Data Analytics for Data Warehousing

Assignment Project Exam Help
WeChat: cstutorcs

https://tutorcs.com

Outline

- Data analytics for data warehousing focuses on numerical value on fact in the star schema
- This chapter focuses on three main data analysis techniques suitable in a data warehousing context, which are:
  1. Regression
  2. Clustering
  3. Classification
- Review on Traditional Data Mining Techniques

1 Traditional Data Mining Techniques vs. Data Analytics for Data Warehousing

- Traditional Data Mining focus on categorical data
- Data Warehousing focus on Fact table with numerical values
- Adaptations are required

1.1 Traditional Data Mining Techniques

- A collection of techniq[illegible] discovery patterns, correlations and knowledge on the [illegible]nput data.

- Some techniques:
  - Association rules
  - Sequential patterns
  - Classification
  - Clustering
- Requires specific data structure

1.1 Traditional Data Mining Techniques: Task Perspective

- Descriptive Mining
  - Describes the dataset … sents interesting general properties of the data.
  - Summarizes the data in terms of its properties and correlation with others.
- Predictive Mining
  - Builds a prediction model from the available set of data and attempts to predict the behaviour of new datasets.

1.1 Traditional Data Mining Techniques: Association Rules

- Discover association relationships among a set of items.
- Commonly used in transaction data analysis.
- *Example:*
  - cereal → milk
  - bread→ cereal, juice

Transaction ID	Items Purchased
1	Bread, Jam, Milk
2	Bread, Cereal, Juice, Milk
3	Bread, Cereal, Jam, Juice
4	Bread, Cereal, Juice
5	Coffee, Milk, Oat

Association Rules

Association Rules	Confidence
Bread→Cereal	75%
Bread→Juice	75%
Cereal→Bread	100%
Cereal→Juice	100%
Jam→Bread	100%
Juice→Bread	100%
Juice→Cereal	100%
Bread→(Cereal,Juice)	75%
Cereal→(Bread,Juice)	100%
Juice→(Bread,Cereal)	100%
(Bread,Cereal)→Juice	100%
(Bread,Juice)→Cereal	100%
(Cereal,Juice)→Bread	100%

1.1 Traditional Data Mining Techniques: Association Rules

Three main issues:

1. Unnormalized Data
2. No numerical fact measure
3. One dimension

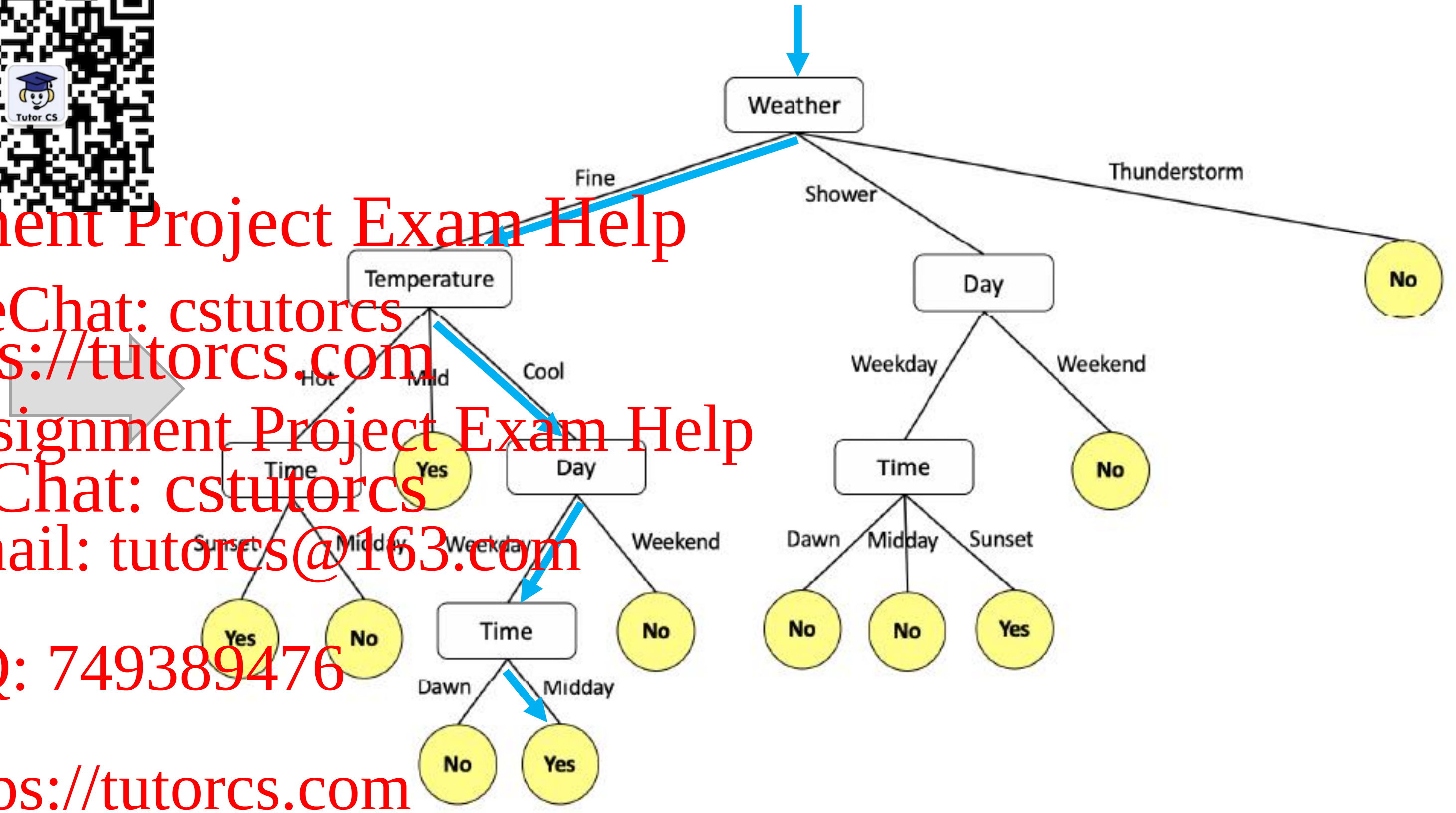
1.1 Traditional Data Mining Techniques: Classification

- Assigning new instances (objects) to predefined categories or classes.
- The most popular method is **Decision Tree**.
  - Create a set of rules that could be used to differentiate one target class from another.
  - Contains Training Dataset and Testing Dataset.
  - The target class is labelled with categorical values.
  - Depends on the root node

1.1 Traditional Data Mining Techniques: Classification (Decision Tree)

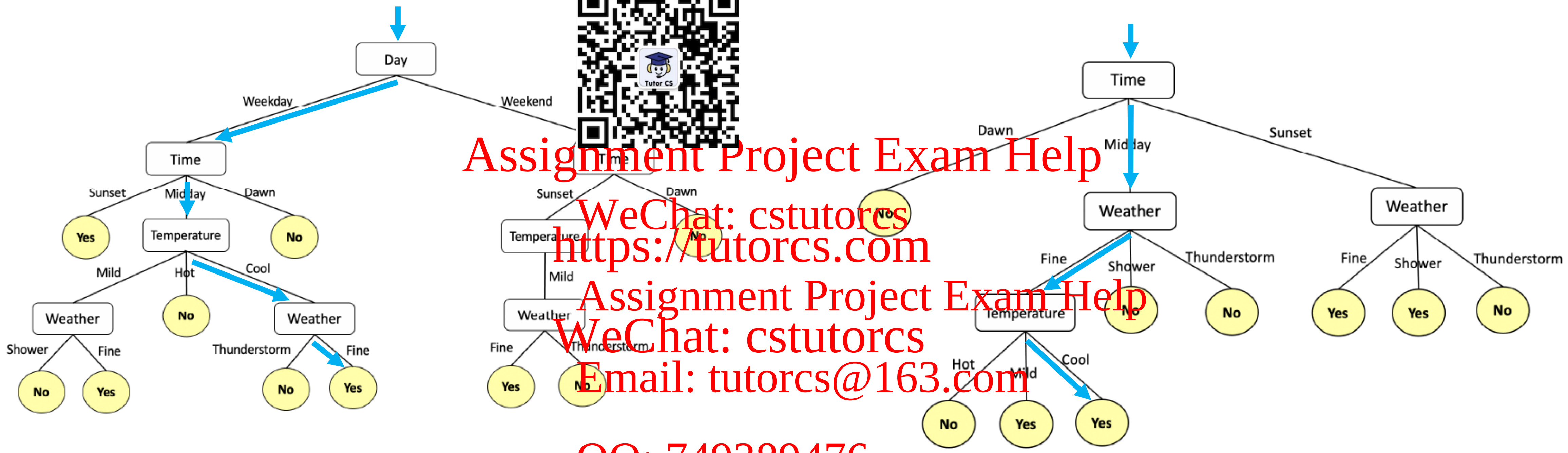
Table 1.4: Decision Training Dataset

No	Weather	Temperature	Time	Day	Walk (Target C
1	Fine	Hot	Sunset	Weekday	Yes
2	Fine	Mild	Sunset	Weekend	Yes
3	Shower	Mild	Midday	Weekday	No
4	Thunderstorm	Mild	Sunset	Weekend	No
5	Fine	Mild	Midday	Weekday	Yes
6	Shower	Hot	Sunset	Weekday	Yes
7	Fine	Cool	Dawn	Weekend	No
8	Shower	Mild	Dawn	Weekday	No
9	Thunderstorm	Hot	Midday	Weekday	No
10	Thunderstorm	Cool	Dawn	Weekend	No
11	Fine	Hot	Midday	Weekday	No
12	Thunderstorm	Cool	Midday	Weekday	No
13	Fine	Cool	Midday	Weekday	Yes
14	Shower	Hot	Dawn	Weekend	No
15	Fine	Cool	Dawn	Weekday	No



If the weather is fine, the temperature is cool, the day is weekday and the time is midday, then walk is possible

1.1 Traditional Data Mining Techniques: Classification (Decision Tree)



If the weather is fine, the temperature is cool, the day is weekday and the time is midday, then walk is possible

1.1 Traditional Data Mining Techniques: Classification (Decision Tree)

Known Issues:

1. Categorical Data

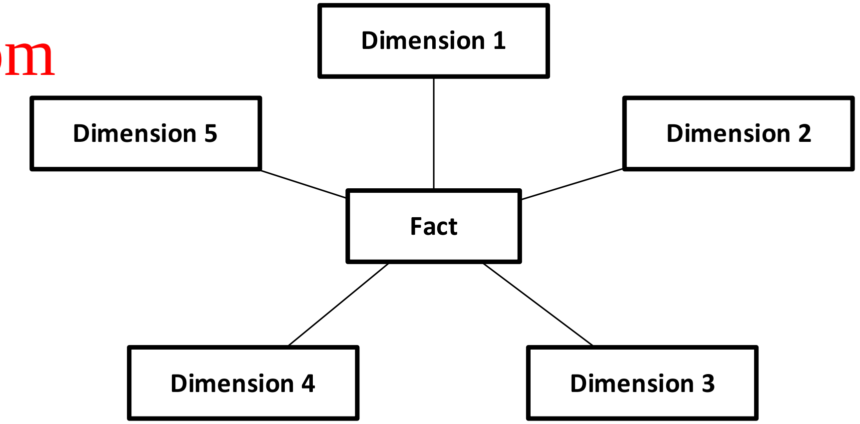
   Fact measures in star schema contains only numerical values.

2. Target Class

   The target class is a categorical value

1.2 Data Analytics Requirements in Data Warehousing

- Use the data in data warehouse using Star Schema.
- Fact measures contains numerical values
- Data analysis in data warehousing is data analysis of numerical values.
- Techniques:
    - Regression
    - Clustering
    - Classification



1.2 Data Analytics Requirements in Data Warehousing

Regression

Clustering and Classification

Dim1 (Timestamp)	Dim2	Dim3	Dim4	Fact Measure 1	Fact Measure 2	Fact Measure 3

Figure 1.5: Fact Table and Time-Series Regression

Dim1	Dim2	Dim3	Dim4	Fact Measure 1	Fact Measure 2	Fact Measure 3

Figure 1.6: Data Analysis of Fact Measures

2 Statistical Method: Regression

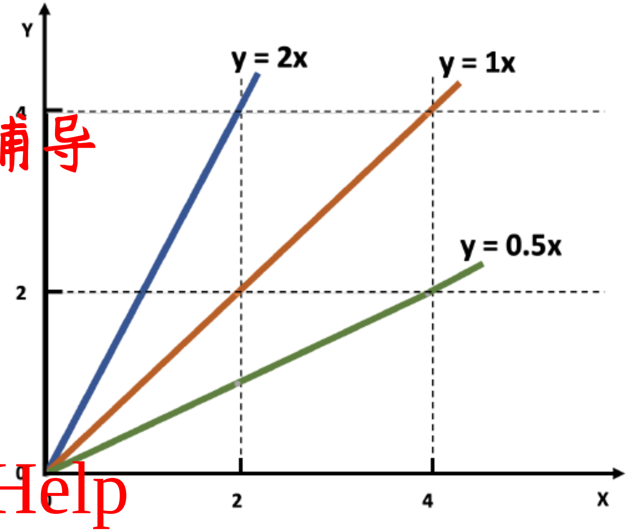
- Estimate the relationship between a dependent variable (**outcome variable**), and one or more independent variables (**predictors/covariates/features**).
- Purposes:
  - For prediction and forecasting
  - Causal relationships between the independent and dependent variables

2 Statistical Method: Regression

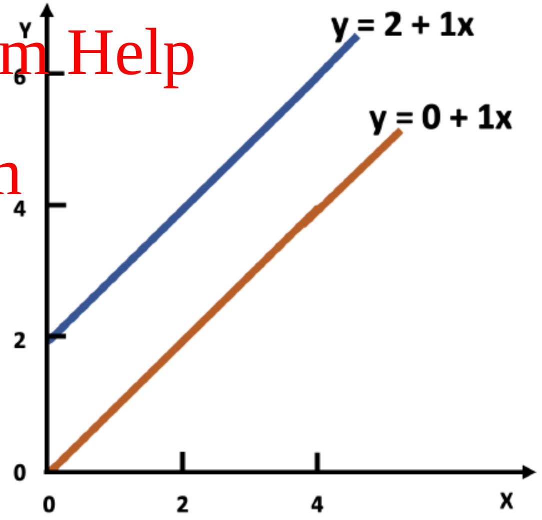
- Regression Types:
  - Simple Linear Regression
  - Polynomial Regression
- Based on Fact measures:
  - Time Series
  - Non Time-Series

2.1 Simple Linear Regression

- Find the equation of a line that is the best fit for a series of data:
  - Slope: line gradient
  - Intercept: height of the line
- The relation between dependent and independent variables to be linear



$y = slope \times x$



$y = intercept + (slope \times x)$

2.1 Simple Linear Regression: Example 1

$$\text{slope} = \frac{\sum(x_i - \bar{x})(y_i - \bar{y})}{\sum(x_i - \bar{x})^2}$$

$$\text{intercept} = \bar{y} - b_1\bar{x}$$

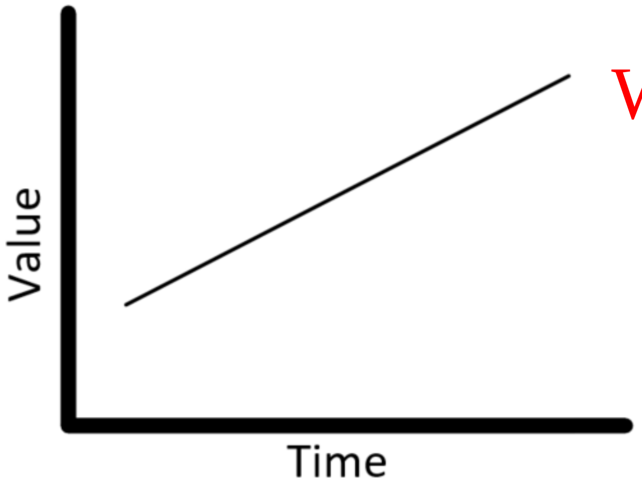


Table 1.5: A Sample Dataset for Regression

Time	Value
1	11
2	27
3	34
4	38
5	45
6	61
7	63

2.1 Simple Linear Regression: Example 1 SQL

Slope and Intercept

```
select slope, y_bar_max - (slope * x_bar_max) as intercept
from (
  select
    sum((x - x_bar) * (y - y_bar)) /
    sum((x - x_bar) * (x - x_bar)) as slope,
    max(x_bar) as x_bar_max,
    max(y_bar) as y_bar_max
  from (
    select avg(x) as x_bar, avg(y) as y_bar
    from dataset) av, dataset
```

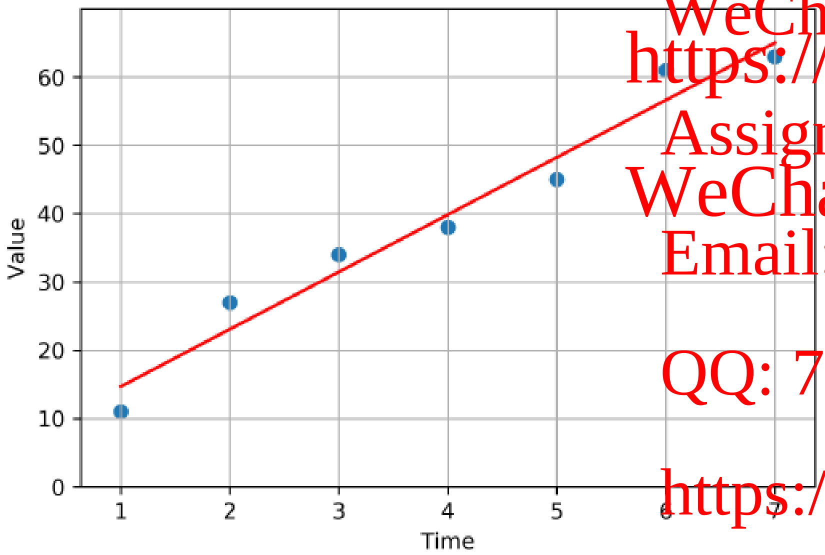
Table for Regression Model

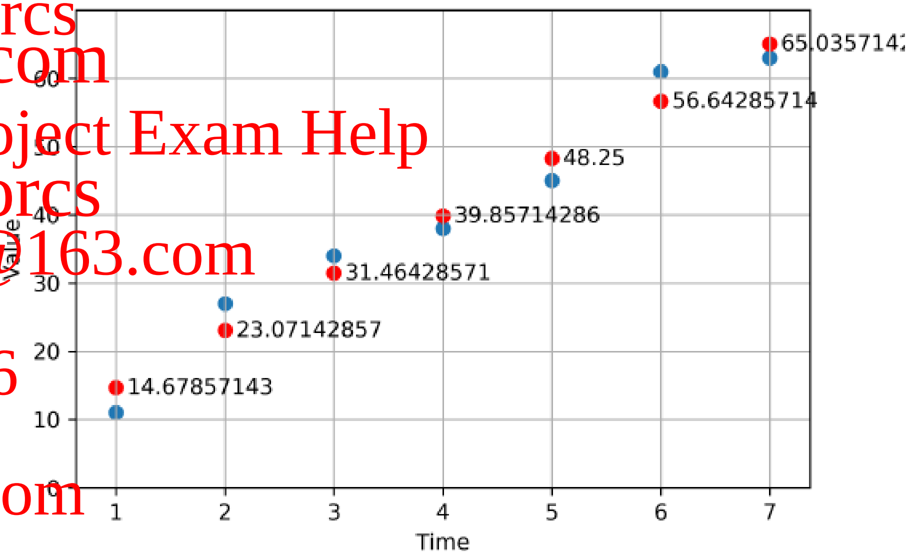
```
create table linear_regression as
select x, y, (intercept + (slope * x)) as y_pred
from
  dataset,
  (select slope, y_bar_max - (slope * x_bar_max) as intercept
  from (
    select
      sum((x - x_bar) * (y - y_bar)) /
      sum((x - x_bar) * (x - x_bar)) as slope,
      max(x_bar) as x_bar_max,
      max(y_bar) as y_bar_max
    from (
      select avg(x) as x_bar, avg(y) as y_bar
      from dataset) av, dataset
  ))
order by x;
```

2.1 Simple Linear Regression: Example 1

Linear Regression as Lin

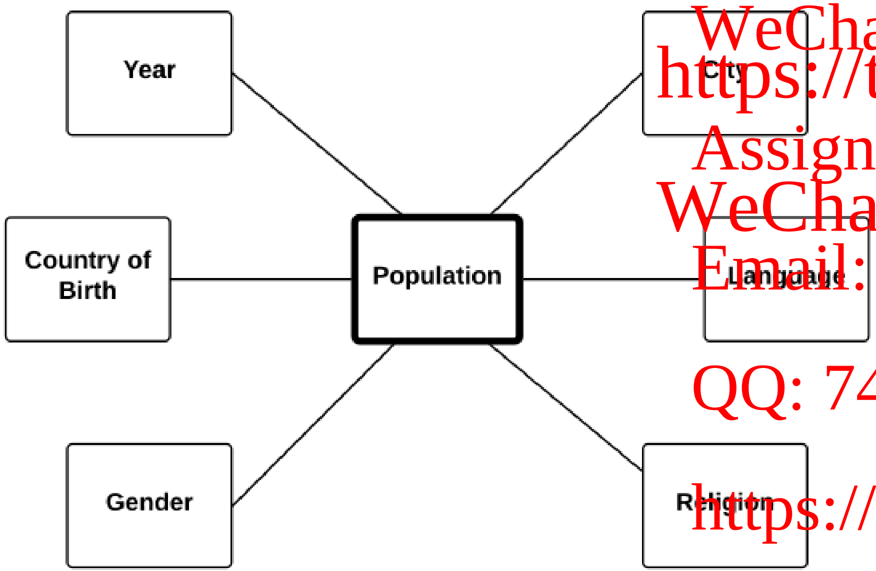
Linear Regression as Discrete Values

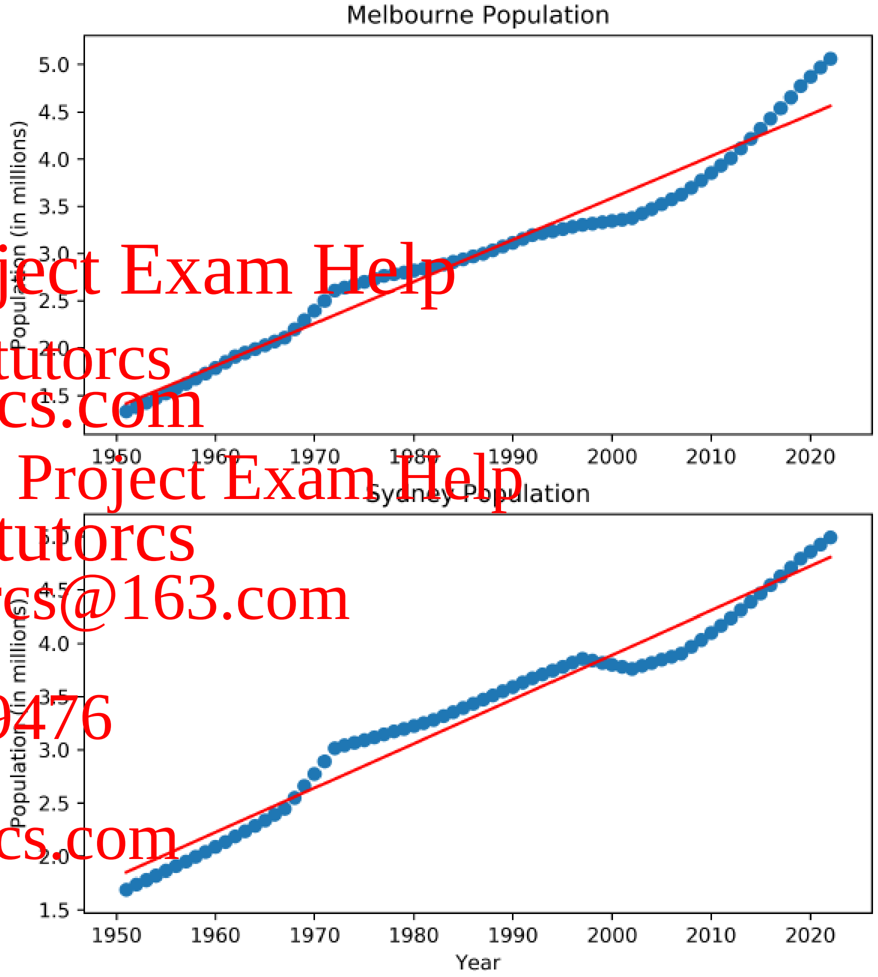




2.1 Simple Linear Regression: Example 1

Population Star Schema



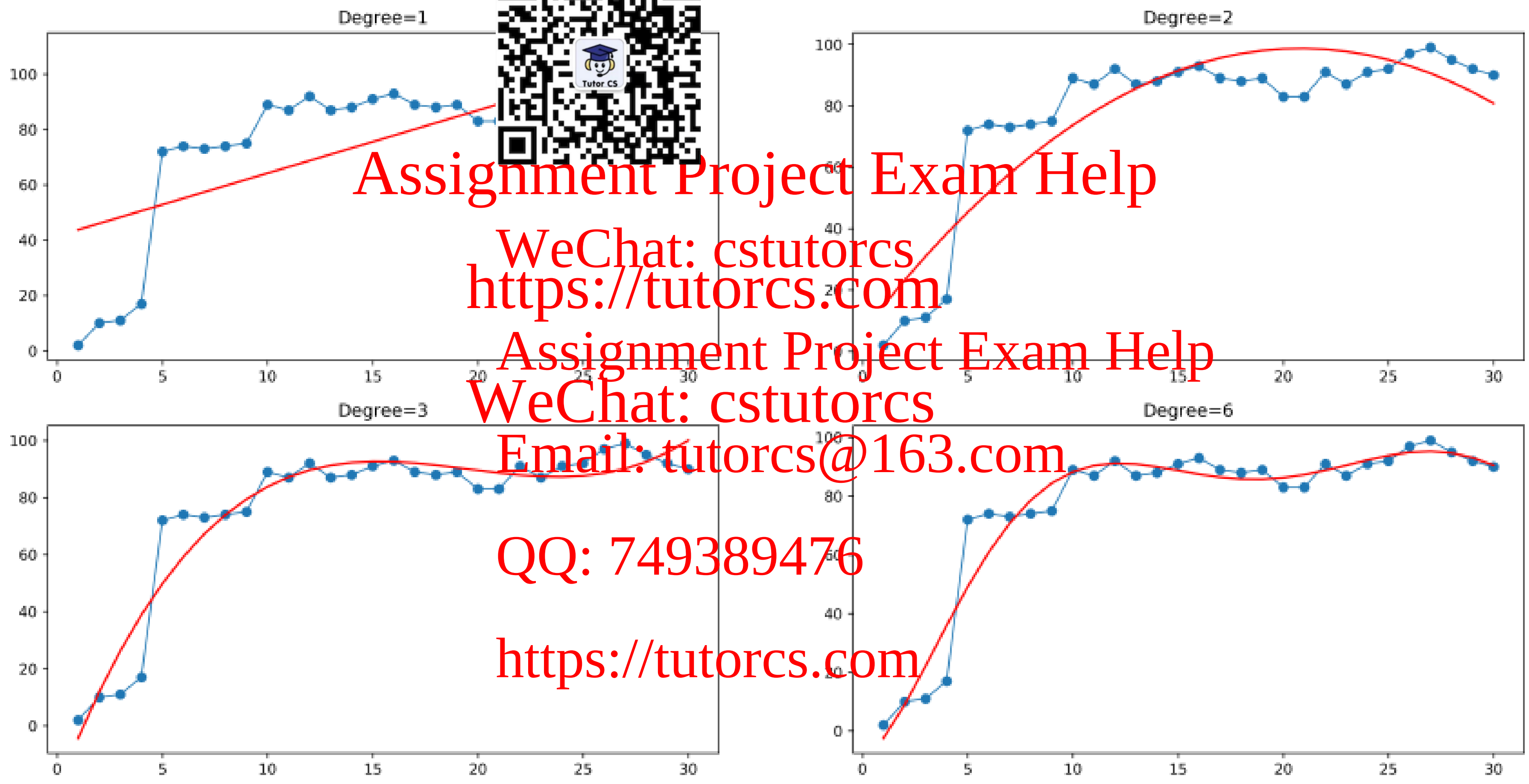


2.2 Polynomial Regression

- Data distribution is more complex and may not be linear.
- The degree is indicated by n
- The model is represented by curve.

$$y = b_0 + (b_1 \times x) + (b_2 \times x^2) + (b_3 \times x^3) + \ldots + (b_n \times x^n)$$

2.2 Polynomial Regression: Example

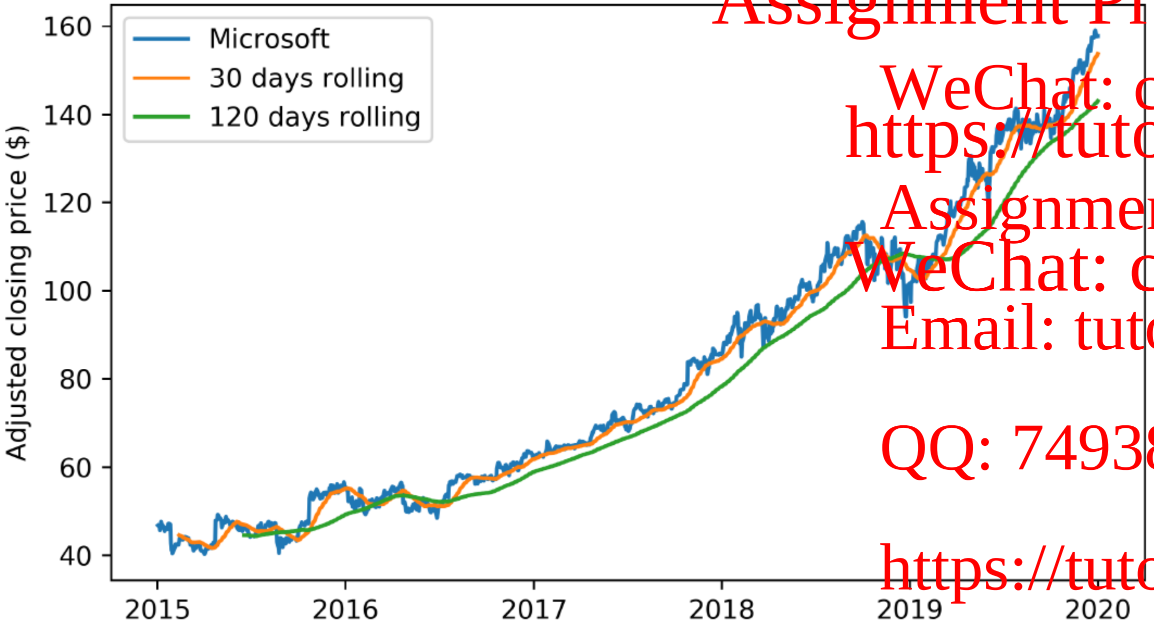


2.3 Rolling Windows vs. Regression

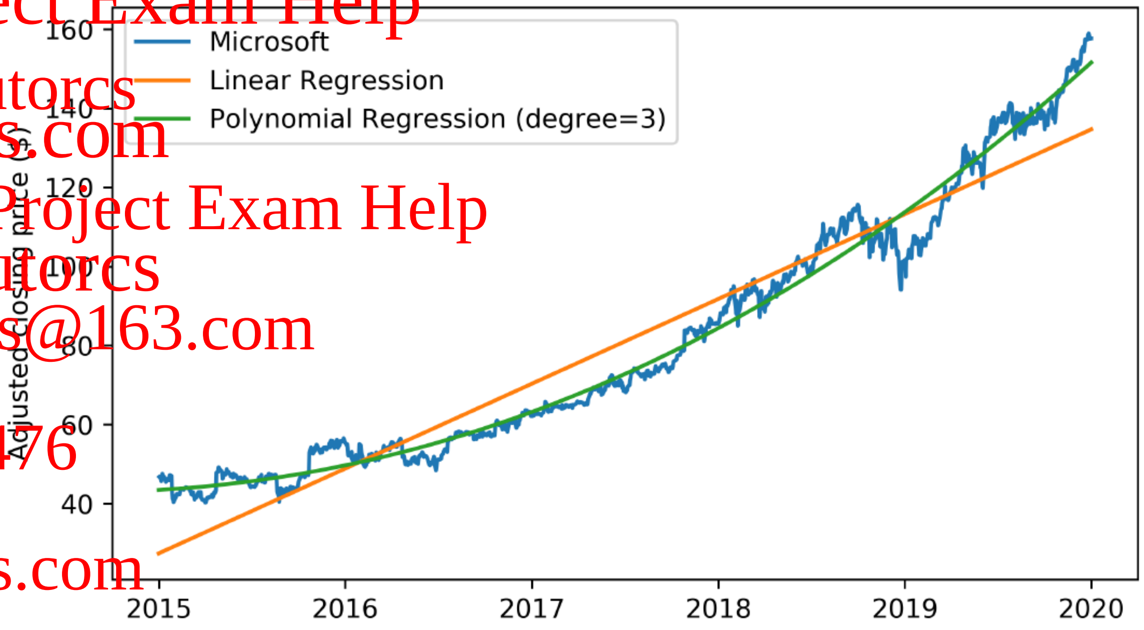
- Rolling Windows is a [illegible] to smooth the original data by providing average val[illegible] limited window.
- Window size will determine the smoothness of the graph representation

2.3 Rolling Windows vs. Regression: Example 1

Regression Model

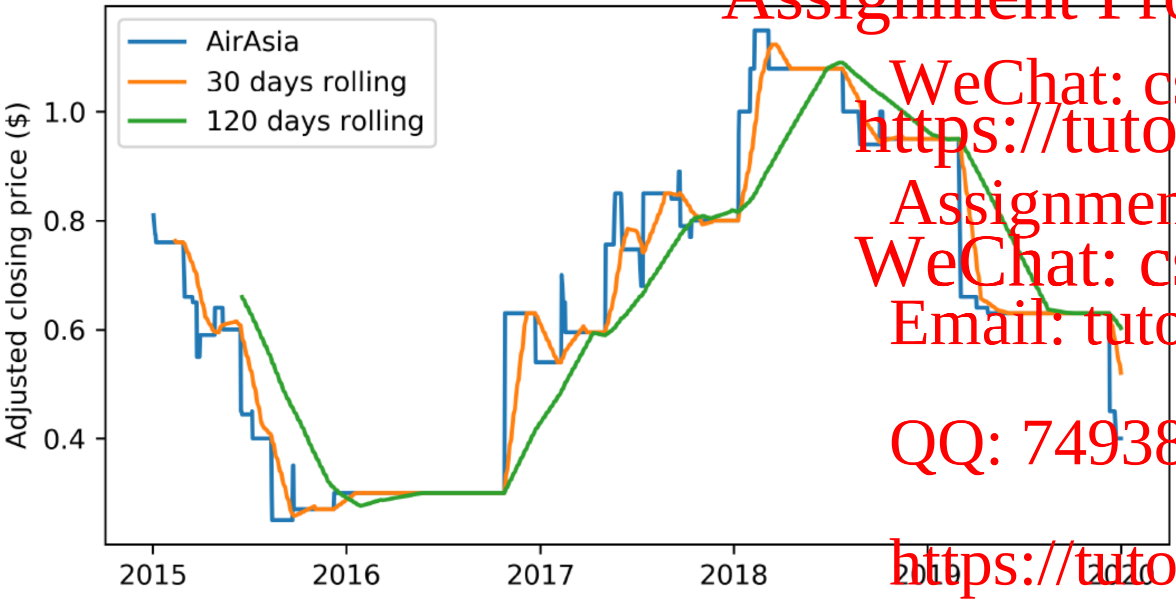


Rolling Windows Model

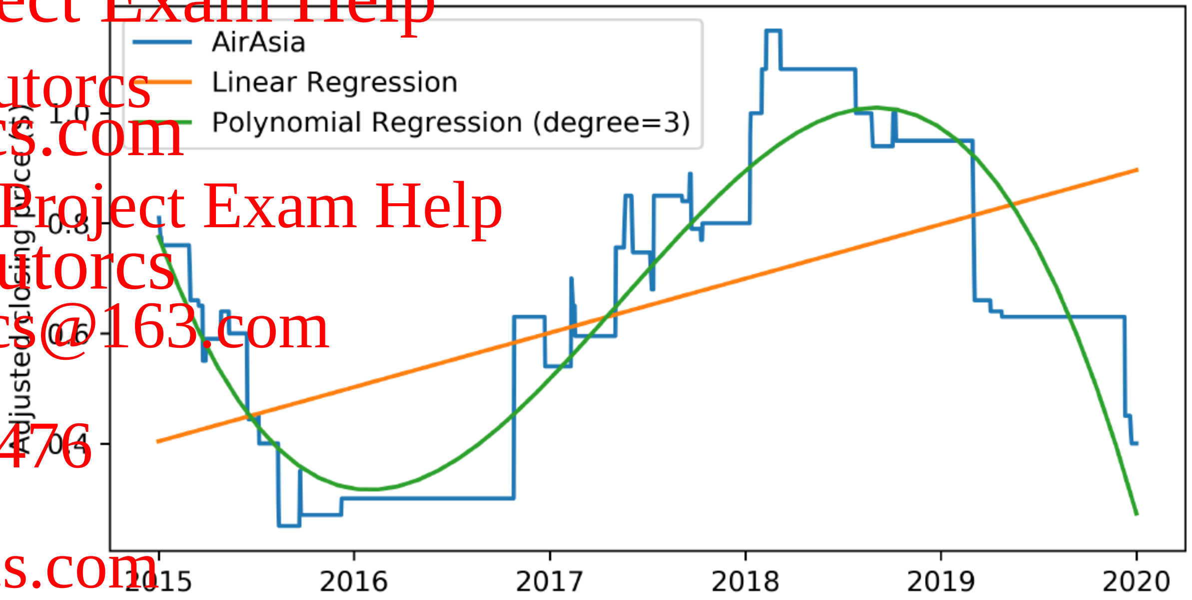


2.3 Rolling Windows vs. Regression: Example 2

Regression Model



Rolling Windows Model

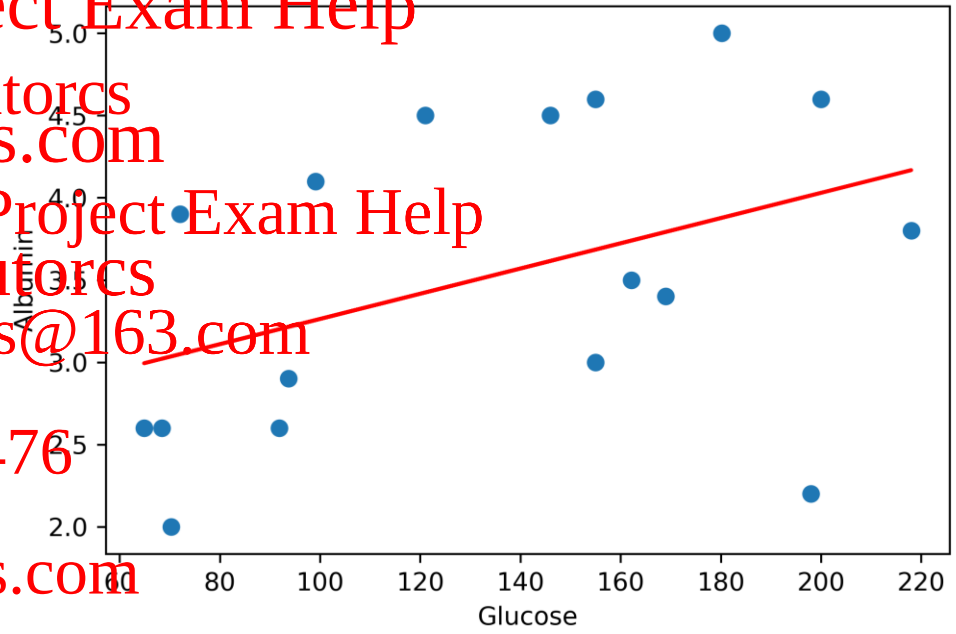


2.3 Non Time-Series Regression

- Prediction may not use time-series data.
- Both independent and dependent variables can be non temporal
- Not a temporal-based relationship
- Future data arrive near and around the line → high accuracy
- Future data arrive in a distance from the prediction line → low accuracy

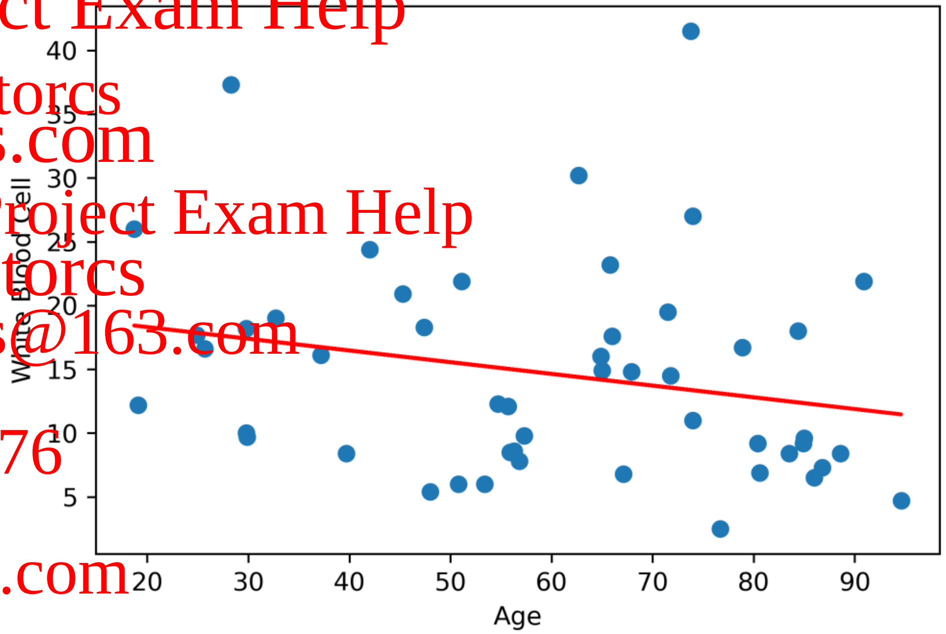
2.3 Non Time-Series Regression: Example 1

- Glucose and Albumin
- Regression line shows an increasing trend of Glucose-Albumin



2.3 Non Time-Series Regression: Example 2

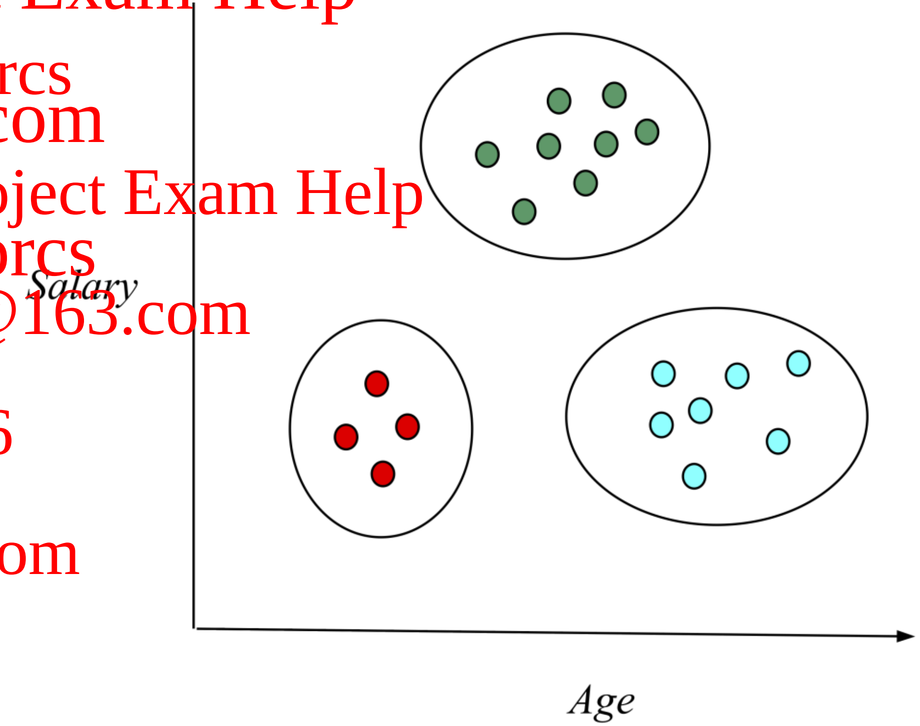
- Age and WBC (White Blood Cell).
- Regression line shows a decreasing trend



3 Clustering Analysis

- Finding groups or clusters in data.
- Members within a cluster are considered to be closer or similar
- Distance formula is used to define the closeness
- No category label
- unsupervised learning

$$dist(x_i, x_j) = \sqrt{\sum_{k=1}^{h}(x_{ik} - x_{jk})^2}$$



3 Clustering Analysis

Two types of Clustering Analysis

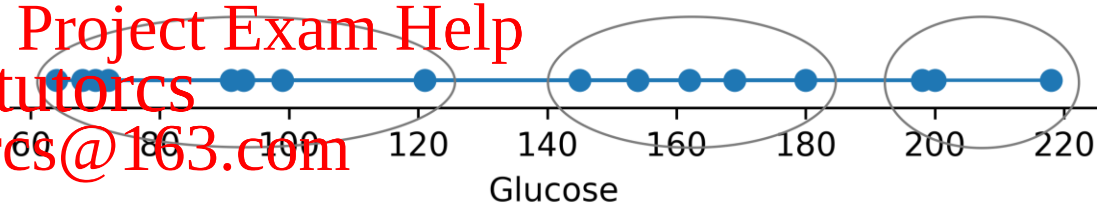
- Centroid-based clustering
  - The number of desired clusters is predefined
  - To divide the objects into the predefined number of clusters
- Density-based clustering
  - Does not require to predefine the number of clusters
  - To determine the ideal number of clusters

3.1 Centroid-based Clustering

- Objects are mutual-exclusively partitioned into the predefined clusters
- Each cluster has a centroid
- Objects will be assigned to the nearest centroid
- Example: *k*-means clustering

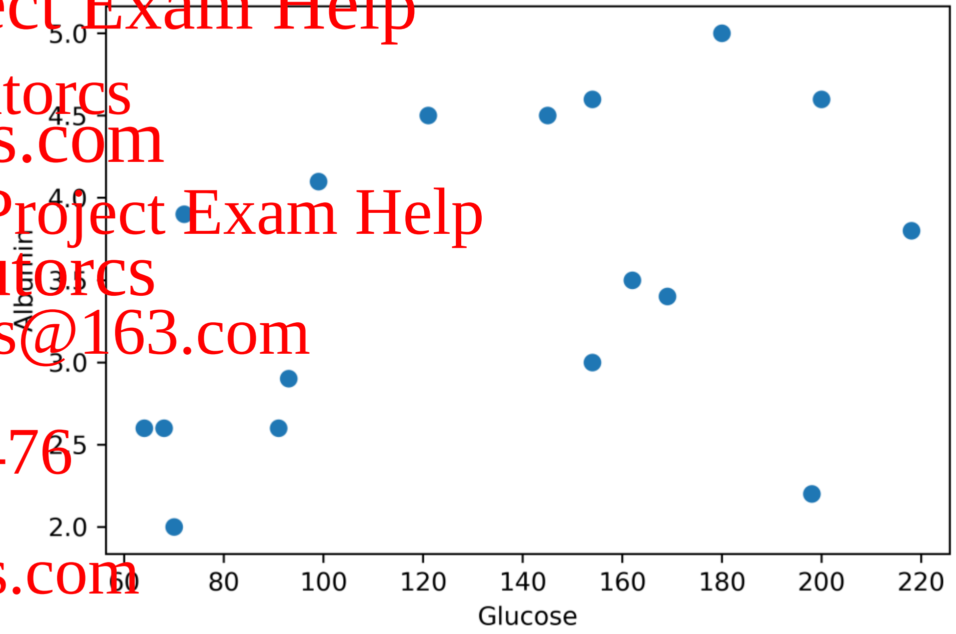
3.1 Centroid-based Clustering: *k*-means: Example 1

- Glucose from 17 patie records
- k=3
- Initial centroids:
  - m_1=162
  - m_2=169
  - m_3=200.



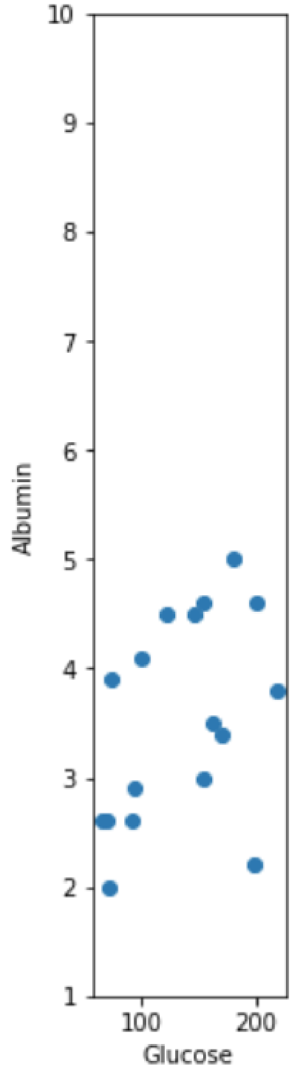
3.1 Centroid-based Clustering: *k*-means: Example 2

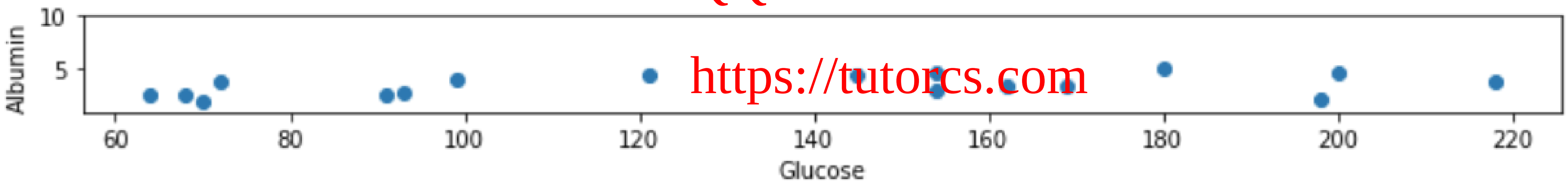
- Glucose and Albumin ... ?
  patient records.
- Glucose and Albumin are fact
  measures



3.1 Centroid-based Clustering: *k*-means: Example 2

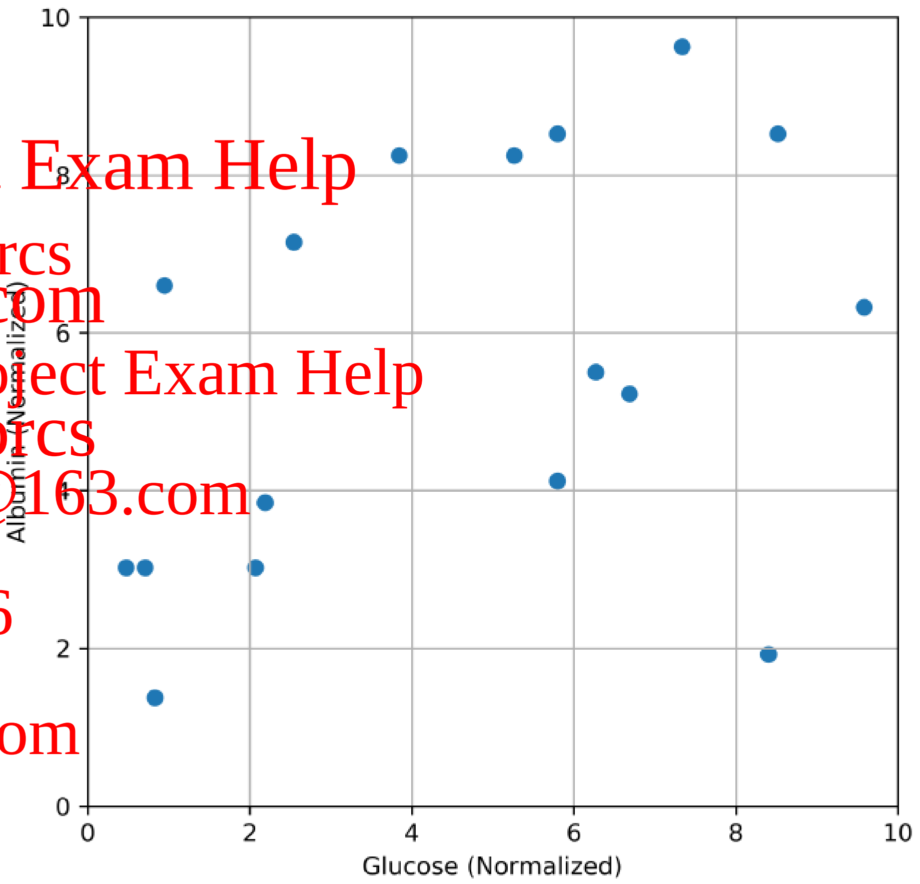
- Non-uniform unit measurement problems
  - Two different stretching direction
  - May produce different clusters
  - Non-uniform distance unit





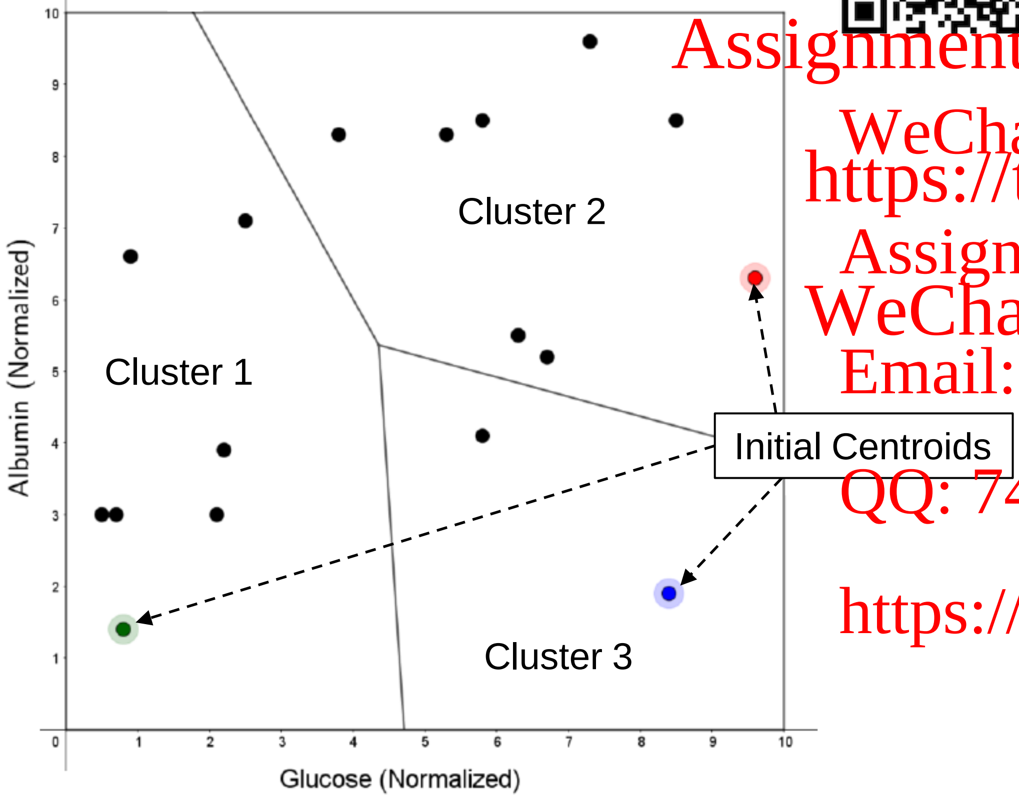
3.1 Centroid-based Clustering: *k*-means: Example 2

- Normalized distance (
- k=3
- Random initial centroid
- Use Voronoi Diagram

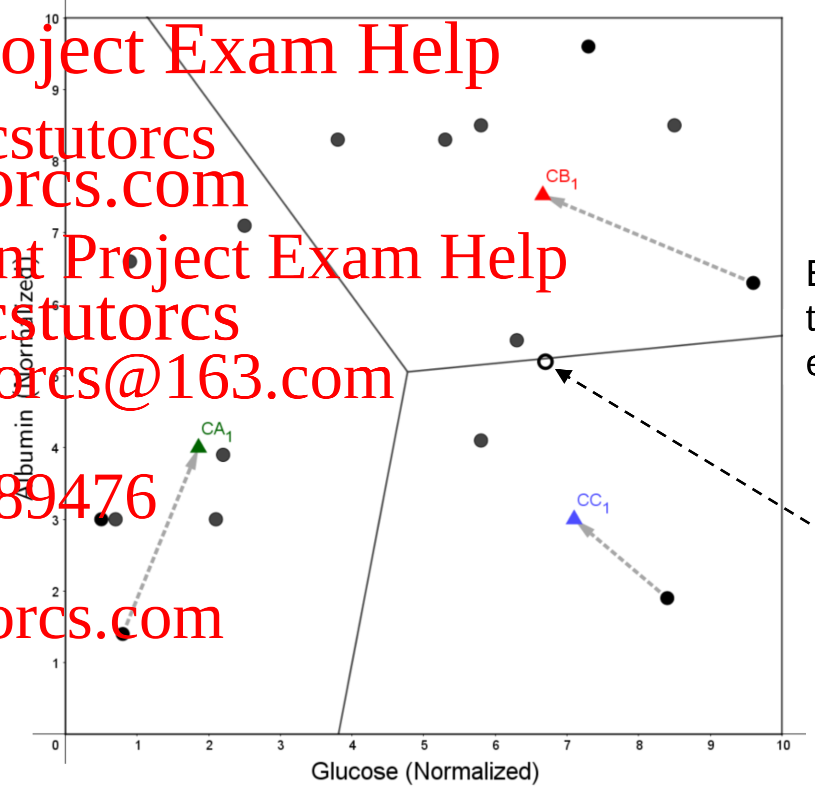


3.1 Centroid-based Clustering: *k*-means: Example 2

Initial Clusters



Step-1

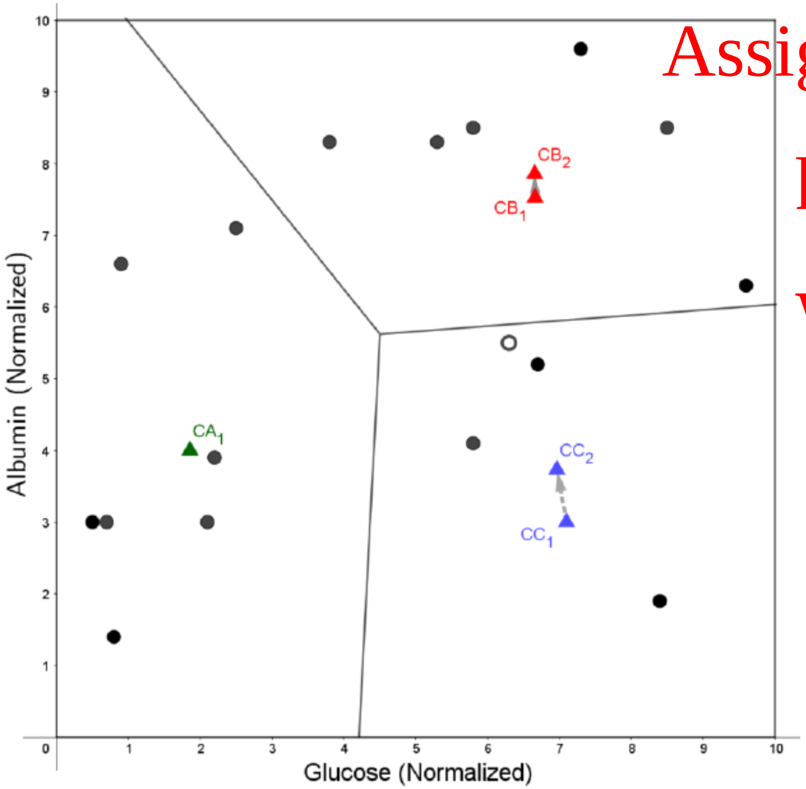


Each cluster will recalculate the new centroids from existing members

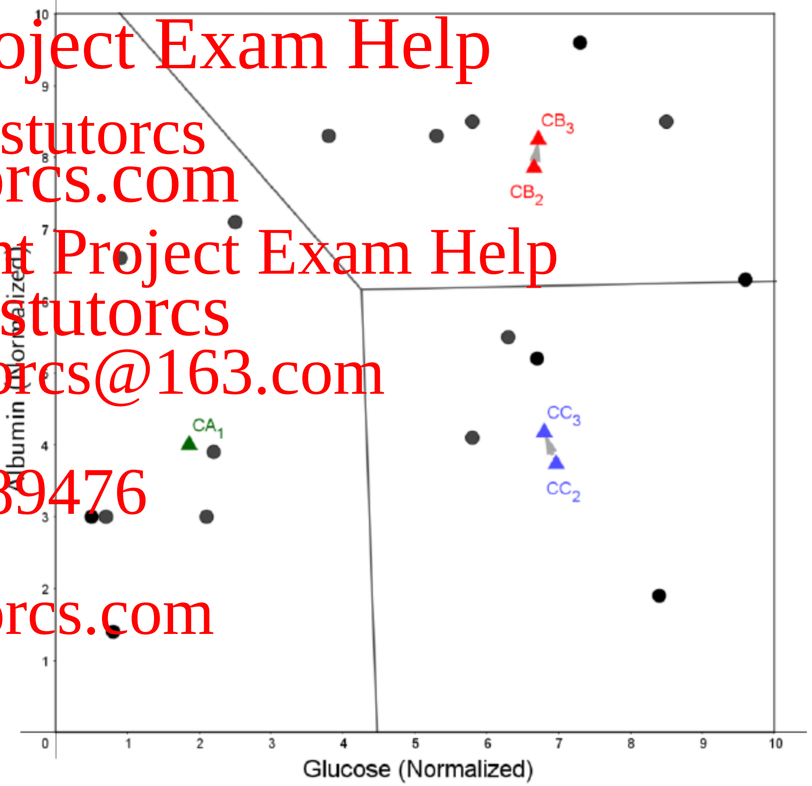
Object movement

3.1 Centroid-based Clustering: *k*-means: Example 2

Step-2

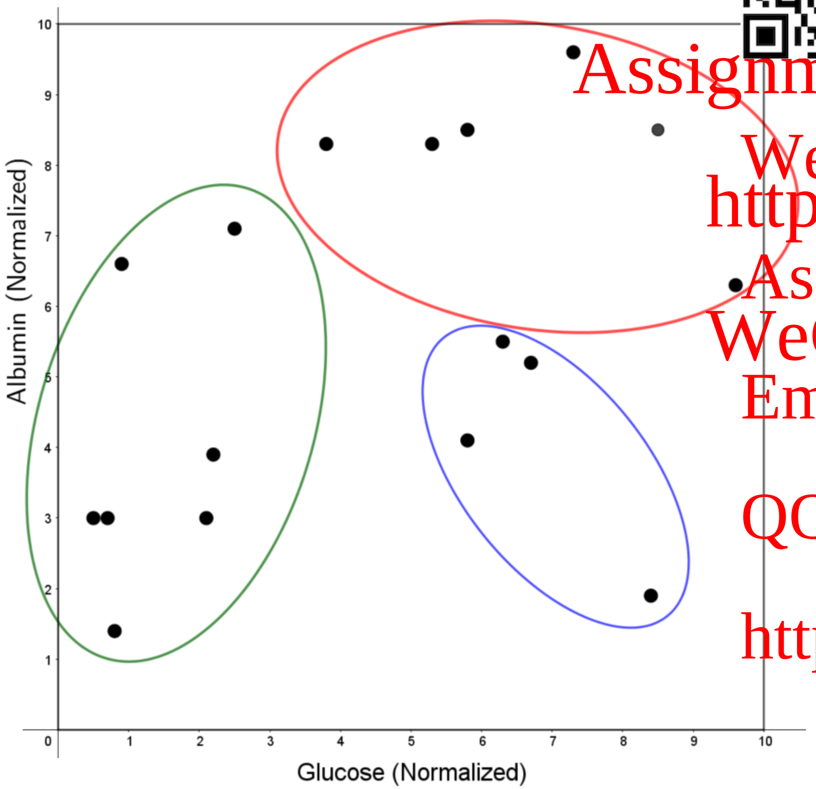


Step-3



3.1 Centroid-based Clustering: *k*-means: Example 2

Result



- The process terminates when no more member movements from the clusters

3.2 Density-based Clustering

- To create a chain of objects in a neighbourhood
- The feature of this cluster is that it is dense
- Tight proximity between one object and another
- Popular method: DBSCAN

3.2 Density-based Clustering: Example

- Five important elements of DBSCAN:
  1. MaxDist
  2. MinPts
  3. Core points
  4. Border points
  5. Outliers.

Clustering Analysis: Summary

k-Means Clustering

- Grouped based on the coverage of the cluster
- Requires predefined k
- All objects will get a cluster
- Will not produce outliers
- Will not produce a long chain of objects in the neighborhood

DBSCAN

- Grouped based on the specific distance range
- Does not require predefined k
- Objects may have no clusters
- May produce outliers
- May produce a long chain of objects in the neighborhood

4 Classification using Regression Trees

- Similar with Decision T
- Works with continuous/numerical values
- Built using a training dataset
- Predicting the target class of incoming data can use the regression model

4 Classification using Regression Trees

- The main process of Regression Trees:
  1. Selecting Root Node
  2. Repeat
     a) Processing the Left Sub-Tree
     b) Processing the Right Sub-Tree
  3. Finalizing the Regression Tree
- Termination Condition
  i. The objects within a partition are cohesive enough.
  ii. The number of objects in a partition is very small

4 Classification using Regression Trees: Example

- 17 patients data
- Fact measures:
  - Glucose
  - Albumin
- Target class:
  - Mortality Prediction

Table 1.6: Emergency Patient Extended Fact Table

Patient ID	...	Glucose	Albumin	...	...	Mortality Prediction
A		162.2	3.5			0.573189504
B		93.7	2.9			0.22
C		68.5	2.6			0.217082562
D		155.0	3.0			0.534815242
E		121.0	4.5			0.475139465
F		198.0	2.2			0.969279952
G		99.1	4.1			0.552492172
H		180.2	5.0			0.752011091
I		169.0	3.4			0.799263517
J		64.9	2.6			0.383771494
K		72.1	3.9			0.45
L		155.0	4.6			0.862259153
M		218.0	3.8			0.99
N		146.0	4.5			0.813710339
P		91.9	2.6			0.496873792
Q		200.0	4.6			0.745266898
R		70.3	2.0			0.132516482

4.1 Selecting the Root Node

- Two aspects in selecting the root node:
  - Which attribute to be used as the root node?
    What condition to be applied to the branches of the root node?
- If the partition has large differences in their target value → sub-optimal partition
- Preserves Cohesiveness using ***Sum of Squared Residual (SSR)***

4.1 Selecting the Root Node

- A Residual value:
  The difference between object and the average value of all objects in the same partition.
- Residual is squared → the difference between each object and it's average is always a positive value.
- The lowest SSR indicates the best partitioning method.

$$SSR = \sum_{i=1}^{n}(r_i - \bar{r})^2 + \sum_{j=1}^{m}(s_j - \bar{s})^2$$

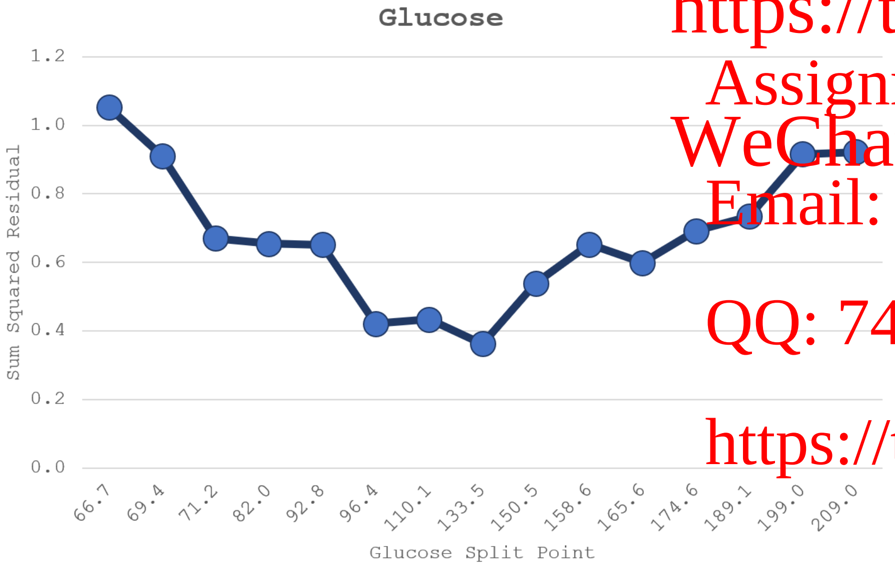
4.1 Selecting the Root Node: Example

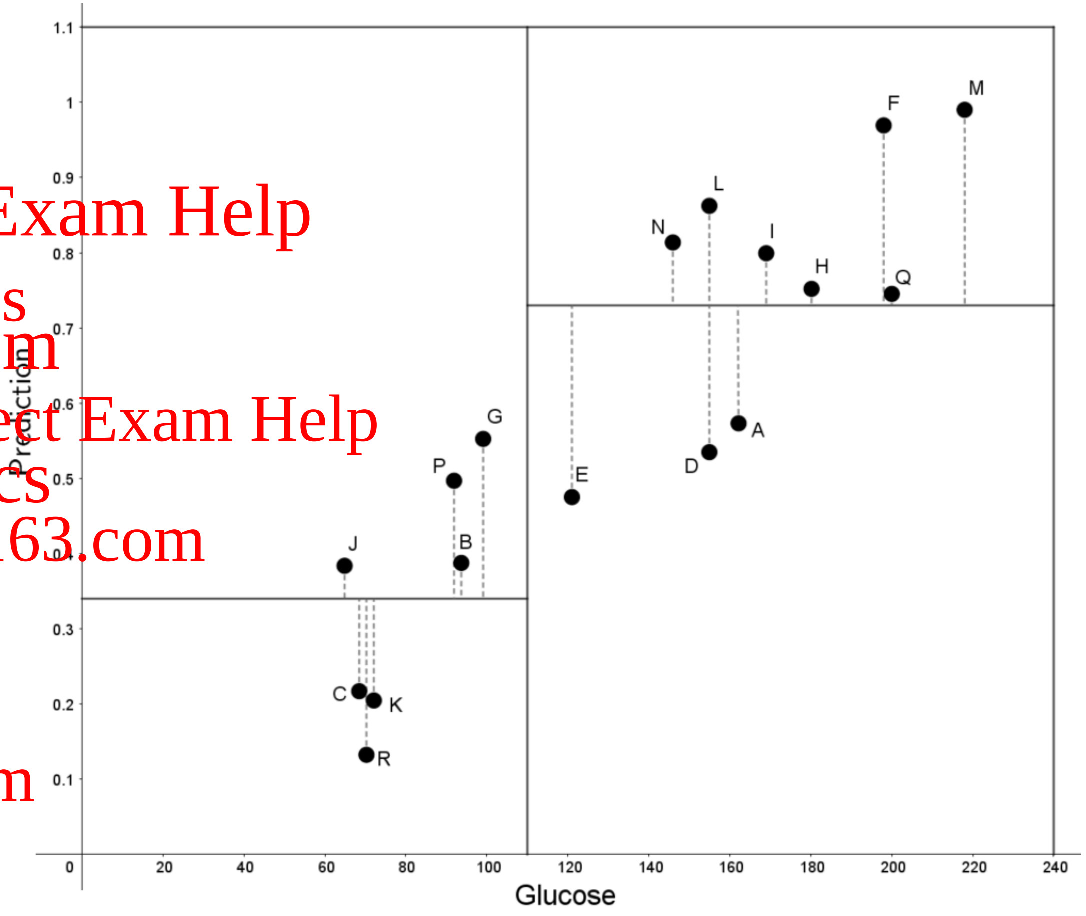
- For Glucose partition:
  - 15 SSR
  - 16 distinct value
- For Albumin partition:
  - 12 SSR
  - 13 distinct value

Patient ID	...	Glucose	Albumin	...	Mortality Prediction
A		162.2	3.5		0.573189504
B		93.7	2.9		0.387614064
C		68.5	2.6		0.217082562
D		155	3.0		0.534815242
E		121	4.5		0.475139465
F		198	2.2		0.969279952
G		99.1	4.1		0.552492172
H		180.2	5.0		0.752011091
I		169	3.4		0.799263517
J		64.9	2.6		0.383771494
K		72.1	3.9		0.204709509
L		155	4.6		0.862259153
M		218	3.8		0.99
N		146	4.5		0.813710339
O		70.3	2.0		0.132516482
P		91.9	2.6		0.496873792
Q		200	4.6		0.745266898

4.1 Selecting the Root Node: Example (Glucose)

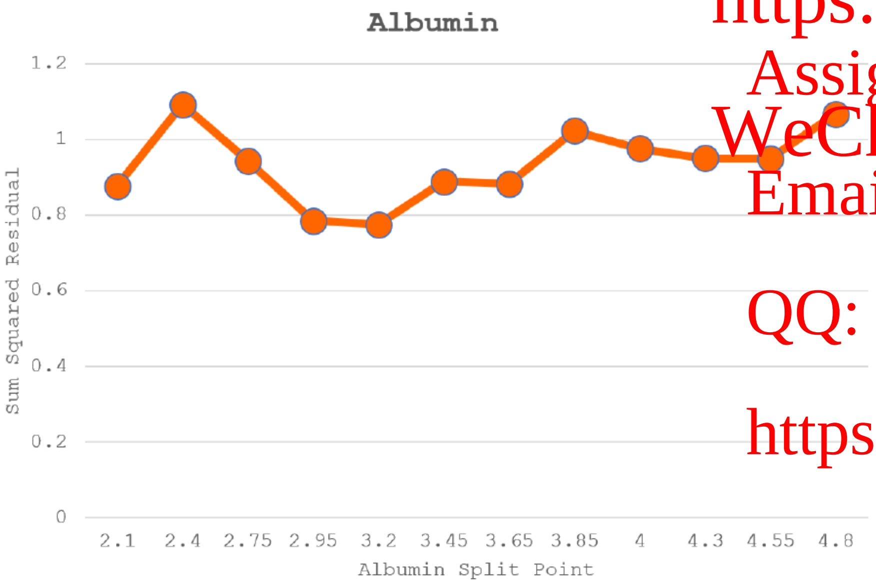
- The lowest SSR (**0.36**
- Between Patients E and
  →133.5

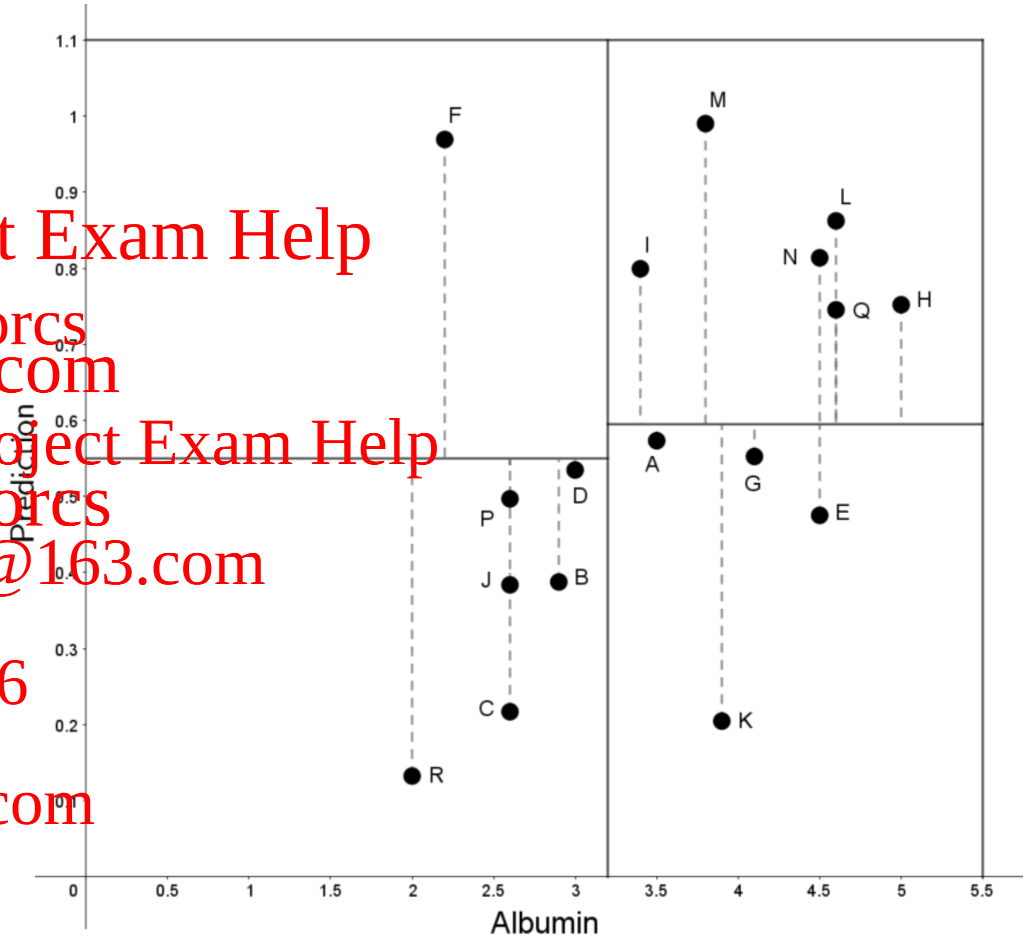




4.1 Selecting the Root Node: Example (Albumin)

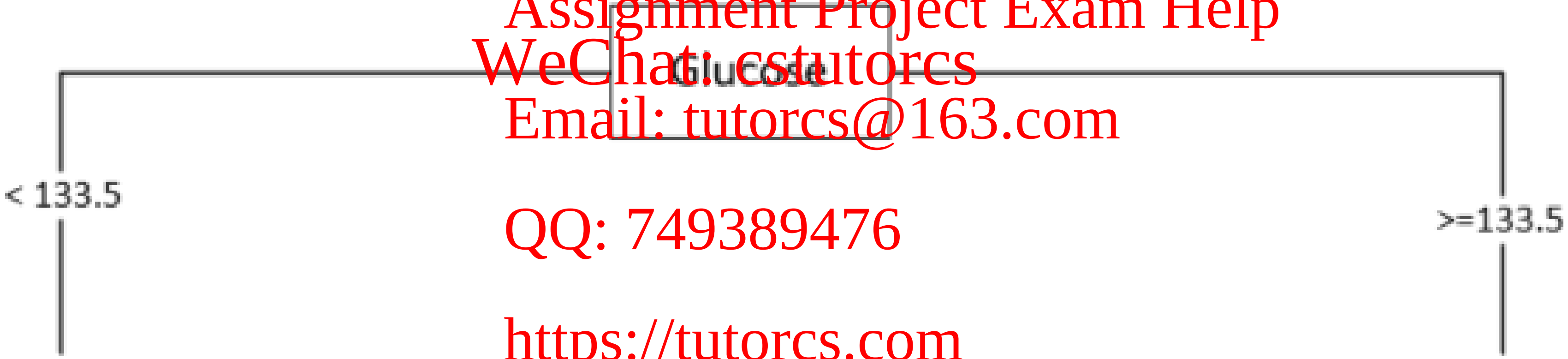
- The lowest SSR (**0.77**
- Between Patients D and
  →3.2



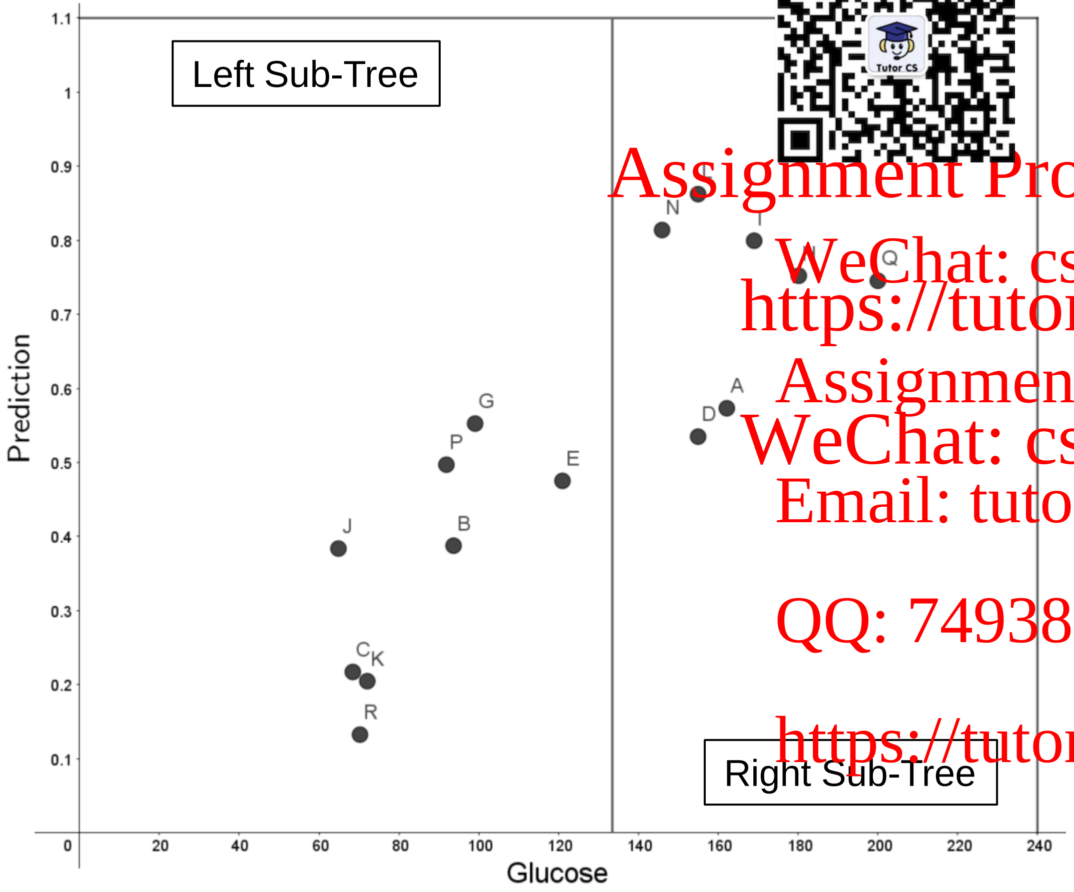


4.1 Selecting the Root Node: Example (Regression Tree)

- Min (SSR Glucose) (**0** [illegible] < Min (SSR Albumin) (**0.7748**)
- Root = Glucose



4.1 Selecting the Root Node: Example (Regression Tree)



Patient ID	...	Glucose	Albumin	...	Mortality Prediction
J		64.9	2.6		0.383771494
C		68.5	2.6		0.217082562
O		70.3	2.0		0.132516482
K		72.1	3.9		0.204709509
P		91.9	2.6		0.496873792
B		93.7	2.9		0.387614064
G		99.1	4.1		0.552492172
E		121	4.5		0.475139465
N		146	4.5		0.813710339
D		155	3.0		0.534815242
L		155	4.6		0.862259153
A		162.2	3.5		0.573189504
I		169	3.4		0.799263517
H		180.2	5.0		0.752011091
F		198	2.2		0.969279952
Q		200	4.6		0.745266898
M		218	3.8		0.99

Left Sub-Tree (rows J–E)

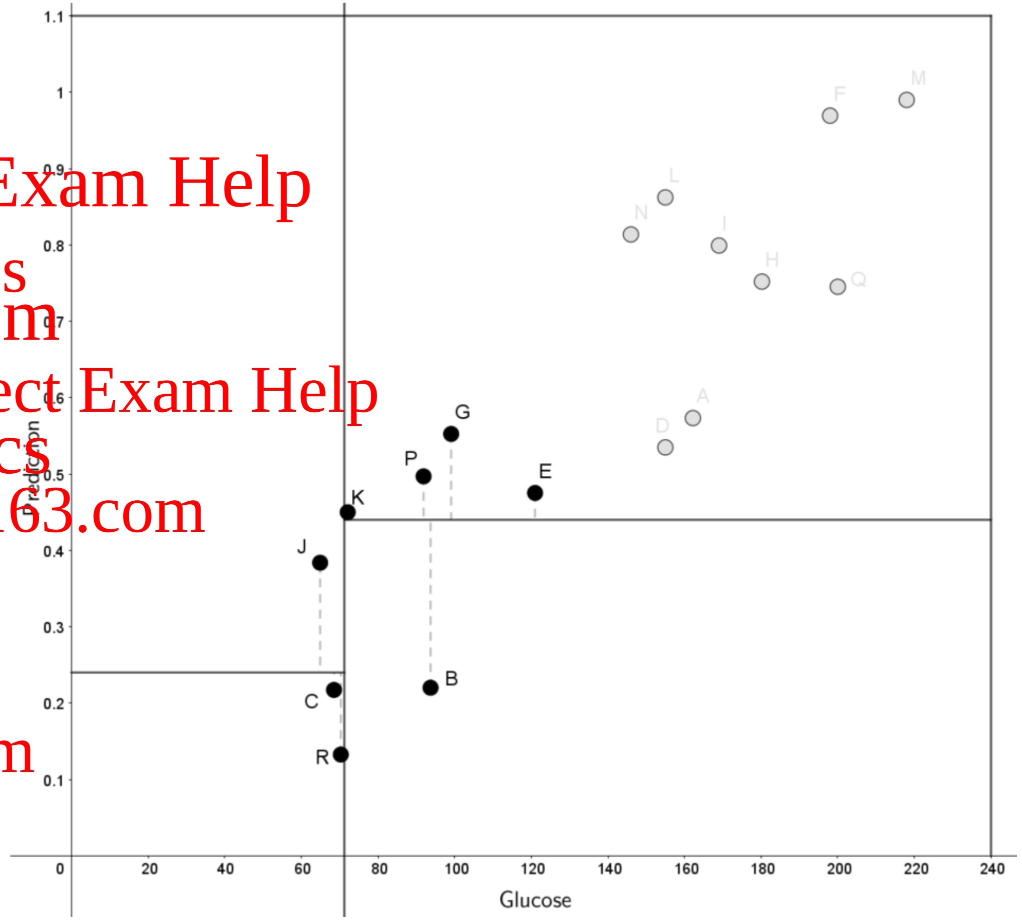
Right Sub-Tree (rows N–M)

4.2 Level 1: Processing the Left Sub-Tree

- Repeat the process on the Left Sub-Tree area partition
- Find the lowest SSR for Glucose and Albumin to be the splitting point on each dimension
- Splitting point = lowest SSR

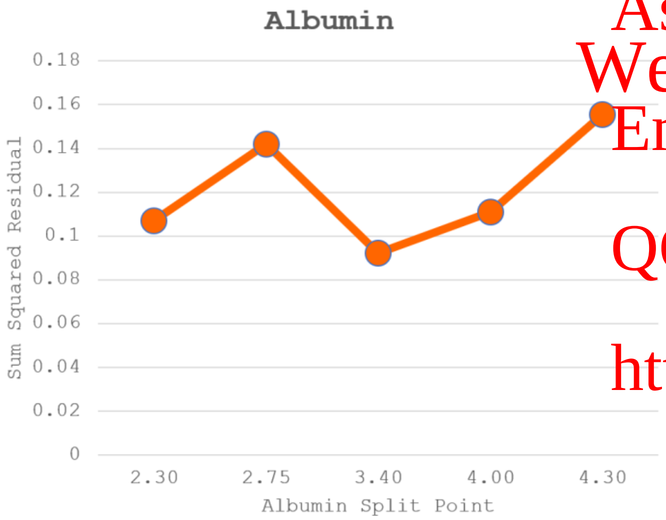
4.2 Level 1: Processing the Left Sub-Tree: Example (Glucose)

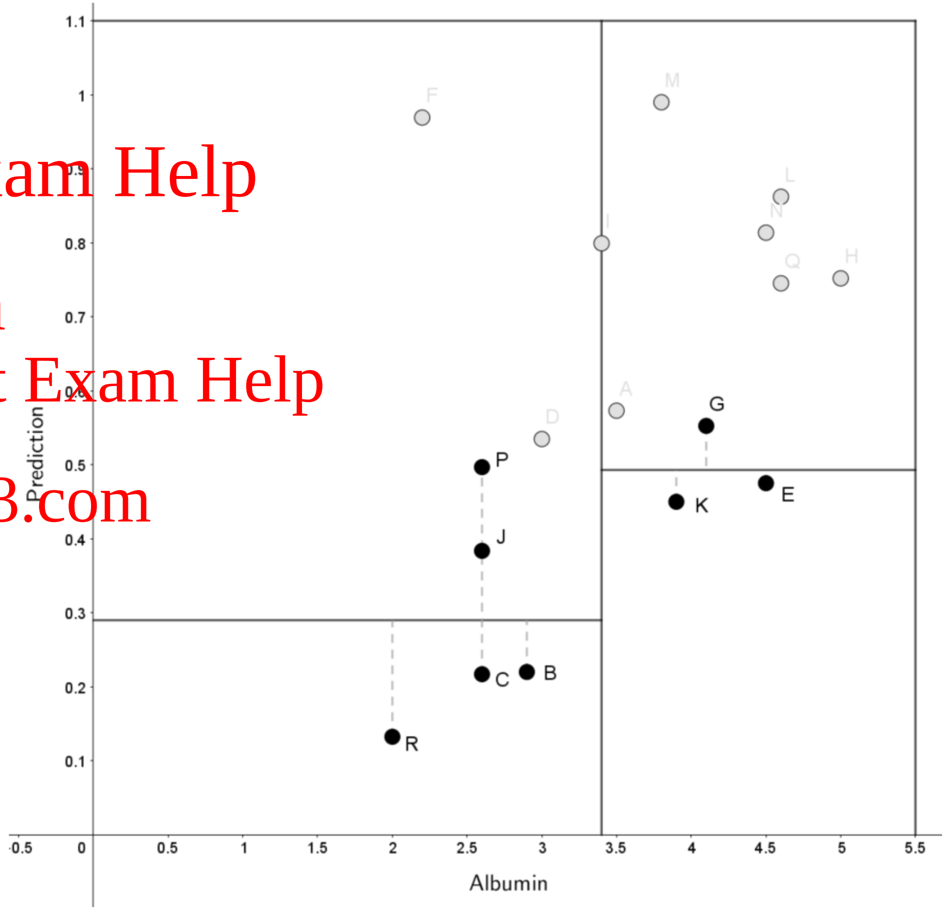
- The lowest SSR (**0.09**
- Between Patients R and
  →71.2



4.2 Level 1: Processing the Left Sub-Tree: Example (Albumin)

- The lowest SSR (**0.09**
- Between Patients B and
  →3.4





4.2 Level 1: Processing the Left Sub-Tree: Example (Regression Tree)

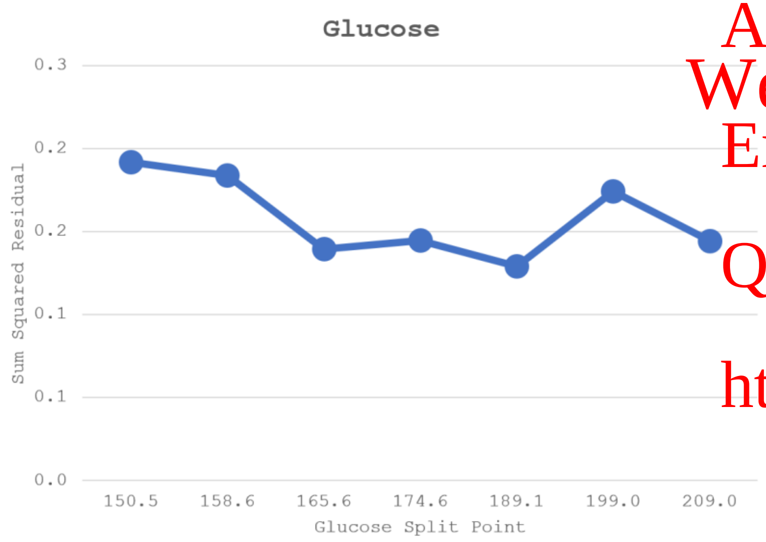
- Min (SSR Glucose) (**0**[illegible]) > Min (SSR Albumin) (**0.0923**)
- The Split Node = Albumin.

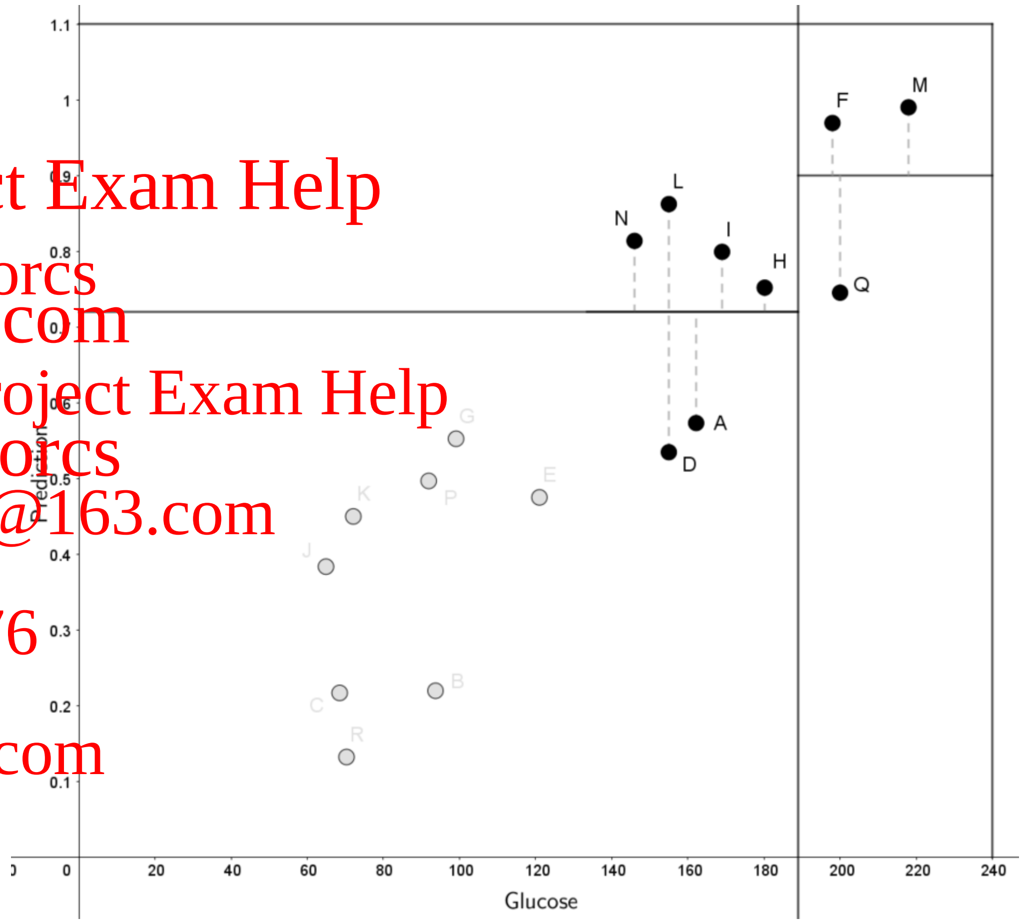
4.2 Level 1: Processing the Right Sub-Tree

- Repeat the process o[illegible] Sub-Tree area.
- Find the lowest SSR from Glucose and Albumin to be the splitting point on each dimension
- Lowest SSR = splitting point for Right Sub-Tree

4.2 Level 1: Processing the Right Sub-Tree: Example (Glucose)

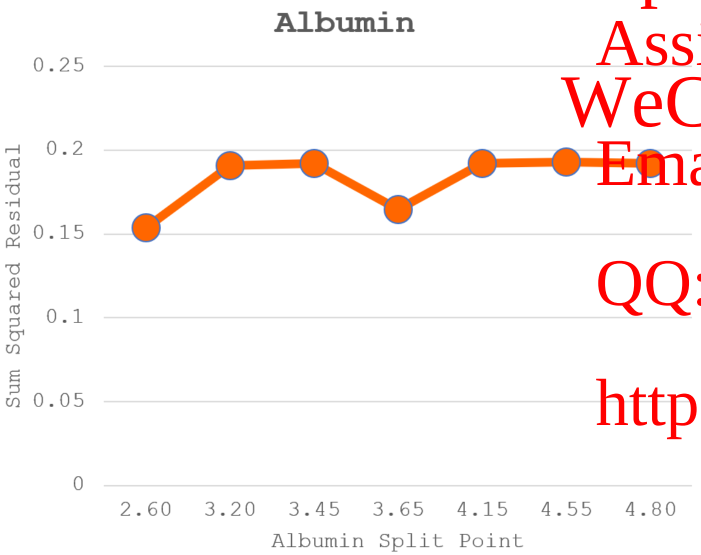
- The lowest SSR (**0.12**
- Between Patients H and
  →189

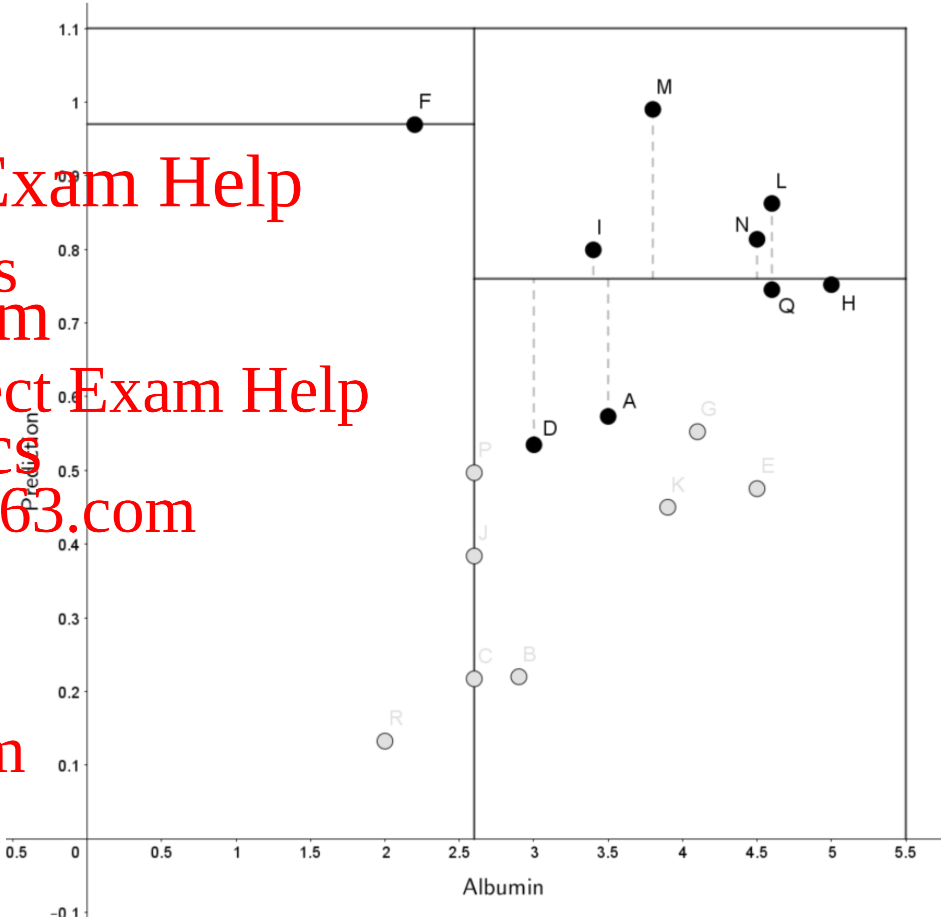




4.2 Level 1: Processing the Right Sub-Tree: Example (Albumin)

- The lowest SSR (**0.15**
- Between Patients B and
  →2.6





4.2 Level 1: Processing the Right Sub-Tree: Example (Regression Tree)

- Min (SSR Glucose) (0 Min (SSR Albumin) (**0.1537**)
- The Split Node = Glucose

4.4 Level 2: Finalizing the Regression Tree

- Last partition
- Glucose < 189 on the right sub-tree

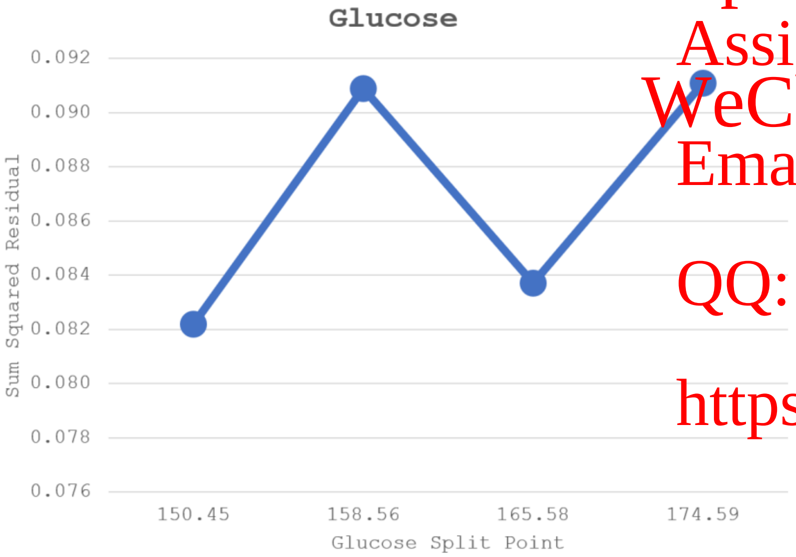
Patient ID	...	Glucose	Albumin	...	Mortality Prediction
J		70.3	2.0		0.383771494
C		64.9	2.6		0.217082562
O		68.5	2.6		0.132516482
K		91.9	2.6		0.204709509
P		93.7	2.9		0.496873792
B		72.1	3.9		0.387614064
G		99.1	4.1		0.552492172
E		121	4.5		0.475139465
N		146	4.5		0.813710339
D		155	3.0		0.534815242
L		155	4.6		0.862259153
A		162.2	3.5		0.573189504
I		169	3.4		0.799263517
H		180.2	5.0		0.752011091
F		198	2.2		0.969279952
Q		200	4.6		0.745266898
M		218	3.8		0.99

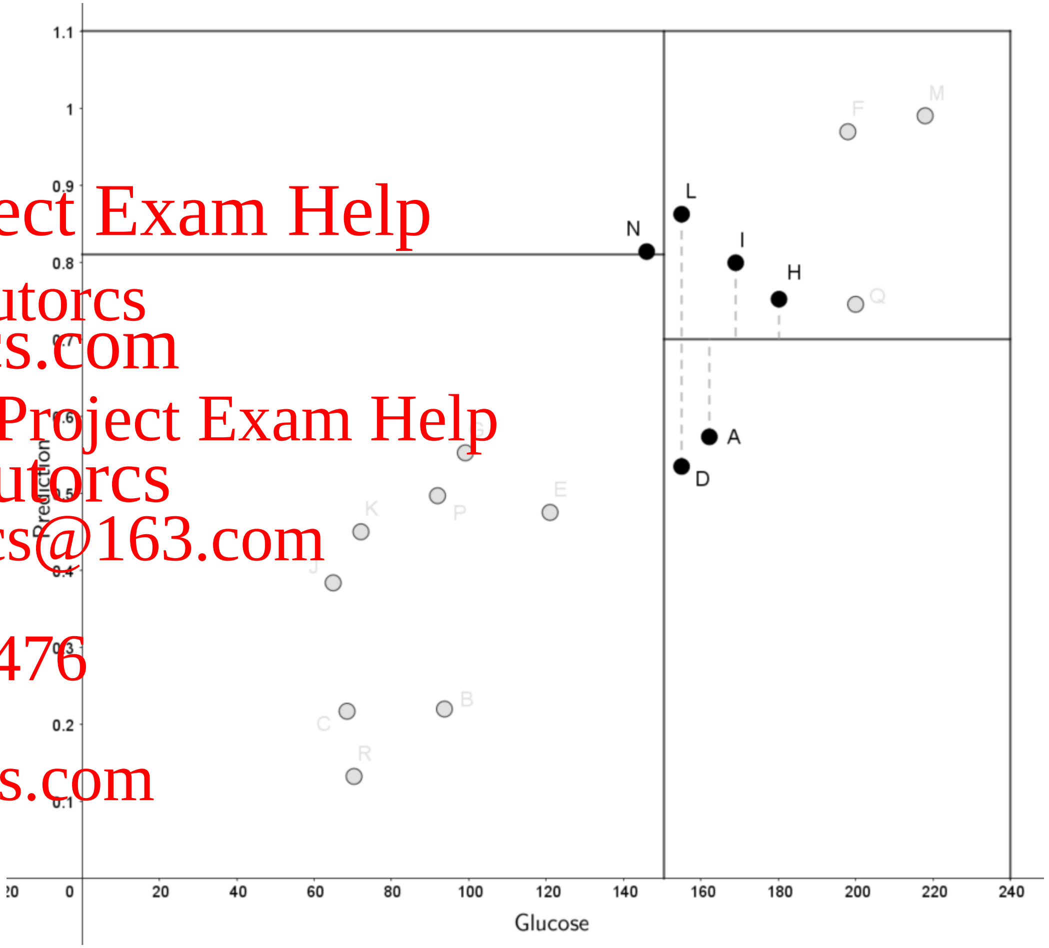
Left Sub-Tree

Right Sub-Tree

4.4 Level 2: Finalizing the Regression Tree Example (Glucose)

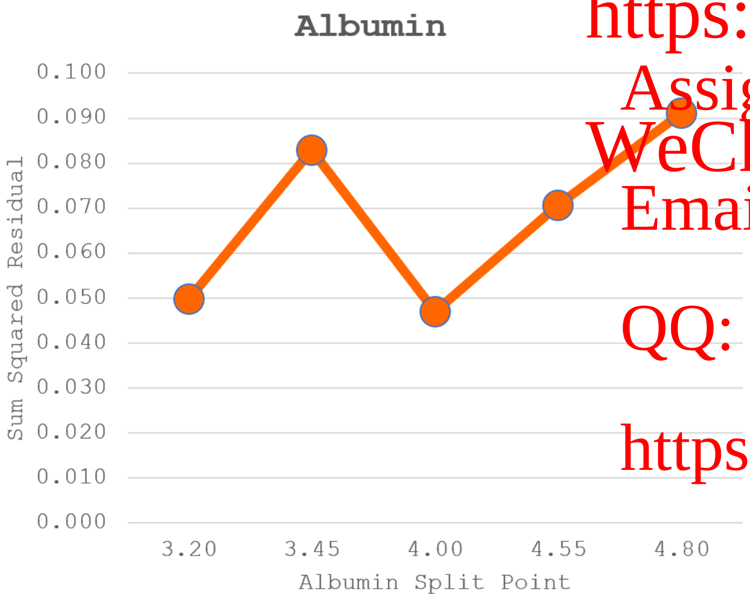
- The lowest SSR (**0.08**
- Between Patients N and
  →150.45

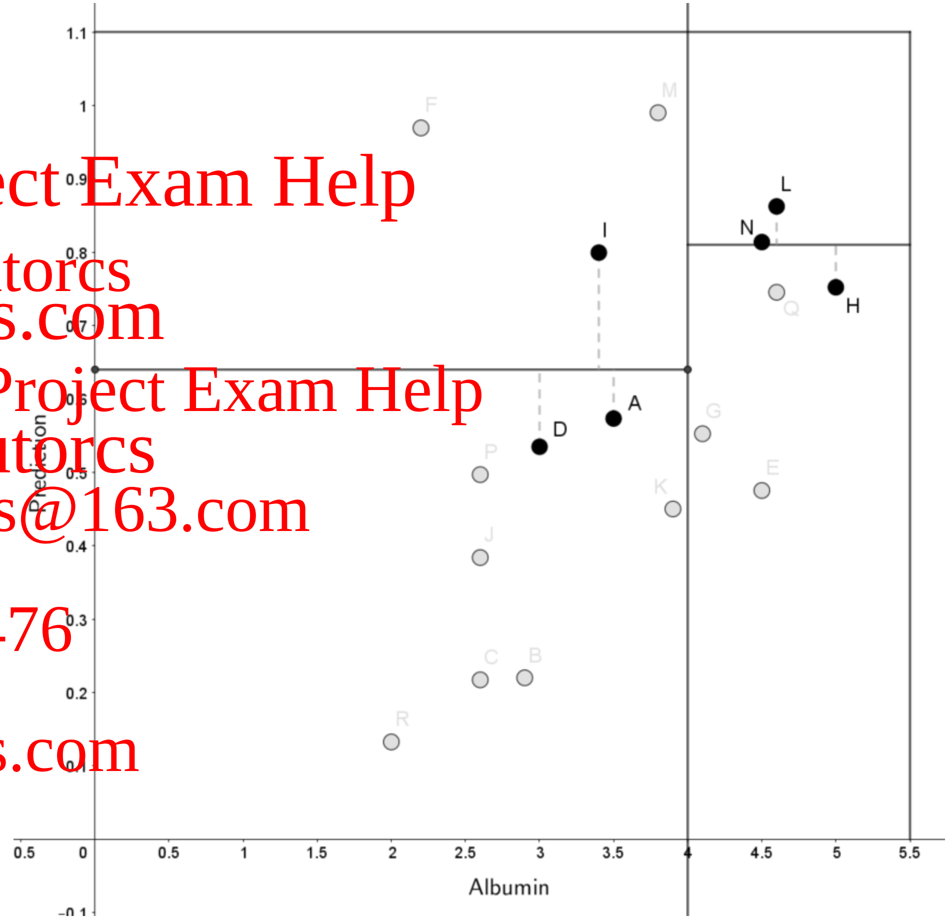




4.4 Level 2: Finalizing the Regression Tree Example (Albumin)

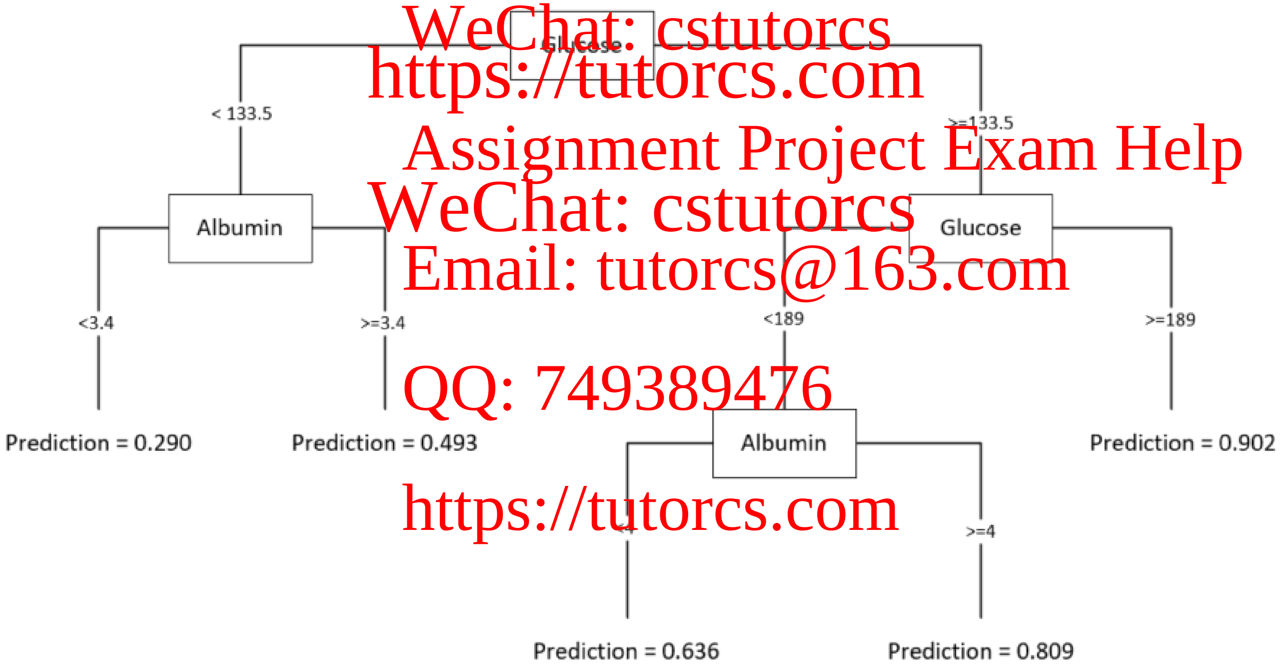
- The lowest SSR (**0.04**
- Between Patients A and
  →4.0



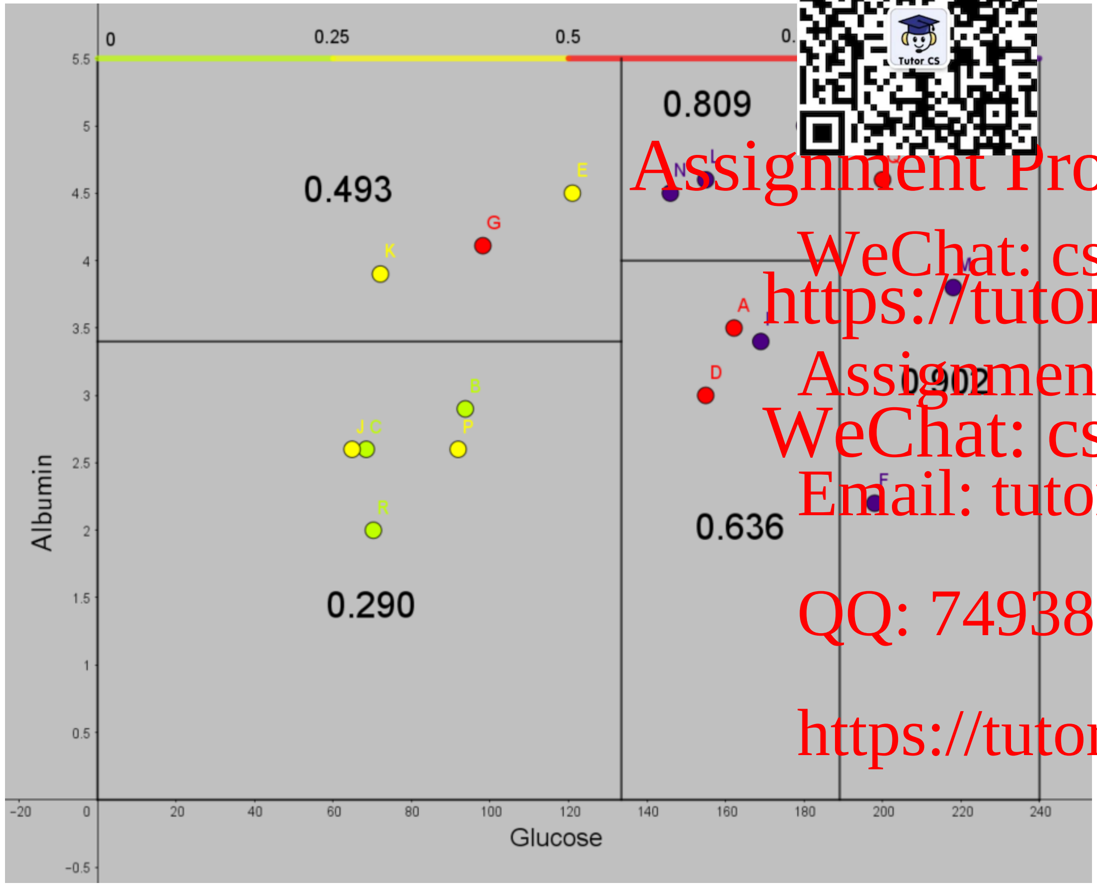


4.4 Level 2: Finalizing the Regression Tree Example (Regression Tree)

- Min(SSR Glucose) (0. Min (SSR Albumin) (**0.047**)
- The Split Node = Albumin.



4.4 Level 2: Finalizing the Regression Tree Example (Final Regression Tree)



- Mortality prediction value is represented with colour (low-high)
- The root → vertical line (Glucose = 133.5)

Regression Tree: Summary

- A regression tree can be seen as a grid partitioning method.
- A regression tree is built using training dataset.
- Testing dataset is matched against the tree, and compare the prediction value that the testing data has and the predicted value given by the regression tree.

Regression Tree vs Decision Tree

Regression Tree

- Binary tree
- Works with numerical attributes and target class
- An attribute can be reused in lower level of sub-tree

Decision Tree

- N-Ary tree
- Works with categorical attributes and target class
- An attribute cannot be used in sub-tree

Summary

- Data analytics for data warehousing focuses on data in the star schema.
- The focus on data analytics in data warehousing is primarily on fact measures which are numerical values.
- Three data analytics techniques suitable for data warehousing:
  i. Regression
  ii. Clustering
    a. Centroid-based
    b. Density-based
  iii. Classification using Regression Tree