Monte Carlo K-Means Clustering SAS Enterprise Miner

Donald K. Wedding, PhD
Director of Data Science
Sprint Corporation
dwedding@acm.org

What Is Clustering?

K-Means Clustering

- Technique can be used on other data such as CUSTOMER data

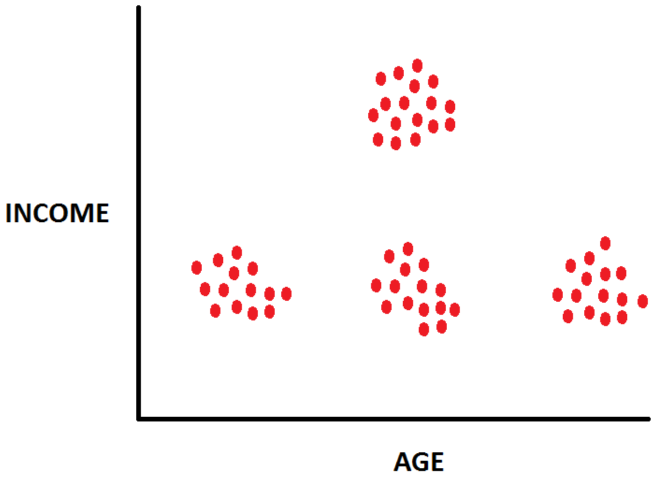
- K-Means clustering allows for grouping multiple variables simultaneously

- More sophisticated treatment of customers than is possible from simple segmentation

K-Means Clustering

Clusters based on AGE and INCOME

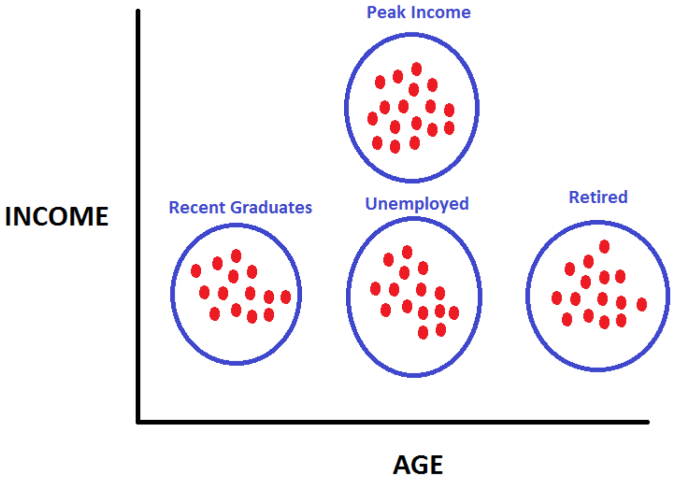
How many clusters do you see?



K-Means Clustering

Visual Inspection “proc eyeball”

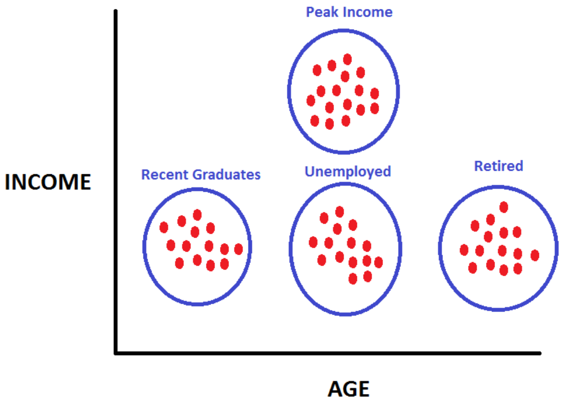
There are FOUR clusters.



K-Means Clustering

A bank might use these clusters for “cross sell”

- Recent Graduates : Overdraft Protection
- Peak Income : Mortgage, Heloc , Investment Account
- Retired : Trust Fund, Retirement Account, Estate Planning
- Unemployed : Unprofitable – “Choose to Lose”



What Affects Cluster Quality?

What Affects Cluster Results?

- How many clusters are there?
- Cluster Starting Points (“Seeds”)?

What Affects Cluster Results?

- How many clusters are there?

- Cluster Starting Points (“Seeds”)?

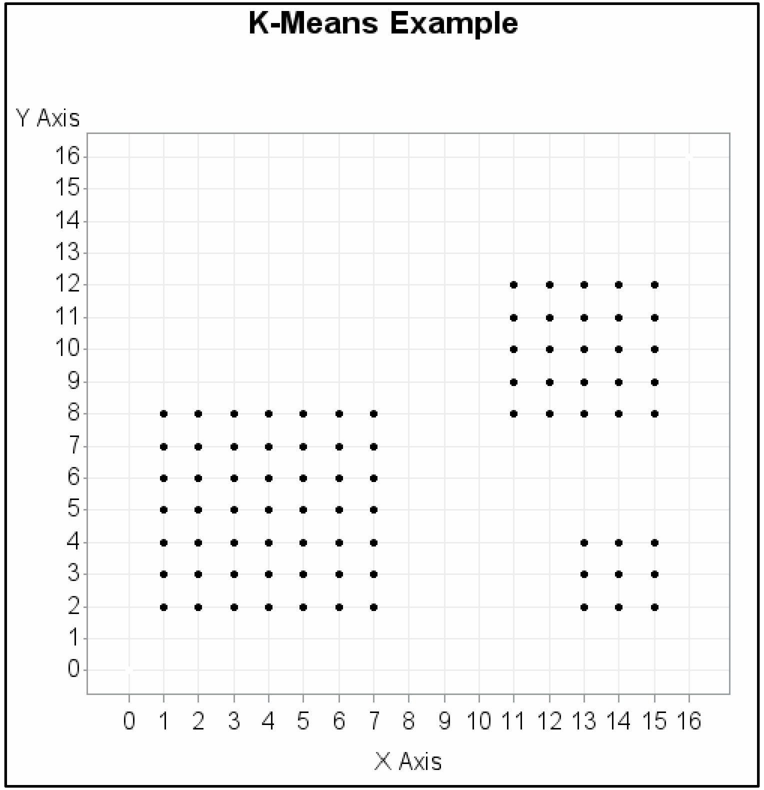
How Many Clusters?

How Many Clusters: Example

Given the Following Data Points

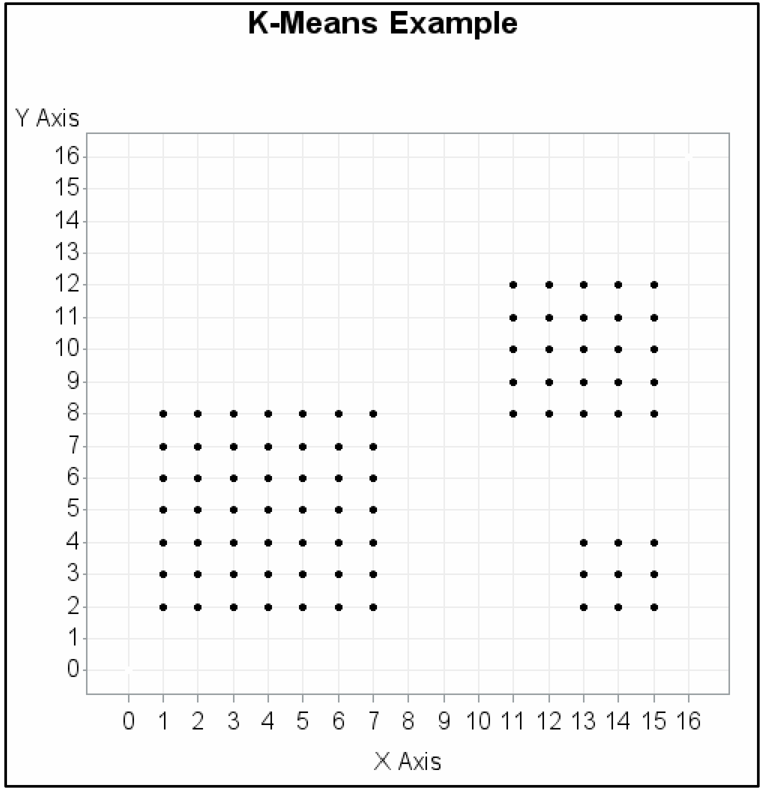
- Find the cluster centers for N=2 Clusters
- Find the cluster centers for N=3 Clusters
- Find the cluster centers for N=4 Clusters

How Many Clusters: Example



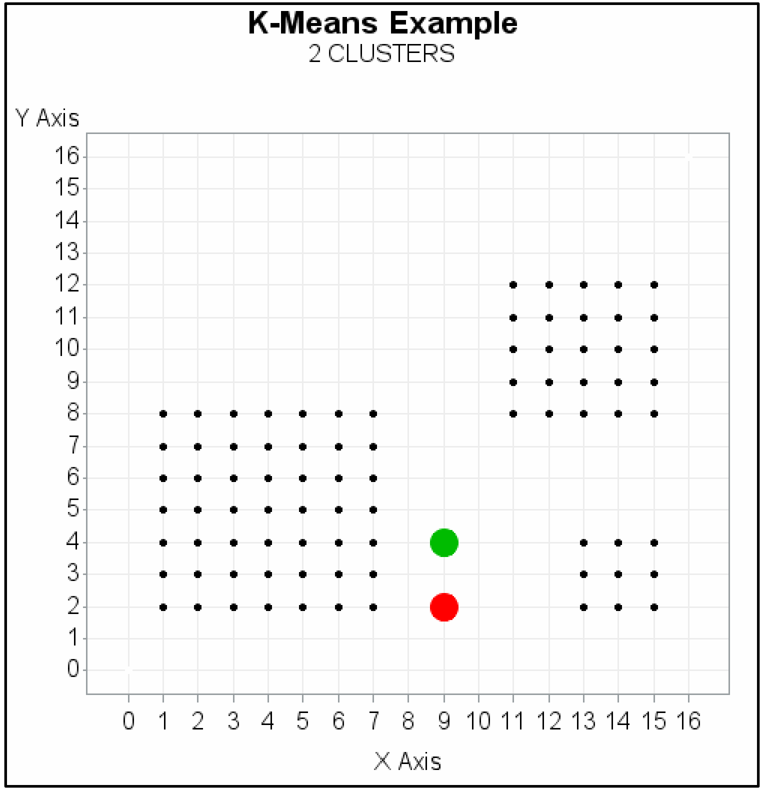
How Many Clusters: Example

2 Clusters



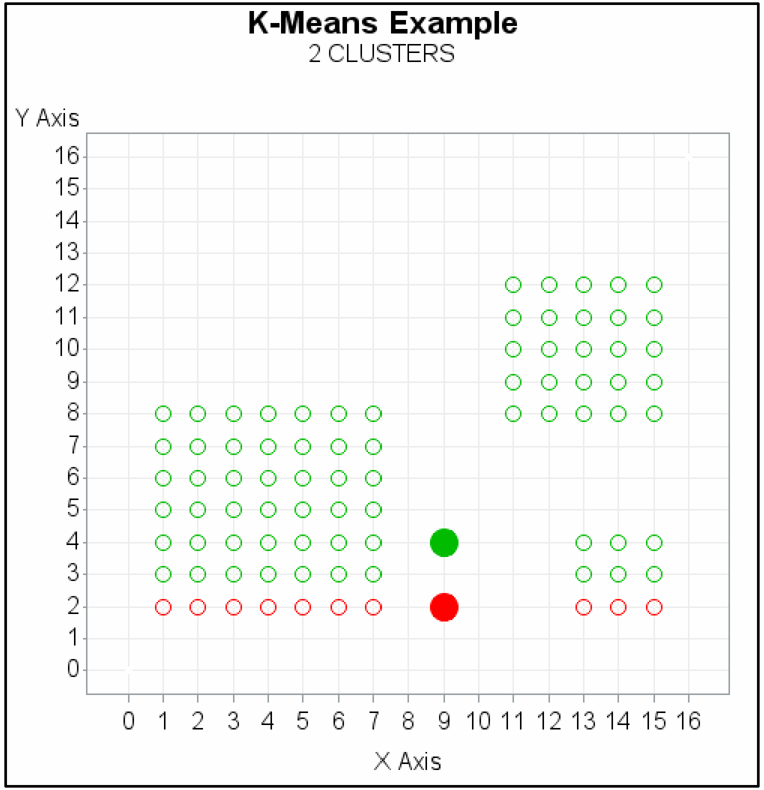
How Many Clusters: Example

2 Clusters



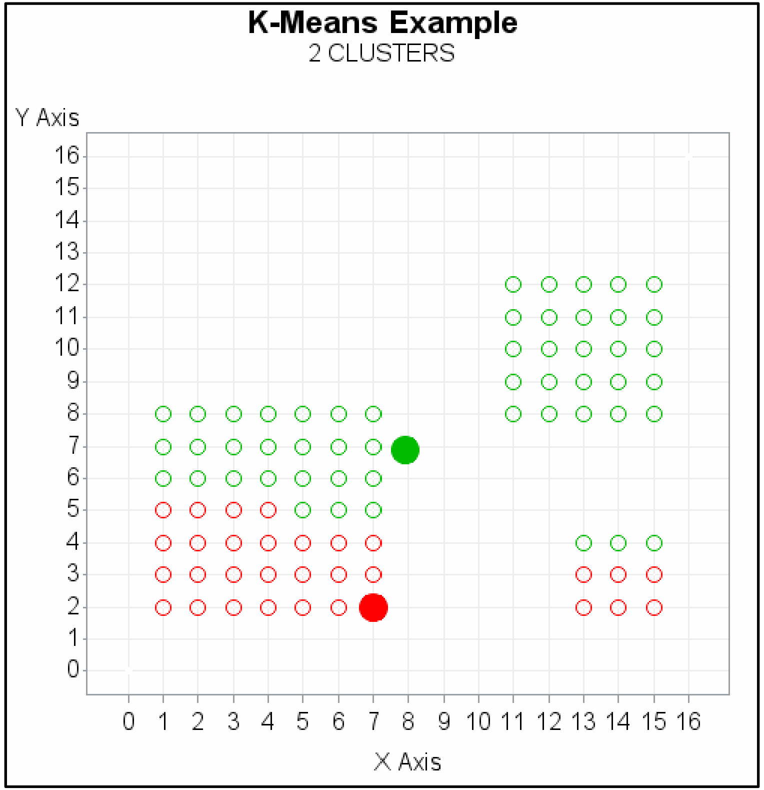
How Many Clusters: Example

2 Clusters



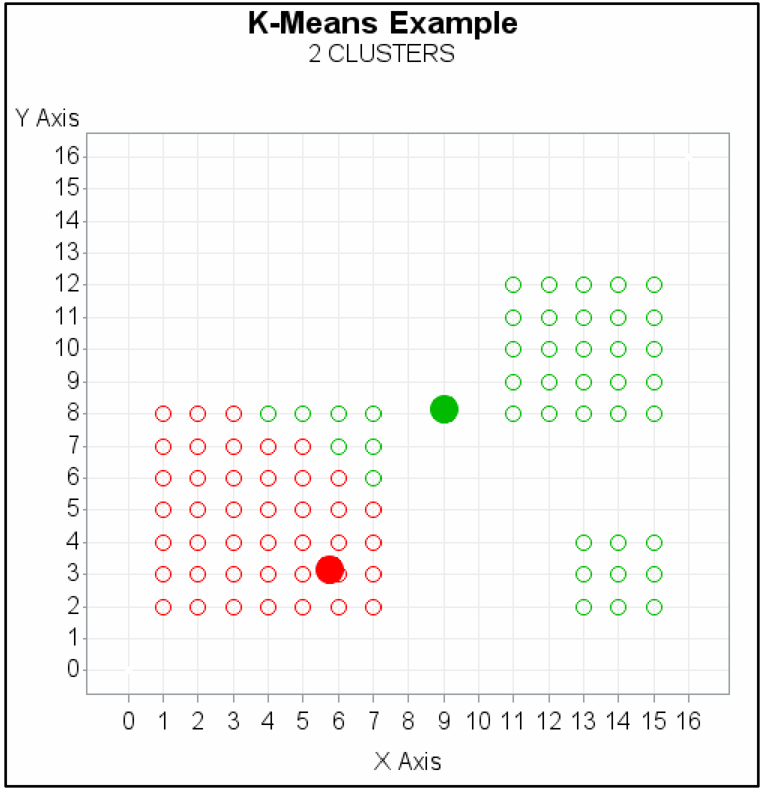
How Many Clusters: Example

2 Clusters



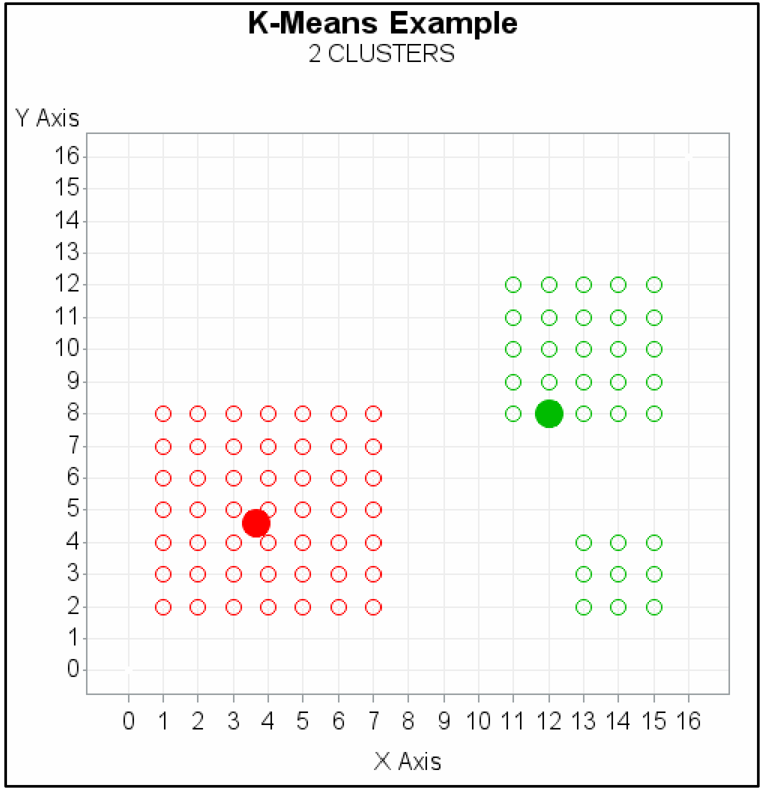
How Many Clusters: Example

2 Clusters



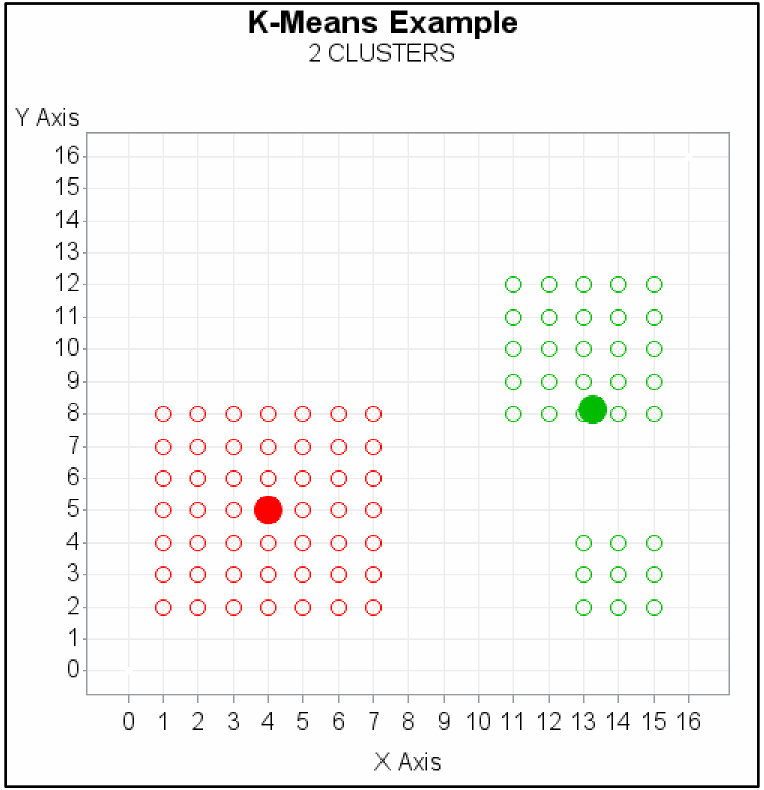
How Many Clusters: Example

2 Clusters



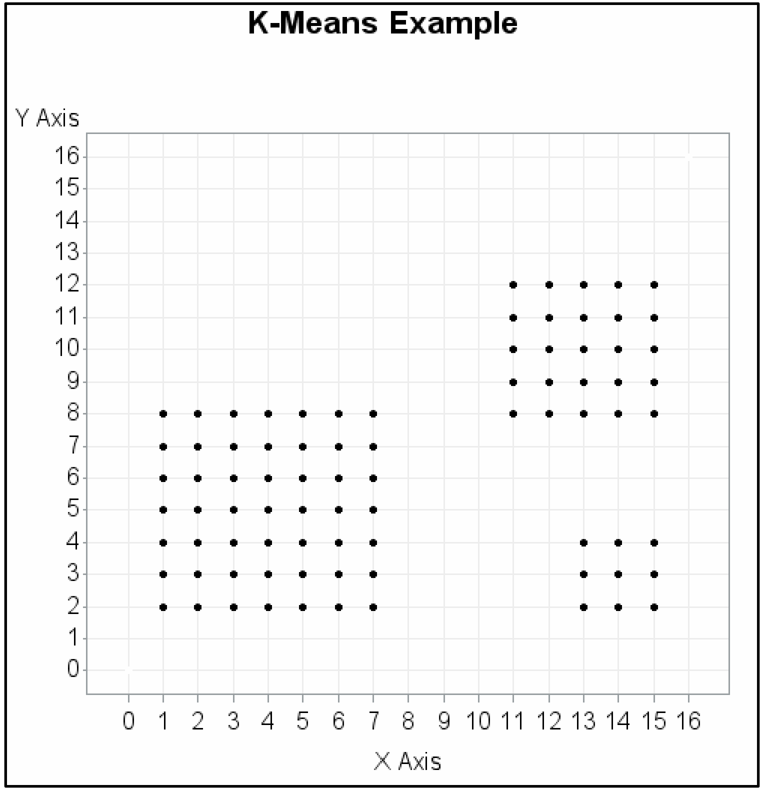
How Many Clusters: Example

2 Clusters



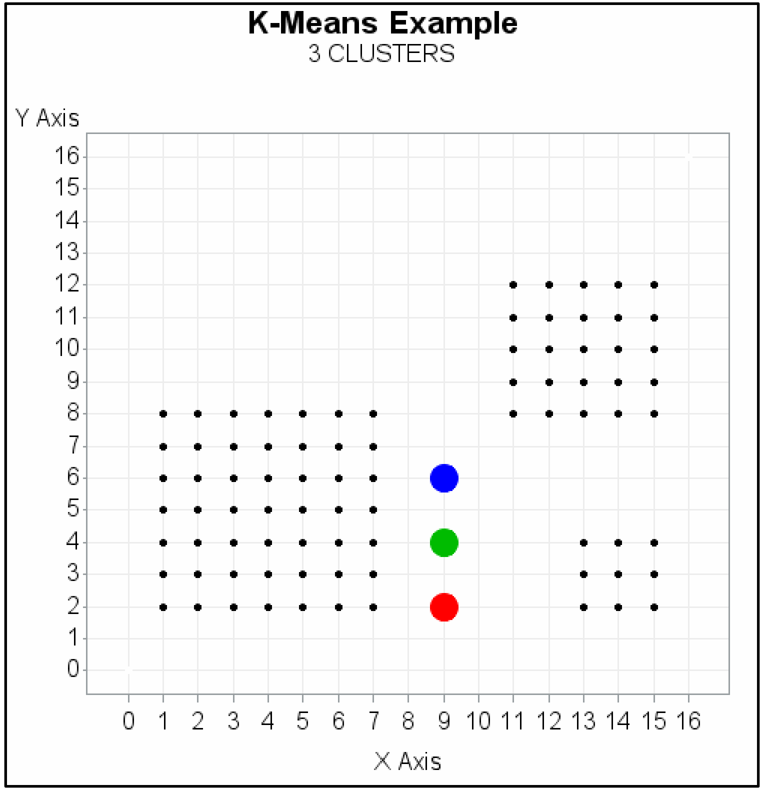
How Many Clusters: Example

3 Clusters



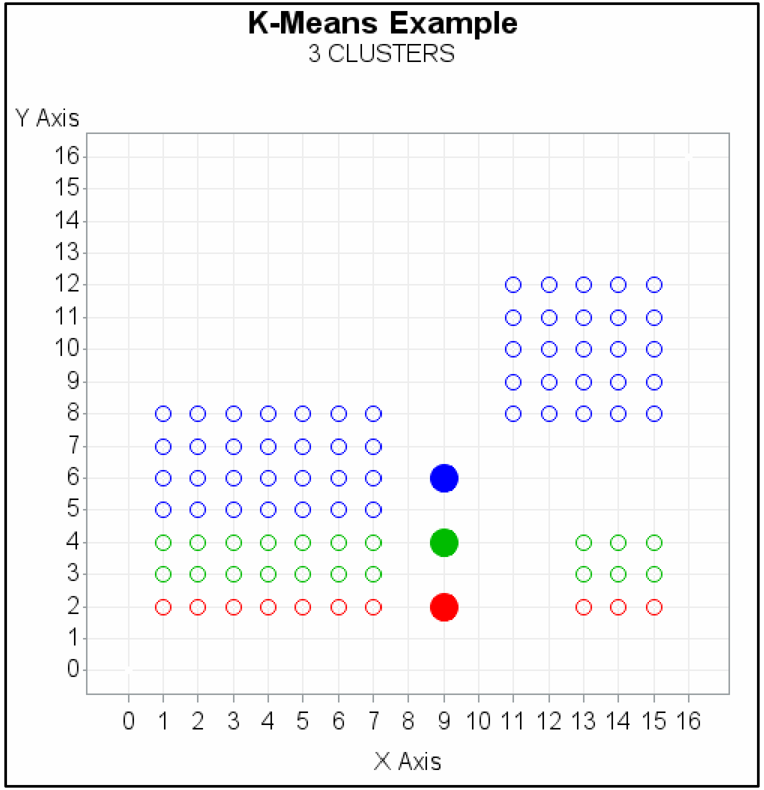
How Many Clusters: Example

3 Clusters



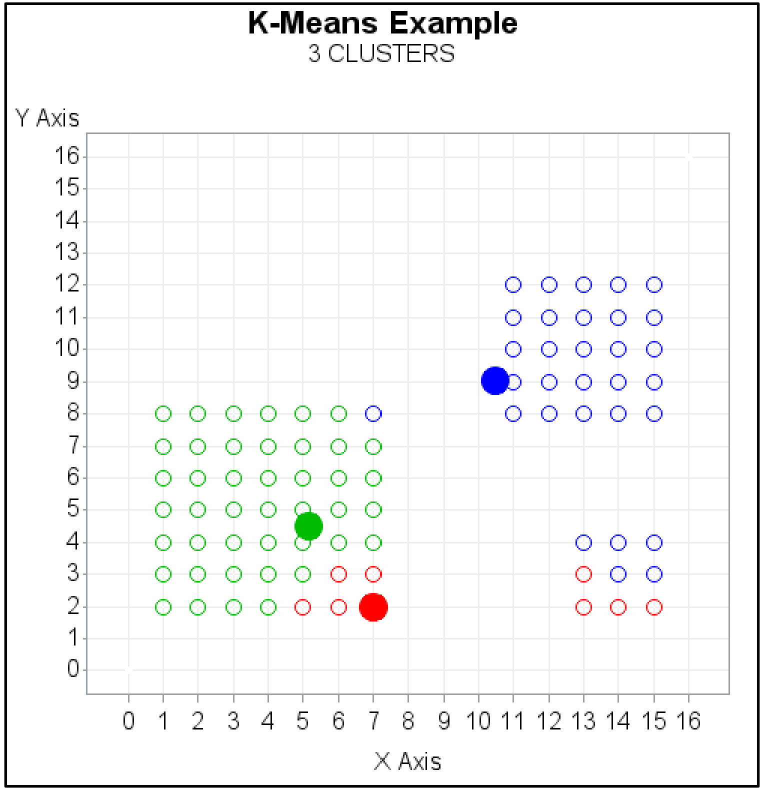
How Many Clusters: Example

3 Clusters



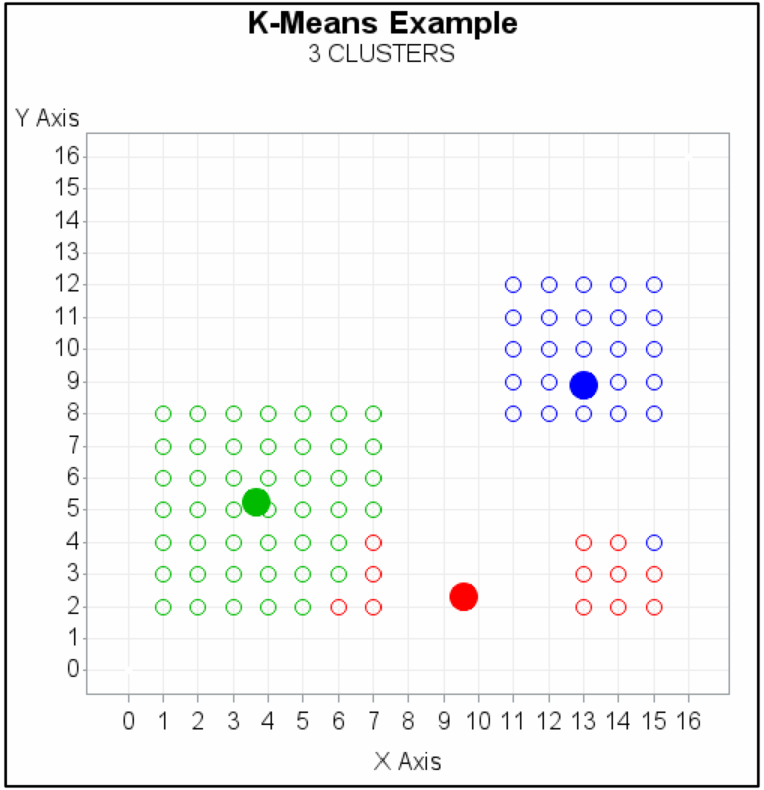
How Many Clusters: Example

3 Clusters



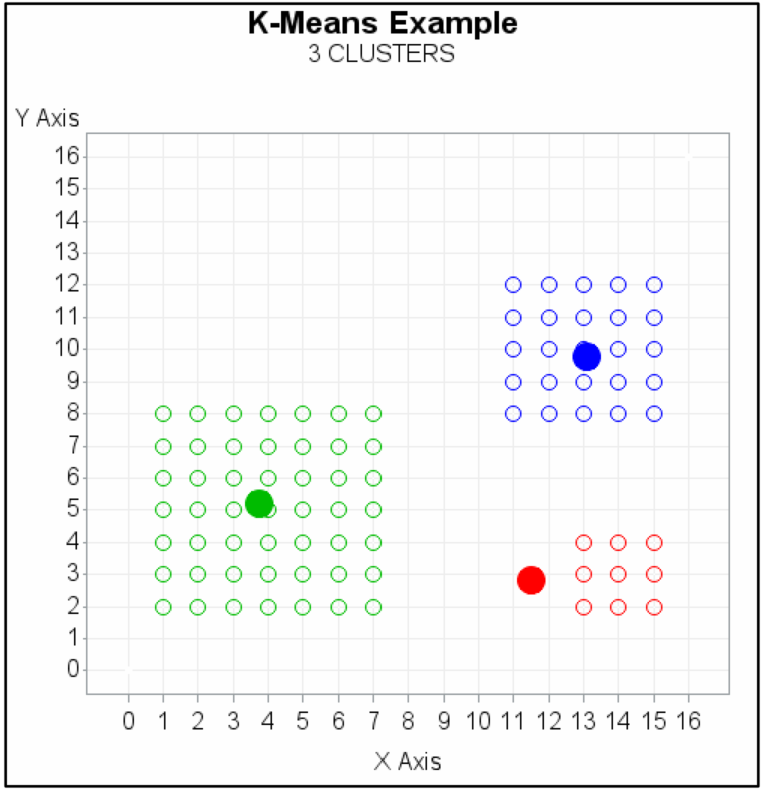
How Many Clusters: Example

3 Clusters



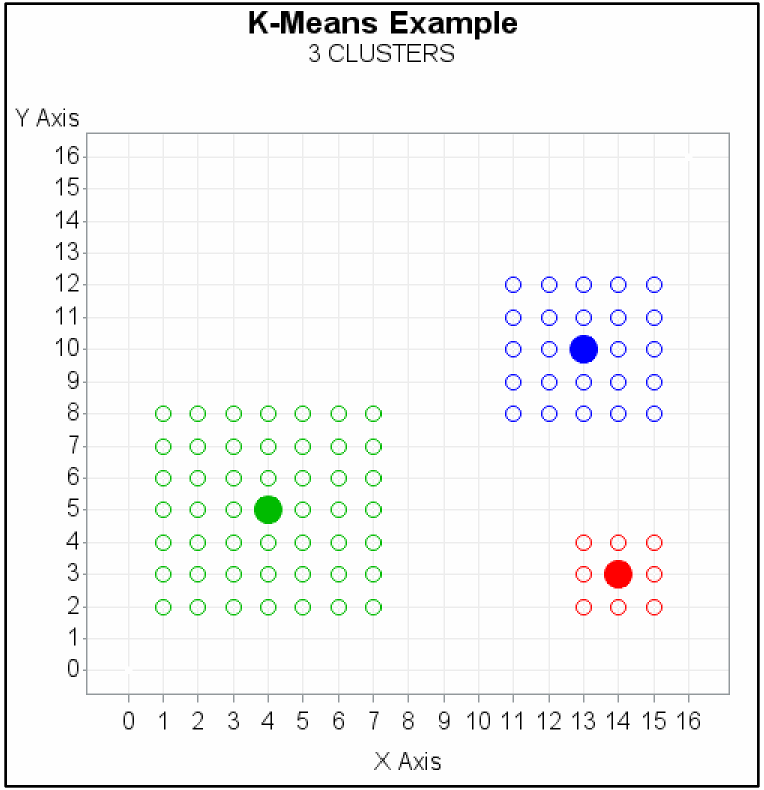
How Many Clusters: Example

3 Clusters



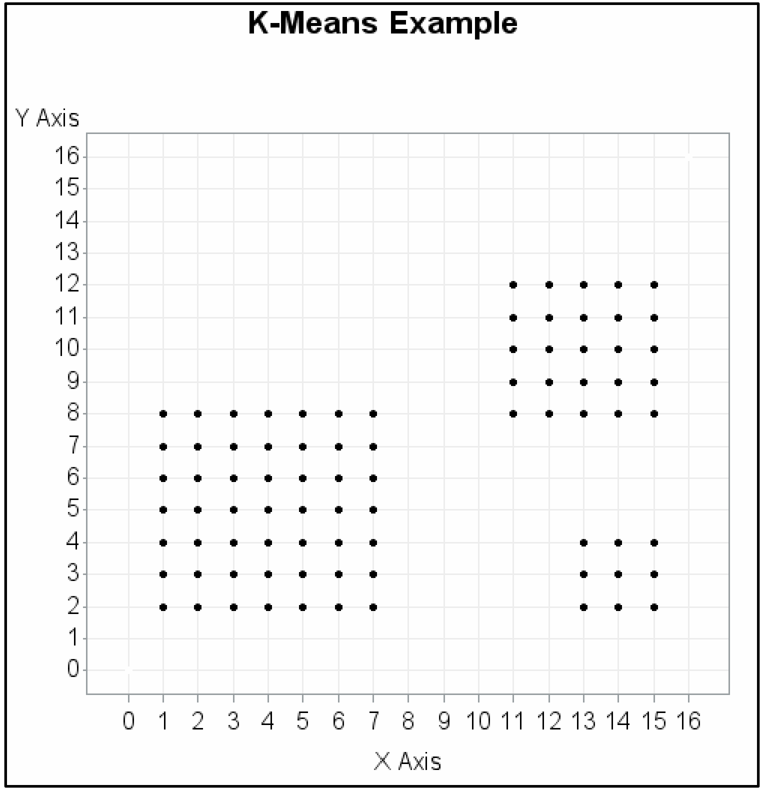
How Many Clusters: Example

3 Clusters



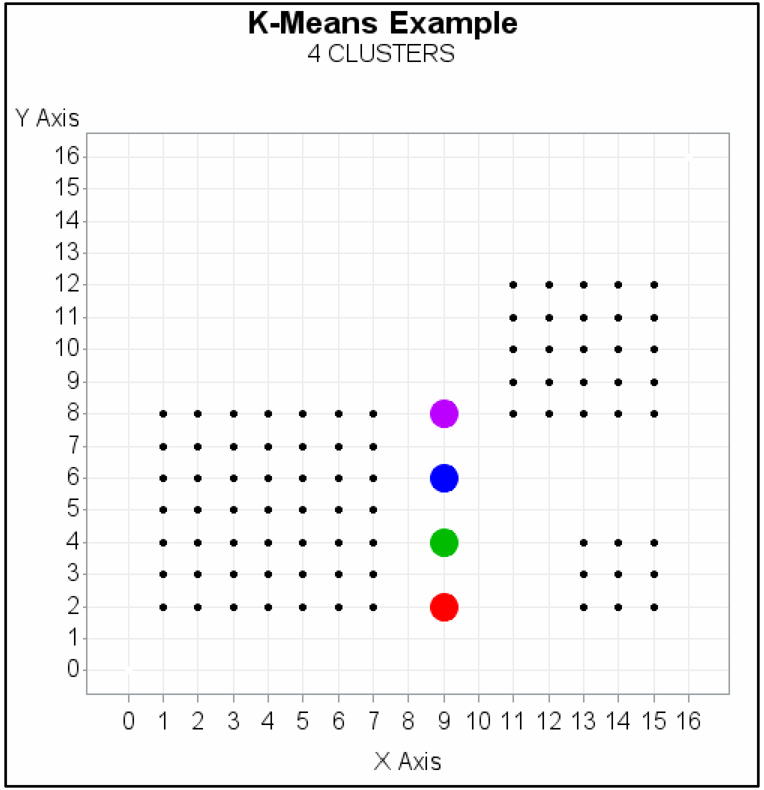
How Many Clusters: Example

4 Clusters



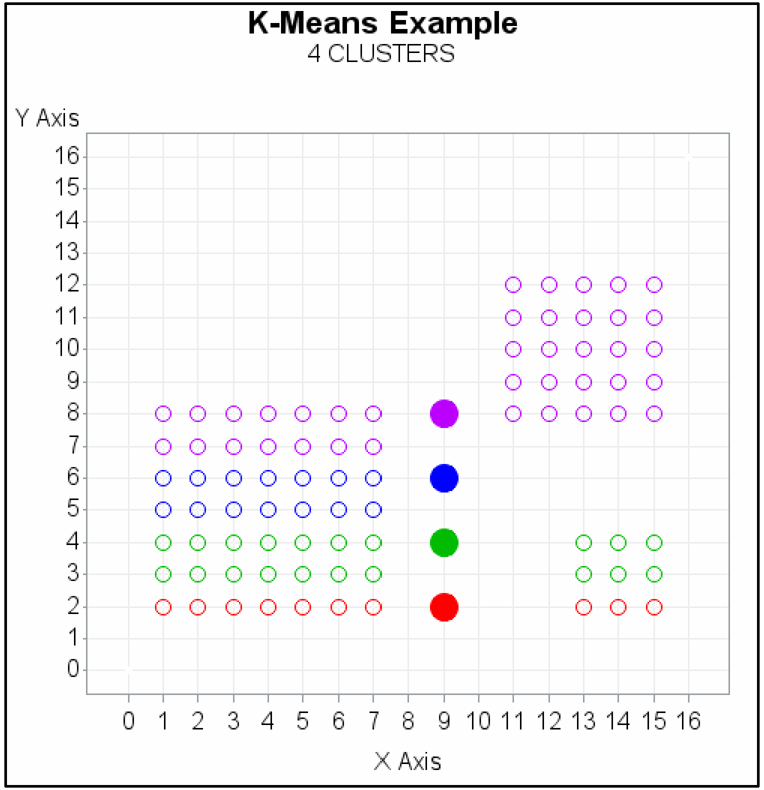
How Many Clusters: Example

4 Clusters



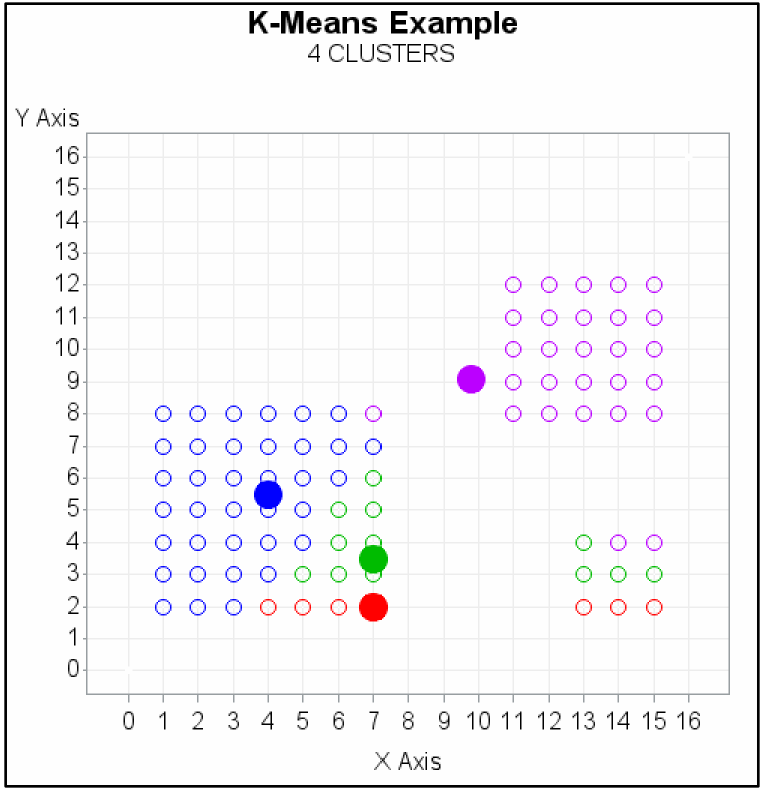
How Many Clusters: Example

4 Clusters



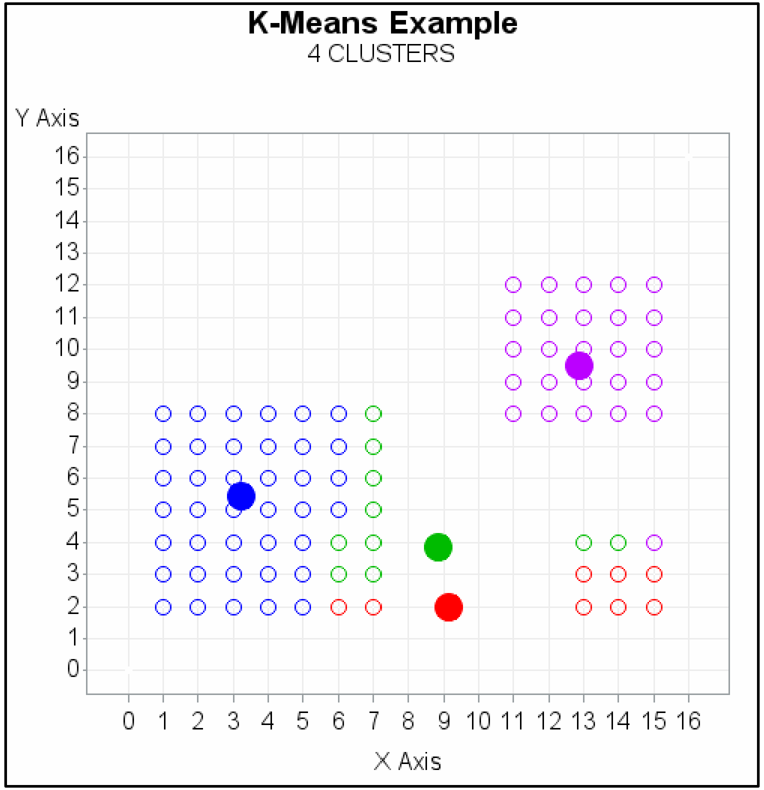
How Many Clusters: Example

4 Clusters



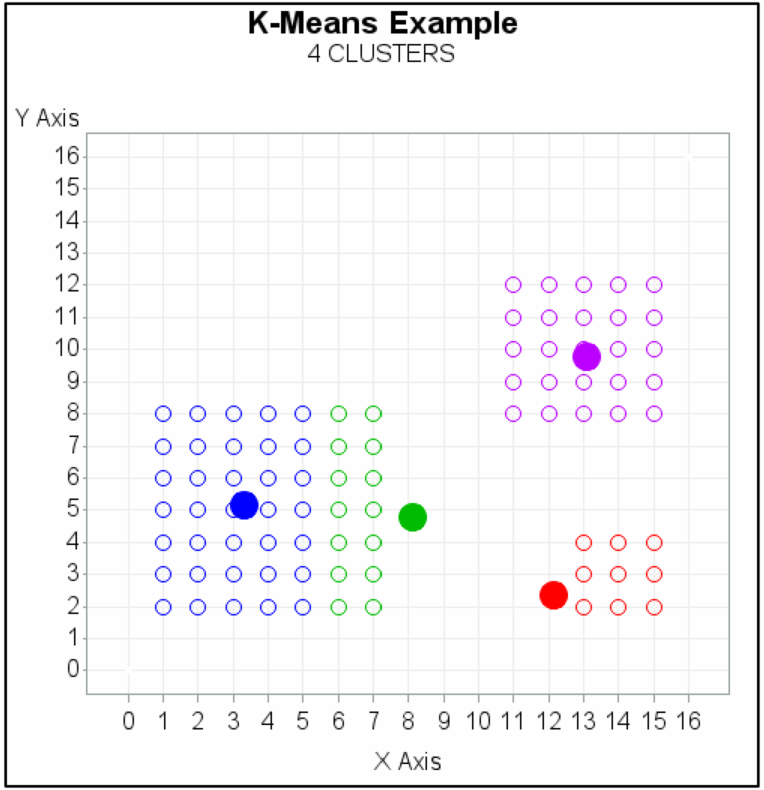
How Many Clusters: Example

4 Clusters



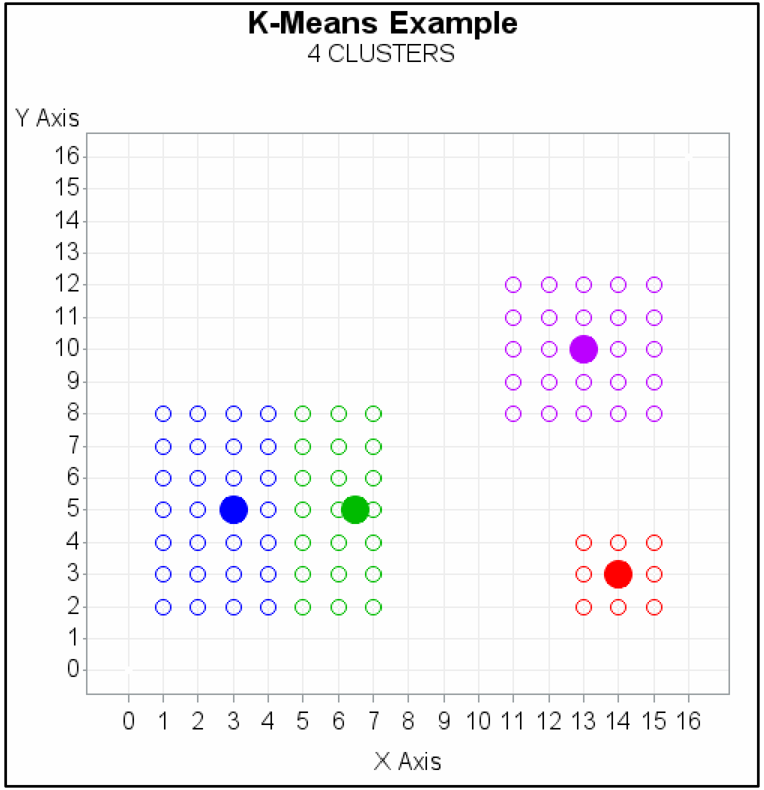
How Many Clusters: Example

4 Clusters



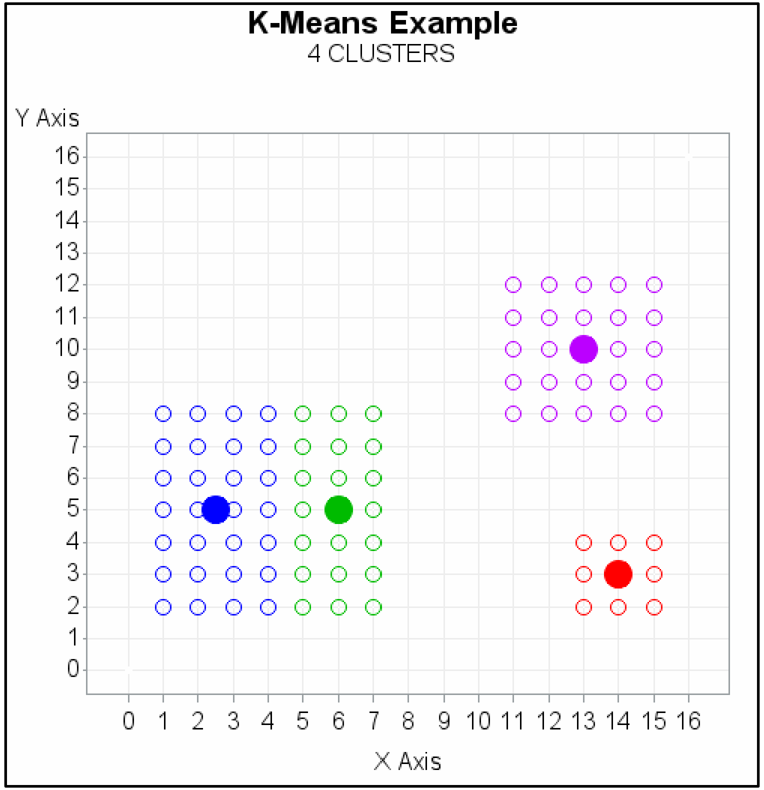
How Many Clusters: Example

4 Clusters



How Many Clusters: Example

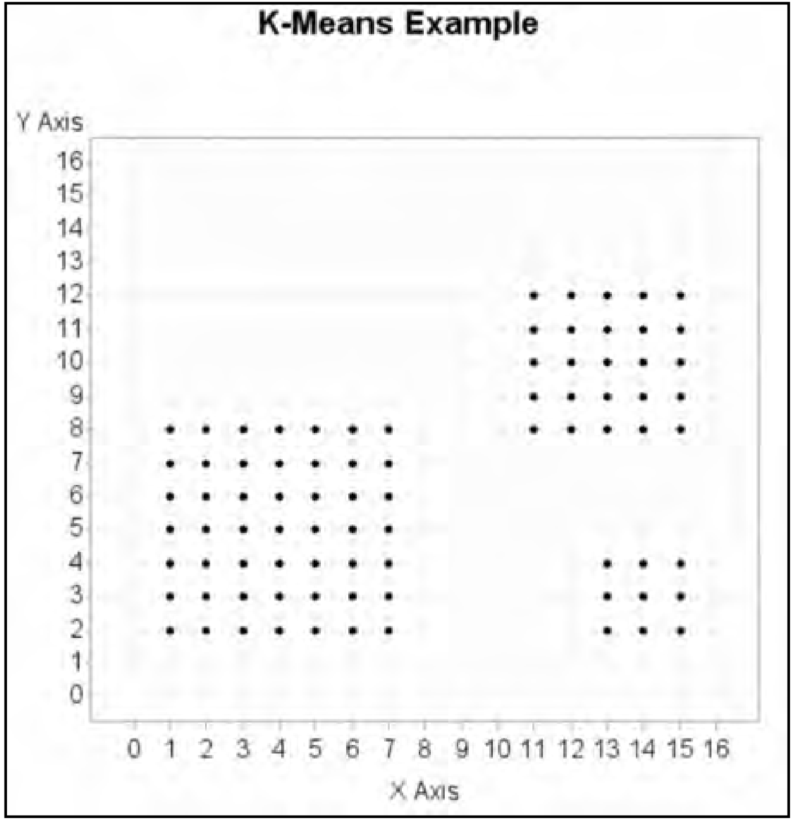
4 Clusters



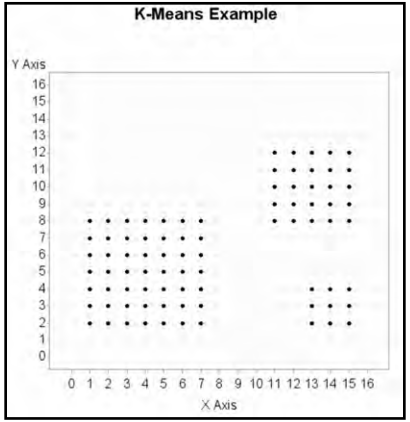
Summary

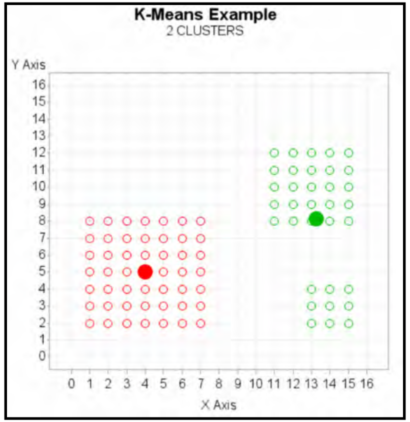
Given the Following Data Points

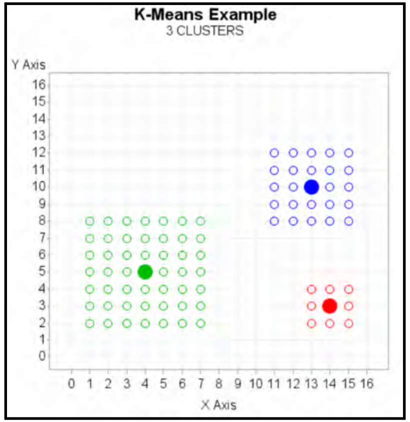
- Find the cluster centers for N=2 Clusters
- Find the cluster centers for N=3 Clusters
- Find the cluster centers for N=4 Clusters

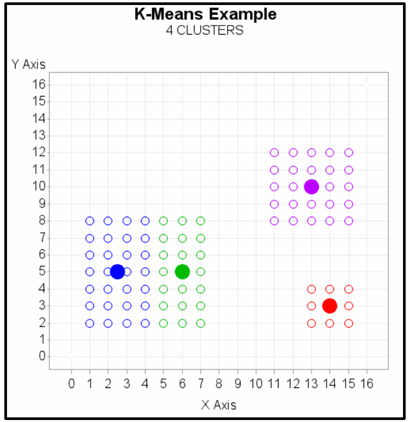


Summary


K-Means Example
Y Axis
X Axis


K-Means Example
2 CLUSTERS
Y Axis
X Axis

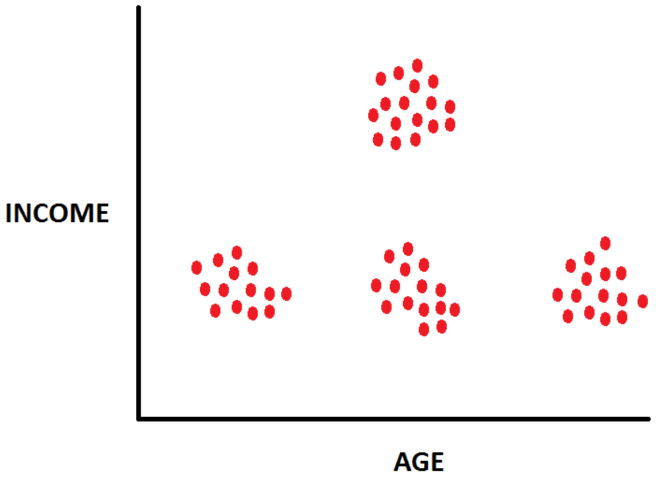

K-Means Example
3 CLUSTERS
Y Axis
X Axis

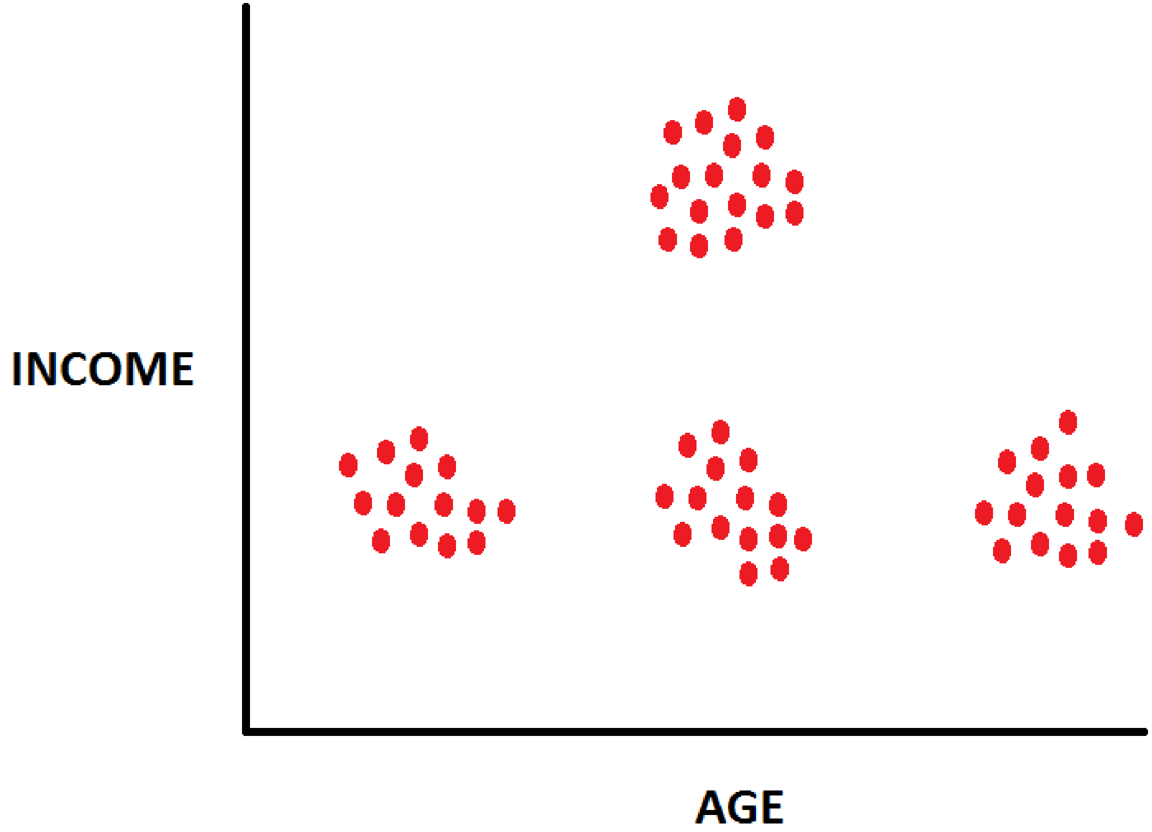

K-Means Example
4 CLUSTERS
Y Axis
X Axis

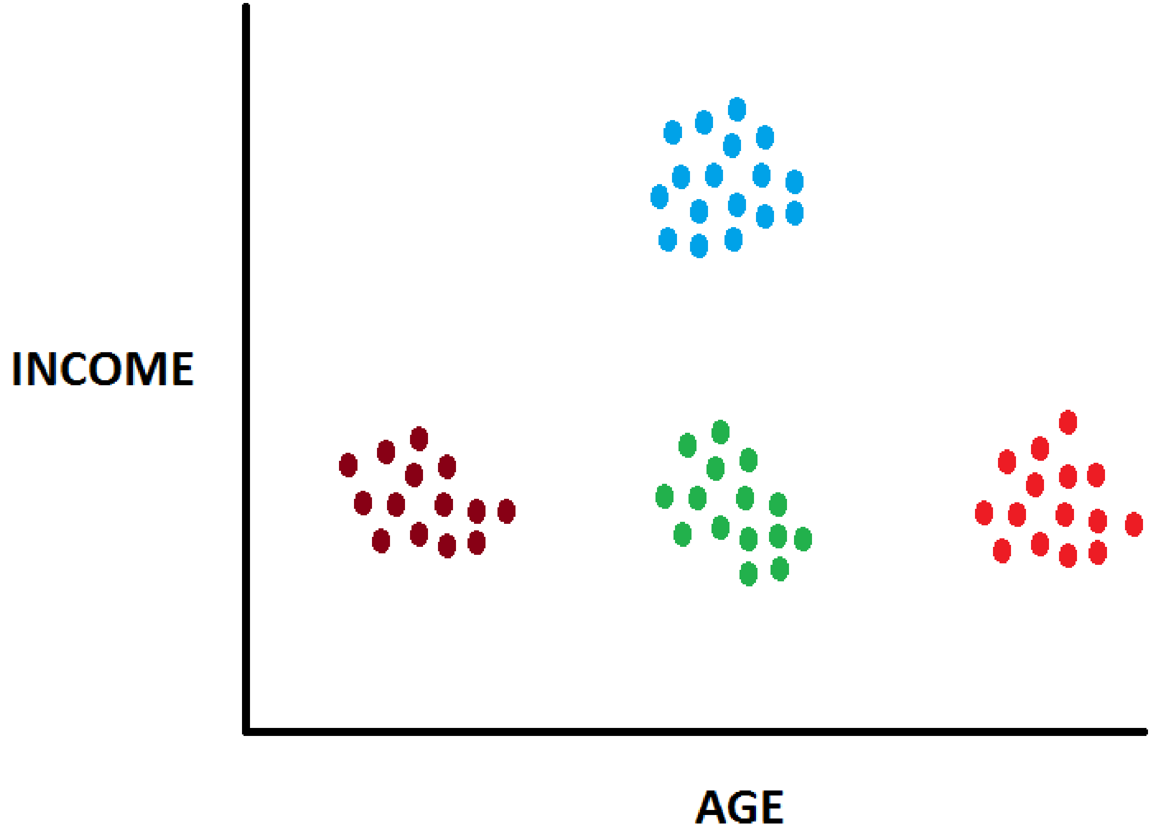
K-Means Clustering

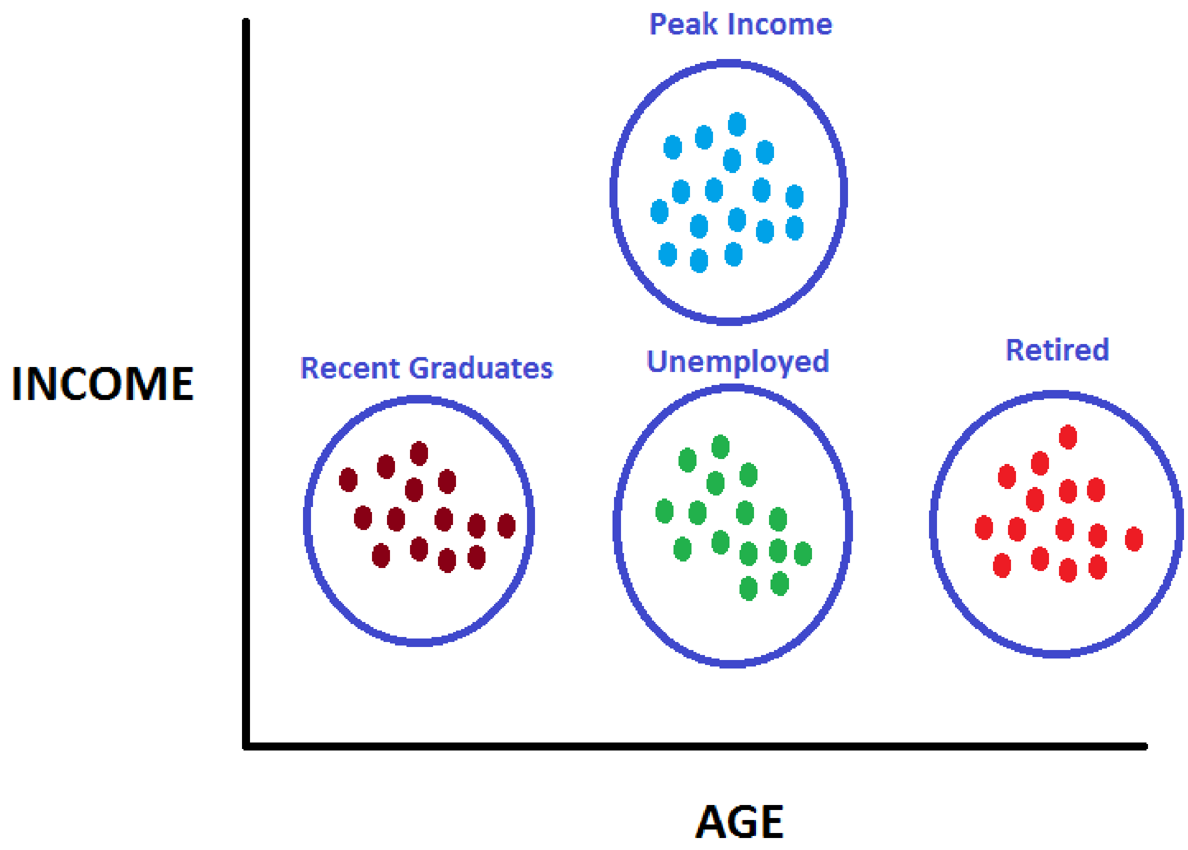
Clusters based on AGE and INCOME

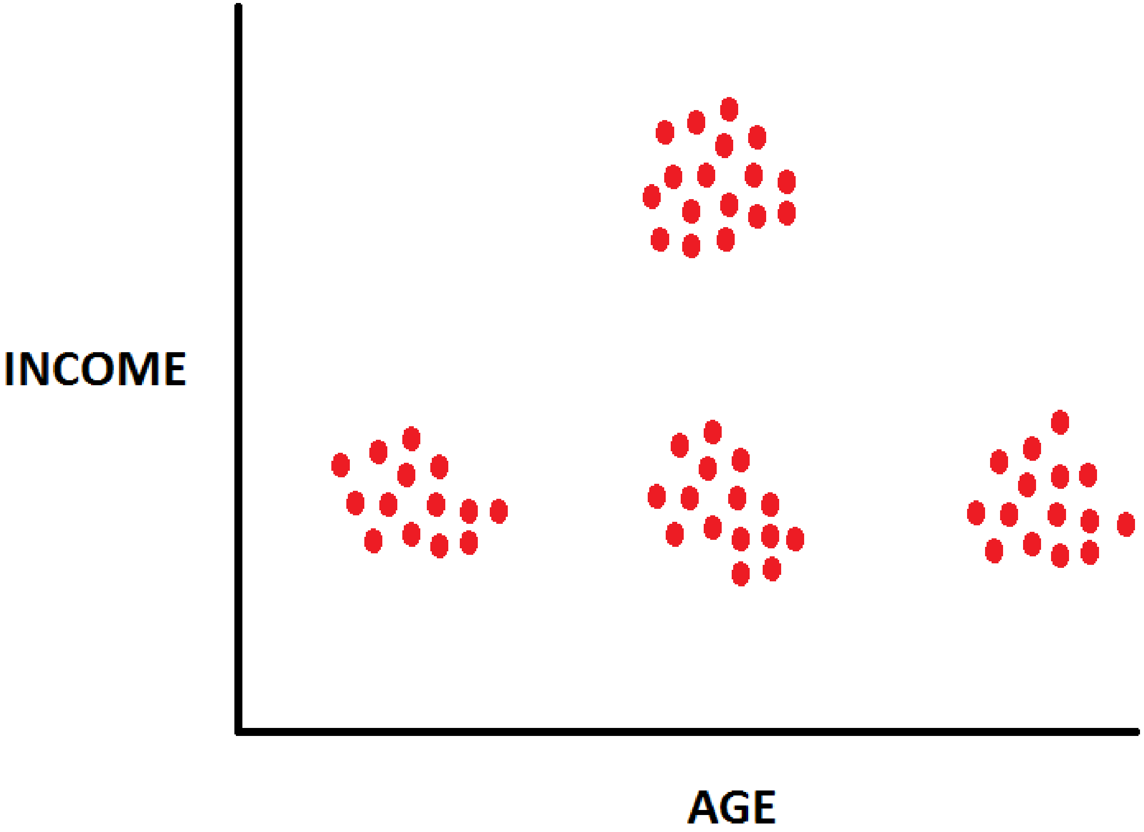
How many clusters do you see?

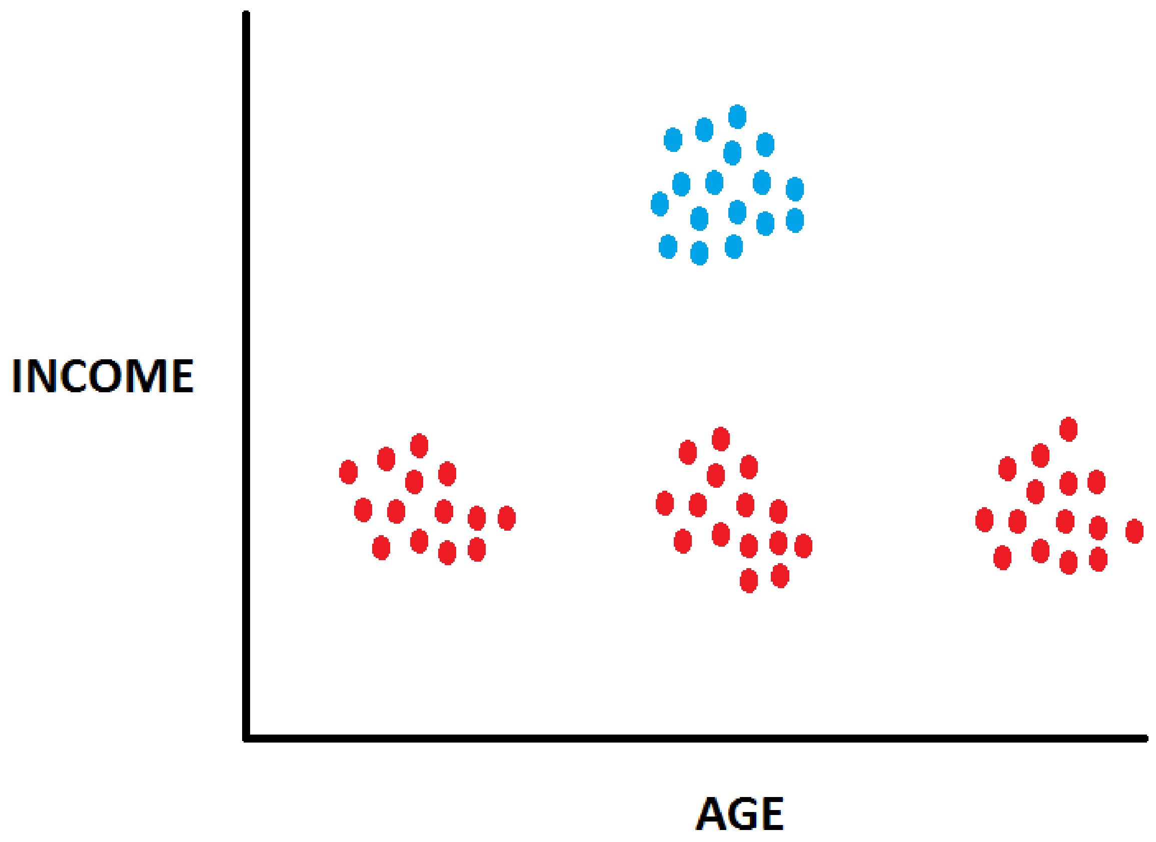


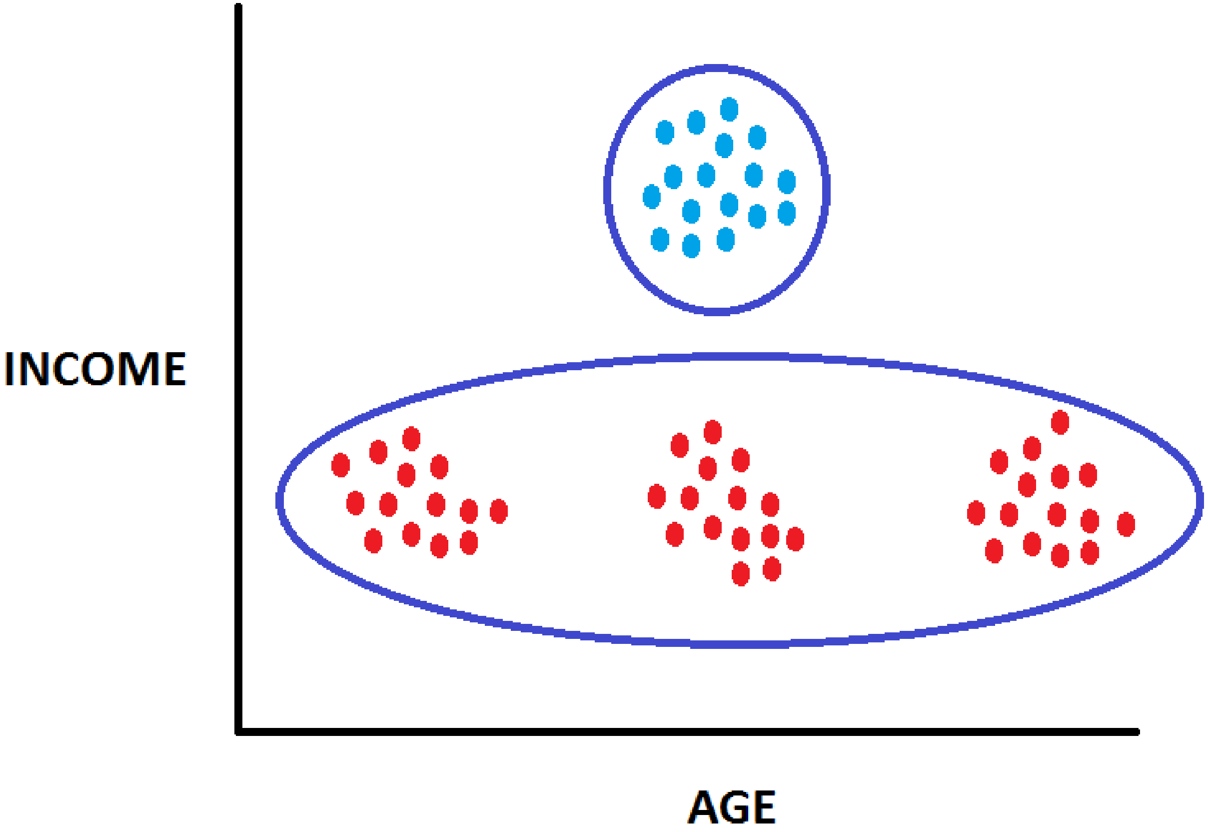

INCOME
AGE

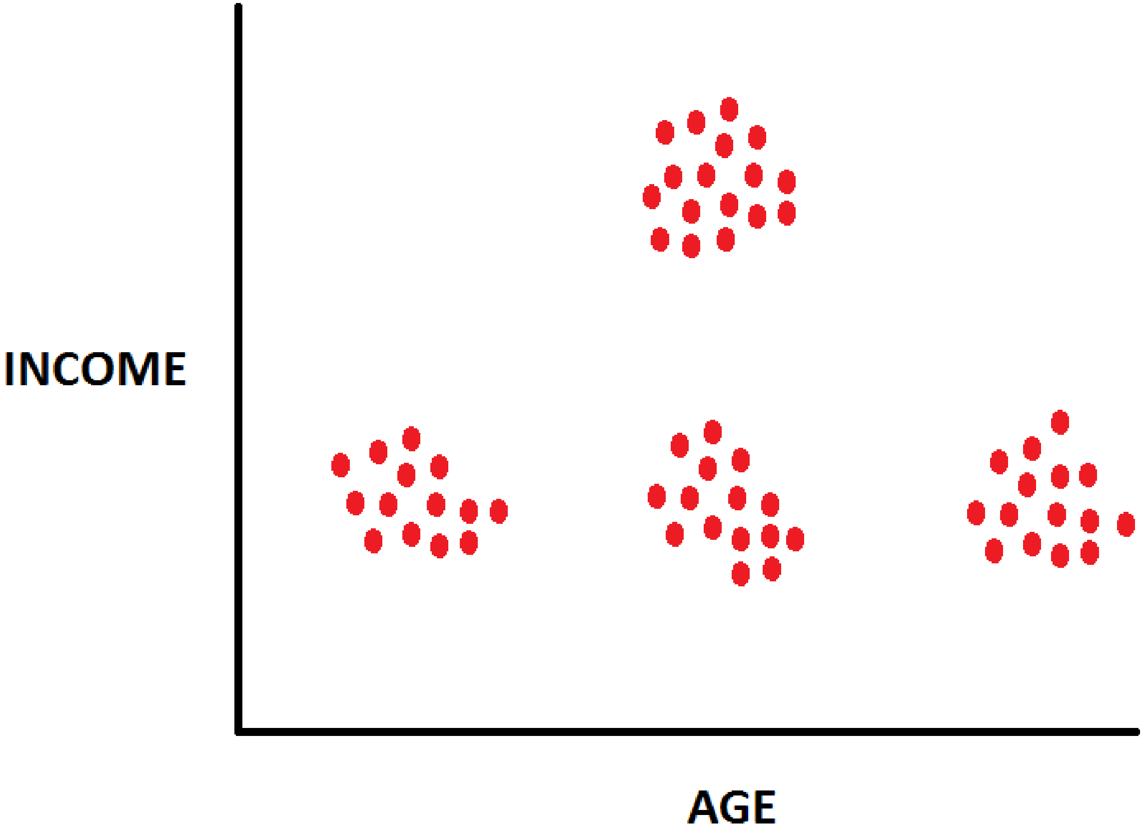

INCOME
AGE

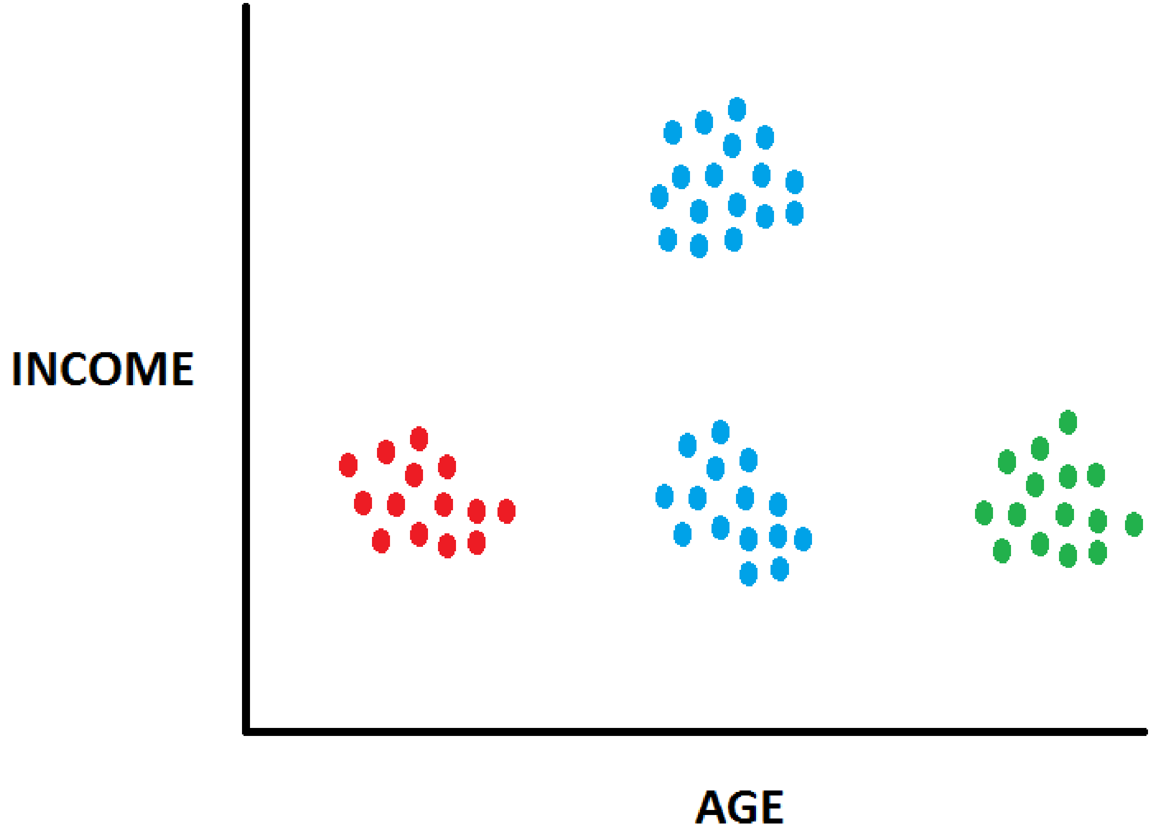

Peak Income
INCOME
Recent Graduates
Unemployed
Retired
AGE

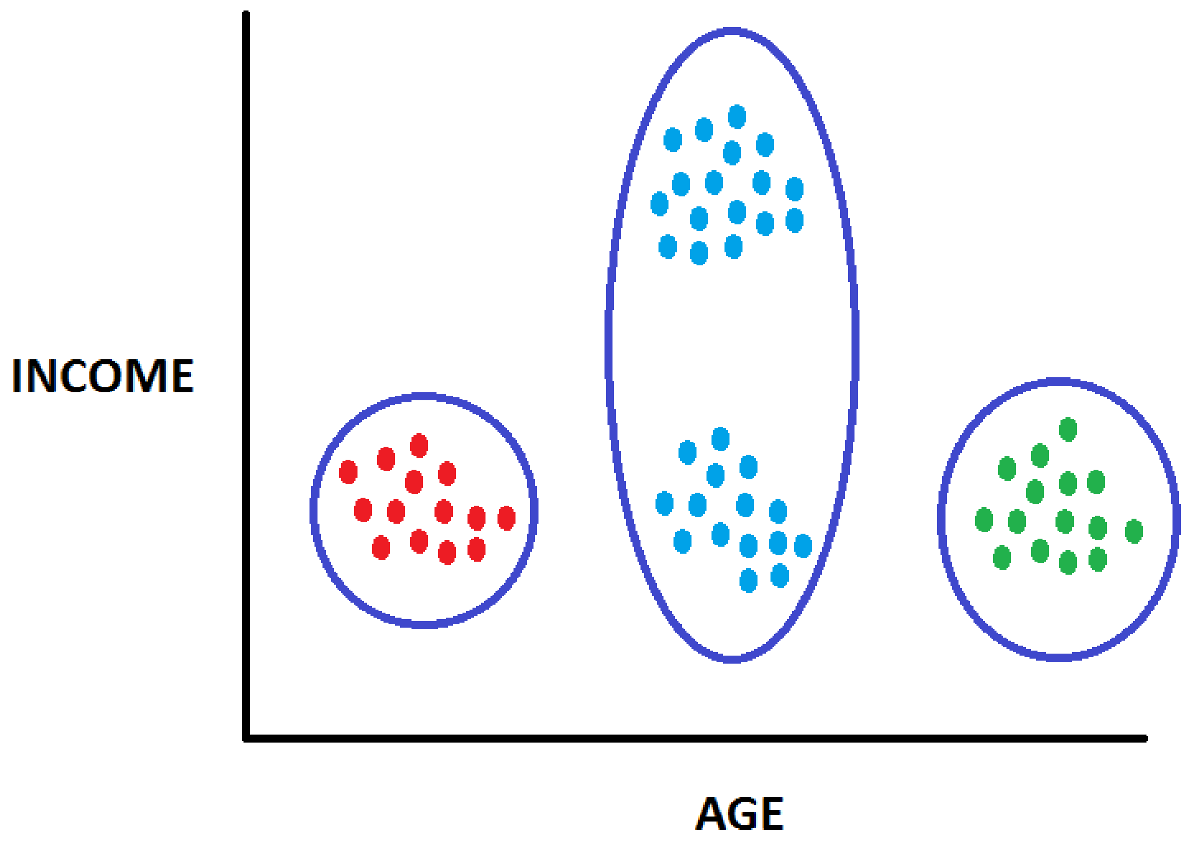

INCOME
AGE

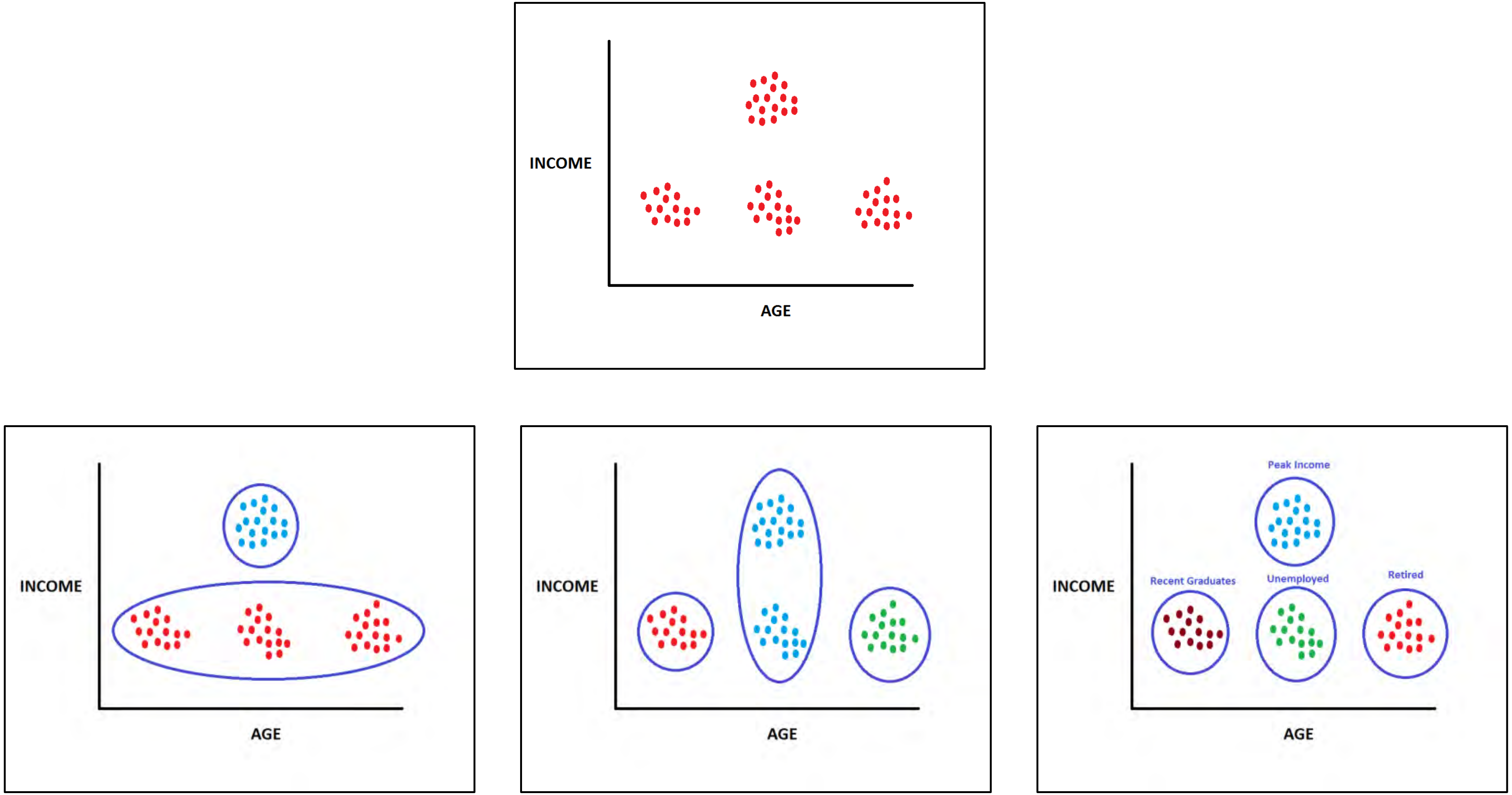

INCOME
AGE


INCOME
AGE


INCOME
AGE


INCOME
AGE


INCOME
AGE


INCOME
AGE
INCOME
AGE
INCOME
AGE
Peak Income
Recent Graduates
Unemployed
Retired
INCOME
AGE

What Affects Cluster Results?

- How many clusters are there?

- Cluster Starting Points (“Seeds”)?

What Affects Cluster Results?

- How many clusters are there?
- Cluster Starting Points (“Seeds”)?

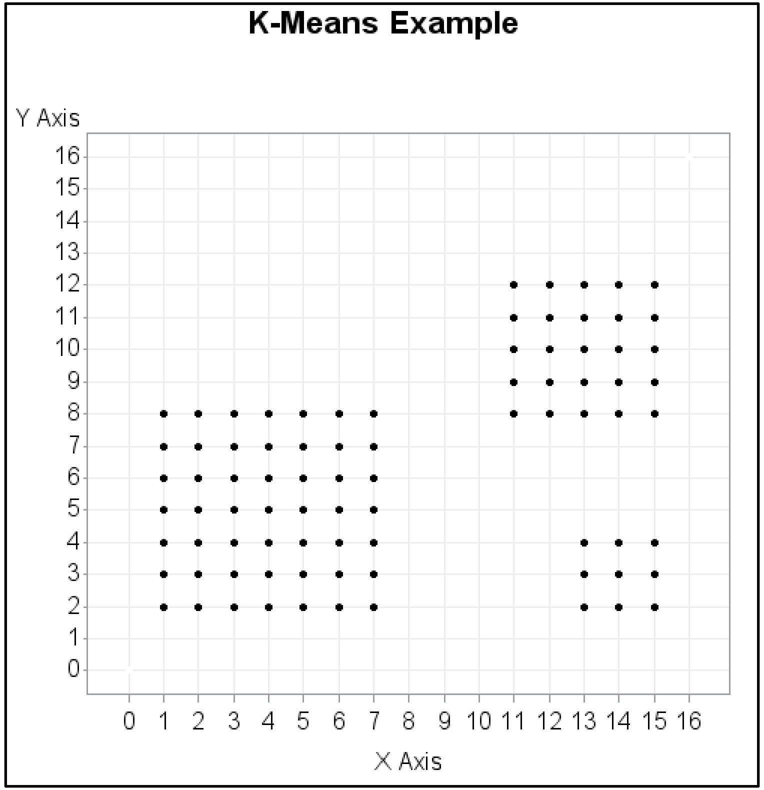
What Are The Center Points?

Cluster Seeds: Example

Given the Following Data Points

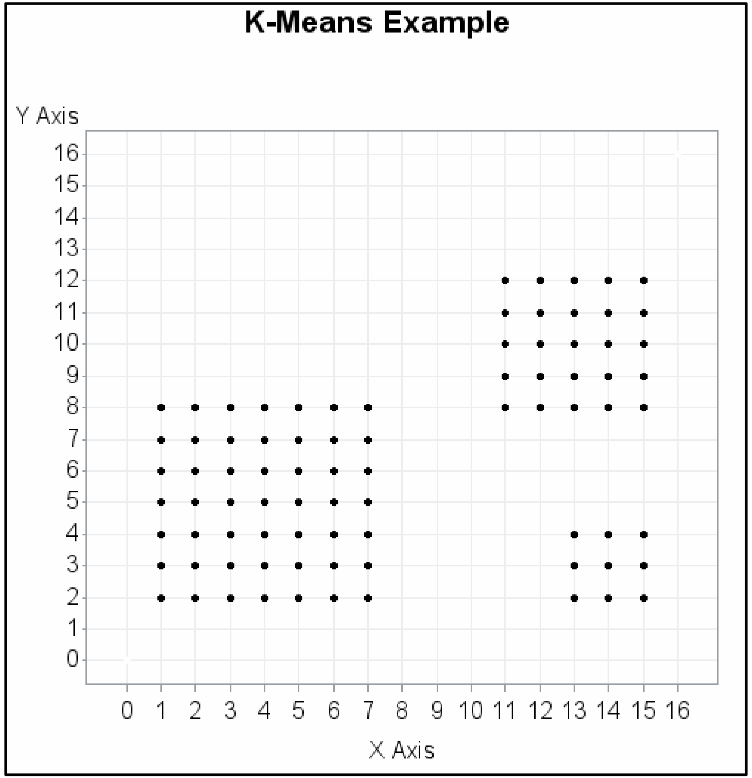
- Find the cluster centers for N=3 Clusters
- Find the cluster centers using Starting Point “A”
- Find the cluster centers using Starting Point “B”

Cluster Seeds: Example



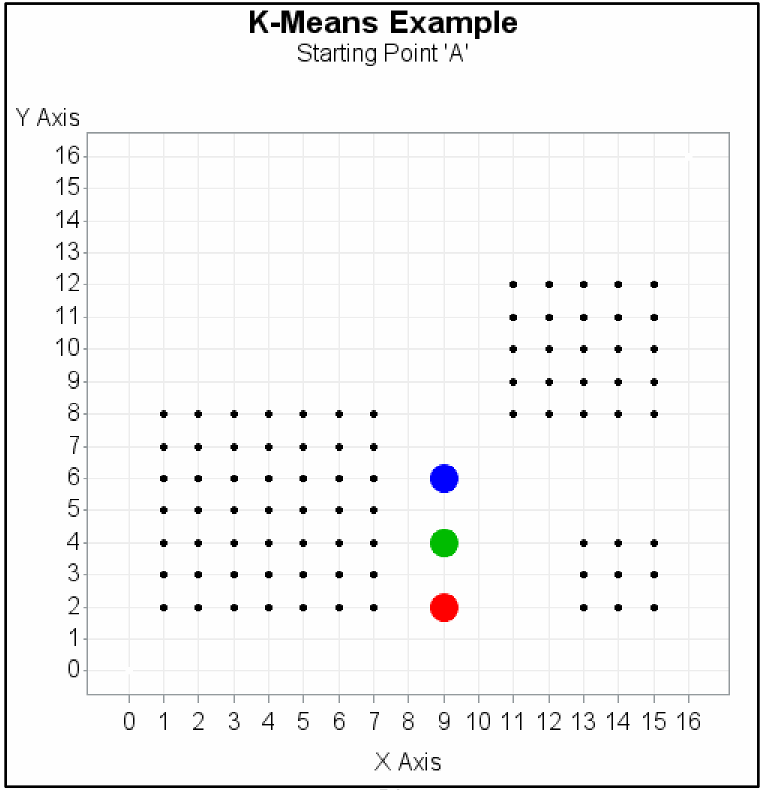
Cluster Starting Points “Seeds”

3 Clusters: Starting Point “A”



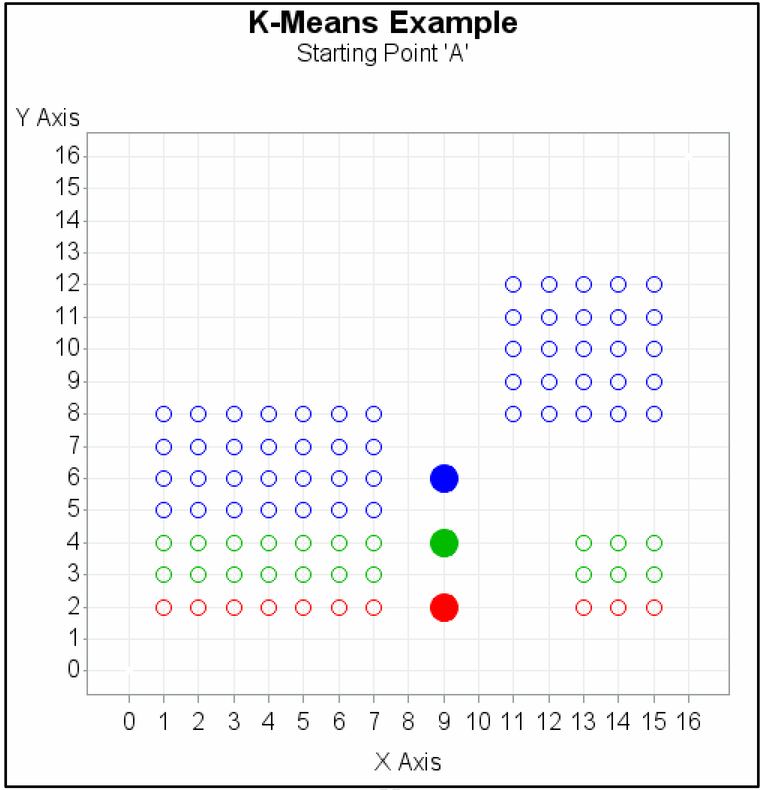
Cluster Starting Points “Seeds”

3 Clusters: Starting Point “A”



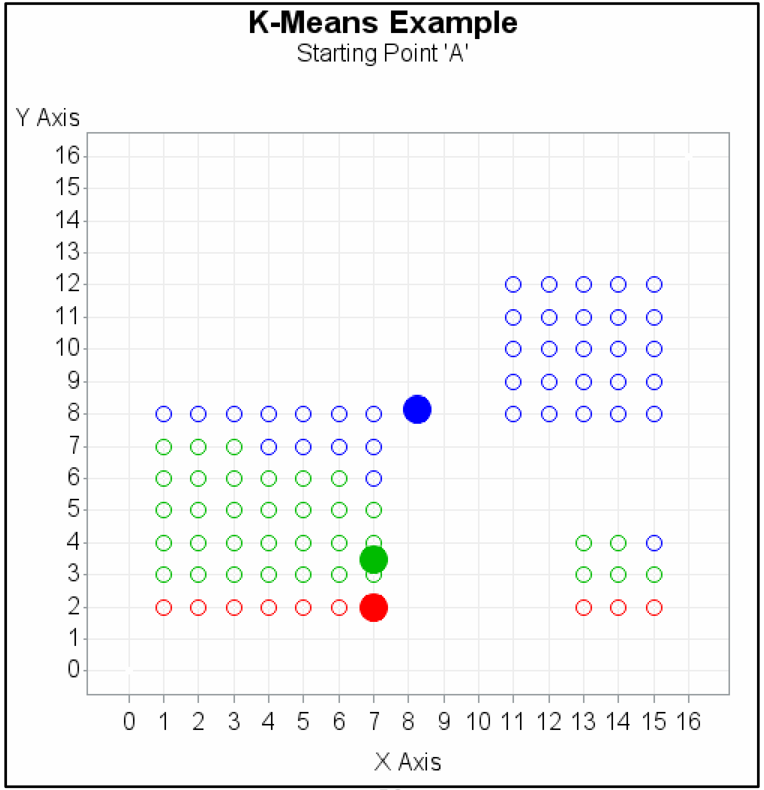
Cluster Starting Points "Seeds"

3 Clusters: Starting Point "A"



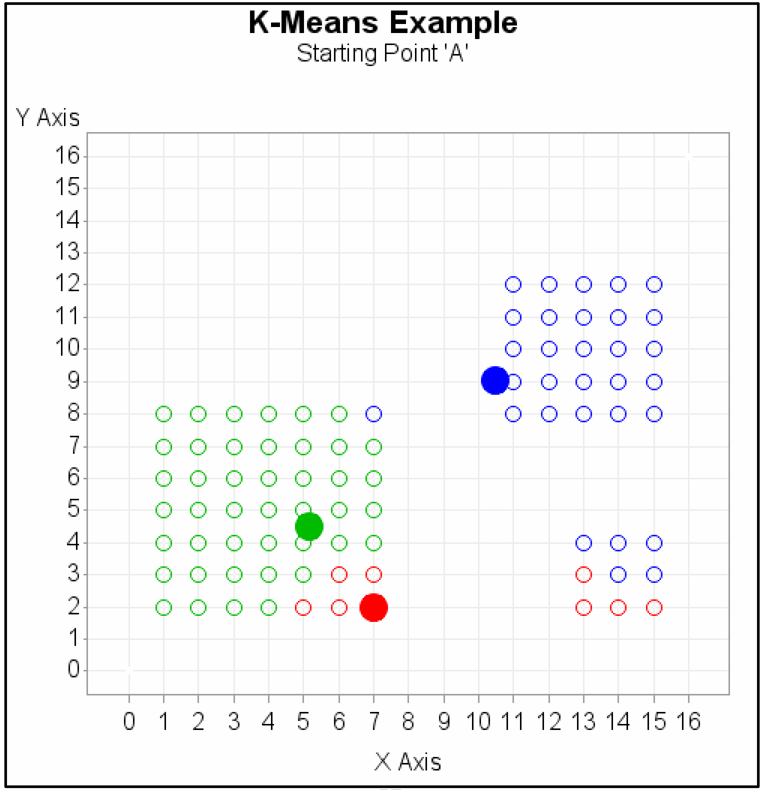
Cluster Starting Points "Seeds"

3 Clusters: Starting Point "A"



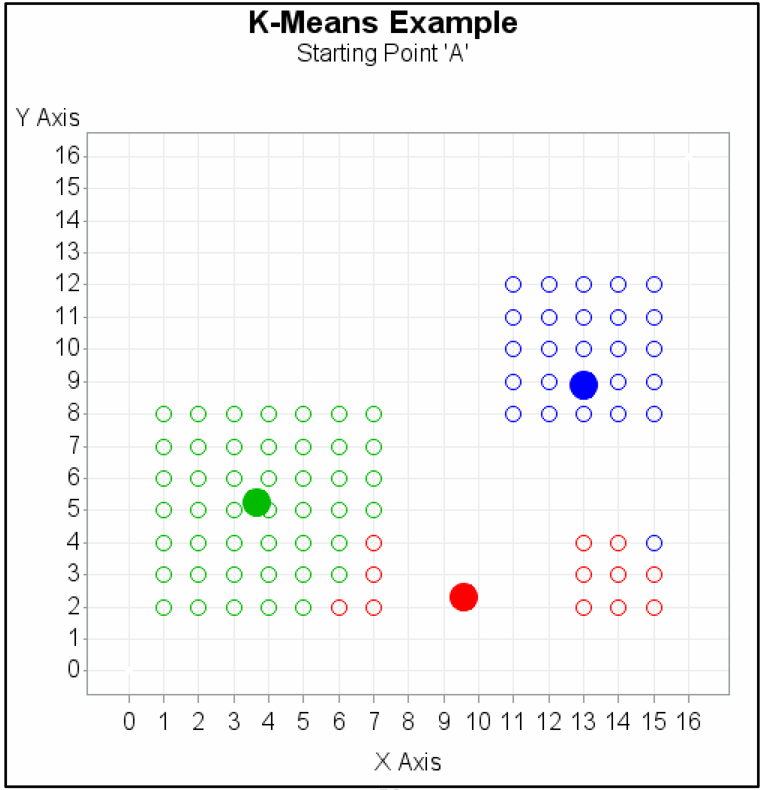
Cluster Starting Points "Seeds"

3 Clusters: Starting Point "A"



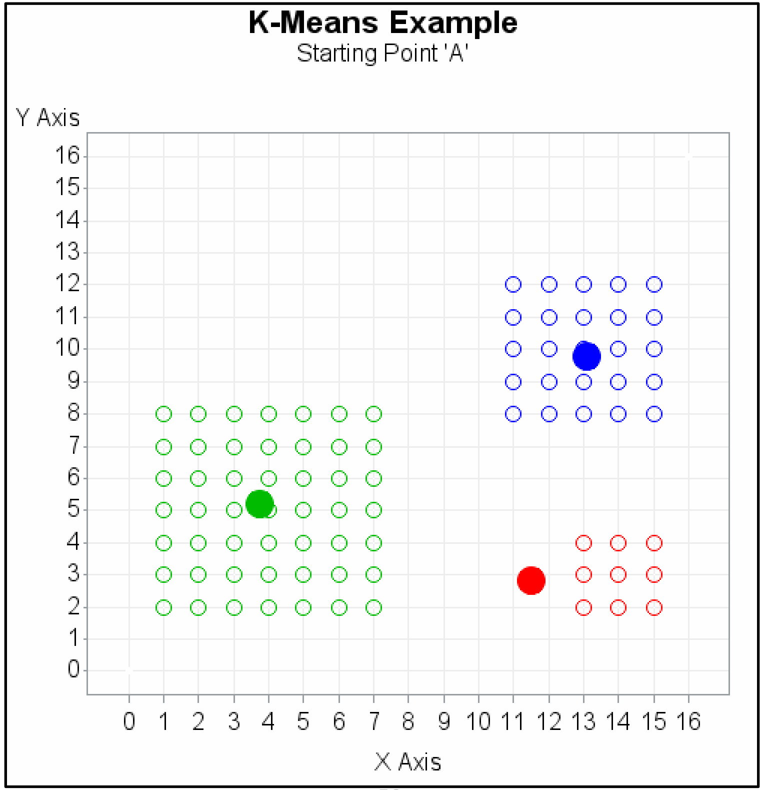
Cluster Starting Points "Seeds"

3 Clusters: Starting Point "A"



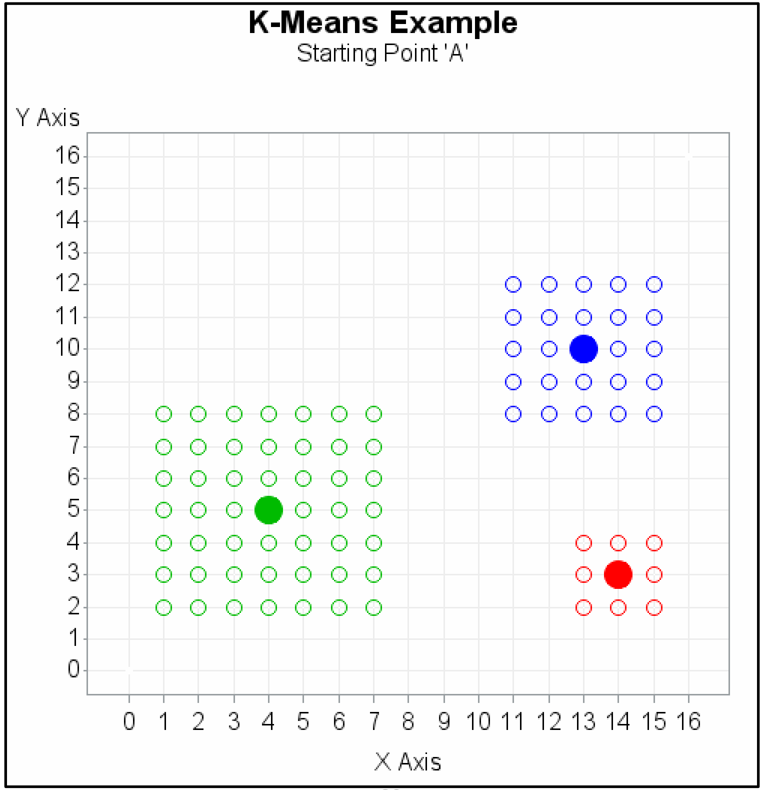
Cluster Starting Points "Seeds"

3 Clusters: Starting Point "A"



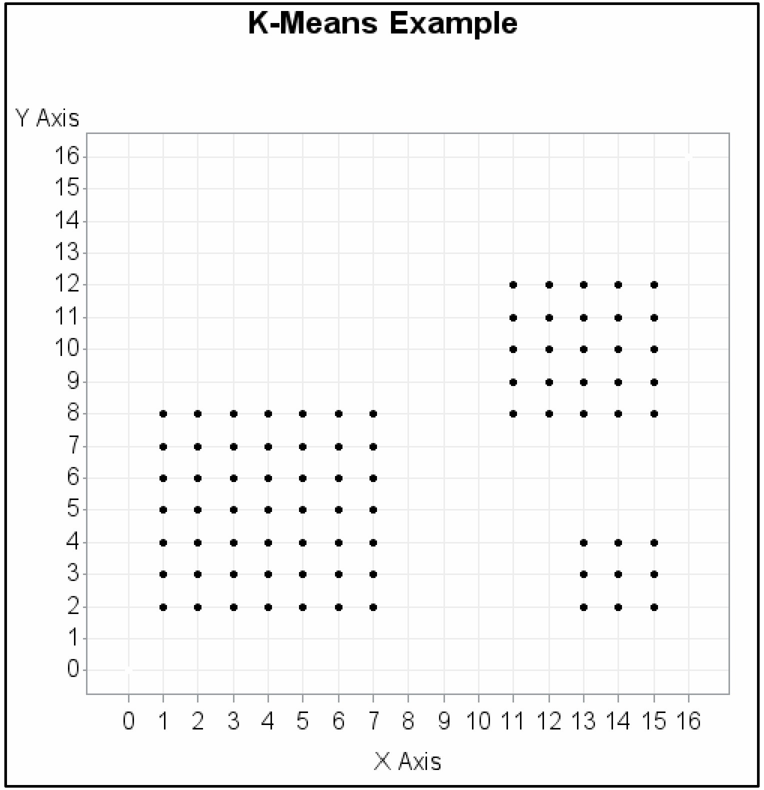
Cluster Starting Points "Seeds"

3 Clusters: Starting Point "A"



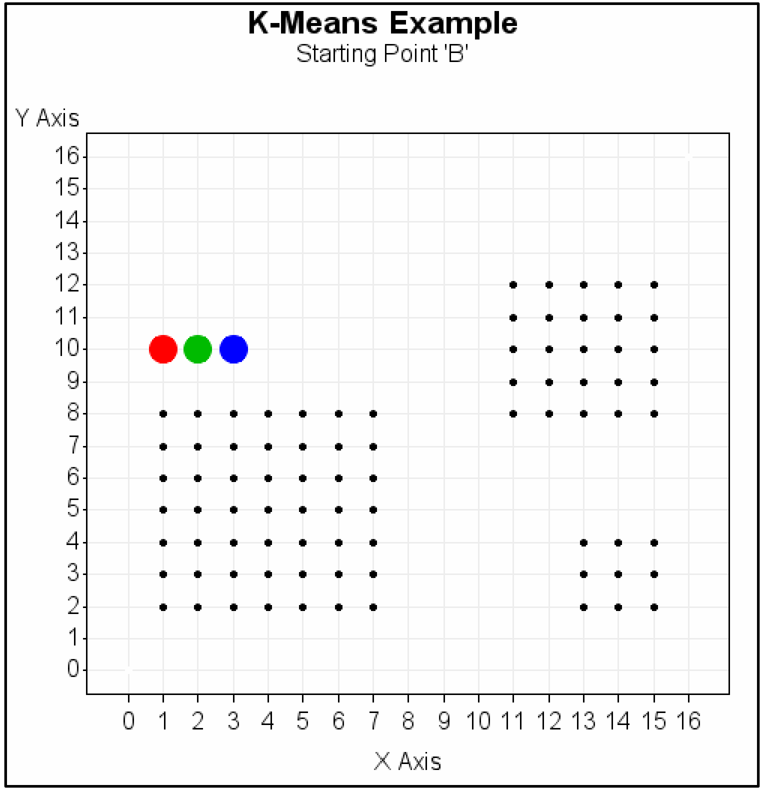
Cluster Starting Points "Seeds"

3 Clusters: Starting Point "B"



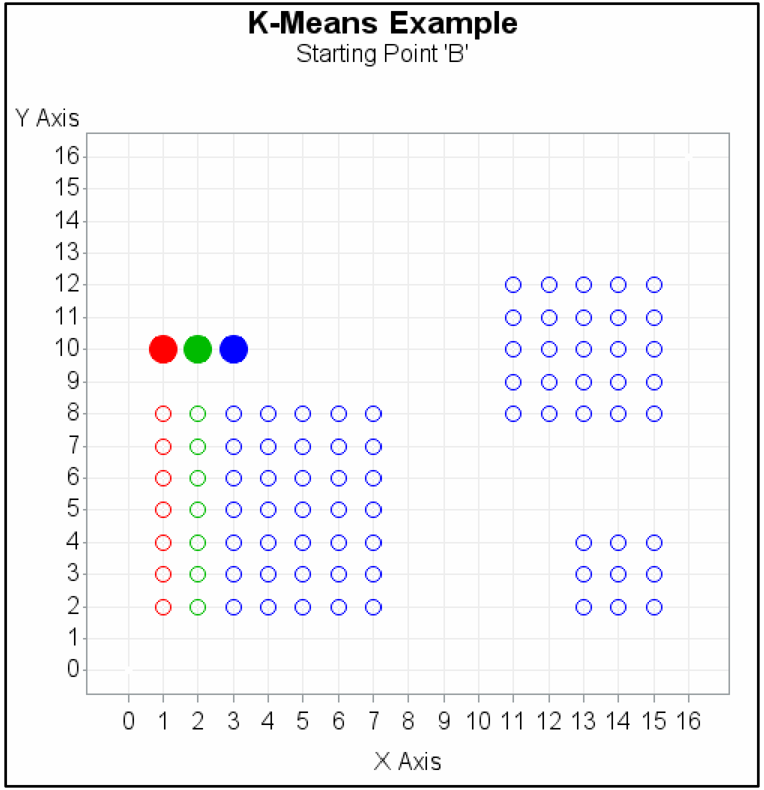
Cluster Starting Points "Seeds"

3 Clusters: Starting Point "B"



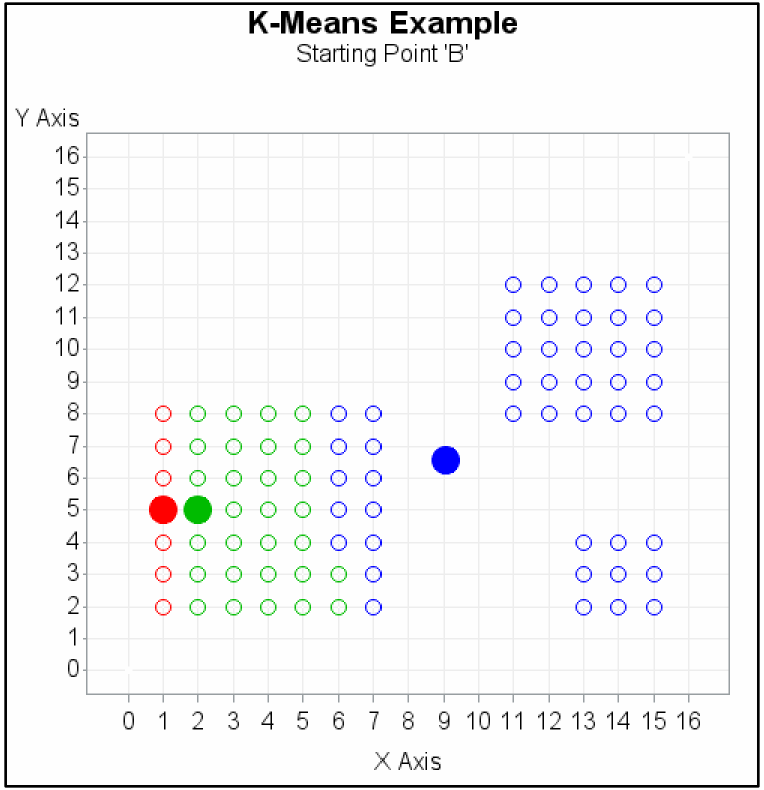
Cluster Starting Points “Seeds”

3 Clusters: Starting Point “B”



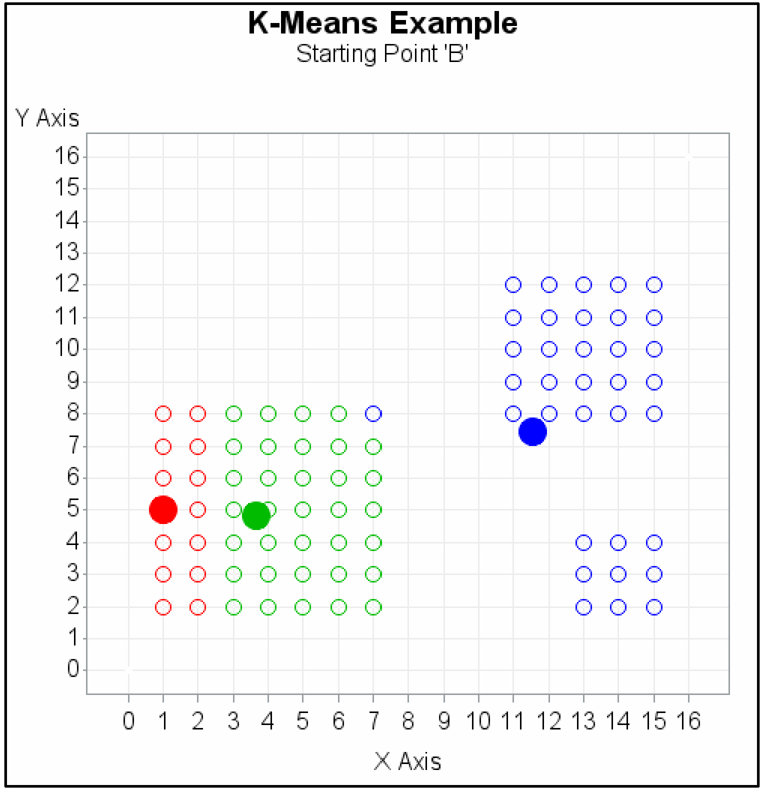
Cluster Starting Points "Seeds"

3 Clusters: Starting Point "B"



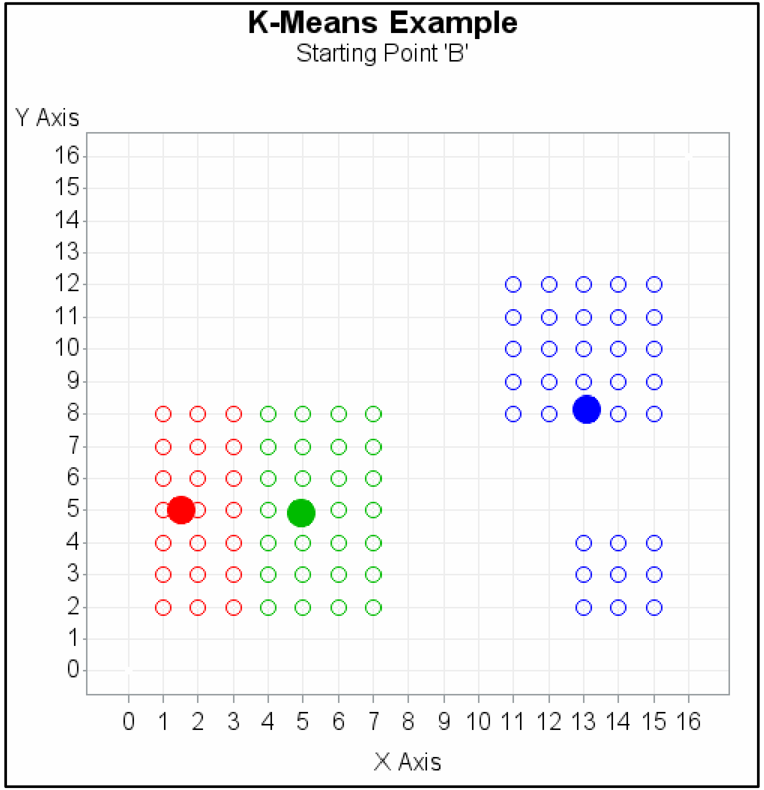
Cluster Starting Points "Seeds"

3 Clusters: Starting Point "B"



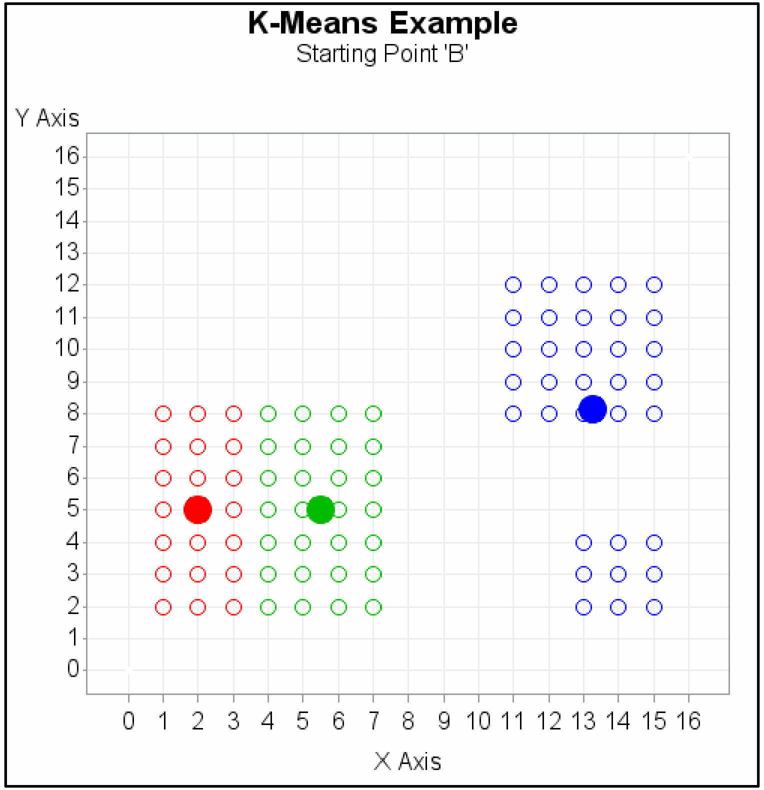
Cluster Starting Points “Seeds”

3 Clusters: Starting Point “B”

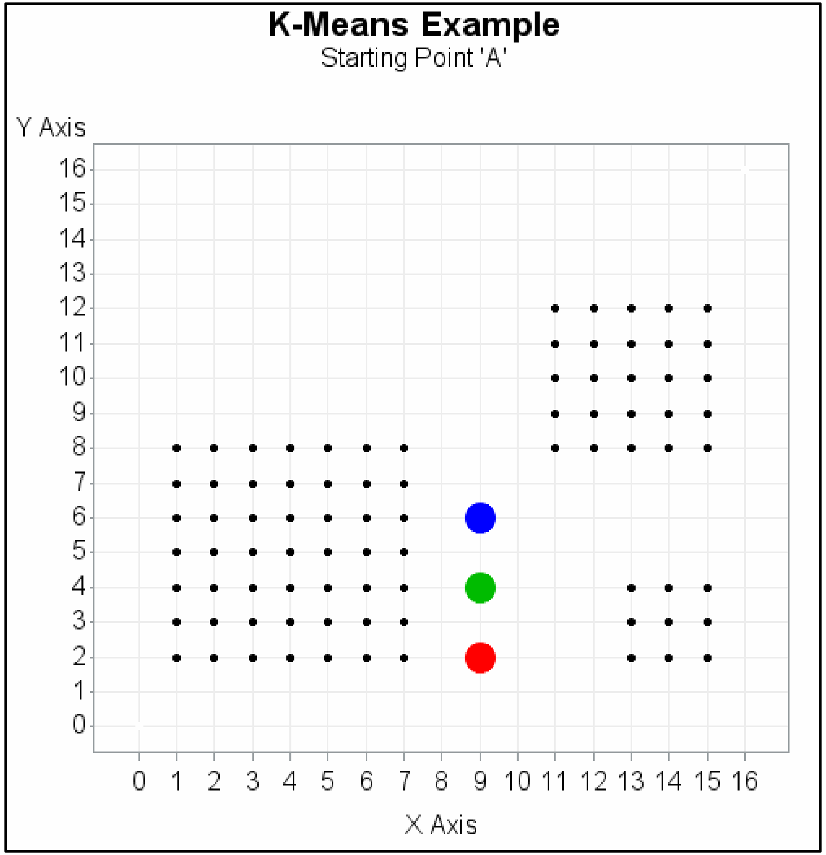


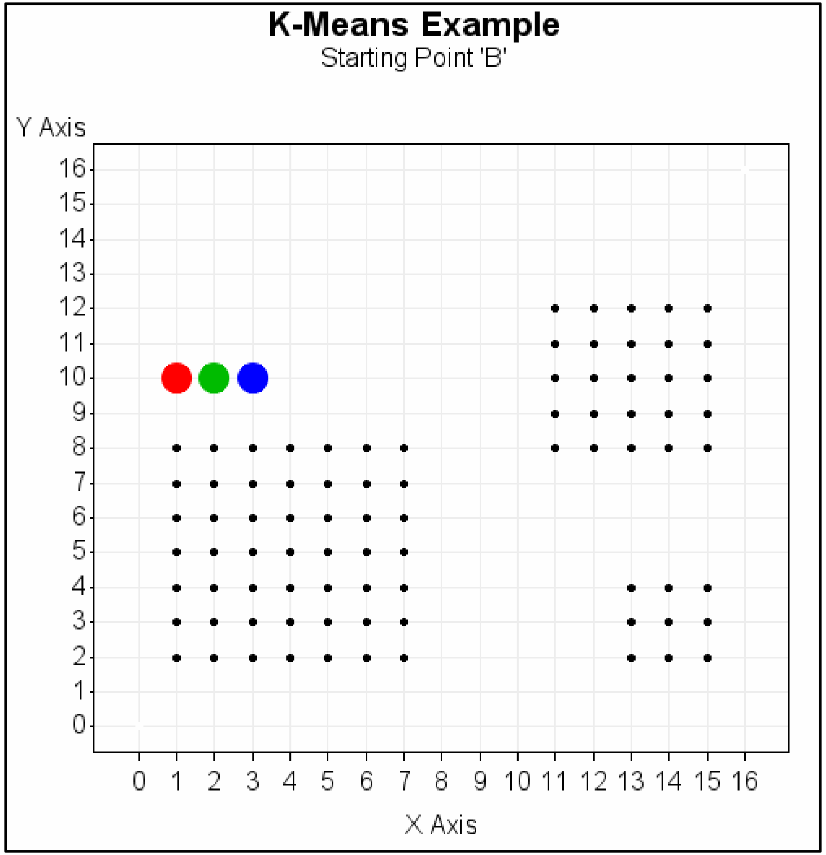
Cluster Starting Points "Seeds"

3 Clusters: Starting Point "B"

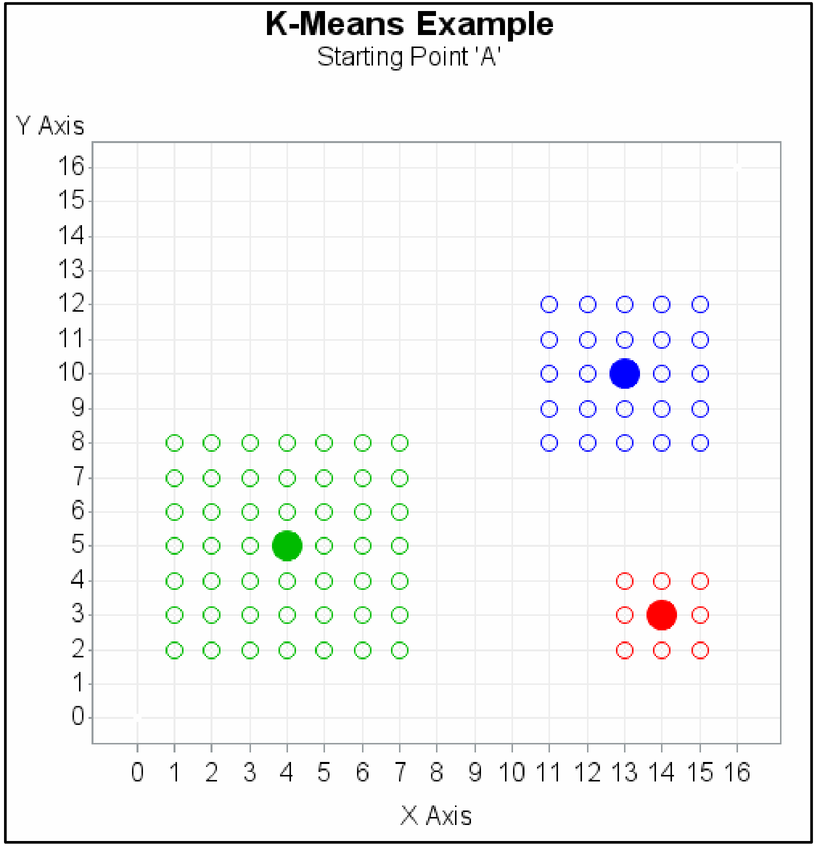


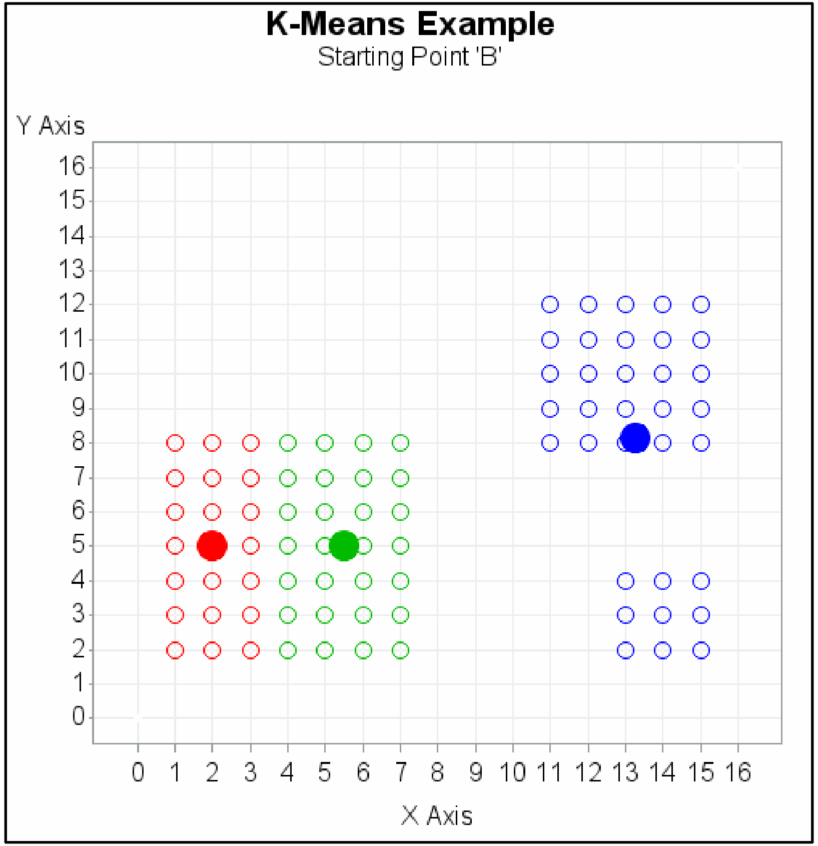
Summary


K-Means Example
Starting Point 'A'
Y Axis
X Axis
0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16


K-Means Example
Starting Point 'B'
Y Axis
X Axis
0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16

Summary

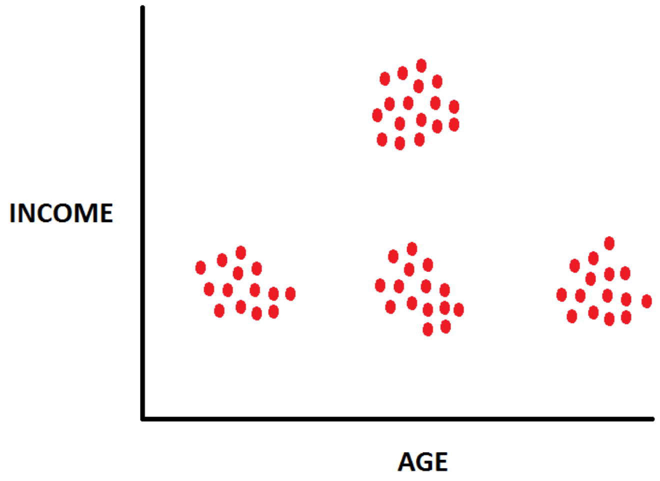

K-Means Example
Starting Point 'A'
Y Axis
X Axis
0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16

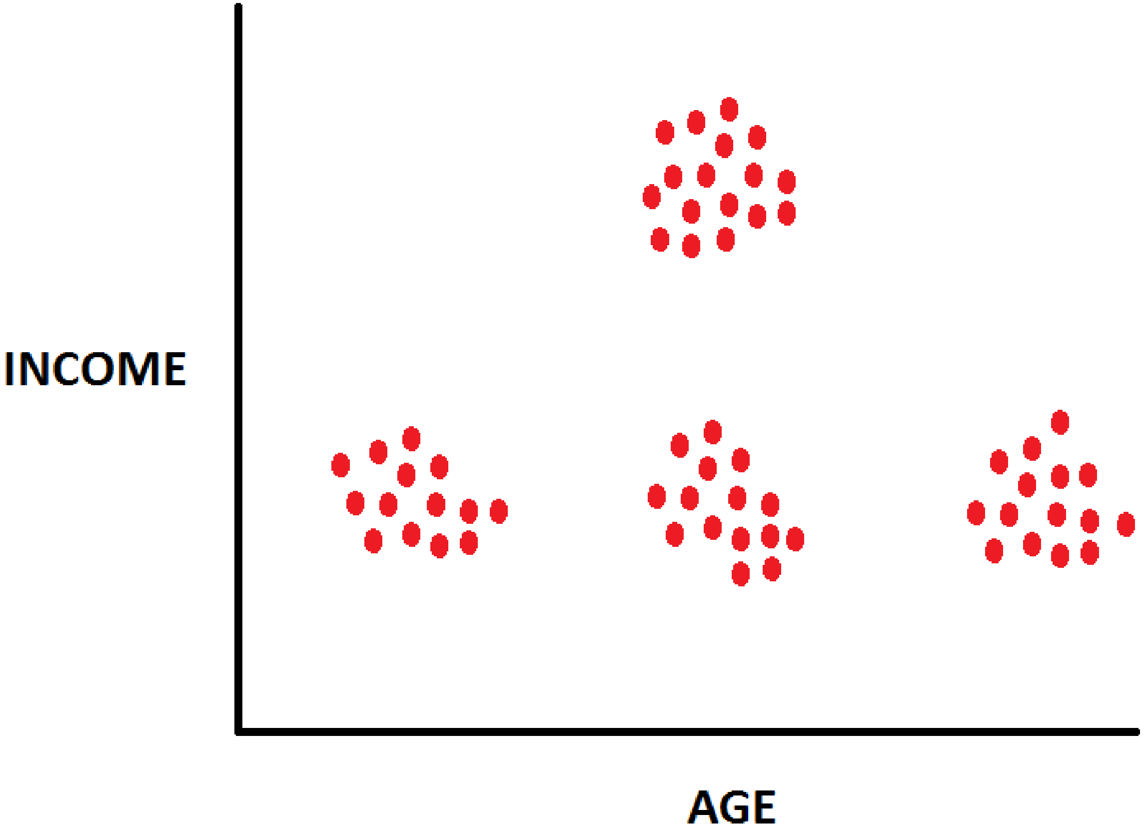

K-Means Example
Starting Point 'B'
Y Axis
X Axis
0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16

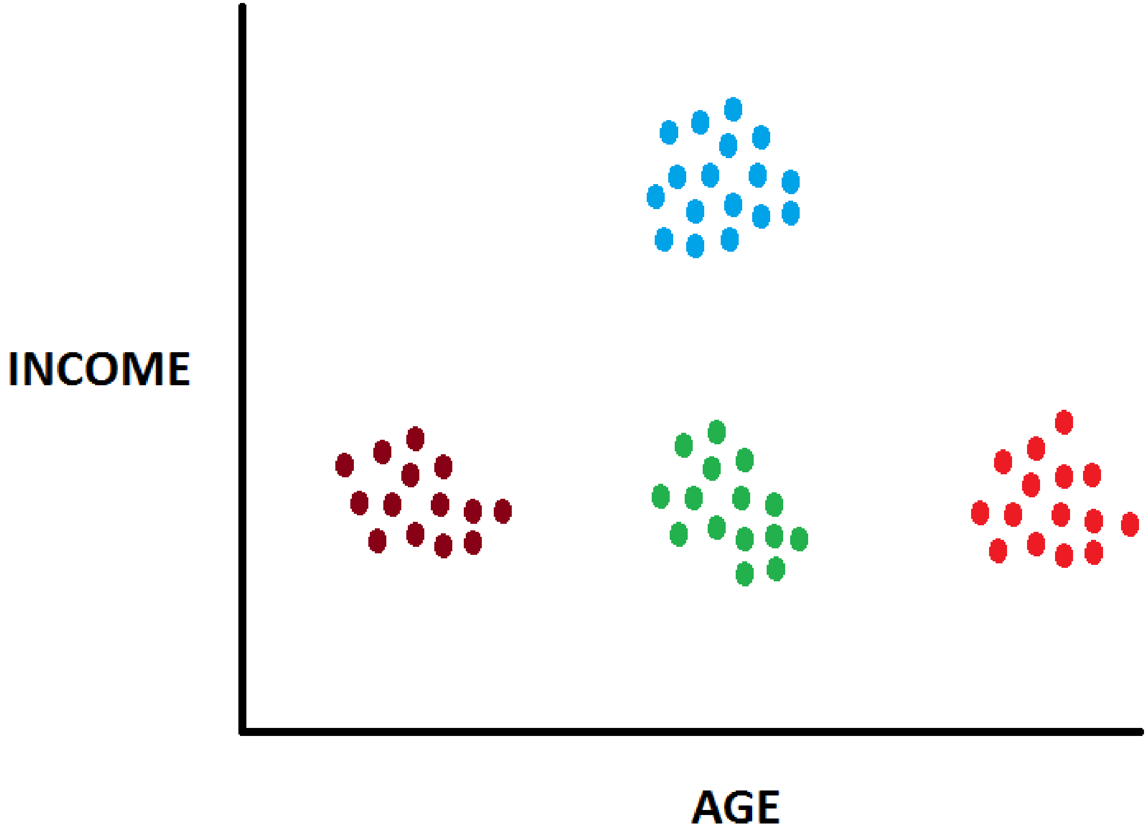
K-Means Clustering

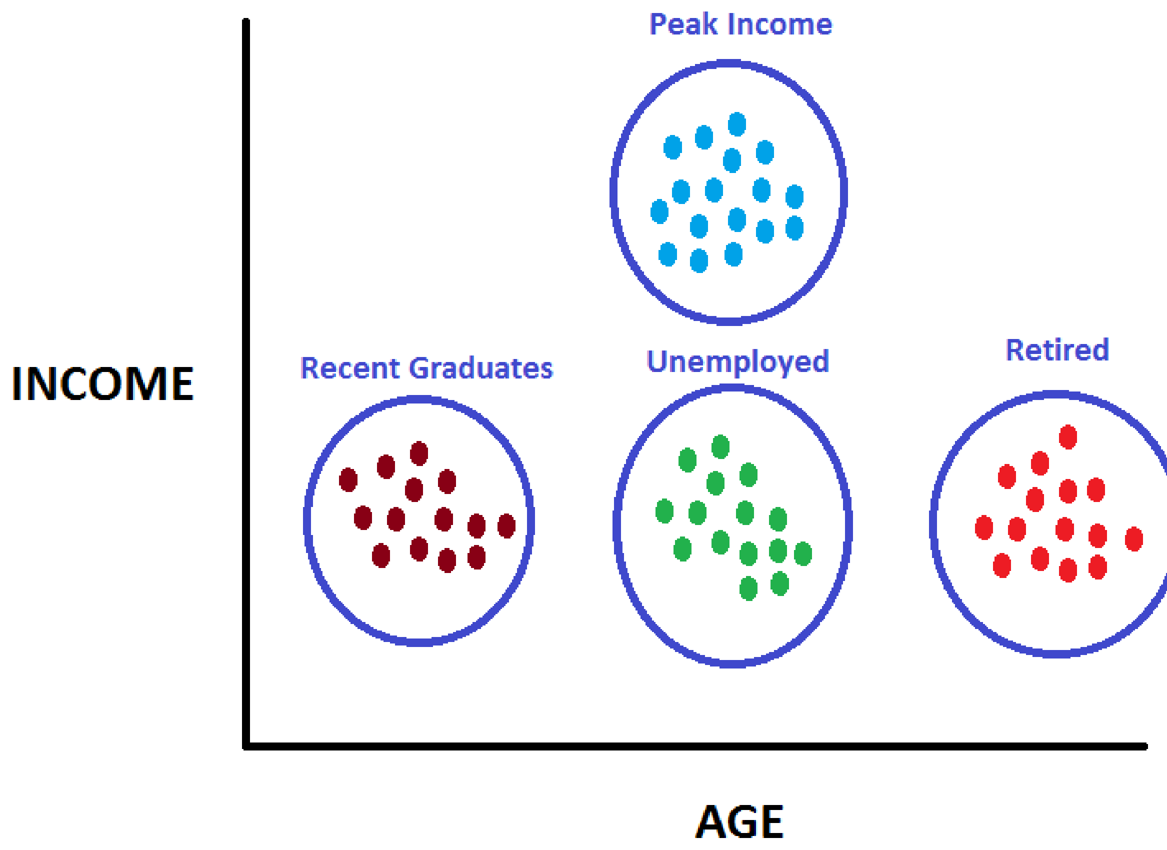
Clusters based on AGE and INCOME

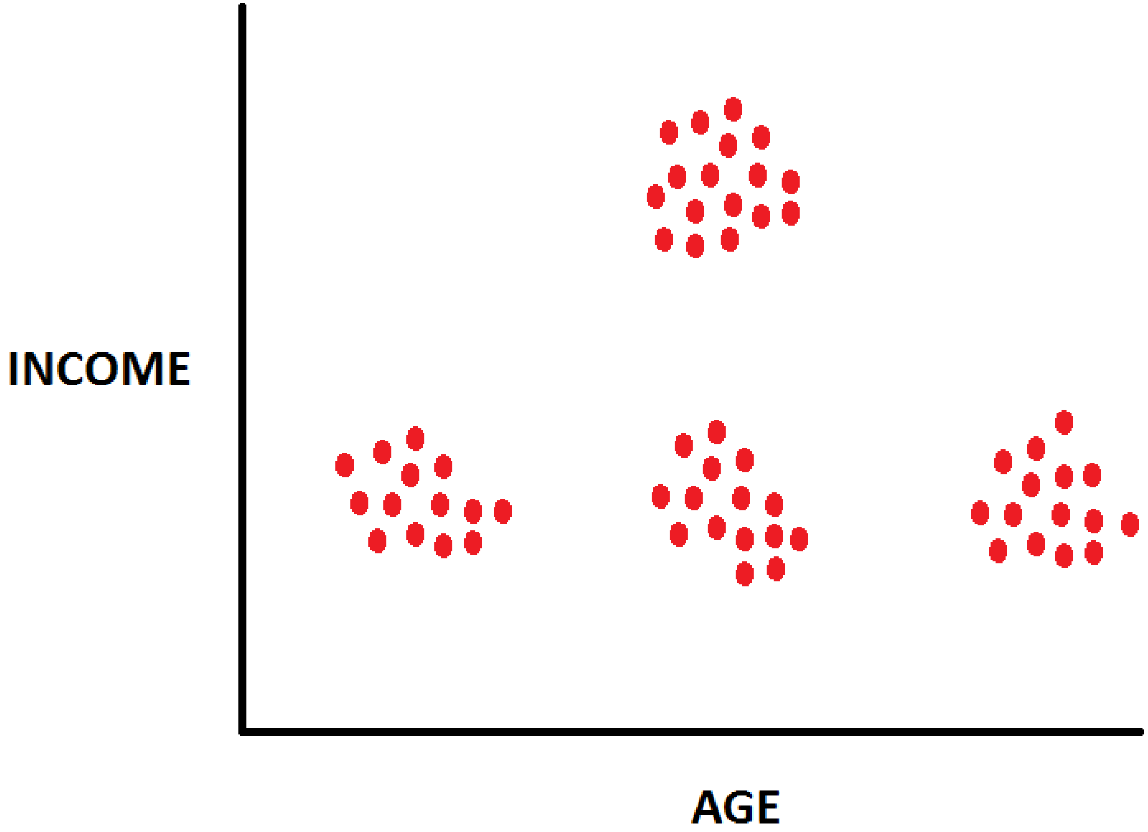
How many clusters do you see?

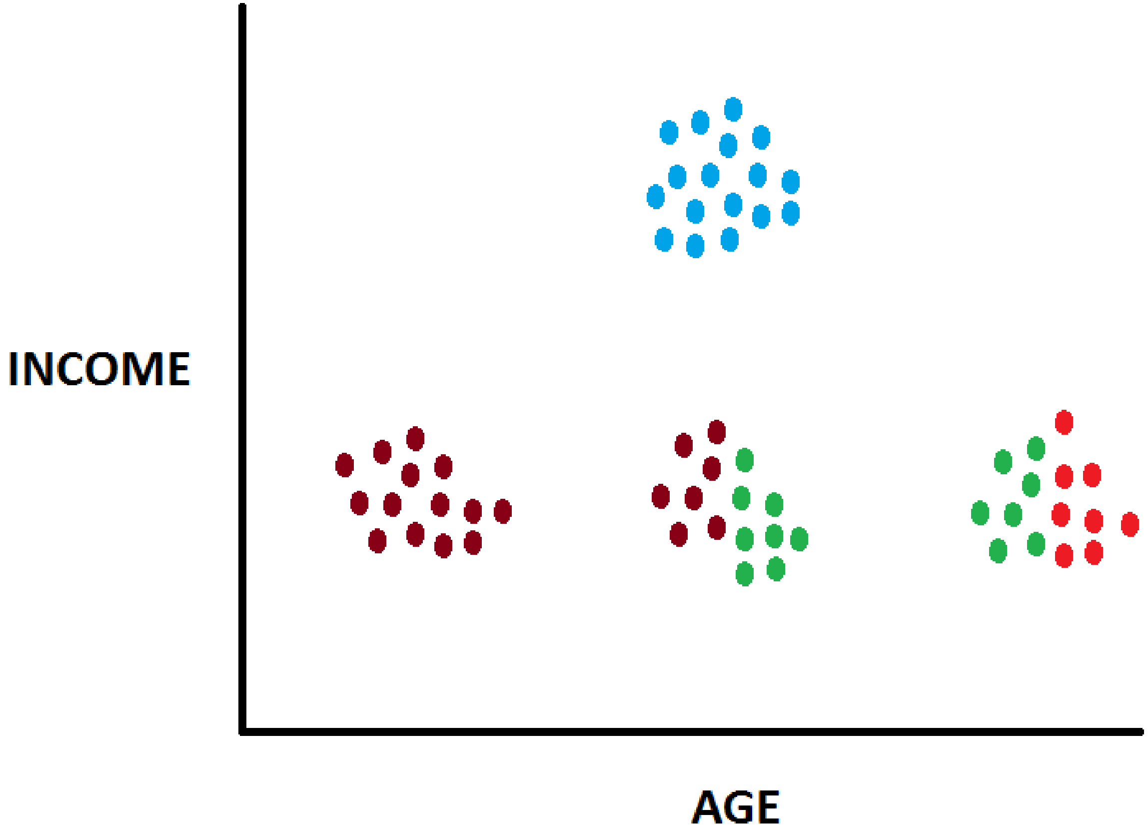


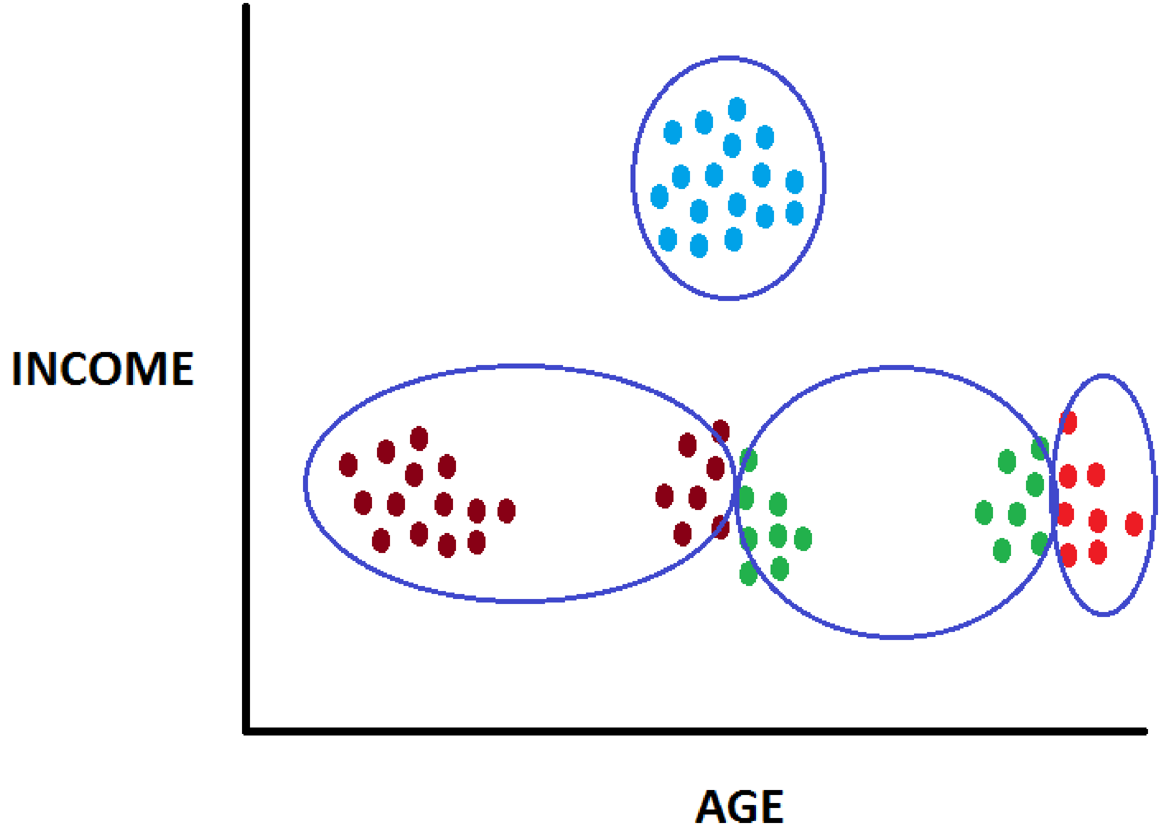

INCOME
AGE


INCOME
AGE

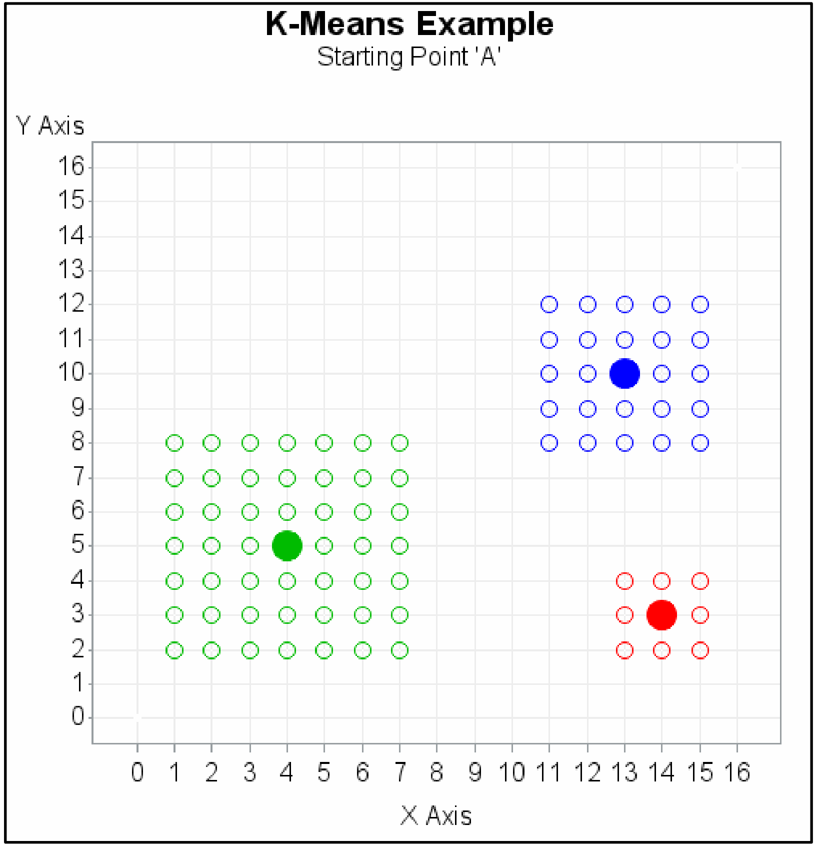

Peak Income
INCOME
Recent Graduates
Unemployed
Retired
AGE


INCOME
AGE

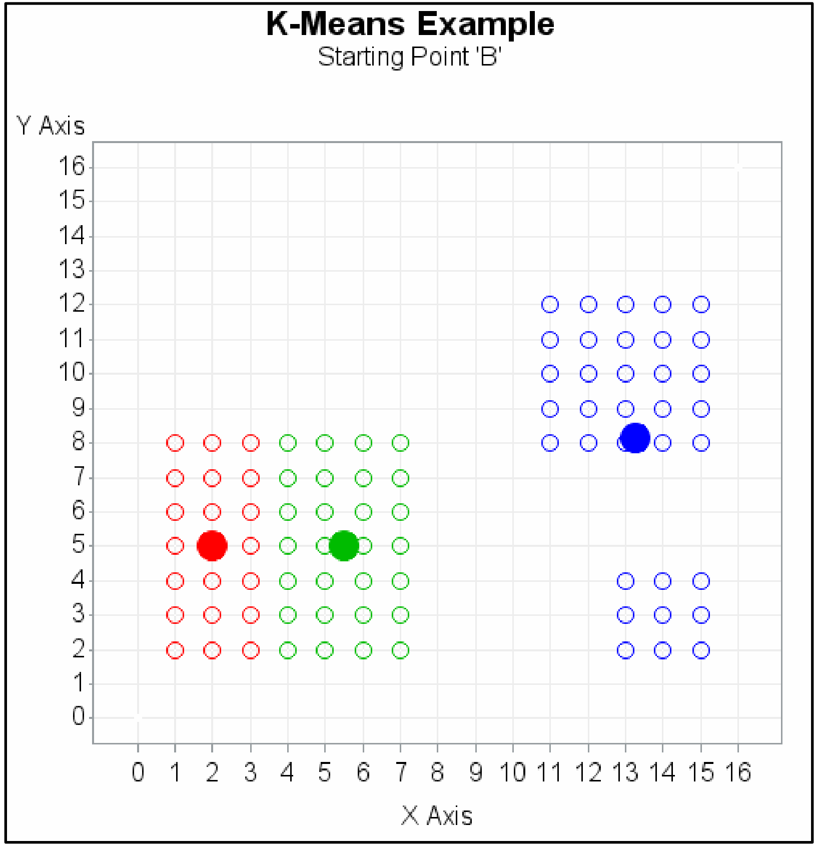

INCOME
AGE


INCOME
AGE

Summary

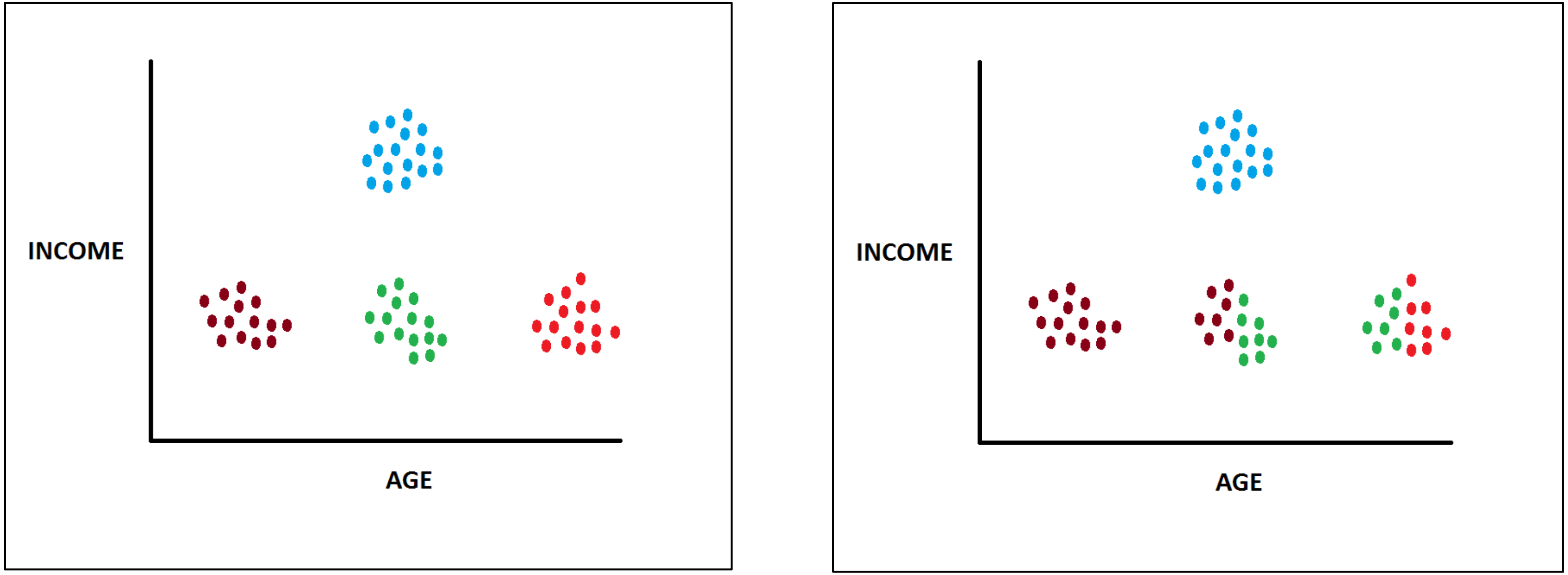


K-Means Example
Starting Point 'A'
Y Axis
X Axis
0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16



K-Means Example
Starting Point 'B'
Y Axis
X Axis
0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16

Summary



What Affects Cluster Results?

- How many clusters are there?
- Cluster Starting Points (“Seeds”)?

What Affects Cluster Results?

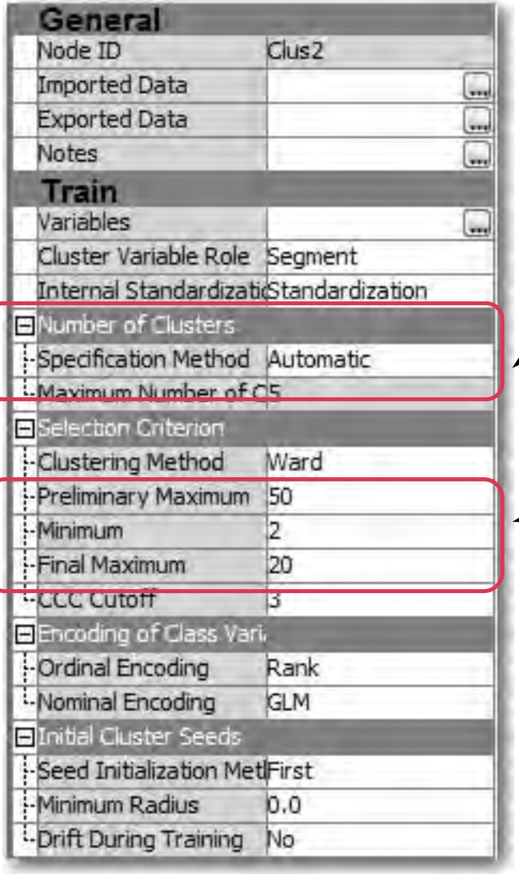
- How many clusters are there?
- Cluster Starting Points (“Seeds”)?

Approximate The Number of Clusters

Diagram 4300


Cluster

How Many Clusters?



Set the number of clusters to"automatic"

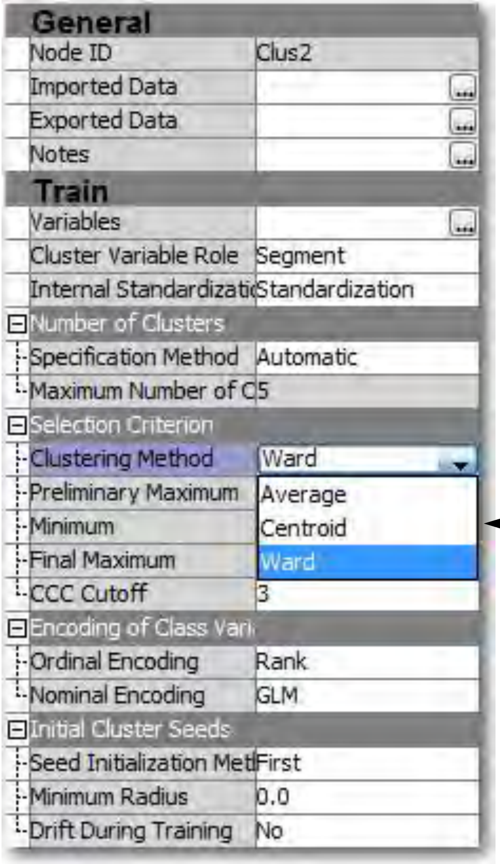
Set the Following Parameters:

- <u>Preliminary Max = 50</u>
  Assume that initially there might be as many as 50 clusters
- <u>Minimum = 2</u>
  When complete, there will be at least 2 clusters.
- <u>Final Maximum = 20</u>
  When complete, there will be no more than 20 clusters.

How Many Clusters?

- SAS Enterprise Miner allows user to “guess” at the number of clusters within a RANGE (example: at least 2 and at most 20 is default)
- SAS Enterprise Miner will estimate the optimal number of clusters
- Optimal number of clusters will vary depending upon clustering parameters.
- STEP1: Narrow the “Search Range” by clustering using multiple parameters

How Many Clusters?



Measurement of cluster distances

- Average
- Centroid
- Ward (Default)

Cluster Selection Methods SAS Enterprise Miner

- Average

  Calculate the average distance from every point in one cluster to every point in another cluster

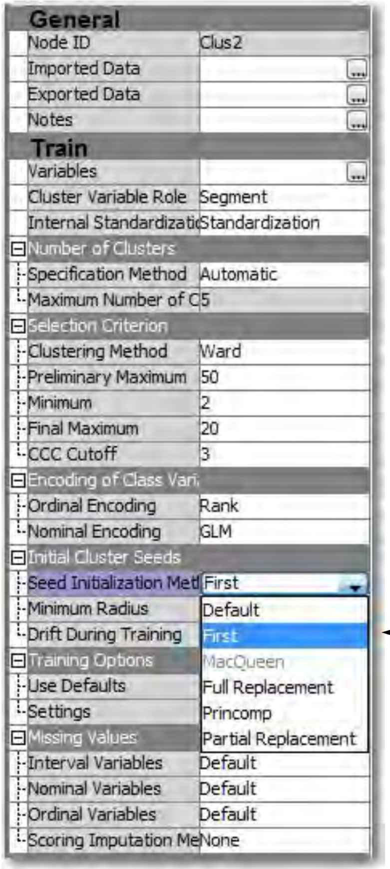
- Centroid

  Find the distance from one cluster center point to another cluster center point

- Ward (Default Method)

  Cluster measurement is based on the ANOVA sum of squares of the two clusters

How Many Clusters?



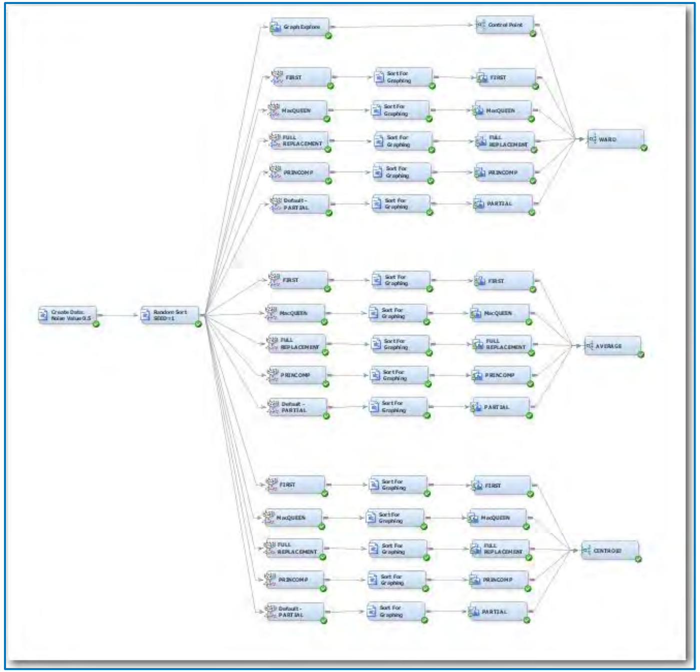
How are Initial Clusters Centers Chosen?

- First “n” Records
- MacQueen Drifting
- Full Replacement
- Princomp
- Partial Replacement (Default)

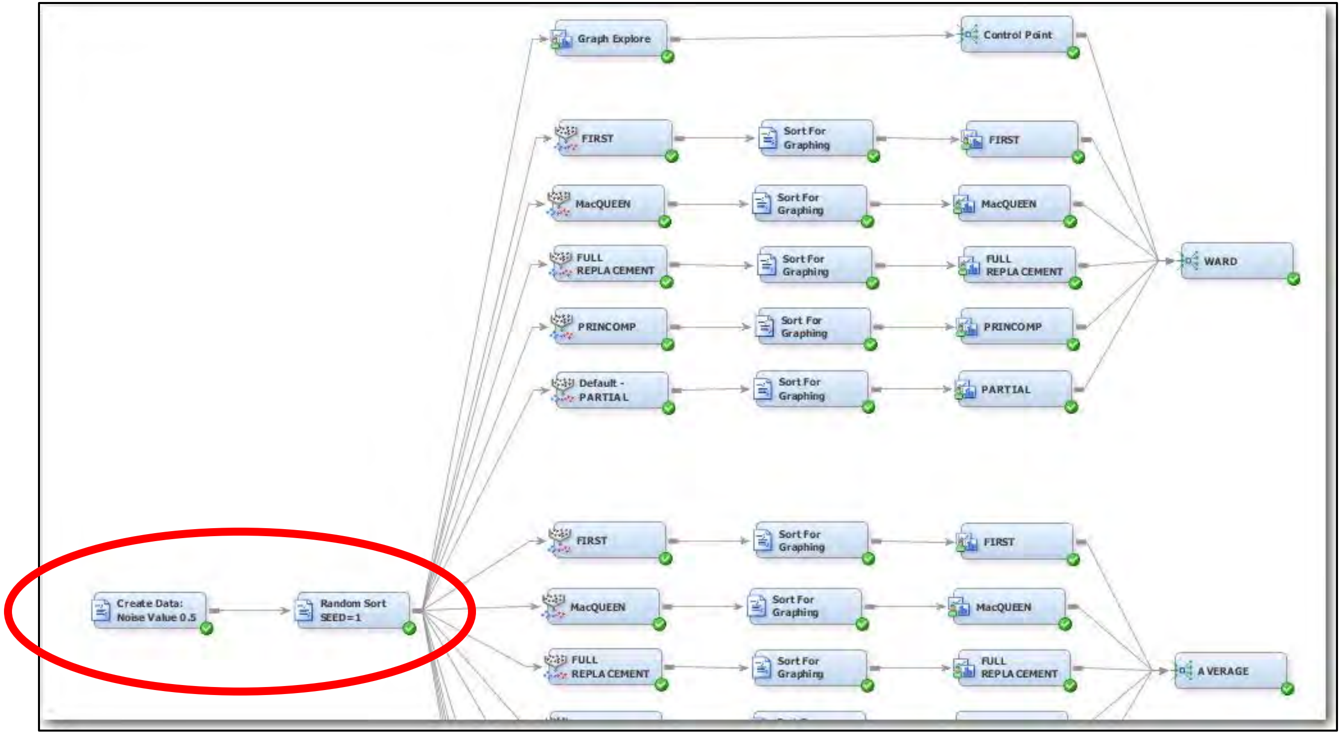
Cluster Seed Selection SAS Enterprise Miner

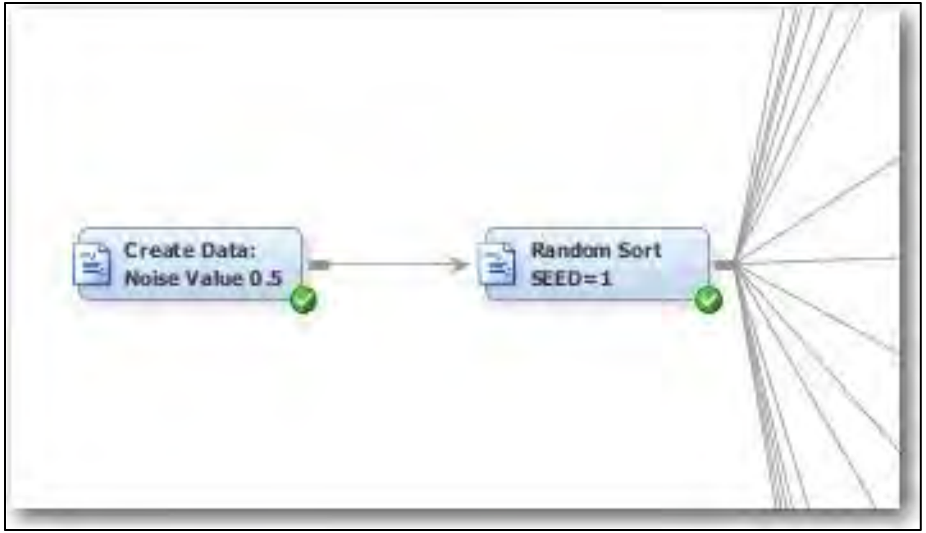
- First "N" Records Method
  - Use the first "N" records in the list as seeds
- Partial Replacement Method (Default)
  - Select "N" records that are far away from each other
- Full Replacement Method
  - Select "N" records that are very far away from each other by looking for outliers.
- Principal Component Method
  - Select "N" evenly spaced records along the first Principal Component Vector
- MacQueen "Drifting" Method
  - Use the first "N" records in the list as seeds. Assign records to clusters one by one and recomputes center after each record is assigned aka "drifting".

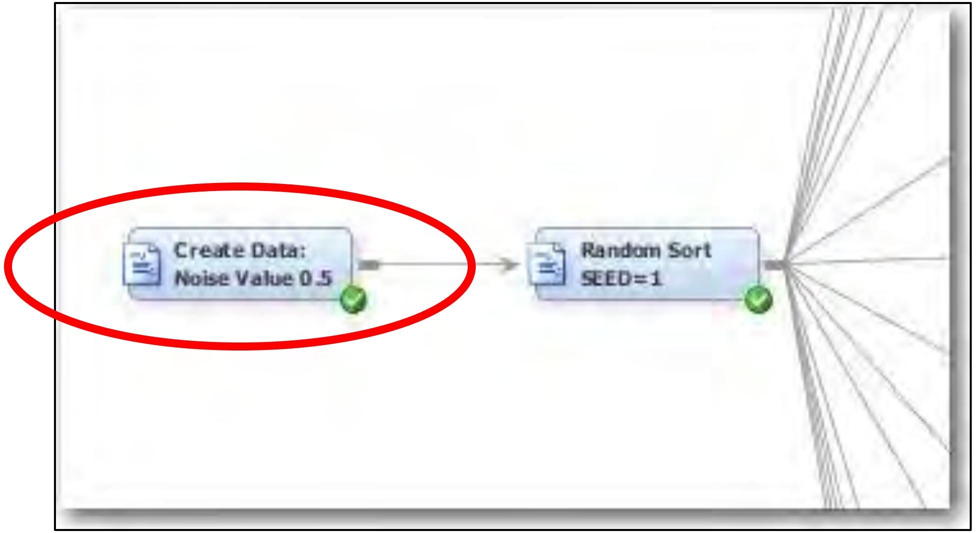
Approximate Number Of Clusters Diagram 4300



Graph Explore
Control Point
FIRST
Sort For Graphing
FIRST
MacQUEEN
Sort For Graphing
MacQUEEN
FULL REPLACEMENT
Sort For Graphing
FULL REPLACEMENT
WARD
PRINCOMP
Sort For Graphing
PRINCOMP
Default - PARTIAL
Sort For Graphing
PARTIAL
Create Data: Noise Value 0.5
Random Sort SEED=1
FIRST
Sort For Graphing
FIRST
MacQUEEN
Sort For Graphing
MacQUEEN
FULL REPLACEMENT
Sort For Graphing
FULL REPLACEMENT
AVERAGE
PRINCOMP
Sort For Graphing
PRINCOMP
Default - PARTIAL
Sort For Graphing
PARTIAL
FIRST
Sort For Graphing
FIRST
MacQUEEN
Sort For Graphing
MacQUEEN
FULL REPLACEMENT
Sort For Graphing
FULL REPLACEMENT
CENTROID
PRINCOMP
Sort For Graphing
PRINCOMP
Default - PARTIAL
Sort For Graphing
PARTIAL


Graph Explore
Control Point
FIRST
Sort For Graphing
FIRST
MacQUEEN
Sort For Graphing
MacQUEEN
FULL REPLACEMENT
Sort For Graphing
FULL REPLACEMENT
WARD
PRINCOMP
Sort For Graphing
PRINCOMP
Default - PARTIAL
Sort For Graphing
PARTIAL
FIRST
Sort For Graphing
FIRST
Create Data: Noise Value 0.5
Random Sort SEED=1
MacQUEEN
Sort For Graphing
MacQUEEN
FULL REPLACEMENT
Sort For Graphing
FULL REPLACEMENT
AVERAGE


Create Data:
Noise Value 0.5
Random Sort
SEED=1


Create Data:
Noise Value 0.5
Random Sort
SEED=1

Example 1: Random Seeds – Synthetic Data

SAS Program to generate synthetic data

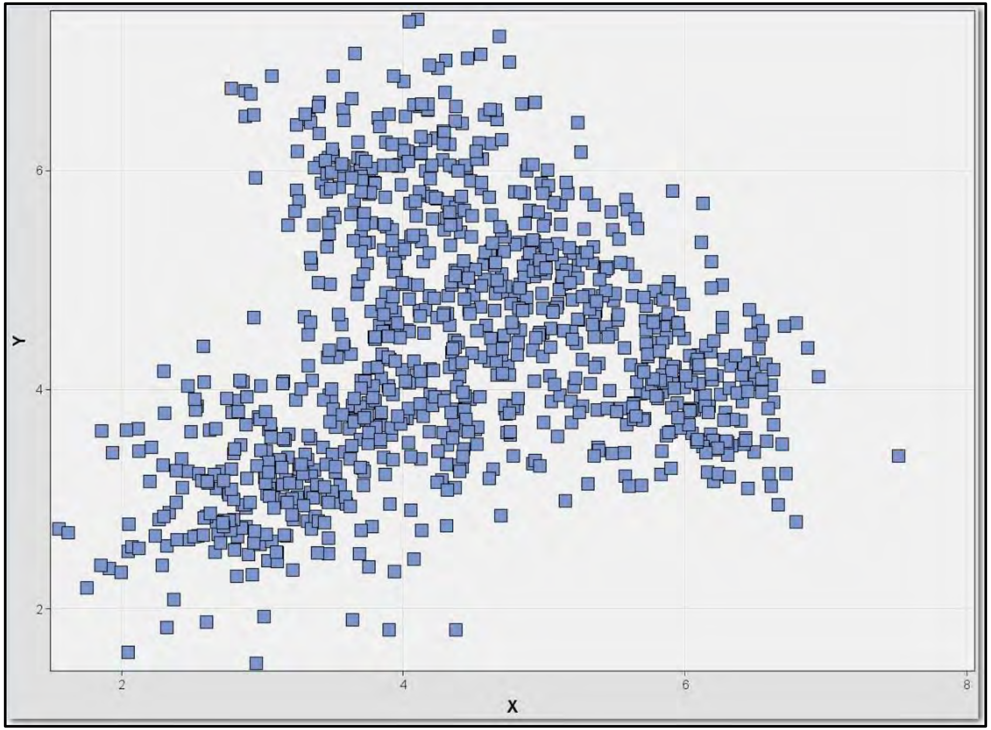
- Program creates 1000 data points with two values: X,Y
  - 200 points centered at (3,3)
  - 200 points centered at (5,5)
  - 200 points centered at (4,6)
  - 200 points centered at (6,4)
  - 200 points centered at (4,4)

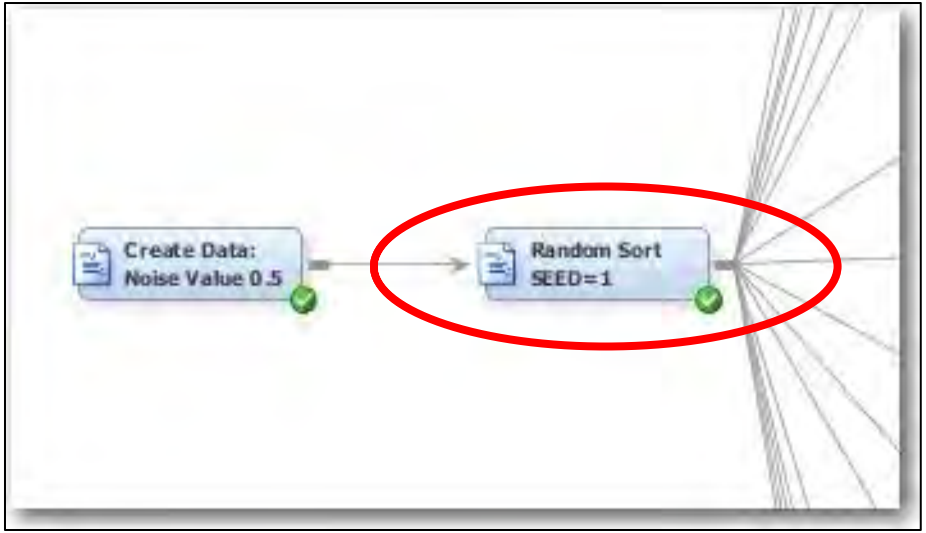
- Each X and Y value has noise added to
  - Normally distributed random number
  - Random number is multiplied by a weight of 0.5

Example 1: Random Seeds – Synthetic Data

```
%let COUNT                = 200
%let WEIGHT               = 0.5;
%let SEED                 = 1;

%let INFILE               = INFILE;
%let OUTFILE              = RANDOM_DATA;

data &INFILE.;
do I = 1 to &COUNT.;
        X                 = 3.0;
        Y                 = 3.0;
        NOISE_X           = rannor(&SEED.);
        NOISE_Y           = rannor(&SEED.);
        output;
        X                 = 5.0;
        Y                 = 5.0;
        NOISE_X           = rannor(&SEED.);
        NOISE_Y           = rannor(&SEED.);
        output;
        X                 = 4.0;
        Y                 = 6.0;
        NOISE_X           = rannor(&SEED.);
        NOISE_Y           = rannor(&SEED.);
        output;
        X                 = 6.0;
        Y                 = 4.0;
        NOISE_X           = rannor(&SEED.);
        NOISE_Y           = rannor(&SEED.);
        output;
        X                 = 4.0;
        Y                 = 4.0;
        NOISE_X           = rannor(&SEED.);
        NOISE_Y           = rannor(&SEED.);
        output;
end;
drop I;
run;

data &OUTFILE.;
set &INFILE.;
        X                 = X + &WEIGHT.*NOISE_X;
        Y                 = Y + &WEIGHT.*NOISE_Y;
drop NOISE_X;
drop NOISE_Y;
run;
```

Noise Level 0.5




Create Data:
Noise Value 0.5
Random Sort
SEED=1

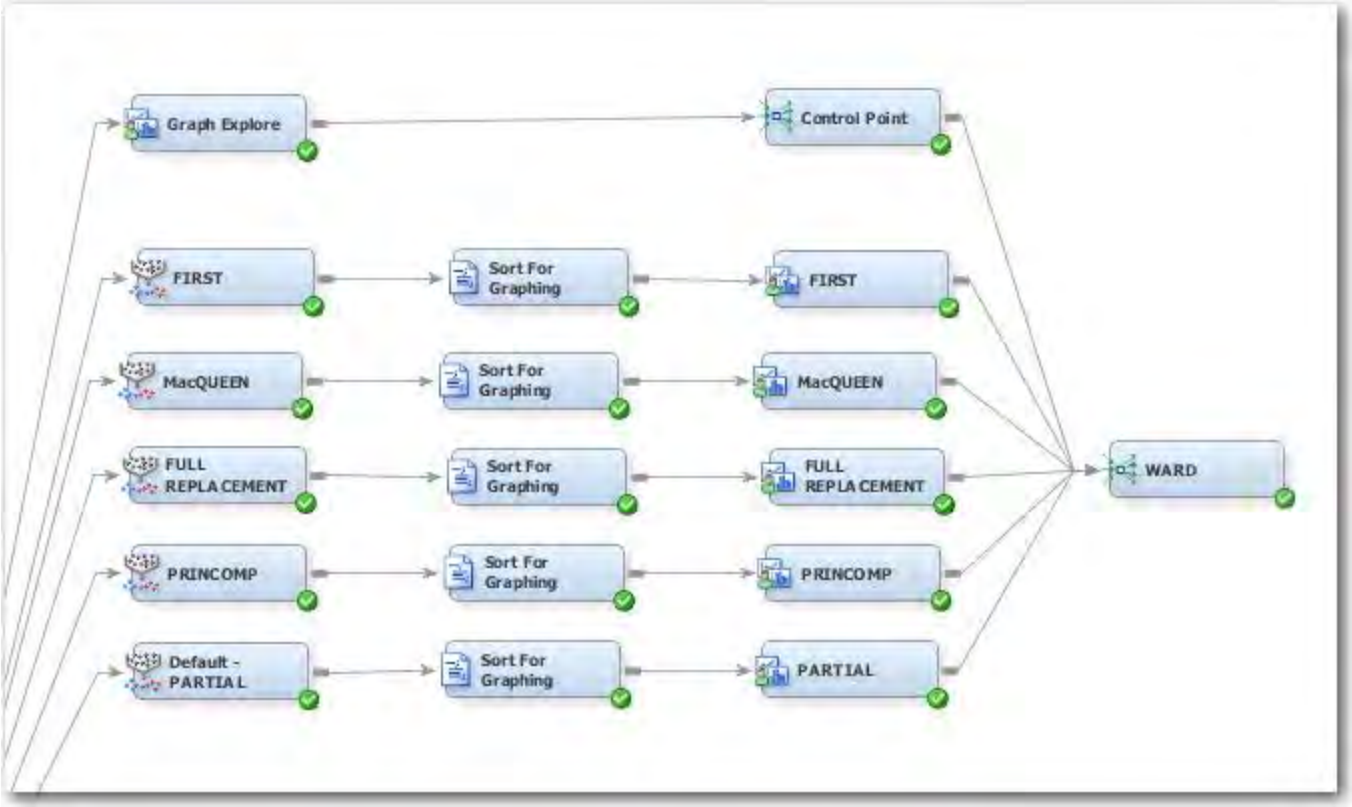
Random Seeds – Shuffle Cards

```
%let SEED           = 1;

%let INFILE         = RANDOM_DATA;
%let TEMPFILE       = TEMPFILE;
%let OUTFILE        = SORTED_DATA;

data &TEMPFILE.;
set &INFILE.;
SORT = ranuni( &SEED. );
run;

proc sort data=&TEMPFILE.;
by SORT;
run;

data &OUTFILE.;
set &TEMPFILE.;
drop SORT;
run;

proc print data=&OUTFILE.(obs=5);
run;
```

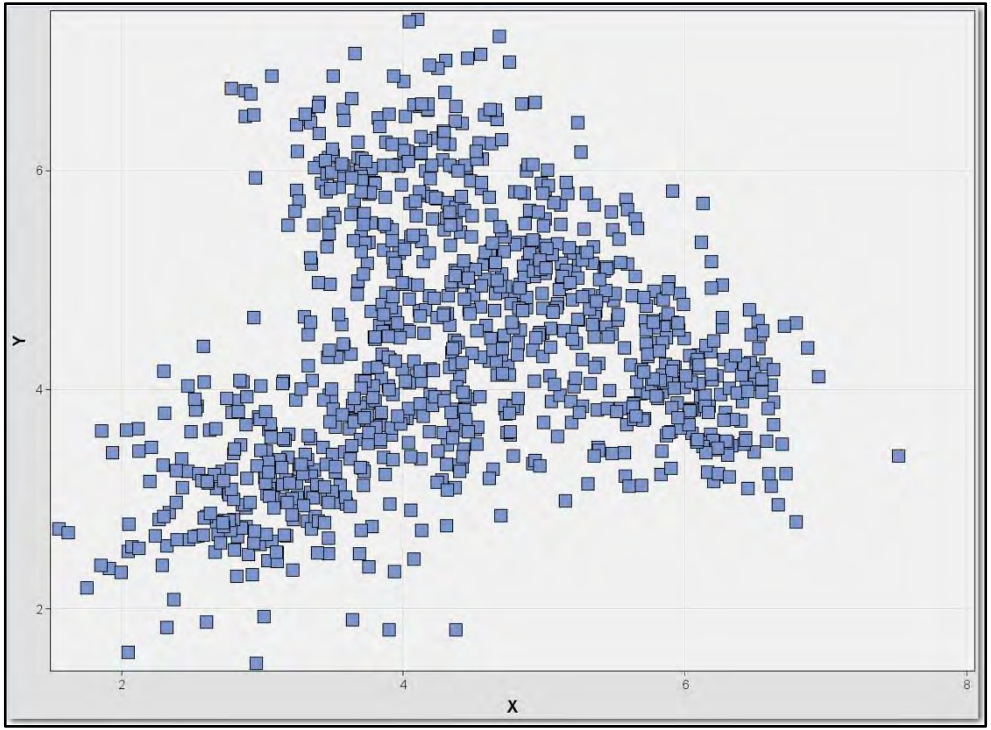
Random Seeds – Shuffle Cards

```
%let SEED            = 1;

%let INFILE          = RANDOM_DATA;
%let TEMPFILE        = TEMPFILE;
%let OUTFILE         = SORTED_DATA;

data &TEMPFILE.;
set &INFILE.;
SORT = ranuni( &SEED. );
run;

proc sort data=&TEMPFILE.;
by SORT;
run;

data &OUTFILE.;
set &TEMPFILE.;
drop SORT;
run;

proc print data=&OUTFILE.(obs=5);
run;
```

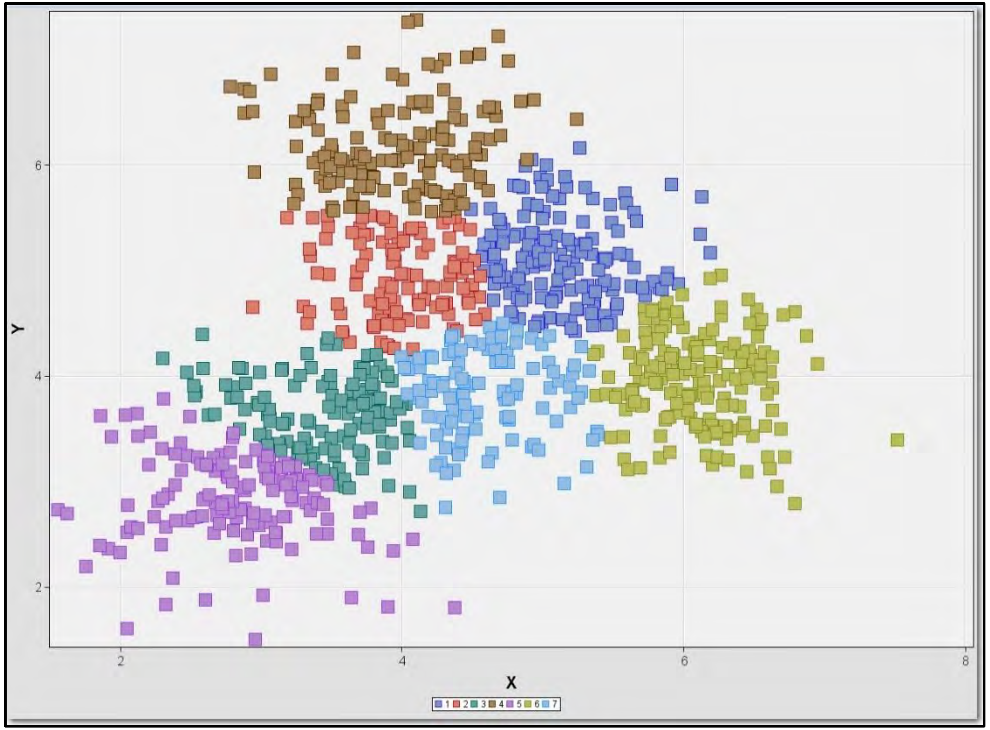
Random Number Seed:
Changing this value will cause the list of data points to be put in a different order ("shuffled")


Graph Explore
Control Point
FIRST
Sort For Graphing
FIRST
MacQUEEN
Sort For Graphing
MacQUEEN
FULL REPLACEMENT
Sort For Graphing
FULL REPLACEMENT
WARD
PRINCOMP
Sort For Graphing
PRINCOMP
Default - PARTIAL
Sort For Graphing
PARTIAL

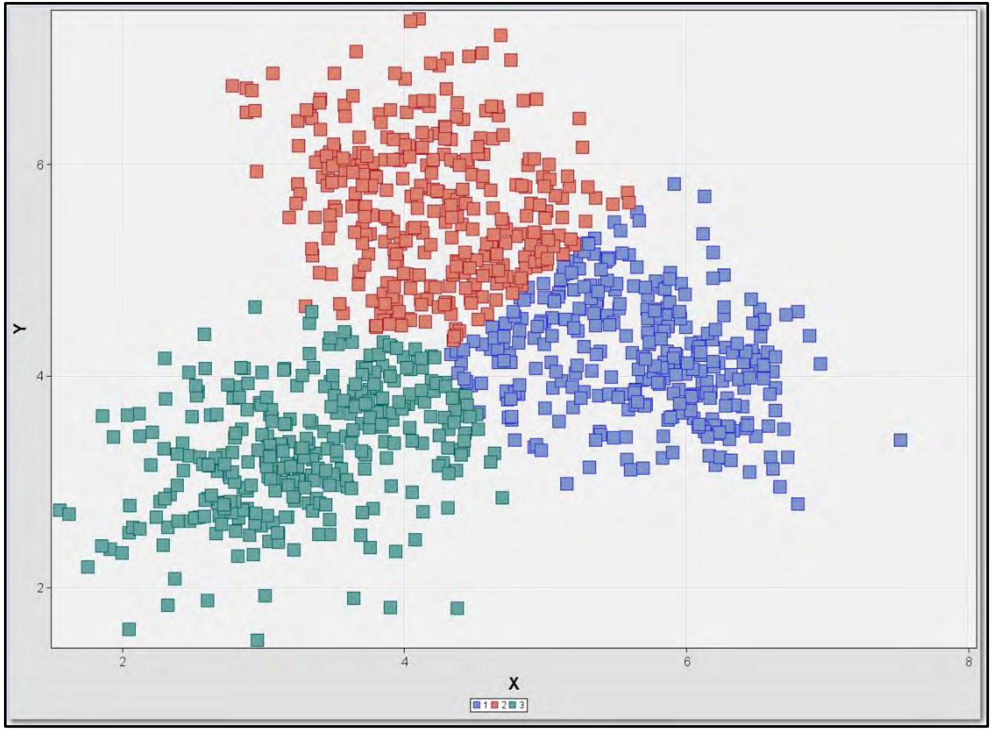
How Many Clusters?



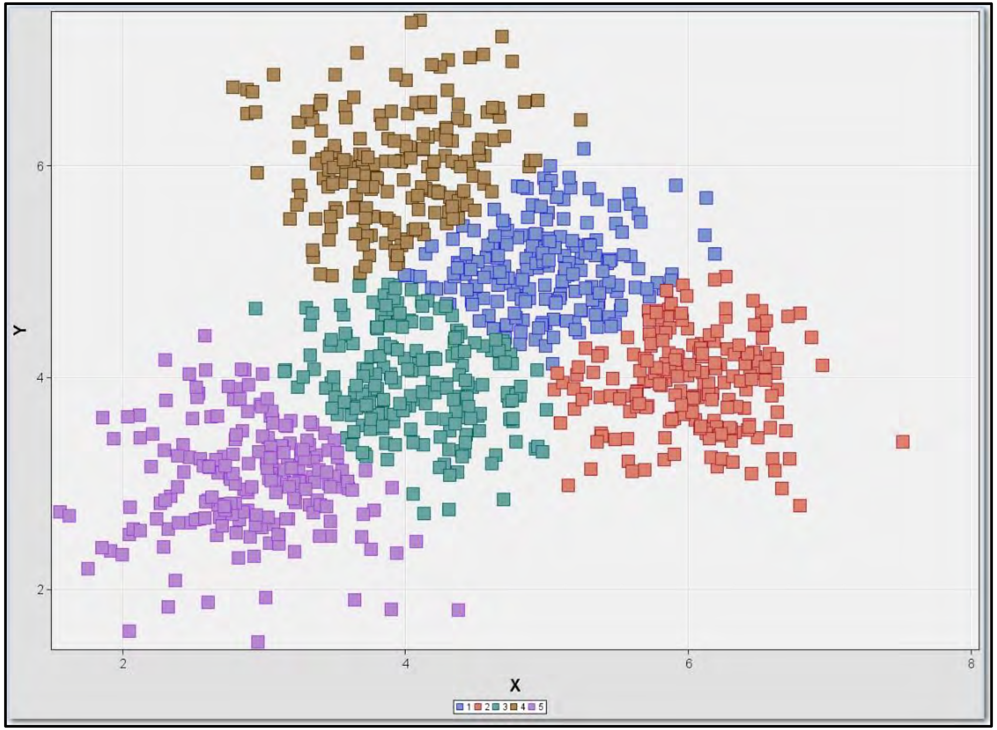
Ward / First = 7 clusters



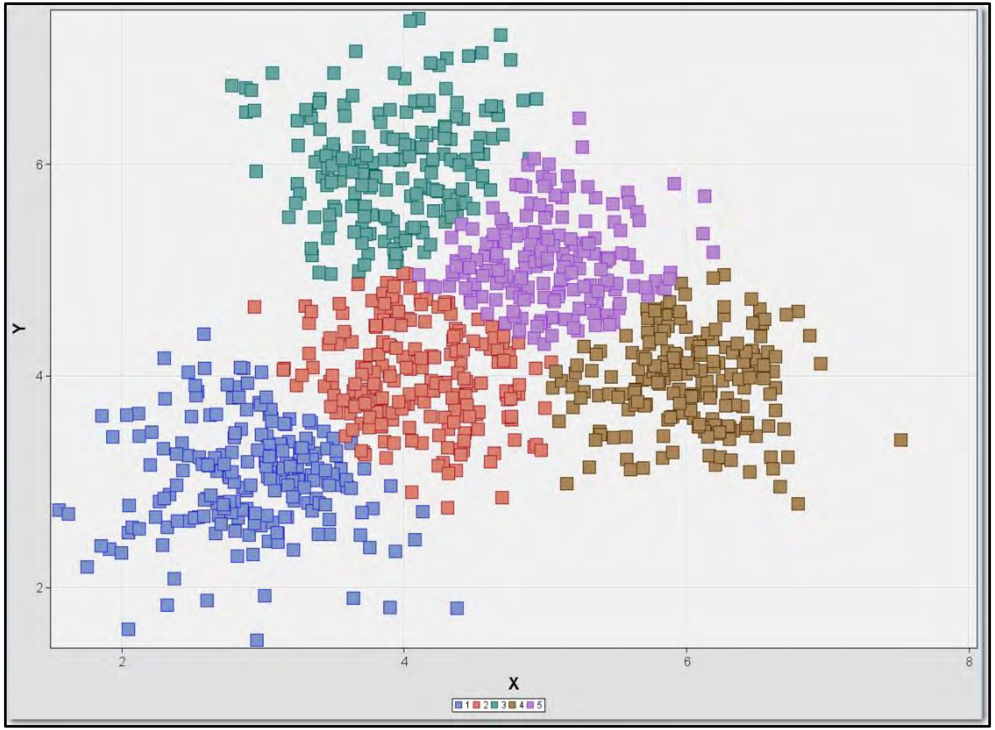
Ward / MacQueen = 3 clusters



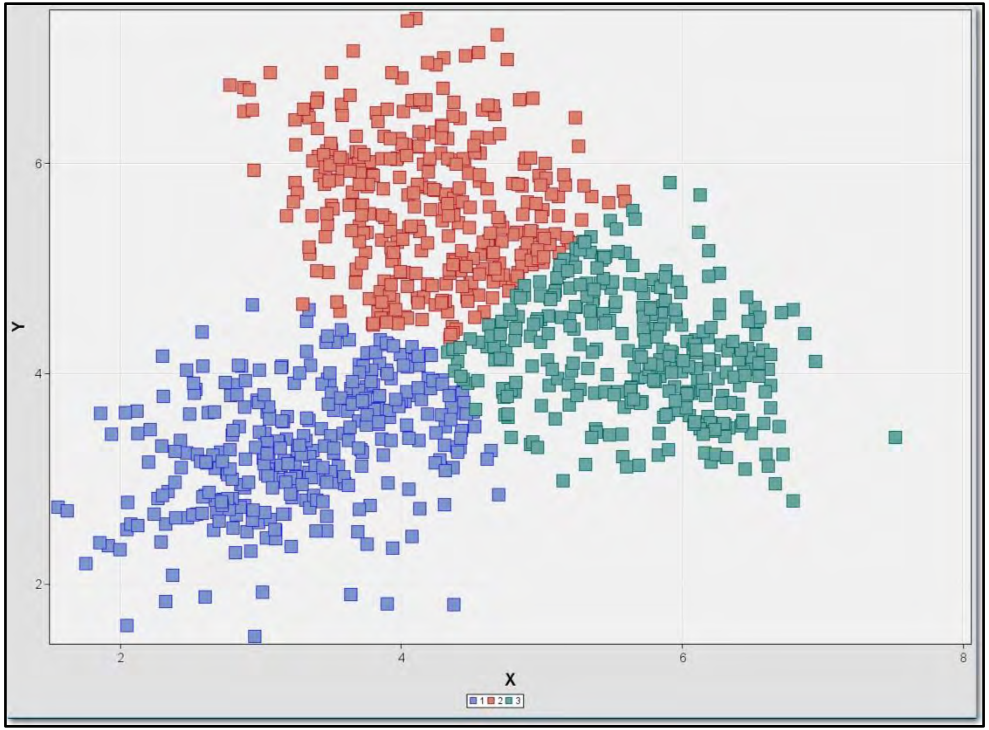
Ward / Full = 5 clusters

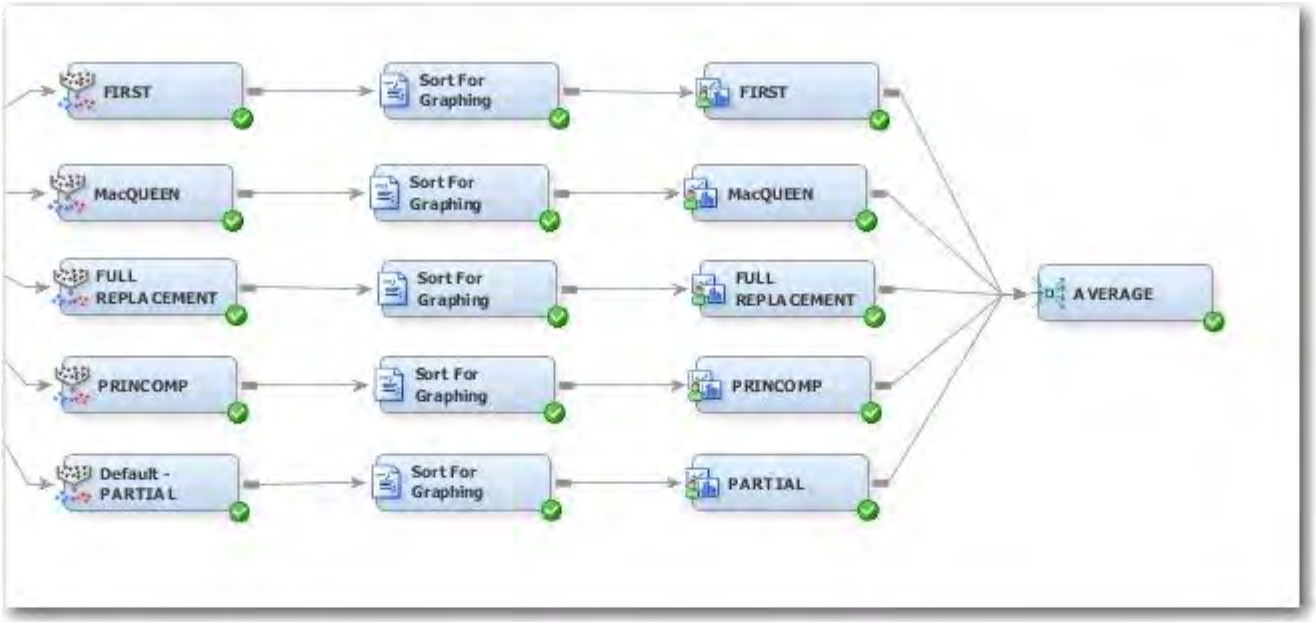


Ward / Princomp = 5 clusters

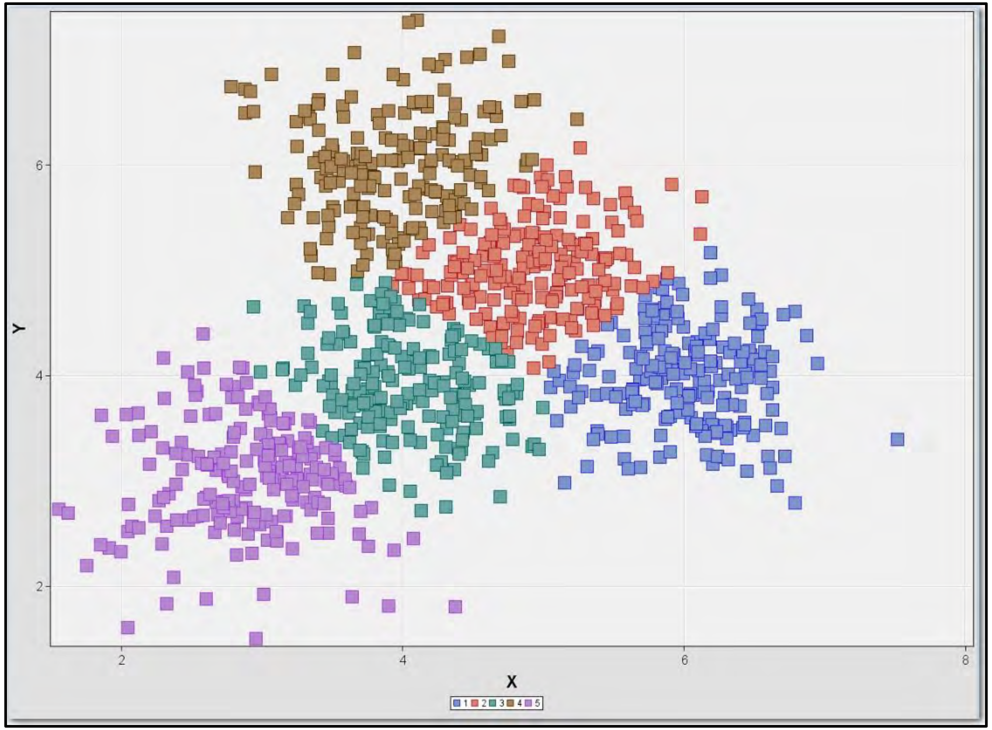


Ward / Partial = 3 clusters

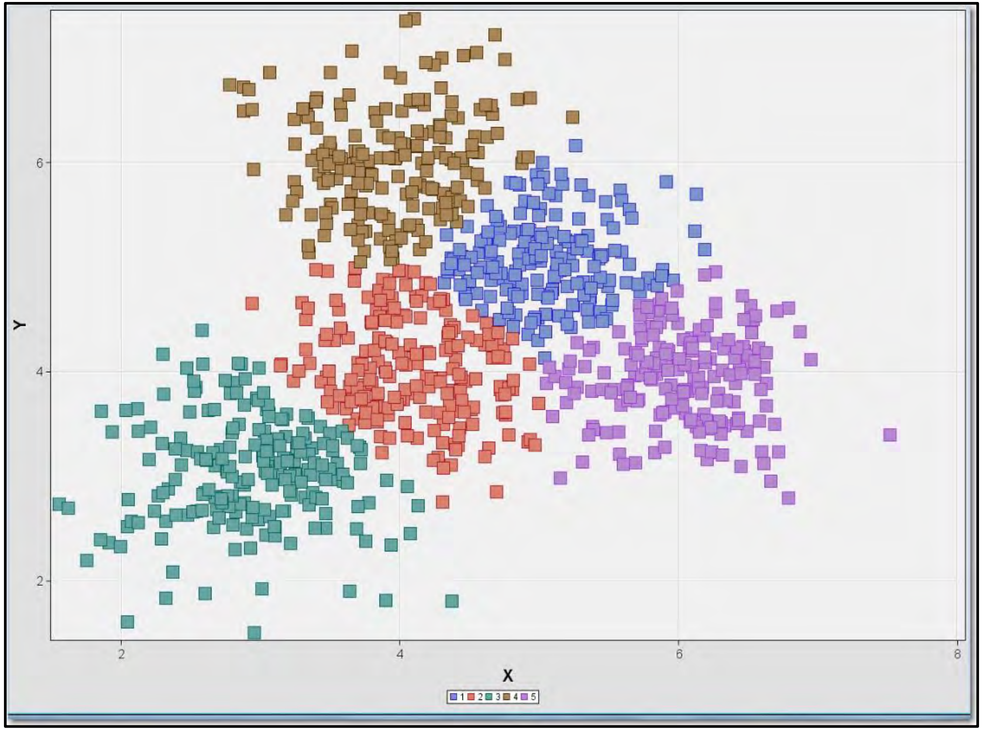



FIRST
Sort For Graphing
FIRST
MacQUEEN
Sort For Graphing
MacQUEEN
FULL REPLACEMENT
Sort For Graphing
FULL REPLACEMENT
AVERAGE
PRINCOMP
Sort For Graphing
PRINCOMP
Default - PARTIAL
Sort For Graphing
PARTIAL

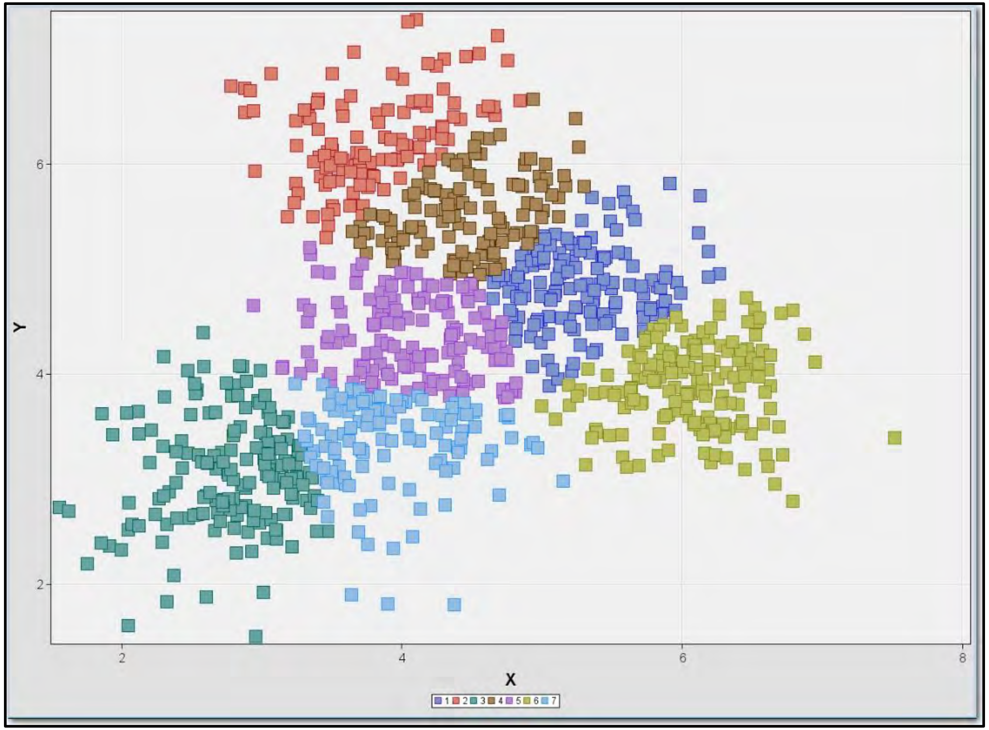
Average / First = 5 clusters



Average / MacQueen = 5 clusters

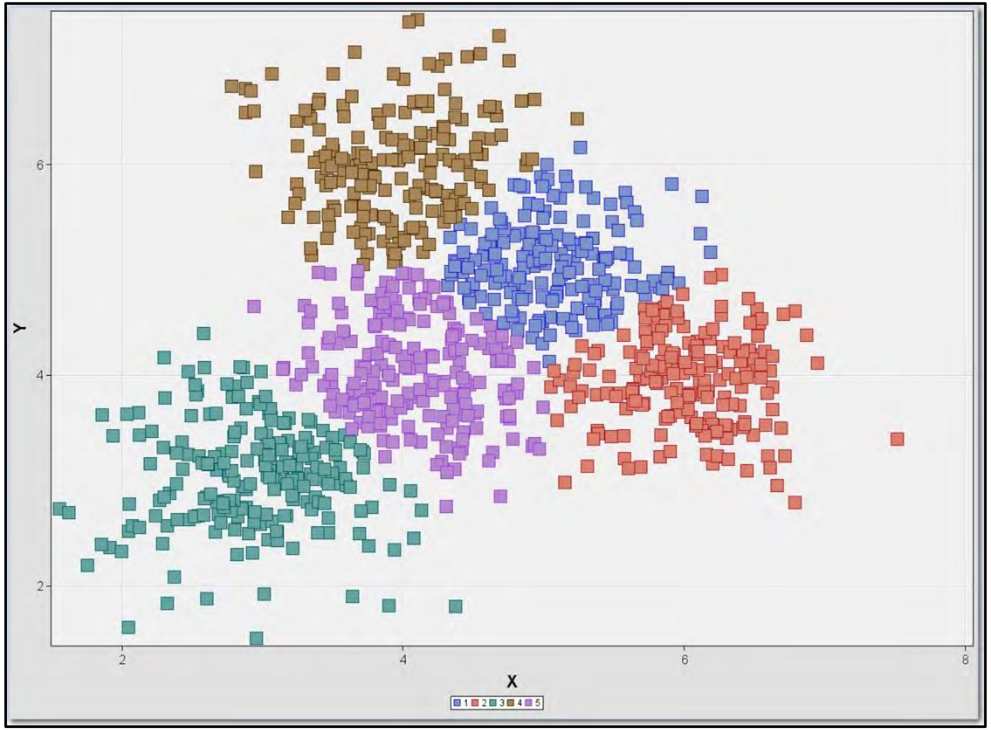


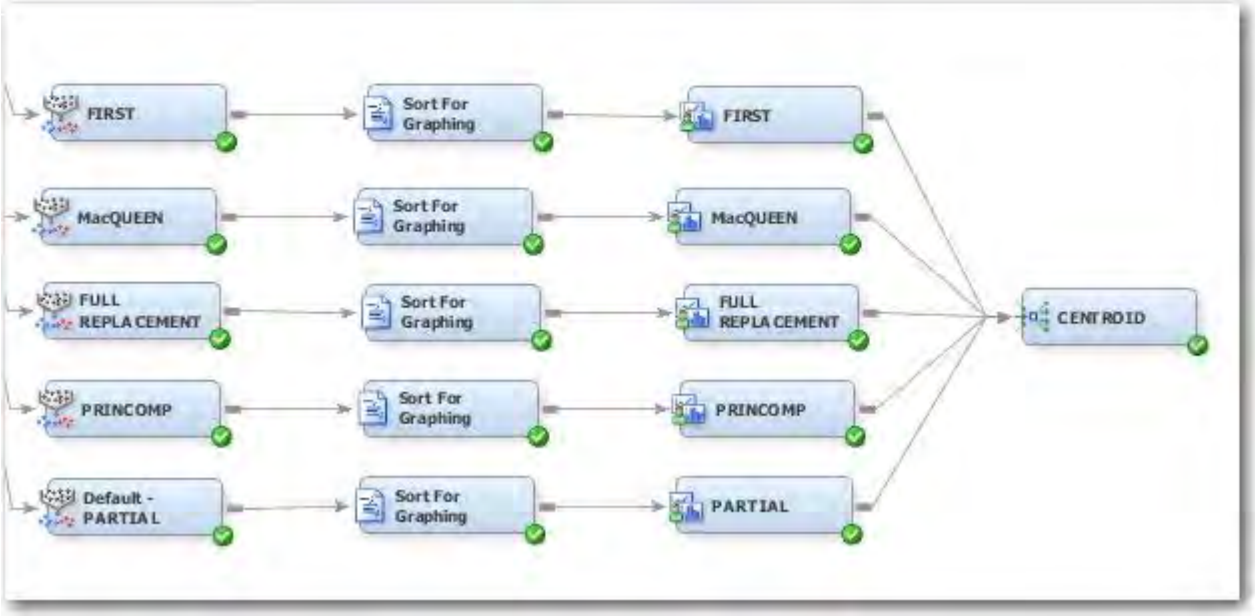
Average / Full = 7 clusters



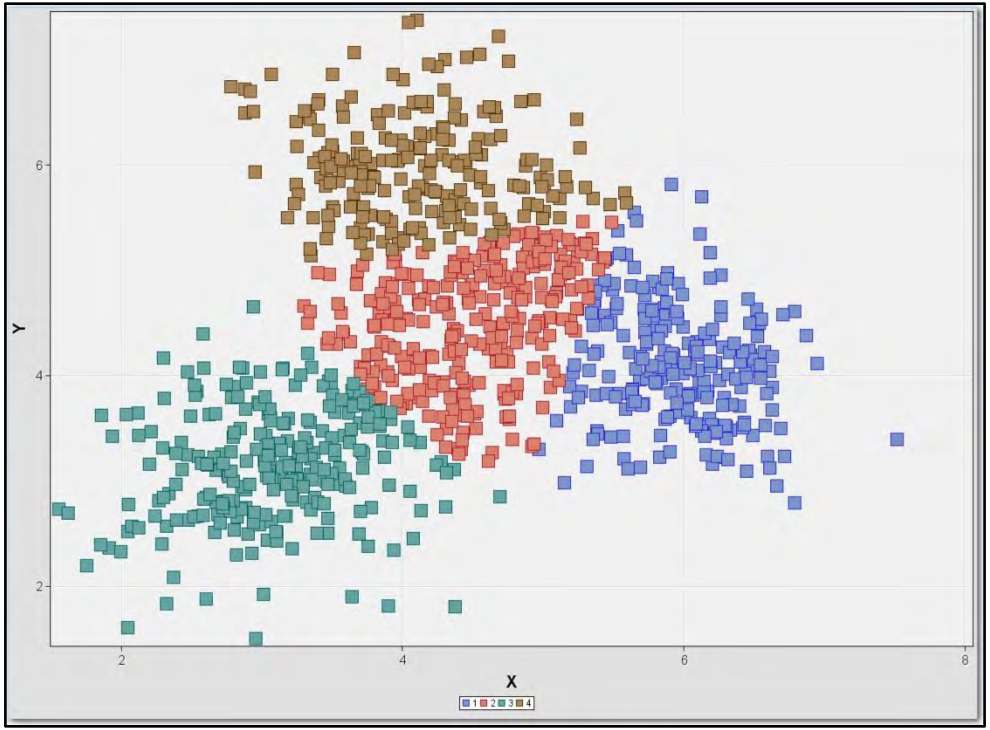
Average / Princomp = 5 clusters

Average / Partial = 5 clusters

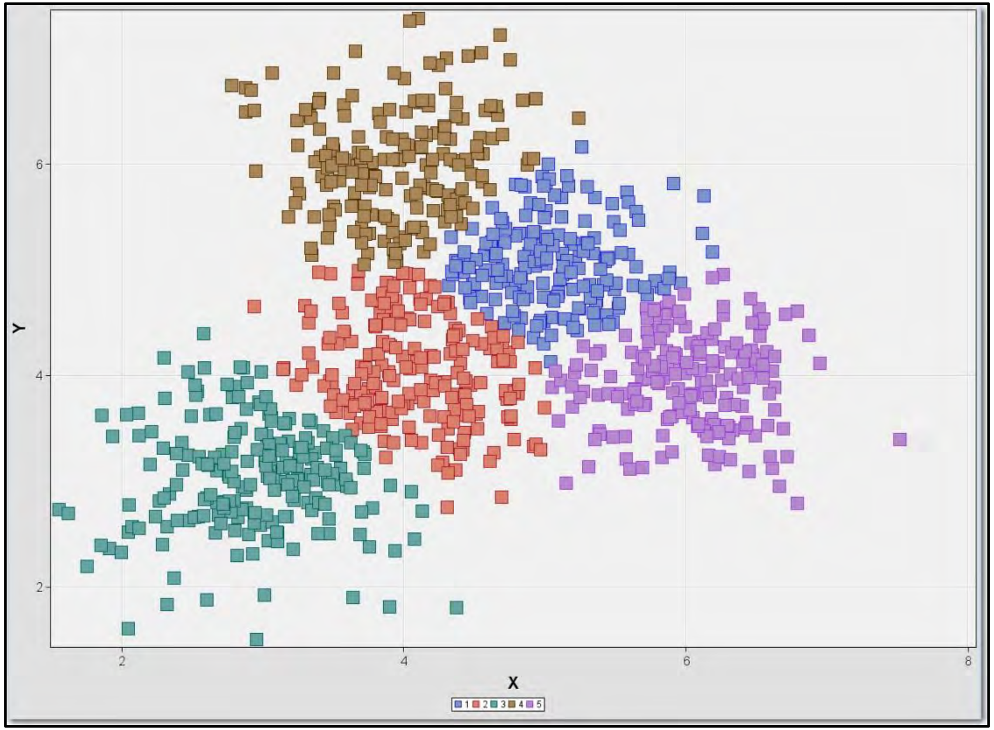



FIRST
MacQUEEN
FULL REPLACEMENT
PRINCOMP
Default - PARTIAL
Sort For Graphing
Sort For Graphing
Sort For Graphing
Sort For Graphing
Sort For Graphing
FIRST
MacQUEEN
FULL REPLACEMENT
PRINCOMP
PARTIAL
CENTROID

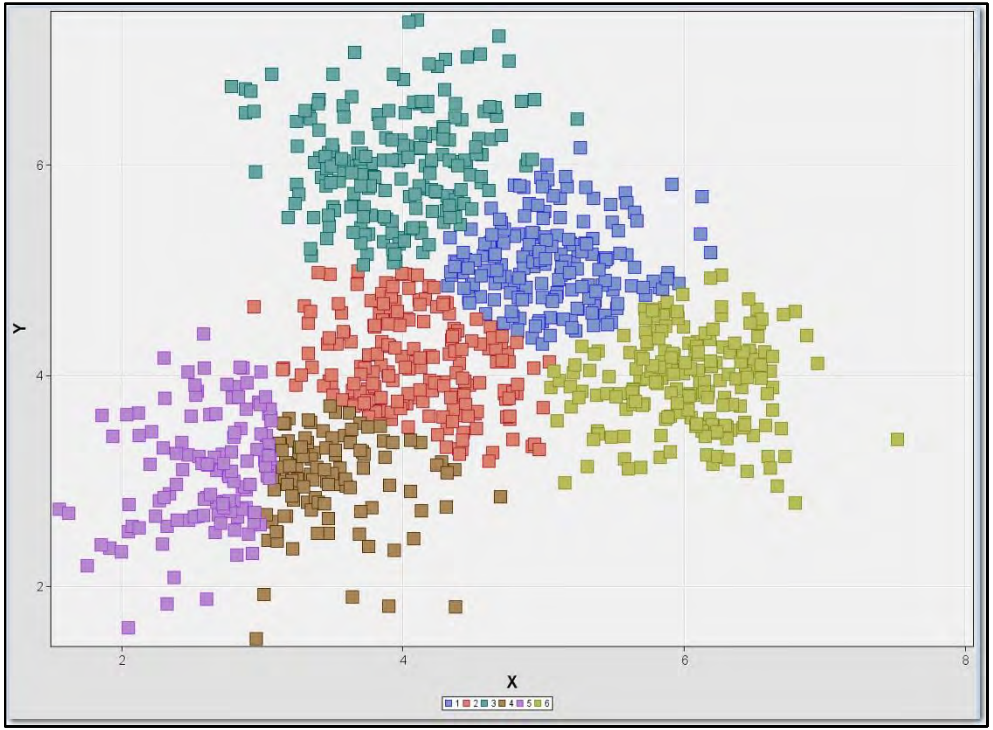
Centroid / First = 4 clusters



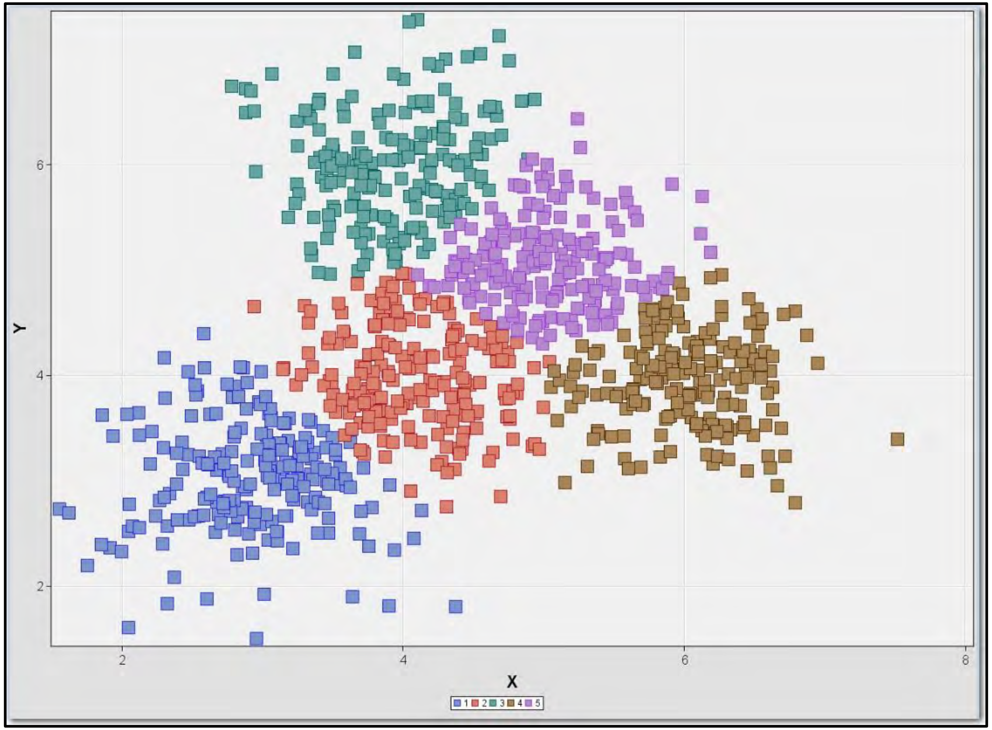
Centroid / MacQueen = 5 clusters



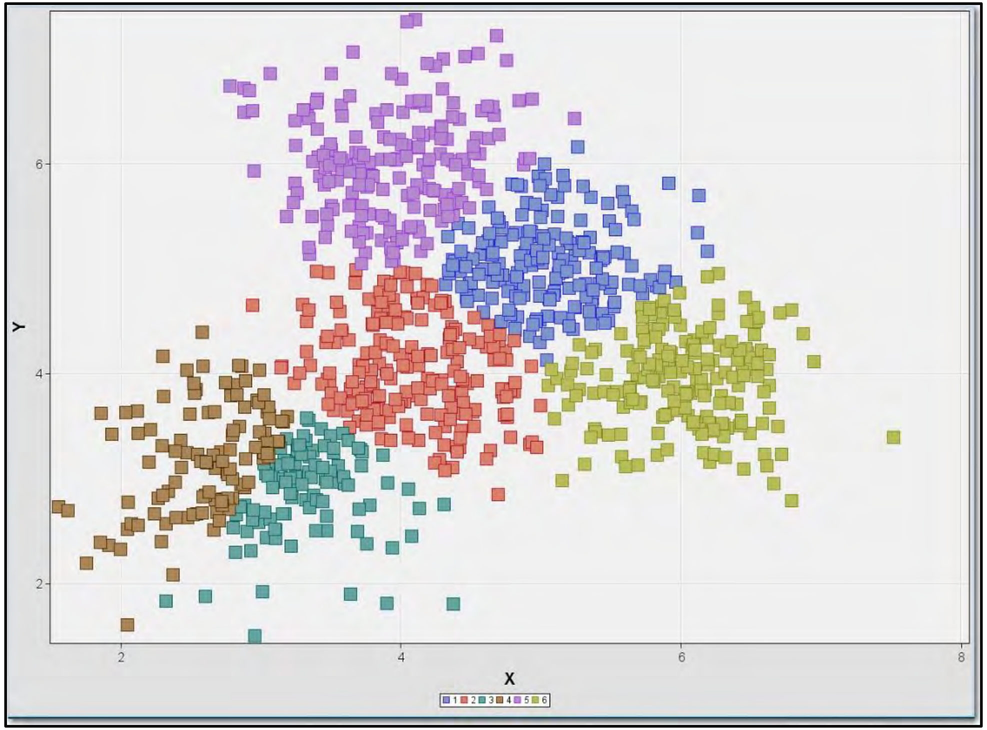
Centroid / Full = 6 clusters



Centroid / Princomp =   5 clusters



Centroid / Partial =   6 clusters



How Many Clusters?

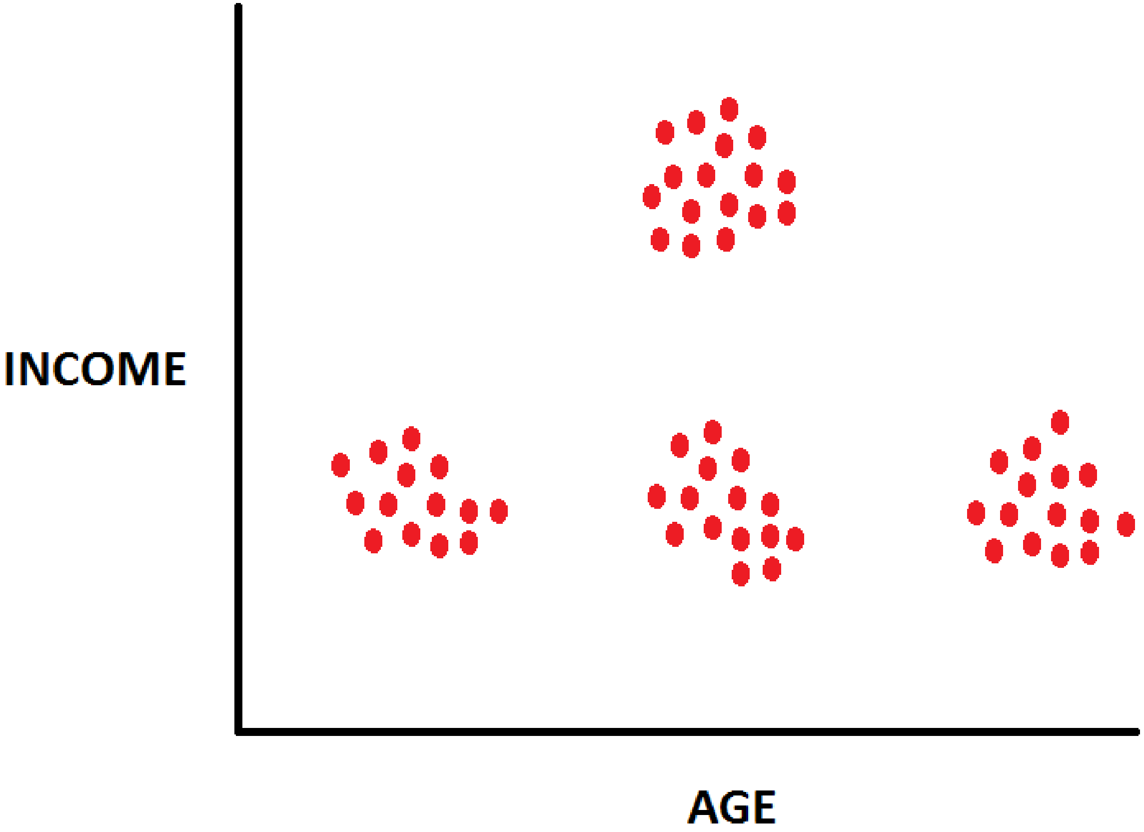
Cluster	Ward	Average	Centroid
First	7	5	4
MacQueen	3	5	5
Full	5	7	6
Princomp	5	5	5
Partial	3	5	6

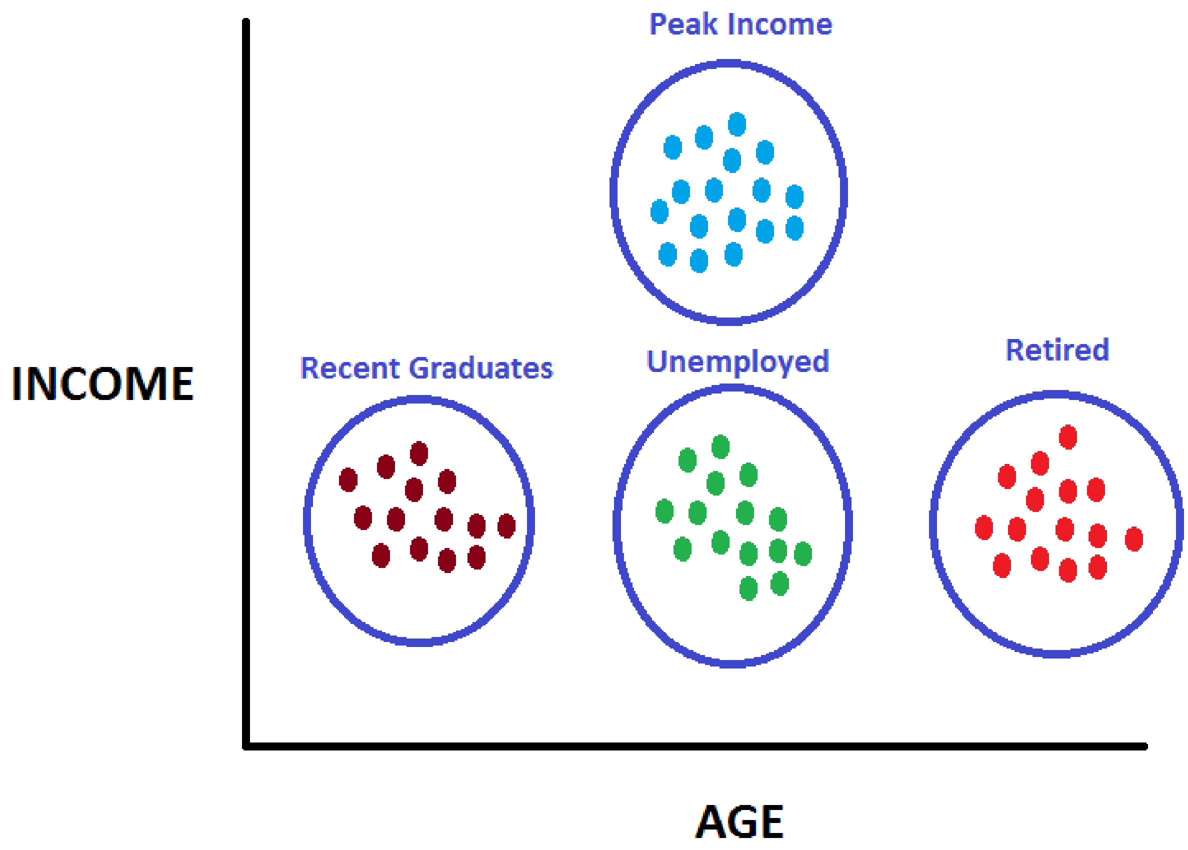
How Many Clusters?

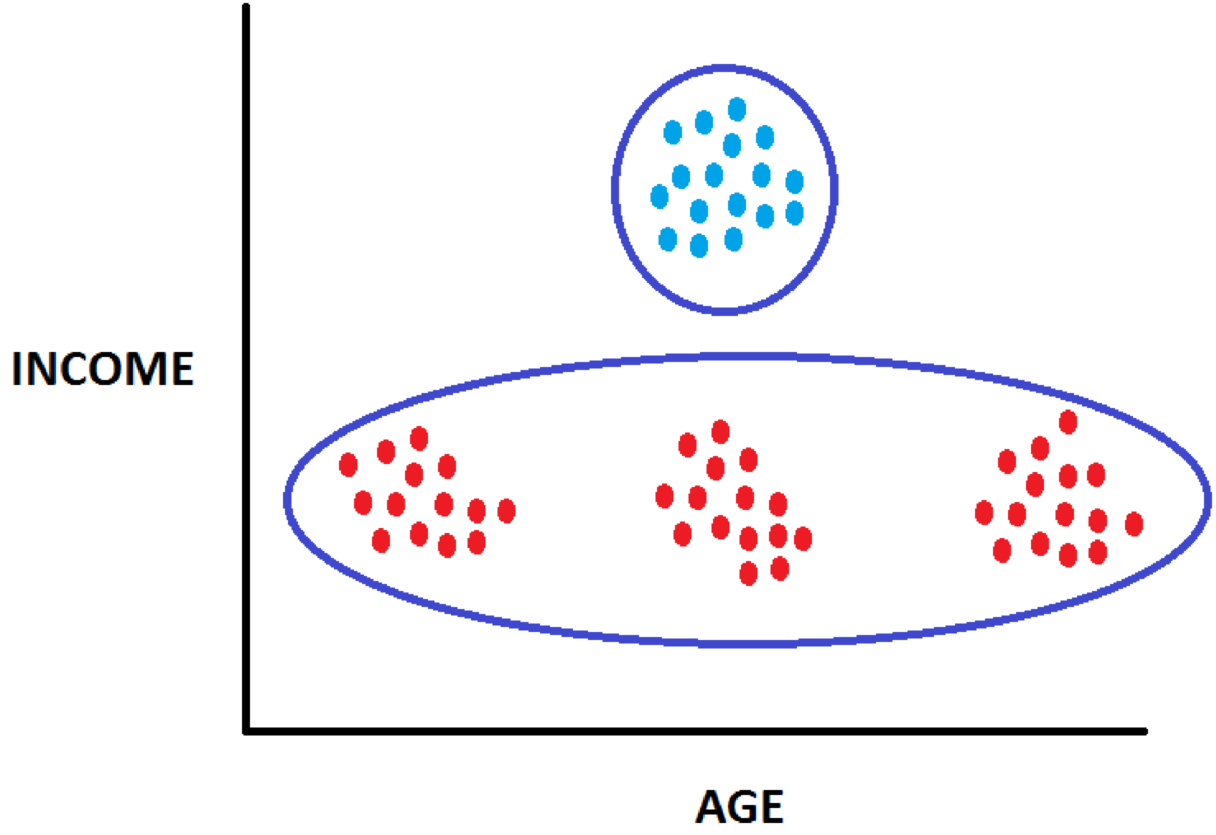
Number of Cluster	Count
3 clusters	2
4 clusters	1
5 clusters	8
6 clusters	2
7 clusters	2

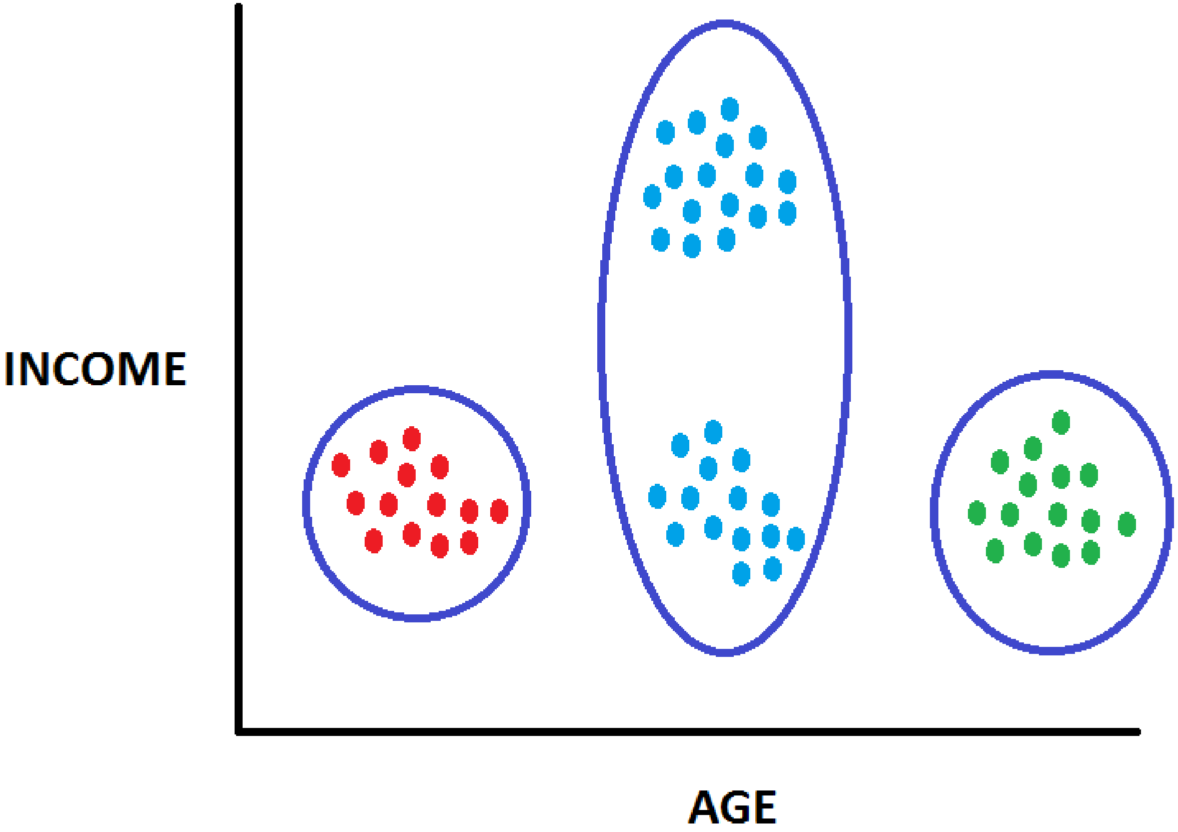
How Many Clusters?

Number of Cluster	Count
3 clusters	2
4 clusters	1
5 clusters	8
6 clusters	2
7 clusters	2


INCOME
AGE


Peak Income
INCOME
Recent Graduates
Unemployed
Retired
AGE


INCOME
AGE


INCOME
AGE

How Many Clusters?

Number of Cluster	Count
3 clusters	2
4 clusters	1
5 clusters	8
6 clusters	2
7 clusters	2

How Many Clusters?

The Number of Clusters Found Depends Upon

- Cluster Starting Points
- Clustering Method

Certain Numbers occur more frequently than others

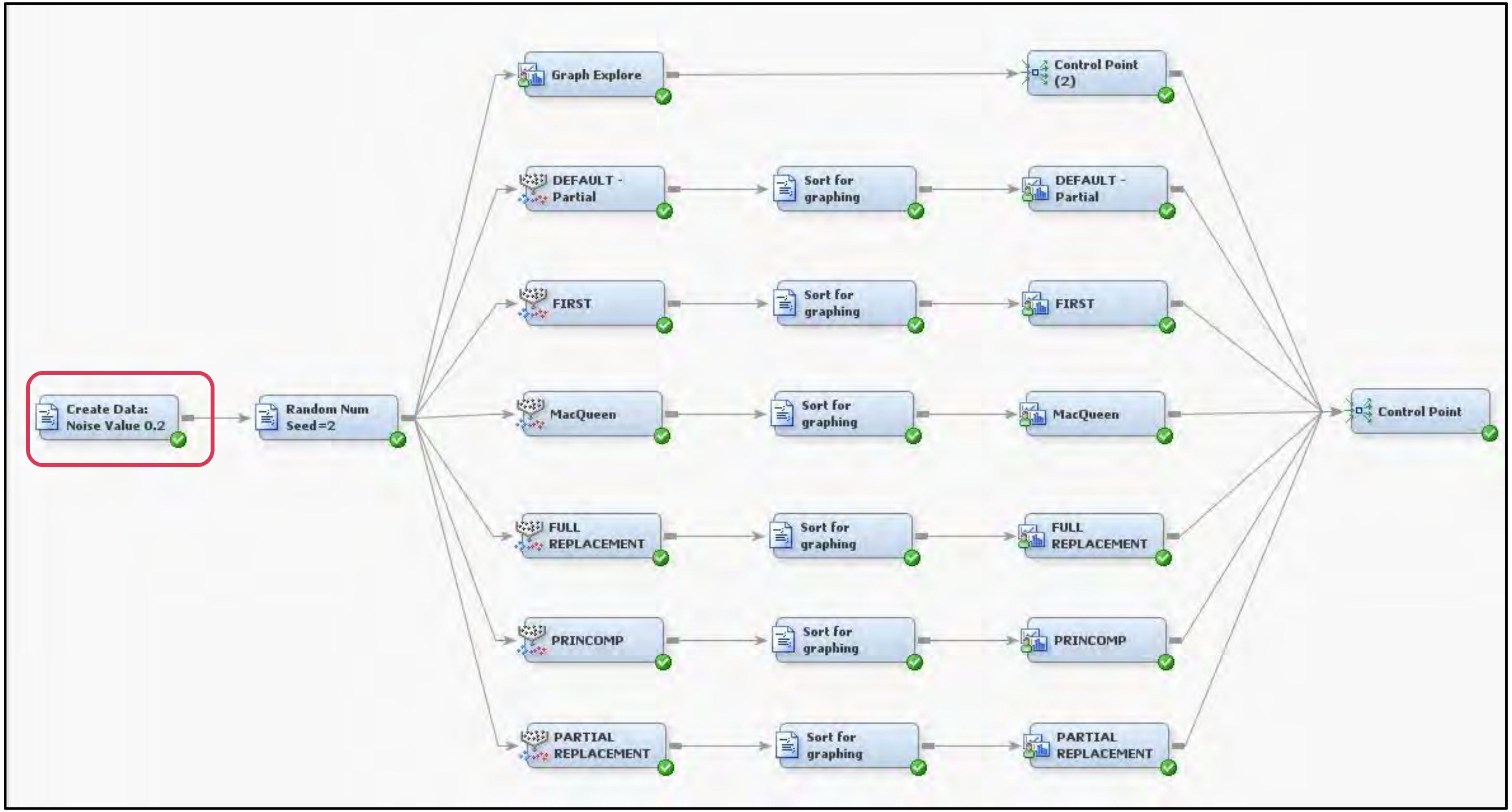
- Trial and Error suggests 3 to 7 Clusters
- Probably 5 Clusters is optimal

Starting Points Affect Clusters

"Your Mileage May Vary"

Diagram 4400

Different Seed Selection Methods: Diagram 4400



Random Seeds – Synthetic Data

```
%let COUNT                = 200
%let WEIGHT               = 0.2;
%let SEED                 = 1;

%let INFILE               = INFILE;
%let OUTFILE              = RANDOM_DATA;

data &INFILE.;
do I = 1 to &COUNT.;
        X                 = 3.0;
        Y                 = 3.0;
        NOISE_X           = rannor(&SEED.);
        NOISE_Y           = rannor(&SEED.);
        output;
        X                 = 5.0;
        Y                 = 5.0;
        NOISE_X           = rannor(&SEED.);
        NOISE_Y           = rannor(&SEED.);
        output;
        X                 = 4.0;
        Y                 = 6.0;
        NOISE_X           = rannor(&SEED.);
        NOISE_Y           = rannor(&SEED.);
        output;
        X                 = 6.0;
        Y                 = 4.0;
        NOISE_X           = rannor(&SEED.);
        NOISE_Y           = rannor(&SEED.);
        output;
        X                 = 4.0;
        Y                 = 4.0;
        NOISE_X           = rannor(&SEED.);
        NOISE_Y           = rannor(&SEED.);
        output;
end;
drop I;
run;

data &OUTFILE.;
set &INFILE.;
        X                 = X + &WEIGHT.*NOISE_X;
        Y                 = Y + &WEIGHT.*NOISE_Y;
drop NOISE_X;
drop NOISE_Y;
run;
```

Random Seeds – Synthetic Data

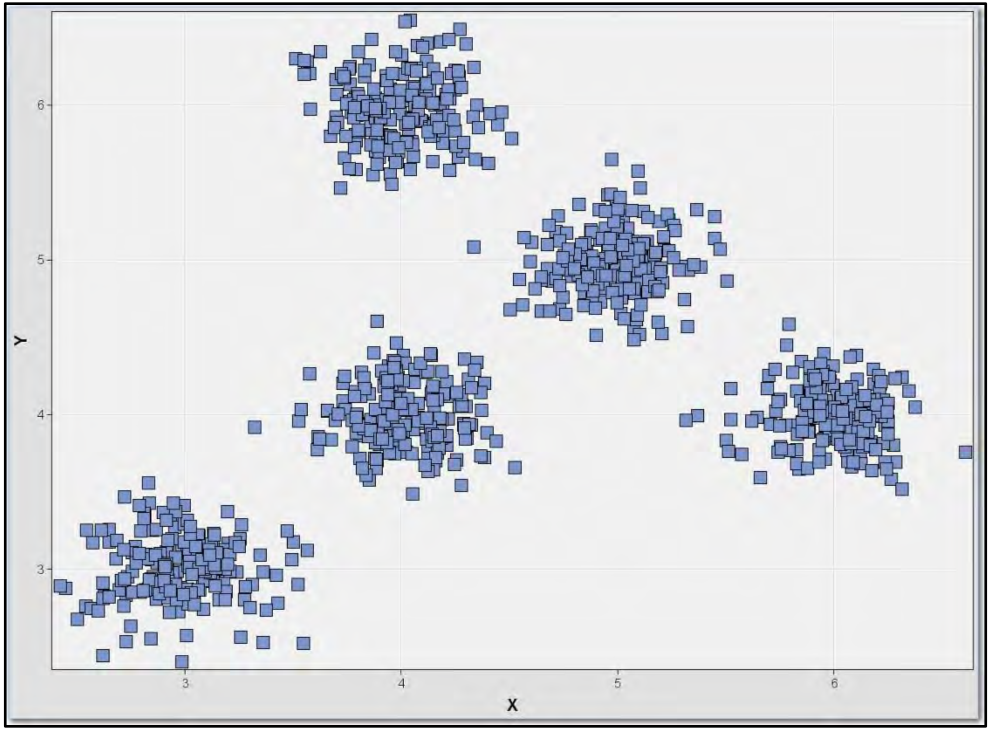
```
%let COUNT             = 200
%let WEIGHT            = 0.2;
%let SEED              = 1;

%let INFILE            = INFILE;
%let OUTFILE           = RANDOM_DATA;

data &INFILE.;
do I = 1 to &COUNT.;
        X              = 3.0;
        Y              = 3.0;
        NOISE_X        = rannor(&SEED.);
        NOISE_Y        = rannor(&SEED.);
        output;
        X              = 5.0;
        Y              = 5.0;
        NOISE_X        = rannor(&SEED.);
        NOISE_Y        = rannor(&SEED.);
        output;
        X              = 4.0;
        Y              = 6.0;
        NOISE_X        = rannor(&SEED.);
        NOISE_Y        = rannor(&SEED.);
        output;
        X              = 6.0;
        Y              = 4.0;
        NOISE_X        = rannor(&SEED.);
        NOISE_Y        = rannor(&SEED.);
        output;
        X              = 4.0;
        Y              = 4.0;
        NOISE_X        = rannor(&SEED.);
        NOISE_Y        = rannor(&SEED.);
        output;
end;
drop I;
run;

data &OUTFILE.;
set &INFILE.;
        X              = X + &WEIGHT.*NOISE_X;
        Y              = Y + &WEIGHT.*NOISE_Y;
drop NOISE_X;
drop NOISE_Y;
run;
```

Changed “Noise Weight”

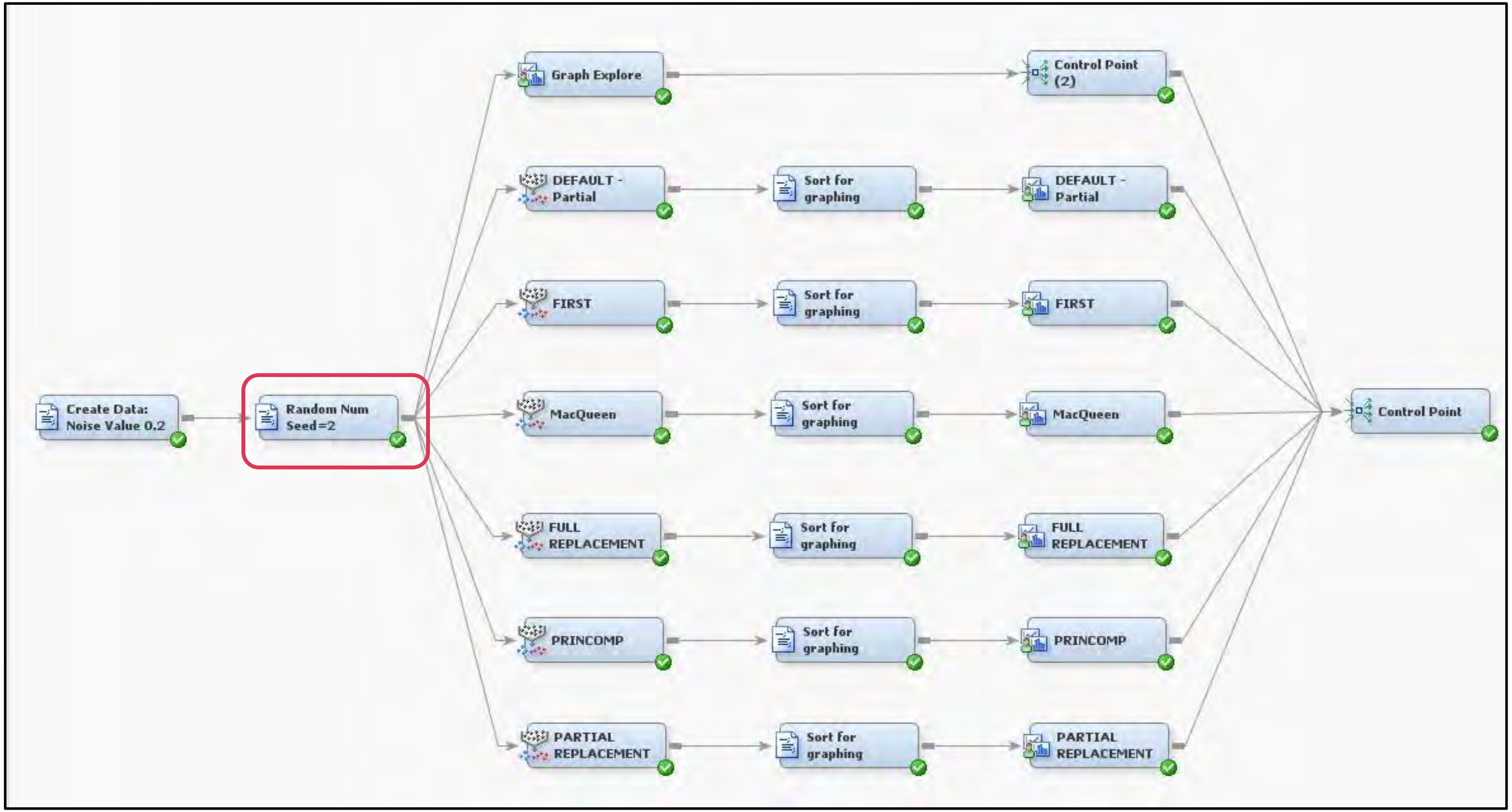
- From 0.5
- To 0.2

Less “Noise” was introduced.
Clusters will be more “defined”

Different Seed Selection Methods



Different Seed Selection Methods



Random Seeds – Shuffle Cards

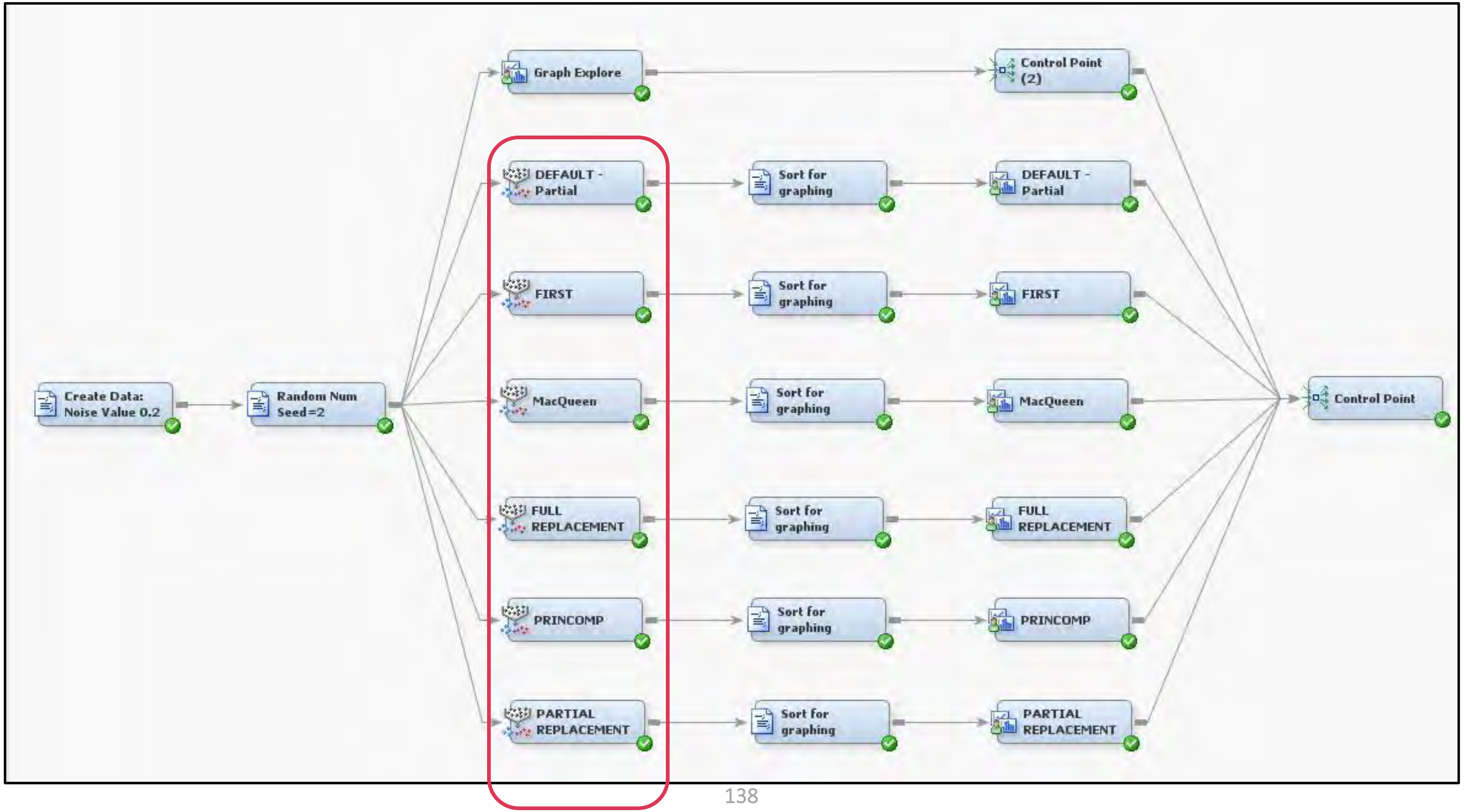
```
%let SEED           = 2;

%let INFILE         = RANDOM_DATA;
%let TEMPFILE       = TEMPFILE;
%let OUTFILE        = SORTED_DATA;

data &TEMPFILE.;
set &INFILE.;
SORT = ranuni( &SEED. );
run;

proc sort data=&TEMPFILE.;
by SORT;
run;

data &OUTFILE.;
set &TEMPFILE.;
drop SORT;
run;

proc print data=&OUTFILE.(obs=5);
run;
```

Random Seeds – Shuffle Cards

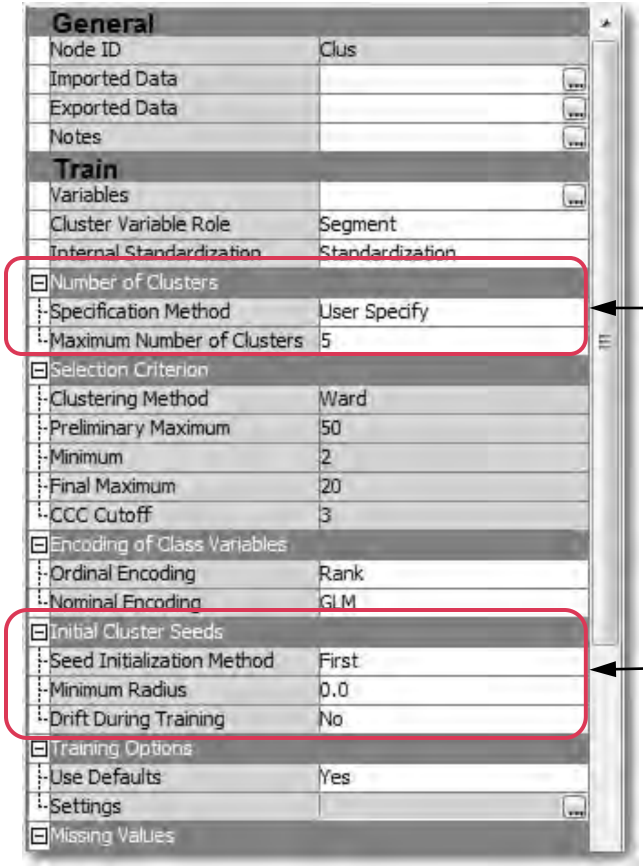
```
%let SEED             = 2;

%let INFILE           = RANDOM_DATA;
%let TEMPFILE         = TEMPFILE;
%let OUTFILE          = SORTED_DATA;

data &TEMPFILE.;
set &INFILE.;
SORT = ranuni( &SEED. );
run;

proc sort data=&TEMPFILE.;
by SORT;
run;

data &OUTFILE.;
set &TEMPFILE.;
drop SORT;
run;

proc print data=&OUTFILE.(obs=5);
run;
```

Random Number Seed:
Changing this value will cause the list of data points to be put in a different order (“shuffled”)

Different Seed Selection Methods



Different Seed Selection Methods

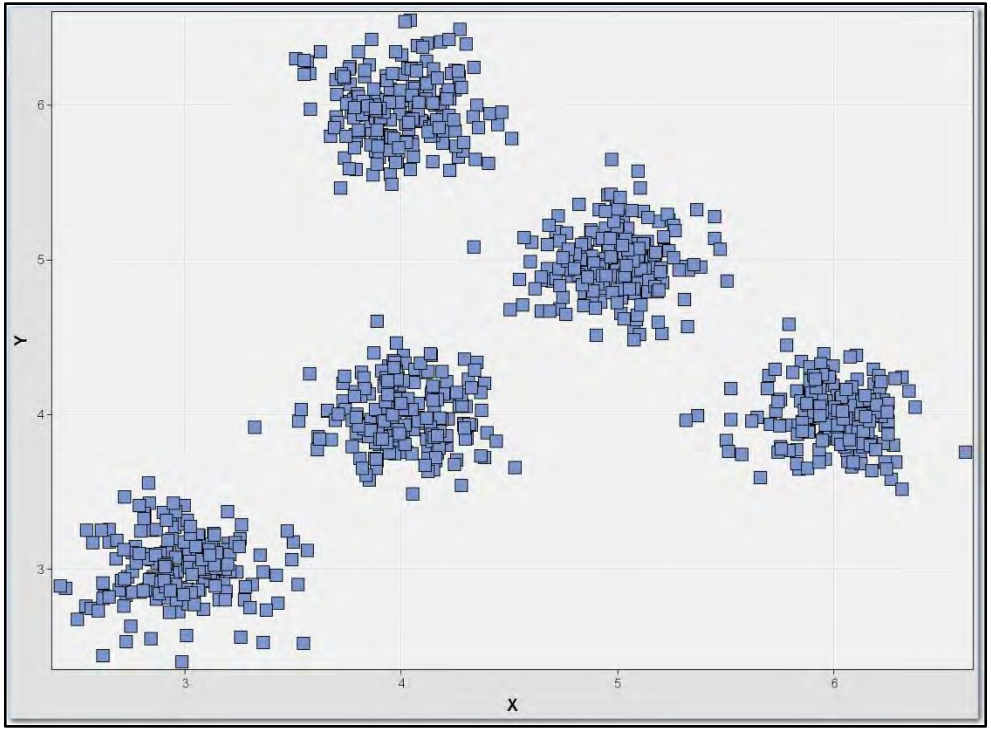


Set the numbers to exactly 5 clusters

Use first 5 data points as cluster seeds.

- Repeat for “Partial”
- Repeat for “Full”
- Repeat for “MacQueen”
- Repeat for “Princomp”

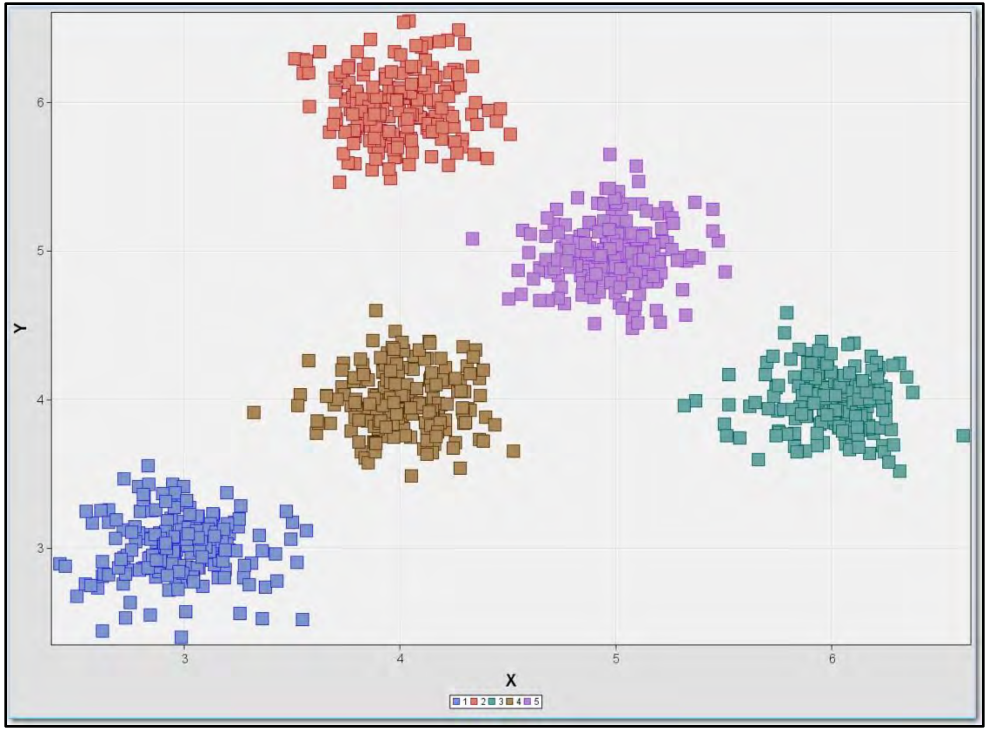
Different Seed Selection Methods



First “N” Selection Method

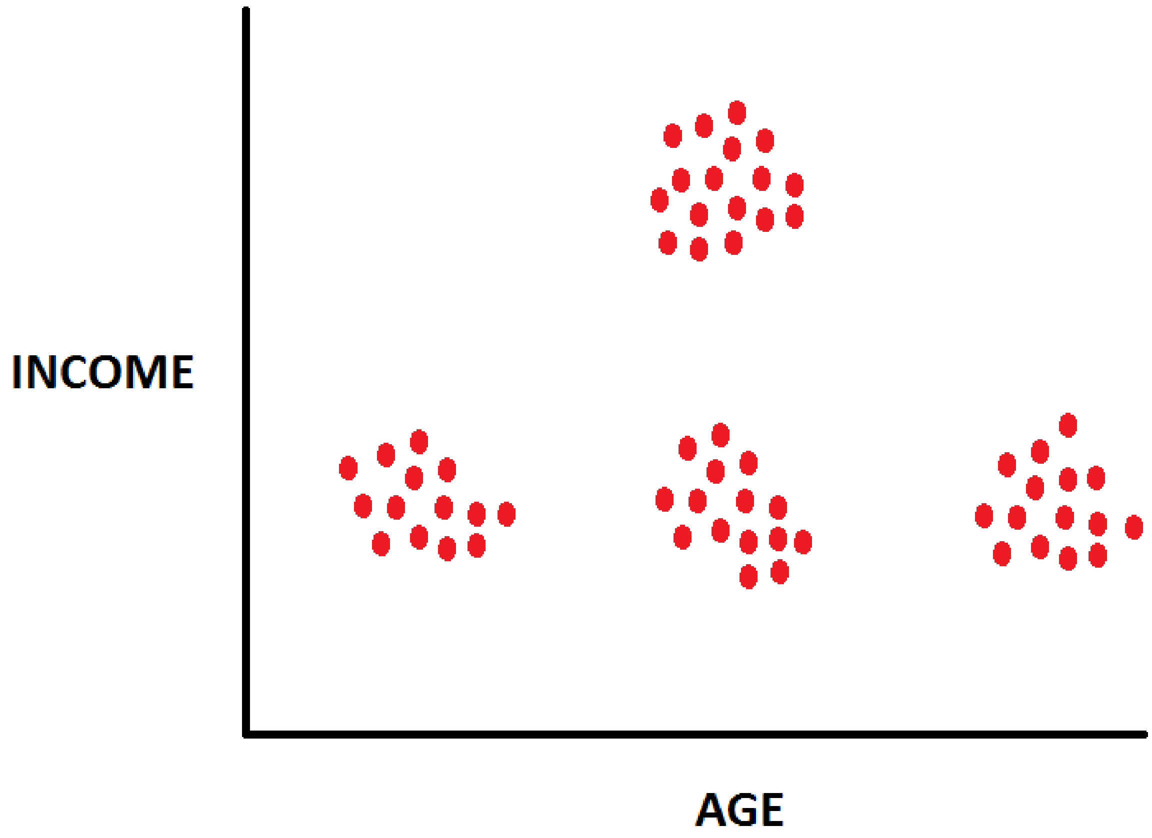
MacQueen Selection Method

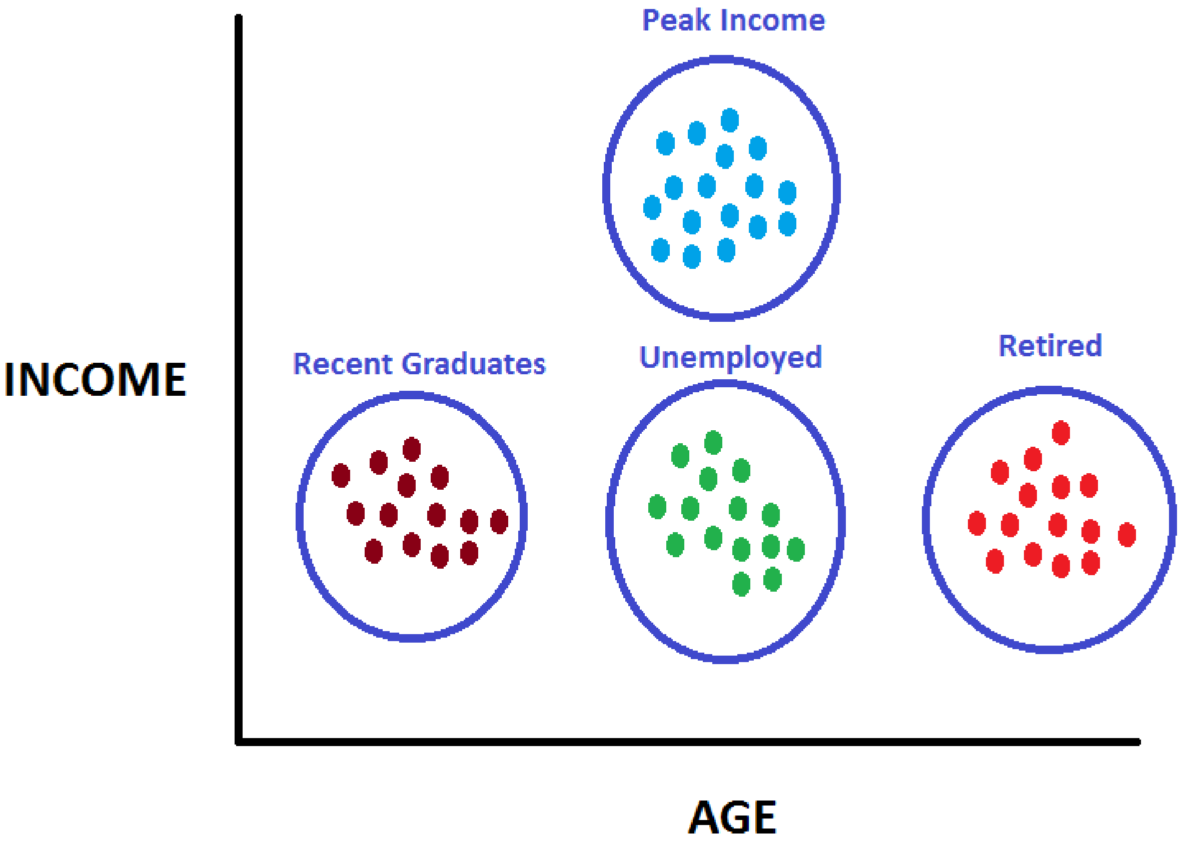
Full Selection Method

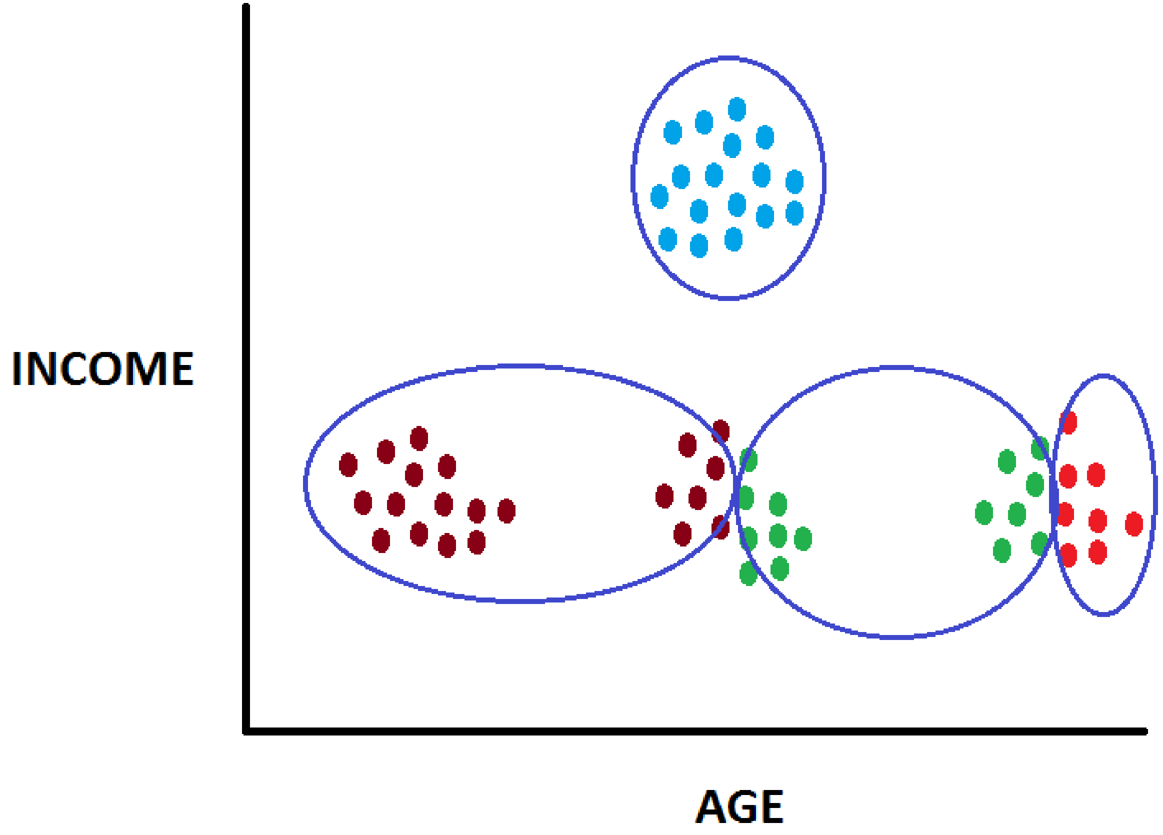


Partial Selection Method

Princomp Selection Method


INCOME
AGE


Peak Income
INCOME
Recent Graduates
Unemployed
Retired
AGE


INCOME
AGE

What Are The Cluster Centers?

Different Starting Points and Settings Can Yield Different Results

- Occasionally sub-optimal clusters are found
- Usually the same optimal clusters are found regardless of starting points and settings

Five different settings

- 2 of 5 have sub optimal Clusters
- 3 of 5 have optimal cluster
  - Even sub-optimal Clusters have some similarity to optimal clusters

Monte Carlo Clustering

Monte Carlo Macros

Monte Carlo Clustering

Cluster data repeatedly:

- Use different methods for determining starting points
- Use different clustering methods

After each clustering algorithm finishes:

- After each iteration, record the number of clusters
- After each iteration, record the cluster centers

After numerous iterations:

- Determine the correct number of clusters
- Cluster the "Cluster Centers"

SAS Macro: Sleep

- Macro will cause the SAS Program to “sleep” for a specified number of seconds.

- This gives the operating system time to write files to disk and prevents deadlocks.

Parameters

```
%SLEEP( HOWLONG );
```

- HowLong : How many seconds should the program “sleep”

SAS Macro: Sleep

```
%macro SLEEP( HOWLONG );

data;
time_slept=sleep(&HOWLONG.,1);
run;

%mend;
```

SAS Macro: Save_Cluster_Info

- Stores the number of clusters and the cluster centers found by
  - SAS Enterprise Miner Cluster Node
  - SAS Enterprise Miner SOM/Kohonen Node
- Results are collected from Enterprise Miner nodes and appended to SAS data files
- Clusters with rare membership are deleted

SAS Macro: Save_Cluster_Info

```
%SAVE_CLUSTER_INFO( CENTERFILE,
                    OUTFILE_CENTERS,
                    OUTFILE_HOWMANY,
                    TEMPFILE          = TEMPFILE,
                    HOWMANY           = _HOWMANY_,
                    CUTOFFPCT         = 0.1,
                    HOWLONG           = 1 );
```

CENTERFILE	: Cluster Centers from SAS Enterprise Miner
OUTFILE_CENTERS	: File to store the Cluster Centers
OUTFILE_HOWMANY	: File to store the number of Cluster Centers
TEMPFILE	: Temporary File to hold data
HOWMANY	: Name of the variable that will store the number of clusters
CUTOFFPCT	: If a clusters has less than this percent of the records, delete it
HOWLONG	: How many seconds to sleep between functions

SAS Macro: Save_Cluster_Info

Sample Run: First run found “7” Clusters

How Many Clusters File:

Obs	_HOWMANY_
1	7

Cluster Center File:

Obs	_HOWMANY_	X	Y
1	7	3.03363	2.74416
2	7	4.16234	3.71581
3	7	6.03689	3.95565
4	7	4.00574	4.82218
5	7	2.71785	3.49664
6	7	3.96880	6.14890
7	7	5.12917	5.06041

SAS Macro: Save_Cluster_Info

Sample Run: Second Run found "5" Clusters

How Many Clusters File:

Obs	_HOWMANY_
1	7
2	5

Cluster Center File:

Obs	_HOWMANY_	X	Y
1	7	3.03363	2.74416
2	7	4.16234	3.71581
3	7	6.03689	3.95565
4	7	4.00574	4.82218
5	7	2.71785	3.49664
6	7	3.96880	6.14890
7	7	5.12917	5.06041
8	5	2.90993	3.03597
9	5	4.06764	3.98039
10	5	6.01687	3.95821
11	5	5.02613	5.04958
12	5	3.94580	6.00564

SAS Macro: Save_Cluster_Info

Sample Run: Third Run found “5” Clusters

How Many Clusters File:

Obs	_HOWMANY_
1	7
2	5
3	5

Cluster Center File:

Obs	_HOWMANY_	X	Y
1	7	3.03363	2.74416
2	7	4.16234	3.71581
3	7	6.03689	3.95565
4	7	4.00574	4.82218
5	7	2.71785	3.49664
6	7	3.96880	6.14890
7	7	5.12917	5.06041
8	5	2.90993	3.03597
9	5	4.06764	3.98039
10	5	6.01687	3.95821
11	5	5.02613	5.04958
12	5	3.94580	6.00564
13	5	6.02094	3.95189
14	5	3.95058	6.00808
15	5	5.05562	5.05453
16	5	4.07909	4.01951
17	5	2.92715	3.03838

SAS Macro: Save_Cluster_Info (page 1 of 3)

```
%macro CLUSTER_SLEEP( HOWLONG );
        data;
        time_slept=sleep(&HOWLONG.,1);
        run;
%mend;


%macro SAVE_CLUSTER_INFO( CENTERFILE,
                          OUTFILE_CENTERS,
                          OUTFILE_HOWMANY,
                          TEMPFILE          = TEMPFILE,
                          HOWMANY           = _HOWMANY_,
                          CUTOFFPCT         = 0.1,
                          HOWLONG           = 1 );


data &TEMPFILE.;
set &CENTERFILE.;
drop _RADIUS_;
drop _CRIT_ _XCONV_ _FCONV_ _RMSSTD_ _NEAR_ _GAP_ _SEGMENT_;
drop _CRIT_ _XCONV_ _FCONV_ SOM_SEGMENT _RMSSTD_ _NEAR_ _GAP_
SOM_DIMENSION1 SOM_DIMENSION2 SOM_ID;
run;
```

SAS Macro: Save_Cluster_Info (page 2 of 3)

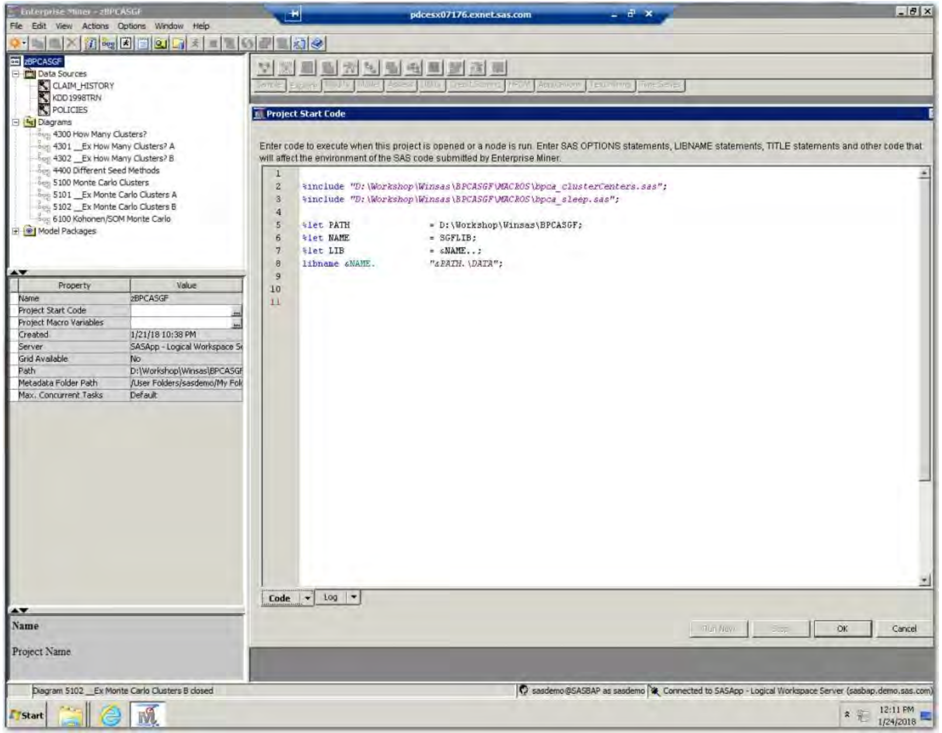
```
data;
set &TEMPFILE.;
retain &HOWMANY.;
if _N_ = 1 then &HOWMANY. = 0;
&HOWMANY. = &HOWMANY. + _FREQ_;
call symput("HOWMANYCOUNT", &HOWMANY. );
run;

data &TEMPFILE.;
set &TEMPFILE.;
if _FREQ_ / &HOWMANYCOUNT. * 100 < &CUTOFFPCT. then delete;
run;

data;
set &TEMPFILE.;
retain &HOWMANY.;
if _N_ = 1 then &HOWMANY. = 0;
&HOWMANY. = &HOWMANY. + 1;
call symput("HOWMANYCOUNT", &HOWMANY. );
run;

data &TEMPFILE.;
length &HOWMANY. 8.;
set &TEMPFILE.;
&HOWMANY. = &HOWMANYCOUNT.;
drop _FREQ_;
run;
```

SAS Macro: Save_Cluster_Info (page 3 of 3)

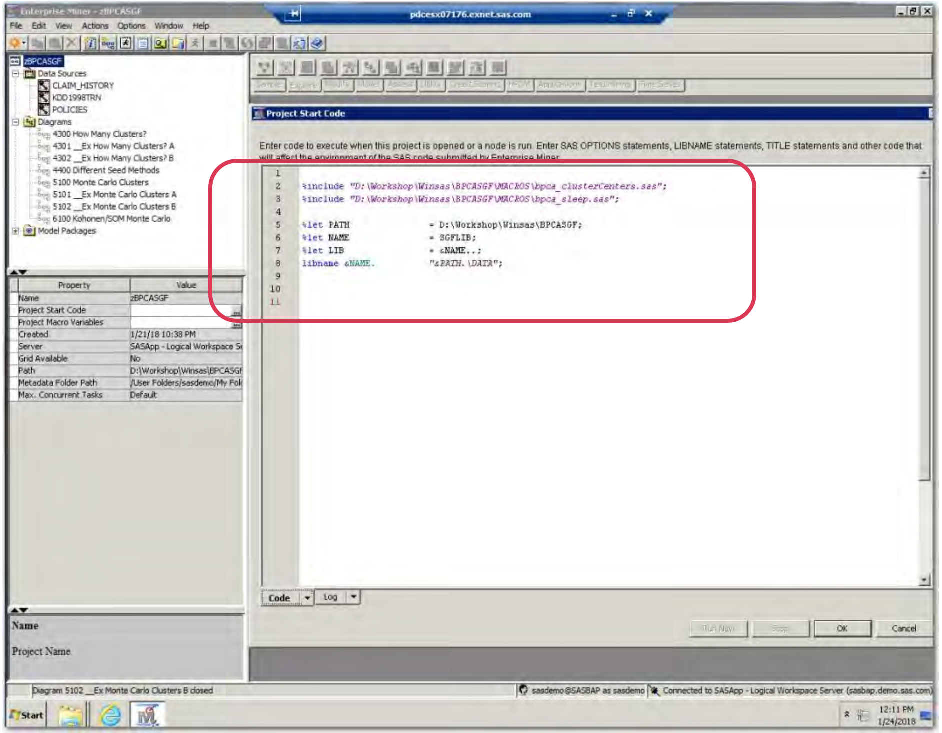
```
%cluster_sleep(&HOWLONG.);

proc append data=&TEMPFILE. out=&OUTFILE_CENTERS. force;
run;

%cluster_sleep(&HOWLONG.);

data &TEMPFILE.;
set &TEMPFILE.(obs=1);
keep &HOWMANY.;
run;

%cluster_sleep(&HOWLONG.);

proc append data=&TEMPFILE. out=&OUTFILE_HOWMANY. force;
run;

%cluster_sleep(&HOWLONG.);

%mend;
```

%include the MACRO

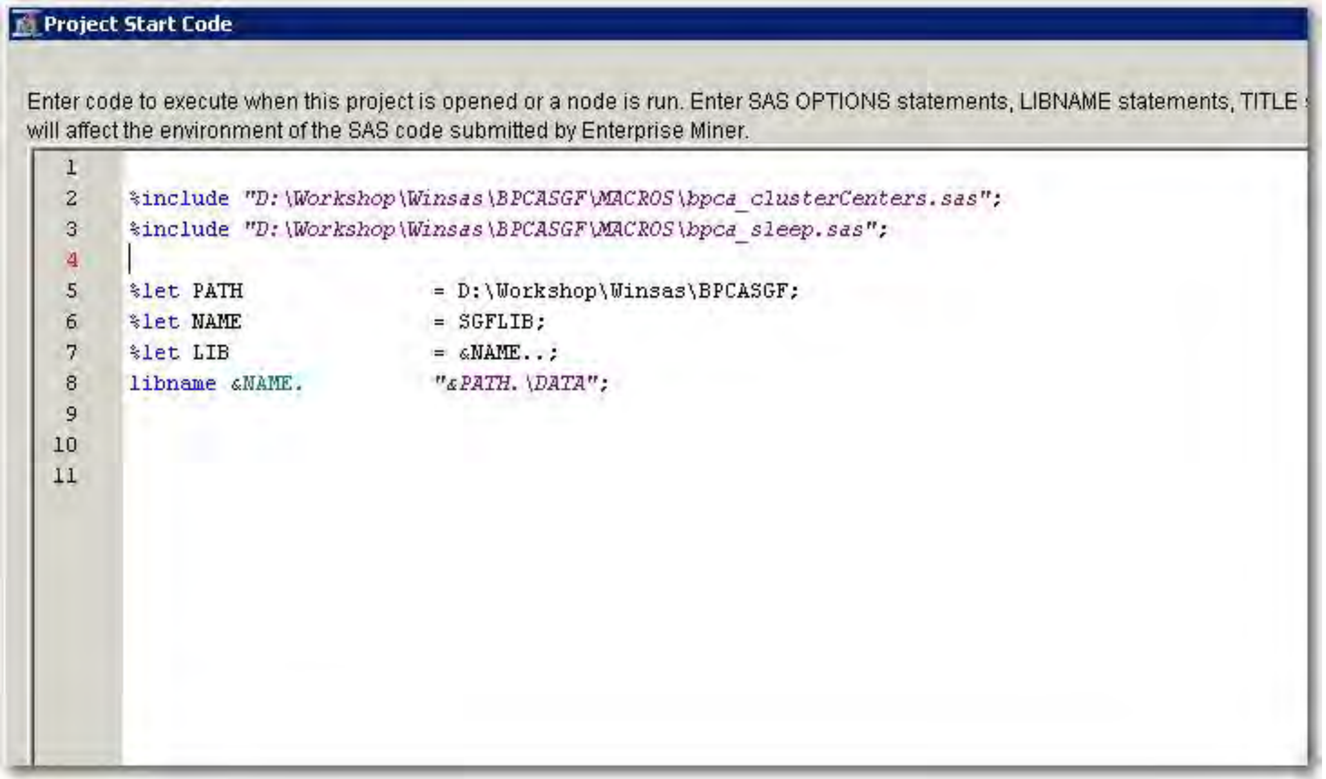
SAS Enterprise Miner
Project Start Code



SAS Enterprise Miner
Project Start Code



SAS Enterprise Miner
Project Start Code



SAS Enterprise Miner
Project Start Code

Project Start Code

Enter code to execute when this project is opened or a node is run. Enter SAS OPTIONS statements, LIBNAME statements, TITLE s
will affect the environment of the SAS code submitted by Enterprise Miner.

```

%include "D:\Workshop\Winsas\BPCASGF\MACROS\bpca_clusterCenters.sas";
%include "D:\Workshop\Winsas\BPCASGF\MACROS\bpca_sleep.sas";

%let PATH            = D:\Workshop\Winsas\BPCASGF;
%let NAME            = SGFLIB;
%let LIB             = &NAME..;
libname &NAME.       "&PATH.\DATA";
```

EXAMPLE-Using the Macro: Diagram 5100

Cluster Node Data Collection

1. Use same Synthetic Data Program as Example 3 of Lecture 4. The Noise factor is set to 0.5
    - 200 points centered at (3,3)
    - 200 points centered at (5,5)
    - 200 points centered at (4,6)
    - 200 points centered at (6,4)
    - 200 points centered at (4,4)
2. Use same "shuffle program" use SEED = -1
    - A value of -1 causes the computer clock to be used as a "seed".
    - This results in a different random seed being used every time the program is executed.

Cluster Node Data Collection

3. Cluster Node Settings

- Ward Clustering (but any method will do)
- Partial Replacement Cluster Seed (but any method will do)
- Automatic Cluster Selection
  - Max 7: (Maximum Value from Example 3 of Lecture 4)
  - Min 3: (Minimum Value from Example 3 of Lecture 4)

4. Save the Cluster Centers using the Save_Cluster_Info Macro.

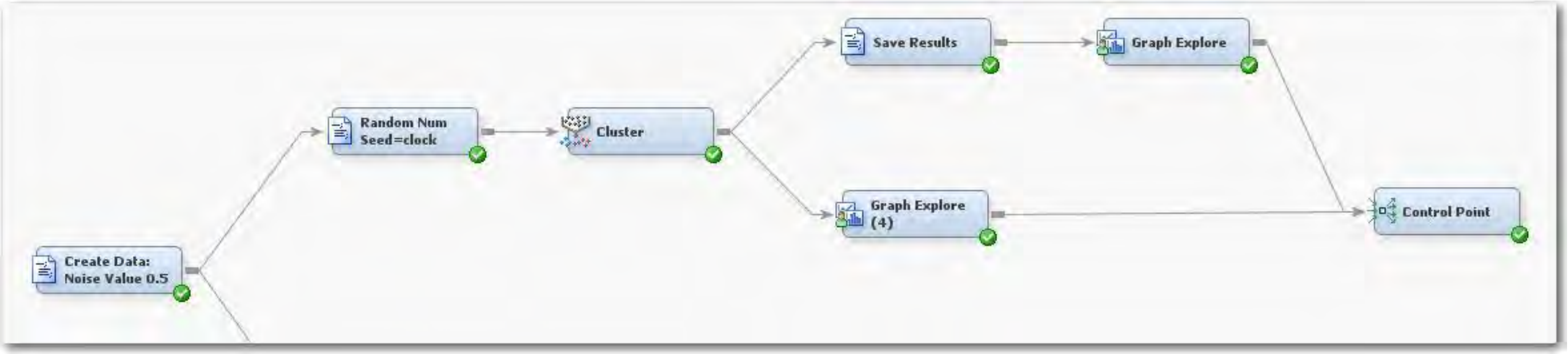
Cluster Node Data Collection

5. Rerun Numerous Times

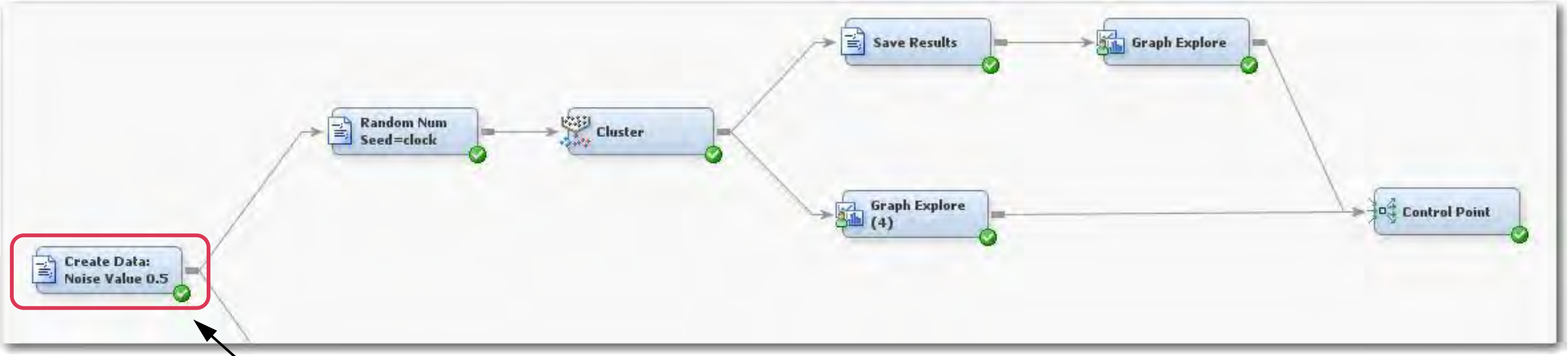
- Shuffle
- Cluster
- Save_Cluster_Info

6. Cluster the Clusters Centers

Cluster Node Data Collection
Enterprise Miner Diagram

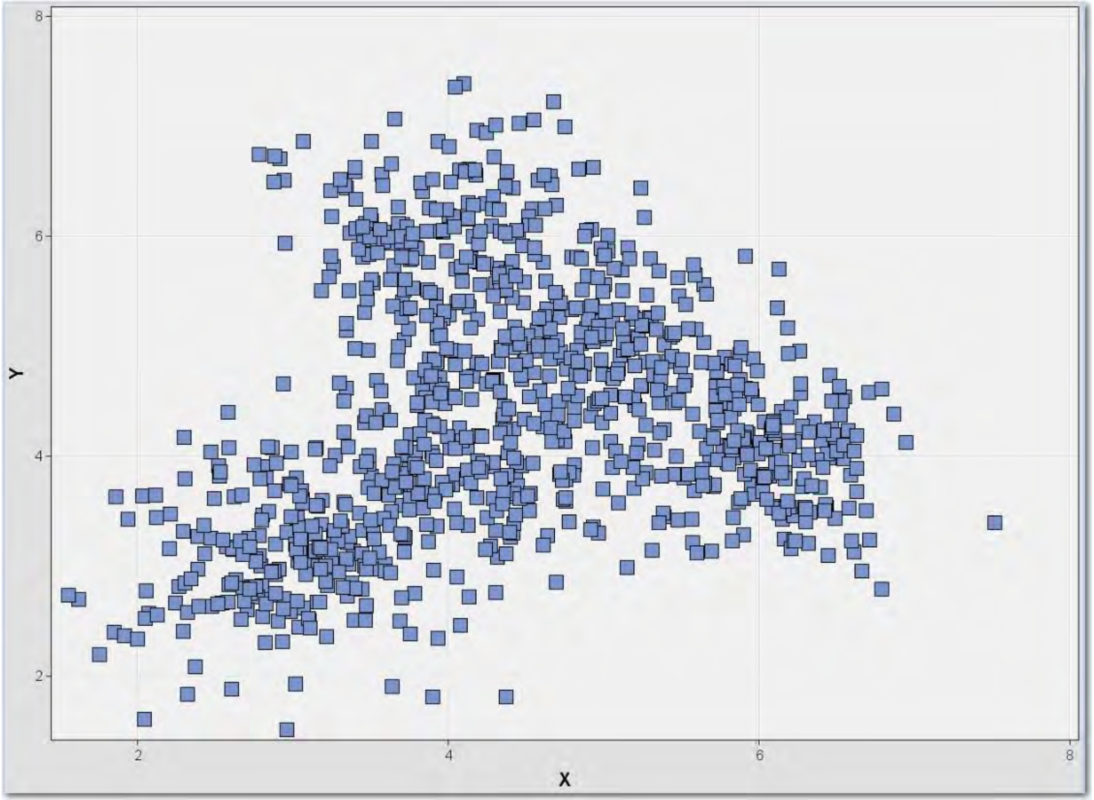


Cluster Node Data Collection Enterprise Miner Diagram

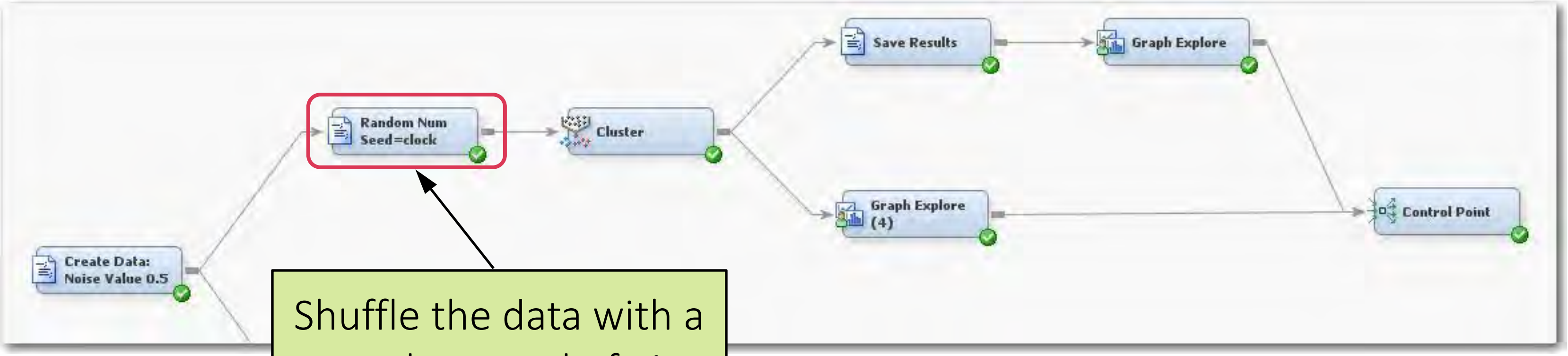


Create synthetic data with the noise factor of 0.5

Cluster Node Data Collection
Enterprise Miner Diagram

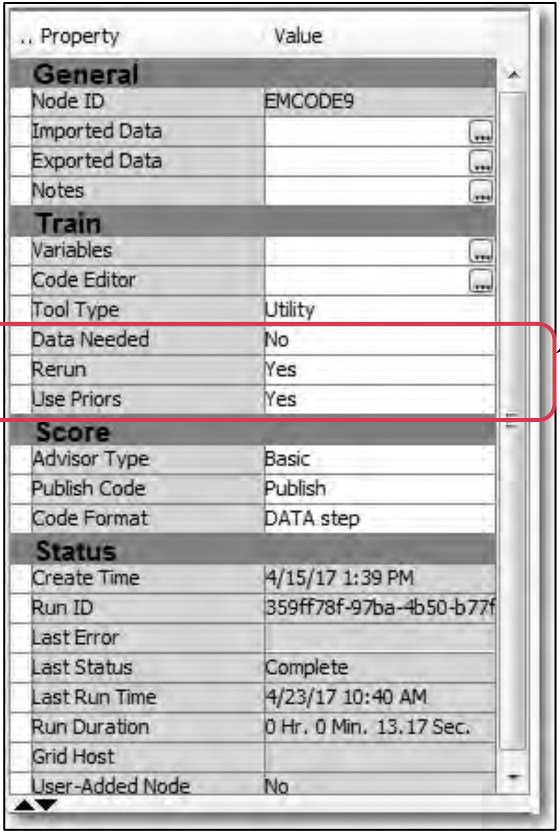


Cluster Node Data Collection
Enterprise Miner Diagram



Shuffle the data with a random seed of -1 (tied to clock)

Cluster Node Data Collection
Enterprise Miner Diagram



Set the “Rerun” to “Yes” so that this code node will rerun every time and will reshuffle the data.

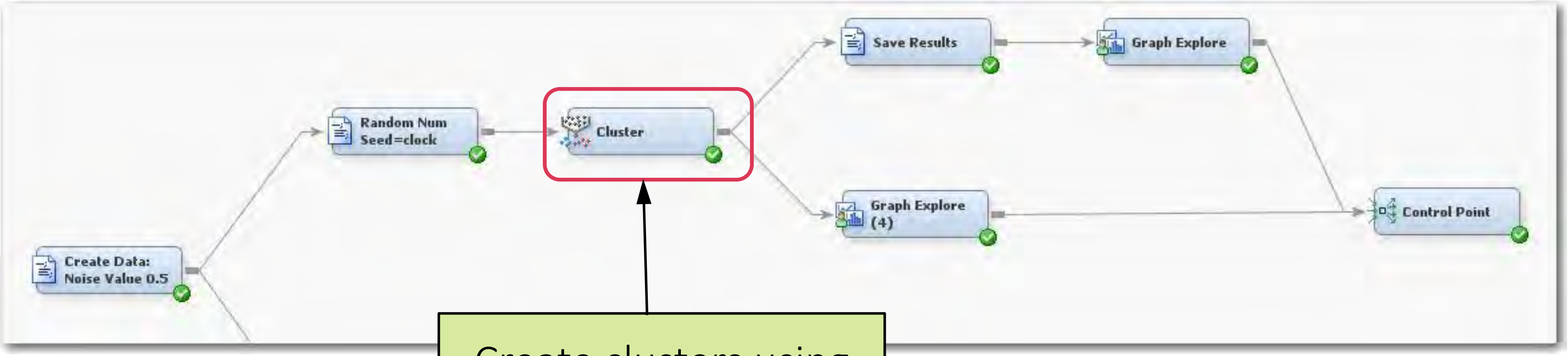
Cluster Node Data Collection
Enterprise Miner Diagram

```
%let SEED                = -1;

%let INFILE              = &EM_IMPORT_DATA.;
%let TEMPFILE            = TEMPFILE;
%let OUTFILE             = SORTED_DATA;

data &TEMPFILE.;
set &INFILE.;
SORT = ranuni( &SEED. );
run;

proc sort data=&TEMPFILE.;
by SORT;
run;

data &OUTFILE.;
set &TEMPFILE.;
drop SORT;
run;

proc print data=&OUTFILE.(obs=5);
run;
```

Random Number Seed is set to -1. This ties the random number seed to the clock.

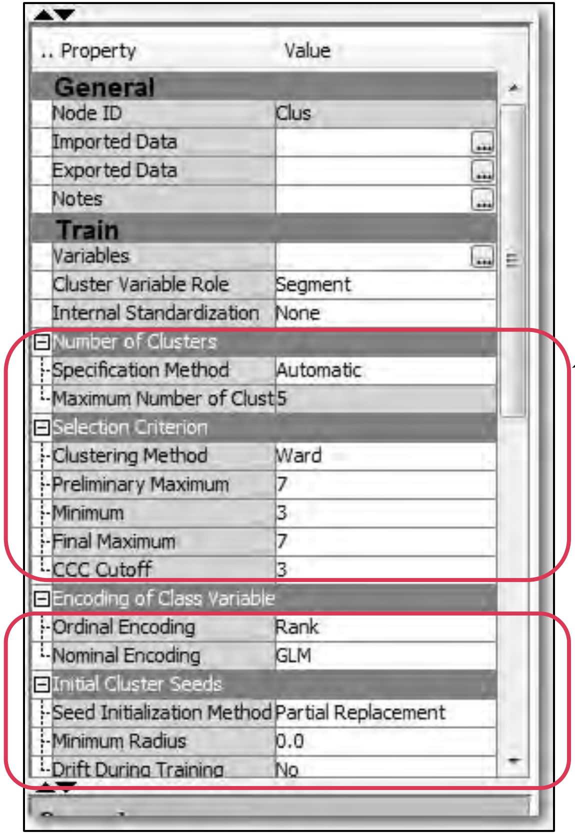
Every time this program is executed, the data will be in a different order.

Cluster Node Data Collection Enterprise Miner Diagram



Create clusters using the data points that were shuffled in the previous node.

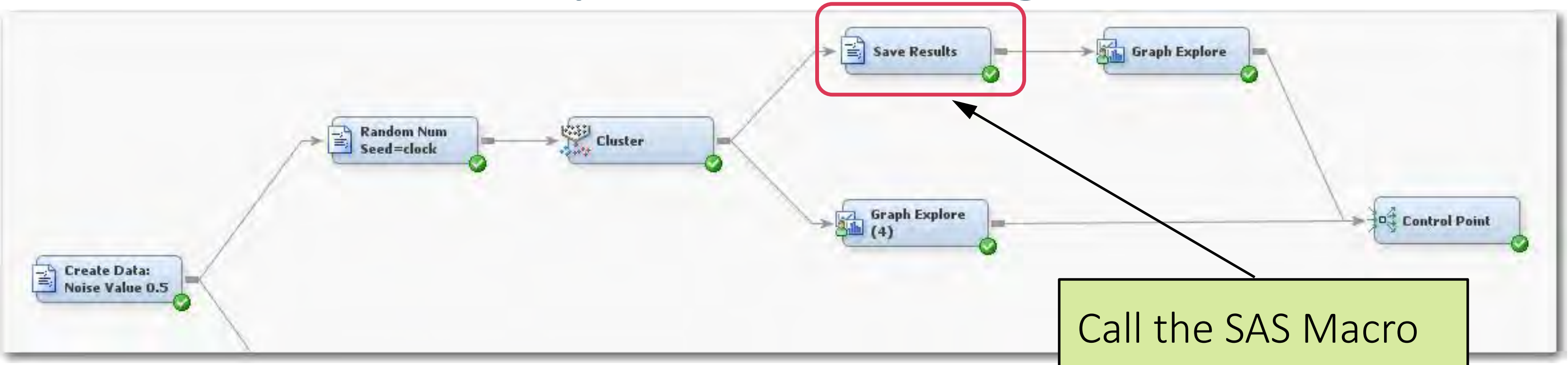
Cluster Node Data Collection Enterprise Miner Diagram



Number of clusters is set to “Automatic”. The MAX is set to “7” and the MIN is set to “3” because that was the range found in Lecture 4 Example 3.
The Clustering Method is set to “Ward”, but “Average” or “Centroid could also be used.

Seed Intitialization is set to “Partial Replacement” but other methods could be used.

Cluster Node Data Collection
Enterprise Miner Diagram



Call the SAS Macro "Save_Cluster_Info" in order to save the results from the cluster node.

Cluster Node Data Collection
Enterprise Miner Diagram

```
%let INFILE          = &EM_IMPORT_CLUSMEAN.;

%let CENTERFILE      = SGFLIB.y5100_CENTERFILE;
%let HOWMANYFILE     = SGFLIB.y5100_HOWMANYFILE;

proc print data=&INFILE.;
run;

%save_cluster_info( &INFILE., &CENTERFILE., &HOWMANYFILE. );

proc print data=&CENTERFILE.(obs=30);
run;

proc print data=&HOWMANYFILE.(obs=10);
run;

proc freq data=&HOWMANYFILE.;
table _HOWMANY_ /missing;
run;

data &EM_EXPORT_TRAIN.;
set &CENTERFILE.;
run;
```

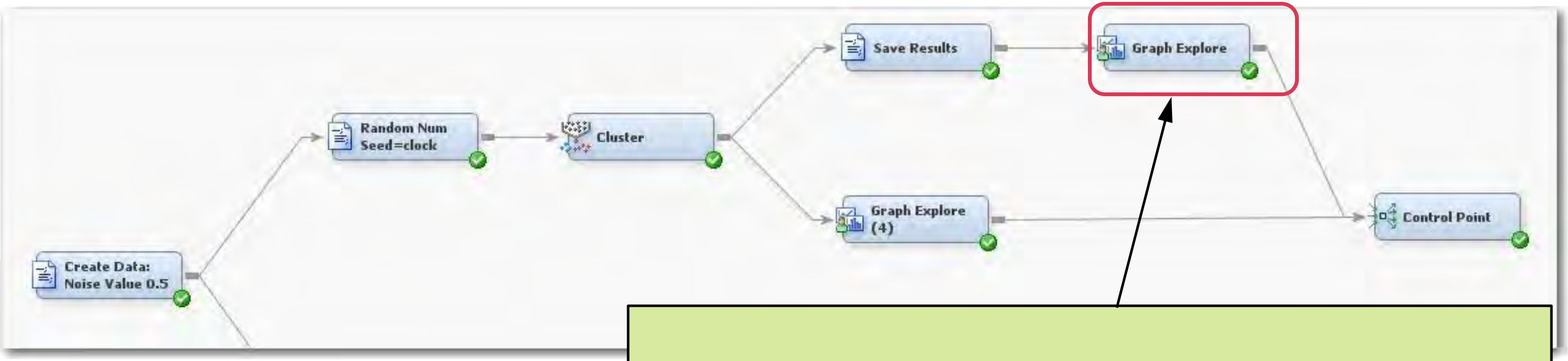
The Macro “Save_Cluster_Info” is called. The number of clusters is stored in a file called “yHOWMANYFILE” and the actual clusters are saved in a file called “yCENTERFILE”.

Cluster Node Data Collection
Enterprise Miner Diagram

```
%let INFILE          = &EM_IMPORT_CLUSMEAN.;

%let CENTERFILE      = SGFLIB.y5100_CENTERFILE;
%let HOWMANYFILE     = SGFLIB.y5100_HOWMANYFILE;

proc print data=&INFILE.;
run;

%save_cluster_info( &INFILE., &CENTERFILE., &HOWMANYFILE. );

proc print data=&CENTERFILE.(obs=30);
run;

proc print data=&HOWMANYFILE.(obs=10);
run;

proc freq data=&HOWMANYFILE.;
table _HOWMANY_ /missing;
run;

data &EM_EXPORT_TRAIN.;
set &CENTERFILE.;
run;
```

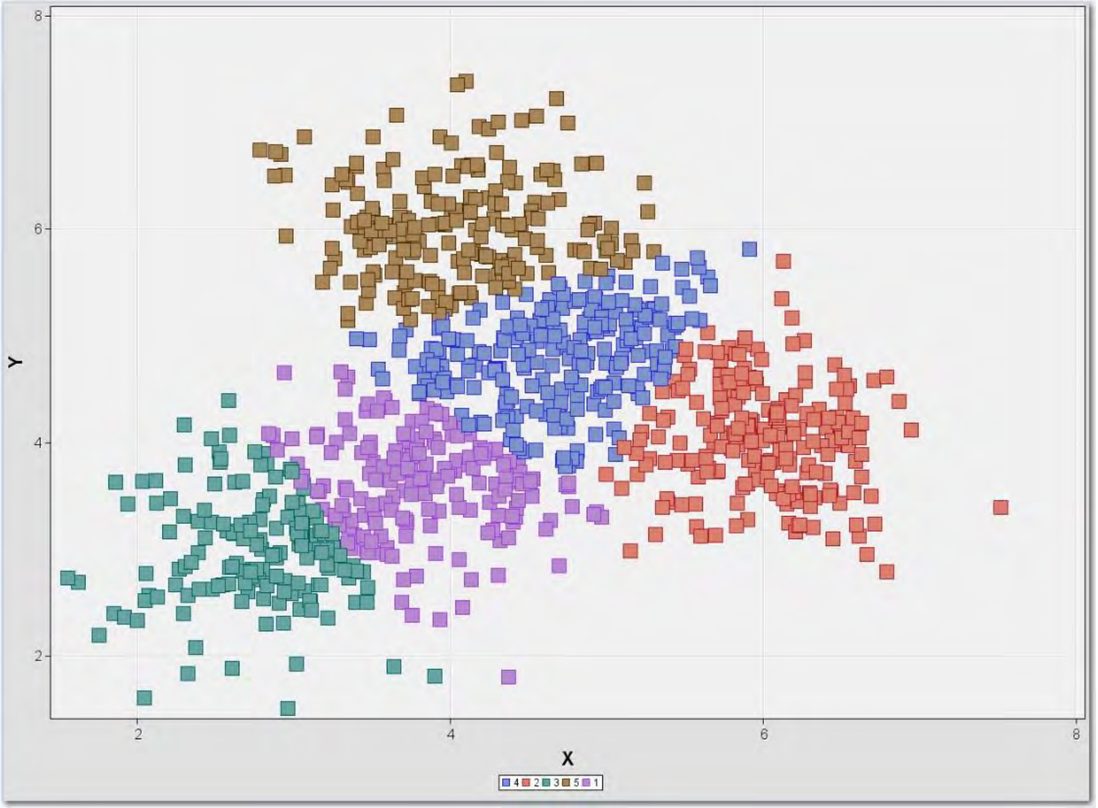
The Macro "Save_Cluster_Info" is called. The number of clusters is stored in a file called "yHOWMANYFILE" and the actual clusters are saved in a file called "yCENTERFILE".

Cluster Node Data Collection Enterprise Miner Diagram

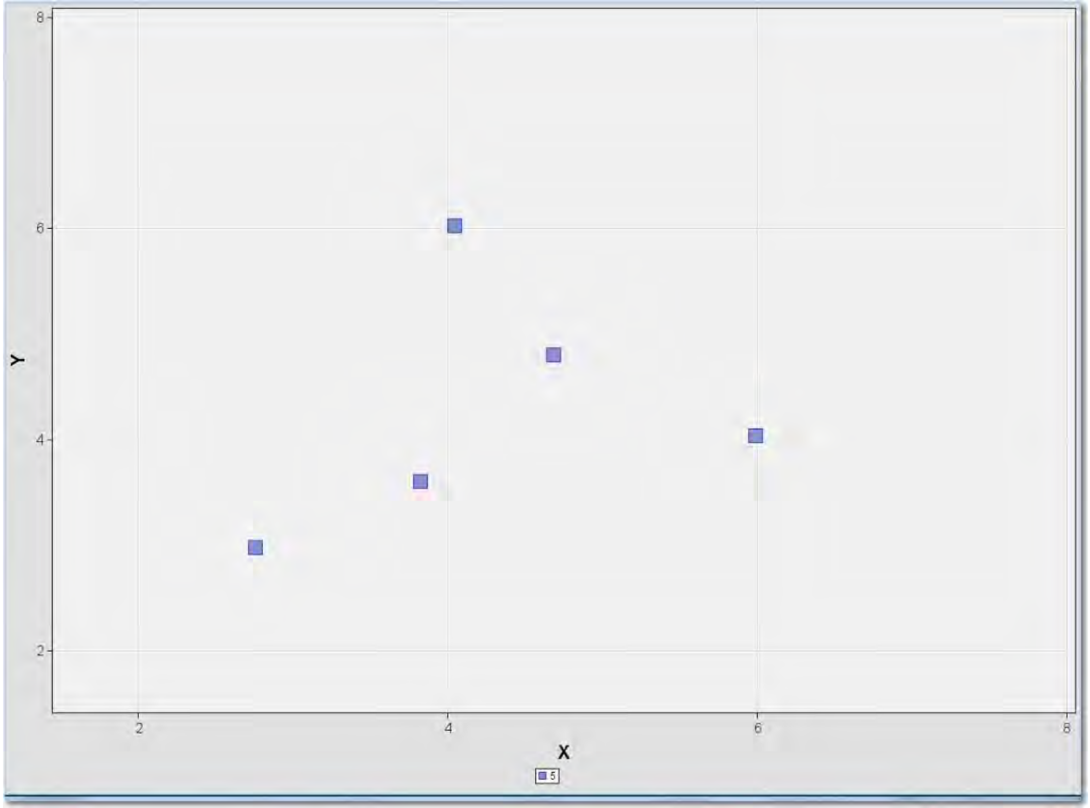


This graphing box is not necessary. It is being used for illustration purposes to display the cluster center points.

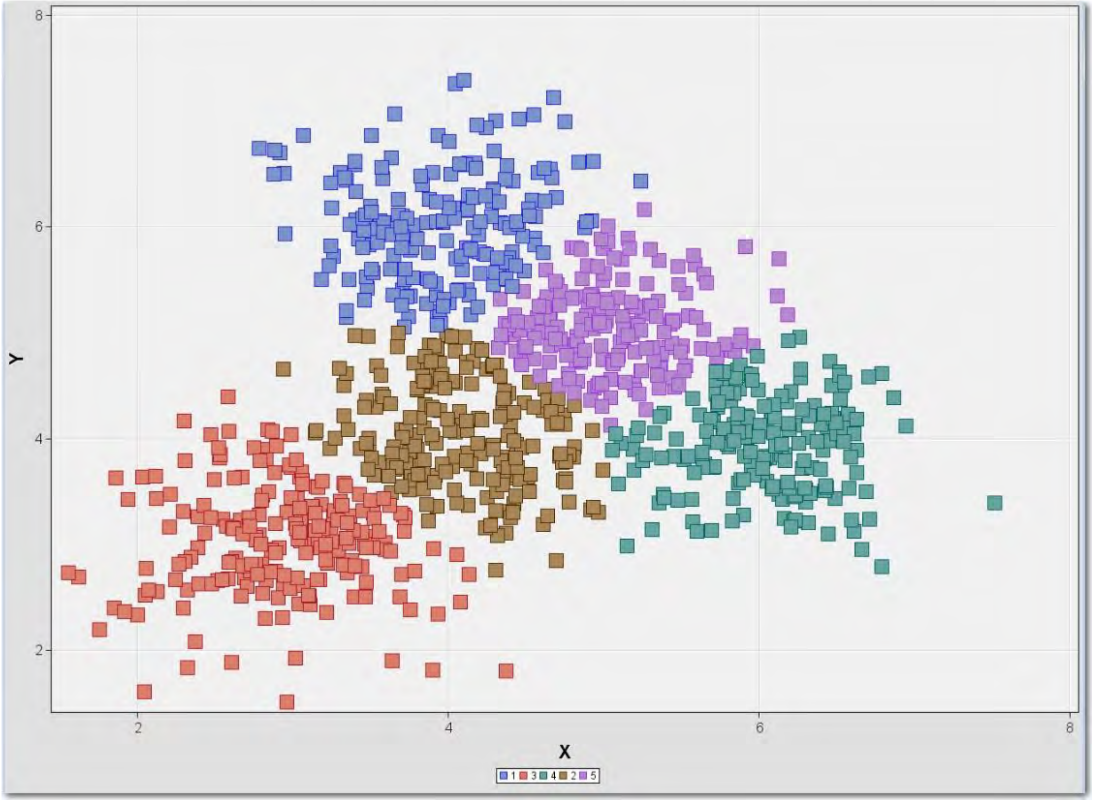
Cluster Node Data Collection Results: Run 1



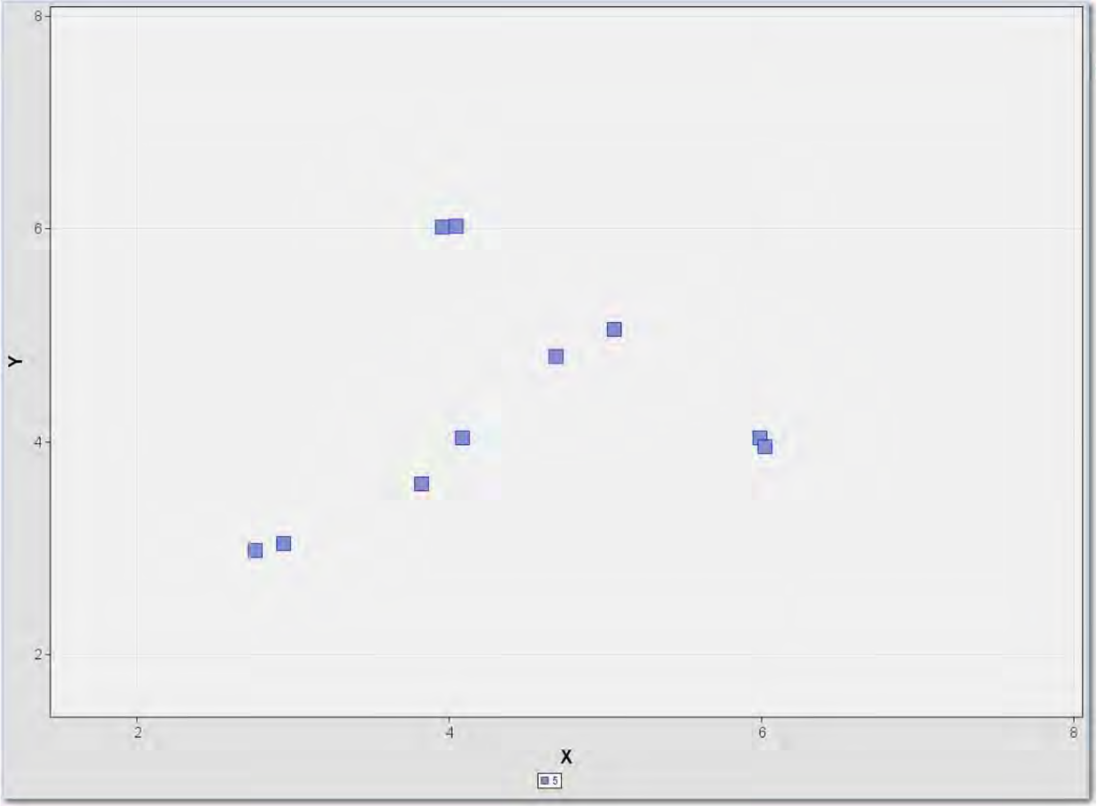
Cluster Node Data Collection
Center Points After 1 Run



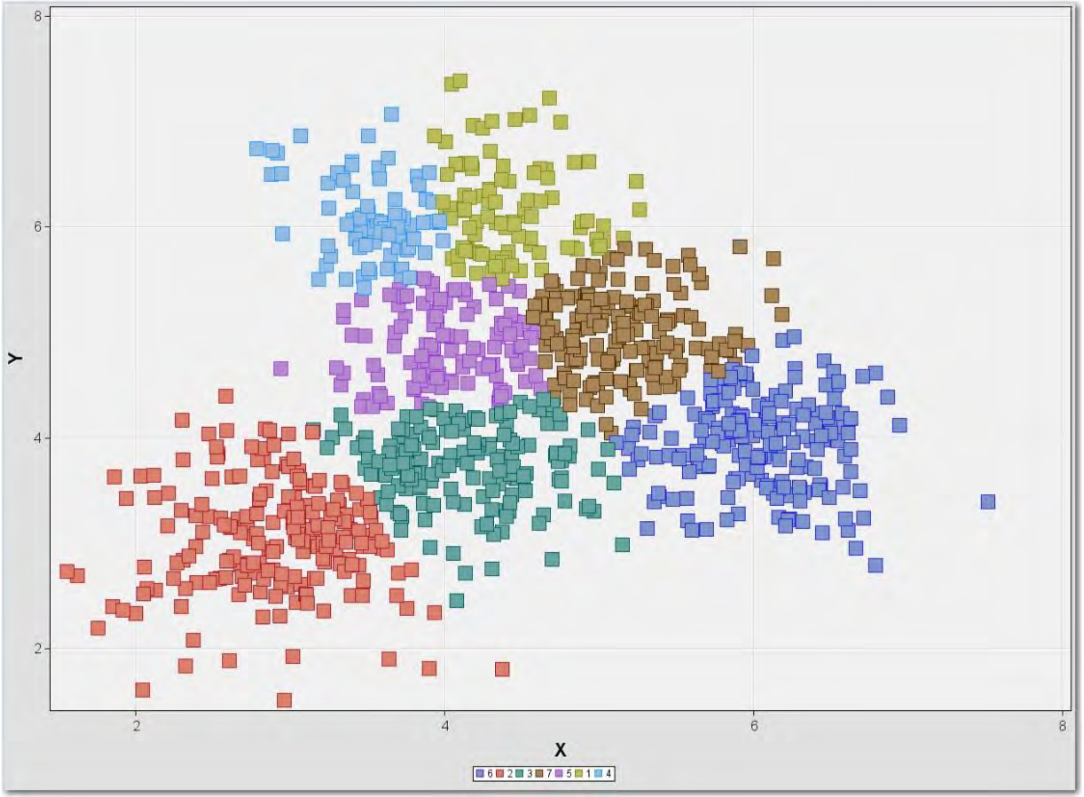
Cluster Node Data Collection Results: Run 2



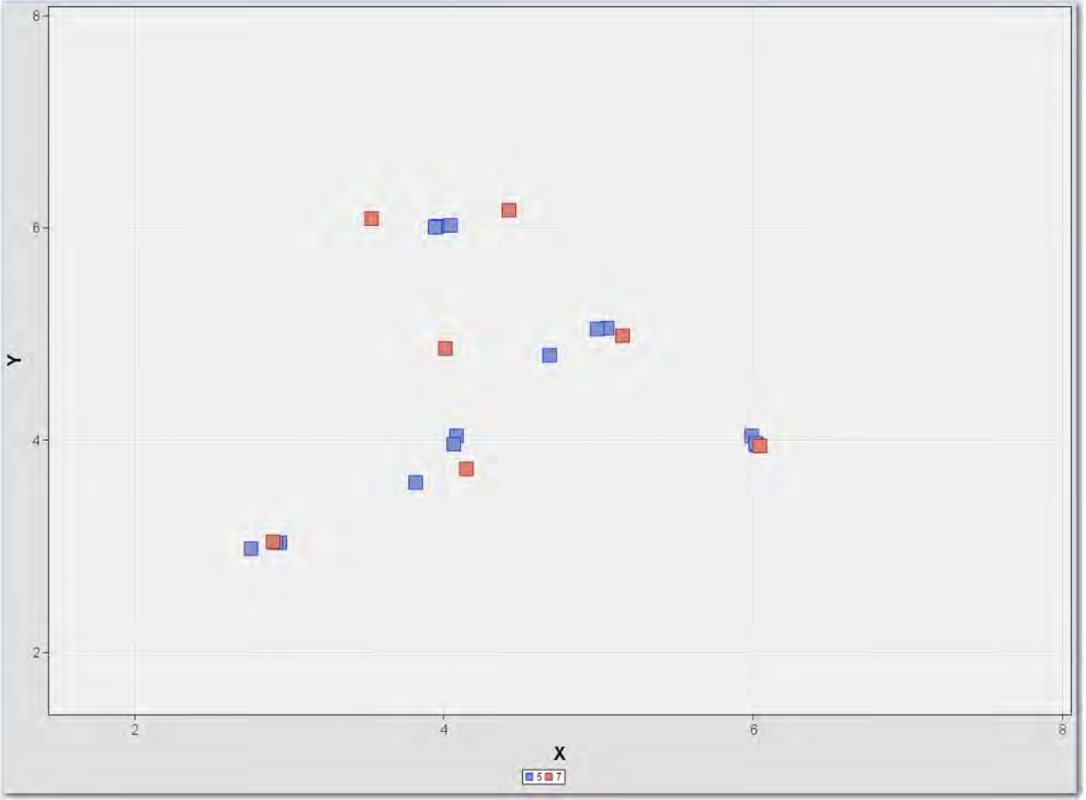
Cluster Node Data Collection
Center Points After 2 Runs



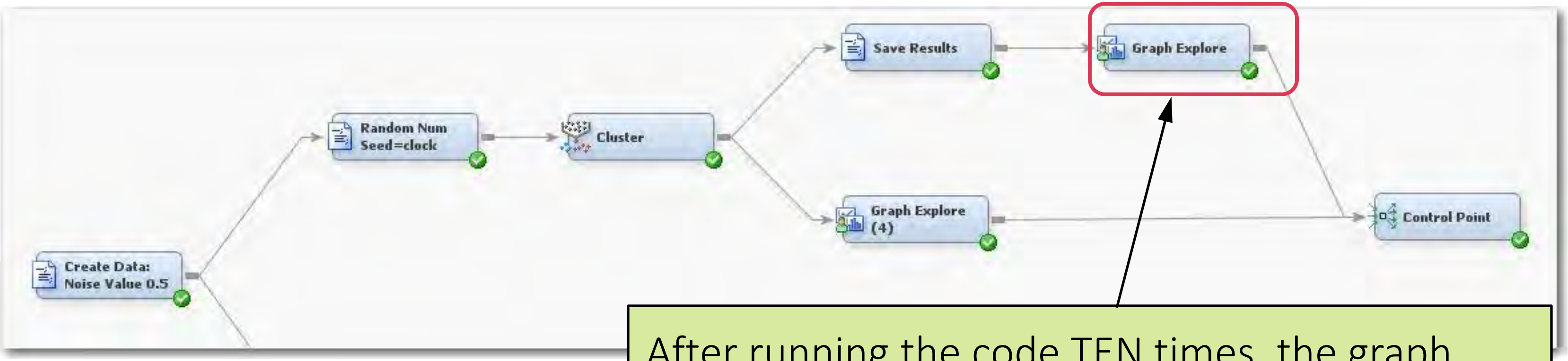
Cluster Node Data Collection Results: Run 4



Cluster Node Data Collection
Center Points After 4 Runs

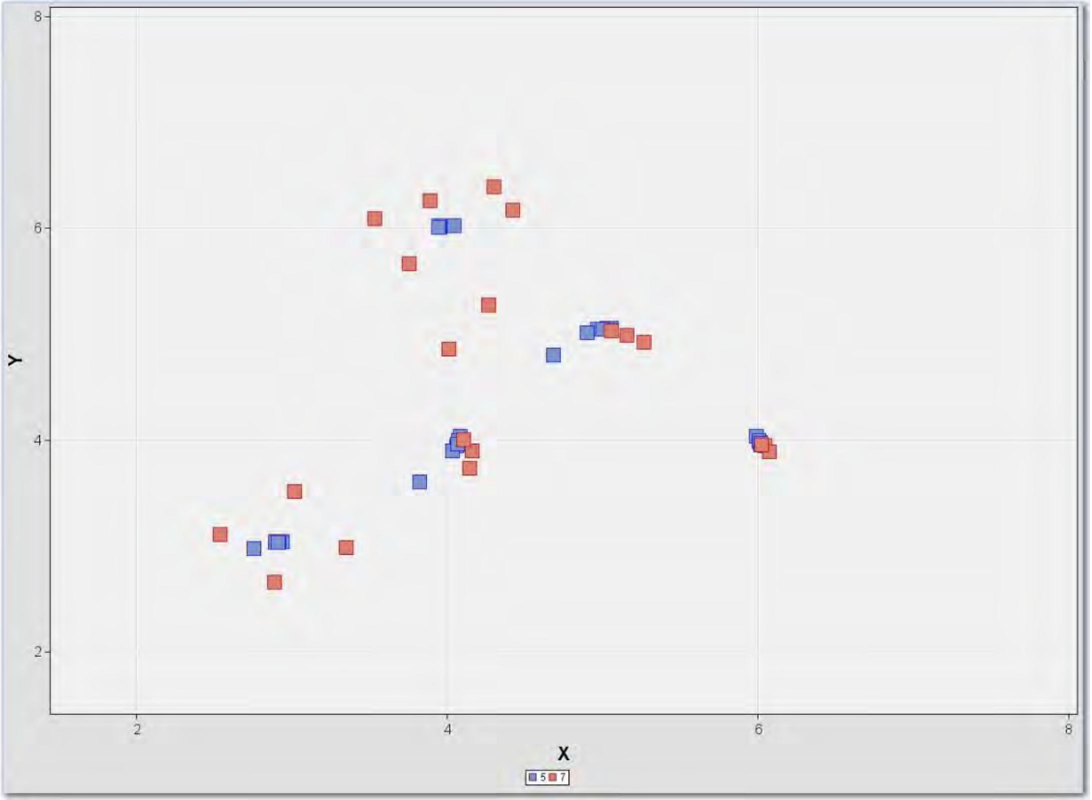


Cluster Node Data Collection
Enterprise Miner Diagram



After running the code TEN times, the graph suggests that the 5 cluster solution (red boxes) will place the centers in roughly the same places. The 7 cluster solution (blue boxes) will place the centers is roughly the same places (but these will be different from the red boxes).

Cluster Node Data Collection Center Points After 10 Runs



Looping in SAS Enterprise Miner

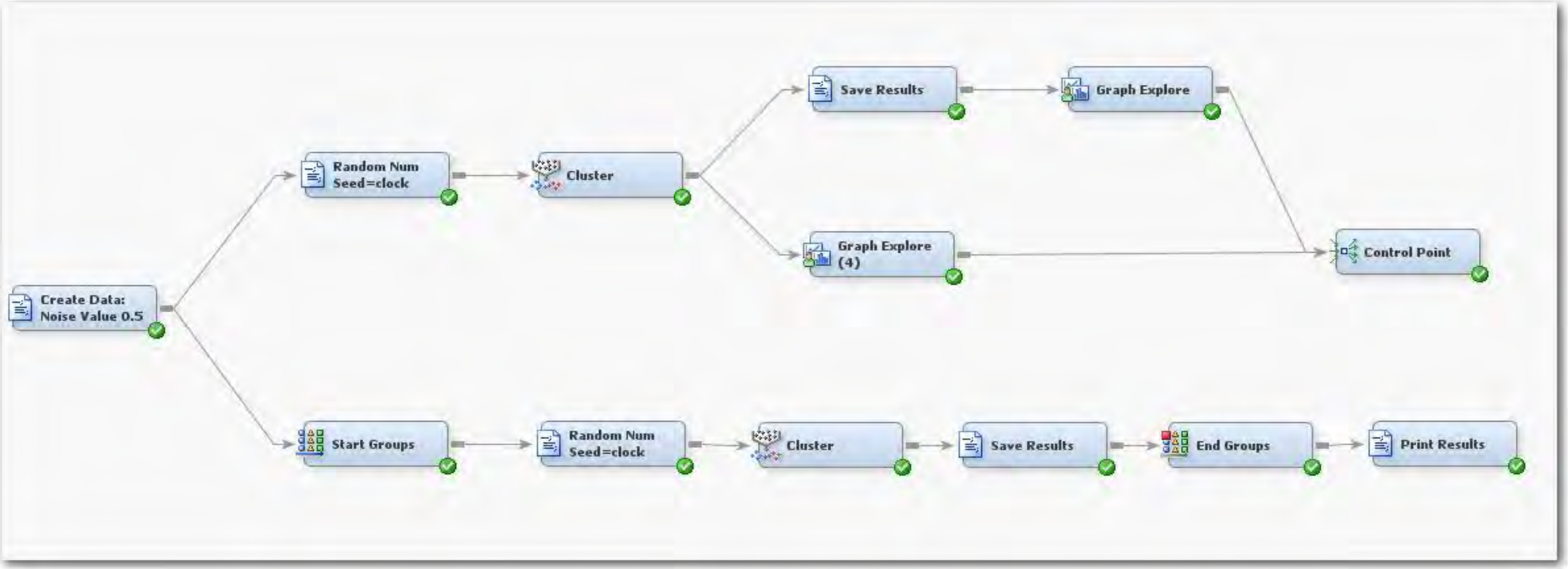
Automated Data Collection

Manually executing the Data Collection Program is time consuming

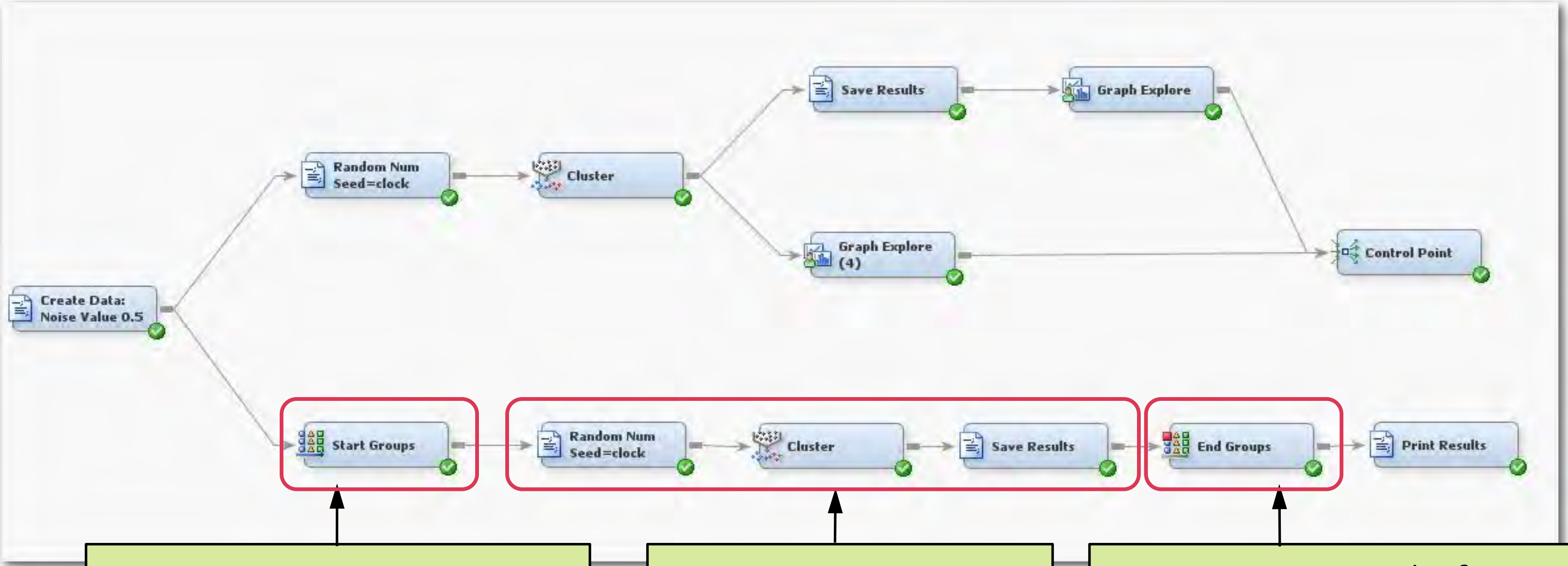
SAS Enterprise Miner has a "Looping" Structure to automate Cluster Data Collection

IMPORTANT: Occasionally when SAS Enterprise Miner is "Looping", then a error might occur. This is usually a result of a file deadlock state. It does not matter. Just exit Enterprise Miner and start running it again if you wish. You might have already collected enough samples by that point in time, so rerunning may not be necessary.

Automated Data Collection
Enterprise Miner Diagram



Automated Data Collection
Enterprise Miner Diagram

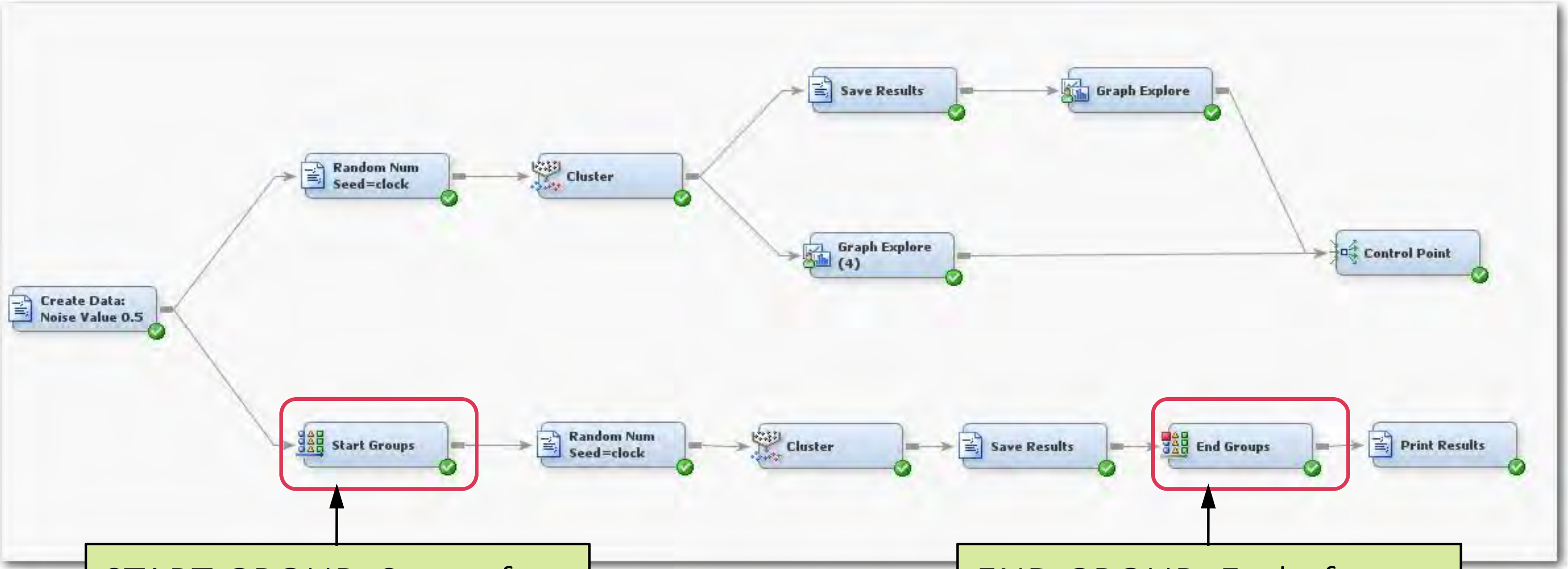


START GROUP: Start of an Enterprise Miner "Loop"

The Nodes Inside the"Start"/"End" Group will execute multiple times

END GROUP: End of an Enterprise Miner "Loop"

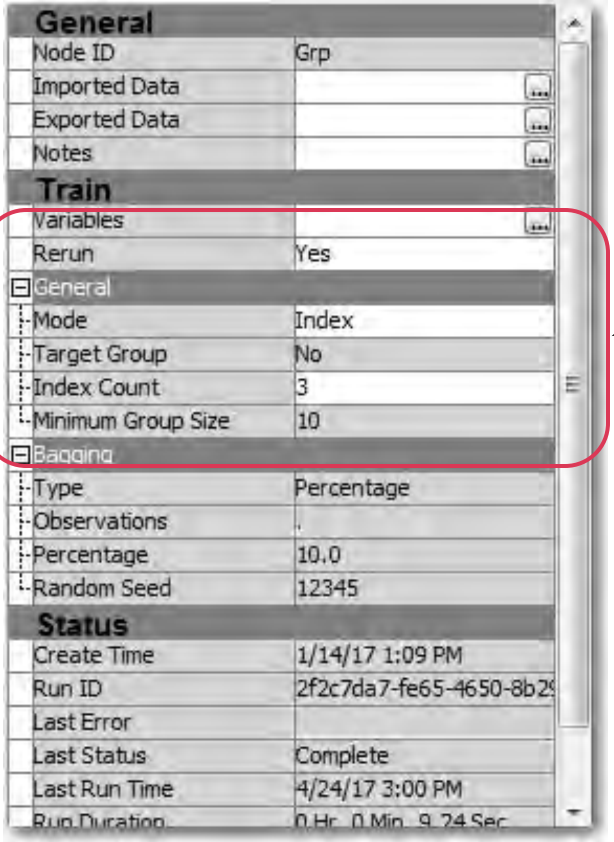
Automated Data Collection
Enterprise Miner Diagram



START GROUP: Start of an Enterprise Miner "Loop"

END GROUP: End of an Enterprise Miner "Loop"

Automated Data Collection
Enterprise Miner Diagram



- Rerun = Yes

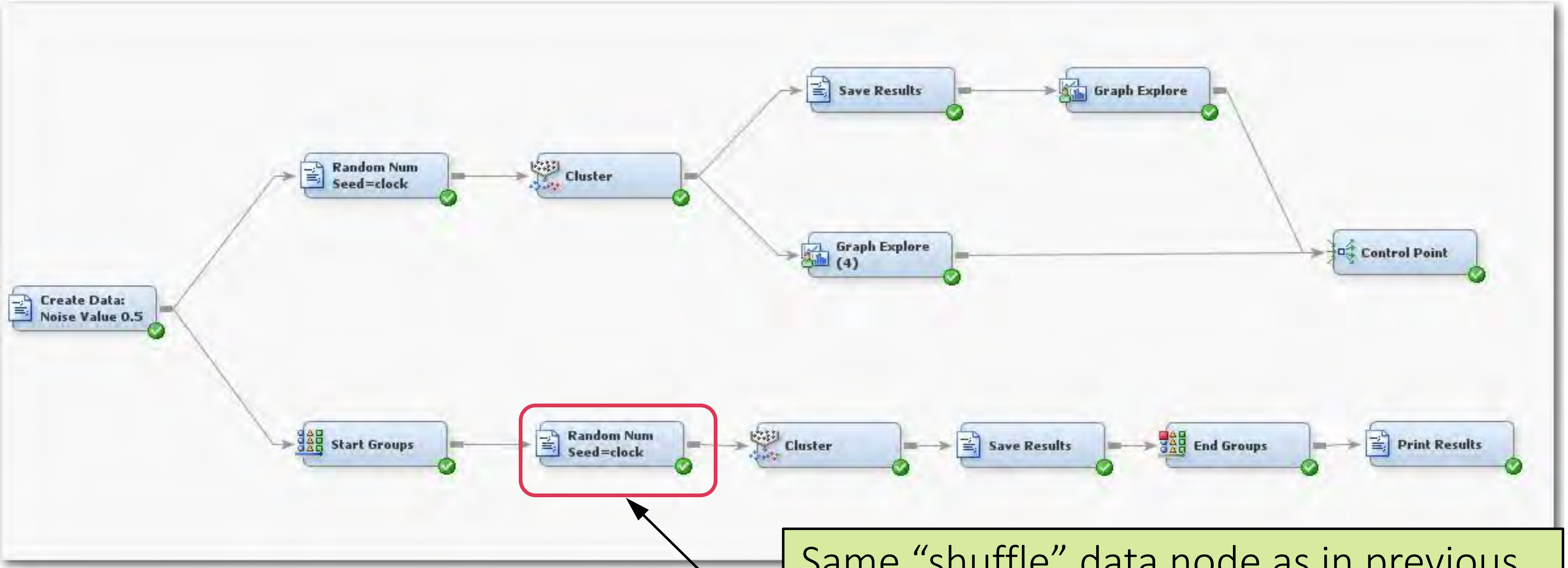
Mode:

- Index Informs SAS that it will loop "N" number of time.

Index Count:

- The Number of times the loop will execute. In this case the number will be "3" but the number can be set to a much higher value if a person plans to be away from their computer for a while.

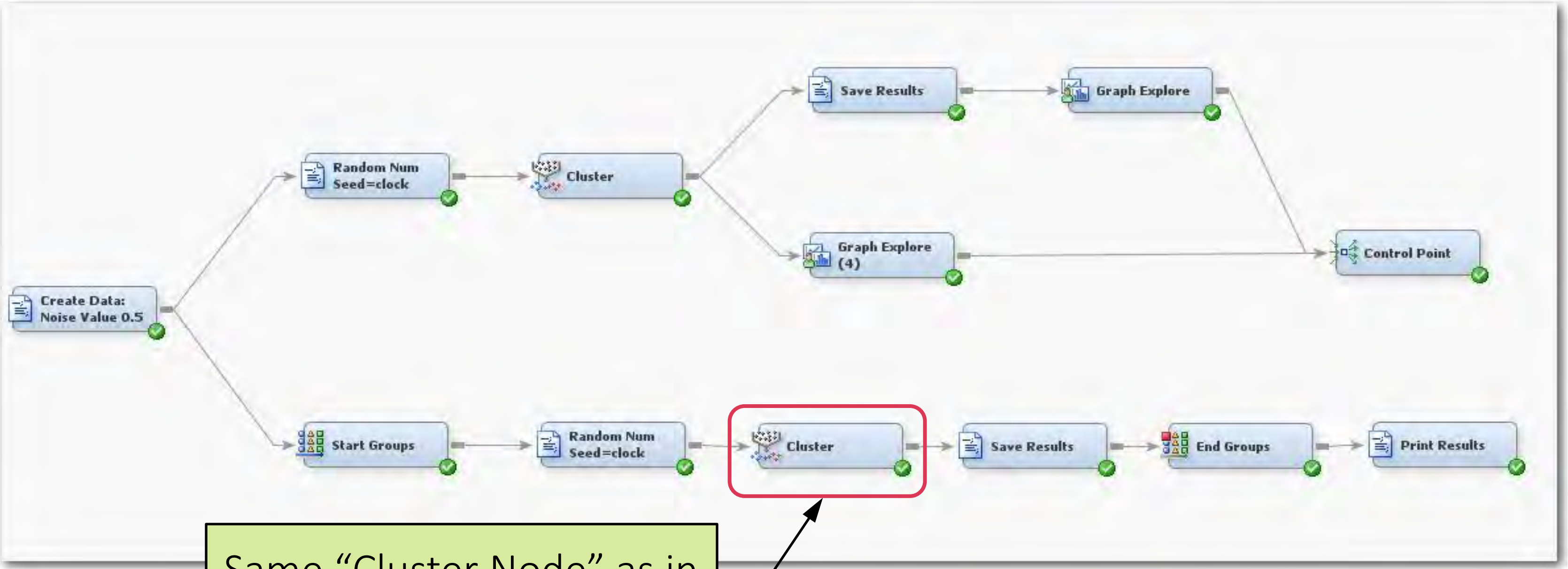
Automated Data Collection
Enterprise Miner Diagram



Same “shuffle” data node as in previous example. The seed is -1 which means it is tied to the clock.

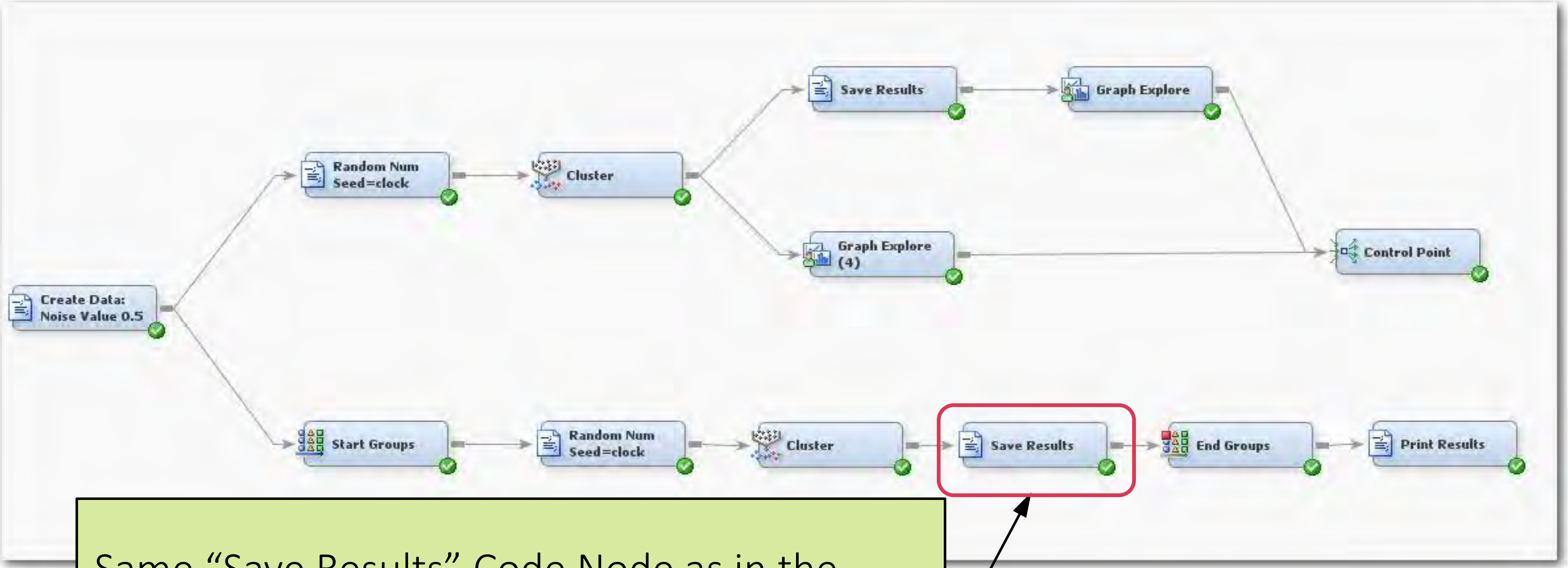
Rerun is set to “YES” just as before.

Automated Data Collection
Enterprise Miner Diagram



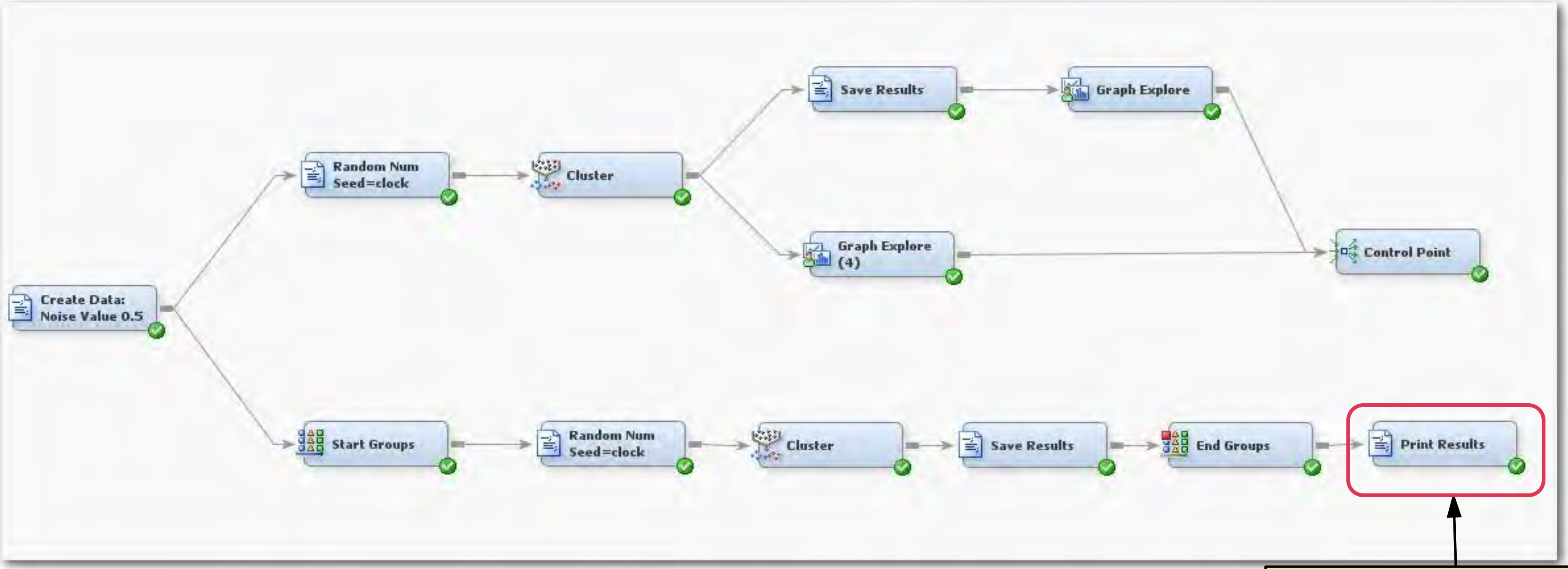
Same “Cluster Node” as in previous example.
All settings should be the same as before.

Automated Data Collection
Enterprise Miner Diagram



Same "Save Results" Code Node as in the previous example. Note: "PROC PRINT" and other output PROCS won't display inside of a loop.

Automated Data Collection
Enterprise Miner Diagram



Print the Results

Automated Data Collection
Enterprise Miner Diagram

```
%let CENTERFILE      = SGFLIB.y5100_CENTERFILE;
%let HOWMANYFILE     = SGFLIB.y5100_HOWMANYFILE;


proc print data=&CENTERFILE.(obs=100);
run;

proc print data=&HOWMANYFILE.(obs=100);
run;


proc freq data=&HOWMANYFILE.;
table _HOWMANY_ /missing;
run;
```

Automated Data Collection
Enterprise Miner Diagram

The FREQ Procedure

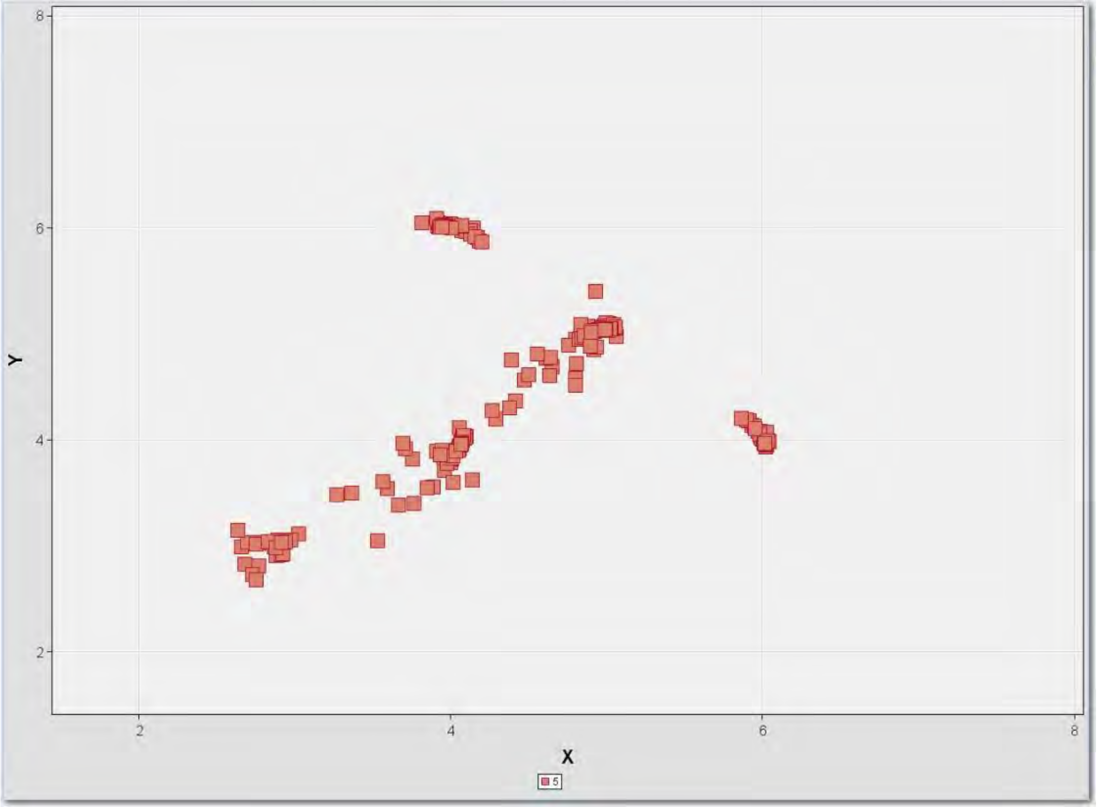
HOWMANY	Frequency	Percent	Cumulative Frequency	Cumulative Percent
3	6	2.58	6	2.58
4	6	2.58	12	5.15
5	161	69.10	173	74.25
7	60	25.75	233	100.00

After running 233 time, it is observed that

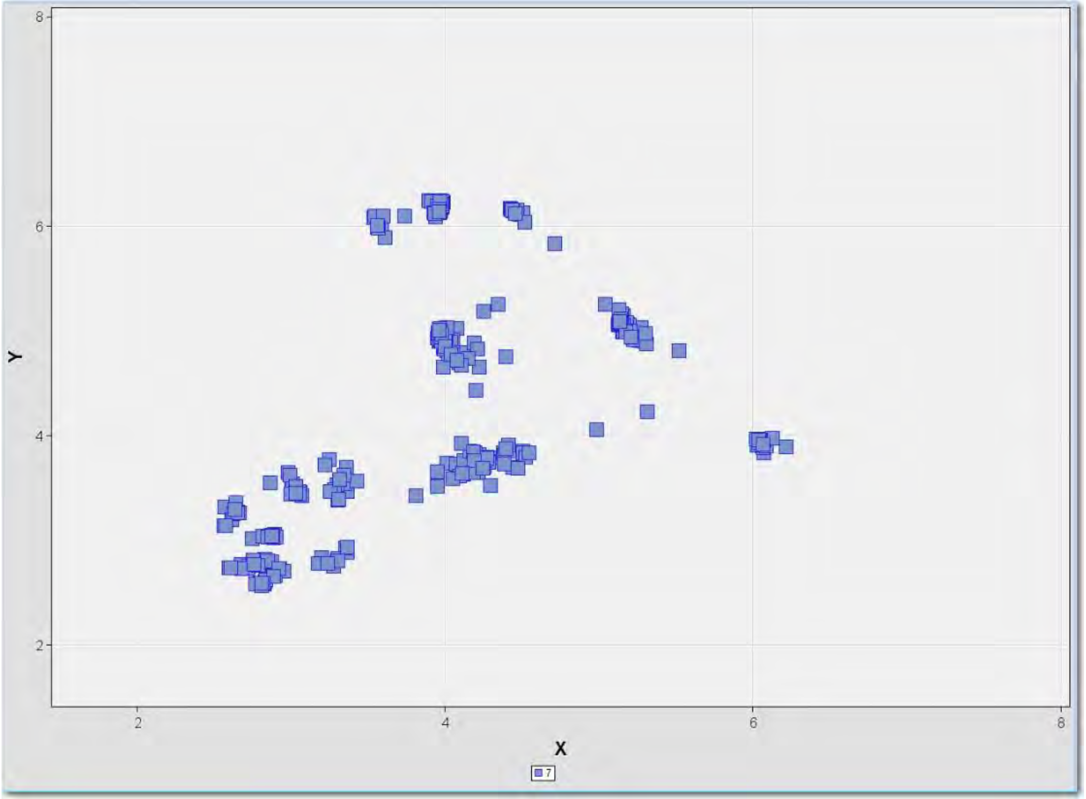
- 70% of the time, 5 clusters are found
- 26% of the time, 7 clusters are found

Note: Because of the nature of the random number generator, rerunning this model might yield slightly different results.

Automated Data Collection
Clusters = 5 Center Points

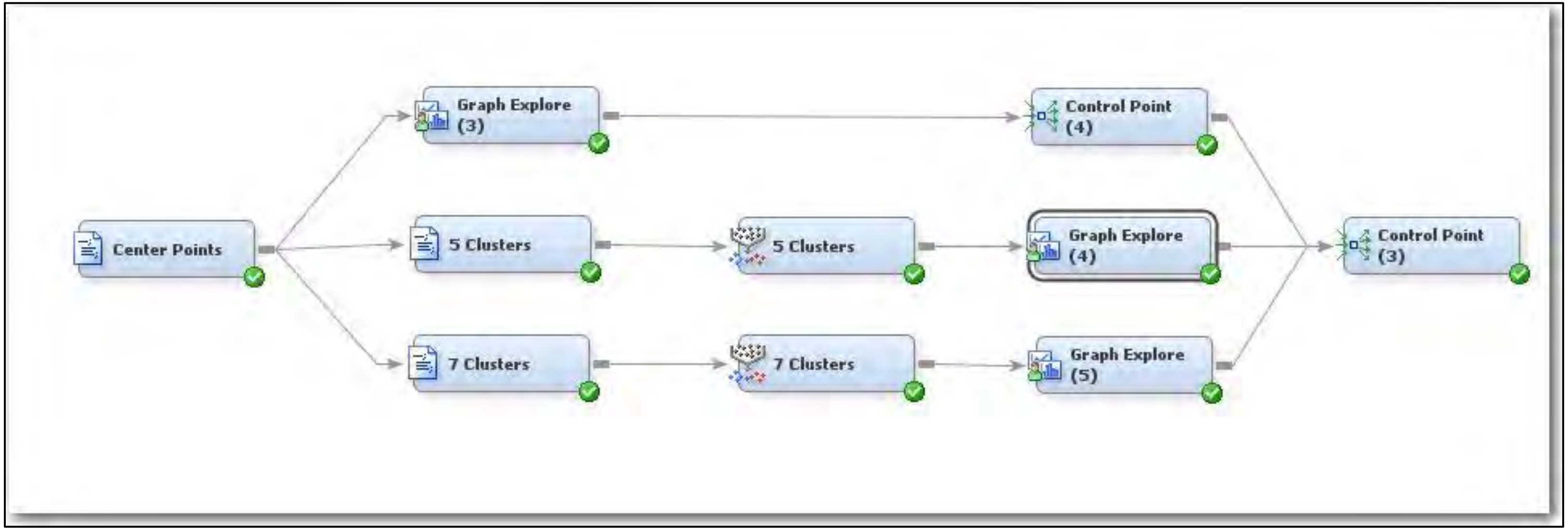


Automated Data Collection
Clusters = 7 Center Points

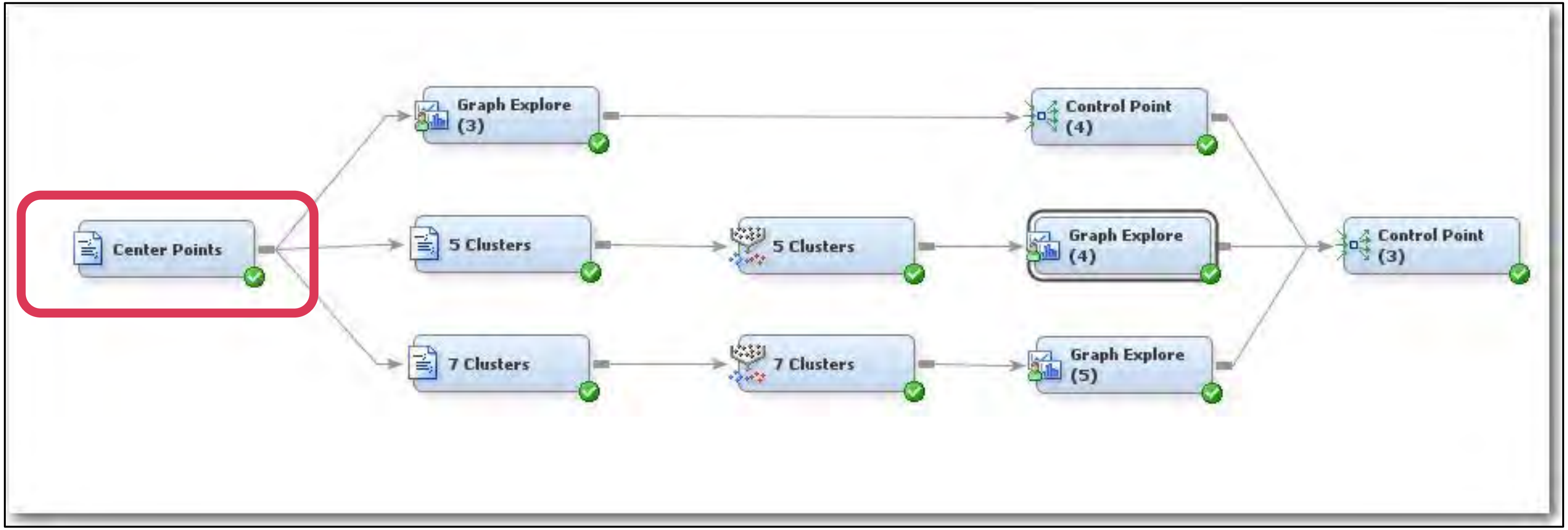


Cluster the Centers

Cluster the Cluster Centers Enterprise Miner Diagram



Cluster the Cluster Centers
Enterprise Miner Diagram



Automated Data Collection
Enterprise Miner Diagram

```
*%let CENTERFILE    = SGFLIB.y5100_CENTERFILE;
*%let HOWMANYFILE   = SGFLIB.y5100_HOWMANYFILE;

%let CENTERFILE     = SGFLIB.z5100_CENTERFILE;
%let HOWMANYFILE    = SGFLIB.z5100_HOWMANYFILE;

%let OUTFILE        = &EM_EXPORT_TRAIN.;


proc print data=&CENTERFILE.(obs=100);
run;

proc print data=&HOWMANYFILE.(obs=100);
run;


proc freq data=&HOWMANYFILE.;
table _HOWMANY_ /missing;
run;


data &OUTFILE.;
set &CENTERFILE.;
run;
```

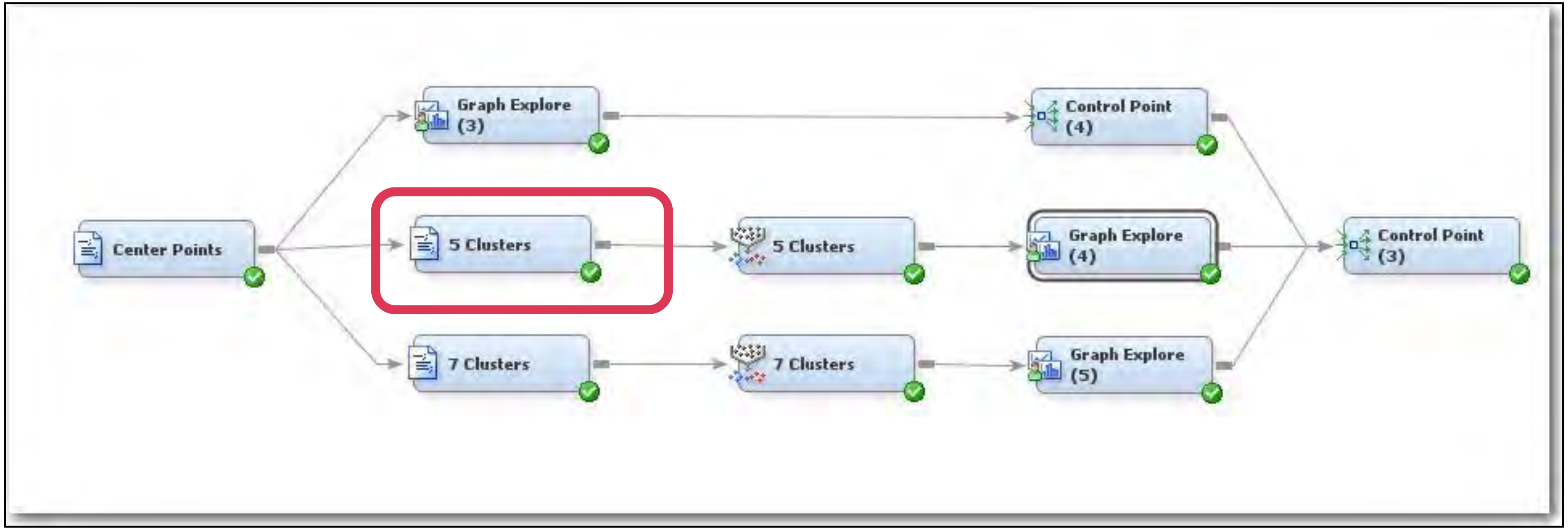
Automated Data Collection
Enterprise Miner Diagram

```
*%let CENTERFILE     = SGFLIB.y5100_CENTERFILE;
*%let HOWMANYFILE    = SGFLIB.y5100_HOWMANYFILE;

%let CENTERFILE      = SGFLIB.z5100_CENTERFILE;
%let HOWMANYFILE     = SGFLIB.z5100_HOWMANYFILE;

%let OUTFILE         = &EM_EXPORT_TRAIN.;


proc print data=&CENTERFILE.(c
run;

proc print data=&HOWMANYFILE.(
run;


proc freq data=&HOWMANYFILE.;
table _HOWMANY_ /missing;
run;


data &OUTFILE.;
set &CENTERFILE.;
run;
```

For convenience, the program was already run 200+ times and the results were stored in the files:

SGFLIB.z5100_CENTERFILE;
SGFLIB.z5100_HOWMANYFILE;

Cluster the Cluster Centers
Enterprise Miner Diagram



Automated Data Collection
Enterprise Miner Diagram

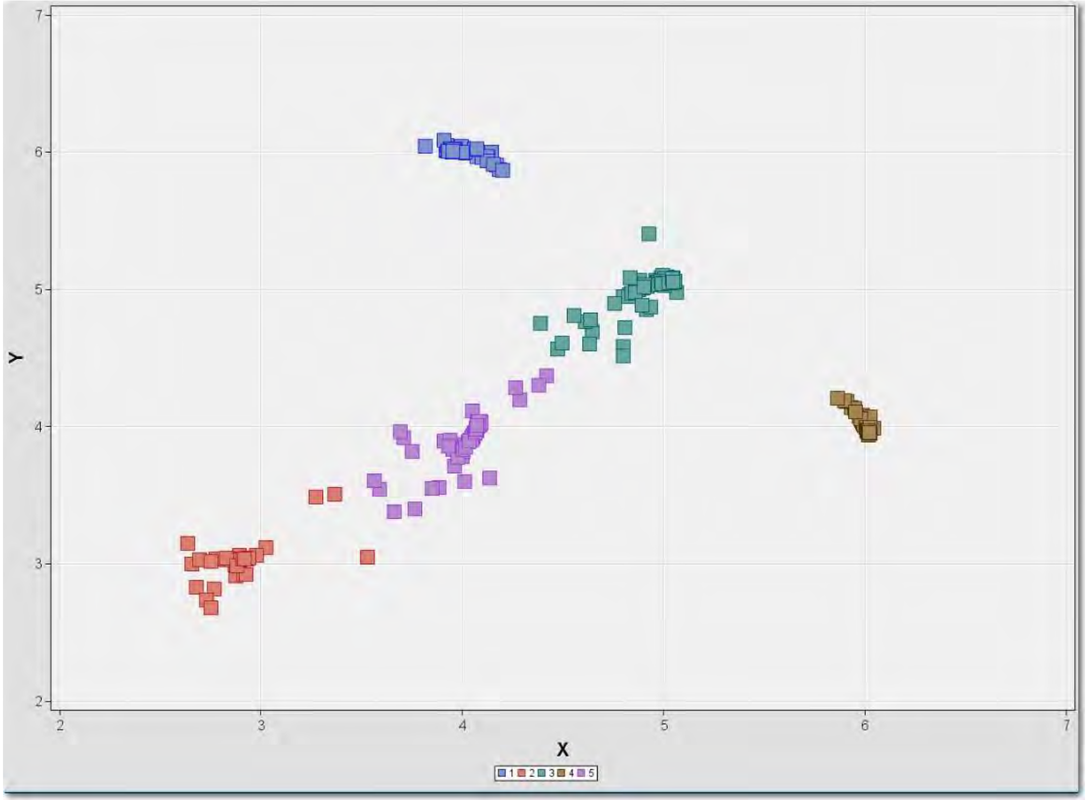
```
%let INFILE            = &EM_IMPORT_DATA.;
%let OUTFILE           = &EM_EXPORT_TRAIN.;
%let HOWMANY           = 5;

proc print data=&INFILE.(obs=100);
run;

data &OUTFILE.;
set &INFILE.;
if _HOWMANY_ = &HOWMANY.;
drop _HOWMANY_;
run;
```

Automated Data Collection
Enterprise Miner Diagram

```
%let INFILE           = &EM_IMPORT_DATA.;
%let OUTFILE          = &EM_EXPORT_TRAIN.;
%let HOWMANY          = 5;

proc print data=&INFILE.(obs=100);
run;

data &OUTFILE.;
set &INFILE.;
if _HOWMANY_  = &HOWMANY.;
drop _HOWMANY_;
run;
```

Only keep the CENTER POINTS for the times when **5 clusters** were found.

Example 3: Cluster the Cluster Centers
Cluster of Center Points = 5

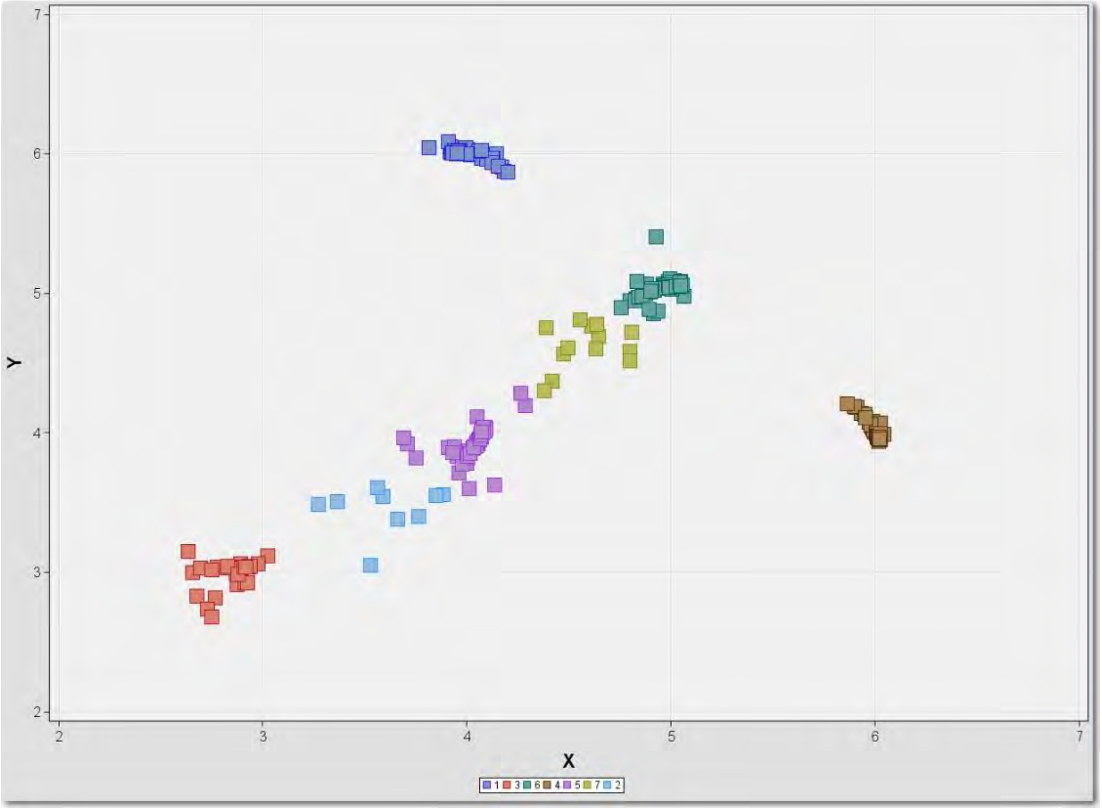


Automated Data Collection
Enterprise Miner Diagram

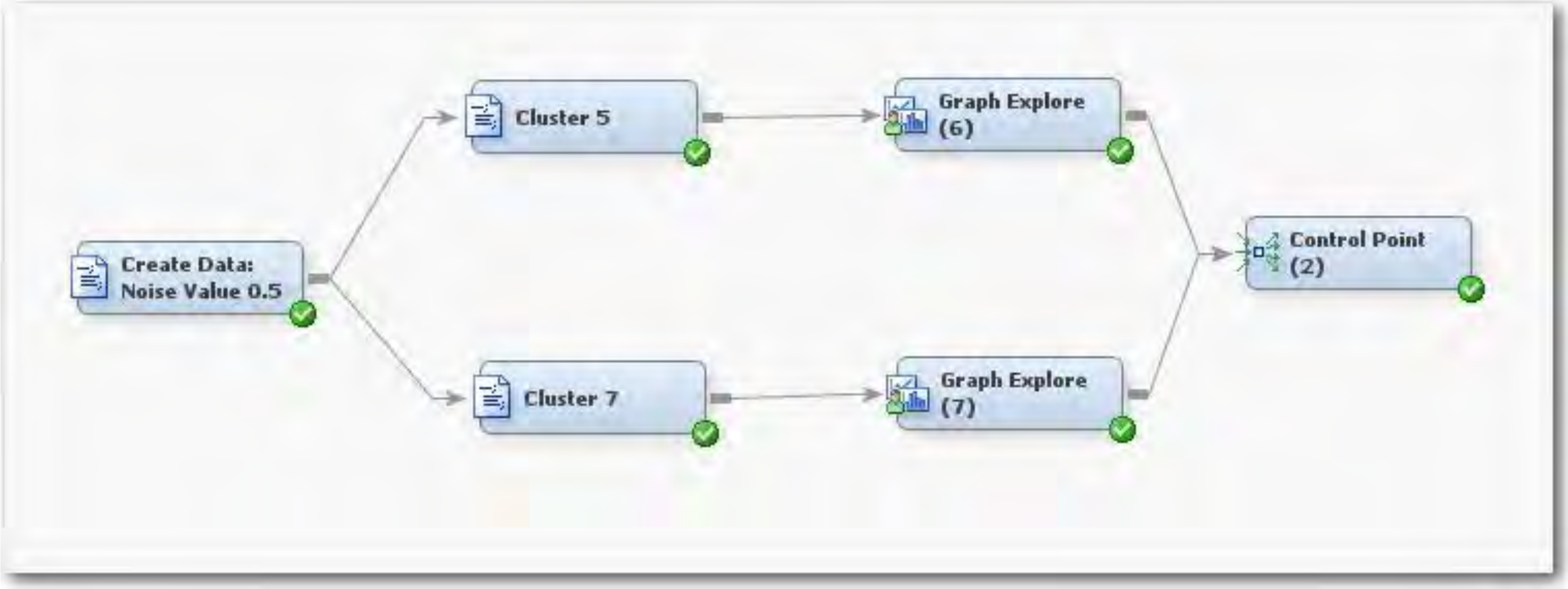
```
%let INFILE             = &EM_IMPORT_DATA.;
%let OUTFILE            = &EM_EXPORT_TRAIN.;
%let HOWMANY            = 7;

proc print data=&INFILE.(obs=100);
run;

data &OUTFILE.;
set &INFILE.;
if _HOWMANY_ = &HOWMANY.;
drop _HOWMANY_;
run;
```

Only keep the CENTER POINTS for the times when **7 clusters** were found.

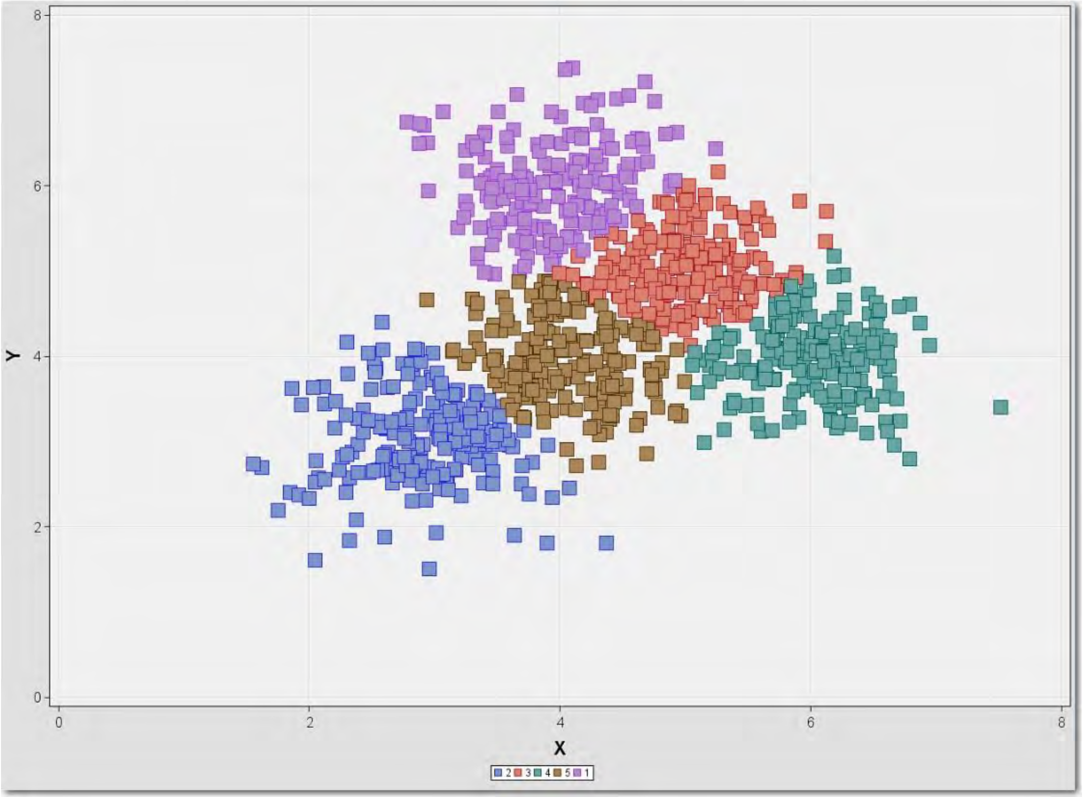
Example 3: Cluster the Cluster Centers
Cluster of Center Points = 7



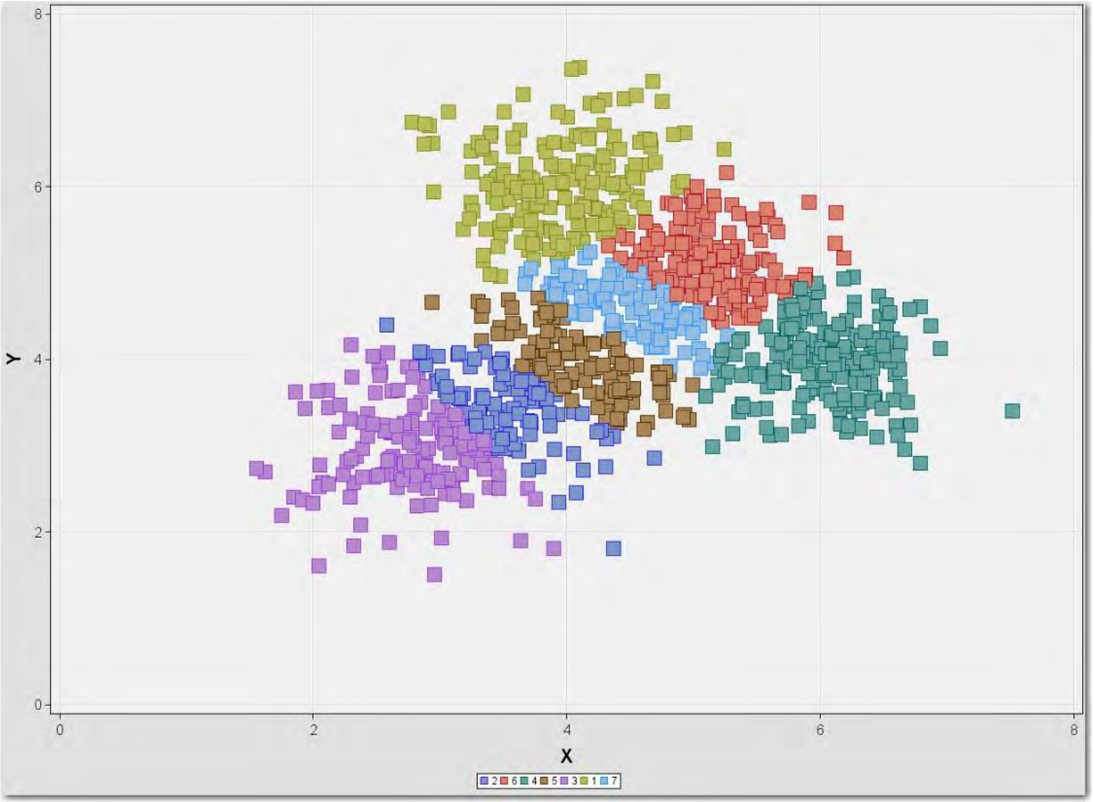
Cluster the Cluster Centers Applied to Original Data



Example 3: Clusters = 5
Applied to Original Data

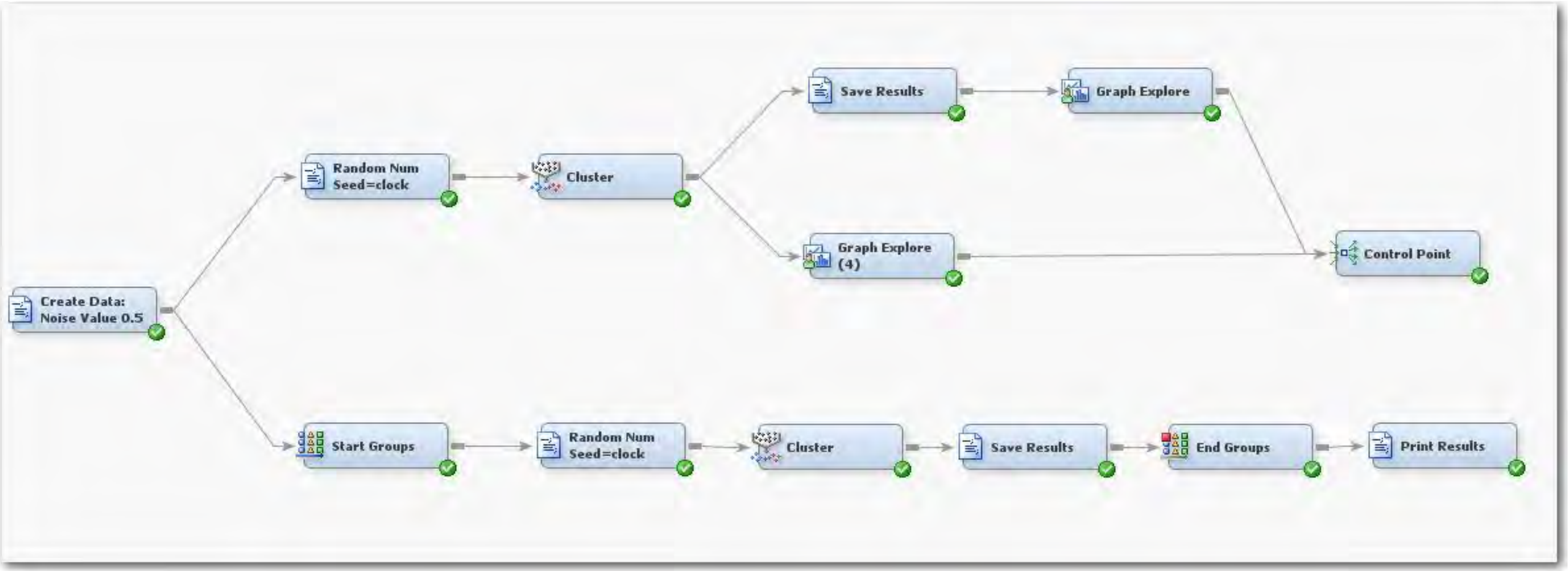


Example 3: Clusters = 7
Applied to Original Data

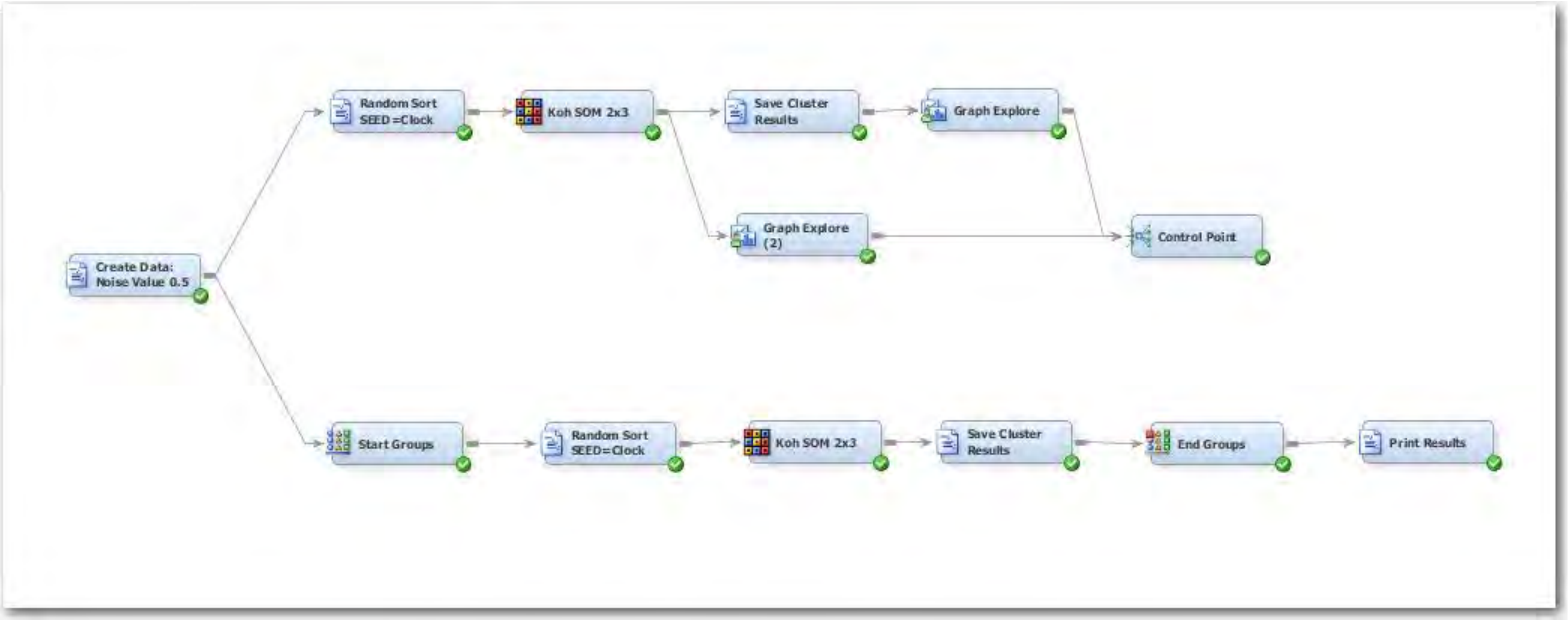


Kohonen/SOM Clusters

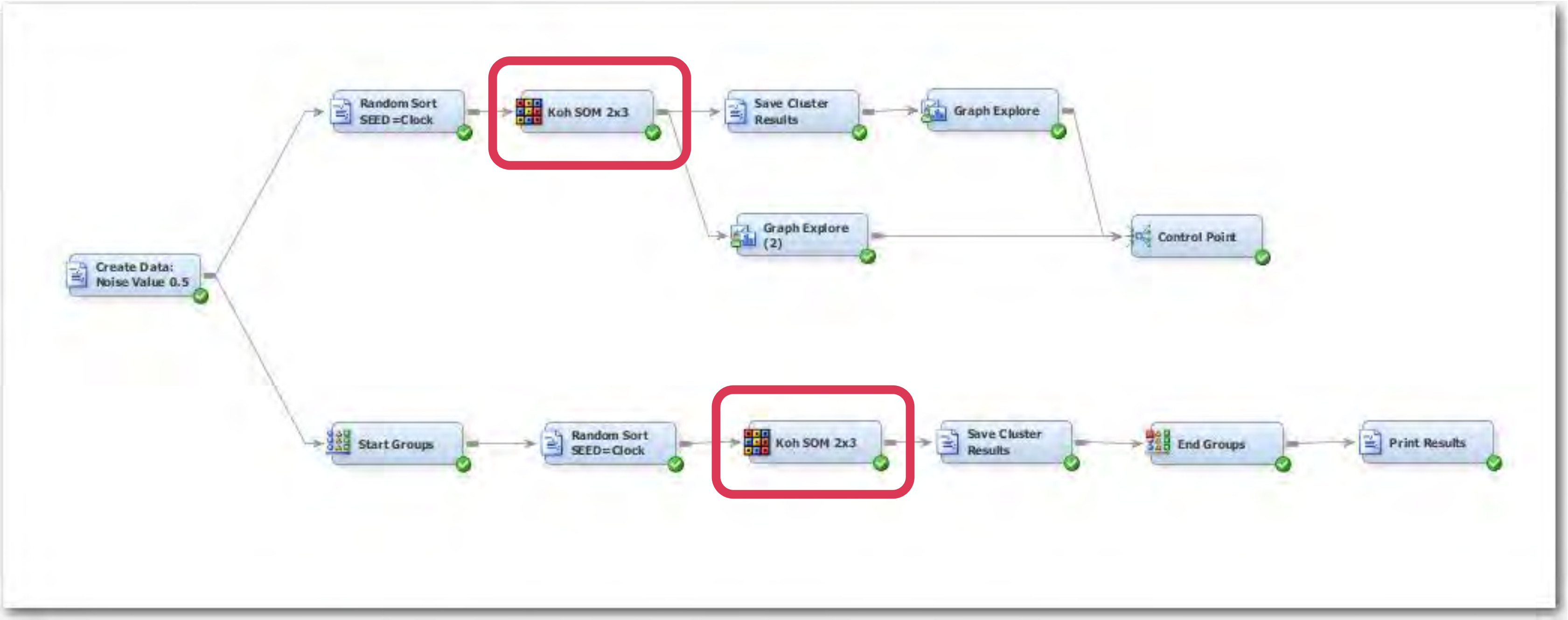
Automated Data Collection
Enterprise Miner Diagram



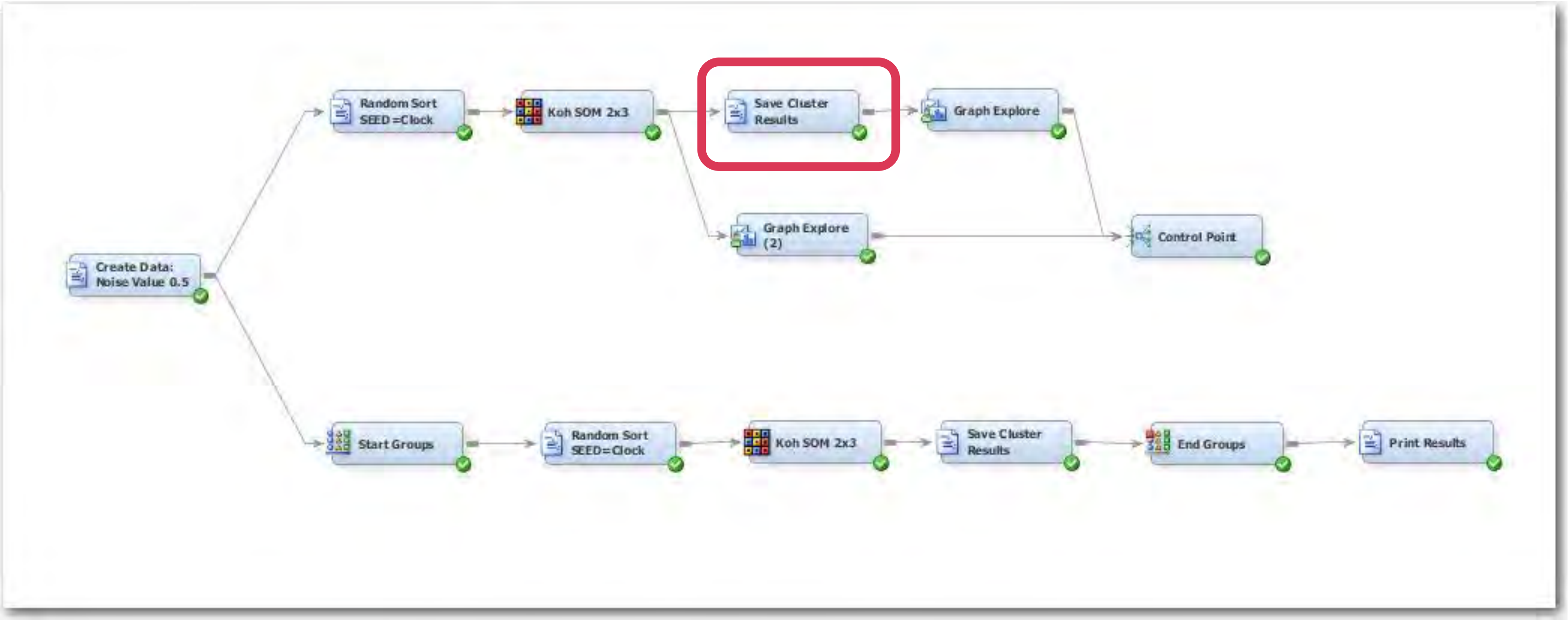
Automated Data Collection Enterprise Miner Diagram



Automated Data Collection
Enterprise Miner Diagram



Automated Data Collection
Enterprise Miner Diagram



Cluster Node Data Collection
Enterprise Miner Diagram

```
%let INFILE          = &EM_LIB..&EM_METASOURCE_NODEID._OUTMEAN;

%let CENTERFILE      = SGFLIB.y6100_CENTERFILE;
%let HOWMANYFILE     = SGFLIB.y6100_HOWMANYFILE;

proc print data=&INFILE.;
run;

%save_cluster_info( &INFILE., &CENTERFILE., &HOWMANYFILE. );

proc print data=&CENTERFILE.(obs=30);
run;

proc print data=&HOWMANYFILE.(obs=10);
run;

proc freq data=&HOWMANYFILE.;
table _HOWMANY_ /missing;
run;

data &EM_EXPORT_TRAIN.;
set &CENTERFILE.;
run;
```

Cluster Node Data Collection
Enterprise Miner Diagram

```
%let INFILE         = &EM_LIB..&EM_METASOURCE_NODEID._OUTMEAN;

%let CENTERFILE     = SGFLIB.y6100_CENTERFILE;
%let HOWMANYFILE    = SGFLIB.y6100_HOWMANYFILE

proc print data=&INFILE.;
run;

%save_cluster_info( &INFILE., &CENTERFILE., &

proc print data=&CENTERFILE.(obs=30);
run;

proc print data=&HOWMANYFILE.(obs=10);
run;

proc freq data=&HOWMANYFILE.;
table _HOWMANY_ /missing;
run;

data &EM_EXPORT_TRAIN.;
set &CENTERFILE.;
run;
```

SAS Enterprise Miner creates a file to hold the Kohonen/SOM center points. But they are not exported.

Therefore, you need to go out and get them!

Questions?