P Charts for Improved Analysis

Vin Kane
Principal Quality / Reliability Engineer, Tellabs
P Charts for Reporting and Improved Analysis

- **Purpose:** Provide support engineers with data analysis and data exploratory tools for -
  - Improved data analysis
  - Reduced reporting time
  - Improved product dispositions
Provides improved data analysis capability

- Simple, fast, improved summaries
- Little or no training
- High Return on Investment

```
// Re do for general weekly reporting
// Removed Part Number clause for P charts
// Rev 4 Added prompt for Product Line Data Table
caption( "Wild Card Is %" );
r=Dialog(
   "Enter Part Number",
   Part = Edit text( "%" ),
   Button( "OK" ),
   Button( "Cancel" )
);
myvalue =r["part"];
show(myvalue);
caption(remove);
```
Database Connections for Tests Results and Predictions

```cpp
dt = open database(
"DSN=Naperville Mercury;APP=JMP;WSID=USNVE1VKANEL;DATABASE=Metrics;Trusted_Connection=Yes",
"select * from datasources..Staging where [Part Number] like "'" || myvalue || "'",
"Staging");

datatable("Staging") << run formulas;
```

```
// Gets prediction table
dt = Open Database(
"DSN=Naperville Mercury;APP=JMP;WSID=USNVE1VKANEL;DATABASE=Mercury;Trusted_Connection=Yes'
"SELECT * FROM dbo.PUBL_Part",
"Predictions"
);
```

```
<table>
<thead>
<tr>
<th>localID</th>
<th>Part Number</th>
<th>Serial Number</th>
<th>module Revision</th>
<th>Fail</th>
<th>Pass</th>
<th>Tested By: Initials</th>
<th>Date Logged</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1</td>
<td>UZ2296391</td>
<td>M</td>
<td>0</td>
<td>1</td>
<td>Stage Team</td>
<td>01/31/2007 12:00</td>
</tr>
<tr>
<td>2</td>
<td>2</td>
<td>UZ2296413</td>
<td>M</td>
<td>0</td>
<td>1</td>
<td>Stage Team</td>
<td>01/31/2007 12:00</td>
</tr>
<tr>
<td>3</td>
<td>3</td>
<td>UZ2296353</td>
<td>M</td>
<td>0</td>
<td>1</td>
<td>Stage Team</td>
<td>01/31/2007 12:00</td>
</tr>
<tr>
<td>4</td>
<td>4</td>
<td>UZ2296326</td>
<td>M</td>
<td>0</td>
<td>1</td>
<td>Stage Team</td>
<td>01/31/2007 12:00</td>
</tr>
<tr>
<td>5</td>
<td>5</td>
<td>UZ2295767</td>
<td>M</td>
<td>0</td>
<td>1</td>
<td>Stage Team</td>
<td>01/31/2007 12:00</td>
</tr>
<tr>
<td>6</td>
<td>6</td>
<td>UZ2295485</td>
<td>M</td>
<td>0</td>
<td>1</td>
<td>Stage Team</td>
<td>01/31/2007 12:00</td>
</tr>
<tr>
<td>7</td>
<td>7</td>
<td>UZ2296488</td>
<td>M</td>
<td>0</td>
<td>1</td>
<td>Stage Team</td>
<td>01/31/2007 12:00</td>
</tr>
</tbody>
</table>
```

```
<table>
<thead>
<tr>
<th>Part Number</th>
<th>Part Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>177</td>
<td>ADSL 6+6 E-ATM G.LITE</td>
<td>ADSL 6+6 G.LITE E-ATM 600 OHM</td>
</tr>
<tr>
<td>178</td>
<td>S.HDSL</td>
<td>G.S.HDSL ATM UNI 6-PORT CHANNEL</td>
</tr>
<tr>
<td>179</td>
<td>ADSL 6+0 ANNEX C - 6 PORTS</td>
<td>ADSL 6+0 ANNEX C - 6 PORTS</td>
</tr>
<tr>
<td>180</td>
<td>L-PAY</td>
<td>LET PAYPHONE CHANNEL UNIT</td>
</tr>
<tr>
<td>181</td>
<td>R-PAY</td>
<td>RST PAYPHONE CHANNEL UNIT</td>
</tr>
<tr>
<td>182</td>
<td>ADSL 4+6 Centillium 600Ohm</td>
<td>ADSL 4+6 E-ATM 600 OHM</td>
</tr>
</tbody>
</table>
```
// Fix Part Number here
datatable("Staging Sort") << new column("Part Number Fix",
    formula(UpperCase( :Part Number )))
datatable("Staging Sort") << run formulas;

// Correct pass fail
datatable("Staging sort") << run formulas;
datatable("Staging Sort") << new column("Stage Pass Fail", character
    formula(If( :Fail > 0,
        "Fail",
        "Pass"
    )));

// Add Pass Fail Column to correct Pass Fail Counts
datatable("Staging sort") << run formulas;
datatable("Staging Sort") << new column("Pass Fix",
    formula(If( :Stage Pass Fail == "Pass",
        1,
        0
    )))

Converting all part number characters to upper case

Standardizes pass/fail counts
### Improving Data Quality

```sql
// Date section

datatable("Staging Sort") << new column("Month", formula(Month(:Date Logged)));
datatable("Staging Sort") << new column("Year", formula(Year(:Date Logged)));

DataTable("Staging Sort") << new column("Date M/Y", format("M/Y"), formula(:Date Logged));
```

<table>
<thead>
<tr>
<th>Pass Fix</th>
<th>Fail Fix</th>
<th>Serial Number Fix</th>
<th>Part Number Fix</th>
<th>Month</th>
<th>Year</th>
<th>Date M/Y</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0</td>
<td>UN0030006</td>
<td></td>
<td>3</td>
<td>2006</td>
<td>03/2006</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>UN0030036</td>
<td></td>
<td>3</td>
<td>2006</td>
<td>03/2006</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>UN0030037</td>
<td></td>
<td>3</td>
<td>2006</td>
<td>03/2006</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>UN0610003</td>
<td></td>
<td>3</td>
<td>2006</td>
<td>03/2006</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>UN0610007</td>
<td></td>
<td>3</td>
<td>2006</td>
<td>03/2006</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>UN0610104</td>
<td></td>
<td>3</td>
<td>2006</td>
<td>03/2006</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>UV0461194</td>
<td></td>
<td>3</td>
<td>2006</td>
<td>03/2006</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>UV0542477</td>
<td></td>
<td>3</td>
<td>2006</td>
<td>03/2006</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>UV0542480</td>
<td></td>
<td>3</td>
<td>2006</td>
<td>03/2006</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>UV0542499</td>
<td></td>
<td>3</td>
<td>2006</td>
<td>03/2006</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>P8P2FKP</td>
<td></td>
<td>131S</td>
<td>2006</td>
<td>03/2006</td>
</tr>
</tbody>
</table>
Manufacturing Identifiers

- Decode the serial number tested and add identifiers for the manufacturer, manufacturing year, week and day

```r
// For decoding the serial number into manufacturing date
dataset("Staging Sort") <- new column("Serial Number Character", character, formula(Uppercase( Substr( :Serial Number, 1, 1 ) )));

dataset("Staging Sort") <- new column("Mfg Date", character, formula(If( :Name("Mfg Year (SN)") == 0,
    Char( :Warranty_Date ) || " - " || Char( :Warranty_Year ),
    Char( :Name("Mfg Year (SN)" ) ) || " - " ||
    Char( :Name("Mfg Month (SN)" ) )
  )
);
dataset("Staging Sort") <- run formulas;

dt <- new column("Year Month", character, formula(Char( :Year ) || " - " || Char( :Month )));
dt <- run formulas;
```

<table>
<thead>
<tr>
<th>Mfg Date</th>
<th>Month</th>
<th>Year</th>
</tr>
</thead>
<tbody>
<tr>
<td>2006-5</td>
<td>6</td>
<td>2006</td>
</tr>
<tr>
<td>2006-5</td>
<td>6</td>
<td>2006</td>
</tr>
<tr>
<td>2006-5</td>
<td>6</td>
<td>2006</td>
</tr>
<tr>
<td>2006-4</td>
<td>6</td>
<td>2006</td>
</tr>
<tr>
<td>2006-4</td>
<td>6</td>
<td>2006</td>
</tr>
<tr>
<td>2006-5</td>
<td>6</td>
<td>2006</td>
</tr>
<tr>
<td>2006-5</td>
<td>6</td>
<td>2006</td>
</tr>
</tbody>
</table>

2008 Indianapolis
Output P Charts
Manufacturing Dates vs. Reporting Dates

- P chart for manufacturing year and week
- P chart for calendar year week report

Highlites ID the manufacturing date code quality issues

2008 Indianapolis
### JMP User Interface to Remote SFDC

**Simple interface minimizes training**

<table>
<thead>
<tr>
<th>SPC Analysis Guad Rev 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
</tr>
<tr>
<td>2</td>
</tr>
<tr>
<td>3</td>
</tr>
<tr>
<td>4</td>
</tr>
<tr>
<td>5</td>
</tr>
<tr>
<td>6</td>
</tr>
<tr>
<td>7</td>
</tr>
<tr>
<td>8</td>
</tr>
<tr>
<td>9</td>
</tr>
<tr>
<td>10</td>
</tr>
<tr>
<td>11</td>
</tr>
<tr>
<td>12</td>
</tr>
<tr>
<td>13</td>
</tr>
<tr>
<td>14</td>
</tr>
<tr>
<td>15</td>
</tr>
<tr>
<td>16</td>
</tr>
</tbody>
</table>

- **Single part number or Wild Card for Product Line**
- **Start Date**
  - 01/01/2007

---

### 2008 Indianapolis
JMP Summary Outputs

Overall failure % by shift by Inspection Test process. Control limits indicate a statistical difference between shifts within Inspection Test Steps.

Line Chart with the % of occurrences the process went beyond the 3 sigma control limits.

Code generated by jmp copied and pasted within overall script.

**Control Chart**

<table>
<thead>
<tr>
<th>P of Fail Total</th>
</tr>
</thead>
<tbody>
<tr>
<td>Proportion for Fail Total</td>
</tr>
<tr>
<td>0.20</td>
</tr>
<tr>
<td>0.15</td>
</tr>
<tr>
<td>0.10</td>
</tr>
<tr>
<td>0.05</td>
</tr>
<tr>
<td>0.00</td>
</tr>
</tbody>
</table>

Audit  Final  Funct Test  ICT  SMT

2008 Indianapolis
Summary Outputs Available

Control Chart

P of Out

Proportion for Out

Year Week

2007 - 6
2007 - 23
2007 - 38
2007 - 51
2007 - 25
2007 - 12
2007 - 31
2007 - 49
2007 - 15
2007 - 25
2007 - 41
2007 - 21
2007 - 9
2008 - 11
2008 - 15
2008 - 21
2007 - 7
2007 - 23
2007 - 49
2008 - 9
2008 - 20

Plots the % of daily occurrences that product exceeded SPC limits with Average-----

Plots the % of daily occurrences average that product exceeded SPC limits

7/10/2008 6:41 AM
Data Table=SPC Summ Split

2008 Indianapolis
Outputs Available, Weekly Summaries

Standard SPC P charts for each Inspection / test Process with points exceeding the control limits identified *

**Control Chart part=81.5517A, Operation=Funct Test**

**P of Fail Total**

![Control Chart Example](chart.png)

Any data point can be removed and the chart recalculated when root cause corrective action is taken.

**P Chart code example**

```plaintext
new window{myvalue||" Daily P charts ",
Data Table( "Daily Yield Totals" ) <<Control Chart{
Sample Label( :Year Week Day ),
Sample Size( :Total ),
KSigma( 3 ),
Chart Col{
  :Fail Total,
  P( Test 1( 1 ), Test Beyond Limits( 1 ) )
},
by( :part,:Operation)
});
```

**Control Chart part=81.5517A, Operation=ICT**

![Control Chart Example](chart2.png)

2008 Indianapolis
Outputs Available, Daily Summaries

When needed, P Chats can be employed at the daily level

Standard SPC P charts at daily level for each Inspection / test Process with points exceeding the control limits identified

Data points can be removed and the chart recalculated when root cause corrective action is taken or process change takes place

2008 Indianapolis
Non Traditional Summaries Example  Day of the Week Analysis

SPC P charts for each Inspection / test Process by test day of the week with code

Control Chart part=81.5517A, Operation=Funct Test

Control Chart part=81.5517A, Operation=ICT

Points outside the control limits indicate a possible assignable cause

new window(myvalue||" Daily P charts ",
Data Table( "Daily Yield Totals" ) <<Control Chart(  
    Sample Label( :Year Week Day ),
    Sample Size( :Total ),
    KSigma( 3 ),
    Chart Col(
        :Fail Total,
        P( Test 1( 1 ), Test Beyond Limits( 1 ) )
    ),
    by( :part, :Operation)
));
Control Chart

P of Fail

SPC Chart for failure rate by the time of day

Points outside the control limits indicate a possible assignable cause is present

datatable("TOD")<< run formulas;
new window((myvalue),
datatable("TOD")<< Control Chart(
    Sample Label( :TOD ),
    Sample Size( :Total ),
    KSigma( 3 ),
    Chart Col( :Fail, P( Test 1( 1 ), Test Beyond Limits( 1 ) ) )
));
User dialog box and database connection scripts mean little to no training required for the quality support engineers.

Provides a confirmation of the subcontractors' process control.

Improved data analysis capability by utilizing various levels of data summaries with P charts to signal the presence of assignable causes, by both manufacturing and reporting calendar.

- By:
  - Product
  - Shift
  - Day
  - Week
  - Month