

**PANDUAN PENGGUNAAN GRAPHICAL USER INTERFACE CONTROL
CHART DENGAN MATLAB**

Disusun Untuk Memenuhi Tugas Tengah Semester

Dosen Pengampu: Dr. Budi Warsito, S.Si., M.Si.



MODUL

Disusun Oleh:

BEATRICE MARIETTA

24050119130078

DEPARTEMEN STATISTIKA

FAKULTAS SAINS DAN MATEMATIKA

UNIVERSITAS DIPONEGORO

SEMARANG

2021

KATA PENGANTAR

Segala puji syukur kami panjatkan kehadirat Tuhan Yang Maha Esa yang telah melimpahkan berkat dan karunia-Nya, sehingga penulis mampu menyelesaikan modul ini yang berjudul “Panduan Penggunaan Graphical User Interface Control Chart dengan MATLAB” guna memenuhi tugas mata kuliah Komputasi Statistika Lanjut. Modul ini disusun dengan tujuan untuk memberikan panduan atau petunjuk serta langkah-langkah dalam mengoperasikan GUI *Control Chart* yang dibuat penulis.

Penulis menyadari bahwa dalam penulisan modul ini masih jauh dari sempurna serta masih terdapat kesalahan dan kekurangan karena terbatasnya pengalaman dan pengetahuan yang penulis miliki. Oleh karena itu, penulis mengharapkan saran dan masukan bahkan kritik dari berbagai pihak yang. Kemudian, apabila terdapat banyak kesalahan pada laporan penelitian ini penulis mohon maaf sebesar-besarnya.

Penulis menyampaikan terima kasih kepada Bapak Dr. Budi Warsito, S.Si., M.Si. selaku pengampu mata kuliah Komputasi Statistika Lanjut yang telah membimbing dan membantu penulis dalam memberikan materi kuliah serta bantuan dari banyak pihak, sehingga penulis dapat menyelesaikan modul ini. Harapan penulis semoga laporan penelitian ini dapat bermanfaat bagi pembaca.

Semarang, 9 Oktober 2021

Penulis

DAFTAR ISI

HALAMAN SAMPUL.....	i
KATA PENGANTAR.....	ii
DAFTAR ISI.....	iii
BAB I.....	1
1.1 Tujuan.....	1
1.2 Manfaat	1
BAB II.....	2
2.1 GUI (Graphical User Interface).....	2
2.2 Peta Kendali (<i>Control Chart</i>)	2
BAB III.....	4
3.1 Tampilan GUI	4
3.2 Panduan Penggunaan GUI	6
3.2.1 Judul Control Chart.....	6
3.2.2 Panel Types of Data & Control Chart	6
3.2.3 Panel Input	12
3.2.4 Pushbutton OK	14
3.2.5 Pushbutton Reset	14
3.2.6 Panel Output	15
3.2.7 Pushbutton Exit	16
3.3 Langkah-Langkah	16
3.4 Hasil Perhitungan.....	17
3.3.1 Data Variabel.....	17
3.3.2 Data Atribut.....	18
DAFTAR PUSTAKA	21
LAMPIRAN.....	22

BAB I

PENDAHULUAN

1.1 Tujuan

Tujuan pembuatan GUI *control chart* dengan MATLAB adalah menerapkan UI dalam perhitungan dan pembuatan peta kendali atau *control chart*.

1.2 Manfaat

Modul “Panduan Penggunaan Graphical User Interface Control Chart pada MATLAB” berisi petunjuk atau panduan penggunaan GUI dalam *control chart* (peta kendali) yang dibuat penulis, sehingga pembaca dapat memahami dan mengerti bagaimana penggunaan GUI tersebut. Tujuan pembuatan GUI *control chart* adalah untuk memudahkan dalam perhitungan yang diperlukan untuk membuat *control chart*. Selain itu, pengolahan data *control chart* dengan GUI akan menghasilkan output grafik atau peta kendali yang dibentuk oleh UCL, CL, dan LCL serta output tabel yang berisi nilai UCL, CL, dan LCL pada tiap sampel.

BAB II

TINJAUAN PUSTAKA

2.1 GUI (Graphical User Interface)

UI (*User Interface*) merupakan sebuah tampilan grafis pada satu atau lebih jendela (*window*) yang berisi kontrol (*control*) yang disebut komponen (*component*), sehingga memungkinkan pengguna untuk melakukan pekerjaan interaktif. Dengan memanipulasi kontrol pada UI, UI akan merespon tindakan secara bergantian. GUI (*Graphical User Interface*) merupakan salah satu *tools* pada MATLAB untuk membuat UI dengan GUIDE (*Graphical User Interface Designer*) atau dengan skrip MATLAB. Tiap kontrol dan UI memiliki satu atau lebih *callback* yang merupakan perintah pada MATLAB untuk melakukan sesuatu, biasanya dipicu dengan menekan tombol atau mengarahkan kursor ke komponen, lalu UI akan merespon kejadian tersebut. *Callback* tidak dapat dikontrol ketika program sedang dijalankan. Terdapat dua cara mengkodekan *callback*, yaitu sebagai fungsi bahasa MATLAB yang disimpan dalam file dan sebagai tulisan yang berisi perintah MATLAB. *Callback* sebagai fungsi yang disimpan dalam file lebih sering digunakan, karena fungsi memiliki akses ke argument serta lebih efektif dan fleksibel. Penggunaan GUIDE akan otomatis membuat M-file yang mengontrol bagaimana GUI beroperasi. M-file akan menginisialisasi GUI dan berisi kerangka kerja (*framework*) untuk semua *callback* GUI.

2.2 Peta Kendali (*Control Chart*)

Control chart pertama kali diperkenalkan oleh Dr. Walter Andrew Shewhart yang bekerja untuk Bell Telephone Laboratories Amerika Serikat. *Control chart* merupakan salah satu alat bantu dalam pengendalian kualitas statistika (*Statistical Quality Control*). *Control chart* merupakan sebuah grafik yang memberi gambaran tentang perilaku sebuah proses. *Control chart* digunakan untuk memeriksa dan mengontrol proses produksi. Proses yang stabil namun berada di luar batas kendali harus diperbaiki untuk menemukan penyebab agar didapatkan hasil perbaikan yang fundamental. Terdapat dua jenis *control chart* berdasarkan karakteristik data yang diamati, yaitu *control chart* untuk data variabel dan *control chart* untuk data atribut. Data variabel merupakan data berupa karakteristik kualitas hasil pengukuran dan dapat dinyatakan dalam satuan ukuran, seperti cm, kg, liter, dan sebagainya. *Control chart* yang termasuk dalam kategori data variabel adalah peta kendali \bar{X} dan R serta peta kendali \bar{X} dan

S. Peta kendali \bar{X} atau rata-rata memvisualisasikan fluktuasi rata-rata dan menunjukkan penyimpangan rata-rata sampel yang akan memberikan gambaran konsistensi proses. Proses cenderung stabil jika rata-rata sampel mendekati nilai rata-rata dan sebaliknya. Peta kendali R atau *range* (rentang) mengukur perbedaan nilai terendah dan tertinggi pada setiap sampel. Peta kendali S atau standar deviasi mengukur nilai simpangan baku pada tiap sampel. Peta kendali R dan S memberi gambaran mengenai variabilitas proses.

Data atribut merupakan data yang bersifat diskrit dan merupakan ukuran kualitas indeks kapabilitas proses dan tidak dapat dinyatakan dalam satuan ukuran. Data atribut umumnya diukur dengan dihitung atau menggunakan daftar pencacahan atau *tally*, seperti jumlah cacat dalam satu *batch* produk, jenis warna, jenis kelamin, dan sebagainya. Terdapat dua tipe teknik pengendalian kualitas pada kategori data atribut, yaitu *yes/no* (ya/tidak) dan terhitung. Tipe data *yes/no* hanya membedakan antara cacat dan tidak cacat. *Control chart* yang termasuk dalam tipe data *yes/no* adalah peta kendali P (mengukur proporsi *defect*/cacat/kegagalan dalam proses) dan peta kendali NP (mengukur jumlah *defect*/cacat/kegagalan dalam proses). Sedangkan, tipe data terhitung jika data yang diobservasi lebih rumit atau perlu analisis mendalam. *Control chart* yang termasuk dalam tipe data terhitung adalah peta kendali C dan peta kendali U yang berfungsi untuk mengukur jumlah *defect*/ketidaksesuaian pada unit yang diproduksi. Perbedaan dalam penggunaan peta kendali NP dengan P dan peta kendali C dengan U adalah peta kendali P dan peta kendali U digunakan jika jumlah sampel (*sample size*) yang diobservasi adalah konstan atau tetap ataupun tidak konstan atau bervariasi, sedangkan peta kendali NP dan peta kendali C hanya dapat digunakan jika jumlah sampel (*sample size*) yang diobservasi adalah konstan atau tetap.

Control chart dibentuk oleh UCL (*Upper Control Limit*) atau batas kendali atas, CL (*Central Line*), dan LCL (*Lower Control Limit*) atau batas kendali bawah. Berikut merupakan perhitungan UCL, CL, dan LCL.

$$CL = \mu$$

$$UCL = \mu + 3 \times \frac{\sigma}{\sqrt{n}}$$

$$LCL = \mu - 3 \times \frac{\sigma}{\sqrt{n}}$$

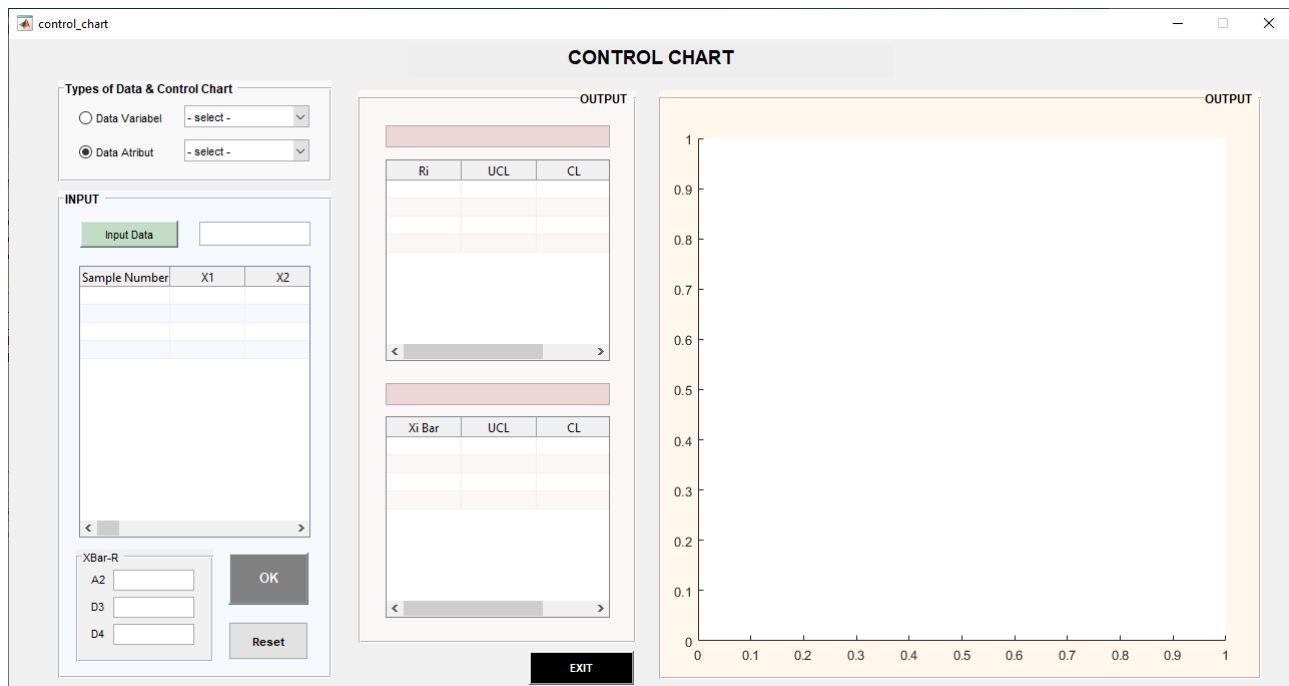
Dengan μ adalah rata-rata atau mean dan σ adalah standard deviation atau simpangan baku serta n adalah ukuran sampel (*sample size*).

BAB III

ANALISIS DAN PEMBAHASAN

3.1 Tampilan GUI

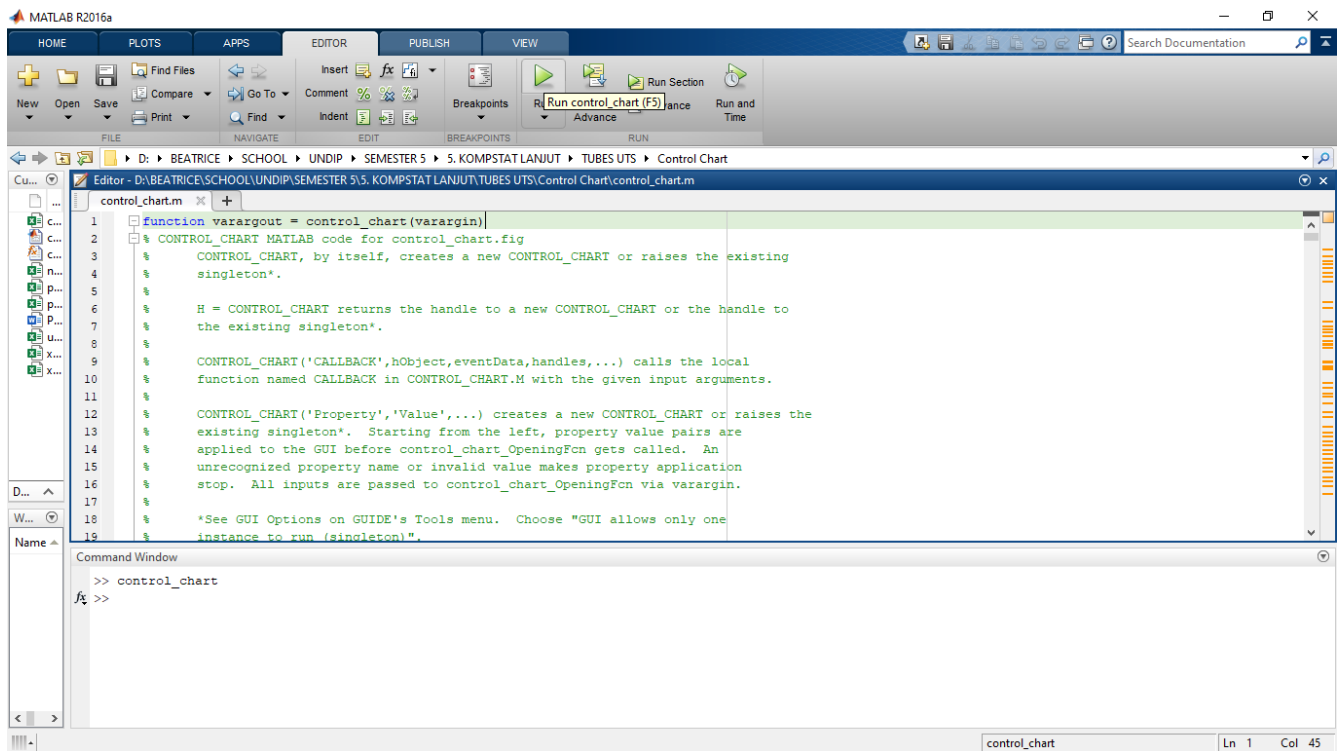
Berikut merupakan tampilan GUI *Control Chart* pada Matlab.



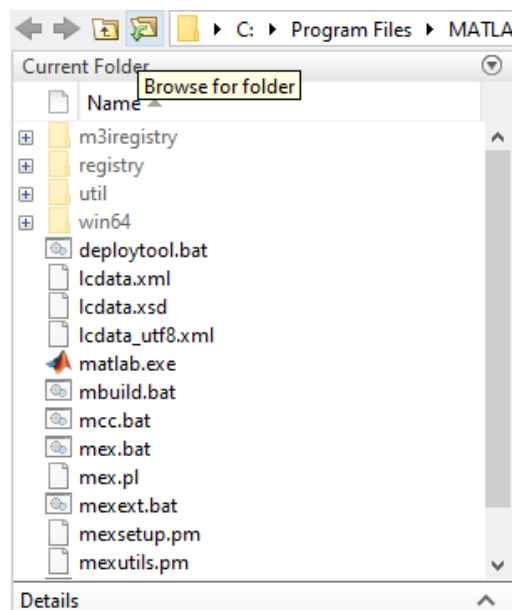
Untuk menampilkan GUI, dapat dengan menulis syntax berupa nama GUI "**control_chart**" yang ingin di run atau dijalankan pada **Command Window**.

The screenshot shows the MATLAB Command Window with the command `control_chart` entered at the prompt `>>`.

Atau dapat dilakukan dengan klik **run** M-file “**control_chart.m**” pada **Editor**.



Sebelum menjalankan GUI, pastikan bahwa folder sudah berada pada tempat penyimpanan file. Untuk mengganti folder, klik Browse for folder and pilih folder dimana file disimpan.



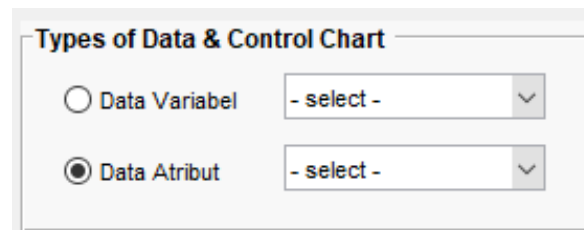
3.2 Panduan Penggunaan GUI

3.2.1 Judul Control Chart

Judul GUI yang dibuat oleh penulis adalah *Control Chart*, dimana GUI yang dibuat digunakan untuk mempermudah dalam melakukan perhitungan dan membuat peta kendali (*control chart*).

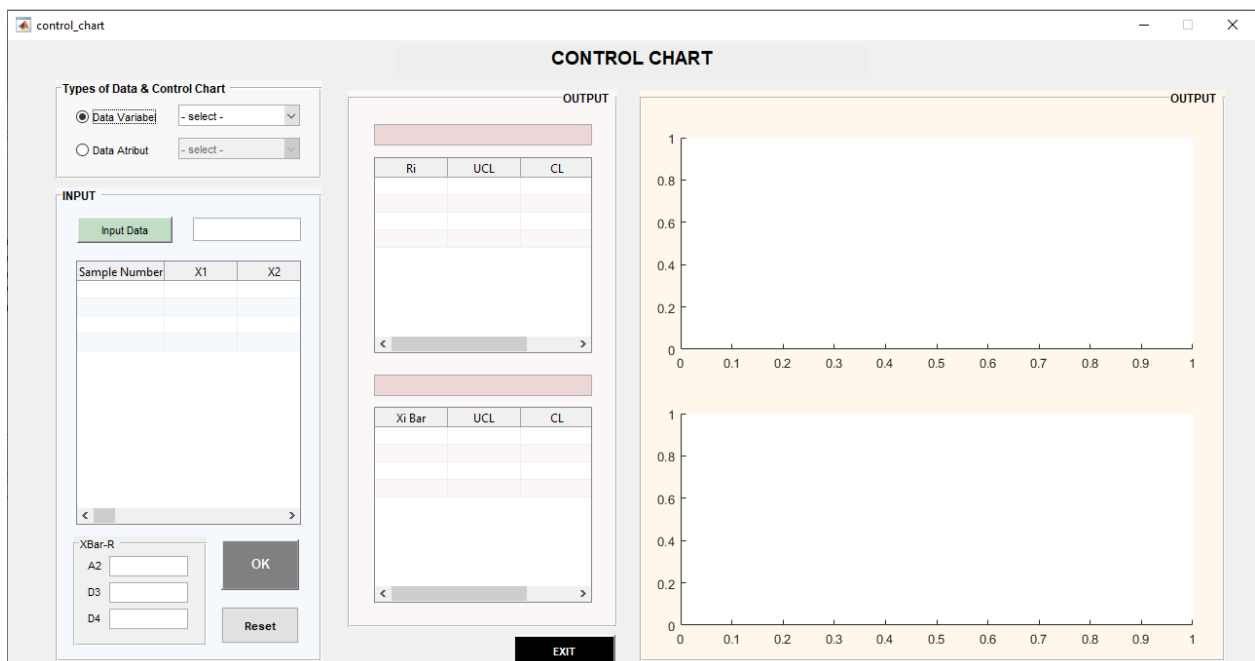
3.2.2 Panel Types of Data & Control Chart

Panel **Types of Data & Control Chart** berisi dua pilihan jenis data yang akan dilakukan perhitungan, yaitu **Data Variabel** dan **Data Atribut**.



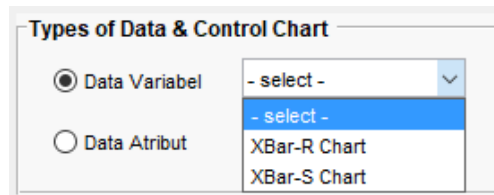
a. Data Variabel

Pemilihan **Data Variabel** akan menghasilkan tampilan seperti di bawah ini, dimana popup-menu untuk **Data Atribut** dinonaktifkan, pada **Output** akan terdapat dua tabel dan dua grafik, serta pada **Input** akan ditampilkan tabel untuk data variabel dan panel untuk memasukkan nilai tabel *control chart*.



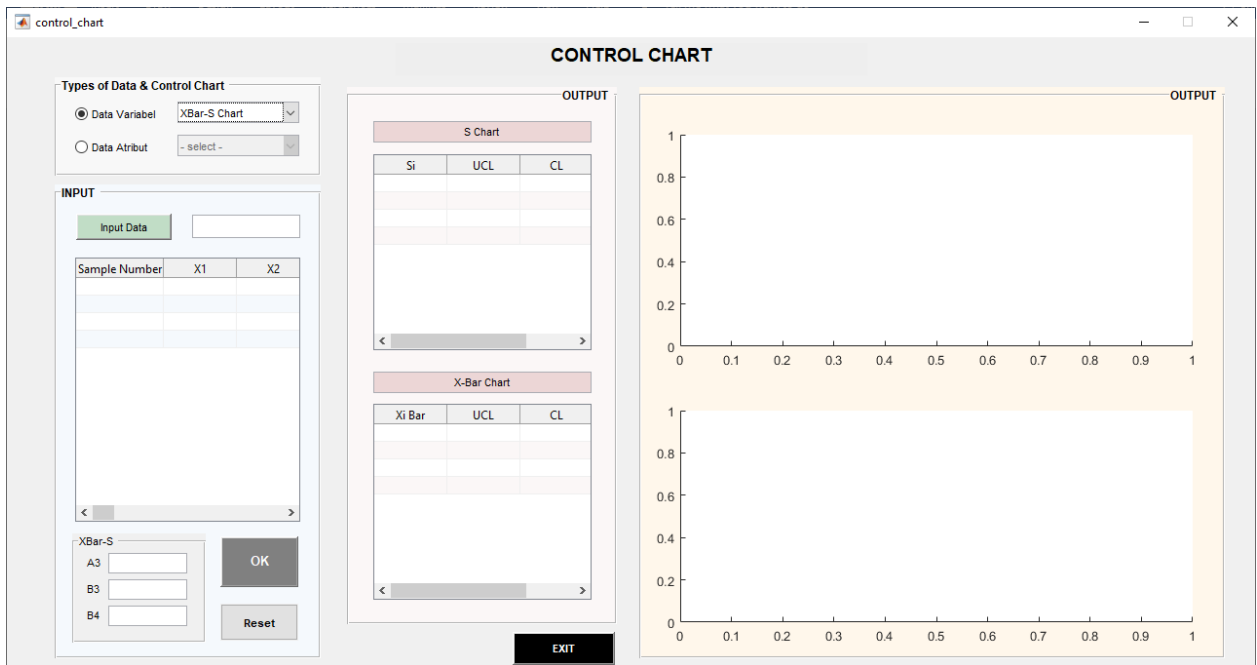
Sebagai cacatan, perhitungan *control chart* pada data variabel hanya dapat digunakan ketika setiap sampel memiliki jumlah pengamatan atau observasi yang sama.

Kemudian, popup-menu digunakan untuk memilih jenis *control chart* pada data variabel, yaitu **XBar-R Chart** dan **XBar-S Chart**.



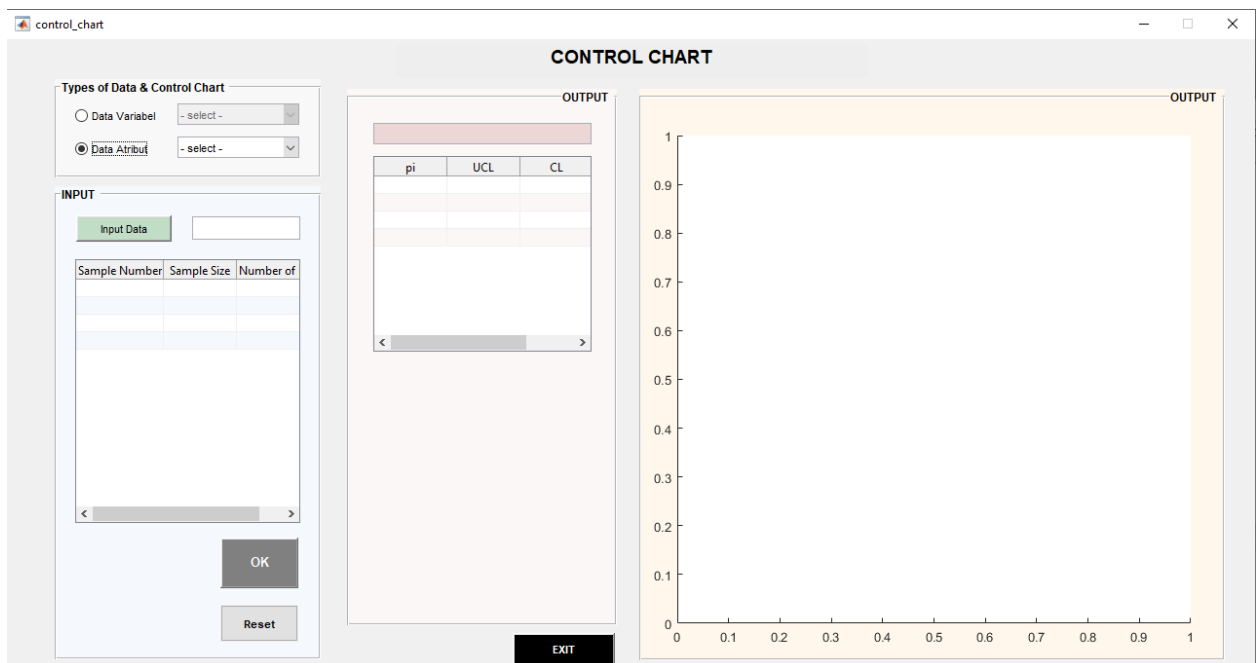
Ketika **XBar-R Chart** dipilih, maka akan menampilkan judul pada kedua tabel **Output**, yaitu **R Chart** dan **XBar Chart** serta pada **Input** akan terdapat panel **XBar-R** untuk memasukkan nilai A2, D3, dan D4 oleh pengguna.

Ketika **XBar-S Chart** dipilih, maka akan menampilkan judul pada kedua tabel **Output**, yaitu **S Chart** dan **XBar Chart** serta pada **Input** akan terdapat panel **XBar-S** untuk memasukkan nilai A3, B3, dan B4 oleh pengguna.



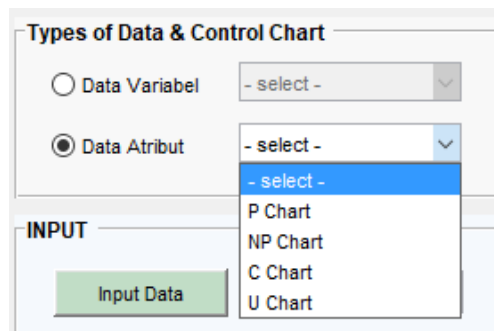
b. Data Atribut

Pemilihan **Data Atribut** akan menghasilkan tampilan seperti di bawah ini, dimana popup-menu untuk **Data Variabel** dinonaktifkan, pada **Output** akan terdapat satu tabel dan satu grafik, serta pada **Input** akan ditampilkan tabel untuk data atribut dan panel untuk memasukkan nilai tabel *control chart* tidak akan ditampilkan.

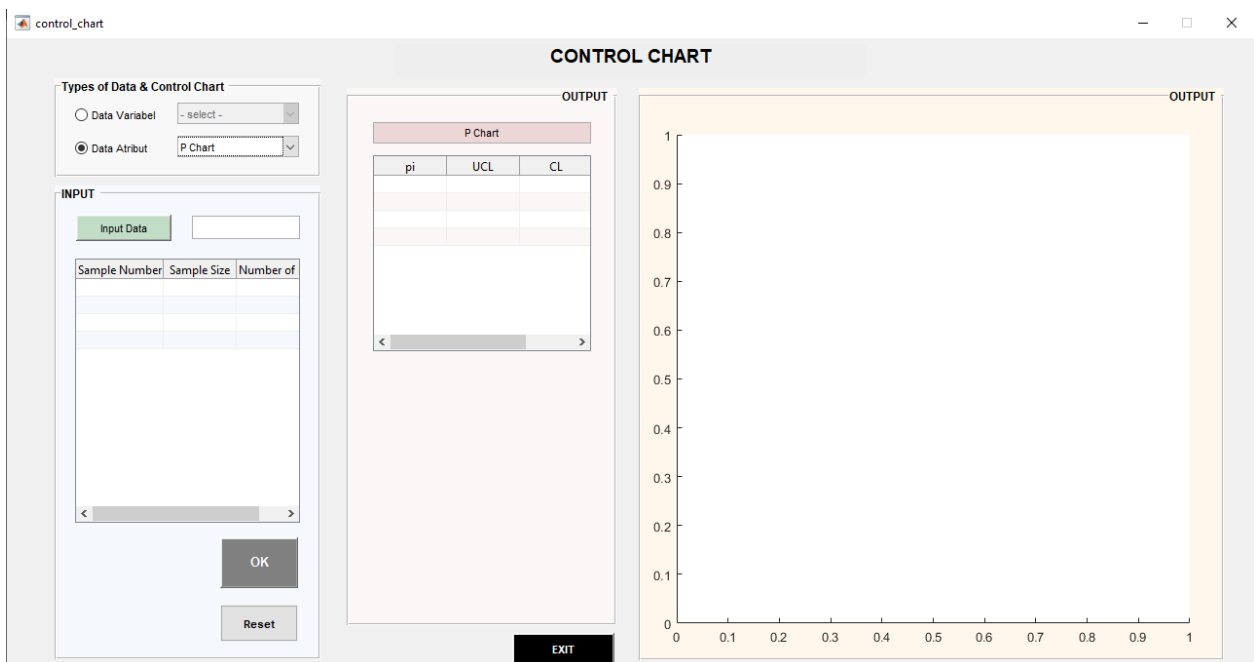


Sebagai catatan, perhitungan *control chart* pada data atribut bisa menggunakan jumlah sampel (*sample size*) yang sama atau berbeda, namun jumlah sampel yang berbeda akan menghasilkan peta kendali dengan batas-batas yang kurang jelas.

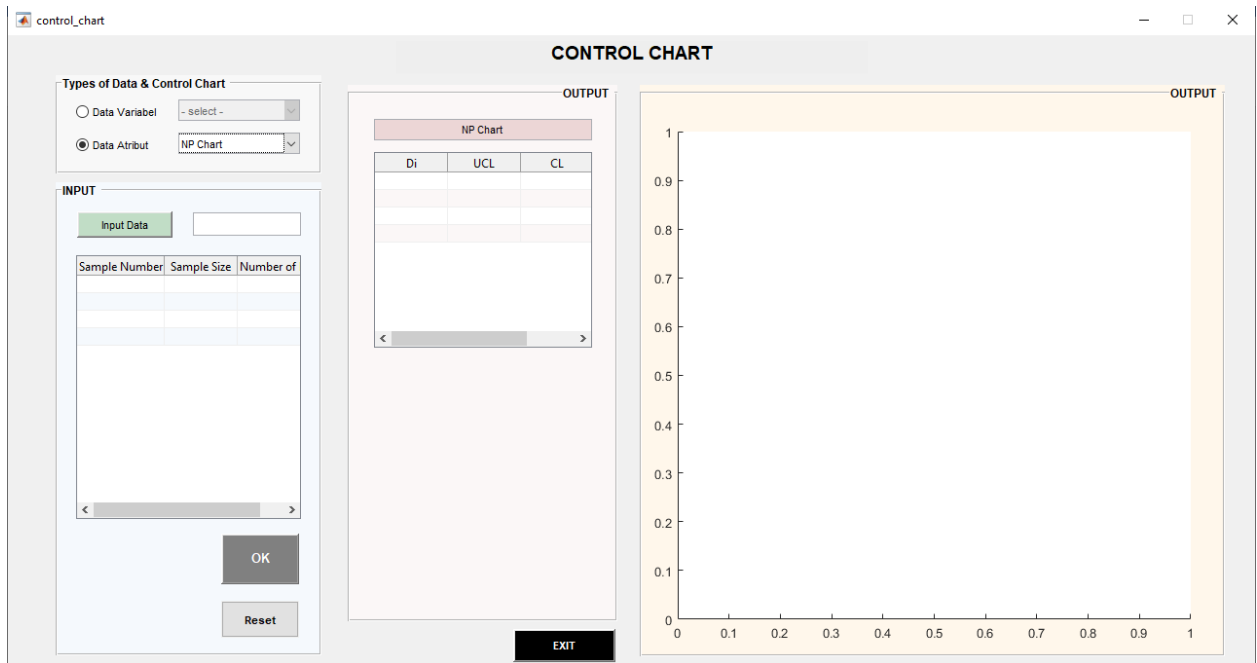
Kemudian, popup-menu digunakan untuk memilih jenis *control chart* pada data atribut, yaitu **P Chart**, **NP Chart**, **C Chart**, dan **U Chart**.



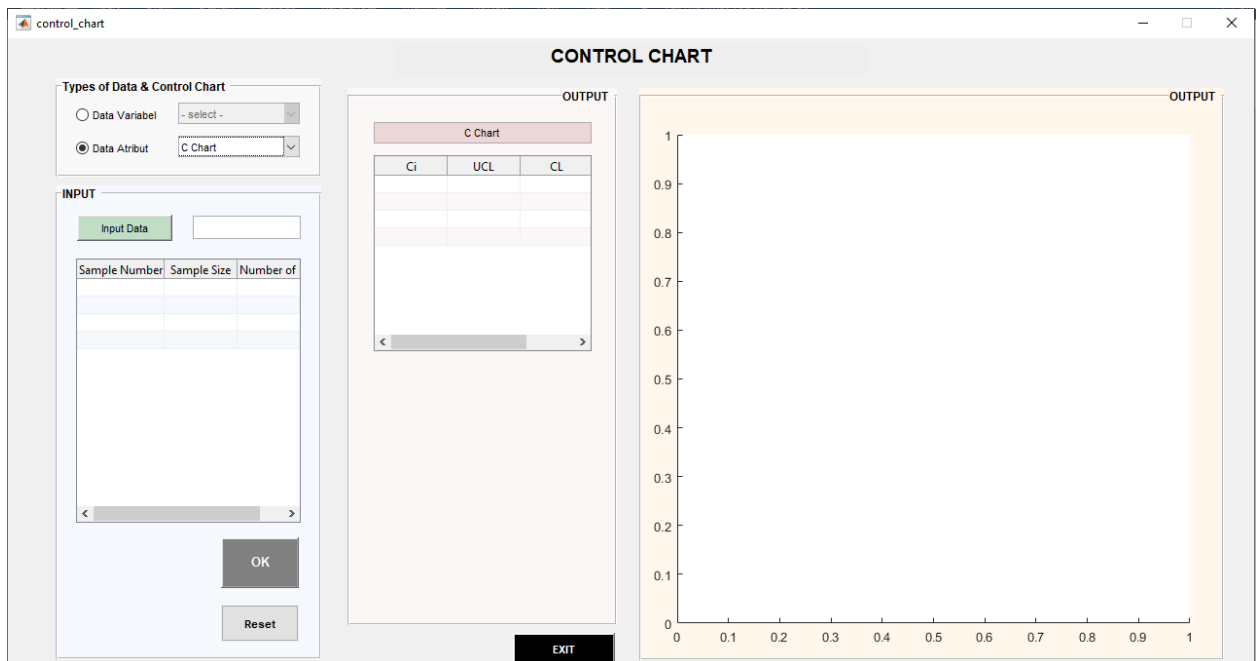
Ketika **P Chart** dipilih, maka akan menampilkan judul **P Chart** pada tabel **Output**.



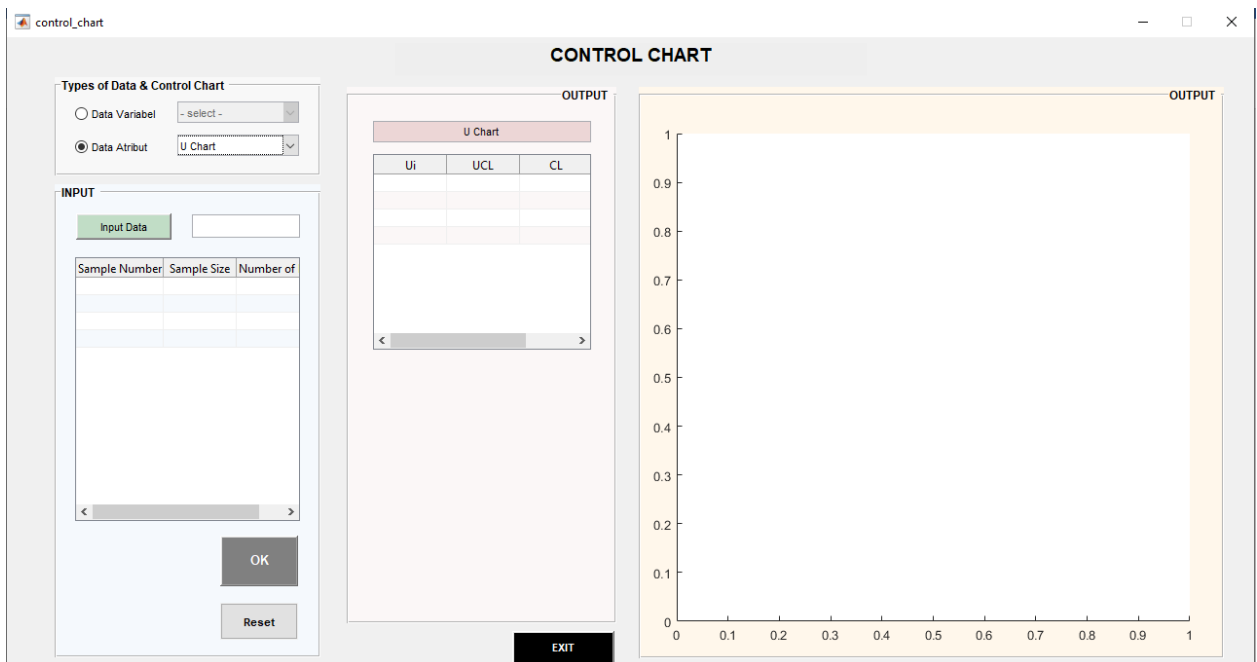
Ketika **NP Chart** dipilih, maka akan menampilkan judul **NP Chart** pada tabel **Output**.



Ketika **C Chart** dipilih, maka akan menampilkan judul **C Chart** pada tabel **Output**.



Ketika **U Chart** dipilih, maka akan menampilkan judul **U Chart** pada tabel **Output**.



Pemilihan jenis *control chart* pada popup-menu akan mengubah tabel **Output** sesuai dengan jenis *control chart* yang dipilih. Ketika popup-menu terpilih "- select -", baik pada **Data Variabel** maupun **Data Atribut**, maka tempat untuk memasukkan data akan dinonaktifkan. Sehingga, pengguna harus memilih jenis *control chart* terlebih dahulu sebelum mengolah data.

3.2.3 Panel Input

Panel **Input** berisi tempat untuk memasukkan data dan pushbutton **Input Data**, tabel yang berisi data yang akan diolah, panel untuk memasukkan nilai tabel *control chart* untuk data variabel, pushbutton **OK**, serta pushbutton **Reset**.

The screenshot shows the 'INPUT' panel with the following components:

- A green 'Input Data' button next to a text input field for file selection.
- A table with 3 columns: 'Sample Number', 'X1', and 'X2'. It contains 4 empty rows for data entry.
- A horizontal scrollbar below the table.
- An 'XBar-R' section containing three input fields labeled 'A2', 'D3', and 'D4'.
- 'OK' and 'Reset' buttons on the right side of the panel.

a. Input Data

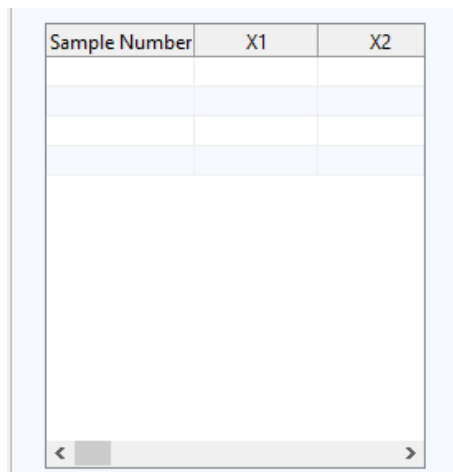
Kotak kosong digunakan untuk menuliskan nama file excel yang di dalamnya terdapat data yang akan diolah, sebagai contoh: xbar-r_chart.xlsx; p_chart.xlsx; dll. Pushbutton **Input Data** merupakan tombol untuk memasukkan data dari excel ke tabel yang ada pada **Input**.

This close-up shows the top part of the 'INPUT' panel, highlighting the green 'Input Data' button and the adjacent text input field for entering a file name.

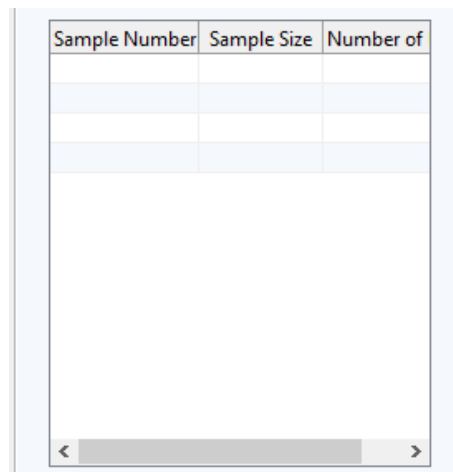
Ketika data sudah dimasukkan, tetapi ada perubahan jenis data maka nama data pada kotak akan terhapus dan data yang sudah dimasukkan dalam tabel akan terhapus.

b. Tabel

Tabel pada **Input** menampilkan data yang akan diolah. Terdapat dua tabel untuk data variabel dan data atribut. Tabel data variabel terdiri dari 26 kolom, dengan kolom ke-1 *Sample Number* adalah kolom untuk menampilkan nomor sampel dan kolom ke-2 hingga kolom ke-26 adalah subgrup X_1, X_2, \dots, X_{25} untuk menampilkan data observasi pada tiap sampel. Tabel data atribut terdiri dari 3 kolom, dengan kolom ke-1 *Sample Number* untuk menampilkan nomor sampel, kolom ke-2 *Sample Size* untuk menampilkan jumlah sampel yang diobservasi, dan kolom ke-3 *Number of Errors* adalah kolom yang menampilkan jumlah *defect*/kecacatan/kesalahan pada tiap sampel yang diamati.



Tabel data variabel



Tabel data atribut

c. Panel XBar-R dan XBar-S

Panel **XBar-R** dan **XBar-S** akan muncul saat memilih data variabel. **XBar-R** berisi tempat untuk memasukkan nilai A_2 , D_3 , dan D_4 tabel *control chart* berdasarkan jumlah observasi atau *subgroup size* pada tiap sampel dan akan muncul ketika memilih jenis *control chart* **XBar-R Chart**. Panel **XBar-S** berisi tempat untuk memasukkan nilai A_3 , B_3 , dan B_4 tabel *control chart* berdasarkan jumlah observasi atau *subgroup size* pada tiap sampel dan akan muncul ketika memilih jenis *control chart* **XBar-S Chart**.

Panel XBar-R interface showing three input fields labeled A2, D3, and D4.

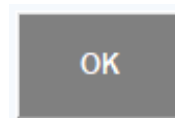
Panel XBar-R

Panel XBar-S interface showing three input fields labeled A3, B3, and B4.

Panel XBar-S

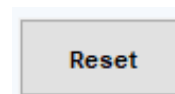
3.2.4 Pushbutton OK

Pushbutton OK adalah tombol untuk mengolah data yang telah dimasukkan berdasarkan jenis data dan jenis *control chart* yang sudah dipilih sebelumnya. Ketika tombol **OK** diklik, maka akan menghasilkan **Output** tabel dan peta kendali.



3.2.5 Pushbutton Reset

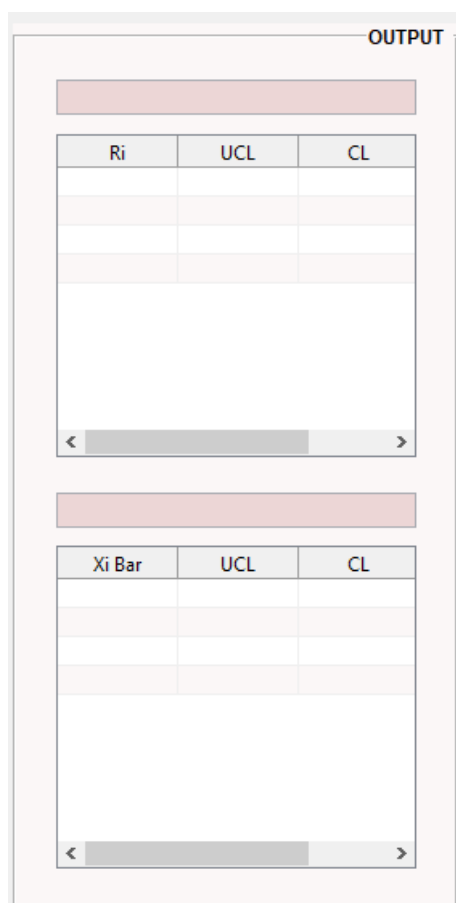
Pushbutton Reset adalah tombol untuk menghapus seluruh data pada tabel **Input** serta menghapus hasil dari data yang telah diolah dengan menghapus nilai pada tabel **Output** dan menghapus peta kendali **Output**. Ketika tombol **Reset** diklik, seluruh data dan hasil perhitungan serta peta kendali akan terhapus, dan pengguna dapat mengolah data yang sama maupun data baru dengan jenis data dan atau jenis *control chart* yang berbeda atau sama dari data sebelumnya.



3.2.6 Panel Output

a. Tabel

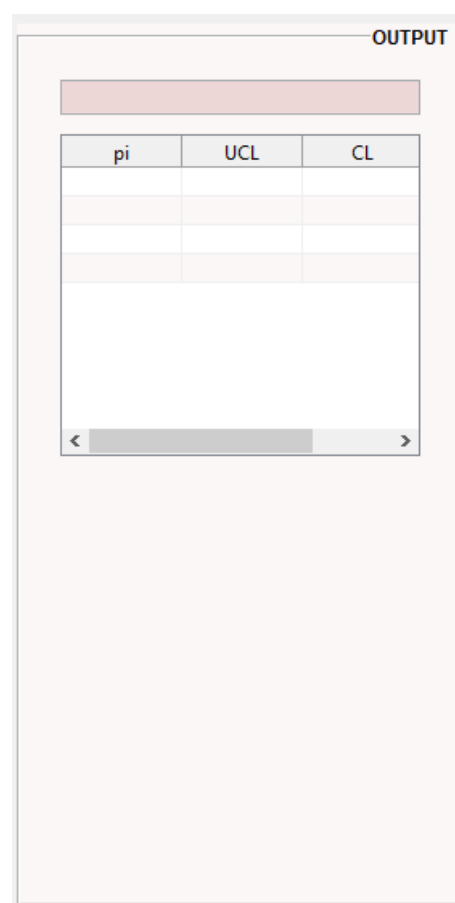
Tabel output akan menampilkan nilai UCL (*Upper Control Limit*) pada kolom ke-2, CL (*Center Line*) pada kolom ke-3, dan LCL (*Lower Control Limit*) pada kolom ke-4 dari hasil perhitungan yang dilakukan setelah mengklik tombol **OK**. Pada kolom ke-1, berisi nilai rata-rata pada tiap sampel. Tabel **Output** akan berbeda, bergantung pada jenis *control chart* yang akan digunakan. Jumlah tabel akan berbeda berdasarkan jenis data yang dipilih, yaitu akan ada 2 tabel jika memilih data variabel dan 1 tabel jika memilih data atribut.



Ri	UCL	CL

Xi Bar	UCL	CL

Tampilan untuk data variabel



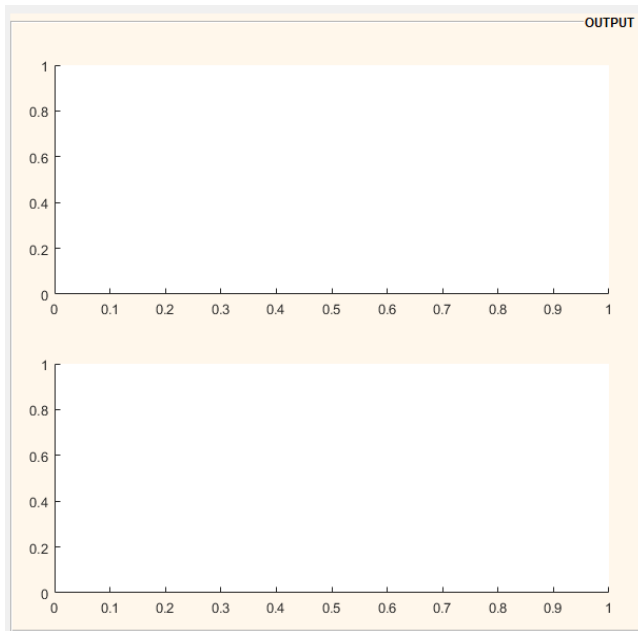
pi	UCL	CL

Tampilan untuk data atribut

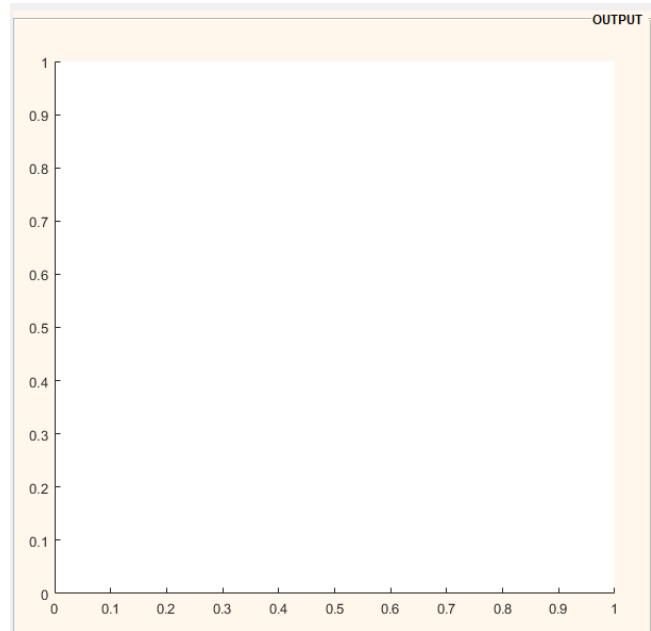
b. Grafik

Grafik pada **Output** akan menampilkan peta kendali sesuai jenis *control chart*. Jumlah grafik output yang ditampilkan akan berbeda tergantung jenis data yang dipilih. **Output**

akan menampilkan 2 grafik peta kendali untuk data variabel dan 1 grafik peta kendali untuk data atribut.



Tampilan untuk data variabel



Tampilan untuk data atribut

3.2.7 Pushbutton Exit

Pushbutton Exit merupakan tombol yang berfungsi untuk menutup GUI yang sedang dijalankan.



3.3 Langkah-Langkah

Langkah-langkah pengolahan data dengan menggunakan GUI *Control Chart* adalah sebagai berikut:

1. Jalankan GUI *control chart* (lihat 3.1)
2. Pilih **jenis data** yang akan digunakan pada panel **Types of Data & Control Chart**.
3. Pilih **jenis control chart** yang akan digunakan, pada popup-menu.
4. Masukkan **nama file** excel pada kotak kosong untuk memasukkan data pada panel **Input**.

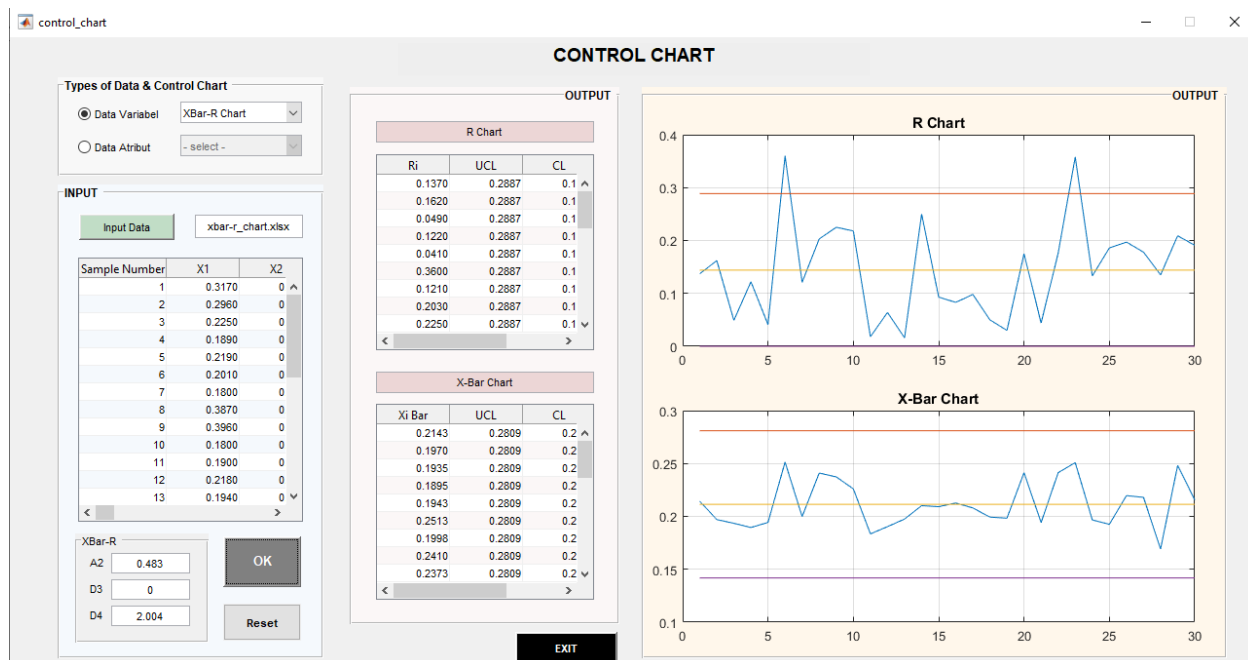
5. Klik pushbutton **Input Data**.
6. Jika jenis data yang digunakan adalah data variabel, maka masukkan nilai tabel *control chart* sesuai dengan jumlah observasi atau *subgroup size* pada tiap sampel pada panel **Xbar-R** atau **XBar-S**. Jika jenis data yang digunakan adalah data atribut, maka tidak perlu memasukkan nilai tabel *control chart* (nilai Tabel *Control Chart* dapat dilihat pada Lampiran B).
7. Klik **OK**. Maka GUI akan memproses data, kemudian nilai hasil perhitungan akan dimasukkan ke **Output** tabel dan **Output** grafik.
8. Klik **Reset** jika ingin mengolah data yang berbeda.
9. Langkah pengolahan data setelah **Reset** sama dengan langkah ke-2 hingga ke-7.
10. Klik **Exit** untuk keluar dari GUI *Control Chart*.

3.4 Hasil Perhitungan

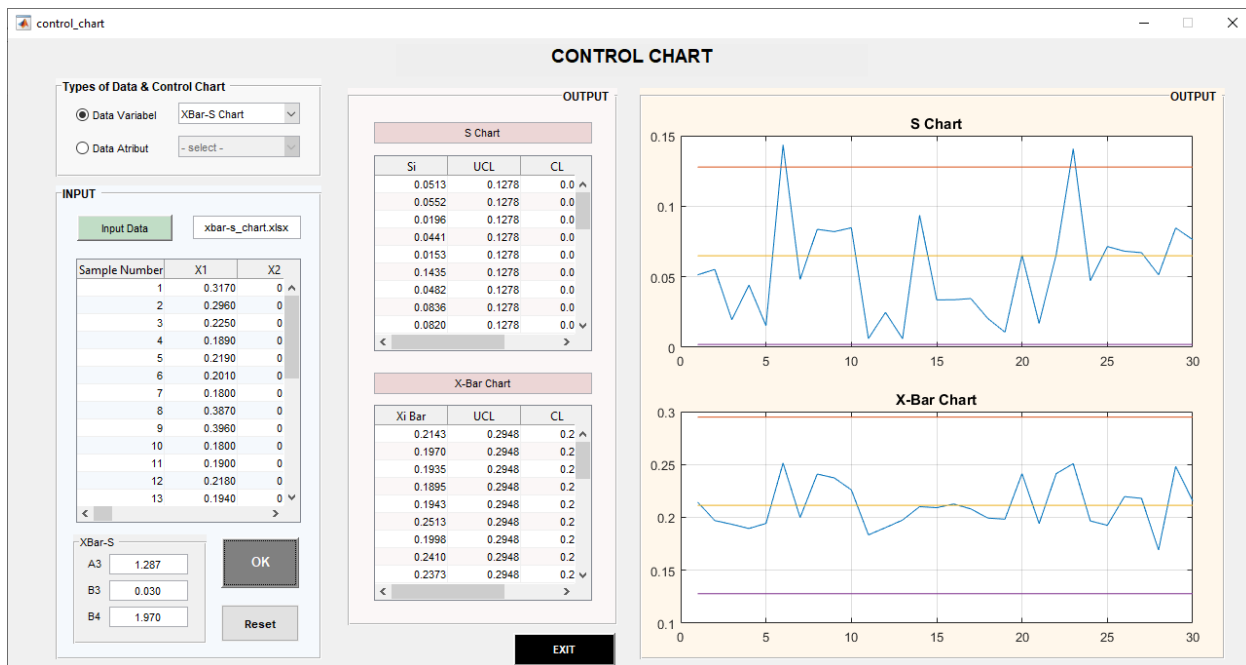
Berikut merupakan hasil pengolahan data dengan GUI *Control Chart*.

3.3.1 Data Variabel

a. XBar-R Chart

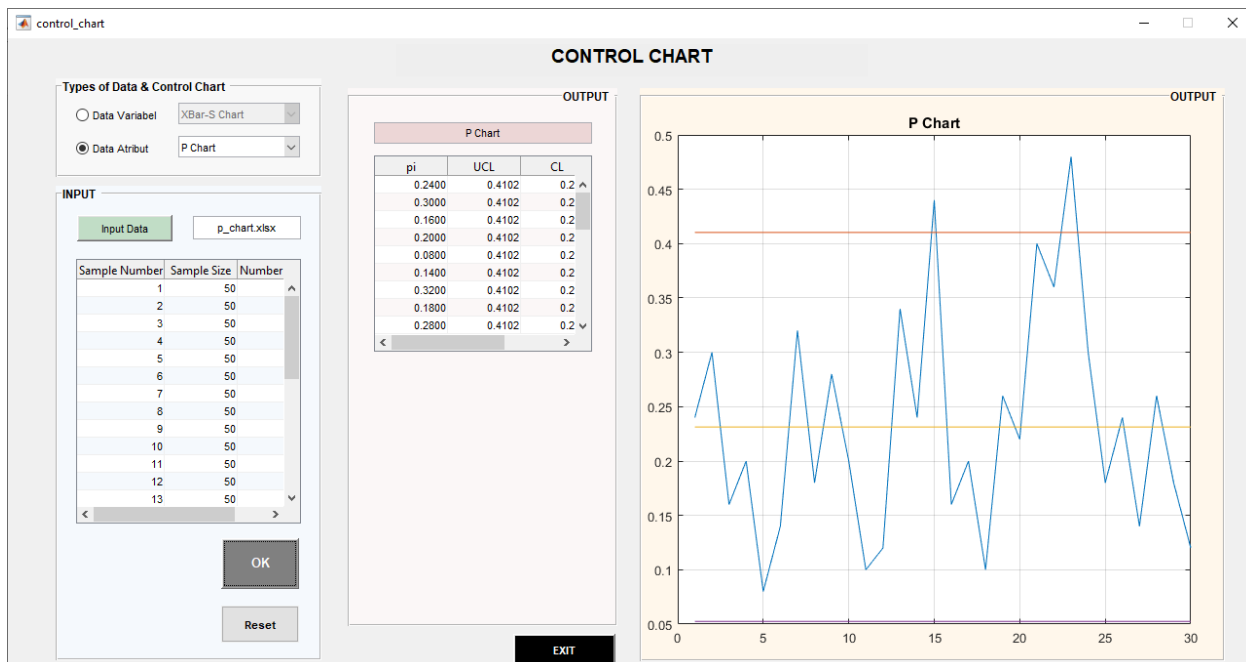


b. XBar-S Chart

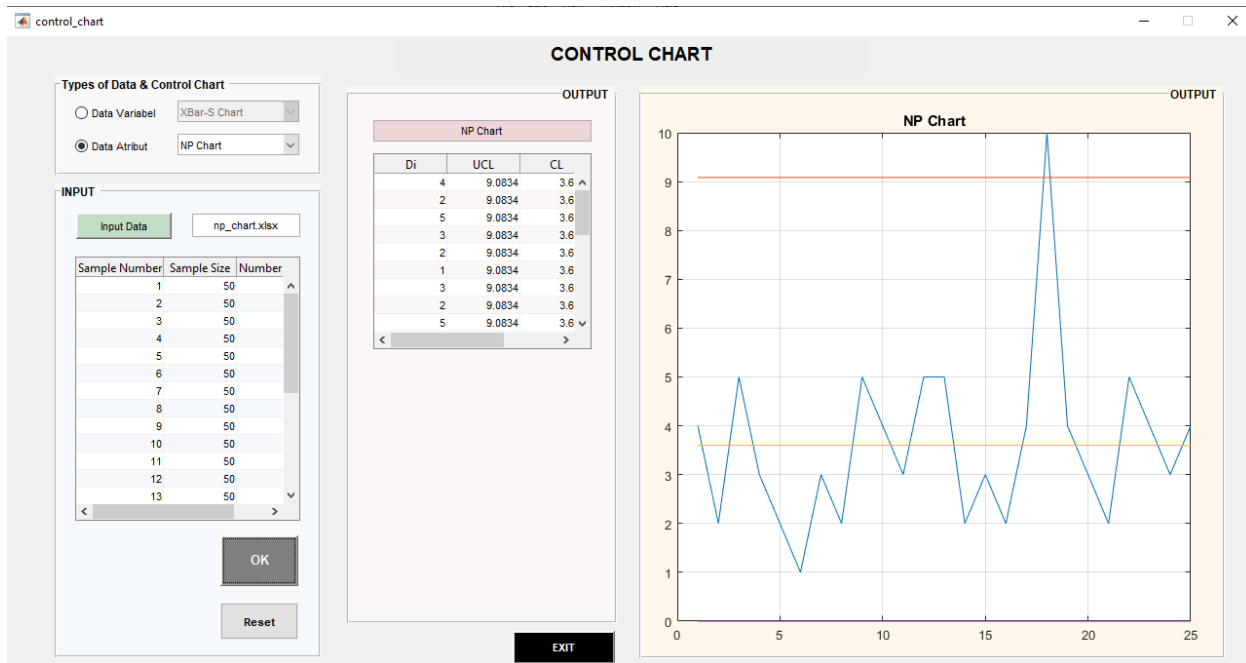


3.3.2 Data Atribut

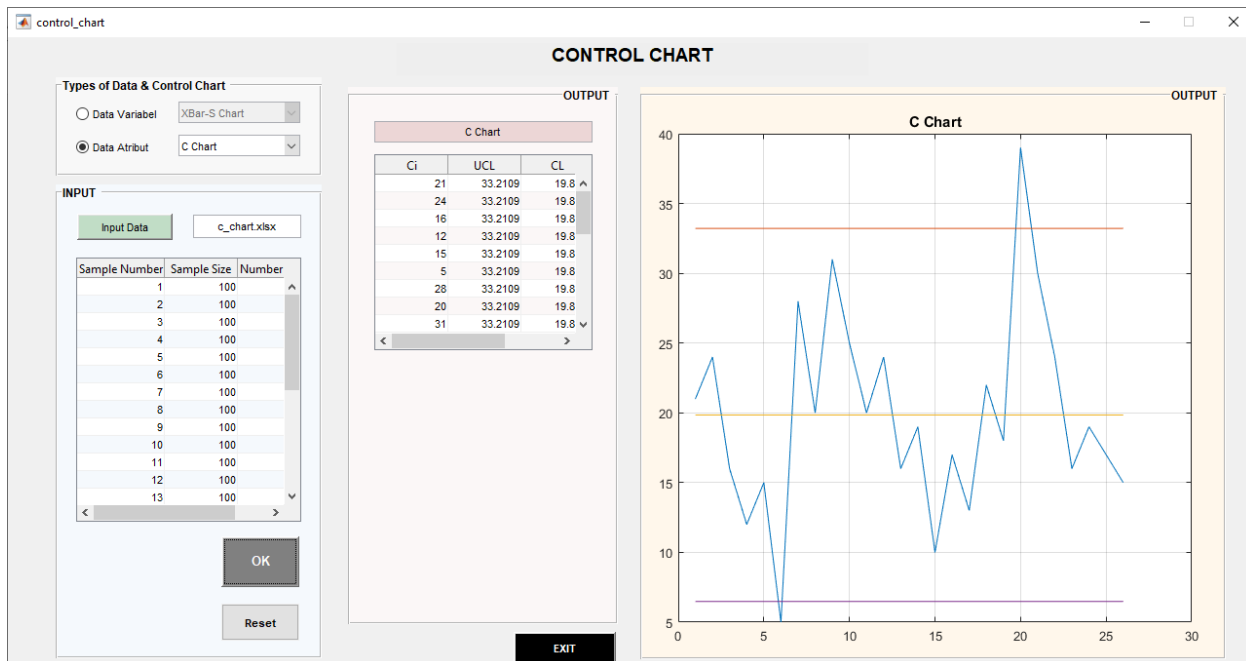
a. P Chart



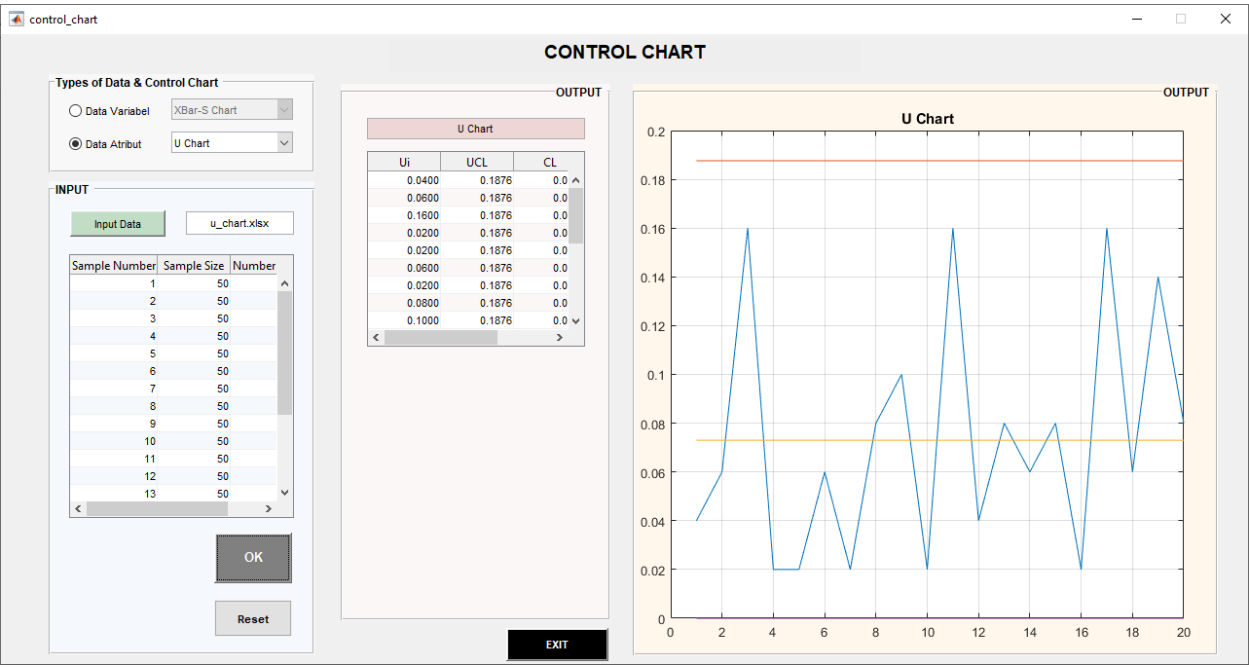
b. NP Chart



c. C Chart



d. U Chart



DAFTAR PUSTAKA

- Azizah, A. A. (2016). *Diagram Kontrol Fuzzy Multinomial Untuk Data Linguistik*. Bandung: Universitas Islam Bandung.
- Baradaran, V., & Dashtbani, H. (2014). A Decision Support System for Monitoring Traffic by Statistical Control Charts. *Management Science Letters* 4, 1661-1670.
- Dunning, D. (2017). *What Are UCL & LCL?* Retrieved October 8, 2021, from Sciencing: <https://sciencing.com/ucl-lcl-12011171.html>

LAMPIRAN

A. Syntax GUI MATLAB

Berikut merupakan syntax GUI **control_chart.m**.

```
function varargout = control_chart(varargin)
% CONTROL_CHART MATLAB code for control_chart.fig
%     CONTROL_CHART, by itself, creates a new CONTROL_CHART or raises the
existing
%     singleton*.
%
%     H = CONTROL_CHART returns the handle to a new CONTROL_CHART or the handle
to
%     the existing singleton*.
%
%     CONTROL_CHART('CALLBACK',hObject,eventData,handles,...) calls the local
function named CALLBACK in CONTROL_CHART.M with the given input arguments.
%
%     CONTROL_CHART('Property','Value',...) creates a new CONTROL_CHART or
raises the
%     existing singleton*. Starting from the left, property value pairs are
applied to the GUI before control_chart_OpeningFcn gets called. An
unrecognized property name or invalid value makes property application
stop. All inputs are passed to control_chart_OpeningFcn via varargin.
%
%     *See GUI Options on GUIDE's Tools menu. Choose "GUI allows only one
instance to run (singleton)".
%
% See also: GUIDE, GUIDATA, GUIHANDLES

% Edit the above text to modify the response to help control_chart

% Last Modified by GUIDE v2.5 08-Oct-2021 00:32:46

% Begin initialization code - DO NOT EDIT
gui_Singleton = 1;
gui_State = struct('gui_Name',       mfilename, ...
                  'gui_Singleton',   gui_Singleton, ...
                  'gui_OpeningFcn', @control_chart_OpeningFcn, ...
                  'gui_OutputFcn',  @control_chart_OutputFcn, ...
                  'gui_LayoutFcn',   [] , ...
                  'gui_Callback',    []);
if nargin && ischar(varargin{1})
    gui_State.gui_Callback = str2func(varargin{1});
end

if nargout
    [varargout{1:nargout}] = gui_mainfcn(gui_State, varargin{:});
else
    gui_mainfcn(gui_State, varargin{:});
end
% End initialization code - DO NOT EDIT

% --- Executes just before control_chart is made visible.
```

```

function control_chart_OpeningFcn(hObject, eventdata, handles, varargin)
% This function has no output args, see OutputFcn.
% hObject    handle to figure
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)
% varargin   command line arguments to control_chart (see VARARGIN)

% Choose default command line output for control_chart
handles.output = hObject;

% Update handles structure
guidata(hObject, handles);

% UIWAIT makes control_chart wait for user response (see UIRESUME)
% uiwait(handles.figure1);

% --- Outputs from this function are returned to the command line.
function varargout = control_chart_OutputFcn(hObject, eventdata, handles)
% varargout  cell array for returning output args (see VARARGOUT);
% hObject    handle to figure
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)

% Get default command line output from handles structure
varargout{1} = handles.output;

% --- Executes on button press in pushbutton_exit.
function pushbutton_exit_Callback(hObject, eventdata, handles)
% hObject    handle to pushbutton_exit (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)
close;

function edit_o2_Callback(hObject, eventdata, handles)
% hObject    handle to edit_o2 (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)

% Hints: get(hObject,'String') returns contents of edit_o2 as text
%        str2double(get(hObject,'String')) returns contents of edit_o2 as a
double

% --- Executes during object creation, after setting all properties.
function edit_o2_CreateFcn(hObject, eventdata, handles)
% hObject    handle to edit_o2 (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    empty - handles not created until after all CreateFcns called

% Hint: edit controls usually have a white background on Windows.
%       See ISPC and COMPUTER.
if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end

```

```

end

function edit_o1_Callback(hObject, eventdata, handles)
% hObject      handle to edit_o1 (see GCBO)
% eventdata    reserved - to be defined in a future version of MATLAB
% handles      structure with handles and user data (see GUIDATA)

% Hints: get(hObject,'String') returns contents of edit_o1 as text
%         str2double(get(hObject,'String')) returns contents of edit_o1 as a
double

% --- Executes during object creation, after setting all properties.
function edit_o1_CreateFcn(hObject, eventdata, handles)
% hObject      handle to edit_o1 (see GCBO)
% eventdata    reserved - to be defined in a future version of MATLAB
% handles      empty - handles not created until after all CreateFcns called

% Hint: edit controls usually have a white background on Windows.
%         See ISPC and COMPUTER.
if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end

% --- Executes on selection change in popupmenu_a.
function popupmenu_a_Callback(hObject, eventdata, handles)
% hObject      handle to popupmenu_a (see GCBO)
% eventdata    reserved - to be defined in a future version of MATLAB
% handles      structure with handles and user data (see GUIDATA)
a=get(handles.popupmenu_a,'Value');
if a==2
    set(handles.uitable_p,'Visible','on');
    set(handles.uitable_np,'Visible','off');
    set(handles.uitable_c,'Visible','off');
    set(handles.uitable_u,'Visible','off');
    set(handles.edit_id,'Enable','on');
    set(handles.edit_o1,'String','P Chart');
elseif a==3
    set(handles.uitable_p,'Visible','off');
    set(handles.uitable_np,'Visible','on');
    set(handles.uitable_c,'Visible','off');
    set(handles.uitable_u,'Visible','off');
    set(handles.edit_id,'Enable','on');
    set(handles.edit_o1,'String','NP Chart');
elseif a==4
    set(handles.uitable_p,'Visible','off');
    set(handles.uitable_np,'Visible','off');
    set(handles.uitable_c,'Visible','on');
    set(handles.uitable_u,'Visible','off');
    set(handles.edit_id,'Enable','on');
    set(handles.edit_o1,'String','C Chart');
elseif a==5
    set(handles.uitable_p,'Visible','off');
    set(handles.uitable_np,'Visible','off');

```

```

        set(handles.uitable_c, 'Visible', 'off');
        set(handles.uitable_u, 'Visible', 'on');
        set(handles.edit_id, 'Enable', 'on');
        set(handles.edit_o1, 'String', 'U Chart');
elseif a==1
    set(handles.edit_id, 'Enable', 'off');
    set(handles.edit_o1, 'String', '');
end
guidata(hObject, handles);
% Hints: contents = cellstr(get(hObject, 'String')) returns popupmenu_a contents
as cell array
%         contents{get(hObject, 'Value')} returns selected item from popupmenu_a

% --- Executes during object creation, after setting all properties.
function popupmenu_a_CreateFcn(hObject, eventdata, handles)
% hObject    handle to popupmenu_a (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    empty - handles not created until after all CreateFcns called

% Hint: popupmenu controls usually have a white background on Windows.
%         See ISPC and COMPUTER.
if ispc && isequal(get(hObject, 'BackgroundColor'),
get(0, 'defaultUiControlBackgroundColor'))
    set(hObject, 'BackgroundColor', 'white');
end

% --- Executes on selection change in popupmenu_v.
function popupmenu_v_Callback(hObject, eventdata, handles)
% hObject    handle to popupmenu_v (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)
v=get(handles.popupmenu_v, 'Value');
if v==2
    set(handles.uipanel_xbarr, 'Visible', 'on');
    set(handles.uipanel_xbars, 'Visible', 'off');
    set(handles.uitable_r, 'Visible', 'on');
    set(handles.uitable_s, 'Visible', 'off');

    set(handles.edit_id, 'Enable', 'on');
    set(handles.edit_o1, 'String', 'R Chart');
    set(handles.edit_o2, 'String', 'X-Bar Chart');
elseif v==3
    set(handles.uipanel_xbars, 'Visible', 'on');
    set(handles.uipanel_xbarr, 'Visible', 'off');
    set(handles.uitable_r, 'Visible', 'off');
    set(handles.uitable_s, 'Visible', 'on');
    set(handles.edit_id, 'Enable', 'on');
    set(handles.edit_o1, 'String', 'S Chart');
    set(handles.edit_o2, 'String', 'X-Bar Chart');
elseif v==1
    set(handles.edit_id, 'Enable', 'off');
    set(handles.edit_o1, 'String', '');
    set(handles.edit_o2, 'String', '');
end
guidata(hObject, handles);

```

```

% Hints: contents = cellstr(get(hObject,'String')) returns popupmenu_v contents
as cell array
%         contents{get(hObject,'Value')} returns selected item from popupmenu_v

% --- Executes during object creation, after setting all properties.
function popupmenu_v_CreateFcn(hObject, eventdata, handles)
% hObject    handle to popupmenu_v (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    empty - handles not created until after all CreateFcns called

% Hint: popupmenu controls usually have a white background on Windows.
%         See ISPC and COMPUTER.
if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUiControlBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end

% --- Executes on button press in pushbutton_id.
function pushbutton_id_Callback(hObject, eventdata, handles)
% hObject    handle to pushbutton_id (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)
data=xlsread(get(handles.edit_id,'string'));
handles.data=data;
if (get(handles.radiobutton_v,'Value')==1)
    set(handles.uitable_v,'Data',data);
else (get(handles.radiobutton_a,'Value')==1)
    set(handles.uitable_a,'Data',data);
end
guidata(hObject,handles);

function edit_id_Callback(hObject, eventdata, handles)
% hObject    handle to edit_id (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)

% Hints: get(hObject,'String') returns contents of edit_id as text
%         str2double(get(hObject,'String')) returns contents of edit_id as a
double

% --- Executes during object creation, after setting all properties.
function edit_id_CreateFcn(hObject, eventdata, handles)
% hObject    handle to edit_id (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    empty - handles not created until after all CreateFcns called

% Hint: edit controls usually have a white background on Windows.
%         See ISPC and COMPUTER.
if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUiControlBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end

```

```

% --- Executes on button press in pushbutton_ok.
function pushbutton_ok_Callback(hObject, eventdata, handles)
% hObject      handle to pushbutton_ok (see GCBO)
% eventdata    reserved - to be defined in a future version of MATLAB
% handles      structure with handles and user data (see GUIDATA)
v=get(handles.popupmenu_v,'Value');
a=get(handles.popupmenu_a,'Value');
if (get(handles.radiobutton_v,'Value')==1)
    if v==2
        A2=str2num(get(handles.edit_a2,'string'));
        D3=str2num(get(handles.edit_d3,'string'));
        D4=str2num(get(handles.edit_d4,'string'));
        data=handles.data;
        a=size(data);
        b=a(:,1);
        c=a(:,2);
        o=data(:,2:c);
        o=o';
        xi=sum(o);
        xi=xi';
        xi_bar=mean(o);
        xi_bar=xi_bar';
        xibarbar=sum(xi_bar)/b;

        Ri=max(o)-min(o);
        Ri=Ri';
        R_bar=sum(Ri)/b;

        CL_xbar=ones(b,1).*xibarbar;
        UCL_xbar=ones(b,1).*(xibarbar+A2*R_bar);
        LCL_xbar=ones(b,1).*(xibarbar-A2*R_bar);

        CL_R=ones(b,1).*R_bar;
        UCL_R=ones(b,1).*(D4*R_bar);
        LCL_R=ones(b,1).*(D3*R_bar);

        output_R=[Ri UCL_R CL_R LCL_R];
        output_xbar=[xi_bar UCL_xbar CL_xbar LCL_xbar];

        axes(handles.axes2)
        plot(output_xbar)
        title('X-Bar Chart');
        grid on;
        axes(handles.axes3)
        plot(output_R)
        title('R Chart');
        grid on;

        set(handles.uitable_r,'Data',output_R);
        set(handles.uitable_xbar,'Data',output_xbar);

    elseif v==3
        A3=str2num(get(handles.edit_a3,'string'));
        B3=str2num(get(handles.edit_b3,'string'));
        B4=str2num(get(handles.edit_b4,'string'));
        data=handles.data;
        a=size(data);

```

```

b=a(:,1);
c=a(:,2);
o=data(:,2:c);
o=o';
xi=sum(o);
xi=xi';
xi_bar=mean(o);
xi_bar=xi_bar';
xibarbar=sum(xi)/((c-1)*b);

for j=1:b
    s(j,:)=o(:,j)-xi_bar(j,1);
end

s2=s.^2;
s2=s2';
sums2=sum(s2);
sums2=sums2';

si2=sums2./(c-1-1);
si=sqrt(si2);

sbar=sqrt((sum((c-1-1)*si.^2))/((c-1-1)*b));

CL_xbar=ones(b,1).*xibarbar;
UCL_xbar=ones(b,1).*(xibarbar+A3*sbar);
LCL_xbar=ones(b,1).*(xibarbar-A3*sbar);

CL_S=ones(b,1).*sbar;
UCL_S=ones(b,1).*(B4*sbar);
LCL_S=ones(b,1).*(B3*sbar);

output_S=[si UCL_S CL_S LCL_S];
output_xbar=[xi_bar UCL_xbar CL_xbar LCL_xbar];

axes(handles.axes2)
plot(output_xbar)
title('X-Bar Chart');
grid on;
axes(handles.axes3)
plot(output_S)
title('S Chart');
grid on;

set(handles.uitable_s,'Data',output_S);
set(handles.uitable_xbar,'Data',output_xbar);

end

else (get(handles.radiobutton_a,'Value')==1)
    if a==2
        data=handles.data;
        x=length(data);
        ni=data(:,2);
        Di=data(:,3);
        Pi=Di./ni;
        pbar=sum(data(:,3))./sum(data(:,2));

```

```

CL_p=ones(x,1).*pbar;
UCL_p=ones(x,1)*(pbar)+(3*sqrt(pbar*(1-pbar))*sqrt(ni).^(-1));
LCL_p=ones(x,1)*(pbar)-(3*sqrt(pbar*(1-pbar))*sqrt(ni).^(-1));
LCL_p(find(LCL_p<0))=0;

output_p=[Pi UCL_p CL_p LCL_p];
axes(handles.axes1)
plot(output_p)
title('P Chart');
grid on;
set(handles.uitable_p,'Data',output_p);

elseif a==3
data=handles.data;
x=length(data);
ni=data(:,2);
Di=data(:,3);
Pi=Di./ni;
pbar=sum(data(:,3))./sum(data(:,2));
npbar=sum(Di)/x;

CL_npbar=ones(x,1).*npbar;
UCL_npbar=ones(x,1)*(npbar+3*sqrt(npbar*(1-pbar)));
LCL_npbar=ones(x,1)*(npbar-3*sqrt(npbar*(1-pbar)));
LCL_npbar(find(LCL_npbar<0))=0;

output_np=[Di UCL_npbar CL_npbar LCL_npbar];
axes(handles.axes1)
plot(output_np)
title('NP Chart');
grid on;
set(handles.uitable_np,'Data',output_np);

elseif a==4
data=handles.data;
x=length(data);
Ci=data(:,3);
cbar=sum(Ci)/x;

CL_c=ones(x,1).*cbar;
UCL_c=ones(x,1)*(cbar)+(3*sqrt(cbar));
LCL_c=ones(x,1)*(cbar)-(3*sqrt(cbar));
LCL_c(find(LCL_c<0))=0;

output_c=[Ci UCL_c CL_c LCL_c];
axes(handles.axes1)
plot(output_c)
title('C Chart');
grid on;
set(handles.uitable_c,'Data',output_c);

elseif a==5
data=handles.data;
x=length(data);
ni=data(:,2);
Xi=data(:,3);
Ui=Xi./ni;

```



```

        ubar=sum(Xi)/sum(ni);

        CL_u=ones(x,1).*ubar;
        UCL_u=ones(x,1)*(ubar)+(3*sqrt(ubar)*sqrt(ni).^(-1));
        LCL_u=ones(x,1)*(ubar)-(3*sqrt(ubar)*sqrt(ni).^(-1));
        LCL_u(find(LCL_u<0))=0;

        output_u=[Ui UCL_u CL_u LCL_u];
        axes(handles.axes1)
        plot(output_u)
        title('U Chart');
        grid on;
        set(handles.uitable_u,'Data',output_u);

    end
end

% --- Executes on button press in pushbutton_res.
function pushbutton_res_Callback(hObject, eventdata, handles)
% hObject      handle to pushbutton_res (see GCBO)
% eventdata    reserved - to be defined in a future version of MATLAB
% handles      structure with handles and user data (see GUIDATA)
set(handles.edit_id,'String','');
set(handles.edit_o1,'String','');
v=get(handles.popupmenu_v,'Value');
a=get(handles.popupmenu_a,'Value');
if (get(handles.radiobutton_v,'Value')==1)
    cla(handles.axes2,'reset');
    cla(handles.axes3,'reset');
    set(handles.uitable_v,'Data',{});
    set(handles.edit_o2,'String','');
    set(handles.uitable_xbar,'Data',{});
    if v==2
        set(handles.edit_a2,'String','');
        set(handles.edit_d3,'String','');
        set(handles.edit_d4,'String','');
        set(handles.uitable_r,'Data',{});
    elseif v==3
        set(handles.edit_a3,'String','');
        set(handles.edit_b3,'String','');
        set(handles.edit_b4,'String','');
        set(handles.uitable_s,'Data',{});
    end
else (get(handles.radiobutton_a,'Value')==1)
    cla(handles.axes1,'reset');
    set(handles.uitable_a,'Data',{});
    if a==2
        set(handles.uitable_p,'Data',{});
    elseif a==3
        set(handles.uitable_np,'Data',{});
    elseif a==4
        set(handles.uitable_c,'Data',{});
    elseif a==5
        set(handles.uitable_u,'Data',{});
    end
end
end

```

```

function edit_d4_Callback(hObject, eventdata, handles)
% hObject      handle to edit_d4 (see GCBO)
% eventdata    reserved - to be defined in a future version of MATLAB
% handles      structure with handles and user data (see GUIDATA)

% Hints: get(hObject,'String') returns contents of edit_d4 as text
%         str2double(get(hObject,'String')) returns contents of edit_d4 as a
double

% --- Executes during object creation, after setting all properties.
function edit_d4_CreateFcn(hObject, eventdata, handles)
% hObject      handle to edit_d4 (see GCBO)
% eventdata    reserved - to be defined in a future version of MATLAB
% handles      empty - handles not created until after all CreateFcns called

% Hint: edit controls usually have a white background on Windows.
%         See ISPC and COMPUTER.
if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUiControlBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end

function edit_d3_Callback(hObject, eventdata, handles)
% hObject      handle to edit_d3 (see GCBO)
% eventdata    reserved - to be defined in a future version of MATLAB
% handles      structure with handles and user data (see GUIDATA)

% Hints: get(hObject,'String') returns contents of edit_d3 as text
%         str2double(get(hObject,'String')) returns contents of edit_d3 as a
double

% --- Executes during object creation, after setting all properties.
function edit_d3_CreateFcn(hObject, eventdata, handles)
% hObject      handle to edit_d3 (see GCBO)
% eventdata    reserved - to be defined in a future version of MATLAB
% handles      empty - handles not created until after all CreateFcns called

% Hint: edit controls usually have a white background on Windows.
%         See ISPC and COMPUTER.
if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUiControlBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end

function edit_a2_Callback(hObject, eventdata, handles)
% hObject      handle to edit_a2 (see GCBO)
% eventdata    reserved - to be defined in a future version of MATLAB
% handles      structure with handles and user data (see GUIDATA)

% Hints: get(hObject,'String') returns contents of edit_a2 as text

```

```

%      str2double(get(hObject,'String')) returns contents of edit_a2 as a
double

% --- Executes during object creation, after setting all properties.
function edit_a2_CreateFcn(hObject, eventdata, handles)
% hObject    handle to edit_a2 (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    empty - handles not created until after all CreateFcns called

% Hint: edit controls usually have a white background on Windows.
%      See ISPC and COMPUTER.
if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end

function edit_b4_Callback(hObject, eventdata, handles)
% hObject    handle to edit_b4 (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)

% Hints: get(hObject,'String') returns contents of edit_b4 as text
%      str2double(get(hObject,'String')) returns contents of edit_b4 as a
double

% --- Executes during object creation, after setting all properties.
function edit_b4_CreateFcn(hObject, eventdata, handles)
% hObject    handle to edit_b4 (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    empty - handles not created until after all CreateFcns called

% Hint: edit controls usually have a white background on Windows.
%      See ISPC and COMPUTER.
if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end

function edit_b3_Callback(hObject, eventdata, handles)
% hObject    handle to edit_b3 (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)

% Hints: get(hObject,'String') returns contents of edit_b3 as text
%      str2double(get(hObject,'String')) returns contents of edit_b3 as a
double

% --- Executes during object creation, after setting all properties.
function edit_b3_CreateFcn(hObject, eventdata, handles)

```

```

% hObject    handle to edit_b3 (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    empty - handles not created until after all CreateFcns called

% Hint: edit controls usually have a white background on Windows.
%         See ISPC and COMPUTER.
if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUiControlBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end

function edit_a3_Callback(hObject, eventdata, handles)
% hObject    handle to edit_a3 (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)

% Hints: get(hObject,'String') returns contents of edit_a3 as text
%         str2double(get(hObject,'String')) returns contents of edit_a3 as a
double

% --- Executes during object creation, after setting all properties.
function edit_a3_CreateFcn(hObject, eventdata, handles)
% hObject    handle to edit_a3 (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    empty - handles not created until after all CreateFcns called

% Hint: edit controls usually have a white background on Windows.
%         See ISPC and COMPUTER.
if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUiControlBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end

% --- Executes on button press in radiobutton_a.
function radiobutton_a_Callback(hObject, eventdata, handles)
% hObject    handle to radiobutton_a (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)
set(handles.uitable_v,'Visible','off');
set(handles.uitable_a,'Visible','on');
set(handles.popupmenu_v,'Enable','off');
set(handles.popupmenu_a,'Enable','on');
set(handles.uipanel_xbarr,'Visible','off');
set(handles.uipanel_xbars,'Visible','off');
set(handles.edit_id,'string','');
set(handles.uitable_v,'Data',{});
set(handles.uipanel_a,'Visible','on');
set(handles.uipanel_v,'Visible','off');
set(handles.uitable_xbar,'Visible','off');
set(handles.edit_o2,'Visible','off');
set(handles.uitable_r,'Visible','off');
set(handles.uitable_s,'Visible','off');
set(handles.uitable_p,'Visible','on');

```

```

set(handles.uitable_np,'Visible','on');
set(handles.uitable_c,'Visible','on');
set(handles.uitable_u,'Visible','on');
guidata(hObject,handles);

% Hint: get(hObject,'Value') returns toggle state of radiobutton_a

% --- Executes on button press in radiobutton_v.
function radiobutton_v_Callback(hObject, eventdata, handles)
% hObject      handle to radiobutton_v (see GCBO)
% eventdata    reserved - to be defined in a future version of MATLAB
% handles      structure with handles and user data (see GUIDATA)
set(handles.uitable_v,'Visible','on');
set(handles.uitable_a,'Visible','off');
set(handles.popupmenu_v,'Enable','on');
set(handles.popupmenu_a,'Enable','off');
set(handles.uipanel_xbarr,'Visible','on');
set(handles.uipanel_xbars,'Visible','on');
set(handles.edit_id,'string','');
set(handles.uitable_a,'Data',{});
set(handles.uipanel_a,'Visible','off');
set(handles.uipanel_v,'Visible','on');
set(handles.uitable_xbar,'Visible','on');
set(handles.edit_o2,'Visible','on');
set(handles.uitable_r,'Visible','on');
set(handles.uitable_s,'Visible','on');
set(handles.uitable_p,'Visible','off');
set(handles.uitable_np,'Visible','off');
set(handles.uitable_c,'Visible','off');
set(handles.uitable_u,'Visible','off');
guidata(hObject,handles);
% Hint: get(hObject,'Value') returns toggle state of radiobutton v

```

B. Tabel Control Chart

Subgroup size	A ₂	A ₃	D ₃	D ₄	B ₃	B ₄
2	1.880	2.659	0	3.267	0	3.267
3	1.023	1.954	0	2.574	0	2.568
4	0.729	1.628	0	2.282	0	2.266
5	0.577	1.427	0	2.114	0	2.089
6	0.483	1.287	0	2.004	0.030	1.970
7	0.419	1.182	0.076	1.924	0.118	1.882
8	0.373	1.099	0.136	1.864	0.185	1.815
9	0.337	1.032	0.184	1.816	0.239	1.761
10	0.308	0.975	0.223	1.777	0.284	1.716
11	0.285	0.927	0.256	1.744	0.321	1.679
12	0.266	0.886	0.283	1.717	0.354	1.646
13	0.249	0.850	0.307	1.693	0.382	1.618
14	0.235	0.817	0.328	1.672	0.406	1.594
15	0.223	0.789	0.347	1.653	0.428	1.572
16	0.212	0.763	0.363	1.637	0.448	1.552
17	0.203	0.739	0.378	1.622	0.466	1.534
18	0.194	0.718	0.391	1.608	0.482	1.518
19	0.187	0.698	0.403	1.597	0.497	1.503
20	0.180	0.680	0.415	1.585	0.510	1.490
21	0.173	0.663	0.425	1.575	0.523	1.477
22	0.167	0.647	0.434	1.566	0.534	1.466
23	0.162	0.633	0.443	1.557	0.545	1.455
24	0.157	0.619	0.451	1.548	0.555	1.445
25	0.153	0.606	0.459	1.541	0.565	1.435

C. Data Penelitian Excel

1. XBar-R Chart

Sample Number	X1	X2	X3	X4	X5	X6
1	0.317	0.2	0.18	0.188	0.191	0.21
2	0.296	0.217	0.183	0.176	0.176	0.134
3	0.225	0.21	0.181	0.176	0.188	0.181
4	0.189	0.172	0.165	0.181	0.154	0.276
5	0.219	0.206	0.184	0.191	0.188	0.178
6	0.201	0.198	0.544	0.193	0.188	0.184
7	0.18	0.178	0.177	0.181	0.185	0.298
8	0.387	0.298	0.198	0.188	0.184	0.191
9	0.396	0.224	0.19	0.245	0.171	0.198
10	0.18	0.191	0.184	0.398	0.204	0.199
11	0.19	0.185	0.172	0.183	0.184	0.187
12	0.218	0.154	0.218	0.191	0.178	0.183
13	0.194	0.2	0.204	0.203	0.196	0.188
14	0.399	0.149	0.164	0.178	0.188	0.183
15	0.271	0.222	0.193	0.19	0.178	0.201
16	0.266	0.241	0.208	0.183	0.188	0.19
17	0.261	0.163	0.194	0.188	0.211	0.231
18	0.2	0.18	0.186	0.183	0.23	0.217
19	0.198	0.218	0.194	0.188	0.201	0.191
20	0.207	0.193	0.363	0.254	0.243	0.188
21	0.206	0.197	0.193	0.177	0.218	0.174
22	0.226	0.182	0.276	0.176	0.236	0.352
23	0.536	0.191	0.226	0.186	0.188	0.178
24	0.167	0.178	0.278	0.191	0.221	0.145
25	0.289	0.267	0.179	0.167	0.15	0.103
26	0.245	0.332	0.217	0.167	0.222	0.135
27	0.343	0.245	0.178	0.189	0.188	0.165
28	0.105	0.24	0.19	0.113	0.178	0.189
29	0.398	0.21	0.2	0.301	0.189	0.191
30	0.188	0.17	0.341	0.278	0.15	0.167

2. XBar-S Chart

Sample Number	X1	X2	X3	X4	X5	X6
1	0.317	0.2	0.18	0.188	0.191	0.21
2	0.296	0.217	0.183	0.176	0.176	0.134
3	0.225	0.21	0.181	0.176	0.188	0.181
4	0.189	0.172	0.165	0.181	0.154	0.276

5	0.219	0.206	0.184	0.191	0.188	0.178
6	0.201	0.198	0.544	0.193	0.188	0.184
7	0.18	0.178	0.177	0.181	0.185	0.298
8	0.387	0.298	0.198	0.188	0.184	0.191
9	0.396	0.224	0.19	0.245	0.171	0.198
10	0.18	0.191	0.184	0.398	0.204	0.199
11	0.19	0.185	0.172	0.183	0.184	0.187
12	0.218	0.154	0.218	0.191	0.178	0.183
13	0.194	0.2	0.204	0.203	0.196	0.188
14	0.399	0.149	0.164	0.178	0.188	0.183
15	0.271	0.222	0.193	0.19	0.178	0.201
16	0.266	0.241	0.208	0.183	0.188	0.19
17	0.261	0.163	0.194	0.188	0.211	0.231
18	0.2	0.18	0.186	0.183	0.23	0.217
19	0.198	0.218	0.194	0.188	0.201	0.191
20	0.207	0.193	0.363	0.254	0.243	0.188
21	0.206	0.197	0.193	0.177	0.218	0.174
22	0.226	0.182	0.276	0.176	0.236	0.352
23	0.536	0.191	0.226	0.186	0.188	0.178
24	0.167	0.178	0.278	0.191	0.221	0.145
25	0.289	0.267	0.179	0.167	0.15	0.103
26	0.245	0.332	0.217	0.167	0.222	0.135
27	0.343	0.245	0.178	0.189	0.188	0.165
28	0.105	0.24	0.19	0.113	0.178	0.189
29	0.398	0.21	0.2	0.301	0.189	0.191
30	0.188	0.17	0.341	0.278	0.15	0.167

3. P Chart

Sample Number	Sample Size	Number of Errors
1	50	12
2	50	15
3	50	8
4	50	10
5	50	4
6	50	7
7	50	16
8	50	9
9	50	14
10	50	10
11	50	5
12	50	6

13	50	17
14	50	12
15	50	22
16	50	8
17	50	10
18	50	5
19	50	13
20	50	11
21	50	20
22	50	18
23	50	24
24	50	15
25	50	9
26	50	12
27	50	7
28	50	13
29	50	9
30	50	6

4. NP Chart

Sample Number	Sample Size	Number of Errors
1	50	4
2	50	2
3	50	5
4	50	3
5	50	2
6	50	1
7	50	3
8	50	2
9	50	5
10	50	4
11	50	3
12	50	5
13	50	5
14	50	2
15	50	3
16	50	2
17	50	4
18	50	10
19	50	4
20	50	3

21	50	2
22	50	5
23	50	4
24	50	3
25	50	4

5. C Chart

Sample Number	Sample Size	Number of Errors
1	100	21
2	100	24
3	100	16
4	100	12
5	100	15
6	100	5
7	100	28
8	100	20
9	100	31
10	100	25
11	100	20
12	100	24
13	100	16
14	100	19
15	100	10
16	100	17
17	100	13
18	100	22
19	100	18
20	100	39
21	100	30
22	100	24
23	100	16
24	100	19
25	100	17
26	100	15

6. U Chart

Sample Number	Sample Size	Number of Errors
1	50	2
2	50	3

3	50	8
4	50	1
5	50	1
6	50	3
7	50	1
8	50	4
9	50	5
10	50	1
11	50	8
12	50	2
13	50	4
14	50	3
15	50	4
16	50	1
17	50	8
18	50	3
19	50	7
20	50	4