

**School of Computer, Data and Mathematical Sciences**

**COMP 7006 Data Science**

**Computer Based Assignment – PART B**

**Spring, 2024**

|  |  |
| --- | --- |
| Complete your details in this section. | |
| STUDENT ID: | 22146011 |
| STUDENT FIRST NAME: | Shushil |
| STUDENT SURNAME: | Rana |

|  |  |
| --- | --- |
| QUESTIONS FORMAT: | Word processed document in PDF format; logically presenting answers to each question incorporating R outputs including graphs and charts. |
| TOTAL MARKS: | **60 Marks** |
| UNIT CO-ORDINATOR: | Dr. Liwan Liyanage |
| TUTOR: | Ms. Prathayne Nanthakumaran |
| TOTAL PAGES: | 32 |

|  |
| --- |
| **INSTRUCTIONS**  Please note that you are expected to answer the questions clearly in this document. Use the template included where relevant to answer. Give the R outputs, comments, and discussion clearly and logically in the rectangular box provided under each question. Attach all the R commands in the Appendix. Write the resulting **model equation** to the relevant questions. Once completed submit the answer scripts as a **PDF** via TurnItin link within vUWS site.  Please note that **10 Marks** are allocated for organization, reasoning, logical flow, and the inclusion of all correct R codes and outputs in the Appendix for both Part A and Part B. |

|  |
| --- |
| **SCENARIO** |
| Recent public health data indicate a troubling increase in kidney disease rates within specific suburban areas, attracting significant attention from public health practitioners. Determined to uncover the root causes and identify actionable risk factors to address this issue, the public health team has embarked on a comprehensive study. They have collected patient records and relevant information on medical factors and water quality, as provided in the dataset. |

**Data Description:**

|  |  |
| --- | --- |
| **Variable** | **Description** |
| PatientID | Unique identifier of each patient |
| Age | Age of the individual |
| Gender | Gender of the individual |
| BloodPressure | Systolic blood pressure in mmHg |
| BloodSugar | Fasting blood sugar levels in mg/dL |
| Cholesterol | Total cholesterol level in mg/dL |
| BodyMassIndex | BMI, a measure of body fat based on height and weight |
| SmokingStatus | Smoking status of the individual [Never/ Former/ Current] |
| ElectricConductivity | Measurement of the water’s ability to conduct electricity, which can indicate contamination in μS/cm |
| pH | pH level of the water |
| DissolvedOxygen | Amount of oxygen dissolved in water in mg/L |
| Turbidity | Measure of water clarity in NTU |
| TotalDissolvedSolids | Measure of dissolved substances in water in mg/L |
| NitriteLevel | Nitrite concentration in water in mg/L |
| NitrateLevel | Nitrate concentration in water in mg/L |
| LeadConcentration | Lead concentration in water in mg/L |
| ArsenicConcentration | Arsenic concentration in water in mg/L |
| Humidity | Ambient humidity level in % |
| KidneyDisease | Presence or absence of kidney disease |

\* Please note that this is a simulated data generated to resemble the real-world data for the purpose of this assignment.

Consider the scenario described, the data set provided and your answers in Part A to answer the following questions.

1. Build a logistic regression model incorporating polynomial terms. Clearly outline and explain each step of the process involved. [*This question is designed to assess your critical thinking and analytical skills. Please note that guidance on how to complete the task will not be provided.*] **(8 Marks)**

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| model\_poly <- glm(KidneyDisease ~ poly(Turbidity, 2)+poly(TotalDissolvedSolids,2) + poly(TotalDissolvedSolids,2), data = KidneyData, family = binomial)  Call:  glm(formula = KidneyDisease ~ poly(Turbidity, 2) + poly(ElectricConductivity,  2) + poly(TotalDissolvedSolids, 2), family = binomial, data = KidneyData)  Coefficients:  Estimate Std. Error z value Pr(>|z|)  (Intercept) 3.9645 0.5196 7.630 2.34e-14 \*\*\*  poly(Turbidity, 2)1 14.9679 3.4241 4.371 1.24e-05 \*\*\*  poly(Turbidity, 2)2 -2.0282 3.2244 -0.629 0.52934  poly(ElectricConductivity, 2)1 -79.3535 14.0366 -5.653 1.57e-08 \*\*\*  poly(ElectricConductivity, 2)2 32.8256 8.2078 3.999 6.35e-05 \*\*\*  poly(TotalDissolvedSolids, 2)1 -26.9294 8.7578 -3.075 0.00211 \*\*  poly(TotalDissolvedSolids, 2)2 5.3941 6.4867 0.832 0.40566  ---  Signif. codes: 0 ‘\*\*\*’ 0.001 ‘\*\*’ 0.01 ‘\*’ 0.05 ‘.’ 0.1 ‘ ’ 1  (Dispersion parameter for binomial family taken to be 1)  Null deviance: 519.21 on 499 degrees of freedom  Residual deviance: 282.97 on 493 degrees of freedom  AIC: 296.97  Number of Fisher Scoring iterations: 8   * Results of all attributes looked upon and only the most significant one has been used i.e Turbidity, ElectricConductivity and TotalDissolvedSolids * KidneyDisease ~ poly(Turbidity, 2) + poly(TotalDissolvedSolids, 2)+ + poly(ElectricConductivity, 2) is the model formula * KidneyDisease in the dependent variable i.e the outcome that we’re trying to predict * Turbidity, ElectricConductivity and TotalDissolvedSolids are the independent variables where poly(attribute,2) indicates that polynomial function of degree 2 has been implemented which ensures the model captures non-linear relationship * Provided model has lowest AIC as compared to other with 296.97 that suggests how better the model fits the data * Four terms of significant predictors can be identified indicating how more it captures complexity between outcome and predictors  |  |  | | --- | --- | | Intercept | P<0.001i.e highly significant | | poly(Turbidity, 2)1 | (p=1.24e-05) highly  significant positive  relationship | | poly(ElectricConductivity, 2)1 | (p = 1.57e-08) strong negative  relationship with kidney disease | | poly(ElectricConductivity, 2)2 | (p = 6.35e-05) significant | | **poly(TotalDissolvedSolids, 2)1** | (p = 0.00211) significant | |

1. Give the resultant accepted model (i.e. write the model equation) based on your findings above. Justify your answer clearly. **(3 Marks)**

|  |
| --- |
| The final model:  KidneyDisease ~ poly(Turbidity, 2) + poly(TotalDissolvedSolids, 2) ) + poly(ElectricConductivity, 2)  Model Equation:  logit(p)=β0​+β1​⋅poly(Turbidity, 2)+β2​⋅poly(ElectricConductivity, 2)+β3​⋅poly(TotalDissolvedSolids, 2)  When compared to other models has lowest residual deviance 282.97 and AIC 296.97 |

1. Use decision tree model to answer the research question. Clearly outline and explain each step of the process involved [*Hint: model building, improvement and evaluation*]. **(12 Marks)**

|  |
| --- |
| > # checking for the the missing values  > colSums(is.na(KidneyData))  PatientID Age Gender BloodPressure BloodSugar  0 0 0 0 0  Cholesterol BMI SmokingStatus ElectricConductivity pH  0 0 0 0 0  DissolvedOxygen Turbidity TotalDissolvedSolids NitriteLevel NitrateLevel  0 0 0 0 0  LeadConcentration ArsenicConcentration Humidity KidneyDisease  0 0 0 0  # Removing PatientID from the dataset because it doesn’t contribute in predicting target variable  KidneyData <- KidneyData[, !(names(KidneyData) %in% c("PatientID"))]  set.seed(123)  # Dataset split into training and testing (80/20)  trainIndex <- createDataPartition(KidneyData$KidneyDisease, p = .8,  list = FALSE,  times = 1)  KidneyTrain <- KidneyData[trainIndex, ]  KidneyTest <- KidneyData[-trainIndex, ]  # Decision tree model building  tree\_model <- rpart(KidneyDisease ~ ., data = KidneyTrain, method = "class")  summary(tree\_model)  # Predictions  y\_pred <- predict(tree\_model, newdata = KidneyTest, type = "class")  # Creating confusion matrix  confusion\_result <- confusionMatrix(factor(y\_pred, levels = levels(KidneyTest$KidneyDisease)), KidneyTest$KidneyDisease)  > # Printing Confusion matrix Results that shows performance of the  Model  > print(confusion\_result)  Confusion Matrix and Statistics  Reference  Prediction 0 1  0 14 3  1 7 75    Accuracy : 0.899  95% CI : (0.8221, 0.9505)  No Information Rate : 0.7879  P-Value [Acc > NIR] : 0.002856    Kappa : 0.6752    Mcnemar's Test P-Value : 0.342782    Sensitivity : 0.6667  Specificity : 0.9615 //96.15% negative cases identified  Pos Pred Value : 0.8235 //positive case prediction rate 82.35  Neg Pred Value : 0.9146  Prevalence : 0.2121  Detection Rate : 0.1414 //actual positive cases detected  Detection Prevalence : 0.1717  Balanced Accuracy : 0.8141    'Positive' Class : 0  decision\_tree\_model <- rpart(KidneyDisease ~ ., data = KidneyTrain, method = "class")  rpart.plot(decision\_tree\_model,  type = 4,  extra = 101,  under = TRUE,  fallen.leaves = TRUE,  digits = 2)  True Negatives=14, False Positives=3, False Negatives=7,  True Positives=75 and Accuracy = 89.9% |

1. Give the resultant model and interpret it. Clearly describe the terminal nodes [i.e. list the profiles]. *[Include the relevant R output]* **(5 Marks)**

|  |
| --- |
| confusion\_result <- confusionMatrix(factor(y\_pred, levels = levels(KidneyTest$KidneyDisease)), KidneyTest$KidneyDisease)  > # Printing the confusion matrix result  > print(confusion\_result)  Confusion Matrix and Statistics  Reference  Prediction 0 1  0 14 3   1. 7 75   Accuracy: 0.899  95% CI: (0.8221, 0.9505)  No Information Rate: 0.7879  P-Value [Acc > NIR]: 0.002856    Kappa: 0.6752    Mcnemar's Test P-Value: 0.342782    Sensitivity: 0.6667  Specificity: 0.9615  Pos Pred Value: 0.8235  Neg Pred Value: 0.9146  Prevalence: 0.2121  Detection Rate: 0.1414  Detection Prevalence: 0.1717  Balanced Accuracy: 0.8141    'Positive' Class: 0     * Terminal nodes starting from left the first one indicates that out of 55 observations 51 are classified as disease and 4 are classified as no * The second one indicates 3 as disease and 4 as no disease * The third one indicates 15 as no disease and 0 as disease * On the fouth one 0 as no disease and 37 as disease * Fifth one indicates 10 as no disease and 7 as disease * For the sixth one 0 as no disease and and 11 as disease * Seventh node indicates 2 no disease and 30 disease * Last one indicates 5 no disease and 222 disease * The prediction on class one (kidney disease) is more on the model excluding first, third and sixth node * For splitting the data ElectricConductivity in the most determinant factor * More prevalence of kidney disease can be identified in the dataset identified through observation |

1. Compare the different resultant models (Part A Question 5, Part B Question 2 and Question 4) you obtained above. **(12 Marks)**

|  |
| --- |
| confusion\_result <- confusionMatrix(factor(y\_pred, levels = levels(KidneyTest$KidneyDisease)), KidneyTest$KidneyDisease)  > # Printing the confusion matrix result  > print(confusion\_result)  Confusion Matrix and Statistics  Reference  Prediction 0 1  0 14 3   * 7 75   Accuracy: 0.899  95% CI: (0.8221, 0.9505)  No Information Rate: 0.7879  P-Value [Acc > NIR]: 0.002856    Kappa: 0.6752    Mcnemar's Test P-Value: 0.342782    Sensitivity: 0.6667  Specificity: 0.9615  Pos Pred Value: 0.8235  Neg Pred Value: 0.9146  Prevalence: 0.2121  Detection Rate: 0.1414  Detection Prevalence: 0.1717  Balanced Accuracy: 0.8141    'Positive' Class: 0  **Accuracy** of 89.9% suggests the model is quite good overall.  **High sensitivity** (96.15%) means the model is very effective at identifying patients with kidney disease.  **Specificity** is lower (66.67%), meaning the model is less effective at identifying patients without kidney disease, leading to some false positives.  Actual  Predicted 0 1  0 4 15  1 16 65  Accuracy = 69%, Sensitivity (1 recall) = 81.25,  Specificity 0 recall) = 20%,  Precision (1) =80.25   The model is performing well in identifying positive cases (class 1), with high sensitivity and precision for detecting kidney disease.   However, it struggles to correctly identify negative cases (class 0), as seen from the low specificity.   Overall, the model may be biased toward predicting positive cases, making it more useful in scenarios where detecting kidney disease is critical, even at the cost of more false positives.  Call:  glm(formula = KidneyDisease ~ poly(Turbidity, 2) + poly(ElectricConductivity,  2) + poly(TotalDissolvedSolids, 2), family = binomial, data = KidneyData)  Coefficients:  Estimate Std. Error z value Pr(>|z|)  (Intercept) 3.9645 0.5196 7.630 2.34e-14 \*\*\*  poly(Turbidity, 2)1 14.9679 3.4241 4.371 1.24e-05 \*\*\*  poly(Turbidity, 2)2 -2.0282 3.2244 -0.629 0.52934  poly(ElectricConductivity,2)1 -79.3535 14.0366 -5.653 1.57e-08 \*\*\*  poly(ElectricConductivity,2)2 32.8256 8.2078 3.999 6.35e-05 \*\*\*  poly(TotalDissolvedSolids,2)1 -26.9294 8.7578 -3.075 0.00211 \*\*  poly(TotalDissolvedSolids,2)2 5.3941 6.4867 0.832 0.40566  ---  Signif. codes: 0 ‘\*\*\*’ 0.001 ‘\*\*’ 0.01 ‘\*’ 0.05 ‘.’ 0.1 ‘ ’ 1  (Dispersion parameter for binomial family taken to be 1)  Null deviance: 519.21 on 499 degrees of freedom  Residual deviance: 282.97 on 493 degrees of freedom  AIC: 296.97  Number of Fisher Scoring iterations: 8  **Residual deviance**: 282.97 on 493 degrees of freedom   * Lower residual deviance suggests a better fit compared to the null deviance (519.21).   **AIC**: 296.97   * A lower AIC indicates a better model fit, and this model has a lower AIC compared to previous models, indicating it's likely better. * **Intercept**:   Estimate: 3.9645  This is the baseline log-odds of having kidney disease when all predictors (polynomial terms) are at their reference levels. A positive value indicates higher baseline odds of kidney disease.   * **Poly (Turbidity, 2)1**:   **p-value: 1.24e-05**  This term is highly significant (**p < 0.001**) and suggests a strong positive association between the first polynomial term of Turbidity and the probability of kidney disease. The higher the Turbidity (up to a certain degree), the higher the likelihood of kidney disease.   * **Poly (Turbidity, 2)2**:   p-value: 0.52934 (not significant)  This term is not statistically significant, implying that higher-degree polynomial effects of Turbidity might not have a strong independent impact on kidney disease prediction.   * **Poly (Electric Conductivity, 2)1**:   **p-value: 1.57e-08**  This term is highly significant (**p < 0.001**) and negatively associated with kidney disease. It suggests that higher levels of Electric Conductivity strongly decrease the odds of having kidney disease.   * **Poly (Electric Conductivity, 2)2**:   **p-value: 6.35e-05**  This second-degree term is also highly significant and positively associated with kidney disease. This suggests a complex non-linear relationship between Electric Conductivity and kidney disease (with both positive and negative effects).   * **Poly (Total Dissolved Solids, 2)1**:   **p-value: 0.00211**  This term is significant (**p < 0.01**) and negatively associated with kidney disease. Higher values of Total Dissolved Solids decrease the odds of having kidney disease.   * **Poly (Total Dissolved Solids, 2)2**:   p-value: 0.40566 (not significant)  This term is not statistically significant, suggesting that the higher-order effect of Total Dissolved Solids does not contribute significantly to the prediction.  This model fits the data well based on the significant predictors and lower AIC score compared to earlier models. Key predictors like **Turbidity**, **Electric Conductivity**, and **Total Dissolved Solids** (particularly their first-degree terms) have a significant impact on kidney disease prediction, and the non-linear relationships are captured by the polynomial terms. |
|  |

1. Give the final accepted model based on your findings above and Part A. Justify your answer. **(5 Marks)**

|  |
| --- |
| Call:  glm(formula = KidneyDisease ~ poly(Turbidity, 2) + poly(ElectricConductivity,  2) + poly(TotalDissolvedSolids, 2), family = binomial, data = KidneyData)  Coefficients:  Estimate Std. Error z value Pr(>|z|)  (Intercept) 3.9645 0.5196 7.630 2.34e-14 \*\*\*  poly(Turbidity, 2)1 14.9679 3.4241 4.371 1.24e-05 \*\*\*  poly(Turbidity, 2)2 -2.0282 3.2244 -0.629 0.52934  poly(ElectricConductivity, 2)1 -79.3535 14.0366 -5.653 1.57e-08 \*\*\*  poly(ElectricConductivity, 2)2 32.8256 8.2078 3.999 6.35e-05 \*\*\*  poly(TotalDissolvedSolids, 2)1 -26.9294 8.7578 -3.075 0.00211 \*\*  poly(TotalDissolvedSolids, 2)2 5.3941 6.4867 0.832 0.40566  Signif. codes: 0 ‘\*\*\*’ 0.001 ‘\*\*’ 0.01 ‘\*’ 0.05 ‘.’ 0.1 ‘ ’ 1  (Dispersion parameter for binomial family taken to be 1)  Null deviance: 519.21 on 499 degrees of freedom  Residual deviance: 282.97 on 493 degrees of freedom  AIC: 296.97  Number of Fisher Scoring iterations: 8   **Accuracy**: 69% indicates the model's overall reliability.   **Sensitivity**: 81.25% shows it is effective at identifying kidney disease.   **fit and significance**: The model has significant predictors like **Turbidity**, **ElectricConductivity**, and **TotalDissolvedSolids**, as reflected in the p-values.   **Lower AIC**: The model has a low AIC (Akaike Information Criterion) of **296.97**, which suggests a better fit compared to other models.   **Residual Deviance**: The residual deviance is **282.97**, indicating how well the model fits the data after accounting for the explanatory variables.   **Accuracy and balanced performance**: While there were some issues with specificity in the confusion matrix the model's overall performance justifies its acceptance, particularly if the priority is to correctly detect positive kidney disease cases. |

1. Apply an unsupervised learning technique of your choice to identify any interesting or hidden patterns in the dataset. Provide a clear explanation of the technique used and thoroughly describe your findings. **(10 Marks)**

|  |
| --- |
| applied K-means with 4 clusters, using 25 random initializations to find the best configuration.  set.seed(123)  kmeans\_result <- kmeans(KidneyData\_scaled, centers = 4, nstart = 25)  print(kmeans\_result)  kmeans\_result$cluster  kmeans\_result$centers  library(ggplot2)  KidneyData\_clustered <- as.data.frame(KidneyData\_scaled)  KidneyData\_clustered$cluster <- as.factor(kmeans\_result$cluster)  ggplot(KidneyData\_clustered, aes(x = KidneyData\_clustered[,1], y = KidneyData\_clustered[,4], color = cluster)) +  geom\_point() +  labs(title = "K-means Clustering Results", x = "TotalDissolvedSolids", y = "Turbidity")  print(kmeans\_result$centers)  print(kmeans\_result$size)   Cluster 1: 129 members   Cluster 2: 144 members   Cluster 3: 128 members   Cluster 4: 99 members  Cluster 2 is the largest, suggesting that most patients exhibit the characteristics that define this group. This could be further analyzed to determine if these characteristics are associated with a higher or lower risk of kidney disease.  Cluster 1: This cluster seems to include individuals with relatively higher **ages**, **blood pressures**, and a tendency towards higher **Cholesterol**, along with notable characteristics of low **BMI** and **Electric Conductivity**. This cluster may represent a group of individuals that could be at higher risk for health issues.  Cluster 2: This group appears to be younger, with lower blood pressure and a more humid environment. They also tend to have moderate **Cholesterol** levels and are characterized by having a relatively lower **Electric Conductivity**.  Cluster 3: This cluster has individuals with average readings across most parameters, but notably high **Total Dissolved Solids** and **Electric Conductivity**, suggesting a particular interest in the water quality, potentially indicating contamination.  Cluster 4: This cluster represents older individuals with higher blood pressure and more significant **Turbidity** and **Total Dissolved Solids**, indicating possible environmental health risks associated with aging. |

1. What are your conclusion and recommendations for this problem? [Hint: Use your results and findings from previous questions to answer this question] **(5 Marks)**

|  |
| --- |
|  **Model 1**: (Polynomial of Electric Conductivity and Total Dissolved Solids) showed a higher AIC (317.83) but had limitations in predictive performance.   **Model 2**: (Polynomial of Turbidity and Total Dissolved Solids) had a poorer fit with an AIC of 355.18 and lower predictive accuracy.   **Final Model**: (Polynomial of Turbidity, Electric Conductivity, and Total Dissolved Solids) performed the best with an AIC of 296.97 and an accuracy of 69%. It demonstrated good sensitivity (81.25%), indicating a solid ability to detect kidney disease but struggled with specificity (20%).  The confusion matrices indicated that the models had differentiationg rate on true positives and true negatives. While the final model correctly identified many positive cases, it had a higher false positive rate, meaning it had a hard time identifying some healthy cases as positive for kidney disease.  High sensitivity is crucial in medical diagnostics to avoid missing cases of kidney disease, which is in alignment with the model purpose.  Given its better fit and balance of sensitivity, the polynomial model incorporating Turbidity, Electric Conductivity, and Total Dissolved Solids should be used for predictions and analysis.  Working with the medical professionals for better results and overall interpretation and insights can be done to make the model work more efficiently.  High sensitivity on the accepted model can be benefitted for screening but a bit of work can be done on specificity.  To maintain reliability regular updates and maintenance will be the key. |

--- End of questions ---

APPENDIX

[Attach all your R codes and outputs here.]

KidneyData <- read.csv("KidneyData.csv")

setwd("C:/Users/Asus/Downloads")

attach(KidneyData)

str(KidneyData)

summary(KidneyData)

KidneyData$SmokingStatus <- as.factor(KidneyData$SmokingStatus)

model <- glm(KidneyDisease ~ Age+Gender+BloodPressure+BloodSugar+

Cholesterol+BMI+SmokingStatus+ElectricConductivity+pH+

DissolvedOxygen+Turbidity+TotalDissolvedSolids+NitriteLevel+

NitrateLevel+LeadConcentration+ArsenicConcentration+Humidity,

data = KidneyData, family = binomial)

model\_poly <- glm(KidneyDisease ~ poly(Turbidity, 2)+

poly(ElectricConductivity,2)+poly(TotalDissolvedSolids,2),

data = KidneyData, family = binomial)

summary(model)

summary(model\_poly)

+

poly(pH,2)+poly(DissolvedOxygen,2)+ poly(Turbidity,2)+

poly(TotalDissolvedSolids,2)

install.packages("rpart")

install.packages("rpart.plot")

install.packages("caret")

library(rpart)

library(rpart.plot)

library(caret)

head(KidneyData)

colSums(is.na(KidneyData))

KidneyData <- KidneyData[, !(names(KidneyData) %in% c("PatientID"))]

trainIndex <- createDataPartition(KidneyData$KidneyDisease, p = .8,

list = FALSE,

times = 1)

KidneyTrain <- KidneyData[trainIndex, ]

KidneyTest <- KidneyData[-trainIndex, ]

tree\_model <- rpart(KidneyDisease ~ ., data = KidneyTrain, method = "class")

summary(tree\_model)

set.seed(123)

=trainIndex <- createDataPartition(KidneyData$KidneyDisease, p = .8,

list = FALSE,

times = 1)

KidneyTrain <- KidneyData[trainIndex, ]

KidneyTest <- KidneyData[-trainIndex, ]

y\_pred <- predict(tree\_model, newdata = KidneyTest, type = "class")

confusion\_result <- confusionMatrix(factor(y\_pred, levels = levels(KidneyTest$KidneyDisease)),

KidneyTest$KidneyDisease)

print(confusion\_result)

decision\_tree\_model <- rpart(KidneyDisease ~ ., data = KidneyTrain, method = "class")

rpart.plot(decision\_tree\_model,

type = 4,

extra = 101,

under = TRUE,

fallen.leaves = TRUE,

digits = 2)

rpart.plot(tree\_model)

numeric\_columns <- sapply(KidneyData, is.numeric)

KidneyData\_numeric <- KidneyData[, numeric\_columns]

KidneyData\_scaled <- scale(KidneyData\_numeric)

set.seed(123)

kmeans\_result <- kmeans(KidneyData\_scaled, centers = 4, nstart = 25)

print(kmeans\_result)

kmeans\_result$cluster

kmeans\_result$centers

library(ggplot2)

KidneyData\_clustered <- as.data.frame(KidneyData\_scaled)

KidneyData\_clustered$cluster <- as.factor(kmeans\_result$cluster)

ggplot(KidneyData\_clustered, aes(x = KidneyData\_clustered[,1], y = KidneyData\_clustered[,4], color = cluster)) +

geom\_point() +

labs(title = "K-means Clustering Results", x = "X", y = "Y")

print(kmeans\_result$centers)

print(kmeans\_result$size)

colnames(KidneyData)

KidneyData\_scaled <- scale(KidneyData)

print(KidneyData.head())

str(KidneyData)

|  |
| --- |
| > colSums(is.na(KidneyData))  PatientID Age Gender BloodPressure BloodSugar  0 0 0 0 0  Cholesterol BMI SmokingStatus ElectricConductivity pH  0 0 0 0 0  DissolvedOxygen Turbidity TotalDissolvedSolids NitriteLevel NitrateLevel  0 0 0 0 0  LeadConcentration ArsenicConcentration Humidity KidneyDisease  0 0 0 0  > # Set the seed for reproducibility  > set.seed(42)  > # Split the dataset into training and testing sets  > trainIndex <- createDataPartition(KidneyData$KidneyDisease, p = .7,  + list = FALSE,  + times = 1)  > KidneyTrain <- KidneyData[trainIndex, ]  > KidneyTest <- KidneyData[-trainIndex, ]  > # Build the Decision Tree model  > tree\_model <- rpart(KidneyDisease ~ ., data = KidneyTrain, method = "class")  > # Plot the tree  > rpart.plot(tree\_model)  > View(tree\_model)  > # Tune the complexity parameter  > control <- trainControl(method = "cv", number = 10)  > # Grid of cp values  > grid <- expand.grid(cp = seq(0.01, 0.1, by = 0.01))  > # Train the model using cross-validation  > tuned\_model <- train(KidneyDisease ~ ., data = KidneyTrain,  + method = "rpart",  + trControl = control,  + tuneGrid = grid)  Warning message:  In train.default(x, y, weights = w, ...) :  You are trying to do regression and your outcome only has two possible values Are you trying to do classification? If so, use a 2 level factor as your outcome column.  > # Best cp value  > print(tuned\_model$bestTune)  cp  10 0.1  > y\_pred <- predict(tuned\_model, newdata = KidneyTest)  Error in model.frame.default(Terms, newdata, na.action = na.action, xlev = object$xlevels) :  factor PatientID has new levels ABC4734, AFI1502, AWC2280, AXL8229, AYI4551, BEP5552, BGB2014, BIU4942, BLD1645, BMI5300, BUQ7112, CBT7965, CTB7560, DAD5631, DEJ3681, DPR7300, EBG5216, EDW6186, EJJ8170, EMD7289, ENP2462, EQO4419, EWH4807, FAW2002, FEA5971, FKX2644, FLQ1417, FME1378, FSC8241, FTY5188, FZR8116, GCX2069, GMH2585, GMT7313, GVX2978, GXU7790, GYM8874, HHJ5626, HJI4546, HLS8164, HZP5936, IEF4164, IEL5208, IFA4522, IOG7669, JAP4615, JAQ2348, JJI5959, JPC8910, JPQ1361, JSG6328, JTB3961, JTI2535, JUR3635, KAW5336, KLO4605, KPD5355, KQB8808, KTF1859, LDS1743, LEO7812, LHV1359, LJT6421, LOO4758, LTG1227, MOL3902, MOT1855, NHN1002, NLA3004, NLV1635, NNJ6179, NNN1931, NPV7863, NQA1345, NSE3535, NTK3362, NYI2098, OAB4243, OAH4339, OOB8800, OPT1801, OQS3338, OQZ8526, OWQ4420, OXM4720, PCS2264, PTI8846, QNQ6338, QWD5419, RCT7570, RJP5342, RMJ1934, ROK8850, RQI7001, RWS6415, SNL1900, SRU1775, STO3441, SWM2049, SYP8379, TJE1313, TKH2546, TLQ7041, TTX6354, TVA1204, TWK6849, TXD29  > confusionMatrix(y\_pred, KidneyTest$KidneyDisease)  > confusionMatrix(TotalDissolvedSolids, KidneyTest$KidneyDisease)  > confusionMatrix(Turbidity, KidneyTest$KidneyDisease)  > KidneyData$KidneyDisease <- as.factor(KidneyData$KidneyDisease)  > levels(KidneyTest$KidneyDisease)  NULL  > # Create confusion matrix  > confusionMatrix(factor(y\_pred, levels = levels(KidneyTest$KidneyDisease)),  + KidneyTest$KidneyDisease)  > View(grid)  > View(KidneyTrain)  > View(KidneyTest)  > View(model)  > View(tree\_model)  > View(tuned\_model)  > View(control)  > View(control)  > View(KidneyData)  > View(KidneyTest)  > str(KidneyTest)  'data.frame': 150 obs. of 19 variables:  $ PatientID : chr "KAW5336" "NPV7863" "TJE1313" "JTI2535" ...  $ Age : int 76 45 38 73 33 44 57 69 57 30 ...  $ Gender : chr "Male" "Male" "Female" "Female" ...  $ BloodPressure : int 194 151 173 165 123 138 154 163 134 119 ...  $ BloodSugar : num 96.9 153.4 116.9 52.6 81.5 ...  $ Cholesterol : int 251 200 162 200 236 241 210 215 210 271 ...  $ BMI : num 26.2 22.2 18 22.4 19 27.5 20.4 16.6 30.3 24.7 ...  $ SmokingStatus : Factor w/ 3 levels "Current","Former",..: 1 1 1 2 1 2 3 3 1 1 ...  $ ElectricConductivity: num 293 235 285 317 231 ...  $ pH : num 7.01 7.46 7.34 7.96 6.31 7.43 6.11 6.56 6.41 7.06 ...  $ DissolvedOxygen : num 9.79 8.72 9.42 7.35 8.61 7.89 8.75 9.2 9.09 7.39 ...  $ Turbidity : num 1.16 0.98 0.91 0.9 0.8 1.08 0.76 1.06 1.21 0.72 ...  $ TotalDissolvedSolids: num 430 364 336 407 332 ...  $ NitriteLevel : num 0.177 0.044 0.024 0.09 0.005 0.126 0.09 0.134 0.089 0.094 ...  $ NitrateLevel : num 1.13 1.13 1.09 0.95 1.36 1.6 0.77 0.56 1.2 0.92 ...  $ LeadConcentration : num 0.012 0.0106 0.0003 0.0088 0.0095 0.0155 0.0137 0.0193 0.0143 0.0164 ...  $ ArsenicConcentration: num 0.0035 0.0062 0.0026 0.006 0.0026 0.0036 0.0051 0.0036 0.0033 0.0021 ...  $ Humidity : num 50.8 70.5 46.7 55.2 77 58.4 60.3 37.2 42.6 41.2 ...  $ KidneyDisease : int 1 1 1 0 1 0 1 1 0 1 ...  > str(KidneyTrain)  'data.frame': 350 obs. of 19 variables:  $ PatientID : chr "TIW5219" "QLJ3151" "GRL2542" "WMM4122" ...  $ Age : int 120 10 58 22 52 53 57 30 63 82 ...  $ Gender : chr "Female" "Female" "Female" "Female" ...  $ BloodPressure : int 118 143 300 20 150 141 140 141 151 168 ...  $ BloodSugar : num 156 162 121 154 159 ...  $ Cholesterol : int 165 214 222 212 600 199 215 205 193 213 ...  $ BMI : num 31.7 23.9 16.3 21.9 23.8 18.3 19.5 25.7 31.5 19.9 ...  $ SmokingStatus : Factor w/ 3 levels "Current","Former",..: 2 3 2 3 1 2 2 2 2 3 ...  $ ElectricConductivity: num 336 297 378 312 222 ...  $ pH : num 7.4 7.48 7.49 6.03 6.77 7.34 7.38 6.7 6.75 6.99 ...  $ DissolvedOxygen : num 9.57 8.49 8.18 7.35 7.4 8 8.04 6.98 6.86 8.15 ...  $ Turbidity : num 1.44 1.21 0.88 1.15 0.73 0.71 1.47 1.1 0.8 0.84 ...  $ TotalDissolvedSolids: num 455 423 434 400 349 ...  $ NitriteLevel : num 0.165 0.075 0.005 0.088 0.119 0.076 0.114 0.042 0.052 0.174 ...  $ NitrateLevel : num 1.97 1.74 1.4 0.88 0.71 1 1.13 0.82 0.88 0.8 ...  $ LeadConcentration : num 0.0099 0.012 0.0173 0.0133 0.0155 0.005 0.0128 0.0145 0.0148 0.0093 ...  $ ArsenicConcentration: num 0.0063 0.0062 0.0092 0.0086 0.0011 0.009 0.0081 0.0046 0.008 0.0026 ...  $ Humidity : num 48.7 65.3 93.2 67.4 43.3 57.6 55.6 72.9 74.1 55.3 ...  $ KidneyDisease : int 0 1 0 1 1 0 0 1 0 1 ...  > # Split the dataset into training and testing sets  > trainIndex <- createDataPartition(KidneyData$KidneyDisease, p = .8,  + list = FALSE,  + times = 1)  > KidneyTrain <- KidneyData[trainIndex, ]  > KidneyTest <- KidneyData[-trainIndex, ]  > str(KidneyTrain)  'data.frame': 401 obs. of 19 variables:  $ PatientID : chr "TIW5219" "QLJ3151" "GRL2542" "LPP8404" ...  $ Age : int 120 10 58 52 76 45 30 63 82 38 ...  $ Gender : chr "Female" "Female" "Female" "Female" ...  $ BloodPressure : int 118 143 300 150 194 151 141 151 168 173 ...  $ BloodSugar : num 155.8 162.5 120.8 158.9 96.9 ...  $ Cholesterol : int 165 214 222 600 251 200 205 193 213 162 ...  $ BMI : num 31.7 23.9 16.3 23.8 26.2 22.2 25.7 31.5 19.9 18 ...  $ SmokingStatus : Factor w/ 3 levels "Current","Former",..: 2 3 2 1 1 1 2 2 3 1 ...  $ ElectricConductivity: num 336 297 378 222 293 ...  $ pH : num 7.4 7.48 7.49 6.77 7.01 7.46 6.7 6.75 6.99 7.34 ...  $ DissolvedOxygen : num 9.57 8.49 8.18 7.4 9.79 8.72 6.98 6.86 8.15 9.42 ...  $ Turbidity : num 1.44 1.21 0.88 0.73 1.16 0.98 1.1 0.8 0.84 0.91 ...  $ TotalDissolvedSolids: num 455 423 434 349 430 ...  $ NitriteLevel : num 0.165 0.075 0.005 0.119 0.177 0.044 0.042 0.052 0.174 0.024 ...  $ NitrateLevel : num 1.97 1.74 1.4 0.71 1.13 1.13 0.82 0.88 0.8 1.09 ...  $ LeadConcentration : num 0.0099 0.012 0.0173 0.0155 0.012 0.0106 0.0145 0.0148 0.0093 0.0003 ...  $ ArsenicConcentration: num 0.0063 0.0062 0.0092 0.0011 0.0035 0.0062 0.0046 0.008 0.0026 0.0026 ...  $ Humidity : num 48.7 65.3 93.2 43.3 50.8 70.5 72.9 74.1 55.3 46.7 ...  $ KidneyDisease : Factor w/ 2 levels "0","1": 1 2 1 2 2 2 2 1 2 2 ...  > str(KidneyTest)  'data.frame': 99 obs. of 19 variables:  $ PatientID : chr "WMM4122" "CIH1298" "SXZ1151" "JTI2535" ...  $ Age : int 22 53 57 73 33 43 27 42 80 61 ...  $ Gender : chr "Female" "Female" "Male" "Female" ...  $ BloodPressure : int 20 141 140 165 123 122 140 153 149 155 ...  $ BloodSugar : num 154.2 131.6 72.8 52.6 81.5 ...  $ Cholesterol : int 212 199 215 200 236 167 179 161 173 234 ...  $ BMI : num 21.9 18.3 19.5 22.4 19 21.7 29.1 21.5 28.7 22.8 ...  $ SmokingStatus : Factor w/ 3 levels "Current","Former",..: 3 2 2 2 1 2 3 1 3 2 ...  $ ElectricConductivity: num 312 422 298 317 231 ...  $ pH : num 6.03 7.34 7.38 7.96 6.31 8.19 7.52 6.85 6.9 7.12 ...  $ DissolvedOxygen : num 7.35 8 8.04 7.35 8.61 7.03 8.06 8.22 7.82 7.58 ...  $ Turbidity : num 1.15 0.71 1.47 0.9 0.8 1.22 0.96 1.21 0.95 0.72 ...  $ TotalDissolvedSolids: num 400 427 416 407 332 ...  $ NitriteLevel : num 0.088 0.076 0.114 0.09 0.005 0.124 0.071 0.064 0.128 0.074 ...  $ NitrateLevel : num 0.88 1 1.13 0.95 1.36 1.57 1.66 1.58 1.47 0.99 ...  $ LeadConcentration : num 0.0133 0.005 0.0128 0.0088 0.0095 0.0045 0.0058 0.017 0.017 0.004 ...  $ ArsenicConcentration: num 0.0086 0.009 0.0081 0.006 0.0026 0.0035 0.0064 0.0068 0.0073 0.0036 ...  $ Humidity : num 67.4 57.6 55.6 55.2 77 63.5 50.3 64.8 70.3 58.2 ...  $ KidneyDisease : Factor w/ 2 levels "0","1": 2 1 1 1 2 2 2 2 2 2 ...  > summary(tree\_model)  Call:  rpart(formula = KidneyDisease ~ ., data = KidneyTrain, method = "class")  n= 350  CP nsplit rel error xerror xstd  1 1.00 0 1 1.0000000 0.10322970  2 0.01 1 0 0.7972973 0.09464698  Variable importance  PatientID ElectricConductivity TotalDissolvedSolids  66 20 12  BloodSugar pH Turbidity  1 1 1  Node number 1: 350 observations, complexity param=1  predicted class=1 expected loss=0.2114286 P(node) =1  class counts: 74 276  probabilities: 0.211 0.789  left son=2 (74 obs) right son=3 (276 obs)  Primary splits:  PatientID splits as RRRRRLLRRRRRRRLLRLRRRRRRRRRLLRRRLRRRLRRRRRRLRLRLRLLRRRRRRRRRRRRLLRRLRRRLRRRRRRRLRRRRLRRRRRLRRRRRRRLRLRRRRLRRRRRRRRLLRRRLRRRRRRLRRRRRRRLRRLRRRRRRLRRRRRRLRRRLRRRLRRRLRRRRRRLRRRRLRRRRRRRRRRRRRLRRLLLLLRRLRRRRRRRLRRRRRRLRLRRLLRRLRRRRRRRRLRRRLRLRRRRRRRLRLRRRRRRRRRRRRRRLRRRRLRLRRRLRRRRRLRRLRLRRRRRRLRRLRRRRRLRRRRRRRRRRRRLRRLRRLLLRRRRRRRRRRRRRRRLRRRRLLRRRRL, improve=116.708600, (0 missing)  ElectricConductivity < 329.6 to the right, improve= 43.733510, (0 missing)  TotalDissolvedSolids < 424.2 to the right, improve= 31.509160, (0 missing)  Turbidity < 1.005 to the left, improve= 7.335926, (0 missing)  BloodPressure < 136.5 to the left, improve= 4.107443, (0 missing)  Surrogate splits:  ElectricConductivity < 329.6 to the right, agree=0.851, adj=0.297, (0 split)  TotalDissolvedSolids < 435.75 to the right, agree=0.826, adj=0.176, (0 split)  BloodSugar < 301.6 to the right, agree=0.791, adj=0.014, (0 split)  pH < 5.93 to the left, agree=0.791, adj=0.014, (0 split)  Turbidity < 1.435 to the right, agree=0.791, adj=0.014, (0 split)  Node number 2: 74 observations  predicted class=0 expected loss=0 P(node) =0.2114286  class counts: 74 0  probabilities: 1.000 0.000  Node number 3: 276 observations  predicted class=1 expected loss=0 P(node) =0.7885714  class counts: 0 276  probabilities: 0.000 1.000  > rpart.plot(tree\_model)  > y\_pred <- predict(tree\_model, newdata = KidneyTest, type = "class")  Error in model.frame.default(Terms, newdata, na.action = na.action, xlev = attr(object, :  factor PatientID has new levels EQO4419, EWH4807, FLQ1417, GMT7313, HHJ5626, JAQ2348, JJI5959, JPC8910, JTI2535, LTG1227, NTK3362, OQZ8526, PCS2264, ROK8850, RWS6415, TLQ7041, TVA1204, UEV2364, UGH5427, UHF3341, UKX5149, UNC5398, UYL8769, VYO3290, VZP3248, WBC7434, WGL1429, WLE4998, YRG2600, YTR1015, ZHU3828  > # Make predictions  > y\_pred <- predict(tree\_model, newdata = KidneyTest, type = "class")  Error in model.frame.default(Terms, newdata, na.action = na.action, xlev = attr(object, :  factor PatientID has new levels EQO4419, EWH4807, FLQ1417, GMT7313, HHJ5626, JAQ2348, JJI5959, JPC8910, JTI2535, LTG1227, NTK3362, OQZ8526, PCS2264, ROK8850, RWS6415, TLQ7041, TVA1204, UEV2364, UGH5427, UHF3341, UKX5149, UNC5398, UYL8769, VYO3290, VZP3248, WBC7434, WGL1429, WLE4998, YRG2600, YTR1015, ZHU3828  > KidneyData <- KidneyData[, !(names(KidneyData) %in% c("PatientID"))]  > set.seed(123) # For reproducibility  > trainIndex <- createDataPartition(KidneyData$KidneyDisease, p = .8,  + list = FALSE,  + times = 1)  > KidneyTrain <- KidneyData[trainIndex, ]  > KidneyTest <- KidneyData[-trainIndex, ]  > tree\_model <- rpart(KidneyDisease ~ ., data = KidneyTrain, method = "class")  > y\_pred <- predict(tree\_model, newdata = KidneyTest, type = "class")  > confusion\_result <- confusionMatrix(factor(y\_pred, levels = levels(KidneyTest$KidneyDisease)),  + KidneyTest$KidneyDisease)  > print(confusion\_result)  Confusion Matrix and Statistics  Reference  Prediction 0 1  0 14 3  1 7 75    Accuracy : 0.899  95% CI : (0.8221, 0.9505)  No Information Rate : 0.7879  P-Value [Acc > NIR] : 0.002856    Kappa : 0.6752    Mcnemar's Test P-Value : 0.342782    Sensitivity : 0.6667  Specificity : 0.9615  Pos Pred Value : 0.8235  Neg Pred Value : 0.9146  Prevalence : 0.2121  Detection Rate : 0.1414  Detection Prevalence : 0.1717  Balanced Accuracy : 0.8141    'Positive' Class : 0    > # Plot the tree  > rpart.plot(tree\_model)  > # Print the confusion matrix result  > print(confusion\_result)  Confusion Matrix and Statistics  Reference  Prediction 0 1  0 14 3  1 7 75    Accuracy : 0.899  95% CI : (0.8221, 0.9505)  No Information Rate : 0.7879  P-Value [Acc > NIR] : 0.002856    Kappa : 0.6752    Mcnemar's Test P-Value : 0.342782    Sensitivity : 0.6667  Specificity : 0.9615  Pos Pred Value : 0.8235  Neg Pred Value : 0.9146  Prevalence : 0.2121  Detection Rate : 0.1414  Detection Prevalence : 0.1717  Balanced Accuracy : 0.8141    'Positive' Class : 0    > # Plot the tree  > rpart.plot(tree\_model)  > # Create confusion matrix  > confusion\_result <- confusionMatrix(factor(y\_pred, levels = levels(KidneyTest$KidneyDisease)),  + KidneyTest$KidneyDisease)  > decision\_tree\_model <- rpart(KidneyDisease ~ ., data = KidneyTrain, method = "class")  > rpart.plot(decision\_tree\_model,  + type = 4,  + extra = 101,  + under = TRUE,  + fallen.leaves = TRUE,  + digits = 2)  > model\_poly <- glm(KidneyDisease ~ poly(Turbidity, 4)+  + poly(ElectricConductivity,4), data = KidneyData, family = binomial)  > summary(model\_poly)  Call:  glm(formula = KidneyDisease ~ poly(Turbidity, 4) + poly(ElectricConductivity,  4), family = binomial, data = KidneyData)  Coefficients:  Estimate Std. Error z value Pr(>|z|)  (Intercept) 4.1846 3.1947 1.310 0.190242  poly(Turbidity, 4)1 14.0853 3.5542 3.963 7.4e-05 \*\*\*  poly(Turbidity, 4)2 -0.1227 3.6827 -0.033 0.973419  poly(Turbidity, 4)3 -14.9100 3.9559 -3.769 0.000164 \*\*\*  poly(Turbidity, 4)4 3.1613 3.9747 0.795 0.426398  poly(ElectricConductivity, 4)1 -101.6966 124.3433 -0.818 0.413432  poly(ElectricConductivity, 4)2 30.4863 116.7651 0.261 0.794023  poly(ElectricConductivity, 4)3 10.5803 68.4995 0.154 0.877248  poly(ElectricConductivity, 4)4 -11.5039 25.6491 -0.449 0.653784  ---  Signif. codes: 0 ‘\*\*\*’ 0.001 ‘\*\*’ 0.01 ‘\*’ 0.05 ‘.’ 0.1 ‘ ’ 1  (Dispersion parameter for binomial family taken to be 1)  Null deviance: 519.21 on 499 degrees of freedom  Residual deviance: 281.49 on 491 degrees of freedom  AIC: 299.49  Number of Fisher Scoring iterations: 13  > KidneyData\_scaled\_unique <- KidneyData\_scaled[!duplicated(KidneyData\_scaled), ]  > nrow(KidneyData\_scaled\_unique)  [1] 0  > numeric\_columns <- sapply(KidneyData, is.numeric)  > KidneyData\_numeric <- KidneyData[, numeric\_columns]  > KidneyData\_scaled <- scale(KidneyData\_numeric)  > distance\_matrix <- dist(KidneyData\_scaled, method = "euclidean")  > hc <- hclust(distance\_matrix, method = "complete")  > plot(hc, main = "Dendrogram for Hierarchical Clustering", xlab = "", sub = "", cex = 0.9)  > cluster\_groups <- cutree(hc, k = 2)  > print(cluster\_groups)  [1] 1 1 1 1 2 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1  [45] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1  [89] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1  [133] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1  [177] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1  [221] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1  [265] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1  [309] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1  [353] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1  [397] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1  [441] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1  [485] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1  > numeric\_columns <- sapply(KidneyData, is.numeric)  > KidneyData\_numeric <- KidneyData[, numeric\_columns]  > KidneyData\_scaled <- scale(KidneyData\_numeric)  > set.seed(123) # Set seed for reproducibility  > kmeans\_result <- kmeans(KidneyData\_scaled, centers = 2, nstart = 25)  > print(kmeans\_result)  K-means clustering with 2 clusters of sizes 263, 237  Cluster means:  Age BloodPressure BloodSugar Cholesterol BMI ElectricConductivity  1 0.03920646 -0.008365804 0.04779144 -0.02432831 -0.1898601 -0.6660079  2 -0.04350759 0.009283572 -0.05303438 0.02699724 0.2106886 0.7390721  pH DissolvedOxygen Turbidity TotalDissolvedSolids NitriteLevel NitrateLevel  1 -0.02968491 -0.02862934 -0.1962029 -0.7055509 -0.04684328 -0.05249536  2 0.03294148 0.03177011 0.2177272 0.7829531 0.05198221 0.05825434  LeadConcentration ArsenicConcentration Humidity  1 -0.08970804 -0.1227803 0.09881629  2 0.09954943 0.1362498 -0.10965690  Clustering vector:  [1] 2 2 2 2 1 2 2 1 2 1 2 1 1 1 2 1 1 1 2 2 1 1 2 2 1 2 1 1 2 2 1 2 1 1 2 2 1 2 1 1 2 1 2 2  [45] 1 2 1 1 2 1 2 2 2 2 2 2 2 2 2 2 1 2 1 1 2 2 2 2 1 1 1 1 2 1 2 2 1 1 2 1 1 2 2 1 2 1 2 1  [89] 1 1 2 2 1 1 2 1 1 2 1 1 1 2 2 1 2 1 1 2 1 2 2 2 1 1 1 1 2 1 1 2 2 1 2 2 2 2 2 1 2 2 2 1  [133] 2 1 2 1 1 1 2 1 2 2 2 1 2 1 1 2 2 1 1 1 1 2 1 1 1 1 2 1 1 2 1 2 2 1 1 1 2 2 1 2 1 2 1 2  [177] 1 2 2 1 1 2 1 2 1 2 1 1 2 1 2 2 1 2 2 1 1 2 2 2 1 2 1 1 2 1 2 1 2 2 2 1 1 1 1 2 1 1 2 2  [221] 2 1 2 2 2 2 2 1 1 1 1 1 2 2 1 1 2 1 1 2 1 2 1 1 1 2 1 2 1 2 2 2 1 1 1 2 1 2 1 1 1 1 2 1  [265] 2 2 2 1 2 2 1 1 2 2 2 1 2 1 1 1 1 2 1 2 1 1 2 1 1 2 2 1 1 2 1 1 2 2 1 2 1 2 1 1 1 1 1 2  [309] 1 1 1 1 2 1 1 2 1 2 2 1 1 2 2 1 2 1 2 1 2 1 2 1 2 1 2 1 2 1 2 2 1 1 1 2 1 1 1 1 2 2 2 2  [353] 2 1 1 1 2 2 2 1 1 2 2 1 1 2 2 2 1 1 1 1 1 2 2 1 2 1 2 1 1 2 1 2 1 1 1 2 2 1 1 1 2 1 1 2  [397] 2 2 1 1 1 1 1 2 2 2 1 2 1 1 2 1 2 2 1 1 1 1 1 1 2 1 1 2 2 2 1 2 2 2 1 2 2 1 2 2 1 1 1 1  [441] 1 2 1 2 1 1 2 2 1 2 2 2 2 2 1 2 1 2 1 2 2 2 1 1 2 1 2 1 1 1 2 1 1 2 1 2 2 1 2 1 1 2 1 1  [485] 2 2 1 1 2 2 1 2 1 1 2 1 1 1 1 1  Within cluster sum of squares by cluster:  [1] 3608.038 3288.856  (between\_SS / total\_SS = 7.9 %)  Available components:  [1] "cluster" "centers" "totss" "withinss" "tot.withinss" "betweenss"  [7] "size" "iter" "ifault"  > # View the cluster assignment for each data point  > kmeans\_result$cluster  [1] 2 2 2 2 1 2 2 1 2 1 2 1 1 1 2 1 1 1 2 2 1 1 2 2 1 2 1 1 2 2 1 2 1 1 2 2 1 2 1 1 2 1 2 2  [45] 1 2 1 1 2 1 2 2 2 2 2 2 2 2 2 2 1 2 1 1 2 2 2 2 1 1 1 1 2 1 2 2 1 1 2 1 1 2 2 1 2 1 2 1  [89] 1 1 2 2 1 1 2 1 1 2 1 1 1 2 2 1 2 1 1 2 1 2 2 2 1 1 1 1 2 1 1 2 2 1 2 2 2 2 2 1 2 2 2 1  [133] 2 1 2 1 1 1 2 1 2 2 2 1 2 1 1 2 2 1 1 1 1 2 1 1 1 1 2 1 1 2 1 2 2 1 1 1 2 2 1 2 1 2 1 2  [177] 1 2 2 1 1 2 1 2 1 2 1 1 2 1 2 2 1 2 2 1 1 2 2 2 1 2 1 1 2 1 2 1 2 2 2 1 1 1 1 2 1 1 2 2  [221] 2 1 2 2 2 2 2 1 1 1 1 1 2 2 1 1 2 1 1 2 1 2 1 1 1 2 1 2 1 2 2 2 1 1 1 2 1 2 1 1 1 1 2 1  [265] 2 2 2 1 2 2 1 1 2 2 2 1 2 1 1 1 1 2 1 2 1 1 2 1 1 2 2 1 1 2 1 1 2 2 1 2 1 2 1 1 1 1 1 2  [309] 1 1 1 1 2 1 1 2 1 2 2 1 1 2 2 1 2 1 2 1 2 1 2 1 2 1 2 1 2 1 2 2 1 1 1 2 1 1 1 1 2 2 2 2  [353] 2 1 1 1 2 2 2 1 1 2 2 1 1 2 2 2 1 1 1 1 1 2 2 1 2 1 2 1 1 2 1 2 1 1 1 2 2 1 1 1 2 1 1 2  [397] 2 2 1 1 1 1 1 2 2 2 1 2 1 1 2 1 2 2 1 1 1 1 1 1 2 1 1 2 2 2 1 2 2 2 1 2 2 1 2 2 1 1 1 1  [441] 1 2 1 2 1 1 2 2 1 2 2 2 2 2 1 2 1 2 1 2 2 2 1 1 2 1 2 1 1 1 2 1 1 2 1 2 2 1 2 1 1 2 1 1  [485] 2 2 1 1 2 2 1 2 1 1 2 1 1 1 1 1  > kmeans\_result$centers  Age BloodPressure BloodSugar Cholesterol BMI ElectricConductivity  1 0.03920646 -0.008365804 0.04779144 -0.02432831 -0.1898601 -0.6660079  2 -0.04350759 0.009283572 -0.05303438 0.02699724 0.2106886 0.7390721  pH DissolvedOxygen Turbidity TotalDissolvedSolids NitriteLevel NitrateLevel  1 -0.02968491 -0.02862934 -0.1962029 -0.7055509 -0.04684328 -0.05249536  2 0.03294148 0.03177011 0.2177272 0.7829531 0.05198221 0.05825434  LeadConcentration ArsenicConcentration Humidity  1 -0.08970804 -0.1227803 0.09881629  2 0.09954943 0.1362498 -0.10965690  > kmeans\_result$tot.withinss  [1] 6896.894  > # Plot clusters (for 2 dimensions)  > library(ggplot2)  > KidneyData\_clustered <- as.data.frame(KidneyData\_scaled)  > KidneyData\_clustered$cluster <- as.factor(kmeans\_result$cluster)  > ggplot(KidneyData\_clustered, aes(x = KidneyData\_clustered[,1], y = KidneyData\_clustered[,2], color = cluster)) +  + geom\_point() +  + labs(title = "K-means Clustering Results", x = "Feature 1", y = "Feature 2")  > kmeans\_result <- kmeans(KidneyData\_scaled, centers = 3, nstart = 25)  > print(kmeans\_result\_3$centers)  > print(kmeans\_result$centers)  Age BloodPressure BloodSugar Cholesterol BMI ElectricConductivity  1 0.3825705 0.4208558 -0.007511913 -0.10484032 0.1863436 0.81251606  2 0.4632341 0.2991968 -0.071482641 -0.06917587 -0.2707498 -0.80991411  3 -1.0163640 -0.8633115 0.095499964 0.20857387 0.1060091 0.01343129  pH DissolvedOxygen Turbidity TotalDissolvedSolids NitriteLevel NitrateLevel  1 -0.01050132 0.108797961 0.24204190 0.832933766 0.09207378 0.15118052  2 -0.11797885 0.007135676 -0.27098258 -0.826996773 0.03516804 0.05843835  3 0.15536032 -0.138161861 0.03998345 0.009809636 -0.15219607 -0.25073889  LeadConcentration ArsenicConcentration Humidity  1 0.13594677 0.25101219 -0.05481330  2 -0.07574255 -0.07273201 0.12866811  3 -0.07012592 -0.21075398 -0.09054828  > ggplot(KidneyData\_clustered, aes(x = KidneyData\_clustered[,1], y = KidneyData\_clustered[,2], color = cluster)) +  + geom\_point() +  + labs(title = "K-means Clustering Results", x = "Feature 1", y = "Feature 2")  > ggplot(KidneyData\_clustered, aes(x = KidneyData\_clustered[,1], y = KidneyData\_clustered[,2], color = cluster)) +  + geom\_point() +  + labs(title = "K-means Clustering Results", x = "Feature 1", y = "Feature 2")  > # Size of each cluster  > print(kmeans\_result$size)  [1] 175 178 147  > kmeans\_result <- kmeans(KidneyData\_scaled, centers = 2, nstart = 25)  > # Print the results of the clustering  > print(kmeans\_result)  K-means clustering with 2 clusters of sizes 238, 262  Cluster means:  Age BloodPressure BloodSugar Cholesterol BMI ElectricConductivity  1 -0.08287622 -0.02435777 -0.08369807 0.04248596 0.1829219 0.7316828  2 0.07528451 0.02212653 0.07603107 -0.03859411 -0.1661657 -0.6646585  pH DissolvedOxygen Turbidity TotalDissolvedSolids NitriteLevel NitrateLevel  1 0.03505465 -0.004873970 0.2223153 0.7936374 0.02452907 0.05923172  2 -0.03184354 0.004427499 -0.2019506 -0.7209378 -0.02228213 -0.05380591  LeadConcentration ArsenicConcentration Humidity  1 0.08840165 0.07959195 -0.1119165  2 -0.08030379 -0.07230108 0.1016646  Clustering vector:  [1] 1 1 1 1 2 1 1 2 1 2 1 2 2 2 1 2 2 2 1 1 2 2 1 2 2 1 2 2 1 1 2 1 1 2 1 1 2 1 2 2 1 2 1 1  [45] 2 1 2 2 1 2 1 1 1 1 1 1 1 1 1 1 2 1 2 2 1 1 1 1 2 2 2 2 1 2 1 1 2 2 1 2 2 1 1 2 1 2 1 2  [89] 2 2 1 1 2 2 1 2 2 1 2 2 2 1 1 2 1 2 2 1 2 1 1 1 2 2 2 2 1 2 2 1 1 2 1 1 1 1 1 2 1 1 1 2  [133] 1 2 1 2 2 2 1 2 1 1 1 2 1 2 2 1 1 2 2 2 2 1 2 2 2 2 1 2 2 1 2 1 1 2 2 2 1 1 2 1 2 1 2 1  [177] 2 1 1 2 2 1 2 1 1 1 2 2 1 2 2 1 2 1 1 2 2 1 1 1 2 1 2 2 1 2 1 2 1 1 1 2 2 2 2 1 2 2 1 1  [221] 2 2 1 1 1 1 1 2 2 2 2 2 1 1 2 2 1 2 2 1 2 1 2 2 2 1 2 1 2 1 1 1 2 2 2 1 2 1 2 2 2 2 1 2  [265] 1 1 1 2 1 1 2 2 1 1 1 2 1 2 2 2 2 1 2 1 2 2 1 2 2 1 1 2 2 1 2 2 1 1 2 1 2 1 2 2 2 2 2 1  [309] 2 2 2 2 1 2 2 1 2 2 1 2 2 1 1 2 1 2 1 2 1 2 2 2 1 2 1 2 1 2 1 1 2 2 2 1 2 2 2 2 1 1 1 1  [353] 1 2 2 2 1 1 1 2 2 1 1 2 2 1 1 1 2 2 2 2 2 1 1 2 1 2 1 2 2 1 2 1 2 1 2 1 1 2 2 2 1 2 2 1  [397] 1 1 2 2 2 2 2 1 1 1 2 1 2 2 1 2 1 1 2 2 1 2 2 2 1 2 2 1 1 1 2 1 1 1 2 1 1 2 1 1 2 2 2 2  [441] 1 1 2 1 2 2 1 1 2 1 1 1 1 1 2 1 2 1 2 1 1 1 2 2 1 2 1 2 2 2 1 2 2 1 2 1 1 2 1 2 2 1 2 1  [485] 1 1 2 2 1 1 2 1 2 2 1 2 2 2 2 2  Within cluster sum of squares by cluster:  [1] 3294.921 3601.249  (between\_SS / total\_SS = 7.9 %)  Available components:  [1] "cluster" "centers" "totss" "withinss" "tot.withinss" "betweenss"  [7] "size" "iter" "ifault"  > KidneyData\_clustered <- as.data.frame(KidneyData\_scaled)  > KidneyData\_clustered$cluster <- as.factor(kmeans\_result$cluster)  > ggplot(KidneyData\_clustered, aes(x = KidneyData\_clustered[,1], y = KidneyData\_clustered[,2], color = cluster)) +  + geom\_point() +  + labs(title = "K-means Clustering Results", x = "Feature 1", y = "Feature 2")  > # View the cluster centers for 3 clusters  > print(kmeans\_result$centers)  Age BloodPressure BloodSugar Cholesterol BMI ElectricConductivity  1 -0.08287622 -0.02435777 -0.08369807 0.04248596 0.1829219 0.7316828  2 0.07528451 0.02212653 0.07603107 -0.03859411 -0.1661657 -0.6646585  pH DissolvedOxygen Turbidity TotalDissolvedSolids NitriteLevel NitrateLevel  1 0.03505465 -0.004873970 0.2223153 0.7936374 0.02452907 0.05923172  2 -0.03184354 0.004427499 -0.2019506 -0.7209378 -0.02228213 -0.05380591  LeadConcentration ArsenicConcentration Humidity  1 0.08840165 0.07959195 -0.1119165  2 -0.08030379 -0.07230108 0.1016646  > # Size of each cluster  > print(kmeans\_result$size)  [1] 238 262  > # View the cluster centers for 3 clusters  > print(kmeans\_result$centers)  Age BloodPressure BloodSugar Cholesterol BMI ElectricConductivity  1 -0.08287622 -0.02435777 -0.08369807 0.04248596 0.1829219 0.7316828  2 0.07528451 0.02212653 0.07603107 -0.03859411 -0.1661657 -0.6646585  pH DissolvedOxygen Turbidity TotalDissolvedSolids NitriteLevel NitrateLevel  1 0.03505465 -0.004873970 0.2223153 0.7936374 0.02452907 0.05923172  2 -0.03184354 0.004427499 -0.2019506 -0.7209378 -0.02228213 -0.05380591  LeadConcentration ArsenicConcentration Humidity  1 0.08840165 0.07959195 -0.1119165  2 -0.08030379 -0.07230108 0.1016646  > # Size of each cluster  > print(kmeans\_result$size)  [1] 238 262  > colnames(KidneyData)  [1] "Age" "Gender" "BloodPressure"  [4] "BloodSugar" "Cholesterol" "BMI"  [7] "SmokingStatus" "ElectricConductivity" "pH"  [10] "DissolvedOxygen" "Turbidity" "TotalDissolvedSolids"  [13] "NitriteLevel" "NitrateLevel" "LeadConcentration"  [16] "ArsenicConcentration" "Humidity" "KidneyDisease"  > KidneyData\_scaled <- scale(KidneyData)  > ggplot(KidneyData\_clustered, aes(x = KidneyData\_clustered[,1], y = KidneyData\_clustered[,2], color = cluster)) +  + geom\_point() +  + labs(title = "K-means Clustering Results", x = "TotalDissolvedSolids", y = "Turbidity")  > KidneyData\_scaled <- scale(KidneyData)  > str(KidneyData)  'data.frame': 500 obs. of 18 variables:  $ Age : int 120 10 58 22 52 53 76 45 57 30 ...  $ Gender : chr "Female" "Female" "Female" "Female" ...  $ BloodPressure : int 118 143 300 20 150 141 194 151 140 141 ...  $ BloodSugar : num 156 162 121 154 159 ...  $ Cholesterol : int 165 214 222 212 600 199 251 200 215 205 ...  $ BMI : num 31.7 23.9 16.3 21.9 23.8 18.3 26.2 22.2 19.5 25.7 ...  $ SmokingStatus : Factor w/ 3 levels "Current","Former",..: 2 3 2 3 1 2 1 1 2 2 ...  $ ElectricConductivity: num 336 297 378 312 222 ...  $ pH : num 7.4 7.48 7.49 6.03 6.77 7.34 7.01 7.46 7.38 6.7 ...  $ DissolvedOxygen : num 9.57 8.49 8.18 7.35 7.4 8 9.79 8.72 8.04 6.98 ...  $ Turbidity : num 1.44 1.21 0.88 1.15 0.73 0.71 1.16 0.98 1.47 1.1 ...  $ TotalDissolvedSolids: num 455 423 434 400 349 ...  $ NitriteLevel : num 0.165 0.075 0.005 0.088 0.119 0.076 0.177 0.044 0.114 0.042 ...  $ NitrateLevel : num 1.97 1.74 1.4 0.88 0.71 1 1.13 1.13 1.13 0.82 ...  $ LeadConcentration : num 0.0099 0.012 0.0173 0.0133 0.0155 0.005 0.012 0.0106 0.0128 0.0145 ...  $ ArsenicConcentration: num 0.0063 0.0062 0.0092 0.0086 0.0011 0.009 0.0035 0.0062 0.0081 0.0046 ...  $ Humidity : num 48.7 65.3 93.2 67.4 43.3 57.6 50.8 70.5 55.6 72.9 ...  $ KidneyDisease : Factor w/ 2 levels "0","1": 1 2 1 2 2 1 2 2 1 2 ...  > # Create a new data frame for predictions  > new\_data <- expand.grid(  + Turbidity = seq(min(KidneyData$Turbidity), max(KidneyData$Turbidity), length.out = 100),  + TotalDissolvedSolids = seq(min(KidneyData$TotalDissolvedSolids), max(KidneyData$TotalDissolvedSolids), length.out = 100)  + )  > new\_data$predicted\_prob <- predict(model\_poly\_turbidity\_tds, newdata = new\_data, type = "response")  > ggplot(new\_data, aes(x = Turbidity, y = predicted\_prob, color = TotalDissolvedSolids)) +  + geom\_line() +  + labs(title = "Predicted Probability of Kidney Disease",  + x = "Turbidity",  + y = "Predicted Probability") +  + theme\_minimal()  > model\_poly <- glm(KidneyDisease ~ poly(Turbidity, 4)+  + poly(ElectricConductivity,4)+poly(TotalDissolvedSolids,4), data = KidneyData, family = binomial)  > summary(model\_poly)  Call:  glm(formula = KidneyDisease ~ poly(Turbidity, 4) + poly(ElectricConductivity,  4) + poly(TotalDissolvedSolids, 4), family = binomial, data = KidneyData)  Coefficients:  Estimate Std. Error z value Pr(>|z|)  (Intercept) 4.6951 2.1149 2.220 0.026417 \*  poly(Turbidity, 4)1 14.9239 3.6845 4.050 5.11e-05 \*\*\*  poly(Turbidity, 4)2 0.9021 3.7655 0.240 0.810663  poly(Turbidity, 4)3 -15.3540 4.1172 -3.729 0.000192 \*\*\*  poly(Turbidity, 4)4 5.5535 4.0580 1.369 0.171141  poly(ElectricConductivity, 4)1 -104.9161 82.6231 -1.270 0.204150  poly(ElectricConductivity, 4)2 45.5221 75.7209 0.601 0.547718  poly(ElectricConductivity, 4)3 0.5985 43.7711 0.014 0.989090  poly(ElectricConductivity, 4)4 -8.3731 16.7199 -0.501 0.616523  poly(TotalDissolvedSolids, 4)1 -16.2878 10.6529 -1.529 0.126275  poly(TotalDissolvedSolids, 4)2 -11.9260 13.2205 -0.902 0.367014  poly(TotalDissolvedSolids, 4)3 15.9827 12.3633 1.293 0.196096  poly(TotalDissolvedSolids, 4)4 -9.7496 10.4346 -0.934 0.350119  ---  Signif. codes: 0 ‘\*\*\*’ 0.001 ‘\*\*’ 0.01 ‘\*’ 0.05 ‘.’ 0.1 ‘ ’ 1  (Dispersion parameter for binomial family taken to be 1)  Null deviance: 519.21 on 499 degrees of freedom  Residual deviance: 261.61 on 487 degrees of freedom  AIC: 287.61  Number of Fisher Scoring iterations: 13  > model\_poly <- glm(KidneyDisease ~ poly(Turbidity, 2)+  + poly(ElectricConductivity,2)+poly(TotalDissolvedSolids,2), data = KidneyData, family = binomial)  > summary(model\_poly)  Call:  glm(formula = KidneyDisease ~ poly(Turbidity, 2) + poly(ElectricConductivity,  2) + poly(TotalDissolvedSolids, 2), family = binomial, data = KidneyData)  Coefficients:  Estimate Std. Error z value Pr(>|z|)  (Intercept) 3.9645 0.5196 7.630 2.34e-14 \*\*\*  poly(Turbidity, 2)1 14.9679 3.4241 4.371 1.24e-05 \*\*\*  poly(Turbidity, 2)2 -2.0282 3.2244 -0.629 0.52934  poly(ElectricConductivity, 2)1 -79.3535 14.0366 -5.653 1.57e-08 \*\*\*  poly(ElectricConductivity, 2)2 32.8256 8.2078 3.999 6.35e-05 \*\*\*  poly(TotalDissolvedSolids, 2)1 -26.9294 8.7578 -3.075 0.00211 \*\*  poly(TotalDissolvedSolids, 2)2 5.3941 6.4867 0.832 0.40566  ---  Signif. codes: 0 ‘\*\*\*’ 0.001 ‘\*\*’ 0.01 ‘\*’ 0.05 ‘.’ 0.1 ‘ ’ 1  (Dispersion parameter for binomial family taken to be 1)  Null deviance: 519.21 on 499 degrees of freedom  Residual deviance: 282.97 on 493 degrees of freedom  AIC: 296.97  Number of Fisher Scoring iterations: 8  > model\_poly <- glm(KidneyDisease ~ poly(Turbidity, 2)+  + poly(ElectricConductivity,2)+poly(TotalDissolvedSolids,2),  + data = KidneyData, family = binomial)  > summary(model\_poly)  Call:  glm(formula = KidneyDisease ~ poly(Turbidity, 2) + poly(ElectricConductivity,  2) + poly(TotalDissolvedSolids, 2), family = binomial, data = KidneyData)  Coefficients:  Estimate Std. Error z value Pr(>|z|)  (Intercept) 3.9645 0.5196 7.630 2.34e-14 \*\*\*  poly(Turbidity, 2)1 14.9679 3.4241 4.371 1.24e-05 \*\*\*  poly(Turbidity, 2)2 -2.0282 3.2244 -0.629 0.52934  poly(ElectricConductivity, 2)1 -79.3535 14.0366 -5.653 1.57e-08 \*\*\*  poly(ElectricConductivity, 2)2 32.8256 8.2078 3.999 6.35e-05 \*\*\*  poly(TotalDissolvedSolids, 2)1 -26.9294 8.7578 -3.075 0.00211 \*\*  poly(TotalDissolvedSolids, 2)2 5.3941 6.4867 0.832 0.40566  ---  Signif. codes: 0 ‘\*\*\*’ 0.001 ‘\*\*’ 0.01 ‘\*’ 0.05 ‘.’ 0.1 ‘ ’ 1  (Dispersion parameter for binomial family taken to be 1)  Null deviance: 519.21 on 499 degrees of freedom  Residual deviance: 282.97 on 493 degrees of freedom  AIC: 296.97  Number of Fisher Scoring iterations: 8  > confusion\_result <- confusionMatrix(factor(y\_pred, levels = levels(KidneyTest$KidneyDisease)),  + KidneyTest$KidneyDisease)  > print(confusion\_result)  Confusion Matrix and Statistics  Reference  Prediction 0 1  0 14 3  1 7 75    Accuracy : 0.899  95% CI : (0.8221, 0.9505)  No Information Rate : 0.7879  P-Value [Acc > NIR] : 0.002856    Kappa : 0.6752    Mcnemar's Test P-Value : 0.342782    Sensitivity : 0.6667  Specificity : 0.9615  Pos Pred Value : 0.8235  Neg Pred Value : 0.9146  Prevalence : 0.2121  Detection Rate : 0.1414  Detection Prevalence : 0.1717  Balanced Accuracy : 0.8141    'Positive' Class : 0    > model\_poly <- glm(KidneyDisease ~ poly(Turbidity, 2)+  + poly(ElectricConductivity,2)+poly(TotalDissolvedSolids,2),  + data = KidneyData, family = binomial)  > summary(model\_poly)  Call:  glm(formula = KidneyDisease ~ poly(Turbidity, 2) + poly(ElectricConductivity,  2) + poly(TotalDissolvedSolids, 2), family = binomial, data = KidneyData)  Coefficients:  Estimate Std. Error z value Pr(>|z|)  (Intercept) 3.9645 0.5196 7.630 2.34e-14 \*\*\*  poly(Turbidity, 2)1 14.9679 3.4241 4.371 1.24e-05 \*\*\*  poly(Turbidity, 2)2 -2.0282 3.2244 -0.629 0.52934  poly(ElectricConductivity, 2)1 -79.3535 14.0366 -5.653 1.57e-08 \*\*\*  poly(ElectricConductivity, 2)2 32.8256 8.2078 3.999 6.35e-05 \*\*\*  poly(TotalDissolvedSolids, 2)1 -26.9294 8.7578 -3.075 0.00211 \*\*  poly(TotalDissolvedSolids, 2)2 5.3941 6.4867 0.832 0.40566  ---  Signif. codes: 0 ‘\*\*\*’ 0.001 ‘\*\*’ 0.01 ‘\*’ 0.05 ‘.’ 0.1 ‘ ’ 1  (Dispersion parameter for binomial family taken to be 1)  Null deviance: 519.21 on 499 degrees of freedom  Residual deviance: 282.97 on 493 degrees of freedom  AIC: 296.97  Number of Fisher Scoring iterations: 8  > KidneyData\_clustered <- as.data.frame(KidneyData\_scaled)  > KidneyData\_clustered$cluster <- as.factor(kmeans\_result$cluster)  > ggplot(KidneyData\_clustered, aes(x = KidneyData\_clustered[,1], y = KidneyData\_clustered[,4], color = cluster)) +  + geom\_point() +  + labs(title = "K-means Clustering Results", x = "TotalDissolvedSolids", y = "Turbidity")  > print(kmeans\_result$centers)  Age BloodPressure BloodSugar Cholesterol BMI ElectricConductivity  1 -0.08287622 -0.02435777 -0.08369807 0.04248596 0.1829219 0.7316828  2 0.07528451 0.02212653 0.07603107 -0.03859411 -0.1661657 -0.6646585  pH DissolvedOxygen Turbidity TotalDissolvedSolids NitriteLevel NitrateLevel  1 0.03505465 -0.004873970 0.2223153 0.7936374 0.02452907 0.05923172  2 -0.03184354 0.004427499 -0.2019506 -0.7209378 -0.02228213 -0.05380591  LeadConcentration ArsenicConcentration Humidity  1 0.08840165 0.07959195 -0.1119165  2 -0.08030379 -0.07230108 0.1016646  > print(kmeans\_result$size)  [1] 238 262  > numeric\_columns <- sapply(KidneyData, is.numeric)  > KidneyData\_numeric <- KidneyData[, numeric\_columns]  > KidneyData\_scaled <- scale(KidneyData\_numeric)  > set.seed(123) # Set seed for reproducibility  > kmeans\_result <- kmeans(KidneyData\_scaled, centers = 4, nstart = 25)  > print(kmeans\_result)  K-means clustering with 4 clusters of sizes 129, 144, 128, 99  Cluster means:  Age BloodPressure BloodSugar Cholesterol BMI ElectricConductivity  1 0.6704254 0.5067659 0.02135505 0.17337830 -0.39206774 -0.7375164  2 -0.8295091 -0.6726932 0.02986499 -0.07781065 0.20440398 -0.5251603  3 -0.3787645 -0.3210074 -0.18744619 0.08680150 0.02483061 0.9324206  4 0.8226898 0.7331714 0.17108840 -0.22496625 0.18145704 0.5193217  pH DissolvedOxygen Turbidity TotalDissolvedSolids NitriteLevel NitrateLevel  1 -0.17108958 0.20646812 -0.4183163 -0.7107107 0.25563864 -0.05363636  2 0.09805321 -0.12804798 0.1209902 -0.5527599 -0.36302165 -0.06643528  3 -0.36158468 -0.01028971 -0.2416839 0.9469990 0.19928309 -0.10347830  4 0.54781548 -0.06947874 0.6815733 0.5056891 -0.06273235 0.30031307  LeadConcentration ArsenicConcentration Humidity  1 -0.114773846 -0.19411648 -0.1764593  2 -0.008148328 -0.14436154 0.2862951  3 0.011066323 -0.02123068 -0.2744147  4 0.147097941 0.49036985 0.1683012  Clustering vector:  [1] 4 2 4 2 1 3 1 2 4 2 3 1 1 2 4 1 2 2 3 3 3 1 3 4 1 3 1 4 3 3 2 4 2 1 4 4 2 3 1 2 4 4 4 4  [45] 1 4 1 1 3 2 2 3 3 3 4 3 3 3 1 2 1 4 4 1 3 3 3 3 2 1 2 2 4 4 3 4 1 1 4 2 2 4 4 1 3 1 4 2  [89] 2 1 4 3 1 2 4 2 1 2 2 1 1 3 2 1 3 2 1 3 2 3 3 4 2 2 1 2 4 1 2 4 3 2 3 4 3 2 4 4 4 3 4 4  [133] 4 1 3 2 2 2 3 2 3 2 3 1 2 1 4 3 2 2 1 1 2 3 2 1 1 4 2 1 1 3 3 3 3 2 2 1 4 4 2 3 1 3 1 3  [177] 1 4 3 1 1 4 1 4 2 3 2 2 3 1 4 3 2 4 4 2 2 3 3 3 1 4 1 1 3 4 4 1 3 3 3 1 1 2 2 4 1 1 3 2  [221] 4 2 3 4 4 3 3 2 2 2 1 2 4 3 2 1 3 1 1 4 2 3 4 1 2 1 4 2 2 3 2 4 2 2 1 1 2 4 2 1 1 1 3 4  [265] 4 3 3 2 4 4 1 1 4 3 4 1 3 2 4 2 2 2 2 3 2 1 3 1 3 3 3 2 2 3 3 2 3 3 1 4 1 4 1 2 2 1 2 3  [309] 2 1 2 1 2 2 1 3 2 1 4 1 1 3 4 1 4 1 4 1 3 4 4 2 3 2 3 2 3 2 4 3 1 1 1 2 1 2 1 1 3 3 3 4  [353] 3 2 1 2 4 2 4 2 1 3 4 2 1 4 4 3 2 1 1 1 2 2 3 2 3 2 3 1 2 3 1 2 1 1 2 4 4 1 1 2 4 1 1 3  [397] 2 3 1 1 2 3 3 2 4 3 1 4 4 2 3 1 3 4 2 1 2 1 3 1 3 1 1 4 2 2 2 3 3 3 1 3 3 2 3 4 1 4 2 1  [441] 2 3 2 4 1 1 4 3 1 3 3 2 4 4 2 4 1 3 2 3 3 4 1 1 3 1 2 2 2 2 3 2 2 3 1 4 4 1 3 2 2 3 1 2  [485] 3 2 2 1 3 2 1 3 2 2 3 2 2 2 1 2  Within cluster sum of squares by cluster:  [1] 1682.801 1763.362 1545.831 1295.107  (between\_SS / total\_SS = 16.0 %)  Available components:  [1] "cluster" "centers" "totss" "withinss" "tot.withinss" "betweenss"  [7] "size" "iter" "ifault"  > kmeans\_result$cluster  [1] 4 2 4 2 1 3 1 2 4 2 3 1 1 2 4 1 2 2 3 3 3 1 3 4 1 3 1 4 3 3 2 4 2 1 4 4 2 3 1 2 4 4 4 4  [45] 1 4 1 1 3 2 2 3 3 3 4 3 3 3 1 2 1 4 4 1 3 3 3 3 2 1 2 2 4 4 3 4 1 1 4 2 2 4 4 1 3 1 4 2  [89] 2 1 4 3 1 2 4 2 1 2 2 1 1 3 2 1 3 2 1 3 2 3 3 4 2 2 1 2 4 1 2 4 3 2 3 4 3 2 4 4 4 3 4 4  [133] 4 1 3 2 2 2 3 2 3 2 3 1 2 1 4 3 2 2 1 1 2 3 2 1 1 4 2 1 1 3 3 3 3 2 2 1 4 4 2 3 1 3 1 3  [177] 1 4 3 1 1 4 1 4 2 3 2 2 3 1 4 3 2 4 4 2 2 3 3 3 1 4 1 1 3 4 4 1 3 3 3 1 1 2 2 4 1 1 3 2  [221] 4 2 3 4 4 3 3 2 2 2 1 2 4 3 2 1 3 1 1 4 2 3 4 1 2 1 4 2 2 3 2 4 2 2 1 1 2 4 2 1 1 1 3 4  [265] 4 3 3 2 4 4 1 1 4 3 4 1 3 2 4 2 2 2 2 3 2 1 3 1 3 3 3 2 2 3 3 2 3 3 1 4 1 4 1 2 2 1 2 3  [309] 2 1 2 1 2 2 1 3 2 1 4 1 1 3 4 1 4 1 4 1 3 4 4 2 3 2 3 2 3 2 4 3 1 1 1 2 1 2 1 1 3 3 3 4  [353] 3 2 1 2 4 2 4 2 1 3 4 2 1 4 4 3 2 1 1 1 2 2 3 2 3 2 3 1 2 3 1 2 1 1 2 4 4 1 1 2 4 1 1 3  [397] 2 3 1 1 2 3 3 2 4 3 1 4 4 2 3 1 3 4 2 1 2 1 3 1 3 1 1 4 2 2 2 3 3 3 1 3 3 2 3 4 1 4 2 1  [441] 2 3 2 4 1 1 4 3 1 3 3 2 4 4 2 4 1 3 2 3 3 4 1 1 3 1 2 2 2 2 3 2 2 3 1 4 4 1 3 2 2 3 1 2  [485] 3 2 2 1 3 2 1 3 2 2 3 2 2 2 1 2  > kmeans\_result$centers  Age BloodPressure BloodSugar Cholesterol BMI ElectricConductivity  1 0.6704254 0.5067659 0.02135505 0.17337830 -0.39206774 -0.7375164  2 -0.8295091 -0.6726932 0.02986499 -0.07781065 0.20440398 -0.5251603  3 -0.3787645 -0.3210074 -0.18744619 0.08680150 0.02483061 0.9324206  4 0.8226898 0.7331714 0.17108840 -0.22496625 0.18145704 0.5193217  pH DissolvedOxygen Turbidity TotalDissolvedSolids NitriteLevel NitrateLevel  1 -0.17108958 0.20646812 -0.4183163 -0.7107107 0.25563864 -0.05363636  2 0.09805321 -0.12804798 0.1209902 -0.5527599 -0.36302165 -0.06643528  3 -0.36158468 -0.01028971 -0.2416839 0.9469990 0.19928309 -0.10347830  4 0.54781548 -0.06947874 0.6815733 0.5056891 -0.06273235 0.30031307  LeadConcentration ArsenicConcentration Humidity  1 -0.114773846 -0.19411648 -0.1764593  2 -0.008148328 -0.14436154 0.2862951  3 0.011066323 -0.02123068 -0.2744147  4 0.147097941 0.49036985 0.1683012  > library(ggplot2)  > KidneyData\_clustered <- as.data.frame(KidneyData\_scaled)  > KidneyData\_clustered$cluster <- as.factor(kmeans\_result$cluster)  > ggplot(KidneyData\_clustered, aes(x = KidneyData\_clustered[,1], y = KidneyData\_clustered[,2], color = cluster)) +  + geom\_point() +  + labs(title = "K-means Clustering Results", x = "TotalDissolvedSolids", y = "Turbidity")  > print(kmeans\_result$size)  [1] 129 144 128 99  > colnames(KidneyData)  [1] "Age" "Gender" "BloodPressure"  [4] "BloodSugar" "Cholesterol" "BMI"  [7] "SmokingStatus" "ElectricConductivity" "pH"  [10] "DissolvedOxygen" "Turbidity" "TotalDissolvedSolids"  [13] "NitriteLevel" "NitrateLevel" "LeadConcentration"  [16] "ArsenicConcentration" "Humidity" "KidneyDisease"  > numeric\_columns <- sapply(KidneyData, is.numeric)  > KidneyData\_numeric <- KidneyData[, numeric\_columns]  > KidneyData\_scaled <- scale(KidneyData\_numeric)  > set.seed(123)  > kmeans\_result <- kmeans(KidneyData\_scaled, centers = 4, nstart = 25)  > print(kmeans\_result)  K-means clustering with 4 clusters of sizes 129, 144, 128, 99  Cluster means:  Age BloodPressure BloodSugar Cholesterol BMI ElectricConductivity  1 0.6704254 0.5067659 0.02135505 0.17337830 -0.39206774 -0.7375164  2 -0.8295091 -0.6726932 0.02986499 -0.07781065 0.20440398 -0.5251603  3 -0.3787645 -0.3210074 -0.18744619 0.08680150 0.02483061 0.9324206  4 0.8226898 0.7331714 0.17108840 -0.22496625 0.18145704 0.5193217  pH DissolvedOxygen Turbidity TotalDissolvedSolids NitriteLevel NitrateLevel  1 -0.17108958 0.20646812 -0.4183163 -0.7107107 0.25563864 -0.05363636  2 0.09805321 -0.12804798 0.1209902 -0.5527599 -0.36302165 -0.06643528  3 -0.36158468 -0.01028971 -0.2416839 0.9469990 0.19928309 -0.10347830  4 0.54781548 -0.06947874 0.6815733 0.5056891 -0.06273235 0.30031307  LeadConcentration ArsenicConcentration Humidity  1 -0.114773846 -0.19411648 -0.1764593  2 -0.008148328 -0.14436154 0.2862951  3 0.011066323 -0.02123068 -0.2744147  4 0.147097941 0.49036985 0.1683012  Clustering vector:  [1] 4 2 4 2 1 3 1 2 4 2 3 1 1 2 4 1 2 2 3 3 3 1 3 4 1 3 1 4 3 3 2 4 2 1 4 4 2 3 1 2 4 4 4 4  [45] 1 4 1 1 3 2 2 3 3 3 4 3 3 3 1 2 1 4 4 1 3 3 3 3 2 1 2 2 4 4 3 4 1 1 4 2 2 4 4 1 3 1 4 2  [89] 2 1 4 3 1 2 4 2 1 2 2 1 1 3 2 1 3 2 1 3 2 3 3 4 2 2 1 2 4 1 2 4 3 2 3 4 3 2 4 4 4 3 4 4  [133] 4 1 3 2 2 2 3 2 3 2 3 1 2 1 4 3 2 2 1 1 2 3 2 1 1 4 2 1 1 3 3 3 3 2 2 1 4 4 2 3 1 3 1 3  [177] 1 4 3 1 1 4 1 4 2 3 2 2 3 1 4 3 2 4 4 2 2 3 3 3 1 4 1 1 3 4 4 1 3 3 3 1 1 2 2 4 1 1 3 2  [221] 4 2 3 4 4 3 3 2 2 2 1 2 4 3 2 1 3 1 1 4 2 3 4 1 2 1 4 2 2 3 2 4 2 2 1 1 2 4 2 1 1 1 3 4  [265] 4 3 3 2 4 4 1 1 4 3 4 1 3 2 4 2 2 2 2 3 2 1 3 1 3 3 3 2 2 3 3 2 3 3 1 4 1 4 1 2 2 1 2 3  [309] 2 1 2 1 2 2 1 3 2 1 4 1 1 3 4 1 4 1 4 1 3 4 4 2 3 2 3 2 3 2 4 3 1 1 1 2 1 2 1 1 3 3 3 4  [353] 3 2 1 2 4 2 4 2 1 3 4 2 1 4 4 3 2 1 1 1 2 2 3 2 3 2 3 1 2 3 1 2 1 1 2 4 4 1 1 2 4 1 1 3  [397] 2 3 1 1 2 3 3 2 4 3 1 4 4 2 3 1 3 4 2 1 2 1 3 1 3 1 1 4 2 2 2 3 3 3 1 3 3 2 3 4 1 4 2 1  [441] 2 3 2 4 1 1 4 3 1 3 3 2 4 4 2 4 1 3 2 3 3 4 1 1 3 1 2 2 2 2 3 2 2 3 1 4 4 1 3 2 2 3 1 2  [485] 3 2 2 1 3 2 1 3 2 2 3 2 2 2 1 2  Within cluster sum of squares by cluster:  [1] 1682.801 1763.362 1545.831 1295.107  (between\_SS / total\_SS = 16.0 %)  Available components:  [1] "cluster" "centers" "totss" "withinss" "tot.withinss" "betweenss"  [7] "size" "iter" "ifault"  > kmeans\_result$cluster  [1] 4 2 4 2 1 3 1 2 4 2 3 1 1 2 4 1 2 2 3 3 3 1 3 4 1 3 1 4 3 3 2 4 2 1 4 4 2 3 1 2 4 4 4 4  [45] 1 4 1 1 3 2 2 3 3 3 4 3 3 3 1 2 1 4 4 1 3 3 3 3 2 1 2 2 4 4 3 4 1 1 4 2 2 4 4 1 3 1 4 2  [89] 2 1 4 3 1 2 4 2 1 2 2 1 1 3 2 1 3 2 1 3 2 3 3 4 2 2 1 2 4 1 2 4 3 2 3 4 3 2 4 4 4 3 4 4  [133] 4 1 3 2 2 2 3 2 3 2 3 1 2 1 4 3 2 2 1 1 2 3 2 1 1 4 2 1 1 3 3 3 3 2 2 1 4 4 2 3 1 3 1 3  [177] 1 4 3 1 1 4 1 4 2 3 2 2 3 1 4 3 2 4 4 2 2 3 3 3 1 4 1 1 3 4 4 1 3 3 3 1 1 2 2 4 1 1 3 2  [221] 4 2 3 4 4 3 3 2 2 2 1 2 4 3 2 1 3 1 1 4 2 3 4 1 2 1 4 2 2 3 2 4 2 2 1 1 2 4 2 1 1 1 3 4  [265] 4 3 3 2 4 4 1 1 4 3 4 1 3 2 4 2 2 2 2 3 2 1 3 1 3 3 3 2 2 3 3 2 3 3 1 4 1 4 1 2 2 1 2 3  [309] 2 1 2 1 2 2 1 3 2 1 4 1 1 3 4 1 4 1 4 1 3 4 4 2 3 2 3 2 3 2 4 3 1 1 1 2 1 2 1 1 3 3 3 4  [353] 3 2 1 2 4 2 4 2 1 3 4 2 1 4 4 3 2 1 1 1 2 2 3 2 3 2 3 1 2 3 1 2 1 1 2 4 4 1 1 2 4 1 1 3  [397] 2 3 1 1 2 3 3 2 4 3 1 4 4 2 3 1 3 4 2 1 2 1 3 1 3 1 1 4 2 2 2 3 3 3 1 3 3 2 3 4 1 4 2 1  [441] 2 3 2 4 1 1 4 3 1 3 3 2 4 4 2 4 1 3 2 3 3 4 1 1 3 1 2 2 2 2 3 2 2 3 1 4 4 1 3 2 2 3 1 2  [485] 3 2 2 1 3 2 1 3 2 2 3 2 2 2 1 2  > kmeans\_result$centers  Age BloodPressure BloodSugar Cholesterol BMI ElectricConductivity  1 0.6704254 0.5067659 0.02135505 0.17337830 -0.39206774 -0.7375164  2 -0.8295091 -0.6726932 0.02986499 -0.07781065 0.20440398 -0.5251603  3 -0.3787645 -0.3210074 -0.18744619 0.08680150 0.02483061 0.9324206  4 0.8226898 0.7331714 0.17108840 -0.22496625 0.18145704 0.5193217  pH DissolvedOxygen Turbidity TotalDissolvedSolids NitriteLevel NitrateLevel  1 -0.17108958 0.20646812 -0.4183163 -0.7107107 0.25563864 -0.05363636  2 0.09805321 -0.12804798 0.1209902 -0.5527599 -0.36302165 -0.06643528  3 -0.36158468 -0.01028971 -0.2416839 0.9469990 0.19928309 -0.10347830  4 0.54781548 -0.06947874 0.6815733 0.5056891 -0.06273235 0.30031307  LeadConcentration ArsenicConcentration Humidity  1 -0.114773846 -0.19411648 -0.1764593  2 -0.008148328 -0.14436154 0.2862951  3 0.011066323 -0.02123068 -0.2744147  4 0.147097941 0.49036985 0.1683012  > library(ggplot2)  > KidneyData\_clustered <- as.data.frame(KidneyData\_scaled)  > KidneyData\_clustered$cluster <- as.factor(kmeans\_result$cluster)  > ggplot(KidneyData\_clustered, aes(x = KidneyData\_clustered[,1], y = KidneyData\_clustered[,4], color = cluster)) +  + geom\_point() +  + labs(title = "K-means Clustering Results", x = "TotalDissolvedSolids", y = "Turbidity")  > print(kmeans\_result$centers)  Age BloodPressure BloodSugar Cholesterol BMI ElectricConductivity  1 0.6704254 0.5067659 0.02135505 0.17337830 -0.39206774 -0.7375164  2 -0.8295091 -0.6726932 0.02986499 -0.07781065 0.20440398 -0.5251603  3 -0.3787645 -0.3210074 -0.18744619 0.08680150 0.02483061 0.9324206  4 0.8226898 0.7331714 0.17108840 -0.22496625 0.18145704 0.5193217  pH DissolvedOxygen Turbidity TotalDissolvedSolids NitriteLevel NitrateLevel  1 -0.17108958 0.20646812 -0.4183163 -0.7107107 0.25563864 -0.05363636  2 0.09805321 -0.12804798 0.1209902 -0.5527599 -0.36302165 -0.06643528  3 -0.36158468 -0.01028971 -0.2416839 0.9469990 0.19928309 -0.10347830  4 0.54781548 -0.06947874 0.6815733 0.5056891 -0.06273235 0.30031307  LeadConcentration ArsenicConcentration Humidity  1 -0.114773846 -0.19411648 -0.1764593  2 -0.008148328 -0.14436154 0.2862951  3 0.011066323 -0.02123068 -0.2744147  4 0.147097941 0.49036985 0.1683012  > print(kmeans\_result$size)  [1] 129 144 128 99  > ggplot(KidneyData\_clustered, aes(x = KidneyData\_clustered[,1], y = KidneyData\_clustered[,4], color = cluster)) +  + geom\_point() +  + labs(title = "K-means Clustering Results", x = "X", y = "Y") |
|  |
| |  | | --- | | > | |