the value of `_SIGMAS_` is computed as $(\text{_UCLU_} - \text{_U_}) / \sqrt{\text{_U_} / \text{_LIMITN_}}$, provided that n_i is a constant. Otherwise, `_SIGMAS_` is assigned the special missing value V .
4. Optional BY variables are saved in the OUTLIMITS= data set.

The OUTLIMITS= data set contains one observation for each *process* specified in the UCHART statement. For an example, see “Saving Control Limits” on page 1422.

OUTHISTORY= Data Set

The OUTHISTORY= data set saves subgroup summary statistics. The following variables are saved:

- the *subgroup-variable*
- a subgroup number of nonconformities per unit variable named by *process* suffixed with U
- a subgroup sample size variable named by *process* suffixed with N

Given a *process* name that contains eight characters, the procedure first shortens the name to its first four characters and its last three characters, and then it adds the suffix. For example, the procedure shortens the *process* NDEFECTS to NDEFCTS before adding the suffix.

Subgroup summary variables are created for each *process* specified in the UCHART statement. For example, consider the following statements:

```
proc shewhart data=fabric;
    uchart (flaws ndefects)*lot / outhistory=summary
                                subgroupn = 10;
run;
```

The data set SUMMARY contains the variables LOT, FLAWSU, FLAWSN, NDEFCTSU, and NDEFCTSN.

Additionally, the following variables, if specified, are included:

- BY variables
- *block-variables*
- *symbol-variable*
- ID variables
- `_PHASE_` (if the OUTPHASE= option is specified)

For an example of an OUTHISTORY= data set, see “Saving Nonconformities per Unit” on page 1428. Note that an OUTHISTORY= data set created with the UCHART statement can be used as a HISTORY= data set by either the CCHART statement or the UCHART statement.

OUTTABLE= Data Set

The OUTTABLE= data set saves subgroup summary statistics, control limits, and related information. The following variables are saved:

Variable	Description
ALPHA	probability (α) of exceeding control limits
EXLIM	control limit exceeded on u chart
LCLU	lower control limit for number of nonconformities per unit
LIMITN	nominal sample size associated with the control limits
SIGMAS	multiple (k) of the standard error associated with the control limits
<i>subgroup</i>	values of the subgroup variable
SUBU	subgroup number of nonconformities per unit
SUBN	subgroup sample size
TESTS	tests for special causes signaled on u chart
UCLU	upper control limit for number of nonconformities per unit
VAR	<i>process</i> specified in the UCHART statement

In addition, the following variables, if specified, are included:

- BY variables
- *block-variables*
- *symbol-variable*
- ID variables
- _PHASE_ (if the READPHASES= option is specified)

Notes:

1. Either the variable _ALPHA_ or the variable _SIGMAS_ is saved, depending on how the control limits are defined (with the ALPHA= or SIGMAS= option, respectively, or with the corresponding variables in a LIMITS= data set).
2. The variable _TESTS_ is saved if you specify the TESTS= option. The k^{th} character of a value of _TESTS_ is k if Test k is positive at that subgroup. For example, if you request the first four tests (the ones appropriate for u charts) and Tests 2 and 4 are positive for a given subgroup, the value of _TESTS_ has a 2 for the second character, a 4 for the fourth character, and blanks for the other six characters.
3. The variables _VAR_, _EXLIM_, and _TESTS_ are character variables of length 8. The variable _PHASE_ is a character variable of length 16. All other variables are numeric.

For an example, see “Saving Control Limits” on page 1422.

ODS Tables

The following table summarizes the ODS tables that you can request with the UCHART statement.

Table 41.22. ODS Tables Produced with the UCHART Statement

Table Name	Description	Options
UCHART	<i>u</i> chart summary statistics	TABLE, TABLEALL, TABLEC, TABLEID, TABLELEG, TABLEOUT, TABLETESTS
Tests	descriptions of tests for special causes requested with the TESTS= option for which at least one positive signal is found	TABLEALL, TABLELEG

Input Data Sets

DATA= Data Set

You can read defect counts for subgroup samples from a DATA= data set specified in the PROC SHEWHART statement. Each *process* specified in the UCHART statement must be a SAS variable in the data set. This variable provides the defect count (number of nonconformities) for subgroup samples indexed by the *subgroup-variable*. The *subgroup-variable*, specified in the UCHART statement, must also be a SAS variable in the DATA= data set. Each observation in a DATA= data set must contain a value for each *process* and a value for the *subgroup-variable*. The data set should contain one observation per subgroup. When you use a DATA= data set with the UCHART statement, the SUBGROUPN= option (which specifies the number of inspection units per subgroup) is required. Other variables that can be read from a DATA= data set include

- `_PHASE_` (if the READPHASES= option is specified)
- *block-variables*
- *symbol-variable*
- BY variables
- ID variables

By default, the SHEWHART procedure reads all of the observations in a DATA= data set. However, if the data set includes the variable `_PHASE_`, you can read selected groups of observations (referred to as *phases*) with the READPHASES= option (for an example, see “Displaying Stratification in Phases” on page 1689).

For an example of a DATA= data set, see “Creating *u* Charts from Defect Count Data” on page 1420.

LIMITS= Data Set

You can read preestablished control limits (or parameters from which the control limits can be calculated) from a LIMITS= data set specified in the PROC SHEWHART statement. For example, the following statements read control limit information from the data set CONLIMS:*

```
proc shewhart data=info limits=conlims;  
    uchart defects*lot / subgroupn = 10;  
run;
```

The LIMITS= data set can be an OUTLIMITS= data set that was created in a previous run of the SHEWHART procedure. Such data sets always contain the variables required for a LIMITS= data set. The LIMITS= data set can also be created directly using a DATA step. When you create a LIMITS= data set, you must provide one of the following:

- the variables `_LCLU_`, `_U_`, and `_UCLU_`, which specify the control limits
- the variable `_U_`, which is used to calculate the control limits (see page 1441)

In addition, note the following:

- The variables `_VAR_` and `_SUBGRP_` are required. These must be character variables of length 8.
- The variable `_INDEX_` is required if you specify the `READINDEX=` option; this must be a character variable of length 16.
- The variables `_LIMITN_`, `_SIGMAS_` (or `_ALPHA_`), and `_TYPE_` are optional, but they are recommended to maintain a complete set of control limit information. The variable `_TYPE_` must be a character variable of length 8; valid values are `ESTIMATE` and `STANDARD`.
- BY variables are required if specified with a BY statement.

For an example, see “Reading Preestablished Control Limits” on page 1424.

HISTORY= Data Set

You can read subgroup summary statistics from a HISTORY= data set specified in the PROC SHEWHART statement. This allows you to reuse OUTHISTORY= data sets that have been created in previous runs of the SHEWHART procedure or to read output data sets created with SAS summarization procedures.

A HISTORY= data set used with the UCHART statement must contain the following variables:

- *subgroup-variable*
- subgroup number of nonconformities per unit variable for each *process*
- subgroup sample size variable (number of units per subgroup) for each *process*

The names of the variables containing the number of nonconformities per unit and subgroup sample sizes must be the *process* name concatenated with the special suffix characters *U* and *N*, respectively. For example, consider the following statements:

*In Release 6.09 and in earlier releases, it is necessary to specify the `READLIMITS` option.

```
proc shewhart history=summary;
  uchart (flaws ndefects)*lot;
run;
```

The data set SUMMARY must include the variables LOT, FLAWSU, FLAWSN, NDEFCTSU, and NDEFCTSN.

Note that if you specify a *process* name that contains eight characters, the names of the summary variables must be formed from the first four characters and the last three characters of the *process* name, suffixed with the appropriate character.

Other variables that can be read from a HISTORY= data set include

- `_PHASE_` (if the READPHASES= option is specified)
- *block-variables*
- *symbol-variable*
- BY variables
- ID variables

By default, the SHEWHART procedure reads all the observations in a HISTORY= data set. However, if the data set includes the variable `_PHASE_`, you can read selected groups of observations (referred to as *phases*) with the READPHASES= option (see “Displaying Stratification in Phases” on page 1689 for an example).

For an example of a HISTORY= data set, see “Creating u Charts from Nonconformities per Unit” on page 1425.

TABLE= Data Set

You can read summary statistics and control limits from a TABLE= data set specified in the PROC SHEWHART statement. This enables you to reuse an OUTTABLE= data set created in a previous run of the SHEWHART procedure or to create your own TABLE= data set. Because the SHEWHART procedure simply displays the information read from a TABLE= data set, you can use TABLE= data sets to create specialized control charts. Examples are provided in Chapter 49, “Specialized Control Charts.”

The following table lists the variables required in a TABLE= data set used with the UCHART statement:

Table 41.23. Variables Required in a TABLE= Data Set

Variable	Description
<code>_LCLU_</code>	lower control limit for nonconformities per unit
<code>_LIMITN_</code>	nominal sample size associated with the control limits
<i>subgroup-variable</i>	values of the <i>subgroup-variable</i>
<code>_SUBN_</code>	subgroup sample size
<code>_SUBU_</code>	subgroup number of nonconformities per unit
<code>_U_</code>	average number of nonconformities per unit
<code>_UCLU_</code>	upper control limit for nonconformities per unit

Other variables that can be read from a TABLE= data set include

- *block-variables*
- *symbol-variable*
- BY variables
- ID variables
- `_PHASE_` (if the READPHASES= option is specified). This variable must be a character variable of length 16.
- `_TESTS_` (if the TESTS= option is specified). This variable is used to flag tests for special causes and must be a character variable of length 8.
- `_VAR_`. This variable is required if more than one *process* is specified or if the data set contains information for more than one *process*. This variable must be a character variable of length 8.

For an example of a TABLE= data set, see “Saving Control Limits” on page 1422.

Axis Labels

You can specify axis labels by assigning labels to particular variables in the input data set, as summarized in the following table:

Axis	Input Data Set	Variable
Horizontal	all	<i>subgroup-variable</i>
Vertical	DATA=	<i>process</i>
Vertical	HISTORY=	subgroup defects per unit variable
Vertical	TABLE=	<code>_SUBU_</code>

For an example, see “Labeling Axes” on page 1719.

Missing Values

An observation read from a DATA=, HISTORY=, or TABLE= data set is not analyzed if the value of the subgroup variable is missing. For a particular process variable, an observation read from a DATA= data set is not analyzed if the value of the process variable is missing. Missing values of process variables generally lead to unequal subgroup sample sizes. For a particular process variable, an observation read from a HISTORY= or TABLE= data set is not analyzed if the values of any of the corresponding summary variables are missing.

Examples

This section provides advanced examples of the UCHART statement.

Example 41.1. Applying Tests for Special Causes

This example illustrates how you can apply tests for special causes to make u charts more sensitive to special causes of variation.

See SHWUEX1
in the SAS/QC
Sample Library

A textile company inspects rolls of fabric for defects. The rolls are one meter wide and 30 meters long. The following statements create a SAS data set named FABRIC3, which contains the number of fabric defects for 20 rolls of fabric:

```
data fabric3;
  input roll defects @@;
  datalines;
  1 6 2 9 3 14 4 17
  5 3 6 8 7 9 8 2
  9 14 10 1 11 3 12 5
  13 6 14 9 15 10 16 12
  17 11 18 4 19 9 20 4
  ;
```

The following statements create a u chart and tabulate the information on the chart. The chart and tables are shown in Output 41.1.1 and Output 41.1.2.

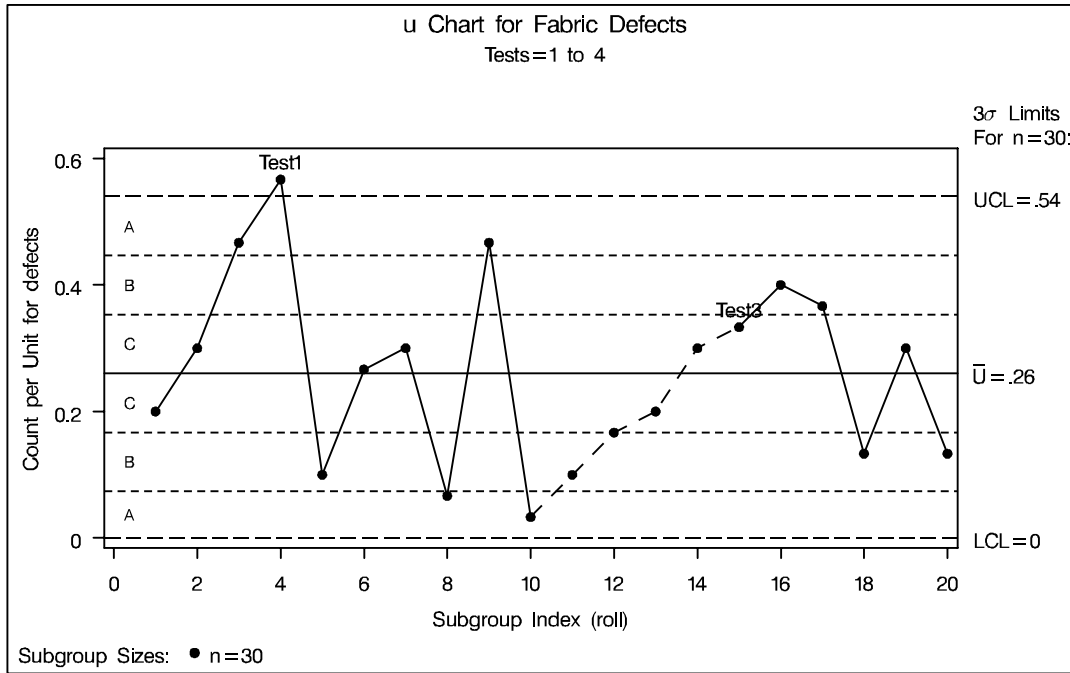
```
symbol v=dot;
title1 'u Chart for Fabric Defects';
title2 'Tests=1 to 4';
proc shewhart data=fabric3;
  uchart defects*roll / subgroupn=30
                    tests    =1 to 4
                    ltests   =20
                    zonelabels
                    tabletests;
run;
```

The TESTS= option requests Tests 1, 2, 3, and 4, which are described in Chapter 48, “Tests for Special Causes.” Only Tests 1, 2, 3, and 4 are recommended for u charts. The ZONELABELS option requests the zone lines, which are used to define the tests, and displays labels for the zones. The LTESTS= option specifies the line type used to connect the points in a pattern for a test that is signaled. The TABLETESTS option requests a table of the values of u_i and the control limits, together with a column indicating the subgroups at which the tests are positive.

Output 41.1.1 and Output 41.1.2 indicate that Test 1 is positive for Roll 4 and Test 3 is positive at Roll 15.

Part 9. The CAPABILITY Procedure

Output 41.1.1. Tests for Special Causes Displayed on *u* Chart



Output 41.1.2. Tabular Form of *u* Chart

u Chart for Fabric Defects
Tests=1 to 4

u Chart Summary for defects

-3 Sigma Limits with n=30 for Count per Unit-

roll	Subgroup Sample Size	Lower Limit	Subgroup Count per Unit	Upper Limit	Special Tests Signaled
1	30.0000	0	0.20000000	0.53928480	
2	30.0000	0	0.30000000	0.53928480	
3	30.0000	0	0.46666667	0.53928480	
4	30.0000	0	0.56666667	0.53928480	1
5	30.0000	0	0.10000000	0.53928480	
6	30.0000	0	0.26666667	0.53928480	
7	30.0000	0	0.30000000	0.53928480	
8	30.0000	0	0.06666667	0.53928480	
9	30.0000	0	0.46666667	0.53928480	
10	30.0000	0	0.03333333	0.53928480	
11	30.0000	0	0.10000000	0.53928480	
12	30.0000	0	0.16666667	0.53928480	
13	30.0000	0	0.20000000	0.53928480	
14	30.0000	0	0.30000000	0.53928480	
15	30.0000	0	0.33333333	0.53928480	3
16	30.0000	0	0.40000000	0.53928480	
17	30.0000	0	0.36666667	0.53928480	
18	30.0000	0	0.13333333	0.53928480	
19	30.0000	0	0.30000000	0.53928480	
20	30.0000	0	0.13333333	0.53928480	

Example 41.2. Specifying a Known Expected Number of Nonconformities

This example illustrates how you can create a u chart based on a known (standard) value u_0 for the expected number of nonconformities per unit.

See SHWUEX2
in the SAS/QC
Sample Library

A u chart is used to monitor the number of defects per square meter of fabric. The defect counts are provided as values of the variable DEFECTS in the data set FABRIC (see page 1420). Based on previous testing, it is known that $u_0 = 0.325$. The following statements create a u chart with control limits derived from this value:

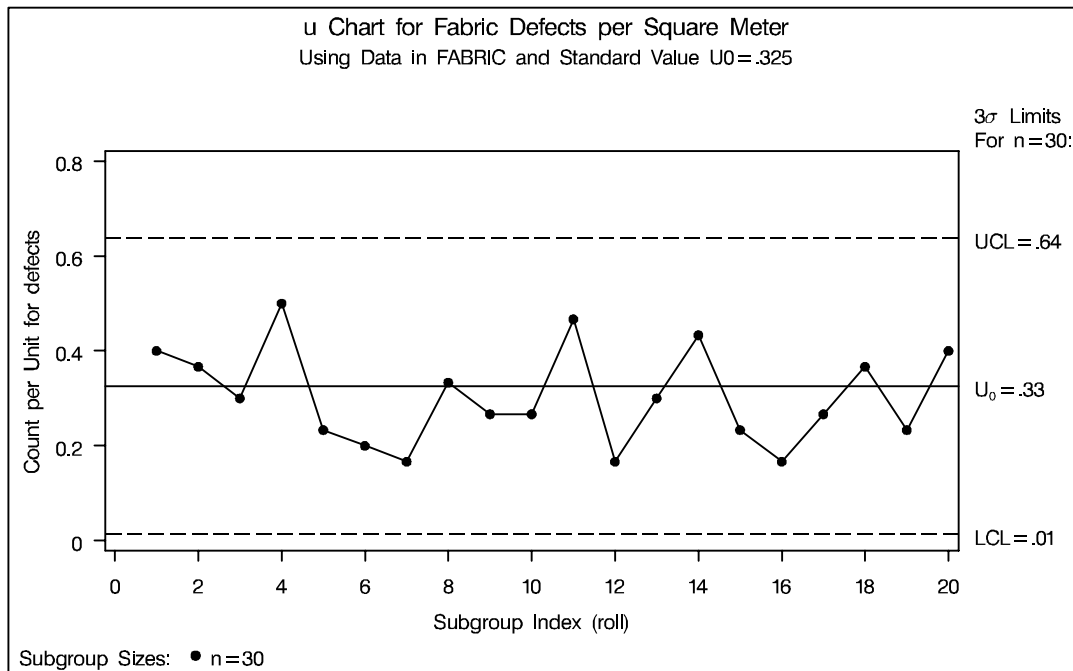
```

title 'u Chart for Fabric Defects per Square Meter';
title2 'Using Data in FABRIC and Standard Value U0=.325';
symbol v=dot;
proc shewhart data=fabric;
    uchart defects*roll / subgroupn=30
                        u0      =0.325
                        usymbol =u0;
run;

```

The chart is shown in Output 41.2.1. The U0= option specifies u_0 , and the USYMBOL= option requests a label for the central line indicating that the line represents a standard value.

Output 41.2.1. A u Chart with Standard Value u_0



Since all the points lie within the control limits, the process is in statistical control.

Alternatively, you can specify u_0 as the value of the variable `_U_` in a LIMITS= data set, as follows:

```

data tlimits;
  length _subgrp_ _var_ _type_ $8;
  _u_     = .325;
  _limitn_ = 30;
  _type_  = 'STANDARD';
  _subgrp_ = 'roll';
  _var_   = 'defects';

proc shewhart data=fabric limits=tlimits;
  uchart defects*roll / subgroupn=30
                        usymbol  =u0;

run;

```

The chart produced by these statements is identical to the one in Output 41.2.1. For further details, see “LIMITS= Data Set” on page 1446.

Example 41.3. Creating u Charts for Varying Numbers of Units

See SHWUEX3
in the SAS/QC
Sample Library

In the fabric manufacturing process described in “Creating u Charts from Defect Count Data” on page 1420, each roll of fabric is 30 meters long, and an inspection unit is defined as one square meter. Thus, there are 30 inspection units in each subgroup sample. Suppose now that the length of each piece of fabric varies. The following statements create a SAS data set (FABRICS2) that contains the number of fabric defects and size (in square meters) of 25 pieces of fabric:

```

data fabrics2;
  input roll defects sqmeters @@;
datalines;
  1  7 30.0   2 11 27.6   3 15 30.4   4  6 34.8   5 11 26.0
  6 15 28.6   7  5 28.0   8 10 30.2   9  8 28.2  10  3 31.4
 11  3 30.3  12 14 27.8  13  3 27.0  14  9 30.0  15  7 32.1
 16  6 34.8  17  7 26.5  18  5 30.0  19 14 31.3  20 13 31.6
 21 11 29.4  22  6 28.6  23  6 27.5  24  9 32.6  25 11 31.7
;

```

A partial listing of FABRICS2 is shown in Output 41.3.1.

Output 41.3.1. The Data Set FABRICS2

Number of Fabric Defects		
roll	defects	sqmeters
1	7	30.0
2	11	27.6
3	15	30.4
.	.	.
.	.	.
.	.	.

The variable ROLL contains the roll number, the variable DEFECTS contains the number of defects in each piece of fabric, and the variable SQMETERS contains the size of each piece.

The following statements request a u chart for the number of defects per square meter:

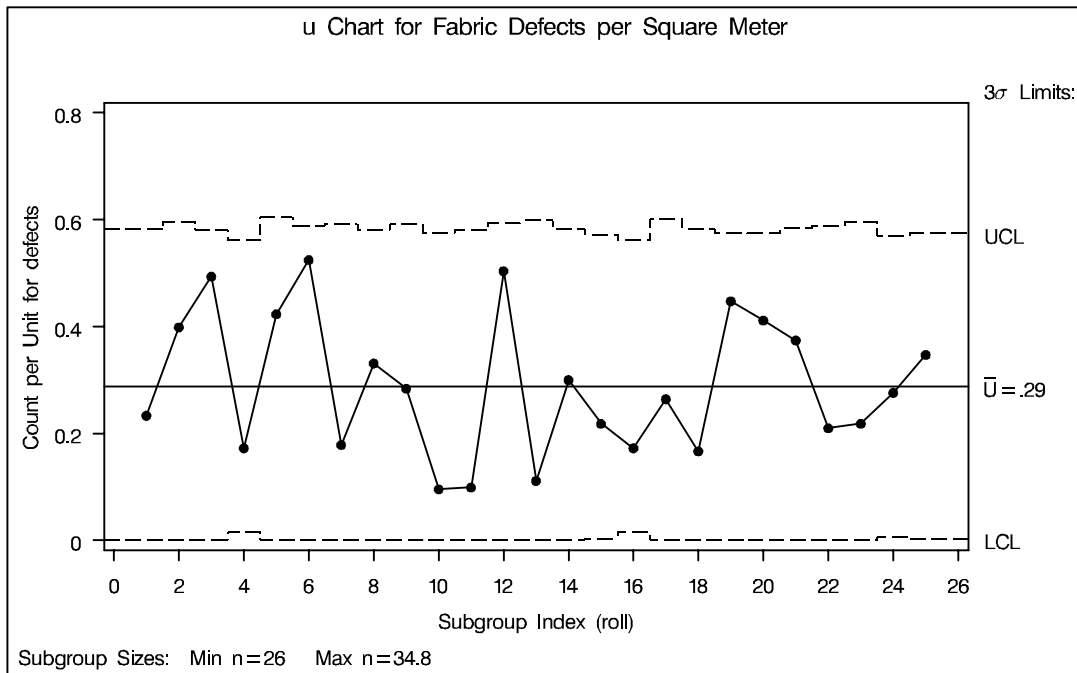
```

title 'u Chart for Fabric Defects per Square Meter';
symbol v=dot;
proc shewhart data=fabrics2;
    uchart defects*roll / subgroupn=sqmeters
        outlimits=flimits;
run;

```

The u chart is shown in Output 41.3.2, and the data set FLIMITS is listed in Output 41.3.3.

Output 41.3.2. A u Chart with Varying Number of Units per Subgroup



Note that the control limits vary with the number of units per subgroup (subgroup sample size). The legend in the lower left corner indicates the minimum and maximum subgroup sample sizes.

Output 41.3.3. The Control Limits Data Set FLIMITS

Control Limits for Fabric Defects								
VAR	_SUBGRP_	_TYPE_	_LIMITN_	_ALPHA_	_SIGMAS_	_LCLU_	_U_	_UCLU_
defects	roll	ESTIMATE	V	V	3	V	0.28805	V

Output 41.3.3 shows that the variables `_LIMITN_`, `_ALPHA_`, `_LCLU_`, and `_UCLU_` have the special missing value `V`, indicating that these variables vary with the sample size.

The following statements request a u chart with a fixed sample size of 30.0 for the control limits. In other words, the control limits are computed as if each piece of fabric were 30 meters long.

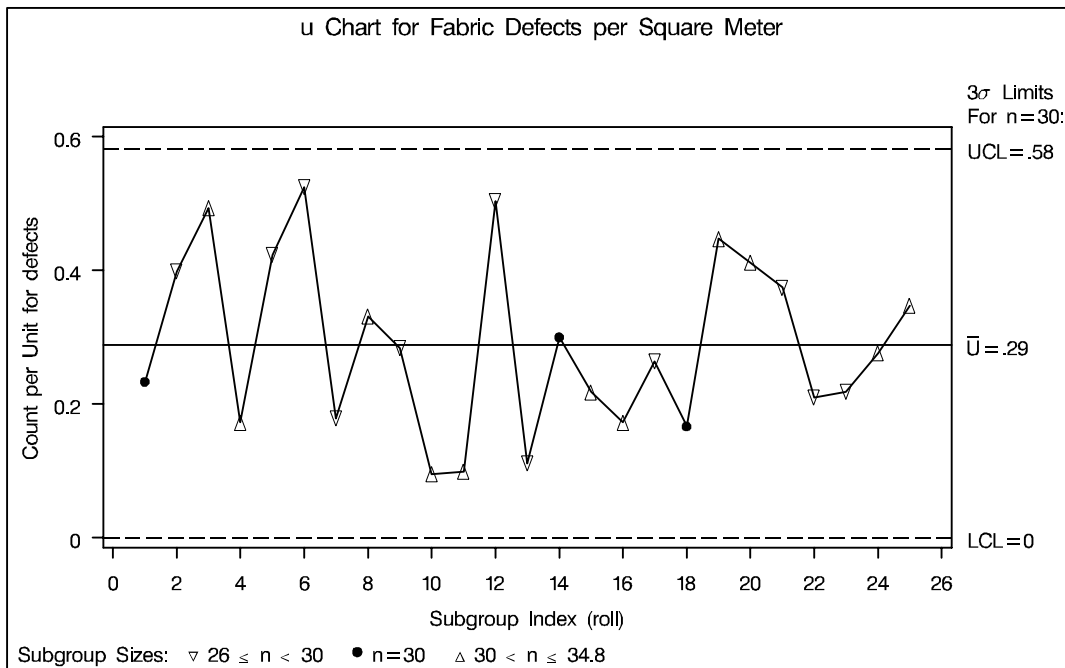
Part 9. The CAPABILITY Procedure

```
proc shewhart data=fabrics2;
  uchart defects*roll / subgroupn=sqmeters
                      outlimits=flimits2
                      limitn   =30
                      alln
                      nmarkers;

run;
```

The ALLN option specifies that points are to be displayed for all subgroups, regardless of their sample size. By default, when you specify the LIMITN= option, only points for subgroups whose sample size matches the LIMITN= value are displayed. The NMARKERS option requests special symbols that identify points for which the subgroup sample size differs from the nominal sample size of 30. The chart is shown in Output 41.3.4.

Output 41.3.4. Control Limits Based on Fixed Subgroup Sample Size



In Output 41.3.4, no points lie outside the control limits, indicating that the process is in control. However, you should be careful when interpreting charts that use a nominal sample size, since the fixed control limits based on this value are only an approximation. Output 41.3.5 lists the data set FLIMITS2, which contains the fixed control limits displayed in Output 41.3.4.

Output 41.3.5. The Fixed Control Limits Data Set FLIMITS2

Fixed Control Limits for Fabric Defects								
VAR	_SUBGROUP_	_TYPE_	_LIMITN_	_ALPHA_	_SIGMAS_	_LCLU_	_U_	_UCLU_
defects	roll	ESTIMATE	30	.002621618	3	0	0.28805	0.58201

The correct bibliographic citation for this manual is as follows: SAS Institute Inc., *SAS/QC[®] User's Guide, Version 8*, Cary, NC: SAS Institute Inc., 1999. 1994 pp.

SAS/QC[®] User's Guide, Version 8

Copyright © 1999 SAS Institute Inc., Cary, NC, USA.

ISBN 1-58025-493-4

All rights reserved. Printed in the United States of America. No part of this publication may be reproduced, stored in a retrieval system, or transmitted, by any form or by any means, electronic, mechanical, photocopying, or otherwise, without the prior written permission of the publisher, SAS Institute Inc.

U.S. Government Restricted Rights Notice. Use, duplication, or disclosure of the software by the government is subject to restrictions as set forth in FAR 52.227-19 Commercial Computer Software-Restricted Rights (June 1987).

SAS Institute Inc., SAS Campus Drive, Cary, North Carolina 27513.

1st printing, October 1999

SAS[®] and all other SAS Institute Inc. product or service names are registered trademarks or trademarks of SAS Institute in the USA and other countries. [®] indicates USA registration.

IBM[®], ACF/VTAM[®], AIX[®], APPN[®], MVS/ESA[®], OS/2[®], OS/390[®], VM/ESA[®], and VTAM[®] are registered trademarks or trademarks of International Business Machines Corporation. [®] indicates USA registration.

Other brand and product names are registered trademarks or trademarks of their respective companies.

The Institute is a private company devoted to the support and further development of its software and related services.